User's Manual

32

The revision list can be viewed directly by clicking the title page.
The revision list summarizes the locations of revisions and additions. Details should always be checked by referring to the relevant text.

SH7080 Group

User's Manual: Hardware

Renesas 32-Bit RISC Microcomputer
SuperH™ RISC engine Family

SH7083	R5F7083
	R5M7083
	R5S7083
SH7084	R5F7084
	R5M7084
	R5S7084
SH7085	R5F7085
	R5M7085
	R5S7085
SH7086	R5F7086

Renesas Electronics
www.renesas.com

Rev.6.00 Oct 2014

Notice

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

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

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

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

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

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

(2012.4)

General Precautions 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 an 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 may occur due to the false recognition of the pin state as an input signal. 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 type number, confirm that the change will not lead to problems.

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

Configuration of This Manual

This manual comprises the following items:

1. General Precautions in the Handling of MPU/MCU Products
2. Configuration of This Manual
3. Preface
4. Contents
5. Overview
6. Description of Functional Modules
   - — CPU and System-Control Modules
   - — On-Chip Peripheral Modules

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

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

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

7. List of Registers
8. Electrical Characteristics
9. Appendix
10. Main Revisions for This Edition (only for revised versions)

The list of revisions is a summary of points that have been revised or added to earlier versions. This does not include all of the revised contents. For details, see the actual locations in this manual.

11. Index

Preface

The SH7083, SH7084, SH7085, and SH7086 Group RISC (Reduced Instruction Set Computer) microcomputers include a Renesas original RISC CPU as its core, and the peripheral functions required to configure a system.

Target Users: This manual was written for users who will be using the SH7083, SH7084, SH7085, and SH7086 Group in the design of application systems. Target users are expected to understand the fundamentals of electrical circuits, logical circuits, and microcomputers.

Objective: This manual was written to explain the hardware functions and electrical characteristics of the SH7083, SH7084, SH7085, and SH7086 Group to the target users.
Refer to the SH-1/SH-2/SH-DSP Software Manual for a detailed description of the instruction set.

Notes on reading this manual:

- In order to understand the overall functions of the chip
  Read the manual according to the contents. This manual can be roughly categorized into parts on the CPU, system control functions, peripheral functions and electrical characteristics.
- In order to understand the details of the CPU's functions
  Read the SH-1/SH-2/SH-DSP Software Manual.
- In order to understand the details of a register when its name is known
  Read the index that is the final part of the manual to find the page number of the entry on the register. The addresses, bits, and initial values of the registers are summarized in section 27, List of Registers.

Examples:		
	Register name:	The following notation is used for cases when the same or a similar function, e.g. serial communication interface, is implemented on more than one channel: XXX_N (XXX is the register name and N is the channel number)
	Bit order:	The MSB is on the left and the LSB is on the right.
	Number notation:	Binary is B'xxxx, hexadecimal is H'xxxx, decimal is xxxx.
	Signal notation:	An overbar is added to a low-active signal: $\overline{\text{xxxx}}$

Related Manuals: The latest versions of all related manuals are available from our web site.
Please ensure you have the latest versions of all documents you require.
http://www.renesas.com/

SH7083, SH7084, SH7085, and SH7086 Group Manuals:

Document Title	Document No.
SH7080 Group Hardware Manual	This manual
SH-1/SH-2/SH-DSP Software Manual	REJ09B0171

User's Manuals for Development Tools:

Document Title	Document No.
SuperH™ RISC engine C/C++ Compiler, Assembler, Optimizing Linkage Editor Compiler Package V.9.04 User's Manual	REJ10J2202
High-performance Embedded Workshop User's Manual	REJ10J2169

Application Note:

Document Title	Document No.
SuperH RISC engine C/C++ Compiler Package Application Note	REJ05B0463

Contents

Figures

Tables

Section 1 Overview

1.1 Features of SH7083, SH7084, SH7085, and SH7086

This LSI is a single-chip RISC (Reduced Instruction Set Computer) microcomputer that integrates a Renesas original RISC CPU core with peripheral functions required for system configuration.

The CPU in this LSI has a RISC-type instruction set. Most instructions can be executed in one state (one system clock cycle), which greatly improves instruction execution speed. In addition, the 32-bit internal-bus architecture enhances data processing power. With this CPU, it has become possible to assemble low-cost, high-performance, and high-functioning systems, even for applications that were previously impossible with microcomputers, such as real-time control, which demands high speeds.

In addition, this LSI includes on-chip peripheral functions necessary for system configuration, such as large-capacity ROM and RAM, a direct memory access controller (DMAC), a data transfer controller (DTC), timers, a serial communication interface (SCI), a serial communication interface with FIFO (SCIF), a synchronous serial communication unit (SSU), an A/D converter, an interrupt controller (INTC), I/O ports, and an I^2C bus interface 2 (IIC2).

This LSI also provides an external memory access support function to enable direct connection to various memory devices or peripheral LSIs.

These on-chip functions significantly reduces costs of designing and manufacturing application systems.

In terms of on-chip ROM, F-ZTAT™ (Flexible Zero Turn Around Time)* version incorporating flash memory and mask ROM version are available. The flash memory can be programmed with a programmer that supports programming of this LSI, and can also be programmed and erased by software. This enables LSI chip to be re-programmed at a user-site while mounted on a board.

The features of this LSI are listed in table 1.1.

Note: * F-ZTAT™ is a trademark of Renesas Electronics Corp.

Table 1.1 Features

Items	Specification
CPU	• Central processing unit with an internal 32-bit RISC (Reduced Instruction Set Computer) architecture • Instruction length: 16-bit fixed length for improved code efficiency • Load-store architecture (basic operations are executed between registers) • Sixteen 32-bit general registers • Five-stage pipeline • On-chip multiplier: Multiplication operations (32 bits × 32 bits → 64 bits) executed in two to five cycles • C language-oriented 62 basic instructions Note: Some specifications on slot illegal instruction exception handling in this LSI differ from those of the conventional SH-2. For details, see section 5.8.4, Notes on Slot Illegal Instruction Exception Handling.
Operating modes	• Operating modes — Single chip mode — Extended ROM enabled mode — Extended ROM disabled mode • Operating states — Program execution state — Exception handling state — Bus release state • Power-down modes — Sleep mode — Software standby mode — Deep software standby mode — Module standby mode
User break controller (UBC)	• Addresses, data values, type of access, and data size can all be set as break conditions • Supports a sequential break function • Two break channels
On-chip ROM	• 256 kbytes or 512 kbytes

Items	Specification
On-chip RAM	• 16 kbytes or 32 kbytes
Bus state controller (BSC)	• Address space divided into nine areas: Eight areas (CS0 to CS7), each a maximum of 64 Mbytes, and one area (CS8) of a maximum of 1 Gbytes (a total of three areas in SH7083, eight areas in SH7084/SH7085, and nine areas in SH7086) • 8-bit external bus • 16-bit external bus • 32-bit external bus (only in SH7085/SH7086) • The following features settable for each area independently — Bus size (8, 16, or 32 bits) — Number of access wait cycles — Idle wait cycle insertion — Specifying the memory to be connected to each area enables direct connection to SRAM, SRAM with byte selection, burst ROM (clock synchronous or asynchronous), MPX-I/O, burst MPX-I/O, SDRAM, and PCMCIA • Outputs a chip select signal according to the target area
Direct memory access controller (DMAC)	• Four channels • External request available • Burst mode and cycle steal mode
Data transfer controller (DTC)	• Data transfer activated by an on-chip peripheral module interrupt can be done independently of the CPU transfer. • Transfer mode selectable for each interrupt source (transfer mode is specified in memory) • Multiple data transfer enabled for one activation source • Various transfer modes Normal mode, repeat mode, or block transfer mode can be selected. • Data transfer size can be specified as byte, word, or longword • The interrupt that activated the DTC can be issued to the CPU. A CPU interrupt can be requested after one data transfer completion. • A CPU interrupt can be requested after all specified data transfer completion.

Items	Specification
Interrupt controller (INTC)	• Nine external interrupt pins (NMI and IRQ7 to IRQ0) • On-chip peripheral interrupts: Priority level set for each module • Vector addresses: A vector address for each interrupt source
User debugging interface (H-UDI) (only in F-ZTAT version)	• E10A emulator support
Advanced user debugger (AUD) (only in F-ZTAT version supporting full functions of E10A)	• E10A emulator support
Clock pulse generator (CPG)	• Clock mode: Input clock can be selected from external input or crystal resonator • Five types of clocks generated: — CPU clock: Maximum 80 MHz — Bus clock: Maximum 40 MHz — Peripheral clock: Maximum 40 MHz — MTU2 clock: Maximum 40 MHz — MTU2S clock: Maximum 80 MHz
Watchdog timer (WDT)	• On-chip one-channel watchdog timer • Interrupt generation is supported.

Items	Specification
Multi-function timer pulse unit 2 (MTU2)	• Maximum 16 lines (maximum 13 lines in SH7083) of pulse input/output and three lines of pulse input based on six channels of 16-bit timers • 21 output compare and input capture registers • A total of 21 independent comparators • Selection of eight counter input clocks • Input capture function • Pulse output modes Toggle, PWM, complementary PWM, and reset-synchronized PWM modes • Synchronization of multiple counters • Complementary PWM output mode — Six-phase (four-phase in SH7083) non-overlapping waveforms output for inverter control — Automatic dead time setting — 0% to 100% PWM duty cycle specifiable — Output suppression — A/D conversion delaying function — Dead time compensation — Interrupt skipping at crest or trough • Reset-synchronized PWM mode Three-phase PWM waveforms in positive and negative phases can be output with a required duty cycle • Phase counting mode Two-phase encoder pulse counting available
Multi-function timer pulse unit 2S (MTU2S)	• Subset of MTU2, including channels 3 to 5 • Operating at 80 MHz max.
Port output enable (POE)	• High-impedance control of waveform output pins in MTU2 and MTU2S
Compare match timer (CMT)	• 16-bit counters • Compare match interrupts can be generated • Two channels
Serial communication interface (SCI)	• Clock synchronous or asynchronous mode • Three channels

Items	Specification
Serial communication interface with FIFO (SCIF)	• Clock synchronous or asynchronous mode • Separate 16-byte FIFO registers for transmission and reception • One channel
Synchronous serial communication unit (SSU)	• Master mode or slave mode selectable • Standard mode or bidirectional mode selectable • Transmit/receive data length can be selected from 8, 16, and 32 bits. • Full-duplex communication (transmission and reception executed simultaneously) • Consecutive serial communication • One channel
I^2C bus interface 2 (IIC2) (SH7084, SH7085, and SH7086 only)	• Conforming to Philips I^2C bus interface • Master mode and slave mode supported • Continuous transfer • I^2C bus format or clock synchronous serial format selectable • One channel
A/D converter (ADC)	• 10 bits × 8 channels (in SH7083/SH7084/SH7085) • 10 bits × 16 channels (in SH7086) • Conversion request by external triggers, MTU2, or MTU2S • Two sample-and-hold function units (two channels can be sampled simultaneously) (in SH7083/SH7084/SH7085) • Three sample-and-hold function units (three channels can be sampled simultaneously) (in SH7086)
I/O ports	• 65 general input/output pins and eight general input pins (SH7083) • 76 general input/output pins and eight general input pins (SH7084) • 100 general input/output pins and eight general input pins (SH7085) • 118 general input/output pins and 16 general input pins (SH7086) • Input or output can be selected for each bit
Packages	• TQFP1414-100 (0.5 pitch) (SH7083) • LQFP2020-112 (0.65 pitch) (SH7084) • LQFP2020-144 (0.5 pitch) (SH7085) • LQFP2424-176 (0.5 pitch) (SH7086)
Power supply voltage	• Vcc: 3.0 to 3.6 V or 4.0 to 5.5 V • AVcc: 4.0 to 5.5 V

1.2 Block Diagram

The block diagram of this LSI is shown in figure 1.1.

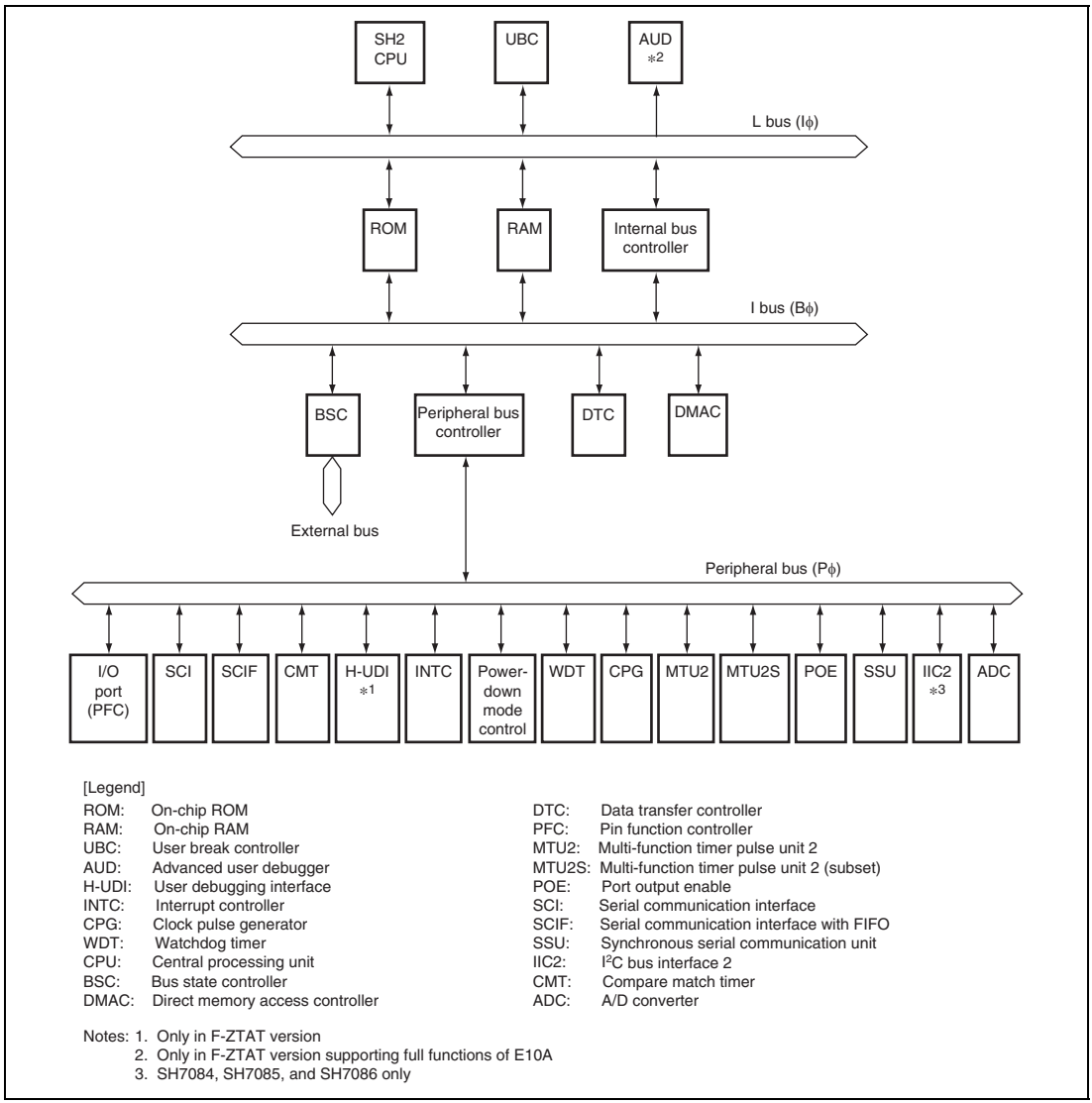
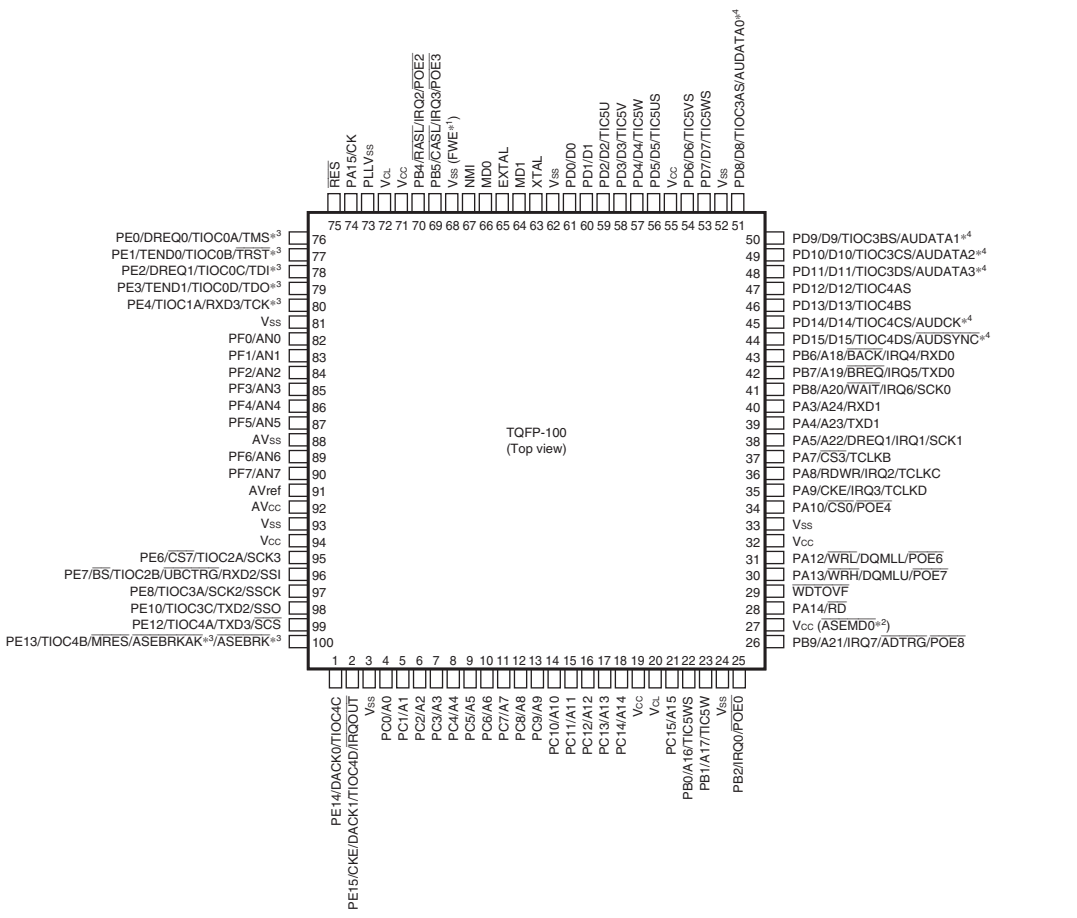


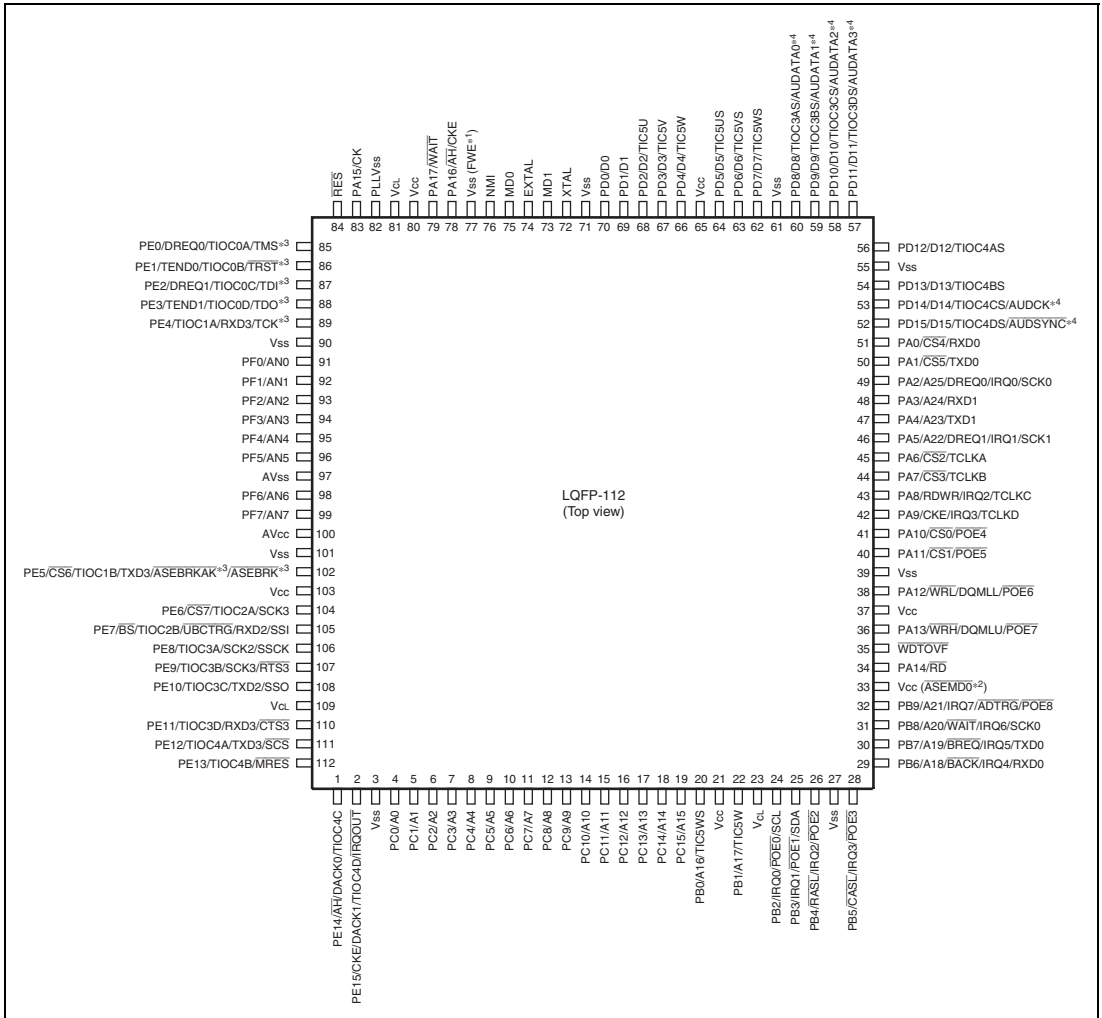
Figure 1.1 Block Diagram

1.3 Pin Assignments



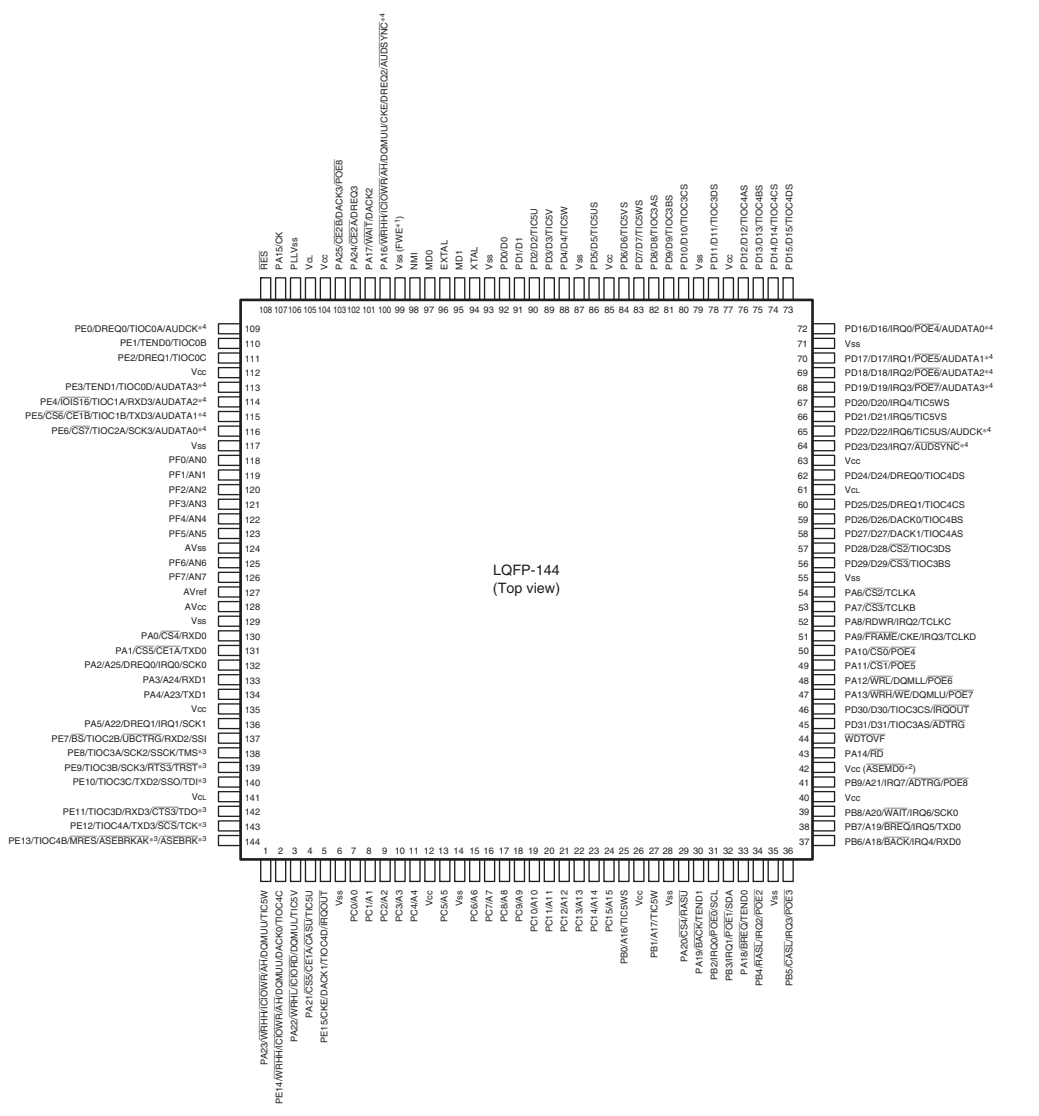
Notes: 1. This pin is fixed to V_{SS} in the mask ROM and ROM-less versions and is used as the FWE input pin in the F-ZTAT version.
2. This pin is for the E10A emulator. It is fixed to V_{CC} in the mask ROM and ROM-less versions and is used as the ASEMD0 input pin in the F-ZTAT version.
3. These pin functions are only available in the F-ZTAT version (i.e., they are not available in the mask ROM and ROM-less versions).
4. These pin functions are only available in the F-ZTAT version supporting full functions of E10A (not available in normal F-ZTAT version).

Figure 1.2 Pin Assignments of SH7083 (TQFP1414-100)



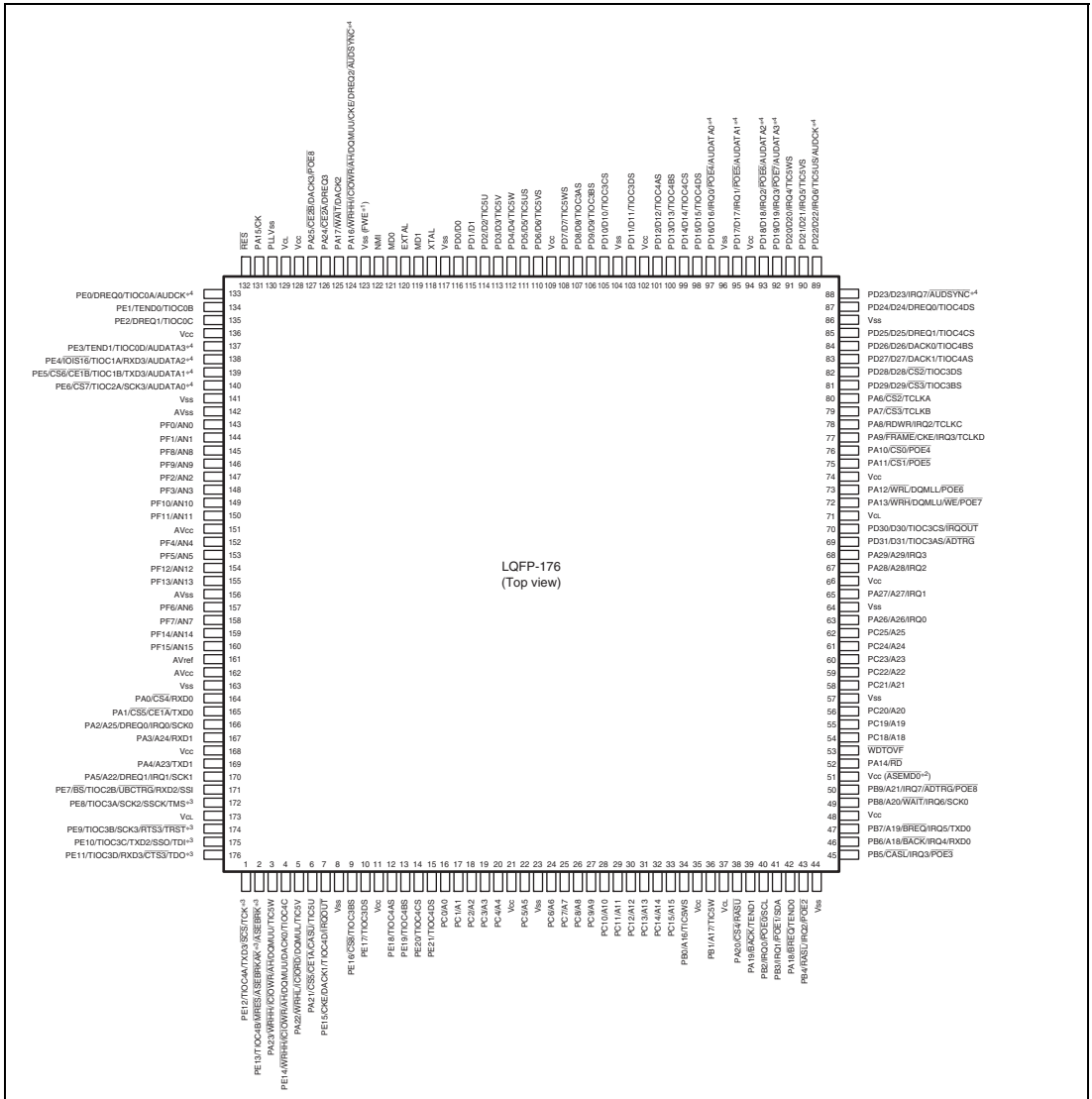
Notes: 1. This pin is fixed to Vss in the mask ROM and ROM-less versions and is used as the FWE input pin in the F-ZTAT version.
2. This pin is for the E10A emulator. It is fixed to Vcc in the mask ROM and ROM-less versions and is used as the ASEMD0 input pin in the F-ZTAT version.
3. These pin functions are only available in the F-ZTAT version (i.e., they are not available in the mask ROM and ROM-less versions).
4. These pin functions are only available in the F-ZTAT version supporting full functions of E10A (not available in normal F-ZTAT version).

Figure 1.3 Pin Assignments of SH7084



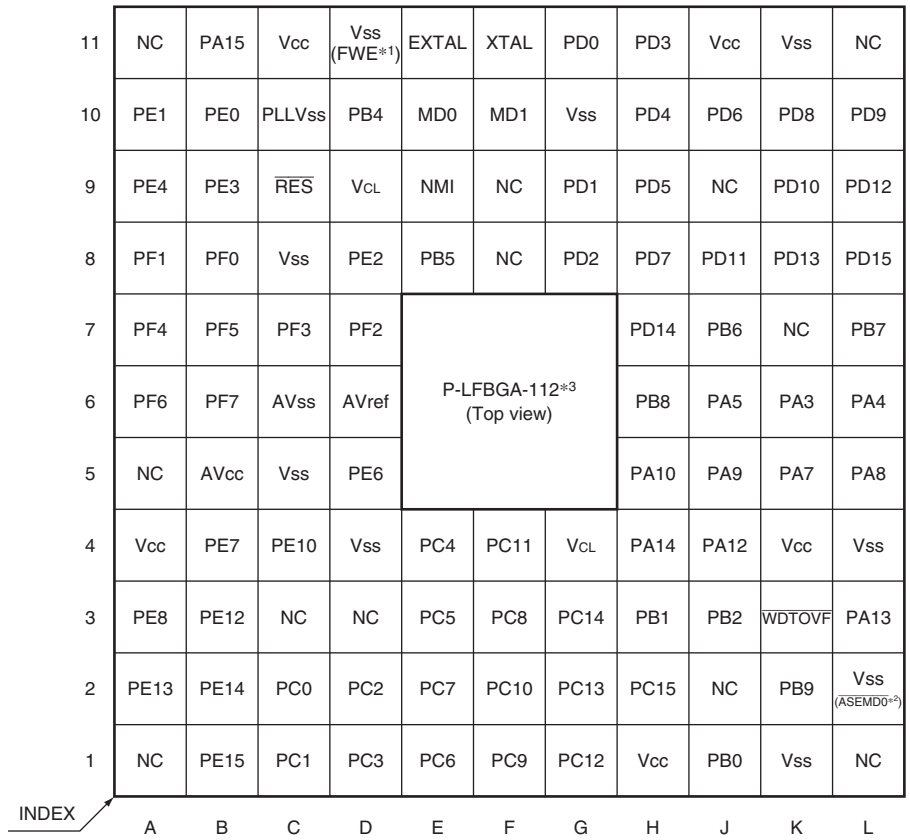
Notes: 1. This pin is fixed to Vss in the mask ROM and ROM-less versions and is used as the FWE input pin in the F-ZTAT version.
2. This pin is for the E10A emulator. It is fixed to Vcc in the mask ROM and ROM-less versions and is used as the ASEMD0 input pin in the F-ZTAT version.
3. These pin functions are only available in the F-ZTAT version (i.e., they are not available in the mask ROM and ROM-less versions).
4. These pin functions are only available in the F-ZTAT version supporting full functions of E10A (not available in normal F-ZTAT version).

Figure 1.4 Pin Assignments of SH7085



Notes: 1. This pin is fixed to Vss in the mask ROM and ROM-less versions and is used as the FWE input pin in the F-ZTAT version.
2. This pin is for the E10A emulator. It is fixed to Vcc in the mask ROM and ROM-less versions and is used as the $\overline{\text{ASEMD0}}$ input pin in the F-ZTAT version.
3. These pin functions are only available in the F-ZTAT version (i.e., they are not available in the mask ROM and ROM-less versions).
4. These pin functions are only available in the F-ZTAT version supporting full functions of E10A (not available in normal F-ZTAT version).

Figure 1.5 Pin Assignments of SH7086



Notes: 1. This pin is fixed to Vss in the mask ROM version and is used as the FWE input pin in the F-ZTAT version.
2. This pin is for the E10A emulator. It is fixed to V_{CC} in the mask ROM version and is used as the $\overline{\text{ASEMD0}}$ input pin in the F-ZTAT version.
3. The multiplexing of pin functions is the same as for the TQFP1414-100. For details on the pin-multiplexed functions, see the manual for the TQFP1414-100.

Figure 1.6 Pin Assignments of SH7083 (P-LFBGA-112)

1.4 Pin Functions

Table 1.2 summarizes the pin functions.

Table 1.2 Pin Functions

Classification	Symbol	I/O	Name	Function
Power supply	Vcc	I	Power supply	Power supply pins. Connect all Vcc pins to the system. There will be no operation if any pins are open.
	Vss	I	Ground	Ground pin. Connect all Vss pins to the system power supply (0V). There will be no operation if any pins are open.
	V_{CL}	O	Power supply for internal power-down	External capacitance pins for internal power-down power supply. Connect these pins to Vss via a 0.47 μF capacitor (placed close to the pins).
Clock	PLLVss	I	PLL ground	Ground pin for the on-chip PLL oscillator
	EXTAL	I	External clock	Connected to a crystal resonator. An external clock signal may also be input to the EXTAL pin.
	XTAL	O	Crystal	Connected to a crystal resonator.
	CK	O	System clock	Supplies the system clock to external devices.
Operating mode control	MD1, MD0	I	Mode set	Sets the operating mode. Do not change values on these pins during operation.
	FWE	I	Flash memory write enable	Pin for flash memory. Flash memory can be protected against programming or erasure through this pin.

Classification	Symbol	I/O	Name	Function
System control	$\overline{\text{RES}}$	I	Power-on reset	When low, this LSI enters the power-on reset state.
	$\overline{\text{MRES}}$	I	Manual reset	When low, this LSI enters the manual reset state.
	$\overline{\text{WDTOVF}}$	O	Watchdog timer overflow	Output signal for the watchdog timer overflow. If this pin need to be pulled down, use the resistor larger than 1 MΩ to pull this pin down.
	$\overline{\text{BREQ}}$	I	Bus-mastership request	Low when an external device requests the release of the bus mastership.
	$\overline{\text{BACK}}$	O	Bus-mastership request acknowledge	Indicates that the bus mastership has been released to an external device. Reception of the $\overline{\text{BACK}}$ signal informs the device which has output the $\overline{\text{BREQ}}$ signal that it has acquired the bus.
Interrupts	NMI	I	Non-maskable interrupt	Non-maskable interrupt request pin. Fix to high or low level when not in use.
	IRQ7 to IRQ0	I	Interrupt requests 7 to 0	Maskable interrupt request pin. Selectable as level input or edge input. The rising edge, falling edge, and both edges are selectable as edges.
	$\overline{\text{IRQOUT}}$	O	Interrupt request output	Shows that an interrupt cause has occurred. The interrupt cause can be recognized even in the bus release state.
Address bus	A29 to A0	O	Address bus	Outputs addresses. A24 to A0 are available in the SH7083. A25 to A0 are available in the SH7084/SH7085.
Data bus	D31 to D0	I/O	Data bus	32-bit bidirectional bus. D15 to D0 are available in the SH7083/SH7084.

Classification	Symbol	I/O	Name	Function
Bus control	$\overline{CS8}$ to $\overline{CS0}$	O	Chip select 8 to 0	Chip-select signal for external memory or devices. $\overline{CS7}$, $\overline{CS3}$, and $\overline{CS0}$ are available in the SH7083. $\overline{CS7}$ to $\overline{CS0}$ are available in the SH7084/SH7085.
	$\overline{RD}$	O	Read	Indicates reading of data from external devices.
	RDWR	O	Read/write	Read/write signal
	$\overline{BS}$	O	Bus start	Bus-cycle start
	$\overline{AH}$	O	Address hold	Address hold timing signal for the device that uses the address/data-multiplexed bus. Available only in the SH7084/SH7085/SH7086.
	$\overline{FRAME}$	O	FRAME signal	In burst MPX-I/O interface mode, negated before the last bus cycle to indicate that the next bus cycle is the last access. Available only in the SH7085/SH7086.
	$\overline{WRHH}$	O	Write to HH byte	Indicates a write access to bits 31 to 24 of the external data. Available only in the SH7085/SH7086.
	$\overline{WRHL}$	O	Write to HL byte	Indicates a write access to bits 23 to 16 of the external data. Available only in the SH7085/SH7086.
	$\overline{WRH}$	O	Write to upper byte	Indicates a write access to bits 15 to 8 of the external data.
	$\overline{WRL}$	O	Write to lower byte	Indicates a write access to bits 7 to 0 of the external data.
	$\overline{WAIT}$	I	Wait	Input signal for inserting a wait cycle into the bus cycles during access to the external space.

Classification	Symbol	I/O	Name	Function
Bus control	$\overline{\text{RASL}}$	O	RAS	Connected to the $\overline{\text{RAS}}$ pin of the SDRAM.
	$\overline{\text{RASU}}$	O	RAS	Connected to the $\overline{\text{RAS}}$ pin of the SDRAM. Available only in the SH7085/SH7086.
	$\overline{\text{CASL}}$	O	CAS	Connected to the $\overline{\text{CAS}}$ pin of the SDRAM.
	$\overline{\text{CASU}}$	O	CAS	Connected to the $\overline{\text{CAS}}$ pin of the SDRAM. Available only in the SH7085/SH7086.
	CKE	O	Clock enable	Connected to the CKE pin of the SDRAM.
	DQMUU	O	HH byte selection	Selects bits 31 to 24 of the SDRAM data bus. Available only in the SH7085/SH7086.
	DQMUL	O	HL byte selection	Selects bits 23 to 16 of the SDRAM data bus. Available only in the SH7085/SH7086.
	DQMLU	O	Upper byte selection	Selects bits 15 to 8 of the SDRAM data bus.
	DQMLL	O	Lower byte selection	Selects bits 7 to 0 of the SDRAM data bus.
	$\overline{\text{CE1A}}$	O	Lower byte selection for PCMCIA card	Chip enable for PCMCIA connected to area 5 Available only in the SH7085/SH7086.
	$\overline{\text{CE1B}}$	O	Lower byte selection for PCMCIA card	Chip enable for PCMCIA connected to area 6 Available only in the SH7085/SH7086.

Classification	Symbol	I/O	Name	Function
Bus control	$\overline{\text{CE2A}}$	O	Upper byte selection for PCMCIA card	Chip enable for PCMCIA connected to area 5 Available only in the SH7085/SH7086.
	$\overline{\text{CE2B}}$	O	Upper byte selection for PCMCIA card	Chip enable for PCMCIA connected to area 6 Available only in the SH7085/SH7086.
	$\overline{\text{ICIOWR}}$	O	Write strobe for PCMCIA I/O	Connected to the PCMCIA I/O write strobe signal Available only in the SH7085/SH7086.
	$\overline{\text{ICIORD}}$	O	Read strobe for PCMCIA I/O	Connected to the PCMCIA I/O read strobe signal Available only in the SH7085/SH7086.
	$\overline{\text{WE}}$	O	Write strobe for PCMCIA memory	Connected to the PCMCIA memory write strobe signal Available only in the SH7085/SH7086.
	$\overline{\text{IOIS16}}$	I	PCMCIA dynamic bus sizing	Indicates 16-bit I/O for PCMCIA in little endian mode. This LSI does not support little endian and this pin must be held low. Available only in the SH7085/SH7086.
Direct memory access controller (DMAC)	DREQ3 to DREQ0	I	DMA-transfer request	Input pins to receive external requests for DMA transfer. Only DREQ1 and DREQ0 are available in the SH7083/SH7084.
	DACK3 to DACK0	O	DMA-transfer strobe	Strobe signal output pins for the external device that has requested DMA transfer. Only DACK1 and DACK0 are available in the SH7083/SH7084.
	TEND1, TEND0	O	DMA-transfer end	Output pins for DMA transfer end signals

Classification	Symbol	I/O	Name	Function
Multi function timer-pulse unit 2 (MTU2)	TCLKA, TCLKB, TCLKC, TCLKD	I	MTU2 timer clock input	External clock input pins for the timer. Only TCLKB, TCLKC, and TCLKD are available in the SH7083.
	TIOC0A, TIOC0B, TIOC0C, TIOC0D	I/O	MTU2 input capture/output compare (channel 0)	The TGRA_0 to TGRD_0 input capture input/output compare output/PWM output pins.
	TIOC1A, TIOC1B	I/O	MTU2 input capture/output compare (channel 1)	The TGRA_1 to TGRB_1 input capture input/output compare output/PWM output pins. Only TIOC1A is available in the SH7083.
	TIOC2A, TIOC2B	I/O	MTU2 input capture/output compare (channel 2)	The TGRA_2 to TGRB_2 input capture input/output compare output/PWM output pins.
	TIOC3A, TIOC3B, TIOC3C, TIOC3D	I/O	MTU2 input capture/output compare (channel 3)	The TGRA_3 to TGRD_3 input capture input/output compare output/PWM output pins. Only TIOC3A and TIOC3C are available in the SH7083.
	TIOC4A, TIOC4B, TIOC4C, TIOC4D	I/O	MTU2 input capture/output compare (channel 4)	The TGRA_4 to TGRD_4 input capture input/output compare output/PWM output pins.
	TIC5U, TIC5V, TIC5W	I	MTU2 input capture (channel 5)	The TGRU_5, TGRV_5, and TGRW_5 input capture input pins.
Multi function timer-pulse unit 2S (MTU2S)	TIOC3AS, TIOC3BS, TIOC3CS, TIOC3DS	I/O	MTU2S input capture/output compare (channel 3)	The TGRA_3S to TGRD_3S input capture input/output compare output/PWM output pins.
	TIOC4AS, TIOC4BS, TIOC4CS, TIOC4DS	I/O	MTU2S input capture/output compare (channel 4)	The TGRA_4S to TGRD_4S input capture input/output compare output/PWM output pins.

Classification	Symbol	I/O	Name	Function
Multi function timer-pulse unit 2S (MTU2S)	TIC5US, TIC5VS, TIC5WS	I	MTU2S input capture (channel 5)	The TGRU_5S, TGRV_5S, and TGRW_5S input capture input pins.
Port output enable (POE)	$\overline{\text{POE8}}$ to $\overline{\text{POE0}}$	I	Port output enable	Request signal input to place the MTU2 and MTU2S waveform output pins in high impedance state. $\overline{\text{POE8}}$ to $\overline{\text{POE6}}$, $\overline{\text{POE4}}$ to $\overline{\text{POE2}}$, and $\overline{\text{POE0}}$ are available in the SH7083.
Serial communication interface (SCI)	TXD2 to TXD0	O	Transmit data	Transmit data output pins
	RXD2 to RXD0	I	Receive data	Receive data input pins
	SCK2 to SCK0	I/O	Serial clock	Clock input/output pins
Serial communication interface with FIFO (SCIF)	TXD3	O	Transmit data	Transmit data output pin
	RXD3	I	Receive data	Receive data input pin
	SCK3	I/O	Serial clock	Clock input/output pin
	$\overline{\text{RTS3}}$	O	Request to send	Modem control pin. This pin is not available in the SH7083.
	$\overline{\text{CTS3}}$	I	Clear to send	Modem control pin. This pin is not available in the SH7083.
Synchronous serial communication unit (SSU)	SSO	I/O	Data	Data input/output pin.
	SSI	I/O	Data	Data input/output pin.
	SSCK	I/O	Clock	Clock input/output pin.
	$\overline{\text{SCS}}$	I/O	Chip select	Chip select input/output pin.
I^2C bus interface 2 (IIC2)	SCL	I/O	I^2C clock input/output	Clock input/output pin for I^2C bus. Available only in the SH7084/SH7085/SH7086.
	SDA	I/O	I^2C data input/output	Data input/output pin for I^2C bus. Available only in the SH7084/SH7085/SH7086.

Classification	Symbol	I/O	Name	Function
A/D converter (ADC)	AN15 to AN0	I	Analog input pins	Analog input pins. AN7 to AN0 are available in the SH7083/SH7084/SH7085.
	$\overline{\text{ADTRG}}$	I	A/D conversion trigger input	External trigger input pin for starting A/D conversion.
	AVref	I	Analog reference power supply	Reference voltage pin for the A/D converter. Available only in the SH7083/SH7085/SH7086. (In the SH7084, this pin is connected to AVcc inside this LSI.)
	AVcc	I	Analog power supply	Power supply pin for the A/D converter. Connect all AVcc pins to the system power supply (Vcc) when the A/D converter is not used. The A/D converter does not work if any pin is open.
	AVss	I	Analog ground	Ground pin for the A/D converter. Connect it to the system ground (0 V). Connect all AVss pins to the system ground (0 V) correctly. The A/D converter does not work if any pin is open.
I/O ports	PA29 to PA0	I/O	General port	30-bit general input/output port pins. PA15 to PA12, PA10 to PA7, and PA5 to PA3 are available in the SH7083. PA17 to PA0 are available in the SH7084. PA25 to PA0 are available in the SH7085.
	PB9 to PB0	I/O	General port	10-bit general input/output port pins. PB9 to PB4 and PB2 to PB0 are available in the SH7083.
	PC25 to PC18, PC15 to PC0	I/O	General port	24-bit general input/output port pins. PC15 to PC0 are available in the SH7083/SH7084/SH7085.

Classification	Symbol	I/O	Name	Function
I/O ports	PD31 to PD0	I/O	General port	32-bit general input/output port pins. PD15 to PD0 are available in the SH7083/SH7084.
	PE21 to PE0	I/O	General port	22-bit general input/output port pins. PE15 to PE12, PE10, PE8 to PE6, and PE4 to PE0 are available in the SH7083. PE15 to PE0 are available in the SH7084/SH7085.
	PF15 to PF0	I	General port	16-bit general input port pins. PF7 to PF0 are available in the SH7083/SH7084/SH7085.
User break controller (UBC)	$\overline{\text{UBCTRG}}$	O	User break trigger output	Trigger output pin for UBC condition match.
User debugging interface (H-UDI) (only in the F-ZTAT version)	TCK	I	Test clock	Test-clock input pin.
	TMS	I	Test mode select	Inputs the test-mode select signal.
	TDI	I	Test data input	Serial input pin for instructions and data.
	TDO	O	Test data output	Serial output pin for instructions and data.
	$\overline{\text{TRST}}$	I	Test reset	Initialization-signal input pin.
Advanced user debugger (AUD) (only in F-ZTAT version supporting full functions of E10A)	AUDATA3 to AUDATA0	O	AUD data	Branch destination address output pins.
	AUDCK	O	AUD clock	Sync-clock output pin.
	$\overline{\text{AUDSYNC}}$	O	AUD sync signal	Data start-position acknowledge-signal output pin.

Classification	Symbol	I/O	Name	Function
E10A interface (only in the F-ZTAT version)	$\overline{\text{ASEMD0}}$	I	ASE mode	Sets the ASE mode. When a low level is input, this LSI enters ASE mode. When a high level is input, this LSI enters the normal mode. The emulator functions are available in ASE mode. When no signal is input, this pin is pulled up inside this LSI.
	$\overline{\text{ASEBRK}}$	I	Break request	E10A emulator break input pin.
	$\overline{\text{ASEBRKAK}}$	O	Break mode acknowledge	Indicates that the E10A emulator has entered its break mode.

Note: The $\overline{\text{WDTOVF}}$ pin should not be pulled down. When absolutely necessary, pull it down through a resistor of 1 MΩ or larger.

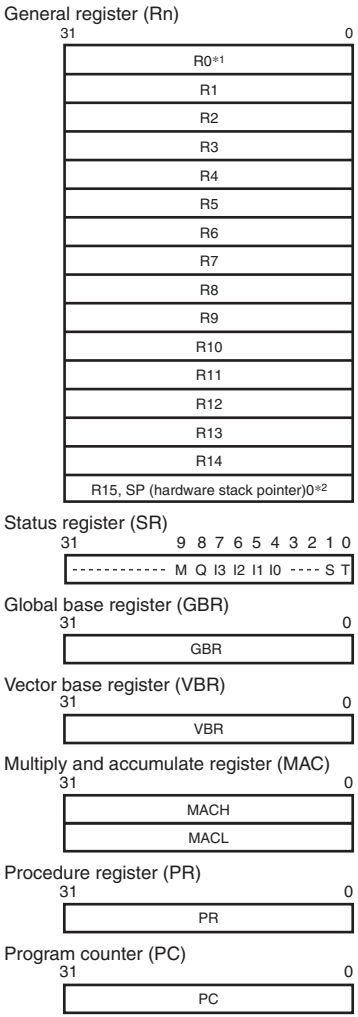
Section 2 CPU

2.1 Features

- General registers: 32-bit register × 16
- Basic instructions: 62
- Addressing modes: 11
  Register direct (Rn)
  Register indirect (@Rn)
  Post-increment register indirect (@Rn+)
  Pre-decrement register indirect (@-Rn)
  Register indirect with displacement (@disp:4, Rn)
  Index register indirect (@R0, Rn)
  GBR indirect with displacement (@disp:8, GBR)
  Index GBR indirect (@R0, GBR)
  PC relative with displacement (@disp:8, PC)
  PC relative (disp:8/disp:12/Rn)
  Immediate (#imm:8)

2.2 Register Configuration

There are three types of registers: general registers (32-bit × 16), control registers (32-bit × 3), and system registers (32-bit × 4).



Notes: 1. R0 can be used as an index register in index register indirect or index GBR indirect addressing mode. For some instructions, only R0 is used as the source or destination register.
2. R15 is used as a hardware stack pointer during exception handling.

Figure 2.1 CPU Internal Register Configuration

2.2.1 General Registers (Rn)

There are sixteen 32-bit general registers (Rn), designated R0 to R15. The general registers are used for data processing and address calculation. R0 is also used as an index register. With a number of instructions, R0 is the only register that can be used. R15 is used as a hardware stack pointer (SP). In exception handling, R15 is used for accessing the stack to save or restore the status register (SR) and program counter (PC) values.

2.2.2 Control Registers

There are three 32-bit control registers, designated status register (SR), global base register (GBR), and vector base register (VBR). SR indicates a processing state. GBR is used as a base address in GBR indirect addressing mode for data transfer of on-chip peripheral module registers. VBR is used as a base address of the exception handling (including interrupts) vector table.

- Status register (SR)

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	M	Q	I[3:0]				-	-	S	T
Initial value:	0	0	0	0	0	0	-	-	1	1	1	1	0	0	-	-
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit name	Default	Read/ Write	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	M	Undefined	R/W	Used by the DIV0U, DIV0S, and DIV1 instructions.
8	Q	Undefined	R/W	Used by the DIV0U, DIV0S, and DIV1 instructions.
7 to 4	I[3:0]	1111	R/W	Interrupt Mask
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit name	Default	Read/ Write	Description
1	S	Undefined	R/W	S Bit Used by the multiply and accumulate instruction.
0	T	Undefined	R/W	T Bit Indicates true (1) or false (0) in the following instructions: MOVT, CMP/cond, TAS, TST, BT (BT/S), BF (BF/S), SETT, CLRT Indicates carry, borrow, overflow, or underflow in the following instructions: ADDV, ADDC, SUBV, SUBC, NEGC, DIV0U, DIV0S, DIV1, SHAR, SHAL, SHLR, SHLL, ROTR, ROTL, ROTCR, ROTCL

- Global-base register (GBR)

  This register indicates a base address in GBR indirect addressing mode. The GBR indirect addressing mode is used for data transfer of the on-chip peripheral module registers and logic operations.
- Vector-base register (VBR)

  This register indicates the base address of the exception handling vector table.

2.2.3 System Registers

There are four 32-bit system registers, designated two multiply and accumulate registers (MACH and MACL), a procedure register (PR), and program counter (PC).

- Multiply and accumulate registers (MACH and MACL)
  This register stores the results of multiplication and multiply-and-accumulate operation.
- Procedure register (PR)
  This register stores the return-destination address from subroutine procedures.
- Program counter (PC)
  The PC indicates the point which is four bytes (two instructions) after the current execution instruction.

2.2.4 Initial Values of Registers

Table 2.1 lists the initial values of registers after a reset.

Table 2.1 Initial Values of Registers

Type of register	Register	Default
General register	R0 to R14	Undefined
	R15 (SP)	SP value set in the exception handling vector table
Control register	SR	I3 to I0: 1111 (H'F) Reserved bits: 0 Other bits: Undefined
	GBR	Undefined
	VBR	H'00000000
System register	MACH, MACL, PR	Undefined
	PC	PC value set in the exception handling vector table

2.3 Data Formats

2.3.1 Register Data Format

The size of register operands is always longwords (32 bits). When loading byte (8 bits) or word (16 bits) data in memory into a register, the data is sign-extended to longword and stored in the register.

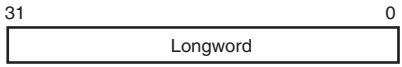


Figure 2.2 Register Data Format

2.3.2 Memory Data Formats

Memory data formats are classified into bytes, words, and longwords. Byte data can be accessed from any address. Locate, however, word data at an address 2n, longword data at 4n. Otherwise, an address error will occur if an attempt is made to access word data starting from an address other than 2n or longword data starting from an address other than 4n. In such cases, the data accessed cannot be guaranteed. The hardware stack area, pointed by the hardware stack pointer (SP, R15), uses only longword data starting from address 4n because this area holds the program counter and status register.

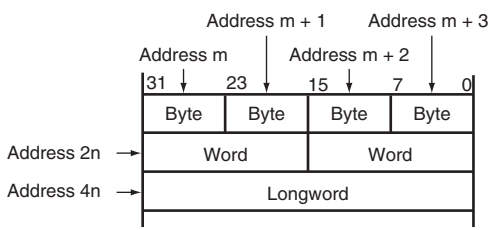


Figure 2.3 Memory Data Format

2.3.3 Immediate Data Formats

Immediate data of eight bits is placed in the instruction code.

For the MOV, ADD, and CMP/EQ instructions, the immediate data is sign-extended to longword and then calculated. For the TST, AND, OR, and XOR instructions, the immediate data is zero-extended to longword and then calculated. Thus, if the immediate data is used for the AND instruction, the upper 24 bits in the destination register are always cleared.

The immediate data of word or longword is not placed in the instruction code. It is placed in a table in memory. The table in memory is accessed by the MOV immediate data instruction in PC relative addressing mode with displacement.

2.4 Features of Instructions

2.4.1 RISC Type

The instructions are RISC-type instructions with the following features:

Fixed 16-Bit Length: All instructions have a fixed length of 16 bits. This improves program code efficiency.

One Instruction per Cycle: Since pipelining is used, basic instructions can be executed in one cycle.

Data Size: The basic data size for operations is longword. Byte, word, or longword can be selected as the memory access size. Byte or word data in memory is sign-extended to longword and then calculated. Immediate data is sign-extended to longword for arithmetic operations or zero-extended to longword size for logical operations.

Table 2.2 Word Data Sign Extension

CPU in this LSI		Description	Example of Other CPUs
MOV.W	@(disp,PC),R1	Sign-extended to 32 bits, R1 becomes H'00001234, and is then operated on by the ADD instruction.	ADD.W #H'1234,R0
ADD	R1,R0		
	.....		
.DATA.W	H'1234		

Note: Immediate data is accessed by @(disp,PC).

Load/Store Architecture: Basic operations are executed between registers. In operations involving memory, data is first loaded into a register (load/store architecture). However, bit manipulation instructions such as AND are executed directly in memory.

Delayed Branching: Unconditional branch instructions means the delayed branch instructions. With a delayed branch instruction, the branch is made after execution of the instruction immediately following the delayed branch instruction. This minimizes disruption of the pipeline when a branch is made. The conditional branch instructions have two types of instructions: conditional branch instructions and delayed branch instructions.

Table 2.3 Delayed Branch Instructions

CPU in this LSI		Description	Example of Other CPUs	
BRA	TRGET	ADD is executed before branch to TRGET.	ADD.W	R1,R0
ADD	R1,R0		BRA	TRGET

Multiply/Multiply-and-Accumulate Operations: A $16 \times 16 \rightarrow 32$ multiply operation is executed in one to two cycles, and a $16 \times 16 + 64 \rightarrow 64$ multiply-and-accumulate operation in two to three cycles. A $32 \times 32 \rightarrow 64$ multiply operation and a $32 \times 32 + 64 \rightarrow 64$ multiply-and-accumulate operation are each executed in two to four cycles.

T Bit: The result of a comparison is indicated by the T bit in SR, and a conditional branch is performed according to whether the result is True or False. Processing speed has been improved by keeping the number of instructions that modify the T bit to a minimum.

Table 2.4 T Bit

CPU in this LSI		Description	Example of Other CPUs	
CMP/GE	R1,R0	When R0 ≥ R1, the T bit is set.	CMP.W	R1,R0
BT	TRGET0	When R0 ≥ R1, a branch is made to TRGET0.	BGE	TRGET0
BF	TRGET1	When R0 < R1, a branch is made to TRGET1.	BLT	TRGET1
ADD	#–1,R0	The T bit is not changed by ADD.	SUB.W	#1,R0
CMP/EQ	#0,R0	When R0 = 0, the T bit is set.	BEQ	TRGET
BT	TRGET	A branch is made when R0 = 0.		

Immediate Data: 8-bit immediate data is placed in the instruction code. Word and longword immediate data is not placed in the instruction code. It is placed in a table in memory. The table in memory is accessed with the MOV immediate data instruction using PC relative addressing mode with displacement.

Table 2.5 Access to Immediate Data

Type	This LSI's CPU	Example of Other CPU
8-bit immediate	MOV #H'12,R0	MOV.B #H'12,R0
16-bit immediate	MOV.W @(disp,PC),R0 ..... .DATA.W H'1234	MOV.W #H'1234,R0
32-bit immediate	MOV.L @(disp,PC),R0 ..... .DATA.L H'12345678	MOV.L #H'12345678,R0

Note: Immediate data is accessed by @(disp,PC).

Absolute Addresses: When data is accessed by absolute address, place the absolute address value in a table in memory beforehand. The absolute address value is transferred to a register using the method whereby immediate data is loaded when an instruction is executed, and the data is accessed using the register indirect addressing mode.

Table 2.6 Access to Absolute Address

Type	CPU in this LSI	Example of Other CPUs
Absolute address	MOV.L @(disp,PC),R1 MOV.B @R1,R0 ..... .DATA.L H'12345678	MOV.B @H'12345678,R0

Note: Immediate data is referenced by @(disp,PC).

16-Bit/32-Bit Displacement: When data is accessed using the 16- or 32-bit displacement addressing mode, the displacement value is placed in a table in memory beforehand. Using the method whereby immediate data is loaded when an instruction is executed, this value is transferred to a register and the data is accessed using index register indirect addressing mode.

Table 2.7 Access with Displacement

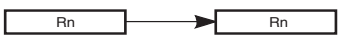
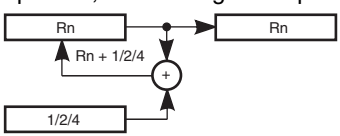
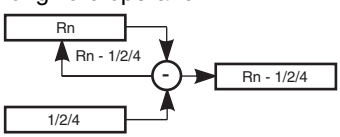
Type	CPU in this LSI	Example of Other CPUs
16-bit displacement	MOV.W @(disp,PC),R0 MOV.W @(R0,R1),R2 ..... .DATA.W H'1234	MOV.W @(H'1234,R1),R2

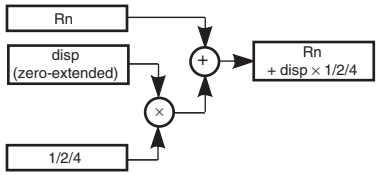
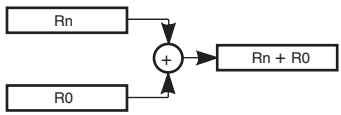
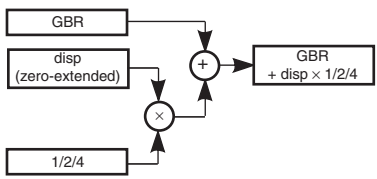
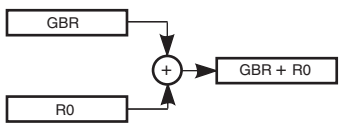
Note: Immediate data is referenced by @(disp,PC).

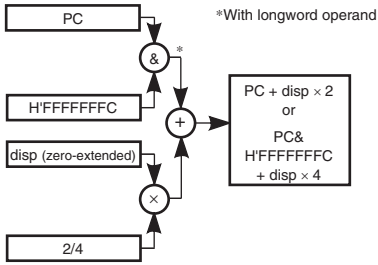
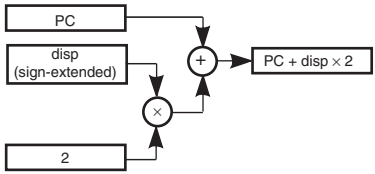
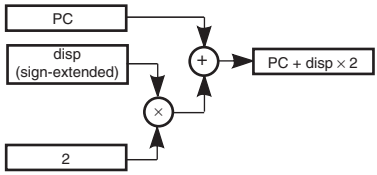
2.4.2 Addressing Modes

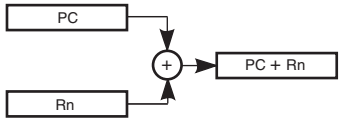
Table 2.8 lists addressing modes and effective address calculation methods.

Table 2.8 Addressing Modes and Effective Addresses

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
Register direct	Rn	Effective address is register Rn. (Operand is register Rn contents.)	—
Register indirect	@Rn	Effective address is register Rn contents. 	Rn
Register indirect with post-increment	@Rn+	Effective address is register Rn contents. A constant is added to Rn after instruction execution: 1 for a byte operand, 2 for a word operand, 4 for a longword operand. 	Rn After instruction execution Byte: Rn + 1 → Rn Word: Rn + 2 → Rn Longword: Rn + 4 → Rn
Register indirect with pre-decrement	@–Rn	Effective address is register Rn contents, decremented by a constant beforehand: 1 for a byte operand, 2 for a word operand, 4 for a longword operand. 	Byte: Rn – 1 → Rn Word: Rn – 2 → Rn Longword: Rn – 4 → Rn (Instruction executed with Rn after calculation)

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
Register indirect with displacement	@(disp:4, Rn)	Effective address is register Rn contents with 4-bit displacement disp added. After disp is zero-extended, it is multiplied by 1 (byte), 2 (word), or 4 (longword), according to the operand size. 	Byte: Rn + disp Word: Rn + disp × 2 Longword: Rn + disp × 4
Index register indirect	@(R0, Rn)	Effective address is sum of register Rn and R0 contents. 	Rn + R0
GBR indirect with displacement	@(disp:8, GBR)	Effective address is register GBR contents with 8-bit displacement disp added. After disp is zero-extended, it is multiplied by 1 (byte), 2 (word), or 4 (longword), according to the operand size. 	Byte: GBR + disp Word: GBR + disp × 2 Longword: GBR + disp × 4
Index GBR indirect	@(R0, GBR)	Effective address is sum of register GBR and R0 contents. 	GBR + R0

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
PC relative with displacement	@(disp:8, PC)	Effective address is PC with 8-bit displacement disp added. After disp is zero-extended, it is multiplied by 2 (word) or 4 (longword), according to the operand size. With a longword operand, the lower 2 bits of PC are masked. 	Word: PC + disp × 2 Longword: PC&H'FFFFFFFC + disp × 4
PC relative	disp:8	Effective address is PC with 8-bit displacement disp added after being sign-extended and multiplied by 2. 	PC + disp × 2
	disp:12	Effective address is PC with 12-bit displacement disp added after being sign-extended and multiplied by 2. 	PC + disp × 2

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

2.4.3 Instruction Formats

This section describes the instruction formats, and the meaning of the source and destination operands. The meaning of the operands depends on the instruction code. The following symbols are used in the table.

`xxxx`: Instruction code

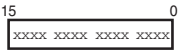
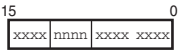
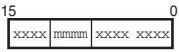
`mmmm`: Source register

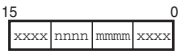
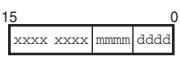
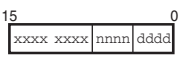
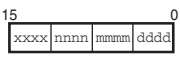
`nnnn`: Destination register

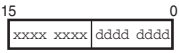
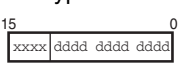
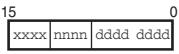
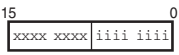
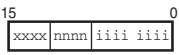
`iiii`: Immediate data

`dddd`: Displacement

Table 2.9 Instruction Formats

Instruction Format	Source Operand	Destination Operand	Sample Instruction
0 type 	—	—	NOP
n type 	—	nnnn: register direct	MOVT Rn
	Control register or system register	nnnn: register direct	STS MACH,Rn
	Control register or system register	nnnn: pre-decrement register indirect	STC.L SR,@-Rn
m type 	mmmm: register direct	Control register or system register	LDC Rm,SR
	mmmm: post-increment register indirect	Control register or system register	LDC.L @Rm+,SR
	mmmm: register indirect	—	JMP @Rm
	PC relative using Rm	—	BRAF Rm

Instruction Format	Source Operand	Destination Operand	Sample Instruction
nm type 	mmmm: register direct	nnnn: register direct	ADD Rm,Rn
	mmmm: register direct	nnnn: register indirect	MOV.L Rm,@Rn
	mmmm: post-increment register indirect (multiply-and-accumulate operation) nnnn: * post-increment register indirect (multiply-and-accumulate operation)	MACH, MACL	MAC.W @Rm+,@Rn+
	mmmm: post-increment register indirect	nnnn: register direct	MOV.L @Rm+,Rn
	mmmm: register direct	nnnn: pre-decrement register indirect	MOV.L Rm,@-Rn
	mmmm: register direct	nnnn: index register indirect	MOV.L Rm,@(R0,Rn)
md type 	mmmmdddd: register indirect with displacement	R0 (register direct)	MOV.B @(disp,Rm),R0
nd4 type 	R0 (register direct)	nnnndddd: register indirect with displacement	MOV.B R0,@(disp,Rn)
nmd type 	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

Instruction Format	Source Operand	Destination Operand	Sample Instruction
d type 	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: PC relative	BF label
d12 type 	—	ddddddddddddd: PC relative	BRA label (label=disp+PC)
nd8 type 	dddddddd: PC relative with displacement	nnnn: register direct	MOV.L @(disp,PC),Rn
i type 	iiiiiiii: immediate	Index GBR indirect	AND.B #imm,@(R0,GBR)
	iiiiiiii: immediate	R0 (register direct)	AND #imm,R0
	iiiiiiii: immediate	—	TRAPA #imm
ni type 	iiiiiiii: immediate	nnnn: register direct	ADD #imm,Rn

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

2.5 Instruction Set

2.5.1 Instruction Set by Type

Table 2.10 lists the instructions classified by type.

Table 2.10 Instruction Types

Type	Kinds of Instruction	Op Code	Function	Number of Instructions
Data transfer instructions	5	MOV	Data transfer Immediate data transfer Peripheral module data transfer Structure data transfer	39
		MOVA	Effective address transfer	
		MOVT	T bit transfer	
		SWAP	Upper/lower swap	
		XTRCT	Extraction of middle of linked registers	
Arithmetic operation instructions	21	ADD	Binary addition	33
		ADDC	Binary addition with carry	
		ADDV	Binary addition with overflow	
		CMP/cond	Comparison	
		DIV1	Division	
		DIV0S	Signed division initialization	
		DIV0U	Unsigned division initialization	
		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	
		MUL	Double-precision multiplication	

Type	Kinds of Instruction	Op Code	Function	Number of Instructions
Arithmetic operation instructions	21	MULS	Signed multiplication	33
		MULU	Unsigned multiplication	
		NEG	Sign inversion	
		NEGC	Sign inversion with borrow	
		SUB	Binary subtraction	
		SUBC	Binary subtraction with carry	
		SUBV	Binary subtraction with underflow	
Logic operation instructions	6	AND	Logical AND	14
		NOT	Bit inversion	
		OR	Logical OR	
		TAS	Memory test and bit setting	
		TST	T bit setting for logical AND	
		XOR	Exclusive logical OR	
Shift instructions	10	ROTL	1-bit left shift	14
		ROTR	1-bit right shift	
		ROTCL	1-bit left shift with T bit	
		ROTCR	1-bit right shift with T bit	
		SHAL	Arithmetic 1-bit left shift	
		SHAR	Arithmetic 1-bit right shift	
		SHLL	Logical 1-bit left shift	
		SHLLn	Logical n-bit left shift	
		SHLR	Logical 1-bit right shift	
		SHLRn	Logical n-bit right shift	

Type	Kinds of Instruction	Op Code	Function	Number of Instructions
Branch instructions	9	BF	Conditional branch, delayed conditional branch (T = 0)	11
		BT	Conditional branch, delayed conditional branch (T = 1)	
		BRA	Unconditional branch	
		BRAF	Unconditional branch	
		BSR	Branch to subroutine procedure	
		BSRF	Branch to subroutine procedure	
		JMP	Unconditional branch	
		JSR	Branch to subroutine procedure	
		RTS	Return from subroutine procedure	
System control instructions	11	CLRT	T bit clear	31
		CLRMAC	MAC register clear	
		LDC	Load into control register	
		LDS	Load into system register	
		NOP	No operation	
		RTE	Return from exception handling	
		SETT	T bit setting	
		SLEEP	Transition to power-down mode	
		STC	Store from control register	
		STS	Store from system register	
		TRAPA	Trap exception handling	
Total:	62			142

The instruction code, operation, and execution cycles of the instructions are listed in the following tables, classified by type.

Instruction	Instruction Code	Summary of Operation	Execution Cycles	T Bit
Indicated by mnemonic.	Indicated in MSB ↔ LSB order.	Indicates summary of operation.	Value when no wait cycles are inserted*1	Value of T bit after instruction is executed Explanation of Symbols —: No change
Explanation of Symbols 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	Explanation of Symbols mmmm: Source register nnnn: Destination register 0000: R0 0001: R1 ····· 1111: R15 iiii: Immediate data dddd: Displacement	Explanation of Symbols →, ←: 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		

Notes: 1. The table shows the minimum number of execution states. In practice, the number of instruction execution states will be increased in cases such as the following:
- When there is contention between an instruction fetch and a data access
- When the destination register of a load instruction (memory → register) is also used by the following instruction

2. Scaled (×1, ×2, or ×4) according to the instruction operand size, etc.
For details, see SH-1/SH-2/SH-DSP Software Manual.

2.5.2 Data Transfer Instructions

Table 2.11 Data Transfer Instructions

Instruction		Operation	Code	Execution Cycles	T Bit
MOV	#imm,Rn	imm → Sign extension → Rn	1110nnnniiiiiiii	1	—
MOV.W	@(disp,PC),Rn	(disp × 2 + PC) → Sign extension → Rn	1001nnnnddddddddd	1	—
MOV.L	@(disp,PC),Rn	(disp × 4 + PC) → Rn	1101nnnnddddddddd	1	—
MOV	Rm,Rn	Rm → Rn	0110nnnnmmmm0011	1	—
MOV.B	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0000	1	—
MOV.W	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0001	1	—
MOV.L	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0010	1	—
MOV.B	@Rm,Rn	(Rm) → Sign extension → Rn	0110nnnnmmmm0000	1	—
MOV.W	@Rm,Rn	(Rm) → Sign extension → Rn	0110nnnnmmmm0001	1	—
MOV.L	@Rm,Rn	(Rm) → Rn	0110nnnnmmmm0010	1	—
MOV.B	Rm,@–Rn	Rn–1 → Rn, Rm → (Rn)	0010nnnnmmmm0100	1	—
MOV.W	Rm,@–Rn	Rn–2 → Rn, Rm → (Rn)	0010nnnnmmmm0101	1	—
MOV.L	Rm,@–Rn	Rn–4 → Rn, Rm → (Rn)	0010nnnnmmmm0110	1	—
MOV.B	@Rm+,Rn	(Rm) → Sign extension → Rn, Rm + 1 → Rm	0110nnnnmmmm0100	1	—
MOV.W	@Rm+,Rn	(Rm) → Sign extension → Rn, Rm + 2 → Rm	0110nnnnmmmm0101	1	—
MOV.L	@Rm+,Rn	(Rm) → Rn,Rm + 4 → Rm	0110nnnnmmmm0110	1	—
MOV.B	R0,@(disp,Rn)	R0 → (disp + Rn)	10000000nnnndddd	1	—
MOV.W	R0,@(disp,Rn)	R0 → (disp × 2 + Rn)	10000001nnnndddd	1	—
MOV.L	Rm,@(disp,Rn)	Rm → (disp × 4 + Rn)	0001nnnnmmmmdddd	1	—
MOV.B	@(disp,Rm),R0	(disp + Rm) → Sign extension → R0	10000100mmmmdddd	1	—
MOV.W	@(disp,Rm),R0	(disp × 2 + Rm) → Sign extension → R0	10000101mmmmdddd	1	—
MOV.L	@(disp,Rm),Rn	(disp × 4 + Rm) → Rn	0101nnnnmmmmdddd	1	—

Instruction		Operation	Code	Execution Cycles	T Bit
MOV.B	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0100	1	—
MOV.W	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0101	1	—
MOV.L	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0110	1	—
MOV.B	@(R0,Rm),Rn	(R0 + Rm) → Sign extension → Rn	0000nnnnmmmm1100	1	—
MOV.W	@(R0,Rm),Rn	(R0 + Rm) → Sign extension → Rn	0000nnnnmmmm1101	1	—
MOV.L	@(R0,Rm),Rn	(R0 + Rm) → Rn	0000nnnnmmmm1110	1	—
MOV.B	R0,@(disp,GBR)	R0 → (disp + GBR)	11000000dddddddd	1	—
MOV.W	R0,@(disp,GBR)	R0 → (disp × 2 + GBR)	11000001dddddddd	1	—
MOV.L	R0,@(disp,GBR)	R0 → (disp × 4 + GBR)	11000010dddddddd	1	—
MOV.B	@(disp,GBR),R0	(disp + GBR) → Sign extension → R0	11000100dddddddd	1	—
MOV.W	@(disp,GBR),R0	(disp × 2 + GBR) → Sign extension → R0	11000101dddddddd	1	—
MOV.L	@(disp,GBR),R0	(disp × 4 + GBR) → R0	11000110dddddddd	1	—
MOVA	@(disp,PC),R0	disp × 4 + PC → R0	11000111dddddddd	1	—
MOVT	Rn	T → Rn	0000nnnn00101001	1	—
SWAP.B	Rm,Rn	Rm → Swap lowest two bytes → Rn	0110nnnnmmmm1000	1	—
SWAP.W	Rm,Rn	Rm → Swap two consecutive words → Rn	0110nnnnmmmm1001	1	—
XTRCT	Rm,Rn	Rm: Middle 32 bits of Rn → Rn	0010nnnnmmmm1101	1	—

2.5.3 Arithmetic Operation Instructions

Table 2.12 Arithmetic Operation Instructions

Instruction		Operation	Code	Execution Cycles	T Bit
ADD	Rm,Rn	Rn + Rm → Rn	0011nnnnmmmm1100	1	—
ADD	#imm,Rn	Rn + imm → Rn	0111nnnniiiiiiii	1	—
ADDC	Rm,Rn	Rn + Rm + T → Rn, Carry → T	0011nnnnmmmm1110	1	Carry
ADDV	Rm,Rn	Rn + Rm → Rn, Overflow → T	0011nnnnmmmm1111	1	Overflow
CMP/EQ	#imm,R0	If R0 = imm, 1 → T	10001000iiiiiiii	1	Comparison result
CMP/EQ	Rm,Rn	If Rn = Rm, 1 → T	0011nnnnmmmm0000	1	Comparison result
CMP/HS	Rm,Rn	If Rn ≥ Rm with unsigned data, 1 → T	0011nnnnmmmm0010	1	Comparison result
CMP/GE	Rm,Rn	If Rn ≥ Rm with signed data, 1 → T	0011nnnnmmmm0011	1	Comparison result
CMP/HI	Rm,Rn	If Rn > Rm with unsigned data, 1 → T	0011nnnnmmmm0110	1	Comparison result
CMP/GT	Rm,Rn	If Rn > Rm with signed data, 1 → T	0011nnnnmmmm0111	1	Comparison result
CMP/PZ	Rn	If Rn ≥ 0, 1 → T	0100nnnn00010001	1	Comparison result
CMP/PL	Rn	If Rn > 0, 1 → T	0100nnnn00010101	1	Comparison result
CMP/STR	Rm,Rn	If Rn and Rm have an equivalent byte, 1 → T	0010nnnnmmmm1100	1	Comparison result
DIV1	Rm,Rn	Single-step division (Rn/Rm)	0011nnnnmmmm0100	1	Calculation result
DIV0S	Rm,Rn	MSB of Rn → Q, MSB of Rm → M, M^ Q → T	0010nnnnmmmm0111	1	Calculation result
DIV0U		0 → M/Q/T	0000000000011001	1	0
DMULS.L	Rm,Rn	Signed operation of Rn × Rm → MACH, MACL 32 × 32 → 64 bits	0011nnnnmmmm1101	2 to 5*	—

Instruction	Operation	Code	Execution Cycles	T Bit
DMULU.L Rm,Rn	Unsigned operation of Rn × Rm → MACH, MACL 32 × 32 → 64 bits	0011nnnnmmmm0101	2 to 5*	—
DT Rn	Rn - 1 → Rn, if Rn = 0, 1 → T, else 0 → T	0100nnnn00010000	1	Comparison result
EXTS.B Rm,Rn	A byte in Rm is sign-extended → Rn	0110nnnnmmmm1110	1	—
EXTS.W Rm,Rn	A word in Rm is sign-extended → Rn	0110nnnnmmmm1111	1	—
EXTU.B Rm,Rn	A byte in Rm is zero-extended → Rn	0110nnnnmmmm1100	1	—
EXTU.W Rm,Rn	A word in Rm is zero-extended → Rn	0110nnnnmmmm1101	1	—
MAC.L @Rm+,@Rn+	Signed operation of (Rn) × (Rm) + MAC → MAC, 32 × 32 + 64 → 64 bits	0000nnnnmmmm1111	2 to 5*	—
MAC.W @Rm+,@Rn+	Signed operation of (Rn) × (Rm) + MAC → MAC, 16 × 16 + 64 → 64 bits	0100nnnnmmmm1111	2 to 4*	—
MUL.L Rm,Rn	Rn × Rm → MACL 32 × 32 → 32 bits	0000nnnnmmmm0111	2 to 5*	—
MULS.W Rm,Rn	Signed operation of Rn × Rm → MAC 16 × 16 → 32 bits	0010nnnnmmmm1111	1 to 3*	—
MULU.W Rm,Rn	Unsigned operation of Rn × Rm → MAC 16 × 16 → 32 bits	0010nnnnmmmm1110	1 to 3*	—
NEG Rm,Rn	0-Rm → Rn	0110nnnnmmmm1011	1	—
NEGC Rm,Rn	0-Rm-T → Rn, Borrow → T	0110nnnnmmmm1010	1	Borrow
SUB Rm,Rn	Rn-Rm → Rn	0011nnnnmmmm1000	1	—
SUBC Rm,Rn	Rn-Rm–T → Rn, Borrow → T	0011nnnnmmmm1010	1	Borrow
SUBV Rm,Rn	Rn-Rm → Rn, Underflow → T	0011nnnnmmmm1011	1	Overflow

Note: * Indicates the number of execution cycles for normal operation.

2.5.4 Logic Operation Instructions

Table 2.13 Logic Operation Instructions

Instruction		Operation	Code	Execution Cycles	T Bit
AND	Rm,Rn	Rn & Rm → Rn	0010nnnnmmmm1001	1	—
AND	#imm,R0	R0 & imm → R0	11001001iiiiiiii	1	—
AND.B	#imm,@(R0,GBR)	(R0 + GBR) & imm → (R0 + GBR)	11001101iiiiiiii	3	—
NOT	Rm,Rn	~Rm → Rn	0110nnnnmmmm0111	1	—
OR	Rm,Rn	Rn | Rm → Rn	0010nnnnmmmm1011	1	—
OR	#imm,R0	R0 | imm → R0	11001011iiiiiiii	1	—
OR.B	#imm,@(R0,GBR)	(R0 + GBR) | imm → (R0 + GBR)	11001111iiiiiiii	3	—
TAS.B	@Rn	If (Rn) is 0, 1 → T; 1 → MSB of (Rn)	0100nnnn00011011	4	Test result
TST	Rm,Rn	Rn & Rm; if the result is 0, 1 → T	0010nnnnmmmm1000	1	Test result
TST	#imm,R0	R0 & imm; if the result is 0, 1 → T	11001000iiiiiiii	1	Test result
TST.B	#imm,@(R0,GBR)	(R0 + GBR) & imm; if the result is 0, 1 → T	11001100iiiiiiii	3	Test result
XOR	Rm,Rn	Rn ^ Rm → Rn	0010nnnnmmmm1010	1	—
XOR	#imm,R0	R0 ^ imm → R0	11001010iiiiiiii	1	—
XOR.B	#imm,@(R0,GBR)	(R0 + GBR) ^ imm → (R0 + GBR)	11001110iiiiiiii	3	—

2.5.5 Shift Instructions

Table 2.14 Shift Instructions

Instruction		Operation	Code	Execution Cycles	T Bit
ROTL	Rn	T ← Rn ← MSB	0100nnnn00000100	1	MSB
ROTR	Rn	LSB → Rn → T	0100nnnn00000101	1	LSB
ROTCL	Rn	T ← Rn ← T	0100nnnn00100100	1	MSB
ROTCR	Rn	T → Rn → T	0100nnnn00100101	1	LSB
SHAL	Rn	T ← Rn ← 0	0100nnnn00100000	1	MSB
SHAR	Rn	MSB → Rn → T	0100nnnn00100001	1	LSB
SHLL	Rn	T ← Rn ← 0	0100nnnn00000000	1	MSB
SHLR	Rn	0 → Rn → T	0100nnnn00000001	1	LSB
SHLL2	Rn	Rn << 2 → Rn	0100nnnn00001000	1	—
SHLR2	Rn	Rn >> 2 → Rn	0100nnnn00001001	1	—
SHLL8	Rn	Rn << 8 → Rn	0100nnnn00011000	1	—
SHLR8	Rn	Rn >> 8 → Rn	0100nnnn00011001	1	—
SHLL16	Rn	Rn << 16 → Rn	0100nnnn00101000	1	—
SHLR16	Rn	Rn >> 16 → Rn	0100nnnn00101001	1	—

2.5.6 Branch Instructions

Table 2.15 Branch Instructions

Instruction		Operation	Code	Execution Cycles	T Bit
BF	label	If T = 0, disp × 2 + PC → PC; if T = 1, nop	10001011dddddddd	3/1*	—
BF/S	label	Delayed branch, if T = 0, disp × 2 + PC → PC; if T = 1, nop	10001111dddddddd	2/1*	—
BT	label	If T = 1, disp × 2 + PC → PC; if T = 0, nop	10001001dddddddd	3/1*	—
BT/S	label	Delayed branch, if T = 1, disp × 2 + PC → PC; if T = 0, nop	10001101dddddddd	2/1*	—
BRA	label	Delayed branch, disp × 2 + PC → PC	1010dddddddddddd	2	—
BRAF	Rm	Delayed branch, Rm + PC → PC	0000mmmm00100011	2	—
BSR	label	Delayed branch, PC → PR, disp × 2 + PC → PC	1011dddddddddddd	2	—
BSRF	Rm	Delayed branch, PC → PR, Rm + PC → PC	0000mmmm00000011	2	—
JMP	@Rm	Delayed branch, Rm → PC	0100mmmm00101011	2	—
JSR	@Rm	Delayed branch, PC → PR, Rm → PC	0100mmmm00001011	2	—
RTS		Delayed branch, PR → PC	0000000000001011	2	—

Note: * One cycle when the branch is not executed.

2.5.7 System Control Instructions

Table 2.16 System Control Instructions

Instruction	Operation	Code	Execution Cycles	T Bit
CLRT	0 → T	0000000000001000	1	0
CLRMAC	0 → MACH, MACL	0000000000101000	1	—
LDC Rm,SR	Rm → SR	0100mmmm00001110	6	LSB
LDC Rm,GBR	Rm → GBR	0100mmmm00011110	4	—
LDC Rm,VBR	Rm → VBR	0100mmmm00101110	4	—
LDC.L @Rm+,SR	(Rm) → SR, Rm + 4 → Rm	0100mmmm00000111	8	LSB
LDC.L @Rm+,GBR	(Rm) → GBR, Rm + 4 → Rm	0100mmmm00010111	4	—
LDC.L @Rm+,VBR	(Rm) → VBR, Rm + 4 → Rm	0100mmmm00100111	4	—
LDS Rm,MACH	Rm → MACH	0100mmmm00001010	1	—
LDS Rm,MACL	Rm → MACL	0100mmmm00011010	1	—
LDS Rm,PR	Rm → PR	0100mmmm00101010	1	—
LDS.L @Rm+,MACH	(Rm) → MACH, Rm + 4 → Rm	0100mmmm00000110	1	—
LDS.L @Rm+,MACL	(Rm) → MACL, Rm + 4 → Rm	0100mmmm00010110	1	—
LDS.L @Rm+,PR	(Rm) → PR, Rm + 4 → Rm	0100mmmm00100110	1	—
NOP	No operation	0000000000001001	1	—
RTE	Delayed branch, Stack area → PC/SR	0000000000101011	5	—
SETT	1 → T	0000000000011000	1	1
SLEEP	Sleep	0000000000011011	4*	—
STC SR,Rn	SR → Rn	0000nnnn00000010	1	—
STC GBR,Rn	GBR → Rn	0000nnnn00010010	1	—
STC VBR,Rn	VBR → Rn	0000nnnn00100010	1	—
STC.L SR,@–Rn	Rn–4 → Rn, SR → (Rn)	0100nnnn00000011	1	—
STC.L GBR,@–Rn	Rn–4 → Rn, GBR → (Rn)	0100nnnn00010011	1	—
STC.L VBR,@–Rn	Rn–4 → Rn, VBR → (Rn)	0100nnnn00100011	1	—

Instruction	Operation	Code	Execution Cycles	T Bit
STS MACH,Rn	MACH → Rn	0000nnnn00001010	1	—
STS MACL,Rn	MACL → Rn	0000nnnn00011010	1	—
STS PR,Rn	PR → Rn	0000nnnn00101010	1	—
STS.L MACH,@–Rn	Rn–4 → Rn, MACH → (Rn)	0100nnnn00000010	1	—
STS.L MACL,@–Rn	Rn–4 → Rn, MACL → (Rn)	0100nnnn00010010	1	—
STS.L PR,@–Rn	Rn–4 → Rn, PR → (Rn)	0100nnnn00100010	1	—
TRAPA #imm	PC/SR → Stack area, (imm × 4 + VBR) → PC	11000011iiiiiiii	8	—

Note: * Number of execution cycles until this LSI enters sleep mode.

About the number of execution cycles:

The table lists the minimum number of execution cycles. In practice, the number of execution cycles will be increased depending on the conditions such as:

- When there is a conflict between instruction fetch and data access
- When the destination register of a load instruction (memory → register) is also used by the instruction immediately after the load instruction.

2.6 Processing States

The CPU has the five processing states: reset, exception handling, bus release, program execution, and power-down. Figure 2.4 shows the CPU state transition.

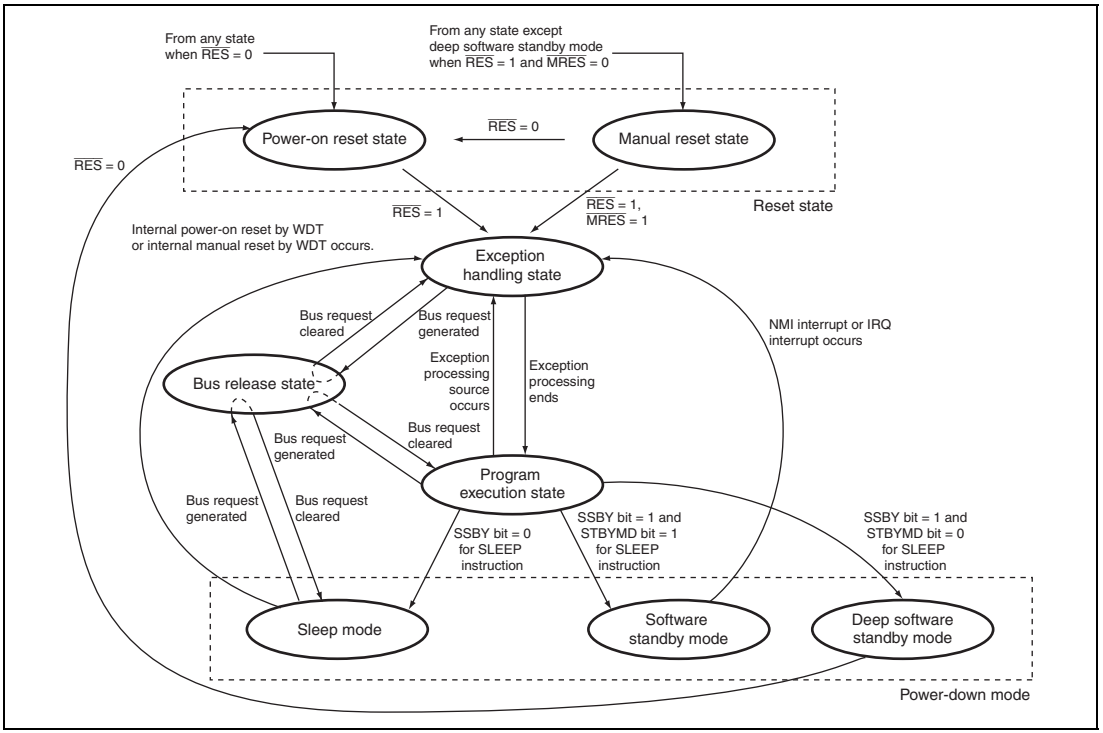


Figure 2.4 Transitions between Processing States

- Reset state

  The CPU is reset. When the $\overline{\text{RES}}$ pin is low, the CPU enters the power-on reset state. When the $\overline{\text{RES}}$ pin is high and $\overline{\text{MRES}}$ pin is low, the CPU enters the manual reset state.

- Exception handling state

  This state is a transitional state in which the CPU processing state changes due to a request for exception handling such as a reset or an interrupt.

  When a reset occurs, the execution start address as the initial value of the program counter (PC) and the initial value of the stack pointer (SP) are fetched from the exception handling vector table. Then, a branch is made for the start address to execute a program.

  When an interrupt occurs, the PC and status register (SR) are saved in the stack area pointed to by SP. The start address of an exception handling routine is fetched from the exception handling vector table and a branch to the address is made to execute a program.

  Then the processing state enters the program execution state.

- Program execution state

  The CPU executes programs sequentially.

- Power-down state

  The CPU stops to reduce power consumption. The SLEEP instruction makes the CPU enter sleep mode, software standby mode, or deep software standby mode.

- Bus release state

  In the bus release state, the CPU releases access rights to the bus to the device that has requested them.

Section 3 MCU Operating Modes

3.1 Selection of Operating Modes

This LSI has four MCU operating modes and three on-chip flash memory programming modes. The operating mode is determined by the setting of FWE, MD1, and MD0 pins. Table 3.1 shows the allowable combinations of these pin settings; do not set these pins in the other way than the shown combinations.

When power is applied to the system, be sure to conduct power-on reset.

The MCU operating mode can be selected from MCU extension modes 0 to 2 and single chip mode. For the on-chip flash memory programming mode, boot mode, user boot mode, and user program mode which are on-chip programming modes are available.

Table 3.1 Selection of Operating Modes*[1]

	Pin Setting					Bus Width of CS0 Space			
Mode No.	**FWE**	**MD1**	**MD0**	**Mode Name**	**On-Chip ROM**	**SH7083**	**SH7084**	**SH7085**	**SH7086**
Mode 0	0	0	0	MCU extension mode 0	Not active	8	8	16	16
Mode 1	0	0	1	MCU extension mode 1	Not active	16	16	32	32
Mode 2	0	1	0	MCU extension mode 2	Active	Set by CS0BCR in BSC			
Mode 3	0	1	1	Single chip mode	Active	—			
Mode 4*[2]	1	0	0	Boot mode	Active	—			
Mode 5*[2]	1	0	1	User boot mode	Active	Set by CS0BCR in BSC			
Mode 6*[2]	1	1	0	User programming mode	Active	Set by CS0BCR in BSC			
Mode 7*[2]	1	1	1			—			

Notes: 1. Do not input a low-level signal to $\overline{\text{ASEMD0}}$ when the MCU is not connected to the E10A. Operation cannot be guaranteed if a low-level signal is input to $\overline{\text{ASEMD0}}$ when the E10A is not connected. For information on connecting the E10A, see SuperH™ Family E10A-USB Emulator, Additional Document for User's Manual: Supplementary Information on Using the SH7083, SH7084, SH7085, and SH7086.

2. These are flash memory programming modes.

3.2 Input/Output Pins

Table 3.2 describes the configuration of operating mode related pin.

Table 3.2 Pin Configuration

Pin Name	Input/Output	Function
MD0	Input	Designates operating mode through the level applied to this pin
MD1	Input	Designates operating mode through the level applied to this pin
FWE	Input	Enables, by hardware, programming/erasing of the on-chip flash memory

3.3 Operating Modes

3.3.1 Mode 0 (MCU Extension Mode 0)

CS0 space becomes external memory spaces with 8-bit bus width in SH7083/SH7084 or 16-bit bus width in SH7085/SH7086.

3.3.2 Mode 1 (MCU Extension Mode 1)

CS0 space becomes external memory spaces with 16-bit bus width in SH7083/SH7084 or 32-bit bus width in SH7085/SH7086.

3.3.3 Mode 2 (MCU Extension Mode 2)

The on-chip ROM is active and CS0 space can be used in this mode.

3.3.4 Mode 3 (Single Chip Mode)

All ports can be used in this mode, however the external address cannot be used.

3.4 Address Map

The address map for the operating modes is shown in figures 3.1 to 3.7.

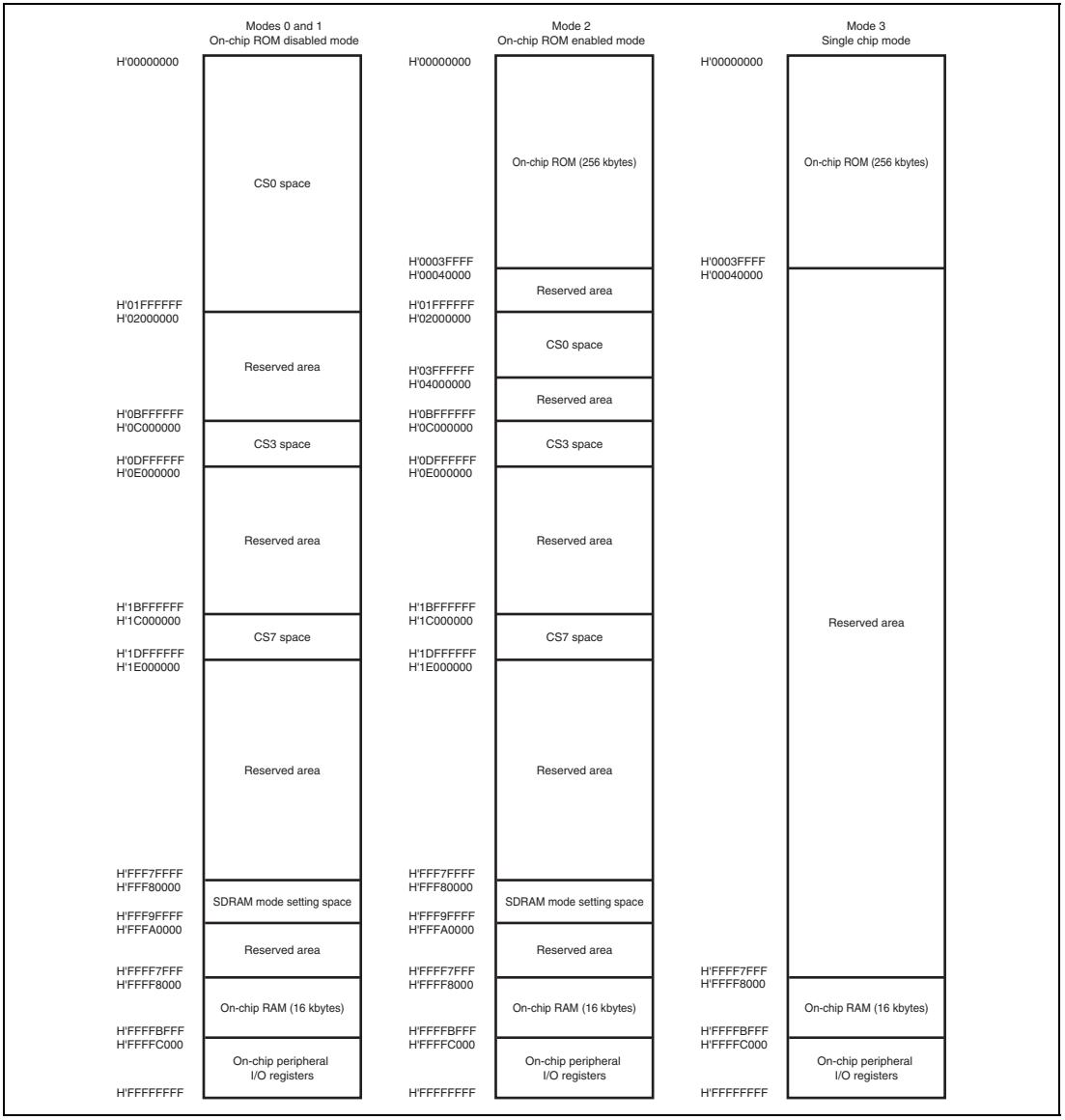


Figure 3.1 Address Map for Each Operating Mode in SH7083 (256-Kbyte Flash Memory Version)

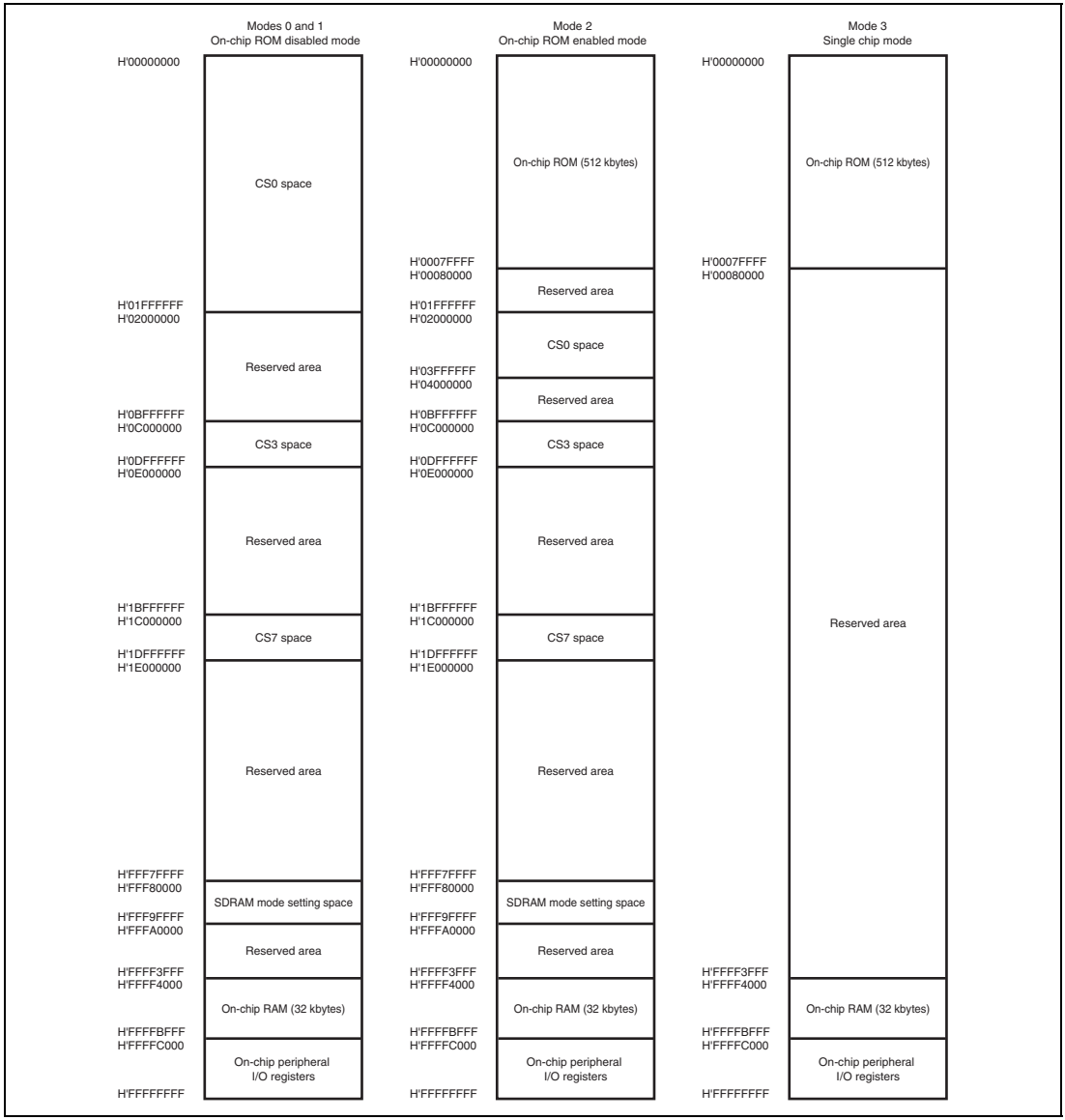


Figure 3.2 Address Map for Each Operating Mode in SH7083 (512-Kbyte Flash Memory Version)

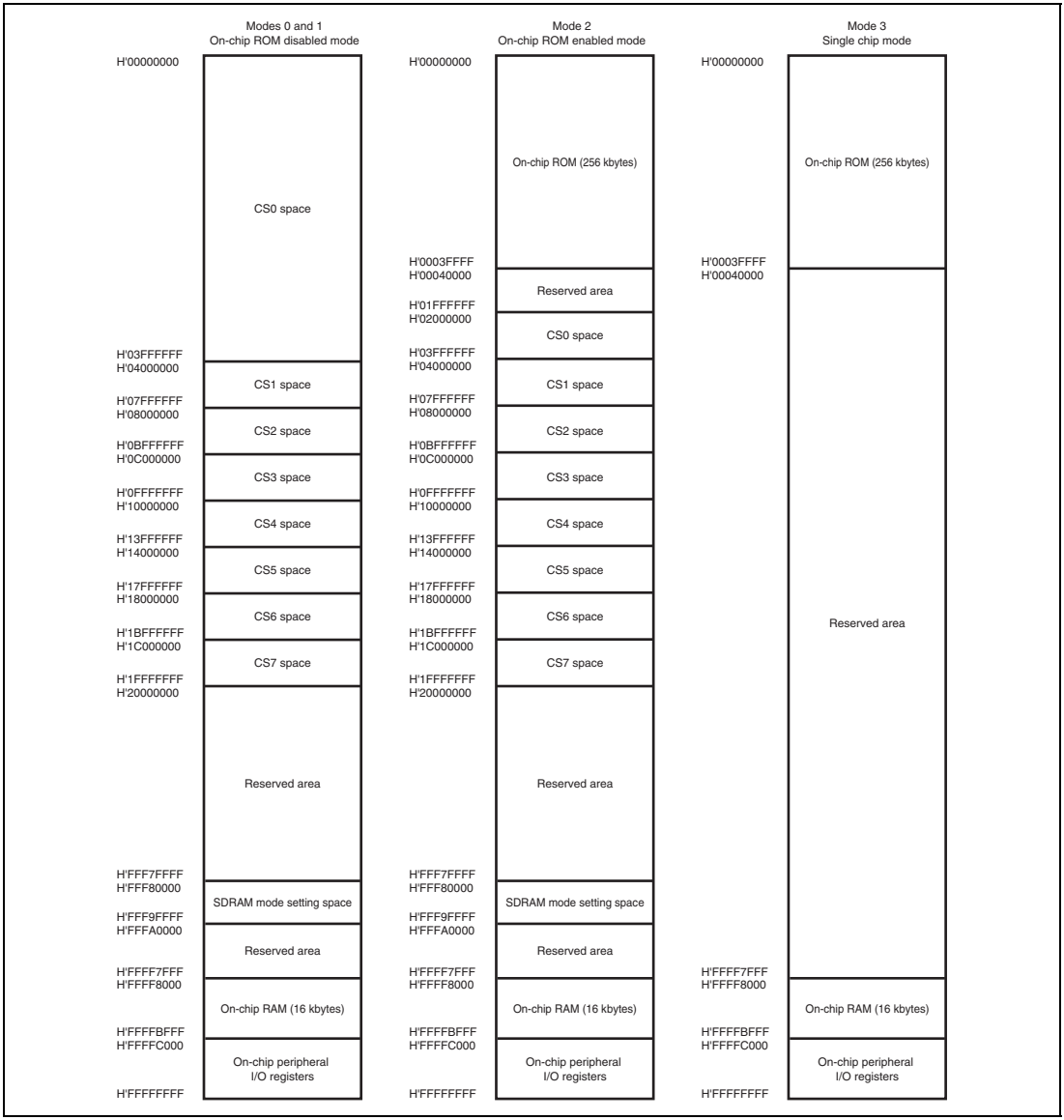


Figure 3.3 Address Map for Each Operating Mode in SH7084 (256-Kbyte Flash Memory Version)

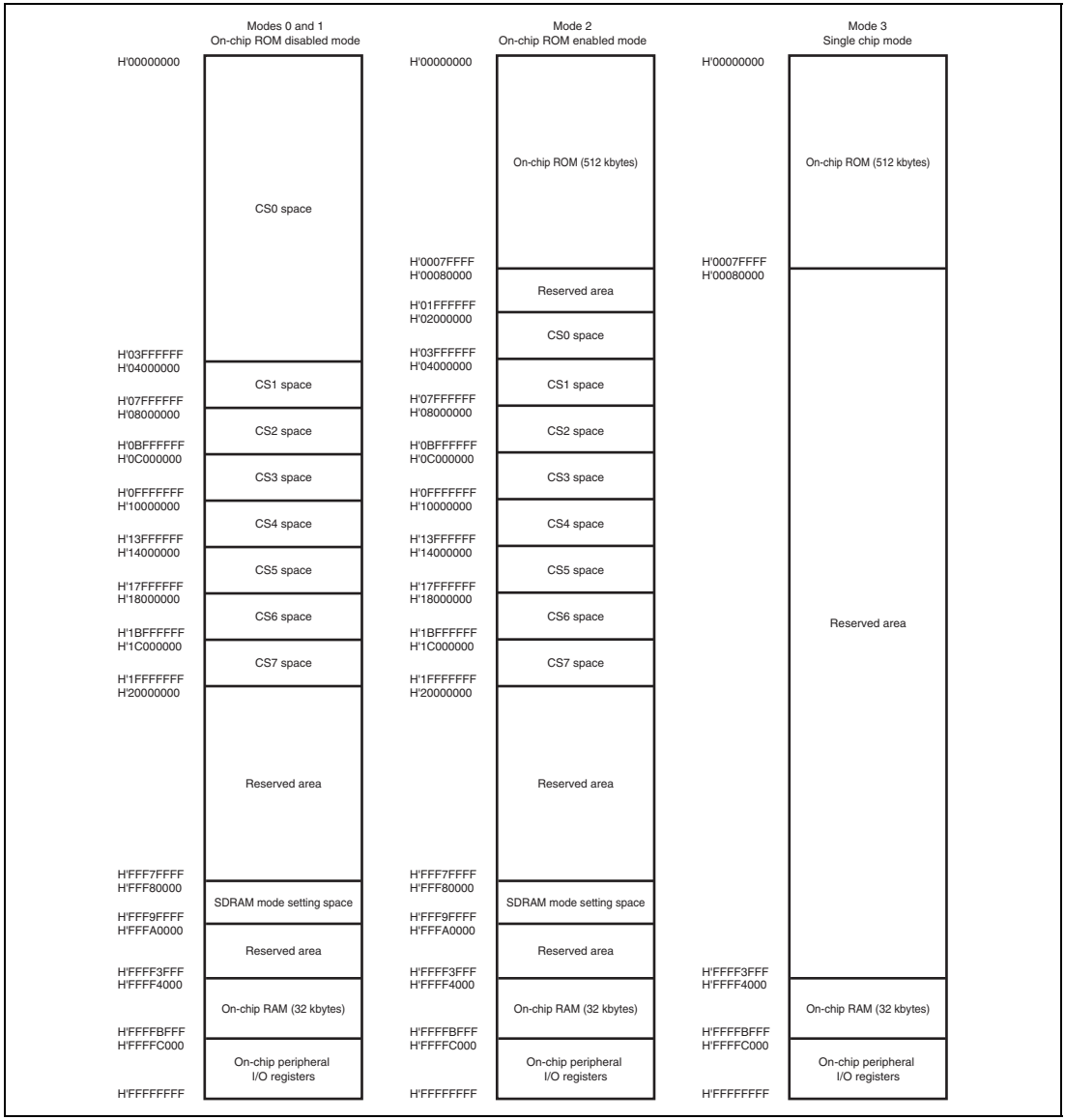


Figure 3.4 Address Map for Each Operating Mode in SH7084 (512-Kbyte Flash Memory Version)

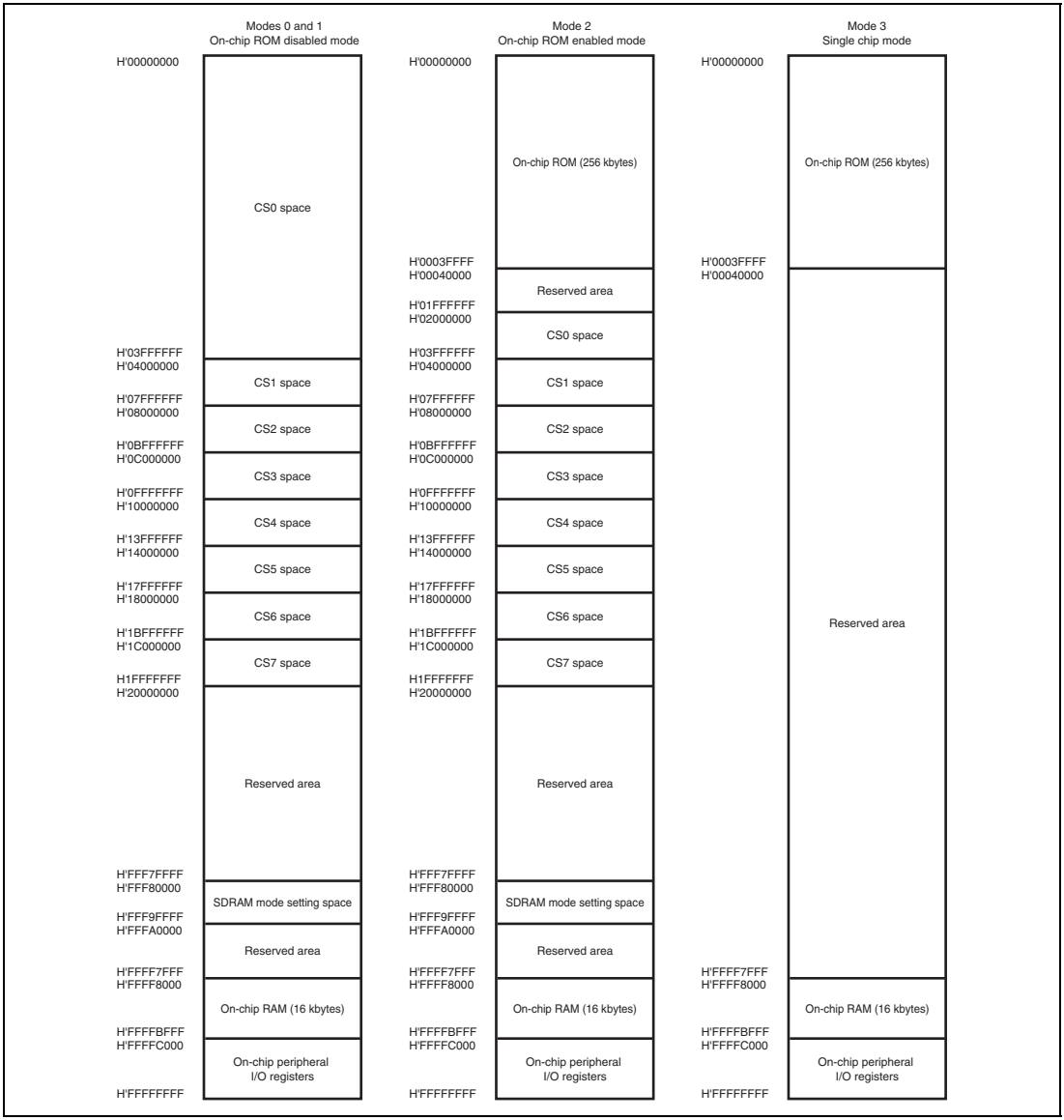


Figure 3.5 Address Map for Each Operating Mode in SH7085 (256-Kbyte Flash Memory Version)

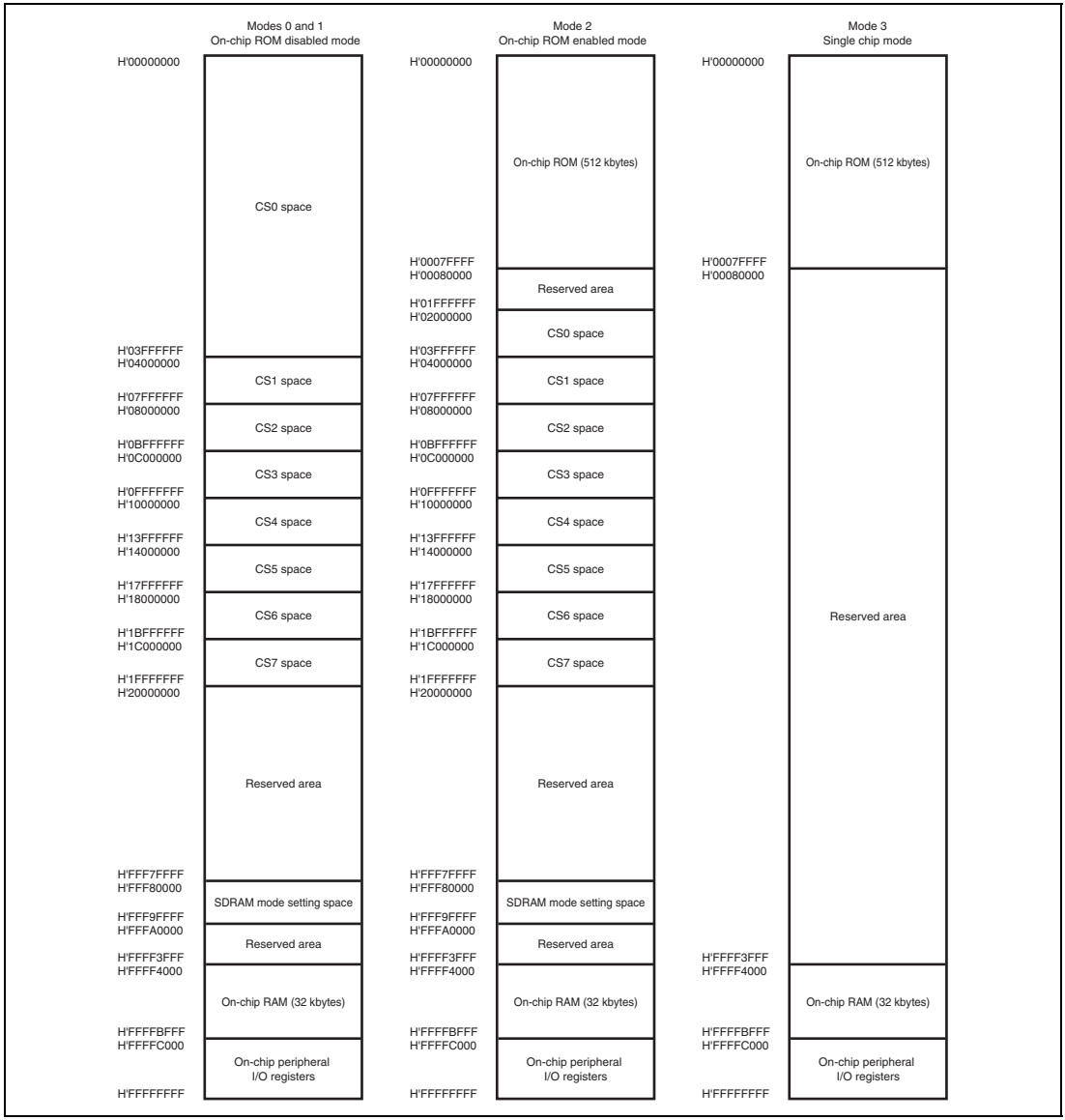


Figure 3.6 Address Map for Each Operating Mode in SH7085 (512-Kbyte Flash Memory Version)

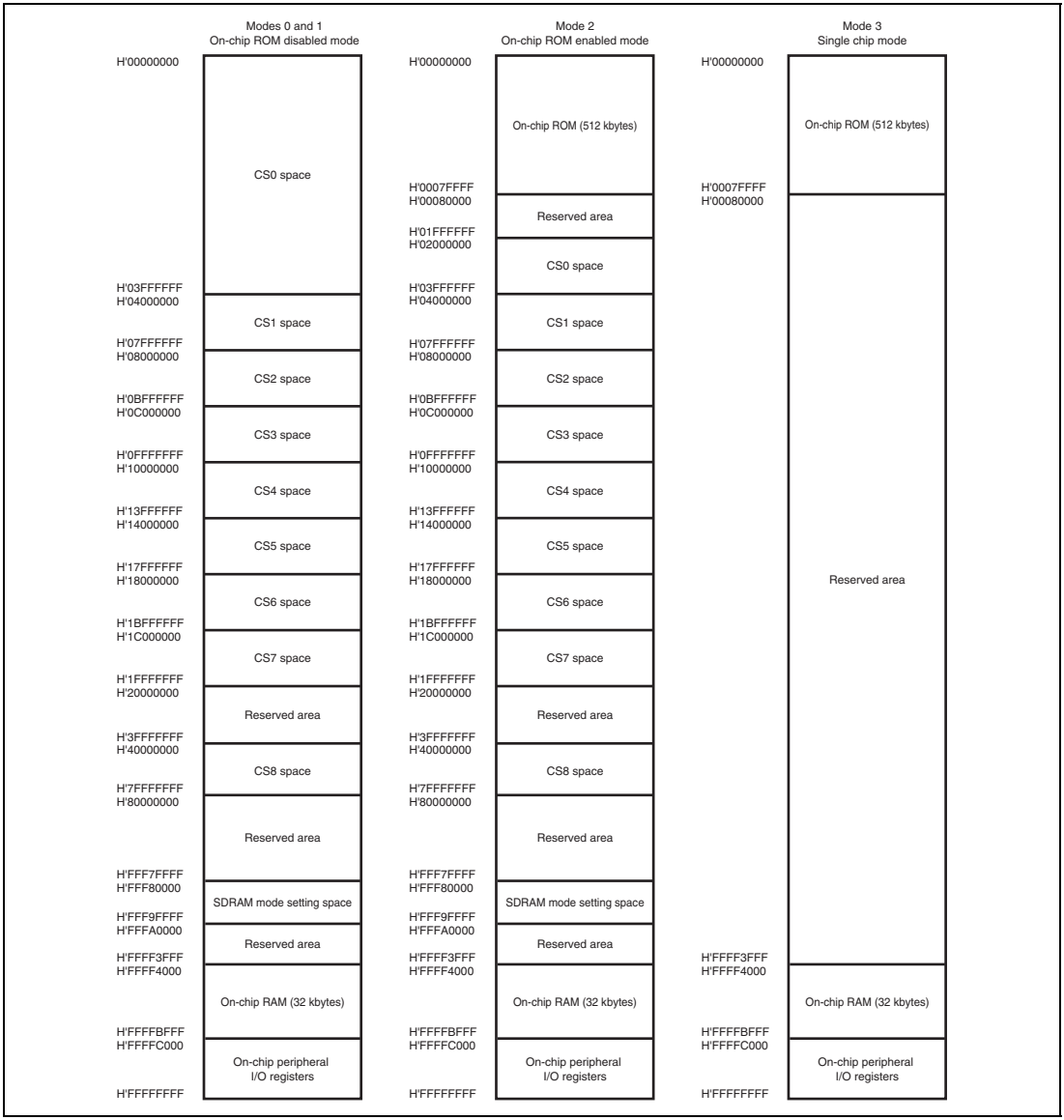


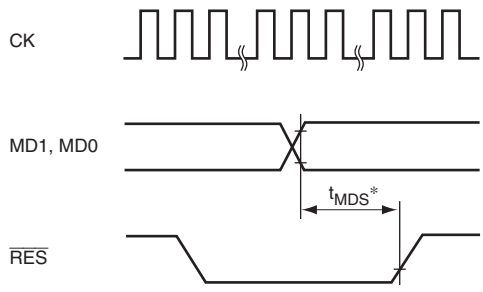
Figure 3.7 Address Map for Each Operating Mode in SH7086

3.5 Initial State in This LSI

In the initial state of this LSI, some of on-chip modules are set in module standby state for saving power. When operating these modules, clear module standby state according to the procedure in section 26, Power-Down Modes.

3.6 Note on Changing Operating Mode

When changing operating mode while power is applied to this LSI, make sure to do it in the power-on reset state (that is, the low level is applied to the $\overline{\text{RES}}$ pin).



Note: * See section 28.3.2, Control Signal Timing.

Figure 3.8 Reset Input Timing when Changing Operating Mode

Section 4 Clock Pulse Generator (CPG)

This LSI has a clock pulse generator (CPG) that generates an internal clock (Iϕ), a bus clock (Bϕ), a peripheral clock (Pϕ), and clocks (MIϕ and MPϕ) for the MTU2S and MTU2 modules. The CPG also controls power-down modes.

4.1 Features

- Five clocks generated independently

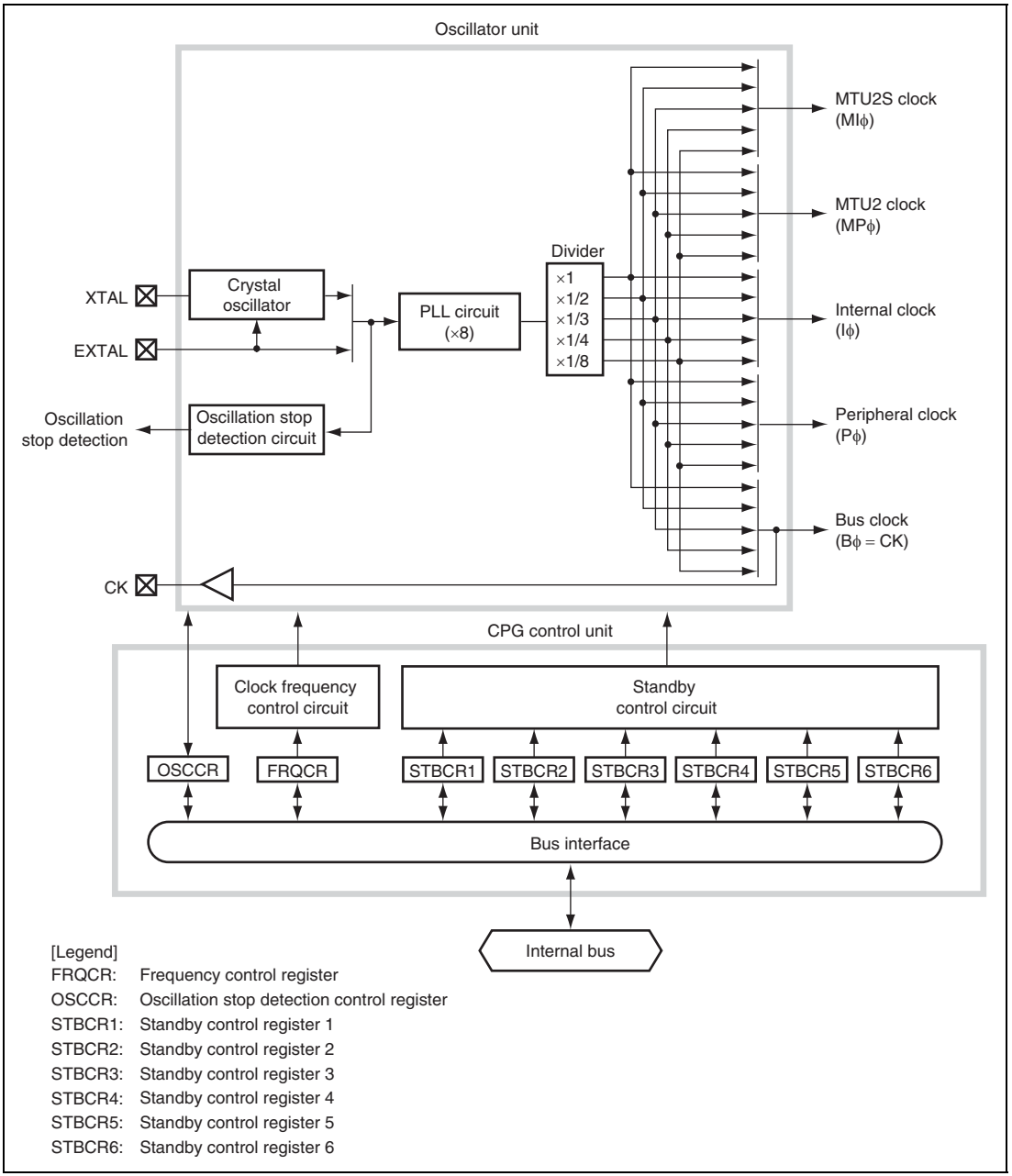
  An internal clock (Iϕ) for the CPU; a peripheral clock (Pϕ) for the on-chip peripheral modules; a bus clock (Bϕ = CK) for the external bus interface; a MTU2S clock (MIϕ) for the on-chip MTU2S module; and a MTU2 clock (MPϕ) for the on-chip MTU2 module.
- Frequency change function

  Frequencies of the internal clock (Iϕ), bus clock (Bϕ), peripheral clock (Pϕ), MTU2S clock (MIϕ), and MTU2 clock (MPϕ) can be changed independently using the divider circuit within the CPG. Frequencies are changed by software using the frequency control register (FRQCR) setting.
- Power-down mode control

  The clock can be stopped in sleep mode and standby mode and specific modules can be stopped using the module standby function.
- Oscillation stop detection

  If the clock supplied through the clock input pin stops for any reason, the timer pins can be automatically placed in the high-impedance state.

Figure 4.1 shows a block diagram of the clock pulse generator.



[Legend]
FRQCR: Frequency control register
OSCCR: Oscillation stop detection control register
STBCR1: Standby control register 1
STBCR2: Standby control register 2
STBCR3: Standby control register 3
STBCR4: Standby control register 4
STBCR5: Standby control register 5
STBCR6: Standby control register 6

Figure 4.1 Block Diagram of Clock Pulse Generator

The clock pulse generator blocks function as follows:

PLL Circuit: The PLL circuit multiples the clock frequency input from the crystal oscillator or the EXTAL pin by 8. The multiplication ratio is fixed at ×8.

Crystal Oscillator: The crystal oscillator is an oscillator circuit when a crystal resonator is connected to the XTAL and EXTAL pins.

Divider: The divider generates clocks with the frequencies to be used by the internal clock (Iϕ), bus clock (Bϕ), peripheral clock (Pϕ), MTU2S clock (MIϕ), and MTU2 clock (MPϕ).

The frequencies can be selected from 1, 1/2, 1/3, 1/4, and 1/8 times the frequency output from the PLL circuit. The division ratio should be specified in the frequency control register (FRQCR).

Oscillation Stop Detection Circuit: This circuit detects an abnormal condition in the crystal oscillator.

Clock Frequency Control Circuit: The clock frequency control circuit controls the clock frequency according to the setting in the frequency control register (FRQCR).

Standby Control Circuit: The standby control circuit controls the state of the on-chip oscillator circuit and other modules in sleep or standby mode.

Frequency Control Register (FRQCR): The frequency control register (FRQCR) has control bits for the frequency division ratios of the internal clock (Iϕ), bus clock (Bϕ), peripheral clock (Pϕ), MTU2S clock (MIϕ), and MTU2 clock (MPϕ).

Oscillation Stop Detection Control Register (OSCCR): The oscillation stop detection control register (OSCCR) has an oscillation stop detection flag and a bit for selecting flag status output through an external pin.

Standby Control Registers 1 to 6 (STBCR1 to STBCR6): The standby control register (STBCR) has bits for controlling the power-down modes. For details, see section 26, Power-Down Modes.

Table 4.1 shows the operating clock for each module.

Table 4.1 Operating Clock for Each Module

Operating Clock	Operating Module	Operating Clock	Operating Module
Internal clock (Iϕ)	CPU	Peripheral clock (Pϕ)	POE
	UBC		SCI
	ROM		SCIF
	RAM		SSU
			IIC2
			A/D
			CMT
			WDT
Bus clock (Bϕ)	BSC	MTU2 clock (MPϕ)	MTU2
	DMAC	MTU2S clock (MIϕ)	MTU2S
	DTC		

4.2 Input/Output Pins

Table 4.2 shows the CPG pin configuration.

Table 4.2 Pin Configuration

Pin Name	Abbr.	I/O	Description
Crystal input/output pins (clock input pins)	XTAL	Output	Connects a crystal resonator.
	EXTAL	Input	Connects a crystal resonator or an external clock.
Clock output pin	CK	Output	Outputs an external clock.

Note: To use the clock output (CK) pin, appropriate settings may be needed for the pin in the pin function controller (PFC) in some cases. For details, refer to section 21, Pin Function Controller (PFC).

4.3 Clock Operating Mode

Table 4.3 shows the clock operating mode of this LSI.

Table 4.3 Clock Operating Mode

Clock Operating Mode	Clock I/O		PLL Circuit	Input to Divider
	Source	Output		
1	EXTAL input or crystal resonator	CK*	ON (×8)	×8

Note: * To output the clock through the clock output (CK) pin, appropriate settings should be made in the pin function controller (PFC). For details, refer to section 21, Pin Function Controller (PFC).

Mode 1: The frequency of the external clock input from the EXTAL pin is multiplied by 8 in the PLL circuit before being supplied to the on-chip modules in this LSI, which eliminates the need to generate a high-frequency clock outside the LSI. Since the input clock frequency ranging from 5 MHz to 12.5 MHz can be used, the internal clock (Iϕ) frequency ranges from 10 MHz to 80 MHz.

Maximum operating frequencies:
Iϕ = 80 MHz, Bϕ = 40 MHz, Pϕ = 40 MHz, MIϕ = 80 MHz, and MPϕ = 40 MHz

Table 4.4 shows the frequency division ratios that can be specified with FRQCR.

Table 4.4 Frequency Division Ratios Specifiable with FRQCR

PLL Multiplication Ratio	FRQCR Division Ratio Setting					Clock Ratio					Clock Frequency (MHz)*					
	Iϕ	Bϕ	Pϕ	MIϕ	MPϕ	Iϕ	Bϕ	Pϕ	MIϕ	MPϕ	Input Clock	Iϕ	Bϕ	Pϕ	MIϕ	MPϕ
×8	1/8	1/8	1/8	1/8	1/8	1	1	1	1	1	10	10	10	10	10	10
	1/4	1/8	1/8	1/8	1/8	2	1	1	1	1		20	10	10	10	10
	1/4	1/8	1/8	1/4	1/8	2	1	1	2	1		20	10	10	20	10
	1/4	1/4	1/8	1/8	1/8	2	2	1	1	1		20	20	10	10	10
	1/4	1/4	1/8	1/4	1/8	2	2	1	2	1		20	20	10	20	10
	1/4	1/4	1/8	1/4	1/4	2	2	1	2	2		20	20	10	20	20
	1/4	1/4	1/4	1/4	1/4	2	2	2	2	2		20	20	20	20	20
	1/3	1/3	1/3	1/3	1/3	8/3	8/3	8/3	8/3	8/3		26	26	26	26	26
	1/2	1/8	1/8	1/8	1/8	4	1	1	1	1		40	10	10	10	10
	1/2	1/8	1/8	1/4	1/8	4	1	1	2	1		40	10	10	20	10
	1/2	1/8	1/8	1/2	1/8	4	1	1	4	1		40	10	10	40	10
	1/2	1/4	1/8	1/8	1/8	4	2	1	1	1		40	20	10	10	10
	1/2	1/4	1/8	1/4	1/8	4	2	1	2	1		40	20	10	20	10
	1/2	1/4	1/8	1/4	1/4	4	2	1	2	2		40	20	10	20	20
	1/2	1/4	1/8	1/2	1/8	4	2	1	4	1		40	20	10	40	10
	1/2	1/4	1/8	1/2	1/4	4	2	1	4	2		40	20	10	40	20
	1/2	1/4	1/4	1/4	1/4	4	2	2	2	2		40	20	20	20	20
	1/2	1/4	1/4	1/2	1/4	4	2	2	4	2		40	20	20	40	20
	1/2	1/2	1/8	1/8	1/8	4	4	1	1	1		40	40	10	10	10
	1/2	1/2	1/8	1/4	1/8	4	4	1	2	1		40	40	10	20	10
	1/2	1/2	1/8	1/4	1/4	4	4	1	2	2		40	40	10	20	20
	1/2	1/2	1/8	1/2	1/8	4	4	1	4	1		40	40	10	40	10
	1/2	1/2	1/8	1/2	1/4	4	4	1	4	2		40	40	10	40	20
	1/2	1/2	1/8	1/2	1/2	4	4	1	4	4		40	40	10	40	40
	1/2	1/2	1/4	1/4	1/4	4	4	2	2	2		40	40	20	20	20
	1/2	1/2	1/4	1/2	1/4	4	4	2	4	2		40	40	20	40	20
	1/2	1/2	1/4	1/2	1/2	4	4	2	4	4		40	40	20	40	40

PLL Multiplication Ratio	FRQCR Division Ratio Setting					Clock Ratio					Clock Frequency (MHz)*					
	Iϕ	Bϕ	Pϕ	MIϕ	MPϕ	Iϕ	Bϕ	Pϕ	MIϕ	MPϕ	Input Clock	Iϕ	Bϕ	Pϕ	MIϕ	MPϕ
×8	1/2	1/2	1/2	1/2	1/2	4	4	4	4	4	10	40	40	40	40	40
	1/1	1/8	1/8	1/8	1/8	8	1	1	1	1		80	10	10	10	10
	1/1	1/8	1/8	1/4	1/8	8	1	1	2	1		80	10	10	20	10
	1/1	1/8	1/8	1/2	1/8	8	1	1	4	1		80	10	10	40	10
	1/1	1/8	1/8	1/1	1/8	8	1	1	8	1		80	10	10	80	10
	1/1	1/4	1/8	1/8	1/8	8	2	1	1	1		80	20	10	10	10
	1/1	1/4	1/8	1/4	1/8	8	2	1	2	1		80	20	10	20	10
	1/1	1/4	1/8	1/4	1/4	8	2	1	2	2		80	20	10	20	20
	1/1	1/4	1/8	1/2	1/8	8	2	1	4	1		80	20	10	40	10
	1/1	1/4	1/8	1/2	1/4	8	2	1	4	2		80	20	10	40	20
	1/1	1/4	1/8	1/1	1/8	8	2	1	8	1		80	20	10	80	10
	1/1	1/4	1/8	1/1	1/4	8	2	1	8	2		80	20	10	80	20
	1/1	1/4	1/4	1/4	1/4	8	2	2	2	2		80	20	20	20	20
	1/1	1/4	1/4	1/2	1/4	8	2	2	4	2		80	20	20	40	20
	1/1	1/4	1/4	1/1	1/4	8	2	2	8	2		80	20	20	80	20
	1/1	1/3	1/3	1/3	1/3	8	8/3	8/3	8/3	8/3		80	26	26	26	26
	1/1	1/3	1/3	1/1	1/3	8	8/3	8/3	8	8/3		80	26	26	80	26
	1/1	1/2	1/8	1/8	1/8	8	4	1	1	1		80	40	10	10	10
	1/1	1/2	1/8	1/4	1/8	8	4	1	2	1		80	40	10	20	10
	1/1	1/2	1/8	1/4	1/4	8	4	1	2	2		80	40	10	20	20
	1/1	1/2	1/8	1/2	1/8	8	4	1	4	1		80	40	10	40	10
	1/1	1/2	1/8	1/2	1/4	8	4	1	4	2		80	40	10	40	20
	1/1	1/2	1/8	1/2	1/2	8	4	1	4	4		80	40	10	40	40
	1/1	1/2	1/8	1/1	1/8	8	4	1	8	1		80	40	10	80	10
	1/1	1/2	1/8	1/1	1/4	8	4	1	8	2		80	40	10	80	20
	1/1	1/2	1/8	1/1	1/2	8	4	1	8	4		80	40	10	80	40
	1/1	1/2	1/4	1/4	1/4	8	4	2	2	2		80	40	20	20	20
	1/1	1/2	1/4	1/2	1/4	8	4	2	4	2		80	40	20	40	20

PLL Multiplication Ratio	FRQCR Division Ratio Setting					Clock Ratio					Clock Frequency (MHz)*					
	Iϕ	Bϕ	Pϕ	MIϕ	MPϕ	Iϕ	Bϕ	Pϕ	MIϕ	MPϕ	Input Clock	Iϕ	Bϕ	Pϕ	MIϕ	MPϕ
×8	1/1	1/2	1/4	1/2	1/2	8	4	2	4	4	10	80	40	20	40	40
	1/1	1/2	1/4	1/1	1/4	8	4	2	8	2		80	40	20	80	20
	1/1	1/2	1/4	1/1	1/2	8	4	2	8	4		80	40	20	80	40
	1/1	1/2	1/2	1/2	1/2	8	4	4	4	4		80	40	40	40	40
	1/1	1/2	1/2	1/1	1/2	8	4	4	8	4		80	40	40	80	40
	1/1	1/1	1/4	1/4	1/4	8	8	2	2	2	5	40	40	10	10	10
	1/1	1/1	1/4	1/2	1/4	8	8	2	4	2		40	40	10	20	10
	1/1	1/1	1/4	1/2	1/2	8	8	2	4	4		40	40	10	20	20
	1/1	1/1	1/4	1/1	1/4	8	8	2	8	2		40	40	10	40	10
	1/1	1/1	1/4	1/1	1/2	8	8	2	8	4		40	40	10	40	20
	1/1	1/1	1/4	1/1	1/1	8	8	2	8	8		40	40	10	40	40
	1/1	1/1	1/3	1/3	1/3	8	8	8/3	8/3	8/3		40	40	13	13	13
	1/1	1/1	1/3	1/1	1/3	8	8	8/3	8	8/3		40	40	13	40	13
	1/1	1/1	1/3	1/1	1/1	8	8	8/3	8	8		40	40	13	40	40
	1/1	1/1	1/2	1/2	1/2	8	8	4	4	4		40	40	20	20	20
	1/1	1/1	1/2	1/1	1/2	8	8	4	8	4		40	40	20	40	20
	1/1	1/1	1/2	1/1	1/1	8	8	4	8	8		40	40	20	40	40
	1/1	1/1	1/1	1/1	1/1	8	8	8	8	8		40	40	40	40	40

Notes: * Clock frequencies when the input clock frequency is assumed to be the shown value.

1. The PLL multiplication ratio is fixed at ×8. The division ratio can be selected from ×1, ×1/2, ×1/3, ×1/4, and ×1/8 for each clock by the setting in the frequency control register.
2. The output frequency of the PLL circuit is the product of the frequency of the input from the crystal resonator or EXTAL pin and the multiplication ratio (×8) of the PLL circuit.
3. The input to the divider is always the output from the PLL circuit.
4. The internal clock (Iϕ) frequency is the product of the frequency of the input from the crystal resonator or EXTAL pin, the multiplication ratio (×8) of the PLL circuit, and the division ratio of the divider. The resultant frequency must be a maximum of 80 MHz (maximum operating frequency).
5. The bus clock (Bϕ) frequency is the product of the frequency of the input from the crystal resonator or EXTAL pin, the multiplication ratio (×8) of the PLL circuit, and the division ratio of the divider. The resultant frequency must be a maximum of 40 MHz and equal to or lower than the internal clock (Iϕ) frequency.
6. The peripheral clock (Pϕ) frequency is the product of the frequency of the input from the crystal resonator or EXTAL pin, the multiplication ratio (×8) of the PLL circuit, and the division ratio of the divider. The resultant frequency must be a maximum of 40 MHz and equal to or lower than the bus clock (Bϕ) frequency.
7. When using the MTU2S and MTU2, the MTU2S clock (MIϕ) frequency must be equal to or lower than the internal clock (Iϕ) frequency and equal to or higher than the MTU2 clock (MPϕ) frequency. The MTU2 clock (MPϕ) frequency must be equal to or lower than the MTU2S clock (MIϕ) frequency and the bus clock (Bϕ) frequency and equal to or higher than the peripheral clock frequency (Pϕ). The MTU2S clock (MIϕ) frequency and MTU2 clock (MPϕ) frequency are the product of the frequency of the input from the crystal resonator or EXTAL pin, the multiplication ratio (×8) of the PLL circuit, and the division ratio of the divider.
8. The frequency of the CK pin is always be equal to the bus clock (Bϕ) frequency.

4.4 Register Descriptions

The CPG has the following registers.

For details on the addresses of these registers and the states of these registers in each processing state, see section 27, List of Registers.

Table 4.5 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Frequency control register	FRQCR	R/W	H'36DB	H'FFFFE800	16
Oscillation stop detection control register	OSCCR	R/W	H'00	H'FFFFE814	8

4.4.1 Frequency Control Register (FRQCR)

FRQCR is a 16-bit readable/writable register that specifies the frequency division ratios for the internal clock (Iϕ), bus clock (Bϕ), peripheral clock (Pϕ), MTU2S clock (MIϕ), and MTU2 clock (MPϕ). FRQCR can be accessed only in words.

FRQCR is initialized to H'36DB only by a power-on reset (except a power-on reset due to a WDT overflow).

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	IFC[2:0]			BFC[2:0]			PFC[2:0]			MIFC[2:0]			MPFC[2:0]		
Initial value:	0	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 12	IFC[2:0]	011	R/W	Internal Clock (Iϕ) Frequency Division Ratio Specify the division ratio of the internal clock (Iϕ) frequency with respect to the output frequency of PLL circuit. If a prohibited value is specified, subsequent operation is not guaranteed. 000: ×1 001: ×1/2 010: ×1/3 011: ×1/4 100: ×1/8 Other than above: Setting prohibited
11 to 9	BFC[2:0]	011	R/W	Bus Clock (Bϕ) Frequency Division Ratio Specify the division ratio of the bus clock (Bϕ) frequency with respect to the output frequency of PLL circuit. If a prohibited value is specified, subsequent operation is not guaranteed. 000: ×1 001: ×1/2 010: ×1/3 011: ×1/4 100: ×1/8 Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
8 to 6	PFC[2:0]	011	R/W	Peripheral Clock (Pϕ) Frequency Division Ratio Specify the division ratio of the peripheral clock (Pϕ) frequency with respect to the output frequency of PLL circuit. If a prohibited value is specified, subsequent operation is not guaranteed. 000: ×1 001: ×1/2 010: ×1/3 011: ×1/4 100: ×1/8 Other than above: Setting prohibited
5 to 3	MIFC[2:0]	011	R/W	MTU2S Clock (MIϕ) Frequency Division Ratio Specify the division ratio of the MTU2S clock (MIϕ) frequency with respect to the output frequency of PLL circuit. If a prohibited value is specified, subsequent operation is not guaranteed. 000: ×1 001: ×1/2 010: ×1/3 011: ×1/4 100: ×1/8 Other than above: Setting prohibited
2 to 0	MPFC[2:0]	011	R/W	MTU2 Clock (MPϕ) Frequency Division Ratio Specify the division ratio of the MTU2 clock (MPϕ) frequency with respect to the output frequency of PLL circuit. If a prohibited value is specified, subsequent operation is not guaranteed. 000: ×1 001: ×1/2 010: ×1/3 011: ×1/4 100: ×1/8 Other than above: Setting prohibited

4.4.2 Oscillation Stop Detection Control Register (OSCCR)

OSCCR is an 8-bit readable/writable register that has an oscillation stop detection flag and selects flag status output to an external pin. OSCCR can be accessed only in bytes.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	OSC STOP	-	OSC ERS
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	OSCSTOP	0	R	Oscillation Stop Detection Flag [Setting conditions] • When a stop in the clock input is detected during normal operation • When software standby mode is entered [Clearing conditions] • By a power-on reset input through the $\overline{\text{RES}}$ pin • When software standby mode is canceled
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	OSCERS	0	R/W	Oscillation Stop Detection Flag Output Select Selects whether to output the oscillation stop detection flag signal through the $\overline{\text{WDTOVF}}$ pin. 0: Outputs only the WDT overflow signal through the $\overline{\text{WDTOVF}}$ pin 1: Outputs the WDT overflow signal and the oscillation stop detection flag signal through the $\overline{\text{WDTOVF}}$ pin

4.5 Changing Frequency

Selecting division ratios for the frequency divider can change the frequencies of the internal clock (Iϕ), bus clock (Bϕ), peripheral clock (Pϕ), MTU2S clock (MIϕ), and MTU2 clock (MPϕ). This is controlled by software through the frequency control register (FRQCR). The following describes how to specify the frequencies.

1. In the initial state, IFC2 to IFC0 = H'011 (×1/4), BFC2 to BFC0 = H'011 (×1/4), PFC2 to PFC0 = H'011 (×1/4), MIFC2 to MIFC0 = H'011 (×1/4), and MPFC2 to MPFC0 = H'011 (×1/4).
2. Stop all modules except the CPU, on-chip ROM, and on-chip RAM.
3. Set the desired values in bits IFC2 to IFC0, BFC2 to BFC0, PFC2 to PFC0, MIFC2 to MIFC0, and MPFC2 to MPFC0 bits. Since the frequency multiplication ratio in the PLL circuit is fixed at ×8, the frequencies are determined only be selecting division ratios. When specifying the frequencies, satisfy the following condition: internal clock (Iϕ) ≥ bus clock (Bϕ) ≥ peripheral clock (Pϕ). When using the MTU2S clock and MTU2 clock, specify the frequencies to satisfy the following condition: internal clock (Iϕ) ≥ MTU2S clock (MIϕ) ≥ MTU2 clock (MPϕ) ≥ peripheral clock (Pϕ) and bus clock (Bϕ) ≥ MTU2 clock (MPϕ).
   Code to rewrite values of FRQCR should be executed in the on-chip ROM or on-chip RAM.
4. After an instruction to rewrite FRQCR has been issued, the actual clock frequencies will change after (1 to 24n) cyc + 11Bϕ + 7Pϕ.
   n: Division ratio specified by the BFC bit in FRQCR (1, 1/2, 1/3, 1/4, or 1/8)
   cyc: Clock obtained by dividing EXTAL by 8 with the PLL.

Note: (1 to 24n) depends on the internal state.

4.6 Oscillator

Clock pulses can be supplied from a connected crystal resonator or an external clock.

4.6.1 Connecting Crystal Resonator

A crystal resonator can be connected as shown in figure 4.2. Use the damping resistance (Rd) listed in table 4.6. Use a crystal resonator that has a resonance frequency of 5 to 12.5 MHz. It is recommended to consult the crystal resonator manufacturer concerning the compatibility of the crystal resonator and the LSI.

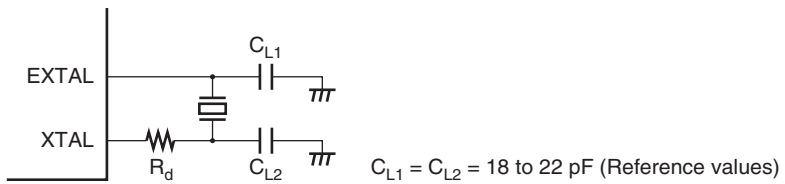


Figure 4.2 Connection of Crystal Resonator (Example)

Table 4.6 Damping Resistance Values (Reference Values)

Frequency (MHz)	5	8	10	12.5
Rd (Ω)(Reference values)	500	200	0	0

Figure 4.3 shows an equivalent circuit of the crystal resonator. Use a crystal resonator with the characteristics listed in table 4.7.

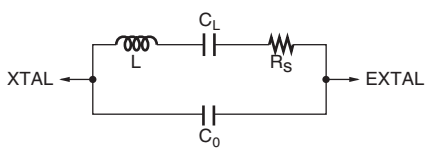


Figure 4.3 Crystal Resonator Equivalent Circuit

Table 4.7 Crystal Resonator Characteristics

Frequency (MHz)	5	8	10	12.5
Rs Max. (Ω) (Reference values)	120	80	60	50
C_0 Max. (pF) (Reference values)	7	7	7	7

4.6.2 External Clock Input Method

Figure 4.4 shows an example of an external clock input connection. In this case, make the external clock high level to stop it when in software standby mode. During operation, make the external input clock frequency 5 to 12.5 MHz.

When leaving the XTAL pin open, make sure the parasitic capacitance is less than 10 pF.

Even when inputting an external clock, be sure to wait at least the oscillation stabilization time in power-on sequence or in releasing software standby mode, in order to ensure the PLL stabilization time.

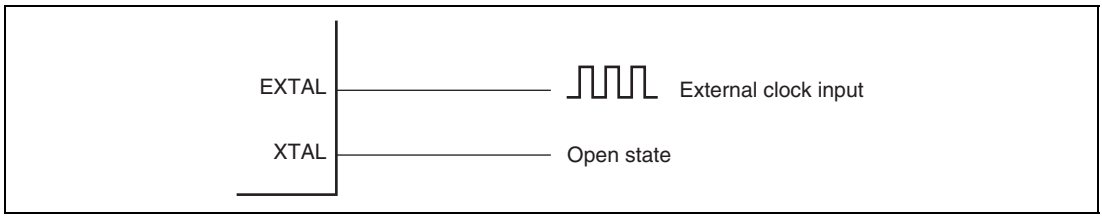


Figure 4.4 Example of External Clock Connection

4.7 Function for Detecting Oscillator Stop

This CPG detects a stop in the clock input if any system abnormality halts the clock supply.

When no change has been detected in the EXTAL input for a certain period, the OSCSTOP bit in OSCCR is set to 1 and this state is retained until a power-on reset is input through the $\overline{\text{RES}}$ pin or software standby mode is canceled. If the OSCERS bit is set to 1 at this time, an oscillation stop detection flag signal is output through the $\overline{\text{WDTOVF}}$ pin. In addition, the high-current ports (pins to which the TIOC3B, TIOC3D, and TIOC4A to TIOC4D signals in the MTU2 and the TIOC3BS, TIOC3DS, and TIOC4AS to TIOC4DS signals in the MTU2S are assigned) can be placed in high-impedance state regardless of the PFC setting. For details, refer to section 21.1.11, High-Current Port Control Register (HCPCR), and appendix A, Pin States.

Even in software standby mode, these pins can be placed in high-impedance state. For details, refer to section 21.1.11, High-Current Port Control Register (HCPCR), and appendix A, Pin States. These pins enter the normal state after software standby mode is canceled. Under an abnormal condition where oscillation stops while the LSI is not in software standby mode, LSI operations other than the oscillation stop detection function become unpredictable. In this case, even after oscillation is restarted, LSI operations including the above high-current pins become unpredictable.

Even while no change is detected in the EXTAL input, the PLL circuit in this LSI continues oscillating at a frequency range from 100 kHz to 10 MHz (depending on the temperature and operating voltage).

4.8 Usage Notes

4.8.1 Note on Crystal Resonator

A sufficient evaluation at the user's site is necessary to use the LSI, by referring the resonator connection examples shown in this section, because various characteristics related to the crystal resonator are closely linked to the user's board design. As the oscillator circuit's circuit constant will depend on the resonator and the floating capacitance of the mounting circuit, the value of each external circuit's component should be determined in consultation with the resonator manufacturer. The design must ensure that a voltage exceeding the maximum rating is not applied to the oscillator pin.

4.8.2 Notes on Board Design

Measures against radiation noise are taken in this LSI. If further reduction in radiation noise is needed, it is recommended to use a multiple layer board and provide a layer exclusive to the system ground.

When using a crystal resonator, place the crystal resonator and its load capacitors as close as possible to the XTAL and EXTAL pins. Do not route any signal lines near the oscillator circuitry as shown in figure 4.5. Otherwise, correct oscillation can be interfered by induction.

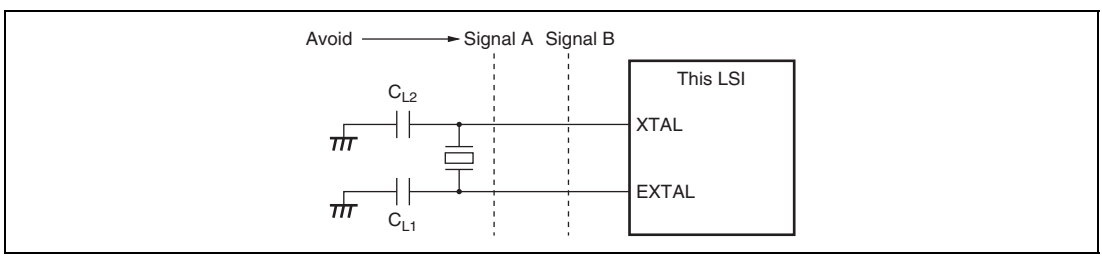
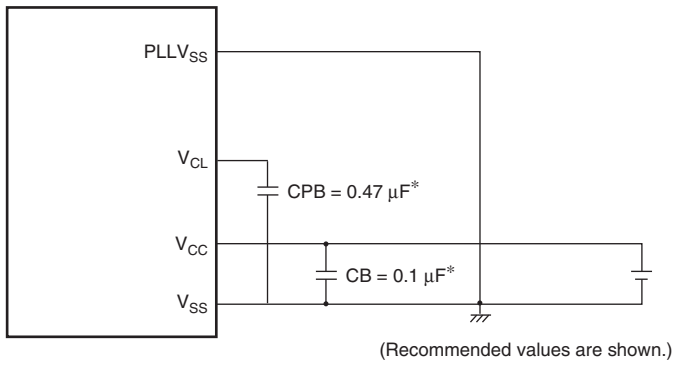


Figure 4.5 Cautions for Oscillator Circuit Board Design

A circuitry shown in figure 4.6 is recommended as an external circuitry around the PLL. Separate the PLL power lines (PLLVss) and the system power lines (Vcc, Vss) at the board power supply source, and be sure to insert bypass capacitors CB and CPB close to the pins.



Note: * CB and CPB are laminated ceramic type.

Figure 4.6 Recommended External Circuitry around PLL

Section 5 Exception Handling

5.1 Overview

5.1.1 Types of Exception Handling and Priority

Exception handling is started by four sources: resets, address errors, interrupts and instructions and have the priority, as shown in table 5.1. When several exceptions are detected at once, they are processed according to the priority.

Table 5.1 Types of Exceptions and Priority

Exception	Exception Source	Priority
Reset	Power-on reset	High
	Manual reset	
Interrupt	User break (break before instruction execution)	
Address error	CPU address error (instruction fetch)	
Instruction	General illegal instructions (undefined code)	
	Illegal slot instruction (undefined code placed immediately after a delayed branch instruction*1 or instruction that changes the PC value*2)	
	Trap instruction (TRAPA instruction)	
Address error	CPU address error (data access)	
Interrupt	User break (break after instruction execution or operand break)	
Address error	DMAC/DTC address error (data access)	
Interrupt	NMI	
	IRQ	
	On-chip peripheral modules	Low

Notes: 1. Delayed branch instructions: JMP, JSR, BRA, BSR, RTS, RTE, BF/S, BT/S, BSRF, and BRAF.
2. Instructions that change the PC value: JMP, JSR, BRA, BSR, RTS, RTE, BT, BF, TRAPA, BF/S, BT/S, BSRF, BRAF, LDC Rm,SR, LDC.L @Rm+,SR.

5.1.2 Exception Handling Operations

The exceptions are detected and the exception handling starts according to the timing shown in table 5.2.

Table 5.2 Timing for Exception Detection and Start of Exception Handling

Exception		Timing of Source Detection and Start of Exception Handling
Reset	Power-on reset	Started when the $\overline{\text{RES}}$ pin changes from low to high or when the WDT overflows.
	Manual reset	Started when the $\overline{\text{MRES}}$ pin changes from low to high or when the WDT overflows.
Address error		Detected during the instruction decode stage and started after the execution of the current instruction is completed.
Interrupt		
Instruction	Trap instruction	Started by the execution of the TRAPA instruction.
	General illegal instructions	Started when an undefined code placed at other than a delay slot (immediately after a delayed branch instruction) is decoded.
	Illegal slot instructions	Started when an undefined code placed at a delay slot (immediately after a delayed branch instruction) or an instruction that changes the PC value is detected.

When exception handling starts, the CPU operates

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 from the address H'00000000 and SP from the address H'00000004 when a power-on reset. PC from the address H'00000008 and SP from the address H'0000000C when a manual reset.). For details, see section 5.1.3, Exception Handling Vector Table. H'00000000 is then written to the vector base register (VBR), and H'F (B'1111) is written to the interrupt mask bits (I3 to I0) in the status register (SR). The program starts from the PC address fetched from the exception handling vector table.

Exception Handling Triggered by Address Error, Interrupt, and Instruction: SR and PC are saved to the stack indicated by R15. For interrupt exception handling, the interrupt priority level is written to the interrupt mask bits (I3 to I0) in SR. For address error and instruction exception handling, bits I3 to I0 are not affected. The start address is then fetched from the exception handling vector table and the program starts from that address.

5.1.3 Exception Handling Vector Table

Before exception handling starts, the exception handling vector table must be set in memory. The exception handling vector table stores the start addresses of exception handling routines. (The reset exception handling table holds the initial values of PC and SP.)

All exception sources are given different vector numbers and vector table address offsets. The vector table addresses are calculated from these vector numbers and vector table address offsets. During exception handling, the start addresses of the exception handling routines are fetched from the exception handling vector table that is indicated by this vector table address.

Table 5.3 shows the vector numbers and vector table address offsets. Table 5.4 shows how vector table addresses are calculated.

Table 5.3 Vector Numbers and Vector Table Address Offsets

Exception Handling Source		Vector Number	Vector Table Address Offset
Power-on reset	PC	0	H'00000000 to H'00000003
	SP	1	H'00000004 to H'00000007
Manual reset	PC	2	H'00000008 to H'0000000B
	SP	3	H'0000000C to H'0000000F
General illegal instruction		4	H'00000010 to H'00000013
(Reserved for system use)		5	H'00000014 to H'00000017
Illegal slot instruction		6	H'00000018 to H'0000001B
(Reserved for system use)		7	H'0000001C to H'0000001F
		8	H'00000020 to H'00000023
CPU address error		9	H'00000024 to H'00000027
DMAC/DTC address error		10	H'00000028 to H'0000002B
Interrupt	NMI	11	H'0000002C to H'0000002F
	User break	12	H'00000030 to H'00000033
(Reserved for system use)		13	H'00000034 to H'00000037
		:	:
		31	H'0000007C to H'0000007F
Trap instruction (user vector)		32	H'00000080 to H'00000083
		:	:
		63	H'000000FC to H'000000FF

Exception Handling Source		Vector Number	Vector Table Address Offset
Interrupt	IRQ0	64	H'00000100 to H'00000103
	IRQ1	65	H'00000104 to H'00000107
	IRQ2	66	H'00000108 to H'0000010B
	IRQ3	67	H'0000010C to H'0000010F
	IRQ4	68	H'00000110 to H'00000113
	IRQ5	69	H'00000114 to H'00000117
	IRQ6	70	H'00000118 to H'0000011B
	IRQ7	71	H'0000011C to H'0000011F
On-chip peripheral module*		72	H'00000120 to H'00000123
		:	:
		255	H'000003FC to H'000003FF

Note: * For details on the vector numbers and vector table address offsets of on-chip peripheral module interrupts, see table 6.3 in section 6, Interrupt Controller (INTC).

Table 5.4 Calculating Exception Handling Vector Table Addresses

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

Notes: 1. VBR: Vector base register
2. Vector table address offset: See table 5.3.
3. Vector number: See table 5.3.

5.2 Resets

5.2.1 Types of Resets

Resets have priority over any exception source. There are two types of resets: power-on resets and manual resets. As table 5.5 shows, both types of resets initialize the internal status of the CPU. In power-on resets, all registers of the on-chip peripheral modules are initialized; in manual resets, they are not.

Table 5.5 Reset Status

	Conditions for Transition to Reset State			Internal State		
Type	$\overline{\text{RES}}$	WDT Overflow	$\overline{\text{MRES}}$	CPU, INTC	On-Chip Peripheral Module	POE, PFC, I/O Port
Power-on reset	Low	—	—	Initialized	Initialized	Initialized
	High	Overflow	High	Initialized	Initialized	Initialized
Manual reset	High	Not overflowed	Low	Initialized	Not initialized	Not initialized

5.2.2 Power-On Reset

Power-On Reset by $\overline{\text{RES}}$ Pin: When the $\overline{\text{RES}}$ pin is driven low, this LSI enters the power-on reset state. To reliably reset this LSI, the $\overline{\text{RES}}$ pin should be kept low for at least the oscillation settling time when applying the power or when in standby mode (when the clock is halted) or at least 20 tcyc when the clock is operating. During the power-on reset state, CPU internal states and all registers of on-chip peripheral modules are initialized. See appendix A, Pin States, for the status of individual pins during power-on reset mode.

In the power-on reset state, power-on reset exception handling starts when driving the $\overline{\text{RES}}$ pin high after driving the pin low for the given time. 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'00000000 and the interrupt mask bits (I3 to I0) of the status register (SR) are set to H'F (B'1111).
4. The values fetched from the exception handling vector table are set in PC and SP, then the program starts.

Be certain to always perform power-on reset exception handling when turning the system power on.

Power-On Reset by WDT: When WTCNT of the WDT overflows while a setting is made so that a power-on reset can be generated in watchdog timer mode of the WDT, this LSI enters the power-on reset state.

The frequency control register (FRQCR) in the clock pulse generator (CPG) and the watchdog timer (WDT) registers are not initialized by the reset signal generated by the WDT (these registers are only initialized by a power-on reset from the $\overline{\text{RES}}$ pin).

If a reset caused by the signal input on the $\overline{\text{RES}}$ pin and a reset caused by a WDT overflow occur simultaneously, the $\overline{\text{RES}}$ pin reset has priority, and the WOVF bit in WTCSR is cleared to 0. When the power-on reset exception handling caused by the WDT is started, 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'00000000 and the interrupt mask bits (I3 to I0) of the status register (SR) are set to H'F (B'1111).
4. The values fetched from the exception handling vector table are set in the PC and SP, then the program starts.

5.2.3 Manual Reset

When the $\overline{\text{RES}}$ pin is high and the $\overline{\text{MRES}}$ pin is driven low, the LSI becomes to be a manual reset state. To reliably reset the LSI, the $\overline{\text{MRES}}$ pin should be kept at low for at least the duration of the oscillation settling time that is set in WDT when in software standby mode (when the clock is halted) or at least 20 t_{cyc} when the clock is operating. During manual reset, the CPU internal status is initialized. Registers of on-chip peripheral modules are not initialized. When the LSI enters manual reset status in the middle of a bus cycle, manual reset exception processing does not start until the bus cycle has ended. Thus, manual resets do not abort bus cycles. However, once $\overline{\text{MRES}}$ is driven low, hold the low level until the CPU becomes to be a manual reset mode after the bus cycle ends. (Keep at low level for at least the longest bus cycle). See appendix A, Pin States, for the status of individual pins during manual reset mode.

In the manual reset status, manual reset exception processing starts when the $\overline{\text{MRES}}$ pin is first kept low for a set period of time and then returned to high. The CPU will then operate in the same procedures as described for power-on resets.

5.3 Address Errors

5.3.1 Address Error Sources

Address errors occur when instructions are fetched or data is read from or written to, as shown in table 5.6.

Table 5.6 Bus Cycles and Address Errors

Bus Cycle			
Type	**Bus Master**	**Bus Cycle Description**	**Address Errors**
Instruction fetch	CPU	Instruction fetched from even address	None (normal)
		Instruction fetched from odd address	Address error occurs
		Instruction fetched from a space other than on-chip peripheral module space	None (normal)
		Instruction fetched from on-chip peripheral module space	Address error occurs
		Instruction fetched from external memory space in single chip mode	Address error occurs
Data read/write	CPU, DMAC, or DTC	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 long-word boundary	Address error occurs
		Byte or word data accessed in on-chip peripheral module space	None (normal)
		Longword data accessed in 16-bit on-chip peripheral module space	None (normal)
		Longword data accessed in 8-bit on-chip peripheral module space	None (normal)
		External memory space accessed when in single chip mode	Address error occurs

5.3.2 Address Error Exception Source

When an address error exception is generated, the bus cycle which caused the address error ends, the current instruction finishes, and then the address error exception handling starts. The CPU operates as follows:

1. The status register (SR) is saved to the stack.
2. The program counter (PC) is saved to the stack. The PC value to be saved is the start address of the instruction which caused an address error exception. When the instruction that caused the exception is placed in the delay slot, the address of the delayed branch instruction which is placed immediately before the delay slot.
3. The start address of the exception handling routine is fetched from the exception handling vector table that corresponds to the generated address error, and the program starts executing from that address. This branch is not a delayed branch.

5.4 Interrupts

5.4.1 Interrupt Sources

Table 5.7 shows the sources that start the interrupt exception handling. They are NMI, user break, IRQ, and on-chip peripheral modules.

Table 5.7 Interrupt Sources

Type	Request Source	Number of Sources
NMI	NMI pin (external input)	1
User break	User break controller (UBC)	1
IRQ	IRQ0 to IRQ7 pins (external input)	8
On-chip peripheral module	Direct memory access controller (DMAC)	8
	Multi-function timer pulse unit 2 (MTU2)	28
	Multi-function timer pulse unit 2S (MTU2S)	13
	Data transfer controller (DTC)	1
	Bus state controller (BSC)	1
	Watchdog timer (WDT)	1
	A/D converter (A/D_0, A/D_1, and A/D_2)	3
	Compare match timer (CMT_0 and CMT_1)	2
	Serial communication interface (SCI_0, SCI_1, and SCI_2)	12
	Serial communication interface with FIFO (SCIF_3)	4
	Synchronous serial communication unit (SSU)	3
	Port output enable (POE)	3
	I^2C bus interface 2 (IIC2)	5

All interrupt sources are given different vector numbers and vector table address offsets. For details on vector numbers and vector table address offsets, see table 6.3 in section 6, Interrupt Controller (INTC).

5.4.2 Interrupt Priority

The interrupt priority is predetermined. When multiple interrupts occur simultaneously (overlapped interruptions), the interrupt controller (INTC) determines their relative priorities and starts the exception handling according to the results.

The priority of interrupts is expressed as priority levels 0 to 16, with priority 0 the lowest and priority 16 the highest. The NMI interrupt has priority 16 and cannot be masked, so it is always accepted. The priority level of the user break interrupt is 15. IRQ interrupt and on-chip peripheral module interrupt priority levels can be set freely using the interrupt priority registers A to F and H to M (IPRA to IPRF and IPRH to IPRM) of the INTC as shown in table 5.8. The priority levels that can be set are 0 to 15. Level 16 cannot be set. For details on IPRA to IPRF, see section 6.3.4, Interrupt Priority Registers A to F and H to M (IPRA to IPRF and IPRH to IPRM).

Table 5.8 Interrupt Priority

Type	Priority Level	Comment
NMI	16	Fixed priority level. Cannot be masked.
User break	15	Fixed priority level. Can be masked.
IRQ	0 to 15	Set with interrupt priority registers A to F and H to M (IPRA to IPRF and IPRH to IPRM).
On-chip peripheral module		

5.4.3 Interrupt Exception Handling

When an interrupt occurs, the interrupt controller (INTC) ascertains its priority level. NMI is always accepted, but other interrupts are only accepted if they have a priority level higher than the priority level set in the interrupt mask bits (I3 to I0) of the status register (SR).

When an interrupt is accepted, exception handling begins. In interrupt exception handling, the CPU saves SR and the program counter (PC) to the stack. The priority level of the accepted interrupt is written to bits I3 to I0 in SR. Although the priority level of the NMI is 16, the value set in bits I3 to I0 is H'F (level 15). Next, the start address of the exception handling routine is fetched from the exception handling vector table for the accepted interrupt, and program execution branches to that address and the program starts. For details on the interrupt exception handling, see section 6.6, Interrupt Operation.

5.5 Exceptions Triggered by Instructions

5.5.1 Types of Exceptions Triggered by Instructions

Exception handling can be triggered by the trap instruction, illegal slot instructions, and general illegal instructions, as shown in table 5.9.

Table 5.9 Types of Exceptions Triggered by Instructions

Type	Source Instruction	Comment
Trap instruction	TRAPA	—
Illegal slot instructions*	Undefined code placed immediately after a delayed branch instruction (delay slot) or instructions that changes the PC value	Delayed branch instructions: JMP, JSR, BRA, BSR, RTS, RTE, BF/S, BT/S, BSRF, BRAF Instructions that changes the PC value: JMP, JSR, BRA, BSR, RTS, RTE, BT, BF, TRAPA, BF/S, BT/S, BSRF, BRAF, LDC Rm,SR, LDC.L @Rm+,SR
General illegal instructions*	Undefined code anywhere besides in a delay slot	—

Note: * The operation is not guaranteed when undefined instructions other than H'F000 to H'FFFF are decoded.

5.5.2 Trap Instructions

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

1. The status register (SR) is saved to the stack.
2. 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.
3. The CPU reads the start address of the exception handling routine from the exception handling vector table that corresponds to the vector number specified in the TRAPA instruction, program execution branches to that address, and then the program starts. This branch is not a delayed branch.

5.5.3 Illegal Slot Instructions

An instruction placed immediately after a delayed branch instruction is called "instruction placed in a delay slot". When the instruction placed in the delay slot is an undefined code, illegal slot exception handling starts after the undefined code is decoded. Illegal slot exception handling also starts when an instruction that changes the program counter (PC) value is placed in a delay slot and the instruction is decoded. The CPU handles an illegal slot instruction as follows:

1. The status register (SR) is saved to the stack.
2. The program counter (PC) is saved to the stack. The PC value saved is the target address of the delayed branch instruction immediately before the undefined code or the instruction that rewrites the PC.
3. The start address of the exception handling routine is fetched from the exception handling vector table that corresponds to the exception that occurred. Program execution branches to that address and the program starts. This branch is not a delayed branch.

5.5.4 General Illegal Instructions

When an undefined code placed anywhere other than immediately after a delayed branch instruction (i.e., in a delay slot) is decoded, general illegal instruction exception handling starts. The CPU handles the general illegal instructions in the same procedures as in the illegal slot instructions. Unlike processing of illegal slot instructions, however, the program counter value that is stacked is the start address of the undefined code.

5.6 Cases when Exceptions Are Accepted

When an exception other than resets occurs during decoding the instruction placed in a delay slot or immediately after an interrupt disabled instruction, it may not be accepted and be held shown in table 5.10. In this case, when an instruction which accepts an interrupt request is decoded, the exception is accepted.

Table 5.10 Delay Slot Instructions, Interrupt Disabled Instructions, and Exceptions

	Exception				
Occurrence Timing	**Address Error**	**General Illegal Instruction**	**Slot Illegal Instruction**	**Trap Instruction**	**Interrupt**
Instruction in delay slot	×*2	—	×*2	—	×*3
Immediately after interrupt disabled instruction*1	√	√	√	√	×*4

[Legend]

√: Accepted

×: Not accepted

—: Does not occur

Notes: 1. Interrupt disabled instructions: LDC, LDC.L, STC, STC.L, LDS, LDS.L, STS, and STS.L

2. An exception is accepted before the execution of a delayed branch instruction. However, when an address error or a slot illegal instruction exception occurs in the delay slot of the RTE instruction, correct operation is not guaranteed.

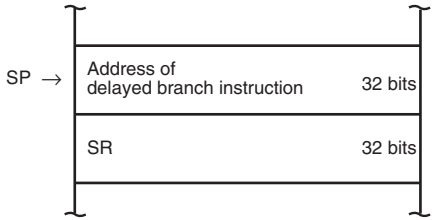
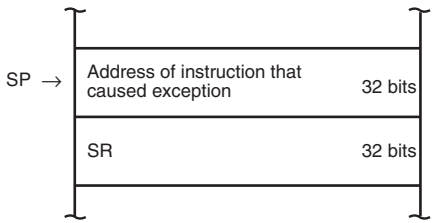
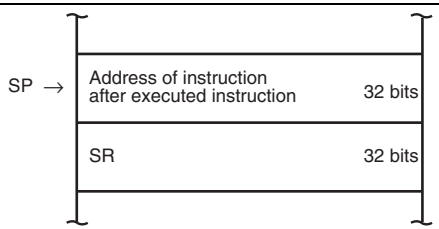
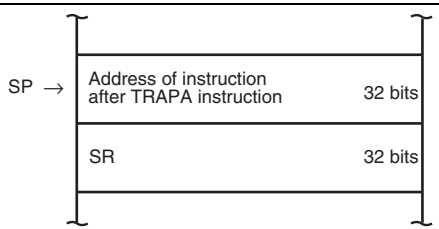
3. An exception is accepted after a delayed branch (between instructions in the delay slot and the branch destination).

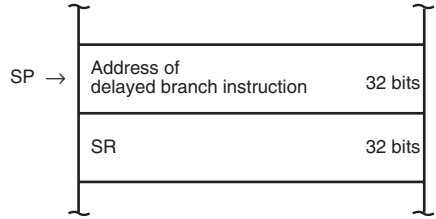
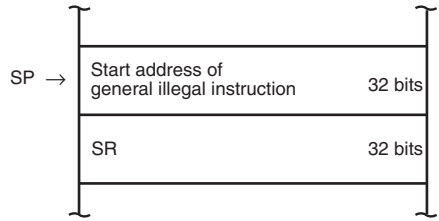
4. An exception is accepted after the execution of the next instruction of an interrupt disabled instruction (before the execution two instructions after an interrupt disabled instruction).

5.7 Stack States after Exception Handling Ends

The stack states after exception handling ends are shown in table 5.11.

Table 5.11 Stack Status after Exception Handling Ends

Types	Stack State
Address error (when the instruction that caused an exception is placed in the delay slot)	
Address error (other than above)	
Interrupt	
Trap instruction	

Types	Stack State
Illegal slot instruction	
General illegal instruction	

5.8 Usage Notes

5.8.1 Value of Stack Pointer (SP)

The SP value must always be a multiple of 4. If it is not, an address error will occur when the stack is accessed during exception handling.

5.8.2 Value of Vector Base Register (VBR)

The VBR value must always be a multiple of 4. If it is not, an address error will occur when the stack is accessed during exception handling.

5.8.3 Address Errors Caused by Stacking for Address Error Exception Handling

When the SP value is not a multiple of 4, an address error will occur when stacking for exception handling (interrupts, etc.) and address error exception handling will start after the first exception handling is ended. Address errors will 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 passed to the handling routine for address error exception and enables error processing.

When an address error occurs during exception handling stacking, the stacking bus cycle (write) is executed. When stacking the SR and PC values, the SP values for both are subtracted by 4, therefore, the SP value is still not a multiple of 4 after the stacking. The address value output during stacking is the SP value whose lower two bits are cleared to 0. So the write data stacked is undefined.

5.8.4 Notes on Slot Illegal Instruction Exception Handling

Some specifications on slot illegal instruction exception handling in this LSI differ from those of the conventional SH-2.

- Conventional SH-2: Instructions LDC Rm,SR and LDC.L @Rm+,SR are not subject to the slot illegal instructions.
- This LSI: Instructions LDC Rm,SR and LDC.L @Rm+,SR are subject to the slot illegal instructions.

The supporting status on our software products regarding this note is as follows:

Compiler

This instruction is not allocated in the delay slot in the compiler V.4 and its subsequent versions.

Real-time OS for μITRON specifications

1. HI7000/4, HI-SH7

   This instruction does not exist in the delay slot within the OS.
2. HI7000

   This instruction is in part allocated to the delay slot within the OS, which may cause the slot illegal instruction exception handling in this LSI.
3. Others

   The slot illegal instruction exception handling may be generated in this LSI in a case where the instruction is described in assembler or when the middleware of the object is introduced.

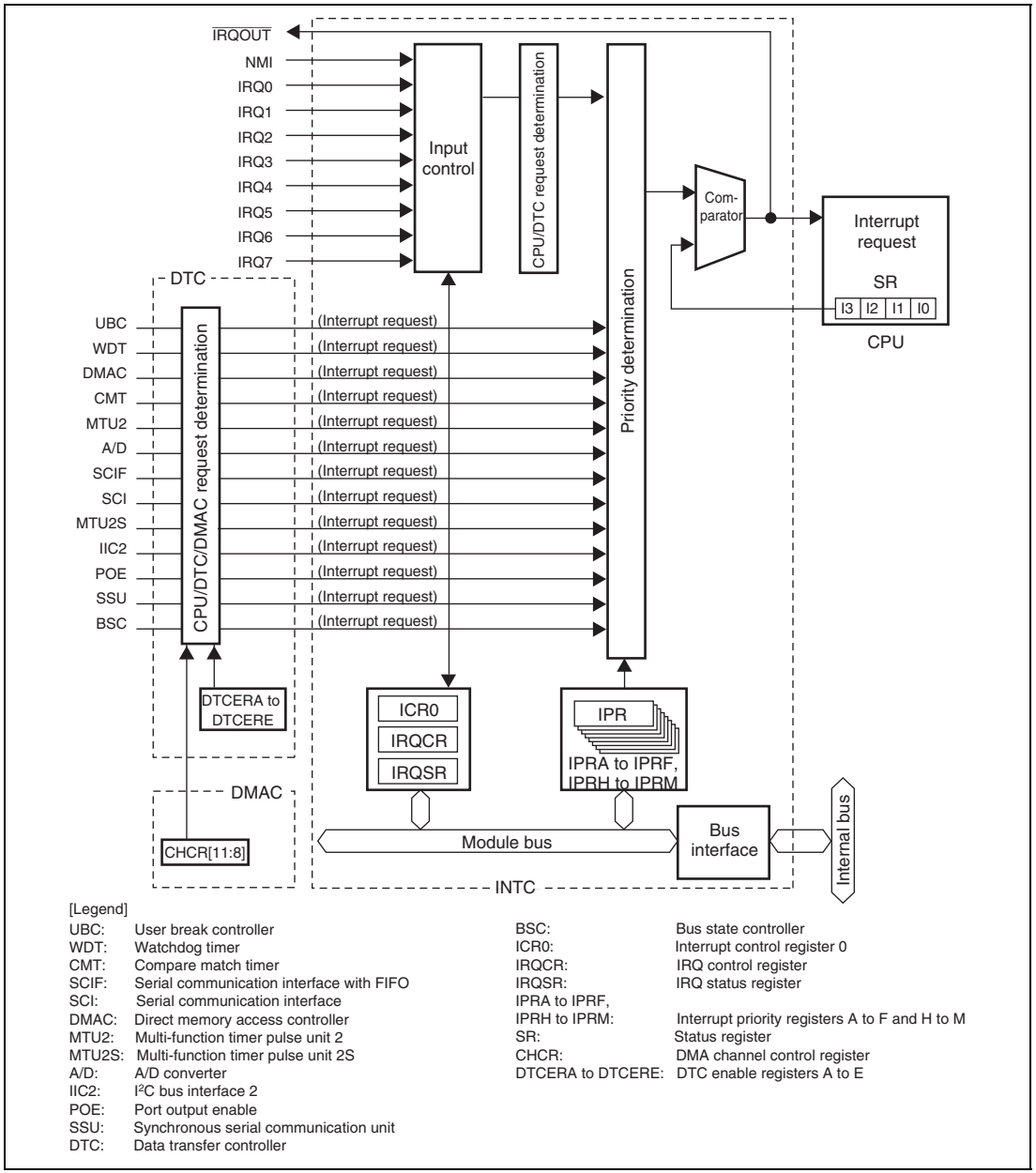
Section 6 Interrupt Controller (INTC)

The interrupt controller (INTC) ascertains the priority of interrupt sources and controls interrupt requests to the CPU.

6.1 Features

- 16 levels of interrupt priority
- NMI noise canceller function
- Occurrence of interrupt can be reported externally ($\overline{\text{IRQOUT}}$ pin)

Figure 6.1 shows a block diagram of the INTC.



[Legend]

UBC:	User break controller	BSC:	Bus state controller
WDT:	Watchdog timer	ICR0:	Interrupt control register 0
CMT:	Compare match timer	IRQCR:	IRQ control register
SCIF:	Serial communication interface with FIFO	IRQSR:	IRQ status register
SCI:	Serial communication interface	IPRA to IPRF, IPRH to IPRM:	Interrupt priority registers A to F and H to M
DMAC:	Direct memory access controller	SR:	Status register
MTU2:	Multi-function timer pulse unit 2	CHCR:	DMA channel control register
MTU2S:	Multi-function timer pulse unit 2S	DTCERA to DTCERE:	DTC enable registers A to E
A/D:	A/D converter		
IIC2:	I^2C bus interface 2		
POE:	Port output enable		
SSU:	Synchronous serial communication unit		
DTC:	Data transfer controller		

Figure 6.1 Block Diagram of INTC

6.2 Input/Output Pins

Table 6.1 shows the INTC pin configuration.

Table 6.1 Pin Configuration

Pin Name	Symbol	I/O	Function
Non-maskable interrupt input pin	NMI	Input	Input of non-maskable interrupt request signal
Interrupt request input pins	IRQ0 to IRQ7	Input	Input of maskable interrupt request signals
Interrupt request output pin	$\overline{\text{IRQOUT}}$	Output	Output of notification signal when an interrupt has occurred

6.3 Register Descriptions

The interrupt controller has the following registers. For details on the addresses of these registers and the states of these registers in each processing state, see section 27, List of Registers.

Table 6.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt control register 0	ICR0	R/W	H'x000	H'FFFFE900	8, 16
IRQ control register	IRQCR	R/W	H'0000	H'FFFFE902	8, 16
IRQ status register	IRQSR	R/W	H'xx00	H'FFFFE904	8, 16
Interrupt priority register A	IPRA	R/W	H'0000	H'FFFFE906	8, 16
Interrupt priority register B	IPRB	R/W	H'0000	H'FFFFE908	8, 16
Interrupt priority register C	IPRC	R/W	H'0000	H'FFFFE980	16
Interrupt priority register D	IPRD	R/W	H'0000	H'FFFFE982	16
Interrupt priority register E	IPRE	R/W	H'0000	H'FFFFE984	16
Interrupt priority register F	IPRF	R/W	H'0000	H'FFFFE986	16
Interrupt priority register H	IPRH	R/W	H'0000	H'FFFFE98A	16
Interrupt priority register I	IPRI	R/W	H'0000	H'FFFFE98C	16
Interrupt priority register J	IPRJ	R/W	H'0000	H'FFFFE98E	16
Interrupt priority register K	IPRK	R/W	H'0000	H'FFFFE990	16
Interrupt priority register L	IPRL	R/W	H'0000	H'FFFFE992	16
Interrupt priority register M	IPRM	R/W	H'0000	H'FFFFE994	16

6.3.1 Interrupt Control Register 0 (ICR0)

ICR0 is a 16-bit register that sets the input signal detection mode of the external interrupt input pin NMI and indicates the input signal level on the NMI pin.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NMIL	-	-	-	-	-	-	NMIE	-	-	-	-	-	-	-	-
Initial value:	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

Note: * The initial value is 1 when the level on the NMI pin is high, and 0 when the level on the pin is low.

Bit	Bit Name	Initial Value	R/W	Description
15	NMIL	*	R	NMI Input Level Indicates the state of the signal input to the NMI pin. This bit can be read to determine the NMI pin level. This bit cannot be modified. 0: State of the NMI input is low 1: State of the NMI input is high
14 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	NMIE	0	R/W	NMI Edge Select 0: Interrupt request is detected on the falling edge of the NMI input 1: Interrupt request is detected on the rising edge of the NMI input
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

6.3.2 IRQ Control Register (IRQCR)

IRQCR is a 16-bit register that sets the input signal detection mode of the external interrupt input pins IRQ0 to IRQ7.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ71S	IRQ70S	IRQ61S	IRQ60S	IRQ51S	IRQ50S	IRQ41S	IRQ40S	IRQ31S	IRQ30S	IRQ21S	IRQ20S	IRQ11S	IRQ10S	IRQ01S	IRQ00S
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	IRQ71S	0	R/W	IRQ7 Sense Select
14	IRQ70S	0	R/W	Set the interrupt request detection mode for pin IRQ7. 00: Interrupt request is detected at the low level of pin IRQ7 01: Interrupt request is detected at the falling edge of pin IRQ7 10: Interrupt request is detected at the rising edge of pin IRQ7 11: Interrupt request is detected at both the falling and rising edges of pin IRQ7
13	IRQ61S	0	R/W	IRQ6 Sense Select
12	IRQ60S	0	R/W	Set the interrupt request detection mode for pin IRQ6. 00: Interrupt request is detected at the low level of pin IRQ6 01: Interrupt request is detected at the falling edge of pin IRQ6 10: Interrupt request is detected at the rising edge of pin IRQ6 11: Interrupt request is detected at both the falling and rising edges of pin IRQ6

Bit	Bit Name	Initial Value	R/W	Description
11	IRQ51S	0	R/W	IRQ5 Sense Select
10	IRQ50S	0	R/W	Set the interrupt request detection mode for pin IRQ5. 00: Interrupt request is detected at the low level of pin IRQ5 01: Interrupt request is detected at the falling edge of pin IRQ5 10: Interrupt request is detected at the rising edge of pin IRQ5 11: Interrupt request is detected at both the falling and rising edges of pin IRQ5
9	IRQ41S	0	R/W	IRQ4 Sense Select
8	IRQ40S	0	R/W	Set the interrupt request detection mode for pin IRQ4. 00: Interrupt request is detected at the low level of pin IRQ4 01: Interrupt request is detected at the falling edge of pin IRQ4 10: Interrupt request is detected at the rising edge of pin IRQ4 11: Interrupt request is detected at both the falling and rising edges of pin IRQ4
7	IRQ31S	0	R/W	IRQ3 Sense Select
6	IRQ30S	0	R/W	Set the interrupt request detection mode for pin IRQ3. 00: Interrupt request is detected at the low level of pin IRQ3 01: Interrupt request is detected at the falling edge of pin IRQ3 10: Interrupt request is detected at the rising edge of pin IRQ3 11: Interrupt request is detected at both the falling and rising edges of pin IRQ3

Bit	Bit Name	Initial Value	R/W	Description
5	IRQ21S	0	R/W	IRQ2 Sense Select
4	IRQ20S	0	R/W	Set the interrupt request detection mode for pin IRQ2. 00: Interrupt request is detected at the low level of pin IRQ2 01: Interrupt request is detected at the falling edge of pin IRQ2 10: Interrupt request is detected at the rising edge of pin IRQ2 11: Interrupt request is detected at both the falling and rising edges of pin IRQ2
3	IRQ11S	0	R/W	IRQ1 Sense Select
2	IRQ10S	0	R/W	Set the interrupt request detection mode for pin IRQ1. 00: Interrupt request is detected at the low level of pin IRQ1 01: Interrupt request is detected at the falling edge of pin IRQ1 10: Interrupt request is detected at the rising edge of pin IRQ1 11: Interrupt request is detected at both the falling and rising edges of pin IRQ1
1	IRQ01S	0	R/W	IRQ0 Sense Select
0	IRQ00S	0	R/W	Set the interrupt request detection mode for pin IRQ0. 00: Interrupt request is detected at the low level of pin IRQ0 01: Interrupt request is detected at the falling edge of pin IRQ0 10: Interrupt request is detected at the rising edge of pin IRQ0 11: Interrupt request is detected at both the falling and rising edges of pin IRQ0

6.3.3 IRQ Status register (IRQSR)

IRQSR is a 16-bit register that indicates the states of the external interrupt input pins IRQ0 to IRQ7 and the status of interrupt request.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ7L	IRQ6L	IRQ5L	IRQ4L	IRQ3L	IRQ2L	IRQ1L	IRQ0L	IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
Initial value:	*	*	*	*	*	*	*	*	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * The initial value is 1 when the level on the corresponding IRQ pin is high, and 0 when the level on the pin is low.

Bit	Bit Name	Initial Value	R/W	Description
15	IRQ7L	*	R	Indicates the state of pin IRQ7. 0: State of pin IRQ7 is low 1: State of pin IRQ7 is high
14	IRQ6L	*	R	Indicates the state of pin IRQ6. 0: State of pin IRQ6 is low 1: State of pin IRQ6 is high
13	IRQ5L	*	R	Indicates the state of pin IRQ5. 0: State of pin IRQ5 is low 1: State of pin IRQ5 is high
12	IRQ4L	*	R	Indicates the state of pin IRQ4. 0: State of pin IRQ4 is low 1: State of pin IRQ4 is high
11	IRQ3L	*	R	Indicates the state of pin IRQ3. 0: State of pin IRQ3 is low 1: State of pin IRQ3 is high
10	IRQ2L	*	R	Indicates the state of pin IRQ2. 0: State of pin IRQ2 is low 1: State of pin IRQ2 is high
9	IRQ1L	*	R	Indicates the state of pin IRQ1. 0: State of pin IRQ1 is low 1: State of pin IRQ1 is high

Bit	Bit Name	Initial Value	R/W	Description
8	IRQ0L	*	R	Indicates the state of pin IRQ0. 0: State of pin IRQ0 is low 1: State of pin IRQ0 is high
7	IRQ7F	0	R/W	Indicates the status of an IRQ7 interrupt request. • When level detection mode is selected 0: An IRQ7 interrupt has not been detected [Clearing condition] Driving pin IRQ7 high 1: An IRQ7 interrupt has been detected [Setting condition] Driving pin IRQ7 low • When edge detection mode is selected 0: An IRQ7 interrupt has not been detected [Clearing conditions] — Writing 0 after reading IRQ7F = 1 — Accepting an IRQ7 interrupt 1: An IRQ7 interrupt request has been detected [Setting condition] Detecting the specified edge of pin IRQ7

Bit	Bit Name	Initial Value	R/W	Description
6	IRQ6F	0	R/W	Indicates the status of an IRQ6 interrupt request. • When level detection mode is selected 0: An IRQ6 interrupt has not been detected [Clearing condition] Driving pin IRQ6 high 1: An IRQ6 interrupt has been detected [Setting condition] Driving pin IRQ6 low • When edge detection mode is selected 0: An IRQ6 interrupt has not been detected [Clearing conditions] — Writing 0 after reading IRQ6F = 1 — Accepting an IRQ6 interrupt 1: An IRQ6 interrupt request has been detected [Setting condition] Detecting the specified edge of pin IRQ6
5	IRQ5F	0	R/W	Indicates the status of an IRQ5 interrupt request. • When level detection mode is selected 0: An IRQ5 interrupt has not been detected [Clearing condition] Driving pin IRQ5 high 1: An IRQ5 interrupt has been detected [Setting condition] Driving pin IRQ5 low • When edge detection mode is selected 0: An IRQ5 interrupt has not been detected [Clearing conditions] — Writing 0 after reading IRQ5F = 1 — Accepting an IRQ5 interrupt 1: An IRQ5 interrupt request has been detected [Setting condition] Detecting the specified edge of pin IRQ5

Bit	Bit Name	Initial Value	R/W	Description
4	IRQ4F	0	R/W	Indicates the status of an IRQ4 interrupt request. • When level detection mode is selected 0: An IRQ4 interrupt has not been detected [Clearing condition] Driving pin IRQ4 high 1: An IRQ4 interrupt has been detected [Setting condition] Driving pin IRQ4 low • When edge detection mode is selected 0: An IRQ4 interrupt has not been detected [Clearing conditions] — Writing 0 after reading IRQ4F = 1 — Accepting an IRQ4 interrupt 1: An IRQ4 interrupt request has been detected [Setting condition] Detecting the specified edge of pin IRQ4
3	IRQ3F	0	R/W	Indicates the status of an IRQ3 interrupt request. • When level detection mode is selected 0: An IRQ3 interrupt has not been detected [Clearing condition] Driving pin IRQ3 high 1: An IRQ3 interrupt has been detected [Setting condition] Driving pin IRQ3 low • When edge detection mode is selected 0: An IRQ3 interrupt has not been detected [Clearing conditions] — Writing 0 after reading IRQ3F = 1 — Accepting an IRQ3 interrupt 1: An IRQ3 interrupt request has been detected [Setting condition] Detecting the specified edge of pin IRQ3

Bit	Bit Name	Initial Value	R/W	Description
2	IRQ2F	0	R/W	Indicates the status of an IRQ2 interrupt request. • When level detection mode is selected 0: An IRQ2 interrupt has not been detected [Clearing condition] Driving pin IRQ2 high 1: An IRQ2 interrupt has been detected [Setting condition] Driving pin IRQ2 low • When edge detection mode is selected 0: An IRQ2 interrupt has not been detected [Clearing conditions] — Writing 0 after reading IRQ2F = 1 — Accepting an IRQ2 interrupt 1: An IRQ2 interrupt request has been detected [Setting condition] Detecting the specified edge of pin IRQ2
1	IRQ1F	0	R/W	Indicates the status of an IRQ1 interrupt request. • When level detection mode is selected 0: An IRQ1 interrupt has not been detected [Clearing condition] Driving pin IRQ1 high 1: An IRQ1 interrupt has been detected [Setting condition] Driving pin IRQ1 low • When edge detection mode is selected 0: An IRQ1 interrupt has not been detected [Clearing conditions] — Writing 0 after reading IRQ1F = 1 — Accepting an IRQ1 interrupt 1: An IRQ1 interrupt request has been detected [Setting condition] Detecting the specified edge of pin IRQ1

Bit	Bit Name	Initial Value	R/W	Description
0	IRQ0F	0	R/W	Indicates the status of an IRQ0 interrupt request. • When level detection mode is selected 0: An IRQ0 interrupt has not been detected [Clearing condition] Driving pin IRQ0 high 1: An IRQ0 interrupt has been detected [Setting condition] Driving pin IRQ0 low • When edge detection mode is selected 0: An IRQ0 interrupt has not been detected [Clearing conditions] — Writing 0 after reading IRQ0F = 1 — Accepting an IRQ0 interrupt 1: An IRQ0 interrupt request has been detected [Setting condition] Detecting the specified edge of pin IRQ0

Note: * The initial value is 1 when the level on the corresponding IRQ pin is high, and 0 when the level on the pin is low.

6.3.4 Interrupt Priority Registers A to F and H to M (IPRA to IPRF and IPRH to IPRM)

Interrupt priority registers are thirteen 16-bit readable/writable registers that set priority levels from 0 to 15 for interrupts except NMI. For the correspondence between interrupt request sources and IPR, refer to table 6.3. Each of the corresponding interrupt priority ranks are established by setting a value from H'0 to H'F in each of the four-bit groups 15 to 12, 11 to 8, 7 to 4 and 3 to 0. Reserved bits that are not assigned should be set H'0 (B'0000).

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	IPR[15:12]	0000	R/W	Set priority levels for the corresponding interrupt source. 0000: Priority level 0 (lowest) 0001: Priority level 1 0010: Priority level 2 0011: Priority level 3 0100: Priority level 4 0101: Priority level 5 0110: Priority level 6 0111: Priority level 7 1000: Priority level 8 1001: Priority level 9 1010: Priority level 10 1011: Priority level 11 1100: Priority level 12 1101: Priority level 13 1110: Priority level 14 1111: Priority level 15 (highest)
11 to 8	IPR[11:8]	0000	R/W	Set priority levels for the corresponding interrupt source. 0000: Priority level 0 (lowest) 0001: Priority level 1 0010: Priority level 2 0011: Priority level 3 0100: Priority level 4 0101: Priority level 5 0110: Priority level 6 0111: Priority level 7 1000: Priority level 8 1001: Priority level 9 1010: Priority level 10 1011: Priority level 11 1100: Priority level 12 1101: Priority level 13 1110: Priority level 14 1111: Priority level 15 (highest)

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	IPR[7:4]	0000	R/W	Set priority levels for the corresponding interrupt source. 0000: Priority level 0 (lowest) 0001: Priority level 1 0010: Priority level 2 0011: Priority level 3 0100: Priority level 4 0101: Priority level 5 0110: Priority level 6 0111: Priority level 7 1000: Priority level 8 1001: Priority level 9 1010: Priority level 10 1011: Priority level 11 1100: Priority level 12 1101: Priority level 13 1110: Priority level 14 1111: Priority level 15 (highest)
3 to 0	IPR[3:0]	0000	R/W	Set priority levels for the corresponding interrupt source. 0000: Priority level 0 (lowest) 0001: Priority level 1 0010: Priority level 2 0011: Priority level 3 0100: Priority level 4 0101: Priority level 5 0110: Priority level 6 0111: Priority level 7 1000: Priority level 8 1001: Priority level 9 1010: Priority level 10 1011: Priority level 11 1100: Priority level 12 1101: Priority level 13 1110: Priority level 14 1111: Priority level 15 (highest)

Note: Name in the tables above is represented by a general name. Name in the list of register is, on the other hand, represented by a module name.

6.4 Interrupt Sources

6.4.1 External Interrupts

There are four types of interrupt sources: User break, NMI, IRQ, and on-chip peripheral modules. Individual interrupts are given priority levels (0 to 16, with 0 the lowest and 16 the highest). Giving an interrupt a priority level of 0 masks it.

NMI Interrupt: The NMI interrupt is given a priority level of 16 and is always accepted. An NMI interrupt is detected at the edge of the pins. Use the NMI edge select bit (NMIE) in interrupt control register 0 (ICR0) to select either the rising or falling edge. In the NMI interrupt exception handler, the interrupt mask level bits (I3 to I0) in the status register (SR) are set to level 15.

IRQ7 to IRQ0 Interrupts: IRQ interrupts are requested by input from pins IRQ0 to IRQ7. Use the IRQ sense select bits (IRQ71S to IRQ 01S and IRQ70S to IRQ00S) in the IRQ control register (IRQCR) to select the detection mode from low level detection, falling edge detection, rising edge detection, and both edge detection for each pin. The priority level can be set from 0 to 15 for each pin using the interrupt priority registers A and B (IPRA and IPRB).

In the case that the low level detection is selected, an interrupt request signal is sent to the INTC while the IRQ pin is driven low. The interrupt request signal stops to be sent to the INTC when the IRQ pin becomes high. It is possible to confirm that an interrupt is requested by reading the IRQ flags (IRQ7F to IRQ0F) in the IRQ status register (IRQSR).

In the case that the edge detection is selected, an interrupt request signal is sent to the INTC when the following change on the IRQ pin is detected: from high to low in falling edge detection mode, from low to high in rising edge detection mode, and from low to high or from high to low in both edge detection mode. The IRQ interrupt request by detecting the change on the pin is held until the interrupt request is accepted. It is possible to confirm that an IRQ interrupt request has been detected by reading the IRQ flags (IRQ7F to IRQ0F) in the IRQ status register (IRQSR). An IRQ interrupt request by detecting the change on the pin can be withdrawn by writing 0 to an IRQ flag after reading 1.

In the IRQ interrupt exception handling, the interrupt mask bits (I3 to I0) in the status register (SR) are set to the priority level value of the accepted IRQ interrupt. Figure 6.2 shows the block diagram of the IRQ7 to IRQ0 interrupts.

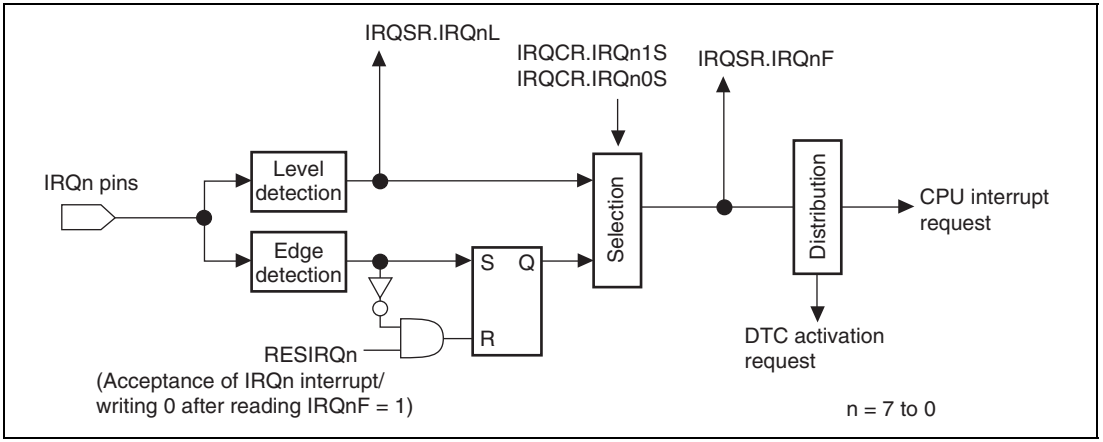


Figure 6.2 Block Diagram of IRQ7 to IRQ0 Interrupts Control

6.4.2 On-Chip Peripheral Module Interrupts

On-chip peripheral module interrupts are interrupts generated by the following on-chip peripheral modules.

Since a different interrupt vector is allocated to each interrupt source, the exception handling routine does not have to decide which interrupt has occurred. Priority levels between 0 and 15 can be allocated to individual on-chip peripheral modules in interrupt priority registers C to F and H to M (IPRC to IPRF and IPRH to IPRM). On-chip peripheral module interrupt exception handling sets the interrupt mask level bits (I3 to I0) in the status register (SR) to the priority level value of the on-chip peripheral module interrupt that was accepted.

6.4.3 User Break Interrupt

A user break interrupt has a priority level of 15, and occurs when the break condition set in the user break controller (UBC) is satisfied. User break interrupt requests are detected by edge and are held until accepted. User break interrupt exception handling sets the interrupt mask level bits (I3 to I0) in the status register (SR) to level 15. For more details on the user break interrupt, see section 7, User Break Controller (UBC).

6.5 Interrupt Exception Handling Vector Table

Table 6.3 lists interrupt sources, their vector numbers, vector table address offsets, and interrupt priorities.

Individual interrupt sources are allocated to different vector numbers and vector table address offsets. Vector table addresses are calculated from the vector numbers and vector table address offsets. For interrupt exception handling, the start address of the exception handling routine is fetched from the vector table address in the vector table. For the details on calculation of vector table addresses, see table 5.4 in section 5, Exception Handling.

IRQ interrupts and on-chip peripheral module interrupt priorities can be set freely between 0 and 15 for each pin or module by setting interrupt priority registers A to F and H to M (IPRA to IPRF and IPRH to IPRM). However, when interrupt sources whose priority levels are allocated with the same IPR are requested, the interrupt of the smaller vector number has priority. This priority cannot be changed. Priority levels of IRQ interrupts and on-chip peripheral module interrupts are initialized to level 0 at a power-on reset. If the same priority level is allocated to two or more interrupt sources and interrupts from those sources occur simultaneously, they are processed by the default priority order shown in table 6.3.

Table 6.3 Interrupt Exception Handling Vectors and Priorities

Interrupt Source	Name	Vector No.	Vector Table Starting Address	IPR	Default Priority
User break		12	H'00000030	—	High
External pin	NMI	11	H'0000002C	—	
	IRQ0	64	H'00000100	IPRA15 to IPRA12	
	IRQ1	65	H'00000104	IPRA11 to IPRA8	
	IRQ2	66	H'00000108	IPRA7 to IPRA4	
	IRQ3	67	H'0000010C	IPRA3 to IPRA0	
	IRQ4	68	H'00000110	IPRB15 to IPRB12	
	IRQ5	69	H'00000114	IPRB11 to IPRB8	
	IRQ6	70	H'00000118	IPRB7 to IPRB4	
	IRQ7	71	H'0000011C	IPRB3 to IPRB0	
DMAC_0	DEI0	72	H'00000120	IPRC15 to IPRC12	
DMAC_1	DEI1	76	H'00000130	IPRC11 to IPRC8	
DMAC_2	DEI2	80	H'00000140	IPRC7 to IPRC4	
DMAC_3	DEI3	84	H'00000150	IPRC3 to IPRC0	
MTU2_0	TGIA_0	88	H'00000160	IPRD15 to IPRD12	
	TGIB_0	89	H'00000164		
	TGIC_0	90	H'00000168		
	TGID_0	91	H'0000016C		
	TCIV_0	92	H'00000170	IPRD11 to IPRD8	
	TGIE_0	93	H'00000174		
	TGIF_0	94	H'00000178		
MTU2_1	TGIA_1	96	H'00000180	IPRD7 to IPRD4	
	TGIB_1	97	H'00000184		
	TCIV_1	100	H'00000190	IPRD3 to IPRD0	
	TCIU_1	101	H'00000194		
MTU2_2	TGIA_2	104	H'000001A0	IPRE15 to IPRE12	
	TGIB_2	105	H'000001A4		
	TCIV_2	108	H'000001B0	IPRE11 to IPRE8	
	TCIU_2	109	H'000001B4		Low

Interrupt Source	Name	Vector No.	Vector Table Starting Address	IPR	Default Priority
MTU2_3	TGIA_3	112	H'000001C0	IPRE7 to IPRE4	High
	TGIB_3	113	H'000001C4		
	TGIC_3	114	H'000001C8		
	TGID_3	115	H'000001CC		
	TCIV_3	116	H'000001D0	IPRE3 to IPRE0	
MTU2_4	TGIA_4	120	H'000001E0	IPRF15 to IPRF12	
	TGIB_4	121	H'000001E4		
	TGIC_4	122	H'000001E8		
	TGID_4	123	H'000001EC		
	TCIV_4	124	H'000001F0	IPRF11 to IPRF8	
MTU2_5	TGIU_5	128	H'00000200	IPRF7 to IPRF4	
	TGIV_5	129	H'00000204		
	TGIW_5	130	H'00000208		
POE (MTU2)	OEI1	132	H'00000210	IPRF3 to IPRF0	
	OEI3	133	H'00000214		
IIC2*	IINAKI	156	H'00000270	IPRH11 to IPRH8	
MTU2S_3	TGIA_3S	160	H'00000280	IPRH7 to IPRH4	
	TGIB_3S	161	H'00000284		
	TGIC_3S	162	H'00000288		
	TGID_3S	163	H'0000028C		
	TCIV_3S	164	H'00000290	IPRH3 to IPRH0	
MTU2S_4	TGIA_4S	168	H'000002A0	IPRI15 to IPRI12	
	TGIB_4S	169	H'000002A4		
	TGIC_4S	170	H'000002A8		
	TGID_4S	171	H'000002AC		
	TCIV_4S	172	H'000002B0	IPRI11 to IPRI8	
MTU2S_5	TGIU_5S	176	H'000002C0	IPRI7 to IPRI4	
	TGIV_5S	177	H'000002C4		
	TGIW_5S	178	H'000002C8		
POE (MTU2S)	OEI2	180	H'000002D0	IPRI3 to IPRI0	
CMT_0	CMI_0	184	H'000002E0	IPRJ15 to IPRJ12	Low

Interrupt Source	Name	Vector No.	Vector Table Starting Address	IPR	Default Priority
CMT_1	CMI_1	188	H'000002F0	IPRJ11 to IPRJ8	High
BSC	CMI	192	H'00000300	IPRJ7 to IPRJ4	
WDT	ITI	196	H'00000310	IPRJ3 to IPRJ0	
A/D_0 and A/D_1	ADI_0	200	H'00000320	IPRK15 to IPRK12	
	ADI_1	201	H'00000324		
A/D_2	ADI_2	204	H'00000330	IPRK11 to IPRK8	
SCI_0	ERI_0	216	H'00000360	IPRL15 to IPRL12	
	RXI_0	217	H'00000364		
	TXI_0	218	H'00000368		
	TEI_0	219	H'0000036C		
SCI_1	ERI_1	220	H'00000370	IPRL11 to IPRL8	
	RXI_1	221	H'00000374		
	TXI_1	222	H'00000378		
	TEI_1	223	H'0000037C		
SCI_2	ERI_2	224	H'00000380	IPRL7 to IPRL4	
	RXI_2	225	H'00000384		
	TXI_2	226	H'00000388		
	TEI_2	227	H'0000038C		
SCIF	ERIF	228	H'00000390	IPRL3 to IPRL0	
	RXIF	229	H'00000394		
	BRIF	230	H'00000398		
	TXIF	231	H'0000039C		
SSU	SSERI	232	H'000003A0	IPRM15 to IPRM12	
	SSRXI	233	H'000003A4		
	SSTXI	234	H'000003A8		
IIC2*	IITEI	236	H'000003B0	IPRM11 to IPRM8	
	IISTPI	237	H'000003B4		
	IITXI	238	H'000003B8		
	IIRXI	239	H'000003BC		Low

Note: * The IIC2 has an interrupt whose vector address is separated from those for the other IIC2 interrupt sources.

6.6 Interrupt Operation

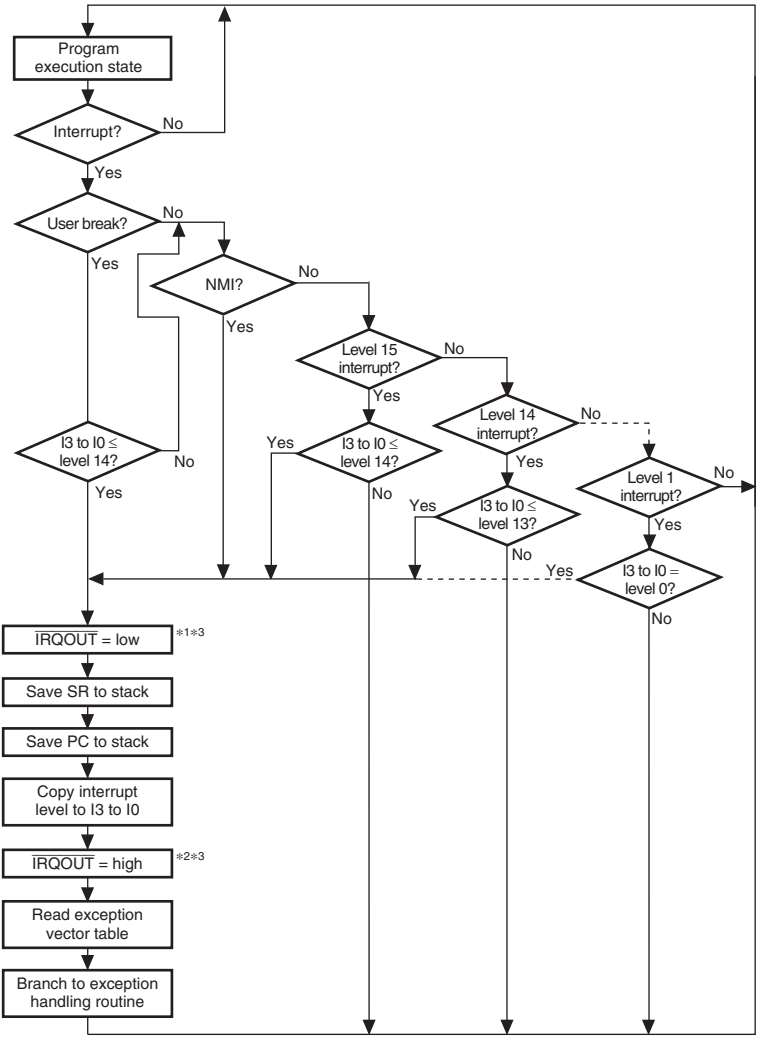
6.6.1 Interrupt Sequence

The sequence of interrupt operations is explained below. Figure 6.3 is a flowchart of the operations.

1. The interrupt request sources send interrupt request signals to the interrupt controller.
2. The interrupt controller selects the highest priority interrupt from interrupt requests sent, according to the priority levels set in interrupt priority registers A to F and H to M (IPRA to IPRF and IPRH to IPRM). Interrupts that have lower-priority than that of the selected interrupt are ignored*. If interrupts that have the same priority level or interrupts within a same module occur simultaneously, the interrupt with the highest priority is selected according to the default priority shown in table 6.3.
3. The interrupt controller compares the priority level of the selected interrupt request with the interrupt mask bits (I3 to I0) in the status register (SR) of the CPU. If the priority level of the selected request is equal to or less than the level set in bits I3 to I0, the request is ignored. If the priority level of the selected request is higher than the level in bits I3 to I0, the interrupt controller accepts the request and sends an interrupt request signal to the CPU.
4. When the interrupt controller accepts an interrupt, a low level is output from the $\overline{\text{IRQOUT}}$ pin.
5. The CPU detects the interrupt request sent from the interrupt controller in the decode stage of an instruction to be executed. Instead of executing the decoded instruction, the CPU starts interrupt exception handling.
6. SR and PC are saved onto the stack.
7. The priority level of the accepted interrupt is copied to bits (I3 to I0) in SR.
8. When the accepted interrupt is sensed by level or is from an on-chip peripheral module, a high level is output from the $\overline{\text{IRQOUT}}$ pin. When the accepted interrupt is sensed by edge, a high level is output from the $\overline{\text{IRQOUT}}$ pin at the moment when the CPU starts interrupt exception processing instead of instruction execution as noted in 5. above. However, if the interrupt controller accepts an interrupt with a higher priority than the interrupt just to be accepted, the $\overline{\text{IRQOUT}}$ pin holds low level.
9. The CPU reads the start address of the exception handling routine from the exception vector table for the accepted interrupt, branches to that address, and starts executing the program. This branch is not a delayed branch.

Notes: The interrupt source flag should be cleared in the interrupt handler. To ensure that an interrupt source that should have been cleared is not inadvertently accepted again, read the interrupt source flag after it has been cleared, confirm that it has been cleared, and then execute an RTE instruction.

* Interrupt requests that are designated as edge-detect type are held pending until the interrupt requests are accepted. IRQ interrupts, however, can be cancelled by accessing the IRQ status register (IRQSR). Interrupts held pending due to edge detection are cleared by a power-on reset or a manual reset.



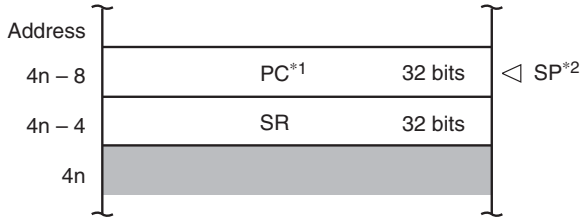
Notes: I3 to I0 are interrupt mask bits in the status register (SR) of the CPU

1. $\overline{\text{IRQOUT}}$ is the same signal as the interrupt request signal to the CPU (see figure 6.1). Therefore, $\overline{\text{IRQOUT}}$ is output when the request priority level is higher than the level in bits I3–I0 of SR.
2. When the accepted interrupt is sensed by edge, a high level is output from the $\overline{\text{IRQOUT}}$ pin at the moment when the CPU starts interrupt exception processing instead of instruction execution (namely, before saving SR to stack). However, if the interrupt controller accepts an interrupt with a higher priority than the interrupt just to be accepted and has output an interrupt request to the CPU, the $\overline{\text{IRQOUT}}$ pin holds low level.
3. The $\overline{\text{IRQOUT}}$ pin change timing depends on a frequency dividing ratio between the internal (Iϕ) and bus (Bϕ) clocks. This flowchart shows that the frequency dividing ratios of the internal (Iϕ) and bus (Bϕ) clocks are the same.

Figure 6.3 Interrupt Sequence Flowchart

6.6.2 Stack after Interrupt Exception Handling

Figure 6.4 shows the stack after interrupt exception handling.



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

Figure 6.4 Stack after Interrupt Exception Handling

6.7 Interrupt Response Time

Table 6.4 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 of the interrupt handling routine begins.

Table 6.4 Interrupt Response Time

Item		Number of Cycles: NMI	Number of Cycles: IRQ	Number of Cycles: Peripheral Modules	Remarks
DMAC/DTC active judgement		—	2 × Bcyc	1 × Pcyc	
Interrupt priority decision and comparison with mask bits in SR		1 × Icyc + 2 × Pcyc	1 × Icyc + 1 × Pcyc	1 × Icyc + 2 × Pcyc	
Wait for completion of sequence currently being executed by CPU		X (≥ 0)	X (≥ 0)	X (≥ 0)	The longest sequence is for interrupt or address-error exception handling (X = 7 × Icyc + m1 + m2 + m3 + m4). If an interrupt-masking instruction follows, however, the time may be even longer.
Time from start of interrupt exception handling until fetch of first instruction of exception handling routine starts		8 × Icyc + m1 + m2 + m3	8 × Icyc + m1 + m2 + m3	8 × Icyc + m1 + m2 + m3	Performs the saving PC and SR, and vector address fetch.
Interrupt response time	Total:	9 × Icyc + 2 × Pcyc + m1 + m2 + m3 + X	9 × Icyc + 1 × Pcyc +2 × Bcyc + m1 + m2 + m3 + X	9 × Icyc + 3 × Pcyc + m1 + m2 + m3 + X	
	Minimum*:	12 × Icyc + 2 × Pcyc	12 × Icyc + 1 × Pcyc + 2 × Bcyc	12 × Icyc + 3 × Pcyc	SR, PC, and vector table are all in on-chip RAM.
	Maximum:	16 × Icyc + 2 × Pcyc + 2 × (m1 + m2 + m3) + m4	16 × Icyc + 1 × Pcyc + 2 × Bcyc + 2 × (m1 + m2 + m3) + m4	16 × Icyc + 3 × Pcyc + 2 × (m1 + m2 + m3) + m4	

Notes: * In the case that m1 = m2 = m3 = m4 = 1 × Icyc.
m1 to m4 are the number of cycles needed for the following memory accesses.
m1: SR save (longword write)
m2: PC save (longword write)
m3: Vector address read (longword read)
m4: Fetch first instruction of interrupt service routine

6.8 Data Transfer with Interrupt Request Signals

The following data transfers can be done using interrupt request signals:

- Activate DMAC only; CPU interrupts do not occur
- Activate DTC only; CPU interrupts depend on DTC settings

Interrupt sources that are assigned for DMAC activation sources are masked without being input to the INTC. The mask condition is as follows:

Mask condition = Interrupt source select (CH0) + interrupt source select (CH1) + interrupt source select (CH2) + interrupt source select (CH3)

The INTC masks a CPU interrupt when the corresponding DTCE bit is 1. The conditions for clearing DTCE and interrupt source flag are shown below.

DTCE clear condition = DTC transfer end • DTCECLR
Interrupt source flag clear condition = DTC transfer end • $\overline{\text{DTCECLR}}$ + DMAC transfer end
where DTCECLR = DISEL + counter 0

Figures 6.5 and 6.6 show control block diagrams.

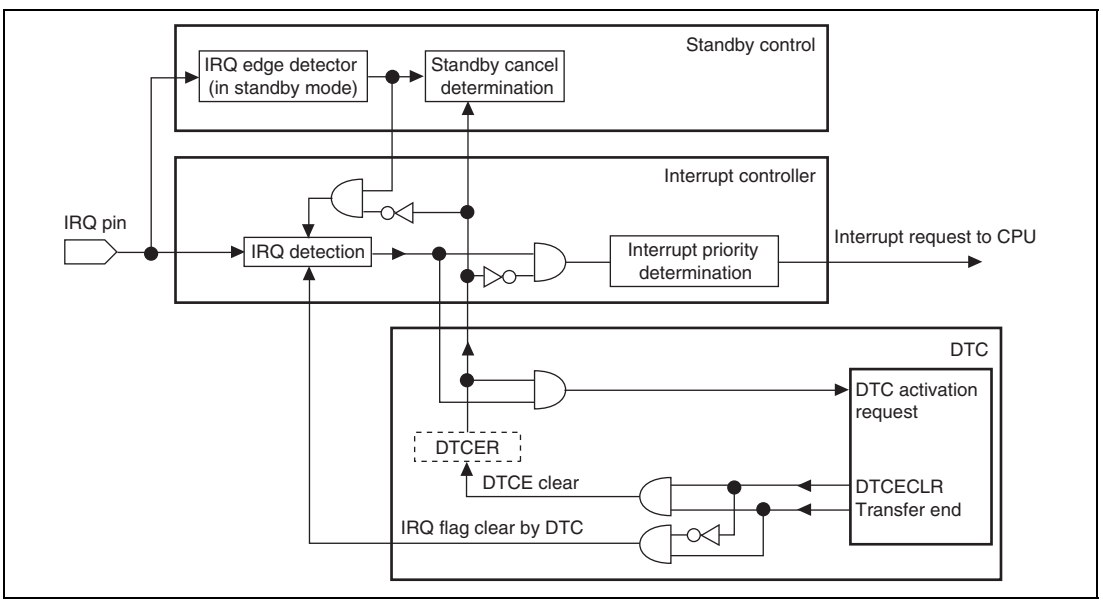


Figure 6.5 IRQ Interrupt Control Block Diagram

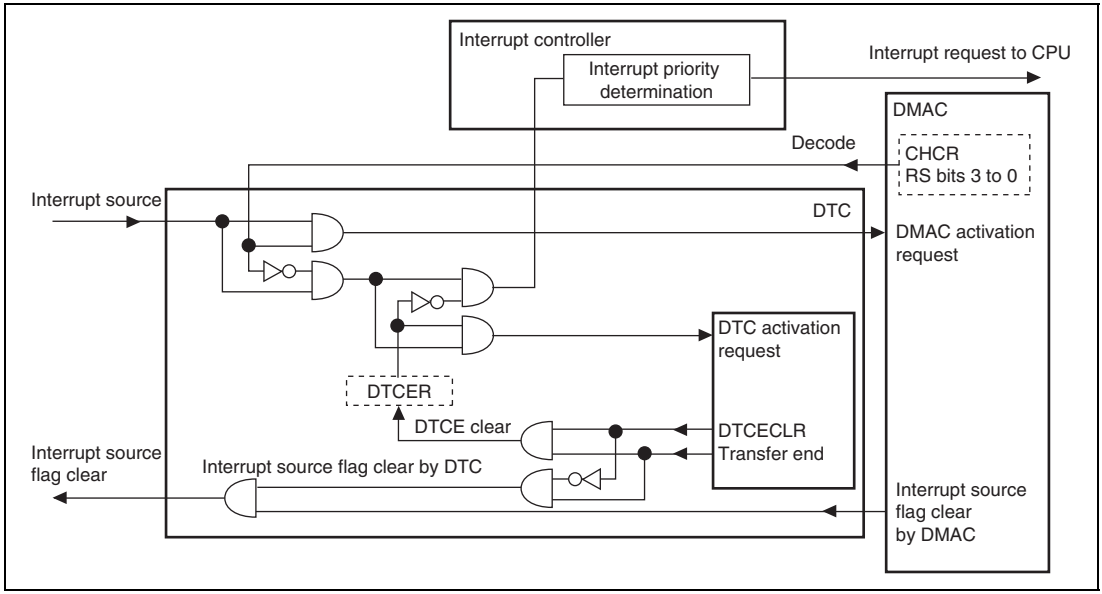


Figure 6.6 On-Chip Module Interrupt Control Block Diagram

6.8.1 Handling Interrupt Request Signals as Sources for DTC Activation and CPU Interrupts, but Not DMAC Activation

1. Do not select DMAC activation sources.
2. For DTC, set the corresponding DTCE bits and DISEL bits to 1.
3. When an interrupt occurs, an activation request is sent to the DTC.
4. When completing a data transfer, the DTC clears the DTCE bit to 0 and sends an interrupt request to the CPU. The activation source is not cleared.
5. The CPU clears the interrupt source in the interrupt handling routine then checks the transfer counter value. When the transfer counter value is not 0, the CPU sets the DTCE bit to 1 and allows the next data transfer. If the transfer counter value = 0, the CPU performs the necessary end processing in the interrupt processing routine.

6.8.2 Handling Interrupt Request Signals as Sources for DMAC Activation, but Not CPU Interrupts and DTC Activation

1. Select DMAC activation sources. Then, CPU interrupts and DTC activation sources are masked regardless of the settings in the interrupt priority registers and the DTC registers.
2. When an interrupt occurs, an activation request is sent to the DMAC.
3. The DMAC clears the interrupt source when starting transfer.

6.8.3 Handling Interrupt Request Signals as Sources for DTC Activation, but Not CPU Interrupts and DMAC Activation

1. Do not select DMAC activation sources.
2. For DTC, set the corresponding DTCE bits to 1 and clear the DISEL bits to 0.
3. When an interrupt occurs, an activation request is sent to the DTC.
4. When completing a data transfer, the DTC clears the activation source. No interrupt request is sent to the CPU because the DTCE bit is held at 1.
5. However, when the transfer counter value = 0, the DTCE bit is cleared to 0 and an interrupt request is sent to the CPU.
6. The CPU performs the necessary end processing in the interrupt handling routine.

6.8.4 Handling Interrupt Request Signals as Sources for CPU Interrupts, but Not DTC and DMAC Activation

1 Do not select DMAC activation sources.
2. For DTC, clear the corresponding DTCE bits to 0.
3. When an interrupt occurs, an interrupt request is sent to the CPU.
4. The CPU clears the interrupt source and performs the necessary processing in the interrupt handling routine.

6.9 Usage Note

The interrupt source flag should be cleared in the interrupt handler. To ensure that an interrupt source that should have been cleared is not inadvertently accepted again, read the interrupt source flag after it has been cleared, confirm that it has been cleared, and then execute an RTE instruction.

Section 7 User Break Controller (UBC)

The user break controller (UBC) provides functions that simplify program debugging. These functions make it easy to design an effective self-monitoring debugger, enabling the chip to debug programs without using an in-circuit emulator. Break conditions that can be set in the UBC are instruction fetch or data read/write access, data size, data contents, address value, and stop timing in the case of instruction fetch.

For the mask ROM version, only the L-bus instruction-fetch address break (2 channels) is

available.

7.1 Features

The UBC has the following features:

1. The following break comparison conditions can be set.

   Number of break channels: two channels (channels A and B)

   User break can be requested as either the independent or sequential condition on channels A and B (sequential break setting: channel A and then channel B match with break conditions, but not in the same bus cycle).

   — Address

   Comparison bits are maskable in 1-bit units.

   One of the two address buses (L-bus address (LAB) and I-bus address (IAB)) can be selected.

   — Data

   32-bit maskable.

   One of the two data buses (L-bus data (LDB) and I-bus data (IDB)) can be selected.

   — Bus cycle

   Instruction fetch or data access

   — Read/write

   — Operand size

   Byte, word, and longword
2. A user-designed user-break interrupt exception processing routine can be run.
3. In an instruction fetch cycle, it can be selected that a user break is set before or after an instruction is executed.
4. Maximum repeat times for the break condition (only for channel B): $2^{12} - 1$ times.

5. Four pairs of branch source/destination buffers (eight pairs for F-ZTAT version supporting full functions of E10A).

Figure 7.1 shows a block diagram of the UBC.

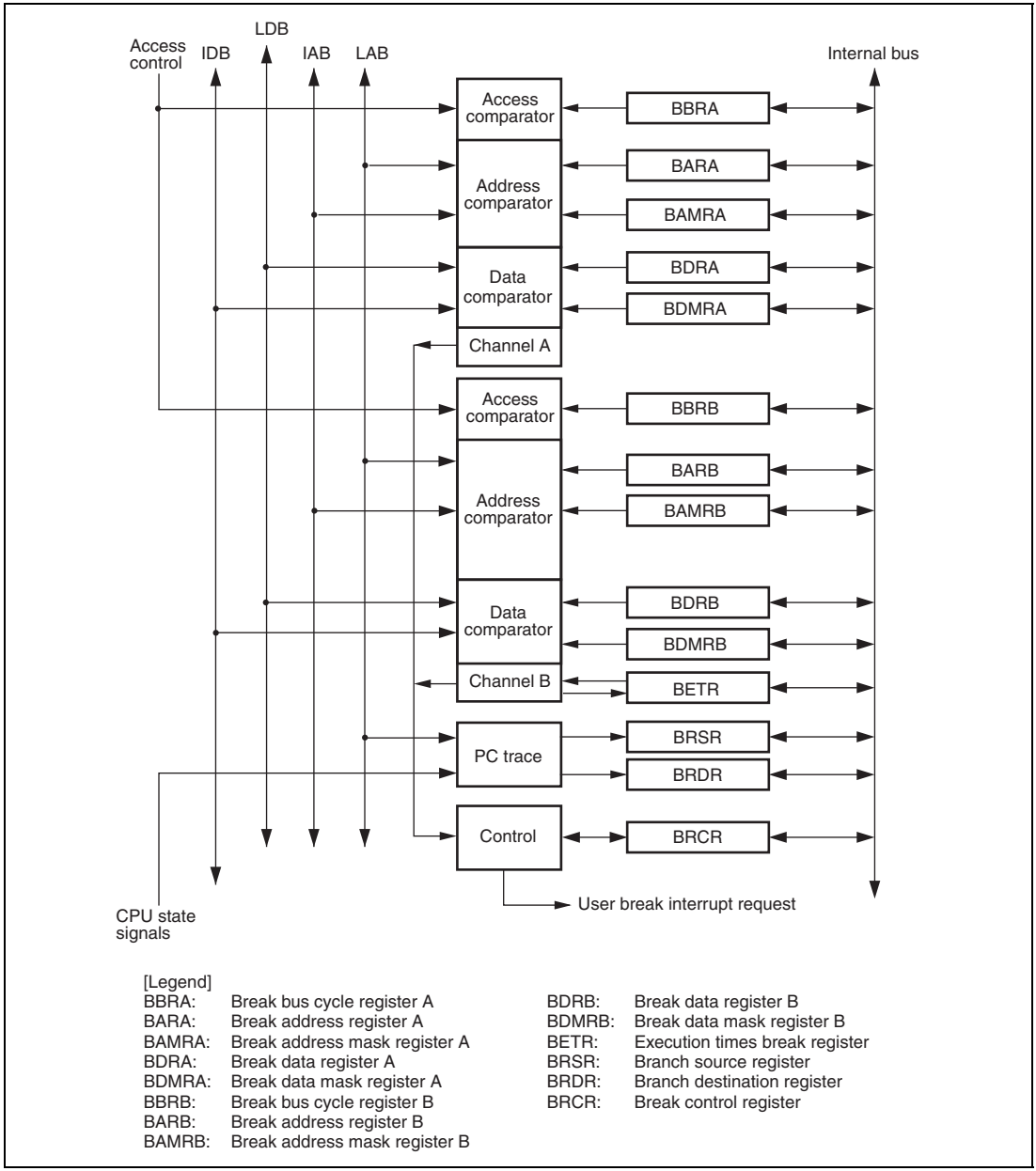


Figure 7.1 Block Diagram of UBC

7.2 Input/Output Pins

Table 7.1 shows the UBC pin configuration.

Table 7.1 Pin Configuration

Pin Name	Symbol	I/O	Function
User break trigger output	$\overline{\text{UBCTRG}}$	Output	UBC condition match trigger output pin.

7.3 Register Descriptions

The user break controller has the following registers. For details on register addresses and register states during each processing, refer to section 27, List of Registers.

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Break address register A	BARA	R/W	H'00000000	H'FFFFF300	32
Break address mask register A	BAMRA	R/W	H'00000000	H'FFFFF304	32
Break bus cycle register A	BBRA	R/W	H'0000	H'FFFFF308	16
Break data register A	BDRA*	R/W	H'00000000	H'FFFFF310	32
Break data mask register A	BDMRA*	R/W	H'00000000	H'FFFFF314	32
Break address register B	BARB	R/W	H'00000000	H'FFFFF320	32
Break address mask register B	BAMRB	R/W	H'00000000	H'FFFFF324	32
Break bus cycle register B	BBRB	R/W	H'0000	H'FFFFF328	16
Break data register B	BDRB*	R/W	H'00000000	H'FFFFF330	32
Break data mask register B	BDMRB*	R/W	H'00000000	H'FFFFF334	32
Break control register	BRCR	R/W	H'00000000	H'FFFFF3C0	32
Branch source register	BRSR*	R	H'0xxxxxxx	H'FFFFF3D0	32
Branch destination register	BRDR*	R	H'0xxxxxxx	H'FFFFF3D4	32
Execution times break register	BETR*	R/W	H'0000	H'FFFFF3DC	16

Note: * Only in F-ZTAT version

7.3.1 Break Address Register A (BARA)

BARA is a 32-bit readable/writable register. BARA specifies the address used as a break condition in channel A.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAA31	BAA30	BAA29	BAA28	BAA27	BAA26	BAA25	BAA24	BAA23	BAA22	BAA21	BAA20	BAA19	BAA18	BAA17	BAA16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAA15	BAA14	BAA13	BAA12	BAA11	BAA10	BAA9	BAA8	BAA7	BAA6	BAA5	BAA4	BAA3	BAA2	BAA1	BAA0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BAA31 to BAA 0	All 0	R/W	Break Address A Store the address on the LAB or IAB specifying break conditions of channel A.

7.3.2 Break Address Mask Register A (BAMRA)

BAMRA is a 32-bit readable/writable register. BAMRA specifies bits masked in the break address specified by BARA.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAMA31	BAMA30	BAMA29	BAMA28	BAMA27	BAMA26	BAMA25	BAMA24	BAMA23	BAMA22	BAMA21	BAMA20	BAMA19	BAMA18	BAMA17	BAMA16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAMA15	BAMA14	BAMA13	BAMA12	BAMA11	BAMA10	BAMA9	BAMA8	BAMA7	BAMA6	BAMA5	BAMA4	BAMA3	BAMA2	BAMA1	BAMA0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BAMA31 to BAMA 0	All 0	R/W	Break Address Mask A Specify bits masked in the channel A break address bits specified by BARA (BAA31 to BAA0). 0: Break address bit BAAn of channel A is included in the break condition 1: Break address bit BAAn of channel A is masked and is not included in the break condition Note: n = 31 to 0

7.3.3 Break Bus Cycle Register A (BBRA)

BBRA is a 16-bit readable/writable register, which specifies (1) bus master for I bus cycle, (2) L bus cycle or I bus cycle, (3) instruction fetch or data access, (4) read or write, and (5) operand size in the break conditions of channel A.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	CPA2*	CPA1*	CPA0*	CDA1*	CDA0	IDA1*	IDA0	RWA1*	RWA0	SZA1*	SZA0*
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * These bits are reserved in the mask ROM and ROM-less versions. These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	CPA2*	0	R/W	Bus Master Select A for I Bus
9	CPA1*	0	R/W	Select the bus master when the I bus is selected as the bus cycle of the channel A break condition. However, when the L bus is selected as the bus cycle, the setting of the CPA2 to CPA0 bits is disabled.
8	CPA0*	0	R/W	000: Condition comparison is not performed xx1: The CPU cycle is included in the break condition x1x: The DMAC cycle is included in the break condition 1xx: The DTC cycle is included in the break condition

Bit	Bit Name	Initial Value	R/W	Description
7	CDA1*	0	R/W	L Bus Cycle/I Bus Cycle Select A
6	CDA0	0	R/W	Select the L bus cycle or I bus cycle as the bus cycle of the channel A break condition. 00: Condition comparison is not performed 01: The break condition is the L bus cycle 10: The break condition is the I bus cycle 11: The break condition is the L bus cycle
5	IDA1*	0	R/W	Instruction Fetch/Data Access Select A
4	IDA0	0	R/W	Select the instruction fetch cycle or data access cycle as the bus cycle of the channel A break condition. 00: Condition comparison is not performed 01: The break condition is the instruction fetch cycle 10: The break condition is the data access cycle 11: The break condition is the instruction fetch cycle or data access cycle
3	RWA1*	0	R/W	Read/Write Select A
2	RWA0	0	R/W	Select the read cycle or write cycle as the bus cycle of the channel A break condition. 00: Condition comparison is not performed 01: The break condition is the read cycle 10: The break condition is the write cycle 11: The break condition is the read cycle or write cycle
1	SZA1*	0	R/W	Operand Size Select A
0	SZA0*	0	R/W	Select the operand size of the bus cycle for the channel A break condition. 00: The break condition does not include operand size 01: The break condition is byte access 10: The break condition is word access 11: The break condition is longword access Note: When specifying the operand size, specify the size which matches the address boundary.

[Legend]

x: Don't care.

Note: * These bits are reserved in the mask ROM and ROM-less versions. These bits are always read as 0. The write value should always be 0.

7.3.4 Break Data Register A (BDRA) (Only in F-ZTAT Version)

BDRA is a 32-bit readable/writable register. The control bits CDA1 and CDA0 in BBRA select one of two data buses for break condition A.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BDA31	BDA30	BDA29	BDA28	BDA27	BDA26	BDA25	BDA24	BDA23	BDA22	BDA21	BDA20	BDA19	BDA18	BDA17	BDA16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BDA15	BDA14	BDA13	BDA12	BDA11	BDA10	BDA9	BDA8	BDA7	BDA6	BDA5	BDA4	BDA3	BDA2	BDA1	BDA0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BDA31 to BDA0	All 0	R/W	Break Data Bit A Stores data which specifies a break condition in channel A. If the I bus is selected in BBRA, the break data on IDB is set in BDA31 to BDA0. If the L bus is selected in BBRA, the break data on LDB is set in BDA31 to BDA0.

Notes: 1. Specify an operand size when including the value of the data bus in the break condition.
2. When the byte size is selected as a break condition, the same byte data must be set in bits 15 to 8 and 7 to 0 in BDRA as the break data.

7.3.5 Break Data Mask Register A (BDMRA) (Only in F-ZTAT Version)

BDMRA is a 32-bit readable/writable register. BDMRA specifies bits masked in the break data specified by BDRA.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BDMA31	BDMA30	BDMA29	BDMA28	BDMA27	BDMA26	BDMA25	BDMA24	BDMA23	BDMA22	BDMA21	BDMA20	BDMA19	BDMA18	BDMA17	BDMA16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BDMA15	BDMA14	BDMA13	BDMA12	BDMA11	BDMA10	BDMA9	BDMA8	BDMA7	BDMA6	BDMA5	BDMA4	BDMA3	BDMA2	BDMA1	BDMA0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BDMA31 to BDMA 0	All 0	R/W	Break Data Mask A Specifies bits masked in the break data of channel A specified by BDRA (BDA31 to BDA0). 0: Break data BDAn of channel A is included in the break condition 1: Break data BDAn of channel A is masked and is not included in the break condition Note: n = 31 to 0

Notes: 1. Specify an operand size when including the value of the data bus in the break condition.
2. When the byte size is selected as a break condition, the same byte data must be set in bits 15 to 8 and 7 to 0 in BDMRA as the break mask data in BDRA.

7.3.6 Break Address Register B (BARB)

BARB is a 32-bit readable/writable register. BARB specifies the address used as a break condition in channel B. Control bits CDB1 and CDB0 in BBRB select one of the two address buses for break condition B.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAB31	BAB30	BAB29	BAB28	BAB27	BAB26	BAB25	BAB24	BAB23	BAB22	BAB21	BAB20	BAB19	BAB18	BAB17	BAB16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAB15	BAB14	BAB13	BAB12	BAB11	BAB10	BAB9	BAB8	BAB7	BAB6	BAB5	BAB4	BAB3	BAB2	BAB1	BAB0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BAB31 to BAB 0	All 0	R/W	Break Address B Stores an address which specifies a break condition in channel B. If the I bus or L bus is selected in BBRB, an IAB or LAB address is set in BAB31 to BAB0.

7.3.7 Break Address Mask Register B (BAMRB)

BAMRB is a 32-bit readable/writable register. BAMRB specifies bits masked in the break address specified by BARB.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAMB31	BAMB30	BAMB29	BAMB28	BAMB27	BAMB26	BAMB25	BAMB24	BAMB23	BAMB22	BAMB21	BAMB20	BAMB19	BAMB18	BAMB17	BAMB16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAMB15	BAMB14	BAMB13	BAMB12	BAMB11	BAMB10	BAMB9	BAMB8	BAMB7	BAMB6	BAMB5	BAMB4	BAMB3	BAMB2	BAMB1	BAMB0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BAMB31 to BAMB 0	All 0	R/W	Break Address Mask B Specifies bits masked in the break address of channel B specified by BARB (BAB31 to BAB0). 0: Break address BABn of channel B is included in the break condition 1: Break address BABn of channel B is masked and is not included in the break condition Note: n = 31 to 0

7.3.8 Break Data Register B (BDRB) (Only in F-ZTAT Version)

BDRB is a 32-bit readable/writable register. The control bits CDB1 and CDB0 in BBRB select one of the two data buses for break condition B.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BDB31	BDB30	BDB29	BDB28	BDB27	BDB26	BDB25	BDB24	BDB23	BDB22	BDB21	BDB20	BDB19	BDB18	BDB17	BDB16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BDB15	BDB14	BDB13	BDB12	BDB11	BDB10	BDB9	BDB8	BDB7	BDB6	BDB5	BDB4	BDB3	BDB2	BDB1	BDB0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BDB31 to BDB0	All 0	R/W	Break Data Bit B Stores data which specifies a break condition in channel B. If the I bus is selected in BBRB, the break data on IDB is set in BDB31 to BDB0. If the L bus is selected in BBRB, the break data on LDB is set in BDB31 to BDB0.

Notes: 1. Specify an operand size when including the value of the data bus in the break condition.
2. When the byte size is selected as a break condition, the same byte data must be set in bits 15 to 8 and 7 to 0 in BDRB as the break data.

7.3.9 Break Data Mask Register B (BDMRB) (Only in F-ZTAT Version)

BDMRB is a 32-bit readable/writable register. BDMRB specifies bits masked in the break data specified by BDRB.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BDMB31	BDMB30	BDMB29	BDMB28	BDMB27	BDMB26	BDMB25	BDMB24	BDMB23	BDMB22	BDMB21	BDMB20	BDMB19	BDMB18	BDMB17	BDMB16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BDMB15	BDMB14	BDMB13	BDMB12	BDMB11	BDMB10	BDMB9	BDMB8	BDMB7	BDMB6	BDMB5	BDMB4	BDMB3	BDMB2	BDMB1	BDMB0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BDMB31 to BDMB 0	All 0	R/W	Break Data Mask B Specifies bits masked in the break data of channel B specified by BDRB (BDB31 to BDB0). 0: Break data BDBn of channel B is included in the break condition 1: Break data BDBn of channel B is masked and is not included in the break condition Note: n = 31 to 0

Notes: 1. Specify an operand size when including the value of the data bus in the break condition.
2. When the byte size is selected as a break condition, the same byte data must be set in bits 15 to 8 and 7 to 0 in BDMRB as the break mask data in BDRB.

7.3.10 Break Bus Cycle Register B (BBRB)

BBRB is a 16-bit readable/writable register, which specifies (1) bus master for I bus cycle, (2) L bus cycle or I bus cycle, (3) instruction fetch or data access, (4) read or write, and (5) operand size in the break conditions of channel B.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	CPB2*	CPB1*	CPB0*	CDB1*	CDB0	IDB1*	IDB0	RWB1*	RWB0	SZB1*	SZB0*
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * These bits are reserved in the mask ROM and ROM-less versions. These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	CPB2*	0	R/W	Bus Master Select B for I Bus
9	CPB1*	0	R/W	Select the bus master when the I bus is selected as the bus cycle of the channel B break condition. However, when the L bus is selected as the bus cycle, the setting of the CPB2 to CPB0 bits is disabled. 000: Condition comparison is not performed xx1: The CPU cycle is included in the break condition x1x: The DMAC cycle is included in the break condition 1xx: The DTC cycle is included in the break condition
8	CPB0*	0	R/W	
7	CDB1*	0	R/W	L Bus Cycle/I Bus Cycle Select B
6	CDB0	0	R/W	Select the L bus cycle or I bus cycle as the bus cycle of the channel B break condition. 00: Condition comparison is not performed 01: The break condition is the L bus cycle 10: The break condition is the I bus cycle 11: The break condition is the L bus cycle

Bit	Bit Name	Initial Value	R/W	Description
5	IDB1*	0	R/W	Instruction Fetch/Data Access Select B
4	IDB0	0	R/W	Select the instruction fetch cycle or data access cycle as the bus cycle of the channel B break condition. 00: Condition comparison is not performed 01: The break condition is the instruction fetch cycle 10: The break condition is the data access cycle 11: The break condition is the instruction fetch cycle or data access cycle
3	RWB1*	0	R/W	Read/Write Select B
2	RWB0	0	R/W	Select the read cycle or write cycle as the bus cycle of the channel B break condition. 00: Condition comparison is not performed 01: The break condition is the read cycle 10: The break condition is the write cycle 11: The break condition is the read cycle or write cycle
1	SZB1*	0	R/W	Operand Size Select B
0	SZB0*	0	R/W	Select the operand size of the bus cycle for the channel B break condition. 00: The break condition does not include operand size 01: The break condition is byte access 10: The break condition is word access 11: The break condition is longword access Note: When specifying the operand size, specify the size which matches the address boundary.

[Legend]

x: Don't care.

Note: * These bits are reserved in the mask ROM and ROM-less versions. These bits are always read as 0. The write value should always be 0.

7.3.11 Break Control Register (BRCR)

BRCR sets the following conditions:

1. Channels A and B are used in two independent channel conditions or under the sequential condition.
2. A user break is set before or after instruction execution.
3. Specify whether to include the number of execution times on channel B in comparison conditions.
4. Determine whether to include data bus on channels A and B in comparison conditions.
5. Enable PC trace.
6. Select the $\overline{\text{UBCTRG}}$ output pulse width.
7. Specify whether to request the user break interrupt when channels A and B match with comparison conditions.

BRCR is a 32-bit readable/writable register that has break conditions match flags and bits for setting a variety of break conditions.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	UTRGW[1:0]		UBIDB	-	UBIDA	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCM FCA	SCM FCB	SCM FDA	SCM FDB	PCTE	PCBA	-	-	DBEA	PCBB	DBEB	-	SEQ	-	-	ETBE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R	R/W	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	UTRGW[1:0]	00	R/W	$\overline{\text{UBCTRG}}$ Output Pulse Width Select Select the $\overline{\text{UBCTRG}}$ output pulse width when the break condition matches. 00: Setting prohibited. 01: $\overline{\text{UBCTRG}}$ output pulse width is 3 to 4 t_{Bcyc} 10: $\overline{\text{UBCTRG}}$ output pulse width is 7 to 8 t_{Bcyc} 11: $\overline{\text{UBCTRG}}$ output pulse width is 15 to 16 t_{Bcyc} Note: t_{Bcyc} indicates the period of one cycle of the external bus clock (Bϕ = CK).
19	UBIDB	0	R/W	User Break Disable B Enables or disables the user break interrupt request when the channel B break conditions are satisfied. 0: User break interrupt request is enabled when break conditions are satisfied 1: User break interrupt request is disabled when break conditions are satisfied
18	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
17	UBIDA	0	R/W	User Break Disable A Enables or disables the user break interrupt request when the channel A break conditions are satisfied. 0: User break interrupt request is enabled when break conditions are satisfied 1: User break interrupt request is disabled when break conditions are satisfied
16	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15	SCMFCA	0	R/W	L Bus Cycle Condition Match Flag A When the L bus cycle condition in the break conditions set for channel A is satisfied, this flag is set to 1. In order to clear this flag, write 0 into this bit. 0: The L bus cycle condition for channel A does not match 1: The L bus cycle condition for channel A matches
14	SCMFCB	0	R/W	L Bus Cycle Condition Match Flag B When the L bus cycle condition in the break conditions set for channel B is satisfied, this flag is set to 1. In order to clear this flag, write 0 into this bit. 0: The L bus cycle condition for channel B does not match 1: The L bus cycle condition for channel B matches
13	SCMFDA	0	R/W	I Bus Cycle Condition Match Flag A When the I bus cycle condition in the break conditions set for channel A is satisfied, this flag is set to 1. In order to clear this flag, write 0 into this bit. 0: The I bus cycle condition for channel A does not match 1: The I bus cycle condition for channel A matches
12	SCMFDB	0	R/W	I Bus Cycle Condition Match Flag B When the I bus cycle condition in the break conditions set for channel B is satisfied, this flag is set to 1. In order to clear this flag, write 0 into this bit. 0: The I bus cycle condition for channel B does not match 1: The I bus cycle condition for channel B matches
11	PCTE	0	R/W	PC Trace Enable 0: Disables PC trace 1: Enables PC trace

Bit	Bit Name	Initial Value	R/W	Description
10	PCBA	0	R/W	PC Break Select A Selects the break timing of the instruction fetch cycle for channel A as before or after instruction execution. 0: PC break of channel A is set before instruction execution 1: PC break of channel A is set after instruction execution
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	DBEA	0	R/W	Data Break Enable A Selects whether or not the data bus condition is included in the break condition of channel A. 0: No data bus condition is included in the condition of channel A 1: The data bus condition is included in the condition of channel A
6	PCBB	0	R/W	PC Break Select B Selects the break timing of the instruction fetch cycle for channel B as before or after instruction execution. 0: PC break of channel B is set before instruction execution 1: PC break of channel B is set after instruction execution
5	DBEB	0	R/W	Data Break Enable B Selects whether or not the data bus condition is included in the break condition of channel B. 0: No data bus condition is included in the condition of channel B 1: The data bus condition is included in the condition of channel B
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3	SEQ	0	R/W	Sequence Condition Select Selects two conditions of channels A and B as independent or sequential conditions. 0: Channels A and B are compared under independent conditions 1: Channels A and B are compared under sequential conditions (channel A, then channel B)
2, 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ETBE	0	R/W	Number of Execution Times Break Enable Enables the execution-times break condition only on channel B. If this bit is 1 (break enable), a user break interrupt is requested when the number of break conditions matches with the number of execution times that is specified by BETR. 0: The execution-times break condition is disabled on channel B 1: The execution-times break condition is enabled on channel B

7.3.12 Execution Times Break Register (BETR) (Only in F-ZTAT Version)

BETR is a 16-bit readable/writable register. When the execution-times break condition of channel B is enabled, this register specifies the number of execution times to make the break. The maximum number is $2^{12} - 1$ times. When a break condition is satisfied, it decreases BETR. A user break interrupt is requested when the break condition is satisfied after BETR becomes H'0001.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	BET[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	BET[11:0]	All 0	R/W	Number of Execution Times

7.3.13 Branch Source Register (BRSR) (Only in F-ZTAT Version)

BRSR is a 32-bit read-only register. BRSR stores bits 27 to 0 in the address of the branch source instruction. BRSR has the flag bit that is set to 1 when a branch occurs. This flag bit is cleared to 0 when BRSR is read, the setting to enable PC trace is made, or BRSR is initialized by a power-on reset or a manual reset. Other bits are not initialized by a power-on reset. The four BRSR registers (eight pairs for the F-ZTAT version supporting full functions of E10A) have a queue structure and a stored register is shifted at every branch.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SVF	-	-	-	BSA27	BSA26	BSA25	BSA24	BSA23	BSA22	BSA21	BSA20	BSA19	BSA18	BSA17	BSA16
Initial value:	0	0	0	0	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSA15	BSA14	BSA13	BSA12	BSA11	BSA10	BSA9	BSA8	BSA7	BSA6	BSA5	BSA4	BSA3	BSA2	BSA1	BSA0
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	SVF	0	R	BRSR Valid Flag Indicates whether the branch source address is stored. This flag bit is set to 1 when a branch occurs. This flag is cleared to 0 when BRSR is read, the setting to enable PC trace is made, or BRSR is initialized by a power-on reset. 0: The value of BRSR register is invalid 1: The value of BRSR register is valid
30 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 0	BSA27 to BSA0	Undefined	R	Branch Source Address Store bits 27 to 0 of the branch source address.

7.3.14 Branch Destination Register (BRDR) (Only in F-ZTAT Version)

BRDR is a 32-bit read-only register. BRDR stores bits 27 to 0 in the address of the branch destination instruction. BRDR has the flag bit that is set to 1 when a branch occurs. This flag bit is cleared to 0 when BRDR is read, the setting to enable PC trace is made, or BRDR is initialized by a power-on reset or a manual reset. Other bits are not initialized by a power-on reset. The four BRSR registers (eight pairs for the F-ZTAT version supporting full functions of E10A) have a queue structure and a stored register is shifted at every branch.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DVF	-	-	-	BDA27	BDA26	BDA25	BDA24	BDA23	BDA22	BDA21	BDA20	BDA19	BDA18	BDA17	BDA16
Initial value:	0	0	0	0	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BDA15	BDA14	BDA13	BDA12	BDA11	BDA10	BDA9	BDA8	BDA7	BDA6	BDA5	BDA4	BDA3	BDA2	BDA1	BDA0
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	DVF	0	R	BRDR Valid Flag Indicates whether a branch destination address is stored. This flag bit is set to 1 when a branch occurs. This flag is cleared to 0 when BRDR is read, the setting to enable PC trace is made, or BRDR is initialized by a power-on reset. 0: The value of BRDR register is invalid 1: The value of BRDR register is valid
30 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 0	BDA27 to BDA0	Undefined	R	Branch Destination Address Store bits 27 to 0 of the branch destination address.

7.4 Operation

7.4.1 Flow of the User Break Operation

The flow from setting of break conditions to user break exception processing is described below:

1. The break addresses are set in the break address registers (BARA or BARB). The masked addresses are set in the break address mask registers (BAMRA or BAMRB). The break data is set in the break data register (BDRA or BDRB). The masked data is set in the break data mask register (BDMRA or BDMRB). The bus break conditions are set in the break bus cycle registers (BBRA or BBRB). Three groups of BBRA or BBRB (L bus cycle/I bus cycle select, instruction fetch/data access select, and read/write select) are each set. No user break will be generated if even one of these groups is set with B'00. The respective conditions are set in the bits of the break control register (BRCR). Make sure to set all registers related to breaks before setting BBRA or BBRB.
2. When the break conditions are satisfied, the UBC sends a user break interrupt request to the CPU and sets the L bus condition match flag (SCMFCA or SCMFCB) and the I bus condition match flag (SCMFDA or SCMFDB) for the appropriate channel.
3. The appropriate condition match flags (SCMFCA, SCMFDA, SCMFCB, and SCMFDB) can be used to check if the set conditions match or not. The matching of the conditions sets flags, but they are not reset. Before using them again, 0 must first be written to them and then reset flags.
4. There is a chance that matches of the break conditions set in channels A and B occur almost at the same time. In this case, there will be only one user break request to the CPU, but these two conditions match flags could be both set.
5. When selecting the I bus as the break condition, note the following:
   - The CPU, DMAC, and DTC are connected to the I bus. The UBC monitors bus cycles generated by all bus masters that are selected by the CPA2 to CPA0 bits in BBRA or the CPB2 to CPB0 bits in BBRB, and compares the conditions for a match.
   - I bus cycles (including read fill cycles) resulting from instruction fetches on the L bus by the CPU are defined as instruction fetch cycles on the I bus, while other bus cycles are defined as data access cycles.
   - The DMAC and DTC only issue data access cycles for I bus cycles.
   - If a break condition is specified for the I bus, even when the condition matches in an I bus cycle resulting from an instruction executed by the CPU, at which instruction the user break is to be accepted cannot be clearly defined.

7.4.2 User Break on Instruction Fetch Cycle

1. When L bus/instruction fetch/read/word, longword, or not including the operand size is set in the break bus cycle register (BBRA or BBRB), the break condition becomes the L bus instruction fetch cycle. Whether it breaks before or after the execution of the instruction can then be selected with the PCBA or PCBB bit in the break control register (BRCR) for the appropriate channel. If an instruction fetch cycle is set as a break condition, clear LSB in the break address register (BARA or BARB) to 0. A user break cannot be generated as long as this bit is set to 1.
2. If the break condition matches when a user break on instruction fetch is specified so that the a break is generated before the execution of the instruction, the user break is generated at the point when it has become deterministic that the instruction will be executed after it is fetched. This means this feature cannot be used on instructions fetched by overrun (instructions fetched at a branch or during an interrupt transition, but not to be executed). When this kind of break condition is set for the delay slot of a delayed branch instruction, the user break is generated prior to execution of the delayed branch instruction.

   Note: If a branch does not occur at a delay condition branch instruction, the subsequent instruction is not recognized as a delay slot.

3. When the break condition is specified so that a user break is generated after execution of the instruction, the instruction that has met the break condition is executed and then the user break is generated before the next instruction is executed. As with pre-execution user breaks, this cannot be used with overrun fetch instructions. When this kind of break condition is set for a delayed branch instruction and its delay slot, a user break is not generated until the first instruction at the branch destination.
4. When an instruction fetch cycle is set, the break data register (BDRA or BDRB) is ignored. Therefore, break data cannot be set for the user break of the instruction fetch cycle.
5. If the I bus is set for a user break of an instruction fetch cycle, the condition is determined for the instruction fetch cycles on the I bus. For details, see 5 in section 7.4.1, Flow of the User Break Operation.

7.4.3 User Break on Data Access Cycle

1. If the L bus is specified as a break condition for data access break, condition comparison is performed for the address (and data) accessed by the executed instructions, and a user break occurs if the condition is satisfied. If the I bus is specified as a break condition, condition comparison is performed for the addresses (and data) of the data access cycles that are issued on the I bus by all bus masters including the CPU, and a user break occurs if the condition is satisfied. For details on the CPU bus cycles issued on the I bus, see 5 in section 7.4.1, Flow of the User Break Operation.
2. The relationship between the data access cycle address and the comparison condition for each operand size is listed in table 7.3.

Table 7.3 Data Access Cycle Addresses and Operand Size Comparison Conditions

Access Size	Address Compared
Longword	Compares break address register bits 31 to 2 to address bus bits 31 to 2
Word	Compares break address register bits 31 to 1 to address bus bits 31 to 1
Byte	Compares break address register bits 31 to 0 to address bus bits 31 to 0

This means that when address H'00001003 is set in the break address register (BARA or BARB), for example, the bus cycle in which the break condition is satisfied is as follows (where other conditions are met).

Longword access at H'00001000

Word access at H'00001002

Byte access at H'00001003

3. When the data value is included in the break conditions:

   When the data value is included in the break conditions, either longword, word, or byte is specified as the operand size of the break bus cycle register (BBRA or BBRB). When data values are included in break conditions, a user break is generated when the address conditions and data conditions both match. To specify byte data for this case, set the same data in two bytes at bits 15 to 8 and bits 7 to 0 of the break data register (BDRA or BDRB) and break data mask register (BDMRA or BDMRB). When word or byte is set, bits 31 to 16 of BDRA or BDRB and BDMRA or BDMRB are ignored.
4. If the L bus is selected, a user break occurs on ending execution of the instruction that matches the break condition, and immediately before the next instruction is executed. However, when data is also specified as the break condition, the break may occur on ending execution of the instruction following the instruction that matches the break condition. If the I bus is selected, the instruction at which the user break will occur cannot be determined. When this kind of user

break occurs at a delayed branch instruction or its delay slot, the user break may not actually take place until the first instruction at the branch destination.

7.4.4 Sequential Break

1. By setting the SEQ bit in BRCR to 1, the sequential break is issued when a channel B break condition matches after a channel A break condition matches. A user break is not generated even if a channel B break condition matches before a channel A break condition matches. When channels A and B conditions match at the same time, the sequential break is not issued. To clear the channel A condition match when a channel A condition match has occurred but a channel B condition match has not yet occurred in a sequential break specification, clear the SEQ bit in BRCR to 0 and clear the condition match flag to 0 in channel A.
2. In sequential break specification, the L or I bus can be selected and the execution times break condition can be also specified. For example, when the execution times break condition is specified, the break condition is satisfied when a channel B condition matches with BETR = H'0001 after a channel A condition has matched.

7.4.5 Value of Saved Program Counter

When a user break occurs, the address of the instruction from where execution is to be resumed is saved in the stack, and the exception handling state is entered. If the L bus is specified as a break condition, the instruction at which the user break should occur can be clearly determined (except for when data is included in the break condition). If the I bus is specified as a break condition, the instruction at which the user break should occur cannot be clearly determined.

1. When instruction fetch (before instruction execution) is specified as a break condition:

   The address of the instruction that matched the break condition is saved in the stack. The instruction that matched the condition is not executed, and the user break occurs before it. However when a delay slot instruction matches the condition, the address of the delayed branch instruction is saved in the stack.
2. When instruction fetch (after instruction execution) is specified as a break condition:

   The address of the instruction following the instruction that matched the break condition is saved in the stack. The instruction that matches the condition is executed, and the user break occurs before the next instruction is executed. However when a delayed branch instruction or delay slot matches the condition, these instructions are executed, and the branch destination address is saved in the stack.

3. When data access (address only) is specified as a break condition:

   The address of the instruction immediately after the instruction that matched the break condition is saved in the stack. The instruction that matches the condition is executed, and the user break occurs before the next instruction is executed. However when a delay slot instruction matches the condition, the branch destination address is saved in the stack.

4. When data access (address + data) is specified as a break condition:

   When a data value is added to the break conditions, the address of an instruction that is within two instructions of the instruction that matched the break condition is saved in the stack. At which instruction the user break occurs cannot be determined accurately.

   When a delay slot instruction matches the condition, the branch destination address is saved in the stack. If the instruction following the instruction that matches the break condition is a branch instruction, the user break may occur after the branch instruction or delay slot has finished. In this case, the branch destination address is saved in the stack.

7.4.6 PC Trace

1. Setting PCTE in BRCR to 1 enables PC traces. When branch (branch instruction, and interrupt exception) is generated, the branch source address and branch destination address are stored in BRSR and BRDR, respectively.
2. The values stored in BRSR and BRDR are as given below due to the kind of branch.
   - If a branch occurs due to a branch instruction, the address of the branch instruction is saved in BRSR and the address of the branch destination instruction is saved in BRDR.
   - If a branch occurs due to an interrupt or exception, the value saved in stack due to exception occurrence is saved in BRSR and the start address of the exception handling routine is saved in BRDR.
3. BRSR and BRDR have four pairs of queue structures (eight pairs for the F-ZTAT version supporting full functions of E10A). The top of queues is read first when the address stored in the PC trace register is read. BRSR and BRDR share the read pointer. Read BRSR and BRDR in order, the queue only shifts after BRDR is read. After switching the PCTE bit (in BRCR) off and on, the values in the queues are invalid.
4. Since four pairs (eight pairs for the F-ZTAT version supporting full functions of E10A) of queue are shared with the AUD, set the PCTE bit in BRCR to 1 after setting the MSTP25 bit in STBCR5 to 0 and the AUDSRST bit in STBCR6 to 1. Although the AUD is only available in the F-ZTAT version supporting full functions of the E10A, this setting should also be made in the normal F-ZTAT version.

7.4.7 Usage Examples

Break Condition Specified for L Bus Instruction Fetch Cycle:

(Example 1-1)

- Register specifications

  BARA = H'00000404, BAMRA = H'00000000, BBRA = H'0054, BDRA = H'00000000, BDMRA = H'00000000, BARB = H'00008010, BAMRB = H'00000006, BBRB = H'0054, BDRB = H'00000000, BDMRB = H'00000000, BRCR = H'00000400

  Specified conditions: Channel A/channel B independent mode

  <Channel A>

  Address: H'00000404, Address mask: H'00000000

  Data: H'00000000, Data mask: H'00000000

  Bus cycle: L bus/instruction fetch (after instruction execution)/read (operand size is not included in the condition)

  <Channel B>

  Address: H'00008010, Address mask: H'00000006

  Data: H'00000000, Data mask: H'00000000

  Bus cycle: L bus/instruction fetch (before instruction execution)/read (operand size is not included in the condition)

  A user break occurs after an instruction of address H'00000404 is executed or before instructions of addresses H'00008010 to H'00008016 are executed.

(Example 1-2)

- Register specifications

  BARA = H'00037226, BAMRA = H'00000000, BBRA = H'0056, BDRA = H'00000000, BDMRA = H'00000000, BARB = H'0003722E, BAMRB = H'00000000, BBRB = H'0056, BDRB = H'00000000, BDMRB = H'00000000, BRCR = H'00000008

  Specified conditions: Channel A/channel B sequential mode

  <Channel A>

  Address: H'00037226, Address mask: H'00000000

  Data: H'00000000, Data mask: H'00000000

  Bus cycle: L bus/instruction fetch (before instruction execution)/read/word

<Channel B>

Address: H'0003722E, Address mask: H'00000000

Data: H'00000000, Data mask: H'00000000

Bus cycle: L bus/instruction fetch (before instruction execution)/read/word

After an instruction with address H'00037226 is executed, a user break occurs before an instruction with address H'0003722E is executed.

(Example 1-3)

- Register specifications

  BARA = H'00027128, BAMRA = H'00000000, BBRA = H'005A, BDRA = H'00000000, BDMRA = H'00000000, BARB = H'00031415, BAMRB = H'00000000, BBRB = H'0054, BDRB = H'00000000, BDMRB = H'00000000, BRCR = H'00000000

  Specified conditions: Channel A/channel B independent mode

  <Channel A>

  Address: H'00027128, Address mask: H'00000000

  Data: H'00000000, Data mask: H'00000000

  Bus cycle: L bus/instruction fetch (before instruction execution)/write/word

  <Channel B>

  Address: H'00031415, Address mask: H'00000000

  Data: H'00000000, Data mask: H'00000000

  Bus cycle: L bus/instruction fetch (before instruction execution)/read (operand size is not included in the condition)

  On channel A, no user break occurs since instruction fetch is not a write cycle. On channel B, no user break occurs since instruction fetch is performed for an even address.

(Example 1-4)

- Register specifications

  BARA = H'00037226, BAMRA = H'00000000, BBRA = H'005A, BDRA = H'00000000, BDMRA = H'00000000, BARB = H'0003722E, BAMRB = H'00000000, BBRB = H'0056, BDRB = H'00000000, BDMRB = H'00000000, BRCR = H'00000008

  Specified conditions: Channel A/channel B sequential mode

  <Channel A>

  Address: H'00037226, Address mask: H'00000000

  Data: H'00000000, Data mask: H'00000000

  Bus cycle: L bus/instruction fetch (before instruction execution)/write/word

<Channel B>

Address: H'0003722E, Address mask: H'00000000

Data: H'00000000, Data mask: H'00000000

Bus cycle: L bus/instruction fetch (before instruction execution)/read/word

Since instruction fetch is not a write cycle on channel A, a sequential condition does not match. Therefore, no user break occurs.

(Example 1-5)

- Register specifications

  BARA = H'00000500, BAMRA = H'00000000, BBRA = H'0057, BDRA = H'00000000, BDMRA = H'00000000, BARB = H'00001000, BAMRB = H'00000000, BBRB = H'0057, BDRB = H'00000000, BDMRB = H'00000000, BRCR = H'00000001, BETR = H'0005

  Specified conditions: Channel A/channel B independent mode

  <Channel A>

  Address: H'00000500, Address mask: H'00000000

  Data: H'00000000, Data mask: H'00000000

  Bus cycle: L bus/instruction fetch (before instruction execution)/read/longword

  The number of execution-times break enable (5 times)

  <Channel B>

  Address: H'00001000, Address mask: H'00000000

  Data: H'00000000, Data mask: H'00000000

  Bus cycle: L bus/instruction fetch (before instruction execution)/read/longword

  On channel A, a user break occurs after the instruction of address H'00000500 is executed four times and before the fifth time.

  On channel B, a user break occurs before an instruction of address H'00001000 is executed.

(Example 1-6)

- Register specifications

  BARA = H'00008404, BAMRA = H'00000FFF, BBRA = H'0054, BDRA = H'00000000, BDMRA = H'00000000, BARB = H'00008010, BAMRB = H'00000006, BBRB = H'0054, BDRB = H'00000000, BDMRB = H'00000000, BRCR = H'00000400

  Specified conditions: Channel A/channel B independent mode

  <Channel A>

  Address: H'00008404, Address mask: H'00000FFF

  Data: H'00000000, Data mask: H'00000000

Bus cycle: L bus/instruction fetch (after instruction execution)/read (operand size is not included in the condition)

<Channel B>

Address: H'00008010, Address mask: H'00000006

Data: H'00000000, Data mask: H'00000000

Bus cycle: L bus/instruction fetch (before instruction execution)/read (operand size is not included in the condition)

A user break occurs after an instruction with addresses H'00008000 to H'00008FFE is executed or before an instruction with addresses H'00008010 to H'00008016 are executed.

Break Condition Specified for L Bus Data Access Cycle:

(Example 2-1)

- Register specifications

  BARA = H'00123456, BAMRA = H'00000000, BBRA = H'0064, BDRA = H'12345678, BDMRA = H'FFFFFFFF, BARB = H'000ABCDE, BAMRB = H'000000FF, BBRB = H'006A, BDRB = H'0000A512, BDMRB = H'00000000, BRCR = H'00000080

  Specified conditions: Channel A/channel B independent mode

  <Channel A>

  Address: H'00123456, Address mask: H'00000000

  Data: H'12345678, Data mask: H'FFFFFFFF

  Bus cycle: L bus/data access/read (operand size is not included in the condition)

  <Channel B>

  Address: H'000ABCDE, Address mask: H'000000FF

  Data: H'0000A512, Data mask: H'00000000

  Bus cycle: L bus/data access/write/word

  On channel A, a user break occurs with longword read from address H'00123454, word read from address H'00123456, or byte read from address H'00123456. On channel B, a user break occurs when word H'A512 is written in addresses H'000ABC00 to H'000ABCFE.

Break Condition Specified for I Bus Data Access Cycle:

(Example 3-1)

- Register specifications

  BARA = H'00314154, BAMRA = H'00000000, BBRA = H'0194, BDRA = H'12345678, BDMRA = H'FFFFFFFF, BARB = H'00055555, BAMRB = H'00000000, BBRB = H'01A9, BDRB = H'00007878, BDMRB = H'00000F0F, BRCR = H'00000080

  Specified conditions: Channel A/channel B independent mode

  <Channel A>

  Address: H'00314154, Address mask: H'00000000

  Data: H'12345678, Data mask: H'FFFFFFFF

  Bus cycle: I bus (CPU cycle)/instruction fetch/read (operand size is not included in the condition)

  <Channel B>

  Address: H'00055555, Address mask: H'00000000

  Data: H'00000078, Data mask: H'0000000F

  Bus cycle: I bus (CPU cycle)/data access/write/byte

  On channel A, a user break occurs when instruction fetch is performed for address H'00314156 in the external memory space.

  On channel B, a user break occurs when byte data H'7x is written in address H'00055555 in the external memory space by the CPU.

7.5 Usage Notes

1. The CPU can read from or write to the UBC registers via the I bus. Accordingly, during the period from executing an instruction to rewrite the UBC register till the new value is actually rewritten, the desired user break may not occur. In order to know the timing when the UBC register is changed, read from the last written register. Instructions after then are valid for the newly written register value.
2. UBC cannot monitor access to the L bus and I bus in the same channel.
3. Note on specification of sequential break:

   A condition match occurs when a B-channel match occurs in a bus cycle after an A-channel match occurs in another bus cycle in sequential break setting. Therefore, no user break occurs even if a bus cycle, in which an A-channel match and a channel B match occur simultaneously, is set.
4. When a user break and another exception occur at the same instruction, which has higher priority is determined according to the priority levels defined in table 5.1 in section 5, Exception Handling. If an exception with higher priority occurs, the user break is not generated.
   - Pre-execution break has the highest priority.
   - When a post-execution break or data access break occurs simultaneously with a re-execution-type exception (including pre-execution break) that has higher priority, the re-execution-type exception is accepted, and the condition match flag is not set (see the exception in the following note). The user break will occur and the condition match flag will be set only after the exception source of the re-execution-type exception has been cleared by the exception handling routine and re-execution of the same instruction has ended.
   - When a post-execution break or data access break occurs simultaneously with a completion-type exception (TRAPA) that has higher priority, though a user break does not occur, the condition match flag is set.
5. Note the following exception for the above note.

   If a post-execution break or data access break is satisfied by an instruction that generates a CPU address error by data access, the CPU address error is given priority to the user break interrupt. Note that the UBC condition match flag is set in this case.
6. Note the following when a user break occurs in a delay slot.

   If a pre-execution break is set at the delay slot instruction of the RTE instruction, the user break does not occur until the branch destination of the RTE instruction.

7. User breaks are disabled during UBC module standby mode. Do not read from or write to the UBC registers during UBC module standby mode; the values are not guaranteed.
8. Do not set a post-execution break at a SLEEP instruction or a branch instruction for which a SLEEP instruction is placed in the delay slot. In addition, do not set a data access break at a SLEEP instruction or one or two instructions before a SLEEP instruction.
9. When the DTC or DMAC is in operation, the UBC cannot correctly determine access to the external space by the CPU via the I bus. To determine access to the external space via the I bus in the above situation, select all bus masters. This makes it impossible to determine conditions of access with specified bus masters. However, when a bus master can be inferred from data values, the relevant data values can be included as a condition that indicates a particular bus master.

Section 8 Data Transfer Controller (DTC)

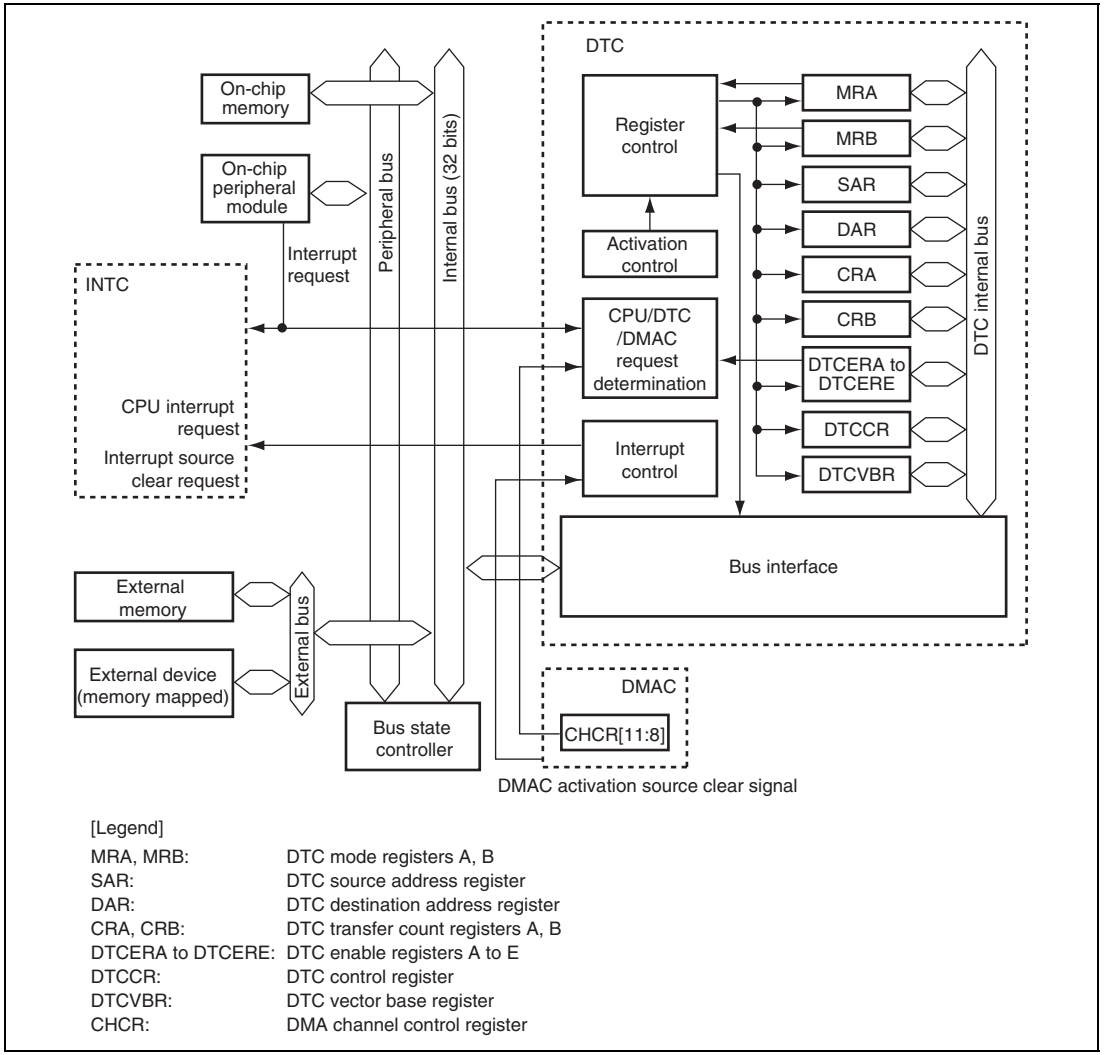
This LSI includes a data transfer controller (DTC). The DTC can be activated to transfer data by an interrupt request.

8.1 Features

- Transfer possible over any number of channels:
- Chain transfer
  Multiple rounds of data transfer is executed in response to a single activation source
  Chain transfer is only possible after data transfer has been done for the specified number of times (i.e. when the transfer counter is 0)
- Three transfer modes
  Normal/repeat/block transfer modes selectable
  Transfer source and destination addresses can be selected from increment/decrement/fixed
- The transfer source and destination addresses can be specified by 32 bits to select a 4-Gbyte address space directly
- Size of data for data transfer can be specified as byte, word, or longword
- A CPU interrupt can be requested for the interrupt that activated the DTC
  A CPU interrupt can be requested after one data transfer completion
  A CPU interrupt can be requested after the specified data transfer completion
- Read skip of the transfer information specifiable
- Writeback skip executed for the fixed transfer source and destination addresses
- Module stop mode specifiable
- Short address mode specifiable
- Bus release timing selectable from five types
- Priority of the DTC activation selectable from two types

Figure 8.1 shows a block diagram of the DTC. The DTC transfer information can be allocated to the data area*.

Note: * When the transfer information is stored in the on-chip RAM, the RAME bit in RAMCR must be set to 1.



[Legend]

MRA, MRB:	DTC mode registers A, B
SAR:	DTC source address register
DAR:	DTC destination address register
CRA, CRB:	DTC transfer count registers A, B
DTCERA to DTCERE:	DTC enable registers A to E
DTCCR:	DTC control register
DTCVBR:	DTC vector base register
CHCR:	DMA channel control register

Figure 8.1 Block Diagram of DTC

8.2 Register Descriptions

DTC has the following registers. For details on the addresses of these registers and the states of these registers in each processing state, see section 27, List of Registers.

These six registers MRA, MRB, SAR, DAR, CRA, and CRB cannot be directly accessed by the CPU. The contents of these registers are stored in the data area as transfer information. When a DTC activation request occurs, the DTC reads a start address of transfer information that is stored in the data area according to the vector address, reads the transfer information, and transfers data. After the data transfer, it writes a set of updated transfer information back to the data area.

On the other hand, DTCERA to DTCERE, DTCCR, and DTCVBR can be directly accessed by the CPU.

Table 8.1 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DTC enable register A	DTCERA	R/W	H'0000	H'FFFFCC80	8, 16
DTC enable register B	DTCERB	R/W	H'0000	H'FFFFCC82	8, 16
DTC enable register C	DTCERC	R/W	H'0000	H'FFFFCC84	8, 16
DTC enable register D	DTCERD	R/W	H'0000	H'FFFFCC86	8, 16
DTC enable register E	DTCERE	R/W	H'0000	H'FFFFCC88	8, 16
DTC control register	DTCCR	R/W	H'00	H'FFFFCC90	8
DTC vector base register	DTCVBR	R/W	H'00000000	H'FFFFCC94	8, 16, 32
Bus function extending register	BSCEHR	R/W	H'0000	H'FFFFE89A	8, 16

8.2.1 DTC Mode Register A (MRA)

MRA selects DTC operating mode. MRA cannot be accessed directly by the CPU.

Bit:	7	6	5	4	3	2	1	0
	MD[1:0]		Sz[1:0]		SM[1:0]		-	-
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

Bit	Bit Name	Initial Value	R/W	Description
7, 6	MD[1:0]	Undefined	—	DTC Mode 1 and 0 Specify DTC transfer mode. 00: Normal transfer mode 01: Repeat transfer mode 10: Block transfer mode 11: Setting prohibited
5, 4	Sz[1:0]	Undefined	—	DTC Data Transfer Size 1 and 0 Specify the size of data to be transferred. 00: Byte-size transfer 01: Word-size transfer 10: Longword-size transfer 11: Setting prohibited
3, 2	SM[1:0]	Undefined	—	Source Address Mode 1 and 0 Specify an SAR operation after a data transfer. 0x: SAR is fixed (SAR writeback is skipped) 10: SAR is incremented after a transfer (by 1 when Sz1 and Sz0 = B'00; by 2 when Sz1 and Sz0 = B'01; by 4 when Sz1 and Sz0 = B'10) 11: SAR is decremented after a transfer (by 1 when Sz1 and Sz0 = B'00; by 2 when Sz1 and Sz0 = B'01; by 4 when Sz1 and Sz0 = B'10)

Bit	Bit Name	Initial Value	R/W	Description
1, 0	—	Undefined	—	Reserved The write value should always be 0.

[Legend]
x: Don't care

8.2.2 DTC Mode Register B (MRB)

MRB selects DTC operating mode. MRB cannot be accessed directly by the CPU.

Bit:	7	6	5	4	3	2	1	0
	CHNE	CHNS	DISEL	DTS	DM[1:0]		-	-
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

Bit	Bit Name	Initial Value	R/W	Description
7	CHNE	Undefined	—	DTC Chain Transfer Enable Specifies the chain transfer. For details, see section 8.5.6, Chain Transfer. The chain transfer condition is selected by the CHNS bit. 0: Disables the chain transfer 1: Enables the chain transfer
6	CHNS	Undefined	—	DTC Chain Transfer Select Specifies the chain transfer condition. If the following transfer is a chain transfer, the completion check of the specified transfer count is not performed and activation source flag or DTCER is not cleared. 0: Chain transfer every time 1: Chain transfer only when transfer counter = 0

Bit	Bit Name	Initial Value	R/W	Description
5	DISEL	Undefined	—	DTC Interrupt Select When this bit is set to 1, an interrupt request is generated to the CPU every time a data transfer or a block transfer ends. When this bit is set to 0, a CPU interrupt request is only generated when the specified number of data transfers end. Note: This bit should be cleared to 0 when the IIC2 is selected as the activation source.
4	DTS	Undefined	—	DTC Transfer Mode Select Specifies either the source or destination as repeat or block area during repeat or block transfer mode. 0: Specifies the destination as repeat or block area 1: Specifies the source as repeat or block area
3, 2	DM[1:0]	Undefined	—	Destination Address Mode 1 and 0 Specify a DAR operation after a data transfer. 0x: DAR is fixed (DAR writeback is skipped) 10: DAR is incremented after a transfer (by 1 when Sz1 and Sz0 = B'00; by 2 when Sz1 and Sz0 = B'01; by 4 when Sz1 and Sz0 = B'10) 11: SAR is decremented after a transfer (by 1 when Sz1 and Sz0 = B'00; by 2 when Sz1 and Sz0 = B'01; by 4 when Sz1 and Sz0 = B'10)
1, 0	—	Undefined	—	Reserved The write value should always be 0.

[Legend]
x: Don't care

8.2.3 DTC Source Address Register (SAR)

SAR is a 32-bit register that designates the source address of data to be transferred by the DTC.

SAR cannot be accessed directly from the CPU.

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

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

*: Undefined

8.2.4 DTC Destination Address Register (DAR)

DAR is a 32-bit register that designates the destination address of data to be transferred by the DTC.

DAR cannot be accessed directly from the CPU.

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

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

*: Undefined

8.2.5 DTC Transfer Count Register A (CRA)

CRA is a 16-bit register that designates the number of times data is to be transferred by the DTC.

In normal transfer mode, CRA functions as a 16-bit transfer counter (1 to 65,536). It is decremented by 1 every time data is transferred, and bit DTCEn (n = 15 to 0) corresponding to the activation source is cleared and then an interrupt is requested to the CPU when the count reaches H'0000. The transfer count is 1 when CRA = H'0001, 65,535 when CRA = H'FFFF, and 65,536 when CRA = H'0000.

In repeat transfer mode, CRA is divided into two parts: the upper eight bits (CRAH) and the lower eight bits (CRAL). CRAH holds the number of transfers while CRAL functions as an 8-bit transfer counter (1 to 256). CRAL is decremented by 1 every time data is transferred, and the contents of CRAH are sent to CRAL when the count reaches H'00. The transfer count is 1 when CRAH = CRAL = H'01, 255 when CRAH = CRAL = H'FF, and 256 when CRAH = CRAL = H'00.

In block transfer mode, CRA is divided into two parts: the upper eight bits (CRAH) and the lower eight bits (CRAL). CRAH holds the block size while CRAL functions as an 8-bit block-size counter (1 to 256 for byte, word, or longword). CRAL is decremented by 1 every time a byte (word or longword) data is transferred, and the contents of CRAH are sent to CRAL when the count reaches H'00. The block size is 1 byte (word or longword) when CRAH = CRAL =H'01, 255 bytes (words or longwords) when CRAH = CRAL = H'FF, and 256 bytes (words or longwords) when CRAH = CRAL =H'00.

CRA cannot be accessed directly from the CPU.

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

*: Undefined

8.2.6 DTC Transfer Count Register B (CRB)

CRB is a 16-bit register that designates the number of times data is to be transferred by the DTC in block transfer mode. It functions as a 16-bit transfer counter (1 to 65,536) that is decremented by 1 every time a block of data is transferred, and bit DTCEn (n = 15 to 0) corresponding to the activation source is cleared and then an interrupt is requested to the CPU when the count reaches H'0000. The transfer count is 1 when CRB = H'0001, 65,535 when CRB = H'FFFF, and 65,536 when CRB = H'0000.

CRB is not available in normal and repeat modes and cannot be accessed directly by the CPU.

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

* : Undefined

8.2.7 DTC Enable Registers A to E (DTCERA to DTCERE)

DTCER which is comprised of eight registers, DTCERA to DTCERE, is a register that specifies DTC activation interrupt sources. The correspondence between interrupt sources and DTCE bits is shown in table 8.2.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTCE15	DTCE14	DTCE13	DTCE12	DTCE11	DTCE10	DTCE9	DTCE8	DTCE7	DTCE6	DTCE5	DTCE4	DTCE3	DTCE2	DTCE1	DTCE0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	DTCE15	0	R/W	DTC Activation Enable 15 to 0 If set to 1, the corresponding interrupt source is specified as a DTC activation source. [Clearing conditions] • Writing 0 to the bit after reading 1 from it • When the DISEL bit is 1 and the data transfer has ended • When the specified number of transfers have ended These bits are not cleared when the DISEL bit is 0 and the specified number of transfers have not ended [Setting condition] • Writing 1 to the bit after reading 0 from it
14	DTCE14	0	R/W	
13	DTCE13	0	R/W	
12	DTCE12	0	R/W	
11	DTCE11	0	R/W	
10	DTCE10	0	R/W	
9	DTCE9	0	R/W	
8	DTCE8	0	R/W	
7	DTCE7	0	R/W	
6	DTCE6	0	R/W	
5	DTCE5	0	R/W	
4	DTCE4	0	R/W	
3	DTCE3	0	R/W	
2	DTCE2	0	R/W	
1	DTCE1	0	R/W	
0	DTCE0	0	R/W	

8.2.8 DTC Control Register (DTCCR)

DTCCR specifies transfer information read skip.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	RRS	RCHNE	-	-	ERR
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R	R	R/(W)*

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	RRS	0	R/W	DTC Transfer Information Read Skip Enable Controls the vector address read and transfer information read. A DTC vector number is always compared with the vector number for the previous activation. If the vector numbers match and this bit is set to 1, the DTC data transfer is started without reading a vector address and transfer information. If the previous DTC activation is a chain transfer, the vector address read and transfer information read are always performed. However, when the DTPR bit in the bus function extending register (BSCEHR) is set to 1, transfer information read skip is not performed regardless of the setting of this bit. 0: Transfer read skip is not performed. 1: Transfer read skip is performed when the vector numbers match.
3	RCHNE	0	R/W	Chain Transfer Enable After DTC Repeat Transfer Enables/disables the chain transfer while transfer counter (CRAL) is 0 in repeat transfer mode. In repeat transfer mode, the CRAH value is written to CRAL when CRAL is 0. Accordingly, chain transfer may not occur when CRAL is 0. If this bit is set to 1, the chain transfer is enabled when CRAH is written to CRAL. 0: Disables the chain transfer after repeat transfer 1: Enables the chain transfer after repeat transfer

Bit	Bit Name	Initial Value	R/W	Description
2, 1	—	All 0	R	Reserved These are read-only bits and cannot be modified.
0	ERR	0	R/(W)*	Transfer Stop Flag Indicates that the DTC address error or NMI interrupt request has occurred. If a DTC address error or NMI interrupt occurs while the DTC is active, address error handling or NMI interrupt handling processing is executed after the DTC has released the bus mastership. The DTC stops in the transfer information writing state after transferring data. 0: No interrupt occurs 1: An interrupt occurs [Clearing condition] • When writing 0 after reading 1

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

8.2.9 DTC Vector Base Register (DTCVBR)

DTCVBR is a 32-bit register that specifies the base address for vector table address calculation.

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

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

Bit	Bit Name	Initial Value	R/W	Description
31 to 12		All 0	R/W	Bits 11 to 0 are always read as 0. The write value should always be 0.
11 to 0	—	All 0	R	

8.2.10 Bus Function Extending Register (BSCEHR)

BSCEHR is a 16-bit register that specifies the timing of bus release by the DTC and other functions. This register can be used to give higher priority to the transfer by the DTC and configure the functions that can reduce the number of cycles over which the DTC is active. For more details, see section 9.4.8, Bus Function Extending Register (BSCEHR).

8.3 Activation Sources

The DTC is activated by an interrupt request. The interrupt source is selected by DTCER. A DTC activation source can be selected by setting the corresponding bit in DTCER; the CPU interrupt source can be selected by clearing the corresponding bit in DTCER. At the end of a data transfer (or the last consecutive transfer in the case of chain transfer), the activation source interrupt flag or corresponding DTCER bit is cleared.

8.4 Location of Transfer Information and DTC Vector Table

Locate the transfer information in the data area. The start address of transfer information should be located at the address that is a multiple of four (4n). Otherwise, the lower two bits are ignored during access ([1:0] = B'00.) Transfer information located in the data area is shown in figure 8.2.

Only in the case where all transfer sources/transfer destinations are in on-chip RAM and on-chip peripheral modules, short address mode can be selected by setting the DTSA bit in the bus function extending register (BSCEHR) to 1 (see section 9.4.8, Bus Function Extending Register (BSCEHR)).
Normally, four longwords of transfer information has to be read. But if short address mode is selected, the size of transfer information is reduced to three longwords, which can shorten the period over which the DTC is active.

The DTC reads the start address of the transfer information from the vector table for every activation source and reads the transfer information from this start address. Figure 8.3 shows correspondences between the DTC vector table and transfer information.

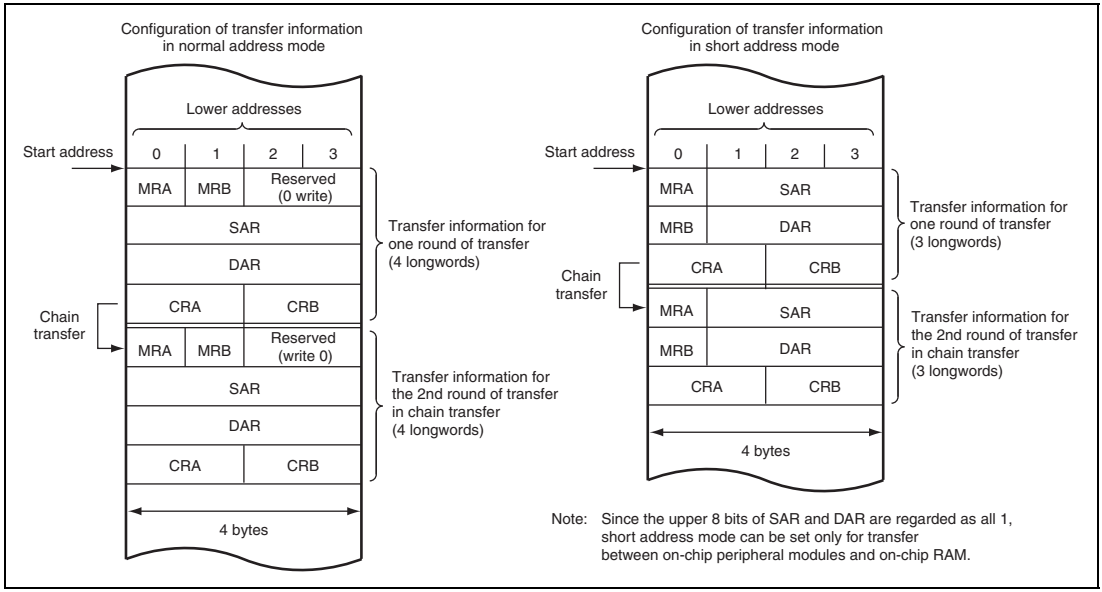


Figure 8.2 Transfer Information on Data Area

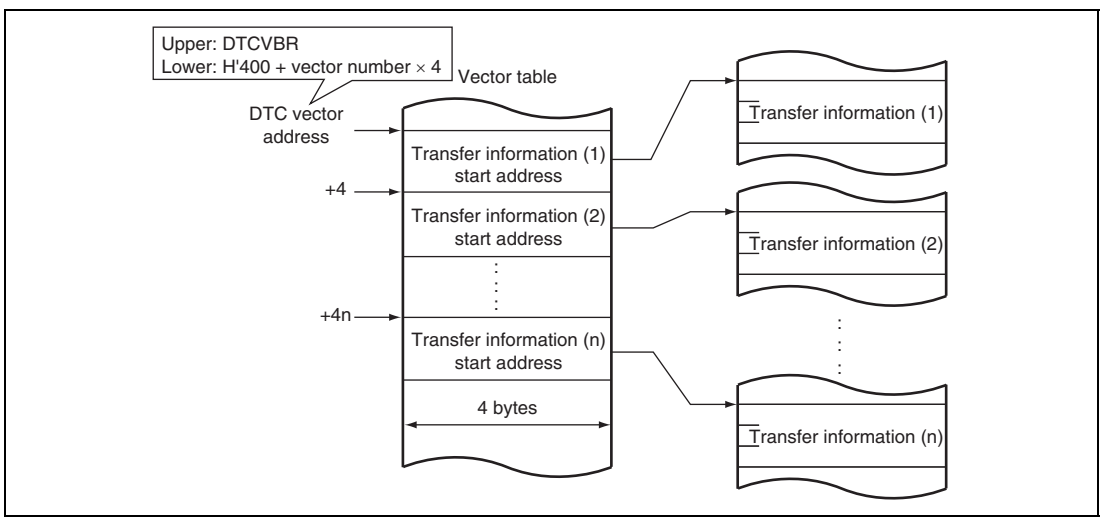


Figure 8.3 Correspondence between DTC Vector Address and Transfer Information

Table 8.2 shows correspondence between the DTC activation source and vector address.

Table 8.2 Interrupt Sources, DTC Vector Addresses, and Corresponding DTCEs

Origin of Activation Source	Activation Source	Vector Number	DTC Vector Address Offset	DTCE*1	Transfer Source	Transfer Destination	Priority
External pin	IRQ0	64	H'500	DTCERA15	Arbitrary*2	Arbitrary*2	High
	IRQ1	65	H'504	DTCERA14	Arbitrary*2	Arbitrary*2	
	IRQ2	66	H'508	DTCERA13	Arbitrary*2	Arbitrary*2	
	IRQ3	67	H'50C	DTCERA12	Arbitrary*2	Arbitrary*2	
	IRQ4	68	H'510	DTCERA11	Arbitrary*2	Arbitrary*2	
	IRQ5	69	H'514	DTCERA10	Arbitrary*2	Arbitrary*2	
	IRQ6	70	H'518	DTCERA9	Arbitrary*2	Arbitrary*2	
	IRQ7	71	H'51C	DTCERA8	Arbitrary*2	Arbitrary*2	
MTU2_0	TGIA_0	88	H'560	DTCERB15	Arbitrary*2	Arbitrary*2	
	TGIB_0	89	H'564	DTCERB14	Arbitrary*2	Arbitrary*2	
	TGIC_0	90	H'568	DTCERB13	Arbitrary*2	Arbitrary*2	
	TGID_0	91	H'56C	DTCERB12	Arbitrary*2	Arbitrary*2	
MTU2_1	TGIA_1	96	H'580	DTCERB11	Arbitrary*2	Arbitrary*2	
	TGIB_1	97	H'584	DTCERB10	Arbitrary*2	Arbitrary*2	
MTU2_2	TGIA_2	104	H'5A0	DTCERB9	Arbitrary*2	Arbitrary*2	
	TGIB_2	105	H'5A4	DTCERB8	Arbitrary*2	Arbitrary*2	
MTU2_3	TGIA_3	112	H'5C0	DTCERB7	Arbitrary*2	Arbitrary*2	
	TGIB_3	113	H'5C4	DTCERB6	Arbitrary*2	Arbitrary*2	
	TGIC_3	114	H'5C8	DTCERB5	Arbitrary*2	Arbitrary*2	
	TGID_3	115	H'5CC	DTCERB4	Arbitrary*2	Arbitrary*2	
MTU2_4	TGIA_4	120	H'5E0	DTCERB3	Arbitrary*2	Arbitrary*2	
	TGIB_4	121	H'5E4	DTCERB2	Arbitrary*2	Arbitrary*2	
	TGIC_4	122	H'5E8	DTCERB1	Arbitrary*2	Arbitrary*2	
	TGID_4	123	H'5EC	DTCERB0	Arbitrary*2	Arbitrary*2	
	TCIV_4	124	H'5F0	DTCERC15	Arbitrary*2	Arbitrary*2	Low

Origin of Activation Source	Activation Source	Vector Number	DTC Vector Address Offset	DTCE*1	Transfer Source	Transfer Destination	Priority
MTU2_5	TGIU_5	128	H'600	DTCERC14	Arbitrary*2	Arbitrary*2	High
	TGIV_5	129	H'604	DTCERC13	Arbitrary*2	Arbitrary*2	
	TGIW_5	130	H'608	DTCERC12	Arbitrary*2	Arbitrary*2	
MTU2S_3	TGIA_3S	160	H'680	DTCERC3	Arbitrary*2	Arbitrary*2	
	TGIB_3S	161	H'684	DTCERC2	Arbitrary*2	Arbitrary*2	
	TGIC_3S	162	H'688	DTCERC1	Arbitrary*2	Arbitrary*2	
	TGID_3S	163	H'68C	DTCERC0	Arbitrary*2	Arbitrary*2	
MTU2S_4	TGIA_4S	168	H'6A0	DTCERD15	Arbitrary*2	Arbitrary*2	
	TGIB_4S	169	H'6A4	DTCERD14	Arbitrary*2	Arbitrary*2	
	TGIC_4S	170	H'6A8	DTCERD13	Arbitrary*2	Arbitrary*2	
	TGID_4S	171	H'6AC	DTCERD12	Arbitrary*2	Arbitrary*2	
	TCIV_4S	172	H'6B0	DTCERD11	Arbitrary*2	Arbitrary*2	
MTU2S_5	TGIU_5S	176	H'6C0	DTCERD10	Arbitrary*2	Arbitrary*2	
	TGIV_5S	177	H'6C4	DTCERD9	Arbitrary*2	Arbitrary*2	
	TGIW_5S	178	H'6C8	DTCERD8	Arbitrary*2	Arbitrary*2	
CMT_0	CMI_0	184	H'6E0	DTCERD7	Arbitrary*2	Arbitrary*2	
CMT_1	CMI_1	188	H'6F0	DTCERD6	Arbitrary*2	Arbitrary*2	
A/D_0, A/D_1	ADI_0	200	H'720	DTCERD5	ADDR0 to ADDR3	Arbitrary*2	
	ADI_1	201	H'724	DTCERD4	ADDR4 to ADDR7	Arbitrary*2	
A/D_2	ADI_2	204	H'730	DTCERD3	ADDR8 to ADDR15	Arbitrary*2	
SCI_0	RXI_0	217	H'764	DTCERE15	SCRDR_0	Arbitrary*2	
	TXI_0	218	H'768	DTCERE14	Arbitrary*2	SCTDR_0	
SCI_1	RXI_1	221	H'774	DTCERE13	SCRDR_1	Arbitrary*2	
	TXI_1	222	H'778	DTCERE12	Arbitrary*2	SCTDR_1	
SCI_2	RXI_2	225	H'784	DTCERE11	SCRDR_2	Arbitrary*2	
	TXI_2	226	H'788	DTCERE10	Arbitrary*2	SCTDR_2	
SCIF	RXIF	229	H'794	DTCERE9	SCFRDR_3	Arbitrary*2	
	TXIF	231	H'79C	DTCERE8	Arbitrary*2	SCFTDR_3	Low

Origin of Activation Source	Activation Source	Vector Number	DTC Vector Address Offset	DTCE*1	Transfer Source	Transfer Destination	Priority
SSU	SSRXI	233	H'7A4	DTCERE7	SSRDR0 to SSRDR3	Arbitrary*2	High
	SSTXI	234	H'7A8	DTCERE6	Arbitrary*2	SSTDR0 to SSTDR3	
IIC2	IITXI	238	H'7B8	DTCERE5	Arbitrary*2	ICDRT	
	IIRXI	239	H'7BC	DTCERE4	ICDRR	Arbitrary*2	Low

Notes: 1. The DTCE bits with no corresponding interrupt are reserved, and the write value should always be 0. To leave software standby mode with an interrupt, write 0 to the corresponding DTCE bit.

2. An external memory, a memory-mapped external device, an on-chip memory, or an on-chip peripheral module (except DMAC, DTC, BSC, UBC, and FLASH) can be selected as the source or destination. Note that at least either the source or destination must be an on-chip peripheral module; transfer cannot be done among an external memory, a memory-mapped external device, and an on-chip memory.

8.5 Operation

There are three transfer modes: normal, repeat, and block transfer modes. Since transfer information is in the data area, it is possible to transfer data over any required number of channels. When activated, the DTC reads transfer information stored in the data are and transfers data according to the transfer information. After the data transfer is complete, it writes updated transfer information back to the data area.

The DTC specifies the source address and destination address in SAR and DAR, respectively. After a transfer, SAR and DAR are incremented, decremented, or fixed independently.

Table 8.3 shows the DTC transfer modes.

Table 8.3 DTC Transfer Modes

Transfer Mode	Size of Data Transferred at One Transfer Request	Memory Address Increment or Decrement	Transfer Count
Normal	1 byte/word/longword	Incremented/decremented by 1, 2, or 4, or fixed	1 to 65536
Repeat*1	1 byte/word/longword	Incremented/decremented by 1, 2, or 4, or fixed	1 to 256*3
Block*2	Block size specified by CRAH (1 to 256 bytes/words/longwords)	Incremented/decremented by 1, 2, or 4, or fixed	1 to 65536*4

Notes: 1. Either source or destination is specified to repeat area.
2. Either source or destination is specified to block area.
3. After transfer of the specified transfer count, initial state is recovered to continue the operation.
4. Number of transfers of the specified block size of data.

Setting the CHNE bit in MRB to 1 makes it possible to perform a number of transfers with a single activation (chain transfer). Setting the CHNS bit in MRB to 1 can also be made to have chain transfer performed only when the transfer counter value is 0.

Figure 8.4 shows a flowchart of DTC operation, and table 8.4 summarizes the conditions for DTC transfers including chain transfer (combinations for performing the second and third transfers are omitted).

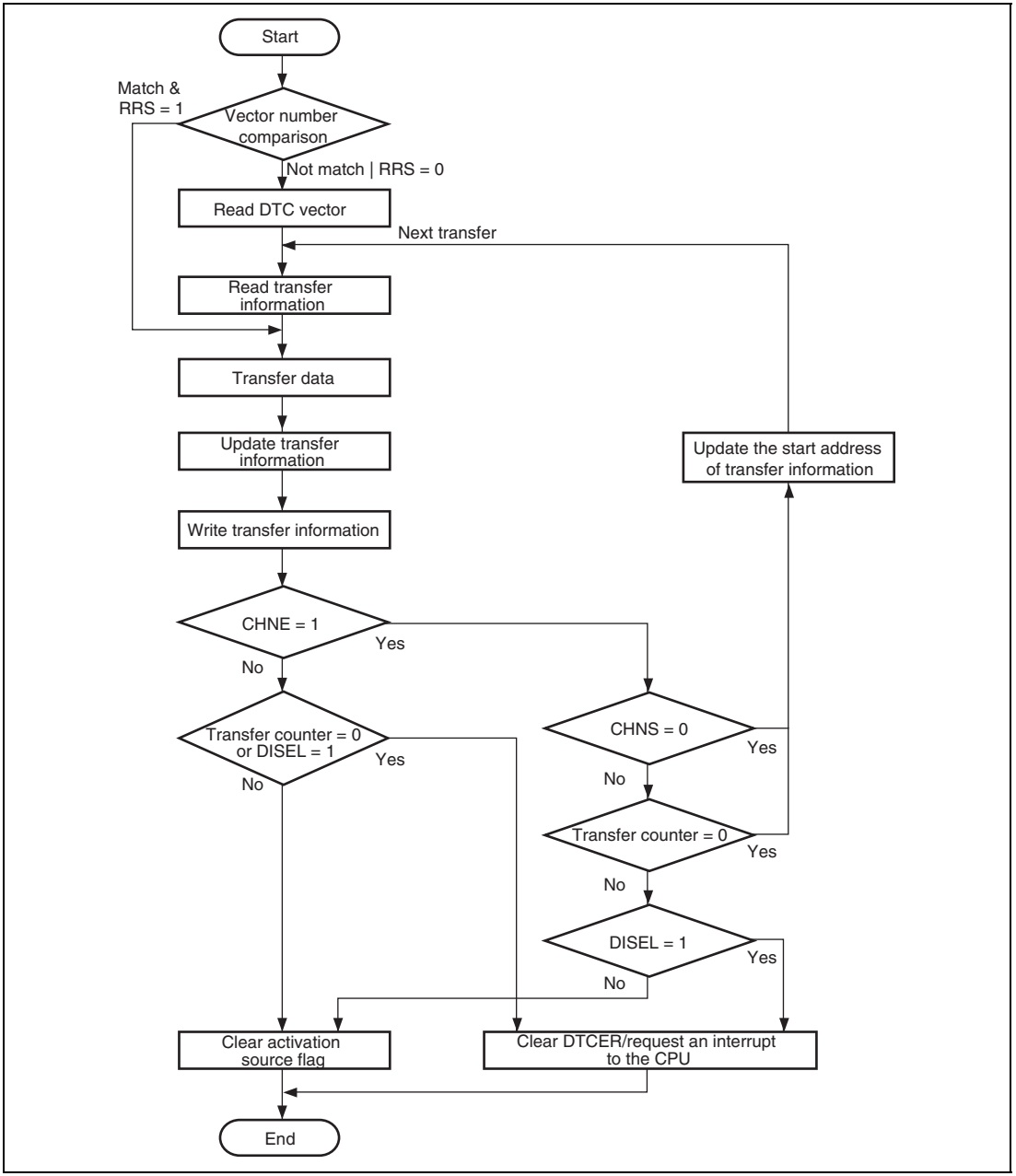


Figure 8.4 Flowchart of DTC Operation

Table 8.4 DTC Transfer Conditions (Chain Transfer Conditions Included)

	1st Transfer					2nd Transfer					
Transfer Mode	CHNE	CHNS	RCHNE	DISEL	Transfer Counter*1	CHNE	CHNS	RCHNE	DISEL	Transfer Counter*1	DTC Transfer
Normal	0	—	—	0	Not 0	—	—	—	—	—	Ends at 1st transfer
	0	—	—	0	0	—	—	—	—	—	Ends at 1st transfer Interrupt request to CPU
	0	—	—	1	—	—	—	—	—	—	
	1	0	—	—	—	0	—	—	0	Not 0	Ends at 2nd transfer
						0	—	—	0	0	Ends at 2nd transfer Interrupt request to CPU
						0	—	—	1	—	
	1	1	—	0	Not 0	—	—	—	—	—	Ends at 1st transfer
	1	1	—	1	Not 0	—	—	—	—	—	Ends at 1st transfer Interrupt request to CPU
	1	1	—	—	0	0	—	—	0	Not 0	Ends at 2nd transfer
						0	—	—	0	0	Ends at 2nd transfer Interrupt request to CPU
						0	—	—	1	—	

	1st Transfer					2nd Transfer					
Transfer Mode	**CHNE**	**CHNS**	**RCHNE**	**DISEL**	**Transfer Counter*1**	**CHNE**	**CHNS**	**RCHNE**	**DISEL**	**Transfer Counter*1**	**DTC Transfer**
Repeat	0	—	—	0	—	—	—	—	—	—	Ends at 1st transfer
	0	—	—	1	—	—	—	—	—	—	Ends at 1st transfer Interrupt request to CPU
	1	0	—	—	—	0	—	—	0	—	Ends at 2nd transfer
						0	—	—	1	—	Ends at 2nd transfer Interrupt request to CPU
	1	1	—	0	Not 0	—	—	—	—	—	Ends at 1st transfer
	1	1	—	1	Not 0	—	—	—	—	—	Ends at 1st transfer Interrupt request to CPU
	1	1	0	0	0*2	—	—	—	—	—	Ends at 1st transfer
	1	1	0	1	0*2	—	—	—	—	—	Ends at 1st transfer Interrupt request to CPU
	1	1	1	—	0*2	0	—	—	0	—	Ends at 2nd transfer
						0	—	—	1	—	Ends at 2nd transfer Interrupt request to CPU

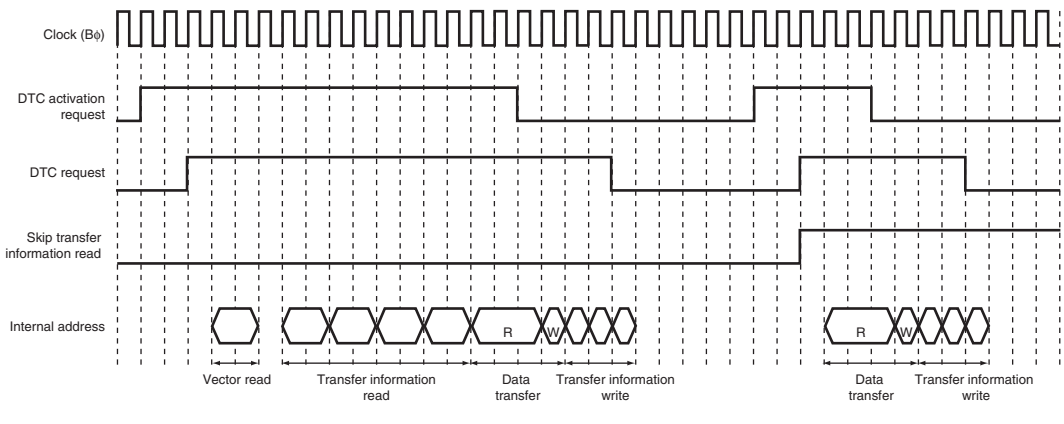
	1st Transfer					2nd Transfer					
Transfer Mode	**CHNE**	**CHNS**	**RCHNE**	**DISEL**	**Transfer Counter*1**	**CHNE**	**CHNS**	**RCHNE**	**DISEL**	**Transfer Counter*1**	**DTC Transfer**
Block	0	—	—	0	Not 0	—	—	—	—	—	Ends at 1st transfer
	0	—	—	0	0	—	—	—	—	—	Ends at 1st transfer Interrupt request to CPU
	0	—	—	1	—	—	—	—	—	—	
	1	0	—	—	—	0	—	—	0	Not 0	Ends at 2nd transfer
						0	—	—	0	0	Ends at 2nd transfer Interrupt request to CPU
						0	—	—	1	—	
	1	1	—	0	—	—	—	—	—	—	Ends at 1st transfer
	1	1	—	1	Not 0	—	—	—	—	—	Ends at 1st transfer Interrupt request to CPU
	1	1	—	1	0	0	—	—	0	Not 0	Ends at 2nd transfer
						0	—	—	0	0	Ends at 2nd transfer Interrupt request to CPU
						0	—	—	1	—	

Notes: 1. CRA in normal mode transfer, CRAL in repeat transfer mode, or CRB in block transfer mode
2. When the contents of the CRAH is written to the CRAL in repeat transfer mode

8.5.1 Transfer Information Read Skip Function

By setting the RRS bit of DTCCR, the vector address read and transfer information read can be skipped. The current DTC vector number is always compared with the vector number of previous activation. If the vector numbers match when RRS = 1, a DTC data transfer is performed without reading the vector address and transfer information. If the previous activation is a chain transfer, the vector address read and transfer information read are always performed. Figure 8.5 shows the transfer information read skip timing.

To modify the vector table and transfer information, temporarily clear the RRS bit to 0, modify the vector table and transfer information, and then set the RRS bit to 1 again. When the RRS bit is cleared to 0, the stored vector number is deleted, and the updated vector table and transfer information are read at the next activation.
If the DTPR bit in the bus function extending register (BSCEHR) is set to 1, this function is always disabled.



Note: The DTC request signal indicates the state of internal bus request after the DTC activation source has been determined.

Figure 8.5 Transfer Information Read Skip Timing
(Activated by On-Chip Peripheral Module; Iϕ: Bϕ: Pϕ =1: 1/2: 1/2;
Data Transferred from On-Chip Peripheral Module to On-Chip RAM;
Transfer Information Is Written in 3 States)

8.5.2 Transfer Information Writeback Skip Function

By specifying bit SM1 in MRA and bit DM1 in MRB to the fixed address mode, a part of transfer information will not be written back. Table 8.5 shows the transfer information writeback skip condition and writeback skipped registers. Note that the CRA and CRB are always written back. The writeback of the MRA and MRB are always skipped.

Table 8.5 Transfer Information Writeback Skip Condition and Writeback Skipped Registers

SM1	DM1	SAR	DAR
0	0	Skipped	Skipped
0	1	Skipped	Written back
1	0	Written back	Skipped
1	1	Written back	Written back

8.5.3 Normal Transfer Mode

In normal transfer mode, data are transferred in one byte, one word, or one longword units in response to a single activation request. From 1 to 65,536 transfers can be specified. The transfer source and destination addresses can be specified as incremented, decremented, or fixed. When the specified number of transfers ends, an interrupt can be requested to the CPU.

Table 8.6 lists the register function in normal transfer mode. Figure 8.6 shows the memory map in normal transfer mode.

Table 8.6 Register Function in Normal Transfer Mode

Register	Function	Written Back Value
SAR	Source address	Incremented/decremented/fixed*
DAR	Destination address	Incremented/decremented/fixed*
CRA	Transfer count A	CRA – 1
CRB	Transfer count B	Not updated

Note: * Transfer information writeback is skipped.

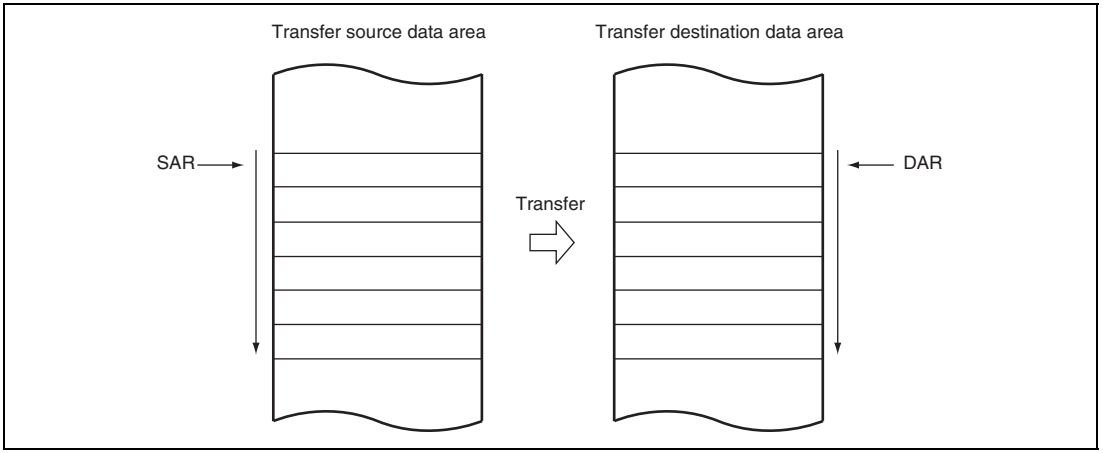


Figure 8.6 Memory Map in Normal Transfer Mode

8.5.4 Repeat Transfer Mode

In repeat transfer mode, data are transferred in one byte, one word, or one longword units in response to a single activation request. By the DTS bit in MRB, either the source or destination can be specified as a repeat area. From 1 to 256 transfers can be specified. When the specified number of transfers ends, the transfer counter and address register specified as the repeat area is restored to the initial state, and transfer is repeated. The other address register is then incremented, decremented, or left fixed. In repeat transfer mode, the transfer counter (CRAL) is updated to the value specified in CRAH when CRAL becomes H'00. Thus the transfer counter value does not reach H'00, and therefore a CPU interrupt cannot be requested when DISEL = 0.

Table 8.7 lists the register function in repeat transfer mode. Figure 8.7 shows the memory map in repeat transfer mode.

Table 8.7 Register Function in Repeat Transfer Mode

		Written Back Value	
Register	**Function**	**CRAL is not 1**	**CRAL is 1**
SAR	Source address	Incremented/decremented/fixed*	DTS = 0: Incremented/ decremented/fixed* DTS = 1: SAR initial value
DAR	Destination address	Incremented/decremented/fixed*	DTS = 0: DAR initial value DTS = 1: Incremented/ decremented/fixed*
CRAH	Transfer count storage	CRAH	CRAH
CRAL	Transfer count A	CRAL – 1	CRAH
CRB	Transfer count B	Not updated	Not updated

Note: * Transfer information writeback is skipped.

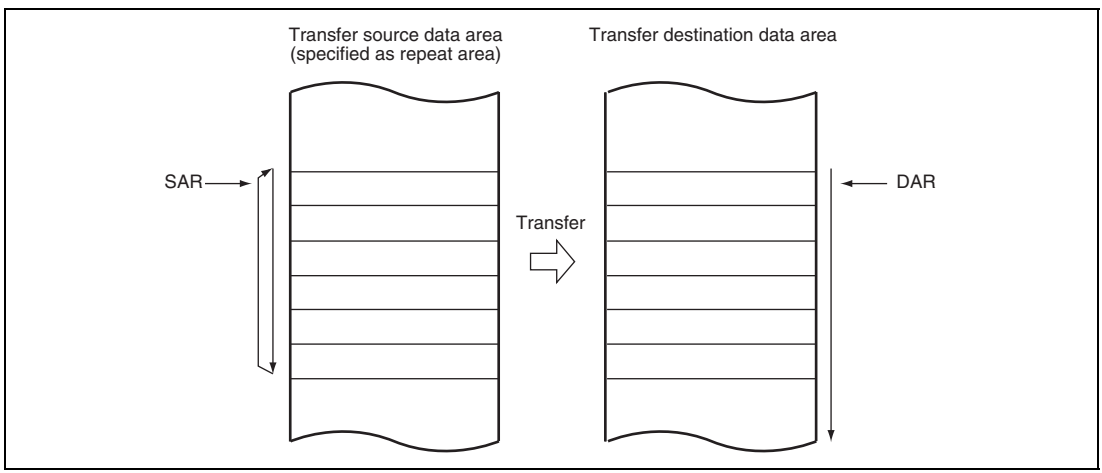


Figure 8.7 Memory Map in Repeat Transfer Mode (When Transfer Source Is Specified as Repeat Area)

8.5.5 Block Transfer Mode

In block transfer mode, data are transferred in block units in response to a single activation request. Either the transfer source or the transfer destination is designated as a block area by the DTS bit in MRB.

The block size is 1 to 256 bytes (1 to 256 words, or 1 to 256 longwords). When the block data transfer of one block ends, the block size counter (CRAL) and address register (SAR when DTS = 1 or DAR when DTS = 0) specified as the block area is restored to the initial state. The other address register is then incremented, decremented, or left fixed. From 1 to 65,536 transfers can be specified. When the specified number of transfers ends, an interrupt is requested to the CPU.

Table 8.8 lists the register function in block transfer mode. Figure 8.8 shows the memory map in block transfer mode.

Table 8.8 Register Function in Block Transfer Mode

Register	Function	Written Back Value
SAR	Source address	DTS = 0: Incremented/decremented/fixed*
		DTS = 1: SAR initial value
DAR	Destination address	DTS = 0: DAR initial value
		DTS = 1: Incremented/decremented/fixed*
CRAH	Block size storage	CRAH
CRAL	Block size counter	CRAH
CRB	Block transfer counter	CRB − 1

Note: * Transfer information writeback is skipped.

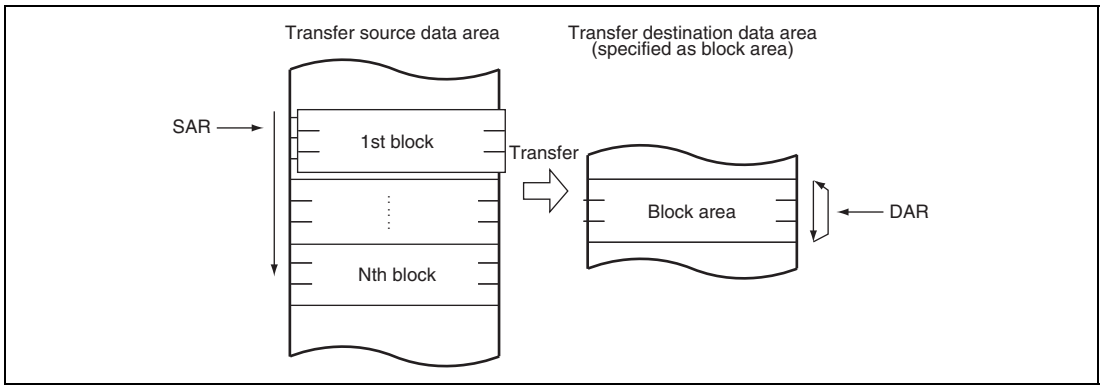


Figure 8.8 Memory Map in Block Transfer Mode (When Transfer Destination Is Specified as Block Area)

8.5.6 Chain Transfer

Setting the CHNE bit in MRB to 1 enables a number of data transfers to be performed consecutively in response to a single transfer request. Setting the CHNE and CHNS bits in MRB set to 1 enables a chain transfer only when the transfer counter reaches 0. SAR, DAR, CRA, CRB, MRA, and MRB, which define data transfers, can be set independently. Figure 8.9 shows the chain transfer operation.

In the case of transfer with CHNE set to 1, an interrupt request to the CPU is not generated at the end of the specified number of transfers or by setting the DISEL bit to 1, and the interrupt source flag for the activation source and DTCER are not affected.

In repeat transfer mode, setting the RCHNE bit in DTCCR and the CHNE and CHNS bits in MRB to 1 enables a chain transfer after transfer with transfer counter = 1 has been completed.

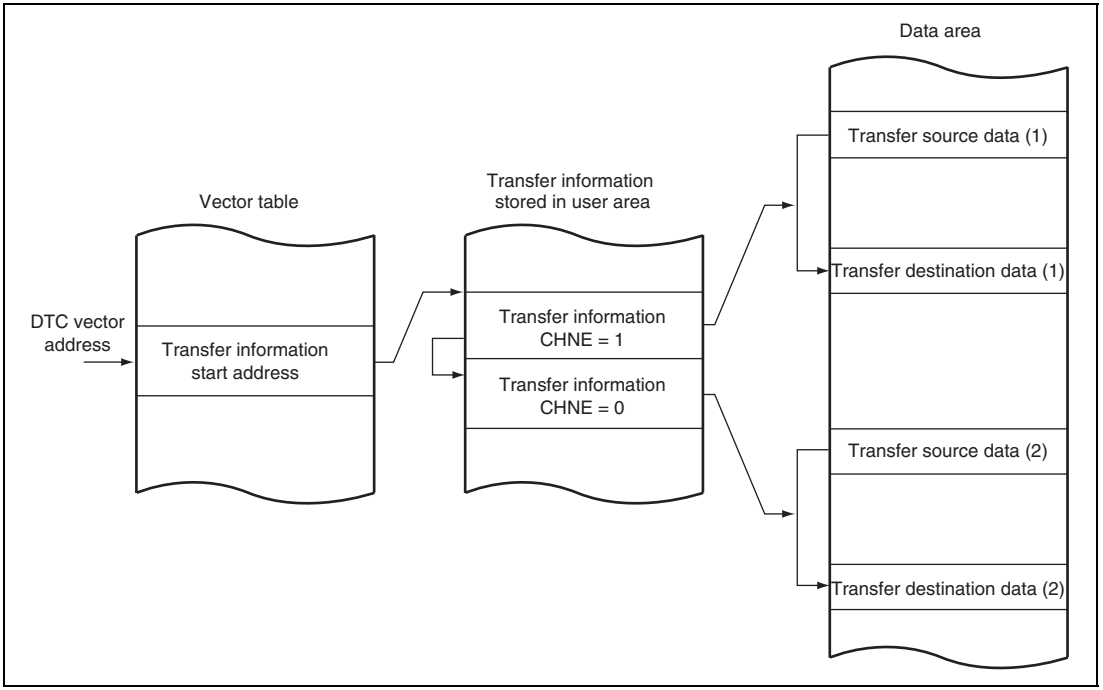
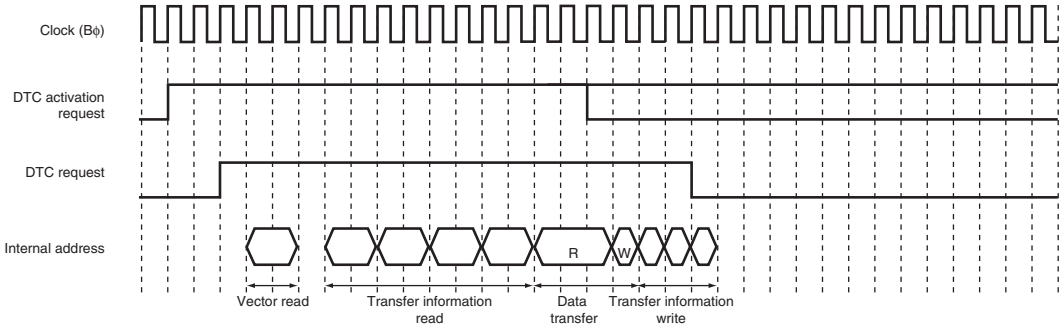


Figure 8.9 Operation of Chain Transfer

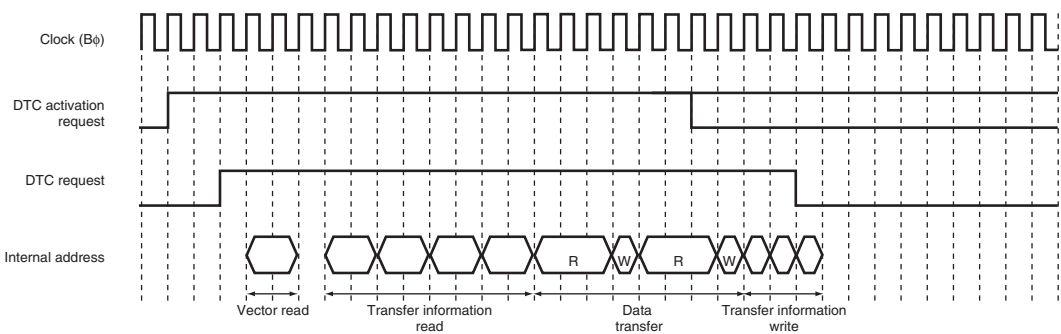
8.5.7 Operation Timing

Figures 8.10 to 8.15 show the DTC operation timings.



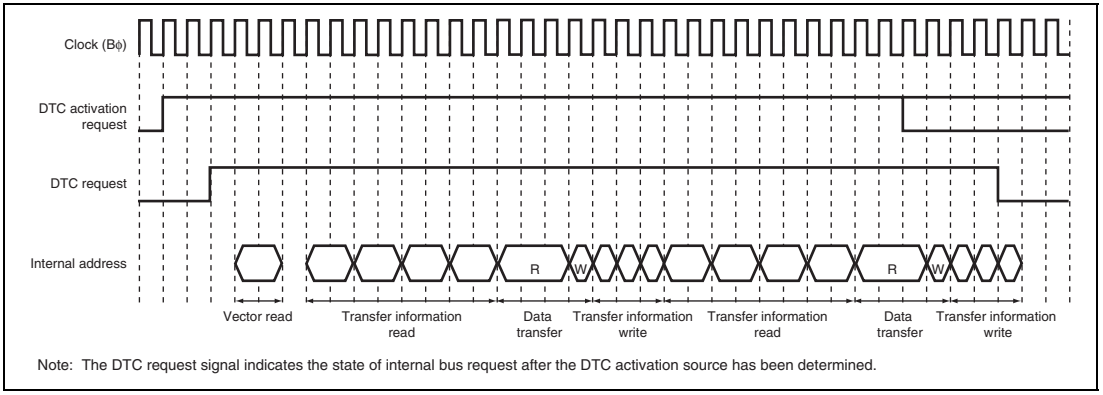
Note: The DTC request signal indicates the state of internal bus request after the DTC activation source has been determined.

Figure 8.10 Example of DTC Operation Timing: Normal Transfer Mode or Repeat Transfer Mode (Activated by On-Chip Peripheral Module; Iϕ: Bϕ: Pϕ =1: 1/2: 1/2; Data Transferred from On-Chip Peripheral Module to On-Chip RAM; Transfer Information Is Written in 3 Cycles)



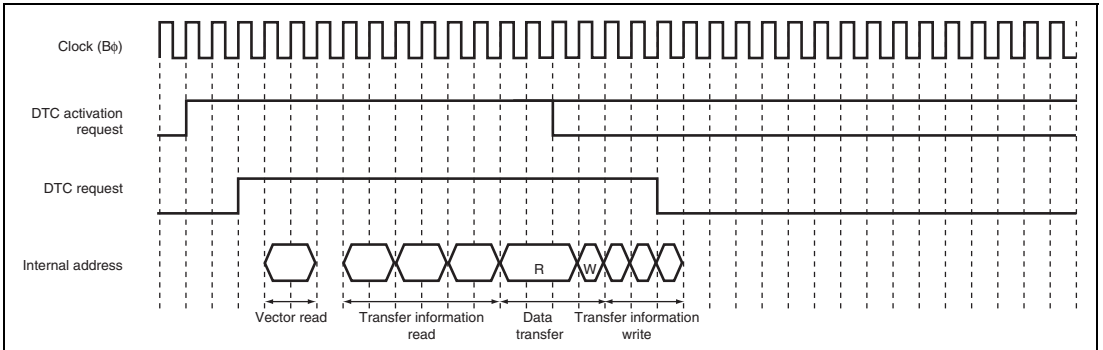
Note: The DTC request signal indicates the state of internal bus request after the DTC activation source has been determined.

Figure 8.11 Example of DTC Operation Timing: Block Transfer Mode with Block Size = 2 (Activated by On-Chip Peripheral Module; Iϕ: Bϕ: Pϕ =1: 1/2: 1/2; Data Transferred from On-Chip Peripheral Module to On-Chip RAM; Transfer Information Is Written in 3 Cycles)



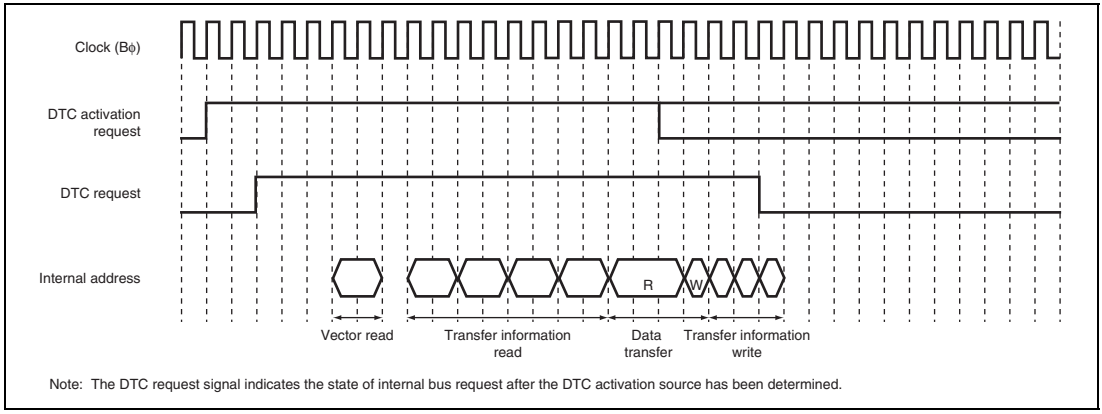
Note: The DTC request signal indicates the state of internal bus request after the DTC activation source has been determined.

Figure 8.12 Example of DTC Operation Timing: Chain Transfer (Activated by On-Chip Peripheral Module; Iφ: Bφ: Pφ =1: 1/2: 1/2; Data Transferred from On-Chip Peripheral Module to On-Chip RAM; Transfer Information Is Written in 3 Cycles)



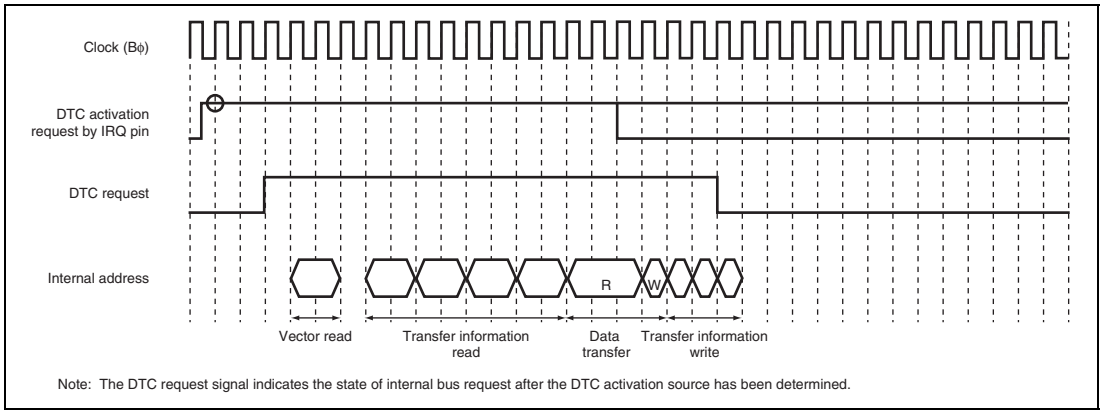
Note: The DTC request signal indicates the state of internal bus request after the DTC activation source has been determined.

Figure 8.13 Example of DTC Operation Timing: Normal or Repeat Transfer in Short Address Mode (Activated by On-Chip Peripheral Module; Iφ: Bφ: Pφ =1: 1/2: 1/2; Data Transferred from On-Chip Peripheral Module to On-Chip RAM; Transfer Information Is Written in 3 Cycles)



Note: The DTC request signal indicates the state of internal bus request after the DTC activation source has been determined.

Figure 8.14 Example of DTC Operation Timing: Normal or Repeat Transfer with DTPR = 1 (Activated by On-Chip Peripheral Module; Iϕ: Bϕ: Pϕ =1: 1/2: 1/2; Data Transferred from On-Chip Peripheral Module to On-Chip RAM; Transfer Information Is Written in 3 Cycles)



Note: The DTC request signal indicates the state of internal bus request after the DTC activation source has been determined.

Figure 8.15 Example of DTC Operation Timing: Normal or Repeat Transfer (Activated by IRQ; Iϕ: Bϕ: Pϕ =1: 1/2: 1/2; Data Transferred from On-Chip Peripheral Module to On-Chip RAM; Transfer Information Is Written in 3 Cycles)

8.5.8 Number of DTC Execution Cycles

Table 8.9 shows the execution status for a single DTC data transfer, and table 8.10 shows the number of cycles required for each execution.

Table 8.9 DTC Execution Status

Mode	Vector Read I		Transfer Information Read J			Transfer Information Write K			Data Read L	Data Write M	Internal Operation N	
Normal	1	0*[1]	4	3*[4]	0*[1]	3	2*[2]	1*[3]	1	1	1	0*[1]
Repeat	1	0*[1]	4	3*[4]	0*[1]	3	2*[2]	1*[3]	1	1	1	0*[1]
Block transfer	1	0*[1]	4	3*[4]	0*[1]	3	2*[2]	1*[3]	1•P	1•P	1	0*[1]

[Legend]

P: Block size (initial setting of CRAH and CRAL)

Notes: 1. When transfer information read is skipped
2. When the SAR or DAR is in fixed mode
3. When the SAR and DAR are in fixed mode
4. When short address mode

Table 8.10 Number of Cycles Required for Each Execution State

Object to be Accessed		On-Chip RAM*1/ROM*2	On-Chip I/O Registers		External Devices*5		
Bus width		32 bits	8 bits*4	16 bits	8 bits	16 bits	32 bits
Access cycles		1Bϕ to 3Bϕ*1*2	2Pϕ	2Pϕ	2Bϕ	2Bϕ	2Bϕ
Execution status	Vector read S_I	1Bϕ to 3Bϕ*1*2	—	—	9Bϕ	5Bϕ	3Bϕ
	Transfer information read S_J	1Bϕ to 3Bϕ*1	—	—	9Bϕ	5Bϕ	3Bϕ
	Transfer information write S_K	1Bϕ to 3Bϕ*1	—	—	2Bϕ*6	2Bϕ*6	2Bϕ*6
	Byte data read S_L	1Bϕ to 3Bϕ*1	1Bϕ + 2Pϕ*3	1Bϕ + 2Pϕ*3	3Bϕ	3Bϕ	3Bϕ
	Word data read S_L	1Bϕ to 3Bϕ*1	—	1Bϕ + 2Pϕ*3	5Bϕ	3Bϕ	3Bϕ
	Longword data read S_L	1Bϕ to 3Bϕ*1	—	1Bϕ + 4Pϕ*3	9Bϕ	5Bϕ	3Bϕ
	Byte data write S_M	1Bϕ to 3Bϕ*1	1Bϕ + 2Pϕ*3	1Bϕ + 2Pϕ*3	2Bϕ*6	2Bϕ*6	2Bϕ*6
	Word data write S_M	1Bϕ to 3Bϕ*1	—	1Bϕ + 2Pϕ*3	2Bϕ*6	2Bϕ*6	2Bϕ*6
	Longword data write S_M	1Bϕ to 3Bϕ*1	—	1Bϕ + 4Pϕ*3	2Bϕ*6	2Bϕ*6	2Bϕ*6
	Internal operation S_N	1					

Notes: 1. Values for on-chip RAM. Number of cycles varies depending on the ratio of Iϕ:Bϕ.

	Read	Write
Iϕ:Bϕ = 1:1	3Bϕ	3Bϕ
Iϕ:Bϕ = 1:1/2	2Bϕ	1Bϕ
Iϕ:Bϕ = 1:1/3	2Bϕ	1Bϕ
Iϕ:Bϕ = 1:1/4 or less	1Bϕ	1Bϕ

2. Values for on-chip ROM. Number of cycles varies depending on the ratio of Iϕ:Bϕ.and are the same as on-chip RAM. Only vector read is possible.
3. The values in the table are those for the fastest case. Depending on the state of the internal bus, replace 1Bϕ by 1Pϕ in a slow case.
4. This applies to the IIC2.
5. Values are different depending on the BSC register setting. The values in the table are the sample for the case with no wait cycles and the WM bit in CSnWCR = 1.
6. Values are different depending on the bus state.
   The number of cycles increases when many external wait cycles are inserted in the case where writing is frequently executed, such as block transfer, and when the external bus is in use because the write buffer cannot be used efficiently in such cases. For details on the write buffer, see section 9.5.14 (2), Access in View of LSI Internal Bus Master.

The number of execution cycles is calculated from the formula below. Note that Σ means the sum of cycles for all transfers initiated by one activation event (the number of 1-valued CHNE bits in transfer information plus 1).

$$\text{Number of execution cycles} = I \bullet S_I + \Sigma (J \bullet S_J + K \bullet S_K + L \bullet S_L + M \bullet S_M) + N \bullet S_N$$

8.5.9 DTC Bus Release Timing

The DTC requests the bus mastership to the bus arbiter when an activation request occurs. The DTC releases the bus after a vector read, NOP cycle generation after a vector read, transfer information read, a single data transfer, or transfer information writeback. The DTC does not release the bus mastership during transfer information read, single data transfer, or transfer information writeback.
The bus release timing can be specified through the bus function extending register (BSCEHR). For details see section 9.4.8, Bus Function Extending Register (BSCEHR). The difference in bus release timing according to the register setting is summarized in table 8.11. Settings other than settings 1 to 5 are not allowed. The setting must not be changed while the DTC is active.

Figure 8.16 is a timing chart showing an example of bus release timing.

Table 8.11 DTC Bus Release Timing

	Bus Function Extending Register (BSCEHR) Setting					Bus Release Timing (O: Bus is released; x: Bus is not released)					
										After write-back of transfer information	
Setting	DTLOCK	CSSTP1	CSSTP2	CSSTP3	DTBST	After vector read	NOP cycle generation※1	After transfer information read	After a single data transfer	Normal transfer	Continuous transfer
Setting 1	1	0	※3	1	0	O	O	O	O	O	O
Setting 2	0	0	0	※3	0	x	O	x	x	O	O
Setting 3	0	1	※3	※3	0	x	x	x	x	O	O
Setting 4※2	0	1	※3	※3	1	x	x	x	x	O	x
Setting 5	1	1	※3	1	0	O	x	O	O	O	O

Notes: 1. The bus mastership is only released for the external space access request from the CPU after a vector read.

2. There are following restrictions in setting 4.
   - Clock setting by the frequency control register (FRQCR) must be Iϕ:Bϕ:Pϕ:MIϕ:MPϕ = 8:4:4:4:4, 4:2:2:2:2, or 2:1:1:1:1.
   - Locate vector information in on-chip ROM or on-chip RAM.
   - Locate transfer information in on-chip RAM.
   - Transfer is allowed between on-chip RAM and on-chip peripheral module or between external memory and on-chip peripheral module.

3. Don't care.

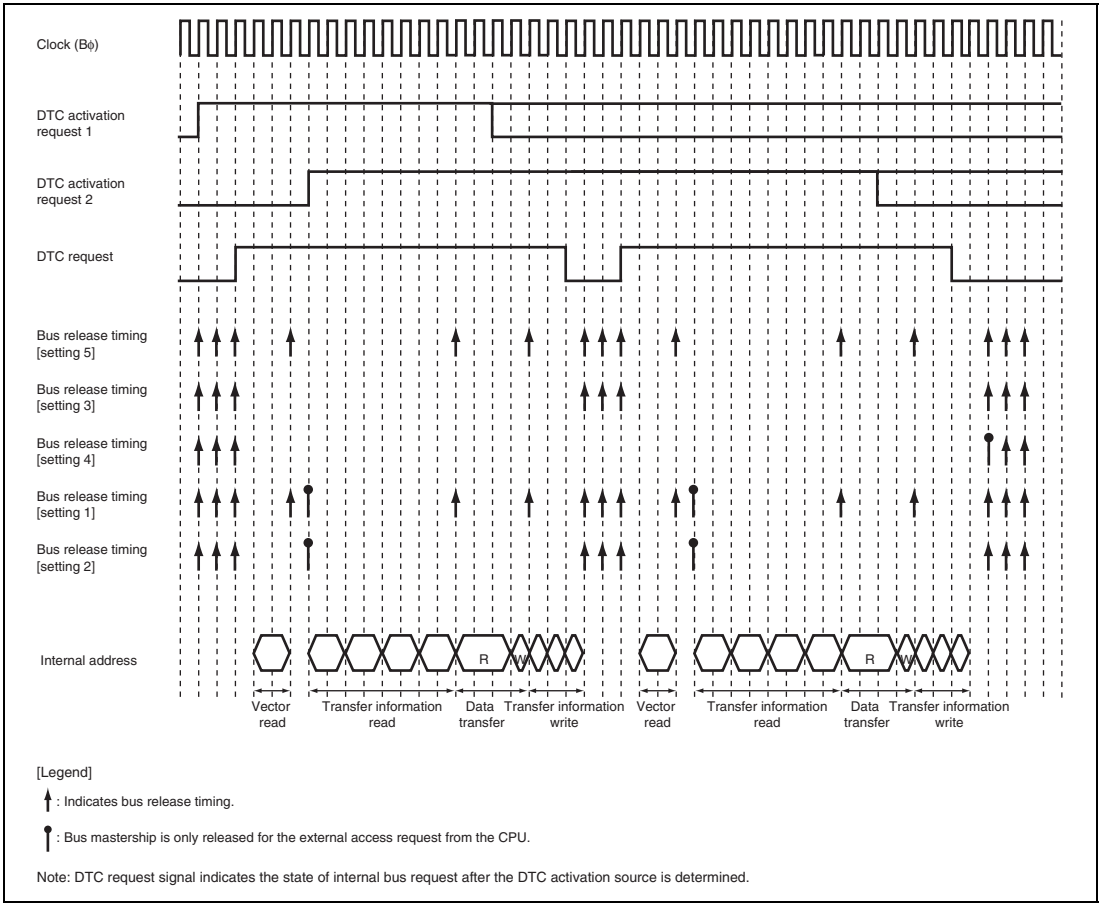


Figure 8.16 Example of DTC Operation Timing: Conflict of Two Activation Requests in Normal Transfer Mode (Activated by On-Chip Peripheral Module; Iϕ: Bϕ: Pϕ = 1: 1/2: 1/2; Data Transferred from On-Chip Peripheral Module to On-Chip RAM; Transfer Information Is Written in 3 Cycles)

8.5.10 DTC Activation Priority Order

In the case where multiple DTC activation requests are generated while the DTC is inactive, it is selectable whether the DTC starts transfer in the order of activation request generation or in the order of priority for DTC activation. This selection is made by the setting of the DTPR bit in the bus function extending register (BSCEHR). On the other hand, if multiple activation requests are generated while the DTC is active, transfer is performed according to the priority order for DTC activation. Figure 8.17 shows an example of DTC activation according to the priority.

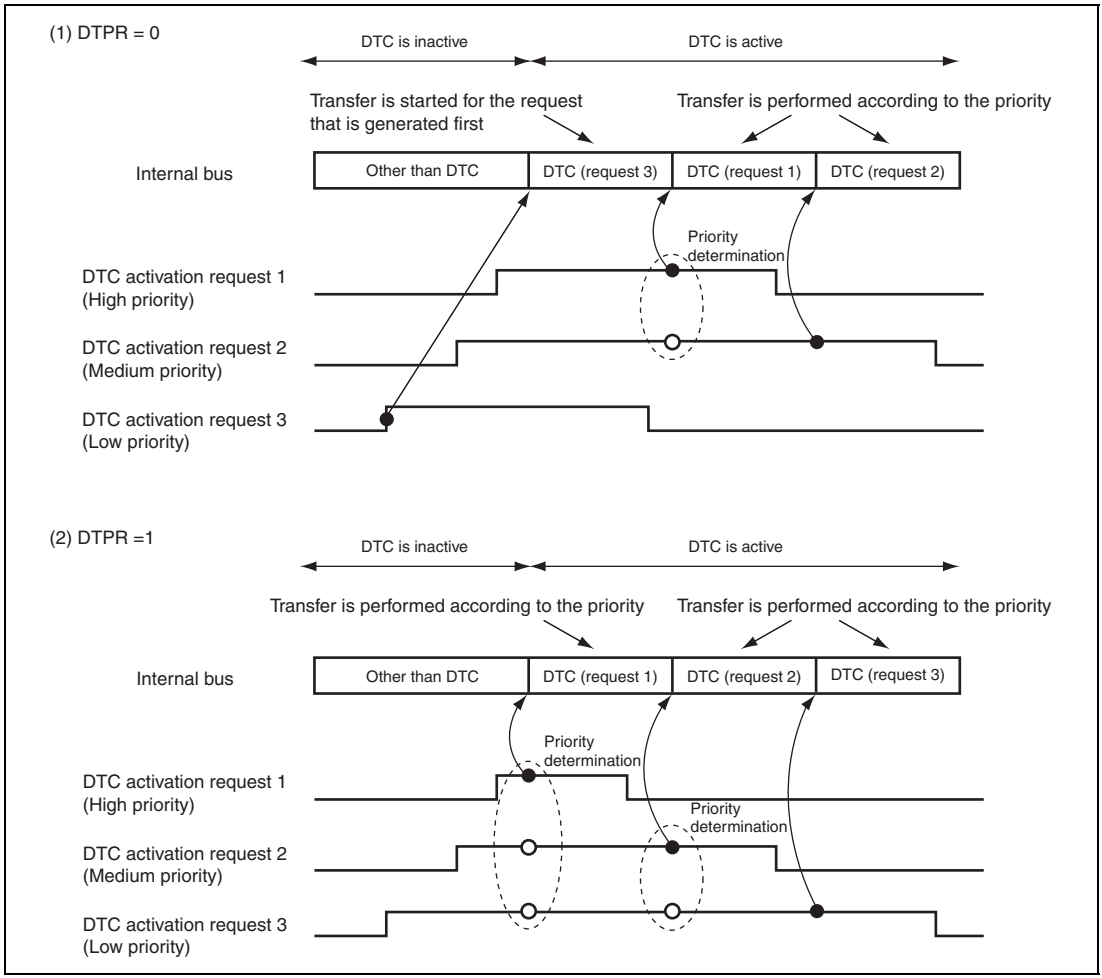
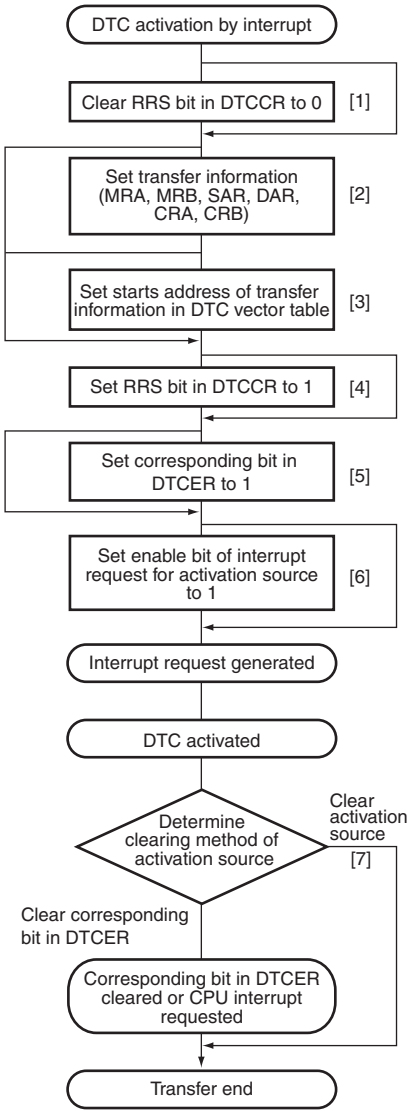


Figure 8.17 Example of DTC Activation in Accordance with Priority

8.6 DTC Activation by Interrupt

The procedure for using the DTC with interrupt activation is shown in figure 8.18.



[1] Clearing the RRS bit in DTCCR to 0 clears the read skip flag of transfer information. Read skip is not performed when the DTC is activated after clearing the RRS bit. When updating transfer information, the RRS bit must be cleared.

[2] Set the MRA, MRB, SAR, DAR, CRA, and CRB transfer information in the data area. For details on setting transfer information, see section 8.2, Register Descriptions. For details on location of transfer information, see section 8.4, Location of Transfer Information and DTC Vector Table.

[3] Set the start address of the transfer information in the DTC vector table. For details on setting DTC vector table, see section 8.4, Location of Transfer Information and DTC Vector Table.

[4] Setting the RRS bit to 1 performs a read skip of second time or later transfer information when the DTC is activated consecutively by the same interrupt source. Setting the RRS bit to 1 is always allowed. However, the value set during transfer will be valid from the next transfer.

[5] Set the bit in DTCER corresponding to the DTC activation interrupt source to 1. For the correspondence of interrupts and DTCER, refer to table 8.2. The bit in DTCER may be set to 1 on the second or later transfer. In this case, setting the bit is not needed.

[6] Set the enable bits for the interrupt sources to be used as the activation sources to 1. The DTC is activated when an interrupt used as an activation source is generated. For details on the settings of the interrupt enable bits, see the corresponding descriptions of the corresponding module.

[7] After the end of one data transfer, the DTC clears the activation source flag or clears the corresponding bit in DTCER and requests an interrupt to the CPU. The operation after transfer depends on the transfer information. For details, see section 8.2, Register Descriptions and figure 8.4.

Figure 8.18 Activation of DTC by Interrupt

8.7 Examples of Use of the DTC

8.7.1 Normal Transfer Mode

An example is shown in which the DTC is used to receive 128 bytes of data via the SCI.

1. Set MRA to fixed source address (SM1 = SM0 = 0), incrementing destination address (DM1 = 1, DM0 = 0), normal transfer mode (MD1 = MD0 = 0), and byte size (Sz1 = Sz0 = 0). The DTS bit can have any value. Set MRB for one data transfer by one interrupt (CHNE = 0, DISEL = 0). Set the RDR address of the SCI in SAR, the start address of the RAM area where the data will be received in DAR, and 128 (H'0080) in CRA. CRB can be set to any value.
2. Set the start address of the transfer information for an RXI interrupt at the DTC vector address.
3. Set the corresponding bit in DTCER to 1.
4. Set the SCI to the appropriate receive mode. Set the RIE bit in SCR to 1 to enable the receive end (RXI) interrupt. Since the generation of a receive error during the SCI reception operation will disable subsequent reception, the CPU should be enabled to accept receive error interrupts.
5. Each time reception of one byte of data ends on the SCI, the RDRF flag in SSR is set to 1, an RXI interrupt is generated, and the DTC is activated. The receive data is transferred from RDR to RAM by the DTC. DAR is incremented and CRA is decremented. The RDRF flag is automatically cleared to 0.
6. When CRA becomes 0 after the 128 data transfers have ended, the RDRF flag is held at 1, the DTCE bit is cleared to 0, and an RXI interrupt request is sent to the CPU. Termination processing should be performed in the interrupt handling routine.

8.7.2 Chain Transfer when Counter = 0

By executing a second data transfer and performing re-setting of the first data transfer only when the counter value is 0, it is possible to perform 256 or more repeat transfers.

An example is shown in which a 128-kbyte input buffer is configured. The input buffer is assumed to have been set to start at lower address H'0000. Figure 8.19 shows the chain transfer when the counter value is 0.

1. For the first transfer, set the normal transfer mode for input data. Set the fixed transfer source address, CRA = H'0000 (65,536 times), CHNE = 1, CHNS = 1, and DISEL = 0.
2. Prepare the upper 8-bit addresses of the start addresses for 65,536-transfer units for the first data transfer in a separate area (in ROM, etc.). For example, if the input buffer is configured at addresses H'200000 to H'21FFFF, prepare H'21 and H'20.

3. For the second transfer, set repeat transfer mode (with the source side as the repeat area) for re-setting the transfer destination address for the first data transfer. Use the upper eight bits of DAR in the first transfer information area as the transfer destination. Set CHNE = DISEL = 0. If the above input buffer is specified as H'200000 to H'21FFFF, set the transfer counter to 2.
4. Execute the first data transfer 65536 times by means of interrupts. When the transfer counter for the first data transfer reaches 0, the second data transfer is started. Set the upper eight bits of the transfer destination address for the first data transfer to H'21. The lower 16 bits of the transfer destination address of the first data transfer and the transfer counter are H'0000.
5. Next, execute the first data transfer the 65536 times specified for the first data transfer by means of interrupts. When the transfer counter for the first data transfer reaches 0, the second data transfer is started. Set the upper eight bits of the transfer destination address for the first data transfer to H'20. The lower 16 bits of the transfer destination address of the first data transfer and the transfer counter are H'0000.
6. Steps 4 and 5 are repeated endlessly. As repeat mode is specified for the second data transfer, no interrupt request is sent to the CPU.

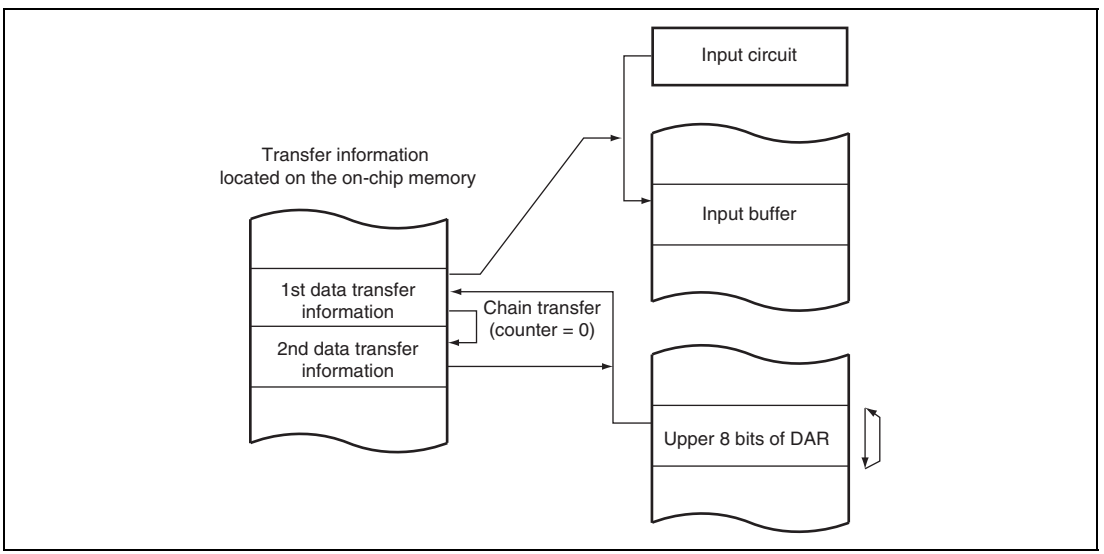


Figure 8.19 Chain Transfer when Counter = 0

8.8 Interrupt Sources

An interrupt request is issued to the CPU when the DTC finishes the specified number of data transfers, or on completion of a single data transfer or a single block data transfer with the DISEL bit set to 1. In the case of interrupt activation, the interrupt set as the activation source is generated. These interrupts to the CPU are subject to CPU mask level and priority level control in the interrupt controller. For details, refer to section 6.8, Data Transfer with Interrupt Request Signals.

8.9 Usage Notes

8.9.1 Module Standby Mode Setting

Operation of the DTC can be disabled or enabled using the standby control register. The initial setting is for operation of the DTC to be disabled. DTC operation is disabled in module standby mode but register access is available. Module standby mode cannot be set while the DTC is activated. Before entering software standby mode or module standby mode, all DTCER registers must be cleared. For details, refer to section 26, Power-Down Modes.

8.9.2 On-Chip RAM

Transfer information can be located in on-chip RAM. In this case, the RAME bit in RAMCR must not be cleared to 0.

8.9.3 DTCE Bit Setting

To set a DTCE bit, disable the corresponding interrupt, read 0 from the bit, and then write 1 to it. While DTC transfer is in progress, do not modify the DTCE bits.

8.9.4 Chain Transfer

When chain transfer is used, clearing of the activation source or DTCER is performed when the last of the chain of data transfers is executed. SCI, SCIF, SSU, IIC2, and A/D converter interrupt/activation sources, on the other hand, are cleared when the DTC reads or writes to the relevant register.

8.9.5 Transfer Information Start Address, Source Address, and Destination Address

The transfer information start address to be specified in the vector table should be address 4n. Transfer information should be placed in on-chip RAM or external memory space.

8.9.6 Access to DMAC or DTC Registers through DTC

Do not access the DMAC or DTC registers by using DTC operation. Do not access the DTC registers by using DMAC operation.

8.9.7 Notes on IRQ Interrupt as DTC Activation Source

- The IRQ interrupt specified as a DTC activation source must not be used to cancel software standby mode.
- The IRQ edge input in software standby mode must not be specified as a DTC activation source.
- When a low level on the IRQ pin is to be detected, if the end of DTC transfer is used to request an interrupt to the CPU (transfer counter = 0 or DISEL = 1), the IRQ signal must be kept low until the CPU accepts the interrupt.

8.9.8 Notes on SCI and SCIF as DTC Activation Sources

- When the TXI interrupt from the SCI is specified as a DTC activation source, the TEND flag in the SCI must not be used as the transfer end flag.
- When the TXIF interrupt from the SCIF is specified as a DTC activation source, the TEND flag in the SCIF must not be used as the transfer end flag.

8.9.9 Clearing Interrupt Source Flag

The interrupt source flag set when the DTC transfer is completed should be cleared in the interrupt handler in the same way as for general interrupt source flags. For details, refer to section 6.9, Usage Note.

8.9.10 Conflict between NMI Interrupt and DTC Activation

When a conflict occurs between the generation of the NMI interrupt and the DTC activation, the NMI interrupt has priority. Thus the ERR bit is set to 1 and the DTC is not activated.

It takes 1 × Bcyc + 3 × Pcyc for determining DTC stop by NMI, 2 × Bcyc for determining DTC activation by IRQ, and 1 × Pcyc for determining DTC activation by peripheral modules.

8.9.11 Operation when a DTC Activation Request Is Cancelled While in Progress

Once the DTC has accepted an activation request, the DTC does not accept the next activation request until the sequence of DTC processing that ends with writeback has been completed.

Section 9 Bus State Controller (BSC)

The bus state controller (BSC) outputs control signals for various types of memory that is connected to the external address space and external devices. BSC functions enable this LSI to connect directly with SRAM, SDRAM, and other memory storage devices, and external devices.

9.1 Features

1. External address space
   - A maximum 64 Mbytes for each of eight areas, CS0 to CS7, and a maximum 1 Gbyte for the CS8 area
   - Can specify the normal space interface, SRAM interface with byte selection, burst ROM (clock synchronous or asynchronous), MPX-I/O, burst MPX-I/O, SDRAM, or PCMCIA for each address space
   - Can select the data bus width (8, 16, or 32 bits) for each address space
   - Controls the insertion of the wait state for each address space.
   - Controls the insertion of the wait state for each read access and write access
   - Can set the independent idling cycle in the continuous access for five cases: read-write (in same space/different space), read-read (in same space/different space), the first cycle is a write access.
2. Normal space interface
   - Supports the interface that can directly connect to the SRAM
3. Burst ROM interface (clock asynchronous)
   - High-speed access to the ROM that has the page mode function
4. MPX-I/O interface
   - Directly connects peripheral LSIs with address/data multiplexing
5. SDRAM interface
   - Can set the SDRAM up to 2 areas
   - Multiplex output for row address/column address
   - Efficient access by single read/single write
   - High-speed access by bank-active mode
   - Supports an auto-refresh and self-refresh
6. SRAM interface with byte selection
   - Supports interfaces that can be connected directly to SRAM with byte selection

7. PCMCIA direct interface
   - Supports the IC memory card and I/O card interface defined in JEIDA specifications Ver. 4.2 (PCMCIA2.1 Rev. 2.1)
   - Wait-cycle insertion controllable by program
8. Burst MPX-I/O interface
   - Directly connects peripheral LSIs with address/data multiplexing
   - Supports burst transfer
9. Burst ROM (clock synchronous) interface
   - Directly connects clock-synchronous burst ROM
10. Refresh function
   - Supports the auto-refresh and self-refresh functions
   - Specifies the refresh interval using the refresh counter and clock selection
   - Can execute concentrated refresh by specifying the refresh counts (1, 2, 4, 6, or 8)
11. Usage as interval timer for refresh counter
   - Generates an interrupt request at compare match

Figure 9.1 shows a block diagram of the BSC.

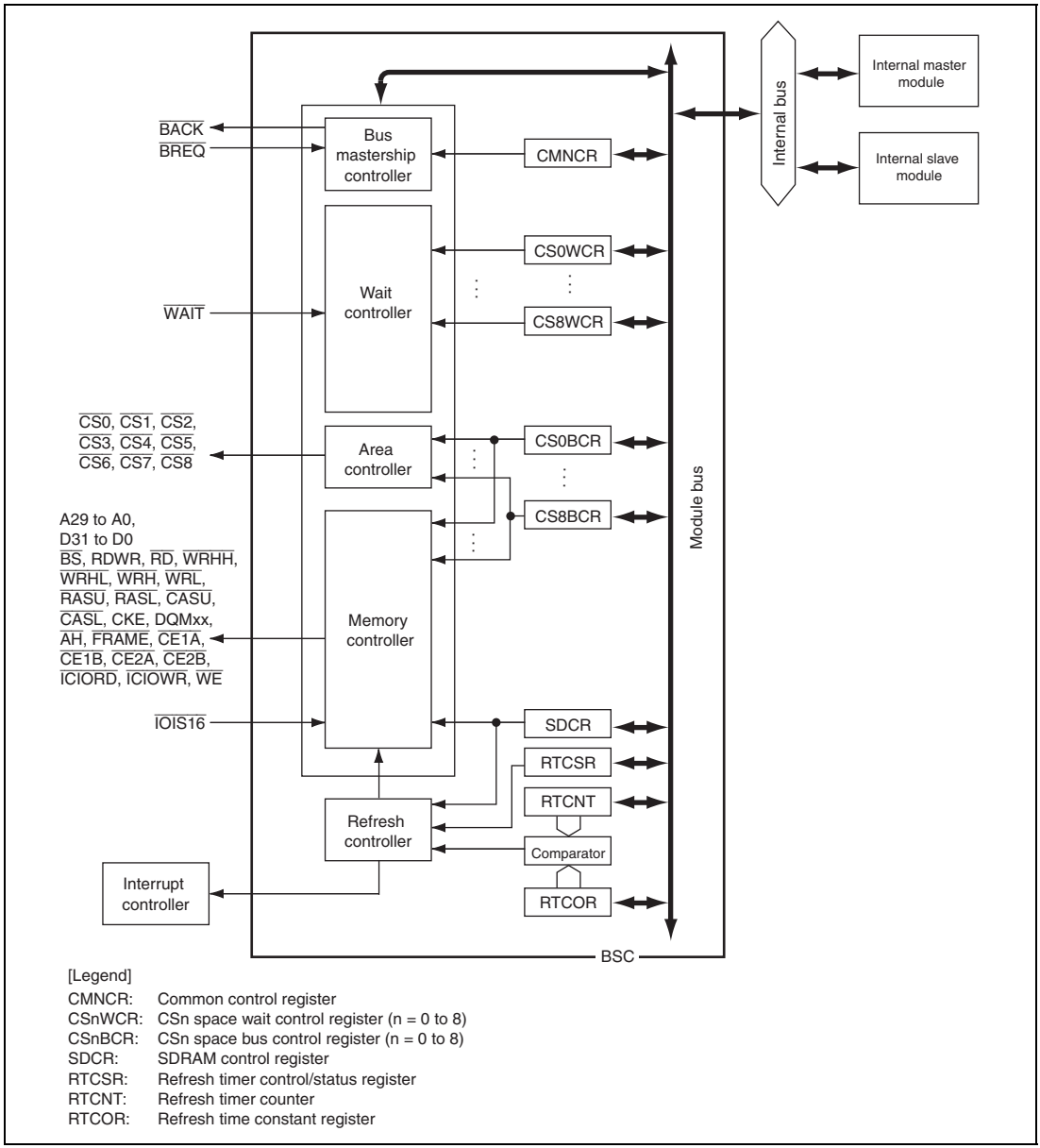


Figure 9.1 Block Diagram of BSC

9.2 Input/Output Pins

The pin configuration of the BSC is listed in table 9.1.

Table 9.1 Pin Configuration

Name	I/O	Function
A29 to A0	Output	Address bus
D31 to D0	I/O	Data bus
$\overline{BS}$	Output	Bus cycle start Asserted when a normal space, burst ROM (clock synchronous or asynchronous), MPX-I/O, burst MPX-I/O, or PCMCIA is accessed. When SDRAM is accessed, this signal is asserted at the same timing as $\overline{CAS}$.
$\overline{CS0}$ to $\overline{CS8}$	Output	Chip select
$\overline{CE1A}$	Output	Chip enable for PCMCIA connected to area 5
$\overline{CE2A}$	Output	Chip enable for PCMCIA connected to area 5
$\overline{CE1B}$	Output	Chip enable for PCMCIA connected to area 6
$\overline{CE2B}$	Output	Chip enable for PCMCIA connected to area 6
RDWR	Output	Read/write Connected to $\overline{WE}$ pin when SDRAM or SRAM with byte selection is used.
$\overline{RD}$	Output	Read pulse signal (read data output enable signal) Strobe signal for indicating memory read cycles when PCMCIA is used.
$\overline{WRHH}$	Output	Indicates byte write through D31 to D24. Connected to the byte select pin when SRAM with byte selection is used.
$\overline{WRHL}$	Output	Indicates byte write through D23 to D16. Connected to the byte select pin when SRAM with byte selection is used.
$\overline{WRH}$	Output	Indicates byte write through D15 to D8. Connected to the byte select pin when SRAM with byte selection is used.
$\overline{WRL}$	Output	Indicates byte write through D7 to D0. Connected to the byte select pin when SRAM with byte selection is used.

Name	I/O	Function
$\overline{\text{RASU}}$, $\overline{\text{RASL}}$	Output	Connected to $\overline{\text{RAS}}$ pin when SDRAM is used.
$\overline{\text{CASU}}$, $\overline{\text{CASL}}$	Output	Connected to $\overline{\text{CAS}}$ pin when SDRAM is used.
CKE	Output	Connected to CKE pin when SDRAM is used.
$\overline{\text{IOIS16}}$	Input	Indicates 16-bit I/O for PCMCIA. This LSI does not support little endian and this pin must be held low.
DQMUU DQMUL DQMLU DQMLL	Output	Connected to the DQMxx pins when SDRAM is used. DQMUU: D31 to D24 select signal DQMUL: D23 to D16 select signal DQMLU: D15 to D8 select signal DQMLL: D7 to D0 select signal
$\overline{\text{AH}}$	Output	Address hold signal when MPX-I/O is used
$\overline{\text{FRAME}}$	Output	FRAME signal when burst MPX-I/O is used
$\overline{\text{WAIT}}$	Input	External wait input
$\overline{\text{BREQ}}$	Input	Bus request input
$\overline{\text{BACK}}$	Output	Bus acknowledge output
$\overline{\text{ICIOWR}}$	Output	I/O write strobe signal when PCMCIA is used
$\overline{\text{ICIORD}}$	Output	I/O read strobe signal when PCMCIA is used
$\overline{\text{WE}}$	Output	Strobe signal for indicating memory write cycles when PCMCIA is used

9.3 Area Overview

9.3.1 Area Division

In the architecture, this LSI has 32-bit address spaces.

As listed in tables 9.2 to 9.15, this LSI can connect nine areas to each type of memory, and it outputs chip select signals ($\overline{CS0}$ to $\overline{CS8}$) for each of them. $\overline{CS0}$ is asserted during area 0 access. In access to SDRAM connected to areas 2 and 3, signals such as $\overline{RASx}$, $\overline{CASx}$, RD/$\overline{WR}$, and DQMxx will be asserted. Furthermore, when the PCMCIA interface is selected in areas 5 and 6, $\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, and $\overline{CE2B}$ as well as $\overline{CS5}$ and $\overline{CS6}$ are asserted, according to the bytes to be accessed.

9.3.2 Address Map

The external address space has a capacity of 1.5 Gbytes and is used by dividing into 9 spaces. The memory to be connected and the data bus width are specified in each space. The address map for the entire address space is listed in tables 9.2 to 9.15.

Table 9.2 Address Map: SH7083 (256-Kbyte Flash Memory Version) in On-Chip ROM-Enabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'0003FFFF	On-chip ROM		256 Kbytes	32 bits
H'00040000 to H'01FFFFFF	Reserved			
H'02000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	32 Mbytes	8 or 16 bits*
H'04000000 to H'0BFFFFFF	Reserved			
H'0C000000 to H'0DFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	32 Mbytes	8 or 16 bits*

Address	Area	Memory Type	Capacity	Bus Width
H'0E000000 to H'1BFFFFFF	Reserved			
H'1C000000 to H'1DFFFFFF	CS7 space	Normal space SRAM with byte selection	32 Mbytes	8 or 16 bits*
H'1E000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF7FFF	Reserved			
H'FFFF8000 to H'FFFFBFFF	On-chip RAM		16 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed. In single-chip mode, only the on-chip ROM, on-chip RAM, and on-chip peripheral modules can be accessed; the other areas cannot be accessed.

* The bus width is selected by the register setting.

Table 9.3 Address Map: SH7083 (256-Kbyte Flash Memory Version) in On-Chip ROM-Disabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'01FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	32 Mbytes	8 or 16 bits*[1]
H'02000000 to H'0BFFFFFF	Reserved			
H'0C000000 to H'0DFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	32 Mbytes	8 or 16 bits*[2]
H'0E000000 to H'1BFFFFFF	Reserved			

Address	Area	Memory Type	Capacity	Bus Width
H'1C000000 to H'1DFFFFFF	CS7 space	Normal space SRAM with byte selection	32 Mbytes	8 or 16 bits*[2]
H'1E000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF7FFF	Reserved			
H'FFFF8000 to H'FFFFBFFF	On-chip RAM		16 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed.

1. The bus width is selected by the mode pins.
2. The bus width is selected by the register setting.

Table 9.4 Address Map: SH7083 (512-Kbyte Flash Memory Version) in On-Chip ROM-Enabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'0007FFFF	On-chip ROM		512 Kbytes	32 bits
H'00080000 to H'01FFFFFF	Reserved			
H'02000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	32 Mbytes	8 or 16 bits*
H'04000000 to H'0BFFFFFF	Reserved			
H'0C000000 to H'0DFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	32 Mbytes	8 or 16 bits*

Address	Area	Memory Type	Capacity	Bus Width
H'0E000000 to H'1BFFFFFF	Reserved			
H'1C000000 to H'1DFFFFFF	CS7 space	Normal space SRAM with byte selection	32 Mbytes	8 or 16 bits*
H'1E000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF3FFF	Reserved			
H'FFFF4000 to H'FFFFBFFF	On-chip RAM		32 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed. In single-chip mode, only the on-chip ROM, on-chip RAM, and on-chip peripheral modules can be accessed; the other areas cannot be accessed.

* The bus width is selected by the register setting.

Table 9.5 Address Map: SH7083 (512-Kbyte Flash Memory Version) in On-Chip ROM-Disabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'01FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	32 Mbytes	8 or 16 bits*[1]
H'02000000 to H'0BFFFFFF	Reserved			
H'0C000000 to H'0DFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	32 Mbytes	8 or 16 bits*[2]
H'0E000000 to H'1BFFFFFF	Reserved			

Address	Area	Memory Type	Capacity	Bus Width
H'1C000000 to H'1DFFFFFF	CS7 space	Normal space SRAM with byte selection	32 Mbytes	8 or 16 bits*[2]
H'1E000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF3FFF	Reserved			
H'FFFF4000 to H'FFFFBFFF	On-chip RAM		32 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed.

1. The bus width is selected by the mode pins.
2. The bus width is selected by the register setting.

Table 9.6 Address Map: SH7084 (256-Kbyte Flash Memory Version) in On-Chip ROM-Enabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'0003FFFF	On-chip ROM		256 Kbytes	32 bits
H'00040000 to H'01FFFFFF	Reserved			
H'02000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	32 Mbytes	8 or 16 bits*
H'04000000 to H'07FFFFFF	CS1 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*
H'08000000 to H'0BFFFFFF	CS2 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8 or 16 bits*

Address	Area	Memory Type	Capacity	Bus Width
H'0C000000 to H'0FFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8 or 16 bits*
H'10000000 to H'13FFFFFF	CS4 space	Normal space SRAM with byte selection Burst ROM (asynchronous)	64 Mbytes	8 or 16 bits*
H'14000000 to H'17FFFFFF	CS5 space	Normal space SRAM with byte selection MPX-I/O	64 Mbytes	8 or 16 bits*
H'18000000 to H'1BFFFFFF	CS6 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*
H'1C000000 to H'1FFFFFFF	CS7 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*
H'20000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF7FFF	Reserved			
H'FFFF8000 to H'FFFFBFFF	On-chip RAM		16 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed. In single-chip mode, only the on-chip ROM, on-chip RAM, and on-chip peripheral modules can be accessed; the other areas cannot be accessed.

* The bus width is selected by the register setting.

Table 9.7 Address Map: SH7084 (256-Kbyte Flash Memory Version) in On-Chip ROM-Disabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	64 Mbytes	8 or 16 bits*[1]
H'04000000 to H'07FFFFFF	CS1 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*[2]
H'08000000 to H'0BFFFFFF	CS2 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8 or 16 bits*[2]
H'0C000000 to H'0FFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8 or 16 bits*[2]
H'10000000 to H'13FFFFFF	CS4 space	Normal space SRAM with byte selection Burst ROM (asynchronous)	64 Mbytes	8 or 16 bits*[2]
H'14000000 to H'17FFFFFF	CS5 space	Normal space SRAM with byte selection MPX-I/O	64 Mbytes	8 or 16 bits*[2]
H'18000000 to H'1BFFFFFF	CS6 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*[2]
H'1C000000 to H'1FFFFFFF	CS7 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*[2]
H'20000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF7FFF	Reserved			

Address	Area	Memory Type	Capacity	Bus Width
H'FFFF8000 to H'FFFFBFFF	On-chip RAM		16 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed.

1. The bus width is selected by the mode pins.
2. The bus width is selected by the register setting.

Table 9.8 Address Map: SH7084 (512-Kbyte Flash Memory Version) in On-Chip ROM-Enabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'0007FFFF	On-chip ROM		512 Kbytes	32 bits
H'00080000 to H'01FFFFFF	Reserved			
H'02000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	32 Mbytes	8 or 16 bits*
H'04000000 to H'07FFFFFF	CS1 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*
H'08000000 to H'0BFFFFFF	CS2 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8 or 16 bits*
H'0C000000 to H'0FFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8 or 16 bits*
H'10000000 to H'13FFFFFF	CS4 space	Normal space SRAM with byte selection Burst ROM (asynchronous)	64 Mbytes	8 or 16 bits*

Address	Area	Memory Type	Capacity	Bus Width
H'14000000 to H'17FFFFFF	CS5 space	Normal space SRAM with byte selection MPX-I/O	64 Mbytes	8 or 16 bits*
H'18000000 to H'1BFFFFFF	CS6 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*
H'1C000000 to H'1FFFFFFF	CS7 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*
H'20000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF3FFF	Reserved			
H'FFFF4000 to H'FFFFBFFF	On-chip RAM		32 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed. In single-chip mode, only the on-chip ROM, on-chip RAM, and on-chip peripheral modules can be accessed; the other areas cannot be accessed.

* The bus width is selected by the register setting.

Table 9.9 Address Map: SH7084 (512-Kbyte Flash Memory Version) in On-Chip ROM-Disabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	64 Mbytes	8 or 16 bits*[1]
H'04000000 to H'07FFFFFF	CS1 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*[2]

Address	Area	Memory Type	Capacity	Bus Width
H'08000000 to H'0BFFFFFF	CS2 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8 or 16 bits*2
H'0C000000 to H'0FFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8 or 16 bits*2
H'10000000 to H'13FFFFFF	CS4 space	Normal space SRAM with byte selection Burst ROM (asynchronous)	64 Mbytes	8 or 16 bits*2
H'14000000 to H'17FFFFFF	CS5 space	Normal space SRAM with byte selection MPX-I/O	64 Mbytes	8 or 16 bits*2
H'18000000 to H'1BFFFFFF	CS6 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*2
H'1C000000 to H'1FFFFFFF	CS7 space	Normal space SRAM with byte selection	64 Mbytes	8 or 16 bits*2
H'20000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF3FFF	Reserved			
H'FFFF4000 to H'FFFFBFFF	On-chip RAM		32 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed.

1. The bus width is selected by the mode pins.
2. The bus width is selected by the register setting.

Table 9.10 Address Map: SH7085 (256-Kbyte Flash Memory Version) in On-Chip ROM-Enabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'0003FFFF	On-chip ROM		256 Kbytes	32 bits
H'00040000 to H'01FFFFFF	Reserved			
H'02000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	32 Mbytes	8, 16, or 32 bits*
H'04000000 to H'07FFFFFF	CS1 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*
H'08000000 to H'0BFFFFFF	CS2 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*
H'0C000000 to H'0FFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*
H'10000000 to H'13FFFFFF	CS4 space	Normal space SRAM with byte selection Burst ROM (asynchronous)	64 Mbytes	8, 16, or 32 bits*
H'14000000 to H'17FFFFFF	CS5 space	Normal space SRAM with byte selection PCMCIA MPX-I/O	64 Mbytes	8, 16, or 32 bits*
H'18000000 to H'1BFFFFFF	CS6 space	Normal space SRAM with byte selection PCMCIA Burst MPX-I/O	64 Mbytes	8, 16, or 32 bits*
H'1C000000 to H'1FFFFFFF	CS7 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*
H'20000000 to H'FFF7FFFF	Reserved			

Address	Area	Memory Type	Capacity	Bus Width
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF7FFF	Reserved			
H'FFFF8000 to H'FFFFBFFF	On-chip RAM		16 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed. In single-chip mode, only the on-chip ROM, on-chip RAM, and on-chip peripheral modules can be accessed; the other areas cannot be accessed.

* The bus width is selected by the register setting.

Table 9.11 Address Map: SH7085 (256-Kbyte Flash Memory Version) in On-Chip ROM-Disabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	64 Mbytes	16 or 32 bits*[1]
H'04000000 to H'07FFFFFF	CS1 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*[2]
H'08000000 to H'0BFFFFFF	CS2 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*[2]
H'0C000000 to H'0FFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*[2]
H'10000000 to H'13FFFFFF	CS4 space	Normal space SRAM with byte selection Burst ROM (asynchronous)	64 Mbytes	8, 16, or 32 bits*[2]

Address	Area	Memory Type	Capacity	Bus Width
H'14000000 to H'17FFFFFF	CS5 space	Normal space SRAM with byte selection PCMCIA MPX-I/O	64 Mbytes	8, 16, or 32 bits*2
H'18000000 to H'1BFFFFFF	CS6 space	Normal space SRAM with byte selection PCMCIA Burst MPX-I/O	64 Mbytes	8, 16, or 32 bits*2
H'1C000000 to H'1FFFFFFF	CS7 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*2
H'20000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF7FFF	Reserved			
H'FFFF8000 to H'FFFFBFFF	On-chip RAM		16 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed.

1. The bus width is selected by the mode pins.
2. The bus width is selected by the register setting.

Table 9.12 Address Map: SH7085 (512-Kbyte Flash Memory Version) in On-Chip ROM-Enabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'0007FFFF	On-chip ROM		512 Kbytes	32 bits
H'00080000 to H'01FFFFFF	Reserved			

Address	Area	Memory Type	Capacity	Bus Width
H'02000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	32 Mbytes	8, 16, or 32 bits*
H'04000000 to H'07FFFFFF	CS1 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*
H'08000000 to H'0BFFFFFF	CS2 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*
H'0C000000 to H'0FFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*
H'10000000 to H'13FFFFFF	CS4 space	Normal space SRAM with byte selection Burst ROM (asynchronous)	64 Mbytes	8, 16, or 32 bits*
H'14000000 to H'17FFFFFF	CS5 space	Normal space SRAM with byte selection PCMCIA MPX-I/O	64 Mbytes	8, 16, or 32 bits*
H'18000000 to H'1BFFFFFF	CS6 space	Normal space SRAM with byte selection PCMCIA Burst MPX-I/O	64 Mbytes	8, 16, or 32 bits*
H'1C000000 to H'1FFFFFFF	CS7 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*
H'20000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF3FFF	Reserved			

Address	Area	Memory Type	Capacity	Bus Width
H'FFFF4000 to H'FFFFBFFF	On-chip RAM		32 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed. In single-chip mode, only the on-chip ROM, on-chip RAM, and on-chip peripheral modules can be accessed; the other areas cannot be accessed.

* The bus width is selected by the register setting.

Table 9.13 Address Map: SH7085 (512-Kbyte Flash Memory Version) in On-Chip ROM-Disabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	64 Mbytes	16 or 32 bits*1
H'04000000 to H'07FFFFFF	CS1 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*2
H'08000000 to H'0BFFFFFF	CS2 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*2
H'0C000000 to H'0FFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*2
H'10000000 to H'13FFFFFF	CS4 space	Normal space SRAM with byte selection Burst ROM (asynchronous)	64 Mbytes	8, 16, or 32 bits*2
H'14000000 to H'17FFFFFF	CS5 space	Normal space SRAM with byte selection PCMCIA MPX-I/O	64 Mbytes	8, 16, or 32 bits*2

Address	Area	Memory Type	Capacity	Bus Width
H'18000000 to H'1BFFFFFF	CS6 space	Normal space SRAM with byte selection PCMCIA Burst MPX-I/O	64 Mbytes	8, 16, or 32 bits*[2]
H'1C000000 to H'1FFFFFFF	CS7 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*[2]
H'20000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF3FFF	Reserved			
H'FFFF4000 to H'FFFFBFFF	On-chip RAM		32 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed.

1. The bus width is selected by the mode pins.
2. The bus width is selected by the register setting.

Table 9.14 Address Map: SH7086 in On-Chip ROM-Enabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'0007FFFF	On-chip ROM		512 Kbytes	32 bits
H'00080000 to H'01FFFFFF	Reserved			
H'02000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	32 Mbytes	8, 16, or 32 bits*
H'04000000 to H'07FFFFFF	CS1 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*

Address	Area	Memory Type	Capacity	Bus Width
H'08000000 to H'0BFFFFFF	CS2 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*
H'0C000000 to H'0FFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*
H'10000000 to H'13FFFFFF	CS4 space	Normal space SRAM with byte selection Burst ROM (asynchronous)	64 Mbytes	8, 16, or 32 bits*
H'14000000 to H'17FFFFFF	CS5 space	Normal space SRAM with byte selection PCMCIA MPX-I/O	64 Mbytes	8, 16, or 32 bits*
H'18000000 to H'1BFFFFFF	CS6 space	Normal space SRAM with byte selection PCMCIA Burst MPX-I/O	64 Mbytes	8, 16, or 32 bits*
H'1C000000 to H'1FFFFFFF	CS7 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*
H'20000000 to H'3FFFFFFF	Reserved			
H'40000000 to H'7FFFFFFF	CS8 space	Normal space SRAM with byte selection	1 Gbyte	8, 16, or 32 bits*
H'80000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF3FFF	Reserved			

Address	Area	Memory Type	Capacity	Bus Width
H'FFFF4000 to H'FFFFBFFF	On-chip RAM		32 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed. In single-chip mode, only the on-chip ROM, on-chip RAM, and on-chip peripheral modules can be accessed; the other areas cannot be accessed.

* The bus width is selected by the register setting.

Table 9.15 Address Map: SH7086 in On-Chip ROM-Disabled Mode

Address	Area	Memory Type	Capacity	Bus Width
H'00000000 to H'03FFFFFF	CS0 space	Normal space SRAM with byte selection Burst ROM (asynchronous) Burst ROM (synchronous)	64 Mbytes	16 or 32 bits*[1]
H'04000000 to H'07FFFFFF	CS1 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*[2]
H'08000000 to H'0BFFFFFF	CS2 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*[2]
H'0C000000 to H'0FFFFFFF	CS3 space	Normal space SRAM with byte selection SDRAM	64 Mbytes	8, 16, or 32 bits*[2]
H'10000000 to H'13FFFFFF	CS4 space	Normal space SRAM with byte selection Burst ROM (asynchronous)	64 Mbytes	8, 16, or 32 bits*[2]
H'14000000 to H'17FFFFFF	CS5 space	Normal space SRAM with byte selection PCMCIA MPX-I/O	64 Mbytes	8, 16, or 32 bits*[2]

Address	Area	Memory Type	Capacity	Bus Width
H'18000000 to H'1BFFFFFF	CS6 space	Normal space SRAM with byte selection PCMCIA Burst MPX-I/O	64 Mbytes	8, 16, or 32 bits*2
H'1C000000 to H'1FFFFFFF	CS7 space	Normal space SRAM with byte selection	64 Mbytes	8, 16, or 32 bits*2
H'20000000 to H'3FFFFFFF	Reserved			
H'40000000 to H'7FFFFFFF	CS8 space	Normal space SRAM with byte selection	1 Gbyte	8, 16, or 32 bits*2
H'80000000 to H'FFF7FFFF	Reserved			
H'FFF80000 to H'FFF9FFFF	SDRAM mode setting space			
H'FFFA0000 to H'FFFF3FFF	Reserved			
H'FFFF4000 to H'FFFFBFFF	On-chip RAM		32 Kbytes	32 bits
H'FFFFC000 to H'FFFFFFFF	On-chip peripheral modules		16 Kbytes	8 or 16 bits

Notes: Do not access the reserved area. If the reserved area is accessed, the correct operation cannot be guaranteed.

1. The bus width is selected by the mode pins.
2. The bus width is selected by the register setting.

9.4 Register Descriptions

The BSC has the following registers. Refer to section 27, List of Registers, for details on the register addresses and register states in each operating mode.

Do not access spaces other than CS0 until the termination of the memory interface setting.

Table 9.16 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Common control register	CMNCR	R/W	H'00001010	H'FFFFF000	32
CS0 space bus control register	CS0BCR	R/W	H'36DB0600	H'FFFFF004	32
CS1 space bus control register	CS1BCR	R/W	H'36DB0600	H'FFFFF008	32
CS2 space bus control register	CS2BCR	R/W	H'36DB0600	H'FFFFF00C	32
CS3 space bus control register	CS3BCR	R/W	H'36DB0600	H'FFFFF010	32
CS4 space bus control register	CS4BCR	R/W	H'36DB0600	H'FFFFF014	32
CS5 space bus control register	CS5BCR	R/W	H'36DB0600	H'FFFFF018	32
CS6 space bus control register	CS6BCR	R/W	H'36DB0600	H'FFFFF01C	32
CS7 space bus control register	CS7BCR	R/W	H'36DB0600	H'FFFFF020	32
CS8 space bus control register	CS8BCR	R/W	H'36DB0600	H'FFFFF024	32
CS0 space wait control register	CS0WCR	R/W	H'00000500	H'FFFFF028	32
CS1 space wait control register	CS1WCR	R/W	H'00000500	H'FFFFF02C	32
CS2 space wait control register	CS2WCR	R/W	H'00000500	H'FFFFF030	32
CS3 space wait control register	CS3WCR	R/W	H'00000500	H'FFFFF034	32
CS4 space wait control register	CS4WCR	R/W	H'00000500	H'FFFFF038	32
CS5 space wait control register	CS5WCR	R/W	H'00000500	H'FFFFF03C	32
CS6 space wait control register	CS6WCR	R/W	H'00000500	H'FFFFF040	32
CS7 space wait control register	CS7WCR	R/W	H'00000500	H'FFFFF044	32
CS8 space wait control register	CS8WCR	R/W	H'00000500	H'FFFFF048	32
SDRAM control register	SDCR	R/W	H'00000000	H'FFFFF04C	32
Refresh timer control/status register	RTCSR	R/W	H'00000000	H'FFFFF050	32
Refresh timer counter	RTCNT	R/W	H'00000000	H'FFFFF054	32
Refresh time constant register	RTCOR	R/W	H'00000000	H'FFFFF058	32
Bus function extending register	BSCEHR	R/W	H'0000	H'FFFFE89A	8,16

9.4.1 Common Control Register (CMNCR)

CMNCR is a 32-bit register that controls the common items for each area.

Do not access external memory other than area 0 until the register initialization is complete.

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	DMAIW[1:0]		DMAIWA	-	-	-	HIZMEM	HIZCNT
Initial value:	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
11 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7, 6	DMAIW[1:0]	00	R/W	Wait Specification between Access Cycles during DMA Single Address Transfer Specify the number of idle cycles to be inserted after data is output from an external device with DACK when DMA single address transfer is performed. The method of inserting idle cycles depends on the setting in the DMAIWA bit described later. 00: No idle cycle inserted 01: 1 idle cycle inserted 10: 2 idle cycles inserted 11: 4 idle cycled inserted

Bit	Bit Name	Initial Value	R/W	Description
5	DMAIWA	0	R/W	Specification for Method of Inserting Wait States between Access Cycles during DMA Single Address Transfer Specifies the method of inserting the idle cycles specified by the DMAIW1 and DMAIW0 bits. Clearing this bit will make this LSI insert the idle cycles when another device, which includes this LSI, drives the data bus after an external device with DACK drove it. When the external device with DACK drives the data bus continuously, idle cycles are not inserted. Setting this bit will make this LSI insert the idle cycles after one access is completed even when the continuous accesses to an external device with DACK are performed. 0: Idle cycles are inserted when another device drives data bus after external device with DACK drives data bus 1: Idle cycles are always inserted after external device with DACK is accessed.
4	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	HIZMEM	0	R/W	Hi-Z Memory Control Specifies the pin state in software standby mode for A29 to A0, $\overline{BS}$, $\overline{CSn}$, RDWR, $\overline{WRxx}$, $\overline{RD}$, $\overline{AH}$, $\overline{FRAME}$, $\overline{ICIORD}$, $\overline{ICIOWR}$, $\overline{WE}$, $\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, and $\overline{CE2B}$. While the bus is released, these pins are in high-impedance state regardless of this bit setting. 0: High impedance in software standby mode 1: Driven in software standby mode

Bit	Bit Name	Initial Value	R/W	Description
0	HIZCNT	0	R/W	Hi-Z Control Specifies the state in software standby mode and when bus mastership is released for CKE, $\overline{\text{RASU}}$, $\overline{\text{RASL}}$, $\overline{\text{CASU}}$, and $\overline{\text{CASL}}$. 0: High impedance in software standby mode and when bus mastership is released for CKE, $\overline{\text{RASU}}$, $\overline{\text{RASL}}$, $\overline{\text{CASU}}$, and $\overline{\text{CASL}}$. 1: Driven in software standby mode and when bus mastership is released for CKE, $\overline{\text{RASU}}$, $\overline{\text{RASL}}$, $\overline{\text{CASU}}$, and $\overline{\text{CASL}}$.

9.4.2 CSn Space Bus Control Register (CSnBCR) (n = 0 to 8)

CSnBCR is a 32-bit readable/writable register that specifies the type of memory connected to the respective space, the data bus width of the space, and the number of wait cycles between access cycles.

Do not access external memory other than area 0 until the register initialization is complete.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	IWW[1:0]		-	IWRWD[1:0]		-	IWRWS[1:0]		-	IWRRD[1:0]		-	IWRRS[1:0]	
Initial value:	0	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1
R/W:	R	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	TYPE[2:0]			-	BSZ[1:0]		-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	1*	1*	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R	R

Note: * When the on-chip ROM is disabled, CS0BCR samples the value input through the MD0 and MD1 external pins that specify the bus width when a power-on reset is performed.

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
29, 28	IWW[1:0]	11	R/W	Specification for Idle Cycles between Write-Read/Write-Write Cycles Specify the number of idle cycles to be inserted after access to memory that is connected to the space. The target cycles are write-read cycles and write-write cycles. 00: No idle cycle inserted 01: 1 idle cycle inserted 10: 2 idle cycles inserted 11: 4 idle cycles inserted
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26, 25	IWRWD[1:0]	11	R/W	Specification for Idle Cycles between Read-Write Cycles in Different Spaces Specify the number of idle cycles to be inserted after access to memory that is connected to the space. The target cycles are continuous read-write cycles in different spaces. 00: No idle cycle inserted 01: 1 idle cycle inserted 10: 2 idle cycles inserted 11: 4 idle cycles inserted
24	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
23, 22	IWRWS[1:0]	11	R/W	Specification for Idle Cycles between Read-Write Cycles in the Same Space Specify the number of idle cycles to be inserted after access to memory that is connected to the space. The target cycles are continuous read-write cycles in the same space. 00: No idle cycle inserted 01: 1 idle cycle inserted 10: 2 idle cycles inserted 11: 4 idle cycles inserted

Bit	Bit Name	Initial Value	R/W	Description
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20, 19	IWRRD[1:0]	11	R/W	Specification for Idle Cycles between Read-Read Cycles in Different Spaces Specify the number of idle cycles to be inserted after access to memory that is connected to the space. The target cycles are continuous read-read cycles in different spaces. 00: No idle cycle inserted 01: 1 idle cycle inserted 10: 2 idle cycles inserted 11: 4 idle cycles inserted
18	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
17, 16	IWRRS[1:0]	11	R/W	Specification for Idle Cycles between Read-Read Cycles in the Same Space Specify the number of idle cycles to be inserted after access to memory that is connected to the space. The target cycles are continuous read-read cycles in the same space. 00: No idle cycle inserted 01: 1 idle cycle inserted 10: 2 idle cycles inserted 11: 4 idle cycles inserted
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14 to 12	TYPE[2:0]	000	R/W	Memory Type Specification Specify the type of memory connected to the space. 000: Normal space 001: Burst ROM (clock asynchronous) 010: MPX-I/O 011: SRAM with byte selection 100: SDRAM 101: PCMCIA 110: Burst MPX-I/O 111: Burst ROM (clock synchronous) For the memory type for each area, see tables 9.2 to 9.15. Notes: 1. When burst MPX-I/O is selected for area 6, do not set areas 2 and 3 for SDRAM space. 2. SDRAM can be selected only for areas 2 and 3. If the SDRAM is only to be connected in one area, select area 3 as the SDRAM space. In this case, specify area 2 as normal space or SRAM with byte selection.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10, 9	BSZ[1:0]	11*	R/W	Data Bus Size Specification Specify the data bus sizes of spaces. 00: Setting prohibited 01: 8-bit size 10: 16-bit size 11: 32-bit size Bus width determined by the address when MPX-I/O is used Notes: 1. When MPX-I/O is selected for area 5, setting these bits to 11 enables the bus width (8 bits or 16 bits) to be determined by the address according to the SZSEL bit setting in CS5WCR. 2. When the on-chip ROM is disabled, the data bus width in area 0 is specified through external input pins. The BSZ1 and BSZ0 bit setting in CS0BCR is ignored. 3. When burst MPX-I/O is selected for area 6, only 32-bit size can be selected for the bus width. 4. When PCMCIA is selected for area 5 or 6, 8-bit or 16-bit size can be selected for the bus width. 5. When SDRAM is selected for area 2 or 3, 16-bit or 32-bit size can be selected for the bus width.
8 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: * When the on-chip ROM is disabled, CS0BCR samples the value input through the MD0 and MD1 external pins that specify the bus width when a power-on reset is performed.

9.4.3 CSn Space Wait Control Register (CSnWCR) (n = 0 to 8)

CSnWCR specifies various wait cycles for memory accesses. The bit configuration of this register varies as shown below according to the memory type (TYPE 2, TYPE 1, or TYPE 0) specified by the CSn space bus control register (CSnBCR). Specify CSnWCR before accessing the target area. CSnWCR should be modified only after CSnBCR setting is completed.

(1) Normal Space, SRAM with Byte Selection

- CS0WCR, CS1WCR, CS2WCR, CS3WCR, CS4WCR, CS5WCR, CS6WCR, CS7WCR, CS8WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	BAS	-	WW[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]		WR[3:0]				WM	-	-	-	-	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	BAS	0	R/W	Byte Access Selection when SRAM with Byte Selection is Used Specifies the $\overline{\text{WRxx}}$ and RDWR signal timing when SRAM interface with byte selection is used. 0: Asserts the $\overline{\text{WRxx}}$ signal at the read/write timing and asserts the RDWR signal during the write access cycle. 1: Asserts the $\overline{\text{WRxx}}$ signal during the read/write access cycle and asserts the RDWR signal at the write timing.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
18 to 16	WW[2:0]	000	R/W	Number of Wait Cycles in Write Access Specify the number of cycles required for write access. 000: The same cycles as WR3 to WR0 settings (read access wait) 001: 0 cycles 010: 1 cycle 011: 2 cycles 100: 3 cycles 101: 4 cycles 110: 5 cycles 111: 6 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address and $\overline{CSn}$ Assertion to $\overline{RD}$ and $\overline{WRxx}$ Assertion Specify the number of delay cycles from address and $\overline{CSn}$ assertion to $\overline{RD}$ and $\overline{WRxx}$ assertion. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	WR[3:0]	1010	R/W	Number of Read Access Wait Cycles Specify the number of wait cycles required for read access. 0000: 0 cycles 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	HW[1:0]	00	R/W	Delay Cycles from $\overline{RD}$ and $\overline{WRxx}$ Negation to Address and $\overline{CSn}$ Negation Specify the number of delay cycles from $\overline{RD}$ and $\overline{WRxx}$ negation to address and $\overline{CSn}$ negation. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

(2) MPX-I/O

- CS5WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	SZSEL	MPXW	-	WW[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]		WR[3:0]				WM	-	-	-	-	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
21	SZSEL	0	R/W	MPX-I/O Interface Bus Width Specification Specifies the address bit to select the bus width when the BSZ1 and BSZ0 bits in CS5BCR are set to 11. This setting is valid only when MPX-I/O is selected for area 5. 0: Address A14 selects the bus width 1: Address A21 selects the bus width The following shows bus width selection through the SZSEL bit and A14 or A21.

SZSEL	A14	A21	Bus Width
0	0	No effect	8 bits
0	1	No effect	16 bits
1	No effect	0	8 bits
1	No effect	1	16 bits

Bit	Bit Name	Initial Value	R/W	Description
20	MPXW	0	R/W	MPX-I/O Interface Address Wait This setting is valid only when MPX-I/O is selected for area 5. This bit specifies insertion of a wait cycle into the address cycle in MPX-I/O interface. 0: No wait 1: Inserts one wait cycle
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
18 to 16	WW[2:0]	000	R/W	Number of Wait Cycles in Write Access Specify the number of cycles required for write access. 000: The same cycles as WR3 to WR0 settings (read access wait) 001: 0 cycles 010: 1 cycle 011: 2 cycles 100: 3 cycles 101: 4 cycles 110: 5 cycles 111: 6 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from the End of the Address Cycles (Ta3) to $\overline{RD}$ and $\overline{WRxx}$ Assertion Specify the number of delay cycles from the end of the address cycles (Ta3) to $\overline{RD}$ and $\overline{WRxx}$ assertion. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	WR[3:0]	1010	R/W	Number of Wait Cycles in Read Access Specify the number of wait cycles required for read access. 0000: 0 cycles 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait is valid 1: External wait is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	HW[1:0]	00	R/W	Delay Cycles from $\overline{RD}$ and $\overline{WRxx}$ Negation to $\overline{CSn}$ Negation Specify the number of delay cycles from $\overline{RD}$ and $\overline{WRxx}$ negation to $\overline{CSn}$ negation. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

(3) Burst ROM (Asynchronous)

- CS0WCR, CS4WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	BEN	-	-	BW[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]		W[3:0]				WM	-	-	-	-	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
20	BEN	0	R/W	Burst Enable Specification Enables or disables 8-burst access to the 16-bit bus and 16-burst access to the 8-bit bus when 16-byte access is required. When this bit is cleared to 0, 2-burst access is performed four times for the 16-bit bus or 4-burst access is performed four times for the 8-bit bus. When using a device that does not support 8-burst and 16-burst access, set this bit to 1. 0: Enables 8-burst access to the 16-bit bus and 16-burst access to the 8-bit bus 1: Disables 8-burst access to the 16-bit bus and 16-burst access to the 8-bit bus
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	BW[1:0]	00	R/W	Number of Burst Wait Cycles Specify the number of wait cycles to be inserted between the second or subsequent access cycles in burst access. 00: 0 cycles 01: 1 cycle 10: 2 cycles 11: 3 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address and $\overline{\text{CSn}}$ Assertion to $\overline{\text{RD}}$ and $\overline{\text{WRxx}}$ Assertion Specify the number of delay cycles from address and $\overline{\text{CSn}}$ assertion to $\overline{\text{RD}}$ and $\overline{\text{WRxx}}$ assertion. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	W[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted in the first access cycles. 0000: 0 cycles 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bit are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	HW[1:0]	00	R/W	Delay Cycles from $\overline{RD}$ and $\overline{WRxx}$ Negation to Address and $\overline{CSn}$ Negation Specify the number of delay cycles from $\overline{RD}$ and $\overline{WRxx}$ negation to address and $\overline{CSn}$ negation. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

(4) SDRAM

When SDRAM is selected in areas 2 and 3, the WTRP1/0, WTRCD0/1, TRWL1/0, and WTRC1/0 bit settings are effective in both areas in common. When SDRAM should be connected to only one area, select area 3 for SDRAM connection. In this case, the normal space or SRAM with byte selection must be selected for area 2.

- CS2WCR

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	A2CL[1:0]		-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

Bit	Bit Name	Initial Value	R/W	Description
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8, 7	A2CL[1:0]	10	R/W	CAS Latency for Area 2 Specify the CAS latency for area 2. 00: 1 cycle 01: 2 cycles 10: 3 cycles 11: 4 cycles
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- CS3WCR

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	WTRP[1:0]		-	WTRCD[1:0]		-	A3CL[1:0]		-	-	TRWL[1:0]		-	WTRC[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14, 13	WTRP[1:0]	00	R/W	Number of Wait Cycles for Precharge Completion Specify the number of minimum wait cycles to be inserted for completion of precharge. • From activation of auto precharge to ACTV command issuance for the same bank. • From issuance of PRE/PALL command to ACTV command issuance for the same bank. • From PALL command issuance to REF command issuance in auto refresh. • From PALL command issuance to SELF command issuance in self-refresh. The setting for areas 2 and 3 is common. 00: 0 cycle (No wait cycles) 01: 1 cycle 10: 2 cycles 11: 3 cycles
12	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
11, 10	WTRCD [1:0]	01	R/W	Number of Wait Cycles from ACTV Command to READ(A)/WRIT(A) Command Specify the number of minimum wait cycles from issuing ACTV command to issuing READ(A)/WRIT(A) command. The setting for areas 2 and 3 is common. 00: 0 cycle (No wait cycles) 01: 1 cycle 10: 2 cycles 11: 3 cycles
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
8, 7	A3CL[1:0]	10	R/W	CAS Latency for Area 3 Specify the CAS latency for area 3. 00: 1 cycle 01: 2 cycles 10: 3 cycles 11: 4 cycles
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4, 3	TRWL[1:0]	00	R/W	Number of Wait Cycles for Precharge Activation Specify the minimum number of wait cycles to be inserted for activation of precharge. • From issuance of WRITA command by this LSI until auto precharge is activated in SDRAM: ACTV command is issued for the same bank in non-bank active mode. See the datasheet of the SDRAM to find the number of cycles taken from the acceptance of WRITA command by SDRAM until auto-precharge is activated. These bits should be set so that the above number of cycles will not exceed the number of cycles specified by these bits. • From issuance of WRIT command by this LSI until issuance of PRE command: Different row addresses in the same bank are accessed in bank active mode. The setting for areas 2 and 3 is common. 00: 0 cycle (No wait cycles) 01: 1 cycle 10: 2 cycles 11: 3 cycles
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	WTRC[1:0]	00	R/W	Number of Idle Cycles from REF Command Issuance/Exit from Self-Refresh Mode until ACTV/REF/MRS Command Issuance Specify the minimum number of idle cycles between commands in the following cases. • From issuance of REF command to issuance of ACTV/REF/MRS command. • From exit from self-refresh mode to issuance of ACTV/REF/MRS command. The setting for areas 2 and 3 is common. 00: 2 cycles 01: 3 cycles 10: 5 cycles 11: 8 cycles

(5) PCMCIA

- CS5WCR, CS6WCR

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	TED[3:0]				PCW[3:0]				WM	-	-	TEH[3:0]			
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
21, 20	SA[1:0]	00	R/W	Space Attribute Specification Selects memory card interface or I/O card interface when PCMCIA interface is selected. SA1: 0: Selects memory card interface for the space for A25 = 1. 1: Selects I/O card interface for the space for A25 = 1. SA0: 0: Selects memory card interface for the space for A25 = 0. 1: Selects I/O card interface for the space for A25 = 0.
19 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14 to 11	TED[3:0]	0000	R/W	Number of Delay Cycles from Address Output to $\overline{RD}$ and $\overline{WE}$ Assertion Specify the number of delay cycles from address output to $\overline{RD}$ and $\overline{WE}$ assertion in PCMCIA interface. 0000: 0.5 cycle 0001: 1.5 cycles 0010: 2.5 cycles 0011: 3.5 cycles 0100: 4.5 cycles 0101: 5.5 cycles 0110: 6.5 cycles 0111: 7.5 cycles 1000: 8.5 cycles 1001: 9.5 cycles 1010: 10.5 cycles 1011: 11.5 cycles 1100: 12.5 cycles 1101: 13.5 cycles 1110: 14.5 cycles 1111: 15.5 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	PCW[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted. 0000: 3 cycles 0001: 6 cycles 0010: 9 cycles 0011: 12 cycles 0100: 15 cycles 0101: 18 cycles 0110: 22 cycles 0111: 26 cycles 1000: 30 cycles 1001: 33 cycles 1010: 36 cycles 1011: 38 cycles 1100: 52 cycles 1101: 60 cycles 1110: 64 cycles 1111: 80 cycles
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait input is valid 1: External wait input is ignored
5, 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	TEH[3:0]	0000	R/W	Delay Cycles from $\overline{RD}$ and $\overline{WE}$ Negation to Address Specify the number of address hold cycles from $\overline{RD}$ and $\overline{WEn}$ negation in PCMCIA interface. 0000: 0.5 cycle 0001: 1.5 cycles 0010: 2.5 cycles 0011: 3.5 cycles 0100: 4.5 cycles 0101: 5.5 cycles 0110: 6.5 cycles 0111: 7.5 cycles 1000: 8.5 cycles 1001: 9.5 cycles 1010: 10.5 cycles 1011: 11.5 cycles 1100: 12.5 cycles 1101: 13.5 cycles 1110: 14.5 cycles 1111: 15.5 cycles

(6) Burst MPX-I/O

- CS6WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	MPXAW[1:0]		MPXMD	-	BW[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	W[3:0]				WM	-	-	-	-	-	-
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	MPXAW[1:0]	00	R/W	Number of Wait Cycles in Address Cycle Specifies the number of wait cycles to be inserted into the address cycle. 00: 0 cycles 01: 1 cycle 10: 2 cycles 11: 3 cycles

Bit	Bit Name	Initial Value	R/W	Description
19	MPXMD	0	R/W	Burst MPX-I/O Interface Mode Specification This bit specifies the number of bursts per access. This setting has effect only when the DMAC is set to the 16-byte transfer unit size. When other transfer unit size is selected, the setting of the MPXMD bit is ignored and there is always one burst per access. 0: Four bursts per access Four consecutive data cycles occur after the address cycle. 1: Two bursts per access Two consecutive data cycles occur after the address cycle. The correspondence between the data (D31 to D29) output in the address cycle and the transfer size is shown below.

D31	D30	D29	Transfer Size
0	0	0	Byte (one byte)
0	0	1	Word (two bytes)
0	1	0	Longword (four bytes)
0	1	1	Quadword (eight bytes) (only when MPXMD = 1)
1	0	0	16 bytes (only when MPXMD = 0)
1	0	1	Reserved
1	1	0	Reserved
1	1	1	Reserved

Bit	Bit Name	Initial Value	R/W	Description
18	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
17, 16	BW[1:0]	00	R/W	Number of Burst Wait Cycles Specifies the number of wait cycles to be inserted into the second or subsequent access cycles in burst access. 00: 0 cycles 01: 1 cycle 10: 2 cycles 11: 3 cycles
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 7	W[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted into the first burst access cycle or single access cycle. 0000: 0 cycles 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)

Bit	Bit Name	Initial Value	R/W	Description
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait is valid 1: External wait is ignored
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

(7) Burst ROM (Clock Synchronous)

- CS0WCR

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	W[3:0]				WM	-	-	-	-	-	-
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	BW[1:0]	00	R/W	Number of Burst Wait Cycles Specify the number of wait cycles to be inserted in the second or subsequent access cycles in burst access. 00: 0 cycles 01: 1 cycle 10: 2 cycles 11: 3 cycles

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 7	W[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted in the first access cycle. 0000: 0 cycles 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or nor the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait input is valid. 1: External wait input is ignored.
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

9.4.4 SDRAM Control Register (SDCR)

SDCR specifies the method to refresh and access SDRAM, and the types of SDRAMs to be connected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	A2ROW[1:0]		-	A2COL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	RFSH	RMODE	-	BACTV	-	-	-	A3ROW[1:0]		-	A3COL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R	R/W	R	R	R	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20, 19	A2ROW[1:0]	00	R/W	Number of Bits of Row Address for Area 2 Specify the number of bits of row address for area 2. 00: 11 bits 01: 12 bits 10: 13 bits 11: Reserved (setting prohibited)
18	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
17, 16	A2COL[1:0]	00	R/W	Number of Bits of Column Address for Area 2 Specify the number of bits of column address for area 2. 00: 8 bits 01: 9 bits 10: 10 bits 11: Reserved (setting prohibited)

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	RFSH	0	R/W	Refresh Control Specifies whether or not the refresh operation of SDRAM is performed. 0: No refresh 1: Refresh
10	RMODE	0	R/W	Refresh Control Specifies whether to perform auto-refresh or self-refresh when the RFSH bit is 1. When the RFSH bit is 1 and this bit is 1, self-refresh starts immediately. When the RFSH bit is 1 and this bit is 0, auto-refresh starts according to the contents that are set in RTCSR, RTCNT, and RTCOR. 0: Auto-refresh is performed 1: Self-refresh is performed
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	BACTV	0	R/W	Bank Active Mode Specifies to access whether in auto-precharge mode (using READA and WRITA commands) or in bank active mode (using READ and WRIT commands). 0: Auto-precharge mode (using READA and WRITA commands) 1: Bank active mode (using READ and WRIT commands) Note: Bank active mode can be used only for the area 3. The bus width can be set as 16 or 32 bits. When both the area 2 and area 3 are set to SDRAM, specify auto-precharge mode.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4, 3	A3ROW[1:0]	00	R/W	Number of Bits of Row Address for Area 3 Specify the number of bits of the row address for area 3. 00: 11 bits 01: 12 bits 10: 13 bits 11: Reserved (setting prohibited)
2	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
1, 0	A3COL[1:0]	00	R/W	Number of Bits of Column Address for Area 3 Specify the number of bits of the column address for area 3. 00: 8 bits 01: 9 bits 10: 10 bits 11: Reserved (setting prohibited)

9.4.5 Refresh Timer Control/Status Register (RTCSR)

RTCSR specifies various items about refresh for SDRAM.

When writing to RTCSR, write data with setting the upper 16 bits to H'A55A to cancel write protection.

Phase matching of the clock input to the refresh timer counter (RTCNT) is only performed on power-on reset. Accordingly, if the timer is started with CKS[2:0] set to other than B'000, there will be an error contained in the period until the first setting of the compare match flag.

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CMF	CMIE	CKS[2:0]			RRC[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R/W	Write Protect Cancellation When writing to RTCSR, write H'A55A to these bits to cancel write protection. These bits are always read as 0.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	CMF	0	R/W	Compare Match Flag This is a status flag which indicates that a compare match occurs between the refresh timer counter (RTCNT) and refresh time constant register (RTCOR). This bit is set or cleared in the following conditions. 0: Clearing condition: When 0 is written in CMF after reading out RTCSR during CMF = 1. 1: Setting condition: When the condition RTCNT = RTCOR is satisfied.

Bit	Bit Name	Initial Value	R/W	Description
6	CMIE	0	R/W	Compare Match Interrupt Enable Enables or disables CMF interrupt requests when the CMF bit in RTCSR is set to 1. 0: Disables CMF interrupt requests. 1: Enables CMF interrupt requests.
5 to 3	CKS[2:0]	000	R/W	Clock Select Select the clock input to count-up the refresh timer counter (RTCNT). 000: Stop the counting-up 001: Bϕ/4 010: Bϕ/16 011: Bϕ/64 100: Bϕ/256 101: Bϕ/1024 110: Bϕ/2048 111: Bϕ/4096
2 to 0	RRC[2:0]	000	R/W	Refresh Count Specify the number of continuous refresh cycles, when the refresh request occurs after the coincidence of the values of the refresh timer counter (RTCNT) and the refresh time constant register (RTCOR). These bits can make the period of occurrence of refresh long. 000: Once 001: Twice 010: 4 times 011: 6 times 100: 8 times 101: Reserved (setting prohibited) 110: Reserved (setting prohibited) 111: Reserved (setting prohibited)

9.4.6 Refresh Timer Counter (RTCNT)

RTCNT is an 8-bit counter that increments using the clock selected by bits CKS2 to CKS0 in RTCSR. When RTCNT matches RTCOR, RTCNT is cleared to 0. The value in RTCNT returns to 0 after counting up to 255. When writing to RTCNT, write data with setting the upper 16 bits to H'A55A to cancel write protection.

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

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

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R/W	Write Protect Cancellation When writing to RTCNT, write H'A55A to these bits to cancel write protection. These bits are always read as 0.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0		All 0	R/W	8-bit counter

9.4.7 Refresh Time Constant Register (RTCOR)

RTCOR is an 8-bit register. When RTCOR matches RTCNT, the CMF bit in RTCSR is set to 1 and RTCNT is cleared to 0.

When the RFSH bit in SDCR is set to 1, a memory refresh request is issued by this matching signal. This request is maintained until the refresh operation is performed. If the request is not processed when the next matching occurs, the previous request is ignored.

When the CMIE bit in RTCSR is set to 1, an interrupt request is issued by this matching signal. The request is output continuously until the CMF bit in RTCSR is cleared. Clearing the CMF bit only affects the interrupt request and does not clear the refresh request. Therefore, a combination of refresh request and interval timer interrupt can be specified so that the number of refresh requests are counted by using timer interrupts while refresh is performed periodically. When writing to RTCOR, write data with setting the upper 16 bits to H'A55A to cancel write protection.

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

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

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R/W	Write Protect Cancellation When writing to RTCOR, write H'A55A to these bits to cancel write protection. These bits are always read as 0.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0		All 0	R/W	8-bit counter

9.4.8 Bus Function Extending Register (BSCEHR)

BSCEHR is a 16-bit register that specifies the timing of bus release by the DTC and DMAC. It also specifies the application of priority in transfer operations and enables or disables the functions that have the effect of decreasing numbers of cycles over which the DTC is active. The differences in DTC operation made by the combinations of the DTLOCK, CSSTP1, and DTBST bits settings are described in section 8.5.9, DTC Bus Releasing Timing.

Setting the CSSTP2 bit can improve the transfer performance of the DMAC in burst-mode transfer and of the DTC when the DTLOCK bit is 0.

Furthermore, setting the CSSTP3 bit selects whether or not access to the external space by the CPU takes priority over DTC or DMAC transfer in cycle-steal mode. The DTC short address mode is implemented by setting the DTSA bit. For details of the short address mode, see section 8.4, Location of Transfer Information and DTC Vector Table.

A DTC activation priority order can be set up for the DTC activation sources. The DTPR bit selects whether or not this priority order is valid or invalid when multiple sources issue activation requests before DTC activation. The corresponding bit from among DMMTU4 to DMMTU0 must be set for MTU2-triggered transfer by the DMAC in the burst mode. Do not modify this register while the DMAC or DTC is active.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTLOCK	CSSTP1	-	CSSTP2	DTBST	DTSA	CSSTP3	DTPR	-	-	-	DMMTU4	DMMTU3	DMMTU2	DMMTU1	DMMTU0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	DTLOCK	0	R/W	DTC Lock Enable Specifies the timing of bus release by the DTC. 0: The DTC releases the bus on generation of the NOP cycle that follows vector read or write-back of transfer information. 1: The DTC releases the bus after vector read, on generation of the NOP cycle that follows vector read, after transfer information read, after a round of data transfer, or after write-back of transfer information.

Bit	Bit Name	Initial Value	R/W	Description
14	CSSTP1	0	R/W	Select Bus Release on NOP Cycle Generation by DTC Specifies whether or not the bus is released in response to requests from the CPU for external space access on generation of the NOP cycle that follows reading of the vector address. If, however, the CSSTP2 bit is 1, bus mastership is retained until all transfer is complete, regardless of the setting of this bit. 0: The bus is released on generation of the NOP cycle by the DTC. 1: The bus is not released on generation of the NOP cycle by the DTC.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	CSSTP2	0	R/W	Select Bus Release during Burst-Mode-DMAC/DTC Transfer This setting applies to DTC transfer when the DTLOCK bit is 0 and burst-mode DMAC transfer when the DMAC is in channel-fixed mode, and the activating request was an external request or was from MTU2. The value specifies whether the bus mastership is or is not to be released after each round of transfer in response to a request from the CPU for access to the external space. • DMAC transfer 0: Release the bus after each round of data transfer. 1: Only release the bus after all data transfer is complete. Note: In round-robin mode, the bus is only released after all data transfer is complete, regardless of the setting of this bit. • DTC transfer 0: When the DTLOCK and CSSTP1 bits are 0, the bus is released on generation of the NOP cycle after reading of the vector address. When the DTLOCK bit is 0 and the CSSTP1 bit is 1, the bus is released after each round of data transfer. 1: Only release the bus mastership after all data transfer is complete.

Bit	Bit Name	Initial Value	R/W	Description
11	DTBST	0	R/W	DTC Burst Enable Selects whether or not the DTC retains the bus mastership and remains continuously active until all transfer operations are complete when multiple DTC activation requests have been generated. 0: Release the bus on the completion of transfer for each individual DTC activation source. 1: Keep the DTC continuously active, i.e. only release the bus on completion of processing for all DTC activation sources. Notes: When this bit is set to 1, the following restrictions apply. 1. Clock setting with the frequency control register (FRQCR) must be Iϕ: Bϕ: Pϕ: MIϕ: MPϕ: = 8: 4: 4: 4: 4, 4: 2: 2: 2: 2, or 2: 1: 1: 1: 1 2. The vector information must be in on-chip ROM or on-chip RAM. 3. The transfer information must be in on-chip RAM. 4. Transfer must be between the on-chip RAM and an on-chip peripheral module or between external memory and an on-chip peripheral module.
10	DTSA	0	R/W	DTC Short Address Mode In this mode, the information that specifies a DTC transfer takes up only 3 longwords. 0: Each transfer information is read out as 4 longwords. The transfer information are arranged as shown in figure 8.2 (normal address mode). 1: Each transfer information is read out as 3 longwords. The transfer information are arranged as shown in figure 8.2 (short-address mode). Note: Transfer in short-address mode is only available between on-chip peripheral modules and on-chip RAM, because the higher-order 8 bits of the SAR and DAR are considered to be all 1.

Bit	Bit Name	Initial Value	R/W	Description
9	CSSTP3	0	R/W	Select Priority for External Memory Access by CPU Specifies whether or not access to the external space by the CPU takes priority over DTC or DMAC transfer in cycle-steal mode. 0: DMAC transfer and DTC transfer have priority. 1: External space access from the CPU has priority. Note: When this bit is 0, and access to internal I/O from the CPU is immediately followed by access to external space from the CPU, a NOP 1Bϕ in duration is inserted between the two access cycles.
8	DTPR	0	R/W	Application of Priority in DTC Activation When multiple DTC activation requests are generated before the DTC is activated, specify whether transfer starts from the first request to have been generated or is in accord with the priority order for DTC activation requests. However, when multiple DTC activation requests have been issued while the DTC is active, the next transfer to be triggered will be that with the highest DTC activation priority. 0: Start transfer in response to the first request to have been generated. 1: Start transfer in accord with DTC activation request priority. Notes: When this bit is set to 1, the following restrictions apply. 1. The vector information must be in on-chip ROM or on-chip RAM. 2. The transfer information must be in on-chip RAM. 3. Skipping of transfer information reading is always disabled.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	DMMTU4	0	R/W	Enable Burst-Mode DMAC Transfer with TGIA_4 Activation Source Setting this bit to 1 enables burst-mode DMA transfer triggered by the TGIA_4 interrupt from MTU2. 0: DMA transfer in burst mode is disabled when TGIA_4 is the activation source. 1: DMA transfer in burst mode is enabled when TGIA_4 is the activation source. Note: Clear this bit during DMA transfer in cycle-steal mode.
3	DMMTU3	0	R/W	Enable Burst-Mode DMAC Transfer with TGIA_3 Activation Source Setting this bit to 1 enables burst-mode DMA transfer triggered by the TGIA_3 interrupt from MTU2. 0: DMA transfer in burst mode is disabled when TGIA_3 is the activation source. 1: DMA transfer in burst mode is enabled when TGIA_3 is the activation source. Note: Clear this bit during DMA transfer in cycle-steal mode.
2	DMMTU2	0	R/W	Enable Burst-Mode DMAC Transfer with TGIA_2 Activation Source Setting this bit to 1 enables burst-mode DMA transfer triggered by the TGIA_2 interrupt from MTU2. 0: DMA transfer in burst mode is disabled when TGIA_2 is the activation source. 1: DMA transfer in burst mode is enabled when TGIA_2 is the activation source. Note: Clear this bit during DMA transfer in cycle-steal mode.

Bit	Bit Name	Initial Value	R/W	Description
1	DMMTU1	0	R/W	Enable Burst-Mode DMAC Transfer with TGIA_1 Activation Source Setting this bit to 1 enables burst-mode DMA transfer triggered by the TGIA_1 interrupt from MTU2. 0: DMA transfer in burst mode is disabled when TGIA_1 is the activation source. 1: DMA transfer in burst mode is enabled when TGIA_1 is the activation source. Note: Clear this bit during DMA transfer in cycle-steal mode.
0	DMMTU0	0	R/W	Enable Burst-Mode DMAC Transfer with TGIA_0 Activation Source Setting this bit to 1 enables burst-mode DMA transfer triggered by the TGIA_0 interrupt from MTU2. 0: DMA transfer in burst mode is disabled when TGIA_0 is the activation source. 1: DMA transfer in burst mode is enabled when TGIA_0 is the activation source. Note: Clear this bit during DMA transfer in cycle-steal mode.

9.5 Operation

9.5.1 Endian/Access Size and Data Alignment

This LSI supports big endian, in which the 0 address is the most significant byte (MSB) in the byte data.

Three data bus widths (8 bits, 16 bits, and 32 bits) are available for normal memory and SRAM with byte selection, and two data bus widths (16 bits and 32 bits) are available for SDRAM. For PCMCIA interface, two data bus widths (8 bits and 16 bits) are available. For MPX-I/O, the data bus width is fixed at 8 bits or 16 bits, or 8 bits or 16 bits can be selected by the access address. For burst MPX-I/O, the data bus width is fixed at 32 bits. Data alignment is performed in accordance with the data bus width of the respective device. This also means that when longword data is read from a byte-width device, the read operation must be done four times. In this LSI, data alignment and conversion of data length are performed automatically between the respective interfaces.

Tables 9.17 to 9.19 show the relationship between device data width and access unit.

Table 9.17 32-Bit External Device Access and Data Alignment

	Data Bus				Strobe Signals			
Operation	D31 to D24	D23 to D16	D15 to D8	D7 to D0	$\overline{\text{WRHH}}$, DQMUU	$\overline{\text{WRHL}}$, DQMUL	$\overline{\text{WRH}}$, DQMLU	$\overline{\text{WRL}}$, DQMLL
Byte access at 0	Data 7 to Data 0	—	—	—	Assert	—	—	—
Byte access at 1	—	Data 7 to Data 0	—	—	—	Assert	—	—
Byte access at 2	—	—	Data 7 to Data 0	—	—	—	Assert	—
Byte access at 3	—	—	—	Data 7 to Data 0	—	—	—	Assert
Word access at 0	Data 15 to Data 8	Data 7 to Data 0	—	—	Assert	Assert	—	—
Word access at 2	—	—	Data 15 to Data 8	Data 7 to Data 0	—	—	Assert	Assert
Longword access at 0	Data 31 to Data 24	Data 23 to Data 16	Data 15 to Data 8	Data 7 to Data 0	Assert	Assert	Assert	Assert

Table 9.18 16-Bit External Device Access and Data Alignment

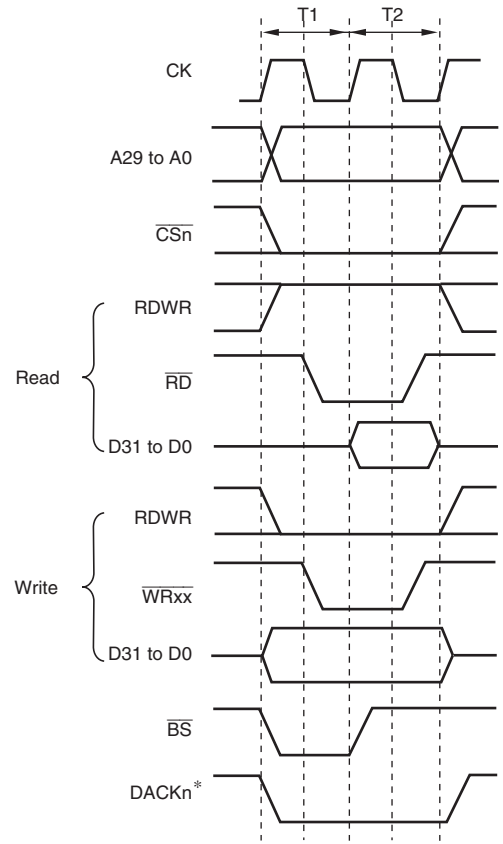
Operation		Data Bus				Strobe Signals			
		D31 to D24	D23 to D16	D15 to D8	D7 to D0	$\overline{\text{WRHH}}$, DQMUU	$\overline{\text{WRHL}}$, DQMUL	$\overline{\text{WRH}}$, DQMLU	$\overline{\text{WRL}}$, DQMLL
Byte access at 0		—	—	Data 7 to Data 0	—	—	—	Assert	—
Byte access at 1		—	—	—	Data 7 to Data 0	—	—	—	Assert
Byte access at 2		—	—	Data 7 to Data 0	—	—	—	Assert	—
Byte access at 3		—	—	—	Data 7 to Data 0	—	—	—	Assert
Word access at 0		—	—	Data 15 to Data 8	Data 7 to Data 0	—	—	Assert	Assert
Word access at 2		—	—	Data 15 to Data 8	Data 7 to Data 0	—	—	Assert	Assert
Longword access at 0	1st time at 0	—	—	Data 31 to Data 24	Data 23 to Data 16	—	—	Assert	Assert
	2nd time at 2	—	—	Data 15 to Data 8	Data 7 to Data 0	—	—	Assert	Assert

Table 9.19 8-Bit External Device Access and Data Alignment

Operation		Data Bus				Strobe Signals			
		D31 to D24	D23 to D16	D15 to D8	D7 to D0	$\overline{\text{WRHH}}$, DQMUU	$\overline{\text{WRHL}}$, DQMUL	$\overline{\text{WRH}}$, DQMLU	$\overline{\text{WRL}}$, DQMLL
Byte access at 0		—	—	—	Data 7 to Data 0	—	—	—	Assert
Byte access at 1		—	—	—	Data 7 to Data 0	—	—	—	Assert
Byte access at 2		—	—	—	Data 7 to Data 0	—	—	—	Assert
Byte access at 3		—	—	—	Data 7 to Data 0	—	—	—	Assert
Word access at 0	1st time at 0	—	—	—	Data 15 to Data 8	—	—	—	Assert
	2nd time at 1	—	—	—	Data 7 to Data 0	—	—	—	Assert
Word access at 2	1st time at 2	—	—	—	Data 15 to Data 8	—	—	—	Assert
	2nd time at 3	—	—	—	Data 7 to Data 0	—	—	—	Assert
Longword access at 0	1st time at 0	—	—	—	Data 31 to Data 24	—	—	—	Assert
	2nd time at 1	—	—	—	Data 23 to Data 16	—	—	—	Assert
	3rd time at 2	—	—	—	Data 15 to Data 8	—	—	—	Assert
	4th time at 3	—	—	—	Data 7 to Data 0	—	—	—	Assert

9.5.2 Normal Space Interface

Basic Timing: For access to a normal space, this LSI uses strobe signal output in consideration of the fact that mainly SRAM without a byte selection will be directly connected. When using SRAM with a byte-selection pin, see section 9.5.8, SRAM Interface with Byte Selection. Figure 9.2 shows the basic timings of normal space access. A no-wait normal access is completed in two cycles. The $\overline{BS}$ signal is asserted for one cycle to indicate the start of a bus cycle.



Note: * The waveform for DACKn is when active low is specified.

Figure 9.2 Normal Space Basic Access Timing (Access Wait 0)

There is no access size specification when reading. The correct access start address is output in the least significant bit of the address, but since there is no access size specification, 32 bits are always read in a 32-bit device or 16 bits are always read in a 16-bit device. When writing, only the $\overline{\text{WRxx}}$ signal for the byte to be written is asserted.

It is necessary to control of outputing the data that has been read using $\overline{\text{RD}}$ when a buffer is established in the data bus. The RDWR signal is in a read state (high output) when no access has been carried out. Therefore, care must be taken when controlling the external data buffer using RDWR, to avoid collision.

Figures 9.3 and 9.4 show the basic timings of continuous accesses to normal space. If the WM bit in CSnWCR is cleared to 0, a Tnop cycle is inserted to evaluate the external wait (figure 9.3). If the WM bit in CSnWCR is set to 1, external waits are ignored and no Tnop cycle is inserted (figure 9.4).

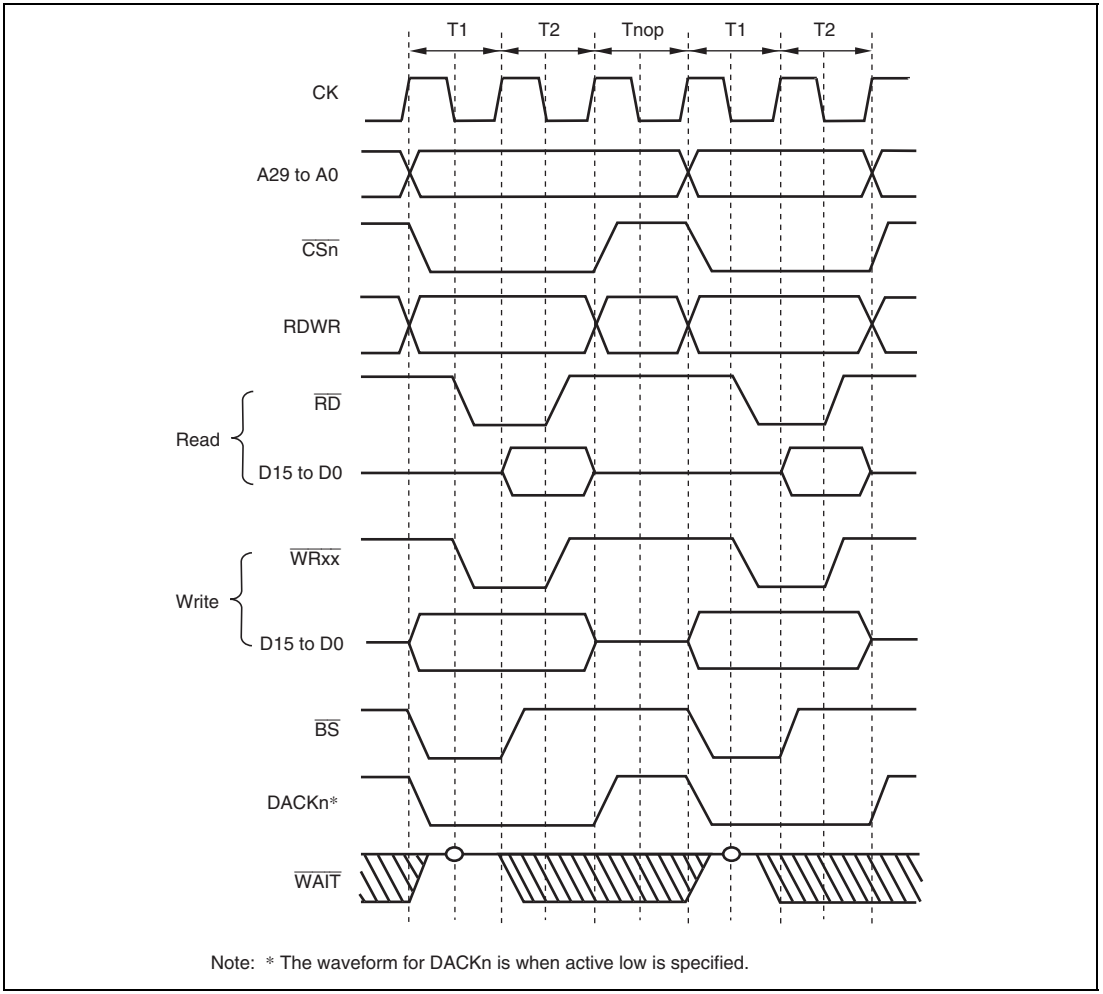
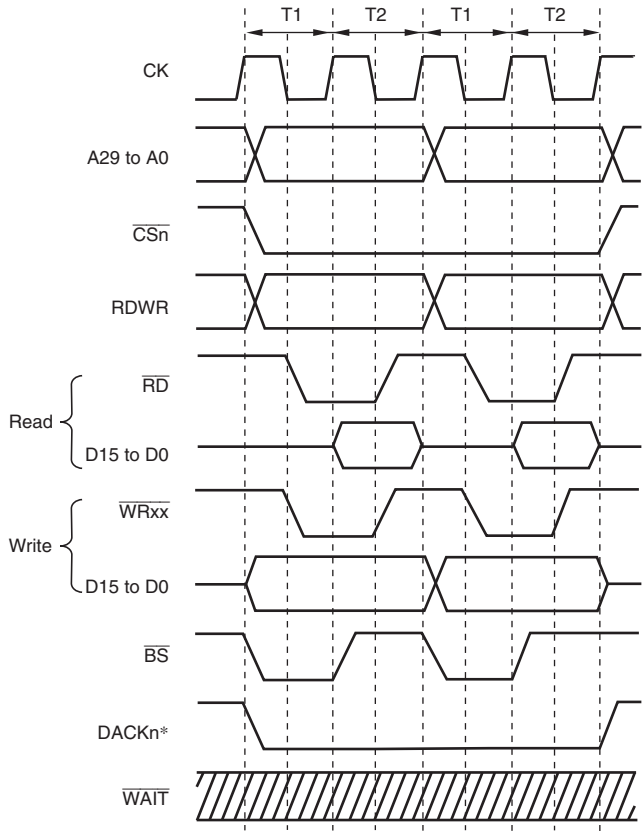


Figure 9.3 Continuous Access for Normal Space 1
Bus Width = 16 Bits, Longword Access, WM Bit in CSnWCR = 0
(Access Wait = 0, Cycle Wait = 0)



Note: * The waveform for DACKn is when active low is specified.

Figure 9.4 Continuous Access for Normal Space 2
Bus Width = 16 Bits, Longword Access, WM Bit in CSnWCR = 1
(Access Wait = 0, Cycle Wait = 0)

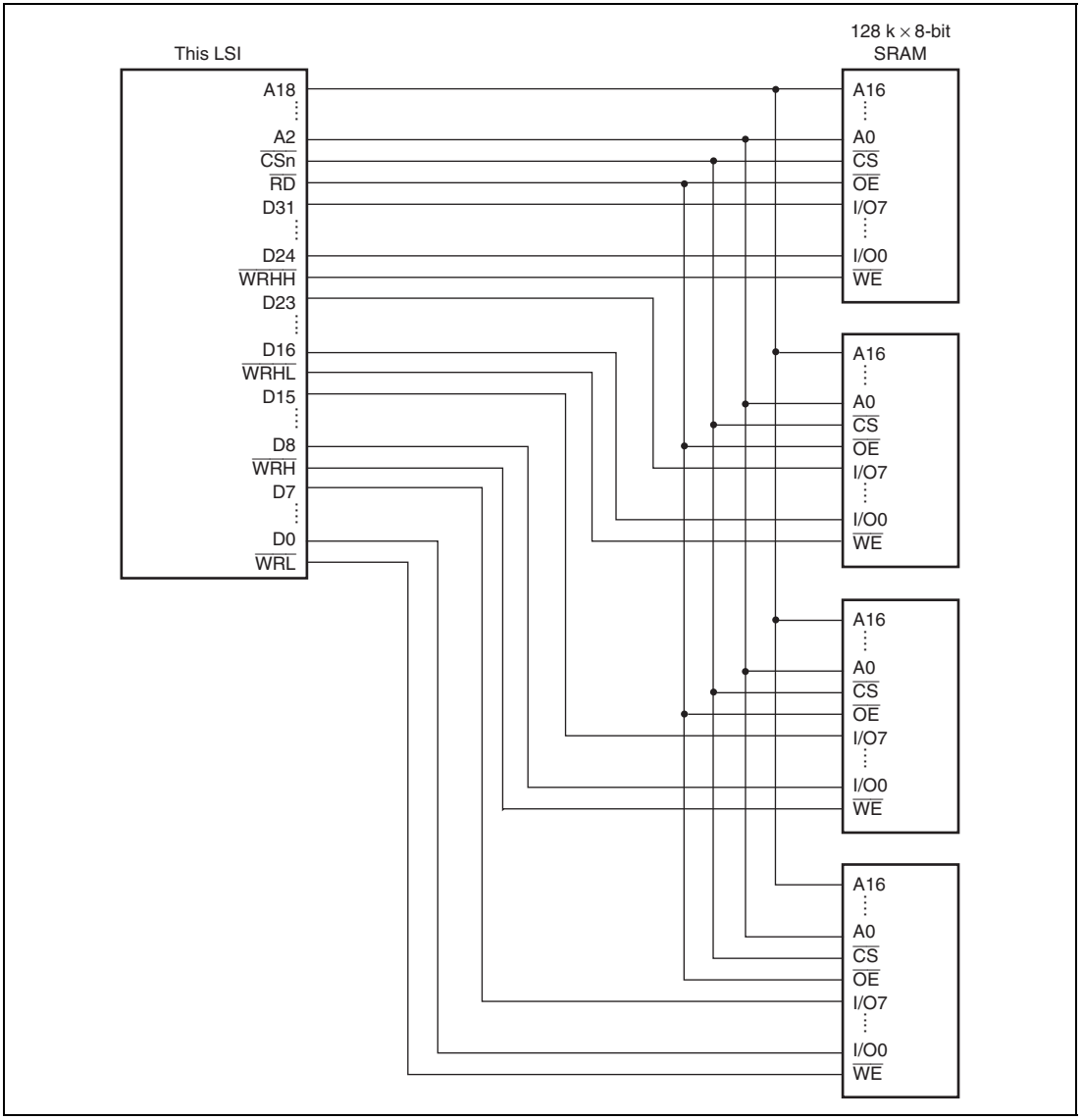


Figure 9.5 Example of 32-Bit Data-Width SRAM Connection

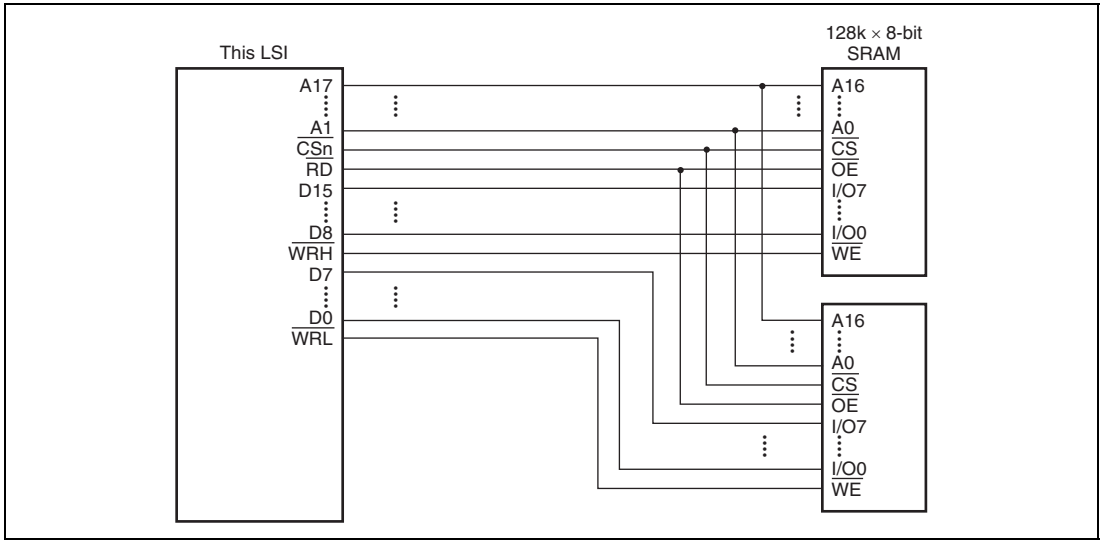


Figure 9.6 Example of 16-Bit Data-Width SRAM Connection

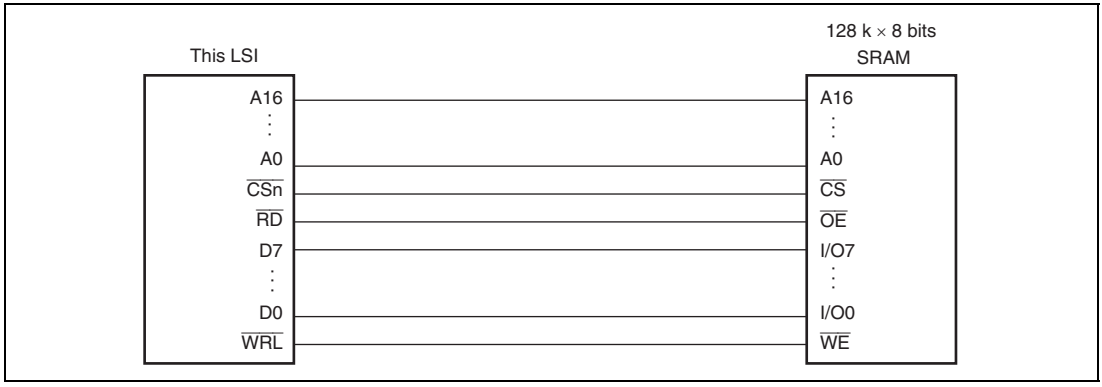
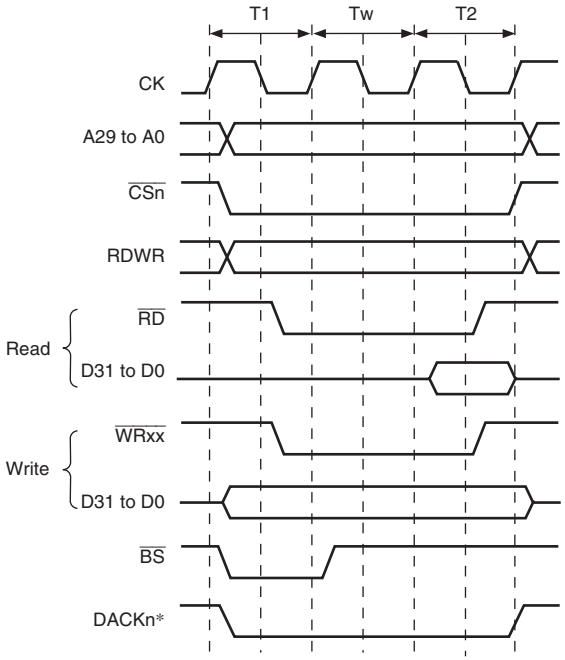


Figure 9.7 Example of 8-Bit Data-Width SRAM Connection

9.5.3 Access Wait Control

Wait cycle insertion on a normal space access can be controlled by the settings of bits WR3 to WR0 in CSnWCR. It is possible to insert wait cycles independently in read access and in write access. The specified number of Tw cycles is inserted as wait cycles in a normal space access shown in figure 9.8.



Note: * The waveform for DACKn is when active low is specified.

Figure 9.8 Wait Timing for Normal Space Access (Software Wait Only)

When the WM bit in CSnWCR is cleared to 0, the external wait input $\overline{\text{WAIT}}$ signal is also sampled. $\overline{\text{WAIT}}$ pin sampling is shown in figure 9.9. A 2-cycle wait is specified as a software wait. The $\overline{\text{WAIT}}$ signal is sampled at the falling edge of CK at the transition from the T1 or Tw cycle to the T2 cycle.

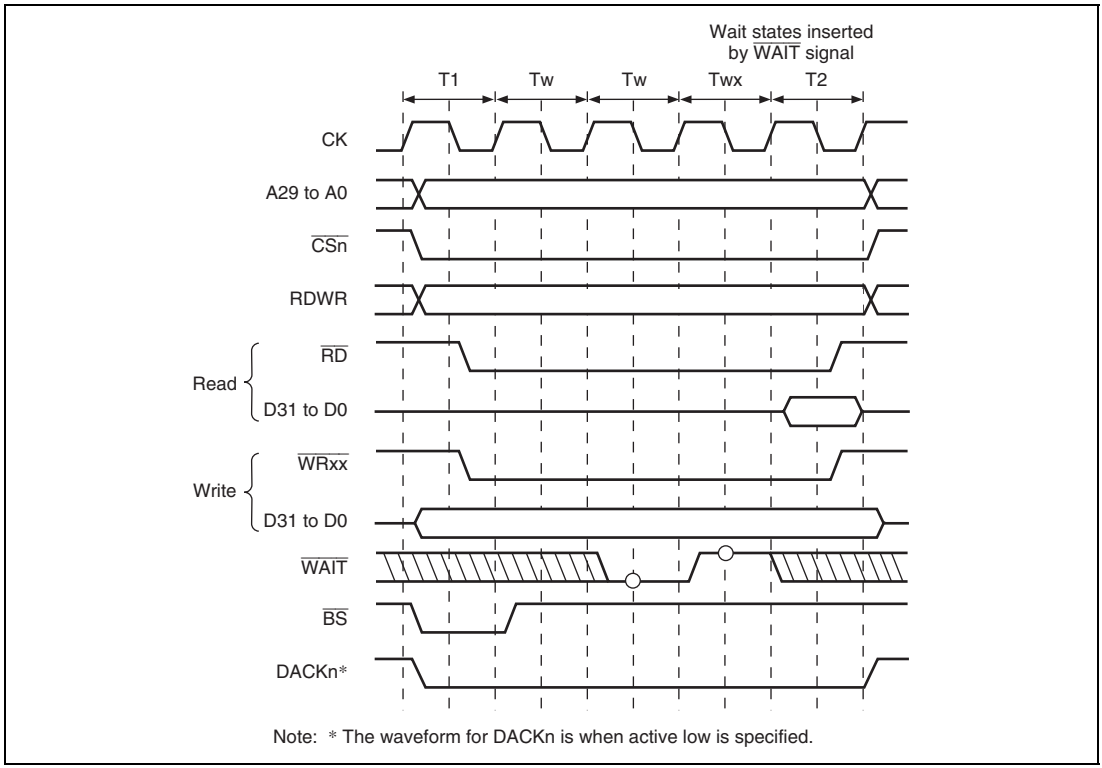


Figure 9.9 Wait State Timing for Normal Space Access (Wait State Insertion Using $\overline{\text{WAIT}}$ Signal)

9.5.4 $\overline{\text{CSn}}$ Assert Period Extension

The number of cycles from $\overline{\text{CSn}}$ assertion to $\overline{\text{RD}}$, $\overline{\text{WRxx}}$ assertion can be specified by setting bits SW1 and SW0 in CSnWCR. The number of cycles from $\overline{\text{RD}}$, $\overline{\text{WRxx}}$ negation to $\overline{\text{CSn}}$ negation can be specified by setting bits HW1 and HW0. Therefore, a flexible interface to an external device can be obtained. Figure 9.10 shows an example. A Th cycle and a Tf cycle are added before and after an ordinary cycle, respectively. In these cycles, $\overline{\text{RD}}$ and $\overline{\text{WRxx}}$ are not asserted, while other signals are asserted. The data output is prolonged to the Tf cycle, and this prolongation is useful for devices with slow writing operations.

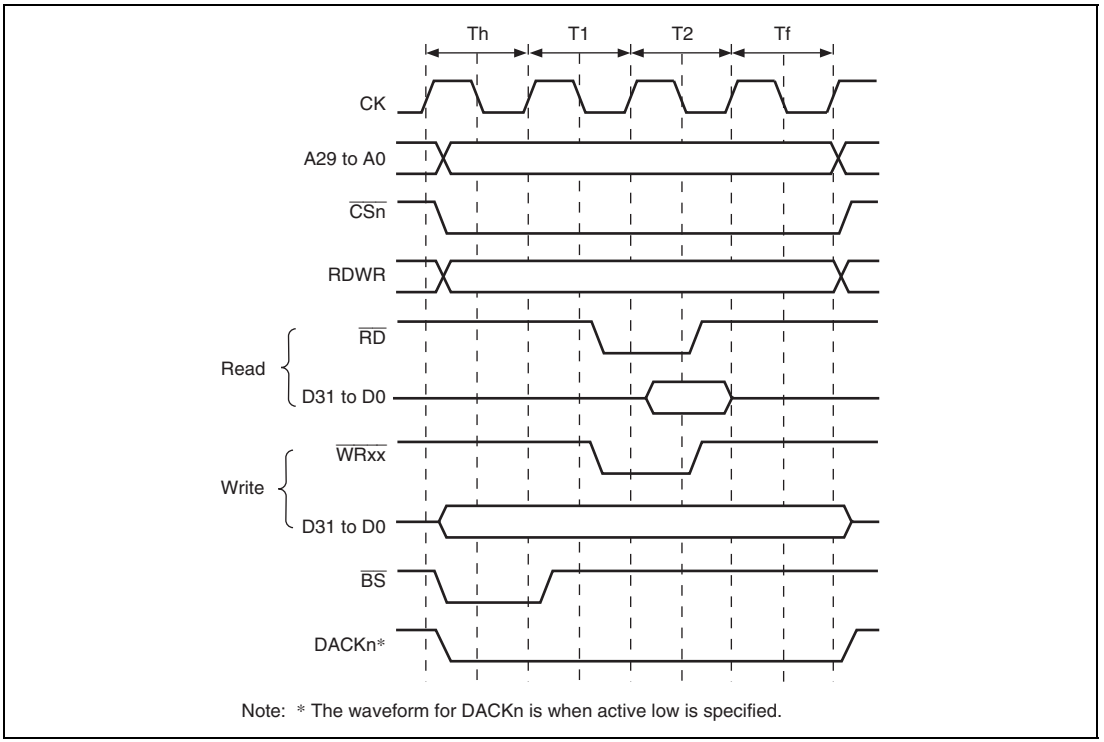


Figure 9.10 $\overline{\text{CSn}}$ Assert Period Extension

9.5.5 MPX-I/O Interface

Access timing for the MPX space is shown below. In the MPX space, $\overline{CSn}$, $\overline{AH}$, $\overline{RD}$, and $\overline{WRxx}$ signals control the accessing. The basic access for the MPX space consists of 2 cycles of address output followed by an access to a normal space. The bus width for the address output cycle or the data input/output cycle is fixed to 8 bits or 16 bits. Alternatively, it can be 8 bits or 16 bits depending on the address to be accessed.

Output of the addresses D15 to D0 or D7 to D0 is performed from cycle Ta2 to cycle Ta3. Because cycle Ta1 has a high-impedance state, collisions of addresses and data can be avoided without inserting idle cycles, even in continuous accesses. Address output is increased to 3 cycles by setting the MPXW bit in the CS5WCR register to 1. The RDWR signal is output at the same time as the $\overline{CSn}$ signal; it is high in the read cycle and low in the write cycle.

The data cycle is the same as that in a normal space access.

The delay cycle of SW[1:0] is inserted between Ta3 and T1 cycle.

The delay cycle of HW[1:0] is added after T2 cycle.

Timing charts are shown in figures 9.11 to 9.14.

Note that the operation timing of the MPX-I/O interface differs between the SH7080 group and the SH7040. For example, the $\overline{AH}$ signal is negated (high-level) in the SH7080 group and asserted (low-level) in the SH7040 group during access to other than MPX-I/O space.

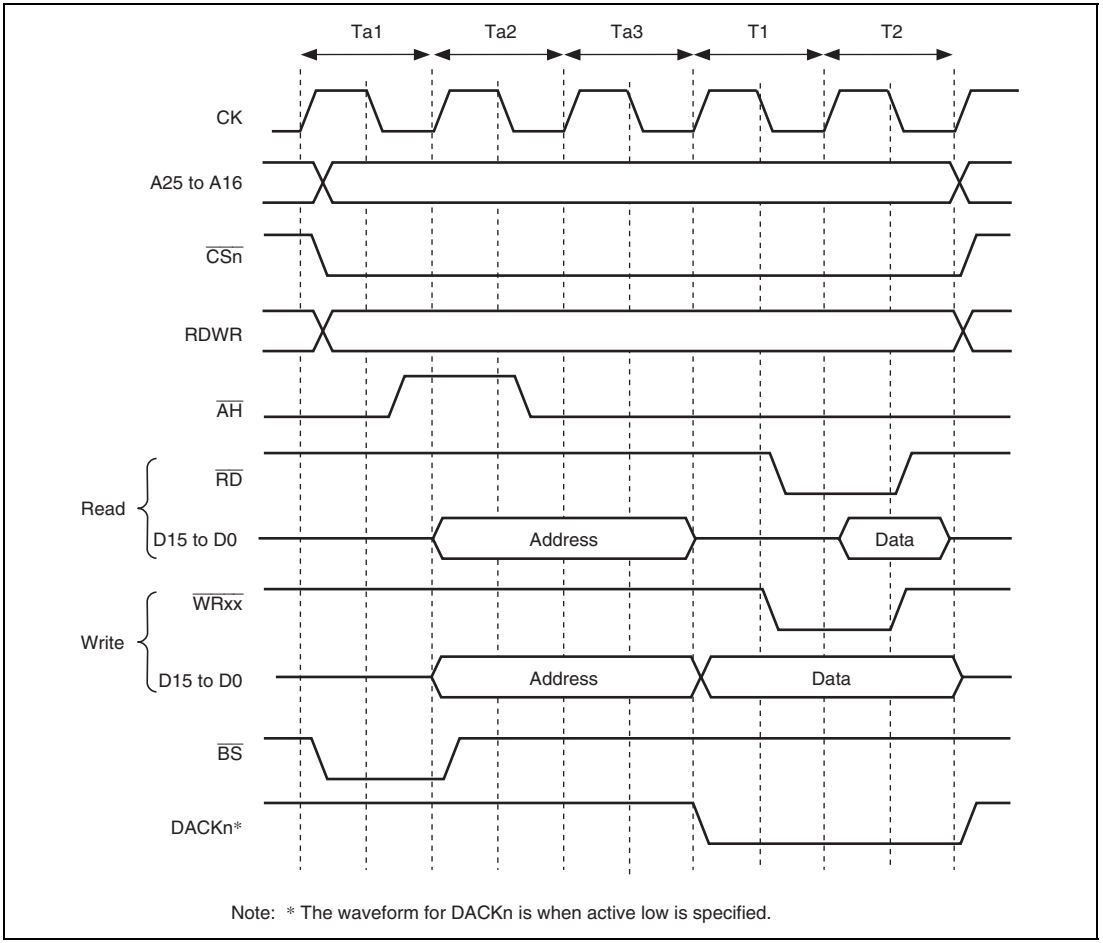


Figure 9.11 Access Timing for MPX Space (Address Cycle No Wait, Data Cycle No Wait)

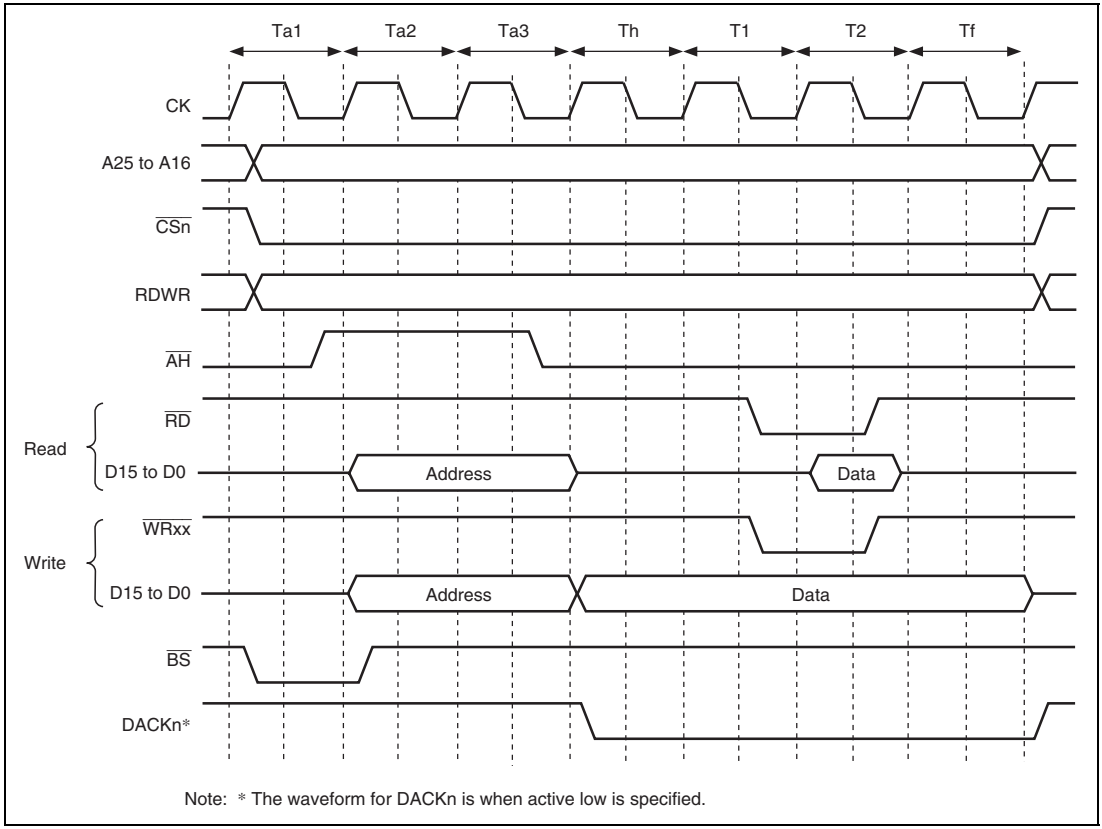


Figure 9.12 Access Timing for MPX Space (Address Cycle No Wait, Assert Period Expansion 1.5, Data Cycle No Wait, Negation Period Expansion 1.5)

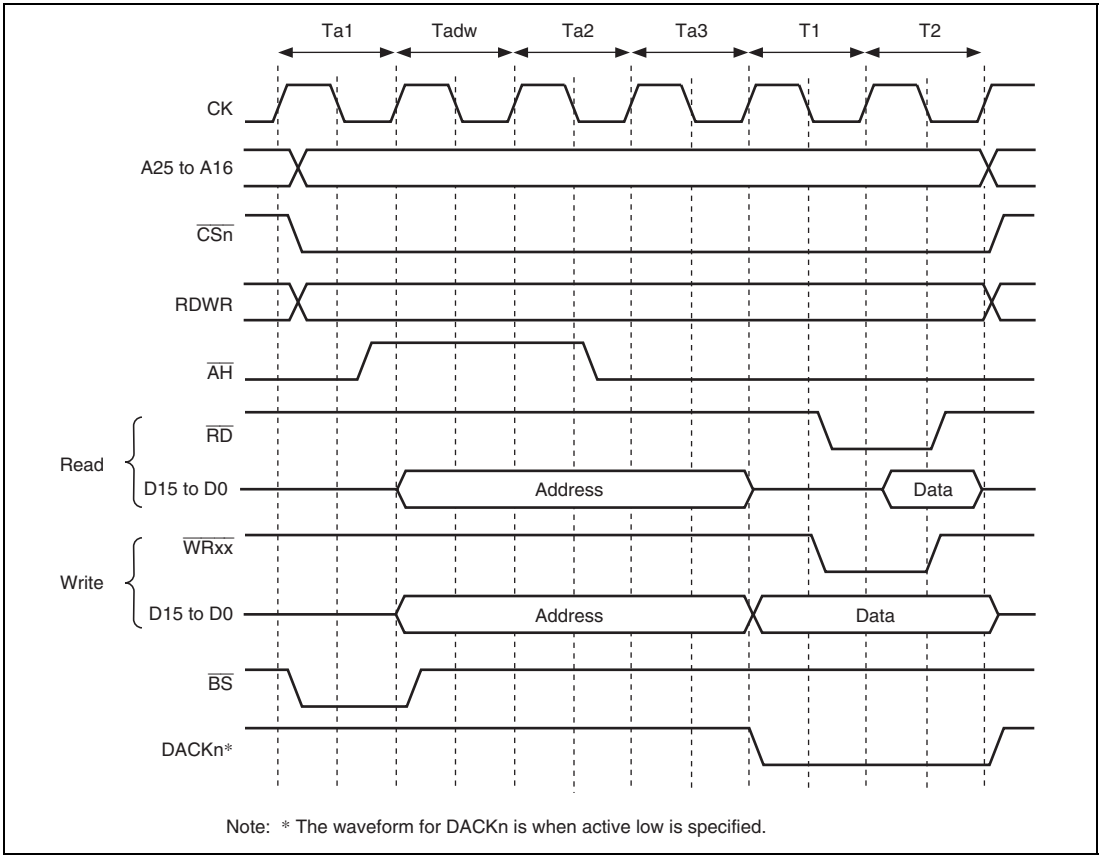


Figure 9.13 Access Timing for MPX Space (Address Cycle Wait 1, Data Cycle No Wait)

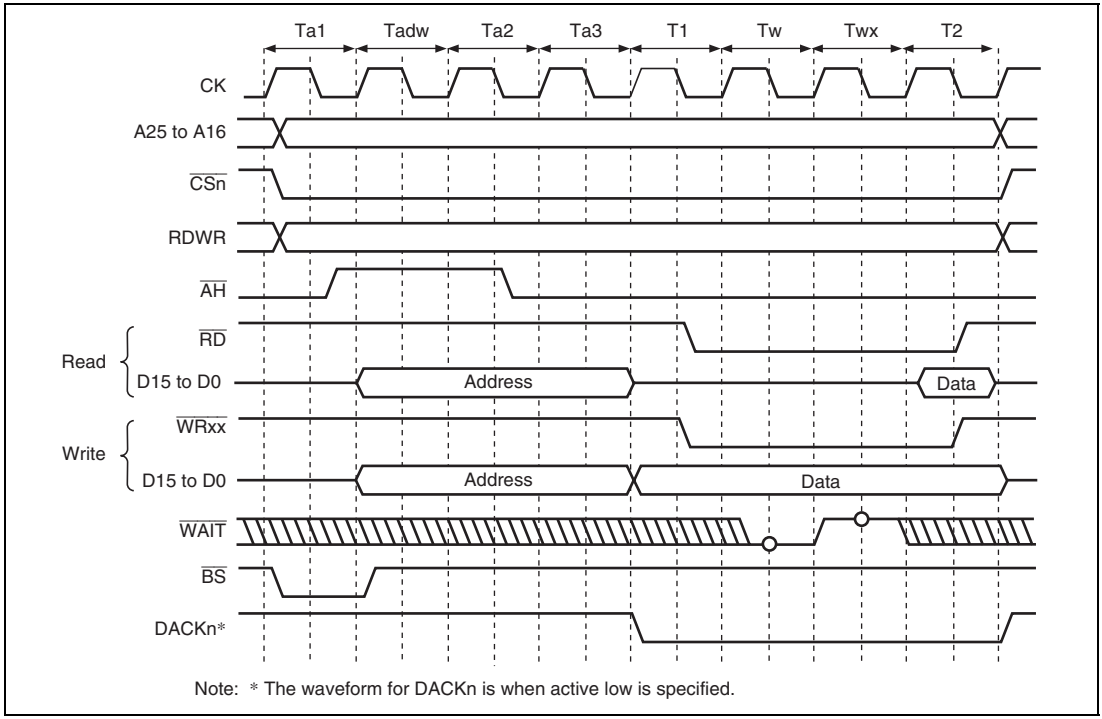


Figure 9.14 Access Timing for MPX Space (Address Cycle Access Wait 1, Data Cycle Wait 1, External Wait 1)

9.5.6 SDRAM Interface

SDRAM Direct Connection: The SDRAM that can be connected to this LSI is a product that has 11/12/13 bits of row address, 8/9/10 bits of column address, 4 or less banks, and uses the A10 pin for setting precharge mode in read and write command cycles.

The control signals for direct connection of SDRAM are $\overline{\text{RASU}}$, $\overline{\text{RASL}}$, $\overline{\text{CASU}}$, $\overline{\text{CASL}}$, RDWR, DQMUU, DQMLU, DQMLL, CKE, $\overline{\text{CS2}}$, and $\overline{\text{CS3}}$. All the signals other than $\overline{\text{CS2}}$ and $\overline{\text{CS3}}$ are common to all areas, and signals other than CKE are valid when $\overline{\text{CS2}}$ or $\overline{\text{CS3}}$ is asserted. SDRAM can be connected to up to 2 spaces. The data bus width of the area that is connected to SDRAM can be set to 32 or 16 bits.

Burst read/single write (burst length 1) and burst read/burst write (burst length 1) are supported as SDRAM operating mode.

Commands for SDRAM can be specified by $\overline{\text{RASU}}$, $\overline{\text{RASL}}$, $\overline{\text{CASU}}$, $\overline{\text{CASL}}$, RDWR, and specific address signals. These commands are shown below.

- NOP
- Auto-refresh (REF)
- Self-refresh (SELF)
- All banks precharge (PALL)
- Specified bank precharge (PRE)
- Bank active (ACTV)
- Read (READ)
- Read with precharge (READA)
- Write (WRIT)
- Write with precharge (WRITA)
- Write mode register (MRS)

The byte to be accessed is specified by DQMUU, DQMUL, DQMLU and DQMLL. Reading or writing is performed for a byte whose corresponding DQMxx low. For details on the relationship between DQMxx and the byte to be accessed, refer to section 9.5.1, Endian/Access Size and Data Alignment.

Figures 9.15 to 9.17 show shows an example of the connection of SDRAM with the LSI.

As shown in figure 9.17, two sets of SDRAMs of 32 Mbytes or smaller can be connected to the same CS space by using $\overline{\text{RASU}}$, $\overline{\text{RASL}}$, $\overline{\text{CASU}}$, and $\overline{\text{CASL}}$. In this case, a total of 8 banks are assigned to the same CS space: 4 banks specified by $\overline{\text{RASL}}$ and $\overline{\text{CASL}}$, and 4 banks specified by $\overline{\text{RASU}}$ and $\overline{\text{CASU}}$. When accessing the address with A25 = 0, $\overline{\text{RASL}}$ and $\overline{\text{CASL}}$ are asserted. When accessing the address with A25 = 1, $\overline{\text{RASU}}$ and $\overline{\text{CASU}}$ are asserted.

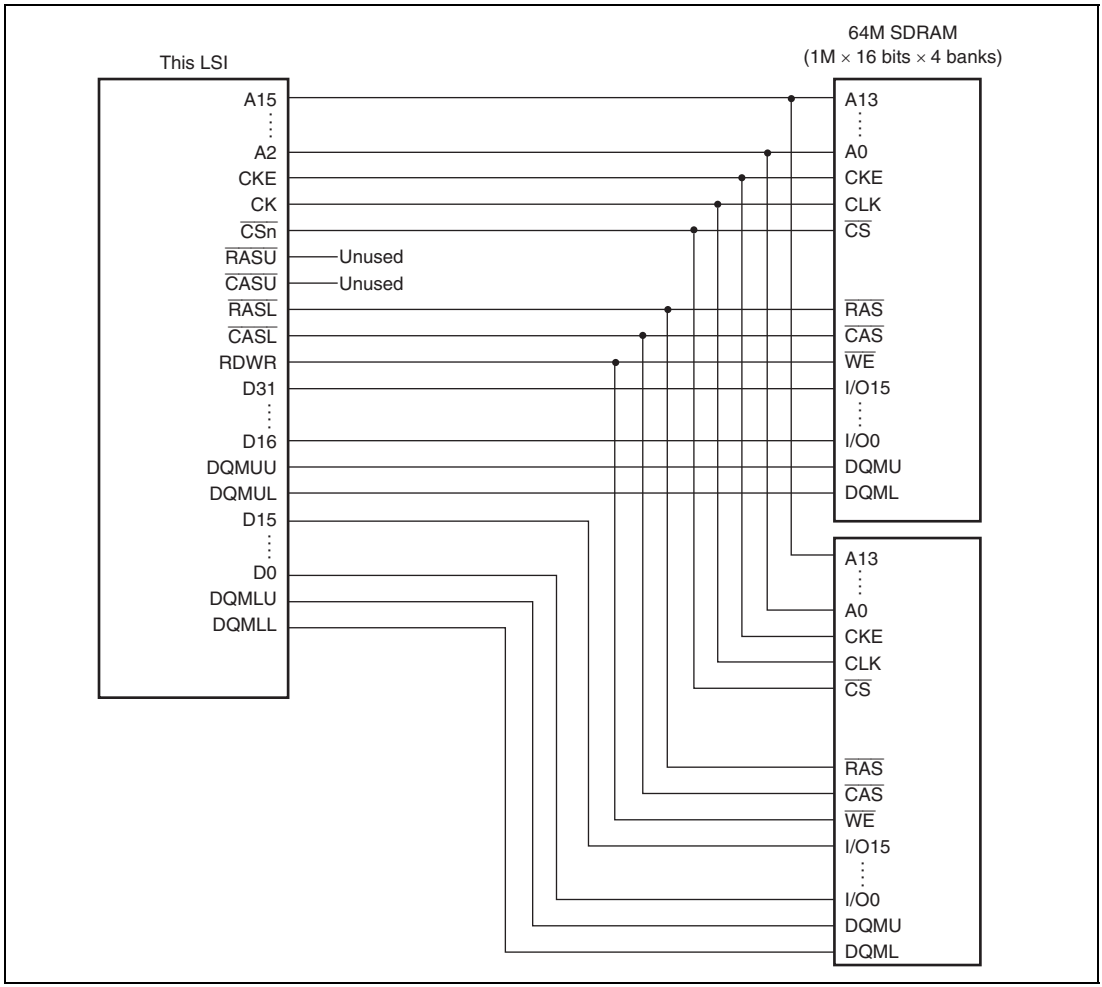


Figure 9.15 Example of 32-Bit Data Width SDRAM Connection ($\overline{\text{RASU}}$ and $\overline{\text{CASU}}$ Are Not Used)

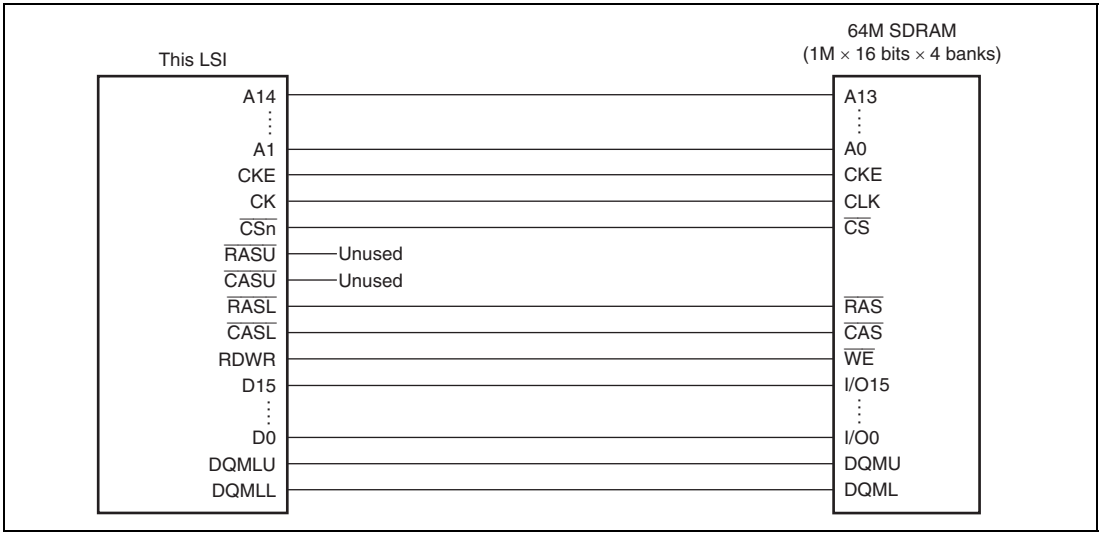


Figure 9.16 Example of 16-Bit Data Width SDRAM Connection ($\overline{\text{RASU}}$ and $\overline{\text{CASU}}$ Are Not Used)

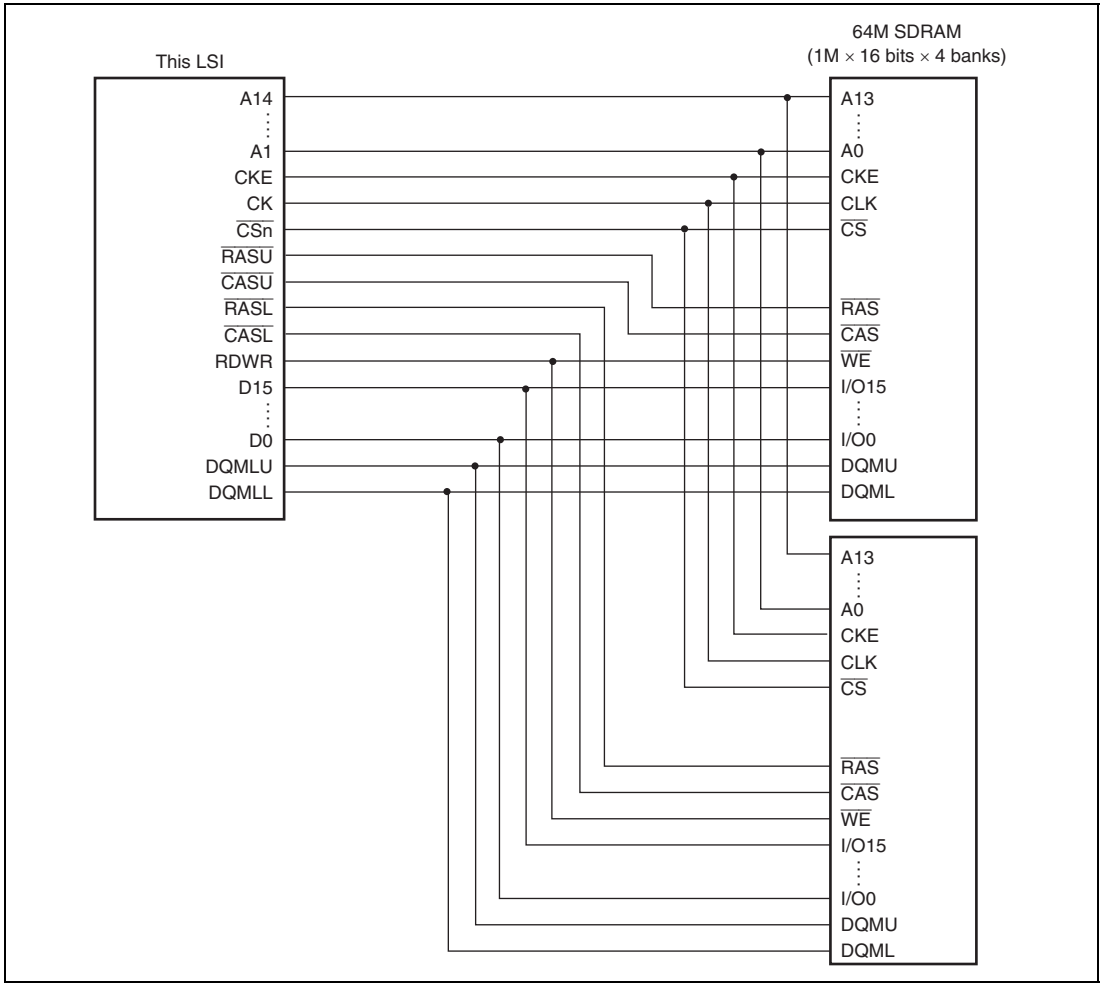


Figure 9.17 Example of 16-Bit Data Width SDRAM Connection ($\overline{\text{RASU}}$ and $\overline{\text{CASU}}$ Are Used)

Address Multiplexing: An address multiplexing is specified so that SDRAM can be connected without external address multiplexing circuitry according to the setting of bits BSZ[1:0] in CSnBCR, and AxROW[1:0] and AxCOL[1:0] in SDCR. Tables 9.20 to 9.25 show the relationship between the settings of bits BSZ[1:0], AxROW[1:0], and AxCOL[1:0] and the bits output at the address pins. Do not specify those bits in the manner other than this table, otherwise the operation of this LSI is not guaranteed. A25 to A18 are not multiplexed and the original values of address are always output at these pins.

When the data bus width is 16 bits (BSZ[1:0] = B'10), the A0 pin of SDRAM specifies a word address. Therefore, connect this A0 pin of SDRAM to the A1 pin of this LSI, then A1 pin to the A2 pin, and so on. When the data bus width is 32 bits (BSZ[1:0] = B'11), the A0 pin of SDRAM specifies a longword address. Therefore, connect this A0 pin of SDRAM to the A2 pin of this LSI, then A1 pin to the A3 pin, and so on.

Table 9.20 Relationship between BSZ[1:0], A2ROW[1:0]/A3ROW[1:0], A2COL[1:0]/A3COL[1:0], and Address Multiplex Output (1)-1

	Setting			
BSZ[1:0]	**A2ROW[1:0]/ A3ROW[1:0]**	**A2COL[1:0]/ A3COL[1:0]**		
11 (32 bits)	**00 (11 bits)**	**00 (8 bits)**		
Output Pin of This LSI	**Row Address Output Cycle**	**Column Address Output Cycle**	**SDRAM Pin**	**Function**
A17	A25	A17		Unused
A16	A24	A16		
A15	A23	A15		
A14	A22*2	A22*2	A12 (BA1)	Specifies bank
A13	A21*2	A21*2	A11 (BA0)	
A12	A20	L/H*1	A10/AP	Specifies address/precharge
A11	A19	A11	A9	Address
A10	A18	A10	A8	
A9	A17	A9	A7	
A8	A16	A8	A6	
A7	A15	A7	A5	
A6	A14	A6	A4	
A5	A13	A5	A3	
A4	A12	A4	A2	
A3	A11	A3	A1	
A2	A10	A2	A0	
A1	A9	A1		Unused
A0	A8	A0		
Example of connected memory				
64-Mbit product (512 kwords × 32 bits × 4 banks, column 8 bits product): 1				
16-Mbit product (512 kwords × 16 bits × 2 banks, column 8 bits product): 2				

Notes: 1. L/H is a bit used in the command specification; it is fixed at low or high according to access mode.
2. Bank address specification

Table 9.20 Relationship between BSZ[1:0], A2ROW[1:0]/A3ROW[1:0], A2COL[1:0]/A3COL[1:0], and Address Multiplex Output (1)-2

	Setting			
BSZ[1:0]	**A2ROW[1:0]/ A3ROW[1:0]**	**A2COL[1:0]/ A3COL[1:0]**		
11 (32 bits)	**01 (12 bits)**	**00 (8 bits)**		
Output Pin of This LSI	**Row Address Output Cycle**	**Column Address Output Cycle**	**SDRAM Pin**	**Function**
A17	A24	A17		Unused
A16	A23	A16		
A15	A23*2	A23*2	A13 (BA1)	Specifies bank
A14	A22*2	A22*2	A12 (BA0)	
A13	A21	A13	A11	Address
A12	A20	L/H*1	A10/AP	Specifies address/precharge
A11	A19	A11	A9	Address
A10	A18	A10	A8	
A9	A17	A9	A7	
A8	A16	A8	A6	
A7	A15	A7	A5	
A6	A14	A6	A4	
A5	A13	A5	A3	
A4	A12	A4	A2	
A3	A11	A3	A1	
A2	A10	A2	A0	
A1	A9	A1		Unused
A0	A8	A0		
Example of connected memory				
128-Mbit product (1 Mword × 32 bits × 4 banks, column 8 bits product): 1				
64-Mbit product (1 Mword × 16 bits × 4 banks, column 8 bits product): 2				

Notes: 1. L/H is a bit used in the command specification; it is fixed at low or high according to access mode.
2. Bank address specification

Table 9.21 Relationship between BSZ[1:0], A2ROW[1:0]/A3ROW[1:0], A2COL[1:0]/A3COL[1:0], and Address Multiplex Output (2)-1

Setting				
BSZ[1:0]	**A2ROW[1:0]/ A3ROW[1:0]**	**A2COL[1:0]/ A3COL[1:0]**		
11 (32 bits)	**01 (12 bits)**	**01 (9 bits)**		
Output Pin of This LSI	**Row Address Output Cycle**	**Column Address Output Cycle**	**SDRAM Pin**	**Function**
A17	A26	A17		Unused
A16	A25	A16		
A15	A24*2	A24*2	A13 (BA1)	Specifies bank
A14	A23*2	A23*2	A12 (BA0)	
A13	A22	A13	A11	Address
A12	A21	L/H*1	A10/AP	Specifies address/precharge
A11	A20	A11	A9	Address
A10	A19	A10	A8	
A9	A18	A9	A7	
A8	A17	A8	A6	
A7	A16	A7	A5	
A6	A15	A6	A4	
A5	A14	A5	A3	
A4	A13	A4	A2	
A3	A12	A3	A1	
A2	A11	A2	A0	
A1	A10	A1		Unused
A0	A9	A0		
Example of connected memory				
256-Mbit product (2 Mwords × 32 bits × 4 banks, column 9 bits product): 1				
128-Mbit product (2 Mwords × 16 bits × 4 banks, column 9 bits product): 2				

Notes: 1. L/H is a bit used in the command specification; it is fixed at low or high according to access mode.
2. Bank address specification

Table 9.21 Relationship between BSZ[1:0], A2ROW[1:0]/A3ROW[1:0], A2COL[1:0]/A3COL[1:0], and Address Multiplex Output (2)-2

Setting				
BSZ[1:0]	**A2ROW[1:0]/ A3ROW[1:0]**	**A2COL[1:0]/ A3COL[1:0]**		
11 (32 bits)	**01 (12 bits)**	**10 (10 bits)**		
Output Pin of This LSI	**Row Address Output Cycle**	**Column Address Output Cycle**	**SDRAM Pin**	**Function**
A17	A27	A17		Unused
A16	A26	A16		
A15	A25*2*3	A25*2*3	A13 (BA1)	Specifies bank
A14	A24*2	A24*2	A12 (BA0)	
A13	A23	A13	A11	Address
A12	A22	L/H*1	A10/AP	Specifies address/precharge
A11	A21	A11	A9	Address
A10	A20	A10	A8	
A9	A19	A9	A7	
A8	A18	A8	A6	
A7	A17	A7	A5	
A6	A16	A6	A4	
A5	A15	A5	A3	
A4	A14	A4	A2	
A3	A13	A3	A1	
A2	A12	A2	A0	
A1	A11	A1		Unused
A0	A10	A0		
Example of connected memory				
512-Mbit product (4 Mwords × 32 bits × 4 banks, column 10 bits product): 1				
256-Mbit product (4 Mwords × 16 bits × 4 banks, column 10 bits product): 2				

Notes: 1. L/H is a bit used in the command specification; it is fixed at low or high according to access mode.
2. Bank address specification
3. Only the $\overline{\text{RASL}}$ pin is asserted because the A25 pin specifies the bank address. $\overline{\text{RASU}}$ is not asserted.

Table 9.22 Relationship between BSZ[1:0], A2ROW[1:0]/A3ROW[1:0], A2COL[1:0]/A3COL[1:0], and Address Multiplex Output (3)

Setting				
BSZ[1:0]	**A2ROW[1:0]/ A3ROW[1:0]**	**A2COL[1:0]/ A3COL[1:0]**		
11 (32 bits)	**10 (13 bits)**	**01 (9 bits)**		
Output Pin of This LSI	**Row Address Output Cycle**	**Column Address Output Cycle**	**SDRAM Pin**	**Function**
A17	A26	A17		Unused
A16	A25*2*3	A25*2*3	A14 (BA1)	Specifies bank
A15	A24*2	A24*2	A13 (BA0)	
A14	A23	A14	A12	Address
A13	A22	A13	A11	
A12	A21	L/H*1	A10/AP	Specifies address/precharge
A11	A20	A11	A9	Address
A10	A19	A10	A8	
A9	A18	A9	A7	
A8	A17	A8	A6	
A7	A16	A7	A5	
A6	A15	A6	A4	
A5	A14	A5	A3	
A4	A13	A4	A2	
A3	A12	A3	A1	
A2	A11	A2	A0	
A1	A10	A1		Unused
A0	A9	A0		
Example of connected memory				
512-Mbit product (4 Mwords × 32 bits × 4 banks, column 9 bits product): 1				
256-Mbit product (4 Mwords × 16 bits × 4 banks, column 9 bits product): 2				

Notes: 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.
2. Bank address specification
3. Only the $\overline{\text{RASL}}$ pin is asserted because the A 25 pin specifies the bank address. $\overline{\text{RASU}}$ is not asserted.

Table 9.23 Relationship between BSZ[1:0], A2ROW[1:0]/A3ROW[1:0], A2COL[1:0]/A3COL[1:0], and Address Multiplex Output (4)-1

Setting				
BSZ[1:0]	**A2ROW[1:0]/ A3ROW[1:0]**	**A2COL[1:0]/ A3COL[1:0]**		
10 (16 bits)	**00 (11 bits)**	**00 (8 bits)**		
Output Pin of This LSI	**Row Address Output Cycle**	**Column Address Output Cycle**	**SDRAM Pin**	**Function**
A17	A25	A17		Unused
A16	A24	A16		
A15	A23	A15		
A14	A22	A14		
A13	A21*2	A21*2	A12 (BA1)	Specifies bank
A12	A20*2	A20*2	A11 (BA0)	
A11	A19	L/H*1	A10/AP	Specifies address/precharge
A10	A18	A10	A9	Address
A9	A17	A9	A8	
A8	A16	A8	A7	
A7	A15	A7	A6	
A6	A14	A6	A5	
A5	A13	A5	A4	
A4	A12	A4	A3	
A3	A11	A3	A2	
A2	A10	A2	A1	
A1	A9	A1	A0	
A0	A8	A0		Unused
Example of connected memory				
16-Mbit product (512 kwords × 16 bits × 2 banks, column 8 bits product): 1				

Notes: 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.
2. Bank address specification

Table 9.23 Relationship between BSZ[1:0], A2ROW[1:0]/A3ROW[1:0], A2COL[1:0]/A3COL[1:0], and Address Multiplex Output (4)-2

	Setting			
BSZ[1:0]	**A2ROW[1:0]/ A3ROW[1:0]**	**A2COL[1:0]/ A3COL[1:0]**		
10 (16 bits)	**01 (12 bits)**	**00 (8 bits)**		
Output Pin of This LSI	**Row Address Output Cycle**	**Column Address Output Cycle**	**SDRAM Pin**	**Function**
A17	A25	A17		Unused
A16	A24	A16		
A15	A23	A15		
A14	A22*[2]	A22*[2]	A13 (BA1)	Specifies bank
A13	A21*[2]	A21*[2]	A12 (BA0)	
A12	A20	A12	A11	Address
A11	A19	L/H*[1]	A10/AP	Specifies address/precharge
A10	A18	A10	A9	Address
A9	A17	A9	A8	
A8	A16	A8	A7	
A7	A15	A7	A6	
A6	A14	A6	A5	
A5	A13	A5	A4	
A4	A12	A4	A3	
A3	A11	A3	A2	
A2	A10	A2	A1	
A1	A9	A1	A0	
A0	A8	A0		Unused
Example of connected memory				
64-Mbit product (1 Mword × 16 bits × 4 banks, column 8 bits product): 1				

Notes: 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.
2. Bank address specification

Table 9.24 Relationship between BSZ[1:0], A2ROW[1:0]/A3ROW[1:0], A2COL[1:0]/A3COL[1:0], and Address Multiplex Output (5)-1

Setting				
BSZ[1:0]	**A2ROW[1:0]/ A3ROW[1:0]**	**A2COL[1:0]/ A3COL[1:0]**		
10 (16 bits)	**01 (12 bits)**	**01 (9 bits)**		
Output Pin of This LSI	**Row Address Output Cycle**	**Column Address Output Cycle**	**SDRAM Pin**	**Function**
A17	A26	A17		Unused
A16	A25	A16		
A15	A24	A15		
A14	A23*2	A23*2	A13 (BA1)	Specifies bank
A13	A22*2	A22*2	A12 (BA0)	
A12	A21	A12	A11	Address
A11	A20	L/H*1	A10/AP	Specifies address/precharge
A10	A19	A10	A9	Address
A9	A18	A9	A8	
A8	A17	A8	A7	
A7	A16	A7	A6	
A6	A15	A6	A5	
A5	A14	A5	A4	
A4	A13	A4	A3	
A3	A12	A3	A2	
A2	A11	A2	A1	
A1	A10	A1	A0	
A0	A9	A0		Unused
Example of connected memory				
128-Mbit product (2 Mwords × 16 bits × 4 banks, column 9 bits product): 1				

Notes: 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.
2. Bank address specification

Table 9.24 Relationship between BSZ[1:0], A2ROW[1:0]/A3ROW[1:0], A2COL[1:0]/A3COL[1:0], and Address Multiplex Output (5)-2

Setting				
BSZ[1:0]	**A2ROW[1:0]/ A3ROW[1:0]**	**A2COL[1:0]/ A3COL[1:0]**		
10 (16 bits)	**01 (12 bits)**	**10 (10 bits)**		
Output Pin of This LSI	**Row Address Output Cycle**	**Column Address Output Cycle**	**SDRAM Pin**	**Function**
A17	A27	A17		Unused
A16	A26	A16		
A15	A25	A15		
A14	A24*[2]	A24*[2]	A13 (BA1)	Specifies bank
A13	A23*[2]	A23*[2]	A12 (BA0)	
A12	A22	A12	A11	Address
A11	A21	L/H*[1]	A10/AP	Specifies address/precharge
A10	A20	A10	A9	Address
A9	A19	A9	A8	
A8	A18	A8	A7	
A7	A17	A7	A6	
A6	A16	A6	A5	
A5	A15	A5	A4	
A4	A14	A4	A3	
A3	A13	A3	A2	
A2	A12	A2	A1	
A1	A11	A1	A0	
A0	A10	A0		Unused
Example of connected memory				
256-Mbit product (4 Mwords × 16 bits × 4 banks, column 10 bits product): 1				

Notes: 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.
2. Bank address specification

Table 9.25 Relationship between BSZ[1:0], A2ROW[1:0]/A3ROW[1:0], A2COL[1:0]/A3COL[1:0], and Address Multiplex Output (6)-1

Setting				
BSZ[1:0]	**A2ROW[1:0]/ A3ROW[1:0]**	**A2COL[1:0]/ A3COL[1:0]**		
10 (16 bits)	**10 (13 bits)**	**01 (9 bits)**		
Output Pin of This LSI	**Row Address Output Cycle**	**Column Address Output Cycle**	**SDRAM Pin**	**Function**
A17	A26	A17		Unused
A16	A25	A16		
A15	A24*2	A24*2	A14 (BA1)	Specifies bank
A14	A23*2	A23*2	A13 (BA0)	
A13	A22	A13	A12	Address
A12	A21	A12	A11	
A11	A20	L/H*1	A10/AP	Specifies address/precharge
A10	A19	A10	A9	Address
A9	A18	A9	A8	
A8	A17	A8	A7	
A7	A16	A7	A6	
A6	A15	A6	A5	
A5	A14	A5	A4	
A4	A13	A4	A3	
A3	A12	A3	A2	
A2	A11	A2	A1	
A1	A10	A1	A0	
A0	A9	A0		Unused
Example of connected memory				
256-Mbit product (4 Mwords × 16 bits × 4 banks, column 9 bits product): 1				

Notes: 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.
2. Bank address specification

Table 9.25 Relationship between BSZ[1:0], A2ROW[1:0]/A3ROW[1:0], A2COL[1:0]/A3COL[1:0], and Address Multiplex Output (6)-2

	Setting			
BSZ[1:0]	**A2ROW[1:0]/ A3ROW[1:0]**	**A2COL[1:0]/ A3COL[1:0]**		
10 (16 bits)	**10 (13 bits)**	**10 (10 bits)**		
Output Pin of This LSI	**Row Address Output Cycle**	**Column Address Output Cycle**	**SDRAM Pin**	**Function**
A17	A27	A17		Unused
A16	A26	A16		
A15	A25*2*3	A25*2*3	A14 (BA1)	Specifies bank
A14	A24*2	A24*2	A13 (BA0)	
A13	A23	A13	A12	Address
A12	A22	A12	A11	
A11	A21	L/H*1	A10/AP	Specifies address/precharge
A10	A20	A10	A9	Address
A9	A19	A9	A8	
A8	A18	A8	A7	
A7	A17	A7	A6	
A6	A16	A6	A5	
A5	A15	A5	A4	
A4	A14	A4	A3	
A3	A13	A3	A2	
A2	A12	A2	A1	
A1	A11	A1	A0	
A0	A10	A0		Unused
Example of connected memory				
512-Mbit product (8 Mwords × 16 bits × 4 banks, column 10 bits product): 1				

Notes: 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.
2. Bank address specification
3. Only the $\overline{\text{RASL}}$ pin is asserted because the A25 pin specifies the bank address. $\overline{\text{RASU}}$ is not asserted.

Burst Read: A burst read occurs in the following cases with this LSI.

- Access size in reading is larger than data bus width
- 16-byte transfer in DMAC

This LSI always accesses SDRAM with burst length 1. For example, read access of burst length 1 is performed consecutively 4 times to read 16-byte continuous data from SDRAM that is connected to a 32-bit data bus. This access is called number of bursts 4.

Table 9.26 shows the relationship between the access size and the number of bursts.

Table 9.26 Relationship between Access Size and Number of Bursts

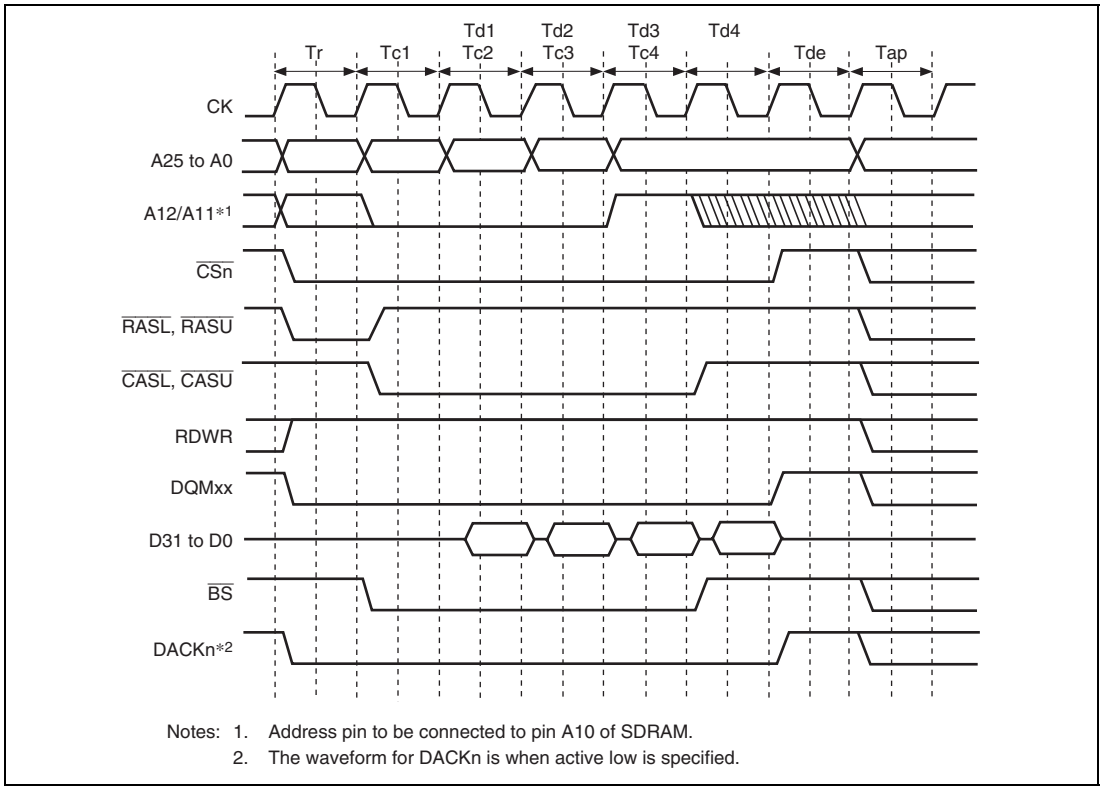
Bus Width	Access Size	Number of Bursts
16 bits	8 bits	1
	16 bits	1
	32 bits	2
	16 bytes	8
32 bits	8 bits	1
	16 bits	1
	32 bits	1
	16 bytes	4

Figures 9.18 and 9.19 show a timing chart in burst read. In burst read, an ACTV command is output in the Tr cycle, the READ command is issued in the Tc1 to Tc3 cycles, the READA command is issued in the Tc4 cycle, and the read data is received at the rising edge of the external clock (CK) in the Td1 to Td4 cycles. The Tap cycle is used to wait for the completion of an auto-precharge induced by the READ command in the SDRAM. In the Tap cycle, a new command will not be issued to the same bank. However, access to another CS space or another bank in the same SDRAM space is enabled. The number of Tap cycles is specified by the WTRP1 and WTRP0 bits in CS3WCR.

In this LSI, wait cycles can be inserted by specifying each bit in CSnWCR to connect the SDRAM in variable frequencies. Figure 9.19 shows an example in which wait cycles are inserted. The number of cycles from the Tr cycle where the ACTV command is output to the Tc1 cycle where the READA command is output can be specified using the WTRCD1 and WTRCD0 bits in CS3WCR. If the WTRCD1 and WTRCD0 bits specify one cycle or more, a Trw cycle where the NOP command is issued is inserted between the Tr cycle and Tc1 cycle. The number of cycles from the Tc1 cycle where the READA command is output to the Td1 cycle where the read data is

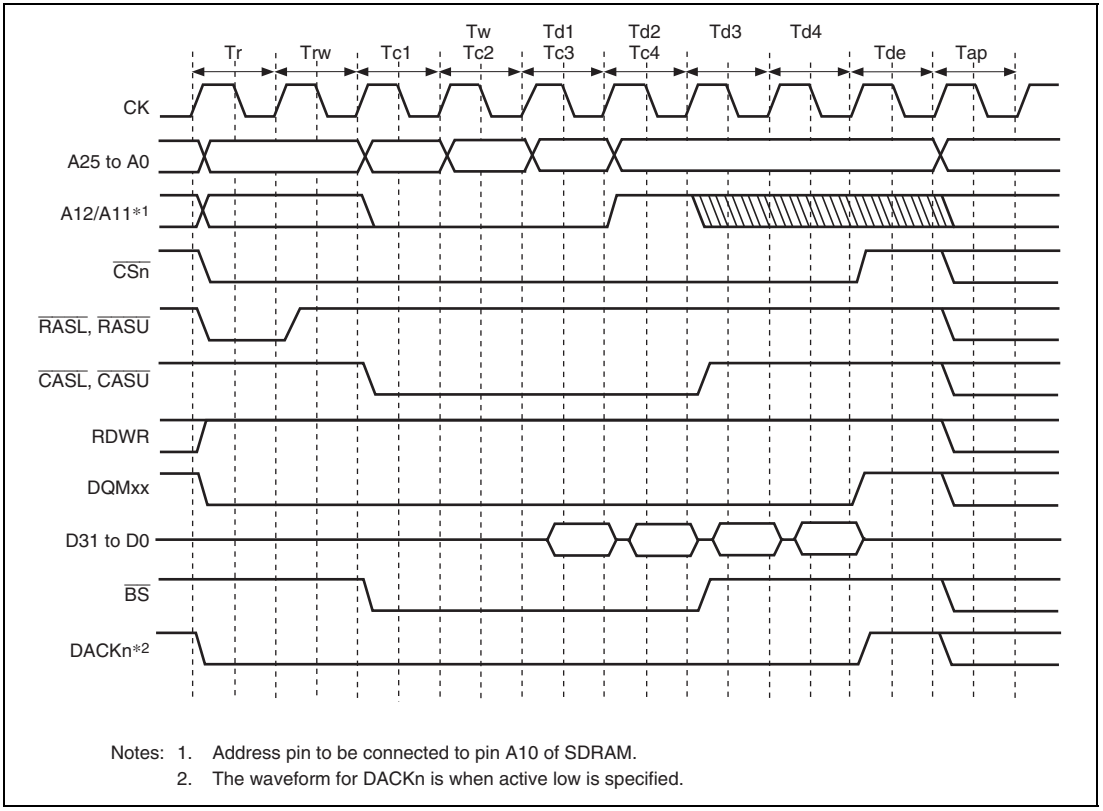
latched can be specified for the CS2 and CS3 spaces independently, using the A2CL1 and A2CL0 bits in CS2WCR or the A3CL1 and A3CL0 bits in CS3WCR. The number of cycles from Tc1 to Td1 corresponds to the synchronous DRAM CAS latency. The CAS latency for the synchronous DRAM is normally defined as up to three cycles. However, the CAS latency in this LSI can be specified as 1 to 4 cycles. This CAS latency can be achieved by connecting a latch circuit between this LSI and the SDRAM.

A Tde cycle is an idle cycle required to transfer the read data into this LSI and occurs once for every burst read or every single read.



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 9.18 Burst Read Basic Timing (Auto-Precharge)

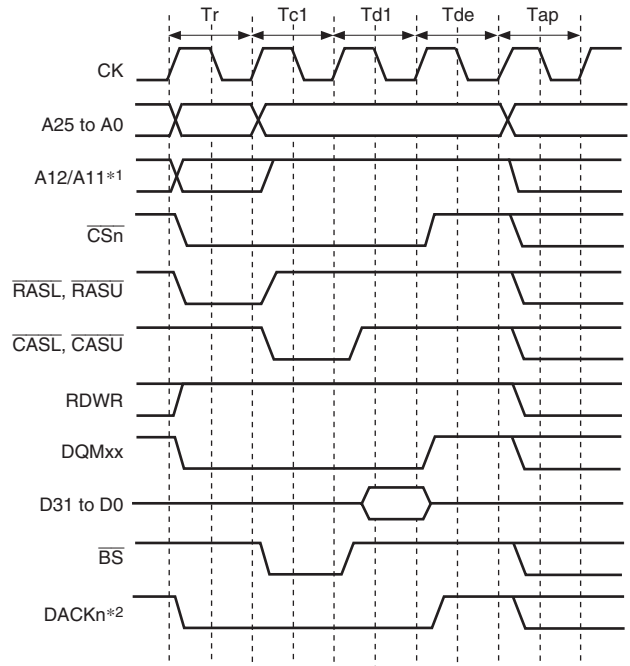


Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 9.19 Burst Read Wait Specification Timing (Auto-Precharge)

Single Read: A read access ends in one cycle when the data bus width is larger than or equal to access size. This is called single read. As the burst length is set to 1 in SDRAM burst read/single write mode, only the required data is output. Consequently, no unnecessary bus cycles are generated.

Figure 9.20 shows the single read basic timing.



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

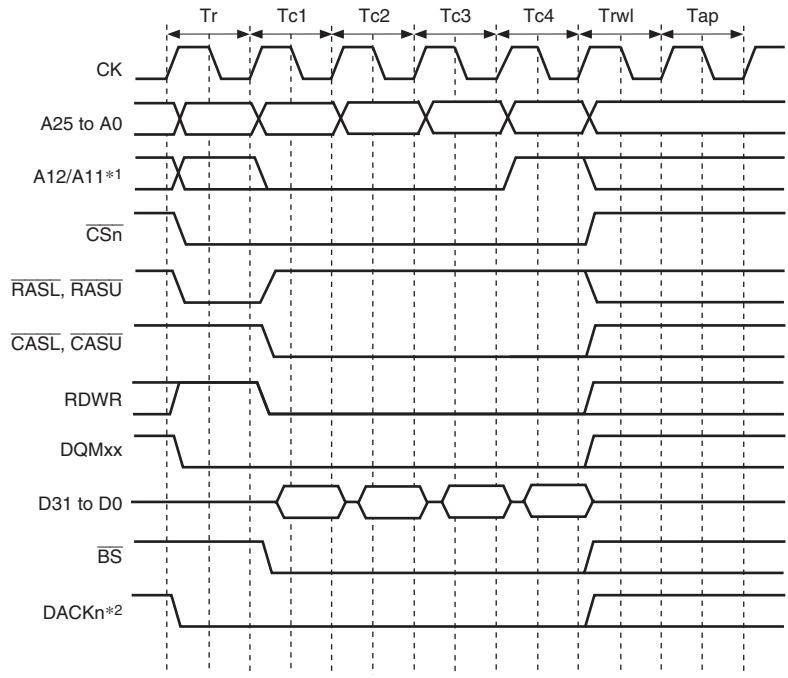
Figure 9.20 Single Read Basic Timing (Auto-Precharge)

Burst Write: A burst write occurs in the following cases in this LSI.

- Access size in writing is larger than data bus width
- 16-byte transfer in DMAC

This LSI always accesses SDRAM with burst length 1. For example, write access of burst length 1 is performed continuously 4 times to write 16-byte continuous data to the SDRAM that is connected to a 32-bit data bus. The relationship between the access size and the number of bursts is shown in table 9.26.

Figure 9.21 shows a timing chart for burst writes. In burst write, an ACTV command is output in the Tr cycle, the WRIT command is issued in the Tc1 to Tc3 cycles, and the WRITA command is issued to execute an auto-precharge in the Tc4 cycle. In the write cycle, the write data is output simultaneously with the write command. After the write command with the auto-precharge is output, the Trw1 cycle that waits for the auto-precharge initiation is followed by the Tap cycle that waits for completion of the auto-precharge induced by the WRITA command in the SDRAM. Between the Trwl and Tap cycles, a new command will not be issued to the same bank. However, access to another CS space or another bank in the same SDRAM space is enabled. The number of cycles in a Trw1 cycle is specified by the TRWL1 and TRWL0 bits in CS3WCR. The number of cycles in a Tap cycle is specified by the WTRP1 and WTRP0 bits in CS3WCR.

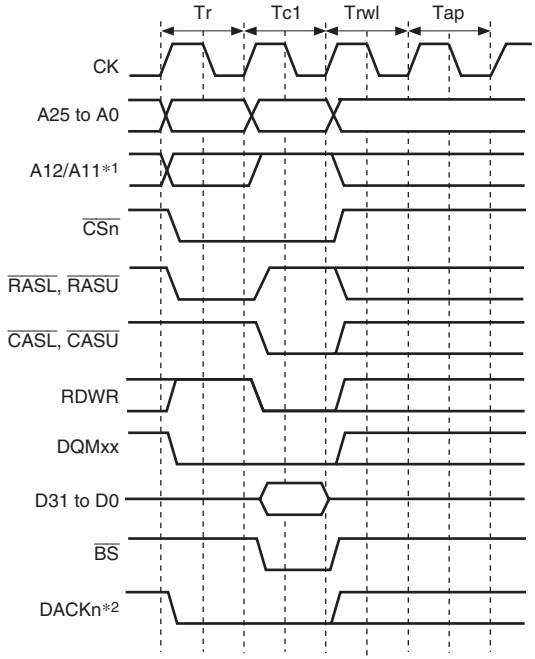


Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 9.21 Basic Timing for SDRAM Burst Write (Auto-Precharge)

Single Write: A write access ends in one cycle when the data bus width is larger than or equal to access size. This is called single write.

Figure 9.22 shows the single write basic timing.



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 9.22 Single Write Basic Timing (Auto-Precharge)

Bank Active: The SDRAM bank function is used to support high-speed accesses to the same row address. When the BACTV bit in SDCR is 1, accesses are performed using commands without auto-precharge (READ or WRIT). This function is called bank-active function. This function is valid only for area 3. When area 3 is set to bank-active mode, area 2 should be set to normal space or SRAM with byte selection. When areas 2 and 3 are both set to SDRAM, auto-precharge mode must be set.

In this case, precharging is not performed when the access ends. When accessing the same row address in the same bank, it is possible to issue the READ or WRIT command immediately, without issuing an ACTV command. As SDRAM is internally divided into several banks, it is possible to activate one row address in each bank. If the next access is to a different row address, a PRE command is first issued to precharge the relevant bank, then when precharging is completed,

the access is performed by issuing an ACTV command followed by a READ or WRIT command. If this is followed by an access to a different row address, the access time will be longer because of the precharging performed after the access request is issued. The number of cycles between issuance of the PRE command and the ACTV command is determined by the WTRP1 and WTRP0 bits in CS3WCR.

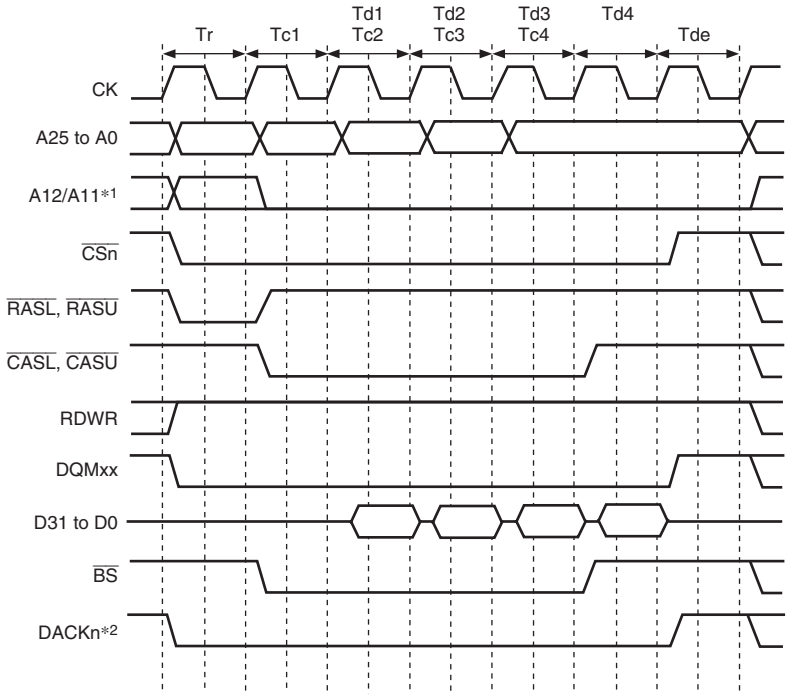
In a write, when an auto-precharge is performed, a command cannot be issued to the same bank for a period of Trwl + Tap cycles after issuance of the WRITA command. When bank active mode is used, READ or WRIT commands can be issued successively if the row address is the same. The number of cycles can thus be reduced by Trwl + Tap cycles for each write.

There is a limit on tRAS, the time for placing each bank in the active state. If there is no guarantee that another row address will be accessed within the period in which this value is maintained by program execution, it is necessary to set the refresh cycle to no more than the value of tRAS.

A burst read cycle without auto-precharge is shown in figure 9.23, a burst read cycle for the same row address in figure 9.24, and a burst read cycle for different row addresses in figure 9.25. Similarly, a single write cycle without auto-precharge is shown in figure 9.26, a single write cycle for the same row address in figure 9.27, and a single write cycle for different row addresses in figure 9.28.

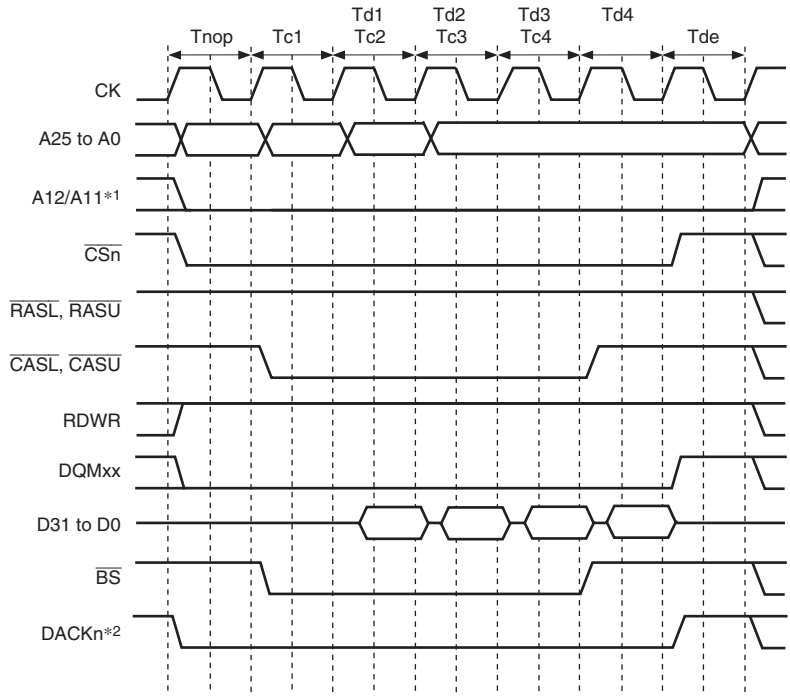
In figure 9.24, a Tnop cycle in which no operation is performed is inserted before the Tc cycle that issues the READ command. The Tnop cycle is inserted to acquire two cycles of CAS latency for the DQMxx signal that specifies the read byte in the data read from the SDRAM. If the CAS latency is specified as two cycles or more, the Tnop cycle is not inserted because the two cycles of latency can be acquired even if the DQMxx signal is asserted after the Tc cycle.

When bank active mode is set, if only accesses to the respective banks in area 3 are considered, as long as accesses to the same row address continue, the operation starts with the cycle in figure 9.23 or 9.26, followed by repetition of the cycle in figure 9.24 or 9.27. An access to a different area or bank during this time has no effect. If there is an access to a different row address in the bank active state, after this is detected the bus cycle in figure 9.24 or 9.27 is executed instead of that in figure 9.25 or 9.28. In bank active mode, too, all banks become inactive after a refresh cycle or after the bus is released as the result of bus arbitration.



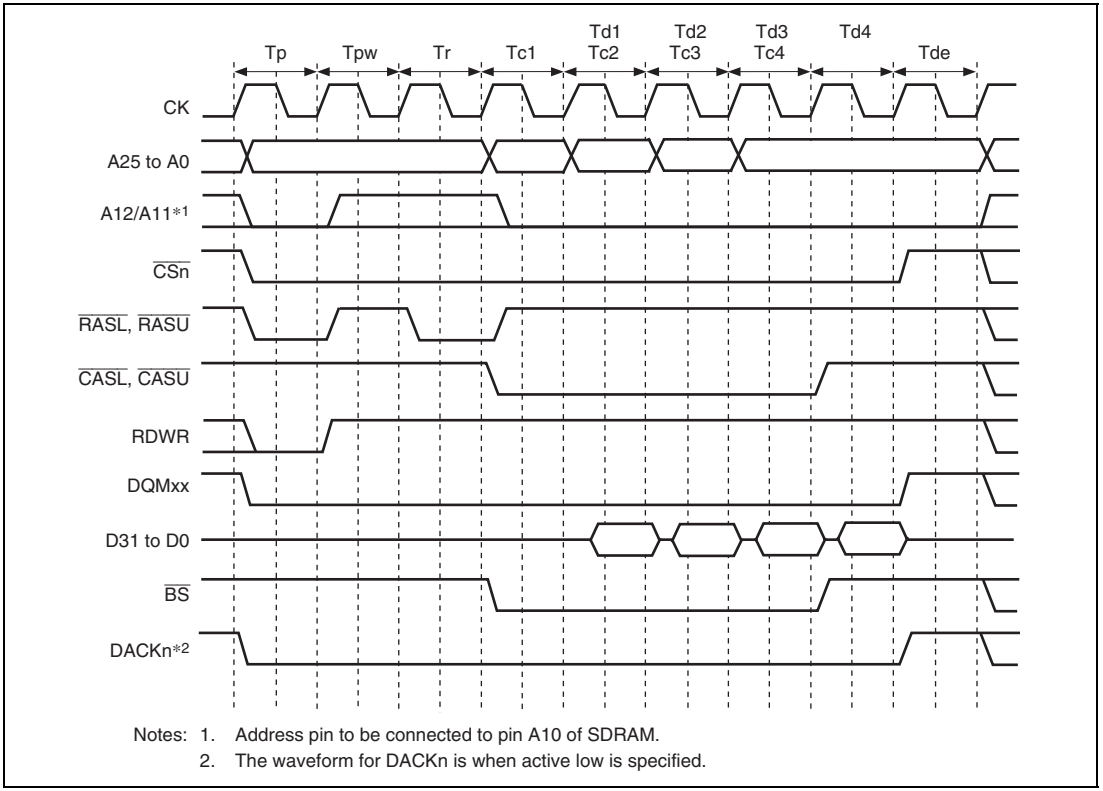
Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 9.23 Burst Read Timing (No Auto-Precharge)



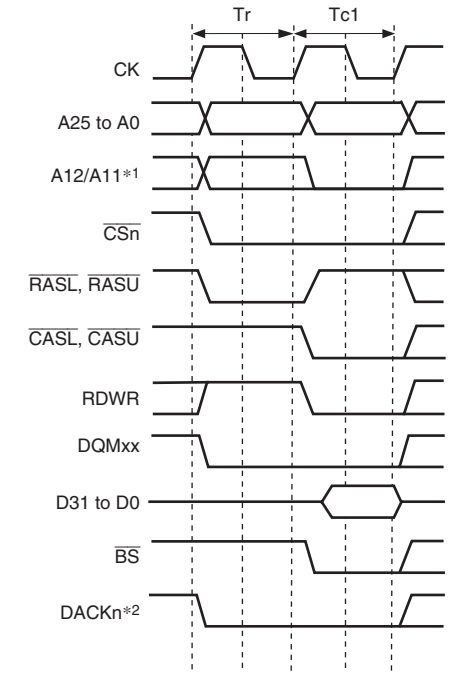
Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 9.24 Burst Read Timing (Bank Active, Same Row Address)



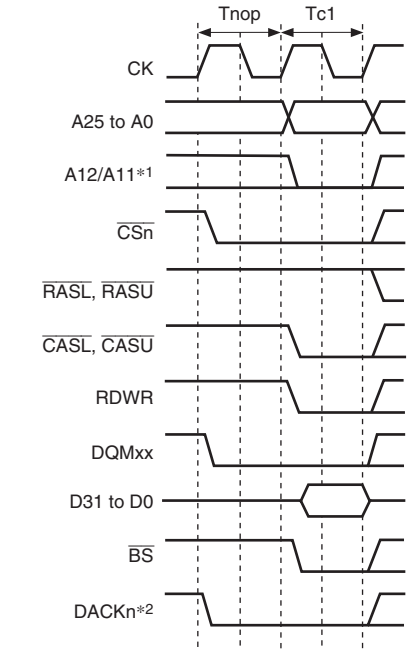
Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 9.25 Burst Read Timing (Bank Active, Different Row Addresses)



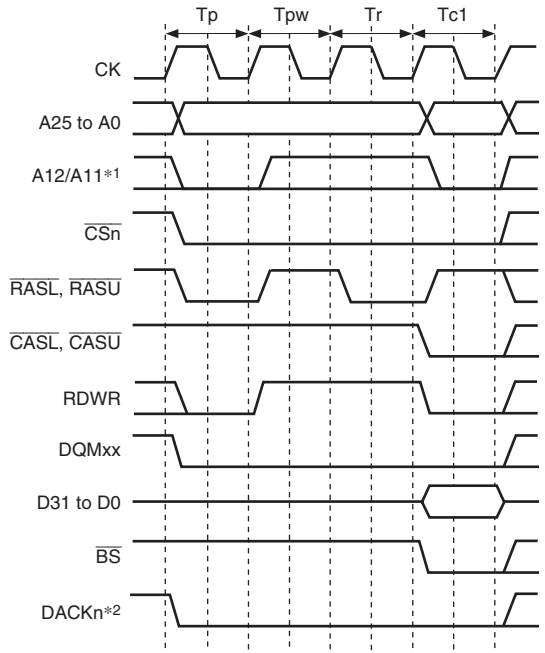
Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 9.26 Single Write Timing (No Auto-Precharge)



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 9.27 Single Write Timing (Bank Active, Same Row Address)



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

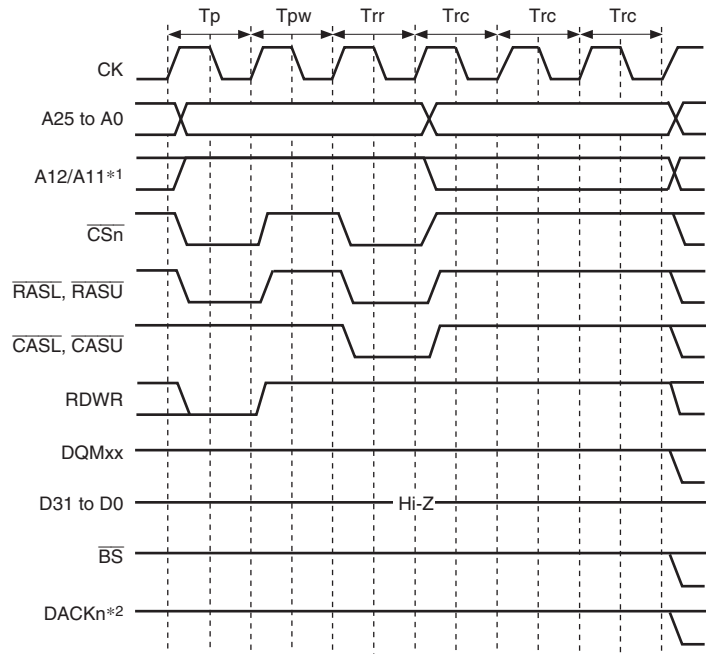
Figure 9.28 Single Write Timing (Bank Active, Different Row Addresses)

Refreshing: This LSI has a function for controlling SDRAM refreshing. Auto-refreshing can be performed by clearing the RMODE bit to 0 and setting the RFSH bit to 1 in SDCR. A continuous refreshing can be performed by setting the RRC[2:0] bits in RTCSR. If SDRAM is not accessed for a long period, self-refresh mode, in which the power consumption is low, can be activated by setting both the RMODE bit and the RFSH bit to 1.

1. Auto-refreshing

   The number of refreshings set by bits RRC[2:0] in RTCSR is performed at intervals determined by the input clock selected by bits CKS[2:0] in RTCSR, and the value set in RTCOR. Register settings should be made so as to satisfy the refresh interval stipulation for the SDRAM used. First make the settings for RTCOR, RTCNT, and the RMODE and RFSH bits in SDCR, then make the CKS[2:0] and RRC[2:0] settings. When the clock is selected by bits CKS[2:0], RTCNT starts counting up from the value at that time. The RTCNT value is constantly compared with the RTCOR value, and if the two values are the same, a refresh request is generated and an auto-refresh is performed for the number of times specified by the RRC[2:0]. At the same time, RTCNT is cleared to 0 and the count-up is restarted.

   Figure 9.29 shows the auto-refresh cycle timing. After starting auto-refreshing, PALL command is issued in the Tp cycle to make all the banks to precharged state from active state when some bank is being precharged. Then REF command is issued in the Trr cycle after inserting idle cycles of which number is specified by the WTRP[1:0] bits in CS3WCR. A new command is not issued for the duration of the number of cycles specified by the WTRC[1:0] bits in CS3WCR after the Trr cycle. The WTRC[1:0] bits must be set so as to satisfy the SDRAM refreshing cycle time stipulation (tRC). An idle cycle is inserted between the Tp cycle and Trr cycle when the setting value of the WTRP[1:0] bits in CS3WCR is one cycle or more.



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 9.29 Auto-Refresh Timing

2. Self-refreshing

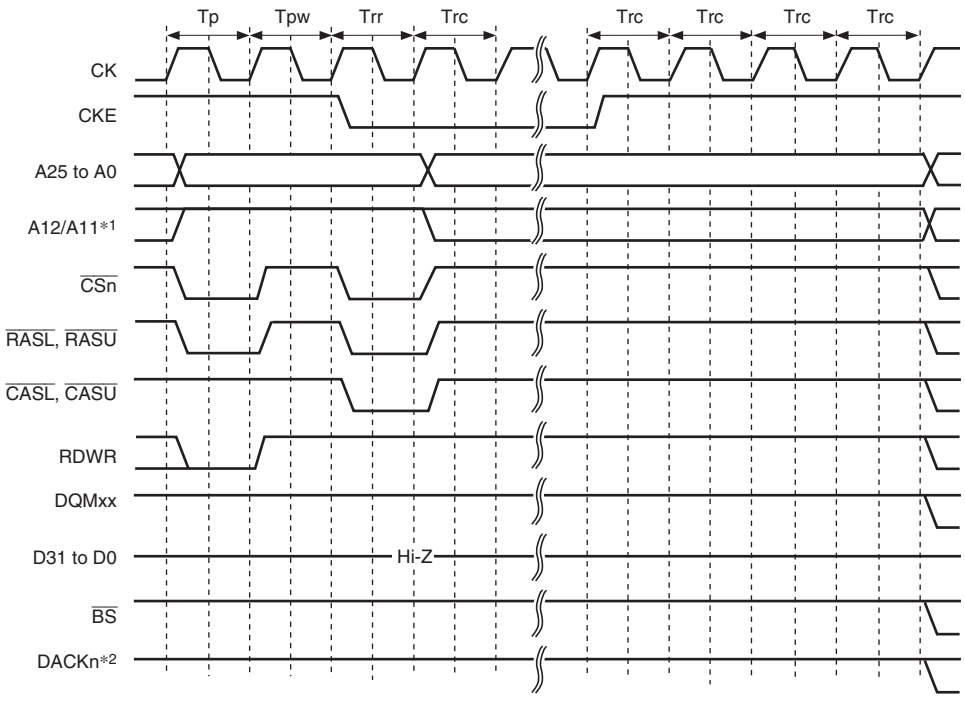
Self-refresh mode is a kind of standby mode in which the refresh timing and refresh addresses are generated within SDRAM. Self-refreshing is activated by setting both the RMODE bit and the RFSH bit in SDCR to 1. After starting the self-refreshing, PALL command is issued in the Tp cycle after the completion of the precharging bank. A SELF command is then issued after inserting idle cycles of which number is specified by the WTRP[1:0] bits in CS3WCR. SDRAM cannot be accessed while in the self-refresh state. Self-refresh mode is cleared by clearing the RMODE bit to 0. After self-refresh mode has been cleared, command issuance is disabled for the number of cycles specified by the WTRC[1:0] bits in CS3WCR.

Self-refresh timing is shown in figure 9.30. After self-refreshing is cleared, settings must be made so that auto-refreshing is performed at the correct intervals. When self-refreshing is activated from the state in which auto-refreshing is set, auto-refreshing is restarted if the RFSH bit is set to 1 and the RMODE bit is cleared to 0 when self-refresh mode is cleared. If the transition from clearing of self-refresh mode to the start of auto-refreshing takes time, making the RTCNT value 1 less than the RTCOR value will enable auto-refreshing to be started immediately.

After self-refreshing has been specified, the SDRAM stays in the self-refresh state even after this LSI enters the standby state. The self-refresh state continues after recovery from the standby state by interrupt. However, the CKE and other pins must be driven in the standby state by setting the HIZCNT bit in the CMNCR register to 1.

The self-refresh state is not cleared by a manual reset.

In case of a power-on reset, the bus state controller's registers are initialized, and therefore the self-refresh state is cleared.



Notes: 1. Address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn is when active low is specified.

Figure 9.30 Self-Refresh Timing

Relationship between Refresh Requests and Bus Cycles: If a refresh request occurs during bus cycle execution, the refresh cycle must wait for the bus cycle to be completed. If a refresh request occurs while the bus is released by the bus arbitration function, the refresh will not be executed until the bus mastership is acquired. This LSI has the $\overline{\text{IRQOUT}}$ pin to request the bus while waiting for refresh execution. This LSI continues to assert $\overline{\text{IRQOUT}}$ (low level) until the bus is acquired.

If a new refresh request occurs while waiting for the previous refresh request, the previous refresh request is deleted. To refresh correctly, a bus cycle longer than the refresh interval or the bus mastership occupation must be prevented from occurring.

If a bus mastership is requested during self-refresh, the bus will not be released until the self-refresh is cleared.

Power-On Sequence: In order to use SDRAM, mode setting must first be made for SDRAM after powering on. To perform SDRAM initialization correctly, the bus state controller registers must first be set, followed by a write to the SDRAM mode register. In SDRAM mode register setting, the address signal value at that time is latched by a combination of the $\overline{\text{CSn}}$, $\overline{\text{RASU}}$, $\overline{\text{RASL}}$, $\overline{\text{CASU}}$, $\overline{\text{CASL}}$, and RDWR signals. If the value to be set is X, the bus state controller provides for value X to be written to the SDRAM mode register by performing a word-write to address H'FFF84000 + X for area 2 SDRAM, and to address H'FFF85000 + X for area 3 SDRAM. In this operation, the write data is ignored. To set burst read/single write (burst length 1), burst read/burst write (burst length 1), CAS latency 2 and 3, wrap type = sequential, and burst length 1 supported by the LSI, arbitrary data is written in a word-size access to the addresses shown in table 9.27. In this time, 0 is output at the external address pins of A12 and later.

Table 9.27 Access Address in SDRAM Mode Register Write

- Setting for Area 2 (SDMR2)

  Burst read/single write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'FFF84440	H'0000440
	3	H'FFF84460	H'0000460
32 bits	2	H'FFF84880	H'0000880
	3	H'FFF848C0	H'00008C0

  Burst read/burst write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'FFF84040	H'0000040
	3	H'FFF84060	H'0000060
32 bits	2	H'FFF84080	H'0000080
	3	H'FFF840C0	H'00000C0

- Setting for Area 3 (SDMR3)

  Burst read/single write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'FFF85440	H'0000440
	3	H'FFF85460	H'0000460
32 bits	2	H'FFF85880	H'0000880
	3	H'FFF858C0	H'00008C0

  Burst read/burst write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'FFF85040	H'0000040
	3	H'FFF85060	H'0000060
32 bits	2	H'FFF85080	H'0000080
	3	H'FFF850C0	H'00000C0

Mode register setting timing is shown in figure 9.31. A PALL command (all bank precharge command) is firstly issued. An REF command (auto-refresh command) is then issued 8 times. An MRS command (mode register write command) is finally issued. Idle cycles, of which number is specified by the WTRP[1:0] bits in CS3WCR, are inserted between the PALL and the first REF. Idle cycles, of which number is specified by the WTRC[1:0] bits in CS3WCR, are inserted between REF and REF, and between the 8th REF and MRS. Idle cycles, of which number is one or more, are inserted between the MRS and a command to be issued next.

It is necessary to keep idle time of certain cycles for SDRAM after power-on before issuing PALL command. Refer to the manual of the SDRAM for the idle time to be needed. When the pulse width of the reset signal is longer than the idle time, mode register setting can be started immediately after the reset, but care should be taken when the pulse width of the reset signal is shorter than the idle time.

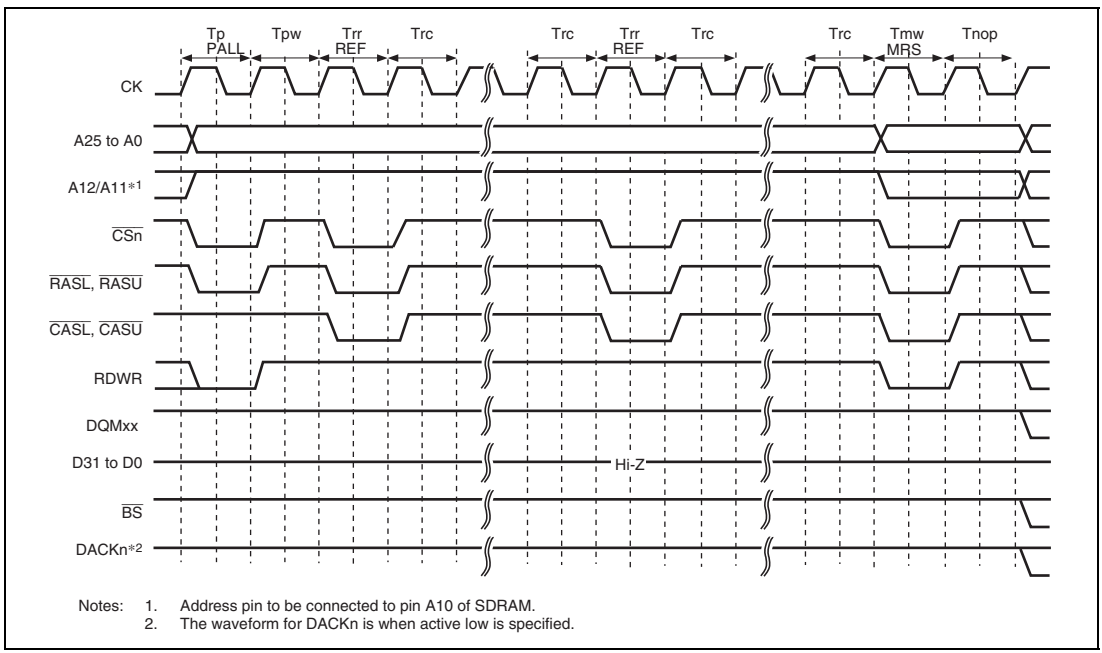


Figure 9.31 SDRAM Mode Register Write Timing (Based on JEDEC)

9.5.7 Burst ROM (Clock Asynchronous) Interface

The burst ROM (clock asynchronous) interface is used to access a memory with a high-speed read function using a method of address switching called burst mode or page mode. In the burst ROM (clock asynchronous) interface, basically the same access as the normal space is performed, but the 2nd and subsequent accesses are performed only by changing the address, without negating the $\overline{RD}$ signal at the end of the first cycle. In the second and subsequent accesses, addresses are changed at the falling edge of the CK.

For the first access cycle, the number of wait cycles specified by the W[3:0] bits in CSnWCR is inserted. For the second and subsequent access cycles, the number of wait cycles specified by the BW[1:0] bits in CSnWCR is inserted.

In the access to the burst ROM (clock asynchronous), the $\overline{BS}$ signal is asserted only to the first access cycle. An external wait input is valid only to the first access cycle.

In the single access that does not perform the burst operation in the burst ROM (clock asynchronous) interface, access timing is the same as a normal space.

Table 9.28 lists the relationship between bus width, access size, and the number of bursts. Figure 9.32 shows a timing chart.

Table 9.28 Relationship between Bus Width, Access Size, and Number of Bursts

Bus Width	BEN Bit	Access Size	Number of Bursts	Number of Accesses
8 bits	Not affected	8 bits	1	1
	Not affected	16 bits	2	1
	Not affected	32 bits	4	1
	0	16 bytes	16	1
	1		4	4
16 bits	Not affected	8 bits	1	1
	Not affected	16 bits	1	1
	Not affected	32 bits	2	1
	0	16 bytes	8	1
	1		2	4
32 bits	Not affected	8 bits	1	1
	Not affected	16 bits	1	1
	Not affected	32 bits	1	1
	Not affected	16 bytes	4	1

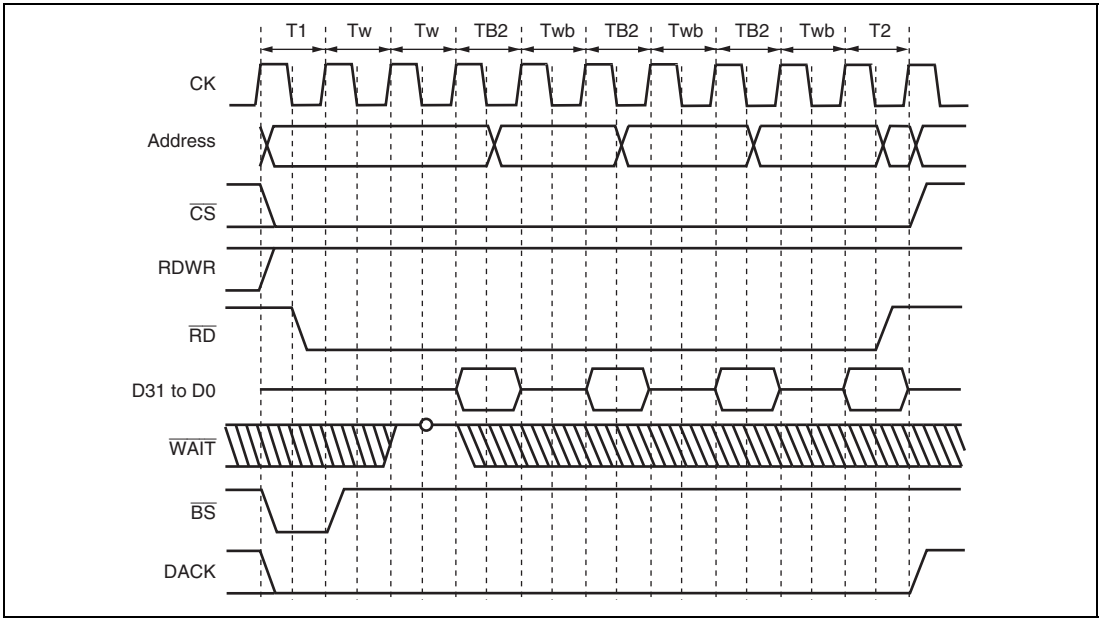


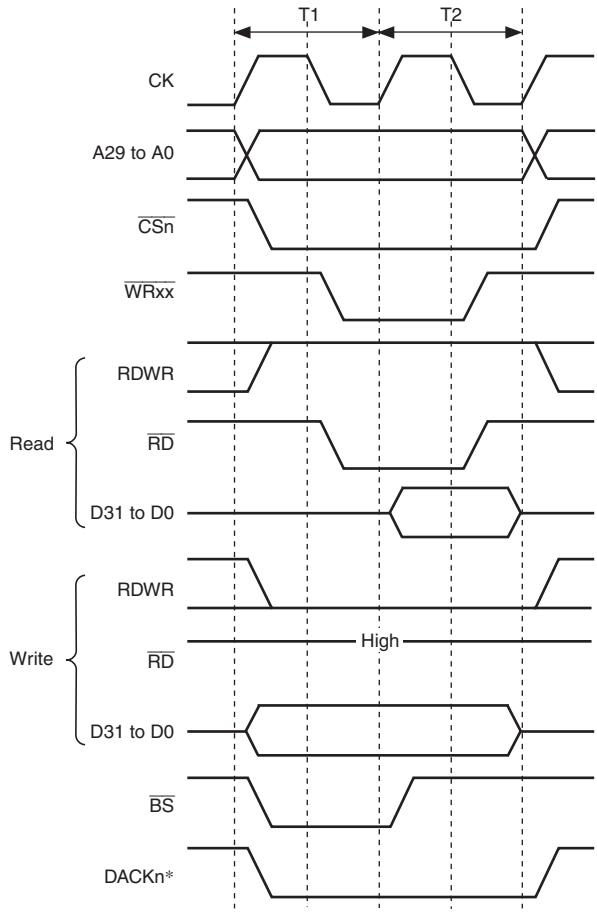
Figure 9.32 Burst ROM (Clock Asynchronous) Access (Bus Width = 32 Bits, 16 byte Transfer (Number of Burst = 4), Access Wait for the 1st time = 2, Access Wait for 2nd Time and after = 1)

9.5.8 SRAM Interface with Byte Selection

The SRAM interface with byte selection is a memory interface which outputs a byte-selection pin ($\overline{\text{WRxx}}$) in a read/write bus cycle. This interface has 16-bit data pins and accesses SRAMs having upper and lower byte-selection pins, such as UB and LB.

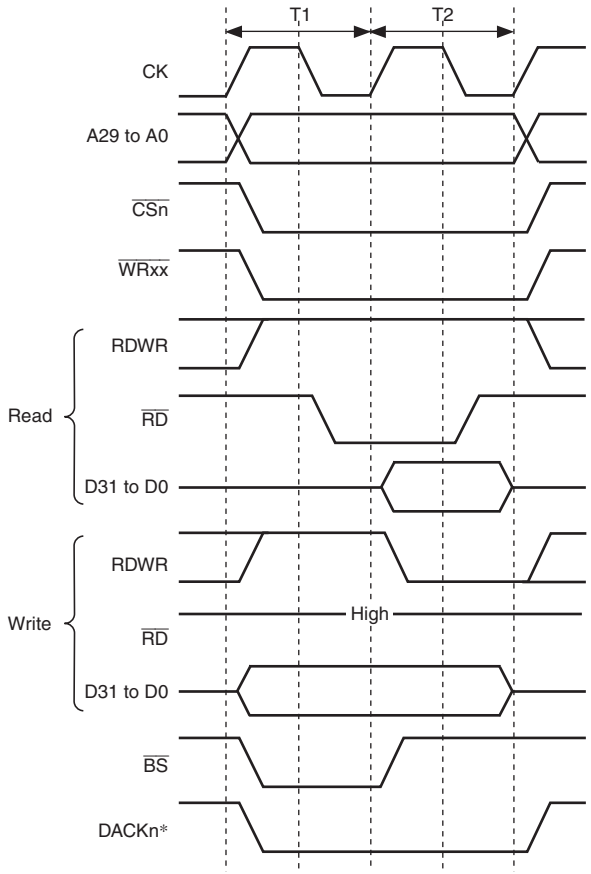
When the BAS bit in CSnWCR is cleared to 0 (initial value), the write access timing of the SRAM interface with byte selection is the same as that for the normal space interface. While in read access of the SRAM interface with byte selection, the byte-selection signal is output from the $\overline{\text{WRxx}}$ pin, which is different from that for the normal space interface. The basic access timing is shown in figure 9.33. In write access, data is written to memory according to the timing of the byte-selection pin ($\overline{\text{WRxx}}$). For details, refer to the Data Sheet for the corresponding memory.

If the BAS bit in CSnWCR is set to 1, the $\overline{\text{WRxx}}$ pin and RDWR pin timings change. Figure 9.34 shows the basic access timing. In write access, data is written to memory according to the timing of the write enable pin (RDWR). The data hold timing from RDWR negation to data write must be acquired by setting the HW1 and HW0 bits in the CSnWCR register. Figure 9.35 shows the access timing when a software wait is specified.



Note: * The waveform for DACKn is when active low is specified.

Figure 9.33 Basic Access Timing for SRAM with Byte Selection (BAS = 0)



Note: * The waveform for DACKn is when active low is specified.

Figure 9.34 Basic Access Timing for SRAM with Byte Selection (BAS = 1)

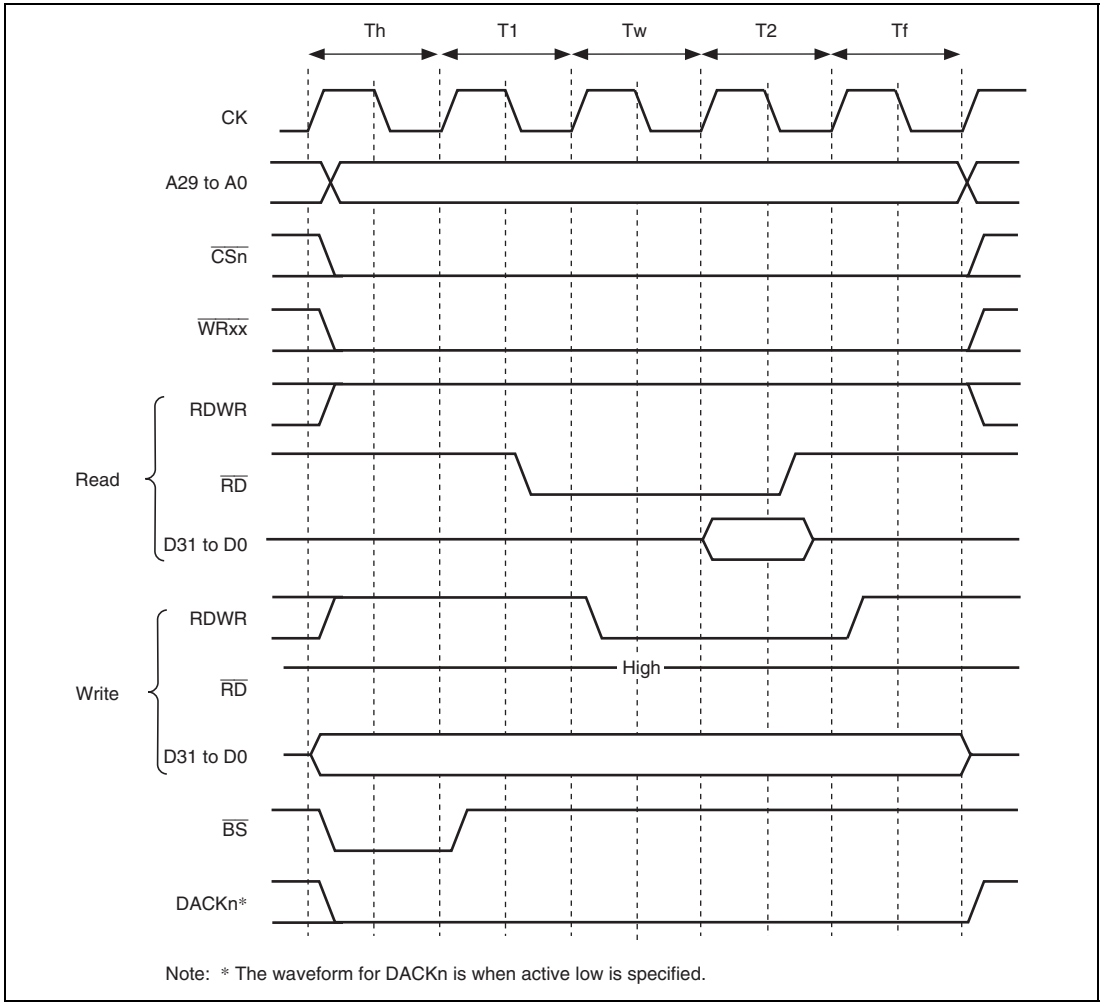


Figure 9.35 Byte Selection SRAM Wait Timing
(BAS = 1, SW[1:0] = 01, WR[3:0] = 0001, HW[1:0] = 01)

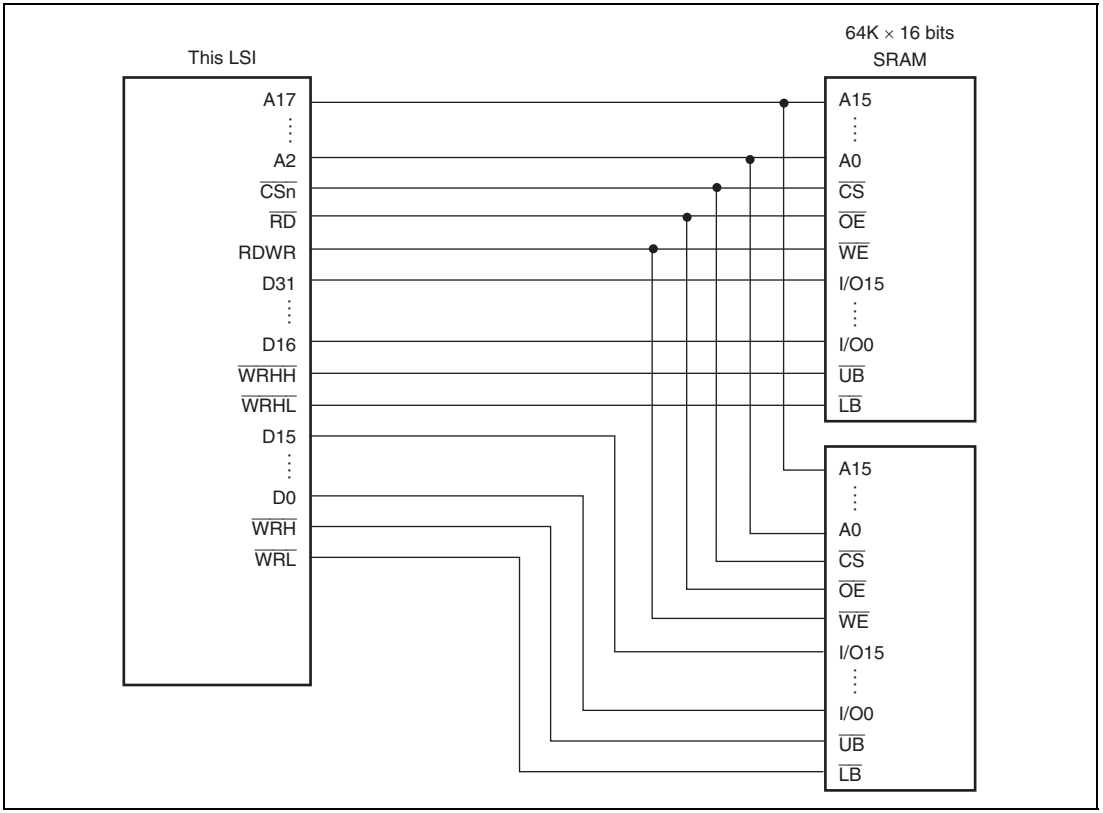


Figure 9.36 Example of Connection with 32-Bit Data Width Byte-Selection SRAM

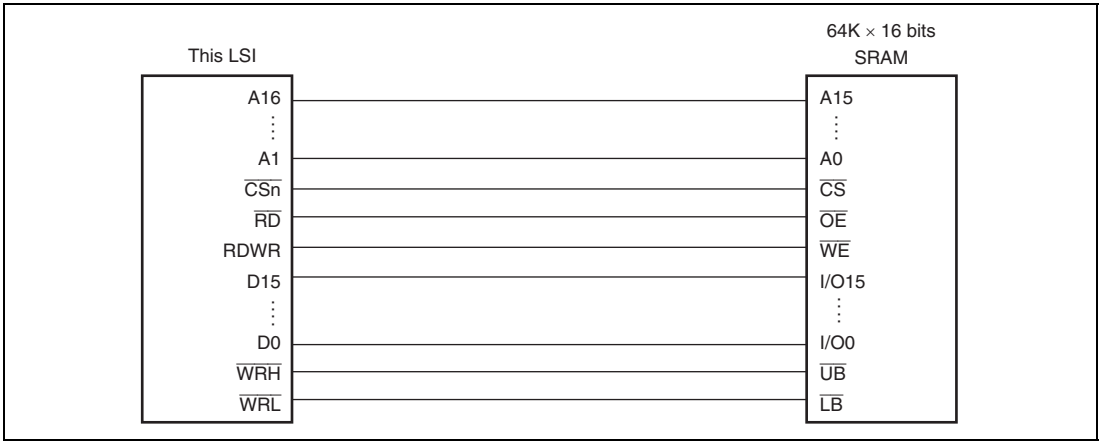


Figure 9.37 Example of Connection with 16-Bit Data Width Byte-Selection SRAM

9.5.9 PCMCIA Interface

With this LSI, the PCMCIA interface can be specified in areas 5 and 6. Areas 5 and 6 can be used for the IC memory card and I/O card interface defined in the JEIDA specifications version 4.2 (PCMCIA2.1 Rev. 2.1) by specifying bits TYPE2 to TYPE0 in CSnBCR (n = 5 and 6) to B'101. In addition, bits SA1 and SA0 in CSnWCR (n = 5 and 6) assign the upper or lower 32 Mbytes of each area to IC memory card or I/O card interface. For example, if bits SA1 and SA0 in CS5WCR are set to 1 and cleared to 0, respectively, the upper 32 Mbytes of area 5 are used as IC memory card interface and the lower 32 Mbytes are used as I/O card interface.

When the PCMCIA interface is used, the bus size must be specified as 8 bits or 16 bits using bits BSZ1 and BSZ0 in CS5BCR or CS6BCR.

Figure 9.38 shows an example of a connection between this LSI and a PCMCIA card. To enable hot swapping (insertion and removal of the PCMCIA card with the system power turned on), tri-state buffers must be connected between the LSI and the PCMCIA card.

In the JEIDA and PCMCIA standards, operation in big endian mode is not clearly defined. Consequently, the provided PCMCIA interface in big endian mode is available only for this LSI.

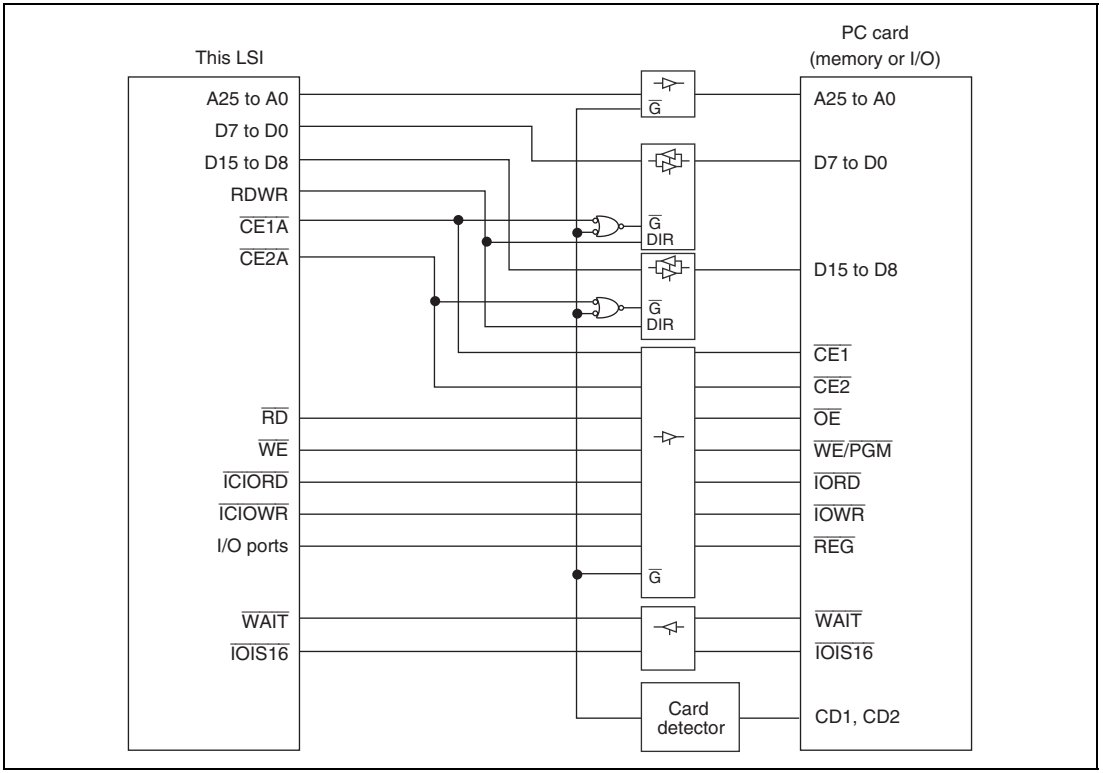


Figure 9.38 Example of PCMCIA Interface Connection

Basic Timing for Memory Card Interface: Figure 9.39 shows the basic timing of the PCMCIA IC memory card interface. If areas 5 and 6 are specified as the PCMCIA interface, accessing the common memory areas in areas 5 and 6 automatically accesses the bus with the IC memory card interface. If the external bus frequency (CK) increases, the setup times and hold times for the address pins (A25 to A0), card enable signals ($\overline{\text{CE1A}}$, $\overline{\text{CE1B}}$, $\overline{\text{CE2A}}$, $\overline{\text{CE2B}}$), and write data (D15 to D0) to the $\overline{\text{RD}}$ and $\overline{\text{WE}}$ signals become insufficient. To prevent this error, this LSI enables the setup times and hold times for areas 5 and 6 to be specified independently, using CS5WCR and CS6WCR. In the PCMCIA interface, as in the normal space interface, a software wait or hardware wait using the $\overline{\text{WAIT}}$ pin can be inserted. Figure 9.40 shows the PCMCIA memory bus wait timing.

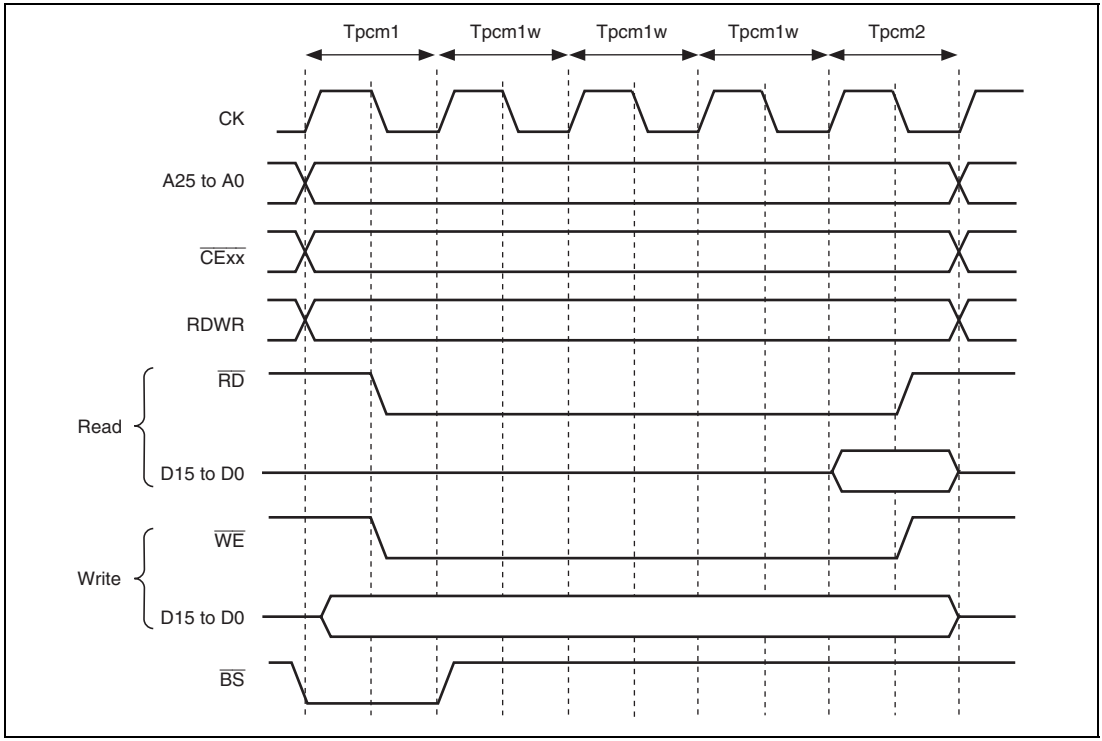


Figure 9.39 Basic Access Timing for PCMCIA Memory Card Interface

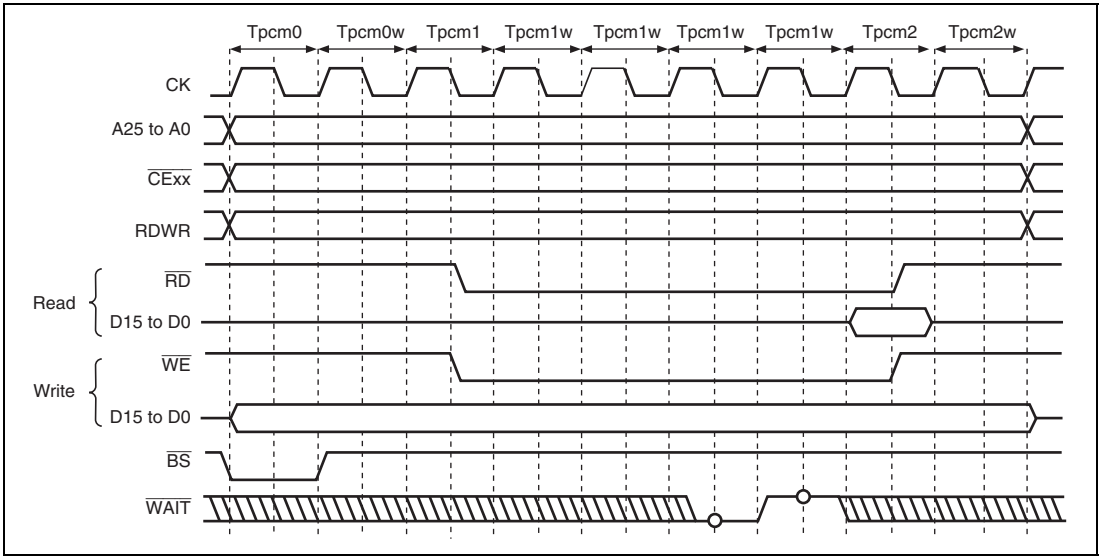


Figure 9.40 Wait Timing for PCMCIA Memory Card Interface (TED[3:0] = B'0010, TEH[3:0] = B'0001, Hardware Wait = 1)

When 32 Mbytes of the memory space are used as IC memory card interface, a port is used to generate the $\overline{\text{REG}}$ signal that switches between the common memory and attribute memory. When the memory space used for the IC memory card interface is 16 Mbytes or less, pin A24 can be used as the $\overline{\text{REG}}$ signal by allocating a 16-Mbyte common memory space and a 16-Mbyte attribute memory space.

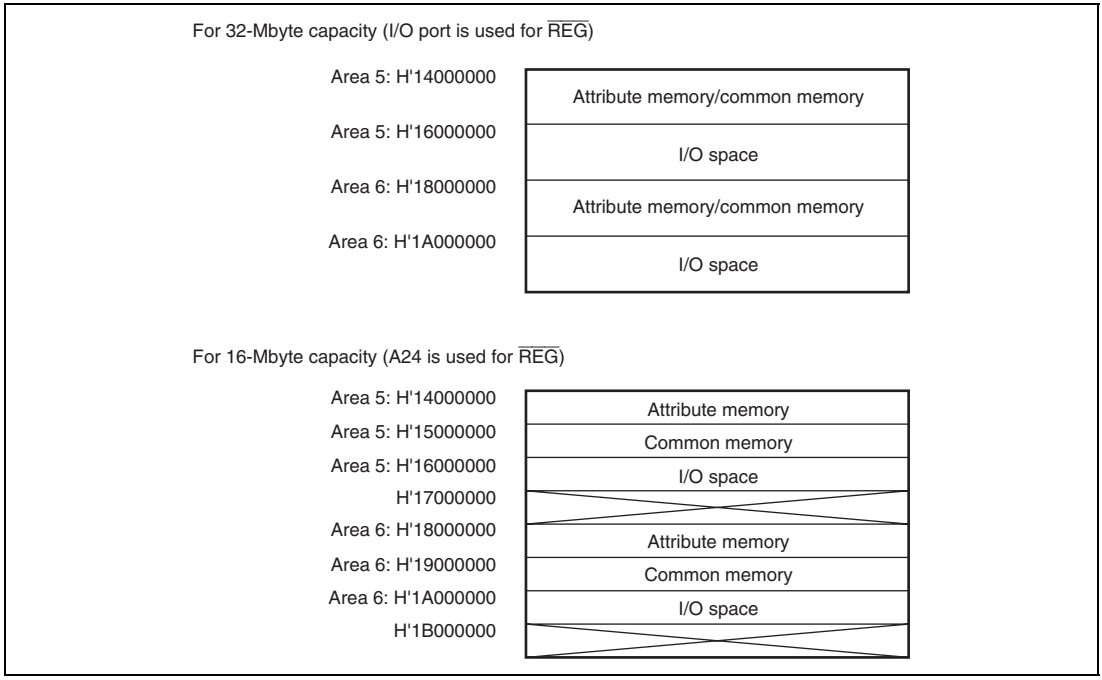


Figure 9.41 Example of PCMCIA Space Assignment (CS5WCR.SA[1:0] = B'10, CS6WCR.SA[1:0] = B'10)

Basic Timing for I/O Card Interface: Figures 9.42 and 9.43 show the basic timings for the PCMCIA I/O card interface.

The I/O card and IC memory card interfaces are switched by an address to be accessed. When area 5 is specified as the PCMCIA and both bits SA1 and SA0 in CS5WCR are set to 1, I/O card areas are allocated to address ranges from H'16000000 to H'17FFFFFF and from H'14000000 to H'15FFFFFF. When area 6 is specified as the PCMCIA and both bits SA1 and SA0 in CS6WCR are set to 1, I/O card areas allocated to address ranges from H'1A000000 to H'1BFFFFFF and from H'18000000 to H'19FFFFFF.

In addition, note that this LSI does not support little endian and the $\overline{\text{IOIS16}}$ signal must be fixed low.

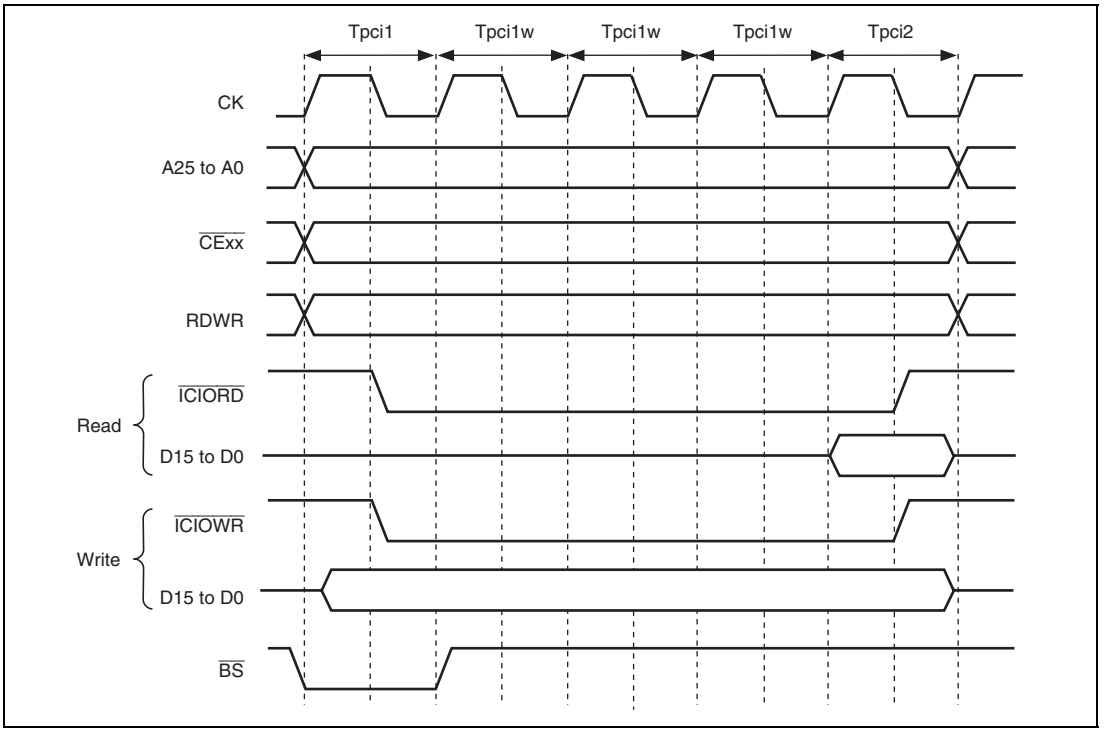


Figure 9.42 Basic Timing for PCMCIA I/O Card Interface

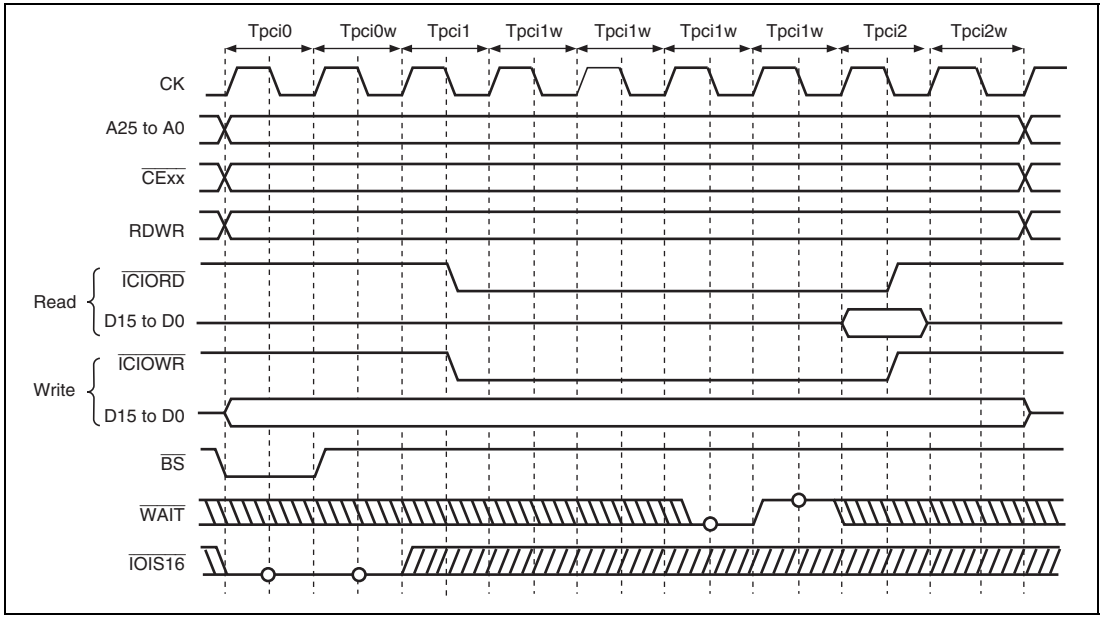


Figure 9.43 Wait Timing for PCMCIA I/O Card Interface Timing (TED[3:0] = B'0010, TEH[3:0] = B'0001, Hardware Wait 1)

9.5.10 Burst MPX-I/O Interface

Figure 9.44 shows an example of a connection between the LSI and the burst MPX device. Figures 9.45 to 9.48 show the burst MPX space access timings.

Area 6 can be specified as the burst address/data multiplex I/O (MPX-I/O) interface using the TYPE2 to TYPE0 bits in the CS6BCR register. This MPX-I/O interface enables the LSI to be easily connected to an external memory controller chip that uses an address/data multiplexed 32-bit single bus. In this case, the address and the access size for the MPX-I/O interface are output to D25 to D0 and D31 to D29, respectively, in address cycles. For the access sizes of D31 to D29, see the description of the CS6WCR register. Address pins A25 to A0 are used to output normal addresses.

In the burst MPX-I/O interface, the bus size is fixed at 32 bits. The BSZ1 and BSZ0 bits in CS6BCR must be specified as 32 bits.

In the burst MPX-I/O interface, a software wait and hardware wait using the $\overline{\text{WAIT}}$ pin can be inserted. In read cycles, a wait cycle is inserted automatically following the address output even if the software wait insertion is specified as 0.

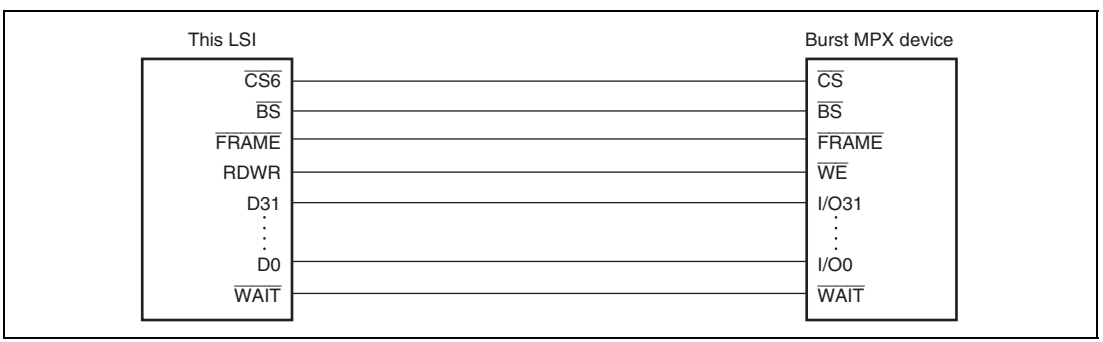
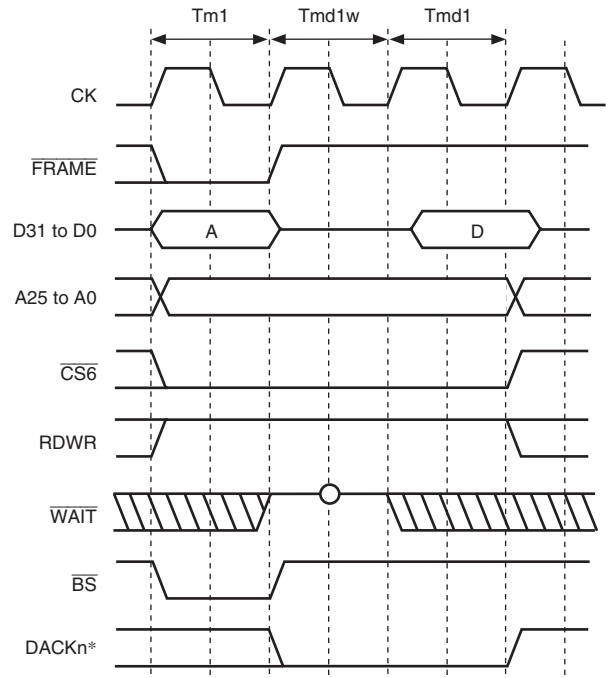
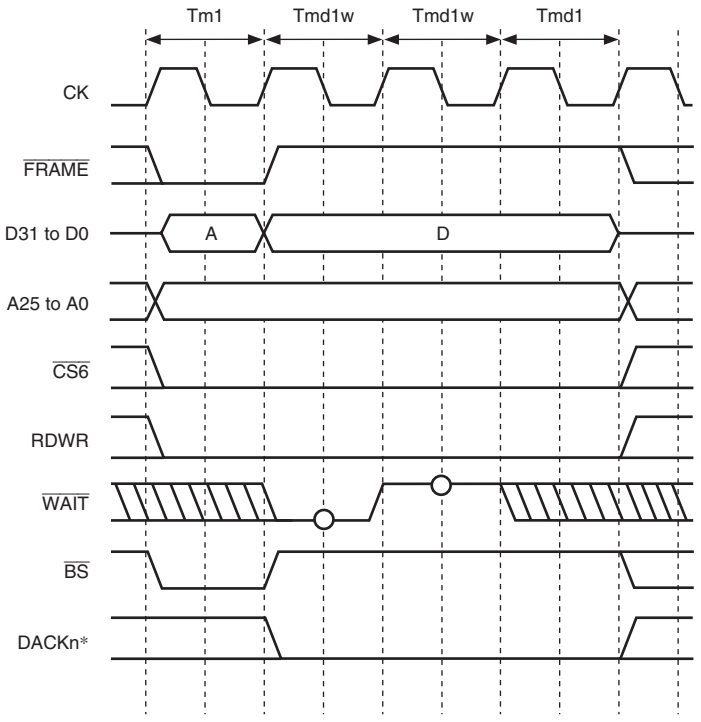


Figure 9.44 Burst MPX Device Connection Example



Note: * The waveform for DACKn is when active low is specified.

Figure 9.45 Burst MPX Space Access Timing (Single Read, No Wait or Software Wait 1)



Note: * The waveform for DACKn is when active low is specified.

Figure 9.46 Burst MPX Space Access Timing (Single Write, Software Wait 1, Hardware Wait 1)

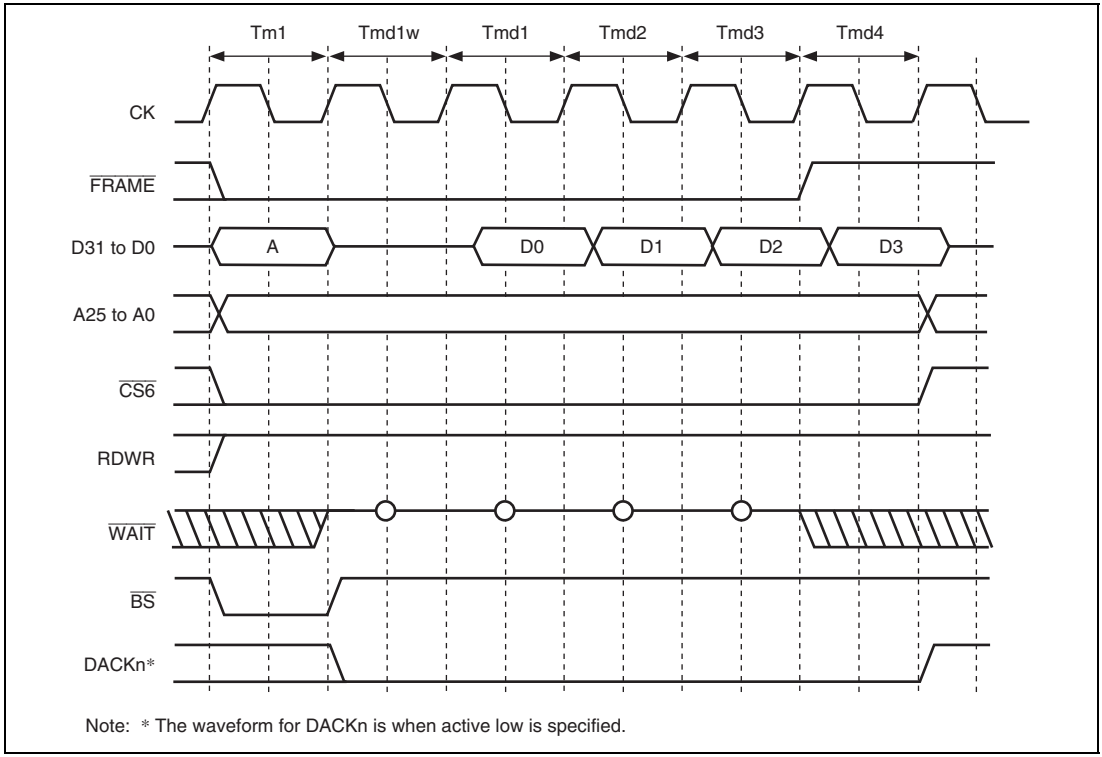
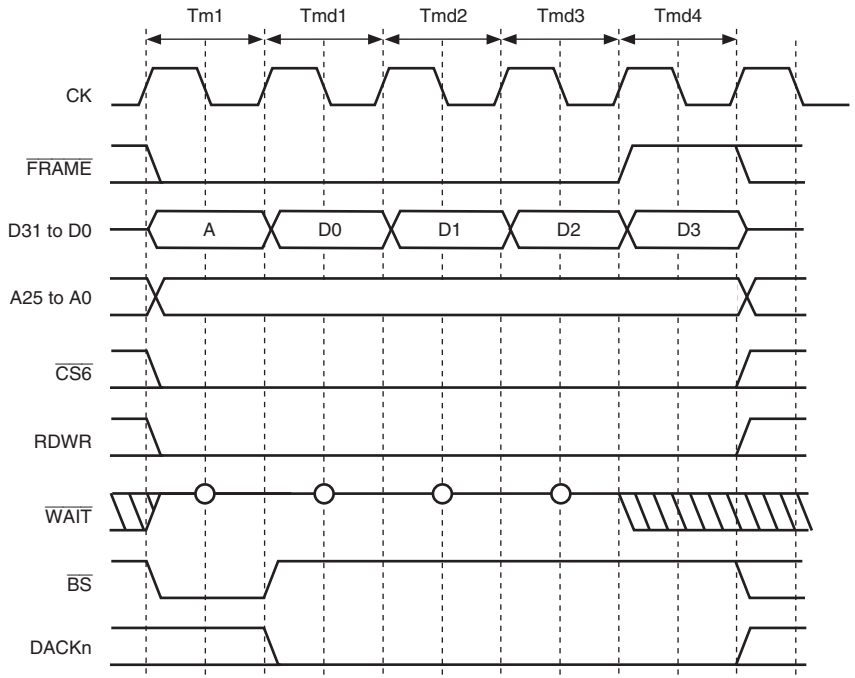


Figure 9.47 Burst MPX Space Access Timing (Burst Read, No Wait or Software Wait 1)



Note: The waveform for DACKn is when active low is specified.

Figure 9.48 Burst MPX Space Access Timing (Burst Write, No Wait)

9.5.11 Burst ROM (Clock Synchronous) Interface

The burst-ROM (clock synchronous) interface provides high-speed access to ROM that has a synchronous burst function. Access through this interface is basically performed in the same way as access to the normal space. If this interface is used, it must be placed in area 0.
For the first access cycle, the number of wait cycles specified by the W3 to W0 bits in CS0WCR are inserted. For the second and subsequent access cycles, the number of wait cycles specified by the BW1 and BW0 bits in CS0WCR are inserted (0 to 3 cycles). In access to the burst ROM (clock synchronous), the $\overline{BS}$ signal is only asserted on the first access cycle. Furthermore, an external wait input is only valid for the first access cycle.

Set burst length = 8 when the bus width is 16 bits and burst length = 4 when the bus width is 32 bits. The 8-bit bus width is not supported. All read access through this interface is in burst mode. For example, in accessing a longword when the bus width is 16 bits, the data from the first two read operations are valid but unnecessary data is read out in the next six dummy operations. This dummy reading slows down memory access rate, lengthening program-execution and DMA-transfer times. Avoiding this requires the effective use of 16-byte DMA transfer.

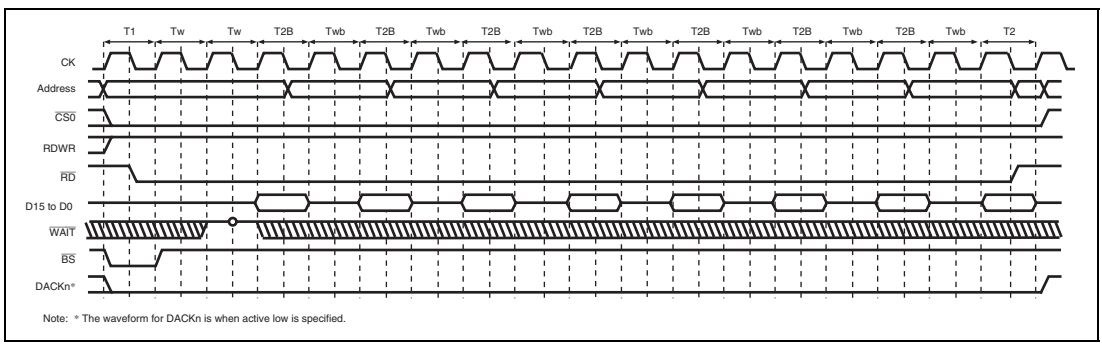


Figure 9.49 Burst ROM (Clock Synchronous) Access Timing (Burst Length = 8, Access Wait for the 1st time = 2, Access Wait for 2nd Time after = 1)

9.5.12 Wait between Access Cycles

As the operating frequency of LSIs becomes higher, the off-operation of the data buffer often collides with the next data output when the data output from devices with slow access speed is completed. As a result of these collisions, the reliability of the device is low and malfunctions may occur. A function that avoids data collisions by inserting wait cycles between continuous access cycles has been newly added.

The number of wait cycles between access cycles can be set by bits IWW[1:0], IWRWD[1:0], IWRWS[1:0], IWRRD[1:0], and IWRRS[1:0] in CSnBCR, and bits DMAIW[1:0] and DMAIWA in CMNCR. The conditions for setting the wait cycles between access cycles (idle cycles) are shown below.

1. Continuous accesses are write-read or write-write
2. Continuous accesses are read-write for different spaces
3. Continuous accesses are read-write for the same space
4. Continuous accesses are read-read for different spaces
5. Continuous accesses are read-read for the same space
6. Data output cycle of an external device caused by DMA transfer in single address mode is followed by data output from another device that includes this LSI (DMAIWA = 0)
7. Data output cycle of an external device caused by DMA transfer in single address mode is followed by any type of access (DMAIWA = 1)

Besides the wait cycles between access cycles (idle cycles) described above, idle cycles must be inserted to reserve the minimum pulse width for a multiplexed pin ($\overline{\text{WRxx}}$), and an interface with an internal bus.

8. Idle cycle of the external bus for the interface with the internal bus
   A. Insert one idle cycle immediately before a write access cycle after an external bus idle cycle or a read cycle.
   B. Insert one idle cycle to transfer the read data to the internal bus when a read cycle of the external bus terminates.

      Insert two to three idle cycles including the idle cycle in A. for the write cycle immediately after a read cycle.
9. Idle cycle of the external bus for accessing different memory

   For accessing different memory, insert idle cycles as follows. The byte-selection SRAM interface with the BAS bit = 1 specified is handled as an SDRAM interface because the $\overline{\text{WRxx}}$ change timing is identical.

A. Insert one idle cycle to access the interface other than the SDRAM interface after the write access cycle is performed in the SDRAM interface.
B. Insert one idle cycle to access the SDRAM interface after the normal space interface with the external wait invalidated or the byte-selection SRAM interface with the BAS bit = 0 specified is accessed.
C. Insert one idle cycle to access the SDRAM interface after the MPX-I/O interface is accessed.
D. Insert two idle cycles to access the MPX-I/O interface from the external bus that is in the idle status.
E. Insert one idle cycle to access the MPX-I/O interface after a read cycle is performed in the normal space interface, byte-selection SRAM interface with the BAS bit = 0, and the SDRAM interface.
F. Insert two idle cycles to access the MPX-I/O interface after a write cycle is performed in the SDRAM interface.

Tables 9.29 to 9.34 list the minimum number of idle cycles to be inserted for the normal space interface and the SDRAM interface. The CSnBCR Idle Setting column in the tables describes the number of idle cycles to be set for IWW, IWRWD, IWRWS, IWRRD, and IWRRS.

Table 9.29 Minimum Number of Idle Cycles between CPU Access Cycles in Normal Space Interface

BSC Register Setting		When Access Size is Less than Bus Width				When Access Size Exceeds Bus Width					
CSnWCR. WM Setting	CSnBCR Idle Setting	Read to Read	Write to Write	Read to Write	Write to Read	Continuous Read*1	Continuous Write*1	Read to Read*2	Write to Write*2	Read to Write*2	Write to Read*2
1	0	1/1/1/1	0/0/0/0	3/3/3/4	0/0/0/0	0/0/0/0	0/0/0/0	1/1/1/1	0/0/0/0	3/3/3/4	0/0/0/0
0	0	1/1/1/1	1/1/1/1	3/3/3/4	1/1/1/1	1/1/1/1	1/1/1/1	1/1/1/1	1/1/1/1	3/3/3/4	1/1/1/1
1	1	1/1/1/1	1/1/1/1	3/3/3/4	1/1/1/1	1/1/1/1	1/1/1/1	1/1/1/1	1/1/1/1	3/3/3/4	1/1/1/1
0	1	1/1/1/1	1/1/1/1	3/3/3/4	1/1/1/1	1/1/1/1	1/1/1/1	1/1/1/1	1/1/1/1	3/3/3/4	1/1/1/1
1	2	2/2/2/2	2/2/2/2	3/3/3/4	2/2/2/2	2/2/2/2	2/2/2/2	2/2/2/2	2/2/2/2	3/3/3/4	2/2/2/2
0	2	2/2/2/2	2/2/2/2	3/3/3/4	2/2/2/2	2/2/2/2	2/2/2/2	2/2/2/2	2/2/2/2	3/3/3/4	2/2/2/2
1	4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4
0	4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4

Notes: The minimum numbers of idle cycles are described sequentially for Iϕ:Bϕ = 4:1, 3:1, 2:1, and 1:1.

1. Minimum number of idle cycles between the word access to address 0 and the word access to address 2 in the 32-bit access with a 16-bit bus width,
   minimum number of idle cycles between the byte access to address 0 and the byte access to address 1 in the 16-bit access with an 8-bit bus width,
   minimum number of idle cycles between the byte accesses to address 0, to address 1, to address 2, and to address 3 in the 32-bit access with an 8-bit bus width, and
   minimum number of idle cycles between consecutive accesses in 16-byte transfer.
2. Other than the above cases

Table 9.30 Minimum Number of Idle Cycles between Access Cycles during DMAC Dual Address Mode and DTC Transfer for the Normal Space Interface

BSC Register Setting		When Access Size is Less than Bus Width		When Access Size Exceeds Bus Width			
CSnWCR. WM Setting	CSnBCR Idle Setting	Read to Write	Write to Read	Continuous Read*1	Read to Write*2	Continuous Write*1	Write to Read*2
1	0	2	0	0	2	0	0
0	0	2	1	1	2	1	1
1	1	2	1	1	2	1	1
0	1	2	1	1	2	1	1
1	2	2	2	2	2	2	2
0	2	2	2	2	2	2	2
1	4	4	4	4	4	4	4
0	4	4	4	4	4	4	4

Notes: DMAC and DTC are driven by Bϕ. The minimum number of idle cycles is not affected by changing a clock ratio.

1. Minimum number of idle cycles between the word access to address 0 and the word access to address 2 in the 32-bit access with a 16-bit bus width,
   minimum number of idle cycles between the byte access to address 0 and the byte access to address 1 in the 16-bit access with an 8-bit bus width,
   minimum number of idle cycles between the byte accesses to address 0, to address 1, to address 2, and to address 3 in the 32-bit access with an 8-bit bus width, and
   minimum number of idle cycles between consecutive accesses in 16-byte transfer.
2. Other than the above cases.

Table 9.31 Minimum Number of Idle Cycles during DMAC Single Address Mode Transfer to the Normal Space Interface from the External Device with DACK

(1) Transfer from the external device with DACK to the normal space interface

BSC Register Setting*3			Minimum Number of Idle Cycles	
CSnWCR.WM Setting	CMNCR.DMAIWA Setting	CMNCR.DMAIW Idle Setting	When Access Size is Greater than Bus Width*1	When Access Size is Less than or Equal to Bus Width*2
1	0	—	0	1*5
0	0	—	1	1
1	1	0	0	1*5
0	1	0	1	1
1	1	1	1	1
0	1	1	1	1
1	1	2	2	2
0	1	2	2	2
1	1	4	4	4
0	1	4	4	4

(2) Transfer from the normal space interface to the external device with DACK

BSC Register Setting*4		Minimum Number of Idle Cycles	
CSnWCR.WM Setting	CSnBCR Idle Setting	When Access Size is Greater than Bus Width*1	When Access Size is Less than or Equal to Bus Width*2
1	0	0	2
0	0	1	3
1	1	1	2
0	1	1	3
1	2	2	2
0	2	2	3
1	4	4	4
0	4	4	4

Notes: DMAC is driven by Bϕ. The minimum number of idle cycles is not affected by changing a clock ratio.

1. Minimum number of idle cycles between the word access to address 0 and the word access to address 2 in the 32-bit access with a 16-bit bus width,
minimum number of idle cycles between the byte access to address 0 and the byte access to address 1 in the 16-bit access with an 8-bit bus width,
minimum number of idle cycles between the byte accesses to address 0, to address 1, to address 2, and to address 3 in the 32-bit access with an 8-bit bus width, and
minimum number of idle cycles between consecutive accesses in 16-byte transfer.
2. Other than the above cases.
3. For single address mode transfer from the external device with DACK to the normal space interface, the minimum number of idle cycles is not affected by the IWW, IWRWD, IWRWS, IWRRD, and IWRRS bits in CSnBCR.
4. For single address mode transfer from the normal space interface to the external device with DACK, the minimum number of idle cycles is not affected by the DMAIWA and DMAIW bits in CMNCR.
5. When the HW[1:0] in the CSnWCR is set to specify 2.5 cycles or more, the number of idle cycles will be 0.

Table 9.32 Minimum Number of Idle Cycles between Access Cycles of CPU, the DMAC Dual Address Mode, and DTC for the SDRAM Interface

BSC Register Setting			CPU Access				DMAC or DTC Access	
CSnBCR Idle Setting	CS3WCR. WTRP Setting	CS3WCR. TRWL Setting	Read to Read	Write to Write	Read to Write	Write to Read	Read to Write	Write to Read
0	1	0	1/1/1/1	0/0/0/0	3/3/3/4	0/0/0/0	2	0
0	1	1	1/1/1/1	1/1/1/1	3/3/3/4	1/1/1/1	2	1
0	1	2	1/1/1/1	2/2/2/2	3/3/3/4	2/2/2/2	2	2
0	1	3	1/1/1/1	3/3/3/3	3/3/3/4	3/3/3/3	2	3
0	2	0	2/2/2/2	1/1/1/1	3/3/3/4	1/1/1/1	2	1
0	2	1	2/2/2/2	2/2/2/2	3/3/3/4	2/2/2/2	2	2
0	2	2	2/2/2/2	3/3/3/3	3/3/3/4	3/3/3/3	2	3
0	2	3	2/2/2/2	4/4/4/4	3/3/3/4	4/4/4/4	2	4
0	3	0	3/3/3/3	2/2/2/2	3/3/3/4	2/2/2/2	3	2
0	3	1	3/3/3/3	3/3/3/3	3/3/3/4	3/3/3/3	3	3
0	3	2	3/3/3/3	4/4/4/4	3/3/3/4	4/4/4/4	3	4
0	3	3	3/3/3/3	5/5/5/5	3/3/3/4	5/5/5/5	3	5
0	4	0	4/4/4/4	3/3/3/3	4/4/4/4	3/3/3/3	4	3
0	4	1	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4	4
0	4	2	4/4/4/4	5/5/5/5	4/4/4/4	5/5/5/5	4	5
0	4	3	4/4/4/4	6/6/6/6	4/4/4/4	6/6/6/6	4	6
1	1	0	2/2/2/2	1/1/1/1	3/3/3/4	1/1/1/1	2	1
1	1	1	2/2/2/2	1/1/1/1	3/3/3/4	1/1/1/1	2	1
1	1	2	2/2/2/2	2/2/2/2	3/3/3/4	2/2/2/2	2	2
1	1	3	2/2/2/2	3/3/3/3	3/3/3/4	3/3/3/3	2	3
1	2	0	2/2/2/2	1/1/1/1	3/3/3/4	1/1/1/1	2	1
1	2	1	2/2/2/2	2/2/2/2	3/3/3/4	2/2/2/2	2	2
1	2	2	2/2/2/2	3/3/3/3	3/3/3/4	3/3/3/3	2	3
1	2	3	2/2/2/2	4/4/4/4	3/3/3/4	4/4/4/4	2	4
1	3	0	3/3/3/3	2/2/2/2	3/3/3/4	2/2/2/2	3	2
1	3	1	3/3/3/3	3/3/3/3	3/3/3/4	3/3/3/3	3	3

BSC Register Setting			CPU Access				DMAC or DTC Access	
CSnBCR Idle Setting	CS3WCR. WTRP Setting	CS3WCR. TRWL Setting	Read to Read	Write to Write	Read to Write	Write to Read	Read to Write	Write to Read
1	3	2	3/3/3/3	4/4/4/4	3/3/3/4	4/4/4/4	3	4
1	3	3	3/3/3/3	5/5/5/5	3/3/3/4	5/5/5/5	3	5
1	4	0	4/4/4/4	3/3/3/3	4/4/4/4	3/3/3/3	4	3
1	4	1	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4	4
1	4	2	4/4/4/4	5/5/5/5	4/4/4/4	5/5/5/5	4	5
1	4	3	4/4/4/4	6/6/6/6	4/4/4/4	6/6/6/6	4	6
2	1	0	3/3/3/3	2/2/2/2	3/3/3/4	2/2/2/2	3	2
2	1	1	3/3/3/3	2/2/2/2	3/3/3/4	2/2/2/2	3	2
2	1	2	3/3/3/3	2/2/2/2	3/3/3/4	2/2/2/2	3	2
2	1	3	3/3/3/3	3/3/3/3	3/3/3/4	3/3/3/3	3	3
2	2	0	3/3/3/3	2/2/2/2	3/3/3/4	2/2/2/2	3	2
2	2	1	3/3/3/3	2/2/2/2	3/3/3/4	2/2/2/2	3	2
2	2	2	3/3/3/3	3/3/3/3	3/3/3/4	3/3/3/3	3	3
2	2	3	3/3/3/3	4/4/4/4	3/3/3/4	4/4/4/4	3	4
2	3	0	3/3/3/3	2/2/2/2	3/3/3/4	2/2/2/2	3	2
2	3	1	3/3/3/3	3/3/3/3	3/3/3/4	3/3/3/3	3	3
2	3	2	3/3/3/3	4/4/4/4	3/3/3/4	4/4/4/4	3	4
2	3	3	3/3/3/3	5/5/5/5	3/3/3/4	5/5/5/5	3	5
2	4	0	4/4/4/4	3/3/3/3	4/4/4/4	3/3/3/3	4	3
2	4	1	4/4/4/4	4/4/4/4	4/4/4/4	4/4/4/4	4	4
2	4	2	4/4/4/4	5/5/5/5	4/4/4/4	5/5/5/5	4	5
2	4	3	4/4/4/4	6/6/6/6	4/4/4/4	6/6/6/6	4	6
4	1	0	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	1	1	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	1	2	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	1	3	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	2	0	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	2	1	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4

BSC Register Setting			CPU Access				DMAC or DTC Access	
CSnBCR Idle Setting	CS3WCR. WTRP Setting	CS3WCR. TRWL Setting	Read to Read	Write to Write	Read to Write	Write to Read	Read to Write	Write to Read
4	2	2	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	2	3	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	3	0	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	3	1	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	3	2	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	3	3	5/5/5/5	5/5/5/5	5/5/5/5	5/5/5/5	5	5
4	4	0	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	4	1	5/5/5/5	4/4/4/4	5/5/5/5	4/4/4/4	5	4
4	4	2	5/5/5/5	5/5/5/5	5/5/5/5	5/5/5/5	5	5
4	4	3	5/5/5/5	6/6/6/6	5/5/5/5	6/6/6/6	5	6

Note: The minimum numbers of idle cycles in CPU Access are described sequentially for Iϕ:Bϕ = 4:1, 3:1, 2:1, and 1:1.

DMAC and DTC are driven by Bϕ. The minimum number of idle cycles is not affected by changing a clock ratio.

Table 9.33 Minimum Number of Idle Cycles between Access Cycles of the DMAC Single Address Mode for the SDRAM Interface (1)

Transfer from the external device with DACK to the SDRAM interface:

BSC Register Setting*1			
CMNCR.DMAIW Setting	**CS3WCR.WTRP Setting**	**CS3WCR.TRWL Setting**	**Minimum Number of Idle Cycles**
0	1	0	1*2
0	1	1	1
0	1	2	2
0	1	3	3
0	2	0	1
0	2	1	2
0	2	2	3
0	2	3	4
0	3	0	2
0	3	1	3
0	3	2	4
0	3	3	5
0	4	0	3
0	4	1	4
0	4	2	5
0	4	3	6
1	1	0	1
1	1	1	1
1	1	2	2
1	1	3	3
1	2	0	1
1	2	1	2
1	2	2	3
1	2	3	4
1	3	0	2
1	3	1	3
1	3	2	4
1	3	3	5
1	4	0	3

BSC Register Setting*1			
CMNCR.DMAIW Setting	**CS3WCR.WTRP Setting**	**CS3WCR.TRWL Setting**	**Minimum Number of Idle Cycles**
1	4	1	4
1	4	2	5
1	4	3	6
2	1	0	2
2	1	1	2
2	1	2	2
2	1	3	3
2	2	0	2
2	2	1	2
2	2	2	3
2	2	3	4
2	3	0	2
2	3	1	3
2	3	2	4
2	3	3	5
2	4	0	3
2	4	1	4
2	4	2	5
2	4	3	6
4	1	0	4
4	1	1	4
4	1	2	4
4	1	3	4
4	2	0	4
4	2	1	4
4	2	2	4
4	2	3	4
4	3	0	4
4	3	1	4
4	3	2	4
4	3	3	5
4	4	0	4
4	4	1	4

BSC Register Setting*1			
CMNCR.DMAIW Setting	**CS3WCR.WTRP Setting**	**CS3WCR.TRWL Setting**	**Minimum Number of Idle Cycles**
4	4	2	5
4	4	3	6

Notes: DMAC is driven by Bϕ. The minimum number of idle cycles is not affected by changing a clock ratio.

1. For single address mode transfer from the external device with DACK to the SDRAM interface, the minimum number of idle cycles is not affected by the IWW, IWRWD, IWRWS, IWRRD, and IWRRS bits in CSnBCR.
2. Set the WTRCD bits to select 1 cycle or less.

Table 9.34 Minimum Number of Idle Cycles between Access Cycles of the DMAC Single Address Mode for the SDRAM Interface (2)

Transfer from the SDRAM interface to the external device with DACK

BSC Register Setting*		Minimum Number of Idle Cycles
CS3BCR Idle Setting	CS3WCR.WTRP Setting	
0	1	3
0	2	3
0	3	3
0	4	4
1	1	3
1	2	3
1	3	3
1	4	4
2	1	3
2	2	3
2	3	3
2	4	4
4	1	5
4	2	5
4	3	5
4	4	5

Notes: DMAC is driven by Bϕ. The minimum number of idle cycles is not affected by changing a clock ratio.

* Other than the following cases.
Single address mode transfer from the external device with DACK to the SDRAM interface, where the minimum number of idle cycles is not affected by the IWW, IWRWD, IWRWS, IWRRD, and IWRRS bits in CSnBCR.

CMNCR.DMAIWA = 0, where the setting is identical to CMNCR.DMAIW[1:0] in table 9.33.

9.5.13 Bus Arbitration

This LSI owns the bus mastership in normal state and releases the bus only when receiving a bus request from an external device. This LSI has three bus masters: CPU, DMAC, and DTC. The bus mastership is given to these bus masters in accordance with the following priority.

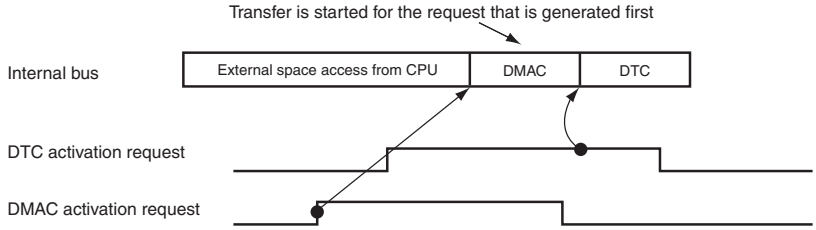
Request for bus mastership by external device ($\overline{\text{BREQ}}$) > CPU > DTC > DMAC > CPU.

However, when DTC or DMAC is requesting the bus mastership, the CPU does not obtain the bus mastership continuously.

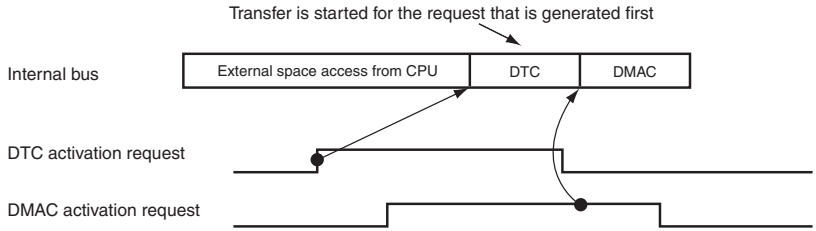
The following cases should be noted regarding the external space access request from the CPU.

1. When the CSSTP2 bit is 1 in the bus function extending register (BSCHER), the external space access request from the CPU has lower priority than the burst transfer request from the DMAC and DTC transfer request with DTLOCK = 0 in the bus function extending register (BSCHER).
2. When an activation request is generated in the order of DMAC and DTC while an external space is being accessed by the CPU, DMA transfer is executed first and then DTC transfer. Figure 9.50 shows the bus arbitration when the DTC and DMAC compete while an external space is accessed by the CPU.

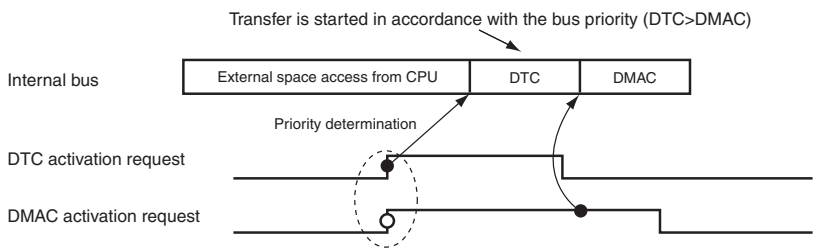
• When activation request is generated in the order of DMAC and DTC during external space access from CPU



• When activation request is generated in the order of DTC and DMAC during external space access from CPU



• When activation request is generated for DTC and DMAC at the same time during external space access from CPU



[Reference] When activation request is generated in the order of DMAC and DTC during access to an on-chip peripheral module by CPU

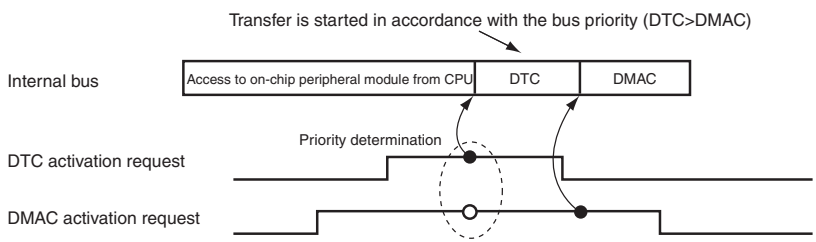
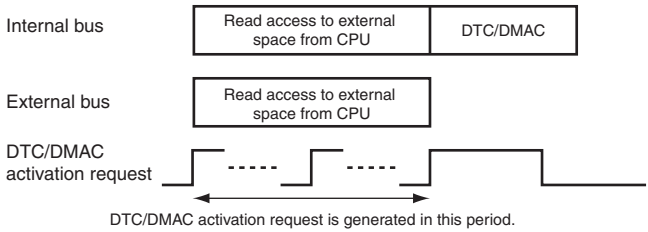


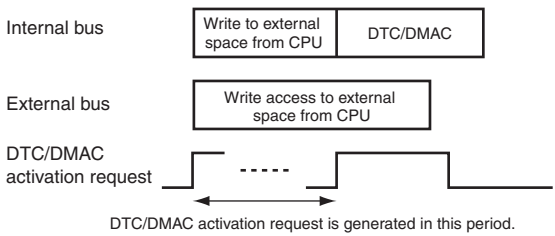
Figure 9.50 Bus Arbitration when DTC and DMAC Compete during External Space Access from CPU

In addition, because the write buffer operates as described in section 9.5.14 (2), Access in View of LSI Internal Bus Master, arbitration between the CPU and DTC/DMAC is different depending on whether the external space access by the CPU is a write or read access. Figure 9.51 shows the bus arbitration when a DTC or DMAC activation request is generated while an external space is accessed by CPU.

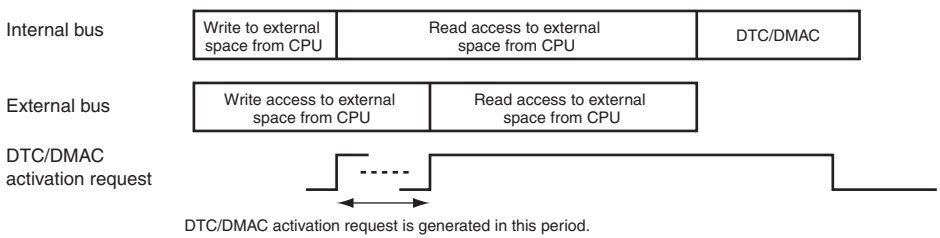
• When DTC/DMAC activation request is generated during read access to external space from CPU



• When DTC/DMAC activation request is generated during write access to external space from CPU (1)



• When DTC/DMAC activation request is generated during write access to external space from CPU (2)
(When external space read request is generated by CPU during execution of write access to external space from CPU)



• When DTC/DMAC activation request is generated during write access to external space from CPU (3)
(When external space write request is generated by CPU during execution of write access to external space from CPU)

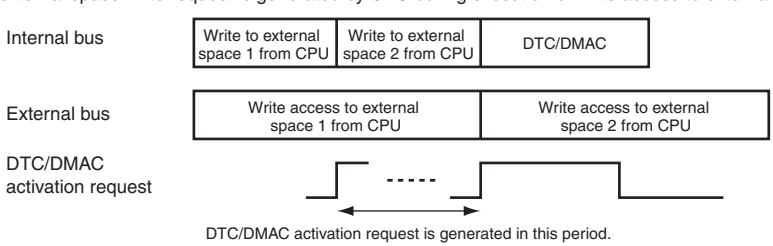


Figure 9.51 Bus Arbitration when DTC or DMAC Activation Request Occurs during External Space Access from CPU

The states that do not allow bus arbitration are shown below.

1. Between the read and write cycles of a TAS instruction
2. Multiple bus cycles generated when the data bus width is smaller than the access size (for example, between bus cycles when longword access is made to a memory with a data bus width of 8 bits)
3. 16-byte transfer by the DMAC

To prevent device malfunction while the bus mastership is transferred to the external device, the LSI negates all of the bus control signals before bus release. When the bus mastership is received, all of the bus control signals are first negated and then driven appropriately. In addition, to prevent noise while the bus control signal is in the high impedance state, pull-up resistors must be connected to these control signals.

Bus mastership is transferred to the external device at the boundary of bus cycles. Namely, bus mastership is released immediately after receiving a bus request when a bus cycle is not being performed. The release of bus mastership is delayed until the bus cycle is complete when a bus cycle is in progress. Even when from outside the LSI it looks like a bus cycle is not being performed, a bus cycle may be performing internally, started by inserting wait cycles between access cycles. Therefore, it cannot be immediately determined whether or not bus mastership has been released by looking at the $\overline{\text{CSn}}$ signal or other bus control signals.

The external bus release by the $\overline{\text{BREQ}}$ and $\overline{\text{BACK}}$ signal handshaking requires some overhead. If the slave has many tasks, multiple bus cycles should be executed in a bus mastership acquisition. Reducing the cycles required for master to slave bus mastership transitions streamlines the system design.

The LSI has the bus mastership until a bus request is received from the external device. Upon acknowledging the assertion (low level) of the external bus request signal $\overline{\text{BREQ}}$, the LSI releases the bus at the completion of the current bus cycle and asserts the $\overline{\text{BACK}}$ signal. After the LSI acknowledges the negation (high level) of the $\overline{\text{BREQ}}$ signal that indicates the slave has released the bus, it negates the $\overline{\text{BACK}}$ signal and resumes the bus usage.

When the SDRAM interface is used, an all bank precharge command (PALL) is issued if any active banks exist and releases the bus after completion of the PALL command.

Processing by this LSI continues even while bus mastership is released to an external device, unless an external device is accessed. When an external device is accessed, the LSI enters the state of waiting for bus mastership to be returned.

While the bus is released, sleep mode, software standby mode, and deep software standby mode cannot be entered.

The bus release sequence is as follows. The address bus and data bus are placed in a high-impedance state synchronized with the rising edge of CK. The bus mastership acknowledge signal is asserted 0.5 cycles after the above high impedance state, synchronized with the falling edge of CK. The bus control signals such as $\overline{\text{CSn}}$ are placed in the high-impedance state at subsequent rising edges of CK. These bus control signals go high one cycle before being placed in the high-impedance state. Bus request signals are sampled at the falling edge of CK. By setting the HIZCNT bit in CMNR, the CKE, $\overline{\text{RASU}}$, $\overline{\text{RASL}}$, $\overline{\text{CASU}}$, and $\overline{\text{CASL}}$ can be continued to be driven even after the bus is released using the values immediately before the bus release.

The sequence for reclaiming the bus mastership from an external device is described below.

At 1.5 cycles after the negation of $\overline{\text{BREQ}}$ is detected at the falling edge of CK, the bus control signals are driven high. The bus acknowledge signal is negated at the next falling edge of the clock. The fastest timing at which actual bus cycles can be resumed after bus control signal assertion is at the rising edge of the CK where address and data signals are driven. Figure 9.52 shows the bus arbitration timing in master mode.

In an original external device designed by the user, multiple bus accesses may be generated continuously to reduce the overhead caused by bus arbitration. In this case, to execute SDRAM refresh correctly, the external device must be designed to release the bus mastership within the refresh interval time.

This LSI has the $\overline{\text{IRQOUT}}$ pin to request the bus while waiting for refresh execution. This LSI continues to assert $\overline{\text{IRQOUT}}$ (low level) until the bus is acquired. When the external device receives this signal and releases the bus, the LSI acquires the bus and executes refresh.

After $\overline{\text{BREQ}}$ assertion (low level; bus request), the $\overline{\text{BREQ}}$ signal should be negated (high level; bus release) only after the $\overline{\text{BACK}}$ is asserted (low level; bus acknowledge). If $\overline{\text{BREQ}}$ is negated before $\overline{\text{BACK}}$ is asserted, $\overline{\text{BACK}}$ may be asserted only for one cycle depending on the $\overline{\text{BREQ}}$ negation timing, and a bus conflict may occur between the external device and this LSI.

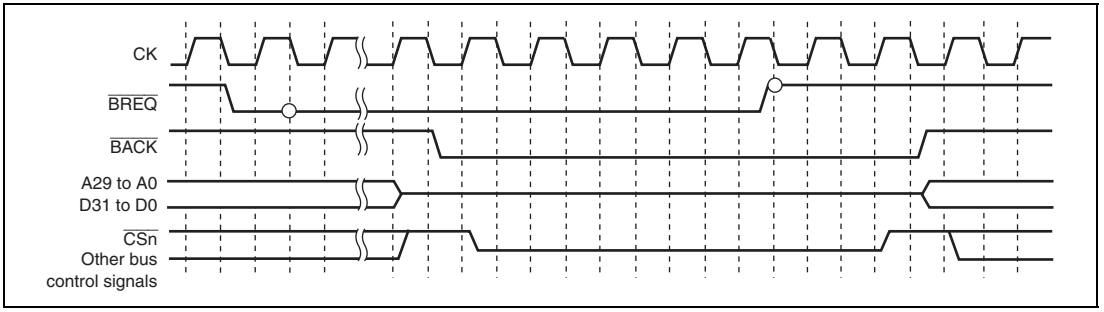


Figure 9.52 Bus Arbitration Timing

Acceptance of mastership for the DMAC in bus arbitration takes 1Bϕ, so a NOP 1Bϕ in duration is inserted on the I bus.

Acceptance of mastership for the DTC in bus arbitration does not require the insertion of a NOP, so bus access proceeds continuously.

9.5.14 Others

(1) Reset

The bus state controller (BSC) can be initialized completely only at a power-on reset. At a power-on reset, all signals are negated and output buffers are turned off regardless of the bus cycle state. All control registers are initialized.

In standby, sleep, and manual reset, control registers of the bus state controller are not initialized. At a manual reset, the current bus cycle being executed is completed and then the access wait state is entered. If a 16-byte transfer is performed by the DMAC is executed, the current access is cancelled in longword units because the access request is cancelled by the bus master at a manual reset. Since the RTCNT continues counting up during manual reset signal assertion, a refresh request occurs to initiate the refresh cycle. However, a bus arbitration request by the $\overline{\text{BREQ}}$ signal cannot be accepted during manual reset signal assertion.

(2) Access in View of LSI Internal Bus Master

There are three types of LSI internal buses: L bus, I bus, and peripheral bus. The CPU is connected to the L bus. The DMAC, DTC, and bus state controller are connected to the I bus. Low-speed peripheral modules are connected to the peripheral bus. On-chip memories are connected bidirectionally to the L bus and I bus.

For an access of an external space or an on-chip peripheral module, the access is initiated via the I bus. Thus, the DMAC and DTC can be activated without bus arbitration with the CPU while the CPU is accessing an on-chip memory.

Since the bus state controller (BSC) incorporates a one-stage write buffer, the BSC can execute an access via the I bus before the previous external bus cycle is completed in a write cycle. If the on-chip peripheral module is read or written after the external low-speed memory is written, the on-chip peripheral module can be accessed before the completion of the external low-speed memory write cycle.

In read cycles, the CPU is placed in the wait state until read operation has been completed. To continue the process after the data write to the device has been completed, perform a dummy read to the same address to check for completion of the write before the next process to be executed.

The write buffer of the BSC functions in the same way for an access by the DMAC and DTC. Accordingly, to perform dual address DMA transfers, the next read cycle is initiated before the previous write cycle is completed. Note, however, that if both the DMA source and destination addresses exist in external memory space, the next read cycle will not be initiated until the previous write cycle is completed.

Since access cannot be performed correctly if any BSC register values are modified while the write buffer is operating, do not modify BSC registers immediately after a write access. If the BSC register need to be modified immediately after a write access, execute dummy read to confirm the completion of the write access, then modify the BSC register.

9.5.15 Access to On-Chip FLASH and On-Chip RAM by CPU

Access to the on-chip FLASH for read is synchronized with Iϕ clock and is executed in one clock cycle. For details on programming and erasing, see section 23, Flash Memory.

Access to the on-chip RAM for read/write is synchronized with I ϕ clock and is executed in one clock cycle. For details, see section 25, RAM.

9.5.16 Access to On-Chip Peripheral I/O Registers by CPU

Table 9.35 shows the number of cycles required for access to the on-chip peripheral I/O registers by the CPU.

Table 9.35 Number of Cycles for Access to On-Chip Peripheral I/O Registers

		Number of Access Cycles*1*2
Pϕ reference	Write	(3 + n) × Iϕ + (1 + m) × Bϕ + 2 × Pϕ*3
	Read	(3 + n) × Iϕ + (1 + m) × Bϕ + 2 × Pϕ*3 + 2 × Iϕ
Bϕ reference	Write	(3 + n) × Iϕ + 3 × Bϕ*4
	Read	(3 + n) × Iϕ + 3 × Bϕ*4*5 + 2Iϕ

Notes: 1. When Iϕ:Bϕ = 8:1, n = 0 to 7.
When Iϕ:Bϕ = 4:1, n = 0 to 3.
When Bϕ:Pϕ = 4:1, m = 0 to 3.
When Iϕ:Bϕ = 3:1, n = 0 to 2.
When Bϕ:Pϕ = 3:1, m = 0 to 2.
When Iϕ:Bϕ = 2:1, n = 0 to 1.
When Bϕ:Pϕ = 2:1, m = 0 to 1.
When Iϕ:Bϕ = 1:1, n = 0.
When Bϕ:Pϕ = 1:1, m = 0.
n and m depend on the internal execution state.
2. The clock ratio of MIϕ and MPϕ does not affect the number of access cycles.
3. The access cycle count is 5 × Pϕ for all flash registers other than RAMER.
4. The access cycle count of the RAMER flash register is 1 × Bϕ.
5. The access cycle count is 1 × Bϕ for all BSC registers other than BSCEHR.

Synchronous logic and a layered bus structure have been adopted for this LSI. Data on each bus are input and output in synchronization with rising edges of the corresponding clock signal. The L bus, I bus, and peripheral bus are synchronized with the Iϕ, Bϕ, and Pϕ clock, respectively. Figure 9.53 shows an example of the timing of write access to a register in 2Pϕ cycle access with the connected peripheral bus width of 16 bits when Iϕ:Bϕ:Pϕ = 4:2:2. In access to the on-chip

peripheral I/O registers, the CPU requires three cycles of Iϕ for preparation of data transfer to the I bus after the data has been output to the L bus. After these three cycles, data can be transferred to the I bus in synchronization with rising edges of Bϕ. However, as there are two Iϕ clock cycles in a single Bϕ clock cycle when Iϕ: Bϕ = 4:2, transfer of data from the L bus to the I bus takes (3 + n) × Iϕ (n = 0 to 1) (3 × Iϕ is indicated in figure 9.53). The relation between the timing of data output to the L bus and the rising edge of Bϕ depends on the state of program execution. In the case shown in the figure, where n = 0 and m = 0, the time required for access is 3 × Iϕ + 1 × Bϕ + 2 × Pϕ.

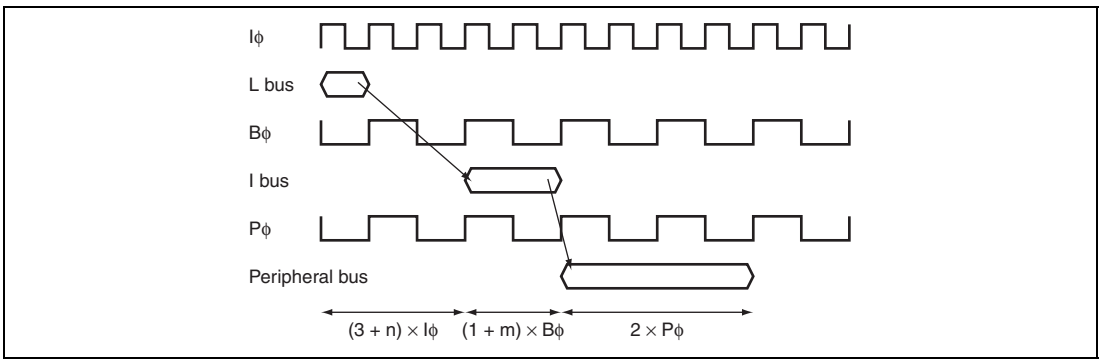


Figure 9.53 Timing of Write Access to On-Chip Peripheral I/O Registers When Iϕ:Bϕ:Pϕ = 4:2:2

Figure 9.54 shows an example of timing of read access to the peripheral bus when Iϕ:Bϕ:Pϕ = 4:2:1. Transfer from the L bus to the peripheral bus is performed in the same way as for writing. In the case of reading, however, values output onto the peripheral bus need to be transferred to the CPU. Although transfers from the peripheral bus to the I bus and from the I bus to the L bus are performed in synchronization with the rising edge of the respective bus clocks, a period of 2 × Iϕ is actually required because Iϕ ≥ Bϕ ≥ Pϕ. In the case shown in the figure, where n = 0 and m = 1, the time required for access is 3 × Iϕ + 2 × Bϕ + 2 × Pϕ + 2 × Iϕ.

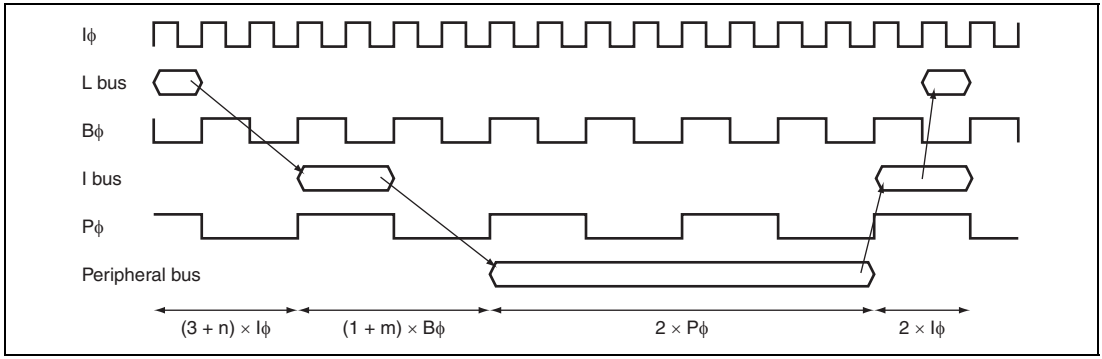


Figure 9.54 Timing of Read Access to On-Chip Peripheral I/O Registers When Iϕ:Bϕ:Pϕ = 4:2:1

9.5.17 Access to External Memory by CPU

Table 9.36 shows the number of cycles required for access to the external memory by the CPU. As the table shows, the number of cycles varies with the clock ratio, the access size, the external bus width of the LSI, and the setting for wait insertion. For details on the wait-insertion setting, see section 9.4, Register Descriptions.

Table 9.36 Number of External Access Cycles

External Bus Width	Access Size	Write/Read	Number of Access Cycles
8 bits	Byte	Write	(1 + n) × Iϕ + (3 + m) × Bϕ
		Read	(1 + n) × Iϕ + (3 + m) × Bϕ + 1 × Iϕ
	Word	Write	(1 + n) × Iϕ + (3 + m) × Bϕ + 1 × (2 + o) × Bϕ
		Read	(1 + n) × Iϕ + (3 + m) × Bϕ + 1 × (2 + o) × Bϕ + 1 × Iϕ
	Longword	Write	(1 + n) × Iϕ + (3 + m) × Bϕ + 3 × (2 + o) × Bϕ
		Read	(1 + n) × Iϕ + (3 + m) × Bϕ + 3 × (2 + o) × Bϕ + 1 × Iϕ
16 bits	Byte/Word	Write	(1 + n) × Iϕ + (3 + m) × Bϕ
		Read	(1 + n) × Iϕ + (3 + m) × Bϕ + 1 × Iϕ
	Longword	Write	(1 + n) × Iϕ + (3 + m) × Bϕ + 1 × (2 + o) × Bϕ
		Read	(1 + n) × Iϕ + (3 + m) × Bϕ + 1 × (2 + o) × Bϕ + 1 × Iϕ
32 bits	Byte/Word/ Longword	Write	(1 + n) × Iϕ + (3 + m) × Bϕ
		Read	(1 + n) × Iϕ + (3 + m) × Bϕ + 1 × Iϕ

Note: n: When Iϕ:Bϕ = 8:1, n = 0 to 7.
When Iϕ:Bϕ = 4:1, n = 0 to 3.
When Iϕ:Bϕ = 3:1, n = 0 to 2.
When Iϕ:Bϕ = 2:1, n = 0 to 1.
When Iϕ:Bϕ = 1:1, n = 0.
m, o: m: Wait setting, o: Wait setting + idle setting
For details, see section 9.4, Register Descriptions.

Synchronous logic and a layered bus structure have been adopted for this LSI circuit. Data on each bus are input and output in synchronization with rising edges of the corresponding clock signal. The L bus and I bus are synchronized with the Iϕ and Bϕ clocks, respectively. Figure 9.55 shows an example of the timing of write access to a word of data over the external bus, with a bus-width of 8 bits, when Iϕ:Bϕ = 2:1. Once the CPU has output the data to the L bus, data are transferred to the I bus in synchronization with rising edges of Bϕ. There are two Iϕ clock cycles in a single Bϕ clock cycle when Iϕ: Bϕ = 2:1. Thus, when Iϕ: Bϕ = 2:1, data transfer from the L bus to the I bus takes (1 + n) × Iϕ (n = 0 to 1) (2 × Iϕ is indicated in figure 9.55). The relation between the timing of data output to the L bus and the rising edge of Bϕ depends on the state of program execution. Data output to the I bus are transferred to the external bus after one cycle of Bϕ. External access to each data takes at least two cycles, and this can be prolonged by the BSC register settings (m and o in the formulae for number of access cycles). In the case shown in figure 9.55, word data is written with a bus-width of 8 bits, two external accesses are required. In the example, since n = 1, m = 0, and o = 0, access takes 2 × Iϕ + 3 × Bϕ + 2 × Bϕ.

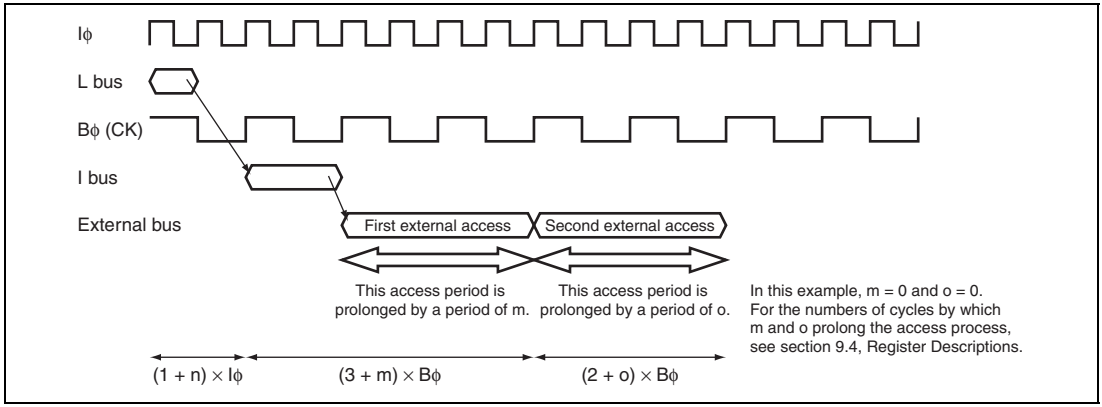


Figure 9.55 Timing of Write Access to Word Data in External Memory When Iϕ:Bϕ = 2:1 and External Bus Width Is 8 Bits

Figure 9.56 shows an example of the timing of read access when the external bus width is greater than or equal to the data width and Iϕ:Bϕ = 4:1. Transfer from the L bus to the external bus is performed in the same way as for write access. In the case of reading, however, values output onto the external bus must be transferred to the CPU. Transfers from the external bus to the I bus and from the I bus to the L bus are again performed in synchronization with rising edges of the respective bus clocks. In the actual operation, transfer from the external bus to the L bus takes one ϕ period. In the case shown in the figure, where n = 2 and m = 0, access takes 3 × Iϕ + 3 × Bϕ + 1 × Iϕ.

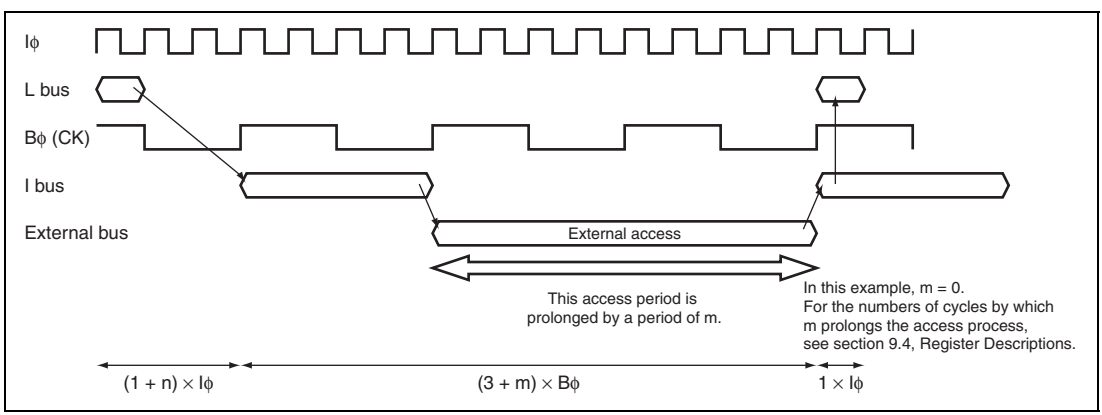


Figure 9.56 Timing of Read Access with Condition Iϕ:Bϕ = 4:1 and External Bus Width ≥ Data Width

For access by the DMAC or the DTC, the access cycles are obtained by subtracting the cycles of Iϕ required for L-bus access from the access cycles required for access by the CPU.

Section 10 Direct Memory Access Controller (DMAC)

This LSI includes the direct memory access controller (DMAC).

The DMAC can be used in place of the CPU to perform high-speed transfers between external devices that have DACK (transfer request acknowledge signal), external memory, on-chip memory, memory-mapped external devices, and on-chip peripheral modules.

10.1 Features

- Four channels (external requests can be received)
- 4-Gbyte physical address space
- Data transfer unit is selectable: Byte, word (2 bytes), longword (4 bytes), and 16 bytes (longword × 4)
- Maximum transfer count: 16,777,216 transfers
- Address mode: Dual address mode or single address mode can be selected.
- Transfer requests:
  External request, on-chip peripheral module request, or auto request can be selected.
- Selectable bus modes:
  Cycle steal mode (normal mode and intermittent mode) or burst mode can be selected.
- Selectable channel priority levels:
  The channel priority levels are selectable between fixed mode and round-robin mode.
- Interrupt request: An interrupt request can be generated to the CPU after transfers end by the specified counts.
- External request detection: There are following four types of DREQ input detection.
  — Low level detection
  — High level detection
  — Rising edge detection
  — Falling edge detection
- Transfer request acknowledge signal:
  Active levels for DACK can be set independently.

Figure 10.1 shows the block diagram of the DMAC.

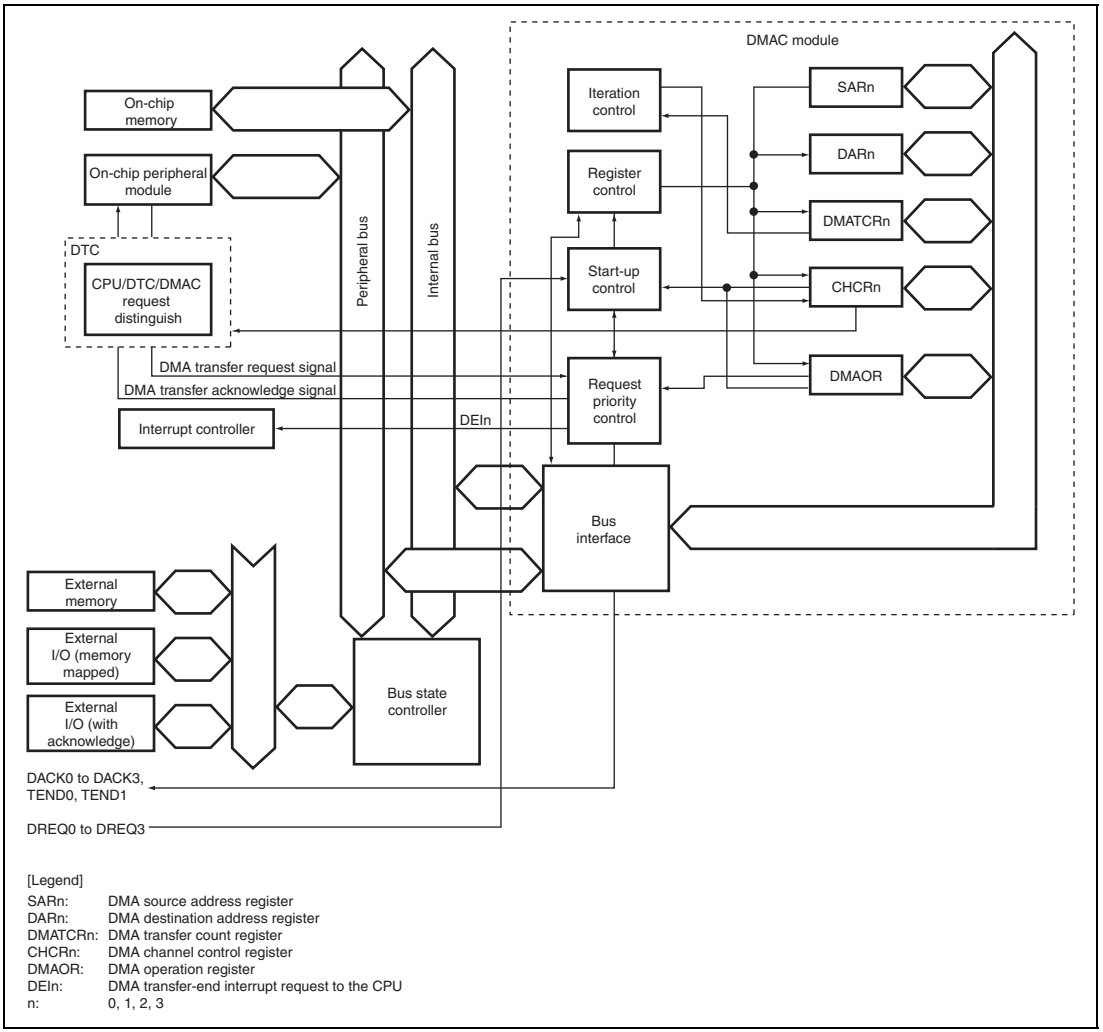


Figure 10.1 Block Diagram of DMAC

10.2 Input/Output Pins

The external pins for the DMAC are described below. Table 10.1 lists the configuration of the pins that are connected to external bus. The DMAC has pins for 4 channels for external bus use.

Table 10.1 Pin Configuration

Channel	Pin Name	Symbol	I/O	Function
0	DMA transfer request	DREQ0	Input	DMA transfer request input from external device to channel 0
	DMA transfer request acknowledge	DACK0	Output	DMA transfer request acknowledge output from channel 0 to external device
	DMA transfer end	TEND0	Output	DMA transfer end output for channel 0
1	DMA transfer request	DREQ1	Input	DMA transfer request input from external device to channel 1
	DMA transfer request acknowledge	DACK1	Output	DMA transfer request acknowledge output from channel 1 to external device
	DMA transfer end	TEND1	Output	DMA transfer end output for channel 1
2	DMA transfer request	DREQ2	Input	DMA transfer request input from external device to channel 2
	DMA transfer request acknowledge	DACK2	Output	DMA transfer request acknowledge output from channel 2 to external device
3	DMA transfer request	DREQ3	Input	DMA transfer request input from external device to channel 3
	DMA transfer request acknowledge	DACK3	Output	DMA transfer request acknowledge output from channel 3 to external device

10.3 Register Descriptions

The DMAC has the following registers. See section 27, List of Registers, for the addresses of these registers and the state of them in each processing status. The SAR for channel 0 is expressed such as SAR_0.

Table 10.2 Register Configuration

Channel	Register Name	Abbreviation	R/W	Initial value	Address	Access Size
0	DMA source address register_0	SAR_0	R/W	H'00000000	H'FFFFEB20	16, 32
	DMA destination address register_0	DAR_0	R/W	H'00000000	H'FFFFEB24	16, 32
	DMA transfer count register_0	DMATCR_0	R/W	H'00000000	H'FFFFEB28	16, 32
	DMA channel control register_0	CHCR_0	R/W	H'00000000	H'FFFFEB2C	8, 16, 32
1	DMA source address register_1	SAR_1	R/W	H'00000000	H'FFFFEB30	16, 32
	DMA destination address register_1	DAR_1	R/W	H'00000000	H'FFFFEB34	16, 32
	DMA transfer count register_1	DMATCR_1	R/W	H'00000000	H'FFFFEB38	16, 32
	DMA channel control register_1	CHCR_1	R/W	H'00000000	H'FFFFEB3C	8, 16, 32
2	DMA source address register_2	SAR_2	R/W	H'00000000	H'FFFFEB40	16, 32
	DMA destination address register_2	DAR_2	R/W	H'00000000	H'FFFFEB44	16, 32
	DMA transfer count register_2	DMATCR_2	R/W	H'00000000	H'FFFFEB48	16, 32
	DMA channel control register_2	CHCR_2	R/W	H'00000000	H'FFFFEB4C	8, 16, 32

Channel	Register Name	Abbreviation	R/W	Initial value	Address	Access Size
3	DMA source address register_3	SAR_3	R/W	H'00000000	H'FFFFEB50	16, 32
	DMA destination address register_3	DAR_3	R/W	H'00000000	H'FFFFEB54	16, 32
	DMA transfer count register_3	DMATCR_3	R/W	H'00000000	H'FFFFEB58	16, 32
	DMA channel control register_3	CHCR_3	R/W	H'00000000	H'FFFFEB5C	8, 16, 32
Common	DMA operation register	DMAOR	R/W	H'0000	H'FFFFEB60	8, 16
	Bus function extending register	BSCEHR	R/W	H'0000	H'FFFFE89A	8, 16

10.3.1 DMA Source Address Registers_0 to _3 (SAR_0 to SAR_3)

SAR are 32-bit readable/writable registers that specify the source address of a DMA transfer. During a DMA transfer, these registers indicate the next source address. When the data is transferred from an external device with the DACK in single address mode, the SAR is ignored.

To transfer data in 16 bits or in 32 bits, specify the address with 16-bit or 32-bit address boundary. When transferring data in 16-byte units, a 16-byte boundary must be set for the source address value. The initial value is undefined.

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

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

10.3.2 DMA Destination Address Registers_0 to _3 (DAR_0 to DAR_3)

DAR are 32-bit readable/writable registers that specify the destination address of a DMA transfer. During a DMA transfer, these registers indicate the next destination address. When the data is transferred from an external device with the DACK in single address mode, the DAR is ignored.

To transfer data in 16 bits or in 32 bits, specify the address with 16-bit or 32-bit address boundary. When transferring data in 16-byte units, a 16-byte boundary must be set for the destination address value. The initial value is undefined.

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

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

10.3.3 DMA Transfer Count Registers_0 to _3 (DMATCR_0 to DMATCR_3)

DMATCR are 32-bit readable/writable registers that specify the DMA transfer count. The number of transfers is 1 when the setting is H'00000001, 16,777,215 when H'00FFFFFF is set, and 16,777,216 (the maximum) when H'00000000 is set. During a DMA transfer, these registers indicate the remaining transfer count.

The upper eight bits of DMATCR are always read as 0, and the write value should always be 0. To transfer data in 16 bytes, one 16-byte transfer (128 bits) counts one. The initial value is undefined.

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

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

10.3.4 DMA Channel Control Registers_0 to _3 (CHCR_0 to CHCR_3)

CHCR are 32-bit readable/writable registers that control the DMA transfer mode.

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DM[1:0]		SM[1:0]		RS[3:0]				DL	DS	TB	TS[1:0]		IE	TE	DE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/(W)*	R/W

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23	DO	0	R/W	DMA Overrun Selects whether DREQ is detected by overrun 0 or by overrun 1. 0: Detects DREQ by overrun 0 1: Detects DREQ by overrun 1
22	TL	0	R/W	Transfer End Level This bit specifies the TEND signal output is high active or low active. 0: Low-active output of TEND 1: High-active output of TEND
21, 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19	—	0	R	Reserved Undefined value is set when the DMAC is activated.
18	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
17	AM	0	R/W	Acknowledge Mode Selects whether DACK is output in data read cycle or in data write cycle in dual address mode. In single address mode, DACK is always output regardless of the specification by this bit. 0: DACK output in read cycle (dual address mode) 1: DACK output in write cycle (dual address mode)
16	AL	0	R/W	Acknowledge Level Specifies whether the DACK signal output is high active or low active. 0: Low-active output of DACK 1: High-active output of DACK
15, 14	DM[1:0]	00	R/W	Destination Address Mode 1, 0 Specify whether the DMA destination address is incremented, decremented, or left fixed. (In single address mode, the DM1 and DM0 bits are ignored when data is transferred to an external device with DACK.) 00: Fixed destination address (setting prohibited in 16-byte transfer) 01: Destination address is incremented (+1 in byte-unit transfer, +2 in word-unit transfer, +4 in longword-unit transfer, +16 in 16-byte transfer) 10: Destination address is decremented (–1 in byte-unit transfer, –2 in word-unit transfer, –4 in longword-unit transfer; setting prohibited in 16-byte transfer) 11: Setting prohibited
13, 12	SM[1:0]	00	R/W	Source Address Mode 1, 0 Specify whether the DMA source address is incremented, decremented, or left fixed. (In single address mode, SM1 and SM0 bits are ignored when data is transferred from an external device with DACK.) 00: Fixed source address (setting prohibited in 16-byte transfer) 01: Source address is incremented (+1 in byte-unit transfer, +2 in word-unit transfer, +4 in longword-unit transfer, +16 in 16-byte transfer) 10: Source address is decremented (–1 in byte-unit transfer, –2 in word-unit transfer, –4 in longword-unit transfer; setting prohibited in 16-byte transfer) 11: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Descriptions
11 to 8	RS[3:0]	0000	R/W	Resource Select 3 to 0 Specify which transfer requests will be sent to the DMAC. The changing of transfer request source should be done in the state that the DMA enable bit (DE) is set to 0.

0	0	0	0	External request, dual address mode
0	0	0	1	Setting prohibited
0	0	1	0	External request, single address mode External address space → External device with DACK
0	0	1	1	External request, single address mode External device with DACK → External address space
0	1	0	0	Auto request
0	1	0	1	Setting prohibited
0	1	1	0	MTU2 (TGIA_0)
0	1	1	1	MTU2 (TGIA_1)
1	0	0	0	MTU2 (TGIA_2)
1	0	0	1	MTU2 (TGIA_3)
1	0	1	0	MTU2 (TGIA_4)
1	0	1	1	A/D_1 (ADI_1)
1	1	0	0	SCI_0 (TXI_0)
1	1	0	1	SCI_0 (RXI_0)
1	1	1	0	SCI_1 (TXI_1)
1	1	1	1	SCI_1 (RXI_1)

7	DL	0	R/W	DREQ Level and DREQ Edge Select
6	DS	0	R/W	Specify the detecting method of the DREQ pin input and the detecting level. If the transfer request source is specified as an on-chip peripheral module or if an auto-request is specified, these bits are invalid. 00: DREQ detected in low level 01: DREQ detected at falling edge 10: DREQ detected in high level 11: DREQ detected at rising edge

Bit	Bit Name	Initial Value	R/W	Descriptions
5	TB	0	R/W	Transfer Bus Mode Specifies the bus mode when DMA transfers data. 0: Cycle steal mode 1: Burst mode Note: When performing DMA transfer in burst mode with MTU2 activation request, set the corresponding bit among DMMTU4 to DMMTU0 in bus function extending register (BSCEHR) (see section 9.4.8, Bus Function Extending Register (BSCEHR)).
4, 3	TS[1:0]	00	R/W	Transfer Size 1, 0 Specify the size of data to be transferred. Select the size of data to be transferred when the source or destination is an on-chip peripheral module register of which transfer size is specified. 00: Byte size 01: Word size (2 bytes) 10: Longword size (4 bytes) 11: 16-byte unit (four longword transfers)
2	IE	0	R/W	Interrupt Enable Specifies whether or not an interrupt request is generated to the CPU at the end of the DMA transfer. Setting this bit to 1 generates an interrupt request (DEI) to the CPU when the TE bit is set to 1. 0: Interrupt request is disabled. 1: Interrupt request is enabled.

Bit	Bit Name	Initial Value	R/W	Descriptions
1	TE	0	R/(W)*	Transfer End Flag Shows that DMA transfer ends. The TE bit is set to 1 when data transfer ends when DMATCR becomes to 0. The TE bit is not set to 1 in the following cases. • DMA transfer ends due to an NMI interrupt or DMA address error before DMATCR is cleared to 0. • DMA transfer is ended by clearing the DE bit and DME bit in the DMA operation register (DMAOR). To clear the TE bit, the TE bit should be written to 0 after reading 1. Even if the DE bit is set to 1 while this bit is set to 1, transfer is not enabled. 0: During the DMA transfer or DMA transfer has been interrupted [Clearing condition] Writing 0 after TE = 1 read 1: DMA transfer ends by the specified count (DMATCR = 0)
0	DE	0	R/W	DMA Enable Enables or disables the DMA transfer. In auto request mode, DMA transfer starts by setting the DE bit and DME bit in DMAOR to 1. In this time, all of the bits TE, NMIF, and AE in DMAOR must be 0. In an external request or peripheral module request, DMA transfer starts if DMA transfer request is generated by the devices or peripheral modules after setting the bits DE and DME to 1. In this case, however, all of the bits TE, NMIF, and AE must be 0, which is the same as in the case of auto request mode. Clearing the DE bit to 0 can terminate the DMA transfer. 0: DMA transfer disabled 1: DMA transfer enabled

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

10.3.5 DMA Operation Register (DMAOR)

DMAOR is a 16-bit readable/writable register that specifies the priority level of channels at the DMA transfer. This register shows the DMA transfer status.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	CMS[1:0]		-	-	PR[1:0]		-	-	-	-	-	AE	NMIF	DME
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R	R/(W)*	R/(W)*	R/W

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	CMS[1:0]	00	R/W	Cycle Steal Mode Select 1, 0 Select either normal mode or intermittent mode in cycle steal mode. It is necessary that all channel's bus modes are set to cycle steal mode to make valid intermittent mode. 00: Normal mode 01: Setting prohibited 10: Intermittent mode 16 Executes one DMA transfer in each of 16 clocks of an external bus clock. 11: Intermittent mode 64 Executes one DMA transfer in each of 64 clocks of an external bus clock.
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9, 8	PR[1:0]	00	R/W	Priority Mode 1, 0 Select the priority level between channels when there are transfer requests for multiple channels simultaneously. 00: CH0 > CH1 > CH2 > CH3 01: CH0 > CH2 > CH3 > CH1 10: Setting prohibited 11: Round-robin mode
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	AE	0	R/(W)*	Address Error Flag Indicates that an address error occurred during DMA transfer. If this bit is set, DMA transfer is disabled even if the DE bit in CHCR and the DME bit in DMAOR are set to 1. This bit can only be cleared by writing 0 after reading 1. 0: No DMAC address error [Clearing condition] • Writing AE = 0 after AE = 1 read 1: DMAC address error occurs

Bit	Bit Name	Initial Value	R/W	Description
1	NMIF	0	R/(W)*	NMI Flag Indicates that an NMI interrupt occurred. If this bit is set, DMA transfer is disabled even if the DE bit in CHCR and the DME bit in DMAOR are set to 1. This bit can only be cleared by writing 0 after reading 1. When the NMI is input, the DMA transfer in progress can be done in one transfer unit. When the DMAC is not in operational, the NMIF bit is set to 1 even if the NMI interrupt was input. 0: No NMI interrupt [Clearing condition] • Writing NMIF = 0 after NMIF = 1 read 1: NMI interrupt occurs Note: If the NMIF bit is read at the same point in time that it is set to 1, in some cases the read value will be 0 but the internal state will be as if it was read as 1. Therefore, subsequently writing 0 to NMIF will clear it to 0 in the same way as writing 0 to the flag after reading it as 1. To prevent the NMIF bit from being cleared to 0 inadvertently, always write 1 to the NMIF bit except in cases when explicitly clearing it. To explicitly clear the NMIF bit, write 0 to it after reading it as 1. Note that if the NMIF bit is not used, there is no problem with always writing 0 to it (and writing 0 to it after reading it as 1 explicitly to clear it).
0	DME	0	R/W	DMA Master Enable Enables or disables DMA transfers on all channels. If the DME bit and the DE bit in CHCR are set to 1, transfer is enabled. In this time, all of the bits TE in CHCR, NMIF, and AE in DMAOR must be 0. If this bit is cleared during transfer, transfers in all channels are terminated. 0: Disables DMA transfers on all channels 1: Enables DMA transfers on all channels

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

10.3.6 Bus Function Extending Register (BSCEHR)

BSCEHR is a 16-bit readable/writable register that specifies the timing of bus release. It also sets the function to perform transfer by the DMAC preferentially. For details, see section 9.4.8, Bus Function Extending Register (BSCEHR).

10.4 Operation

When there is a DMA transfer request, the DMAC starts the transfer according to the predetermined channel priority; when the transfer end conditions are satisfied, it ends the transfer. Transfers can be requested in three modes: auto request, external request, and on-chip peripheral module request. In bus mode, burst mode or cycle steal mode can be selected.

10.4.1 DMA Transfer Flow

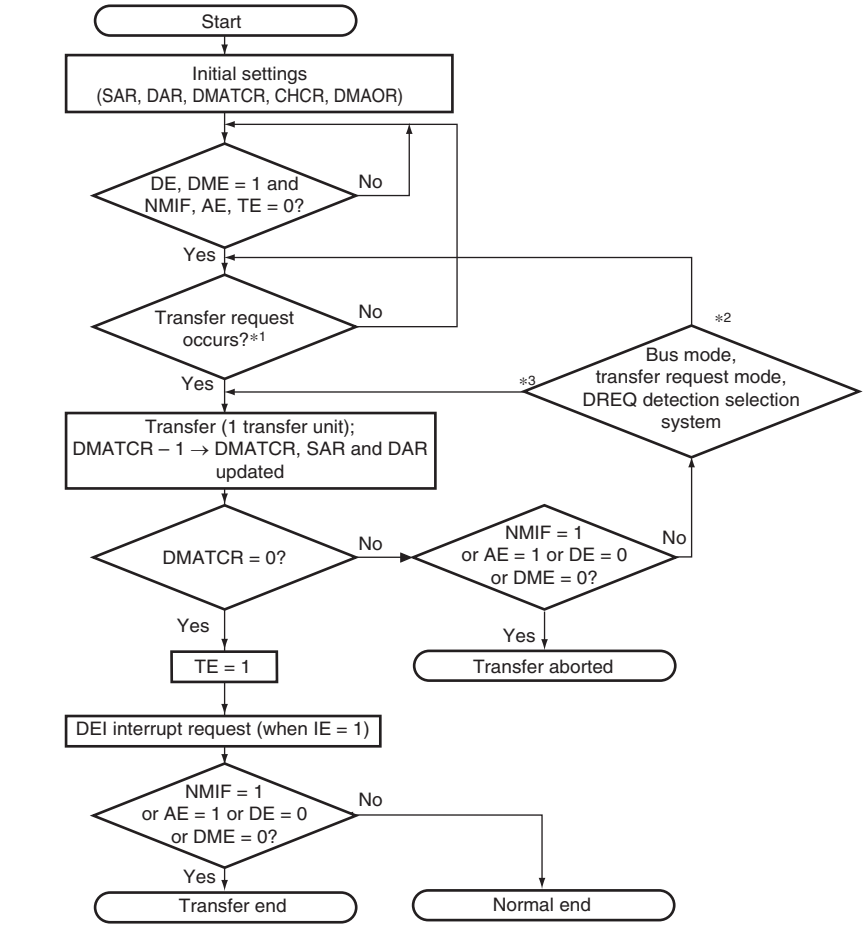
After the DMA source address registers (SAR), DMA destination address registers (DAR), DMA transfer count registers (DMATCR), DMA channel control registers (CHCR), and DMA operation register (DMAOR) are set, the DMAC transfers data according to the following procedure:

1. Checks to see if transfer is enabled (DE = 1, DME = 1, TE = 0, AE = 0, NMIF = 0)
2. When a transfer request occurs while transfer is enabled, the DMAC transfers one transfer unit of data (depending on the TS0 and TS1 settings). In auto request mode, the transfer begins automatically when the DE bit and DME bit are set to 1. The DMATCR value will be decremented for each transfer. The actual transfer flows vary by address mode and bus mode.
3. When the specified number of transfer have been completed (when DMATCR reaches 0), the transfer ends normally. If the IE bit in CHCR is set to 1 at this time, a DEI interrupt is sent to the CPU.
4. When an address error or an NMI interrupt is generated, the transfer is aborted. Transfers are also aborted when the DE bit in CHCR or the DME bit in DMAOR is changed to 0.

Notes: The state of data transfer and registers when transfer by the DMAC is interrupted

1. When DMAC address error has occurred: Data transfer is not performed. However, SAR, DAR, and DMATCR are updated.
2. When an NMI interrupt has occurred: Data transfer is stopped after transferring one transfer unit of data. SAR, DAR, and DMATCR are properly updated.
3. When the DE bit in the CHCR and the DME bit in DMAOR are cleared: Data transfer is stopped after transferring one transfer unit of data. SAR, DAR, and DMATCR are properly updated.

Figure 10.2 shows a flowchart of this procedure.



Notes: 1. In auto-request mode, transfer begins when the NMIF, AE, and TE bits are all 0 and the DE and DME bits are set to 1.
2. DREQ = level detection in burst mode (external request) or cycle-steal mode.
3. DREQ = edge detection in burst mode (external request), or auto-request mode in burst mode.

Figure 10.2 DMA Transfer Flowchart

10.4.2 DMA Transfer Requests

DMA transfer requests are basically generated in either the data transfer source or destination, but they can also be generated by external devices or on-chip peripheral modules that are neither the source nor the destination. Transfers can be requested in three modes: auto request, external request, and on-chip peripheral module request. The request mode is selected in the RS3 to RS0 bits in CHCR0 to CHCR3.

(1) Auto-Request Mode

When there is no transfer request signal from an external source, as in a memory-to-memory transfer or a transfer between memory and an on-chip peripheral module unable to request a transfer, auto-request mode allows the DMAC to automatically generate a transfer request signal internally. When the DE bits in CHCR0 to CHCR3 and the DME bit in DMAOR are set to 1, the transfer begins so long as the AE and NMIF bits in DMAOR and the TE bits in CHCR0 to CHCR3 are all 0.

(2) External Request Mode

In this mode, a transfer is performed at the request signals (DREQ0 to DREQ3) of an external device. Choose one of the modes shown in table 10.3 according to the application system. When this mode is selected, if the DMA transfer is enabled (DE = 1, DME = 1, TE = 0, AE = 0, NMIF = 0), a transfer is performed upon a request at the DREQ input.

Table 10.3 Selecting External Request Modes with RS Bits

RS3	RS2	RS1	RS0	Address Mode	Source	Destination
0	0	0	0	Dual address mode	Any	Any
		1	0	Single address mode	External memory, memory-mapped external device	External device with DACK
			1		External device with DACK	External memory, memory-mapped external device

Choose to detect DREQ by either the edge or level of the signal input with the DL bit and DS bit in CHCR_0 to CHCR_3 as shown in table 10.4. The source of the transfer request does not have to be the data transfer source or destination.

Table 10.4 Selecting External Request Detection with DL, DS Bits

CHCR_0 to CHCR_3		
DL	**DS**	**Detection of External Request**
0	0	Low level detection
	1	Falling edge detection
1	0	High level detection
	1	Rising edge detection

Note: Prior to setting CHCR_0 to CHCR_3, select the DREQ pin function by the pin function controller (PFC).

When DREQ is accepted, the DREQ pin becomes request accept disabled state. After issuing acknowledge signal DACK for the accepted DREQ, the DREQ pin again becomes request accept enabled state.

When DREQ is used by level detection, there are following two cases by the timing to detect the next DREQ after outputting DACK.

- Overrun 0: Transfer is aborted after the same number of transfer has been performed as requests.
- Overrun 1: Transfer is aborted after transfers have been performed for (the number of requests plus 1) times.

The DO bits in CHCR_0 to CHCR_3 select this overrun 0 or overrun 1.

Table 10.5 Selecting External Request Detection with DO Bit

CHCR_0 to CHCR_3	
DO	**External Request**
0	Overrun 0
1	Overrun 1

(3) On-Chip Peripheral Module Request Mode

In this mode, a transfer is performed at the transfer request signal of an on-chip peripheral module. The DMA receives ten transfer request signals in total: five compare match and input capture interrupts from multi-function timer pulse unit 2 (MTU2), receive data full interrupts (RXI) and transmit data empty interrupts (TXI) from two serial communication interface (SCI) channels, and the A/D conversion end interrupt (ADI) from the A/D converter.

When this mode is selected, if the DMA transfer is enabled (DE = 1, DME = 1, TE = 0, AE = 0, NMIF = 0), a transfer is performed upon the input of a transfer request signal.

The transfer request source does not need to be the data transfer source or destination. However, when the transmit data empty transfer request (TXI) from the SCI is specified as the transfer request source, the transfer destination must be the transmit data register (TDR) in the respective SCI channel. Similarly, when the receive data full transfer request (RXI) in the SCI is specified as the request source, the transfer source must be the receive data register (RDR) in the respective SCI channel. When the A/D conversion end transfer request (ADI) is specified as the transfer request source, the transfer source must be the respective register in the A/D converter.

Table 10.6 Selecting On-Chip Peripheral Module Request Modes with RS3 to RS0 Bits

RS3	RS2	RS1	RS0	Transfer Request Source	Transfer Request Signal	Source	Destination	Bus Mode
0	1	1	0	MTU2	TGIA_0	Any*	Any*	Burst or cycle steal
			1	MTU2	TGIA_1	Any*	Any*	Burst or cycle steal
1	0	0	0	MTU2	TGIA_2	Any*	Any*	Burst or cycle steal
			1	MTU2	TGIA_3	Any*	Any*	Burst or cycle steal
		1	0	MTU2	TGIA_4	Any*	Any*	Burst or cycle steal
			1	A/D_1	ADI1	ADDR4 to ADDR7	Any*	Cycle steal
	1	0	0	SCI_0 transmitter	TXI_0	Any*	SCTDR_0	Cycle steal
			1	SCI_0 receiver	RXI_0	SCRDR_0	Any*	Cycle steal
		1	0	SCI_1 transmitter	TXI_1	Any*	SCTDR_1	Cycle steal
			1	SCI_1 receiver	RXI_1	SCRDR_1	Any*	Cycle steal

Notes: MTU2: Multi-function timer pulse unit 2
SCI_0 and SCI_1: Serial communication interface channels 0 and 1
ADDR4 to ADDR7: A/D data register in A/D converter channel 1
SCTDR_0 and SCTDR_1: Transmit data registers in SCI_0 and SCI_1
SCRDR_0 and SCRDR_1: Receive data registers in SCI_0 and SCI_1
* An external memory, a memory-mapped external device, an on-chip memory, or an on-chip peripheral module (except DMAC, DTC, BSC, and UBC)

To output a transfer request signal from an on-chip peripheral module, set the interrupt enable bit corresponding to the transfer request signal in the on-chip peripheral module.

When an interrupt request signal in an on-chip peripheral module is used to request DMA transfer, no interrupt is requested to the CPU. For details, refer to section 6.8, Data Transfer with Interrupt Request Signals.

The transfer request signal shown in table 10.7 is automatically cancelled when the corresponding DMA transfer is performed. This cancellation occurs when one transfer unit is completed in cycle steal mode or at the end of burst transfer in burst mode.

10.4.3 Channel Priority

When the DMAC receives simultaneous transfer requests on two or more channels, it transfers data according to a predetermined priority. Two modes (fixed mode and round-robin mode) are selected by the bits PR1 and PR0 in DMAOR.

(1) Fixed Mode

In this mode, the priority levels among the channels remain fixed. There are two kinds of fixed modes as follows:

- CH0 > CH1 > CH2 > CH3
- CH0 > CH2 > CH3 > CH1

These are selected by the PR1 and the PR0 bits in DMAOR.

(2) Round-Robin Mode

In round-robin mode each time data of one transfer unit (word, byte, longword, or 16-byte unit) is transferred on one channel, the priority is rotated. The channel on which the transfer was just finished rotates to the bottom of the priority. The round-robin mode operation is shown in figure 10.3. The priority of round-robin mode is CH0 > CH1 > CH2 > CH3 immediately after reset.

When round-robin mode is specified, do not mix the cycle steal mode and the burst mode in multiple channels' bus modes.

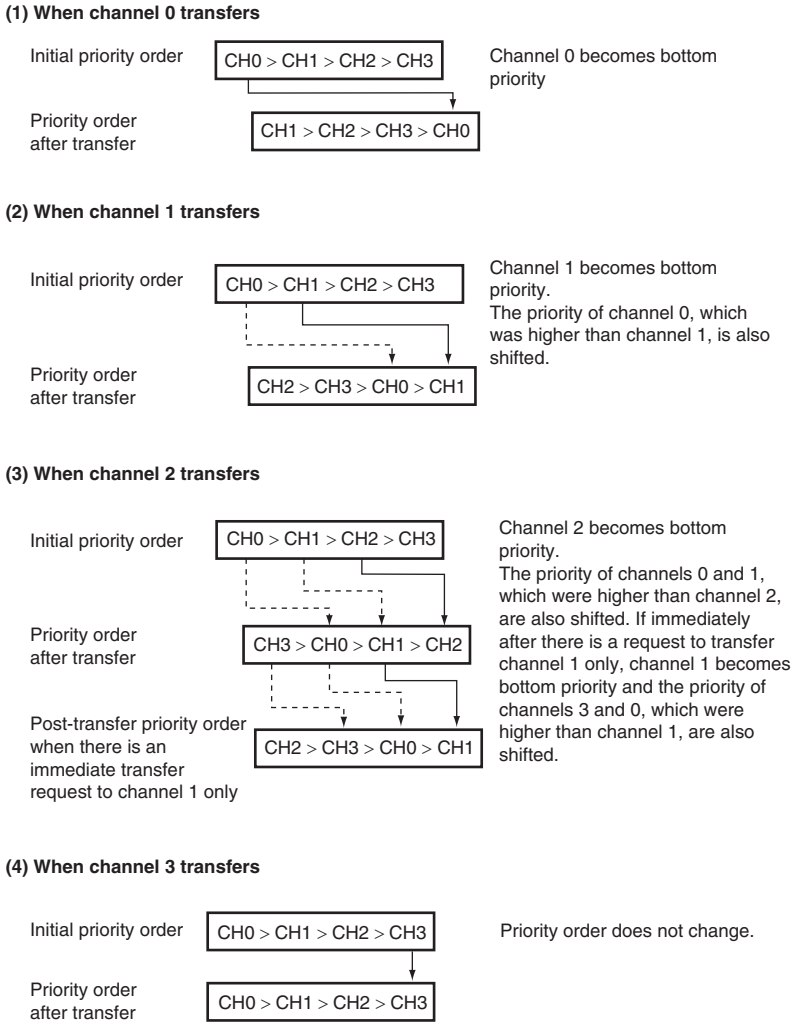


Figure 10.3 Round-Robin Mode

Figure 10.4 shows how the priority changes when channel 0 and channel 3 transfers are requested simultaneously and a channel 1 transfer is requested during the channel 0 transfer. The DMAC operates as follows:

1. Transfer requests are generated simultaneously to channels 0 and 3.
2. Channel 0 has a higher priority, so the channel 0 transfer begins first (channel 3 waits for transfer).
3. A channel 1 transfer request occurs during the channel 0 transfer (channels 1 and 3 are both waiting)
4. When the channel 0 transfer ends, channel 0 becomes lowest priority.
5. At this point, channel 1 has a higher priority than channel 3, so the channel 1 transfer begins (channel 3 waits for transfer).
6. When the channel 1 transfer ends, channel 1 becomes lowest priority.
7. The channel 3 transfer begins.
8. When the channel 3 transfer ends, channels 3 and 2 shift downward in priority so that channel 3 becomes the lowest priority.

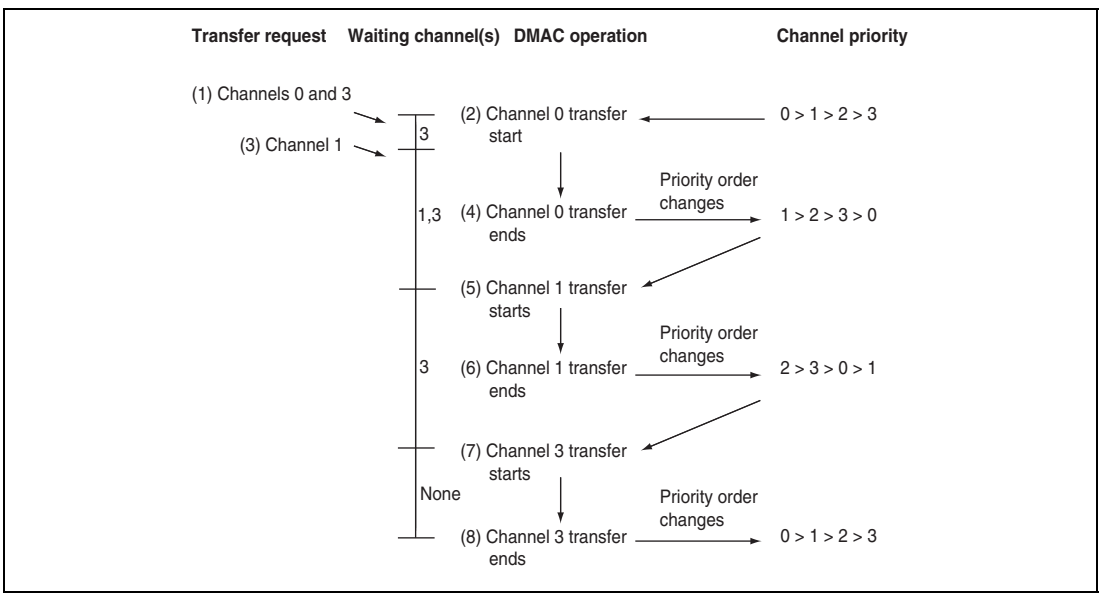


Figure 10.4 Changes in Channel Priority in Round-Robin Mode

(3) Activation priority when multiple DMAC activation requests are generated

When multiple DMAC activation requests are generated, transfer is performed in the order of activation priority. However, if multiple DMAC activation requests are generated while the DMAC is not the bus master, transfer is started for the first activation request. In addition, if activation requests are generated while the CPU is accessing an external space, transfer is started for the first activation request and then the second request. Figure 10.5 shows the example of activation priority operation of the DMAC.

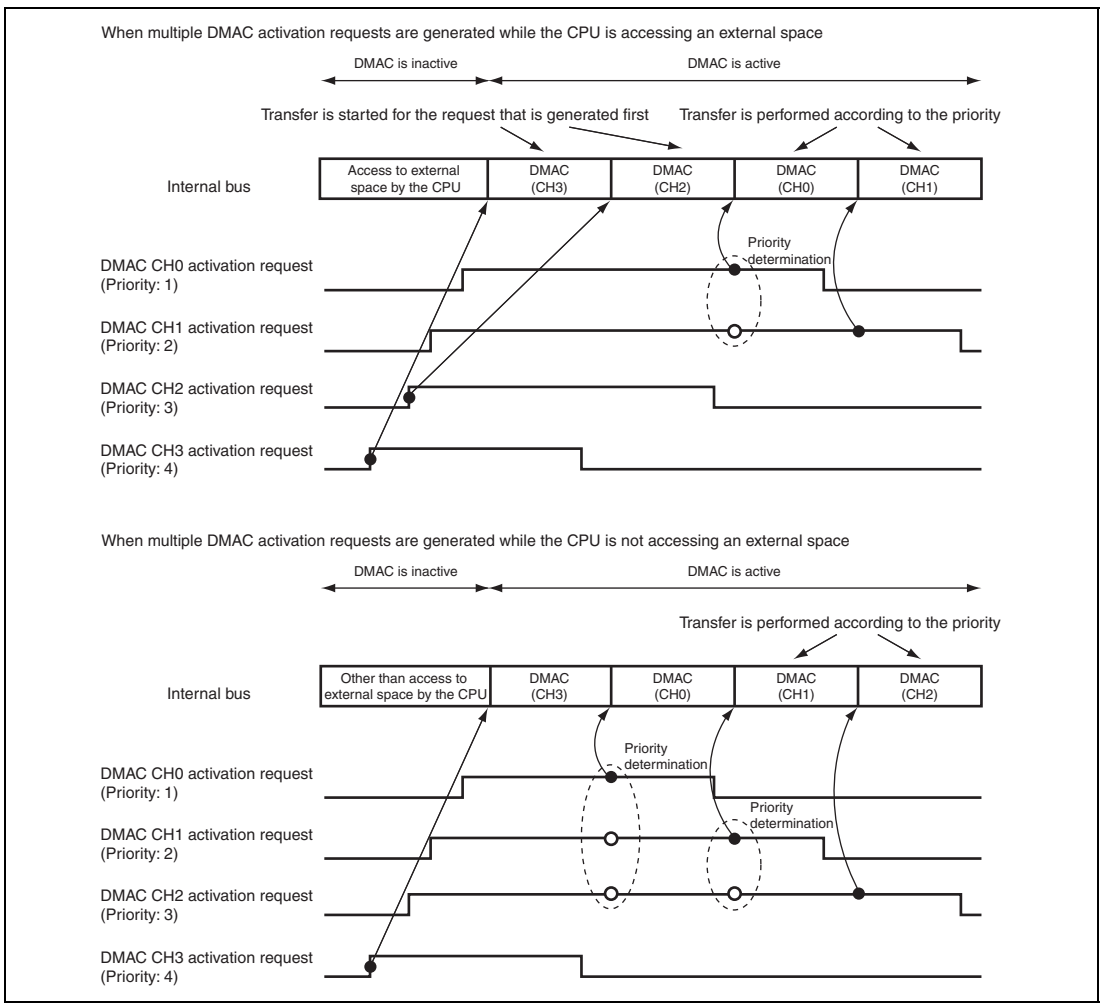


Figure 10.5 Example of Activation Priority Operation of DMAC (Priority Fixed Mode (CH0 > CH1 > CH2 > CH3))

10.4.4 DMA Transfer Types

DMA transfer has two types; single address mode transfer and dual address mode transfer. They depend on the number of bus cycles of access to source and destination. A data transfer timing depends on the bus mode, which has cycle steal mode and burst mode. The DMAC supports the transfers shown in table 10.7.

Table 10.7 Supported DMA Transfers

	Destination				
Source	**External Device with DACK**	**External Memory**	**Memory-Mapped External Device**	**On-Chip Peripheral Module**	**On-Chip Memory**
External device with DACK	Not available	Dual, single	Dual, single	Not available	Not available
External memory	Dual, single	Dual	Dual	Dual	Dual
Memory-mapped external device	Dual, single	Dual	Dual	Dual	Dual
On-chip peripheral module	Not available	Dual	Dual	Dual	Dual
On-chip memory	Not available	Dual	Dual	Dual	Dual

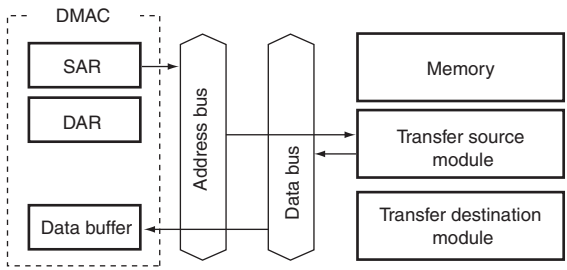
Notes: 1. Dual: Dual address mode
2. Single: Single address mode
3. For on-chip peripheral modules, 16-byte transfer is available only by registers which can be accessed in longword units.

Address Modes:

- Dual Address Mode

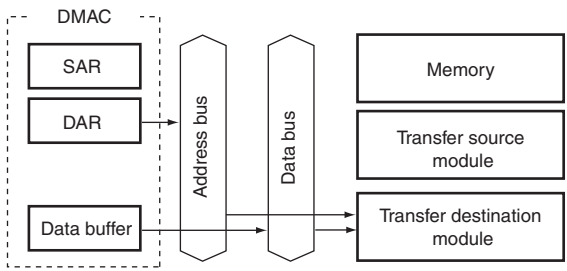
  In dual address mode, both the transfer source and destination are accessed by an address. The source and destination can be located externally or internally.

  DMA transfer requires two bus cycles because data is read from the transfer source in a data read cycle and written to the transfer destination in a data write cycle. At this time, transfer data is temporarily stored in the DMAC. In the transfer between external memories as shown in figure 10.6, data is read to the DMAC from one external memory in a data read cycle, and then that data is written to the other external memory in a write cycle.



The SAR value is an address, data is read from the transfer source module, and the data is temporarily stored in the DMAC.

First bus cycle



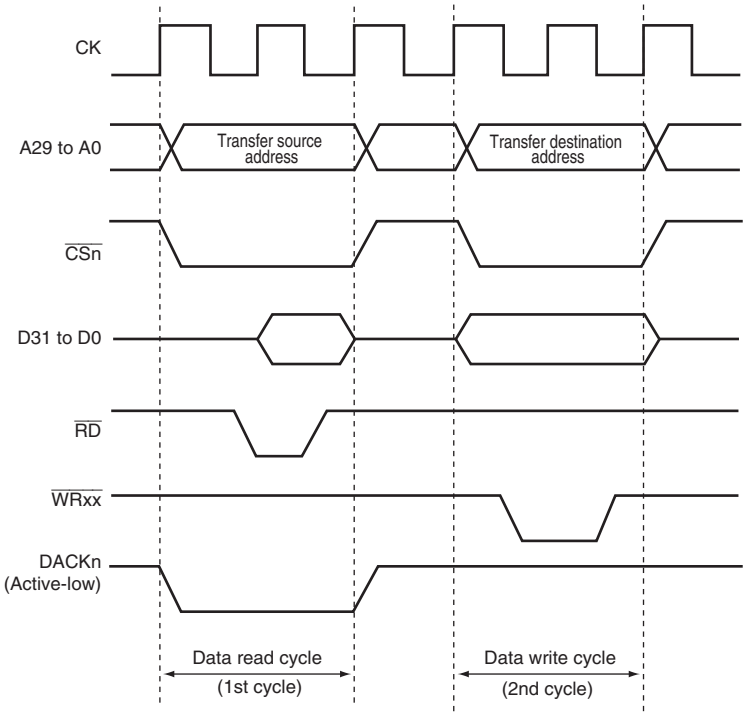
The DAR value is an address and the value stored in the data buffer in the DMAC is written to the transfer destination module.

Second bus cycle

Figure 10.6 Data Flow of Dual Address Mode

Auto request, external request, and on-chip peripheral module request are available for the transfer request. DACK can be output in read cycle or write cycle in dual address mode. The AM bit in the channel control register (CHCR) can specify whether the DACK is output in read cycle or write cycle.

Figure 10.7 shows an example of DMA transfer timing in dual address mode.



Note: In transfer between external memories, with DACK output in the read cycle, DACK output timing is the same as that of CSn.

Figure 10.7 Example of DMA Transfer Timing in Dual Mode (Source: Ordinary Memory, Destination: Ordinary Memory)

- Single Address Mode

  In single address mode, either the transfer source or transfer destination peripheral device is accessed (selected) by means of the DACK signal, and the other device is accessed by an address. In this mode, the DMAC performs one DMA transfer in one bus cycle, accessing one of the external devices by outputting the DACK transfer request acknowledge signal to it, and at the same time outputting an address to the other device involved in the transfer. For example, in the case of transfer between external memory and an external device with DACK shown in figure 10.8, when the external device outputs data to the data bus, that data is written to the external memory in the same bus cycle.

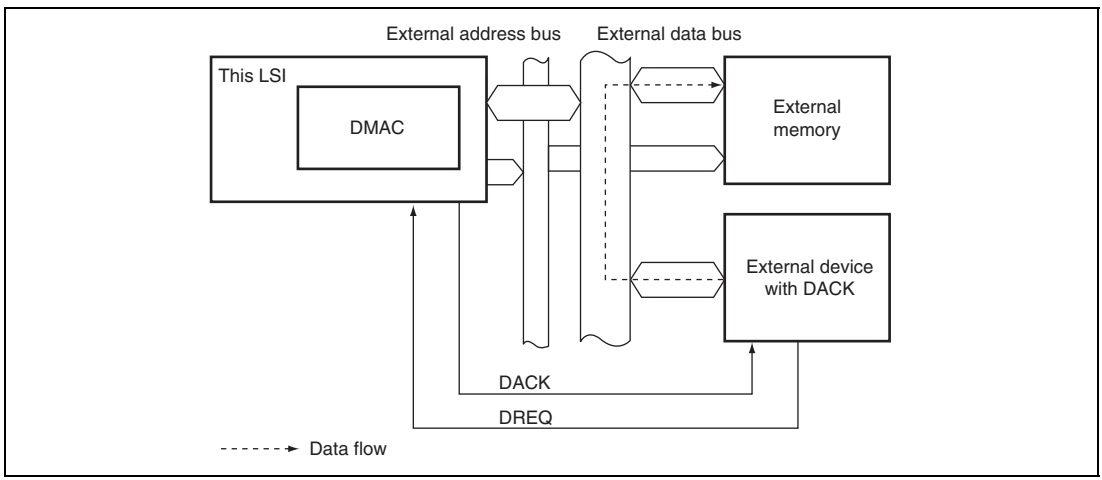


Figure 10.8 Data Flow in Single Address Mode

Two kinds of transfer are possible in single address mode: transfer between an external device with DACK and a memory-mapped external device, and transfer between an external device with DACK and external memory. In both cases, only the external request signal (DREQ) is used for transfer requests.

Figure 10.9 shows an example of DMA transfer timing in single address mode.

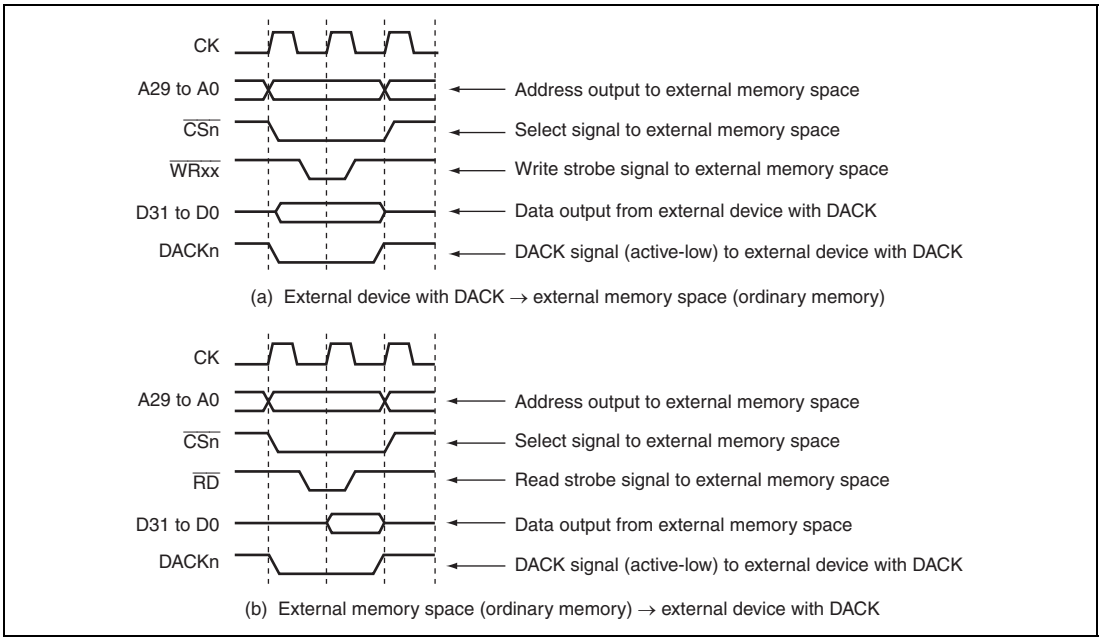


Figure 10.9 Example of DMA Transfer Timing in Single Address Mode

Bus Modes: There are two bus modes: cycle steal mode and burst mode. Select the mode in the TB bits in the channel control register (CHCR).

- Cycle-Steal Mode
  - — Normal mode

    In cycle-steal normal mode, the bus mastership is given to another bus master after a one-transfer-unit (byte, word, longword, or 16-byte unit) DMA transfer. When another transfer request occurs, the bus mastership is obtained from the other bus master and a transfer is performed for one transfer unit. When that transfer ends, the bus mastership is passed to the other bus master. This is repeated until the transfer end conditions are satisfied.

    In cycle-steal normal mode, transfer areas are not affected regardless of settings of the transfer request source, transfer source, and transfer destination.

  Figure 10.10 shows an example of DMA transfer timing in cycle-steal normal mode. Transfer conditions shown in the figure are:

  — Dual address mode

  — DREQ low level detection

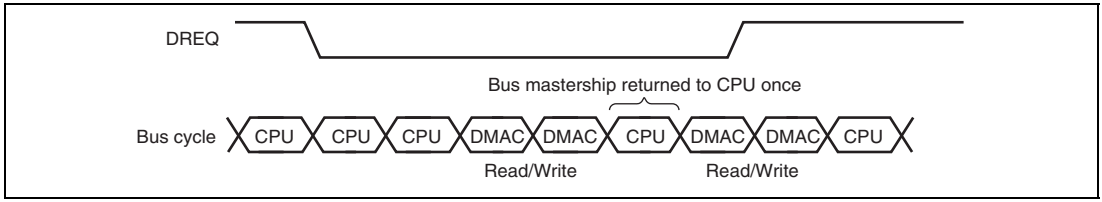


Figure 10.10 DMA Transfer Example in Cycle-Steal Normal Mode (Dual Address, DREQ Low Level Detection)

— Intermittent mode 16 and intermittent mode 64

In intermittent mode of cycle steal, the DMAC returns the bus mastership to other bus master whenever a unit of transfer (byte, word, longword, or 16-byte unit) is complete. If the next transfer request occurs after that, the DMAC gets the bus mastership from other bus master after waiting for 16 or 64 clocks in Bϕ count. The DMAC then transfers data of one unit and returns the bus mastership to other bus master. These operations are repeated until the transfer end condition is satisfied. It is thus possible to make lower the ratio of bus occupation by DMA transfer than cycle-steal normal mode.

This intermittent mode can be used for all transfer section; transfer request source, transfer source, and transfer destination. The bus modes, however, must be cycle steal mode in all channels.

Figure 10.11 shows an example of DMA transfer timing in cycle steal intermittent mode. Transfer conditions shown in the figure are:

— Dual address mode

— DREQ low level detection

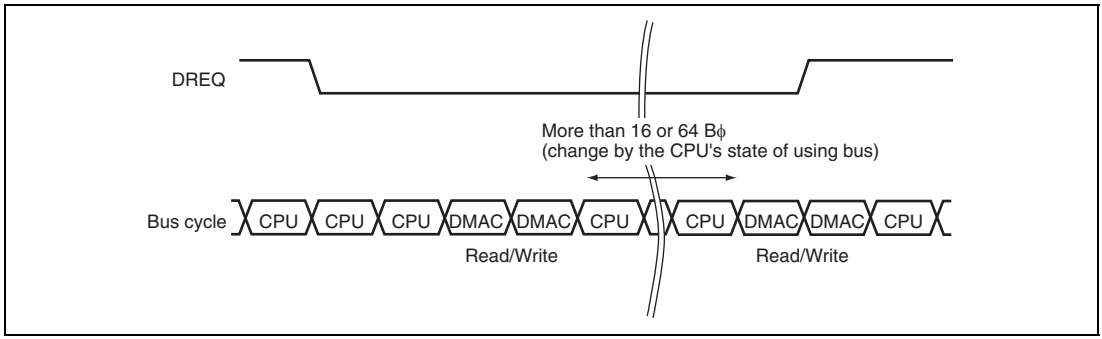


Figure 10.11 Example of DMA Transfer in Cycle Steal Intermittent Mode (Dual Address, DREQ Low Level Detection)

- Burst Mode

  In burst mode, once the DMAC obtains the bus mastership, the transfer is performed continuously without releasing the bus mastership until the transfer end condition is satisfied. In external request mode with level detection of the DREQ pin, however, when the DREQ pin is not active, the bus mastership passes to the other bus master after the DMAC transfer request that has already been accepted ends, even if the transfer end conditions have not been satisfied.

Figure 10.12 shows DMA transfer timing in burst mode.

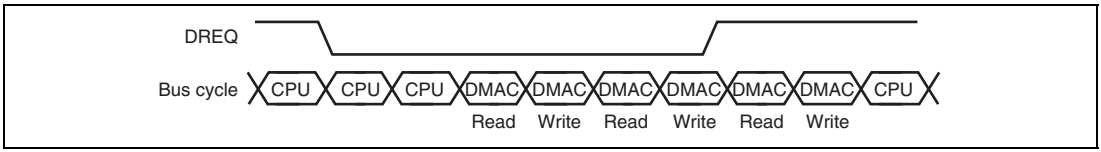


Figure 10.12 DMA Transfer Example in Burst Mode (Dual Address, DREQ Low Level Detection)

Relationship between Request Modes and Bus Modes by DMA Transfer Category: Table 10.8 shows the relationship between request modes and bus modes by DMA transfer category.

Table 10.8 Relationship between Request Modes and Bus Modes by DMA Transfer Category

Address Mode	Transfer Category	Request Mode	Bus Mode	Transfer Size (Bits)	Usable Channels
Dual	External device with DACK and external memory	External	B/C	8/16/32/128	0 to 3
	External device with DACK and memory-mapped external device	External	B/C	8/16/32/128	0 to 3
	External memory and external memory	All*1	B/C	8/16/32/128	0 to 3
	External memory and memory-mapped external device	All*1	B/C	8/16/32/128	0 to 3
	Memory-mapped external device and memory-mapped external device	All*1	B/C	8/16/32/128	0 to 3
	External memory and on-chip peripheral module	All*2	B/C*3	8/16/32/128*4	0 to 3
	Memory-mapped external device and on-chip peripheral module	All*2	B/C*3	8/16/32/128*4	0 to 3
	On-chip peripheral module and on-chip peripheral module	All*2	B/C*3	8/16/32/128*4	0 to 3
	On-chip memory and On-chip memory	All*1	B/C	8/16/32/128	0 to 3
	On-chip memory and memory-mapped external device	All*1	B/C	8/16/32/128	0 to 3
	On-chip memory and on-chip peripheral module	All*2	B/C*3	8/16/32/128*4	0 to 3
	On-chip memory and external memory	All*1	B/C	8/16/32/128	0 to 3
Single	External device with DACK and external memory	External	B/C	8/16/32	0 to 3
	External device with DACK and memory-mapped external device	External	B/C	8/16/32	0 to 3

B: Burst mode, C: Cycle steal mode

Notes: 1. External requests, auto requests, and on-chip peripheral module requests are all available. For on-chip peripheral module requests, however, the SCI and A/D converter cannot be specified as the transfer request source.

2. External requests, auto requests, and on-chip peripheral module requests are all available. However, when the SCI or A/D converter is the transfer request source, the request source register must be designated as the transfer source or the transfer destination.

3. Only cycle steal when the SCI or A/D converter is the transfer request source.

4. Access size permitted for the on-chip peripheral module register functioning as the transfer source or transfer destination.

Bus Mode and Channel Priority: When the priority is set in fixed mode (CH0 > CH1) and channel 1 is transferring in burst mode, if there is a transfer request to channel 0 with a higher priority, the transfer of channel 0 will begin immediately.

At this time, if channel 0 is also operating in burst mode, the channel 1 transfer will continue after the channel 0 transfer has completely finished.

When channel 0 is in cycle steal mode, channel 0 with a higher priority performs the transfer of one transfer unit and the channel 1 transfer is continuously performed without releasing the bus mastership. The bus mastership will then switch between the two in the order channel 0, channel 1, channel 0, and channel 1. Therefore, the bus state is such that the CPU cycle after the completion of cycle steal mode transfer has been replaced with the channel 1 burst mode transfer. (Hereinafter referred to as burst mode priority execution.)

This example is shown in figure 10.13. When multiple channels are operating in burst modes, the channel with the highest priority is executed first.

When DMA transfer is executed in the multiple channels, the bus mastership will not be given to the bus master until all competing burst transfers are complete.

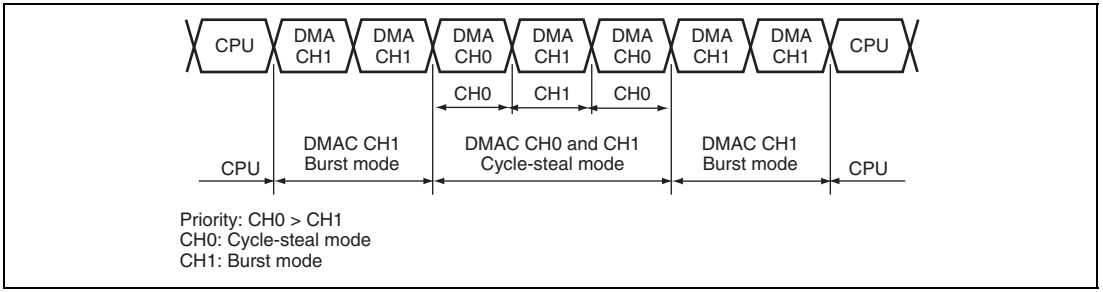


Figure 10.13 Bus State when Multiple Channels Are Operating

In round-robin mode, the priority changes according to the specification shown in figure 10.3. However, the channel in cycle steal mode cannot be mixed with the channel in burst mode.

10.4.5 Number of Bus Cycle States and DREQ Pin Sampling Timing

Number of Bus Cycle States: When the DMAC is the bus master, the number of bus cycle states is controlled by the bus state controller (BSC) in the same way as when the CPU is the bus master. For details, see section 9, Bus State Controller (BSC).

DREQ Pin Sampling Timing: Figures 10.14 to 10.17 show the sample timing of the DREQ input in each bus mode, respectively.
Determination of DMAC activation by DREQ takes 3 × Bcyc (Bcyc is the external clock (Bϕ = CK) cycle). Timing of the DACK output for the first DREQ acceptance differs depending on the internal bus state, the AM bit setting in CHCR, and the configuration of the BSC regarding the transfer source/destination areas, but the fastest case is 6 × Bcyc.

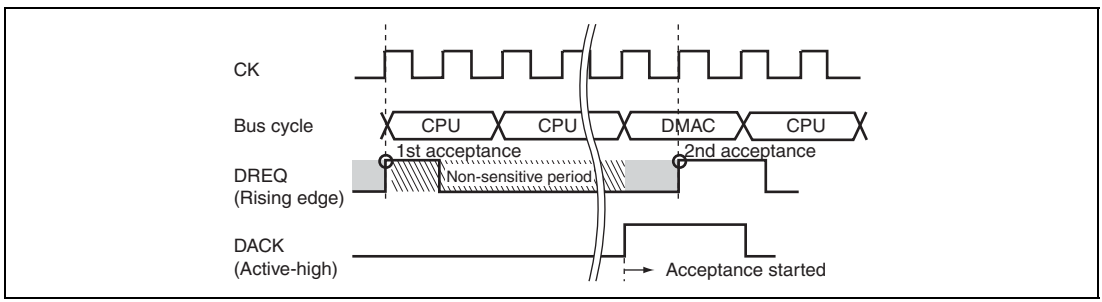


Figure 10.14 Example of DREQ Input Detection in Cycle Steal Mode Edge Detection

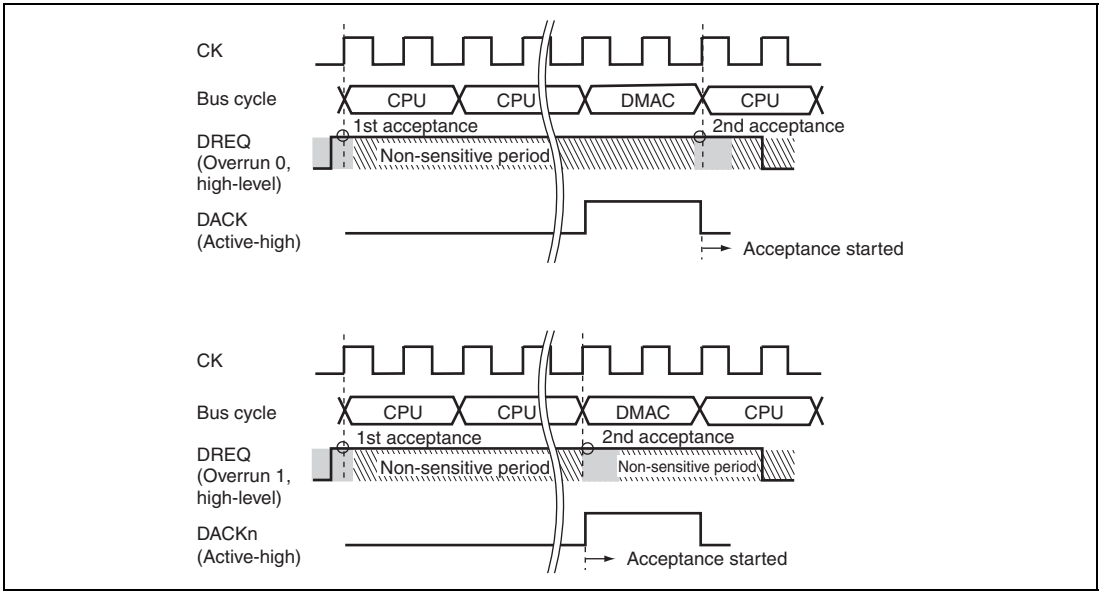


Figure 10.15 Example of DREQ Input Detection in Cycle Steal Mode Level Detection

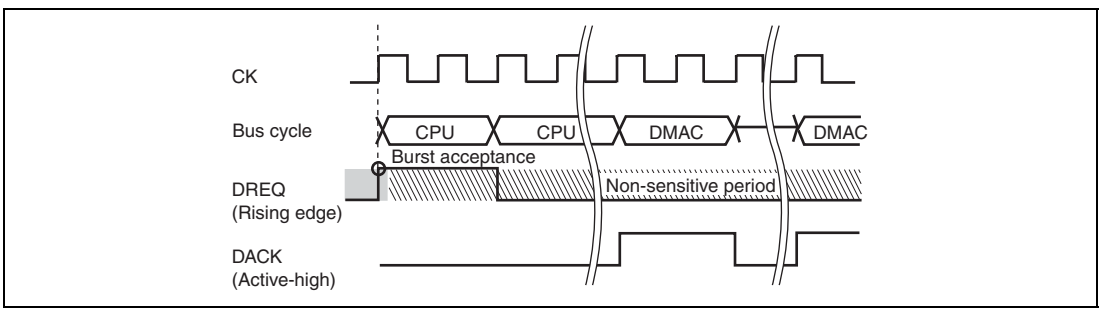


Figure 10.16 Example of DREQ Input Detection in Burst Mode Edge Detection

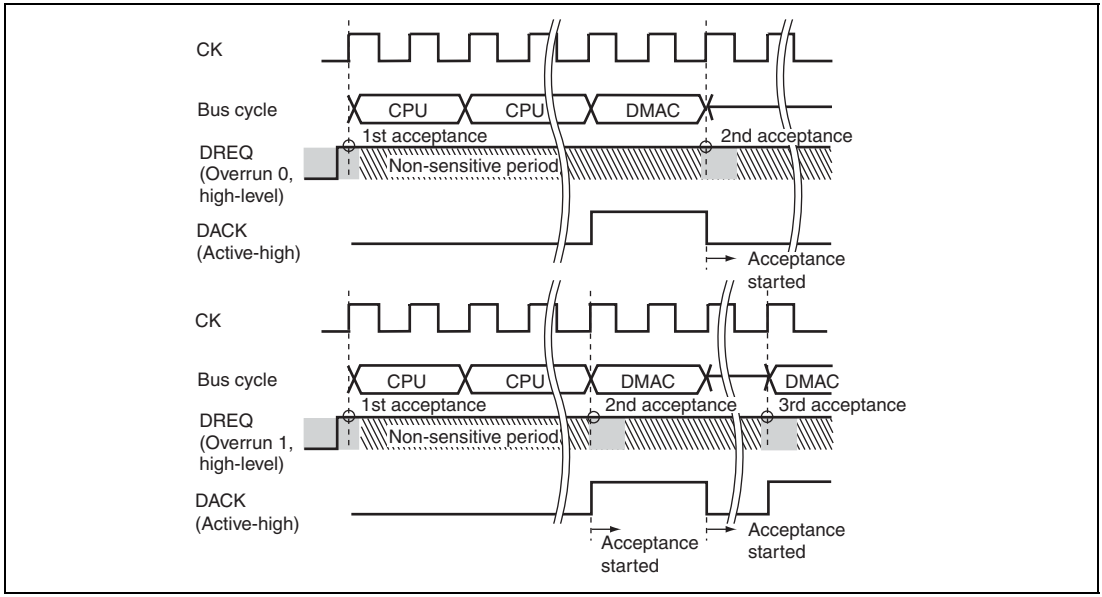


Figure 10.17 Example of DREQ Input Detection in Burst Mode Level Detection

Figure 10.18 shows the TEND output timing.

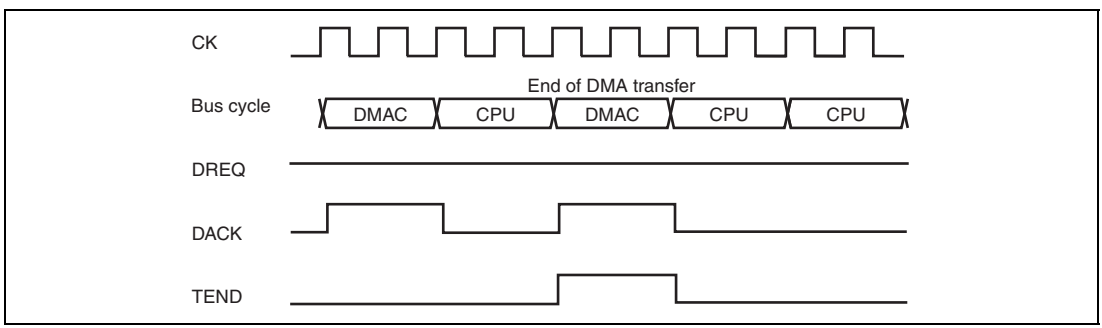
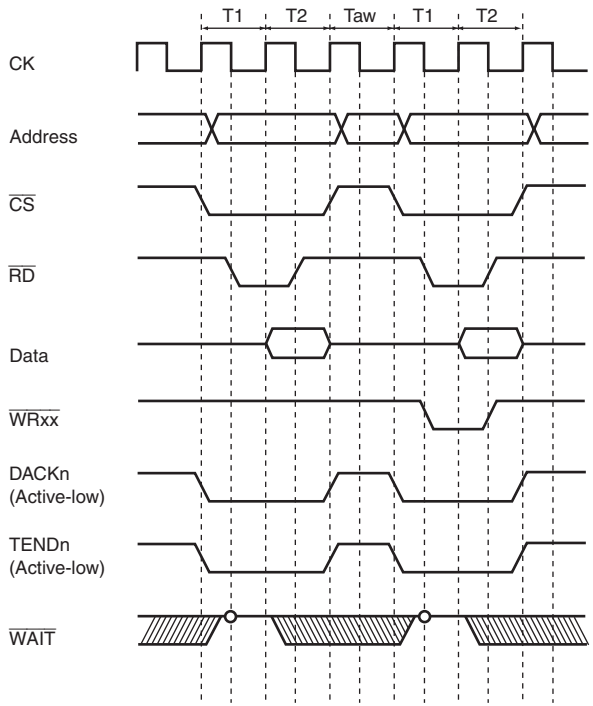


Figure 10.18 DMA Transfer End Timing (in Cycle Steal Level Detection)

When 16-byte transfer to an external device is performed, when 8-bit or 16-bit external device is accessed in longword units, or when an 8-bit external device is accessed in word units, each DMA transfer unit is divided into multiple bus cycles. If negation of $\overline{CS}$ between bus cycles is specified in these cases, the DACK and TEND outputs are also divided for data alignment. This example is illustrated in figure 10.19.

With divided DACK, sampling of DREQ is not detected correctly and a maximum of one extra overrun may occur. To avoid this, use the settings with which DACK is not divided, or in the case when DACK is divided, specify the transfer size that is smaller than the bus width of the external device.

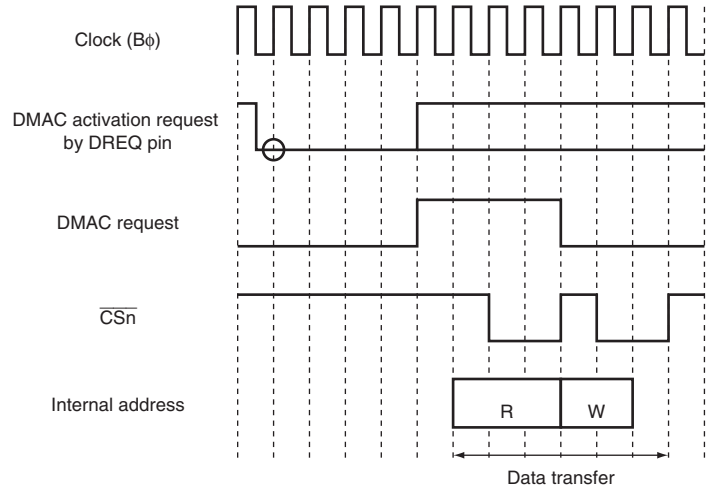


Note: The DACK and TEND are asserted for the last transfer unit of the DMA transfer. When the transfer unit is divided into several bus cycles and the CS is negated between bus cycles, the DACK and TEND are also divided.

Figure 10.19 BSC Ordinary Memory Access (No Wait, Idle Cycle 1, Longword Access to 16-Bit Device)

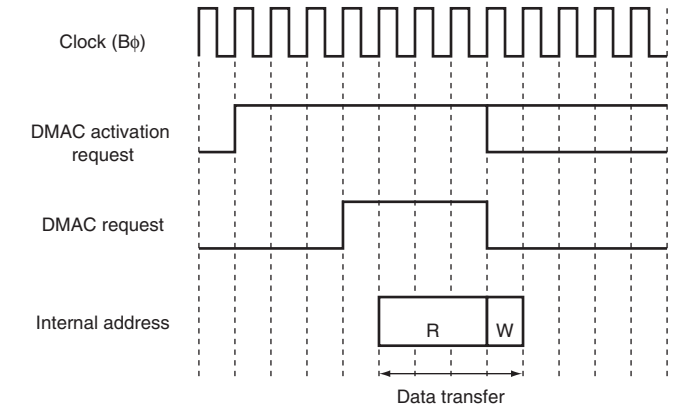
10.4.6 Operation Timing

Figures 10.20 and 10.21 illustrate the timing of DMAC operations.



Note: The DMAC request indicates the state of an internal bus request after determination of the source for DMAC activation. For details on the DREQ and DACK timing in each operating mode, see section 10.4.5, Number of Bus Cycle States and DREQ Pin Sampling Timing.

Figure 10.20 Example of Timing of DMAC Operation—Activation by DREQ (in the Case of Cycle Stealing Transfer, Dual Address Mode, Low-Level Detection, Iϕ:Bϕ:Pϕ = 1:1/2:1/2, Data Transfer from External Memory to External Memory, and Idle/Wait = 0)



Note: The DMAC request indicates the state of an internal bus request after determination of the source for DMAC activation.

Figure 10.21 Example of DMAC Operation Timing—Activation by an On-Chip Peripheral Module (in the Case of Cycle Stealing Transfer, Dual Address Mode, Low-Level Detection, Iϕ:Bϕ:Pϕ = 1:1/2:1/2, and Data Transfer from On-Chip Peripheral Module to On-Chip RAM)

10.5 Usage Notes

10.5.1 Notes on Output from DACK Pin

When burst mode and cycle steal mode are specified in multiple channels at the same time, unnecessary DACK may be asserted at the end of burst transfer if the following conditions are all satisfied.

1. When DMA transfer is performed with burst mode and cycle steal mode being specified in multiple channels at the same time.
2. When dual address mode is specified for the channel used in burst mode and DACK output is enabled in write cycles.
3. When the DMAC cannot acquire the bus mastership after the end of burst transfer while the DMAC has accepted a cycle-steal transfer request.

This can be avoided by one of the following actions.

Action 1: After confirming the end of burst transfer (TE bit = 1), perform remaining DMA transfer in cycle steal mode.

Action 2: Do not enable DACK in write cycles for the channel used in burst mode.

Action 3: When performing DMA transfer in multiple channels at the same time, select either burst mode or cycle steal mode for all channels.

10.5.2 DMA Transfer by Peripheral Modules

During DMA transfer by peripheral modules, do not set the clock ratio of bus clock (Bϕ):peripheral clock (Pϕ), bus clock (Bϕ):MTU2 clock (MPϕ), and bus clock (Bϕ):MTU2S clock (MIϕ) to 1:1/3 or 1:1/4

10.5.3 Module Standby Mode Setting

DMAC operation can be enabled or disabled by setting the standby control register. The DMAC is disabled by default. After module standby mode is canceled, access to the DMAC registers is enabled.

Note that software standby mode or module standby mode must not be entered while the DMAC is operating. Before entering software standby mode or module standby mode, return the channel control registers (CHCR_0 to CHCR_3) and DMA operation register (DMAOR) to their initial values. For details, refer to section 26, Power-Down Modes.

10.5.4 Access to DMAC and DTC Registers through DMAC

Do not access the DMAC or DTC registers through DMAC operation. Do not access the DMAC registers through DTC operation.

10.5.5 Note on SCI as DMAC Activation Source

When the TXI interrupt in SCI is specified as a DMAC activation source, the TEND flag in the SCI must not be used as the transfer end flag.

10.5.6 CHCR Setting

Before modifying the CHCR setting, be sure to clear the DE bit in the respective channel.

10.5.7 Note on Multiple Channel Activation

Do not use the same on-chip request in multiple channels.

10.5.8 Note on Transfer Request Input

Transfer requests must be input after DMAC settings are completed.

10.5.9 Conflict between NMI Interrupt and DMAC Activation

When a conflict occurs between the generation of the NMI interrupt and the DMAC activation, the NMI interrupt has priority. Thus the NMI bit is set to 1 and the DMAC is not activated.

It takes 1 × Bcyc + 3× Pcyc for determining DMAC stop by NMI, 3 × Bcyc for determining DMAC activation by DREQ, and 1 × Pcyc for determining DMAC activation by peripheral modules (Bcyc is the external bus clock cycle, and Pcyc is the peripheral clock cycle).

10.5.10 Notes on Using Peripheral Module Request Modes

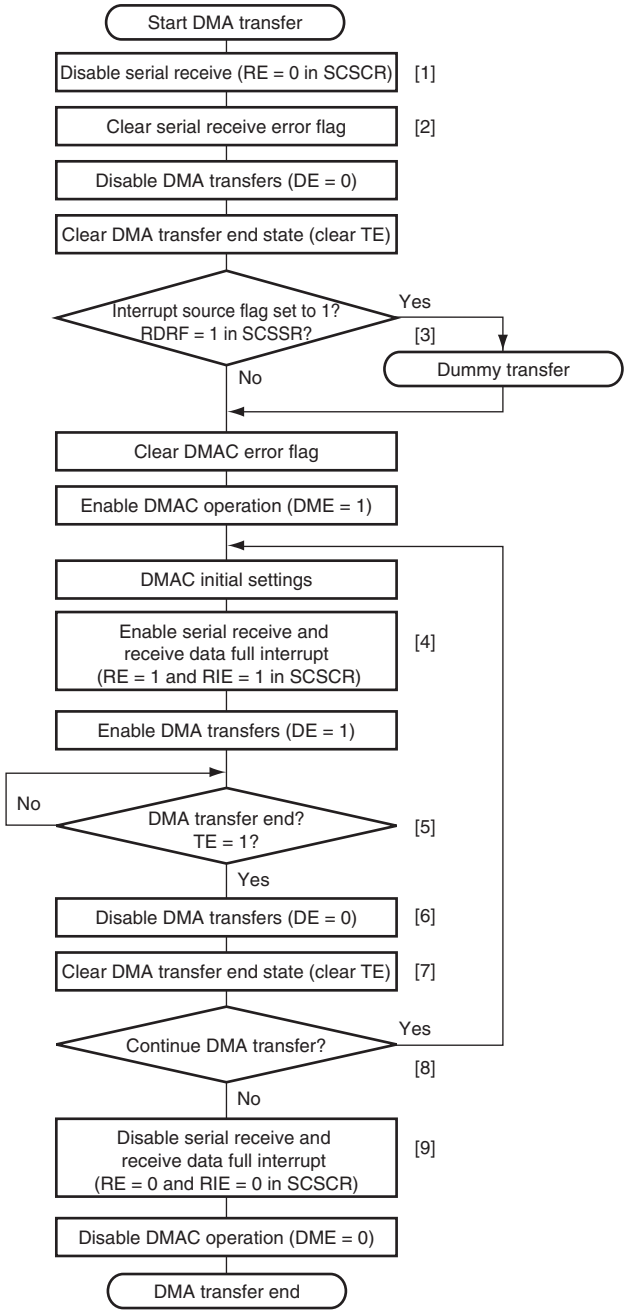
When using a DMA transfer for which A/D_1 (ADI_1), SCI_0 (TXI_0), SCI_0 (RXI_0), SCI_1 (TXI_1), or SCI_1 (RXI_1) is selected as the transfer request source, any interrupt requests from the peripheral module selected as the transfer request source are accepted and held as DMA transfer requests. As a result, a DMA transfer may occur in the absence of a DMA transfer request when one of the following conditions is met:

1. When A/D_1 (ADI_1), SCI_0 (TXI_0), SCI_0 (RXI_0), SCI_1 (TXI_1), or SCI_1 (RXI_1) is selected as the DMA transfer request source after an interrupt request (ADI_1, TXI_0, RXI_0, TXI_1, or RXI_1) was generated and the interrupt source flag (ADF in ADCSR, TDRE or RDRF in SCSSR) was cleared by the CPU or DTC.
2. When the interrupt source flag (ADF in ADCSR, TDRE or RDRF in SCSSR) is cleared by the CPU after A/D_1 (ADI_1), SCI_0 (TXI_0), SCI_0 (RXI_0), SCI_1 (TXI_1), or SCI_1 (RXI_1) was selected as the DMA transfer request source.
3. When DMA transfers are enabled (DE = 1) while interrupts are disabled (ADIE = 0 in ADCSR, TIE = 0 or RIE = 0 in SCSCR) for the peripheral module selected as the DMA transfer request source.

This problem can be avoided by using one of the following workarounds:

1. When SCI_0 (TXI_0) or SCI_1 (TXI_1) is the transfer request source
   - Enable DMA transfers (DE = 1) after confirming that the transmit data empty interrupt (TE = 1 and TIE = 1 in SCSCR and TDRE = 1 in SCSSR) has been generated.
2. When A/D_1 (ADI_1), SCI_0 (RXI_0), or SCI_1 (RXI_1) is the transfer request source
   - Before using a DMA transfer, do not use a CPU interrupt or DTC transfer that uses the same interrupt request.
   - Only clear the interrupt source flag (ADF in ADCSR or RDRF in SCSSR) by a DMA transfer. Do not clear it by the CPU or DTC. When it is necessary to clear the interrupt source flag, perform one (dummy) DMA transfer while operation of the peripheral module is disabled (ADST = 0 in ADCR or RE = 0 in SCSCR).
   - Enable DMA transfers (DE = 1) after enabling the interrupt request (ADIE = 1 in ADCSR or RIE = 1 in SCSCR).

Figure 10.22 shows an example DMA transfer sequence when SCI_0 (RXI_0) or SCI_1 (RXI_1) is selected as the transfer request source, and figure 10.23 shows an example dummy transfer sequence.



[1] [1] Disable serial receive so no interrupt (RXI) occurs during the transfer sequence.

[2] Clear the serial receive error flag. At this point, do not clear the interrupt source flag (RDRF in SCSSR).

[3] If the interrupt source flag has been set to 1, perform a dummy transfer to clear it.

[4] After enabling serial receive and the receive data full interrupt (RE = 1 and RIE = 1 in SCSCR), enable DMA transfers (DE = 1).

[5] Wait for the DMA transfer to end. If the DEI interrupt is enabled (IE = 1), a DEI interrupt will be generated when the DMA transfer ends.

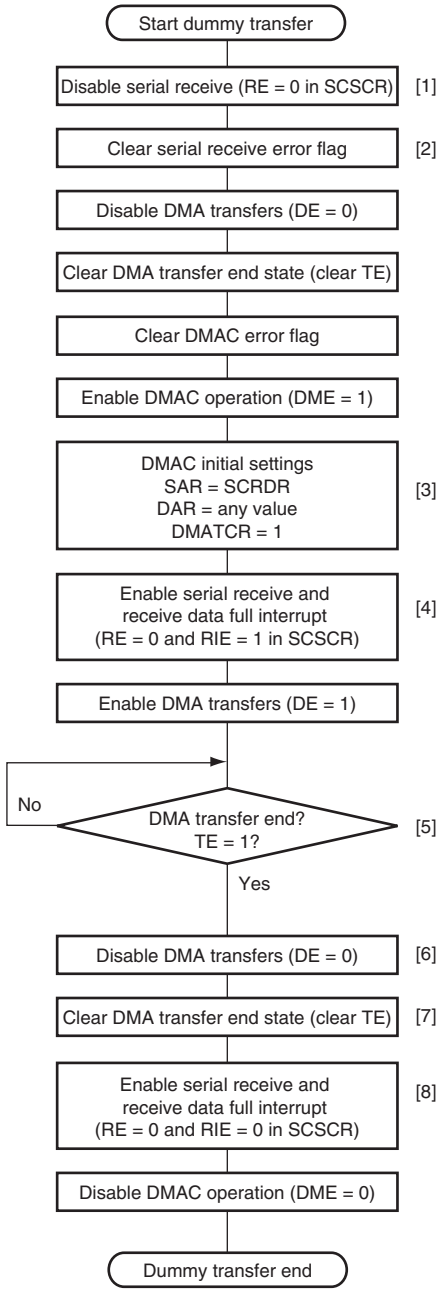
[6] Disable DMA transfers (DE = 0).

[7] Clear the DMA transfer end state (clear TE).

[8] To continue DMA transfer operation, reset the DMAC. No dummy transfer is needed in this case.

[9] Disable the receive data full interrupt (RIE = 0 in SCSCR) to disable interrupt requests.

Figure 10.22 Example DMA Transfer Sequence in Peripheral Module Request Mode (RXI)



[1] [1] Disable serial receive so no interrupt (RXI) occurs during the transfer sequence.

[2] Clear the serial receive error flag. At this point, do not clear the interrupt source flag (RDRF in SCSSR).

[3] Initialize the DMAC. Set the DMA transfer count to 1 (DMATCR = 1).

[4] While serial receive remains disabled, enable the receive data full interrupt (RE = 0 and RIE = 1 in SCSCR).

[5] Wait for the DMA transfer to end. If the DEI interrupt is enabled (IE = 1), a DEI interrupt will be generated when the DMA transfer ends.

[6] Disable DMA transfers (DE = 0).

[7] Clear the DMA transfer end state (clear TE).

[8] Disable the receive data full interrupt (RIE = 0 in SCSCR) to disable interrupt requests.

Figure 10.23 Example Dummy Transfer Sequence (RXI)

10.5.11 Number of Cycles per Access to On-Chip RAM by DMAC

The number of cycles required for read/write access to on-chip RAM from the DMAC is as shown in table 10.9, which differs depending on the frequency ratio of Iϕ (internal clock) to Bϕ (external bus clock).

Table 10.9 Number of Cycles per Access to On-Chip RAM by DMAC

Setting of Iϕ:Bϕ	Read	Write
1:1	3 × Bcyc	3 × Bcyc
1:1/2	2 × Bcyc	1 × Bcyc
1:1/3	2 × Bcyc	1 × Bcyc
1:1/4 or less	1 × Bcyc	1 × Bcyc

Notes: 1. Bcyc is the external bus clock cycle.
2. The number of cycles for access to the on-chip peripheral I/O or an external device are indicated in section 9.5.16, Access to On-Chip Peripheral I/O Registers by CPU, and section 9.5.17, Access to External Memory by CPU. The access cycles are obtained by subtracting the cycles of Iϕ required for L-bus access from the cycles required for access by the CPU.

10.5.12 Note on DMAC Transfer in Burst Mode when Activation Source Is MTU2

The corresponding bit among DMMTU4 to DMMTU0 in the bus function extending register (BSCEHR) must be set when performing DMA transfer in burst mode with the MTU2 specified as the activation source. For details, see section 9.4.8, Bus Function Extending Register (BSCEHR).

10.5.13 Bus Function Extending Register (BSCEHR)

With the bus function extending register (BSCEHR), it is possible to set the function to perform transfer by the DMAC preferentially. For details, see section 9.4.8, Bus Function Extending Register (BSCEHR).

Section 11 Multi-Function Timer Pulse Unit 2 (MTU2)

This LSI has an on-chip multi-function timer pulse unit 2 (MTU2) that comprises six 16-bit timer channels.

11.1 Features

- Maximum 16 pulse input/output lines and three pulse input lines
- Selection of eight counter input clocks for each channel (four clocks for channel 5)
- The following operations can be set for channels 0 to 4:
  - Waveform output at compare match
  - Input capture function
  - Counter clear operation
  - Multiple timer counters (TCNT) can be written to simultaneously
  - Simultaneous clearing by compare match and input capture is possible
  - Register simultaneous input/output is possible by synchronous counter operation
  - A maximum 12-phase PWM output is possible in combination with synchronous operation
- Buffer operation settable for channels 0, 3, and 4
- Phase counting mode settable independently for each of channels 1 and 2
- Cascade connection operation
- Fast access via internal 16-bit bus
- 28 interrupt sources
- Automatic transfer of register data
- A/D converter start trigger can be generated
- Module standby mode can be settable
- A total of six-phase waveform output, which includes complementary PWM output, and positive and negative phases of reset PWM output by interlocking operation of channels 3 and 4, is possible.
- AC synchronous motor (brushless DC motor) drive mode using complementary PWM output and reset PWM output is settable by interlocking operation of channels 0, 3, and 4, and the selection of two types of waveform outputs (chopping and level) is possible.
- Dead time compensation counter available in channel 5
- External pulse width measurement available in channel 5
- In complementary PWM mode, interrupts at the crest and trough of the counter value and A/D converter start triggers can be skipped.

Table 11.1 MTU2 Functions

Item		Channel 0	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5
Count clock		MPϕ/1 MPϕ/4 MPϕ/16 MPϕ/64 TCLKA TCLKB TCLKC TCLKD	MPϕ/1 MPϕ/4 MPϕ/16 MPϕ/64 MPϕ/256 TCLKA TCLKB	MPϕ/1 MPϕ/4 MPϕ/16 MPϕ/64 MPϕ/1024 TCLKA TCLKB TCLKC	MPϕ/1 MPϕ/4 MPϕ/16 MPϕ/64 MPϕ/256 MPϕ/1024 TCLKA TCLKB	MPϕ/1 MPϕ/4 MPϕ/16 MPϕ/64 MPϕ/256 MPϕ/1024 TCLKA TCLKB	MPϕ/1 MPϕ/4 MPϕ/16 MPϕ/64
General registers		TGRA_0 TGRB_0 TGRE_0	TGRA_1 TGRB_1	TGRA_2 TGRB_2	TGRA_3 TGRB_3	TGRA_4 TGRB_4	TGRU_5 TGRV_5 TGRW_5
General registers/ buffer registers		TGRC_0 TGRD_0 TGRF_0	—	—	TGRC_3 TGRD_3	TGRC_4 TGRD_4	—
I/O pins		TIOC0A TIOC0B TIOC0C TIOC0D	TIOC1A TIOC1B	TIOC2A TIOC2B	TIOC3A TIOC3B TIOC3C TIOC3D	TIOC4A TIOC4B TIOC4C TIOC4D	Input pins TIC5U TIC5V TIC5W
Counter clear function		TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture
Compare match output	0 output	√	√	√	√	√	—
	1 output	√	√	√	√	√	—
	Toggle output	√	√	√	√	√	—
Input capture function		√	√	√	√	√	√
Synchronous operation		√	√	√	√	√	—
PWM mode 1		√	√	√	√	√	—
PWM mode 2		√	√	√	—	—	—
Complementary PWM mode		—	—	—	√	√	—
Reset PWM mode		—	—	—	√	√	—
AC synchronous motor drive mode		√	—	—	√	√	—

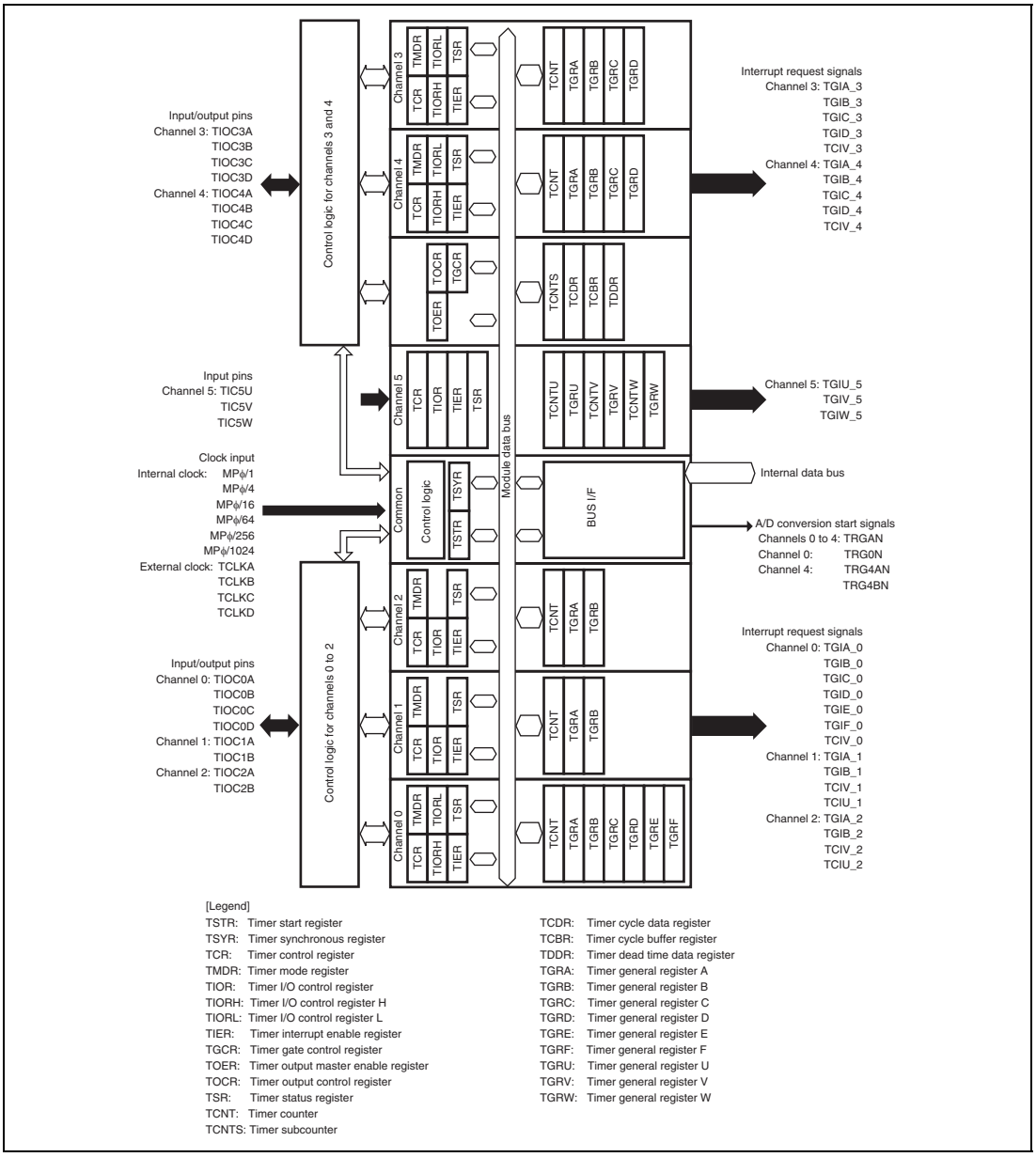
Item	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5
Phase counting mode	—	√	√	—	—	—
Buffer operation	√	—	—	√	√	—
Dead time compensation counter function	—	—	—	—	—	√
External pulse width measurement	—	—	—	—	—	√
DMAC activation	TGRA_0 compare match or input capture	TGRA_1 compare match or input capture	TGRA_2 compare match or input capture	TGRA_3 compare match or input capture	TGRA_4 compare match or input capture	—
DTC activation	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow or underflow	TGR compare match or input capture
A/D converter start trigger	TGRA_0 compare match or input capture TGRE_0 compare match	TGRA_1 compare match or input capture	TGRA_2 compare match or input capture	TGRA_3 compare match or input capture	TGRA_4 compare match or input capture TCNT_4 underflow (trough) in complementary PWM mode	—

Item	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5
Interrupt sources	7 sources	4 sources	4 sources	5 sources	5 sources	3 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	• Compare match or input capture 1A • Compare match or input capture 1B • Overflow • Underflow	• Compare match or input capture 2A • Compare match or input capture 2B • Overflow • Underflow	• 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	• Compare match or input capture 4A • Compare match or input capture 4B • Compare match or input capture 4C • Compare match or input capture 4D • Overflow or underflow	• Compare match or input capture 5U • Compare match or input capture 5V • Compare match or input capture 5W

Item	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5
A/D converter start request delaying function	—	—	—	—	• A/D converter start request at a match between TADCORA_4 and TCNT_4 • A/D converter start request at a match between TADCORB_4 and TCNT_4	—
Interrupt skipping function	—	—	—	• Skips TGRA_3 compare match interrupts	• Skips TCIV_4 interrupts	—

[Legend]
√: Possible
—: Not possible

Figure 11.1 shows a block diagram of the MTU2.



[Legend]

TSTR: Timer start register
TSYR: Timer synchronous register
TCR: Timer control register
TMDR: Timer mode register
TIOR: Timer I/O control register
TIORH: Timer I/O control register H
TIORL: Timer I/O control register L
TIER: Timer interrupt enable register
TGCR: Timer gate control register
TOER: Timer output master enable register
TOCR: Timer output control register
TSR: Timer status register
TCNT: Timer counter
TCNTS: Timer subcounter
TCDR: Timer cycle data register
TCBR: Timer cycle buffer register
TDDR: Timer dead time data register
TGRA: Timer general register A
TGRB: Timer general register B
TGRC: Timer general register C
TGRD: Timer general register D
TGRE: Timer general register E
TGRF: Timer general register F
TGRU: Timer general register U
TGRV: Timer general register V
TGRW: Timer general register W

Figure 11.1 Block Diagram of MTU2

11.2 Input/Output Pins

Table 11.2 Pin Configuration

Channel	Pin Name	I/O	Function
Common	TCLKA	Input	External clock A input pin (Channel 1 phase counting mode A phase input)
	TCLKB	Input	External clock B input pin (Channel 1 phase counting mode B phase input)
	TCLKC	Input	External clock C input pin (Channel 2 phase counting mode A phase input)
	TCLKD	Input	External clock D input pin (Channel 2 phase counting mode B phase input)
0	TIOC0A	I/O	TGRA_0 input capture input/output compare output/PWM output pin
	TIOC0B	I/O	TGRB_0 input capture input/output compare output/PWM output pin
	TIOC0C	I/O	TGRC_0 input capture input/output compare output/PWM output pin
	TIOC0D	I/O	TGRD_0 input capture input/output compare output/PWM output pin
1	TIOC1A	I/O	TGRA_1 input capture input/output compare output/PWM output pin
	TIOC1B	I/O	TGRB_1 input capture input/output compare output/PWM output pin
2	TIOC2A	I/O	TGRA_2 input capture input/output compare output/PWM output pin
	TIOC2B	I/O	TGRB_2 input capture input/output compare output/PWM output pin
3	TIOC3A	I/O	TGRA_3 input capture input/output compare output/PWM output pin
	TIOC3B	I/O	TGRB_3 input capture input/output compare output/PWM output pin
	TIOC3C	I/O	TGRC_3 input capture input/output compare output/PWM output pin
	TIOC3D	I/O	TGRD_3 input capture input/output compare output/PWM output pin
4	TIOC4A	I/O	TGRA_4 input capture input/output compare output/PWM output pin
	TIOC4B	I/O	TGRB_4 input capture input/output compare output/PWM output pin
	TIOC4C	I/O	TGRC_4 input capture input/output compare output/PWM output pin
	TIOC4D	I/O	TGRD_4 input capture input/output compare output/PWM output pin
5	TIC5U	Input	TGRU_5 input capture input/external pulse input pin
	TIC5V	Input	TGRV_5 input capture input/external pulse input pin
	TIC5W	Input	TGRW_5 input capture input/external pulse input pin

11.3 Register Descriptions

The MTU2 has the following registers. For details on register addresses and register states during each process, refer to section 27, List of Registers. To distinguish registers in each channel, an underscore and the channel number are added as a suffix to the register name; TCR for channel 0 is expressed as TCR_0.

Table 11.3 Register Configuration

Register Name	Abbreviation	R/W	Initial value	Address	Access Size
Timer control register_3	TCR_3	R/W	H'00	H'FFFFC200	8, 16, 32
Timer control register_4	TCR_4	R/W	H'00	H'FFFFC201	8
Timer mode register_3	TMDR_3	R/W	H'00	H'FFFFC202	8, 16
Timer mode register_4	TMDR_4	R/W	H'00	H'FFFFC203	8
Timer I/O control register H_3	TIORH_3	R/W	H'00	H'FFFFC204	8, 16, 32
Timer I/O control register L_3	TIORL_3	R/W	H'00	H'FFFFC205	8
Timer I/O control register H_4	TIORH_4	R/W	H'00	H'FFFFC206	8, 16
Timer I/O control register L_4	TIORL_4	R/W	H'00	H'FFFFC207	8
Timer interrupt enable register_3	TIER_3	R/W	H'00	H'FFFFC208	8, 16
Timer interrupt enable register_4	TIER_4	R/W	H'00	H'FFFFC209	8
Timer output master enable register	TOER	R/W	H'C0	H'FFFFC20A	8
Timer gate control register	TGCR	R/W	H'80	H'FFFFC20D	8
Timer output control register 1	TOCR1	R/W	H'00	H'FFFFC20E	8, 16
Timer output control register 2	TOCR2	R/W	H'00	H'FFFFC20F	8
Timer counter_3	TCNT_3	R/W	H'0000	H'FFFFC210	16, 32
Timer counter_4	TCNT_4	R/W	H'0000	H'FFFFC212	16
Timer cycle data register	TCDR	R/W	H'FFFF	H'FFFFC214	16, 32
Timer dead time data register	TDDR	R/W	H'FFFF	H'FFFFC216	16
Timer general register A_3	TGRA_3	R/W	H'FFFF	H'FFFFC218	16, 32
Timer general register B_3	TGRB_3	R/W	H'FFFF	H'FFFFC21A	16
Timer general register A_4	TGRA_4	R/W	H'FFFF	H'FFFFC21C	16, 32
Timer general register B_4	TGRB_4	R/W	H'FFFF	H'FFFFC21E	16

Register Name	Abbreviation	R/W	Initial value	Address	Access Size
Timer subcounter	TCNTS	R	H'0000	H'FFFFC220	16, 32
Timer cycle buffer register	TCBR	R/W	H'FFFF	H'FFFFC222	16
Timer general register C_3	TGRC_3	R/W	H'FFFF	H'FFFFC224	16, 32
Timer general register D_3	TGRD_3	R/W	H'FFFF	H'FFFFC226	16
Timer general register C_4	TGRC_4	R/W	H'FFFF	H'FFFFC228	16, 32
Timer general register D_4	TGRD_4	R/W	H'FFFF	H'FFFFC22A	16
Timer status register_3	TSR_3	R/W	H'C0	H'FFFFC22C	8, 16
Timer status register_4	TSR_4	R/W	H'C0	H'FFFFC22D	8
Timer interrupt skipping set register	TITCR	R/W	H'00	H'FFFFC230	8, 16
Timer interrupt skipping counter	TITCNT	R	H'00	H'FFFFC231	8
Timer buffer transfer set register	TBTER	R/W	H'00	H'FFFFC232	8
Timer dead time enable register	TDER	R/W	H'01	H'FFFFC234	8
Timer output level buffer register	TOLBR	R/W	H'00	H'FFFFC236	8
Timer buffer operation transfer mode register_3	TBTM_3	R/W	H'00	H'FFFFC238	8, 16
Timer buffer operation transfer mode register_4	TBTM_4	R/W	H'00	H'FFFFC239	8
Timer A/D converter start request control register	TADCR	R/W	H'0000	H'FFFFC240	16
Timer A/D converter start request cycle set register A_4	TADCORA_4	R/W	H'FFFF	H'FFFFC244	16, 32
Timer A/D converter start request cycle set register B_4	TADCORB_4	R/W	H'FFFF	H'FFFFC246	16
Timer A/D converter start request cycle set buffer register A_4	TADCOBRA_4	R/W	H'FFFF	H'FFFFC248	16, 32
Timer A/D converter start request cycle set buffer register B_4	TADCOBRB_4	R/W	H'FFFF	H'FFFFC24A	16

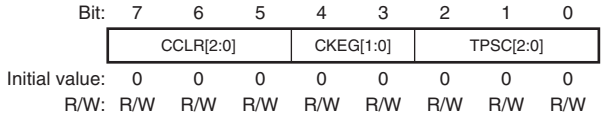
Register Name	Abbreviation	R/W	Initial value	Address	Access Size
Timer waveform control register	TWCR	R/W	H'00	H'FFFFC260	8
Timer start register	TSTR	R/W	H'00	H'FFFFC280	8, 16
Timer synchronous register	TSYR	R/W	H'00	H'FFFFC281	8
Timer counter synchronous start register	TCSYSTR	R/W	H'00	H'FFFFC282	8
Timer read/write enable register	TRWER	R/W	H'01	H'FFFFC284	8
Timer control register_0	TCR_0	R/W	H'00	H'FFFFC300	8, 16, 32
Timer mode register_0	TMDR_0	R/W	H'00	H'FFFFC301	8
Timer I/O control register H_0	TIORH_0	R/W	H'00	H'FFFFC302	8, 16
Timer I/O control register L_0	TIORL_0	R/W	H'00	H'FFFFC303	8
Timer interrupt enable register_0	TIER_0	R/W	H'00	H'FFFFC304	8, 16, 32
Timer status register_0	TSR_0	R/W	H'C0	H'FFFFC305	8
Timer counter_0	TCNT_0	R/W	H'0000	H'FFFFC306	16
Timer general register A_0	TGRA_0	R/W	H'FFFF	H'FFFFC308	16, 32
Timer general register B_0	TGRB_0	R/W	H'FFFF	H'FFFFC30A	16
Timer general register C_0	TGRC_0	R/W	H'FFFF	H'FFFFC30C	16, 32
Timer general register D_0	TGRD_0	R/W	H'FFFF	H'FFFFC30E	16
Timer general register E_0	TGRE_0	R/W	H'FFFF	H'FFFFC320	16, 32
Timer general register F_0	TGRF_0	R/W	H'FFFF	H'FFFFC322	16
Timer interrupt enable register 2_0	TIER2_0	R/W	H'00	H'FFFFC324	8, 16
Timer status register 2_0	TSR2_0	R/W	H'C0	H'FFFFC325	8
Timer buffer operation transfer mode register_0	TBTM_0	R/W	H'00	H'FFFFC326	8
Timer control register_1	TCR_1	R/W	H'00	H'FFFFC380	8, 16
Timer mode register_1	TMDR_1	R/W	H'00	H'FFFFC381	8
Timer I/O control register_1	TIOR_1	R/W	H'00	H'FFFFC382	8
Timer interrupt enable register_1	TIER_1	R/W	H'00	H'FFFFC384	8, 16, 32
Timer status register_1	TSR_1	R/W	H'C0	H'FFFFC385	8

Register Name	Abbreviation	R/W	Initial value	Address	Access Size
Timer counter_1	TCNT_1	R/W	H'0000	H'FFFFC386	16
Timer general register A_1	TGRA_1	R/W	H'FFFF	H'FFFFC388	16, 32
Timer general register B_1	TGRB_1	R/W	H'FFFF	H'FFFFC38A	16
Timer input capture control register	TICCR	R/W	H'00	H'FFFFC390	8
Timer control register_2	TCR_2	R/W	H'00	H'FFFFC400	8, 16
Timer mode register_2	TMDR_2	R/W	H'00	H'FFFFC401	8
Timer I/O control register_2	TIOR_2	R/W	H'00	H'FFFFC402	8
Timer interrupt enable register_2	TIER_2	R/W	H'00	H'FFFFC404	8, 16, 32
Timer status register_2	TSR_2	R/W	H'C0	H'FFFFC405	8
Timer counter_2	TCNT_2	R/W	H'0000	H'FFFFC406	16
Timer general register A_2	TGRA_2	R/W	H'FFFF	H'FFFFC408	16, 32
Timer general register B_2	TGRB_2	R/W	H'FFFF	H'FFFFC40A	16
Timer counter U_5	TCNTU_5	R/W	H'0000	H'FFFFC480	16, 32
Timer general register U_5	TGRU_5	R/W	H'FFFF	H'FFFFC482	16
Timer control register U_5	TCRU_5	R/W	H'00	H'FFFFC484	8
Timer I/O control register U_5	TIORU_5	R/W	H'00	H'FFFFC486	8
Timer counter V_5	TCNTV_5	R/W	H'0000	H'FFFFC490	16, 32
Timer general register V_5	TGRV_5	R/W	H'FFFF	H'FFFFC492	16
Timer control register V_5	TCRV_5	R/W	H'00	H'FFFFC494	8
Timer I/O control register V_5	TIORV_5	R/W	H'00	H'FFFFC496	8
Timer counter W_5	TCNTW_5	R/W	H'0000	H'FFFFC4A0	16, 32
Timer general register W_5	TGRW_5	R/W	H'FFFF	H'FFFFC4A2	16
Timer control register W_5	TCRW_5	R/W	H'00	H'FFFFC4A4	8
Timer I/O control register W_5	TIORW_5	R/W	H'00	H'FFFFC4A6	8
Timer status register_5	TSR_5	R/W	H'00	H'FFFFC4B0	8
Timer interrupt enable register_5	TIER_5	R/W	H'00	H'FFFFC4B2	8
Timer start register_5	TSTR_5	R/W	H'00	H'FFFFC4B4	8
Timer compare match clear register	TCNTCMPCLR	R/W	H'00	H'FFFFC4B6	8

11.3.1 Timer Control Register (TCR)

The TCR registers are 8-bit readable/writable registers that control the TCNT operation for each channel. The MTU2 has a total of eight TCR registers, one each for channels 0 to 4 and three (TCRU_5, TCRV_5, and TCRW_5) for channel 5. TCR register settings should be conducted only when TCNT operation is stopped.

- TCR_0, TCR_1, TCR_2, TCR_3, TCR_4



Bit	Bit Name	Initial Value	R/W	Description
7 to 5	CCLR[2:0]	000	R/W	Counter Clear These bits select the TCNT counter clearing source. See tables 11.4 and 11.5 for details.
4, 3	CKEG[1:0]	00	R/W	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. MPϕ/4 both edges = MPϕ/2 rising edge). If phase counting mode is used on channels 1 and 2, this setting is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the input clock is MPϕ/4 or slower. When MPϕ/1, or the overflow/underflow of another channel is selected for the input clock, although values can be written, counter operation compiles with the initial value. 00: Count at rising edge 01: Count at falling edge 1x: Count at both edges
2 to 0	TPSC[2:0]	000	R/W	Time Prescaler These bits select the TCNT counter clock. The clock source can be selected independently for each channel. See tables 11.6 to 11.9 for details.

[Legend]
x: Don't care

Table 11.4 CCLR[2:0] (Channels 0, 3, and 4)

Channel	Bit 7 CCLR2	Bit 6 CCLR1	Bit 5 CCLR0	Description
0, 3, 4	0	0	0	TCNT clearing disabled
			1	TCNT cleared by TGRA compare match/input capture
		1	0	TCNT cleared by TGRB compare match/input capture
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/ synchronous operation*1
	1	0	0	TCNT clearing disabled
			1	TCNT cleared by TGRC compare match/input capture*2
		1	0	TCNT cleared by TGRD compare match/input capture*2
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/ synchronous operation*1

Notes: 1. Synchronous operation is set by setting the SYNC bit in TSYR to 1.
2. When TGRC or TGRD is used as a buffer register, TCNT is not cleared because the buffer register setting has priority, and compare match/input capture does not occur.

Table 11.5 CCLR[2:0] (Channels 1 and 2)

Channel	Bit 7 Reserved*2	Bit 6 CCLR1	Bit 5 CCLR0	Description
1, 2	0	0	0	TCNT clearing disabled
			1	TCNT cleared by TGRA compare match/input capture
		1	0	TCNT cleared by TGRB compare match/input capture
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/ synchronous operation*1

Notes: 1. Synchronous operation is selected by setting the SYNC bit in TSYR to 1.
2. Bit 7 is reserved in channels 1 and 2. It is always read as 0 and cannot be modified.

Table 11.6 TPSC[2:0] (Channel 0)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
0	0	0	0	Internal clock: counts on MPϕ/1
			1	Internal clock: counts on MPϕ/4
		1	0	Internal clock: counts on MPϕ/16
			1	Internal clock: counts on MPϕ/64
	1	0	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input
		1	0	External clock: counts on TCLKC pin input
			1	External clock: counts on TCLKD pin input

Table 11.7 TPSC[2:0] (Channel 1)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
1	0	0	0	Internal clock: counts on MPϕ/1
			1	Internal clock: counts on MPϕ/4
		1	0	Internal clock: counts on MPϕ/16
			1	Internal clock: counts on MPϕ/64
	1	0	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input
		1	0	Internal clock: counts on MPϕ/256
			1	Counts on TCNT_2 overflow/underflow

Note: This setting is ignored when channel 1 is in phase counting mode.

Table 11.8 TPSC[2:0] (Channel 2)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
2	0	0	0	Internal clock: counts on MPϕ/1
			1	Internal clock: counts on MPϕ/4
		1	0	Internal clock: counts on MPϕ/16
			1	Internal clock: counts on MPϕ/64
	1	0	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input
		1	0	External clock: counts on TCLKC pin input
			1	Internal clock: counts on MPϕ/1024

Note: This setting is ignored when channel 2 is in phase counting mode.

Table 11.9 TPSC[2:0] (Channels 3 and 4)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
3, 4	0	0	0	Internal clock: counts on MPϕ/1
			1	Internal clock: counts on MPϕ/4
		1	0	Internal clock: counts on MPϕ/16
			1	Internal clock: counts on MPϕ/64
	1	0	0	Internal clock: counts on MPϕ/256
			1	Internal clock: counts on MPϕ/1024
		1	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input

- TCRU_5, TCRV_5, TCRW_5

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TPSC[1:0]	
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

Note: The counter clear source for channel 5 is specified in TCNTCMPCLR.
For details, see 11.3.4, Timer Compare Match Clear Register (TCNTCMPCLR).

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	TPSC[1:0]	00	R/W	Time Prescaler These bits select the TCNT counter clock. The clock source can be selected independently for each channel. See table 11.10 for details.

Table 11.10 TPSC[1:0] (Channel 5)

Channel	Bit 1 TPSC1	Bit 0 TPSC0	Description
5	0	0	Internal clock: counts on MPϕ/1
		1	Internal clock: counts on MPϕ/4
	1	0	Internal clock: counts on MPϕ/16
		1	Internal clock: counts on MPϕ/64

11.3.2 Timer Mode Register (TMDR)

The TMDR registers are 8-bit readable/writable registers that are used to set the operating mode of each channel. The MTU2 has five TMDR registers, one each for channels 0 to 4. TMDR register settings should be changed only when TCNT operation is stopped.

Bit:	7	6	5	4	3	2	1	0
	-	BFE	BFB	BFA	MD[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W:	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	—	Reserved This bit is always read as 0. The write value should always be 0.
6	BFE	0	R/W	Buffer Operation E Specifies whether TGRE_0 and TGRF_0 are to operate in the normal way or to be used together for buffer operation. Compare match with TGRF occurs even when TGRF is used as a buffer register. In channels 1 to 4, this bit is reserved. It is always read as 0 and the write value should always be 0. 0: TGRE_0 and TGRF_0 operate normally 1: TGRE_0 and TGRF_0 used together for buffer operation
5	BFB	0	R/W	Buffer Operation B*[1]*[2] Specifies whether TGRB is to operate in the normal way, or TGRB and TGRD are to be used together for buffer operation. When TGRD is used as a buffer register, TGRD input capture/output compare do not take place in modes other than complementary PWM mode, but compare match with TGRD occurs in complementary PWM mode. Since the TGFD flag will be set if a compare match occurs during Tb interval in complementary PWM mode, the TGIED bit in timer interrupt enable register 3/4 (TIER_3/4) should be cleared to 0. In channels 1 and 2, which have no TGRD, bit 5 is reserved. It is always read as 0 and cannot be modified. 0: TGRB and TGRD operate normally 1: TGRB and TGRD used together for buffer operation

Bit	Bit Name	Initial Value	R/W	Description
4	BFA	0	R/W	Buffer Operation A*1*2 Specifies whether TGRA is to operate in the normal way, or TGRA and TGRC are to be used together for buffer operation. When TGRC is used as a buffer register, TGRC input capture/output compare do not take place in modes other than complementary PWM mode, but compare match with TGRC occurs in complementary PWM mode. Since the TGFC flag will be set if a compare match occurs on channel 4 during Tb interval in complementary PWM mode, the TGIEC bit in timer interrupt enable register 4 (TIER_4) should be cleared to 0. In channels 1 and 2, which have no TGRC, bit 4 is reserved. It is always read as 0 and cannot be modified. 0: TGRA and TGRC operate normally 1: TGRA and TGRC used together for buffer operation
3 to 0	MD[3:0]	0000	R/W	Modes 0 to 3 These bits are used to set the timer operating mode. See table 11.11 for details.

Notes: 1. To enable buffer operation in reset-synchronized PWM mode, set the BFB or BFA bit to 1 on channel 3 and clear the BFB or BFA bit to 0 on channel 4. Buffer operation on channel 4 takes place according to the settings for channel 3.

2. To enable buffer operation in complementary PWM mode, set the BFB or BFA bit to 1 on channel 3. The settings of the BFB or BFA bit on channel 4 are ignored. Buffer operation on channel 4 takes place according to the settings for channel 3.

Table 11.11 Setting of Operation Mode by Bits MD[3:0]

Bit 3 MD3	Bit 2 MD2	Bit 1 MD1	Bit 0 MD0	Description
0	0	0	0	Normal operation
			1	Setting prohibited
		1	0	PWM mode 1
			1	PWM mode 2*1
	1	0	0	Phase counting mode 1*2
			1	Phase counting mode 2*2
		1	0	Phase counting mode 3*2
			1	Phase counting mode 4*2
1	0	0	0	Reset synchronous PWM mode*3
			1	Setting prohibited
		1	x	Setting prohibited
	1	0	0	Setting prohibited
			1	Complementary PWM mode 1 (transmit at crest)*3
		1	0	Complementary PWM mode 2 (transmit at trough)*3
			1	Complementary PWM mode 2 (transmit at crest and trough)*3

[Legend]

x: Don't care

Notes: 1. PWM mode 2 can not be set for channels 3 and 4.

2. Phase counting mode can not be set for channels 0, 3, and 4.

3. Reset synchronous PWM mode and complementary PWM mode can only be set for channel 3. When channel 3 is set to reset synchronous PWM mode or complementary PWM mode, the channel 4 settings become ineffective and automatically conform to the channel 3 settings. However, do not set channel 4 to reset synchronous PWM mode or complementary PWM mode. Reset synchronous PWM mode and complementary PWM mode cannot be set for channels 0, 1, and 2.

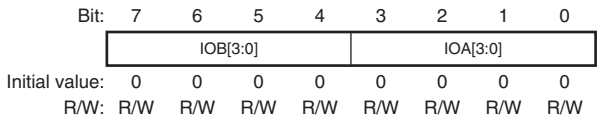
11.3.3 Timer I/O Control Register (TIOR)

The TIOR registers are 8-bit readable/writable registers that control the TGR registers. The MTU2 has a total of eleven TIOR registers, two each for channels 0, 3, and 4, one each for channels 1 and 2, and three (TIORU_5, TIORV_5, and TIORW_5) for channel 5.

TIOR should be set when TMDR is set to select normal operation, PWM mode, or phase counting mode. The initial output specified by TIOR is valid when the counter is stopped (the CST bit in TSTR is cleared to 0). Note also that, in PWM mode 2, the output at the point at which the counter is cleared to 0 is specified.

When TGRC or TGRD is designated for buffer operation, this setting is invalid and the register operates as a buffer register.

- TIORH_0, TIOR_1, TIOR_2, TIORH_3, TIORH_4



Bit	Bit Name	Initial Value	R/W	Description
7 to 4	IOB[3:0]	0000	R/W	I/O Control B0 to B3 Specify the function of TGRB. See the following tables. TIORH_0: Table 11.12 TIOR_1: Table 11.14 TIOR_2: Table 11.15 TIORH_3: Table 11.16 TIORH_4: Table 11.18
3 to 0	IOA[3:0]	0000	R/W	I/O Control A0 to A3 Specify the function of TGRA. See the following tables. TIORH_0: Table 11.20 TIOR_1: Table 11.22 TIOR_2: Table 11.23 TIORH_3: Table 11.24 TIORH_4: Table 11.26

- TIORL_0, TIORL_3, TIORL_4

Bit:	7	6	5	4	3	2	1	0
	IOD[3:0]				IOC[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	IOD[3:0]	0000	R/W	I/O Control D0 to D3 Specify the function of TGRD. See the following tables. TIORL_0: Table 11.13 TIORL_3: Table 11.17 TIORL_4: Table 11.19
3 to 0	IOC[3:0]	0000	R/W	I/O Control C0 to C3 Specify the function of TGRC. See the following tables. TIORL_0: Table 11.21 TIORL_3: Table 11.25 TIORL_4: Table 11.27

- TIORU_5, TIORV_5, TIORW_5

Bit:	7	6	5	4	3	2	1	0
	-	-	-	IOC[4:0]				
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	IOC[4:0]	00000	R/W	I/O Control C0 to C4 Specify the function of TGRU_5, TGRV_5, and TGRW_5. For details, see table 11.28.

Table 11.12 TIORH_0 (Channel 0)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_0 Function**	**TIOC0B Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges
	1	x	x		Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down

[Legend]

x: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 11.13 TIORL_0 (Channel 0)

				Description	
Bit 7 IOD3	**Bit 6 IOD2**	**Bit 5 IOD1**	**Bit 4 IOD0**	**TGRD_0 Function**	**TIOC0D Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges
	1	x	x		Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down

[Legend]

x: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFB bit in TMDR_0 is set to 1 and TGRD_0 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 11.14 TIOR_1 (Channel 1)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_1 Function**	**TIOC1B Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges
	1	x	x		Input capture at generation of TGRC_0 compare match/input capture

[Legend]

x: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 11.15 TIOR_2 (Channel 2)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_2 Function**	**TIOC2B Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges

[Legend]

x: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 11.16 TIORH_3 (Channel 3)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_3 Function**	**TIOC3B Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges

[Legend]

x: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 11.17 TIORL_3 (Channel 3)

				Description	
Bit 7 IOD3	**Bit 6 IOD2**	**Bit 5 IOD1**	**Bit 4 IOD0**	**TGRD_3 Function**	**TIOC3D Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges

[Legend]

x: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFB bit in TMDR_3 is set to 1 and TGRD_3 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 11.18 TIORH_4 (Channel 4)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_4 Function**	**TIOC4B Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges

[Legend]

x: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 11.19 TIORL_4 (Channel 4)

				Description	
Bit 7 IOD3	**Bit 6 IOD2**	**Bit 5 IOD1**	**Bit 4 IOD0**	**TGRD_4 Function**	**TIOC4D Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges

[Legend]

x: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFB bit in TMDR_4 is set to 1 and TGRD_4 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 11.20 TIORH_0 (Channel 0)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_0 Function**	**TIOC0A Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges
	1	x	x		Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down

[Legend]

x: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 11.21 TIORL_0 (Channel 0)

				Description	
Bit 3 IOC3	**Bit 2 IOC2**	**Bit 1 IOC1**	**Bit 0 IOC0**	**TGRC_0 Function**	**TIOC0C Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges
	1	x	x		Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down

[Legend]

x: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFA bit in TMDR_0 is set to 1 and TGRC_0 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 11.22 TIOR_1 (Channel 1)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_1 Function**	**TIOC1A Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges
	1	x	x		Input capture at generation of channel 0/TGRA_0 compare match/input capture

[Legend]

x: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 11.23 TIOR_2 (Channel 2)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_2 Function**	**TIOC2A Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges

[Legend]

x: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 11.24 TIORH_3 (Channel 3)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_3 Function**	**TIOC3A Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges

[Legend]

x: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 11.25 TIORL_3 (Channel 3)

				Description	
Bit 3 IOC3	**Bit 2 IOC2**	**Bit 1 IOC1**	**Bit 0 IOC0**	**TGRC_3 Function**	**TIOC3C Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges

[Legend]

x: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFA bit in TMDR_3 is set to 1 and TGRC_3 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 11.26 TIORH_4 (Channel 4)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_4 Function**	**TIOC4A Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges

[Legend]

x: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 11.27 TIORL_4 (Channel 4)

				Description	
Bit 3 IOC3	**Bit 2 IOC2**	**Bit 1 IOC1**	**Bit 0 IOC0**	**TGRC_4 Function**	**TIOC4C Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	x		Input capture at both edges

[Legend]

x: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFA bit in TMDR_4 is set to 1 and TGRC_4 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 11.28 TIORU_5, TIORV_5, and TIORW_5 (Channel 5)

					Description	
Bit 4 IOC4	Bit 3 IOC3	Bit 2 IOC2	Bit 1 IOC1	Bit 0 IOC0	TGRU_5, TGRV_5, and TGRW_5 Function	TIC5U, TIC5V, and TIC5W Pin Function
0	0	0	0	0	Compare match register	Compare match
				1		Setting prohibited
			1	x		Setting prohibited
		1	x	x		Setting prohibited
	1	x	x	x		Setting prohibited
1	0	0	0	0	Input capture register	Setting prohibited
				1		Input capture at rising edge
			1	0		Input capture at falling edge
				1		Input capture at both edges
		1	x	x		Setting prohibited
	1	0	0	0		Setting prohibited
				1		Measurement of low pulse width of external input signal Capture at trough of complementary PWM mode
			1	0		Measurement of low pulse width of external input signal Capture at crest of complementary PWM mode
				1		Measurement of low pulse width of external input signal Capture at crest and trough of complementary PWM mode
		1	0	0		Setting prohibited
				1		Measurement of high pulse width of external input signal Capture at trough of complementary PWM mode
			1	0		Measurement of high pulse width of external input signal Capture at crest of complementary PWM mode
				1		Measurement of high pulse width of external input signal Capture at crest and trough of complementary PWM mode

[Legend]

x: Don't care

11.3.4 Timer Compare Match Clear Register (TCNTCMPCLR)

TCNTCMPCLR is an 8-bit readable/writable register that specifies requests to clear TCNTU_5, TCNTV_5, and TCNTW_5. The MTU2 has one TCNTCMPCLR in channel 5.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	CMPCLR5U	CMPCLR5V	CMPCLR5W
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	CMPCLR5U	0	R/W	TCNT Compare Clear 5U Enables or disables requests to clear TCNTU_5 at TGRU_5 compare match or input capture. 0: Disables TCNTU_5 to be cleared to H'0000 at TCNTU_5 and TGRU_5 compare match or input capture 1: Enables TCNTU_5 to be cleared to H'0000 at TCNTU_5 and TGRU_5 compare match or input capture
1	CMPCLR5V	0	R/W	TCNT Compare Clear 5V Enables or disables requests to clear TCNTV_5 at TGRV_5 compare match or input capture. 0: Disables TCNTV_5 to be cleared to H'0000 at TCNTV_5 and TGRV_5 compare match or input capture 1: Enables TCNTV_5 to be cleared to H'0000 at TCNTV_5 and TGRV_5 compare match or input capture

Bit	Bit Name	Initial Value	R/W	Description
0	CMPCLR5W	0	R/W	TCNT Compare Clear 5W Enables or disables requests to clear TCNTW_5 at TGRW_5 compare match or input capture. 0: Disables TCNTW_5 to be cleared to H'0000 at TCNTW_5 and TGRW_5 compare match or input capture 1: Enables TCNTW_5 to be cleared to H'0000 at TCNTW_5 and TGRW_5 compare match or input capture

11.3.5 Timer Interrupt Enable Register (TIER)

The TIER registers are 8-bit readable/writable registers that control enabling or disabling of interrupt requests for each channel. The MTU2 has seven TIER registers, two for channel 0 and one each for channels 1 to 5.

- TIER_0, TIER_1, TIER_2, TIER_3, TIER_4

Bit:	7	6	5	4	3	2	1	0
	TTGE	TTGE2	TCIEU	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TTGE	0	R/W	A/D Converter Start Request Enable Enables or disables generation of A/D converter start requests by TGRA input capture/compare match. 0: A/D converter start request generation disabled 1: A/D converter start request generation enabled

Bit	Bit Name	Initial Value	R/W	Description
6	TTGE2	0	R/W	A/D Converter Start Request Enable 2 Enables or disables generation of A/D converter start requests by TCNT_4 underflow (trough) in complementary PWM mode. In channels 0 to 3, bit 6 is reserved. It is always read as 0 and the write value should always be 0. 0: A/D converter start request generation by TCNT_4 underflow (trough) disabled 1: A/D converter start request generation by TCNT_4 underflow (trough) enabled
5	TCIEU	0	R/W	Underflow Interrupt Enable Enables or disables interrupt requests (TCIU) by the TCFU flag when the TCFU flag in TSR is set to 1 in channels 1 and 2. In channels 0, 3, and 4, bit 5 is reserved. It is always read as 0 and the write value should always be 0. 0: Interrupt requests (TCIU) by TCFU disabled 1: Interrupt requests (TCIU) by TCFU enabled
4	TCIEV	0	R/W	Overflow Interrupt Enable Enables or disables interrupt requests (TCIV) by the TCFV flag when the TCFV flag in TSR is set to 1. 0: Interrupt requests (TCIV) by TCFV disabled 1: Interrupt requests (TCIV) by TCFV enabled
3	TGIED	0	R/W	TGR Interrupt Enable D Enables or disables interrupt requests (TGID) by the TGFD bit when the TGFD bit in TSR is set to 1 in channels 0, 3, and 4. In complementary PWM mode, clear the TGIED bit to 0 on channels 3 and 4. In channels 1 and 2, bit 3 is reserved. It is always read as 0 and the write value should always be 0. 0: Interrupt requests (TGID) by TGFD bit disabled 1: Interrupt requests (TGID) by TGFD bit enabled

Bit	Bit Name	Initial Value	R/W	Description
2	TGIEC	0	R/W	TGR Interrupt Enable C Enables or disables interrupt requests (TGIC) by the TGFC bit when the TGFC bit in TSR is set to 1 in channels 0, 3, and 4. In complementary PWM mode, clear the TGIEC bit to 0 on channel 4. In channels 1 and 2, bit 2 is reserved. It is always read as 0 and the write value should always be 0. 0: Interrupt requests (TGIC) by TGFC bit disabled 1: Interrupt requests (TGIC) by TGFC bit enabled
1	TGIEB	0	R/W	TGR Interrupt Enable B Enables or disables interrupt requests (TGIB) by the TGFB bit when the TGFB bit in TSR is set to 1. 0: Interrupt requests (TGIB) by TGFB bit disabled 1: Interrupt requests (TGIB) by TGFB bit enabled
0	TGIEA	0	R/W	TGR Interrupt Enable A Enables or disables interrupt requests (TGIA) by the TGFA bit when the TGFA bit in TSR is set to 1. 0: Interrupt requests (TGIA) by TGFA bit disabled 1: Interrupt requests (TGIA) by TGFA bit enabled

- TIER2_0

Bit:	7	6	5	4	3	2	1	0
	TTGE2	-	-	-	-	-	TGIEF	TGIEE
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TTGE2	0	R/W	A/D Converter Start Request Enable 2 Enables or disables generation of A/D converter start requests by compare match between TCNT_0 and TGRE_0. 0: A/D converter start request generation by compare match between TCNT_0 and TGRE_0 disabled 1: A/D converter start request generation by compare match between TCNT_0 and TGRE_0 enabled
6 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	TGIEF	0	R/W	TGR Interrupt Enable F Enables or disables interrupt requests by compare match between TCNT_0 and TGRF_0. 0: Interrupt requests (TGIF) by TGFE bit disabled 1: Interrupt requests (TGIF) by TGFE bit enabled
0	TGIEE	0	R/W	TGR Interrupt Enable E Enables or disables interrupt requests by compare match between TCNT_0 and TGRE_0. 0: Interrupt requests (TGIE) by TGEE bit disabled 1: Interrupt requests (TGIE) by TGEE bit enabled

- TIER_5

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	TGIE5U	TGIE5V	TGIE5W
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	TGIE5U	0	R/W	TGR Interrupt Enable 5U Enables or disables interrupt requests (TGIU_5) by the CMFU5 bit when the CMFU5 bit in TSR_5 is set to 1. 0: Interrupt requests (TGIU_5) disabled 1: Interrupt requests (TGIU_5) enabled
1	TGIE5V	0	R/W	TGR Interrupt Enable 5V Enables or disables interrupt requests (TGIV_5) by the CMFV5 bit when the CMFV5 bit in TSR_5 is set to 1. 0: Interrupt requests (TGIV_5) disabled 1: Interrupt requests (TGIV_5) enabled
0	TGIE5W	0	R/W	TGR Interrupt Enable 5W Enables or disables interrupt requests (TGIW_5) by the CMFW5 bit when the CMFW5 bit in TSR_5 is set to 1. 0: Interrupt requests (TGIW_5) disabled 1: Interrupt requests (TGIW_5) enabled

11.3.6 Timer Status Register (TSR)

The TSR registers are 8-bit readable/writable registers that indicate the status of each channel. The MTU2 has seven TSR registers, two for channel 0 and one each for channels 1 to 5.

- TSR_0, TSR_1, TSR_2, TSR_3, TSR_4

Bit:	7	6	5	4	3	2	1	0
	TCFD	-	TCFU	TCFV	TGFD	TGFC	TGFB	TGFA
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Description
7	TCFD	1	R	Count Direction Flag Status flag that shows the direction in which TCNT counts in channels 1 to 4. In channel 0, bit 7 is reserved. It is always read as 1 and the write value should always be 1. 0: TCNT counts down 1: TCNT counts up
6	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
5	TCFU	0	R/(W)*[1]	Underflow Flag Status flag that indicates that TCNT underflow has occurred when channels 1 and 2 are set to phase counting mode. Only 0 can be written, for flag clearing. In channels 0, 3, and 4, bit 5 is reserved. It is always read as 0 and the write value should always be 0. [Setting condition] • When the TCNT value underflows (changes from H'0000 to H'FFFF) [Clearing condition] • When 0 is written to TCFU after reading TCFU = 1*[2]

Bit	Bit Name	Initial Value	R/W	Description
4	TCFV	0	R/(W)*1	Overflow Flag Status flag that indicates that TCNT overflow has occurred. Only 0 can be written, for flag clearing. [Setting condition] • When the TCNT value overflows (changes from H'FFFF to H'0000) In channel 4, when the TCNT_4 value underflows (changes from H'0001 to H'0000) in complementary PWM mode, this flag is also set. [Clearing condition] • When 0 is written to TCFV after reading TCFV = 1*2 In cannel 4, when DTC is activated by TCIV interrupt and the DISEL bit of MRB in DTC is 0, this flag is also cleared.
3	TGFD	0	R/(W)*1	Input Capture/Output Compare Flag D*3 Status flag that indicates the occurrence of TGRD input capture or compare match in channels 0, 3, and 4. Only 0 can be written, for flag clearing. In channels 1 and 2, bit 3 is reserved. It is always read as 0 and the write value should always be 0. [Setting conditions] • When TCNT = TGRD and TGRD is functioning as output compare register • When TCNT value is transferred to TGRD by input capture signal and TGRD is functioning as input capture register [Clearing conditions] • When DTC is activated by TGID interrupt and the DISEL bit of MRB in DTC is 0 • When 0 is written to TGFD after reading TGFD = 1*2

Bit	Bit Name	Initial Value	R/W	Description
2	TGFC	0	R/(W)*1	Input Capture/Output Compare Flag C*3 Status flag that indicates the occurrence of TGRC input capture or compare match in channels 0, 3, and 4. Only 0 can be written, for flag clearing. In channels 1 and 2, bit 2 is reserved. It is always read as 0 and the write value should always be 0. [Setting conditions] • When TCNT = TGRC and TGRC is functioning as output compare register • When TCNT value is transferred to TGRC by input capture signal and TGRC is functioning as input capture register [Clearing conditions] • When DTC is activated by TGIC interrupt and the DISEL bit of MRB in DTC is 0 • When 0 is written to TGFC after reading TGFC = 1*2
1	TGFB	0	R/(W)*1	Input Capture/Output Compare Flag B Status flag that indicates the occurrence of TGRB input capture or compare match. Only 0 can be written, for flag clearing. [Setting conditions] • When TCNT = TGRB and TGRB is functioning as output compare register • When TCNT value is transferred to TGRB by input capture signal and TGRB is functioning as input capture register [Clearing conditions] • When DTC is activated by TGIB interrupt and the DISEL bit of MRB in DTC is 0 • When 0 is written to TGFB after reading TGFB = 1*2

Bit	Bit Name	Initial Value	R/W	Description
0	TGFA	0	R/(W)*[1]	Input Capture/Output Compare Flag A Status flag that indicates the occurrence of TGRA input capture or compare match. Only 0 can be written, for flag clearing. [Setting conditions] • When TCNT = TGRA and TGRA is functioning as output compare register • When TCNT value is transferred to TGRA by input capture signal and TGRA is functioning as input capture register [Clearing conditions] • When DMAC is activated by TGIA interrupt • When DTC is activated by TGIA interrupt and the DISEL bit of MRB in DTC is 0 • When 0 is written to TGFA after reading TGFA = 1*[2]

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.
2. If another flag setting condition occurs before writing 0 to the bit after reading it as 1, the flag will not be cleared by writing 0 to it once. In this case, read the bit as 1 again and write 0 to it.
3. TGFC or TGFD may be set to 1 when a compare match occurs during the Tb interval even when the BFB and BFA bits in TMDR have been set to 1 to make TGRC and TGRD operate as buffers in complementary PWM mode.

- TSR2_0

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TGFF	TGFE
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/(W)*	R/(W)*

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	TGFF	0	R/(W)*[1]	Compare Match Flag F Status flag that indicates the occurrence of compare match between TCNT_0 and TGRF_0. [Setting condition] • When TCNT_0 = TGRF_0 and TGRF_0 is functioning as a compare register [Clearing condition] • When 0 is written to TGFF after reading TGFF = 1*[2]
0	TGFE	0	R/(W)*[1]	Compare Match Flag E Status flag that indicates the occurrence of compare match between TCNT_0 and TGRE_0. [Setting condition] • When TCNT_0 = TGRE_0 and TGRE_0 is functioning as compare register [Clearing condition] • When 0 is written to TGFE after reading TGFE = 1*[2]

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.
2. If another flag setting condition occurs before writing 0 to the bit after reading it as 1, the flag will not be cleared by writing 0 to it once. In this case, read the bit as 1 again and write 0 to it.

- TSR_5

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	CMFU5	CMFV5	CMFW5
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	CMFU5	0	R/(W)*[1]	Compare Match/Input Capture Flag U5 Status flag that indicates the occurrence of TGRU_5 input capture or compare match. [Setting conditions] • When TCNTU_5 = TGRU_5 and TGRU_5 is functioning as output compare register • When TCNTU_5 value is transferred to TGRU_5 by input capture signal and TGRU_5 is functioning as input capture register • When TCNTU_5 value is transferred to TGRU_5 and TGRU_5 is functioning as a register for measuring the pulse width of the external input signal. The transfer timing is specified by the IOC bits in timer I/O control register U_5 (TIORU_5).*[2] [Clearing conditions] • When DTC is activated by a TGIU_5 interrupt and the DISEL bit of MRB in DTC is 0 • When 0 is written to CMFU5 after reading CMFU5 = 1

Bit	Bit Name	Initial Value	R/W	Description
1	CMFV5	0	R/(W)*[1]	Compare Match/Input Capture Flag V5 Status flag that indicates the occurrence of TGRV_5 input capture or compare match. [Setting conditions] • When TCNTV_5 = TGRV_5 and TGRV_5 is functioning as output compare register • When TCNTV_5 value is transferred to TGRV_5 by input capture signal and TGRV_5 is functioning as input capture register • When TCNTV_5 value is transferred to TGRV_5 and TGRV_5 is functioning as a register for measuring the pulse width of the external input signal. The transfer timing is specified by the IOC bits in timer I/O control register V_5 (TIORV_5).*[2] [Clearing conditions] • When DTC is activated by a TGIV_5 interrupt and the DISEL bit of MRB in DTC is 0 • When 0 is written to CMFV5 after reading CMFV5 = 1

Bit	Bit Name	Initial Value	R/W	Description
0	CMFW5	0	R/(W)*[1]	Compare Match/Input Capture Flag W5 Status flag that indicates the occurrence of TGRW_5 input capture or compare match. [Setting conditions] • When TCNTW_5 = TGRW_5 and TGRW_5 is functioning as output compare register • When TCNTW_5 value is transferred to TGRW_5 by input capture signal and TGRW_5 is functioning as input capture register • When TCNTW_5 value is transferred to TGRW_5 and TGRW_5 is functioning as a register for measuring the pulse width of the external input signal. The transfer timing is specified by the IOC bits in timer I/O control register W_5 (TIORW_5).*[2] [Clearing conditions] • When DTC is activated by a TGIW_5 interrupt and the DISEL bit of MRB in DTC is 0 • When 0 is written to CMFW5 after reading CMFW5 = 1

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.
2. The transfer timing is specified by the IOC bit in timer I/O control registers U_5/V_5/W_5 (TIORU_5, TIORV_5, TIORW_5).

11.3.7 Timer Buffer Operation Transfer Mode Register (TBTM)

The TBTM registers are 8-bit readable/writable registers that specify the timing for transferring data from the buffer register to the timer general register in PWM mode. The MTU2 has three TBTM registers, one each for channels 0, 3, and 4.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	TTSE	TTSB	TTSA
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	TTSE	0	R/W	Timing Select E Specifies the timing for transferring data from TGRF_0 to TGRE_0 when they are used together for buffer operation. In channels 3 and 4, bit 2 is reserved. It is always read as 0 and the write value should always be 0. When using channel 0 in other than PWM mode, do not set this bit to 1. 0: When compare match E occurs in channel 0 1: When TCNT_0 is cleared
1	TTSB	0	R/W	Timing Select B Specifies the timing for transferring data from TGRD to TGRB in each channel when they are used together for buffer operation. When using a channel in other than PWM mode, do not set this bit to1. 0: When compare match B occurs in each channel 1: When TCNT is cleared in each channel

Bit	Bit Name	Initial Value	R/W	Description
0	TTSA	0	R/W	Timing Select A Specifies the timing for transferring data from TGRC to TGRA in each channel when they are used together for buffer operation. When using a channel in other than PWM mode, do not set this bit to 1. 0: When compare match A occurs in each channel 1: When TCNT is cleared in each channel

11.3.8 Timer Input Capture Control Register (TICCR)

TICCR is an 8-bit readable/writable register that specifies input capture conditions when TCNT_1 and TCNT_2 are cascaded. The MTU2 has one TICCR in channel 1.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	I2BE	I2AE	I1BE	I1AE
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	I2BE	0	R/W	Input Capture Enable Specifies whether to include the TIOC2B pin in the TGRB_1 input capture conditions. 0: Does not include the TIOC2B pin in the TGRB_1 input capture conditions 1: Includes the TIOC2B pin in the TGRB_1 input capture conditions

Bit	Bit Name	Initial Value	R/W	Description
2	I2AE	0	R/W	Input Capture Enable Specifies whether to include the TIOC2A pin in the TGRA_1 input capture conditions. 0: Does not include the TIOC2A pin in the TGRA_1 input capture conditions 1: Includes the TIOC2A pin in the TGRA_1 input capture conditions
1	I1BE	0	R/W	Input Capture Enable Specifies whether to include the TIOC1B pin in the TGRB_2 input capture conditions. 0: Does not include the TIOC1B pin in the TGRB_2 input capture conditions 1: Includes the TIOC1B pin in the TGRB_2 input capture conditions
0	I1AE	0	R/W	Input Capture Enable Specifies whether to include the TIOC1A pin in the TGRA_2 input capture conditions. 0: Does not include the TIOC1A pin in the TGRA_2 input capture conditions 1: Includes the TIOC1A pin in the TGRA_2 input capture conditions

11.3.9 Timer Synchronous Clear Register (TSYCR)

TSYCR is an 8-bit readable/writable register that specifies conditions for clearing TCNT_3S and TCNT_4S in the MTU2S in synchronization with the MTU2. The MTU2S has one TSYCRS in channel 3 but the MTU2 has no TSYCR.

Bit:	7	6	5	4	3	2	1	0
	CE0A	CE0B	CE0C	CE0D	CE1A	CE1B	CE2A	CE2B
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	CE0A	0	R/W	Clear Enable 0A Enables or disables counter clearing when the TGFA flag of TSR_0 in the MTU2 is set. 0: Disables counter clearing by the TGFA flag in TSR_0 1: Enables counter clearing by the TGFA flag in TSR_0
6	CE0B	0	R/W	Clear Enable 0B Enables or disables counter clearing when the TGFB flag of TSR_0 in the MTU2 is set. 0: Disables counter clearing by the TGFB flag in TSR_0 1: Enables counter clearing by the TGFB flag in TSR_0
5	CE0C	0	R/W	Clear Enable 0C Enables or disables counter clearing when the TGFC flag of TSR_0 in the MTU2 is set. 0: Disables counter clearing by the TGFC flag in TSR_0 1: Enables counter clearing by the TGFC flag in TSR_0
4	CE0D	0	R/W	Clear Enable 0D Enables or disables counter clearing when the TGFD flag of TSR_0 in the MTU2 is set. 0: Disables counter clearing by the TGFD flag in TSR_0 1: Enables counter clearing by the TGFD flag in TSR_0

Bit	Bit Name	Initial Value	R/W	Description
3	CE1A	0	R/W	Clear Enable 1A Enables or disables counter clearing when the TGFA flag of TSR_1 in the MTU2 is set. 0: Disables counter clearing by the TGFA flag in TSR_1 1: Enables counter clearing by the TGFA flag in TSR_1
2	CE1B	0	R/W	Clear Enable 1B Enables or disables counter clearing when the TGFB flag of TSR_1 in the MTU2 is set. 0: Disables counter clearing by the TGFB flag in TSR_1 1: Enables counter clearing by the TGFB flag in TSR_1
1	CE2A	0	R/W	Clear Enable 2A Enables or disables counter clearing when the TGFA flag of TSR_2 in the MTU2 is set. 0: Disables counter clearing by the TGFA flag in TSR_2 1: Enables counter clearing by the TGFA flag in TSR_2
0	CE2B	0	R/W	Clear Enable 2B Enables or disables counter clearing when the TGFB flag of TSR_2 in the MTU2 is set. 0: Disables counter clearing by the TGFB flag in TSR_2 1: Enables counter clearing by the TGFB flag in TSR_2

11.3.10 Timer A/D Converter Start Request Control Register (TADCR)

TADCR is a 16-bit readable/writable register that enables or disables A/D converter start requests and specifies whether to link A/D converter start requests with interrupt skipping operation. The MTU2 has one TADCR in channel 4.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BF[1:0]		-	-	-	-	-	-	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
Initial value:	0*	0	0	0	0	0	0	0	0	0*	0	0*	0*	0*	0*	0*
R/W:	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Notes: Accessing TADCR in 8-bit units is prohibited. Always access TADCR in 16-bit units.
* Set to 0 when complementary PWM mode is not selected.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	BF[1:0]	0*[1]0	R/W	TADCOBRA_4/TADCOBRB_4 Transfer Timing Select Select the timing for transferring data from TADCOBRA_4 and TADCOBRB_4 to TADCORA_4 and TADCORB_4. For details, see table 11.29.
13 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	UT4AE	0	R/W	Up-Count TRG4AN Enable Enables or disables A/D converter start requests (TRG4AN) during TCNT_4 up-count operation. 0: A/D converter start requests (TRG4AN) are disabled during TCNT_4 up-count operation 1: A/D converter start requests (TRG4AN) are enabled during TCNT_4 up-count operation
6	DT4AE	0*[1]	R/W	Down-Count TRG4AN Enable Enables or disables A/D converter start requests (TRG4AN) during TCNT_4 down-count operation. 0: A/D converter start requests (TRG4AN) are disabled during TCNT_4 down-count operation 1: A/D converter start requests (TRG4AN) are enabled during TCNT_4 down-count operation

Bit	Bit Name	Initial Value	R/W	Description
5	UT4BE	0	R/W	Up-Count TRG4BN Enable Enables or disables A/D converter start requests (TRG4BN) during TCNT_4 up-count operation. 0: A/D converter start requests (TRG4BN) are disabled during TCNT_4 up-count operation 1: A/D converter start requests (TRG4BN) are enabled during TCNT_4 up-count operation
4	DT4BE	0*[1]	R/W	Down-Count TRG4BN Enable Enables or disables A/D converter start requests (TRG4BN) during TCNT_4 down-count operation. 0: A/D converter start requests (TRG4BN) are disabled during TCNT_4 down-count operation 1: A/D converter start requests (TRG4BN) are enabled during TCNT_4 down-count operation
3	ITA3AE	0*[1]*[2]*[3]	R/W	TGIA_3 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4AN) with TGIA_3 interrupt skipping operation. 0: Does not link with TGIA_3 interrupt skipping operation 1: Links with TGIA_3 interrupt skipping operation
2	ITA4VE	0*[1]*[2]*[3]	R/W	TCIV_4 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4AN) with TCIV_4 interrupt skipping operation. 0: Does not link with TCIV_4 interrupt skipping operation 1: Links with TCIV_4 interrupt skipping operation
1	ITB3AE	0*[1]*[2]*[3]	R/W	TGIA_3 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4BN) with TGIA_3 interrupt skipping operation. 0: Does not link with TGIA_3 interrupt skipping operation 1: Links with TGIA_3 interrupt skipping operation

Bit	Bit Name	Initial Value	R/W	Description
0	ITB4VE	0*1*2*3	R/W	TCIV_4 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4BN) with TCIV_4 interrupt skipping operation. 0: Does not link with TCIV_4 interrupt skipping operation 1: Links with TCIV_4 interrupt skipping operation

Notes: 1. Set to 0 when complementary PWM mode is not selected.
2. Clear this bit to 0 when interrupt skipping is disabled (when the T3AEN or T4VEN bit in the timer interrupt skipping set register (TITCR) is cleared to 0 or when the skipping count set bit (3ACOR or 4VCOR) in TITCR is cleared to 0).
3. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D converter start requests will not be issued.

Table 11.29 Setting of Transfer Timing by BF[1:0] Bits

Bit 15	Bit 14	Description	
BF1	**BF0**	**Complementary PWM Mode**	**Reset-Synchronized PWM Mode**
0	0	Does not transfer data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4).	Does not transfer data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4).
0	1	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) at the crest of TCNT_4.	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) when a compare match occurs between TCNT_3 and TGRA_3.
1	0	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) at the trough of TCNT_4.	Setting prohibited
1	1	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) at the crest and trough of TCNT_4.	Setting prohibited

Bit 15	Bit 14	Description	
BF1	**BF0**	**PWM Mode 1**	**Normal Mode**
0	0	Does not transfer data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4).	Does not transfer data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4).
0	1	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) when a compare match occurs between TCNT_4 and TGRA_4.	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) when a compare match occurs between TCNT_4 and TGRA_4.
1	0	Setting prohibited	Setting prohibited
1	1	Setting prohibited	Setting prohibited

11.3.11 Timer A/D Converter Start Request Cycle Set Registers (TADCORA_4 and TADCORB_4)

TADCORA_4 and TADCORB_4 are 16-bit readable/writable registers. When the TCNT_4 count reaches the value in TADCORA_4 or TADCORB_4, a corresponding A/D converter start request will be issued.

TADCORA_4 and TADCORB_4 are initialized to H'FFFF.

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

Note: TADCORA_4 and TADCORB_4 must not be accessed in eight bits; they should always be accessed in 16 bits.

11.3.12 Timer A/D Converter Start Request Cycle Set Buffer Registers (TADCOBRA_4 and TADCOBRB_4)

TADCOBRA_4 and TADCOBRB_4 are 16-bit readable/writable registers. When the crest or trough of the TCNT_4 count is reached, these register values are transferred to TADCORA_4 and TADCORB_4, respectively.

TADCOBRA_4 and TADCOBRB_4 are initialized to H'FFFF.

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

Note: TADCOBRA_4 and TADCOBRB_4 must not be accessed in eight bits; they should always be accessed in 16 bits.

11.3.13 Timer Counter (TCNT)

The TCNT counters are 16-bit readable/writable counters. The MTU2 has eight TCNT counters, one each for channels 0 to 4 and three (TCNTU_5, TCNTV_5, and TCNTW_5) for channel 5.

The TCNT counters are initialized to H'0000 by a reset.

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

Note: The TCNT counters must not be accessed in eight bits; they should always be accessed in 16 bits.

11.3.14 Timer General Register (TGR)

The TGR registers are 16-bit readable/writable registers. The MTU2 has 21 TGR registers, six for channel 0, two each for channels 1 and 2, four each for channels 3 and 4, and three for channel 5.

TGRA, TGRB, TGRC, and TGRD function as either output compare or input capture registers. TGRC and TGRD for channels 0, 3, and 4 can also be designated for operation as buffer registers. TGR buffer register combinations are TGRA and TGRC, and TGRB and TGRD.

TGRE_0 and TGRF_0 function as compare registers. When the TCNT_0 count matches the TGRE_0 value, an A/D converter start request can be issued. TGRF can also be designated for operation as a buffer register. TGR buffer register combination is TGRE and TGRF.

TGRU_5, TGRV_5, and TGRW_5 function as compare match, input capture, or external pulse width measurement registers.

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

Note: The TGR registers must not be accessed in eight bits; they should always be accessed in 16 bits.
TGR registers are initialized to H'FFFF.

11.3.15 Timer Start Register (TSTR)

TSTR is an 8-bit readable/writable register that selects operation/stoppage of TCNT for channels 0 to 4.

TSTR_5 is an 8-bit readable/writable register that selects operation/stoppage of TCNTU_5, TCNTV_5, and TCNTW_5 for channel 5.

When setting the operating mode in TMDR or setting the count clock in TCR, first stop the TCNT counter.

- TSTR

Bit:	7	6	5	4	3	2	1	0
	CST4	CST3	-	-	-	CST2	CST1	CST0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	CST4	0	R/W	Counter Start 4 and 3
6	CST3	0	R/W	These bits select operation or stoppage for TCNT. If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. If TIOR is written to when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value. 0: TCNT_4 and TCNT_3 count operation is stopped 1: TCNT_4 and TCNT_3 performs count operation
5 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	CST2	0	R/W	Counter Start 2 to 0 These bits select operation or stoppage for TCNT. If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. If TIOR is written to when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value. 0: TCNT_2 to TCNT_0 count operation is stopped 1: TCNT_2 to TCNT_0 performs count operation
1	CST1	0	R/W	
0	CST0	0	R/W	

- TSTR_5

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	CSTU5	CSTV5	CSTW5
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	CSTU5	0	R/W	Counter Start U5 Selects operation or stoppage for TCNTU_5. 0: TCNTU_5 count operation is stopped 1: TCNTU_5 performs count operation
1	CSTV5	0	R/W	Counter Start V5 Selects operation or stoppage for TCNTV_5. 0: TCNTV_5 count operation is stopped 1: TCNTV_5 performs count operation
0	CSTW5	0	R/W	Counter Start W5 Selects operation or stoppage for TCNTW_5. 0: TCNTW_5 count operation is stopped 1: TCNTW_5 performs count operation

11.3.16 Timer Synchronous Register (TSYR)

TSYR is an 8-bit readable/writable register that selects independent operation or synchronous operation for the channel 0 to 4 TCNT counters. A channel performs synchronous operation when the corresponding bit in TSYR is set to 1.

Bit:	7	6	5	4	3	2	1	0
	SYNC4	SYNC3	-	-	-	SYNC2	SYNC1	SYNC0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	SYNC4	0	R/W	Timer Synchronous operation 4 and 3
6	SYNC3	0	R/W	These bits are used to select whether operation is independent of or synchronized with other channels. When synchronous operation is selected, the TCNT synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channel, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits CCLR[2:0] in TCR. 0: TCNT_4 and TCNT_3 operate independently (TCNT presetting/clearing is unrelated to other channels) 1: TCNT_4 and TCNT_3 performs synchronous operation TCNT synchronous presetting/synchronous clearing is possible
5 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	SYNC2	0	R/W	Timer Synchronous operation 2 to 0 These bits are used to select whether operation is independent of or synchronized with other channels. When synchronous operation is selected, the TCNT synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channel, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits CCLR[2:0] in TCR. 0: TCNT_2 to TCNT_0 operates independently (TCNT presetting /clearing is unrelated to other channels) 1: TCNT_2 to TCNT_0 performs synchronous operation TCNT synchronous presetting/synchronous clearing is possible
1	SYNC1	0	R/W	
0	SYNC0	0	R/W	

11.3.17 Timer Counter Synchronous Start Register (TCSYSTR)

TCSYSTR is an 8-bit readable/writable register that specifies synchronous start of the MTU2 and MTU2S counters. Note that the MTU2S does not have TCSYSTR.

Bit:	7	6	5	4	3	2	1	0
	SCH0	SCH1	SCH2	SCH3	SCH4	-	SCH3S	SCH4S
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R/(W)*	R/(W)*

Note: * Only 1 can be written to set the register.

Bit	Bit Name	Initial Value	R/W	Description
7	SCH0	0	R/(W)*	Synchronous Start Controls synchronous start of TCNT_0 in the MTU2. 0: Does not specify synchronous start for TCNT_0 in the MTU2 1: Specifies synchronous start for TCNT_0 in the MTU2 [Clearing condition] • When 1 is set to the CST0 bit of TSTR in MTU2 while SCH0 = 1
6	SCH1	0	R/(W)*	Synchronous Start Controls synchronous start of TCNT_1 in the MTU2. 0: Does not specify synchronous start for TCNT_1 in the MTU2 1: Specifies synchronous start for TCNT_1 in the MTU2 [Clearing condition] • When 1 is set to the CST1 bit of TSTR in MTU2 while SCH1 = 1

Bit	Bit Name	Initial Value	R/W	Description
5	SCH2	0	R/(W)*	Synchronous Start Controls synchronous start of TCNT_2 in the MTU2. 0: Does not specify synchronous start for TCNT_2 in the MTU2 1: Specifies synchronous start for TCNT_2 in the MTU2 [Clearing condition] • When 1 is set to the CST2 bit of TSTR in MTU2 while SCH2 = 1
4	SCH3	0	R/(W)*	Synchronous Start Controls synchronous start of TCNT_3 in the MTU2. 0: Does not specify synchronous start for TCNT_3 in the MTU2 1: Specifies synchronous start for TCNT_3 in the MTU2 [Clearing condition] • When 1 is set to the CST3 bit of TSTR in MTU2 while SCH3 = 1
3	SCH4	0	R/(W)*	Synchronous Start Controls synchronous start of TCNT_4 in the MTU2. 0: Does not specify synchronous start for TCNT_4 in the MTU2 1: Specifies synchronous start for TCNT_4 in the MTU2 [Clearing condition] • When 1 is set to the CST4 bit of TSTR in MTU2 while SCH4 = 1
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	SCH3S	0	R/(W)*	Synchronous Start Controls synchronous start of TCNT_3S in the MTU2S. 0: Does not specify synchronous start for TCNT_3S in the MTU2S 1: Specifies synchronous start for TCNT_3S in the MTU2S [Clearing condition] • When 1 is set to the CST3 bit of TSTRS in MTU2S while SCH3S = 1
0	SCH4S	0	R/(W)*	Synchronous Start Controls synchronous start of TCNT_4S in the MTU2S. 0: Does not specify synchronous start for TCNT_4S in the MTU2S 1: Specifies synchronous start for TCNT_4S in the MTU2S [Clearing condition] • When 1 is set to the CST4 bit of TSTRS in MTU2S while SCH4S = 1

Note: * Only 1 can be written to set the register.

11.3.18 Timer Read/Write Enable Register (TRWER)

TRWER is an 8-bit readable/writable register that enables or disables access to the registers and counters which have write-protection capability against accidental modification in channels 3 and 4.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	RWE
Initial value:	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RWE	1	R/W	Read/Write Enable Enables or disables access to the registers which have write-protection capability against accidental modification. 0: Disables read/write access to the registers 1: Enables read/write access to the registers [Clearing condition] • When 0 is written to the RWE bit after reading RWE = 1

- Registers and counters having write-protection capability against accidental modification
  22 registers: TCR_3, TCR_4, TMDR_3, TMDR_4, TIORH_3, TIORH_4, TIORL_3, TIORL_4, TIER_3, TIER_4, TGRA_3, TGRA_4, TGRB_3, TGRB_4, TOER, TOCR1, TOCR2, TGCR, TCDR, TDDR, TCNT_3, and TCNT4.

11.3.19 Timer Output Master Enable Register (TOER)

TOER is an 8-bit readable/writable register that enables/disables output settings for output pins TIOC4D, TIOC4C, TIOC3D, TIOC4B, TIOC4A, and TIOC3B. These pins do not output correctly if the TOER bits have not been set. Set TOER of CH3 and CH4 prior to setting TIOR of CH3 and CH4. Set TOER when count operation of TCNT channels 3 and 4 is halted (See figures 11.35 and 11.38).

Bit:	7	6	5	4	3	2	1	0
	-	-	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
5	OE4D	0	R/W	Master Enable TIOC4D This bit enables/disables the TIOC4D pin MTU2 output. 0: MTU2 output is disabled (inactive level)* 1: MTU2 output is enabled
4	OE4C	0	R/W	Master Enable TIOC4C This bit enables/disables the TIOC4C pin MTU2 output. 0: MTU2 output is disabled (inactive level)* 1: MTU2 output is enabled
3	OE3D	0	R/W	Master Enable TIOC3D This bit enables/disables the TIOC3D pin MTU2 output. 0: MTU2 output is disabled (inactive level)* 1: MTU2 output is enabled
2	OE4B	0	R/W	Master Enable TIOC4B This bit enables/disables the TIOC4B pin MTU2 output. 0: MTU2 output is disabled (inactive level)* 1: MTU2 output is enabled
1	OE4A	0	R/W	Master Enable TIOC4A This bit enables/disables the TIOC4A pin MTU2 output. 0: MTU2 output is disabled (inactive level)* 1: MTU2 output is enabled

Bit	Bit Name	Initial Value	R/W	Description
0	OE3B	0	R/W	Master Enable TIOC3B This bit enables/disables the TIOC3B pin MTU2 output. 0: MTU2 output is disabled (inactive level)* 1: MTU2 output is enabled

Note: * The inactive level is determined by the settings in timer output control registers 1 and 2 (TOCR1 and TOCR2). For details, refer to section 11.3.20, Timer Output Control Register 1 (TOCR1), and section 11.3.21, Timer Output Control Register 2 (TOCR2). Set these bits to 1 to enable MTU2 output in other than complementary PWM or reset-synchronized PWM mode. When these bits are set to 0, low level is output.

11.3.20 Timer Output Control Register 1 (TOCR1)

TOCR1 is an 8-bit readable/writable register that enables/disables PWM synchronized toggle output in complementary PWM mode/reset synchronized PWM mode, and controls output level inversion of PWM output.

Bit:	7	6	5	4	3	2	1	0
	-	PSYE	-	-	TOCL	TOCS	OLSN	OLSP
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R/(W)*	R/W	R/W	R/W

Note: * This bit can be set to 1 only once after a power-on reset. After 1 is written, 0 cannot be written to the bit.

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PSYE	0	R/W	PWM Synchronous Output Enable This bit selects the enable/disable of toggle output synchronized with the PWM period. 0: Toggle output is disabled 1: Toggle output is enabled
5, 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3	TOCL	0	R/(W)*1	TOC Register Write Protection*2 This bit selects the enable/disable of write access to the TOCS, OLSN, and OLSP bits in TOCR1. 0: Write access to the TOCS, OLSN, and OLSP bits is enabled 1: Write access to the TOCS, OLSN, and OLSP bits is disabled
2	TOCS	0	R/W	TOC Select This bit selects either the TOCR1 or TOCR2 setting to be used for the output level in complementary PWM mode and reset-synchronized PWM mode. 0: TOCR1 setting is selected 1: TOCR2 setting is selected
1	OLSN	0	R/W	Output Level Select N*3*4 This bit selects the reverse phase output level in reset-synchronized PWM mode/complementary PWM mode. See table 11.30.
0	OLSP	0	R/W	Output Level Select P*3*4 This bit selects the positive phase output level in reset-synchronized PWM mode/complementary PWM mode. See table 11.31.

Notes: 1. This bit can be set to 1 only once after a power-on 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 TOCS0 bit to 0 makes this bit setting valid.
4. The inverse-phase output is the exact inverse of the positive-phase output unless dead time is generated. When no dead time is generated, only the OLSP setting is valid

Table 11.30 Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLSN	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The reverse phase waveform initial output value changes to active level after elapse of the dead time after count start.

Table 11.31 Output Level Select Function

Bit 0	Function			
			Compare Match Output	
OLSP	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Figure 11.2 shows an example of complementary PWM mode output (1 phase) when OLSN = 1, OLSP = 1.

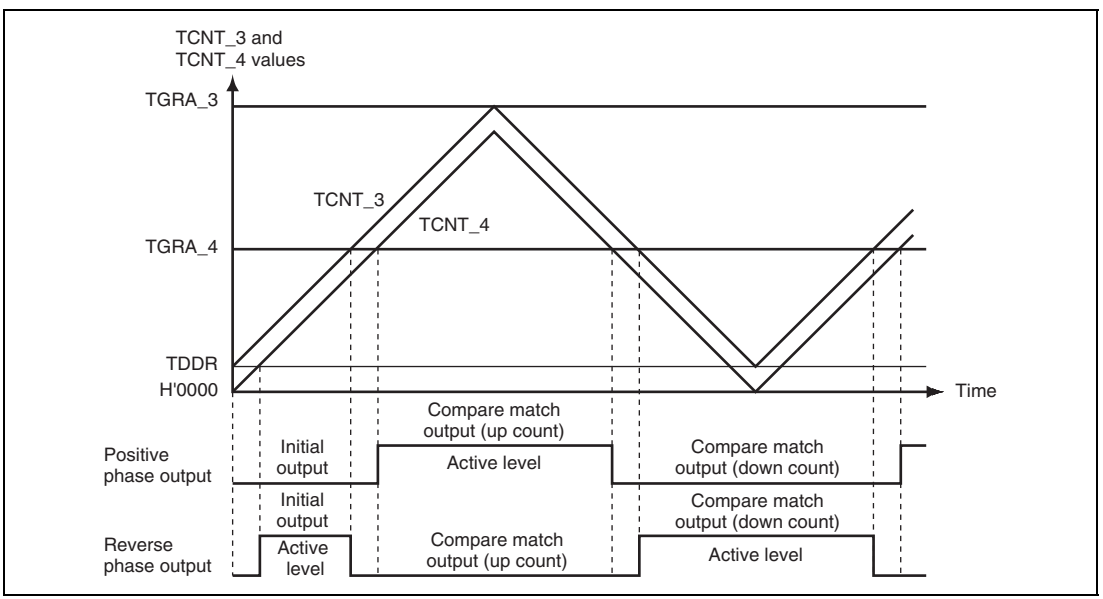


Figure 11.2 Complementary PWM Mode Output Level Example

11.3.21 Timer Output Control Register 2 (TOCR2)

TOCR2 is an 8-bit readable/writable register that controls output level inversion of PWM output in complementary PWM mode and reset-synchronized PWM mode.

Bit:	7	6	5	4	3	2	1	0
	BF[1:0]		OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7, 6	BF[1:0]	00	R/W	TOLBR Buffer Transfer Timing Select These bits select the timing for transferring data from TOLBR to TOCR2. For details, see table 11.32.
5	OLS3N	0	R/W	Output Level Select 3N*1*2 This bit selects the output level on TIOC4D in reset-synchronized PWM mode/complementary PWM mode. See table 11.33.
4	OLS3P	0	R/W	Output Level Select 3P*1*2 This bit selects the output level on TIOC4B in reset-synchronized PWM mode/complementary PWM mode. See table 11.34.
3	OLS2N	0	R/W	Output Level Select 2N*1*2 This bit selects the output level on TIOC4C in reset-synchronized PWM mode/complementary PWM mode. See table 11.35.
2	OLS2P	0	R/W	Output Level Select 2P*1*2 This bit selects the output level on TIOC4A in reset-synchronized PWM mode/complementary PWM mode. See table 11.36.
1	OLS1N	0	R/W	Output Level Select 1N*1*2 This bit selects the output level on TIOC3D in reset-synchronized PWM mode/complementary PWM mode. See table 11.37.

Bit	Bit Name	Initial value	R/W	Description
0	OLS1P	0	R/W	Output Level Select 1P*1*2 This bit selects the output level on TIOC3B in reset-synchronized PWM mode/complementary PWM mode. See table 11.38.

Notes: 1. Setting the TOCS bit in TOCR1 to 1 makes this bit setting valid.
2. The inverse-phase output is the exact inverse of the positive-phase output unless dead time is generated. When no dead time is generated, only the OLSiP setting is valid.

Table 11.32 Setting of Bits BF[1:0]

Bit 7	Bit 6	Description	
BF1	BF0	Complementary PWM Mode	Reset-Synchronized PWM Mode
0	0	Does not transfer data from the buffer register (TOLBR) to TOCR2.	Does not transfer data from the buffer register (TOLBR) to TOCR2.
0	1	Transfers data from the buffer register (TOLBR) to TOCR2 at the crest of the TCNT_4 count.	Transfers data from the buffer register (TOLBR) to TOCR2 when TCNT_3/TCNT_4 is cleared
1	0	Transfers data from the buffer register (TOLBR) to TOCR2 at the trough of the TCNT_4 count.	Setting prohibited
1	1	Transfers data from the buffer register (TOLBR) to TOCR2 at the crest and trough of the TCNT_4 count.	Setting prohibited

Table 11.33 TIOC4D Output Level Select Function

Bit 5	Function			
			Compare Match Output	
OLS3N	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The reverse phase waveform initial output value changes to the active level after elapse of the dead time after count start.

Table 11.34 TIOC4B Output Level Select Function

Bit 4	Function			
			Compare Match Output	
OLS3P	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Table 11.35 TIOC4C Output Level Select Function

Bit 3	Function			
			Compare Match Output	
OLS2N	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The reverse phase waveform initial output value changes to the active level after elapse of the dead time after count start.

Table 11.36 TIOC4A Output Level Select Function

Bit 2	Function			
			Compare Match Output	
OLS2P	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Table 11.37 TIOC3D Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLS1N	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The reverse phase waveform initial output value changes to the active level after elapse of the dead time after count start.

Table 11.38 TIOC4B Output Level Select Function

Bit 0	Function			
			Compare Match Output	
OLS1P	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

11.3.22 Timer Output Level Buffer Register (TOLBR)

TOLBR is an 8-bit readable/writable register that functions as a buffer for TOCR2 and specifies the PWM output level in complementary PWM mode and reset-synchronized PWM mode.

Bit:	7	6	5	4	3	2	1	0
	-	-	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	OLS3N	0	R/W	Specifies the buffer value to be transferred to the OLS3N bit in TOCR2.
4	OLS3P	0	R/W	Specifies the buffer value to be transferred to the OLS3P bit in TOCR2.
3	OLS2N	0	R/W	Specifies the buffer value to be transferred to the OLS2N bit in TOCR2.
2	OLS2P	0	R/W	Specifies the buffer value to be transferred to the OLS2P bit in TOCR2.
1	OLS1N	0	R/W	Specifies the buffer value to be transferred to the OLS1N bit in TOCR2.
0	OLS1P	0	R/W	Specifies the buffer value to be transferred to the OLS1P bit in TOCR2.

Figure 11.3 shows an example of the PWM output level setting procedure in buffer operation.

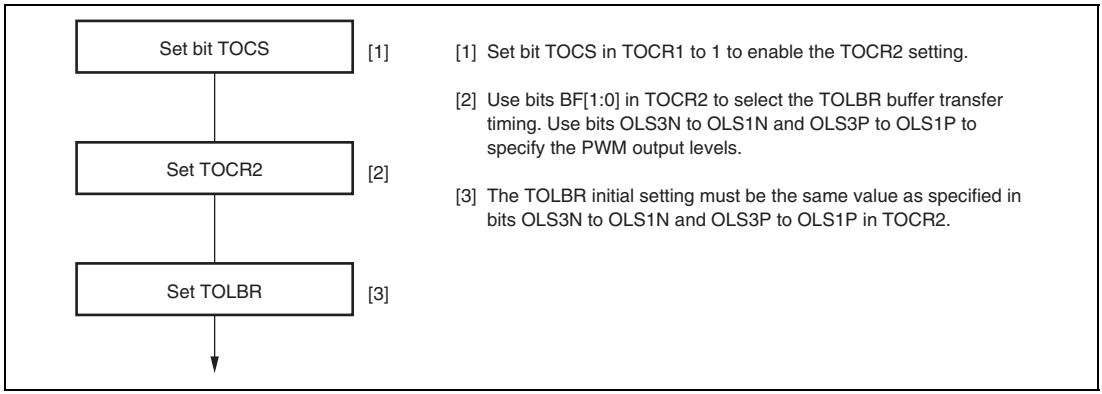


Figure 11.3 PWM Output Level Setting Procedure in Buffer Operation

11.3.23 Timer Gate Control Register (TGCR)

TGCR is an 8-bit readable/writable register that controls the waveform output necessary for brushless DC motor control in reset-synchronized PWM mode/complementary PWM mode. These register settings are ineffective for anything other than complementary PWM mode/reset-synchronized PWM mode.

Bit:	7	6	5	4	3	2	1	0
	-	BDC	N	P	FB*	WF	VF	UF
Initial value:	1	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
6	BDC	0	R/W	Brushless DC Motor This bit selects whether to make the functions of this register (TGCR) effective or ineffective. 0: The function of this register is disabled. 1: The function of this register is enabled.

Bit	Bit Name	Initial value	R/W	Description
5	N	0	R/W	Reverse Phase Output (N) Control This bit selects between level output by the output level select function (table 11.39) and reset-synchronized PWM/complementary PWM output when a reverse pin (TIOC3D, TIOC4C, or TIOC3D) is in the on state. 0: Level output 1: Reset synchronized PWM/complementary PWM output
4	P	0	R/W	Positive Phase Output (P) Control This bit selects between level output by the output level select function (table 11.39) and reset-synchronized PWM/complementary PWM output when a positive pin (TIOC3B, TIOC4A, and TIOC4B) is in the on state. 0: Level output 1: Reset synchronized PWM/complementary PWM output
3	FB*	0	R/W	External Feedback Signal Enable This bit selects whether the switching of the output of the positive/reverse phase is carried out automatically with the TIOC0A, TIOC0B, TIOC0C input signals or the UF, VF, and WF bits in TGCR. 0: Output switching is external input (TIOC0A, TIOC0B, TIOC0C) 1: Output switching is carried out by software (UF, VF, WF settings).
2	WF	0	R/W	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. See table 11.39.
1	VF	0	R/W	
0	UF	0	R/W	

Note: * When the MTU2S is used to set the BDC bit to 1, set the FB bit to 1.

Table 11.39 Output level Select Function

			Function					
Bit 2	**Bit 1**	**Bit 0**	**TIOC3B**	**TIOC4A**	**TIOC4B**	**TIOC3D**	**TIOC4C**	**TIOC4D**
WF (TIOC0C)	**VF (TIOC0B)**	**UF (TIOC0A)**	**U Phase**	**V Phase**	**W Phase**	**U Phase**	**V Phase**	**W Phase**
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
		1	ON	OFF	OFF	OFF	OFF	ON
	1	0	OFF	ON	OFF	ON	OFF	OFF
		1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
		1	ON	OFF	OFF	OFF	ON	OFF
	1	0	OFF	OFF	ON	ON	OFF	OFF
		1	OFF	OFF	OFF	OFF	OFF	OFF

11.3.24 Timer Subcounter (TCNTS)

TCNTS is a 16-bit read-only counter that is used only in complementary PWM mode.

The initial value of TCNTS is H'0000.

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

Note: Accessing the TCNTS in 8-bit units is prohibited. Always access in 16-bit units.

11.3.25 Timer Dead Time Data Register (TDDR)

TDDR is a 16-bit register, used only in complementary PWM mode, that specifies the TCNT_3 and TCNT_4 counter offset values. In complementary PWM mode, when the TCNT_3 and TCNT_4 counters are cleared and then restarted, the TDDR register value is loaded into the TCNT_3 counter and the count operation starts.

The initial value of TDDR is H'FFFF.

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

Note: Accessing the TDDR in 8-bit units is prohibited. Always access in 16-bit units.

11.3.26 Timer Cycle Data Register (TCDR)

TCDR is a 16-bit register used only in complementary PWM mode. Set half the PWM carrier sync value (note that this value should be at least double the value specified in TDDR + 3) as the TCDR register value. This register is constantly compared with the TCNTS counter in complementary PWM mode, and when a match occurs, the TCNTS counter switches direction (decrement to increment).

The initial value of TCDR is H'FFFF.

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

Note: Accessing the TCDR in 8-bit units is prohibited. Always access in 16-bit units.

11.3.27 Timer Cycle Buffer Register (TCBR)

TCBR is a 16-bit register used only in complementary PWM mode. It functions as a buffer register for the TCDR register. The TCBR register values are transferred to the TCDR register with the transfer timing set in the TMDR register.

The initial value of TCBR is H'FFFF.

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

Note: Accessing the TCBR in 8-bit units is prohibited. Always access in 16-bit units.

11.3.28 Timer Interrupt Skipping Set Register (TITCR)

TITCR is an 8-bit readable/writable register that enables or disables interrupt skipping and specifies the interrupt skipping count. The MTU2 has one TITCR.

Bit:	7	6	5	4	3	2	1	0
	T3AEN	3ACOR[2:0]			T4VEN	4VCOR[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7	T3AEN	0	R/W	TGIA_3 Interrupt Skipping Enable Enables or disables TGIA_3 interrupt skipping. 0: TGIA_3 interrupt skipping disabled 1: TGIA_3 interrupt skipping enabled
6 to 4	3ACOR[2:0]	000	R/W	TGIA_3 Interrupt Constant register These bits specify the TGIA_3 interrupt skipping count within the range from 0 to 7.* For details, see table 11.40.
3	T4VEN	0	R/W	TCIV_4 Interrupt Skipping Enable Enables or disables TCIV_4 interrupt skipping. 0: TCIV_4 interrupt skipping disabled 1: TCIV_4 interrupt skipping enabled

Bit	Bit Name	Initial value	R/W	Description
2 to 0	4VCOR[2:0]	000	R/W	TCIV_4 Interrupt Constant register These bits specify the TCIV_4 interrupt skipping count within the range from 0 to 7.* For details, see table 11.41.

Note: * When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed. Before changing the interrupt skipping count, be sure to clear the T3AEN and T4VEN bits to 0 to clear the skipping counter (TITCNT).

Table 11.40 Setting of Interrupt Skipping Count by Bits 3ACOR[2:0]

Bit 6	Bit 5	Bit 4	
3ACOR2	3ACOR1	3ACOR0	Description
0	0	0	Does not skip TGIA_3 interrupts.
0	0	1	Sets the TGIA_3 interrupt skipping count to 1.
0	1	0	Sets the TGIA_3 interrupt skipping count to 2.
0	1	1	Sets the TGIA_3 interrupt skipping count to 3.
1	0	0	Sets the TGIA_3 interrupt skipping count to 4.
1	0	1	Sets the TGIA_3 interrupt skipping count to 5.
1	1	0	Sets the TGIA_3 interrupt skipping count to 6.
1	1	1	Sets the TGIA_3 interrupt skipping count to 7.

Table 11.41 Setting of Interrupt Skipping Count by Bits 4VCOR[2:0]

Bit 2	Bit 1	Bit 0	
4VCOR2	4VCOR1	4VCOR0	Description
0	0	0	Does not skip TCIV_4 interrupts.
0	0	1	Sets the TCIV_4 interrupt skipping count to 1.
0	1	0	Sets the TCIV_4 interrupt skipping count to 2.
0	1	1	Sets the TCIV_4 interrupt skipping count to 3.
1	0	0	Sets the TCIV_4 interrupt skipping count to 4.
1	0	1	Sets the TCIV_4 interrupt skipping count to 5.
1	1	0	Sets the TCIV_4 interrupt skipping count to 6.
1	1	1	Sets the TCIV_4 interrupt skipping count to 7.

11.3.29 Timer Interrupt Skipping Counter (TITCNT)

TITCNT is an 8-bit readable/writable counter. The MTU2 has one TITCNT. TITCNT retains its value even after stopping the count operation of TCNT_3 and TCNT_4.

Bit:	7	6	5	4	3	2	1	0
	-	3ACNT[2:0]			-	4VCNT[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0.
6 to 4	3ACNT[2:0]	000	R	TGIA_3 Interrupt Counter While the T3AEN bit in TITCR is set to 1, the count in these bits is incremented every time a TGIA_3 interrupt occurs. [Clearing conditions] • When the 3ACNT[2:0] value in TITCNT matches the 3ACOR[2:0] value in TITCR • When the T3AEN bit in TITCR is cleared to 0 • When the 3ACOR[2:0] bits in TITCR are cleared to 0
3	—	0	R	Reserved This bit is always read as 0.
2 to 0	4VCNT[2:0]	000	R	TCIV_4 Interrupt Counter While the T4VEN bit in TITCR is set to 1, the count in these bits is incremented every time a TCIV_4 interrupt occurs. [Clearing conditions] • When the 4VCNT[2:0] value in TITCNT matches the 4VCOR[2:0] value in TITCR • When the T4VEN bit in TITCR is cleared to 0 • When the 4VCOR[2:0] bits in TITCR are cleared to 0

Note: To clear the TITCNT, clear the T3AEN and T4VEN bits in TITCR to 0.

11.3.30 Timer Buffer Transfer Set Register (TBTER)

TBTER is an 8-bit readable/writable register that enables or disables transfer from the buffer registers* used in complementary PWM mode to the temporary registers and specifies whether to link the transfer with interrupt skipping operation. The MTU2 has one TBTER.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	BTE[1:0]	
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	BTE[1:0]	00	R/W	Buffer Transfer Suppression and Interrupt Skipping Link Enable These bits enable or disable transfer from the buffer registers* used in complementary PWM mode to the temporary registers and specify whether to link the transfer with interrupt skipping operation. For details, see table 11.42.

Note: * Applicable buffer registers:
TGRC_3, TGRD_3, TGRC_4, TGRD_4, and TCBR

Table 11.42 Setting of Bits BTE[1:0]

Bit 1	Bit 0	
BTE1	**BTE0**	**Description**
0	0	Enables transfer from the buffer registers to the temporary registers*[1] and does not link the transfer with interrupt skipping operation.
0	1	Disables transfer from the buffer registers to the temporary registers.*[1]
1	0	Links transfer from the buffer registers to the temporary registers with interrupt skipping operation.*[1]*[2]
1	1	Setting prohibited

Notes: 1. Transfers from the temporary registers to the compare registers take place in accordance with the setting of the MD[3:0] bit field in TMDR, regardless of the setting of the BTE[1:0] bit field. For details, refer to section 11.4.8, Complementary PWM Mode.

2. When interrupt skipping is disabled (the T3AEN and T4VEN bits are cleared to 0 in the timer interrupt skipping set register (TITCR) or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0)), be sure to disable link of buffer transfer with interrupt skipping (clear the BTE1 bit in the timer buffer transfer set register (TBTER) to 0). If link with interrupt skipping is enabled while interrupt skipping is disabled, buffer transfer will not be performed.

11.3.31 Timer Dead Time Enable Register (TDER)

TDER is an 8-bit readable/writable register that controls dead time generation in complementary PWM mode. The MTU2 has one TDER in channel 3. TDER must be modified only while TCNT stops.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TDER
Initial value:	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/(W)

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TDER	1	R/(W)	Dead Time Enable Specifies whether to generate dead time. 0: Does not generate dead time 1: Generates dead time* [Clearing condition] • When 0 is written to TDER after reading TDER = 1

Note: * TDDR must be set to 1 or a larger value.

11.3.32 Timer Waveform Control Register (TWCR)

TWCR is an 8-bit readable/writable register. It controls the output waveform when synchronous counter clearing of TCNT_3 and TCNT_4 occurs in complementary PWM mode, specifies the MTU2-MTU2S counter synchronous clearing setting, and specifies whether or not counter clearing occurs at TGRA_3 compare match. The CCE bit and WRE bit in TWCR must be modified only while TCNT stops.

Bit:	7	6	5	4	3	2	1	0
	CCE	-	-	-	-	-	SCC	WRE
Initial value:	0*	0	0	0	0	0	0	0
R/W:	R/(W)	R	R	R	R	R	R/(W)	R/(W)

Note: * Set to 0 when complementary PWM mode is not selected.

Bit	Bit Name	Initial Value	R/W	Description
7	CCE	0*	R/(W)	Compare Match Clear Enable Specifies whether to clear counters at TGRA_3 compare match in complementary PWM mode. 0: Does not clear counters at TGRA_3 compare match 1: Clears counters at TGRA_3 compare match [Setting condition] • When 1 is written to CCE after reading CCE = 0
6 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	SCC	0	R/(W)	Synchronous Clearing Control (only in MTU2S) Specifies whether to clear TCNT_3S and TCNT_4S in the MTU2S when synchronous counter clearing between the MTU2 and MTU2S occurs in complementary PWM mode. When using this control, place the MTU2S in complementary PWM mode. When modifying the SCC bit while the counters are operating, do not modify the CCE or WRE bits. Counter clearing synchronized with the MTU2 is disabled by the SCC bit setting only when synchronous clearing occurs outside the Tb interval at the trough. When synchronous clearing occurs in the Tb interval at the trough including the period immediately after TCNT_3S and TCNT_4S start operation, TCNT_3S and TCNT_4S in the MTU2S are cleared. For the Tb interval at the trough in complementary PWM mode, see figure 11.40. In the MTU2, this bit is reserved. It is always read as 0 and the write value should always be 0. 0: Enables clearing of TCNT_3S and TCNT_4S in the MTU2S by MTU2–MTU2S synchronous clearing operation 1: Disables clearing of TCNT_3S and TCNT_4S in the MTU2S by MTU2–MTU2S synchronous clearing operation [Setting condition] • When 1 is written to SCC after reading SCC = 0

Bit	Bit Name	Initial Value	R/W	Description
0	WRE	0	R/(W)	Initial Output Suppression Enable Selects the waveform output when synchronous counter clearing occurs in complementary PWM mode. The initial output is suppressed only when synchronous clearing occurs within the Tb interval at the trough in complementary PWM mode. When synchronous clearing occurs outside this interval, the initial value specified in TOCR is output regardless of the WRE bit setting. The initial value is also output when synchronous clearing occurs in the Tb interval at the trough immediately after TCNT_3 and TCNT_4 start operation. For the Tb interval at the trough in complementary PWM mode, see figure 11.40. 0: Outputs the initial value specified in TOCR 1: Suppresses initial output [Setting condition] • When 1 is written to WRE after reading WRE = 0

Note: * Set to 0 when complementary PWM mode is not selected.

11.3.33 Bus Master Interface

The timer counters (TCNT), general registers (TGR), timer subcounter (TCNTS), timer cycle buffer register (TCBR), timer dead time data register (TDDR), timer cycle data register (TCDR), timer A/D converter start request control register (TADCR), timer A/D converter start request cycle set registers (TADCOR), and timer A/D converter start request cycle set buffer registers (TADCOBR) are 16-bit registers. A 16-bit data bus to the bus master enables 16-bit read/writes. 8-bit read/write is not possible. Always access in 16-bit units.

All registers other than the above registers are 8-bit registers. These are connected to the CPU by a 16-bit data bus, so 16-bit read/writes and 8-bit read/writes are both possible.

11.4 Operation

11.4.1 Basic Functions

Each channel has a TCNT and TGR register. TCNT performs up-counting, and is also capable of free-running operation, cycle counting, and external event counting.

Each TGR can be used as an input capture register or output compare register.

Always select MTU2 external pins set function using the pin function controller (PFC).

Counter Operation:

When one of bits CST0 to CST4 in TSTR or bits CSTU5, CSTV5, and CSTW5 in TSTR_5 is set to 1, the TCNT counter for the corresponding channel begins counting. TCNT can operate as a free-running counter, periodic counter, for example.

1. Example of Count Operation Setting Procedure
   Figure 11.4 shows an example of the count operation setting procedure.

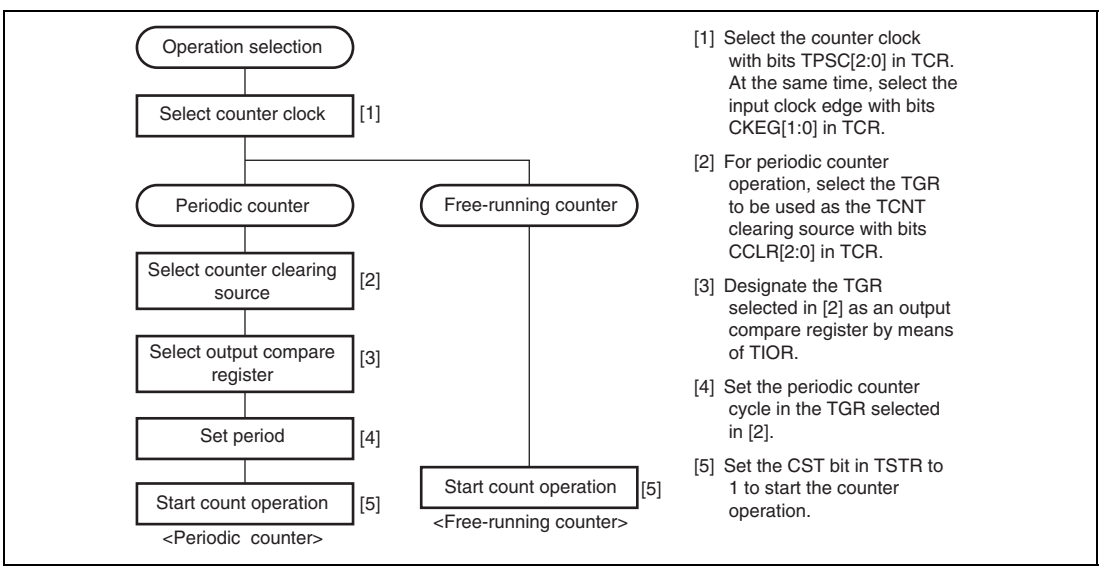


Figure 11.4 Example of Counter Operation Setting Procedure

2. Free-Running Count Operation and Periodic Count Operation:

   Immediately after a reset, the MTU2's TCNT counters are all designated as free-running counters. When the relevant bit in TSTR is set to 1 the corresponding TCNT counter starts up-count operation as a free-running counter. When TCNT overflows (from H'FFFF to H'0000), the TCFV bit in TSR is set to 1. If the value of the corresponding TCIEV bit in TIER is 1 at this point, the MTU2 requests an interrupt. After overflow, TCNT starts counting up again from H'0000.

   Figure 11.5 illustrates free-running counter operation.

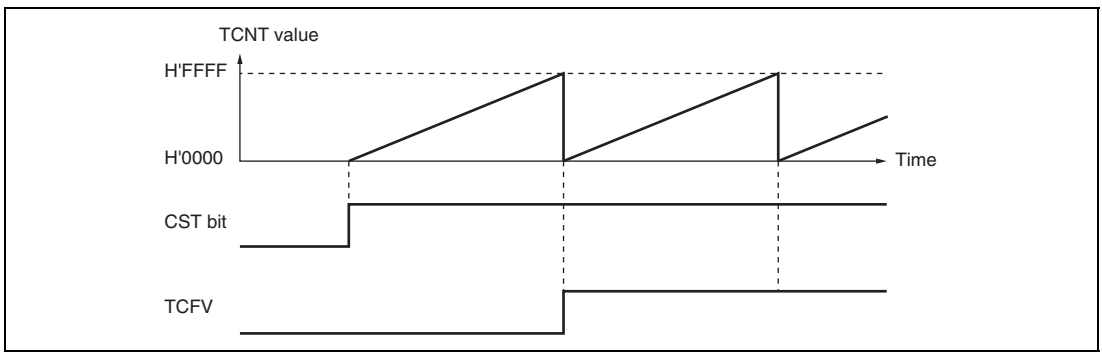


Figure 11.5 Free-Running Counter Operation

When compare match is selected as the TCNT clearing source, the TCNT counter for the relevant channel performs periodic count operation. The TGR register for setting the period is designated as an output compare register, and counter clearing by compare match is selected by means of bits CCLR[2:0] in TCR. After the settings have been made, TCNT starts up-count operation as a periodic counter when the corresponding bit in TSTR is set to 1. When the count value matches the value in TGR, the TGF bit in TSR is set to 1 and TCNT is cleared to H'0000.

If the value of the corresponding TGIE bit in TIER is 1 at this point, the MTU2 requests an interrupt. After a compare match, TCNT starts counting up again from H'0000.

Figure 11.6 illustrates periodic counter operation.

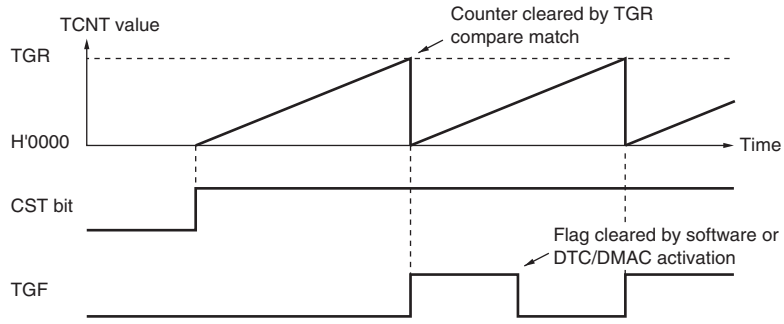
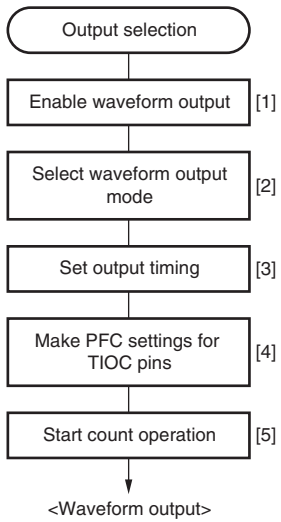


Figure 11.6 Periodic Counter Operation

Waveform Output by Compare Match:

The MTU2 can perform 0, 1, or toggle output from the corresponding output pin using compare match.

1. Example of Setting Procedure for Waveform Output by Compare Match
   Figure 11.7 shows an example of the setting procedure for waveform output by compare match



[1] Enable waveform output on the TIOC pins of channels 3 and 4 in TOER. Make settings to TOER before TIOR.

[2] Select initial value 0 output or 1 output, and compare match output value 0 output, 1 output, or toggle output, by means of TIOR. The set initial value is output at the TIOC pin until the first compare match occurs.

[3] Set the timing for compare match generation in TGR.

[4] Make settings to the PFC (port control register and port I/O register) for the TIOC pins to be used for waveform output. Set the corresponding IOR bit in the port I/O register to 1 to set each TIOC pin to output.

[5] Set the CST bit in TSTR to 1 to start the count operation.

Figure 11.7 Example of Setting Procedure for Waveform Output by Compare Match

2. Examples of Waveform Output Operation:

Figure 11.8 shows an example of 0 output/1 output.

In this example TCNT has been designated as a free-running counter, and settings have been made such that 1 is output by compare match A, and 0 is output by compare match B. When the set level and the pin level coincide, the pin level does not change.

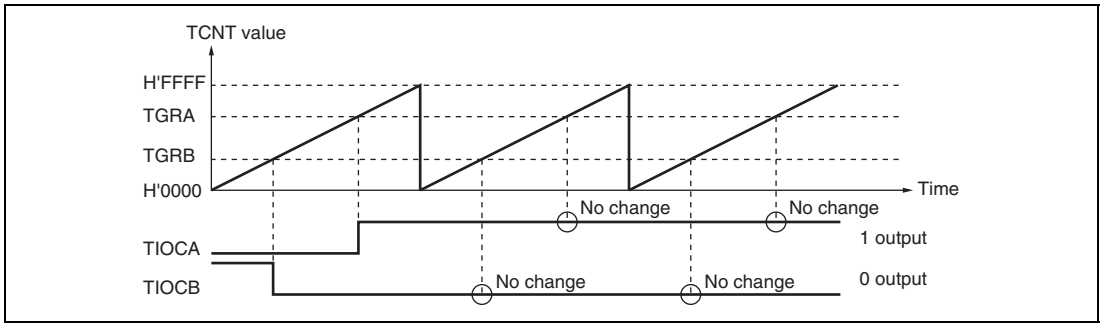


Figure 11.8 Example of 0 Output/1 Output Operation

Figure 11.9 shows an example of toggle output.

In this example, TCNT has been designated as a periodic counter (with counter clearing on compare match B), and settings have been made such that the output is toggled by both compare match A and compare match B.

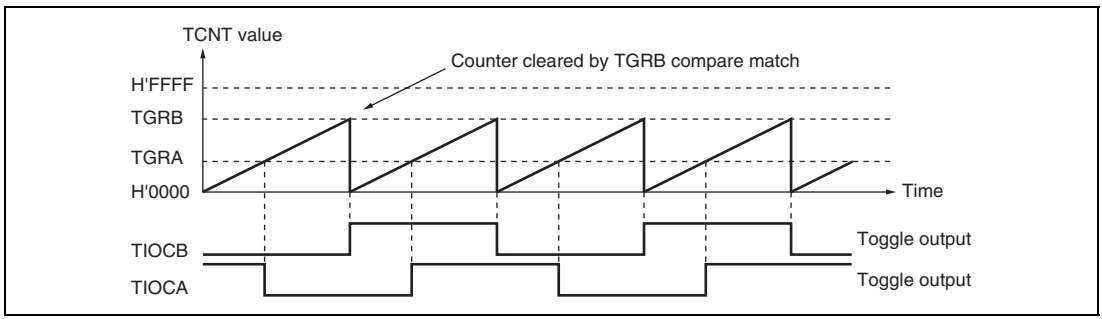


Figure 11.9 Example of Toggle Output Operation

Input Capture Function:

The TCNT value can be transferred to TGR on detection of the TIOC pin input edge.

Rising edge, falling edge, or both edges can be selected as the detected edge. For channels 0 and 1, it is also possible to specify another channel's counter input clock or compare match signal as the input capture source.

Note: When another channel's counter input clock is used as the input capture input for channels 0 and 1, MPϕ/1 should not be selected as the counter input clock used for input capture input. Input capture will not be generated if MPϕ/1 is selected.

1. Example of Input Capture Operation Setting Procedure
   Figure 11.10 shows an example of the input capture operation setting procedure.

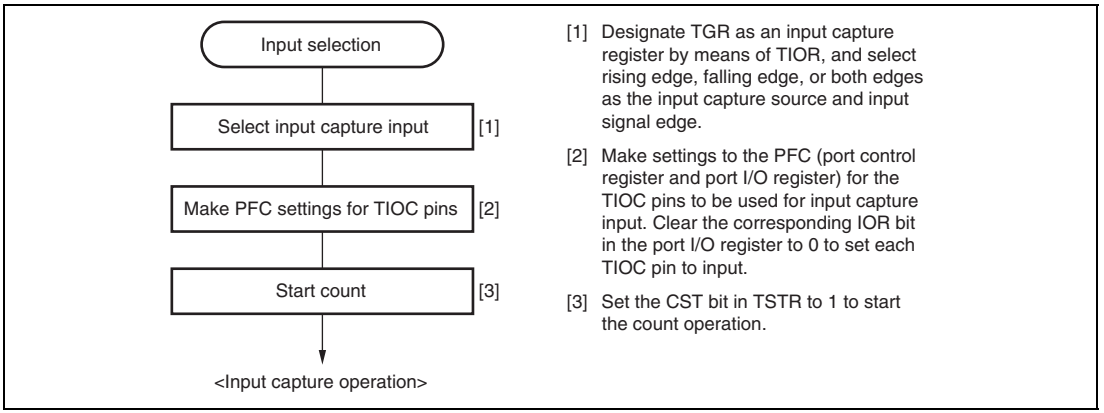


Figure 11.10 Example of Input Capture Operation Setting Procedure

2. Example of Input Capture Operation:

   Figure 11.11 shows an example of input capture operation.

   In this example both rising and falling edges have been selected as the TIOCA pin input capture input edge, the falling edge has been selected as the TIOCB pin input capture input edge, and counter clearing by TGRB input capture has been designated for TCNT.

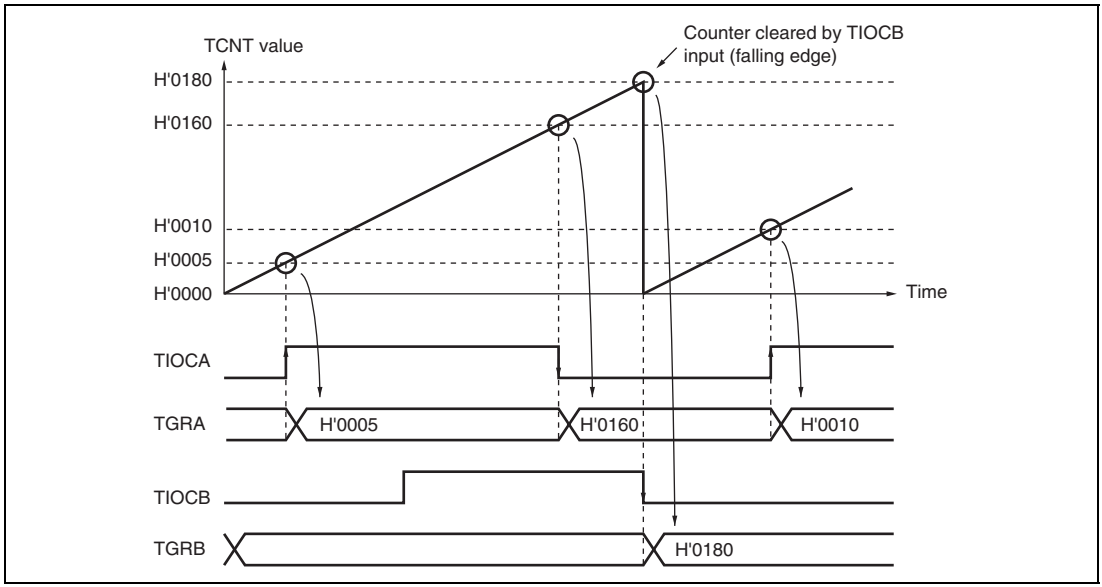


Figure 11.11 Example of Input Capture Operation

11.4.2 Synchronous Operation

In synchronous operation, the values in a number of TCNT counters can be rewritten simultaneously (synchronous presetting). Also, a number of TCNT counters can be cleared simultaneously by making the appropriate setting in TCR (synchronous clearing).

Synchronous operation enables TGR to be incremented with respect to a single time base.

Channels 0 to 4 can all be designated for synchronous operation. Channel 5 cannot be used for synchronous operation.

Example of Synchronous Operation Setting Procedure:

Figure 11.12 shows an example of the synchronous operation setting procedure.

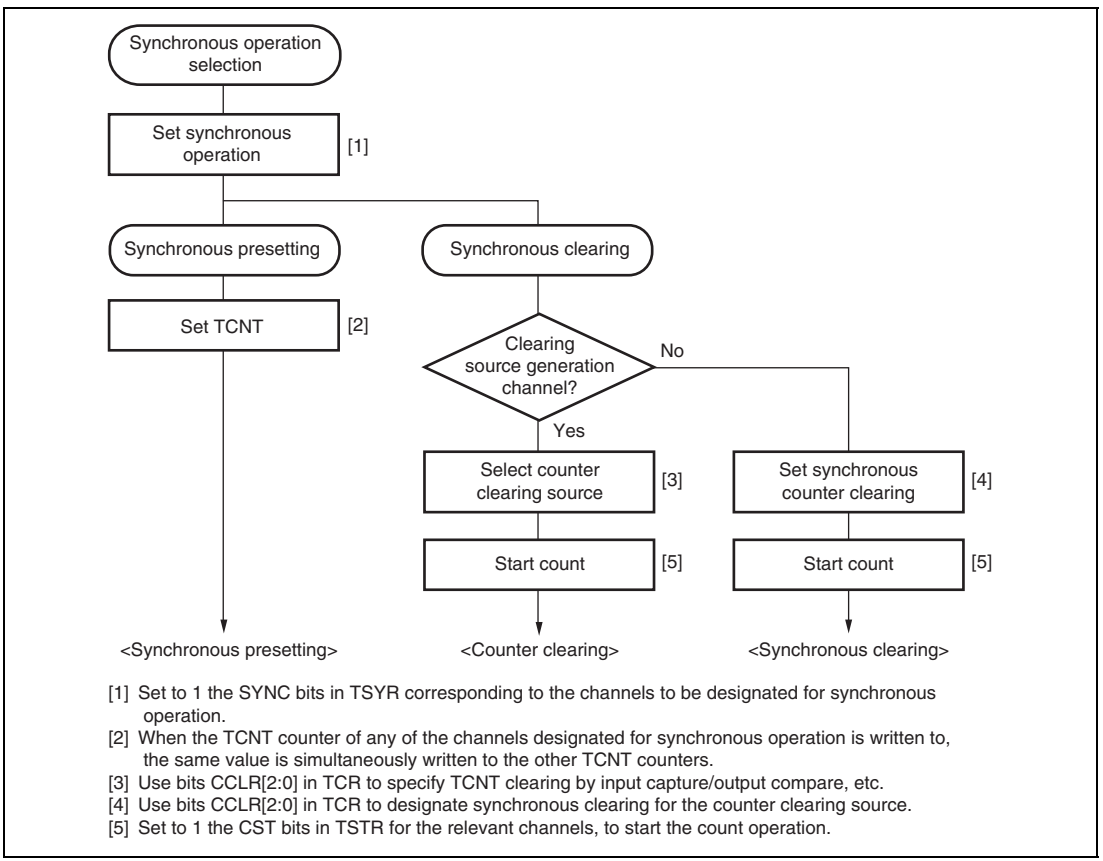


Figure 11.12 Example of Synchronous Operation Setting Procedure

Example of Synchronous Operation: Figure 11.13 shows an example of synchronous operation.

In this example, synchronous operation and PWM mode 1 have been designated for channels 0 to 2, TGRB_0 compare match has been set as the channel 0 counter clearing source, and synchronous clearing has been set for the channel 1 and 2 counter clearing source.

Three-phase PWM waveforms are output from pins TIOC0A, TIOC1A, and TIOC2A. At this time, synchronous presetting, and synchronous clearing by TGRB_0 compare match, are performed for channel 0 to 2 TCNT counters, and the data set in TGRB_0 is used as the PWM cycle.

For details of PWM modes, see section 11.4.5, PWM Modes.

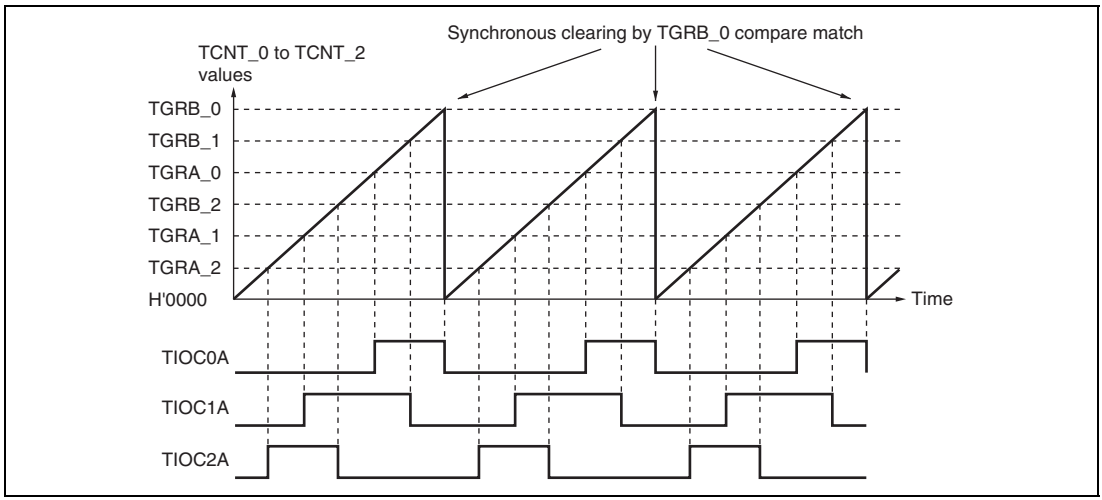


Figure 11.13 Example of Synchronous Operation

11.4.3 Buffer Operation

Buffer operation, provided for channels 0, 3, and 4, enables TGRC and TGRD to be used as buffer registers. In channel 0, TGRF can also be used as a buffer register.

Buffer operation differs depending on whether TGR has been designated as an input capture register or as a compare match register.

Note: TGRE_0 cannot be designated as an input capture register and can only operate as a compare match register.

Table 11.43 shows the register combinations used in buffer operation.

Table 11.43 Register Combinations in Buffer Operation

Channel	Timer General Register	Buffer Register
0	TGRA_0	TGRC_0
	TGRB_0	TGRD_0
	TGRE_0	TGRF_0
3	TGRA_3	TGRC_3
	TGRB_3	TGRD_3
4	TGRA_4	TGRC_4
	TGRB_4	TGRD_4

- When TGR is an output compare register

  When a compare match occurs, the value in the buffer register for the corresponding channel is transferred to the timer general register.

  This operation is illustrated in figure 11.14.

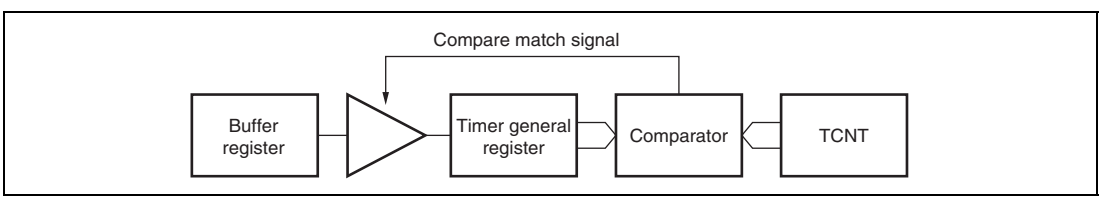


Figure 11.14 Compare Match Buffer Operation

- When TGR is an input capture register

  When input capture occurs, the value in TCNT is transferred to TGR and the value previously held in the timer general register is transferred to the buffer register.

  This operation is illustrated in figure 11.15.

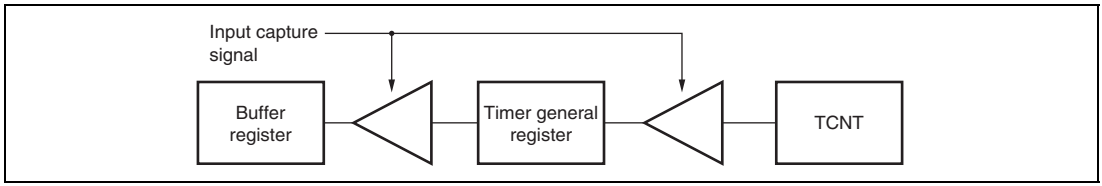


Figure 11.15 Input Capture Buffer Operation

Example of Buffer Operation Setting Procedure: Figure 11.16 shows an example of the buffer operation setting procedure.

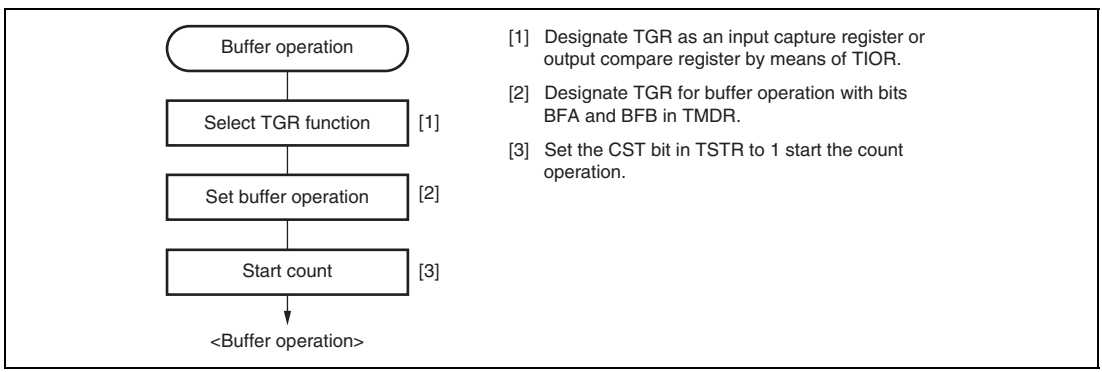


Figure 11.16 Example of Buffer Operation Setting Procedure

Examples of Buffer Operation:

1. When TGR is an output compare register

   Figure 11.17 shows an operation example in which PWM mode 1 has been designated for channel 0, and buffer operation has been designated for TGRA and TGRC. The settings used in this example are TCNT clearing by compare match B, 1 output at compare match A, and 0 output at compare match B. In this example, the TTSA bit in TBTM is cleared to 0.

   As buffer operation has been set, when compare match A occurs the output changes and the value in buffer register TGRC is simultaneously transferred to timer general register TGRA. This operation is repeated each time that compare match A occurs.

   For details of PWM modes, see section 11.4.5, PWM Modes.

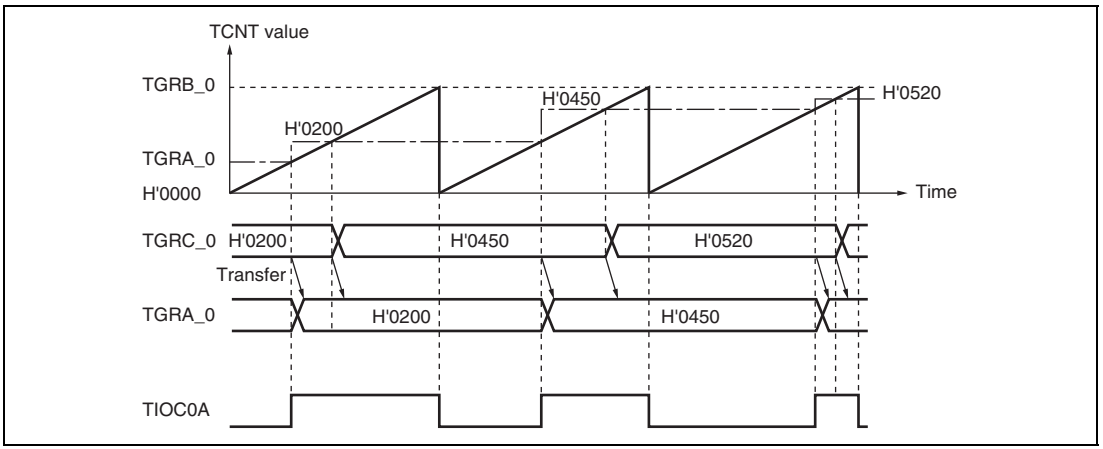


Figure 11.17 Example of Buffer Operation (1)

2. When TGR is an input capture register

   Figure 11.18 shows an operation example in which TGRA has been designated as an input capture register, and buffer operation has been designated for TGRA and TGRC.

   Counter clearing by TGRA input capture has been set for TCNT, and both rising and falling edges have been selected as the TIOCA pin input capture input edge.

   As buffer operation has been set, when the TCNT value is stored in TGRA upon the occurrence of input capture A, the value previously stored in TGRA is simultaneously transferred to TGRC.

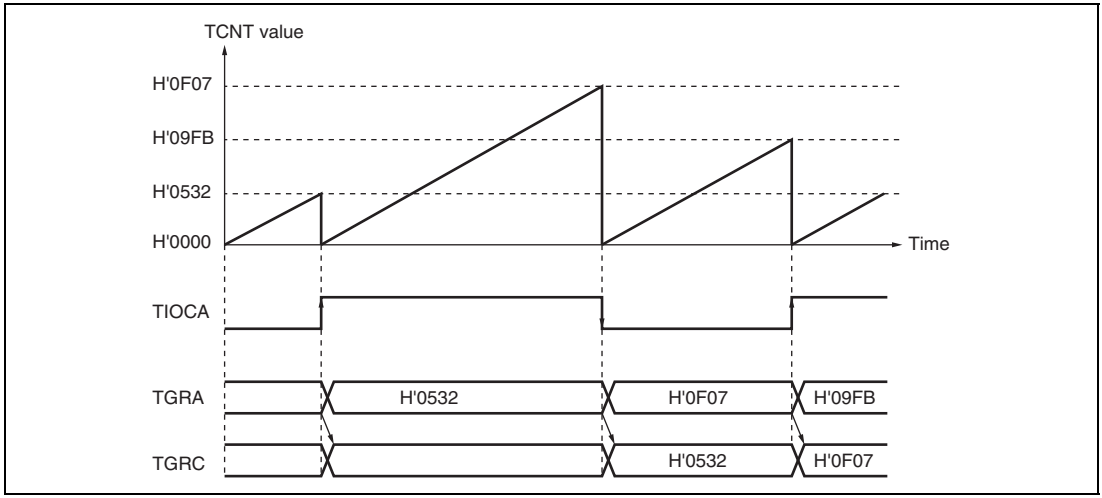


Figure 11.18 Example of Buffer Operation (2)

Selecting Timing for Transfer from Buffer Registers to Timer General Registers in Buffer Operation: The timing for transfer from buffer registers to timer general registers can be selected in PWM mode 1 or 2 for channel 0 or in PWM mode 1 for channels 3 and 4 by setting the buffer operation transfer mode registers (TBTM_0, TBTM_3, and TBTM_4). Either compare match (initial setting) or TCNT clearing can be selected for the transfer timing. TCNT clearing as transfer timing is one of the following cases.

- When TCNT overflows (H'FFFF to H'0000)
- When H'0000 is written to TCNT during counting
- When TCNT is cleared to H'0000 under the condition specified in the CCLR[2:0] bits in TCR

Note: TBTM must be modified only while TCNT stops.

Figure 11.19 shows an operation example in which PWM mode 1 is designated for channel 0 and buffer operation is designated for TGRA_0 and TGRC_0. The settings used in this example are TCNT_0 clearing by compare match B, 1 output at compare match A, and 0 output at compare match B. The TTSA bit in TBTM_0 is set to 1.

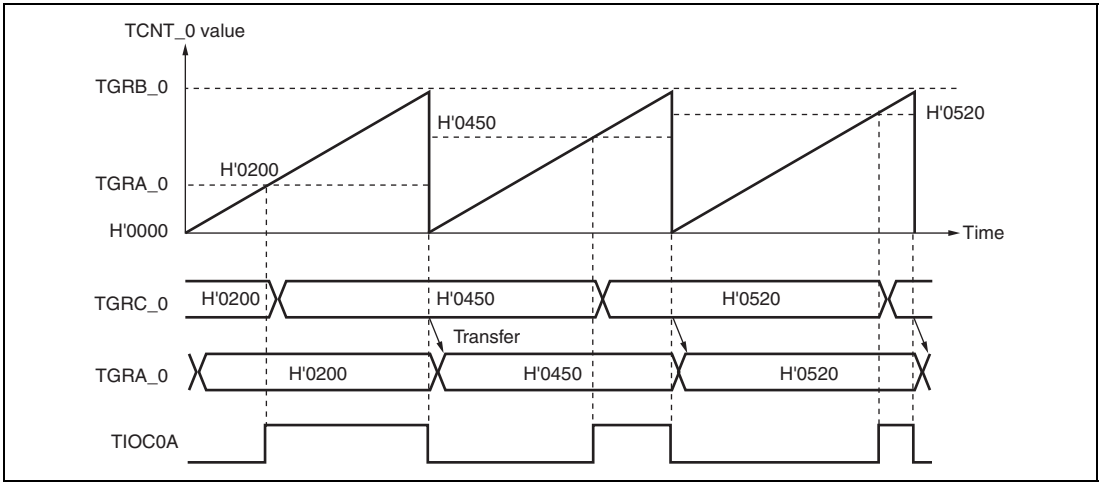


Figure 11.19 Example of Buffer Operation When TCNT_0 Clearing Is Selected for TGRC_0 to TGRA_0 Transfer Timing

11.4.4 Cascaded Operation

In cascaded operation, two 16-bit counters for different channels are used together as a 32-bit counter.

This function works by counting the channel 1 counter clock upon overflow/underflow of TCNT_2 as set in bits TPSC[2:0] in TCR.

Underflow occurs only when the lower 16-bit TCNT is in phase-counting mode.

Table 11.44 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 11.44 Cascaded Combinations

Combination	Upper 16 Bits	Lower 16 Bits
Channels 1 and 2	TCNT_1	TCNT_2

For simultaneous input capture of TCNT_1 and TCNT_2 during cascaded operation, additional input capture input pins can be specified by the input capture control register (TICCR). Edge detection as the condition for input capture is the detection of edges in the signal produced by taking the logical OR of the signals on the main and additional pins. For details, refer to (4),

Cascaded Operation Example (c). For input capture in cascade connection, refer to section 11.7.22, Simultaneous Capture of TCNT_1 and TCNT_2 in Cascade Connection.

Table 11.45 shows the TICCR setting and input capture input pins.

Table 11.45 TICCR Setting and Input Capture Input Pins

Target Input Capture	TICCR Setting	Input Capture Input Pins
Input capture from TCNT_1 to TGRA_1	I2AE bit = 0 (initial value)	TIOC1A
	I2AE bit = 1	TIOC1A, TIOC2A
Input capture from TCNT_1 to TGRB_1	I2BE bit = 0 (initial value)	TIOC1B
	I2BE bit = 1	TIOC1B, TIOC2B
Input capture from TCNT_2 to TGRA_2	I1AE bit = 0 (initial value)	TIOC2A
	I1AE bit = 1	TIOC2A, TIOC1A
Input capture from TCNT_2 to TGRB_2	I1BE bit = 0 (initial value)	TIOC2B
	I1BE bit = 1	TIOC2B, TIOC1B

Example of Cascaded Operation Setting Procedure: Figure 11.20 shows an example of the setting procedure for cascaded operation.

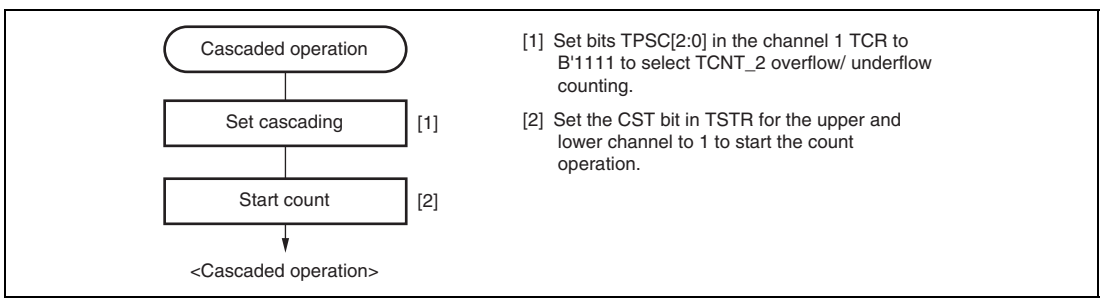


Figure 11.20 Cascaded Operation Setting Procedure

Cascaded Operation Example (a): Figure 11.21 illustrates the operation when TCNT_2 overflow/underflow counting has been set for TCNT_1 and phase counting mode 1 has been designated for channel 2.

TCNT_1 is incremented by TCNT_2 overflow and decremented by TCNT_2 underflow.

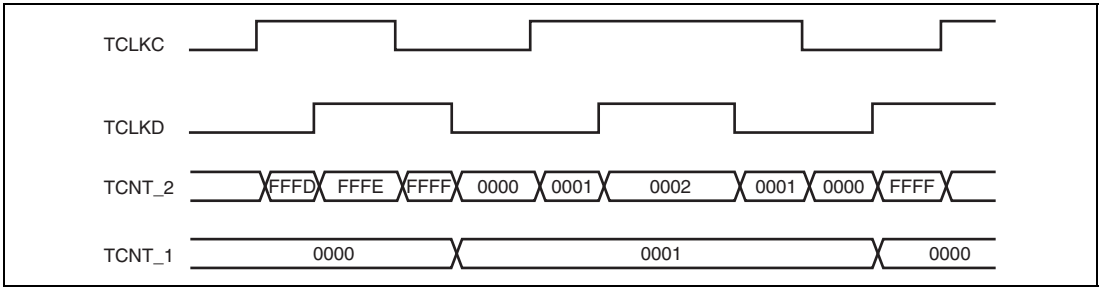


Figure 11.21 Cascaded Operation Example (a)

Cascaded Operation Example (b): Figure 11.22 illustrates the operation when TCNT_1 and TCNT_2 have been cascaded and the I2AE bit in TICCR has been set to 1 to include the TIOC2A pin in the TGRA_1 input capture conditions. In this example, the IOA[3:0] bits in TIOR_1 have selected the TIOC1A rising edge for the input capture timing while the IOA[3:0] bits in TIOR_2 have selected the TIOC2A rising edge for the input capture timing.

Under these conditions, the rising edge of both TIOC1A and TIOC2A is used for the TGRA_1 input capture condition. For the TGRA_2 input capture condition, the TIOC2A rising edge is used.

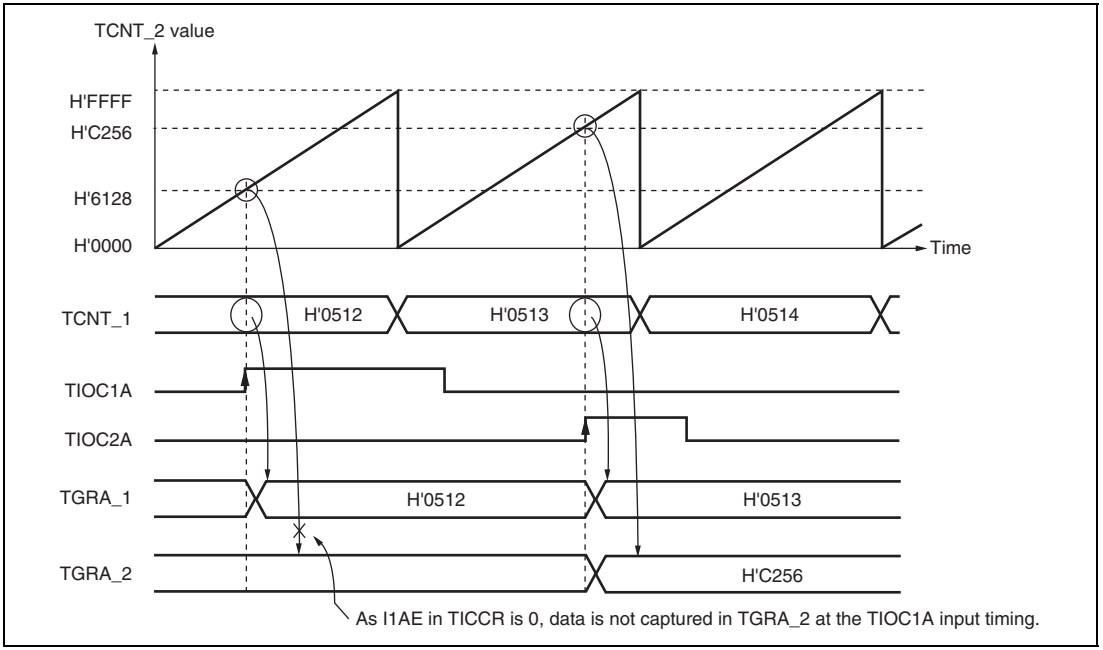


Figure 11.22 Cascaded Operation Example (b)

Cascaded Operation Example (c): Figure 11.23 illustrates the operation when TCNT_1 and TCNT_2 have been cascaded and the I2AE and I1AE bits in TICCR have been set to 1 to include the TIOC2A and TIOC1A pins in the TGRA_1 and TGRA_2 input capture conditions, respectively. In this example, the IOA[3:0] bits in both TIOR_1 and TIOR_2 have selected both the rising and falling edges for the input capture timing. Under these conditions, the ORed result of TIOC1A and TIOC2A input is used for the TGRA_1 and TGRA_2 input capture conditions.

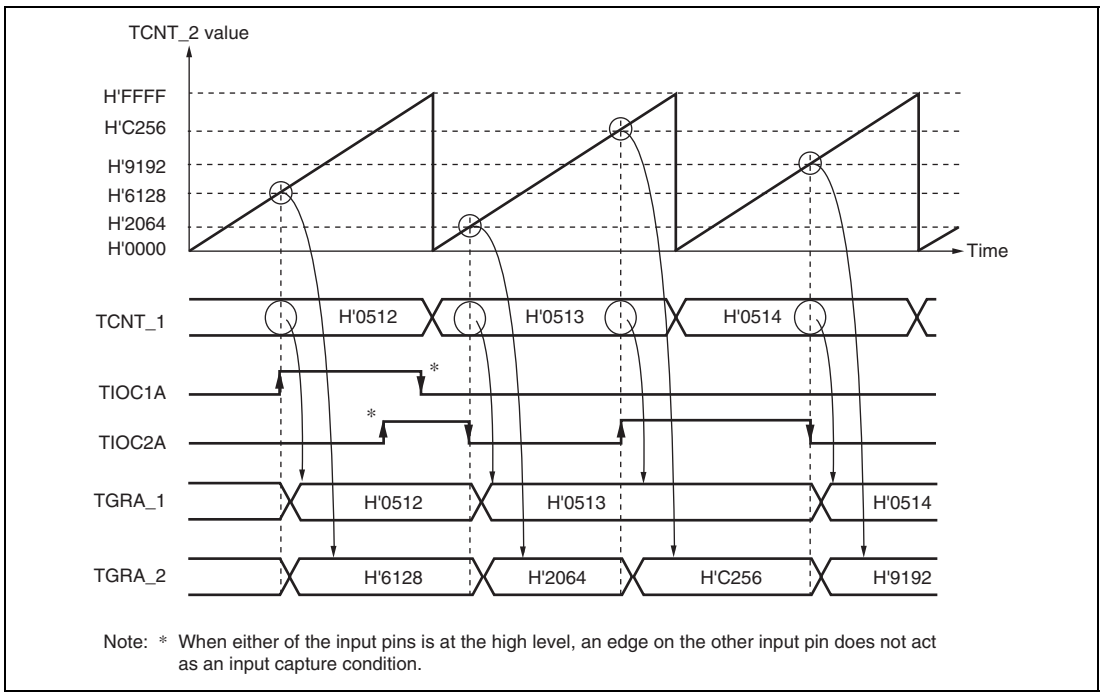


Figure 11.23 Cascaded Operation Example (c)

Cascaded Operation Example (d): Figure 11.24 illustrates the operation when TCNT_1 and TCNT_2 have been cascaded and the I2AE bit in TICCR has been set to 1 to include the TIOC2A pin in the TGRA_1 input capture conditions. In this example, the IOA[3:0] bits in TIOR_1 have selected TGRA_0 compare match or input capture occurrence for the input capture timing while the IOA[3:0] bits in TIOR_2 have selected the TIOC2A rising edge for the input capture timing.

Under these conditions, as TIOR_1 has selected TGRA_0 compare match or input capture occurrence for the input capture timing, the TIOC2A edge is not used for TGRA_1 input capture condition although the I2AE bit in TICCR has been set to 1.

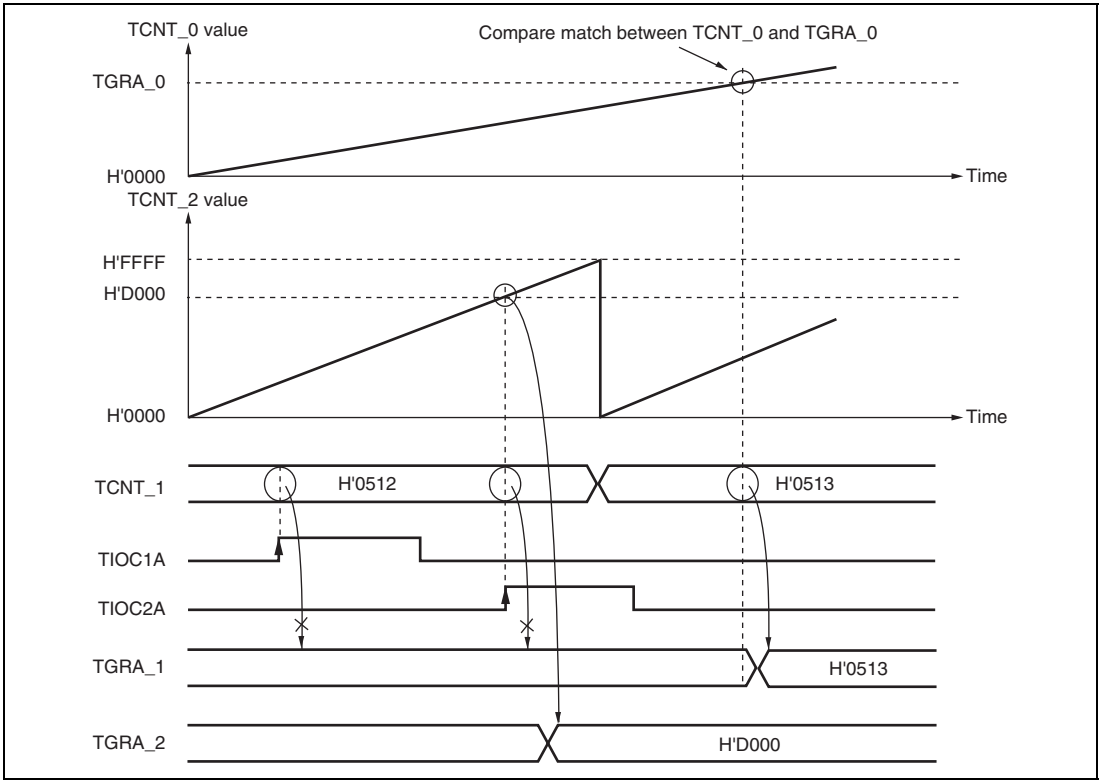


Figure 11.24 Cascaded Operation Example (d)

11.4.5 PWM Modes

In PWM mode, PWM waveforms are output from the output pins. The output level can be selected as 0, 1, or toggle output in response to a compare match of each TGR.

TGR registers settings can be used to output a PWM waveform in the range of 0% to 100% duty.

Designating TGR compare match as the counter clearing source enables the period to be set in that register. All channels can be designated for PWM mode independently. Synchronous operation is also possible.

There are two PWM modes, as described below.

1. PWM mode 1

   PWM output is generated from the TIOCA and TIOCC pins by pairing TGRA with TGRB and TGRC with TGRD. The output specified by bits IOA[3:0] and IOC[3:0] in TIOR is output from the TIOCA and TIOCC pins at compare matches A and C, and the output specified by bits IOB[3:0] and IOD[3:0] in TIOR is output at compare matches B and D. The initial output value is the value set in IOA[3:0] or IOC[3:0]. If the set values of paired TGRs are identical, the output value does not change when a compare match occurs.

   In PWM mode 1, a maximum 8-phase PWM output is possible.

2. PWM mode 2

   PWM output is generated using one TGR as the cycle register and the others as duty registers. The output specified in TIOR is performed by means of compare matches. Upon counter clearing by a cycle register compare match, the output value of each pin is the initial value set in TIOR. If the set values of the cycle and duty registers are identical, the output value does not change when a compare match occurs.

   In PWM mode 2, a maximum 8-phase PWM output is possible in combination use with synchronous operation.

The correspondence between PWM output pins and registers is shown in table 11.46.

Table 11.46 PWM Output Registers and Output Pins

		Output Pins	
Channel	**Registers**	**PWM Mode 1**	**PWM Mode 2**
0	TGRA_0	TIOC0A	TIOC0A
	TGRB_0		TIOC0B
	TGRC_0	TIOC0C	TIOC0C
	TGRD_0		TIOC0D
1	TGRA_1	TIOC1A	TIOC1A
	TGRB_1		TIOC1B
2	TGRA_2	TIOC2A	TIOC2A
	TGRB_2		TIOC2B
3	TGRA_3	TIOC3A	Cannot be set
	TGRB_3		Cannot be set
	TGRC_3	TIOC3C	Cannot be set
	TGRD_3		Cannot be set
4	TGRA_4	TIOC4A	Cannot be set
	TGRB_4		Cannot be set
	TGRC_4	TIOC4C	Cannot be set
	TGRD_4		Cannot be set

Note: In PWM mode 2, PWM output is not possible for the TGR register in which the period is set.

Example of PWM Mode Setting Procedure: Figure 11.25 shows an example of the PWM mode setting procedure.

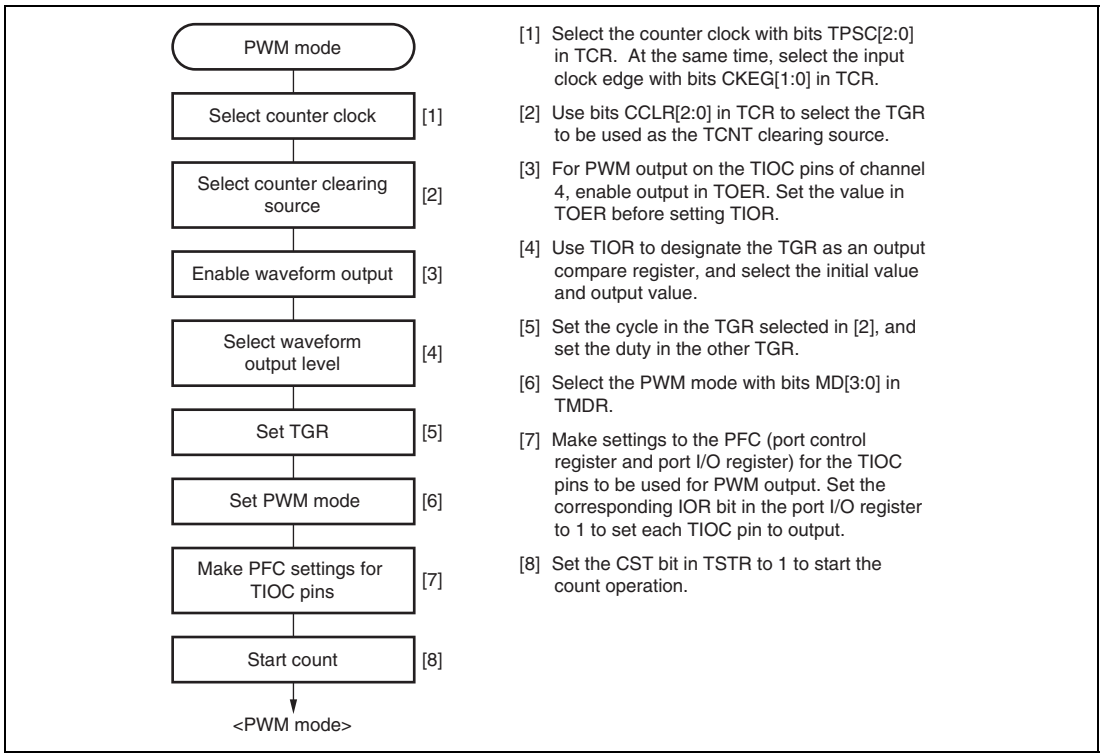


Figure 11.25 Example of PWM Mode Setting Procedure

Examples of PWM Mode Operation: Figure 11.26 shows an example of PWM mode 1 operation.

In this example, TGRA compare match is set as the TCNT clearing source, 0 is set for the TGRA initial output value and output value, and 1 is set as the TGRB output value.

In this case, the value set in TGRA is used as the period, and the values set in the TGRB registers are used as the duty levels.

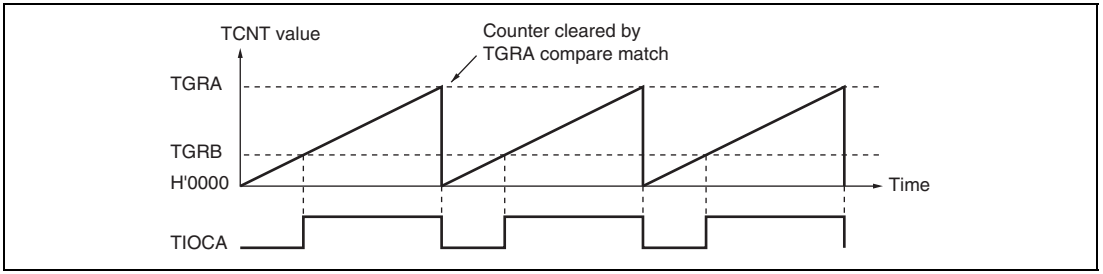


Figure 11.26 Example of PWM Mode 1 Operation

Figure 11.27 shows an example of PWM mode 2 operation.

In this example, synchronous operation is designated for channels 0 and 1, TGRB_1 compare match is set as the TCNT clearing source, and 0 is set for the initial output value and 1 for the output value of the other TGR registers (TGRA_0 to TGRD_0, TGRA_1), outputting a 5-phase PWM waveform.

In this case, the value set in TGRB_1 is used as the cycle, and the values set in the other TGRs are used as the duty levels.

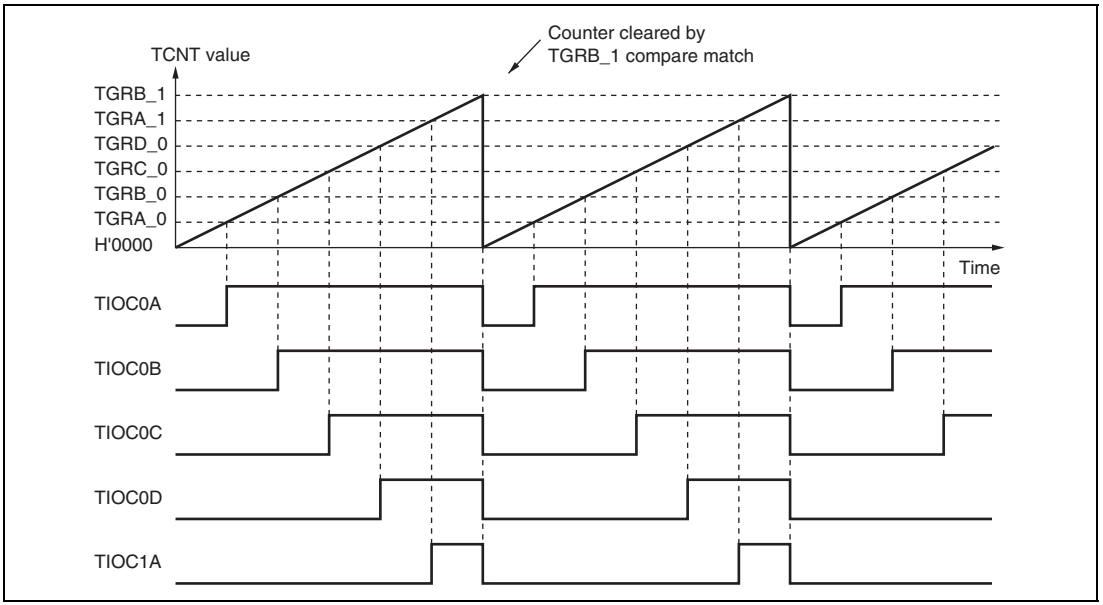


Figure 11.27 Example of PWM Mode 2 Operation

Figure 11.28 shows examples of PWM waveform output with 0% duty and 100% duty in PWM mode.

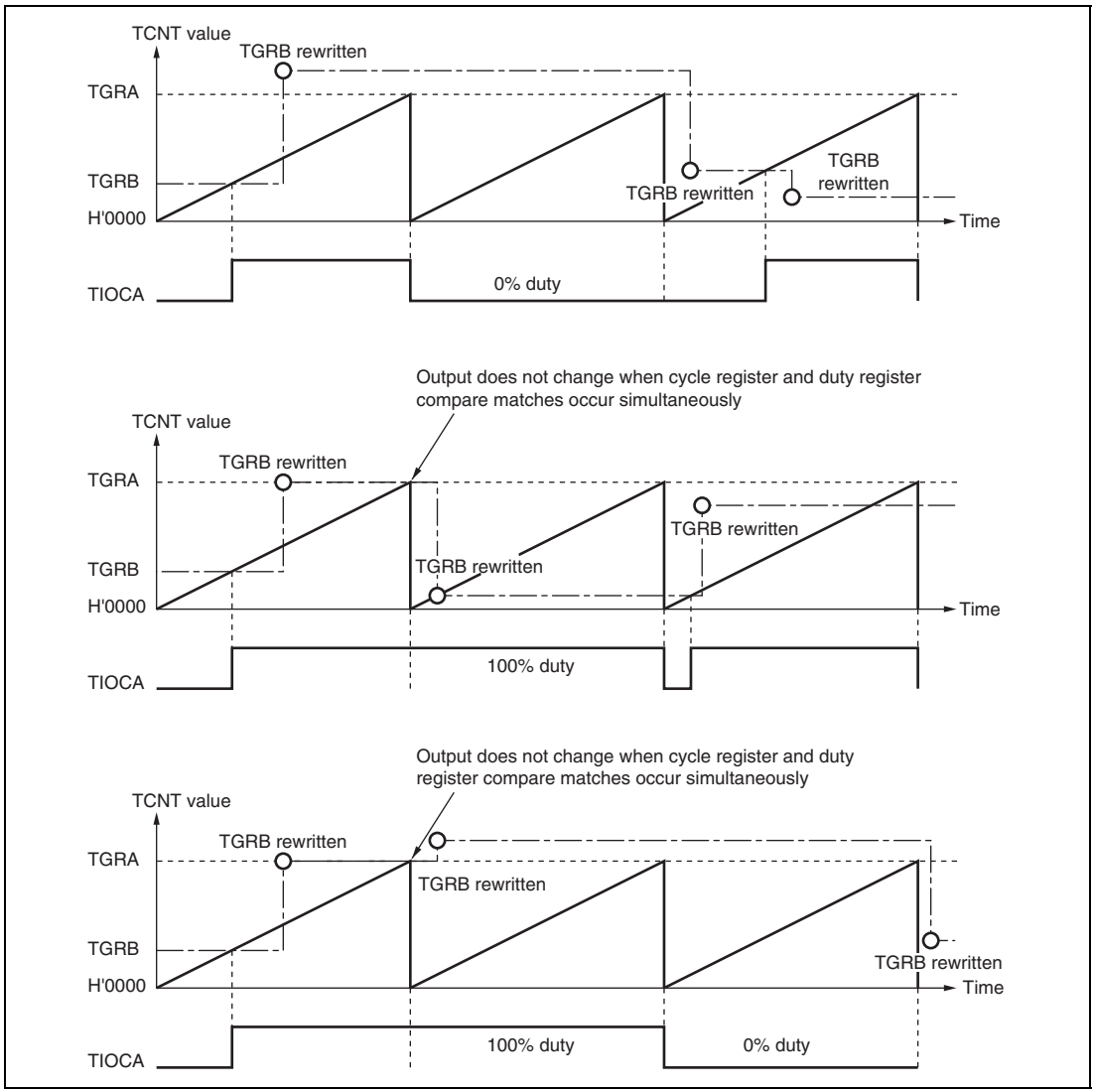


Figure 11.28 Example of PWM Mode Operation (PWM Waveform Output with 0% Duty and 100% Duty)

11.4.6 Phase Counting Mode

In phase counting mode, the phase difference between two external clock inputs is detected and TCNT is incremented/decremented accordingly. This mode can be set for channels 1 and 2.

When phase counting mode is set, an external clock is selected as the counter input clock and TCNT operates as an up/down-counter regardless of the setting of bits TPSC[2:0] and bits CKEG[1:0] in TCR. However, the functions of bits CCLR[1:0] in TCR, and of TIOR, TIER, and TGR, are valid, and input capture/compare match and interrupt functions can be used.

This can be used for two-phase encoder pulse input.

If overflow occurs when TCNT is counting up, the TCFV flag in TSR is set; if underflow occurs when TCNT is counting down, the TCFU flag is set.

The TCFD bit in TSR is the count direction flag. Reading the TCFD flag reveals whether TCNT is counting up or down.

In phase counting mode, the external clock pins TCLKA, TCLKB, TCLKC, and TCLKD can be used as two-phase encoder pulse inputs. Table 11.47 shows the correspondence between external clock pins and channels.

Table 11.47 Phase Counting Mode Clock Input Pins

	External Clock Pins	
Channels	**A-Phase**	**B-Phase**
When channel 1 is set to phase counting mode	TCLKA	TCLKB
When channel 2 is set to phase counting mode	TCLKC	TCLKD

Example of Phase Counting Mode Setting Procedure: Figure 11.29 shows an example of the phase counting mode setting procedure.

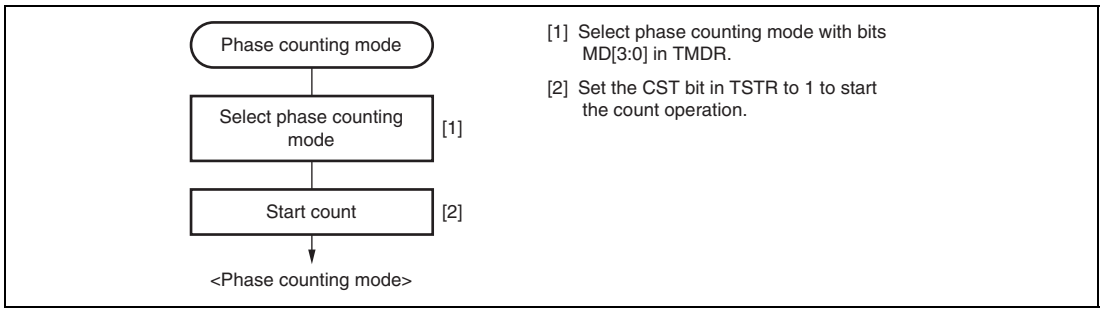


Figure 11.29 Example of Phase Counting Mode Setting Procedure

Examples of Phase Counting Mode Operation: In phase counting mode, TCNT counts up or down according to the phase difference between two external clocks. There are four modes, according to the count conditions.

1. Phase counting mode 1

   Figure 11.30 shows an example of phase counting mode 1 operation, and table 11.48 summarizes the TCNT up/down-count conditions.

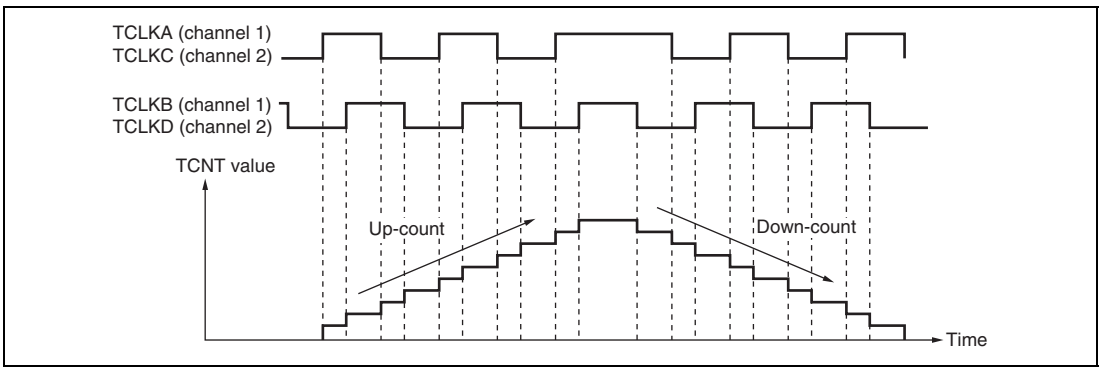


Figure 11.30 Example of Phase Counting Mode 1 Operation

Table 11.48 Up/Down-Count Conditions in Phase Counting Mode 1

TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level	↑	Up-count
Low level	↓	
↑	Low level	
↓	High level	
High level	↓	Down-count
Low level	↑	
↑	High level	
↓	Low level	

[Legend]

↑: Rising edge

↓: Falling edge

2. Phase counting mode 2

Figure 11.31 shows an example of phase counting mode 2 operation, and table 11.49 summarizes the TCNT up/down-count conditions.

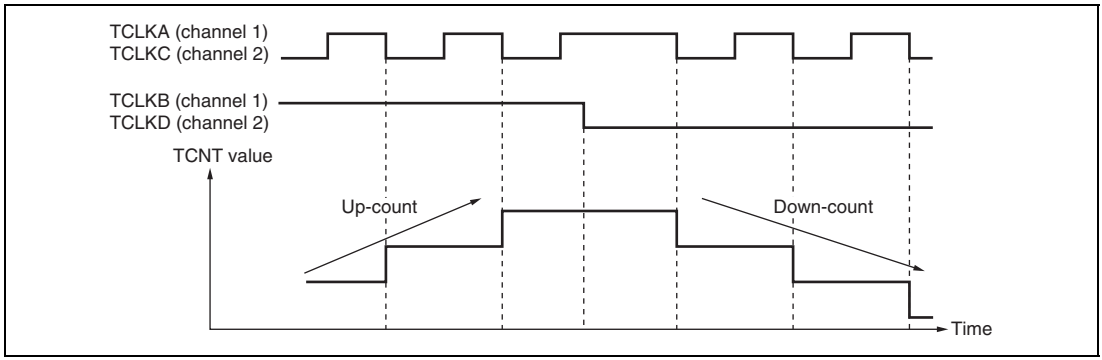


Figure 11.31 Example of Phase Counting Mode 2 Operation

Table 11.49 Up/Down-Count Conditions in Phase Counting Mode 2

TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level	⌐↑	Don't care
Low level	⌐↓	Don't care
⌐↑	Low level	Don't care
⌐↓	High level	Up-count
High level	⌐↓	Don't care
Low level	⌐↑	Don't care
⌐↑	High level	Don't care
⌐↓	Low level	Down-count

[Legend]

⌐↑: Rising edge

⌐↓: Falling edge

3. Phase counting mode 3

Figure 11.32 shows an example of phase counting mode 3 operation, and table 11.50 summarizes the TCNT up/down-count conditions.

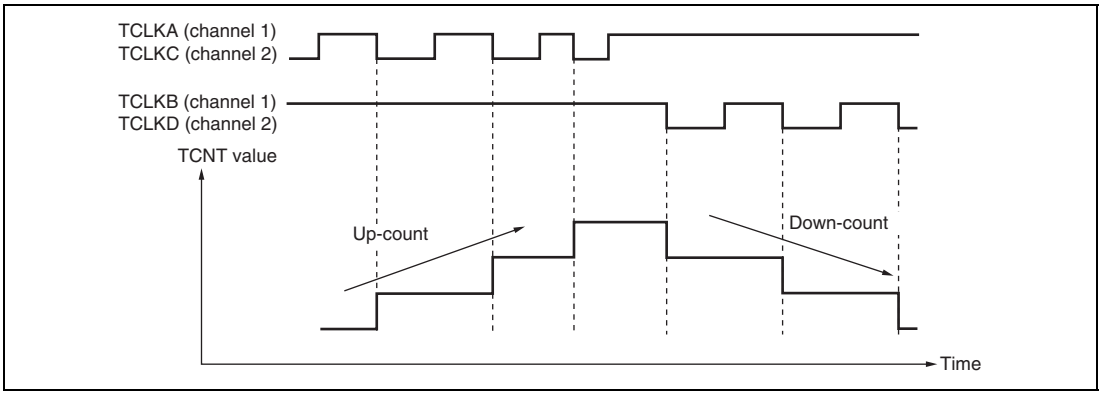


Figure 11.32 Example of Phase Counting Mode 3 Operation

Table 11.50 Up/Down-Count Conditions in Phase Counting Mode 3

TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level	↑	Don't care
Low level	↓	Don't care
↑	Low level	Don't care
↓	High level	Up-count
High level	↓	Down-count
Low level	↑	Don't care
↑	High level	Don't care
↓	Low level	Don't care

[Legend]

↑: Rising edge

↓: Falling edge

4. Phase counting mode 4

Figure 11.33 shows an example of phase counting mode 4 operation, and table 11.51 summarizes the TCNT up/down-count conditions.

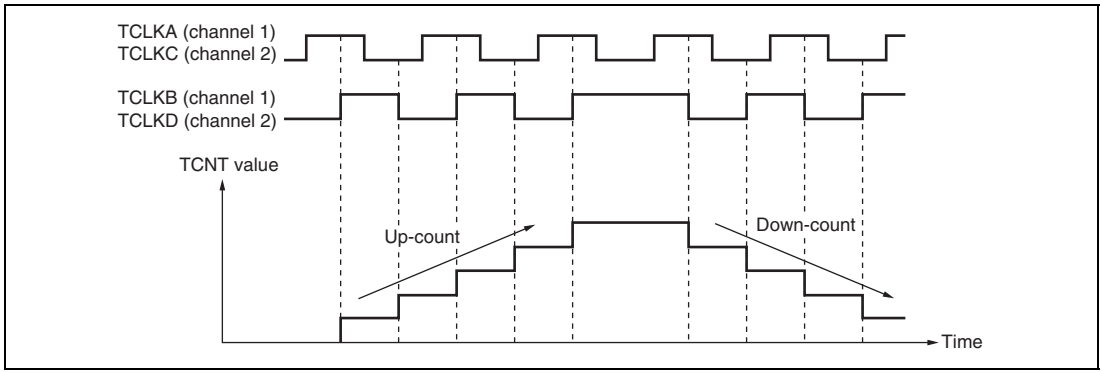


Figure 11.33 Example of Phase Counting Mode 4 Operation

Table 11.51 Up/Down-Count Conditions in Phase Counting Mode 4

TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level	⌜	Up-count
Low level	⌞	
⌜	Low level	Don't care
⌞	High level	
High level	⌞	Down-count
Low level	⌜	
⌜	High level	Don't care
⌞	Low level	

[Legend]
⌜: Rising edge
⌞: Falling edge

Phase Counting Mode Application Example: Figure 11.34 shows an example in which channel 1 is in phase counting mode, and channel 1 is coupled with channel 0 to input servo motor 2-phase encoder pulses in order to detect position or speed.

Channel 1 is set to phase counting mode 1, and the encoder pulse A-phase and B-phase are input to TCLKA and TCLKB.

Channel 0 operates with TCNT counter clearing by TGRC_0 compare match; TGRA_0 and TGRC_0 are used for the compare match function and are set with the speed control period and position control period. TGRB_0 is used for input capture, with TGRB_0 and TGRD_0 operating in buffer mode. The channel 1 counter input clock is designated as the TGRB_0 input capture source, and the pulse widths of 2-phase encoder 4-multiplication pulses are detected.

TGRA_1 and TGRB_1 for channel 1 are designated for input capture, and channel 0 TGRA_0 and TGRC_0 compare matches are selected as the input capture source and store the up/down-counter values for the control periods.

This procedure enables the accurate detection of position and speed.

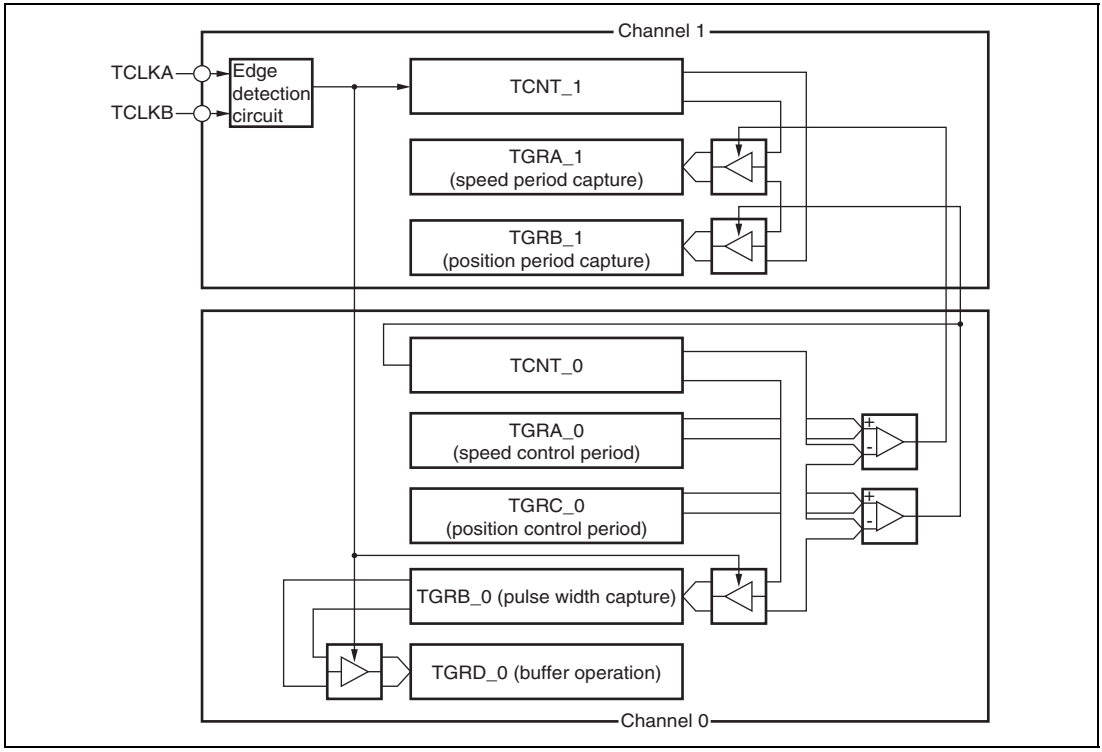


Figure 11.34 Phase Counting Mode Application Example

11.4.7 Reset-Synchronized PWM Mode

In the reset-synchronized PWM mode, three-phase output of positive and negative PWM waveforms that share a common wave transition point can be obtained by combining channels 3 and 4.

When reset-synchronized PWM mode is selected, the TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, and TIOC4D pins function as PWM output pins, and the TIOC3A pin can be set as a toggle output synchronized with the PWM period. In addition, TCNT_3 and TCNT_4 function as up-counters.

Table 11.52 shows the PWM output pins used. Table 11.53 shows the settings of the registers.

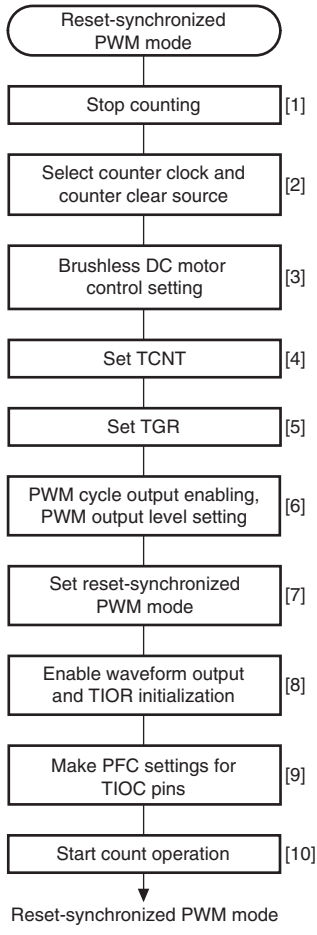
Table 11.52 Output Pins for Reset-Synchronized PWM Mode

Channel	Output Pin	Description
3	TIOC3A	Toggle output synchronized with the PWM period (or I/O port)
	TIOC3B	PWM output pin 1
	TIOC3D	PWM output pin 1' (negative-phase waveform of PWM output 1)
4	TIOC4A	PWM output pin 2
	TIOC4C	PWM output pin 2' (negative-phase waveform of PWM output 2)
	TIOC4B	PWM output pin 3
	TIOC4D	PWM output pin 3' (negative-phase waveform of PWM output 3)

Table 11.53 Register Settings for Reset-Synchronized PWM Mode

Register	Description of Setting
TCNT_3	Initial setting of H'0000
TCNT_4	Initial setting of H'0000
TGRA_3	Set count cycle for TCNT_3
TGRB_3	Sets the turning point for PWM waveform output by the TIOC3B and TIOC3D pins
TGRA_4	Sets the turning point for PWM waveform output by the TIOC4A and TIOC4C pins
TGRB_4	Sets the turning point for PWM waveform output by the TIOC4B and TIOC4D pins

Procedure for Selecting the Reset-Synchronized PWM Mode: Figure 11.35 shows an example of procedure for selecting the reset synchronized PWM mode.



[1] Clear the CST3 and CST4 bits in the TSTR to 0 to halt the counting of TCNT. The reset-synchronized PWM mode must be set up while TCNT_3 and TCNT_4 are halted.

[2] Set bits TPSC[2:0] and CKEG[1:0] in the TCR_3 to select the counter clock and clock edge for channel 3. Set bits CCLR[2:0] in the TCR_3 to select TGRA compare-match as a counter clear source.

[3] When performing brushless DC motor control, set bit BDC in the timer gate control register (TGCR) and set the feedback signal input source and output chopping or gate signal direct output.

[4] Reset TCNT_3 and TCNT_4 to H'0000.

[5] TGRA_3 is the period register. Set the waveform period value in TGRA_3. Set the transition timing of the PWM output waveforms in TGRB_3, TGRA_4, and TGRB_4. Set times within the compare-match range of TCNT_3.

$X \leq$ TGRA_3 (X: set value).

[6] Select enabling/disabling of toggle output synchronized with the PMW cycle using bit PSYE in the timer output control register (TOCR1), and set the PWM output level with bits OLSP and OLSN. When specifying the PWM output level by using TOLBR as a buffer for TOCR2, see figure 11.3.

[7] Set bits MD[3:0] in TMDR_3 to B'1000 to select the reset-synchronized PWM mode. Do not set to TMDR_4.

[8] Set the enabling/disabling of the PWM waveform output pin in TOER. Next, set TIORH_3, TIORL_3, TIORH_4, and TIORL_4 to H'00.

[9] Make settings to the PFC (port control register and port I/O register) for the TIOC pins to be used for PWM input. Set the corresponding IOR bit in the port I/O register to 1 to set each TIOC pin to output.

[10] Set the CST3 bit in the TSTR to 1 to start the count operation.

Note: The output waveform starts to toggle operation at the point of TCNT_3 = TGRA_3 = X by setting X = TGRA, i.e., cycle = duty.

Figure 11.35 Procedure for Selecting Reset-Synchronized PWM Mode

Reset-Synchronized PWM Mode Operation: Figure 11.36 shows an example of operation in the reset-synchronized PWM mode. TCNT_3 and TCNT_4 operate as upcounters. The counter is cleared when a TCNT_3 and TGRA_3 compare-match occurs, and then begins incrementing from H'0000. The PWM output pin output toggles with each occurrence of a TGRB_3, TGRA_4, TGRB_4 compare-match, and upon counter clears.

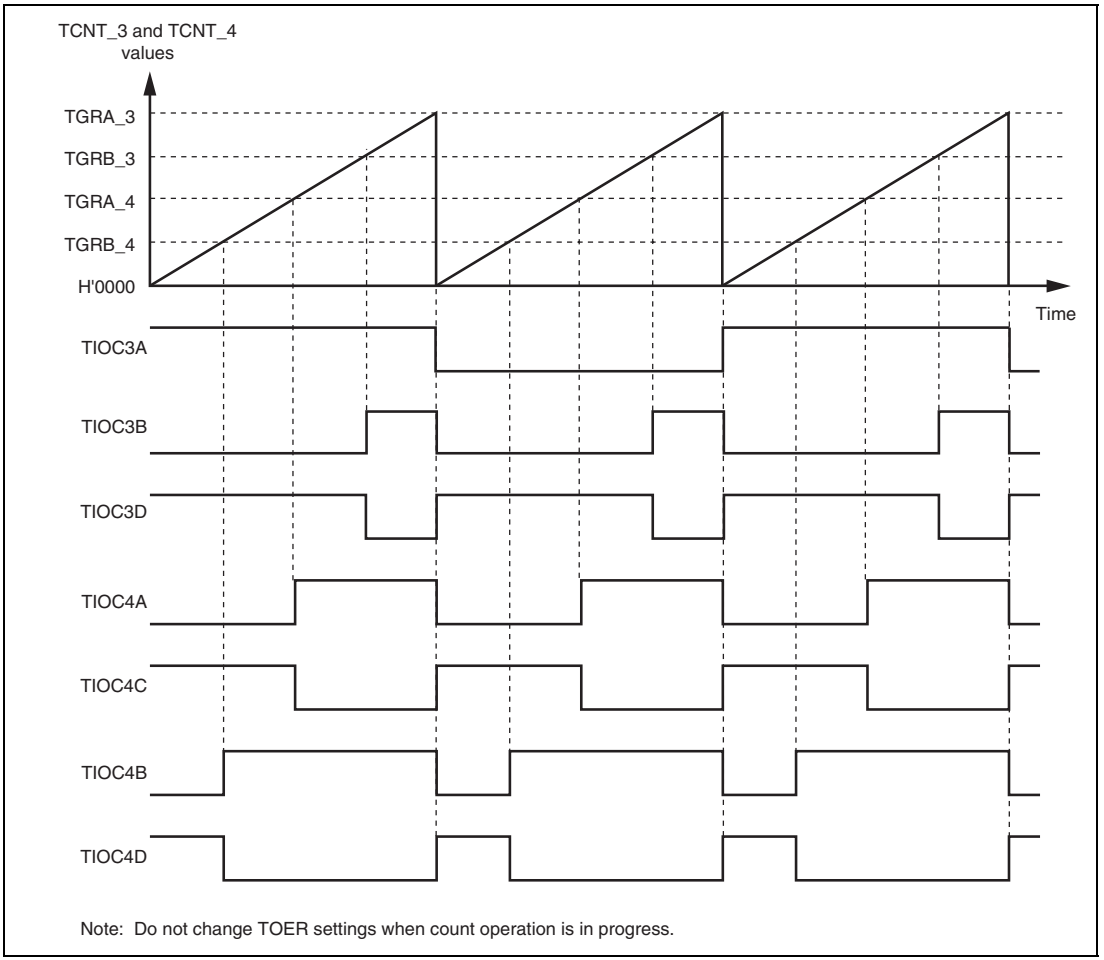


Figure 11.36 Reset-Synchronized PWM Mode Operation Example (When TOCR's OLSN = 1 and OLSP = 1)

11.4.8 Complementary PWM Mode

In the complementary PWM mode, three-phase output of non-overlapping positive and negative PWM waveforms can be obtained by combining channels 3 and 4. PWM waveforms without non-overlapping interval is also available.

In complementary PWM mode, TIOC3B, TIOC3D, TIOC4A, TIOC4B, TIOC4C, and TIOC4D pins function as PWM output pins, the TIOC3A pin can be set for toggle output synchronized with the PWM period. TCNT_3 and TCNT_4 function as up/down counters.

Table 11.54 shows the PWM output pins used. Table 11.55 shows the settings of the registers used. Figure 11.37 describes a block diagram of channels 3 and 4 in complementary PWM mode.

A function to directly cut off the PWM output by using an external signal is supported as a port function.

Table 11.54 Output Pins for Complementary PWM Mode

Channel	Output Pin	Description
3	TIOC3A	Toggle output synchronized with PWM period (or I/O port)
	TIOC3B	PWM output pin 1
	TIOC3C	I/O port*
	TIOC3D	PWM output pin 1' (non-overlapping negative-phase waveform of PWM output 1; PWM output without non-overlapping interval is also available)
4	TIOC4A	PWM output pin 2
	TIOC4B	PWM output pin 3
	TIOC4C	PWM output pin 2' (non-overlapping negative-phase waveform of PWM output 2; PWM output without non-overlapping interval is also available)
	TIOC4D	PWM output pin 3' (non-overlapping negative-phase waveform of PWM output 3; PWM output without non-overlapping interval is also available)

Note: * Avoid setting the TIOC3C pin as a timer I/O pin in the complementary PWM mode.

Table 11.55 Register Settings for Complementary PWM Mode

Channel	Counter/Register	Description	Read/Write from CPU
3	TCNT_3	Start of up-count from value set in dead time register	Maskable by TRWER setting*
	TGRA_3	Set TCNT_3 upper limit value (1/2 PWM cycle + dead time)	Maskable by TRWER setting*
	TGRB_3	PWM output 1 compare register	Maskable by TRWER setting*
	TGRC_3	TGRA_3 buffer register	Always readable/writable
	TGRD_3	PWM output 1/TGRB_3 buffer register	Always readable/writable
4	TCNT_4	Up-count start, initialized to H'0000	Maskable by TRWER setting*
	TGRA_4	PWM output 2 compare register	Maskable by TRWER setting*
	TGRB_4	PWM output 3 compare register	Maskable by TRWER setting*
	TGRC_4	PWM output 2/TGRA_4 buffer register	Always readable/writable
	TGRD_4	PWM output 3/TGRB_4 buffer register	Always readable/writable
Timer dead time data register (TDDR)		Set TCNT_4 and TCNT_3 offset value (dead time value)	Maskable by TRWER setting*
Timer cycle data register (TCDR)		Set TCNT_4 upper limit value (1/2 PWM cycle)	Maskable by TRWER setting*
Timer cycle buffer register (TCBR)		TCDR buffer register	Always readable/writable
Subcounter (TCNTS)		Subcounter for dead time generation	Read-only
Temporary register 1 (TEMP1)		PWM output 1/TGRB_3 temporary register	Not readable/writable
Temporary register 2 (TEMP2)		PWM output 2/TGRA_4 temporary register	Not readable/writable
Temporary register 3 (TEMP3)		PWM output 3/TGRB_4 temporary register	Not readable/writable
Temporary register 4 (TEMP4)		TGRA_3 temporary register	Not readable/writable
Temporary register 5 (TEMP5)		TCDR temporary register	Not readable/writable

Note: * Access can be enabled or disabled according to the setting of bit 0 (RWE) in TRWER (timer read/write enable register).

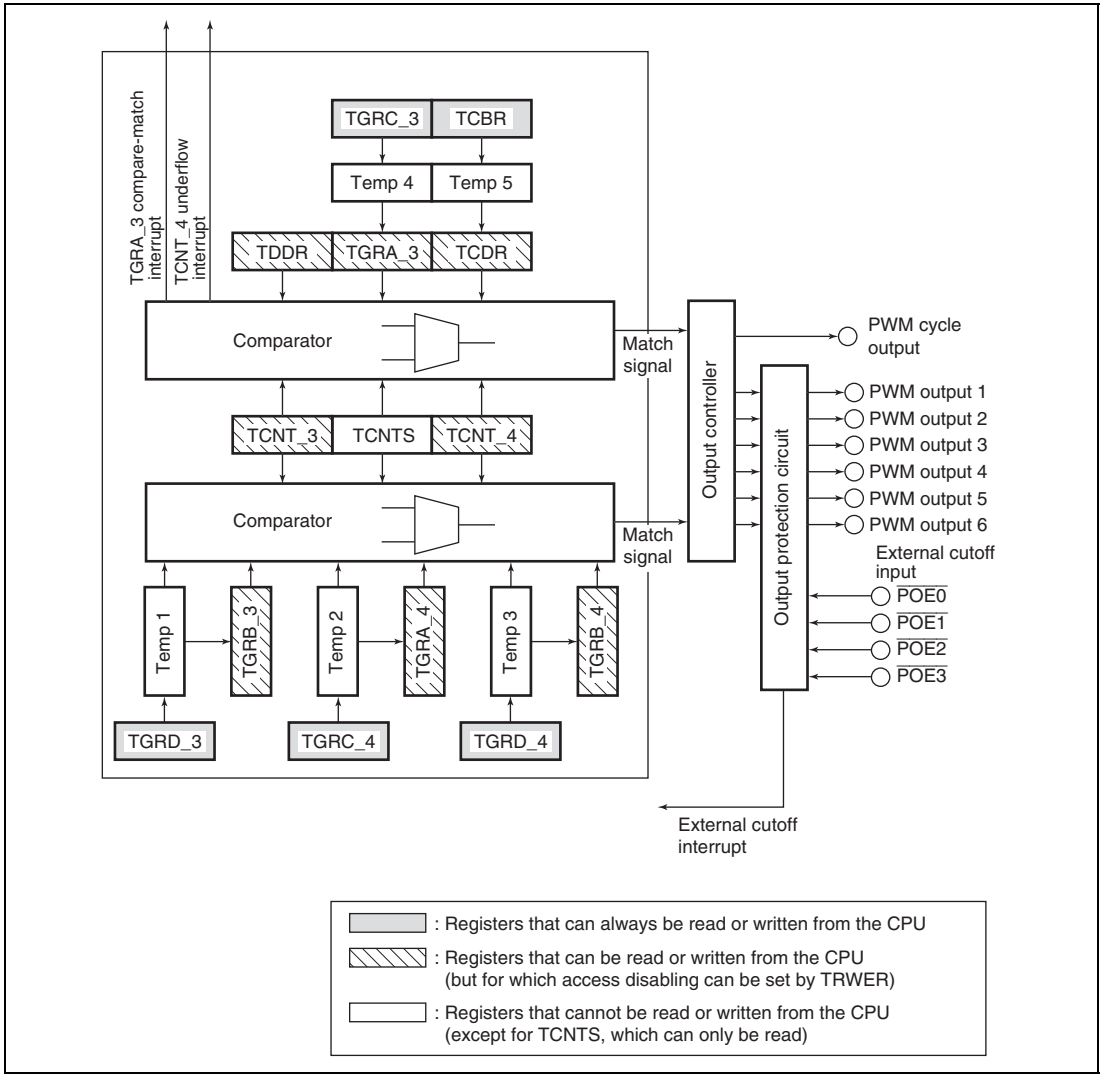
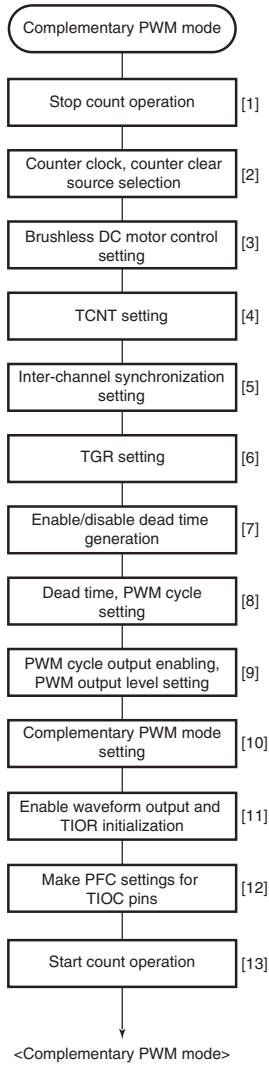


Figure 11.37 Block Diagram of Channels 3 and 4 in Complementary PWM Mode

(1) Example of Complementary PWM Mode Setting Procedure

An example of the complementary PWM mode setting procedure is shown in figure 11.38.



[1] Clear bits CST3 and CST4 in the timer start register (TSTR) to 0, and halt timer counter (TCNT) operation. Perform complementary PWM mode setting when TCNT_3 and TCNT_4 are stopped.

[2] Set the same counter clock and clock edge for channels 3 and 4 with bits TPSC[2:0] and bits CKEG[1:0] in the timer control register (TCR). Use bits CCLR[2:0] to set synchronous clearing only when restarting by a synchronous clear from another channel during complementary PWM mode operation.

[3] When performing brushless DC motor control, set bit BDC in the timer gate control register (TGCR) and set the feedback signal input source and output chopping or gate signal direct output.

[4] Set the dead time in TCNT_3. Set TCNT_4 to H'0000.

[5] Set only when restarting by a synchronous clear from another channel during complementary PWM mode operation. In this case, synchronize the channel generating the synchronous clear with channels 3 and 4 using the timer synchro register (TSYR).

[6] Set the output PWM duty in the duty registers (TGRB_3, TGRA_4, TGRB_4) and buffer registers (TGRD_3, TGRC_4, TGRD_4). Set the same initial value in each corresponding TGR.

[7] This setting is necessary only when no dead time should be generated. Make appropriate settings in the timer dead time enable register (TDER) so that no dead time is generated.

[8] Set the dead time in the dead time register (TDDR), 1/2 the PWM cycle in the timer cycle data register (TCDR) and timer cycle buffer register (TCBR), and 1/2 the PWM cycle plus the dead time in TGRA_3 and TGRC_3. When no dead time generation is selected, set 1 in TDDR and 1/2 the PWM cycle + 1 in TGRA_3 and TGRC_3.

[9] Select enabling/disabling of toggle output synchronized with the PWM cycle using bit PSYE in the timer output control register 1 (TOCR1), and set the PWM output level with bits OLSP and OLSN. When specifying the PWM output level by using TOLBR as a buffer for TOCR_2, see figure 11.3.

[10] Select complementary PWM mode in timer mode register 3 (TMDR_3). Do not set in TMDR_4.

[11] Set enabling/disabling of PWM waveform output pin output in the timer output master enable register (TOER). Next, set TIORH_3, TIORL_3, TIORH_4, and TIORL_4 to H'00.

[12] Make settings to the PFC (port control register and port I/O register) for the TIOC pins to be used for PWM input. Set the corresponding IOR bit in the port I/O register to 1 to set each TIOC pin to output.

[13] Set bits CST3 and CST4 in TSTR to 1 simultaneously to start the count operation.

Figure 11.38 Example of Complementary PWM Mode Setting Procedure

(2) Outline of Complementary PWM Mode Operation

In complementary PWM mode, 6-phase PWM output is possible. Figure 11.39 illustrates counter operation in complementary PWM mode, and figure 11.40 shows an example of complementary PWM mode operation.

(a) Counter Operation

In complementary PWM mode, three counters—TCNT_3, TCNT_4, and TCNTS—perform up/down-count operations.

TCNT_3 is automatically initialized to the value set in TDDR when complementary PWM mode is selected and the CST bit in TSTR is 0.

When the CST bit is set to 1, TCNT_3 counts up to the value set in TGRA_3, then switches to down-counting when it matches TGRA_3. When the TCNT3 value matches TDDR, the counter switches to up-counting, and the operation is repeated in this way.

TCNT_4 is initialized to H'0000.

When the CST bit is set to 1, TCNT4 counts up in synchronization with TCNT_3, and switches to down-counting when it matches TCDR. On reaching H'0000, TCNT4 switches to up-counting, and the operation is repeated in this way.

TCNTS is a read-only counter. It need not be initialized.

When TCNT_3 matches TCDR during TCNT_3 and TCNT_4 up-counting, down-counting is started, and when TCNTS matches TCDR, the operation switches to up-counting. When TCNTS matches TGRA_3, it is cleared to H'0000.

When TCNT_4 matches TDDR during TCNT_3 and TCNT_4 down-counting, up-counting is started, and when TCNTS matches TDDR, the operation switches to down-counting. When TCNTS reaches H'0000, it is set with the value in TGRA_3.

TCNTS is compared with the compare register and temporary register in which the PWM duty is set during the count operation only.

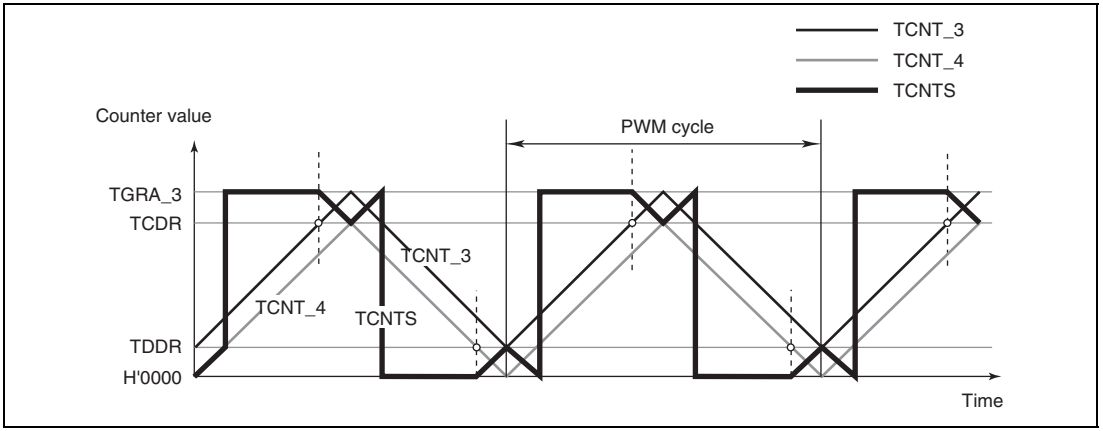


Figure 11.39 Complementary PWM Mode Counter Operation

(b) Register Operation

In complementary PWM mode, nine registers, comprising compare registers, buffer registers, and temporary registers, are used to control the PWM duty. Figure 11.40 shows an example of complementary PWM mode operation.

The registers which are constantly compared with the counters to perform PWM output are TGRB_3, TGRA_4, and TGRB_4. When one of these registers matches the counter, the level set in the corresponding timer output control register (TOCR1 or TOCR2) is output on the PWM pin.

The buffer registers for these compare registers are TGRD_3, TGRC_4, and TGRD_4.

Between a buffer register and compare register there is a temporary register. The temporary registers cannot be accessed by the CPU.

Data in a compare register is changed by writing the new data to the corresponding buffer register. The buffer registers can be read or written at any time.

When overwriting the data in the buffer registers, always write to TGRD_4 last to enable data transfer from the buffer registers to the temporary registers. At this time, transfer is also enabled from the timer cycle register buffer registers (TGRA_3 and TCBR) to the temporary registers. All five temporary registers can be used simultaneously for transfers.

When transfer is enabled during the Ta interval, data written to the buffer register is transferred immediately to the temporary register. Transfer to the temporary register does not take place in the Tb1 or Tb2 interval. Data for which transfer is enabled during either of these intervals is transferred to the temporary register after the interval ends.

The value transferred to a temporary register is transferred to the compare register either when the Tb1 interval ends (when TGRA_3 is matched if TCNTS is counting up) or when the Tb2 interval ends (when H'0000 is matched if TCNTS is counting down). The timing for transfer from the temporary register to the compare register can be selected with bits MD[3:0] in the timer mode register (TMDR). Figure 11.40 shows an example in which the mode is selected in which the change is made in the trough.

In the Tb interval (Tb1 in figure 11.40) in which data transfer to the temporary register is not performed, the temporary register has the same function as the compare register, and is compared with the counter. In this interval, therefore, there are two compare match registers for one-phase output, with the compare register containing the pre-change data, and the temporary register containing the new data. In this interval, the three counters—TCNT_3, TCNT_4, and TCNTS—and two registers—compare register and temporary register—are compared, and PWM output controlled accordingly.

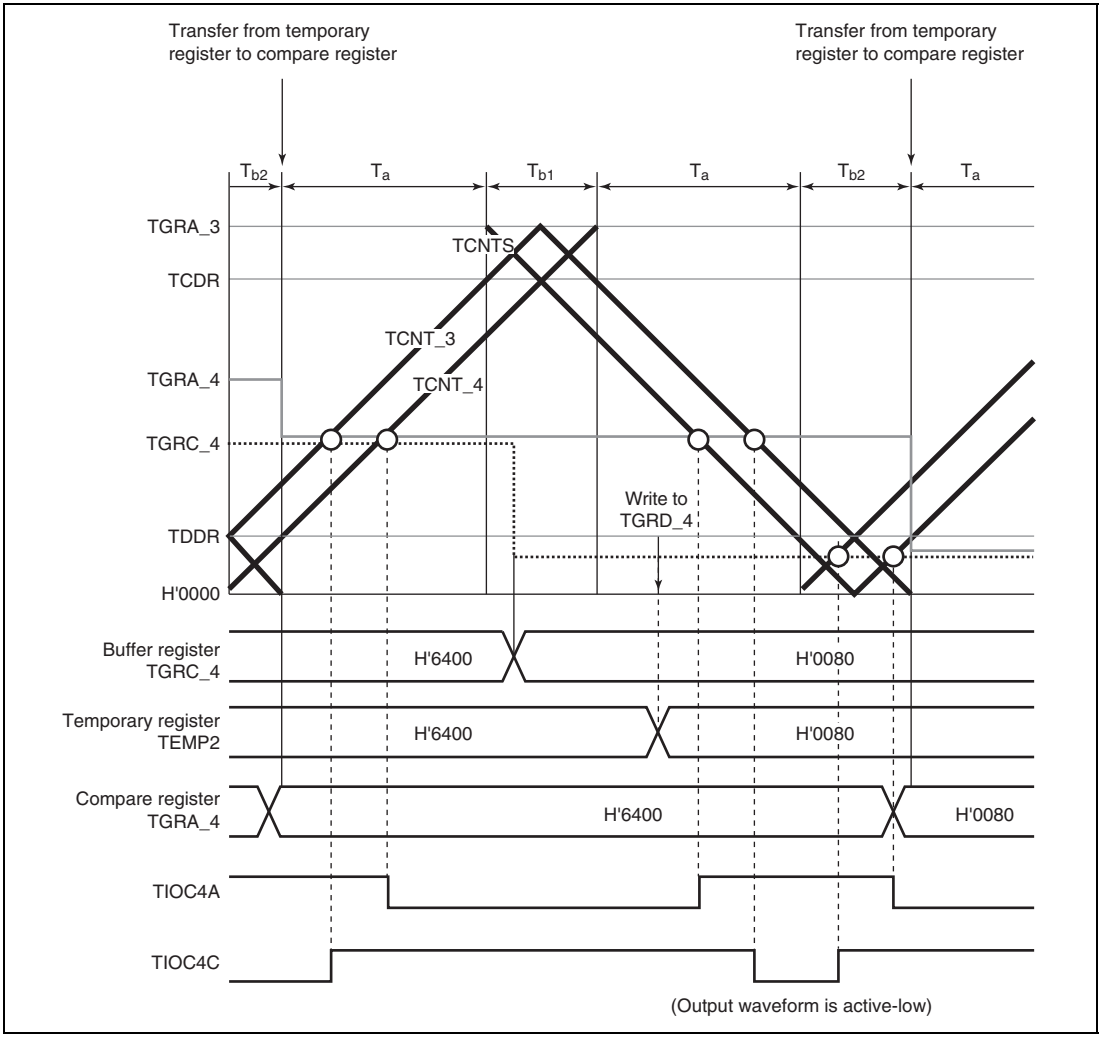


Figure 11.40 Example of Complementary PWM Mode Operation

(c) Initialization

In complementary PWM mode, there are nine registers that must be initialized. In addition, there is a register that specifies whether to generate dead time (it should be used only when dead time generation should be disabled).

Before setting complementary PWM mode with bits MD[3:0] in the timer mode register (TMDR), the following initial register values must be set.

TGRC_3 operates as the buffer register for TGRA_3, and should be set with 1/2 the PWM carrier cycle + dead time Td. The timer cycle buffer register (TCBR) operates as the buffer register for the timer cycle data register (TCDR), and should be set with 1/2 the PWM carrier cycle. Set dead time Td in the timer dead time data register (TDDR).

When dead time is not needed, the TDER bit in the timer dead time enable register (TDER) should be cleared to 0, TGRC_3 and TGRA_3 should be set to 1/2 the PWM carrier cycle + 1, and TDDR should be set to 1.

Set the respective initial PWM duty values in buffer registers TGRD_3, TGRC_4, and TGRD_4.

The values set in the five buffer registers excluding TDDR are transferred simultaneously to the corresponding compare registers when complementary PWM mode is set.

Set TCNT_4 to H'0000 before setting complementary PWM mode.

Table 11.56 Registers and Counters Requiring Initialization

Register/Counter	Set Value
TGRC_3	1/2 PWM cycle + dead time Td (1/2 PWM cycle + 1 when dead time generation is disabled by TDER)
TDDR	Dead time Td (1 when dead time generation is disabled by TDER)
TCBR	1/2 PWM carrier cycle
TGRD_3, TGRC_4, TGRD_4	Initial PWM duty value for each phase
TCNT_4	H'0000
TOCR1, TOCR2	PWM output level setting

Note: The TGRC_3 set value must be the sum of 1/2 the PWM cycle set in TCBR and dead time Td set in TDDR. When dead time generation is disabled by TDER, TGRC_3 must be set to 1/2 the PWM cycle + 1.

(d) PWM Output Level Setting

In complementary PWM mode, the PWM pulse output level is set with bits OLSN and OLSP in timer output control register 1 (TOCR1) or bits OLS1P to OLS3P and OLS1N to OLS3N in timer output control register 2 (TOCR2).

The output level can be set for each of the three positive phases and three negative phases of 6-phase output.

Complementary PWM mode should be cleared before setting or changing output levels.

(e) Dead Time Setting

In complementary PWM mode, PWM pulses are output with a non-overlapping relationship between the positive and negative phases. This non-overlap time is called the dead time.

The non-overlap time is set in the timer dead time data register (TDDR). The value set in TDDR is used as the TCNT_3 counter start value, and creates non-overlap between TCNT_3 and TCNT_4. Complementary PWM mode should be cleared before changing the contents of TDDR.

(f) Dead Time Suppressing

Dead time generation is suppressed by clearing the TDER bit in the timer dead time enable register (TDER) to 0. TDER can be cleared to 0 only when 0 is written to it after reading TDER = 1.

TGRA_3 and TGRC_3 should be set to 1/2 PWM carrier cycle + 1 and the timer dead time data register (TDDR) should be set to 1.

By the above settings, PWM waveforms without dead time can be obtained. Figure 11.41 shows an example of operation without dead time.

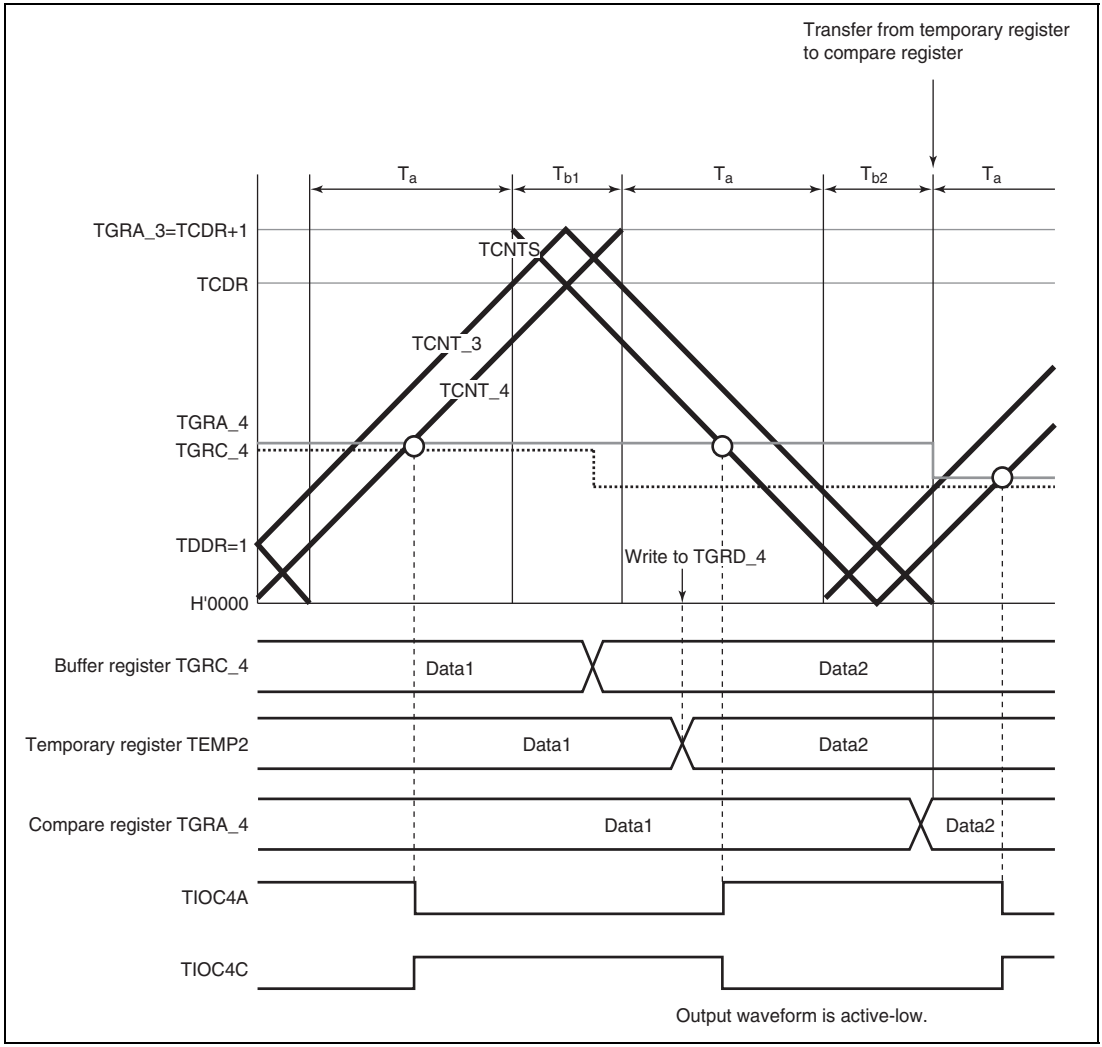


Figure 11.41 Example of Operation without Dead Time

(g) PWM Cycle Setting

In complementary PWM mode, the PWM pulse cycle is set in two registers—TGRA_3, in which the TCNT_3 upper limit value is set, and TCDR, in which the TCNT_4 upper limit value is set. The settings should be made so as to achieve the following relationship between these two registers:

With dead time: TGRA_3 set value = TCDR set value + TDDR set value
TCDR set value > Double the TDDR set value + 2
Without dead time: TGRA_3 set value = TCDR set value + 1
TCDR set value > 4

The TGRA_3 and TCDR settings are made by setting the values in buffer registers TGRC_3 and TCBR. When a write is performed to TGRD_4 and transfer is enabled, the values set in TGRC_3 and TCBR are transferred simultaneously to TGRA_3 and TCDR in accordance with the transfer timing selected by bits MD[3:0] in the timer mode register.

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 11.42 illustrates the operation when the PWM cycle is updated at the crest.

See the following section, Register Data Updating, for the method of updating the data in each buffer register.

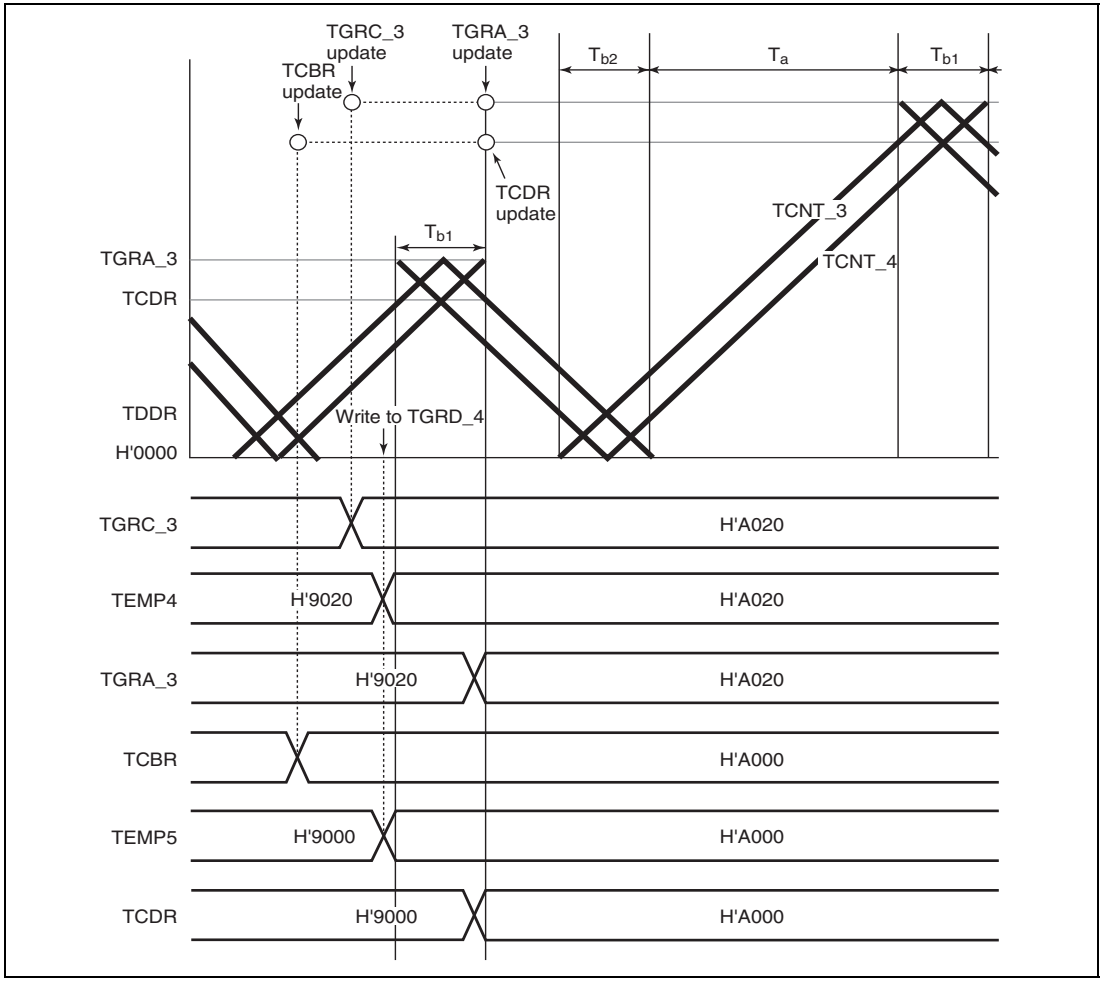


Figure 11.42 Example of PWM Cycle Updating

(h) Register Data Updating

In complementary PWM mode, the buffer register is used to update the data in a compare register and timer cycle register. The update data can be written to the buffer register at any time. There are five PWM duty and PWM cycle registers that have buffer registers and can be updated during operation.

There is a temporary register between each of these registers and its buffer register. When subcounter TCNTS is not counting, the temporary register value is also rewritten in the Ta interval when buffer register data is updated. Transfer is not performed from buffer registers to temporary registers in the Tb interval when TCNTS is counting; in this case, the value written to the buffer register is transferred after TCNTS halts.

The temporary register value is transferred to the compare register at the data update timing set with bits MD[3:0] in the timer mode register (TMDR). Figure 11.43 shows an example of data updating in complementary PWM mode. This example shows the mode in which data updating is performed at both the counter crest and trough.

When rewriting buffer register data, a write to TGRD_4 must be performed at the end of the update. Data transfer from the buffer registers to the temporary registers is performed simultaneously for all five registers after the write to TGRD_4.

A write to TGRD_4 must be performed after writing data to the registers to be updated, even when not updating all five registers, or when updating the TGRD_4 data. In this case, the data written to TGRD_4 should be the same as the data prior to the write operation.

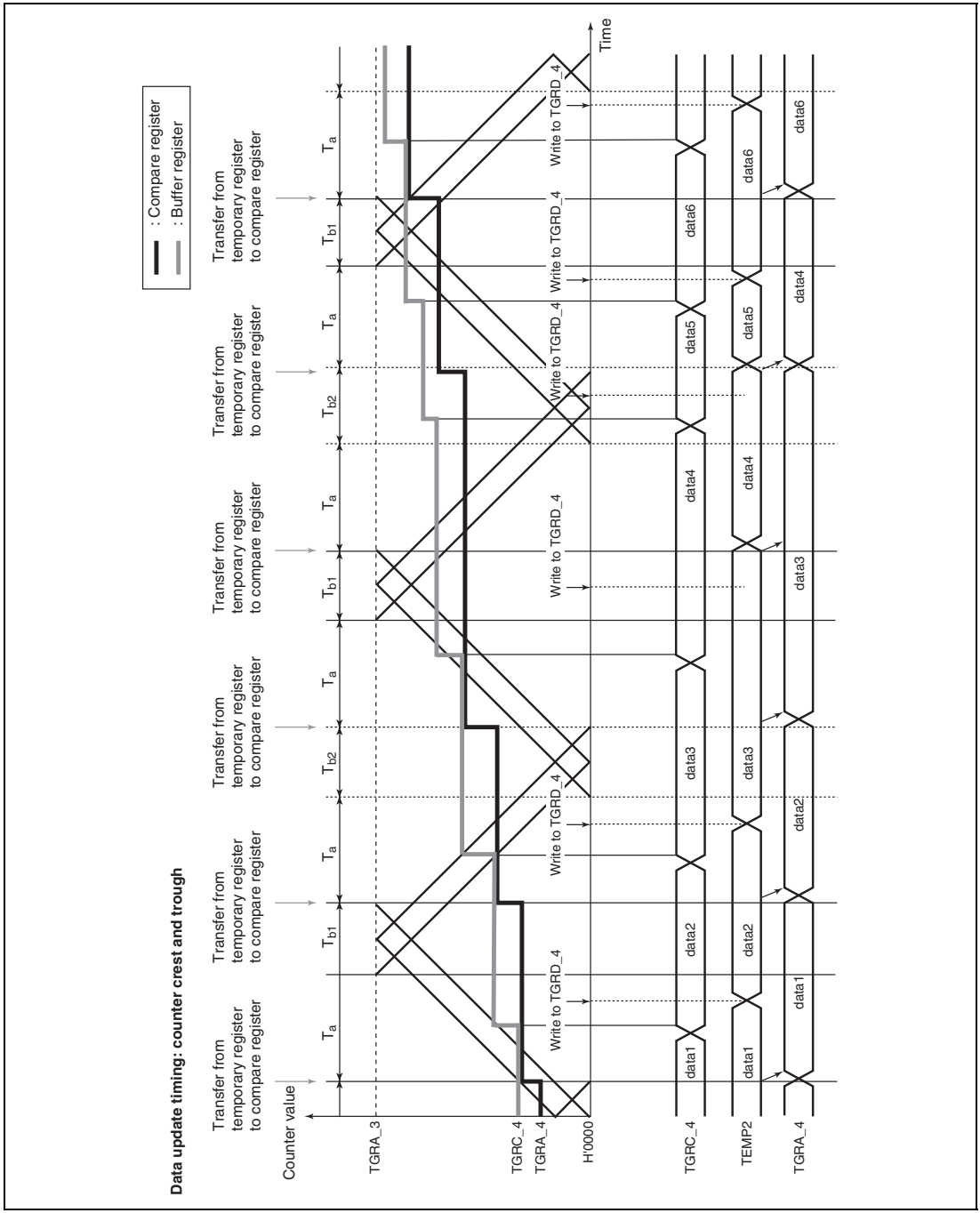


Figure 11.43 Example of Data Update in Complementary PWM Mode

(i) Initial Output in Complementary PWM Mode

In complementary PWM mode, the initial output is determined by the setting of bits OLSN and OLSP in timer output control register 1 (TOCR1) or bits OLS1N to OLS3N and OLS1P to OLS3P in timer output control register 2 (TOCR2).

This initial output is the PWM pulse non-active level, and is output from when complementary PWM mode is set with the timer mode register (TMDR) until TCNT_4 exceeds the value set in the dead time register (TDDR). Figure 11.44 shows an example of the initial output in complementary PWM mode.

An example of the waveform when the initial PWM duty value is smaller than the TDDR value is shown in figure 11.45.

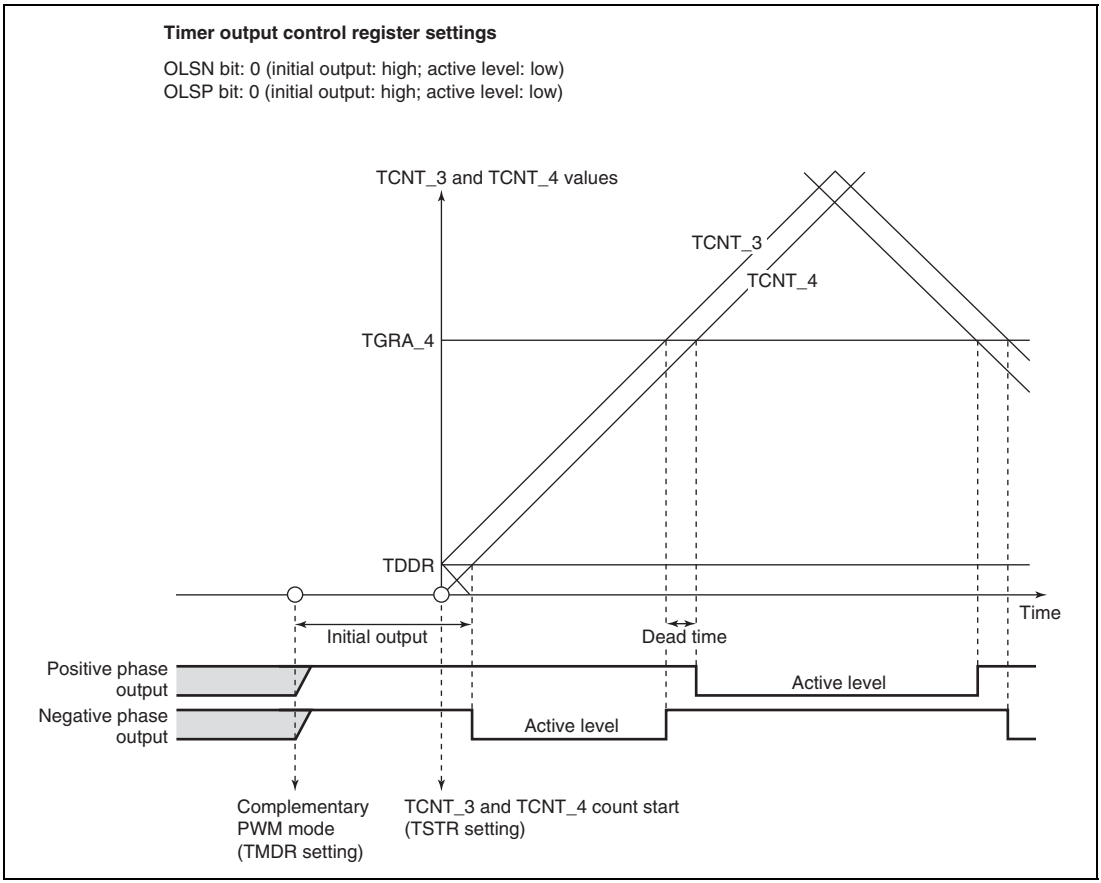


Figure 11.44 Example of Initial Output in Complementary PWM Mode (1)

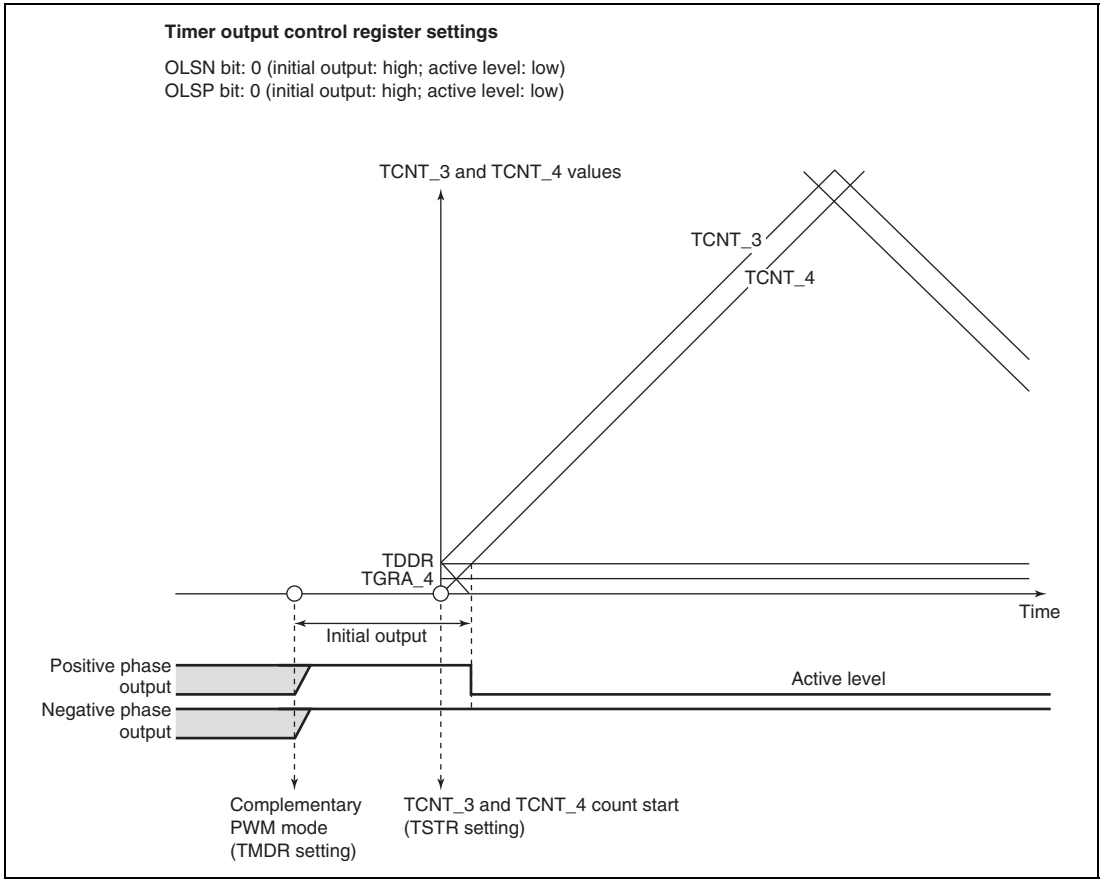


Figure 11.45 Example of Initial Output in Complementary PWM Mode (2)

(j) Complementary PWM Mode PWM Output Generation Method

In complementary PWM mode, 3-phase output is performed of PWM waveforms with a non-overlap time between the positive and negative phases. This non-overlap time is called the dead time.

A PWM waveform is generated by output of the output level selected in the timer output control register in the event of a compare-match between a counter and compare register. While TCNTS is counting, compare register and temporary register values are simultaneously compared to create consecutive PWM pulses from 0 to 100%. The relative timing of on and off compare-match occurrence may vary, but the compare-match that turns off each phase takes precedence to secure the dead time and ensure that the positive phase and negative phase on times do not overlap. Figures 11.46 to 11.48 show examples of waveform generation in complementary PWM mode.

The positive phase/negative phase off timing is generated by a compare-match with the solid-line counter, and the on timing by a compare-match with the dotted-line counter operating with a delay of the dead time behind the solid-line counter. In the T1 period, compare-match **a** that turns off the negative phase has the highest priority, and compare-matches occurring prior to **a** are ignored. In the T2 period, compare-match **c** that turns off the positive phase has the highest priority, and compare-matches occurring prior to **c** are ignored.

In normal cases, compare-matches occur in the order **a** → **b** → **c** → **d** (or **c** → **d** → **a'** → **b'**), as shown in figure 11.46.

If compare-matches deviate from the **a** → **b** → **c** → **d** order, since the time for which the negative phase is off is less than twice the dead time, the figure shows the positive phase is not being turned on. If compare-matches deviate from the **c** → **d** → **a'** → **b'** order, since the time for which the positive phase is off is less than twice the dead time, the figure shows the negative phase is not being turned on.

If compare-match **c** occurs first following compare-match **a**, as shown in figure 11.47, compare-match **b** is ignored, and the negative phase is turned on 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 11.48, compare-match **a'** with the new data in the temporary register occurs before compare-match **c**, but other compare-matches occurring up to **c**, which turns off the positive phase, are ignored. As a result, the negative phase is not turned on.

Thus, in complementary PWM mode, compare-matches at turn-off timings take precedence, and turn-on timing compare-matches that occur before a turn-off timing compare-match are ignored.

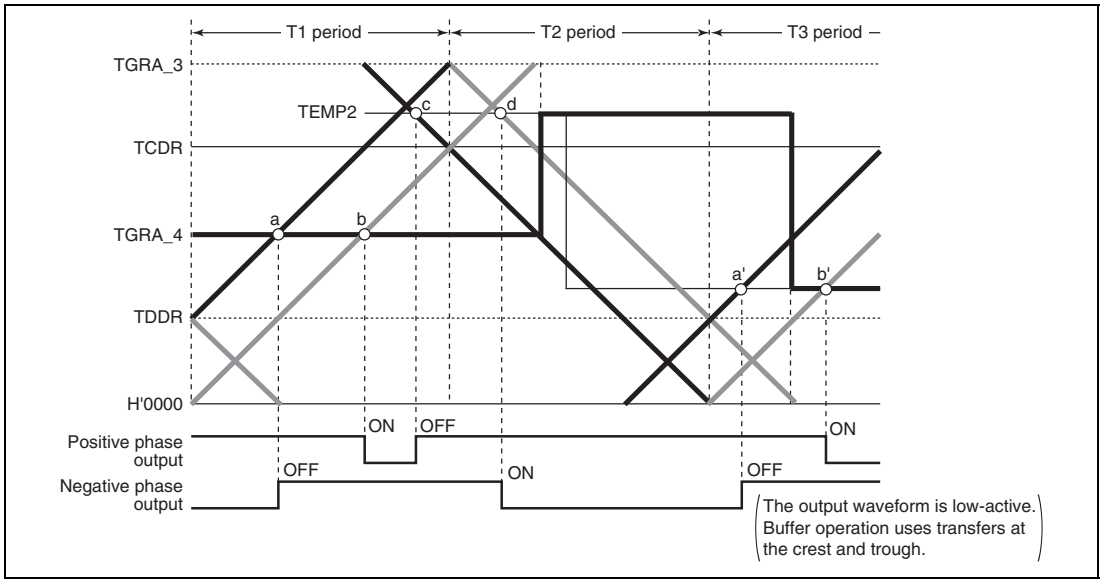


Figure 11.46 Example of Complementary PWM Mode Waveform Output (1)

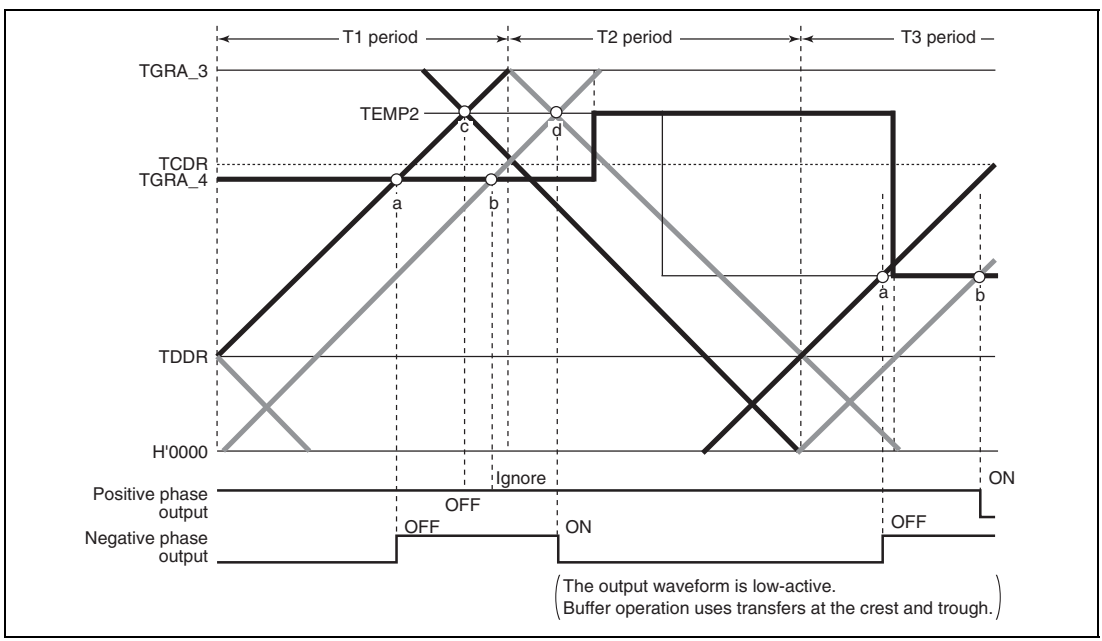


Figure 11.47 Example of Complementary PWM Mode Waveform Output (2)

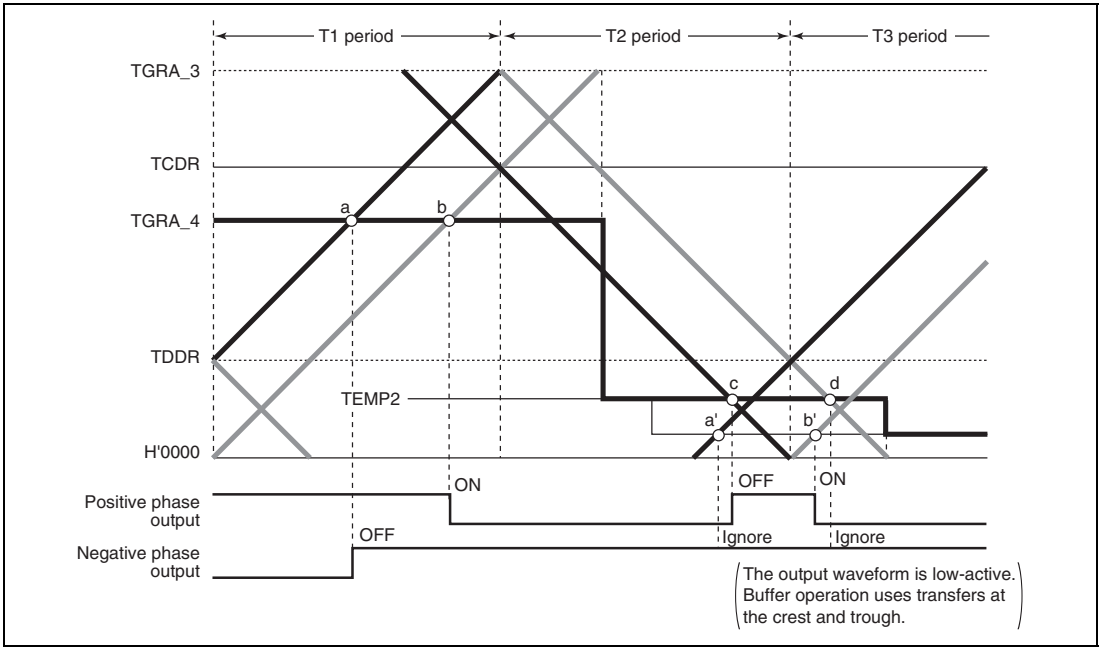


Figure 11.48 Example of Complementary PWM Mode Waveform Output (3)

(k) Complementary PWM Mode 0% and 100% Duty Output

In complementary PWM mode, 0% and 100% duty cycles can be output as required. Figures 11.49 to 11.53 show output examples.

100% duty output is performed when the compare 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 compare register value is set to the same value as TGRA_3. The waveform in this case has a positive phase with a 100% off-state.

On and off compare-matches occur simultaneously, but if a turn-on compare-match and turn-off compare-match for the same phase occur simultaneously, both compare-matches are ignored and the waveform does not change.

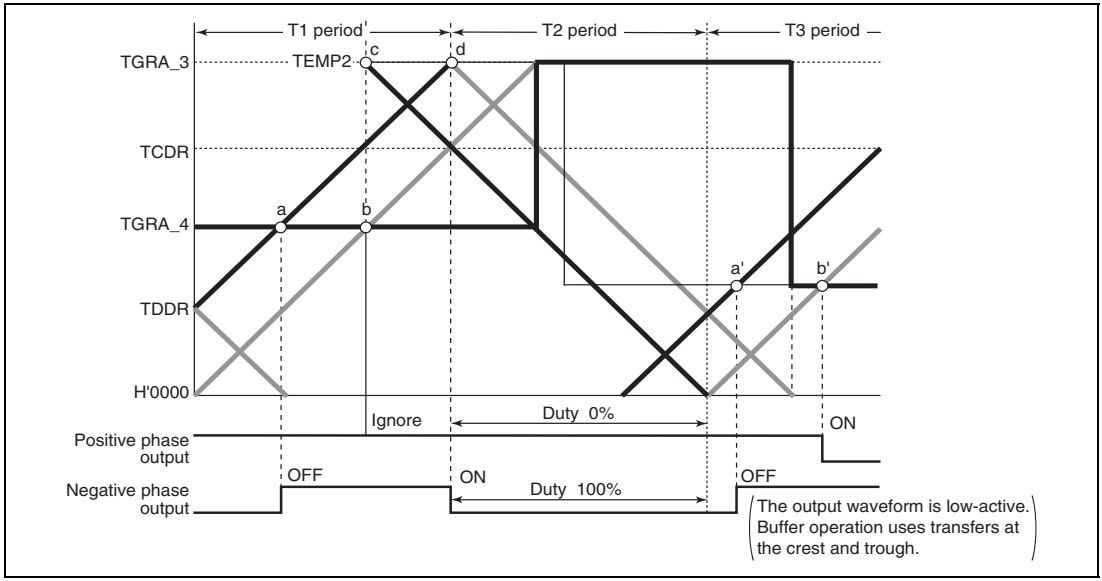


Figure 11.49 Example of Complementary PWM Mode 0% and 100% Waveform Output (1)

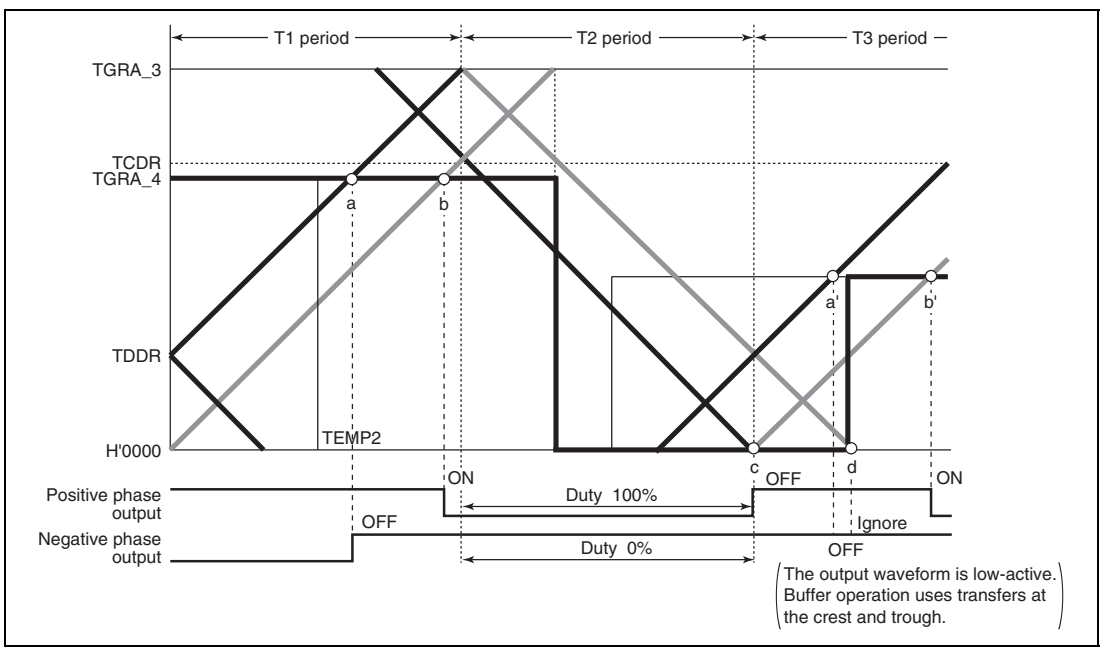


Figure 11.50 Example of Complementary PWM Mode 0% and 100% Waveform Output (2)

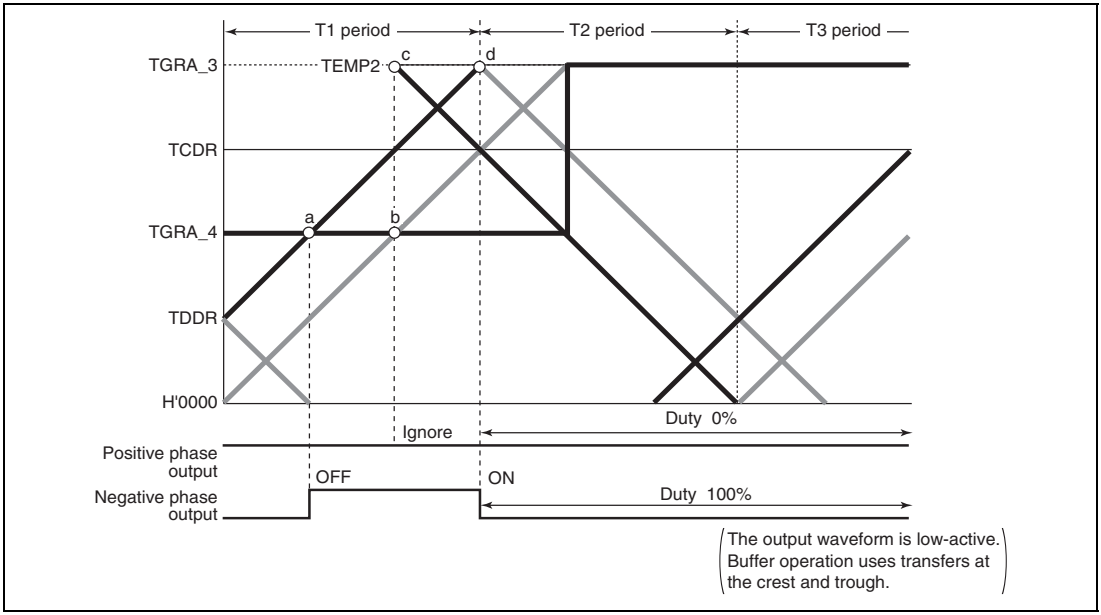


Figure 11.51 Example of Complementary PWM Mode 0% and 100% Waveform Output (3)

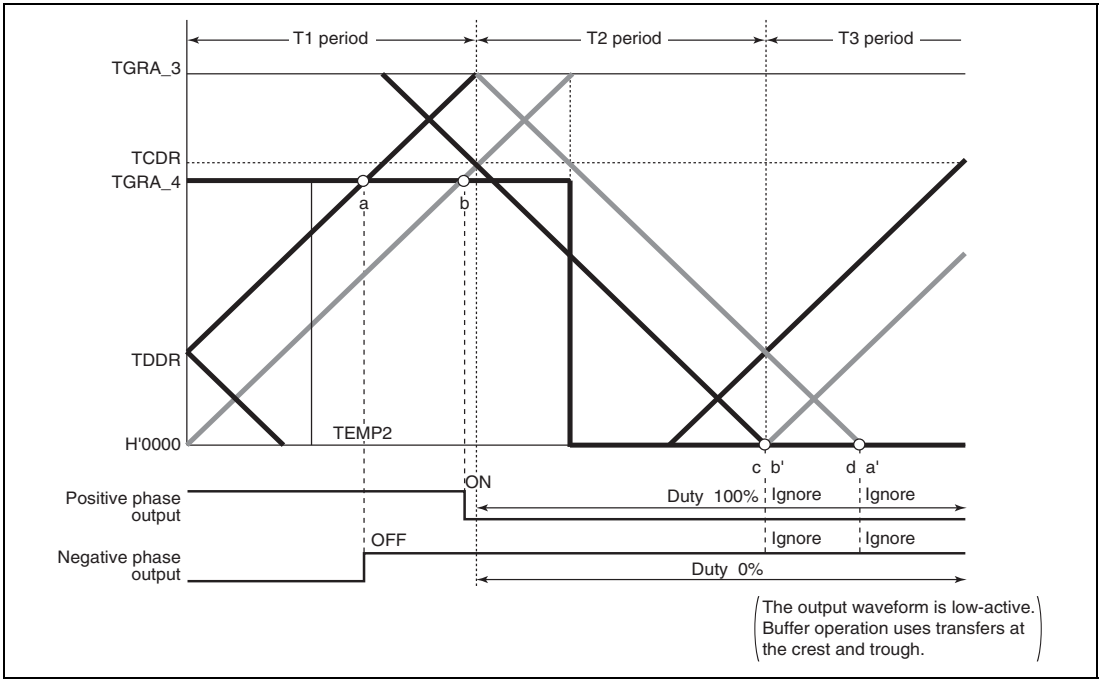


Figure 11.52 Example of Complementary PWM Mode 0% and 100% Waveform Output (4)

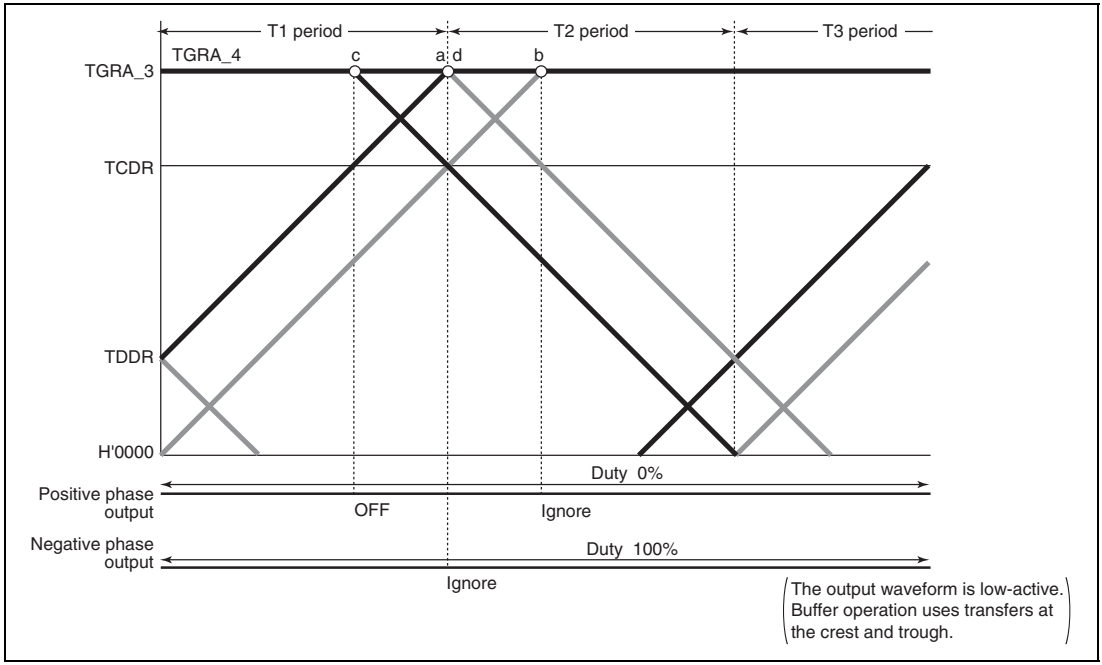


Figure 11.53 Example of Complementary PWM Mode 0% and 100% Waveform Output (5)

(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 the timer output control register (TOCR1). An example of a toggle output waveform is shown in figure 11.54.

This output is toggled by a compare-match between TCNT_3 and TGRA_3 and a compare-match between TCNT4 and H'0000.

The output pin for this toggle output is the TIOC3A pin. The initial output is high.

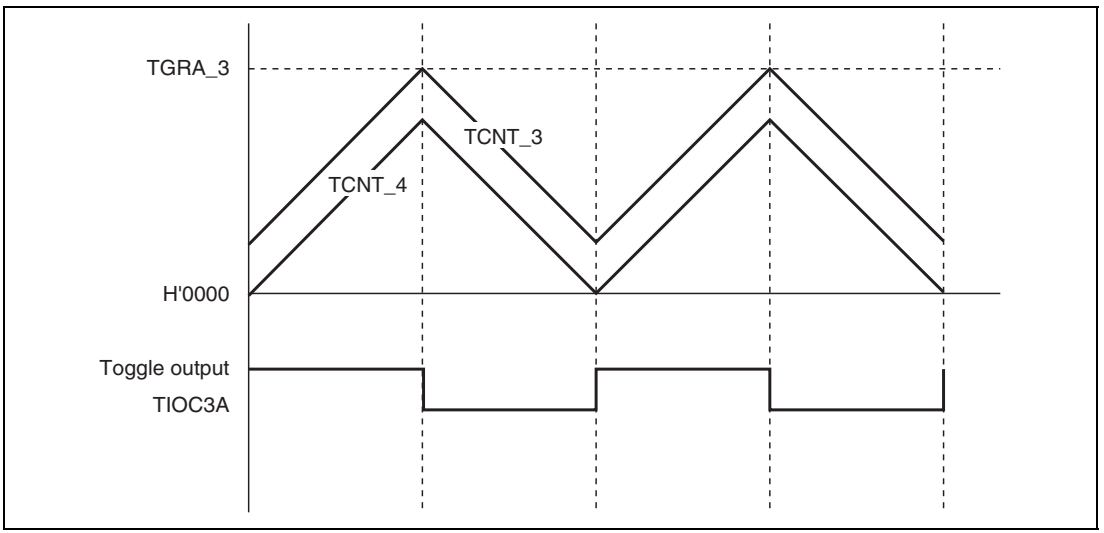


Figure 11.54 Example of Toggle Output Waveform Synchronized with PWM Output

(m) Counter Clearing by Another Channel

In complementary PWM mode, by setting a mode for synchronization with another channel by means of the timer synchronous register (TSYR), and selecting synchronous clearing with bits CCLR[2:0] in the timer control register (TCR), it is possible to have TCNT_3, TCNT_4, and TCNTS cleared by another channel.

Figure 11.55 illustrates the operation.

Use of this function enables counter clearing and restarting to be performed by means of an external signal.

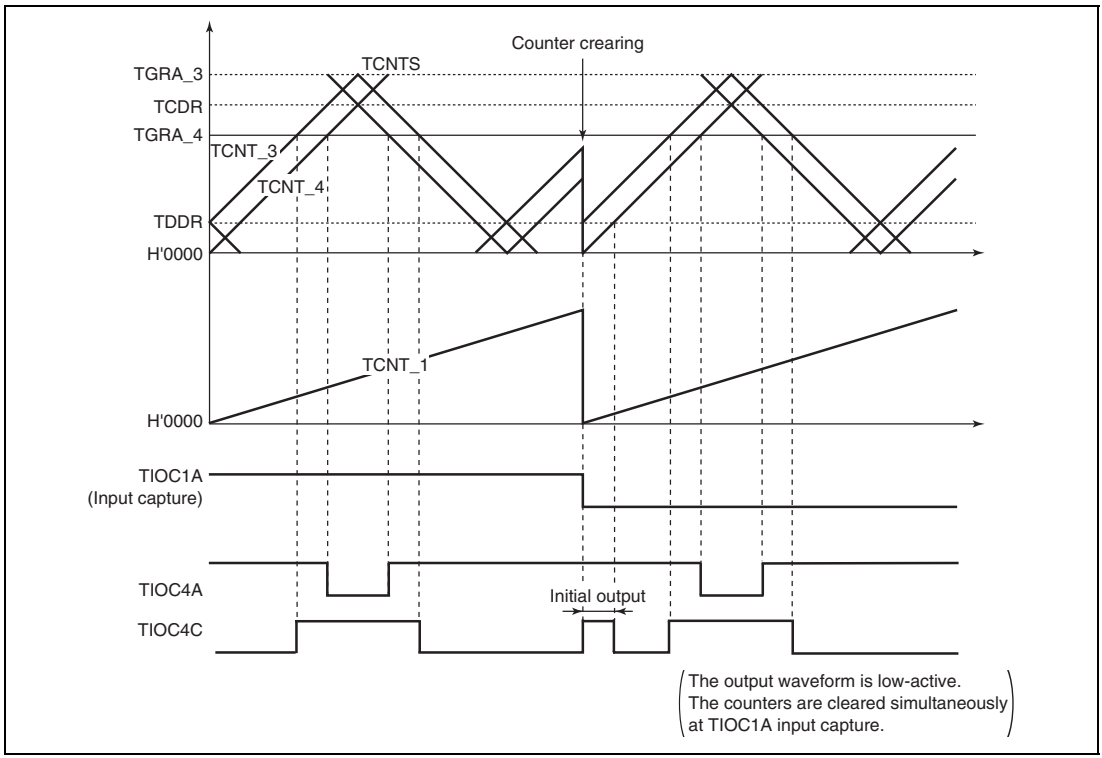


Figure 11.55 Counter Clearing Synchronized with Another Channel

(n) Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Setting the WRE bit in TWCR to 1 suppresses initial output when synchronous counter clearing occurs in the Tb2 interval 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 Tb2 interval as indicated by (10) or (11) in figure 11.56. When synchronous clearing occurs outside that interval, the initial value specified by the OLS bits in TOCR is output. Even in the Tb2 interval, if synchronous clearing occurs in the initial value output period (indicated by (1) in figure 11.56) immediately after the counters start operation, initial value output is not suppressed.

When using the initial output suppression function, make sure to set compare registers TGRB_3, TGRA_4, and TGRB_4 to a value twice or more the setting of dead time data register TDDR. If synchronous clearing occurs with the compare registers set to a value less than twice the setting of TDDR, the PWM output dead time may be too short (or nonexistent) or illegal active-level PWM negative-phase output may occur during the initial output suppression interval. For details, see section 11.7.23, Notes on Output Waveform Control During Synchronous Counter Clearing in Complementary PWM Mode.

This function can be used in both the MTU2 and MTU2S. In the MTU2, synchronous clearing generated in channels 0 to 2 in the MTU2 can cause counter clearing in complementary PWM mode; in the MTU2S, compare match or input capture flag setting in channels 0 to 2 in the MTU2 can cause counter clearing.

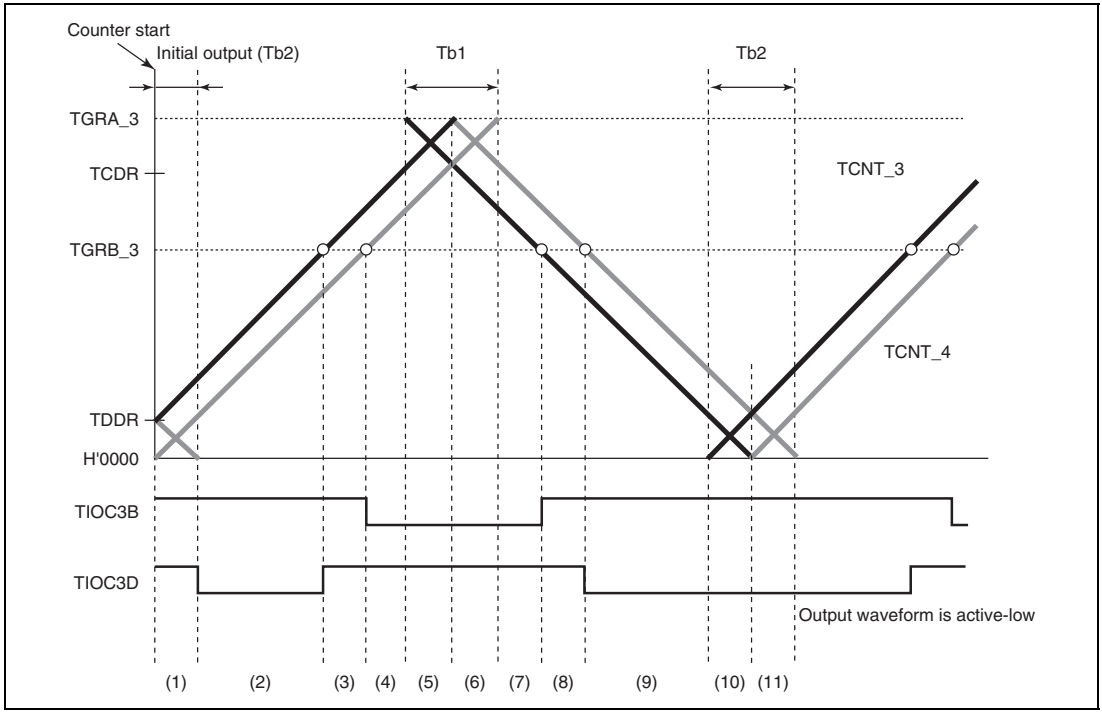


Figure 11.56 Timing for Synchronous Counter Clearing

- Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

  An example of the procedure for setting output waveform control at synchronous counter clearing in complementary PWM mode is shown in figure 11.57.

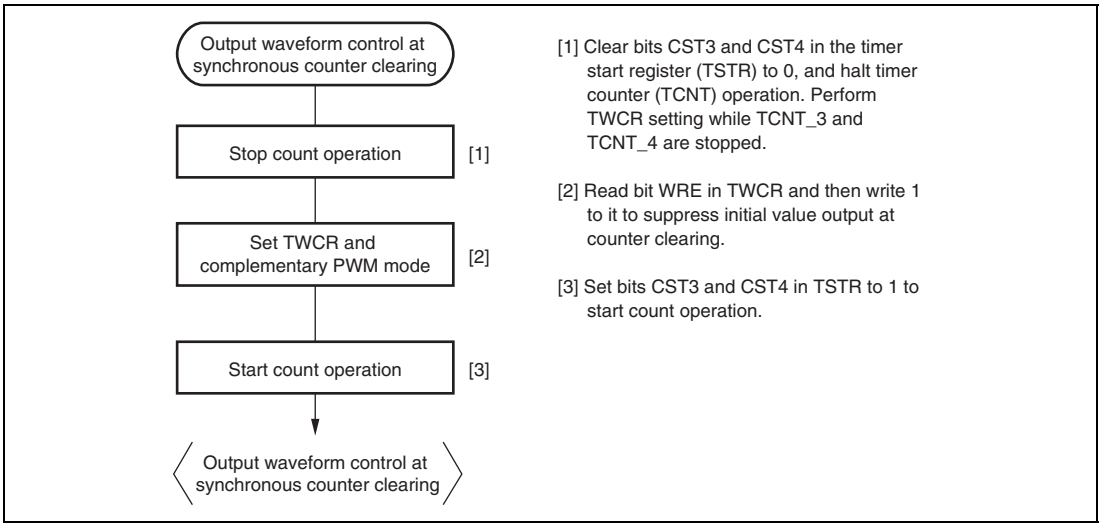


Figure 11.57 Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

- Examples of Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

  Figures 11.58 to 11.61 show examples of output waveform control in which the MTU2 operates in complementary PWM mode and synchronous counter clearing is generated while the WRE bit in TWCR is set to 1. In the examples shown in figures 11.58 to 11.61, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in figure 11.56, respectively.

  In the MTU2S, these examples are equivalent to the cases when the MTU2S operates 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 TWCRS.

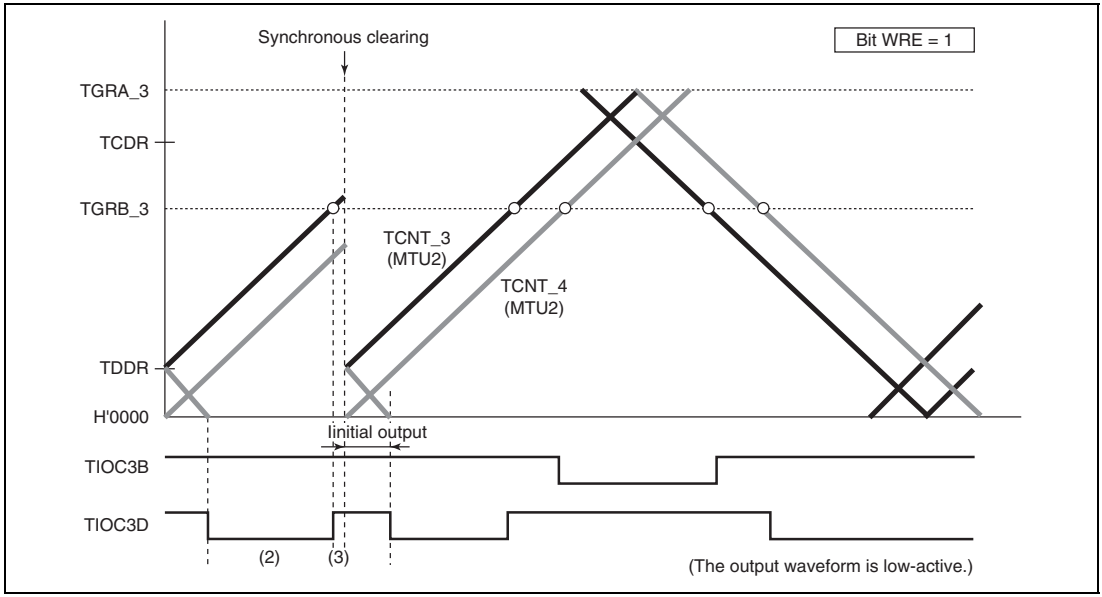


Figure 11.58 Example of Synchronous Clearing in Dead Time during Up-Counting (Timing (3) in Figure 11.56; Bit WRE of TWCR in MTU2 Is 1)

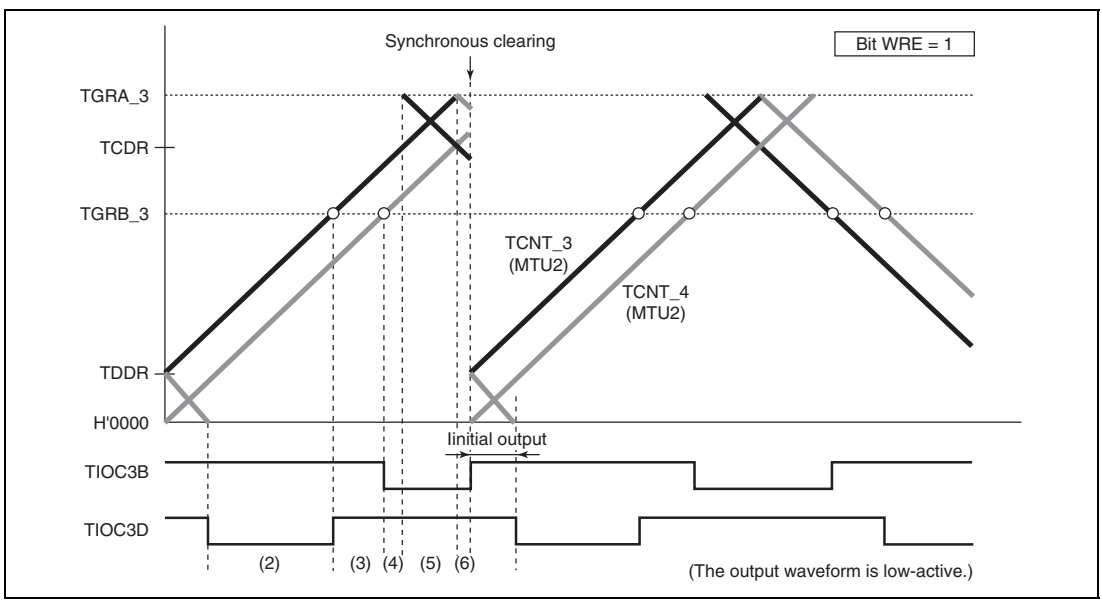


Figure 11.59 Example of Synchronous Clearing in Interval Tb1 (Timing (6) in Figure 11.56; Bit WRE of TWCR in MTU2 Is 1)

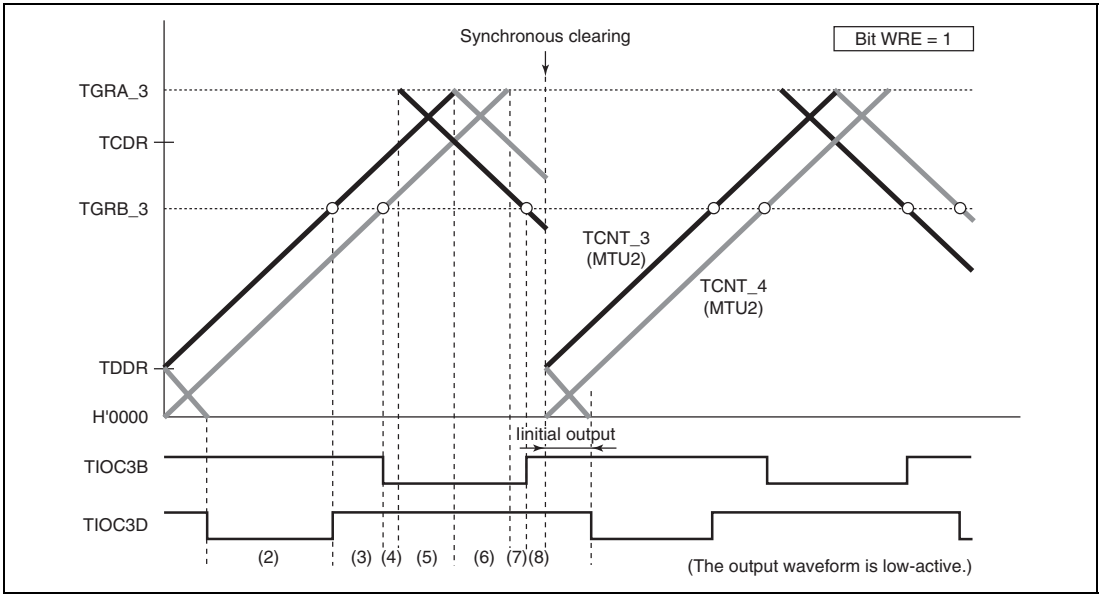


Figure 11.60 Example of Synchronous Clearing in Dead Time during Down-Counting (Timing (8) in Figure 11.56; Bit WRE of TWCR Is 1)

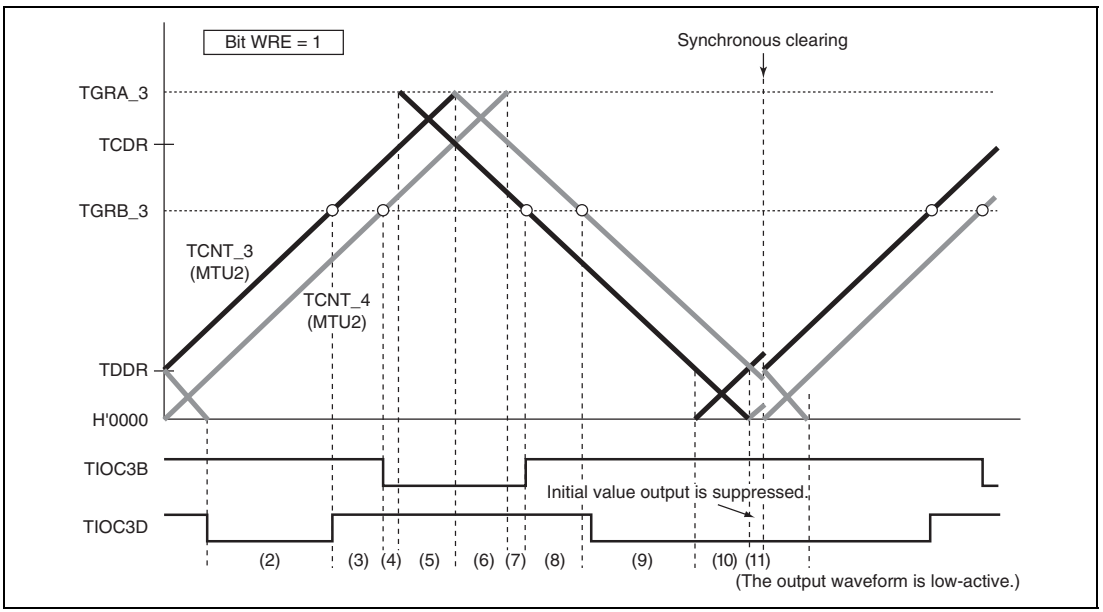


Figure 11.61 Example of Synchronous Clearing in Interval Tb2 (Timing (11) in Figure 11.56; Bit WRE of TWCR Is 1)

(o) Suppressing MTU2–MTU2S Synchronous Counter Clearing

In the MTU2S, setting the SCC bit in TWCRS to 1 suppresses synchronous counter clearing caused by the MTU2.

Synchronous counter clearing is suppressed only within the interval shown in figure 11.62. When using this function, the MTU2S should be set to complementary PWM mode.

For details of synchronous clearing caused by the MTU2, refer to the description about MTU2S counter clearing caused by MTU2 flag setting source (MTU2-MTU2S synchronous counter clearing) in section 11.4.10, MTU2–MTU2S Synchronous Operation.

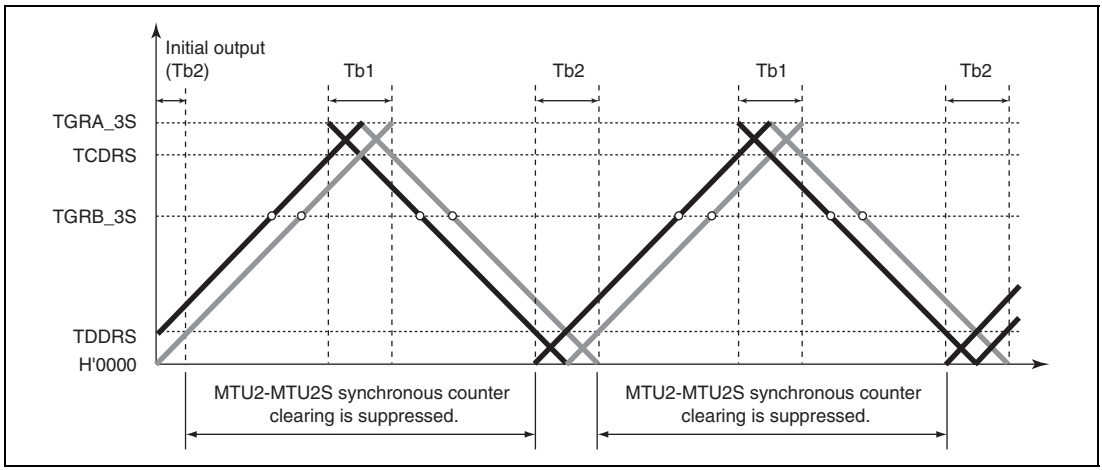


Figure 11.62 MTU2–MTU2S Synchronous Clearing-Suppressed Interval Specified by SCC Bit in TWCRS

- Example of Procedure for Suppressing MTU2–MTU2S Synchronous Counter Clearing

  An example of the procedure for suppressing MTU2–MTU2S synchronous counter clearing is shown in figure 11.63.

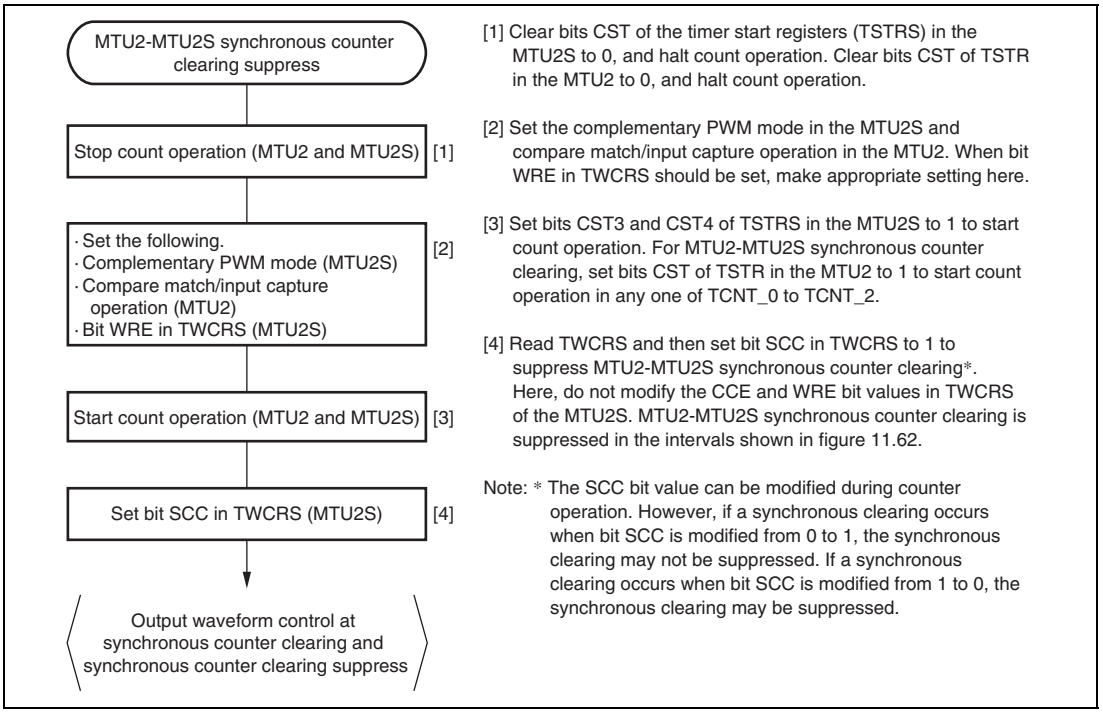


Figure 11.63 Example of Procedure for Suppressing MTU2–MTU2S Synchronous Counter Clearing

- Examples of Suppression of MTU2–MTU2S Synchronous Counter Clearing

  Figures 11.64 to 11.67 show examples of operation in which the MTU2S operates in complementary PWM mode and MTU2–MTU2S synchronous counter clearing is suppressed by setting the SCC bit in TWCRS in the MTU2S to 1. In the examples shown in figures 11.64 to 11.67, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in figure 11.56, respectively.

  In these examples, the WRE bit in TWCRS of the MTU2S is set to 1.

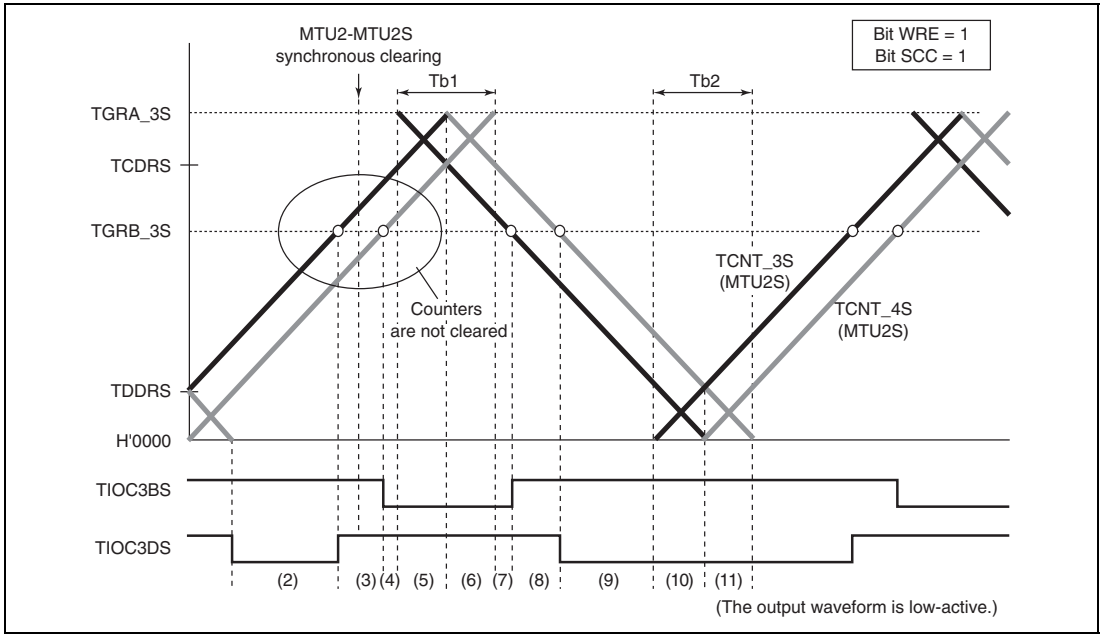


Figure 11.64 Example of Synchronous Clearing in Dead Time during Up-Counting (Timing (3) in Figure 11.56; Bit WRE Is 1 and Bit SCC Is 1 in TWCRS of MTU2S)

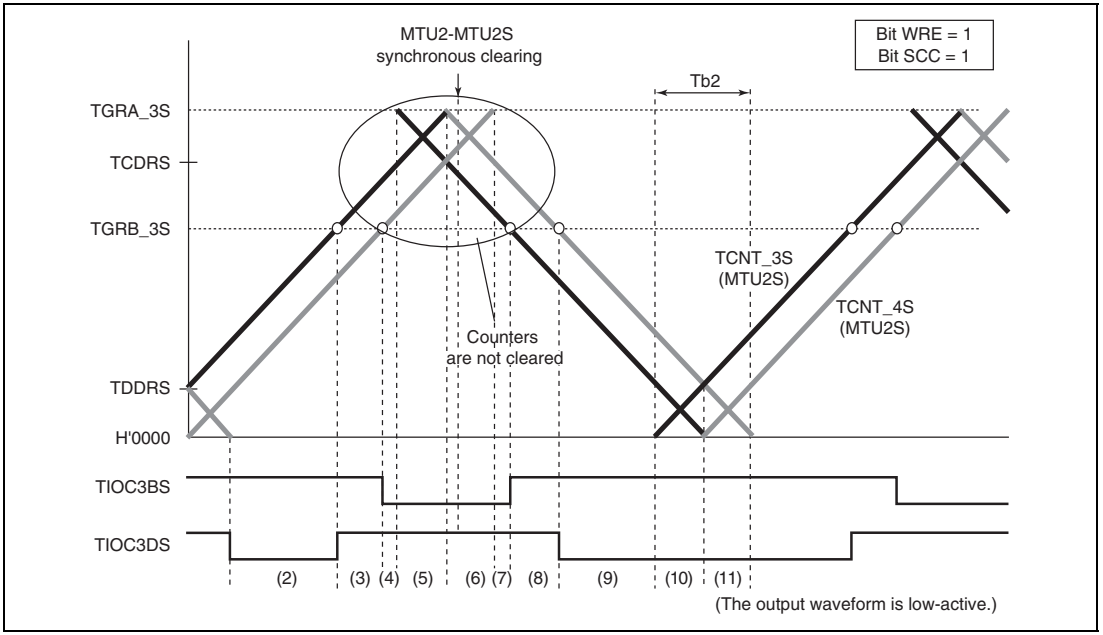


Figure 11.65 Example of Synchronous Clearing in Interval Tb1
(Timing (6) in Figure 11.56; Bit WRE Is 1 and Bit SCC Is 1 in TWCRS of MTU2S)

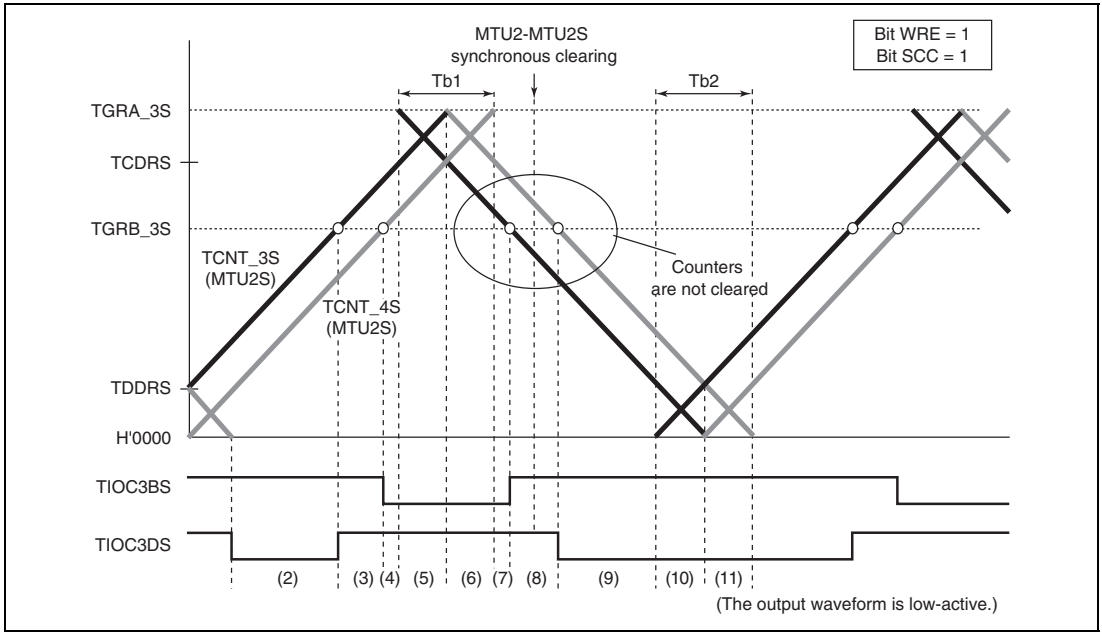


Figure 11.66 Example of Synchronous Clearing in Dead Time during Down-Counting (Timing (8) in Figure 11.56; Bit WRE Is 1 and Bit SCC Is 1 in TWCRS of MTU2S)

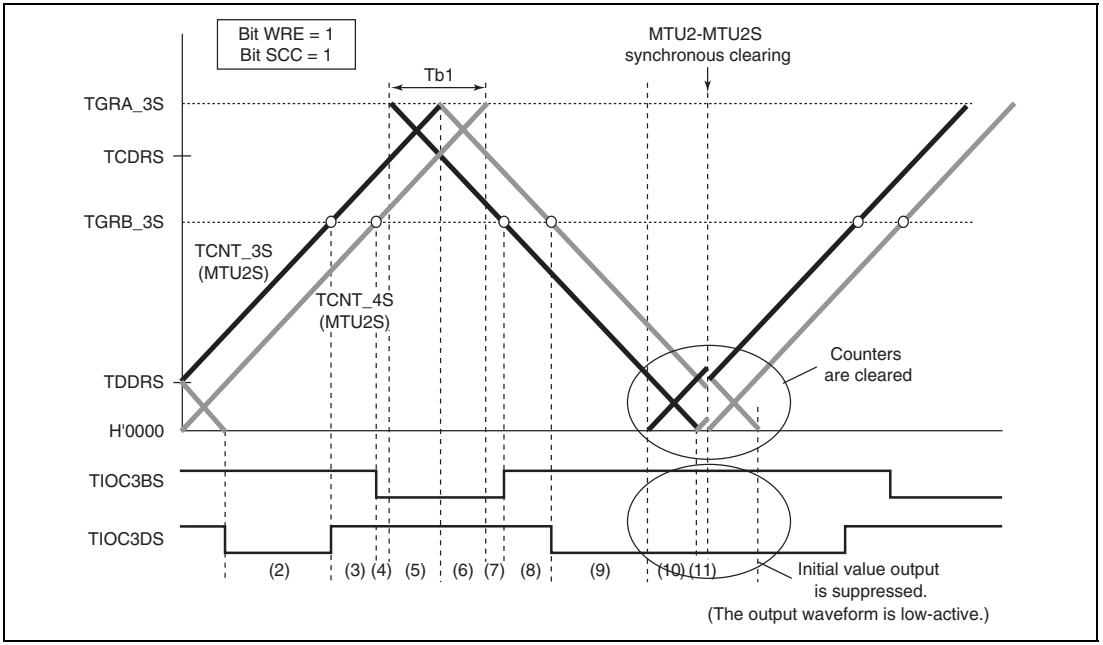


Figure 11.67 Example of Synchronous Clearing in Interval Tb2
(Timing (11) in Figure 11.56; Bit WRE Is 1 and Bit SCC Is 1 in TWCRS of MTU2S)

(p) Counter Clearing by TGRA_3 Compare Match

In complementary PWM mode, by setting the CCE bit in the timer waveform control register (TWCR), it is possible to have TCNT_3, TCNT_4, and TCNTS cleared by TGRA_3 compare match.

Figure 11.68 illustrates an operation example.

Notes: 1. Use this function only in complementary PWM mode 1 (transfer at crest)
2. Do not specify synchronous clearing by another channel (do not set the SYNC0 to SYNC4 bits in the timer synchronous register (TSYR) to 1 or the CE0A, CE0B, CE0C, CE0D, CE1A, CE1B, CE1C, and CE1D bits in the timer synchronous clear register (TSYCR) to 1).
3. Do not set the compare registers (TGRB_3, TGRA_4, TGRB_4) value to H'0000.
4. Do not set the PSYE bit in timer output control register 1 (TOCR1) to 1.

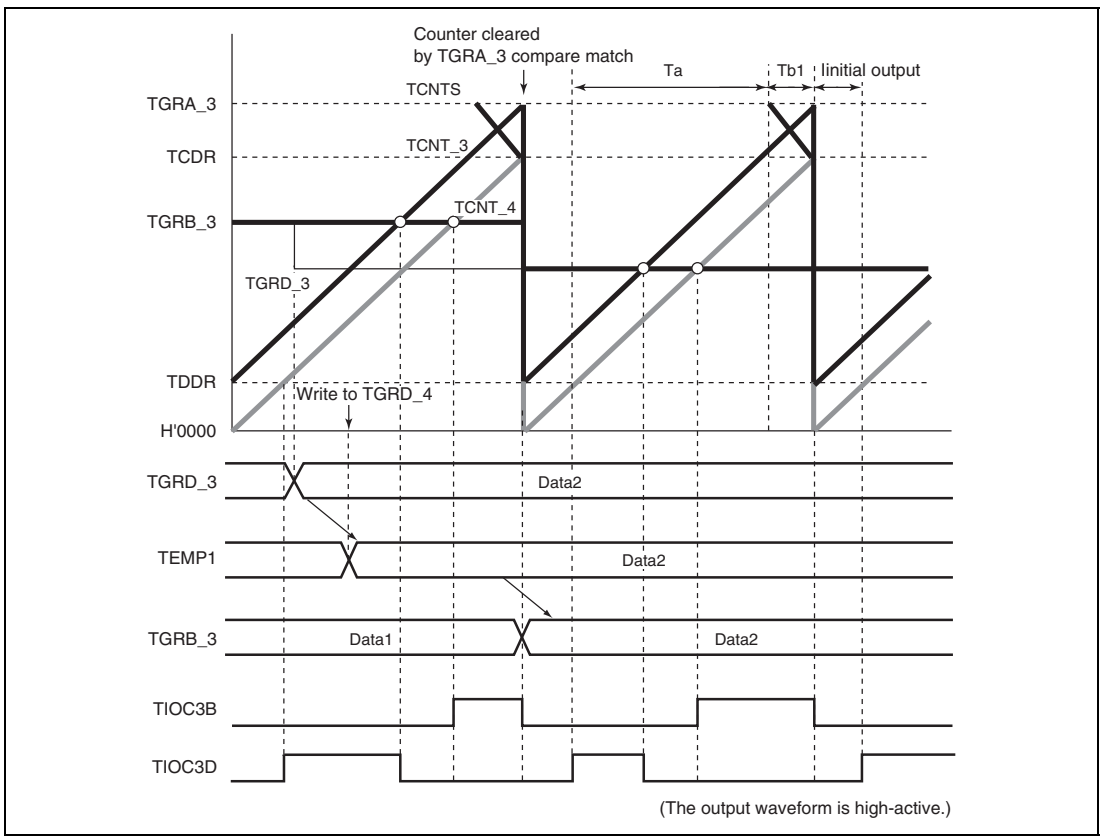


Figure 11.68 Example of Counter Clearing Operation by TGRA_3 Compare Match

(q) AC Synchronous Motor (Brushless DC Motor) Drive Waveform Output

In complementary PWM mode, a brushless DC motor can easily be controlled using the timer gate control register (TGCR). Figures 11.69 to 11.72 show examples of brushless DC motor drive waveforms created using TGCR.

To perform output phase switching for a 3-phase brushless DC motor by means of external signals detected with a Hall element, etc., clear the FB bit in TGCR to 0 and input the external signals indicating the polarity position to channel 0 timer input pins TIOC0A, TIOC0B, and TIOC0C (clear to 0 the corresponding IOR bits in the PFC's I/O register). The output is switched on and off automatically according to the signals input on pins TIOC0A, TIOC0B, and TIOC0C.

When the FB bit is set to 1, output is switched on and off according to the settings of the UF, VF, and WF bits in TGCR (table 11.39).

The drive waveforms are output from the complementary PWM mode 6-phase output pins.

With this 6-phase output, in the case of on output, it is possible to use complementary PWM mode output and perform chopping output by setting the N bit or P bit to 1. When the N bit or P bit is 0, level output is selected.

The 6-phase output active level (on output level) can be set with the OLSN and OLSP bits in the timer output control register (TOCR1, TOCR2) regardless of the setting of the N and P bits.

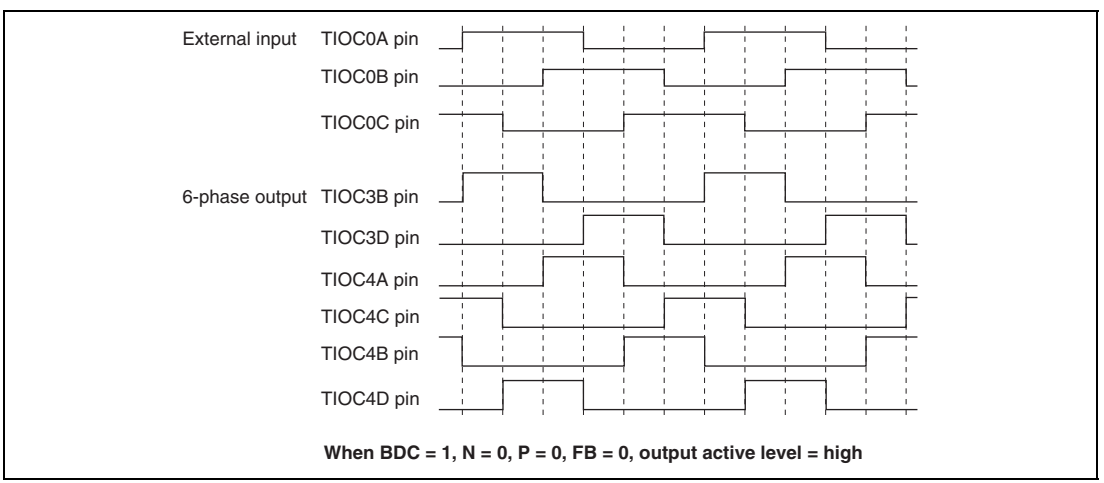


Figure 11.69 Example of Output Phase Switching by External Input (1)

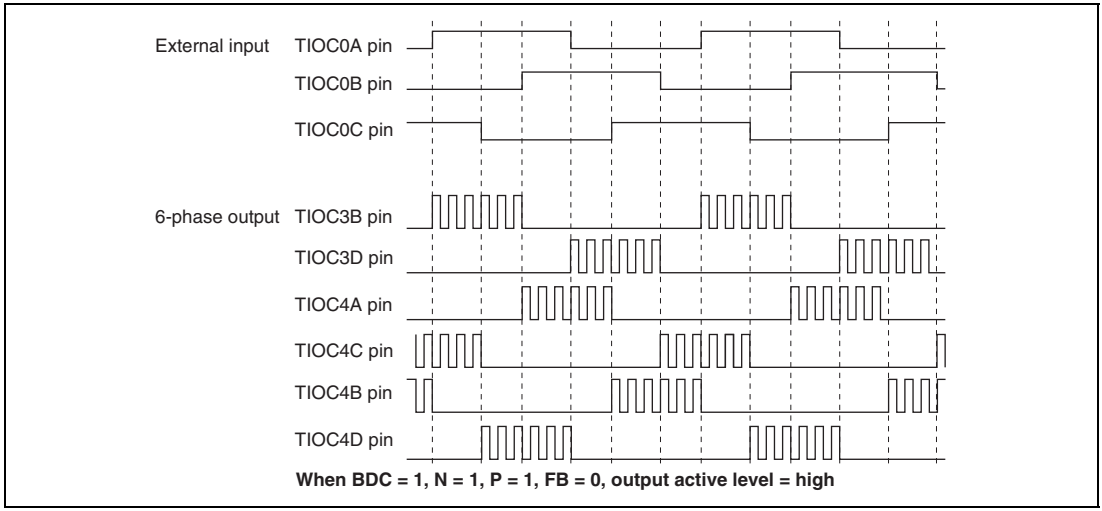


Figure 11.70 Example of Output Phase Switching by External Input (2)

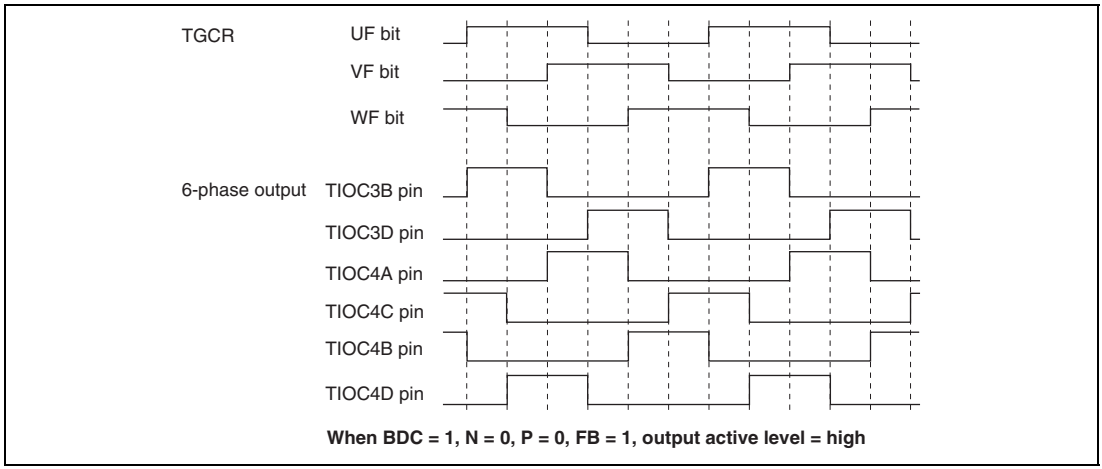


Figure 11.71 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (1)

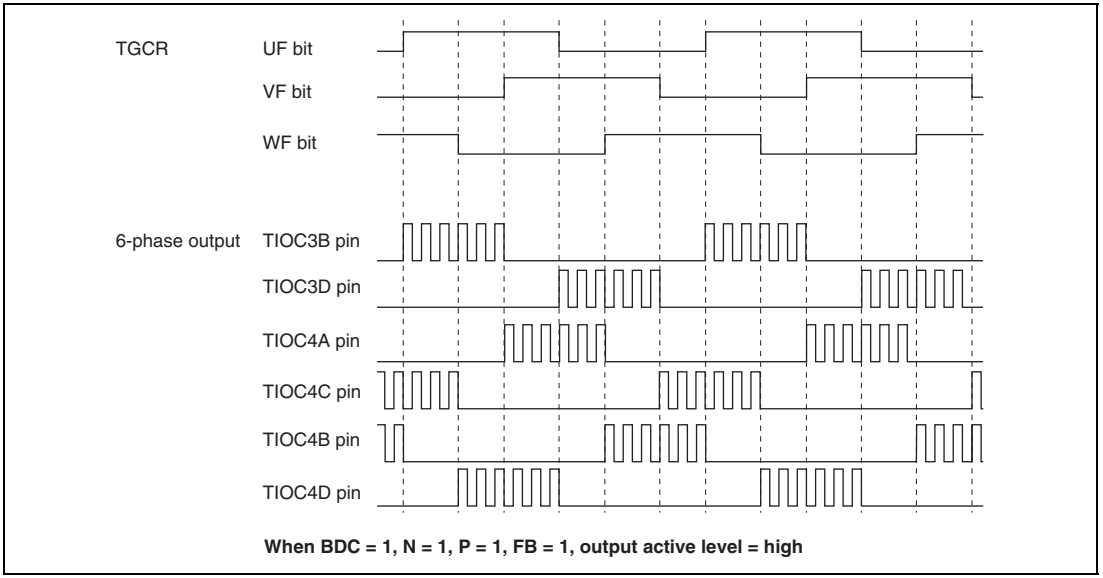


Figure 11.72 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (2)

(r) A/D Converter Start Request Setting

In complementary PWM mode, an A/D converter start request can be issued using a TGRA_3 compare-match, or TCNT_4 underflow (trough).

When start requests using a TGRA_3 compare-match are specified, A/D conversion can be started at the crest of the TCNT_3 count.

A/D converter start requests can be set by setting the TTGE bit to 1 in the timer interrupt enable register (TIER). To issue an A/D converter start request at a TCNT_4 underflow (trough), set the TTGE2 bit in TIER_4 to 1.

For information on the A/D converter start request delaying function, see 11.4.9, A/D Converter Start Request Delaying Function.

(3) Interrupt Skipping in Complementary PWM Mode

Interrupts TGIA_3 (at the crest) and TCIV_4 (at the trough) in channels 3 and 4 can be skipped up to seven times by making settings in the timer interrupt skipping set register (TITCR).

Transfers from a buffer register to a temporary register or a compare register can be skipped in coordination with interrupt skipping by making settings in the timer buffer transfer register (TBTER). For the linkage with buffer registers, refer to description 3, Buffer Transfer Control Linked with Interrupt Skipping, below.

A/D converter start requests generated by the A/D converter start request delaying function can also be skipped in coordination with interrupt skipping by making settings in the timer A/D converter request control register (TADCR). For the linkage with the A/D converter start request delaying function, refer to section 11.4.9, A/D Converter Start Request Delaying Function.

The setting of the timer interrupt skipping setting register (TITCR) must be done while the TGIA_3 and TCIV_4 interrupt requests are disabled by the settings of registers TIER_3 and TIER_4 along with under the conditions in which TGFA_3 and TCFV_4 flag settings by compare match never occur. Before changing the skipping count, be sure to clear the T3AEN and T4VEN bits to 0 to clear the skipping counter.

(a) Example of Interrupt Skipping Operation Setting Procedure

Figure 11.73 shows an example of the interrupt skipping operation setting procedure. Figure 11.74 shows the periods during which interrupt skipping count can be changed.

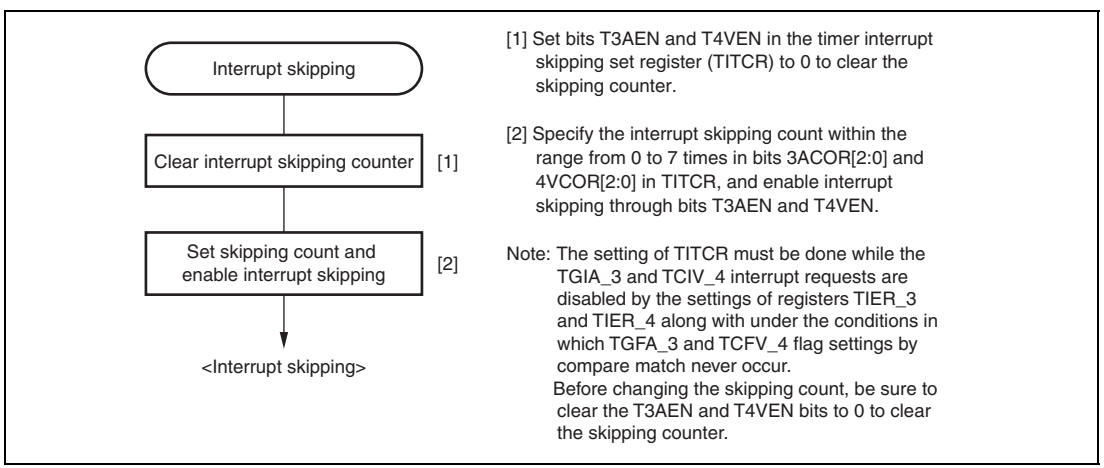


Figure 11.73 Example of Interrupt Skipping Operation Setting Procedure

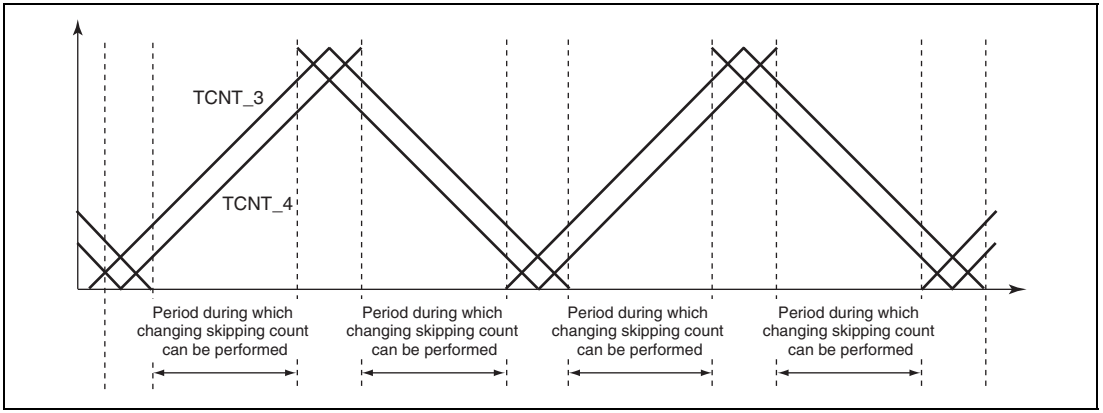


Figure 11.74 Periods during which Interrupt Skipping Count Can Be Changed

(b) Example of Interrupt Skipping Operation

Figure 11.75 shows an example of TGIA_3 interrupt skipping in which the interrupt skipping count is set to three by the 3ACOR bit and the T3AEN bit is set to 1 in the timer interrupt skipping set register (TITCR).

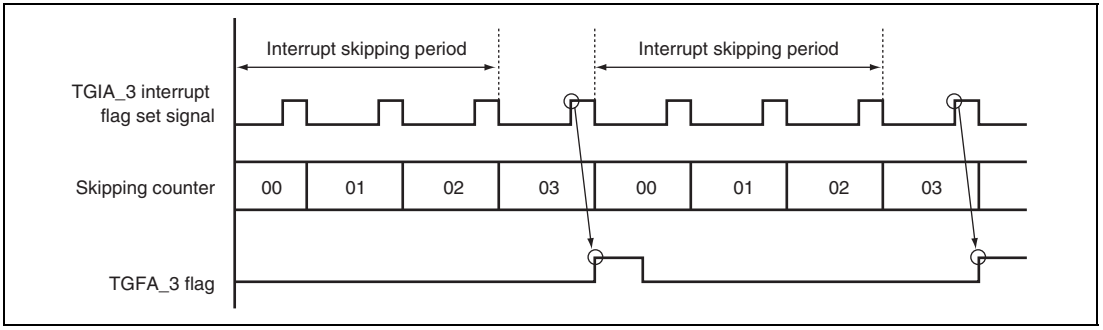


Figure 11.75 Example of Interrupt Skipping Operation

(c) Buffer Transfer Control Linked with Interrupt Skipping

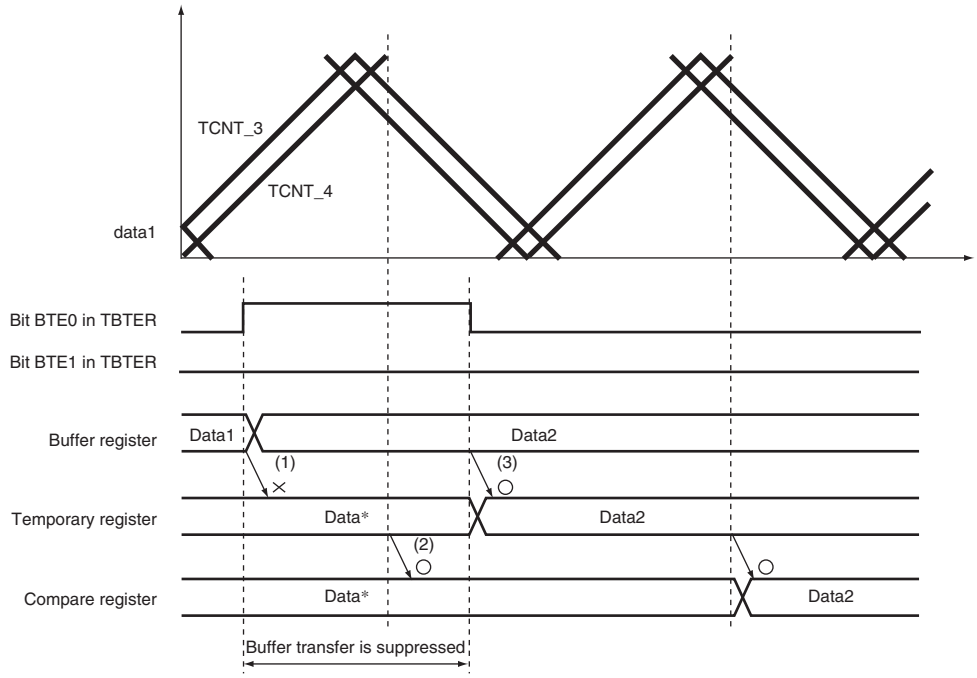
In complementary PWM mode, whether to transfer data from a buffer register to a temporary register and whether to link the transfer with interrupt skipping can be specified with the BTE[1:0] bits in the timer buffer transfer set register (TBTER).

Figure 11.76 shows an example of operation when buffer transfer is suppressed (BTE[1:0] = B'01). While this setting is valid, data is not transferred from the buffer register to the temporary register.

Figure 11.77 shows an example of operation when buffer transfer is linked with interrupt skipping (BTE[1:0] = B'10). While this setting is valid, data is not transferred from the buffer register outside the buffer transfer-enabled period. Depending on the timing of interrupt generation and writing to the buffer register, the timing of transfer from the buffer register to the temporary register and from the temporary register to the compare register is one of two types.

Note that the buffer transfer-enabled period depends on the T3AEN and T4VEN bit settings in the timer interrupt skipping set register (TITCR). Figure 11.78 shows the relationship between the T3AEN and T4VEN bit settings in TITCR and buffer transfer-enabled period.

Note: This function must always be used in combination with interrupt skipping.
When interrupt skipping is disabled (the T3AEN and T4VEN bits in the timer interrupt skipping set register (TITCR) are cleared to 0 or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0), make sure that buffer transfer is not linked with interrupt skipping (clear the BTE1 bit in the timer buffer transfer set register (TBTER) to 0). If buffer transfer is linked with interrupt skipping while interrupt skipping is disabled, buffer transfer is never performed.

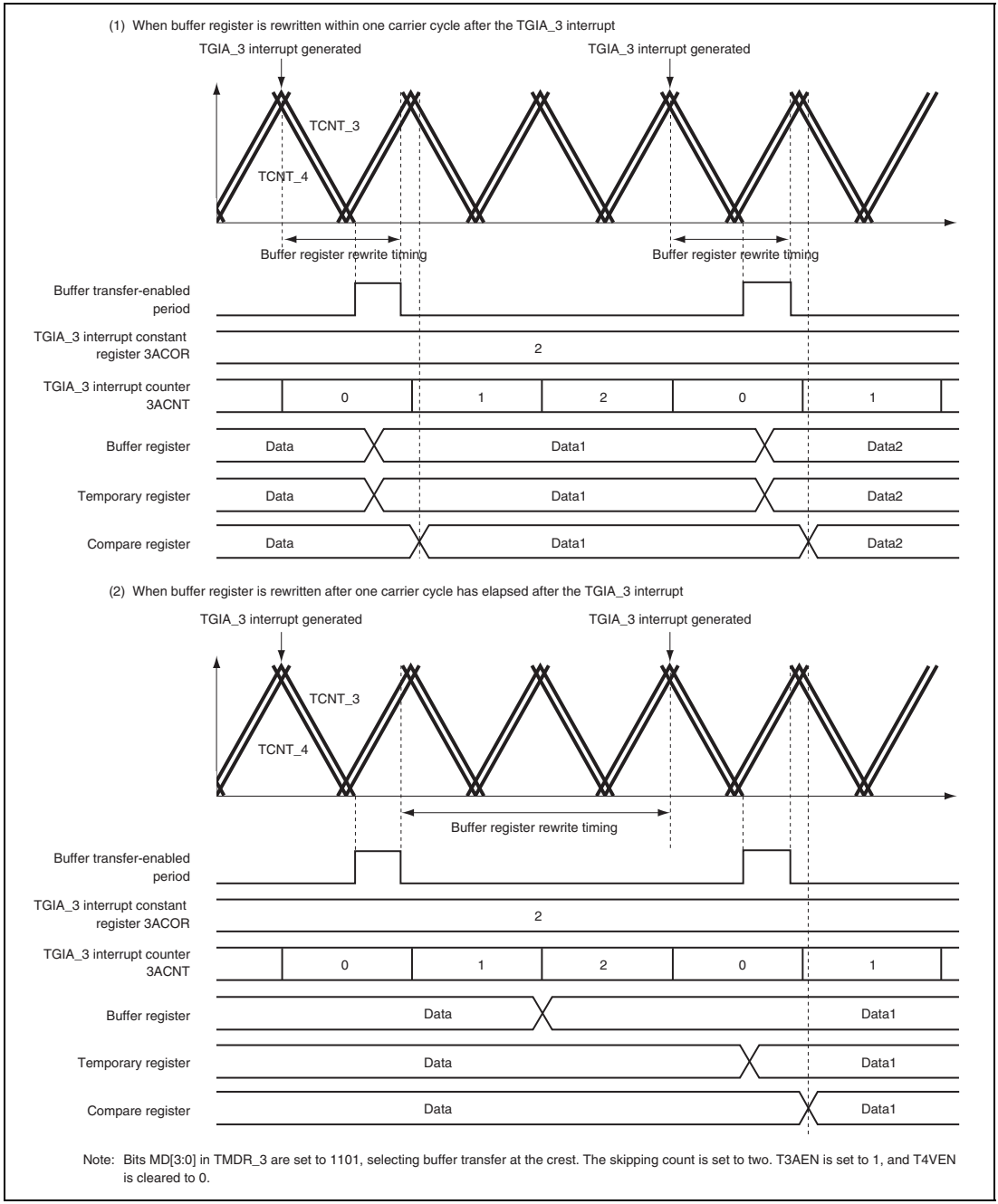


[Legend]

(1) No data is transferred from the buffer register to the temporary register in the buffer transfer-disabled period (bits BTE[1:0] in TBTER are set to B'01, respectively).
(2) Data is transferred from the temporary register to the compare 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.

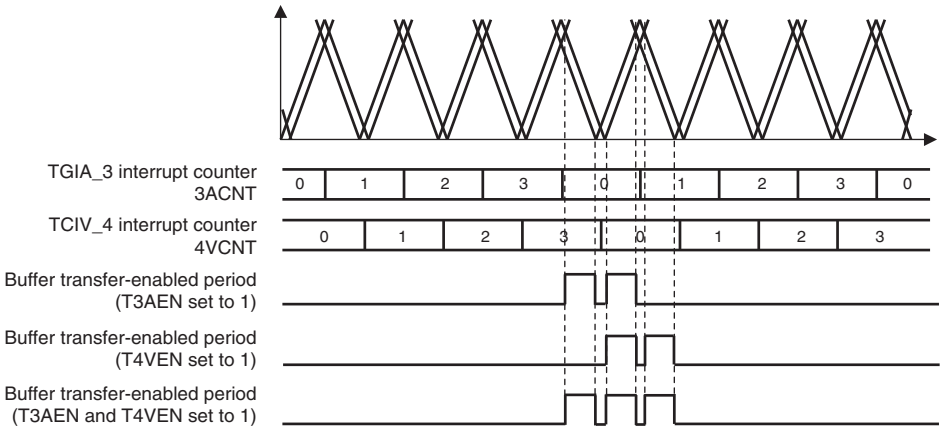
Note: * When buffer transfer at the crest is selected.

Figure 11.76 Example of Operation when Buffer Transfer Is Suppressed (BTE[1:0] = B'01)



Note: Bits MD[3:0] in TMDR_3 are set to 1101, selecting buffer transfer at the crest. The skipping count is set to two. T3AEN is set to 1, and T4VEN is cleared to 0.

Figure 11.77 Example of Operation when Buffer Transfer Is Linked with Interrupt Skipping (BTE[1:0] = B'10)



Note: Bits MD[3:0] in TMDR_3 are set to 1111, selecting buffer transfer at the crest and trough. The skipping count is set to three. T3AEN and T4VEN are set to 1.

Figure 11.78 Relationship between Bits T3AEN and T4VEN in Timer Interrupt Skipping Set Register (TITCR) 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 the timer read/write enable register (TRWER). The applicable registers are some (21 in total) of the registers in channels 3 and 4 shown in the following:

- TCR_3 and TCR_4, TMDR_3 and TMDR_4, TIORH_3 and TIORH_4, TIORL_3 and TIORL_4, TIER_3 and TIER_4, TCNT_3 and TCNT_4, TGRA_3 and TGRA_4, TGRB_3 and TGRB_4, TOER, TOCR, TGCR, TCDR, and TDDR.

This function enables miswriting due to CPU runaway to be prevented by disabling CPU access to the mode registers, control registers, and counters. When the applicable registers are read in the access-disabled state, undefined values are returned. Writing to these registers is ignored.

(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. There are four external signal input pins.

See section 13, Port Output Enable (POE), for details.

(c) Halting of PWM output when oscillator is stopped

If it is detected that the clock input to this LSI has stopped, the 6-phase PWM output pins automatically go to the high-impedance state. The pin states are not guaranteed when the clock is restarted.

See section 4.7, Function for Detecting Oscillator Stop.

11.4.9 A/D Converter Start Request Delaying Function

A/D converter start requests can be issued in channel 4 by making settings in the timer A/D converter start request control register (TADCR), timer A/D converter start request cycle set registers (TADCORA_4 and TADCORB_4), and timer A/D converter start request cycle set buffer registers (TADCOBRA_4 and TADCOBRB_4).

The A/D converter start request delaying function compares TCNT_4 with TADCORA_4 or TADCORB_4, and when their values match, the function issues a respective A/D converter start request (TRG4AN or TRG4BN).

A/D converter start requests (TRG4AN and TRG4BN) can be skipped in coordination with interrupt skipping by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in TADCR.

(a) Example of Procedure for Specifying A/D Converter Start Request Delaying Function

Figure 11.79 shows an example of procedure for specifying the A/D converter start request delaying function.

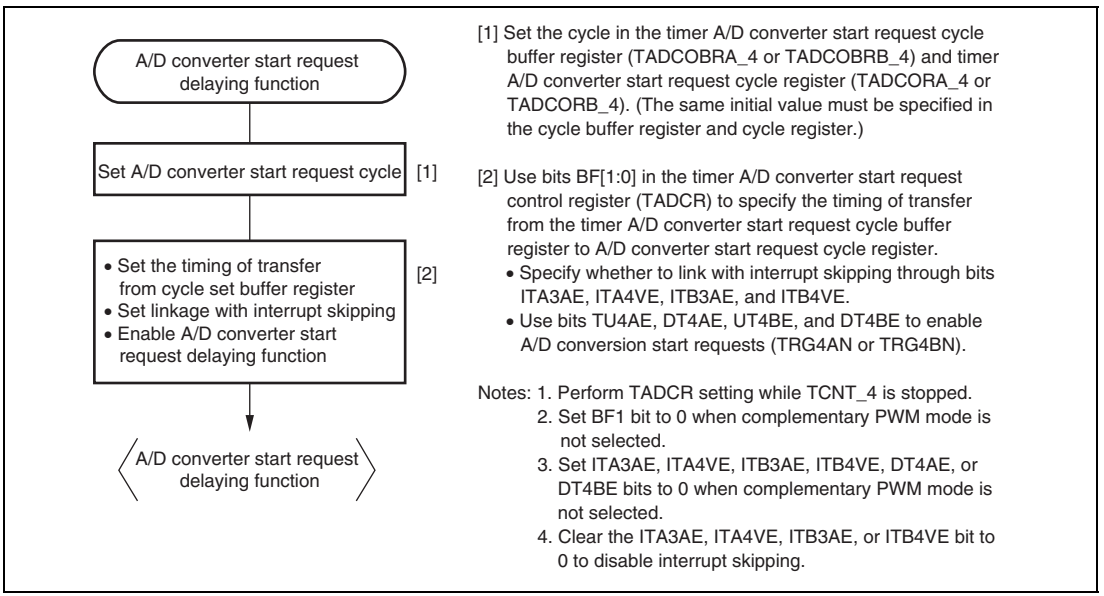


Figure 11.79 Example of Procedure for Specifying A/D Converter Start Request Delaying Function

(b) Basic Operation Example of A/D Converter Start Request Delaying Function

Figure 11.80 shows a basic example of A/D converter request signal (TRG4AN) operation when the trough of TCNT_4 is specified for the buffer transfer timing and an A/D converter start request signal is output during TCNT_4 down-counting.

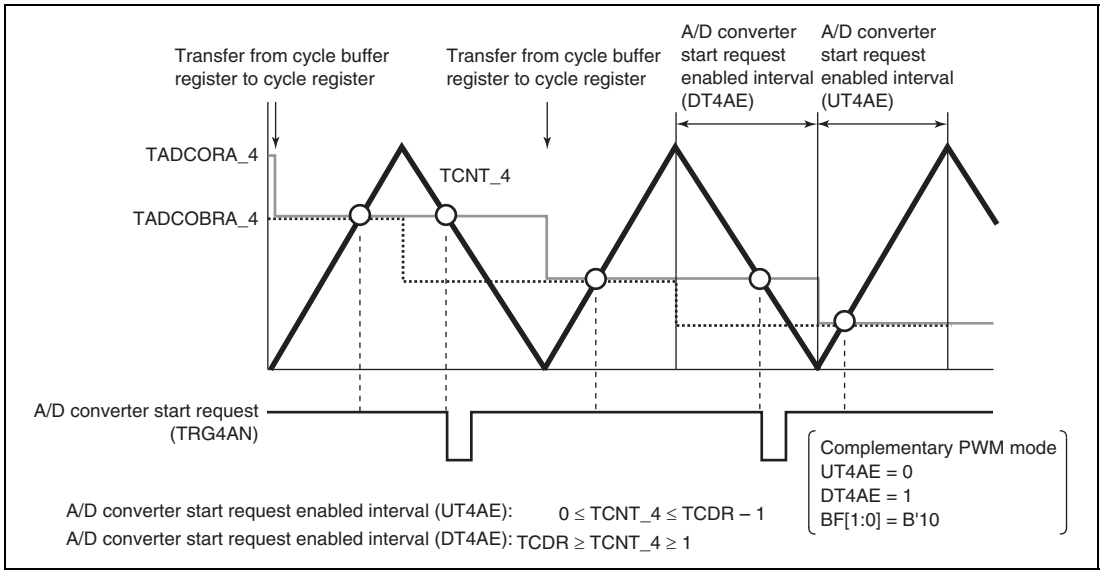


Figure 11.80 Basic Example of A/D Converter Start Request Signal (TRG4AN) Operation

(c) A/D Converter Start Request Enabled Interval

When TCNT_4 and TADCORA_4/TADCORB_4 match during the interval enabled by the UT4AE, DT4AE, UT4BE, or DT4BE bit in TADCR, a start request is issued for the corresponding A/D converter (TRG4AN or TRG4BN).

If the UT4AE or UT4BE bit is set to 1 in complementary PWM mode, A/D converter start requests are enabled during the TCNT_4 up-counting interval (0 ≤ TCNT_4 ≤ TCDR−1). A/D converter start requests are enabled during the TCNT_4 down-counting interval (TCDR ≥ TCNT_4 ≥ 1) if the DT4AE or DT4BE bit is set to 1 (figure 11.80).

Clear the DT4AE and DT4BE bits to 0 when not in complementary PWM mode. Setting the UT4AE or UT4BE bit to 1 causes an A/D converter start request to be generated at a compare match between TCNT_4 and TADCORA_4/TADCORB_4, regardless of whether TCNT_4 is counting up or down.

(d) Buffer Transfer

The data in the timer A/D converter start request cycle set registers (TADCORA_4 and TADCORB_4) is updated by writing data to the timer A/D converter start request cycle set buffer registers (TADCOBRA_4 and TADCOBRB_4). Data is transferred from the buffer registers to the respective cycle set registers at the timing selected with the BF[1:0] bits in the timer A/D converter start request control register (TADCR).

When using buffer transfer in complementary PWM mode, exercise care regarding the buffer transfer timing. For details, see 11.7.24, Notes on Using the A/D Converter Start Request Delaying Function in Complementary PWM Mode.

Also, clear the BF1 bit to 0 when not in complementary PWM mode.

(e) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping

In complementary PWM mode, A/D converter start requests (TRG4AN and TRG4BN) can be issued in coordination with interrupt skipping by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the timer A/D converter start request control register (TADCR).

Figure 11.81 shows an example of A/D converter start request signal (TRG4AN) operation when TRG4AN output is enabled during TCNT_4 up-counting and down-counting and A/D converter start requests are linked with interrupt skipping.

Figure 11.82 shows another example of A/D converter start request signal (TRG4AN) operation when TRG4AN output is enabled during TCNT_4 up-counting and A/D converter start requests are linked with interrupt skipping.

The A/D converter start request delaying function linked to the interrupt skipping function cannot be used when not in complementary PWM mode. In this case, clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in TADCR to 0.

Note: This function must be used in combination with interrupt skipping.
When interrupt skipping is disabled (the T3AEN and T4VEN bits in the timer interrupt skipping set register (TITCR) are cleared to 0 or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0), make sure that A/D converter start requests are not linked with interrupt skipping (clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the timer A/D converter start request control register (TADCR) to 0).

Furthermore, when this function is to be used, set TADCORA_4 and TADCORB_4 to a value between H'0002 and the TCDR setting minus two.

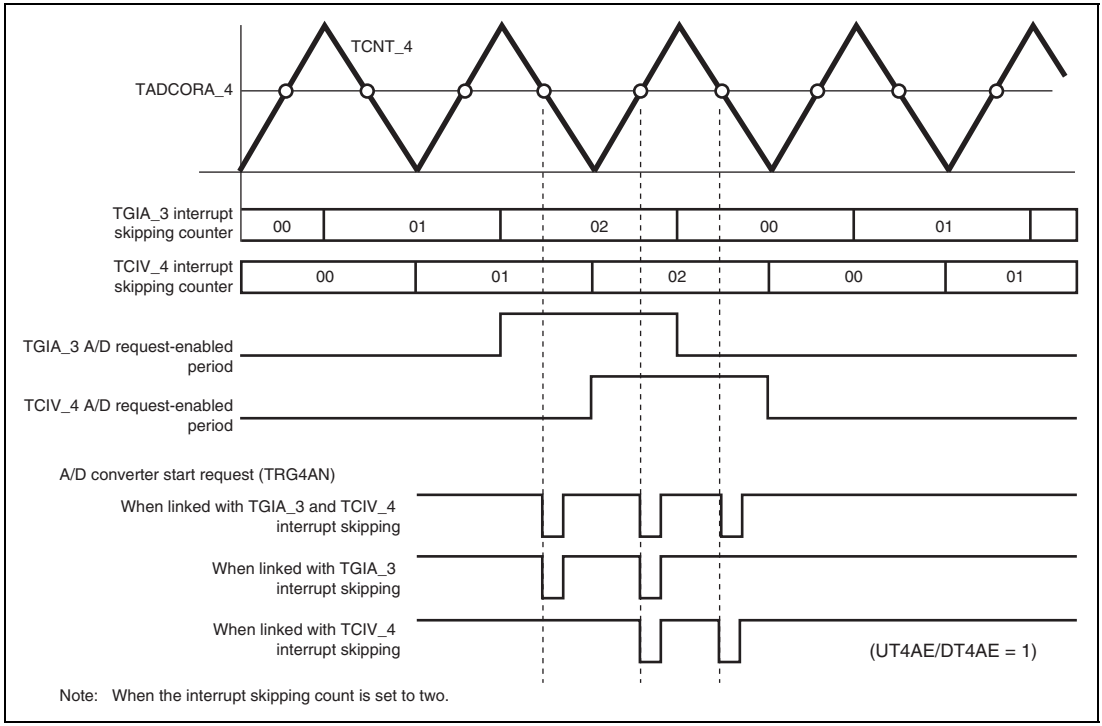


Figure 11.81 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping

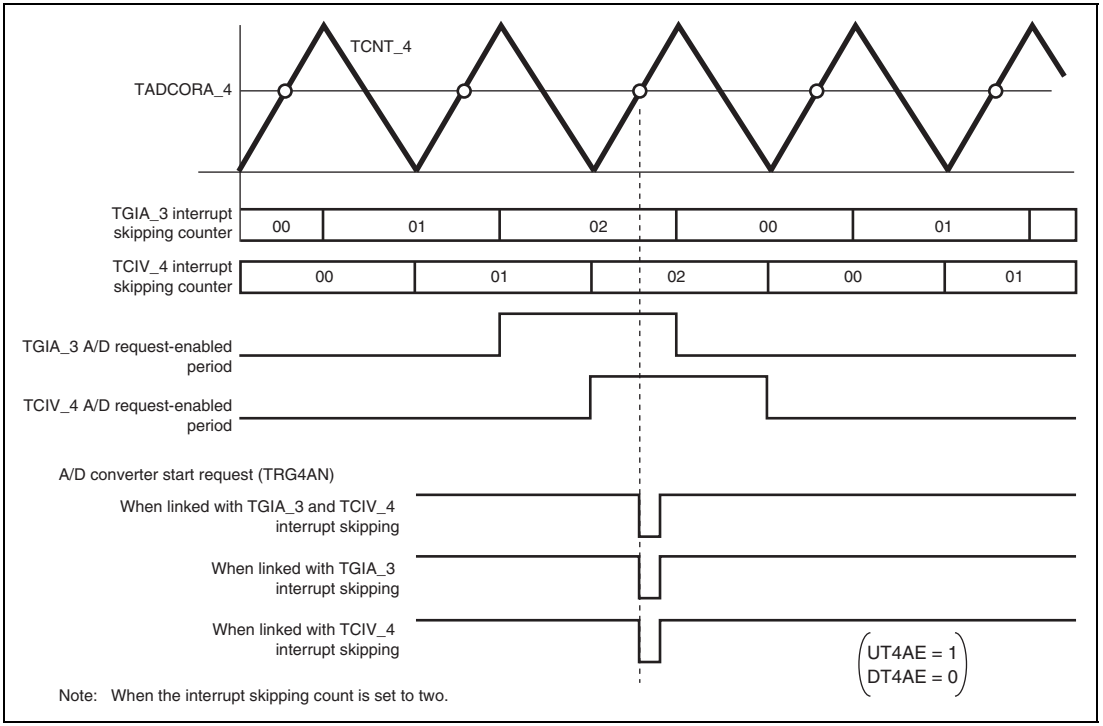


Figure 11.82 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping

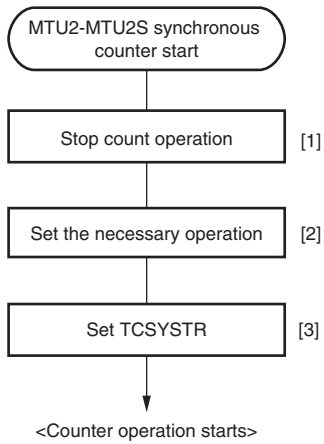
11.4.10 MTU2–MTU2S Synchronous Operation

(1) MTU2–MTU2S Synchronous Counter Start

The counters in the MTU2 and MTU2S which operate at different clock systems can be started synchronously by making the TCSYSTR settings in the MTU2.

(a) Example of MTU2–MTU2S Synchronous Counter Start Setting Procedure

Figure 11.83 shows an example of synchronous counter start setting procedure.



[1] Use TSTR registers in the MTU2 and MTU2S and halt the counters used for synchronous start operation.

[2] Specify necessary operation with appropriate registers such as TCR and TMDR.

[3] In TCSYSTR in the MTU2, set the bits corresponding to the counters to be started synchronously to 1. The TSTRs are automatically set appropriately and the counters start synchronously.

Notes: 1. Even if a bit in TCSYSTR corresponding to an operating counter is cleared to 0, the counter will not stop. To stop the counter, clear the corresponding bits in TSTR and TSTRS to 0 directly.

2. To start channels 3 and 4 in reset-synchronized PWM mode or complementary PWM mode, make appropriate settings in TCYSTR according to the TSTR setting for the respective mode. For details, refer to section 11.4.7, Reset-Synchronized PWM Mode, and section 11.4.8, Complementary PWM Mode.

Figure 11.83 Example of Synchronous Counter Start Setting Procedure

(b) Examples of Synchronous Counter Start Operation

Figures 11.84 (1), 11.84 (2), 11.84 (3), and 11.84 (4) show examples of synchronous counter start operation when the clock frequency ratio between the MTU2 and MTU2S is 1:1, 1:2, 1:3, and 1:4, respectively.

In these examples, the count clock settings are MPϕ/1 (MTU2) and MIϕ/1 (MTU2S).

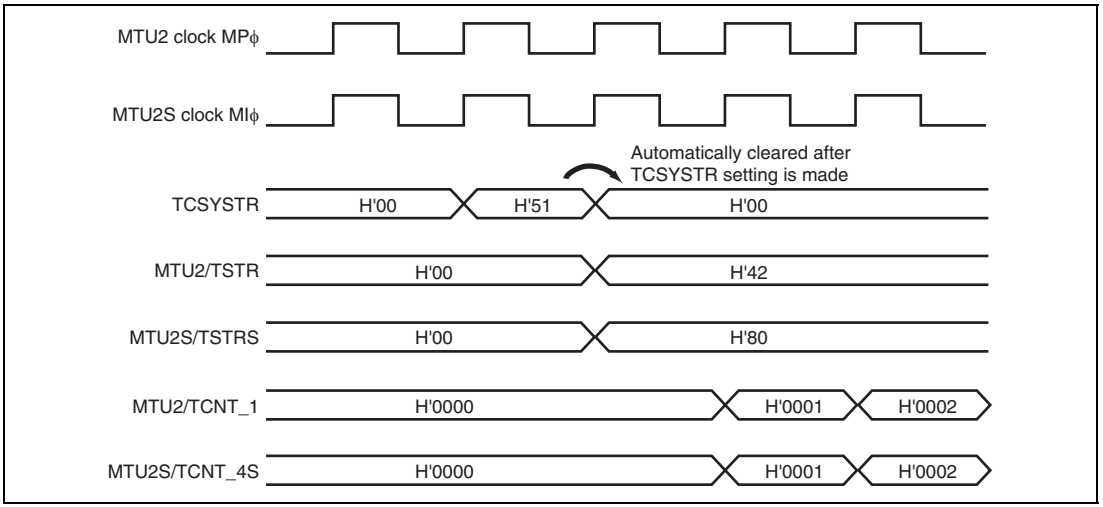


Figure 11.84 (1) Example of Synchronous Counter Start Operation (MTU2-to-MTU2S Clock Frequency Ratio = 1:1)

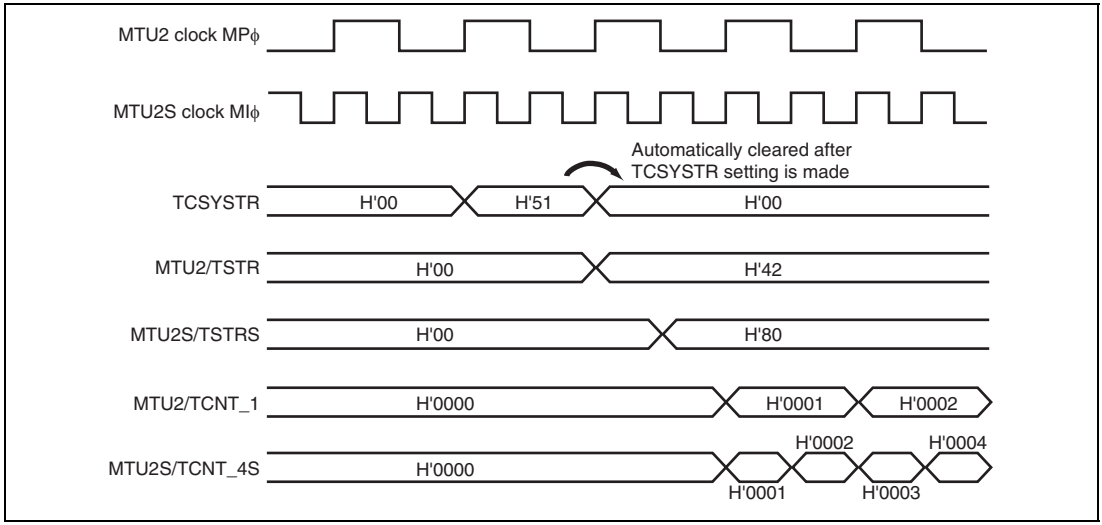


Figure 11.84 (2) Example of Synchronous Counter Start Operation (MTU2-to-MTU2S Clock Frequency Ratio = 1:2)

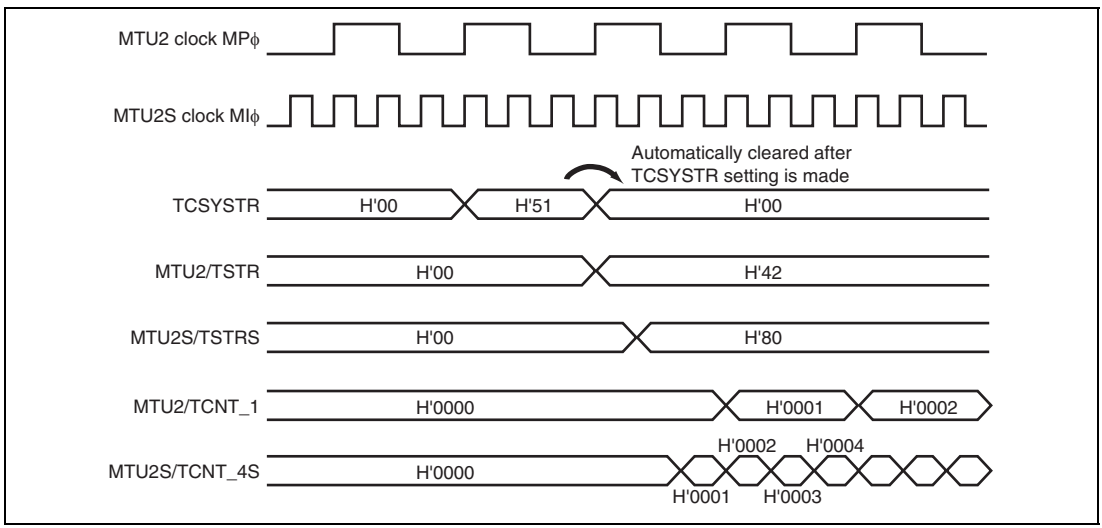


Figure 11.84 (3) Example of Synchronous Counter Start Operation (MTU2-to-MTU2S Clock Frequency Ratio = 1:3)

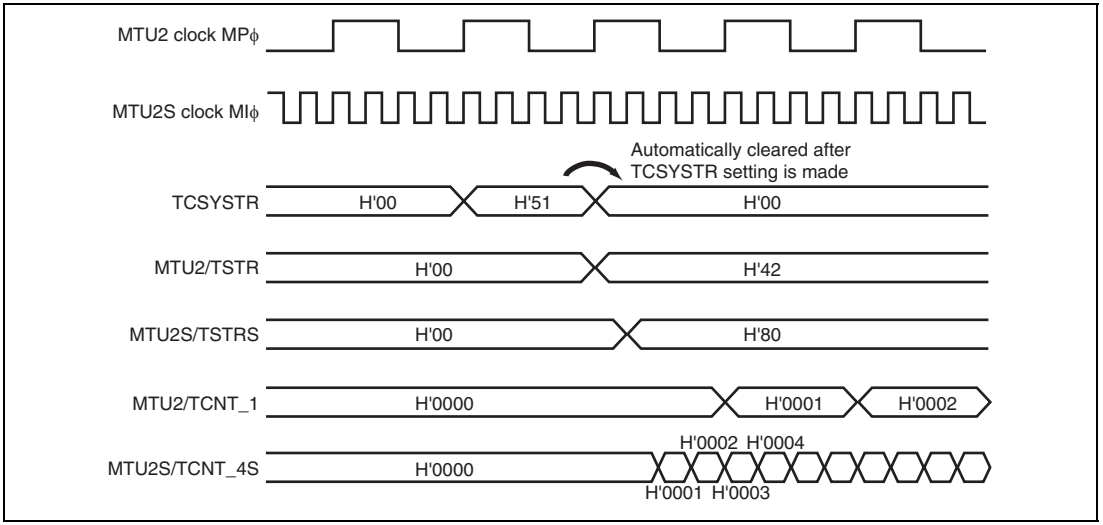


Figure 11.84 (4) Example of Synchronous Counter Start Operation (MTU2-to-MTU2S Clock Frequency Ratio = 1:4)

(2) MTU2S Counter Clearing Caused by MTU2 Flag Setting Source (MTU2–MTU2S Synchronous Counter Clearing)

The MTU2S counters can be cleared by sources for setting the flags in TSR_0 to TSR_2 in the MTU2 through the TSYCRS settings in the MTU2S.

(a) Example of Procedure for Specifying MTU2S Counter Clearing by MTU2 Flag Setting Source

Figure 11.85 shows an example of procedure for specifying MTU2S counter clearing by MTU2 flag setting source.

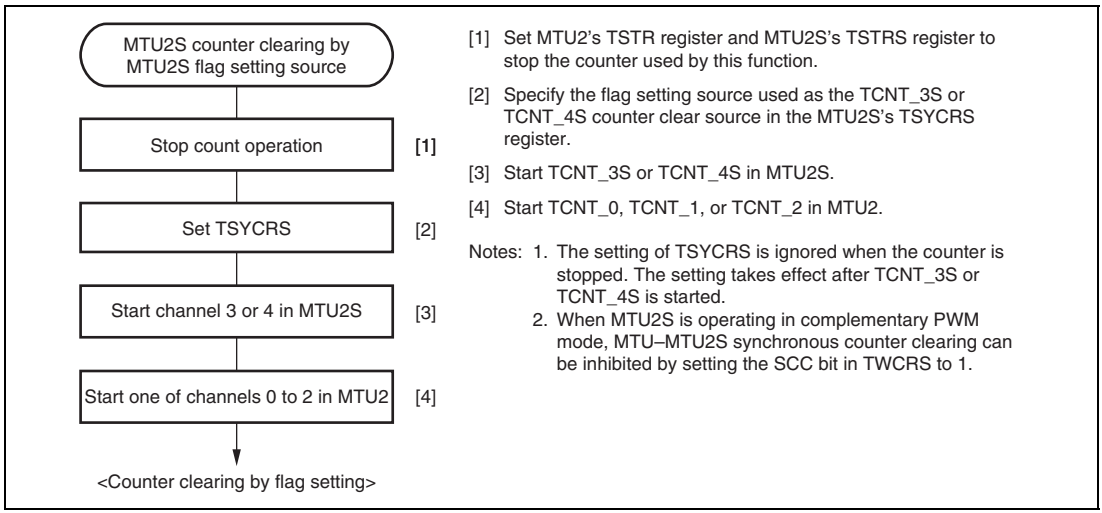


Figure 11.85 Example of Procedure for Specifying MTU2S Counter Clearing by MTU2 Flag Setting Source

(b) Examples of MTU2S Counter Clearing Caused by MTU2 Flag Setting Source

Figures 11.86 (1) and 11.86 (2) show examples of MTS2S counter clearing caused by MTU2 flag setting source.

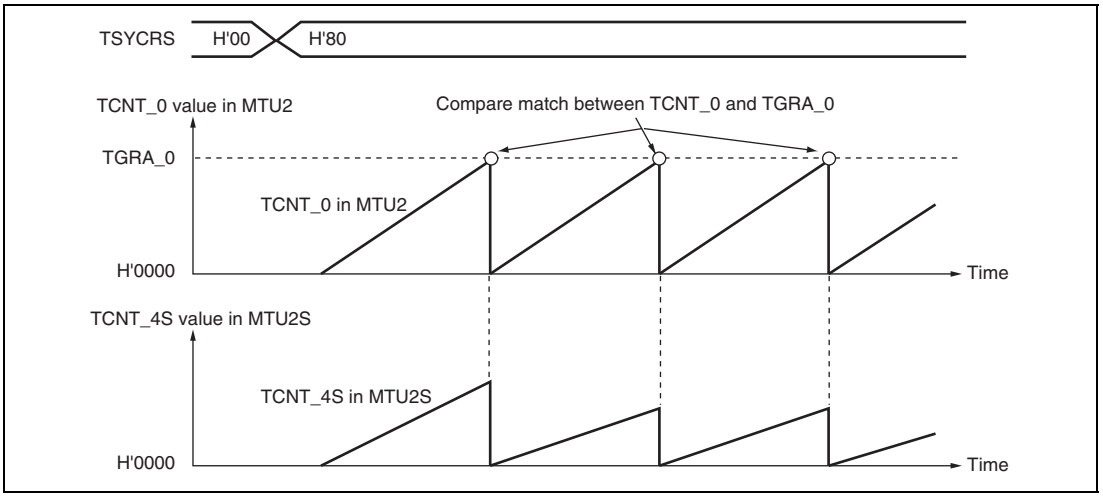


Figure 11.86 (1) Example of MTU2S Counter Clearing Caused by MTU2 Flag Setting Source (1)

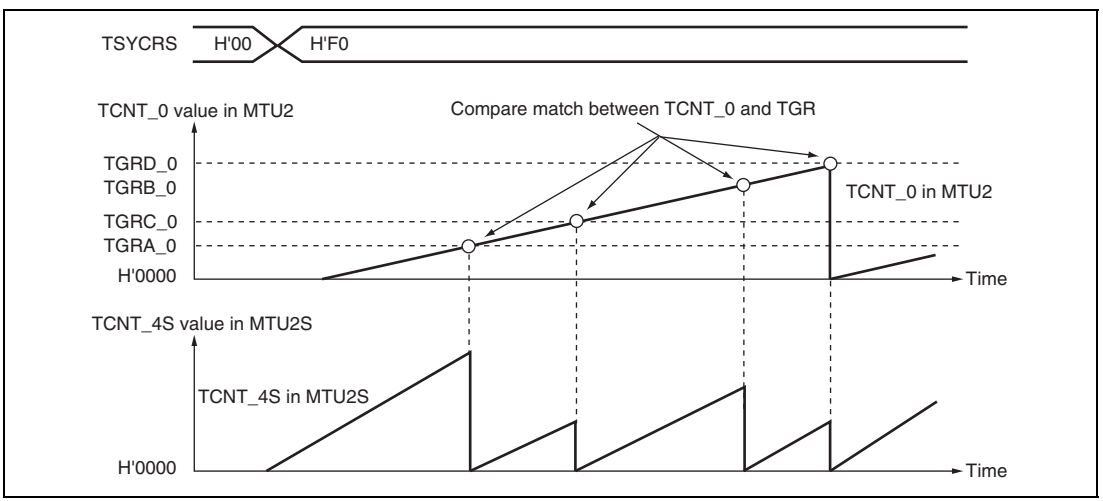


Figure 11.86 (2) Example of MTU2S Counter Clearing Caused by MTU2 Flag Setting Source (2)

11.4.11 External Pulse Width Measurement

The pulse widths of up to three external input lines can be measured in channel 5.

If pulse width measurement is specified by the setting of the IOC[4:0] bit field in TIORU_5/TIORV_5/TIORW_5, the pulse width of the signal input on TIC5U/TIC5V/TIC5W is measured. TCNTU_5/TCNTV_5/TCNTW_5 is incremented when the input level is that specified by the IOC[4:0] bit field.

Figure 11.87 shows an example of the external pulse width measurement setting procedure, and figure 11.88 shows an example of operation.

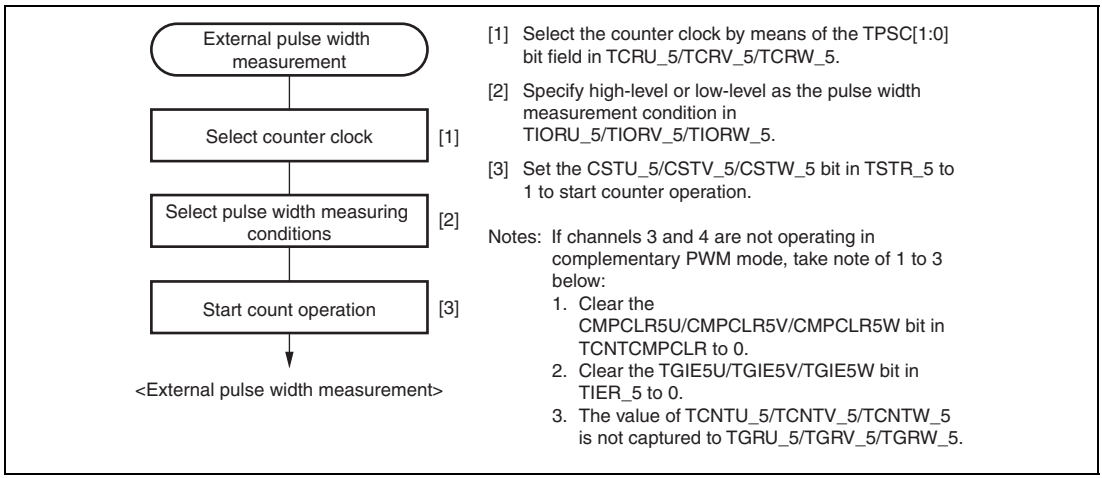


Figure 11.87 Example of External Pulse Width Measurement Setting Procedure

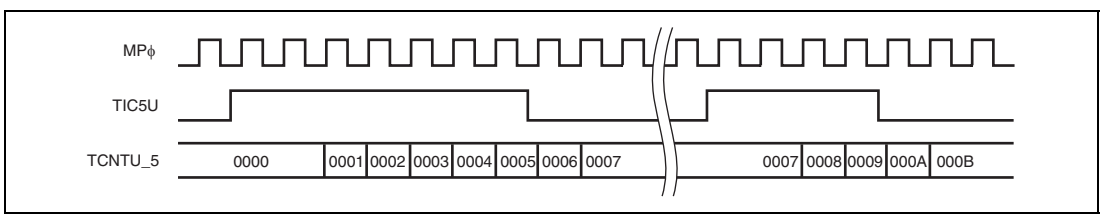


Figure 11.88 Example of External Pulse Width Measurement (Measuring High Pulse Width)

11.4.12 Dead Time Compensation

A motor control circuit (figure 11.89) is provided that feeds back to channel 5 the dead time delay (delay between the complementary PWM output and the inverter output). Dead time compensation can be applied to the PWM output waveform by using the external pulse width measurement function on channel 5 to measure the dead time delay and adjusting the PWM duty accordingly (figure 11.90).

Figure 11.91 shows an example of the procedure for setting the dead time compensation when using channel 5. For details the operation of channel 5, see 11.4.13, TCNTU_5/TCNTV_5/TCNTW_5 Capture at Crest and/or Trough in Complementary PWM Operation.

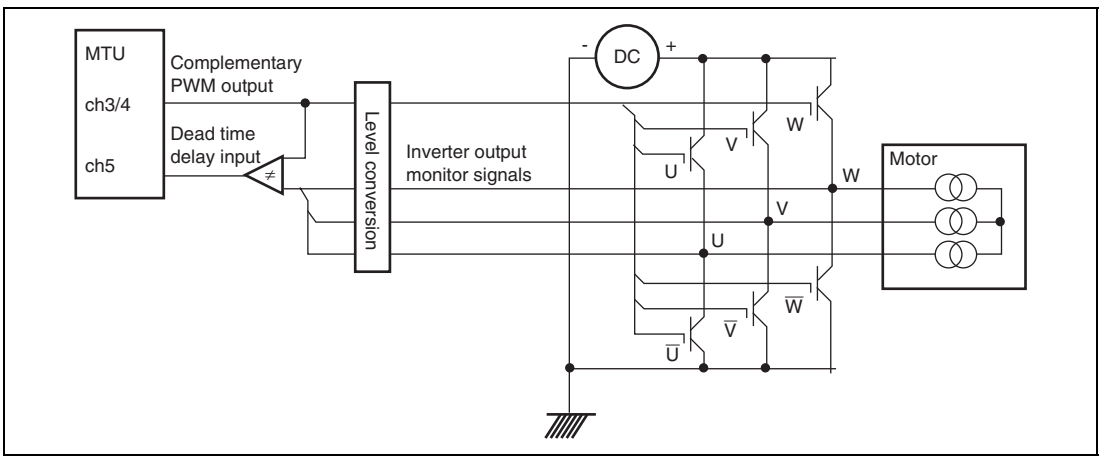


Figure 11.89 Example of Motor Control Circuit Configuration

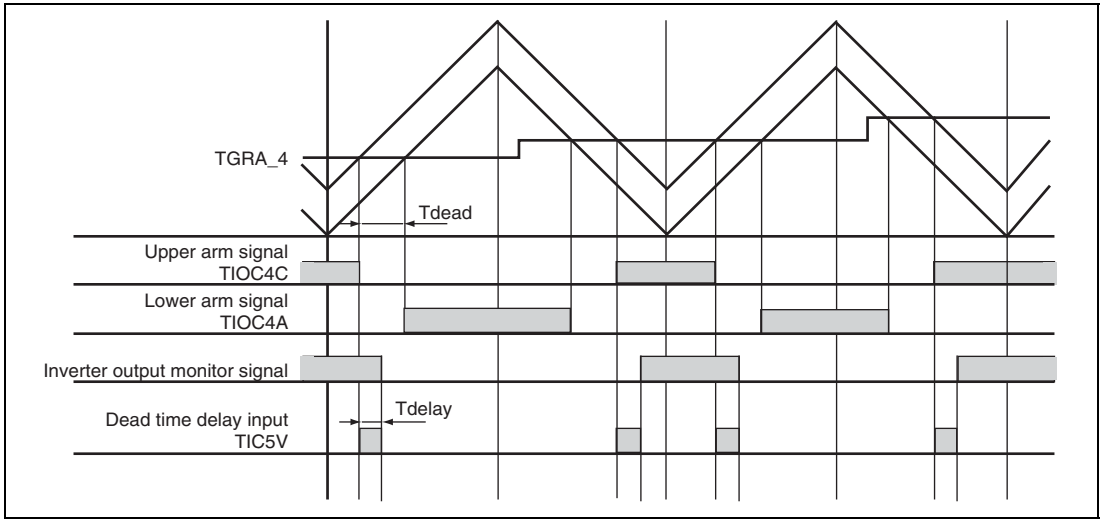
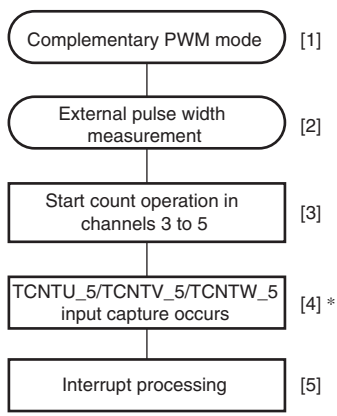


Figure 11.90 Delay in Dead Time in Complementary PWM Operation



[1] Place channels 3 and 4 in complementary PWM mode. For details, refer to section 11.4.8, Complementary PWM Mode.

[2] Specify the external pulse width measurement function for the target TIOR in channel 5. For details, refer to section 11.4.11, External Pulse Width Measurement.

[3] Set bits CST3 and CST4 in TSTR and bit CSTU_5/CSTV_5/CSTW_5 in TSTR_5 to 1 to start count operation.

[4] When the capture condition specified in TIORU_5/TIORV_5/TIORW_5 is satisfied, the TCNTU_5/TCNTV_5/TCNTW_5 value is captured in TGRU_5/TGRV_5/TGRW_5.

[5] For U-phase dead time compensation, when an interrupt is generated at the crest (TGIA_3) or trough (TCIV_4) in complementary PWM mode, read the TGRU_5 value, calculate the difference in time in TGRB_3, and write the corrected value to TGRD_3 in the interrupt processing. For the V phase and W phase, read the TGRV_5 and TGRW_5 values and write the corrected values to TGRC_4 and TGRD_4, respectively, in the same way as for U-phase compensation.
The TCNTU_5/TCNTV_5/TCNTW_5 value should be cleared through the TCNTCMPCLR setting or by software.

Notes: The PFC settings must be completed in advance.
* As an interrupt flag is set under the capture condition specified in TIOR, do not enable interrupt requests in TIER_5.

Figure 11.91 Example of Dead Time Compensation Setting Procedure

11.4.13 TCNTU_5/TCNTV_5/TCNTW_5 Capture at Crest and/or Trough in Complementary PWM Operation

The external pulse width measurement function on channel 5 captures the value of TCNTU_5/TCNTV_5/TCNTW_5 at the crest, the trough, or both the crest and trough, and stores it in TGRU_5/TGRV_5/TGRW_5, during complementary PWM operation. The timing for capturing to TGRU_5/TGRV_5/TGRW_5 is selected by TIORU_5/TIORV_5/TIORW_5. Also, setting to 1 the CMPCLRU/CMPCLRV/CMPCLRW bit in TCNTCMPCLR causes TCNTU_5/TCNTV_5/TCNTW_5 to be cleared when the capture takes place.

Figure 11.92 is an operating example in which TCNTU_5/TCNTV_5/TCNTW_5 is used as a free-running counter without being cleared, and the TCNTU_5/TCNTV_5/TCNTW_5 value is captured to TGRU_5/TGRV_5/TGRW_5 at the crest and trough during complementary PWM operation.

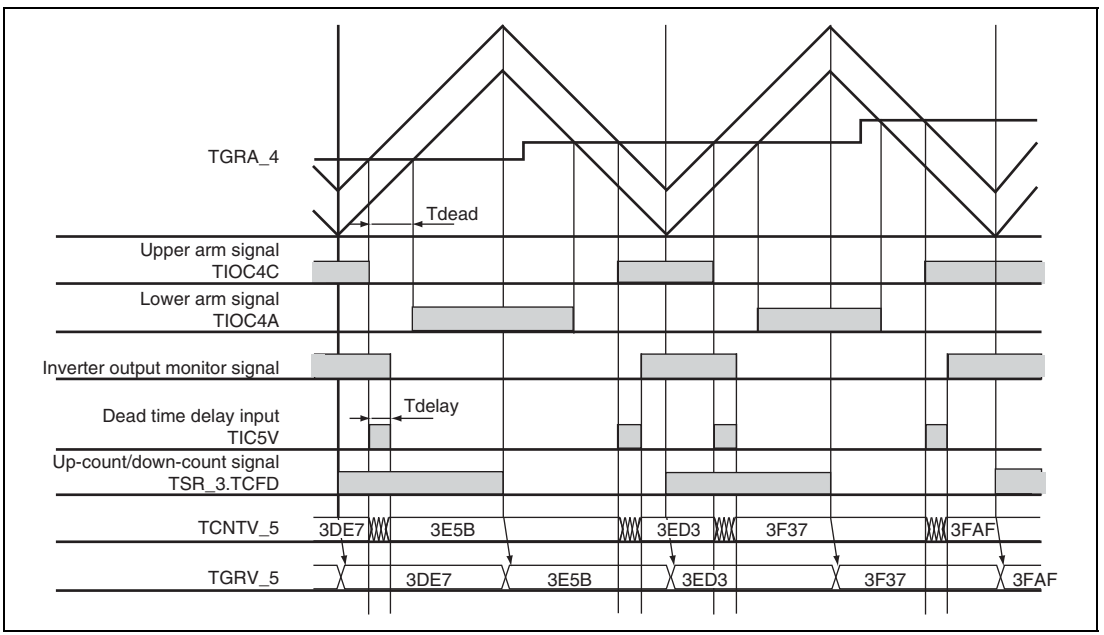


Figure 11.92 TCNTU_5/TCNTV_5/TCNTW_5 Capturing at Crest and/or Trough in Complementary PWM Operation

11.5 Interrupt Sources

11.5.1 Interrupt Sources and Priorities

There are three kinds of MTU2 interrupt source; TGR input capture/compare match, TCNT overflow, and TCNT underflow. Each interrupt source has its own status flag and enable/disabled bit, allowing the generation of interrupt request signals to be enabled or disabled individually.

When an interrupt request is generated, the corresponding status flag in TSR is set to 1. If the corresponding enable/disable bit in TIER is set to 1 at this time, an interrupt is requested. The interrupt request is cleared by clearing the status flag to 0.

Relative channel priorities can be changed by the interrupt controller, however the priority order within a channel is fixed. For details, see section 6, Interrupt Controller (INTC).

Table 11.57 lists the MTU2 interrupt sources.

Table 11.57 MTU2 Interrupts

Channel	Name	Interrupt Source	Interrupt Flag	DMAC Activation	DTC Activation	Priority
0	TGIA_0	TGRA_0 input capture/compare match	TGFA_0	Possible	Possible	High
	TGIB_0	TGRB_0 input capture/compare match	TGFB_0	Not possible	Possible	
	TGIC_0	TGRC_0 input capture/compare match	TGFC_0	Not possible	Possible	
	TGID_0	TGRD_0 input capture/compare match	TGFD_0	Not possible	Possible	
	TCIV_0	TCNT_0 overflow	TCFV_0	Not possible	Not possible	
	TGIE_0	TGRE_0 compare match	TGFE_0	Not possible	Not possible	
	TGIF_0	TGRF_0 compare match	TGFF_0	Not possible	Not possible	
1	TGIA_1	TGRA_1 input capture/compare match	TGFA_1	Possible	Possible	
	TGIB_1	TGRB_1 input capture/compare match	TGFB_1	Not possible	Possible	
	TCIV_1	TCNT_1 overflow	TCFV_1	Not possible	Not possible	
	TCIU_1	TCNT_1 underflow	TCFU_1	Not possible	Not possible	
2	TGIA_2	TGRA_2 input capture/compare match	TGFA_2	Possible	Possible	
	TGIB_2	TGRB_2 input capture/compare match	TGFB_2	Not possible	Possible	
	TCIV_2	TCNT_2 overflow	TCFV_2	Not possible	Not possible	
	TCIU_2	TCNT_2 underflow	TCFU_2	Not possible	Not possible	
3	TGIA_3	TGRA_3 input capture/compare match	TGFA_3	Possible	Possible	
	TGIB_3	TGRB_3 input capture/compare match	TGFB_3	Not possible	Possible	
	TGIC_3	TGRC_3 input capture/compare match	TGFC_3	Not possible	Possible	
	TGID_3	TGRD_3 input capture/compare match	TGFD_3	Not possible	Possible	
	TCIV_3	TCNT_3 overflow	TCFV_3	Not possible	Not possible	
4	TGIA_4	TGRA_4 input capture/compare match	TGFA_4	Possible	Possible	
	TGIB_4	TGRB_4 input capture/compare match	TGFB_4	Not possible	Possible	
	TGIC_4	TGRC_4 input capture/compare match	TGFC_4	Not possible	Possible	
	TGID_4	TGRD_4 input capture/compare match	TGFD_4	Not possible	Possible	
	TCIV_4	TCNT_4 overflow/underflow	TCFV_4	Not possible	Possible	
5	TGIU_5	TGRU_5 input capture/compare match	TGFU_5	Not possible	Possible	
	TGIV_5	TGRV_5 input capture/compare match	TGFV_5	Not possible	Possible	
	TGIW_5	TGRW_5 input capture/compare match	TGFW_5	Not possible	Possible	Low

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

(1) Input Capture/Compare Match Interrupt

An interrupt is requested if the TGIE bit in TIER is set to 1 when the TGF flag in TSR is set to 1 by the occurrence of a TGR input capture/compare match on a particular channel. The interrupt request is cleared by clearing the TGF flag to 0. The MTU2 has 21 input capture/compare match interrupts, six for channel 0, four each for channels 3 and 4, 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 TIER is set to 1 when the TCFV flag in TSR is set to 1 by the occurrence of TCNT overflow on a channel. The interrupt request is cleared by clearing the TCFV flag to 0. The MTU2 has five overflow interrupts, one for each channel.

(3) Underflow Interrupt

An interrupt is requested if the TCIEU bit in TIER is set to 1 when the TCFU flag in TSR is set to 1 by the occurrence of TCNT underflow on a channel. The interrupt request is cleared by clearing the TCFU flag to 0. The MTU2 has two underflow interrupts, one each for channels 1 and 2.

11.5.2 DTC/DMAC Activation

(1) DTC Activation

The DTC can be activated by the TGR input capture/compare match interrupt in each channel or the overflow interrupt in channel 4. For details, see section 8, Data Transfer Controller (DTC).

A total of 20 MTU2 input capture/compare match interrupts and overflow interrupts can be used as DTC activation sources, four each for channels 0 and 3, two each for channels 1 and 2, five for channel 4, and three for channel 5.

(2) DMAC Activation

The DMAC can be activated by the TGRA input capture/compare match interrupt in each channel. For details, see section 10, Direct Memory Access Controller (DMAC).

In the MTU2, a total of five TGRA input capture/compare match interrupts can be used as DMAC activation sources, one each for channels 0 to 4.

When the DMAC is activated by the MTU2, the activation source is cleared at the point the DMAC requests the internal bus mastership. Therefore, the request for DMAC transfer may be kept pending for a certain period even after the activation source is cleared depending on the

internal bus state. To initiate burst transfer by the DMAC using an MTU2 interrupt, setting of the bus function extending register (BSCEHR) is necessary. For details, see section 9.4.8, Bus Function Extending Register (BSCEHR).

11.5.3 A/D Converter Activation

The A/D converter can be activated by one of the following three methods in the MTU2. Table 11.58 shows the relationship between interrupt sources and A/D converter start request signals.

(1) A/D Converter Activation by TGRA Input Capture/Compare Match or at TCNT_4 Trough in Complementary PWM Mode

The A/D converter can be activated by the occurrence of a TGRA input capture/compare match in each channel. In addition, if complementary PWM operation is performed while the TTGE2 bit in TIER_4 is set to 1, the A/D converter can be activated at the trough of TCNT_4 count (TCNT_4 = H'0000).

A/D converter start request signal TRGAN is issued to the A/D converter under either one of the following conditions.

- When the TGFA flag in TSR is set to 1 by the occurrence of a TGRA input capture/compare match on a particular channel while the TTGE bit in TIER is set to 1
- When the TCNT_4 count reaches the trough (TCNT_4 = H'0000) during complementary PWM operation while the TTGE2 bit in TIER_4 is set to 1

When either condition is satisfied, if A/D converter start signal TRGAN from the MTU2 is selected as the trigger in the A/D converter, A/D conversion will start.

(2) A/D Converter Activation by Compare Match between TCNT_0 and TGRE_0

The A/D converter can be activated by generating A/D converter start request signal TRG0N when a compare match occurs between TCNT_0 and TGRE_0 in channel 0.

When the TGFE flag in TSR2_0 is set to 1 by the occurrence of a compare match between TCNT_0 and TGRE_0 in channel 0 while the TTGE2 bit in TIER2_0 is set to 1, A/D converter start request TGR0N is issued to the A/D converter. If A/D converter start signal TGR0N from the MTU2 is selected as the trigger in the A/D converter, A/D conversion will start.

(3) A/D Converter Activation by A/D Converter Start Request Delaying Function

The A/D converter can be activated by generating A/D converter start request signal TRG4AN or TRG4BN when the TCNT_4 count matches the TADCORA or TADCORB value if the UT4AE, DT4AE, UT4BE, or DT4BE bit in the A/D converter start request control register (TADCR) is set to 1. For details, refer to section 11.4.9, A/D Converter Start Request Delaying Function.

A/D conversion will start if A/D converter start signal TRG4AN from the MTU2 is selected as the trigger in the A/D converter when TRG4AN is generated or if TRG4BN from the MTU2 is selected as the trigger in the A/D converter when TRG4BN is generated.

Table 11.58 Interrupt Sources and A/D Converter Start Request Signals

Target Registers	Interrupt Source	A/D Converter Start Request Signal
TGRA_0 and TCNT_0	Input capture/compare match	TRGAN
TGRA_1 and TCNT_1		
TGRA_2 and TCNT_2		
TGRA_3 and TCNT_3		
TGRA_4 and TCNT_4		
TCNT_4	TCNT_4 Trough in complementary PWM mode	
TGRE_0 and TCNT_0	Compare match	TRG0N
TADCORA and TCNT_4		TRG4AN
TADCORB and TCNT_4		TRG4BN

11.6 Operation Timing

11.6.1 Input/Output Timing

(1) TCNT Count Timing

Figures 11.93 and 11.94 show TCNT count timing in internal clock operation, and figure 11.95 shows TCNT count timing in external clock operation (normal mode), and figure 11.96 shows TCNT count timing in external clock operation (phase counting mode).

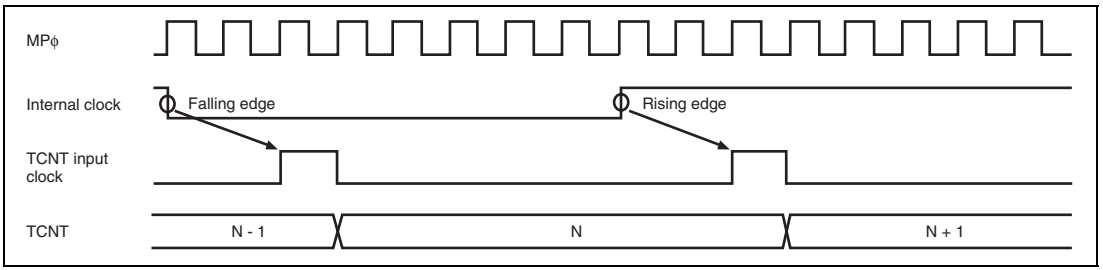


Figure 11.93 Count Timing in Internal Clock Operation (Channels 0 to 4)

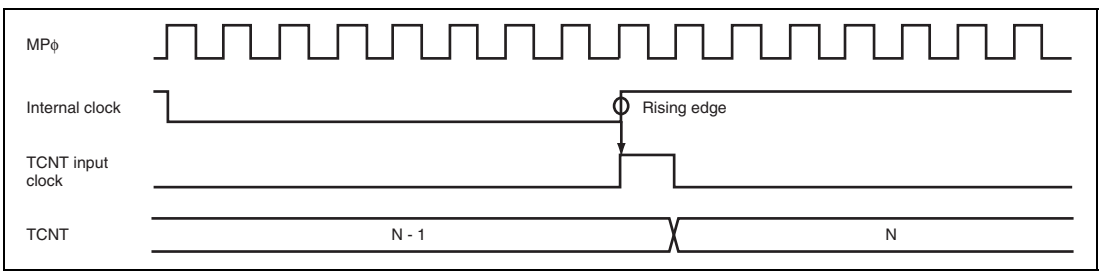


Figure 11.94 Count Timing in Internal Clock Operation (Channel 5)

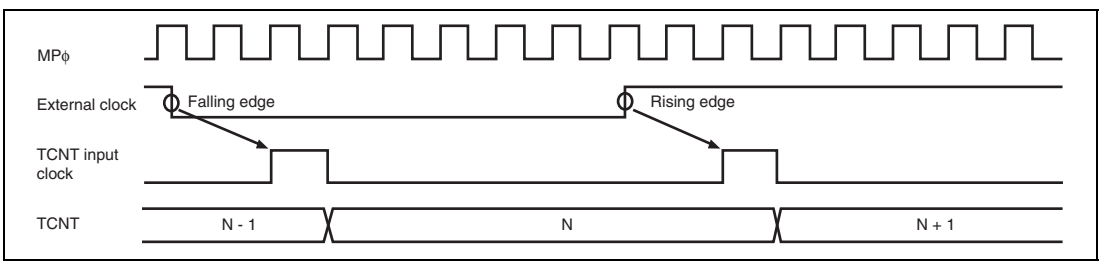


Figure 11.95 Count Timing in External Clock Operation (Channels 0 to 4)

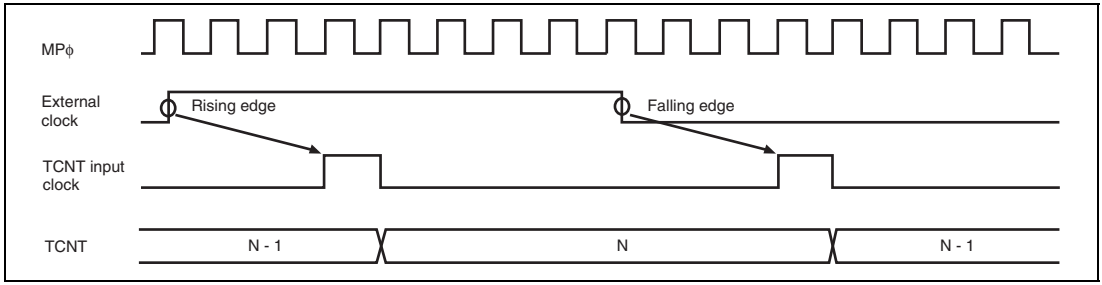


Figure 11.96 Count Timing in External Clock Operation (Phase Counting Mode)

(2) Output Compare Output Timing

A compare match signal is generated in the final state in which TCNT and TGR match (the point at which the count value matched by TCNT is updated). When a compare match signal is generated, the output value set in TIOR is output at the output compare output pin (TIOC pin). After a match between TCNT and TGR, the compare match signal is not generated until the TCNT input clock is generated.

Figure 11.97 shows output compare output timing (normal mode and PWM mode) and figure 11.98 shows output compare output timing (complementary PWM mode and reset synchronous PWM mode).

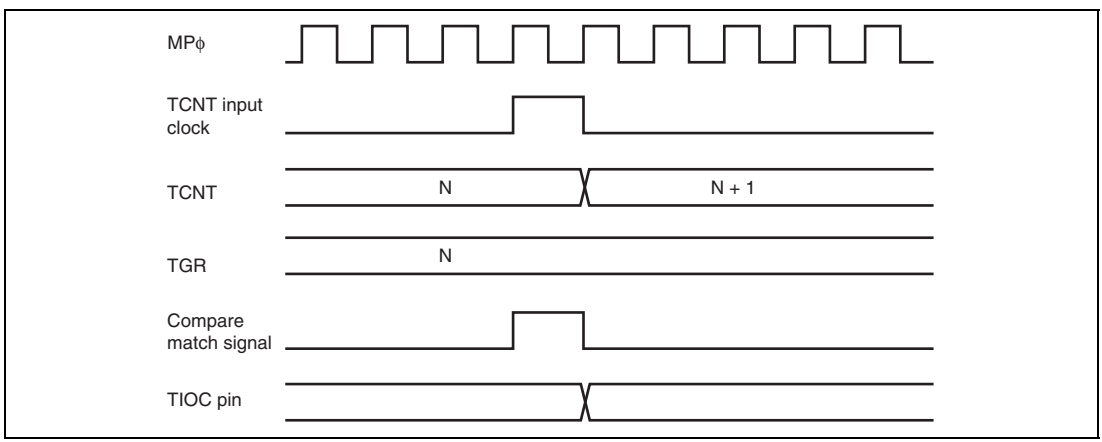


Figure 11.97 Output Compare Output Timing (Normal Mode/PWM Mode)

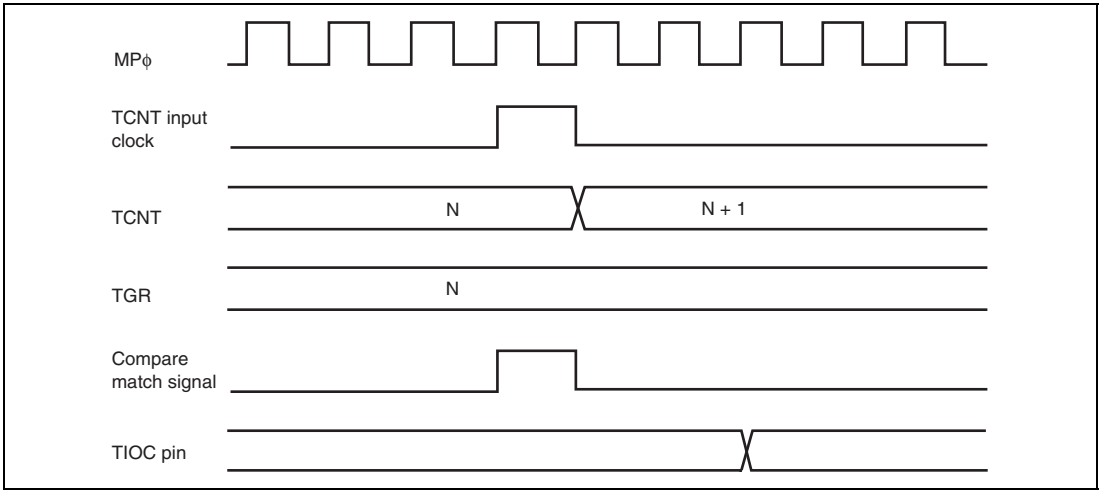


Figure 11.98 Output Compare Output Timing (Complementary PWM Mode/Reset Synchronous PWM Mode)

(3) Input Capture Signal Timing

Figure 11.99 shows input capture signal timing.

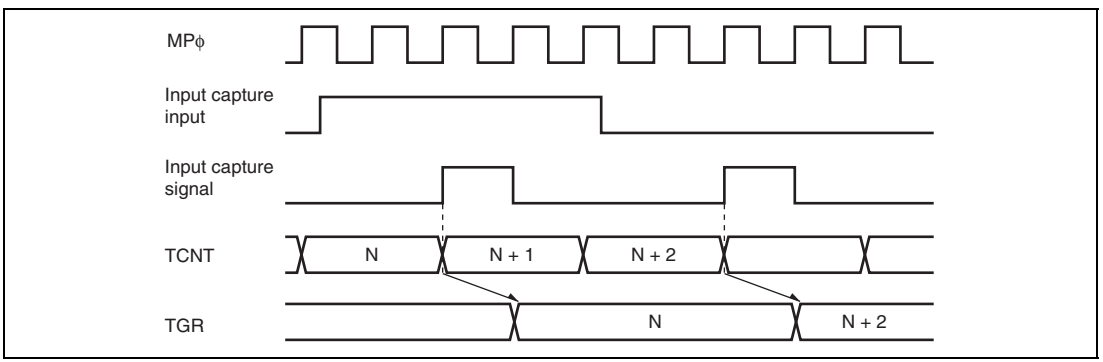


Figure 11.99 Input Capture Input Signal Timing

(4) Timing for Counter Clearing by Compare Match/Input Capture

Figures 11.100 and 11.101 show the timing when counter clearing on compare match is specified, and figure 11.102 shows the timing when counter clearing on input capture is specified.

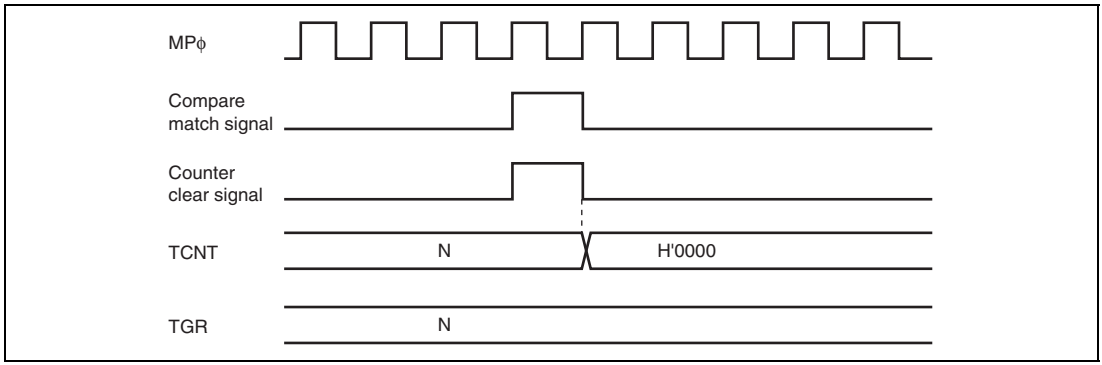


Figure 11.100 Counter Clear Timing (Compare Match) (Channels 0 to 4)

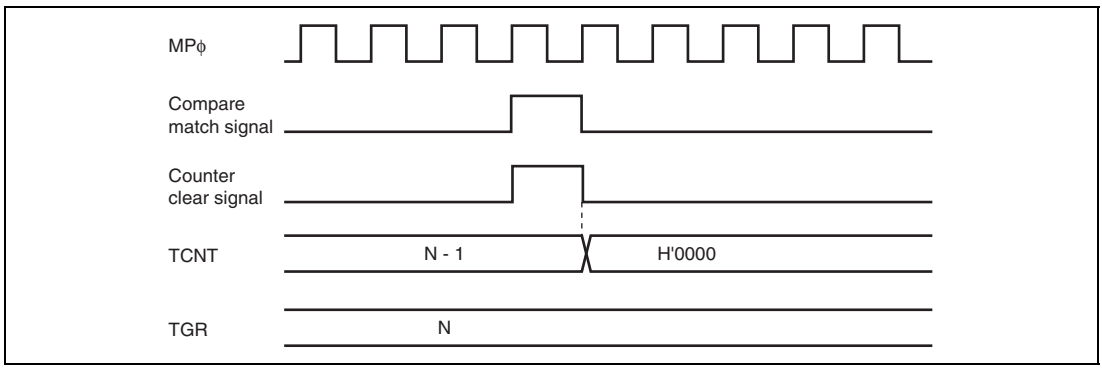


Figure 11.101 Counter Clear Timing (Compare Match) (Channel 5)

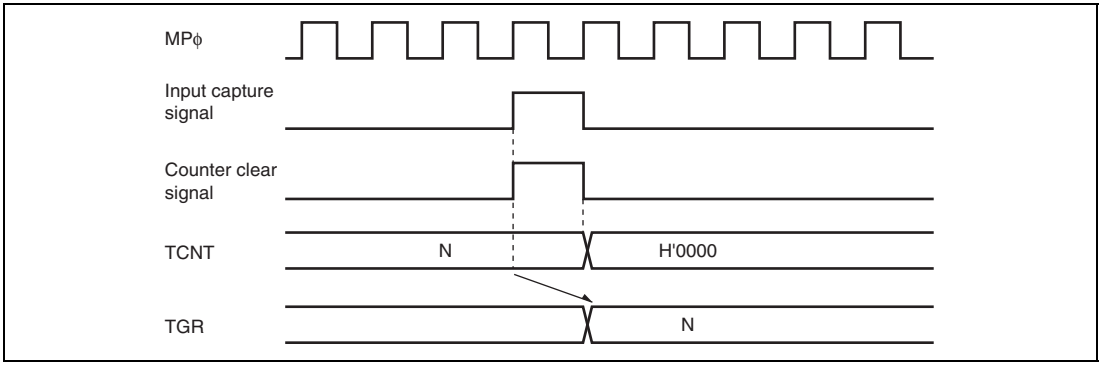


Figure 11.102 Counter Clear Timing (Input Capture) (Channels 0 to 5)

(5) Buffer Operation Timing

Figures 11.103 to 11.105 show the timing in buffer operation.

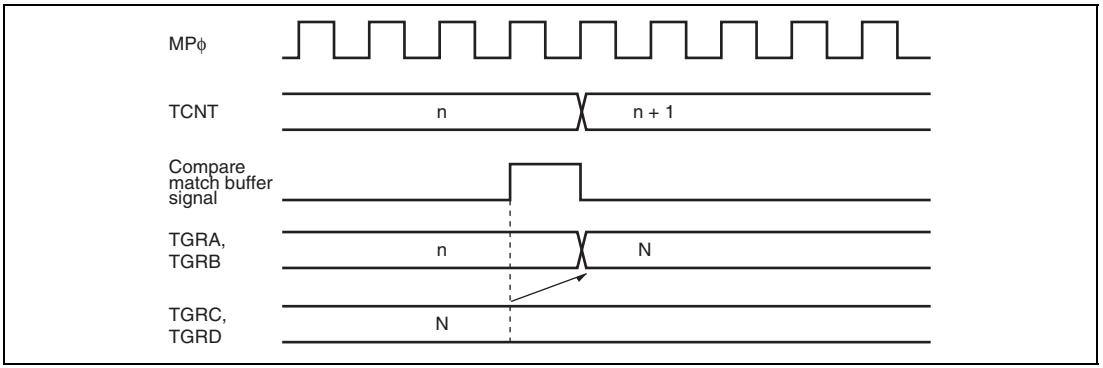


Figure 11.103 Buffer Operation Timing (Compare Match)

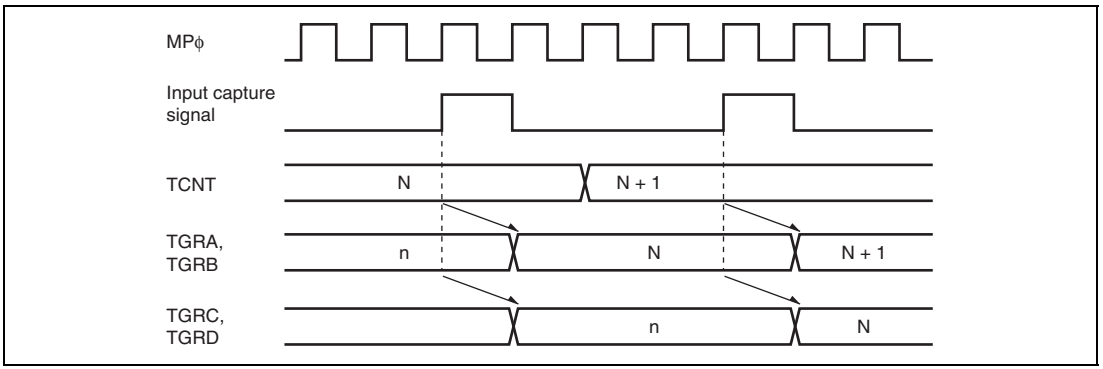


Figure 11.104 Buffer Operation Timing (Input Capture)

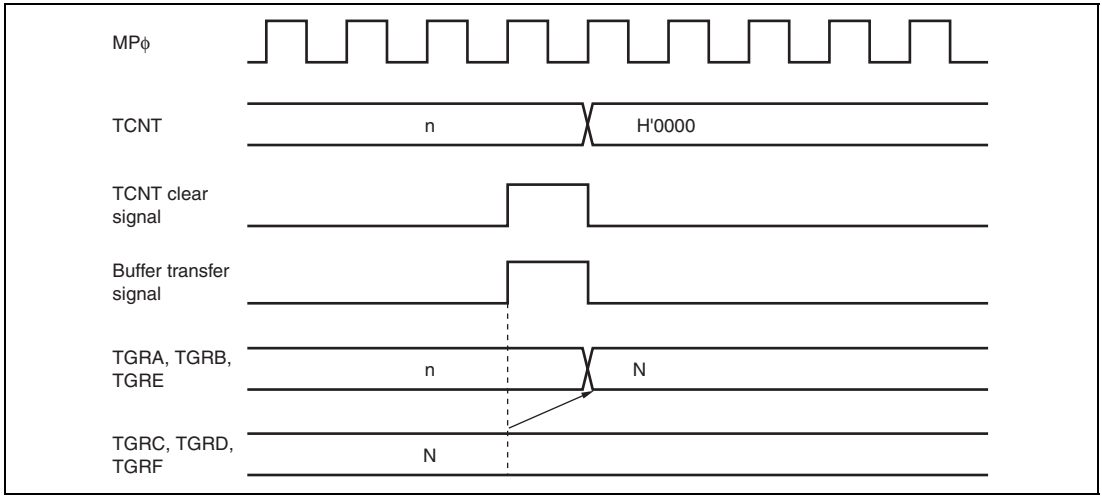


Figure 11.105 Buffer Transfer Timing (when TCNT Cleared)

(6) Buffer Transfer Timing (Complementary PWM Mode)

Figures 11.106 to 11.108 show the buffer transfer timing in complementary PWM mode.

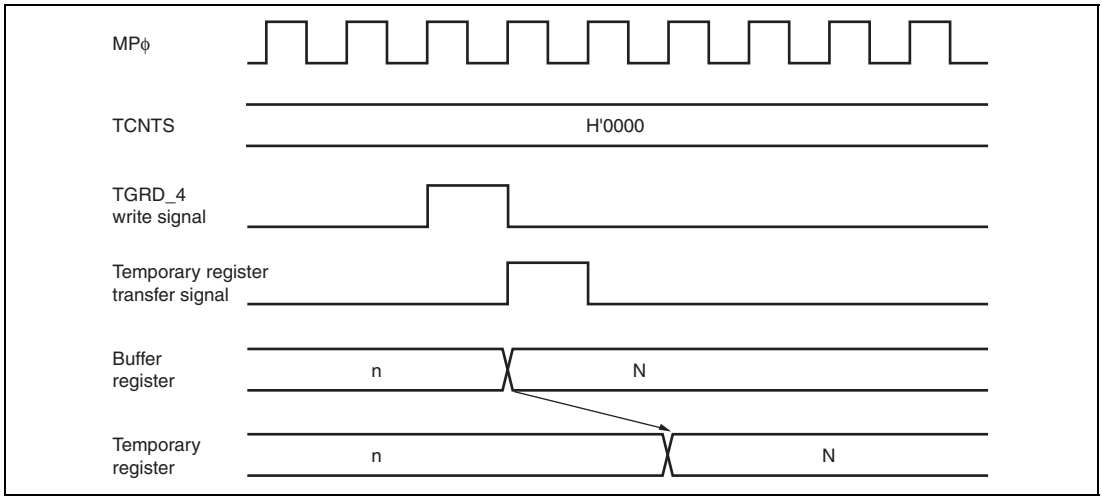


Figure 11.106 Transfer Timing from Buffer Register to Temporary Register (TCNTS Stop)

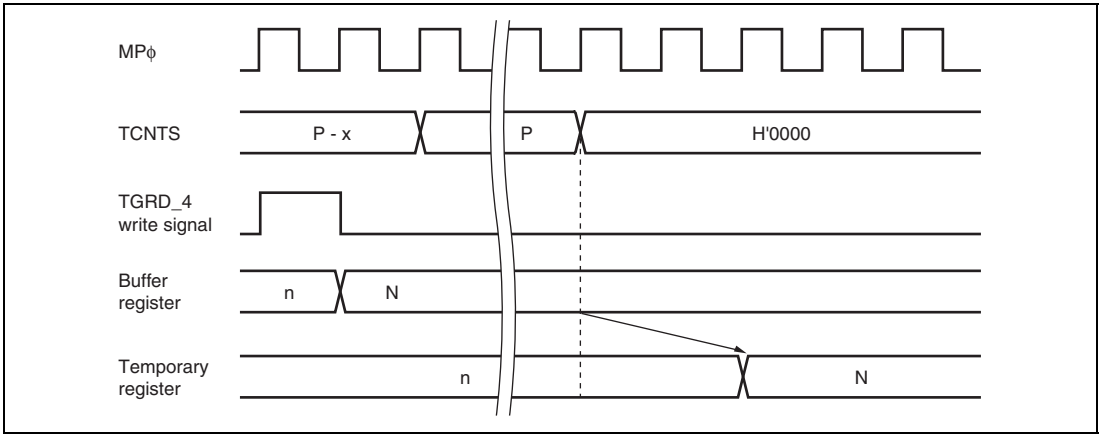


Figure 11.107 Transfer Timing from Buffer Register to Temporary Register (TCNTS Operating)

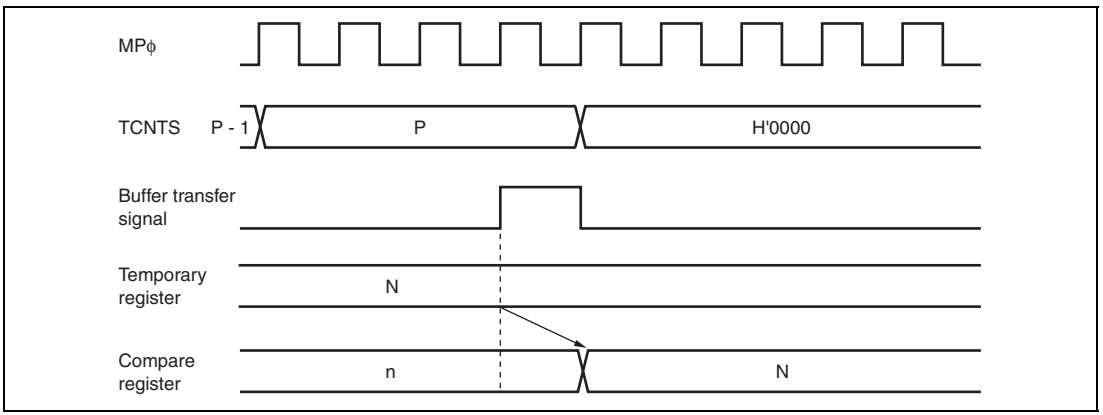


Figure 11.108 Transfer Timing from Temporary Register to Compare Register

11.6.2 Interrupt Signal Timing

(1) TGF Flag Setting Timing in Case of Compare Match

Figures 11.109 and 11.110 show the timing for setting of the TGF flag in TSR on compare match, and TGI interrupt request signal timing.

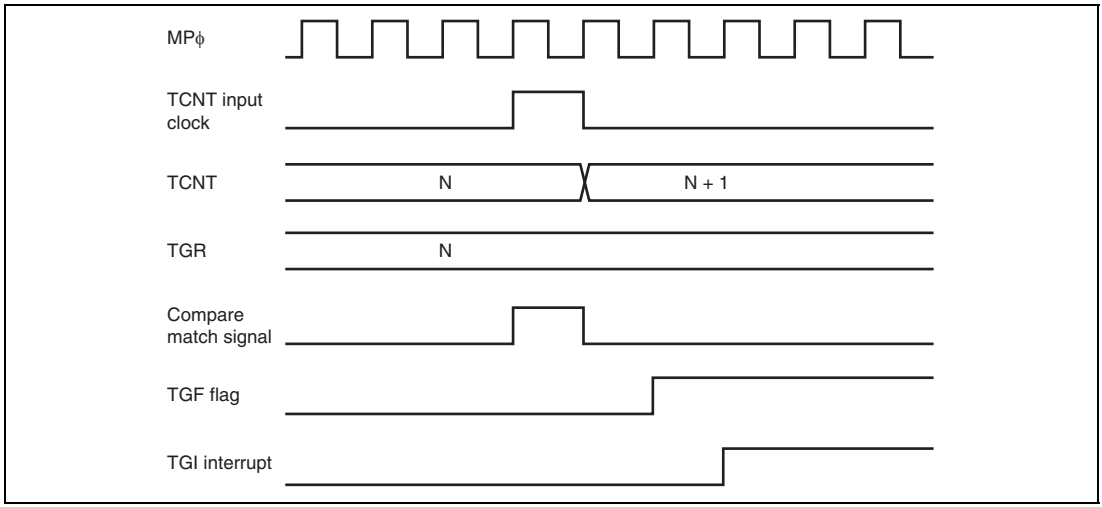


Figure 11.109 TGI Interrupt Timing (Compare Match) (Channels 0 to 4)

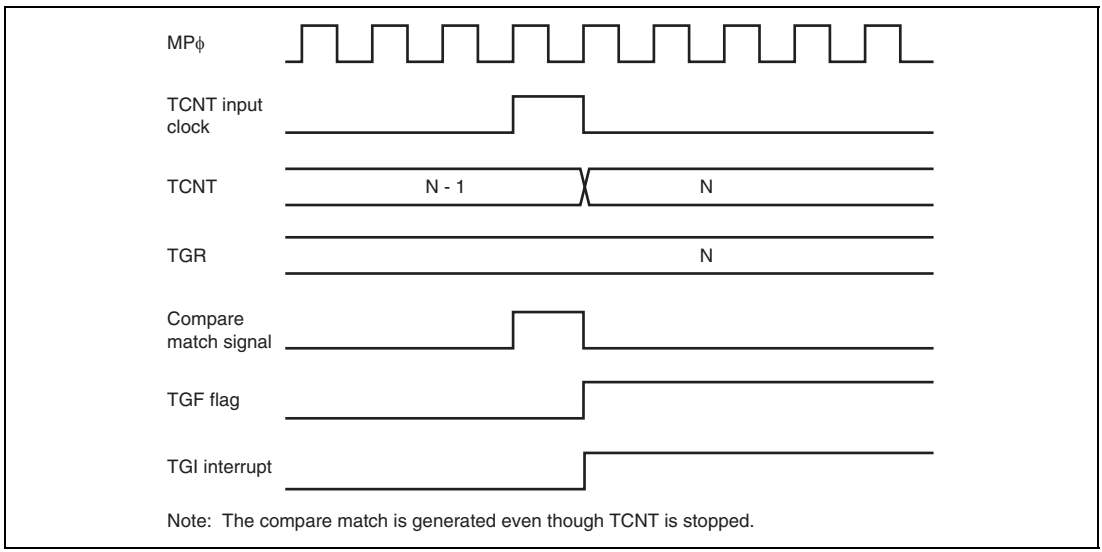


Figure 11.110 TGI Interrupt Timing (Compare Match) (Channel 5)

(2) TGF Flag Setting Timing in Case of Input Capture

Figures 11.111 and 11.112 show the timing for setting of the TGF flag in TSR on input capture, and TGI interrupt request signal timing.

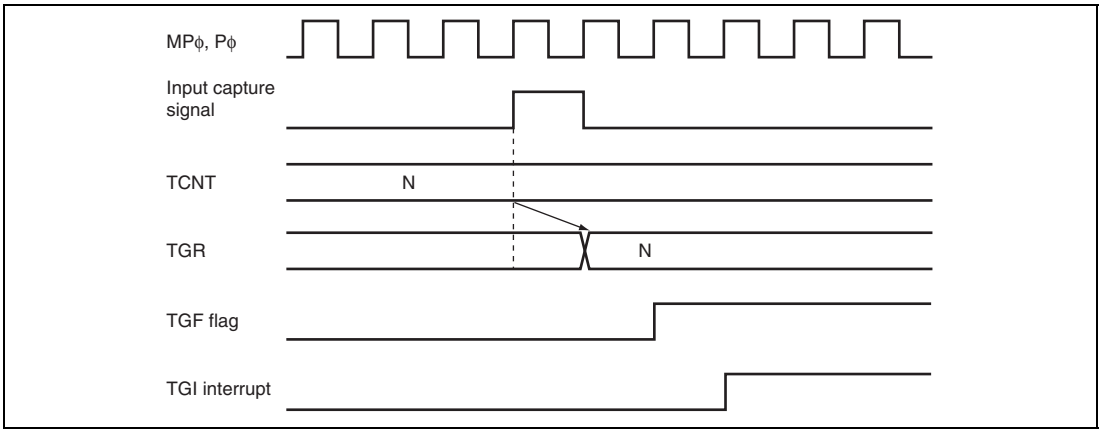


Figure 11.111 TGI Interrupt Timing (Input Capture) (Channels 0 to 4)

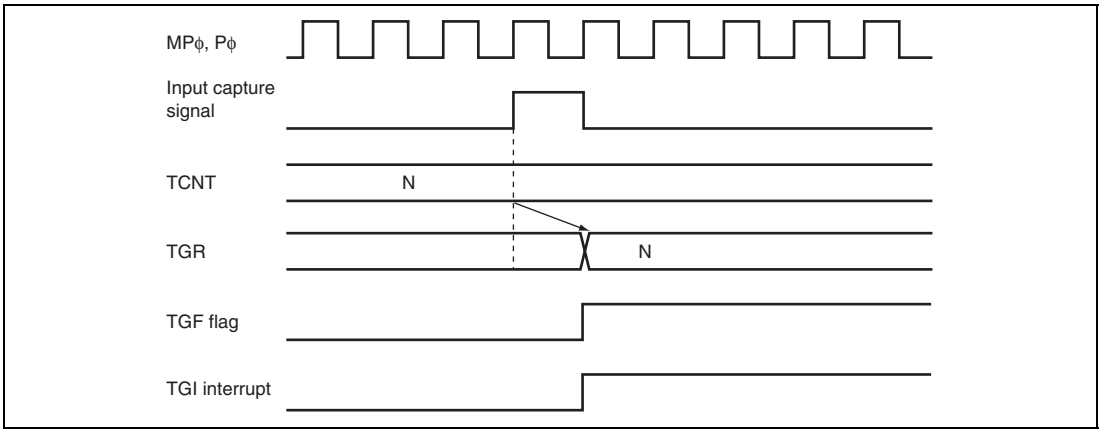


Figure 11.112 TGI Interrupt Timing (Input Capture) (Channel 5)

(3) TCFV Flag/TCFU Flag Setting Timing

Figure 11.113 shows the timing for setting of the TCFV flag in TSR on overflow, and TCIV interrupt request signal timing.

Figure 11.114 shows the timing for setting of the TCFU flag in TSR on underflow, and TCIU interrupt request signal timing.

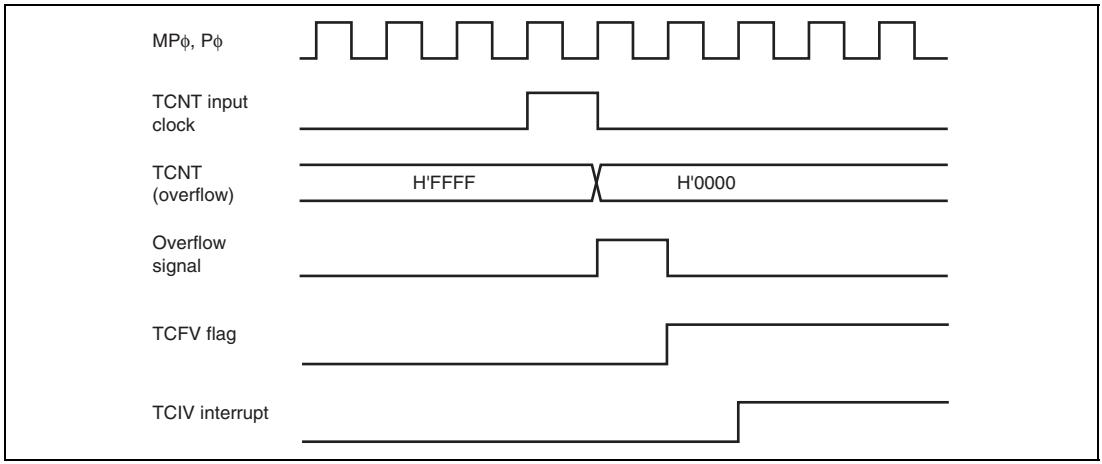


Figure 11.113 TCIV Interrupt Setting Timing

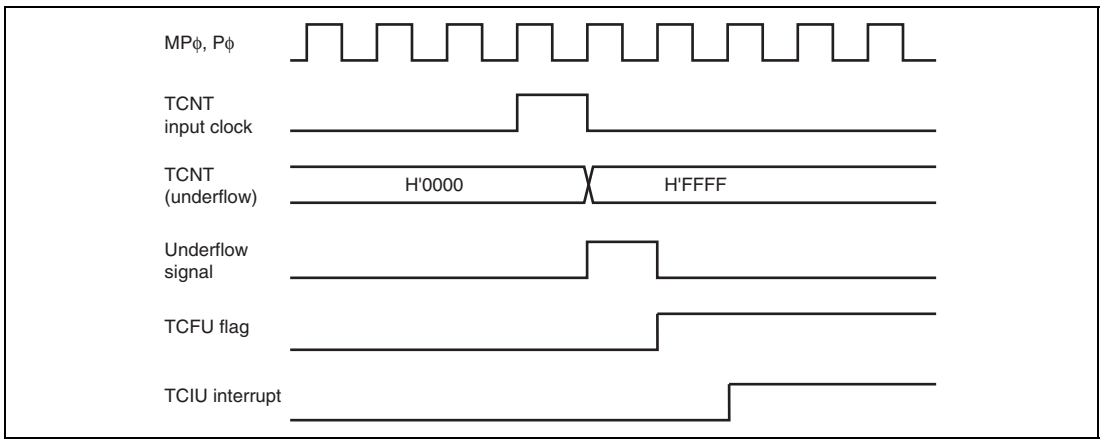


Figure 11.114 TCIU Interrupt Setting Timing

(4) Status Flag Clearing Timing

After a status flag is read as 1 by the CPU, it is cleared by writing 0 to it. When the DTC/DMAC is activated, the flag is cleared automatically. Figures 11.115 and 11.116 show the timing for status flag clearing by the CPU, and figures 11.117 to 11.119 show the timing for status flag clearing by the DTC/DMAC.

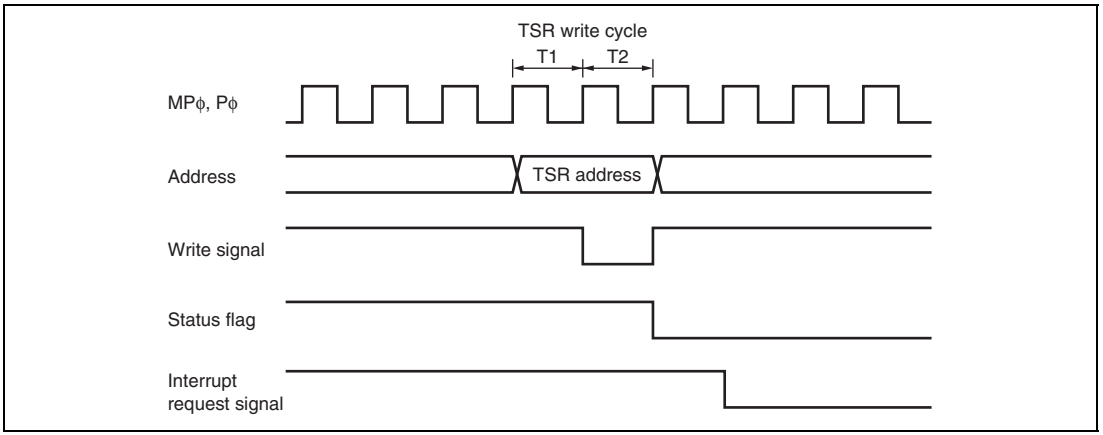


Figure 11.115 Timing for Status Flag Clearing by CPU (Channels 0 to 4)

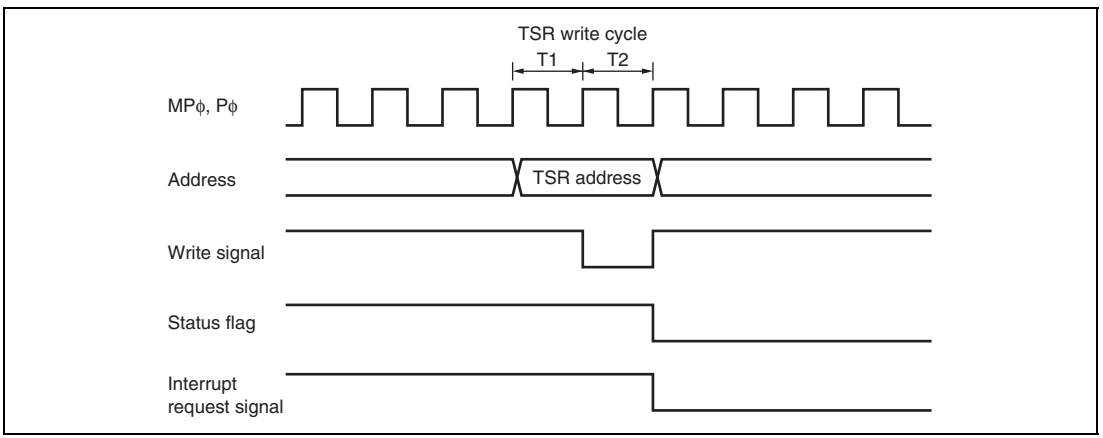


Figure 11.116 Timing for Status Flag Clearing by CPU (Channel 5)

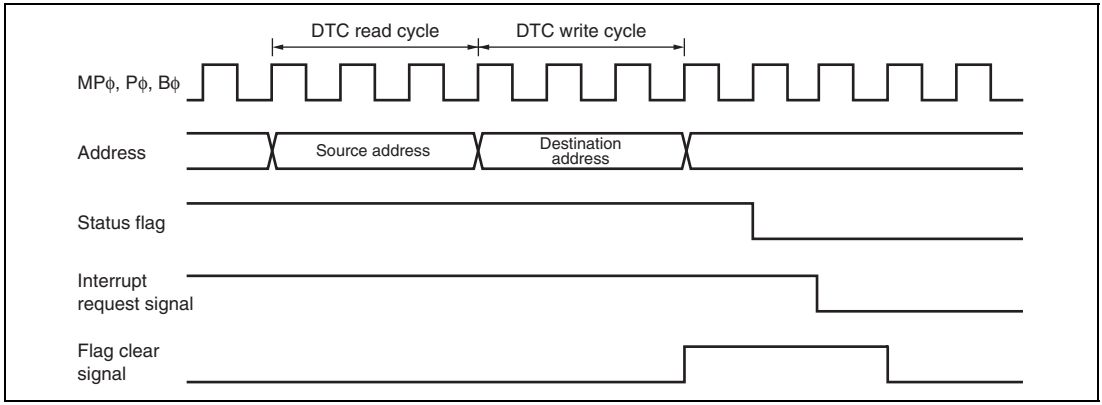


Figure 11.117 Timing for Status Flag Clearing by DTC Activation (Channels 0 to 4)

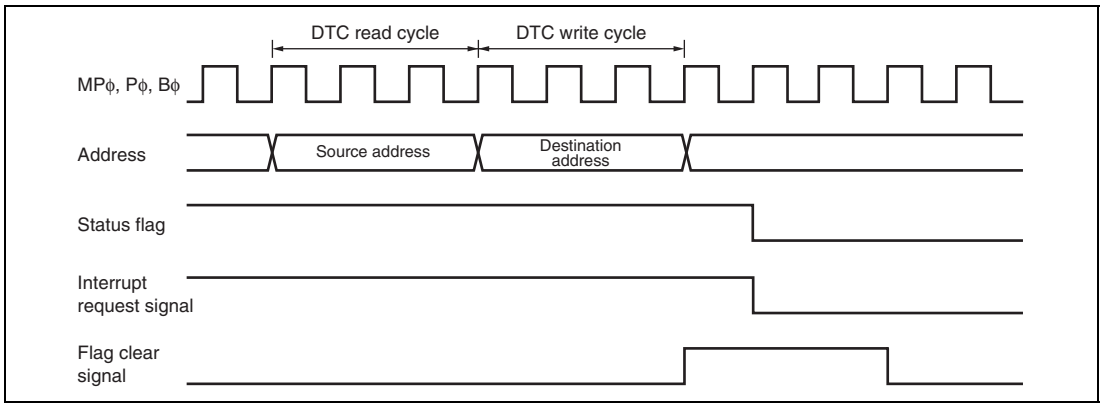


Figure 11.118 Timing for Status Flag Clearing by DTC Activation (Channel 5)

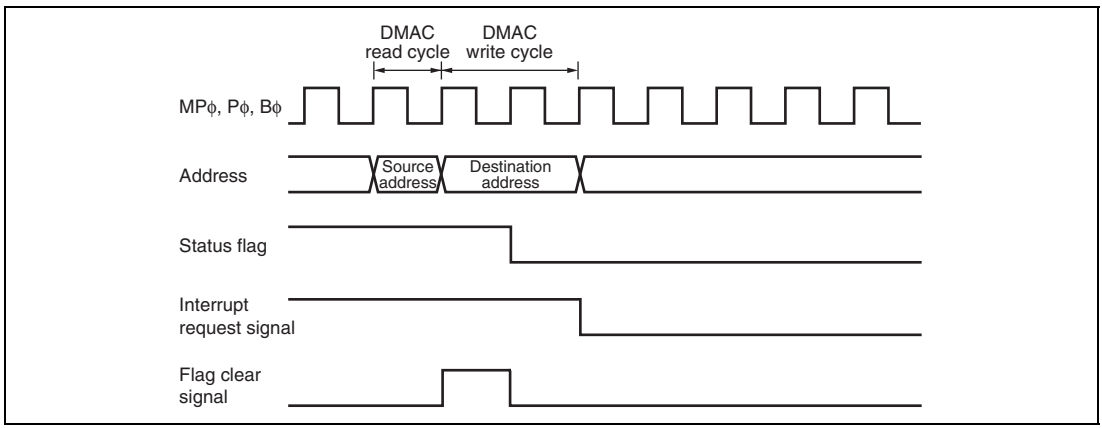


Figure 11.119 Timing for Status Flag Clearing by DMAC Activation

11.7 Usage Notes

11.7.1 Module Standby Mode Setting

MTU2 operation can be disabled or enabled using the standby control register. The initial setting is for MTU2 operation to be halted. Register access is enabled by clearing module standby mode. For details, refer to section 26, Power-Down Modes.

11.7.2 Input Clock Restrictions

The input clock pulse width must be at least 1.5 states in the case of single-edge detection, and at least 2.5 states in the case of both-edge detection. The MTU2 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 11.120 shows the input clock conditions in phase counting mode.

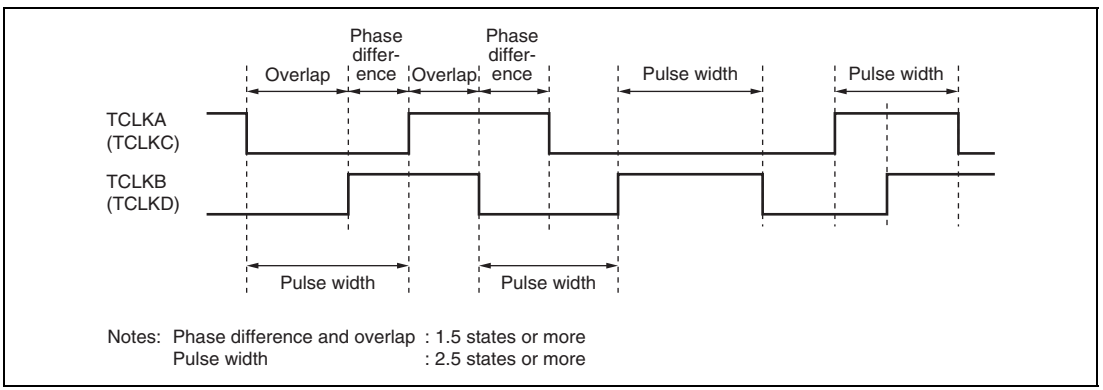


Figure 11.120 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

11.7.3 Caution on Period Setting

When counter clearing on compare match is set, TCNT is cleared in the final state in which it matches the TGR value (the point at which the count value matched by TCNT is updated). Consequently, the actual counter frequency is given by the following formula:

- Channels 0 to 4

$$f = \frac{MP\phi}{(N + 1)}$$

- Channel 5

$$f = \frac{MP\phi}{N}$$

Where f: Counter frequency
MPϕ: MTU2 peripheral clock operating frequency
N: TGR set value

11.7.4 Contention between TCNT Write and Clear Operations

If the counter clear signal is generated in the T2 state of a TCNT write cycle, TCNT clearing takes precedence and the TCNT write is not performed.

Figure 11.121 shows the timing in this case.

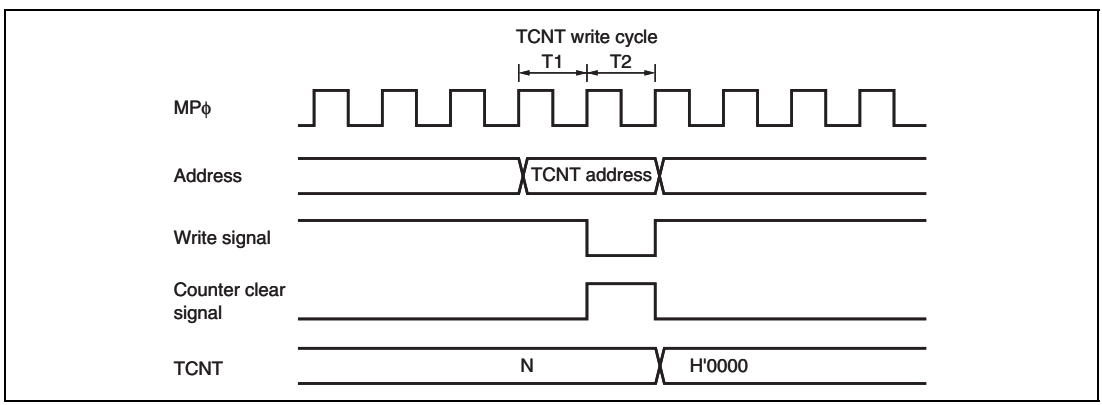


Figure 11.121 Contention between TCNT Write and Clear Operations

11.7.5 Contention between TCNT Write and Increment Operations

If incrementing occurs in the T2 state of a TCNT write cycle, the TCNT write takes precedence and TCNT is not incremented.

Figure 11.122 shows the timing in this case.

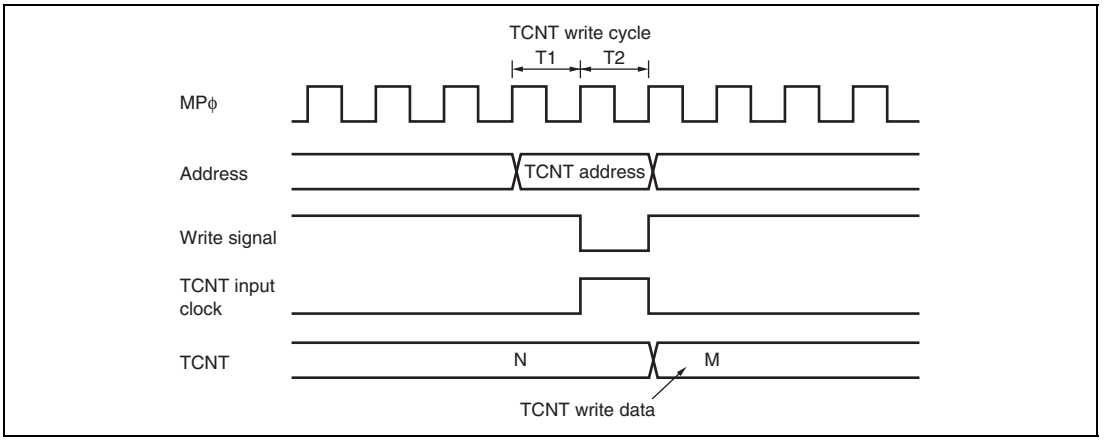


Figure 11.122 Contention between TCNT Write and Increment Operations

11.7.6 Contention between TGR Write and Compare Match

If a compare match occurs in the T2 state of a TGR write cycle, the TGR write is executed and the compare match signal is also generated.

Figure 11.123 shows the timing in this case.

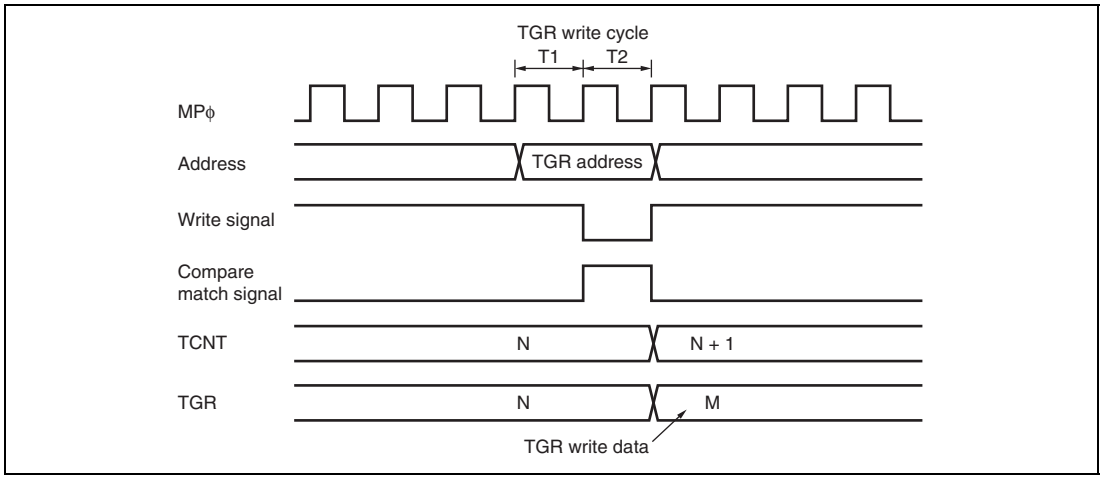


Figure 11.123 Contention between TGR Write and Compare Match

11.7.7 Contention between Buffer Register Write and Compare Match

If a compare match occurs in the T2 state of a TGR write cycle, the data that is transferred to TGR by the buffer operation is the data before write.

Figure 11.124 shows the timing in this case.

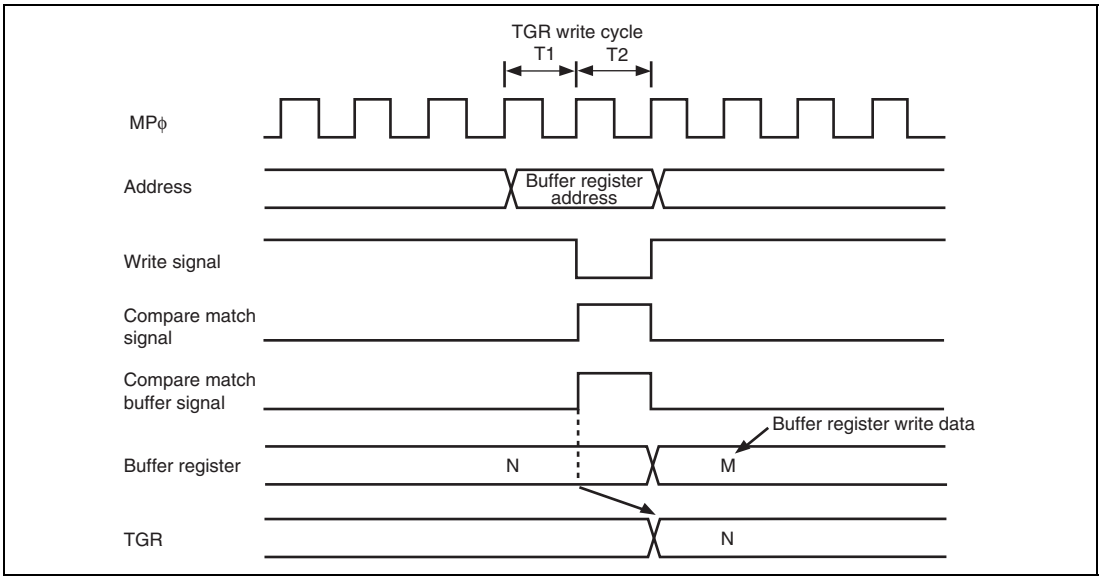


Figure 11.124 Contention between Buffer Register Write and Compare Match

11.7.8 Contention between Buffer Register Write and TCNT Clear

When the buffer transfer timing is set at the TCNT clear by the buffer transfer mode register (TBTM), if TCNT clear occurs in the T2 state of a TGR write cycle, the data that is transferred to TGR by the buffer operation is the data before write.

Figure 11.125 shows the timing in this case.

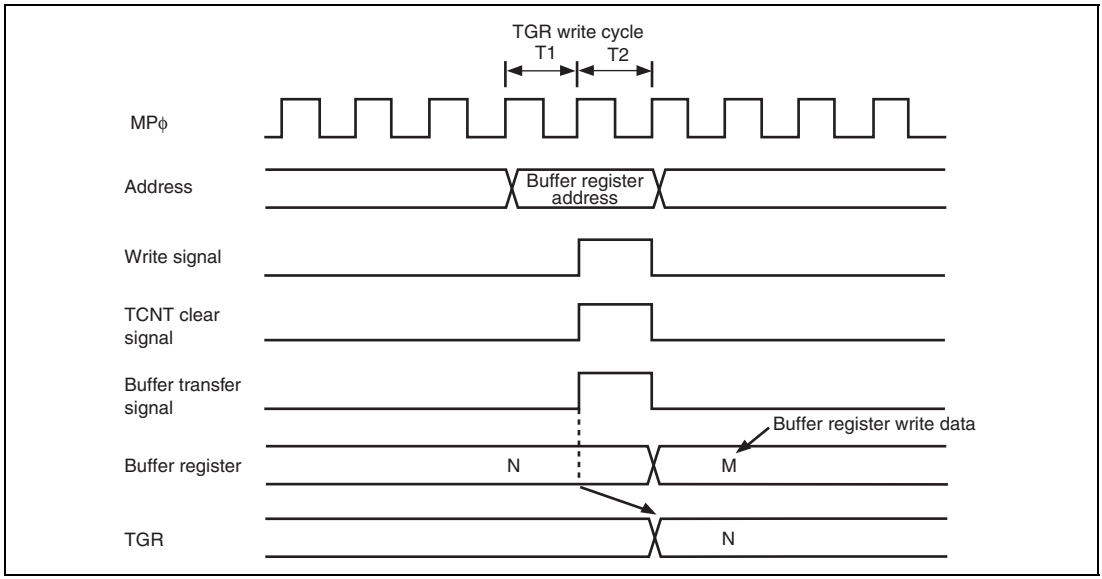


Figure 11.125 Contention between Buffer Register Write and TCNT Clear

11.7.9 Contention between TGR Read and Input Capture

If an input capture signal is generated in the T1 state of a TGR read cycle, the data that is read will be the data in the buffer before input capture transfer for channels 0 to 4, and the data after input capture transfer for channel 5.

Figures 11.126 and 11.127 show the timing in this case.

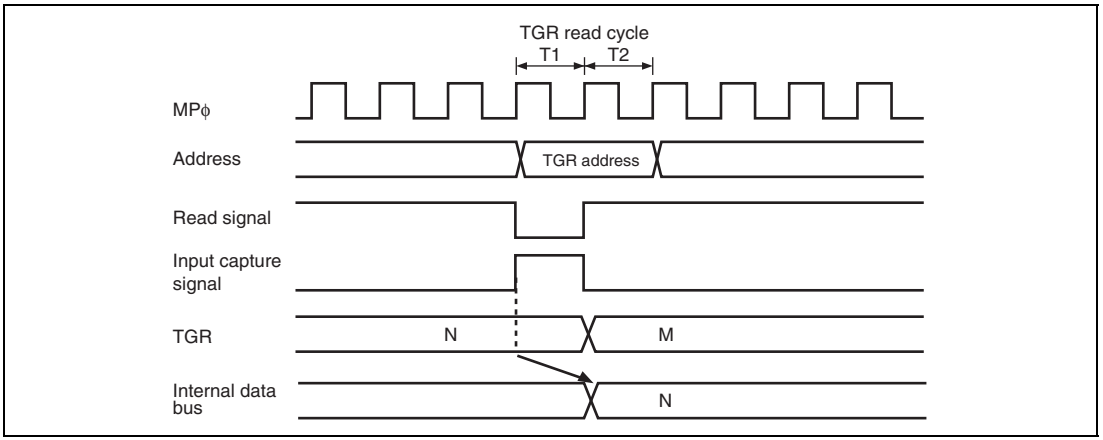


Figure 11.126 Contention between TGR Read and Input Capture (Channels 0 to 4)

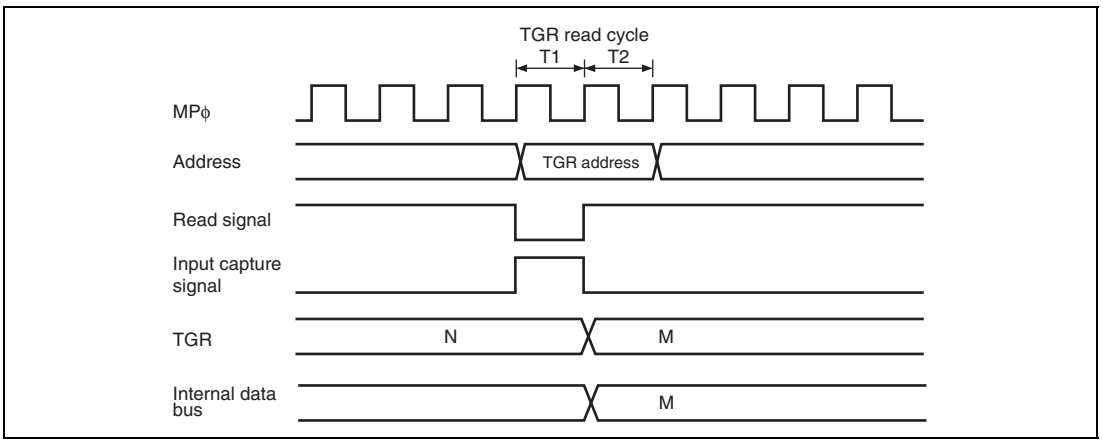


Figure 11.127 Contention between TGR Read and Input Capture (Channel 5)

11.7.10 Contention between TGR Write and Input Capture

If an input capture signal is generated in the T2 state of a TGR write cycle, the input capture operation takes precedence and the write to TGR is not performed for channels 0 to 4. For channel 5, write to TGR is performed and the input capture signal is generated.

Figures 11.128 and 11.129 show the timing in this case.

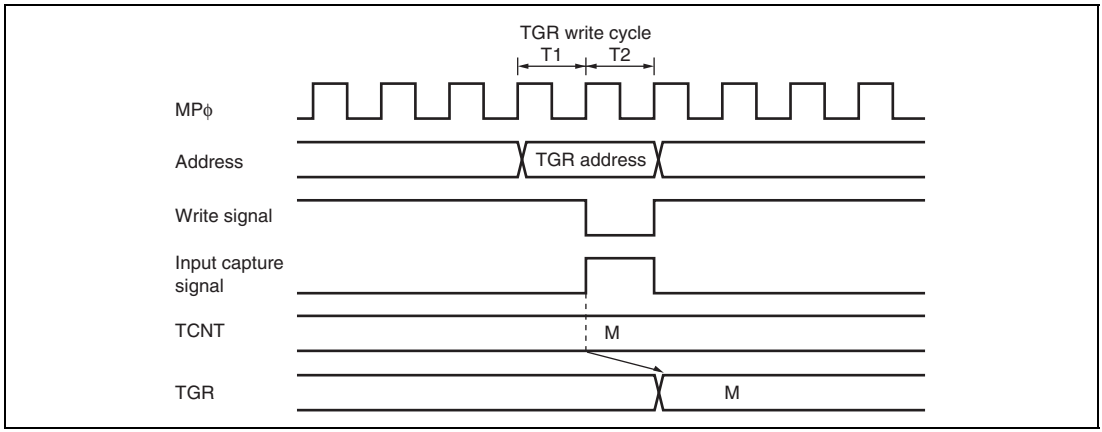


Figure 11.128 Contention between TGR Write and Input Capture (Channels 0 to 4)

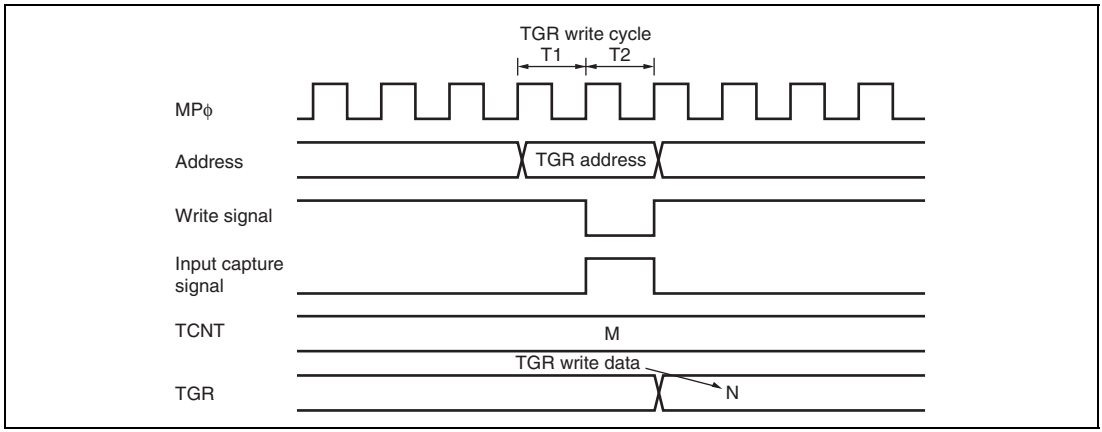


Figure 11.129 Contention between TGR Write and Input Capture (Channel 5)

11.7.11 Contention between Buffer Register Write and Input Capture

If an input capture signal is generated in the T2 state of a buffer register write cycle, the buffer operation takes precedence and the write to the buffer register is not performed.

Figure 11.130 shows the timing in this case.

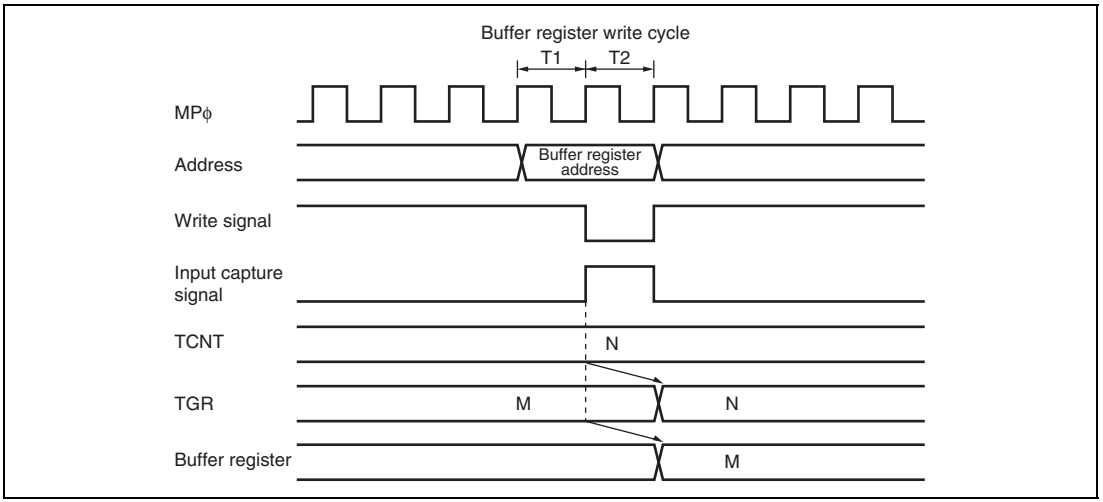


Figure 11.130 Contention between Buffer Register Write and Input Capture

11.7.12 TCNT_2 Write and Overflow/Underflow Contention in Cascade Connection

With timer counters TCNT_1 and TCNT_2 in a cascade connection, when a contention occurs during TCNT_1 count (during a TCNT_2 overflow/underflow) in the T2 state of the TCNT_2 write cycle, the write to TCNT_2 is conducted, and the TCNT_1 count signal is disabled. At this point, if there is match with TGRA_1 and the TCNT_1 value, a compare signal is issued. Furthermore, when the TCNT_1 count clock is selected as the input capture source of channel 0, TGRA_0 to TGRD_0 carry out the input capture operation. In addition, when the compare match/input capture is selected as the input capture source of TGRB_1, TGRB_1 carries out input capture operation. The timing is shown in figure 11.131.

For cascade connections, be sure to synchronize settings for channels 1 and 2 when setting TCNT clearing.

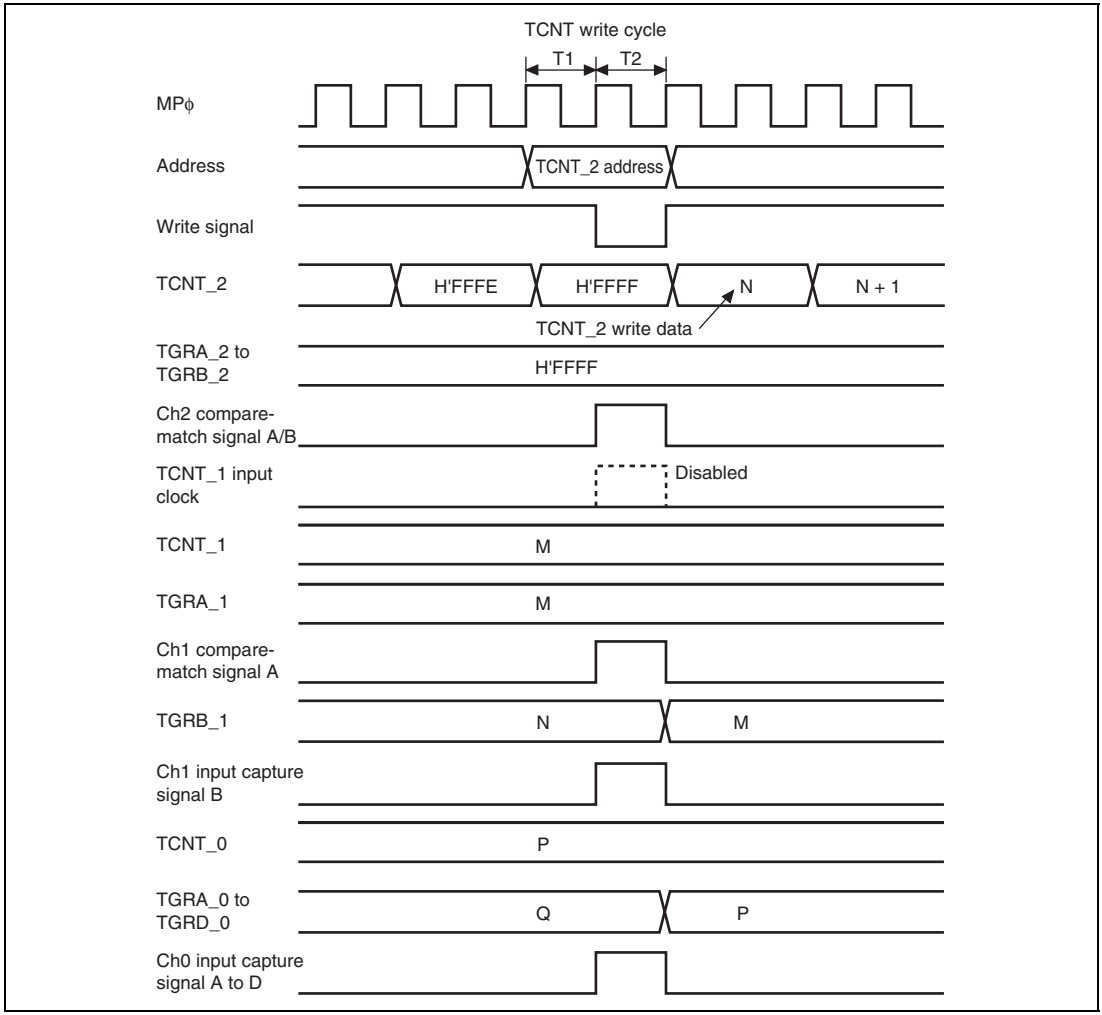


Figure 11.131 TCNT_2 Write and Overflow/Underflow Contention with Cascade Connection

11.7.13 Counter Value during Complementary PWM Mode Stop

When counting operation is suspended with TCNT_3 and TCNT_4 in complementary PWM mode, TCNT_3 has the timer dead time register (TDDR) value, and TCNT_4 is held at H'0000.

When restarting complementary PWM mode, counting begins automatically from the initialized state. This explanatory diagram is shown in figure 11.132.

When counting begins in another operating mode, be sure that TCNT_3 and TCNT_4 are set to the initial values.

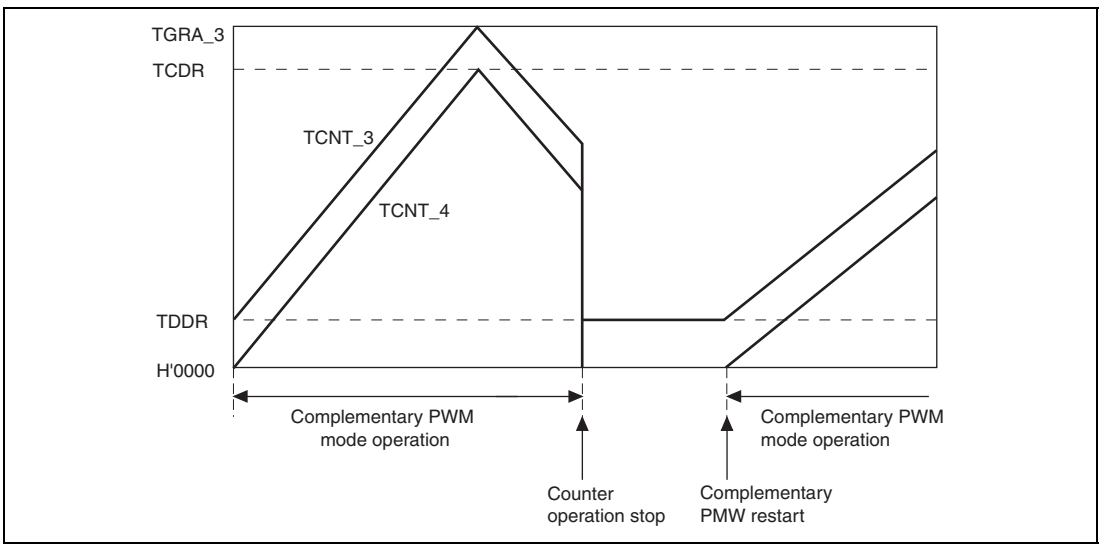


Figure 11.132 Counter Value during Complementary PWM Mode Stop

11.7.14 Buffer Operation Setting in Complementary PWM Mode

In complementary PWM mode, conduct rewrites by buffer operation for the PWM cycle setting register (TGRA_3), timer cycle data register (TCDR), and duty setting registers (TGRB_3, TGRA_4, and TGRB_4).

In complementary PWM mode, channel 3 and channel 4 buffers operate in accordance with bit settings BFA and BFB of TMDR_3. When TMDR_3's BFA bit is set to 1, TGRC_3 functions as a buffer register for TGRA_3. At the same time, TGRC_4 functions as the buffer register for TGRA_4, and TCBR functions as the TCDR's buffer register.

11.7.15 Reset Sync PWM Mode Buffer Operation and Compare Match Flag

When setting buffer operation for reset sync PWM mode, set the BFA and BFB bits in TMDR_4 to 0. The TIOC4C pin will be unable to produce its waveform output if the BFA bit in TMDR_4 is set to 1.

In reset sync PWM mode, the channel 3 and channel 4 buffers operate in accordance with the BFA and BFB bit settings of TMDR_3. For example, if the BFA bit in TMDR_3 is set to 1, TGRC_3 functions as the buffer register for TGRA_3. At the same time, TGRC_4 functions as the buffer register for TGRA_4.

The TGFC bit and TGFD bit in TSR_3 and TSR_4 are not set when TGRC_3 and TGRD_3 are operating as buffer registers.

Figure 11.133 shows an example of operations for TGR_3, TGR_4, TIOC3, and TIOC4, with TMDR_3's BFA and BFB bits set to 1, and TMDR_4's BFA and BFB bits set to 0.

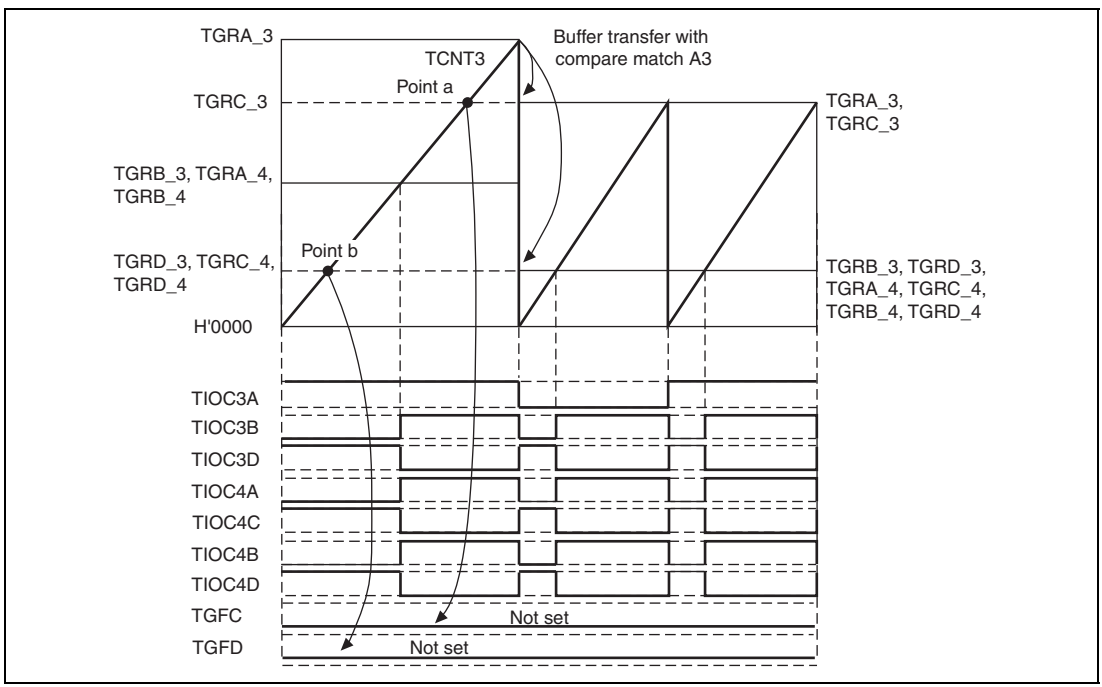


Figure 11.133 Buffer Operation and Compare-Match Flags in Reset Synchronous PWM Mode

11.7.16 Overflow Flags in Reset Synchronous PWM Mode

When set to reset synchronous PWM mode, TCNT_3 and TCNT_4 start counting when the CST3 bit of TSTR is set to 1. At this point, TCNT_4's count clock source and count edge obey the TCR_3 setting.

In reset synchronous PWM mode, with cycle register TGRA_3's set value at H'FFFF, when specifying TGR3A compare-match for the counter clear source, TCNT_3 and TCNT_4 count up to H'FFFF, then a compare-match occurs with TGRA_3, and TCNT_3 and TCNT_4 are both cleared. At this point, TSR's overflow flag TCFV bit is not set.

Figure 11.134 shows a TCFV bit operation example in reset synchronous PWM mode with a set value for cycle register TGRA_3 of H'FFFF, when a TGRA_3 compare-match has been specified without synchronous setting for the counter clear source.

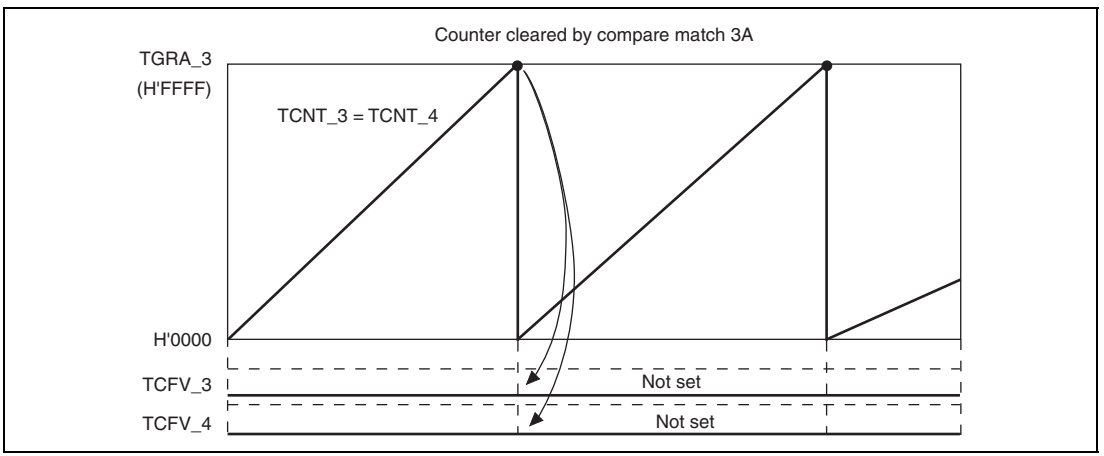


Figure 11.134 Reset Synchronous PWM Mode Overflow Flag

11.7.17 Contention between Overflow/Underflow and Counter Clearing

If overflow/underflow and counter clearing occur simultaneously, the TCFV/TCFU flag in TSR is not set and TCNT clearing takes precedence.

Figure 11.135 shows the operation timing when a TGR compare match is specified as the clearing source, and when H'FFFF is set in TGR.

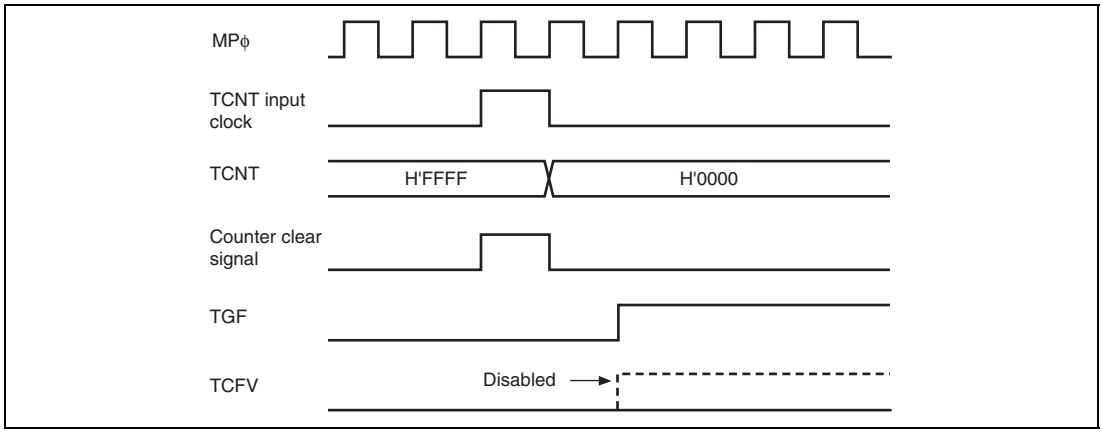


Figure 11.135 Contention between Overflow and Counter Clearing

11.7.18 Contention between TCNT Write and Overflow/Underflow

If there is an up-count or down-count in the T2 state of a TCNT write cycle, and overflow/underflow occurs, the TCNT write takes precedence and the TCFV/TCFU flag in TSR is not set.

Figure 11.136 shows the operation timing when there is contention between TCNT write and overflow.

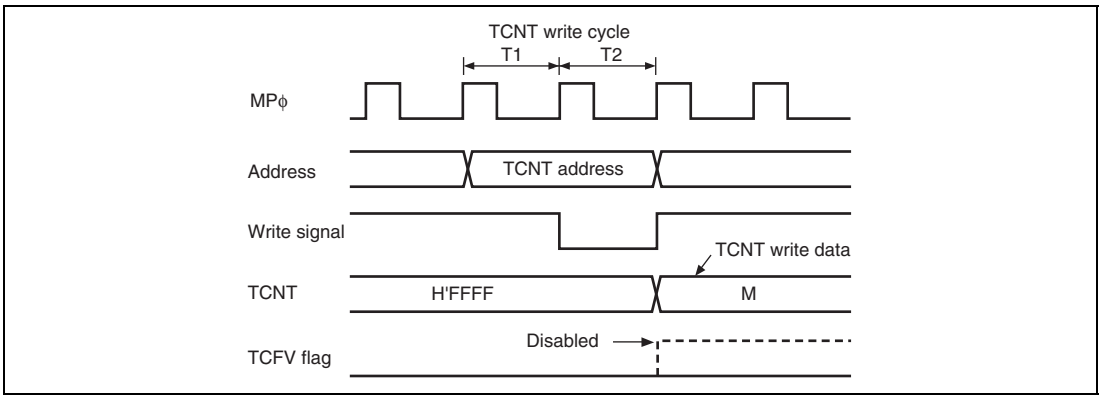


Figure 11.136 Contention between TCNT Write and Overflow

11.7.19 Cautions on Transition from Normal Operation or PWM Mode 1 to Reset-Synchronized PWM Mode

When making a transition from channel 3 or 4 normal operation or PWM mode 1 to reset-synchronized PWM mode, if the counter is halted with the output pins (TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, TIOC4D) in the high-level state, followed by the transition to reset-synchronized PWM mode and operation in that mode, the initial pin output will not be correct.

When making a transition from normal operation to reset-synchronized PWM mode, write H'11 to registers TIORH_3, TIORL_3, TIORH_4, and TIORL_4 to initialize the output pins to low level output, then set an initial register value of H'00 before making the mode transition.

When making a transition from PWM mode 1 to reset-synchronized PWM mode, first switch to normal operation, then initialize the output pins to low level output and set an initial register value of H'00 before making the transition to reset-synchronized PWM mode.

11.7.20 Output Level in Complementary PWM Mode and Reset-Synchronized PWM Mode

When channels 3 and 4 are in complementary PWM mode or reset-synchronized PWM mode, the PWM waveform output level is set with the OLSP and OLSN bits in the timer output control register (TOCR). In the case of complementary PWM mode or reset-synchronized PWM mode, TIOR should be set to H'00.

11.7.21 Interrupts in Module Standby Mode

If module standby mode is entered when an interrupt has been requested, it will not be possible to clear the CPU interrupt source or the DTC/DMAC activation source. Interrupts should therefore be disabled before entering module standby mode.

11.7.22 Simultaneous Capture of TCNT_1 and TCNT_2 in Cascade Connection

When timer counters 1 and 2 (TCNT_1 and TCNT_2) are operated as a 32-bit counter in cascade connection, the cascade counter value cannot be captured successfully even if input-capture input is simultaneously done to TIOC1A and TIOC2A or to TIOC1B and TIOC2B. This is because the input timing of TIOC1A and TIOC2A or of TIOC1B and TIOC2B may not be the same when external input-capture signals to be input into TCNT_1 and TCNT_2 are taken in synchronization with the internal clock. For example, TCNT_1 (the counter for upper 16 bits) does not capture the count-up value by overflow from TCNT_2 (the counter for lower 16 bits) but captures the count value before the count-up. In this case, the values of TCNT_1 = H'FFF1 and TCNT_2 = H'0000 should be transferred to TGRA_1 and TGRA_2 or to TGRB_1 and TGRB_2, but the values of TCNT_1 = H'FFF0 and TCNT_2 = H'0000 are erroneously transferred.

The MTU2 has a new function that allows simultaneous capture of TCNT_1 and TCNT_2 with a single input-capture input as the trigger. This function allows reading of the 32-bit counter such that TCNT_1 and TCNT_2 are captured at the same time. For details, see section, 11.3.8, Timer Input Capture Control Register (TICCR).

11.7.23 Notes on Output Waveform Control During Synchronous Counter Clearing in Complementary PWM Mode

In complementary PWM mode, when output waveform control during synchronous counter clearing is enabled (WRE in the TWCR register set to 1), the following problems may occur when condition (1) or condition (2), below, is satisfied.

- Dead time for the PWM output pins may be too short (or nonexistent).
- Active-level output from the PWM negative-phase pins may occur outside the correct active-level output interval

Condition (1): When synchronous clearing occurs in the PWM output dead time interval within initial output suppression interval (10) (figure 11.137).

Condition (2): When synchronous clearing occurs within initial output suppression interval (10) or (11) and TGRB_3 ≤ TDDR, TGRA_4 ≤ TDDR, or TGRB_4 ≤ TDDR is true (figure 11.138).

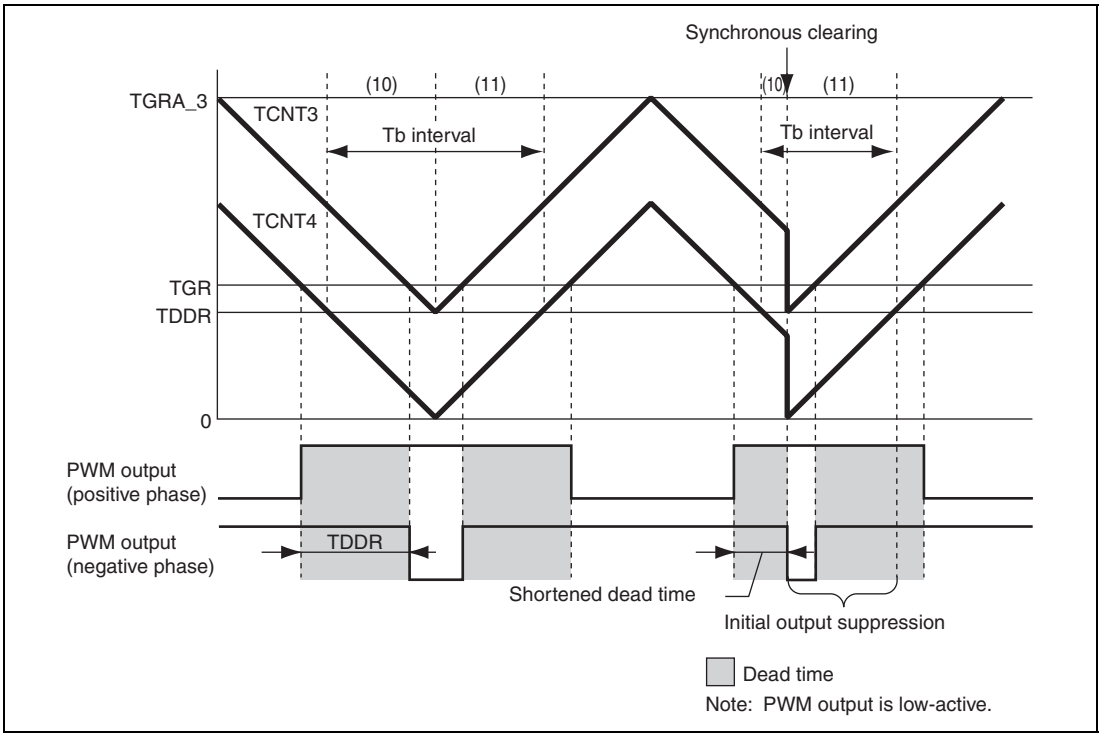


Figure 11.137 Condition (1) Synchronous Clearing Example

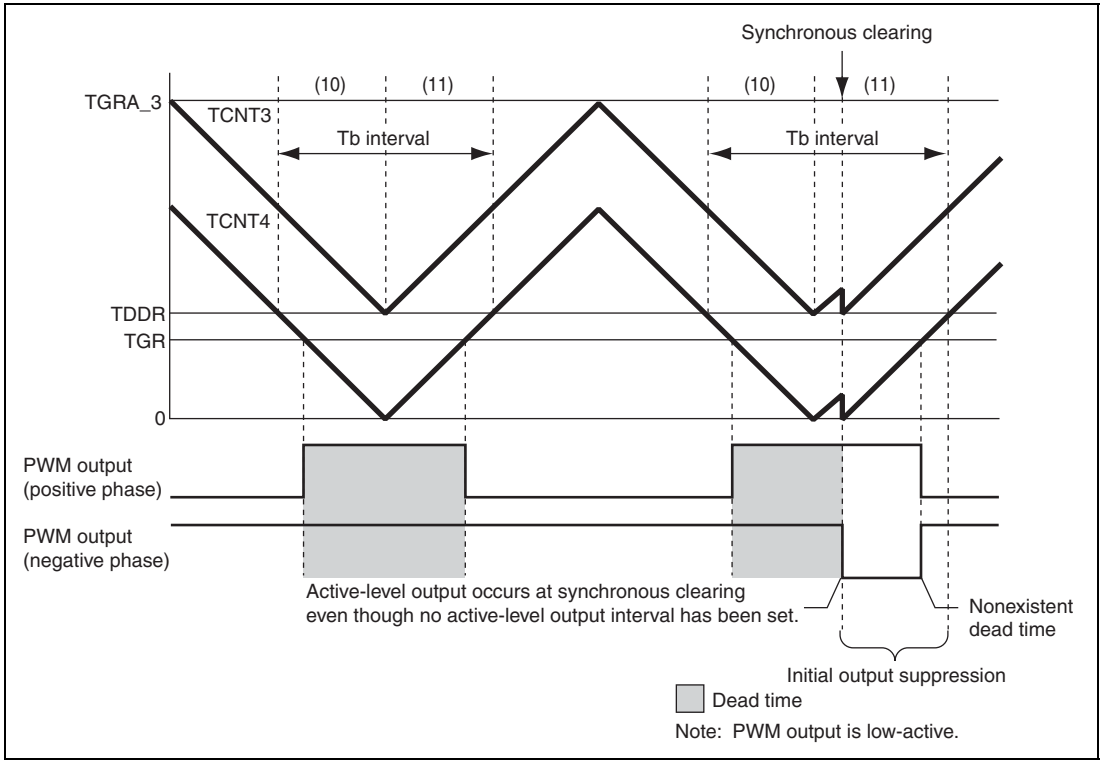


Figure 11.138 Condition (2) Synchronous Clearing Example

The following workaround can be used to avoid these problems.

When using synchronous clearing, make sure to set compare registers TGRB_3, TGRA_4, and TGRB_4 to a value twice or more the setting of dead time data register TDDR.

11.7.24 Notes on Using the A/D Converter Start Request Delaying Function in Complementary PWM Mode

- When TADCOBRA_4/TADCOBRB_4 is set to 0 and the UT4AE or UT4BE bit in TADCR is set to 1, and then buffer transfer takes place at the trough of TCNT_4, no A/D converter start request occurs during the up-counting interval immediately after the transfer (figure 11.139).
- When the value of TADCOBRA_4/TADCOBRB_4 is the same as that of TCDR and the DT4AE or DT4BE bit in TADCR is set to 1, and then buffer transfer takes place at the crest of TCNT_4, no A/D converter start request occurs during the down-counting interval immediately after the transfer (figure 11.140).

- When A/D converter start requests are linked to the interrupt skipping function, set TADCORA_4/TADCORB_4 such that the condition of 2 ≤ TADCORA_4/TADCORB_4 ≤ TCDR – 2 is met.

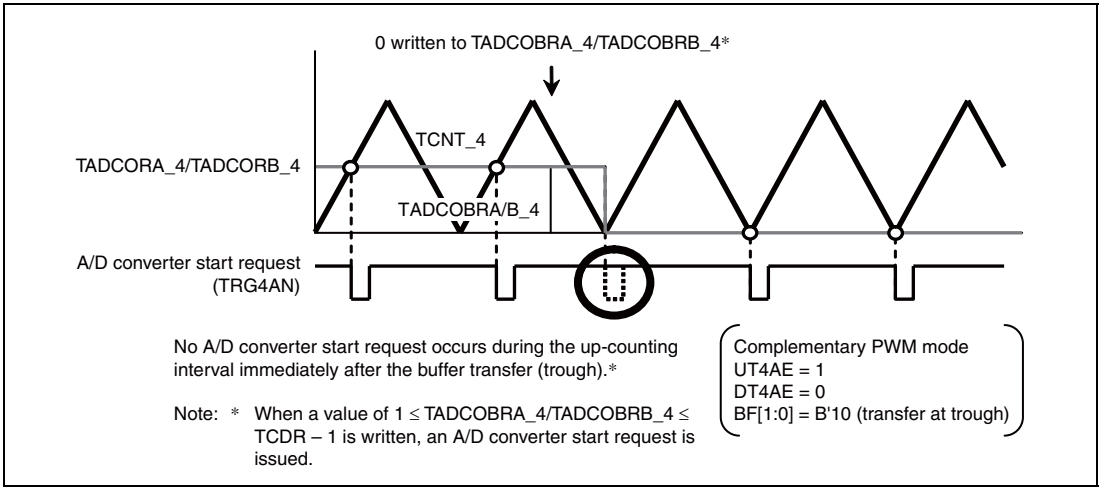


Figure 11.139 A/D Converter Start Request Operation when 0 Is Written to TADCOBRA_4/TADCOBRB_4

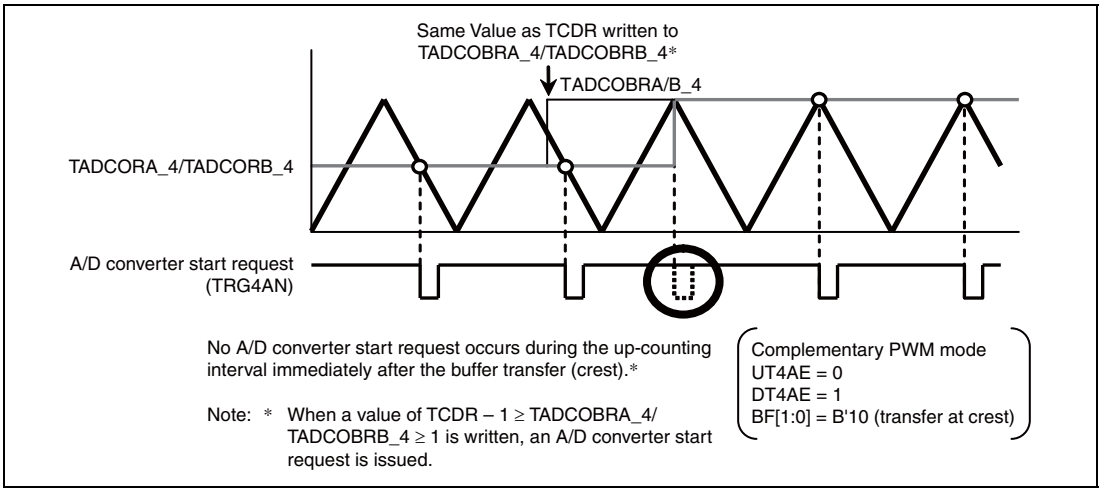


Figure 11.140 A/D Converter Start Request Operation when Same Value as TCDR Is Written to TADCOBRA_4/TADCOBRB_4

11.8 MTU2 Output Pin Initialization

11.8.1 Operating Modes

The MTU2 has the following six operating modes. Waveform output is possible in all of these modes.

- Normal mode (channels 0 to 4)
- PWM mode 1 (channels 0 to 4)
- PWM mode 2 (channels 0 to 2)
- Phase counting modes 1 to 4 (channels 1 and 2)
- Complementary PWM mode (channels 3 and 4)
- Reset-synchronized PWM mode (channels 3 and 4)

The MTU2 output pin initialization method for each of these modes is described in this section.

11.8.2 Reset Start Operation

The MTU2 output pins (TIOC*) are initialized low by a reset and in standby mode. Since MTU2 pin function selection is performed by the pin function controller (PFC), when the PFC is set, the MTU2 pin states at that point are output to the ports. When MTU2 output is selected by the PFC immediately after a reset, the MTU2 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 PFC setting should be made after initialization of the MTU2 output pins is completed.

Note: Channel number and port notation are substituted for *.

11.8.3 Operation in Case of Re-Setting Due to Error During Operation, etc.

If an error occurs during MTU2 operation, MTU2 output should be cut by the system. Cutoff is performed by switching the pin output to port output with the PFC and outputting the inverse of the active level. For large-current pins, output can also be cut by hardware, using port output enable (POE). 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 MTU2 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 11.59.

Table 11.59 Mode Transition Combinations

	After					
Before	**Normal**	**PWM1**	**PWM2**	**PCM**	**CPWM**	**RPWM**
Normal	(1)	(2)	(3)	(4)	(5)	(6)
PWM1	(7)	(8)	(9)	(10)	(11)	(12)
PWM2	(13)	(14)	(15)	(16)	None	None
PCM	(17)	(18)	(19)	(20)	None	None
CPWM	(21)	(22)	None	None	(23) (24)	(25)
RPWM	(26)	(27)	None	None	(28)	(29)

[Legend]
Normal: Normal mode
PWM1: PWM mode 1
PWM2: PWM mode 2
PCM: Phase counting modes 1 to 4
CPWM: Complementary PWM mode
RPWM: Reset-synchronized PWM mode

11.8.4 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation, etc.

- When making a transition to a mode (Normal, PWM1, PWM2, PCM) in which the pin output level is selected by the timer I/O control register (TIOR) setting, initialize the pins by means of a TIOR setting.
- In PWM mode 1, since a waveform is not output to the TIOC*B (TIOC *D) pin, setting TIOR will not initialize the pins. If initialization is required, carry it out in normal mode, then switch to PWM mode 1.
- In PWM mode 2, since a waveform is not output to the cycle register pin, setting TIOR will not initialize the pins. If initialization is required, carry it out in normal mode, then switch to PWM mode 2.
- In normal mode or PWM mode 2, if TGRC and TGRD operate as buffer registers, setting TIOR will not initialize the buffer register pins. If initialization is required, clear buffer mode, carry out initialization, then set buffer mode again.
- In PWM mode 1, if either TGRC or TGRD operates as a buffer register, setting TIOR will not initialize the TGRC pin. To initialize the TGRC pin, clear buffer mode, carry out initialization, then set buffer mode again.
- When making a transition to a mode (CPWM, RPWM) in which the pin output level is selected by the timer output control register (TOCR) setting, switch to normal mode and perform initialization with TIOR, then restore TIOR to its initial value, and temporarily disable channel 3 and 4 output with the timer output master enable register (TOER). Then operate the unit in accordance with the mode setting procedure (TOCR setting, TMDR setting, TOER setting).

Note: Channel number is substituted for * indicated in this article.

Pin initialization procedures are described below for the numbered combinations in table 11.59. The active level is assumed to be low.

Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Normal Mode: Figure 11.141 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in normal mode after re-setting.

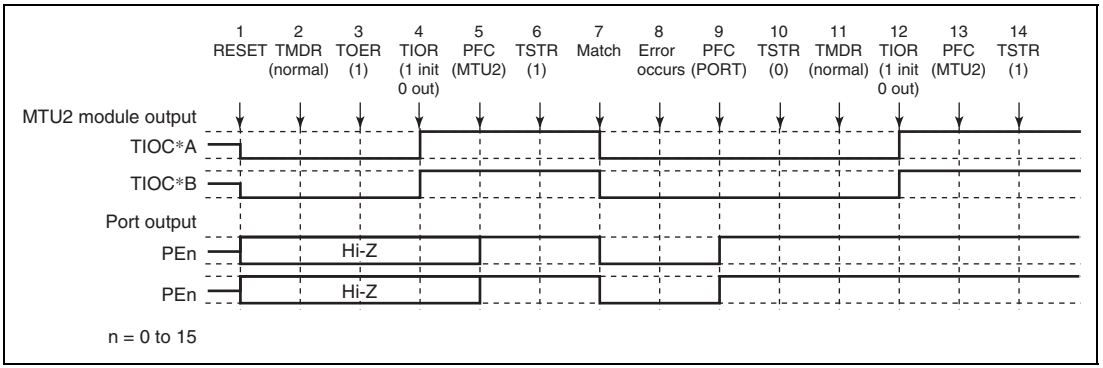


Figure 11.141 Error Occurrence in Normal Mode, Recovery in Normal Mode

1. After a reset, MTU2 output is low and ports are in the high-impedance state.
2. After a reset, the TMDR setting is for normal mode.
3. For channels 3 and 4, enable output with TOER before initializing the pins with TIOR.
4. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence.)
5. Set MTU2 output with the PFC.
6. The count operation is started by TSTR.
7. Output goes low on compare-match occurrence.
8. An error occurs.
9. Set port output with the PFC and output the inverse of the active level.
10. The count operation is stopped by TSTR.
11. Not necessary when restarting in normal mode.
12. Initialize the pins with TIOR.
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in PWM Mode 1: Figure 11.142 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.

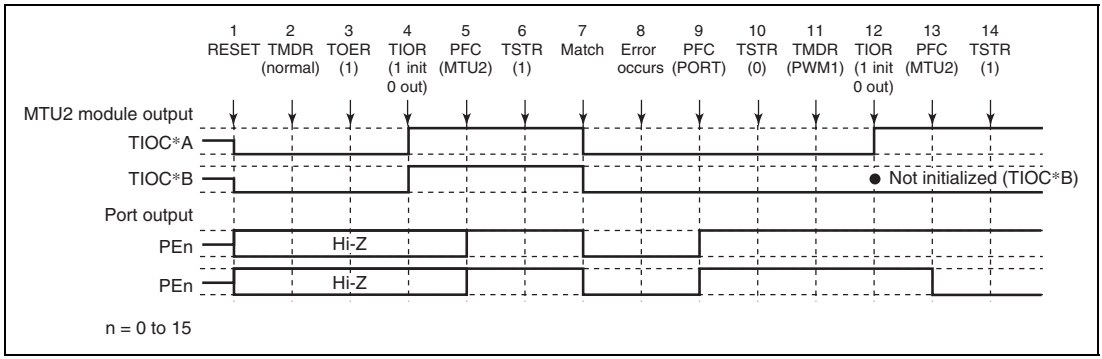


Figure 11.142 Error Occurrence in Normal Mode, Recovery in PWM Mode 1

1 to 10 are the same as in figure 11.141.

11. Set PWM mode 1.
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC*B side is not initialized. If initialization is required, initialize in normal mode, then switch to PWM mode 1.)
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in PWM Mode 2: Figure 11.143 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.

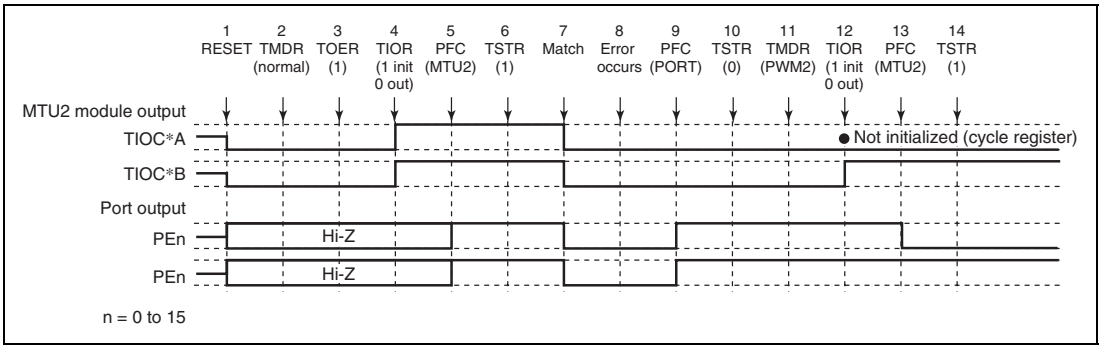


Figure 11.143 Error Occurrence in Normal Mode, Recovery in PWM Mode 2

1 to 10 are the same as in figure 11.141.

11. Set PWM mode 2.
12. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized. If initialization is required, initialize in normal mode, then switch to PWM mode 2.)
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Note: PWM mode 2 can only be set for channels 0 to 2, and therefore TOER setting is not necessary.

Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Phase Counting Mode: Figure 11.144 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.

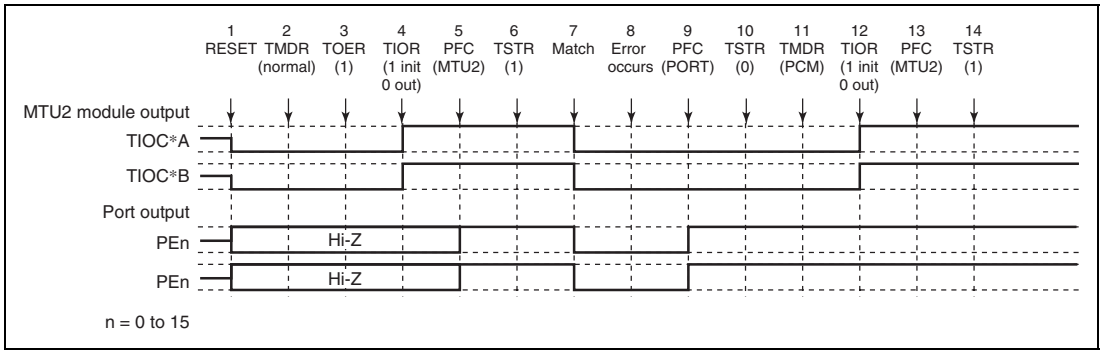


Figure 11.144 Error Occurrence in Normal Mode, Recovery in Phase Counting Mode

1 to 10 are the same as in figure 11.141.

11. Set phase counting mode.
12. Initialize the pins with TIOR.
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Note: Phase counting mode can only be set for channels 1 and 2, and therefore TOER setting is not necessary.

Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Complementary PWM Mode: Figure 11.145 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.

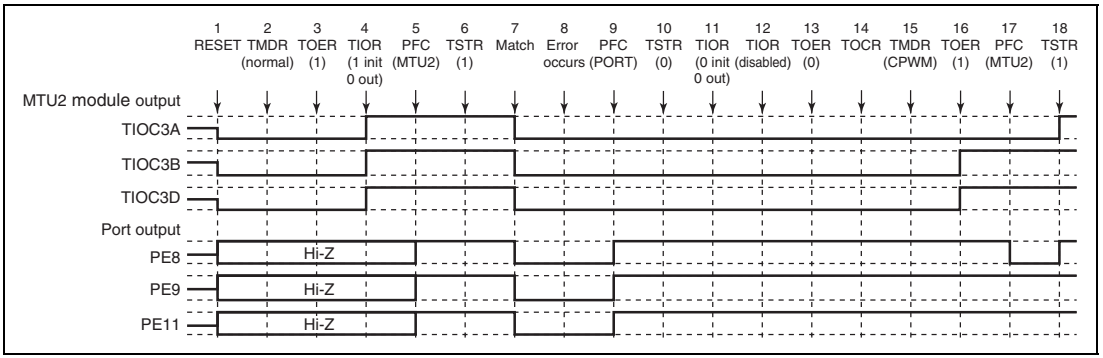


Figure 11.145 Error Occurrence in Normal Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 11.141.

11. Initialize the normal mode waveform generation section with TIOR.
12. Disable operation of the normal mode waveform generation section with TIOR.
13. Disable channel 3 and 4 output with TOER.
14. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
15. Set complementary PWM.
16. Enable channel 3 and 4 output with TOER.
17. Set MTU2 output with the PFC.
18. Operation is restarted by TSTR.

Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode: Figure 11.146 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.

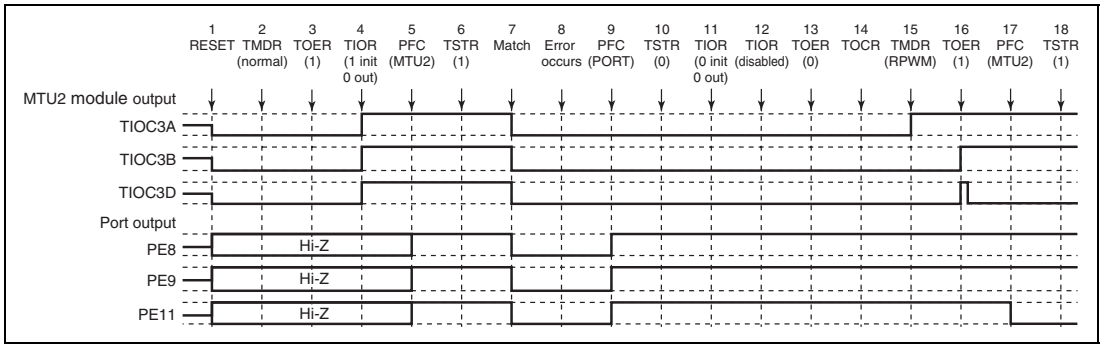


Figure 11.146 Error Occurrence in Normal Mode, Recovery in Reset-Synchronized PWM Mode

1 to 13 are the same as in figure 11.141.

14. Select the reset-synchronized PWM output level and cyclic output enabling/disabling with TOCR.
15. Set reset-synchronized PWM.
16. Enable channel 3 and 4 output with TOER.
17. Set MTU2 output with the PFC.
18. Operation is restarted by TSTR.

Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Normal Mode: Figure 11.147 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.

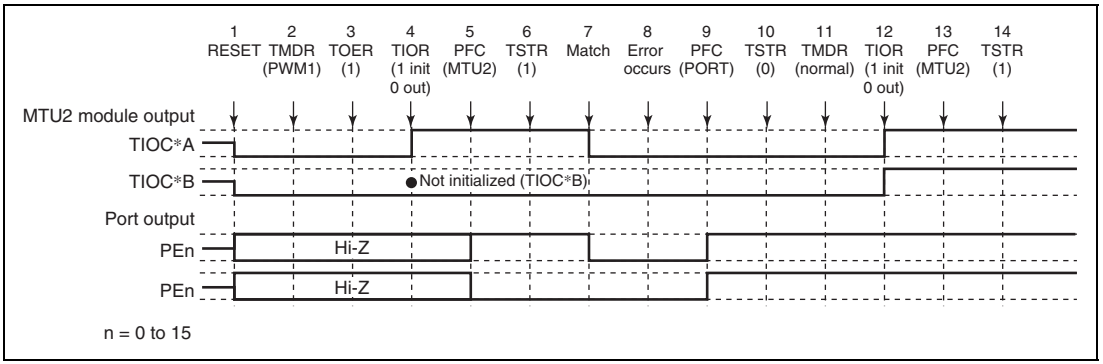


Figure 11.147 Error Occurrence in PWM Mode 1, Recovery in Normal Mode

1. After a reset, MTU2 output is low and ports are in the high-impedance state.
2. Set PWM mode 1.
3. For channels 3 and 4, enable output with TOER before initializing the pins with TIOR.
4. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence. In PWM mode 1, the TIOC*B side is not initialized.)
5. Set MTU2 output with the PFC.
6. The count operation is started by TSTR.
7. Output goes low on compare-match occurrence.
8. An error occurs.
9. Set port output with the PFC and output the inverse of the active level.
10. The count operation is stopped by TSTR.
11. Set normal mode.
12. Initialize the pins with TIOR.
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in PWM Mode 1: Figure 11.148 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.

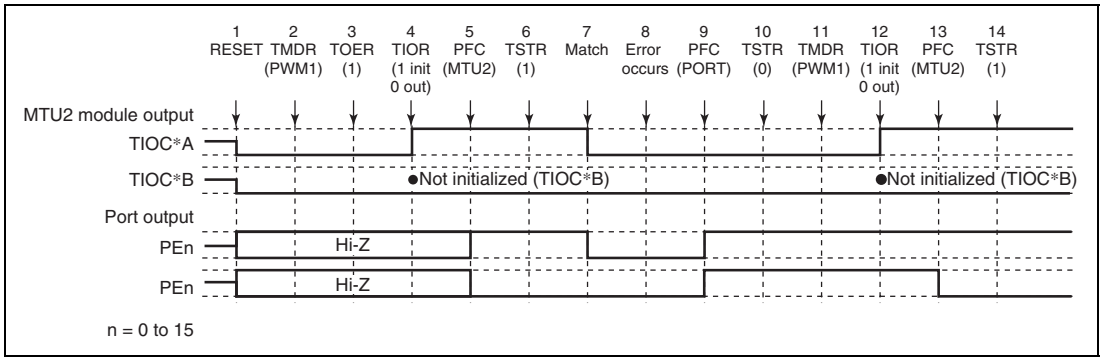


Figure 11.148 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 1

1 to 10 are the same as in figure 11.147.

11. Not necessary when restarting in PWM mode 1.
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC*B side is not initialized.)
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in PWM Mode 2: Figure 11.149 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.

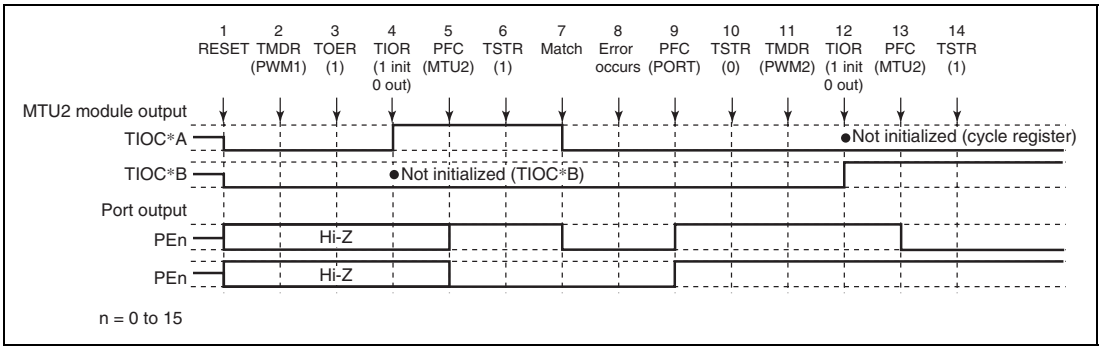


Figure 11.149 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 2

1 to 10 are the same as in figure 11.147.

11. Set PWM mode 2.
12. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Note: PWM mode 2 can only be set for channels 0 to 2, and therefore TOER setting is not necessary.

Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Phase Counting Mode: Figure 11.150 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.

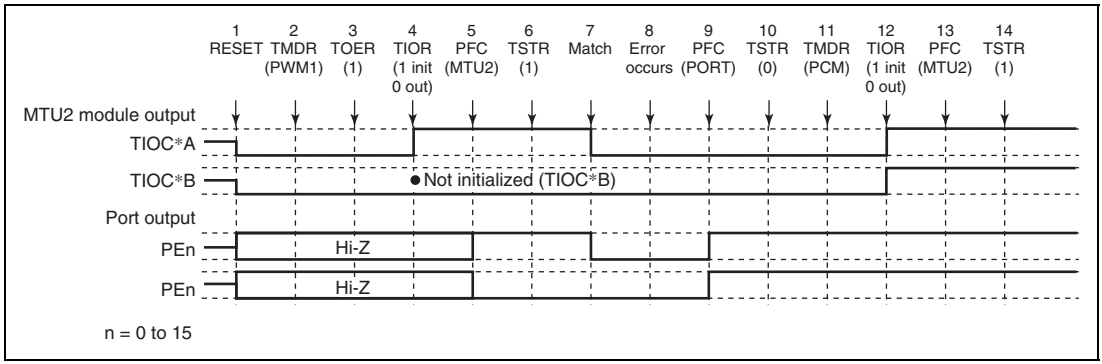


Figure 11.150 Error Occurrence in PWM Mode 1, Recovery in Phase Counting Mode

1 to 10 are the same as in figure 11.147.

11. Set phase counting mode.
12. Initialize the pins with TIOR.
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Note: Phase counting mode can only be set for channels 1 and 2, and therefore TOER setting is not necessary.

Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Complementary PWM Mode: Figure 11.151 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.

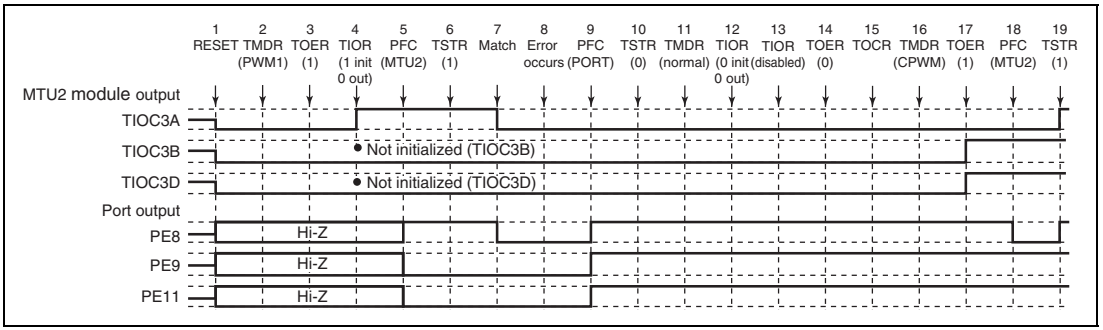


Figure 11.151 Error Occurrence in PWM Mode 1, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 11.147.

11. Set normal mode for initialization of the normal mode waveform generation section.
12. Initialize the PWM mode 1 waveform generation section with TIOR.
13. Disable operation of the PWM mode 1 waveform generation section with TIOR.
14. Disable channel 3 and 4 output with TOER.
15. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
16. Set complementary PWM.
17. Enable channel 3 and 4 output with TOER.
18. Set MTU2 output with the PFC.
19. Operation is restarted by TSTR.

Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Reset-Synchronized PWM Mode: Figure 11.152 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.

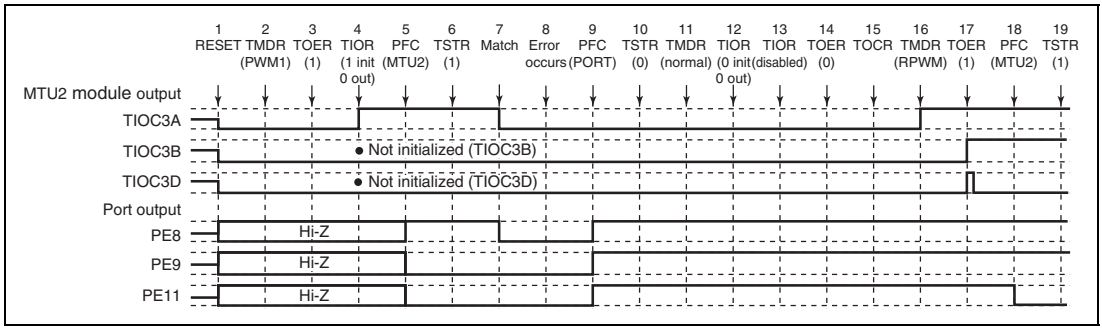


Figure 11.152 Error Occurrence in PWM Mode 1, Recovery in Reset-Synchronized PWM Mode

1 to 14 are the same as in figure 11.151.

15. Select the reset-synchronized PWM output level and cyclic output enabling/disabling with TOCR.
16. Set reset-synchronized PWM.
17. Enable channel 3 and 4 output with TOER.
18. Set MTU2 output with the PFC.
19. Operation is restarted by TSTR.

Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in Normal Mode: Figure 11.153 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.

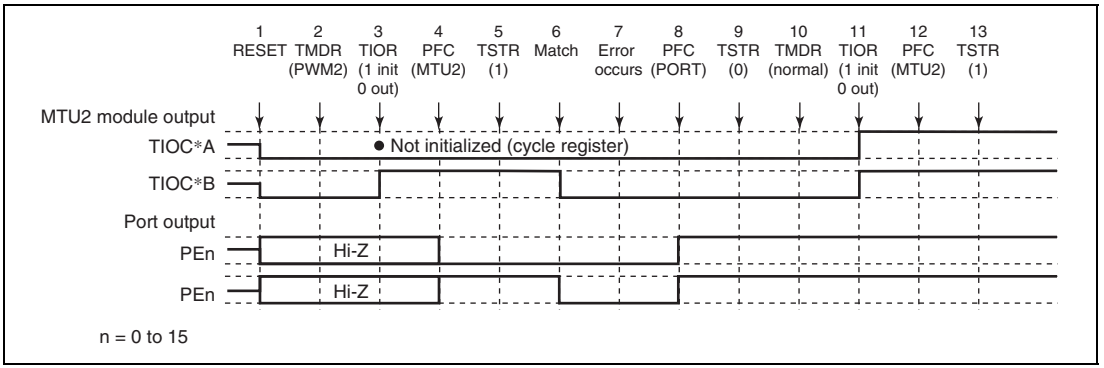


Figure 11.153 Error Occurrence in PWM Mode 2, Recovery in Normal Mode

1. After a reset, MTU2 output is low and ports are in the high-impedance state.
2. Set PWM mode 2.
3. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence. In PWM mode 2, the cycle register pins are not initialized. In the example, TIOC *A is the cycle register.)
4. Set MTU2 output with the PFC.
5. The count operation is started by TSTR.
6. Output goes low on compare-match occurrence.
7. An error occurs.
8. Set port output with the PFC and output the inverse of the active level.
9. The count operation is stopped by TSTR.
10. Set normal mode.
11. Initialize the pins with TIOR.
12. Set MTU2 output with the PFC.
13. Operation is restarted by TSTR.

Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in PWM Mode 1: Figure 11.154 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.

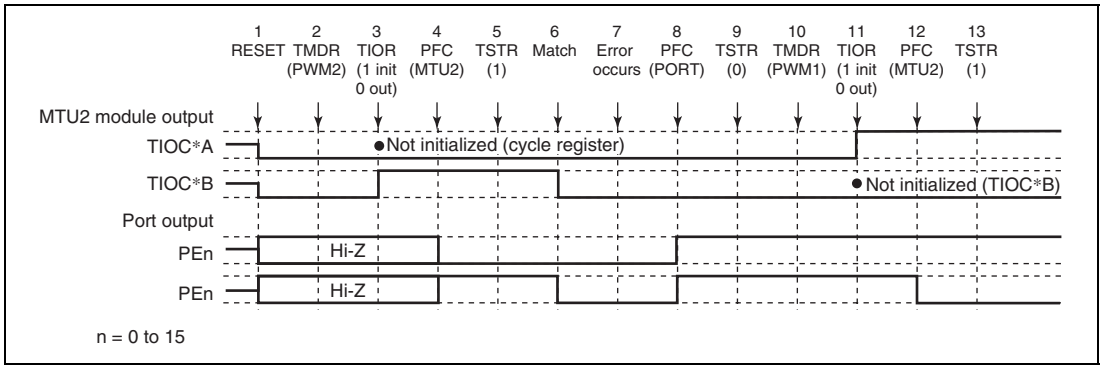


Figure 11.154 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 1

1 to 9 are the same as in figure 11.153.

10. Set PWM mode 1.
11. Initialize the pins with TIOR. (In PWM mode 1, the TIOC*B side is not initialized.)
12. Set MTU2 output with the PFC.
13. Operation is restarted by TSTR.

Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in PWM Mode 2: Figure 11.155 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.

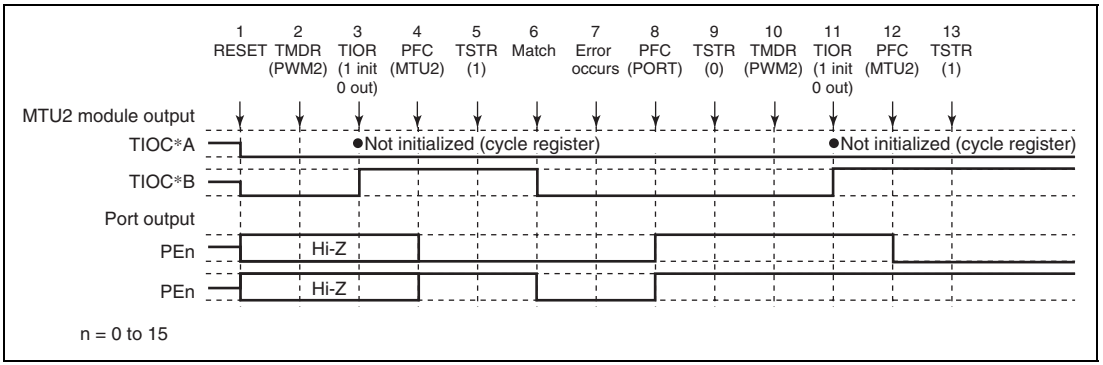


Figure 11.155 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 2

1 to 9 are the same as in figure 11.153.

10. Not necessary when restarting in PWM mode 2.
11. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
12. Set MTU2 output with the PFC.
13. Operation is restarted by TSTR.

Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in Phase Counting Mode: Figure 11.156 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.

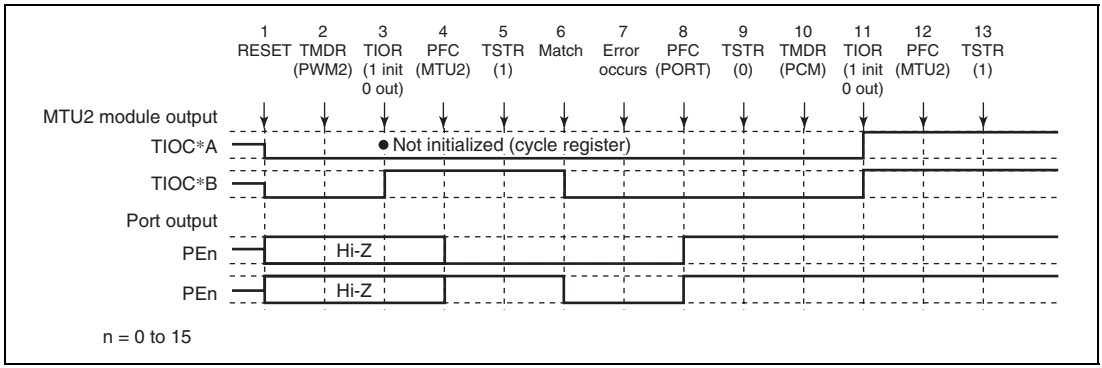


Figure 11.156 Error Occurrence in PWM Mode 2, Recovery in Phase Counting Mode

1 to 9 are the same as in figure 11.153.

10. Set phase counting mode.
11. Initialize the pins with TIOR.
12. Set MTU2 output with the PFC.
13. Operation is restarted by TSTR.

Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in Normal Mode: Figure 11.157 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.

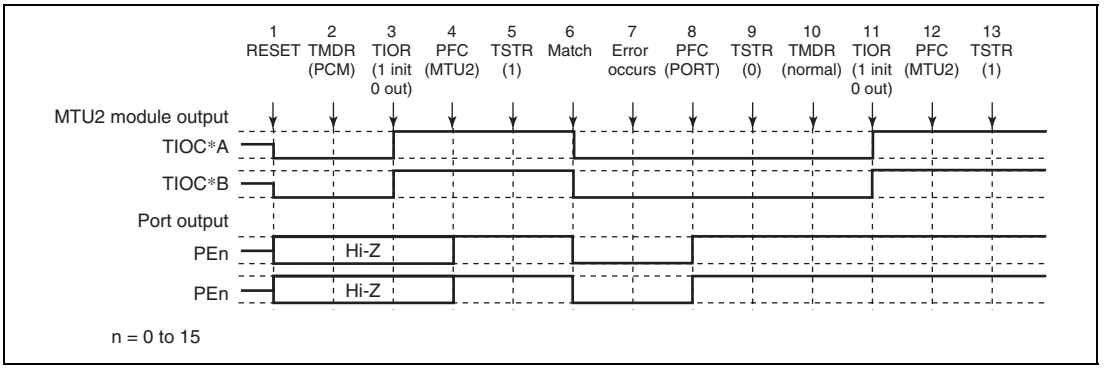


Figure 11.157 Error Occurrence in Phase Counting Mode, Recovery in Normal Mode

1. After a reset, MTU2 output is low and ports are in the high-impedance state.
2. Set phase counting mode.
3. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence.)
4. Set MTU2 output with the PFC.
5. The count operation is started by TSTR.
6. Output goes low on compare-match occurrence.
7. An error occurs.
8. Set port output with the PFC and output the inverse of the active level.
9. The count operation is stopped by TSTR.
10. Set in normal mode.
11. Initialize the pins with TIOR.
12. Set MTU2 output with the PFC.
13. Operation is restarted by TSTR.

Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in PWM Mode 1: Figure 11.158 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.

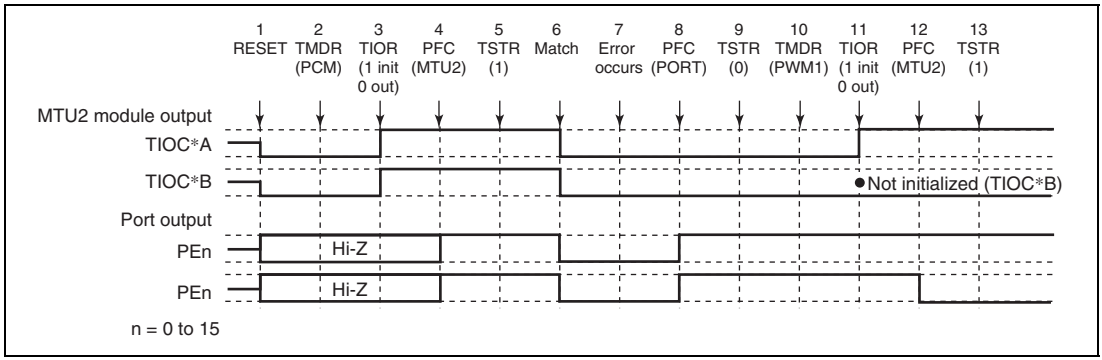


Figure 11.158 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 1

1 to 9 are the same as in figure 11.157.

10. Set PWM mode 1.
11. Initialize the pins with TIOR. (In PWM mode 1, the TIOC *B side is not initialized.)
12. Set MTU2 output with the PFC.
13. Operation is restarted by TSTR.

Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in PWM Mode 2: Figure 11.159 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in PWM mode 2 after re-setting.

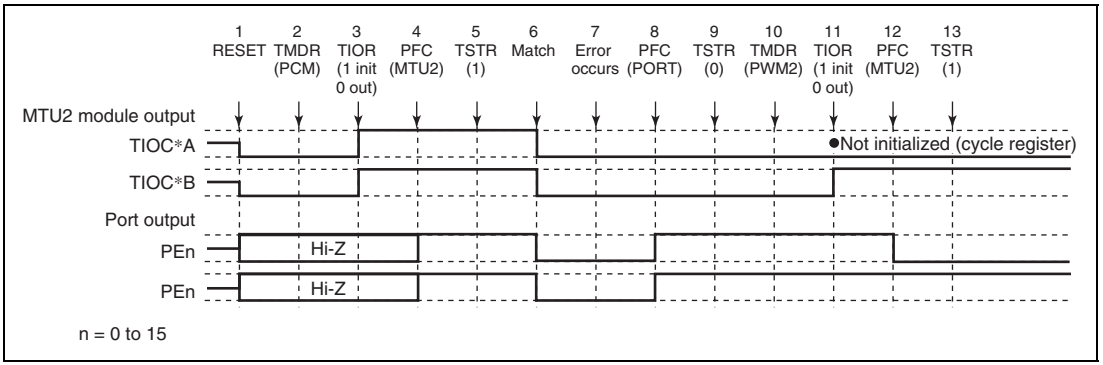


Figure 11.159 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 2

1 to 9 are the same as in figure 11.157.

10. Set PWM mode 2.
11. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
12. Set MTU2 output with the PFC.
13. Operation is restarted by TSTR.

Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in Phase Counting Mode: Figure 11.160 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.

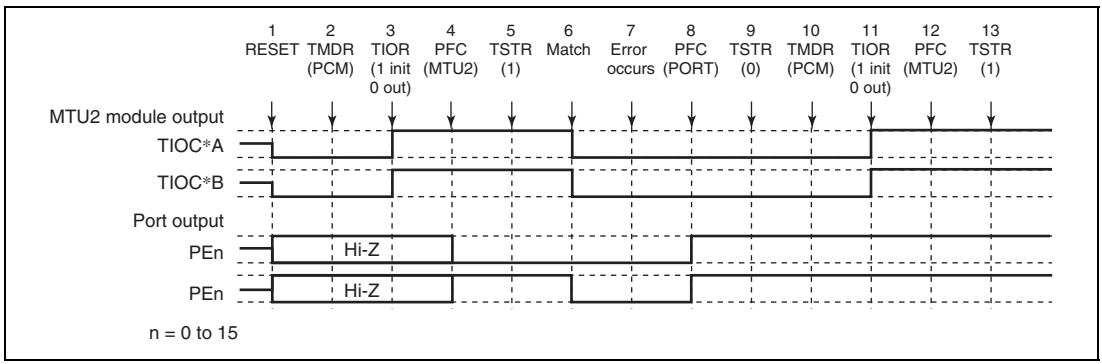


Figure 11.160 Error Occurrence in Phase Counting Mode, Recovery in Phase Counting Mode

1 to 9 are the same as in figure 11.157.

10. Not necessary when restarting in phase counting mode.
11. Initialize the pins with TIOR.
12. Set MTU2 output with the PFC.
13. Operation is restarted by TSTR.

Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Normal Mode: Figure 11.161 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.

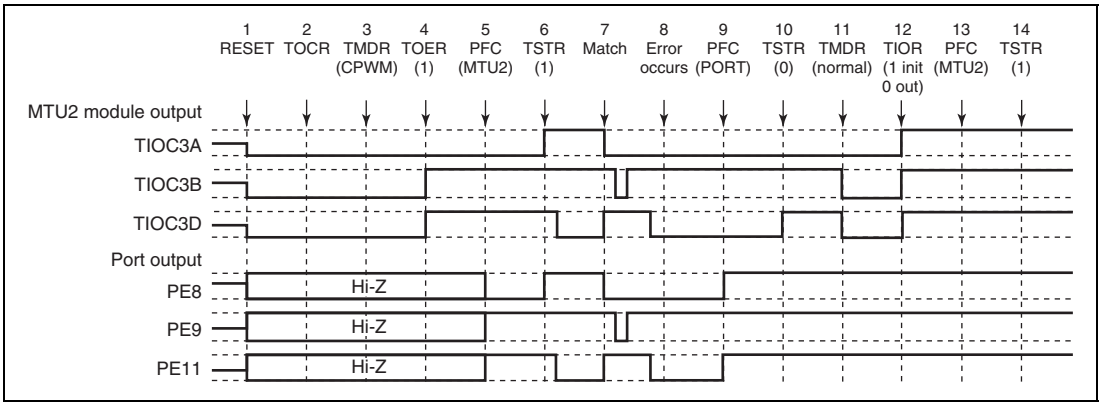


Figure 11.161 Error Occurrence in Complementary PWM Mode, Recovery in Normal Mode

1. After a reset, MTU2 output is low and ports are in the high-impedance state.
2. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
3. Set complementary PWM.
4. Enable channel 3 and 4 output with TOER.
5. Set MTU2 output with the PFC.
6. The count operation is started by TSTR.
7. The complementary PWM waveform is output on compare-match occurrence.
8. An error occurs.
9. Set port output with the PFC and output the inverse of the active level.
10. The count operation is stopped by TSTR. (MTU2 output becomes the complementary PWM output initial value.)
11. Set normal mode. (MTU2 output goes low.)
12. Initialize the pins with TIOR.
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in PWM Mode 1: Figure 11.162 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in PWM mode 1 after re-setting.

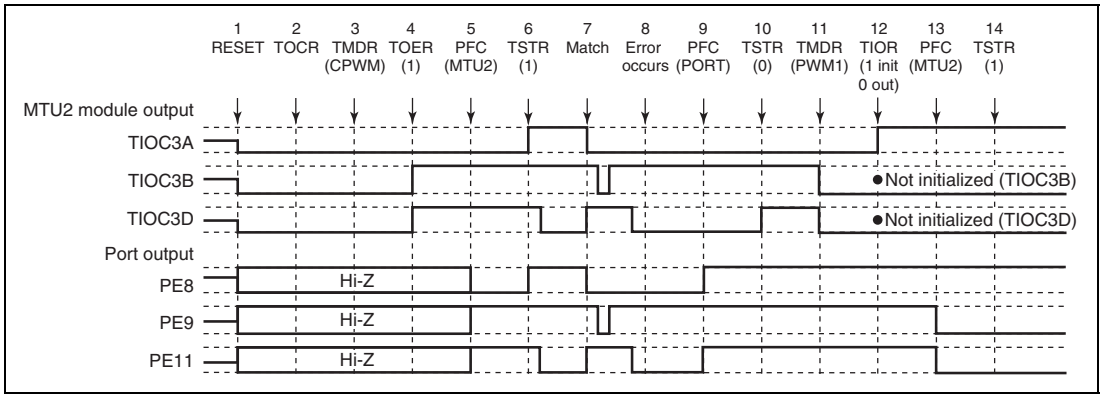


Figure 11.162 Error Occurrence in Complementary PWM Mode, Recovery in PWM Mode 1

1 to 10 are the same as in figure 11.161.

11. Set PWM mode 1. (MTU2 output goes low.)
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC *B side is not initialized.)
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode: Figure 11.163 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).

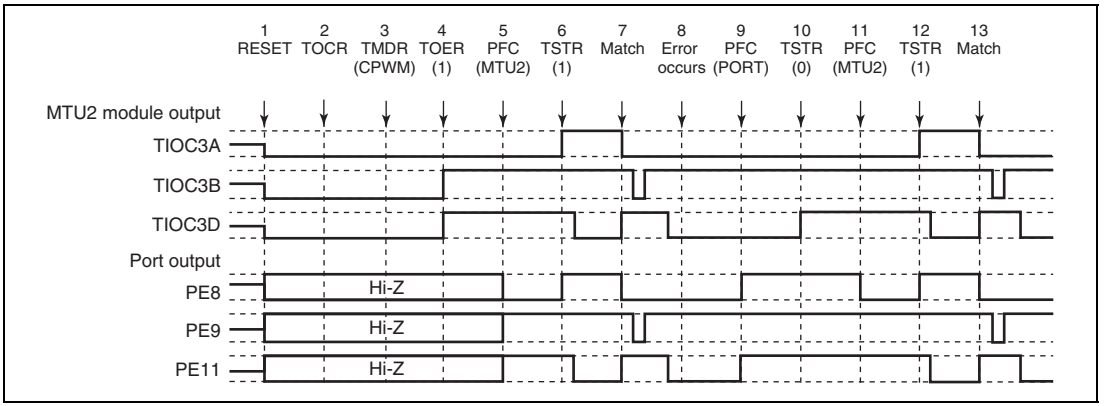


Figure 11.163 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 11.161.

11. Set MTU2 output with the PFC.
12. Operation is restarted by TSTR.
13. The complementary PWM waveform is output on compare-match occurrence.

Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode: Figure 11.164 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).

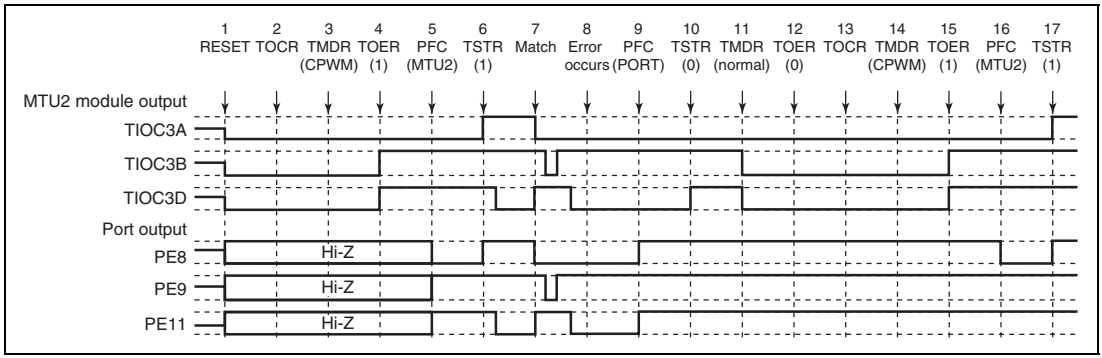


Figure 11.164 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 11.161.

11. Set normal mode and make new settings. (MTU2 output goes low.)
12. Disable channel 3 and 4 output with TOER.
13. Select the complementary PWM mode output level and cyclic output enabling/disabling with TOCR.
14. Set complementary PWM.
15. Enable channel 3 and 4 output with TOER.
16. Set MTU2 output with the PFC.
17. Operation is restarted by TSTR.

Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode: Figure 11.165 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.

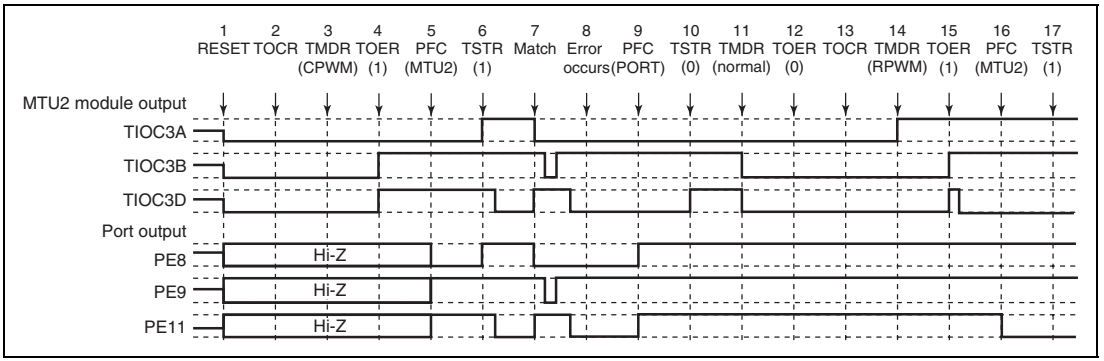


Figure 11.165 Error Occurrence in Complementary PWM Mode, Recovery in Reset-Synchronized PWM Mode

1 to 10 are the same as in figure 11.161.

11. Set normal mode. (MTU2 output goes low.)
12. Disable channel 3 and 4 output with TOER.
13. Select the reset-synchronized PWM mode output level and cyclic output enabling/disabling with TOCR.
14. Set reset-synchronized PWM.
15. Enable channel 3 and 4 output with TOER.
16. Set MTU2 output with the PFC.
17. Operation is restarted by TSTR.

Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Normal Mode: Figure 11.166 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.

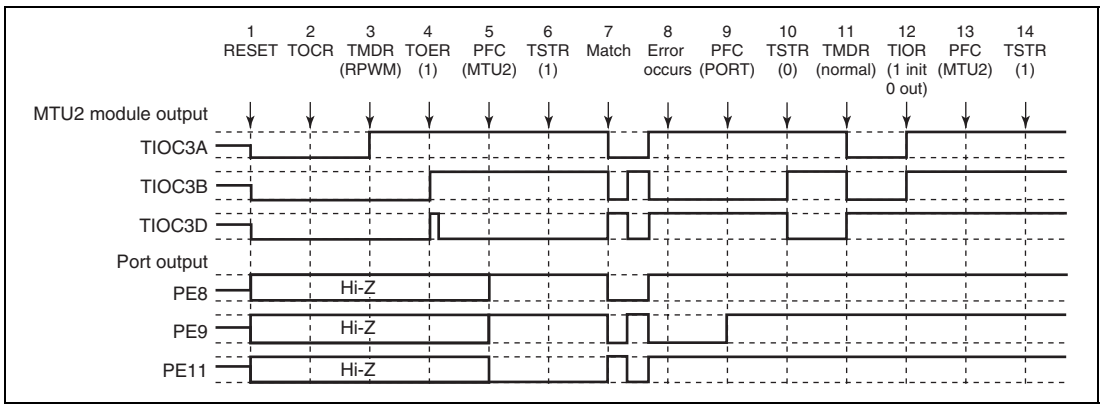


Figure 11.166 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Normal Mode

1. After a reset, MTU2 output is low and ports are in the high-impedance state.
2. Select the reset-synchronized PWM output level and cyclic output enabling/disabling with TOCR.
3. Set reset-synchronized PWM.
4. Enable channel 3 and 4 output with TOER.
5. Set MTU2 output with the PFC.
6. The count operation is started by TSTR.
7. The reset-synchronized PWM waveform is output on compare-match occurrence.
8. An error occurs.
9. Set port output with the PFC and output the inverse of the active level.
10. The count operation is stopped by TSTR. (MTU2 output becomes the reset-synchronized PWM output initial value.)
11. Set normal mode. (MTU2 positive phase output is low, and negative phase output is high.)
12. Initialize the pins with TIOR.
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in PWM Mode 1: Figure 11.167 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.

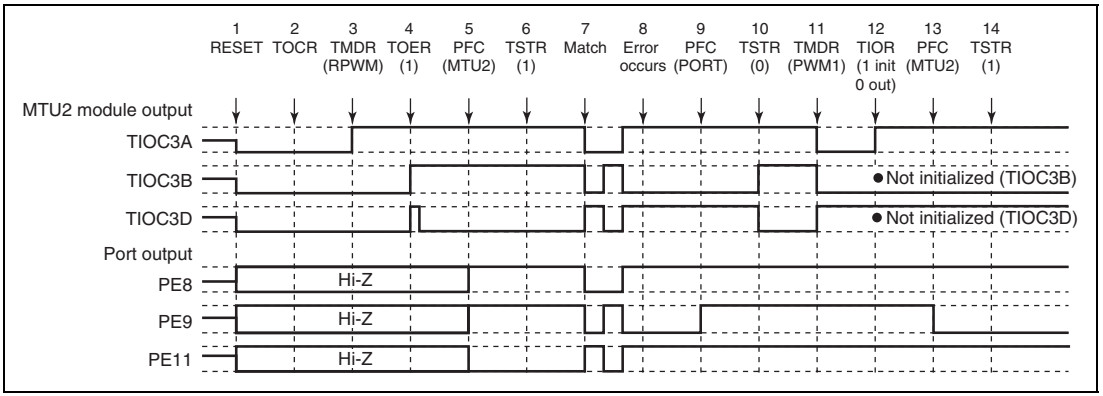


Figure 11.167 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in PWM Mode 1

1 to 10 are the same as in figure 11.166.

11. Set PWM mode 1. (MTU2 positive phase output is low, and negative phase output is high.)
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC *B side is not initialized.)
13. Set MTU2 output with the PFC.
14. Operation is restarted by TSTR.

Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode: Figure 11.168 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.

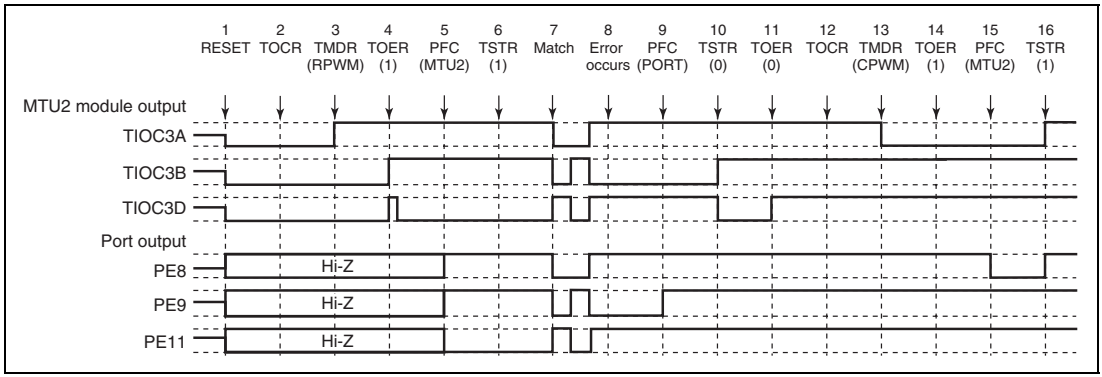


Figure 11.168 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 11.166.

11. Disable channel 3 and 4 output with TOER.
12. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
13. Set complementary PWM. (The MTU2 cyclic output pin goes low.)
14. Enable channel 3 and 4 output with TOER.
15. Set MTU2 output with the PFC.
16. Operation is restarted by TSTR.

Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode: Figure 11.169 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.

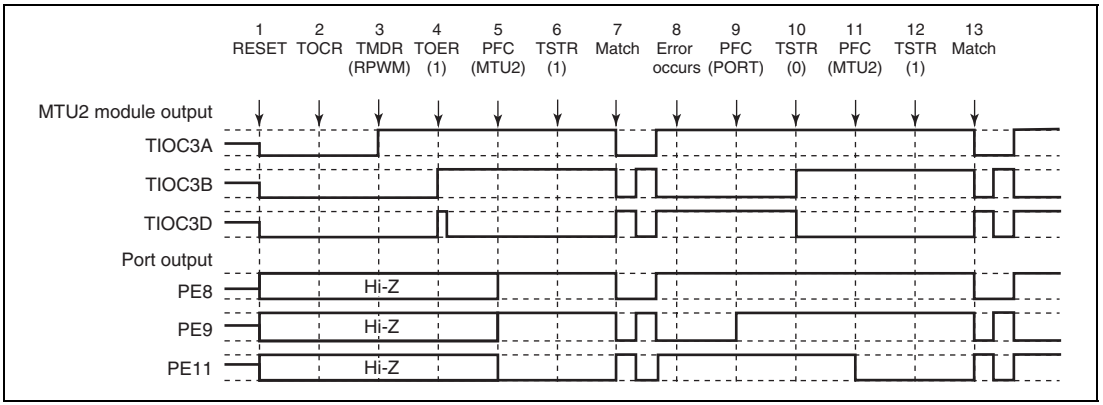


Figure 11.169 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Reset-Synchronized PWM Mode

1 to 10 are the same as in figure 11.166.

11. Set MTU2 output with the PFC.
12. Operation is restarted by TSTR.
13. The reset-synchronized PWM waveform is output on compare-match occurrence.

Section 12 Multi-Function Timer Pulse Unit 2S (MTU2S)

This LSI has an on-chip multi-function timer pulse unit 2S (MTU2S) that comprises three 16-bit timer channels. The MTU2S includes channels 3 to 5 of the MTU2. For details, refer to section 11, Multi-Function Timer Pulse Unit 2 (MTU2). To distinguish from the MTU2, "S" is added to the end of the MTU2S input/output pin and register names. For example, TIOC3A is called TIOC3AS and TGRA_3 is called TGRA_3S in this section.

The MTU2S can operate at 80 MHz max. for complementary PWM output functions or at 40 MHz max. for the other functions.

Table 12.1 MTU2S Functions

Item		Channel 3	Channel 4	Channel 5
Count clock		MIϕ/1 MIϕ/4 MIϕ/16 MIϕ/64 MIϕ/256 MIϕ/1024	MIϕ/1 MIϕ/4 MIϕ/16 MIϕ/64 MIϕ/256 MIϕ/1024	MIϕ/1 MIϕ/4 MIϕ/16 MIϕ/64
General registers (TGR)		TGRA_3S TGRB_3S	TGRA_4S TGRB_4S	TGRU_5S TGRV_5S TGRW_5S
General registers/ buffer registers		TGRC_3S TGRD_3S	TGRC_4S TGRD_4S	—
I/O pins		TIOC3AS TIOC3BS TIOC3CS TIOC3DS	TIOC4AS TIOC4BS TIOC4CS TIOC4DS	Input pins TIC5US TIC5VS TIC5WS
Counter clear function		TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture
Compare match output	0 output	√	√	—
	1 output	√	√	—
	Toggle output	√	√	—
Input capture function		√	√	√
Synchronous operation		√	√	—
PWM mode 1		√	√	—
PWM mode 2		—	—	—
Complementary PWM mode		√	√	—
Reset PWM mode		√	√	—
AC synchronous motor drive mode		—	—	—
Phase counting mode		—	—	—
Buffer operation		√	√	—

Item	Channel 3	Channel 4	Channel 5
Counter function of compensation for dead time	—	—	√
DTC activation	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow or underflow	TGR compare match or input capture
A/D converter start trigger	TGRA_3S compare match or input capture	TGRA_4S compare match or input capture TCNT_4S underflow (trough) in complementary PWM mode	—
Interrupt sources	5 sources • Compare match or input capture 3AS • Compare match or input capture 3BS • Compare match or input capture 3CS • Compare match or input capture 3DS • Overflow	5 sources • Compare match or input capture 4AS • Compare match or input capture 4BS • Compare match or input capture 4CS • Compare match or input capture 4DS • Overflow or underflow	3 sources • Compare match or input capture 5US • Compare match or input capture 5VS • Compare match or input capture 5WS
A/D converter start request delaying function	—	• A/D converter start request at a match between TADCORA_4S and TCNT_4S • A/D converter start request at a match between TADCORB_4S and TCNT_4S	—

Item	Channel 3	Channel 4	Channel 5
Interrupt skipping function	• Skips TGRA_3S compare match interrupts	• Skips TCIV_4S interrupts	—

[Legend]
√: Possible
—: Not possible

12.1 Input/Output Pins

Table 12.2 Pin Configuration

Channel	Symbol	I/O	Function
3	TIOC3AS	I/O	TGRA_3S input capture input/output compare output/PWM output pin
	TIOC3BS	I/O	TGRB_3S input capture input/output compare output/PWM output pin
	TIOC3CS	I/O	TGRC_3S input capture input/output compare output/PWM output pin
	TIOC3DS	I/O	TGRD_3S input capture input/output compare output/PWM output pin
4	TIOC4AS	I/O	TGRA_4S input capture input/output compare output/PWM output pin
	TIOC4BS	I/O	TGRB_4S input capture input/output compare output/PWM output pin
	TIOC4CS	I/O	TGRC_4S input capture input/output compare output/PWM output pin
	TIOC4DS	I/O	TGRD_4S input capture input/output compare output/PWM output pin
5	TIC5US	Input	TGRU_5S input capture input/external pulse input pin
	TIC5VS	Input	TGRV_5S input capture input/external pulse input pin
	TIC5WS	Input	TGRW_5S input capture input/external pulse input pin

12.2 Register Descriptions

The MTU2S has the following registers. For details on register addresses and register states during each process, refer to section 27, List of Registers. To distinguish registers in each channel, an underscore and the channel number are added as a suffix to the register name; TCR for channel 3 is expressed as TCR_3S.

Table 12.3 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Timer control register_3S	TCR_3S	R/W	H'00	H'FFFFC600	8, 16, 32
Timer control register_4S	TCR_4S	R/W	H'00	H'FFFFC601	8
Timer mode register_3S	TMDR_3S	R/W	H'00	H'FFFFC602	8, 16
Timer mode register_4S	TMDR_4S	R/W	H'00	H'FFFFC603	8
Timer I/O control register H_3S	TIORH_3S	R/W	H'00	H'FFFFC604	8, 16, 32
Timer I/O control register L_3S	TIORL_3S	R/W	H'00	H'FFFFC605	8
Timer I/O control register H_4S	TIORH_4S	R/W	H'00	H'FFFFC606	8, 16
Timer I/O control register L_4S	TIORL_4S	R/W	H'00	H'FFFFC607	8
Timer interrupt enable register_3S	TIER_3S	R/W	H'00	H'FFFFC608	8, 16
Timer interrupt enable register_4S	TIER_4S	R/W	H'00	H'FFFFC609	8
Timer output master enable register S	TOERS	R/W	H'C0	H'FFFFC60A	8
Timer gate control register S	TGCRS	R/W	H'80	H'FFFFC60D	8
Timer output control register 1S	TOCR1S	R/W	H'00	H'FFFFC60E	8, 16
Timer output control register 2S	TOCR2S	R/W	H'00	H'FFFFC60F	8
Timer counter_3S	TCNT_3S	R/W	H'0000	H'FFFFC610	16, 32
Timer counter_4S	TCNT_4S	R/W	H'0000	H'FFFFC612	16
Timer cycle data register S	TCDRS	R/W	H'FFFF	H'FFFFC614	16, 32
Timer dead time data register S	TDDRS	R/W	H'FFFF	H'FFFFC616	16
Timer general register A_3S	TGRA_3S	R/W	H'FFFF	H'FFFFC618	16, 32
Timer general register B_3S	TGRB_3S	R/W	H'FFFF	H'FFFFC61A	16
Timer general register A_4S	TGRA_4S	R/W	H'FFFF	H'FFFFC61C	16, 32
Timer general register B_4S	TGRB_4S	R/W	H'FFFF	H'FFFFC61E	16

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Timer subcounter S	TCNTSS	R	H'0000	H'FFFFC620	16, 32
Timer cycle buffer register S	TCBRS	R/W	H'FFFF	H'FFFFC622	16
Timer general register C_3S	TGRC_3S	R/W	H'FFFF	H'FFFFC624	16, 32
Timer general register D_3S	TGRD_3S	R/W	H'FFFF	H'FFFFC626	16
Timer general register C_4S	TGRC_4S	R/W	H'FFFF	H'FFFFC628	16, 32
Timer general register D_4S	TGRD_4S	R/W	H'FFFF	H'FFFFC62A	16
Timer status register_3S	TSR_3S	R/W	H'C0	H'FFFFC62C	8, 16
Timer status register_4S	TSR_4S	R/W	H'C0	H'FFFFC62D	8
Timer interrupt skipping set register S	TITCRS	R/W	H'00	H'FFFFC630	8, 16
Timer interrupt skipping counter S	TITCNTS	R	H'00	H'FFFFC631	8
Timer buffer transfer set register S	TBTERS	R/W	H'00	H'FFFFC632	8
Timer dead time enable register S	TDERS	R/W	H'01	H'FFFFC634	8
Timer output level buffer register S	TOLBRS	R/W	H'00	H'FFFFC636	8
Timer buffer operation transfer mode register_3S	TBTM_3S	R/W	H'00	H'FFFFC638	8, 16
Timer buffer operation transfer mode register_4S	TBTM_4S	R/W	H'00	H'FFFFC639	8
Timer A/D converter start request control register S	TADCRS	R/W	H'0000	H'FFFFC640	16
Timer A/D converter start request cycle set register A_4S	TADCORA_4S	R/W	H'FFFF	H'FFFFC644	16, 32
Timer A/D converter start request cycle set register B_4S	TADCORB_4S	R/W	H'FFFF	H'FFFFC646	16
Timer A/D converter start request cycle set buffer register A_4S	TADCOBRA_4S	R/W	H'FFFF	H'FFFFC648	16, 32
Timer A/D converter start request cycle set buffer register B_4S	TADCOBRB_4S	R/W	H'FFFF	H'FFFFC64A	16

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Timer synchronous clear register S	TSYCRS	R/W	H'00	H'FFFFC650	8
Timer waveform control register S	TWCRS	R/W	H'00	H'FFFFC660	8
Timer start register S	TSTRS	R/W	H'00	H'FFFFC680	8, 16
Timer synchronous register S	TSYRS	R/W	H'00	H'FFFFC681	8
Timer read/write enable register S	TRWERS	R/W	H'01	H'FFFFC684	8
Timer counter U_5S	TCNTU_5S	R/W	H'0000	H'FFFFC880	16, 32
Timer general register U_5S	TGRU_5S	R/W	H'FFFF	H'FFFFC882	16
Timer control register U_5S	TCRU_5S	R/W	H'00	H'FFFFC884	8
Timer I/O control register U_5S	TIORU_5S	R/W	H'00	H'FFFFC886	8
Timer counter V_5S	TCNTV_5S	R/W	H'0000	H'FFFFC890	16, 32
Timer general register V_5S	TGRV_5S	R/W	H'FFFF	H'FFFFC892	16
Timer control register V_5S	TCRV_5S	R/W	H'00	H'FFFFC894	8
Timer I/O control register V_5S	TIORV_5S	R/W	H'00	H'FFFFC896	8
Timer counter W_5S	TCNTW_5S	R/W	H'0000	H'FFFFC8A0	16, 32
Timer general register W_5S	TGRW_5S	R/W	H'FFFF	H'FFFFC8A2	16
Timer control register W_5S	TCRW_5S	R/W	H'00	H'FFFFC8A4	8
Timer I/O control register W_5S	TIORW_5S	R/W	H'00	H'FFFFC8A6	8
Timer status register_5S	TSR_5S	R/W	H'00	H'FFFFC8B0	8
Timer interrupt enable register_5S	TIER_5S	R/W	H'00	H'FFFFC8B2	8
Timer start register_5S	TSTR_5S	R/W	H'00	H'FFFFC8B4	8
Timer compare match clear register S	TCNTCMPCLRS	R/W	H'00	H'FFFFC8B6	8

Section 13 Port Output Enable (POE)

The port output enable (POE) can be used to place the high-current pins (pins multiplexed with TIOC3B, TIOC3D, TIOC4A, TIOC4B, TIOC4C, and TIOC4D in the MTU2 and TIOC3BS, TIOC3DS, TIOC4AS, TIOC4BS, TIOC4CS, and TIOC4DS in the MTU2S) and the pins for channel 0 of the MTU2 (pins multiplexed with TIOC0A, TIOC0B, TIOC0C, and TIOC0D) in high-impedance state, depending on the change on $\overline{\text{POE0}}$ to $\overline{\text{POE8}}$ input pins and the output status of the high-current pins, or by modifying register settings. It can also simultaneously generate interrupt requests.

13.1 Features

- Each of the $\overline{\text{POE0}}$ to $\overline{\text{POE8}}$ input pins can be set for falling edge, Pϕ/8 × 16, Pϕ/16 × 16, or Pϕ/128 × 16 low-level sampling.
- High-current pins and the pins for channel 0 of the MTU2 can be placed in high-impedance state by $\overline{\text{POE0}}$ to $\overline{\text{POE8}}$ pin falling-edge or low-level sampling.
- High-current pins can be placed in high-impedance state when the high-current pin output levels are compared and simultaneous active-level output continues for one cycle or more.
- High-current pins and the pins for channel 0 of the MTU2 can be placed in high-impedance state by modifying the POE register settings.
- Interrupts can be generated by input-level sampling or output-level comparison results.

The POE has input level detection circuits, output level comparison circuits, and a high-impedance request/interrupt request generating circuit as shown in figure 13.1.

In addition to control by the POE, high-current pins can be placed in high-impedance state when the oscillator stops or in software standby state. For details, refer to section 21.1.11, High-Current Port Control Register (HCPCR), and appendix A, Pin States.

Figure 13.1 shows a block diagram of the POE.

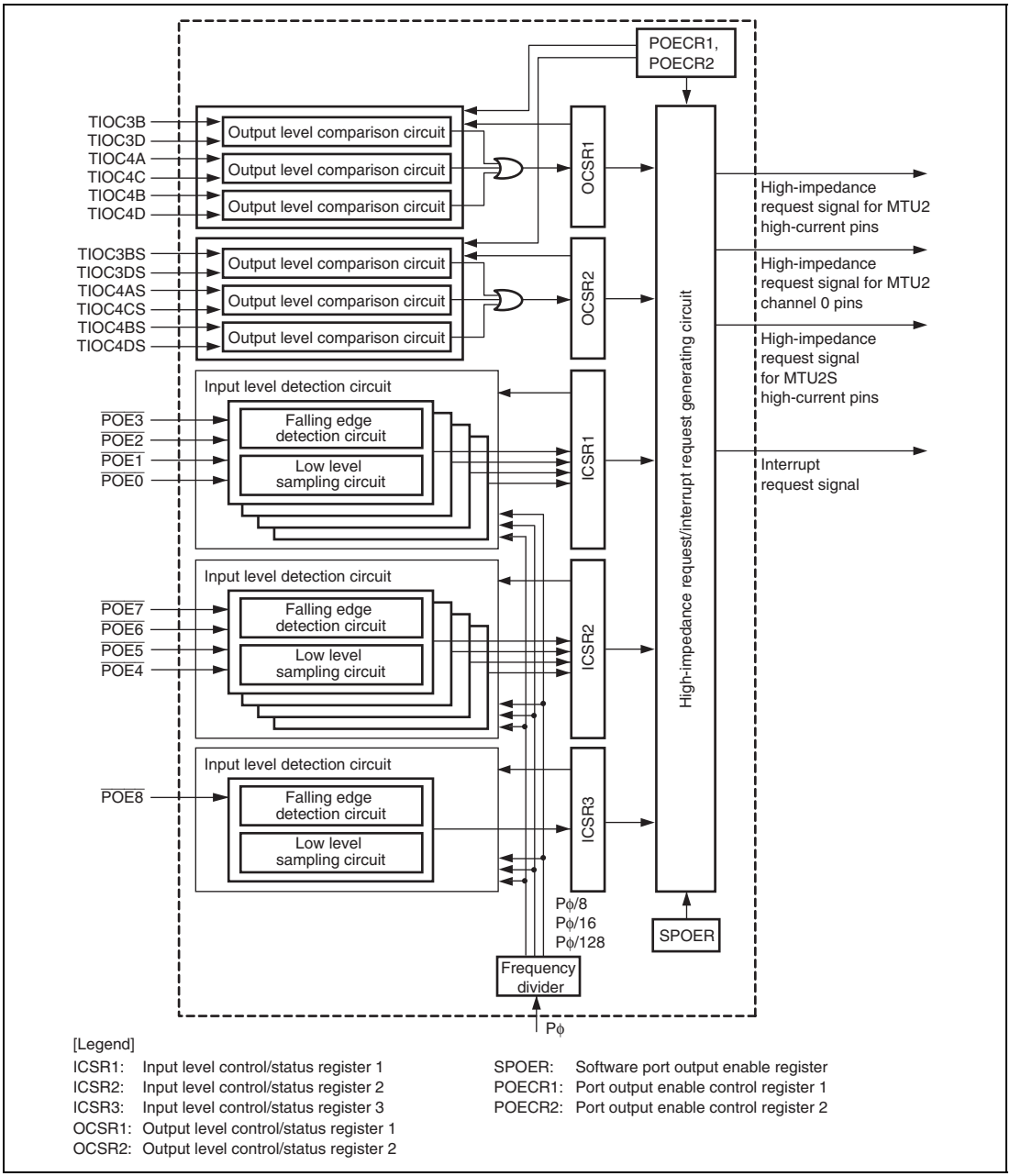


Figure 13.1 Block Diagram of POE

13.2 Input/Output Pins

Table 13.1 Pin Configuration

Pin Name	Symbol	I/O	Description
Port output enable input pins 0 to 3	$\overline{\text{POE0}}$ to $\overline{\text{POE3}}$	Input	Input request signals to place high-current pins for MTU2 in high-impedance state
Port output enable input pins 4 to 7	$\overline{\text{POE4}}$ to $\overline{\text{POE7}}$	Input	Input request signals to place high-current pins for MTU2S in high-impedance state
Port output enable input pin 8	$\overline{\text{POE8}}$	Input	Inputs a request signal to place pins for channel 0 in MTU2 in high-impedance state

Table 13.2 shows output-level comparisons with pin combinations.

Table 13.2 Pin Combinations

Pin Combination	I/O	Description
PE9/TIOC3B and PE11/TIOC3D	Output	The high-current pins for the MTU2 are placed in high-impedance state when the pins simultaneously output an active level (low level when the output level select P (OLSP) bit of the timer output control register (TOCR) in the MTU2 is 0 or high level when the bit is 1) for one or more cycles of the peripheral clock (Pϕ). This active level comparison is done when the MTU2 output function or general output function is selected in the pin function controller. If another function is selected, the output level is not checked. Pin combinations for output comparison and high-impedance control can be selected by POE registers.
PE12/TIOC4A and PE14/TIOC4C		
PE13/TIOC4B and PE15/TIOC4D		
PD9/TIOC3BS and PD11/TIOC3DS	Output	The high-current pins for the MTU2S are placed in high-impedance state when the pins simultaneously output an active level (low level when the output level select P (OLSP) bit of the timer output control register (TOCR) in the MTU2S is 0 or high level when the bit is 1) for one or more cycles of the peripheral clock (Pϕ). This active level comparison is done when the MTU2S output function or general output function is selected in the pin function controller. If another function is selected, the output level is not checked. Pin combinations for output comparison and high-impedance control can be selected by POE registers.
PD12/TIOC4AS and PD14/TIOC4CS		
PD13/TIOC4BS and PD15/TIOC4DS		
PD29/TIOC3BS and PD28/TIOC3DS		
PD27/TIOC4AS and PD25/TIOC4CS		
PD26/TIOC4BS and PD24/TIOC4DS		
PE16/TIOC3BS and PE17/TIOC3DS		
PE18/TIOC4AS and PE20/TIOC4CS		
PE19/TIOC4BS and PE21/TIOC4DS		

13.3 Register Descriptions

The POE has the following registers. For details on register addresses and register states during each processing, refer to section 27, List of Registers.

Table 13.3 Register Configuration

Register Name	Abbreviation	R/W	Initial value	Address	Access Size
Input level control/status register 1	ICSR1	R/W	H'0000	H'FFFFD000	8, 16, 32
Output level control/status register 1	OCSR1	R/W	H'0000	H'FFFFD002	8, 16
Input level control/status register 2	ICSR2	R/W	H'0000	H'FFFFD004	8, 16, 32
Output level control/status register 2	OCSR2	R/W	H'0000	H'FFFFD006	8, 16
Input level control/status register 3	ICSR3	R/W	H'0000	H'FFFFD008	8, 16
Software port output enable register	SPOER	R/W	H'00	H'FFFFD00A	8
Port output enable control register 1	POECR1	R/W	H'00	H'FFFFD00B	8
Port output enable control register 2	POECR2	R/W	H'7700	H'FFFFD00C	8, 16

13.3.1 Input Level Control/Status Register 1 (ICSR1)

ICSR1 is a 16-bit readable/writable register that selects the $\overline{POE0}$ to $\overline{POE3}$ pin input modes, controls the enable/disable of interrupts, and indicates status.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	POE3F	POE2F	POE1F	POE0F	-	-	-	PIE1	POE3M[1:0]		POE2M[1:0]		POE1M[1:0]		POE0M[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W:R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R	R	R	R/W	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

2. Can be modified only once after a power-on reset.

Bit	Bit Name	Initial value	R/W	Description
15	POE3F	0	R/(W)*1	POE3 Flag This flag indicates that a high impedance request has been input to the $\overline{POE3}$ pin. [Clearing conditions] • By writing 0 to POE3F after reading POE3F = 1 (when the falling edge is selected by bits 7 and 6 in ICSR1) • By writing 0 to POE3F after reading POE3F = 1 after a high level input to POE3 is sampled at Pϕ/8, Pϕ/16, or Pϕ/128 clock (when low-level sampling is selected by bits 7 and 6 in ICSR1) [Setting condition] • When the input set by ICSR1 bits 7 and 6 occurs at the $\overline{POE3}$ pin

Bit	Bit Name	Initial value	R/W	Description
14	POE2F	0	R/(W)*[1]	POE2 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE2}}$ pin. [Clearing conditions] • By writing 0 to POE2F after reading POE2F = 1 (when the falling edge is selected by bits 5 and 4 in ICSR1) • By writing 0 to POE2F after reading POE2F = 1 after a high level input to POE2 is sampled at Pϕ/8, Pϕ/16, or Pϕ/128 clock (when low-level sampling is selected by bits 5 and 4 in ICSR1) [Setting condition] • When the input set by ICSR1 bits 5 and 4 occurs at the $\overline{\text{POE2}}$ pin
13	POE1F	0	R/(W)*[1]	POE1 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE1}}$ pin. [Clearing conditions] • By writing 0 to POE1F after reading POE1F = 1 (when the falling edge is selected by bits 3 and 2 in ICSR1) • By writing 0 to POE1F after reading POE1F = 1 after a high level input to POE1 is sampled at Pϕ/8, Pϕ/16, or Pϕ/128 clock (when low-level sampling is selected by bits 3 and 2 in ICSR1) [Setting condition] • When the input set by ICSR1 bits 3 and 2 occurs at the $\overline{\text{POE1}}$ pin

Bit	Bit Name	Initial value	R/W	Description
12	POE0F	0	R/(W)*[1]	POE0 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE0}}$ pin. [Clearing conditions] • By writing 0 to POE0F after reading POE0F = 1 (when the falling edge is selected by bits 1 and 0 in ICSR1) • By writing 0 to POE0F after reading POE0F = 1 after a high level input to POE0 is sampled at Pϕ/8, Pϕ/16, or Pϕ/128 clock (when low-level sampling is selected by bits 1 and 0 in ICSR1) [Setting condition] • When the input set by ICSR1 bits 1 and 0 occurs at the $\overline{\text{POE0}}$ pin
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PIE1	0	R/W	Port Interrupt Enable 1 This bit enables/disables interrupt requests when any one of the POE0F to POE3F bits of the ICSR1 is set to 1. 0: Interrupt requests disabled 1: Interrupt requests enabled
7, 6	POE3M[1:0]	00	R/W*[2]	POE3 mode 1, 0 These bits select the input mode of the $\overline{\text{POE3}}$ pin. 00: Accept request on falling edge of POE3 input 01: Accept request when POE3 input has been sampled for 16 Pϕ/8 clock pulses and all are low level. 10: Accept request when POE3 input has been sampled for 16 Pϕ/16 clock pulses and all are low level. 11: Accept request when POE3 input has been sampled for 16 Pϕ/128 clock pulses and all are low level.

Bit	Bit Name	Initial value	R/W	Description
5, 4	POE2M[1:0]	00	R/W*[2]	POE2 mode 1, 0 These bits select the input mode of the $\overline{\text{POE2}}$ pin. 00: Accept request on falling edge of POE2 input 01: Accept request when POE2 input has been sampled for 16 Pϕ/8 clock pulses and all are low level. 10: Accept request when POE2 input has been sampled for 16 Pϕ/16 clock pulses and all are low level. 11: Accept request when POE2 input has been sampled for 16 Pϕ/128 clock pulses and all are low level.
3, 2	POE1M[1:0]	00	R/W*[2]	POE1 mode 1, 0 These bits select the input mode of the $\overline{\text{POE1}}$ pin. 00: Accept request on falling edge of POE1 input 01: Accept request when POE1 input has been sampled for 16 Pϕ/8 clock pulses and all are low level. 10: Accept request when POE1 input has been sampled for 16 Pϕ/16 clock pulses and all are low level. 11: Accept request when POE1 input has been sampled for 16 Pϕ/128 clock pulses and all are low level.
1, 0	POE0M[1:0]	00	R/W*[2]	POE0 mode 1, 0 These bits select the input mode of the $\overline{\text{POE0}}$ pin. 00: Accept request on falling edge of POE0 input 01: Accept request when POE0 input has been sampled for 16 Pϕ/8 clock pulses and all are low level. 10: Accept request when POE0 input has been sampled for 16 Pϕ/16 clock pulses and all are low level. 11: Accept request when POE0 input has been sampled for 16 Pϕ/128 clock pulses and all are low level.

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.
2. Can be modified only once after a power-on reset.

13.3.2 Output Level Control/Status Register 1 (OCSR1)

OCSR1 is a 16-bit readable/writable register that controls the enable/disable of both output level comparison and interrupts, and indicates status.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSF1	-	-	-	-	-	OCE1	OIE1	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*1	R	R	R	R	R	R/W*2	R/W	R	R	R	R	R	R	R	R

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

2. Can be modified only once after a power-on reset.

Bit	Bit Name	Initial value	R/W	Description
15	OSF1	0	R/(W)*1	Output Short Flag 1 This flag indicates that any one of the three pairs of MTU2 2-phase outputs to be compared has simultaneously become an active level. [Clearing condition] • By writing 0 to OSF1 after reading OSF1 = 1 [Setting condition] • When any one of the three pairs of 2-phase outputs has simultaneously become an active level
14 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	OCE1	0	R/W*2	Output Short High-Impedance Enable 1 This bit specifies whether to place the pins in high-impedance state when the OSF1 bit in OCSR1 is set to 1. 0: Does not place the pins in high-impedance state 1: Places the pins in high-impedance state
8	OIE1	0	R/W	Output Short Interrupt Enable 1 This bit enables or disables interrupt requests when the OSF1 bit in OCSR is set to 1. 0: Interrupt requests disabled 1: Interrupt requests enabled

Bit	Bit Name	Initial value	R/W	Description
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.
2. Can be modified only once after a power-on reset.

13.3.3 Input Level Control/Status Register 2 (ICSR2)

ICSR2 is a 16-bit readable/writable register that selects the $\overline{\text{POE4}}$ to $\overline{\text{POE7}}$ pin input modes, controls the enable/disable of interrupts, and indicates status.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	POE7F	POE6F	POE5F	POE4F	-	-	-	PIE2	POE7M[1:0]		POE6M[1:0]		POE5M[1:0]		POE4M[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R	R	R	R/W	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.
2. Can be modified only once after a power-on reset.

Bit	Bit Name	Initial value	R/W	Description
15	POE7F	0	R/(W)*1	POE7 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE7}}$ pin. [Clearing conditions] • By writing 0 to POE7F after reading POE7F = 1 (when the falling edge is selected by bits 7 and 6 in ICSR2) • By writing 0 to POE7F after reading POE7F = 1 after a high level input to POE7 is sampled at Pϕ/8, Pϕ/16, or Pϕ/128 clock (when low-level sampling is selected by bits 7 and 6 in ICSR2) [Setting condition] • When the input condition set by bits 7 and 6 in ICSR2 occurs at the $\overline{\text{POE7}}$ pin

Bit	Bit Name	Initial value	R/W	Description
14	POE6F	0	R/(W)*[1]	POE6 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE6}}$ pin. [Clearing conditions] • By writing 0 to POE6F after reading POE6F = 1 (when the falling edge is selected by bits 5 and 4 in ICSR2) • By writing 0 to POE6F after reading POE6F = 1 after a high level input to POE6 is sampled at Pϕ/8, Pϕ/16, or Pϕ/128 clock (when low-level sampling is selected by bits 5 and 4 in ICSR2) [Setting condition] • When the input condition set by bits 5 and 4 in ICSR2 occurs at the $\overline{\text{POE6}}$ pin
13	POE5F	0	R/(W)*[1]	POE5 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE5}}$ pin. [Clearing conditions] • By writing 0 to POE5F after reading POE5F = 1 (when the falling edge is selected by bits 3 and 2 in ICSR2) • By writing 0 to POE5F after reading POE5F = 1 after a high level input to POE5 is sampled at Pϕ/8, Pϕ/16, or Pϕ/128 clock (when low-level sampling is selected by bits 3 and 2 in ICSR2) [Setting condition] • When the input condition set by bits 3 and 2 in ICSR2 occurs at the $\overline{\text{POE5}}$ pin

Bit	Bit Name	Initial value	R/W	Description
12	POE4F	0	R/(W)*[1]	POE4 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE4}}$ pin. [Clearing conditions] • By writing 0 to POE4F after reading POE4F = 1 (when the falling edge is selected by bits 1 and 0 in ICSR2) • By writing 0 to POE4F after reading POE4F = 1 after a high level input to POE4 is sampled at Pϕ/8, Pϕ/16, or Pϕ/128 clock (when low-level sampling is selected by bits 1 and 0 in ICSR2) [Setting condition] • When the input condition set by bits 1 and 0 in ICSR2 occurs at the $\overline{\text{POE4}}$ pin
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PIE2	0	R/W	Port Interrupt Enable 2 This bit enables/disables interrupt requests when any one of the POE4F to POE7F bits of the ICSR2 is set to 1. 0: Interrupt requests disabled 1: Interrupt requests enabled
7, 6	POE7M[1:0]	00	R/W*[2]	POE7 mode 1 and 0 These bits select the input mode of the $\overline{\text{POE7}}$ pin. 00: Accept request on falling edge of POE7 input 01: Accept request when POE7 input has been sampled for 16 Pϕ/8 clock pulses and all are at a low level. 10: Accept request when POE7 input has been sampled for 16 Pϕ/16 clock pulses and all are at a low level. 11: Accept request when POE7 input has been sampled for 16 Pϕ/128 clock pulses and all are at a low level.

Bit	Bit Name	Initial value	R/W	Description
5, 4	POE6M[1:0]	00	R/W*[2]	POE6 mode 1 and 0 These bits select the input mode of the $\overline{\text{POE6}}$ pin. 00: Accept request on falling edge of POE6 input 01: Accept request when POE6 input has been sampled for 16 Pϕ/8 clock pulses and all are at a low level. 10: Accept request when POE6 input has been sampled for 16 Pϕ/16 clock pulses and all are at a low level. 11: Accept request when POE6 input has been sampled for 16 Pϕ/128 clock pulses and all are at a low level.
3, 2	POE5M[1:0]	00	R/W*[2]	POE5 mode 1 and 0 These bits select the input mode of the $\overline{\text{POE5}}$ pin. 00: Accept request on falling edge of POE5 input 01: Accept request when POE5 input has been sampled for 16 Pϕ/8 clock pulses and all are at a low level. 10: Accept request when POE5 input has been sampled for 16 Pϕ/16 clock pulses and all are at a low level. 11: Accept request when POE5 input has been sampled for 16 Pϕ/128 clock pulses and all are at a low level.
1, 0	POE4M[1:0]	00	R/W*[2]	POE4 mode 1 and 0 These bits select the input mode of the $\overline{\text{POE4}}$ pin. 00: Accept request on falling edge of POE4 input 01: Accept request when POE4 input has been sampled for 16 Pϕ/8 clock pulses and all are at a low level. 10: Accept request when POE4 input has been sampled for 16 Pϕ/16 clock pulses and all are at a low level. 11: Accept request when POE4 input has been sampled for 16 Pϕ/128 clock pulses and all are at a low level.

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.
2. Can be modified only once after a power-on reset.

13.3.4 Output Level Control/Status Register 2 (OCSR2)

OCSR2 is a 16-bit readable/writable register that controls the enable/disable of both output level comparison and interrupts, and indicates status.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSF2	-	-	-	-	-	OCE2	OIE2	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*1	R	R	R	R	R	R/W*2	R/W	R	R	R	R	R	R	R	R

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.
2. Can be modified only once after a power-on reset.

Bit	Bit Name	Initial value	R/W	Description
15	OSF2	0	R/(W)*1	Output Short Flag 2 This flag indicates that any one of the three pairs of MTU2S 2-phase outputs to be compared has simultaneously become an active level. [Clearing condition] • By writing 0 to OSF2 after reading OSF2 = 1 [Setting condition] • When any one of the three pairs of 2-phase outputs has simultaneously become an active level
14 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	OCE2	0	R/W*2	Output Short High-Impedance Enable 2 This bit specifies whether to place the pins in high-impedance state when the OSF2 bit in OCSR2 is set to 1. 0: Does not place the pins in high-impedance state 1: Places the pins in high-impedance state
8	OIE2	0	R/W	Output Short Interrupt Enable 2 This bit enables or disables interrupt requests when the OSF2 bit in OCSR2 is set to 1. 0: Interrupt requests disabled 1: Interrupt requests enabled

Bit	Bit Name	Initial value	R/W	Description
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.
2. Can be modified only once after a power-on reset.

13.3.5 Input Level Control/Status Register 3 (ICSR3)

ICSR3 is a 16-bit readable/writable register that selects the $\overline{\text{POE8}}$ pin input mode, controls the enable/disable of interrupts, and indicates status.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	POE8F	-	-	POE8E	PIE3	-	-	-	-	-	-	POE8M[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/(W)*1	R	R	R/W*2	R/W	R	R	R	R	R	R	R/W*2	R/W*2

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.
2. Can be modified only once after a power-on reset.

Bit	Bit Name	Initial value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	POE8F	0	R/(W)*1	POE8 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE8}}$ pin. [Clearing conditions] • By writing 0 to POE8F after reading POE8F = 1 (when the falling edge is selected by bits 1 and 0 in ICSR3) • By writing 0 to POE8F after reading POE8F = 1 after a high level input to POE8 is sampled at Pϕ/8, Pϕ/16, or Pϕ/128 clock (when low-level sampling is selected by bits 1 and 0 in ICSR3) [Setting condition] • When the input condition set by bits 1 and 0 in ICSR3 occurs at the $\overline{\text{POE8}}$ pin

Bit	Bit Name	Initial value	R/W	Description
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	POE8E	0	R/W*2	POE8 High-Impedance Enable This bit specifies whether to place the pins in high-impedance state when the POE8F bit in ICSR3 is set to 1. 0: Does not place the pins in high-impedance state 1: Places the pins in high-impedance state
8	PIE3	0	R/W	Port Interrupt Enable 3 This bit enables or disables interrupt requests when the POE8 bit in ICSR3 is set to 1. 0: Interrupt requests disabled 1: Interrupt requests enabled
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	POE8M[1:0]	00	R/W*2	POE8 mode 1 and 0 These bits select the input mode of the $\overline{\text{POE8}}$ pin. 00: Accept request on falling edge of POE8 input 01: Accept request when POE8 input has been sampled for 16 Pϕ/8 clock pulses and all are low level. 10: Accept request when POE8 input has been sampled for 16 Pϕ/16 clock pulses and all are low level. 11: Accept request when POE8 input has been sampled for 16 Pϕ/128 clock pulses and all are low level.

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.
2. Can be modified only once after a power-on reset.

13.3.6 Software Port Output Enable Register (SPOER)

SPOER is an 8-bit readable/writable register that controls high-impedance state of the pins.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	MTU2S HIZ	MTU2 CH0HIZ	MTU2 CH34HIZ
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	MTU2SHIZ	0	R/W	MTU2S Output High-Impedance This bit specifies whether to place the high-current pins for the MTU2S in high-impedance state. 0: Does not place the pins in high-impedance state [Clearing conditions] • Power-on reset • By writing 0 to MTU2SHIZ after reading MTU2SHIZ = 1 1: Places the pins in high-impedance state [Setting condition] • By writing 1 to MTU2SHIZ
1	MTU2CH0HIZ	0	R/W	MTU2 Channel 0 Output High-Impedance This bit specifies whether to place the pins for channel 0 in the MTU2 in high-impedance state. 0: Does not place the pins in high-impedance state [Clearing conditions] • Power-on reset • By writing 0 to MTU2CH0HIZ after reading MTU2CH0HIZ = 1 1: Places the pins in high-impedance state [Setting condition] • By writing 1 to MTU2CH0HIZ

Bit	Bit Name	Initial value	R/W	Description
0	MTU2CH34HIZ	0	R/W	MTU2 Channel 3 and 4 Output High-Impedance This bit specifies whether to place the high-current pins for the MTU2 in high-impedance state. 0: Does not place the pins in high-impedance state [Clearing conditions] • Power-on reset • By writing 0 to MTU2CH34HIZ after reading MTU2CH34HIZ = 1 1: Places the pins in high-impedance state [Setting condition] • By writing 1 to MTU2CH34HIZ

13.3.7 Port Output Enable Control Register 1 (POECR1)

POECR1 is an 8-bit readable/writable register that controls high-impedance state of the pins.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	MTU2 PE3ZE	MTU2 PE2ZE	MTU2 PE1ZE	MTU2 PE0ZE
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W*	R/W*	R/W*	R/W*

Note: * Can be modified only once after a power-on reset.

Bit	Bit Name	Initial value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	MTU2PE3ZE	0	R/W*	MTU2 PE3 High-Impedance Enable This bit specifies whether to place the PE3/TIOC0D pin for channel 0 in the MTU2 in high-impedance state when either POE8F or MTU2CH0HIZ bit is set to 1. 0: Does not place the pin in high-impedance state 1: Places the pin in high-impedance state

Bit	Bit Name	Initial value	R/W	Description
2	MTU2PE2ZE	0	R/W*	MTU2 PE2 High-Impedance Enable This bit specifies whether to place the PE2/TIOC0C pin for channel 0 in the MTU2 in high-impedance state when either POE8F or MTU2CH0HIZ bit is set to 1. 0: Does not place the pin in high-impedance state 1: Places the pin in high-impedance state
1	MTU2PE1ZE	0	R/W*	MTU2 PE1 High-Impedance Enable This bit specifies whether to place the PE1/TIOC0B pin for channel 0 in the MTU2 in high-impedance state when either POE8F or MTU2CH0HIZ bit is set to 1. 0: Does not place the pin in high-impedance state 1: Places the pin in high-impedance state
0	MTU2PE0ZE	0	R/W*	MTU2 PE0 High-Impedance Enable This bit specifies whether to place the PE0/TIOC0A pin for channel 0 in the MTU2 in high-impedance state when either POE8F or MTU2CH0HIZ bit is set to 1. 0: Does not place the pin in high-impedance state 1: Places the pin in high-impedance state

Note: * Can be modified only once after a power-on reset.

13.3.8 Port Output Enable Control Register 2 (POECR2)

POECR2 is a 16-bit readable/writable register that controls high-impedance state of the pins.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	MTU2 P1CZE	MTU2 P2CZE	MTU2 P3CZE	-	MTU2S P1CZE	MTU2S P2CZE	MTU2S P3CZE	-	MTU2S P4CZE	MTU2S P5CZE	MTU2S P6CZE	-	MTU2S P7CZE	MTU2S P8CZE	MTU2S P9CZE
Initial value:	0	1	1	1	0	1	1	1	0	0	0	0	0	0	0	0
R/W:	R	R/W*	R/W*	R/W*	R	R/W*	R/W*	R/W*	R	R/W*	R/W*	R/W*	R	R/W*	R/W*	R/W*

Note: * Can be modified only once after a power-on reset.

Bit	Bit Name	Initial value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	MTU2P1CZE	1	R/W*	MTU2 Port 1 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2 high-current PE9/TIOC3B and PE11/TIOC3D pins and to place them in high-impedance state when the OSF1 bit is set to 1 while the OCE1 bit is 1 or when any one of the POE0F, POE1F, POE2F, POE3F, and MTU2CH34HIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state

Bit	Bit Name	Initial value	R/W	Description
13	MTU2P2CZE	1	R/W*	MTU2 Port 2 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2 high-current PE12/TIOC4A and PE14/TIOC4C pins and to place them in high-impedance state when the OSF1 bit is set to 1 while the OCE1 bit is 1 or when any one of the POE0F, POE1F, POE2F, POE3F, and MTU2CH34HIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state
12	MTU2P3CZE	1	R/W*	MTU2 Port 3 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2 high-current PE13/TIOC4B and PE15/TIOC4D pins and to place them in high-impedance state when the OSF1 bit is set to 1 while the OCE1 bit is 1 or when any one of the POE0F, POE1F, POE2F, POE3F, and MTU2CH34HIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial value	R/W	Description
10	MTU2SP1CZE	1	R/W*	MTU2S Port 1 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2S high-current PE16/TIOC3BS and PE17/TIOC3DS pins and to place them in high-impedance state when the OSF2 bit is set to 1 while the OCE2 bit is 1 or when any one of the POE4F, POE5F, POE6F, POE7F, and MTU2SHIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state
9	MTU2SP2CZE	1	R/W*	MTU2S Port 2 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2S high-current PE18/TIOC4AS and PE20/TIOC4CS pins and to place them in high-impedance state when the OSF2 bit is set to 1 while the OCE2 bit is 1 or when any one of the POE4F, POE5F, POE6F, POE7F, and MTU2SHIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state
8	MTU2SP3CZE	1	R/W*	MTU2S Port 3 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2S high-current PE19/TIOC4BS and PE21/TIOC4DS pins and to place them in high-impedance state when the OSF2 bit is set to 1 while the OCE2 bit is 1 or when any one of the POE4F, POE5F, POE6F, POE7F, and MTU2SHIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	MTU2SP4CZE	0	R/W*	MTU2S Port 4 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2S high-current PD9/TIOC3BS and PD11/TIOC3DS pins and to place them in high-impedance state when the OSF2 bit is set to 1 while the OCE2 bit is 1 or when any one of the POE4F, POE5F, POE6F, POE7F, and MTU2SHIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state
5	MTU2SP5CZE	0	R/W*	MTU2S Port 5 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2S high-current PD12/TIOC4AS and PD14/TIOC4CS pins and to place them in high-impedance state when the OSF2 bit is set to 1 while the OCE2 bit is 1 or when any one of the POE4F, POE5F, POE6F, POE7F, and MTU2SHIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state

Bit	Bit Name	Initial value	R/W	Description
4	MTU2SP6CZE	0	R/W*	MTU2S Port 6 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2S high-current PD13/TIOC4BS and PD15/TIOC4DS pins and to place them in high-impedance state when the OSF2 bit is set to 1 while the OCE2 bit is 1 or when any one of the POE4F, POE5F, POE6F, POE7F, and MTU2SHIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	MTU2SP7CZE	0	R/W*	MTU2S Port 7 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2S high-current PD29/TIOC3BS and PD28/TIOC3DS pins and to place them in high-impedance state when the OSF2 bit is set to 1 while the OCE2 bit is 1 or when any one of the POE4F, POE5F, POE6F, POE7F, and MTU2SHIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state

Bit	Bit Name	Initial value	R/W	Description
1	MTU2SP8CZE	0	R/W*	MTU2S Port 8 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2S high-current PD27/TIOC4AS and PD25/TIOC4CS pins and to place them in high-impedance state when the OSF2 bit is set to 1 while the OCE2 bit is 1 or when any one of the POE4F, POE5F, POE6F, POE7F, and MTU2SHIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state
0	MTU2SP9CZE	0	R/W*	MTU2S Port 9 Output Comparison/High-Impedance Enable This bit specifies whether to compare output levels for the MTU2S high-current PD26/TIOC4BS and PD24/TIOC4DS pins and to place them in high-impedance state when the OSF2 bit is set to 1 while the OCE2 bit is 1 or when any one of the POE4F, POE5F, POE6F, POE7F, and MTU2SHIZ bits is set to 1. 0: Does not compare output levels or place the pins in high-impedance state 1: Compares output levels and places the pins in high-impedance state

Note: * Can be modified only once after a power-on reset.

13.4 Operation

Table 13.4 shows the target pins for high-impedance control and conditions to place the pins in high-impedance state.

Table 13.4 Target Pins and Conditions for High-Impedance Control

Pins	Conditions	Detailed Conditions
MTU2 high-current pins (PE9/TIOC3B and PE11/TIOC3D)	Input level detection, output level comparison, or SPOER setting	MTU2P1CZE • ((POE3F + POE2F + POE1F + POE0F) + (OSF1 • OCE1) + (MTU2CH34HIZ))
MTU2 high-current pins (PE12/TIOC4A and PE14/TIOC4C)	Input level detection, output level comparison, or SPOER setting	MTU2P2CZE • ((POE3F + POE2F + POE1F + POE0F) + (OSF1 • OCE1) + (MTU2CH34HIZ))
MTU2 high-current pins (PE13/TIOC4B and PE15/TIOC4D)	Input level detection, output level comparison, or SPOER setting	MTU2P3CZE • ((POE3F + POE2F + POE1F + POE0F) + (OSF1 • OCE1) + (MTU2CH34HIZ))
MTU2S high-current pins (PE16/TIOC3BS and PE17/TIOC3DS)	Input level detection, output level comparison, or SPOER setting	MTU2SP1CZE • ((POE4F + POE5F + POE6F + POE7F) + (OSF2 • OCE2) + (MTU2SHIZ))
MTU2S high-current pins (PE18/TIOC4AS and PE20/TIOC4CS)	Input level detection, output level comparison, or SPOER setting	MTU2SP2CZE • ((POE4F + POE5F + POE6F + POE7F) + (OSF2 • OCE2) + (MTU2SHIZ))
MTU2S high-current pins (PE19/TIOC4BS and PE21/TIOC4DS)	Input level detection, output level comparison, or SPOER setting	MTU2SP3CZE • ((POE4F + POE5F + POE6F + POE7F) + (OSF2 • OCE2) + (MTU2SHIZ))
MTU2S high-current pins (PD9/TIOC3BS and PD11/TIOC3DS)	Input level detection, output level comparison, or SPOER setting	MTU2SP4CZE • ((POE4F + POE5F + POE6F + POE7F) + (OSF2 • OCE2) + (MTU2SHIZ))
MTU2S high-current pins (PD12/TIOC4AS and PD14/TIOC4CS)	Input level detection, output level comparison, or SPOER setting	MTU2SP5CZE • ((POE4F + POE5F + POE6F + POE7F) + (OSF2 • OCE2) + (MTU2SHIZ))
MTU2S high-current pins (PD13/TIOC4BS and PD15/TIOC4DS)	Input level detection, output level comparison, or SPOER setting	MTU2SP6CZE • ((POE4F + POE5F + POE6F + POE7F) + (OSF2 • OCE2) + (MTU2SHIZ))
MTU2S high-current pins (PD29/TIOC3BS and PD28/TIOC3DS)	Input level detection, output level comparison, or SPOER setting	MTU2SP7CZE • ((POE4F + POE5F + POE6F + POE7F) + (OSF2 • OCE2) + (MTU2SHIZ))

Pins	Conditions	Detailed Conditions
MTU2S high-current pins (PD27/TIOC4AS and PD25/TIOC4CS)	Input level detection, output level comparison, or SPOER setting	MTU2SP8CZE • ((POE4F + POE5F + POE6F + POE7F) + (OSF2 • OCE2) + (MTU2SHIZ))
MTU2S high-current pins (PD26/TIOC4BS and PD24/TIOC4DS)	Input level detection, output level comparison, or SPOER setting	MTU2SP9CZE • ((POE4F + POE5F + POE6F + POE7F) + (OSF2 • OCE2) + (MTU2SHIZ))
MTU2 channel 0 pin (PE0/TIOC0A)	Input level detection or SPOER setting	MTU2PE0ZE ((POE8F • POE8E) + (MTU2CH0HIZ))
MTU2 channel 0 pin (PE1/TIOC0B)	Input level detection or SPOER setting	MTU2PE1ZE ((POE8F • POE8E) + (MTU2CH0HIZ))
MTU2 channel 0 pin (PE2/TIOC0C)	Input level detection or SPOER setting	MTU2PE2ZE ((POE8F • POE8E) + (MTU2CH0HIZ))
MTU2 channel 0 pin (PE3/TIOC0D)	Input level detection or SPOER setting	MTU2PE3ZE ((POE8F • POE8E) + (MTU2CH0HIZ))

13.4.1 Input Level Detection Operation

If the input conditions set by ICSR1 to ICSR3 occur on the $\overline{\text{POE0}}$ to $\overline{\text{POE8}}$ pins, the high-current pins and the pins for channel 0 of the MTU2 are placed in high-impedance state. Note however, that these high-current and MTU2 pins enter high-impedance state only when general input/output function, MTU2 function, or MTU2S function is selected for these pins.

(1) Falling Edge Detection

When a change from a high to low level is input to the $\overline{\text{POE0}}$ to $\overline{\text{POE8}}$ pins, the high-current pins and the pins for channel 0 of the MTU2 are placed in high-impedance state. Figure 13.2 shows a sample timing after the level changes in input to the $\overline{\text{POE0}}$ to $\overline{\text{POE8}}$ pins until the respective pins enter high-impedance state.

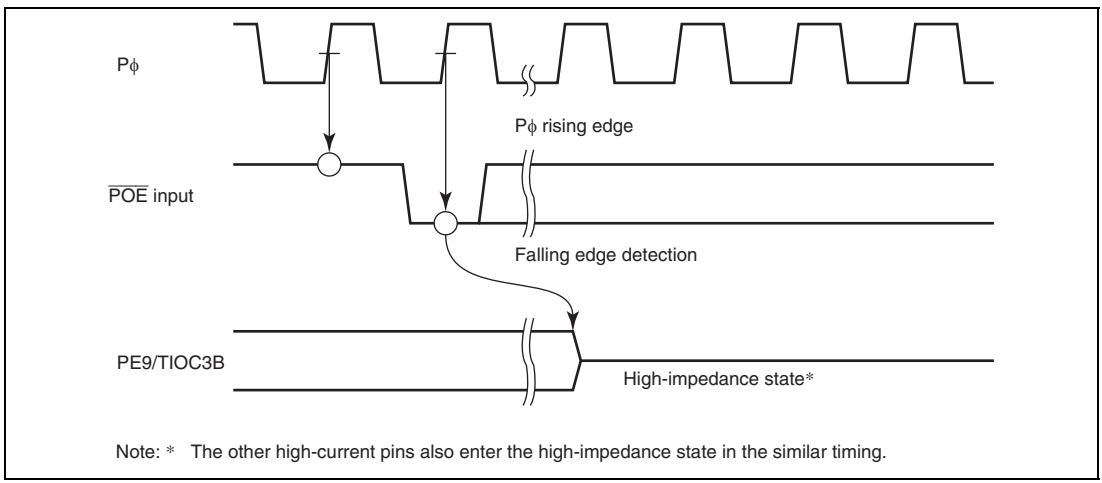


Figure 13.2 Falling Edge Detection

(2) Low-Level Detection

Figure 13.3 shows the low-level detection operation. Sixteen continuous low levels are sampled with the sampling clock selected by ICSR1 to ICSR3. If even one high level is detected during this interval, the low level is not accepted.

The timing when the high-current pins enter the high-impedance state after the sampling clock is input is the same in both falling-edge detection and in low-level detection.

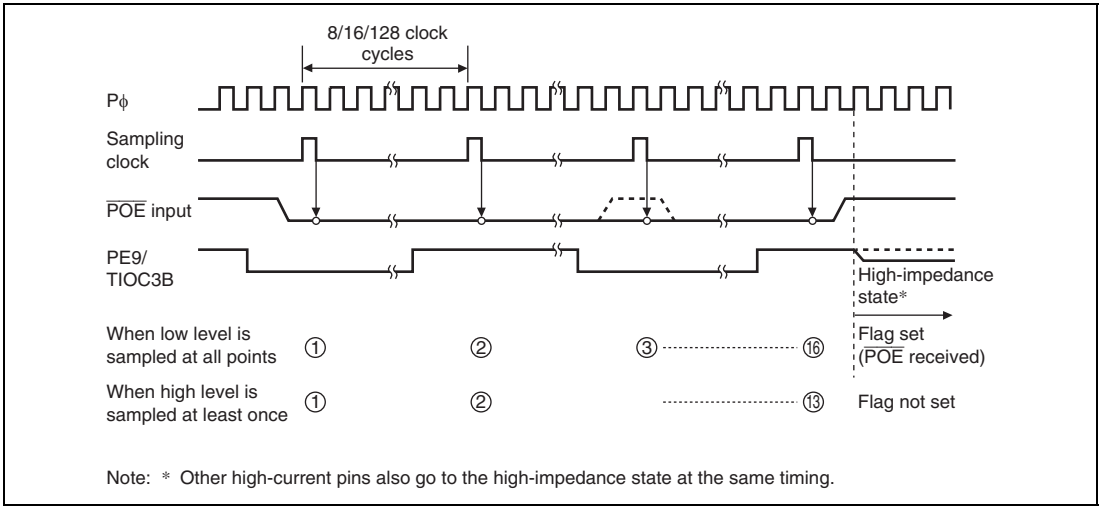


Figure 13.3 Low-Level Detection Operation

13.4.2 Output-Level Compare Operation

Figure 13.4 shows an example of the output-level compare operation for the combination of TIOC3B and TIOC3D. The operation is the same for the other pin combinations.

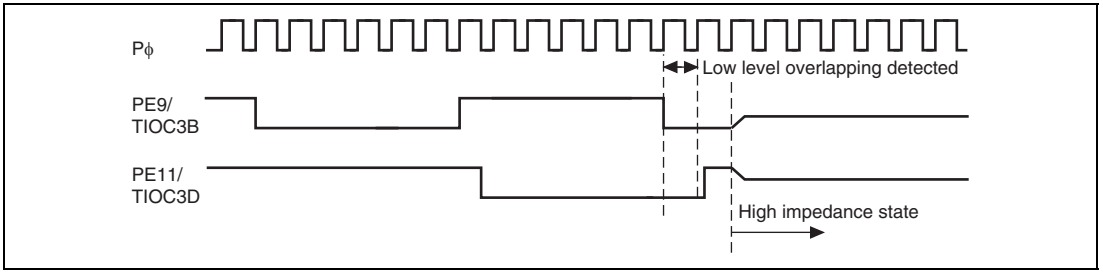


Figure 13.4 Output-Level Compare Operation

13.4.3 Release from High-Impedance State

High-current pins that have entered high-impedance state due to input-level detection can be released either by returning them to their initial state with a power-on reset, or by clearing all of the flags in bits 12 to 15 (POE0F to POE8F) of ICSR1 to ICSR3. However, note that when low-level sampling is selected by bits 0 to 7 in ICSR1 to ICSR3, just writing 0 to a flag is ignored (the flag is not cleared); flags can be cleared by writing 0 to it only after a high level is input to the POE pin and is sampled.

High-current pins that have entered high-impedance state due to output-level detection can be released either by returning them to their initial state with a power-on reset, or by clearing the flag in bit 15 (OCF1 and OCF2) in OCSR1 and OCSR2. However, note that just writing 0 to a flag is ignored (the flag is not cleared); flags can be cleared only after an inactive level is output from the high-current pins. Inactive-level outputs can be achieved by setting the MTU2 and MTU2S internal registers.

13.5 Interrupts

The POE issues a request to generate an interrupt when the specified condition is satisfied during input level detection or output level comparison. Table 13.5 shows the interrupt sources and their conditions.

Table 13.5 Interrupt Sources and Conditions

Name	Interrupt Source	Interrupt Flag	Condition
OEI1	Output enable interrupt 1	POE3F, POE2F, POE1F, POE0F, and OSF1	PIE1 • (POE3F + POE2F + POE1F + POE0F) + OIE1 • OSF1
OEI3	Output enable interrupt 3	POE8F	PIE3 • POE8F
OEI2	Output enable interrupt 2	POE4F, POE5F, POE6F, POE7F, and OSF2	PIE2 • (POE4F + POE5F + POE6F + POE7F) + OIE2 • OSF2

13.6 Usage Note

13.6.1 Pin State when a Power-On Reset Is Issued from the Watchdog Timer

When a power-on reset is issued from the watchdog timer (WDT), initialization of the pin function controller (PFC) sets initial values that select the general input function for the I/O ports. However, when a power-on reset is issued from the WDT while a pin is being handled as high impedance by the port output enable (POE), the pin is placed in the output state for one cycle of the peripheral clock (Pϕ), after which the function is switched to general input.

This also occurs when a power-on reset is issued from the WDT for pins that are being handled as high impedance due to short-circuit detection by the MTU2 and MTU2S.

Figure 13.5 shows the state of a pin for which the POE input has selected high impedance handling with the timer output selected when a power-on reset is issued from the WDT.

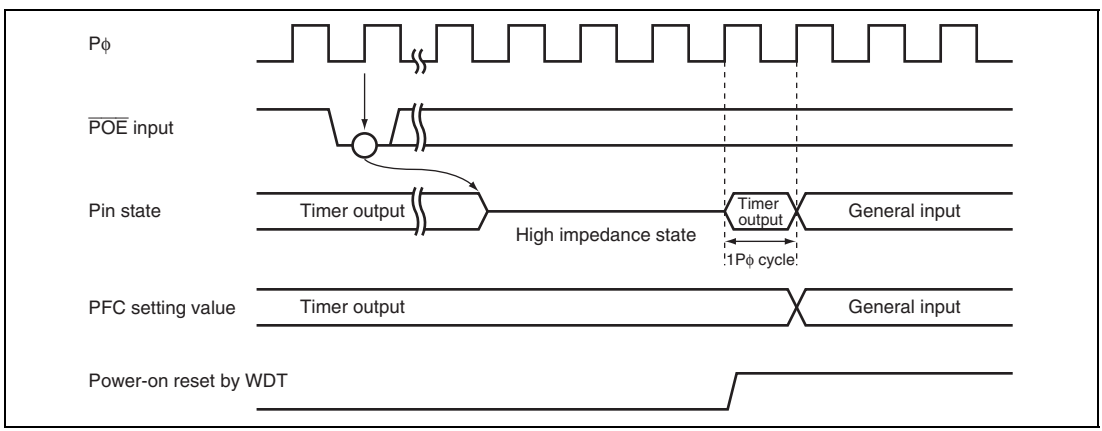


Figure 13.5 Pin State when a Power-On Reset Is Issued from the Watchdog Timer

Section 14 Watchdog Timer (WDT)

This LSI includes the watchdog timer (WDT).

This LSI can be reset by the overflow of the counter when the value of the counter has not been updated because of a system runaway.

The watchdog timer (WDT) is a single-channel timer that uses a peripheral clock as an input and counts the clock settling time when revoking software standby mode. It can also be used as an interval timer.

14.1 Features

- Can be used to ensure the clock settling time: Use the WDT to revoke software standby mode.
- Can switch between watchdog timer mode and interval timer mode.
- Generates internal resets in watchdog timer mode: Internal resets occur after counter overflow.
- An interrupt is generated in interval timer mode
  An interval timer interrupt is generated when the counter overflows.
- Choice of eight counter input clocks
  Eight clocks (×1 to ×1/4096) that are obtained by dividing the peripheral clock can be chosen.
- Choice of two resets
  Power-on reset and manual reset are available.

Figure 14.1 shows a block diagram of the WDT.

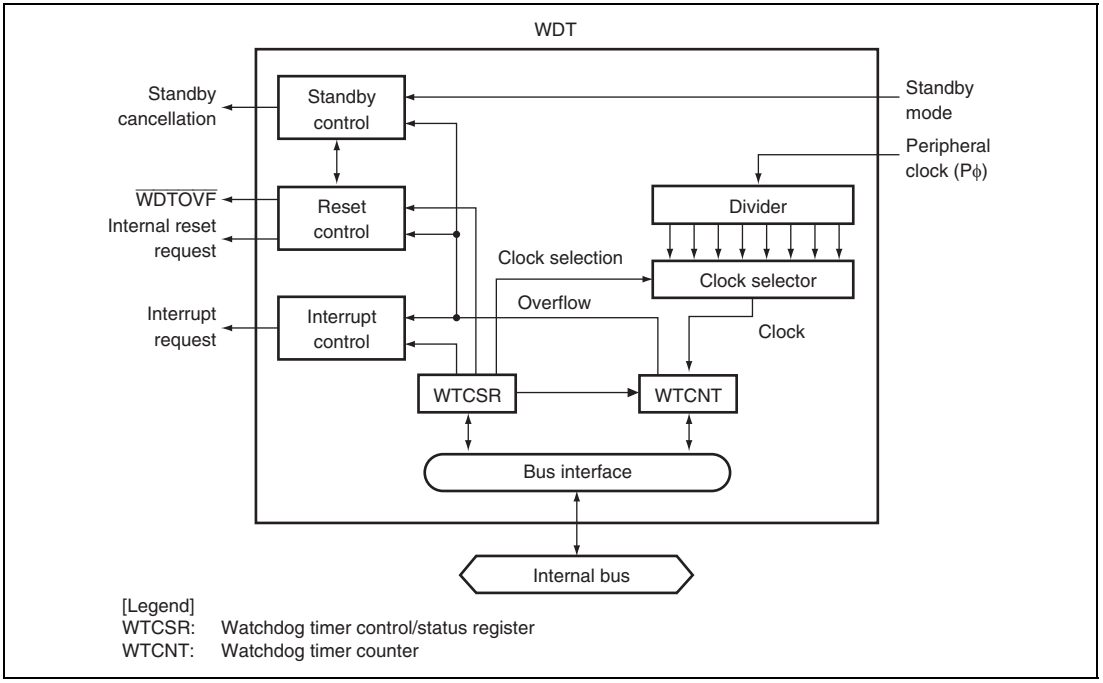


Figure 14.1 Block Diagram of WDT

14.2 Input/Output Pin for WDT

Table 14.1 lists the WDT pin configuration.

Table 14.1 WDT Pin Configuration

Pin Name	Symbol	I/O	Description
Watchdog timer overflow	$\overline{\text{WDTOVF}}$	Output	When an overflow occurs in watchdog timer mode, an internal reset is generated and this pin outputs the low level for one clock cycle specified by the CKS2 to CKS0 bits in WTCSR.

14.3 Register Descriptions

The WDT has the following two registers. Refer to section 27, List of Registers, for the details of the addresses of these registers and the state of registers in each operating mode.

Table 14.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Watchdog timer counter	WTCNT	R/W	H'00	H'FFFFE810	8, 16
Watchdog timer control/status register	WTCSR	R/W	H'00	H'FFFFE812	8, 16

14.3.1 Watchdog Timer Counter (WTCNT)

WTCNT is an 8-bit readable/writable register that increments on the selected clock. When an overflow occurs, it generates a reset in watchdog timer mode and an interrupt in interval time mode. The WTCNT counter is not initialized by an internal reset due to the WDT overflow. The WTCNT counter is initialized to H'00 only by a power-on reset using the $\overline{\text{RES}}$ pin. Use a word access to write to the WTCNT counter, with H'5A in the upper byte. Use a byte access to read WTCNT.

Note: WTCNT differs from other registers in that it is more difficult to write to. See section 14.3.3, Notes on Register Access, for details.

Bit:	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

14.3.2 Watchdog Timer Control/Status Register (WTCSR)

WTCSR is an 8-bit readable/writable register composed of bits to select the clock used for the count, bits to select the timer mode, and overflow flags. WTCSR holds its value in an internal reset due to the WDT overflow. WTCSR is initialized to H'00 only by a power-on reset using the $\overline{\text{RES}}$ pin.

When used to count the clock settling time for revoking a software standby, it retains its value after counter overflow. Use a word access to write to WTCSR, with H'A5 in the upper byte. Use a byte access to read WTCSR.

Note: WTCSR differs from other registers in that it is more difficult to write to. See section 14.3.3, Notes on Register Access, for details.

Bit:	7	6	5	4	3	2	1	0
	TME	WT/$\overline{\text{IT}}$	RSTS	WOVF	IOVF	CKS[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TME	0	R/W	Timer Enable Starts and stops timer operation. Clear this bit to 0 when using the WDT to revoke software standby mode. 0: Timer disabled: Count-up stops and WTCNT value is retained 1: Timer enabled
6	WT/$\overline{\text{IT}}$	0	R/W	Timer Mode Select Selects whether to use the WDT as a watchdog timer or an interval timer. 0: Interval timer mode 1: Watchdog timer mode Note: When the WTCNT overflows in watchdog timer mode, the $\overline{\text{WDTOVF}}$ signal is output externally. If this bit is modified when the WDT is running, the up-count may not be performed correctly.

Bit	Bit Name	Initial Value	R/W	Description
5	RSTS	0	R/W	Reset Select Selects the type of reset when the WTCNT overflows in watchdog timer mode. In interval timer mode, this setting is ignored. 0: Power-on reset 1: Manual reset
4	WOVF	0	R/W	Watchdog Timer Overflow Indicates that the WTCNT has overflowed in watchdog timer mode. This bit is not set in interval timer mode. 0: No overflow 1: WTCNT has overflowed in watchdog timer mode
3	IOVF	0	R/W	Interval Timer Overflow Indicates that the WTCNT has overflowed in interval timer mode. This bit is not set in watchdog timer mode. 0: No overflow 1: WTCNT has overflowed in interval timer mode
2 to 0	CKS[2:0]	000	R/W	Clock Select 2 to 0 These bits select the clock to be used for the WTCNT count from the eight types obtainable by dividing the peripheral clock (Pϕ). The overflow period that is shown inside the parenthesis in the table is the value when the peripheral clock (Pϕ) is 40 MHz. 000: Pϕ (6.4 μs) 001: Pϕ /4 (25.6 μs) 010: Pϕ /16 (102.4 μs) 011: Pϕ /32 (204.8 μs) 100: Pϕ /64 (409.6 μs) 101: Pϕ /256 (1.64 ms) 110: Pϕ /1024 (6.55 ms) 111: Pϕ /4096 (26.21 ms) Note: If bits CKS2 to CKS0 are modified when the WDT is operating, the up-count may not be performed correctly. Ensure that these bits are modified only when the WDT is not operating.

14.3.3 Notes on Register Access

The watchdog timer counter (WTCNT) and watchdog timer control/status register (WTCSR) are more difficult to write to than other registers. The procedures for reading or writing to these registers are given below.

(1) Writing to WTCNT and WTCSR

These registers must be written by a word transfer instruction. They cannot be written by a byte or longword transfer instruction. When writing to WTCNT, set the upper byte to H'5A and transfer the lower byte as the write data, as shown in figure 14.2. When writing to WTCSR, set the upper byte to H'A5 and transfer the lower byte as the write data. This transfer procedure writes the lower byte data to WTCNT or WTCSR.

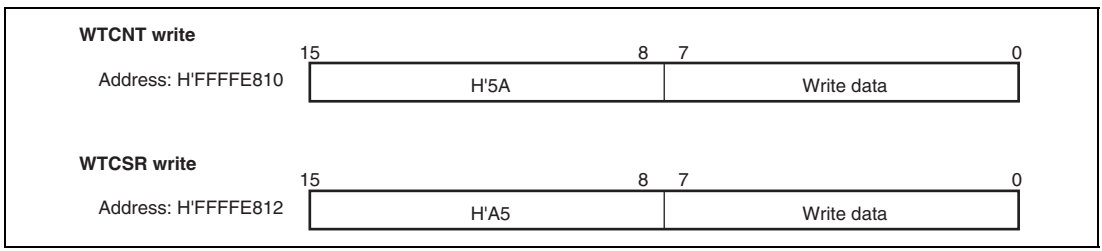


Figure 14.2 Writing to WTCNT and WTCSR

(2) Reading from WTCNT and WTCSR

WTCNT and WTCSR are read in a method similar to other registers. WTCSR is allocated to address H'FFFFE812 and WTCNT to address H'FFFFE810. Byte transfer instructions must be used for reading from these registers.

14.4 Operation

14.4.1 Revoking Software Standbys

The WDT can be used to revoke software standby mode with an NMI interrupt or external interrupt (IRQ). The procedure is described below. (The WDT does not run when resets are used for revoking, so keep the $\overline{\text{RES}}$ or $\overline{\text{MRES}}$ pin low until the clock stabilizes.)

1. Before transition to software standby mode, always clear the TME bit in WTCSR to 0. When the TME bit is 1, an erroneous reset or interval timer interrupt may be generated when the count overflows.
2. Set the type of count clock used in the CKS2 to CKS0 bits in WTCSR and the initial values for the counter in the WTCNT counter. These values should ensure that the time till count overflow is longer than the clock oscillation settling time.
3. After setting the STBY bit to 1 in standby control register 1 (STBCR1: see section 26, Power-Down Modes) and the STBYMD bit to 1 in standby control register 6 (STBCR6: see section 26, Power-Down Modes), execute a SLEEP instruction to transition to software standby mode and stop the clock.
4. The WDT starts counting by detecting a change in the level input to the NMI or IRQ pin.
5. When the WDT count overflows, the CPG starts supplying the clock and the LSI resumes operation. The WOVF flag in WTCSR is not set when this happens.

14.4.2 Using Watchdog Timer Mode

While operating in watchdog timer mode, the WDT generates an internal reset of the type specified by the RSTS bit in WTCSR and asserts a signal through the $\overline{\text{WDTOVF}}$ pin every time the counter overflows.

1. Set the WT/$\overline{\text{IT}}$ bit in WTCSR to 1, set the reset type in the RSTS bit, set the type of count clock in the CKS2 to CKS0 bits, and set the initial value of the counter in the WTCNT counter.
2. Set the TME bit in WTCSR to 1 to start the count in watchdog timer mode.
3. While operating in watchdog timer mode, rewrite the counter periodically to prevent the counter from overflowing.
4. When the counter overflows, the WDT sets the WOVF flag in WTCSR to 1, asserts a signal through the $\overline{\text{WDTOVF}}$ pin for one cycle of the count clock specified by the CKS2 to CKS0 bits, and generates a reset of the type specified by the RSTS bit. The counter then resumes counting.

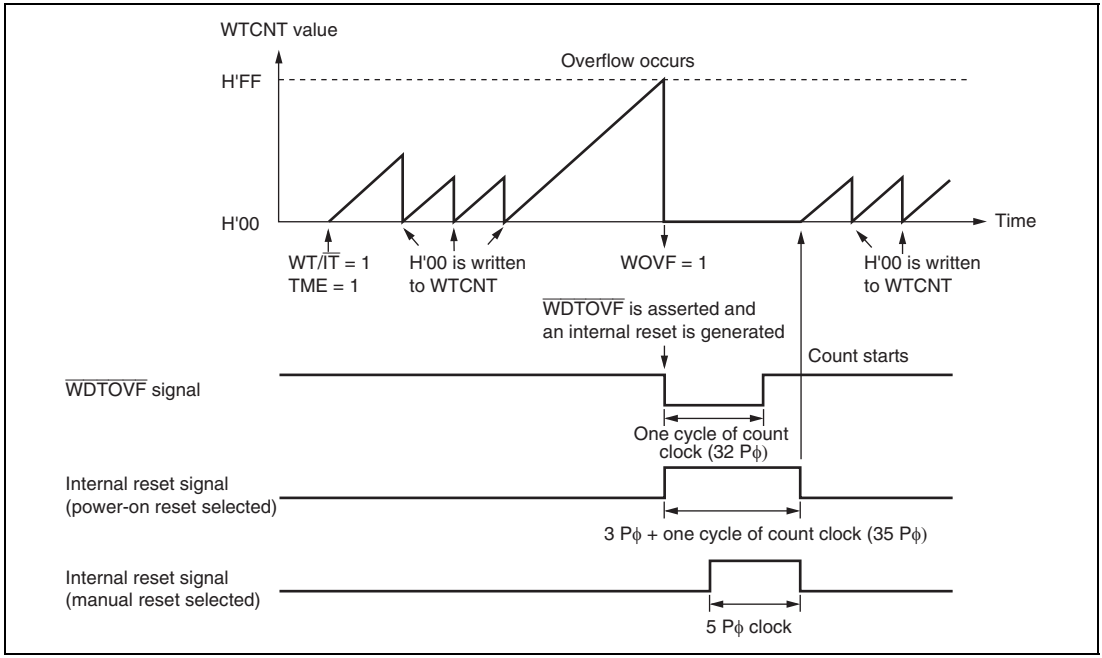


Figure 14.3 Operation in Watchdog Timer Mode (When WTCNT Count Clock Is Specified to Pϕ/32 by CKS2 to CKS0)

14.4.3 Using Interval Timer Mode

When operating in interval timer mode, interval timer interrupts are generated at every overflow of the counter. This enables interrupts to be generated at set periods.

1. Clear the WT/$\overline{IT}$ bit in WTCSR to 0, set the type of count clock in the CKS2 to CKS0 bits, and set the initial value of the counter in the WTCNT counter.
2. Set the TME bit in WTCSR to 1 to start the count in interval timer mode.
3. When the counter overflows, the WDT sets the IOVF flag in WTCSR to 1 and an interval timer interrupt request is sent to the INTC. The counter then resumes counting.

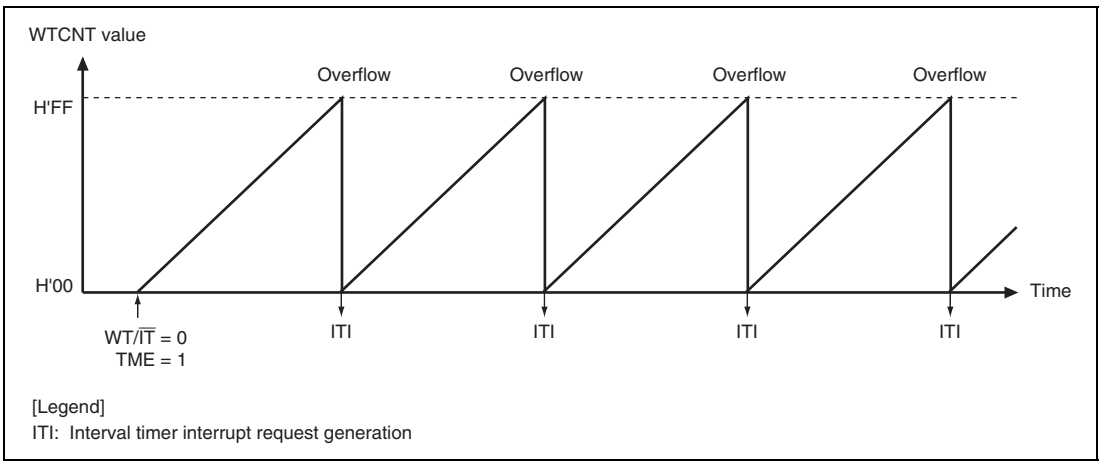


Figure 14.4 Operation in Interval Timer Mode

14.5 Interrupt Source

The WDT has one interrupt source: the interval timer interrupt (ITI).

Table 14.3 shows this interrupt source. An interval timer interrupt (ITI) is generated when the interval timer overflow flag bit (IOVF) in the watchdog timer control status register (WTSCR) is set to 1.

The interrupt request is canceled by clearing the interrupt flag to 0.

Table 14.3 Interrupt Source

Name	Interrupt Source	Interrupt Enable Bit	Interrupt Flag Bit
ITI	Interval timer interrupt	—	Interval timer overflow flag (IOVF)

14.6 Usage Notes

Pay attention to the following points when using the WDT in either the interval timer or watchdog timer mode.

14.6.1 WTCNT Setting Value

If the timer is stopped and WTCNT is set to H'FF in interval timer mode, an overflow does not occur when WTCNT changes from H'FF to H'00 after one cycle of the count clock, but an overflow does occur when WTCNT changes from H'FE to H'FF after 256 cycles of the count clock. If the timer is operating and WTCNT is set to H'FF, an interval timer interrupt is generated immediately.

Do not set WTCNT to H'FF in watchdog timer mode. If WTCNT is set to H'FF, a WDT reset is generated immediately, regardless of the clock selected by bits CKS[2:0]. In this case, the assertion periods of the $\overline{\text{WDTOVF}}$ signal and internal reset signal are shortened.

14.6.2 Timer Variation

After timer operation has started, the period from the power-on reset point to the first count up timing of WTCNT varies depending on the time period that is set by the TME bit of WTCSR. The shortest such time period is thus one cycle of the peripheral clock, Pϕ, while the longest is the result of frequency division according to the value in the CKS[2:0] bits. The timing of subsequent incrementation is in accord with the selected frequency division ratio. Accordingly, this time difference is referred to as timer variation.

This also applies to the timing of the first incrementation after WTCNT has been written to during timer operation.

14.6.3 System Reset by $\overline{\text{WDTOVF}}$ Signal

If the $\overline{\text{WDTOVF}}$ signal is input to the $\overline{\text{RES}}$ pin of this LSI, this LSI cannot be initialized correctly.

Avoid input of the $\overline{\text{WDTOVF}}$ signal to the $\overline{\text{RES}}$ pin of this LSI through glue logic circuits. To reset the entire system with the $\overline{\text{WDTOVF}}$ signal, use the circuit shown in figure 14.5.

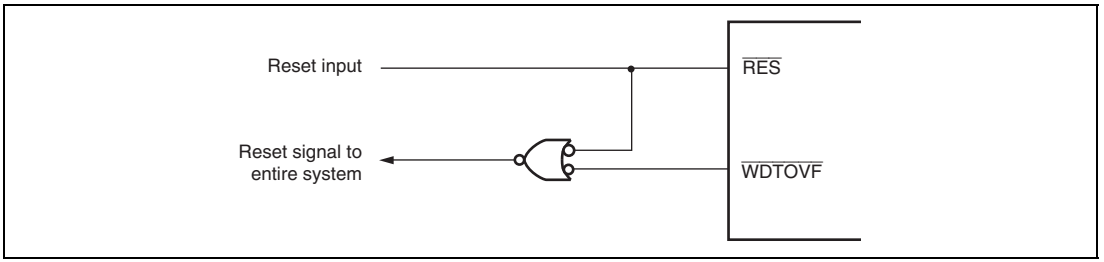


Figure 14.5 Example of System Reset Circuit Using $\overline{\text{WDTOVF}}$ Signal

14.6.4 Manual Reset in Watchdog Timer Mode

When a manual reset occurs in watchdog timer mode, the current bus cycle is continued. If a manual reset occurs while the bus is released or during DMAC burst transfer, manual reset exception handling will be pended until the CPU acquires the bus mastership.

14.6.5 Internal Reset in Watchdog Timer Mode

When an internal reset is generated by an overflow of the watchdog timer counter (WTCNT) in watchdog timer mode, the watchdog timer control/status register (WTCSR) is not initialized and its value is maintained.

Section 15 Serial Communication Interface (SCI)

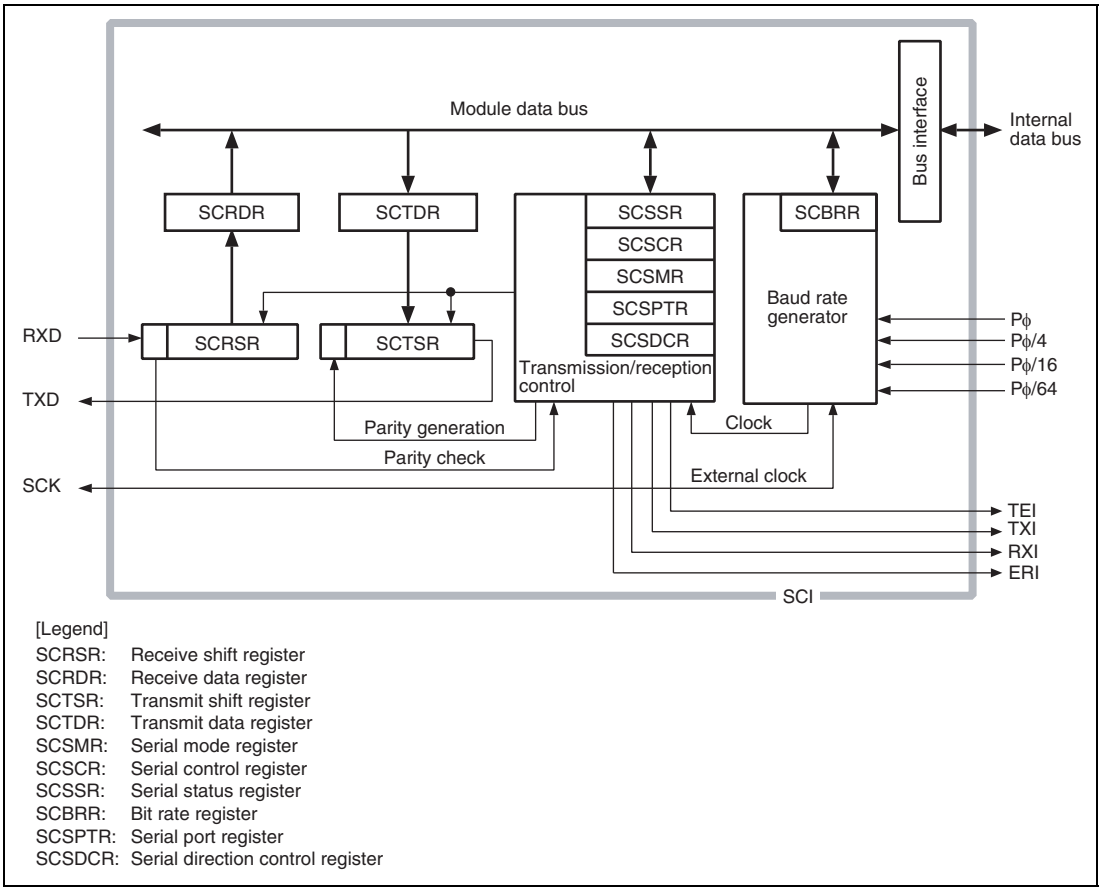
This LSI has three independent serial communication interface (SCI) channels. The SCI can handle both asynchronous and clock synchronous serial communication. In asynchronous serial communication mode, serial data communication can be carried out with standard asynchronous communication chips such as a Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communication Interface Adapter (ACIA). A function is also provided for serial communication between processors (multiprocessor communication function).

15.1 Features

- Choice of asynchronous or clock synchronous serial communication mode
- Asynchronous mode:
  - Serial data communication is performed by start-stop in character units. The SCIF can communicate with a universal asynchronous receiver/transmitter (UART), an asynchronous communication interface adapter (ACIA), or any other communications chip that employs a standard asynchronous serial system. There are twelve selectable serial data communication formats.
  - Data length: 7 or 8 bits
  - Stop bit length: 1 or 2 bits
  - Parity: Even, odd, or none
  - Multiprocessor communications
  - Receive error detection: Parity, overrun, and framing errors
  - Break detection: Break is detected by reading the RXD pin level directly when a framing error occurs.
- Clock synchronous mode:
  - Serial data communication is synchronized with a clock signal. The SCIF can communicate with other chips having a clock synchronous communication function.
  - Data length: 8 bits
  - Receive error detection: Overrun errors
- Full duplex communication: The transmitting and receiving sections are independent, so the SCI can transmit and receive simultaneously. Both sections use double buffering, so high-speed continuous data transfer is possible in both the transmit and receive directions.
- On-chip baud rate generator with selectable bit rates
- Internal or external transmit/receive clock source: From either baud rate generator (internal clock) or SCK pin (external clock)

- Choice of LSB-first or MSB-first data transfer (except for 7-bit data in asynchronous mode)
- Four types of interrupts: There are four interrupt sources, transmit-data-empty, transmit end, receive-data-full, and receive error interrupts, and each interrupt can be requested independently. The direct memory access controller (DMAC) or the data transfer controller (DTC) can be activated by the transmit-data-empty interrupt or receive-data-full interrupt to transfer data.
- Module standby mode can be set

Figure 15.1 shows a block diagram of the SCI.



[Legend]
SCRSR: Receive shift register
SCRDR: Receive data register
SCTSR: Transmit shift register
SCTDR: Transmit data register
SCSMR: Serial mode register
SCSCR: Serial control register
SCSSR: Serial status register
SCBRR: Bit rate register
SCSPTR: Serial port register
SCSDCR: Serial direction control register

Figure 15.1 Block Diagram of SCI

15.2 Input/Output Pins

The SCI has the serial pins summarized in table 15.1.

Table 15.1 Pin Configuration

Channel	Pin Name*	I/O	Function
0	SCK0	I/O	SCI0 clock input/output
	RXD0	Input	SCI0 receive data input
	TXD0	Output	SCI0 transmit data output
1	SCK1	I/O	SCI1 clock input/output
	RXD1	Input	SCI1 receive data input
	TXD1	Output	SCI1 transmit data output
2	SCK2	I/O	SCI2 clock input/output
	RXD2	Input	SCI2 receive data input
	TXD2	Output	SCI2 transmit data output

Note: * Pin names SCK, RXD, and TXD are used in the description for all channels, omitting the channel designation.

15.3 Register Descriptions

The SCI has the following registers for each channel. For details on register addresses and register states during each processing, refer to section 27, List of Registers.

Table 15.2 Register Configuration

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	Serial mode register_0	SCSMR_0	R/W	H'00	H'FFFFC000	8
	Bit rate register_0	SCBRR_0	R/W	H'FF	H'FFFFC002	8
	Serial control register_0	SCSCR_0	R/W	H'00	H'FFFFC004	8
	Transmit data register_0	SCTDR_0	R/W	H'xx	H'FFFFC006	8
	Serial status register_0	SCSSR_0	R/W	H'84	H'FFFFC008	8
	Receive data register_0	SCRDR_0	R	H'xx	H'FFFFC00A	8
	Serial direction control register_0	SCSDCR_0	R/W	H'F2	H'FFFFC00C	8
	Serial port register_0	SCSPTR_0	R/W	H'0x	H'FFFFC00E	8
1	Serial mode register_1	SCSMR_1	R/W	H'00	H'FFFFC080	8
	Bit rate register_1	SCBRR_1	R/W	H'FF	H'FFFFC082	8
	Serial control register_1	SCSCR_1	R/W	H'00	H'FFFFC084	8
	Transmit data register_1	SCTDR_1	R/W	H'xx	H'FFFFC086	8
	Serial status register_1	SCSSR_1	R/W	H'84	H'FFFFC088	8
	Receive data register_1	SCRDR_1	R	H'xx	H'FFFFC08A	8
	Serial direction control register_1	SCSDCR_1	R/W	H'F2	H'FFFFC08C	8
	Serial port register_1	SCSPTR_1	R/W	H'0x	H'FFFFC08E	8
2	Serial mode register_2	SCSMR_2	R/W	H'00	H'FFFFC100	8
	Bit rate register_2	SCBRR_2	R/W	H'FF	H'FFFFC102	8
	Serial control register_2	SCSCR_2	R/W	H'00	H'FFFFC104	8
	Transmit data register_2	SCTDR_2	R/W	H'xx	H'FFFFC106	8
	Serial status register_2	SCSSR_2	R/W	H'84	H'FFFFC108	8
	Receive data register_2	SCRDR_2	R	H'xx	H'FFFFC10A	8
	Serial direction control register_2	SCSDCR_2	R/W	H'F2	H'FFFFC10C	8
	Serial port register_2	SCSPTR_2	R/W	H'0x	H'FFFFC10E	8

15.3.1 Receive Shift Register (SCRSR)

SCRSR receives serial data. The SCI converts serial data input to SCRSR via the RXD pin to parallel data. When one byte has been received, it is automatically transferred to SCRDR. The CPU cannot read or write to SCRSR directly.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

15.3.2 Receive Data Register (SCRDR)

SCRDR is a register that stores serial receive data. After receiving one byte of serial data, the SCI transfers the received data from the receive shift register (SCRSR) into SCRDR for storage and completes operation. After that, SCRSR is ready to receive data.

Since SCRSR and SCRDR work as a double buffer in this way, data can be received continuously.

SCRDR is a read-only register and SCRDR can be read but not written to by the CPU.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

15.3.3 Transmit Shift Register (SCTSR)

SCTSR transmits serial data. The SCI loads transmit data from the transmit data register (SCTDR) into SCTSR, then transmits the data serially from the TXD pin. After transmitting one data byte, the SCI automatically loads the next transmit data from SCTDR into SCTSR and starts transmitting again. If the TDRE flag in the serial status register (SCSSR) is set to 1, the SCI does not transfer data from SCTDR to SCTSR. The CPU cannot read or write to SCTSR directly.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

15.3.4 Transmit Data Register (SCTDR)

SCTDR is an 8-bit register that stores data for serial transmission. When the SCI detects that the transmit shift register (SCTSR) is empty, it moves transmit data written in the SCTDR into SCTSR and starts serial transmission. If the next transmit data has been written to SCTDR during serial transmission from SCTSR, the SCI can transmit data continuously. SCTDR can always be written or read to by the CPU.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

15.3.5 Serial Mode Register (SCSMR)

SCSMR is an 8-bit register that specifies the SCI serial communication format and selects the clock source for the baud rate generator.

The CPU can always read and write to SCSMR.

Bit:	7	6	5	4	3	2	1	0
	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS[1:0]	
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

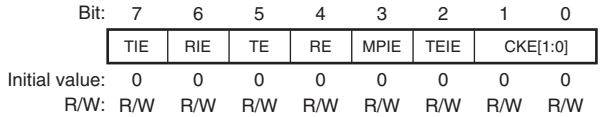
Bit	Bit Name	Initial value	R/W	Description
7	C/$\overline{A}$	0	R/W	Communication Mode Selects whether the SCI operates in asynchronous or clock synchronous mode. 0: Asynchronous mode 1: Clock synchronous mode
6	CHR	0	R/W	Character Length Selects 7-bit or 8-bit data in asynchronous mode. In the clock synchronous mode, the data length is always eight bits, regardless of the CHR setting. When 7-bit data is selected, the MSB (bit 7) of the transmit data register is not transmitted. 0: 8-bit data 1: 7-bit data

Bit	Bit Name	Initial value	R/W	Description
5	PE	0	R/W	Parity Enable Selects whether to add a parity bit to transmit data and to check the parity of receive data, in asynchronous mode. In clock synchronous mode, a parity bit is neither added nor checked, regardless of the PE setting. 0: Parity bit not added or checked 1: Parity bit added and checked* Note: * When PE is set to 1, an even or odd parity bit is added to transmit data, depending on the parity mode (O/$\overline{E}$) setting. Receive data parity is checked according to the even/odd (O/$\overline{E}$) mode setting.
4	O/$\overline{E}$	0	R/W	Parity mode Selects even or odd parity when parity bits are added and checked. The O/$\overline{E}$ setting is used only in asynchronous mode and only when the parity enable bit (PE) is set to 1 to enable parity addition and checking. The O/$\overline{E}$ setting is ignored in clock synchronous mode, or in asynchronous mode when parity addition and checking is disabled. 0: Even parity 1: Odd parity If even parity is selected, the parity bit is added to transmit data to make an even number of 1s in the transmitted character and parity bit combined. Receive data is checked to see if it has an even number of 1s in the received character and parity bit combined. If odd parity is selected, the parity bit is added to transmit data to make an odd number of 1s in the transmitted character and parity bit combined. Receive data is checked to see if it has an odd number of 1s in the received character and parity bit combined.

Bit	Bit Name	Initial value	R/W	Description
3	STOP	0	R/W	Stop Bit Length Selects one or two bits as the stop bit length in asynchronous mode. This setting is used only in asynchronous mode. It is ignored in clock synchronous mode because no stop bits are added. 0: One stop bit*[1] 1: Two stop bits*[2] When receiving, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit, but if the second stop bit is 0, it is treated as the start bit of the next incoming character. Notes: 1. When transmitting, a single 1-bit is added at the end of each transmitted character. 2. When transmitting, two 1 bits are added at the end of each transmitted character.
2	MP	0	R/W	Multiprocessor Mode (only in asynchronous mode) Enables or disables multiprocessor mode. The PE and O/$\overline{E}$ bit settings are ignored in multiprocessor mode. 0: Multiprocessor mode disabled 1: Multiprocessor mode enabled
1, 0	CKS[1:0]	00	R/W	Clock Select 1 and 0 Select the internal clock source of the on-chip baud rate generator. Four clock sources are available. Pϕ, Pϕ/4, Pϕ/16 and Pϕ/64. For further information on the clock source, bit rate register settings, and baud rate, see section 15.3.10, Bit Rate Register (SCBRR). 00: Pϕ 01: Pϕ/4 10: Pϕ/16 11: Pϕ/64 Note: Pϕ: Peripheral clock

15.3.6 Serial Control Register (SCSCR)

SCSCR is an 8-bit register that enables or disables SCI transmission/reception and interrupt requests and selects the transmit/receive clock source. The CPU can always read and write to SCSCR.



Bit:	7	6	5	4	3	2	1	0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable Enables or disables a transmit-data-empty interrupt (TXI) to be issued when the TDRE flag in the serial status register (SCSSR) is set to 1 after serial transmit data is sent from the transmit data register (SCTDR) to the transmit shift register (SCTSR). TXI can be canceled by clearing the TDRE flag to 0 after reading TDRE = 1 or by clearing the TIE bit to 0. 0: Transmit-data-empty interrupt request (TXI) is disabled 1: Transmit-data-empty interrupt request (TXI) is enabled
6	RIE	0	R/W	Receive Interrupt Enable Enables or disables a receive-data-full interrupt (RXI) and a receive error interrupt (ERI) to be issued when the RDRF flag in SCSSR is set to 1 after the serial data received is transferred from the receive shift register (SCRSR) to the receive data register (SCRDR). RXI can be canceled by clearing the RDRF flag after reading RDRF =1. ERI can be canceled by clearing the FER, PER, or ORER flag to 0 after reading 1 from the flag. Both RXI and ERI can also be canceled by clearing the RIE bit to 0. 0: Receive-data-full interrupt (RXI) and receive-error interrupt (ERI) requests are disabled 1: Receive-data-full interrupt (RXI) and receive-error interrupt (ERI) requests are enabled

Bit	Bit Name	Initial value	R/W	Description
5	TE	0	R/W	Transmit Enable Enables or disables the SCI serial transmitter. 0: Transmitter disabled*[1] 1: Transmitter enabled*[2] Notes: 1. The TDRE flag in SCSSR is fixed at 1. 2. Serial transmission starts after writing transmit data into SCTDR and clearing the TDRE flag in SCSSR to 0 while the transmitter is enabled. Select the transmit format in the serial mode register (SCSMR) before setting TE to 1.
4	RE	0	R/W	Receive Enable Enables or disables the SCI serial receiver. 0: Receiver disabled*[1] 1: Receiver enabled*[2] Notes: 1. Clearing RE to 0 does not affect the receive flags (RDRF, FER, PER, and ORER). These flags retain their previous values. 2. Serial reception starts when a start bit is detected in asynchronous mode, or synchronous clock input is detected in clock synchronous mode. Select the receive format in SCSMR before setting RE to 1.
3	MPIE	0	R/W	Multiprocessor Interrupt Enable (only when MP = 1 in SCSMR in asynchronous mode) When this bit is set to 1, receive data in which the multiprocessor bit is 0 is skipped and setting of the RDRF, FER, and ORER status flags in SCSSR is prohibited. On receiving data in which the multiprocessor bit is 1, this bit is automatically cleared to 0 and normal receiving operation is resumed. For details, refer to section 15.4.4, Multiprocessor Communication Function.

Bit	Bit Name	Initial value	R/W	Description
2	TEIE	0	R/W	Transmit End Interrupt Enable Enables or disables a transmit end interrupt (TEI) to be issued when no valid transmit data is found in SCTDR during MSB data transmission. TEI can be canceled by clearing the TEND flag to 0 (by clearing the TDRE flag in SCSSR to 0 after reading TDRE = 1) or by clearing the TEIE bit to 0. 0: Transmit end interrupt request (TEI) is disabled 1: Transmit end interrupt request (TEI) is enabled
1, 0	CKE[1:0]	00	R/W	Clock Enable Select the SCI clock source and enable or disable clock output from the SCK pin. Depending on the combination of CKE[1:0], the SCK pin can be used for serial clock output or serial clock input. When selecting the clock output in clock synchronous mode, set the C/$\overline{A}$ bit in SCSMR to 1 and then set bits CKE[1:0]. For details on clock source selection, refer to table 15.11. • Asynchronous mode 00: Internal clock, SCK pin used for input pin (The input signal is ignored.) The state of the SCK pin depends on the settings of bits SPB1IO and SPB1DT in SCSPTR. 01: Internal clock, SCK pin used for clock output*[1] 10: External clock, SCK pin used for clock input*[2] 11: External clock, SCK pin used for clock input*[2] • Clock synchronous mode 00: Internal clock, SCK pin used for synchronous clock output 01: Internal clock, SCK pin used for synchronous clock output 10: External clock, SCK pin used for synchronous clock input 11: External clock, SCK pin used for synchronous clock input Notes: 1. The output clock frequency is 16 times the bit rate. 2. The input clock frequency is 16 times the bit rate.

15.3.7 Serial Status Register (SCSSR)

SCSSR is an 8-bit register that contains status flags to indicate the SCI operating state.

The CPU can always read and write to SCSSR, but cannot write 1 to status flags TDRE, RDRF, ORER, PER, and FER. These flags can be cleared to 0 only after 1 is read from the flags. The TEND flag is a read-only bit and cannot be modified.

Bit:	7	6	5	4	3	2	1	0
	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
Initial value:	1	0	0	0	0	1	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R/W

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial value	R/W	Description
7	TDRE	1	R/(W)*	Transmit Data Register Empty Indicates whether data has been transferred from the transmit data register (SCTDR) to the transmit shift register (SCTSR) and SCTDR has become ready to be written with next serial transmit data. 0: Indicates that SCTDR holds valid transmit data [Clearing conditions] • When 0 is written to TDRE after reading TDRE = 1 • When the DMAC is activated by a TXI interrupt and transmit data is written to SCTDR • When the DTC is activated by a TXI interrupt and transmit data is transferred to SCTDR while the DISEL bit of MRB in the DTC is 0 (except when the transfer counter value of the DTC is H'0000) 1: Indicates that SCTDR does not hold valid transmit data [Setting conditions] • By a power-on reset or in standby mode • When the TE bit in SCSCR is 0 • When data is transferred from SCTDR to SCTSR and data can be written to SCTDR

Bit	Bit Name	Initial value	R/W	Description
6	RDRF	0	R/(W)*	Receive Data Register Full Indicates that the received data is stored in the receive data register (SCRDR). 0: Indicates that valid received data is not stored in SCRDR [Clearing conditions] • By a power-on reset or in standby mode • When 0 is written to RDRF after reading RDRF = 1 • When the DMAC is activated by an RXI interrupt and data is transferred from SCRDR • When the DTC is activated by an RXI interrupt and the received data is transferred from SCRDR while the DISEL bit of MRB in the DTC is 0 (except when the transfer counter value of the DTC is H'0000) 1: Indicates that valid received data is stored in SCRDR [Setting condition] • When serial reception ends normally and receive data is transferred from SCRSR to SCRDR Note: SCRDR and the RDRF flag are not affected and retain their previous states even if an error is detected during data reception or if the RE bit in the serial control register (SCSCR) is cleared to 0. If reception of the next data is completed while the RDRF flag is still set to 1, an overrun error will occur and the received data will be lost.

Bit	Bit Name	Initial value	R/W	Description
5	ORER	0	R/(W)*	Overrun Error Indicates that an overrun error occurred during reception, causing abnormal termination. 0: Indicates that reception is in progress or was completed successfully*1 [Clearing conditions] • By a power-on reset or in standby mode • When 0 is written to ORER after reading ORER = 1 1: Indicates that an overrun error occurred during reception*2 [Setting condition] • When the next serial reception is completed while RDRF = 1 Notes: 1. The ORER flag is not affected and retains its previous value when the RE bit in SCSCR is cleared to 0. 2. The receive data prior to the overrun error is retained in SCRDR, and the data received subsequently is lost. Subsequent serial reception cannot be continued while the ORER flag is set to 1.

Bit	Bit Name	Initial value	R/W	Description
4	FER	0	R/(W)*	Framing Error Indicates that a framing error occurred during data reception in asynchronous mode, causing abnormal termination. 0: Indicates that reception is in progress or was completed successfully*[1] [Clearing conditions] • By a power-on reset or in standby mode • When 0 is written to FER after reading FER = 1 1: Indicates that a framing error occurred during reception [Setting condition] • When the SCI founds that the stop bit at the end of the received data is 0 after completing reception*[2] Notes: 1. The FER flag is not affected and retains its previous value when the RE bit in SCSCR is cleared to 0. 2. In 2-stop-bit mode, only the first stop bit is checked for a value to 1; the second stop bit is not checked. If a framing error occurs, the receive data is transferred to SCRDR but the RDRF flag is not set. Subsequent serial reception cannot be continued while the FER flag is set to 1.

Bit	Bit Name	Initial value	R/W	Description
3	PER	0	R/(W)*	Parity Error Indicates that a parity error occurred during data reception in asynchronous mode, causing abnormal termination. 0: Indicates that reception is in progress or was completed successfully*[1] [Clearing conditions] • By a power-on reset or in standby mode • When 0 is written to PER after reading PER = 1 1: Indicates that a parity error occurred during reception*[2] [Setting condition] • When the number of 1s in the received data and parity does not match the even or odd parity specified by the O/$\overline{E}$ bit in the serial mode register (SCSMR). Notes: 1. The PER flag is not affected and retains its previous value when the RE bit in SCSCR is cleared to 0. 2. If a parity error occurs, the receive data is transferred to SCRDR but the RDRF flag is not set. Subsequent serial reception cannot be continued while the PER flag is set to 1.

Bit	Bit Name	Initial value	R/W	Description
2	TEND	1	R	Transmit End Indicates that no valid data was in SCTDR during transmission of the last bit of the transmit character and transmission has ended. The TEND flag is read-only and cannot be modified. 0: Indicates that transmission is in progress [Clearing condition] • When 0 is written to TDRE after reading TDRE = 1 1: Indicates that transmission has ended [Setting conditions] • By a power-on reset or in standby mode • When the TE bit in SCSCR is 0 • When TDRE = 1 during transmission of the last bit of a 1-byte serial transmit character Note: The TEND flag value becomes undefined if data is written to SCTDR by activating the DMAC or DTC by a TXI interrupt. In this case, do not use the TEND flag as the transmit end flag.
1	MPB	0	R	Multiprocessor Bit Stores the multiprocessor bit found in the receive data. When the RE bit in SCSCR is cleared to 0, its previous state is retained.
0	MPBT	0	R/W	Multiprocessor Bit Transfer Specifies the multiprocessor bit value to be added to the transmit frame.

Note: * Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

15.3.8 Serial Port Register (SCSPTR)

SCSPTR is an 8-bit register that controls input/output and data for the ports multiplexed with the SCI function pins. It controls break signals during serial transmission and reception when writing of output data to the TXD pin is enabled by the settings of bits SPB0IO and SPB0DT. Also, bits SPB1IO and SPB1DT can be used to write output data to the SCK pin. The EIO bit enables or disables RXI interrupts. The CPU can always read and write to SCSPTR. When reading the value on the SCI pins, use the respective port register. For details, refer to section 22, I/O Ports.

Bit:	7	6	5	4	3	2	1	0
	EIO	-	-	-	SPB1IO	SPB1DT	SPB0IO	SPB0DT
Initial value:	0	0	0	0	0	-	0	1
R/W:	R/W	-	-	-	R/W	W	R/W	W

Bit	Bit Name	Initial value	R/W	Description
7	EIO	0	R/W	Error Interrupt Only Enables or disables RXI interrupts. While the EIO bit is set to 1, the SCI does not request an RXI interrupt to the CPU even if the RIE bit is set to 1. 0: While the RIE bit is 1, RXI and ERI interrupts are sent to the INTC. 1: While the RIE bit is 1, only the ERI interrupt is sent to the INTC.
6 to 4	—	All 0	—	Reserved These bits are always read as 0. The write value should always be 0.
3	SPB1IO	0	R/W	Clock Port Input/Output in Serial Port Controls the SCK pin in combination with the SPB1DT bit, the C/$\overline{A}$ bit in SCSMR, and the CKE[1:0] bits in SCSCR.

Bit	Bit Name	Initial value	R/W	Description
2	SPB1DT	Undefined	W	Clock Port Data in Serial Port Controls the SCK pin in combination with the SPB1IO bit, the C/$\overline{A}$ bit in SCSMR, and the CKE[1:0] bits in SCSCR. Note that the SCK pin function needs to have been selected with the pin function controller (PFC). Also, this bit is write-only. When read its value is undefined.

C/$\overline{A}$	CKE1	CKE0	SPB 1IO	SPB 1DT	SCK pin state
0	0	0	0	×	SCK pin functions as input pin.
0	0	0	1	0	Low-level output
0	0	0	1	1	High-level output
0	0	1	×	×	SCK pin functions as clock output.
0	1	0	×	×	SCK pin functions as clock input.
0	1	1	×	×	SCK pin functions as clock input.
1	0	0	×	×	SCK pin functions as sync clock output.
1	0	1	×	×	SCK pin functions as sync clock output.
1	1	0	×	×	SCK pin functions as sync clock input.
1	1	1	×	×	SCK pin functions as sync clock input.

Note: ×: Don't care

1	SPB0IO	0	R/W	Serial Port Break Output Together with the SPB0DT bit and the TE bit in SCSCR, controls the TXD pin.

Bit	Bit Name	Initial value	R/W	Description
0	SPB0DT	1	W	Serial Port Break Data Together with the SPB0IO bit and TE bit in SCSCR, controls the TXD pin. Note that the TXD pin function needs to have been selected with the pin function controller (PFC). Also, this bit is write-only. When read its value is undefined.

TE bit setting in SCSCR	SPB0IO bit setting	SPB0DT bit setting	State of TXD pin
0	0	*	SPB0DT output disabled (initial state)
0	1	0	Output, low level
0	1	1	Output, high level
1	*	*	Transmit data output

Note: * Don't care

15.3.9 Serial Direction Control Register (SCSDCR)

The DIR bit in the serial direction control register (SCSDCR) selects LSB-first or MSB-first transfer. With an 8-bit data length, LSB-first/MSB-first selection is available regardless of the communication mode.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	DIR	-	-	-
Initial value:	1	1	1	1	0	0	1	0
R/W:	R	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
3	DIR	0	R/W	Data Transfer Direction Selects the serial/parallel conversion format. Valid for an 8-bit transmit/receive format. 0: SCTDR contents are transmitted in LSB-first order Receive data is stored in SCRDR in LSB-first 1: SCTDR contents are transmitted in MSB-first order Receive data is stored in SCRDR in MSB-first
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

15.3.10 Bit Rate Register (SCBRR)

SCBRR is an 8-bit register that, together with the baud rate generator clock source selected by the CKS1 and CKS0 bits in the serial mode register (SCSMR), determines the serial transmit/receive bit rate.

The CPU can always read and write to SCBRR.

The SCBRR setting is calculated as follows:

Bit:	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Asynchronous mode:

$$N = \frac{P\phi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

- Clock synchronous mode:

$$N = \frac{P\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bits/s)
N: SCBRR setting for baud rate generator ($0 \leq N \leq 255$)
(The setting value should satisfy the electrical characteristics.)
$P\phi$: Operating frequency for peripheral modules (MHz)
n: Baud rate generator clock source (n = 0, 1, 2, 3) (for the clock sources and values of n, see table 15.3.)

Table 15.3 SCSMR Settings

		SCSMR Settings	
n	**Clock Source**	**CKS1**	**CKS0**
0	Pϕ	0	0
1	Pϕ/4	0	1
2	Pϕ/16	1	0
3	Pϕ/64	1	1

Note: The bit rate error in asynchronous is given by the following formula:

$$\text{Error (\%)} = \left\{ \frac{P\phi \times 10^6}{(N + 1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

Tables 15.4 shows examples of SCBRR settings in asynchronous mode, and tables 15.5 shows examples of SCBRR settings in clock synchronous mode.

Table 15.4 Bit Rates and SCBRR Settings in Asynchronous Mode (1)

	Pϕ (MHz)																	
Bit Rate (bits/s)	10			12			14			16			18			20		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	177	-0.25	2	212	0.03	2	248	-0.17	3	70	0.03	3	79	-0.12	3	88	-0.25
150	2	129	0.16	2	155	0.16	2	181	0.16	2	207	0.16	2	233	0.16	3	64	0.16
300	2	64	0.16	2	77	0.16	2	90	0.16	2	103	0.16	2	116	0.16	2	129	0.16
600	1	129	0.16	1	155	0.16	1	181	0.16	1	207	0.16	1	233	0.16	2	64	0.16
1200	1	64	0.16	1	77	0.16	1	90	0.16	1	103	0.16	1	116	0.16	1	129	0.16
2400	0	129	0.16	0	155	0.16	0	181	0.16	0	207	0.16	0	233	0.16	1	64	0.16
4800	0	64	0.16	0	77	0.16	0	90	0.16	0	103	0.16	0	116	0.16	0	129	0.16
9600	0	32	-1.36	0	38	0.16	0	45	-0.93	0	51	0.16	0	58	-0.69	0	64	0.16
14400	0	21	-1.36	0	25	0.16	0	29	1.27	0	34	-0.79	0	38	0.16	0	42	0.94
19200	0	15	1.73	0	19	-2.34	0	22	-0.93	0	25	0.16	0	28	1.02	0	32	-1.36
28800	0	10	-1.36	0	12	0.16	0	14	1.27	0	16	2.12	0	19	-2.34	0	21	-1.36
31250	0	9	0.00	0	11	0.00	0	13	0.00	0	15	0.00	0	17	0.00	0	19	0.00
38400	0	7	1.73	0	9	-2.34	0	10	3.57	0	12	0.16	0	14	-2.34	0	15	1.73
115200	0	2	-9.58	0	2	8.51	0	3	-5.06	0	3	8.51	0	4	-2.34	0	4	8.51
500000	0	0*	-37.5	0	0*	-25.0	0	0*	-12.5	0	0*	0.00	0	0*	12.5	0	0*	25.0

Table 15.4 Bit Rates and SCBRR Settings in Asynchronous Mode (2)

Bit Rate (bits/s)	Pϕ (MHz)																	
	22			24			26			28			30			32		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	3	97	-0.35	3	106	-0.44	3	114	0.36	3	123	0.23	3	132	0.13	3	141	0.03
150	3	71	-0.54	3	77	0.16	3	84	-0.43	3	90	0.16	3	97	-0.35	3	103	0.16
300	2	142	0.16	2	155	0.16	2	168	0.16	2	181	0.16	2	194	0.16	2	207	0.16
600	2	71	-0.54	2	77	0.16	2	84	-0.43	2	90	0.16	2	97	-0.35	2	103	0.16
1200	1	142	0.16	1	155	0.16	1	168	0.16	1	181	0.16	1	194	0.16	1	207	0.16
2400	1	71	-0.54	1	77	0.16	1	84	-0.43	1	90	0.16	1	97	-0.35	1	103	0.16
4800	0	142	0.16	0	155	0.16	0	168	0.16	0	181	0.16	0	194	0.16	0	207	0.16
9600	0	71	-0.54	0	77	0.16	0	84	-0.43	0	90	0.16	0	97	-0.35	0	103	0.16
14400	0	47	-0.54	0	51	0.16	0	55	0.76	0	60	-0.39	0	64	0.16	0	68	0.64
19200	0	35	-0.54	0	38	0.16	0	41	0.76	0	45	-0.93	0	48	-0.35	0	51	0.16
28800	0	23	-0.54	0	25	0.16	0	27	0.76	0	29	1.27	0	32	-1.36	0	34	-0.79
31250	0	21	0.00	0	23	0.00	0	25	0.00	0	27	0.00	0	29	0.00	0	31	0.00
38400	0	17	-0.54	0	19	-2.34	0	20	0.76	0	22	-0.93	0	23	1.73	0	25	0.16
115200	0	5	-0.54	0	6	-6.99	0	6	0.76	0	7	-5.06	0	7	1.73	0	8	-3.55
500000	0	0*	37.5	0	1	-25.0	0	1	-18.8	0	1	-12.5	0	1	-6.25	0	1	0.00

Table 15.4 Bit Rates and SCBRR Settings in Asynchronous Mode (3)

Bit Rate (bits/s)	Pϕ (MHz)											
	34			36			38			40		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	3	150	-0.05	3	159	-0.12	3	168	-0.19	3	177	-0.25
150	3	110	-0.29	3	116	0.16	3	123	-0.24	3	129	0.16
300	2	220	0.16	2	233	0.16	2	246	0.16	3	64	0.16
600	2	110	-0.29	2	116	0.16	2	123	-0.24	2	129	0.16
1200	1	220	0.16	1	233	0.16	1	246	0.16	2	64	0.16
2400	1	110	-0.29	1	116	0.16	1	123	-0.24	1	129	0.16
4800	0	220	0.16	0	233	0.16	0	246	0.16	1	64	0.16
9600	0	110	-0.29	0	116	0.16	0	123	-0.24	0	129	0.16
14400	0	73	-0.29	0	77	0.16	0	81	0.57	0	86	-0.22
19200	0	54	0.62	0	58	-0.69	0	61	-0.24	0	64	0.16
28800	0	36	-0.29	0	38	0.16	0	40	0.57	0	42	0.94
31250	0	33	0.00	0	35	0.00	0	37	0.00	0	39	0.00
38400	0	27	-1.18	0	28	1.02	0	30	-0.24	0	32	-1.36
115200	0	8	2.48	0	9	-2.34	0	9	3.08	0	10	-1.36
500000	0	1	6.25	0	1	12.5	0	1	18.8	0	2	-16.7

Table 15.5 Bit Rates and SCBRR Settings in Clock Synchronous Mode (1)

Bit Rate (bits/s)	Pϕ (MHz)											
	10		12		14		16		18		20	
	n	N	n	N	n	N	n	N	n	N	n	N
250	3	155	3	187	3	218	3	249				
500	3	77	3	93	3	108	3	124	3	140	3	155
1000	2	155	2	187	2	218	2	249	3	69	3	77
2500	1	249	2	74	2	87	2	99	2	112	2	124
5000	1	124	1	149	1	174	1	199	1	224	1	249
10000	0	249	1	74	1	87	1	99	1	112	1	124
25000	0	99	0	119	0	139	0	159	0	179	0	199
50000	0	49	0	59	0	69	0	79	0	89	0	99
100000	0	24	0	29	0	34	0	39	0	44	0	49
250000	0	9	0	11	0	13	0	15	0	17	0	19
500000	0	4	0	5	0	6	0	7	0	8	0	9
1000000	—	—	0	2	—	—	0	3	—	—	0	4
2500000	0	0*	—	—	—	—	—	—	—	—	0	1
5000000			—	—	—	—	—	—	—	—	0	0*

Table 15.5 Bit Rates and SCBRR Settings in Clock Synchronous Mode (2)

Bit Rate (bits/s)	Pϕ (MHz)											
	22		24		26		28		30		32	
	n	N	n	N	n	N	n	N	n	N	n	N
250												
500	3	171	3	187	3	202	3	218	3	233	3	249
1000	3	85	3	93	3	101	3	108	3	116	3	124
2500	2	137	2	149	2	162	2	174	2	187	2	199
5000	2	68	2	74	2	80	2	87	2	93	2	99
10000	1	137	1	149	1	162	1	174	1	187	1	199
25000	0	219	0	239	1	64	1	69	1	74	1	79
50000	0	109	0	119	0	129	0	139	0	149	0	159
100000	0	54	0	59	0	64	0	69	0	74	0	79
250000	0	21	0	23	0	25	0	27	0	29	0	31
500000	0	10	0	11	0	12	0	13	0	14	0	15
1000000	—	—	0	5	—	—	0	6	—	—	0	7
2500000	—	—	—	—	—	—	—	—	0	2	—	—
5000000	—	—	—	—	—	—	—	—	—	—	—	—

Table 15.5 Bit Rates and SCBRR Settings in Clock Synchronous Mode (3)

Bit Rate (bits/s)	Pϕ (MHz)							
	34		36		38		40	
	n	N	n	N	n	N	n	N
250								
500								
1000	3	132	3	140	3	147	3	155
2500	2	212	2	224	2	237	2	249
5000	2	105	2	112	2	118	2	124
10000	1	212	1	224	1	237	1	249
25000	1	84	1	89	1	94	1	99
50000	0	169	0	179	0	189	0	199
100000	0	84	0	89	0	94	0	99
250000	0	33	0	35	0	37	0	39
500000	0	16	0	17	0	18	0	19
1000000	—	—	0	8	—	—	0	9
2500000	—	—	—	—	—	—	0	3
5000000	—	—	—	—	—	—	0	1

[Legend]
Blank: No setting possible
—: Setting possible, but error occurs
*: Continuous transmission/reception is disabled.
Note: Settings with an error of 1% or less are recommended.

Table 15.6 indicates the maximum bit rates in asynchronous mode when the baud rate generator is used. Table 15.7 lists the maximum bit rates in clock-synchronous mode when the baud rate generator is used. Tables 15.8 and 15.9 list the maximum rates for external clock input.

Table 15.6 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (Asynchronous Mode)

	Discontinuous Transmission/Reception			Continuous Transmission/Reception		
		Settings			Settings	
Pϕ (MHz)	Maximum Bit Rate (bits/s)	n	N	Maximum Bit Rate (bits/s)	n	N
10	312500	0	0	156250	0	1
12	375000	0	0	187500	0	1
14	437500	0	0	218750	0	1
16	500000	0	0	250000	0	1
18	562500	0	0	281250	0	1
20	625000	0	0	312500	0	1
22	687500	0	0	343750	0	1
24	750000	0	0	375000	0	1
26	812500	0	0	406250	0	1
28	875000	0	0	437500	0	1
30	937500	0	0	468750	0	1
32	1000000	0	0	500000	0	1
34	1062500	0	0	531250	0	1
36	1125000	0	0	562500	0	1
38	1187500	0	0	593750	0	1
40	1250000	0	0	625000	0	1

Table 15.7 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (Clock Synchronous Mode)

	Discontinuous Transmission/Reception			Continuous Transmission/Reception		
		Settings			Settings	
Pϕ (MHz)	Maximum Bit Rate (bits/s)	n	N	Maximum Bit Rate (bits/s)	n	N
10	2500000	0	0	1250000	0	1
12	3000000	0	0	1500000	0	1
14	3500000	0	0	1750000	0	1
16	4000000	0	0	2000000	0	1
18	4500000	0	0	2250000	0	1
20	5000000	0	0	2500000	0	1
22	5500000	0	0	2750000	0	1
24	6000000	0	0	3000000	0	1
26	6500000	0	0	3250000	0	1
28	7000000	0	0	3500000	0	1
30	7500000	0	0	3750000	0	1
32	8000000	0	0	4000000	0	1
34	8500000	0	0	4250000	0	1
36	9000000	0	0	4500000	0	1
38	9500000	0	0	4750000	0	1
40	10000000	0	0	5000000	0	1

Table 15.8 Maximum Bit Rates with External Clock Input (Asynchronous Mode)

Pϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bits/s)
10	2.5	156250
12	3.0	187500
14	3.5	218750
16	4.0	250000
18	4.5	281250
20	5.0	312500
22	5.5	343750
24	6.0	375000
26	6.5	406250
28	7.0	437500
30	7.5	468750
32	8.0	500000
34	8.5	531250
36	9.0	562500
38	9.5	593750
40	10.0	625000

Table 15.9 Maximum Bit Rates with External Clock Input (Clock Synchronous Mode)

Pϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bits/s)
10	1.6667	1666666
12	2.0000	2000000
14	2.3333	2333333
16	2.6667	2666666
18	3.0000	3000000
20	3.3333	3333333
22	3.6667	3666666
24	4.0000	4000000
26	4.3333	4333333
28	4.6667	4666666
30	5.0000	5000000
32	5.3333	5333333
34	5.6667	5666666
36	6.0000	6000000
38	6.3333	6333333
40	6.6667	6666666

15.4 Operation

15.4.1 Overview

For serial communication, the SCI has an asynchronous mode in which characters are synchronized individually, and a clock synchronous mode in which communication is synchronized with clock pulses.

Asynchronous or clock synchronous mode is selected and the transmit format is specified in the serial mode register (SCSMR) as shown in table 15.10. The SCI clock source is selected by the combination of the C/$\overline{A}$ bit in SCSMR and the CKE1 and CKE0 bits in the serial control register (SCSCR) as shown in table 15.11.

(1) Asynchronous Mode

- Data length is selectable: 7 or 8 bits.
- Parity bit is selectable. So is the stop bit length (1 or 2 bits). The combination of the preceding selections constitutes the communication format and character length.
- In receiving, it is possible to detect framing errors, parity errors, overrun errors, and breaks.
- An internal or external clock can be selected as the SCI clock source.
  - When an internal clock is selected, the SCI operates using the clock supplied by the on-chip baud rate generator and can output a clock with a frequency 16 times the bit rate.
  - When an external clock is selected, the external clock input must have a frequency 16 times the bit rate. (The on-chip baud rate generator is not used.)

(2) Clock Synchronous Mode

- The transmission/reception format has a fixed 8-bit data length.
- In receiving, it is possible to detect overrun errors.
- An internal or external clock can be selected as the SCI clock source.
  - When an internal clock is selected, the SCI operates using the on-chip baud rate generator, and outputs a serial clock signal to external devices.
  - When an external clock is selected, the SCI operates on the input serial clock. The on-chip baud rate generator is not used.

Table 15.10 SCSMR Settings and SCI Communication Formats

SCSMR Settings					SCI Communication Format		
Bit 7 C/$\overline{A}$	Bit 6 CHR	Bit 5 PE	Bit 3 STOP	Mode	Data Length	Parity Bit	Stop Bit Length
0	0	0	0	Asynchronous	8-bit	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
	1	0	0		7-bit	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
1	x	x	x	Clock synchronous	8-bit	Not set	None

[Legend]

x: Don't care

Table 15.11 SCSMR and SCSCR Settings and SCI Clock Source Selection

SCSMR	SCSCR Settings				
Bit 7 C/$\overline{A}$	Bit 1 CKE1	Bit 0 CKE0	Mode	Clock Source	SCK Pin Function
0	0	0	Asynchronous	Internal	SCI does not use the SCK pin.
		1			Clock with a frequency 16 times the bit rate is output.
	1	0		External	Input a clock with frequency 16 times the bit rate.
		1			
1	0	0	Clock synchronous	Internal	Serial clock is output.
		1			
	1	0		External	Input the serial clock.
		1			

15.4.2 Operation in Asynchronous Mode

In asynchronous mode, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections of the SCI are independent, so full duplex communication is possible. Both the transmitter and receiver have a double-buffered structure so that data can be read or written during transmission or reception, enabling continuous data transfer.

Figure 15.2 shows the general format of asynchronous serial communication. In asynchronous serial communication, the communication line is normally held in the mark (high) state. The SCI monitors the line and starts serial communication when the line goes to the space (low) state, indicating a start bit. One serial character consists of a start bit (low), data (LSB first when LSB-first transfer is selected), parity bit (high or low), and stop bit (high), in that order.

When receiving in asynchronous mode, the SCI synchronizes at the falling edge of the start bit. The SCI samples each data bit on the eighth pulse of a clock with a frequency 16 times the bit rate. Receive data is latched at the center of each bit.

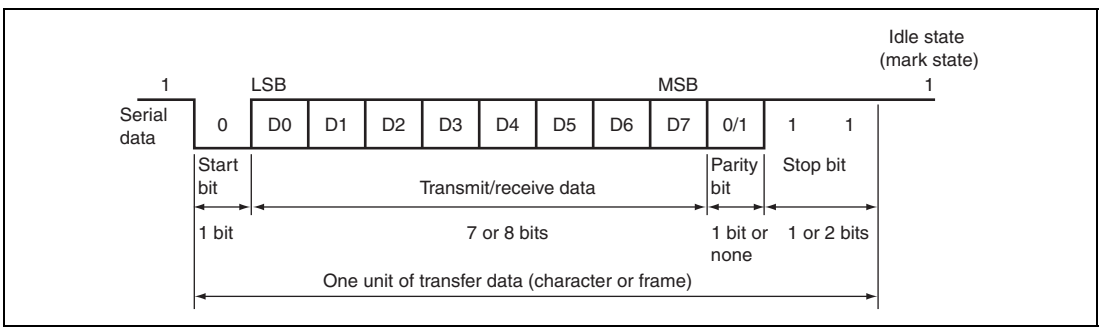


Figure 15.2 Example of Data Format in Asynchronous Communication (8-Bit Data with Parity and Two Stop Bits when LSB-First Transfer is Selected)

(1) Transmit/Receive Formats

Table 15.12 shows the transfer formats that can be selected in asynchronous mode. Any of 12 transfer formats can be selected according to the SCSMR settings.

Table 15.12 Serial Transfer Formats (Asynchronous Mode)

SCSMR Settings				Serial Transfer Format and Frame Length
CHR	PE	MP	STOP	1 2 3 4 5 6 7 8 9 10 11 12
0	0	0	0	S 8-bit data STOP
0	0	0	1	S 8-bit data STOP STOP
0	1	0	0	S 8-bit data P STOP
0	1	0	1	S 8-bit data P STOP STOP
1	0	0	0	S 7-bit data STOP
1	0	0	1	S 7-bit data STOP STOP
1	1	0	0	S 7-bit data P STOP
1	1	0	1	S 7-bit data P STOP STOP
0	x	1	0	S 8-bit data MPB STOP
0	x	1	1	S 8-bit data MPB STOP STOP
1	x	1	0	S 7-bit data MPB STOP
1	x	1	1	S 7-bit data MPB STOP STOP

[Legend]
S: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multiprocessor bit
x: Don't care

(2) Clock

An internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCI transmit/receive clock. The clock source is selected by the C/$\overline{A}$ bit in the serial mode register (SCSMR) and bits CKE1 and CKE0 in the serial control register (SCSCR) (table 15.11).

When an external clock is input at the SCK pin, it must have a frequency equal to 16 times the desired bit rate.

When the SCI operates on an internal clock, it can output a clock signal at the SCK pin. The frequency of this output clock is equal to 16 times the desired bit rate.

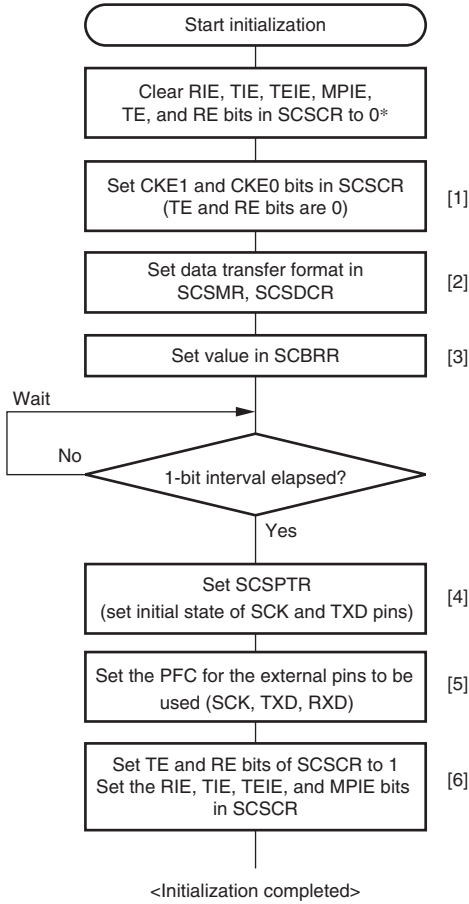
(3) Transmitting and Receiving Data

SCI Initialization (Asynchronous Mode):

Before transmitting or receiving, clear the TE and RE bits to 0 in the serial control register (SCSCR), then initialize the SCI as follows.

When changing the operation mode or the communication format, always clear the TE and RE bits to 0 before following the procedure given below. Clearing the TE bit to 0 sets the TDRE flag to 1 and initializes the transmit shift register (SCTSR). Clearing the RE bit to 0, however, does not initialize the RDRF, PER, FER, and ORER flags or receive data register (SCRDR), which retain their previous contents.

When an external clock is used, the clock should not be stopped during initialization or subsequent operation. SCI operation becomes unreliable if the clock is stopped.



[1] Set the clock selection in SCSCR.

[2] Set the data transfer format in SCSMR and SCSDCR.

[3] Write a value corresponding to the bit rate to SCBRR. Not necessary if an external clock is used.

[4] Set SCSPTR to specify the initial state of the SCK and TXD pins. If it is not necessary to specify an initial state, clear bits SPB1IO and SPB0IO to 0.

[5] Make PFC settings for the external pins to be used. If a clock output setting has been specified, clock output from the SCK pin starts at this point in time.

[6] Set the TE bit or RE bit in SCSCR to 1.* Also make settings of the RIE, TIE, TEIE, and MPIE bits. At this time, the TXD, RXD, and SCK pins are ready to be used. The TXD pin is in a mark state during transmitting, and RXD pin is in an idle state for waiting the start bit during receiving.

Note : * In simultaneous transmit/receive operation, the TE and RE bits must be cleared to 0 or set to 1 simultaneously.

Figure 15.3 Sample Flowchart for SCI Initialization (Asynchronous Mode)

Transmitting Serial Data (Asynchronous Mode):

Figure 15.4 shows a sample flowchart for serial transmission.

Use the following procedure for serial data transmission after enabling the SCI for transmission.

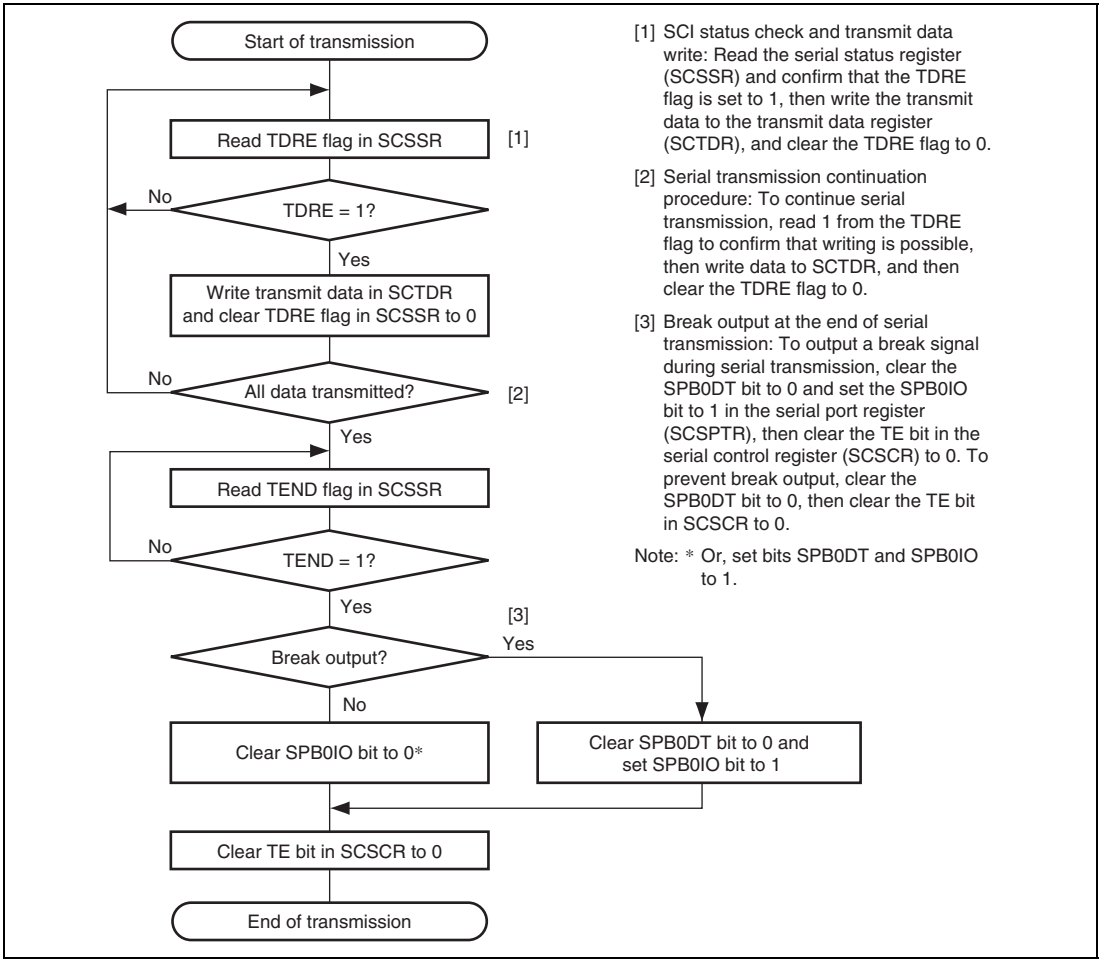


Figure 15.4 Sample Flowchart for Transmitting Serial Data (Asynchronous Mode)

In serial transmission, the SCI operates as described below.

1. The SCI monitors the TDRE flag in the serial status register (SCSSR). If it is cleared to 0, the SCI recognizes that data has been written to the transmit data register (SCTDR) and transfers the data from SCTDR to the transmit shift register (SCTSR).
2. After transferring data from SCTDR to SCTSR, the SCI sets the TDRE flag to 1 and starts transmission. If the TIE bit in the serial control register (SCSCR) is set to 1 at this time, a transmit-data-empty interrupt (TXI) request is generated.

   The serial transmit data is sent from the TXD pin in the following order.

   A. Start bit: One-bit 0 is output.
   B. Transmit data: 8-bit or 7-bit data is output in LSB-first order (when LSB-first transfer is selected).
   C. Parity bit or multiprocessor bit: One parity bit (even or odd parity) or one multiprocessor bit is output. (A format in which neither parity nor multiprocessor bit is output can also be selected.)
   D. Stop bit(s): One or two 1 bits (stop bits) are output.
   E. Mark state: 1 is output continuously until the start bit that starts the next transmission is sent.
3. The SCI checks the TDRE flag at the timing for sending the stop bit.
   If the TDRE flag is 0, the data is transferred from SCTDR to SCTSR, the stop bit is sent, and then serial transmission of the next frame is started.
   If the TDRE flag is 1, the TEND flag in SCSSR is set to 1, the stop bit is sent, and then the "mark state" is entered in which 1 is output. If the TEIE bit in SCSCR is set to 1 at this time, a TEI interrupt request is generated.

Figure 15.5 shows an example of the operation for transmission.

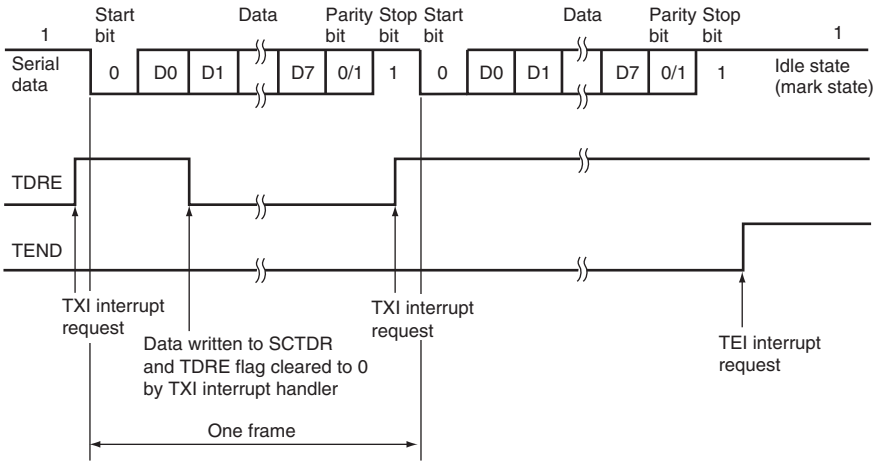


Figure 15.5 Example of Transmission in Asynchronous Mode (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

Receiving Serial Data (Asynchronous Mode):

Figure 15.6 shows a sample flowchart for serial reception.

Use the following procedure for serial data reception after enabling the SCI for reception.

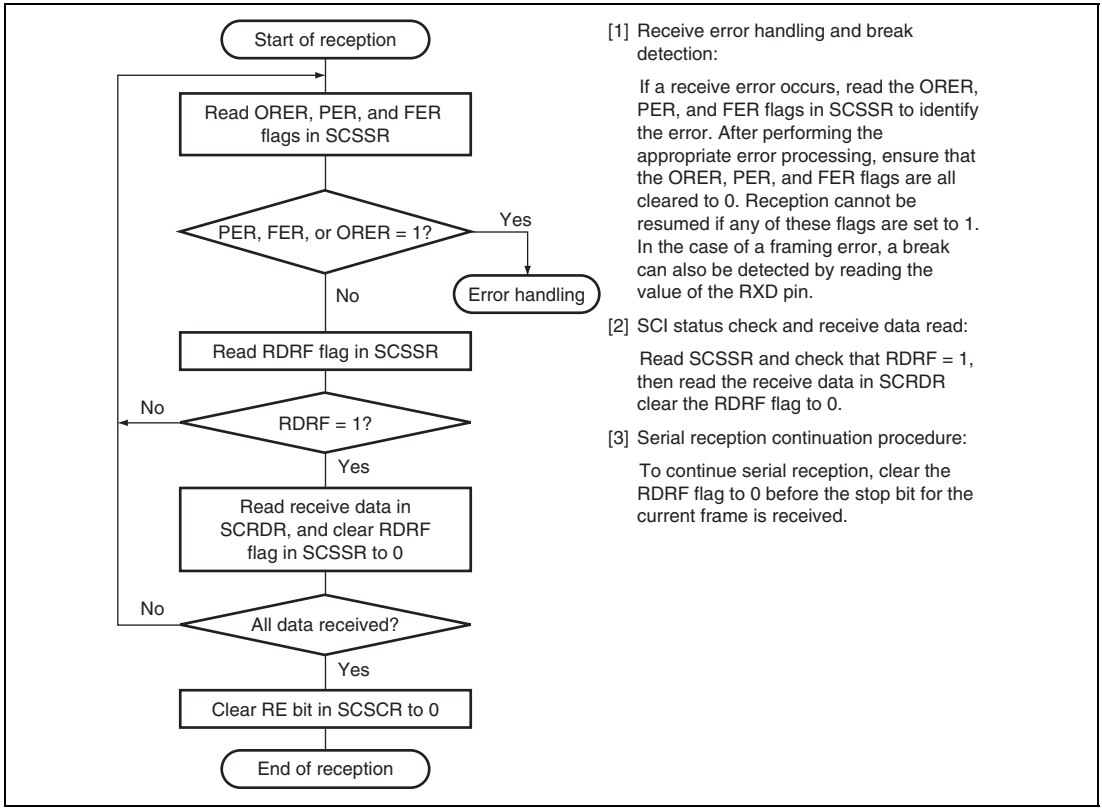


Figure 15.6 Sample Flowchart for Receiving Serial Data (Asynchronous Mode) (1)

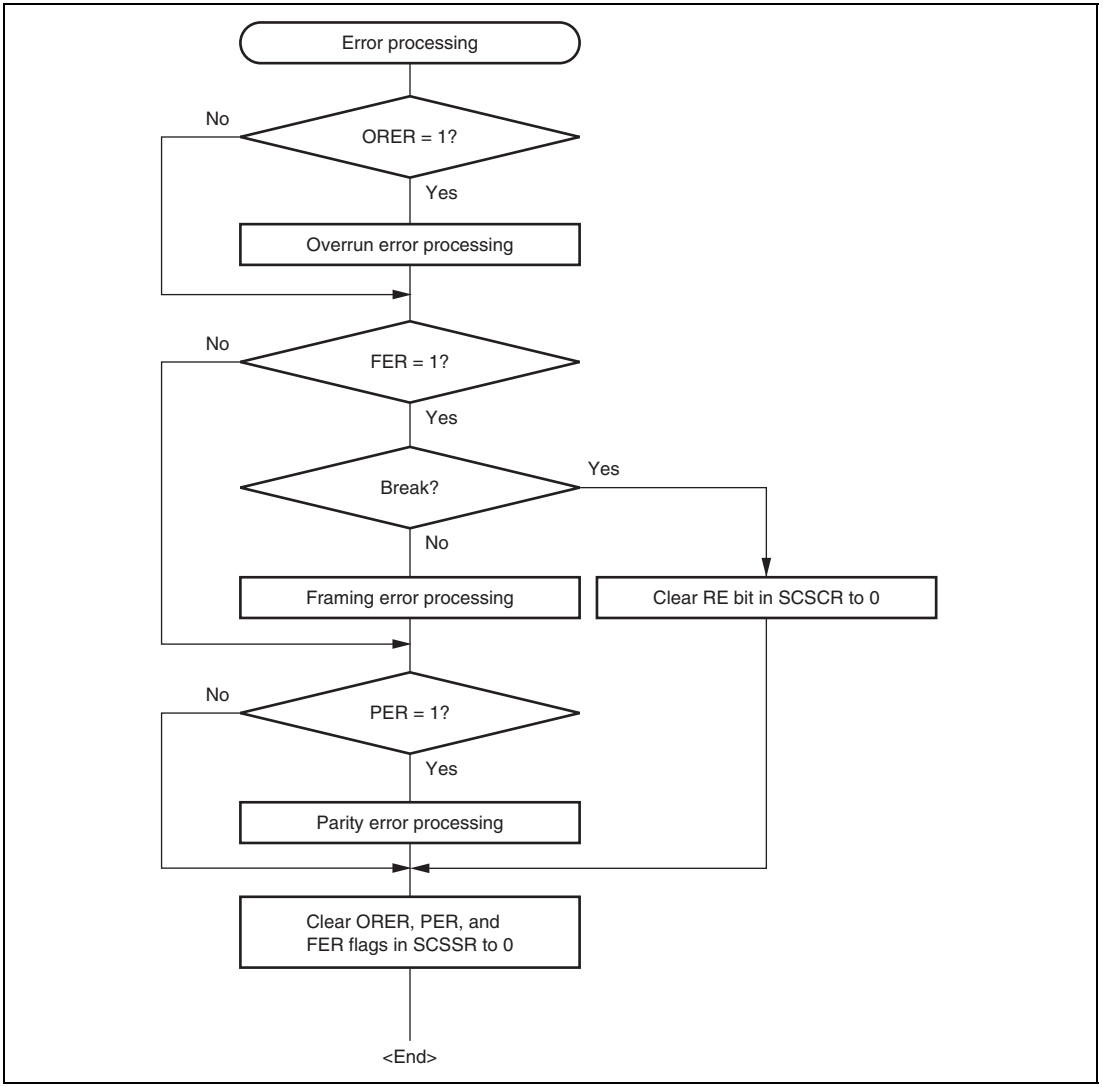


Figure 15.6 Sample Flowchart for Receiving Serial Data (Asynchronous Mode) (2)

In serial reception, the SCI operates as described below.

1. The SCI monitors the transmission line, and if a 0 start bit is detected, performs internal synchronization and starts reception.
2. The received data is stored in SCRSR in LSB-to-MSB order.
3. The parity bit and stop bit are received.
   After receiving these bits, the SCI carries out the following checks.
   A. Parity check: The SCI counts the number of 1s in the received data and checks whether the count matches the even or odd parity specified by the O/$\overline{E}$ bit in the serial mode register (SCSMR).
   B. Stop bit check: The SCI checks whether the stop bit is 1. If there are two stop bits, only the first is checked.
   C. Status check: The SCI checks whether the RDRF flag is 0 and the received data can be transferred from the receive shift register (SCRSR) to SCRDR.

   If all the above checks are passed, the RDRF flag is set to 1 and the received data is stored in SCRDR. If a receive error is detected, the SCI operates as shown in table 15.13

   Note: When a receive error occurs, subsequent reception cannot be continued. In addition, the RDRF flag will not be set to 1 after reception; be sure to clear the error flag to 0.

4. If the EIO bit in SCSPTR is cleared to 0 and the RIE bit in SCSCR is set to 1 when the RDRF flag changes to 1, a receive-data-full interrupt (RXI) request is generated. If the RIE bit in SCSCR is set to 1 when the ORER, PER, or FER flag changes to 1, a receive error interrupt (ERI) request is generated.

Table 15.13 Receive Errors and Error Conditions

Receive Error	Abbreviation	Error Condition	Data Transfer
Overrun error	ORER	When the next data reception is completed while the RDRF flag in SCSSR is set to 1	The received data is not transferred from SCRSR to SCRDR.
Framing error	FER	When the stop bit is 0	The received data is transferred from SCRSR to SCRDR.
Parity error	PER	When the received data does not match the even or odd parity specified in SCSMR	The received data is transferred from SCRSR to SCRDR.

Figure 15.7 shows an example of the operation for reception.

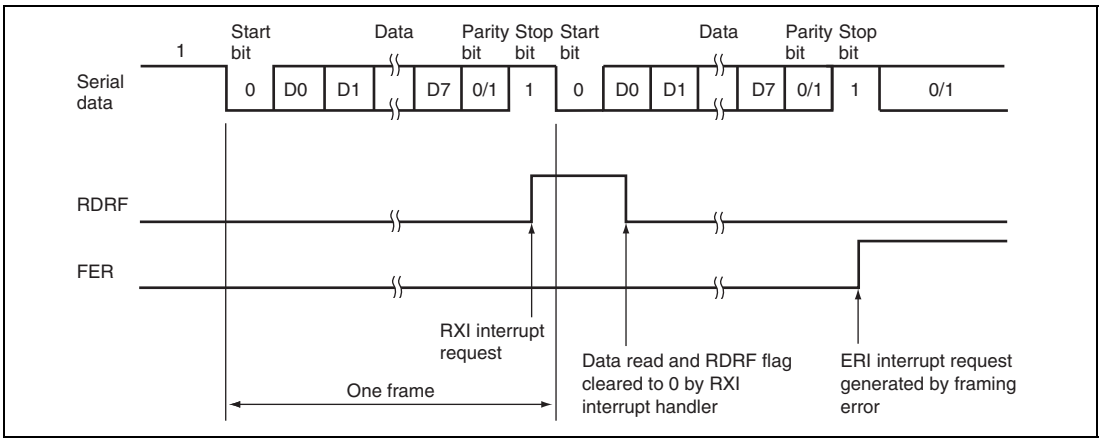


Figure 15.7 Example of SCI Receive Operation (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

15.4.3 Clock Synchronous Mode

In clock synchronous mode, the SCIF transmits and receives data in synchronization with clock pulses. This mode is suitable for high-speed serial communication.

The SCI transmitter and receiver are independent, so full-duplex communication is possible while sharing the same clock. Both the transmitter and receiver have a double-buffered structure so that data can be read or written during transmission or reception, enabling continuous data transfer.

Figure 15.8 shows the general format in clock synchronous serial communication.

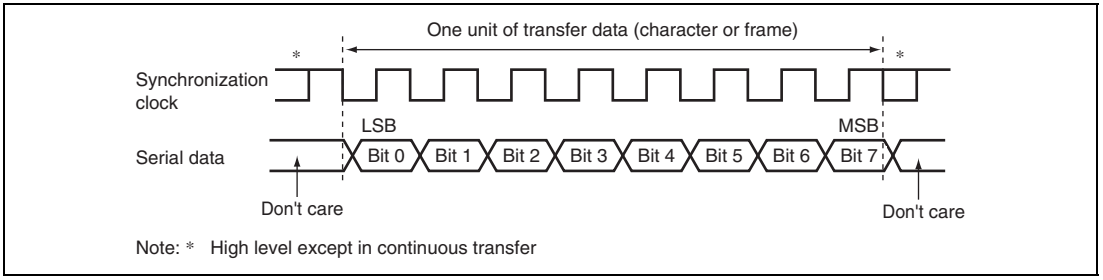


Figure 15.8 Data Format in Clock Synchronous Communication (when LSB-First Transfer is Selected)

In clock synchronous serial communication, each data bit is output on the communication line from one falling edge of the serial clock to the next. Data is guaranteed valid at the rising edge of the serial clock. In each character, the serial data bits are transmitted in order from the LSB (first) to the MSB (last). After output of the MSB, the communication line remains in the state of the MSB (when LSB-first transfer is selected). In clock synchronous mode, the SCI transmits or receives data by synchronizing with the rising edge of the serial clock.

(1) Communication Format

The data length is fixed at eight bits. No parity bit can be added.

(2) Clock

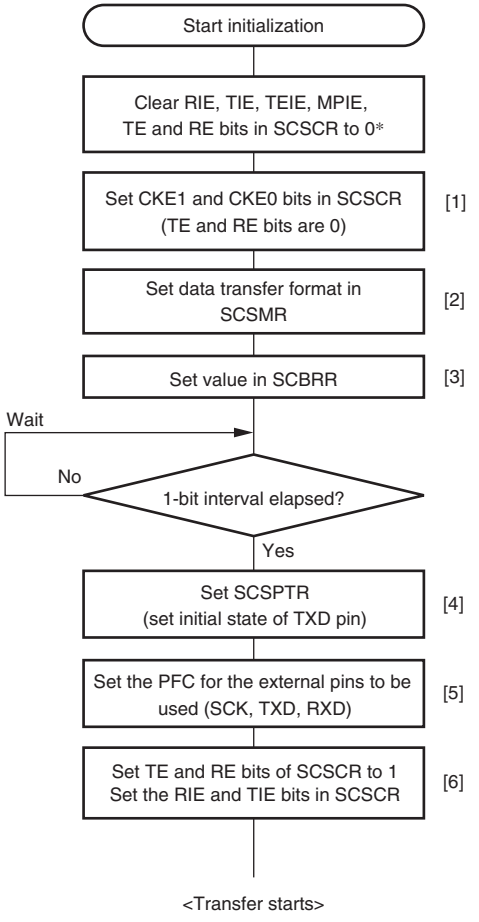
An internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCI transmit/receive clock. For selection of the SCI clock source, see table 15.11.

When the SCI operates on an internal clock, it outputs the clock signal at the SCK pin. Eight clock pulses are output per transmitted or received character. When the SCI does not perform transmission or reception, the clock signal remains in the high state. When only reception is performed, output of the synchronous clock continues until an overrun error occurs or the RE bit is cleared to 0. For the reception of n characters, select the external clock as the clock source. If the internal clock has to be used, set RE and TE to 1, then transmit n characters of dummy data at the same time as receiving the n characters of data.

(3) Transmitting and Receiving Data

SCI Initialization (Clock Synchronous Mode): Before transmitting, receiving, or changing the mode or communication format, the software must clear the TE and RE bits to 0 in the serial control register (SCSCR), then initialize the SCI. Clearing TE to 0 sets the TDRE flag to 1 and initializes the transmit shift register (SCTSR). Clearing RE to 0, however, does not initialize the RDRF, PER, FER, and ORER flags and receive data register (SCRDR), which retain their previous contents.

Figure 15.9 shows a sample flowchart for initializing the SCI.



[1] Set the clock selection in SCSCR.

[2] Set the data transfer format in SCSMR.

[3] Write a value corresponding to the bit rate to SCBRR. Not necessary if an external clock is used.

[4] Set SCSPTR to specify the initial state of the TXD pin. If it is not necessary to specify an initial state, clear the SPB0IO bit to 0.

[5] Make PFC settings for the external pins to be used.

[6] Set the TE bit or RE bit in SCR to 1.* Also make settings of the RIE, TIE, TEIE, and MPIE bits. At this time, the TXD, RXD, and SCK pins are ready to be used. The TXD pin is in a mark state during transmitting. Reception (but not transmission) is enabled in clock-synchronous mode and, if a sync clock output (clock master) setting has been specified, clock output from the SCK pin starts at this point in time.

Note: * In simultaneous transmit and receive operations, the TE and RE bits should both be cleared to 0 or set to 1 simultaneously.

Figure 15.9 Sample Flowchart for SCI Initialization (Clock Synchronous Mode)

Transmitting Serial Data (Clock Synchronous Mode): Figure 15.10 shows a sample flowchart for transmitting serial data.

Use the following procedure for serial data transmission after enabling the SCI for transmission.

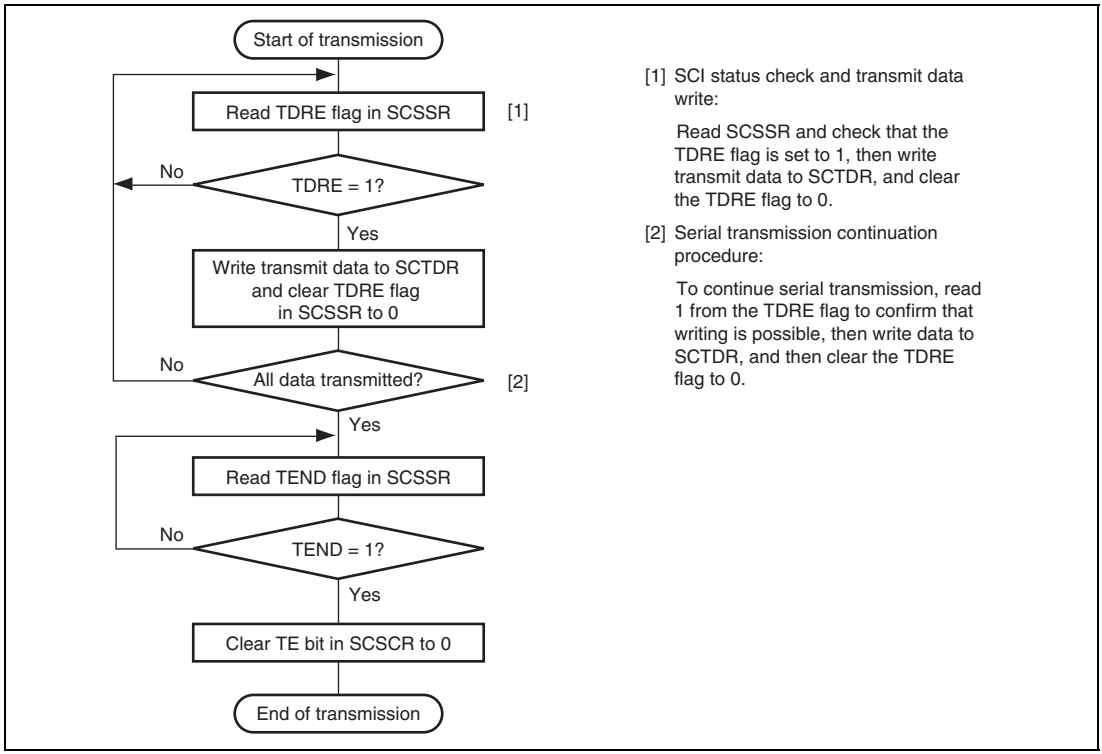


Figure 15.10 Sample Flowchart for Transmitting Serial Data (Clock Synchronous Mode)

In transmitting serial data, the SCI operates as follows:

1. The SCI monitors the TDRE flag in the serial status register (SCSSR). If it is cleared to 0, the SCI recognizes that data has been written to the transmit data register (SCTDR) and transfers the data from SCTDR to the transmit shift register (SCTSR).
2. After transferring data from SCTDR to SCTSR, the SCI sets the TDRE flag to 1 and starts transmission. If the transmit-data-empty interrupt enable bit (TIE) in the serial control register (SCSCR) is set to 1 at this time, a transmit-data-empty interrupt (TXI) request is generated.

   If clock output mode is selected, the SCI outputs eight synchronous clock pulses. If an external clock source is selected, the SCI outputs data in synchronization with the input clock. Data is output from the TXD pin in order from the LSB (bit 0) to the MSB (bit 7) (when LSB-first transfer is selected).
3. The SCI checks the TDRE flag at the timing for sending the last bit. If the TDRE flag is 0, the data is transferred from SCTDR to SCTSR and serial transmission of the next frame is started. If the TDRE flag is 1, the TEND flag in SCSSR is set to 1, the last bit is sent, and then the TXD pin holds the states.

   If the TEIE bit in SCSCR is set to 1 at this time, a TEI interrupt request is generated.
4. After the end of serial transmission, the SCK pin is held in the high state.

Figure 15.11 shows an example of SCI transmit operation.

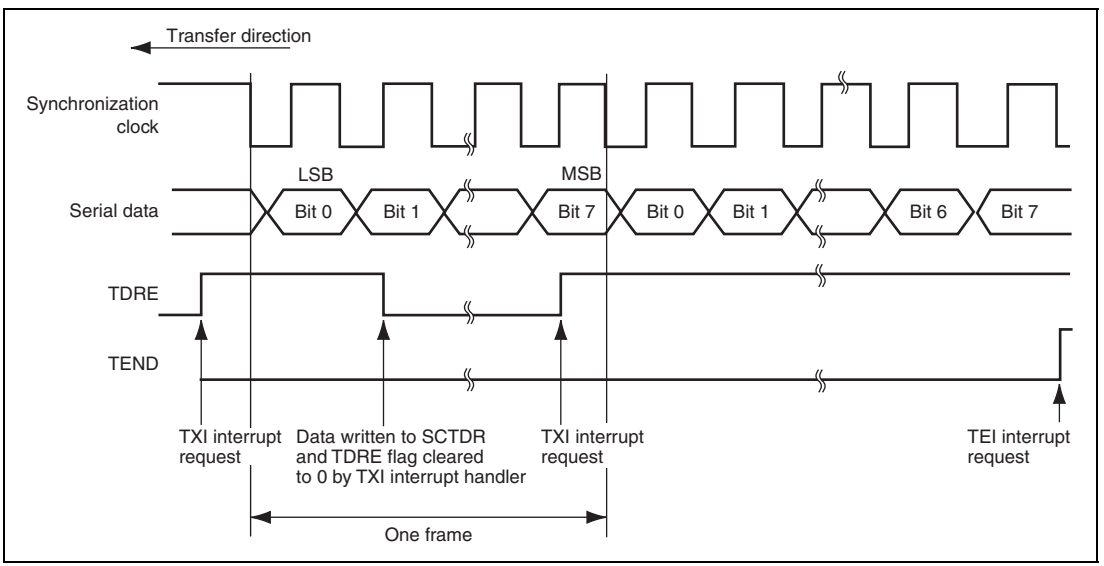


Figure 15.11 Example of SCI Transmit Operation (when LSB-First Transfer is Selected)

Receiving Serial Data (Clock Synchronous Mode): Figure 15.12 shows a sample flowchart for receiving serial data. Use the following procedure for serial data reception after enabling the SCIF for reception.

When switching from asynchronous mode to clock synchronous mode, make sure that the ORER, PER, and FER flags are all cleared to 0. If the FER or PER flag is set to 1, the RDRF flag will not be set and data reception cannot be started.

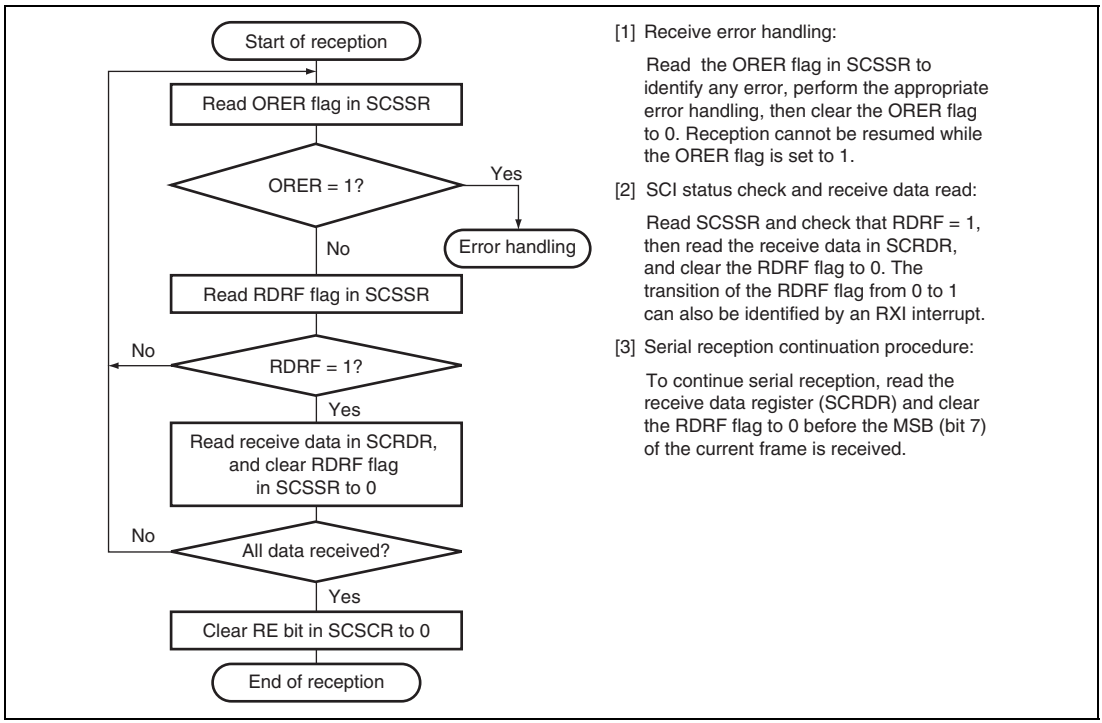


Figure 15.12 Sample Flowchart for Receiving Serial Data (Clock Synchronous Mode) (1)

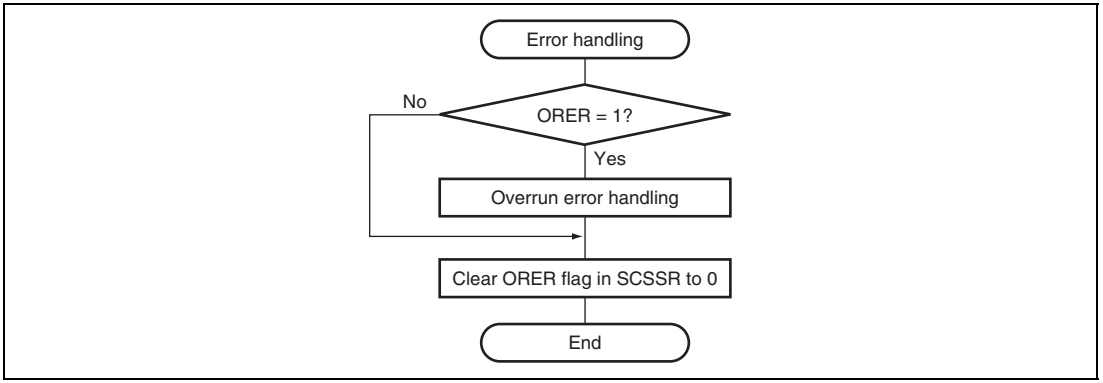


Figure 15.12 Sample Flowchart for Receiving Serial Data (Clock Synchronous Mode) (2)

In receiving, the SCI operates as follows:

1. The SCI synchronizes with serial clock input or output and initializes internally.
2. Receive data is shifted into SCRSR in order from the LSB to the MSB (when LSB-first transfer is selected). After receiving the data, the SCI checks whether the RDRF flag is 0 and the receive data can be transferred from SCRSR to SCRDR. If this check is passed, the SCI sets the RDRF flag to 1 and stores the received data in SCRDR. If a receive error is detected, the SCI operates as shown in table 15.13. In this state, subsequent reception cannot be continued. In addition, the RDRF flag will not be set to 1 after reception; be sure to clear the RDRF flag to 0.
3. After setting RDRF to 1, if the receive-data-full interrupt enable bit (RIE) is set to 1 in SCSCR, the SCI requests a receive-data-full interrupt (RXI). If the ORER bit is set to 1 and the RIE bit in SCSCR is also set to 1, the SCI requests a receive error interrupt (ERI).

Figure 15.13 shows an example of SCI receive operation.

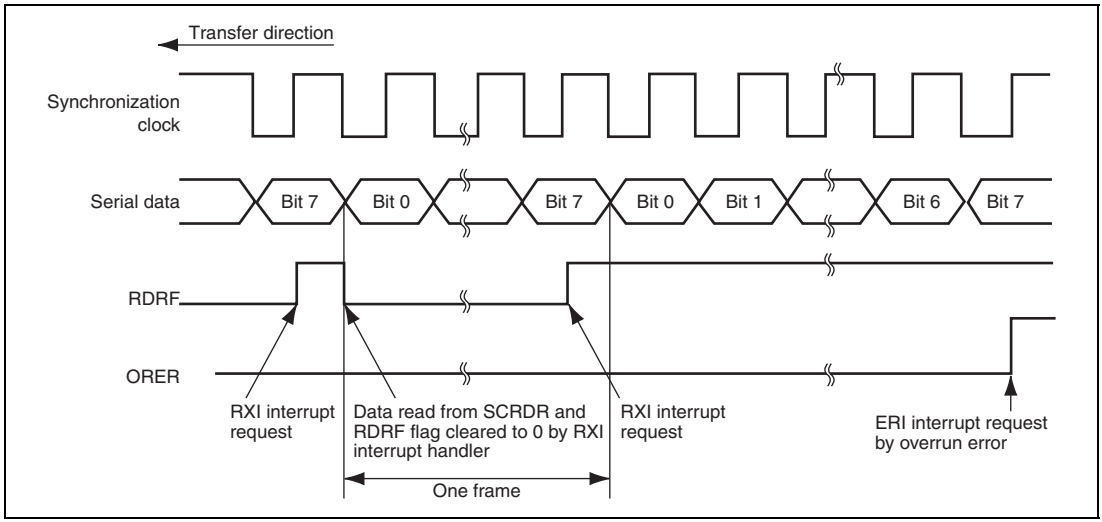
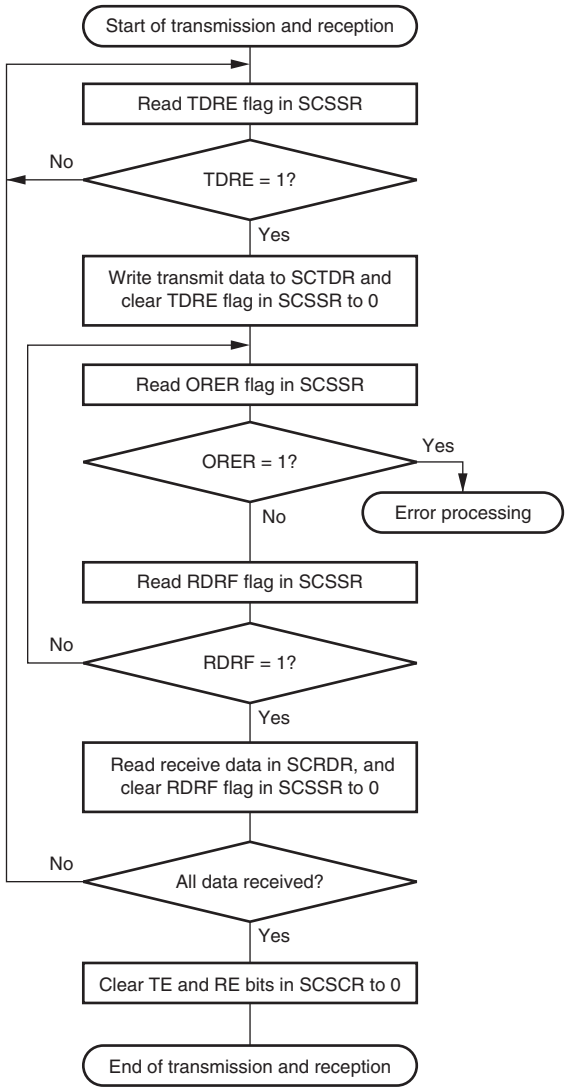


Figure 15.13 Example of SCI Receive Operation (when LSB-First Transfer is Selected)

Transmitting and Receiving Serial Data Simultaneously (Clock Synchronous Mode): Figure 15.14 shows a sample flowchart for transmitting and receiving serial data simultaneously.

Use the following procedure for serial data transmission and reception after enabling the SCI for transmission and reception.



[1] SCI status check and transmit data write:
Read SCSSR and check that the TDRE flag is set to 1, then write transmit data to SCTDR and clear the TDRE flag to 0.
Transition of the TDRE flag from 0 to 1 can also be identified by a TXI interrupt.

[2] Receive error processing:
If a receive error occurs, read the ORER flag in SCSSR, and after performing the appropriate error processing, clear the ORER flag to 0. Reception cannot be resumed if the ORER flag is set to 1.

[3] SCI status check and receive data read:
Read SCSSR and check that the RDRF flag is set to 1, then read the receive data in SCRDR and clear the RDRF flag to 0. Transition of the RDRF flag from 0 to 1 can also be identified by an RXI interrupt.

[4] Serial transmission/reception continuation procedure:
To continue serial transmission/reception, before the last bit of the current frame is received, finish reading the RDRF flag, reading SCRDR, and clearing the RDRF flag to 0. Also, before the last bit of the current frame is transmitted, read 1 from the TDRE flag to confirm that writing is possible. Then write data to SCTDR and clear the TDRE flag to 0.

Note: When switching from transmit or receive operation to simultaneous transmit and receive operations, first clear the TE bit and RE bit to 0, then set both these bits to 1 simultaneously.

Figure 15.14 Sample Flowchart for Transmitting/Receiving Serial Data (Clock Synchronous Mode)

15.4.4 Multiprocessor Communication Function

Use of the multiprocessor communication function enables data transfer to be performed among a number of processors sharing communication lines by means of asynchronous serial communication using the multiprocessor format, in which a multiprocessor bit is added to the transfer data. When multiprocessor communication is carried out, each receiving station is addressed by a unique ID code. The serial communication cycle consists of two component cycles: an ID transmission cycle which specifies the receiving station, and a data transmission cycle. The multiprocessor bit is used to differentiate between the ID transmission cycle and the data transmission cycle. If the multiprocessor bit is 1, the cycle is an ID transmission cycle, and if the multiprocessor bit is 0, the cycle is a data transmission cycle. Figure 15.15 shows an example of inter-processor communication using the multiprocessor format. The transmitting station first sends the ID code of the receiving station with which it wants to perform serial communication as data with a 1 multiprocessor bit added. It then sends transmit data as data with a 0 multiprocessor bit added. The receiving station skips data until data with a 1 multiprocessor bit is sent. When data with a 1 multiprocessor bit is received, the receiving station compares that data with its own ID. The station whose ID matches then receives the data sent next. Stations whose ID does not match continue to skip data until data with a 1 multiprocessor bit is again received.

The SCI uses the MPIE bit in SCSCR to implement this function. When the MPIE bit is set to 1, transfer of receive data from SCRSR to SCRDR, error flag detection, and setting the SCSSR status flags, RDRF, FER, and OER to 1 are inhibited until data with a 1 multiprocessor bit is received. On reception of receive character with a 1 multiprocessor bit, the MPBR bit in SCSSR is set to 1 and the MPIE bit is automatically cleared, thus normal reception is resumed. If the RIE bit in SCSCR is set to 1 at this time, an RXI interrupt is generated.

When the multiprocessor format is selected, the parity bit setting is invalid. All other bit settings are the same as those in normal asynchronous mode. The clock used for multiprocessor communication is the same as that in normal asynchronous mode.

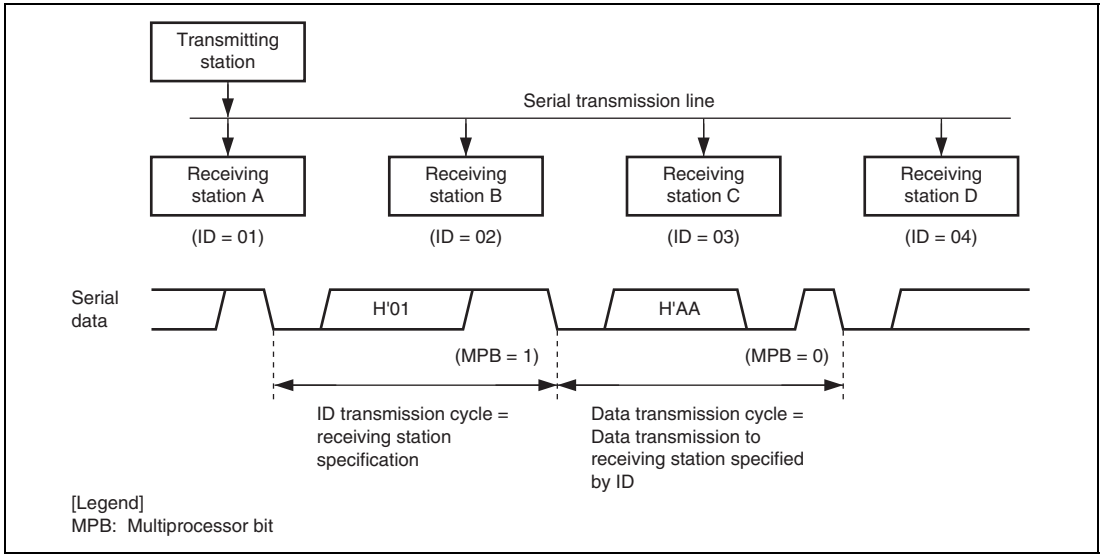
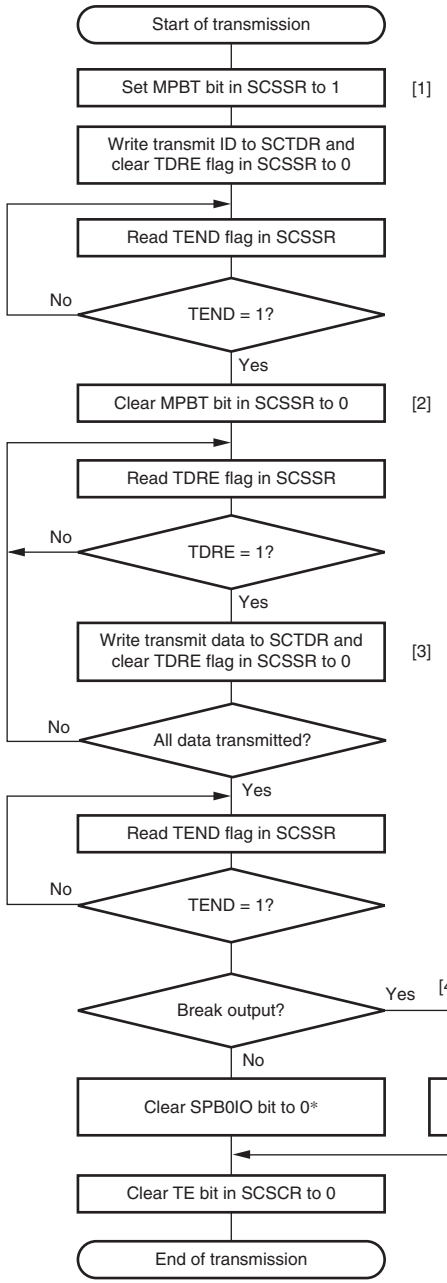


Figure 15.15 Example of Communication Using Multiprocessor Format (Transmission of Data H'AA to Receiving Station A)

15.4.5 Multiprocessor Serial Data Transmission

Figure 15.16 shows a sample flowchart for multiprocessor serial data transmission. For an ID transmission cycle, set the MPBT bit in SCSSR to 1 before transmission, and maintain the MPBT value at 1 until the ID transmission actually completes. For a data transmission cycle, clear the MPBT bit in SCSSR to 0 before transmission. All other SCI operations are the same as those in asynchronous mode.



[1] ID transmission cycle: Set the MPBT bit in the serial status register (SCSSR) to 1. Write the ID to the transmit data register (SCTDR), and clear the TDRE flag to 0. Wait until ID transmission completes.

[2] Data transmission cycle: Clear the MPBT bit in SCSSR to 0. Write the transmit data to SCTDR, and clear the TDRE flag to 0.

[3] Data transmission cycle continuation procedure: To continue the data transmission cycle, read 1 from the TDRE flag to confirm that writing is possible, then write the transmit data to SCTDR, and then clear the TDRE flag to 0.

[4] Break output at the end of the data transmission cycle: To output a break signal during serial transmission, clear the SPB0DT bit to 0 and set the SPB0IO bit to 1 in the serial port register (SCSPTR), then clear the TE bit in the serial control register (SCSCR) to 0. To prevent break output, clear the SPB0IO bit to 0, then clear the TE bit in SCSCR to 0.

Note: * Or, set bits SPB0DT and SPB0IO to 1.

Figure 15.16 Sample Multiprocessor Serial Transmission Flowchart

15.4.6 Multiprocessor Serial Data Reception

Figure 15.18 shows a sample flowchart for multiprocessor serial data reception. If the MPIE bit in SCSCR is set to 1, data is skipped until data with a 1 multiprocessor bit is sent. On receiving data with a 1 multiprocessor bit, the receive data is transferred to SCRDR. An RXI interrupt request is generated at this time. All other SCI operations are the same as in asynchronous mode. Figure 15.17 shows an example of SCI operation for multiprocessor format reception.

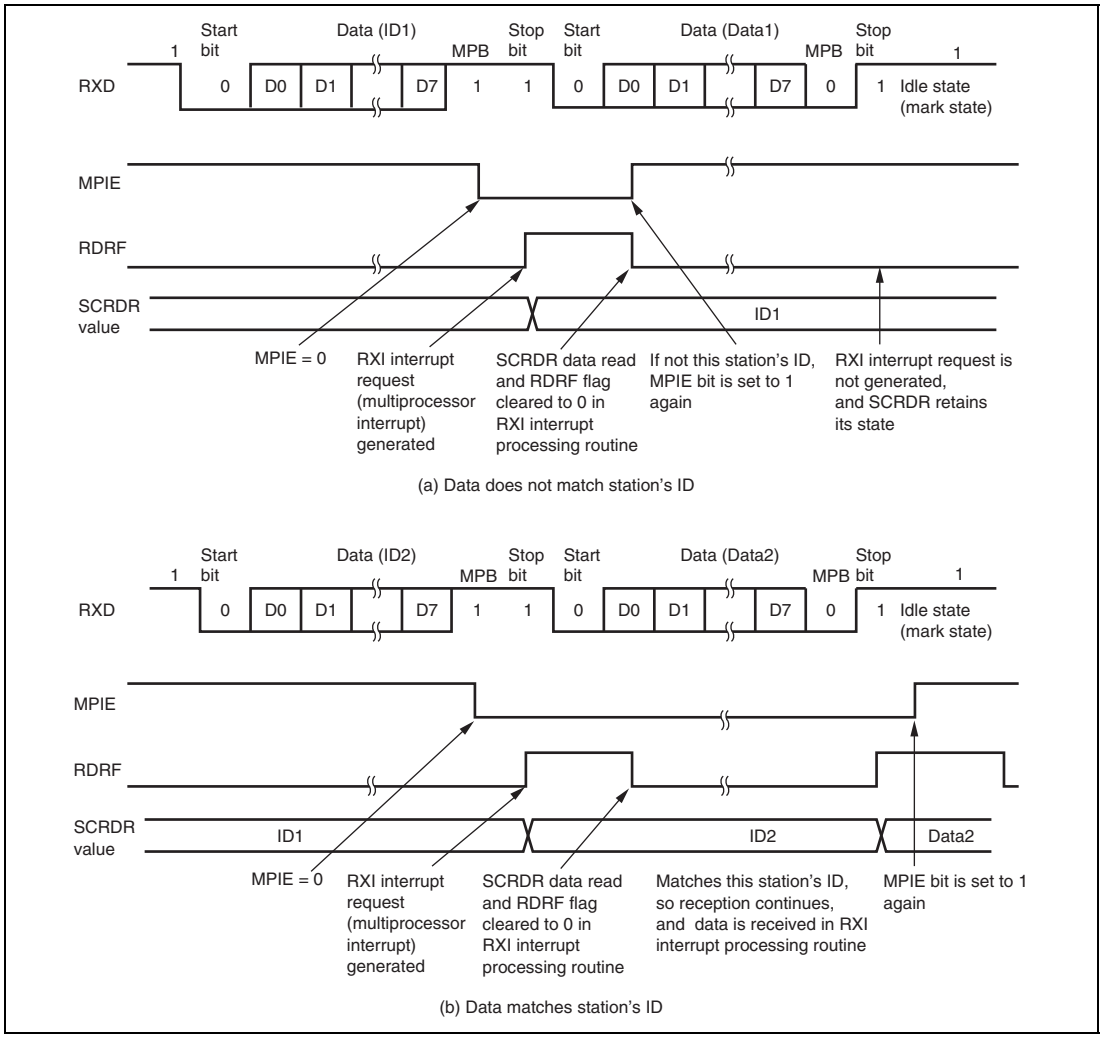


Figure 15.17 Example of SCI Operation in Reception (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit, LSB-first)

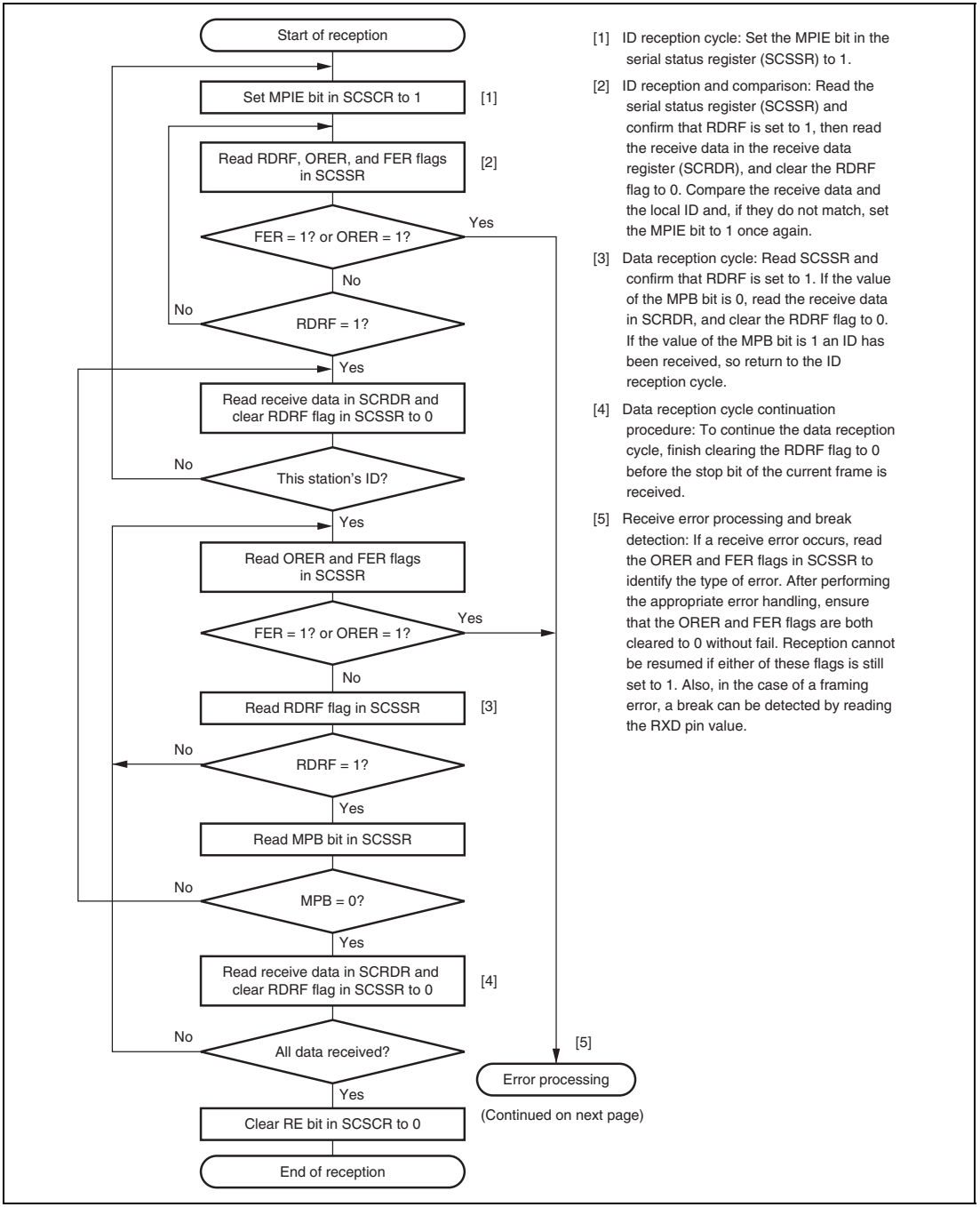


Figure 15.18 Sample Multiprocessor Serial Reception Flowchart (1)

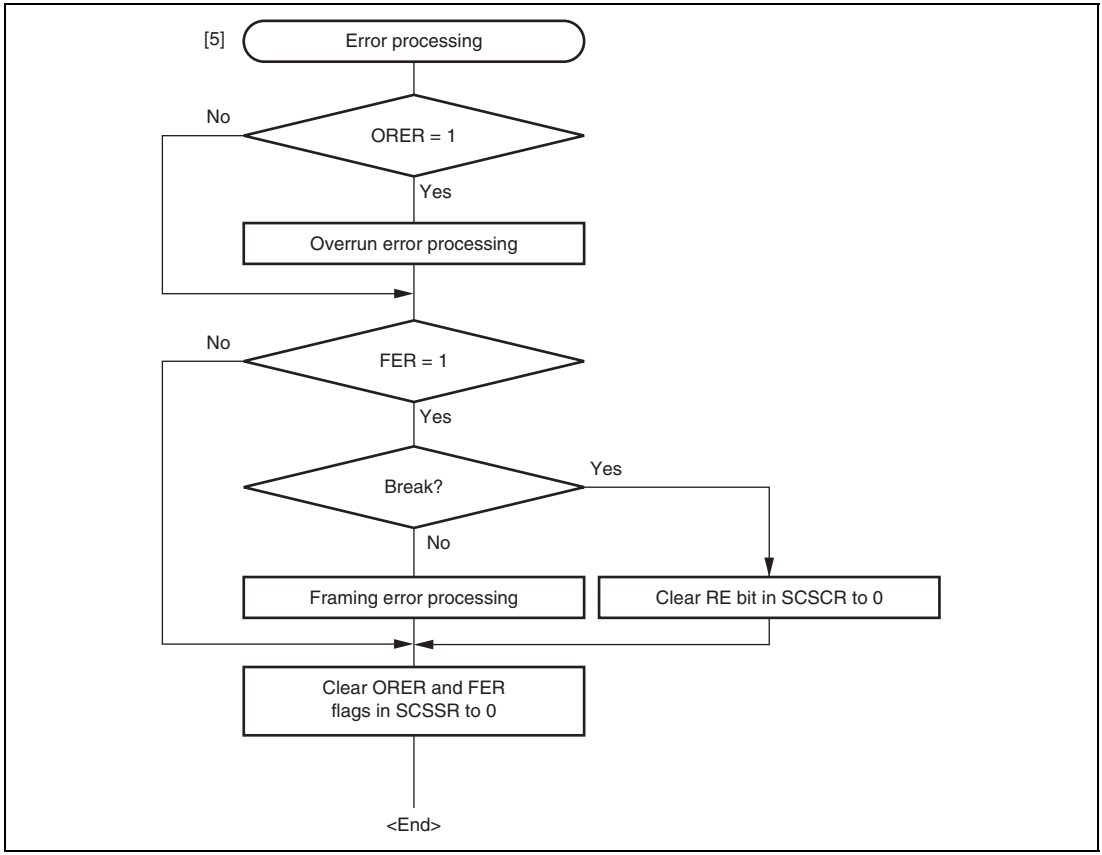


Figure 15.18 Sample Multiprocessor Serial Reception Flowchart (2)

15.5 Interrupt Sources and DMAC/DTC

The SCI has four interrupt sources: transmit end (TEI), receive error (ERI), receive-data-full (RXI), and transmit-data-empty (TXI) interrupt requests.

Table 15.14 shows the interrupt sources and priority. The interrupt sources are enabled or disabled by means of the TIE, RIE, and TEIE bits in SCSCR and the EIO bit in SCSPTR. A separate interrupt request is sent to the interrupt controller for each of these interrupt sources.

When the TDRE flag in the serial status register (SCSSR) is set to 1, a TXI interrupt request is generated. The TXI interrupt request can be used to activate the direct memory access controller (DMAC) or the data transfer controller (DTC) to transfer data. When the DMAC is activated to transfer data, the TDRE flag is cleared to 0 automatically when data is written to the transmit data register (SCTDR), and no TXI interrupt request to the CPU is generated. When the DTC is activated to transfer data, and if the value of DTC's DISEL bit is 0 and the transfer counter value is other than 0, the TDRE flag is cleared to 0 automatically when data is written to SCTDR, and no TXI interrupt request to the CPU is generated. However, if the value of the DISEL bit is 0 and the transfer counter value is 0, or if the value of the DISEL bit is 1, the TDRE flag is not cleared to 0 automatically when data is written to SCTDR, and a TXI interrupt request to the CPU is generated after the data write to SCTDR.

When the RDRF flag in SCSSR is set to 1, an RXI interrupt request is generated. The RXI interrupt request can be used to activate the DMAC or DTC to transfer data. When the DMAC is activated to transfer data, the RDRF flag is cleared to 0 automatically when data is read from the receive data register (SCRDR), and no RXI interrupt request to the CPU is generated. When the DTC is activated to transfer data, and if the value of DTC's DISEL bit is 0 and the transfer counter value is other than 0, the RDRF flag is cleared to 0 automatically when data is read from the receive data register (SCRDR), and no RXI interrupt request to the CPU is generated. However, if the value of the DISEL bit is 0 and the transfer counter value is 0, or if the value of the DISEL bit is 1, the RDRF flag is not cleared to 0 automatically when data is read from SCRDR, and an RXI interrupt request to the CPU is generated after the data write to SCRDR.

When the ORER, FER, or PER flag in SCSSR is set to 1, an ERI interrupt request is generated. This request cannot be used to activate the DMAC or DTC. It is possible to disable generation of RXI interrupt requests and allow only ERI interrupt requests to be generated during data reception processing. To accomplish this, set the RIE bit to 1 and the EIO bit in SCSPTR to 1. Note that setting the EIO bit to 1 will prevent the DMAC or DTC from transferring received data because no RXI interrupt requests are generated.

When the TEND flag in SCSSR is set to 1, a TEI interrupt request is generated. This request cannot be used to activate the DMAC or DTC.

The TXI interrupt indicates that transmit data can be written, and the TEI interrupt indicates that transmission has been completed.

Table 15.14 SCI Interrupt Sources

Interrupt Source	Description	Interrupt Enable Bit	DMAC/DTC Activation	Priority
ERI	Interrupt caused by receive error (ORER, FER, or PER)	RIE = 1	Not possible	High
RXI	Interrupt caused by receive data full (RDRF)	RIE = 1 and EIO = 0	Possible	
TXI	Interrupt caused by transmit data empty (TDRE)	TIE = 1	Possible	
TEI	Interrupt caused by transmit end (TEND)	TEIE = 1	Not possible	Low

15.6 Serial Port Register (SCSPTR) and SCI Pins

The relationship between SCSPTR and the SCI pins is shown in figures 15.19 and 15.20.

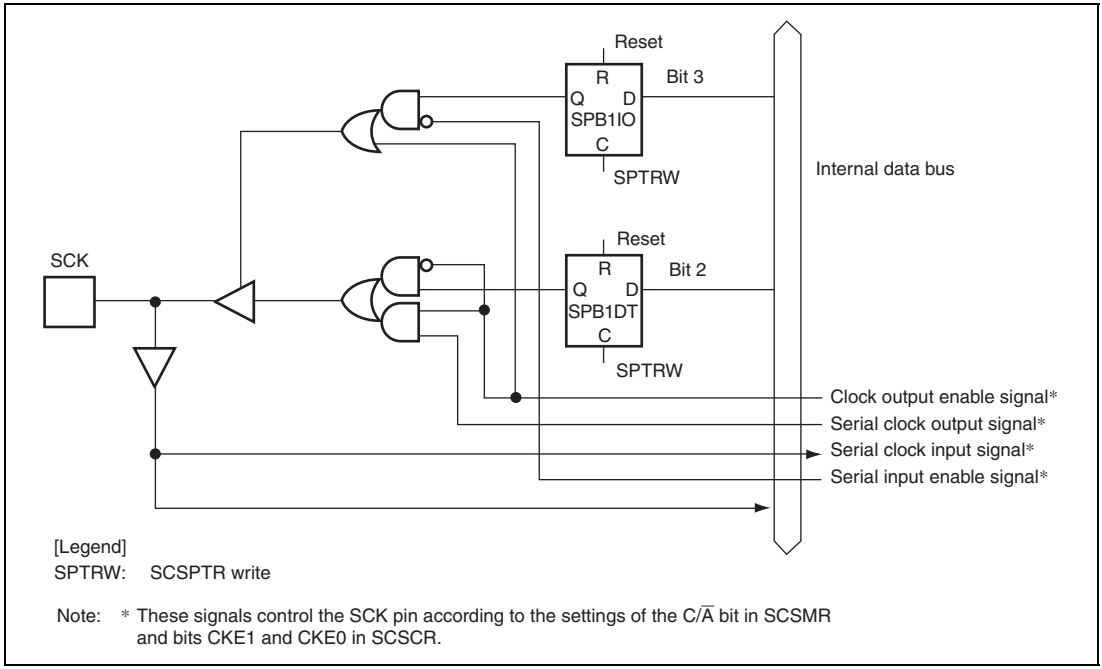


Figure 15.19 SPB1IO Bit, SPB1DT bit, and SCK Pin

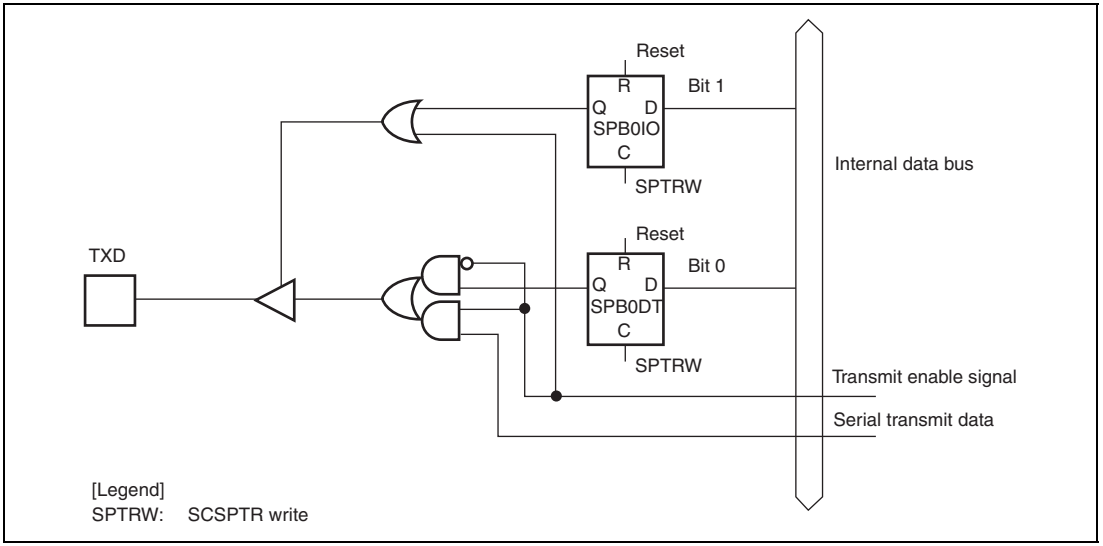


Figure 15.20 SPB0IO Bit, SPB0DT bit, and TXD Pin

15.7 Usage Notes

15.7.1 SCTDR Writing and TDRE Flag

The TDRE flag in the serial status register (SCSSR) is a status flag indicating transferring of transmit data from SCTDR into SCTSR. The SCI sets the TDRE flag to 1 when it transfers data from SCTDR to SCTSR.

Data can be written to SCTDR regardless of the TDRE bit status.

If new data is written in SCTDR when TDRE is 0, however, the old data stored in SCTDR will be lost because the data has not yet been transferred to SCTSR. Before writing transmit data to SCTDR, be sure to check that the TDRE flag is set to 1.

15.7.2 Multiple Receive Error Occurrence

If multiple receive errors occur at the same time, the status flags in SCSSR are set as shown in table 15.15. When an overrun error occurs, data is not transferred from the receive shift register (SCRSR) to the receive data register (SCRDR) and the received data will be lost.

Table 15.15 SCSSR Status Flag Values and Transfer of Received Data

	SCSSR Status Flags				Receive Data Transfer from SCRSR to SCRDR
Receive Errors Generated	**RDRF**	**ORER**	**FER**	**PER**	
Overrun error	1	1	0	0	Not transferred
Framing error	0	0	1	0	Transferred
Parity error	0	0	0	1	Transferred
Overrun error + framing error	1	1	1	0	Not transferred
Overrun error + parity error	1	1	0	1	Not transferred
Framing error + parity error	0	0	1	1	Transferred
Overrun error + framing error + parity error	1	1	1	1	Not transferred

15.7.3 Break Detection and Processing

Break signals can be detected by reading the RXD pin directly when a framing error (FER) is detected. In the break state the input from the RXD pin consists of all 0s, so the FER flag is set and the parity error flag (PER) may also be set.

After a break is received, the SCI halts receive operation. At this time not only transfer of receive data from SCRSR to SCRDR stops; setting in SCRSR of serial data input on the RXD pin stops as well.

To restart receive operation, input a high-level signal on the RXD pin, and clear the overrun error (ORER), FER, and PER flags.

15.7.4 Sending a Break Signal

The I/O condition and level of the TXD pin are determined by the SPB0IO and SPB0DT bits in the serial port register (SCSPTR). This feature can be used to send a break signal.

Until TE bit is set to 1 (enabling transmission) after initializing, TXD pin does not work. During the period, mark status is performed by SPB0DT bit. Therefore, the SPB0IO and SPB0DT bits should be set to 1 (high level output).

To send a break signal during serial transmission, set the SPB0IO bit to 1 and clear the SPB0DT bit to 0 (low level), then clear the TE bit to 0 (halting transmission). When the TE bit is cleared to 0, the transmitter is initialized regardless of the current transmission state, and the low level is output from the TXD pin.

15.7.5 Receive Data Sampling Timing and Receive Margin (Asynchronous Mode)

The SCI operates on a base clock with a frequency of 16 times the transfer rate in asynchronous mode. In reception, the SCI synchronizes internally with the fall of the start bit, which it samples on the base clock. Receive data is latched at the rising edge of the eighth base clock pulse. The timing is shown in figure 15.21.

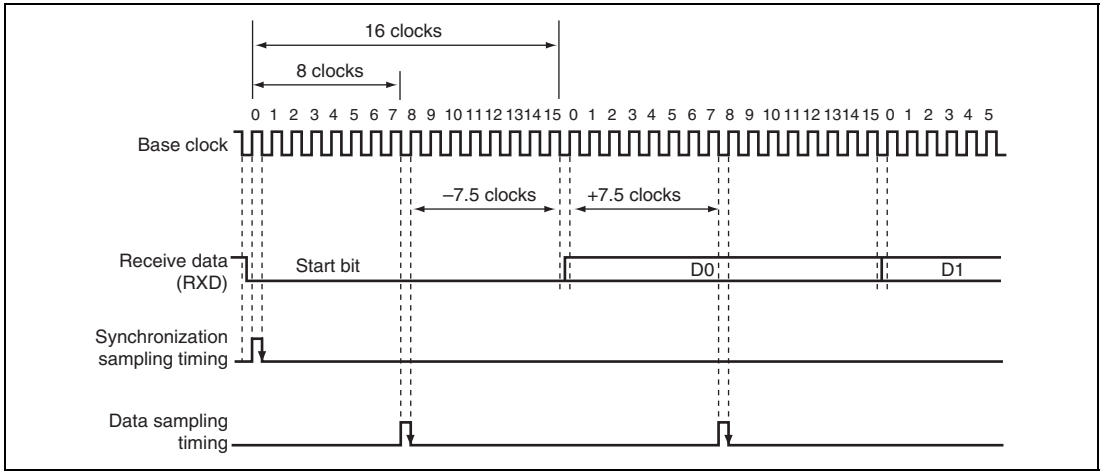


Figure 15.21 Receive Data Sampling Timing in Asynchronous Mode

The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

Equation 1:

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)\,F - \frac{|D - 0.5|}{N}(1+F) \right| \times 100\ \%$$

Where: M: Receive margin (%)
N: Ratio of bit rate to clock (N = 16)
D: Clock duty (D = 0 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute deviation of clock frequency

From equation 1, if F = 0 and D = 0.5, the receive margin is 46.875%, as given by equation 2.

Equation 2:

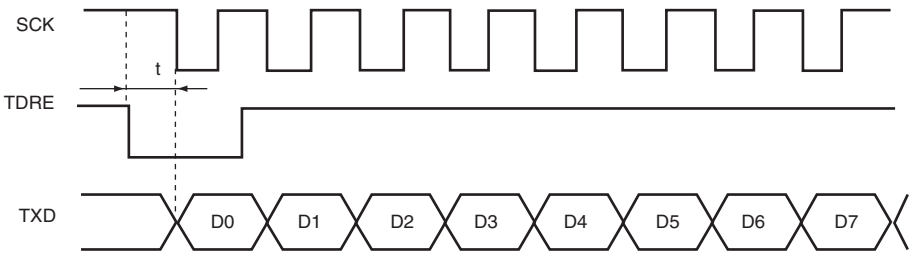
When D = 0.5 and F = 0:

$$M = (0.5 - 1/(2 \times 16)) \times 100\%$$
$$= 46.875\%$$

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

15.7.6 Note on Using DMAC or DTC

When the external clock source is used for the clock for synchronization, input the external clock after waiting for five or more cycles of the peripheral operating clock after SCTDR is modified through the DMAC or DTC. If a transmit clock is input within four cycles after SCTDR is modified, a malfunction may occur (figure 15.22).



Note: When using the external clock, t must be set to larger than 4 cycles.

Figure 15.22 Example of Clock Synchronous Transfer Using DMAC or DTC

When data is written to SCTDR by activating the DMAC or DTC by a TXI interrupt, the TEND flag value becomes undefined. In this case, do not use the TEND flag as the transmit end flag.

15.7.7 Note on Using External Clock in Clock Synchronous Mode

TE and RE must be set to 1 after waiting for four or more cycles of the peripheral operating clock after the SCK external clock is changed from 0 to 1.

TE and RE must be set to 1 only while the SCK external clock is 1.

15.7.8 Module Standby Mode Setting

SCI operation can be disabled or enabled using the standby control register. The initial setting is for SCI operation to be halted. Register access is enabled by clearing module standby mode. For details, refer to section 26, Power-Down Modes.

Section 16 Serial Communication Interface with FIFO (SCIF)

This LSI has a channel serial communication interface with FIFO (SCIF) that supports both asynchronous and clock synchronous serial communication. It also has 16-stage FIFO registers for both transmission and reception independently that enable this LSI to perform efficient high-speed continuous communication.

16.1 Features

- Asynchronous serial communication:
  - Serial data communication is performed by start-stop in character units. The SCIF can communicate with a universal asynchronous receiver/transmitter (UART), an asynchronous communication interface adapter (ACIA), or any other communications chip that employs a standard asynchronous serial system. There are eight selectable serial data communication formats.
  - Data length: 7 or 8 bits
  - Stop bit length: 1 or 2 bits
  - Parity: Even, odd, or none
  - Receive error detection: Parity, framing, and overrun errors
  - Break detection: Break is detected when a framing error is followed by at least one frame at the space 0 level (low level). It is also detected by reading the RXD level directly from the serial port data register when a framing error occurs.
- Clock Synchronous mode:
  - Serial data communication is synchronized with a clock signal. The SCIF can communicate with other chips having a clock synchronous communication function. There is one serial data communication format.
  - Data length: 8 bits
  - Receive error detection: Overrun errors
- Full duplex communication: The transmitting and receiving sections are independent, so the SCIF can transmit and receive simultaneously. Both sections use 16-stage FIFO buffering, so high-speed continuous data transfer is possible in both the transmit and receive directions.
- On-chip baud rate generator with selectable bit rates
- Internal or external transmit/receive clock source: From either baud rate generator (internal) or SCK pin (external)

- Four types of interrupts: Transmit-FIFO-data-empty, break, receive-FIFO-data-full, and receive-error interrupts are requested independently.
- The transmit FIFO data empty and receive FIFO data full requests can activate the data transfer controller (DTC) for data transfer.
- When the SCIF is not in use, it can be stopped by halting the clock supplied to it, saving power.
- In asynchronous, on-chip modem control functions ($\overline{\text{RTS}}$ and $\overline{\text{CTS}}$).
- The number of data in the transmit and receive FIFO registers and the number of receive errors of the receive data in the receive FIFO register can be ascertained.
- A time-out error (DR) can be detected when receiving in asynchronous mode.

Figure 16.1 shows a block diagram of the SCIF.

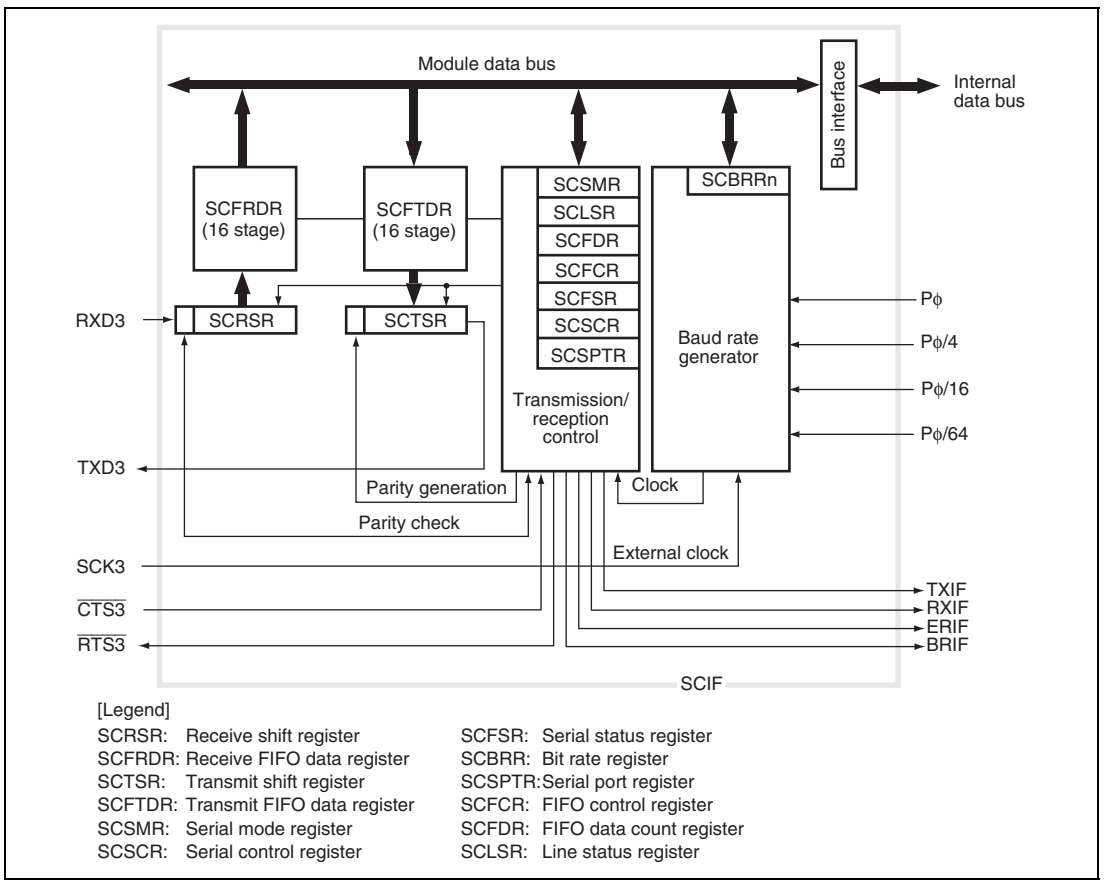


Figure 16.1 Block Diagram of SCIF

16.2 Input/Output Pins

The SCIF has the serial pins summarized in table 16.1.

Table 16.1 Pin Configuration

Channel	Pin Name	Abbreviation	I/O	Function
3	Serial clock pin	SCK3	I/O	Clock I/O
	Receive data pin	RXD3	Input	Receive data input
	Transmit data pin	TXD3	Output	Transmit data output
	Request to send pin	$\overline{\text{RTS3}}$	Output	Request to send
	Clear to send pin	$\overline{\text{CTS3}}$	Input	Clear to send

Note: In the following descriptions in this section, the channel number in the abbreviated pin names is omitted as in SCK, RXD, TXD, $\overline{\text{RTS}}$, and $\overline{\text{CTS}}$.

16.3 Register Descriptions

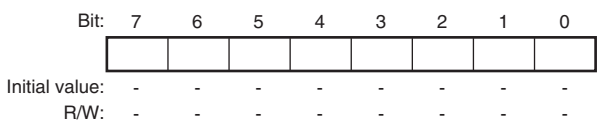
The SCIF has the following registers. These registers specify the data format and bit rate, and control the transmitter and receiver sections.

Table 16.2 Register Configuration

Register Name	Abbr.	R/W	Initial Value	Address	Access Size
Serial mode register_3	SCSMR_3	R/W	H'0000	H'FFFFC180	16
Bit rate register_3	SCBRR_3	R/W	H'FF	H'FFFFC182	8
Serial control register_3	SCSCR_3	R/W	H'0000	H'FFFFC184	16
Transmit FIFO data register_3	SCFTDR_3	W	H'xx	H'FFFFC186	8
Serial status register_3	SCFSR_3	R/W	H'0060	H'FFFFC188	16
Receive FIFO data register_3	SCFRDR_3	R	H'xx	H'FFFFC18A	8
FIFO control register_3	SCFCR_3	R/W	H'0000	H'FFFFC18C	16
FIFO data count register_3	SCFDR_3	R	H'0000	H'FFFFC18E	16
Serial port register_3	SCSPTR_3	R/W	H'00xx	H'FFFFC190	16
Line status register_3	SCLSR_3	R/W	H'0000	H'FFFFC192	16

16.3.1 Receive Shift Register (SCRSR)

SCRSR receives serial data. Data input at the RXD pin is loaded into SCRSR in the order received, LSB (bit 0) first, converting the data to parallel form. When one byte has been received, it is automatically transferred to SCFRDR, the receive FIFO data register. The CPU cannot read or write to SCRSR directly.



16.3.2 Receive FIFO Data Register (SCFRDR)

SCFRDR is a 16-stage 8-bit FIFO register that stores serial receive data. The SCIF completes the reception of one byte of serial data by moving the received data from the receive shift register (SCRSR) into SCFRDR for storage. Continuous reception is possible until 16 bytes are stored.

The CPU can read but not write to SCFRDR. If data is read when there is no receive data in the SCFRDR, the value is undefined. When this register is full of receive data, subsequent serial data is lost.

Bit	Bit Name	Initial value	R/W	Description
7 to 0		Undefined	R	FIFO for receive serial data

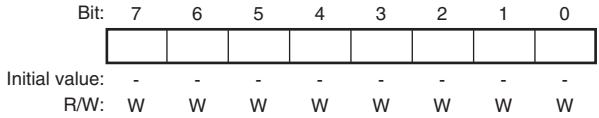
16.3.3 Transmit Shift Register (SCTSR)

SCTSR transmits serial data. The SCIF loads transmit data from the transmit FIFO data register (SCFTDR) into SCTSR, then transmits the data serially from the TXD pin, LSB (bit 0) first. After transmitting one data byte, the SCIF automatically loads the next transmit data from SCFTDR into SCTSR and starts transmitting again. The CPU cannot read or write to SCTSR directly.

16.3.4 Transmit FIFO Data Register (SCFTDR)

SCFTDR is a 16-stage 8-bit FIFO register that stores data for serial transmission. When the SCIF detects that the transmit shift register (SCTSR) is empty, it moves transmit data written in the SCFTDR into SCTSR and starts serial transmission. Continuous serial transmission is performed until there is no transmit data left in SCFTDR. SCFTDR can always be written to by the CPU.

When SCFTDR is full of transmit data (16 bytes), no more data can be written. If writing of new data is attempted, the data is ignored.

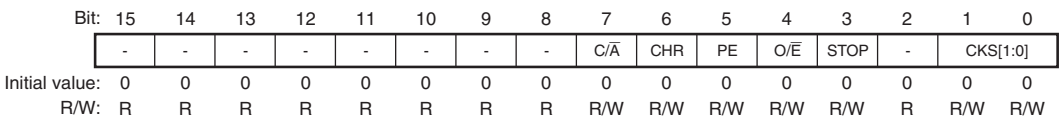


Bit	Bit Name	Initial value	R/W	Description
7 to 0		Undefined	W	FIFO for transmits serial data

16.3.5 Serial Mode Register (SCSMR)

SCSMR is a 16-bit register that specifies the SCIF serial communication format and selects the clock source for the baud rate generator.

The CPU can always read and write to SCSMR.



Bit	Bit Name	Initial value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial value	R/W	Description
7	C/$\overline{A}$	0	R/W	Communication Mode Selects whether the SCIF operates in asynchronous or clock synchronous mode. 0: Asynchronous mode 1: Clock synchronous mode
6	CHR	0	R/W	Character Length Selects 7-bit or 8-bit data in asynchronous mode. In the clock synchronous mode, the data length is always eight bits, regardless of the CHR setting. 0: 8-bit data 1: 7-bit data* Note: * When 7-bit data is selected, the MSB (bit 7) of the transmit FIFO data register is not transmitted.
5	PE	0	R/W	Parity Enable Selects whether to add a parity bit to transmit data and to check the parity of receive data, in asynchronous mode. In clock synchronous mode, a parity bit is neither added nor checked, regardless of the PE setting. 0: Parity bit not added or checked 1: Parity bit added and checked* Note: * When PE is set to 1, an even or odd parity bit is added to transmit data, depending on the parity mode (O/$\overline{E}$) setting. Receive data parity is checked according to the even/odd (O/$\overline{E}$) mode setting.

Bit	Bit Name	Initial value	R/W	Description
4	O/$\overline{E}$	0	R/W	Parity mode Selects even or odd parity when parity bits are added and checked. The O/$\overline{E}$ setting is used only in asynchronous mode and only when the parity enable bit (PE) is set to 1 to enable parity addition and checking. The O/$\overline{E}$ setting is ignored in clock synchronous mode, or in asynchronous mode when parity addition and checking is disabled. 0: Even parity*1 1: Odd parity*2 Notes: 1. If even parity is selected, the parity bit is added to transmit data to make an even number of 1s in the transmitted character and parity bit combined. Receive data is checked to see if it has an even number of 1s in the received character and parity bit combined. 2. If odd parity is selected, the parity bit is added to transmit data to make an odd number of 1s in the transmitted character and parity bit combined. Receive data is checked to see if it has an odd number of 1s in the received character and parity bit combined.
3	STOP	0	R/W	Stop Bit Length Selects one or two bits as the stop bit length in asynchronous mode. This setting is used only in asynchronous mode. It is ignored in clock synchronous mode because no stop bits are added. When receiving, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit, but if the second stop bit is 0, it is treated as the start bit of the next incoming character. 0: One stop bit When transmitting, a single 1-bit is added at the end of each transmitted character. 1: Two stop bits When transmitting, two 1 bits are added at the end of each transmitted character.

Bit	Bit Name	Initial value	R/W	Description
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1, 0	CKS[1:0]	00	R/W	Clock Select 1 and 0 Select the internal clock source of the on-chip baud rate generator. Four clock sources are available. Pϕ, Pϕ/4, Pϕ/16 and Pϕ/64. For further information on the clock source, bit rate register settings, and baud rate, see section 16.3.8, Bit Rate Register (SCBRR). 00: Pϕ 01: Pϕ/4 10: Pϕ/16 11: Pϕ/64 Note: Pϕ: Peripheral clock

16.3.6 Serial Control Register (SCSCR)

SCSCR is a 16-bit register that operates the SCIF transmitter/receiver, enables/disables interrupt requests, and selects the transmit/receive clock source. The CPU can always read and write to SCSCR.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TIE	RIE	TE	RE	REIE	-	CKE[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	TIE	0	R/W	Transmit Interrupt Enable Enables or disables the transmit-FIFO-data-empty interrupt (TXIF). Serial transmit data in the transmit FIFO data register (SCFTDR) is send to the transmit shift register (SCTSR). Then, the TDFE flag in the serial status register (SCFSR) is set to1 when the number of data in SCFTDR becomes less than the number of transmission triggers. At this time, a TXIF is requested. 0: Transmit-FIFO-data-empty interrupt request (TXIF) is disabled* 1: Transmit-FIFO-data-empty interrupt request (TXIF) is enabled Note: * The TXIF interrupt request can be cleared by writing a greater number of transmit data than the specified transmission trigger number to SCFTDR and by clearing the TDFE bit to 0 after reading 1 from the TDFE bit, or can be cleared by clearing this bit to 0.

Bit	Bit Name	Initial value	R/W	Description
6	RIE	0	R/W	Receive Interrupt Enable Enables or disables the receive-data-full (RXIF) interrupts requested when the RDF flag or DR flag in serial status register (SCFSR) is set to1, receive-error (ERIF) interrupts requested when the ER flag in SCFSR is set to1, and break (BRIF) interrupts requested when the BRK flag in SCFSR or the ORER flag in line status register (SCLSR) is set to1. 0: Receive-data-full interrupt (RXIF), receive-error interrupt (ERIF), and break interrupt (BRIF) requests are disabled* 1: Receive-data-full interrupt (RXIF), receive-error interrupt (ERIF), and break interrupt (BRIF) requests are enabled Note: * RXIF interrupt requests can be cleared by reading the DR or RDF flag after it has been set to 1, then clearing the flag to 0, or by clearing RIE to 0. ERIF or BRIF interrupt requests can be cleared by reading the ER, BR or ORER flag after it has been set to 1, then clearing the flag to 0, or by clearing RIE and REIE to 0.
5	TE	0	R/W	Transmit Enable Enables or disables the SCIF serial transmitter. 0: Transmitter disabled 1: Transmitter enabled* Note: * Serial transmission starts after writing of transmit data into SCFTDR. Select the transmit format in SCSMR and SCFCR and reset the transmit FIFO before setting TE to 1.

Bit	Bit Name	Initial value	R/W	Description
4	RE	0	R/W	Receive Enable Enables or disables the SCIF serial receiver. 0:Receiver disabled*[1] 1: Receiver enabled*[2] Notes: 1. Clearing RE to 0 does not affect the receive flags (DR, ER, BRK, RDF, FER, PER, and ORER). These flags retain their previous values. 2. Serial reception starts when a start bit is detected in asynchronous mode, or synchronous clock input is detected in clock synchronous mode. Select the receive format in SCSMR and SCFCR and reset the receive FIFO before setting RE to 1.
3	REIE	0	R/W	Receive Error Interrupt Enable Enables or disables the receive-error (ERIF) interrupts and break (BRIF) interrupts. The setting of REIE bit is valid only when RIE bit is set to 0. 0: Receive-error interrupt (ERIF) and break interrupt (BRIF) requests are disabled* 1: Receive-error interrupt (ERIF) and break interrupt (BRIF) requests are enabled Note: * ERIF or BRIF interrupt requests can be cleared by reading the ER, BR or ORER flag after it has been set to 1, then clearing the flag to 0, or by clearing RIE and REIE to 0. Even if RIE is set to 0, when REIE is set to 1, ERIF or BRIF interrupt requests are enabled.

Bit	Bit Name	Initial value	R/W	Description
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1, 0	CKE[1:0]	00	R/W	Clock Enable 1 and 0 Select the SCIF clock source and enable or disable clock output from the SCK pin. Depending on the combination of CKE1 and CKE0, the SCK pin can be used for serial clock output or serial clock input. The CKE0 setting is valid only when the SCIF is operating on the internal clock (CKE1 = 0). The CKE0 setting is ignored when an external clock source is selected (CKE1 = 1). In clock synchronous mode, select the SCIF operating mode in the serial mode register (SCSMR), then set CKE1 and CKE0. • Asynchronous mode 00: Internal clock, SCK pin used for input pin (The input signal is ignored. The state of the SCK pin depends on both the SCKIO and SCKDT bits.) 01: Internal clock, SCK pin used for clock output (The output clock frequency is 16 times the bit rate.) 10: External clock, SCK pin used for clock input (The input clock frequency is 16 times the bit rate.) 11: Setting prohibited • Clock synchronous mode 00: Internal clock, SCK pin used for serial clock output 01: Internal clock, SCK pin used for serial clock output 10: External clock, SCK pin used for serial clock input 11: Setting prohibited

16.3.7 Serial Status Register (SCFSR)

SCFSR is a 16-bit register. The upper 8 bits indicate the number of receives errors in the SCFRDR data, and the lower 8 bits indicate the status flag indicating SCIF operating state.

The CPU can always read and write to SCFSR, but cannot write 1 to the status flags (ER, TEND, TDFE, BRK, RDF, and DR). These flags can be cleared to 0 only if they have first been read (after being set to 1). Bits 3 (FER) and 2 (PER) are read-only bits that cannot be written.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PER[3:0]				FER[3:0]				ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Initial value:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R/(W)*	R/(W)*

Note: * To clear the flag, only 0 can be written after reading 1.

Bit	Bit Name	Initial value	R/W	Description
15 to 12	PER[3:0]	0000	R	Number of Parity Errors Indicate the number of data including a parity error in the receive data stored in the receive FIFO data register (SCFRDR). After the ER bit in SCFSR is set to 1, the value indicated by bits 15 to 12 indicates the number of parity errors in SCFRDR. When parity errors have occurred in all 16-byte receive data in SCFRDR, PER3 to PER0 show 0.
11 to 8	FER[3:0]	0000	R	Number of Framing Errors Indicate the number of data including a framing error in the receive data stored in SCFRDR. After the ER bit in SCFSR is set to 1, the value indicated by bits 11 to 8 indicates the number of framing errors in SCFRDR. When framing errors have occurred in all 16-byte receive data in SCFRDR, FER3 to FER0 show 0.

Bit	Bit Name	Initial value	R/W	Description
7	ER	0	R/(W)*	Receive Error Indicates the occurrence of a framing error, or of a parity error when receiving data that includes parity. *1 0: Receiving is in progress or has ended normally [Clearing conditions] • ER is cleared to 0 a power-on reset • ER is cleared to 0 when the chip is when 0 is written after 1 is read from ER 1: A framing error or parity error has occurred. [Setting conditions] • ER is set to 1 when the stop bit is 0 after checking whether or not the last stop bit of the received data is 1 at the end of one data receive operation*2 • ER is set to 1 when the total number of 1s in the receive data plus parity bit does not match the even/odd parity specified by the O/$\overline{E}$ bit in SCSMR Notes: 1. Clearing the RE bit to 0 in SCSCR does not affect the ER bit, which retains its previous value. Even if a receive error occurs, the receive data is transferred to SCFRDR and the receive operation is continued. Whether or not the data read from SCFRDR includes a receive error can be detected by the FER and PER bits in SCFSR. 2. In two stop bits mode, only the first stop bit is checked; the second stop bit is not checked.

Bit	Bit Name	Initial value	R/W	Description
6	TEND	1	R/(W)*	Transmit End Indicates that when the last bit of a serial character was transmitted, SCFTDR did not contain valid data, so transmission has ended. 0: Transmission is in progress [Clearing condition] • TEND is cleared to 0 when 0 is written after 1 is read from TEND after transmit data is written in SCFTDR 1: End of transmission [Setting conditions] • TEND is set to 1 when the chip is a power-on reset • TEND is set to 1 when TE is cleared to 0 in the serial control register (SCSCR) • TEND is set to 1 when SCFTDR does not contain receive data when the last bit of a one-byte serial character is transmitted Note: When data is written to SCFTDR by activating the DTC through a TXIF interrupt, the TEND flag value is undefined. In this case, do not use the TEND flag as a transmit end flag.

Bit	Bit Name	Initial value	R/W	Description
5	TDFE	1	R/(W)*	Transmit FIFO Data Empty Indicates that data has been transferred from the transmit FIFO data register (SCFTDR) to the transmit shift register (SCTSR), the number of data in SCFTDR has become less than the transmission trigger number specified by the TTRG1 and TTRG0 bits in the FIFO control register (SCFCR), and writing of transmit data to SCFTDR is enabled. 0: The number of transmit data written to SCFTDR is greater than the specified transmission trigger number [Clearing conditions] • TDFE is cleared to 0 when data exceeding the specified transmission trigger number is written to SCFTDR after 1 is read from the TDFE bit and then 0 is written • TDFE is cleared to 0 when data exceeding the specified transmission trigger number is written to SCFTDR by using the DTC 1: The number of transmit data in SCFTDR is less than or equal to the specified transmission trigger number* [Setting conditions] • TDFE is set to 1 by a power-on reset • TDFE is set to 1 when the number of transmit data in SCFTDR becomes less than or equal to the specified transmission trigger number as a result of transmission Note: * Since SCFTDR is a 16-byte FIFO register, the maximum number of data that can be written when TDFE is 1 is "16 minus the specified transmission trigger number". If an attempt is made to write additional data, the data is ignored. The number of data in SCFTDR is indicated by the upper 8 bits of SCFDR.

Bit	Bit Name	Initial value	R/W	Description
4	BRK	0	R/(W)*	Break Detection Indicates that a break signal has been detected in receive data. 0: No break signal received [Clearing conditions] • BRK is cleared to 0 when the chip is a power-on reset • BRK is cleared to 0 when software reads BRK after it has been set to 1, then writes 0 to BRK 1: Break signal received* [Setting condition] • BRK is set to 1 when data including a framing error is received, and a framing error occurs with space 0 in the subsequent receive data Note: * When a break is detected, transfer of the receive data (H'00) to SCFRDR stops after detection. When the break ends and the receive signal becomes mark 1, the transfer of receive data resumes.
3	FER	0	R	Framing Error Indicates a framing error in the data read from the next receive FIFO data register (SCFRDR) in asynchronous mode. 0: No receive framing error occurred in the next data read from SCFRDR [Clearing conditions] • FER is cleared to 0 when the chip undergoes a power-on reset • FER is cleared to 0 when no framing error is present in the next data read from SCFRDR 1: A receive framing error occurred in the next data read from SCFRDR. [Setting condition] • FER is set to 1 when a framing error is present in the next data read from SCFRDR

Bit	Bit Name	Initial value	R/W	Description
2	PER	0	R	Parity Error Indicates a parity error in the data read from the next receive FIFO data register (SCFRDR) in asynchronous mode. 0: No receive parity error occurred in the next data read from SCFRDR [Clearing conditions] • PER is cleared to 0 when the chip undergoes a power-on reset • PER is cleared to 0 when no parity error is present in the next data read from SCFRDR 1: A receive parity error occurred in the next data read from SCFRDR [Setting condition] • PER is set to 1 when a parity error is present in the next data read from SCFRDR

Bit	Bit Name	Initial value	R/W	Description
1	RDF	0	R/(W)*	Receive FIFO Data Full Indicates that receive data has been transferred to the receive FIFO data register (SCFRDR), and the number of data in SCFRDR has become more than the receive trigger number specified by the RTRG1 and RTRG0 bits in the FIFO control register (SCFCR). 0: The number of transmit data written to SCFRDR is less than the specified receive trigger number [Clearing conditions] • RDF is cleared to 0 by a power-on reset • RDF is cleared to 0 when the SCFRDR is read until the number of receive data in SCFRDR becomes less than the specified receive trigger number after 1 is read from RDF and then 0 is written • RDF is cleared to 0 when the SCFRDR is read by using the DTC until the number of receive data in SCFRDR becomes less than the specified receive trigger number 1: The number of receive data in SCFRDR is more than the specified receive trigger number [Setting condition] • RDF is set to 1 when a number of receive data more than the specified receive trigger number is stored in SCFRDR* Note: * SCFTDR is a 16-byte FIFO register. When RDF is 1, the specified receive trigger number of data can be read at the maximum. If an attempt is made to read after all the data in SCFRDR has been read, the data is undefined. The number of receive data in SCFRDR is indicated by the lower 8 bits of SCFDR.

Bit	Bit Name	Initial value	R/W	Description
0	DR	0	R/(W)*	Receive Data Ready Indicates that the number of data in the receive FIFO data register (SCFRDR) is less than the specified receive trigger number, and that the next data has not yet been received after the elapse of 15 ETU from the last stop bit in asynchronous mode. In clock synchronous mode, this bit is not set to 1. 0: Receiving is in progress, or no receive data remains in SCFRDR after receiving ended normally [Clearing conditions] • DR is cleared to 0 when the chip undergoes a power-on reset • DR is cleared to 0 when all receive data are read after 1 is read from DR and then 0 is written • DR is cleared to 0 when all receive data in SCFRDR are read by the DTC 1: Next receive data has not been received [Setting condition] • DR is set to 1 when SCFRDR contains less data than the specified receive trigger number, and the next data has not yet been received after the elapse of 15 ETU from the last stop bit.* Note: * This is equivalent to 1.5 frames with the 8-bit, 1-stop-bit format. (ETU: Elementary time unit)

Note: * To clear the flag, only 0 can be written after reading 1.

16.3.8 Bit Rate Register (SCBRR)

SCBRR is an 8-bit register that, together with the baud rate generator clock source selected by the CKS1 and CKS0 bits in the serial mode register (SCSMR), determines the serial transmit/receive bit rate.

The CPU can always read and write to SCBRR. SCBRR is initialized to H'FF by a power-on reset.

The SCBRR setting is calculated as follows:

Bit:	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Asynchronous mode:

$$N = \frac{P\phi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

- Clock synchronous mode:

$$N = \frac{P\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bits/s)
N: SCBRR setting for baud rate generator ($0 \leq N \leq 255$)
(The setting value should satisfy the electrical characteristics.)
$P\phi$: Operating frequency for peripheral modules (MHz)
n: Baud rate generator clock source (n = 0, 1, 2, 3) (for the clock sources and values of n, see table 16.3.)

Table 16.3 SCSMR Settings

		SCSMR Settings	
n	**Clock Source**	**CKS1**	**CKS0**
0	Pϕ	0	0
1	Pϕ/4	0	1
2	Pϕ/16	1	0
3	Pϕ/64	1	1

Note: The bit rate error in asynchronous is given by the following formula:

$$\text{Error (\%)} = \left\{ \frac{P\phi \times 10^6}{(N + 1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

Tables 16.4 to 16.6 list examples of SCBRR settings in asynchronous mode, and tables 16.7 to 16.9 list examples of SCBRR settings in clock synchronous mode.

Table 16.4 Bit Rates and SCBRR Settings in Asynchronous Mode

	Pϕ (MHz)																	
Bit Rate (bits/s)	10			12			14			16			18			20		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	177	-0.25	2	212	0.03	2	248	-0.17	3	70	0.03	3	79	-0.12	3	88	-0.25
150	2	129	0.16	2	155	0.16	2	181	0.16	2	207	0.16	2	233	0.16	3	64	0.16
300	2	64	0.16	2	77	0.16	2	90	0.16	2	103	0.16	2	116	0.16	2	129	0.16
600	1	129	0.16	1	155	0.16	1	181	0.16	1	207	0.16	1	233	0.16	2	64	0.16
1200	1	64	0.16	1	77	0.16	1	90	0.16	1	103	0.16	1	116	0.16	1	129	0.16
2400	0	129	0.16	0	155	0.16	0	181	0.16	0	207	0.16	0	233	0.16	1	64	0.16
4800	0	64	0.16	0	77	0.16	0	90	0.16	0	103	0.16	0	116	0.16	0	129	0.16
9600	0	32	-1.36	0	38	0.16	0	45	-0.93	0	51	0.16	0	58	-0.69	0	64	0.16
14400	0	21	-1.36	0	25	0.16	0	29	1.27	0	34	-0.79	0	38	0.16	0	42	0.94
19200	0	15	1.73	0	19	-2.34	0	22	-0.93	0	25	0.16	0	28	1.02	0	32	-1.36
28800	0	10	-1.36	0	12	0.16	0	14	1.27	0	16	2.12	0	19	-2.34	0	21	-1.36
31250	0	9	0.00	0	11	0.00	0	13	0.00	0	15	0.00	0	17	0.00	0	19	0.00
38400	0	7	1.73	0	9	-2.34	0	10	3.57	0	12	0.16	0	14	-2.34	0	15	1.73

Table 16.5 Bit Rates and SCBRR Settings in Asynchronous Mode

Bit Rate (bits/s)	Pϕ (MHz)																	
	22			24			26			28			30			32		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	3	97	-0.35	3	106	-0.44	3	114	0.36	3	123	0.23	3	132	0.13	3	141	0.03
150	3	71	-0.54	3	77	0.16	3	84	-0.43	3	90	0.16	3	97	-0.35	3	103	0.16
300	2	142	0.16	2	155	0.16	2	168	0.16	2	181	0.16	2	194	0.16	2	207	0.16
600	2	71	-0.54	2	77	0.16	2	84	-0.43	2	90	0.16	2	97	-0.35	2	103	0.16
1200	1	142	0.16	1	155	0.16	1	168	0.16	1	181	0.16	1	194	0.16	1	207	0.16
2400	1	71	-0.54	1	77	0.16	1	84	-0.43	1	90	0.16	1	97	-0.35	1	103	0.16
4800	0	142	0.16	0	155	0.16	0	168	0.16	0	181	0.16	0	194	0.16	0	207	0.16
9600	0	71	-0.54	0	77	0.16	0	84	-0.43	0	90	0.16	0	97	-0.35	0	103	0.16
14400	0	47	-0.54	0	51	0.16	0	55	0.76	0	60	-0.39	0	64	0.16	0	68	0.64
19200	0	35	-0.54	0	38	0.16	0	41	0.76	0	45	-0.93	0	48	-0.35	0	51	0.16
28800	0	23	-0.54	0	25	0.16	0	27	0.76	0	29	1.27	0	32	-1.36	0	34	-0.79
31250	0	21	0.00	0	23	0.00	0	25	0.00	0	27	0.00	0	29	0.00	0	31	0.00
38400	0	17	-0.54	0	19	-2.34	0	20	0.76	0	22	-0.93	0	23	1.73	0	25	0.16

Table 16.6 Bit Rates and SCBRR Settings in Asynchronous Mode

Bit Rate (bits/s)	Pϕ (MHz) 34 n	N	Error (%)	36 n	N	Error (%)	38 n	N	Error (%)	40 n	N	Error (%)
110	3	150	-0.05	3	159	-0.12	3	168	-0.19	3	177	-0.25
150	3	110	-0.29	3	116	0.16	3	123	-0.24	3	129	0.16
300	2	220	0.16	2	233	0.16	2	246	0.16	3	64	0.16
600	2	110	-0.29	2	116	0.16	2	123	-0.24	2	129	0.16
1200	1	220	0.16	1	233	0.16	1	246	0.16	2	64	0.16
2400	1	110	-0.29	1	116	0.16	1	123	-0.24	1	129	0.16
4800	0	220	0.16	0	233	0.16	0	246	0.16	1	64	0.16
9600	0	110	-0.29	0	116	0.16	0	123	-0.24	0	129	0.16
14400	0	73	-0.29	0	77	0.16	0	81	0.57	0	86	-0.22
19200	0	54	0.62	0	58	-0.69	0	61	-0.24	0	64	0.16
28800	0	36	-0.29	0	38	0.16	0	40	0.57	0	42	0.94
31250	0	33	0.00	0	35	0.00	0	37	0.00	0	39	0.00
38400	0	27	-1.18	0	28	1.02	0	30	-0.24	0	32	-1.36

Table 16.7 Bit Rates and SCBRR Settings in Clock Synchronous Mode

Bit Rate (bits/s)	Pϕ (MHz)											
	10		12		14		16		18		20	
	n	N	n	N	n	N	n	N	n	N	n	N
250	3	155	3	187	3	218	3	249				
500	3	77	3	93	3	108	3	124	3	140	3	155
1000	2	155	2	187	2	218	2	249	3	69	3	77
2500	1	249	2	74	2	87	2	99	2	112	2	124
5000	1	124	1	149	1	174	1	199	1	224	1	249
10000	0	249	1	74	1	87	1	99	1	112	1	124
25000	0	99	0	119	0	139	0	159	0	179	0	199
50000	0	49	0	59	0	69	0	79	0	89	0	99
100000	0	24	0	29	0	34	0	39	0	44	0	49
250000	0	9	0	11	0	13	0	15	0	17	0	19
500000	0	4	0	5	0	6	0	7	0	8	0	9
1000000	—	—	0	2	—	—	0	3	—	—	0	4
2500000	0	0*	—	—	—	—	—	—	—	—	0	1
5000000			—	—	—	—	—	—	—	—	0	0*

Table 16.8 Bit Rates and SCBRR Settings in Clock Synchronous Mode

Bit Rate (bits/s)	Pϕ (MHz)											
	22		24		26		28		30		32	
	n	N	n	N	n	N	n	N	n	N	n	N
250												
500	3	171	3	187	3	202	3	218	3	233	3	249
1000	3	85	3	93	3	101	3	108	3	116	3	124
2500	2	137	2	149	2	162	2	174	2	187	2	199
5000	2	68	2	74	2	80	2	87	2	93	2	99
10000	1	137	1	149	1	162	1	174	1	187	1	199
25000	0	219	0	239	1	64	1	69	1	74	1	79
50000	0	109	0	119	0	129	0	139	0	149	0	159
100000	0	54	0	59	0	64	0	69	0	74	0	79
250000	0	21	0	23	0	25	0	27	0	29	0	31
500000	0	10	0	11	0	12	0	13	0	14	0	15
1000000	—	—	0	5	—	—	0	6	—	—	0	7
2500000	—	—	—	—	—	—	—	—	0	2	—	—
5000000	—	—	—	—	—	—	—	—	—	—	—	—

Table 16.9 Bit Rates and SCBRR Settings in Clock Synchronous Mode

Bit Rate (bits/s)	Pϕ (MHz)							
	34		36		38		40	
	n	N	n	N	n	N	n	N
250								
500								
1000	3	132	3	140	3	147	3	155
2500	2	212	2	224	2	237	2	249
5000	2	105	2	112	2	118	2	124
10000	1	212	1	224	1	237	1	249
25000	1	84	1	89	1	94	1	99
50000	0	169	0	179	0	189	0	199
100000	0	84	0	89	0	94	0	99
250000	0	33	0	35	0	37	0	39
500000	0	16	0	17	0	18	0	19
1000000	—	—	0	8	—	—	0	9
2500000	—	—	—	—	—	—	0	3
5000000	—	—	—	—	—	—	0	1

[Legend]
Blank: No setting possible
—: Setting possible, but error occurs
*: Continuous transmission/reception is disabled.
Note: Settings with an error of 1% or less are recommended.

Table 16.10 indicates the maximum bit rates in asynchronous mode when the baud rate generator is used. Tables 16.11 and 16.12 list the maximum rates for external clock input.

Table 16.10 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (Asynchronous Mode)

		Settings	
Pϕ (MHz)	**Maximum Bit Rate (bits/s)**	**n**	**N**
10	312500	0	0
12	375000	0	0
14	437500	0	0
16	500000	0	0
18	562500	0	0
20	625000	0	0
22	687500	0	0
24	750000	0	0
26	812500	0	0
28	875000	0	0
30	937500	0	0
32	1000000	0	0
34	1062500	0	0
36	1125000	0	0
38	1187500	0	0
40	1250000	0	0

Table 16.11 Maximum Bit Rates with External Clock Input (Asynchronous Mode)

Pϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bits/s)
10	2.5000	156250
12	3.0000	187500
14	3.5000	218750
16	4.0000	250000
18	4.5000	281250
20	5.0000	312500
22	5.5000	343750
24	6.0000	375000
26	6.5000	406250
28	7.0000	437500
30	7.5000	468750
32	8.0000	500000
34	8.5000	531250
36	9.0000	562500
38	9.5000	593750
40	10.0000	625000

Table 16.12 Maximum Bit Rates with External Clock Input (Clock Synchronous Mode)

Pϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bits/s)
10	1.6667	1666666.7
12	2.0000	2000000.0
14	2.3333	2333333.3
16	2.6667	2666666.7
18	3.0000	3000000.0
20	3.3333	3333333.3
22	3.6667	3666666.7
24	4.0000	4000000.0
26	4.3333	4333333.3
28	4.6667	4666666.7
30	5.0000	5000000.0
32	5.3333	5333333.3
34	5.6667	5666666.7
36	6.0000	6000000.0
38	6.3333	6333333.3
40	6.6667	6666666.7

16.3.9 FIFO Control Register (SCFCR)

SCFCR is a 16-bit register that resets the number of data in the transmit and receive FIFO registers, sets the trigger data number, and contains an enable bit for loop-back testing. SCFCR can always be read and written to by the CPU.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	RSTRG[2:0]			RTRG[1:0]		TTRG[1:0]		MCE	TFRST	RFRST	LOOP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	RSTRG[2:0]	000	R/W	$\overline{\text{RTS}}$ Output Active Trigger When the number of receive data in the receive FIFO register (SCFRDR) becomes more than the number shown below, the $\overline{\text{RTS}}$ signal is set to high. These bits are valid only when modem control signals are enabled in asynchronous mode. 000: 15 001: 1 010: 4 011: 6 100: 8 101: 10 110: 12 111: 14

Bit	Bit Name	Initial value	R/W	Description
7, 6	RTRG[1:0]	00	R/W	Receive FIFO Data Trigger Set the specified receive trigger number. The receive data full (RDF) flag in the serial status register (SCFSR) is set when the number of receive data stored in the receive FIFO register (SCFRDR) exceeds the specified trigger number shown below. • Asynchronous mode 00: 1 01: 4 10: 8 11: 14 • Clock synchronous mode 00: 1 01: 2 10: 8 11: 14
5, 4	TTRG[1:0]	00	R/W	Transmit FIFO Data Trigger 1 and 0 Set the specified transmit trigger number. The transmit FIFO data register empty (TDFE) flag in the serial status register (SCFSR) is set when the number of transmit data in the transmit FIFO data register (SCFTDR) becomes less than the specified trigger number shown below. 00: 8 (8)* 01: 4 (12)* 10: 2 (14)* 11: 0 (16)* Note: * Values in parentheses mean the number of remaining bytes in SCFTDR when the TDFE flag is set to 1.

Bit	Bit Name	Initial value	R/W	Description
3	MCE	0	R/W	Modem Control Enable Enables modem control signals $\overline{\text{CTS}}$ and $\overline{\text{RTS}}$. In clock synchronous mode, clear this bit to 0. 0: Modem signal disabled* 1: Modem signal enabled Note: * Regardless of the input value, the $\overline{\text{CTS}}$ level has no effect on transmit operation and the $\overline{\text{RTS}}$ level has no effect on receive operation.
2	TFRST	0	R/W	Transmit FIFO Data Register Reset Disables the transmit data in the transmit FIFO data register and resets the data to the empty state. 0: Reset operation disabled* 1: Reset operation enabled Note: * Reset operation is executed by a power-on reset.
1	RFRST	0	R/W	Receive FIFO Data Register Reset Disables the receive data in the receive FIFO data register and resets the data to the empty state. 0: Reset operation disabled* 1: Reset operation enabled Note: * Reset operation is executed by a power-on reset.
0	LOOP	0	R/W	Loop-Back Test Internally connects the transmit output pin (TXD) and receive input pin (RXD) and internally connects the $\overline{\text{RTS}}$ pin and $\overline{\text{CTS}}$ pin and enables loop-back testing. 0: Loop back test disabled 1: Loop back test enabled

16.3.10 FIFO Data Count Register (SCFDR)

SCFDR is a 16-bit register which indicates the number of data stored in the transmit FIFO data register (SCFTDR) and the receive FIFO data register (SCFRDR). It indicates the number of transmit data in SCFTDR with the upper eight bits, and the number of receive data in SCFRDR with the lower eight bits. SCFDR can always be read from by the CPU.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	T[4:0]					-	-	-	R[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	T[4:0]	00000	R	Indicate the number of non-transmitted data stored in SCFTDR. H'00 means no transmit data, and H'10 means that SCFTDR is full of transmit data.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	R[4:0]	00000	R	Indicate the number of receive data stored in SCFRDR. H'00 means no receive data, and H'10 means that SCFRDR full of receive data.

16.3.11 Serial Port Register (SCSPTR)

SCSPTR is a 16-bit register that controls input/output and data for the pins multiplexed to the SCIF function. Bits 7 and 6 can control the $\overline{\text{RTS}}$ pin, bits 5 and 4 can control the $\overline{\text{CTS}}$ pin, and bits 3 and 2 can control the SCK pin. Bits 1 and 0 can be used to output data to the TXD pin, so they control break of serial transfer. In addition to descriptions of individual bits shown below, see section 16.6, Serial Port Register (SCSPTR) and SCIF Pins.

SCSPTR can always be read from or written to by the CPU. Note that the respective port registers should be used to read the values on the SCIF pins. For details, refer to section 22, I/O Ports.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPBIO	SPBDT
Initial value:	0	0	0	0	0	0	0	0	0	-	0	-	0	-	0	-
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	RTSIO	0	R/W	RTS Port Input/Output Control Controls the $\overline{\text{RTS}}$ pin in combination with the RTSDT bit in this register and the MCE bit in SCFCR.
6	RTSDT	Undefined	R/W	RTS Port Data Controls the $\overline{\text{RTS}}$ pin in combination with the RTSIO bit in this register and the MCE bit in SCFCR. Select the $\overline{\text{RTS}}$ pin function in the PFC (pin function controller) beforehand. MCE RTSIO RTSDT: $\overline{\text{RTS}}$ pin state 0 0 ×: Setting prohibited (initial state) 0 1 0: Low level output 0 1 1: High level output 1 × ×: Sequence output according to modem control logic Note: ×: Don't care

Bit	Bit Name	Initial value	R/W	Description
5	CTSIO	0	R/W	CTS Port Input/Output Control Controls the $\overline{\text{CTS}}$ pin in combination with the CTSDT bit in this register and the MCE bit in SCFCR.
4	CTSDT	Undefined	R/W	CTS Port Data Controls the $\overline{\text{CTS}}$ pin in combination with the CTSIO bit in this register and the MCE bit in SCFCR. Select the $\overline{\text{CTS}}$ pin function in the PFC (pin function controller) beforehand. MCE CTSIO CTSDT: $\overline{\text{CTS}}$ pin state 0 0 ×: Setting prohibited (initial state) 0 1 0: Low level output 0 1 1: High level output 1 × ×: Input to modem control logic Note: ×: Don't care
3	SCKIO	0	R/W	SCK Port Input/Output Control Controls the SCK pin in combination with the SCKDT bit in this register, the C/A bit in SCSMR, and bits CKE1 and CKE0 in SCSCR.

Bit	Bit Name	Initial value	R/W	Description
2	SCKDT	Undefined	R/W	SCK Port Data Controls the SCK pin in combination with the SCKIO bit in this register, the C/$\overline{A}$ bit in SCSMR, and bits CKE1 and CKE0 in SCSCR. Select the SCK pin function in the PFC (pin function controller) beforehand. C/$\overline{A}$ CKE1 CKE0 SCKIO SCKDT: SCK pin state 0 0 0 0 ×: Setting prohibited (initial state) 0 0 0 1 0: Low level output 0 0 0 1 1: High level output 0 0 1 × ×: Internal clock output according to serial core logic 0 1 0 × ×: External clock input to serial core logic 0 1 1 × ×: Setting prohibited 1 0 0 × ×: Internal clock output according to serial core logic 1 0 1 × ×: Internal clock output according to serial core logic 1 1 0 × ×: External clock input to serial core logic 1 1 1 × ×: Setting prohibited Note: ×: Don't care
1	SPBIO	0	R/W	Serial Port Break Output Control Controls the TXD pin in combination with the SPBDT bit in this register and the TE bit in SCSCR.
0	SPBDT	Undefined	R/W	Serial Port Break Data Controls the TXD pin in combination with the SPBIO bit in this register and the TE bit in SCSCR. Select the TXD pin function in the PFC (pin function controller) beforehand. TE SPBIO SPBDT: TXD pin state 0 0 ×: Setting prohibited (initial state) 0 1 0: Low level output 0 1 1: High level output 0 × ×: Transmit data output according to serial core logic Note: ×: Don't care

16.3.12 Line Status Register (SCLSR)

SCLSR is a 16-bit readable/writable register which can always be read from and written to by the CPU. However, a 1 cannot be written to the ORER flag. This flag can be cleared to 0 only if it has first been read (after being set to 1).

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ORER
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)*

Note: * To clear the flag, only 0 can be written after reading 1.

Bit	Bit Name	Initial value	R/W	Description
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ORER	0	R/(W)*	Overrun Error Indicates the occurrence of an overrun error. 0: Receiving is in progress or has ended normally *1 [Clearing conditions] • ORER is cleared to 0 when the chip is a power-on reset • ORER is cleared to 0 when 0 is written after 1 is read from ORER. 1: An overrun error has occurred *2 [Setting condition] • ORER is set to 1 when the next serial receiving is finished while receive FIFO data are full. Notes: 1. Clearing the RE bit to 0 in SCSCR does not affect the ORER bit, which retains its previous value. 2. The receive FIFO data register (SCFRDR) hold the data before an overrun error is occurred, and the next receive data is extinguished. When ORER is set to 1, SCIF can not continue the next serial receiving.

16.4 Operation

16.4.1 Overview

For serial communication, the SCIF has an asynchronous mode in which characters are synchronized individually, and a clock synchronous mode in which communication is synchronized with clock pulses. The SCIF has a 16-byte FIFO buffer for both transmit and receive operations, reducing the overhead of the CPU, and enabling continuous high-speed communication. Moreover, it has $\overline{\text{RTS}}$ and $\overline{\text{CTS}}$ signals as modem control signals. The transmission format is selected in the serial mode register (SCSMR) as shown in table 16.13. The SCIF clock source is selected by the combination of the CKE1 and CKE0 bits in the serial control register (SCSCR) as shown in table 16.14.

(1) Asynchronous Mode

- Data length is selectable: 7 or 8 bits.
- Parity bit is selectable. So is the stop bit length (1 or 2 bits). The combination of the preceding selections constitutes the communication format and character length.
- In receiving, it is possible to detect framing errors, parity errors, receive FIFO data full, overrun errors, receive data ready, and breaks.
- The number of stored data bytes is indicated for both the transmit and receive FIFO registers.
- An internal or external clock can be selected as the SCIF clock source.
  - When an internal clock is selected, the SCIF operates using the on-chip baud rate generator.
  - When an external clock is selected, the external clock input must have a frequency 16 times the bit rate. (The on-chip baud rate generator is not used.)

(2) Clock Synchronous Mode

- The transmission/reception format has a fixed 8-bit data length.
- In receiving, it is possible to detect overrun errors (ORER).
- An internal or external clock can be selected as the SCIF clock source.
  - When an internal clock is selected, the SCIF operates using the on-chip baud rate generator, and outputs a serial clock signal to external devices.
  - When an external clock is selected, the SCIF operates on the input serial clock. The on-chip baud rate generator is not used.

Table 16.13 SCSMR Settings and SCIF Communication Formats

SCSMR Settings					SCIF Communication Format		
Bit 7 C/$\overline{A}$	Bit 6 CHR	Bit 5 PE	Bit 3 STOP	Mode	Data Length	Parity Bit	Stop Bit Length
0	0	0	0	Asynchronous	8-bit	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
	1	0	0		7-bit	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
1	x	x	x	Clock synchronous	8-bit	Not set	None

[Legend]

x: Don't care

Table 16.14 SCSMR and SCSCR Settings and SCIF Clock Source Selection

SCSMR	SCSCR Settings			SCIF Transmit/Receive Clock	
Bit 7 C/$\overline{A}$	Bit 1 CKE1	Bit 0 CKE0	Mode	Clock Source	SCK Pin Function
0	0	0	Asynchronous	Internal	SCIF does not use the SCK pin. The state of the SCK pin depends on both the SCKIO and SCKDT bits.
		1			Clock with a frequency 16 times the bit rate is output.
	1	0		External	Input a clock with frequency 16 times the bit rate.
		1		—	Setting prohibited.
1	0	x	Clock synchronous	Internal	Serial clock is output.
	1	0		External	Input the serial clock.
		1		—	Setting prohibited.

[Legend]

x: Don't care

16.4.2 Operation in Asynchronous Mode

In asynchronous mode, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections of the SCIF are independent, so full duplex communication is possible. The transmitter and receiver are 16-byte FIFO buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmitting and receiving. Figure 16.2 shows the general format of asynchronous serial communication.

In asynchronous serial communication, the communication line is normally held in the mark (high) state. The SCIF monitors the line and starts serial communication when the line goes to the space (low) state, indicating a start bit. One serial character consists of a start bit (low), data (LSB first), parity bit (high or low), and stop bit (high), in that order.

When receiving in asynchronous mode, the SCIF synchronizes at the falling edge of the start bit. The SCIF samples each data bit on the eighth pulse of a clock with a frequency 16 times the bit rate. Receive data is latched at the center of each bit.

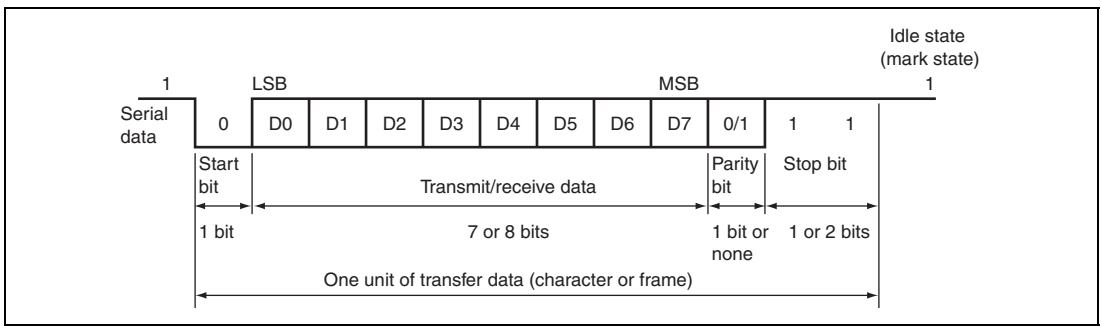


Figure 16.2 Example of Data Format in Asynchronous Communication (8-Bit Data with Parity and Two Stop Bits)

(1) Transmit/Receive Formats

Table 16.15 lists the eight communication formats that can be selected in asynchronous mode. The format is selected by settings in the serial mode register (SCSMR).

Table 16.15 Serial Communication Formats (Asynchronous Mode)

SCSMR Bits			Serial Transmit/Receive Format and Frame Length											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START	8-bit data								STOP		
0	0	1	START	8-bit data								STOP	STOP	
0	1	0	START	8-bit data								P	STOP	
0	1	1	START	8-bit data								P	STOP	STOP
1	0	0	START	7-bit data							STOP			
1	0	1	START	7-bit data							STOP	STOP		
1	1	0	START	7-bit data							P	STOP		
1	1	1	START	7-bit data							P	STOP	STOP	

[Legend]
START: Start bit
STOP: Stop bit
P: Parity bit

(2) Clock

An internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCIF transmit/receive clock. The clock source is selected by the C/$\overline{A}$ bit in the serial mode register (SCSMR) and bits CKE1 and CKE0 in the serial control register (SCSCR). For selection of the SCIF clock source, see table 16.14.

When an external clock is input at the SCK pin, it must have a frequency equal to 16 times the desired bit rate.

When the SCIF operates on an internal clock, it can output a clock signal at the SCK pin. The frequency of this output clock is equal to 16 times the desired bit rate.

(3) Transmitting and Receiving Data

SCIF Initialization (Asynchronous Mode):

Before transmitting or receiving, clear the TE and RE bits to 0 in the serial control register (SCSCR), then initialize the SCIF as follows.

When changing the operation mode or the communication format, always clear the TE and RE bits to 0 before following the procedure given below. Clearing TE to 0 initializes the transmit shift register (SCTSR). Clearing TE and RE to 0, however, does not initialize the serial status register (SCFSR), transmit FIFO data register (SCFTDR), or receive FIFO data register (SCFRDR), which retain their previous contents. Clear TE to 0 after all transmit data has been transmitted and the TEND flag in the SCFSR is set. The TE bit can be cleared to 0 during transmission, but the transmit data goes to the Mark state after the bit is cleared to 0. Set the TFRST bit in SCFCR to 1 and reset SCFTDR before TE is set again to start transmission.

When an external clock is used, the clock should not be stopped during initialization or subsequent operation. SCIF operation becomes unreliable if the clock is stopped.

Figure 16.3 shows a sample flowchart for initializing the SCIF.

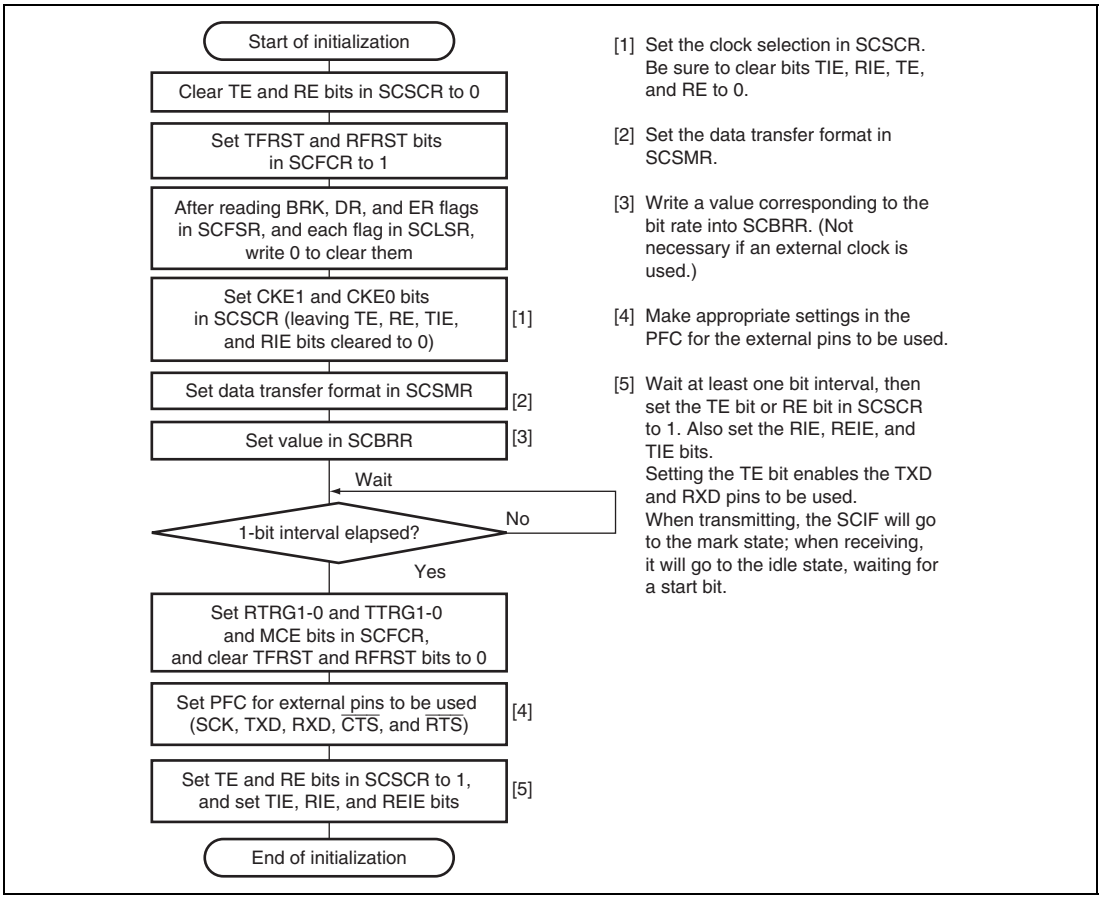


Figure 16.3 Sample Flowchart for SCIF Initialization

Transmitting Serial Data (Asynchronous Mode):

Figure 16.4 shows a sample flowchart for serial transmission.

Use the following procedure for serial data transmission after enabling the SCIF for transmission.

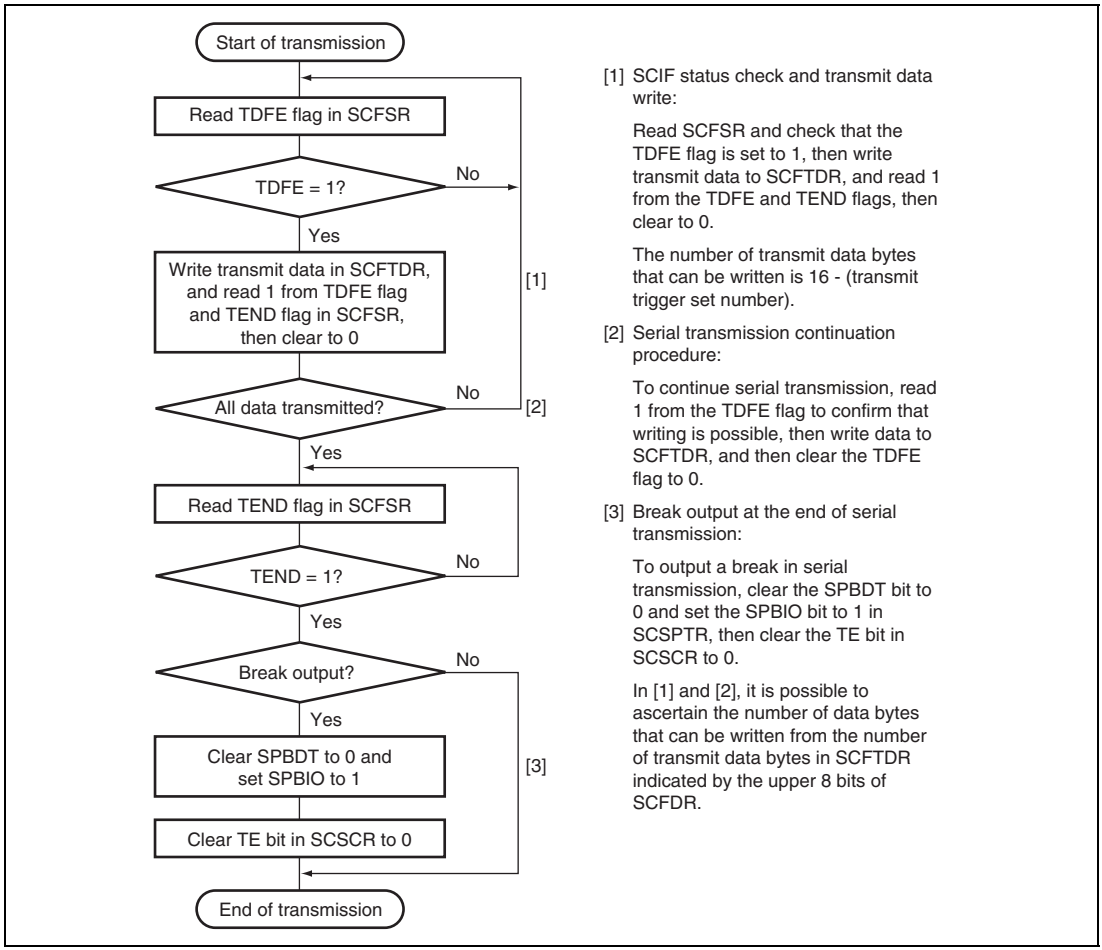


Figure 16.4 Sample Flowchart for Transmitting Serial Data

In serial transmission, the SCIF operates as described below.

1. When data is written into the transmit FIFO data register (SCFTDR), the SCIF transfers the data from SCFTDR to the transmit shift register (SCTSR) and starts transmitting. Confirm that the TDFE flag in the serial status register (SCFSR) is set to 1 before writing transmit data to SCFTDR. The number of data bytes that can be written is (16 – transmit trigger setting).
2. When data is transferred from SCFTDR to SCTSR and transmission is started, consecutive transmit operations are performed until there is no transmit data left in SCFTDR. When the number of transmit data bytes in SCFTDR falls below the transmit trigger number set in the FIFO control register (SCFCR), the TDFE flag is set. If the TIE bit in the serial control register (SCSR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXIF) request is generated.

   The serial transmit data is sent from the TXD pin in the following order.

   A. Start bit: One-bit 0 is output.
   B. Transmit data: 8-bit or 7-bit data is output in LSB-first order.
   C. Parity bit: One parity bit (even or odd parity) is output. (A format in which a parity bit is not output can also be selected.)
   D. Stop bit(s): One or two 1 bits (stop bits) are output.
   E. Mark state: 1 is output continuously until the start bit that starts the next transmission is sent.
3. The SCIF checks the SCFTDR transmit data at the timing for sending the stop bit. If data is present, the data is transferred from SCFTDR to SCTSR, the stop bit is sent, and then serial transmission of the next frame is started. If there is no transmit data, the TEND flag in SCFSR is set to 1, the stop bit is sent, and then the line goes to the mark state in which 1 is output continuously.

Figure 16.5 shows an example of the operation for transmission.

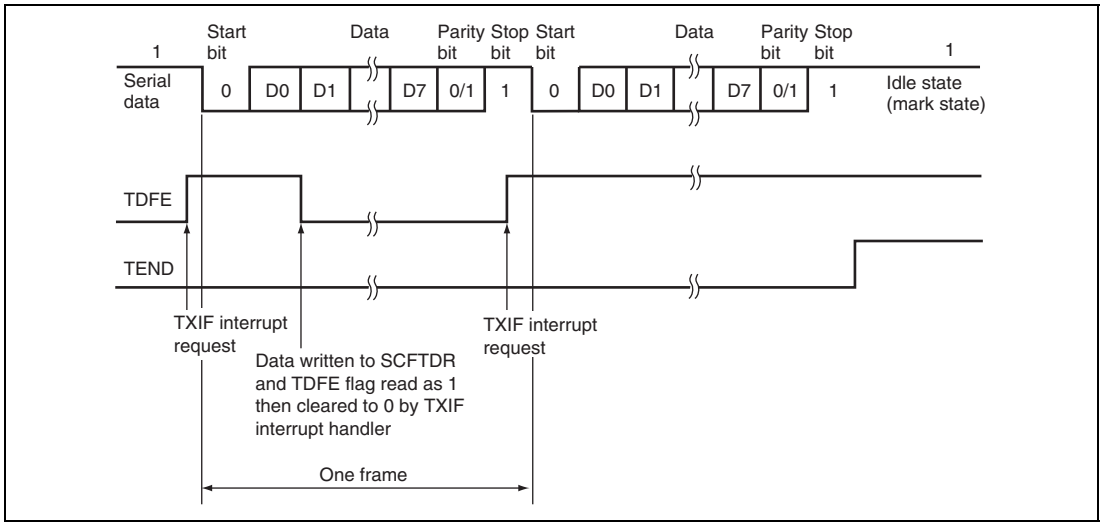


Figure 16.5 Example of Transmit Operation (8-Bit Data, Parity, One Stop Bit)

4. When modem control is enabled, transmission can be stopped and restarted in accordance with the $\overline{\text{CTS}}$ input value. When $\overline{\text{CTS}}$ is set to 1, if transmission is in progress, the line goes to the mark state after transmission of one frame. When $\overline{\text{CTS}}$ is set to 0, the next transmit data is output starting from the start bit.

   Figure 16.6 shows an example of the operation when modem control is used.

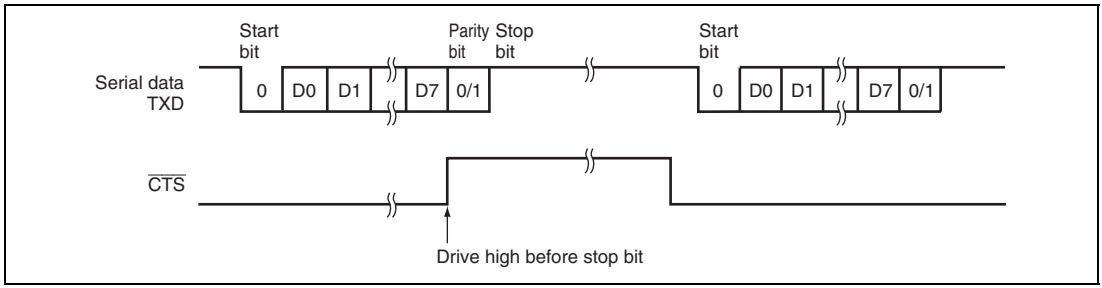
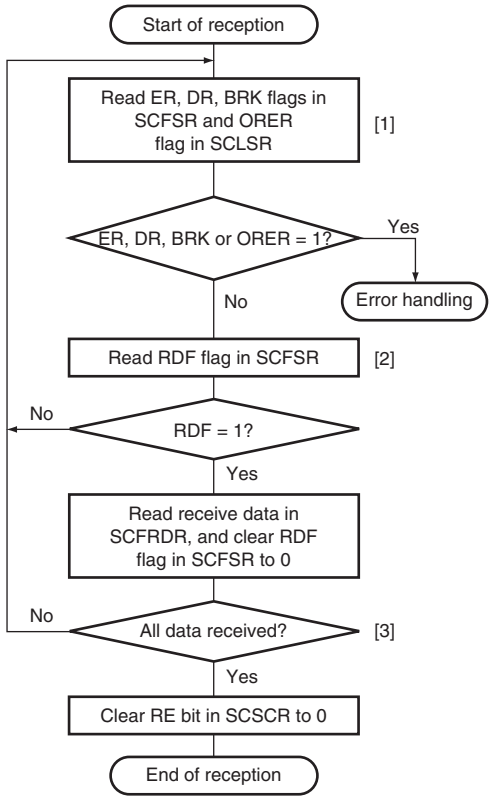


Figure 16.6 Example of Operation Using Modem Control ($\overline{\text{CTS}}$)

Receiving Serial Data (Asynchronous Mode):

Figures 16.7 and 16.8 show a sample flowchart for serial reception.

Use the following procedure for serial data reception after enabling the SCIF for reception.



[1] Receive error handling and break detection:

Read the DR, ER, and BRK flags in SCFSR, and the ORER flag in SCLSR, to identify any error, perform the appropriate error handling, then clear the DR, ER, BRK, and ORER flags to 0. In the case of a framing error, a break can also be detected by reading the value of the RXD pin.

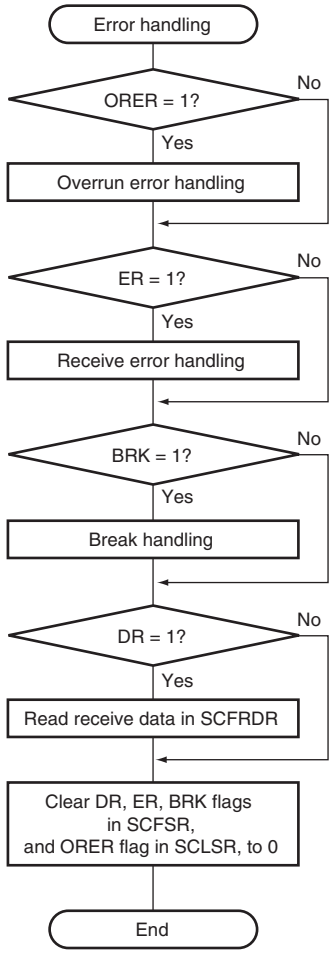
[2] SCIF status check and receive data read:

Read SCFSR and check that RDF = 1, then read the receive data in SCFRDR, read 1 from the RDF flag, and then clear the RDF flag to 0. The transition of the RDF flag from 0 to 1 can also be identified by an RXIF interrupt.

[3] Serial reception continuation procedure:

To continue serial reception, read at least the receive trigger number of receive data bytes from SCFRDR, read 1 from the RDF flag, then clear the RDF flag to 0. The number of receive data bytes in SCFRDR can be ascertained by reading from the lower 8 bits of SCFDR.

Figure 16.7 Sample Flowchart for Receiving Serial Data



[1] Whether a framing error or parity error has occurred in the receive data that is to be read from SCFRDR can be ascertained from the FER and PER bits in SCFSR.

[2] When a break signal is received, receive data is not transferred to SCFRDR while the BRK flag is set. However, note that the last data in SCFRDR is H'00, and the break data in which a framing error occurred is stored.

Figure 16.8 Sample Flowchart for Receiving Serial Data (cont)

In serial reception, the SCIF operates as described below.

1. The SCIF monitors the transmission line, and if a 0 start bit is detected, performs internal synchronization and starts reception.
2. The received data is stored in SCRSR in LSB-to-MSB order.
3. The parity bit and stop bit are received.
   After receiving these bits, the SCIF carries out the following checks.
   A. Stop bit check: The SCIF checks whether the stop bit is 1. If there are two stop bits, only the first is checked.
   B. The SCIF checks whether receive data can be transferred from the receive shift register (SCRSR) to SCFRDR.
   C. Overrun check: The SCIF checks that the ORER flag is 0, indicating that the overrun error has not occurred.
   D. Break check: The SCIF checks that the BRK flag is 0, indicating that the break state is not set.

   If all the above checks are passed, the receive data is stored in SCFRDR.

   Note: When a parity error or a framing error occurs, reception is not suspended.
4. If the RIE bit in SCSCR is set to 1 when the RDF or DR flag changes to 1, a receive-FIFO-data-full interrupt (RXIF) request is generated. If the RIE bit or the REIE bit in SCSCR is set to 1 when the ER flag changes to 1, a receive-error interrupt (ERIF) request is generated. If the RIE bit or the REIE bit in SCSCR is set to 1 when the BRK or ORER flag changes to 1, a break reception interrupt (BRIF) request is generated.

   Figure 16.9 shows an example of the operation for reception.

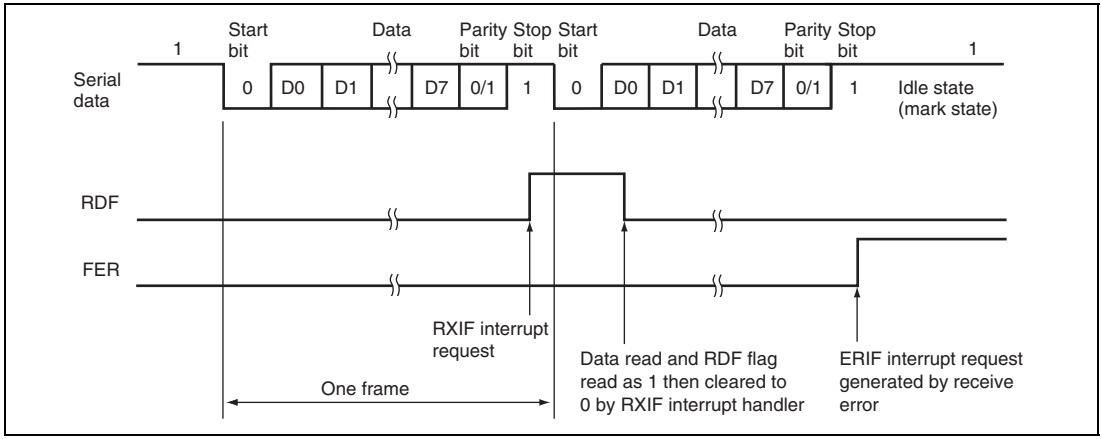


Figure 16.9 Example of SCIF Receive Operation (8-Bit Data, Parity, One Stop Bit)

5. When modem control is enabled, the $\overline{\text{RTS}}$ signal is output depending on the empty status of SCFRDR. When $\overline{\text{RTS}}$ is 0, reception is possible. When $\overline{\text{RTS}}$ is 1, this indicates that the SCFRDR is full and no extra data can be received.

   Figure 16.10 shows an example of the operation when modem control is used.

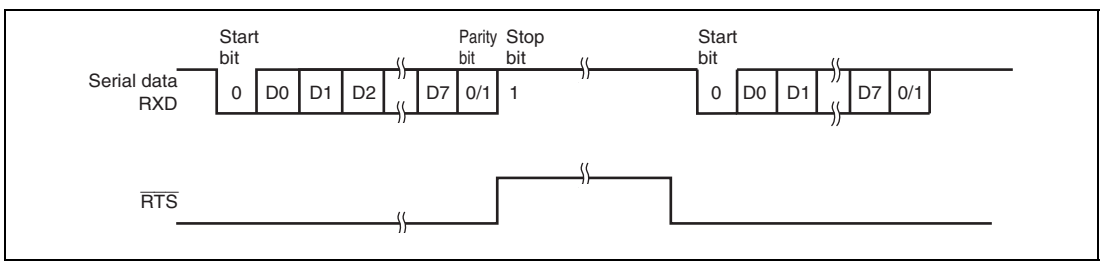


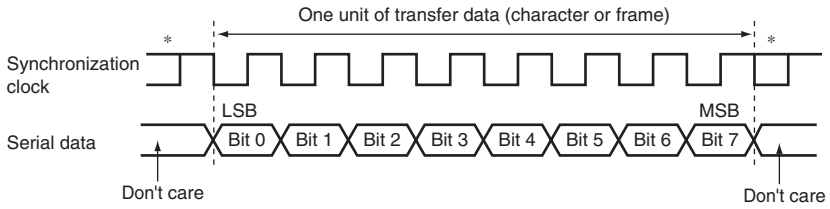
Figure 16.10 Example of Operation Using Modem Control ($\overline{\text{RTS}}$)

16.4.3 Clock Synchronous Mode

In clock synchronous mode, the SCIF transmits and receives data in synchronization with clock pulses. This mode is suitable for high-speed serial communication.

The SCIF transmitter and receiver are independent, so full-duplex communication is possible while sharing the same clock. The transmitter and receiver are also 16-byte FIFO buffered, so continuous transmitting or receiving is possible by reading or writing data while transmitting or receiving is in progress.

Figure 16.11 shows the general format in clock synchronous serial communication.



Note: * High except in continuous transfer

Figure 16.11 Data Format in Clock Synchronous Communication

In clock synchronous serial communication, each data bit is output on the communication line from one falling edge of the serial clock to the next. Data is guaranteed valid at the rising edge of the serial clock. In each character, the serial data bits are transmitted in order from the LSB (first) to the MSB (last). After output of the MSB, the communication line remains in the state of the MSB. In clock synchronous mode, the SCIF receives data in synchronization with the rising edge of the serial clock.

(1) Communication Format

The data length is fixed at eight bits. No parity bit can be added.

(2) Clock

An internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCIF transmit/receive clock.

When the SCIF operates on an internal clock, it outputs the clock signal at the SCK pin. Eight clock pulses are output per transmitted or received character. When the SCIF is not transmitting or receiving, the clock signal remains in the high state. When only receiving, the clock signal outputs while the RE bit of SCSCR is 1 and the number of data in receive FIFO is more than the receive FIFO data trigger number. In this case, $8 \times (16 + 1) = 136$ pulses of synchronous clock are output. To perform reception of n characters of data, select an external clock as the clock source. If an internal clock should be used, set RE = 1 and TE = 1 and receive n characters of data simultaneously with the transmission of n characters of dummy data.

(3) Transmitting and Receiving Data

SCIF Initialization (Clock Synchronous Mode): Before transmitting, receiving, or changing the mode or communication format, the software must clear the TE and RE bits to 0 in the serial control register (SCSCR), then initialize the SCIF. Clearing TE to 0 initializes the transmit shift register (SCTSR). Clearing RE to 0, however, does not initialize the RDF, PER, FER, and ORER flags and receive data register (SCRDR), which retain their previous contents.

Figure 16.12 shows a sample flowchart for initializing the SCIF.

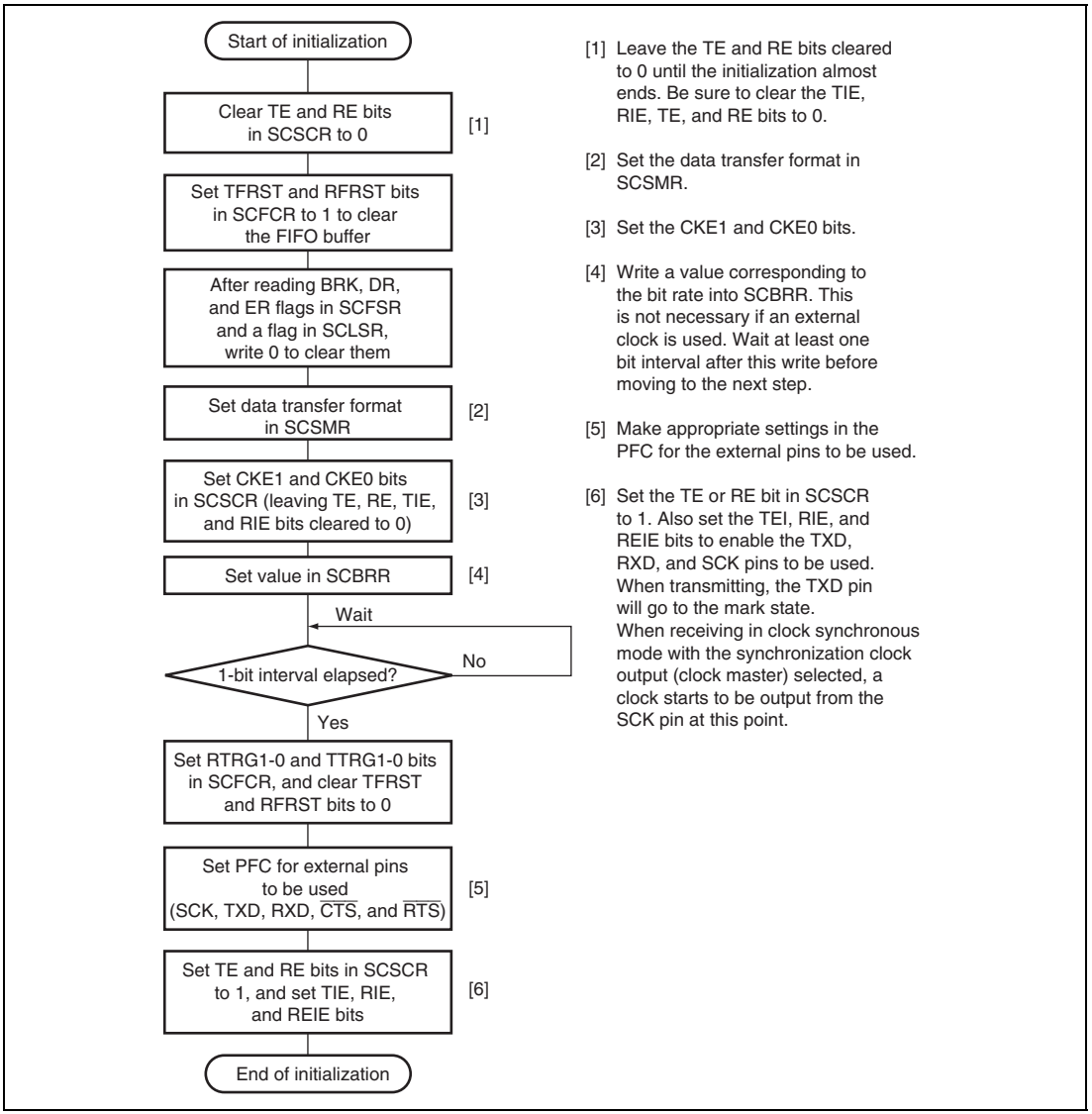


Figure 16.12 Sample Flowchart for SCIF Initialization

Transmitting Serial Data (Clock Synchronous Mode): Figure 16.13 shows a sample flowchart for transmitting serial data.

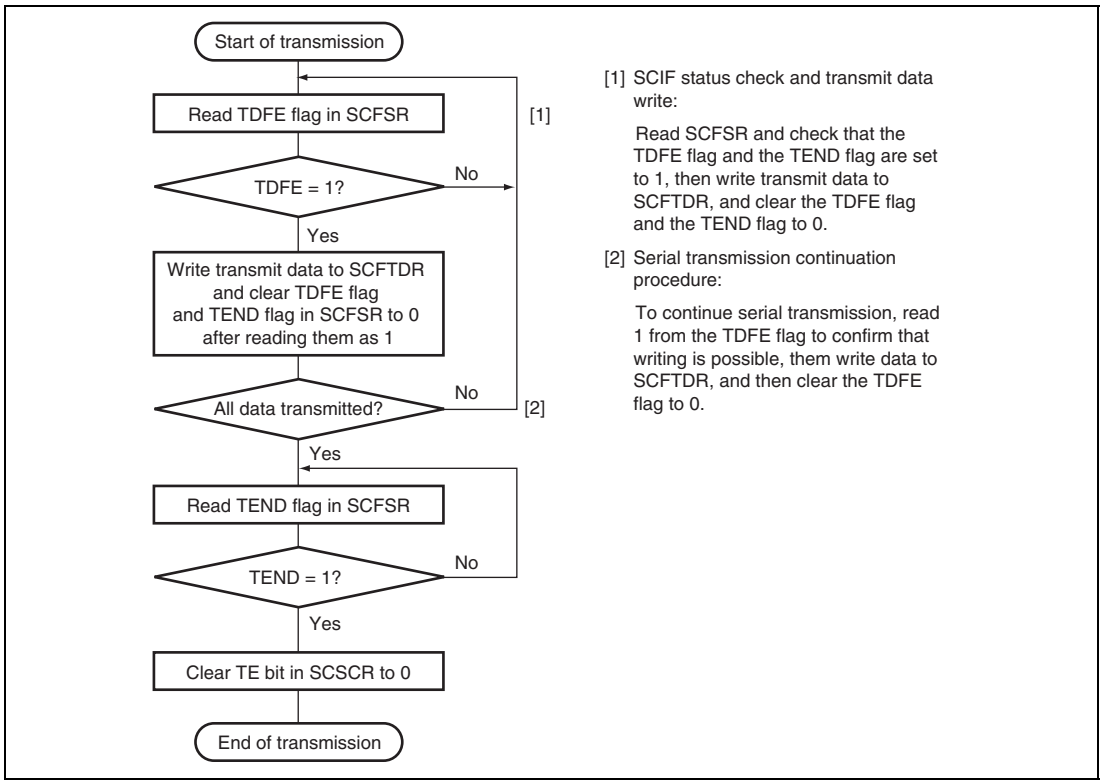


Figure 16.13 Sample Flowchart for Transmitting Serial Data

In transmitting serial data, the SCIF operates as follows:

1. When data is written into the transmit FIFO data register (SCFTDR), the SCIF transfers the data from SCFTDR to the transmit shift register (SCTSR) and starts transmitting. Confirm that the TDFE flag in the serial status register (SCFSR) is set to 1 before writing transmit data to SCFTDR. The number of data bytes that can be written is (16 – transmit trigger setting).
2. When data is transferred from SCFTDR to SCTSR and transmission is started, consecutive transmit operations are performed until there is no transmit data left in SCFTDR. When the number of transmit data bytes in SCFTDR falls below the transmit trigger number set in the FIFO control register (SCFCR), the TDFE flag is set. If the TIE bit in the serial control register (SCSR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXIF) request is generated.

   If clock output mode is selected, the SCIF outputs eight synchronous clock pulses. If an external clock source is selected, the SCIF outputs data in synchronization with the input clock. Data is output from the TXD pin in order from the LSB (bit 0) to the MSB (bit 7).
3. The SCIF checks the SCFTDR transmit data at the timing for sending the MSB (bit 7). If data is present, the data is transferred from SCFTDR to SCTSR, the MSB (bit 7) is sent, and then serial transmission of the next frame is started. If there is no transmit data, the TEND flag in SCFSR is set to 1, the MSB (bit 7) is sent, and then the TXD pin holds the states.
4. After the end of serial transmission, the SCK pin is held in the high state.

Figure 16.14 shows an example of SCIF transmit operation.

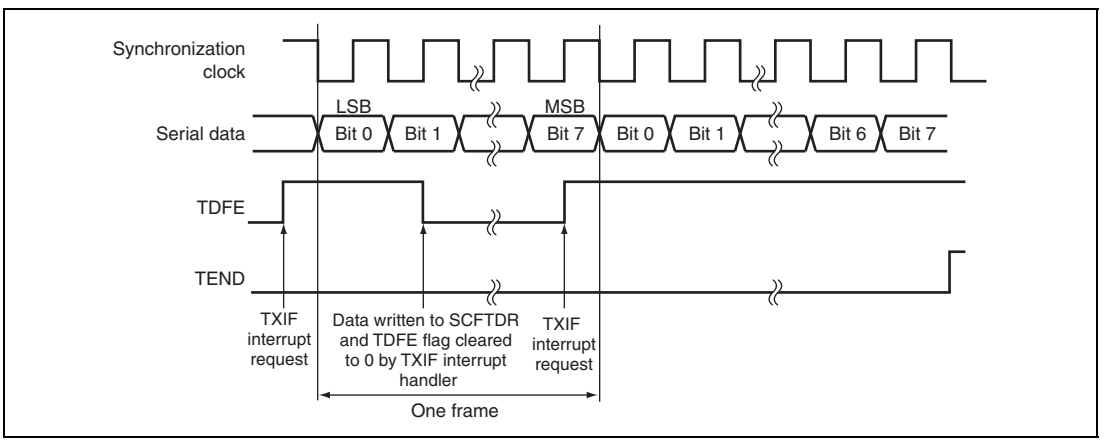


Figure 16.14 Example of SCIF Transmit Operation

Receiving Serial Data (Clock Synchronous Mode): Figures 16.15 and 16.16 show a sample flowchart for receiving serial data. When switching from asynchronous mode to clock synchronous mode without SCIF initialization, make sure that ORER, PER, and FER are cleared to 0.

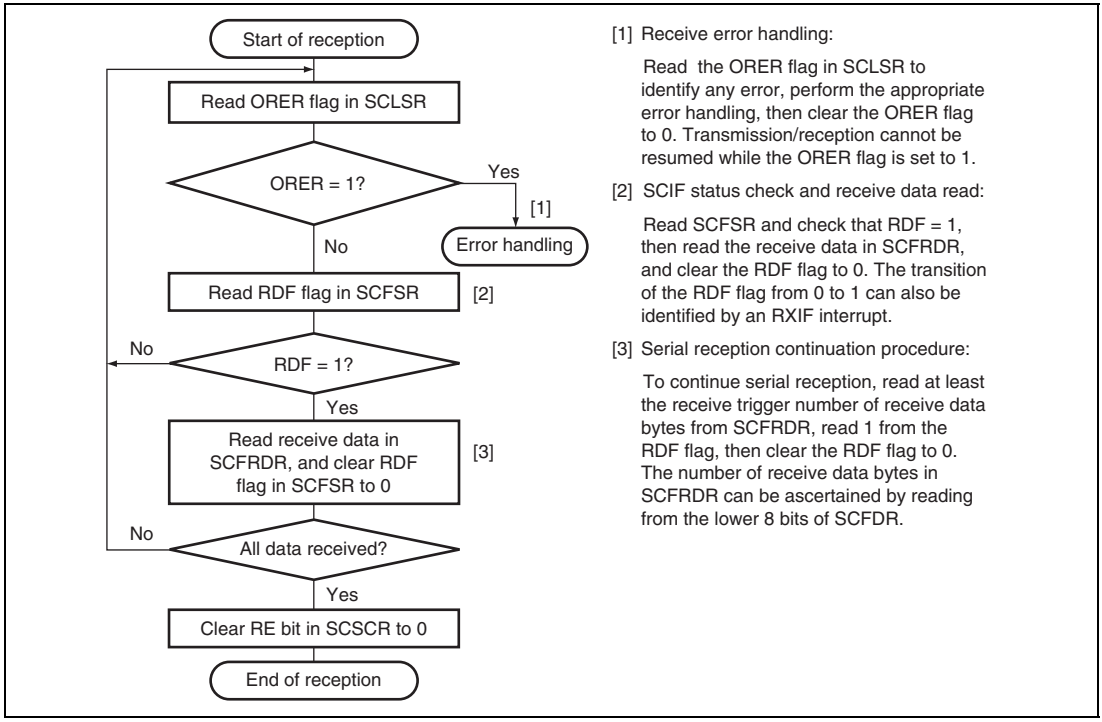


Figure 16.15 Sample Flowchart for Receiving Serial Data (1)

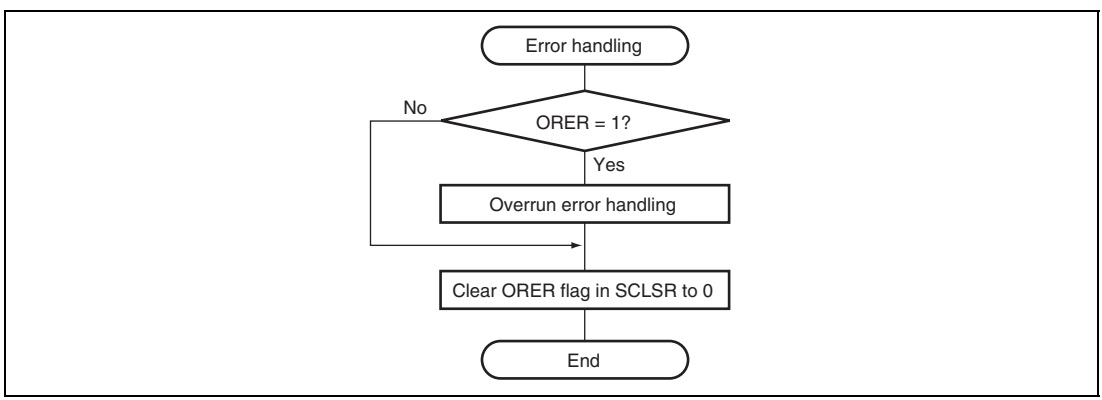


Figure 16.16 Sample Flowchart for Receiving Serial Data (2)

In receiving, the SCIF operates as follows:

1. The SCIF synchronizes with serial clock input or output and initializes internally.
2. Receive data is shifted into SCRSR in order from the LSB to the MSB. After receiving the data, the SCIF checks the receive data can be loaded from SCRSR into SCFRDR or not. If this check is passed, the SCIF stores the received data in SCFRDR. If the check is not passed (overrun error is detected), further reception is prevented.
3. After setting RDF to 1, if the receive-data-full interrupt enable bit (RIE) is set to 1 in SCSCR, the SCIF requests a receive-data-full interrupt (RXIF). If the ORER bit is set to 1 and the RIE bit or REIE bit in SCSCR is also set to 1, the SCIF requests a break interrupt (BRIF).

Figure 16.17 shows an example of SCIF receive operation.

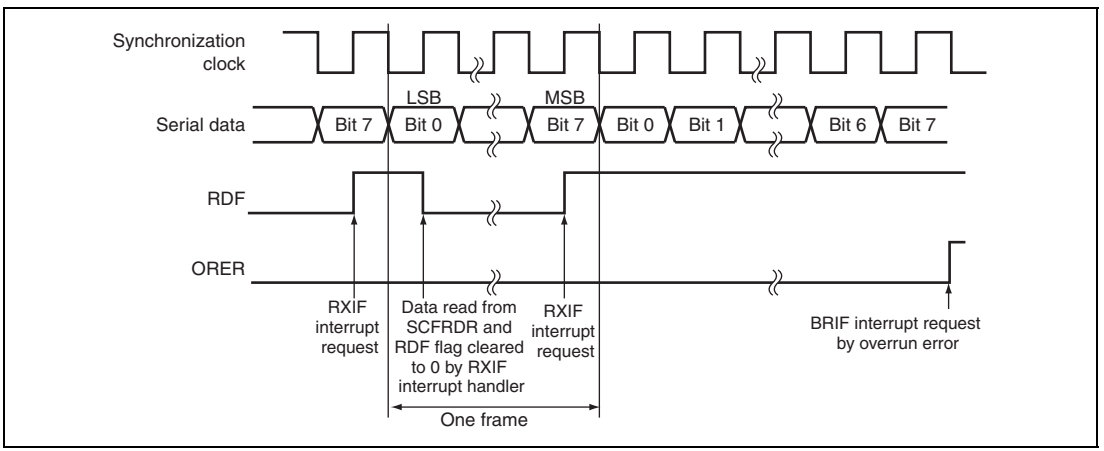
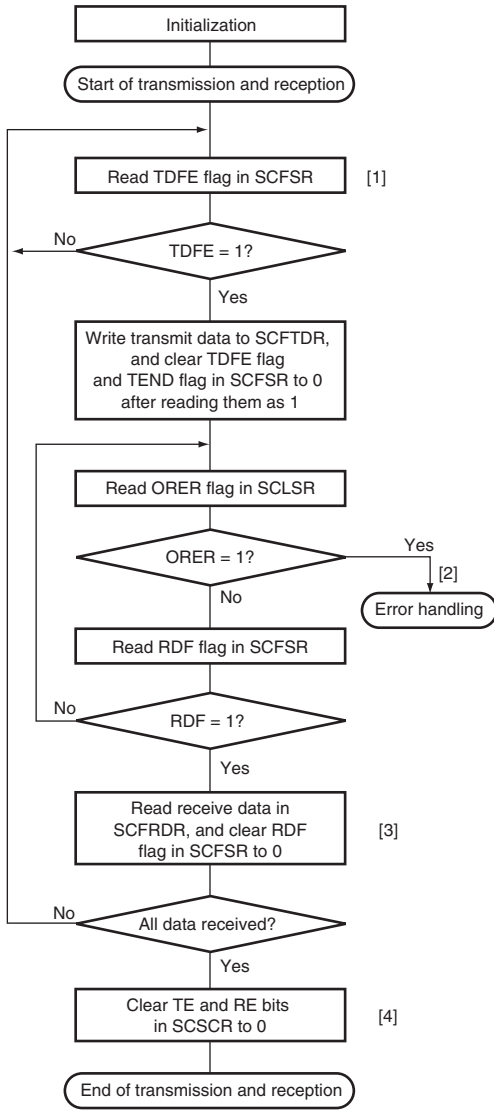


Figure 16.17 Example of SCIF Receive Operation

Transmitting and Receiving Serial Data Simultaneously (Clock Synchronous Mode): Figure 16.18 shows a sample flowchart for transmitting and receiving serial data simultaneously.



[1] SCIF status check and transmit data write:

Read SCFSR and check that the TDFE flag and the TEND flag are set to 1, then write transmit data to SCFTDR, and clear the TDFE flag and the TEND flag to 0. The transition of the TDFE flag from 0 to 1 can also be identified by a TXIF interrupt.

[2] Receive error handling:

Read the ORER flag in SCLSR to identify any error, perform the appropriate error handling, then clear the ORER flag to 0. Transmission/reception cannot be resumed while the ORER flag is set to 1.

[3] SCIF status check and receive data read:

Read SCFSR and check that RDF = 1, then read the receive data in SCFRDR, and clear the RDF flag to 0. The transition of the RDF flag from 0 to 1 can also be identified by an RXIF interrupt.

[4] Serial transmission and reception continuation procedure:

To continue serial transmission and reception, read 1 from the RDF flag and the receive data in SCFRDR, and clear the RDF flag to 0 before receiving the last bit in the current frame. Similarly, read 1 from the TDFE flag to confirm that writing is possible before transmitting the MSB in the current frame. Then write data to SCFTDR and clear the TDFE flag to 0.

Note: When switching from a transmit operation or receive operation to simultaneous transmission and reception operations, clear the TE and RE bits to 0, and then set them simultaneously to 1.

Figure 16.18 Sample Flowchart for Transmitting/Receiving Serial Data

16.5 SCIF Interrupt Sources and DTC

The SCIF has four interrupt sources: transmit-FIFO-data-empty (TXIF), receive-error (ERIF), receive-data-full (RXIF), and break (BRIF).

Table 16.16 shows the interrupt sources and their order of priority. The interrupt sources are enabled or disabled by means of the TIE, RIE, and REIE bits in SCSCR. A separate interrupt request is sent to the interrupt controller for each of these interrupt sources.

When TXIF request is enabled by TIE bit and the TDFE flag in the serial status register (SCFSR) is set to 1, a TXIF interrupt request is generated.

When RXIF request is enabled by RIE bit and the RDF or DR flag in SCFSR is set to 1, an RXIF interrupt request is generated. The RXIF interrupt request caused by DR flag is generated only in asynchronous mode.

When BRIF request is enabled by RIE bit or REIE bit and the BRK flag in SCFSR or ORER flag in SCLSR is set to 1, a BRIF interrupt request is generated.

When ERIF request is enabled by RIE bit or REIE bit and the ER flag in SCFCR is set to 1, an ERIF interrupt request is generated.

When the RIE bit is set to 0 and the REIE bit is set to 1, SCIF request ERIF interrupt and BRIF interrupt without requesting RXIF interrupt.

The TXIF interrupt indicates that transmit data can be written, and the RXIF interrupt indicates that there is receive data in SCFRDR.

Table 16.16 SCIF Interrupt Sources

Interrupt Source	Description	Interrupt Enable Bit	DTC Activation
ERIF	Interrupt initiated by receive error (ER)	RIE or REIE	—
RXIF	Interrupt initiated by receive data FIFO full (RDF) or data ready (DR)	RIE	√
BRIF	Interrupt initiated by break (BRK) or overrun error (ORER)	RIE or REIE	—
TXIF	Interrupt initiated by transmit FIFO data empty (TDFE)	TIE	√

16.6 Serial Port Register (SCSPTR) and SCIF Pins

The relationship between SCSPTR and the SCIF pins is shown in figures 16.19 to 16.22.

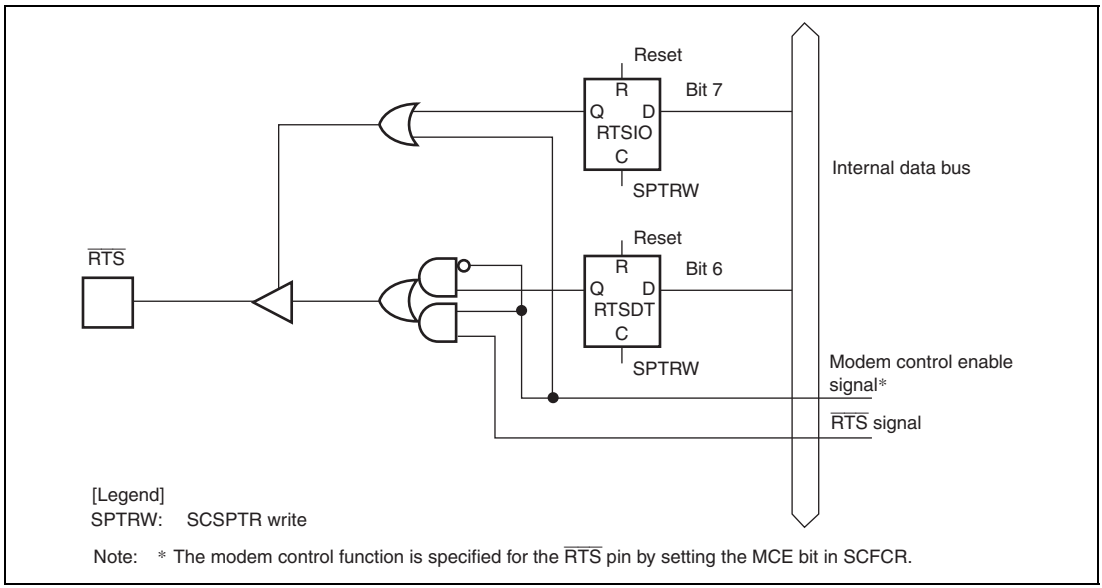
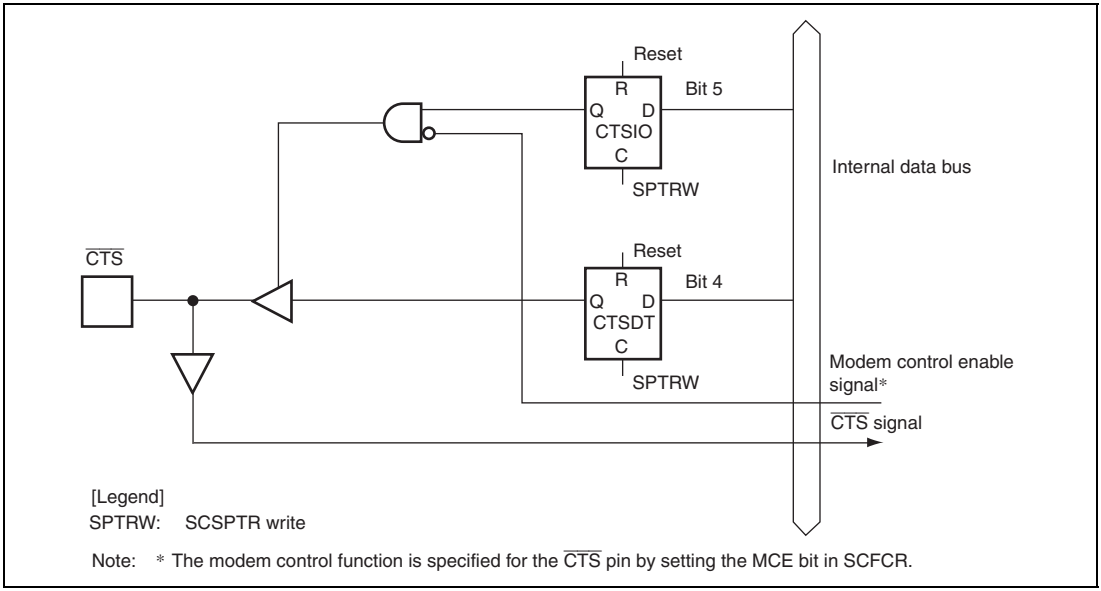
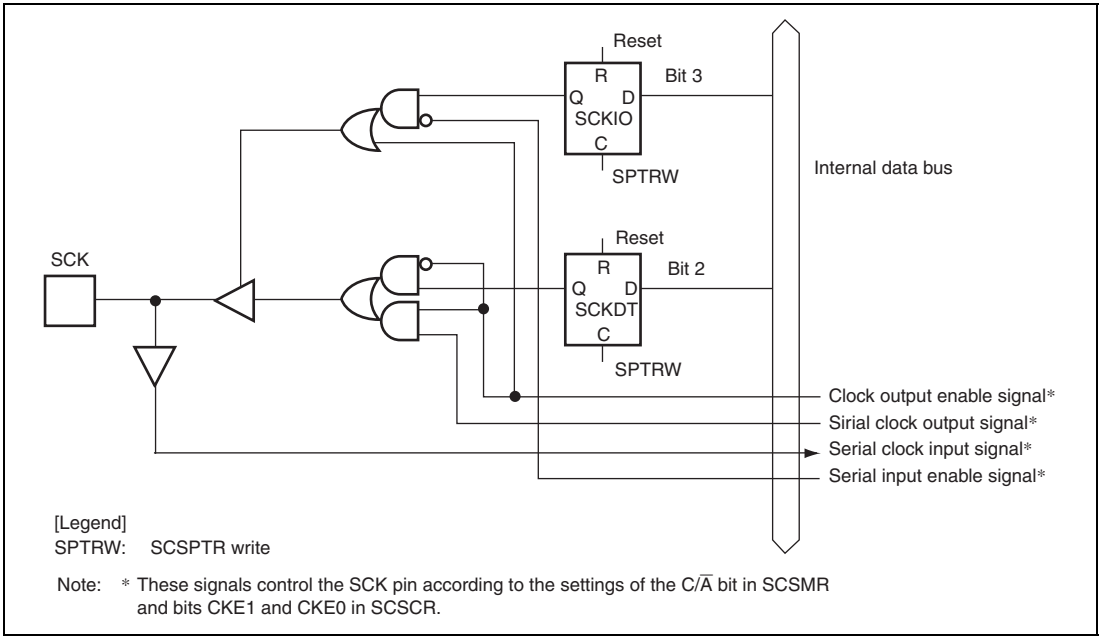


Figure 16.19 RTSIO Bit, RTSDT Bit, and $\overline{\text{RTS}}$ Pin



Note: * The modem control function is specified for the $\overline{\text{CTS}}$ pin by setting the MCE bit in SCFCR.

Figure 16.20 CTSIO Bit, CTSDT bit, and $\overline{\text{CTS}}$ Pin



Note: * These signals control the SCK pin according to the settings of the C/$\overline{\text{A}}$ bit in SCSMR and bits CKE1 and CKE0 in SCSCR.

Figure 16.21 SCKIO Bit, SCKDT bit, and SCK Pin

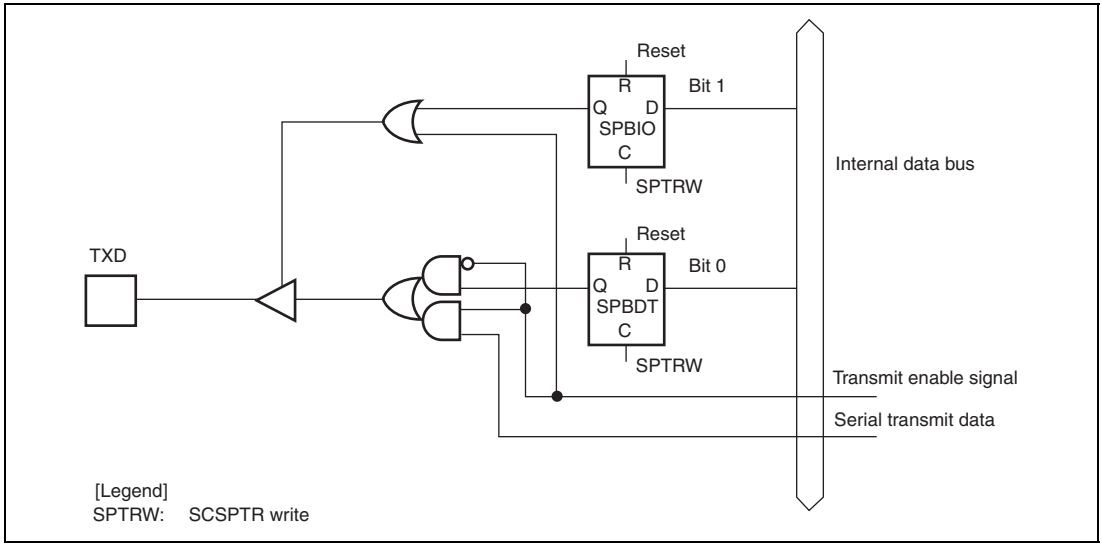


Figure 16.22 SPBIO Bit, SPBDT bit, and TXD Pin

16.7 Usage Notes

Note the following when using the SCIF.

16.7.1 SCFTDR Writing and TDFE Flag

The TDFE flag in the serial status register (SCFSR) is set when the number of transmit data bytes written in the transmit FIFO data register (SCFTDR) has fallen below the transmit trigger number set by bits TTRG1 and TTRG0 in the FIFO control register (SCFCR). After TDFE is set, transmit data up to the number of empty bytes in SCFTDR can be written, allowing efficient continuous transmission.

However, if the number of data bytes written in SCFTDR is equal to or less than the transmit trigger number, the TDFE flag will be set to 1 again after being read as 1 and cleared to 0. TDFE clearing should therefore be carried out when SCFTDR contains more than the transmit trigger number of transmit data bytes.

The number of transmit data bytes in SCFTDR can be found from the upper 8 bits of the FIFO data count register (SCFDR).

16.7.2 SCFRDR Reading and RDF Flag

The RDF flag in the serial status register (SCFSR) is set when the number of receive data bytes in the receive FIFO data register (SCFRDR) has become equal to or greater than the receive trigger number set by bits RTRG1 and RTRG0 in the FIFO control register (SCFCR). After RDF is set, receive data equivalent to the trigger number can be read from SCFRDR, allowing efficient continuous reception.

However, if the number of data bytes in SCFRDR is equal to or greater than the trigger number, the RDF flag will be set to 1 again if it is cleared to 0. RDF should therefore be cleared to 0 after being read as 1 after reading the number of the received data in the receive FIFO data register (SCFRDR) which is less than the trigger number.

The number of receive data bytes in SCFRDR can be found from the lower 8 bits of the FIFO data count register (SCFDR).

16.7.3 Break Detection and Processing

When data containing a framing error is received and then space 0 (low level) is input for more than one frame length, a break (BRK) is detected. When a break is detected, not only the transfer of receive data (H'00) to SCFRDR but also the setting in SCRSR of serial data input on the RXD pin is stopped. If the RIE or REIE bit in SCSCR is set to 1, a break interrupt request (BRI) is issued. Reception resumes when the break ends and the receive signal is mark 1 (high level).

It is also possible to perform break detection by reading the value of the RXD pin directly when a framing error (FER) is detected. Use the port register to read the value of the RXD pin. In the break state the input from the RXD pin consists of all 0s, so the FER flag is set and the parity error flag (PER) may also be set.

16.7.4 Sending a Break Signal

The I/O condition and level of the TXD pin are determined by the SPBIO and SPBDT bits in the serial port register (SCSPTR). This feature can be used to send a break signal.

Until TE bit is set to 1 (enabling transmission) after initializing, TXD pin does not work. During the period, mark status is performed by SPBDT bit. Therefore, the SPBIO and SPBDT bits should be set to 1 (high level output).

To send a break signal during serial transmission, clear the SPBDT bit to 0 (designating low level), then clear the TE bit to 0 (halting transmission). When the TE bit is cleared to 0, the transmitter is initialized regardless of the current transmission state, and 0 is output from the TXD pin.

16.7.5 Receive Data Sampling Timing and Receive Margin (Asynchronous Mode)

The SCIF operates on a base clock with a frequency of 16 times the transfer rate. In reception, the SCIF synchronizes internally with the fall of the start bit, which it samples on the base clock. Receive data is latched at the rising edge of the eighth base clock pulse. The timing is shown in figure 16.23.

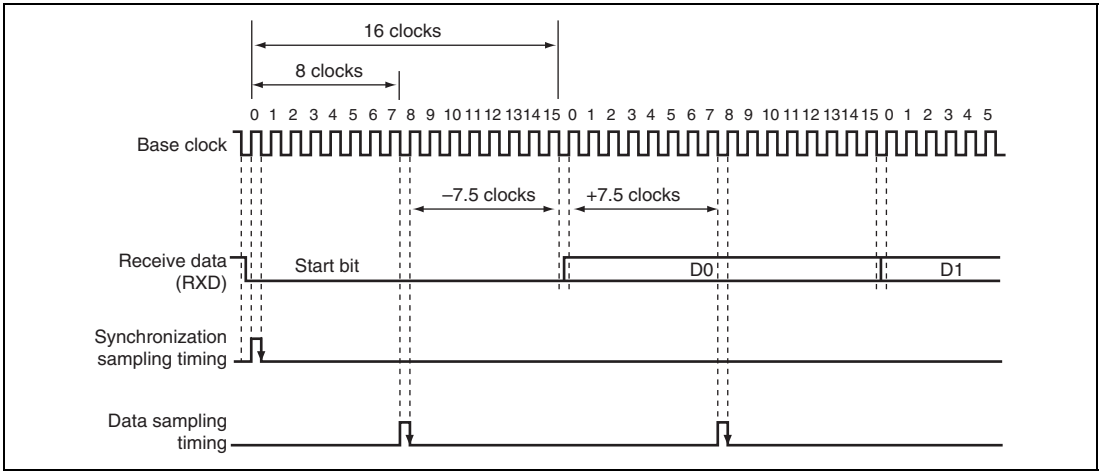


Figure 16.23 Receive Data Sampling Timing in Asynchronous Mode

The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

Equation 1:

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1+F) \right| \times 100\ \%$$

Where: M: Receive margin (%)
N: Ratio of clock frequency to bit rate (N = 16)
D: Clock duty (D = 0 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute deviation of clock frequency

From equation 1, if F = 0 and D = 0.5, the receive margin is 46.875%, as given by equation 2.

Equation 2:

When D = 0.5 and F = 0:

$M = (0.5 - 1/(2 \times 16)) \times 100\%$
$= 46.875\%$

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

16.7.6 Module Standby Mode Setting

The SCIF operation can be disabled or enabled using the standby control register. The initial setting is for SCIF operation to be halted. Access to registers is enabled by clearing module standby mode. For details, refer to section 26, Power-Down Modes.

16.7.7 Note on Using DTC

When data is written to SCFTDR by activating the DTC through a TXIF interrupt, the TEND flag value is undefined. In this case, do not use the TEND flag as a transmit end flag.

16.7.8 FER Flag and PER Flag of Serial Status Register (SCFSR)

The FER flag and PER flag in the serial status register (SCFSR) are status flags that apply to next entry to be read from the receive FIFO data register (SCFRDR). After the CPU or DTC reads the receive FIFO data register, the flags of framing errors and parity errors will disappear.

To check the received data for the states of framing errors and parity errors, only read the receive FIFO register after reading the serial status register.

Section 17 Synchronous Serial Communication Unit (SSU)

This LSI has an independent synchronous serial communication unit (SSU) channel. The SSU has master mode in which this LSI outputs clocks as a master device for synchronous serial communication and slave mode in which clocks are input from an external device for synchronous serial communication. Synchronous serial communication can be performed with devices having different clock polarity and clock phase.

17.1 Features

- Choice of SSU mode and clock synchronous mode
- Choice of master mode and slave mode
- Choice of standard mode and bidirectional mode
- Synchronous serial communication with devices with different clock polarity and clock phase
- Choice of 8/16/32-bit width of transmit/receive data
- Full-duplex communication capability

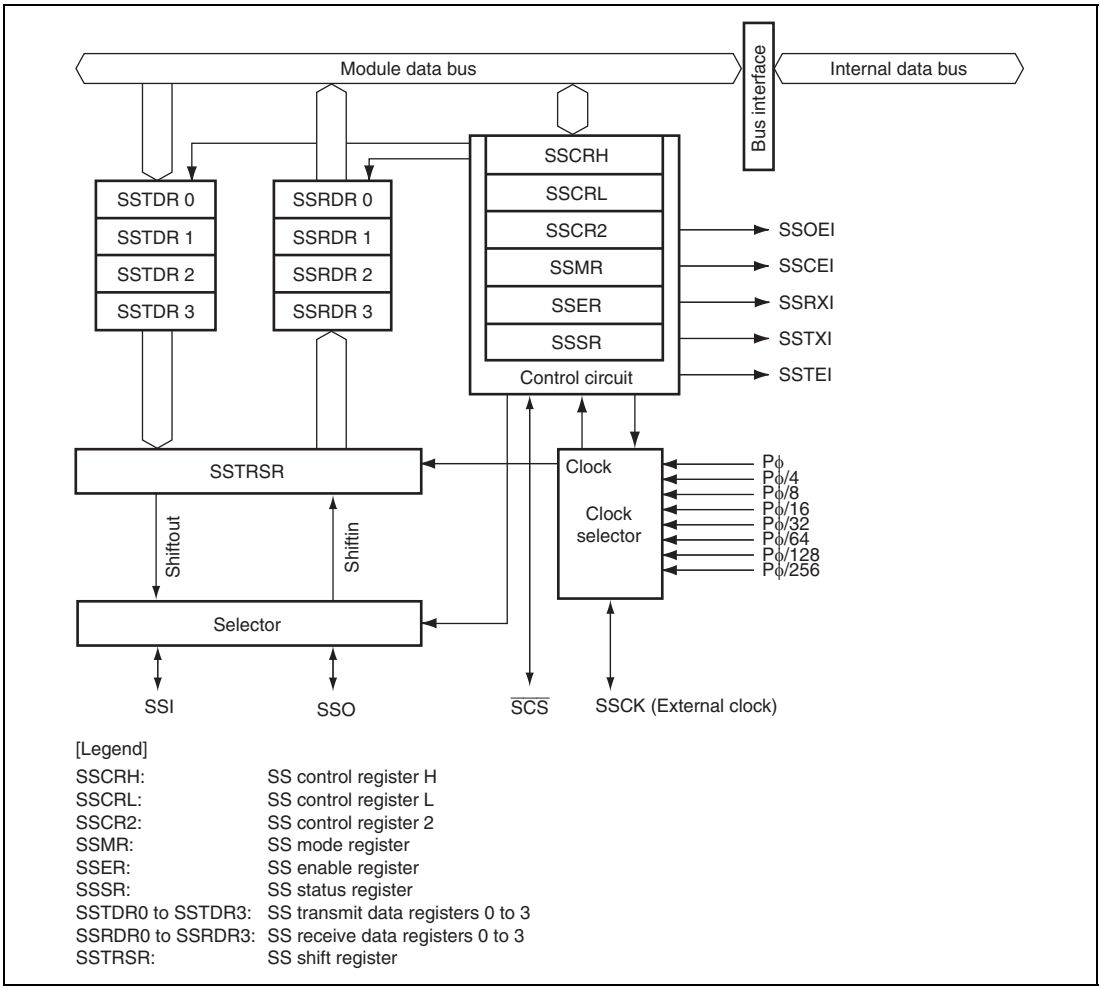
  The shift register is incorporated, enabling transmission and reception to be executed simultaneously.
- Consecutive serial communication
- Choice of LSB-first or MSB-first transfer
- Choice of a clock source

  Pϕ/4, Pϕ/8, Pϕ/16, Pϕ/32, Pϕ/64, Pϕ/128, Pϕ/256, or an external clock
- Five interrupt sources

  Transmit end, transmit data register empty, receive data full, overrun error, and conflict error.

  The data transfer controller (DTC) can be activated by a transmit data register empty request or a receive data full request to transfer data.
- Module standby mode can be set

Figure 17.1 shows a block diagram of the SSU.



[Legend]

SSCRH:	SS control register H
SSCRL:	SS control register L
SSCR2:	SS control register 2
SSMR:	SS mode register
SSER:	SS enable register
SSSR:	SS status register
SSTDR0 to SSTDR3:	SS transmit data registers 0 to 3
SSRDR0 to SSRDR3:	SS receive data registers 0 to 3
SSTRSR:	SS shift register

Figure 17.1 Block Diagram of SSU

17.2 Input/Output Pins

Table 17.1 shows the SSU pin configuration.

Table 17.1 Pin Configuration

Symbol	I/O	Function
SSCK	I/O	SSU clock input/output
SSI	I/O	SSU data input/output
SSO	I/O	SSU data input/output
$\overline{\text{SCS}}$	I/O	SSU chip select input/output

17.3 Register Descriptions

The SSU has the following registers. For details on the addresses of these registers and the states of these registers in each processing state, see section 27, List of Registers.

Table 17.2 Register Configuration

Register Name	Abbreviation	R/W	Initial value	Address	Access Size
SS control register H	SSCRH	R/W	H'0D	H'FFFFCD00	8, 16
SS control register L	SSCRL	R/W	H'00	H'FFFFCD01	8
SS mode register	SSMR	R/W	H'00	H'FFFFCD02	8, 16
SS enable register	SSER	R/W	H'00	H'FFFFCD03	8
SS status register	SSSR	R/W	H'04	H'FFFFCD04	8, 16
SS control register 2	SSCR2	R/W	H'00	H'FFFFCD05	8
SS transmit data register 0	SSTDR0	R/W	H'00	H'FFFFCD06	8, 16
SS transmit data register 1	SSTDR1	R/W	H'00	H'FFFFCD07	8
SS transmit data register 2	SSTDR2	R/W	H'00	H'FFFFCD08	8, 16
SS transmit data register 3	SSTDR3	R/W	H'00	H'FFFFCD09	8
SS receive data register 0	SSRDR0	R	H'00	H'FFFFCD0A	8, 16
SS receive data register 1	SSRDR1	R	H'00	H'FFFFCD0B	8
SS receive data register 2	SSRDR2	R	H'00	H'FFFFCD0C	8, 16
SS receive data register 3	SSRDR3	R	H'00	H'FFFFCD0D	8

17.3.1 SS Control Register H (SSCRH)

SSCRH specifies master/slave device selection, bidirectional mode enable, SSO pin output value selection, SSCK pin selection, and $\overline{SCS}$ pin selection.

Bit:	7	6	5	4	3	2	1	0
	MSS	BIDE	-	SOL	SOLP	-	CSS[1:0]	
Initial value:	0	0	0	0	1	1	0	1
R/W:	R/W	R/W	R	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSS	0	R/W	Master/Slave Device Select Selects that this module is used in master mode or slave mode. When master mode is selected, transfer clocks are output from the SSCK pin. When the CE bit in SSSR is set, this bit is automatically cleared. 0: Slave mode is selected. 1: Master mode is selected.
6	BIDE	0	R/W	Bidirectional Mode Enable Selects that both serial data input pin and output pin are used or one of them is used. However, transmission and reception are not performed simultaneously when bidirectional mode is selected. For details, section 17.4.3, Relationship between Data Input/Output Pins and Shift Register. 0: Standard mode (two pins are used for data input and output) 1: Bidirectional mode (one pin is used for data input and output)
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	SOL	0	R/W	Serial Data Output Value Select The serial data output retains its level of the last bit after completion of transmission. The output level before or after transmission can be specified by setting this bit. When specifying the output level, use the MOV instruction after clearing the SOLP bit to 0. Since writing to this bit during data transmission causes malfunctions, this bit should not be changed. 0: Serial data output is changed to low. 1: Serial data output is changed to high.
3	SOLP	1	R/W	SOL Bit Write Protect When changing the output level of serial data, set the SOL bit to 1 or clear the SOL bit to 0 after clearing the SOLP bit to 0 using the MOV instruction. 0: Output level can be changed by the SOL bit 1: Output level cannot be changed by the SOL bit. This bit is always read as 1.
2	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
1, 0	CSS[1:0]	01	R/W	$\overline{\text{SCS}}$ Pin Select Select that the $\overline{\text{SCS}}$ pin functions as $\overline{\text{SCS}}$ input or output. 00: Setting prohibited 01: Setting prohibited 10: Function as $\overline{\text{SCS}}$ automatic input/output (function as $\overline{\text{SCS}}$ input before and after transfer and output a low level during transfer) 11: Function as $\overline{\text{SCS}}$ automatic output (outputs a high level before and after transfer and outputs a low level during transfer)

17.3.2 SS Control Register L (SSCRL)

SSCRL selects operating mode, software reset, and transmit/receive data length.

Bit:	7	6	5	4	3	2	1	0
	FCLRM	SSUMS	SRES	-	-	-	DATS[1:0]	
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	FCLRM	0	R/W	Flag Clear Mode Selects whether the SSRXI and SSTXI interrupt flags are cleared on writing to SSTDR or reading from SSRDR or on completion of DTC transfer. When using the DTC, set this bit to 0. 0: Flags are cleared when DTC transfer is completed (except when transfer counter value is H'0000) 1: Flags are cleared on SSTDR or SSRDR access
6	SSUMS	0	R/W	Selects transfer mode from SSU mode and clock synchronous mode. 0: SSU mode 1: Clock synchronous mode
5	SRES	0	R/W	Software Reset Setting this bit to 1 forcibly resets the SSU internal sequencer. After that, this bit is automatically cleared. The ORER, TEND, TDRE, RDRF, and CE bits in SSSR and the TE and RE bits in SSER are also initialized. Values of other bits for SSU registers are held. To stop transfer, set this bit to 1 to reset the SSU internal sequencer.
4 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	DATS[1:0]	00	R/W	Transmit/Receive Data Length Select Select serial data length. 00: 8 bits 01: 16 bits 10: 32 bits 11: Setting prohibited

17.3.3 SS Mode Register (SSMR)

SSMR selects the MSB first/LSB first, clock polarity, clock phase, and clock rate of synchronous serial communication.

Bit:	7	6	5	4	3	2	1	0
	MLS	CPOS	CPHS	-	-	CKS[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MLS	0	R/W	MSB First/LSB First Select Selects that the serial data is transmitted in MSB first or LSB first. 0: LSB first 1: MSB first
6	CPOS	0	R/W	Clock Polarity Select Selects the SSCK clock polarity. 0: High output in idle mode, and low output in active mode 1: Low output in idle mode, and high output in active mode
5	CPHS	0	R/W	Clock Phase Select (Only for SSU Mode) Selects the SSCK clock phase. 0: Data changes at the first edge. 1: Data is latched at the first edge.

Bit	Bit Name	Initial Value	R/W	Description
4, 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	CKS[2:0]	000	R/W	Transfer Clock Rate Select Select the transfer clock rate (prescaler division rate) when an internal clock is selected. 000: Reserved 001: Pϕ/4 010: Pϕ/8 011: Pϕ/16 100: Pϕ/32 101: Pϕ/64 110: Pϕ/128 111: Pϕ/256

17.3.4 SS Enable Register (SSER)

SSER performs transfer/receive control of synchronous serial communication and setting of interrupt enable.

Bit:	7	6	5	4	3	2	1	0
	TE	RE	-	-	TEIE	TIE	RIE	CEIE
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TE	0	R/W	Transmit Enable When this bit is set to 1, transmission is enabled.
6	RE	0	R/W	Receive Enable When this bit is set to 1, reception is enabled.
5, 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3	TEIE	0	R/W	Transmit End Interrupt Enable When this bit is set to 1, a SSTEI interrupt request is enabled.
2	TIE	0	R/W	Transmit Interrupt Enable When this bit is set to 1, a SSTXI interrupt request is enabled.
1	RIE	0	R/W	Receive Interrupt Enable When this bit is set to 1, an SSRXI interrupt request and an SSOEI interrupt request are enabled.
0	CEIE	0	R/W	Conflict Error Interrupt Enable When this bit is set to 1, a SSCEI interrupt request is enabled.

17.3.5 SS Status Register (SSSR)

SSSR is a status flag register for interrupts.

Bit:	7	6	5	4	3	2	1	0
	-	ORER	-	-	TEND	TDRE	RDRF	CE
Initial value:	0	0	0	0	0	1	0	0
R/W:	R	R/W	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	ORER	0	R/W	Overrun Error If the next data is received while RDRF = 1, an overrun error occurs, indicating abnormal termination. SSRDR stores 1-frame receive data before an overrun error occurs and loses data to be received later. While ORER = 1, consecutive serial reception cannot be continued. Serial transmission cannot be continued, either. [Setting condition] • When one byte of the next reception is completed with RDRF = 1 [Clearing condition] • When writing 0 after reading ORER = 1
5, 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3	TEND	0	R/W	Transmit End [Setting conditions] • When the last bit of transmit data is transmitted while the TENDSTS bit in SSCR2 is cleared to 0 and the TDRE bit is set to 1 • After the last bit of transmit data is transmitted while the TENDSTS bit in SSCR2 is set to 1 and the TDRE bit is set to 1 [Clearing conditions] • When writing 0 after reading TEND = 1 • When writing data to SSTDR with FCLRM = 1 • When the DTC is activated by an SSTXI interrupt and transmit data is written to SSTDR while the DISEL bit in MRB of the DTC is 0 (except when DTC transfer counter value is H'0000)*[1]
2	TDRE	1	R/W	Transmit Data Empty Indicates whether or not SSTDR contains transmit data. [Setting conditions] • When the TE bit in SSER is 0 • When data is transferred from SSTDR to SSTRSR and SSTDR is ready to be written to. [Clearing conditions] • When writing 0 after reading TDRE = 1 • When writing data to SSTDR with TE = 1 and FCLRM = 1 • When the DTC is activated by an SSTXI interrupt and transmit data is written to SSTDR while the DISEL bit in MRB of the DTC is 0 (except when DTC transfer counter value is H'0000)*[1]

Bit	Bit Name	Initial Value	R/W	Description
1	RDRF	0	R/W	Receive Data Register Full Indicates whether or not SSRDR contains receive data. [Setting condition] • When receive data is transferred from SSTRSR to SSRDR after successful serial data reception [Clearing conditions] • When writing 0 after reading RDRF = 1 • When reading receive data from SSRDR with FCLRM = 1 • When the DTC is activated by an SSRXI interrupt and receive data is read into SSRDR while the DISEL bit in MRB of the DTC is 0 (except when DTC transfer counter value is H'0000)*1

Bit	Bit Name	Initial Value	R/W	Description
0	CE	0	R/W	Conflict/Incomplete Error Indicates that a conflict error has occurred when 0 is externally input to the $\overline{SCS}$ pin with SSUMS = 0 (SSU mode) and MSS = 1 (master mode). If the $\overline{SCS}$ pin level changes to 1 with SSUMS = 0 (SSU mode) and MSS = 0 (slave mode), an incomplete error occurs because it is determined that a master device has terminated the transfer. In addition, when SSUMS = 0 (SSU mode) and MSS = 0 (slave mode) and the next serial receive operation starts while RDRF = 1, an incomplete error occurs even if the data received from SSRDR is read before the completion of reception and RDRF is cleared to 0 before the $\overline{SCS}$ pin is set to 1. Data reception does not continue while the CE bit is set to 1. Serial transmission also does not continue. Reset the SSU internal sequencer by setting the SRES bit in SSCRL to 1 before resuming transfer after incomplete error. [Setting conditions] • When a low level is input to the $\overline{SCS}$ pin in master mode (the MSS bit in SSCRH is set to 1) • When the $\overline{SCS}$ pin is changed to 1 during transfer in slave mode (the MSS bit in SSCRH is cleared to 0) • When in slave mode (MSS = 0 in SSCRH), the next serial receive operation starts while RDRF = 1, and data is read from SSRDR before the completion of reception, after which the $\overline{SCS}$ pin is set to 1 [Clearing condition] • When writing 0 after reading CE = 1

Note: * Clearing the FCLRM bit to 0 enables transfer by the DTC.

17.3.6 SS Control Register 2 (SSCR2)

SSCR2 is a register that selects the assert timing of the $\overline{SCS}$ pin, data output timing of the SSO pin, and set timing of the TEND bit.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	TENDSTS	SCSATS	SSODTS	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R	R

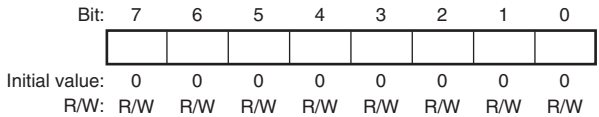
Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TENDSTS	0	R/W	Selects the timing of setting the TEND bit (valid in SSU and master mode). 0: Sets the TEND bit when the last bit is being transmitted 1: Sets the TEND bit after the last bit is transmitted
3	SCSATS	0	R/W	Selects the assertion timing of the $\overline{SCS}$ pin (valid in SSU and master mode). 0: Min. values of t_{LEAD} and t_{LAG} are 1/2 × t_{SUcyc} 1: Min. values of t_{LEAD} and t_{LAG} are 3/2 × t_{SUcyc}
2	SSODTS	0	R/W	Selects the data output timing of the SSO pin (valid in SSU and master mode) 0: While BIDE = 0, MSS = 1, and TE = 1 or while BIDE = 1, TE = 1, and RE = 0, the SSO pin outputs data 1: While BIDE = 0, MSS = 1, and TE = 1 or while BIDE = 1, TE = 1, and RE = 0, the SSO pin outputs data while the $\overline{SCS}$ pin is driven low
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

17.3.7 SS Transmit Data Registers 0 to 3 (SSTDR0 to SSTDR3)

SSTDR is an 8-bit register that stores transmit data. When 8-bit data length is selected by bits DATS1 and DATS0 in SSCRL, SSTDR0 is valid. When 16-bit data length is selected, SSTDR0 and SSTDR1 are valid. When 32-bit data length is selected, SSTDR0 to SSTDR3 are valid. Do not access SSTDR that is not valid.

When the SSU detects that SSTRSR is empty, it transfers the transmit data written in SSTDR to SSTRSR and starts serial transmission. If the next transmit data has already been written to SSTDR during serial transmission, the SSU performs consecutive serial transmission.

Although SSTDR can always be read from or written to by the CPU and DTC, to achieve reliable serial transmission, write transmit data to SSTDR after confirming that the TDRE bit in SSSR is set to 1.



Bit	Bit Name	Initial Value	R/W	Description
7 to 0		All 0	R/W	Serial transmit data

Table 17.3 Setting of DATS Bits in SSCRL and Corresponding SSTDR

	DATS[1:0] Setting			
	00	**01**	**10**	**11 (Invalid setting)**
SSTDR0	Valid	Valid	Valid	Invalid
SSTDR1	Invalid	Valid	Valid	Invalid
SSTDR2	Invalid	Invalid	Valid	Invalid
SSTDR3	Invalid	Invalid	Valid	Invalid

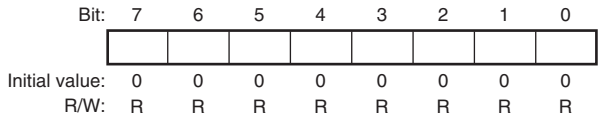
17.3.8 SS Receive Data Registers 0 to 3 (SSRDR0 to SSRDR3)

SSRDR is an 8-bit register that stores receive data. When 8-bit data length is selected by bits DATS1 and DATS0 in SSCRL, SSRDR0 is valid. When 16-bit data length is selected, SSRDR0 and SSRDR1 are valid. When 32-bit data length is selected, SSRDR0 to SSRDR3 are valid. Do not access SSRDR that is not valid.

When the SSU has received 1-byte data, it transfers the received serial data from SSTRSR to SSRDR where it is stored. After this, SSTRSR is ready for reception. Since SSTRSR and SSRDR function as a double buffer in this way, consecutive receive operations can be performed.

Read SSRDR after confirming that the RDRF bit in SSSR is set to 1.

SSRDR is a read-only register, therefore, cannot be written to by the CPU.



Bit	Bit Name	Initial Value	R/W	Description
7 to 0		All 0	R	Serial receive data

Table 17.4 Setting of DATS Bit in SSCRL and Corresponding SSRDR

	DATS[1:0] Setting			
	00	**01**	**10**	**11 (Invalid setting)**
SSRDR0	Valid	Valid	Valid	Invalid
SSRDR1	Invalid	Valid	Valid	Invalid
SSRDR2	Invalid	Invalid	Valid	Invalid
SSRDR3	Invalid	Invalid	Valid	Invalid

17.3.9 SS Shift Register (SSTRSR)

SSTRSR is a shift register that transmits and receives serial data.

When data is transferred from SSTDR to SSTRSR, bit 0 of transmit data is bit 0 in the SSTDR contents (MLS = 0: LSB first communication) and is bit 7 in the SSTDR contents (MLS = 1: MSB first communication). The SSU transfers data from the LSB (bit 0) in SSTRSR to the SSO pin to perform serial data transmission.

In reception, the SSU sets serial data that has been input via the SSI pin in SSTRSR from the LSB (bit 0). When 1-byte data has been received, the SSTRSR contents are automatically transferred to SSRDR. SSTRSR cannot be directly accessed by the CPU.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

17.4 Operation

17.4.1 Transfer Clock

A transfer clock can be selected from seven internal clocks and an external clock. Before using this module, enable the SSCK pin function in the PFC. When the MSS bit in SSCRH is 1, an internal clock is selected and the SSCK pin is used as an output pin. When transfer is started, the clock with the transfer rate set by bits CKS2 to CKS0 in SSMR is output from the SSCK pin. When MSS = 0, an external clock is selected and the SSCK pin is used as an input pin.

17.4.2 Relationship of Clock Phase, Polarity, and Data

The relationship of clock phase, polarity, and transfer data depends on the combination of the CPOS and CPHS bits in SSMR when the value of the SSUMS bit in SSCRL is 0. Figure 17.2 shows the relationship. When SSUMS = 1, the CPHS setting is invalid although the CPOS setting is valid.

Setting the MLS bit in SSMR selects that MSB or LSB first communication. When MLS = 0, data is transferred from the LSB to the MSB. When MLS = 1, data is transferred from the MSB to the LSB.

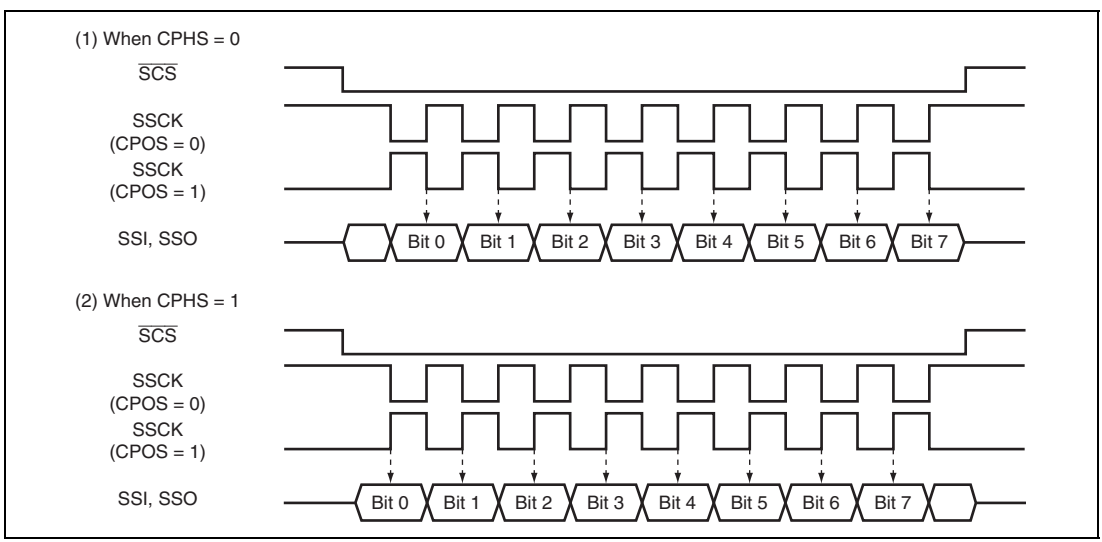


Figure 17.2 Relationship of Clock Phase, Polarity, and Data

17.4.3 Relationship between Data Input/Output Pins and Shift Register

The connection between data input/output pins and the SS shift register (SSTRSR) depends on the combination of the MSS and BIDE bits in SSCRH and the SSUMS bit in SSCRL. Figure 17.3 shows the relationship.

The SSU transmits serial data from the SSO pin and receives serial data from the SSI pin when operating with BIDE = 0 and MSS = 1 (standard, master mode) (see figure 17.3 (1)). The SSU transmits serial data from the SSI pin and receives serial data from the SSO pin when operating with BIDE = 0 and MSS = 0 (standard, slave mode) (see figure 17.3 (2)).

The SSU transmits and receives serial data from the SSO pin regardless of master or slave mode when operating with BIDE = 1 (bidirectional mode) (see figures 17.3 (3) and (4)).

However, even if both the TE and RE bits are set to 1, transmission and reception are not performed simultaneously. Either the TE or RE bit must be selected.

The SSU transmits serial data from the SSO pin and receives serial data from the SSI pin when operating with SSUMS = 1. The SSCK pin outputs the internal clock when MSS = 1 and function as an input pin when MSS = 0 (see figures 17.3 (5) and (6)).

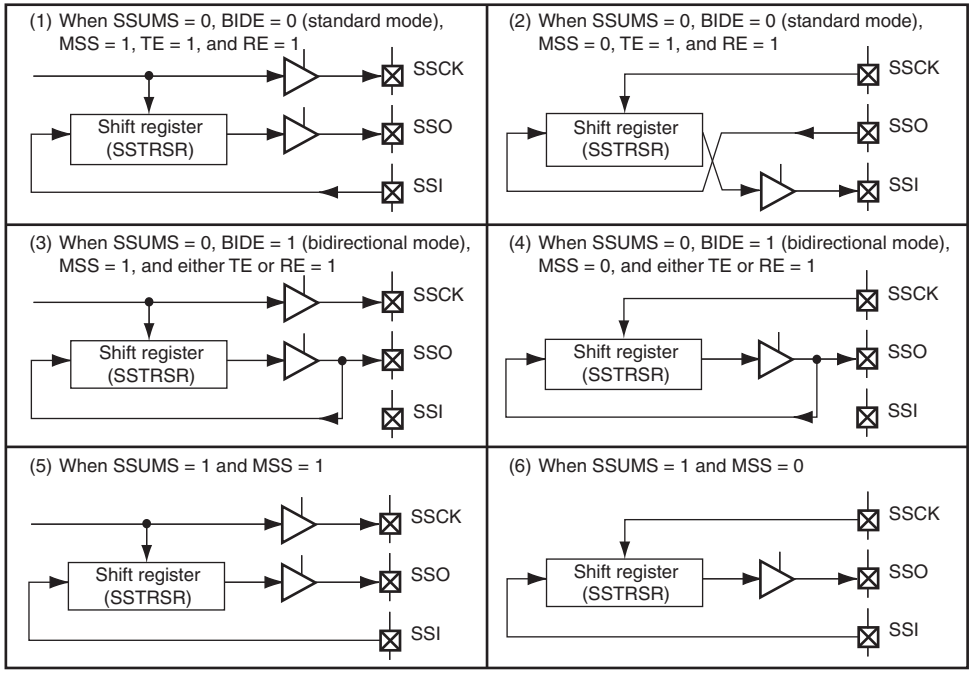


Figure 17.3 Relationship between Data Input/Output Pins and the Shift Register

17.4.4 Communication Modes and Pin Functions

The SSU switches the input/output pin (SSI, SSO, SSCK, and $\overline{SCS}$) functions according to the communication modes and register settings. The input/output directions of the pins should be selected in the port I/O registers. The relationship of communication modes and input/output pin functions are shown in tables 17.5 to 17.7.

Table 17.5 Communication Modes and Pin States of SSI and SSO Pins

Communication Mode	Register Setting					Pin State	
	SSUMS	BIDE	MSS	TE	RE	SSI	SSO
SSU communication mode	0	0	0	0	1	—	Input
				1	0	Output	—
					1	Output	Input
			1	0	1	Input	—
				1	0	—	Output
					1	Input	Output
SSU (bidirectional) communication mode	0	1	0	0	1	—	Input
				1	0	—	Output
			1	0	1	—	Input
				1	0	—	Output
Clock synchronous communication mode	1	0	0	0	1	Input	—
				1	0	—	Output
					1	Input	Output
			1	0	1	Input	—
				1	0	—	Output
					1	Input	Output

[Legend]

—: Not used as SSU pin

Table 17.6 Communication Modes and Pin States of SSCK Pin

	Register Setting		Pin State
Communication Mode	**SSUMS**	**MSS**	**SSCK**
SSU communication mode	0	0	Input
		1	Output
Clock synchronous communication mode	1	0	Input
		1	Output

[Legend]

—: Not used as SSU pin

Table 17.7 Communication Modes and Pin States of $\overline{\text{SCS}}$ Pin

Communication Mode	Register Setting				Pin State
	SSUMS	**MSS**	**CSS1**	**CSS0**	**$\overline{\text{SCS}}$**
SSU communication mode	0	0	x	x	Input
		1	0	0	—
			0	1	—
			1	0	Automatic input/output
			1	1	Output
Clock synchronous communication mode	1	x	x	x	—

[Legend]

x: Don't care

—: Not used as SSU pin

17.4.5 SSU Mode

In SSU mode, data communications are performed via four lines: clock line (SSCK), data input line (SSI or SSO), data output line (SSI or SSO), and chip select line ($\overline{SCS}$).

In addition, the SSU supports bidirectional mode in which a single pin functions as data input and data output lines.

(1) Initial Settings in SSU Mode

Figure 17.4 shows an example of the initial settings in SSU mode. Before data transfer, clear both the TE and RE bits in SSER to 0 to set the initial values.

Note: Before changing operating modes and communications formats, clear both the TE and RE bits to 0. Although clearing the TE bit to 0 sets the TDRE bit to 1, clearing the RE bit to 0 does not change the values of the RDRF and ORER bits and SSRDR. Those bits retain the previous values.

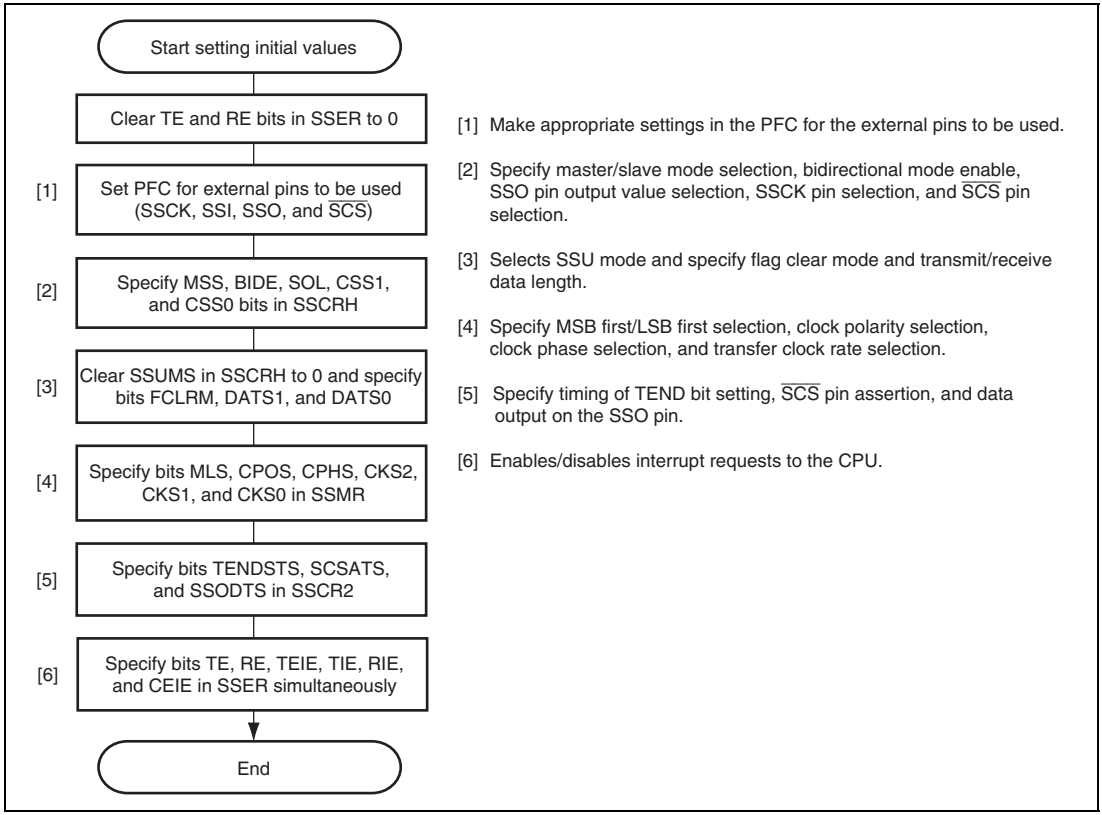


Figure 17.4 Example of Initial Settings in SSU Mode

(2) Data Transmission

Figure 17.5 shows an example of transmission operation, and figure 17.6 shows a flowchart example of data transmission.

When transmitting data, the SSU operates as shown below.

In master mode, the SSU outputs a transfer clock and data. In slave mode, when a low level signal is input to the $\overline{\text{SCS}}$ pin and a transfer clock is input to the SSCK pin, the SSU outputs data in synchronization with the transfer clock.

Writing transmit data to SSTDR after the TE bit is set to 1 clears the TDRE bit in SSSR to 0, and the SSTDR contents are transferred to SSTRSR. After that, the SSU sets the TDRE bit to 1 and starts transmission. At this time, if the TIE bit in SSER is set to 1, a TXI interrupt is generated.

When 1-frame data has been transferred with TDRE = 0, the SSTDR contents are transferred to SSTRSR to start the next frame transmission. When the 8th bit of transmit data has been transferred with TDRE = 1, the TEND bit in SSSR is set to 1 and the state is retained. At this time, if the TEIE bit is set to 1, a TEI interrupt is generated. After transmission, the output level of the SSCK pin is fixed high when CPOS = 0 and low when CPOS = 1.

While the ORER bit in SSSR is set to 1, transmission is not performed. Check that the ORER bit is cleared to 0 before transmission.

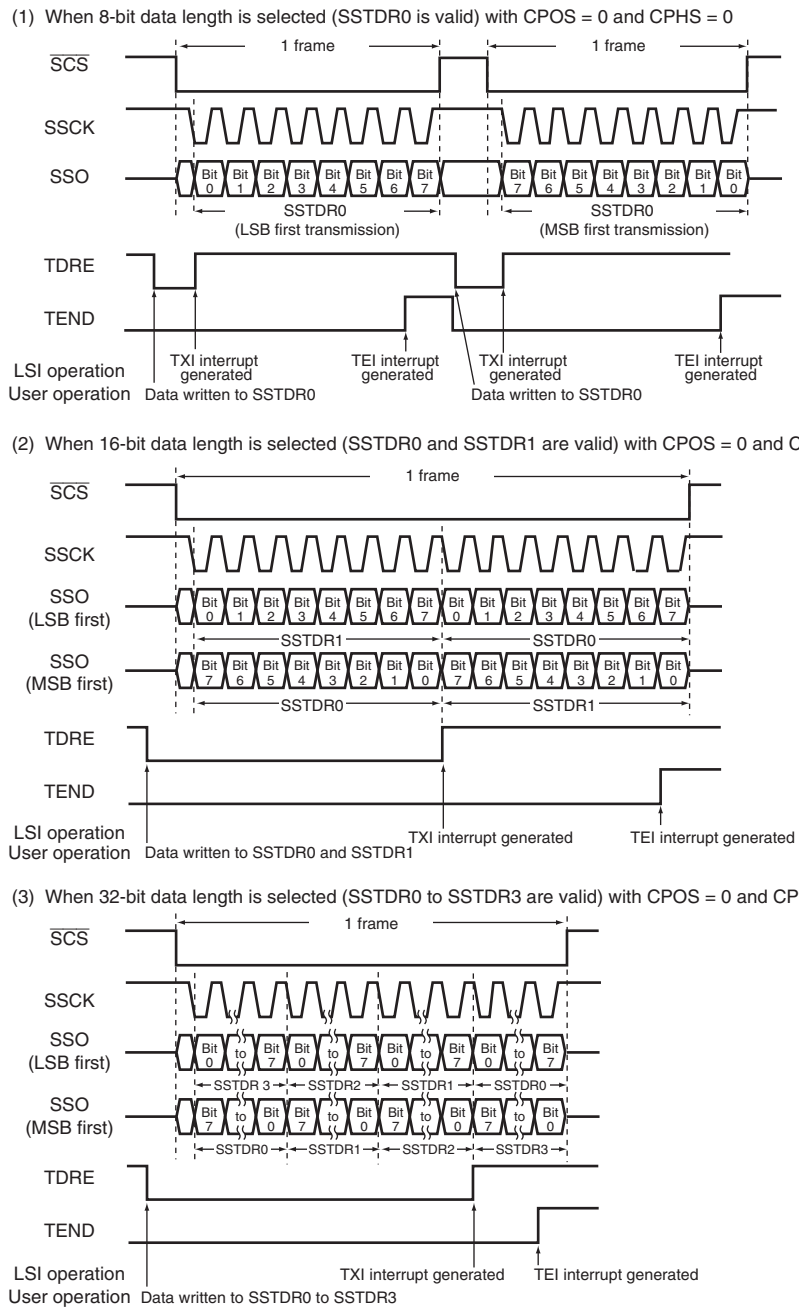
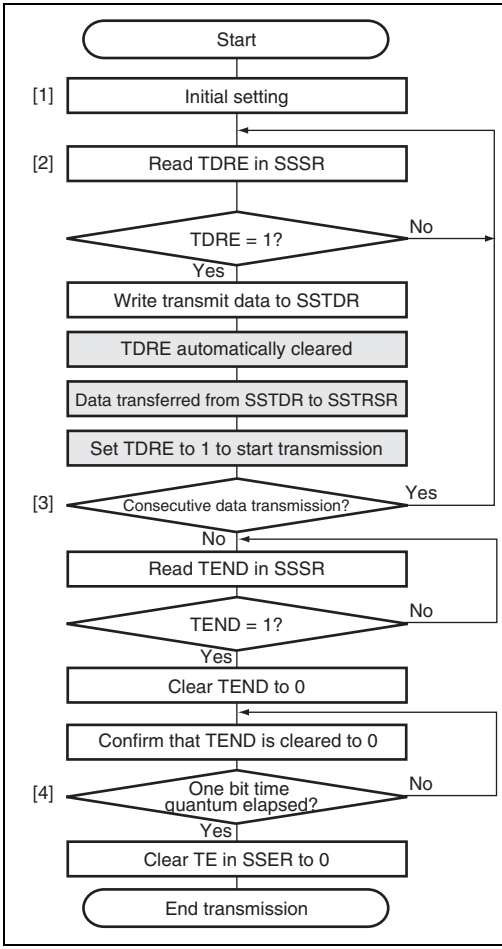


Figure 17.5 Example of Transmission Operation (SSU Mode)



[1] Initial setting:
Specify the transmit data format.

[2] Check that the SSU state and write transmit data:
Write transmit data to SSTDR after reading and confirming that the TDRE bit is 1. The TDRE bit is automatically cleared to 0 and transmission is started by writing data to SSTDR.

[3] Procedure for consecutive data transmission:
To continue data transmission, confirm that the TDRE bit is 1 meaning that SSTDR is ready to be written to. After that, data can be written to SSTDR. The TDRE bit is automatically cleared to 0 by writing data to SSTDR.

[4] Procedure for data transmission end:
To end data transmission, confirm that the TEND bit is cleared to 0. After completion of transmitting the last bit, clear the TE bit to 0.

Note: Hatching boxes represent SSU internal operations.

Figure 17.6 Flowchart Example of Data Transmission (SSU Mode)

(3) Data Reception

Figure 17.7 shows an example of reception operation, and figure 17.8 shows a flowchart example of data reception. When receiving data, the SSU operates as shown below.

After setting the RE bit to 1 and dummy-reading SSRDR, the SSU starts data reception.

In master mode, the SSU outputs a transfer clock and receives data. In slave mode, when a low level signal is input to the $\overline{\text{SCS}}$ pin and a transfer clock is input to the SSCK pin, the SSU receives data in synchronization with the transfer clock.

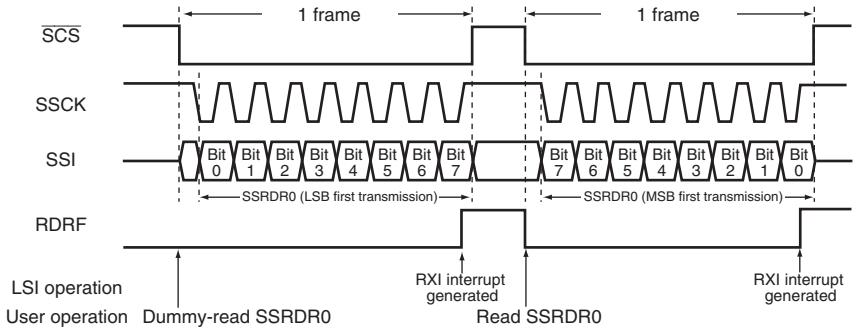
When 1-frame data has been received, the RDRF bit in SSSR is set to 1 and the receive data is stored in SSRDR. At this time, if the RIE bit in SSER is set to 1, an RXI interrupt is generated. The RDRF bit is automatically cleared to 0 by reading SSRDR.

When the RDRF bit has been set to 1 at the 8th rising edge of the transfer clock, the ORER bit in SSSR is set to 1. This indicates that an overrun error (OEI) has occurred. At this time, data reception is stopped. While the ORER bit in SSSR is set to 1, reception is not performed. To resume the reception, clear the ORER bit to 0.

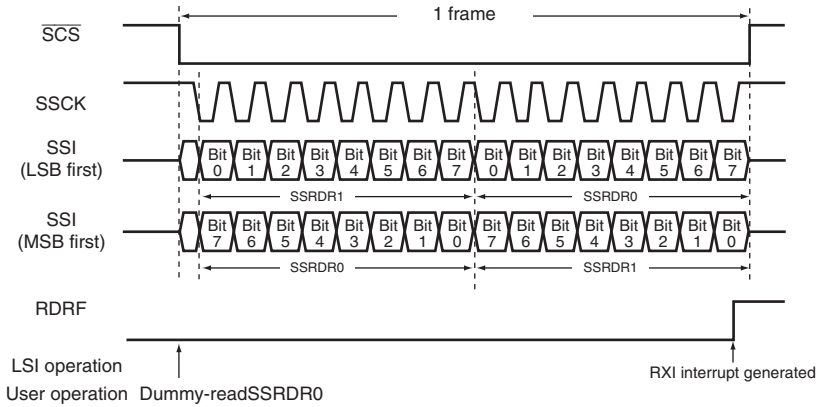
When setting the SSU to slave mode to perform continuous reception, read SSRDR before starting the next receive operation. If the next receive operation starts before SSRDR is read and RDRF is cleared to 0, and SSRDR is read before reception completes, CE in SSSR is set to 1 after the completion of reception.

In addition, if the next receive operation starts before SSRDR is read and RDRF is cleared to 0, and SSRDR is not read until after reception completes, the receive data is discarded even though neither CE nor ORER in SSSR is set to 1.

(1) When 8-bit data length is selected (SSRDR0 is valid) with CPOS = 0 and CPHS = 0



(2) When 16-bit data length is selected (SSRDR0 and SSRDR1 are valid) with CPOS = 0 and CPHS = 0



(3) When 32-bit data length is selected (SSRDR0 to SSRDR3 are valid) with CPOS = 0 and CPHS = 0

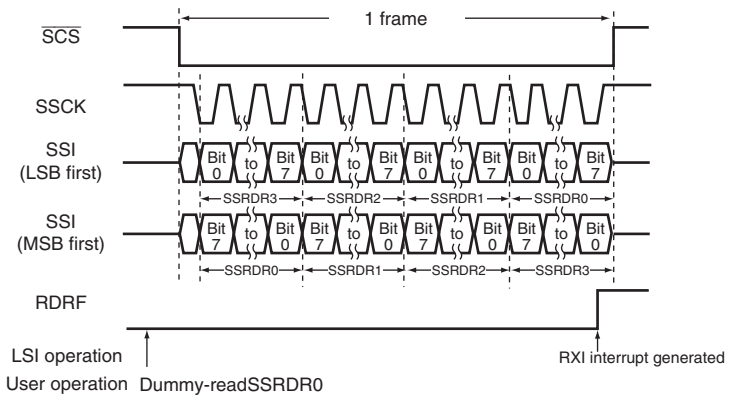
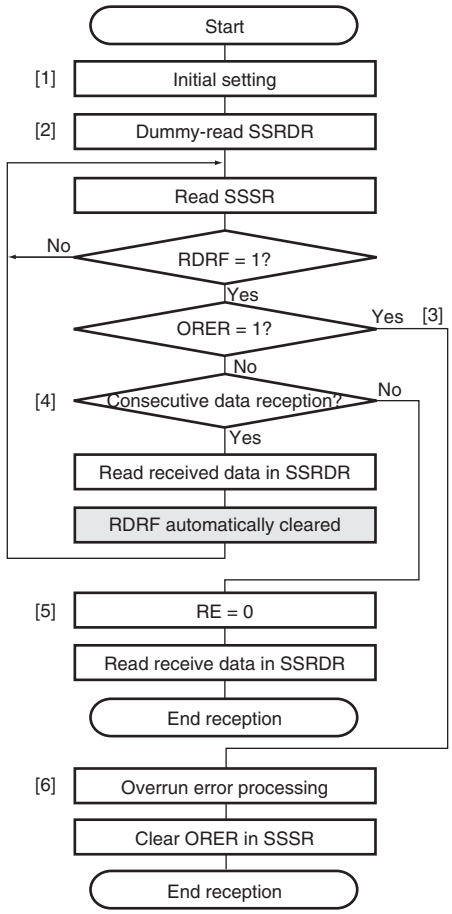


Figure 17.7 Example of Reception Operation (SSU Mode)



[1] Initial setting:
Specify the receive data format.

[2] Start reception:
When SSRDR is dummy-read with RE = 1, reception is started.

[3], [6] Receive error processing:
When a receive error occurs, execute the designated error processing after reading the ORER bit in SSSR. After that, clear the ORER bit to 0. While the ORER bit is set to 1, transmission or reception is not resumed.

[4] To continue single reception:
When continuing single reception, wait for time of t_{SUcyc} while the RDRF flag is set to 1 and then read receive data in SSRDR.
The next single reception starts after reading receive data in SSRDR.

[5] To complete reception:
To complete reception, read receive data after clearing the RE bit to 0. When reading SSRDR without clearing the RE bit, reception is resumed.

Note: Hatching boxes represent SSU internal operations.

Figure 17.8 Flowchart Example of Data Reception (SSU Mode)

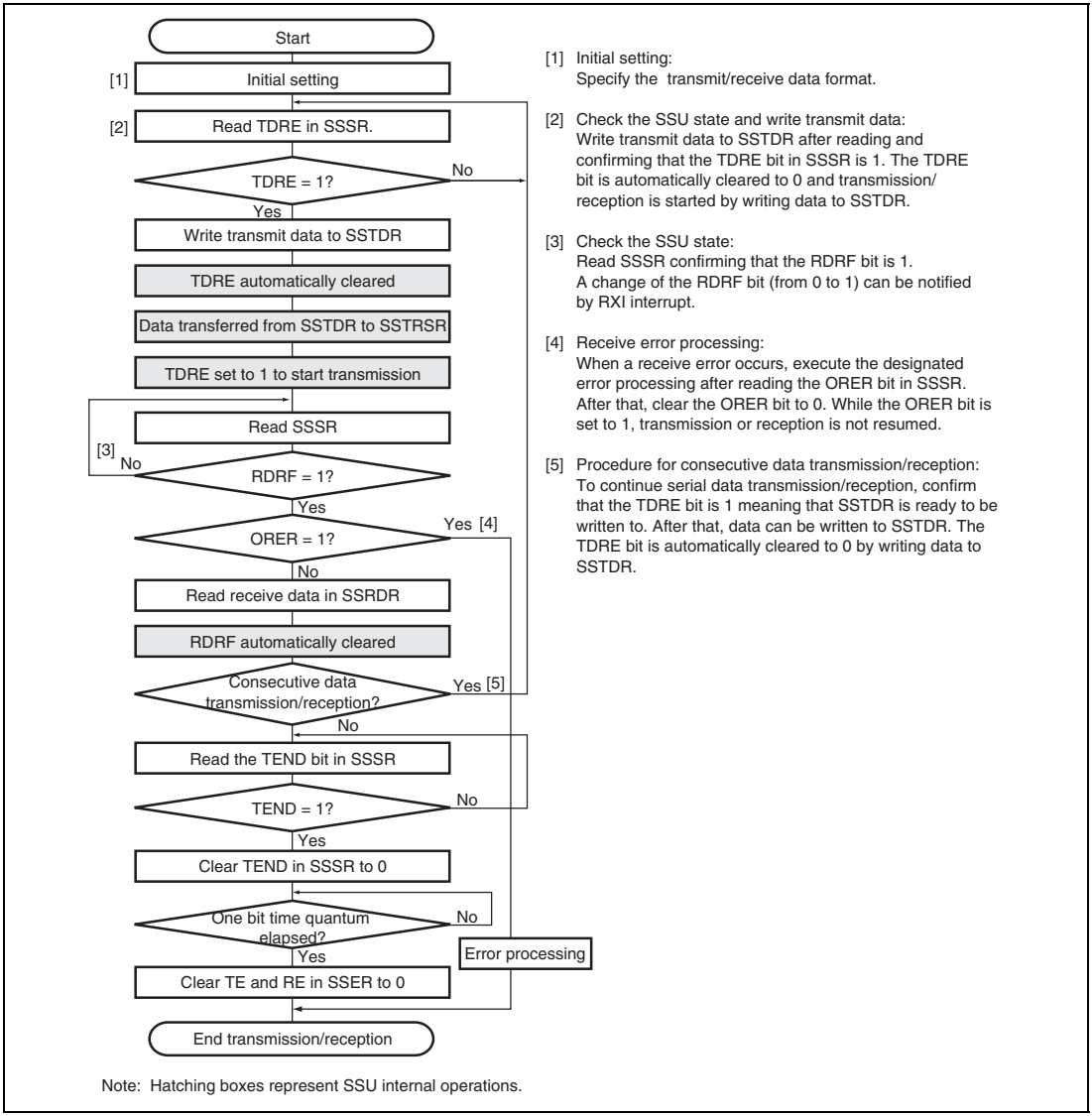
(4) Data Transmission/Reception

Figure 17.9 shows a flowchart example of simultaneous transmission/reception. The data transmission/reception is performed combining the data transmission and data reception as mentioned above. The data transmission/reception is started by writing transmit data to SSTDR with TE = RE = 1.

Before switching transmission mode (TE = 1) or reception mode (RE = 1) to transmission/reception mode (TE = RE = 1), clear the TE and RE bits to 0. When starting the

transfer, confirm that the TEND, RDRF, and ORER bits are cleared to 0 before setting the TE or RE bit to 1.

If the value of RDRF is 1 when the 8th clock rises, ORER in SSSR is set to 1, an overrun error occurs, and reception halts. Receive operation is not possible while ORER is set to 1. To restart reception, first clear ORER to 0.



Note: Hatching boxes represent SSU internal operations.

Figure 17.9 Flowchart Example of Simultaneous Transmission/Reception (SSU Mode)

17.4.6 $\overline{\text{SCS}}$ Pin Control and Conflict Error

When bits CSS1 and CSS0 in SSCRH are set to B'10 and the SSUMS bit in SSCRL is cleared to 0, the SCS pin becomes an input pin (Hi-Z) before the serial transfer is started and after the serial transfer is complete. Because of this, the SSU performs conflict error detection during these periods. If a low level signal is input to the SCS pin during these periods, it is detected as a conflict error. At this time, the CE bit in SSSR is set to 1 and the MSS bit is cleared to 0.

Note: While the CE bit is set to 1, transmission or reception is not resumed. Clear the CE bit to 0 before resuming the transmission or reception.

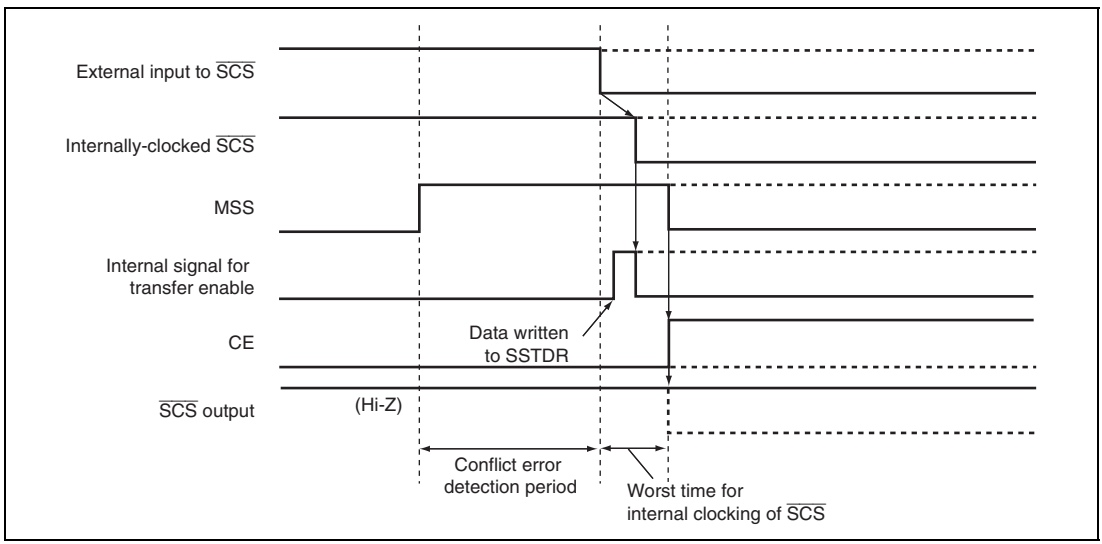


Figure 17.10 Conflict Error Detection Timing (Before Transfer)

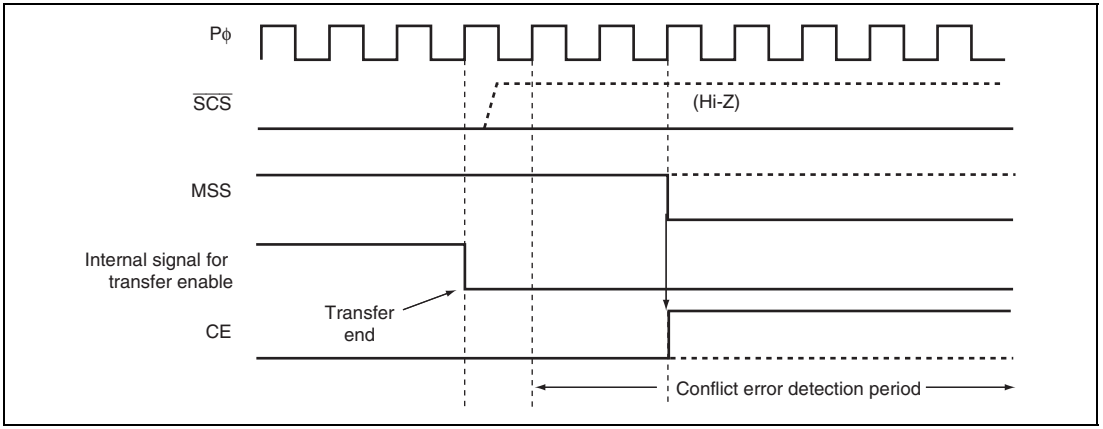


Figure 17.11 Conflict Error Detection Timing (After Transfer End)

17.4.7 Clock Synchronous Communication Mode

In clock synchronous communication mode, data communications are performed via three lines: clock line (SSCK), data input line (SSI), and data output line (SSO).

(1) Initial Settings in Clock Synchronous Communication Mode

Figure 17.12 shows an example of the initial settings in clock synchronous communication mode. Before data transfer, clear both the TE and RE bits in SSER to 0 to set the initial values.

Note: Before changing operating modes and communications formats, clear both the TE and RE bits to 0. Although clearing the TE bit to 0 sets the TDRE bit to 1, clearing the RE bit to 0 does not change the values of the RDRF and ORER bits and SSRDR. Those bits retain the previous values.

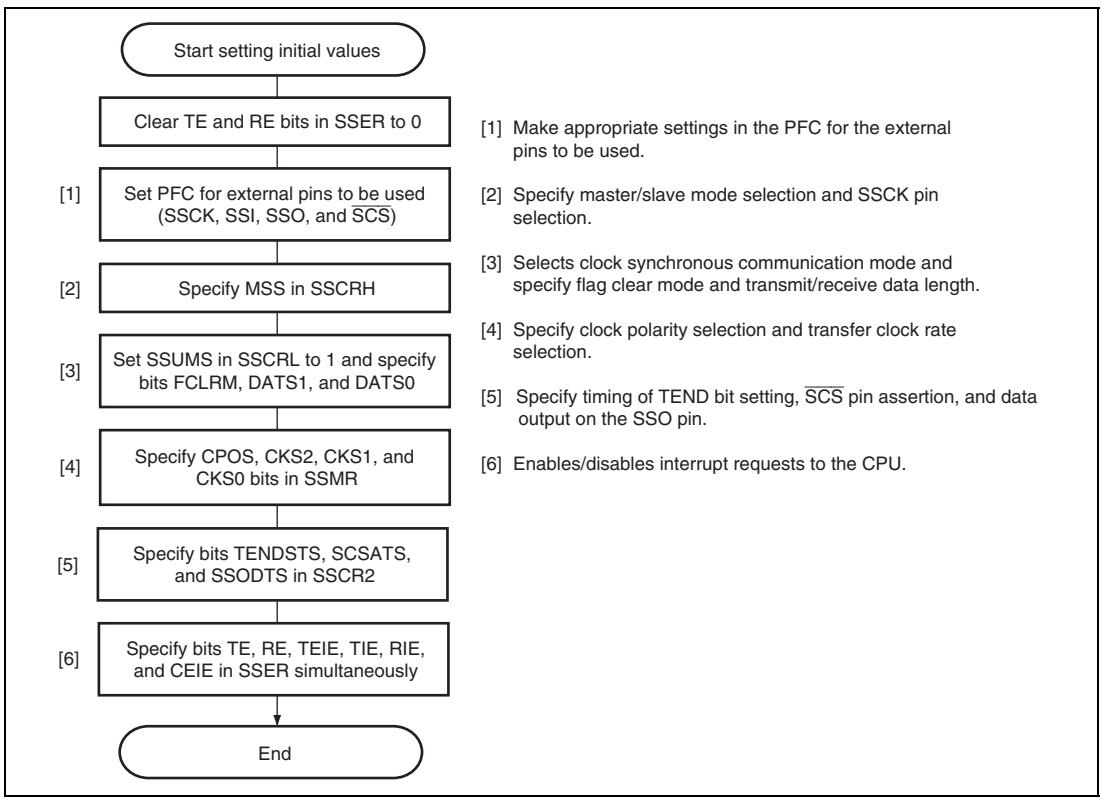


Figure 17.12 Example of Initial Settings in Clock Synchronous Communication Mode

(2) Data Transmission

Figure 17.13 shows an example of transmission operation, and figure 17.14 shows a flowchart example of data transmission. When transmitting data in clock synchronous communication mode, the SSU operates as shown below.

In master mode, the SSU outputs a transfer clock and data. In slave mode, when a transfer clock is input to the SSCK pin, the SSU outputs data in synchronization with the transfer clock.

Writing transmit data to SSTDR after the TE bit is set to 1 clears the TDRE bit in SSSR to 0, and the SSTDR contents are transferred to SSTRSR. After that, the SSU sets the TDRE bit to 1 and starts transmission. At this time, if the TIE bit in SSER is set to 1, a TXI interrupt is generated.

When 1-frame data has been transferred with TDRE = 0, the SSTDR contents are transferred to SSTRSR to start the next frame transmission. When the 8th bit of transmit data has been transferred with TDRE = 1, the TEND bit in SSSR is set to 1 and the state is retained. At this time, if the TEIE bit is set to 1, a TEI interrupt is generated.

While the ORER bit in SSSR is set to 1, transmission is not performed. Check that the ORER bit is cleared to 0 before transmission.

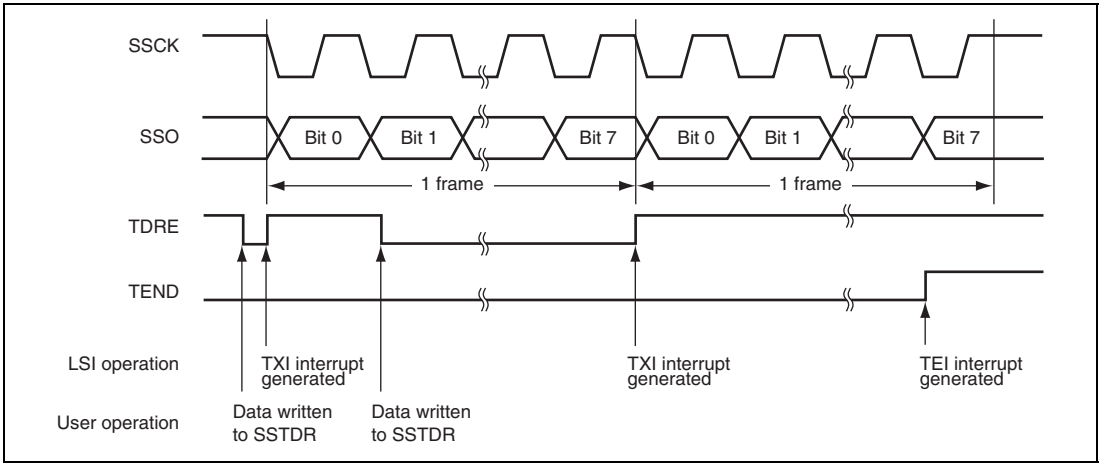
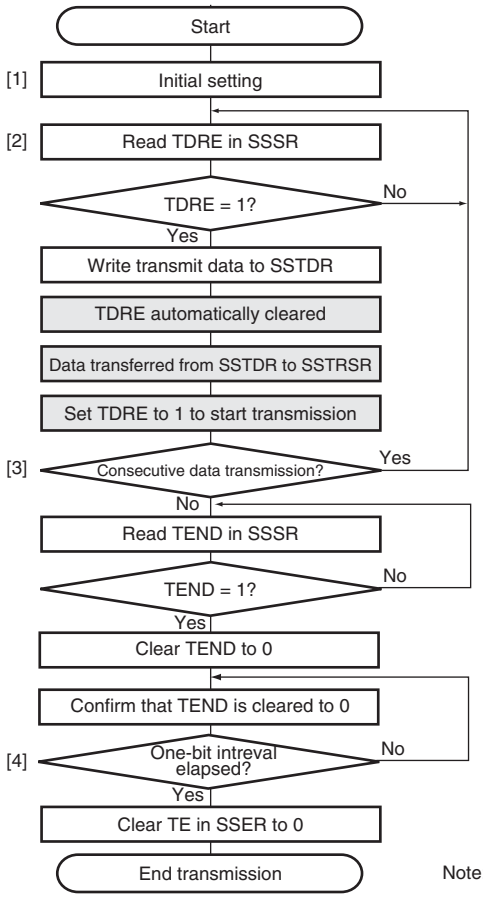


Figure 17.13 Example of Transmission Operation (Clock Synchronous Communication Mode)



[1] Initial setting:
Specify the transmit data format.

[2] Check that the SSU state and write transmit data:
Write transmit data to SSTDR after reading and confirming that the TDRE bit is 1. The TDRE bit is automatically cleared to 0 and transmission is started by writing data to SSTDR.

[3] Procedure for consecutive data transmission:
To continue data transmission, confirm that the TDRE bit is 1 meaning that SSTDR is ready to be written to. After that, data can be written to SSTDR. The TDRE bit is automatically cleared to 0 by writing data to SSTDR.

[4] Procedure for data transmission end:
To end data transmission, confirm that the TEND bit is cleared to 0. After completion of transmitting the last bit, clear the TE bit to 0.

Note: Hatched boxes represent SSU internal operations.

Figure 17.14 Flowchart Example of Transmission Operation (Clock Synchronous Communication Mode)

(3) Data Reception

Figure 17.15 shows an example of reception operation, and figure 17.16 shows a flowchart example of data reception. When receiving data, the SSU operates as shown below.

After setting the RE bit in SSER to 1, the SSU starts data reception.

In master mode, the SSU outputs a transfer clock and receives data. In slave mode, when a transfer clock is input to the SSCK pin, the SSU receives data in synchronization with the transfer clock.

When 1-frame data has been received, the RDRF bit in SSSR is set to 1 and the receive data is stored in SSRDR. At this time, if the RIE bit is set to 1, an RXI interrupt is generated. The RDRF bit is automatically cleared to 0 by reading SSRDR.

When setting the SSU to slave mode to perform continuous reception, read SSRDR before starting the next receive operation. If the next receive operation starts before SSRDR is read and RDRF is cleared to 0, the integrity of subsequent data cannot be guaranteed.

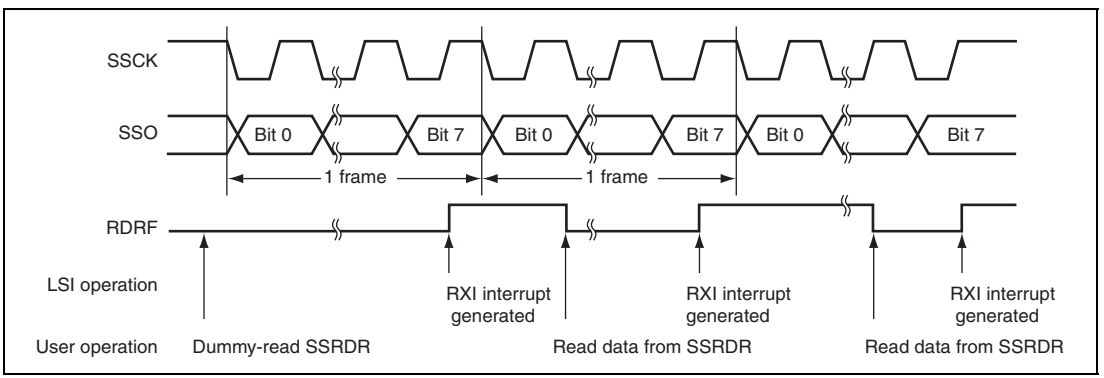
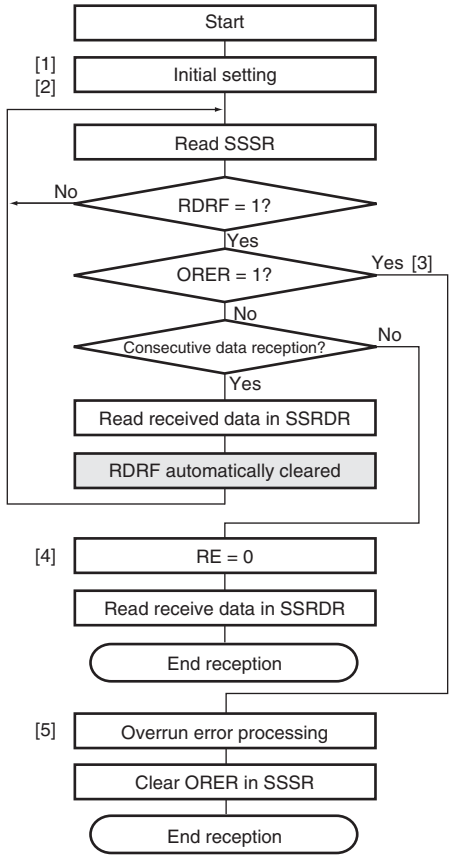


Figure 17.15 Example of Reception Operation (Clock Synchronous Communication Mode)



[1] Initial setting:
Specify the receive data format.

[2] Start reception:
When setting the RE bit to 1, reception is started.

[3], [5] Receive error processing:
When a receive error occurs, execute the designated error processing after reading the ORER bit in SSSR. After that, clear the ORER bit to 0. While the ORER bit is set to 1, transmission or reception is not resumed.

[4] To complete reception:
To complete reception, read receive data after clearing the RE bit to 0. When reading SSRDR without clearing the RE bit, reception is resumed.

Note: Hatching boxes represent SSU internal operations.

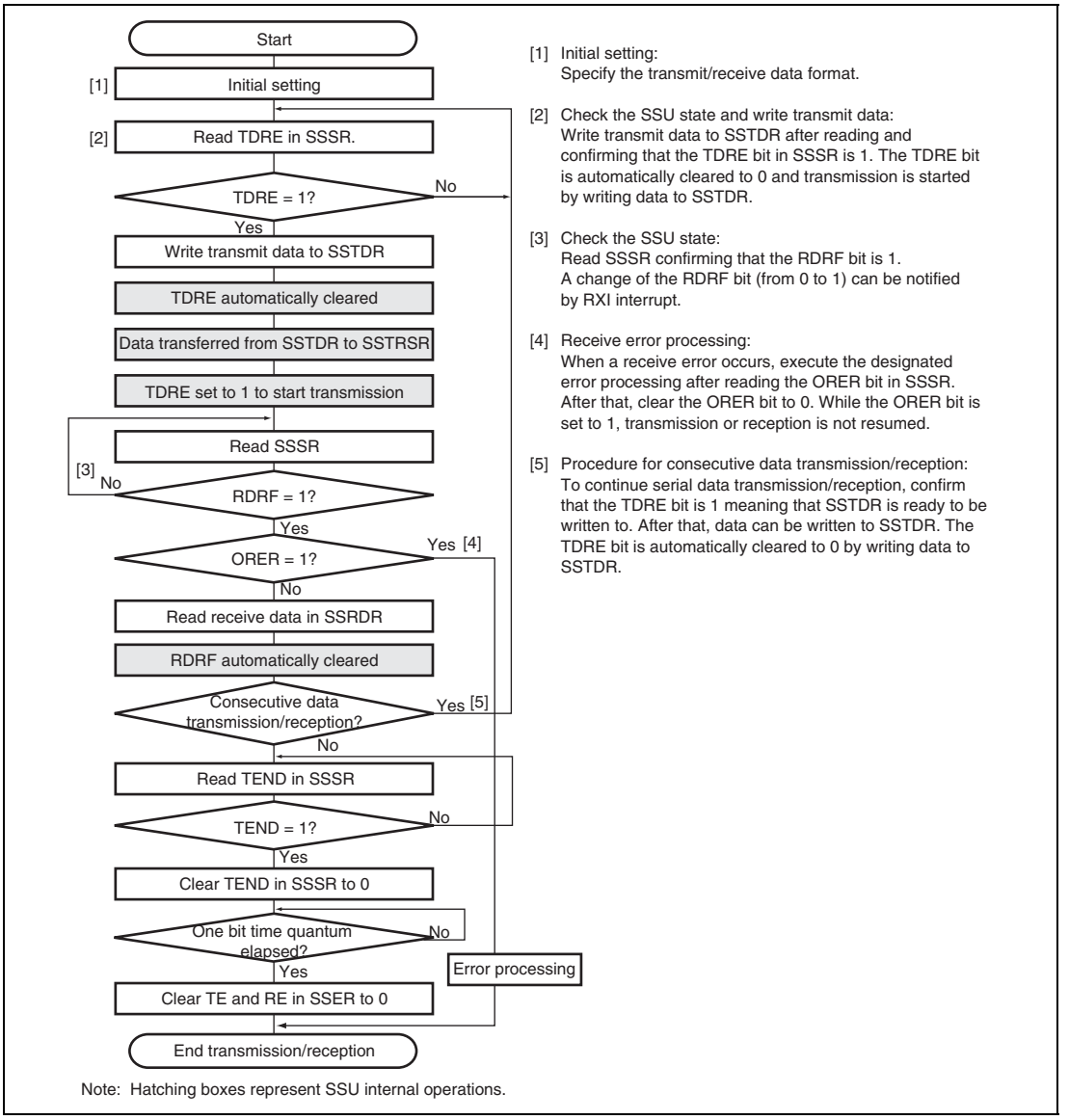
Figure 17.16 Flowchart Example of Data Reception (Clock Synchronous Communication Mode)

(4) Data Transmission/Reception

Figure 17.17 shows a flowchart example of simultaneous transmission/reception. The data transmission/reception is performed combining the data transmission and data reception as mentioned above. The data transmission/reception is started by writing transmit data to SSTDR with TE = RE = 1.

Before switching transmission mode (TE = 1) or reception mode (RE = 1) to transmission/reception mode (TE = RE = 1), clear the TE and RE bits to 0. When starting the transfer, confirm that the TEND, RDRF, and ORER bits are cleared to 0 before setting the TE or RE bits to 1.

If the value of RDRF is 1 when the 8th clock rises, ORER in SSSR is set to 1, an overrun error occurs, and reception halts. Receive operation is not possible while ORER is set to 1. To restart reception, first clear ORER to 0.



Note: Hatching boxes represent SSU internal operations.

Figure 17.17 Flowchart Example of Simultaneous Transmission/Reception (Clock Synchronous Communication Mode)

17.5 SSU Interrupt Sources and DTC

The SSU interrupt requests are an overrun error, a conflict error, a receive data register full, transmit data register empty, and a transmit end interrupts. Of these interrupt sources, a receive data register full, and a transmit data register empty can activate the DTC for data transfer.

Since both an overrun error and a conflict error interrupts are allocated to the SSERI vector address, and both a transmit data register empty and a transmit end interrupts are allocated to the SSTXI vector address, the interrupt source should be decided by their flags. Table 17.8 lists the interrupt sources.

When an interrupt condition shown in table 17.8 is satisfied, an interrupt is requested. Clear the interrupt source by CPU or DTC data transfer.

Table 17.8 SSU Interrupt Sources

Abbreviation	Interrupt Source	Symbol	Interrupt Condition	DTC Activation
SSERI	Overrun error	SSOEI	(RIE = 1) • (ORER = 1)	—
	Conflict error	SSCEI	(CEIE = 1) • (CE = 1)	—
SSRXI	Receive data register full	SSRXI	(RIE = 1) • (RDRF = 1)	Yes
SSTXI	Transmit data register empty	SSTXI	(TIE = 1) • (TDRE = 1)	Yes
	Transmit end	SSTEI	(TEIE = 1) • (TEND = 1)	—

17.6 Usage Notes

17.6.1 Module Standby Mode Setting

The SSU operation can be disabled or enabled using the standby control register. The initial setting is for SSU operation to be halted. Access to registers is enabled by clearing module standby mode. For details, refer to section 26, Power-Down Modes.

17.6.2 Access to SSTDR and SSRDR Registers

Do not access SSTDR and SSRDR registers not validated by the setting of the DATS bits of the SSCRL register. If accessed, transmission or reception thereafter may not be performed normally.

17.6.3 Continuous Transmission/Reception in SSU Slave Mode

During continuous transmission/reception in SSU slave mode, negate the $\overline{\text{SCS}}$ pin (high level) for every frame. If the $\overline{\text{SCS}}$ pin is kept asserted (low level) for more than one frame, transmission or reception cannot be performed correctly.

17.6.4 Note for Reception Operations in SSU Slave Mode

In continuous reception when slave reception in SSU mode has been selected, read the SS receive-data register (SSRDR) before each next round of reception starts (i.e. before an externally connected master device starts a next round of transmission).

If the next round of reception starts after the SS status register receive-data full (RDRF) bit has been set to 1 but before the SSRDR has been read, and the SSRDR is read before the reception of one frame is complete, the conflict/incomplete error bit in SSSR will be set to 1 on completion of reception.

Furthermore, when the next round of reception starts after the receive-data full (RDRF) bit has been set to 1 and before the SSRDR has been read, and the SSRDR has not been read by the end of the reception of the frame, the CE and overflow-error (ORER) bits will not have been set, but the received data will be discarded.

Further note that this point for caution does not apply to simultaneous transmission and reception in SSU slave mode or to clock-synchronous mode.

17.6.5 Note on Master Transmission and Master Transmission/Reception Operations in SSU Mode

To perform master transmission or transmission/reception in SSU mode, perform one of the following operations:

- After the TDRE flag in the SSSR register is set to 1, store the next byte of transmit data in SSTDR before transmission of the second to last bit starts.
- Store the next byte of transmit data in SSTDR after confirming that the TEND flag in the SSSR register has been set to 1.
- Use the SSU with TENDSTS in the SSCR2 register cleared to 0, or with both TENDSTS and SCSATS in the SSCR2 register set to 1.

17.6.6 Note on DTC Transfers

When a DTC transfer occurs with SSTXI as the activation source, TDRE is not cleared when the transfer counter reaches H'0000 but communication operation starts anyway.

When using the SSTXI interrupt to clear the flag, perform interrupt handling first.

However, do not clear the flag within the SSTXI interrupt handler when the initial value of the DTC's transfer counter is set to H'0001 and DISEL is set to 1. In this case, clearing the flag by the interrupt handler may cause the SSU to start communication operation a second time.

Section 18 I²C Bus Interface 2 (IIC2)

The I²C bus interface 2 conforms to and provides a subset of the Philips I²C (Inter-IC) bus interface functions. However, the configuration of the registers that control the I²C bus differs partly from the Philips register configuration.

18.1 Features

- Selection of I²C format or clock synchronous serial format
- Continuous transmission/reception

  Since the shift register, transmit data register, and receive data register are independent from each other, the continuous transmission/reception can be performed.
- Module standby mode can be set

I²C bus format:

- Start and stop conditions generated automatically in master mode
- Selection of acknowledge output levels when receiving
- Automatic loading of acknowledge bit when transmitting
- Bit synchronization function

  In master mode, the state of SCL is monitored per bit, and the timing is synchronized automatically.

  If transmission/reception is not yet possible, set the SCL to low until preparations are completed.
- Six interrupt sources

  Transmit data empty (including slave-address match), transmit end, receive data full (including slave-address match), arbitration lost, NACK detection, and stop condition detection

  The data transfer controller (DTC) can be activated by a transmit-data-empty interrupt or receive-data-full interrupt to transfer data.
- Direct bus drive

  Two pins, SCL and SDA pins, function as NMOS open-drain outputs when the bus drive function is selected.

Clock synchronous serial format:

- Four interrupt sources

  Transmit-data-empty, transmit-end, receive-data-full, and overrun error
- The data transfer controller (DTC) can be activated by a transmit-data-empty request or receive-data-full request to transfer data.

Figure 18.1 shows a block diagram of the I²C bus interface.

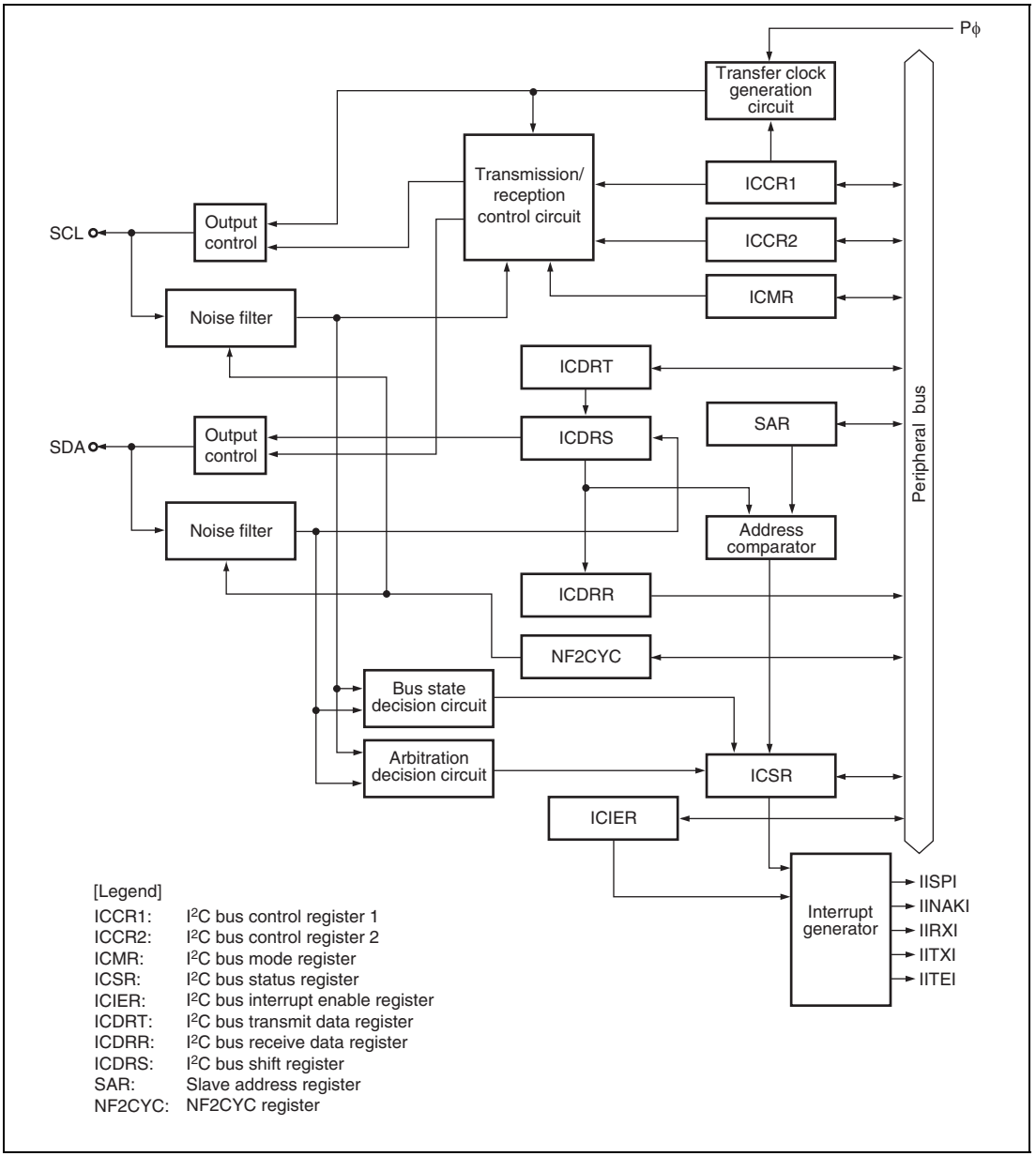


Figure 18.1 Block Diagram of I²C Bus Interface 2

18.2 Input/Output Pins

Table 18.1 shows the pin configuration for the I²C bus interface 2.

Table 18.1 I²C Bus Interface Pin Configuration

Pin Name	Symbol	I/O	Function
Serial clock	SCL	I/O	I²C serial clock input/output
Serial data	SDA	I/O	I²C serial data input/output

Figure 18.2 shows an example of I/O pin connections to external circuits.

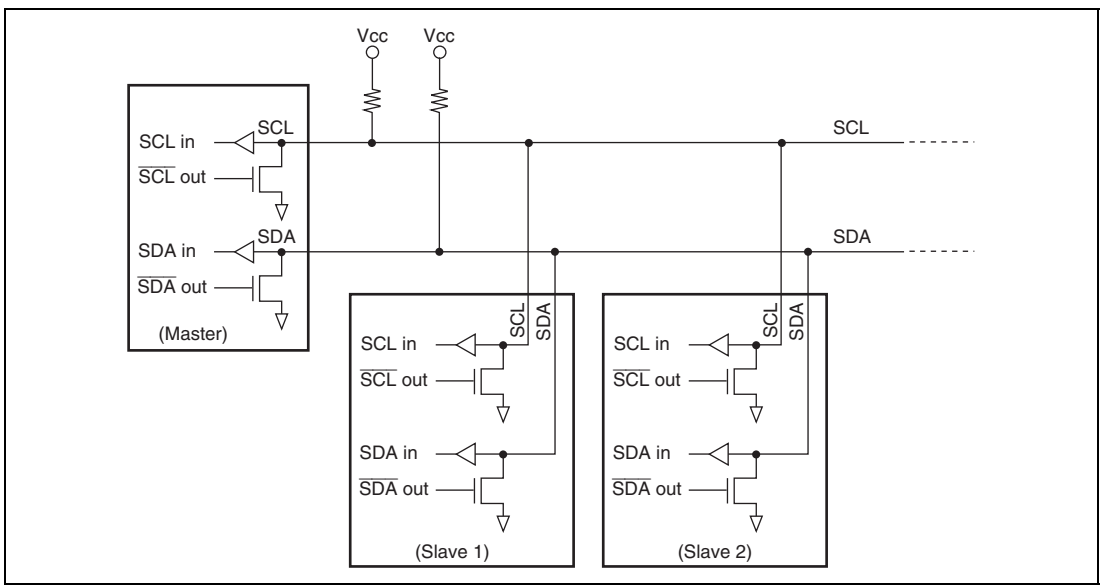


Figure 18.2 External Circuit Connections of I/O Pins

18.3 Register Descriptions

The I²C bus interface 2 has the following registers. For details on register addresses and register states during each processing, refer to section 27, List of Registers.

Table 18.2 Register Configuration

Register Name	Abbreviation	R/W	Initial value	Address	Access Size
I²C bus control register 1	ICCR1	R/W	H'00	H'FFFFCD80	8
I²C bus control register 2	ICCR2	R/W	H'7D	H'FFFFCD81	8
I²C bus mode register	ICMR	R/W	H'38	H'FFFFCD82	8
I²C bus interrupt enable register	ICIER	R/W	H'00	H'FFFFCD83	8
I²C bus status register	ICSR	R/W	H'00	H'FFFFCD84	8
I²C bus slave address register	SAR	R/W	H'00	H'FFFFCD85	8
I²C bus transmit data register	ICDRT	R/W	H'FF	H'FFFFCD86	8
I²C bus receive data register	ICDRR	R	H'FF	H'FFFFCD87	8
NF2CYC register	NF2CYC	R/W	H'00	H'FFFFCD88	8

18.3.1 I²C Bus Control Register 1 (ICCR1)

ICCR1 is an 8-bit readable/writable register that enables or disables the I²C bus interface 2, controls transmission or reception, and selects master or slave mode, transmission or reception, and transfer clock frequency in master mode.

ICCR1 is initialized to H'00 by a power-on reset.

Bit:	7	6	5	4	3	2	1	0
	ICE	RCVD	MST	TRS	CKS[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ICE	0	R/W	I²C Bus Interface 2 Enable 0: SCL and SDA output is disabled. (Input to SCL and SDA is enabled) 1: This bit is enabled for transfer operations. (SCL and SDA pins are bus drive state.)
6	RCVD	0	R/W	Reception Disable When TRS = 0, this bit enables or disables continuous reception without reading of ICDRR. In master receive mode, when ICDRR cannot be read before the rising edge of the 8th clock of SCL, set RCVD to 1 so that data is received in byte units. 0: Enables continuous reception 1: Disables continuous reception
5	MST	0	R/W	Master/Slave Select
4	TRS	0	R/W	Transmit/Receive Select In master mode with the I²C bus format, when arbitration is lost, MST and TRS are both reset by hardware, causing a transition to slave receive mode. Modification of the TRS bit should be made between transfer frames. When seven bits after the start condition is issued in slave receive mode match the slave address set to SAR and the 8th bit is set to 1, TRS is automatically set to 1. If an overrun error occurs in master receive mode with the clock synchronous serial format, MST is cleared and the mode changes to slave receive mode. Operating modes are described below according to MST and TRS combination. When clock synchronous serial format is selected and MST = 1, clock is output. 00: Slave receive mode 01: Slave transmit mode 10: Master receive mode 11: Master transmit mode

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	CKS[3:0]	0000	R/W	Transfer Clock Select 3 to 0 These bits should be set according to the necessary transfer rate (table 18.3) in master mode. In slave mode, these bits should be used to specify the data setup time in transmission mode. The setup time is set to 10 tpcyc when CKS3 = 0 or 20 tpcyc when CKS3 = 1 (tpcyc is one Pϕ cycle).

Table 18.3 Transfer Rate

Bit 3	Bit 2	Bit 1	Bit 0		Transfer Rate					
CKS3	CKS2	CKS1	CKS0	Clock	Pϕ=10 MHz	Pϕ=16 MHz	Pϕ=20 MHz	Pϕ=25 MHz	Pϕ=33 MHz	Pϕ=40 MHz
0	0	0	0	Pϕ/28	357 kHz	571 kHz	714 kHz	893 kHz	1.18 MHz	1.43 MHz
			1	Pϕ/40	250 kHz	400 kHz	500 kHz	625 kHz	825 kHz	1.00 MHz
		1	0	Pϕ/48	208 kHz	333 kHz	417 kHz	521 kHz	688 kHz	833 kHz
			1	Pϕ/64	156 kHz	250 kHz	313 kHz	391 kHz	516 kHz	625 kHz
	1	0	0	Pϕ/80	125 kHz	200 kHz	250 kHz	313 kHz	413 kHz	500 kHz
			1	Pϕ/100	100 kHz	160 kHz	200 kHz	250 kHz	330 kHz	400 kHz
		1	0	Pϕ/112	89.3 kHz	143 kHz	179 kHz	223 kHz	295 kHz	357 kHz
			1	Pϕ/128	78.1 kHz	125 kHz	156 kHz	195 kHz	258 kHz	313 kHz
1	0	0	0	Pϕ/112	89.3 kHz	143 kHz	179 kHz	223 kHz	295 kHz	357 kHz
			1	Pϕ/160	62.5 kHz	100 kHz	125 kHz	156 kHz	206 kHz	250 kHz
		1	0	Pϕ/192	52.1 kHz	83.3 kHz	104 kHz	130 kHz	172 kHz	208 kHz
			1	Pϕ/256	39.1 kHz	62.5 kHz	78.1 kHz	97.7 kHz	129 kHz	156 kHz
	1	0	0	Pϕ/320	31.3 kHz	50.0 kHz	62.5 kHz	78.1 kHz	103 kHz	125 kHz
			1	Pϕ/400	25.0 kHz	40.0 kHz	50.0 kHz	62.5 kHz	82.5 kHz	100 kHz
		1	0	Pϕ/448	22.3 kHz	35.7 kHz	44.6 kHz	55.8 kHz	73.7 kHz	89.3 kHz
			1	Pϕ/512	19.5 kHz	31.3 kHz	39.1 kHz	48.8 kHz	64.5 kHz	78.1 kHz

18.3.2 I²C Bus Control Register 2 (ICCR2)

ICCR2 is an 8-bit readable/writable register that issues start/stop conditions, manipulates the SDA pin, monitors the SCL pin, and controls reset in the control part of the I²C bus interface 2.

ICCR2 is initialized to H'7D by a power-on reset.

Bit:	7	6	5	4	3	2	1	0
	BBSY	SCP	SDAO	SDAOP	SCLO	-	IICRST	-
Initial value:	0	1	1	1	1	1	0	1
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7	BBSY	0	R/W	Bus Busy This bit enables to confirm whether the I²C bus is occupied or released and to issue start/stop conditions in master mode. With the clock synchronous serial format, this bit is always read as 0. With the I²C bus format, this bit is set to 1 when the SDA level changes from high to low under the condition of SCL = high, assuming that the start condition has been issued. This bit is cleared to 0 when the SDA level changes from low to high under the condition of SCL = high, assuming that the stop condition has been issued. To issue a start condition, simultaneously write 1 to BBSY and 0 to SCP. Follow this procedure also when transmitting a repeated start condition. To issue a stop condition, simultaneously write 0 to BBSY and 0 to SCP.
6	SCP	1	R/W	Start/Stop Issue Condition Disable The SCP bit controls the issue of start/stop conditions in master mode. To issue a start condition, simultaneously write 1 to BBSY and 0 to SCP. A repeated start condition is issued in the same way. To issue a stop condition, simultaneously write 0 to BBSY and 0 to SCP. This bit is always read as 1. Even if 1 is written to this bit, the data will not be stored.

Bit	Bit Name	Initial Value	R/W	Description
5	SDAO	1	R/W	SDA Output Value Control This bit is used with SDAOP when modifying output level of SDA. This bit should not be manipulated during transfer. 0: When reading, SDA pin outputs low. When writing, SDA pin is changed to output low. 1: When reading, SDA pin outputs high. When writing, SDA pin is changed to output Hi-Z (outputs high by external pull-up resistance).
4	SDAOP	1	R/W	SDAO Write Protect This bit controls change of output level of the SDA pin by modifying the SDAO bit. To change the output level, clear SDAO and SDAOP to 0 or set SDAO to 1 and clear SDAOP to 0. This bit is always read as 1.
3	SCLO	1	R	SCL Output Level This bit monitors SCL output level. When SCLO is 1, SCL pin outputs high. When SCLO is 0, SCL pin outputs low.
2	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
1	IICRST	0	R/W	IIC Control Part Reset IICRST resets the BC[2:0] bits in the ICMR register and the internal circuits of IIC2. If the MCU hangs because of a communication failure while the I²C bus is operating, the BC[2:0] bits in the ICMR register and the internal circuits of IIC2 can be reset by setting the IICRST bit to 1.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

Notes: When 1 is written to the IICRST bit in ICCR2, the state becomes as follows.

- The SDAO and SCLO bits in ICCR2 are set to 1.
- If the module is in master transmit mode or slave transmit mode, the TDRE bit in ICSR is set to 1.
- Writing to the BBSY, SCP, and SDAO bits in ICCR2 is invalid while a reset is being applied by writing 1 to IICRST.

- Writing 1 to IICRST does not clear the BBSY bit in ICCR2 to 0. However, if the states of the SCL and SDA pins lead to the generation of a stop condition, (rising edge on SDA while SCL is at the high level), the BBSY bit may be cleared to 0 as a result. This can also affect other bits in the same way.
- Data transfer stops while a reset is being applied by writing 1 to IICRST. However, functions for detecting start conditions, stop conditions, and failure in bus contention continue to operate. Signals input to the SCL and SDA pins may alter the states of ICCR1, ICCR2, and ICSR.

18.3.3 I²C Bus Mode Register (ICMR)

ICMR is an 8-bit readable/writable register that selects whether the MSB or LSB is transferred first and selects the transfer bit count.

ICMR is initialized to H'38 by a power-on reset. Bits BC[2:0] are initialized to B'000 by the IICRST bit in ICCR2.

Bit:	7	6	5	4	3	2	1	0
	MLS	-	-	-	BCWP	BC[2:0]		
Initial value:	0	0	1	1	1	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MLS	0	R/W	MSB-First/LSB-First Select 0: MSB-first 1: LSB-first Set this bit to 0 when the I²C bus format is used.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5, 4	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.

Bit	Bit Name	Initial Value	R/W	Description
3	BCWP	1	R/W	BC Write Protect This bit controls the BC[2:0] modifications. When modifying BC[2:0], this bit should be cleared to 0. In clock synchronous serial mode, BC[2:0] should not be modified. 0: When writing, values of BC[2:0] are set. 1: When reading, 1 is always read. When writing, settings of BC[2:0] are invalid.
2 to 0	BC[2:0]	000	R/W	Bit Counter These bits specify the number of bits to be transferred next. When read, the remaining number of transfer bits is indicated. With the I²C bus format, the data is transferred with one addition acknowledge bit. Should be made between transfer frames. If these bits are set to a value other than B'000, the setting should be made while the SCL pin is low. The value returns to B'000 at the end of a data transfer, including the acknowledge bit. These bits are automatically set to B'111 after a stop condition is detected. These bits are cleared by a power-on reset, software standby mode, and module standby mode. These bits are also cleared by setting IICRST of ICCR2 to 1. With the clock synchronous serial format, these bits should not be modified. I²C Bus Format / Clock Synchronous Serial Format 000: 9 bits / 000: 8 bits 001: 2 bits / 001: 1 bit 010: 3 bits / 010: 2 bits 011: 4 bits / 011: 3 bits 100: 5 bits / 100: 4 bits 101: 6 bits / 101: 5 bits 110: 7 bits / 110: 6 bits 111: 8 bits / 111: 7 bits

18.3.4 I²C Bus Interrupt Enable Register (ICIER)

ICIER is an 8-bit readable/writable register that enables or disables interrupt sources and acknowledge bits, sets acknowledge bits to be transferred, and confirms acknowledge bits received.

ICIER is initialized to H'00 by a power-on reset.

Bit:	7	6	5	4	3	2	1	0
	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable When the TDRE bit in ICSR is set to 1 or 0, this bit enables or disables the transmit data empty interrupt (IITXI). 0: Transmit data empty interrupt request (IITXI) is disabled. 1: Transmit data empty interrupt request (IITXI) is enabled.
6	TEIE	0	R/W	Transmit End Interrupt Enable This bit enables or disables the transmit end interrupt (IITEI) at the rising of the ninth clock while the TDRE bit in ICSR is 1. IITEI can be canceled by clearing the TEND bit or the TEIE bit to 0. 0: Transmit end interrupt request (IITEI) is disabled. 1: Transmit end interrupt request (IITEI) is enabled.
5	RIE	0	R/W	Receive Interrupt Enable RIE enables or disables the receive data full interrupt request (IIRXI) when receive data is transferred from ICDRS to ICDRR and the RDRF bit in ICSR is set to 1. IIRXI can be canceled by clearing the RDRF or RIE bit to 0. 0: Receive data full interrupt request (IIRXI) are disabled. 1: Receive data full interrupt request (IIRXI) are enabled.

Bit	Bit Name	Initial Value	R/W	Description
4	NAKIE	0	R/W	NACK Receive Interrupt Enable NAKIE enables or disables the NACK detection and arbitration lost/overrun error interrupt request (IINAKI) when the NACKF or AL/OVE bit in ICSR is set to 1. IINAKI can be canceled by clearing the NACKF, AL/OVE, or NAKIE bit to 0. 0: NACK detection and arbitration lost/overrun error interrupt request (IINAKI) is disabled. 1: NACK detection and arbitration lost/overrun error interrupt request (IINAKI) is enabled.
3	STIE	0	R/W	Stop Condition Detection Interrupt Enable This bit enables or disables the stop condition detection interrupt request (IISTPI) when the STOP bit in ICSR is set. 0: Stop condition detection interrupt request (IISTPI) is disabled. 1: Stop condition detection interrupt request (IISTPI) is enabled.
2	ACKE	0	R/W	Acknowledge Bit Judgment Select 0: The value of the receive acknowledge bit is ignored, and continuous transfer is performed. 1: If the receive acknowledge bit is 1, continuous transfer is halted.
1	ACKBR	0	R	Receive Acknowledge In transmit mode, this bit stores the acknowledge data that are returned by the receive device. This bit cannot be modified. This bit can be canceled by setting the BBSY bit in ICCR2 to 1. 0: Receive acknowledge = 0 1: Receive acknowledge = 1
0	ACKBT	0	R/W	Transmit Acknowledge In receive mode, this bit specifies the bit to be sent at the acknowledge timing. 0: 0 is sent at the acknowledge timing. 1: 1 is sent at the acknowledge timing.

18.3.5 I²C Bus Status Register (ICSR)

ICSR is an 8-bit readable/writable register that confirms interrupt request flags and their status.

ICSR is initialized to H'00 by a power-on reset.

Bit:	7	6	5	4	3	2	1	0
	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Note: * These flags can be cleared only by writing 0 after reading the flag's value as 1.

Bit	Bit Name	Initial Value	R/W	Description
7	TDRE	0	R/(W)*[1]	Transmit Data Register Empty [Clearing conditions] • When 0 is written to TDRE after reading TDRE = 1 • When data is written to ICDRT • DTC is activated by IITXI interrupt and the DISEL bit in MRB of DTC is 0. [Setting conditions] • When data is transferred from ICDRT to ICDRS and ICDRT becomes empty • When TRS is set • When the start condition (including retransmission) is issued • When slave mode is changed from receive mode to transmit mode

Bit	Bit Name	Initial Value	R/W	Description
6	TEND	0	R/(W)*1	Transmit End [Clearing conditions] • When 0 is written to TEND after reading TEND = 1 • When data is written to ICDRT • DTC is activated by IITXI interrupt and the DISEL bit in MRB of DTC is 0. [Setting conditions] • When the ninth clock of SCL rises with the I²C bus format while the TDRE flag is 1 • When the final bit of transmit frame is sent with the clock synchronous serial format
5	RDRF	0	R/(W)*1	Receive Data Register Full [Clearing conditions] • When 0 is written to RDRF after reading RDRF = 1 • When ICDRR is read • DTC is activated by IIRXI interrupt and the DISEL bit in MRB of DTC is 0. [Setting condition] • When a receive data is transferred from ICDRS to ICDRR
4	NACKF	0	R/(W)*1	No Acknowledge Detection Flag*2 [Clearing condition] • When 0 is written to NACKF after reading NACKF = 1 [Setting condition] • When no acknowledge is detected from the receive device in transmission while the ACKE bit in ICIER is 1

Bit	Bit Name	Initial Value	R/W	Description
3	STOP	0	R/(W)*[1]	Stop Condition Detection Flag [Clearing condition] • When 0 is written to STOP after reading STOP = 1 [Setting conditions] • In master mode, when a stop condition is detected after frame transfer • In slave mode, when a stop condition is detected after the slave address in the first byte that came following the detection of a start condition have matched the address set in SAR.
2	AL/OVE	0	R/W	Arbitration Lost Flag/Overrun Error Flag This flag indicates that arbitration was lost in master mode with the I²C bus format and that the final bit has been received while RDRF = 1 with the clock synchronous format. When two or more master devices attempt to seize the bus at nearly the same time, if the I²C bus interface 2 detects data differing from the data it sent, it sets AL to 1 to indicate that the bus has been occupied by another master. [Clearing condition] • When 0 is written to AL/OVE after reading AL/OVE = 1 [Setting conditions] • If the internal SDA and SDA pin disagree at the rise of SCL in master transmit mode • When the SDA pin outputs high in master mode while a start condition is detected • When the final bit is received with the clock synchronous format while RDRF = 1

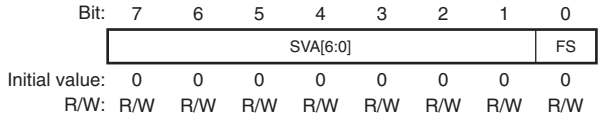
Bit	Bit Name	Initial Value	R/W	Description
1	AAS	0	R/W	Slave Address Recognition Flag In slave receive mode, this flag is set to 1 if the first frame following a start condition matches bits SVA[6:0] in SAR. [Clearing condition] • When 0 is written to AAS after reading AAS=1 [Setting conditions] • When the slave address is detected in slave receive mode • When the general call address is detected in slave receive mode.
0	ADZ	0	R/W	General Call Address Recognition Flag This bit is valid in slave receive mode with the I²C bus format. [Clearing condition] • When 0 is written to ADZ after reading ADZ=1 [Setting condition] • When the general call address is detected in slave receive mode

Notes: 1. These flags can be cleared only by writing 0 after reading the flag's value as 1.
2. When NACKF = 1 is detected, be sure to clear NACKF in the transfer end processing. Until the flag is cleared, next transmission or reception cannot be started.

18.3.6 I²C Bus Slave Address Register (SAR)

SAR is an 8-bit readable/writable register that selects the communications format and sets the slave address. In slave mode with the I²C bus format, if the upper seven bits of SAR match the upper seven bits of the first frame received after a start condition, this module operates as the slave device.

SAR is initialized to H'00 by a power-on reset.



Bit	Bit Name	Initial Value	R/W	Description
7 to 1	SVA[6:0]	All 0	R/W	Slave Address These bits set a unique address in bits SVA6 to SVA0, differing form the addresses of other slave devices connected to the I²C bus.
0	FS	0	R/W	Format Select 0: I²C bus format is selected 1: Clock synchronous serial format is selected

18.3.7 I²C Bus Transmit Data Register (ICDRT)

ICDRT is an 8-bit readable/writable register that stores the transmit data. When ICDRT detects the space in the shift register (ICDRS), it transfers the transmit data which is written in ICDRT to ICDRS and starts transferring data. If the next transfer data is written to ICDRT during transferring data of ICDRS, continuous transfer is possible. If ICDRT is read while the LSB-first setting is enabled (MLS bit in ICMR set to 1), data is read with the MSB–LSB order reversed relative to the value that was written to ICDRT. ICDRT is initialized to H'FF.

Bit:	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

18.3.8 I²C Bus Receive Data Register (ICDRR)

ICDRR is an 8-bit register that stores the receive data. When data of one byte is received, ICDRR transfers the receive data from ICDRS to ICDRR and the next data can be received. ICDRR is a receive-only register, therefore the CPU cannot write to this register.

ICDRR is initialized to H'FF by a power-on reset.

Bit:	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R

18.3.9 I²C Bus Shift Register (ICDRS)

ICDRS is a register that is used to transfer/receive data. In transmission, data is transferred from ICDRT to ICDRS and the data is sent from the SDA pin. In reception, data is transferred from ICDRS to ICDRR after data of one byte is received. This register cannot be read directly from the CPU.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

18.3.10 NF2CYC Register (NF2CYC)

NF2CYC is an 8-bit readable/writable register that selects the range of the noise filtering for the SCL and SDA pins. For details of the noise filter, see section 18.4.7, Noise Filter.

NF2CYC is initialized to H'00 by a power-on reset.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	NF2CYC
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	NF2CYC	0	R/W	Noise Filtering Range Select 0: The noise less than one cycle of the peripheral clock can be filtered out 1: The noise less than two cycles of the peripheral clock can be filtered out

18.4 Operation

The I²C bus interface 2 can communicate either in I²C bus mode or clock synchronous serial mode by setting FS in SAR.

18.4.1 I²C Bus Format

Figure 18.3 shows the I²C bus formats. Figure 18.4 shows the I²C bus timing. The first frame following a start condition always consists of eight bits.

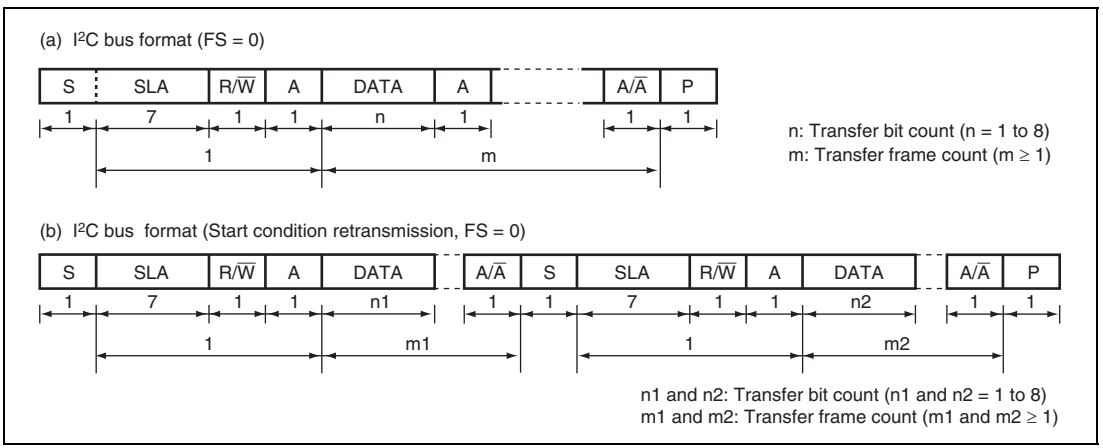


Figure 18.3 I²C Bus Formats

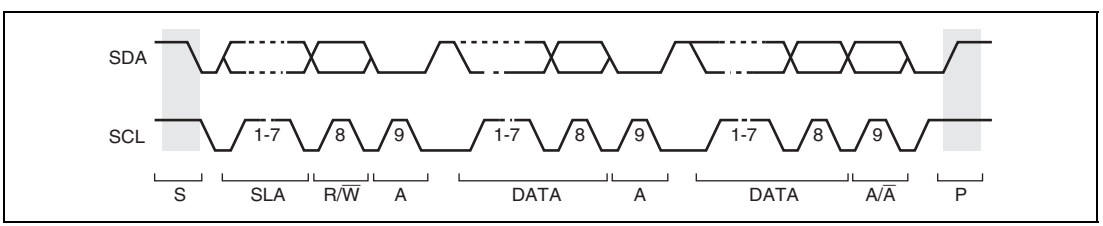


Figure 18.4 I²C Bus Timing

[Legend]
S: Start condition. The master device drives SDA from high to low while SCL is high.
SLA: Slave address
R/W: Indicates the direction of data transfer: from the slave device to the master device when R/W is high, or from the master device to the slave device when R/W is low.
A: Acknowledge. The receive device drives SDA to low.
DATA: Transfer data
P: Stop condition. The master device drives SDA from low to high while SCL is high.

18.4.2 Master Transmit Operation

In master transmit mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal. For master transmit mode operation timing, refer to figures 18.5 and 18.6. The transmission procedure and operations in master transmit mode are described below.

1. Initialize IIC2 (figure 18.7). After initialization, set the ICE bit in ICCR1 to 1.
2. Read the BBSY flag in ICCR2 to confirm that the bus is free. Set the MST and TRS bits in ICCR1 to select master transmit mode. Then, write 1 to BBSY and 0 to SCP. (Start condition issued) This generates the start condition.
3. After issuing the start condition, write the transmit data to ICDRT (the 1st frame consists of data indicating the slave address and the R/$\overline{W}$ bit). At this time, TDRE is automatically cleared to 0, and data is transferred from ICDRT to ICDRS. TDRE is set again.
4. When transmission of first frame data is completed, TEND in ICSR is set to 1 at the rise of the 9th transmit clock pulse. Read the ACKBR bit in ICIER, and confirm that the slave device has been selected. Then, write second frame data to ICDRT. When ACKBR is 1, the slave device has not been acknowledged, so issue the stop condition. To issue the stop condition, write 0 to BBSY and to SCP after waiting for 0 to be read from SCLO in ICCR2. SCL is fixed low until the transmit data is prepared or the stop condition is issued.
5. The transmit data after the second frame is written to ICDRT every time TDRE is set.
6. Write the last transmit data to ICDRT, then wait until the TEND bit is set to 1 (the end of the last byte of data transmission). Alternately, wait for a NACK (NACKF in ICSR = 1) from the receive device while the ACKE bit in ICIER is set to 1. Once TEND or NACKF is set to 1, wait for 0 to be read from SCLO in ICCR2. Next, clear TEND and NACK, and issue the stop condition.
7. When the STOP bit in ICSR is set to 1, clear MST and TRS to return to slave receive mode.

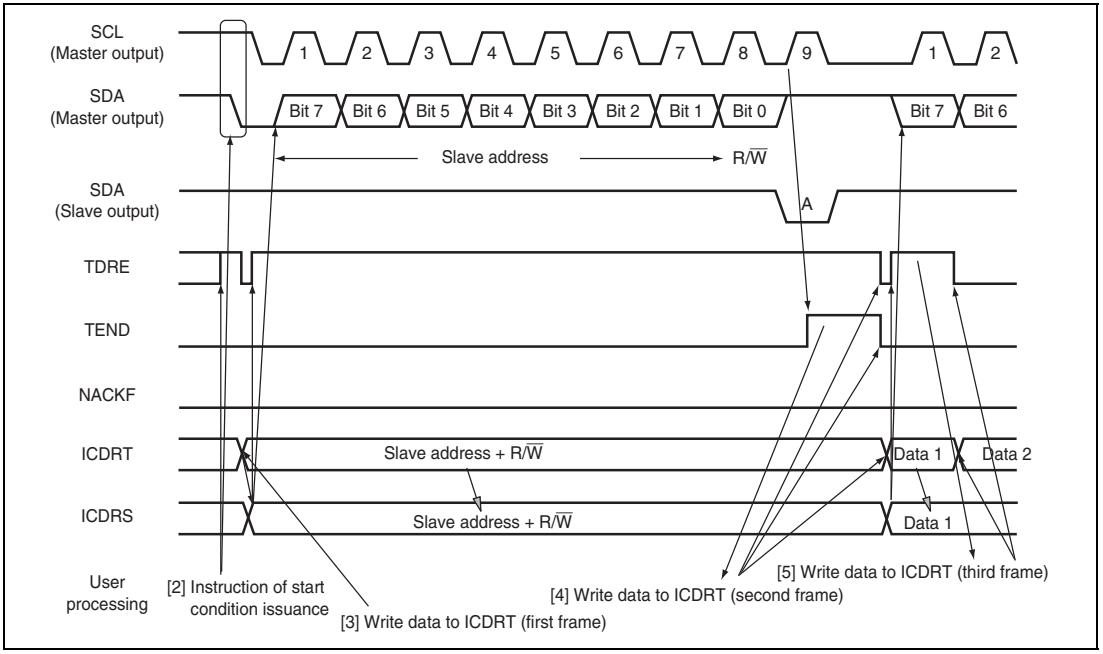


Figure 18.5 Master Transmit Mode Operation Timing (1)

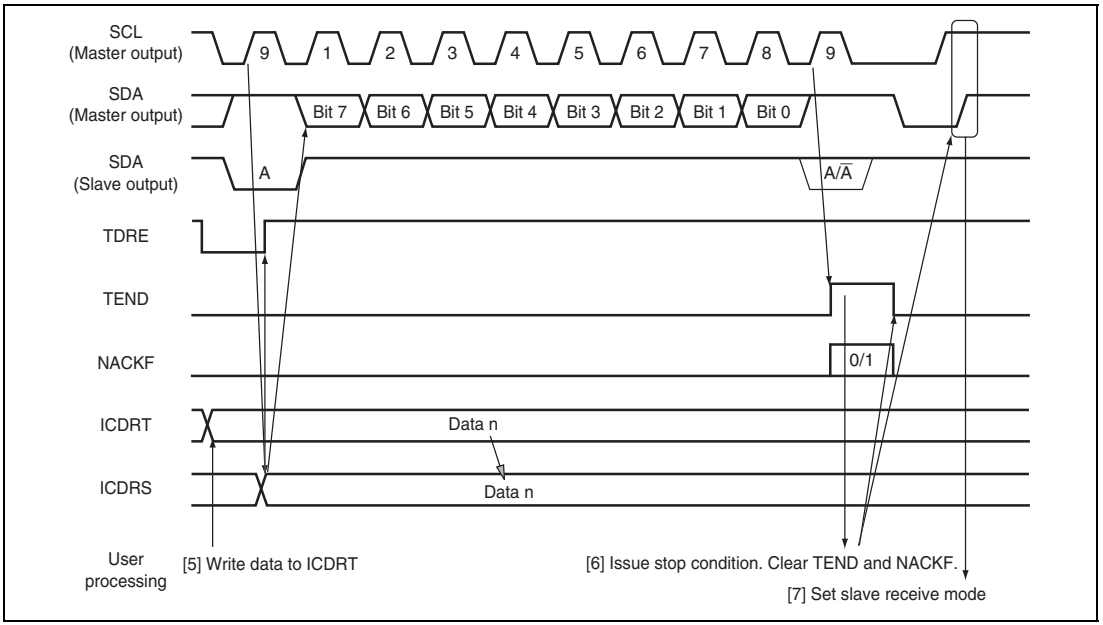
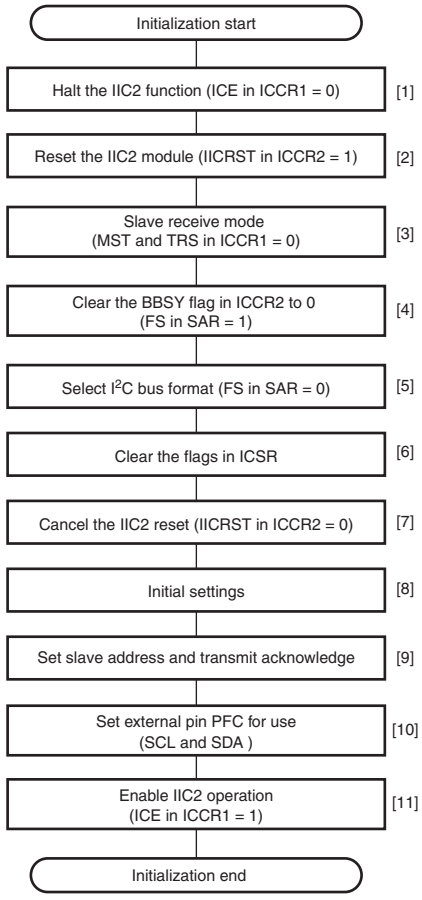


Figure 18.6 Master Transmit Mode Operation Timing (2)



[1] Write 0 to the ICE bit in ICCR1 to halt functioning of the IIC2.

[2] Write 1 to the IICRST bit in ICCR2 to reset some registers and the contol part of the IIC2 module.

[3] Write 0 to the MST and TRS bits in ICCR1 to switch the operating mode to slave receiver mode.

[4] Write 1 to the FS bit in SAR to clear the BBSY flag in ICCR2 to 0.

[5] Write 0 to the FS bit in the SAR to select I²C bus format.

[6] Clear the flags in ICSR to 0.

[7] Write 0 to the IICRST bit to release the IIC2 module from the reset state.

[8] Initialize IIC2 registers (ICCR1, ICCR2, ICMR, ICIER, and NF2CYC). Clear the ICE, MST, and TRS bits in the ICCR1 to 0.

[9] Specify the slave address in the SVA[6:0] bit field in the SAR. Clear ACKBT in the ICIER to 0.

[10] Set external pin PFC for use. Set PFC while SCL and SDA are high-level.

[11] Write 1 to the ICE bit in the ICCR1 to enable transfer operation.

Note: For the procedure for initializing IIC2 when communication is in progress, see 18.4.8, Using the IICRST Bit to Reset I²C Bus Interface 2.

Figure 18.7 Flowchart of Initialization of I²C Bus Interface 2

18.4.3 Master Receive Operation

In master receive mode, the master device outputs the receive clock, receives data from the slave device, and returns an acknowledge signal. For master receive mode operation timing, refer to figures 18.8 and 18.9. The reception procedure and operations in master receive mode are shown below. For operation up to transmission of the 1st frame (slave address + R/$\overline{W}$), see 18.4.2, Master Transmit Operation.

1. After clearing TEND in ICSR to 0, clear TRS in ICCR1 to 0 to switch from master transmit mode to master receive mode. Then clear TDRE to 0.
2. When ICDRR is read (dummy data read), reception is started, and the receive clock is output, and data received, in synchronization with the internal clock. The master device outputs the level specified by ACKBT in ICIER to SDA, at the 9th receive clock pulse.
3. After the reception of first frame data is completed, the RDRF bit in ICSR is set to 1 at the rise of 9th receive clock pulse. At this time, the receive data is read by reading ICDRR, and RDRF is cleared to 0. The continuous reception is performed by reading ICDRR every time RDRF is set. If reading of ICDRR cannot take place before the rising edge of the 8th clock pulse of SCL, set RCVD in ICCR1 to 1 and perform communication one byte at a time.
4. If next frame is the last receive data, set the RCVD bit in ICCR1 and ACKBT bit in ICIER to 1 before reading ICDRR. This enables the issuance of the stop condition after the next reception.
5. When RDRF is set to 1 at the 9th receive clock pulse, wait until 0 is read from SCLO in ICCR2. Then issue the stop condition.
6. When STOP is set to 1, read the final receive data from ICDRR.
7. Clear RCVD and MST to 0 to return to slave receive mode.

Note: If only one byte is received, read ICDRR (dummy-read) after the RCVD bit in ICCR1 is set.

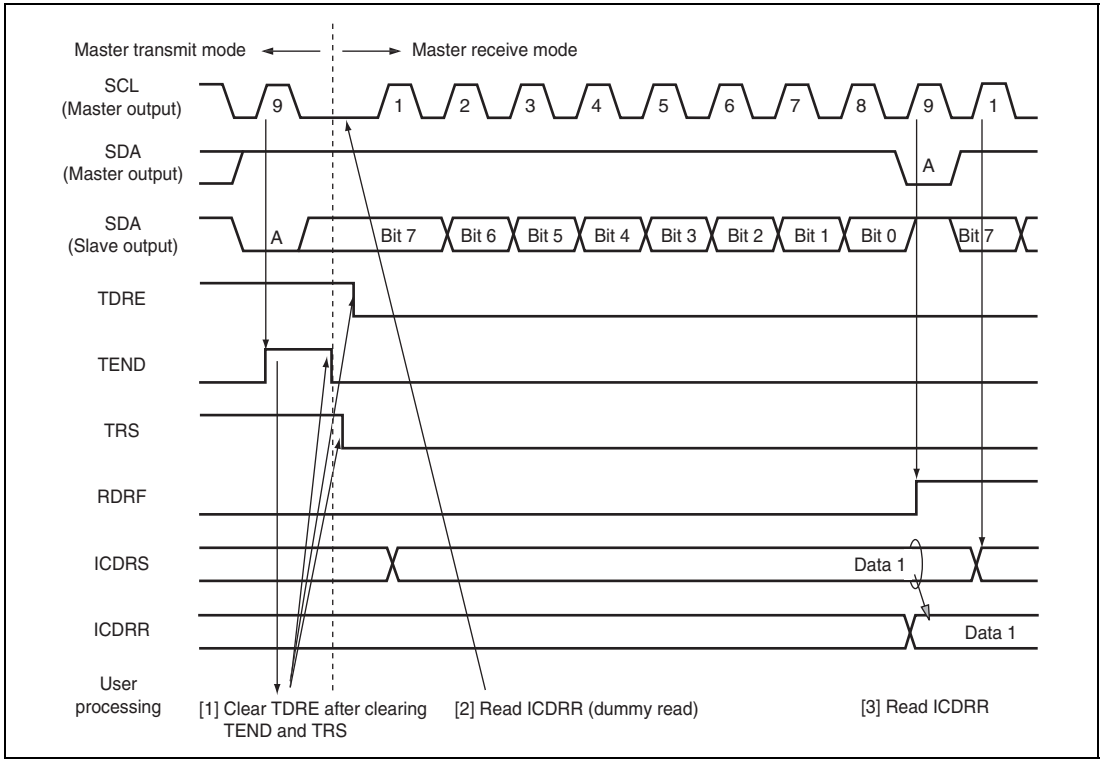


Figure 18.8 Master Receive Mode Operation Timing (1)

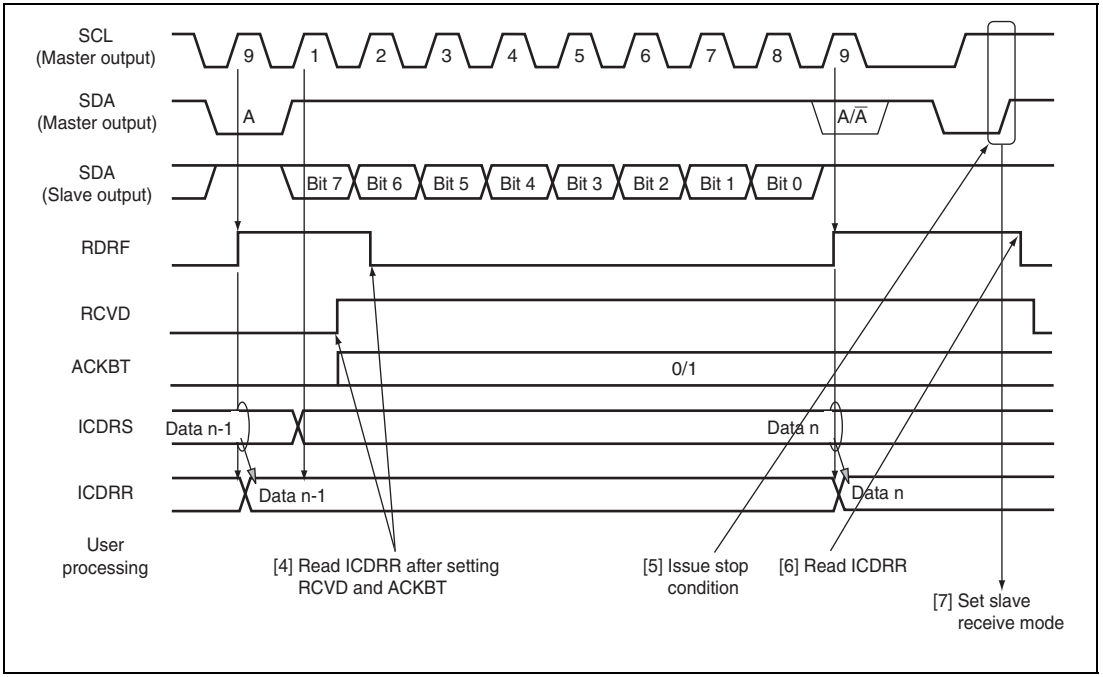


Figure 18.9 Master Receive Mode Operation Timing (2)

18.4.4 Slave Transmit Operation

In slave transmit mode, the slave device outputs the transmit data, while the master device outputs the receive clock and returns an acknowledge signal. For slave transmit mode operation timing, refer to figures 18.10 and 18.11.

The transmission procedure and operations in slave transmit mode are described below.

1. Initialize IIC2 (figure 18.7). After initialization, set the ICE bit in ICCR1 to 1. Set the MST and TRS bits in ICCR1 to select slave receive mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start condition, the slave device outputs the level specified by ACKBT in ICIER to SDA, at the rise of the 9th clock pulse. At this time, if the 8th bit data (R/$\overline{W}$) is high, the TRS bit in ICCR1 and the TDRE bit in ICSR are set to 1, and the mode changes to slave transmit mode automatically. If the 8th bit of data is low-level, operation continues in slave receive mode.
3. Write the transmit data to ICDRT. At this time, TDRE is automatically cleared to 0, the data is transferred from ICDRT to ICDRS, and TDRE is set to 1 once again. Write the subsequent transmit data to ICDRT each time TDRE is set to 1.
4. Write the last transmit data to ICDRT, wait until TEND is set to 1 (final frame transmit-end). Alternately, wait for a NACK from the receive device (NACKF in ICSR = 1) with ACKE in ICIER set to 1.
5. Clear TRS for the end processing, and read ICDRR (dummy read). SCL is free.
6. Clear TDRE, TEND, and NACKF.

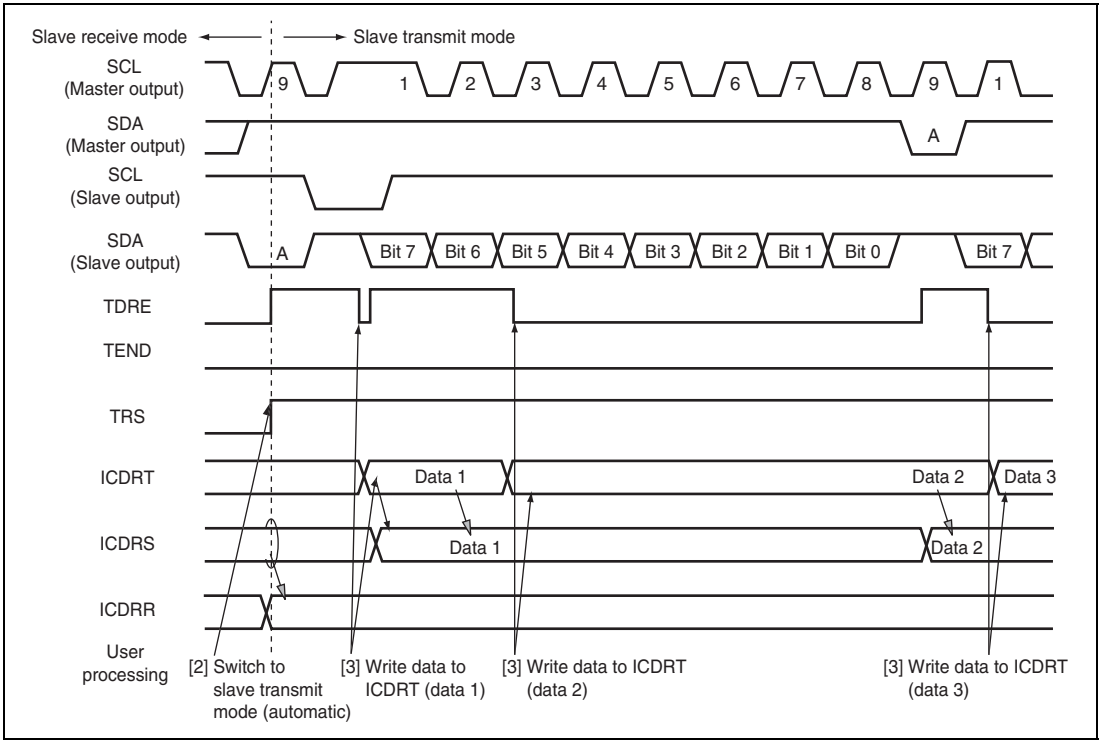


Figure 18.10 Slave Transmit Mode Operation Timing (1)

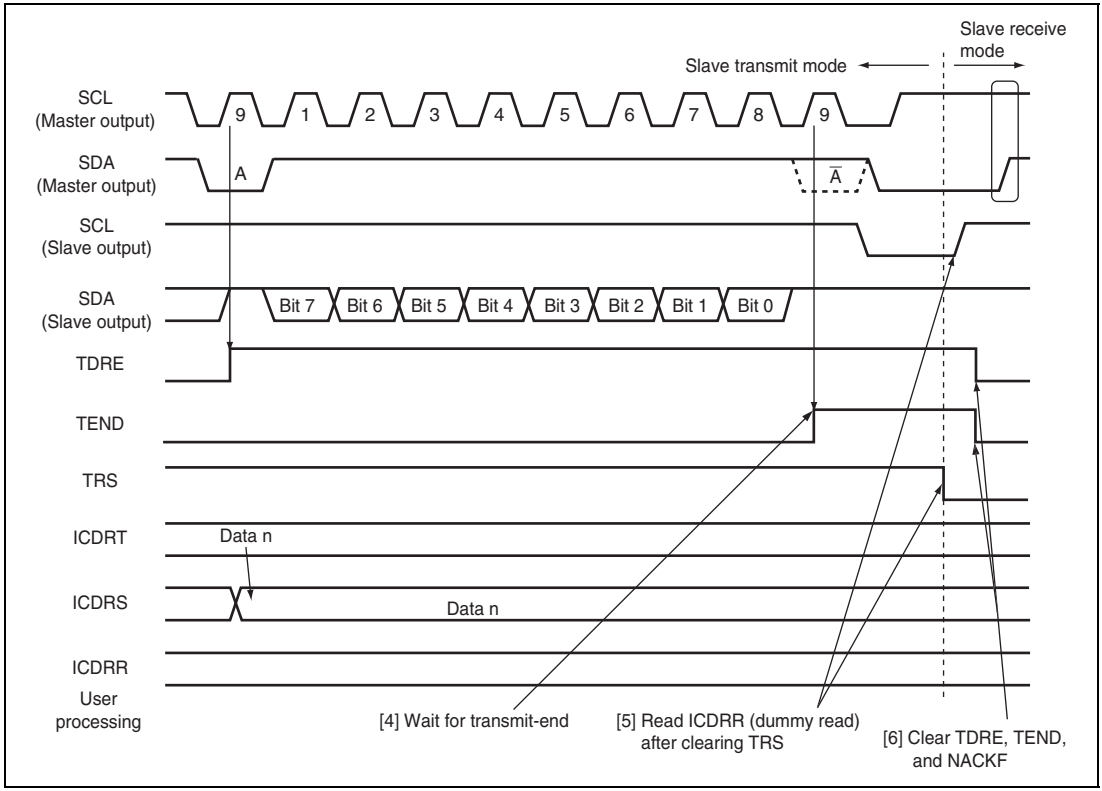


Figure 18.11 Slave Transmit Mode Operation Timing (2)

18.4.5 Slave Receive Operation

In slave receive mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal. For slave receive mode operation timing, refer to figures 18.12 and 18.13.

The reception procedure and operations in slave receive mode are described below. For operation up to reception of the 1st frame (slave address + R/$\overline{W}$), see 18.4.4, Slave Transmit Operation.

1. Perform a dummy read of ICDRR. (The read data indicates slave address + R/$\overline{W}$, and is therefore unnecessary.) The slave device outputs to SDA the level indicated by the setting of ACKBT in ICIER at the 9th clock pulse of the receive clock.
2. After reception of one frame of data finishes, RDRF in ICSR is set to 1 at the rising edge of the 9th receive clock pulse. At this time, the received data can be read by reading ICDRR, and RDRF is cleared to 0 simultaneously. Continuous receive operation can be accomplished by reading ICDRR each time RDRF is set to 1.
   If 8th receive clock pulse falls while RDRF is 1, SCL is fixed low until ICDRR is read.
3. If the next receive operation is the final frame, set ACKBT in ICIER to 1 before reading ICDRR.
4. When RDRF in ICSR is set to 1, read the final receive data from ICDRR.

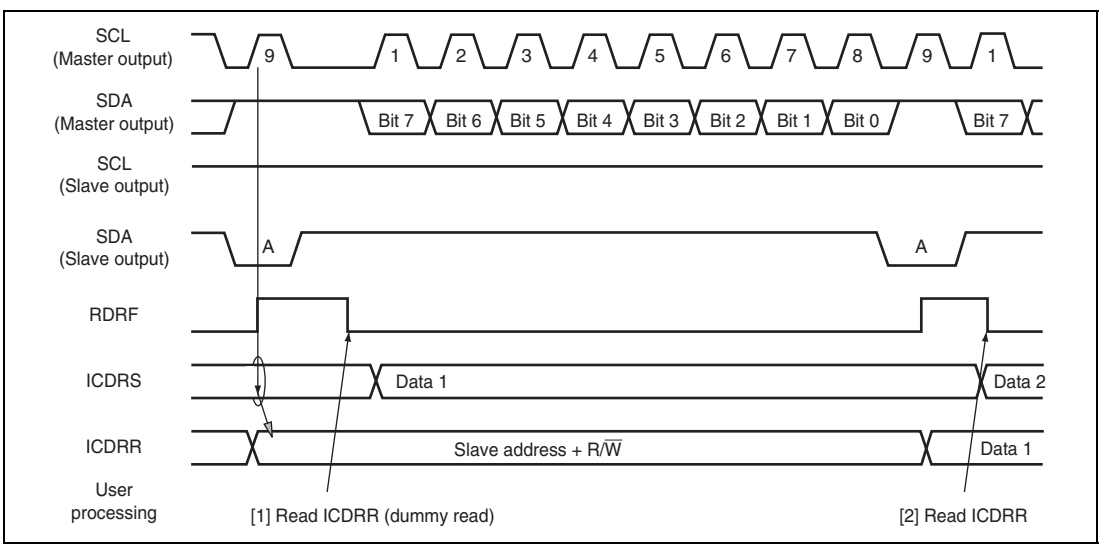


Figure 18.12 Slave Receive Mode Operation Timing (1)

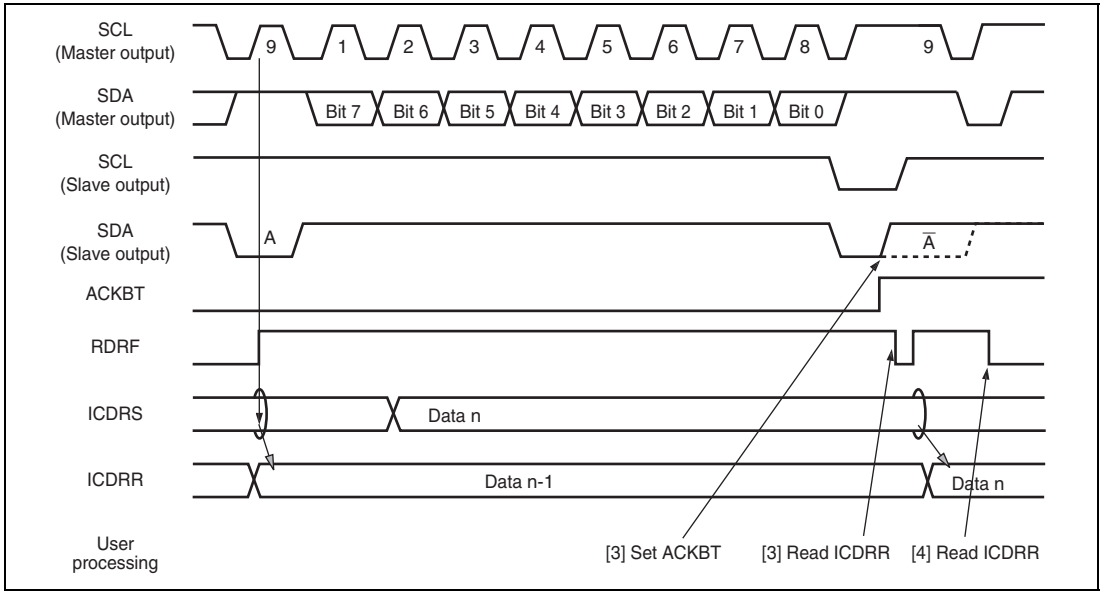


Figure 18.13 Slave Receive Mode Operation Timing (2)

18.4.6 Clock Synchronous Serial Format

This module can be operated with the clock synchronous serial format, by setting the FS bit in SAR to 1. When the MST bit in ICCR1 is 1, the transfer clock output from SCL is selected. When MST is 0, the external clock input is selected.

(1) Data Transfer Format

Figure 18.14 shows the clock synchronous serial transfer format.

The transfer data is output from the fall to the fall of the SCL clock, and the data at the rising edge of the SCL clock is guaranteed. The MLS bit in ICMR sets the order of data transfer, in either the MSB first or LSB first. The output level of SDA can be changed during the transfer wait, by the SDAO bit in ICCR2.

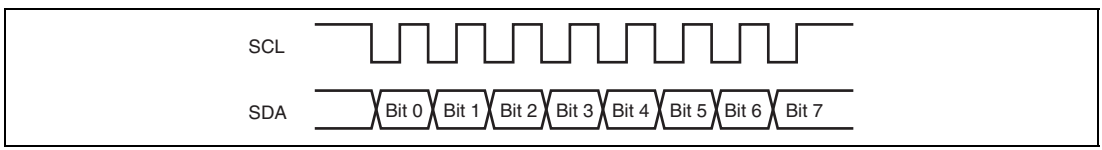


Figure 18.14 Clock Synchronous Serial Transfer Format (LSB-First Operation)

(2) Transmit Operation

In transmit mode, transmit data is output from SDA, in synchronization with the fall of the transfer clock. The transfer clock is output when MST in ICCR1 is 1, and is input when MST is 0. For transmit mode operation timing, refer to figure 18.15. The transmission procedure and operations in transmit mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set the MST and CKS[3:0] bits in ICCR1. (Initial setting)
2. Set the TRS bit in ICCR1 to select the transmit mode. Then, TDRE in ICSR is set.
3. Confirm that TDRE has been set. Then, write the transmit data to ICDRT. The data is transferred from ICDRT to ICDRS, and TDRE is set automatically. The continuous transmission is performed by writing data to ICDRT every time TDRE is set. When changing from transmit mode to receive mode, clear TRS while TDRE is 1.

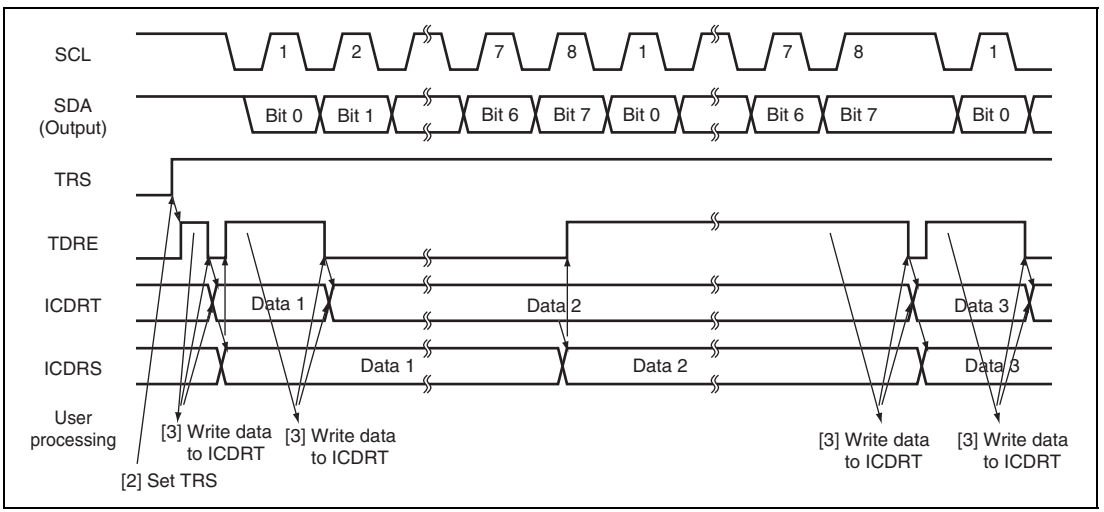


Figure 18.15 Transmit Mode Operation Timing (LSB-First Operation)

(3) Receive Operation

In receive mode, data is latched at the rise of the transfer clock. The transfer clock is output when MST in ICCR1 is 1, and is input when MST is 0. For receive mode operation timing, refer to figure 18.16. The reception procedure and operations in receive mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set the MST and CKS[3:0] bits in ICCR1. (Initial setting)
2. When the transfer clock is output, set MST to 1 to start outputting the receive clock.
3. When the receive operation is completed, data is transferred from ICDRS to ICDRR and RDRF in ICSR is set. When MST = 1, the next byte can be received, so the clock is continually output. The continuous reception is performed by reading ICDRR every time RDRF is set. When the 8th clock is risen while RDRF is 1, the overrun is detected and AL/OVE in ICSR is set. At this time, the previous reception data is retained in ICDRR.
4. To stop receiving when MST = 1, set RCVD in ICCR1 to 1, then read ICDRR. Then, SCL is fixed high after receiving the next byte data.

Notes: Follow the steps below to receive only one byte with MST=1 specified. See figure 18.17 for the operation timing.

1. Set the ICE bit in ICCR1 to 1. Set bits CKS[3:0] in ICCR1. (Initial setting)
2. Set MST=1 while the RCVD bit in ICCR1 is 0. This causes the receive clock to be output.
3. Check if the BC[2] bit in ICMR is set to 1 and then set the RCVD bit in ICCR1 to 1. This causes the SCL to be fixed to the high level after outputting one byte of the receive clock.

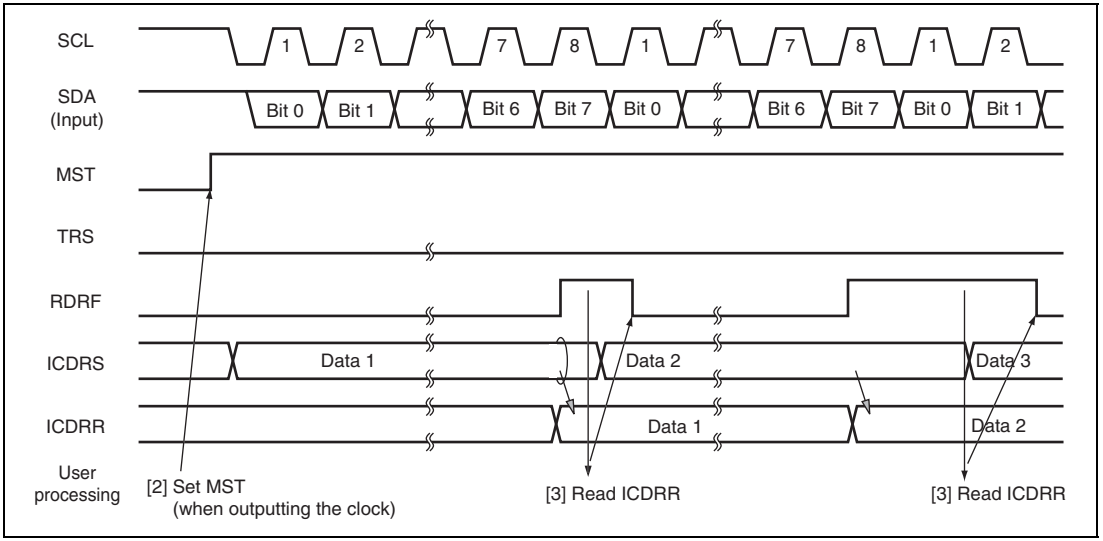


Figure 18.16 Receive Mode Operation Timing (LSB-First Operation)

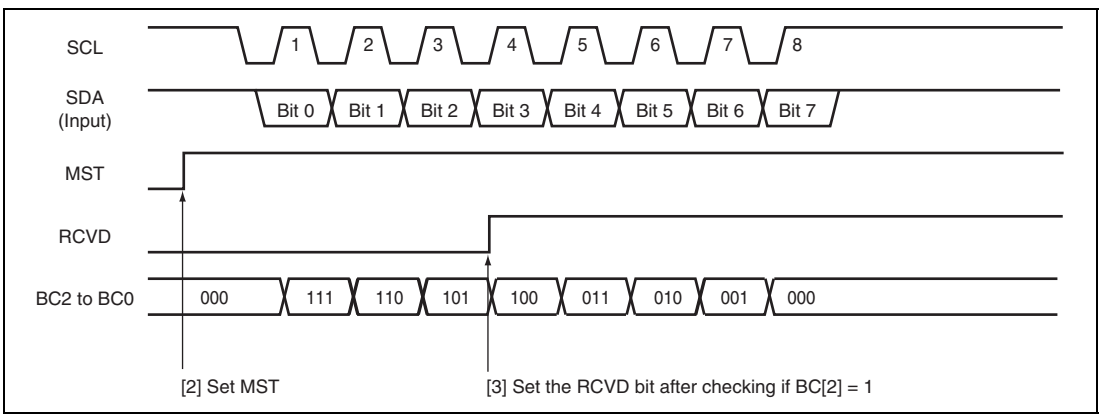


Figure 18.17 Operation Timing For Receiving One Byte (LSB-First Operation)

18.4.7 Noise Filter

The logic levels at the SCL and SDA pins are routed through noise filters before being latched internally. Figure 18.18 shows a block diagram of the noise filter circuit.

The noise filter consists of three cascaded latches and a match detector. The SCL (or SDA) input signal is sampled on the peripheral clock. When NF2CYC is set to 0, this signal is not passed forward to the next circuit unless the outputs of both latches agree. When NF2CYC is set to 1, this signal is not passed forward to the next circuit unless the outputs of three latches agree. If they do not agree, the previous value is held.

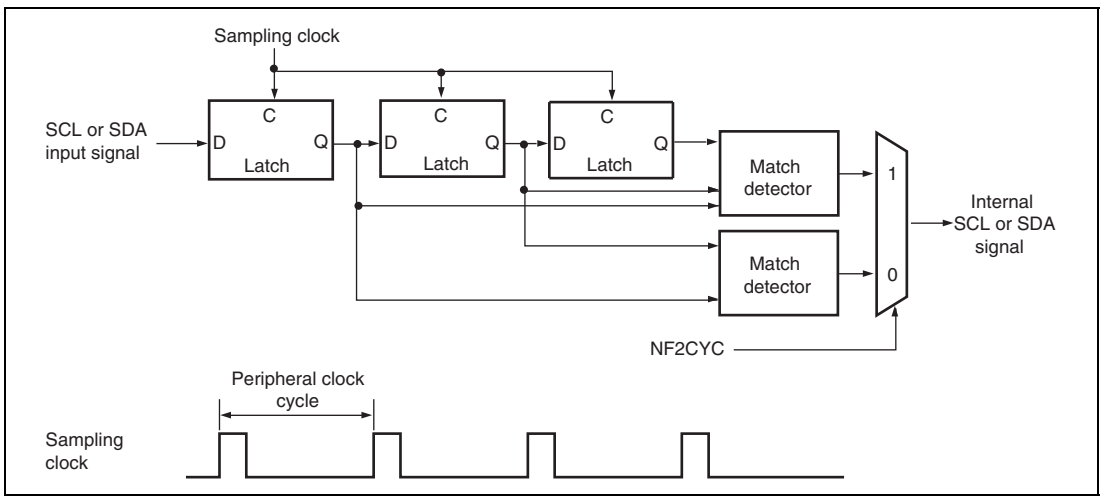


Figure 18.18 Block Diagram of Noise Filter

18.4.8 Using the IICRST Bit to Reset I²C Bus Interface 2

Some registers and the control part for I²C of the I²C bus interface 2 can be reset by writing 1 to the IICRST bit in ICCR2. Figure 18.19 shows an example of the sequence for resetting the I²C bus interface 2 by using the IICRST bit.

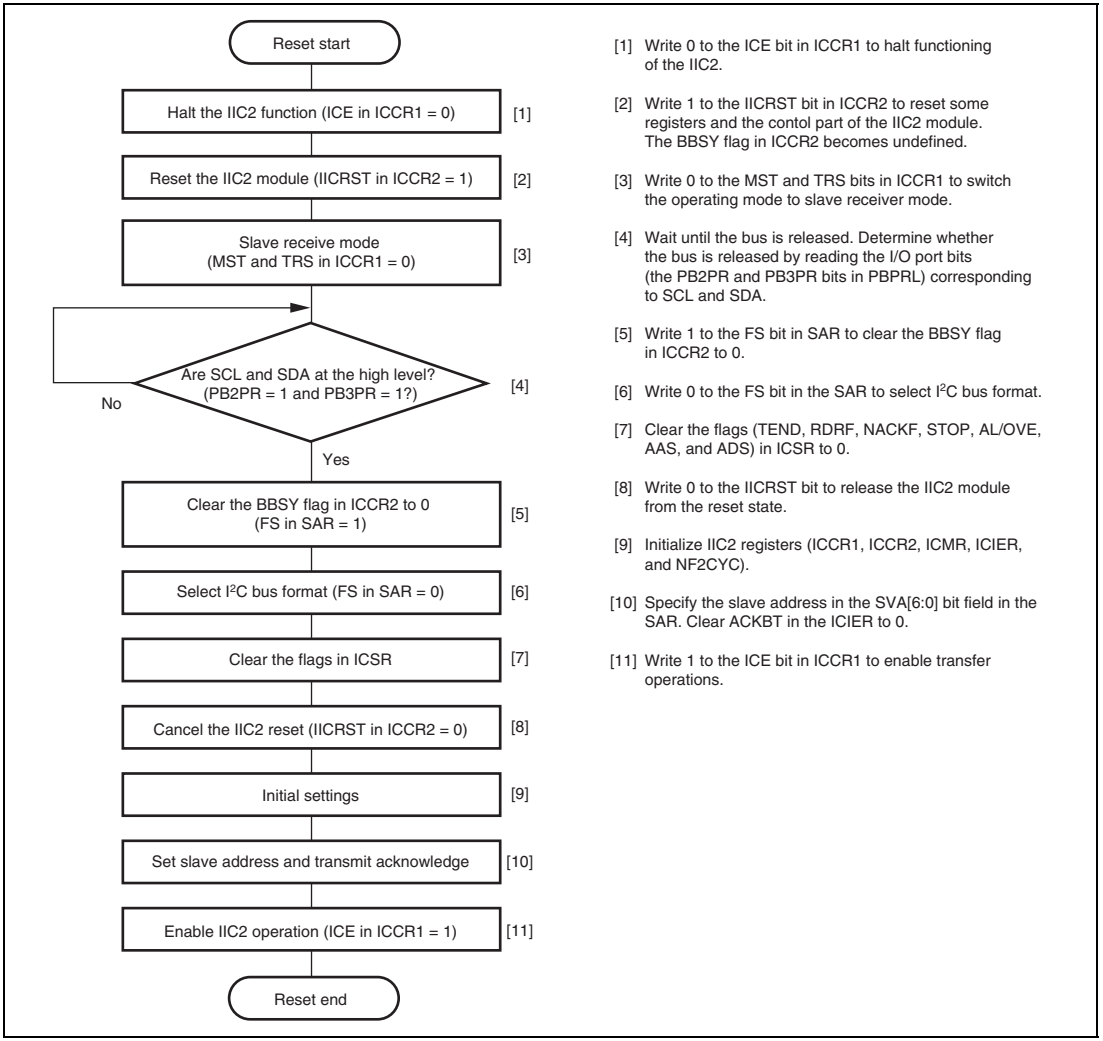


Figure 18.19 Sequence for Using the IICRST Bit to Reset I²C Bus Interface 2

18.4.9 Example of Use

Flowcharts in respective modes that use the I²C bus interface 2 are shown in figures 18.20 to 18.23.

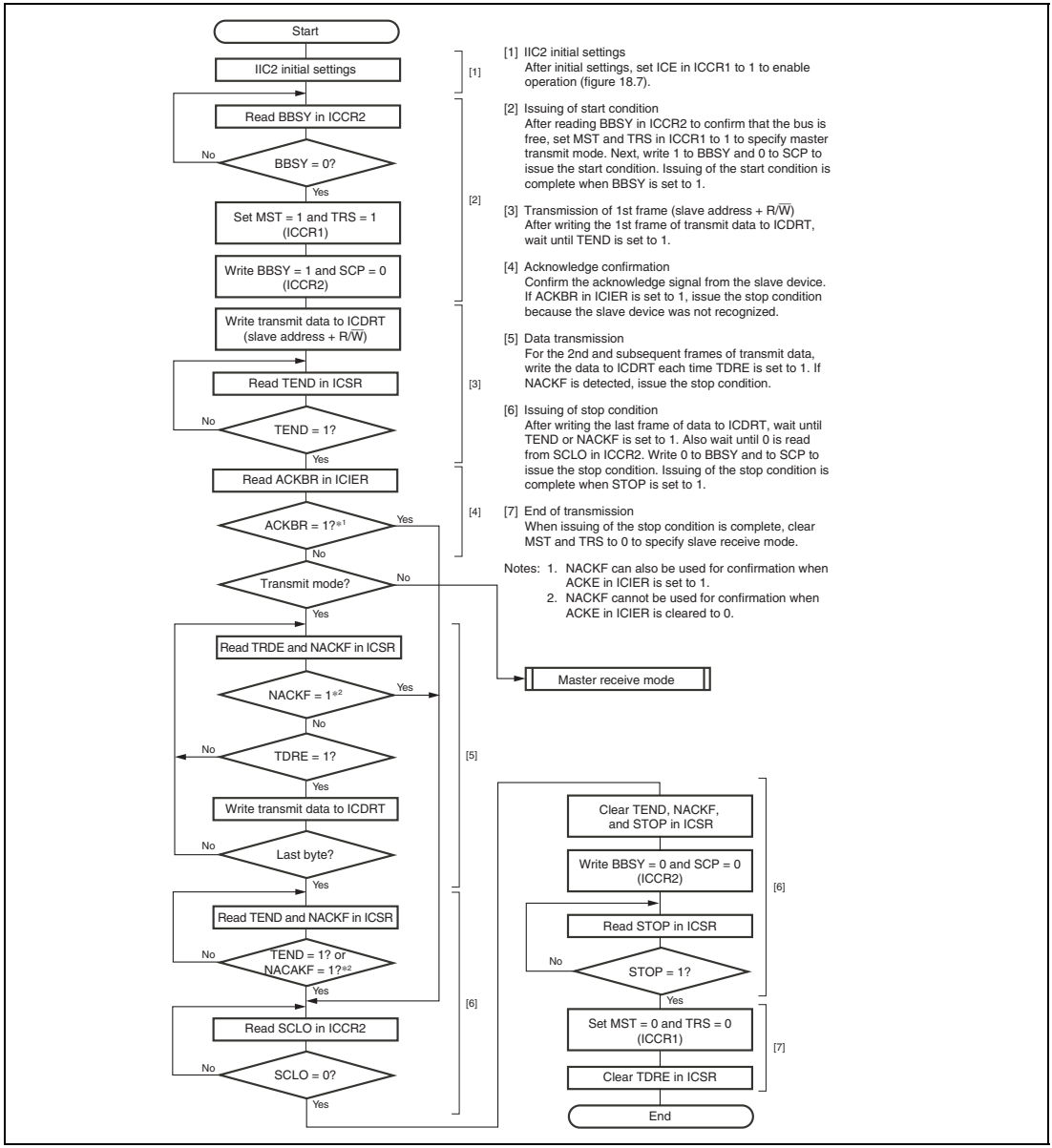
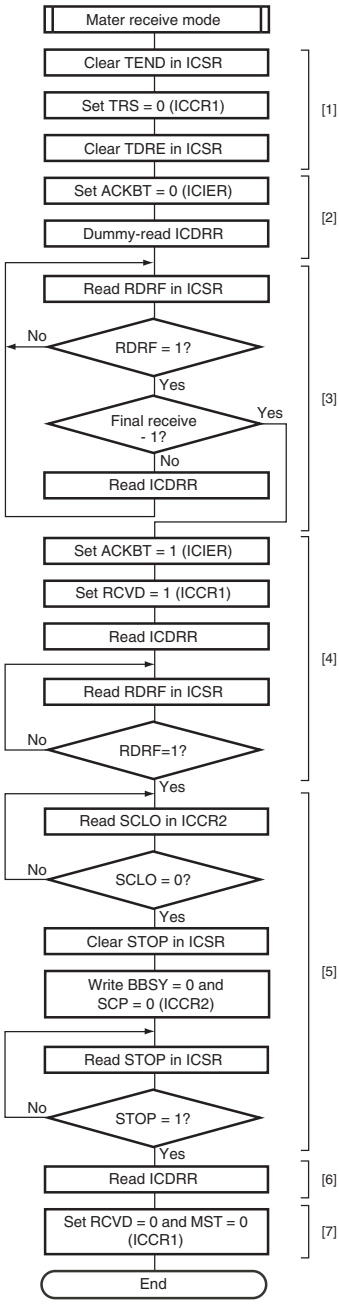


Figure 18.20 Sample Flowchart for Master Transmit Mode



[1] Specification of master receive mode*1
After clearing TEND in ICSR to 0, clear TRS in ICCR1 to 0 to switch from master transmit mode to master receive mode. Then clear TDRE to 0.
See figure 18.20 for the processing up to transmission of the 1st frame (slave address + R/$\overline{W}$).

[2] Start of reception*1*2
After setting ACKBT in ICIER to send an acknowledge signal to the transmit device, perform a dummy read of ICDRR. The dummy read causes clock output to start, and reception starts.

[3] Data reception*2
When reception of one frame finishes, RDRF in ICSR is set to 1. The received data can be read by reading ICDRR. Continuous receive operation can be accomplished by reading ICDRR each time RDRF is set to 1.

[4] Reception of final frame
If the next receive operation is the final frame, set RCVD and ACKBT to 1 before reading ICDRR. SCL is limited to low-level when receive finishes.

[5] Issuing of stop condition
When RDRF is set to 1, wait until 0 is read from SCLO in ICCR2. Next, write 0 to BBSY and to SCP to issue the stop condition. Issuing of the stop condition is complete when STOP in ICSR is set to 1.

[6] Reading of final receive data
When STOP is set to 1, read the final receive data from ICDRR.

[7] End of reception
When issuing of the stop condition is complete, clear RCVD and MST to 0 to specify slave receive mode.

Notes: 1. Ensure that no interrupts occur during the processing of steps [1] and [2].
2. For one-byte reception, after processing step [1], steps [2] and [3] are omitted and processing jumps to step [4]. The reading of ICDRR in step [4] is a dummy read.

Figure 18.21 Sample Flowchart for Master Receive Mode

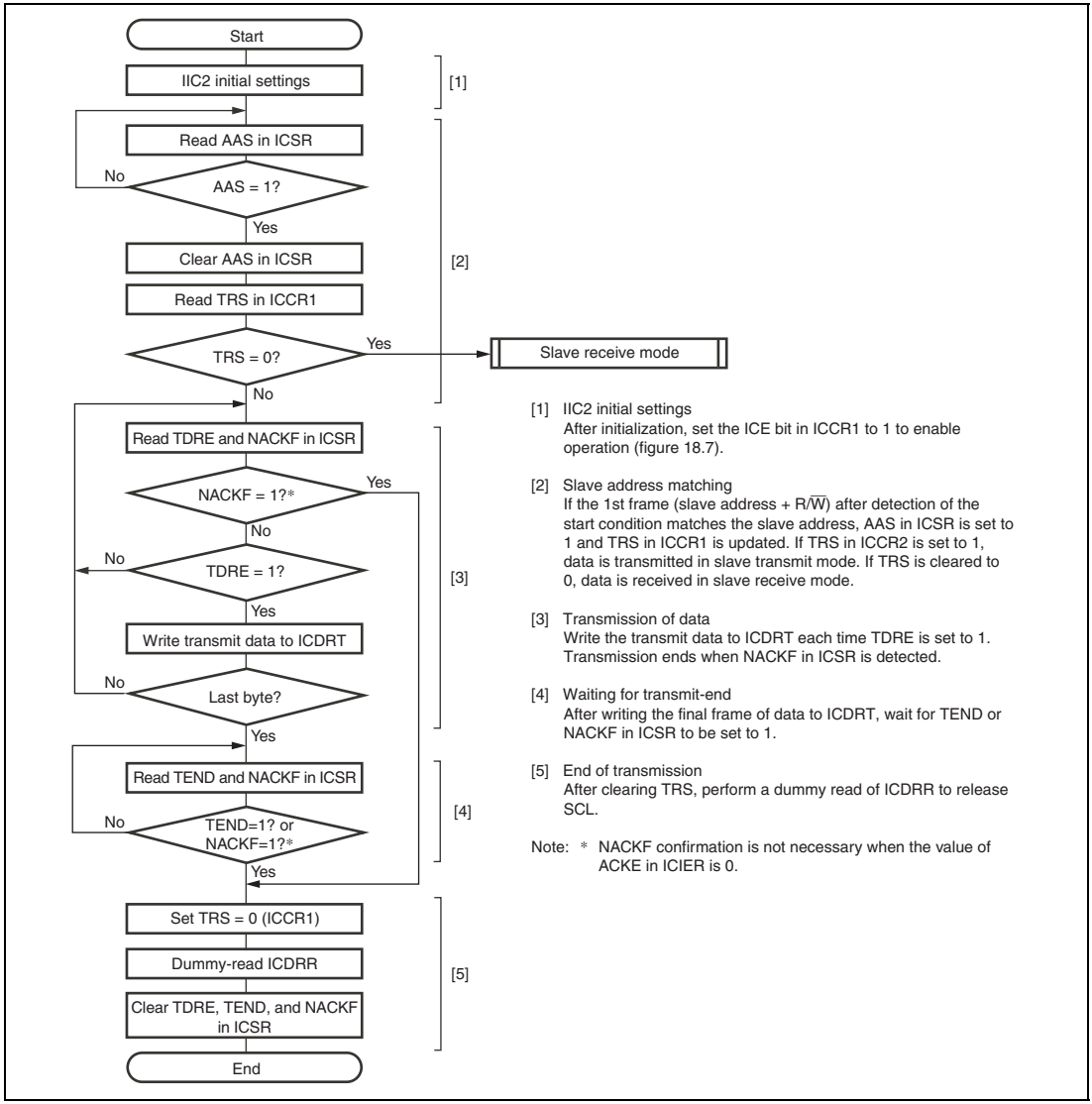


Figure 18.22 Sample Flowchart for Slave Transmit Mode

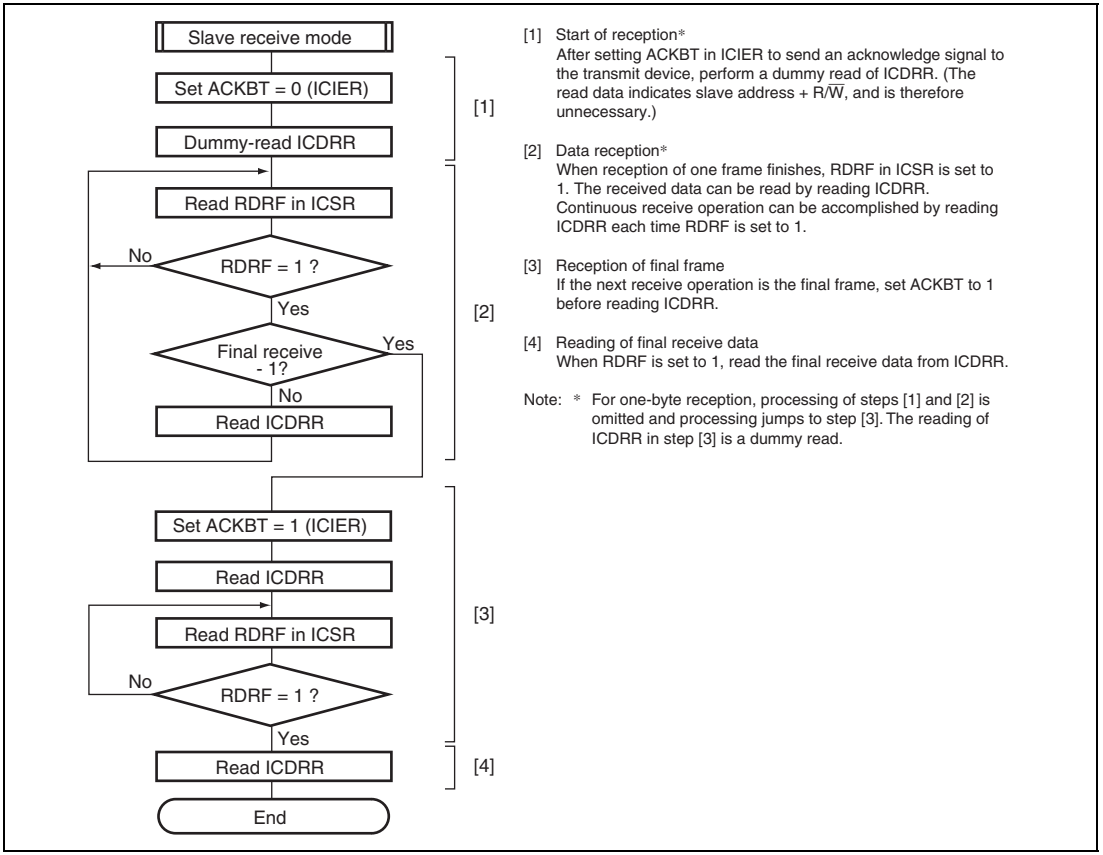


Figure 18.23 Sample Flowchart for Slave Receive Mode

18.5 Interrupt Sources and DTC

The IIC2 module has six interrupt sources; these are for the transmit data empty interrupt request (IITXI), transmit end interrupt request (IITEI), receive data full interrupt request (IIRXI), stop condition detection interrupt request (IISTPI), and NACK detection, arbitration lost or overrun error interrupt request (IINAKI).

The interrupt sources and their order of priority are listed in table 18.4. The TIE, RIE, TEIE, NAKIE, and STIE bits in the I²C bus interrupt enable register (ICIER) enable or disable the various interrupt sources. Furthermore, each of the corresponding interrupt requests is independently conveyed to the interrupt controller.

A IITXI interrupt request is generated when the TDRE flag in the I²C bus status register (ICSR) is set to 1. A IITXI interrupt request can activate the data transfer controller (DTC) to handle data transfer. When the DTC is activated to handle transfer, if the DISEL bit of the DTC is 0 and the value of the transfer counter is non-zero, the TDRE flag is automatically cleared to 0 once the DTC has written to the ICDRT, so a IITXI interrupt request is not sent to the CPU. If the DISEL bit of the DTC is 0 and the value of the transfer counter is 0, or if the DISEL bit is 1, writing to ICDRT does not lead to automatic clearing of the TDRE flag, so a IITXI interrupt request is subsequently generated for the CPU.

A IIRXI interrupt request is generated when the RDRF flag in ICSR is set to 1. A IIRXI interrupt request can activate the DTC to handle data transfer. When the DTC is activated to handle transfer, if the DISEL bit of the DTC is 0 and the value of the transfer counter is non-zero, the RDRF flag is automatically cleared to 0 once the DTC has read from ICDRR, so a IIRXI interrupt request is not sent to the CPU. If the DISEL bit of the DTC is 0 and the value of the transfer counter is 0, or if the DISEL bit is 1, Reading from ICDRR does not lead to automatic clearing of the RDRF flag, so a IIRXI interrupt request is subsequently generated for the CPU.

A IINAKI interrupt request is generated when the NACKF or AL/OVE flag in ICSR is set to 1. A IINAKI interrupt request is not capable of activating the DTC. Setting of the NACKF to 1 only leads to a IINAKI interrupt request when communications are in I²C format.

A IISTPI interrupt request is generated when the STOP flag in ICSR is set to 1. A IISTPI interrupt request is not capable of activating the DTC. Setting of the STOP flag to 1 only leads to a IISTPI interrupt request when communications are in I²C format.

A IITEI interrupt request is generated when the TEND flag in ICSR is set to 1. A IITEI interrupt request is not capable of activating the DTC.

Caution is required because writing data for transmission to ICDRT automatically clears TDRE and TEND and reading from ICDRR automatically clears RDRF. In particular, if TDRE is again set at the same time as data for transmission are written to ICDRT, an extra byte may be transmitted when TDRE is then cleared.

Table 18.4 Interrupt Requests

Interrupt Request	Abbreviation	Interrupt Condition	I²C Bus Format	Clocked Synchronous Serial Format	DTC Activation	Priority
NACK detection	IINAKI*	{(NACKF = 1) + (AL/OVE = 1)} • (NAKIE = 1)	√	—	—	High
Arbitration lost/ overrun error			√	√	—	
Transmit end	IITEI	(TEND = 1) • (TEIE = 1)	√	√	—	
Stop condition detection	IISTPI	(STOP = 1) • (STIE = 1)	√	—	—	
Transmit data empty	IITXI	(TDRE = 1) • (TIE = 1)	√	√	√	
Receive data full	IIRXI	(RDRF = 1) • (RIE = 1)	√	√	√	Low

Note: * In the case of IINAKI, the IPR bit in the INTC, which determines the priority, is different. Depending on the setting of the IPR bit, the priority may be lower than that of IIRXI.

18.6 Operation Using the DTC

In the I²C bus format, since the slave device or the direction of transfer is selected by the slave address or the R/W bit, and the acknowledge bit may indicate the end of reception or reception of the final frame, the continuous transfer of data by the DTC must be performed combined with the CPU processing by the interrupt.

Table 18.5 shows some example of processing using the DTC. These examples assume that the number of transfer data bytes is known in slave mode.

Table 18.5 Example of Processing Using DTC

Item	Master Transmit Mode	Master Receive Mode	Slave Transmit Mode	Slave Receive Mode
Slave address + R/$\overline{W}$ bit transmit/receive	Transmitted by DTC* (ICDRT writing)	Transmitted by CPU (ICDRT writing)	Received by CPU (ICDRR reading)	Received by CPU (ICDRR reading)
Dummy data read	—	Processed by CPU (ICDRR reading)	—	Processed by CPU (ICDRR reading)
Main data transmit/receive	Transmitted by DTC (ICDRT writing)	Received by DTC (ICDRR reading)	Transmitted by DTC (ICDRT writing)	Received by DTC (ICDRR reading)
Last frame processing	Not necessary	Received by CPU (ICDRR reading)	Not necessary	Received by CPU (ICDRR reading)
DTC transfer data frame count setting	Transmission: Actual data count + 1 (+1 is required for the slave address + R/$\overline{W}$ bit transfer)	Reception: Actual data count – 1 (–1 is required for processing of the last frame)	Transmission; Actual data count	Reception: Actual data count – 1 (–1 is required for processing of the last frame)

Note: * After issuing a start condition (writing 1 to BBSY and 0 to SCP), enable DTC transfers.

18.7 Bit Synchronous Circuit

In master mode, this module has a possibility that high level period may be short in the two states described below.

- When SCL is driven to low by the slave device
- When the rising speed of SCL is lowered by the load of the SCL line (load capacitance or pull-up resistance)

Therefore, it monitors SCL and communicates by bit with synchronization.

Figure 18.24 shows the timing of the bit synchronous circuit and table 18.6 shows the time when SCL output changes from low to Hi-Z then SCL is monitored.

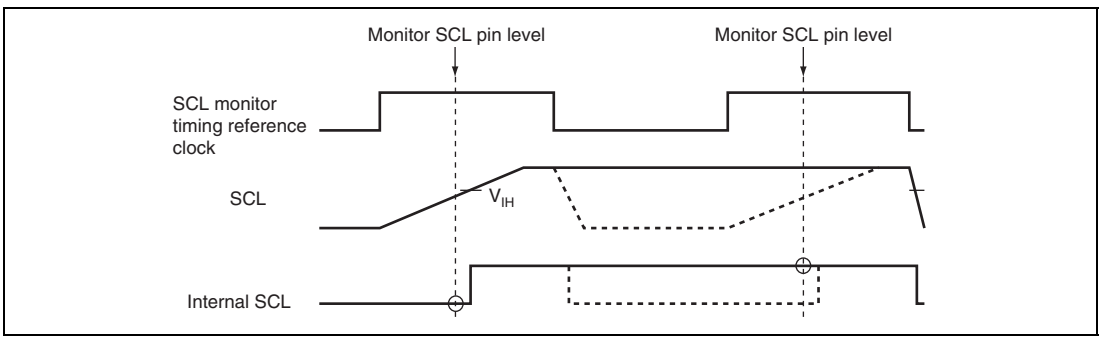


Figure 18.24 The Timing of the Bit Synchronous Circuit

Table 18.6 Time for Monitoring SCL

CKS3	CKS2	NF2CYC	Time for Monitoring SCL*1
0	0	0	6.5 t_{pcyc}*2
		1	5.5 t_{pcyc}*2
	1	0	18.5 t_{pcyc}*2
		1	17.5 t_{pcyc}*2
1	0	0	16.5 t_{pcyc}*2
		1	15.5 t_{pcyc}*2
	1	0	40.5 t_{pcyc}*2
		1	39.5 t_{pcyc}*2

Notes: 1. SCL pin level is monitored after "time for monitoring SCL" has elapsed from the rising edge of the reference clock for monitoring SCL.
2. t_{pcyc} indicates the period of the peripheral clock.

18.8 Usage Note

18.8.1 Module Standby Mode Setting

The I²C2 operation can be disabled or enabled using the standby control register. The initial setting is for I²C2 operation to be halted. Access to registers is enabled by clearing module standby mode. For details, refer to section 26, Power-Down Modes.

18.8.2 Issuance of Stop Condition and Repeated Start Condition

A stop condition or repeated start condition should be issued after the fall of the ninth clock pulse is recognized. The fall of the ninth clock pulse can be recognized by checking the SCLO bit in the I²C bus control register 2 (ICCR2).

18.8.3 Issuance of a Start Condition and Stop Condition in Sequence

Do not issue a start condition and stop condition in sequence. If a start condition and stop condition are to be issued in sequence, be sure to transmit a slave address before issuing the stop condition.

18.8.4 Settings for Multi-Master Operation

1. Transfer rate setting

   In multi-master operation, specify a transfer rate of at least 1/1.8 of the fastest transfer rate among the other masters. For example, when the fastest of the other masters is at 400 kbps, the IIC transfer rate of this LSI must be specified as 223 kbps (= 400/1.8) or a higher rate.
2. MST and TRS bits in ICCR1

   In multi-master operation, use the MOV instruction to set the MST and TRS bits in ICCR1.
3. Loss of arbitration

   When arbitration is lost, check whether the MST and TRS bits in ICCR1 are 0. If the MST and TRS bits in ICCR1 have been set to a value other than 0, clear the bits to 0.

18.8.5 Reading ICDRR in Master Receive Mode

In master receive mode, read ICDRR before the rising edge of the 8th clock of SCL. If ICDRR cannot be read before the rising edge of the 8th clock so that the next round of reception proceeds with the RDRF bit in ICSR set to 1, the 8the clock is fixed low and the 9th clock is output.

If ICDRR cannot be read before the rising edge of the 8th clock of SCL, set the RCVD bit in ICCR1 to 1 so that transfer proceeds in byte units.

18.8.6 Access to ICE and IICRST Bits during I²C Bus Operations

Writing 0 to the ICE bit in ICCR1 or 1 to the IICRST bit in ICCR2 while this LSI is in any of the following states (1 to 4) causes the BBSY flag in ICCR2 and the STOP flag in ICSR to become undefined.

1. This module is the I²C bus master in master transmit mode (MST = 1 and TRS = 1 in ICCR1).
2. This module is the I²C bus master in master receive mode (MST = 1 and TRS = 0 in ICCR1).
3. This module is transmitting data in slave transmit mode (MST = 0 and TRS = 1 in ICCR1).
4. This module is transmitting acknowledge signals in slave receive mode (MST = 0 and TRS = 0 in ICCR1).

Executing any of the following procedures releases the BBSY flag in ICCR2 from the undefined state.

- Input a start condition (falling edge of SDA while SCL is at the high level) to set the BBSY flag to 1.
- Input a stop condition (rising edge of SDA while SCL is at the high level) to clear the BBSY flag to 0.
- If the module is in master transmit mode, issue a start condition by writing 1 and 0 to the BBSY flag and the SCP bit in ICCR2, respectively, while SCL and SDA are at the high level. The BBSY flag is set to 1 on output of the start condition (falling edge of SDA while SCL is at the high level).
- With the module in master transmit or master receive mode, SDA at the low level, and no other device holding SCL at the low level, issue a stop condition by writing 0 to the BBSY flag and the SCP bit in ICCR2. The BBSY flag is cleared to 0 on output of the stop condition (rising edge of SDA while SCL is at the high level).
- Writing 1 to the FS bit in SAR clears the BBST flag to 0.

18.8.7 Using the IICRST Bit to Initialize the Registers

- Writing 1 to the IICRST bit in the ICCR2 register sets the SDAO and SCLO bits in the ICCR2 register to 1.
- Writing 1 to the IICRST bit in master transmit mode or slave transmit mode sets the TDRE flag in ICSR to 1.
- During a reset due to the IICRST bit being set to 1, writing to the BBSY flag and the SCP and SDAO bits is invalid.
- The BBSY flag in the ICCR2 register is not cleared to 0 even when 1 is written to the IICRST bit. However, in some cases the conditions of pins SCL and SDA may generate a stop condition (rising edge of SDA while SCL is at the high level), causing the BBSY flag to be cleared as a result.
- Data transmit and receive stops during a reset due to the IICRST bit being set to 1. However, the function that detects start conditions, stop conditions, and loss in bus contention continues to operate. The state of the ICCR1, ICCR2, and ICSR registers may be updated by signals input on SCL and SDA.

18.8.8 Operation of I²C Bus Interface 2 while ICE = 0

Writing 0 to the ICE bit in ICCR1 disables output on SCL and SDA. However, input on SCL and SDA remains valid. This module operates in accord with the signals input on SCL and SDA.

18.8.9 Notes on Switching from Master Transmit Mode to Master Receive Mode

If TRS is cleared to 0 to switch from master transmit mode to master receive mode before the falling edge of the 9th clock cycle in master transmit mode, the IIC2 module outputs the reception clock in synchronization with the internal clock whether ICDRR is read (dummy read) or not.

Countermeasures for this phenomenon are listed below.

1. Design the timing so that ICDRR is read (dummy read) before the 9th cycle of the reception clock when switching from master transmit mode to master receive mode.
2. Clear TRS to 0 on or after the falling edge of the 9th cycle of the transmission clock when switching from master transmit mode to master receive mode.

For countermeasure 2., confirm that the SCLO bit (SCL monitor flag) in ICCR2 has been set to 0 (the SCL pin outputs the low level) before clearing TRS to 0 on or after the falling edge of the 9th cycle of the transmission clock.

18.8.10 DTC Transfers Using the IIRXI Interrupt as the Source

If the handler of the IIRXI interrupt generated after a DTC transfer clears RDRF in ICSR to 0 or reads ICDRR, it may not be possible to receive the next data.

As a workaround for this issue, perform both 1 and 2 below:

1. For DTC transfers that use the IIRXI interrupt as the source, clear to 0 the DISEL bit in the DTC's MRB register.
2. The processing routine for the IIRXI interrupt must clear the RDRF flag in ICSR by the time of the rising edge of the ninth clock cycle of the next frame to be transferred.

18.8.11 DTC Transfers Using the IITXI Interrupt as the Source

If the handler of the IITXI interrupt generated after a DTC transfer clears TDRE in ICSR to 0, unintended data transmission may occur. Also, writing transmit data to ICDRT may disrupt transmission of the last data transferred by the DTC.

As a workaround for this issue, perform all of 1 to 4 below:

1. For DTC transfers that use the IITXI interrupt as the source, clear to 0 the DISEL bit in the DTC's MRB register.
2. In transmit mode (TRS = 1), do not access ICSR to clear TDRE to 0.
3. Ensure that the IITXI interrupt handler clears TIE in ICIER to 0 to disable the IITXI interrupt. After TIE is cleared, read ICIER before ending handling of the IITXI interrupt.
4. When writing transmit data to ICDRT after a DTC transfer finishes, make sure to set TEND to 1 before writing to ICDRT. (After TEND is set to 1, enable DTC transfers using the IITXI interrupt as the source.)

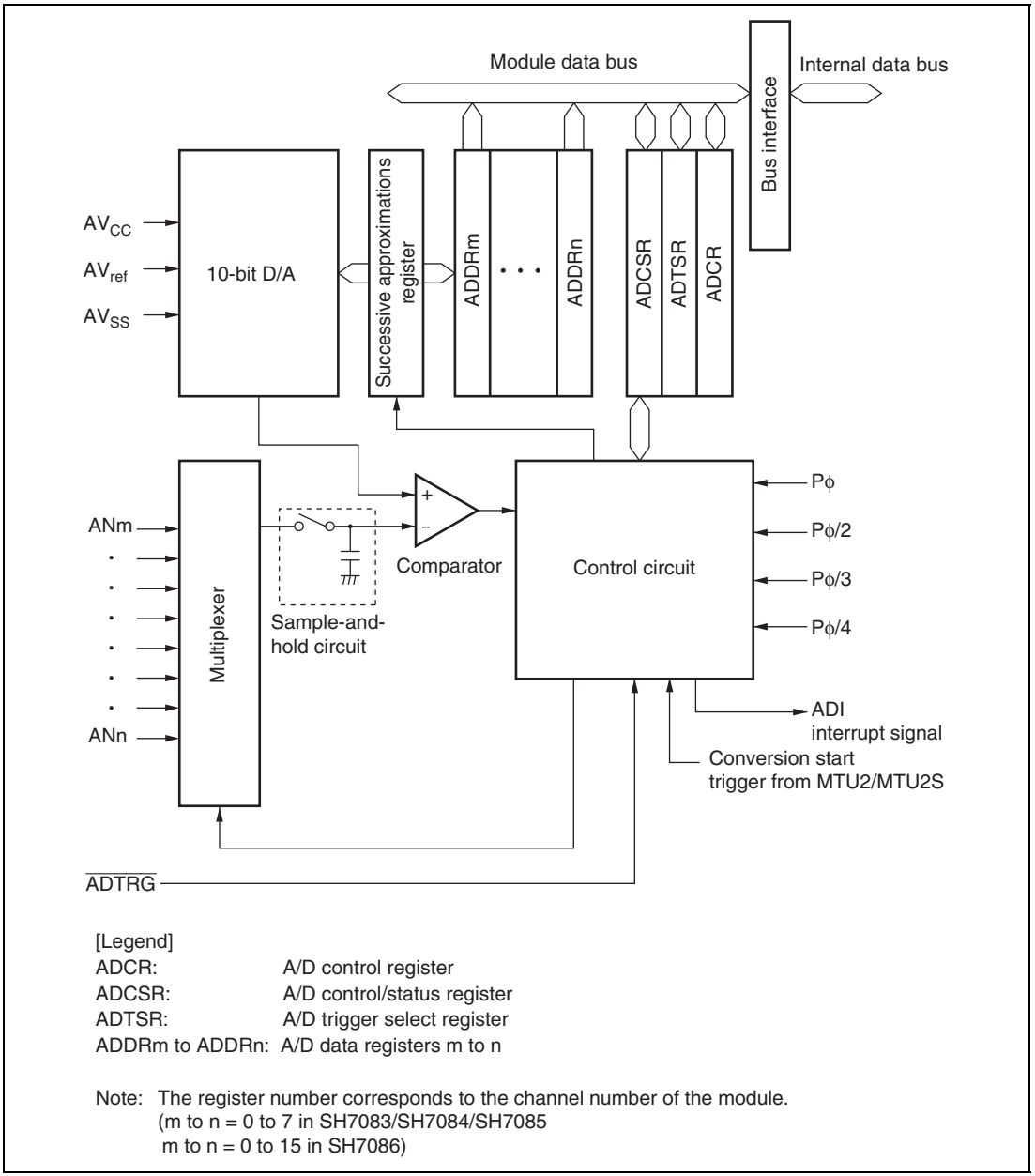
Section 19 A/D Converter (ADC)

This LSI includes a successive approximation type 10-bit A/D converter.

19.1 Features

- 10-bit resolution
- Input channels
  - 8 channels in SH7083/SH7084/SH7085 (two independent A/D conversion modules)
  - 16 channels in SH7086 (three independent A/D conversion modules)
- Conversion time: 2.0 μs per channel (operation when Pϕ = 25 MHz)
- Three operating modes
  - Single mode: Single-channel A/D conversion
  - Continuous scan mode: Repetitive A/D conversion on up to four channels in SH7083/SH7084/SH7085 and up to eight channels in SH7086
  - Single-cycle scan mode: Continuous A/D conversion on up to four channels in SH7083/SH7084/SH7085 and up to eight channels in SH7086
- Data registers
  - Conversion results are held in a 16-bit data register for each channel
- Sample-and-hold function
- Three methods for conversion start
  - Software
  - Conversion start trigger from multifunction timer pulse unit 2 (MTU2) or multifunction timer pulse unit 2S (MTU2S)
  - External trigger signal
- Interrupt request
  - An A/D conversion end interrupt request (ADI) can be generated
- Module standby mode can be set

Figure 19.1 shows a block diagram of the A/D converter.



[Legend]
ADCR: A/D control register
ADCSR: A/D control/status register
ADTSR: A/D trigger select register
ADDRm to ADDRn: A/D data registers m to n

Note: The register number corresponds to the channel number of the module.
(m to n = 0 to 7 in SH7083/SH7084/SH7085
m to n = 0 to 15 in SH7086)

Figure 19.1 Block Diagram of A/D Converter (for One Module)

19.2 Input/Output Pins

Table 19.1 summarizes the input pins used by the A/D converter. The SH7083/SH7084/SH7085 has two A/D conversion modules and the SH7086 has three A/D conversion modules, each of which can be operated independently. The input channels of A/D modules 0 and 1 are divided into two channel groups.

Table 19.1 Pin Configuration

Module Type	Pin Name	I/O	Function		Product Classification SH7083	SH7084	SH7085	SH7086
Common	AV_{CC}	Input	Analog block power supply and reference voltage		√	√	√	√
	AV_{ref}	Input	A/D conversion reference voltage		√	—	√	√
	AV_{SS}	Input	Analog block ground and reference voltage		√	√	√	√
	$\overline{ADTRG}$	Input	A/D external trigger input pin		√	√	√	√
A/D module 0 (A/D_0)	AN0	Input	Analog input pin 0	Group 0	√	√	√	√
	AN1	Input	Analog input pin 1		√	√	√	√
	AN2	Input	Analog input pin 2	Group 1	√	√	√	√
	AN3	Input	Analog input pin 3		√	√	√	√
A/D module 1 (A/D_1)	AN4	Input	Analog input pin 4	Group 0	√	√	√	√
	AN5	Input	Analog input pin 5		√	√	√	√
	AN6	Input	Analog input pin 6	Group 1	√	√	√	√
	AN7	Input	Analog input pin 7		√	√	√	√
A/D module 2 (A/D_2)	AN8	Input	Analog input pin 8		—	—	—	√
	AN9	Input	Analog input pin 9		—	—	—	√
	AN10	Input	Analog input pin 10		—	—	—	√
	AN11	Input	Analog input pin 11		—	—	—	√
	AN12	Input	Analog input pin 12		—	—	—	√
	AN13	Input	Analog input pin 13		—	—	—	√
	AN14	Input	Analog input pin 14		—	—	—	√
	AN15	Input	Analog input pin 15		—	—	—	√

Note: The connected A/D module differs for each pin. The control registers of each module must be set.

19.3 Register Descriptions

The A/D converter has the following registers. For details on register addresses and register states in each processing state, refer to section 27, List of Registers.

Table 19.2 Register Configuration

Register Name	Abbreviation	R/W	Initial value	Address	Access Size
A/D data register 0	ADDR0	R	H'0000	H'FFFFC900	16
A/D data register 1	ADDR1	R	H'0000	H'FFFFC902	16
A/D data register 2	ADDR2	R	H'0000	H'FFFFC904	16
A/D data register 3	ADDR3	R	H'0000	H'FFFFC906	16
A/D control/status register_0	ADCSR_0	R/W	H'0000	H'FFFFC910	16
A/D control register_0	ADCR_0	R/W	H'0000	H'FFFFC912	16
A/D data register 4	ADDR4	R	H'0000	H'FFFFC980	16
A/D data register 5	ADDR5	R	H'0000	H'FFFFC982	16
A/D data register 6	ADDR6	R	H'0000	H'FFFFC984	16
A/D data register 7	ADDR7	R	H'0000	H'FFFFC986	16
A/D control/status register_1	ADCSR_1	R/W	H'0000	H'FFFFC990	16
A/D control register_1	ADCR_1	R/W	H'0000	H'FFFFC992	16
A/D data register 8	ADDR8	R	H'0000	H'FFFFCA00	16
A/D data register 9	ADDR9	R	H'0000	H'FFFFCA02	16
A/D data register 10	ADDR10	R	H'0000	H'FFFFCA04	16
A/D data register 11	ADDR11	R	H'0000	H'FFFFCA06	16
A/D data register 12	ADDR12	R	H'0000	H'FFFFCA08	16
A/D data register 13	ADDR13	R	H'0000	H'FFFFCA0A	16
A/D data register 14	ADDR14	R	H'0000	H'FFFFCA0C	16
A/D data register 15	ADDR15	R	H'0000	H'FFFFCA0E	16
A/D control/status register_2	ADCSR_2	R/W	H'0000	H'FFFFCA10	16
A/D control register_2	ADCR_2	R/W	H'0000	H'FFFFCA12	16
A/D trigger select register_0	ADTSR_0	R/W	H'0000	H'FFFFE890	8, 16
A/D trigger select register_1	ADTSR_1	R/W	H'0000	H'FFFFE892	8, 16

19.3.1 A/D Data Registers 0 to 15 (ADDR0 to ADDR15)

ADDRs are 16-bit read-only registers. The conversion result for each analog input channel is stored in ADDR with the corresponding number. For example, the conversion result of AN4 is stored in ADDR4.

The converted 10-bit data is stored in bits 6 to 15. The lower 6 bits are always read as 0.

The initial value of ADDR is H'0000.

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 6		All 0	R	Bit Data (10 bits)
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

19.3.2 A/D Control/Status Registers_0 to _2 (ADCSR_0 to ADCSR_2)

ADCSR for each module controls A/D conversion operations.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADF	ADIE	-	-	TRGE	-	CONADF	STC	CKSL[1:0]		ADM[1:0]		ADCS	CH[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * Writing 0 to this bit after reading it as 1 is clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Description
15	ADF	0	R/(W)*	A/D End Flag A status flag that indicates the end of A/D conversion. [Setting conditions] • When A/D conversion ends in single mode • When A/D conversion ends on all specified channels in scan mode [Clearing conditions] • When 0 is written after reading ADF = 1 • When the DMAC is activated by an ADI interrupt and ADDR is read • When the DTC is activated by an ADI interrupt and ADDR is read while the DISEL bit in the MRB register of the DTC is cleared to 0
14	ADIE	0	R/W	A/D Interrupt Enable The A/D conversion end interrupt (ADI) request is enabled when 1 is set When changing the operating mode, first clear the ADST bit to 0.
13, 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
11	TRGE	0	R/W	Trigger Enable Enables or disables triggering of A/D conversion by $\overline{\text{ADTRG}}$, an MTU2 trigger, or an MTU2S trigger. 0: A/D conversion triggering is disabled 1: A/D conversion triggering is enabled When changing the operating mode, first clear the ADST bit to 0.
10	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
9	CONADF	0	R/W	ADF Control Controls setting of the ADF bit in 2-channel scan mode. The setting of this bit is valid only when triggering of A/D conversion is enabled (TRGE = 1) in 2-channel scan mode. The setting of this bit is ignored in single mode, 4-channel scan mode, or 8-channel scan mode. 0: The ADF bit is set when A/D conversion started by the group 0 trigger or group 1 trigger has finished. 1: The ADF bit is set when A/D conversion started by the group 0 trigger and A/D conversion started by the group 1 trigger have both finished. Note that the triggering order has no affect. When changing the operating mode, first clear the ADST bit to 0.
8	STC	0	R/W	State Control Sets the A/D conversion time in combination with the CKSL1 and CKSL0 bits. 0: 50 states 1: 64 states When changing the A/D conversion time, first clear the ADST bit to 0.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	CKSL[1:0]	00	R/W	Clock Select 1 and 0 Select the A/D conversion time. 00: Pϕ/4 01: Pϕ/3 10: Pϕ/2 11: Pϕ When changing the A/D conversion time, first clear the ADST bit to 0. CKSL[1:0] = B'11 can be set while Pϕ ≤ 25 [MHz].
5, 4	ADM[1:0]	00	R/W	A/D Mode 1 and 0 Select the A/D conversion mode. 2-channel scan mode is supported only by A/D modules 0 and 1. Do not select 2-channel scan mode in A/D module 2. 00: Single mode 01: 4-channel scan mode 10: 8-channel scan mode 11: 2-channel scan mode When changing the operating mode, first clear the ADST bit to 0.
3	ADCS	0	R/W	A/D Continuous Scan Selects either single-cycle scan or continuous scan in scan mode. This bit is valid only when scan mode is selected. 0: Single-cycle scan 1: Continuous scan When changing the operating mode, first clear the ADST bit to 0.
2 to 0	CH[2:0]	000	R/W	Channel Select 2 to 0 Select analog input channels. See table 19.3. When changing the operating mode, first clear the ADST bit to 0.

Note: * Writing 0 to this bit after reading it as 1 is clears the flag and is the only allowed way.

19.3.3 A/D Control Registers_0 to _2 (ADCR_0 to ADCR_2)

ADCR for each module controls A/D conversion.

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

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	ADST	0	R/W	A/D Start Starts or stops A/D conversion. When this bit is set to 1, A/D conversion is started. When this bit is cleared to 0, A/D conversion is stopped and the A/D converter enters the idle state. In single or single-cycle scan mode, this bit is automatically cleared to 0 when A/D conversion ends on the selected single channel. In continuous scan mode, A/D conversion is continuously performed for the selected channels in sequence until this bit is cleared by software, reset, or in software standby mode or module standby mode.
12 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Table 19.3 Channel Select List

- Single Mode

			Analog Input Channels		
Bit 2	Bit 1	Bit 0	Single Mode		
CH2	CH1	CH0	A/D_0	A/D_1	A/D_2
0	0	0	AN0	AN4	AN8
		1	AN1	AN5	AN9
	1	0	AN2	AN6	AN10
		1	AN3	AN7	AN11
1	0	0	Setting prohibited	Setting prohibited	AN12
		1			AN13
	1	0			AN14
		1			AN15

- 2-Channel Scan Mode

			Analog Input Channels							
			Software Activation			Other than Software Activation				
Bit 2	Bit 1	Bit 0				A/D_0		A/D_1		
CH2	CH1	CH0	A/D_0	A/D_1	A/D_2	Group 0	Group 1	Group 0	Group 1	A/D_2
0	0	0	AN0	AN4	Setting prohibited	AN0	AN2	AN4	AN6	Setting prohibited
		1	AN0, AN1	AN4, AN5		AN0, AN1	AN2, AN3	AN4, AN5	AN6, AN7	
	1	0	AN2	AN6		Setting prohibited	Setting prohibited	Setting prohibited	Setting prohibited	
		1	AN2, AN3	AN6, AN7						
1	0	0	Setting prohibited	Setting prohibited						
		1								
	1	0								
		1								

Note: When 2-, 4-, or 8-channel scan mode has been selected, the channels for operation are selected only by bits CH2 to CH0. For example, when 8-channel scan mode has been selected with continuous scan mode, continuous conversion will be performed on AN8 if bits CH2 to CH0 have been set to 000.

- 4-Channel Scan Mode

			Analog Input Channels		
Bit 2	**Bit 1**	**Bit 0**	**4-Channel Scan Mode***		
CH2	**CH1**	**CH0**	**A/D_0**	**A/D_1**	**A/D_2**
0	0	0	AN0	AN4	AN8
		1	AN0, AN1	AN4, AN5	AN8, AN9
	1	0	AN0 to AN2	AN4 to AN6	AN8 to AN10
		1	AN0 to AN3	AN4 to AN7	AN8 to AN11
1	0	0	Setting prohibited	Setting prohibited	AN12
		1			AN12, AN13
	1	0			AN12 to AN14
		1			AN12 to AN15

Note: * Continuous scan mode or single-scan mode can be selected with the ADCS bit. When 2-, 4-, or 8-channel scan mode has been selected, the channels for operation are selected only by bits CH2 to CH0. For example, when 8-channel scan mode has been selected with continuous scan mode, continuous conversion will be performed on AN8 if bits CH2 to CH0 have been set to 000.

- 8-Channel Scan Mode

			Analog Input Channels
Bit 2	**Bit 1**	**Bit 0**	**8-Channel Scan Mode***
CH2	**CH1**	**CH0**	**A/D_2**
0	0	0	AN8
		1	AN8, AN9
	1	0	AN8 to AN10
		1	AN8 to AN11
1	0	0	AN8 to AN12
		1	AN8 to AN13
	1	0	AN8 to AN14
		1	AN8 to AN15

Note: * Continuous scan mode or single-scan mode can be selected with the ADCS bit. When 2-, 4-, or 8-channel scan mode has been selected, the channels for operation are selected only by bits CH2 to CH0. For example, when 8-channel scan mode has been selected with continuous scan mode, continuous conversion will be performed on AN8 if bits CH2 to CH0 have been set to 000.

19.3.4 A/D Trigger Select Registers_0 and _1 (ADTSR_0 and ADTSR_1)

The ADTSR enables an A/D conversion started by an external trigger signal.

In particular, the four channels in A/D module 0 and A/D module 1 are divided into two groups (group 0 and group 1) and the A/D trigger can be specified for each group independently in 2-channel scan mode.

- ADTSR_0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRG11S[3:0]				TRG01S[3:0]				TRG1S[3:0]				TRG0S[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	TRG11S[3:0]	0000	R/W	A/D Trigger 1 Group 1 Select 3 to 0 Select an external trigger, MTU2 trigger, or MTU2S trigger to start A/D conversion for group 1 when A/D module 1 is in 2-channel scan mode. 0000: External trigger pin ($\overline{\text{ADTRG}}$) input 0001: TGRA input capture/compare match on each MTU2 channel or TCNT_4 trough in complementary PWM mode (TRGAN) 0010: MTU2 CH0 compare match (TRG0N) 0011: MTU2 A/D conversion start request delaying (TRG4AN) 0100: MTU2 A/D conversion start request delaying (TRG4BN) 0101: TGRA input capture/compare match on each MTU2S channel or TCNT_4 trough in complementary PWM mode (TRGAN) 0110: Setting prohibited 0111: MTU2S A/D conversion start request delaying (TRG4AN) 1000: MTU2S A/D conversion start request delaying (TRG4BN) 1001: Setting prohibited 101x: Setting prohibited 11xx: Setting prohibited When switching the selector, first clear the ADST bit in the A/D control register (ADCR) to 0. Specify different trigger sources for the group 0 and group 1 conversion requests so that a group 0 conversion request is not generated simultaneously with a group 1 conversion request in 2-channel scan mode.

Bit	Bit Name	Initial Value	R/W	Description
11 to 8	TRG01S[3:0]	0000	R/W	A/D Trigger 0 Group 1 Select 3 to 0 Select an external trigger, MTU2 trigger, or MTU2S trigger to start A/D conversion for group 1 when A/D module 0 is in 2-channel scan mode. 0000: External trigger pin ($\overline{\text{ADTRG}}$) input 0001: TGRA input capture/compare match on each MTU2 channel or TCNT_4 trough in complementary PWM mode (TRGAN) 0010: MTU2 CH0 compare match (TRG0N) 0011: MTU2 A/D conversion start request delaying (TRG4AN) 0100: MTU2 A/D conversion start request delaying (TRG4BN) 0101: TGRA input capture/compare match on each MTU2S channel or TCNT_4 trough in complementary PWM mode (TRGAN) 0110: Setting prohibited 0111: MTU2S A/D conversion start request delaying (TRG4AN) 1000: MTU2S A/D conversion start request delaying (TRG4BN) 1001: Setting prohibited 101x: Setting prohibited 11xx: Setting prohibited When switching the selector, first clear the ADST bit in the A/D control register (ADCR) to 0. Specify different trigger sources for the group 0 and group 1 conversion requests so that a group 0 conversion request is not generated simultaneously with a group 1 conversion request in 2-channel scan mode.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	TRG1S[3:0]	0000	R/W	A/D Trigger 1 Select 3 to 0 Select an external trigger, MTU2 trigger, or MTU2S trigger to start A/D conversion for A/D module 1. In 2-channel scan mode, these bits select an external trigger, MTU2 trigger, or MTU2S trigger to start A/D conversion for group 0. 0000: External trigger pin ($\overline{\text{ADTRG}}$) input 0001: TGRA input capture/compare match on each MTU2 channel or TCNT_4 trough in complementary PWM mode (TRGAN) 0010: MTU2 CH0 compare match (TRG0N) 0011: MTU2 A/D conversion start request delaying (TRG4AN) 0100: MTU2 A/D conversion start request delaying (TRG4BN) 0101: TGRA input capture/compare match on each MTU2S channel or TCNT_4 trough in complementary PWM mode (TRGAN) 0110: Setting prohibited 0111: MTU2S A/D conversion start request delaying (TRG4AN) 1000: MTU2S A/D conversion start request delaying (TRG4BN) 1001: Setting prohibited 101x: Setting prohibited 11xx: Setting prohibited When switching the selector, first clear the ADST bit in the A/D control register (ADCR) to 0. Specify different trigger sources for the group 0 and group 1 conversion requests so that a group 0 conversion request is not generated simultaneously with a group 1 conversion request in 2-channel scan mode.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	TRG0S[3:0]	0000	R/W	A/D Trigger 0 Select 3 to 0 Select an external trigger, MTU2 trigger, or MTU2S trigger to start A/D conversion for A/D module 0. In 2-channel scan mode, these bits select an external trigger, MTU2 trigger, or MTU2S trigger to start A/D conversion for group 0. 0000: External trigger pin ($\overline{\text{ADTRG}}$) input 0001: TGRA input capture/compare match on each MTU2 channel or TCNT_4 trough in complementary PWM mode (TRGAN) 0010: MTU2 CH0 compare match (TRG0N) 0011: MTU2 A/D conversion start request delaying (TRG4AN) 0100: MTU2 A/D conversion start request delaying (TRG4BN) 0101: TGRA input capture/compare match on each MTU2S channel or TCNT_4 trough in complementary PWM mode (TRGAN) 0110: Setting prohibited 0111: MTU2S A/D conversion start request delaying (TRG4AN) 1000: MTU2S A/D conversion start request delaying (TRG4BN) 1001: Setting prohibited 101x: Setting prohibited 11xx: Setting prohibited When switching the selector, first clear the ADST bit in the A/D control register (ADCR) to 0. Specify different trigger sources for the group 0 and group 1 conversion requests so that a group 0 conversion request is not generated simultaneously with a group 1 conversion request in 2-channel scan mode.

[Legend]

x: Don't care

- ADTSR_1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRG2S[3:0]				-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	TRG2S[3:0]	0000	R/W	A/D Trigger 2 Select 3 to 0 Select an external trigger, MTU2 trigger, or MTU2S trigger to start A/D conversion for A/D module 2. 0000: External trigger pin ($\overline{\text{ADTRG}}$) input 0001: TGRA input capture/compare match for each MTU2 channel or TCNT_4 trough in complementary PWM mode (TRGAN) 0010: MTU2 CH0 compare match (TRG0N) 0011: MTU2 A/D conversion start request delaying (TRG4AN) 0100: MTU2 A/D conversion start request delaying (TRG4BN) 0101: TGRA input capture/compare match for each MTU2 channel or TCNT_4 trough in complementary PWM mode (TRGAN) 0110: Setting prohibited 0111: MTU2S A/D conversion start request delaying (TRG4AN) 1000: MTU2S A/D conversion start request delaying (TRG4BN) 1001: Setting prohibited 101x: Setting prohibited 11xx: Setting prohibited When switching the selector, first clear the ADST bit in the A/D control register (ADCR) to 0.
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

[Legend]

x: Don't care

19.4 Operation

The A/D converter operates by successive approximation with 10-bit resolution. It has two operating modes; single mode and scan mode. There are two kinds of scan mode: continuous mode and single-cycle mode. When changing the operating mode or analog input channel, in order to prevent incorrect operation, first clear the ADST bit to 0 in ADCR.

19.4.1 Single Mode

In single mode, A/D conversion is to be performed only once on the specified single channel. The operations are as follows.

1. A/D conversion is started when the ADST bit in ADCR is set to 1, according to software, MTU2, MTU2S, or external trigger input.
2. When A/D conversion is completed, the result is transferred to the A/D data register corresponding to the channel.
3. On completion of conversion, the ADF bit in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt request is generated.
4. The ADST bit remains set to 1 during A/D conversion. When A/D conversion ends, the ADST bit is automatically cleared to 0 and the A/D converter enters the idle state.
   When the ADST bit is cleared to 0 during A/D conversion, A/D conversion stops and the A/D converter enters the idle state.

19.4.2 Continuous Scan Mode

In continuous scan mode, A/D conversion is to be performed sequentially on the specified channels (up to four channels in SH7083/SH7084/SH7085 and up to eight channels in SH7086).

1. When the ADST bit in ADCR is set to 1 by software, MTU2, MTU2S, or external trigger input, A/D conversion starts on the channel with the lowest number in the group (AN0, AN1, ..., AN7).
2. When A/D conversion for each channel is completed, the result is sequentially transferred to the A/D data register corresponding to each channel.
3. When conversion of all the selected channels is completed, the ADF bit in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt is requested after A/D conversion ends. Conversion of the first channel in the group starts again.
4. Steps 2 to 3 are repeated as long as the ADST bit remains set to 1. When the ADST bit is cleared to 0, A/D conversion stops and the A/D converter enters the idle state.

19.4.3 Single-Cycle Scan Mode

In single-cycle scan mode, A/D conversion is to be performed once on the specified channels (up to four channels in SH7083/SH7084/SH7085 and up to eight channels in SH7086).

1. When the ADST bit in ADCR is set to 1 by a software, MTU2, MTU2S, or external trigger input, A/D conversion starts on the channel with the lowest number in the group (AN0, AN1, ..., AN7).
2. When A/D conversion for each channel is completed, the result is sequentially transferred to the A/D data register corresponding to each channel.
3. When conversion of all the selected channels is completed, the ADF bit in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt is requested after A/D conversion ends.
4. After A/D conversion ends, the ADST bit is automatically cleared to 0 and the A/D converter enters the idle state. When the ADST bit is cleared to 0 during A/D conversion, A/D conversion stops and the A/D converter enters the idle state.

19.4.4 Input Sampling and A/D Conversion Time

The A/D converter has a built-in sample-and-hold circuit for each module. The A/D converter samples the analog input when the A/D conversion start delay time (t_D) has passed after the ADST bit in ADCR is set to 1, then starts conversion. Figure 19.2 shows the A/D conversion timing. Table 19.4 shows the A/D conversion time.

As indicated in figure 19.2, the A/D conversion time (t_{CONV}) includes t_D and the input sampling time (t_{SPL}). The length of t_D varies depending on the timing of the write access to ADCR. The total conversion time therefore varies within the ranges indicated in table 19.4.

In scan mode, the values given in table 19.4 apply to the first conversion time. The values given in table 19.5 apply to the second and subsequent conversions.

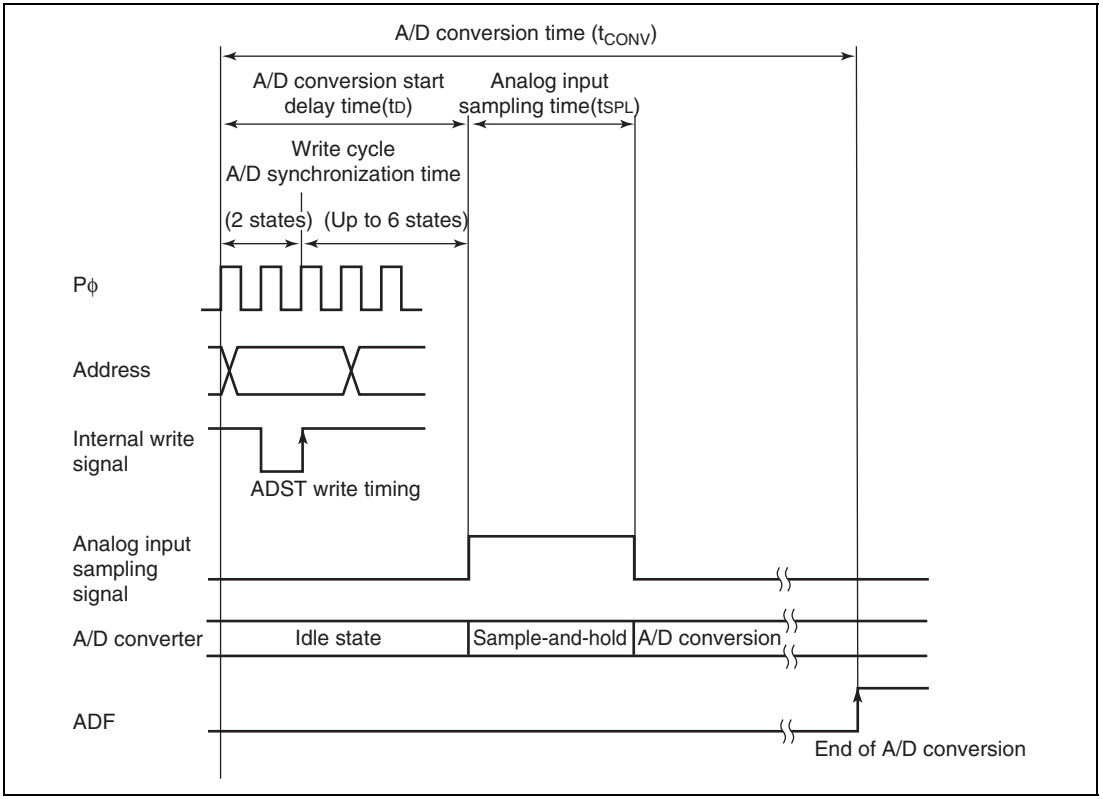


Figure 19.2 A/D Conversion Timing

Table 19.4 A/D Conversion Time (Single Mode)

		STC = 0											
		CKSL1 = 0						CKSL1 = 1					
		CKSL0 = 0			CKSL0 = 1			CKSL0 = 0			CKSL0 = 1		
Item	**Symbol**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**
A/D conversion start delay time	t_D	2	—	6	2	—	5	2	—	4	2	—	3
Input sampling time	t_{SPL}	—	24	—	—	18	—	—	12	—	—	6	—
A/D conversion time	t_{CONV}	202	—	206	152	—	155	102	—	104	52	—	53

		STC = 1											
		CKSL1 = 0						CKSL1 = 1					
		CKSL0 = 0			CKSL0 = 1			CKSL0 = 0			CKSL0 = 1		
Item	**Symbol**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**
A/D conversion start delay time	t_D	2	—	6	2	—	5	2	—	4	2	—	3
Input sampling time	t_{SPL}	—	36	—	—	27	—	—	18	—	—	9	—
A/D conversion time	t_{CONV}	258	—	262	194	—	197	130	—	132	66	—	67

Note: All values represent the number of states for Pϕ.

Table 19.5 A/D Conversion Time (Scan Mode)

				Conversion Time Calculation Example	
STC	**CKSL1**	**CKSL0**	**Conversion Time (State)**	**Pϕ = 25 MHz**	**Pϕ = 40 MHz**
0	0	0	200 (Fixed)	8 μs	5 μs
		1	150 (Fixed)	6 μs	3.8 μs
	1	0	100 (Fixed)	4 μs	2.5 μs
		1	50 (Fixed)	2 μs	Setting prohibited
1	0	0	256 (Fixed)	10.2 μs	6.4 μs
		1	192 (Fixed)	7.7 μs	4.8 μs
	1	0	128 (Fixed)	5.1 μs	3.2 μs
		1	64 (Fixed)	2.6 μs	Setting prohibited

19.4.5 A/D Converter Activation by MTU2 or MTU2S

The A/D converter can be independently activated by an A/D conversion request from the interval timer of the MTU2 or MTU2S.

To activate the A/D converter by the MTU2 or MTU2S, first set the TRGE bit in the A/D control/status register (ADCSR) to 1, and then set the A/D trigger select register (ADTSR). After this register setting has been made, the ADST bit in ADCR is automatically set to 1 when an A/D conversion request from the interval timer of the MTU2 or MTU2S occurs. The timing from setting of the ADST bit until the start of A/D conversion is the same as when 1 is written to the ADST bit by software.

19.4.6 External Trigger Input Timing

A/D conversion can be externally triggered. When the TRGE bit in the A/D control/status register (ADCSR) is set to 1 while the A/D trigger select registers_0 and _1 (ADTSR_0 and ADSTR_1) are set to external trigger pin input, external trigger input is enabled at the $\overline{\text{ADTRG}}$ pin. A falling edge of the $\overline{\text{ADTRG}}$ pin sets the ADST bit to 1 in ADCR, starting A/D conversion. Other operations, in both single and scan modes, are the same as when the ADST bit has been set to 1 by software. Figure 19.3 shows the timing.

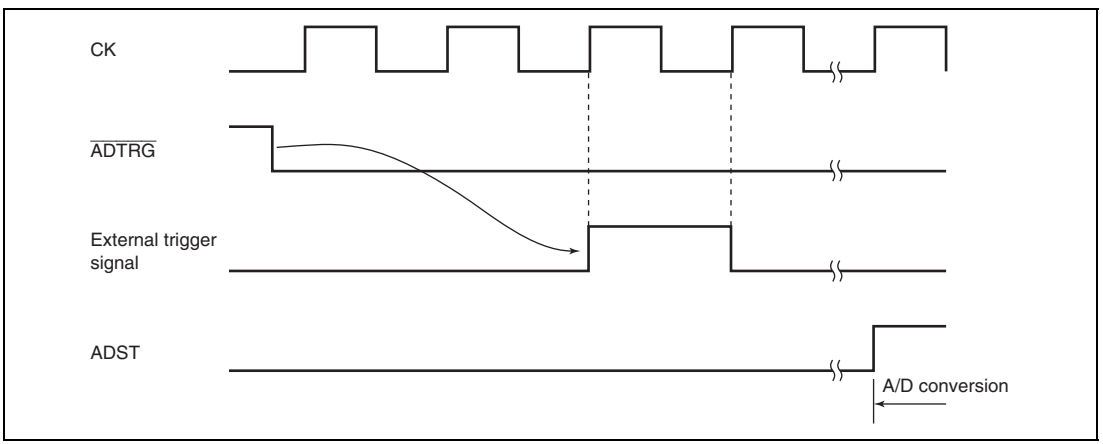


Figure 19.3 External Trigger Input Timing

19.4.7 2-Channel Scanning

In 2-channel scan mode, since the four channels of analog input are divided into groups 0 and 1, triggers for activation of groups 0 and 1 are independently specifiable. Conversion end interrupts in 2-channel scan mode can be generated either on completion of group 0 or group 1 or on completion of group 0 and group 1. If conversion is to be started by triggers, the different sources for groups 0 and 1 are specified in ADTSR. A request for conversion by group 1 generated during conversion by group 0 is ignored. Figure 19.4 shows an example of operation when TRG4AN of the MTU2 has been specified as the A/D conversion start request by group 0 and TRG4BN of the MTU2 has been specified as the A/D conversion start request by group 1.

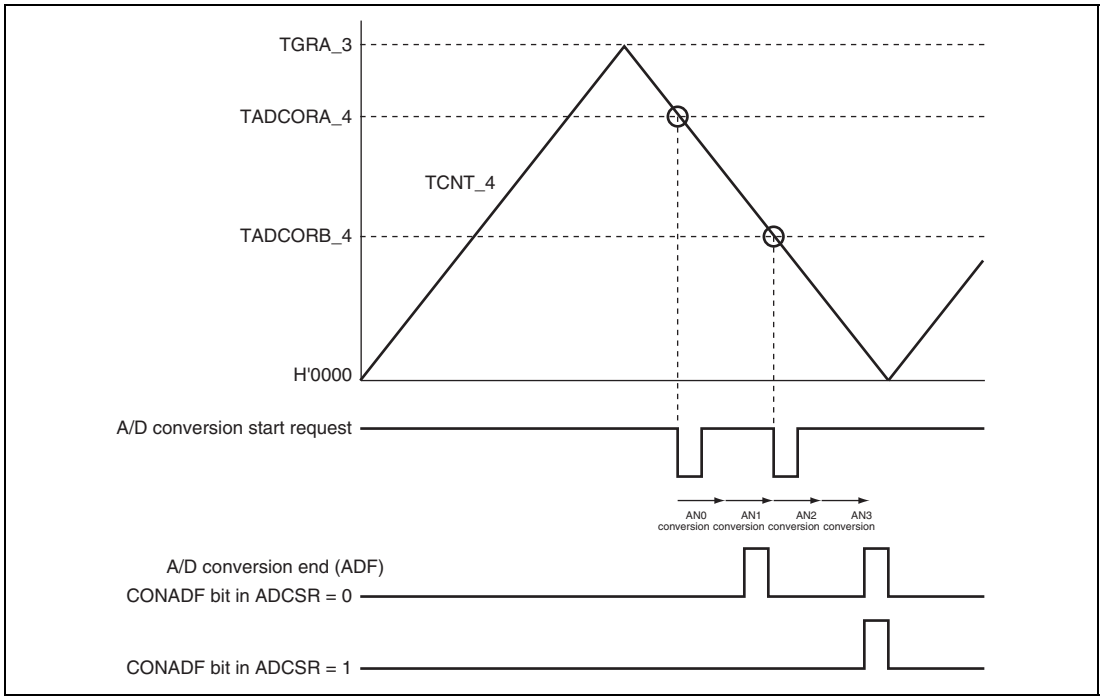


Figure 19.4 Example of 2-Channel Scanning

19.5 Interrupt Sources and DMAC and DTC Transfer Requests

The A/D converter can generate an A/D conversion end interrupt request (ADI). The ADI interrupt can be enabled by setting the ADIE bit in the A/D control/status register (ADCSR) to 1, or disabled by clearing the ADIE bit to 0.

The DTC or DMAC can be activated by an ADI interrupt. In this case an interrupt request is not sent to the CPU.

When the DTC or DMAC is activated by an ADI interrupt, the ADF bit in ADCSR is automatically cleared when data is transferred by the DTC or DMAC. Having the converted data read by the DTC or DMAC in response to an ADI interrupt enables continuous conversion to be achieved without imposing a load on software.

Table 19.6 A/D Converter Interrupt Source

Name	Interrupt Source	Interrupt Source Flag	DTC Activation	DMAC Activation
ADI0	A/D_0 conversion completed	ADF in ADCSR_0	Possible	Impossible
ADI1	A/D_1 conversion completed	ADF in ADCSR_1	Possible	Possible
ADI2	A/D_2 conversion completed	ADF in ADCSR_2	Possible	Impossible

19.6 Definitions of A/D Conversion Accuracy

This LSI's A/D conversion accuracy definitions are given below.

- Resolution

  The number of A/D converter digital output codes
- Quantization error

  The deviation inherent in the A/D converter, given by 1/2 LSB (see figure 19.5).
- Offset error

  The deviation of the analog input voltage value from the ideal A/D conversion characteristic when the digital output changes from the minimum voltage value B'0000000000 (H'000) to B'0000000001 (H'001) (see figure 19.6).
- Full-scale error

  The deviation of the analog input voltage value from the ideal A/D conversion characteristic when the digital output changes from B'1111111110 (H'3FE) to B'1111111111 (H'3FF) (see figure 19.6).
- Nonlinearity error

  The error with respect to the ideal A/D conversion characteristic between zero voltage and full-scale voltage. Does not include offset error, full-scale error, or quantization error (see figure 19.6).
- Absolute accuracy

  The deviation between the digital value and the analog input value. Includes offset error, full-scale error, quantization error, and nonlinearity error.

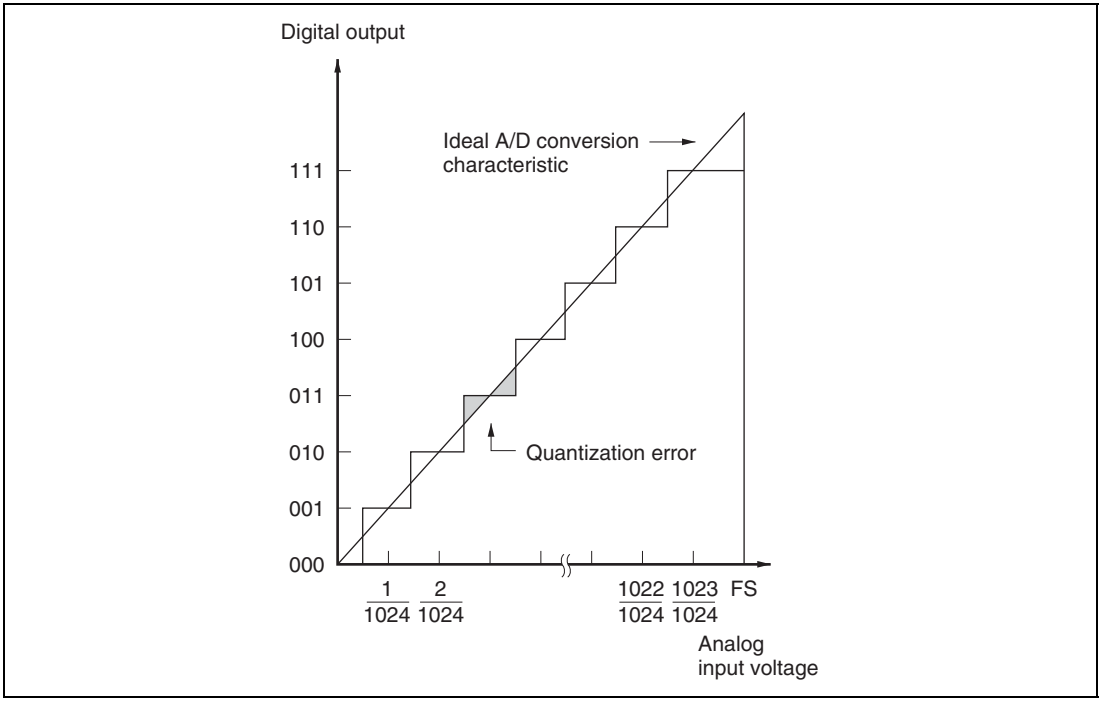


Figure 19.5 Definitions of A/D Conversion Accuracy

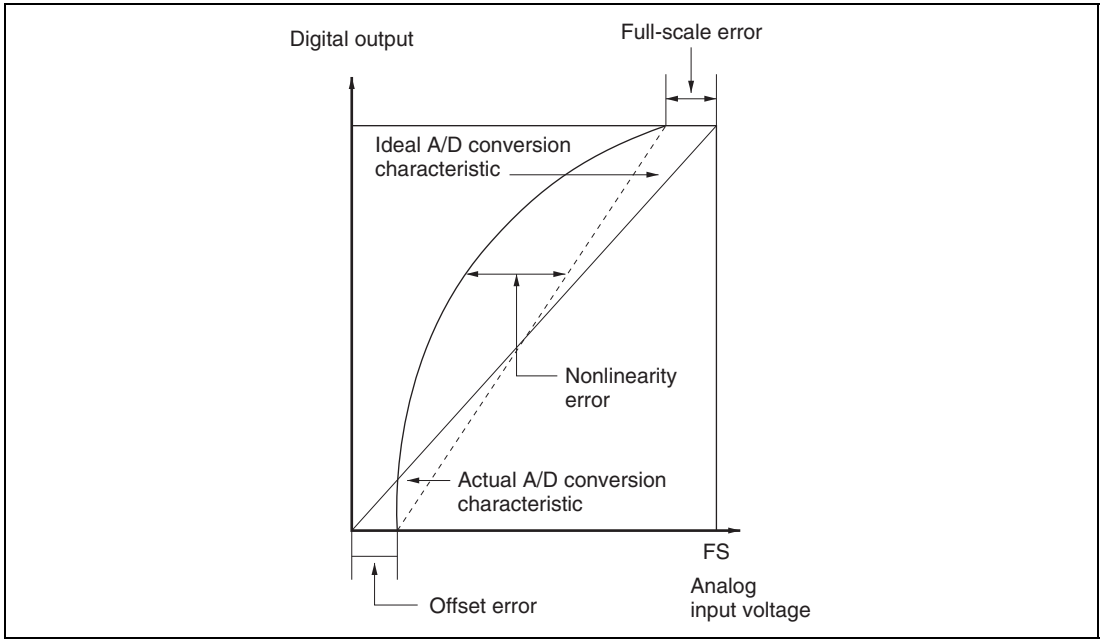


Figure 19.6 Definitions of A/D Conversion Accuracy

19.7 Usage Notes

19.7.1 Module Standby Mode Setting

Operation of the A/D converter can be disabled or enabled using the standby control register. The initial setting is for operation of the A/D converter to be halted. Register access is enabled by clearing module standby mode. For details, refer to section 26, Power-Down Modes.

19.7.2 Permissible Signal Source Impedance

This LSI's analog input is designed such that conversion accuracy is guaranteed for an input signal for which the signal source impedance is 1 kΩ or less. This specification is provided to enable the A/D converter's sample-and-hold circuit input capacitance to be charged within the sampling time; if the sensor output impedance exceeds 1 kΩ, charging may be insufficient and it may not be possible to guarantee A/D conversion accuracy. However, for A/D conversion in single mode with a large capacitance provided externally, the input load will essentially comprise only the internal input resistance of 10 kΩ, and the signal source impedance is ignored. However, as a low-pass filter effect is obtained in this case, it may not be possible to follow an analog signal with a large differential coefficient (e.g., 5 mV/μs or greater) (see figure 19.7). When converting a high-speed analog signal or converting in scan mode, a low-impedance buffer should be inserted.

19.7.3 Influences on Absolute Accuracy

Adding capacitance results in coupling with GND, and therefore noise in GND may adversely affect absolute precision. Be sure to make the connection to an electrically stable GND such as AVss.

Care is also required to insure that filter circuits do not interfere in the accuracy by the printed circuit digital signals on the mounting board (i.e. acting as antennas).

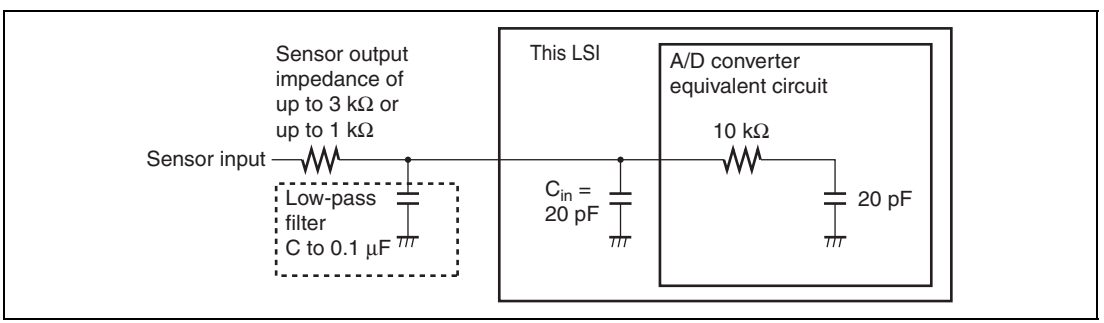


Figure 19.7 Example of Analog Input Circuit

19.7.4 Range of Analog Power Supply and Other Pin Settings

If the conditions below are not met, the reliability of the device may be adversely affected.

- Analog input voltage range
  The voltage applied to analog input pin ANn during A/D conversion should be in the range AVss ≤ VAN ≤ AV_{ref}.
- Relationship between AVcc, AVss and Vcc, Vss
  Set AVss = Vss for the relationship between AVcc, AVss and Vcc, Vss. If the A/D converter is not used, the AVcc and AVss pins must not be left open.
- AV_{ref} input voltage range
  The voltage applied to the AV_{ref} pin should be in the range AV_{ref} ≤ AVcc.
  If the A/D converter is not used, set AV_{ref} = AVcc.

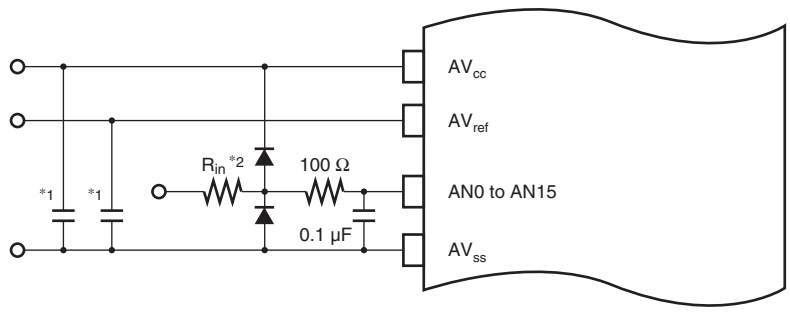
19.7.5 Notes on Board Design

In board design, digital circuitry and analog circuitry should be as mutually isolated as possible, and layout in which digital circuit signal lines and analog circuit signal lines cross or are in close proximity should be avoided as far as possible. Failure to do so may result in incorrect operation of the analog circuitry due to inductance, adversely affecting A/D conversion values. Also, digital circuitry must be isolated from the analog input signals (AN0 to AN15), and analog power supply (AVcc) by the analog ground (AVss). Also, the analog ground (AVss) should be connected at one point to a stable ground (Vss) on the board.

19.7.6 Notes on Noise Countermeasures

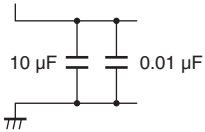
A protection circuit should be connected in order to prevent damage due to abnormal voltage, such as an excessive surge at the analog input pins (AN0 to AN15), between AVcc and AVss, as shown in figure 19.8. Also, the bypass capacitors connected to AVcc and the filter capacitor connected to AN0 to AN15 must be connected to AVss.

If a filter capacitor is connected, the input currents at the analog input pins (AN0 to AN15) are averaged, and so an error may arise. Also, when A/D conversion is performed frequently, as in scan mode, if the current charged and discharged by the capacitance of the sample-and-hold circuit in the A/D converter exceeds the current input via the input impedance (R_{in}), an error will arise in the analog input pin voltage. Careful consideration is therefore required when deciding circuit constants.



Notes: Values are reference values.

1.



2. R_{in}: Input impedance

Figure 19.8 Example of Analog Input Protection Circuit

Table 19.7 Analog Pin Specifications

Item	Min.	Max.	Unit	Condition
Analog input capacitance	—	20	pF	—
Permissible signal source impedance	—	3	kΩ	Pϕ ≤ 20 MHz
		1		Pϕ > 20 MHz

Section 20 Compare Match Timer (CMT)

This LSI has an on-chip compare match timer (CMT) consisting of a 2-channel 16-bit timer. The CMT has a16-bit counter, and can generate interrupts at set intervals.

20.1 Features

- Selection of four counter input clocks
  Any of four internal clocks (Pϕ/8, Pϕ/32, Pϕ/128, and Pϕ/512) can be selected independently for each channel.
- Selection of DTC transfer request or interrupt request generation on compare match by DTC setting
- When not in use, the CMT can be stopped by halting its clock supply to reduce power consumption.

Figure 20.1 shows a block diagram of CMT.

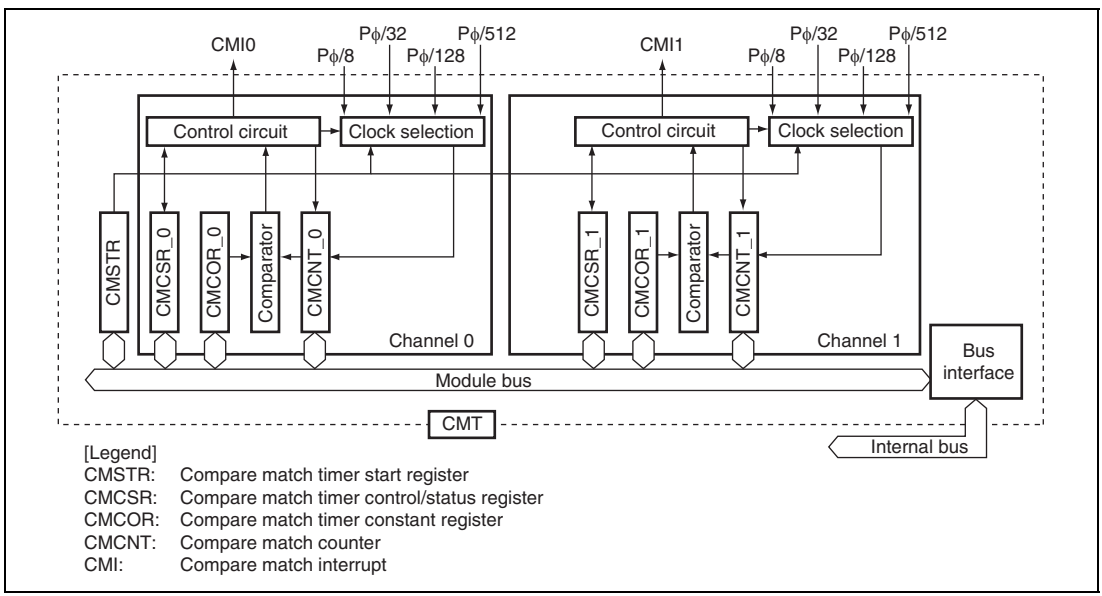


Figure 20.1 Block Diagram of CMT

20.2 Register Descriptions

The CMT has the following registers. For the states of these registers in each processing status, refer to section 27, List of Registers.

Table 20.1 Register Configuration

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Common	Compare match timer start register	CMSTR	R/W	H'0000	H'FFFFCE00	8, 16, 32
0	Compare match timer control/ status register_0	CMCSR_0	R/W	H'0000	H'FFFFCE02	8, 16
	Compare match counter_0	CMCNT_0	R/W	H'0000	H'FFFFCE04	8, 16, 32
	Compare match constant register_0	CMCOR_0	R/W	H'FFFF	H'FFFFCE06	8, 16
1	Compare match timer control/ status register_1	CMCSR_1	R/W	H'0000	H'FFFFCE08	8, 16, 32
	Compare match counter_1	CMCNT_1	R/W	H'0000	H'FFFFCE0A	8, 16
	Compare match constant register_1	CMCOR_1	R/W	H'FFFF	H'FFFFCE0C	8, 16, 32

20.2.1 Compare Match Timer Start Register (CMSTR)

CMSTR is a 16-bit register that selects whether compare match counter (CMCNT) operates or is stopped.

CMSTR is initialized to H'0000 when a power-on reset or a transition to standby mode occurs.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STR1	STR0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
15 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	STR1	0	R/W	Count Start 1 Specifies whether compare match counter 1 operates or is stopped. 0: CMCNT_1 count is stopped 1: CMCNT_1 count is started
0	STR0	0	R/W	Count Start 0 Specifies whether compare match counter 0 operates or is stopped. 0: CMCNT_0 count is stopped 1: CMCNT_0 count is started

20.2.2 Compare Match Timer Control/Status Register (CMCSR)

CMCSR is a 16-bit register that indicates compare match generation, enables/disables interrupts and selects the counter input clock.

CMCSR is initialized to H'0000 when a power-on reset or a transition to standby mode occurs.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CMF	CMIE	-	-	-	-	CKS[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	(R/W)*	R/W	R	R	R	R	R/W	R/W

Note: * Only 0 can be written to clear the flag after 1 is read.

Bit	Bit Name	Initial value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	CMF	0	R/(W)*[1]	Compare Match Flag Indicates whether or not the values of CMCNT and CMCOR match. 0: CMCNT and CMCOR values do not match [Clearing conditions] • When a power-on reset or a transition to standby mode occurs • When 0 is written to this bit after reading CMF=1*[2] • When CMT registers are accessed when the value of the DISEL bit of MRB in the DTC is 0 after activating the DTC by CMI interrupts (except when the DTC transfer counter value has become H'0000). 1: CMCNT and CMCOR values match [Setting condition] • When CMCNT and CMCOR values match
6	CMIE	0	R/W	Compare Match Interrupt Enable Enables or disables compare match interrupt (CMI) generation when CMCNT and CMCOR values match (CMF=1). 0: Compare match interrupt (CMI) disabled 1: Compare match interrupt (CMI) enabled

Bit	Bit Name	Initial value	R/W	Description
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	CKS[1:0]	00	R/W	Clock Select Select the clock to be input to CMCNT from four internal clocks obtained by dividing the peripheral operating clock (Pϕ). When the STR bit in CMSTR is set to 1, CMCNT starts counting on the clock selected with CKS[1:0] bits. 00: Pϕ/8 01: Pϕ/32 10: Pϕ/128 11: Pϕ/512

Notes: 1. Only 0 can be written to clear the flag after 1 is read.
2. If the flag is set by another compare match before writing 0 to the bit after reading it as 1, the flag will not be cleared by writing 0 to it once. In this case, read the bit as 1 again and write 0 to it.

20.2.3 Compare Match Counter (CMCNT)

CMCNT is a 16-bit register used as an up-counter. When the counter input clock is selected with CKS[1:0] bits in CMCSR and the STR bit in CMSTR is set to 1, CMCNT starts counting using the selected clock. When the value in CMCNT and the value in compare match constant register (CMCOR) match, CMCNT is cleared to H'0000 and the CMF flag in CMCSR is set to 1.

CMCNT is initialized to H'0000 when a power-on reset or a transition to standby mode occurs.

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

20.2.4 Compare Match Constant Register (CMCOR)

CMCOR is a 16-bit register that sets the interval up to a compare match with CMCNT.

CMCOR is initialized to H'FFFF when a power-on reset or a transition to standby mode occurs.

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

20.3 Operation

20.3.1 Interval Count Operation

When an internal clock is selected with CKS[1:0] bits in CMCSR and the STR bit in CMSTR is set to 1, CMCNT starts counting up using the selected clock. When the values in CMCNT and CMCOR match, CMCNT is cleared to H'0000 and the CMF flag in CMCSR is set to 1. When the CMIE bit in CMCSR is set to 1, a compare match interrupt (CMI) is requested. CMCNT then starts counting up again from H'0000.

Figure 20.2 shows the operation of the compare match counter.

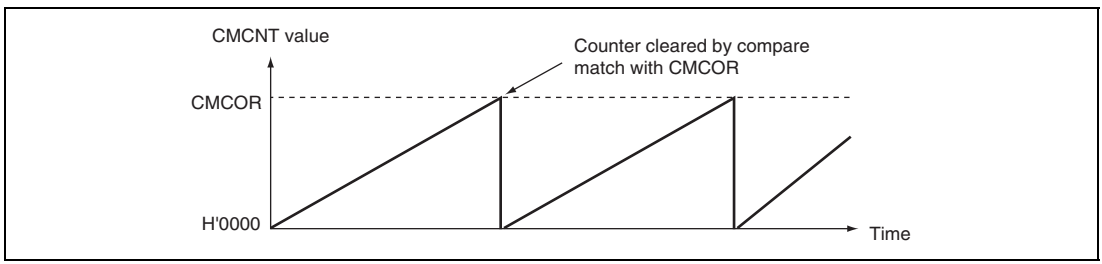


Figure 20.2 Counter Operation

20.3.2 CMCNT Count Timing

One of four internal clocks (Pϕ/8, Pϕ/32, Pϕ/128, and Pϕ/512) obtained by dividing the Pϕ clock can be selected with CKS[1:0] bits in CMCSR. Figure 20.3 shows the timing.

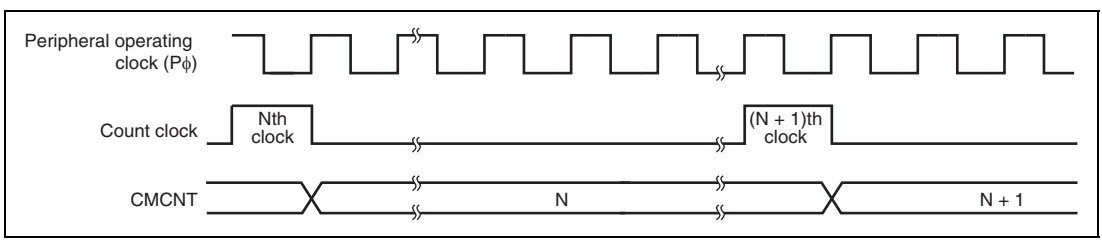


Figure 20.3 Count Timing

20.4 Interrupts

20.4.1 Interrupt Sources and DTC Transfer Requests

The CMT has two channels, and each of them to which a different vector address is allocated has a compare match interrupt as shown in table 20.2. When both the interrupt request flag (CMF) and the interrupt enable bit (CMIE) are set to 1, the corresponding interrupt request is output. When the interrupt is used to activate a CPU interrupt, the priority of channels can be changed by the interrupt controller settings. For details, see section 6, Interrupt Controller (INTC).

Clear the CMF bit to 0 from within the user exception handling routine. If this is not done, the interrupt will be generated again. If the next compare match sets the CMF flag before 0 is written to this flag after it has been read as 1, writing 0 to this flag will not clear it. Therefore, read this flag as 1 again before writing 0 to it.

A compare match interrupt request can activate the data transfer controller (DTC). Since data transfer due to DTC activation automatically clears the flag while the DISEL bit of the DTC is 0 and the transfer counter value is not 0, an interrupt is not issued to the CPU in this case. However, if the DISEL bit and the transfer counter value are both 0, or the DISEL bit is 1, data transfer does not clear the flag, so an interrupt request for the CPU is generated after completion of the data transfer.

Table 20.2 Interrupt Sources

Channel	Interrupt Source	Interrupt Enable Bit	Interrupt Flag	DTC Activation	Priority
0	CMI_0	CMIE of CMCSR_0	CMF of CMCSR_0	Possible	High
1	CMI_1	CMIE of CMCSR_1	CMF of CMCSR_1	Possible	Low

20.4.2 Timing of Setting Compare Match Flag

When CMCOR and CMCNT match, a compare match signal is generated in the last state in which the values match (when the CMCNT value is updated to H'0000) and the CMF bit in CMCSR is set to 1. That is, after a match between CMCOR and CMCNT, the compare match signal is not generated until the next CMCNT counter clock input. Figure 20.4 shows the timing of CMF bit setting.

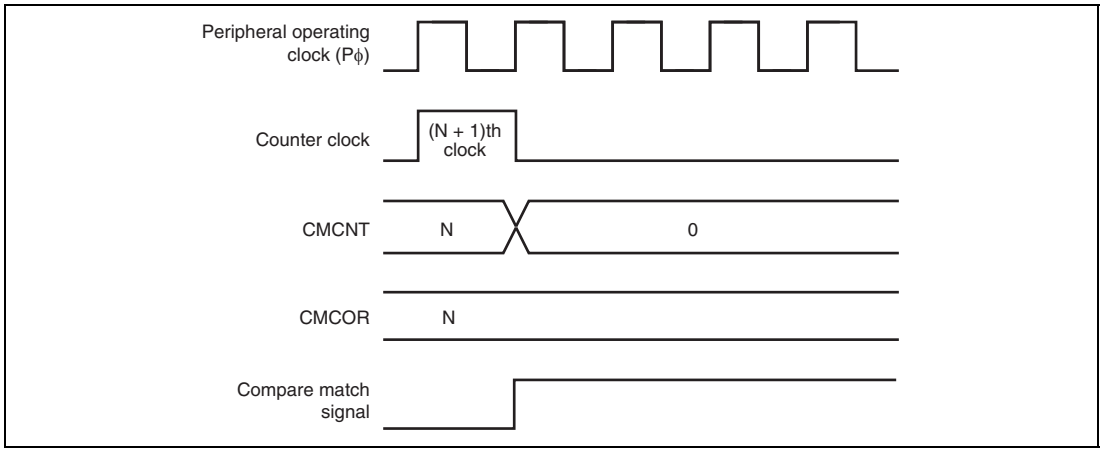


Figure 20.4 Timing of CMF Setting

20.4.3 Timing of Clearing Compare Match Flag

The CMF bit in CMCSR is cleared by reading 1 from this bit, then writing 0. However, in the case of the DTC being activated, the CMF bit is automatically cleared to 0 when data is transferred by the DTC. (except when the DTC's DISEL bit is cleared to 0 and the counter value is 0, or when the DISEL bit is set to 1).

20.5 Usage Notes

20.5.1 Module Standby Mode Setting

The CMT operation can be disabled or enabled using the standby control register. The initial setting is for CMT operation to be halted. Access to a register is enabled by clearing module standby mode. For details, refer to section 26, Power-Down Modes.

20.5.2 Conflict between Write and Compare-Match Processes of CMCNT

When the compare match signal is generated in the T2 cycle while writing to CMCNT, clearing CMCNT has priority over writing to it. In this case, CMCNT is not written to. Figure 20.5 shows the timing to clear the CMCNT counter.

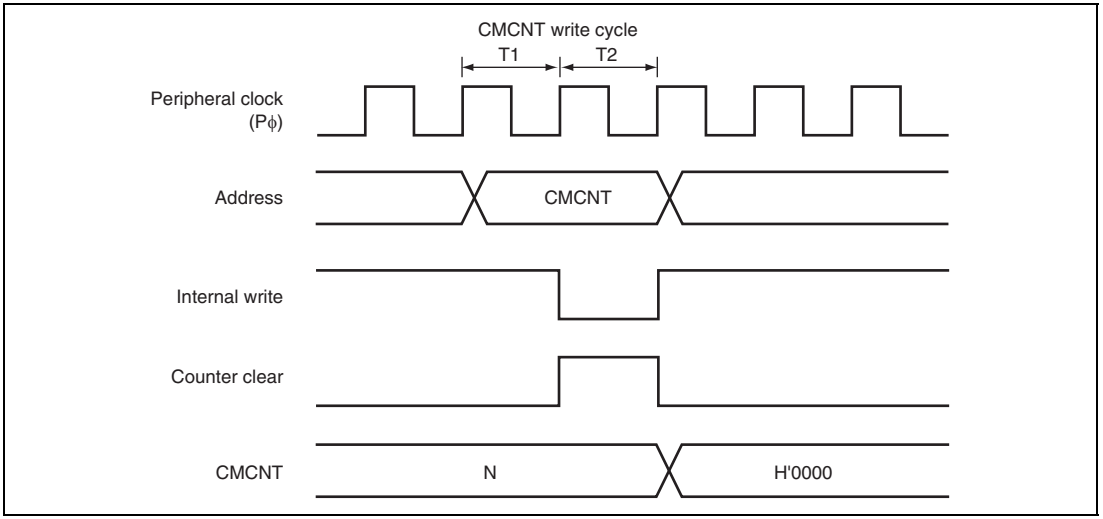


Figure 20.5 Conflict between Write and Compare-Match Processes of CMCNT

20.5.3 Conflict between Word-Write and Count-Up Processes of CMCNT

Even when the count-up occurs in the T2 cycle while writing to CMCNT counter in words, the writing has priority over the count-up. In this case, the count-up is not performed. Figure 20.6 shows the timing to write to CMCNT in words.

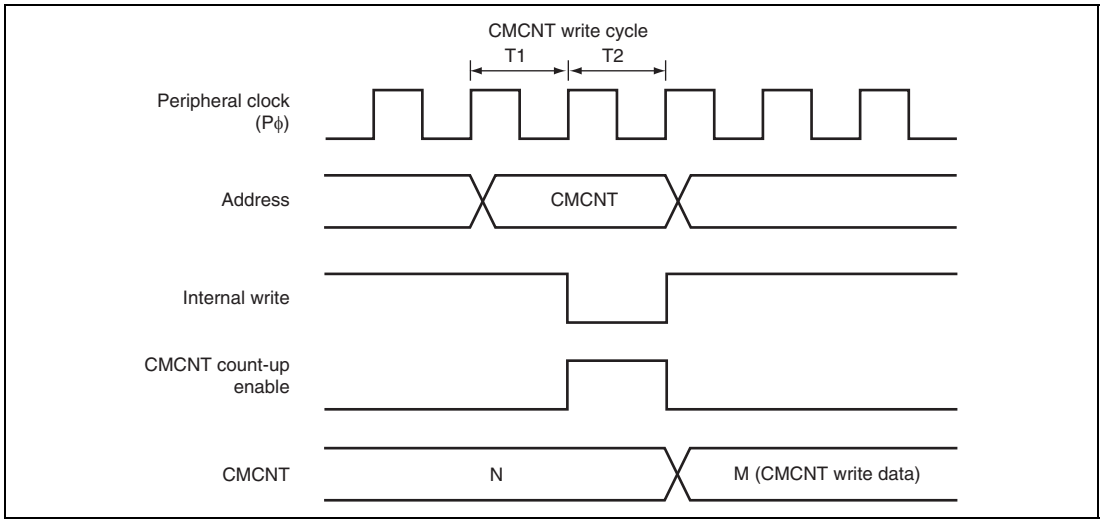


Figure 20.6 Conflict between Word-Write and Count-Up Processes of CMCNT

20.5.4 Conflict between Byte-Write and Count-Up Processes of CMCNT

Even when the count-up occurs in the T2 cycle while writing to CMCNT counter in bytes, the byte-writing has priority over the count-up. In this case, the count-up is not performed. The byte data on another side, which is not written to, is also not counted and the previous contents remain.

Figure 20.7 shows the timing when the count-up occurs in the T2 cycle while writing to CMCNT in bytes.

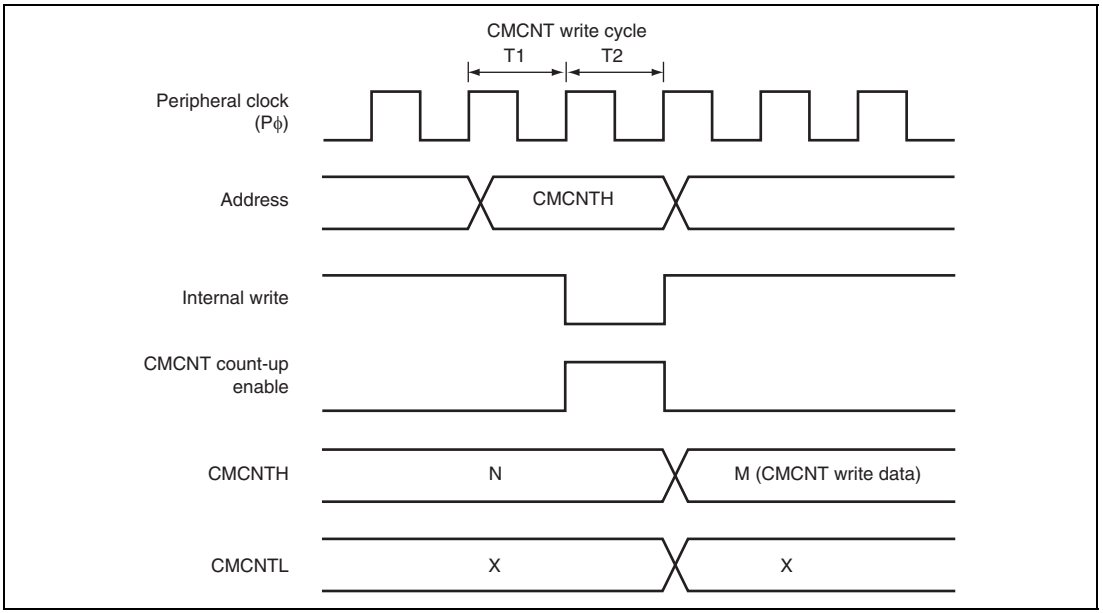


Figure 20.7 Conflict between Byte-Write and Count-Up Processes of CMCNT

20.5.5 Compare Match between CMCNT and CMCOR

Do not set the same value in CMCNT and CMCOR while CMCNT is not counting. If set, the CMF bit in CMCSR is set to 1 and CMCNT is cleared to H'0000.

Section 21 Pin Function Controller (PFC)

The pin function controller (PFC) is composed of registers that are used to select the functions of multiplexed pins and assign pins to be inputs or outputs. Tables 21.1 to 21.16 list the multiplexed pins of this LSI.

Tables 21.17 to 21.20 list the pin functions in each operating mode.

Table 21.1 SH7083 Multiplexed Pins (Port A)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
A	PA3 I/O (port)	A24 output (BSC)	RXD1 input (SCI)	—	—
	PA4 I/O (port)	A23 output (BSC)	TXD1 output (SCI)	—	—
	PA5 I/O (port)	A22 output (BSC)	DREQ1 input (DMAC)	IRQ1 input (INTC)	SCK1 I/O (SCI)
	PA7 I/O (port)	$\overline{\text{CS3}}$ output (BSC)	TCLKB input (MTU2)	—	—
	PA8 I/O (port)	RDWR output (BSC)	IRQ2 input (INTC)	TCLKC input (MTU2)	—
	PA9 I/O (port)	CKE output (BSC)	IRQ3 input (INTC)	TCLKD input (MTU2)	—
	PA10 I/O (port)	$\overline{\text{CS0}}$ output (BSC)	$\overline{\text{POE4}}$ input (POE)	—	—
	PA12 I/O (port)	$\overline{\text{WRL}}$/DQMLL output (BSC)	$\overline{\text{POE6}}$ input (POE)	—	—
	PA13 I/O (port)	$\overline{\text{WRH}}$/DQMLU output (BSC)	$\overline{\text{POE7}}$ input (POE)	—	—
	PA14 I/O (port)	$\overline{\text{RD}}$ output (BSC)	—	—	—
	PA15 I/O (port)	CK output (CPG)	—	—	—

Table 21.2 SH7084 Multiplexed Pins (Port A)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
A	PA0 I/O (port)	$\overline{\text{CS4}}$ output (BSC)	RXD0 input (SCI)	—	—
	PA1 I/O (port)	$\overline{\text{CS5}}$ output (BSC)	TXD0 output (SCI)	—	—
	PA2 I/O (port)	A25 output (BSC)	DREQ0 input (DMAC)	IRQ0 input (INTC)	SCK0 I/O (SCI)
	PA3 I/O (port)	A24 output (BSC)	RXD1 input (SCI)	—	—
	PA4 I/O (port)	A23 output (BSC)	TXD1 output (SCI)	—	—
	PA5 I/O (port)	A22 output (BSC)	DREQ1 input (DMAC)	IRQ1 input (INTC)	SCK1 I/O (SCI)
	PA6 I/O (port)	$\overline{\text{CS2}}$ output (BSC)	TCLKA input (MTU2)	—	—
	PA7 I/O (port)	$\overline{\text{CS3}}$ output (BSC)	TCLKB input (MTU2)	—	—

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
A	PA8 I/O (port)	RDWR output (BSC)	IRQ2 input (INTC)	TCLKC input (MTU2)	—
	PA9 I/O (port)	CKE output (BSC)	IRQ3 input (INTC)	TCLKD input (MTU2)	—
	PA10 I/O (port)	$\overline{\text{CS0}}$ output (BSC)	$\overline{\text{POE4}}$ input (POE)	—	—
	PA11 I/O (port)	$\overline{\text{CS1}}$ output (BSC)	$\overline{\text{POE5}}$ input (POE)	—	—
	PA12 I/O (port)	$\overline{\text{WRL}}$/DQMLL output (BSC)	$\overline{\text{POE6}}$ input (POE)	—	—
	PA13 I/O (port)	$\overline{\text{WRH}}$/DQMLU output (BSC)	$\overline{\text{POE7}}$ input (POE)	—	—
	PA14 I/O (port)	$\overline{\text{RD}}$ output (BSC)	—	—	—
	PA15 I/O (port)	CK output (CPG)	—	—	—
	PA16 I/O (port)	$\overline{\text{AH}}$ output (BSC)	CKE output (BSC)	—	—
	PA17 I/O (port)	$\overline{\text{WAIT}}$ input (BSC)	—	—	—

Table 21.3 SH7085 Multiplexed Pins (Port A)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
A	PA0 I/O (port)	$\overline{\text{CS4}}$ output (BSC)	RXD0 input (SCI)	—	—
	PA1 I/O (port)	$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$ output (BSC)	TXD0 output (SCI)	—	—
	PA2 I/O (port)	A25 output (BSC)	DREQ0 input (DMAC)	IRQ0 input (INTC)	SCK0 I/O (SCI)
	PA3 I/O (port)	A24 output (BSC)	RXD1 input (SCI)	—	—
	PA4 I/O (port)	A23 output (BSC)	TXD1 output (SCI)	—	—
	PA5 I/O (port)	A22 output (BSC)	DREQ1 input (DMAC)	IRQ1 input (INTC)	SCK1 I/O (SCI)
	PA6 I/O (port)	$\overline{\text{CS2}}$ output (BSC)	TCLKA input (MTU2)	—	—
	PA7 I/O (port)	$\overline{\text{CS3}}$ output (BSC)	TCLKB input (MTU2)	—	—
	PA8 I/O (port)	RDWR output (BSC)	IRQ2 input (INTC)	TCLKC input (MTU2)	—
	PA9 I/O (port)	$\overline{\text{FRAME}}$ output (BSC)	CKE output (BSC)	IRQ3 input (INTC)	TCLKD input (MTU2)
	PA10 I/O (port)	$\overline{\text{CS0}}$ output (BSC)	$\overline{\text{POE4}}$ input (POE)	—	—
	PA11 I/O (port)	$\overline{\text{CS1}}$ output (BSC)	$\overline{\text{POE5}}$ input (POE)	—	—
	PA12 I/O (port)	$\overline{\text{WRL}}$/DQMLL output (BSC)	$\overline{\text{POE6}}$ input (POE)	—	—
	PA13 I/O (port)	$\overline{\text{WRH}}$/$\overline{\text{WE}}$/DQMLU output (BSC)	$\overline{\text{POE7}}$ input (POE)	—	—
	PA14 I/O (port)	$\overline{\text{RD}}$ output (BSC)	—	—	—
	PA15 I/O (port)	CK output (CPG)	—	—	—

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
A	PA16 I/O (port)	WRHH/ICIOWR/AH/ DQMUU output (BSC)	CKE output (BSC)	DREQ2 input (DMAC)	AUDSYNC output (AUD) *
	PA17 I/O (port)	WAIT input (BSC)	DACK2 output (DMAC)	—	—
	PA18 I/O (port)	BREQ input (BSC)	TEND0 output (DMAC)	—	—
	PA19 I/O (port)	BACK output (BSC)	TEND1 output (DMAC)	—	—
	PA20 I/O (port)	CS4 output (BSC)	RASU output (BSC)	—	—
	PA21 I/O (port)	CS5/CE1A output (BSC)	CASU output (BSC)	TIC5U input (MTU2)	—
	PA22 I/O (port)	WRHL/ICIORD /DQMUL output (BSC)	TIC5V input (MTU2)	—	—
	PA23 I/O (port)	WRHH/ICIOWR/AH/ DQMUU output (BSC)	TIC5W input (MTU2)	—	—
	PA24 I/O (port)	CE2A output (BSC)	DREQ3 input (DMAC)	—	—
	PA25 I/O (port)	CE2B output (BSC)	DACK3 output (DMAC)	POE8 input (POE)	—

Note: * Only in F-ZTAT version supporting full functions of E10A.

Table 21.4 SH7086 Multiplexed Pins (Port A)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
A	PA0 I/O (port)	CS4 output (BSC)	RXD0 input (SCI)	—	—
	PA1 I/O (port)	CS5/CE1A output (BSC)	TXD0 output (SCI)	—	—
	PA2 I/O (port)	A25 output (BSC)	DREQ0 input (DMAC)	IRQ0 input (INTC)	SCK0 I/O (SCI)
	PA3 I/O (port)	A24 output (BSC)	RXD1 input (SCI)	—	—
	PA4 I/O (port)	A23 output (BSC)	TXD1 output (SCI)	—	—
	PA5 I/O (port)	A22 output (BSC)	DREQ1 input (DMAC)	IRQ1 input (INTC)	SCK1 I/O (SCI)
	PA6 I/O (port)	CS2 output (BSC)	TCLKA input (MTU2)	—	—
	PA7 I/O (port)	CS3 output (BSC)	TCLKB input (MTU2)	—	—
	PA8 I/O (port)	RDWR output (BSC)	IRQ2 input (INTC)	TCLKC input (MTU2)	—
	PA9 I/O (port)	FRAME output (BSC)	CKE output (BSC)	IRQ3 input (INTC)	TCLKD input (MTU2)
	PA10 I/O (port)	CS0 output (BSC)	POE4 input (POE)	—	—
	PA11 I/O (port)	CS1 output (BSC)	POE5 input (POE)	—	—

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
A	PA12 I/O (port)	$\overline{\text{WRL}}$/DQMLL output (BSC)	$\overline{\text{POE6}}$ input (POE)	—	—
	PA13 I/O (port)	$\overline{\text{WRH}}$/DQMLU/$\overline{\text{WE}}$ output (BSC)	$\overline{\text{POE7}}$ input (POE)	—	—
	PA14 I/O (port)	$\overline{\text{RD}}$ output (BSC)	—	—	—
	PA15 I/O (port)	CK output (CPG)	—	—	—
	PA16 I/O (port)	$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/ DQMUU output (BSC)	CKE output (BSC)	DREQ2 input (DMAC)	$\overline{\text{AUDSYNC}}$ output (AUD) *
	PA17 I/O (port)	$\overline{\text{WAIT}}$ input (BSC)	DACK2 output (DMAC)	—	—
	PA18 I/O (port)	$\overline{\text{BREQ}}$ input (BSC)	TEND0 output (DMAC)	—	—
	PA19 I/O (port)	$\overline{\text{BACK}}$ output (BSC)	TEND1 output (DMAC)	—	—
	PA20 I/O (port)	$\overline{\text{CS4}}$ output (BSC)	$\overline{\text{RASU}}$ output (BSC)	—	—
	PA21 I/O (port)	$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$ output (BSC)	$\overline{\text{CASU}}$ output (BSC)	TIC5U input (MTU2)	—
	PA22 I/O (port)	$\overline{\text{WRHL}}$/$\overline{\text{ICIORD}}$ /DQMUL output (BSC)	TIC5V input (MTU2)	—	—
	PA23 I/O (port)	$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/ DQMUU output (BSC)	TIC5W input (MTU2)	—	—
	PA24 I/O (port)	$\overline{\text{CE2A}}$ output (BSC)	DREQ3 input (DMAC)	—	—
	PA25 I/O (port)	$\overline{\text{CE2B}}$ output (BSC)	DACK3 output (DMAC)	$\overline{\text{POE8}}$ input (POE)	—
	PA26 I/O (port)	A26 output (BSC)	IRQ0 input (INTC)	—	—
	PA27 I/O (port)	A27 output (BSC)	IRQ1 input (INTC)	—	—
	PA28 I/O (port)	A28 output (BSC)	IRQ2 input (INTC)	—	—
	PA29 I/O (port)	A29 output (BSC)	IRQ3 input (INTC)	—	—

Note: * Only in F-ZTAT version supporting full functions of E10A.

Table 21.5 SH7083 Multiplexed Pins (Port B)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
B	PB0 I/O (port)	A16 output (BSC)	TIC5WS input (MTU2S)	—	—
	PB1 I/O (port)	A17 output (BSC)	TIC5W input (MTU2)	—	—
	PB2 I/O (port)	IRQ0 input (INTC)	$\overline{\text{POE0}}$ input (POE)	—	—
	PB4 I/O (port)	$\overline{\text{RASL}}$ output (BSC)	IRQ2 input (INTC)	$\overline{\text{POE2}}$ input (POE)	—
	PB5 I/O (port)	$\overline{\text{CASL}}$ output (BSC)	IRQ3 input (INTC)	$\overline{\text{POE3}}$ input (POE)	—
	PB6 I/O (port)	A18 output (BSC)	$\overline{\text{BACK}}$ output (BSC)	IRQ4 input (INTC)	RXD0 input (SCI)
	PB7 I/O (port)	A19 output (BSC)	$\overline{\text{BREQ}}$ input (BSC)	IRQ5 input (INTC)	TXD0 output (SCI)
	PB8 I/O (port)	A20 output (BSC)	$\overline{\text{WAIT}}$ input (BSC)	IRQ6 input (INTC)	SCK0 I/O (SCI)
	PB9 I/O (port)	A21 output (BSC)	IRQ7 input (INTC)	$\overline{\text{ADTRG}}$ input (A/D)	$\overline{\text{POE8}}$ input (POE)

Table 21.6 SH7084/SH7085/SH7086 Multiplexed Pins (Port B)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
B	PB0 I/O (port)	A16 output (BSC)	TIC5WS input (MTU2S)	—	—
	PB1 I/O (port)	A17 output (BSC)	TIC5W input (MTU2)	—	—
	PB2 I/O (port)	IRQ0 input (INTC)	$\overline{\text{POE0}}$ input (POE)	SCL I/O (IIC2)	—
	PB3 I/O (port)	IRQ1 input (INTC)	$\overline{\text{POE1}}$ input (POE)	SDA I/O (IIC2)	—
	PB4 I/O (port)	$\overline{\text{RASL}}$ output (BSC)	IRQ2 input (INTC)	$\overline{\text{POE2}}$ input (POE)	—
	PB5 I/O (port)	$\overline{\text{CASL}}$ output (BSC)	IRQ3 input (INTC)	$\overline{\text{POE3}}$ input (POE)	—
	PB6 I/O (port)	A18 output (BSC)	$\overline{\text{BACK}}$ output (BSC)	IRQ4 input (INTC)	RXD0 input (SCI)
	PB7 I/O (port)	A19 output (BSC)	$\overline{\text{BREQ}}$ input (BSC)	IRQ5 input (INTC)	TXD0 output (SCI)
	PB8 I/O (port)	A20 output (BSC)	$\overline{\text{WAIT}}$ input (BSC)	IRQ6 input (INTC)	SCK0 I/O (SCI)
	PB9 I/O (port)	A21 output (BSC)	IRQ7 input (INTC)	$\overline{\text{ADTRG}}$ input (A/D)	$\overline{\text{POE8}}$ input (POE)

Table 21.7 SH7083/SH7084/SH7085 Multiplexed Pins (Port C)

Port	Function 1 (Related Module)	Function 2 (Related Module)
C	PC0 I/O (port)	A0 output (BSC)
	PC1 I/O (port)	A1 output (BSC)
	PC2 I/O (port)	A2 output (BSC)
	PC3 I/O (port)	A3 output (BSC)
	PC4 I/O (port)	A4 output (BSC)
	PC5 I/O (port)	A5 output (BSC)
	PC6 I/O (port)	A6 output (BSC)
	PC7 I/O (port)	A7 output (BSC)
	PC8 I/O (port)	A8 output (BSC)
	PC9 I/O (port)	A9 output (BSC)
	PC10 I/O (port)	A10 output (BSC)
	PC11 I/O (port)	A11 output (BSC)
	PC12 I/O (port)	A12 output (BSC)
	PC13 I/O (port)	A13 output (BSC)
	PC14 I/O (port)	A14 output (BSC)
	PC15 I/O (port)	A15 output (BSC)

Table 21.8 SH7086 Multiplexed Pins (Port C)

Port	Function 1 (Related Module)	Function 2 (Related Module)
C	PC0 I/O (port)	A0 output (BSC)
	PC1 I/O (port)	A1 output (BSC)
	PC2 I/O (port)	A2 output (BSC)
	PC3 I/O (port)	A3 output (BSC)
	PC4 I/O (port)	A4 output (BSC)
	PC5 I/O (port)	A5 output (BSC)
	PC6 I/O (port)	A6 output (BSC)
	PC7 I/O (port)	A7 output (BSC)
	PC8 I/O (port)	A8 output (BSC)
	PC9 I/O (port)	A9 output (BSC)
	PC10 I/O (port)	A10 output (BSC)
	PC11 I/O (port)	A11 output (BSC)

Port	Function 1 (Related Module)	Function 2 (Related Module)
C	PC12 I/O (port)	A12 output (BSC)
	PC13 I/O (port)	A13 output (BSC)
	PC14 I/O (port)	A14 output (BSC)
	PC15 I/O (port)	A15 output (BSC)
	PC18 I/O (port)	A18 output (BSC)
	PC19 I/O (port)	A19 output (BSC)
	PC20 I/O (port)	A20 output (BSC)
	PC21 I/O (port)	A21 output (BSC)
	PC22 I/O (port)	A22 output (BSC)
	PC23 I/O (port)	A23 output (BSC)
	PC24 I/O (port)	A24 output (BSC)
	PC25 I/O (port)	A25 output (BSC)

Table 21.9 SH7083/SH7084 Multiplexed Pins (Port D)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)
D	PD0 I/O (port)	D0 I/O (BSC)	—	—
	PD1 I/O (port)	D1 I/O (BSC)	—	—
	PD2 I/O (port)	D2 I/O (BSC)	TIC5U input (MTU2)	—
	PD3 I/O (port)	D3 I/O (BSC)	TIC5V input (MTU2)	—
	PD4 I/O (port)	D4 I/O (BSC)	TIC5W input (MTU2)	—
	PD5 I/O (port)	D5 I/O (BSC)	TIC5US input (MTU2S)	—
	PD6 I/O (port)	D6 I/O (BSC)	TIC5VS input (MTU2S)	—
	PD7 I/O (port)	D7 I/O (BSC)	TIC5WS input (MTU2S)	—
	PD8 I/O (port)	D8 I/O (BSC)	TIOC3AS I/O (MTU2S)	AUDATA0 output (AUD)*
	PD9 I/O (port)	D9 I/O (BSC)	TIOC3BS I/O (MTU2S)	AUDATA1 output (AUD)*
	PD10 I/O (port)	D10 I/O (BSC)	TIOC3CS I/O (MTU2S)	AUDATA2 output (AUD)*
	PD11 I/O (port)	D11 I/O (BSC)	TIOC3DS I/O (MTU2S)	AUDATA3 output (AUD)*
	PD12 I/O (port)	D12 I/O (BSC)	TIOC4AS I/O (MTU2S)	—
	PD13 I/O (port)	D13 I/O (BSC)	TIOC4BS I/O (MTU2S)	—
	PD14 I/O (port)	D14 I/O (BSC)	TIOC4CS I/O (MTU2S)	AUDCK output (AUD)*
	PD15 I/O (port)	D15 I/O (BSC)	TIOC4DS I/O (MTU2S)	$\overline{\text{AUDSYNC}}$ output (AUD)*

Note: * Only in F-ZTAT version supporting full functions of E10A.

Table 21.10 SH7085/SH7086 Multiplexed Pins (Port D)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
D	PD0 I/O (port)	D0 I/O (BSC)	—	—	—
	PD1 I/O (port)	D1 I/O (BSC)	—	—	—
	PD2 I/O (port)	D2 I/O (BSC)	TIC5U input (MTU2)	—	—
	PD3 I/O (port)	D3 I/O (BSC)	TIC5V input (MTU2)	—	—
	PD4 I/O (port)	D4 I/O (BSC)	TIC5W input (MTU2)	—	—
	PD5 I/O (port)	D5 I/O (BSC)	TIC5US input (MTU2S)	—	—
	PD6 I/O (port)	D6 I/O (BSC)	TIC5VS input (MTU2S)	—	—
	PD7 I/O (port)	D7 I/O (BSC)	TIC5WS input (MTU2S)	—	—
	PD8 I/O (port)	D8 I/O (BSC)	TIOC3AS I/O (MTU2S)	—	—
	PD9 I/O (port)	D9 I/O (BSC)	TIOC3BS I/O (MTU2S)	—	—
	PD10 I/O (port)	D10 I/O (BSC)	TIOC3CS I/O (MTU2S)	—	—
	PD11 I/O (port)	D11 I/O (BSC)	TIOC3DS I/O (MTU2S)	—	—
	PD12 I/O (port)	D12 I/O (BSC)	TIOC4AS I/O (MTU2S)	—	—
	PD13 I/O (port)	D13 I/O (BSC)	TIOC4BS I/O (MTU2S)	—	—
	PD14 I/O (port)	D14 I/O (BSC)	TIOC4CS I/O (MTU2S)	—	—
	PD15 I/O (port)	D15 I/O (BSC)	TIOC4DS I/O (MTU2S)	—	—
	PD16 I/O (port)	D16 I/O (BSC)	IRQ0 input (INTC)	$\overline{\text{POE4}}$ input (POE)	AUDATA0 output (AUD)*
	PD17 I/O (port)	D17 I/O (BSC)	IRQ1 input (INTC)	$\overline{\text{POE5}}$ input (POE)	AUDATA1 output (AUD)*
	PD18 I/O (port)	D18 I/O (BSC)	IRQ2 input (INTC)	$\overline{\text{POE6}}$ input (POE)	AUDATA2 output (AUD)*
	PD19 I/O (port)	D19 I/O (BSC)	IRQ3 input (INTC)	$\overline{\text{POE7}}$ input (POE)	AUDATA3 output (AUD)*

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)
D	PD20 I/O (port)	D20 I/O (BSC)	IRQ4 input (INTC)	TIC5WS input (MTU2S)	—
	PD21 I/O (port)	D21 I/O (BSC)	IRQ5 input (INTC)	TIC5VS input (MTU2S)	—
	PD22 I/O (port)	D22 I/O (BSC)	IRQ6 input (INTC)	TIC5US input (MTU2S)	AUDCK output (AUD)*
	PD23 I/O (port)	D23 I/O (BSC)	IRQ7 input (INTC)	$\overline{\text{AUDSYNC}}$ output (AUD)*	—
	PD24 I/O (port)	D24 I/O (BSC)	DREQ0 input (DMAC)	TIOC4DS I/O (MTU2S)	—
	PD25 I/O (port)	D25 I/O (BSC)	DREQ1 input (DMAC)	TIOC4CS I/O (MTU2S)	—
	PD26 I/O (port)	D26 I/O (BSC)	DACK0 output (DMAC)	TIOC4BS I/O (MTU2S)	—
	PD27 I/O (port)	D27 I/O (BSC)	DACK1 output (DMAC)	TIOC4AS I/O (MTU2S)	—
	PD28 I/O (port)	D28 I/O (BSC)	$\overline{\text{CS2}}$ output (BSC)	TIOC3DS I/O (MTU2S)	—
	PD29 I/O (port)	D29 I/O (BSC)	$\overline{\text{CS3}}$ output (BSC)	TIOC3BS I/O (MTU2S)	—
	PD30 I/O (port)	D30 I/O (BSC)	TIOC3CS I/O (MTU2S)	$\overline{\text{IRQOUT}}$ output (INTC)	—
	PD31 I/O (port)	D31 I/O (BSC)	TIOC3AS I/O (MTU2S)	$\overline{\text{ADTRG}}$ input (A/D)	—

Note: * Only in F-ZTAT version supporting full functions of E10A.

Table 21.11 SH7083 Multiplexed Pins (Port E)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
E	PE0 I/O (port)	DREQ0 input (DMAC)	TIOC0A I/O (MTU2)	TMS input (H-UDI)*	—	—
	PE1 I/O (port)	TEND0 output (DMAC)	TIOC0B I/O (MTU2)	$\overline{\text{TRST}}$ input (H-UDI)*	—	—
	PE2 I/O (port)	DREQ1 input (DMAC)	TIOC0C I/O (MTU2)	TDI input (H-UDI)*	—	—
	PE3 I/O (port)	TEND1 output (DMAC)	TIOC0D I/O (MTU2)	TDO output (H-UDI)*	—	—
	PE4 I/O (port)	TIOC1A I/O (MTU2)	RXD3 input (SCIF)	TCK input (H-UDI)*	—	—
	PE6 I/O (port)	$\overline{\text{CS7}}$ output (BSC)	TIOC2A I/O (MTU2)	SCK3 I/O (SCIF)	—	—
	PE7 I/O (port)	$\overline{\text{BS}}$ output (BSC)	TIOC2B I/O (MTU2)	$\overline{\text{UBCTRG}}$ output (UBC)	RXD2 input (SCI)	SSI I/O (SSU)
	PE8 I/O (port)	TIOC3A I/O (MTU2)	SCK2 I/O (SCI)	SSCK I/O (SSU)	—	—
	PE10 I/O (port)	TIOC3C I/O (MTU2)	TXD2 output (SCI)	SSO I/O (SSU)	—	—
	PE12 I/O (port)	TIOC4A I/O (MTU2)	TXD3 output (SCIF)	$\overline{\text{SCS}}$ I/O (SSU)	—	—
	PE13 I/O (port)	TIOC4B I/O (MTU2)	$\overline{\text{MRES}}$ input (INTC)	$\overline{\text{ASEBRKAK}}$ output (E10A)*	$\overline{\text{ASEBRK}}$ input (E10A)*	—
	PE14 I/O (port)	DACK0 output (DMAC)	TIOC4C I/O (MTU2)	—	—	—
	PE15 I/O (port)	CKE output (BSC)	DACK1 output (DMAC)	TIOC4D I/O (MTU2)	$\overline{\text{IRQOUT}}$ output (INTC)	—

Note: * Only in F-ZTAT version.

Table 21.12 SH7084 Multiplexed Pins (Port E)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
E	PE0 I/O (port)	DREQ0 input (DMAC)	TIOC0A I/O (MTU2)	TMS input (H-UDI)*	—	—
	PE1 I/O (port)	TEND0 output (DMAC)	TIOC0B I/O (MTU2)	$\overline{\text{TRST}}$ input (H-UDI)*	—	—
	PE2 I/O (port)	DREQ1 input (DMAC)	TIOC0C I/O (MTU2)	TDI input (H-UDI)*	—	—
	PE3 I/O (port)	TEND1 output (DMAC)	TIOC0D I/O (MTU2)	TDO output (H-UDI)*	—	—
	PE4 I/O (port)	TIOC1A I/O (MTU2)	RXD3 input (SCIF)	TCK input (H-UDI)*	—	—
	PE5 I/O (port)	$\overline{\text{CS6}}$ output (BSC)	TIOC1B I/O (MTU2)	TXD3 output (SCIF)	$\overline{\text{ASEBRKAK}}$ output (E10A)*	$\overline{\text{ASEBRK}}$ input (E10A)*
	PE6 I/O (port)	$\overline{\text{CS7}}$ output (BSC)	TIOC2A I/O (MTU2)	SCK3 I/O (SCIF)	—	—
	PE7 I/O (port)	$\overline{\text{BS}}$ output (BSC)	TIOC2B I/O (MTU2)	$\overline{\text{UBCTRG}}$ output (UBC)	RXD2 input (SCI)	SSI I/O (SSU)
	PE8 I/O (port)	TIOC3A I/O (MTU2)	SCK2 I/O (SCI)	SSCK I/O (SSU)	—	—
	PE9 I/O (port)	TIOC3B I/O (MTU2)	SCK3 I/O (SCIF)	$\overline{\text{RTS3}}$ output (SCIF)	—	—
	PE10 I/O (port)	TIOC3C I/O (MTU2)	TXD2 output (SCI)	SSO I/O (SSU)	—	—
	PE11 I/O (port)	TIOC3D I/O (MTU2)	RXD3 input (SCIF)	$\overline{\text{CTS3}}$ input (SCIF)	—	—
	PE12 I/O (port)	TIOC4A I/O (MTU2)	TXD3 output (SCIF)	$\overline{\text{SCS}}$ I/O (SSU)	—	—
	PE13 I/O (port)	TIOC4B I/O (MTU2)	$\overline{\text{MRES}}$ input (INTC)	—	—	—
	PE14 I/O (port)	$\overline{\text{AH}}$ output (BSC)	DACK0 output (DMAC)	TIOC4C I/O (MTU2)	—	—
	PE15 I/O (port)	CKE output (BSC)	DACK1 output (DMAC)	TIOC4D I/O (MTU2)	$\overline{\text{IRQOUT}}$ output (INTC)	—

Note: * Only in F-ZTAT version.

Table 21.13 SH7085 Multiplexed Pins (Port E)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
E	PE0 I/O (port)	DREQ0 input (DMAC)	TIOC0A I/O (MTU2)	AUDCK output (AUD)*1	—	—
	PE1 I/O (port)	TEND0 output (DMAC)	TIOC0B I/O (MTU2)	—	—	—
	PE2 I/O (port)	DREQ1 input (DMAC)	TIOC0C I/O (MTU2)	—	—	—
	PE3 I/O (port)	TEND1 output (DMAC)	TIOC0D I/O (MTU2)	AUDATA3 output (AUD)*1	—	—
	PE4 I/O (port)	$\overline{\text{IOIS16}}$ input (BSC)	TIOC1A I/O (MTU2)	RXD3 input (SCIF)	AUDATA2 output (AUD)*1	—
	PE5 I/O (port)	$\overline{\text{CS6}}$/$\overline{\text{CE1B}}$ output (BSC)	TIOC1B I/O (MTU2)	TXD3 output (SCIF)	AUDATA1 output (AUD)*1	—
	PE6 I/O (port)	$\overline{\text{CS7}}$ output (BSC)	TIOC2A I/O (MTU2)	SCK3 I/O (SCIF)	AUDATA0 output (AUD)*1	—
	PE7 I/O (port)	$\overline{\text{BS}}$ output (BSC)	TIOC2B I/O (MTU2)	$\overline{\text{UBCTRG}}$ output (UBC)	RXD2 input (SCI)	SSI I/O (SSU)
	PE8 I/O (port)	TIOC3A I/O (MTU2)	SCK2 I/O (SCI)	SSCK I/O (SSU)	TMS input (H-UDI)*2	—
	PE9 I/O (port)	TIOC3B I/O (MTU2)	SCK3 I/O (SCIF)	$\overline{\text{RTS3}}$ output (SCIF)	$\overline{\text{TRST}}$ input (H-UDI)*2	—
	PE10 I/O (port)	TIOC3C I/O (MTU2)	TXD2 output (SCI)	SSO I/O (SSU)	TDI input (H-UDI)*2	—
	PE11 I/O (port)	TIOC3D I/O (MTU2)	RXD3 input (SCIF)	$\overline{\text{CTS3}}$ input (SCIF)	TDO output (H-UDI)*2	—
	PE12 I/O (port)	TIOC4A I/O (MTU2)	TXD3 output (SCIF)	$\overline{\text{SCS}}$ I/O (SSU)	TCK input (H-UDI)*2	—
	PE13 I/O (port)	TIOC4B I/O (MTU2)	$\overline{\text{MRES}}$ input (INTC)	$\overline{\text{ASEBRKAK}}$ output (E10A)*2	$\overline{\text{ASEBRK}}$ input (E10A)*2	—
	PE14 I/O (port)	$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$ /DQMUU output (BSC)	DACK0 output (DMAC)	TIOC4C I/O (MTU2)	—	—
	PE15 I/O (port)	CKE output (BSC)	DACK1 output (DMAC)	TIOC4D I/O (MTU2)	$\overline{\text{IRQOUT}}$ output (INTC)	—

Notes: 1. Only in F-ZTAT version supporting full functions of E10A.
2. Only in F-ZTAT version.

Table 21.14 SH7086 Multiplexed Pins (Port E)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
E	PE0 I/O (port)	DREQ0 input (DMAC)	TIOC0A I/O (MTU2)	AUDCK output (AUD)*1	—	—
	PE1 I/O (port)	TEND0 output (DMAC)	TIOC0B I/O (MTU2)	—	—	—
	PE2 I/O (port)	DREQ1 input (DMAC)	TIOC0C I/O (MTU2)	—	—	—
	PE3 I/O (port)	TEND1 output (DMAC)	TIOC0D I/O (MTU2)	AUDATA3 output (AUD)*1	—	—
	PE4 I/O (port)	$\overline{\text{IOIS16}}$ input (BSC)	TIOC1A I/O (MTU2)	RXD3 input (SCIF)	AUDATA2 output (AUD)*1	—
	PE5 I/O (port)	$\overline{\text{CS6}}$/$\overline{\text{CE1B}}$ output (BSC)	TIOC1B I/O (MTU2)	TXD3 output (SCIF)	AUDATA1 output (AUD)*1	—
	PE6 I/O (port)	$\overline{\text{CS7}}$ output (BSC)	TIOC2A I/O (MTU2)	SCK3 I/O (SCIF)	AUDATA0 output (AUD)*1	—
	PE7 I/O (port)	$\overline{\text{BS}}$ output (BSC)	TIOC2B I/O (MTU2)	$\overline{\text{UBCTRG}}$ output (UBC)	RXD2 input (SCI)	SSI I/O (SSU)
	PE8 I/O (port)	TIOC3A I/O (MTU2)	SCK2 I/O (SCI)	SSCK I/O (SSU)	TMS input (H-UDI)*2	—
	PE9 I/O (port)	TIOC3B I/O (MTU2)	SCK3 I/O (SCIF)	$\overline{\text{RTS3}}$ output (SCIF)	$\overline{\text{TRST}}$ input (H-UDI)*2	—
	PE10 I/O (port)	TIOC3C I/O (MTU2)	TXD2 output (SCI)	SSO I/O (SSU)	TDI input (H-UDI)*2	—
	PE11 I/O (port)	TIOC3D I/O (MTU2)	RXD3 input (SCIF)	$\overline{\text{CTS3}}$ input (SCIF)	TDO output (H-UDI)*2	—
	PE12 I/O (port)	TIOC4A I/O (MTU2)	TXD3 output (SCIF)	$\overline{\text{SCS}}$ I/O (SSU)	TCK input (H-UDI)*2	—
	PE13 I/O (port)	TIOC4B I/O (MTU2)	$\overline{\text{MRES}}$ input (INTC)	$\overline{\text{ASEBRKAK}}$ output (E10A)*2	$\overline{\text{ASEBRK}}$ input (E10A)*2	—
	PE14 I/O (port)	$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$ /DQMUU output (BSC)	DACK0 output (DMAC)	TIOC4C I/O (MTU2)	—	—
	PE15 I/O (port)	CKE output (BSC)	DACK1 output (DMAC)	TIOC4D I/O (MTU2)	$\overline{\text{IRQOUT}}$ output (INTC)	—
	PE16 I/O (port)	$\overline{\text{CS8}}$ output (BSC)	TIOC3BS I/O (MTU2S)	—	—	—
	PE17 I/O (port)	TIOC3DS I/O (MTU2S)	—	—	—	—

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
E	PE18 I/O (port)	TIOC4AS I/O (MTU2S)	—	—	—	—
	PE19 I/O (port)	TIOC4BS I/O (MTU2S)	—	—	—	—
	PE20 I/O (port)	TIOC4CS I/O (MTU2S)	—	—	—	—
	PE21 I/O (port)	TIOC4DS I/O (MTU2S)	—	—	—	—

Notes: 1. Only in F-ZTAT version supporting full functions of E10A.
2. Only in F-ZTAT version.

Table 21.15 SH7083/SH7084/SH7085 Multiplexed Pins (Port F)

Port	Function 1 (Related Module)	Function 2 (Related Module)
F	PF0 input (port)	AN0 input (A/D)
	PF1 input (port)	AN1 input (A/D)
	PF2 input (port)	AN2 input (A/D)
	PF3 input (port)	AN3 input (A/D)
	PF4 input (port)	AN4 input (A/D)
	PF5 input (port)	AN5 input (A/D)
	PF6 input (port)	AN6 input (A/D)
	PF7 input (port)	AN7 input (A/D)

Note: During A/D conversion, the AN input function is enabled.

Table 21.16 SH7086 Multiplexed Pins (Port F)

Port	Function 1 (Related Module)	Function 2 (Related Module)
F	PF0 input (port)	AN0 input (A/D)
	PF1 input (port)	AN1 input (A/D)
	PF2 input (port)	AN2 input (A/D)
	PF3 input (port)	AN3 input (A/D)
	PF4 input (port)	AN4 input (A/D)
	PF5 input (port)	AN5 input (A/D)
	PF6 input (port)	AN6 input (A/D)
	PF7 input (port)	AN7 input (A/D)
	PF8 input (port)	AN8 input (A/D)
	PF9 input (port)	AN9 input (A/D)
	PF10 input (port)	AN10 input (A/D)
	PF11 input (port)	AN11 input (A/D)
	PF12input (port)	AN12 input (A/D)
	PF13 input (port)	AN13 input (A/D)
	PF14 input (port)	AN14 input (A/D)
	PF15 input (port)	AN15 input (A/D)

Note: During A/D conversion, the AN input function is enabled.

Table 21.17 SH7083 Pin Functions in Each Operating Mode (1)

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
19, 32, 55, 71, 94	Vcc	Vcc	Vcc	Vcc
3, 24, 33, 52, 62, 81, 93	Vss	Vss	Vss	Vss
20, 72	V_{CL}	V_{CL}	V_{CL}	V_{CL}
92	AVcc	AVcc	AVcc	AVcc
88	AVss	AVss	AVss	AVss
91	AVref	AVref	AVref	AVref
73	PLLVss	PLLVss	PLLVss	PLLVss
65	EXTAL	EXTAL	EXTAL	EXTAL
63	XTAL	XTAL	XTAL	XTAL
66	MD0	MD0	MD0	MD0
64	MD1	MD1	MD1	MD1
68	FWE	FWE	FWE	FWE
75	$\overline{RES}$	$\overline{RES}$	$\overline{RES}$	$\overline{RES}$
29	$\overline{WDTOVF}$	$\overline{WDTOVF}$	$\overline{WDTOVF}$	$\overline{WDTOVF}$
67	NMI	NMI	NMI	NMI
27	$\overline{ASEMD0}$	$\overline{ASEMD0}$	$\overline{ASEMD0}$	$\overline{ASEMD0}$
40	PA3	PA3/A24/RXD1	PA3	PA3/A24/RXD1
39	PA4	PA4/A23/TXD1	PA4	PA4/A23/TXD1
38	PA5	PA5/A22/DREQ1/IRQ1/SCK1	PA5	PA5/A22/DREQ1/IRQ1/SCK1
37	PA7	PA7/$\overline{CS3}$/TCLKB	PA7	PA7/$\overline{CS3}$/TCLKB
36	PA8	PA8/RDWR/IRQ2/TCLKC	PA8	PA8/RDWR/IRQ2/TCLKC
35	PA9	PA9/CKE/IRQ3/TCLKD	PA9	PA9/CKE/IRQ3/TCLKD
34	$\overline{CS0}$	PA10/$\overline{CS0}$/$\overline{POE4}$	$\overline{CS0}$	PA10/$\overline{CS0}$/$\overline{POE4}$
31	$\overline{WRL}$	PA12/$\overline{WRL}$/DQMLL/$\overline{POE6}$	$\overline{WRL}$	PA12/$\overline{WRL}$/DQMLL/$\overline{POE6}$
30	$\overline{WRH}$	PA13/$\overline{WRH}$/DQMLU/$\overline{POE7}$	$\overline{WRH}$	PA13/$\overline{WRH}$/DQMLU/$\overline{POE7}$
28	$\overline{RD}$	PA14/$\overline{RD}$	$\overline{RD}$	PA14/$\overline{RD}$

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
74	CK	PA15/CK	CK	PA15/CK
22	A16	PB0/A16/TIC5WS	A16	PB0/A16/TIC5WS
23	A17	PB1/A17/TIC5W	A17	PB1/A17/TIC5W
25	PB2	PB2/IRQ0/$\overline{\text{POE0}}$	PB2	PB2/IRQ0/$\overline{\text{POE0}}$
70	PB4	PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$	PB4	PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$
69	PB5	PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$	PB5	PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$
43	PB6	PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0	PB6	PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0
42	PB7	PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0	PB7	PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0
41	PB8	PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0	PB8	PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0
26	PB9	PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$	PB9	PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$
4	A0	PC0/A0	A0	PC0/A0
5	A1	PC1/A1	A1	PC1/A1
6	A2	PC2/A2	A2	PC2/A2
7	A3	PC3/A3	A3	PC3/A3
8	A4	PC4/A4	A4	PC4/A4
9	A5	PC5/A5	A5	PC5/A5
10	A6	PC6/A6	A6	PC6/A6
11	A7	PC7/A7	A7	PC7/A7
12	A8	PC8/A8	A8	PC8/A8
13	A9	PC9/A9	A9	PC9/A9
14	A10	PC10/A10	A10	PC10/A10
15	A11	PC11/A11	A11	PC11/A11
16	A12	PC12/A12	A12	PC12/A12
17	A13	PC13/A13	A13	PC13/A13
18	A14	PC14/A14	A14	PC14/A14
21	A15	PC15/A15	A15	PC15/A15
61	D0	PD0/D0	D0	PD0/D0
60	D1	PD1/D1	D1	PD1/D1
59	D2	PD2/D2/TIC5U	D2	PD2/D2/TIC5U

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
58	D3	PD3/D3/TIC5V	D3	PD3/D3/TIC5V
57	D4	PD4/D4/TIC5W	D4	PD4/D4/TIC5W
56	D5	PD5/D5/TIC5US	D5	PD5/D5/TIC5US
54	D6	PD6/D6/TIC5VS	D6	PD6/D6/TIC5VS
53	D7	PD7/D7/TIC5WS	D7	PD7/D7/TIC5WS
51	PD8/(AUDATA0*2)	PD8/D8/TIOC3AS	D8/(AUDATA0*2)	PD8/D8/TIOC3AS
50	PD9/(AUDATA1*2)	PD9/D9/TIOC3BS	D9/(AUDATA1*2)	PD9/D9/TIOC3BS
49	PD10 /(AUDATA2*2)	PD10/D10/TIOC3CS	D10/(AUDATA2*2)	PD10/D10/TIOC3CS
48	PD11 /(AUDATA3*2)	PD11/D11/TIOC3DS	D11/(AUDATA3*2)	PD11/D11/TIOC3DS
47	PD12	PD12/D12/TIOC4AS	D12	PD12/D12/TIOC4AS
46	PD13	PD13/D13/TIOC4BS	D13	PD13/D13/TIOC4BS
45	PD14/(AUDCK*2)	PD14/D14/TIOC4CS	D14/(AUDCK*2)	PD14/D14/TIOC4CS
44	PD15 /($\overline{\text{AUDSYNC}}$*2)	PD15/D15/TIOC4DS	D15/($\overline{\text{AUDSYNC}}$*2)	PD15/D15/TIOC4DS
76	PE0/(TMS*1)	PE0/DREQ0/TIOC0A	PE0/(TMS*1)	PE0/DREQ0/TIOC0A
77	PE1/($\overline{\text{TRST}}$*1)	PE1/TEND0/TIOC0B	PE1/($\overline{\text{TRST}}$*1)	PE1/TEND0/TIOC0B
78	PE2/(TDI*1)	PE2/DREQ1/TIOC0C	PE2/(TDI*1)	PE2/DREQ1/TIOC0C
79	PE3/(TDO*1)	PE3/TEND1/TIOC0D	PE3/(TDO*1)	PE3/TEND1/TIOC0D
80	PE4/(TCK*1)	PE4/TIOC1A/RXD3	PE4/(TCK*1)	PE4/TIOC1A/RXD3
95	PE6	PE6/$\overline{\text{CS7}}$/TIOC2A/SCK3	PE6	PE6/$\overline{\text{CS7}}$/TIOC2A/SCK3
96	PE7	PE7/$\overline{\text{BS}}$/TIOC2B/$\overline{\text{UBCTRG}}$ /RXD2/SSI	PE7	PE7/$\overline{\text{BS}}$/TIOC2B/$\overline{\text{UBCTRG}}$ /RXD2/SSI
97	PE8	PE8/TIOC3A/SCK2/SSCK	PE8	PE8/TIOC3A/SCK2/SSCK
98	PE10	PE10/TIOC3C/TXD2/SSO	PE10	PE10/TIOC3C/TXD2/SSO
99	PE12	PE12/TIOC4A/TXD3/$\overline{\text{SCS}}$	PE12	PE12/TIOC4A/TXD3/$\overline{\text{SCS}}$
100	PE13/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE13/TIOC4B/$\overline{\text{MRES}}$	PE13/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE13/TIOC4B/$\overline{\text{MRES}}$
1	PE14	PE14/DACK0/TIOC4C	PE14	PE14/DACK0/TIOC4C

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
2	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{\text{IRQOUT}}$	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{\text{IRQOUT}}$
82	PF0/AN0	PF0/AN0	PF0/AN0	PF0/AN0
83	PF1/AN1	PF1/AN1	PF1/AN1	PF1/AN1
84	PF2/AN2	PF2/AN2	PF2/AN2	PF2/AN2
85	PF3/AN3	PF3/AN3	PF3/AN3	PF3/AN3
86	PF4/AN4	PF4/AN4	PF4/AN4	PF4/AN4
87	PF5/AN5	PF5/AN5	PF5/AN5	PF5/AN5
89	PF6/AN6	PF6/AN6	PF6/AN6	PF6/AN6
90	PF7/AN7	PF7/AN7	PF7/AN7	PF7/AN7

Notes: 1. Fixed to TMS, $\overline{\text{TRST}}$, TDI, TDO, TCK, and $\overline{\text{ASEBRKAK}}$/$\overline{\text{ASEBRK}}$ when using the E10A (in $\overline{\text{ASEMD0}}$ = low).

2. Only in F-ZTAT version supporting full functions of E10A. Fixed as AUD pins when using the AUD function of the E10A.

Table 21.17 SH7083 Pin Functions in Each Operating Mode (2)

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
19, 32, 55, 71, 94	Vcc	Vcc	Vcc	Vcc
3, 24, 33, 52, 62, 81, 93	Vss	Vss	Vss	Vss
20, 72	V_{CL}	V_{CL}	V_{CL}	V_{CL}
92	AVcc	AVcc	AVcc	AVcc
88	AVss	AVss	AVss	AVss
91	AVref	AVref	AVref	AVref
73	PLLVss	PLLVss	PLLVss	PLLVss
65	EXTAL	EXTAL	EXTAL	EXTAL
63	XTAL	XTAL	XTAL	XTAL
66	MD0	MD0	MD0	MD0
64	MD1	MD1	MD1	MD1
68	FWE	FWE	FWE	FWE
75	$\overline{RES}$	$\overline{RES}$	$\overline{RES}$	$\overline{RES}$
29	$\overline{WDTOVF}$	$\overline{WDTOVF}$	$\overline{WDTOVF}$	$\overline{WDTOVF}$
67	NMI	NMI	NMI	NMI
27	$\overline{ASEMD0}$	$\overline{ASEMD0}$	$\overline{ASEMD0}$	$\overline{ASEMD0}$
40	PA3	PA3/A24/RXD1	PA3	PA3/RXD1
39	PA4	PA4/A23/TXD1	PA4	PA4/TXD1
38	PA5	PA5/A22/DREQ1/IRQ1/SCK1	PA5	PA5/DREQ1/IRQ1/SCK1
37	PA7	PA7/$\overline{CS3}$/TCLKB	PA7	PA7/TCLKB
36	PA8	PA8/RDWR/IRQ2/TCLKC	PA8	PA8/IRQ2/TCLKC
35	PA9	PA9/CKE/IRQ3/TCLKD	PA9	PA9/IRQ3/TCLKD
34	PA10	PA10/$\overline{CS0}$/$\overline{POE4}$	PA10	PA10/$\overline{POE4}$
31	PA12	PA12/$\overline{WRL}$/DQMLL/$\overline{POE6}$	PA12	PA12/$\overline{POE6}$
30	PA13	PA13/$\overline{WRH}$/DQMLU/$\overline{POE7}$	PA13	PA13/$\overline{POE7}$
28	PA14	PA14/$\overline{RD}$	PA14	PA14

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
74	CK	PA15/CK	PA15	PA15
22	PB0	PB0/A16/TIC5WS	PB0	PB0/TIC5WS
23	PB1	PB1/A17/TIC5W	PB1	PB1/TIC5W
25	PB2	PB2/IRQ0/$\overline{\text{POE0}}$	PB2	PB2/IRQ0/$\overline{\text{POE0}}$
70	PB4	PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$	PB4	PB4/IRQ2/$\overline{\text{POE2}}$
69	PB5	PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$	PB5	PB5/IRQ3/$\overline{\text{POE3}}$
43	PB6	PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0	PB6	PB6/IRQ4/RXD0
42	PB7	PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0	PB7	PB7/IRQ5/TXD0
41	PB8	PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0	PB8	PB8/IRQ6/SCK0
26	PB9	PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$	PB9	PB9/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$
4	PC0	PC0/A0	PC0	PC0
5	PC1	PC1/A1	PC1	PC1
6	PC2	PC2/A2	PC2	PC2
7	PC3	PC3/A3	PC3	PC3
8	PC4	PC4/A4	PC4	PC4
9	PC5	PC5/A5	PC5	PC5
10	PC6	PC6/A6	PC6	PC6
11	PC7	PC7/A7	PC7	PC7
12	PC8	PC8/A8	PC8	PC8
13	PC9	PC9/A9	PC9	PC9
14	PC10	PC10/A10	PC10	PC10
15	PC11	PC11/A11	PC11	PC11
16	PC12	PC12/A12	PC12	PC12
17	PC13	PC13/A13	PC13	PC13
18	PC14	PC14/A14	PC14	PC14
21	PC15	PC15/A15	PC15	PC15
61	PD0	PD0/D0	PD0	PD0
60	PD1	PD1/D1	PD1	PD1
59	PD2	PD2/D2/TIC5U	PD2	PD2/TIC5U

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
58	PD3	PD3/D3/TIC5V	PD3	PD3/TIC5V
57	PD4	PD4/D4/TIC5W	PD4	PD4/TIC5W
56	PD5	PD5/D5/TIC5US	PD5	PD5/TIC5US
54	PD6	PD6/D6/TIC5VS	PD6	PD6/TIC5VS
53	PD7	PD7/D7/TIC5WS	PD7	PD7/TIC5WS
51	PD8/(AUDATA0*2)	PD8/D8/TIOC3AS	D8/(AUDATA0*2)	PD8/TIOC3AS
50	PD9/(AUDATA1*2)	PD9/D9/TIOC3BS	D9/(AUDATA1*2)	PD9/TIOC3BS
49	PD10 /(AUDATA2*2)	PD10/D10/TIOC3CS	D10/(AUDATA2*2)	PD10/TIOC3CS
48	PD11 /(AUDATA3*2)	PD11/D11/TIOC3DS	D11/(AUDATA3*2)	PD11/TIOC3DS
47	PD12	PD12/D12/TIOC4AS	D12	PD12/TIOC4AS
46	PD13	PD13/D13/TIOC4BS	D13	PD13/TIOC4BS
45	PD14/(AUDCK*2)	PD14/D14/TIOC4CS	D14/(AUDCK*2)	PD14/TIOC4CS
44	PD15 /($\overline{\text{AUDSYNC}}$*2)	PD15/D15/TIOC4DS	D15/($\overline{\text{AUDSYNC}}$*2)	PD15/TIOC4DS
76	PE0/(TMS*1)	PE0/DREQ0/TIOC0A	PE0/(TMS*1)	PE0/DREQ0/TIOC0A
77	PE1/($\overline{\text{TRST}}$*1)	PE1/TEND0/TIOC0B	PE1/($\overline{\text{TRST}}$*1)	PE1/TIOC0B
78	PE2/(TDI*1)	PE2/DREQ1/TIOC0C	PE2/(TDI*1)	PE2/DREQ1/TIOC0C
79	PE3/(TDO*1)	PE3/TEND1/TIOC0D	PE3/(TDO*1)	PE3/TIOC0D
80	PE4/(TCK*1)	PE4/TIOC1A/RXD3	PE4/(TCK*1)	PE4/TIOC1A/RXD3
95	PE6	PE6/$\overline{\text{CS7}}$/TIOC2A/SCK3	PE6	PE6/TIOC2A/SCK3
96	PE7	PE7/$\overline{\text{BS}}$/TIOC2B/$\overline{\text{UBCTRG}}$ /RXD2/SSI	PE7	PE7/TIOC2B/$\overline{\text{UBCTRG}}$ /RXD2/SSI
97	PE8	PE8/TIOC3A/SCK2/SSCK	PE8	PE8/TIOC3A/SCK2/SSCK
98	PE10	PE10/TIOC3C/TXD2/SSO	PE10	PE10/TIOC3C/TXD2/SSO
99	PE12	PE12/TIOC4A/TXD3/$\overline{\text{SCS}}$	PE12	PE12/TIOC4A/TXD3/$\overline{\text{SCS}}$
100	PE13/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE13/TIOC4B/$\overline{\text{MRES}}$	PE13/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE13/TIOC4B/$\overline{\text{MRES}}$
1	PE14	PE14/DACK0/TIOC4C	PE14	PE14/TIOC4C

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
2	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{\text{IRQOUT}}$	PE15	PE15/TIOC4D/$\overline{\text{IRQOUT}}$
82	PF0/AN0	PF0/AN0	PF0/AN0	PF0/AN0
83	PF1/AN1	PF1/AN1	PF1/AN1	PF1/AN1
84	PF2/AN2	PF2/AN2	PF2/AN2	PF2/AN2
85	PF3/AN3	PF3/AN3	PF3/AN3	PF3/AN3
86	PF4/AN4	PF4/AN4	PF4/AN4	PF4/AN4
87	PF5/AN5	PF5/AN5	PF5/AN5	PF5/AN5
89	PF6/AN6	PF6/AN6	PF6/AN6	PF6/AN6
90	PF7/AN7	PF7/AN7	PF7/AN7	PF7/AN7

Notes: 1. Fixed to TMS, $\overline{\text{TRST}}$, TDI, TDO, TCK, and $\overline{\text{ASEBRKAK}}$/$\overline{\text{ASEBRK}}$ when using the E10A (in $\overline{\text{ASEMD0}}$ = low).

2. Only in F-ZTAT version supporting full functions of E10A. Fixed as AUD pins when using the AUD function of the E10A.

Table 21.18 SH7084 Pin Functions in Each Operating Mode (1)

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
21, 37, 65, 80, 103	Vcc	Vcc	Vcc	Vcc
3, 27, 39, 55, 61, 71, 90, 101	Vss	Vss	Vss	Vss
23, 81, 109	VCL	VCL	VCL	VCL
100	AVcc	AVcc	AVcc	AVcc
97	AVss	AVss	AVss	AVss
82	PLLVss	PLLVss	PLLVss	PLLVss
74	EXTAL	EXTAL	EXTAL	EXTAL
72	XTAL	XTAL	XTAL	XTAL
75	MD0	MD0	MD0	MD0
73	MD1	MD1	MD1	MD1
77	FWE	FWE	FWE	FWE
84	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$
35	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$
76	NMI	NMI	NMI	NMI
33	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$
51	PA0	PA0/$\overline{\text{CS4}}$/RXD0	PA0	PA0/$\overline{\text{CS4}}$/RXD0
50	PA1	PA1/$\overline{\text{CS5}}$/TXD0	PA1	PA1/$\overline{\text{CS5}}$/TXD0
49	PA2	PA2/A25/DREQ0/IRQ0/SCK0	PA2	PA2/A25/DREQ0/IRQ0/SCK0
48	PA3	PA3/A24/RXD1	PA3	PA3/A24/RXD1
47	PA4	PA4/A23/TXD1	PA4	PA4/A23/TXD1
46	PA5	PA5/A22/DREQ1/IRQ1/SCK1	PA5	PA5/A22/DREQ1/IRQ1/SCK1
45	PA6	PA6/$\overline{\text{CS2}}$/TCLKA	PA6	PA6/$\overline{\text{CS2}}$/TCLKA
44	PA7	PA7/$\overline{\text{CS3}}$/TCLKB	PA7	PA7/$\overline{\text{CS3}}$/TCLKB
43	PA8	PA8/RDWR/IRQ2/TCLKC	PA8	PA8/RDWR/IRQ2/TCLKC
42	PA9	PA9/CKE/IRQ3/TCLKD	PA9	PA9/CKE/IRQ3/TCLKD
41	$\overline{\text{CS0}}$	PA10/$\overline{\text{CS0}}$/$\overline{\text{POE4}}$	$\overline{\text{CS0}}$	PA10/$\overline{\text{CS0}}$/$\overline{\text{POE4}}$

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
40	$\overline{CS1}$	PA11/$\overline{CS1}$/$\overline{POE5}$	$\overline{CS1}$	PA11/$\overline{CS1}$/$\overline{POE5}$
38	$\overline{WRL}$	PA12/$\overline{WRL}$/DQMLL/$\overline{POE6}$	$\overline{WRL}$	PA12/$\overline{WRL}$/DQMLL/$\overline{POE6}$
36	$\overline{WRH}$	PA13/$\overline{WRH}$/DQMLU/$\overline{POE7}$	$\overline{WRH}$	PA13/$\overline{WRH}$/DQMLU/$\overline{POE7}$
34	$\overline{RD}$	PA14/$\overline{RD}$	$\overline{RD}$	PA14/$\overline{RD}$
83	CK	PA15/CK	CK	PA15/CK
78	PA16	PA16/$\overline{AH}$/CKE	PA16	PA16/$\overline{AH}$/CKE
79	PA17	PA17/$\overline{WAIT}$	PA17	PA17/$\overline{WAIT}$
20	A16	PB0/A16/TIC5WS	A16	PB0/A16/TIC5WS
22	A17	PB1/A17/TIC5W	A17	PB1/A17/TIC5W
24	PB2	PB2/IRQ0/$\overline{POE0}$/SCL	PB2	PB2/IRQ0/$\overline{POE0}$/SCL
25	PB3	PB3/IRQ1/$\overline{POE1}$/SDA	PB3	PB3/IRQ1/$\overline{POE1}$/SDA
26	PB4	PB4/$\overline{RASL}$/IRQ2/$\overline{POE2}$	PB4	PB4/$\overline{RASL}$/IRQ2/$\overline{POE2}$
28	PB5	PB5/$\overline{CASL}$/IRQ3/$\overline{POE3}$	PB5	PB5/$\overline{CASL}$/IRQ3/$\overline{POE3}$
29	PB6	PB6/A18/$\overline{BACK}$/IRQ4/RXD0	PB6	PB6/A18/$\overline{BACK}$/IRQ4/RXD0
30	PB7	PB7/A19/$\overline{BREQ}$/IRQ5/TXD0	PB7	PB7/A19/$\overline{BREQ}$/IRQ5/TXD0
31	PB8	PB8/A20/$\overline{WAIT}$/IRQ6/SCK0	PB8	PB8/A20/$\overline{WAIT}$/IRQ6/SCK0
32	PB9	PB9/A21/IRQ7/$\overline{ADTRG}$/$\overline{POE8}$	PB9	PB9/A21/IRQ7/$\overline{ADTRG}$/$\overline{POE8}$
4	A0	PC0/A0	A0	PC0/A0
5	A1	PC1/A1	A1	PC1/A1
6	A2	PC2/A2	A2	PC2/A2
7	A3	PC3/A3	A3	PC3/A3
8	A4	PC4/A4	A4	PC4/A4
9	A5	PC5/A5	A5	PC5/A5
10	A6	PC6/A6	A6	PC6/A6
11	A7	PC7/A7	A7	PC7/A7
12	A8	PC8/A8	A8	PC8/A8
13	A9	PC9/A9	A9	PC9/A9
14	A10	PC10/A10	A10	PC10/A10
15	A11	PC11/A11	A11	PC11/A11

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
16	A12	PC12/A12	A12	PC12/A12
17	A13	PC13/A13	A13	PC13/A13
18	A14	PC14/A14	A14	PC14/A14
19	A15	PC15/A15	A15	PC15/A15
70	D0	PD0/D0	D0	PD0/D0
69	D1	PD1/D1	D1	PD1/D1
68	D2	PD2/D2/TIC5U	D2	PD2/D2/TIC5U
67	D3	PD3/D3/TIC5V	D3	PD3/D3/TIC5V
66	D4	PD4/D4/TIC5W	D4	PD4/D4/TIC5W
64	D5	PD5/D5/TIC5US	D5	PD5/D5/TIC5US
63	D6	PD6/D6/TIC5VS	D6	PD6/D6/TIC5VS
62	D7	PD7/D7/TIC5WS	D7	PD7/D7/TIC5WS
60	PD8/(AUDATA0*2)	PD8/D8/TIOC3AS	D8/(AUDATA0*2)	PD8/D8/TIOC3AS
59	PD9/(AUDATA1*2)	PD9/D9/TIOC3BS	D9/(AUDATA1*2)	PD9/D9/TIOC3BS
58	PD10 /(AUDATA2*2)	PD10/D10/TIOC3CS	D10/(AUDATA2*2)	PD10/D10/TIOC3CS
57	PD11 /(AUDATA3*2)	PD11/D11/TIOC3DS	D11/(AUDATA3*2)	PD11/D11/TIOC3DS
56	PD12	PD12/D12/TIOC4AS	D12	PD12/D12/TIOC4AS
54	PD13	PD13/D13/TIOC4BS	D13	PD13/D13/TIOC4BS
53	PD14/(AUDCK*2)	PD14/D14/TIOC4CS	D14/(AUDCK*2)	PD14/D14/TIOC4CS
52	PD15 /($\overline{\text{AUDSYNC}}$*2)	PD15/D15/TIOC4DS	D15/($\overline{\text{AUDSYNC}}$*2)	PD15/D15/TIOC4DS
85	PE0/(TMS*1)	PE0/DREQ0/TIOC0A	PE0/(TMS*1)	PE0/DREQ0/TIOC0A
86	PE1/($\overline{\text{TRST}}$*1)	PE1/TEND0/TIOC0B	PE1/($\overline{\text{TRST}}$*1)	PE1/TEND0/TIOC0B
87	PE2/(TDI*1)	PE2/DREQ1/TIOC0C	PE2/(TDI*1)	PE2/DREQ1/TIOC0C
88	PE3/(TDO*1)	PE3/TEND1/TIOC0D	PE3/(TDO*1)	PE3/TEND1/TIOC0D
89	PE4/(TCK*1)	PE4/TIOC1A/RXD3	PE4/(TCK*1)	PE4/TIOC1A/RXD3
102	PE5/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE5/$\overline{\text{CS6}}$/TIOC1B/TXD3	PE5/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE5/$\overline{\text{CS6}}$/TIOC1B/TXD3

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
104	PE6	PE6/$\overline{CS7}$/TIOC2A/SCK3	PE6	PE6/$\overline{CS7}$/TIOC2A/SCK3
105	PE7	PE7/$\overline{BS}$/TIOC2B/$\overline{UBCTRG}$ /RXD2/SSI	PE7	PE7/$\overline{BS}$/TIOC2B/$\overline{UBCTRG}$ /RXD2/SSI
106	PE8	PE8/TIOC3A/SCK2/SSCK	PE8	PE8/TIOC3A/SCK2/SSCK
107	PE9	PE9/TIOC3B/SCK3/$\overline{RTS3}$	PE9	PE9/TIOC3B/SCK3/$\overline{RTS3}$
108	PE10	PE10/TIOC3C/TXD2/SSO	PE10	PE10/TIOC3C/TXD2/SSO
110	PE11	PE11/TIOC3D/RXD3/$\overline{CTS3}$	PE11	PE11/TIOC3D/RXD3/$\overline{CTS3}$
111	PE12	PE12/TIOC4A/TXD3/$\overline{SCS}$	PE12	PE12/TIOC4A/TXD3/$\overline{SCS}$
112	PE13	PE13/TIOC4B/$\overline{MRES}$	PE13	PE13/TIOC4B/$\overline{MRES}$
1	PE14	PE14/$\overline{AH}$/DACK0/TIOC4C	PE14	PE14/$\overline{AH}$/DACK0/TIOC4C
2	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{IRQOUT}$	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{IRQOUT}$
91	PF0/AN0	PF0/AN0	PF0/AN0	PF0/AN0
92	PF1/AN1	PF1/AN1	PF1/AN1	PF1/AN1
93	PF2/AN2	PF2/AN2	PF2/AN2	PF2/AN2
94	PF3/AN3	PF3/AN3	PF3/AN3	PF3/AN3
95	PF4/AN4	PF4/AN4	PF4/AN4	PF4/AN4
96	PF5/AN5	PF5/AN5	PF5/AN5	PF5/AN5
98	PF6/AN6	PF6/AN6	PF6/AN6	PF6/AN6
99	PF7/AN7	PF7/AN7	PF7/AN7	PF7/AN7

Notes: 1. Fixed to TMS, $\overline{TRST}$, TDI, TDO, TCK, and $\overline{ASEBRKAK}$/$\overline{ASEBRK}$ when using the E10A (in $\overline{ASEMD0}$ = low).

2. Only in F-ZTAT version supporting full functions of E10A. Fixed as AUD pins when using the AUD function of the E10A.

Table 21.18 SH7084 Pin Functions in Each Operating Mode (2)

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
21, 37, 65, 80, 103	Vcc	Vcc	Vcc	Vcc
3, 27, 39, 55, 61, 71, 90, 101	Vss	Vss	Vss	Vss
23, 81, 109	VCL	VCL	VCL	VCL
100	AVcc	AVcc	AVcc	AVcc
97	AVss	AVss	AVss	AVss
82	PLLVss	PLLVss	PLLVss	PLLVss
74	EXTAL	EXTAL	EXTAL	EXTAL
72	XTAL	XTAL	XTAL	XTAL
75	MD0	MD0	MD0	MD0
73	MD1	MD1	MD1	MD1
77	FWE	FWE	FWE	FWE
84	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$
35	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$
76	NMI	NMI	NMI	NMI
33	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$
51	PA0	PA0/$\overline{\text{CS4}}$/RXD0	PA0	PA0/RXD0
50	PA1	PA1/$\overline{\text{CS5}}$/TXD0	PA1	PA1/TXD0
49	PA2	PA2/A25/DREQ0/IRQ0/SCK0	PA2	PA2/DREQ0/IRQ0/SCK0
48	PA3	PA3/A24/RXD1	PA3	PA3/RXD1
47	PA4	PA4/A23/TXD1	PA4	PA4/TXD1
46	PA5	PA5/A22/DREQ1/IRQ1/SCK1	PA5	PA5/IRQ1/SCK1
45	PA6	PA6/$\overline{\text{CS2}}$/TCLKA	PA6	PA6/TCLKA
44	PA7	PA7/$\overline{\text{CS3}}$/TCLKB	PA7	PA7/TCLKB
43	PA8	PA8/RDWR/IRQ2/TCLKC	PA8	PA8/IRQ2/TCLKC
42	PA9	PA9/CKE/IRQ3/TCLKD	PA9	PA9/IRQ3/TCLKD
41	PA10	PA10/$\overline{\text{CS0}}$/$\overline{\text{POE4}}$	PA10	PA10/$\overline{\text{POE4}}$

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
40	PA11	PA11/$\overline{\text{CS1}}$/$\overline{\text{POE5}}$	PA11	PA11/$\overline{\text{POE5}}$
38	PA12	PA12/$\overline{\text{WRL}}$/DQMLL/$\overline{\text{POE6}}$	PA12	PA12/$\overline{\text{POE6}}$
36	PA13	PA13/$\overline{\text{WRH}}$/DQMLU/$\overline{\text{POE7}}$	PA13	PA13/$\overline{\text{POE7}}$
34	PA14	PA14/$\overline{\text{RD}}$	PA14	PA14
83	CK	PA15/CK	PA15	PA15
78	PA16	PA16/$\overline{\text{AH}}$/CKE	PA16	PA16
79	PA17	PA17/$\overline{\text{WAIT}}$	PA17	PA17
20	PB0	PB0/A16/TIC5WS	PB0	PB0/TIC5WS
22	PB1	PB1/A17/TIC5W	PB1	PB1/TIC5W
24	PB2	PB2/IRQ0/$\overline{\text{POE0}}$/SCL	PB2	PB2/IRQ0/$\overline{\text{POE0}}$/SCL
25	PB3	PB3/IRQ1/$\overline{\text{POE1}}$/SDA	PB3	PB3/IRQ1/$\overline{\text{POE1}}$/SDA
26	PB4	PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$	PB4	PB4/IRQ2/$\overline{\text{POE2}}$
28	PB5	PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$	PB5	PB5/IRQ3/$\overline{\text{POE3}}$
29	PB6	PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0	PB6	PB6/IRQ4/RXD0
30	PB7	PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0	PB7	PB7/IRQ5/TXD0
31	PB8	PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0	PB8	PB8/IRQ6/SCK0
32	PB9	PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$	PB9	PB9/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$
4	PC0	PC0/A0	PC0	PC0
5	PC1	PC1/A1	PC1	PC1
6	PC2	PC2/A2	PC2	PC2
7	PC3	PC3/A3	PC3	PC3
8	PC4	PC4/A4	PC4	PC4
9	PC5	PC5/A5	PC5	PC5
10	PC6	PC6/A6	PC6	PC6
11	PC7	PC7/A7	PC7	PC7
12	PC8	PC8/A8	PC8	PC8/
13	PC9	PC9/A9	PC9	PC9
14	PC10	PC10/A10	PC10	PC10
15	PC11	PC11/A11	PC11	PC11

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
16	PC12	PC12/A12	PC12	PC12
17	PC13	PC13/A13	PC13	PC13
18	PC14	PC14/A14	PC14	PC14
19	PC15	PC15/A15	PC15	PC15
70	PD0	PD0/D0	PD0	PD0
69	PD1	PD1/D1	PD1	PD1
68	PD2	PD2/D2/TIC5U	PD2	PD2/TIC5U
67	PD3	PD3/D3/TIC5V	PD3	PD3/TIC5V
66	PD4	PD4/D4/TIC5W	PD4	PD4/TIC5W
64	PD5	PD5/D5/TIC5US	PD5	PD5/TIC5US
63	PD6	PD6/D6/TIC5VS	PD6	PD6/TIC5VS
62	PD7	PD7/D7/TIC5WS	PD7	PD7/TIC5WS
60	PD8/(AUDATA0*2)	PD8/D8/TIOC3AS	D8/(AUDATA0*2)	PD8/TIOC3AS
59	PD9/(AUDATA1*2)	PD9/D9/TIOC3BS	D9/(AUDATA1*2)	PD9/TIOC3BS
58	PD10 /(AUDATA2*2)	PD10/D10/TIOC3CS	D10/(AUDATA2*2)	PD10/TIOC3CS
57	PD11 /(AUDATA3*2)	PD11/D11/TIOC3DS	D11/(AUDATA3*2)	PD11/TIOC3DS
56	PD12	PD12/D12/TIOC4AS	D12	PD12/TIOC4AS
54	PD13	PD13/D13/TIOC4BS	D13	PD13/TIOC4BS
53	PD14/(AUDCK*2)	PD14/D14/TIOC4CS	D14/(AUDCK*2)	PD14/TIOC4CS
52	PD15 /($\overline{\text{AUDSYNC}}$*2)	PD15/D15/TIOC4DS	D15/($\overline{\text{AUDSYNC}}$*2)	PD15/TIOC4DS
85	PE0/(TMS*1)	PE0/DREQ0/TIOC0A	PE0/(TMS*1)	PE0/DREQ0/TIOC0A
86	PE1/($\overline{\text{TRST}}$*1)	PE1/TEND0/TIOC0B	PE1/($\overline{\text{TRST}}$*1)	PE1/TIOC0B
87	PE2/(TDI*1)	PE2/DREQ1/TIOC0C	PE2/(TDI*1)	PE2/DREQ1/TIOC0C
88	PE3/(TDO*1)	PE3/TEND1/TIOC0D	PE3/(TDO*1)	PE3/TIOC0D
89	PE4/(TCK*1)	PE4/TIOC1A/RXD3	PE4/(TCK*1)	PE4/TIOC1A/RXD3
102	PE5/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE5/$\overline{\text{CS6}}$/TIOC1B/TXD3	PE5/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE5/TIOC1B/TXD3

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
104	PE6	PE6/$\overline{CS7}$/TIOC2A/SCK3	PE6	PE6/TIOC2A/SCK3
105	PE7	PE7/$\overline{BS}$/TIOC2B/$\overline{UBCTRG}$ /RXD2/SSI	PE7	PE7/TIOC2B/$\overline{UBCTRG}$ /RXD2/SSI
106	PE8	PE8/TIOC3A/SCK2/SSCK	PE8	PE8/TIOC3A/SCK2/SSCK
107	PE9	PE9/TIOC3B/SCK3/$\overline{RTS3}$	PE9	PE9/TIOC3B/SCK3/$\overline{RTS3}$
108	PE10	PE10/TIOC3C/TXD2/SSO	PE10	PE10/TIOC3C/TXD2/SSO
110	PE11	PE11/TIOC3D/RXD3/$\overline{CTS3}$	PE11	PE11/TIOC3D/RXD3/$\overline{CTS3}$
111	PE12	PE12/TIOC4A/TXD3/$\overline{SCS}$	PE12	PE12/TIOC4A/TXD3/$\overline{SCS}$
112	PE13	PE13/TIOC4B/$\overline{MRES}$	PE13	PE13/TIOC4B/$\overline{MRES}$
1	PE14	PE14/$\overline{AH}$/DACK0/TIOC4C	PE14	PE14/TIOC4C
2	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{IRQOUT}$	PE15	PE15/TIOC4D/$\overline{IRQOUT}$
91	PF0/AN0	PF0/AN0	PF0/AN0	PF0/AN0
92	PF1/AN1	PF1/AN1	PF1/AN1	PF1/AN1
93	PF2/AN2	PF2/AN2	PF2/AN2	PF2/AN2
94	PF3/AN3	PF3/AN3	PF3/AN3	PF3/AN3
95	PF4/AN4	PF4/AN4	PF4/AN4	PF4/AN4
96	PF5/AN5	PF5/AN5	PF5/AN5	PF5/AN5
98	PF6/AN6	PF6/AN6	PF6/AN6	PF6/AN6
99	PF7/AN7	PF7/AN7	PF7/AN7	PF7/AN7

Notes: 1. Fixed to TMS, $\overline{TRST}$, TDI, TDO, TCK, and $\overline{ASEBRKAK}$/$\overline{ASEBRK}$ when using the E10A (in $\overline{ASEMD0}$ = low).

2. Only in F-ZTAT version supporting full functions of E10A. Fixed as AUD pins when using the AUD function of the E10A.

Table 21.19 SH7085 Pin Functions in Each Operating Mode (1)

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
12, 26, 40, 63, 77, 85, 104, 112, 135	Vcc	Vcc	Vcc	Vcc
6, 14, 28, 35, 55, 71, 79, 87, 93, 117, 129	Vss	Vss	Vss	Vss
61, 105, 141	VCL	VCL	VCL	VCL
128	AVcc	AVcc	AVcc	AVcc
124	AVss	AVss	AVss	AVss
127	AVref	AVref	AVref	AVref
106	PLLVss	PLLVss	PLLVss	PLLVss
96	EXTAL	EXTAL	EXTAL	EXTAL
94	XTAL	XTAL	XTAL	XTAL
97	MD0	MD0	MD0	MD0
95	MD1	MD1	MD1	MD1
99	FWE	FWE	FWE	FWE
108	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$
44	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$
98	NMI	NMI	NMI	NMI
42	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$
130	PA0	PA0/$\overline{\text{CS4}}$/RXD0	PA0	PA0/$\overline{\text{CS4}}$/RXD0
131	PA1	PA1/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/TXD0	PA1	PA1/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/TXD0
132	PA2	PA2/A25/DREQ0/IRQ0/SCK0	PA2	PA2/A25/DREQ0/IRQ0/SCK0
133	PA3	PA3/A24/RXD1	PA3	PA3/A24/RXD1
134	PA4	PA4/A23/TXD1	PA4	PA4/A23/TXD1
136	PA5	PA5/A22/DREQ1/IRQ1/SCK1	PA5	PA5/A22/DREQ1/IRQ1/SCK1
54	PA6	PA6/$\overline{\text{CS2}}$/TCLKA	PA6	PA6/$\overline{\text{CS2}}$/TCLKA
53	PA7	PA7/$\overline{\text{CS3}}$/TCLKB	PA7	PA7/$\overline{\text{CS3}}$/TCLKB

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
52	PA8	PA8/RDWR/IRQ2/TCLKC	PA8	PA8/RDWR/IRQ2/TCLKC
51	PA9	PA9/$\overline{\text{FRAME}}$/CKE/IRQ3/TCLKD	PA9	PA9/$\overline{\text{FRAME}}$/CKE/IRQ3/TCLKD
50	$\overline{\text{CS0}}$	PA10/$\overline{\text{CS0}}$/$\overline{\text{POE4}}$	$\overline{\text{CS0}}$	PA10/$\overline{\text{CS0}}$/$\overline{\text{POE4}}$
49	$\overline{\text{CS1}}$	PA11/$\overline{\text{CS1}}$/$\overline{\text{POE5}}$	$\overline{\text{CS1}}$	PA11/$\overline{\text{CS1}}$/$\overline{\text{POE5}}$
48	$\overline{\text{WRL}}$	PA12/$\overline{\text{WRL}}$/DQMLL/$\overline{\text{POE6}}$	$\overline{\text{WRL}}$	PA12/$\overline{\text{WRL}}$/DQMLL/$\overline{\text{POE6}}$
47	$\overline{\text{WRH}}$	PA13/$\overline{\text{WRH}}$/$\overline{\text{WE}}$/DQMLU/$\overline{\text{POE7}}$	$\overline{\text{WRH}}$	PA13/$\overline{\text{WRH}}$/$\overline{\text{WE}}$/DQMLU/$\overline{\text{POE7}}$
43	$\overline{\text{RD}}$	PA14/$\overline{\text{RD}}$	$\overline{\text{RD}}$	PA14/$\overline{\text{RD}}$
107	CK	PA15/CK	CK	PA15/CK
100	PA16 /($\overline{\text{AUDSYNC}}$*2)	PA16/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$ /DQMUU/CKE/DREQ2	PA16 /($\overline{\text{AUDSYNC}}$*2)	PA16/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$ /DQMUU/CKE/DREQ2
101	PA17	PA17/$\overline{\text{WAIT}}$/DACK2	PA17	PA17/$\overline{\text{WAIT}}$/DACK2
33	PA18	PA18/$\overline{\text{BREQ}}$/TEND0	PA18	PA18/$\overline{\text{BREQ}}$/TEND0
30	PA19	PA19/$\overline{\text{BACK}}$/TEND1	PA19	PA19/$\overline{\text{BACK}}$/TEND1
29	PA20	PA20/$\overline{\text{CS4}}$/$\overline{\text{RASU}}$	PA20	PA20/$\overline{\text{CS4}}$/$\overline{\text{RASU}}$
4	PA21	PA21/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/$\overline{\text{CASU}}$/TIC5U	PA21	PA21/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/$\overline{\text{CASU}}$/TIC5U
3	PA22	PA22/$\overline{\text{WRHL}}$/$\overline{\text{ICIORD}}$/DQMUL /TIC5V	$\overline{\text{WRHL}}$	PA22/$\overline{\text{WRHL}}$/$\overline{\text{ICIORD}}$/DQMUL /TIC5V
1	PA23	PA23/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$ /DQMUU/TIC5W	$\overline{\text{WRHH}}$	PA23/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$ /DQMUU/TIC5W
102	PA24	PA24/$\overline{\text{CE2A}}$/DREQ3	PA24	PA24/$\overline{\text{CE2A}}$/DREQ3
103	PA25	PA25/$\overline{\text{CE2B}}$/DACK3/$\overline{\text{POE8}}$	PA25	PA25/$\overline{\text{CE2B}}$/DACK3/$\overline{\text{POE8}}$
25	A16	PB0/A16/TIC5WS	A16	PB0/A16/TIC5WS
27	A17	PB1/A17/TIC5W	A17	PB1/A17/TIC5W
31	PB2	PB2/IRQ0/$\overline{\text{POE0}}$/SCL	PB2	PB2/IRQ0/$\overline{\text{POE0}}$/SCL
32	PB3	PB3/IRQ1/$\overline{\text{POE1}}$/SDA	PB3	PB3/IRQ1/$\overline{\text{POE1}}$/SDA
34	PB4	PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$	PB4	PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$
36	PB5	PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$	PB5	PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$
37	PB6	PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0	PB6	PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0
38	PB7	PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0	PB7	PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
39	PB8	PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0	PB8	PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0
41	PB9	PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$	PB9	PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$
7	A0	PC0/A0	A0	PC0/A0
8	A1	PC1/A1	A1	PC1/A1
9	A2	PC2/A2	A2	PC2/A2
10	A3	PC3/A3	A3	PC3/A3
11	A4	PC4/A4	A4	PC4/A4
13	A5	PC5/A5	A5	PC5/A5
15	A6	PC6/A6	A6	PC6/A6
16	A7	PC7/A7	A7	PC7/A7
17	A8	PC8/A8	A8	PC8/A8
18	A9	PC9/A9	A9	PC9/A9
19	A10	PC10/A10	A10	PC10/A10
20	A11	PC11/A11	A11	PC11/A11
21	A12	PC12/A12	A12	PC12/A12
22	A13	PC13/A13	A13	PC13/A13
23	A14	PC14/A14	A14	PC14/A14
24	A15	PC15/A15	A15	PC15/A15
92	D0	PD0/D0	D0	PD0/D0
91	D1	PD1/D1	D1	PD1/D1
90	D2	PD2/D2/TIC5U	D2	PD2/D2/TIC5U
89	D3	PD3/D3/TIC5V	D3	PD3/D3/TIC5V
88	D4	PD4/D4/TIC5W	D4	PD4/D4/TIC5W
86	D5	PD5/D5/TIC5US	D5	PD5/D5/TIC5US
84	D6	PD6/D6/TIC5VS	D6	PD6/D6/TIC5VS
83	D7	PD7/D7/TIC5WS	D7	PD7/D7/TIC5WS
82	D8	PD8/D8/TIOC3AS	D8	PD8/D8/TIOC3AS
81	D9	PD9/D9/TIOC3BS	D9	PD9/D9/TIOC3BS
80	D10	PD10/D10/TIOC3CS	D10	PD10/D10/TIOC3CS

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
78	D11	PD11/D11/TIOC3DS	D11	PD11/D11/TIOC3DS
76	D12	PD12/D12/TIOC4AS	D12	PD12/D12/TIOC4AS
75	D13	PD13/D13/TIOC4BS	D13	PD13/D13/TIOC4BS
74	D14	PD14/D14/TIOC4CS	D14	PD14/D14/TIOC4CS
73	D15	PD15/D15/TIOC4DS	D15	PD15/D15/TIOC4DS
72	PD16 /(AUDATA0*2)	PD16/D16/IRQ0/$\overline{\text{POE4}}$	D16 /(AUDATA0*2)	PD16/D16/IRQ0/$\overline{\text{POE4}}$
70	PD17 /(AUDATA1*2)	PD17/D17/IRQ1/$\overline{\text{POE5}}$	D17 /(AUDATA1*2)	PD17/D17/IRQ1/$\overline{\text{POE5}}$
69	PD18 /(AUDATA2*2)	PD18/D18/IRQ2/$\overline{\text{POE6}}$	D18 /(AUDATA2*2)	PD18/D18/IRQ2/$\overline{\text{POE6}}$
68	PD19 /(AUDATA3*2)	PD19/D19/IRQ3/$\overline{\text{POE7}}$	D19 /(AUDATA3*2)	PD19/D19/IRQ3/$\overline{\text{POE7}}$
67	PD20	PD20/D20/IRQ4/TIC5WS	D20	PD20/D20/IRQ4/TIC5WS
66	PD21	PD21/D21/IRQ5/TIC5VS	D21	PD21/D21/IRQ5/TIC5VS
65	PD22/(AUDCK*2)	PD22/D22/IRQ6/TIC5US	D22/(AUDCK*2)	PD22/D22/IRQ6/TIC5US
64	PD23 /($\overline{\text{AUDSYNC}}$*2)	PD23/D23/IRQ7	D23/($\overline{\text{AUDSYNC}}$*2)	PD23/D23/IRQ7
62	PD24	PD24/D24/DREQ0/TIOC4DS	D24	PD24/D24/DREQ0/TIOC4DS
60	PD25	PD25/D25/DREQ1/TIOC4CS	D25	PD25/D25/DREQ1/TIOC4CS
59	PD26	PD26/D26/DACK0/TIOC4BS	D26	PD26/D26/DACK0/TIOC4BS
58	PD27	PD27/D27/DACK1/TIOC4AS	D27	PD27/D27/DACK1/TIOC4AS
57	PD28	PD28/D28/$\overline{\text{CS2}}$/TIOC3DS	D28	PD28/D28/$\overline{\text{CS2}}$/TIOC3DS
56	PD29	PD29/D29/$\overline{\text{CS3}}$/TIOC3BS	D29	PD29/D29/$\overline{\text{CS3}}$/TIOC3BS
46	PD30	PD30/D30/TIOC3CS/$\overline{\text{IRQOUT}}$	D30	PD30/D30/TIOC3CS/$\overline{\text{IRQOUT}}$
45	PD31	PD31/D31/TIOC3AS/$\overline{\text{ADTRG}}$	D31	PD31/D31/TIOC3AS/$\overline{\text{ADTRG}}$
109	PE0/(AUDCK*2)	PE0/DREQ0/TIOC0A	PE0/(AUDCK*2)	PE0/DREQ0/TIOC0A
110	PE1	PE1/TEND0/TIOC0B	PE1	PE1/TEND0/TIOC0B
111	PE2	PE2/DREQ1/TIOC0C	PE2	PE2/DREQ1/TIOC0C

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
113	PE3/(AUDATA3*2)	PE3/TEND1/TIOC0D	PE3/(AUDATA3*2)	PE3/TEND1/TIOC0D
114	PE4/(AUDATA2*2)	PE4/$\overline{\text{IOIS16}}$/TIOC1A/RXD3	PE4/(AUDATA2*2)	PE4/$\overline{\text{IOIS16}}$/TIOC1A/RXD3
115	PE5/(AUDATA1*2)	PE5/$\overline{\text{CS6}}$/$\overline{\text{CE1B}}$/TIOC1B/TXD3	PE5/(AUDATA1*2)	PE5/$\overline{\text{CS6}}$/$\overline{\text{CE1B}}$/TIOC1B/TXD3
116	PE6/(AUDATA0*2)	PE6/$\overline{\text{CS7}}$/TIOC2A/SCK3	PE6/(AUDATA0*2)	PE6/$\overline{\text{CS7}}$/TIOC2A/SCK3
137	PE7	PE7/$\overline{\text{BS}}$/TIOC2B/$\overline{\text{UBCTRG}}$ /RXD2/SSI	PE7	PE7/$\overline{\text{BS}}$/TIOC2B/$\overline{\text{UBCTRG}}$ /RXD2/SSI
138	PE8/(TMS*1)	PE8/TIOC3A/SCK2/SSCK	PE8/(TMS*1)	PE8/TIOC3A/SCK2/SSCK
139	PE9/($\overline{\text{TRST}}$*1)	PE9/TIOC3B/SCK3/$\overline{\text{RTS3}}$	PE9/($\overline{\text{TRST}}$*1)	PE9/TIOC3B/SCK3/$\overline{\text{RTS3}}$
140	PE10/(TDI*1)	PE10/TIOC3C/TXD2/SSO	PE10/(TDI*1)	PE10/TIOC3C/TXD2/SSO
142	PE11/(TDO*1)	PE11/TIOC3D/RXD3/$\overline{\text{CTS3}}$	PE11/(TDO*1)	PE11/TIOC3D/RXD3/$\overline{\text{CTS3}}$
143	PE12/(TCK*1)	PE12/TIOC4A/TXD3/$\overline{\text{SCS}}$	PE12/(TCK*1)	PE12/TIOC4A/TXD3/$\overline{\text{SCS}}$
144	PE13/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE13/TIOC4B/$\overline{\text{MRES}}$	PE13/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE13/TIOC4B/$\overline{\text{MRES}}$
2	PE14	PE14/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$ /DQMUU/DACK0/TIOC4C	PE14	PE14/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$ /DQMUU/DACK0/TIOC4C
5	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{\text{IRQOUT}}$	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{\text{IRQOUT}}$
118	PF0/AN0	PF0/AN0	PF0/AN0	PF0/AN0
119	PF1/AN1	PF1/AN1	PF1/AN1	PF1/AN1
120	PF2/AN2	PF2/AN2	PF2/AN2	PF2/AN2
121	PF3/AN3	PF3/AN3	PF3/AN3	PF3/AN3
122	PF4/AN4	PF4/AN4	PF4/AN4	PF4/AN4
123	PF5/AN5	PF5/AN5	PF5/AN5	PF5/AN5
125	PF6/AN6	PF6/AN6	PF6/AN6	PF6/AN6
126	PF7/AN7	PF7/AN7	PF7/AN7	PF7/AN7

Notes: 1. Fixed to TMS, $\overline{\text{TRST}}$, TDI, TDO, TCK, and $\overline{\text{ASEBRKAK}}$/$\overline{\text{ASEBRK}}$ when using the E10A (in $\overline{\text{ASEMD0}}$ = low).

2. Only in F-ZTAT version supporting full functions of E10A. Fixed as AUD pins when using the AUD function of the E10A.

Table 21.19 SH7085 Pin Functions in Each Operating Mode (2)

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
12, 26, 40, 63, 77, 85, 104, 112, 135	Vcc	Vcc	Vcc	Vcc
6, 14, 28, 35, 55, 71, 79, 87, 93, 117, 129	Vss	Vss	Vss	Vss
61, 105, 141	VCL	VCL	VCL	VCL
128	AVcc	AVcc	AVcc	AVcc
124	AVss	AVss	AVss	AVss
127	AVref	AVref	AVref	AVref
106	PLLVss	PLLVss	PLLVss	PLLVss
96	EXTAL	EXTAL	EXTAL	EXTAL
94	XTAL	XTAL	XTAL	XTAL
97	MD0	MD0	MD0	MD0
95	MD1	MD1	MD1	MD1
99	FWE	FWE	FWE	FWE
108	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$
44	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$
98	NMI	NMI	NMI	NMI
42	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$
130	PA0	PA0/$\overline{\text{CS4}}$/RXD0	PA0	PA0/RXD0
131	PA1	PA1/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/TXD0	PA1	PA1/TXD0
132	PA2	PA2/A25/DREQ0/IRQ0/SCK0	PA2	PA2/DREQ0/IRQ0/SCK0
133	PA3	PA3/A24/RXD1	PA3	PA3/RXD1
134	PA4	PA4/A23/TXD1	PA4	PA4/TXD1
136	PA5	PA5/A22/DREQ1/IRQ1/SCK1	PA5	PA5/DREQ1/IRQ1/SCK1
54	PA6	PA6/$\overline{\text{CS2}}$/TCLKA	PA6	PA6/TCLKA
53	PA7	PA7/$\overline{\text{CS3}}$/TCLKB	PA7	PA7/TCLKB

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
52	PA8	PA8/RDWR/IRQ2/TCLKC	PA8	PA8/IRQ2/TCLKC
51	PA9	PA9/$\overline{\text{FRAME}}$/CKE/IRQ3/TCLKD	PA9	PA9/IRQ3/TCLKD
50	PA10	PA10/$\overline{\text{CS0}}$/$\overline{\text{POE4}}$	PA10	PA10/$\overline{\text{POE4}}$
49	PA11	PA11/$\overline{\text{CS1}}$/$\overline{\text{POE5}}$	PA11	PA11/$\overline{\text{POE5}}$
48	PA12	PA12/$\overline{\text{WRL}}$/DQMLL/$\overline{\text{POE6}}$	PA12	PA12/$\overline{\text{POE6}}$
47	PA13	PA13/$\overline{\text{WRH}}$/$\overline{\text{WE}}$/DQMLU/$\overline{\text{POE7}}$	PA13	PA13/$\overline{\text{POE7}}$
43	PA14	PA14/$\overline{\text{RD}}$	PA14	PA14
107	CK	PA15/CK	PA15	PA15
100	PA16 /($\overline{\text{AUDSYNC}}$*2)	PA16/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$ /DQMUU/CKE/DREQ2	PA16 /($\overline{\text{AUDSYNC}}$*2)	PA16/DREQ2
101	PA17	PA17/$\overline{\text{WAIT}}$/DACK2	PA17	PA17
33	PA18	PA18/$\overline{\text{BREQ}}$/TEND0	PA18	PA18
30	PA19	PA19/$\overline{\text{BACK}}$/TEND1	PA19	PA19
29	PA20	PA20/$\overline{\text{CS4}}$/$\overline{\text{RASU}}$	PA20	PA20
4	PA21	PA21/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/$\overline{\text{CASU}}$/TIC5U	PA21	PA21/TIC5U
3	PA22	PA22/$\overline{\text{WRHL}}$/$\overline{\text{ICIORD}}$/DQMUL /TIC5V	PA22	PA22/TIC5V
1	PA23	PA23/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$ /DQMUU/TIC5W	PA23	PA23/TIC5W
102	PA24	PA24/$\overline{\text{CE2A}}$/DREQ3	PA24	PA24/DREQ3
103	PA25	PA25/$\overline{\text{CE2B}}$/DACK3/$\overline{\text{POE8}}$	PA25	PA25/$\overline{\text{POE8}}$
25	PB0	PB0/A16/TIC5WS	PB0	PB0/TIC5WS
27	PB1	PB1/A17/TIC5W	PB1	PB1/TIC5W
31	PB2	PB2/IRQ0/$\overline{\text{POE0}}$/SCL	PB2	PB2/IRQ0/$\overline{\text{POE0}}$/SCL
32	PB3	PB3/IRQ1/$\overline{\text{POE1}}$/SDA	PB3	PB3/IRQ1/$\overline{\text{POE1}}$/SDA
34	PB4	PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$	PB4	PB4/IRQ2/$\overline{\text{POE2}}$
36	PB5	PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$	PB5	PB5/IRQ3/$\overline{\text{POE3}}$
37	PB6	PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0	PB6	PB6/IRQ4/RXD0
38	PB7	PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0	PB7	PB7/IRQ5/TXD0

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
39	PB8	PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0	PB8	PB8/IRQ6/SCK0
41	PB9	PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$	PB9	PB9/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$
7	PC0	PC0/A0	PC0	PC0
8	PC1	PC1/A1	PC1	PC1
9	PC2	PC2/A2	PC2	PC2
10	PC3	PC3/A3	PC3	PC3
11	PC4	PC4/A4	PC4	PC4
13	PC5	PC5/A5	PC5	PC5
15	PC6	PC6/A6	PC6	PC6
16	PC7	PC7/A7	PC7	PC7
17	PC8	PC8/A8	PC8	PC8
18	PC9	PC9/A9	PC9	PC9
19	PC10	PC10/A10	PC10	PC10
20	PC11	PC11/A11	PC11	PC11
21	PC12	PC12/A12	PC12	PC12
22	PC13	PC13/A13	PC13	PC13
23	PC14	PC14/A14	PC14	PC14
24	PC15	PC15/A15	PC15	PC15
92	PD0	PD0/D0	PD0	PD0
91	PD1	PD1/D1	PD1	PD1
90	PD2	PD2/D2/TIC5U	PD2	PD2/TIC5U
89	PD3	PD3/D3/TIC5V	PD3	PD3/TIC5V
88	PD4	PD4/D4/TIC5W	PD4	PD4/TIC5W
86	PD5	PD5/D5/TIC5US	PD5	PD5/TIC5US
84	PD6	PD6/D6/TIC5VS	PD6	PD6/TIC5VS
83	PD7	PD7/D7/TIC5WS	PD7	PD7/TIC5WS
82	PD8	PD8/D8/TIOC3AS	PD8	PD8/TIOC3AS
81	PD9	PD9/D9/TIOC3BS	PD9	PD9/TIOC3BS
80	PD10	PD10/D10/TIOC3CS	PD10	PD10/TIOC3CS

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
78	PD11	PD11/D11/TIOC3DS	PD11	PD11/TIOC3DS
76	PD12	PD12/D12/TIOC4AS	PD12	PD12/TIOC4AS
75	PD13	PD13/D13/TIOC4BS	PD13	PD13/TIOC4BS
74	PD14	PD14/D14/TIOC4CS	PD14	PD14/TIOC4CS
73	PD15	PD15/D15/TIOC4DS	PD15	PD15/TIOC4DS
72	PD16 /(AUDATA0*2)	PD16/D16/IRQ0/$\overline{\text{POE4}}$	PD16 /(AUDATA0*2)	PD16/IRQ0/$\overline{\text{POE4}}$
70	PD17 /(AUDATA1*2)	PD17/D17/IRQ1/$\overline{\text{POE5}}$	PD17 /(AUDATA1*2)	PD17/IRQ1/$\overline{\text{POE5}}$
69	PD18 /(AUDATA2*2)	PD18/D18/IRQ2/$\overline{\text{POE6}}$	PD18 /(AUDATA2*2)	PD18/IRQ2/$\overline{\text{POE6}}$
68	PD19 /(AUDATA3*2)	PD19/D19/IRQ3/$\overline{\text{POE7}}$	PD19 /(AUDATA3*2)	PD19/IRQ3/$\overline{\text{POE7}}$
67	PD20	PD20/D20/IRQ4/TIC5WS	PD20	PD20/IRQ4/TIC5WS
66	PD21	PD21/D21/IRQ5/TIC5VS	PD21	PD21/IRQ5/TIC5VS
65	PD22 /(AUDCK*2)	PD22/D22/IRQ6/TIC5US	PD22 /(AUDCK*2)	PD22/IRQ6/TIC5US
64	PD23 /($\overline{\text{AUDSYNC}}$*2)	PD23/D23/IRQ7	PD23 /($\overline{\text{AUDSYNC}}$*2)	PD23/IRQ7
62	PD24	PD24/D24/DREQ0/TIOC4DS	PD24	PD24/DREQ0
60	PD25	PD25/D25/DREQ1/TIOC4CS	PD25	PD25/DREQ1
59	PD26	PD26/D26/DACK0/TIOC4BS	PD26	PD26/TIOC4BS
58	PD27	PD27/D27/DACK1/TIOC4AS	PD27	PD27/TIOC4AS
57	PD28	PD28/D28/$\overline{\text{CS2}}$/TIOC3DS	PD28	PD28/TIOC3DS
56	PD29	PD29/D29/$\overline{\text{CS3}}$/TIOC3BS	PD29	PD29/TIOC3BS
46	PD30	PD30/D30/TIOC3CS/$\overline{\text{IRQOUT}}$	PD30	PD30/TIOC3CS/$\overline{\text{IRQOUT}}$
45	PD31	PD31/D31/TIOC3AS/$\overline{\text{ADTRG}}$	D31	PD31/TIOC3AS/$\overline{\text{ADTRG}}$
109	PE0/(AUDCK*2)	PE0/DREQ0/TIOC0A	PE0/(AUDCK*2)	PE0/DREQ0/TIOC0A
110	PE1	PE1/TEND0/TIOC0B	PE1	PE1/TIOC0B

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
111	PE2	PE2/DREQ1/TIOC0C	PE2	PE2/DREQ1/TIOC0C
113	PE3/(AUDATA3*2)	PE3/TEND1/TIOC0D	PE3/(AUDATA3*2)	PE3/TIOC0D
114	PE4/(AUDATA2*2)	PE4/$\overline{\text{IOIS16}}$/TIOC1A/RXD3	PE4/(AUDATA2*2)	PE4/TIOC1A/RXD3
115	PE5/(AUDATA1*2)	PE5/$\overline{\text{CS6}}$/$\overline{\text{CE1B}}$/TIOC1B/TXD3	PE5/(AUDATA1*2)	PE5/TIOC1B/TXD3
116	PE6/(AUDATA0*2)	PE6/$\overline{\text{CS7}}$/TIOC2A/SCK3	PE6/(AUDATA0*2)	PE6/TIOC2A/SCK3
137	PE7	PE7/$\overline{\text{BS}}$/TIOC2B/$\overline{\text{UBCTRG}}$ /RXD2/SSI	PE7	PE7/TIOC2B/$\overline{\text{UBCTRG}}$ /RXD2/SSI
138	PE8/(TMS*1)	PE8/TIOC3A/SCK2/SSCK	PE8/(TMS*1)	PE8/TIOC3A/SCK2/SSCK
139	PE9/($\overline{\text{TRST}}$*1)	PE9/TIOC3B/SCK3/$\overline{\text{RTS3}}$	PE9/($\overline{\text{TRST}}$*1)	PE9/TIOC3B/SCK3/$\overline{\text{RTS3}}$
140	PE10/(TDI*1)	PE10/TIOC3C/TXD2/SSO	PE10/(TDI*1)	PE10/TIOC3C/TXD2/SSO
142	PE11/(TDO*1)	PE11/TIOC3D/RXD3/$\overline{\text{CTS3}}$	PE11/(TDO*1)	PE11/TIOC3D/RXD3/$\overline{\text{CTS3}}$
143	PE12/(TCK*1)	PE12/TIOC4A/TXD3/$\overline{\text{SCS}}$	PE12/(TCK*1)	PE12/TIOC4A/TXD3/$\overline{\text{SCS}}$
144	PE13/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE13/TIOC4B/$\overline{\text{MRES}}$	PE13/($\overline{\text{ASEBRKAK}}$ /$\overline{\text{ASEBRK}}$*1)	PE13/TIOC4B/$\overline{\text{MRES}}$
2	PE14	PE14/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$ /DQMUU/DACK0/TIOC4C	PE14	PE14/TIOC4C
5	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{\text{IRQOUT}}$	PE15	PE15/TIOC4D/$\overline{\text{IRQOUT}}$
118	PF0/AN0	PF0/AN0	PF0/AN0	PF0/AN0
119	PF1/AN1	PF1/AN1	PF1/AN1	PF1/AN1
120	PF2/AN2	PF2/AN2	PF2/AN2	PF2/AN2
121	PF3/AN3	PF3/AN3	PF3/AN3	PF3/AN3
122	PF4/AN4	PF4/AN4	PF4/AN4	PF4/AN4
123	PF5/AN5	PF5/AN5	PF5/AN5	PF5/AN5
125	PF6/AN6	PF6/AN6	PF6/AN6	PF6/AN6
126	PF7/AN7	PF7/AN7	PF7/AN7	PF7/AN7

Notes: 1. Fixed to TMS, $\overline{\text{TRST}}$, TDI, TDO, TCK, and $\overline{\text{ASEBRKAK}}$/$\overline{\text{ASEBRK}}$ when using the E10A (in $\overline{\text{ASEMD0}}$ = low).

2. Only in F-ZTAT version supporting full functions of E10A. Fixed as AUD pins when using the AUD function of the E10A.

Table 21.20 SH7086 Pin Functions in Each Operating Mode (1)

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
11, 21, 35, 48, 66, 74, 94, 102, 109, 128, 136, 168	Vcc	Vcc	Vcc	Vcc
8, 23, 44, 57, 64, 86, 96, 104, 117, 141, 163	Vss	Vss	Vss	Vss
37, 71, 129, 173	V_{CL}	V_{CL}	V_{CL}	V_{CL}
151, 162	AVcc	AVcc	AVcc	AVcc
142, 156	AVss	AVss	AVss	AVss
161	AVref	AVref	AVref	AVref
130	PLLVss	PLLVss	PLLVss	PLLVss
120	EXTAL	EXTAL	EXTAL	EXTAL
118	XTAL	XTAL	XTAL	XTAL
121	MD0	MD0	MD0	MD0
119	MD1	MD1	MD1	MD1
123	FWE	FWE	FWE	FWE
132	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$
53	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$
122	NMI	NMI	NMI	NMI
51	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$
164	PA0	PA0/$\overline{\text{CS4}}$/RXD0	PA0	PA0/$\overline{\text{CS4}}$/RXD0
165	PA1	PA1/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/TXD0	PA1	PA1/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/TXD0
166	PA2	PA2/A25/DREQ0/IRQ0/SCK0	PA2	PA2/A25/DREQ0/IRQ0/SCK0
167	PA3	PA3/A24/RXD1	PA3	PA3/A24/RXD1
169	PA4	PA4/A23/TXD1	PA4	PA4/A23/TXD1

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
170	PA5	PA5/A22/DREQ1/IRQ1/SCK1	PA5	PA5/A22/DREQ1/IRQ1/SCK1
80	PA6	PA6/$\overline{CS2}$/TCLKA	PA6	PA6/$\overline{CS2}$/TCLKA
79	PA7	PA7/$\overline{CS3}$/TCLKB	PA7	PA7/$\overline{CS3}$/TCLKB
78	PA8	PA8/RDWR/IRQ2/TCLKC	PA8	PA8/RDWR/IRQ2/TCLKC
77	PA9	PA9/$\overline{FRAME}$/CKE/IRQ3/TCLKD	PA9	PA9/$\overline{FRAME}$/CKE/IRQ3/TCLKD
76	$\overline{CS0}$	PA10/$\overline{CS0}$/$\overline{POE4}$	$\overline{CS0}$	PA10/$\overline{CS0}$/$\overline{POE4}$
75	$\overline{CS1}$	PA11/$\overline{CS1}$/$\overline{POE5}$	$\overline{CS1}$	PA11/$\overline{CS1}$/$\overline{POE5}$
73	$\overline{WRL}$	PA12/$\overline{WRL}$/DQMLL/$\overline{POE6}$	$\overline{WRL}$	PA12/$\overline{WRL}$/DQMLL/$\overline{POE6}$
72	$\overline{WRH}$	PA13/$\overline{WRH}$/DQMLU/$\overline{WE}$/$\overline{POE7}$	$\overline{WRH}$	PA13/$\overline{WRH}$/DQMLU/$\overline{WE}$/$\overline{POE7}$
52	$\overline{RD}$	PA14/$\overline{RD}$	$\overline{RD}$	PA14/$\overline{RD}$
131	CK	PA15/CK	CK	PA15/CK
124	PA16 /($\overline{AUDSYNC}$*2)	PA16/$\overline{WRHH}$/$\overline{ICIOWR}$/$\overline{AH}$ /DQMUU/CKE/DREQ2	PA16 /($\overline{AUDSYNC}$*2)	PA16/$\overline{WRHH}$/$\overline{ICIOWR}$/$\overline{AH}$ /DQMUU/CKE/DREQ2
125	PA17	PA17/$\overline{WAIT}$/DACK2	PA17	PA17/$\overline{WAIT}$/DACK2
42	PA18	PA18/$\overline{BREQ}$/TEND0	PA18	PA18/$\overline{BREQ}$/TEND0
39	PA19	PA19/$\overline{BACK}$/TEND1	PA19	PA19/$\overline{BACK}$/TEND1
38	PA20	PA20/$\overline{CS4}$/$\overline{RASU}$	PA20	PA20/$\overline{CS4}$/$\overline{RASU}$
6	PA21	PA21/$\overline{CS5}$/$\overline{CE1A}$/$\overline{CASU}$/TIC5U	PA21	PA21/$\overline{CS5}$/$\overline{CE1A}$/$\overline{CASU}$/TIC5U
5	PA22	PA22/$\overline{WRHL}$/$\overline{ICIORD}$/DQMUL /TIC5V	$\overline{WRHL}$	PA22/$\overline{WRHL}$/$\overline{ICIORD}$/DQMUL /TIC5V
3	PA23	PA23/$\overline{WRHH}$/$\overline{ICIOWR}$/$\overline{AH}$ /DQMUU/TIC5W	$\overline{WRHH}$	PA23/$\overline{WRHH}$/$\overline{ICIOWR}$/$\overline{AH}$ /DQMUU/TIC5W
126	PA24	PA24/$\overline{CE2A}$/DREQ3	PA24	PA24/$\overline{CE2A}$/DREQ3
127	PA25	PA25/$\overline{CE2B}$/DACK3/$\overline{POE8}$	PA25	PA25/$\overline{CE2B}$/DACK3/$\overline{POE8}$
63	PA26	PA26/A26/IRQ0	PA26	PA26/A26/IRQ0
65	PA27	PA27/A27/IRQ1	PA27	PA27/A27/IRQ1
67	PA28	PA28/A28/IRQ2	PA28	PA28/A28/IRQ2
68	PA29	PA29/A29/IRQ3	PA29	PA29/A29/IRQ3
34	A16	PB0/A16/TIC5WS	A16	PB0/A16/TIC5WS
36	A17	PB1/A17/TIC5W	A17	PB1/A17/TIC5W

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
40	PB2	PB2/IRQ0/$\overline{\text{POE0}}$/SCL	PB2	PB2/IRQ0/$\overline{\text{POE0}}$/SCL
41	PB3	PB3/IRQ1/$\overline{\text{POE1}}$/SDA	PB3	PB3/IRQ1/$\overline{\text{POE1}}$/SDA
43	PB4	PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$	PB4	PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$
45	PB5	PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$	PB5	PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$
46	PB6	PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0	PB6	PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0
47	PB7	PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0	PB7	PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0
49	PB8	PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0	PB8	PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0
50	PB9	PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$	PB9	PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$
16	A0	PC0/A0	A0	PC0/A0
17	A1	PC1/A1	A1	PC1/A1
18	A2	PC2/A2	A2	PC2/A2
19	A3	PC3/A3	A3	PC3/A3
20	A4	PC4/A4	A4	PC4/A4
22	A5	PC5/A5	A5	PC5/A5
24	A6	PC6/A6	A6	PC6/A6
25	A7	PC7/A7	A7	PC7/A7
26	A8	PC8/A8	A8	PC8/A8
27	A9	PC9/A9	A9	PC9/A9
28	A10	PC10/A10	A10	PC10/A10
29	A11	PC11/A11	A11	PC11/A11
30	A12	PC12/A12	A12	PC12/A12
31	A13	PC13/A13	A13	PC13/A13
32	A14	PC14/A14	A14	PC14/A14
33	A15	PC15/A15	A15	PC15/A15
54	A18	PC18/A18	A18	PC18/A18
55	A19	PC19/A19	A19	PC19/A19
56	A20	PC20/A20	A20	PC20/A20
58	A21	PC21/A21	A21	PC21/A21

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
59	A22	PC22/A22	A22	PC22/A22
60	A23	PC23/A23	A23	PC23/A23
61	A24	PC24/A24	A24	PC24/A24
62	A25	PC25/A25	A25	PC25/A25
116	D0	PD0/D0	D0	PD0/D0
115	D1	PD1/D1	D1	PD1/D1
114	D2	PD2/D2/TIC5U	D2	PD2/D2/TIC5U
113	D3	PD3/D3/TIC5V	D3	PD3/D3/TIC5V
112	D4	PD4/D4/TIC5W	D4	PD4/D4/TIC5W
111	D5	PD5/D5/TIC5US	D5	PD5/D5/TIC5US
110	D6	PD6/D6/TIC5VS	D6	PD6/D6/TIC5VS
108	D7	PD7/D7/TIC5WS	D7	PD7/D7/TIC5WS
107	D8	PD8/D8/TIOC3AS	D8	PD8/D8/TIOC3AS
106	D9	PD9/D9/TIOC3BS	D9	PD9/D9/TIOC3BS
105	D10	PD10/D10/TIOC3CS	D10	PD10/D10/TIOC3CS
103	D11	PD11/D11/TIOC3DS	D11	PD11/D11/TIOC3DS
101	D12	PD12/D12/TIOC4AS	D12	PD12/D12/TIOC4AS
100	D13	PD13/D13/TIOC4BS	D13	PD13/D13/TIOC4BS
99	D14	PD14/D14/TIOC4CS	D14	PD14/D14/TIOC4CS
98	D15	PD15/D15/TIOC4DS	D15	PD15/D15/TIOC4DS
97	PD16 /(AUDATA0*2)	PD16/D16/IRQ0/$\overline{\text{POE4}}$	D16/(AUDATA0*2)	PD16/D16/IRQ0/$\overline{\text{POE4}}$
95	PD17 /(AUDATA1*2)	PD17/D17/IRQ1/$\overline{\text{POE5}}$	D17/(AUDATA1*2)	PD17/D17/IRQ1/$\overline{\text{POE5}}$
93	PD18 /(AUDATA2*2)	PD18/D18/IRQ2/$\overline{\text{POE6}}$	D18/(AUDATA2*2)	PD18/D18/IRQ2/$\overline{\text{POE6}}$
92	PD19 /(AUDATA3*2)	PD19/D19/IRQ3/$\overline{\text{POE7}}$	D19/(AUDATA3*2)	PD19/D19/IRQ3/$\overline{\text{POE7}}$
91	PD20	PD20/D20/IRQ4/TIC5WS	D20	PD20/D20/IRQ4/TIC5WS

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
90	PD21	PD21/D21/IRQ5/TIC5VS	D21	PD21/D21/IRQ5/TIC5VS
89	PD22/(AUDCK*2)	PD22/D22/IRQ6/TIC5US	D22/(AUDCK*2)	PD22/D22/IRQ6/TIC5US
88	PD23 /(AUDSYNC*2)	PD23/D23/IRQ7	D23/(AUDSYNC*2)	PD23/D23/IRQ7
87	PD24	PD24/D24/DREQ0/TIOC4DS	D24	PD24/D24/DREQ0/TIOC4DS
85	PD25	PD25/D25/DREQ1/TIOC4CS	D25	PD25/D25/DREQ1/TIOC4CS
84	PD26	PD26/D26/DACK0/TIOC4BS	D26	PD26/D26/DACK0/TIOC4BS
83	PD27	PD27/D27/DACK1/TIOC4AS	D27	PD27/D27/DACK1/TIOC4AS
82	PD28	PD28/D28/CS2/TIOC3DS	D28	PD28/D28/CS2/TIOC3DS
81	PD29	PD29/D29/CS3/TIOC3BS	D29	PD29/D29/CS3/TIOC3BS
70	PD30	PD30/D30/TIOC3CS/IRQOUT	D30	PD30/D30/TIOC3CS/IRQOUT
69	PD31	PD31/D31/TIOC3AS/ADTRG	D31	PD31/D31/TIOC3AS/ADTRG
133	PE0/(AUDCK*2)	PE0/DREQ0/TIOC0A	PE0/(AUDCK*2)	PE0/DREQ0/TIOC0A
134	PE1	PE1/TEND0/TIOC0B	PE1	PE1/TEND0/TIOC0B
135	PE2	PE2/DREQ1/TIOC0C	PE2	PE2/DREQ1/TIOC0C
137	PE3/(AUDATA3*2)	PE3/TEND1/TIOC0D	PE3/(AUDATA3*2)	PE3/TEND1/TIOC0D
138	PE4/(AUDATA2*2)	PE4/IOIS16/TIOC1A/RXD3	PE4/(AUDATA2*2)	PE4/IOIS16/TIOC1A/RXD3
139	PE5/(AUDATA1*2)	PE5/CS6/CE1B/TIOC1B/TXD3	PE5/(AUDATA1*2)	PE5/CS6/CE1B/TIOC1B/TXD3
140	PE6/(AUDATA0*2)	PE6/CS7/TIOC2A/SCK3	PE6/(AUDATA0*2)	PE6/CS7/TIOC2A/SCK3
171	PE7	PE7/BS/TIOC2B/UBCTRG /RXD2/SSI	PE7	PE7/BS/TIOC2B/UBCTRG /RXD2/SSI
172	PE8/(TMS*1)	PE8/TIOC3A/SCK2/SSCK	PE8/(TMS*1)	PE8/TIOC3A/SCK2/SSCK
174	PE9/(TRST*1)	PE9/TIOC3B/SCK3/RTS3	PE9/(TRST*1)	PE9/TIOC3B/SCK3/RTS3
175	PE10/(TDI*1)	PE10/TIOC3C/TXD2/SSO	PE10/(TDI*1)	PE10/TIOC3C/TXD2/SSO
176	PE11/(TDO*1)	PE11/TIOC3D/RXD3/CTS3	PE11/(TDO*1)	PE11/TIOC3D/RXD3/CTS3
1	PE12/(TCK*1)	PE12/TIOC4A/TXD3/SCS	PE12/(TCK*1)	PE12/TIOC4A/TXD3/SCS
2	PE13/(ASEBRKAK /ASEBRK*1)	PE13/TIOC4B/MRES	PE13/(ASEBRKAK /ASEBRK*1)	PE13/TIOC4B/MRES
4	PE14	PE14/WRHH/ICIOWR/AH /DQMUU/DACK0/TIOC4C	PE14	PE14/WRHH/ICIOWR/AH /DQMUU/DACK0/TIOC4C

	Pin Name			
	On-Chip ROM Disabled (MCU Mode 0)		On-Chip ROM Disabled (MCU Mode 1)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
7	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{\text{IRQOUT}}$	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{\text{IRQOUT}}$
9	PE16	PE16/$\overline{\text{CS8}}$/TIOC3BS	PE16	PE16/$\overline{\text{CS8}}$/TIOC3BS
10	PE17	PE17/TIOC3DS	PE17	PE17/TIOC3DS
12	PE18	PE18/TIOC4AS	PE18	PE18/TIOC4AS
13	PE19	PE19/TIOC4BS	PE19	PE19/TIOC4BS
14	PE20	PE20/TIOC4CS	PE20	PE20/TIOC4CS
15	PE21	PE21/TIOC4DS	PE21	PE21/TIOC4DS
143	PF0/AN0	PF0/AN0	PF0/AN0	PF0/AN0
144	PF1/AN1	PF1/AN1	PF1/AN1	PF1/AN1
147	PF2/AN2	PF2/AN2	PF2/AN2	PF2/AN2
148	PF3/AN3	PF3/AN3	PF3/AN3	PF3/AN3
152	PF4/AN4	PF4/AN4	PF4/AN4	PF4/AN4
153	PF5/AN5	PF5/AN5	PF5/AN5	PF5/AN5
157	PF6/AN6	PF6/AN6	PF6/AN6	PF6/AN6
158	PF7/AN7	PF7/AN7	PF7/AN7	PF7/AN7
145	PF8/AN8	PF8/AN8	PF8/AN8	PF8/AN8
146	PF9/AN9	PF9/AN9	PF9/AN9	PF9/AN9
149	PF10/AN10	PF10/AN10	PF10/AN10	PF10/AN10
150	PF11/AN11	PF11/AN11	PF11/AN11	PF11/AN11
154	PF12/AN12	PF12/AN12	PF12/AN12	PF12/AN12
155	PF13/AN13	PF13/AN13	PF13/AN13	PF13/AN13
159	PF14/AN14	PF14/AN14	PF14/AN14	PF14/AN14
160	PF15/AN15	PF15/AN15	PF15/AN15	PF15/AN15

Notes: 1. Fixed to TMS, $\overline{\text{TRST}}$, TDI, TDO, TCK, and $\overline{\text{ASEBRKAK}}$/$\overline{\text{ASEBRK}}$ when using the E10A (in $\overline{\text{ASEMD0}}$ = low).

2. Only in F-ZTAT version supporting full functions of E10A. Fixed as AUD pins when using the AUD function of the E10A.

Table 21.20 SH7086 Pin Functions in Each Operating Mode (2)

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
11, 21, 35, 48, 66, 74, 94, 102, 109, 128, 136, 168	Vcc	Vcc	Vcc	Vcc
8, 23, 44, 57, 64, 86, 96, 104, 117, 141, 163	Vss	Vss	Vss	Vss
37, 71, 129, 173	V_{CL}	V_{CL}	V_{CL}	V_{CL}
151, 162	AVcc	AVcc	AVcc	AVcc
142, 156	AVss	AVss	AVss	AVss
161	AVref	AVref	AVref	AVref
130	PLLVss	PLLVss	PLLVss	PLLVss
120	EXTAL	EXTAL	EXTAL	EXTAL
118	XTAL	XTAL	XTAL	XTAL
121	MD0	MD0	MD0	MD0
119	MD1	MD1	MD1	MD1
123	FWE	FWE	FWE	FWE
132	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$
53	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$
122	NMI	NMI	NMI	NMI
51	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$	$\overline{\text{ASEMD0}}$
164	PA0	PA0/$\overline{\text{CS4}}$/RXD0	PA0	PA0/RXD0
165	PA1	PA1/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/TXD0	PA1	PA1/TXD0
166	PA2	PA2/A25/DREQ0/IRQ0/SCK0	PA2	PA2/DREQ0/IRQ0/SCK0
167	PA3	PA3/A24/RXD1	PA3	PA3/RXD1
169	PA4	PA4/A23/TXD1	PA4	PA4/TXD1

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
170	PA5	PA5/A22/DREQ1/IRQ1/SCK1	PA5	PA5/DREQ1/IRQ1/SCK1
80	PA6	PA6/$\overline{CS2}$/TCLKA	PA6	PA6/TCLKA
79	PA7	PA7/$\overline{CS3}$/TCLKB	PA7	PA7/TCLKB
78	PA8	PA8/RDWR/IRQ2/TCLKC	PA8	PA8/IRQ2/TCLKC
77	PA9	PA9/$\overline{FRAME}$/CKE/IRQ3/TCLKD	PA9	PA9/IRQ3/TCLKD
76	PA10	PA10/$\overline{CS0}$/$\overline{POE4}$	PA10	PA10/$\overline{POE4}$
75	PA11	PA11/$\overline{CS1}$/$\overline{POE5}$	PA11	PA11/$\overline{POE5}$
73	PA12	PA12/$\overline{WRL}$/DQMLL/$\overline{POE6}$	PA12	PA12/$\overline{POE6}$
72	PA13	PA13/$\overline{WRH}$/DQMLU/$\overline{WE}$/$\overline{POE7}$	PA13	PA13/$\overline{POE7}$
52	PA14	PA14/$\overline{RD}$	PA14	PA14
131	CK	PA15/CK	PA15	PA15
124	PA16 /($\overline{AUDSYNC}$*2)	PA16/$\overline{WRHH}$/$\overline{ICIOWR}$/$\overline{AH}$ /DQMUU/CKE/DREQ2	PA16 /($\overline{AUDSYNC}$*2)	PA16/DREQ2
125	PA17	PA17/$\overline{WAIT}$/DACK2	PA17	PA17
42	PA18	PA18/$\overline{BREQ}$/TEND0	PA18	PA18
39	PA19	PA19/$\overline{BACK}$/TEND1	PA19	PA19
38	PA20	PA20/$\overline{CS4}$/$\overline{RASU}$	PA20	PA20
6	PA21	PA21/$\overline{CS5}$/$\overline{CE1A}$/$\overline{CASU}$/TIC5U	PA21	PA21/TIC5U
5	PA22	PA22/$\overline{WRHL}$/$\overline{ICIORD}$/DQMUL /TIC5V	PA22	PA22/TIC5V
3	PA23	PA23/$\overline{WRHH}$/$\overline{ICIOWR}$/$\overline{AH}$ /DQMUU/TIC5W	PA23	PA23/TIC5W
126	PA24	PA24/$\overline{CE2A}$/DREQ3	PA24	PA24/DREQ3
127	PA25	PA25/$\overline{CE2B}$/DACK3/$\overline{POE8}$	PA25	PA25/$\overline{POE8}$
63	PA26	PA26/A26/IRQ0	PA26	PA26/IRQ0
65	PA27	PA27/A27/IRQ1	PA27	PA27/IRQ1
67	PA28	PA28/A28/IRQ2	PA28	PA28/IRQ2
68	PA29	PA29/A29/IRQ3	PA29	PA29/IRQ3
34	PB0	PB0/A16/TIC5WS	PB0	PB0/TIC5WS

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
36	PB1	PB1/A17/TIC5W	PB1	PB1/TIC5W
40	PB2	PB2/IRQ0/$\overline{\text{POE0}}$/SCL	PB2	PB2/IRQ0/$\overline{\text{POE0}}$/SCL
41	PB3	PB3/IRQ1/$\overline{\text{POE1}}$/SDA	PB3	PB3/IRQ1/$\overline{\text{POE1}}$/SDA
43	PB4	PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$	PB4	PB4/IRQ2/$\overline{\text{POE2}}$
45	PB5	PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$	PB5	PB5/IRQ3/$\overline{\text{POE3}}$
46	PB6	PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0	PB6	PB6/IRQ4/RXD0
47	PB7	PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0	PB7	PB7/IRQ5/TXD0
49	PB8	PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0	PB8	PB8/IRQ6/SCK0
50	PB9	PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$	PB9	PB9/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$
16	PC0	PC0/A0	PC0	PC0
17	PC1	PC1/A1	PC1	PC1
18	PC2	PC2/A2	PC2	PC2
19	PC3	PC3/A3	PC3	PC3
20	PC4	PC4/A4	PC4	PC4
22	PC5	PC5/A5	PC5	PC5
24	PC6	PC6/A6	PC6	PC6
25	PC7	PC7/A7	PC7	PC7
26	PC8	PC8/A8	PC8	PC8
27	PC9	PC9/A9	PC9	PC9
28	PC10	PC10/A10	PC10	PC10
29	PC11	PC11/A11	PC11	PC11
30	PC12	PC12/A12	PC12	PC12
31	PC13	PC13/A13	PC13	PC13
32	PC14	PC14/A14	PC14	PC14
33	PC15	PC15/A15	PC15	PC15
54	PC18	PC18/A18	PC18	PC18
55	PC19	PC19/A19	PC19	PC19
56	PC20	PC20/A20	PC20	PC20
58	PC21	PC21/A21	PC21	PC21

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
59	PC22	PC22/A22	PC22	PC22
60	PC23	PC23/A23	PC23	PC23
61	PC24	PC24/A24	PC24	PC24
62	PC25	PC25/A25	PC25	PC25
116	PD0	PD0/D0	PD0	PD0
115	PD1	PD1/D1	PD1	PD1
114	PD2	PD2/D2/TIC5U	PD2	PD2/TIC5U
113	PD3	PD3/D3/TIC5V	PD3	PD3/TIC5V
112	PD4	PD4/D4/TIC5W	PD4	PD4/TIC5W
111	PD5	PD5/D5/TIC5US	PD5	PD5/TIC5US
110	PD6	PD6/D6/TIC5VS	PD6	PD6/TIC5VS
108	PD7	PD7/D7/TIC5WS	PD7	PD7/TIC5WS
107	PD8	PD8/D8/TIOC3AS	PD8	PD8/TIOC3AS
106	PD9	PD9/D9/TIOC3BS	PD9	PD9/TIOC3BS
105	PD10	PD10/D10/TIOC3CS	PD10	PD10/TIOC3CS
103	PD11	PD11/D11/TIOC3DS	PD11	PD11/TIOC3DS
101	PD12	PD12/D12/TIOC4AS	PD12	PD12/TIOC4AS
100	PD13	PD13/D13/TIOC4BS	PD13	PD13/TIOC4BS
99	PD14	PD14/D14/TIOC4CS	PD14	PD14/TIOC4CS
98	PD15	PD15/D15/TIOC4DS	PD15	PD15/TIOC4DS
97	PD16 /(AUDATA0*2)	PD16/D16/IRQ0/$\overline{\text{POE4}}$	PD16 /(AUDATA0*2)	PD16/IRQ0/$\overline{\text{POE4}}$
95	PD17 /(AUDATA1*2)	PD17/D17/IRQ1/$\overline{\text{POE5}}$	PD17 /(AUDATA1*2)	PD17/IRQ1/$\overline{\text{POE5}}$
93	PD18 /(AUDATA2*2)	PD18/D18/IRQ2/$\overline{\text{POE6}}$	PD18 /(AUDATA2*2)	PD18/IRQ2/$\overline{\text{POE6}}$
92	PD19 /(AUDATA3*2)	PD19/D19/IRQ3/$\overline{\text{POE7}}$	PD19 /(AUDATA3*2)	PD19/IRQ3/$\overline{\text{POE7}}$
91	PD20	PD20/D20/IRQ4/TIC5WS	PD20	PD20/IRQ4/TIC5WS

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
90	PD21	PD21/D21/IRQ5/TIC5VS	PD21	PD21/IRQ5/TIC5VS
89	PD22 /(AUDCK*2)	PD22/D22/IRQ6/TIC5US	PD22 /(AUDCK*2)	PD22/IRQ6/TIC5US
88	PD23 /(AUDSYNC*2)	PD23/D23/IRQ7	PD23 /(AUDSYNC*2)	PD23/IRQ7
87	PD24	PD24/D24/DREQ0/TIOC4DS	PD24	PD24/DREQ0/TIOC4DS
85	PD25	PD25/D25/DREQ1/TIOC4CS	PD25	PD25/DREQ1/TIOC4CS
84	PD26	PD26/D26/DACK0/TIOC4BS	PD26	PD26/TIOC4BS
83	PD27	PD27/D27/DACK1/TIOC4AS	PD27	PD27/TIOC4AS
82	PD28	PD28/D28/CS2/TIOC3DS	PD28	PD28/TIOC3DS
81	PD29	PD29/D29/CS3/TIOC3BS	PD29	PD29/TIOC3BS
70	PD30	PD30/D30/TIOC3CS/IRQOUT	PD30	PD30/TIOC3CS/IRQOUT
69	PD31	PD31/D31/TIOC3AS/ADTRG	PD31	PD31/TIOC3AS/ADTRG
133	PE0/(AUDCK*2)	PE0/DREQ0/TIOC0A	PE0/(AUDCK*2)	PE0/DREQ0/TIOC0A
134	PE1	PE1/TEND0/TIOC0B	PE1	PE1/TIOC0B
135	PE2	PE2/DREQ1/TIOC0C	PE2	PE2/DREQ1/TIOC0C
137	PE3/(AUDATA3*2)	PE3/TEND1/TIOC0D	PE3/(AUDATA3*2)	PE3/TIOC0D
138	PE4/(AUDATA2*2)	PE4/IOIS16/TIOC1A/RXD3	PE4/(AUDATA2*2)	PE4/TIOC1A/RXD3
139	PE5/(AUDATA1*2)	PE5/CS6/CE1B/TIOC1B/TXD3	PE5/(AUDATA1*2)	PE5/TIOC1B/TXD3
140	PE6/(AUDATA0*2)	PE6/CS7/TIOC2A/SCK3	PE6/(AUDATA0*2)	PE6/TIOC2A/SCK3
171	PE7	PE7/BS/TIOC2B/UBCTRG /RXD2/SSI	PE7	PE7/TIOC2B/UBCTRG /RXD2/SSI
172	PE8/(TMS*1)	PE8/TIOC3A/SCK2/SSCK	PE8/(TMS*1)	PE8/TIOC3A/SCK2/SSCK
174	PE9/(TRST*1)	PE9/TIOC3B/SCK3/RTS3	PE9/(TRST*1)	PE9/TIOC3B/SCK3/RTS3
175	PE10/(TDI*1)	PE10/TIOC3C/TXD2/SSO	PE10/(TDI*1)	PE10/TIOC3C/TXD2/SSO
176	PE11/(TDO*1)	PE11/TIOC3D/RXD3/CTS3	PE11/(TDO*1)	PE11/TIOC3D/RXD3/CTS3
1	PE12/(TCK*1)	PE12/TIOC4A/TXD3/SCS	PE12/(TCK*1)	PE12/TIOC4A/TXD3/SCS
2	PE13/(ASEBRKAK /ASEBRK*1)	PE13/TIOC4B/MRES	PE13/(ASEBRKAK /ASEBRK*1)	PE13/TIOC4B/MRES

	Pin Name			
	On-Chip ROM Enabled (MCU Mode 2)		Single-Chip Mode (MCU Mode 3)	
Pin No.	Initial Function	PFC Selected Function Possibilities	Initial Function	PFC Selected Function Possibilities
4	PE14	PE14/$\overline{\text{WRHH}}$/ICIOWR/$\overline{\text{AH}}$ /DQMUU/DACK0/TIOC4C	PE14	PE14/TIOC4C
7	PE15	PE15/CKE/DACK1/TIOC4D /$\overline{\text{IRQOUT}}$	PE15	PE15/TIOC4D/$\overline{\text{IRQOUT}}$
9	PE16	PE16/$\overline{\text{CS8}}$/TIOC3BS	PE16	PE16/TIOC3BS
10	PE17	PE17/TIOC3DS	PE17	PE17/TIOC3DS
12	PE18	PE18/TIOC4AS	PE18	PE18/TIOC4AS
13	PE19	PE19/TIOC4BS	PE19	PE19/TIOC4BS
14	PE20	PE20/TIOC4CS	PE20	PE20/TIOC4CS
15	PE21	PE21/TIOC4DS	PE21	PE21/TIOC4DS
143	PF0/AN0	PF0/AN0	PF0/AN0	PF0/AN0
144	PF1/AN1	PF1/AN1	PF1/AN1	PF1/AN1
147	PF2/AN2	PF2/AN2	PF2/AN2	PF2/AN2
148	PF3/AN3	PF3/AN3	PF3/AN3	PF3/AN3
152	PF4/AN4	PF4/AN4	PF4/AN4	PF4/AN4
153	PF5/AN5	PF5/AN5	PF5/AN5	PF5/AN5
157	PF6/AN6	PF6/AN6	PF6/AN6	PF6/AN6
158	PF7/AN7	PF7/AN7	PF7/AN7	PF7/AN7
145	PF8/AN8	PF8/AN8	PF8/AN8	PF8/AN8
146	PF9/AN9	PF9/AN9	PF9/AN9	PF9/AN9
149	PF10/AN10	PF10/AN10	PF10/AN10	PF10/AN10
150	PF11/AN11	PF11/AN11	PF11/AN11	PF11/AN11
154	PF12/AN12	PF12/AN12	PF12/AN12	PF12/AN12
155	PF13/AN13	PF13/AN13	PF13/AN13	PF13/AN13
159	PF14/AN14	PF14/AN14	PF14/AN14	PF14/AN14
160	PF15/AN15	PF15/AN15	PF15/AN15	PF15/AN15

Notes: 1. Fixed to TMS, $\overline{\text{TRST}}$, TDI, TDO, TCK, and $\overline{\text{ASEBRKAK}}$/$\overline{\text{ASEBRK}}$ when using the E10A (in $\overline{\text{ASEMD0}}$ = low).

2. Only in F-ZTAT version supporting full functions of E10A. Fixed as AUD pins when using the AUD function of the E10A.

21.1 Register Descriptions

The PFC has the following registers. For details on register addresses and register states in each processing state, refer to section 27, List of Registers.

Table 21.21 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Port A I/O register H	PAIORH	R/W	H'0000	H'FFFFD104	8, 16, 32
Port A I/O register L	PAIORL	R/W	H'0000	H'FFFFD106	8, 16
Port A control register H4	PACRH4	R/W	H'0000	H'FFFFD108	8, 16, 32
Port A control register H3	PACRH3	R/W	H'0000	H'FFFFD10A	8, 16
Port A control register H2	PACRH2	R/W	H'0000*	H'FFFFD10C	8, 16, 32
Port A control register H1	PACRH1	R/W	H'0000	H'FFFFD10E	8, 16
Port A control register L4	PACRL4	R/W	H'0000*	H’FFFFD110	8, 16, 32
Port A control register L3	PACRL3	R/W	H'0000*	H’FFFFD112	8, 16
Port A control register L2	PACRL2	R/W	H'0000	H’FFFFD114	8, 16, 32
Port A control register L1	PACRL1	R/W	H'0000	H’FFFFD116	8, 16
Port B I/O register L	PBIORL	R/W	H'0000	H'FFFFD186	8, 16
Port B control register L3	PBCRL3	R/W	H'0000	H'FFFFD192	8, 16
Port B control register L2	PBCRL2	R/W	H'0000	H'FFFFD194	8, 16, 32
Port B control register L1	PBCRL1	R/W	H'0000*	H'FFFFD196	8, 16
Port C I/O register H	PCIORH	R/W	H'0000	H'FFFFD204	8, 16, 32
Port C I/O register L	PCIORL	R/W	H'0000	H'FFFFD206	8, 16
Port C control register H3	PCCRH3	R/W	H'0000*	H'FFFFD20A	8, 16
Port C control register H2	PCCRH2	R/W	H'0000*	H'FFFFD20C	8, 16, 32
Port C control register H1	PCCRH1	R/W	H'0000*	H'FFFFD20E	8, 16
Port C control register L4	PCCRL4	R/W	H'0000*	H'FFFFD210	8, 16, 32
Port C control register L3	PCCRL3	R/W	H'0000*	H'FFFFD212	8, 16
Port C control register L2	PCCRL2	R/W	H'0000*	H'FFFFD214	8, 16, 32
Port C control register L1	PCCRL1	R/W	H'0000*	H'FFFFD216	8, 16
Port D I/O register H	PDIORH	R/W	H'0000	H'FFFFD284	8, 16, 32
Port D I/O register L	PDIORL	R/W	H'0000	H'FFFFD286	8, 16

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Port D control register H4	PDCRH4	R/W	H'0000*	H'FFFFD288	8, 16, 32
Port D control register H3	PDCRH3	R/W	H'0000*	H'FFFFD28A	8, 16
Port D control register H2	PDCRH2	R/W	H'0000*	H'FFFFD28C	8, 16, 32
Port D control register H1	PDCRH1	R/W	H'0000*	H'FFFFD28E	8, 16
Port D control register L4	PDCRL4	R/W	H'0000*	H'FFFFD290	8, 16, 32
Port D control register L3	PDCRL3	R/W	H'0000*	H'FFFFD292	8, 16
Port D control register L2	PDCRL2	R/W	H'0000*	H'FFFFD294	8, 16, 32
Port D control register L1	PDCRL1	R/W	H'0000*	H'FFFFD296	8, 16
Port E I/O register H	PEIORH	R/W	H'0000	H'FFFFD304	8, 16, 32
Port E I/O register L	PEIORL	R/W	H'0000	H'FFFFD306	8, 16
Port E control register H2	PECRH2	R/W	H'0000	H'FFFFD30C	8, 16, 32
Port E control register H1	PECRH1	R/W	H'0000	H'FFFFD30E	8, 16
Port E control register L4	PECRL4	R/W	H'0000	H'FFFFD310	8, 16, 32
Port E control register L3	PECRL3	R/W	H'0000	H'FFFFD312	8, 16
Port E control register L2	PECRL2	R/W	H'0000	H'FFFFD314	8, 16, 32
Port E control register L1	PECRL1	R/W	H'0000	H'FFFFD316	8, 16
High-current port control register	HCPCR	R/W	H'000F	H'FFFFD320	8, 16, 32
IRQOUT function control register	IFCR	R/W	H'0000	H'FFFFD322	8, 16

Note: * The initial values of registers differ depending on the operating mode and the product type in use. For details, refer to register descriptions in this section.

21.1.1 Port A I/O Register L, H (PAIORL, PAIORH)

PAIORL and PAIORH are 16-bit readable/writable registers that are used to set the pins on port A as inputs or outputs. Bits PA29IOR to PA0IOR correspond to pins PA29 to PA0 (names of multiplexed pins are here given as port names and pin numbers alone). PAIORL is enabled when the port A pins are functioning as general-purpose inputs/outputs (PA15 to PA0). In other states, PAIORL is disabled. PAIORH is enabled when the port A pins are functioning as general-purpose input/output (PA29 to PA16). In other states, PAIORH is disabled.

A given pin on port A will be an output pin if the corresponding bit in PAIORH or PAIORL is set to 1, and an input pin if the bit is cleared to 0.

However, bits 13 to 0 of PAIORH, and bits 11, 6, and 2 to 0 of PAIORL are disabled in SH7083. Bits 13 to 2 of PAIORH are disabled in SH7084. Bits 13 to 10 of PAIORH are disabled in SH7085.

Bits 15 and 14 of PAIORH are reserved. These bits are always read as 0. The write value should always be 0.

The initial values of PAIORL and PAIORH are H'0000, respectively.

- Port A I/O Register H (PAIORH)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PA29 IOR	PA28 IOR	PA27 IOR	PA26 IOR	PA25 IOR	PA24 IOR	PA23 IOR	PA22 IOR	PA21 IOR	PA20 IOR	PA19 IOR	PA18 IOR	PA17 IOR	PA16 IOR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Port A I/O Register L (PAIORL)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PA15 IOR	PA14 IOR	PA13 IOR	PA12 IOR	PA11 IOR	PA10 IOR	PA9 IOR	PA8 IOR	PA7 IOR	PA6 IOR	PA5 IOR	PA4 IOR	PA3 IOR	PA2 IOR	PA1 IOR	PA0 IOR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

21.1.2 Port A Control Registers L1 to L4, H1 to H4 (PACRL1 to PACRL4, PACRH1 to PACRH4)

PACRL1 to PACRL4 and PACRH1 to PACRH4 are 16-bit readable/writable registers that are used to select the functions of the multiplexed pins on port A.

SH7083:

- Port A Control Registers H4 to H1 (PACRH4 to PACRH1)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- Port A Control Register L4 (PACRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA15 MD2	PA15 MD1	PA15 MD0	-	PA14 MD2	PA14 MD1	PA14 MD0	-	PA13 MD2	PA13 MD1	PA13 MD0	-	PA12 MD2	PA12 MD1	PA12 MD0
Initial value:	0	0	0	0*1	0	0	0	0*2	0	0	0	0*2	0	0	0	0*2
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Notes: 1. The initial value is 1 in the on-chip ROM enabled/disabled external-extension mode.
2. The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PA15MD2	0	R/W	PA15 Mode
13	PA15MD1	0	R/W	Select the function of the PA15/CK pin.
12	PA15MD0	0*1	R/W	000: PA15 I/O (port) 001: CK output (CPG)*3 Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA14MD2	0	R/W	PA14 Mode
9	PA14MD1	0	R/W	Select the function of the PA14/$\overline{RD}$ pin.
8	PA14MD0	0*2	R/W	000: PA14 I/O (port) 001: $\overline{RD}$ output (BSC)*3 Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PA13MD2	0	R/W	PA13 Mode
5	PA13MD1	0	R/W	Select the function of the PA13/$\overline{WRH}$/DQMLU/$\overline{POE7}$ pin.
4	PA13MD0	0*2	R/W	000: PA13 I/O (port) 001: $\overline{WRH}$/DQMLU output (BSC)*3 011: $\overline{POE7}$ input (POE) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PA12MD2	0	R/W	PA12 Mode
1	PA12MD1	0	R/W	Select the function of the PA12/$\overline{WRL}$/DQMLL/$\overline{POE6}$ pin.
0	PA12MD0	0*2	R/W	000: PA12 I/O (port) 001: $\overline{WRL}$/DQMLL output (BSC)*3 011: $\overline{POE6}$ input (POE) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM enabled/disabled external-extension mode.
2. The initial value is 1 in the on-chip ROM disabled external-extension mode.
3. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L3 (PACRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	PA10 MD2	PA10 MD1	PA10 MD0	-	PA9 MD2	PA9 MD1	PA9 MD0	-	PA8 MD2	PA8 MD1	PA8 MD0
Initial value:	0	0	0	0	0	0	0	0*	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	PA10MD2	0	R/W	PA10 Mode
9	PA10MD1	0	R/W	Select the function of the PA10/$\overline{\text{CS0}}$/$\overline{\text{POE4}}$ pin.
8	PA10MD0	0*[1]	R/W	000: PA10 I/O (port) 001: $\overline{\text{CS0}}$ output (BSC)*[2] 011: $\overline{\text{POE4}}$ input (POE) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PA9MD2	0	R/W	PA9 Mode
5	PA9MD1	0	R/W	Select the function of the PA9/CKE/IRQ3/TCLKD pin.
4	PA9MD0	0	R/W	000: PA9 I/O (port) 001: TCLKD input (MTU2) 010: IRQ3 input (INTC) 101: CKE output (BSC)*[2] Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PA8MD2	0	R/W	PA8 Mode
1	PA8MD1	0	R/W	Select the function of the PA8/RDWR/IRQ2/TCLKC pin.
0	PA8MD0	0	R/W	000: PA8 I/O (port)
				001: TCLKC input (MTU2)
				010: IRQ2 input (INTC)
				101: RDWR output (BSC)*2
				Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L2 (PACRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA7 MD2	PA7 MD1	PA7 MD0	-	-	-	-	-	PA5 MD2	PA5 MD1	PA5 MD0	-	PA4 MD2	PA4 MD1	PA4 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
14	PA7MD2	0	R/W	PA7 Mode
13	PA7MD1	0	R/W	Select the function of the PA7/$\overline{CS3}$/TCLKB pin.
12	PA7MD0	0	R/W	000: PA7 I/O (port)
				001: TCLKB input (MTU2)
				010: $\overline{CS3}$ output (BSC)*
				Other than above: Setting prohibited
11 to 7	—	All 0	R	Reserved
				These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PA5MD2	0	R/W	PA5 Mode
5	PA5MD1	0	R/W	Select the function of the PA5/A22/DREQ1/IRQ1/SCK1 pin.
4	PA5MD0	0	R/W	000: PA5 I/O (port) 001: SCK1 I/O (SCI) 010: DREQ1 input (DMAC) 011: IRQ1 input (INTC) 101: A22 output (BSC)* Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PA4MD2	0	R/W	PA4 Mode
1	PA4MD1	0	R/W	Select the function of the PA4/A23/TXD1 pin.
0	PA4MD0	0	R/W	000: PA4 I/O (port) 001: TXD1 output (SCI) 101: A23 output (BSC)* Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L1 (PACRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA3 MD2	PA3 MD1	PA3 MD0	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PA3MD2	0	R/W	PA3 Mode
13	PA3MD1	0	R/W	Select the function of the PA3/A24/RXD1 pin.
12	PA3MD0	0	R/W	000: PA3 I/O (port) 001: RXD1 output (SCI) 101: A24 output (BSC)* Other than above: Setting prohibited
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

SH7084:

- Port A Control Registers H4 to H2 (PACRH4 to PACRH2)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- Port A Control Register H1 (PACRH1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	PA17 MD1	PA17 MD0	-	PA16 MD2	PA16 MD1	PA16 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PA17MD1	0	R/W	PA17 Mode
4	PA17MD0	0	R/W	Select the function of the PA17/$\overline{\text{WAIT}}$ pin. 00: PA17 I/O (port) 01: $\overline{\text{WAIT}}$ input (BSC)* Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PA16MD2	0	R/W	PA16 Mode
1	PA16MD1	0	R/W	Select the function of the PA16/$\overline{\text{AH}}$/CKE pin.
0	PA16MD0	0	R/W	000: PA16 I/O (port) 001: $\overline{\text{AH}}$ output (BSC)* 011: CKE output (BSC)* Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L4 (PACRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA15 MD2	PA15 MD1	PA15 MD0	-	PA14 MD2	PA14 MD1	PA14 MD0	-	PA13 MD2	PA13 MD1	PA13 MD0	-	PA12 MD2	PA12 MD1	PA12 MD0
Initial value:	0	0	0	0*1	0	0	0	0*2	0	0	0	0*2	0	0	0	0*2
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Notes: 1. The initial value is 1 in the on-chip ROM enabled/disabled external-extension mode.
2. The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PA15MD2	0	R/W	PA15 Mode
13	PA15MD1	0	R/W	Select the function of the PA15/CK pin.
12	PA15MD0	0*1	R/W	000: PA15 I/O (port) 001: CK output (CPG)*3 Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA14MD2	0	R/W	PA14 Mode
9	PA14MD1	0	R/W	Select the function of the PA14/$\overline{RD}$ pin.
8	PA14MD0	0*2	R/W	000: PA14 I/O (port) 001: $\overline{RD}$ output (BSC)*3 Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PA13MD2	0	R/W	PA13 Mode
5	PA13MD1	0	R/W	Select the function of the PA13/$\overline{WRH}$/DQMLU/$\overline{POE7}$ pin.
4	PA13MD0	0*2	R/W	000: PA13 I/O (port) 001: $\overline{WRH}$/DQMLU output (BSC)*3 011: $\overline{POE7}$ input (POE) Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PA12MD2	0	R/W	PA12 Mode
1	PA12MD1	0	R/W	Select the function of the PA12/$\overline{\text{WRL}}$/DQMLL/$\overline{\text{POE6}}$ pin.
0	PA12MD0	0*[2]	R/W	000: PA12 I/O (port) 001: $\overline{\text{WRL}}$/DQMLL output (BSC)*[3] 011: $\overline{\text{POE6}}$ input (POE) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM enabled/disabled external-extension mode.
2. The initial value is 1 in the on-chip ROM disabled external-extension mode.
3. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L3 (PACRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA11 MD2	PA11 MD1	PA11 MD0	-	PA10 MD2	PA10 MD1	PA10 MD0	-	PA9 MD2	PA9 MD1	PA9 MD0	-	PA8 MD2	PA8 MD1	PA8 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PA11MD2	0	R/W	PA11 Mode
13	PA11MD1	0	R/W	Select the function of the PA11/$\overline{\text{CS1}}$/$\overline{\text{POE5}}$ pin.
12	PA11MD0	0*[1]	R/W	000: PA11 I/O (port) 001: $\overline{\text{CS1}}$ output (BSC)*[3] 011: $\overline{\text{POE5}}$ input (POE)*[2] Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA10MD2	0	R/W	PA10 Mode
9	PA10MD1	0	R/W	Select the function of the PA10/$\overline{CS0}$/$\overline{POE4}$ pin.
8	PA10MD0	0*[1]	R/W	000: PA10 I/O (port) 001: $\overline{CS0}$ output (BSC)*[3] 011: $\overline{POE4}$ input (POE) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PA9MD2	0	R/W	PA9 Mode
5	PA9MD1	0	R/W	Select the function of the PA9/CKE/IRQ3/TCLKD pin.
4	PA9MD0	0	R/W	000: PA9 I/O (port) 001: TCLKD input (MTU2) 010: IRQ3 input (INTC) 101: CKE output (BSC)*[3] Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PA8MD2	0	R/W	PA8 Mode
1	PA8MD1	0	R/W	Select the function of the PA8/RDWR/IRQ2/TCLKC pin.
0	PA8MD0	0	R/W	000: PA8 I/O (port) 001: TCLKC input (MTU2) 010: IRQ2 input (INTC) 101: RDWR output (BSC)*[3] Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. When the $\overline{POE5}$ input is selected, this setting cannot be changed.
3. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L2 (PACRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA7 MD2	PA7 MD1	PA7 MD0	-	PA6 MD2	PA6 MD1	PA6 MD0	-	PA5 MD2	PA5 MD1	PA5 MD0	-	PA4 MD2	PA4 MD1	PA4 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PA7MD2	0	R/W	PA7 Mode
13	PA7MD1	0	R/W	Select the function of the PA7/$\overline{CS3}$/TCLKB pin.
12	PA7MD0	0	R/W	000: PA7 I/O (port) 001: TCLKB input (MTU2) 010: $\overline{CS3}$ output (BSC)* Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA6MD2	0	R/W	PA6 Mode
9	PA6MD1	0	R/W	Select the function of the PA6/$\overline{CS2}$/TCLKA pin.
8	PA6MD0	0	R/W	000: PA6 I/O (port) 001: TCLKA input (MTU2) 010: $\overline{CS2}$ output (BSC)* Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PA5MD2	0	R/W	PA5 Mode
5	PA5MD1	0	R/W	Select the function of the
4	PA5MD0	0	R/W	PA5/A22/DREQ1/IRQ1/SCK1 pin.
				000: PA5 I/O (port)
				001: SCK1 I/O (SCI)
				010: DREQ1 input (DMAC)
				011: IRQ1 input (INTC)
				101: A22 output (BSC)*
				Other than above: Setting prohibited
3	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
2	PA4MD2	0	R/W	PA4 Mode
1	PA4MD1	0	R/W	Select the function of the PA4/A23/TXD1 pin.
0	PA4MD0	0	R/W	000: PA4 I/O (port)
				001: TXD1 output (SCI)
				101: A23 output (BSC)*
				Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L1 (PACRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA3 MD2	PA3 MD1	PA3 MD0	-	PA2 MD2	PA2 MD1	PA2 MD0	-	PA1 MD2	PA1 MD1	PA1 MD0	-	PA0 MD2	PA0 MD1	PA0 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PA3MD2	0	R/W	PA3 Mode
13	PA3MD1	0	R/W	Select the function of the PA3/A24/RXD1 pin.
12	PA3MD0	0	R/W	000: PA3 I/O (port) 001: RXD1 input (SCI) 101: A24 output (BSC)* Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA2MD2	0	R/W	PA2 Mode
9	PA2MD1	0	R/W	Select the function of the PA2/A25/DREQ0/IRQ0/SCK0 pin.
8	PA2MD0	0	R/W	000: PA2 I/O (port) 001: SCK0 I/O (SCI) 010: DREQ0 input (DMAC) 011: IRQ0 input (INTC) 101: A25 output (BSC)* Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PA1MD2	0	R/W	PA1 Mode
5	PA1MD1	0	R/W	Select the function of the PA1/$\overline{CS5}$/TXD0 pin.
4	PA1MD0	0	R/W	000: PA1 I/O (port) 001: TXD0 output (SCI) 101: $\overline{CS5}$ output (BSC)* Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PA0MD2	0	R/W	PA0 Mode
1	PA0MD1	0	R/W	Select the function of the PA0/$\overline{CS4}$/RXD0 pin.
0	PA0MD0	0	R/W	000: PA0 I/O (port)
				001: RXD0 input (SCI)
				101: $\overline{CS4}$ output (BSC)*
				Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

SH7085:

- Port A Control Register H4 (PACRH4)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved
				These bits are always read as 0. The write value should always be 0.

- Port A Control Register H3 (PACRH3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	PA25 MD1	PA25 MD0	-	-	PA24 MD1	PA24 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 6	—	All 0	R	Reserved
				These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5	PA25MD1	0	R/W	PA25 Mode
4	PA25MD0	0	R/W	Select the function of the PA25/$\overline{\text{CE2B}}$/DACK3/$\overline{\text{POE8}}$ pin. 00: PA25 I/O (port) 01: $\overline{\text{CE2B}}$ output (BSC)* 10: DACK3 output (DMAC)* 11: $\overline{\text{POE8}}$ input (POE)
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PA24MD1	0	R/W	PA24 Mode
0	PA24MD0	0	R/W	Select the function of the PA24/$\overline{\text{CE2A}}$/DREQ3 pin. 00: PA24 I/O (port) 01: $\overline{\text{CE2A}}$ output (BSC)* 10: DREQ3 input (DMAC) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register H2 (PACRH2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PA23 MD1	PA23 MD0	-	-	PA22 MD1	PA22 MD0	-	-	PA21 MD1	PA21 MD0	-	-	PA20 MD1	PA20 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
13	PA23MD1	0	R/W	PA23 Mode
12	PA23MD0	0*[1]	R/W	Select the function of the PA23/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/DQMUU/TIC5W pin. 00: PA23 I/O (port) 01: $\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/DQMUU output (BSC)*[2] 11: TIC5W input (MTU2) Other than above: Setting prohibited
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PA22MD1	0	R/W	PA22 Mode
8	PA22MD0	0*[1]	R/W	Select the function of the PA22/$\overline{\text{WRHL}}$/$\overline{\text{ICIORD}}$/DQMUL/TIC5V pin. 00: PA22 I/O (port) 01: $\overline{\text{WRHL}}$/$\overline{\text{ICIORD}}$/DQMUL output (BSC)*[2] 11: TIC5V input (MTU2) Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PA21MD1	0	R/W	PA21 Mode
4	PA21MD0	0	R/W	Select the function of the PA21/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/$\overline{\text{CASU}}$/TIC5U pin. 00: PA21 I/O (port) 01: $\overline{\text{CS5}}$/$\overline{\text{CE1A}}$ output (BSC)*[2] 10: $\overline{\text{CASU}}$ output (BSC)*[2] 11: TIC5U input (MTU2)
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	PA20MD1	0	R/W	PA20 Mode
0	PA20MD0	0	R/W	Select the function of the PA20/$\overline{CS4}$/$\overline{RASU}$ pin. 00: PA20 I/O (port) 01: $\overline{CS4}$ output (BSC)*2 10: $\overline{RASU}$ output (BSC)*2 Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register H1 (PACRH1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PA19 MD1	PA19 MD0	-	-	PA18 MD1	PA18 MD0	-	-	PA17 MD1	PA17 MD0	-	PA16 MD2	PA16 MD1	PA16 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PA19MD1	0	R/W	PA19 Mode
12	PA19MD0	0	R/W	Select the function of the PA19/$\overline{BACK}$/TEND1 pin. 00: PA19 I/O (port) 01: $\overline{BACK}$ output (BSC)* 10: TEND1 output (DMAC)* Other than above: Setting prohibited
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9	PA18MD1	0	R/W	PA18 Mode
8	PA18MD0	0	R/W	Select the function of the PA18/$\overline{\text{BREQ}}$/TEND0 pin. 00: PA18 I/O (port) 01: $\overline{\text{BREQ}}$ input (BSC)* 10: TEND0 output (DMAC)* Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PA17MD1	0	R/W	PA17 Mode
4	PA17MD0	0	R/W	Select the function of the PA17/$\overline{\text{WAIT}}$/DACK2 pin. 00: PA17 I/O (port) 01: $\overline{\text{WAIT}}$ input (BSC)* 10: DACK2 output (DMAC)* Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PA16MD2	0	R/W	PA16 Mode
1	PA16MD1	0	R/W	Select the function of the
0	PA16MD0	0	R/W	PA16/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/DQMUU/CKE/DREQ2 /$\overline{\text{AUDSYNC}}$ pin. Fixed to $\overline{\text{AUDSYNC}}$ output when using the AUD function of the E10A. 000: PA16 I/O (port) 001: $\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/DQMUU output (BSC)* 010: DREQ2 input (DMAC) 101: CKE output (BSC)* Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L4 (PACRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA15 MD2	PA15 MD1	PA15 MD0	-	PA14 MD2	PA14 MD1	PA14 MD0	-	PA13 MD2	PA13 MD1	PA13 MD0	-	PA12 MD2	PA12 MD1	PA12 MD0
Initial value:	0	0	0	0*1	0	0	0	0*2	0	0	0	0*2	0	0	0	0*2
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Notes: 1. The initial value is 1 in the on-chip ROM enabled/disabled external-extension mode.
2. The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PA15MD2	0	R/W	PA15 Mode
13	PA15MD1	0	R/W	Select the function of the PA15/CK pin.
12	PA15MD0	0*1	R/W	000: PA15 I/O (port) 001: CK output (CPG)*3 Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA14MD2	0	R/W	PA14 Mode
9	PA14MD1	0	R/W	Select the function of the PA14/$\overline{RD}$ pin.
8	PA14MD0	0*2	R/W	000: PA14 I/O (port) 001: $\overline{RD}$ output (BSC)*3 Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PA13MD2	0	R/W	PA13 Mode
5	PA13MD1	0	R/W	Select the function of the
4	PA13MD0	0*[2]	R/W	PA13/$\overline{\text{WRH}}$/$\overline{\text{WE}}$/DQMLU/$\overline{\text{POE7}}$ pin. 000: PA13 I/O (port) 001: $\overline{\text{WRH}}$/$\overline{\text{WE}}$/DQMLU output (BSC)*[3] 011: $\overline{\text{POE7}}$ input (POE) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PA12MD2	0	R/W	PA12 Mode
1	PA12MD1	0	R/W	Select the function of the PA12/$\overline{\text{WRL}}$/DQMLL/$\overline{\text{POE6}}$
0	PA12MD0	0*[2]	R/W	pin. 000: PA12 I/O (port) 001: $\overline{\text{WRL}}$/DQMLL output (BSC)*[3] 011: $\overline{\text{POE6}}$ input (POE) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM enabled/disabled external-extension mode.
2. The initial value is 1 in the on-chip ROM disabled external-extension mode.
3. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L3 (PACRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA11 MD2	PA11 MD1	PA11 MD0	-	PA10 MD2	PA10 MD1	PA10 MD0	-	PA9 MD2	PA9 MD1	PA9 MD0	-	PA8 MD2	PA8 MD1	PA8 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PA11MD2	0	R/W	PA11 Mode
13	PA11MD1	0	R/W	Select the function of the PA11/$\overline{\text{CS1}}$/$\overline{\text{POE5}}$ pin.
12	PA11MD0	0*[1]	R/W	000: PA11 I/O (port) 001: $\overline{\text{CS1}}$ output (BSC)*[3] 011: $\overline{\text{POE5}}$ input (POE)*[2] Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA10MD2	0	R/W	PA10 Mode
9	PA10MD1	0	R/W	Select the function of the PA10/$\overline{\text{CS0}}$/$\overline{\text{POE4}}$ pin.
8	PA10MD0	0*[1]	R/W	000: PA10 I/O (port) 001: $\overline{\text{CS0}}$ output (BSC)*[3] 011: $\overline{\text{POE4}}$ input (POE) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PA9MD2	0	R/W	PA9 Mode
5	PA9MD1	0	R/W	Select the function of the PA9/$\overline{\text{FRAME}}$/CKE/IRQ3/TCLKD pin.
4	PA9MD0	0	R/W	000: PA9 I/O (port) 001: TCLKD input (MTU2) 010: IRQ3 input (INTC) 011: $\overline{\text{FRAME}}$ output (BSC)*[3] 101: CKE output (BSC)*[3] Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PA8MD2	0	R/W	PA8 Mode
1	PA8MD1	0	R/W	Select the function of the PA8/RDWR/IRQ2/TCLKC pin.
0	PA8MD0	0	R/W	000: PA8 I/O (port) 001: TCLKC input (MTU2) 010: IRQ2 input (INTC) 101: RDWR output (BSC)*3 Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. When the $\overline{\text{POE5}}$ input is selected, this setting cannot be changed.
3. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L2 (PACRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA7 MD2	PA7 MD1	PA7 MD0	-	PA6 MD2	PA6 MD1	PA6 MD0	-	PA5 MD2	PA5 MD1	PA5 MD0	-	PA4 MD2	PA4 MD1	PA4 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PA7MD2	0	R/W	PA7 Mode
13	PA7MD1	0	R/W	Select the function of the PA7/$\overline{\text{CS3}}$/TCLKB pin.
12	PA7MD0	0	R/W	000: PA7 I/O (port) 001: TCLKB input (MTU2) 010: $\overline{\text{CS3}}$ output (BSC)* Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	PA6MD2	0	R/W	PA6 Mode
9	PA6MD1	0	R/W	Select the function of the PA6/$\overline{CS2}$/TCLKA pin.
8	PA6MD0	0	R/W	000: PA6 I/O (port)
				001: TCLKA input (MTU2)
				010: $\overline{CS2}$ output (BSC)*
				Other than above: Setting prohibited
7	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
6	PA5MD2	0	R/W	PA5 Mode
5	PA5MD1	0	R/W	Select the function of the PA5/A22/DREQ1/IRQ1/SCK1 pin.
4	PA5MD0	0	R/W	000: PA5 I/O (port)
				001: SCK1 I/O (SCI)
				010: DREQ1 input (DMAC)
				011: IRQ1 input (INTC)
				101: A22 output (BSC)*
				Other than above: Setting prohibited
3	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
2	PA4MD2	0	R/W	PA4 Mode
1	PA4MD1	0	R/W	Select the function of the PA4/A23/TXD1 pin.
0	PA4MD0	0	R/W	000: PA4 I/O (port)
				001: TXD1 output (SCI)
				101: A23 output (BSC)*
				Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L1 (PACRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA3 MD2	PA3 MD1	PA3 MD0	-	PA2 MD2	PA2 MD1	PA2 MD0	-	PA1 MD2	PA1 MD1	PA1 MD0	-	PA0 MD2	PA0 MD1	PA0 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PA3MD2	0	R/W	PA3 Mode
13	PA3MD1	0	R/W	Select the function of the PA3/A24/RXD1 pin.
12	PA3MD0	0	R/W	000: PA3 I/O (port) 001: RXD1 input (SCI) 101: A24 output (BSC)* Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA2MD2	0	R/W	PA2 Mode
9	PA2MD1	0	R/W	Select the function of the
8	PA2MD0	0	R/W	PA2/A25/DREQ0/IRQ0/SCK0 pin. 000: PA2 I/O (port) 001: SCK0 I/O (SCI) 010: DREQ0 input (DMAC) 011: IRQ0 input (INTC) 101: A25 output (BSC)* Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PA1MD2	0	R/W	PA1 Mode
5	PA1MD1	0	R/W	Select the function of the PA1/$\overline{CS5}$/$\overline{CE1A}$/TXD0 pin.
4	PA1MD0	0	R/W	000: PA1 I/O (port)
				001: TXD0 output (SCI)
				101: $\overline{CS5}$/$\overline{CE1A}$ output (BSC)*
				Other than above: Setting prohibited
3	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
2	PA0MD2	0	R/W	PA0 Mode
1	PA0MD1	0	R/W	Select the function of the PA0/$\overline{CS4}$/RXD0 pin.
0	PA0MD0	0	R/W	000: PA0 I/O (port)
				001: RXD0 input (SCI)
				101: $\overline{CS4}$ output (BSC)*
				Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

SH7086:

- Port A Control Register H4 (PACRH4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	PA29 MD1	PA29 MD0	-	-	PA28 MD1	PA28 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 6	—	All 0	R	Reserved
				These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5	PA29MD1	0	R/W	PA29 Mode
4	PA29MD0	0	R/W	Select the function of the PA29/A29/IRQ3 pin. 00: PA29 I/O (port) 01: A29 output (BSC)* 11: IRQ3 input (INTC) Other than above: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PA28MD1	0	R/W	PA28 Mode
0	PA28MD0	0	R/W	Select the function of the PA28/A28/IRQ2 pin. 00: PA28 I/O (port) 01: A28 output (BSC)* 11: IRQ2 input (INTC) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register H3 (PACRH3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PA27 MD1	PA27 MD0	-	-	PA26 MD1	PA26 MD0	-	-	PA25 MD1	PA25 MD0	-	-	PA24 MD1	PA24 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
13	PA27MD1	0	R/W	PA27 Mode
12	PA27MD0	0	R/W	Select the function of the PA27/A27/IRQ1 pin. 00: PA27 I/O (port) 01: A27 output (BSC)* 11: IRQ1 input (INTC) Other than above: Setting prohibited
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PA26MD1	0	R/W	PA26 Mode
8	PA26MD0	0	R/W	Select the function of the PA26/A26/IRQ0 pin. 00: PA26 I/O (port) 01: A26 output (BSC)* 11: IRQ0 input (INTC) Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PA25MD1	0	R/W	PA25 Mode
4	PA25MD0	0	R/W	Select the function of the PA25/$\overline{\text{CE2B}}$/DACK3/$\overline{\text{POE8}}$ pin. 00: PA25 I/O (port) 01: $\overline{\text{CE2B}}$ output (BSC)* 10: DACK3 output (DMAC)* 11: $\overline{\text{POE8}}$ input (POE)
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	PA24MD1	0	R/W	PA24 Mode
0	PA24MD0	0	R/W	Select the function of the PA24/$\overline{\text{CE2A}}$/DREQ3 pin. 00: PA24 I/O (port) 01: $\overline{\text{CE2A}}$ output (BSC)* 10: DREQ3 input (DMAC) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register H2 (PACRH2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PA23 MD1	PA23 MD0	-	-	PA22 MD1	PA22 MD0	-	-	PA21 MD1	PA21 MD0	-	-	PA20 MD1	PA20 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PA23MD1	0	R/W	PA23 Mode
12	PA23MD0	0*[1]	R/W	Select the function of the PA23/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/DQMUU/TIC5W pin. 00: PA23 I/O (port) 01: $\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/DQMUU output (BSC)*[2] 11: TIC5W input (MTU2) Other than above: Setting prohibited
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9	PA22MD1	0	R/W	PA22 Mode
8	PA22MD0	0*1	R/W	Select the function of the PA22/$\overline{\text{WRHL}}$/$\overline{\text{ICIORD}}$/DQMUL/TIC5V pin. 00: PA22 I/O (port) 01: $\overline{\text{WRHL}}$/$\overline{\text{ICIORD}}$/DQMUL output (BSC)*2 11: TIC5V input (MTU2) Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PA21MD1	0	R/W	PA21 Mode
4	PA21MD0	0	R/W	Select the function of the PA21/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/$\overline{\text{CASU}}$/TIC5U pin. 00: PA21 I/O (port) 01: $\overline{\text{CS5}}$/$\overline{\text{CE1A}}$ output (BSC)*2 10: $\overline{\text{CASU}}$ output (BSC)*2 11: TIC5U input (MTU2)
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PA20MD1	0	R/W	PA20 Mode
0	PA20MD0	0	R/W	Select the function of the PA20/$\overline{\text{CS4}}$/$\overline{\text{RASU}}$ pin. 00: PA20 I/O (port) 01: $\overline{\text{CS4}}$ output (BSC)*2 10: $\overline{\text{RASU}}$ output (BSC)*2 Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register H1 (PACRH1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PA19 MD1	PA19 MD0	-	-	PA18 MD1	PA18 MD0	-	-	PA17 MD1	PA17 MD0	-	PA16 MD2	PA16 MD1	PA16 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PA19MD1	0	R/W	PA19 Mode
12	PA19MD0	0	R/W	Select the function of the PA19/$\overline{\text{BACK}}$/TEND1 pin. 00: PA19 I/O (port) 01: $\overline{\text{BACK}}$ output (BSC)* 10: TEND1 output (DMAC)* Other than above: Setting prohibited
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PA18MD1	0	R/W	PA18 Mode
8	PA18MD0	0	R/W	Select the function of the PA18/$\overline{\text{BREQ}}$/TEND0 pin. 00: PA18 I/O (port) 01: $\overline{\text{BREQ}}$ input (BSC)* 10: TEND0 output (DMAC)* Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PA17MD1	0	R/W	PA17 Mode
4	PA17MD0	0	R/W	Select the function of the PA17/$\overline{\text{WAIT}}$/DACK2 pin. 00: PA17 I/O (port) 01: $\overline{\text{WAIT}}$ input (BSC)* 10: DACK2 output (DMAC)* Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PA16MD2	0	R/W	PA16 Mode
1	PA16MD1	0	R/W	Select the function of the
0	PA16MD0	0	R/W	PA16/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/DQMUU/CKE/DREQ2/$\overline{\text{AUDSYNC}}$ pin. Fixed to $\overline{\text{AUDSYNC}}$ output when using the AUD function of the E10A. 000: PA16 I/O (port) 001: $\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/DQMUU output (BSC)* 010: DREQ2 input (DMAC) 101: CKE output (BSC)* Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L4 (PACRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA15 MD2	PA15 MD1	PA15 MD0	-	PA14 MD2	PA14 MD1	PA14 MD0	-	PA13 MD2	PA13 MD1	PA13 MD0	-	PA12 MD2	PA12 MD1	PA12 MD0
Initial value:	0	0	0	0*1	0	0	0	0*2	0	0	0	0*2	0	0	0	0*2
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Notes: 1. The initial value is 1 in the on-chip ROM enabled/disabled external-extension mode.
2. The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PA15MD2	0	R/W	PA15 Mode
13	PA15MD1	0	R/W	Select the function of the PA15/CK pin.
12	PA15MD0	0*1	R/W	000: PA15 I/O (port) 001: CK output (CPG)*3 Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA14MD2	0	R/W	PA14 Mode
9	PA14MD1	0	R/W	Select the function of the PA14/$\overline{RD}$ pin.
8	PA14MD0	0*2	R/W	000: PA14 I/O (port) 001: $\overline{RD}$ output (BSC)*3 Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PA13MD2	0	R/W	PA13 Mode
5	PA13MD1	0	R/W	Select the function of the
4	PA13MD0	0*2	R/W	PA13/$\overline{WRH}$/DQMLU/$\overline{WE}$/$\overline{POE7}$ pin. 000: PA13 I/O (port) 001: $\overline{WRH}$/DQMLU/$\overline{WE}$ output (BSC)*3 011: $\overline{POE7}$ input (POE) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PA12MD2	0	R/W	PA12 Mode
1	PA12MD1	0	R/W	Select the function of the PA12/$\overline{WRL}$/DQMLL/$\overline{POE6}$
0	PA12MD0	0*2	R/W	pin. 000: PA12 I/O (port) 001: $\overline{WRL}$/DQMLL output (BSC)*3 011: $\overline{POE6}$ input (POE) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM enabled/disabled external-extension mode.
2. The initial value is 1 in the on-chip ROM disabled external-extension mode.
3. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L3 (PACRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA11 MD2	PA11 MD1	PA11 MD0	-	PA10 MD2	PA10 MD1	PA10 MD0	-	PA9 MD2	PA9 MD1	PA9 MD0	-	PA8 MD2	PA8 MD1	PA8 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PA11MD2	0	R/W	PA11 Mode
13	PA11MD1	0	R/W	Select the function of the PA11/$\overline{\text{CS1}}$/$\overline{\text{POE5}}$ pin.
12	PA11MD0	0*[1]	R/W	000: PA11 I/O (port) 001: $\overline{\text{CS1}}$ output (BSC)*[3] 011: $\overline{\text{POE5}}$ input (POE)*[2] Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA10MD2	0	R/W	PA10 Mode
9	PA10MD1	0	R/W	Select the function of the PA10/$\overline{\text{CS0}}$/$\overline{\text{POE4}}$ pin.
8	PA10MD0	0*[1]	R/W	000: PA10 I/O (port) 001: $\overline{\text{CS0}}$ output (BSC)*[3] 011: $\overline{\text{POE4}}$ input (POE) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PA9MD2	0	R/W	PA9 Mode
5	PA9MD1	0	R/W	Select the function of the
4	PA9MD0	0	R/W	PA9/$\overline{\text{FRAME}}$/CKE/IRQ3/TCLKD pin. 000: PA9 I/O (port) 001: TCLKD input (MTU2) 010: IRQ3 input (INTC) 011: $\overline{\text{FRAME}}$ output (BSC)*3 101: CKE output (BSC)*3 Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PA8MD2	0	R/W	PA8 Mode
1	PA8MD1	0	R/W	Select the function of the PA8/RDWR/IRQ2/TCLKC
0	PA8MD0	0	R/W	pin. 000: PA8 I/O (port) 001: TCLKC input (MTU2) 010: IRQ2 input (INTC) 101: RDWR output (BSC)*3 Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. When the $\overline{\text{POE5}}$ input is selected, this setting cannot be changed.
3. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L2 (PACRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA7 MD2	PA7 MD1	PA7 MD0	-	PA6 MD2	PA6 MD1	PA6 MD0	-	PA5 MD2	PA5 MD1	PA5 MD0	-	PA4 MD2	PA4 MD1	PA4 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PA7MD2	0	R/W	PA7 Mode
13	PA7MD1	0	R/W	Select the function of the PA7/$\overline{CS3}$/TCLKB pin.
12	PA7MD0	0	R/W	000: PA7 I/O (port) 001: TCLKB input (MTU2) 010: $\overline{CS3}$ output (BSC)* Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA6MD2	0	R/W	PA6 Mode
9	PA6MD1	0	R/W	Select the function of the PA6/$\overline{CS2}$/TCLKA pin.
8	PA6MD0	0	R/W	000: PA6 I/O (port) 001: TCLKA input (MTU2) 010: $\overline{CS2}$ output (BSC)* Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PA5MD2	0	R/W	PA5 Mode
5	PA5MD1	0	R/W	Select the function of the PA5/A22/DREQ1/IRQ1/SCK1 pin.
4	PA5MD0	0	R/W	000: PA5 I/O (port) 001: SCK1 I/O (SCI) 010: DREQ1 input (DMAC) 011: IRQ1 input (INTC) 101: A22 output (BSC)* Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PA4MD2	0	R/W	PA4 Mode
1	PA4MD1	0	R/W	Select the function of the PA4/A23/TXD1 pin.
0	PA4MD0	0	R/W	000: PA4 I/O (port) 001: TXD1 output (SCI) 101: A23 output (BSC)* Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port A Control Register L1 (PACRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA3 MD2	PA3 MD1	PA3 MD0	-	PA2 MD2	PA2 MD1	PA2 MD0	-	PA1 MD2	PA1 MD1	PA1 MD0	-	PA0 MD2	PA0 MD1	PA0 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PA3MD2	0	R/W	PA3 Mode
13	PA3MD1	0	R/W	Select the function of the PA3/A24/RXD1 pin.
12	PA3MD0	0	R/W	000: PA3 I/O (port) 001: RXD1 input (SCI) 101: A24 output (BSC)* Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA2MD2	0	R/W	PA2 Mode
9	PA2MD1	0	R/W	Select the function of the PA2/A25/DREQ0/IRQ0/SCK0 pin.
8	PA2MD0	0	R/W	000: PA2 I/O (port) 001: SCK0 I/O (SCI) 010: DREQ0 input (DMAC) 011: IRQ0 input (INTC) 101: A25 output (BSC)* Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PA1MD2	0	R/W	PA1 Mode
5	PA1MD1	0	R/W	Select the function of the PA1/$\overline{\text{CS5}}$/$\overline{\text{CE1A}}$/TXD0 pin.
4	PA1MD0	0	R/W	000: PA1 I/O (port) 001: TXD0 output (SCI) 101: $\overline{\text{CS5}}$/$\overline{\text{CE1A}}$ output (BSC)* Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PA0MD2	0	R/W	PA0 Mode
1	PA0MD1	0	R/W	Select the function of the PA0/$\overline{CS4}$/RXD0 pin.
0	PA0MD0	0	R/W	000: PA0 I/O (port)
				001: RXD0 input (SCI)
				101: $\overline{CS4}$ output (BSC)*
				Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

21.1.3 Port B I/O Register L (PBIORL)

PBIORL is a 16-bit readable/writable register that is used to set the pins on port B as inputs or outputs. Bits PB9IOR to PB0IOR correspond to pins PB9 to PB0 (names of multiplexed pins are here given as port names and pin numbers alone). PBIORL is enabled when the port B pins are functioning as general-purpose inputs/outputs (PB9 to PB0). In other states, PBIORL is disabled.

A given pin on port B will be an output pin if the corresponding bit in PBIORL is set to 1, and an input pin if the bit is cleared to 0. However, bit 3 of PBIORL is disabled in SH7083.

Bits 15 to 10 of PBIORL are reserved. These bits are always read as 0. The write value should always be 0.

The initial value of PBIORL is H'0000.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PB9 IOR	PB8 IOR	PB7 IOR	PB6 IOR	PB5 IOR	PB4 IOR	PB3 IOR	PB2 IOR	PB1 IOR	PB0 IOR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

21.1.4 Port B Control Registers L1 to L3 (PBCRL1 to PBCRL3)

PBCRL1 to PBCRL3 are 16-bit readable/writable registers that are used to select the function of the multiplexed pins on port B.

SH7083:

- Port B Control Register L3 (PBCRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	PB9 MD2	PB9 MD1	PB9 MD0	-	PB8 MD2	PB8 MD1	PB8 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PB9MD2	0	R/W	PB9 Mode
5	PB9MD1	0	R/W	Select the function of the PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$ pin.
4	PB9MD0	0	R/W	000: PB9 I/O (port)
				001: IRQ7 input (INTC)
				010: A21 output (BSC)*
				011: $\overline{\text{ADTRG}}$ input (A/D)
				110: $\overline{\text{POE8}}$ input (POE)
				Other than above: Setting prohibited
3	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
2	PB8MD2	0	R/W	PB8 Mode
1	PB8MD1	0	R/W	Select the function of the PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0 pin.
0	PB8MD0	0	R/W	000: PB8 I/O (port)
				001: IRQ6 input (INTC)
				010: A20 output (BSC)*
				011: $\overline{\text{WAIT}}$ input (BSC)*
				100: SCK0 I/O (SCI)
				Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port B Control Register L2 (PBCRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PB7 MD2	PB7 MD1	PB7 MD0	-	PB6 MD2	PB6 MD1	PB6 MD0	-	PB5 MD2	PB5 MD1	PB5 MD0	-	PB4 MD2	PB4 MD1	PB4 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PB7MD2	0	R/W	PB7 Mode
13	PB7MD1	0	R/W	Select the function of the PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0 pin.
12	PB7MD0	0	R/W	000: PB7 I/O (port) 001: IRQ5 input (INTC) 010: A19 output (BSC)*[1] 011: $\overline{\text{BREQ}}$ input (BSC)*[1] 100: TXD0 output (SCI) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PB6MD2	0	R/W	PB6 Mode
9	PB6MD1	0	R/W	Select the function of the PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0 pin.
8	PB6MD0	0	R/W	000: PB6 I/O (port) 001: IRQ4 input (INTC) 010: A18 output (BSC)*[1] 011: $\overline{\text{BACK}}$ output (BSC)*[1] 100: RXD0 input (SCI) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PB5MD2	0	R/W	PB5 Mode
5	PB5MD1	0	R/W	Select the function of the PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$ pin.
4	PB5MD0	0	R/W	000: PB5 I/O (port) 001: IRQ3 input (INTC) 010: $\overline{\text{POE3}}$ input (POE)*2 100: $\overline{\text{CASL}}$ output (BSC)*1 Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PB4MD2	0	R/W	PB4 Mode
1	PB4MD1	0	R/W	Select the function of the PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$ pin.
0	PB4MD0	0	R/W	000: PB4 I/O (port) 001: IRQ2 input (INTC) 010: $\overline{\text{POE2}}$ input (POE) 100: $\overline{\text{RASL}}$ output (BSC)*1 Other than above: Setting prohibited

Notes: 1. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.
2. When the $\overline{\text{POE3}}$ input is selected, this setting cannot be changed.

- Port B Control Register L1 (PBCRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	PB2 MD2	PB2 MD1	PB2 MD0	-	PB1 MD2	PB1 MD1	PB1 MD0	-	PB0 MD2	PB0 MD1	PB0 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0*	0	0	0	0*
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	PB2MD2	0	R/W	PB2 Mode
9	PB2MD1	0	R/W	Select the function of the PB2/IRQ0/$\overline{\text{POE0}}$ pin.
8	PB2MD0	0	R/W	000: PB2 I/O (port)
				001: IRQ0 input (INTC)
				010: $\overline{\text{POE0}}$ input (POE)
				Other than above: Setting prohibited
7	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
6	PB1MD2	0	R/W	PB1 Mode
5	PB1MD1	0	R/W	Select the function of the PB1/A17/TIC5W pin.
4	PB1MD0	0*[1]	R/W	000: PB1 I/O (port)
				001: A17 output (BSC)*[2]
				011: TIC5W input (MTU2)
				Other than above: Setting prohibited
3	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
2	PB0MD2	0	R/W	PB0 Mode
1	PB0MD1	0	R/W	Select the function of the PB0/A16/TIC5WS pin.
0	PB0MD0	0*[1]	R/W	000: PB0 I/O (port)
				001: A16 output (BSC)*[2]
				011: TIC5WS input (MTU2S)
				Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

SH7084/SH7085/SH7086:

- Port B Control Register L3 (PBCRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	PB9MD2	PB9MD1	PB9MD0	-	PB8MD2	PB8MD1	PB8MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	PB9MD2	0	R/W	PB9 Mode
5	PB9MD1	0	R/W	Select the function of the PB9/A21/IRQ7/$\overline{\text{ADTRG}}$/$\overline{\text{POE8}}$ pin.
4	PB9MD0	0	R/W	000: PB9 I/O (port) 001: IRQ7 input (INTC) 010: A21 output (BSC)* 011: $\overline{\text{ADTRG}}$ input (A/D) 110: $\overline{\text{POE8}}$ input (POE) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PB8MD2	0	R/W	PB8 Mode
1	PB8MD1	0	R/W	Select the function of the PB8/A20/$\overline{\text{WAIT}}$/IRQ6/SCK0 pin.
0	PB8MD0	0	R/W	000: PB8 I/O (port) 001: IRQ6 input (INTC) 010: A20 output (BSC)* 011: $\overline{\text{WAIT}}$ input (BSC)* 100: SCK0 I/O (SCI) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port B Control Register L2 (PBCRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PB7 MD2	PB7 MD1	PB7 MD0	-	PB6 MD2	PB6 MD1	PB6 MD0	-	PB5 MD2	PB5 MD1	PB5 MD0	-	PB4 MD2	PB4 MD1	PB4 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PB7MD2	0	R/W	PB7 Mode
13	PB7MD1	0	R/W	Select the function of the PB7/A19/$\overline{\text{BREQ}}$/IRQ5/TXD0 pin.
12	PB7MD0	0	R/W	000: PB7 I/O (port) 001: IRQ5 input (INTC) 010: A19 output (BSC)*2 011: $\overline{\text{BREQ}}$ input (BSC)*2 100: TXD0 output (SCI) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PB6MD2	0	R/W	PB6 Mode
9	PB6MD1	0	R/W	Select the function of the PB6/A18/$\overline{\text{BACK}}$/IRQ4/RXD0 pin.
8	PB6MD0	0	R/W	000: PB6 I/O (port) 001: IRQ4 input (INTC) 010: A18 output (BSC)*2 011: $\overline{\text{BACK}}$ output (BSC)*2 100: RXD0 input (SCI) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PB5MD2	0	R/W	PB5 Mode
5	PB5MD1	0	R/W	Select the function of the PB5/$\overline{\text{CASL}}$/IRQ3/$\overline{\text{POE3}}$ pin.
4	PB5MD0	0	R/W	000: PB5 I/O (port) 001: IRQ3 input (INTC) 010: $\overline{\text{POE3}}$ input (POE)*[1] 100: $\overline{\text{CASL}}$ output (BSC)*[2] Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PB4MD2	0	R/W	PB4 Mode
1	PB4MD1	0	R/W	Select the function of the PB4/$\overline{\text{RASL}}$/IRQ2/$\overline{\text{POE2}}$ pin.
0	PB4MD0	0	R/W	000: PB4 I/O (port) 001: IRQ2 input (INTC) 010: $\overline{\text{POE2}}$ input (POE) 100: $\overline{\text{RASL}}$ output (BSC)*[2] Other than above: Setting prohibited

Notes: 1. When the $\overline{\text{POE3}}$ input is selected, this setting cannot be changed.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port B Control Register L1 (PBCRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PB3 MD2	PB3 MD1	PB3 MD0	-	PB2 MD2	PB2 MD1	PB2 MD0	-	PB1 MD2	PB1 MD1	PB1 MD0	-	PB0 MD2	PB0 MD1	PB0 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0*	0	0	0	0*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PB3MD2	0	R/W	PB3 Mode
13	PB3MD1	0	R/W	Select the function of the PB3/IRQ1/$\overline{\text{POE1}}$/SDA pin.
12	PB3MD0	0	R/W	000: PB3 I/O (port)
				001: IRQ1 input (INTC)
				010: $\overline{\text{POE1}}$ input (POE)
				100: SDA I/O (IIC2)
				Other than above: Setting prohibited
11	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
10	PB2MD2	0	R/W	PB2 Mode
9	PB2MD1	0	R/W	Select the function of the PB2/IRQ0/$\overline{\text{POE0}}$/SCL pin.
8	PB2MD0	0	R/W	000: PB2 I/O (port)
				001: IRQ0 input (INTC)
				010: $\overline{\text{POE0}}$ input (POE)
				100: SCL I/O (IIC2)
				Other than above: Setting prohibited
7	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
6	PB1MD2	0	R/W	PB1 Mode
5	PB1MD1	0	R/W	Select the function of the PB1/A17/TIC5W pin.
4	PB1MD0	0*[1]	R/W	000: PB1 I/O (port)
				001: A17 output (BSC)*[2]
				011: TIC5W input (MTU2)
				Other than above: Setting prohibited
3	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PB0MD2	0	R/W	PB0 Mode
1	PB0MD1	0	R/W	Select the function of the PB0/A16/TIC5WS pin.
0	PB0MD0	0*1	R/W	000: PB0 I/O (port)
				001: A16 output (BSC)*2
				011: TIC5WS input (MTU2S)
				Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

21.1.5 Port C I/O Registers L, H (PCIORL, PCIORH)

PCIORL and PCIORH are 16-bit readable/writable registers that are used to set the pins on port C as inputs or outputs. Bits PC25IOR to PC18IOR and PC15IOR to PC0IOR correspond to pins PC25 to PC18 and PC15 to PC0, respectively (names of multiplexed pins are here given as port names and pin numbers alone). PCIORL is enabled when the port C pins are functioning as general-purpose inputs/outputs (PC15 to PC0). In other states, PCIORL is disabled. PCIORH is enabled when the port C pins are functioning as general-purpose input/output (PC25 to PC18). In other states, PCIORH is disabled.

A given pin on port C will be an output pin if the corresponding bit in PCIORH or PCIORL is set to 1, and an input pin if the bit is cleared to 0.

However, bits 9 to 2 of PCIORH are disabled in SH7083/SH7084/SH7085.

Bits 15 to 10, 1 and 0 of PCIORH are reserved. These bits are always read as 0. The write value should always be 0.

The initial values of PCIORL and PCIORH are H'0000, respectively.

- Port C I/O Register H (PCIORH)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PC25 IOR	PC24 IOR	PC23 IOR	PC22 IOR	PC21 IOR	PC20 IOR	PC19 IOR	PC18 IOR	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

- Port C I/O Register L (PCIORL)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PC15 IOR	PC14 IOR	PC13 IOR	PC12 IOR	PC11 IOR	PC10 IOR	PC9 IOR	PC8 IOR	PC7 IOR	PC6 IOR	PC5 IOR	PC4 IOR	PC3 IOR	PC2 IOR	PC1 IOR	PC0 IOR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

21.1.6 Port C Control Registers L1 to L4, H1 to H3 (PCCRL1 to PCCRL4, PCCRH1 to PCCRH3)

PCCRL1 to PCCRL4 and PCCRH1 to PCCRH3 are 16-bit readable/writable registers that are used to select the functions of the multiplexed pins on port C.

SH7083/SH7084/SH7085:

- Port C Control Registers H3 to H1 (PCCRH3 to PCCRH1)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- Port C Control Register L4 (PCCRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC15 MD0	-	-	-	PC14 MD0	-	-	-	PC13 MD0	-	-	-	PC12 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	PC15MD0	0*[1]	R/W	PC15 Mode Select the function of the PC15/A15 pin. 0: PC15 I/O (port) 1: A15 output (BSC)*[2]

Bit	Bit Name	Initial Value	R/W	Description
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PC14MD0	0*[1]	R/W	PC14 Mode Select the function of the PC14/A14 pin. 0: PC14 I/O (port) 1: A14 output (BSC)*[2]
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PC13MD0	0*[1]	R/W	PC13 Mode Select the function of the PC13/A13 pin. 0: PC13 I/O (port) 1: A13 output (BSC)*[2]
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PC12MD0	0*[1]	R/W	PC12 Mode Select the function of the PC12/A12 pin. 0: PC12 I/O (port) 1: A12 output (BSC)*[2]

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port C Control Register L3 (PCCRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC11 MD0	-	-	-	PC10 MD0	-	-	-	PC9 MD0	-	-	-	PC8 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	PC11MD0	0*[1]	R/W	PC11 Mode Select the function of the PC11/A11 pin. 0: PC11 I/O (port) 1: A11 output (BSC)*[2]
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PC10MD0	0*[1]	R/W	PC10 Mode Select the function of the PC10/A10 pin. 0: PC10 I/O (port) 1: A10 output (BSC)*[2]
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PC9MD0	0*[1]	R/W	PC9 Mode Select the function of the PC9/A9 pin. 0: PC9 I/O (port) 1: A9 output (BSC)*[2]
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	PC8MD0	0*[1]	R/W	PC8 Mode Select the function of the PC8/A8 pin. 0: PC8 I/O (port) 1: A8 output (BSC)*[2]

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port C Control Register L2 (PCCRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC7 MD0	-	-	-	PC6 MD0	-	-	-	PC5 MD0	-	-	-	PC4 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	PC7MD0	0*[1]	R/W	PC7 Mode Select the function of the PC7/A7 pin. 0: PC7 I/O (port) 1: A7 output (BSC)*[2]
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PC6MD0	0*[1]	R/W	PC6 Mode Select the function of the PC6/A6 pin. 0: PC6 I/O (port) 1: A6 output (BSC)*[2]
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	PC5MD0	0*[1]	R/W	PC5 Mode Select the function of the PC5/A5 pin. 0: PC5 I/O (port) 1: A5 output (BSC)*[2]
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PC4MD0	0*[1]	R/W	PC4 Mode Select the function of the PC4/A4 pin. 0: PC4 I/O (port) 1: A4 output (BSC)*[2]

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port C Control Register L1 (PCCRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC3MD0	-	-	-	PC2MD0	-	-	-	PC1MD0	-	-	-	PC0MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	PC3MD0	0*[1]	R/W	PC3 Mode Select the function of the PC3/A3 pin. 0: PC3 I/O (port) 1: A3 output (BSC)*[2]
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
8	PC2MD0	0*[1]	R/W	PC2 Mode Select the function of the PC2/A2 pin. 0: PC2 I/O (port) 1: A2 output (BSC)*[2]
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PC1MD0	0*[1]	R/W	PC1 Mode Select the function of the PC1/A1 pin. 0: PC1 I/O (port) 1: A1 output (BSC)*[2]
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PC0MD0	0*[1]	R/W	PC0 Mode Select the function of the PC0/A0 pin. 0: PC0 I/O (port) 1: A0 output (BSC)*[2]

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

SH7086:

- Port C Control Register H3 (PCCRH3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	PC25 MD0	-	-	-	PC24 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0*	0	0	0	0*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PC25MD0	0*[1]	R/W	PC25 Mode Select the function of the PC25/A25 pin. 0: PC25 I/O (port) 1: A25 output (BSC)*[2]
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PC24MD0	0*[1]	R/W	PC24 Mode Select the function of the PC24/A24 pin. 0: PC24 I/O (port) 1: A24 output (BSC)*[2]

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port C Control Register H2 (PCCRH2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC23 MD0	-	-	-	PC22 MD0	-	-	-	PC21 MD0	-	-	-	PC20 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	PC23MD0	0*[1]	R/W	PC23 Mode Select the function of the PC23/A23 pin. 0: PC23 I/O (port) 1: A23 output (BSC)*[2]
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PC22MD0	0*[1]	R/W	PC22 Mode Select the function of the PC22/A22 pin. 0: PC22 I/O (port) 1: A22 output (BSC)*[2]
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PC21MD0	0*[1]	R/W	PC21 Mode Select the function of the PC21/A21 pin. 0: PC21 I/O (port) 1: A21 output (BSC)*[2]
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PC20MD0	0*[1]	R/W	PC20 Mode Select the function of the PC20/A20 pin. 0: PC20 I/O (port) 1: A20 output (BSC)*[2]

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port C Control Register H1 (PCCRH1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC19 MD0	-	-	-	PC18 MD0	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	PC19MD0	0*[1]	R/W	PC19 Mode Select the function of the PC19/A19 pin. 0: PC19 I/O (port) 1: A19 output (BSC)*[2]
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PC18MD0	0*[1]	R/W	PC18 Mode Select the function of the PC18/A18 pin. 0: PC18 I/O (port) 1: A18 output (BSC)*[2]
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port C Control Register L4 (PCCRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC15 MD0	-	-	-	PC14 MD0	-	-	-	PC13 MD0	-	-	-	PC12 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	PC15MD0	0*[1]	R/W	PC15 Mode Select the function of the PC15/A15 pin. 0: PC15 I/O (port) 1: A15 output (BSC)*[2]
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PC14MD0	0*[1]	R/W	PC14 Mode Select the function of the PC14/A14 pin. 0: PC14 I/O (port) 1: A14 output (BSC)*[2]
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PC13MD0	0*[1]	R/W	PC13 Mode Select the function of the PC13/A13 pin. 0: PC13 I/O (port) 1: A13 output (BSC)*[2]
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PC12MD0	0*[1]	R/W	PC12 Mode Select the function of the PC12/A12 pin. 0: PC12 I/O (port) 1: A12 output (BSC)*[2]

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port C Control Register L3 (PCCRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC11 MD0	-	-	-	PC10 MD0	-	-	-	PC9 MD0	-	-	-	PC8 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	PC11MD0	0*[1]	R/W	PC11 Mode Select the function of the PC11/A11 pin. 0: PC11 I/O (port) 1: A11 output (BSC)*[2]
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PC10MD0	0*[1]	R/W	PC10 Mode Select the function of the PC10/A10 pin. 0: PC10 I/O (port) 1: A10 output (BSC)*[2]
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PC9MD0	0*[1]	R/W	PC9 Mode Select the function of the PC9/A9 pin. 0: PC9 I/O (port) 1: A9 output (BSC)*[2]
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	PC8MD0	0*[1]	R/W	PC8 Mode Select the function of the PC8/A8 pin. 0: PC8 I/O (port) 1: A8 output (BSC)*[2]

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port C Control Register L2 (PCCRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC7 MD0	-	-	-	PC6 MD0	-	-	-	PC5 MD0	-	-	-	PC4 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	PC7MD0	0*[1]	R/W	PC7 Mode Select the function of the PC7/A7 pin. 0: PC7 I/O (port) 1: A7 output (BSC)*[2]
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PC6MD0	0*[1]	R/W	PC6 Mode Select the function of the PC6/A6 pin. 0: PC6 I/O (port) 1: A6 output (BSC)*[2]
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	PC5MD0	0*[1]	R/W	PC5 Mode Select the function of the PC5/A5 pin. 0: PC5 I/O (port) 1: A5 output (BSC)*[2]
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PC4MD0	0*[1]	R/W	PC4 Mode Select the function of the PC4/A4 pin. 0: PC4 I/O (port) 1: A4 output (BSC)*[2]

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port C Control Register L1 (PCCRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC3MD0	-	-	-	PC2MD0	-	-	-	PC1MD0	-	-	-	PC0MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	PC3MD0	0*[1]	R/W	PC3 Mode Select the function of the PC3/A3 pin. 0: PC3 I/O (port) 1: A3 output (BSC)*[2]
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
8	PC2MD0	0*[1]	R/W	PC2 Mode Select the function of the PC2/A2 pin. 0: PC2 I/O (port) 1: A2 output (BSC)*[2]
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PC1MD0	0*[1]	R/W	PC1 Mode Select the function of the PC1/A1 pin. 0: PC1 I/O (port) 1: A1 output (BSC)*[2]
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PC0MD0	0*[1]	R/W	PC0 Mode Select the function of the PC0/A0 pin. 0: PC0 I/O (port) 1: A0 output (BSC)*[2]

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

21.1.7 Port D I/O Registers L, H (PDIORL, PDIORH)

PDIORL and PDIORH are 16-bit readable/writable registers that are used to set the pins on port D as inputs or outputs. Bits PD31IOR to PD0IOR correspond to pins PD31 to PD0 (names of multiplexed pins are here given as port names and pin numbers alone). PDIORL is enabled when the port D pins are functioning as general-purpose inputs/outputs (PD15 to PD0), and the TIOC pin is functioning as inputs/outputs of MTU2S. In other states, PDIORL is disabled. PDIORH is enabled when the port D pins are functioning as general-purpose inputs/outputs (PD31 to PD16), and the TIOC pin is functioning as inputs/outputs of MTU2S. In other states, PDIORH is disabled.

A given pin on port D will be an output pin if the corresponding bit in PDIORH or PDIORL is set to 1, and an input pin if the bit is cleared to 0. However, PDIORH is disabled in SH7083/SH7084.

The initial values of PDIORL and PDIORH are H'0000, respectively.

- Port D I/O Register H (PDIORH)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD31 IOR	PD30 IOR	PD29 IOR	PD28 IOR	PD27 IOR	PD26 IOR	PD25 IOR	PD24 IOR	PD23 IOR	PD22 IOR	PD21 IOR	PD20 IOR	PD19 IOR	PD18 IOR	PD17 IOR	PD16 IOR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Port D I/O Register L (PDIORL)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD15 IOR	PD14 IOR	PD13 IOR	PD12 IOR	PD11 IOR	PD10 IOR	PD9 IOR	PD8 IOR	PD7 IOR	PD6 IOR	PD5 IOR	PD4 IOR	PD3 IOR	PD2 IOR	PD1 IOR	PD0 IOR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

21.1.8 Port D Control Registers L1 to L4, H1 to H4 (PDCRL1 to PDCRL4, PDCRH1 to PDCRH4)

PDCRL1 to PDCRL4 and PDCRH1 to PDCRH4 are 16-bit readable/writable registers that are used to select the functions of the multiplexed pins on port D.

SH7083/SH7084:

- Port D Control Registers H4 to H1 (PDCRH4 to PDCRH1)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- Port D Control Register L4 (PDCRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PD15 MD1	PD15 MD0	-	-	PD14 MD1	PD14 MD0	-	-	PD13 MD1	PD13 MD0	-	-	PD12 MD1	PD12 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled 16-bit external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
13	PD15MD1	0	R/W	PD15 Mode
12	PD15MD0	0*[1]	R/W	Select the function of the PD15/D15/TIOC4DS/$\overline{\text{AUDSYNC}}$ pin. Fixed to $\overline{\text{AUDSYNC}}$ output when using the AUD function of the E10A. 00: PD15 I/O (port) 01: D15 I/O (BSC)*[2] 11: TIOC4DS I/O (MTU2S) Other than above: Setting prohibited
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PD14MD1	0	R/W	PD14 Mode
8	PD14MD0	0*[1]	R/W	Select the function of the PD14/D14/TIOC4CS/AUDCK pin. Fixed to AUDCK output when using the AUD function of the E10A. 00: PD14 I/O (port) 01: D14 I/O (BSC)*[2] 11: TIOC4CS I/O (MTU2S) Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PD13MD1	0	R/W	PD13 Mode
4	PD13MD0	0*[1]	R/W	Select the function of the PD13/D13/TIOC4BS pin. 00: PD13 I/O (port) 01: D13 I/O (BSC)*[2] 11: TIOC4BS I/O (MTU2S) Other than above: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	PD12MD1	0	R/W	PD12 Mode
0	PD12MD0	0*[1]	R/W	Select the function of the PD12/D12/TIOC4AS pin. 00: PD12 I/O (port) 01: D12 I/O (BSC)*[2] 11: TIOC4AS I/O (MTU2S) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled 16-bit external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port D Control Register L3 (PDCRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PD11 MD1	PD11 MD0	-	PD10 MD2	PD10 MD1	PD10 MD0	-	PD9 MD2	PD9 MD1	PD9 MD0	-	PD8 MD2	PD8 MD1	PD8 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled 16-bit external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PD11MD1	0	R/W	PD11 Mode
12	PD11MD0	0*[1]	R/W	Select the function of the PD11/D11/TIOC3DS/AUDATA3 pin. Fixed to AUDATA3 output when using the AUD function of the E10A. 00: PD11 I/O (port) 01: D11 I/O (BSC)*[2] 11: TIOC3DS I/O (MTU2S) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	PD10MD2	0	R/W	PD10 Mode
9	PD10MD1	0	R/W	Select the function of the
8	PD10MD0	0*[1]	R/W	PD10/D10/TIOC3CS/AUDATA2 pin. Fixed to AUDATA2 output when using the AUD function of the E10A. 000: PD10 I/O (port) 001: D10 I/O (BSC)*[2] 011: TIOC3CS I/O (MTU2S) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PD9MD2	0	R/W	PD9 Mode
5	PD9MD1	0	R/W	Select the function of the
4	PD9MD0	0*[1]	R/W	PD9/D9/TIOC3BS/AUDATA1 pin. Fixed to AUDATA1 output when using the AUD function of the E10A. 000: PD9 I/O (port) 001: D9 I/O (BSC)*[2] 011: TIOC3BS I/O (MTU2S) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PD8MD2	0	R/W	PD8 Mode
1	PD8MD1	0	R/W	Select the function of the
0	PD8MD0	0*[1]	R/W	PD8/D8/TIOC3AS/AUDATA0 pin. Fixed to AUDATA0 output when using the AUD function of the E10A. 000: PD8 I/O (port) 001: D8 I/O (BSC)*[2] 011: TIOC3AS I/O (MTU2S) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled 16-bit external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port D Control Register L2 (PDCRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PD7 MD2	PD7 MD1	PD7 MD0	-	PD6 MD2	PD6 MD1	PD6 MD0	-	PD5 MD2	PD5 MD1	PD5 MD0	-	PD4 MD2	PD4 MD1	PD4 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PD7MD2	0	R/W	PD7 Mode
13	PD7MD1	0	R/W	Select the function of the PD7/D7/TIC5WS pin.
12	PD7MD0	0*[1]	R/W	000: PD7 I/O (port) 001: D7 I/O (BSC)*[2] 010: TIC5WS input (MTU2S) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PD6MD2	0	R/W	PD6 Mode
9	PD6MD1	0	R/W	Select the function of the PD6/D6/TIC5VS pin.
8	PD6MD0	0*[1]	R/W	000: PD6 I/O (port) 001: D6 I/O (BSC)*[2] 010: TIC5VS input (MTU2S) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PD5MD2	0	R/W	PD5 Mode
5	PD5MD1	0	R/W	Select the function of the PD5/D5/TIC5US pin.
4	PD5MD0	0*[1]	R/W	000: PD5 I/O (port) 001: D5 I/O (BSC)*[2] 010: TIC5US input (MTU2S) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PD4MD2	0	R/W	PD4 Mode
1	PD4MD1	0	R/W	Select the function of the PD4/D4/TIC5W pin.
0	PD4MD0	0*[1]	R/W	000: PD4 I/O (port) 001: D4 I/O (BSC)*[2] 010: TIC5W input (MTU2) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port D Control Register L1 (PDCRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PD3 MD2	PD3 MD1	PD3 MD0	-	PD2 MD2	PD2 MD1	PD2 MD0	-	PD1 MD2	PD1 MD1	PD1 MD0	-	PD0 MD2	PD0 MD1	PD0 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PD3MD2	0	R/W	PD3 Mode
13	PD3MD1	0	R/W	Select the function of the PD3/D3/TIC5V pin.
12	PD3MD0	0*[1]	R/W	000: PD3 I/O (port)
				001: D3 I/O (BSC)*[2]
				010: TIC5V input (MTU2)
				Other than above: Setting prohibited
11	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
10	PD2MD2	0	R/W	PD2 Mode
9	PD2MD1	0	R/W	Select the function of the PD2/D2/TIC5U pin.
8	PD2MD0	0*[1]	R/W	000: PD2 I/O (port)
				001: D2 I/O (BSC)*[2]
				010: TIC5U input (MTU2)
				Other than above: Setting prohibited
7	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
6	PD1MD2	0	R/W	PD1 Mode
5	PD1MD1	0	R/W	Select the function of the PD1/D1 pin.
4	PD1MD0	0*[1]	R/W	000: PD1 I/O (port)
				001: D1 I/O (BSC)*[2]
				Other than above: Setting prohibited
3	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
2	PD0MD2	0	R/W	PD0 Mode
1	PD0MD1	0	R/W	Select the function of the PD0/D0 pin.
0	PD0MD0	0*[1]	R/W	000: PD0 I/O (port)
				001: D0 I/O (BSC)*[2]
				Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

SH7085/SH7086:

- Port D Control Register H4 (PDCRH4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PD31 MD1	PD31 MD0	-	-	PD30 MD1	PD30 MD0	-	-	PD29 MD1	PD29 MD0	-	-	PD28 MD1	PD28 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PD31MD1	0	R/W	PD31 Mode
12	PD31MD0	0*[1]	R/W	Select the function of the PD31/D31/TIOC3AS/$\overline{\text{ADTRG}}$ pin. 00: PD31 I/O (port) 01: D31 I/O (BSC)*[2] 10: $\overline{\text{ADTRG}}$ input (A/D) 11: TIOC3AS I/O (MTU2S)
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PD30MD1	0	R/W	PD30 Mode
8	PD30MD0	0*[1]	R/W	Select the function of the PD30/D30/TIOC3CS/$\overline{\text{IRQOUT}}$ pin. 00: PD30 I/O (port) 01: D30 I/O (BSC)*[2] 10: $\overline{\text{IRQOUT}}$ output (INTC) 11: TIOC3CS I/O (MTU2S)
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5	PD29MD1	0	R/W	PD29 Mode
4	PD29MD0	0*[1]	R/W	Select the function of the PD29/D29/$\overline{CS3}$/TIOC3BS pin. 00: PD29 I/O (port) 01: D29 I/O (BSC)*[2] 10: $\overline{CS3}$ output (BSC)*[2] 11: TIOC3BS I/O (MTU2S)
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PD28MD1	0	R/W	PD28 Mode
0	PD28MD0	0*[1]	R/W	Select the function of the PD28/D28/$\overline{CS2}$/TIOC3DS pin. 00: PD28 I/O (port) 01: D28 I/O (BSC)*[2] 10: $\overline{CS2}$ output (BSC)*[2] 11: TIOC3DS I/O (MTU2S)

Notes: 1. The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port D Control Register H3 (PDCRH3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PD27 MD1	PD27 MD0	-	-	PD26 MD1	PD26 MD0	-	-	PD25 MD1	PD25 MD0	-	-	PD24 MD1	PD24 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
13	PD27MD1	0	R/W	PD27 Mode
12	PD27MD0	0*[1]	R/W	Select the function of the PD27/D27/DACK1/TIOC4AS pin. 00: PD27 I/O (port) 01: D27 I/O (BSC)*[2] 10: DACK1 output (DMAC)*[2] 11: TIOC4AS I/O (MTU2S)
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PD26MD1	0	R/W	PD26 Mode
8	PD26MD0	0*[1]	R/W	Select the function of the PD26/D26/DACK0/TIOC4BS pin. 00: PD26 I/O (port) 01: D26 I/O (BSC)*[2] 10: DACK0 output (DMAC)*[2] 11: TIOC4BS I/O (MTU2S)
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PD25MD1	0	R/W	PD25 Mode
4	PD25MD0	0*[1]	R/W	Select the function of the PD25/D25/DREQ1/TIOC4CS pin. 00: PD25 I/O (port) 01: D25 I/O (BSC)*[2] 10: DREQ1 input (DMAC) 11: TIOC4CS I/O (MTU2S)
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	PD24MD1	0	R/W	PD24 Mode
0	PD24MD0	0*[1]	R/W	Select the function of the PD24/D24/DREQ0/TIOC4DS pin. 00: PD24 I/O (port) 01: D24 I/O (BSC)*[2] 10: DREQ0 input (DMAC) 11: TIOC4DS I/O (MTU2S)

Notes: 1. The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port D Control Register H2 (PDCRH2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PD23 MD1	PD23 MD0	-	PD22 MD2	PD22 MD1	PD22 MD0	-	PD21 MD2	PD21 MD1	PD21 MD0	-	PD20 MD2	PD20 MD1	PD20 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PD23MD1	0	R/W	PD23 Mode
12	PD23MD0	0*[1]	R/W	Select the function of the PD23/D23/IRQ7/$\overline{\text{AUDSYNC}}$ pin. Fixed to $\overline{\text{AUDSYNC}}$ output when using the AUD function of the E10A. 00: PD23 I/O (port) 01: D23 I/O (BSC)*[2] 10: IRQ7 input (INTC) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	PD22MD2	0	R/W	PD22 Mode
9	PD22MD1	0	R/W	Select the function of the
8	PD22MD0	0*1	R/W	PD22/D22/IRQ6/TIC5US/AUDCK pin. Fixed to AUDCK output when using the AUD function of the E10A. 000: PD22 I/O (port) 001: D22 I/O (BSC)*2 010: IRQ6 input (INTC) 100: TIC5US I/O (MTU2S) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PD21MD2	0	R/W	PD21 Mode
5	PD21MD1	0	R/W	Select the function of the PD21/D21/IRQ5/TIC5VS
4	PD21MD0	0*1	R/W	pin. 000: PD21 I/O (port) 001: D21 I/O (BSC)*2 010: IRQ5 input (INTC) 100: TIC5VS I/O (MTU2S) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PD20MD2	0	R/W	PD20 Mode
1	PD20MD1	0	R/W	Select the function of the PD20/D20/IRQ4/TIC5WS
0	PD20MD0	0*1	R/W	pin. 000: PD20 I/O (port) 001: D20 I/O (BSC)*2 010: IRQ4 input (INTC) 100: TIC5WS I/O (MTU2S) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port D Control Register H1 (PDCRH1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PD19 MD2	PD19 MD1	PD19 MD0	-	PD18 MD2	PD18 MD1	PD18 MD0	-	PD17 MD2	PD17 MD1	PD17 MD0	-	PD16 MD2	PD16 MD1	PD16 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PD19MD2	0	R/W	PD19 Mode
13	PD19MD1	0	R/W	Select the function of the
12	PD19MD0	0*[1]	R/W	PD19/D19/IRQ3/$\overline{\text{POE7}}$/AUDATA3 pin. Fixed to AUDATA3 output when using the AUD function of the E10A. 000: PD19 I/O (port) 001: D19 I/O (BSC)*[2] 010: IRQ3 input (INTC) 100: $\overline{\text{POE7}}$ input (POE) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PD18MD2	0	R/W	PD18 Mode
9	PD18MD1	0	R/W	Select the function of the
8	PD18MD0	0*[1]	R/W	PD18/D18/IRQ2/$\overline{\text{POE6}}$/AUDATA2 pin. Fixed to AUDATA2 output when using the AUD function of the E10A. 000: PD18 I/O (port) 001: D18 I/O (BSC)*[2] 010: IRQ2 input (INTC) 100: $\overline{\text{POE6}}$ input (POE) Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PD17MD2	0	R/W	PD17 Mode
5	PD17MD1	0	R/W	Select the function of the
4	PD17MD0	0*1	R/W	PD17/D17/IRQ1/$\overline{\text{POE5}}$/AUDATA1 pin. Fixed to AUDATA1 output when using the AUD function of the E10A. 000: PD17 I/O (port) 001: D17 I/O (BSC)*2 010: IRQ1 input (INTC) 100: $\overline{\text{POE5}}$ input (POE) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PD16MD2	0	R/W	PD16 Mode
1	PD16MD1	0	R/W	Select the function of the
0	PD16MD0	0*1	R/W	PD16/D16/IRQ0/$\overline{\text{POE4}}$/AUDATA0 pin. Fixed to AUDATA0 output when using the AUD function of the E10A. 000: PD16 I/O (port) 001: D16 I/O (BSC)*2 010: IRQ0 input (INTC) 100: $\overline{\text{POE4}}$ input (POE) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled 32-bit external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port D Control Register L4 (PDCRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PD15 MD1	PD15 MD0	-	-	PD14 MD1	PD14 MD0	-	-	PD13 MD1	PD13 MD0	-	-	PD12 MD1	PD12 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PD15MD1	0	R/W	PD15 Mode
12	PD15MD0	0*[1]	R/W	Select the function of the PD15/D15/TIOC4DS pin. 00: PD15 I/O (port) 01: D15 I/O (BSC)*[2] 11: TIOC4DS I/O (MTU2S) Other than above: Setting prohibited
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PD14MD1	0	R/W	PD14 Mode
8	PD14MD0	0*[1]	R/W	Select the function of the PD14/D14/TIOC4CS pin. 00: PD14 I/O (port) 01: D14 I/O (BSC)*[2] 11: TIOC4CS I/O (MTU2S) Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PD13MD1	0	R/W	PD13 Mode
4	PD13MD0	0*[1]	R/W	Select the function of the PD13/D13/TIOC4BS pin. 00: PD13 I/O (port) 01: D13 I/O (BSC)*[2] 11: TIOC4BS I/O (MTU2S) Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PD12MD1	0	R/W	PD12 Mode
0	PD12MD0	0*[1]	R/W	Select the function of the PD12/D12/TIOC4AS pin. 00: PD12 I/O (port) 01: D12 I/O (BSC)*[2] 11: TIOC4AS I/O (MTU2S) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port D Control Register L3 (PDCRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PD11 MD1	PD11 MD0	-	PD10 MD2	PD10 MD1	PD10 MD0	-	PD9 MD2	PD9 MD1	PD9 MD0	-	PD8 MD2	PD8 MD1	PD8 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PD11MD1	0	R/W	PD11 Mode
12	PD11MD0	0*[1]	R/W	Select the function of the PD11/D11/TIOC3DS pin. 00: PD11 I/O (port) 01: D11 I/O (BSC)*[2] 11: TIOC3DS I/O (MTU2S) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	PD10MD2	0	R/W	PD10 Mode
9	PD10MD1	0	R/W	Select the function of the PD10/D10/TIOC3CS pin.
8	PD10MD0	0*1	R/W	000: PD10 I/O (port) 001: D10 I/O (BSC)*2 011: TIOC3CS I/O (MTU2S) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PD9MD2	0	R/W	PD9 Mode
5	PD9MD1	0	R/W	Select the function of the PD9/D9/TIOC3BS pin.
4	PD9MD0	0*1	R/W	000: PD9 I/O (port) 001: D9 I/O (BSC)*2 011: TIOC3BS I/O (MTU2S) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PD8MD2	0	R/W	PD8 Mode
1	PD8MD1	0	R/W	Select the function of the PD8/D8/TIOC3AS pin.
0	PD8MD0	0*1	R/W	000: PD8 I/O (port) 001: D8 I/O (BSC)*2 011: TIOC3AS I/O (MTU2S) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port D Control Register L2 (PDCRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PD7 MD2	PD7 MD1	PD7 MD0	-	PD6 MD2	PD6 MD1	PD6 MD0	-	PD5 MD2	PD5 MD1	PD5 MD0	-	PD4 MD2	PD4 MD1	PD4 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PD7MD2	0	R/W	PD7 Mode
13	PD7MD1	0	R/W	Select the function of the PD7/D7/TIC5WS pin.
12	PD7MD0	0*[1]	R/W	000: PD7 I/O (port) 001: D7 I/O (BSC)*[2] 010: TIC5WS input (MTU2S) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PD6MD2	0	R/W	PD6 Mode
9	PD6MD1	0	R/W	Select the function of the PD6/D6/TIC5VS pin.
8	PD6MD0	0*[1]	R/W	000: PD6 I/O (port) 001: D6 I/O (BSC)*[2] 010: TIC5VS input (MTU2S) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PD5MD2	0	R/W	PD5 Mode
5	PD5MD1	0	R/W	Select the function of the PD5/D5/TIC5US pin.
4	PD5MD0	0*[1]	R/W	000: PD5 I/O (port) 001: D5 I/O (BSC)*[2] 010: TIC5US input (MTU2S) Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PD4MD2	0	R/W	PD4 Mode
1	PD4MD1	0	R/W	Select the function of the PD4/D4/TIC5W pin.
0	PD4MD0	0*[1]	R/W	000: PD4 I/O (port) 001: D4 I/O (BSC)*[2] 010: TIC5W input (MTU2) Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port D Control Register L1 (PDCRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PD3 MD2	PD3 MD1	PD3 MD0	-	PD2 MD2	PD2 MD1	PD2 MD0	-	PD1 MD2	PD1 MD1	PD1 MD0	-	PD0 MD2	PD0 MD1	PD0 MD0
Initial value:	0	0	0	0*	0	0	0	0*	0	0	0	0*	0	0	0	0*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Note: * The initial value is 1 in the on-chip ROM disabled external-extension mode.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PD3MD2	0	R/W	PD3 Mode
13	PD3MD1	0	R/W	Select the function of the PD3/D3/TIC5V pin.
12	PD3MD0	0*[1]	R/W	000: PD3 I/O (port) 001: D3 I/O (BSC)*[2] 010: TIC5V input (MTU2) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	PD2MD2	0	R/W	PD2 Mode
9	PD2MD1	0	R/W	Select the function of the PD2/D2/TIC5U pin.
8	PD2MD0	0*[1]	R/W	000: PD2 I/O (port) 001: D2 I/O (BSC)*[2] 010: TIC5U input (MTU2) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PD1MD2	0	R/W	PD1 Mode
5	PD1MD1	0	R/W	Select the function of the PD1/D1 pin.
4	PD1MD0	0*[1]	R/W	000: PD1 I/O (port) 001: D1 I/O (BSC)*[2] Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PD0MD2	0	R/W	PD0 Mode
1	PD0MD1	0	R/W	Select the function of the PD0/D0 pin.
0	PD0MD0	0*[1]	R/W	000: PD0 I/O (port) 001: D0 I/O (BSC)*[2] Other than above: Setting prohibited

Notes: 1. The initial value is 1 in the on-chip ROM disabled external-extension mode.
2. This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

21.1.9 Port E I/O Registers L, H (PEIORL, PEIORH)

PEIORL and PEIORH are 16-bit readable/writable registers that are used to set the pins on port E as inputs or outputs. PE21IOR to PE0IOR correspond to pins PE21 to PE0 (names of multiplexed pins are here given as port names and pin numbers alone). PEIORL is enabled when the port E pins are functioning as general-purpose inputs/outputs (PE15 to PE0) and the TIOC pin is functioning as inputs/outputs of MTU2. In other states, PEIORL is disabled. PEIORH is enabled when the port E pins are functioning as general-purpose inputs/outputs (PE21 to PE16), and the TIOC pin is functioning as inputs/outputs of MTU2S. In other states, PEIORH is disabled.

A given pin on port E will be an output pin if the corresponding bit in PEIORH or PEIORL is set to 1, and an input pin if the bit is cleared to 0.

However, bits 11, 9, and 5 of PEIORH and PEIORL are disabled in SH7083. PEIORH is disabled in SH7084/SH7085. Bits 15 to 6 of PEIORH are reserved. These bits are always read as 0. The write value should always be 0.

The initial values of PEIORL and PEIORH are H'0000, respectively.

- Port E I/O Register H (PEIORH)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	PE21 IOR	PE20 IOR	PE19 IOR	PE18 IOR	PE17 IOR	PE16 IOR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

- Port E I/O Register L (PEIORL)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15 IOR	PE14 IOR	PE13 IOR	PE12 IOR	PE11 IOR	PE10 IOR	PE9 IOR	PE8 IOR	PE7 IOR	PE6 IOR	PE5 IOR	PE4 IOR	PE3 IOR	PE2 IOR	PE1 IOR	PE0 IOR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

21.1.10 Port E Control Registers L1 to L4, H1, H2 (PECRL1 to PECRL4, PECRH1, PECRH2)

PECRL1 to PECRL4, PECRH1 and PECRH2 are 16-bit readable/writable registers that are used to select the functions of the multiplexed pins on port E.

SH7083:

- Port E Control Registers H2 and H1 (PECRH2 and PECRH1)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- Port E Control Register L4 (PECRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE15 MD2	PE15 MD1	PE15 MD0	-	PE14 MD2	PE14 MD1	PE14 MD0	-	-	PE13 MD1	PE13 MD0	-	PE12 MD2	PE12 MD1	PE12 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PE15MD2	0	R/W	PE15 Mode
13	PE15MD1	0	R/W	Select the function of the PE15/CKE/DACK1/TIOC4D/$\overline{\text{IRQOUT}}$ pin.
12	PE15MD0	0	R/W	000: PE15 I/O (port)
				001: TIOC4D I/O (MTU2)
				010: DACK1 output (DMAC)*
				011: $\overline{\text{IRQOUT}}$ output (INTC)
				101: CKE output (BSC)*
				Other than above: Setting prohibited
11	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
10	PE14MD2	0	R/W	PE14 Mode
9	PE14MD1	0	R/W	Select the function of the PE14/DACK0/TIOC4C pin.
8	PE14MD0	0	R/W	000: PE14 I/O (port)
				001: TIOC4C I/O (MTU2)
				010: DACK0 output (DMAC)*
				Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved
				These bits are always read as 0. The write value should always be 0.
5	PE13MD1	0	R/W	PE13 Mode
4	PE13MD0	0	R/W	Select the function of the PE13/TIOC4B/$\overline{\text{MRES}}$/$\overline{\text{ASEBRKAK}}$/$\overline{\text{ASEBRK}}$ pin. Fixed to $\overline{\text{ASEBRKAK}}$ output/$\overline{\text{ASEBRK}}$ input when using E10A (in $\overline{\text{ASEMD0}}$ = low).
				00: PE13 I/O (port)
				01: TIOC4B I/O (MTU2)
				10: $\overline{\text{MRES}}$ input (INTC)
				Other than above: Setting prohibited
3	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PE12MD2	0	R/W	PE12 Mode
1	PE12MD1	0	R/W	Select the function of the PE12/TIOC4A/TXD3/$\overline{SCS}$
0	PE12MD0	0	R/W	pin.
				000: PE12 I/O (port)
				001: TIOC4A I/O (MTU2)
				011: TXD3 output (SCIF)
				101: $\overline{SCS}$ I/O (SSU)
				Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port E Control Register L3 (PECRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	PE10 MD2	PE10 MD1	PE10 MD0	-	-	-	-	-	PE8 MD2	PE8 MD1	PE8 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved
				These bits are always read as 0. The write value should always be 0.
10	PE10MD2	0	R/W	PE10 Mode
9	PE10MD1	0	R/W	Select the function of the PE10/TIOC3C/TXD2/SSO
8	PE10MD0	0	R/W	pin.
				000: PE10 I/O (port)
				001: TIOC3C I/O (MTU2)
				010: TXD2 output (SCI)
				101: SSO I/O (SSU)
				Other than above: Setting prohibited
7 to 3	—	All 0	R	Reserved
				These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PE8MD2	0	R/W	PE8 Mode
1	PE8MD1	0	R/W	Select the function of the PE8/TIOC3A/SCK2/SSCK pin.
0	PE8MD0	0	R/W	000: PE8 I/O (port) 001: TIOC3A I/O (MTU2) 010: SCK2 I/O (SCI) 101: SSCK I/O (SSU) Other than above: Setting prohibited

- Port E Control Register L2 (PECRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE7 MD2	PE7 MD1	PE7 MD0	-	PE6 MD2	PE6 MD1	PE6 MD0	-	-	-	-	-	PE4 MD2	PE4 MD1	PE4 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PE7MD2	0	R/W	PE7 Mode
13	PE7MD1	0	R/W	Select the function of the PE7/$\overline{BS}$/TIOC2B/$\overline{UBCTRG}$/RXD2/SSI pin.
12	PE7MD0	0	R/W	000: PE7 I/O (port) 001: TIOC2B I/O (MTU2) 010: RXD2 input (SCI) 011: $\overline{BS}$ output (BSC)* 101: SSI I/O (SSU) 111: $\overline{UBCTRG}$ output (UBC) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	PE6MD2	0	R/W	PE6 Mode
9	PE6MD1	0	R/W	Select the function of the PE6/$\overline{CS7}$/TIOC2A/SCK3 pin.
8	PE6MD0	0	R/W	000: PE6 I/O (port)
				001: TIOC2A I/O (MTU2)
				010: SCK3 I/O (SCIF)
				101: $\overline{CS7}$ output (BSC)*
				Other than above: Setting prohibited
7 to 3	—	All 0	R	Reserved
				These bits are always read as 0. The write value should always be 0.
2	PE4MD2	0	R/W	PE4 Mode
1	PE4MD1	0	R/W	Select the function of the PE4/TIOC1A/RXD3/TCK pin. Fixed to TCK input when using E10A (in $\overline{ASEMD0}$ = low).
0	PE4MD0	0	R/W	000: PE4 I/O (port)
				001: TIOC1A I/O (MTU2)
				010: RXD3 input (SCIF)
				Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port E Control Register L1 (PECRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE3 MD2	PE3 MD1	PE3 MD0	-	PE2 MD2	PE2 MD1	PE2 MD0	-	PE1 MD2	PE1 MD1	PE1 MD0	-	-	PE0 MD1	PE0 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PE3MD2	0	R/W	PE3 Mode
13	PE3MD1	0	R/W	Select the function of the PE3/TEND1/TIOC0D/TDO
12	PE3MD0	0	R/W	pin. Fixed to TDO output when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE3 I/O (port) 001: TIOC0D I/O (MTU2) 010: TEND1 output (DMAC)* Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PE2MD2	0	R/W	PE2 Mode
9	PE2MD1	0	R/W	Select the function of the PE2/DREQ1/TIOC0C/TDI
8	PE2MD0	0	R/W	pin. Fixed to TDI input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE2 I/O (port) 001: TIOC0C I/O (MTU2) 010: DREQ1 input (DMAC) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PE1MD2	0	R/W	PE1 Mode
5	PE1MD1	0	R/W	Select the function of the PE1/TEND0/TIOC0B/$\overline{\text{TRST}}$
4	PE1MD0	0	R/W	pin. Fixed to $\overline{\text{TRST}}$ input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE1 I/O (port) 001: TIOC0B I/O (MTU2) 010: TEND0 output (DMAC)* Other than above: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	PE0MD1	0	R/W	PE0 Mode
0	PE0MD0	0	R/W	Select the function of the PE0/DREQ0/TIOC0A/TMS pin. Fixed to TMS input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 00: PE0 I/O (port) 01: TIOC0A I/O (MTU2) 10: DREQ0 input (DMAC) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

SH7084:

- Port E Control Registers H2 and H1 (PECRH2 and PECRH1)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- Port E Control Register L4 (PECRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE15 MD2	PE15 MD1	PE15 MD0	-	PE14 MD2	PE14 MD1	PE14 MD0	-	-	PE13 MD1	PE13 MD0	-	PE12 MD2	PE12 MD1	PE12 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PE15MD2	0	R/W	PE15 Mode
13	PE15MD1	0	R/W	Select the function of the PE15/CKE/DACK1/TIOC4D/$\overline{\text{IRQOUT}}$ pin.
12	PE15MD0	0	R/W	000: PE15 I/O (port) 001: TIOC4D I/O (MTU2) 010: DACK1 output (DMAC)* 011: $\overline{\text{IRQOUT}}$ output (INTC) 101: CKE output (BSC)* Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PE14MD2	0	R/W	PE14 Mode
9	PE14MD1	0	R/W	Select the function of the PE14/$\overline{\text{AH}}$/DACK0/TIOC4C pin.
8	PE14MD0	0	R/W	000: PE14 I/O (port) 001: TIOC4C I/O (MTU2) 010: DACK0 output (DMAC)* 101: $\overline{\text{AH}}$ output (BSC)* Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PE13MD1	0	R/W	PE13 Mode
4	PE13MD0	0	R/W	Select the function of the PE13/TIOC4B/$\overline{\text{MRES}}$ pin. 00: PE13 I/O (port) 01: TIOC4B I/O (MTU2) 10: $\overline{\text{MRES}}$ input (INTC) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PE12MD2	0	R/W	PE12 Mode
1	PE12MD1	0	R/W	Select the function of the PE12/TIOC4A/TXD3/$\overline{SCS}$
0	PE12MD0	0	R/W	pin.
				000: PE12 I/O (port)
				001: TIOC4A I/O (MTU2)
				011: TXD3 output (SCIF)
				101: $\overline{SCS}$ I/O (SSU)
				Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port E Control Register L3 (PECRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE11 MD2	PE11 MD1	PE11 MD0	-	PE10 MD2	PE10 MD1	PE10 MD0	-	PE9 MD2	PE9 MD1	PE9 MD0	-	PE8 MD2	PE8 MD1	PE8 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
14	PE11MD2	0	R/W	PE11 Mode
13	PE11MD1	0	R/W	Select the function of the PE11/TIOC3D/RXD3/$\overline{CTS3}$
12	PE11MD0	0	R/W	pin.
				000: PE11 I/O (port)
				001: TIOC3D I/O (MTU2)
				011: RXD3 input (SCIF)
				100: $\overline{CTS3}$ input (SCIF)
				Other than above: Setting prohibited
11	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	PE10MD2	0	R/W	PE10 Mode
9	PE10MD1	0	R/W	Select the function of the PE10/TIOC3C/TXD2/SSO
8	PE10MD0	0	R/W	pin. 000: PE10 I/O (port) 001: TIOC3C I/O (MTU2) 010: TXD2 output (SCI) 101: SSO I/O (SSU) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PE9MD2	0	R/W	PE9 Mode
5	PE9MD1	0	R/W	Select the function of the PE9/TIOC3B/SCK3/$\overline{\text{RTS3}}$
4	PE9MD0	0	R/W	pin. 000: PE9 I/O (port) 001: TIOC3B I/O (MTU2) 011: SCK3 I/O (SCIF) 100: $\overline{\text{RTS3}}$ output (SCIF) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PE8MD2	0	R/W	PE8 Mode
1	PE8MD1	0	R/W	Select the function of the PE8/TIOC3A/SCK2/SSCK
0	PE8MD0	0	R/W	pin. 000: PE8 I/O (port) 001: TIOC3A I/O (MTU2) 010: SCK2 I/O (SCI) 101: SSCK I/O (SSU) Other than above: Setting prohibited

- Port E Control Register L2 (PECRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE7 MD2	PE7 MD1	PE7 MD0	-	PE6 MD2	PE6 MD1	PE6 MD0	-	PE5 MD2	PE5 MD1	PE5 MD0	-	PE4 MD2	PE4 MD1	PE4 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PE7MD2	0	R/W	PE7 Mode
13	PE7MD1	0	R/W	Select the function of the PE7/$\overline{BS}$/TIOC2B/$\overline{UBCTRG}$/RXD2/SSI pin.
12	PE7MD0	0	R/W	000: PE7 I/O (port) 001: TIOC2B I/O (MTU2) 010: RXD2 input (SCI) 011: $\overline{BS}$ output (BSC)* 101: SSI I/O (SSU) 111: $\overline{UBCTRG}$ output (UBC) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PE6MD2	0	R/W	PE6 Mode
9	PE6MD1	0	R/W	Select the function of the PE6/$\overline{CS7}$/TIOC2A/SCK3 pin.
8	PE6MD0	0	R/W	000: PE6 I/O (port) 001: TIOC2A I/O (MTU2) 010: SCK3 I/O (SCIF) 101: $\overline{CS7}$ output (BSC)* Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PE5MD2	0	R/W	PE5 Mode
5	PE5MD1	0	R/W	Select the function of the
4	PE5MD0	0	R/W	PE5/$\overline{CS6}$/TIOC1B/TXD3/$\overline{ASEBRKAK}$/$\overline{ASEBRK}$ pin. Fixed to $\overline{ASEBRKAK}$ output/$\overline{ASEBRK}$ input when using E10A (in $\overline{ASEMD0}$ = low). 000: PE5 I/O (port) 001: TIOC1B I/O (MTU2) 010: TXD3 output (SCIF) 101: $\overline{CS6}$ output (BSC)* Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PE4MD2	0	R/W	PE4 Mode
1	PE4MD1	0	R/W	Select the function of the PE4/TIOC1A/RXD3/TCK
0	PE4MD0	0	R/W	pin. Fixed to TCK input when using E10A (in $\overline{ASEMD0}$ = low). 000: PE4 I/O (port) 001: TIOC1A I/O (MTU2) 010: RXD3 input (SCIF) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port E Control Register L1 (PECRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE3 MD2	PE3 MD1	PE3 MD0	-	PE2 MD2	PE2 MD1	PE2 MD0	-	PE1 MD2	PE1 MD1	PE1 MD0	-	-	PE0 MD1	PE0 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PE3MD2	0	R/W	PE3 Mode
13	PE3MD1	0	R/W	Select the function of the PE3/TEND1/TIOC0D/TDO
12	PE3MD0	0	R/W	pin. Fixed to TDO output when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE3 I/O (port) 001: TIOC0D I/O (MTU2) 010: TEND1 output (DMAC)* Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PE2MD2	0	R/W	PE2 Mode
9	PE2MD1	0	R/W	Select the function of the PE2/DREQ1/TIOC0C/TDI
8	PE2MD0	0	R/W	pin. Fixed to TDI input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE2 I/O (port) 001: TIOC0C I/O (MTU2) 010: DREQ1 input (DMAC) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PE1MD2	0	R/W	PE1 Mode
5	PE1MD1	0	R/W	Select the function of the PE1/TEND0/TIOC0B/$\overline{\text{TRST}}$
4	PE1MD0	0	R/W	pin. Fixed to $\overline{\text{TRST}}$ input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE1 I/O (port) 001: TIOC0B I/O (MTU2) 010: TEND0 output (DMAC)* Other than above: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	PE0MD1	0	R/W	PE0 Mode
0	PE0MD0	0	R/W	Select the function of the PE0/DREQ0/TIOC0A/TMS pin. Fixed to TMS input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 00: PE0 I/O (port) 01: TIOC0A I/O (MTU2) 10: DREQ0 input (DMAC) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

SH7085:

- Port E Control Registers H2 and H1 (PECRH2 and PECRH1)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- Port E Control Register L4 (PECRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE15 MD2	PE15 MD1	PE15 MD0	-	PE14 MD2	PE14 MD1	PE14 MD0	-	-	PE13 MD1	PE13 MD0	-	PE12 MD2	PE12 MD1	PE12 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PE15MD2	0	R/W	PE15 Mode
13	PE15MD1	0	R/W	Select the function of the PE15/CKE/DACK1/TIOC4D/$\overline{\text{IRQOUT}}$ pin.
12	PE15MD0	0	R/W	000: PE15 I/O (port)
				001: TIOC4D I/O (MTU2)
				010: DACK1 output (DMAC)*
				011: $\overline{\text{IRQOUT}}$ output (INTC)
				101: CKE output (BSC)*
				Other than above: Setting prohibited
11	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
10	PE14MD2	0	R/W	PE14 Mode
9	PE14MD1	0	R/W	Select the function of the
8	PE14MD0	0	R/W	PE14/$\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/DQMUU/DACK0/TIOC4C pin.
				000: PE14 I/O (port)
				001: TIOC4C I/O (MTU2)
				010: DACK0 output (DMAC)*
				101: $\overline{\text{WRHH}}$/$\overline{\text{ICIOWR}}$/$\overline{\text{AH}}$/DQMUU output (BSC)*
				Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved
				These bits are always read as 0. The write value should always be 0.
5	PE13MD1	0	R/W	PE13 Mode
4	PE13MD0	0	R/W	Select the function of the PE13/TIOC4B/$\overline{\text{MRES}}$/$\overline{\text{ASEBRKAK}}$/$\overline{\text{ASEBRK}}$ pin. Fixed to $\overline{\text{ASEBRKAK}}$ output/$\overline{\text{ASEBRK}}$ input when using E10A (in $\overline{\text{ASEMD0}}$ = low).
				00: PE13 I/O (port)
				01: TIOC4B I/O (MTU2)
				10: $\overline{\text{MRES}}$ input (INTC)
				Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PE12MD2	0	R/W	PE12 Mode
1	PE12MD1	0	R/W	Select the function of the
0	PE12MD0	0	R/W	PE12/TIOC4A/TXD3/$\overline{SCS}$/TCK pin. Fixed to TCK input when using E10A (in $\overline{ASEMD0}$ = low). 000: PE12 I/O (port) 001: TIOC4A I/O (MTU2) 011: TXD3 output (SCIF) 101: $\overline{SCS}$ I/O (SSU) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port E Control Register L3 (PECRL3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE11 MD2	PE11 MD1	PE11 MD0	-	PE10 MD2	PE10 MD1	PE10 MD0	-	PE9 MD2	PE9 MD1	PE9 MD0	-	PE8 MD2	PE8 MD1	PE8 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PE11MD2	0	R/W	PE11 Mode
13	PE11MD1	0	R/W	Select the function of the
12	PE11MD0	0	R/W	PE11/TIOC3D/RXD3/$\overline{CTS3}$/TDO pin. Fixed to TDO output when using E10A (in $\overline{ASEMD0}$ = low). 000: PE11 I/O (port) 001: TIOC3D I/O (MTU2) 011: RXD3 input (SCIF) 100: $\overline{CTS3}$ input (SCIF) Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PE10MD2	0	R/W	PE10 Mode
9	PE10MD1	0	R/W	Select the function of the
8	PE10MD0	0	R/W	PE10/TIOC3C/TXD2/SSO/TDI pin. Fixed to TDI input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE10 I/O (port) 001: TIOC3C I/O (MTU2) 010: TXD2 output (SCI) 101: SSO I/O (SSU) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PE9MD2	0	R/W	PE9 Mode
5	PE9MD1	0	R/W	Select the function of the
4	PE9MD0	0	R/W	PE9/TIOC3B/SCK3/$\overline{\text{RTS3}}$/$\overline{\text{TRST}}$ pin. Fixed to $\overline{\text{TRST}}$ input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE9 I/O (port) 001: TIOC3B I/O (MTU2) 011: SCK3 I/O (SCIF) 100: $\overline{\text{RTS3}}$ output (SCIF) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PE8MD2	0	R/W	PE8 Mode
1	PE8MD1	0	R/W	Select the function of the
0	PE8MD0	0	R/W	PE8/TIOC3A/SCK2/SSCK/TMS pin. Fixed to TMS input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE8 I/O (port) 001: TIOC3A I/O (MTU2) 010: SCK2 I/O (SCI) 101: SSCK I/O (SSU) Other than above: Setting prohibited

- Port E Control Register L2 (PECRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE7 MD2	PE7 MD1	PE7 MD0	-	PE6 MD2	PE6 MD1	PE6 MD0	-	PE5 MD2	PE5 MD1	PE5 MD0	-	PE4 MD2	PE4 MD1	PE4 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PE7MD2	0	R/W	PE7 Mode
13	PE7MD1	0	R/W	Select the function of the
12	PE7MD0	0	R/W	PE7/$\overline{\text{BS}}$/TIOC2B/$\overline{\text{UBCTRG}}$/RXD2/SSI pin. 000: PE7 I/O (port) 001: TIOC2B I/O (MTU2) 010: RXD2 input (SCI) 011: $\overline{\text{BS}}$ output (BSC)* 101: SSI I/O (SSU) 111: $\overline{\text{UBCTRG}}$ output (UBC) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	PE6MD2	0	R/W	PE6 Mode
9	PE6MD1	0	R/W	Select the function of the
8	PE6MD0	0	R/W	PE6/$\overline{CS7}$/TIOC2A/SCK3/AUDATA0 pin. Fixed to AUDATA0 output when using the AUD function of the E10A. 000: PE6 I/O (port) 001: TIOC2A I/O (MTU2) 010: SCK3 I/O (SCIF) 101: $\overline{CS7}$ output (BSC)* Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PE5MD2	0	R/W	PE5 Mode
5	PE5MD1	0	R/W	Select the function of the
4	PE5MD0	0	R/W	PE5/$\overline{CS6}$/$\overline{CE1B}$/TIOC1B/TXD3/AUDATA1 pin. Fixed to AUDATA1 output when using the AUD function of the E10A. 000: PE5 I/O (port) 001: TIOC1B I/O (MTU2) 010: TXD3 output (SCIF) 101: $\overline{CS6}$/$\overline{CE1B}$ output (BSC)* Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PE4MD2	0	R/W	PE4 Mode
1	PE4MD1	0	R/W	Select the function of the
0	PE4MD0	0	R/W	PE4/$\overline{\text{IOIS16}}$/TIOC1A/RXD3/AUDATA2 pin. Fixed to AUDATA2 output when using the AUD function of the E10A. 000: PE4 I/O (port) 001: TIOC1A I/O (MTU2) 010: RXD3 input (SCIF) 101: $\overline{\text{IOIS16}}$ input (BSC)* Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port E Control Register L1 (PECRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE3 MD2	PE3 MD1	PE3 MD0	-	PE2 MD2	PE2 MD1	PE2 MD0	-	PE1 MD2	PE1 MD1	PE1 MD0	-	-	PE0 MD1	PE0 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PE3MD2	0	R/W	PE3 Mode
13	PE3MD1	0	R/W	Select the function of the
12	PE3MD0	0	R/W	PE3/TEND1/TIOC0D/AUDATA3 pin. Fixed to AUDATA3 output when using the AUD function of the E10A. 000: PE3 I/O (port) 001: TIOC0D I/O (MTU2) 010: TEND1 output (DMAC)* Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	PE2MD2	0	R/W	PE2 Mode
9	PE2MD1	0	R/W	Select the function of the PE2/DREQ1/TIOC0C pin.
8	PE2MD0	0	R/W	000: PE2 I/O (port) 001: TIOC0C I/O (MTU2) 010: DREQ1 input (DMAC) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PE1MD2	0	R/W	PE1 Mode
5	PE1MD1	0	R/W	Select the function of the PE1/TEND0/TIOC0B pin.
4	PE1MD0	0	R/W	000: PE1 I/O (port) 001: TIOC0B I/O (MTU2) 010: TEND0 output (DMAC)* Other than above: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PE0MD1	0	R/W	PE0 Mode
0	PE0MD0	0	R/W	Select the function of the PE0/DREQ0/TIOC0A/AUDCK pin. Fixed to AUDCK output when using the AUD function of the E10A. 00: PE0 I/O (port) 01: TIOC0A I/O (MTU2) 10: DREQ0 input (DMAC) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

SH7086:

- Port E Control Register H2 (PECRH2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	PE21 MD1	PE21 MD0	-	-	PE20 MD1	PE20 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PE21MD1	0	R/W	PE21 Mode
4	PE21MD0	0	R/W	Select the function of the PE21/TIOC4DS pin. 00: PE21 I/O (port) 01: TIOC4DS I/O (MTU2S) Other than above: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PE20MD1	0	R/W	PE20 Mode
0	PE20MD0	0	R/W	Select the function of the PE20/TIOC4CS pin. 00: PE20 I/O (port) 01: TIOC4CS I/O (MTU2S) Other than above: Setting prohibited

- Port E Control Register H1 (PECRH1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PE19 MD1	PE19 MD0	-	-	PE18 MD1	PE18 MD0	-	-	PE17 MD1	PE17 MD0	-	PE16 MD2	PE16 MD1	PE16 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PE19MD1	0	R/W	PE19 Mode
12	PE19MD0	0	R/W	Select the function of the PE19/TIOC4BS pin. 00: PE19 I/O (port) 01: TIOC4BS I/O (MTU2S) Other than above: Setting prohibited
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PE18MD1	0	R/W	PE18 Mode
8	PE18MD0	0	R/W	Select the function of the PE18/TIOC4AS pin. 00: PE18 I/O (port) 01: TIOC4AS I/O (MTU2S) Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PE17MD1	0	R/W	PE17 Mode
4	PE17MD0	0	R/W	Select the function of the PE17/TIOC3DS pin. 00: PE17 I/O (port) 01: TIOC3DS I/O (MTU2S) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PE16MD2	0	R/W	PE16 Mode
1	PE16MD1	0	R/W	Select the function of the PE16/$\overline{CS8}$/TIOC3BS pin.
0	PE16MD0	0	R/W	000: PE16 I/O (port)
				001: TIOC3BS I/O (MTU2S)
				101: $\overline{CS8}$ output (BSC)*
				Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port E Control Register L4 (PECRL4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE15 MD2	PE15 MD1	PE15 MD0	-	PE14 MD2	PE14 MD1	PE14 MD0	-	-	PE13 MD1	PE13 MD0	-	PE12 MD2	PE12 MD1	PE12 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.
14	PE15MD2	0	R/W	PE15 Mode
13	PE15MD1	0	R/W	Select the function of the
12	PE15MD0	0	R/W	PE15/CKE/DACK1/TIOC4D/$\overline{IRQOUT}$ pin.
				000: PE15 I/O (port)
				001: TIOC4D I/O (MTU2)
				010: DACK1 output (DMAC)*
				011: $\overline{IRQOUT}$ output (INTC)
				101: CKE output (BSC)*
				Other than above: Setting prohibited
11	—	0	R	Reserved
				This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	PE14MD2	0	R/W	PE14 Mode
9	PE14MD1	0	R/W	Select the function of the
8	PE14MD0	0	R/W	PE14/WRHH/ICIOWR/AH/DQMUU/DACK0/TIOC4C pin. 000: PE14 I/O (port) 001: TIOC4C I/O (MTU2) 010: DACK0 output (DMAC)* 101: WRHH/ICIOWR/AH/DQMUU output (BSC)* Other than above: Setting prohibited
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PE13MD1	0	R/W	PE13 Mode
4	PE13MD0	0	R/W	Select the function of the PE13/TIOC4B/MRES/ASEBRKAK/ASEBRK pin. Fixed to ASEBRKAK output/ASEBRK input when using E10A (in ASEMD0 = low). 00: PE13 I/O (port) 01: TIOC4B I/O (MTU2) 10: MRES input (INTC) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PE12MD2	0	R/W	PE12 Mode
1	PE12MD1	0	R/W	Select the function of the
0	PE12MD0	0	R/W	PE12/TIOC4A/TXD3/SCS/TCK pin. Fixed to TCK input when using E10A (in ASEMD0 = low). 000: PE12 I/O (port) 001: TIOC4A I/O (MTU2) 011: TXD3 output (SCIF) 101: SCS I/O (SSU) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port E Control Register L3 (PECRL3)
-

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE11 MD2	PE11 MD1	PE11 MD0	-	PE10 MD2	PE10 MD1	PE10 MD0	-	PE9 MD2	PE9 MD1	PE9 MD0	-	PE8 MD2	PE8 MD1	PE8 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PE11MD2	0	R/W	PE11 Mode
13	PE11MD1	0	R/W	Select the function of the
12	PE11MD0	0	R/W	PE11/TIOC3D/RXD3/$\overline{\text{CTS3}}$/TDO pin. Fixed to TDO output when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE11 I/O (port) 001: TIOC3D I/O (MTU2) 011: RXD3 input (SCIF) 100: $\overline{\text{CTS3}}$ input (SCIF) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PE10MD2	0	R/W	PE10 Mode
9	PE10MD1	0	R/W	Select the function of the
8	PE10MD0	0	R/W	PE10/TIOC3C/TXD2/SSO/TDI pin. Fixed to TDI input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE10 I/O (port) 001: TIOC3C I/O (MTU2) 010: TXD2 output (SCI) 101: SSO I/O (SSU) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PE9MD2	0	R/W	PE9 Mode
5	PE9MD1	0	R/W	Select the function of the
4	PE9MD0	0	R/W	PE9/TIOC3B/SCK3/$\overline{\text{RTS3}}$/$\overline{\text{TRST}}$ pin. Fixed to $\overline{\text{TRST}}$ input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE9 I/O (port) 001: TIOC3B I/O (MTU2) 011: SCK3 I/O (SCIF) 100: $\overline{\text{RTS3}}$ output (SCIF) Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PE8MD2	0	R/W	PE8 Mode
1	PE8MD1	0	R/W	Select the function of the
0	PE8MD0	0	R/W	PE8/TIOC3A/SCK2/SSCK/TMS pin. Fixed to TMS input when using E10A (in $\overline{\text{ASEMD0}}$ = low). 000: PE8 I/O (port) 001: TIOC3A I/O (MTU2) 010: SCK2 I/O (SCI) 101: SSCK I/O (SSU) Other than above: Setting prohibited

- Port E Control Register L2 (PECRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE7 MD2	PE7 MD1	PE7 MD0	-	PE6 MD2	PE6 MD1	PE6 MD0	-	PE5 MD2	PE5 MD1	PE5 MD0	-	PE4 MD2	PE4 MD1	PE4 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	PE7MD2	0	R/W	PE7 Mode
13	PE7MD1	0	R/W	Select the function of the PE7/$\overline{BS}$/TIOC2B/$\overline{UBCTRG}$/RXD2/SSI pin.
12	PE7MD0	0	R/W	000: PE7 I/O (port) 001: TIOC2B I/O (MTU2) 010: RXD2 input (SCI) 011: $\overline{BS}$ output (BSC)* 101: SSI I/O (SSU) 111: $\overline{UBCTRG}$ output (UBC) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PE6MD2	0	R/W	PE6 Mode
9	PE6MD1	0	R/W	Select the function of the PE6/$\overline{CS7}$/TIOC2A/SCK3/AUDATA0 pin. Fixed to AUDATA0 output when using the AUD function of the E10A.
8	PE6MD0	0	R/W	000: PE6 I/O (port) 001: TIOC2A I/O (MTU2) 010: SCK3 I/O (SCIF) 101: $\overline{CS7}$ output (BSC)* Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PE5MD2	0	R/W	PE5 Mode
5	PE5MD1	0	R/W	Select the function of the
4	PE5MD0	0	R/W	PE5/$\overline{\text{CS6}}$/$\overline{\text{CE1B}}$/TIOC1B/TXD3/AUDATA1 pin. Fixed to AUDATA1 output when using the AUD function of the E10A. 000: PE5 I/O (port) 001: TIOC1B I/O (MTU2) 010: TXD3 output (SCIF) 101: $\overline{\text{CS6}}$/$\overline{\text{CE1B}}$ output (BSC)* Other than above: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PE4MD2	0	R/W	PE4 Mode
1	PE4MD1	0	R/W	Select the function of the
0	PE4MD0	0	R/W	PE4/$\overline{\text{IOIS16}}$/TIOC1A/RXD3/AUDATA2 pin. Fixed to AUDATA2 output when using the AUD function of the E10A. 000: PE4 I/O (port) 001: TIOC1A I/O (MTU2) 010: RXD3 input (SCIF) 101: $\overline{\text{IOIS16}}$ input (BSC)* Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

- Port E Control Register L1 (PECRL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE3 MD2	PE3 MD1	PE3 MD0	-	PE2 MD2	PE2 MD1	PE2 MD0	-	PE1 MD2	PE1 MD1	PE1 MD0	-	-	PE0 MD1	PE0 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PE3MD2	0	R/W	PE3 Mode
13	PE3MD1	0	R/W	Select the function of the
12	PE3MD0	0	R/W	PE3/TEND1/TIOC0D/AUDATA3 pin. Fixed to AUDATA3 output when using the AUD function of the E10A. 000: PE3 I/O (port) 001: TIOC0D I/O (MTU2) 010: TEND1 output (DMAC)* Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PE2MD2	0	R/W	PE2 Mode
9	PE2MD1	0	R/W	Select the function of the PE2/DREQ1/TIOC0C pin.
8	PE2MD0	0	R/W	000: PE2 I/O (port) 001: TIOC0C I/O (MTU2) 010: DREQ1 input (DMAC) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	PE1MD2	0	R/W	PE1 Mode
5	PE1MD1	0	R/W	Select the function of the PE1/TEND0/TIOC0B pin.
4	PE1MD0	0	R/W	000: PE1 I/O (port) 001: TIOC0B I/O (MTU2) 010: TEND0 output (DMAC)* Other than above: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PE0MD1	0	R/W	PE0 Mode
0	PE0MD0	0	R/W	Select the function of the PE0/DREQ0/TIOC0A/AUDCK pin. Fixed to AUDCK output when using the AUD function of the E10A. 00: PE0 I/O (port) 01: TIOC0A I/O (MTU2) 10: DREQ0 input (DMAC) Other than above: Setting prohibited

Note: * This function is enabled only in the on-chip ROM enabled/disabled external-extension mode. Do not set 1 in single-chip mode.

21.1.11 High-Current Port Control Register (HCPCR)

HCPCR is a 16-bit readable/writable register that is used to control the high-current ports (10 pins (PD9, PD11 to PD15, and PE12 to PE15) in the SH7083, 12 pins (PD9, PD11 to PD15, PE9, and PE11 to PE15) in the SH7084, 18 pins (PD9, PD11 to PD15, PD24 to PD29, PE9, and PE11 to PE15) in the SH7085, and 24 pins (PD9, PD11 to PD15, PD24 to PD29, PE9, and PE11 to PE21) in the SH7086).

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	MZI ZDH	MZI ZDL	MZI ZEH	MZI ZEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	MZIZDH	1	R/W	Port D High-Current Port High-Impedance H Selects whether or not the high-current ports PD24 to PD29 are set to high-impedance regardless of the PFC setting when detecting oscillation halt or in software standby mode. 0: Set to high-impedance 1: Not set to high-impedance If this bit is set to 1, the pin state is retained when detecting oscillation halt. For the pin state in software standby mode, refer to appendix A, Pin States.
2	MZIZDL	1	R/W	Port D High-Current Port High-Impedance L Selects whether or not the high-current ports PD9 and PD11 to PD15 are set to high-impedance regardless of the PFC setting when detecting oscillation halt or in software standby mode. 0: Set to high-impedance 1: Not set to high-impedance If this bit is set to 1, the pin state is retained when detecting oscillation halt. For the pin state in software standby mode, refer to appendix A, Pin States.

Bit	Bit Name	Initial Value	R/W	Description
1	MZIZEH	1	R/W	Port E High-Current Port High-Impedance H Selects whether or not the high-current ports PE16 to PE21 are set to high-impedance regardless of the PFC setting when detecting oscillation halt or in software standby mode. 0: Set to high-impedance 1: Not set to high-impedance If this bit is set to 1, the pin state is retained when detecting oscillation halt. For the pin state in software standby mode, refer to appendix A, Pin States.
0	MZIZEL	1	R/W	Port E High-Current Port High-Impedance L Selects whether or not the high-current ports PE9 and PE11 to PE15 are set to high-impedance regardless of the PFC setting when detecting oscillation halt or in software standby mode. 0: Set to high-impedance 1: Not set to high-impedance If this bit is set to 1, the pin state is retained when detecting oscillation halt. For the pin state in software standby mode, refer to appendix A, Pin States.

21.1.12 IRQOUT Function Control Register (IFCR)

IFCR is a 16-bit readable/writable register that is used to control the IRQOUT pin output when it is selected as the multiplexed pin function by port D control register H4 (PDCRH4) and port E control register L4 (PECRL4). When PDCRH4 or PECRL4 selects another function, the IFCR setting does not affect the pin function.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	IRQ MD3	IRQ MD2	IRQ MD1	IRQ MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	IRQMD3	0	R/W	Port D $\overline{\text{IRQOUT}}$ Pin Function Select
2	IRQMD2	0	R/W	Select the $\overline{\text{IRQOUT}}$ pin function when bits 9 and 8 (PD30MD1 and PD30MD0) in PDCRH4 are set to B'10. 00: Interrupt request accept signal output 01: Refresh signal output 10: Interrupt request accept signal output or refresh signal output (depends on the operating state) 11: Always high-level output
1	IRQMD1	0	R/W	Port E $\overline{\text{IRQOUT}}$ Pin Function Select
0	IRQMD0	0	R/W	Select the $\overline{\text{IRQOUT}}$ pin function when bits 14 to 12 (PE15MD2 to PE15MD0) in PECRL4 are set to B'011. 00: Interrupt request accept signal output 01: Refresh signal output 10: Interrupt request accept signal output or refresh signal output (depends on the operating state) 11: Always high-level output

21.2 Usage Notes

1. In this LSI, the same function is available as a multiplexed function on multiple pins. This approach is intended to increase the number of selectable pin functions and to allow the easier design of boards. If two or more pins are specified for one function, however, there are two cautions shown below.
   — When the pin function is input
   Signals input to several pins are formed as one signal through OR or AND logic and the signal is transmitted into the LSI. Therefore, a signal that differs from the input signals may be transmitted to the LSI depending on the input signals in other pins that have the same functions. Table 21.22 shows the transmit forms of input functions allocated to several pins. When using one of the functions shown below in multiple pins, use it with care of signal polarity considering the transmit forms.

Table 21.22 Transmit Forms of Input Functions Allocated to Multiple Pins

OR Type	AND Type
SCK0, SCK3, RXD0, RXD3, TIOC3AS to TIOC3DS, TIOC4AS to TIOC4DS, TIC5U, TIC5V, TIC5W, TIC5US, TIC5VS, TIC5WS	IRQ0 to IRQ7, DREQ0, DREQ1, $\overline{\text{BREQ}}$, $\overline{\text{WAIT}}$, $\overline{\text{ADTRG}}$, $\overline{\text{POE4}}$ to $\overline{\text{POE8}}$

   OR type: Signals input to several pins are formed as one signal through OR logic and the signal is transmitted into the LSI.
   AND type: Signals input to several pins are formed as one signal through AND logic and the signal is transmitted into the LSI.
   — When the pin function is output
   Each selected pin can output the same function.
2. When the port input is switched from a low level to the DREQ or the IRQ edge for the pins that are multiplexed with input/output and DREQ or IRQ, the corresponding edge is detected.
3. Do not set functions other than those specified in tables 21.17 to 21.20. Otherwise, correct operation cannot be guaranteed.
4. PFC setting in single-chip mode (MCU operating mode 3)
   In single-chip mode, do not set the PFC to select address bus, data bus, bus control, or the $\overline{\text{BREQ}}$, $\overline{\text{BACK}}$, CK, DACK, or TEND signals. If they are selected, address bus signals function as high- or low-level outputs, data bus signals function as high-impedance outputs, and the other output signals function as high-level outputs. As $\overline{\text{BREQ}}$ and $\overline{\text{WAIT}}$ function as inputs, do not leave them open. However, the bus-mastership-request inputs and external waits are disabled.

Section 22 I/O Ports

The SH7083 has six ports: A to F. Port A is an 11-bit port, port B is a 9-bit port, ports C and D are 16-bit ports, and port E is a 13-bit port. Port F is an 8-bit input-only port.

The SH7084 has six ports: A to F. Port A is an 18-bit port, port B is a 10-bit port, and ports C, D, and E are 16-bit ports. Port F is an 8-bit input-only port.

The SH7085 has six ports: A to F. Port A is a 26-bit port, port B is a 10-bit port, port C is a 16-bit port, port D is a 32-bit port, and port E is a 16-bit port. Port F is an 8-bit input-only port.

The SH7086 has six ports: A to F. Port A is a 30-bit port, port B is a 10-bit port, port C is a 24-bit port, port D is a 32-bit port, and port E is a 22-bit port. Port F is a 16-bit input-only port.

All the port pins are multiplexed as general input/output pins and special function pins. The functions of the multiplex pins are selected by means of the pin function controller (PFC). Each port is provided with a data register for storing the pin data.

22.1 Port A

Port A in the SH7083 is an input/output port with the 11 pins shown in figure 22.1.

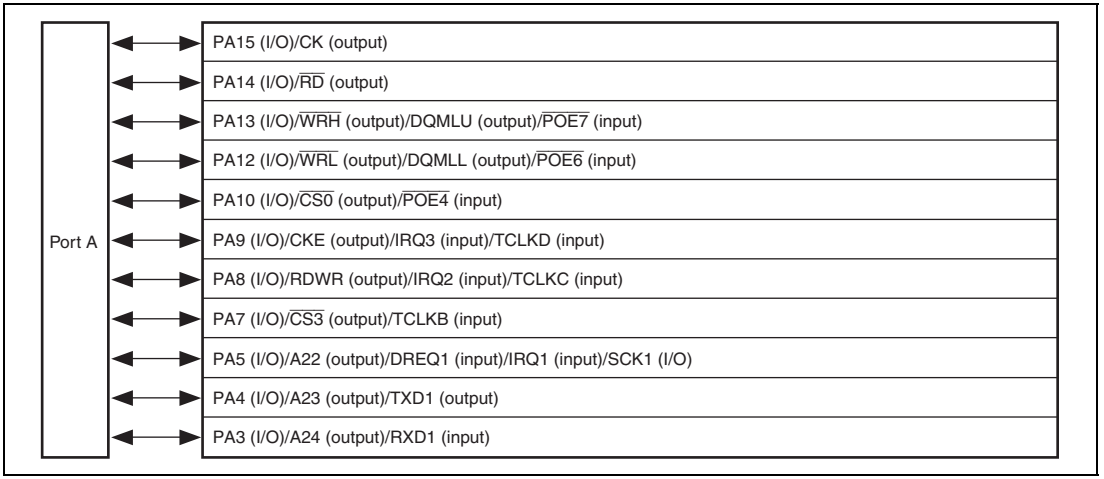


Figure 22.1 Port A (SH7083)

Port A in the SH7084 is an input/output port with the 18 pins shown in figure 22.2.

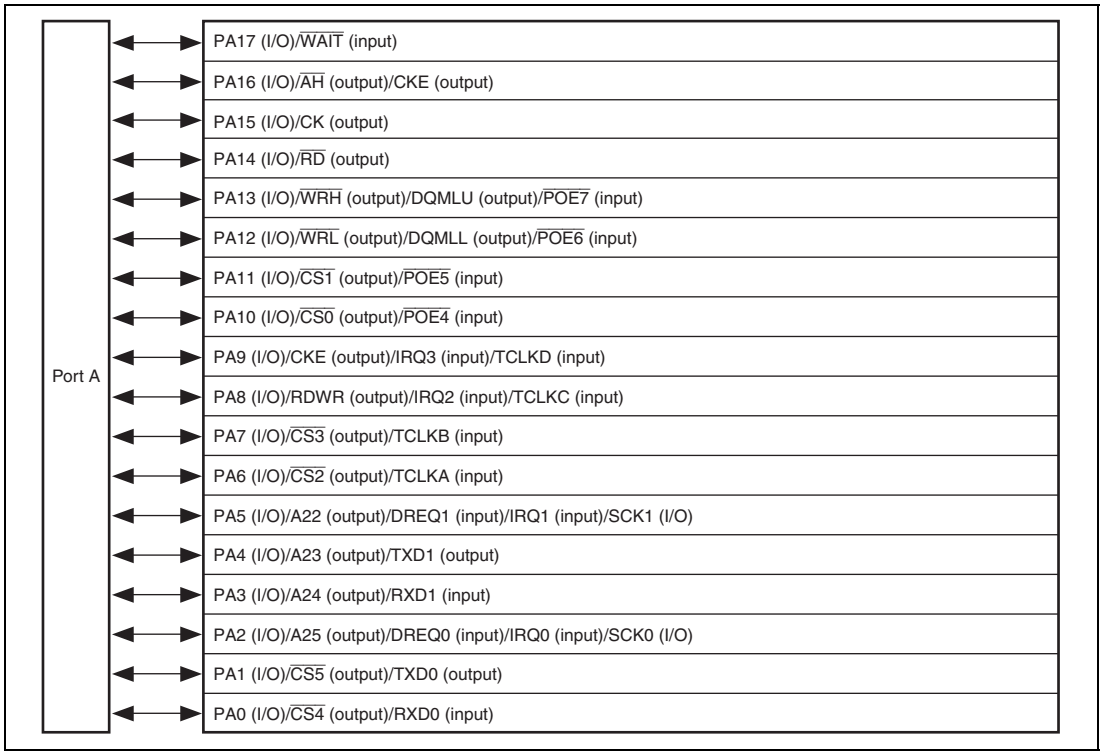
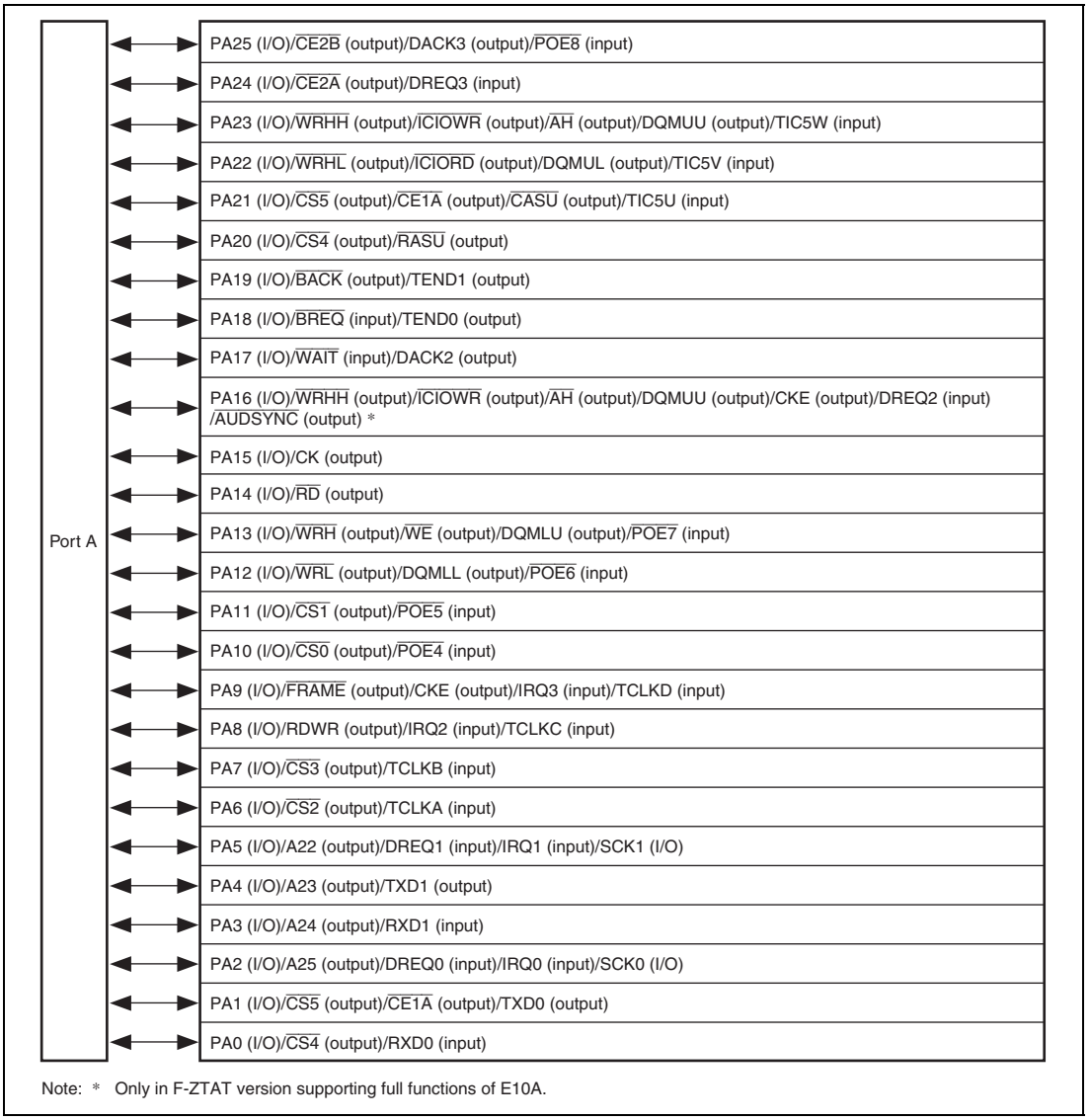


Figure 22.2 Port A (SH7084)

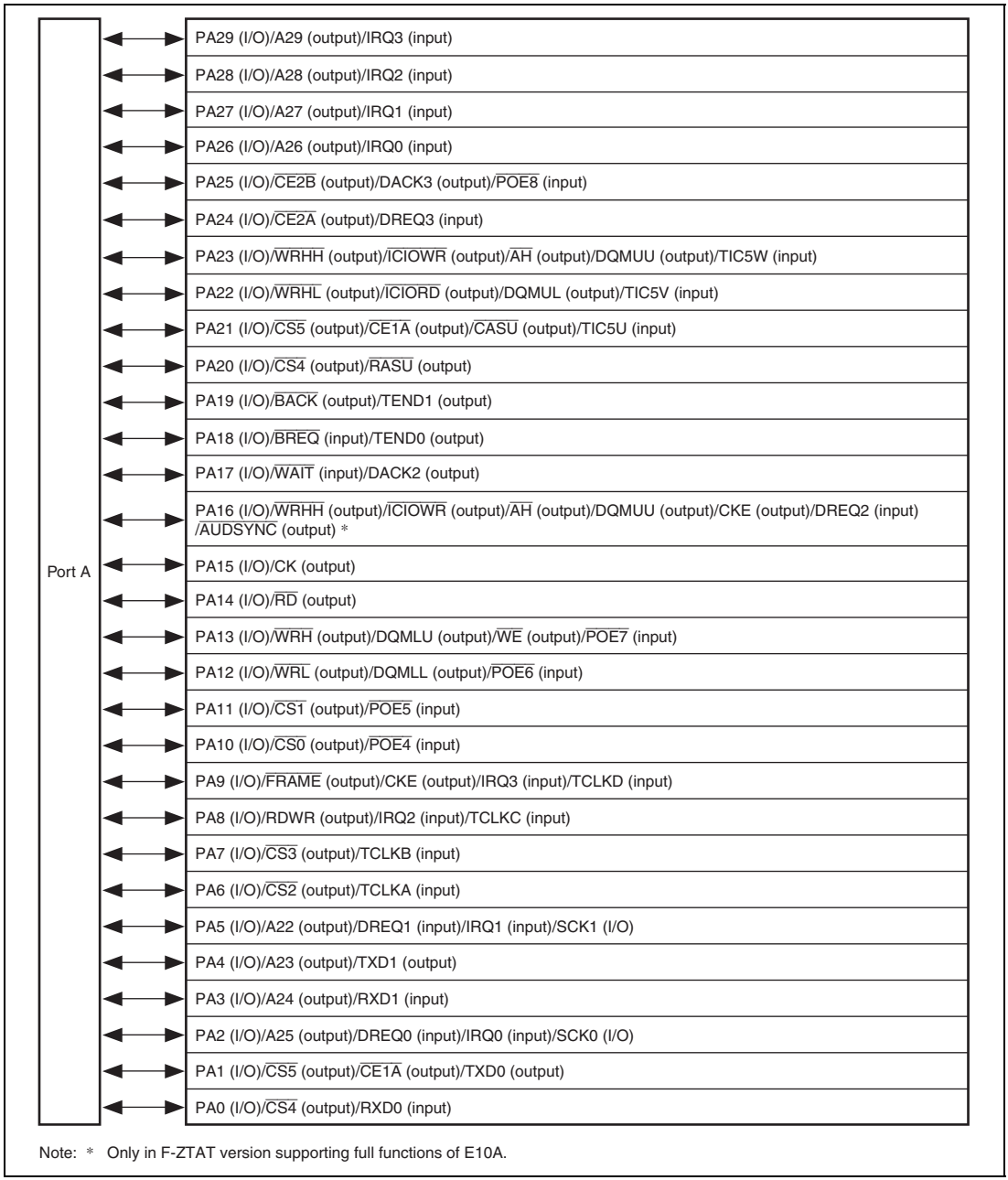
Port A in the SH7085 is an input/output port with the 26 pins shown in figure 22.3.



Note: * Only in F-ZTAT version supporting full functions of E10A.

Figure 22.3 Port A (SH7085)

Port A in the SH7086 is an input/output port with the 30 pins shown in figure 22.4.



Note: * Only in F-ZTAT version supporting full functions of E10A.

Figure 22.4 Port A (SH7086)

22.1.1 Register Descriptions

Port A is an 11-bit input/output port in the SH7083; an 18-bit input/output port in the SH7084; a 26-bit input/output port in the SH7085; a 30-bit input/output port in the SH7086. Port A has the following registers. For details on register addresses and register states during each processing, refer to section 27, List of Registers.

Table 22.1 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Port A data register H	PADRH	R/W	H'0000	H'FFFFD100	8, 16, 32
Port A data register L	PADRL	R/W	H'0000	H'FFFFD102	8, 16
Port A port register H	PAPRH	R	—	H'FFFFD11C	8, 16, 32
Port A port register L	PAPRL	R	—	H'FFFFD11E	8 16

22.1.2 Port A Data Registers H and L (PADRH and PADRL)

The port A data registers H and L (PADRH and PADRL) are 16-bit readable/writable registers that store port A data. Bits PA15DR to PA12DR, PA10DR to PA7DR, and PA5DR to PA3DR correspond to pins PA15 to PA12, PA10 to PA7, and PA5 to PA3, respectively (multiplexed functions omitted here) in the SH7083. Bits PA17DR to PA0DR correspond to pins PA17 to PA0 (multiplexed functions omitted here) in the SH7084. Bits PA25DR to PA0DR correspond to pins PA25 to PA0 (multiplexed functions omitted here) in the SH7085. Bits PA29DR to PA0DR correspond to pins PA29 to PA0 (multiplexed functions omitted here) in the SH7086.

When a pin function is general output, if a value is written to PADRH or PADRL, that value is output directly from the pin, and if PADRH or PADRL is read, the register value is returned directly regardless of the pin state.

When a pin function is general input, if PADRH or PADRL is read, the pin state, not the register value, is returned directly. If a value is written to PADRH or PADRL, although that value is written into PADRH or PADRL, it does not affect the pin state. Table 22.2 summarizes port A data register read/write operations.

- PADRH (SH7083)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- PADRH (SH7084)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PA17DR	PA16DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PA17DR	0	R/W	See table 22.2.
0	PA16DR	0	R/W	

- PADRH (SH7085)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PA25DR	PA24DR	PA23DR	PA22DR	PA21DR	PA20DR	PA19DR	PA18DR	PA17DR	PA16DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PA25DR	0	R/W	See table 22.2.
8	PA24DR	0	R/W	
7	PA23DR	0	R/W	
6	PA22DR	0	R/W	
5	PA21DR	0	R/W	
4	PA20DR	0	R/W	
3	PA19DR	0	R/W	
2	PA18DR	0	R/W	
1	PA17DR	0	R/W	
0	PA16DR	0	R/W	

- PADRH (SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PA29 DR	PA28 DR	PA27 DR	PA26 DR	PA25 DR	PA24 DR	PA23 DR	PA22 DR	PA21 DR	PA20 DR	PA19 DR	PA18 DR	PA17 DR	PA16 DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PA29DR	0	R/W	See table 22.2.
12	PA28DR	0	R/W	
11	PA27DR	0	R/W	
10	PA26DR	0	R/W	
9	PA25DR	0	R/W	
8	PA24DR	0	R/W	
7	PA23DR	0	R/W	
6	PA22DR	0	R/W	
5	PA21DR	0	R/W	
4	PA20DR	0	R/W	
3	PA19DR	0	R/W	
2	PA18DR	0	R/W	
1	PA17DR	0	R/W	
0	PA16DR	0	R/W	

- PADRL (SH7083)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PA15 DR	PA14 DR	PA13 DR	PA12 DR	-	PA10 DR	PA9 DR	PA8 DR	PA7 DR	-	PA5 DR	PA4 DR	PA3 DR	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	PA15DR	0	R/W	See table 22.2.
14	PA14DR	0	R/W	
13	PA13DR	0	R/W	
12	PA12DR	0	R/W	
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA10DR	0	R/W	See table 22.2.
9	PA9DR	0	R/W	
8	PA8DR	0	R/W	
7	PA7DR	0	R/W	
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5	PA5DR	0	R/W	See table 22.2.
4	PA4DR	0	R/W	
3	PA3DR	0	R/W	
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- PADRL (SH7084, SH7085, SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PA15 DR	PA14 DR	PA13 DR	PA12 DR	PA11 DR	PA10 DR	PA9 DR	PA8 DR	PA7 DR	PA6 DR	PA5 DR	PA4 DR	PA3 DR	PA2 DR	PA1 DR	PA0 DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PA15DR	0	R/W	See table 22.2.
14	PA14DR	0	R/W	
13	PA13DR	0	R/W	
12	PA12DR	0	R/W	
11	PA11DR	0	R/W	
10	PA10DR	0	R/W	
9	PA9DR	0	R/W	
8	PA8DR	0	R/W	
7	PA7DR	0	R/W	
6	PA6DR	0	R/W	
5	PA5DR	0	R/W	
4	PA4DR	0	R/W	
3	PA3DR	0	R/W	
2	PA2DR	0	R/W	
1	PA1DR	0	R/W	
0	PA0DR	0	R/W	

Table 22.2 Port A Data Register (PADR) Read/Write Operations

- PADRH Bits 13 to 0 and PADRL Bits 15 to 0

PAIOR	Pin Function	Read	Write
0	General input	Pin state	Can write to PADRH and PADRL, but it has no effect on pin state
	Other than general input	Pin state	Can write to PADRH and PADRL, but it has no effect on pin state
1	General output	PADRH or PADRL value	Value written is output from pin
	Other than general output	PADRH or PADRL value	Can write to PADRH and PADRL, but it has no effect on pin state

22.1.3 Port A Port Registers H and L (PAPRH and PAPRL)

The port A port registers H and L (PAPRH and PAPRL) are 16-bit read-only registers that always return the states of the pins regardless of the PFC setting. Bits PA15PR to PA12PR, PA10PR to PA7PR, and PA5PR to PA3PR correspond to pins PA15 to PA12, PA10 to PA7, and PA5 to PA3, respectively (multiplexed functions omitted here) in the SH7083. Bits PA17PR to PA0PR correspond to pins PA17 to PA0 (multiplexed functions omitted here) in the SH7084. Bits PA25PR to PA0PR correspond to pins PA25 to PA0 (multiplexed functions omitted here) in the SH7085. Bits PA29PR to PA0PR correspond to pins PA29 to PA0 (multiplexed functions omitted here) in the SH7086.

- PAPRH (SH7083)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- PAPRH (SH7084)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PA17PR	PA16PR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PA17PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
0	PA16PR	Pin state	R	

- PAPRH (SH7085)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PA25PR	PA24PR	PA23PR	PA22PR	PA21PR	PA20PR	PA19PR	PA18PR	PA17PR	PA16PR
Initial value:	0	0	0	0	0	0	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PA25PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
8	PA24PR	Pin state	R	
7	PA23PR	Pin state	R	
6	PA22PR	Pin state	R	
5	PA21PR	Pin state	R	
4	PA20PR	Pin state	R	
3	PA19PR	Pin state	R	
2	PA18PR	Pin state	R	
1	PA17PR	Pin state	R	
0	PA16PR	Pin state	R	

- PAPRH (SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PA29 PR	PA28 PR	PA27 PR	PA26 PR	PA25 PR	PA24 PR	PA23 PR	PA22 PR	PA21 PR	PA20 PR	PA19 PR	PA18 PR	PA17 PR	PA16 PR
Initial value:	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PA29PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
12	PA28PR	Pin state	R	
11	PA27PR	Pin state	R	
10	PA26PR	Pin state	R	
9	PA25PR	Pin state	R	
8	PA24PR	Pin state	R	
7	PA23PR	Pin state	R	
6	PA22PR	Pin state	R	
5	PA21PR	Pin state	R	
4	PA20PR	Pin state	R	
3	PA19PR	Pin state	R	
2	PA18PR	Pin state	R	
1	PA17PR	Pin state	R	
0	PA16PR	Pin state	R	

- PAPRL (SH7083)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PA15 PR	PA14 PR	PA13 PR	PA12 PR	-	PA10 PR	PA9 PR	PA8 PR	PA7 PR	-	PA5 PR	PA4 PR	PA3 PR	-	-	-
Initial value:	*	*	*	*	0	*	*	*	*	0	*	*	*	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	PA15PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
14	PA14PR	Pin state	R	
13	PA13PR	Pin state	R	
12	PA12PR	Pin state	R	
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA10PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
9	PA9PR	Pin state	R	
8	PA8PR	Pin state	R	
7	PA7PR	Pin state	R	
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5	PA5PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
4	PA4PR	Pin state	R	
3	PA3PR	Pin state	R	
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- PAPRL (SH7084, SH7085, SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PA15 PR	PA14 PR	PA13 PR	PA12 PR	PA11 PR	PA10 PR	PA9 PR	PA8 PR	PA7 PR	PA6 PR	PA5 PR	PA4 PR	PA3 PR	PA2 PR	PA1 PR	PA0 PR
Initial value:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	PA15PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
14	PA14PR	Pin state	R	
13	PA13PR	Pin state	R	
12	PA12PR	Pin state	R	
11	PA11PR	Pin state	R	
10	PA10PR	Pin state	R	
9	PA9PR	Pin state	R	
8	PA8PR	Pin state	R	
7	PA7PR	Pin state	R	
6	PA6PR	Pin state	R	
5	PA5PR	Pin state	R	
4	PA4PR	Pin state	R	
3	PA3PR	Pin state	R	
2	PA2PR	Pin state	R	
1	PA1PR	Pin state	R	
0	PA0PR	Pin state	R	

22.2 Port B

Port B in the SH7083 is an input/output port with the nine pins shown in figure 22.5.

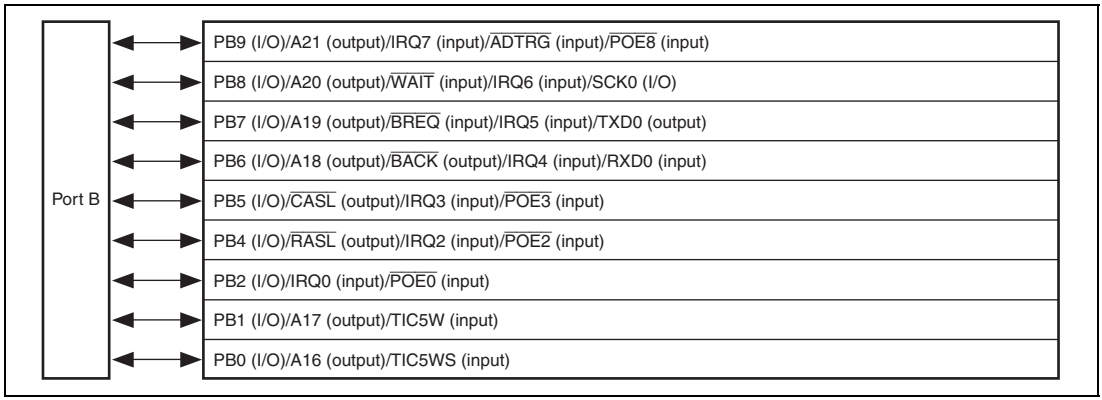


Figure 22.5 Port B (SH7083)

Port B in the SH7084, SH7085, and SH7086 is an input/output port with the 10 pins shown in figure 22.6.

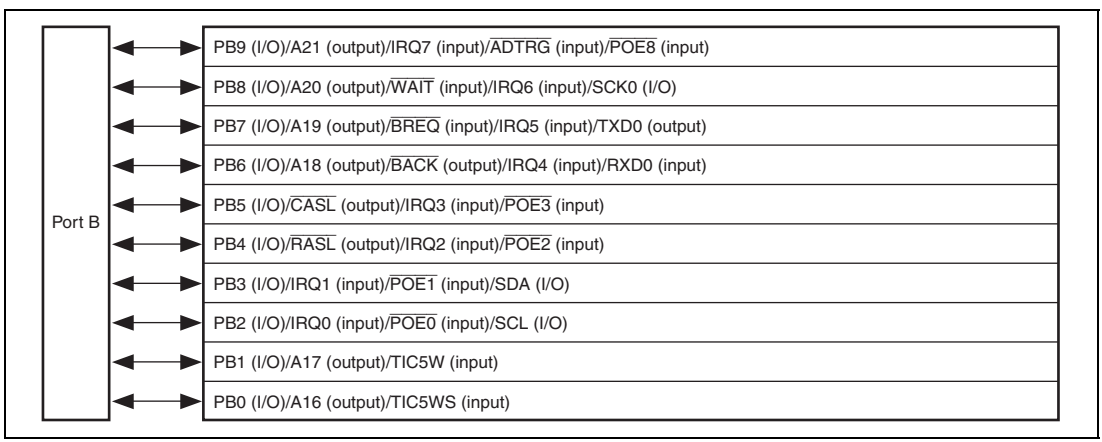


Figure 22.6 Port B (SH7084, SH7085, SH7086)

22.2.1 Register Descriptions

Port B is a 9-bit input/output port in the SH7083; a 10-bit input/output port in the SH7084, SH7085, and SH7086. Port B has the following register. For details on register addresses and register states during each processing, refer to section 27, List of Registers.

Table 22.3 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Port B data register L	PBDRL	R/W	H'0000	H'FFFFD182	8, 16
Port B port register L	PBPRL	R	H'0xxx	H'FFFFD19E	8, 16

22.2.2 Port B Data Register L (PBDRL)

The port B data register L (PBDRL) is a 16-bit readable/writable register that stores port B data. Bits PB9DR to PB4DR and PB2DR to PB0DR correspond to pins PB9 to PB4 and PB2 to PB0, respectively (multiplexed functions omitted here) in the SH7083. Bits PB9DR to PB0DR correspond to pins PB9 to PB0 (multiplexed functions omitted here) in the SH7084, SH7085, and SH7086.

When a pin function is general output, if a value is written to PBDRL, that value is output directly from the pin, and if PBDRL is read, the register value is returned directly regardless of the pin state.

When a pin function is general input, if PBDRL is read, the pin state, not the register value, is returned directly. If a value is written to PBDRL, although that value is written into PBDRL, it does not affect the pin state. Table 22.4 summarizes port B data register read/write operations.

- PBDRL (SH7083)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PB9DR	PB8DR	PB7DR	PB6DR	PB5DR	PB4DR	-	PB2DR	PB1DR	PB0DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PB9DR	0	R/W	See table 22.4.
8	PB8DR	0	R/W	
7	PB7DR	0	R/W	
6	PB6DR	0	R/W	
5	PB5DR	0	R/W	
4	PB4DR	0	R/W	
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PB2DR	0	R/W	See table 22.4.
1	PB1DR	0	R/W	
0	PB0DR	0	R/W	

- PBDRL (SH7084, SH7085, SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PB9 DR	PB8 DR	PB7 DR	PB6 DR	PB5 DR	PB4 DR	PB3 DR	PB2 DR	PB1 DR	PB0 DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PB9DR	0	R/W	See table 22.4.
8	PB8DR	0	R/W	
7	PB7DR	0	R/W	
6	PB6DR	0	R/W	
5	PB5DR	0	R/W	
4	PB4DR	0	R/W	
3	PB3DR	0	R/W	
2	PB2DR	0	R/W	
1	PB1DR	0	R/W	
0	PB0DR	0	R/W	

Table 22.4 Port B Data Register L (PBDRL) Read/Write Operations

- PBDRL Bits 9 to 0

PBIOR	Pin Function	Read	Write
0	General input	Pin state	Can write to PBDRL, but it has no effect on pin state
	Other than general input	Pin state	Can write to PBDRL, but it has no effect on pin state
1	General output	PBDRL value	Value written is output from pin
	Other than general output	PBDRL value	Can write to PBDRL, but it has no effect on pin state

22.2.3 Port B Port Register L (PBPRL)

The port B port register L (PBPRL) is a 16-bit read-only register that always returns the states of the pins regardless of the PFC setting. Bits PB9PR to PB4PR and PB2PR to PB0PR correspond to pins PB9 to PB4 and PB2 to PB0, respectively (multiplexed functions omitted here) in the SH7083. Bits PB9PR to PB0PR correspond to pins PB9 to PB0 (multiplexed functions omitted here) in the SH7084, SH7085, and SH7086

- PBPRL (SH7083)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PB9PR	PB8PR	PB7PR	PB6PR	PB5PR	PB4PR	-	PB2PR	PB1PR	PB0PR
Initial value:	0	0	0	0	0	0	*	*	*	*	*	*	0	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PB9PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
8	PB8PR	Pin state	R	
7	PB7PR	Pin state	R	
6	PB6PR	Pin state	R	
5	PB5PR	Pin state	R	
4	PB4PR	Pin state	R	
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	PB2PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
1	PB1PR	Pin state	R	
0	PB0PR	Pin state	R	

- PBPRL (SH7084, SH7085, SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PB9PR	PB8PR	PB7PR	PB6PR	PB5PR	PB4PR	PB3PR	PB2PR	PB1PR	PB0PR
Initial value:	0	0	0	0	0	0	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PB9PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
8	PB8PR	Pin state	R	
7	PB7PR	Pin state	R	
6	PB6PR	Pin state	R	
5	PB5PR	Pin state	R	
4	PB4PR	Pin state	R	
3	PB3PR	Pin state	R	
2	PB2PR	Pin state	R	
1	PB1PR	Pin state	R	
0	PB0PR	Pin state	R	

22.3 Port C

Port C in the SH7083, SH7084, and SH7085 is an input/output port with the 16 pins shown in figure 22.7.

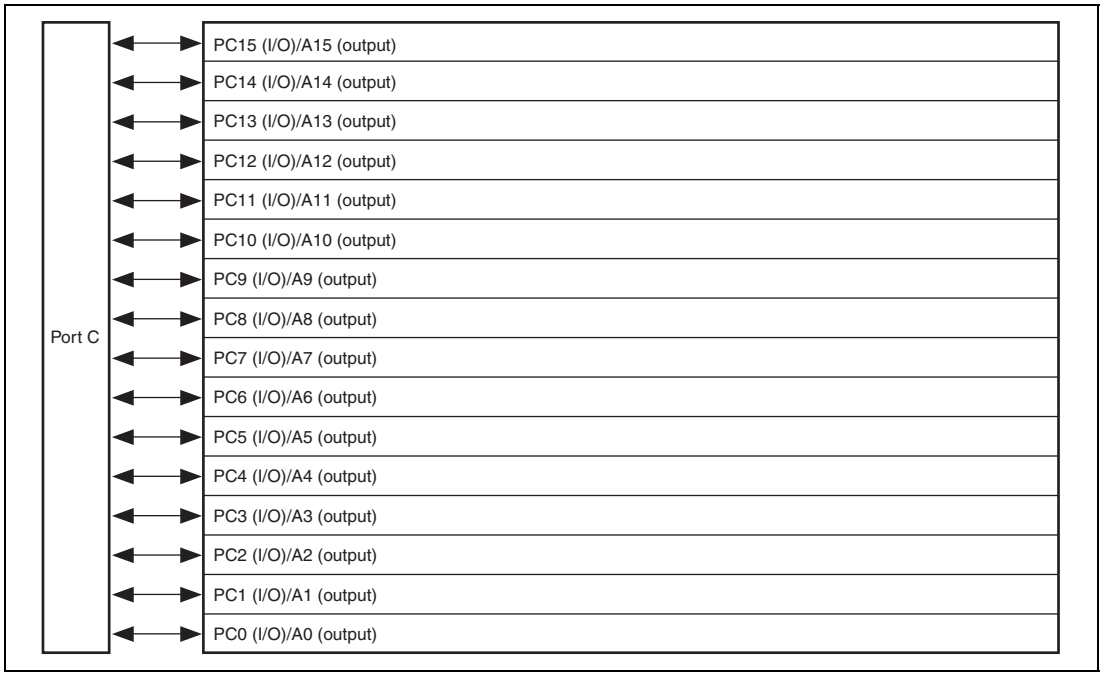


Figure 22.7 Port C (SH7083, SH7084, SH7085)

Port C in the SH7086 is an input/output port with the 24 pins shown in figure 22.8.

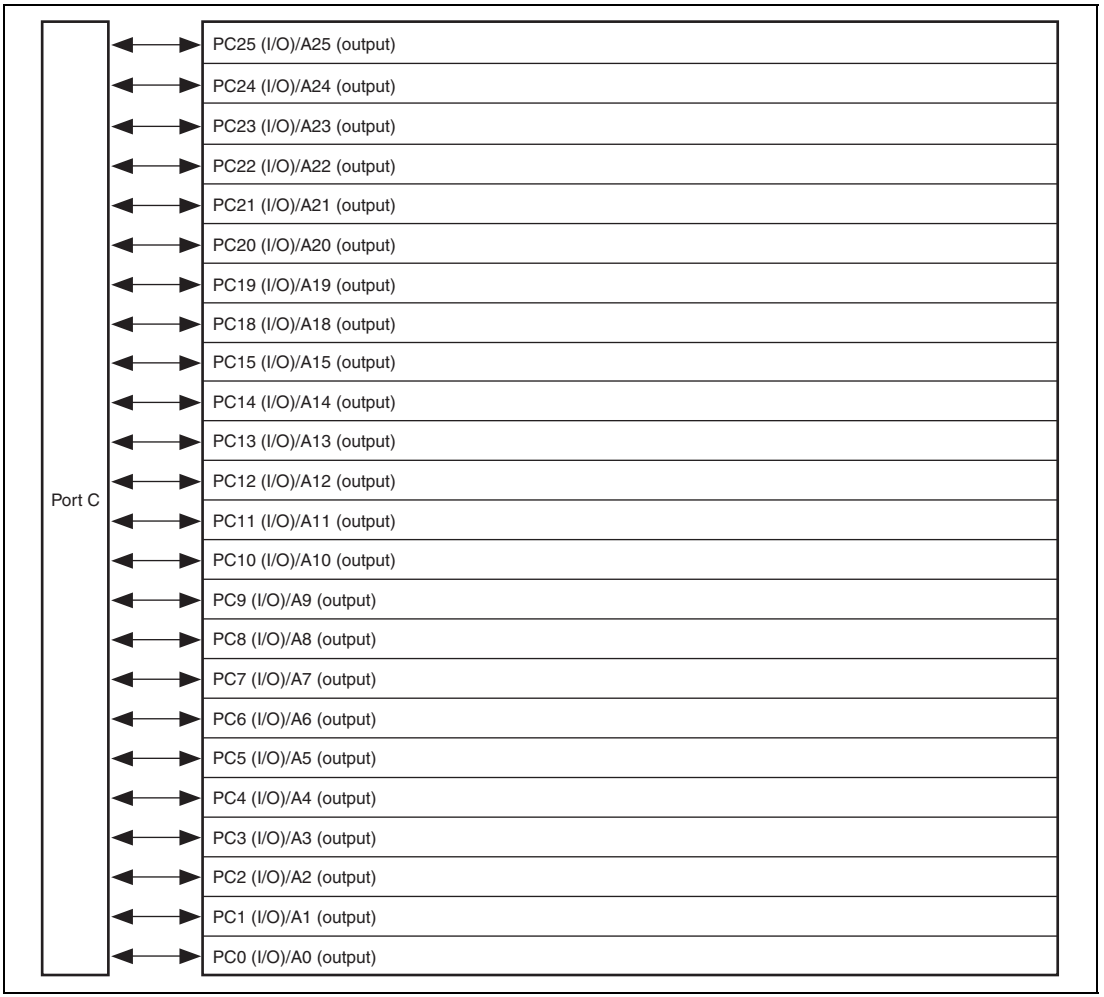


Figure 22.8 Port C (SH7086)

22.3.1 Register Descriptions

Port C is a 16-bit input/output port in the SH7083, SH7084, and SH7085; a 24-bit input/output port in the SH7086. Port C has the following registers. For details on register addresses and register states during each processing, refer to section 27, List of Registers.

Table 22.5 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Port C data register H	PCDRH	R/W	H'0000	H'FFFFD200	8, 16, 32
Port C data register L	PCDRL	R/W	H'0000	H'FFFFD202	8, 16
Port C port register H	PCPRH	R	H'xxxx	H'FFFFD21C	8, 16, 32
Port C port register L	PCPRL	R	H'xxxx	H'FFFFD21E	8, 16

22.3.2 Port C Data Registers H and L (PCDRH and PCDRL)

The port C data registers H and L (PCDRH and PCDRL) are 16-bit readable/writable registers that store port C data. Bits PC15DR to PC0DR correspond to pins PC15 to PC0 (multiplexed functions omitted here) in the SH7083, SH7084, and SH7085. Bits PC25DR to PC18DR and PC15DR to PC0DR correspond to pins PC25 to PC18 and PC15 to PC0, respectively (multiplexed functions omitted here) in the SH7086.

When a pin function is general output, if a value is written to PCDRH or PCDRL, that value is output directly from the pin, and if PCDRH or PCDRL is read, the register value is returned directly regardless of the pin state.

When a pin function is general input, if PCDRH or PCDRL is read, the pin state, not the register value, is returned directly. If a value is written to PCDRH or PCDRL, although that value is written into PCDRH or PCDRL, it does not affect the pin state. Table 22.6 summarizes port C data register read/write operations.

- PCDRH (SH7083, SH7084, SH7085)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- PCDRH (SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PC25 DR	PC24 DR	PC23 DR	PC22 DR	PC21 DR	PC20 DR	PC19 DR	PC18 DR	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PC25DR	0	R/W	See table 22.6.
8	PC24DR	0	R/W	
7	PC23DR	0	R/W	
6	PC22DR	0	R/W	
5	PC21DR	0	R/W	
4	PC20DR	0	R/W	
3	PC19DR	0	R/W	
2	PC18DR	0	R/W	
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- PCDRL

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PC15DR	PC14DR	PC13DR	PC12DR	PC11DR	PC10DR	PC9DR	PC8DR	PC7DR	PC6DR	PC5DR	PC4DR	PC3DR	PC2DR	PC1DR	PC0DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PC15DR	0	R/W	See table 22.6.
14	PC14DR	0	R/W	
13	PC13DR	0	R/W	
12	PC12DR	0	R/W	
11	PC11DR	0	R/W	
10	PC10DR	0	R/W	
9	PC9DR	0	R/W	
8	PC8DR	0	R/W	
7	PC7DR	0	R/W	
6	PC6DR	0	R/W	
5	PC5DR	0	R/W	
4	PC4DR	0	R/W	
3	PC3DR	0	R/W	
2	PC2DR	0	R/W	
1	PC1DR	0	R/W	
0	PC0DR	0	R/W	

Table 22.6 Port C Data Register (PCDR) Read/Write Operations

- PCDRH Bits 9 to 2 and PCDRL Bits 15 to 0

PCIOR	Pin Function	Read	Write
0	General input	Pin state	Can write to PCDRH and PCDRL, but it has no effect on pin state
	Other than general input	Pin state	Can write to PCDRH and PCDRL, but it has no effect on pin state
1	General output	PCDRH or PCDRL value	Value written is output from pin
	Other than general output	PCDRH or PCDRL value	Can write to PCDRH and PCDRL, but it has no effect on pin state

22.3.3 Port C Port Registers H and L (PCPRH and PCPRL)

The port C port registers H and L (PCPRH and PCPRL) are 16-bit read-only registers that always return the states of the pins regardless of the PFC setting. Bits PC15PR to PC0PR correspond to pins PC15 to PC0 (multiplexed functions omitted here) in the SH7083, SH7084, and SH7085. Bits PC25PR to PC18PR and PC15PR to PC0PR correspond to pins PC25 to PC18 and PC15 to PC0, respectively (multiplexed functions omitted here) in the SH7086.

- PCPRH (SH7083, SH7084, SH7085)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- PCPRH (SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PC25PR	PC24PR	PC23PR	PC22PR	PC21PR	PC20PR	PC19PR	PC18PR	-	-
Initial value:	0	0	0	0	0	0	*	*	*	*	*	*	*	*	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PC25PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
8	PC24PR	Pin state	R	
7	PC23PR	Pin state	R	
6	PC22PR	Pin state	R	
5	PC21PR	Pin state	R	
4	PC20PR	Pin state	R	
3	PC19PR	Pin state	R	
2	PC18PR	Pin state	R	
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

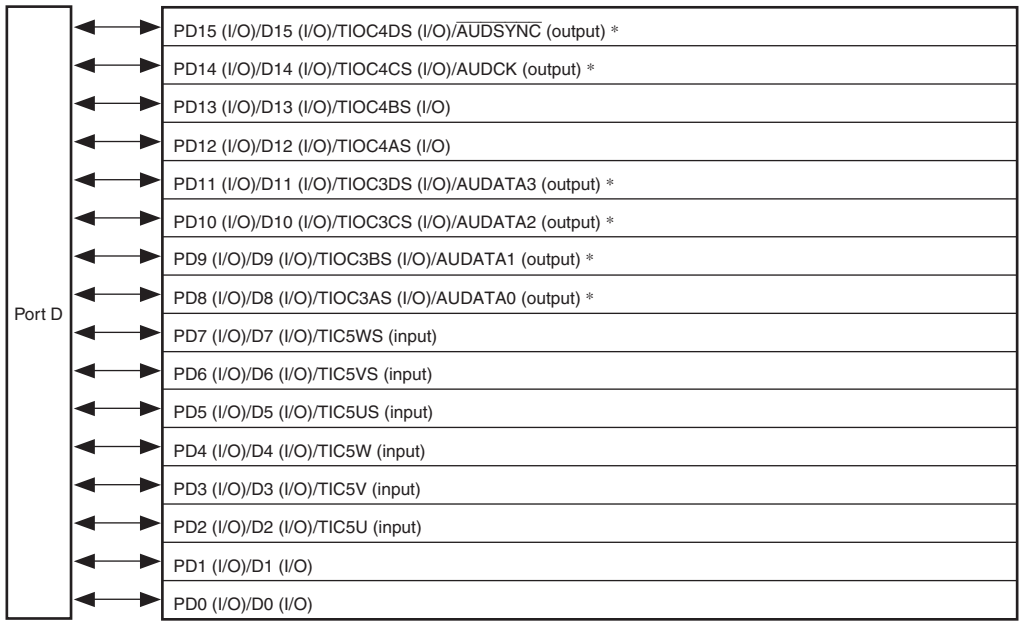
- PCPRL

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PC15PR	PC14PR	PC13PR	PC12PR	PC11PR	PC10PR	PC9PR	PC8PR	PC7PR	PC6PR	PC5PR	PC4PR	PC3PR	PC2PR	PC1PR	PC0PR
Initial value:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	PC15PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
14	PC14PR	Pin state	R	
13	PC13PR	Pin state	R	
12	PC12PR	Pin state	R	
11	PC11PR	Pin state	R	
10	PC10PR	Pin state	R	
9	PC9PR	Pin state	R	
8	PC8PR	Pin state	R	
7	PC7PR	Pin state	R	
6	PC6PR	Pin state	R	
5	PC5PR	Pin state	R	
4	PC4PR	Pin state	R	
3	PC3PR	Pin state	R	
2	PC2PR	Pin state	R	
1	PC1PR	Pin state	R	
0	PC0PR	Pin state	R	

22.4 Port D

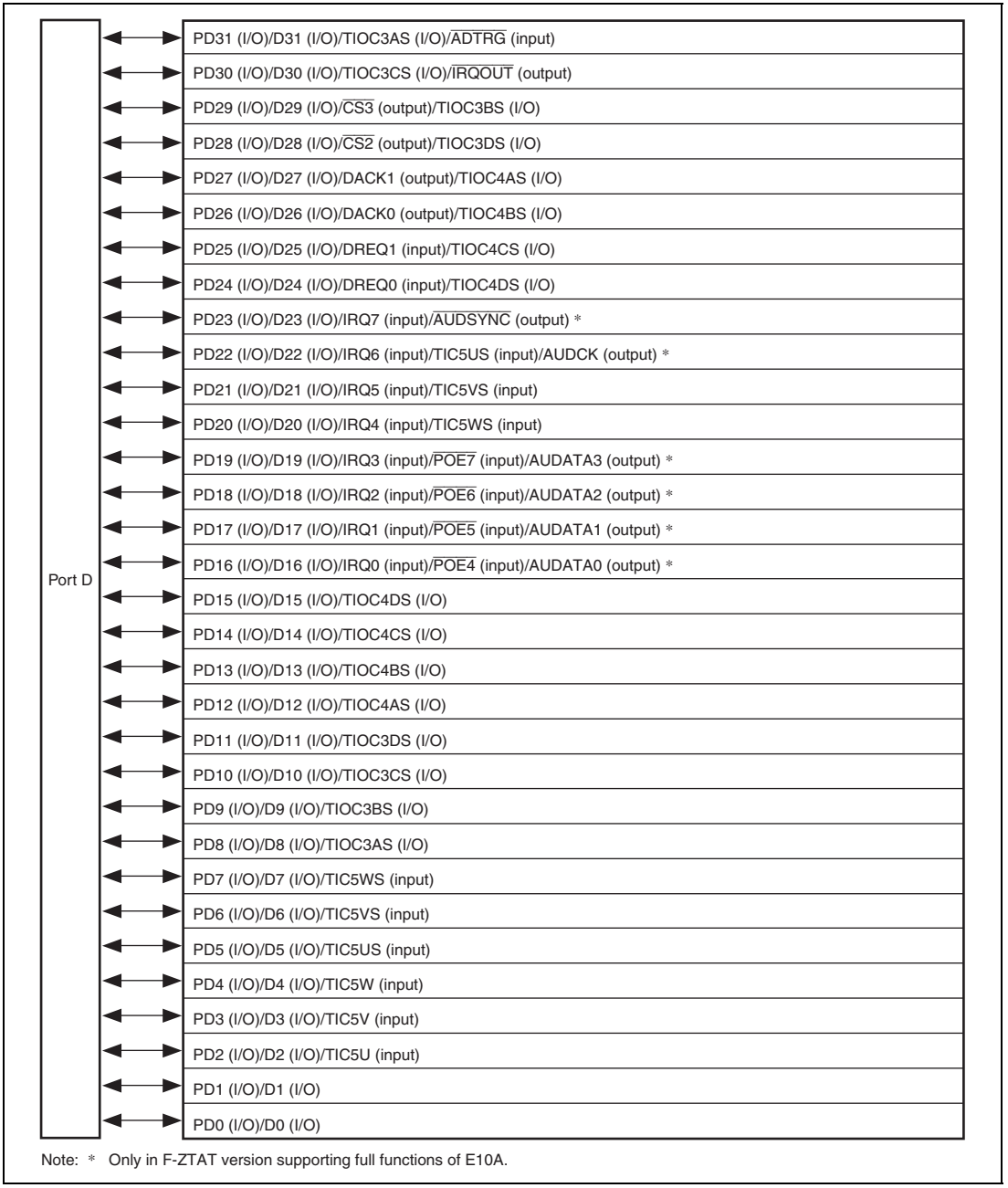
Port D in the SH7083 and SH7084 is an input/output port with the 16 pins shown in figure 22.9.



Note: * Only in F-ZTAT version supporting full functions of E10A.

Figure 22.9 Port D (SH7083, SH7084)

Port D in the SH7085 and SH7086 is an input/output port with the 32 pins shown in figure 22.10.



Note: * Only in F-ZTAT version supporting full functions of E10A.

Figure 22.10 Port D (SH7085, SH7086)

22.4.1 Register Descriptions

Port D is a 16-bit input/output port in the SH7083 and SH7084; a 32-bit input/output port in the SH7085 and SH7086. Port D has the following registers. For details on register addresses and register states during each processing, refer to section 27, List of Registers.

Table 22.7 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Port D data register H	PDDRH	R/W	H'0000	H'FFFFD280	8, 16, 32
Port D data register L	PDDRL	R/W	H'0000	H'FFFFD282	8, 16
Port D port register H	PDPRH	R	H'xxxx	H'FFFFD29C	8, 16, 32
Port D port register L	PDPRL	R	H'xxxx	H'FFFFD29E	8, 16

22.4.2 Port D Data Registers H and L (PDDRH and PDDRL)

The port D data registers H and L (PDDRH and PDDRL) are 16-bit readable/writable registers that store port D data. Bits PD15DR to PD0DR correspond to pins PD15 to PD0 (multiplexed functions omitted here) in the SH7083 and SH7084. Bits PD31DR to PD0DR correspond to pins PD31 to PD0 (multiplexed functions omitted here) in the SH7085 and SH7086.

When a pin function is general output, if a value is written to PDDRH or PDDRL, that value is output directly from the pin, and if PDDRH or PDDRL is read, the register value is returned directly regardless of the pin state.

When a pin function is general input, if PDDRH or PDDRL is read, the pin state, not the register value, is returned directly. If a value is written to PDDRH or PDDRL, although that value is written into PDDRH or PDDRL, it does not affect the pin state. Table 22.8 summarizes port D data register read/write operations.

- PDDRH (SH7083, SH7084)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- PDDRH (SH7085, SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD31 DR	PD30 DR	PD29 DR	PD28 DR	PD27 DR	PD26 DR	PD25 DR	PD24 DR	PD23 DR	PD22 DR	PD21 DR	PD20 DR	PD19 DR	PD18 DR	PD17 DR	PD16 DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PD31DR	0	R/W	See table 22.8.
14	PD30DR	0	R/W	
13	PD29DR	0	R/W	
12	PD28DR	0	R/W	
11	PD27DR	0	R/W	
10	PD26DR	0	R/W	
9	PD25DR	0	R/W	
8	PD24DR	0	R/W	
7	PD23DR	0	R/W	
6	PD22DR	0	R/W	
5	PD21DR	0	R/W	
4	PD20DR	0	R/W	
3	PD19DR	0	R/W	
2	PD18DR	0	R/W	
1	PD17DR	0	R/W	
0	PD16DR	0	R/W	

- PDDRL

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD15 DR	PD14 DR	PD13 DR	PD12 DR	PD11 DR	PD10 DR	PD9 DR	PD8 DR	PD7 DR	PD6 DR	PD5 DR	PD4 DR	PD3 DR	PD2 DR	PD1 DR	PD0 DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PD15DR	0	R/W	See table 22.8.
14	PD14DR	0	R/W	
13	PD13DR	0	R/W	
12	PD12DR	0	R/W	
11	PD11DR	0	R/W	
10	PD10DR	0	R/W	
9	PD9DR	0	R/W	
8	PD8DR	0	R/W	
7	PD7DR	0	R/W	
6	PD6DR	0	R/W	
5	PD5DR	0	R/W	
4	PD4DR	0	R/W	
3	PD3DR	0	R/W	
2	PD2DR	0	R/W	
1	PD1DR	0	R/W	
0	PD0DR	0	R/W	

Table 22.8 Port D Data Register (PDDR) Read/Write Operations

- PDDRH Bits 15 to 0 and PDDRL Bits 15 to 0

PDIORH	Pin Function	Read	Write
0	General input	Pin state	Can write to PDDRH or PDDRL, but it has no effect on pin state
	Other than general input	Pin state	Can write to PDDRH or PDDRL, but it has no effect on pin state
1	General output	PDDRH or PDDRL value	Value written is output from pin
	Other than general output	PDDRH or PDDRL value	Can write to PDDRH or PDDRL, but it has no effect on pin state

22.4.3 Port D Port Registers H and L (PDPRH and PDPRL)

The port D port registers H and L (PDPRH and PDPRL) are 16-bit read-only registers that always return the states of the pins regardless of the PFC setting. Bits PD15PR to PD0PR correspond to pins PD15 to PD0 (multiplexed functions omitted here) in the SH7083 and SH7084. Bits PD31PR to PD0PR correspond to pins PD31 to PD0 (multiplexed functions omitted here) in the SH7085 and SH7086.

- PDPRH (SH7083, SH7084)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- PDPRH (SH7085, SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD31PR	PD30PR	PD29PR	PD28PR	PD27PR	PD26PR	PD25PR	PD24PR	PD23PR	PD22PR	PD21PR	PD20PR	PD19PR	PD18PR	PD17PR	PD16PR
Initial value:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	PD31PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
14	PD30PR	Pin state	R	
13	PD29PR	Pin state	R	
12	PD28PR	Pin state	R	
11	PD27PR	Pin state	R	
10	PD26PR	Pin state	R	
9	PD25PR	Pin state	R	
8	PD24PR	Pin state	R	
7	PD23PR	Pin state	R	
6	PD22PR	Pin state	R	
5	PD21PR	Pin state	R	
4	PD20PR	Pin state	R	
3	PD19PR	Pin state	R	
2	PD18PR	Pin state	R	
1	PD17PR	Pin state	R	
0	PD16PR	Pin state	R	

- PDPRL

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD15PR	PD14PR	PD13PR	PD12PR	PD11PR	PD10PR	PD9PR	PD8PR	PD7PR	PD6PR	PD5PR	PD4PR	PD3PR	PD2PR	PD1PR	PD0PR
Initial value:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	PD15PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
14	PD14PR	Pin state	R	
13	PD13PR	Pin state	R	
12	PD12PR	Pin state	R	
11	PD11PR	Pin state	R	
10	PD10PR	Pin state	R	
9	PD9PR	Pin state	R	
8	PD8PR	Pin state	R	
7	PD7PR	Pin state	R	
6	PD6PR	Pin state	R	
5	PD5PR	Pin state	R	
4	PD4PR	Pin state	R	
3	PD3PR	Pin state	R	
2	PD2PR	Pin state	R	
1	PD1PR	Pin state	R	
0	PD0PR	Pin state	R	

22.5 Port E

Port E in the SH7083 is an input/output port with the 13 pins shown in figure 22.11.

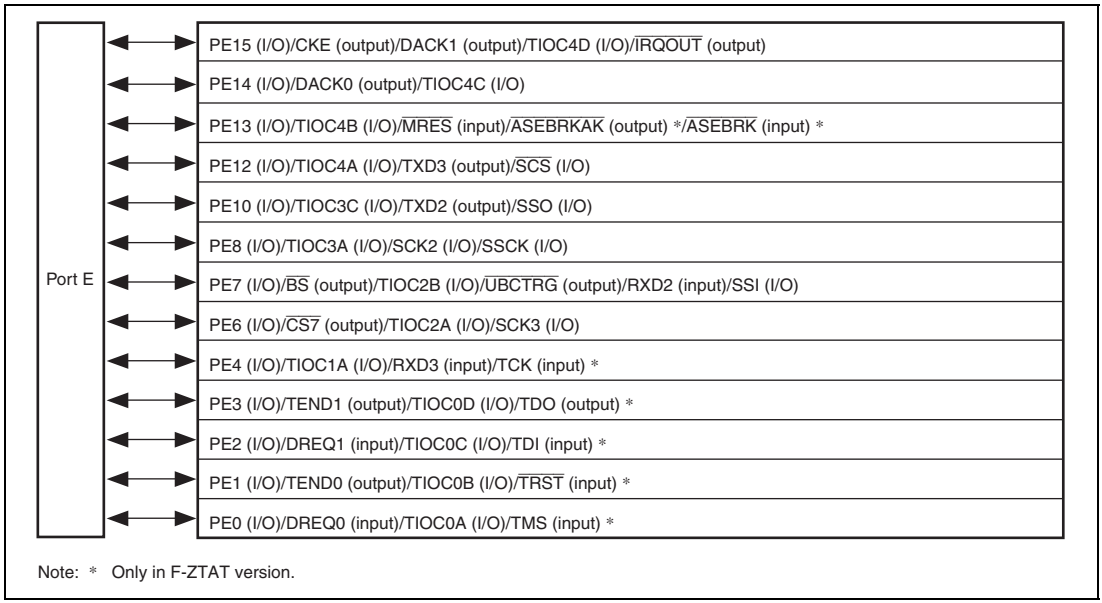
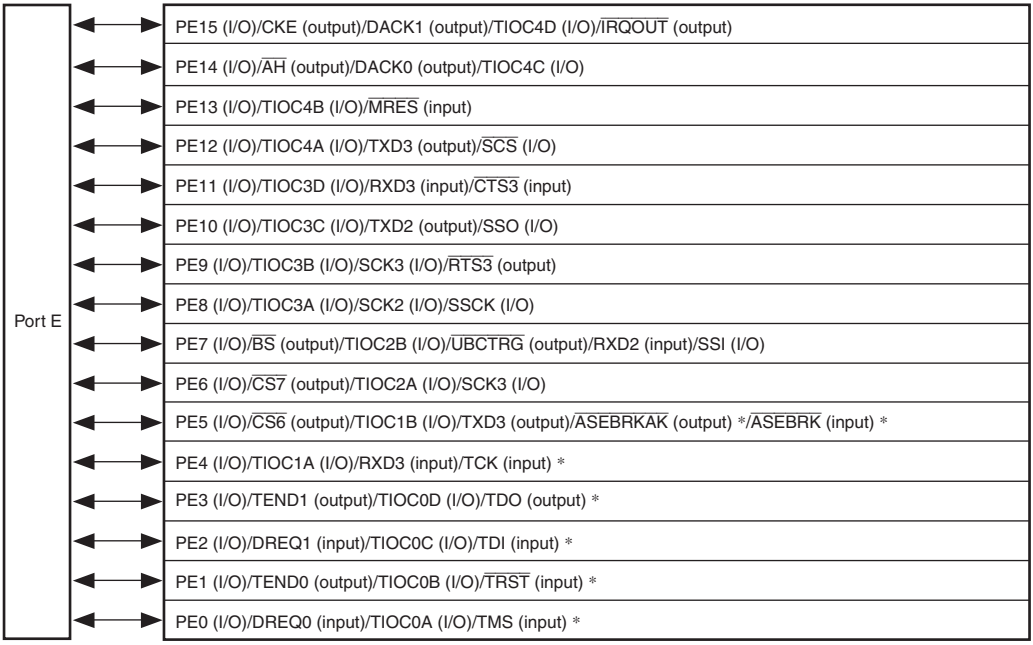


Figure 22.11 Port E (SH7083)

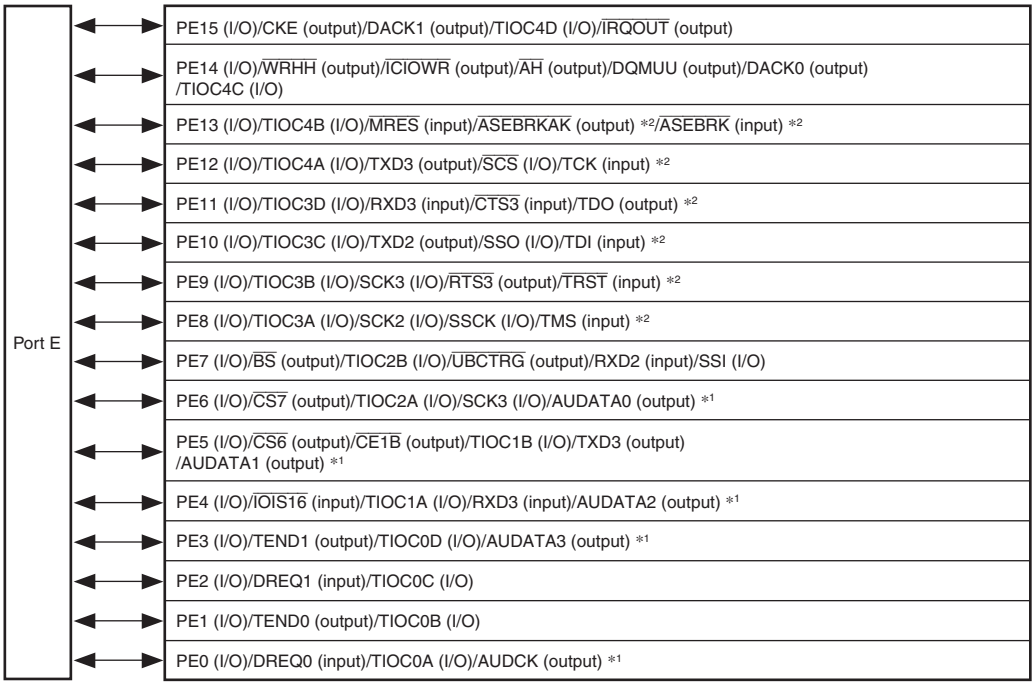
Port E in the SH7084 is an input/output port with the 16 pins shown in figure 22.12.



Note: * Only in F-ZTAT version.

Figure 22.12 Port E (SH7084)

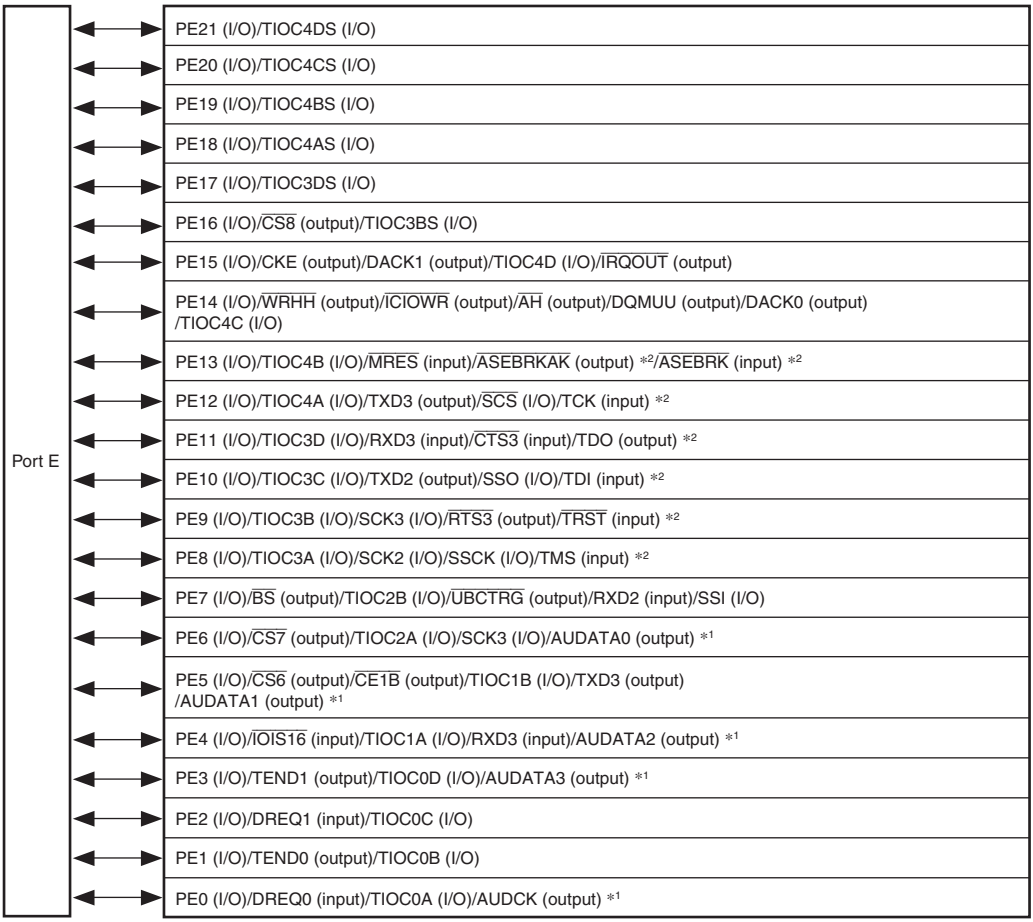
Port E in the SH7085 is an input/output port with the 16 pins shown in figure 22.13.



Notes: 1. Only in F-ZTAT version supporting full functions of E10A.
2. Only in F-ZTAT version.

Figure 22.13 Port E (SH7085)

Port E in the SH7086 is an input/output port with the 22 pins shown in figure 22.14.



Notes: 1. Only in F-ZTAT version supporting full functions of E10A.
2. Only in F-ZTAT version.

Figure 22.14 Port E (SH7086)

22.5.1 Register Descriptions

Port E is a 13-bit input/output port in the SH7083; a 16-bit input/output port in the SH7084 and SH7085; a 22-bit input/output port in the SH7086. Port E has the following registers. For details on register addresses and register states during each processing, refer to section 27, List of Registers.

Table 22.9 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Port E data register H	PEDRH	R/W	H'0000	H'FFFFD300	8, 16, 32
Port E data register L	PEDRL	R/W	H'0000	H'FFFFD302	8, 16
Port E port register H	PEPRH	R	H'xxxx	H'FFFFD31C	8, 16, 32
Port E port register L	PEPRL	R	H'xxxx	H'FFFFD31E	8, 16

22.5.2 Port E Data Registers H and L (PEDRH and PEDRL)

The port E data registers H and L (PEDRH and PEDRL) are 16-bit readable/writable registers that store port E data. Bits PE15DR to PE12DR, PE10DR, PE8DR to PE6DR, and PE4DR to PE0DR correspond to pins PE15 to PE12, PE10, PE8 to PE6, and PE4 to PE0, respectively (multiplexed functions omitted here) in the SH7083. Bits PE15DR to PE0DR correspond to pins PE15 to PE0 (multiplexed functions omitted here) in the SH7084 and SH7085. Bits PE21DR to PE0DR correspond to pins PE21 to PE0, respectively (multiplexed functions omitted here) in the SH7086.

When a pin function is general output, if a value is written to PEDRH or PEDRL, that value is output directly from the pin, and if PEDRH or PEDRL is read, the register value is returned directly regardless of the pin state.

When a pin function is general input, if PEDRH or PEDRL is read, the pin state, not the register value, is returned directly. If a value is written to PEDRH or PEDRL, although that value is written into PEDRH or PEDRL, it does not affect the pin state. Table 22.10 summarizes port E data register read/write operations.

- PEDRH (SH7083, SH7084, SH7085)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- PEDRH (SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	PE21 DR	PE20 DR	PE19 DR	PE18 DR	PE17 DR	PE16 DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PE21DR	0	R/W	See table 22.10.
4	PE20DR	0	R/W	
3	PE19DR	0	R/W	
2	PE18DR	0	R/W	
1	PE17DR	0	R/W	
0	PE16DR	0	R/W	

- PEDRL (SH7083)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15DR	PE14DR	PE13DR	PE12DR	-	PE10DR	-	PE8DR	PE7DR	PE6DR	-	PE4DR	PE3DR	PE2DR	PE1DR	PE0DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PE15DR	0	R/W	See table 22.10.
14	PE14DR	0	R/W	
13	PE13DR	0	R/W	
12	PE12DR	0	R/W	
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PE10DR	0	R/W	See table 22.10.
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	PE8DR	0	R/W	See table 22.10.
7	PE7DR	0	R/W	
6	PE6DR	0	R/W	
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4	PE4DR	0	R/W	See table 22.10.
3	PE3DR	0	R/W	
2	PE2DR	0	R/W	
1	PE1DR	0	R/W	
0	PE0DR	0	R/W	

- PEDRL (SH7084, SH7085, SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15DR	PE14DR	PE13DR	PE12DR	PE11DR	PE10DR	PE9DR	PE8DR	PE7DR	PE6DR	PE5DR	PE4DR	PE3DR	PE2DR	PE1DR	PE0DR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PE15DR	0	R/W	See table 22.10.
14	PE14DR	0	R/W	
13	PE13DR	0	R/W	
12	PE12DR	0	R/W	
11	PE11DR	0	R/W	
10	PE10DR	0	R/W	
9	PE9DR	0	R/W	
8	PE8DR	0	R/W	
7	PE7DR	0	R/W	
6	PE6DR	0	R/W	
5	PE5DR	0	R/W	
4	PE4DR	0	R/W	
3	PE3DR	0	R/W	
2	PE2DR	0	R/W	
1	PE1DR	0	R/W	
0	PE0DR	0	R/W	

Table 22.10 Port E Data Register (PEDR) Read/Write Operations

- PEDRH Bits 5 to 0 and PEDRL Bits 15 to 0

PEIOR	Pin Function	Read	Write
0	General input	Pin state	Can write to PEDRH and PEDRL, but it has no effect on pin state
	Other than general input	Pin state	Can write to PEDRH and PEDRL, but it has no effect on pin state
1	General output	PEDRH or PEDRL value	Value written is output from pin
	Other than general output	PEDRH or PEDRL value	Can write to PEDRH and PEDRL, but it has no effect on pin state

22.5.3 Port E Port Registers H and L (PEPRH and PEPRL)

The port E port registers H and L (PEPRH and PEPRL) are 16-bit read-only registers that always return the states of the pins regardless of the PFC setting. Bits PE15PR to PE12PR, PE10PR, PE8PR to PE6PR, and PE4PR to PE0PR correspond to pins PE15 to PE12, PE10, PE8 to PE6, and PE4 to PE0, respectively (multiplexed functions omitted here) in the SH7083. Bits PE15PR to PE0PR correspond to pins PE15 to PE0 (multiplexed functions omitted here) in the SH7084 and SH7085. Bits PE21PR to PE0PR correspond to pins PE21 to PE0, respectively (multiplexed functions omitted here) in the SH7086.

- PEPRH (SH7083, SH7084, SH7085)

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

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- PEPRH (SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	PE21PR	PE20PR	PE19PR	PE18PR	PE17PR	PE16PR
Initial value:	0	0	0	0	0	0	0	0	0	0	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	PE21PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
4	PE20PR	Pin state	R	
3	PE19PR	Pin state	R	
2	PE18PR	Pin state	R	
1	PE17PR	Pin state	R	
0	PE16PR	Pin state	R	

- PEPRL (SH7083)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15PR	PE14PR	PE13PR	PE12PR	-	PE10PR	-	PE8PR	PE7PR	PE6PR	-	PE4PR	PE3PR	PE2PR	PE1PR	PE0PR
Initial value:	*	*	*	*	0	*	0	*	*	*	0	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	PE15PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
14	PE14PR	Pin state	R	
13	PE13PR	Pin state	R	
12	PE12PR	Pin state	R	
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PE10PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	PE8PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
7	PE7PR	Pin state	R	
6	PE6PR	Pin state	R	
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4	PE4PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
3	PE3PR	Pin state	R	
2	PE2PR	Pin state	R	
1	PE1PR	Pin state	R	
0	PE0PR	Pin state	R	

- PEPRL (SH7084, SH7085, SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15PR	PE14PR	PE13PR	PE12PR	PE11PR	PE10PR	PE9PR	PE8PR	PE7PR	PE6PR	PE5PR	PE4PR	PE3PR	PE2PR	PE1PR	PE0PR
Initial value:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	PE15PR	Pin state	R	The pin state is returned regardless of the PFC setting. These bits cannot be modified.
14	PE14PR	Pin state	R	
13	PE13PR	Pin state	R	
12	PE12PR	Pin state	R	
11	PE11PR	Pin state	R	
10	PE10PR	Pin state	R	
9	PE9PR	Pin state	R	
8	PE8PR	Pin state	R	
7	PE7PR	Pin state	R	
6	PE6PR	Pin state	R	
5	PE5PR	Pin state	R	
4	PE4PR	Pin state	R	
3	PE3PR	Pin state	R	
2	PE2PR	Pin state	R	
1	PE1PR	Pin state	R	
0	PE0PR	Pin state	R	

22.6 Port F

Port F in the SH7083, SH7084, and SH7085 is an input-only port with the 8 pins shown in figure 22.15.

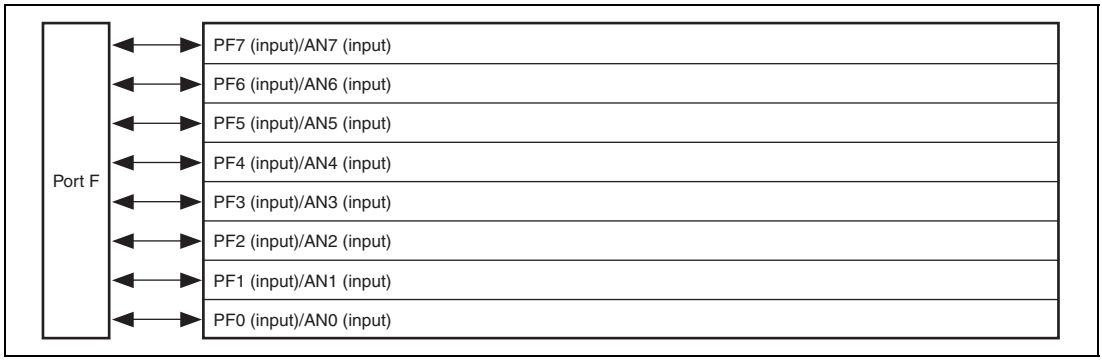


Figure 22.15 Port F (SH7083, SH7084, SH7085)

Port F in the SH7086 is an input-only port with the 16 pins shown in figure 22.16.

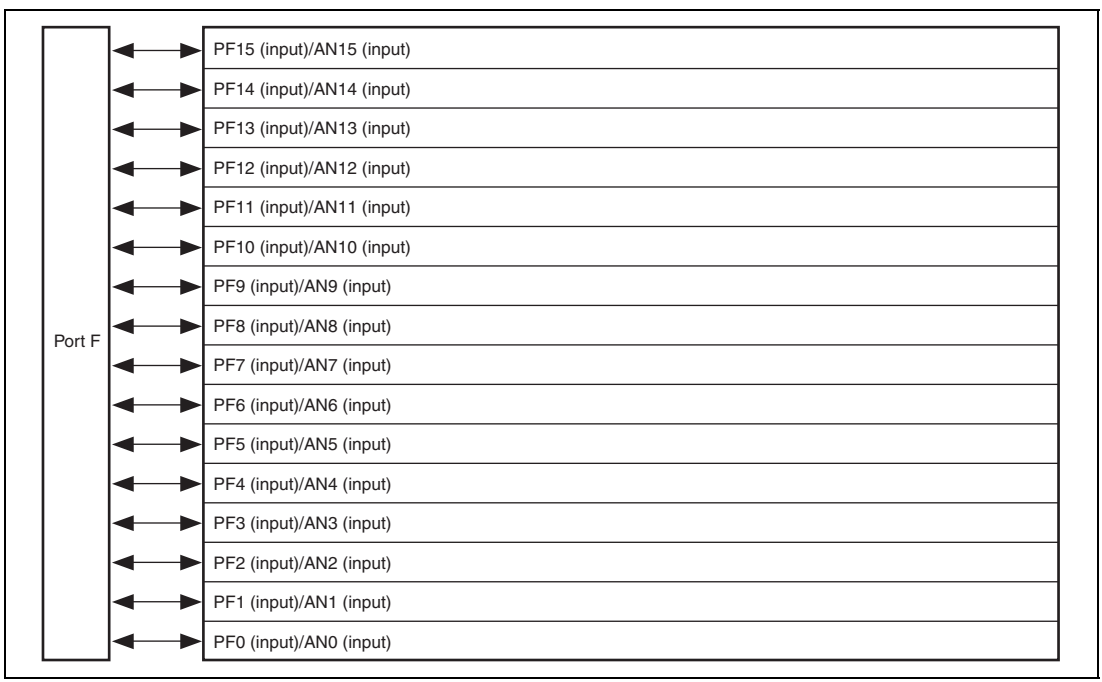


Figure 22.16 Port F (SH7086)

22.6.1 Register Descriptions

Port F is an 8-bit input-only port in the SH7083, SH7084, and SH7085; a 16-bit input-only port in the SH7086. Port F has the following register. For details on register addresses and register states during each processing, refer to section 27, List of Registers.

Table 22.11 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Port F data register L	PFDRL	R	H'xxxx	H'FFFFD382	8, 16

22.6.2 Port F Data Register L (PFDRL)

The port F data register L (PFDRL) is a 16-bit read-only register that stores port F data. Bits PF7DR to PF0DR correspond to pins PF7 to PF0 (multiplexed functions omitted here) in the SH7083, SH7084, and SH7085. Bits PF15DR to PF0DR correspond to pins PF15 to PF0 (multiplexed functions omitted here) in the SH7086.

Any value written into these bits is ignored, and there is no effect on the state of the pins. When any of the bits are read, the pin state rather than the bit value is read directly. However, when an A/D converter analog input is being sampled, values of 1 are read out. Table 22.12 summarizes port F data register L read/write operations.

- PFDRL (SH7083, SH7084, SH7085)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	PF7 DR	PF6 DR	PF5 DR	PF4 DR	PF3 DR	PF2 DR	PF1 DR	PF0 DR
Initial value:	0	0	0	0	0	0	0	0	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	PF7DR	Pin state	R/W	See table 22.12.
6	PF6DR	Pin state	R/W	
5	PF5DR	Pin state	R/W	
4	PF4DR	Pin state	R/W	
3	PF3DR	Pin state	R/W	
2	PF2DR	Pin state	R/W	
1	PF1DR	Pin state	R/W	
0	PF0DR	Pin state	R/W	

- PFDRL (SH7086)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PF15 DR	PF14 DR	PF13 DR	PF12 DR	PF11 DR	PF10 DR	PF9 DR	PF8 DR	PF7 DR	PF6 DR	PF5 DR	PF4 DR	PF3 DR	PF2 DR	PF1 DR	PF0 DR
Initial value:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	PF15DR	Pin state	R	See table 22.12.
14	PF14DR	Pin state	R	
13	PF13DR	Pin state	R	
12	PF12DR	Pin state	R	
11	PF11DR	Pin state	R	
10	PF10DR	Pin state	R	
9	PF9DR	Pin state	R	
8	PF8DR	Pin state	R	
7	PF7DR	Pin state	R	
6	PF6DR	Pin state	R	
5	PF5DR	Pin state	R	
4	PF4DR	Pin state	R	
3	PF3DR	Pin state	R	
2	PF2DR	Pin state	R	
1	PF1DR	Pin state	R	
0	PF0DR	Pin state	R	

Table 22.12 Port F Data Register L (PFDRL) Read/Write Operations

- PFDRL Bits 15 to 0

Pin Function	Read	Write
General input	Pin state	Ignored (no effect on pin state)
ANn input (analog input)	1	Ignored (no effect on pin state)

Section 23 Flash Memory

This LSI has 512-kbyte or 256-kbyte on-chip flash memory. The flash memory has the following features.

23.1 Features

- Two flash-memory MATs, with one selected by the mode in which the LSI starts up

  The on-chip flash memory has two memory spaces in the same address space (hereafter referred to as memory MATs). The mode setting when the LSI starts up determines the memory MAT that is currently mapped. The MAT can be switched by bank-switching after the LSI has started up.

  — Size of the user MAT, from which booting-up proceeds after a power-on reset in user mode: 512 kbytes or 256 kbytes

  — Size of the user boot MAT, from which booting-up proceeds after a power-on reset in user boot mode: 12 kbytes

- Three on-board programming modes and one off-board programming mode

  On-board programming modes

  Boot Mode: The on-chip SCI interface is used for programming in this mode. Either the user MAT or user-boot MAT can be programmed, and the bit rate for data transfer between the host and this LSI are automatically adjusted.

  User Program Mode: This mode allows programming of the user MAT via any desired interface.

  User Boot Mode: This mode allows writing of a user boot program via any desired interface and programming of the user MAT.

  Off-board programming mode

  Programmer Mode: This mode allows programming of the user MAT and user boot MAT with the aid of a PROM programmer.

- Downloading of an on-chip program to provide an interface for programming/erasure

  This LSI has a dedicated programming/erasing program. After this program has been downloaded to the on-chip RAM, programming or erasing can be performed by setting parameters as arguments. “User branching” is also supported.

— User branching

Programming is performed in 128-byte units. Each round of programming consists of application of the programming pulse, reading for verification, and several other steps. Erasing is performed in block units and each round of erasing consists of several steps. A user-processing routine can be executed between each round of erasing, and making the setting for this is called the addition of a user branch.

- Using on-chip RAM to emulate flash memory

  By laying on-chip RAM over part of the flash memory, flash-memory programming can be emulated in real time.

- Protection modes

  There are two modes of protection: software protection is applied by register settings and hardware protection is applied by the level on the FWE pin. Protection of the flash memory from programming or erasure can be selected.

  When an abnormal state is detected, such as runaway execution of programming/erasing, the protection modes initiate the transition to the error protection state and suspend programming/erasing processing.

- Programming/erasing time

  The time taken to program 128 bytes of flash memory in a single round is t_P ms (typ.), which is equivalent to t_P/128 ms per byte. The erasing time is t_Es (typ.) per block.

- Number of programming operations

  The flash memory can be programmed up to N_{WEC} times.

- Operating frequency for programming/erasing

  The operating frequency for programming/erasing is a maximum of 40 MHz (Pϕ).

23.2 Overview

23.2.1 Block Diagram

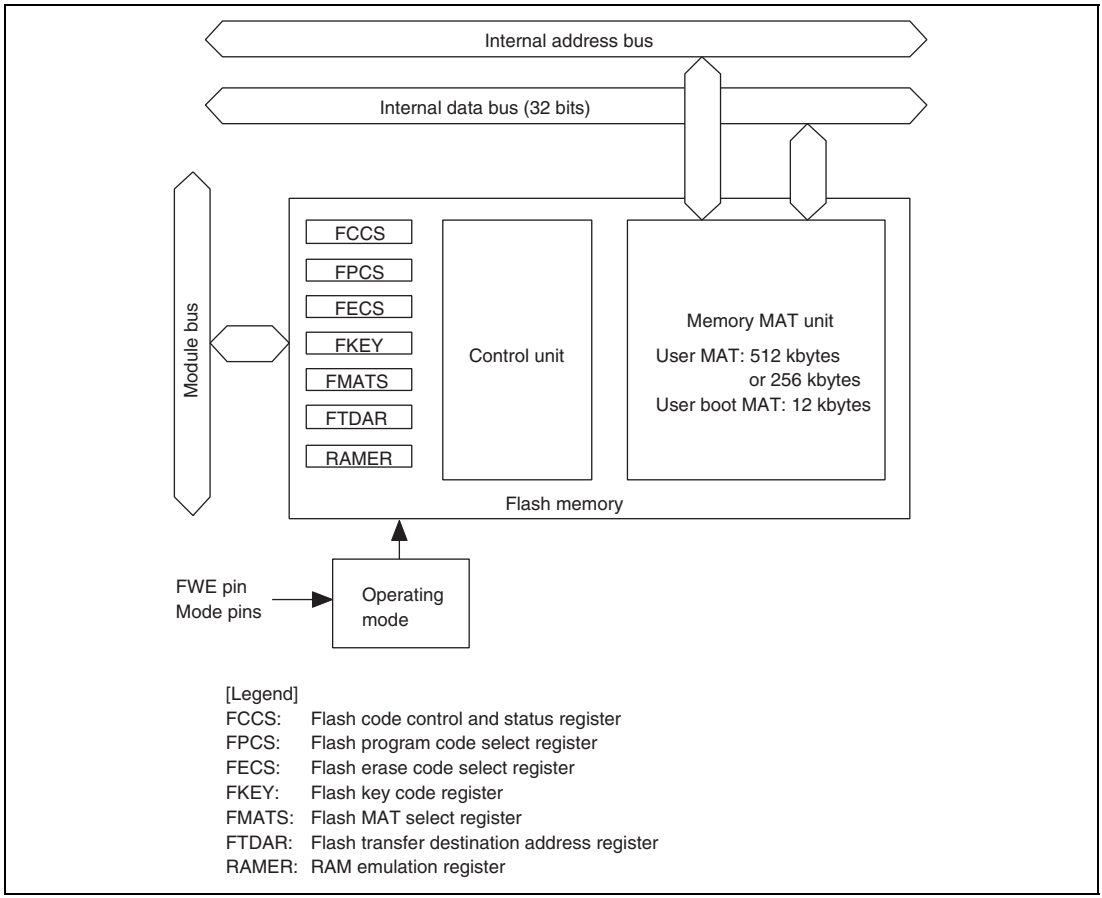


Figure 23.1 Block Diagram of Flash Memory

23.2.2 Operating Mode

When each mode pin and the FWE pin are set in the reset state and the reset signal is released, the microcomputer enters each operating mode as shown in figure 23.2. For the setting of each mode pin and the FWE pin, see table 23.1.

- Flash memory cannot be read, programmed, or erased in ROM invalid mode. The programming/erasing interface registers cannot be written to. When these registers are read, H'00 is always read.
- Flash memory can be read in user mode, but cannot be programmed or erased.
- Flash memory can be read, programmed, or erased on the board only in user program mode, user boot mode, and boot mode.
- Flash memory can be read, programmed, or erased by means of the PROM programmer in programmer mode.

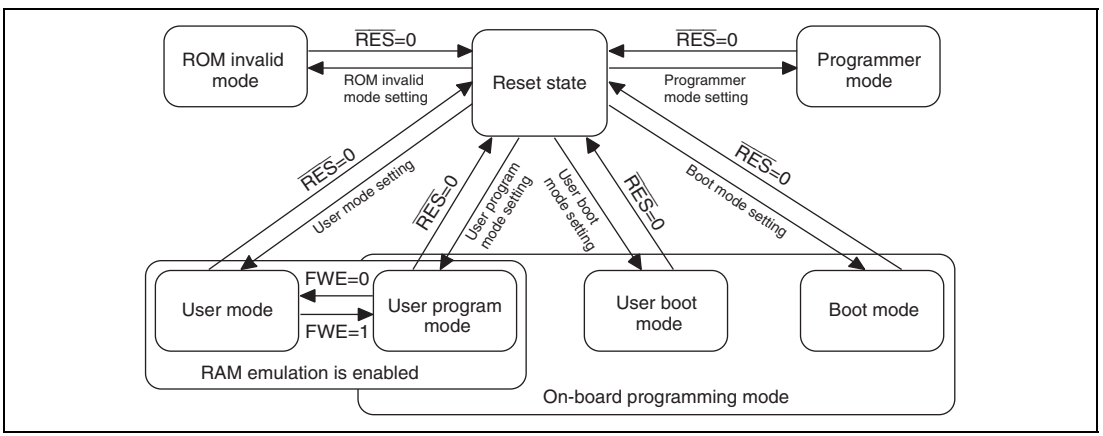


Figure 23.2 Mode Transition of Flash Memory

Table 23.1 (1) Relationship between FWE and MD Pins and Operating Modes (SH7083 and SH7084)

Pin	Reset State	ROM Invalid Mode	ROM Valid Mode	User Program Mode	User Boot Mode	Boot Mode	Programmer Mode
$\overline{RES}$	0	1	1	1	1	1	Setting value depends on the condition of the specialized PROM programmer.
FWE	0/1	0	0	1	1	1	
MD0	0/1	0/1*1	0/1*2	0/1*2	1	0	
MD1	0/1	0	1	1	0	0	

Notes: 1. MD0 = 0: 8-bit external bus, MD0 = 1: 16-bit external bus
2. MD0 = 0: External bus can be used, MD0 = 1: Single-chip mode (external bus cannot be used)

Table 23.1 (2) Relationship between FWE and MD Pins and Operating Modes (SH7085 and SH7086)

Pin	Reset State	ROM Invalid Mode	ROM Valid Mode	User Program Mode	User Boot Mode	Boot Mode	Programmer Mode
$\overline{RES}$	0	1	1	1	1	1	Setting value depends on the condition of the specialized PROM programmer.
FWE	0/1	0	0	1	1	1	
MD0	0/1	0/1*1	0/1*2	0/1*2	1	0	
MD1	0/1	0	1	1	0	0	

Notes: 1. MD0 = 0: 16-bit external bus, MD0 = 1: 32-bit external bus
2. MD0 = 0: External bus can be used, MD0 = 1: Single-chip mode (external bus cannot be used)

23.2.3 Mode Comparison

The comparison table of programming and erasing related items about boot mode, user program mode, user boot mode, and programmer mode is shown in table 23.2.

Table 23.2 Comparison of Programming Modes

	Boot Mode	User Program Mode	User Boot Mode	Programmer Mode
Programming/ erasing environment	On-board programming	On-board programming	On-board programming	Off-board programming
Programming/ erasing enable MAT	User MAT User boot MAT	User MAT	User MAT	User MAT User boot MAT
Programming/ erasing control	Command method	Programming/ erasing interface	Programming/ erasing interface	—
All erasure	Possible (Automatic)	Possible	Possible	Possible (Automatic)
Block division erasure	Possible*1	Possible	Possible	Not possible
Program data transfer	From host via SCI	From optional device via RAM	From optional device via RAM	Via programmer
User branch function	Not possible	Possible	Possible	Not possible
RAM emulation	Not possible	Possible	Not possible	Not possible
Reset initiation MAT	Embedded program storage MAT	User MAT	User boot MAT*2	Embedded program storage MAT
Transition to user mode	Mode setting change and reset	FWE setting change	Mode setting change and reset	—

Notes: 1. All-erasure is performed. After that, the specified block can be erased.
2. Initiation starts from the embedded program storage MAT. After checking the flash-memory related registers, initiation starts from the reset vector of the user MAT.

- The user boot MAT can be programmed or erased only in boot mode and programmer mode.
- The user MAT and user boot MAT are all erased in boot mode. Then, the user MAT and user boot MAT can be programmed by means of the command method. However, the contents of the MAT cannot be read until this state.
  Only user boot MAT is programmed and the user MAT is programmed in user boot mode or only user MAT is programmed because user boot mode is not used.
- In user boot mode, the boot operation of the optional interface can be performed by a mode pin setting different from user program mode.

23.2.4 Flash Memory Configuration

This LSI's flash memory is configured by the 512-kbyte or 256-kbyte user MAT and 12-kbyte user boot MAT.

The start address is allocated to the same address in the user MAT and user boot MAT. Therefore, when the program execution or data access is performed between the two MATs, the MAT must be switched by using FMATS.

The user MAT or user boot MAT can be read in all modes if it is in ROM valid mode. However, the user boot MAT can be programmed only in boot mode and programmer mode.

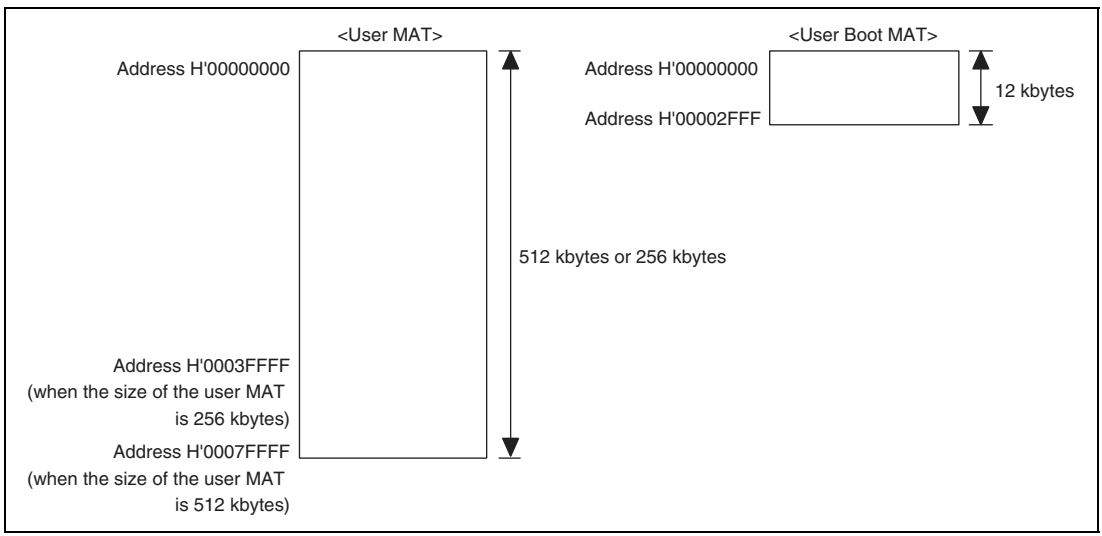


Figure 23.3 Flash Memory Configuration

The user MAT and user boot MAT have different memory sizes. Do not access a user boot MAT that is 12 kbytes or more. When a user boot MAT exceeding 12 kbytes is read from, an undefined value is read.

23.2.5 Block Division

The user MAT is divided into 64 kbytes (512-kbyte version: seven blocks, 256-kbyte version: three blocks), 32 kbytes (one block), and 4 kbytes (eight blocks) as shown in figure 23.4. The user MAT can be erased in this divided-block units and the erase-block number of EB0 to EB15 is specified when erasing.

The RAM emulation can be performed in the eight blocks of 4 kbytes.

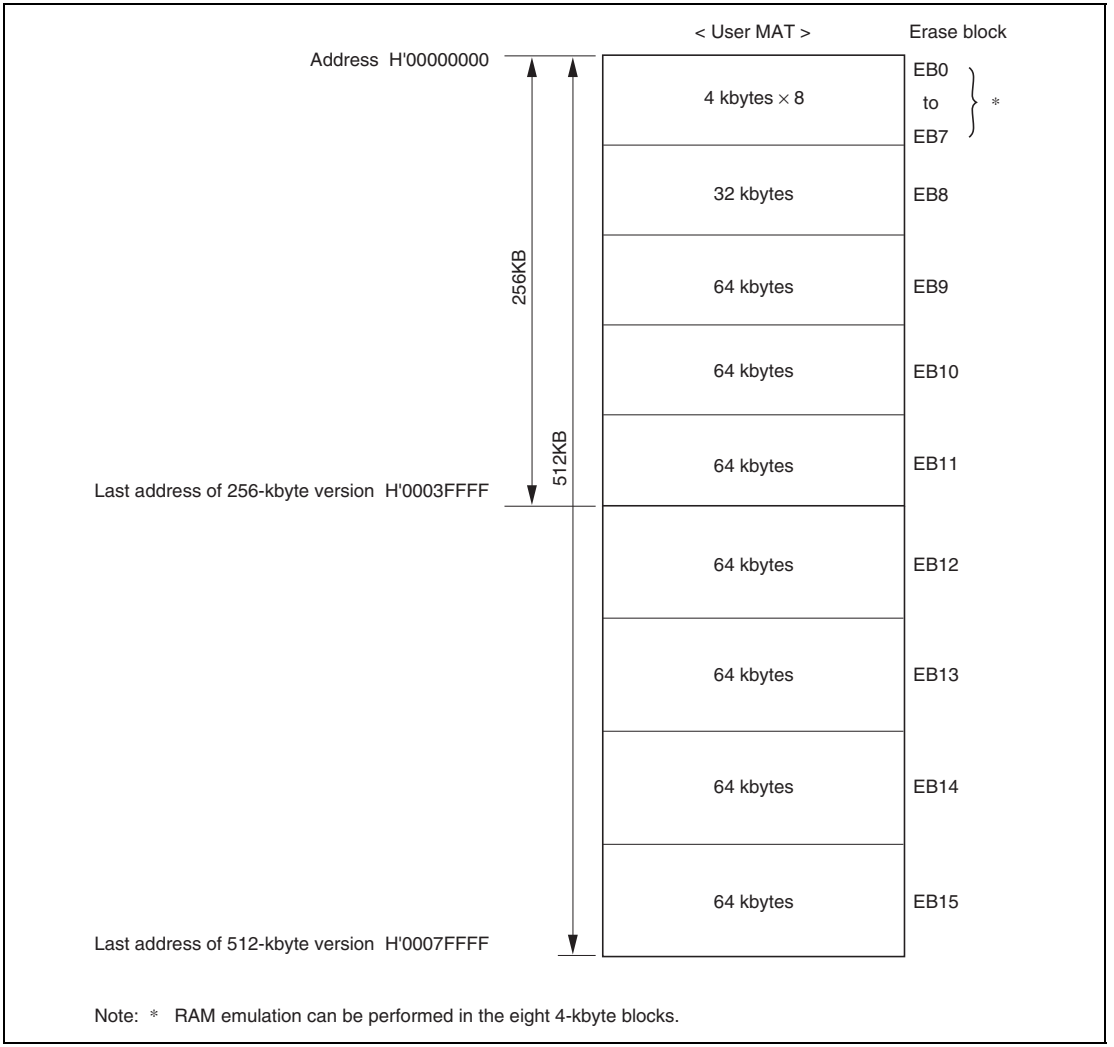


Figure 23.4 Block Division of User MAT

23.2.6 Programming/Erasing Interface

Programming/erasing is executed by downloading the on-chip program to the on-chip RAM and specifying the program address/data and erase block by using the interface registers/parameters.

The procedure program is made by the user in user program mode and user boot mode. The overview of the procedure is as follows. For details, see section 23.5.2, User Program Mode.

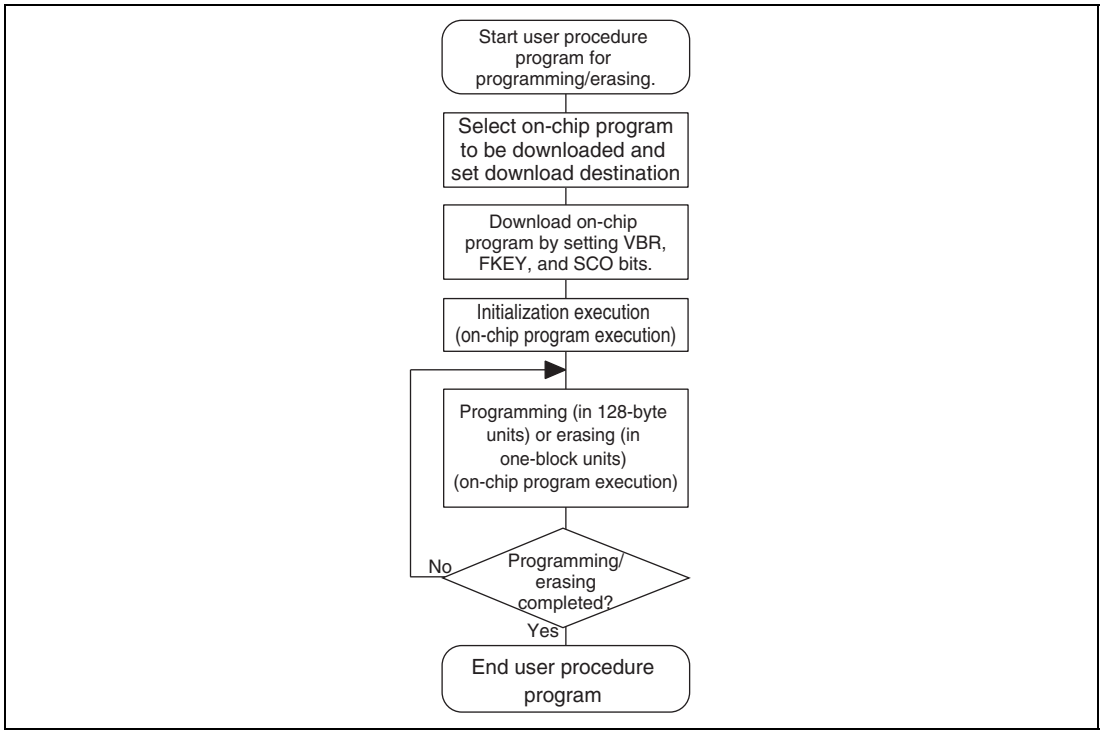


Figure 23.5 Overview of User Procedure Program

(1) Selection of On-Chip Program to be Downloaded and Setting of Download Destination

This LSI has programming/erasing programs and they can be downloaded to the on-chip RAM. The on-chip program to be downloaded is selected by setting the corresponding bits in the programming/erasing interface registers. The download destination can be specified by FTDAR.

(2) Download of On-Chip Program

The on-chip program is automatically downloaded by clearing VBR of the CPU to H'84000000 and then setting the SCO bit in the flash code control and status register (FCCS) and the flash key code register (FKEY), which are programming/erasing interface registers.

The user MAT is replaced to the embedded program storage area when downloading. Since the flash memory cannot be read when programming/erasing, the procedure program, which is working from download to completion of programming/erasing, must be executed in a space other than the flash memory to be programmed/erased (for example, on-chip RAM).

Since the result of download is returned to the programming/erasing interface parameters, whether the normal download is executed or not can be confirmed.

Note that VBR can be changed after download is completed.

(3) Initialization of Programming/Erasing

The operating frequency and user branch are set before execution of programming/erasing. The user branch destination must be in an area other than the user MAT area which is in the middle of programming and the area where the on-chip program is downloaded. These settings are performed by using the programming/erasing interface parameters.

(4) Programming/Erasing Execution

To program or erase, the FWE pin must be brought high and user program mode must be entered.

The program data/programming destination address is specified in 128-byte units when programming.

The block to be erased is specified in erase-block units when erasing.

These specifications are set by using the programming/erasing interface parameters and the on-chip program is initiated. The on-chip program is executed by using the JSR or BSR instruction to perform the subroutine call of the specified address in the on-chip RAM. The execution result is returned to the programming/erasing interface parameters.

The area to be programmed must be erased in advance when programming flash memory.

Ensure that no interrupts, including NMI and IRQ, are generated during programming or erasure.

(5) When Programming/Erasing is Executed Consecutively

When the processing is not ended by the 128-byte programming or one-block erasure, the program address/data and erase-block number must be updated and consecutive programming/erasing is required.

Since the downloaded on-chip program is left in the on-chip RAM after the processing, download and initialization are not required when the same processing is executed consecutively.

23.3 Input/Output Pins

Flash memory is controlled by the pins as shown in table 23.3.

Table 23.3 Pin Configuration

Pin Name	Symbol	Input/Output	Function
Power-on reset	$\overline{\text{RES}}$	Input	Reset
Flash programming enable	FWE	Input	Hardware protection when programming flash memory
Mode 1	MD1	Input	Sets operating mode of this LSI
Mode 0	MD0	Input	Sets operating mode of this LSI
Transmit data	TXD1 (PA4)	Output	Serial transmit data output (used in boot mode)
Receive data	RXD1 (PA3)	Input	Serial receive data input (used in boot mode)

23.4 Register Descriptions

23.4.1 Registers

The registers/parameters which control flash memory when the on-chip flash memory is valid are shown in table 23.4.

There are several operating modes for accessing flash memory, for example, read mode/program mode.

There are two memory MATs: user MAT and user boot MAT. The dedicated registers/parameters are allocated for each operating mode and MAT selection. The correspondence of operating modes and registers/parameters for use is shown in table 23.5.

Table 23.4 (1) Register Configuration

Register Name	Abbreviation*4	R/W	Initial Value	Address	Access Size
Flash code control and status register	FCCS	R, W*1	H'00*2 H'80*2	H'FFFFCC00	8
Flash program code select register	FPCS	R/W	H'00	H'FFFFCC01	8
Flash erase code select register	FECS	R/W	H'00	H'FFFFCC02	8
Flash key code register	FKEY	R/W	H'00	H'FFFFCC04	8
Flash MAT select register	FMATS	R/W	H'00*3 H'AA*3	H'FFFFCC05	8
Flash transfer destination address register	FTDAR	R/W	H'00	H'FFFFCC06	8
RAM emulation register	RAMER	R/W	H'0000	H'FFFFF108	16

Notes: 1. The bits except the SCO bit are read-only bits. The SCO bit is a programming-only bit. (The value which can be read is always 0.)
2. The initial value of the FWE bit is 0 when the FWE pin goes low. The initial value of the FWE bit is 1 when the FWE pin goes high.
3. The initial value at initiation in user mode or user program mode is H'00. The initial value at initiation in user boot mode is H'AA.
4. All registers except for RAMER can be accessed only in bytes. RAMER can be accessed in words.

Table 23.4 (2) Parameter Configuration

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Download pass/fail result	DPFR	R/W	Undefined	On-chip RAM*	8, 16, 32
Flash pass/fail result	FPFR	R/W	Undefined	R0 of CPU	8, 16, 32
Flash multipurpose address area	FMPAR	R/W	Undefined	R5 of CPU	8, 16, 32
Flash multipurpose data destination area	FMPDR	R/W	Undefined	R4 of CPU	8, 16, 32
Flash erase block select	FEBS	R/W	Undefined	R4 of CPU	8, 16, 32
Flash program and erase frequency control	FPEFEQ	R/W	Undefined	R4 of CPU	8, 16, 32
Flash user branch address set parameter	FUBRA	R/W	Undefined	R5 of CPU	8, 16, 32

Note: * One byte of the start address in the on-chip RAM area specified by FTDAR is valid.

Table 23.5 Register/Parameter and Target Mode

		Download	Initialization	Programming	Erasure	Read	RAM Emulation
Programming/ erasing interface registers	FCCS	√	—	—	—	—	—
	FPCS	√	—	—	—	—	—
	PECS	√	—	—	—	—	—
	FKEY	√	—	√	√	—	—
	FMATS	—	—	√*1	√*1	√*2	—
	FTDAR	√	—	—	—	—	—
Programming/ erasing interface parameters	DPFR	√	—	—	—	—	—
	FPFR	—	√	√	√	—	—
	FPEFEQ	—	√	—	—	—	—
	FUBRA	—	√	—	—	—	—
	FMPAR	—	—	√	—	—	—
	FMPDR	—	—	√	—	—	—
	FEBS	—	—	—	√	—	—
RAM emulation	RAMER	—	—	—	—	—	√

Notes: 1. The setting is required when programming or erasing user MAT in user boot mode.
2. The setting may be required according to the combination of initiation mode and read target MAT.

23.4.2 Programming/Erasing Interface Registers

The programming/erasing interface registers are as described below. They are all 8-bit registers that can be accessed in bytes.

(1) Flash Code Control and Status Register (FCCS)

FCCS is configured by bits which request the monitor of the FWE pin state and error occurrence during programming or erasing flash memory and the download of the on-chip program.

Bit:	7	6	5	4	3	2	1	0
	FWE	MAT	-	FLER	-	-	-	SCO
Initial value:	1/0	1/0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	(R)/W

Bit	Bit Name	Initial Value	R/W	Description
7	FWE	1/0	R	Flash Programming Enable Monitors the level which is input to the FWE pin that performs hardware protection of the flash memory programming or erasing. The initial value is 0 or 1 according to the FWE pin state. 0: When the FWE pin goes low (in hardware protection state) 1: When the FWE pin goes high
6	MAT	1/0	R	MAT Bit Indicates whether the user MAT or user boot MAT is selected. 0: User MAT is selected 1: User boot MAT is selected
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	FLER	0	R	Flash Memory Error Indicates an error occurs during programming and erasing flash memory. When FLER is set to 1, flash memory enters the error protection state. When FLER is set to 1, high voltage is applied to the internal flash memory. To reduce the damage to flash memory, the reset signal must be released after the reset period of 100 μs which is longer than normal. 0: Flash memory operates normally Programming/erasing protection for flash memory (error protection) is invalid. [Clearing condition] At a power-on reset 1: Indicates an error occurs during programming/erasing flash memory. Programming/erasing protection for flash memory (error protection) is valid. [Setting condition] See section 23.6.3, Error Protection.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	SCO	0	(R)/W	Source Program Copy Operation Requests the on-chip programming/erasing program to be downloaded to the on-chip RAM. When this bit is set to 1, the on-chip program which is selected by FPCS/FECS is automatically downloaded in the on-chip RAM area specified by FTDAR. In order to set this bit to 1, RAM emulation state must be canceled, H'A5 must be written to FKEY, and this operation must be in the on-chip RAM. Four NOP instructions must be executed immediately after setting this bit to 1. For interrupts during download, see section 23.8.2, Interrupts during Programming/Erasing. For the download time, see section 23.8.3, Other Notes. Since this bit is cleared to 0 when download is completed, this bit cannot be read as 1. Download by setting the SCO bit to 1 requires a special interrupt processing that performs bank switching to the on-chip program storage area. Therefore, before issuing a download request (SCO = 1), set VBR to H'84000000. Otherwise, the CPU gets out of control. Once download end is confirmed, VBR can be changed to any other value. The mode in which the FWE pin is high must be used when using the SCO function. 0: Download of the on-chip programming/erasing program to the on-chip RAM is not executed. [Clearing condition] When download is completed 1: Request that the on-chip programming/erasing program is downloaded to the on-chip RAM is generated [Setting conditions] When all of the following conditions are satisfied and 1 is written to this bit • FKEY is written to H'A5 • During execution in the on-chip RAM • Not in RAM emulation mode (RAMS in RAMCR = 0)

(2) Flash Program Code Select Register (FPCS)

FPCS selects the on-chip programming program to be downloaded.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	PPVS
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PPVS	0	R/W	Program Pulse Single Selects the programming program. 0: On-chip programming program is not selected [Clearing condition] When transfer is completed 1: On-chip programming program is selected

(3) Flash Erase Code Select Register (FECS)

FECS selects download of the on-chip erasing program.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	EPVB
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	EPVB	0	R/W	Erase Pulse Verify Block Selects the erasing program. 0: On-chip erasing program is not selected [Clearing condition] When transfer is completed 1: On-chip erasing program is selected

(4) Flash Key Code Register (FKEY)

FKEY is a register for software protection that enables download of the on-chip program and programming/erasing of flash memory. Before setting the SCO bit to 1 in order to download the on-chip program or executing the downloaded programming/erasing program, these processings cannot be executed if the key code is not written.

Bit:	7	6	5	4	3	2	1	0
	K[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	K[7:0]	All 0	R/W	Key Code Only when H'A5 is written, writing to the SCO bit is valid. When a value other than H'A5 is written to FKEY, 1 cannot be written to the SCO bit. Therefore downloading to the on-chip RAM cannot be executed. Only when H'5A is written, programming/erasing of flash memory can be executed. Even if the on-chip programming/erasing program is executed, flash memory cannot be programmed or erased when a value other than H'5A is written to FKEY. H'A5: Writing to the SCO bit is enabled (The SCO bit cannot be set by a value other than H'A5.) H'5A: Programming/erasing is enabled (A value other than H'5A enables software protection state.) H'00: Initial value

(5) Flash MAT Select Register (FMATS)

FMATS specifies whether user MAT or user boot MAT is selected.

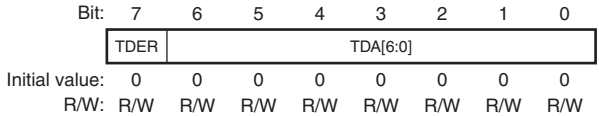
Bit:	7	6	5	4	3	2	1	0
	MS7	MS6	MS5	MS4	MS3	MS2	MS1	MS0
Initial value:	0/1	0	0/1	0	0/1	0	0/1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MS7	0/1	R/W	MAT Select
6	MS6	0	R/W	These bits are in user-MAT selection state when a value other than H'AA is written and in user-boot-MAT selection state when H'AA is written.
5	MS5	0/1	R/W	
4	MS4	0	R/W	The MAT is switched by writing a value in FMATS with the on-chip RAM instrunction.
3	MS3	0/1	R/W	
2	MS2	0	R/W	When the MAT is switched, follow section 23.8.1, Switching between User MAT and User Boot MAT. (The user boot MAT cannot be programmed in user program mode if user boot MAT is selected by FMATS. The user boot MAT must be programmed in boot mode or in programmer mode.)
1	MS1	0/1	R/W	
0	MS0	0	R/W	H'AA: The user boot MAT is selected (in user-MAT selection state when the value of these bits are other than H'AA) Initial value when these bits are initiated in user boot mode. H'00: Initial value when these bits are initiated in a mode except for user boot mode (in user-MAT selection state) [Programmable condition] These bits are in the execution state in the on-chip RAM.

(6) Flash Transfer Destination Address Register (FTDAR)

FTDAR specifies the on-chip RAM address to which the on-chip program is downloaded.

Make settings for FTDAR before writing 1 to the SCO bit in FCCS. The initial value is H'00 which points to the start address (H'FFFF9000) in on-chip RAM.



Bit	Bit Name	Initial Value	R/W	Description
7	TDER	0	R/W	Transfer Destination Address Setting Error This bit is set to 1 when there is an error in the download start address set by bits 6 to 0 (TDA6 to TDA0). Whether the address setting is erroneous or not is tested by checking whether the setting of TDA6 to TDA0 is in the range of H'00 to H'04 after setting the SCO bit in FCCS to 1 and performing download. Before setting the SCO bit to 1 be sure to set the FTDAR value between H'00 to H'04 as well as clearing this bit to 0. 0: Setting of TDA6 to TDA0 is normal 1: Setting of TDER and TDA6 to TDA0 is H'05 to H'FF and download has been aborted
6 to 0	TDA[6:0]	All 0	R/W	Transfer Destination Address These bits specify the download start address. A value from H'00 to H'04 can be set to specify the download start address in on-chip RAM in 2-kbyte units. A value from H'05 to H'7F cannot be set. If such a value is set, the TDER bit (bit 7) in this register is set to 1 to prevent download from being executed. H'00: Download start address is set to H'FFFF9000 H'01: Download start address is set to H'FFFF9800 H'02: Download start address is set to H'FFFFA000 H'03: Download start address is set to H'FFFFA800 H'04: Download start address is set to H'FFFFB000 H'05 to H'7F: Setting prohibited. If this value is set, the TDER bit (bit 7) is set to 1 to abort the download processing.

23.4.3 Programming/Erasing Interface Parameters

The programming/erasing interface parameters specify the operating frequency, user branch destination address, storage place for program data, programming destination address, and erase block and exchanges the processing result for the downloaded on-chip program. This parameter uses the general registers of the CPU (R4, R5, and R0) or the on-chip RAM area. The initial value is undefined.

At download all CPU registers are stored, and at initialization or when the on-chip program is executed, CPU registers except for R0 are stored. The return value of the processing result is written in R0. Since the stack area is used for storing the registers or as a work area, the stack area must be saved at the processing start. (The maximum size of a stack area to be used is 128 bytes.)

The programming/erasing interface parameters are used in the following four items.

1. Download control
2. Initialization before programming or erasing
3. Programming
4. Erasing

These items use different parameters. The correspondence table is shown in table 23.6.

The processing results of initialization, programming, and erasing are returned, but the bit contents have different meanings according to the processing program. See the description of FPFR for each processing.

Table 23.6 Usable Parameters and Target Modes

Name of Parameter	Abbreviation	Download	Initialization	Programming	Erasure	R/W	Initial Value	Allocation
Download pass/fail result	DPFR	√	—	—	—	R/W	Undefined	On-chip RAM*
Flash pass/fail result	FPFR	—	√	√	√	R/W	Undefined	R0 of CPU
Flash programming/ erasing frequency control	FPEFEQ	—	√	—	—	R/W	Undefined	R4 of CPU
Flash user branch address set	FUBRA	—	√	—	—	R/W	Undefined	R5 of CPU
Flash multipurpose address area	FMPAR	—	—	√	—	R/W	Undefined	R5 of CPU
Flash multipurpose data destination area	FMPDR	—	—	√	—	R/W	Undefined	R4 of CPU
Flash erase block select	FEBS	—	—	—	√	R/W	Undefined	R4 of CPU

Note: * One byte of start address of download destination specified by FTDAR

(1) Download Control

The on-chip program is automatically downloaded by setting the SCO bit to 1. The on-chip RAM area to be downloaded is the area as much as 3 kbytes starting from the start address specified by FTDAR. For the address map of the on-chip RAM, see figure 23.10.

The download control is set by using the programming/erasing interface registers. The return value is given by the DPFR parameter.

(a) Download pass/fail result parameter (DPFR: one byte of start address of on-chip RAM specified by FTDAR)

This parameter indicates the return value of the download result. The value of this parameter can be used to determine if downloading is executed or not. Since the confirmation whether the SCO bit is set to 1 is difficult, the certain determination must be performed by setting one byte of the start address of the on-chip RAM area specified by FTDAR to a value other than the return value of download (for example, H'FF) before the download start (before setting the SCO bit to 1). For the checking method of download results, see section 23.5.2 (2), Programming Procedure in User Program Mode.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	SS	FK	SF
Initial value:	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	Undefined	R/W	Unused Return 0.
2	SS	Undefined	R/W	Source Select Error Detect The on-chip program which can be downloaded can be specified as only one type. When more than two types of the program are selected, the program is not selected, or the program is selected without mapping, an error occurs. 0: Download program can be selected normally 1: Download error occurs (Multi-selection or program which is not mapped is selected)
1	FK	Undefined	R/W	Flash Key Register Error Detect Returns the check result whether the value of FKEY is set to H'A5. 0: FKEY setting is normal (FKEY = H'A5) 1: FKEY setting is abnormal (FKEY = value other than H'A5)
0	SF	Undefined	R/W	Success/Fail Returns the result whether download has ended normally or not. 0: Downloading on-chip program has ended normally (no error) 1: Downloading on-chip program has ended abnormally (error occurs)

(2) Programming/Erasing Initialization

The on-chip programming/erasing program to be downloaded includes the initialization program.

The specified period pulse must be applied when programming or erasing. The specified pulse width is made by the method in which wait loop is configured by the CPU instruction. The operating frequency of the CPU must be set. Since the user branch function is supported, the user branch destination address must be set.

The initial program is set as a parameter of the programming/erasing program which has downloaded these settings.

(2.1) Flash programming/erasing frequency parameter (FPEFEQ: general register R4 of CPU)

This parameter sets the operating frequency of the CPU.

For the range of the operating frequency of this LSI, see table 28.5, Maximum Operating Frequency.

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	F15	F14	F13	F12	F11	F10	F9	F8	F7	F6	F5	F4	F3	F2	F1	F0
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	Undefined	R/W	Unused Return 0.
15 to 0	F15 to F0	Undefined	R/W	Frequency Set Set the operating frequency of the CPU. The setting value must be calculated as the following methods. 1. The operating frequency which is shown in MHz units must be rounded in a number to three decimal places and be shown in a number of two decimal places. 2. The centuplicated value is converted to the binary digit and is written to the FPEFEQ parameter (general register R4). For example, when the operating frequency of the CPU is 28.882 MHz, the value is as follows. — The number to three decimal places of 28.882 is rounded and the value is thus 28.88. — The formula that 28.88 × 100 = 2888 is converted to the binary digit and B'0000, B'1011, B'0100, B'1000 (H'0B48) is set to B'R4.

(2.2) Flash user branch address setting parameter (FUBRA: general register R5 of CPU)

This parameter sets the user branch destination address. The user program which has been set can be executed in specified processing units when programming and erasing.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	UA31	UA30	UA29	UA28	UA27	UA26	UA25	UA24	UA23	UA22	UA21	UA20	UA19	UA18	UA17	UA16
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UA15	UA14	UA13	UA12	UA11	UA10	UA9	UA8	UA7	UA6	UA5	UA4	UA3	UA2	UA1	UA0
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	UA31 to UA0	Undefined	R/W	User Branch Destination Address When the user branch is not required, address 0 (H'00000000) must be set. The user branch destination must be an area other than the flash memory, an area other than the RAM area in which on-chip program has been transferred, or the external bus space. Note that the CPU must not branch to an area without the execution code and get out of control. The on-chip program download area and stack area must not be overwritten. If CPU runaway occurs or the download area or stack area is overwritten, the value of flash memory cannot be guaranteed. The download of the on-chip program, initialization, initiation of the programming/erasing program must not be executed in the processing of the user branch destination. Programming or erasing cannot be guaranteed when returning from the user branch destination. The program data which has already been prepared must not be programmed. Store general registers R8 to R15. General registers R0 to R7 are available without storing them. Moreover, the programming/erasing interface registers must not be written to or RAM emulation mode must not be entered in the processing of the user branch destination. After the processing of the user branch has ended, the programming/erasing program must be returned to by using the RTS instruction. For the execution intervals of the user branch processing, see note 2 (User branch processing intervals) in section 23.8.3, Other Notes.

(2.3) Flash pass/fail result parameter (FPFR: general register R0 of CPU)

This parameter indicates the return value of the initialization result.

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	BR	FQ	SF
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	Undefined	R/W	Unused Return 0.
2	BR	Undefined	R/W	User Branch Error Detect Returns the check result whether the specified user branch destination address is in the area other than the storage area of the programming/erasing program which has been downloaded. 0: User branch address setting is normal 1: User branch address setting is abnormal
1	FQ	Undefined	R/W	Frequency Error Detect Returns the check result whether the specified operating frequency of the CPU is in the range of the supported operating frequency. 0: Setting of operating frequency is normal 1: Setting of operating frequency is abnormal
0	SF	Undefined	R/W	Success/Fail Indicates whether initialization is completed normally. 0: Initialization has ended normally (no error) 1: Initialization has ended abnormally (error occurs)

(3) Programming Execution

When flash memory is programmed, the programming destination address and programming data on the user MAT must be passed to the programming program in which the program data is downloaded.

1. The start address of the programming destination on the user MAT is set in general register R5 of the CPU. This parameter is called FMPAR (flash multipurpose address area parameter).
   Since the program data is always in 128-byte units, the lower eight bits (MOA7 to MOA0) must be H'00 or H'80 as the boundary of the programming start address on the user MAT.
2. The program data for the user MAT must be prepared in the consecutive area. The program data must be in the consecutive space which can be accessed by using the MOV.B instruction of the CPU and is not the flash memory space.
   When data to be programmed does not satisfy 128 bytes, the 128-byte program data must be prepared by embedding the dummy code (H'FF).
   The start address of the area in which the prepared program data is stored must be set in general register R4. This parameter is called FMPDR (flash multipurpose data destination area parameter).
   For details on the programming procedure, see section 23.5.2, User Program Mode.

(3.1) Flash multipurpose address area parameter (FMPAR: general register R5 of CPU)

This parameter indicates the start address of the programming destination on the user MAT.

When an address in an area other than the flash memory space is set, an error occurs.

The start address of the programming destination must be at the 128-byte boundary. If this boundary condition is not satisfied, an error occurs. The error occurrence is indicated by the WA bit (bit 1) in FPFR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MOA31	MOA30	MOA29	MOA28	MOA27	MOA26	MOA25	MOA24	MOA23	MOA22	MOA21	MOA20	MOA19	MOA18	MOA17	MOA16
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MOA15	MOA14	MOA13	MOA12	MOA11	MOA10	MOA9	MOA8	MOA7	MOA6	MOA5	MOA4	MOA3	MOA2	MOA1	MOA0
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MOA31 to MOA0	Undefined	R/W	MOA31 to MOA0 Store the start address of the programming destination on the user MAT. The consecutive 128-byte programming is executed starting from the specified start address of the user MAT. The MOA6 to MOA0 bits are always 0 because the start address of the programming destination is at the 128-byte boundary.

(3.2) Flash multipurpose data destination area parameter (FMPDR: general register R4 of CPU)

This parameter indicates the start address in the area which stores the data to be programmed in the user MAT. When the storage destination of the program data is in flash memory, an error occurs. The error occurrence is indicated by the WD bit (bit 2) in FPFR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MOD31	MOD30	MOD29	MOD28	MOD27	MOD26	MOD25	MOD24	MOD23	MOD22	MOD21	MOD20	MOD19	MOD18	MOD17	MOD16
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MOD15	MOD14	MOD13	MOD12	MOD11	MOD10	MOD9	MOD8	MOD7	MOD6	MOD5	MOD4	MOD3	MOD2	MOD1	MOD0
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MOD31 to MOD0	Undefined	R/W	MOD31 to MOD0 Store the start address of the area which stores the program data for the user MAT. The consecutive 128-byte data is programmed to the user MAT starting from the specified start address.

(3.3) Flash pass/fail result parameter (FPFR: general register R0 of CPU)

This parameter indicates the return value of the program processing result.

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	MD	EE	FK	-	WD	WA	SF
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	Undefined	R/W	Unused Return 0.
6	MD	Undefined	R/W	Programming Mode Related Setting Error Detect Returns the check result of whether the signal input to the FWE pin is high and whether the error protection state is not entered. When a low-level signal is input to the FWE pin or the error protection state is entered, 1 is written to this bit. The input level to the FWE pin and the error protection state can be confirmed with the FWE bit (bit 7) and the FLER bit (bit 4) in FCCS, respectively. For conditions to enter the error protection state, see section 23.6.3, Error Protection. 0: FWE and FLER settings are normal (FWE = 1, FLER = 0) 1: FWE = 0 or FLER = 1, and programming cannot be performed

Bit	Bit Name	Initial Value	R/W	Description
5	EE	Undefined	R/W	Programming Execution Error Detect 1 is returned to this bit when the specified data could not be written because the user MAT was not erased or when flash-memory related register settings are partially changed on returning from the user branch processing. If this bit is set to 1, there is a high possibility that the user MAT is partially rewritten. In this case, after removing the error factor, erase the user MAT. If FMATS is set to H'AA and the user boot MAT is selected, an error occurs when programming is performed. In this case, both the user MAT and user boot MAT are not rewritten. Programming of the user boot MAT must be executed in boot mode or programmer mode. 0: Programming has ended normally 1: Programming has ended abnormally (programming result is not guaranteed)
4	FK	Undefined	R/W	Flash Key Register Error Detect Returns the check result of the value of FKEY before the start of the programming processing. 0: FKEY setting is normal (FKEY = H'5A) 1: FKEY setting is error (FKEY = value other than H'5A)
3	—	Undefined	R/W	Unused Return 0.
2	WD	Undefined	R/W	Write Data Address Error Detect When an address in the flash memory area is specified as the start address of the storage destination of the program data, an error occurs. 0: Setting of write data address is normal 1: Setting of write data address is abnormal

Bit	Bit Name	Initial Value	R/W	Description
1	WA	Undefined	R/W	Write Address Error Detect When the following items are specified as the start address of the programming destination, an error occurs. • The programming destination address is an area other than flash memory • The specified address is not at the 128-byte boundary (A6 to A0 are not 0) 0: Setting of programming destination address is normal 1: Setting of programming destination address is abnormal
0	SF	Undefined	R/W	Success/Fail Indicates whether the program processing has ended normally or not. 0: Programming has ended normally (no error) 1: Programming has ended abnormally (error occurs)

(4) Erasure Execution

When flash memory is erased, the erase-block number on the user MAT must be passed to the erasing program which is downloaded. This is set to the FEBS parameter (general register R4).

One block is specified from the block number 0 to 15.

For details on the erasing procedure, see section 23.5.2, User Program Mode.

(4.1) Flash erase block select parameter (FEBS: general register R4 of CPU)

This parameter specifies the erase-block number. Several block numbers cannot be specified.

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	EBS[7:0]							
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	Undefined	R/W	Unused Return 0.
7 to 0	EBS[7:0]	Undefined	R/W	• 512-kbyte flash memory Set the erase-block number in the range from 0 to 15. 0 corresponds to the EB0 block and 15 corresponds to the EB15 block. An error occurs when a number other than 0 to 15 (H'00 to H'0F) is set. • 256-kbyte flash memory Set the erase-block number in the range from 0 to 11. 0 corresponds to the EB0 block and 11 corresponds to the EB11 block. An error occurs when a number other than 0 to 11 (H'00 to H'0B) is set.

(4.2) Flash pass/fail result parameter (FPFR: general register R0 of CPU)

This parameter returns the value of the erasing processing result.

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

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	MD	EE	FK	EB	-	-	SF
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	Undefined	R/W	Unused Return 0.
6	MD	Undefined	R/W	Erasure Mode Related Setting Error Detect Returns the check result of whether the signal input to the FWE pin is high and whether the error protection state is not entered. When a low-level signal is input to the FWE pin or the error protection state is entered, 1 is written to this bit. The input level to the FWE pin and the error protection state can be confirmed with the FWE bit (bit 7) and the FLER bit (bit 4) in FCCS, respectively. For conditions to enter the error protection state, see section 23.6.3, Error Protection. 0: FWE and FLER settings are normal (FWE = 1, FLER = 0) 1: FWE = 0 or FLER = 1, and erasure cannot be performed

Bit	Bit Name	Initial Value	R/W	Description
5	EE	Undefined	R/W	Erasure Execution Error Detect 1 is returned to this bit when the user MAT could not be erased or when flash-memory related register settings are partially changed on returning from the user branch processing. If this bit is set to 1, there is a high possibility that the user MAT is partially erased. In this case, after removing the error factor, erase the user MAT. If FMATS is set to H'AA and the user boot MAT is selected, an error occurs when erasure is performed. In this case, both the user MAT and user boot MAT are not erased. Erasure of the user boot MAT must be executed in boot mode or programmer mode. 0: Erasure has ended normally 1: Erasure has ended abnormally (erasure result is not guaranteed)
4	FK	Undefined	R/W	Flash Key Register Error Detect Returns the check result of FKEY value before start of the erasing processing. 0: FKEY setting is normal (FKEY = H'5A) 1: FKEY setting is error (FKEY = value other than H'5A)
3	EB	Undefined	R/W	Erase Block Select Error Detect Returns the check result whether the specified erase-block number is in the block range of the user MAT. 0: Setting of erase-block number is normal 1: Setting of erase-block number is abnormal
2, 1	—	Undefined	R/W	Unused Return 0.
0	SF	Undefined	R/W	Success/Fail Indicates whether the erasing processing has ended normally or not. 0: Erasure has ended normally (no error) 1: Erasure has ended abnormally (error occurs)

23.4.4 RAM Emulation Register (RAMER)

When the realtime programming of the user MAT is emulated, RAMER sets the area of the user MAT which is overlapped with a part of the on-chip RAM. The RAM emulation must be executed in user mode or in user program mode.

For the division method of the user-MAT area, see table 23.7. In order to operate the emulation function certainly, the target MAT of the RAM emulation must not be accessed immediately after RAMER is programmed. If it is accessed, the normal access is not guaranteed.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	RAMS	RAM[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	RAMS	0	R/W	RAM Select Sets whether the user MAT is emulated or not. When RAMS = 1, all blocks of the user MAT are in the programming/erasing protection state. 0: Emulation is not selected Programming/erasing protection of all user-MAT blocks is invalid 1: Emulation is selected Programming/erasing protection of all user-MAT blocks is valid
2 to 0	RAM[2:0]	000	R/W	User MAT Area Select These bits are used with bit 3 to select the user-MAT area to be overlapped with the on-chip RAM. (See table 23.7.)

Table 23.7 Overlapping of RAM Area and User MAT Area

RAM Area	Block Name	RAMS	RAM2	RAM1	RAM0
H'FFFFA000 to H'FFFFAFFF	RAM area (4 kbytes)	0	x	x	x
H'00000000 to H'00000FFF	EB0 (4 kbytes)	1	0	0	0
H'00001000 to H'00001FFF	EB1 (4 kbytes)	1	0	0	1
H'00002000 to H'00002FFF	EB2 (4 kbytes)	1	0	1	0
H'00003000 to H'00003FFF	EB3 (4 kbytes)	1	0	1	1
H'00004000 to H'00004FFF	EB4 (4 kbytes)	1	1	0	0
H'00005000 to H'00005FFF	EB5 (4 kbytes)	1	1	0	1
H'00006000 to H'00006FFF	EB6 (4 kbytes)	1	1	1	0
H'00007000 to H'00007FFF	EB7 (4 kbytes)	1	1	1	1

Note: x: Don't care.

23.5 On-Board Programming Mode

When the pin is set in on-board programming mode and the reset start is executed, the on-board programming state that can program/erase the on-chip flash memory is entered. On-board programming mode has three operating modes: user program mode, user boot mode, and boot mode.

For details on the pin setting for entering each mode, see table 23.1. For details on the state transition of each mode for flash memory, see figure 23.2.

23.5.1 Boot Mode

Boot mode executes programming/erasing user MAT and user boot MAT by means of the control command and program data transmitted from the host using the on-chip SCI. The tool for transmitting the control command and program data must be prepared in the host. The SCI communication mode is set to asynchronous mode. When reset start is executed after this LSI's pin is set in boot mode, the boot program in the microcomputer is initiated. After the SCI bit rate is automatically adjusted, the communication with the host is executed by means of the control command method.

The system configuration diagram in boot mode is shown in figure 23.6. For details on the pin setting in boot mode, see table 23.1. Interrupts are ignored in boot mode, so do not generate them. Note that the AUD cannot be used during boot mode operation.

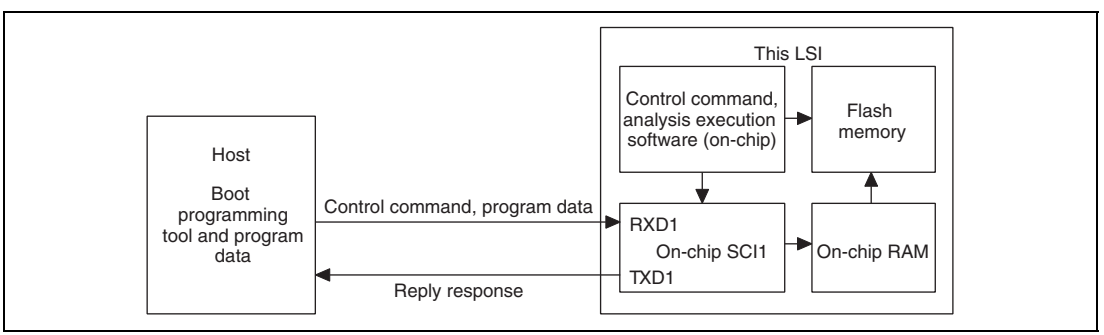


Figure 23.6 System Configuration in Boot Mode

(1) SCI Interface Setting by Host

When boot mode is initiated, this LSI measures the low period of asynchronous SCI-communication data (H'00), which is transmitted consecutively by the host. The SCI transmit/receive format is set to 8-bit data, 1 stop bit, and no parity. This LSI calculates the bit rate of transmission by the host by means of the measured low period and transmits the bit adjustment end sign (1 byte of H'00) to the host. The host must confirm that this bit adjustment end sign (H'00) has been received normally and transmits 1 byte of H'55 to this LSI. When reception is not executed normally, boot mode is initiated again (reset) and the operation described above must be executed. The bit rate between the host and this LSI is not matched because of the bit rate of transmission by the host and system clock frequency of this LSI. To operate the SCI normally, the transfer bit rate of the host must be set to 9,600 bps or 19,200 bps.

The system clock frequency which can automatically adjust the transfer bit rate of the host and the bit rate of this LSI is shown in table 23.8. Boot mode must be initiated in the range of this system clock. Note that the internal clock division ratio of ×1/3 is not supported in boot mode.

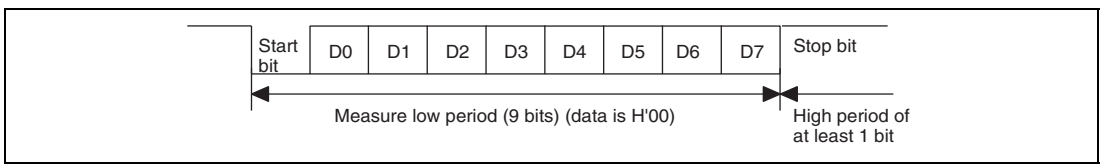


Figure 23.7 Automatic Adjustment Operation of SCI Bit Rate

Table 23.8 Peripheral Clock (Pϕ) Frequency that Can Automatically Adjust Bit Rate of This LSI

Host Bit Rate	Peripheral Clock (Pϕ) Frequency Which Can Automatically Adjust LSI's Bit Rate
9,600 bps	10 to 40 MHz
19,200 bps	10 to 40 MHz

Note: The internal clock division ratio of ×1/3 is not supported in boot mode.

(2) State Transition Diagram

Figure 23.8 gives an overview of the state transitions after the chip has been started up in boot mode. For details on boot mode, see section 23.9.1, Specifications of the Standard Serial Communications Interface in Boot Mode.

1. Bit-rate matching

   After the chip has been started up in boot mode, bit-rate matching between the SCI and the host proceeds.

2. Waiting for inquiry and selection commands

   The chip sends the requested information to the host in response to inquiries regarding the size and configuration of the user MAT, start addresses of the MATs, information on supported devices, etc.

3. Automatic erasure of the entire user MAT and user boot MAT

   After all necessary inquiries and selections have been made and the command for transition to the programming/erasure state is sent by the host, the entire user MAT and user boot MAT are automatically erased.

4. Waiting for programming/erasure command

— On receiving the programming selection command, the chip waits for data to be programmed. To program data, the host transmits the programming command code followed by the address where programming should start and the data to be programmed. This is repeated as required while the chip is in the programming-selected state. To terminate programming, H'FFFFFFFF should be transmitted as the first address of the area for programming. This makes the chip return to the programming/erasure command waiting state from the programming data waiting state.

— On receiving the erasure select command, the chip waits for the block number of a block to be erased. To erase a block, the host transmits the erasure command code followed by the number of the block to be erased. This is repeated as required while the chip is in the erasure-selected state. To terminate erasure, H'FF should be transmitted as the block number. This makes the chip return to the programming/erasure command waiting state from the erasure block number waiting state. Erasure should only be executed when a specific block is to be reprogrammed without executing a reset-start of the chip after the flash memory has been programmed in boot mode. If all desired programming is done in a single operation, such erasure processing is not necessary because all blocks are erased before the chip enters the programming/erasure/other command waiting state.

— In addition to the programming and erasure commands, commands for sum checking and blank checking (checking for erasure) of the user MAT and user boot MAT, reading data from the user MAT/user boot MAT, and acquiring current state information are provided.

Note that the command for reading from the user MAT/user boot MAT can only read data that has been programmed after automatic erasure of the entire user MAT and user boot MAT.

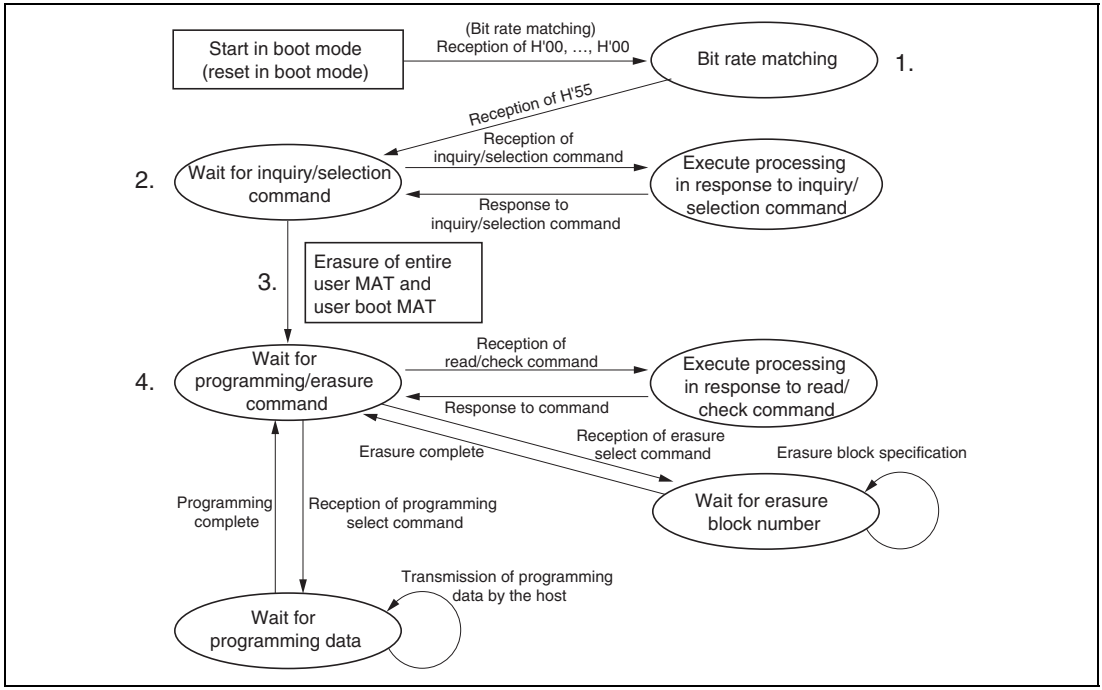


Figure 23.8 State Transitions in Boot Mode

23.5.2 User Program Mode

The user MAT can be programmed/erased in user program mode. (The user boot MAT cannot be programmed/erased.)

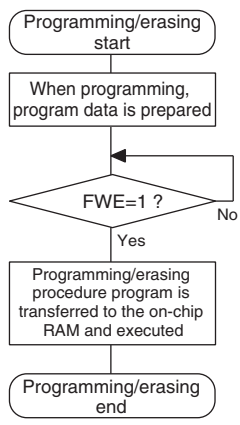
Programming/erasing is executed by downloading the program in the microcomputer.

The overview flow is shown in figure 23.9.

High voltage is applied to internal flash memory during the programming/erasing processing. Therefore, transition to reset must not be executed. Doing so may cause damage or destroy flash memory. If reset is executed accidentally, the reset signal must be released after the reset input period, which is longer than the normal 100 μs.

For details on the programming procedure, see the description in section 23.5.2 (2), Programming Procedure in User Program Mode. For details on the erasing procedure, see the description in section 23.5.2 (3), Erasing Procedure in User Program Mode.

For the overview of a processing that repeats erasing and programming by downloading the programming program and the erasing program in separate on-chip ROM areas using FTDAR, see the description in section 23.5.2 (4), Erasing and Programming Procedure in User Program Mode.



1. RAM emulation mode must be canceled in advance. Download cannot be executed in emulation mode.
2. When the program data is made by means of emulation, the download destination must be changed by FTDAR.
3. Inputting high level to the FWE pin sets the FWE bit to 1.
4. Programming/erasing is executed only in the on-chip RAM. However, if the program data is in a consecutive area and can be accessed by the MOV.B instruction of the CPU like SRAM/ROM, the program data can be in an external space.
5. After programming/erasing is finished, low level must be input to the FWE pin for protection.

Figure 23.9 Programming/Erasing Overview Flow

(1) On-Chip RAM Address Map when Programming/Erasing is Executed

Parts of the procedure program that are made by the user, like download request, programming/erasing procedure, and decision of the result, must be executed in the on-chip RAM. All of the on-chip program that is to be downloaded is in on-chip RAM. Note that on-chip RAM must be controlled so that these parts do not overlap.

Figure 23.10 shows the program area to be downloaded.

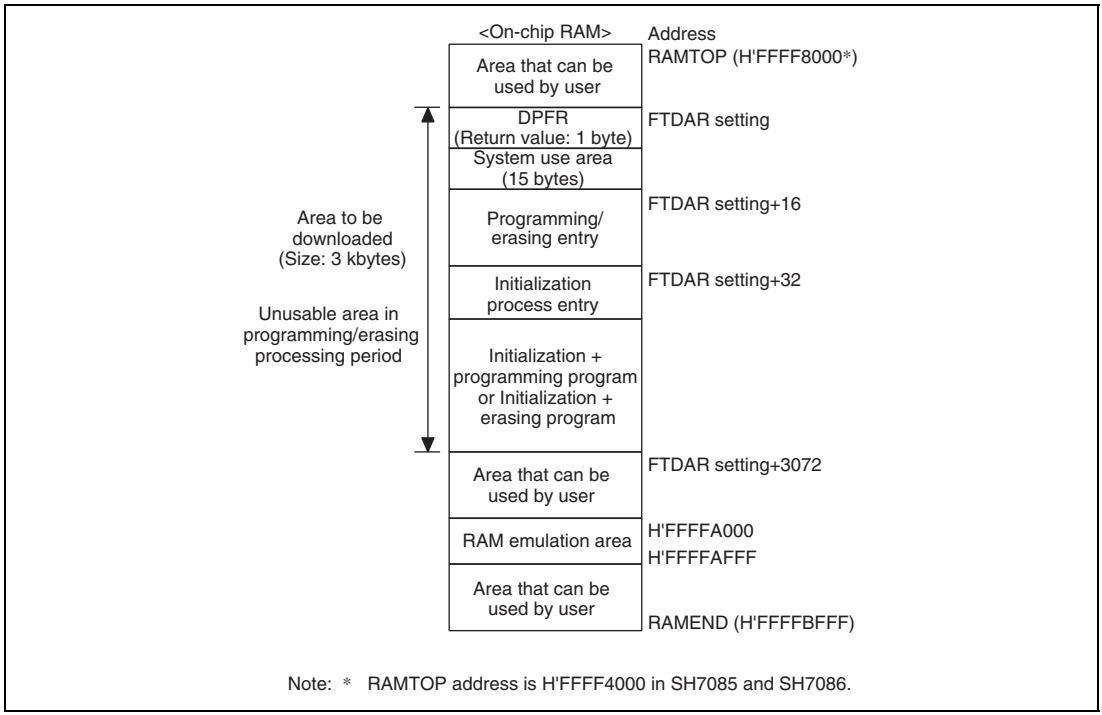


Figure 23.10 RAM Map after Download

(2) Programming Procedure in User Program Mode

The procedures for download, initialization, and programming are shown in figure 23.11.

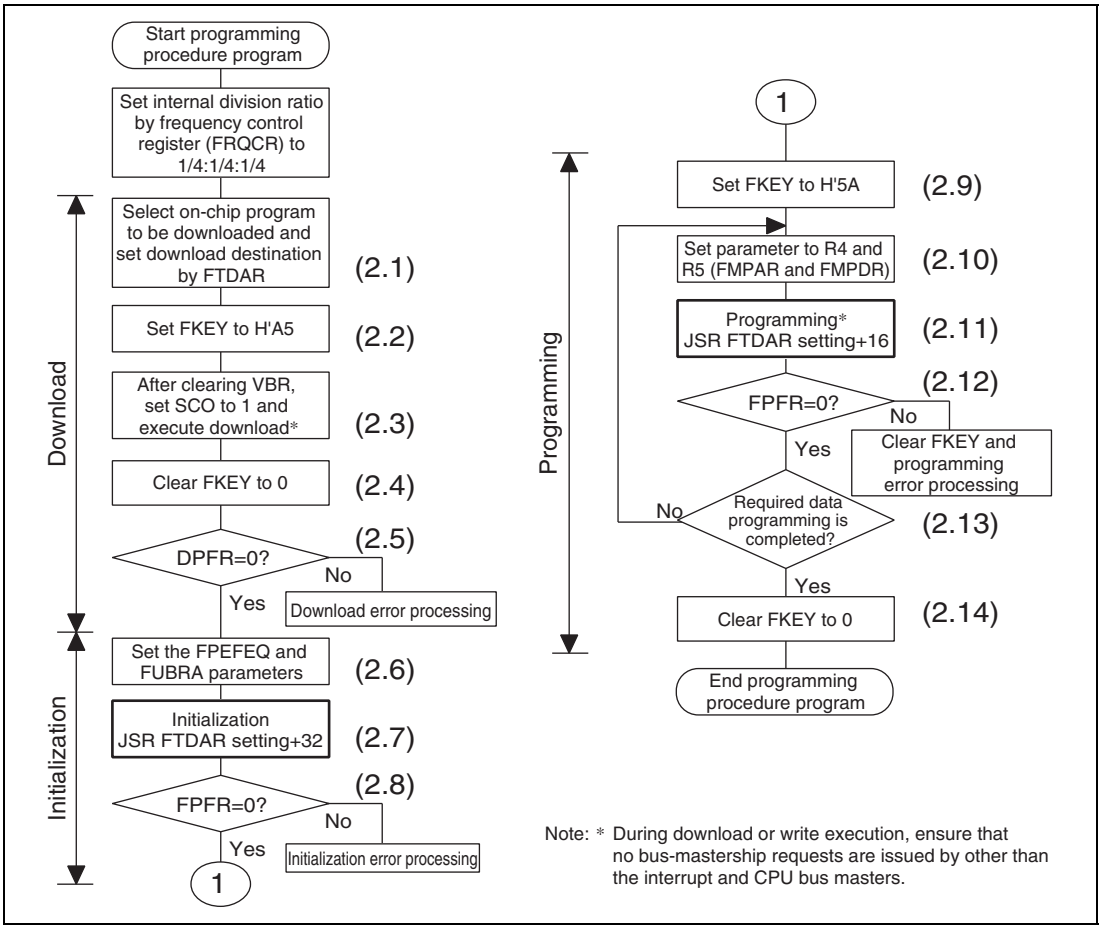


Figure 23.11 Programming Procedure

The details of the programming procedure are described below. The procedure program must be executed in an area other than the flash memory to be programmed. Especially the part where the SCO bit in FCCS is set to 1 for downloading must be executed in the on-chip RAM. Specify 1/4 (initial value) as the frequency division ratios of an internal clock (Iϕ), a bus clock (Bϕ), and a peripheral clock (Pϕ) through the frequency control register (FRQCR).

After the programming/erasing program has been downloaded and the SCO bit is cleared to 0, the setting of the frequency control register (FRQCR) can be changed to the desired value.

The area that can be executed in the steps of the user procedure program (on-chip RAM, user MAT, and external space) is shown in section 23.9.2, Areas for Storage of the Procedural Program and Data for Programming.

The following description assumes the area to be programmed on the user MAT is erased and program data is prepared in the consecutive area. When erasing has not been executed, carry out erasing before writing.

128-byte programming is performed in one program processing. When more than 128-byte programming is performed, programming destination address/program data parameter is updated in 128-byte units and programming is repeated.

When less than 128-byte programming is performed, data must total 128 bytes by adding the invalid data. If the invalid data to be added is H'FF, the program processing period can be shortened.

(2.1) Select the on-chip program to be downloaded

When the PPVS bit of FPCS is set to 1, the programming program is selected.
Several programming/erasing programs cannot be selected at one time. If several programs are set, download is not performed and a download error is returned to the source select error detect (SS) bit in the DPFR parameter.
Specify the start address of the download destination by FTDAR.

(2.2) Write H'A5 in FKEY

If H'A5 is not written to FKEY for protection, 1 cannot be written to the SCO bit for a download request.

(2.3) VBR is set to 0 and 1 is written to the SCO bit of FCCS, and then download is executed.

VBR must always be set to H'84000000 before setting the SCO bit to 1.
To write 1 to the SCO bit, the following conditions must be satisfied.

- RAM emulation mode is canceled.
- H'A5 is written to FKEY.
- The SCO bit writing is executed in the on-chip RAM.

When the SCO bit is set to 1, download is started automatically. When execution returns to the user procedure program, the SCO bit is cleared to 0. Therefore, the SCO bit cannot be confirmed to be 1 in the user procedure program.
The download result can be confirmed only by the return value of the DPFR parameter. Before the SCO bit is set to 1, incorrect decision must be prevented by setting the DPFR parameter, that is one byte of the start address of the on-chip RAM area specified by FTDAR, to a value other than the return value (H'FF).
When download is executed, particular interrupt processing, which is accompanied by the bank switch as described below, is performed as an internal microcomputer processing, so VBR need to be set to H'84000000. Four NOP instructions are executed immediately after the instructions that set the SCO bit to 1.

- The user MAT space is switched to the on-chip program storage area.
- After the selection condition of the download program and the address set in FTDAR are checked, the transfer processing is executed starting to the on-chip RAM address specified by FTDAR.
- The SCO bits in FCCS, FPCS, and FECS are cleared to 0.
- The return value is set to the DPFR parameter.
- After the on-chip program storage area is returned to the user MAT space, execution returns to the user procedure program.

After download is completed and the user procedure program is running, the VBR setting can be changed.

The notes on download are as follows.

In the download processing, the values of the general registers of the CPU are retained.

During download processing, ensure that no bus-mastership requests are issued by other than the interrupt and CPU bus masters. For details, see section 23.8.2, Interrupts during Programming/Erasing.

Since a stack area of maximum 128 bytes is used, an area of at least 128 bytes must be saved before setting the SCO bit to 1.

(2.4) FKEY is cleared to H'00 for protection.

(2.5) The value of the DPFR parameter must be checked to confirm the download result.

A recommended procedure for confirming the download result is shown below.

- Check the value of the DPFR parameter (one byte of start address of the download destination specified by FTDAR). If the value is H'00, download has been performed normally. If the value is not H'00, the source that caused download to fail can be investigated by the description below.
- If the value of the DPFR parameter is the same as before downloading (e.g. H'FF), the address setting of the download destination in FTDAR may be abnormal. In this case, confirm the setting of the TDER bit (bit 7) in FTDAR.
- If the value of the DPFR parameter is different from before downloading, check the SS bit (bit 2) and the FK bit (bit 1) in the DPFR parameter to ensure that the download program selection and FKEY register setting were normal, respectively.

(2.6) The operating frequency is set to the FPEFEQ parameter and the user branch destination is set to the FUBRA parameter for initialization.

- The current frequency of the CPU clock is set to the FPEFEQ parameter (general register R4). For the settable range of the FPEFEQ parameter, see section 28.3.1, Clock Timing.
  When the frequency is set out of this range, an error is returned to the FPFR parameter of the initialization program and initialization is not performed. For details on the

frequency setting, see the description in section 23.4.3 (2.1), Flash programming/erasing frequency parameter (FPEFEQ: general register R4 of CPU).

- The start address in the user branch destination is set to the (FUBRA: CPU general register R5) parameter.
  When the user branch processing is not required, 0 must be set to FUBRA.
  When the user branch is executed, the branch destination is executed in flash memory other than the one that is to be programmed. The area of the on-chip program that is downloaded cannot be set.
  The program processing must be returned from the user branch processing by the RTS instruction.
  See the description in section 23.4.3 (2.2), Flash user branch address setting parameter (FUBRA: general register R5 of CPU).

(2.7) Initialization

When a programming program is downloaded, the initialization program is also downloaded to on-chip RAM. There is an entry point of the initialization program in the area from (download start address set by FTDAR) + 32 bytes. The subroutine is called and initialization is executed by using the following steps.

```
MOV.L  #DLTOP+32,R1         ; Set entry address to R1
JSR    @R1                  ; Call initialization routine
NOP
```

- The general registers other than R0 are saved in the initialization program.
- R0 is a return value of the FPFR parameter.
- Since the stack area is used in the initialization program, a stack area of maximum 128 bytes must be reserved in RAM.
- Interrupts can be accepted during the execution of the initialization program. However, the program storage area and stack area in on-chip RAM and register values must not be destroyed.

(2.8) The return value of the initialization program, FPFR (general register R0) is checked.

(2.9) FKEY must be set to H'5A and the user MAT must be prepared for programming.

(2.10) The parameter which is required for programming is set.

The start address of the programming destination of the user MAT (FMPAR) is set to general register R5. The start address of the program data storage area (FMPDR) is set to general register R4.

- FMPAR setting

  FMPAR specifies the programming destination start address. When an address other than one in the user MAT area is specified, even if the programming program is

executed, programming is not executed and an error is returned to the return value parameter FPFR. Since the unit is 128 bytes, the lower eight bits (MOA7 to MOA0) must be in the 128-byte boundary of H'00 or H'80.

- FMPDR setting

  If the storage destination of the program data is flash memory, even when the program execution routine is executed, programming is not executed and an error is returned to the FPFR parameter. In this case, the program data must be transferred to on-chip RAM and then programming must be executed.

(2.11) Programming

There is an entry point of the programming program in the area from (download start address set by FTDAR) + 16 bytes of on-chip RAM. The subroutine is called and programming is executed by using the following steps.

```
MOV.L  #DLTOP+16,R1          ; Set entry address to R1
JSR    @R1                   ; Call programming routine
NOP
```

— The general registers other than R0 are saved in the programming program.

— R0 is a return value of the FPFR parameter.

— Since the stack area is used in the programming program, a stack area of maximum 128 bytes must be reserved in RAM.

— During write processing, ensure that no bus-mastership requests are issued by other than the interrupt and CPU bus masters. For details, see section 23.8.2, Interrupts during Programming/Erasing.

(2.12) The return value in the programming program, FPFR (general register R0) is checked.

(2.13) Determine whether programming of the necessary data has finished.

If more than 128 bytes of data are to be programmed, specify FMPAR and FMPDR in 128-byte units, and repeat steps (2.10) to (2.13). Increment the programming destination address by 128 bytes and update the programming data pointer correctly. If an address which has already been programmed is written to again, not only will a programming error occur, but also flash memory will be damaged.

(2.14) After programming finishes, clear FKEY and specify software protection.

If this LSI is restarted by a power-on reset immediately after user MAT programming has finished, secure a reset period (period of $\overline{RES}$ = 0) that is at least as long as the normal 100 μs.

(3) Erasing Procedure in User Program Mode

The procedures for download, initialization, and erasing are shown in figure 23.12.

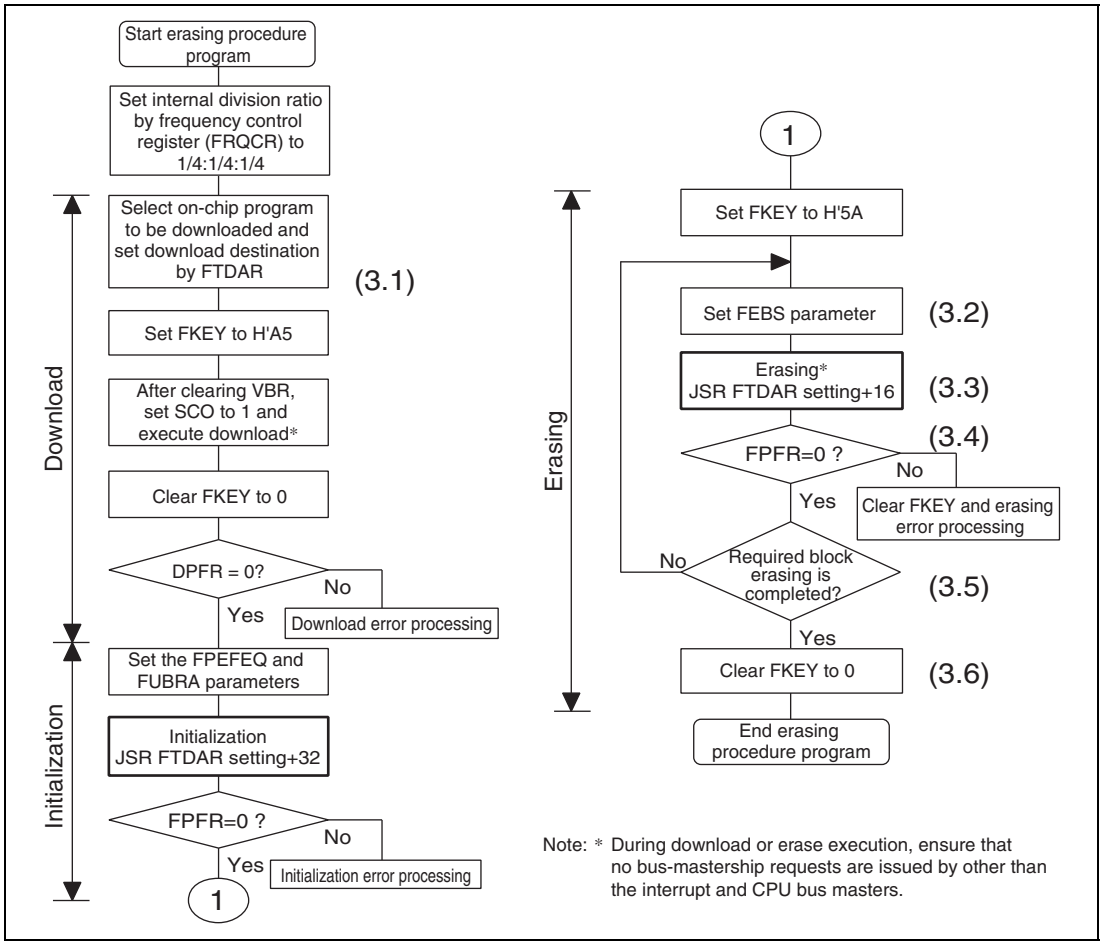


Figure 23.12 Erasing Procedure

The details of the erasing procedure are described below. The procedure program must be executed in an area other than the user MAT to be erased.

Especially the part where the SCO bit in FCCS is set to 1 for downloading must be executed in on-chip RAM.

The area that can be executed in the steps of the user procedure program (on-chip RAM, user MAT, and external space) is shown in section 23.9.2, Areas for Storage of the Procedural Program and Data for Programming.

The frequency division ratio of an internal clock (Iϕ), a bus clock (Bϕ), and a peripheral clock (Pϕ) is specified as ×1/4 (initial value) by the frequency control register (FRQCR).

After the programming/erasing program has been downloaded and the SCO bit is cleared to 0, the setting of the frequency control register (FRQCR) can be changed to the desired value.

For the downloaded on-chip program area, see the RAM map for programming/erasing in figure 23.10.

A single divided block is erased by one erasing processing. For block divisions, see figure 23.4. To erase two or more blocks, update the erase block number and perform the erasing processing for each block.

(3.1) Select the on-chip program to be downloaded

Set the EPVB bit in FECS to 1.

Several programming/erasing programs cannot be selected at one time. If several programs are set, download is not performed and a download error is returned to the source select error detect (SS) bit in the DPFR parameter.

Specify the start address of the download destination by FTDAR.

The procedures to be carried out after setting FKEY, e.g. download and initialization, are the same as those in the programming procedure. For details, see the description in section 23.5.2 (2), Programming Procedure in User Program Mode.

(3.2) Set the FEBS parameter necessary for erasure

Set the erase block number of the user MAT in the flash erase block select parameter (FEBS: general register R4). If a value other than an erase block number of the user MAT is set, no block is erased even though the erasing program is executed, and an error is returned to the return value parameter FPFR.

(3.3) Erasure

Similar to as in programming, there is an entry point of the erasing program in the area from (download start address set by FTDAR) + 16 bytes of on-chip RAM. The subroutine is called and erasing is executed by using the following steps.

```
MOV.L  #DLTOP+16,R1          ; Set entry address to R1
JSR    @R1                   ; Call erasing routine
NOP
```

— The general registers other than R0 are saved in the erasing program.

— R0 is a return value of the FPFR parameter.

— Since the stack area is used in the erasing program, a stack area of maximum 128 bytes must be reserved in RAM.

— During erase processing, ensure that no bus-mastership requests are issued by other than the interrupt and CPU bus masters. For details, see section 23.8.2, Interrupts during Programming/Erasing.

(3.4) The return value in the erasing program, FPFR (general register R0) is checked.

(3.5) Determine whether erasure of the necessary blocks has finished.

If more than one block is to be erased, update the FEBS parameter and repeat steps (3.2) to (3.5). Blocks that have already been erased can be erased again.

(3.6) After erasure finishes, clear FKEY and specify software protection.

If this LSI is restarted by a power-on reset immediately after user MAT erasing has finished, secure a reset period (period of $\overline{RES} = 0$) that is at least as long as the normal 100 μs.

(4) Erasing and Programming Procedure in User Program Mode

By changing the on-chip RAM address of the download destination in FTDAR, the erasing program and programming program can be downloaded to separate on-chip RAM areas.

Figure 23.13 shows an example of repetitively executing RAM emulation, erasing, and programming.

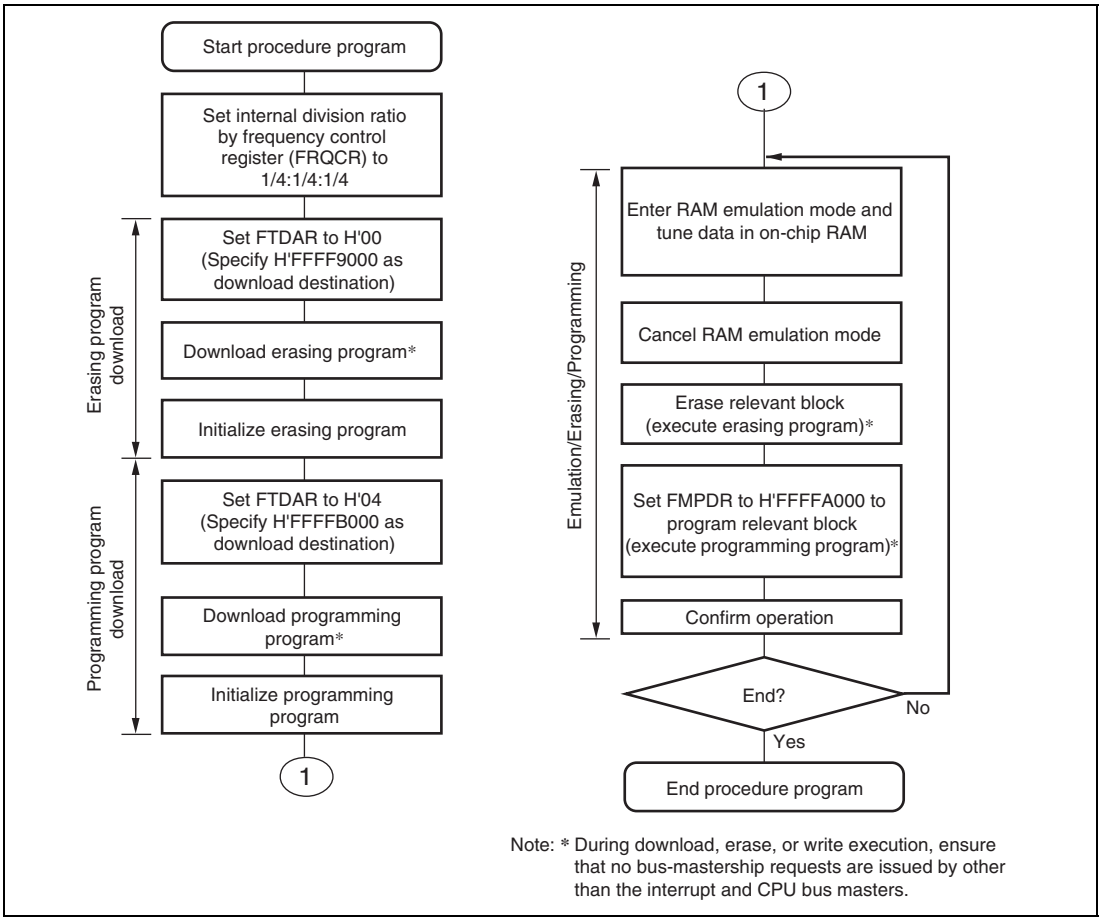


Figure 23.13 Sample Procedure of Repeating RAM Emulation, Erasing, and Programming (Overview)

In the above example, the erasing program and programming program are downloaded to areas excluding addresses (H'FFFFA000 to H'FFFFAFFF) to execute RAM emulation.

Download and initialization are performed only once at the beginning.

In this kind of operation, note the following:

- Be careful not to destroy on-chip RAM with overlapped settings.

  In addition to the RAM emulation area, erasing program area, and programming program area, areas for the user procedure programs, work area, and stack area are reserved in on-chip RAM. Do not make settings that will overwrite data in these areas.
- Be sure to initialize both the erasing program and programming program.

  Initialization by setting the FPEFEQ and FUBRA parameters must be performed for both the erasing program and the programming program. Initialization must be executed for both entry addresses: (download start address for erasing program) + 32 bytes (H'FFFF9020 in this example) and (download start address for programming program) + 32 bytes (H'FFFFB020 in this example).

23.5.3 User Boot Mode

This LSI has user boot mode which is initiated with different mode pin settings than those in user program mode or boot mode. User boot mode is a user-arbitrary boot mode, unlike boot mode that uses the on-chip SCI.

Only the user MAT can be programmed/erased in user boot mode. Programming/erasing of the user boot MAT is only enabled in boot mode or programmer mode.

(1) User Boot Mode Initiation

For the mode pin settings to start up user boot mode, see table 23.1.

When the reset start is executed in user boot mode, the check routine for flash-memory related registers runs. The RAM area about 1.2 kbytes from H'FFFF9800 and 4 bytes from H'FFFFAFFC (a stack area) is used by the routine. While the check routine is running, NMI and all other interrupts cannot be accepted. Neither can the AUD be used in this period. This period is 100 μs while operating at an internal frequency of 40 MHz.

Next, processing starts from the execution start address of the reset vector in the user boot MAT. At this point, H'AA is set to the flash MAT select register (FMATS) because the execution MAT is the user boot MAT.

(2) User MAT Programming in User Boot Mode

For programming the user MAT in user boot mode, additional processings made by setting FMATS are required: switching from user-boot-MAT selection state to user-MAT selection state, and switching back to user-boot-MAT selection state after programming completes.

Figure 23.14 shows the procedure for programming the user MAT in user boot mode.

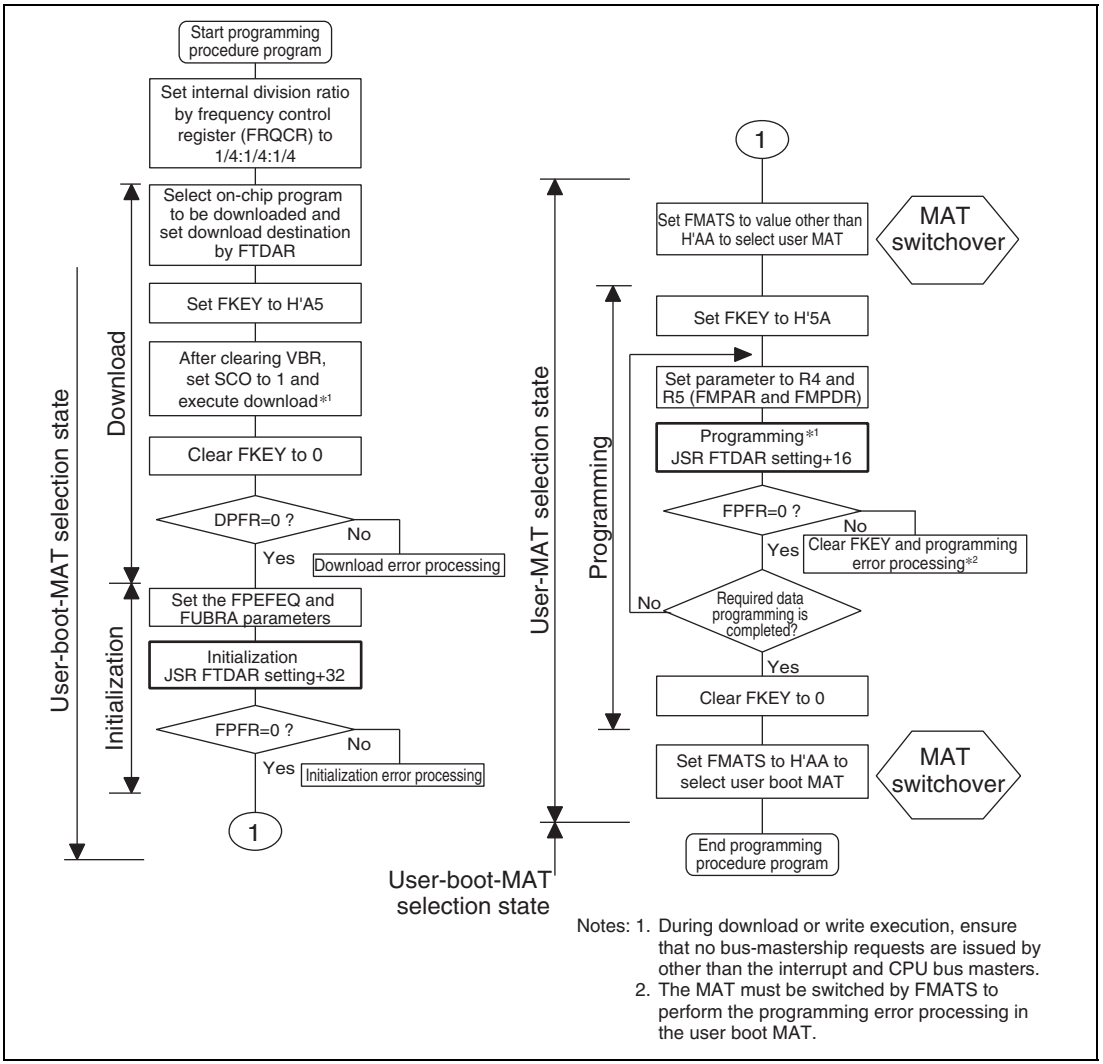


Figure 23.14 Procedure for Programming User MAT in User Boot Mode

The difference between the programming procedures in user program mode and user boot mode is whether the MAT is switched or not as shown in figure 23.14.

In user boot mode, the user boot MAT can be seen in the flash memory space with the user MAT hidden in the background. The user MAT and user boot MAT are switched only while the user MAT is being programmed. Because the user boot MAT is hidden while the user MAT is being programmed, the procedure program must be located in an area other than flash memory. After programming finishes, switch the MATs again to return to the first state.

MAT switchover is enabled by writing a specific value to FMATS. However note that while the MATs are being switched, the LSI is in an unstable state, e.g. access to a MAT is not allowed until MAT switching is completely finished, and if an interrupt occurs, from which MAT the interrupt vector is read from is undetermined. Perform MAT switching in accordance with the description in section 23.8.1, Switching between User MAT and User Boot MAT.

Except for MAT switching, the programming procedure is the same as that in user program mode.

The area that can be executed in the steps of the user procedure program (on-chip RAM, user MAT, and external space) is shown in section 23.9.2, Areas for Storage of the Procedural Program and Data for Programming.

(3) User MAT Erasing in User Boot Mode

For erasing the user MAT in user boot mode, additional processings made by setting FMATS are required: switching from user-boot-MAT selection state to user-MAT selection state, and switching back to user-boot-MAT selection state after erasing completes.

Figure 23.15 shows the procedure for erasing the user MAT in user boot mode.

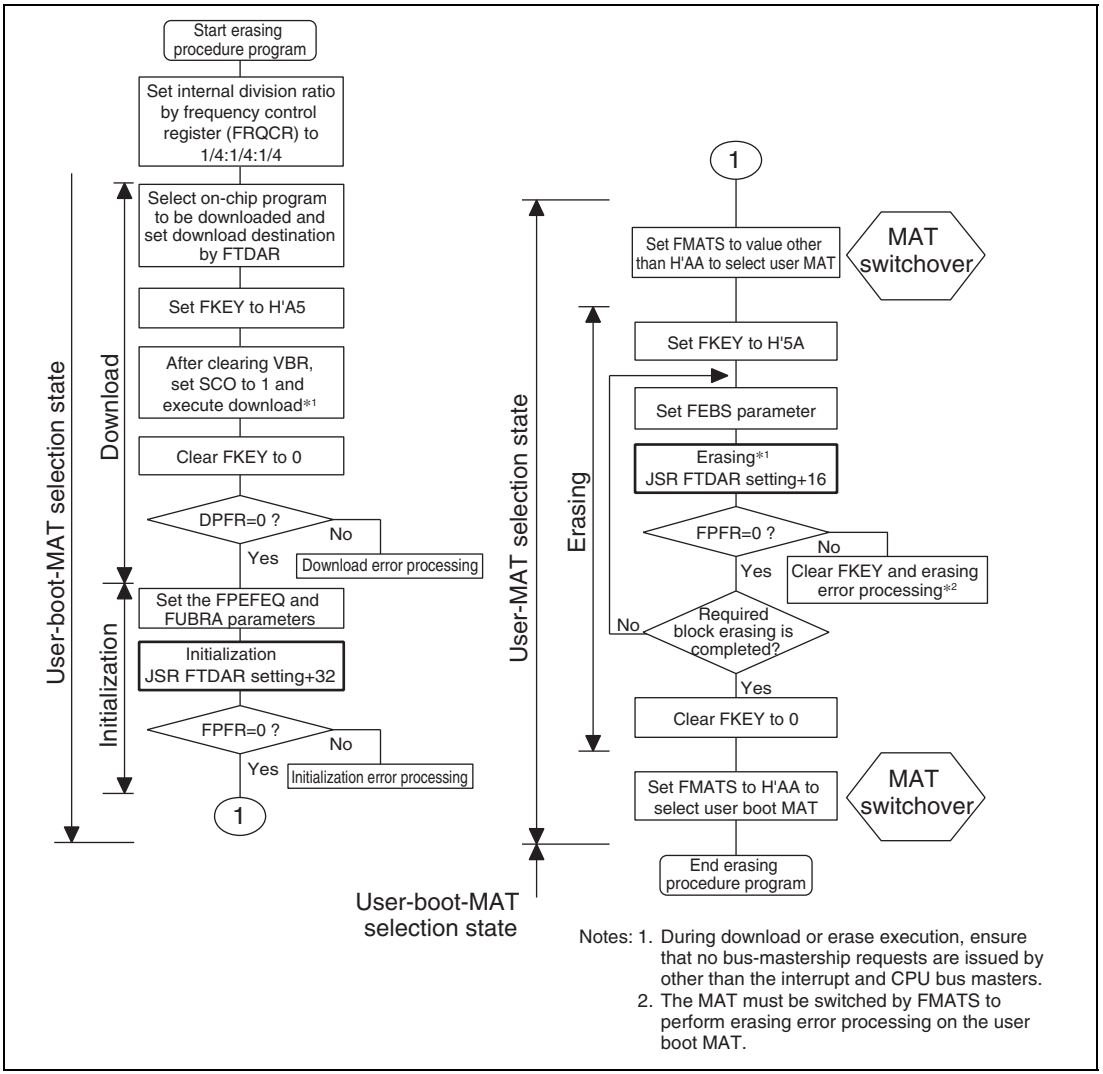


Figure 23.15 Procedure for Erasing User MAT in User Boot Mode

The difference between the erasing procedures in user program mode and user boot mode depends on whether the MAT is switched or not as shown in figure 23.15.

MAT switching is enabled by writing a specific value to FMATS. However note that while the MATs are being switched, the LSI is in an unstable state, e.g. access to a MAT is not allowed until MAT switching is completed finished, and if an interrupt occurs, from which MAT the interrupt vector is read from is undetermined. Perform MAT switching in accordance with the description in section 23.8.1, Switching between User MAT and User Boot MAT.

Except for MAT switching, the erasing procedure is the same as that in user program mode.

The area that can be executed in the steps of the user procedure program (on-chip RAM, user MAT, and external space) is shown in section 23.9.2, Areas for Storage of the Procedural Program and Data for Programming.

23.6 Protection

There are three kinds of flash memory program/erase protection: hardware, software, and error protection.

23.6.1 Hardware Protection

Programming and erasing of flash memory is forcibly disabled or suspended by hardware protection. In this state, the downloading of an on-chip program and initialization of the flash memory are possible. However, an activated program for programming or erasure cannot program or erase locations in a user MAT, and the error in programming/erasing is reported in the FPFR parameter.

Table 23.9 Hardware Protection

		Function to be Protected	
Item	**Description**	**Download**	**Programming/ Erasure**
FWE-pin protection	The input of a low-level signal on the FWE pin clears the FWE bit of FCCS and the LSI enters a programming/erasing-protected state.	—	√
Reset/standby protection	• A power-on reset (including a power-on reset by the WDT) and entry to standby mode initializes the programming/erasing interface registers and the LSI enters a programming/erasing-protected state. • Resetting by means of the $\overline{\text{RES}}$ pin after power is initially supplied will not make the LSI enter the reset state unless the $\overline{\text{RES}}$ pin is held low until oscillation has stabilized. In the case of a reset during operation, hold the $\overline{\text{RES}}$ pin low for the $\overline{\text{RES}}$ pulse width that is specified in the section on AC characteristics. If the LSI is reset during programming or erasure, data in the flash memory is not guaranteed. In this case, execute erasure and then execute programming again.	√	√

23.6.2 Software Protection

Software protection is set up in any of three ways: by disabling the downloading of on-chip programs for programming and erasing, by means of a key code, and by the RAM emulation register (RAMER).

Table 23.10 Software Protection

		Function to be Protected	
Item	**Description**	**Download**	**Programming/ Erasure**
Protection by the SCO bit	Clearing the SCO bit in FCCS disables downloading of the programming/erasing program, thus making the LSI enter a programming/erasing-protected state.	√	√
Protection by FKEY	Downloading and programming/erasing are disabled unless the required key code is written in FKEY. Different key codes are used for downloading and for programming/erasing.	√	√
Emulation protection	Setting the RAMS bit in RAMER to 1 makes the LSI enter a programming/erasing-protected state.	√	√

23.6.3 Error Protection

Error protection is a mechanism for aborting programming or erasure when an error occurs, in the form of the microcomputer getting out of control during programming/erasing of the flash memory or operations that are not in accordance with the established procedures for programming/erasing. Aborting programming or erasure in such cases prevents damage to the flash memory due to excessive programming or erasing.

If the microcomputer malfunctions during programming/erasing of the flash memory, the FLER bit in FCCS is set to 1 and the LSI enters the error protection state, thus aborting programming or erasure.

The FLER bit is set to 1 in the following conditions:

- When the relevant bank area of flash memory is read during programming/erasing (including a vector read or an instruction fetch)
- When a SLEEP instruction (including software standby mode) is executed during programming/erasing

Error protection is cancelled (FLER bit is cleared) only by a power-on reset.

Note that the reset signal should only be released after providing a reset input over a period longer than the normal 100 µs. Since high voltages are applied during programming/erasing of the flash memory, some voltage may still remain even after the error protection state has been entered. For this reason, it is necessary to reduce the risk of damage to the flash memory by extending the reset period so that the charge is released.

The state-transition diagram in figure 23.16 shows transitions to and from the error protection state.

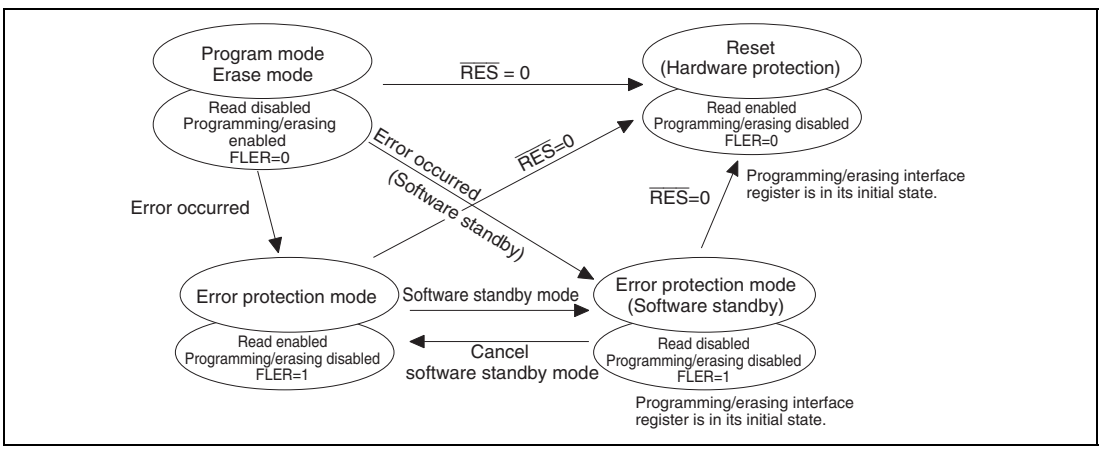


Figure 23.16 Transitions to and from Error Protection State

23.7 Flash Memory Emulation in RAM

To provide real-time emulation in RAM of data that is to be written to the flash memory, a part of the RAM can be overlaid on an area of flash memory (user MAT) that has been specified by the RAM emulation register (RAMER). After the RAMER setting is made, the RAM is accessible in both the user MAT area and as the RAM area that has been overlaid on the user MAT area. Such emulation is possible in user mode and user program mode.

Figure 23.17 shows an example of the emulation of realtime programming of the user MAT area.

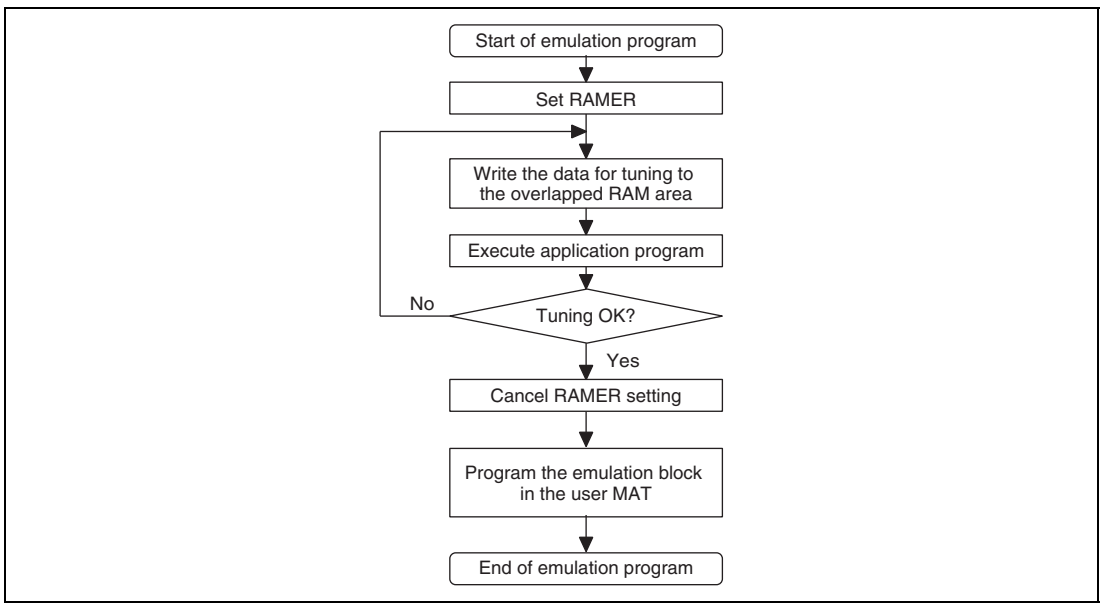


Figure 23.17 Emulation of Flash Memory in RAM

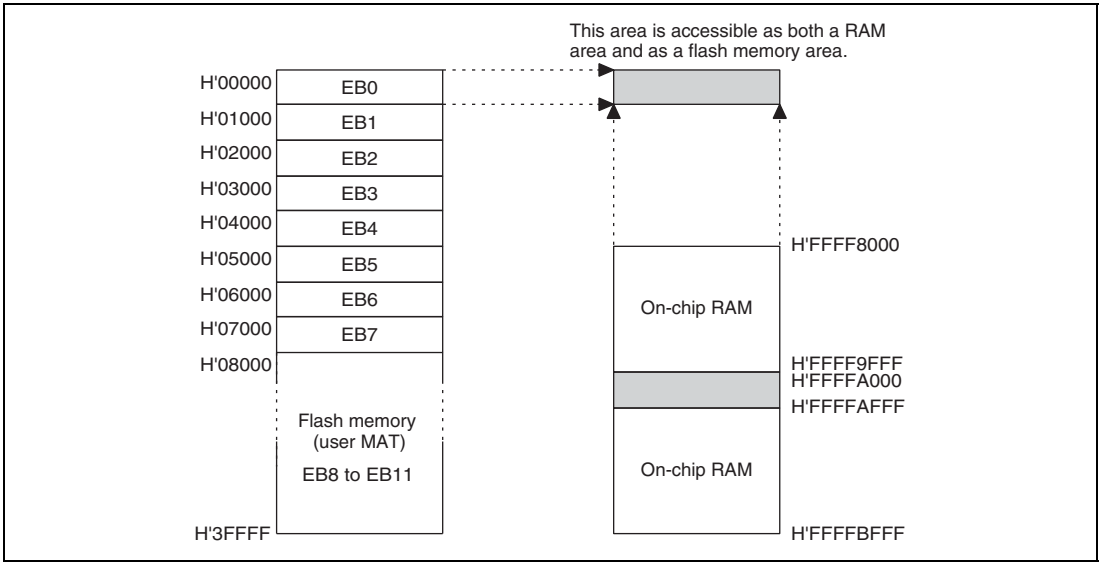


Figure 23.18 Example of Overlapped RAM Operation (SH7083: 256-kbyte Flash Memory Version)

Figure 23.18 shows an example of an overlap on block area EB0 of the flash memory.

Emulation is possible for a single area selected from among the eight areas, from EB0 to EB7, of the user MAT. The area is selected by the setting of the RAM2 to RAM0 bits in RAMER.

1. To overlap a part of the RAM on area EB0, to allow realtime programming of the data for this area, set the RAMS bit in RAMER to 1, and each of the RAM2 to RAM0 bits to 0.
2. Realtime programming is carried out using the overlaid area of RAM.

In programming or erasing the user MAT, it is necessary to run a program that implements a series of procedural steps, including the downloading of an on-chip program. In this process, set the download area with FTDAR so that the overlaid RAM area and the area where the on-chip program is to be downloaded do not overlap.

Figure 23.19 shows an example of programming data that has been emulated to the EB0 area in the user MAT.

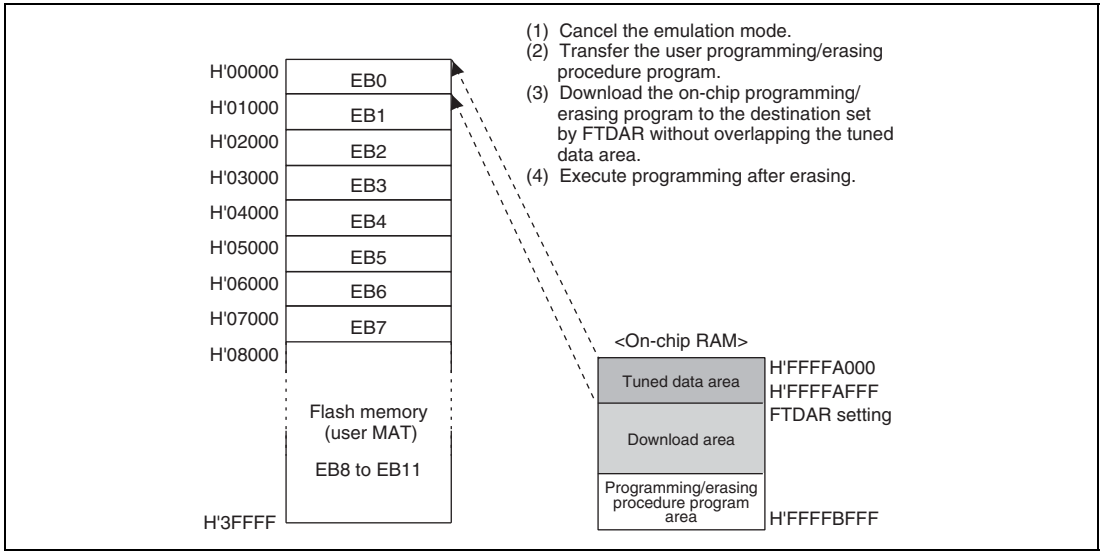


Figure 23.19 Programming of Tuned Data (SH7083: 256-kbyte Flash Memory Version)

1. After the data to be programmed has fixed values, clear the RAMS bit to 0 to cancel the overlap of RAM. Emulation mode is canceled and emulation protection is also cleared.
2. Transfer the user programming/erasing procedure program to RAM.
3. Run the programming/erasing procedure program in RAM and download the on-chip programming/erasing program.
   Specify the download start address with FTDAR so that the tuned data area does not overlap with the download area.
4. When the EB0 area of the user MAT has not been erased, erasing must be performed before programming. Set the parameters FMPAR and FMPDR so that the tuned data is designated, and execute programming.

Note: Setting the RAMS bit to 1 puts all the blocks in flash memory in the programming/erasing-protected state regardless of the values of the RAM2 to RAM0 bits (emulation protection). Clear the RAMS bit to 0 before actual programming or erasure. Though RAM emulation can also be carried out with the user boot MAT selected, the user boot MAT can be erased or programmed only in boot mode or programmer mode.

23.8 Usage Notes

23.8.1 Switching between User MAT and User Boot MAT

It is possible to switch between the user MAT and user boot MAT. However, the following procedure is required because these MATs are allocated to address 0.

(Switching to the user boot MAT disables programming and erasing. Programming of the user boot MAT must take place in boot mode or programmer mode.)

1. MAT switching by FMATS should always be executed from the on-chip RAM. The SH microcomputer prefetches execution instructions. Therefore, a switchover during program execution in the user MAT causes an instruction code in the user MAT to be prefetched or an instruction in the newly selected user boot MAT to be prefetched, thus resulting in unstable operation.
2. To ensure that the MAT that has been switched to is accessible, execute four NOP instructions in on-chip RAM immediately after writing to FMATS of on-chip RAM (this prevents access to the flash memory during MAT switching).
3. If an interrupt occurs during switching, there is no guarantee of which memory MAT is being accessed.
   Always mask the maskable interrupts before switching MATs. In addition, configuring the system so that NMI interrupts do not occur during MAT switching is recommended.
4. After the MATs have been switched, take care because the interrupt vector table will also have been switched.
   If the same interrupt processings are to be executed before and after MAT switching or interrupt requests cannot be disabled, transfer the interrupt processing routine to on-chip RAM, and use the VBR setting to place the interrupt vector table in on chip RAM. In this case, make sure the VBR setting change does not conflict with the interrupt occurrence.
5. Memory sizes of the user MAT and user boot MAT are different. When accessing the user boot MAT, do not access addresses exceeding the 12-kbyte memory space. If access goes beyond the 12-kbyte space, the values read are undefined.

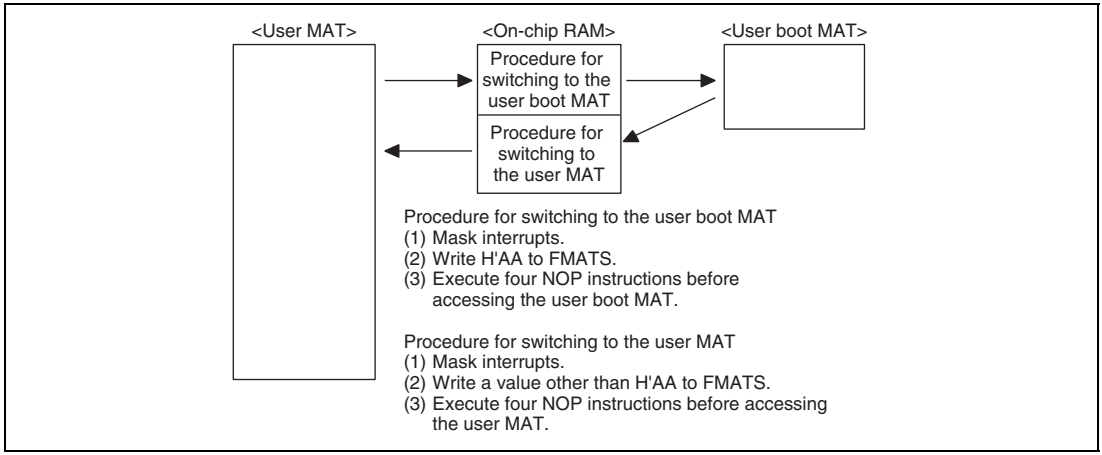


Figure 23.20 Switching between User MAT and User Boot MAT

23.8.2 Interrupts during Programming/Erasing

(1) Download of On-Chip Program

(1.1) VBR setting change

Before downloading the on-chip program, VBR must be set to H'84000000. If VBR is set to a value other than H'84000000, the interrupt vector table is placed in the user MAT (FMATS is not H'AA) or the user boot MAT (FMATS is H'AA) on setting H'84000000 to VBR.

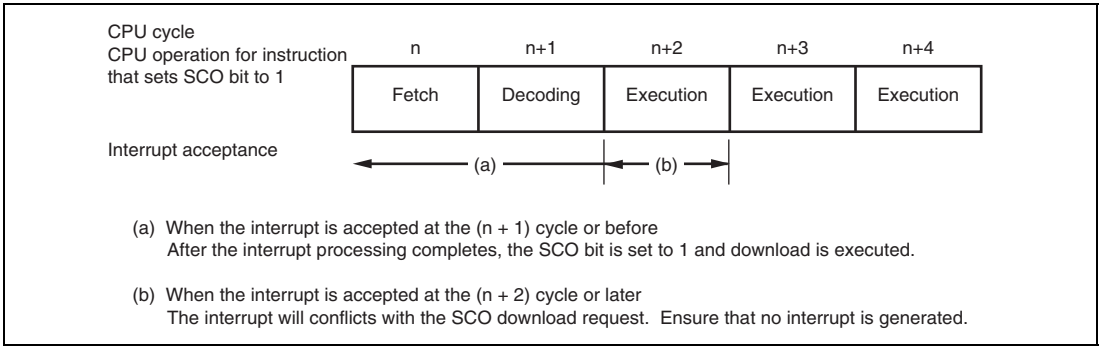
When VBR setting change conflicts with interrupt occurrence, whether the vector table before or after VBR is changed is referenced may cause an error.

Therefore, for cases where VBR setting change may conflict with interrupt occurrence, prepare a vector table to be referenced when VBR is H'00000000 (initial value) at the start of the user MAT or user boot MAT.

(1.2) SCO download request and interrupt request

Download of the on-chip programming/erasing program that is initiated by setting the SCO bit in FCCS to 1 generates a particular interrupt processing accompanied by MAT switchover. Operation when the SCO download request and interrupt request conflicts is described below.

1. Contention between SCO download request and interrupt request
   Figure 23.21 shows the timing of contention between execution of the instruction that sets the SCO bit in FCCS to 1 and interrupt acceptance.



(a) When the interrupt is accepted at the (n + 1) cycle or before
After the interrupt processing completes, the SCO bit is set to 1 and download is executed.

(b) When the interrupt is accepted at the (n + 2) cycle or later
The interrupt will conflicts with the SCO download request. Ensure that no interrupt is generated.

Figure 23.21 Timing of Contention between SCO Download Request and Interrupt Request

2. Generation of interrupt requests during downloading
Securing of bus-mastership by other than the interrupt and CPU bus masters (DMA transfer, DTC transfer, SDRAM refresh) is prohibited during SCO download execution.

(2) Interrupts during programming/erasing

Securing of bus-mastership by other than the interrupt and CPU bus masters (DMA transfer, DTC transfer, SDRAM refresh) is prohibited during programming or erase execution by a downloaded on-chip program.

23.8.3 Other Notes

1. Download time of on-chip program

   The programming program that includes the initialization routine and the erasing program that includes the initialization routine are each 3 kbytes or less. Accordingly, when the CPU clock frequency is 20 MHz, the download for each program takes approximately 10 ms at maximum.

2. User branch processing intervals

   The intervals for executing the user branch processing differs in programming and erasing. The processing phase also differs. Table 23.11 lists the maximum intervals for initiating the user branch processing when the CPU clock frequency is 80 MHz.

Table 23.11 Initiation Intervals of User Branch Processing

Processing Name	Maximum Interval
Programming	Approximately 2 ms
Erasing	Approximately 15 ms

However, when operation is done with CPU clock of 80 MHz, maximum values of the time until first user branch processing are as shown in table 23.12.

Table 23.12 Initial User Branch Processing Time

Processing Name	Max.
Programming	Approximately 2 ms
Erasing	Approximately 15 ms

3. Write to flash-memory related registers by DMAC or DTC

   While an instruction in on-chip RAM is being executed, the DMAC or DTC can write to the SCO bit in FCCS that is used for a download request or FMATS that is used for MAT switching. Make sure that these registers are not accidentally written to, otherwise an on-chip program may be downloaded and destroy RAM or a MAT switchover may occur and the CPU get out of control.

4. State in which interrupts are ignored

   In the following modes or period, interrupt requests are ignored; they are not executed and the interrupt sources are not retained.

   — Boot mode

   — Programmer mode

5. Note on programming the product having a 256-Kbyte user MAT

   If an attempt is made to program the product having a 256-Kbyte user MAT with more than 256 Kbytes, data programmed after the first 256 Kbytes are not guaranteed.

6. Compatibility with programming/erasing program of conventional F-ZTAT SH microcomputer

   A programming/erasing program for flash memory used in the conventional F-ZTAT SH microcomputer which does not support download of the on-chip program by a SCO transfer request cannot run in this LSI.
   Be sure to download the on-chip program to execute programming/erasing of flash memory in this LSI.

7. Monitoring runaway by WDT

   Unlike the conventional F-ZTAT SH microcomputer, no countermeasures are available for a runaway by WDT during programming/erasing by the downloaded on-chip program.
   Prepare countermeasures (e.g. use of the user branch routine and periodic timer interrupts) for WDT while taking the programming/erasing time into consideration as required.

23.9 Supplementary Information

23.9.1 Specifications of the Standard Serial Communications Interface in Boot Mode

The boot program activated in boot mode communicates with the host via the on-chip SCI of the LSI. The specifications of the serial communications interface between the host and the boot program are described below.

- States of the boot program

The boot program has three states.

1. Bit-rate matching state

   In this state, the boot program adjusts the bit rate to match that of the host. When the chip starts up in boot mode, the boot program is activated and enters the bit-rate matching state, in which it receives commands from the host and adjusts the bit rate accordingly. After bit-rate matching is complete, the boot program proceeds to the inquiry-and-selection state.
2. Inquiry-and-selection state

   In this state, the boot program responds to inquiry commands from the host. The device, clock mode, and bit rate are selected in this state. After making these selections, the boot program enters the programming/erasure state in response to the transition-to-programming/erasure state command. The boot program transfers the erasure program to RAM and executes erasure of the user MAT and user boot MAT before it enters the programming/erasure state.
3. Programming/erasure state

   In this state, programming/erasure are executed. The boot program transfers the program for programming/erasure to RAM in line with the command received from the host and executes programming/erasure. It also performs sum checking and blank checking as directed by the respective commands.

Figure 23.22 shows the flow of processing by the boot program.

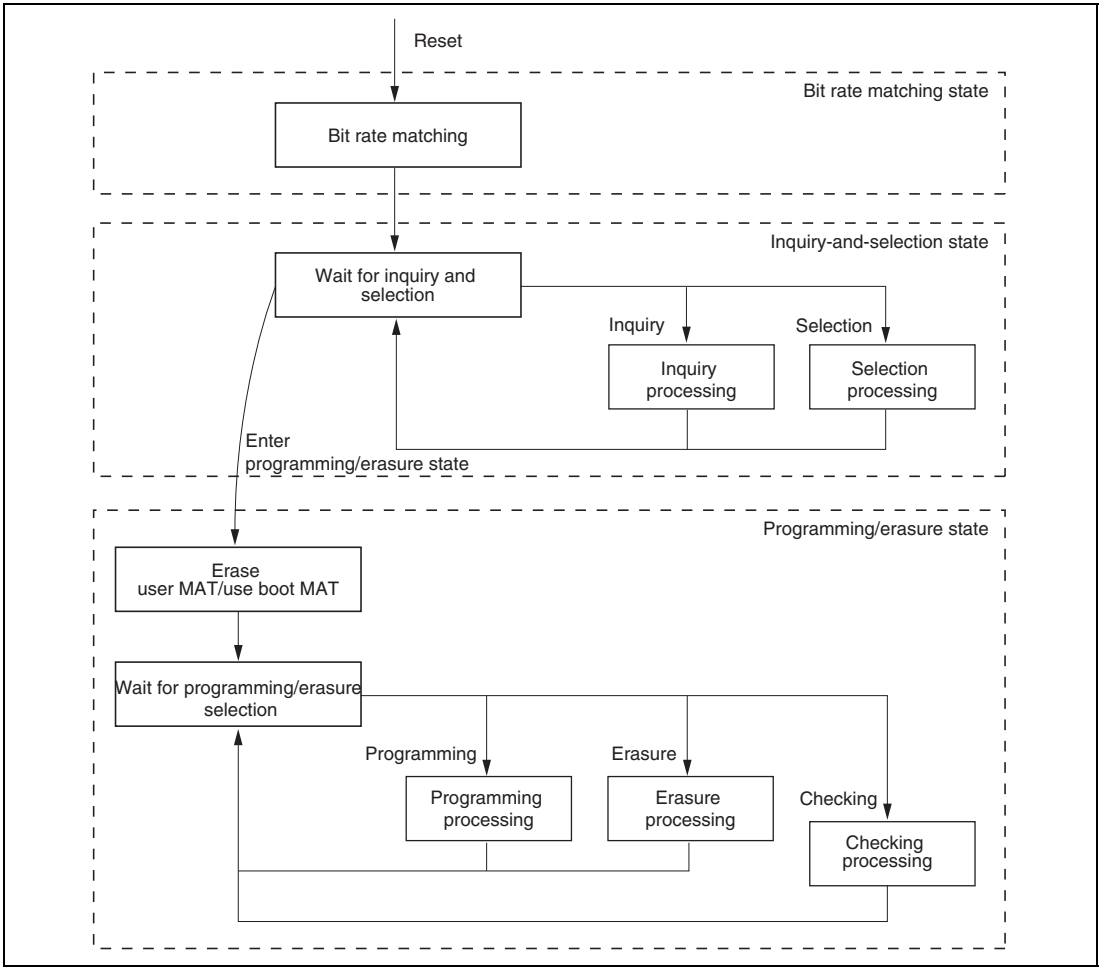


Figure 23.22 Flow of Processing by the Boot Program

- Bit-rate matching state

In bit-rate matching, the boot program measures the low-level intervals in a signal carrying H'00 data that is transmitted by the host, and calculates the bit rate from this. The bit rate can be changed by the new-bit-rate selection command. On completion of bit-rate matching, the boot program goes to the inquiry and selection state. The sequence of processing in bit-rate matching is shown in figure 23.23.

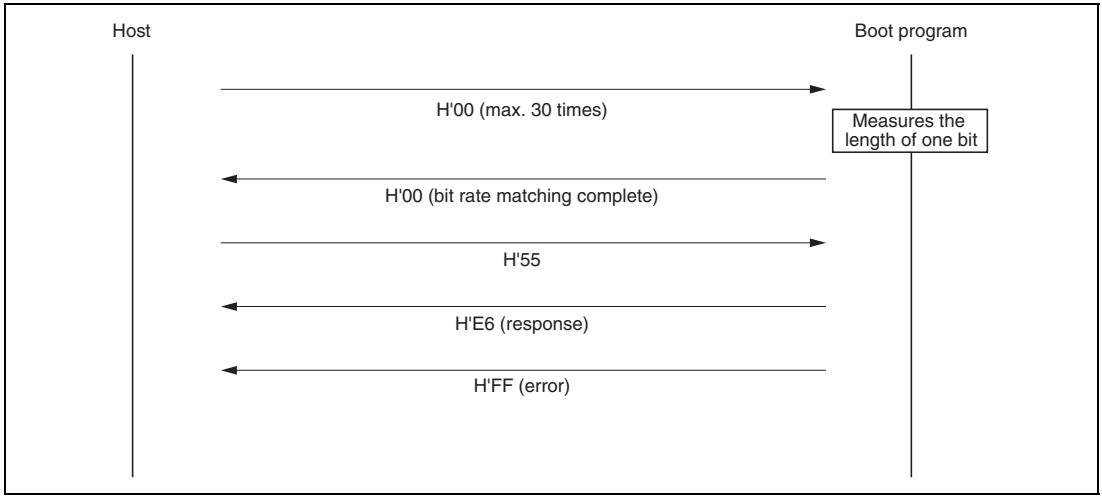


Figure 23.23 Sequence of Bit-Rate Matching

- Communications protocol

Formats in the communications protocol between the host and boot program after completion of the bit-rate matching are as follows.

1. One-character command or one-character response

   A command or response consisting of a single character used for an inquiry or the ACK code indicating normal completion.
2. n-character command or n-character response

   A command or response that requires n bytes of data, which is used as a selection command or response to an inquiry. The length of programming data is treated separately below.
3. Error response

   Response to a command in case of an error: two bytes, consisting of the error response and error code.
4. 128-byte programming command

   The command itself does not include data-size information. The data length is known from the response to the command for inquiring about the programming size.
5. Response to a memory reading command

   This response includes four bytes of size information.

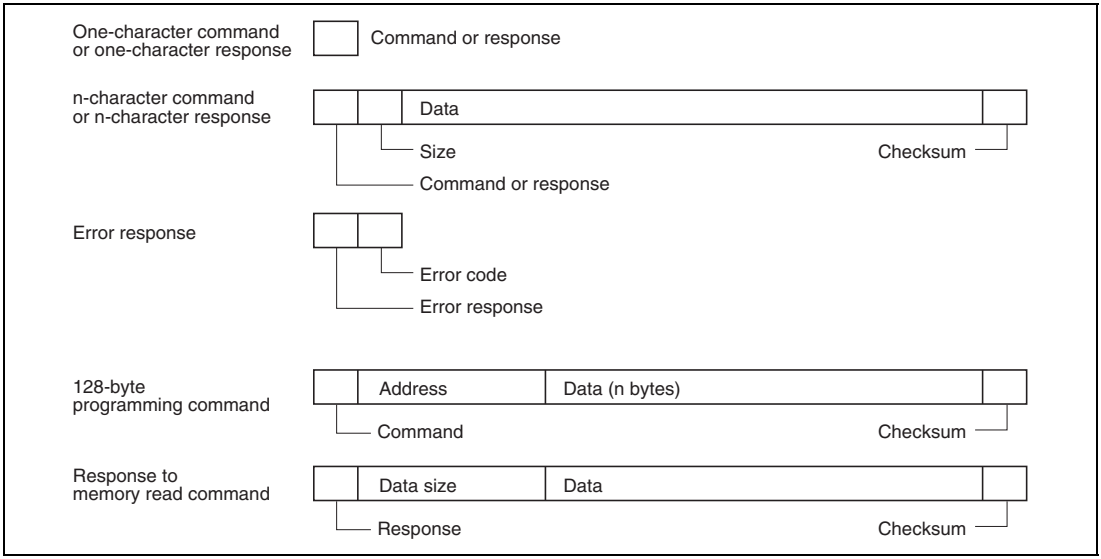


Figure 23.24 Formats in the Communications Protocol

— Command (1 byte): Inquiry, selection, programming, erasure, checking, etc.

— Response (1 byte): Response to an inquiry

— Size (one or two bytes): The length of data for transfer, excluding the command/response code, size, and checksum.

— Data (n bytes): Particular data for the command or response

— Checksum (1 byte): Set so that the total sum of byte values from the command code to the checksum is H'00 in the lower-order 1 byte.

— Error response (1 byte): Error response to a command

— Error code (1 byte): Indicates the type of error.

— Address (4 bytes): Address for programming

— Data (n bytes): Data to be programmed. "n" is known from the response to the command used to inquire about the programming size.

— Data size (4 bytes): Four-byte field included in the response to a memory reading command.

- **Inquiry-and-Selection State**

In this state, the boot program returns information on the flash ROM in response to inquiry commands sent from the host, and selects the device, clock mode, and bit rate in response to the respective selection commands.

The inquiry and selection commands are listed in table 23.13.

Table 23.13 Inquiry and Selection Commands

Command	Command Name	Function
H'20	Inquiry on supported devices	Requests the device codes and their respective boot program names.
H'10	Device selection	Selects a device code.
H'21	Inquiry on clock modes	Requests the number of available clock modes and their respective values.
H'11	Clock-mode selection	Selects a clock mode.
H'22	Inquiry on frequency multipliers	Requests the number of clock signals for which frequency multipliers and divisors are selectable, the number of multiplier and divisor settings for the respective clocks, and the values of the multipliers and divisors.
H'23	Inquiry on operating frequency	Requests the minimum and maximum values for operating frequency of the main clock and peripheral clock.
H'24	Inquiry on user boot MATs	Requests the number of user boot MAT areas along with their start and end addresses.
H'25	Inquiry on user MATs	Requests the number of user MAT areas along with their start and end addresses.
H'26	Inquiry on erasure blocks	Requests the number of erasure blocks along with their start and end addresses.
H'27	Inquiry on programming size	Requests the unit of data for programming.
H'3F	New bit rate selection	Selects a new bit rate.
H'40	Transition to programming/erasure state	On receiving this command, the boot program erases the user MAT and user boot MAT and enters the programming/erasure state.
H'4F	Inquiry on boot program state	Requests information on the current state of boot processing.

The selection commands should be sent by the host in this order: device selection (H'10), clock-mode selection (H'11), new bit rate selection (H'3F). These commands are mandatory. If the same selection command is sent two or more times, the command that is sent last is effective.

All commands in the above table, except for the boot program state inquiry command (H'4F), are valid until the boot program accepts the transition-to-programming/erasure state command (H'40). That is, until the transition command is accepted, the host can continue to send commands listed in the above table until it has made the necessary inquiries and selections. The host can send the boot program state inquiry command (H'4F) even after acceptance of the transition-to-programming/erasure state command (H'40) by the boot program.

(1) Inquiry on supported devices

In response to the inquiry on supported devices, the boot program returns the device codes of the devices it supports and the product names of their respective boot programs.

Command	H'20

— Command H'20 (1 byte): Inquiry on supported devices

Response	H'30	Size	No. of devices	
	Number of characters	Device code		Product name
	…			
	SUM			

— Response H'30 (1 byte): Response to the inquiry on supported devices

— Size (1 byte): The length of data for transfer excluding the command code, this field (size), and the checksum. Here, it is the total number of bytes taken up by the number of devices, number of characters, device code, and product name fields.

— Number of devices (1 byte): The number of device models supported by the boot program embedded in the microcomputer.

— Number of characters (1 byte): The number of characters in the device code and product name fields.

— Device code (4 bytes): Device code of a supported device (ASCII encoded)

— Product name (n bytes): Product code of the boot program (ASCII encoded)

— SUM (1 byte): Checksum
This is set so that the total sum of all bytes from the command code to the checksum is H'00.

(2) Device selection

In response to the device selection command, the boot program sets the specified device as the selected device. The boot program will return the information on the selected device in response to subsequent inquiries.

Command	H'10	Size	Device code	SUM

- Command H'10 (1 byte): Device selection
- Size (1 byte): Number of characters in the device code (fixed at 4)
- Device code (4 bytes): A device code that was returned in response to an inquiry on supported devices (ASCII encoded)
- SUM (1 byte): Checksum

Response	H'06

- Response H'06 (1 byte): Response to device selection
  The ACK code is returned when the specified device code matches one of the supported devices.

Error response	H'90	ERROR

- Error response H'90 (1 byte): Error response to device selection
- ERROR (1 byte): Error code
  H'11: Sum-check error
  H'21: Non-matching device code

(3) Inquiry on clock modes

In response to the inquiry on clock modes, the boot program returns the number of available clock modes.

Command	H'21

- Command H'21 (1 byte): Inquiry on clock modes

Response	H'31	Size	Mode	…	SUM

— Response H'31 (1 byte): Response to the inquiry on clock modes
— Size (1 byte): The total length of the number of modes and mode data fields.
— Mode (1 byte): Selectable clock mode (example: H'01 denotes clock mode 1)
— SUM (1 byte): Checksum

(4) Clock-mode selection

In response to the clock-mode selection command, the boot program sets the specified clock mode. The boot program will return the information on the selected clock mode in response to subsequent inquiries.

Command	H'11	Size	Mode	SUM

— Command H'11 (1 byte): Clock mode selection
— Size (1 byte): Number of characters in the clock-mode field (fixed at 1)
— Mode (1 byte): A clock mode returned in response to the inquiry on clock modes
— SUM (1 byte): Checksum

Response	H'06

— Response H'06 (1 byte): Response to clock mode selection
The ACK code is returned when the specified clock-mode matches one of the available clock modes.

Error response	H'91	ERROR

— Error response H'91 (1 byte): Error response to clock mode selection
— ERROR (1 byte): Error code
H'11: Sum-check error
H'22: Non-matching clock mode

(5) Inquiry on frequency multipliers

In response to the inquiry on frequency multipliers, the boot program returns information on the settable frequency multipliers or divisors.

Command	H'22

— Command H'22 (1 byte): Inquiry on frequency multipliers

Response	H'32	Size	No. of operating clocks					
	No. of multipliers	Multiplier	…					
	…							
	SUM							

— Response H'32 (1 byte): Response to the inquiry on frequency multipliers

— Size (1 byte): The total length of the number of operating clocks, number of multipliers, and multiplier fields.

— Number of operating clocks (1 byte): The number of operating clocks for which multipliers can be selected
(for example, if frequency multiplier settings can be made for the frequencies of the main and peripheral operating clocks, the value should be H'02).

— Number of multipliers (1 byte): The number of multipliers selectable for the operating frequency of the main or peripheral modules

— Multiplier (1 byte):
Multiplier: Numerical value in the case of frequency multiplication (e.g. H'04 for ×4)
Divisor: Two's complement negative numerical value in the case of frequency division (e.g. H'FE [-2] for ×1/2)
As many multiplier fields are included as there are multipliers or divisors, and combinations of the number of multipliers and multiplier fields are repeated as many times as there are operating clocks.

— SUM (1 byte): Checksum

(6) Inquiry on operating frequency

In response to the inquiry on operating frequency, the boot program returns the number of operating frequencies and the maximum and minimum values.

Command	H'23

— Command H'23 (1 byte): Inquiry on operating frequency

Response	H'33	Size	No. of operating clocks
	Operating freq. (min)		Operating freq. (max)
	…		
	SUM		

— Response H'33 (1 byte): Response to the inquiry on operating frequency

— Size (1 byte): The total length of the number of operating clocks, and maximum and minimum values of operating frequency fields.

— Number of operating clocks (1 byte): The number of operating clock frequencies required within the device.
For example, the value two indicates main and peripheral operating clock frequencies.

— Minimum value of operating frequency (2 bytes): The minimum frequency of a frequency-multiplied or -divided clock signal.
The value in this field and in the maximum value field is the frequency in MHz to two decimal places, multiplied by 100 (for example, if the frequency is 20.00 MHz, the value multiplied by 100 is 2000, so H'07D0 is returned here).

— Maximum value of operating frequency (2 bytes): The maximum frequency of a frequency-multiplied or -divided clock signal.
As many pairs of minimum/maximum values are included as there are operating clocks.

— SUM (1 byte): Checksum

(7) Inquiry on user boot MATs

In response to the inquiry on user boot MATs, the boot program returns the number of user boot MAT areas and their addresses.

Command	H'24

— Command H'24 (1 byte): Inquiry on user boot MAT information

Response	H'34	Size	No. of areas	
	First address of the area			Last address of the area
	…			
	SUM			

— Response H'34 (1 byte): Response to the inquiry on user boot MATs
— Size (1 byte): The total length of the number of areas and first and last address fields.
— Number of areas (1 byte): The number of user boot MAT areas.
H'01 is returned if the entire user boot MAT area is continuous.
— First address of the area (4 bytes)
— Last address of the area (4 bytes)
As many pairs of first and last address field are included as there are areas.
— SUM (1 byte): Checksum

(8) Inquiry on user MATs

In response to the inquiry on user MATs, the boot program returns the number of user MAT areas and their addresses.

Command	H'25

— Command H'25 (1 byte): Inquiry on user MAT information

Response	H'35	Size	No. of areas	
	First address of the area			Last address of the area
	…			
	SUM			

— Response H'35 (1 byte): Response to the inquiry on user MATs
— Size (1 byte): The total length of the number of areas and first and last address fields.
— Number of areas (1 byte): The number of user MAT areas.
H'01 is returned if the entire user MAT area is continuous.
— First address of the area (4 bytes)
— Last address of the area (4 bytes)
As many pairs of first and last address field are included as there are areas.
— SUM (1 byte): Checksum

(9) Inquiry on erasure blocks

In response to the inquiry on erasure blocks, the boot program returns the number of erasure blocks in the user MAT and the addresses where each block starts and ends.

Command	H'26

— Command H'26 (1 byte): Inquiry on erasure blocks

Response	H'36	Size	No. of blocks	
	First address of the block			Last address of the block
	…			
	SUM			

— Response H'36 (1 byte): Response to the inquiry on erasure blocks
— Size (2 bytes): The total length of the number of blocks and first and last address fields.
— Number of blocks (1 byte): The number of erasure blocks in flash memory
— First address of the block (4 bytes)
— Last address of the block (4 bytes)
As many pairs of first and last address data are included as there are blocks.
— SUM (1 byte): Checksum

(10) Inquiry on programming size

In response to the inquiry on programming size, the boot program returns the size, in bytes, of the unit for programming.

Command	H'27	

— Command H'27 (1 byte): Inquiry on programming size

Response	H'37	Size	Programming size	SUM

— Response H'37 (1 byte): Response to the inquiry on programming size
— Size (1 byte): The number of characters in the programming size field (fixed at 2)
— Programming size (2 bytes): The size of the unit for programming
This is the unit for the reception of data to be programmed.
— SUM (1 byte): Checksum

(11) New bit rate selection

In response to the new-bit-rate selection command, the boot program changes the bit rate setting to the new bit rate and, if the setting was successful, responds to the ACK sent by the host by returning another ACK at the new bit rate.

The new-bit-rate selection command should be sent after clock-mode selection.

Command	H'3F	Size	Bit rate	Input frequency
	No. of multipliers	Multiplier 1	Multiplier 2	
	SUM			

— Command H'3F (1 byte): New bit rate selection
— Size (1 byte): The total length of the bit rate, input frequency, number of multipliers, and multiplier fields
— Bit rate (2 bytes): New bit rate
The bit rate value divided by 100 should be set here (for example, to select 19200 bps, the set H'00C0, which is 192 in decimal notation).
— Input frequency (2 bytes): The frequency of the clock signal fed to the boot program
This should be the frequency in MHz to the second decimal place, multiplied by 100 (for example, if the frequency is 28.882 MHz, the values is truncated to the second decimal place and multiplied by 100, making 2888; so H'0B48 should be set in this field).

— Number of multipliers (1 byte): The number of selectable frequency multipliers and divisors for the device.
This is normally 2, which indicates the main operating frequency and the operating frequency of the peripheral modules.

— Multiplier 1 (1 byte): Multiplier or divisor for the main operating frequency
Multiplier: Numerical value of the frequency multiplier (e.g. H'04 for ×4)
Divisor: Two's complement negative numerical value in the case of frequency division (e.g. H'FE [-2] for ×1/2)

— Multiplier 2 (1 byte): Multiplier or divisor for the peripheral operating frequency
Multiplier: Numerical value of the frequency multiplier (e.g. H'04 for ×4)
Divisor: Two's complement negative numerical value in the case of frequency division (e.g. H'FE [-2] for ×1/2)

— SUM (1 byte): Checksum

Response	H'06

— Response H'06 (1 byte): Response to the new-bit-rate selection command
The ACK code is returned if the specified bit rate was selectable.

Error response	H'BF	ERROR

— Error response H'BF (1 byte): Error response to new bit rate selection

— ERROR (1 byte): Error code

H'11: Sum-check error

H'24: Bit rate selection error (the specified bit rate is not selectable).

H'25: Input frequency error (the specified input frequency is not within the range from the minimum to the maximum value).

H'26: Frequency multiplier error (the specified multiplier does not match an available one).

H'27: Operating frequency error (the specified operating frequency is not within the range from the minimum to the maximum value).

The received data are checked in the following ways.

1. Input frequency

   The value of the received input frequency is checked to see if it is within the range of the minimum and maximum values of input frequency for the selected clock mode of the selected device. A value outside the range generates an input frequency error.

2. Multiplier

   The value of the received multiplier is checked to see if it matches a multiplier or divisor that is available for the selected clock mode of the selected device. A value that does not match an available ratio generates a frequency multiplier error.

3. Operating frequency

   The operating frequency is calculated from the received input frequency and the frequency multiplier or divisor. The input frequency is the frequency of the clock signal supplied to the LSI, while the operating frequency is the frequency at which the LSI is actually driven. The following formulae are used for this calculation.

   Operating frequency = input frequency × multiplier, or

   Operating frequency = input frequency / divisor

   The calculated operating frequency is checked to see if it is within the range of the minimum and maximum values of the operating frequency for the selected clock mode of the selected device. A value outside the range generates an operating frequency error.

4. Bit rate

   From the peripheral operating frequency (Pϕ) and the bit rate (B), the value (= n) of the clock select bits (CKS) in the serial mode register (SCSMR) and the value (= N) of the bit rate register (SCBRR) are calculated, after which the error in the bit rate is calculated. This error is checked to see if it is smaller than 4%. A result greater than or equal to 4% generates a bit rate selection error. The following formula is use to calculate the error.

$$\text{Error (\%)} = \left\{ \left[\frac{P\phi \times 10^6}{(N+1) \times B \times 64 \times 2^{2n-1}} \right] - 1 \right\} \times 100$$

When the new bit rate is selectable, the boot program returns an ACK code to the host and then makes the register setting to select the new bit rate. The host then sends an ACK code at the new bit rate, and the boot program responds to this with another ACK code, this time at the new bit rate.

Acknowledge	H'06

— Acknowledge H'06 (1 byte): The ACK code sent by the host to acknowledge the new bit rate.

Response	H'06

— Response H'06 (1 byte): The ACK code transferred in response to acknowledgement of the new bit rate

The sequence of new bit rate selection is shown in figure 23.25.

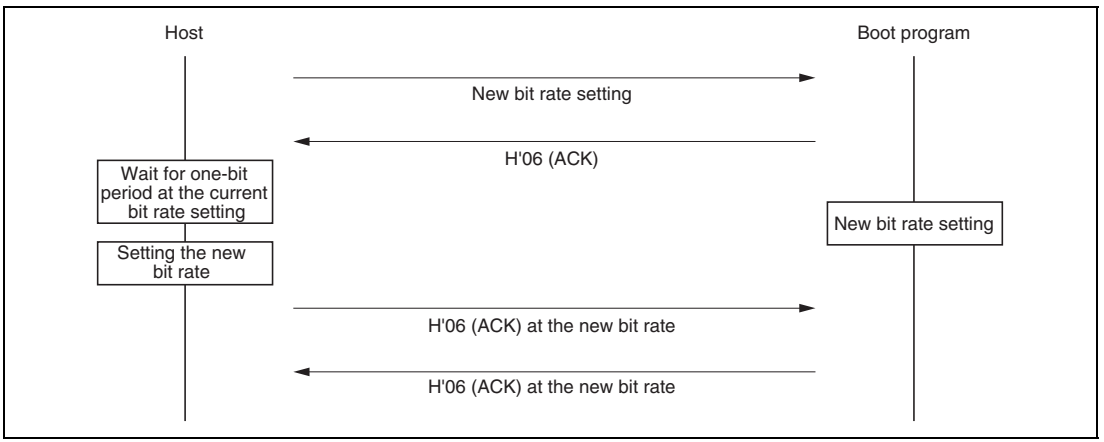


Figure 23.25 Sequence of New Bit Rate Selection

(12) Transition to the programming/erasure state

In response to the transition to the programming/erasure state command, the boot program transfers the erasing program and runs it to erase any data in the user MAT and then the user boot MAT. On completion of this erasure, the boot program returns the ACK code and enters the programming/erasure state.

Before sending the programming selection command and data for programming, the host must select the device, clock mode, and new bit rate for the LSI by issuing the device selection command, clock-mode selection command, new-bit-rate selection command, and then initiate the transition to the programming/erasure state by sending the corresponding command to the boot program.

Command	H'40

— Command H'40 (1 byte): Transition to programming/erasure state

Response	H'06

— Response H'06 (1 byte): Response to the transition-to-programming/erasure state command
This is returned as ACK when erasure of the user boot MAT and user MAT has succeeded after transfer of the erasure program.

Error response	H'C0	H'51

— Error response H'C0 (1 byte): Error response to the transition-to-programming/erasure state command
— ERROR (1 byte): Error code
H'51: Erasure error (Erasure did not succeed because of an error.)

- Command Error

Command errors are generated by undefined commands, commands sent in an incorrect order, and the inability to accept a command. For example, sending the clock-mode selection command before device selection or an inquiry command after the transition-to-programming/erasure state command generates a command error.

Error response	H'80	H'xx

— Error response H'80 (1 byte): Command error
— Command H'xx (1 byte): Received command

- Order of Commands

In the inquiry-and-selection state, commands should be sent in the following order.

1. Send the inquiry on supported devices command (H'20) to get the list of supported devices.
2. Select a device from the returned device information, and send the device selection command (H'10) to select that device.
3. Send the inquiry on clock mode command (H'21) to get the available clock modes.
4. Select a clock mode from among the returned clock modes, and send the clock-mode selection command (H'11).
5. After selection of the device and clock mode, send the commands to inquire about frequency multipliers (H'22) and operating frequencies (H'23) to get the information required to select a new bit rate.
6. Taking into account the returned information on the frequency multipliers and operating frequencies, send a new-bit-rate selection command (H'3F).
7. After the device and clock mode have been selected, get the information required for programming and erasure of the user boot MAT and user MAT by sending the commands to inquire about the user boot MAT (H'24), user MAT (H'25), erasure block (H'26), and programming size (H'27).
8. After making all necessary inquiries and the new bit rate selection, send the transition-to-programming/erasure state command (H'40) to place the boot program in the programming/erasure state.

- **Programming/Erasure State**

In this state, the boot program must select the form of programming corresponding to the programming-selection command and then write data in response to 128-byte programming commands, or perform erasure in block units in response to the erasure-selection and block-erasure commands.

The programming and erasure commands are listed in table 23.14.

Table 23.14 Programming and Erasure Commands

Command	Command Name	Function
H'42	Selection of user boot MAT programming	Selects transfer of the program for user boot MAT programming.
H'43	Selection of user MAT programming	Selects transfer of the program for user MAT programming.
H'50	128-byte programming	Executes 128-byte programming.
H'48	Erasure selection	Selects transfer of the erasure program.
H'58	Block erasure	Executes erasure of the specified block.
H'52	Memory read	Reads from memory.
H'4A	Sum checking of user boot MAT	Executes sum checking of the user boot MAT.
H'4B	Sum checking of user MAT	Executes sum checking of the user MAT.
H'4C	Blank checking of user boot MAT	Executes blank checking of the user boot MAT.
H'4D	Blank checking of user MAT	Executes blank checking of the user MAT.
H'4F	Inquiry on boot program state	Requests information on the state of boot processing.

- Programming

Programming is performed by issuing a programming-selection command and the 128-byte programming command.

Firstly, the host issues the programming-selection command to select the MAT to be programmed. Two programming-selection commands are provided for the selection of either of the two target areas.

1. Selection of user boot MAT programming
2. Selection of user MAT programming

Next, the host issues a 128-byte programming command. 128 bytes of data for programming by the method selected by the preceding programming selection command are expected to follow the command. To program more than 128 bytes, repeatedly issue 128-byte programming commands. To terminate programming, the host should send another 128-byte programming command with the address H'FFFFFFFF. On completion of programming, the boot program waits for the next programming/erasure selection command.

To then program the other MAT, start by sending the programming select command.

The sequence of programming by programming-selection and 128-byte programming commands is shown in figure 23.26.

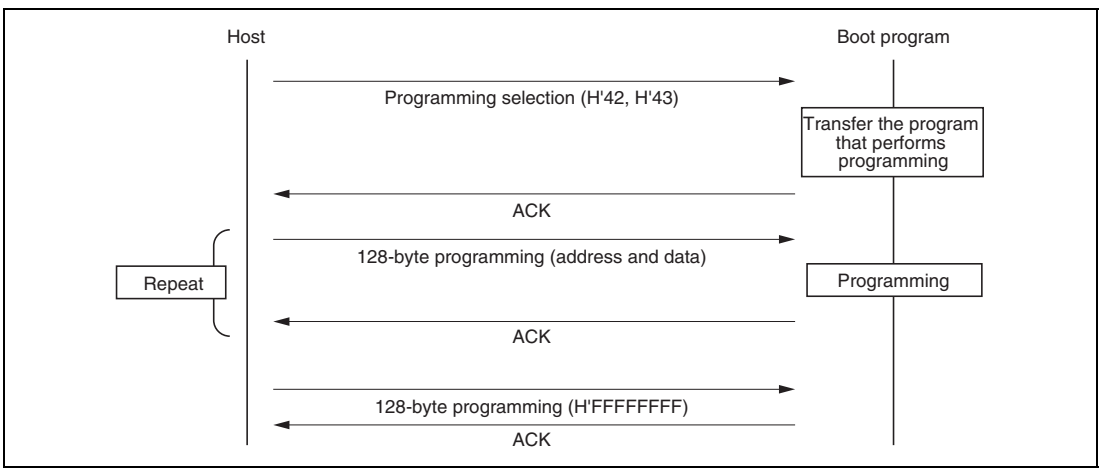


Figure 23.26 Sequence of Programming

(1) Selection of user boot MAT programming

In response to the command for selecting programming of the user boot MAT, the boot program transfers the corresponding flash-writing program, i.e. the program for writing to the user boot MAT.

Command	H'42

— Command H'42 (1 byte): Selects programming of the user boot MAT.

Response	H'06

— Response H'06 (1 byte): Response to selection of user boot MAT programming
This ACK code is returned after transfer of the program that performs writing to the user boot MAT.

Error response	H'C2	ERROR

— Error response H'C2 (1 byte): Error response to selection of user boot MAT programming
— ERROR (1 byte): Error code
H'54: Error in selection processing (processing was not completed because of a transfer error)

(2) Selection of user MAT programming

In response to the command for selecting programming of the user MAT, the boot program transfers the corresponding flash-writing program, i.e. the program for writing to the user MAT.

Command	H'43

— Command H'43 (1 byte): Selects programming of the user MAT.

Response	H'06

— Response H'06 (1 byte): Response to selection of user MAT programming
This ACK code is returned after transfer of the program that performs writing to the user MAT.

Error response	H'C3	ERROR

— Error response H'C3 (1 byte): Error response to selection of user MAT programming

— ERROR (1 byte): Error code
H'54: Error in selection processing (processing was not completed because of a transfer error)

(3) 128-byte programming

In response to the 128-byte programming command, the boot program executes the flash-writing program transferred in response to the command to select programming of the user boot MAT or user MAT.

Command	H'50	Address for programming						
	Data	…						
	…							
	SUM							

— Command H'50 (1 byte): 128-byte programming

— Address for programming (4 bytes): Address where programming starts
This should be the address of a 128-byte boundary.
[Example] H'00, H01, H'00, H'00: H'00010000

— Programming data (n bytes): Data for programming
The length of the programming data is the size returned in response to the programming size inquiry command.

— SUM (1 byte): Checksum

Response	H'06

— Response H'06 (1 byte): Response to 128-byte programming
The ACK code is returned on completion of the requested programming.

Error response	H'D0	ERROR

— Error response H'D0 (1 byte): Error response to 128-byte programming
— ERROR (1 byte): Error code
H'11: Sum-check error
H'2A: Address error (the address is not within the range for the selected MAT)
H'53: Programming error (programming failed because of an error in programming)

The specified address should be on a boundary corresponding to the unit of programming (programming size). For example, when programming 128 bytes of data, the lowest byte of the address should be either H'00 or H'80. When less than 128 bytes of data are to be programmed, the host should transmit the data after padding the vacant bytes with H'FF.

To terminate programming of a given MAT, send a 128-byte programming command with the address field H'FFFFFFFF. This informs the boot program that all data for the selected MAT have been sent; the boot program then waits for the next programming/erasure selection command.

Command	H'50	Address for programming	SUM

— Command H'50 (1 byte): 128-byte programming
— Address for programming (4 bytes): Terminating code (H'FF, H'FF, H'FF, H'FF)
— SUM (1 byte): Checksum

Response	H'06

— Response H'06 (1 byte): Response to 128-byte programming
This ACK code is returned on completion of the requested programming.

Error response	H'D0	ERROR

— Error response H'D0 (1 byte): Error response to 128-byte programming
— ERROR (1 byte): Error code
H'11: Sum-check error
H'53: Programming error

- Erasure

Erasure is performed by issuing the erasure selection command and then one or more block erasure commands.

Firstly, the host sends the erasure selection command to select erasure; after that, it sends a block erasure command to actually erase a specific block. To erase multiple blocks, send further block erasure commands. To terminate erasure, the host should send a block erasure command with the block number H'FF. After this, the boot program waits for the next programming/erasure selection command.

The sequence of erasure by the erasure selection command and block erasure command is shown in figure 23.27.

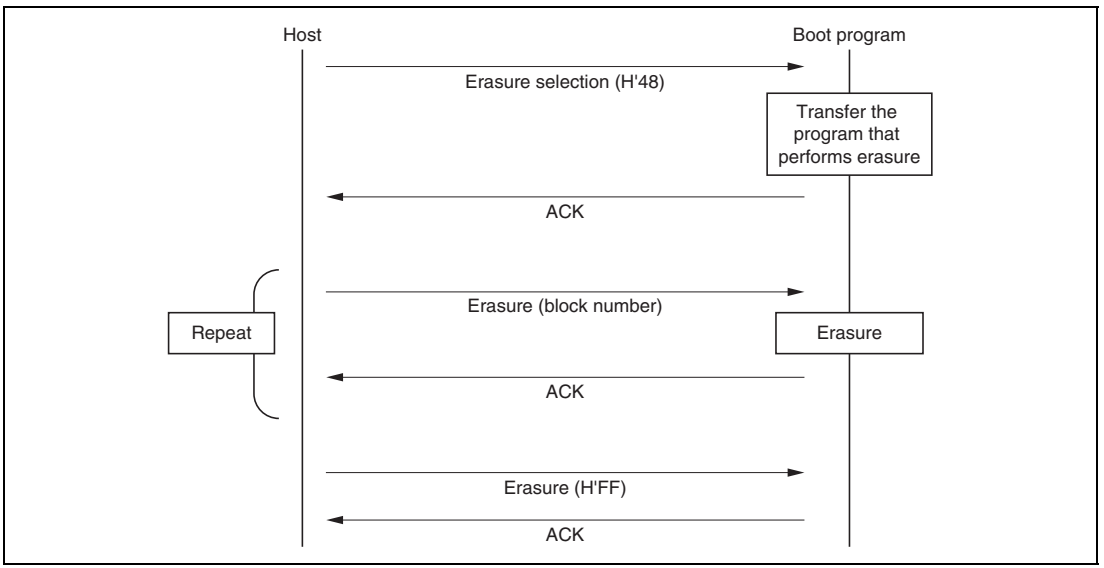


Figure 23.27 Sequence of Erasure

(1) Select erasure

In response to the erasure selection command, the boot program transfers the program that performs erasure, i.e. erases data in the user MAT.

Command	H'48	

— Command H'48 (1 byte): Selects erasure.

Response	H'06	

— Response H'06 (1 byte): Response to selection of erasure
This ACK code is returned after transfer of the program that performs erasure.

Error response	H'C8	ERROR

— Error response H'C8 (1 byte): Error response to selection of erasure
— ERROR (1 byte): Error code
H'54: Error in selection processing (processing was not completed because of a transfer error.)

(2) Block erasure

In response to the block erasure command, the boot program erases the data in a specified block of the user MAT.

Command	H'58	Size	Block number	SUM

— Command H'58 (1 byte): Erasure of a block
— Size (1 byte): The number of characters in the block number field (fixed at 1)
— Block number (1 byte): Block number of the block to be erased
— SUM (1 byte): Checksum

Response	H'06	

— Response H'06 (1 byte): Response to the block erasure command
This ACK code is returned when the block has been erased.

Error response	H'D8	ERROR

— Error response H'D8 (1 byte): Error response to the block erasure command
— ERROR (1 byte): Error code
H'11: Sum-check error
H'29: Block number error (the specified block number is incorrect.)
H'51: Erasure error (an error occurred during erasure.)

On receiving the command with H'FF as the block number, the boot program stops erasure processing and waits for the next programming/erasure selection command.

Command	H'58	Size	Block number	SUM

— Command H'58 (1 byte): Erasure of a block
— Size (1 byte): The number of characters in the block number field (fixed at 1)
— Block number (1 byte): H'FF (erasure terminating code)
— SUM (1 byte): Checksum

Response	H'06

— Response H'06 (1 byte): ACK code to indicate response to the request for termination of erasure

To perform erasure again after having issued the command with the block number specified as H'FF, execute the process from the selection of erasure.

- Memory read

In response to the memory read command, the boot program returns the data from the specified address.

Command	H'52	Size	Area	First address for reading
	Amount to read			SUM

— Command H'52 (1 byte): Memory read
— Size (1 byte): The total length of the area, address for reading, and amount to read fields (fixed value of 9)

— Area (1 byte):
H'00: User boot MAT
H'01: User MAT
An incorrect area specification will produce an address error.
— Address where reading starts (4 bytes)
— Amount to read (4 bytes): The amount of data to be read
— SUM (1 byte): Checksum

Response

H'52	Amount to read						
Data	…						
SUM							

— Response H'52 (1 byte): Response to the memory read command
— Amount to read (4 bytes): The amount to read as specified in the memory read command
— Data (n bytes): The specified amount of data read out from the specified address
— SUM (1 byte): Checksum

Error response

H'D2	ERROR

— Error response H'D2 (1 byte): Error response to memory read command
— ERROR (1 byte): Error code
H'11: Sum-check error
H'2A: Address error (the address specified for reading is beyond the range of the MAT)
H'2B: Size error (the specified amount is greater than the size of the MAT,
the last address for reading as calculated from the specified address for the start of reading and the amount to read is beyond the MAT area, or "0" was specified as the amount to read)

- Sum checking of the user boot MAT

In response to the command for sum checking of the user boot MAT, the boot program adds all bytes of data in the user boot MAT and returns the result.

Command	H'4A

— Command H'4A (1 byte): Sum checking of the user boot MAT

Response	H'5A	Size	Checksum for the MAT	SUM

— Response H'5A (1 byte): Response to sum checking of the user boot MAT
— Size (1 byte): The number of characters in the checksum for the MAT (fixed at 4)
— Checksum for the MAT (4 bytes): Result of checksum calculation for the user boot MAT: the total of all data in the MAT, in byte units.
— SUM (1 byte): Checksum (for the transmitted data)

- Sum checking of the user MAT

In response to the command for sum checking of the user MAT, the boot program adds all bytes of data in the user MAT and returns the result.

Command	H'4B

— Command H'4B (1 byte): Sum checking of the user MAT

Response	H'5B	Size	Checksum for the MAT	SUM

— Response H'5B (1 byte): Response to sum checking of the user MAT
— Size (1 byte): The number of characters in the checksum for the MAT (fixed at 4)
— Checksum for the MAT (4 bytes): Result of checksum calculation for the user MAT: the total of all data in the MAT, in byte units.
— SUM (1 byte): Checksum (for the transmitted data)

- Blank checking of the user boot MAT

In response to the command for blank checking of the user boot MAT, the boot program checks to see if the whole of the user boot MAT is blank; the value returned indicates the result.

Command	H'4C

— Command H'4C (1 byte): Blank checking of the user boot MAT

Response	H'06

— Response H'06 (1 byte): Response to blank checking of the user boot MAT
This ACK code is returned when the whole area is blank (all bytes are H'FF).

Error response	H'CC	H'52

— Error response H'CC (1 byte): Error response to blank checking of the user boot MAT
— Error code H'52 (1 byte): Non-erased error

- Blank checking of the user MAT

In response to the command for blank checking of the user MAT, the boot program checks to see if the whole of the user MAT is blank; the value returned indicates the result.

Command	H'4D

— Command H'4D (1 byte): Blank checking of the user boot MAT

Response	H'06

— Response H'06 (1 byte): Response to blank checking of the user MAT
The ACK code is returned when the whole area is blank (all bytes are H'FF).

Error response	H'CD	H'52

— Error response H'CD (1 byte): Error response to blank checking of the user MAT
— Error code H'52 (1 byte): Non-erased error

- Inquiry on boot program state

In response to the command for inquiry on the state of the boot program, the boot program returns an indicator of its current state and error information. This inquiry can be made in the inquiry-and-selection state or the programming/erasure state.

Command	H'4F				

— Command H'4F (1 byte): Inquiry on boot program state

Response	H'5F	Size	STATUS	ERROR	SUM

— Response H'5F (1 byte): Response to the inquiry regarding boot-program state

— Size (1 byte): The number of characters in STATUS and ERROR (fixed at 2)

— STATUS (1 byte): State of the standard boot program
See table 23.15, Status Codes.

— ERROR (1 byte): Error state (indicates whether the program is in normal operation or an error has occurred)
ERROR = 0: Normal
ERROR ≠ 0: Error
See table 23.16, Error Codes.

— SUM (1 byte): Checksum

Table 23.15 Status Codes

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 programming/erasure status (bit-rate selection complete)
H'31	Erasing the user MAT or user boot MAT
H'3F	Waiting for programming/erasure selection (erasure complete)
H'4F	Waiting to receive data for programming (programming complete)
H'5F	Waiting for erasure block specification (erasure complete)

Table 23.16 Error Codes

Code	Description
H'00	No error
H'11	Sum check error
H'21	Non-matching device code error
H'22	Non-matching clock mode error
H'24	Bit-rate selection failure
H'25	Input frequency error
H'26	Frequency multiplier error
H'27	Operating frequency error
H'29	Block number error
H'2A	Address error
H'2B	Data length error (size error)
H'51	Erasure error
H'52	Non-erased error
H'53	Programming error
H'54	Selection processing error
H'80	Command error
H'FF	Bit-rate matching acknowledge error

23.9.2 Areas for Storage of the Procedural Program and Data for Programming

In the descriptions in the previous section, storable areas for the programming/erasing procedure programs and program data are assumed to be in on-chip RAM. However, the procedure programs and data can be stored in and executed from other areas (e.g. external address space) as long as the following conditions are satisfied.

1. The on-chip programming/erasing program is downloaded from the address set by FTDAR in on-chip RAM, therefore, this area is not available for use.
2. The on-chip programming/erasing program will use 128 bytes or more as a stack. Make sure this area is reserved.
3. Since download by setting the SCO bit to 1 will cause the MATs to be switched, it should be executed in on-chip RAM.
4. The flash memory is accessible until the start of programming or erasing, that is, until the result of downloading has been decided. When in a mode in which the external address space

is not accessible, such as single-chip mode, the required procedure programs, interrupt vector table, interrupt processing routine, and user branch program should be transferred to on-chip RAM before programming/erasing of the flash memory starts.

5. The flash memory is not accessible during programming/erasing operations. Therefore, the programming/erasing program must be downloaded to on-chip RAM in advance. Areas for executing each procedure program for initiating programming/erasing and the user program at the user branch destination for programming/erasing must be located in on-chip memory other than flash memory or the external address space.
6. After programming/erasing, access to flash memory is inhibited until FKEY is cleared.

   A reset state ($\overline{RES}$ = 0) for more than at least 100 μs must be taken when the LSI mode is changed to reset on completion of a programming/erasing operation.

   Transitions to the reset state during programming/erasing are inhibited. When the reset signal is accidentally input to the LSI, a longer period in the reset state than usual (100 μs) is needed before the reset signal is released.
7. Switching of the MATs by FMATS is needed for programming/erasing of the user MAT in user boot mode. The program which switches the MATs should be executed from the on-chip RAM. For details, see section 23.8.1, Switching between User MAT and User Boot MAT. Please make sure you know which MAT is selected when switching the MATs.
8. When the program data storage area indicated by the FMPDR parameter in the programming processing is within the flash memory area, an error will occur. Therefore, temporarily transfer the program data to on-chip RAM to change the address set in FMPDR to an address other than flash memory.

Based on these conditions, tables 23.17 and 23.18 show the areas in which the program data can be stored and executed according to the operation type and mode.

Table 23.17 Executable MAT

	Initiated Mode	
Operation	**User Program Mode**	**User Boot Mode***
Programming	Table 23.18 (1)	Table 23.18 (3)
Erasing	Table 23.18 (2)	Table 23.18 (4)

Note: * Programming/Erasing is possible to user MATs.

Table 23.18 (1) Usable Area for Programming in User Program Mode

		Storable/Executable Area			Selected MAT	
	Item	On-Chip RAM	User MAT	External Space	User MAT	Embedded Program Storage MAT
Programming procedure	Program data storage area	√	X*	√	—	—
	Selecting on-chip program to be downloaded	√	√	√	√	
	Writing H'A5 to key register	√	√	√	√	
	Writing 1 to SCO in FCCS (download)	√	X	X		√
	Key register clearing	√	√	√	√	
	Judging download result	√	√	√	√	
	Download error processing	√	√	√	√	
	Setting initialization parameters	√	√	√	√	
	Initialization	√	X	X	√	
	Judging initialization result	√	√	√	√	
	Initialization error processing	√	√	√	√	
	Writing H'5A to key register	√	√	√	√	
	Setting programming parameters	√	X	√	√	
	Programming	√	X	X	√	
	Judging programming result	√	X	√	√	
	Programming error processing	√	X	√	√	
	Key register clearing	√	X	√	√	

Note: * If the data has been transferred to on-chip RAM in advance, this area can be used.

Table 23.18 (2) Usable Area for Erasure in User Program Mode

		Storable/Executable Area			Selected MAT	
	Item	On-Chip RAM	User MAT	External Space	User MAT	Embedded Program Storage MAT
	Selecting on-chip program to be downloaded	√	√	√	√	
	Writing H'A5 to key register	√	√	√	√	
	Writing 1 to SCO in FCCS (download)	√	X	X		√
	Key register clearing	√	√	√	√	
	Judging download result	√	√	√	√	
	Download error processing	√	√	√	√	
	Setting initialization parameters	√	√	√	√	
	Initialization	√	X	X	√	
Erasing procedure	Judging initialization result	√	√	√	√	
	Initialization error processing	√	√	√	√	
	Writing H'5A to key register	√	√	√	√	
	Setting erasure parameters	√	X	√	√	
	Erasure	√	X	X	√	
	Judging erasure result	√	X	√	√	
	Erasing error processing	√	X	√	√	
	Key register clearing	√	X	√	√	

Table 23.18 (3) Usable Area for Programming in User Boot Mode

		Storable/Executable Area			Selected MAT		
	Item	On-Chip RAM	User Boot MAT	External Space	User MAT	User Boot MAT	Embedded Program Storage Area
Programming procedure	Program data storage area	√	X*[1]	√	—	—	—
	Selecting on-chip program to be downloaded	√	√	√		√	
	Writing H'A5 to key register	√	√	√		√	
	Writing 1 to SCO in FCCS (download)	√	X	X			√
	Key register clearing	√	√	√		√	
	Judging download result	√	√	√		√	
	Download error processing	√	√	√		√	
	Setting initialization parameters	√	√	√		√	
	Initialization	√	X	X		√	
	Judging initialization result	√	√	√		√	
	Initialization error processing	√	√	√		√	
	Switching MATs by FMATS	√	X	X	√		
	Writing H'5A to Key Register	√	X	√	√		

Table 23.18 (3) Usable Area for Programming in User Boot Mode (cont)

		Storable/Executable Area			Selected MAT		
	Item	On-Chip RAM	User Boot MAT	External Space	User MAT	User Boot MAT	Embedded Program Storage Area
Programming procedure	Setting programming parameters	√	X	√	√		
	Programming	√	X	X	√		
	Judging programming result	√	X	√	√		
	Programming error processing	√	X*2	√	√		
	Key register clearing	√	X	√	√		
	Switching MATs by FMATS	√	X	X		√	

Notes: 1. If the data has been transferred to on-chip RAM in advance, this area can be used.
2. If the MATs have been switched by FMATS in on-chip RAM, this MAT can be used.

Table 23.18 (4) Usable Area for Erasure in User Boot Mode

		Storable/Executable Area			Selected MAT		
	Item	On-Chip RAM	User Boot MAT	External Space	User MAT	User Boot MAT	Embedded Program Storage Area
Erasing procedure	Selecting on-chip program to be downloaded	√	√	√		√	
	Writing H'A5 to key register	√	√	√		√	
	Writing 1 to SCO in FCCS (download)	√	X	X			√
	Key register clearing	√	√	√		√	
	Judging download result	√	√	√		√	
	Download error processing	√	√	√		√	
	Setting initialization parameters	√	√	√		√	
	Initialization	√	X	X		√	
	Judging initialization result	√	√	√		√	
	Initialization error processing	√	√	√		√	
	Switching MATs by FMATS	√	X	X		√	
	Writing H'5A to key register	√	X	√	√		
	Setting erasure parameters	√	X	√	√		

Table 23.18 (4) Usable Area for Erasure in User Boot Mode (cont)

		Storable/Executable Area			Selected MAT		
	Item	On-Chip RAM	User Boot MAT	External Space	User MAT	User Boot MAT	Embedded Program Storage Area
↓	Erasure	√	X	X	√		
	Judging erasure result	√	X	√	√		
Erasing procedure	Erasing error processing	√	X*	√	√		
	Key register clearing	√	X	√	√		
	Switching MATs by FMATS	√	X	X		√	

Note: * If the MATs have been switched by FMATS in on-chip RAM, this MAT can be used.

23.10 Programmer Mode

In programmer mode, a PROM programmer can be used to perform programming/erasing via a socket adapter, just as for a discrete flash memory. Use a PROM programmer that supports the Renesas 256 or 512-kbyte flash memory on-chip MCU device type.

Section 24 Mask ROM

This LSI is available with 256 Kbytes of on-chip mask ROM. The on-chip ROM is connected to the CPU, direct memory access controller (DMAC), and data transfer controller (DTC) through a 32-bit data bus (figure 24.1). The CPU, DMAC, and DTC can access the on-chip ROM in 8, 16 and 32-bit widths. Data in the on-chip ROM can always be accessed in one cycle.

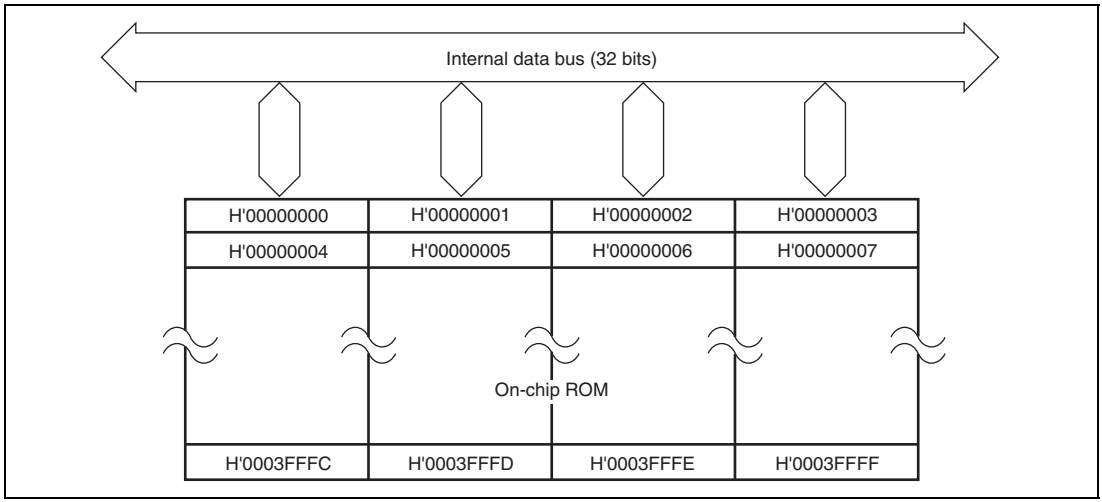


Figure 24.1 Mask ROM Block Diagram

The operating mode determines whether the on-chip ROM is valid or not. The operating mode is selected using mode-setting pins FWE, MD1, and MD0. If you are using the on-chip ROM, select mode 2 or mode 3; if you are not, select mode 0 or 1. The on-chip ROM is allocated to addresses H'00000000 to H'0003FFFF of memory area 0.

24.1 Usage Notes

24.1.1 Module Standby Mode Setting

Access to the mask ROM can be enabled/disabled by the standby control register. The initial value enables the mask ROM access. The mask ROM access is disabled by setting the module standby mode. For details, see section 26, Power-Down Modes.

Section 25 RAM

This LSI has an on-chip high-speed static RAM. The on-chip RAM is connected to the CPU by a 32-bit data bus (L bus), and to the direct memory access controller (DMAC), and data transfer controller (DTC) by a 32-bit data bus (I bus), enabling 8, 16, or 32-bit width access to data in the on-chip RAM.

The on-chip RAM is allocated to different addresses according to each product as shown in figure 25.1, and the on-chip RAM is divided into page 0 and page 1 based on the addresses. The on-chip RAM can be accessed from the CPU (via the L bus) and the DMAC/DTC (via the I bus). When different buses request to access the same page simultaneously, the priority becomes I bus (DMAC/DTC) > L bus (CPU). Since such kind of conflict degrades the RAM access performance, software should be created so as to avoid conflicts. For example, conflict does not occur when the buses access different pages. An access from the L bus (CPU) is a 1-cycle access as long as page conflict does not occur. The number of bus cycles in accesses from the I bus (DMAC/DTC) differ depending on the ratio between the internal clock (Iϕ) and bus clock (Bϕ), and the operating state of the DMAC/DTC. The contents of the on-chip RAM are retained in sleep mode or software standby mode, and at a power-on reset or manual reset. However, the contents of the on-chip RAM are not retained in deep software standby mode.

The on-chip RAM can be enabled or disabled by means of the RAME bit in the RAM control register (RAMCR). For details on the RAM control register (RAMCR), refer to section 26.3.7, RAM Control Register (RAMCR).

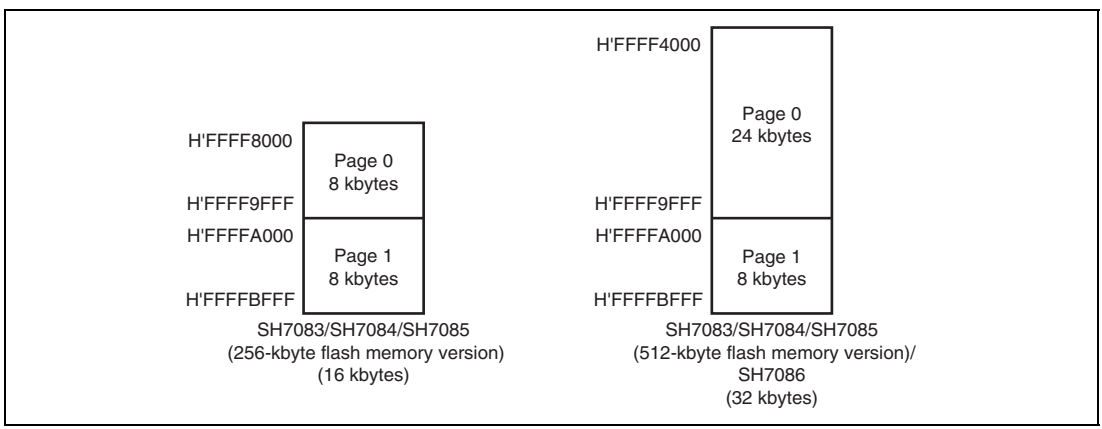


Figure 25.1 On-chip RAM Addresses

25.1 Usage Notes

25.1.1 Module Standby Mode Setting

RAM can be enabled/disabled by the standby control register. The initial value enables RAM operation. RAM access is disabled by setting the module standby mode. For details, see section 26, Power-Down Modes.

25.1.2 Address Error

When an address error in write access to the on-chip RAM occurs, the contents of the on-chip RAM may be corrupted.

25.1.3 Initial Values in RAM

After power has been supplied, initial values in RAM remain undefined until RAM is written.

Section 26 Power-Down Modes

This LSI supports the following power-down modes: sleep mode, software standby mode, deep software standby mode, and module standby mode.

26.1 Features

- Supports sleep mode, software standby mode, module standby mode, and deep software standby mode.

26.1.1 Types of Power-Down Modes

This LSI has the following power-down modes.

- Sleep mode
- Software standby mode
- Deep software standby mode
- Module standby mode

Table 26.1 shows the methods to make a transition from the program execution state, as well as the CPU and peripheral module states in each mode and the procedures for canceling each mode.

Table 26.1 States of Power-Down Modes

		State					
Mode	Transition Method	CPG	CPU	CPU Register	On-Chip Memory	On-Chip Peripheral Modules	Canceling Procedure
Sleep	Execute SLEEP instruction with STBY bit in STBCR1 cleared to 0.	Runs	Halts	Held	Runs	Run	• Reset
Software standby	Execute SLEEP instruction with STBY bit in STBCR1 and STBYMD bit in STBCR6 set to 1.	Halts	Halts	Held	Halts (contents retained)	Halt	• Interrupt by NMI or IRQ • Power-on reset by the $\overline{\text{RES}}$ pin • Manual reset by the $\overline{\text{MRES}}$ pin
Deep software standby	Execute SLEEP instruction with STBY bit in STBCR1 set to 1 and STBYMD bit in STBCR6 cleared to 0.	Halts	Halts	Undefined	Halts (contents undefined)	Halt	• Power-on reset by the $\overline{\text{RES}}$ pin
Module standby	Set MSTP bits in STBCR2 to STBCR5 to 1.	Runs	Runs	Held	Specified module halts (contents retained)	Specified module halts	• Clear MSTP bit to 0 • Power-on reset (for modules whose MSTP bit has an initial value of 0)

Note: For details on the states of on-chip peripheral module registers in each mode, refer to section 27.3, Register States in Each Operating Mode. For details on the pin states in each mode, refer to appendix A, Pin States.

26.2 Input/Output Pins

Table 26.2 lists the pins used for the power-down modes.

Table 26.2 Pin Configuration

Pin Name	Symbol	I/O	Description
Power-on reset	$\overline{\text{RES}}$	Input	Power-on reset input signal. Power-on reset by low level.
Manual reset	$\overline{\text{MRES}}$	Input	Manual reset input signal. Manual reset by low level.

26.3 Register Descriptions

There are following registers used for the power-down modes. For details on the addresses of these registers and the states of these registers in each processing state, see section 27, List of Registers.

Table 26.3 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Standby control register 1	STBCR1	R/W	H'00	H'FFFFE802	8
Standby control register 2	STBCR2	R/W	H'38	H'FFFFE804	8
Standby control register 3	STBCR3	R/W	H'FF	H'FFFFE806	8
Standby control register 4	STBCR4	R/W	H'FF	H'FFFFE808	8
Standby control register 5	STBCR5	R/W	H'03	H'FFFFE80A	8
Standby control register 6	STBCR6	R/W	H'00	H'FFFFE80C	8
RAM control register	RAMCR	R/W	H'10	H'FFFFE880	8

26.3.1 Standby Control Register 1 (STBCR1)

STBCR1 is an 8-bit readable/writable register that specifies the state of the power-down mode.

Bit:	7	6	5	4	3	2	1	0
	STBY	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	STBY	0	R/W	Standby Specifies transition to software standby mode. 0: Executing SLEEP instruction makes this LSI sleep mode 1: Executing SLEEP instruction makes this LSI software standby mode or deep software standby mode

Bit	Bit Name	Initial Value	R/W	Description
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

26.3.2 Standby Control Register 2 (STBCR2)

STBCR2 is an 8-bit readable/writable register that controls the operation of modules in power-down mode.

Bit:	7	6	5	4	3	2	1	0
	MSTP7	MSTP6	-	MSTP4	MSTP3	-	-	-
Initial value:	0	0	1	1	1	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP7	0	R/W	Module Stop Bit 7 When this bit is set to 1, the supply of the clock to the RAM is halted. 0: RAM operates 1: Clock supply to RAM halted
6	MSTP6	0	R/W	Module Stop Bit 6 When this bit is set to 1, the supply of the clock to the ROM is halted. 0: ROM operates 1: Clock supply to ROM halted
5	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
4	MSTP4	1	R/W	Module Stop Bit 4 When this bit is set to 1, the supply of the clock to the DTC is halted. 0: DTC operates 1: Clock supply to the DTC halted

Bit	Bit Name	Initial Value	R/W	Description
3	MSTP3	1	R/W	Module Stop Bit 3 When this bit is set to 1, the supply of the clock to the DMAC is halted. 0: DMAC operates 1: Clock supply to the DMAC halted
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

26.3.3 Standby Control Register 3 (STBCR3)

STBCR3 is an 8-bit readable/writable register that controls the operation of modules in power-down mode.

Bit:	7	6	5	4	3	2	1	0
	MSTP15	MSTP14	MSTP13	MSTP12	MSTP11	MSTP10	-	-
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP15	1	R/W	Module Stop Bit 15 When this bit is set to 1, the supply of the clock to the IIC2 is halted. 0: IIC2 operates 1: Clock supply to IIC2 halted
6	MSTP14	1	R/W	Module Stop Bit 14 When this bit is set to 1, the supply of the clock to the SCIF is halted. 0: SCIF operates 1: Clock supply to SCIF halted

Bit	Bit Name	Initial Value	R/W	Description
5	MSTP13	1	R/W	Module Stop Bit 13 When this bit is set to 1, the supply of the clock to the SCI_2 is halted. 0: SCI_2 operates 1: Clock supply to SCI_2 halted
4	MSTP12	1	R/W	Module Stop Bit 12 When this bit is set to 1, the supply of the clock to the SCI_1 is halted. 0: SCI_1 operates 1: Clock supply to SCI_1 halted
3	MSTP11	1	R/W	Module Stop Bit 11 When this bit is set to 1, the supply of the clock to the SCI_0 is halted. 0: SCI_0 operates 1: Clock supply to SCI_0 halted
2	MSTP10	1	R/W	Module Stop Bit 10 When this bit is set to 1, the supply of the clock to the SSU is halted. 0: SSU operates 1: Clock supply to SSU halted
1, 0	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.

26.3.4 Standby Control Register 4 (STBCR4)

STBCR4 is an 8-bit readable/writable register that controls the operation of modules in power-down mode.

Bit:	7	6	5	4	3	2	1	0
	MSTP23	MSTP22	MSTP21	-	-	MSTP18	MSTP17	MSTP16
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP23	1	R/W	Module Stop Bit 23 When this bit is set to 1, the supply of the clock to the MTU2S is halted. 0: MTU2S operates 1: Clock supply to MTU2S halted
6	MSTP22	1	R/W	Module Stop Bit 22 When this bit is set to 1, the supply of the clock to the MTU2 is halted. 0: MTU2 operates 1: Clock supply to MTU2 halted
5	MSTP21	1	R/W	Module Stop Bit 21 When this bit is set to 1, the supply of the clock to the CMT is halted. 0: CMT operates 1: Clock supply to CMT halted
4, 3	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
2	MSTP18	1	R/W	Module Stop Bit 18 When this bit is set to 1, the supply of the clock to the A/D_2 is halted. 0: A/D_2 operates 1: Clock supply to A/D_2 halted
1	MSTP17	1	R/W	Module Stop Bit 17 When this bit is set to 1, the supply of the clock to the A/D_1 is halted. 0: A/D_1 operates 1: Clock supply to A/D_1 halted
0	MSTP16	1	R/W	Module Stop Bit 16 When this bit is set to 1, the supply of the clock to the A/D_0 is halted. 0: A/D_0 operates 1: Clock supply to A/D_0 halted

26.3.5 Standby Control Register 5 (STBCR5)

STBCR5 is an 8-bit readable/writable register that controls the operation of modules in power-down mode.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	MSTP25	MSTP24
Initial value:	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	MSTP25	1	R/W	Module Stop Bit 25 When this bit is set to 1, the supply of the clock to the AUD is halted. 0: AUD operates 1: Clock supply to AUD halted
0	MSTP24	1	R/W	Module Stop Bit 24 When this bit is set to 1, the supply of the clock to the UBC is halted. 0: UBC operates 1: Clock supply to UBC halted

26.3.6 Standby Control Register 6 (STBCR6)

STBCR6 is an 8-bit readable/writable register that specifies the state of the power-down modes.

Bit:	7	6	5	4	3	2	1	0
	AUD SRST	HIZ	-	-	-	-	STBY MD	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7	AUDSRST	0	R/W	AUD Software Reset This bit controls the AUD reset by software. When 0 is written to AUDSRST, the AUD module shifts to the power-on reset state. 0: Shifts to the AUD reset state 1: Clears the AUD reset When setting this bit to 1, the MSTP25 bit in STBCR5 should be 0.
6	HIZ	0	R/W	Port High-Impedance In software standby mode, this bit selects whether the pin state is retained or changed to high-impedance. 0: In software standby mode, the pin state is retained 1: In software standby mode, the pin state is changed to high-impedance
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	STBYMD	0	R/W	Software Standby Mode Select This bit selects a transition to software standby mode or deep software standby mode by executing the SLEEP instruction when the STBY bit is 1 in STBCR1. 0: Transition to deep software standby mode 1: Transition to software standby mode
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

26.3.7 RAM Control Register (RAMCR)

RAMCR is an 8-bit readable/writable register that enables/disables the access to the on-chip RAM.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	RAME	-	-	-	-
Initial value:	0	0	0	1	0	0	0	-
R/W:	R	R	R	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	RAME	1	R/W	RAM Enable This bit enables/disables the on-chip RAM. 0: On-chip RAM disabled 1: On-chip RAM enabled When this bit is cleared to 0, the access to the on-chip RAM is disabled. In this case, an undefined value is returned when reading or fetching the data or instruction from the on-chip RAM, and writing to the on-chip RAM is ignored. When RAME is cleared to 0 to disable the on-chip RAM, an instruction to access the on-chip RAM should not be set next to the instruction to write RAMCR. If such an instruction is set, normal access is not guaranteed. When RAME is set to 1 to enable the on-chip RAM, an instruction to read RAMCR should be set next to the instruction to write to RAMCR. If an instruction to access the on-chip RAM is set next to the instruction to write to RAMCR, normal access is not guaranteed.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

26.4 Sleep Mode

26.4.1 Transition to Sleep Mode

Executing the SLEEP instruction when the STBY bit in STBCR1 is 0 causes a transition from the program execution state to sleep mode. However, sleep mode cannot be entered when the bus is released (low-level input to $\overline{\text{BREQ}}$ pin). Although the CPU halts immediately after executing the SLEEP instruction, the contents of its internal registers remain unchanged. The on-chip peripheral modules continue to operate.

26.4.2 Canceling Sleep Mode

Sleep mode is canceled by a reset.

Do not cancel sleep mode with an interrupt.

(1) Canceling with Reset

Sleep mode is canceled by a power-on reset with the $\overline{\text{RES}}$ pin, a manual reset with the $\overline{\text{MRES}}$ pin, or an internal power-on/manual reset by WDT.

26.5 Software Standby Mode

26.5.1 Transition to Software Standby Mode

This LSI switches from a program execution state to software standby mode by executing the SLEEP instruction when the STBY bit in STBCR1 and the STBYMD bit in STBCR6 are set to 1. However, software standby mode cannot be entered when the bus is released (low-level input to $\overline{\text{BREQ}}$ pin). Execute the SLEEP instruction after halting the DMAC and DTC. In software standby mode, not only the CPU but also the clock and on-chip peripheral modules halt.

The contents of the CPU registers and the data of the on-chip RAM remain unchanged. Some registers of on-chip peripheral modules are, however, initialized. For details on the states of on-chip peripheral module registers in software standby mode, refer to section 27.3, Register States in Each Operating Mode. For details on the pin states in software standby mode, refer to appendix A, Pin States.

The procedure for switching to software standby mode is as follows:

1. Clear the TME bit in the timer control register (WTCSR) of the WDT to 0 to stop the WDT.
2. Set the timer counter (WTCNT) of the WDT to 0 and bits CKS2 to CKS0 in WTCSR to appropriate values to secure the specified oscillation settling time.
3. If the DMAC and DTC are operating, stop their operation.
4. If the bus is released (low-level input to $\overline{\text{BREQ}}$ pin), acquire the bus mastership (high-level input to $\overline{\text{BREQ}}$ pin).
5. After setting the STBY bit in STBCR1 and the STBYMD bit in STBCR6 to 1, execute the SLEEP instruction.
6. Software standby mode is entered and the clocks within this LSI are halted.

26.5.2 Canceling Software Standby Mode

Software standby mode is canceled by interrupts (NMI, IRQ) or a reset.

(1) Canceling with Interrupt

The WDT can be used for hot starts. When an NMI or IRQ interrupt (edge detection) is detected, the clock will be supplied to the entire LSI and software standby mode will be canceled after the time set in the timer control/status register of the WDT has elapsed. Interrupt exception handling is then executed.

When the priority level of an IRQ interrupt is lower than the interrupt mask level set in the status register (SR) of the CPU, an interrupt request is not accepted preventing software standby mode from being canceled.

When falling-edge detection is selected for the NMI pin, drive the NMI pin high before making a transition to software standby mode. When rising-edge detection is selected for the NMI pin, drive the NMI pin low before making a transition to software standby mode.

Similarly, when falling-edge detection is selected for the IRQ pin, drive the IRQ pin high before making a transition to software standby mode. When rising-edge detection is selected for the IRQ pin, drive the IRQ pin low before making a transition to software standby mode.

(2) Canceling with Power-on Reset

Software standby mode is canceled by a power-on reset with the $\overline{\text{RES}}$ pin. Keep the $\overline{\text{RES}}$ pin low until the clock oscillation settles.

(3) Canceling with Manual Reset

Software standby mode is canceled by a manual reset with the $\overline{\text{MRES}}$ pin. Keep the $\overline{\text{MRES}}$ pin low until the clock oscillation settles.

26.6 Deep Software Standby Mode

26.6.1 Transition to Deep Software Standby Mode

This LSI shifts from a program execution state to deep software standby mode by executing the SLEEP instruction when the STBY bit in STBCR1 is 1 and the STBYMD bit in STBCR6 is 0. However, deep software standby mode cannot be entered when the bus is released (low-level input to $\overline{\text{BREQ}}$ pin). Execute the SLEEP instruction after halting the DMAC and DTC. In deep software standby mode, not only the CPU but also the clock and on-chip peripheral modules halt. Furthermore, the internal power supply of this LSI is turned off.

The contents of the CPU registers and the data of the on-chip RAM become undefined. The registers of on-chip peripheral modules are initialized. For details on the pin states in deep software standby mode, refer to appendix A, Pin States.

The procedure for a transition to deep software standby mode is as follows:

1. Clear the TME bit in the timer control register (WTCSR) of the WDT to 0 to stop the WDT.
2. If the DMAC and DTC are operating, stop their operation.
3. If the bus is released (low-level input to $\overline{\text{BREQ}}$ pin), acquire the bus mastership (high-level input to $\overline{\text{BREQ}}$ pin).
4. After setting the STBY bit in STBCR1 to 1 and clearing the STBYMD bit in STBCR6 to 0, execute the SLEEP instruction.
5. Deep software standby mode is entered, the clocks within this LSI are halted, and the internal power supply of this LSI is turned off.

26.6.2 Canceling Deep Software Standby Mode

Deep software standby mode is canceled by a power-on reset with the $\overline{\text{RES}}$ pin. Keep the $\overline{\text{RES}}$ pin low until the clock oscillation settles.

26.7 Module Standby Mode

26.7.1 Transition to Module Standby Mode

Setting the MSTP bits in the standby control registers (STBCR2 to STBCR5) to 1 halts the supply of clocks to the corresponding on-chip peripheral modules. This function can be used to reduce the power consumption in normal mode.

Do not access registers of an on-chip peripheral module which has been set to enter module standby mode. For details on the states of on-chip peripheral module registers in module standby mode, refer to section 27.3, Register States in Each Operating Mode.

26.7.2 Canceling Module Standby Function

The module standby function can be canceled by clearing the MSTP bits in STBCR2 to STBCR5 to 0. The module standby function can be canceled by a power-on reset for modules whose MSTP bit has an initial value of 0.

26.8 Usage Note

26.8.1 Current Consumption while Waiting for Oscillation to Be Stabilized

The current consumption while waiting for oscillation to be stabilized is higher than that while oscillation is stabilized.

26.8.2 Executing the SLEEP Instruction

Apply either of the following measures before executing the SLEEP instruction to initiate the transition to sleep mode or software standby mode.

Measure A: Stop the operation of the DMAC and DTC and the generation of interrupts from on-chip peripheral modules, IRQ interrupts, and the NMI interrupt before executing the SLEEP instruction.

Measure B: Change the value in FRQCR to the initial value, H'36DB, and then dummy-read FRQCR twice before executing the SLEEP instruction.

Section 27 List of Registers

This section gives information on internal I/O registers. The contents of this section are as follows:

1. Register Address Table (in the order from a lower address)
   - Registers are listed in the order from lower allocated addresses.
   - As for reserved addresses, the register name column is indicated with —. Do not access reserved addresses.
   - As for 16- or 32-bit address, the MSB addresses are shown.
   - The list is classified according to module names.
   - The numbers of access cycles are given.
2. Register Bit Table
   - Bit configurations are shown in the order of the register address table.
   - As for reserved bits, the bit name column is indicated with —.
   - As for the blank column of the bit names, the whole register is allocated to the counter or data.
   - As for 16- or 32-bit registers, bits are indicated from the MSB.
3. Register State in Each Operating Mode
   - Register states are listed in the order of the register address table.
   - Register states in the basic operating mode are shown. As for modules including their specific states such as reset, see the sections of those modules.

27.1 Register Address Table (In the Order from Lower Addresses)

Access sizes are indicated with the number of bits. Access states are indicated with the number of specified reference clock states. These values are those at 8-bit access (B), 16-bit access (W), or 32-bit access (L).

Note: Access to undefined or reserved addresses is prohibited. Correct operation cannot be guaranteed if these addresses are accessed.

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Serial mode register_0	SCSMR_0	8	H'FFFFC000	SCI	8	Pϕ reference	16 bits
Bit rate register_0	SCBRR_0	8	H'FFFFC002	(Channel 0)	8	B:2	
Serial control register_0	SCSCR_0	8	H'FFFFC004		8		
Transmit data register_0	SCTDR_0	8	H'FFFFC006		8		
Serial status register_0	SCSSR_0	8	H'FFFFC008		8		
Receive data register_0	SCRDR_0	8	H'FFFFC00A		8		
Serial direction control register_0	SCSDCR_0	8	H'FFFFC00C		8		
Serial port register_0	SCSPTR_0	8	H'FFFFC00E		8		
Serial mode register_1	SCSMR_1	8	H'FFFFC080	SCI	8		
Bit rate register_1	SCBRR_1	8	H'FFFFC082	(Channel 1)	8		
Serial control register_1	SCSCR_1	8	H'FFFFC084		8		
Transmit data register_1	SCTDR_1	8	H'FFFFC086		8		
Serial status register_1	SCSSR_1	8	H'FFFFC088		8		
Receive data register_1	SCRDR_1	8	H'FFFFC08A		8		
Serial direction control register_1	SCSDCR_1	8	H'FFFFC08C		8		
Serial port register_1	SCSPTR_1	8	H'FFFFC08E		8		
Serial mode register_2	SCSMR_2	8	H'FFFFC100	SCI	8		
Bit rate register_2	SCBRR_2	8	H'FFFFC102	(Channel 2)	8		
Serial control register_2	SCSCR_2	8	H'FFFFC104		8		
Transmit data register_2	SCTDR_2	8	H'FFFFC106		8		
Serial status register_2	SCSSR_2	8	H'FFFFC108		8		
Receive data register_2	SCRDR_2	8	H'FFFFC10A		8		
Serial direction control register_2	SCSDCR_2	8	H'FFFFC10C		8		
Serial port register_2	SCSPTR_2	8	H'FFFFC10E		8		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Serial mode register_3	SCSMR_3	16	H'FFFFC180	SCIF	16	Pϕ reference	16 bits
Bit rate register_3	SCBRR_3	8	H'FFFFC182	(Channel 3)	8	B:2	
Serial control register_3	SCSCR_3	16	H'FFFFC184		16	W:2	
Transmit FIFO data register_3	SCFTDR_3	8	H'FFFFC186		8		
Serial status register_3	SCFSR_3	16	H'FFFFC188		16		
Receive FIFO data register_3	SCFRDR_3	8	H'FFFFC18A		8		
FIFO control register_3	SCFCR_3	16	H'FFFFC18C		16		
FIFO data count register_3	SCFDR_3	16	H'FFFFC18E		16		
Serial port register_3	SCSPTR_3	16	H'FFFFC190		16		
Line status register_3	SCLSR_3	16	H'FFFFC192		16		
Timer control register_3	TCR_3	8	H'FFFFC200	MTU2	8, 16, 32	Pϕ reference	16 bits
Timer control register_4	TCR_4	8	H'FFFFC201		8	B:2	
Timer mode register_3	TMDR_3	8	H'FFFFC202		8, 16	W:2	
Timer mode register_4	TMDR_4	8	H'FFFFC203		8	L:4	
Timer I/O control register H_3	TIORH_3	8	H'FFFFC204		8, 16, 32		
Timer I/O control register L_3	TIORL_3	8	H'FFFFC205		8		
Timer I/O control register H_4	TIORH_4	8	H'FFFFC206		8, 16		
Timer I/O control register L_4	TIORL_4	8	H'FFFFC207		8		
Timer interrupt enable register_3	TIER_3	8	H'FFFFC208		8, 16		
Timer interrupt enable register_4	TIER_4	8	H'FFFFC209		8		
Timer output master enable register	TOER	8	H'FFFFC20A		8		
Timer gate control register	TGCR	8	H'FFFFC20D		8		
Timer output control register 1	TOCR1	8	H'FFFFC20E		8, 16		
Timer output control register 2	TOCR2	8	H'FFFFC20F		8		
Timer counter_3	TCNT_3	16	H'FFFFC210		16, 32		
Timer counter_4	TCNT_4	16	H'FFFFC212		16		
Timer cycle data register	TCDR	16	H'FFFFC214		16, 32		
Timer dead time data register	TDDR	16	H'FFFFC216		16		
Timer general register A_3	TGRA_3	16	H'FFFFC218		16, 32		
Timer general register B_3	TGRB_3	16	H'FFFFC21A		16		
Timer general register A_4	TGRA_4	16	H'FFFFC21C		16, 32		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Timer general register B_4	TGRB_4	16	H'FFFFC21E	MTU2	16	Pϕ reference	16 bits
Timer sub-counter	TCNTS	16	H'FFFFC220		16, 32	B:2	
Timer cycle buffer register	TCBR	16	H'FFFFC222		16	W:2	
Timer general register C_3	TGRC_3	16	H'FFFFC224		16, 32	L:4	
Timer general register D_3	TGRD_3	16	H'FFFFC226		16		
Timer general register C_4	TGRC_4	16	H'FFFFC228		16, 32		
Timer general register D_4	TGRD_4	16	H'FFFFC22A		16		
Timer status register_3	TSR_3	8	H'FFFFC22C		8, 16		
Timer status register_4	TSR_4	8	H'FFFFC22D		8		
Timer interrupt skipping set register	TITCR	8	H'FFFFC230		8, 16		
Timer interrupt skipping counter	TITCNT	8	H'FFFFC231		8		
Timer buffer transfer set register	TBTER	8	H'FFFFC232		8		
Timer dead time enable register	TDER	8	H'FFFFC234		8		
Timer output level buffer register	TOLBR	8	H'FFFFC236		8		
Timer buffer operation transfer mode register_3	TBTM_3	8	H'FFFFC238		8, 16		
Timer buffer operation transfer mode register_4	TBTM_4	8	H'FFFFC239		8		
Timer A/D converter start request control register	TADCR	16	H'FFFFC240		16		
Timer A/D converter start request cycle set register A_4	TADCORA_4	16	H'FFFFC244		16, 32		
Timer A/D converter start request cycle set register B_4	TADCORB_4	16	H'FFFFC246		16		
Timer A/D converter start request cycle set buffer register A_4	TADCOBRA_4	16	H'FFFFC248		16, 32		
Timer A/D converter start request cycle set buffer register B_4	TADCOBRB_4	16	H'FFFFC24A		16		
Timer waveform control register	TWCR	8	H'FFFFC260		8		
Timer start register	TSTR	8	H'FFFFC280		8, 16		
Timer synchronous register	TSYR	8	H'FFFFC281		8		
Timer counter synchronous start register	TCSYSTR	8	H'FFFFC282		8		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Timer read/write enable register	TRWER	8	H'FFFFC284	MTU2	8	Pϕ reference	16 bits
Timer control register_0	TCR_0	8	H'FFFFC300		8, 16, 32	B:2	
Timer mode register_0	TMDR_0	8	H'FFFFC301		8	W:2	
Timer I/O control register H_0	TIORH_0	8	H'FFFFC302		8, 16	L:4	
Timer I/O control register L_0	TIORL_0	8	H'FFFFC303		8		
Timer interrupt enable register_0	TIER_0	8	H'FFFFC304		8, 16, 32		
Timer status register_0	TSR_0	8	H'FFFFC305		8		
Timer counter_0	TCNT_0	16	H'FFFFC306		16		
Timer general register A_0	TGRA_0	16	H'FFFFC308		16, 32		
Timer general register B_0	TGRB_0	16	H'FFFFC30A		16		
Timer general register C_0	TGRC_0	16	H'FFFFC30C		16, 32		
Timer general register D_0	TGRD_0	16	H'FFFFC30E		16		
Timer general register E_0	TGRE_0	16	H'FFFFC320		16, 32		
Timer general register F_0	TGRF_0	16	H'FFFFC322		16		
Timer interrupt enable register 2_0	TIER2_0	8	H'FFFFC324		8, 16		
Timer status register 2_0	TSR2_0	8	H'FFFFC325		8		
Timer buffer operation transfer mode register_0	TBTM_0	8	H'FFFFC326		8		
Timer control register_1	TCR_1	8	H'FFFFC380		8, 16		
Timer mode register_1	TMDR_1	8	H'FFFFC381		8		
Timer I/O control register_1	TIOR_1	8	H'FFFFC382		8		
Timer interrupt enable register_1	TIER_1	8	H'FFFFC384		8, 16, 32		
Timer status register_1	TSR_1	8	H'FFFFC385		8		
Timer counter_1	TCNT_1	16	H'FFFFC386		16		
Timer general register A_1	TGRA_1	16	H'FFFFC388		16, 32		
Timer general register B_1	TGRB_1	16	H'FFFFC38A		16		
Timer input capture control register	TICCR	8	H'FFFFC390		8		
Timer control register_2	TCR_2	8	H'FFFFC400		8, 16		
Timer mode register_2	TMDR_2	8	H'FFFFC401		8		
Timer I/O control register_2	TIOR_2	8	H'FFFFC402		8		
Timer interrupt enable register_2	TIER_2	8	H'FFFFC404		8, 16, 32		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Timer status register_2	TSR_2	8	H'FFFFC405	MTU2	8	Pϕ reference	16 bits
Timer counter_2	TCNT_2	16	H'FFFFC406		16	B:2	
Timer general register A_2	TGRA_2	16	H'FFFFC408		16, 32	W:2	
Timer general register B_2	TGRB_2	16	H'FFFFC40A		16	L:4	
Timer counter U_5	TCNTU_5	16	H'FFFFC480		16, 32		
Timer general register U_5	TGRU_5	16	H'FFFFC482		16		
Timer control register U_5	TCRU_5	8	H'FFFFC484		8		
Timer I/O control register U_5	TIORU_5	8	H'FFFFC486		8		
Timer counter V_5	TCNTV_5	16	H'FFFFC490		16, 32		
Timer general register V_5	TGRV_5	16	H'FFFFC492		16		
Timer control register V_5	TCRV_5	8	H'FFFFC494		8		
Timer I/O control register V_5	TIORV_5	8	H'FFFFC496		8		
Timer counter W_5	TCNTW_5	16	H'FFFFC4A0		16, 32		
Timer general register W_5	TGRW_5	16	H'FFFFC4A2		16		
Timer control register W_5	TCRW_5	8	H'FFFFC4A4		8		
Timer I/O control register W_5	TIORW_5	8	H'FFFFC4A6		8		
Timer status register_5	TSR_5	8	H'FFFFC4B0		8		
Timer interrupt enable register_5	TIER_5	8	H'FFFFC4B2		8		
Timer start register_5	TSTR_5	8	H'FFFFC4B4		8		
Timer compare match clear register	TCNTCMPCLR	8	H'FFFFC4B6		8		
Timer control register_3S	TCR_3S	8	H'FFFFC600	MTU2S	8, 16, 32	Pϕ reference	16 bits
Timer control register_4S	TCR_4S	8	H'FFFFC601		8	B:2	
Timer mode register_3S	TMDR_3S	8	H'FFFFC602		8, 16	W:2	
Timer mode register_4S	TMDR_4S	8	H'FFFFC603		8	L:4	
Timer I/O control register H_3S	TIORH_3S	8	H'FFFFC604		8, 16, 32		
Timer I/O control register L_3S	TIORL_3S	8	H'FFFFC605		8		
Timer I/O control register H_4S	TIORH_4S	8	H'FFFFC606		8, 16		
Timer I/O control register L_4S	TIORL_4S	8	H'FFFFC607		8		
Timer interrupt enable register_3S	TIER_3S	8	H'FFFFC608		8, 16		
Timer interrupt enable register_4S	TIER_4S	8	H'FFFFC609		8		
Timer output master enable register S	TOERS	8	H'FFFFC60A		8		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Timer gate control register S	TGCRS	8	H'FFFFC60D	MTU2S	8	Pϕ reference	16 bits
Timer output control register 1S	TOCR1S	8	H'FFFFC60E		8, 16	B:2	
Timer output control register 2S	TOCR2S	8	H'FFFFC60F		8	W:2	
Timer counter_3S	TCNT_3S	16	H'FFFFC610		16, 32	L:4	
Timer counter_4S	TCNT_4S	16	H'FFFFC612		16		
Timer cycle data register S	TCDRS	16	H'FFFFC614		16, 32		
Timer dead time data register S	TDDRS	16	H'FFFFC616		16		
Timer general register A_3S	TGRA_3S	16	H'FFFFC618		16, 32		
Timer general register B_3S	TGRB_3S	16	H'FFFFC61A		16		
Timer general register A_4S	TGRA_4S	16	H'FFFFC61C		16, 32		
Timer general register B_4S	TGRB_4S	16	H'FFFFC61E		16		
Timer sub-counter S	TCNTSS	16	H'FFFFC620		16, 32		
Timer cycle buffer register S	TCBRS	16	H'FFFFC622		16		
Timer general register C_3S	TGRC_3S	16	H'FFFFC624		16, 32		
Timer general register D_3S	TGRD_3S	16	H'FFFFC626		16		
Timer general register C_4S	TGRC_4S	16	H'FFFFC628		16, 32		
Timer general register D_4S	TGRD_4S	16	H'FFFFC62A		16		
Timer status register_3S	TSR_3S	8	H'FFFFC62C		8, 16		
Timer status register_4S	TSR_4S	8	H'FFFFC62D		8		
Timer interrupt skipping set register S	TITCRS	8	H'FFFFC630		8, 16		
Timer interrupt skipping counter S	TITCNTS	8	H'FFFFC631		8		
Timer buffer transfer set register S	TBTERS	8	H'FFFFC632		8		
Timer dead time enable register S	TDERS	8	H'FFFFC634		8		
Timer output level buffer register S	TOLBRS	8	H'FFFFC636		8		
Timer buffer operation transfer mode register_3S	TBTM_3S	8	H'FFFFC638		8, 16		
Timer buffer operation transfer mode register_4S	TBTM_4S	8	H'FFFFC639		8		
Timer A/D converter start request control register S	TADCRS	16	H'FFFFC640		16		
Timer A/D converter start request cycle set register A_4S	TADCORA_4S	16	H'FFFFC644		16, 32		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Timer A/D converter start request cycle set register B_4S	TADCORB_4S	16	H'FFFFC646	MTU2S	16	Pϕ reference	16 bits
Timer A/D converter start request cycle set buffer register A_4S	TADCOBRA_4S	16	H'FFFFC648		16, 32	B:2 W:2 L:4	
Timer A/D converter start request cycle set buffer register B_4S	TADCOBRB_4S	16	H'FFFFC64A		16		
Timer synchronous clear register S	TSYCRS	8	H'FFFFC650		8		
Timer waveform control register S	TWCRS	8	H'FFFFC660		8		
Timer start register S	TSTRS	8	H'FFFFC680		8, 16		
Timer synchronous register S	TSYRS	8	H'FFFFC681		8		
Timer read/write enable register S	TRWERS	8	H'FFFFC684		8		
Timer counter U_5S	TCNTU_5S	16	H'FFFFC880		16, 32		
Timer general register U_5S	TGRU_5S	16	H'FFFFC882		16		
Timer control register U_5S	TCRU_5S	8	H'FFFFC884		8		
Timer I/O control register U_5S	TIORU_5S	8	H'FFFFC886		8		
Timer counter V_5S	TCNTV_5S	16	H'FFFFC890		16, 32		
Timer general register V_5S	TGRV_5S	16	H'FFFFC892		16		
Timer control register V_5S	TCRV_5S	8	H'FFFFC894		8		
Timer I/O control register V_5S	TIORV_5S	8	H'FFFFC896		8		
Timer counter W_5S	TCNTW_5S	16	H'FFFFC8A0		16, 32		
Timer general register W_5S	TGRW_5S	16	H'FFFFC8A2		16		
Timer control register W_5S	TCRW_5S	8	H'FFFFC8A4		8		
Timer I/O control register W_5S	TIORW_5S	8	H'FFFFC8A6		8		
Timer status register_5S	TSR_5S	8	H'FFFFC8B0		8		
Timer interrupt enable register_5S	TIER_5S	8	H'FFFFC8B2		8		
Timer start register_5S	TSTR_5S	8	H'FFFFC8B4		8		
Timer compare match clear register S	TCNTCMPCLRS	8	H'FFFFC8B6		8		
A/D data register 0	ADDR0	16	H'FFFFC900	A/D	16	Pϕ reference	16 bits
A/D data register 1	ADDR1	16	H'FFFFC902	(Channel 0)	16	B:2	
A/D data register 2	ADDR2	16	H'FFFFC904		16	W:2	
A/D data register 3	ADDR3	16	H'FFFFC906		16		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
A/D control/status register_0	ADCSR_0	16	H'FFFFC910	A/D	16	Pϕ reference	16 bits
A/D control register_0	ADCR_0	16	H'FFFFC912	(Channel 0)	16	B:2	
A/D data register 4	ADDR4	16	H'FFFFC980	A/D	16	W:2	
A/D data register 5	ADDR5	16	H'FFFFC982	(Channel 1)	16		
A/D data register 6	ADDR6	16	H'FFFFC984		16		
A/D data register 7	ADDR7	16	H'FFFFC986		16		
A/D control/status register_1	ADCSR_1	16	H'FFFFC990		16, 8, 32		
A/D control register_1	ADCR_1	16	H'FFFFC992		16		
A/D data register 8	ADDR8	16	H'FFFFCA00	A/D	16		
A/D data register 9	ADDR9	16	H'FFFFCA02	(Channel 2)	16		
A/D data register 10	ADDR10	16	H'FFFFCA04		16		
A/D data register 11	ADDR11	16	H'FFFFCA06		16		
A/D data register 12	ADDR12	16	H'FFFFCA08		16		
A/D data register 13	ADDR13	16	H'FFFFCA0A		16		
A/D data register 14	ADDR14	16	H'FFFFCA0C		16		
A/D data register 15	ADDR15	16	H'FFFFCA0E		16		
A/D control/status register_2	ADCSR_2	16	H'FFFFCA10		16		
A/D control register_2	ADCR_2	16	H'FFFFCA12		16		
Flash code control/status register	FCCS	8	H'FFFFCC00	FLASH	8	Pϕ reference	16 bits
Flash program code select register	FPCS	8	H'FFFFCC01		8	B:5	
Flash erase code select register	FECS	8	H'FFFFCC02		8		
Flash key code register	FKEY	8	H'FFFFCC04		8		
Flash MAT select register	FMATS	8	H'FFFFCC05		8		
Flash transfer destination address register	FTDAR	8	H'FFFFCC06		8		
DTC enable register A	DTCERA	16	H'FFFFCC80	DTC	8, 16	Pϕ reference	16 bits
DTC enable register B	DTCERB	16	H'FFFFCC82		8, 16	B:2	
DTC enable register C	DTCERC	16	H'FFFFCC84		8, 16	W:2	
DTC enable register D	DTCERD	16	H'FFFFCC86		8, 16	L:4	
DTC enable register E	DTCERE	16	H'FFFFCC88		8, 16		
DTC control register	DTCCR	8	H'FFFFCC90		8		
DTC vector base register	DTCVBR	32	H'FFFFCC94		8, 16, 32		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
I²C bus control register 1	ICCR1	8	H'FFFFCD80	IIC2	8	Pϕ reference	8 bits
I²C bus control register 2	ICCR2	8	H'FFFFCD81		8	B:2	
I²C bus mode register	ICMR	8	H'FFFFCD82		8		
I²C bus interrupt enable register	ICIER	8	H'FFFFCD83		8		
I²C bus status register	ICSR	8	H'FFFFCD84		8		
Slave address register	SAR	8	H'FFFFCD85		8		
I²C bus transmit data register	ICDRT	8	H'FFFFCD86		8		
I²C bus receive data register	ICDRR	8	H'FFFFCD87		8		
NF2CYC register	NF2CYC	8	H'FFFFCD88		8		
SS control register H	SSCRH	8	H'FFFFCD00	SSU	8, 16	Pϕ reference	16 bits
SS control register L	SSCRL	8	H'FFFFCD01		8	B:2	
SS mode register	SSMR	8	H'FFFFCD02		8, 16	W:2	
SS enable register	SSER	8	H'FFFFCD03		8		
SS status register	SSSR	8	H'FFFFCD04		8, 16		
SS control register 2	SSCR2	8	H'FFFFCD05		8		
SS transmit data register 0	SSTDR0	8	H'FFFFCD06		8, 16		
SS transmit data register 1	SSTDR1	8	H'FFFFCD07		8		
SS transmit data register 2	SSTDR2	8	H'FFFFCD08		8, 16		
SS transmit data register 3	SSTDR3	8	H'FFFFCD09		8		
SS receive data register 0	SSRDR0	8	H'FFFFCD0A		8, 16		
SS receive data register 1	SSRDR1	8	H'FFFFCD0B		8		
SS receive data register 2	SSRDR2	8	H'FFFFCD0C		8, 16		
SS receive data register 3	SSRDR3	8	H'FFFFCD0D		8		
Compare match timer start register	CMSTR	16	H'FFFFCE00	CMT	8, 16, 32	Pϕ reference	16 bits
Compare match timer control/status register_0	CMCSR_0	16	H'FFFFCE02		8, 16	B:2 W:2	
Compare match counter_0	CMCNT_0	16	H'FFFFCE04		8, 16, 32	L:4	
Compare match constant register_0	CMCOR_0	16	H'FFFFCE06		8, 16		
Compare match timer control/status register_1	CMCSR_1	16	H'FFFFCE08		8, 16, 32		
Compare match counter_1	CMCNT_1	16	H'FFFFCE0A		8, 16		
Compare match constant register_1	CMCOR_1	16	H'FFFFCE0C		8, 16, 32		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Input level control/status register 1	ICSR1	16	H'FFFFD000	POE	8, 16, 32	Pϕ reference	16 bits
Output level control/status register 1	OCSR1	16	H'FFFFD002		8, 16	B:2	
Input level control/status register 2	ICSR2	16	H'FFFFD004		8, 16, 32	W:2	
Output level control/status register 2	OCSR2	16	H'FFFFD006		8, 16	L:4	
Input level control/status register 3	ICSR3	16	H'FFFFD008		8, 16		
Software port output enable register	SPOER	8	H'FFFFD00A		8		
Port output enable control register 1	POECR1	8	H'FFFFD00B		8		
Port output enable control register 2	POECR2	16	H'FFFFD00C		8, 16		
Port A data register H	PADRH	16	H'FFFFD100	I/O	8, 16, 32	Pϕ reference	16 bits
Port A data register L	PADRL	16	H'FFFFD102		8, 16	B:2	
Port A I/O register H	PAIORH	16	H'FFFFD104	PFC	8, 16, 32	W:2	
Port A I/O register L	PAIORL	16	H'FFFFD106		8, 16	L:4	
Port A control register H4	PACRH4	16	H'FFFFD108		8, 16, 32		
Port A control register H3	PACRH3	16	H'FFFFD10A		8, 16		
Port A control register H2	PACRH2	16	H'FFFFD10C		8, 16, 32		
Port A control register H1	PACRH1	16	H'FFFFD10E		8, 16		
Port A control register L4	PACRL4	16	H'FFFFD110		8, 16, 32		
Port A control register L3	PACRL3	16	H'FFFFD112		8, 16		
Port A control register L2	PACRL2	16	H'FFFFD114		8, 16, 32		
Port A control register L1	PACRL1	16	H'FFFFD116		8, 16		
Port A port register H	PAPRH	16	H'FFFFD11C	I/O	8, 16, 32		
Port A port register L	PAPRL	16	H'FFFFD11E		8, 16		
Port B data register L	PBDRL	16	H'FFFFD182		8, 16		
Port B I/O register L	PBIORL	16	H'FFFFD186	PFC	8, 16		
Port B control register L3	PBCRL3	16	H'FFFFD192		8, 16		
Port B control register L2	PBCRL2	16	H'FFFFD194		8, 16, 32		
Port B control register L1	PBCRL1	16	H'FFFFD196		8, 16		
Port B port register L	PBPRL	16	H'FFFFD19E	I/O	8, 16		
Port C data register H	PCDRH	16	H'FFFFD200		8, 16, 32		
Port C data register L	PCDRL	16	H'FFFFD202		8, 16		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Port C I/O register H	PCIORH	16	H'FFFFD204	PFC	8, 16, 32	Pϕ reference	16 bits
Port C I/O register L	PCIORL	16	H'FFFFD206		8, 16	B:2	
Port C control register H3	PCCRH3	16	H'FFFFD20A		8, 16	W:2	
Port C control register H2	PCCRH2	16	H'FFFFD20C		8, 16, 32	L:4	
Port C control register H1	PCCRH1	16	H'FFFFD20E		8, 16		
Port C control register L4	PCCRL4	16	H'FFFFD210		8, 16, 32		
Port C control register L3	PCCRL3	16	H'FFFFD212		8, 16		
Port C control register L2	PCCRL2	16	H'FFFFD214		8, 16, 32		
Port C control register L1	PCCRL1	16	H'FFFFD216		8, 16		
Port C port register H	PCPRH	16	H'FFFFD21C	I/O	8, 16, 32		
Port C port register L	PCPRL	16	H'FFFFD21E		8, 16		
Port D data register H	PDDRH	16	H'FFFFD280		8, 16, 32		
Port D data register L	PDDRL	16	H'FFFFD282		8, 16		
Port D I/O register H	PDIORH	16	H'FFFFD284	PFC	8, 16, 32		
Port D I/O register L	PDIORL	16	H'FFFFD286		8, 16		
Port D control register H4	PDCRH4	16	H'FFFFD288		8, 16, 32		
Port D control register H3	PDCRH3	16	H'FFFFD28A		8, 16		
Port D control register H2	PDCRH2	16	H'FFFFD28C		8, 16, 32		
Port D control register H1	PDCRH1	16	H'FFFFD28E		8, 16		
Port D control register L4	PDCRL4	16	H'FFFFD290		8, 16, 32		
Port D control register L3	PDCRL3	16	H'FFFFD292		8, 16		
Port D control register L2	PDCRL2	16	H'FFFFD294		8, 16, 32		
Port D control register L1	PDCRL1	16	H'FFFFD296		8, 16		
Port D port register H	PDPRH	16	H'FFFFD29C	I/O	8, 16, 32		
Port D port register L	PDPRL	16	H'FFFFD29E		8, 16		
Port E data register H	PEDRH	16	H'FFFFD300		8, 16, 32		
Port E data register L	PEDRL	16	H'FFFFD302		8, 16		
Port E I/O register H	PEIORH	16	H'FFFFD304	PFC	8, 16, 32		
Port E I/O register L	PEIORL	16	H'FFFFD306		8, 16		
Port E control register H2	PECRH2	16	H'FFFFD30C		8, 16, 32		
Port E control register H1	PECRH1	16	H'FFFFD30E		8, 16		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Port E control register L4	PECRL4	16	H'FFFFD310	PFC	8, 16, 32	Pϕ reference	16 bits
Port E control register L3	PECRL3	16	H'FFFFD312		8, 16	B:2	
Port E control register L2	PECRL2	16	H'FFFFD314		8, 16, 32	W:2	
Port E control register L1	PECRL1	16	H'FFFFD316		8, 16	L:4	
Port E port register H	PEPRH	16	H'FFFFD31C	I/O	8, 16, 32		
Port E port register L	PEPRL	16	H'FFFFD31E		8, 16		
High-current port control register	HCPCR	16	H'FFFFD320	PFC	8, 16, 32		
IRQOUT function control register	IFCR	16	H'FFFFD322		8, 16		
Port F data register L	PFDRL	16	H'FFFFD382	I/O	8, 16		
Frequency control register	FRQCR	16	H'FFFFE800	CPG	16	Pϕ reference W:2	16 bits
Standby control register 1	STBCR1	8	H'FFFFE802	Power-down modes	8	Pϕ reference	16 bits
Standby control register 2	STBCR2	8	H'FFFFE804		8	B:2	
Standby control register 3	STBCR3	8	H'FFFFE806		8		
Standby control register 4	STBCR4	8	H'FFFFE808		8		
Standby control register 5	STBCR5	8	H'FFFFE80A		8		
Standby control register 6	STBCR6	8	H'FFFFE80C		8		
Watchdog timer counter	WTCNT	8	H'FFFFE810	WDT	8*1, 16*2	Pϕ reference	16 bits
Watchdog timer control/status register	WTCSR	8	H'FFFFE812	*1: Read *2: Write	8*1, 16*2	B:2*1 W:2*2	
Oscillation stop detection control register	OSCCR	8	H'FFFFE814	CPG	8	Pϕ reference B:2	16 bits
RAM control register	RAMCR	8	H'FFFFE880	Power-down modes	8	Pϕ reference B:2	16 bits
A/D trigger select register 0	ADTSR_0	16	H'FFFFE890	A/D	8, 16	Pϕ reference	16 bits
A/D trigger select register 1	ADTSR_1	16	H'FFFFE892		8, 16	B:2 W:2	
Bus function extending register	BSCEHR	16	H'FFFFE89A	BSC	8, 16	Pϕ reference B:2 W:2	16 bits

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Interrupt control register 0	ICR0	16	H'FFFFE900	INTC	8, 16	Pϕ reference	16 bits
IRQ control register	IRQCR	16	H'FFFFE902		8, 16	B:2	
IRQ status register	IRQSR	16	H'FFFFE904		8, 16	W:2	
Interrupt priority register A	IPRA	16	H'FFFFE906		8, 16		
Interrupt priority register B	IPRB	16	H'FFFFE908		8, 16		
Interrupt priority register C	IPRC	16	H'FFFFE980		16		
Interrupt priority register D	IPRD	16	H'FFFFE982		16		
Interrupt priority register E	IPRE	16	H'FFFFE984		16		
Interrupt priority register F	IPRF	16	H'FFFFE986		16		
Interrupt priority register H	IPRH	16	H'FFFFE98A		16		
Interrupt priority register I	IPRI	16	H'FFFFE98C		16		
Interrupt priority register J	IPRJ	16	H'FFFFE98E		16		
Interrupt priority register K	IPRK	16	H'FFFFE990		16		
Interrupt priority register L	IPRL	16	H'FFFFE992		16		
Interrupt priority register M	IPRM	16	H'FFFFE994		16		
DMA source address register_0	SAR_0	32	H'FFFFEB20	DMAC	16, 32	Pϕ reference	16 bits
DMA destination address register_0	DAR_0	32	H'FFFFEB24		16, 32	B:2	
DMA transfer count register_0	DMATCR_0	32	H'FFFFEB28		16, 32	W:2	
DMA channel control register_0	CHCR_0	32	H'FFFFEB2C		8, 16, 32	L:4	
DMA source address register_1	SAR_1	32	H'FFFFEB30		16, 32		
DMA destination address register_1	DAR_1	32	H'FFFFEB34		16, 32		
DMA transfer count register_1	DMATCR_1	32	H'FFFFEB38		16, 32		
DMA channel control register_1	CHCR_1	32	H'FFFFEB3C		8, 16, 32		
DMA source address register_2	SAR_2	32	H'FFFFEB40		16, 32		
DMA destination address register_2	DAR_2	32	H'FFFFEB44		16, 32		
DMA transfer count register_2	DMATCR_2	32	H'FFFFEB48		16, 32		
DMA channel control register_2	CHCR_2	32	H'FFFFEB4C		8, 16, 32		
DMA source address register_3	SAR_3	32	H'FFFFEB50		16, 32		
DMA destination address register_3	DAR_3	32	H'FFFFEB54		16, 32		
DMA transfer count register_3	DMATCR_3	32	H'FFFFEB58		16, 32		
DMA channel control register_3	CHCR_3	32	H'FFFFEB5C		8, 16, 32		
DMA operation register	DMAOR	16	H'FFFFEB60		8, 16		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Common control register	CMNCR	32	H'FFFFF000	BSC	32	Bϕ reference	16 bits
CS0 space bus control register	CS0BCR	32	H'FFFFF004	*1: Read	32	L:1*1	
CS1 space bus control register	CS1BCR	32	H'FFFFF008	*2: Write	32	L:3*2	
CS2 space bus control register	CS2BCR	32	H'FFFFF00C		32		
CS3 space bus control register	CS3BCR	32	H'FFFFF010		32		
CS4 space bus control register	CS4BCR	32	H'FFFFF014		32		
CS5 space bus control register	CS5BCR	32	H'FFFFF018		32		
CS6 space bus control register	CS6BCR	32	H'FFFFF01C		32		
CS7 space bus control register	CS7BCR	32	H'FFFFF020		32		
CS8 space bus control register	CS8BCR	32	H'FFFFF024		32		
CS0 space wait control register	CS0WCR	32	H'FFFFF028		32		
CS1 space wait control register	CS1WCR	32	H'FFFFF02C		32		
CS2 space wait control register	CS2WCR	32	H'FFFFF030		32		
CS3 space wait control register	CS3WCR	32	H'FFFFF034		32		
CS4 space wait control register	CS4WCR	32	H'FFFFF038		32		
CS5 space wait control register	CS5WCR	32	H'FFFFF03C		32		
CS6 space wait control register	CS6WCR	32	H'FFFFF040		32		
CS7 space wait control register	CS7WCR	32	H'FFFFF044		32		
CS8 space wait control register	CS8WCR	32	H'FFFFF048		32		
SDRAM control register	SDCR	32	H'FFFFF04C		32		
Refresh timer control/status register	RTCSR	32	H'FFFFF050		32		
Refresh timer counter	RTCNT	32	H'FFFFF054		32		
Refresh time constant register	RTCOR	32	H'FFFFF058		32		
RAM emulation register	RAMER	16	H'FFFFF108	FLASH	16	Bϕ reference W:1	16 bits
Break address register A	BARA	32	H'FFFFF300	UBC	32	Bϕ reference	16 bits
Break address mask register A	BAMRA	32	H'FFFFF304		32	W:3	
Break bus cycle register A	BBRA	16	H'FFFFF308		16	L:3	
Break data register A	BDRA	32	H'FFFFF310		32		
Break data mask register A	BDMRA	32	H'FFFFF314		32		

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Break address register B	BARB	32	H'FFFFF320	UBC	32	Bϕ reference	16 bits
Break address mask register B	BAMRB	32	H'FFFFF324		32	W:3	
Break bus cycle register B	BBRB	16	H'FFFFF328		16	L:3	
Break data register B	BDRB	32	H'FFFFF330		32		
Break data mask register B	BDMRB	32	H'FFFFF334		32		
Break control register	BRCR	32	H'FFFFF3C0		32		
Branch source register	BRSR	32	H'FFFFF3D0		32		
Branch destination register	BRDR	32	H'FFFFF3D4		32		
Execution times break register	BETR	16	H'FFFFF3DC		16		

27.2 Register Bit List

Addresses and bit names of each on-chip peripheral module are shown below.

As for 16-bit or 32-bit registers, they are shown in two or four rows.

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
SCSMR_0	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS[1:0]		SCI
SCBRR_0									(Channel 0)
SCSCR_0	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]		
SCTDR_0									
SCSSR_0	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT	
SCRDR_0									
SCSDCR_0	—	—	—	—	DIR	—	—	—	
SCSPTR_0	EIO	—	—	—	SPB1IO	SPB1DT	SPB0IO	SPB0DT	
SCSMR_1	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS[1:0]		SCI
SCBRR_1									(Channel 1)
SCSCR_1	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]		
SCTDR_1									
SCSSR_1	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT	
SCRDR_1									
SCSDCR_1	—	—	—	—	DIR	—	—	—	
SCSPTR_1	EIO	—	—	—	SPB1IO	SPB1DT	SPB0IO	SPB0DT	
SCSMR_2	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS[1:0]		SCI
SCBRR_2									(Channel 2)
SCSCR_2	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]		
SCTDR_2									
SCSSR_2	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT	
SCRDR_2									
SCSDCR_2	—	—	—	—	DIR	—	—	—	
SCSPTR_2	EIO	—	—	—	SPB1IO	SPB1DT	SPB0IO	SPB0DT	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
SCSMR_3	—	—	—	—	—	—	—	—	SCIF
	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	—	CKS[1:0]		(Channel 3)
SCBRR_3									
SCSCR_3	—	—	—	—	—	—	—	—	
	TIE	RIE	TE	RE	REIE	—	CKE[1:0]		
SCFTDR_3									
SCFSR_3	PER[3:0]				FER[3:0]				
	ER	TEND	TDFE	BRK	FER	PER	RDF	DR	
SCFRDR_3									
SCFCR_3	—	—	—	—	—	RSTRG[2:0]			
	RTRG[1:0]		TTRG[1:0]		MCE	TFRST	RFRST	LOOP	
SCFDR_3	—	—	—	T[4:0]					
	—	—	—	R[4:0]					
SCSPTR_3	—	—	—	—	—	—	—	—	
	RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPBIO	SPBDT	
SCLSR_3	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	ORER	
TCR_3	CCLR[2:0]			CKEG[1:0]		TPSC[2:0]			MTU2
TCR_4	CCLR[2:0]			CKEG[1:0]		TPSC[2:0]			
TMDR_3	—	—	BFB	BFA	MD[3:0]				
TMDR_4	—	—	BFB	BFA	MD[3:0]				
TIORH_3	IOB[3:0]				IOA[3:0]				
TIORL_3	IOD[3:0]				IOC[3:0]				
TIORH_4	IOB[3:0]				IOA[3:0]				
TIORL_4	IOD[3:0]				IOC[3:0]				
TIER_3	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
TIER_4	TTGE	TTGE2	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
TOER	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B	
TGCR	—	BDC	N	P	FB	WF	VF	UF	
TOCR1	—	PSYE	—	—	TOCL	TOCS	OLSN	OLSP	
TOCR2	BF[1:0]		OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
TCNT_3									MTU2
TCNT_4									
TCDR									
TDDR									
TGRA_3									
TGRB_3									
TGRA_4									
TGRB_4									
TCNTS									
TCBR									
TGRC_3									
TGRD_3									
TGRC_4									
TGRD_4									
TSR_3	TCFD	—	—	TCFV	TGFD	TGFC	TGFB	TGFA	
TSR_4	TCFD	—	—	TCFV	TGFD	TGFC	TGFB	TGFA	
TITCR	T3AEN	3ACOR[2:0]			T4VEN	4VCOR[2:0]			

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
TITCNT	—	3ACNT[2:0]			—	4VCNT[2:0]			MTU2
TBTER	—	—	—	—	—	—	BTE[1:0]		
TDER	—	—	—	—	—	—	—	TDER	
TOLBR	—	—	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P	
TBTM_3	—	—	—	—	—	—	TTSB	TTSA	
TBTM_4	—	—	—	—	—	—	TTSB	TTSA	
TADCR	BF[1:0]		—	—	—	—	—	—	
	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE	
TADCORA_4									
TADCORB_4									
TADCOBRA_4									
TADCOBRB_4									
TWCR	CCE	—	—	—	—	—	—	WRE	
TSTR	CST4	CST3	—	—	—	CST2	CST1	CST0	
TSYR	SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0	
TCSYSTR	SCH0	SCH1	SCH2	SCH3	SCH4	—	SCH3S	SCH4S	
TRWER	—	—	—	—	—	—	—	RWE	
TCR_0	CCLR[2:0]			CKEG[1:0]		TPSC[2:0]			
TMDR_0	—	BFE	BFB	BFA	MD[3:0]				
TIORH_0	IOB[3:0]				IOA[3:0]				
TIORL_0	IOD[3:0]				IOC[3:0]				
TIER_0	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
TSR_0	—	—	—	TCFV	TGFD	TGFC	TGFB	TGFA	
TCNT_0									
TGRA_0									

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
TGRB_0									MTU2
TGRC_0									
TGRD_0									
TGRE_0									
TGRF_0									
TIER2_0	TTGE2	—	—	—	—	—	TGIEF	TGIEE	
TSR2_0	—	—	—	—	—	—	TGFF	TGFE	
TBTM_0	—	—	—	—	—	TTSE	TTSB	TTSA	
TCR_1	—	CCLR[1:0]		CKEG[1:0]		TPSC[2:0]			
TMDR_1	—	—	—	—	MD[3:0]				
TIOR_1	IOB[3:0]				IOA[3:0]				
TIER_1	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA	
TSR_1	TCFD	—	TCFU	TCFV	—	—	TGFB	TGFA	
TCNT_1									
TGRA_1									
TGRB_1									
TICCR	—	—	—	—	I2BE	I2AE	I1BE	I1AE	
TCR_2	—	CCLR[1:0]		CKEG[1:0]		TPSC[2:0]			
TMDR_2	—	—	—	—	MD[3:0]				
TIOR_2	IOB[3:0]				IOA[3:0]				
TIER_2	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA	
TSR_2	TCFD	—	TCFU	TCFV	—	—	TGFB	TGFA	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
TCNT_2									MTU2
TGRA_2									
TGRB_2									
TCNTU_5									
TGRU_5									
TCRU_5	—	—	—	—	—	—	TPSC[1:0]		
TIORU_5	—	—	—	IOC[4:0]					
TCNTV_5									
TGRV_5									
TCRV_5	—	—	—	—	—	—	TPSC[1:0]		
TIORV_5	—	—	—	IOC[4:0]					
TCNTW_5									
TGRW_5									
TCRW_5	—	—	—	—	—	—	TPSC[1:0]		
TIORW_5	—	—	—	IOC[4:0]					
TSR_5	—	—	—	—	—	CMFU5	CMFV5	CMFW5	
TIER_5	—	—	—	—	—	TGIE5U	TGIE5V	TGIE5W	
TSTR_5	—	—	—	—	—	CSTU5	CSTV5	CSTW5	
TCNTCMPCLR	—	—	—	—	—	CMPCLR5U	CMPCLR5V	CMPCLR5W	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
TCR_3S	CCLR[2:0]			CKEG[1:0]		TPSC[2:0]			MTU2S
TCR_4S	CCLR[2:0]			CKEG[1:0]		TPSC[2:0]			
TMDR_3S	—	—	BFB	BFA	MD[3:0]				
TMDR_4S	—	—	BFB	BFA	MD[3:0]				
TIORH_3S	IOB[3:0]				IOA[3:0]				
TIORL_3S	IOD[3:0]				IOC[3:0]				
TIORH_4S	IOB[3:0]				IOA[3:0]				
TIORL_4S	IOD[3:0]				IOC[3:0]				
TIER_3S	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
TIER_4S	TTGE	TTGE2	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
TOERS	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B	
TGCRS	—	BDC	N	P	FB	WF	VF	UF	
TOCR1S	—	PSYE	—	—	TOCL	TOCS	OLSN	OLSP	
TOCR2S	BF[1:0]		OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P	
TCNT_3S									
TCNT_4S									
TCDRS									
TDDRS									
TGRA_3S									
TGRB_3S									
TGRA_4S									
TGRB_4S									

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
TCNTSS									MTU2S
TCBRS									
TGRC_3S									
TGRD_3S									
TGRC_4S									
TGRD_4S									
TSR_3S	TCFD	—	—	TCFV	TGFD	TGFC	TGFB	TGFA	
TSR_4S	TCFD	—	—	TCFV	TGFD	TGFC	TGFB	TGFA	
TITCRS	T3AEN	3ACOR[2:0]			T4VEN	4VCOR[2:0]			
TITCNTS	—	3ACNT[2:0]			—	4VCNT[2:0]			
TBTERS	—	—	—	—	—	—	BTE[1:0]		
TDERS	—	—	—	—	—	—	—	TDER	
TOLBRS	—	—	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P	
TBTM_3S	—	—	—	—	—	—	TTSB	TTSA	
TBTM_4S	—	—	—	—	—	—	TTSB	TTSA	
TADCRS	BF[1:0]		—	—	—	—	—	—	
	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE	
TADCORA_4S									
TADCORB_4S									
TADCOBRA_4S									
TADCOBRB_4S									

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
TSYCRS	CE0A	CE0B	CE0C	CE0D	CE1A	CE1B	CE2A	CE2B	MTU2S
TWCRS	CCE	—	—	—	—	—	SCC	WRE	
TSTRS	CST4	CST3	—	—	—	CST2	CST1	CST0	
TSYRS	SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0	
TRWERS	—	—	—	—	—	—	—	RWE	
TCNTU_5S									
TGRU_5S									
TCRU_5S	—	—	—	—	—	—	TPSC[1:0]		
TIORU_5S	—	—	—	IOC[4:0]					
TCNTV_5S									
TGRV_5S									
TCRV_5S	—	—	—	—	—	—	TPSC[1:0]		
TIORV_5S	—	—	—	IOC[4:0]					
TCNTW_5S									
TGRW_5S									
TCRW_5S	—	—	—	—	—	—	TPSC[1:0]		
TIORW_5S	—	—	—	IOC[4:0]					
TSR_5S	—	—	—	—	—	CMFU5	CMFV5	CMFW5	
TIER_5S	—	—	—	—	—	TGIE5U	TGIE5V	TGIE5W	
TSTR_5S	—	—	—	—	—	CSTU5	CSTV5	CSTW5	
TCNTCMPCLRS	—	—	—	—	—	CMPCLR5U	CMPCLR5V	CMPCLR5W	
ADDR0	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D (Channel 0)
	AD1	AD0	—	—	—	—	—	—	
ADDR1	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
ADDR2	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D (Channel 0)
	AD1	AD0	—	—	—	—	—	—	
ADDR3	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADCSR_0	ADF	ADIE	—	—	TRGE	—	CONADF	STC	
	CKSL[1:0]		ADM[1:0]		ADCS	CH[2:0]			
ADCR_0	—	—	ADST	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
ADDR4	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D (Channel 1)
	AD1	AD0	—	—	—	—	—	—	
ADDR5	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADDR6	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADDR7	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADCSR_1	ADF	ADIE	—	—	TRGE	—	CONADF	STC	
	CKSL[1:0]		ADM[1:0]		ADCS	CH[2:0]			
ADCR_1	—	—	ADST	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
ADDR8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D (Channel 2)
	AD1	AD0	—	—	—	—	—	—	
ADDR9	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADDR10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADDR11	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADDR12	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
ADDR13	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D (Channel 2)
	AD1	AD0	—	—	—	—	—	—	
ADDR14	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADDR15	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADCSR_2	ADF	ADIE	—	—	TRGE	—	CONADF	STC	
	CKSL[1:0]		ADM[1:0]		ADCS	CH[2:0]			
ADCR_2	—	—	ADST	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
FCCS	FWE	MAT	—	FLER	—	—	—	SCO	FLASH
FPCS	—	—	—	—	—	—	—	PPVS	
FECS	—	—	—	—	—	—	—	EPVB	
FKEY	K[7:0]								
FMATS	MS7	MS6	MS5	MS4	MS3	MS2	MS1	MS0	
FTDAR	TDER	TDA[6:0]							
DTCERA	DTCERA15	DTCERA14	DTCERA13	DTCERA12	DTCERA11	DTCERA10	DTCERA9	DTCERA8	DTC
	—	—	—	—	—	—	—	—	
DTCERB	DTCERB15	DTCERB14	DTCERB13	DTCERB12	DTCERB11	DTCERB10	DTCERB9	DTCERB8	
	DTCERB7	DTCERB6	DTCERB5	DTCERB4	DTCERB3	DTCERB2	DTCERB1	DTCERB0	
DTCERC	DTCERC15	DTCERC14	DTCERC13	DTCERC12	—	—	—	—	
	—	—	—	—	DTCERC3	DTCERC2	DTCERC1	DTCERC0	
DTCERD	DTCERD15	DTCERD14	DTCERD13	DTCERD12	DTCERD11	DTCERD10	DTCERD9	DTCERD8	
	DTCERD7	DTCERD6	DTCERD5	DTCERD4	DTCERD3	—	—	—	
DTCERE	DTCERE15	DTCERE14	DTCERE13	DTCERE12	DTCERE11	DTCERE10	DTCERE9	DTCERE8	
	DTCERE7	DTCERE6	DTCERE5	DTCERE4	—	—	—	—	
DTCCR	—	—	—	RRS	RCHNE	—	—	ERR	
DTCVBR									
					—	—	—	—	
	—	—	—	—	—	—	—	—	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
ICCR1	ICE	RCVD	MST	TRS	CKS[3:0]				IIC2
ICCR2	BBSY	SCP	SDAO	SDAOP	SCLO	—	IICRST	—	
ICMR	MLS	WAIT	—	—	BCWP	BC[2:0]			
ICIER	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT	
ICSR	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ	
SAR	SVA[6:0]							FS	
ICDRT									
ICDRR									
NF2CYC	—	—	—	—	—	—	—	NF2CYC	
SSCRH	MSS	BIDE	—	SOL	SOLP	—	CSS[1:0]		SSU
SSCRL	FCLRM	SSUMS	SRES	—	—	—	DATS[1:0]		
SSMR	MLS	CPOS	CPHS	—	—	CKS[2:0]			
SSER	TE	RE	—	—	TEIE	TIE	RIE	CEIE	
SSSR	—	ORER	—	—	TEND	TDRE	RDRF	CE	
SSCR2	—	—	—	TENDSTS	SCSATS	SSODTS	—	—	
SSTDR0									
SSTDR1									
SSTDR2									
SSTDR3									
SSRDR0									
SSRDR1									
SSRDR2									
SSRDR3									
CMSTR	—	—	—	—	—	—	—	—	CMT
	—	—	—	—	—	—	STR1	STR0	
CMCSR_0	—	—	—	—	—	—	—	—	
	CMF	CMIE	—	—	—	—	CKS[1:0]		
CMCNT_0									
CMCOR_0									

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
CMCSR_1	—	—	—	—	—	—	—	—	CMT
	CMF	CMIE	—	—	—	—	CKS[1:0]		
CMCNT_1									
CMCOR_1									
ICSR1	POE3F	POE2F	POE1F	POE0F	—	—	—	PIE1	POE
	POE3M[1:0]		POE2M[1:0]		POE1M[1:0]		POE0M[1:0]		
OCSR1	OSF1	—	—	—	—	—	OCE1	OIE1	
	—	—	—	—	—	—	—	—	
ICSR2	POE7F	POE6F	POE5F	POE4F	—	—	—	PIE2	
	POE7M[1:0]		POE6M[1:0]		POE5M[1:0]		POE4M[1:0]		
OCSR2	OSF2	—	—	—	—	—	OCE2	OIE2	
	—	—	—	—	—	—	—	—	
ICSR3	—	—	—	POE8F	—	—	POE8E	PIE3	
	—	—	—	—	—	—	POE8M[1:0]		
SPOER	—	—	—	—	—	MTU2SHIZ	MTU2CH0HIZ	MTU2CH34HIZ	
POECR1	—	—	—	—	MTU2PE3ZE	MTU2PE2ZE	MTU2PE1ZE	MTU2PE0ZE	
POECR2	—	MTU2P1CZE	MTU2P2CZE	MTU2P3CZE	—	MTU2SP1CZE	MTU2SP2CZE	MTU2SP3CZE	
	—	MTU2SP4CZE	MTU2SP5CZE	MTU2SP6CZE	—	MTU2SP7CZE	MTU2SP8CZE	MTU2SP9CZE	
PADRH*[1]	—	—	PA29DR	PA28DR	PA27DR	PA26DR	PA25DR	PA24DR	I/O
	PA23DR	PA22DR	PA21DR	PA20DR	PA19DR	PA18DR	PA17DR	PA16DR	
PADRL*[1]	PA15DR	PA14DR	PA13DR	PA12DR	PA11DR	PA10DR	PA9DR	PA8DR	
	PA7DR	PA6DR	PA5DR	PA4DR	PA3DR	PA2DR	PA1DR	PA0DR	
PAIORH	—	—	PA29IOR	PA28IOR	PA27IOR	PA26IOR	PA25IOR	PA24IOR	PFC
	PA23IOR	PA22IOR	PA21IOR	PA20IOR	PA19IOR	PA18IOR	PA17IOR	PA16IOR	
PAIORL	PA15IOR	PA14IOR	PA13IOR	PA12IOR	PA11IOR	PA10IOR	PA9IOR	PA8IOR	
	PA7IOR	PA6IOR	PA5IOR	PA4IOR	PA3IOR	PA2IOR	PA1IOR	PA0IOR	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
PACRH4*1	—	—	—	—	—	—	—	—	PFC
	—	—	PA29MD1	PA29MD0	—	—	PA28MD1	PA28MD0	
PACRH3*1	—	—	PA27MD1	PA27MD0	—	—	PA26MD1	PA26MD0	
	—	—	PA25MD1	PA25MD0	—	—	PA24MD1	PA24MD0	
PACRH2*1	—	—	PA23MD1	PA23MD0	—	—	PA22MD1	PA22MD0	
	—	—	PA21MD1	PA21MD0	—	—	PA20MD1	PA20MD0	
PACRH1*1	—	—	PA19MD1	PA19MD0	—	—	PA18MD1	PA18MD0	
	—	—	PA17MD1	PA17MD0	—	PA16MD2	PA16MD1	PA16MD0	
PACRL4*1	—	PA15MD2	PA15MD1	PA15MD0	—	PA14MD2	PA14MD1	PA14MD0	
	—	PA13MD2	PA13MD1	PA13MD0	—	PA12MD2	PA12MD1	PA12MD0	
PACRL3*1	—	PA11MD2	PA11MD1	PA11MD0	—	PA10MD2	PA10MD1	PA10MD0	
	—	PA9MD2	PA9MD1	PA9MD0	—	PA8MD2	PA8MD1	PA8MD0	
PACRL2*1	—	PA7MD2	PA7MD1	PA7MD0	—	PA6MD2	PA6MD1	PA6MD0	
	—	PA5MD2	PA5MD1	PA5MD0	—	PA4MD2	PA4MD1	PA4MD0	
PACRL1*1	—	PA3MD2	PA3MD1	PA3MD0	—	PA2MD2	PA2MD1	PA2MD0	
	—	PA1MD2	PA1MD1	PA1MD0	—	PA0MD2	PA0MD1	PA0MD0	
PAPRH*1	—	—	PA29PR	PA28PR	PA27PR	PA26PR	PA25PR	PA24PR	I/O
	PA23PR	PA22PR	PA21PR	PA20PR	PA19PR	PA18PR	PA17PR	PA16PR	
PAPRL*1	PA15PR	PA14PR	PA13PR	PA12PR	PA11PR	PA10PR	PA9PR	PA8PR	
	PA7PR	PA6PR	PA5PR	PA4PR	PA3PR	PA2PR	PA1PR	PA0PR	
PBDRL*1	—	—	—	—	—	—	PB9DR	PB8DR	
	PB7DR	PB6DR	PB5DR	PB4DR	PB3DR	PB2DR	PB1DR	PB0DR	
PBIORL	—	—	—	—	—	—	PB9IOR	PB8IOR	PFC
	PB7IOR	PB6IOR	PB5IOR	PB4IOR	PB3IOR	PB2IOR	PB1IOR	PB0IOR	
PBCRL3	—	—	—	—	—	—	—	—	
	—	PB9MD2	PB9MD1	PB9MD0	—	PB8MD2	PB8MD1	PB8MD0	
PBCRL2	—	PB7MD2	PB7MD1	PB7MD0	—	PB6MD2	PB6MD1	PB6MD0	
	—	PB5MD2	PB5MD1	PB5MD0	—	PB4MD2	PB4MD1	PB4MD0	
PBCRL1	—	PB3MD2	PB3MD1	PB3MD0	—	PB2MD2	PB2MD1	PB2MD0	
	—	PB1MD2	PB1MD1	PB1MD0	—	PB0MD2	PB0MD1	PB0MD0	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
PBPRL	—	—	—	—	—	—	PB9PR	PB8PR	I/O
	PB7PR	PB6PR	PB5PR	PB4PR	PB3PR	PB2PR	PB1PR	PB0PR	
PCDRH	—	—	—	—	—	—	PC25DR	PC24DR	
	PC23DR	PC22DR	PC21DR	PC20DR	PC19DR	PC18DR	—	—	
PCDRL	PC15DR	PC14DR	PC13DR	PC12DR	PC11DR	PC10DR	PC9DR	PC8DR	
	PC7DR	PC6DR	PC5DR	PC4DR	PC3DR	PC2DR	PC1DR	PC0DR	
PCIORH	—	—	—	—	—	—	PC25IOR	PC24IOR	PFC
	PC23IOR	PC22IOR	PC21IOR	PC20IOR	PC19IOR	PC18IOR	—	—	
PCIORL	PC15IOR	PC14IOR	PC13IOR	PC12IOR	PC11IOR	PC10IOR	PC9IOR	PC8IOR	
	PC7IOR	PC6 IOR	PC5IOR	PC4IOR	PC3IOR	PC2IOR	PC1IOR	PC0IOR	
PCCRH3*[1]	—	—	—	—	—	—	—	—	
	—	—	—	PC25MD0	—	—	—	PC24MD0	
PCCRH2*[1]	—	—	—	PC23MD0	—	—	—	PC22MD0	
	—	—	—	PC21MD0	—	—	—	PC20MD0	
PCCRH1*[1]	—	—	—	PC19MD0	—	—	—	PC18MD0	
	—	—	—	—	—	—	—	—	
PCCRL4	—	—	—	PC15MD0	—	—	—	PC14MD0	
	—	—	—	PC13MD0	—	—	—	PC12MD0	
PCCRL3	—	—	—	PC11MD0	—	—	—	PC10MD0	
	—	—	—	PC9MD0	—	—	—	PC8MD0	
PCCRL2	—	—	—	PC7MD0	—	—	—	PC6MD0	
	—	—	—	PC5MD0	—	—	—	PC4MD0	
PCCRL1	—	—	—	PC3MD0	—	—	—	PC2MD0	
	—	—	—	PC1MD0	—	—	—	PC0MD0	
PCPRH*[1]	—	—	—	—	—	—	PC25PR	PC24PR	I/O
	PC23PR	PC22PR	PC21PR	PC20PR	PC19PR	PC18PR	—	—	
PCPRL	PC15PR	PC14PR	PC13PR	PC12PR	PC11PR	PC10PR	PC9PR	PC8PR	
	PC7PR	PC6PR	PC5PR	PC4PR	PC3PR	PC2PR	PC1PR	PC0PR	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
PDDRH*1	PD31DR	PD30DR	PD29DR	PD28DR	PD27DR	PD26DR	PD25DR	PD24DR	I/O
	PD23DR	PD22DR	PD21DR	PD20DR	PD19DR	PD18DR	PD17DR	PD16DR	
PDDRL	PD15DR	PD14DR	PD13DR	PD12DR	PD11DR	PD10DR	PD9DR	PD8DR	
	PD7DR	PD6DR	PD5DR	PD4DR	PD3DR	PD2DR	PD1DR	PD0DR	
PDIORH	PD31IOR	PD30IOR	PD29IOR	PD28IOR	PD27IOR	PD26IOR	PD25IOR	PD24IOR	PFC
	PD23IOR	PD22IOR	PD21IOR	PD20IOR	PD19IOR	PD18IOR	PD17IOR	PD16IOR	
PDIORL	PD15IOR	PD14IOR	PD13IOR	PD12IOR	PD11IOR	PD10IOR	PD9IOR	PD8IOR	
	PD7IOR	PD6IOR	PD5IOR	PD4IOR	PD3IOR	PD2IOR	PD1IOR	PD0IOR	
PDCRH4*1	—	—	PD31MD1	PD31MD0	—	—	PD30MD1	PD30MD0	
	—	—	PD29MD1	PD29MD0	—	—	PD28MD1	PD28MD0	
PDCRH3*1	—	—	PD27MD1	PD27MD0	—	—	PD26MD1	PD26MD0	
	—	—	PD25MD1	PD25MD0	—	—	PD24MD1	PD24MD0	
PDCRH2*1	—	—	PD23MD1	PD23MD0	—	PD22MD2	PD22MD1	PD22MD0	
	—	PD21MD2	PD21MD1	PD21MD0	—	PD20MD0	PD20MD1	PD20MD0	
PDCRH1*1	—	PD19MD2	PD19MD1	PD19MD0	—	PD18MD2	PD18MD1	PD18MD0	
	—	PD17MD2	PD17MD1	PD17MD0	—	PD16MD2	PD16MD1	PD16MD0	
PDCRL4	—	—	PD15MD1	PD15MD0	—	—	PD14MD1	PD14MD0	
	—	—	PD13MD1	PD13MD0	—	—	PD12MD1	PD12MD0	
PDCRL3	—	—	PD11MD1	PD11MD0	—	PD10MD2	PD10MD1	PD10MD0	
	—	PD9MD2	PD9MD1	PD9MD0	—	PD8MD2	PD8MD1	PD8MD0	
PDCRL2	—	PD7MD2	PD7MD1	PD7MD0	—	PD6MD2	PD6MD1	PD6MD0	
	—	PD5MD2	PD5MD1	PD5MD0	—	PD4MD2	PD4MD1	PD4MD0	
PDCRL1	—	PD3MD2	PD3MD1	PD3MD0	—	PD2MD2	PD2MD1	PD2MD0	
	—	PD1MD2	PD1MD1	PD1MD0	—	PD0MD2	PD0MD1	PD0MD0	
PDPRH*1	PD31PR	PD30PR	PD29PR	PD28PR	PD27PR	PD26PR	PD25PR	PD24PR	I/O
	PD23PR	PD22PR	PD21PR	PD20PR	PD19PR	PD18PR	PD17PR	PD16PR	
PDPRL	PD15PR	PD14PR	PD13PR	PD12PR	PD11PR	PD10PR	PD9PR	PD8PR	
	PD7PR	PD6PR	PD5PR	PD4PR	PD3PR	PD2PR	PD1PR	PD0PR	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
PEDRH*[1]	—	—	—	—	—	—	—	—	I/O
	—	—	PE21DR	PE20DR	PE19DR	PE18DR	PE17DR	PE16DR	
PEDRL*[1]	PE15DR	PE14DR	PE13DR	PE12DR	PE11DR	PE10DR	PE9DR	PE8DR	
	PE7DR	PE6DR	PE5DR	PE4DR	PE3DR	PE2DR	PE1DR	PE0DR	
PEIORH	—	—	—	—	—	—	—	—	PFC
	—	—	PE21IOR	PE20IOR	PE19IOR	PE18IOR	PE17IOR	PE16IOR	
PEIORL	PE15IOR	PE14IOR	PE13IOR	PE12IOR	PE11IOR	PE10IOR	PE9IOR	PE8IOR	
	PE7IOR	PE6IOR	PE5IOR	PE4IOR	PE3IOR	PE2IOR	PE1IOR	PE0IOR	
PECRH2*[1]	—	—	—	—	—	—	—	—	
	—	—	PE21MD1	PE21MD0	—	—	PE20MD1	PE20MD0	
PECRH1*[1]	—	—	PE19MD1	PE19MD0	—	—	PE18MD1	PE18MD0	
	—	—	PE17MD1	PE17MD0	—	PE16MD2	PE16MD1	PE16MD0	
PECRL4	—	PE15MD2	PE15MD1	PE15MD0	—	PE14MD2	PE14MD1	PE14MD0	
	—	—	PE13MD1	PE13MD0	—	PE12MD2	PE12MD1	PE12MD0	
PECRL3*[1]	—	PE11MD2	PE11MD1	PE11MD0	—	PE10MD2	PE10MD1	PE10MD0	
	—	PE9MD2	PE9MD1	PE9MD0	—	PE8MD2	PE8MD1	PE8MD0	
PECRL2*[1]	—	PE7MD2	PE7MD1	PE7MD0	—	PE6MD2	PE6MD1	PE6MD0	
	—	PE5MD2	PE5MD1	PE5MD0	—	PE4MD2	PE4MD1	PE4MD0	
PECRL1	—	PE3MD2	PE3MD1	PE3MD0	—	PE2MD2	PE2MD1	PE2MD0	
	—	PE1MD2	PE1MD1	PE1MD0	—	—	PE0MD1	PE0MD0	
PEPRH*[1]	—	—	—	—	—	—	—	—	I/O
	—	—	PE21PR	PE20PR	PE19PR	PE18PR	PE17PR	PE16PR	
PEPRL*[1]	PE15PR	PE14PR	PE13PR	PE12PR	PE11PR	PE10PR	PE9PR	PE8PR	
	PE7PR	PE6PR	PE5PR	PE4PR	PE3PR	PE2PR	PE1PR	PE0PR	
HCPCR	—	—	—	—	—	—	—	—	PFC
	—	—	—	—	MZIZDH	MZIZDL	MZIZEH	MZIZEL	
IFCR	—	—	—	—	—	—	—	—	
	—	—	—	—	IRQMD3	IRQMD2	IRQMD1	IRQMD0	
PFDRL*[1]	PF15DR	PF14DR	PF13DR	PF12DR	PF11DR	PF10DR	PF9DR	PF8DR	I/O
	PF7DR	PF6DR	PF5DR	PF4DR	PF3DR	PF2DR	PF1DR	PF0DR	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
FRQCR	—	IFC[2:0]			BFC[2:0]			PFC[2]	CPG
	PFC[1:0]		MIFC[2:0]			MPFC[2:0]			
STBCR1	STBY	—	—	—	—	—	—	—	Power-down modes
STBCR2	MSTP7	MSTP6	—	MSTP4	MSTP3	—	—	—	
STBCR3	MSTP15	MSTP14	MSTP13	MSTP12	MSTP11	MSTP10	—	—	
STBCR4	MSTP23	MSTP22	MSTP21	—	—	MSTP18	MSTP17	MSTP16	
STBCR5	—	—	—	—	—	—	MSTP25	MSTP24	
STBCR6	AUDSRST	HIZ	—	—	—	—	STBYMD	—	
WTCNT									WDT
WTCSR	TME	WT/$\overline{IT}$	RSTS	WOVF	IOVF	CKS[2:0]			
OSCCR	—	—	—	—	—	OSCSTOP	—	OSCERS	CPG
RAMCR	—	—	—	RAME	—	—	—	—	Power-down modes
ADTSR_0	TRG11S[3:0]				TRG01S[3:0]				A/D
	TRG1S[3:0]				TRG0S[3:0]				
ADTSR_1	TRG2S[3:0]				—	—	—	—	
	—	—	—	—	—	—	—	—	
BSCEHR	DTLOCK	CSSTP1	—	CSSTP2	DTBST	DTSA	CSSTP3	DTPR	BSC
	—	—	—	DMMTU4	DMMTU3	DMMTU2	DMMTU1	DMMTU0	
ICR0	NMIL	—	—	—	—	—	—	NMIE	INTC
	—	—	—	—	—	—	—	—	
IRQCR	IRQ71S	IRQ70S	IRQ61S	IRQ60S	IRQ51S	IRQ50S	IRQ41S	IRQ40S	
	IRQ31S	IRQ30S	IRQ21S	IRQ20S	IRQ11S	IRQ10S	IRQ01S	IRQ00S	
IRQSR	IRQ7L	IRQ6L	IRQ5L	IRQ4L	IRQ3L	IRQ2L	IRQ1L	IRQ0L	
	IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F	
IPRA	IRQ0	IRQ0	IRQ0	IRQ0	IRQ1	IRQ1	IRQ1	IRQ1	
	IRQ2	IRQ2	IRQ2	IRQ2	IRQ3	IRQ3	IRQ3	IRQ3	
IPRB	IRQ4	IRQ4	IRQ4	IRQ4	IRQ5	IRQ5	IRQ5	IRQ5	
	IRQ6	IRQ6	IRQ6	IRQ6	IRQ7	IRQ7	IRQ7	IRQ7	
IPRC	DMAC_0	DMAC_0	DMAC_0	DMAC_0	DMAC_1	DMAC_1	DMAC_1	DMAC_1	
	DMAC_2	DMAC_2	DMAC_2	DMAC_2	DMAC_3	DMAC_3	DMAC_3	DMAC_3	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
IPRD	MTU2_0	MTU2_0	MTU2_0	MTU2_0	MTU2_0	MTU2_0	MTU2_0	MTU2_0	INTC
	MTU2_1	MTU2_1	MTU2_1	MTU2_1	MTU2_1	MTU2_1	MTU2_1	MTU2_1	
IPRE	MTU2_2	MTU2_2	MTU2_2	MTU2_2	MTU2_2	MTU2_2	MTU2_2	MTU2_2	
	MTU2_3	MTU2_3	MTU2_3	MTU2_3	MTU2_3	MTU2_3	MTU2_3	MTU2_3	
IPRF	MTU2_4	MTU2_4	MTU2_4	MTU2_4	MTU2_4	MTU2_4	MTU2_4	MTU2_4	
	MTU2_5	MTU2_5	MTU2_5	MTU2_5	POE(MTU2)	POE(MTU2)	POE(MTU2)	POE(MTU2)	
IPRH	—	—	—	—	IIC2	IIC2	IIC2	IIC2	
	MTU2S_3	MTU2S_3	MTU2S_3	MTU2S_3	MTU2S_3	MTU2S_3	MTU2S_3	MTU2S_3	
IPRI	MTU2S_4	MTU2S_4	MTU2S_4	MTU2S_4	MTU2S_4	MTU2S_4	MTU2S_4	MTU2S_4	
	MTU2S_5	MTU2S_5	MTU2S_5	MTU2S_5	POE(MTU2S)	POE(MTU2S)	POE(MTU2S)	POE(MTU2S)	
IPRJ	CMT_0	CMT_0	CMT_0	CMT_0	CMT_1	CMT_1	CMT_1	CMT_1	
	BSC	BSC	BSC	BSC	WDT	WDT	WDT	WDT	
IPRK	A/D_0,1	A/D_0,1	A/D_0,1	A/D_0,1	A/D_2	A/D_2	A/D_2	A/D_2	
	—	—	—	—	—	—	—	—	
IPRL	SCI_0	SCI_0	SCI_0	SCI_0	SCI_1	SCI_1	SCI_1	SCI_1	
	SCI_2	SCI_2	SCI_2	SCI_2	SCIF	SCIF	SCIF	SCIF	
IPRM	SSU	SSU	SSU	SSU	IIC2	IIC2	IIC2	IIC2	
	—	—	—	—	—	—	—	—	
SAR_0									DMAC
DAR_0									
DMATCR_0									

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
CHCR_0	—	—	—	—	—	—	—	—	DMAC
	DO	TL	—	—	—	—	AM	AL	
	DM[1:0]		SM[1:0]		RS[3:0]				
	DL	DS	TB	TS[1:0]		IE	TE	DE	
SAR_1									
DAR_1									
DMATCR_1									
CHCR_1	—	—	—	—	—	—	—	—	
	DO	TL	—	—	—	—	AM	AL	
	DM[1:0]		SM[1:0]		RS[3:0]				
	DL	DS	TB	TS[1:0]		IE	TE	DE	
SAR_2									
DAR_2									

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
DMATCR_2									DMAC
CHCR_2	—	—	—	—	—	—	—	—	
	DO	TL	—	—	—	—	AM	AL	
	DM[1:0]		SM[1:0]		RS[3:0]				
	DL	DS	TB	TS[1:0]		IE	TE	DE	
SAR_3									
DAR_3									
DMATCR_3									
CHCR_3	—	—	—	—	—	—	—	—	
	DO	TL	—	—	—	—	AM	AL	
	DM[1:0]		SM[1:0]		RS[3:0]				
	DL	DS	TB	TS[1:0]		IE	TE	DE	
DMAOR	—	—	CMS[1:0]		—	—	PR[1:0]		
	—	—	—	—	—	AE	NMIF	DME	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
CMNCR	—	—	—	—	—	—	—	—	BSC
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	DMAIW[1:0]		DMAIWA	—	—	—	HIZMEM	HIZCNT	
CS0BCR	—	—	IWW[1:0]		—	IWRWD[1:0]		—	
	IWRWS[1:0]		—	IWRRD[1:0]		—	IWRRS[1:0]		
	—	TYPE[2:0]			—	BSZ[1:0]		—	
	—	—	—	—	—	—	—	—	
CS1BCR	—	—	IWW[1:0]		—	IWRWD[1:0]		—	
	IWRWS[1:0]		—	IWRRD[1:0]		—	IWRRS[1:0]		
	—	TYPE[2:0]			—	BSZ[1:0]		—	
	—	—	—	—	—	—	—	—	
CS2BCR	—	—	IWW[1:0]		—	IWRWD[1:0]		—	
	IWRWS[1:0]		—	IWRRD[1:0]		—	IWRRS[1:0]		
	—	TYPE[2:0]			—	BSZ[1:0]		—	
	—	—	—	—	—	—	—	—	
CS3BCR	—	—	IWW[1:0]		—	IWRWD[1:0]		—	
	IWRWS[1:0]		—	IWRRD[1:0]		—	IWRRS[1:0]		
	—	TYPE[2:0]			—	BSZ[1:0]		—	
	—	—	—	—	—	—	—	—	
CS4BCR	—	—	IWW[1:0]		—	IWRWD[1:0]		—	
	IWRWS[1:0]		—	IWRRD[1:0]		—	IWRRS[1:0]		
	—	TYPE[2:0]			—	BSZ[1:0]		—	
	—	—	—	—	—	—	—	—	
CS5BCR	—	—	IWW[1:0]		—	IWRWD[1:0]		—	
	IWRWS[1:0]		—	IWRRD[1:0]		—	IWRRS[1:0]		
	—	TYPE[2:0]			—	BSZ[1:0]		—	
	—	—	—	—	—	—	—	—	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
CS6BCR	—	—	IWW[1:0]		—	IWRWD[1:0]		—	BSC
	IWRWS[1:0]		—	IWRRD[1:0]		—	IWRRS[1:0]		
	—	TYPE[2:0]			—	BSZ[1:0]		—	
	—	—	—	—	—	—	—	—	
CS7BCR	—	—	IWW[1:0]		—	IWRWD[1:0]		—	
	IWRWS[1:0]		—	IWRRD[1:0]		—	IWRRS[1:0]		
	—	TYPE[2:0]			—	BSZ[1:0]		—	
	—	—	—	—	—	—	—	—	
CS8BCR	—	—	IWW[1:0]		—	IWRWD[1:0]		—	
	IWRWS[1:0]		—	IWRRD[1:0]		—	IWRRS[1:0]		
	—	TYPE[2:0]			—	BSZ[1:0]		—	
	—	—	—	—	—	—	—	—	
CS0WCR*2	—	—	—	—	—	—	—	—	
	—	—	—	BAS	—	WW[2:0]			
	—	—	—	SW[1:0]		WR[3:1]			
	WR[0]	WM	—	—	—	—	HW[1:0]		
CS0WCR*3	—	—	—	—	—	—	—	—	
	—	—	—	BEN	—	—	BW[1:0]		
	—	—	—	SW[1:0]		W[3:1]			
	W[0]	WM	—	—	—	—	HW[1:0]		
CS0WCR*4	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	BW[1:0]		
	—	—	—	—	—	W[3:1]			
	W[0]	WM	—	—	—	—	—	—	
CS1WCR*2	—	—	—	—	—	—	—	—	
	—	—	—	BAS	—	WW[2:0]			
	—	—	—	SW[1:0]		WR[3:1]			
	WR[0]	WM	—	—	—	—	HW[1:0]		

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
CS2WCR*2	—	—	—	—	—	—	—	—	BSC
	—	—	—	BAS	—	WW[2:0]			
	—	—	—	SW[1:0]		WR[3:1]			
	WR[0]	WM	—	—	—	—	HW[1:0]		
CS2WCR*5	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	A2CL[1]	
	A2CL[0]	—	—	—	—	—	—	—	
CS3WCR*2	—	—	—	—	—	—	—	—	
	—	—	—	BAS	—	WW[2:0]			
	—	—	—	SW[1:0]		WR[3:1]			
	WR[0]	WM	—	—	—	—	HW[1:0]		
CS3WCR*5	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	WTRP[1:0]		—	WTRCD[1:0]		—	A3CL[1]	
	A3CL[0]	—	—	TRWL[1:0]		—	WTRC[1:0]		
CS4WCR*2	—	—	—	—	—	—	—	—	
	—	—	—	BAS	—	WW[2:0]			
	—	—	—	SW[1:0]		WR[3:1]			
	WR[0]	WM	—	—	—	—	HW[1:0]		
CS4WCR*3	—	—	—	—	—	—	—	—	
	—	—	—	BEN	—	—	BW[1:0]		
	—	—	—	SW[1:0]		W[3:1]			
	W[0]	WM	—	—	—	—	HW[1:0]		
CS5WCR*2	—	—	—	—	—	—	—	—	
	—	—	—	BAS	—	WW[2:0]			
	—	—	—	SW[1:0]		WR[3:1]			
	WR[0]	WM	—	—	—	—	HW[1:0]		

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
CS5WCR*6	—	—	—	—	—	—	—	—	BSC
	—	—	SZSEL	MPXW	—	WW[2:0]			
	—	—	—	SW[1:0]		WR[3:1]			
	WR[0]	WM	—	—	—	—	HW[1:0]		
CS5WCR*7	—	—	—	—	—	—	—	—	
	—	—	SA[1:0]		—	—	—	—	
	—	TED[3:0]				PCW[3:1]			
	PCW[0]	WM	—	—	TEH[3:0]				
CS6WCR*2	—	—	—	—	—	—	—	—	
	—	—	—	BAS		WW[2:0]			
	—	—	—	SW[1:0]		WR[3:1]			
	WR[0]	WM	—	—	—	—	HW[1:0]		
CS6WCR*7	—	—	—	—	—	—	—	—	
	—	—	SA[1:0]		—	—	—	—	
	—	TED[3:0]				PCW[3:1]			
	PCW[0]	WM	—	—	TEH[3:0]				
CS6WCR*8	—	—	—	—	—	—	—	—	
	—	—	MPXAW[1:0]		MPXMD	—	BW[1:0]		
	—	—	—	—	—	W[3:1]			
	W[0]	WM	—	—	—	—	—	—	
CS7WCR*2	—	—	—	—	—	—	—	—	
	—	—	—	BAS	—	WW[2:0]			
	—	—	—	SW[1:0]		WR[3:1]			
	WR[0]	WM	—	—	—	—	HW[1:0]		
CS8WCR*2	—	—	—	—	—	—	—	—	
	—	—	—	BAS	—	WW[2:0]			
	—	—	—	SW[1:0]		WR[3:1]			
	WR[0]	WM	—	—	—	—	HW[1:0]		

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
SDCR	—	—	—	—	—	—	—	—	BSC
	—	—	—	A2ROW[1:0]		—	A2COL[1:0]		
	—	—	—	—	RFSH	RMODE	—	BACTV	
	—	—	—	A3ROW[1:0]		—	A3COL[1:0]		
RTCSR	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	CMF	CMIE	CKS[2:0]			RRC[2:0]			
RTCNT	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
RTCOR	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
RAMER	—	—	—	—	—	—	—	—	FLASH
	—	—	—	—	RAMS	RAM[2:0]			
BARA	BAA31	BAA30	BAA29	BAA28	BAA27	BAA26	BAA25	BAA24	UBC
	BAA23	BAA22	BAA21	BAA20	BAA19	BAA18	BAA17	BAA16	
	BAA15	BAA14	BAA13	BAA12	BAA11	BAA10	BAA9	BAA8	
	BAA7	BAA6	BAA5	BAA4	BAA3	BAA2	BAA1	BAA0	
BAMRA	BAMA31	BAMA30	BAMA29	BAMA28	BAMA27	BAMA26	BAMA25	BAMA24	
	BAMA23	BAMA22	BAMA21	BAMA20	BAMA19	BAMA18	BAMA17	BAMA16	
	BAMA15	BAMA14	BAMA13	BAMA12	BAMA11	BAMA10	BAMA9	BAMA8	
	BAMA7	BAMA6	BAMA5	BAMA4	BAMA3	BAMA2	BAMA1	BAMA0	
BBRA	—	—	—	—	—	CPA[2:0]			
	CDA[1:0]		IDA[1:0]		RWA[1:0]		SZA[1:0]		

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
BDRA	BDA31	BDA30	BDA29	BDA28	BDA27	BDA26	BDA25	BDA24	UBC
	BDA23	BDA22	BDA21	BDA20	BDA19	BDA18	BDA17	BDA16	
	BDA15	BDA14	BDA13	BDA12	BDA11	BDA10	BDA9	BDA8	
	BDA7	BDA6	BDA5	BDA4	BDA3	BDA2	BDA1	BDA0	
BDMRA	BDMA31	BDMA30	BDMA29	BDMA28	BDMA27	BDMA26	BDMA25	BDMA24	
	BDMA23	BDMA22	BDMA21	BDMA20	BDMA19	BDMA18	BDMA17	BDMA16	
	BDMA15	BDMA14	BDMA13	BDMA12	BDMA11	BDMA10	BDMA9	BDMA8	
	BDMA7	BDMA6	BDMA5	BDMA4	BDMA3	BDMA2	BDMA1	BDMA0	
BARB	BAB31	BAB30	BAB29	BAB28	BAB27	BAB26	BAB25	BAB24	
	BAB23	BAB22	BAB21	BAB20	BAB19	BAB18	BAB17	BAB16	
	BAB15	BAB14	BAB13	BAB12	BAB11	BAB10	BAB9	BAB8	
	BAB7	BAB6	BAB5	BAB4	BAB3	BAB2	BAB1	BAB0	
BAMRB	BAMB31	BAMB30	BAMB29	BAMB28	BAMB27	BAMB26	BAMB25	BAMB24	
	BAMB23	BAMB22	BAMB21	BAMB20	BAMB19	BAMB18	BAMB17	BAMB16	
	BAMB15	BAMB14	BAMB13	BAMB12	BAMB11	BAMB10	BAMB9	BAMB8	
	BAMB7	BAMB6	BAMB5	BAMB4	BAMB3	BAMB2	BAMB1	BAMB0	
BBRB	—	—	—	—	—	CPB[2:0]			
	CDB[1:0]		IDB[1:0]		RWB[1:0]		SZB[1:0]		
BDRB	BDB31	BDB30	BDB29	BDB28	BDB27	BDB26	BDB25	BDB24	
	BDB23	BDB22	BDB21	BDB20	BDB19	BDB18	BDB17	BDB16	
	BDB15	BDB14	BDB13	BDB12	BDB11	BDB10	BDB9	BDB8	
	BDB7	BDB6	BDB5	BDB4	BDB3	BDB2	BDB1	BDB0	
BDMRB	BDMB31	BDMB30	BDMB29	BDMB28	BDMB27	BDMB26	BDMB25	BDMB24	
	BDMB23	BDMB22	BDMB21	BDMB20	BDMB19	BDMB18	BDMB17	BDMB16	
	BDMB15	BDMB14	BDMB13	BDMB12	BDMB11	BDMB10	BDMB9	BDMB8	
	BDMB7	BDMB6	BDMB5	BDMB4	BDMB3	BDMB2	BDMB1	BDMB0	
BRCR	—	—	—	—	—	—	—	—	
	—	—	UTRGW[1:0]		UBIDB	—	UBIDA	—	
	SCMFCA	SCMFCB	SCMFDA	SCMFDB	PCTE	PCBA	—	—	
	DBEA	PCBB	DBEB	—	SEQ	—	—	ETBE	

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
BRSR	SVF	—	—	—	BSA27	BSA26	BSA25	BSA24	UBC
	BSA23	BSA22	BSA21	BSA20	BSA19	BSA18	BSA17	BSA16	
	BSA15	BSA14	BSA13	BSA12	BSA11	BSA10	BSA9	BSA8	
	BSA7	BSA6	BSA5	BSA4	BSA3	BSA2	BSA1	BSA0	
BRDR	DVF	—	—	—	BDA27	BDA26	BDA25	BDA24	
	BDA23	BDA22	BDA21	BDA20	BDA19	BDA18	BDA17	BDA16	
	BDA15	BDA14	BDA13	BDA12	BDA11	BDA10	BDA9	BDA8	
	BDA7	BDA6	BDA5	BDA4	BDA3	BDA2	BDA1	BDA0	
BETR	—	—	—	—	BET[11:8]				
	BET[7:0]								

Notes: 1. The bit value in the register depends on the individual product. For details, see the description of each register.
2. When the memory type of the normal space or SRAM with byte selection is set.
3. When the memory type of the burst ROM (clock asynchronous) is set.
4. When the memory type of the burst ROM (clock synchronous) is set.
5. When the memory type of the SDRAM is set.
6. When the memory type of the MPX-I/O is set.
7. When the memory type of the PCMCIA is set.
8. When the memory type of the burst MPX-I/O is set.

27.3 Register States in Each Operating Mode

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
SCSMR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	SCI (Channel 0)
SCBRR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSCR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCTDR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSSR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCRDR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSDCR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSPTR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSMR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	SCI (Channel 1)
SCBRR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSCR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCTDR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSSR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCRDR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSDCR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSPTR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSMR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	SCI (Channel 2)
SCBRR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSCR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCTDR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSSR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCRDR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSDCR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSPTR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSMR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	SCIF (Channel 3)
SCBRR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSCR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCFTDR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCFSR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
SCFRDR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	SCIF (Channel 3)
SCFCR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCFDR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCSPTR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SCLSR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	MTU2
TCR_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TMDR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TMDR_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORH_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORL_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORH_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORL_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIER_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIER_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TOER	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGCR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TOCR1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TOCR2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNT_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNT_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCDR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TDDR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRA_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRB_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRA_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRB_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNTS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCBR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRC_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRD_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
TGRC_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	MTU2
TGRD_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSR_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSR_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TITCR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TITCNT	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TBTER	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TDER	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TOLBR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TBTM_3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TBTM_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TADCR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TADCORA_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TADCORB_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TADCOBRA_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TADCOBRB_4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TWCR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSTR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSYR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCSYSTR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TRWER	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TMDR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORH_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORL_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIER_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNT_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRA_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRB_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRC_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
TGRD_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	MTU2
TGRE_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRF_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIER2_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSR2_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TBTM_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TMDR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIOR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIER_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNT_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRA_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRB_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TICCR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TMDR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIOR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIER_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNT_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRA_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRB_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNTU_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRU_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCRU_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORU_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNTV_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRV_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCRV_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORV_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
TCNTW_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	MTU2
TGRW_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCRW_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORW_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSR_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIER_5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSTR5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNTCMPCLR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCR_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	MTU2S
TCR_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TMDR_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TMDR_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORH_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORL_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORH_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORL_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIER_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIER_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TOERS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGCRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TOCR1S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TOCR2S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNT_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNT_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCDRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TDDRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRA_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRB_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRA_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRB_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNTSS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
TCBRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	MTU2S
TGRC_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRD_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRC_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRD_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSR_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSR_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TITCRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TITCNTS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TBTERS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TDERS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TOLBRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TBTM_3S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TBTM_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TADCRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TADCORA_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TADCORB_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TADCOBRA_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TADCOBRB_4S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSYCRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TWCRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSTRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSYRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TRWERS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNTU_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRU_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCRU_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORU_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNTV_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRV_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCRV_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
TIORV_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	MTU2S
TCNTW_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TGRW_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCRW_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIORW_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSR_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TIER_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TSTR_5S	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
TCNTCMPCLRS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	A/D (Channel 0)
ADDR1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADCSR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADCR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR4	Initialized	Retained	Initialized	Initialized	Initialized	Retained	A/D (Channel 1)
ADDR5	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR6	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR7	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADCSR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADCR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR8	Initialized	Retained	Initialized	Initialized	Initialized	Retained	A/D (Channel 2)
ADDR9	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR10	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR11	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR12	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR13	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR14	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADDR15	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADCSR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ADCR_2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
FCCS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	FLASH
FPCS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
FECS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
FKEY	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
FMATS	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
FTDAR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
DTCERA	Initialized	Retained	Retained	Initialized	Retained	Retained	DTC
DTCERB	Initialized	Retained	Retained	Initialized	Retained	Retained	
DTCERC	Initialized	Retained	Retained	Initialized	Retained	Retained	
DTCERD	Initialized	Retained	Retained	Initialized	Retained	Retained	
DTCERE	Initialized	Retained	Retained	Initialized	Retained	Retained	
DTCCR	Initialized	Retained	Retained	Initialized	Retained	Retained	
DTCVBR	Initialized	Retained	Retained	Initialized	Retained	Retained	
ICCR1	Initialized	Retained	Retained	Initialized	Retained	Retained	IIC2
ICCR2	Initialized	Retained	Retained	Initialized	Retained	Retained	
ICMR	Initialized	Retained	Retained	Initialized	Retained	Retained	
ICIER	Initialized	Retained	Retained	Initialized	Retained	Retained	
ICSR	Initialized	Retained	Retained	Initialized	Retained	Retained	
SAR	Initialized	Retained	Retained	Initialized	Retained	Retained	
ICDRT	Initialized	Retained	Retained	Initialized	Retained	Retained	
ICDRR	Initialized	Retained	Retained	Initialized	Retained	Retained	
NF2CYC	Initialized	Retained	Retained	Initialized	Retained	Retained	
SSCRH	Initialized	Retained	Initialized	Initialized	Initialized	Retained	SSU
SSCRL	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SSMR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SSER	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SSSR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SSCR2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SSTDR0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SSTDR1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SSTDR2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
SSTDR3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	SSU
SSRDR0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SSRDR1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SSRDR2	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
SSRDR3	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
CMSTR	Initialized	Retained	Initialized	Initialized	Initialized	Retained	CMT
CMCSR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
CMCNT_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
CMCOR_0	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
CMCSR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
CMCNT_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
CMCOR_1	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
ICSR1	Initialized	Retained	Retained	Initialized	—	Retained	POE
OCSR1	Initialized	Retained	Retained	Initialized	—	Retained	
ICSR2	Initialized	Retained	Retained	Initialized	—	Retained	
OCSR2	Initialized	Retained	Retained	Initialized	—	Retained	
ICSR3	Initialized	Retained	Retained	Initialized	—	Retained	
SPOER	Initialized	Retained	Retained	Initialized	—	Retained	
POECR1	Initialized	Retained	Retained	Initialized	—	Retained	
POECR2	Initialized	Retained	Retained	Initialized	—	Retained	
PADRH	Initialized	Retained	Retained	Initialized	—	Retained	I/O
PADRL	Initialized	Retained	Retained	Initialized	—	Retained	
PAIORH	Initialized	Retained	Retained	Initialized	—	Retained	PFC
PAIORL	Initialized	Retained	Retained	Initialized	—	Retained	
PACRH4	Initialized	Retained	Retained	Initialized	—	Retained	
PACRH3	Initialized	Retained	Retained	Initialized	—	Retained	
PACRH2	Initialized	Retained	Retained	Initialized	—	Retained	
PACRH1	Initialized	Retained	Retained	Initialized	—	Retained	
PACRL4	Initialized	Retained	Retained	Initialized	—	Retained	
PACRL3	Initialized	Retained	Retained	Initialized	—	Retained	

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
PACRL2	Initialized	Retained	Retained	Initialized	—	Retained	PFC
PACRL1	Initialized	Retained	Retained	Initialized	—	Retained	
PAPRH	Initialized	Retained	Retained	Initialized	—	Retained	I/O
PAPRL	Initialized	Retained	Retained	Initialized	—	Retained	
PBDRL	Initialized	Retained	Retained	Initialized	—	Retained	
PBIORL	Initialized	Retained	Retained	Initialized	—	Retained	PFC
PBCRL3	Initialized	Retained	Retained	Initialized	—	Retained	
PBCRL2	Initialized	Retained	Retained	Initialized	—	Retained	
PBCRL1	Initialized	Retained	Retained	Initialized	—	Retained	
PBPRL	Initialized	Retained	Retained	Initialized	—	Retained	I/O
PCDRH	Initialized	Retained	Retained	Initialized	—	Retained	
PCDRL	Initialized	Retained	Retained	Initialized	—	Retained	
PCIORH	Initialized	Retained	Retained	Initialized	—	Retained	PFC
PCIORL	Initialized	Retained	Retained	Initialized	—	Retained	
PCCRH3	Initialized	Retained	Retained	Initialized	—	Retained	
PCCRH2	Initialized	Retained	Retained	Initialized	—	Retained	
PCCRH1	Initialized	Retained	Retained	Initialized	—	Retained	
PCCRL4	Initialized	Retained	Retained	Initialized	—	Retained	
PCCRL3	Initialized	Retained	Retained	Initialized	—	Retained	
PCCRL2	Initialized	Retained	Retained	Initialized	—	Retained	
PCCRL1	Initialized	Retained	Retained	Initialized	—	Retained	
PCPRH	Initialized	Retained	Retained	Initialized	—	Retained	I/O
PCPRL	Initialized	Retained	Retained	Initialized	—	Retained	
PDDRH	Initialized	Retained	Retained	Initialized	—	Retained	
PDDRL	Initialized	Retained	Retained	Initialized	—	Retained	
PDIORH	Initialized	Retained	Retained	Initialized	—	Retained	PFC
PDIORL	Initialized	Retained	Retained	Initialized	—	Retained	
PDCRH4	Initialized	Retained	Retained	Initialized	—	Retained	
PDCRH3	Initialized	Retained	Retained	Initialized	—	Retained	
PDCRH2	Initialized	Retained	Retained	Initialized	—	Retained	
PDCRH1	Initialized	Retained	Retained	Initialized	—	Retained	

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
PDCRL4	Initialized	Retained	Retained	Initialized	—	Retained	PFC
PDCRL3	Initialized	Retained	Retained	Initialized	—	Retained	
PDCRL2	Initialized	Retained	Retained	Initialized	—	Retained	
PDCRL1	Initialized	Retained	Retained	Initialized	—	Retained	
PDPRL	Initialized	Retained	Retained	Initialized	—	Retained	I/O
PEDRH	Initialized	Retained	Retained	Initialized	—	Retained	
PEDRL	Initialized	Retained	Retained	Initialized	—	Retained	
PEIORH	Initialized	Retained	Retained	Initialized	—	Retained	PFC
PEIORL	Initialized	Retained	Retained	Initialized	—	Retained	
PECRH2	Initialized	Retained	Retained	Initialized	—	Retained	
PECRH1	Initialized	Retained	Retained	Initialized	—	Retained	
PECRL4	Initialized	Retained	Retained	Initialized	—	Retained	
PECRL3	Initialized	Retained	Retained	Initialized	—	Retained	
PECRL2	Initialized	Retained	Retained	Initialized	—	Retained	
PECRL1	Initialized	Retained	Retained	Initialized	—	Retained	
PEPRH	Initialized	Retained	Retained	Initialized	—	Retained	I/O
PEPRL	Initialized	Retained	Retained	Initialized	—	Retained	
HCPCR	Initialized	Retained	Retained	Initialized	—	Retained	PFC
IFCR	Initialized	Retained	Retained	Initialized	—	Retained	
PFDRL	Initialized	Retained	Retained	Initialized	—	Retained	I/O
FRQCR	Initialized*1	Retained	Retained	Initialized	—	Retained	CPG
STBCR1	Initialized	Retained	Retained	Initialized	—	Retained	Power-down modes
STBCR2	Initialized	Retained	Retained	Initialized	—	Retained	
STBCR3	Initialized	Retained	Retained	Initialized	—	Retained	
STBCR4	Initialized	Retained	Retained	Initialized	—	Retained	
STBCR5	Initialized	Retained	Retained	Initialized	—	Retained	
STBCR6	Initialized	Retained	Retained	Initialized	—	Retained	
WTCNT	Initialized*1	Retained	Retained	Initialized	—	Retained	WDT
WTCSR	Initialized*1	Retained	Retained	Initialized	—	Retained	
OSCCR	Initialized*2	Retained	Retained*3	Initialized	—	Retained	CPG

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
RAMCR	Initialized	Retained	Retained	Initialized	—	Retained	Power-down modes
ADTSR_0	Initialized	Retained	Retained	Initialized	Retained	Retained	A/D
ADTSR_1	Initialized	Retained	Retained	Initialized	Retained	Retained	
BSCEHR	Initialized	Retained	Retained	Initialized	—	Retained	BSC
ICR0	Initialized	Initialized	Retained	Initialized	—	Retained	INTC
IRQCR	Initialized	Initialized	Retained	Initialized	—	Retained	
IRQSR	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRA	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRB	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRC	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRD	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRE	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRF	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRH	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRI	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRJ	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRK	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRL	Initialized	Initialized	Retained	Initialized	—	Retained	
IPRM	Initialized	Initialized	Retained	Initialized	—	Retained	
SAR_0	Initialized	Retained	Retained	Initialized	Retained	Retained	DMAC
DAR_0	Initialized	Retained	Retained	Initialized	Retained	Retained	
DMATCR_0	Initialized	Retained	Retained	Initialized	Retained	Retained	
CHCR_0	Initialized	Retained	Retained	Initialized	Retained	Retained	
SAR_1	Initialized	Retained	Retained	Initialized	Retained	Retained	
DAR_1	Initialized	Retained	Retained	Initialized	Retained	Retained	
DMATCR_1	Initialized	Retained	Retained	Initialized	Retained	Retained	
CHCR_1	Initialized	Retained	Retained	Initialized	Retained	Retained	
SAR_2	Initialized	Retained	Retained	Initialized	Retained	Retained	
DAR_2	Initialized	Retained	Retained	Initialized	Retained	Retained	
DMATCR_2	Initialized	Retained	Retained	Initialized	Retained	Retained	

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
CHCR_2	Initialized	Retained	Retained	Initialized	Retained	Retained	DMAC
SAR_3	Initialized	Retained	Retained	Initialized	Retained	Retained	
DAR_3	Initialized	Retained	Retained	Initialized	Retained	Retained	
DMATCR_3	Initialized	Retained	Retained	Initialized	Retained	Retained	
CHCR_3	Initialized	Retained	Retained	Initialized	Retained	Retained	
DMAOR	Initialized	Retained	Retained	Initialized	Retained	Retained	
CMNCR	Initialized	Retained	Retained	Initialized	—	Retained	BSC
CS0BCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS1BCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS2BCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS3BCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS4BCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS5BCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS6BCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS7BCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS8BCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS0WCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS1WCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS2WCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS3WCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS4WCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS5WCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS6WCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS7WCR	Initialized	Retained	Retained	Initialized	—	Retained	
CS8WCR	Initialized	Retained	Retained	Initialized	—	Retained	
SDCR	Initialized	Retained	Retained	Initialized	—	Retained	
RTCSR	Initialized	Retained	Retained	Initialized	—	Retained	
RTCNT	Initialized	Retained	Retained	Initialized	—	Retained	
RTCOR	Initialized	Retained	Retained	Initialized	—	Retained	
RAMER	Initialized	Initialized	Retained	Initialized	Retained	Retained	FLASH

Register Abbreviation	Power-on reset	Manual reset	Software Standby	Deep Software Standby	Module Standby	Sleep	Module
BARA	Initialized	Retained	Retained	Initialized	Initialized	Retained	UBC
BAMRA	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BBRA	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BDRA*4	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BDMRA*4	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BARB	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BAMRB	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BBRB	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BDRB*4	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BDMRB*4	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BRCR	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BRSR*4	Initialized	Initialized	Retained	Initialized	Initialized	Retained	
BRDR*4	Initialized	Initialized	Retained	Initialized	Initialized	Retained	
BETR*4	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Notes: 1. Not initialized by a WDT power-on reset.
2. The OSCSTOP bit is not initialized by a WDT power-on reset.
3. The OSCSTOP bit is initialized.
4. Only in F-ZTAT version.

Section 28 Electrical Characteristics

28.1 Absolute Maximum Ratings

Table 28.1 lists the absolute maximum ratings.

Table 28.1 Absolute Maximum Ratings

Item		Symbol	Value	Unit
Power supply voltage		V_{CC}	–0.3 to +7.0	V
Input voltage (except analog input pins and SCL and SDA pins)		V_{in}	–0.3 to V_{CC} +0.3	V
Input voltage (SCL and SDA pins)		V_{in}	–0.3 to +7.0	V
Analog power supply voltage		AV_{CC}	–0.3 to +7.0	V
Analog reference voltage		AV_{ref}	–0.3 to AV_{CC} +0.3	V
Analog input voltage		V_{an}	–0.3 to AV_{CC} +0.3	V
Operating temperature	Consumer applications	T_{opr}	–20 to +85	°C
	Industrial applications		–40 to +85	°C
Storage temperature		T_{stg}	–55 to +125	°C

[Operating Precautions]

Operating the LSI in excess of the absolute maximum ratings may result in permanent damage.

28.2 DC Characteristics

Tables 28.2 and 28.3 list DC characteristics.

Table 28.2 DC Characteristics

Conditions: V_{CC} = 3.0 V to 3.6 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC},
V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications),
T_a = –40°C to +85°C (industrial applications)

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input high-level voltage (except Schmitt trigger input voltage)	$\overline{RES}$, $\overline{MRES}$, NMI, FWE, MD1, MD0, $\overline{ASEMD0}$, EXTAL	V_{IH}	V_{CC}–0.5	—	V_{CC}+0.3	V	
	Analog ports		2.2	—	AV_{CC}+0.3	V	
	Other input pins		2.2	—	V_{CC}+0.3	V	
Input low-level voltage (except Schmitt trigger input voltage)	$\overline{RES}$, $\overline{MRES}$, NMI, FWE, MD1, MD0, $\overline{ASEMD0}$, EXTAL	V_{IL}	–0.3	—	0.5	V	
	Other input pins		–0.3	—	0.8	V	
Schmitt trigger input voltage	IRQ7 to IRQ0, $\overline{POE8}$ to $\overline{POE0}$, TCLKA to TCLKD, TIOC0A to TIOC0D, TIOC1A, TIOC1B, TIOC2A, TIOC2B, TIOC3A to TIOC3D, TIOC4A to TIOC4D, TIC5U, TIC5V, TIC5W, TIOC3AS to TIOC3DS, TIOC4AS to TIOC4DS, TIC5US, TIC5VS, TIC5WS, SCK0 to SCK3, RXD0 to RXD3, $\overline{CTS3}$, SSCK, $\overline{SCS}$, SSI, SSO, SCL, SDA	V_{T+}	V_{CC}–0.5	—	—	V	
		V_{T-}	—	—	0.5	V	
		V_{T+}–V_{T-}	0.2	—	—	V	
Input leak current	All input pins (except $\overline{ASEMD0}$)	$\lvert I_{in} \rvert$	—	—	1.0	µA	

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input pull-up MOS current	$\overline{\text{ASEMD0}}$	$-I_{pu}$	—	—	350	µA	V_{in} = 0 V
Three-state leak current (OFF state)	Ports A, B, C, D, E	$\lvert I_{tsi} \rvert$	—	—	1.0	µA	
Output high-level voltage	All output pins (expect PB2, PB3 in SH7084/ SH7085/SH7086)	V_{OH}	V_{CC}–0.5	—	—	V	I_{OH} = –200 µA
			V_{CC}–1.0	—	—	V	I_{OH} = –1 mA
	PB2, PB3 (only SH7084/SH7085/ SH7086)		1.0	—	—	V	I_{OH} = –200 µA
	TIOC3B, TIOC3D, TIOC4A to TIOC4D, TIOC3BS, TIOC3DS, TIOC4AS to TIOC4DS		V_{CC}–1.0	—	—	V	I_{OH} = –5 mA
	PD9, PD11 to PD15, PD24 to PD29, PE9, PE11 to PE21		V_{CC}–2.0	—	—	V	I_{OH} = –5 mA
Output low-level voltage	All output pins	V_{OL}	—	—	0.4	V	I_{OL} = 1.6 mA
	SCL, SDA		—	—	0.4	V	I_{OL} = 3 mA
			—	—	0.5	V	I_{OL} = 8 mA
	TIOC3B, TIOC3D, TIOC4A to TIOC4D, TIOC3BS, TIOC3DS, TIOC4AS to TIOC4DS		—	—	0.9	V	I_{OL} = 15 mA
	PD9, PD11 to PD15, PD24 to PD29, PE9, PE11 to PE21		—	—	2.0	V	I_{OL} = 15 mA
Input capacitance	All input pins	C_{in}	—	—	20	pF	V_{in} = 0 V f = 1 MHz Ta = 25°C
Supply current	Normal operation	I_{CC}	—	100 (150)*	135 (165)*	mA	Iϕ = 80 MHz Bϕ = 40 MHz Pϕ = 40 MHz MPϕ = 40 MHz MIϕ = 80 MHz

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Supply current	Sleep	I_{CC}	—	65 (140)*	110 (150)*	mA	Bϕ = 40 MHz Pϕ = 40 MHz MPϕ = 40 MHz MIϕ = 80 MHz
	Software standby		—	10 (20)*	40 (60)*	mA	$T_a \leq 50°C$
			—	—	80 (120)*	mA	$50°C < T_a$
	Deep software standby		—	5 (20)*	30 (50)*	µA	$T_a \leq 50°C$
			—	—	80 (120)*	µA	$50°C < T_a$
Analog power supply current (except SH7084)	During A/D conversion	AI_{CC}	—	2	3.5	mA	The value per A/D converter module.
	Waiting for A/D conversion		—	—	1	mA	
	Standby		—	—	10	µA	
Reference power supply current (except SH7084)	During A/D conversion	AI_{ref}	—	—	2.5	mA	The value per A/D converter module.
	Waiting for A/D conversion		—	—	2.5	mA	
	Standby		—	—	10	µA	
Analog power supply current (SH7084)	During A/D conversion	AI_{CC}	—	3	6	mA	The value per A/D converter module.
	Waiting for A/D conversion		—	—	3.5	mA	
	Standby		—	—	10	µA	
RAM standby voltage		VRAM	2.0	—	—	V	V_{CC}

[Operating Precautions]

1. When the A/D converter is not used, do not leave the AV_{CC}, AV_{SS} and AV_{ref} pins open.
2. The supply current was measured when V_{IH} (Min.) = V_{CC} – 0.5 V, V_{IL} (Max.) = 0.5 V, with all output pins unloaded.

Note: * F-ZTAT version supporting full functions of E10A.

Table 28.3 DC Characteristics

Conditions: V_{CC} = 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC},
V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications),
T_a = –40°C to +85°C (industrial applications)

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input high-level voltage (except Schmitt trigger input voltage)	$\overline{RES}$, $\overline{MRES}$, NMI, FWE, MD1, MD0, $\overline{ASEMD0}$, EXTAL	V_{IH}	V_{CC}–0.7	—	V_{CC}+0.3	V	
	Analog ports		2.2	—	AV_{CC}+0.3	V	
	Other input pins		2.2	—	V_{CC}+0.3	V	
Input low-level voltage (except Schmitt trigger input voltage)	$\overline{RES}$, $\overline{MRES}$, NMI, FWE, MD1, MD0, $\overline{ASEMD0}$, EXTAL	V_{IL}	–0.3	—	0.5	V	
	Other input pins		–0.3	—	0.8	V	
Schmitt trigger input voltage	IRQ7 to IRQ0, $\overline{POE8}$ to $\overline{POE0}$, TCLKA to TCLKD, TIOC0A to TIOC0D, TIOC1A, TIOC1B, TIOC2A, TIOC2B, TIOC3A to TIOC3D, TIOC4A to TIOC4D, TIC5U, TIC5V, TIC5W, TIOC3AS to TIOC3DS, TIOC4AS to TIOC4DS, TIC5US, TIC5VS, TIC5WS, SCK0 to SCK3, RXD0 to RXD3, $\overline{CTS3}$, SSCK, $\overline{SCS}$, SSI, SSO, SCL, SDA	V_{T+}	V_{CC}–0.5	—	—	V	
		V_{T-}	—	—	1.0	V	
		V_{T+}–V_{T-}	0.4	—	—	V	
Input leak current	All input pins (except $\overline{ASEMD0}$)	$\lvert I_{in} \rvert$	—	—	1.0	µA	
Input pull-up MOS current	$\overline{ASEMD0}$	$-I_{pu}$	—	—	800	µA	V_{in} = 0 V

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Three-state leak current (OFF state)	Ports A, B, C, D, E	$\lvert I_{tsi} \rvert$	—	—	1.0	µA	
Output high-level voltage	All output pins (expect PB2, PB3 in SH7084/ SH7085/SH7086)	V_{OH}	V_{CC}–0.5	—	—	V	I_{OH} = –200 µA
			V_{CC}–1.0	—	—	V	I_{OH} = –1 mA
	PB2, PB3 (only SH7084/SH7085/ SH7086)		1.0	—	—	V	I_{OH} = –200 µA
	TIOC3B, TIOC3D, TIOC4A to TIOC4D, TIOC3BS, TIOC3DS, TIOC4AS to TIOC4DS		V_{CC}–1.0	—	—	V	I_{OH} = –5 mA
	PD9, PD11 to PD15, PD24 to PD29, PE9, PE11 to PE21		V_{CC}–2.0	—	—	V	I_{OH} = –5 mA
Output low-level voltage	All output pins	V_{OL}	—	—	0.4	V	I_{OL} = 1.6 mA
	SCL, SDA		—	—	0.4	V	I_{OL} = 3 mA
			—	—	0.5	V	I_{OL} = 8 mA
	TIOC3B, TIOC3D, TIOC4A to TIOC4D, TIOC3BS, TIOC3DS, TIOC4AS to TIOC4DS		—	—	1.4	V	I_{OL} = 15 mA
	PD9, PD11 to PD15, PD24 to PD29, PE9, PE11 to PE21		—	—	1.5	V	I_{OL} = 15 mA
Input capacitance	All input pins	C_{in}	—	—	20	pF	V_{in} = 0 V f = 1 MHz Ta = 25°C
Supply current	Normal operation	I_{CC}	—	100 (150)*	135 (165)*	mA	Iϕ = 80 MHz Bϕ = 40 MHz Pϕ = 40 MHz MPϕ = 40 MHz MIϕ = 80 MHz

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Supply current	Sleep	I_{CC}	—	65 (140)*	110 (150)*	mA	Bϕ = 40 MHz Pϕ = 40 MHz MPϕ = 40 MHz MIϕ = 80 MHz
	Software standby		—	10 (20)*	40 (60)*	mA	$T_a \leq 50°C$
			—	—	80 (120)*	mA	$50°C < T_a$
	Deep software standby		—	5 (20)*	30 (50)*	µA	$T_a \leq 50°C$
			—	—	80 (120)*	µA	$50°C < T_a$
Analog power supply current (except SH7084)	During A/D conversion	AI_{CC}	—	2	3.5	mA	The value per A/D converter module.
	Waiting for A/D conversion		—	—	1	mA	
	Standby		—	—	10	µA	
Reference power supply current (except SH7084)	During A/D conversion	AI_{ref}	—	—	2.5	mA	The value per A/D converter module.
	Waiting for A/D conversion		—	—	2.5	mA	
	Standby		—	—	10	µA	
Analog power supply current (SH7084)	During A/D conversion	AI_{CC}	—	3	6	mA	The value per A/D converter module.
	Waiting for A/D conversion		—	—	3.5	mA	
	Standby		—	—	10	µA	
RAM standby voltage		VRAM	2.0	—	—	V	V_{CC}

[Operating Precautions]

1. When the A/D converter is not used, do not leave the AV_{CC}, AV_{SS} and AV_{ref} pins open.
2. The supply current was measured when V_{IH} (Min.) = V_{CC} – 0.5 V, V_{IL} (Max.) = 0.5 V, with all output pins unloaded.

Note: * F-ZTAT version supporting full functions of E10A.

Table 28.4 Permitted Output Current Values

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Typ.	Max.	Unit
Output low-level permissible current (per pin)	I_{OL}	—	—	2.0*	mA
Output low-level permissible current (total)	ΣI_{OL}	—	—	80	mA
Output high-level permissible current (per pin)	$-I_{OH}$	—	—	2.0*	mA
Output high-level permissible current (total)	$\Sigma -I_{OH}$	—	—	25	mA

[Operating Precautions]

To assure LSI reliability, do not exceed the output values listed in table 28.4.

Note: * I_{OL} = 15 mA (Max.)/$-I_{OH}$ = 5 mA (Max.) about pins PD9, PD11 to PD15, PD24 to PD29, PE9, and PE11 to PE21. I_{OL} = 8 mA (Max.) about pins SCL and SDA. However, at most three pins are permitted to have simultaneously $I_{OL}/-I_{OH} > 2.0$ mA among these pins.

28.3 AC Characteristics

Signals input to this LSI are basically handled as signals in synchronization with a clock. The setup and hold times for input pins must be followed.

Table 28.5 Maximum Operating Frequency

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
Operating frequency	CPU (Iϕ)	f	10	—	80	MHz	
	External bus (Bϕ)		10	—	40		
	Peripheral module (Pϕ)		10	—	40		
	MTU2 (MPϕ)		10	—	40		
	MTU2S (MIϕ)		10	—	80		

28.3.1 Clock Timing

Table 28.6 Clock Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Max.	Unit	Reference Figure
EXTAL clock input frequency	f_{EX}	5	12.5	MHz	Figure 28.1
EXTAL clock input cycle time	t_{EXcyc}	80	200	ns	
EXTAL clock input low pulse width	t_{EXL}	20	—	ns	
EXTAL clock input high pulse width	t_{EXH}	20	—	ns	
EXTAL clock input rise time	t_{EXr}	—	5	ns	
EXTAL clock input fall time	t_{EXf}	—	5	ns	
CK clock output frequency	f_{OP}	10	40	MHz	Figure 28.2
CK clock output cycle time	t_{cyc}	25	100	ns	
CK clock output low pulse width	t_{CKL}	$1/2t_{cyc}$–7.5	—	ns	
CK clock output high pulse width	t_{CKH}	$1/2t_{cyc}$–7.5	—	ns	
CK clock output rise time	t_{CKr}	—	5	ns	
CK clock output fall time	t_{CKf}	—	5	ns	
Power-on oscillation settling time	t_{OSC1}	10	—	ms	Figure 28.3
Standby return oscillation settling time 1	t_{OSC2}	10	—	ms	Figure 28.4
Standby return oscillation settling time 2	t_{OSC3}	10	—	ms	Figure 28.5

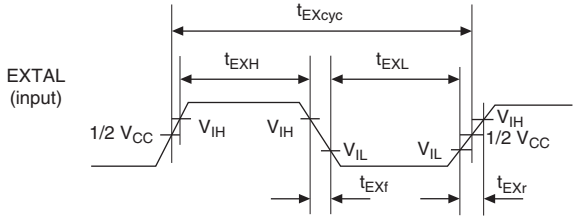


Figure 28.1 EXTAL Clock Input Timing

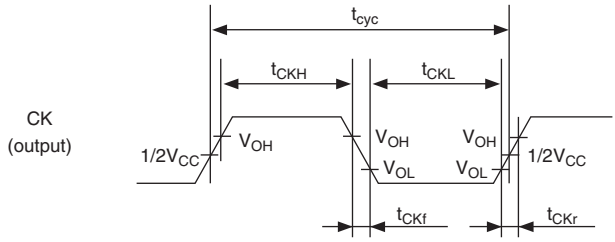
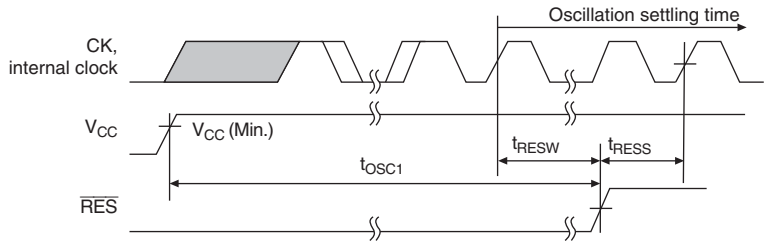
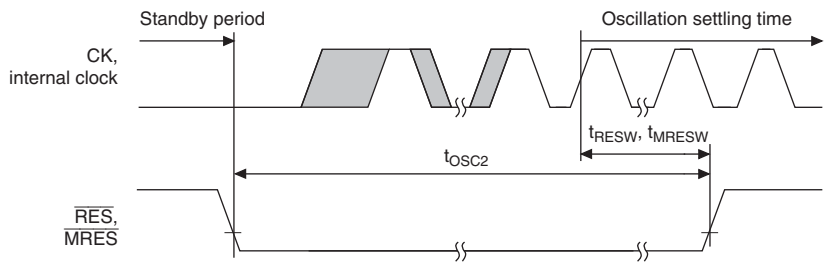


Figure 28.2 CK Clock Output Timing



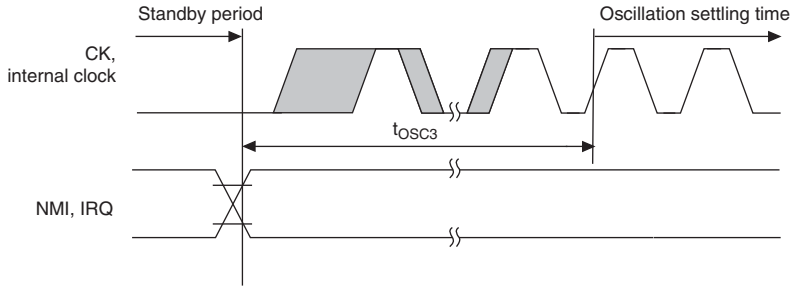
Note: Oscillation settling time when on-chip oscillator is used.

Figure 28.3 Power-On Oscillation Settling Timing



Note: Oscillation settling time when on-chip oscillator is used.

Figure 28.4 Oscillation Settling Timing on Return from Standby (Return by Reset)



Note: Oscillation settling time when on-chip oscillator is used.

Figure 28.5 Oscillation Settling Timing on Return from Standby (Return by NMI or IRQ)

28.3.2 Control Signal Timing

Table 28.7 Control Signal Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Max.	Unit	Reference Figure
$\overline{RES}$ pulse width	t_{RESW}	20*[2]	—	t_{Bcyc}*[4]	Figures 28.3, 28.4,
$\overline{RES}$ setup time*[1]	t_{RESS}	65	—	ns	28.6, 28.7
$\overline{RES}$ hold time	t_{RESH}	15	—	ns	
$\overline{MRES}$ pulse width	t_{MRESW}	20*[3]	—	t_{Bcyc}*[4]	
$\overline{MRES}$ setup time*[1]	t_{MRESS}	25	—	ns	
$\overline{MRES}$ hold time	t_{MRESH}	15	—	ns	
MD1, MD0, FWE setup time	t_{MDS}	20	—	t_{Bcyc}*[4]	Figure 28.6
$\overline{BREQ}$ setup time	t_{BREQS}	$1/2t_{Bcyc}$ + 15	—	ns	Figure 28.9
$\overline{BREQ}$ hold time	t_{BREQH}	$1/2t_{Bcyc}$ + 10	—	ns	
NMI setup time*[1]	t_{NMIS}	60	—	ns	Figure 28.7
NMI hold time	t_{NMIH}	10	—	ns	
IRQ7 to IRQ0 setup time*[1]	t_{IRQS}	35	—	ns	
IRQ7 to IRQ0 hold time	t_{IRQH}	35	—	ns	
$\overline{IRQOUT}$ output delay time	t_{IRQOD}	—	100	ns	Figure 28.8
$\overline{BACK}$ delay time	t_{BACKD}	—	$1/2t_{Bcyc}$ + 20	ns	Figures 28.9, 28.10
Bus tri-state delay time	t_{BOFF}	0	100	ns	
Bus buffer on time	t_{BON}	0	100	ns	

Notes: 1. The $\overline{RES}$, $\overline{MRES}$, NMI, $\overline{BREQ}$, and IRQ7 to IRQ0 signals are asynchronous signals. When the setup time is satisfied, change of signal level is detected at the rising edge of the clock. If not, the detection is delayed until the next rising edge of the clock.

2. In standby mode, t_{RESW} = t_{OSC2} (10 ms).
3. In standby mode, t_{MRESW} = t_{OSC2} (10 ms).
4. t_{Bcyc} indicates external bus clock cycle time (Bϕ = CK).

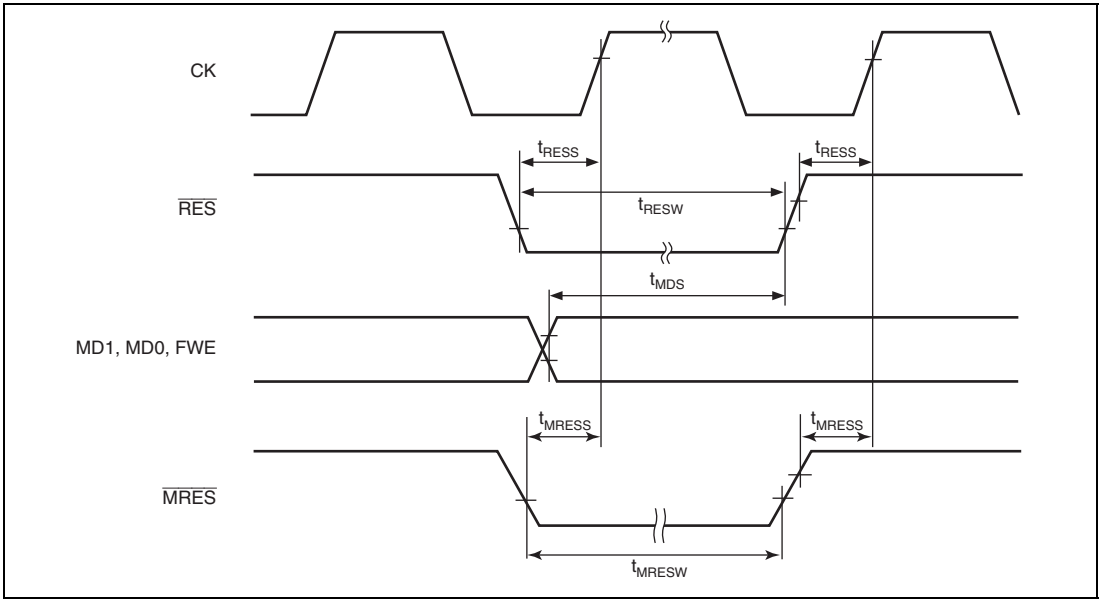


Figure 28.6 Reset Input Timing

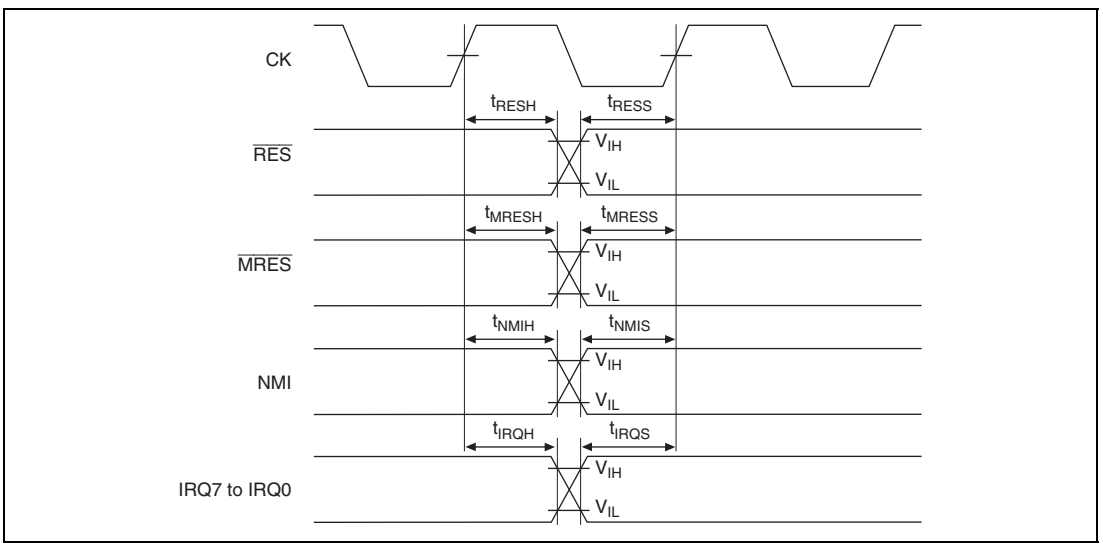


Figure 28.7 Interrupt Signal Input Timing

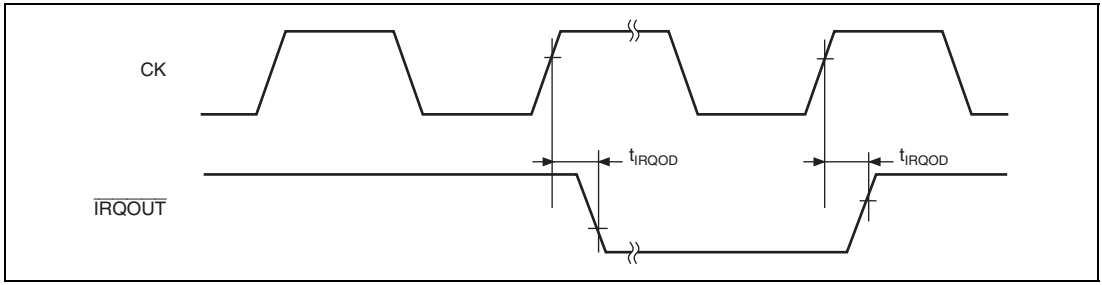


Figure 28.8 Interrupt Signal Output Timing

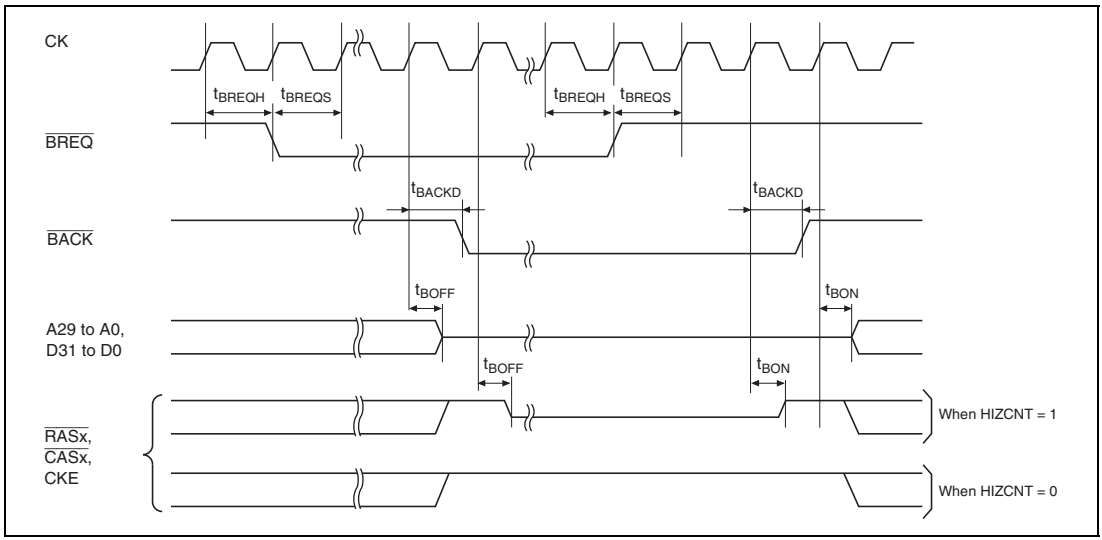


Figure 28.9 Bus Release Timing

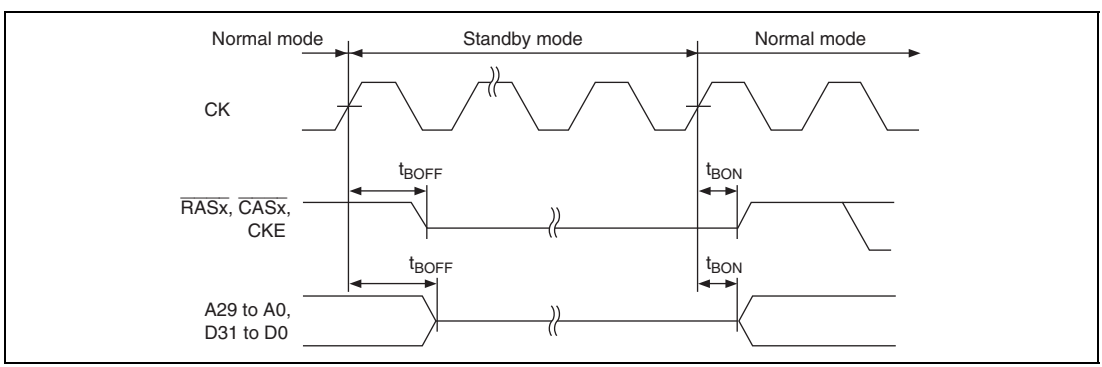


Figure 28.10 Pin Driving Timing in Standby Mode

28.3.3 AC Bus Timing

Table 28.8 Bus Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

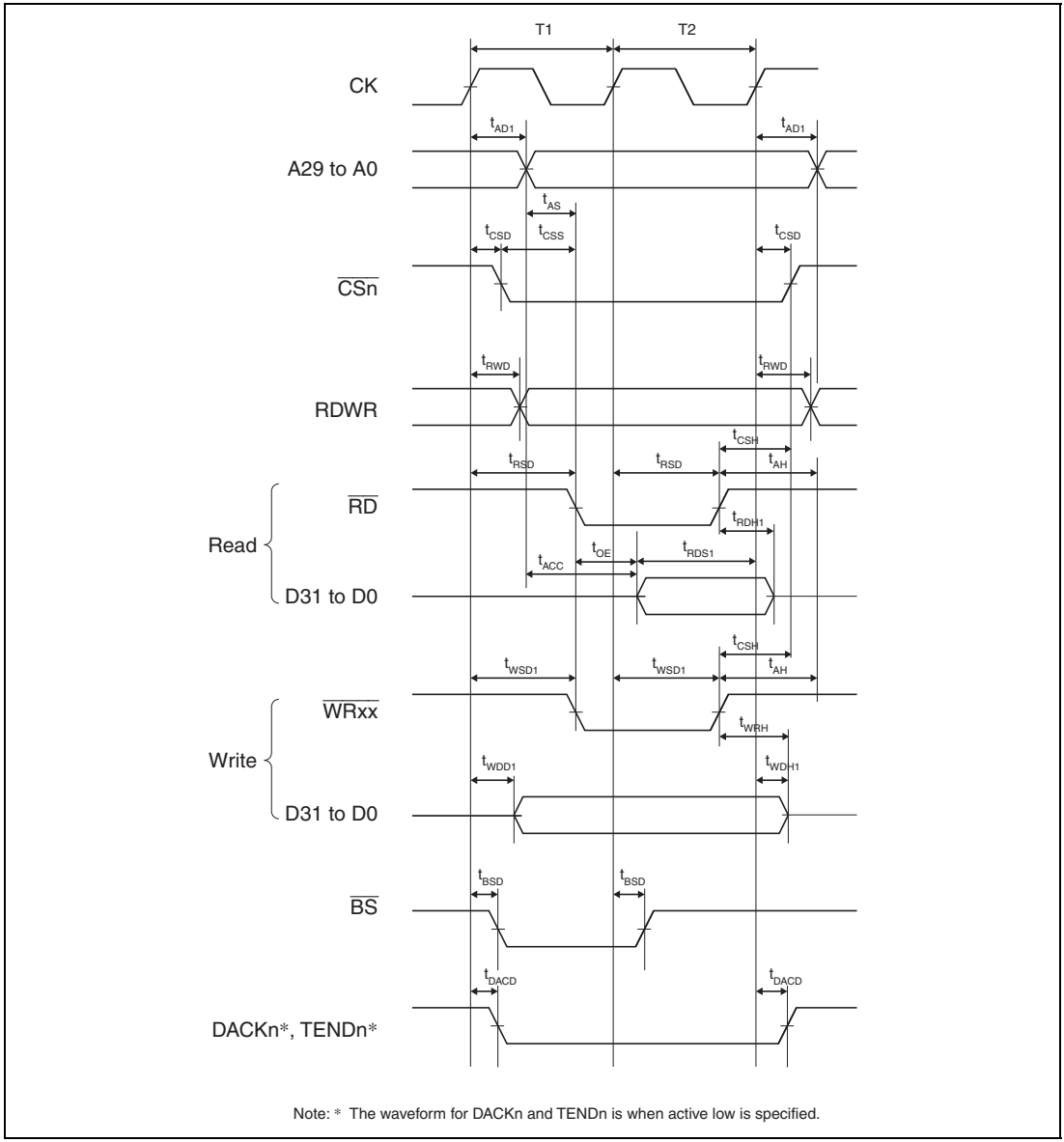
Item	Symbol	Min.	Max.	Unit	Reference Figure
Address delay time 1	t_{AD1}	1	18	ns	Figures 28.11 to 28.44
Address delay time 2	t_{AD2}	$1/2t_{Bcyc}$ + 1	$1/2t_{Bcyc}$ +18	ns	Figure 28.23
Address setup time	t_{AS}	0	—	ns	Figures 28.11 to 28.14, 28.18
Address hold time	t_{AH}	0	—	ns	Figures 28.11 to 28.14, 28.18
$\overline{BS}$ delay time	t_{BSD}	—	18	ns	Figures 28.11 to 28.37, 28.41 to 28.44
$\overline{CS}$ delay time	t_{CSD}	1	18	ns	Figures 28.11 to 28.44
$\overline{CS}$ setup time	t_{CSS}	0	—	ns	Figures 28.11 to 28.14
$\overline{CS}$ hold time	t_{CSH}	0	—	ns	Figures 28.11 to 28.14
Read write delay time	t_{RWD}	1	18	ns	Figures 28.11 to 28.44
Read strobe delay time	t_{RSD}	$1/2t_{Bcyc}$ + 1	$1/2t_{Bcyc}$ + 18	ns	Figures 28.11 to 28.18, 28.23, 28.41, 28.42
Read data setup time 1	t_{RDS1}	$1/2t_{Bcyc}$ + 18	—	ns	Figures 28.11 to 28.18, 28.41 to 28.44
Read data setup time 2	t_{RDS2}	19	—	ns	Figures 28.20 to 28.22, 28.24 to 28.27, 28.32 to 28.34
Read data setup time 3	t_{RDS3}	$1/2t_{Bcyc}$ + 18	—	ns	Figure 28.23

Item	Symbol	Min.	Max.	Unit	Reference Figure
Read data hold time 1	t_{RDH1}	0	—	ns	Figures 28.11 to 28.18, 28.41 to 28.44
Read data hold time 2	t_{RDH2}	2	—	ns	Figures 28.20 to 28.22, 28.24 to 28.27, 28.32 to 28.34
Read data hold time 3	t_{RDH3}	0	—	ns	Figure 28.23
Read data access time	t_{ACC}*2	$t_{Bcyc} \times (n + 1.5) - 33$*1	—	ns	Figures 28.11 to 28.17
Access time from read strobe	t_{OE}*2	$t_{Bcyc} \times (n + 1) - 31$*1	—	ns	Figures 28.11 to 28.17
Write strobe delay time 1	t_{WSD1}	$1/2t_{Bcyc} + 1$	$1/2t_{Bcyc} + 18$	ns	Figures 28.11 to 28.16, 28.41, 28.42
Write strobe delay time 2	t_{WSD2}	—	18	ns	Figure 28.17
Write data delay time 1	t_{WDD1}	—	18	ns	Figures 28.11 to 28.22, 28.41 to 28.44
Write data delay time 2	t_{WDD2}	—	18	ns	Figures 28.28 to 28.31, 28.35 to 28.37
Write data hold time 1	t_{WDH1}	1	11	ns	Figures 28.11 to 28.22, 28.41 to 28.44
Write data hold time 2	t_{WDH2}	1	—	ns	Figures 28.28 to 28.31, 28.35 to 28.37
Write data hold time	t_{WRH}	0	—	ns	Figures 28.11 to 28.14, 28.18
$\overline{WAIT}$ setup time	t_{WTS}	$1/2t_{Bcyc} + 17$	—	ns	Figures 28.12 to 28.23, 28.42, 28.44
$\overline{WAIT}$ hold time	t_{WTH}	$1/2t_{Bcyc} + 7$	—	ns	Figures 28.12 to 28.23, 28.42, 28.44
$\overline{RAS}$ delay time	t_{RASD}	1	18	ns	Figures 28.24 to 28.35, 28.37 to 28.40

Item	Symbol	Min.	Max.	Unit	Reference Figure
$\overline{CAS}$ delay time	t_{CASD}	1	18	ns	Figures 28.24 to 28.40
DQM delay time	t_{DQMD}	1	18	ns	Figures 28.24 to 28.37
CKE delay time	t_{CKED}	1	18	ns	Figure 28.39
$\overline{AH}$ delay time	t_{AHD}	$1/2t_{Bcyc} + 1$	$1/2t_{Bcyc} + 18$	ns	Figure 28.18
Multiplexed address delay time	t_{MAD}	—	18	ns	Figure 28.18
Multiplexed address hold time	t_{MAH}	1	—	ns	Figure 28.18
DACK, TEND delay time	t_{DACD}	1	18	ns	Figures 28.11 to 28.35
$\overline{FRAME}$ delay time	t_{FMD}	1	18	ns	Figure 28.19 to 28.22
$\overline{ICIORD}$ delay time	t_{ICRSD}	$1/2t_{Bcyc} + 1$	$1/2t_{Bcyc} + 18$	ns	Figures 28.43, 28.44
$\overline{ICIOWR}$ delay time	t_{ICWSD}	$1/2t_{Bcyc} + 1$	$1/2t_{Bcyc} + 18$	ns	Figures 28.43, 28.44
$\overline{IOIS16}$ setup time	t_{IO16S}	$1/2t_{Bcyc} + 13$	—	ns	Figure 28.44
$\overline{IOIS16}$ hold time	t_{IO16H}	$1/2t_{Bcyc} + 10$	—	ns	Figure 28.44

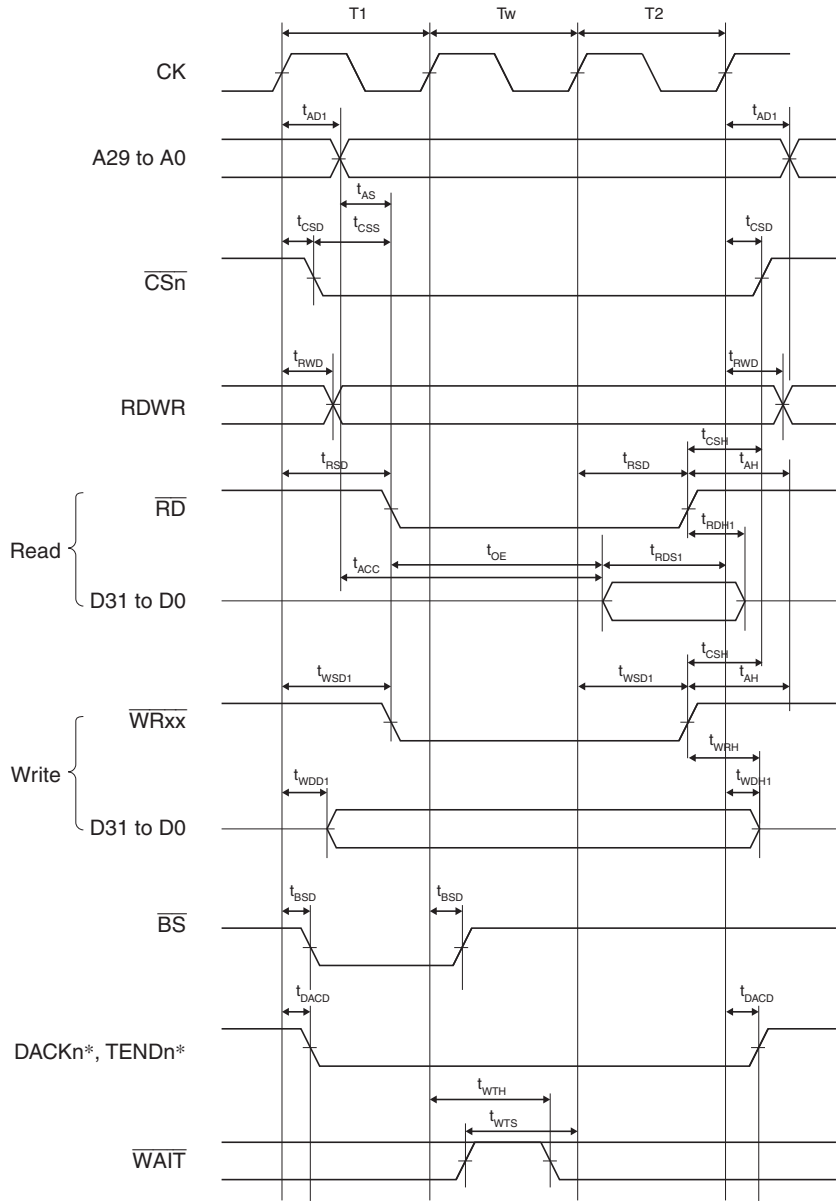
Notes: t_{Bcyc} indicates external bus clock period (Bϕ = CK).

1. n denotes the number of wait cycles.
2. If the access time conditions are satisfied, the t_{RDS1} condition does not need to be satisfied.



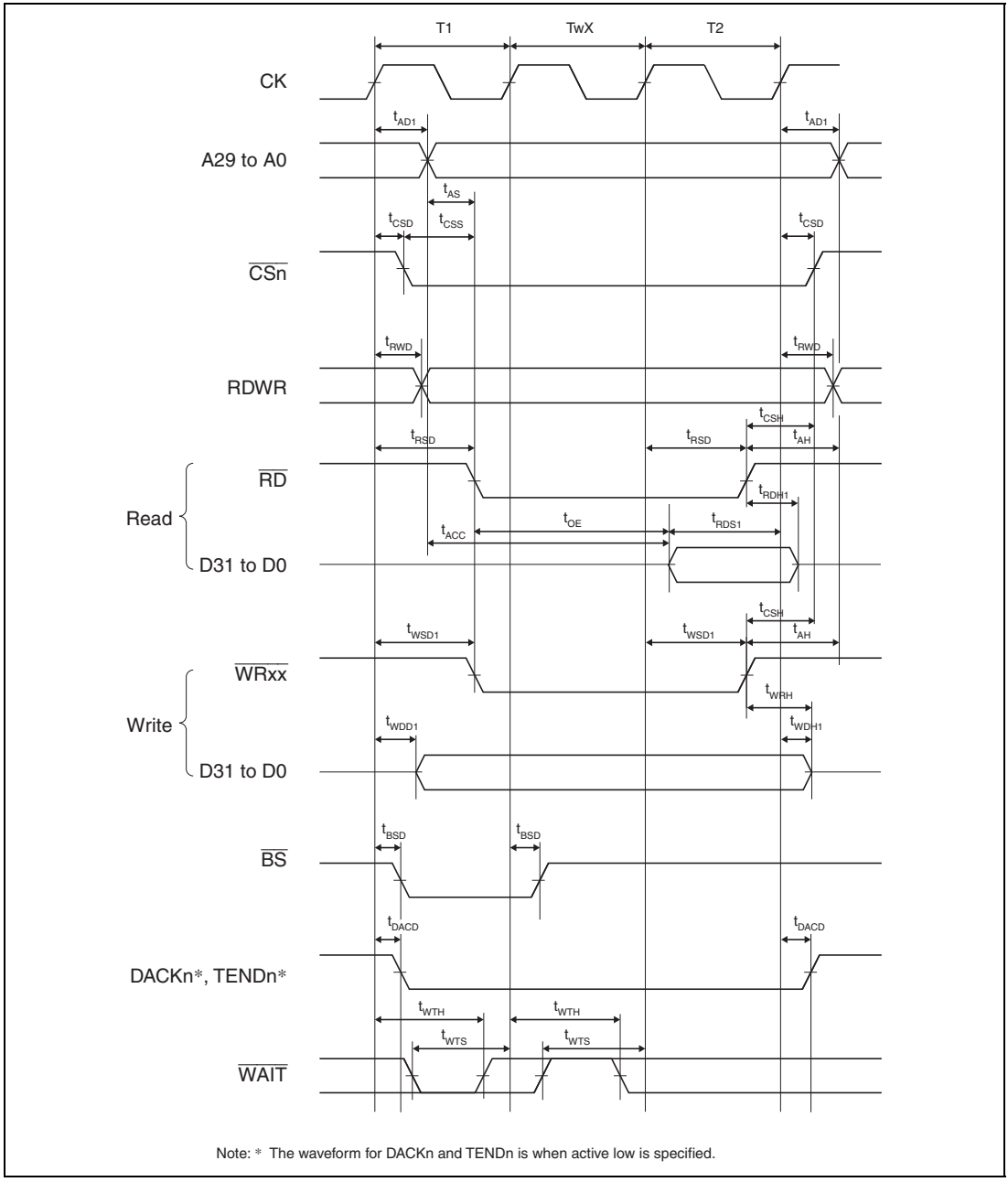
Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 28.11 Basic Bus Timing for Normal Space (No Wait)



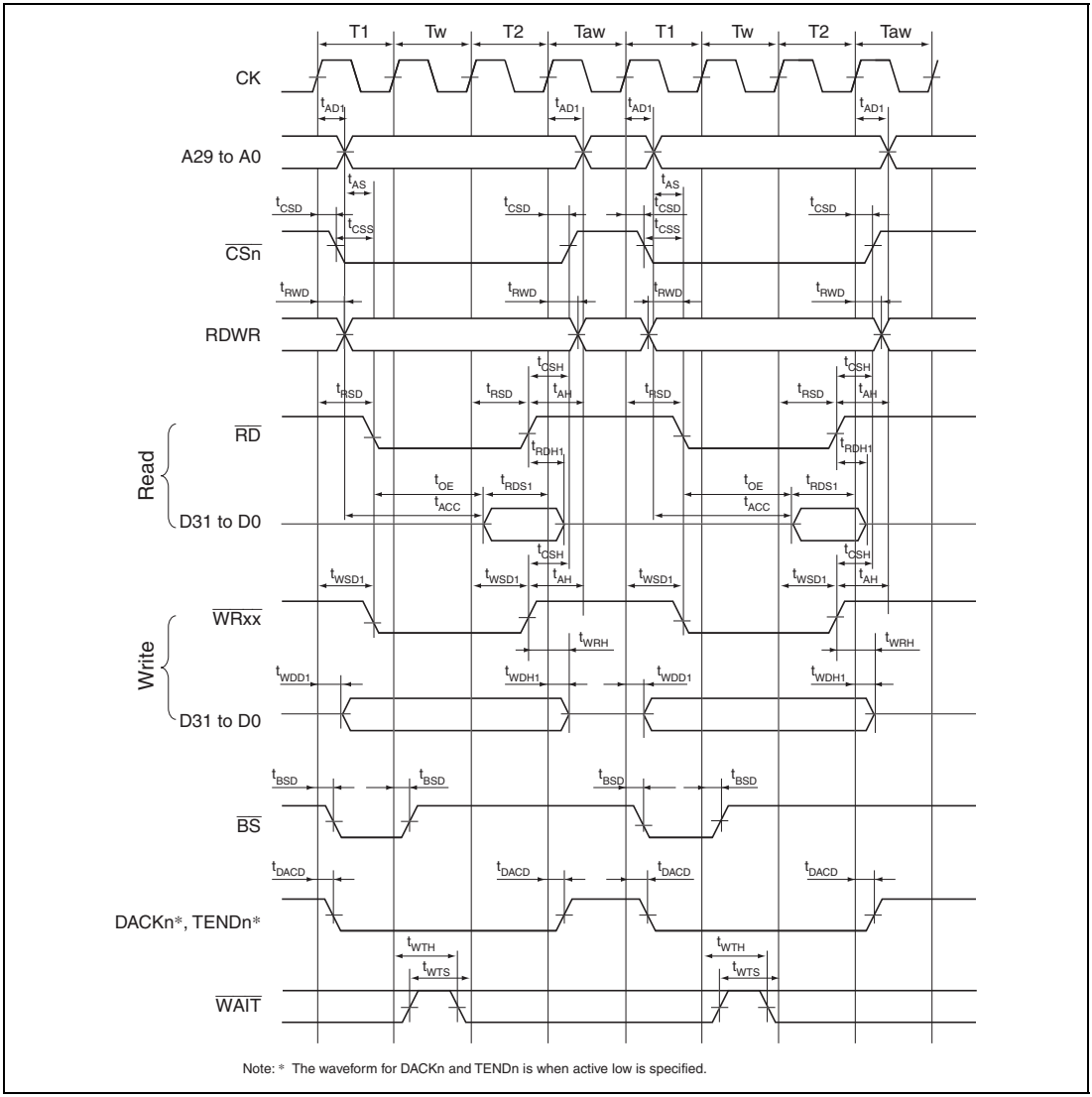
Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 28.12 Basic Bus Timing for Normal Space (One Software Wait Cycle)



Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 28.13 Basic Bus Timing for Normal Space (One External Wait Cycle)



Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 28.14 Basic Bus Timing for Normal Space (One Software Wait Cycle, External Wait Cycle Valid (WM Bit = 0), No Idle Cycle)

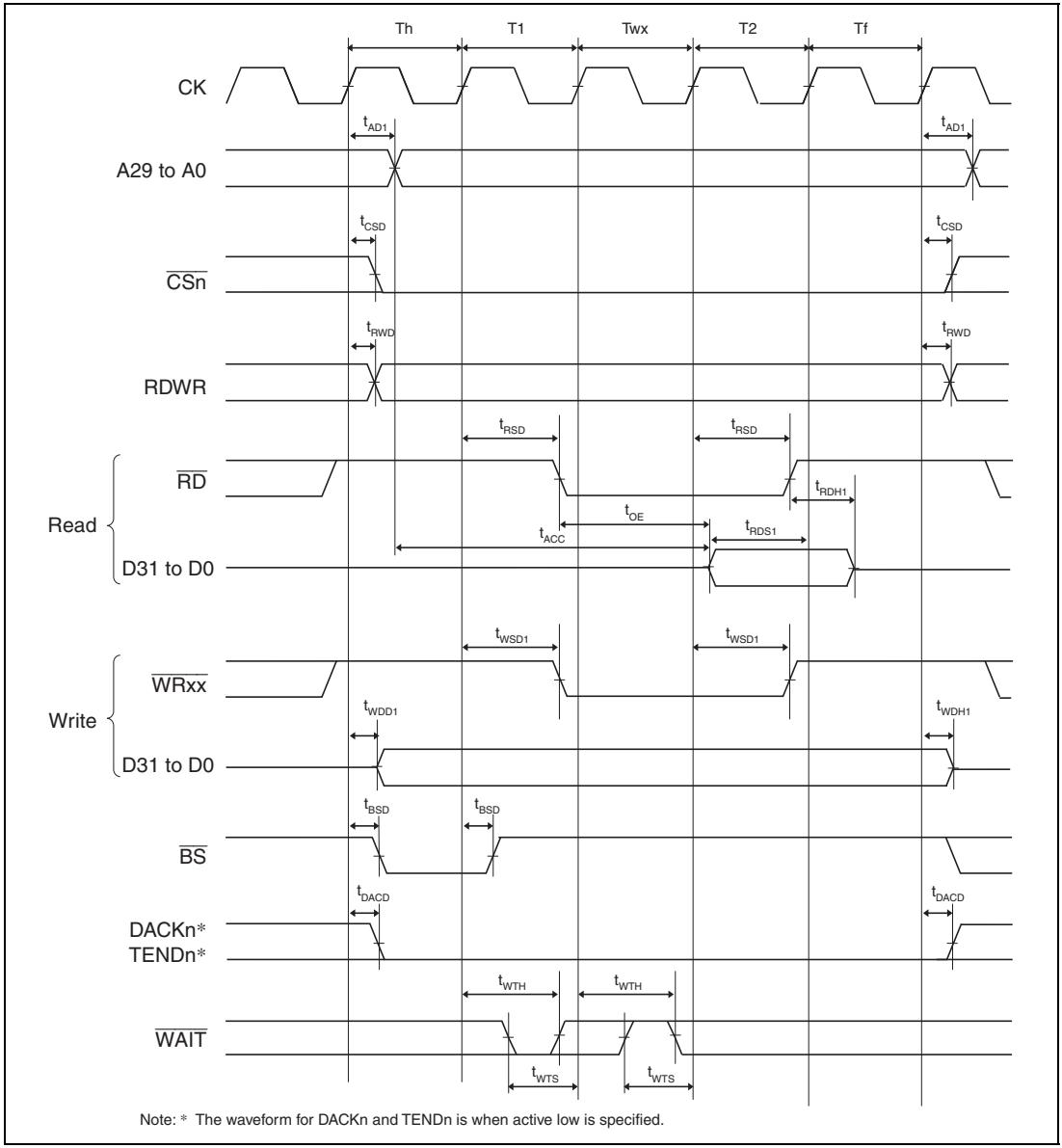


Figure 28.15 CS Extended Bus Cycle for Normal Space (SW = 1 Cycle, HW = 1 Cycle, One External Wait Cycle)

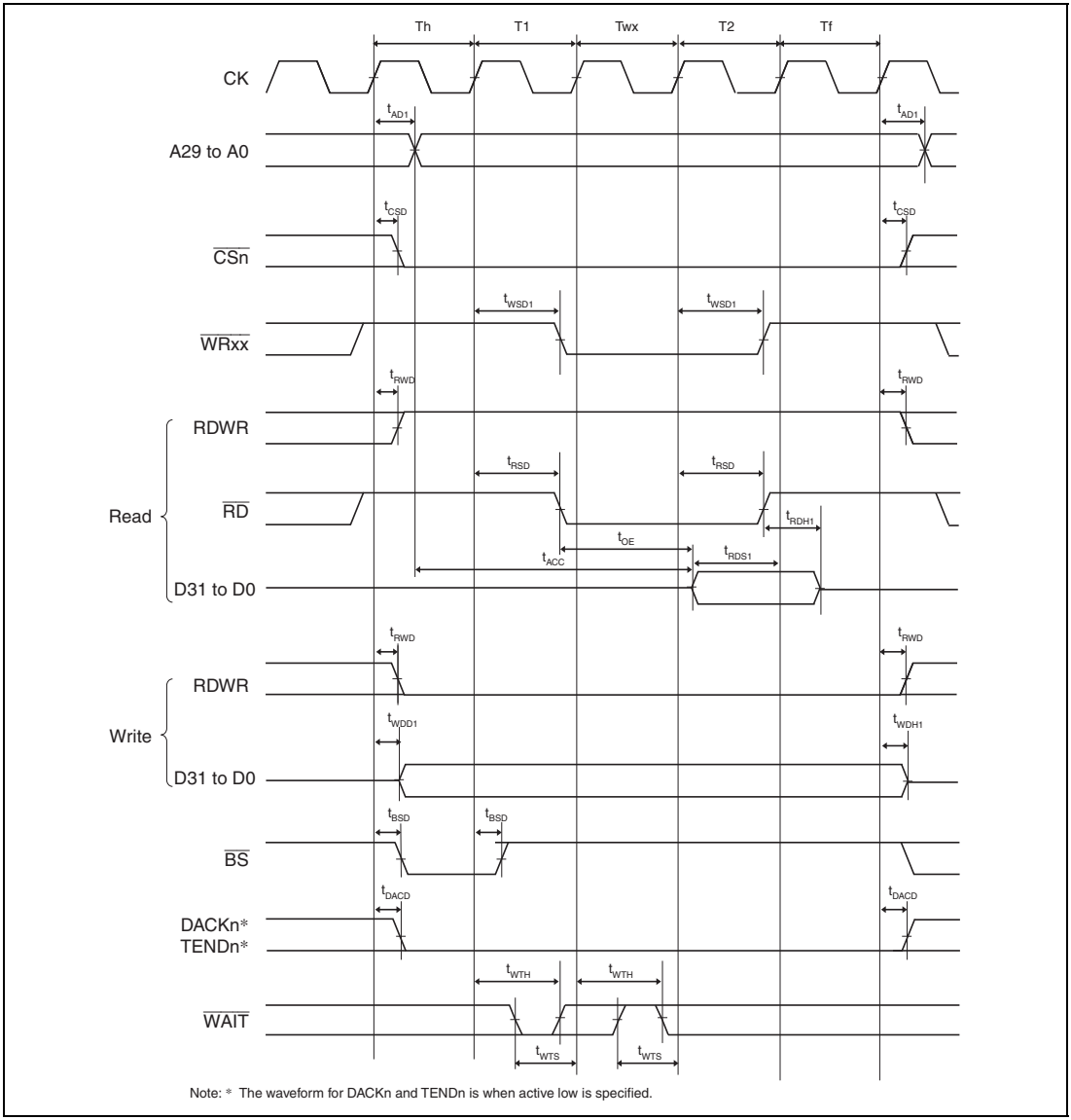
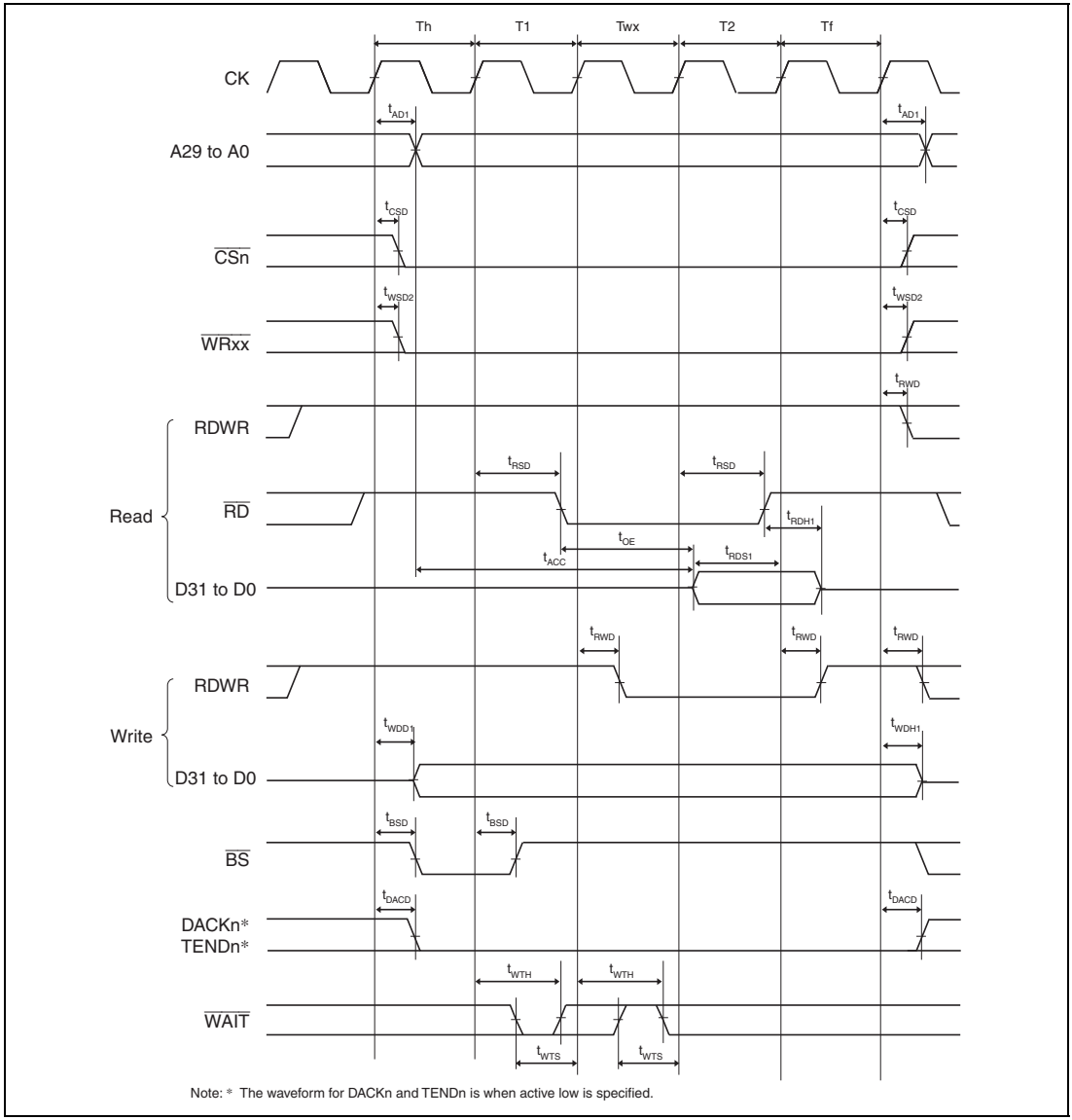
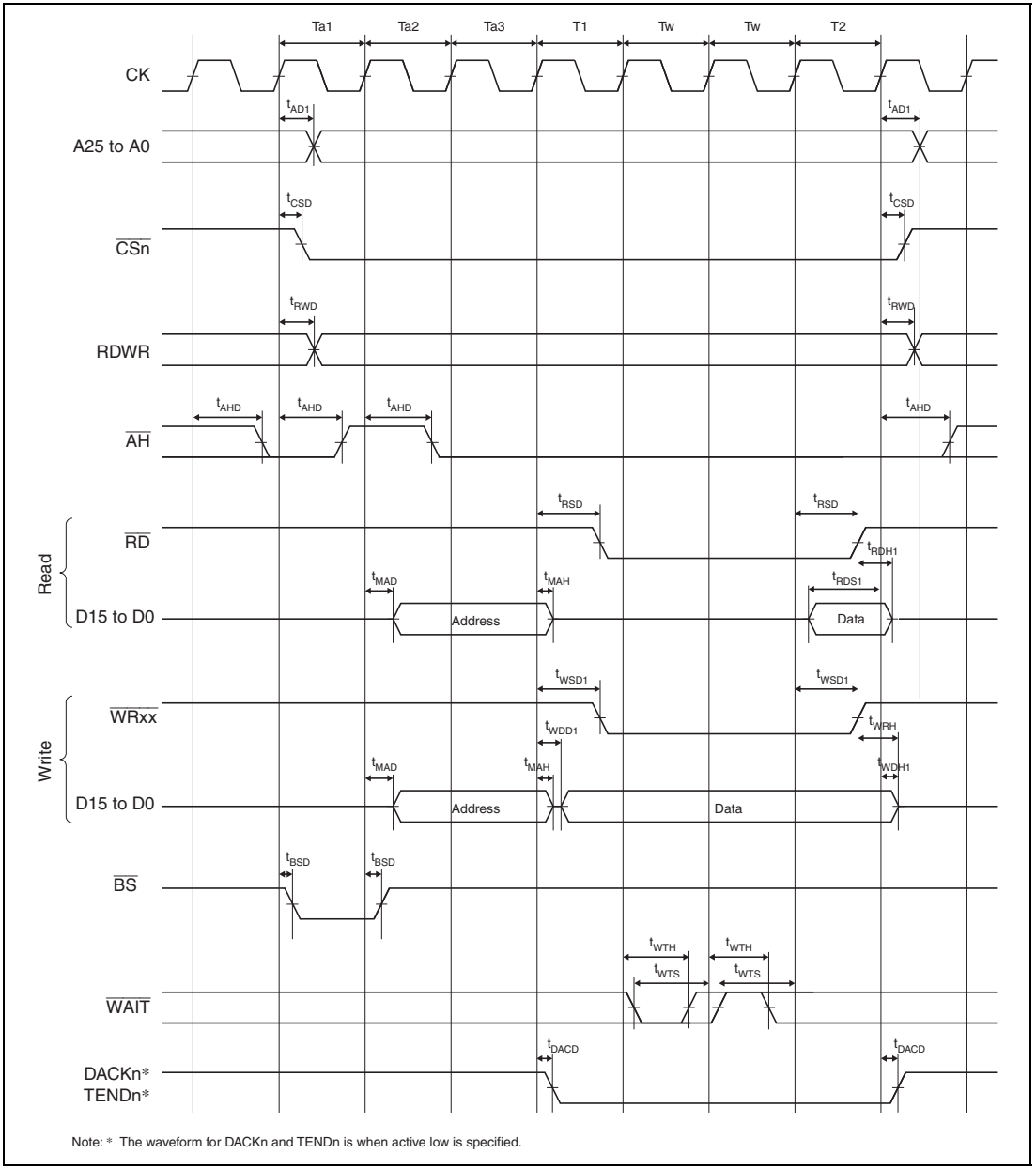


Figure 28.16 Bus Cycle of SRAM with Byte Selection (SW = 1 Cycle, HW = 1 Cycle, One External Wait Cycle, BAS = 0 ($\overline{UB}$/$\overline{LB}$ in Write Cycle Controlled))



Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 28.17 Bus Cycle of SRAM with Byte Selection (SW = 1 Cycle, HW = 1 Cycle, One External Wait Cycle, BAS = 1 ($\overline{WE}$ in Write Cycle Controlled))



Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 28.18 MPX-I/O Interface Bus Cycle
(Three Address Cycles, One Software Wait Cycle, One External Wait Cycle)

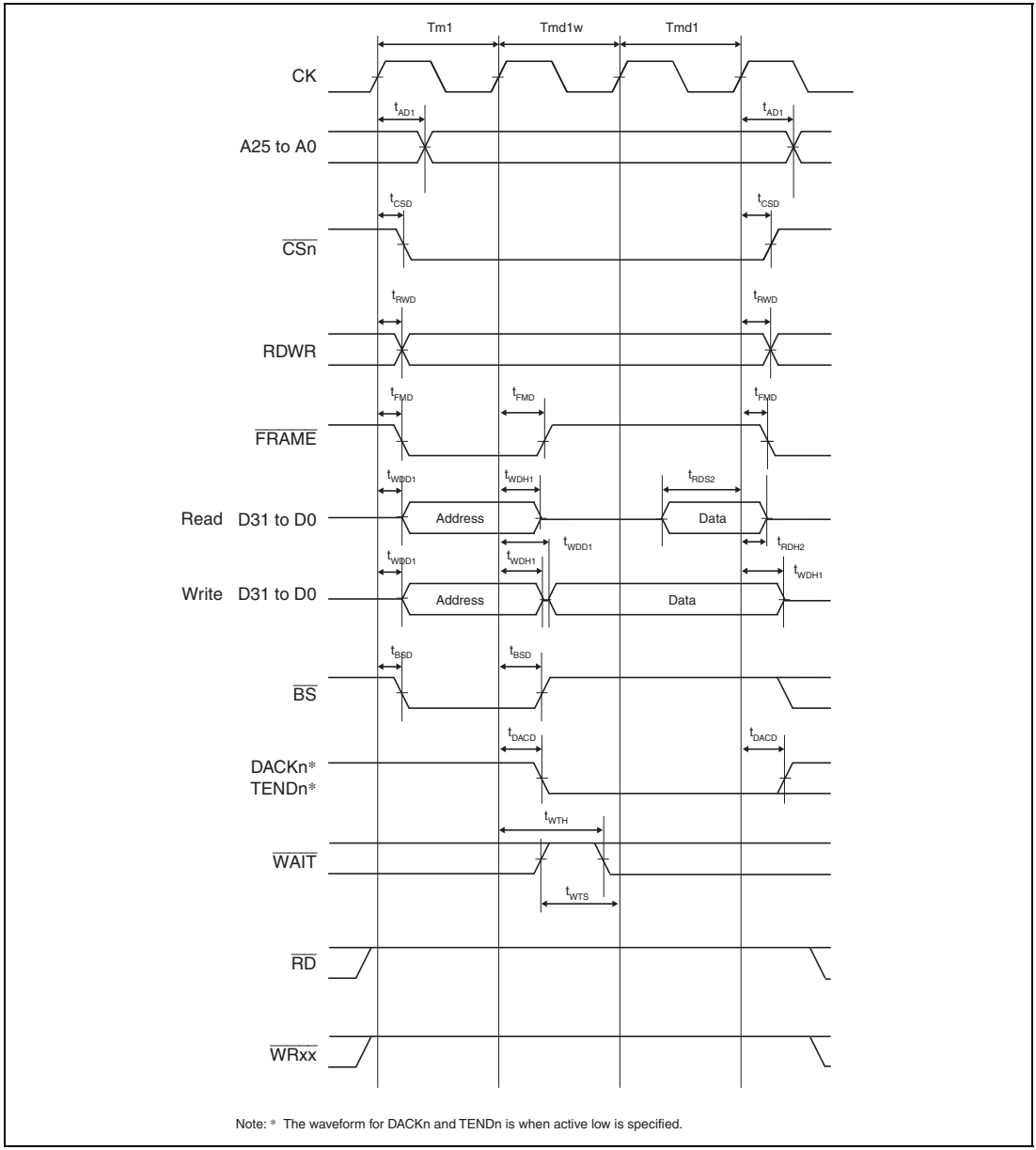


Figure 28.19 Burst MPX-I/O Interface Bus Cycle Single Read Write (One Address Cycle, One Software Wait Cycle)

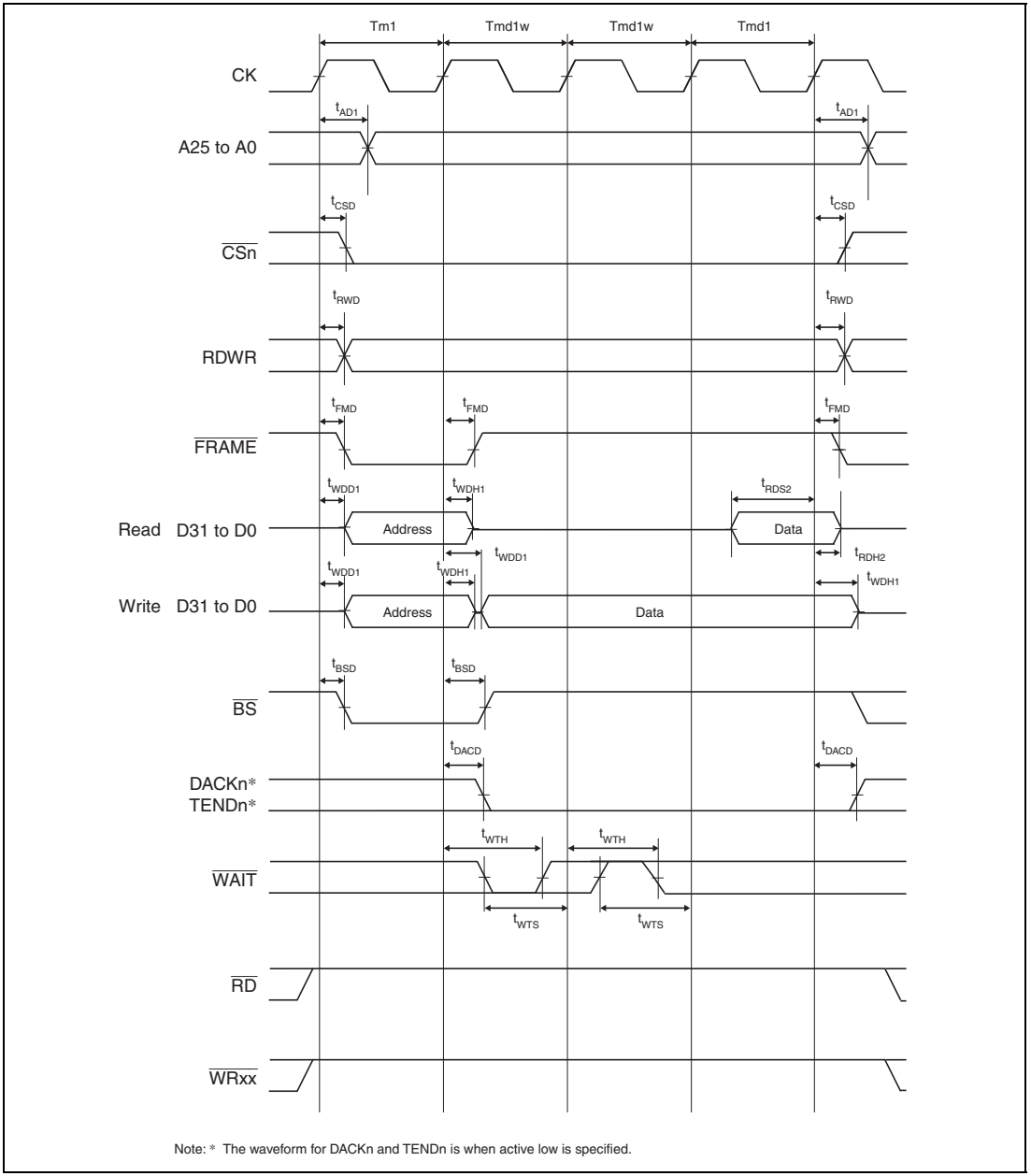


Figure 28.20 Burst MPX-I/O Interface Bus Cycle Single Read Write (One Address Cycle, One Software Wait Cycle, One External Wait Cycle)

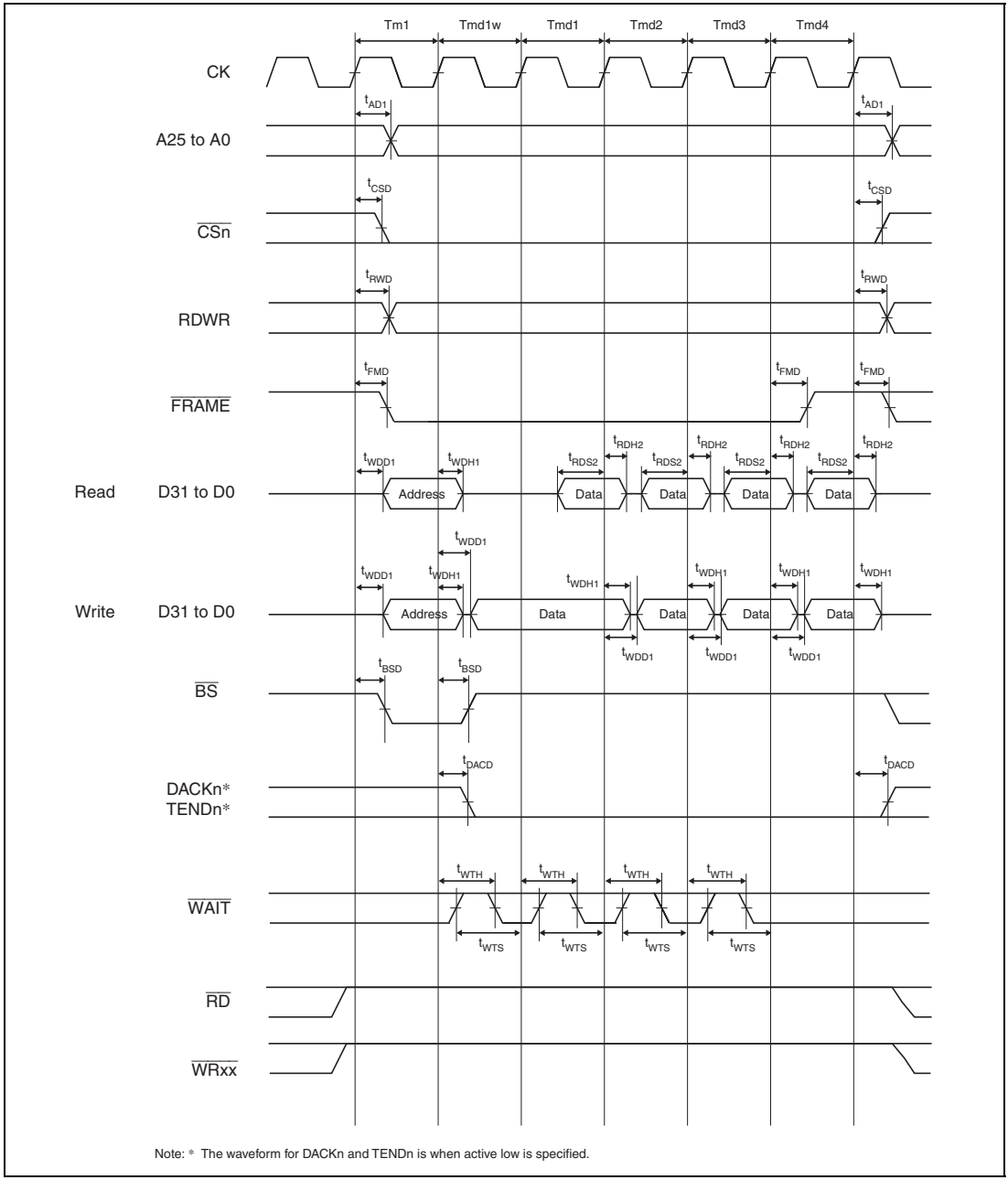


Figure 28.21 Burst MPX-I/O Interface Bus Cycle Burst Read Write (One Address Cycle, One Software Wait Cycle)

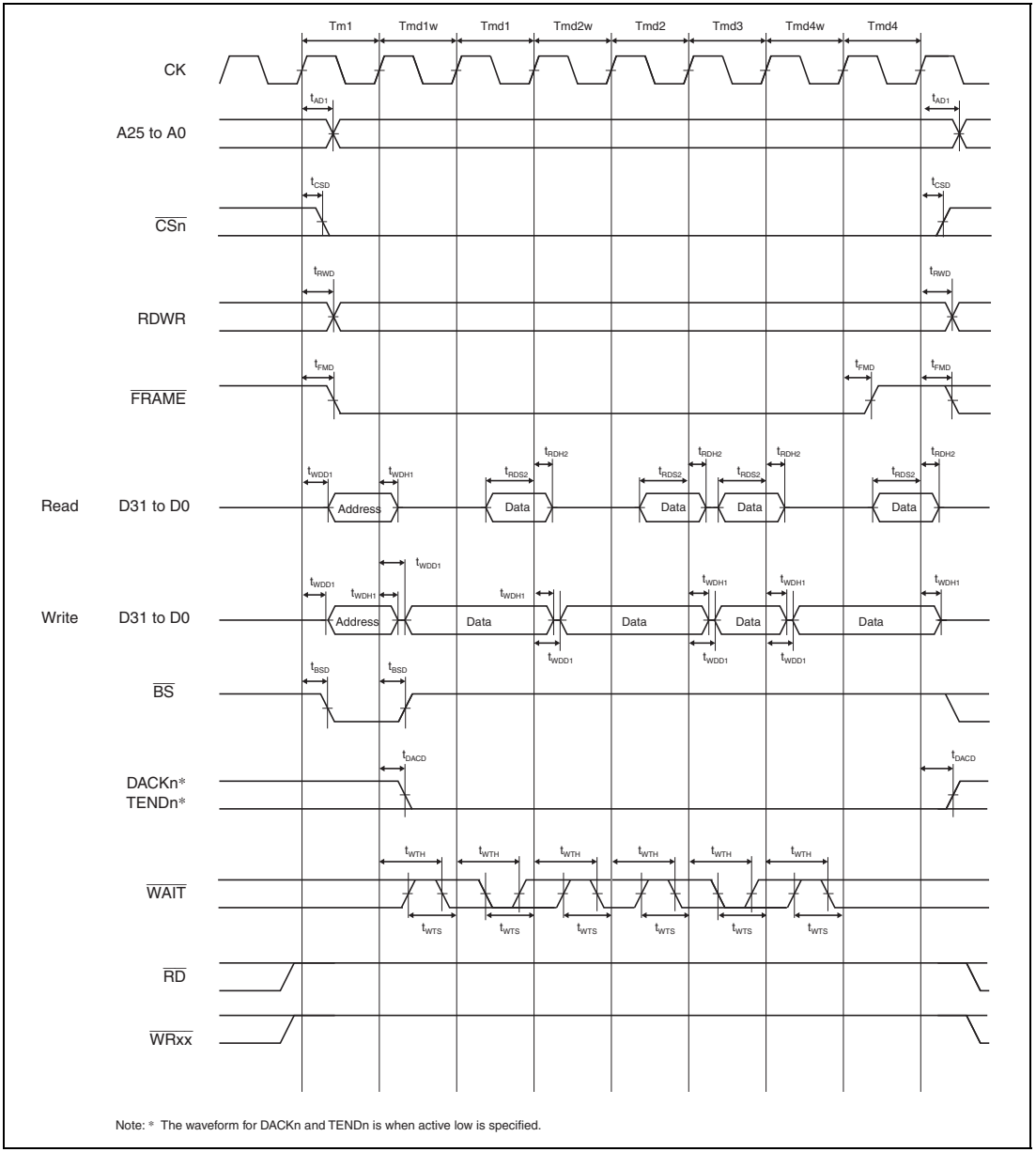
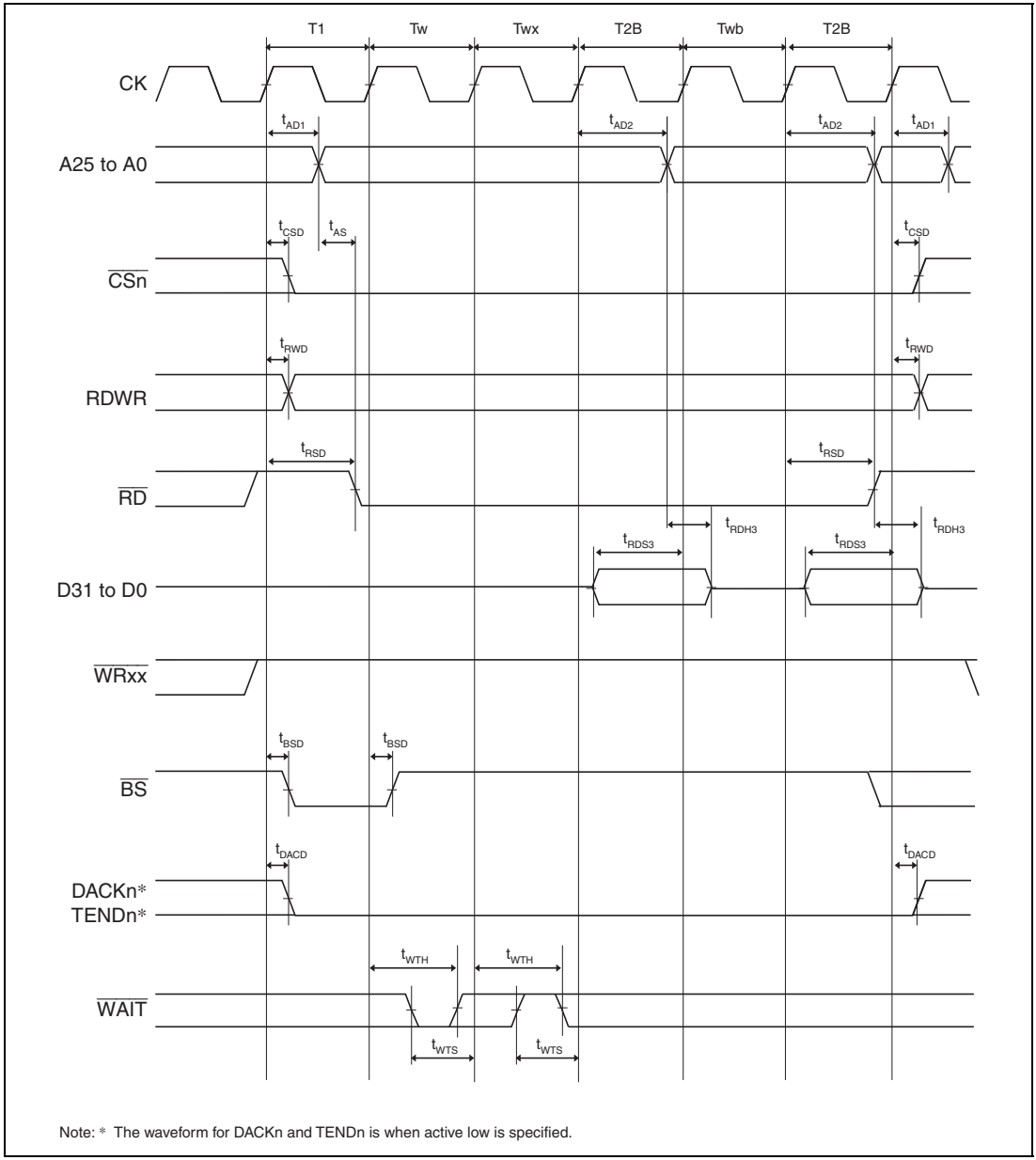
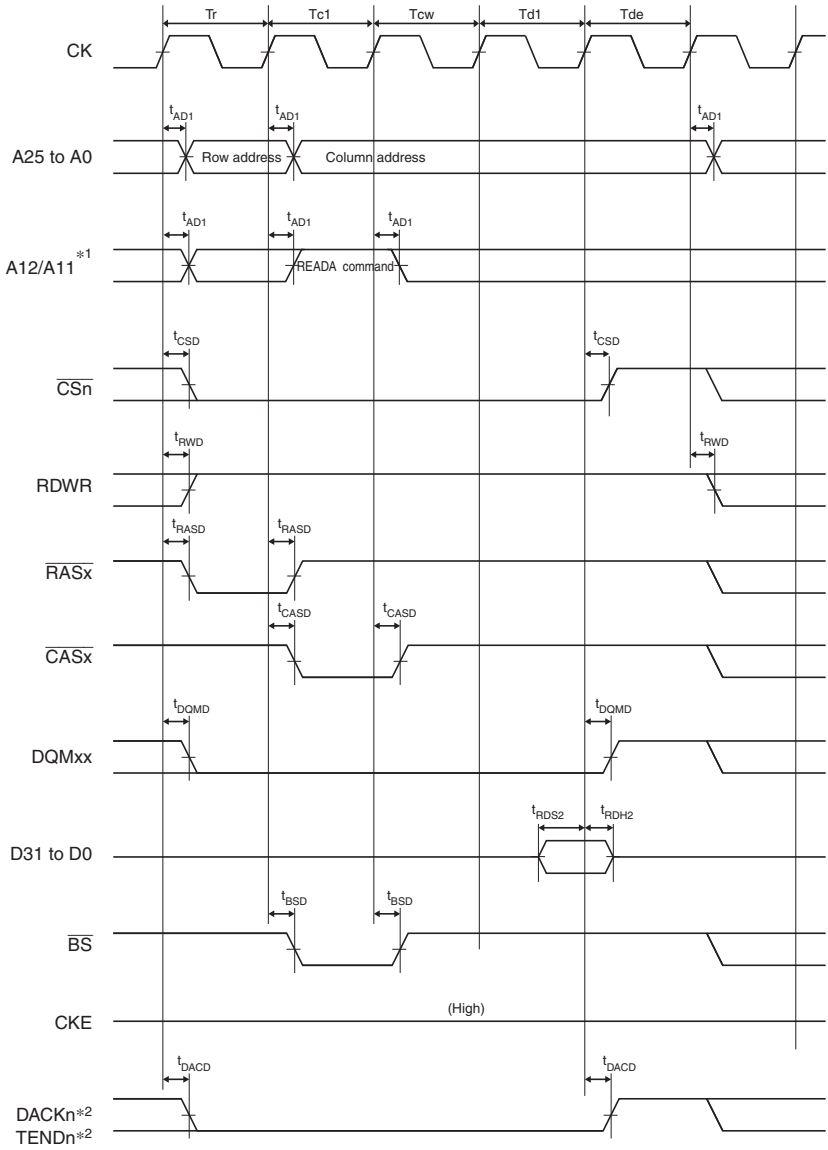


Figure 28.22 Burst MPX-I/O Interface Bus Cycle Burst Read Write (One Address Cycle, One Software Wait Cycle, External Wait Cycle)



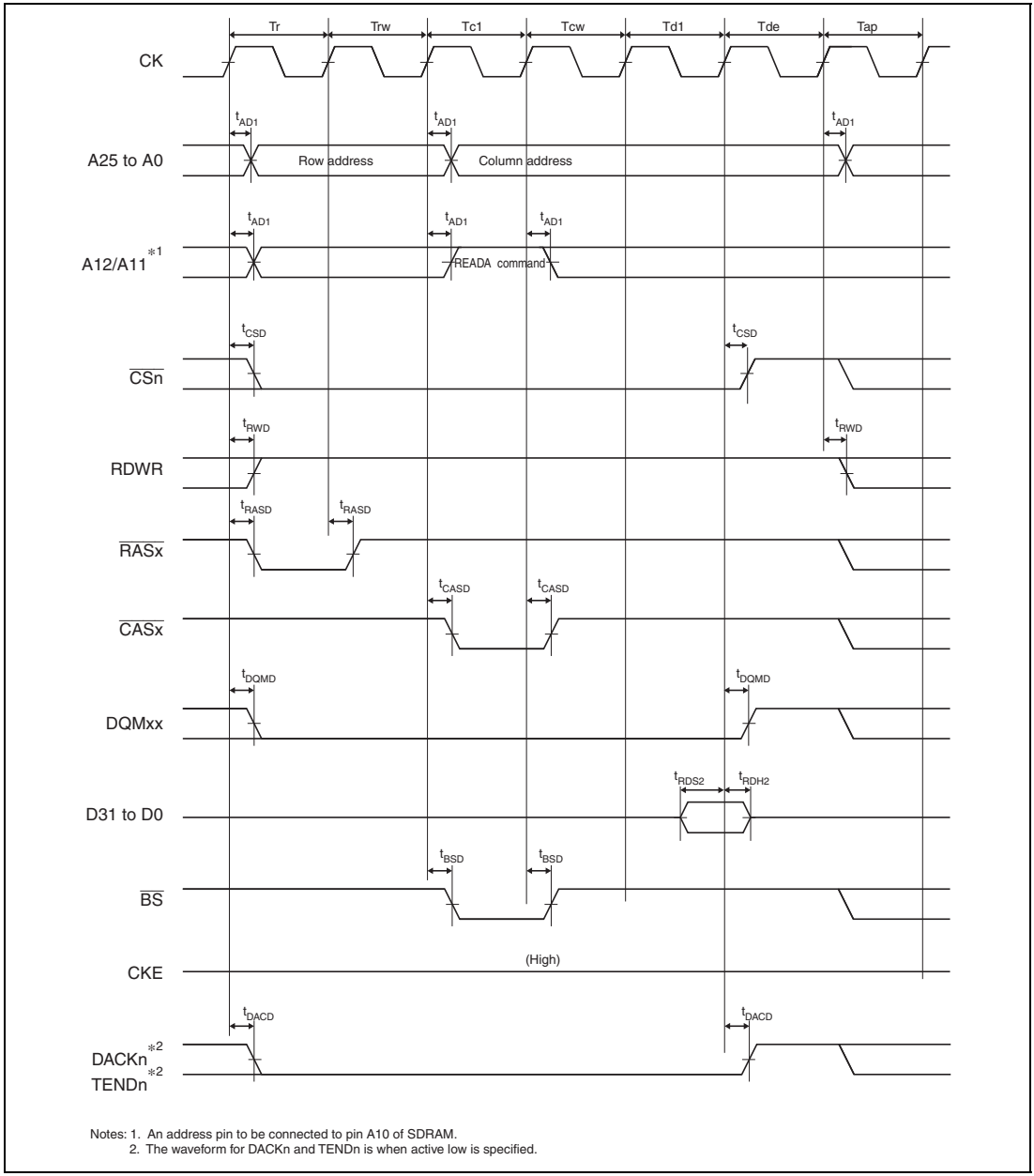
Note: * The waveform for DACKn and TENDn is when active low is specified.

Figure 28.23 Burst ROM Read Cycle
(One Software Wait Cycle, One External Wait Cycle, One Burst Wait Cycle, Two Bursts)



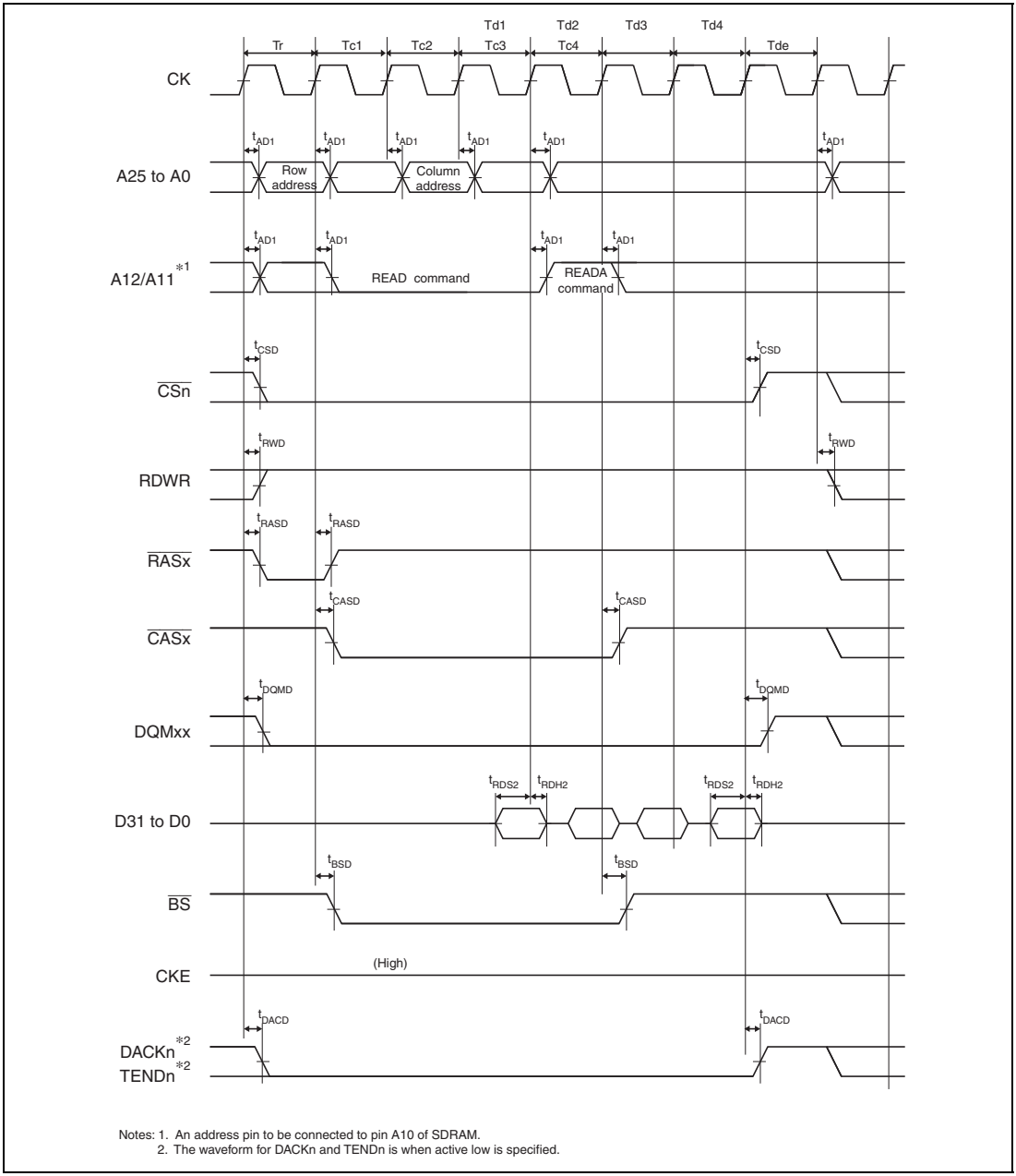
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.24 Synchronous DRAM Single Read Bus Cycle (Auto Precharge, CAS Latency 2, WTRCD = 0 Cycle, WTRP = 0 Cycle)



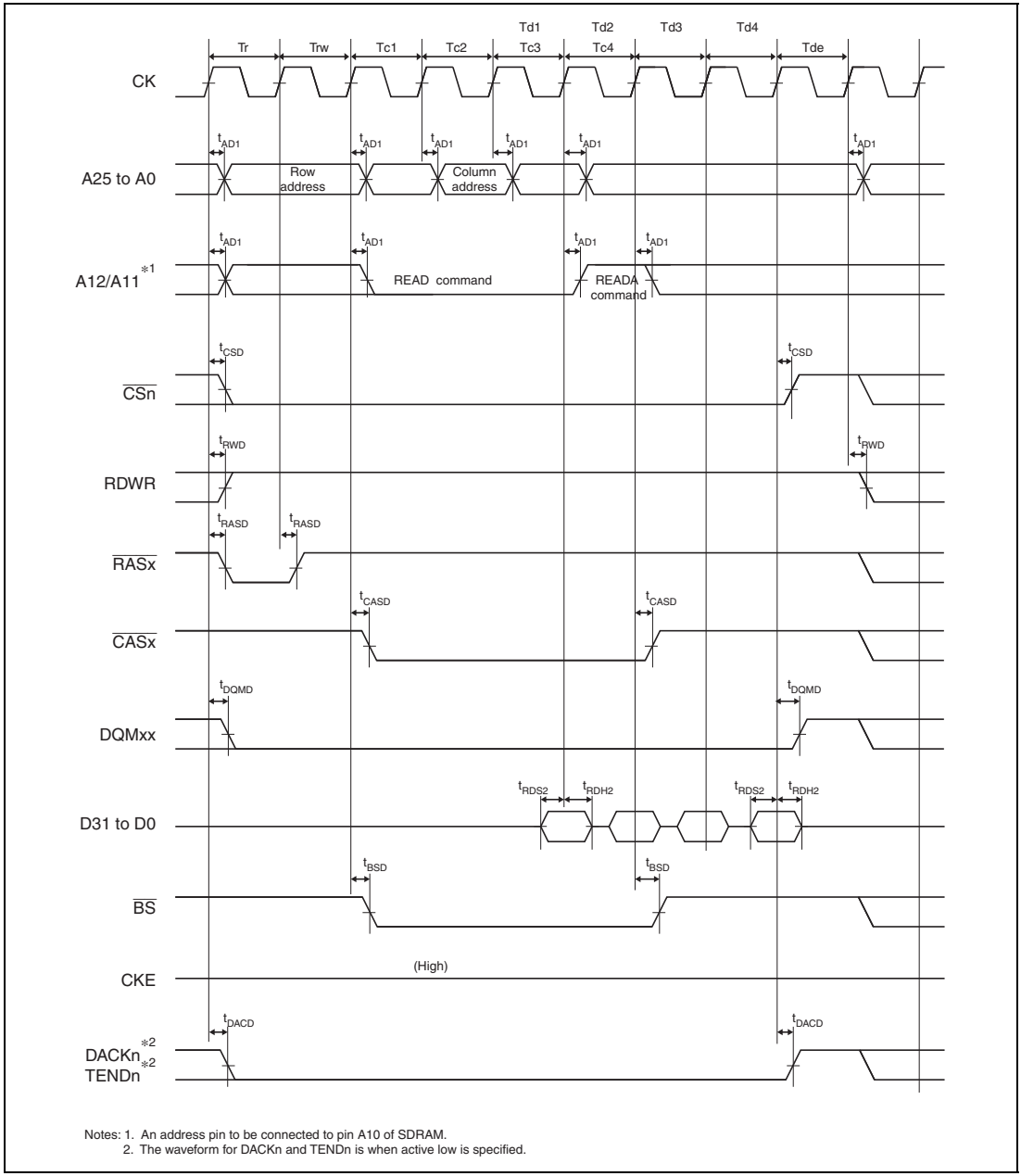
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.25 Synchronous DRAM Single Read Bus Cycle (Auto Precharge, CAS Latency 2, WTRCD = 1 Cycle, WTRP = 1 Cycle)



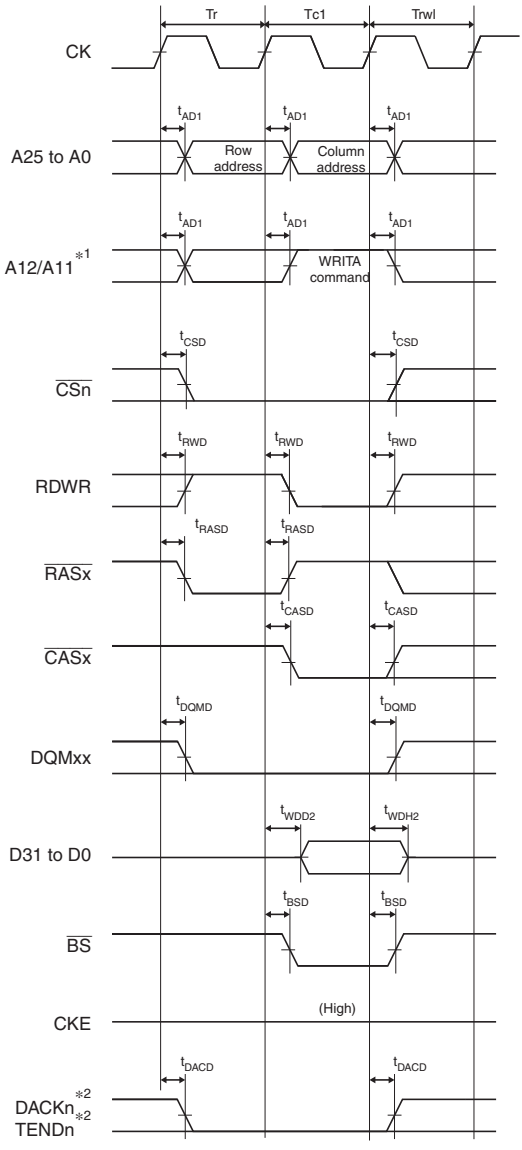
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.26 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Auto Precharge, CAS Latency 2, WTRCD = 0 Cycle, WTRP = 1 Cycle)



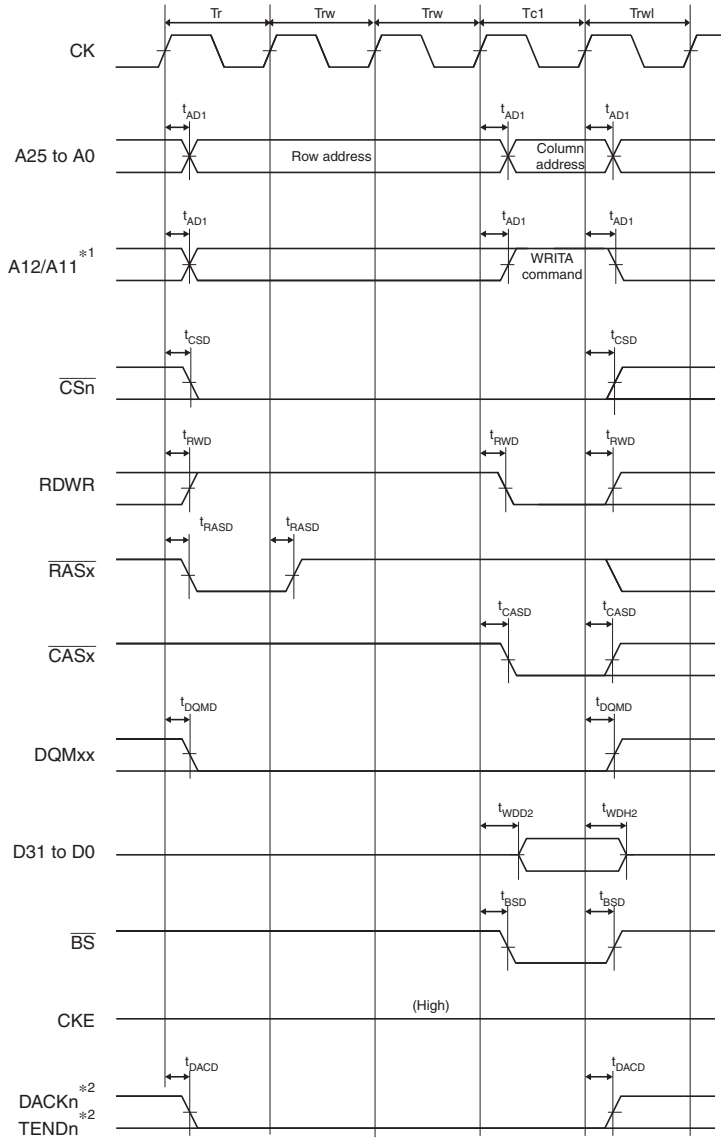
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.27 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Auto Precharge, CAS Latency 2, WTRCD = 1 Cycle, WTRP = 0 Cycle)



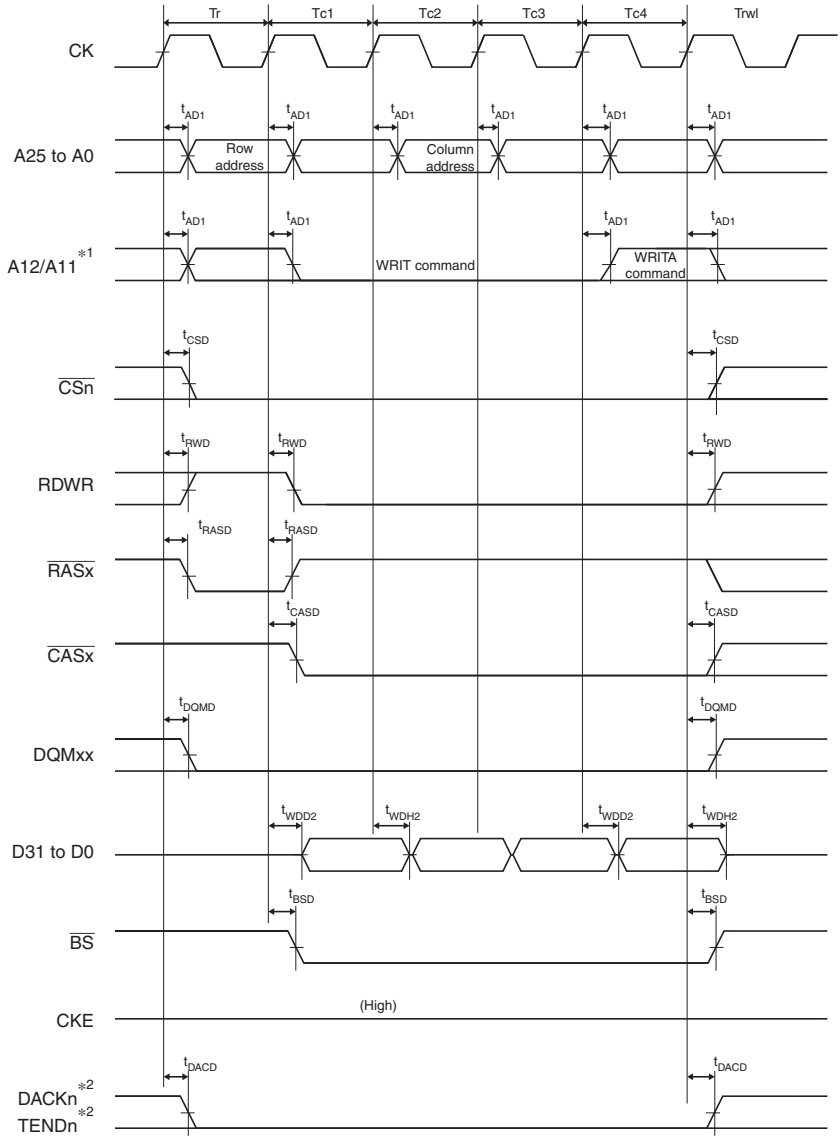
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.28 Synchronous DRAM Single Write Bus Cycle (Auto Precharge, TRWL = 1 Cycle)



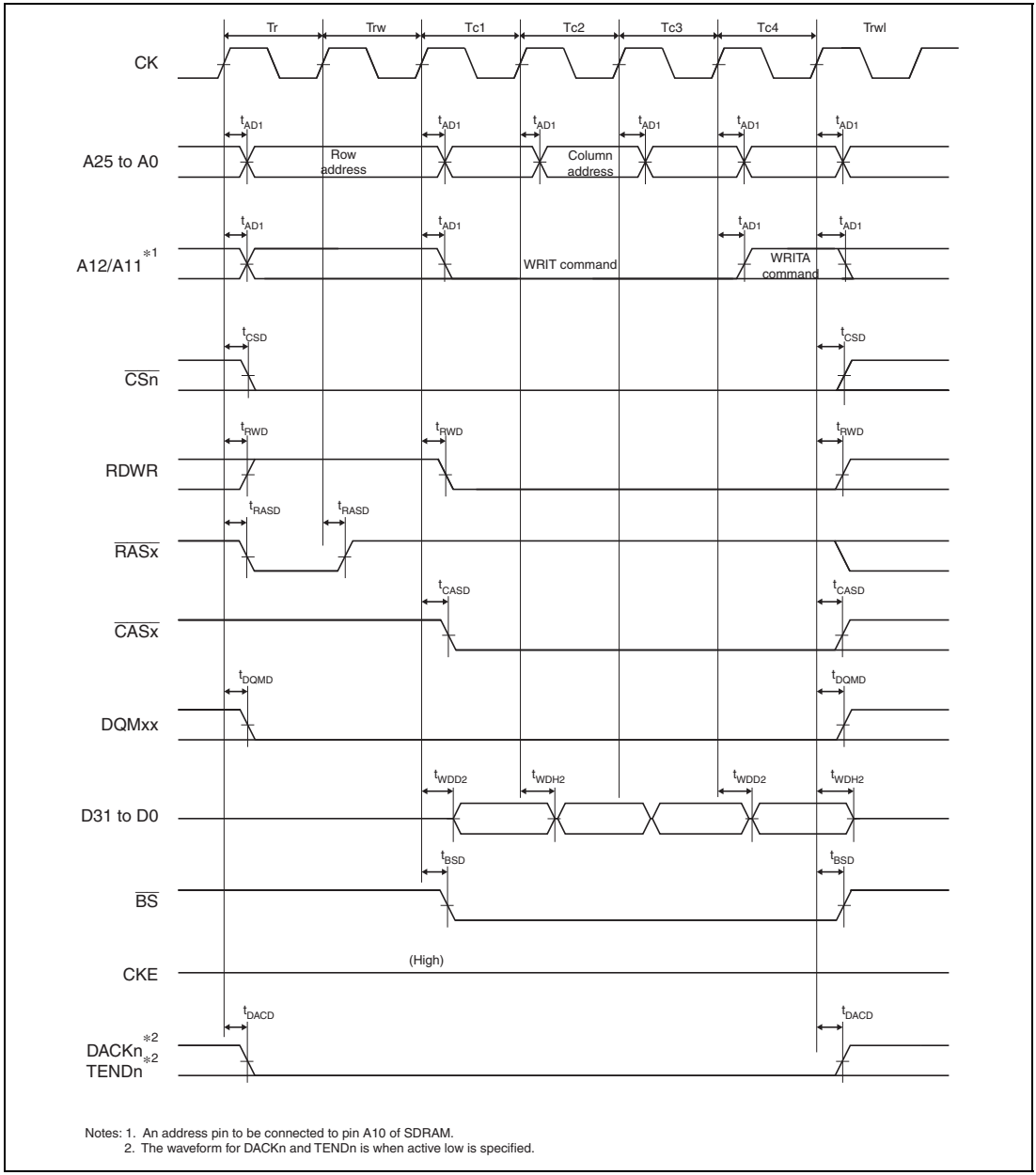
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.29 Synchronous DRAM Single Write Bus Cycle (Auto Precharge, WTRCD = 2 Cycles, TRWL = 1 Cycle)



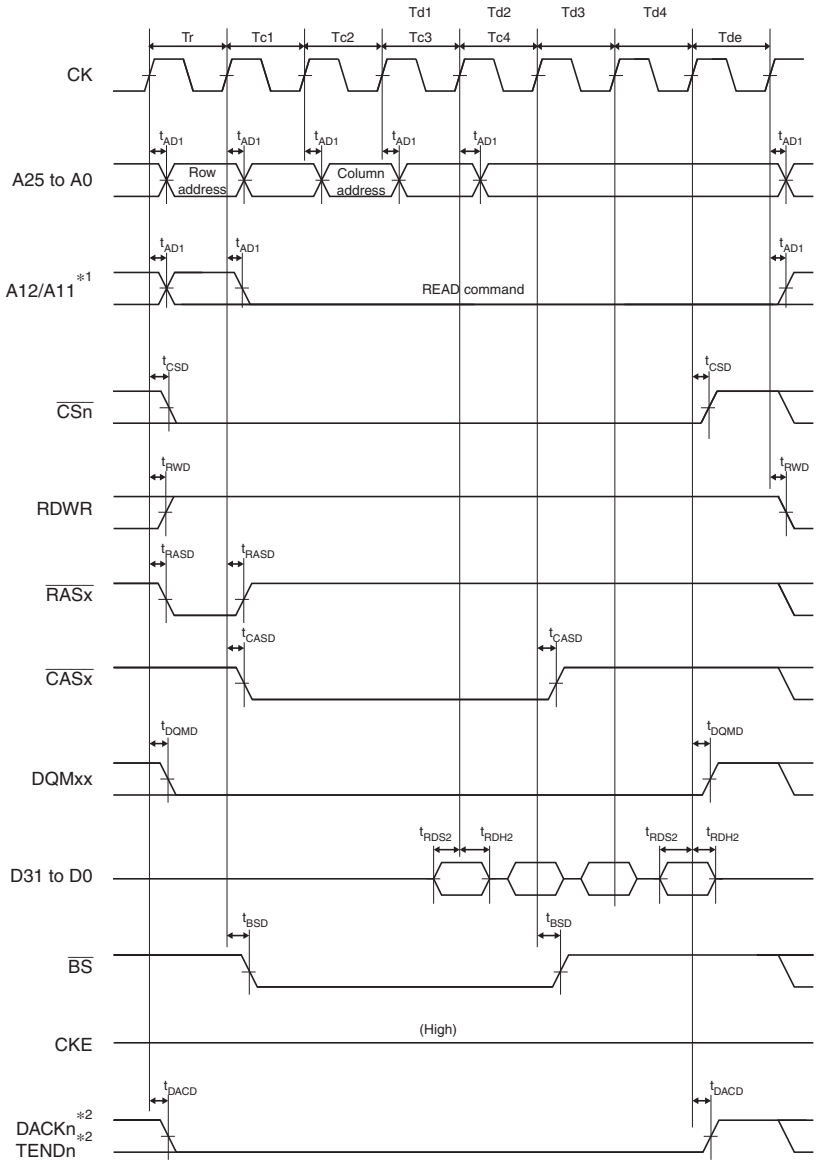
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.30 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Auto Precharge, WTRCD = 0 Cycle, TRWL = 1 Cycle)



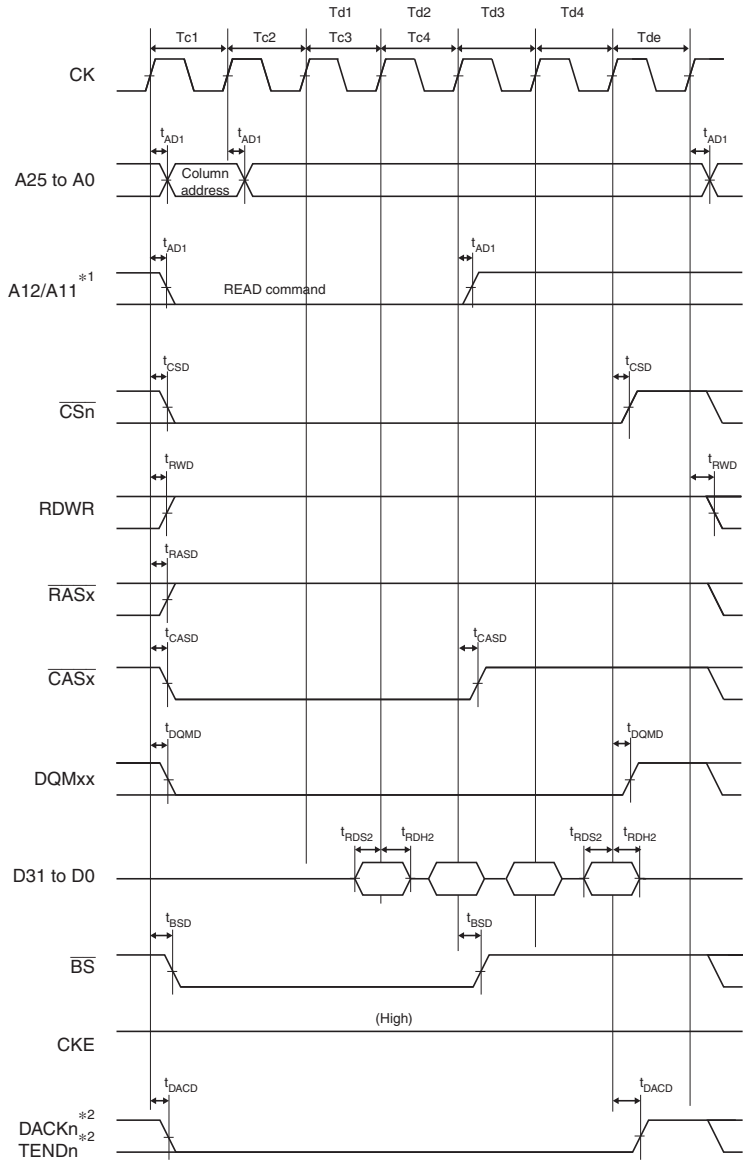
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.31 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Auto Precharge, WTRCD = 1 Cycle, TRWL = 1 Cycle)



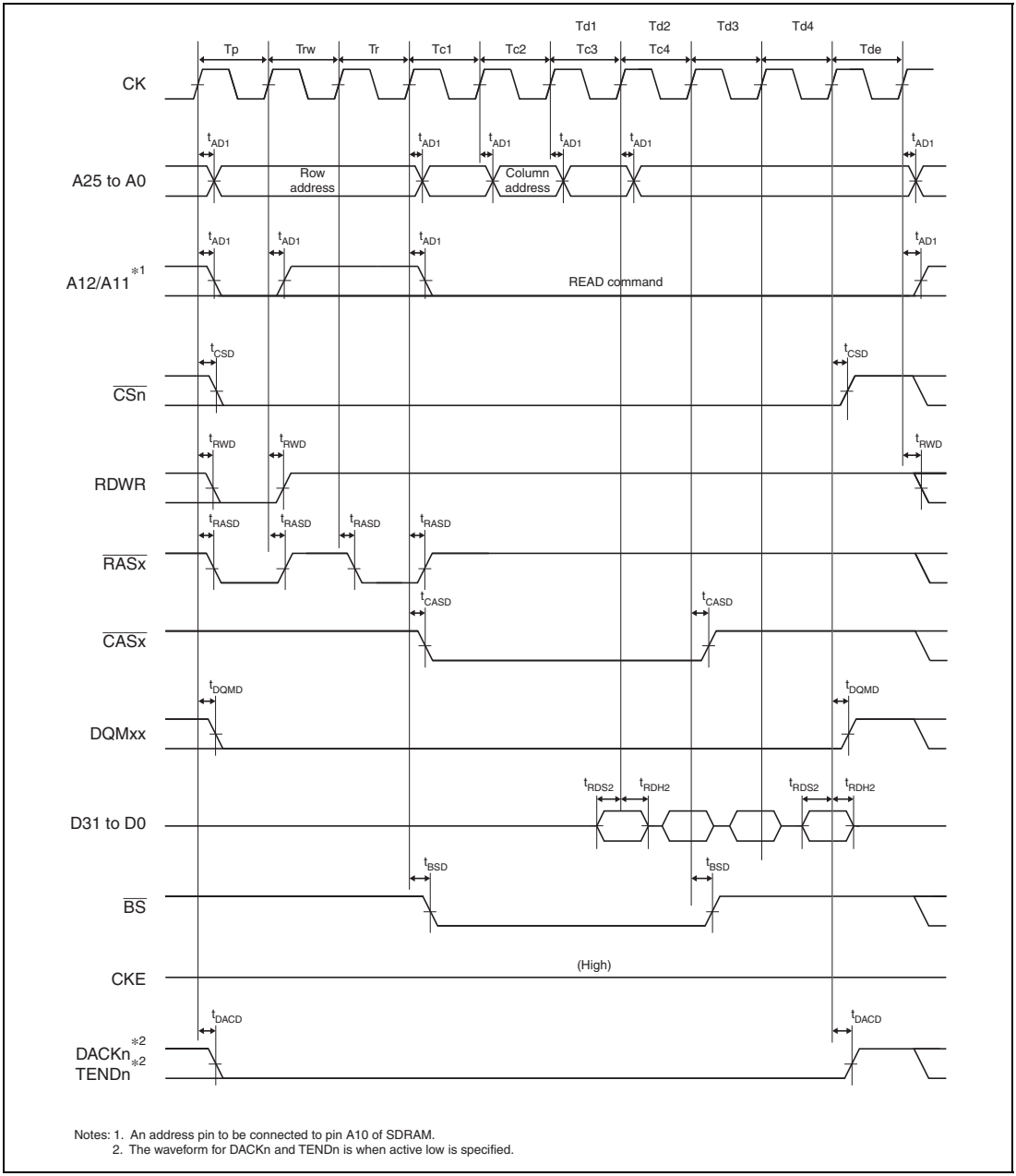
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.32 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Bank Active Mode: ACT + READ Commands, CAS Latency 2, WTRCD = 0 Cycle)



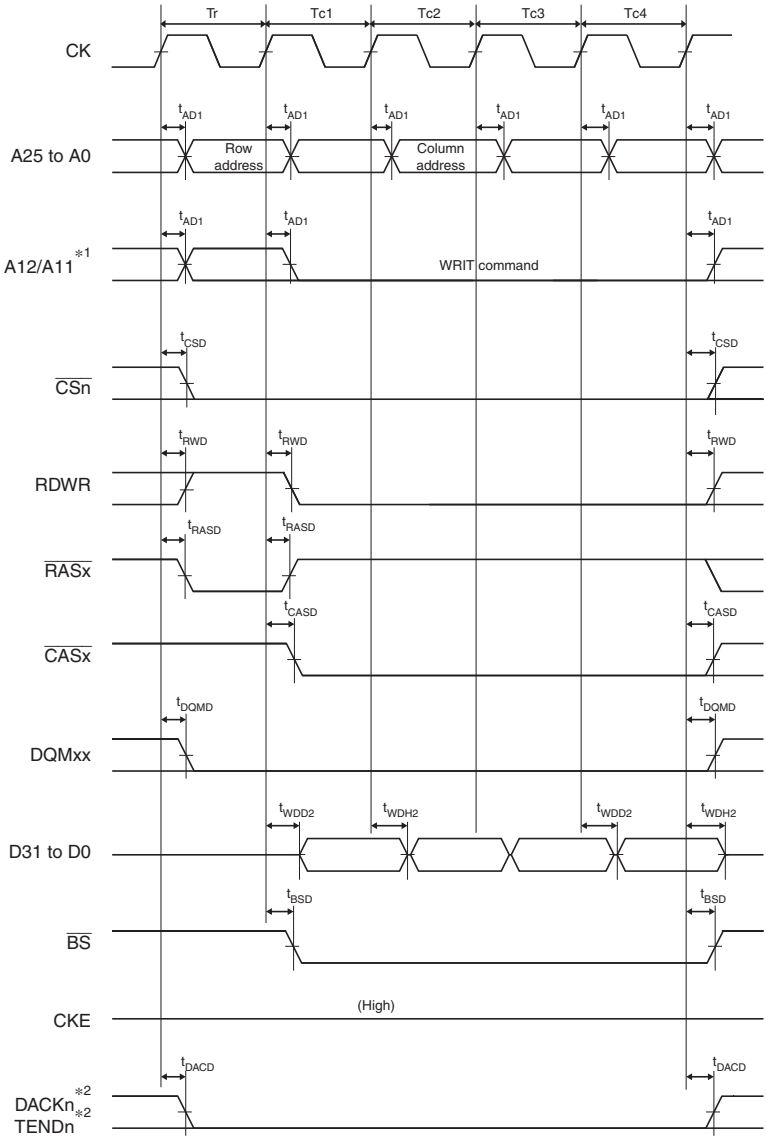
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.33 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles)
(Bank Active Mode: READ Command, Same Row Address, CAS Latency 2, WTRCD = 0 Cycle)



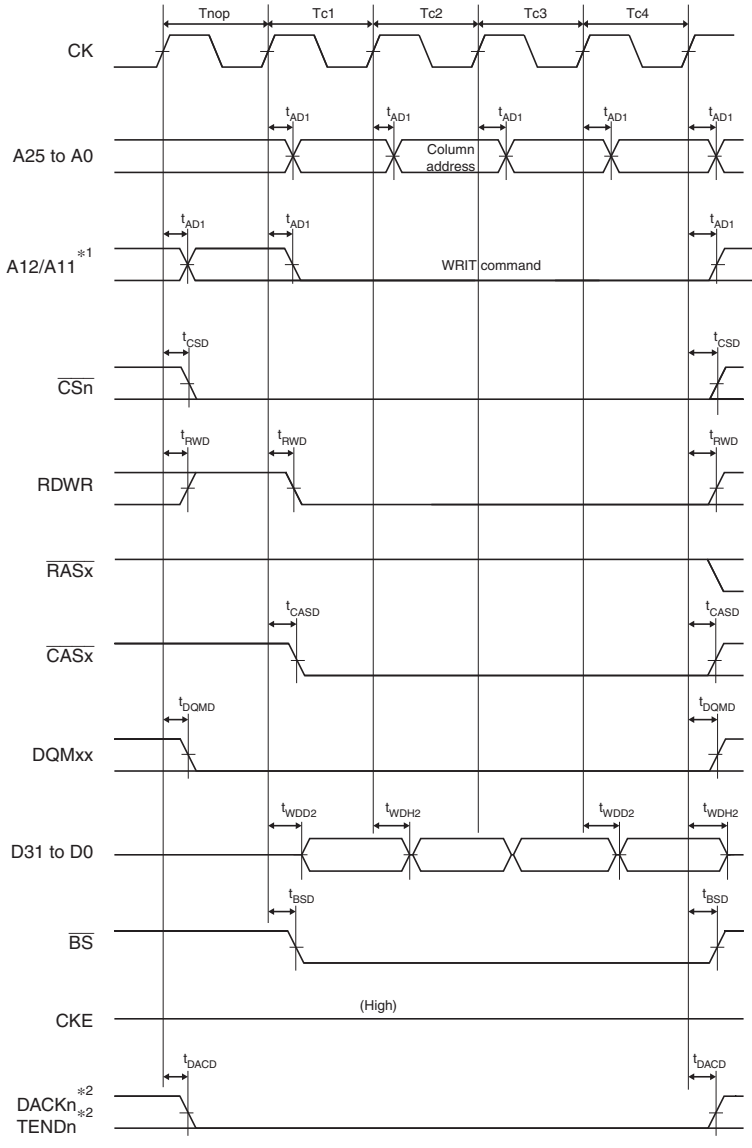
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.34 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Bank Active Mode: PRE + ACT + READ Commands, Different Row Addresses, CAS Latency 2, WTRCD = 0 Cycle)



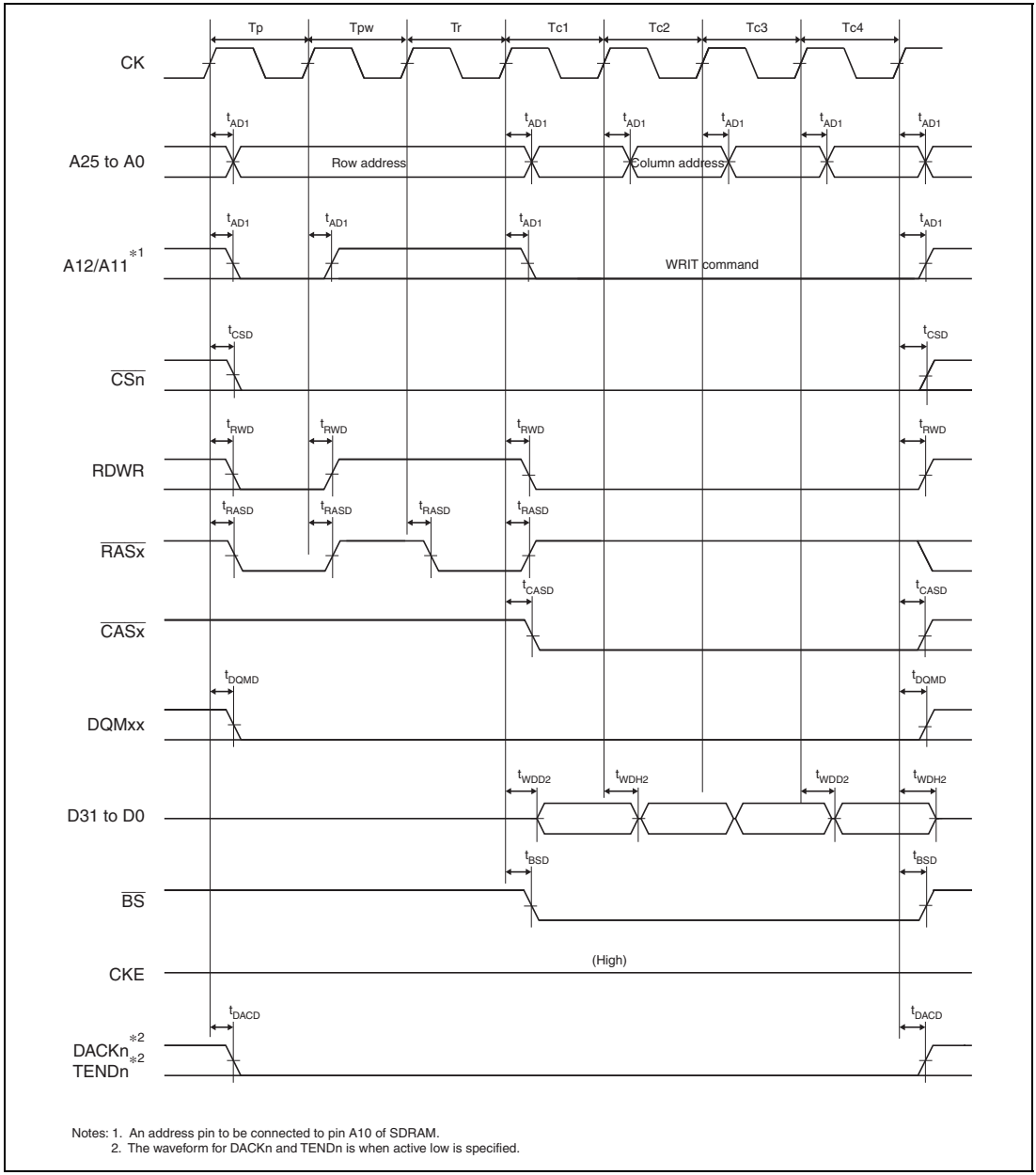
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.35 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Bank Active Mode: ACT + WRITE Commands, WTRCD = 0 Cycle, TRWL = 0 Cycle)



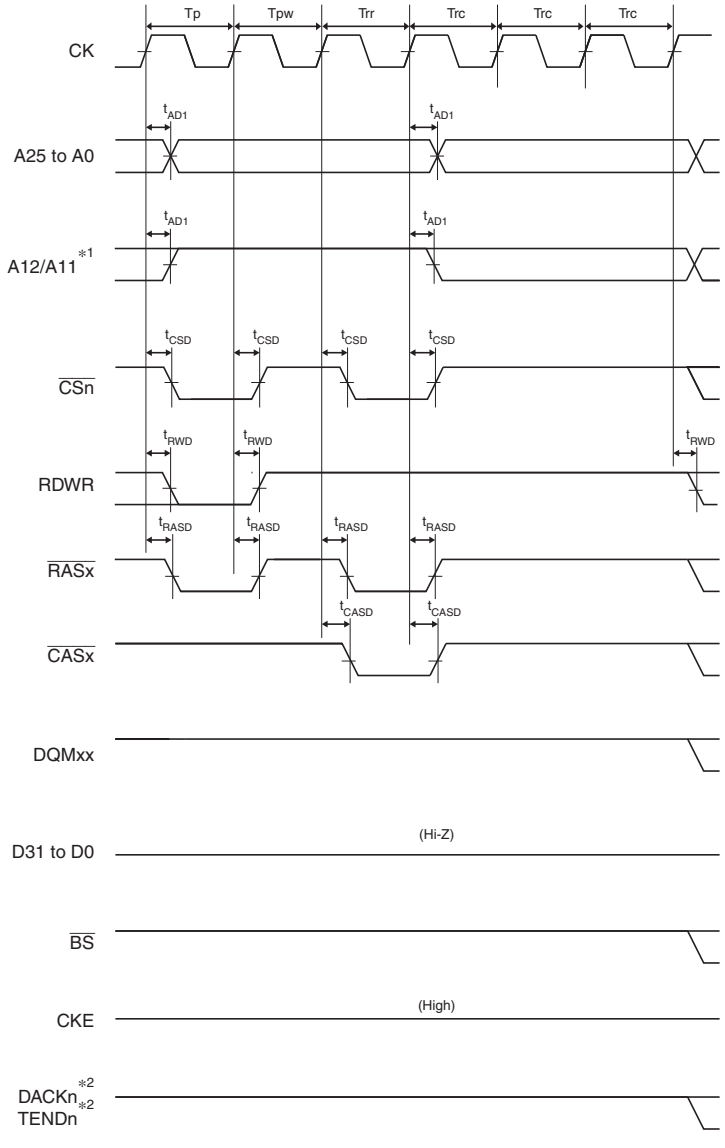
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.36 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Bank Active Mode: WRITE Command, Same Row Address, WTRCD = 0 Cycle, TRWL = 0 Cycle)



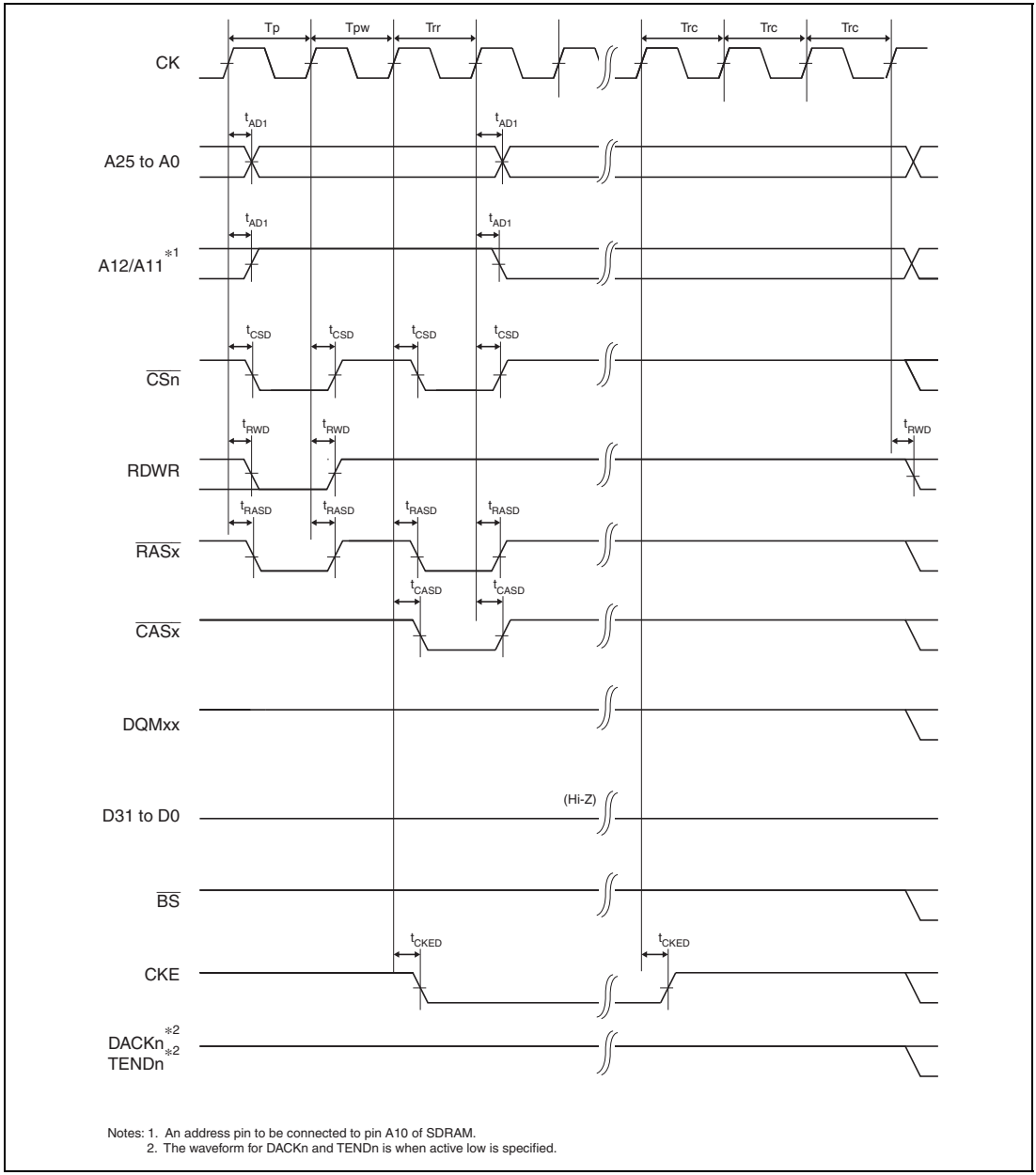
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.37 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Bank Active Mode: PRE + ACT + WRITE Commands, Different Row Addresses, WTRCD = 0 Cycle, TRWL = 0 Cycle)



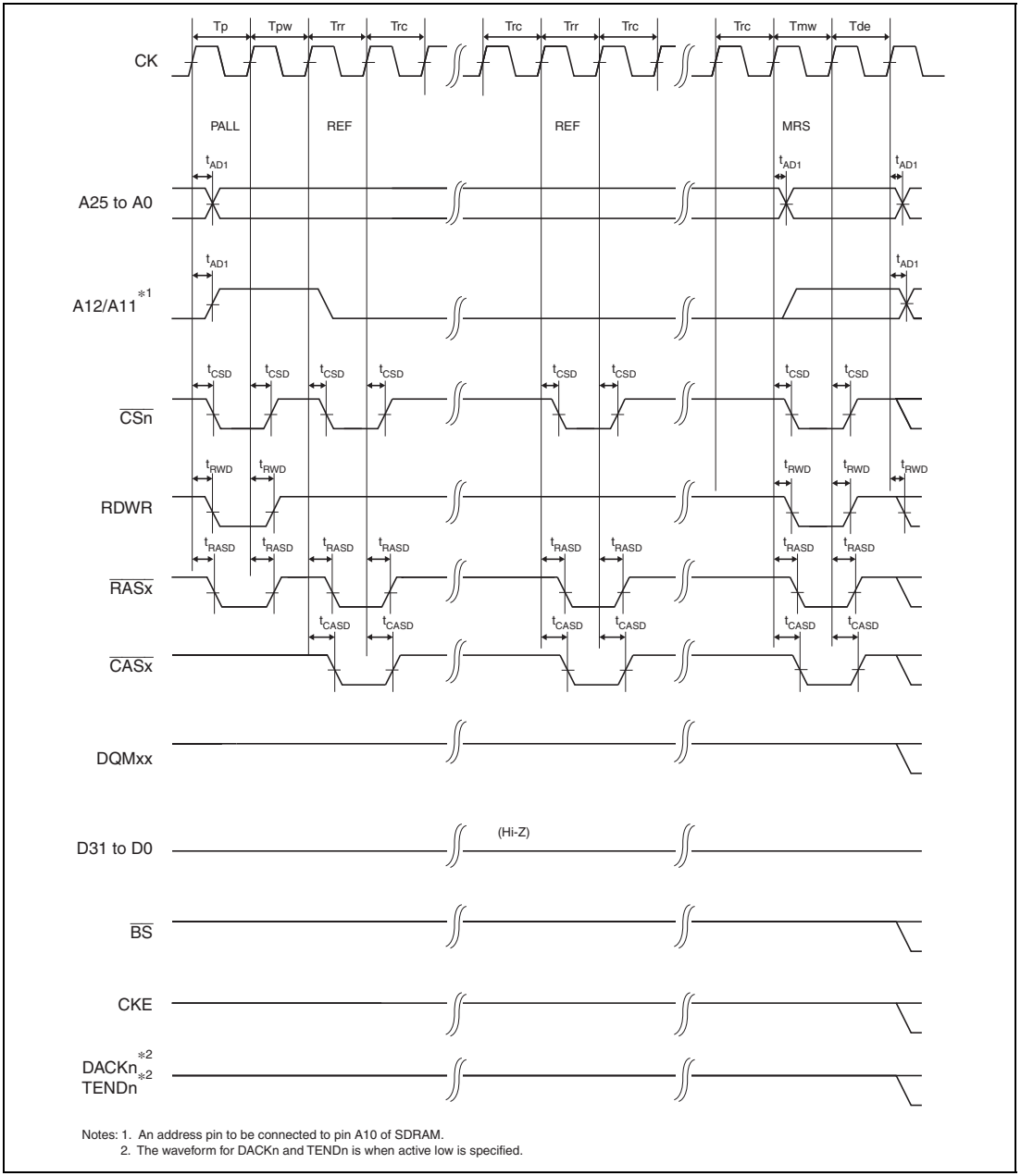
Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.38 Synchronous DRAM Auto-Refreshing Timing (WTRP = 1 Cycle, WTRC = 3 Cycles)



Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.39 Synchronous DRAM Self-Refreshing Timing (WTRP = 1 Cycle, WTRC = 3 Cycles)



Notes: 1. An address pin to be connected to pin A10 of SDRAM.
2. The waveform for DACKn and TENDn is when active low is specified.

Figure 28.40 Synchronous DRAM Mode Register Write Timing (WTRP = 1 Cycle)

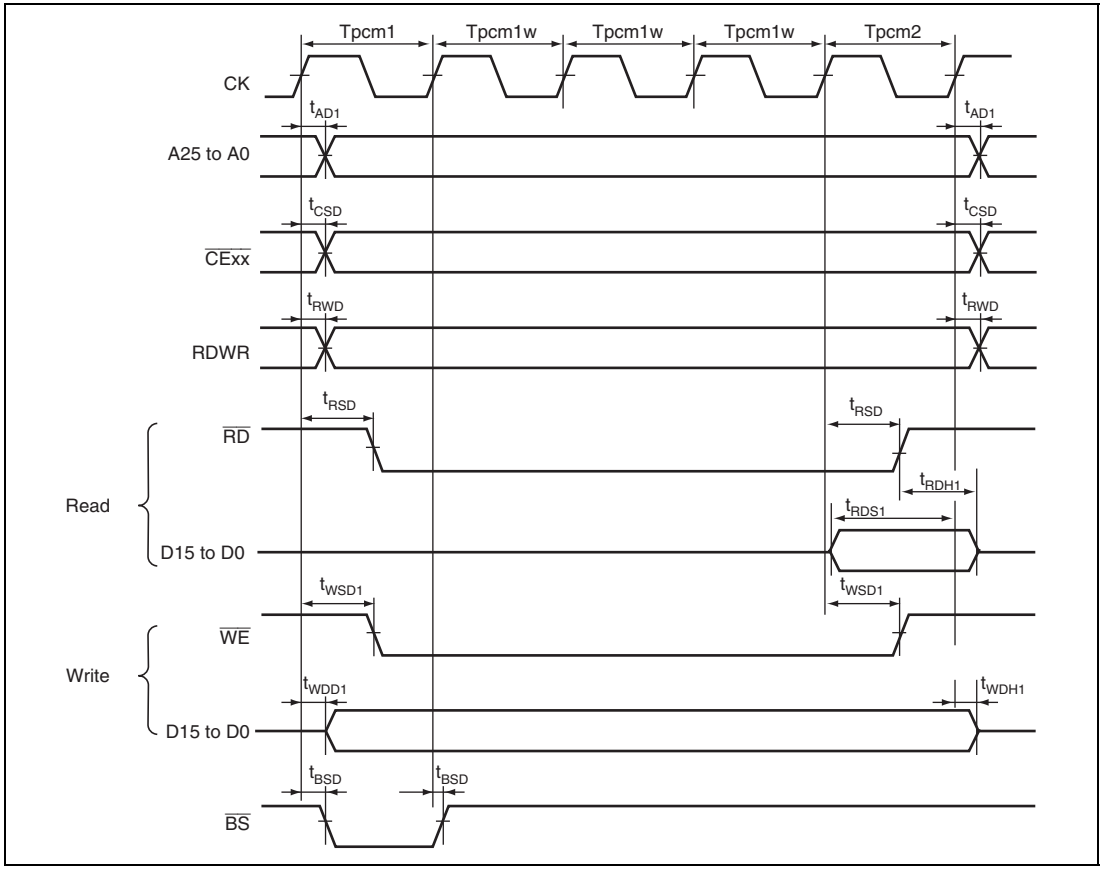


Figure 28.41 PCMCIA Memory Card Interface Bus Timing

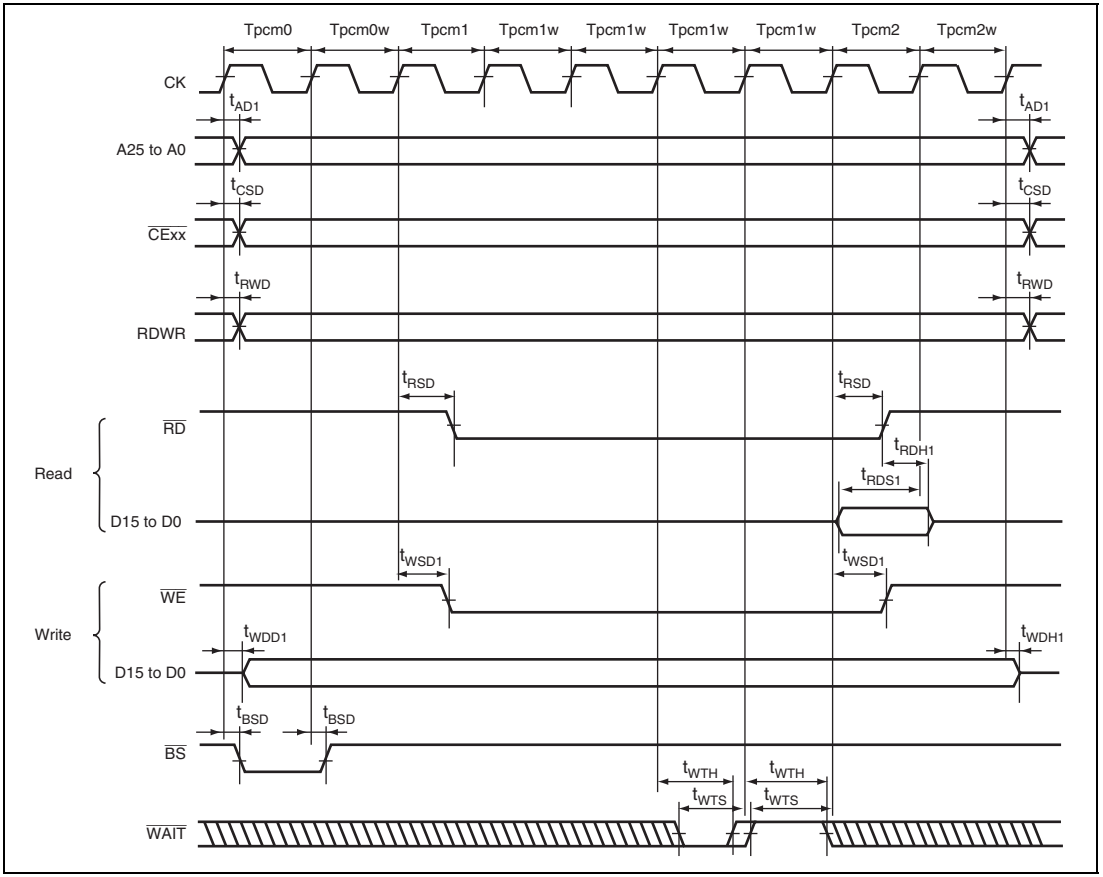


Figure 28.42 PCMCIA Memory Card Interface Bus Timing (TED = 2.5 Cycles, TEH = 1.5 Cycles, One External Wait Cycle)

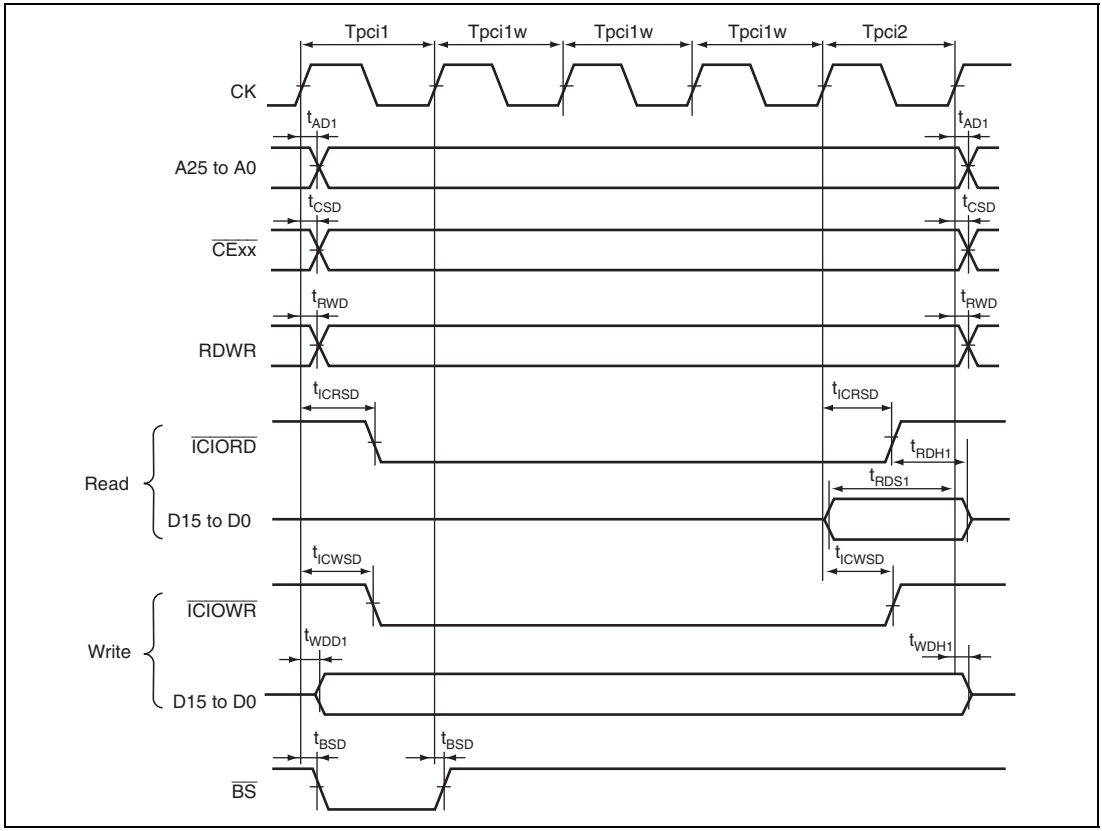


Figure 28.43 PCMCIA I/O Card Interface Bus Timing

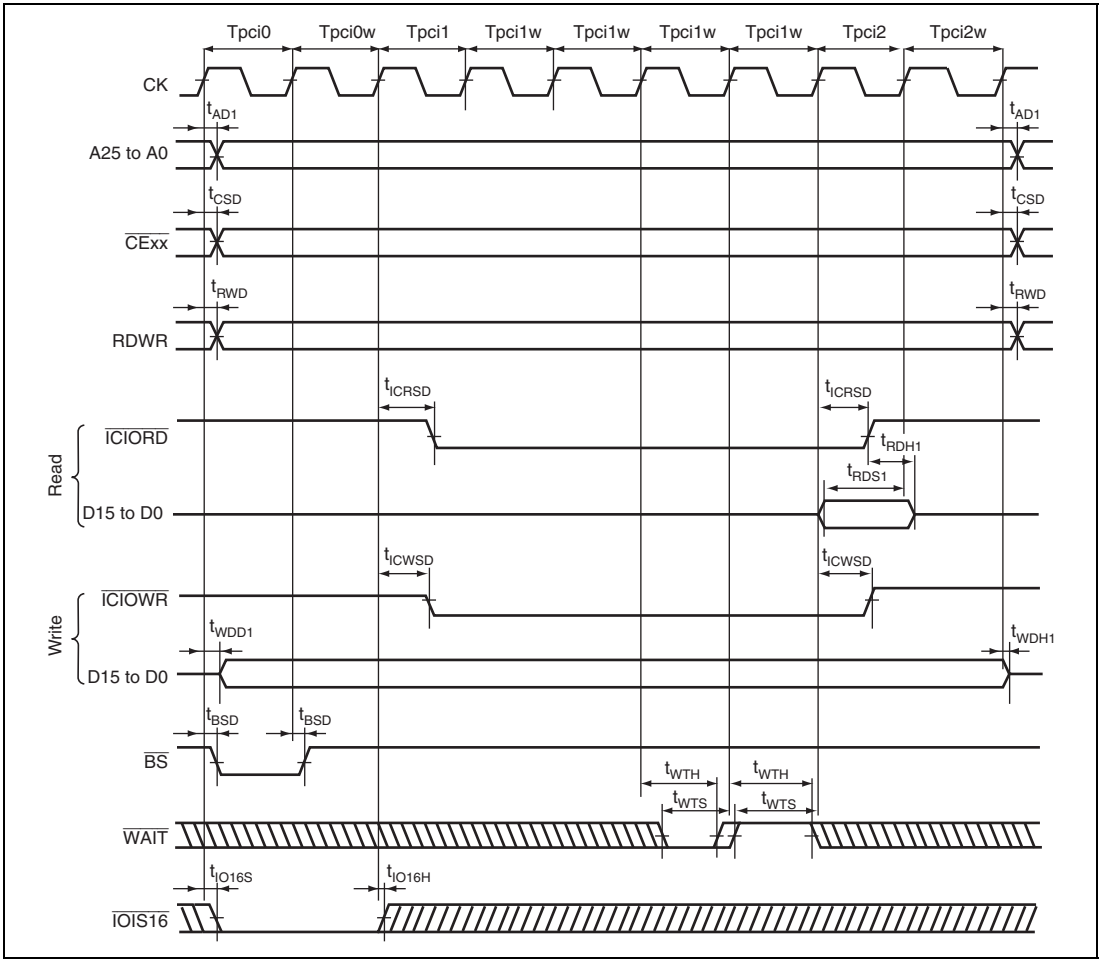


Figure 28.44 PCMCIA I/O Card Interface Bus Timing (TED = 2.5 Cycles, TEH = 1.5 Cycles, One External Wait Cycle)

28.3.4 Direct Memory Access Controller (DMAC) Timing

Table 28.9 Direct Memory Access Controller (DMAC) Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Max.	Unit	Reference Figure
DREQ setup time	t_{DRQS}	20	—	ns	Figure 28.45
DREQ hold time	t_{DRQH}	20	—	ns	

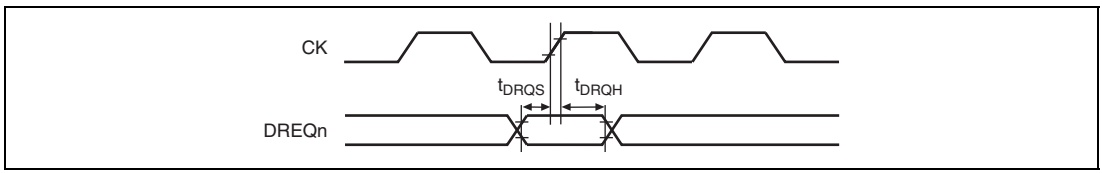


Figure 28.45 DREQ Input Timing

28.3.5 Multi Function Timer Pulse Unit 2 (MTU2) Timing

Table 28.10 Multi Function Timer Pulse Unit 2 (MTU2) Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Max.	Unit	Reference Figure
Output compare output delay time	t_{TOCD}	—	50	ns	Figure 28.46
Input capture input setup time	t_{TICS}	20	—	ns	
Input capture input pulse width (single edge)	t_{TICW}	1.5	—	t_{MPcyc}	
Input capture input pulse width (both edges)	t_{TICW}	2.5	—	t_{MPcyc}	
Timer input setup time	t_{TCKS}	20	—	ns	Figure 28.47
Timer clock pulse width (single edge)	$t_{TCKWH/L}$	1.5	—	t_{MPcyc}	
Timer clock pulse width (both edges)	$t_{TCKWH/L}$	2.5	—	t_{MPcyc}	
Timer clock pulse width (phase counting mode)	$t_{TCKWH/L}$	2.5	—	t_{MPcyc}	

Note: t_{MPcyc} indicates the MTU2 clock (MPϕ) cycle.

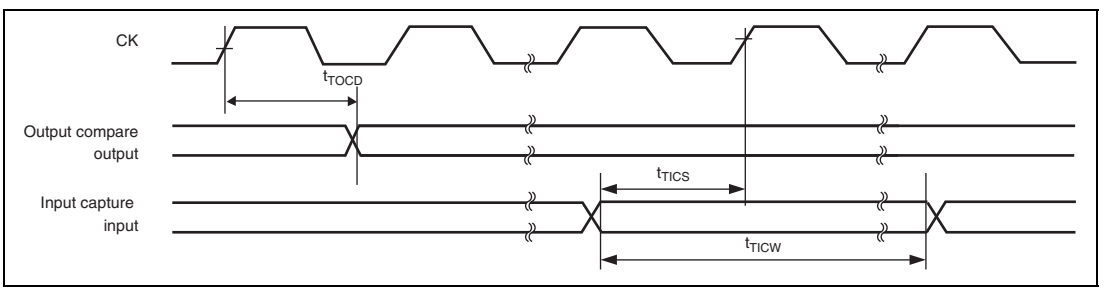


Figure 28.46 MTU2 Input/Output Timing

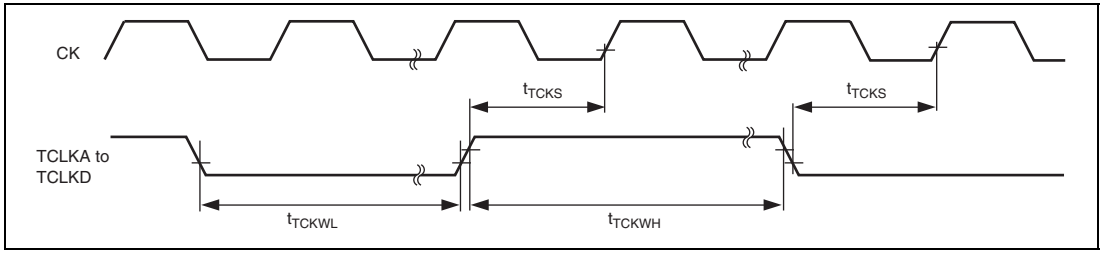


Figure 28.47 MTU2 Clock Input Timing

28.3.6 Multi Function Timer Pulse Unit 2S (MTU2S) Timing

Table 28.11 Multi Function Timer Pulse Unit 2S (MTU2S) Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Max.	Unit	Reference Figure
Output compare output delay time	t_{TOCD}	—	50	ns	Figure 28.48
Input capture input setup time	t_{TICS}	20	—	ns	
Input capture input pulse width (single edge)	t_{TICW}	1.5	—	t_{MIcyc}	
Input capture input pulse width (both edges)	t_{TICW}	2.5	—	t_{MIcyc}	

Note: t_{MIcyc} indicates the MTU2S clock (MIϕ) cycle.

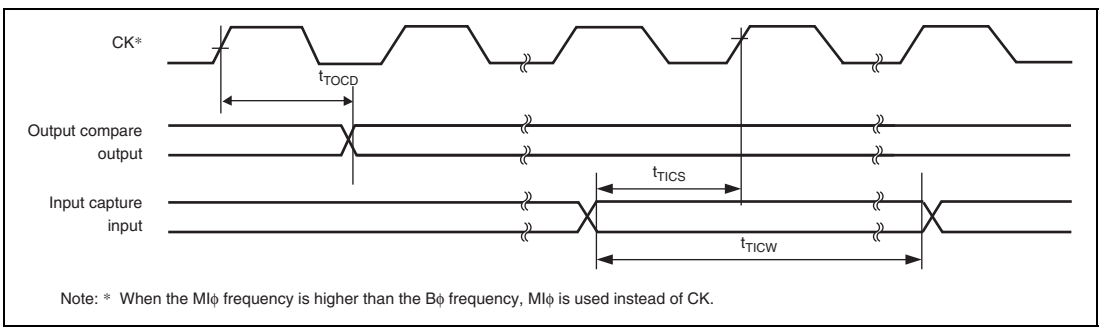


Figure 28.48 MTU2S Input/Output Timing

28.3.7 I/O Port Timing

Table 28.12 I/O Port Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Max.	Unit	Reference Figure
Port output data delay time	t_{PWD}	—	50	ns	Figure 28.49
Port input hold time	t_{PRH}	20	—	ns	
Port input setup time	t_{PRS}	20	—	ns	

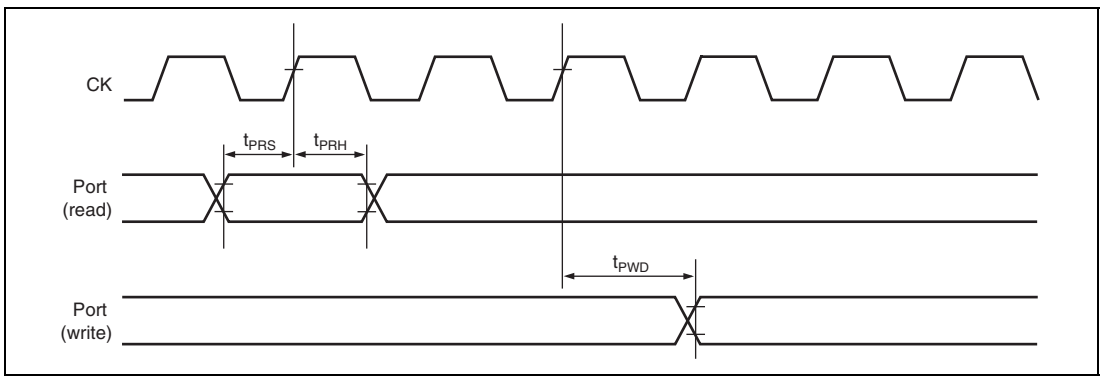


Figure 28.49 I/O Port Input/Output Timing

28.3.8 Watchdog Timer (WDT) Timing

Table 28.13 Watchdog Timer (WDT) Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Max.	Unit	Reference Figure
$\overline{\text{WDTOVF}}$ delay time	t_{WOVD}	—	50	ns	Figure 28.50

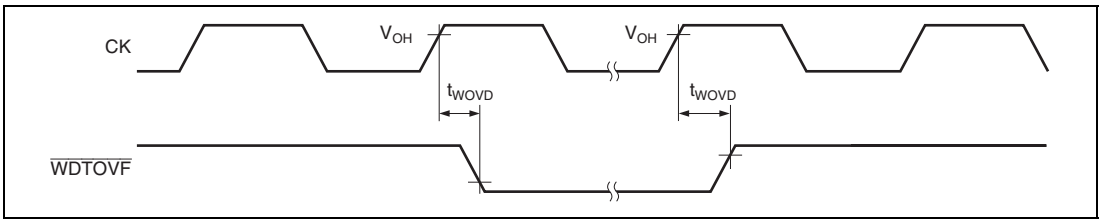


Figure 28.50 WDT Timing

28.3.9 Serial Communication Interface (SCI) Timing

Table 28.14 Serial Communication Interface (SCI) Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item		Symbol	Min.	Max.	Unit	Reference Figure
Input clock cycle (asynchronous)		t_{scyc}	4	—	t_{pcyc}	Figure 28.51
Input clock cycle (clock synchronous)		t_{scyc}	6	—	t_{pcyc}	
Input clock pulse width		t_{sckw}	0.4	0.6	t_{scyc}	
Input clock rise time		t_{sckr}	—	1.5	t_{pcyc}	
Input clock fall time		t_{sckf}	—	1.5	t_{pcyc}	
Transmit data delay time	Asynchronous	t_{TXD}	—	4 t_{pcyc} + 10	ns	Figure 28.52
Receive data setup time		t_{RXS}	4 t_{pcyc}	—	ns	
Receive data hold time		t_{RXH}	4 t_{pcyc}	—	ns	
Transmit data delay time	Clock synchronous	t_{TXD}	—	3 t_{pcyc} + 10	ns	
Receive data setup time		t_{RXS}	2 t_{pcyc} + 50	—	ns	
Receive data hold time		t_{RXH}	2 t_{pcyc}	—	ns	

Note: t_{pcyc} indicates the peripheral clock (Pϕ) cycle.

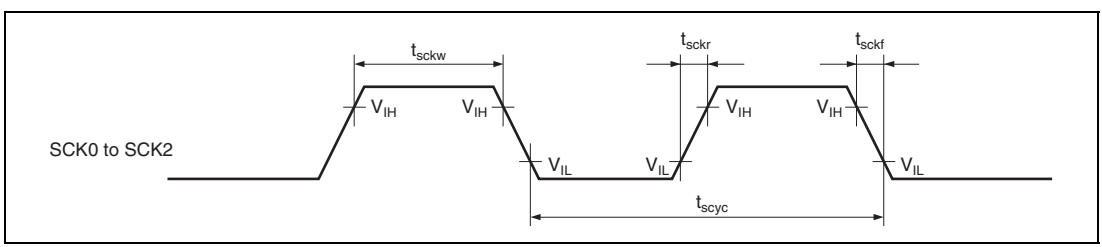


Figure 28.51 Input Clock Timing

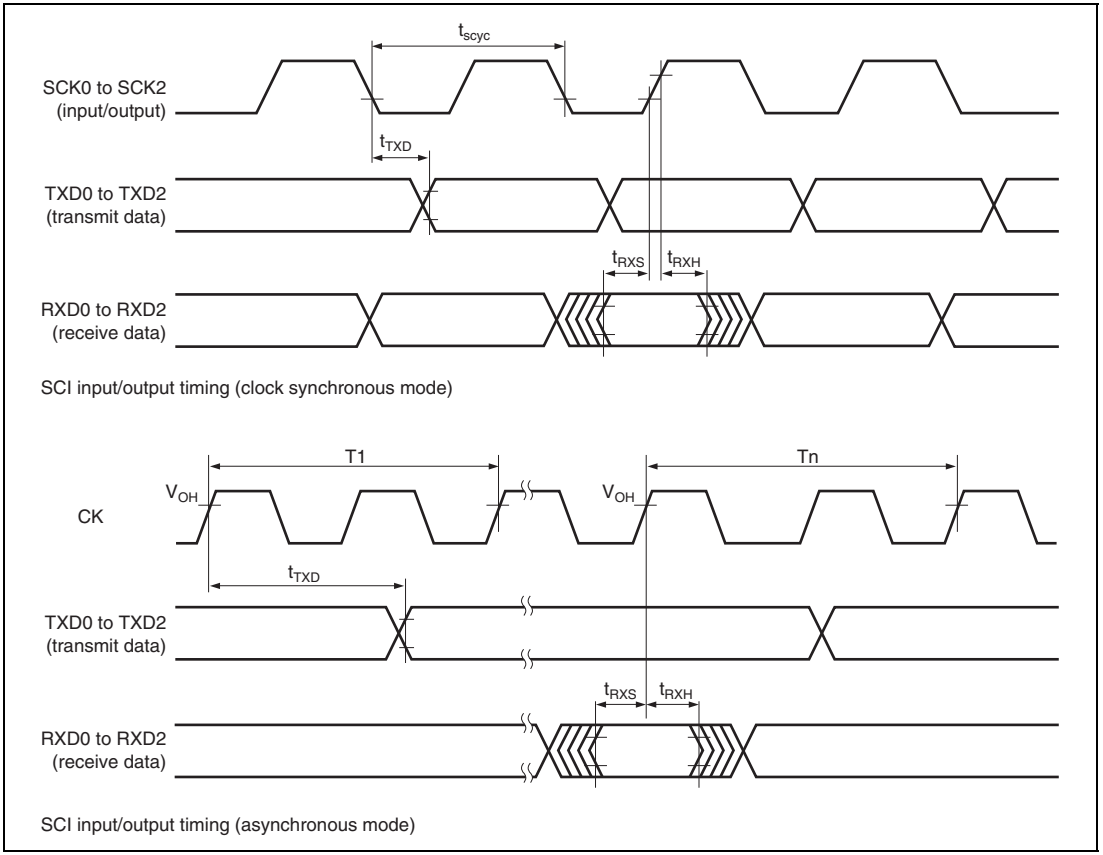


Figure 28.52 SCI Input/Output Timing

28.3.10 Serial Communication Interface with FIFO (SCIF) Timing

Table 28.15 Serial Communication Interface with FIFO (SCIF) Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item		Symbol	Min.	Max.	Unit	Reference Figure
Input clock cycle (asynchronous)		t_{scyc}	4	—	t_{pcyc}	Figure 28.53
Input clock cycle (clock synchronous)		t_{scyc}	6	—	t_{pcyc}	
Input clock pulse width		t_{sckw}	0.4	0.6	t_{scyc}	
Input clock rise time		t_{sckr}	—	1.5	t_{pcyc}	
Input clock fall time		t_{sckf}	—	1.5	t_{pcyc}	
Transmit data delay time	Asynchronous	t_{TXD}	—	4 t_{pcyc} + 10	ns	Figure 28.54
Receive data setup time		t_{RXS}	4 t_{pcyc}	—	ns	
Receive data hold time		t_{RXH}	4 t_{pcyc}	—	ns	
Transmit data delay time	Clock synchronous	t_{TXD}	—	3 t_{pcyc} + 10	ns	
Receive data setup time		t_{RXS}	2 t_{pcyc} + 50	—	ns	
Receive data hold time		t_{RXH}	2 t_{pcyc}	—	ns	
$\overline{RTS}$ delay time	Asynchronous	t_{RTSD}	—	4 t_{pcyc} + 10	ns	
$\overline{CTS}$ setup time		t_{CTSS}	4 t_{pcyc}	—	ns	
$\overline{CTS}$ hold time		t_{CTSH}	4 t_{pcyc}	—	ns	

Note: t_{pcyc} indicates the peripheral clock (Pϕ) cycle.

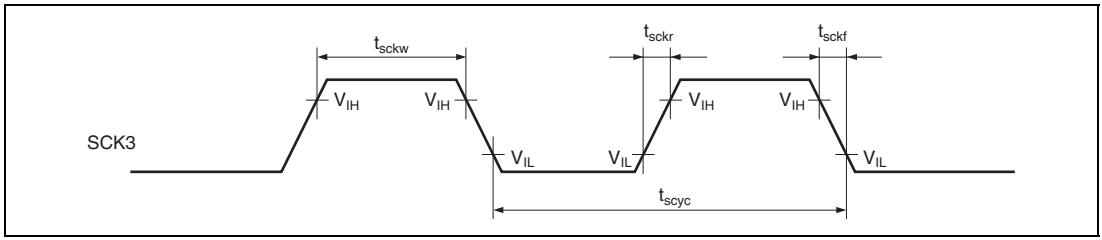


Figure 28.53 Input Clock Timing

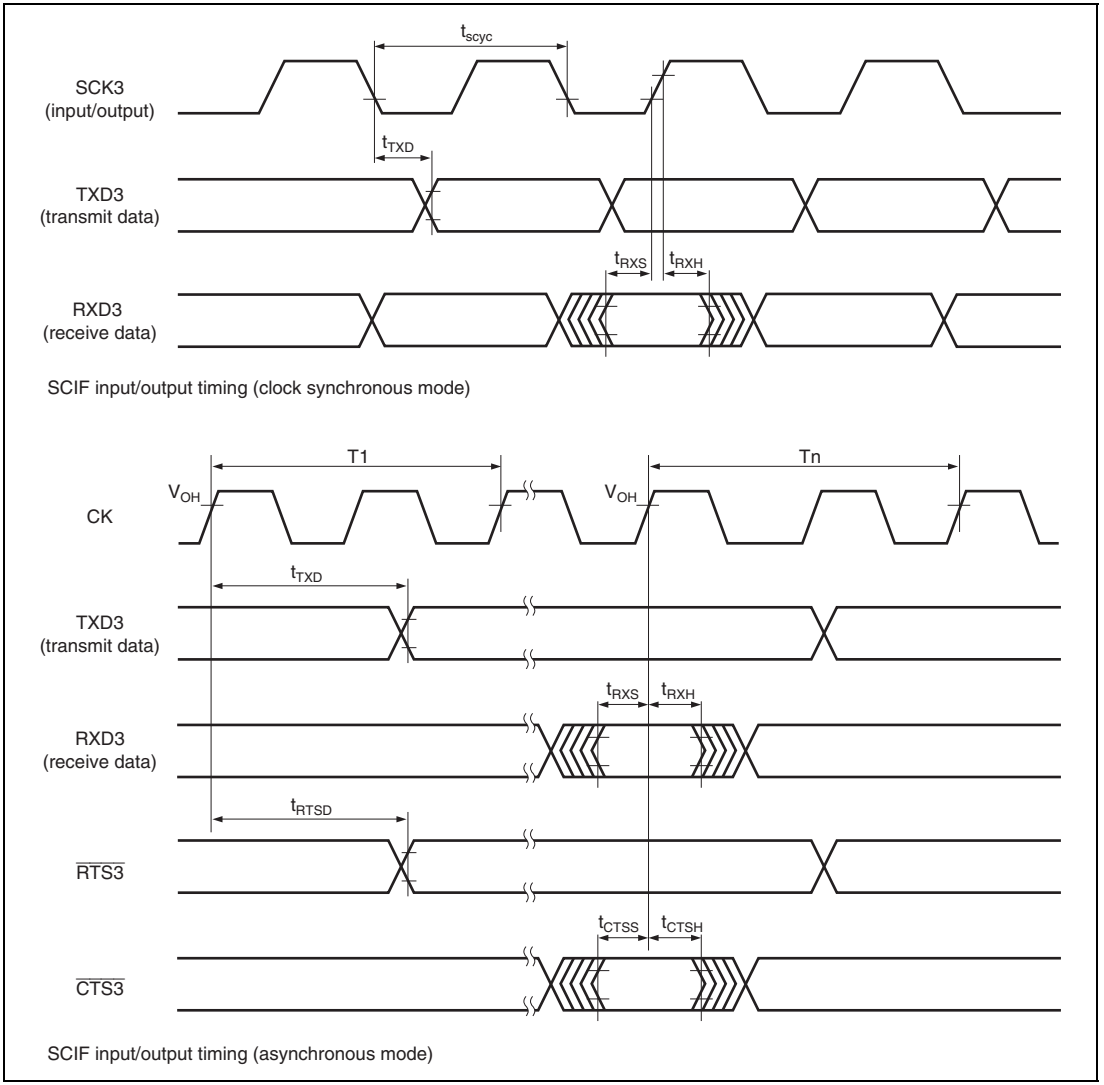


Figure 28.54 SCIF Input/Output Timing

28.3.11 Synchronous Serial Communication Unit (SSU) Timing

Table 28.16 Synchronous Serial Communication Unit (SSU) Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item		Symbol	Min.	Max.	Unit	Reference Figure
Clock cycle	Master	t_{SUcyc}	4	256	t_{pcyc}	Figures 28.55 to 28.58
	Slave		4	256		
Clock high pulse width	Master	t_{HI}	60	—	ns	
	Slave		60	—		
Clock low pulse width	Master	t_{LO}	60	—	ns	
	Slave		60	—		
Clock rise time		t_{RISE}	—	20	ns	
Clock fall time		t_{FALL}	—	20	ns	
Data input setup time	Master	t_{SU}	25	—	ns	
	Slave		30	—		
Data input hold time	Master	t_{H}	10	—	ns	
	Slave		10	—		
$\overline{SCS}$ setup time	Master	t_{LEAD}	1.5	—	t_{pcyc}	
	Slave		1.5	—		
$\overline{SCS}$ hold time	Master	t_{LAG}	1.5	—	t_{pcyc}	
	Slave		1.5	—		
Data output delay time	Master	t_{OD}	—	40	ns	
	Slave		—	40		
Data output hold time	Master	t_{OH}	30	—	ns	
	Slave		30	—		
Continuous transmission delay time	Master	t_{TD}	1.5	—	t_{pcyc}	
	Slave		1.5	—		
Slave access time		t_{SA}	—	1	t_{pcyc}	Figures 28.57, 28.58
Slave out release time		t_{REL}	—	1	t_{pcyc}	

Note: t_{pcyc} indicates the peripheral clock (Pϕ) cycle.

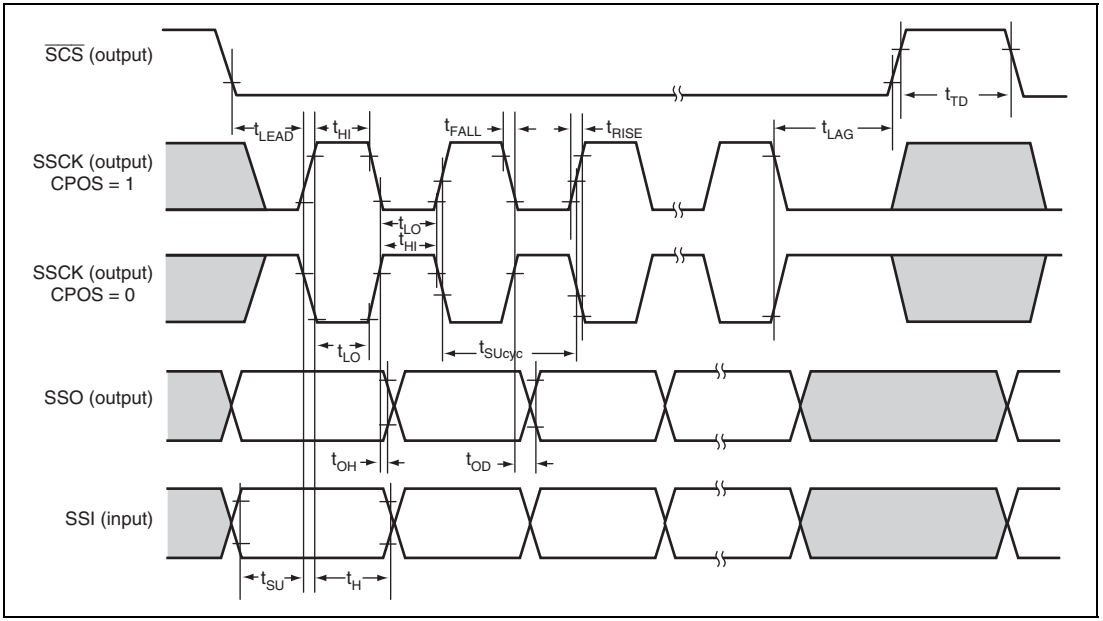


Figure 28.55 SSU Timing (Master, CPHS = 1)

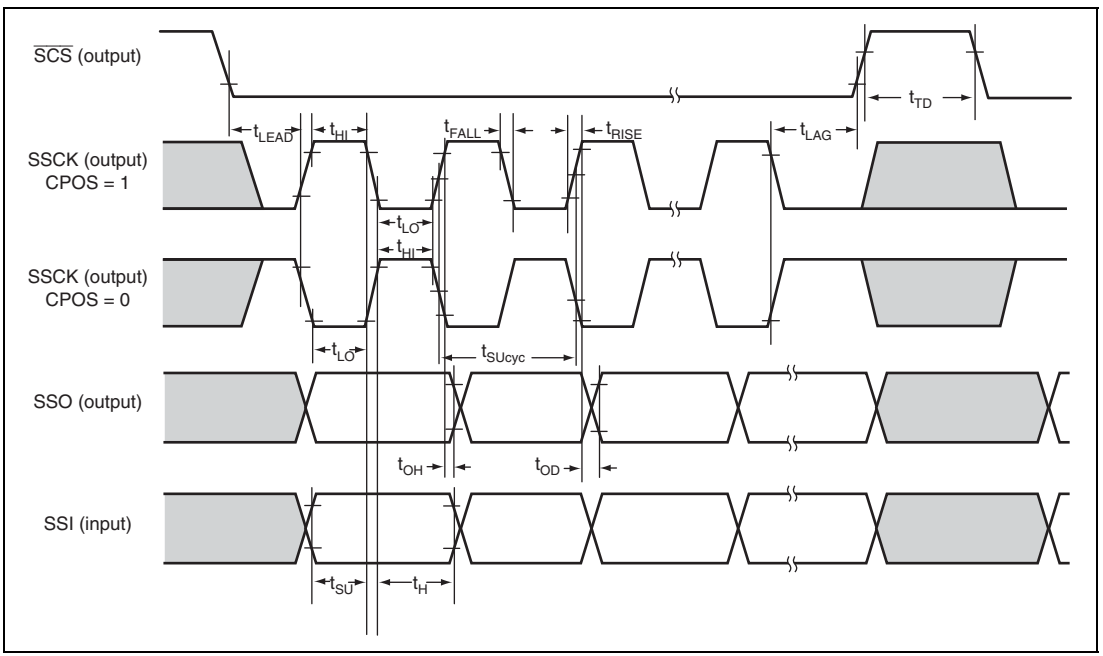


Figure 28.56 SSU Timing (Master, CPHS = 0)

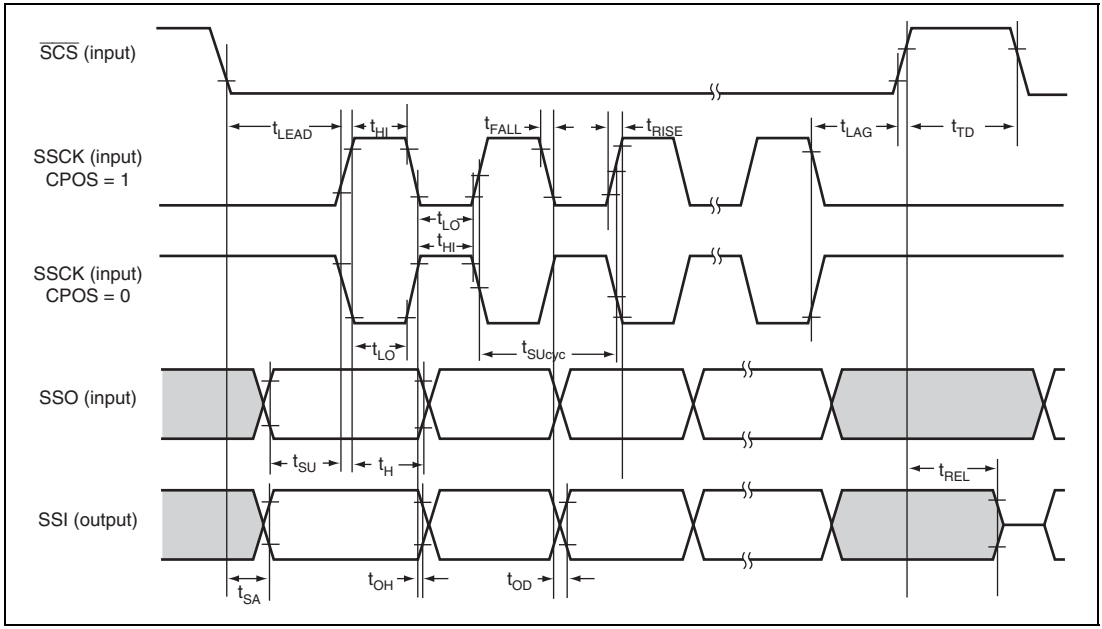


Figure 28.57 SSU Timing (Slave, CPHS = 1)

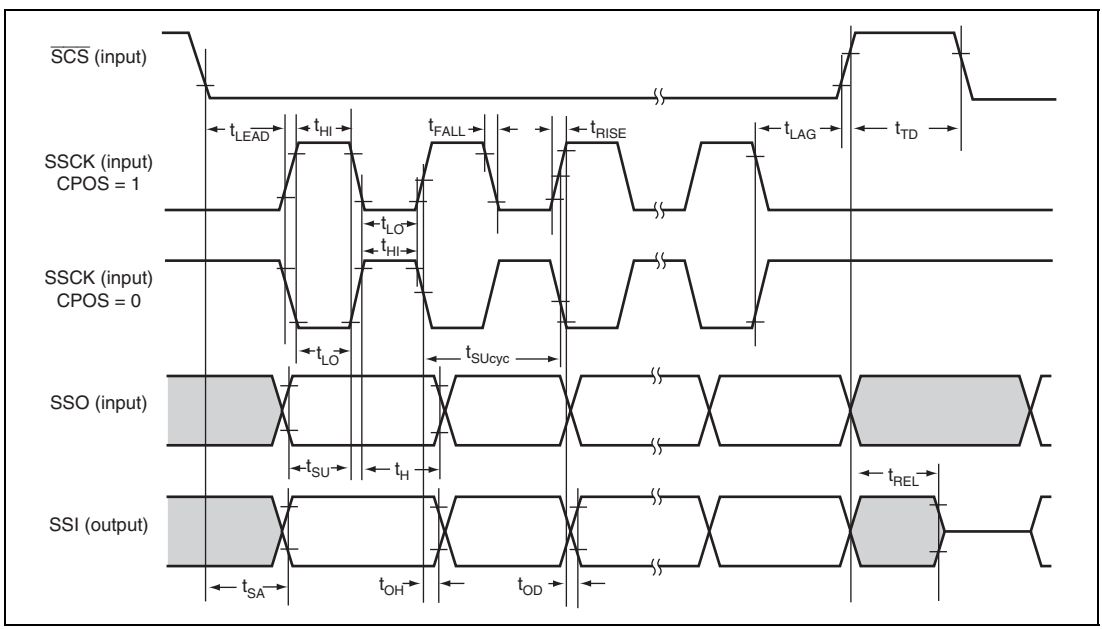


Figure 28.58 SSU Timing (Slave, CPHS = 0)

28.3.12 Port Output Enable (POE) Timing

Table 28.17 Port Output Enable (POE) Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Max.	Unit	Reference Figure
$\overline{POE}$ input setup time	t_{POES}	50	—	ns	Figure 28.59
$\overline{POE}$ input pulse width	t_{POEW}	1.5	—	t_{pcyc}	

Note: t_{pcyc} indicates the peripheral clock (Pϕ) cycle.

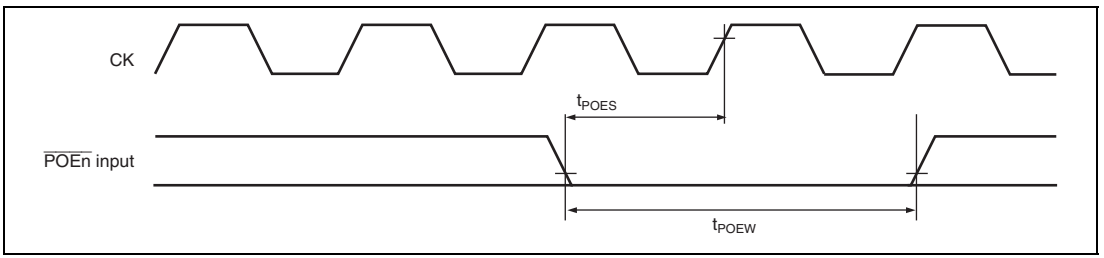


Figure 28.59 $\overline{POE}$ Input Timing

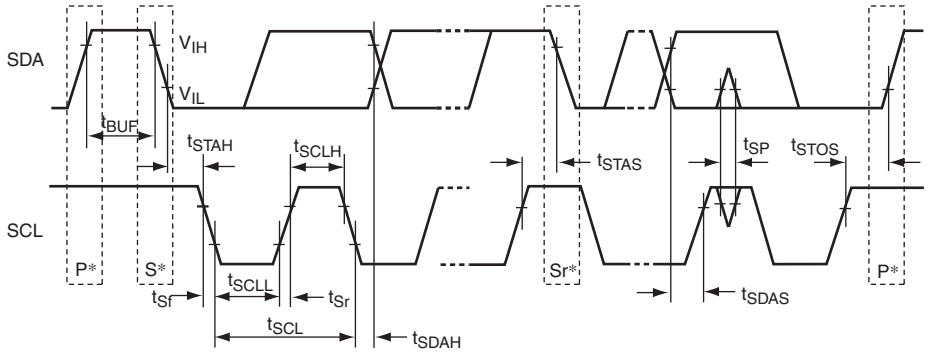
28.3.13 I²C Bus Interface 2 (IIC2) Timing

Table 28.18 I²C Bus Interface 2 (IIC2) Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Typ.	Max.	Unit	Reference Figure
SCL input cycle time	t_{SCL}	12 t_{pcyc} + 600	—	—	ns	Figure 28.60
SCL input high pulse width	t_{SCLH}	3 t_{pcyc} + 300	—	—	ns	
SCL input low pulse width	t_{SCLL}	5 t_{pcyc} + 300	—	—	ns	
SCL and SDA input fall time	t_{Sf}	—	—	300	ns	
SCL and SDA input spike pulse removal time	t_{SP}	—	—	1 t_{pcyc}	ns	
SDA input bus free time	t_{BUF}	5	—	—	t_{pcyc}	
Start condition input hold time	t_{STAH}	3	—	—	t_{pcyc}	
Repeated start condition input setup time	t_{STAS}	3	—	—	t_{pcyc}	
Halt condition input setup time	t_{STOS}	3	—	—	t_{pcyc}	
Data input setup time	t_{SDAS}	1 t_{pcyc} + 20	—	—	ns	
Data input hold time	t_{SDAH}	0	—	—	ns	
SCL and SDA capacity load	C_b	0	—	400	pF	
SCL and SDA output fall time	t_{Sf}	—	—	250	ns	

Note: t_{pcyc} indicates the peripheral clock (Pϕ) cycle.



[Legend]
S: Start condition
P: Stop condition
Sr: Repeated start condition

Figure 28.60 I^2C2 Input/Output Timing

28.3.14 UBC Trigger Timing

Table 28.19 UBC Trigger Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Max.	Unit	Reference Figure
$\overline{\text{UBCTRG}}$ delay time	t_{UBCTGD}	—	150	ns	Figure 28.61

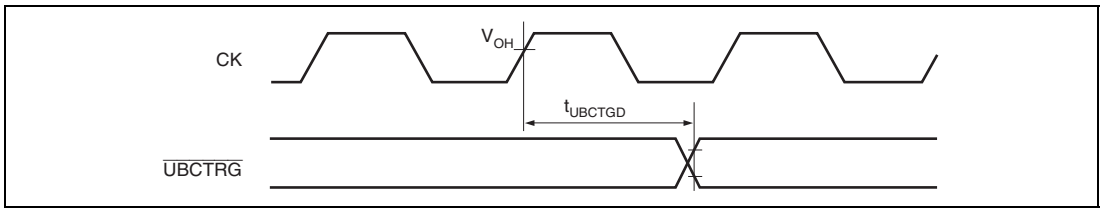


Figure 28.61 UBC Trigger Timing

28.3.15 A/D Converter Timing

Table 28.20 A/D Converter Timing

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Symbol	Min.	Typ.	Max.	Unit	Figure
External trigger input start delay time	t_{TRGS}	25	—	—	ns	Figure 28.62

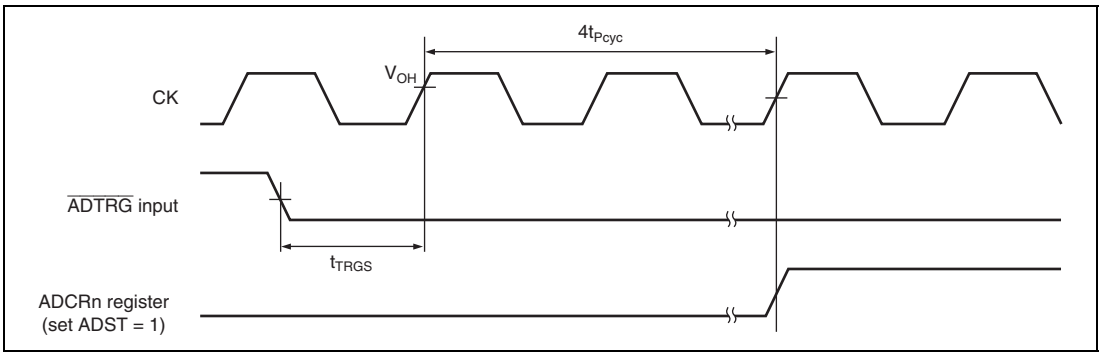
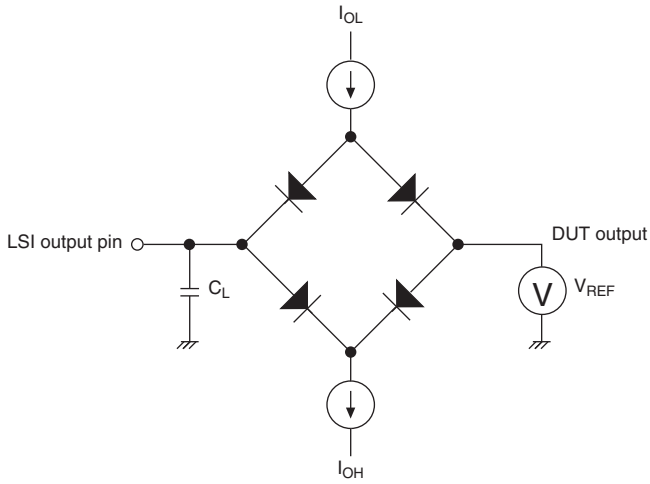


Figure 28.62 External Trigger Input Timing

28.3.16 AC Characteristics Measurement Conditions

- Input signal level: V_{IL} (Max.)/V_{IH} (Min.)
- Output signal reference level: High level: 2.0 V, Low level: 0.8 V



Notes: 1. C_L is the total value that includes the capacitance of measurement tools. Each pin is set as follows:
20pF: CK
30pF: All other output pins
2. Test conditions include I_{OL} = 1.6 mA and I_{OH} = -200 µA.

Figure 28.63 Output Load Circuit

28.4 A/D Converter Characteristics

Table 28.21 A/D Converter Characteristics

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

Item	Min.	Typ.	Max.	Unit
Resolution	10	10	10	bit
A/D conversion time	2.0	—	—	μs
Analog input capacitance	—	—	20	pF
Permitted analog signal source impedance	—	—	1*[1]/3*[2]	kΩ
Non-linear error	—	—	±3.0*[1]/±5.0*[2]	LSB
Offset error	—	—	±3.0*[1]/±5.0*[2]	LSB
Full-scale error	—	—	±3.0*[1]/±5.0*[2]	LSB
Quantization error	—	—	±0.5	LSB
Absolute error	—	—	±4.0*[1]/±6.0*[2]	LSB

Notes: 1. It is assumed that A/D conversion time ≥ 4.0 μs.
2. It is assumed that A/D conversion time < 4.0 μs.

28.5 Flash Memory Characteristics

Table 28.22 Flash Memory Characteristics

Conditions: V_{CC} = 3.0 V to 3.6 V or 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, AV_{ref} = 4.0 V to AV_{CC}, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +85°C (consumer applications), T_a = –40°C to +85°C (industrial applications)

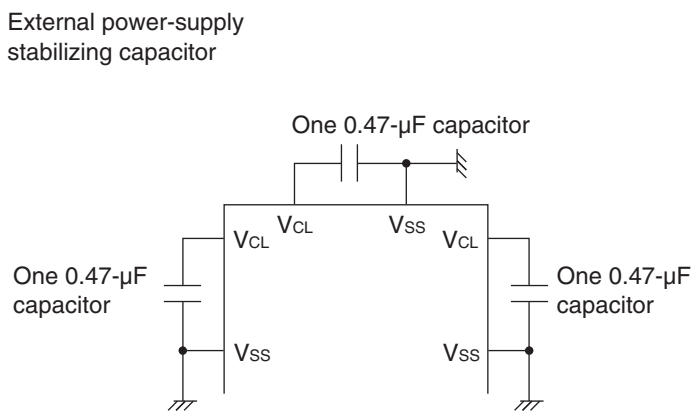
Item	Symbol	Min.	Typ.	Max.	Unit
Programming time*1*2*4	t_P	—	1	20	ms/128 bytes
Erase time*1*2*4	t_E	—	40	280	ms/4-Kbyte block
		—	300	1500	ms/32-Kbyte block
		—	600	3000	ms/64-Kbyte block
Programming time (total)*1*2*4	Σt_P	—	4.6	24	s/512 Kbytes
		—	2.3	12	s/256 Kbytes
Erase time (total)*1*2*4	Σt_E	—	4.6	24	s/512 Kbytes
		—	2.3	12	s/256 Kbytes
Programming and erase time (total)*1*2*4	Σt_{PE}	—	9.2	48	s/512 Kbytes
		—	4.6	24	s/256 Kbytes
Reprogramming count	N_{WEC}	500*3	—	—	Times

Notes: 1. Programming and erase time vary depending on the data.
2. Programming and erase time do not include data transfer time.
3. The minimum number of times for which all characteristics are guaranteed after reprogramming (guaranteed for once to the minimum number of reprogramming times).
4. These characteristics only apply when reprogramming is performed within the range of minimum number of times.

28.6 Usage Note

28.6.1 Notes on Connecting V_{CL} Capacitor

This LSI includes an internal step-down circuit to automatically reduce the internal power supply voltage to an appropriate level. Between this internal stepped-down power supply (V_{CL} pin) and the V_{SS} pin, a capacitor (0.47 μF) for stabilizing the internal voltage needs to be connected. Connection of the external capacitor is shown in figure 28.64. The external capacitor should be located near the pin. Do not apply any power supply voltage to the V_{CL} pin.



Note: Do not apply any power supply voltage to the V_{CL} pin.
Use multilayer ceramic capacitors (one 0.47-μF capacitor for each V_{CL} pin), which should be located near the pin.

Figure 28.64 Connection of V_{CL} Capacitor

Appendix

A. Pin States

Pin initial states differ according to MCU operating modes. Refer to section 21, Pin Function Controller (PFC), for details.

Table A.1 Pin States (SH7083)

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 8 bits	Expansion without ROM 16 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
Clock	CK	O			Z	O	Z	H*1	O	O	O	O
	XTAL	O				O	L	L	O	O	O	O
	EXTAL	I				I	Z	I	I	I	I	I
System control	$\overline{RES}$	I				I	I	I	I	I	I	I
	$\overline{MRES}$	Z				I	Z	I*5	I	I	I*5	I
	$\overline{WDTOVF}$	O*6				O	O	O	O	O	O	O
	$\overline{BREQ}$	Z				I	Z	Z	I	I	I	I
	$\overline{BACK}$	Z				O	Z	Z	O	L	O	O
Operating mode control	MD0, MD1	I				I	I	I	I	I	I	I
	$\overline{ASEMD0}$	I*7				I*7	I*7	I*7	I*7	I*7	I*7	I*7
	FWE	I				I	I	I	I	I	I	I
Interrupt	NMI	I				I	I	I	I	I	I	I
	IRQ0 to IRQ7	Z				I	Z	I	I	I	I	I
	$\overline{IRQOUT}$	Z				O	Z	Z (MZIZEL in HCPCR = 0) H*1 (MZIZEL in HCPCR = 1)	O	O	O*5	O

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 8 bits	Expansion without ROM 16 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
Address bus	A0 to A17	O		Z		O	Z	Z*3	O	Z	O	O
	A18 to A24	Z				O	Z	Z*3	O	Z	O	O
Data bus	D0 to D8, D10	Z				I/O	Z	Z	I/O	Z	I/O	I/O
	D9, D11 to D15	Z				I/O	Z	Z	I/O	Z	I/O*4	I/O
Bus control	$\overline{\text{WAIT}}$	Z				I	Z	Z	I	Z	I	I
	$\overline{\text{CS0}}$	H		Z		O	Z	Z*3	O	Z	O	O
	$\overline{\text{CS3}}$, $\overline{\text{CS7}}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{\text{BS}}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{\text{RASL}}$	Z				O	Z	Z*2	O	Z*2	O	O
	$\overline{\text{CASL}}$	Z				O	Z	Z*2	O	Z*2	O	O
	DQMLU, DQMLL	Z				O	Z	Z*3	O	Z	O	O
	RDWR	Z				O	Z	Z*3	O	Z	O	O
	$\overline{\text{RD}}$	H		Z		O	Z	Z*3	O	Z	O	O
	$\overline{\text{WRH}}$, $\overline{\text{WRL}}$	H		Z		O	Z	Z*3	O	Z	O	O
	CKE (PE15)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*2 (MZIZEL in HCPCR = 1)	O	Z*2	O*5	O
	CKE (PA9)	Z				O	Z	Z*2	O	Z*2	O	O
DMAC	DREQ0, DREQ1	Z				I	Z	Z	I	I	I	I
	DACK0, DACK1	Z				O	Z	Z (MZIZEL in HCPCR = 0) O*1 (MZIZEL in HCPCR = 1)	O	O	O*5	O

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 8 bits	Expansion without ROM 16 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
DMAC	TEND0, TEND1	Z				O	Z	O*1	O	O	O	O
MTU2	TCLKB to TCLKD	Z				I	Z	Z	I	I	I	I
	TIOC0A to TIOC0D	Z				I/O	Z	K*1	I/O	I/O	I/O	Z
	TIOC1A	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC2A, TIOC2B	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC3A, TIOC3C	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC4A to TIOC4D	Z				I/O	Z	Z (MZIZEL in HCPCR = 0) K*1 (MZIZEL in HCPCR = 1)	I/O	I/O	I/O*5	Z
	TIC5U, TIC5V, TIC5W	Z				I	Z	Z	I	I	I	I
MTU2S	TIOC3AS, TIOC3CS	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC3BS, TIOC3DS	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K*1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O*4	Z
	TIOC4AS to TIOC4DS	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K*1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O*4	Z

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
		Expansion without ROM		Expansion with ROM	Single-chip		Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
Type	Pin Name	8 bits	16 bits			Manual						
MTU2S	TIC5US, TIC5VS, TIC5WS	Z				I	Z	Z	I	I	I	I
POE	$\overline{POE0}$, $\overline{POE2}$ to $\overline{POE4}$, $\overline{POE6}$ to $\overline{POE8}$	Z				I	Z	Z	I	I	I	I
SCI	SCK0 to SCK2	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	RXD0 to RXD2	Z				I	Z	Z	I	I	I	I
	TXD0 to TXD2	Z				O	Z	O*[1]	O	O	O	O
SCIF	SCK3	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	RXD3	Z				I	Z	Z	I	I	I	I
	TXD3	Z				O	Z	Z (MZIZEL in HCPCR = 0) O*[1] (MZIZEL in HCPCR = 1)	O	O	O*[5]	O
SSU	SSCK	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	$\overline{SCS}$	Z				I/O	Z	Z	I/O	I/O	I/O*[5]	I/O
	SSI	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	SSO	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
UBC	$\overline{UBCTRG}$	Z				O	Z	O*[1]	O	O	O	O
A/D Converter	AN0 to AN7	Z				I	Z	Z	I	I	I	I
	$\overline{ADTRG}$	Z				I	Z	Z	I	I	I	I
I/O Port	PA3 to PA5, PA7 to PA10, PA12 to PA15	Z				I/O	Z	K*[1]	I/O	I/O	I/O	I/O
	PB0 to PB2, PB4 to PB9	Z				I/O	Z	K*[1]	I/O	I/O	I/O	I/O
	PC0 to PC15	Z				I/O	Z	K*[1]	I/O	I/O	I/O	I/O

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 8 bits	Expansion without ROM 16 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
I/O Port	PD0 to PD8, PD10	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	PD9, PD11 to PD15	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K*1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O*4	Z
	PE0 to PE3	Z				I/O	Z	K*1	I/O	I/O	I/O	Z
	PE4, PE6 to PE8, PE10	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	PE12 to PE15	Z				I/O	Z	Z (MZIZEL in HCPCR = 0) K*1 (MZIZEL in HCPCR = 1)	I/O	I/O	I/O*5	Z
	PF0 to PF7	Z				I	Z	Z	I	I	I	I

[Legend]

I: Input

O: Output

H: High-level output

L: Low-level output

Z: High-impedance

K: Input pins become high-impedance, and output pins retain their state.

Notes: 1. Output pins become high-impedance when the HIZ bit in standby control register 6 (STBCR6) is set to 1.

2. Becomes output when the HIZCNT bit in the common control register (CMNCR) is set to 1.

3. Becomes output when the HIZMEM bit in the common control register (CMNCR) is set to 1.

4. Becomes high-impedance when the MZIZDL bit in the high-current port control register (HCPCR) is cleared to 0.

5. Becomes high-impedance when the MZIZEL bit in the high-current port control register (HCPCR) is cleared to 0.
6. Becomes input during a power-on reset. Pull-up to prevent erroneous operation. Pull-down with a resistance of at least 1 MΩ as required.
7. Pulled-up inside the LSI when there is no input.

Table A.2 Pin States (SH7084)

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 8 bits	Expansion without ROM 16 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
Clock	CK	O			Z	O	Z	H*1	O	O	O	O
	XTAL	O				O	L	L	O	O	O	O
	EXTAL	I				I	Z	I	I	I	I	I
System control	$\overline{RES}$	I				I	I	I	I	I	I	I
	$\overline{MRES}$	Z				I	Z	I*5	I	I	I*5	I
	$\overline{WDTOVF}$	O*6				O	O	O	O	O	O	O
	$\overline{BREQ}$	Z				I	Z	Z	I	I	I	I
	$\overline{BACK}$	Z				O	Z	Z	O	L	O	O
Operating mode control	MD0, MD1	I				I	I	I	I	I	I	I
	$\overline{ASEMD0}$	I*7				I*7	I*7	I*7	I*7	I*7	I*7	I*7
	FWE	I				I	I	I	I	I	I	I
Interrupt	NMI	I				I	I	I	I	I	I	I
	IRQ0 to IRQ7	Z				I	Z	I	I	I	I	I
	$\overline{IRQOUT}$	Z				O	Z	Z (MZIZEL in HCPCR = 0) H*1 (MZIZEL in HCPCR = 1)	O	O	O*5	O
Address bus	A0 to A17	O		Z		O	Z	Z*3	O	Z	O	O
	A18 to A25	Z				O	Z	Z*3	O	Z	O	O
Data bus	D0 to D8, D10	Z				I/O	Z	Z	I/O	Z	I/O	I/O
	D9, D11 to D15	Z				I/O	Z	Z	I/O	Z	I/O*4	I/O
Bus control	$\overline{WAIT}$	Z				I	Z	Z	I	Z	I	I
	$\overline{CS0}$, $\overline{CS1}$	H		Z		O	Z	Z*3	O	Z	O	O
	$\overline{CS2}$ to $\overline{CS7}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{BS}$	Z				O	Z	Z*3	O	Z	O	O

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 8 bits	Expansion without ROM 16 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
Bus control	$\overline{\text{RASL}}$	Z				O	Z	Z*2	O	Z*2	O	O
	$\overline{\text{CASL}}$	Z				O	Z	Z*2	O	Z*2	O	O
	DQMLU, DQMLL	Z				O	Z	Z*3	O	Z	O	O
	$\overline{\text{AH}}$ (PA16)	Z				O	Z	Z*3	O	Z	O	O
	$\overline{\text{AH}}$ (PE14)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*3 (MZIZEL in HCPCR = 1)	O	Z	O*5	O
	RDWR	Z				O	Z	Z*3	O	Z	O	O
	$\overline{\text{RD}}$	H	Z			O	Z	Z*3	O	Z	O	O
	$\overline{\text{WRH}}$, $\overline{\text{WRL}}$	H	Z			O	Z	Z*3	O	Z	O	O
	CKE (PE15)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*2 (MZIZEL in HCPCR = 1)	O	Z*2	O*5	O
	CKE (PA9/PA16)	Z				O	Z	Z*2	O	Z*2	O	O
DMAC	DREQ0, DREQ1	Z				I	Z	Z	I	I	I	I
	DACK0, DACK1	Z				O	Z	Z (MZIZEL in HCPCR = 0) O*1 (MZIZEL in HCPCR = 1)	O	O	O*5	O
	TEND0, TEND1	Z				O	Z	O*1	O	O	O	O

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 8 bits	Expansion without ROM 16 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
MTU2	TCLKA to TCLKD	Z				I	Z	Z	I	I	I	I
	TIOC0A to TIOC0D	Z				I/O	Z	K*1	I/O	I/O	I/O	Z
	TIOC1A, TIOC1B	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC2A, TIOC2B	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC3A, TIOC3C	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC3B, TIOC3D	Z				I/O	Z	Z (MZIZEL in HCPCR = 0) K*1 (MZIZEL in HCPCR = 1)	I/O	I/O	I/O*5	Z
	TIOC4A to TIOC4D	Z				I/O	Z	Z (MZIZEL in HCPCR = 0) K*1 (MZIZEL in HCPCR = 1)	I/O	I/O	I/O*5	Z
	TIC5U, TIC5V, TIC5W	Z				I	Z	Z	I	I	I	I
MTU2S	TIOC3AS, TIOC3CS	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC3BS, TIOC3DS	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K*1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O*4	Z

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
		Expansion without ROM		Expansion with ROM	Single-chip		Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
Type	Pin Name	8 bits	16 bits			Manual						
MTU2S	TIOC4AS to TIOC4DS	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K*1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O*4	Z
	TIC5US, TIC5VS, TIC5WS	Z				I	Z	Z	I	I	I	I
POE	$\overline{POE0}$ to $\overline{POE8}$	Z				I	Z	Z	I	I	I	I
SCI	SCK0 to SCK2	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	RXD0 to RXD2	Z				I	Z	Z	I	I	I	I
	TXD0 to TXD2	Z				O	Z	O*1	O	O	O	O
SCIF	SCK3 (PE6)	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	SCK3 (PE9)	Z				I/O	Z	Z	I/O	I/O	I/O*5	I/O
	RXD3 (PE4)	Z				I	Z	Z	I	I	I	I
	RXD3 (PE11)	Z				I	Z	Z	I	I	I*5	I
	TXD3 (PE5)	Z				O	Z	O*1	O	O	O	O
	TXD3 (PE12)	Z				O	Z	Z (MZIZEL in HCPCR = 0) O*1 (MZIZEL in HCPCR = 1)	O	O	O*5	O
	$\overline{RTS3}$	Z				O	Z	Z (MZIZEL in HCPCR = 0) O*1 (MZIZEL in HCPCR = 1)	O	O	O*5	O
	$\overline{CTS3}$	Z				I	Z	Z	I	I	I*5	I

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 8 bits	Expansion without ROM 16 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
SSU	SSCK	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	$\overline{SCS}$	Z				I/O	Z	Z	I/O	I/O	I/O※5	I/O
	SSI	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	SSO	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
IIC2	SCL	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	SDA	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
UBC	$\overline{UBCTRG}$	Z				O	Z	O※1	O	O	O	O
A/D Converter	AN0 to AN7	Z				I	Z	Z	I	I	I	I
	$\overline{ADTRG}$	Z				I	Z	Z	I	I	I	I
I/O Port	PA0 to PA17	Z				I/O	Z	K※1	I/O	I/O	I/O	I/O
	PB0 to PB9	Z				I/O	Z	K※1	I/O	I/O	I/O	I/O
	PC0 to PC15	Z				I/O	Z	K※1	I/O	I/O	I/O	I/O
	PD0 to PD8, PD10	Z				I/O	Z	K※1	I/O	I/O	I/O	I/O
	PD9, PD11 to PD15	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K※1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O※4	Z
	PE0 to PE3	Z				I/O	Z	K※1	I/O	I/O	I/O	Z
	PE4 to PE8, PE10	Z				I/O	Z	K※1	I/O	I/O	I/O	I/O
	PE9, PE11 to PE15	Z				I/O	Z	Z (MZIZEL in HCPCR= 0) K※1 (MZIZEL in HCPCR = 1)	I/O	I/O	I/O※5	Z
	PF0 to PF7	Z				I	Z	Z	I	I	I	I

[Legend]

I: Input

O: Output

H: High-level output

L: Low-level output

Z: High-impedance

K: Input pins become high-impedance, and output pins retain their state.

Notes:
1. Output pins become high-impedance when the HIZ bit in standby control register 6 (STBCR6) is set to 1.
2. Becomes output when the HIZCNT bit in the common control register (CMNCR) is set to 1.
3. Becomes output when the HIZMEM bit in the common control register (CMNCR) is set to 1.
4. Becomes high-impedance when the MZIZDL bit in the high-current port control register (HCPCR) is cleared to 0.
5. Becomes high-impedance when the MZIZEL bit in the high-current port control register (HCPCR) is cleared to 0.
6. Becomes input during a power-on reset. Pull-up to prevent erroneous operation. Pull-down with a resistance of at least 1 MΩ as required.
7. Pulled-up inside the LSI when there is no input.

Table A.3 Pin States (SH7085)

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
Clock	CK	O			Z	O	Z	H*1	O	O	O	O
	XTAL	O				O	L	L	O	O	O	O
	EXTAL	I				I	Z	I	I	I	I	I
System control	$\overline{\text{RES}}$	I				I	I	I	I	I	I	I
	$\overline{\text{MRES}}$	Z				I	Z	I*6	I	I	I*6	I
	$\overline{\text{WDTOVF}}$	O*7				O	O	O	O	O	O	O
	$\overline{\text{BREQ}}$	Z				I	Z	Z	I	I	I	I
	$\overline{\text{BACK}}$	Z				O	Z	Z	O	L	O	O
Operating mode control	MD0, MD1	I				I	I	I	I	I	I	I
	$\overline{\text{ASEMD0}}$	I*8				I*8	I*8	I*8	I*8	I*8	I*8	I*8
	FWE	I				I	I	I	I	I	I	I
Interrupt	NMI	I				I	I	I	I	I	I	I
	IRQ0 to IRQ7	Z				I	Z	I	I	I	I	I
	$\overline{\text{IRQOUT}}$ (PE15)	Z				O	Z	Z (MZIZEL in HCPCR = 0) H*1 (MZIZEL in HCPCR = 1)	O	O	O*6	O
	$\overline{\text{IRQOUT}}$ (PD30)	Z				O	Z	H*1	O	O	O	O
Address bus	A0 to A17	O		Z		O	Z	Z*3	O	Z	O	O
	A18 to A25	Z				O	Z	Z*3	O	Z	O	O
Data bus	D0 to D8, D10, D16 to D23, D30, D31	Z				I/O	Z	Z	I/O	Z	I/O	I/O
	D9, D11 to D15	Z				I/O	Z	Z	I/O	Z	I/O*5	I/O
	D24 to D29	Z				I/O	Z	Z	I/O	Z	I/O*4	I/O

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
Bus control	$\overline{\text{WAIT}}$	Z				I	Z	Z	I	Z	I	I
	$\overline{\text{CS0}}$, $\overline{\text{CS1}}$	H		Z		O	Z	Z*3	O	Z	O	O
	$\overline{\text{CS2}}$ (PA6), $\overline{\text{CS3}}$ (PA7), $\overline{\text{CS4}}$ to $\overline{\text{CS7}}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{\text{CS2}}$ (PD28), $\overline{\text{CS3}}$ (PD29)	Z				O	Z	Z (MZIZDH in HCPCR = 0) Z*3 (MZIZDH in HCPCR = 1)	O	Z	O*4	O
	$\overline{\text{CE1A}}$, $\overline{\text{CE1B}}$, $\overline{\text{CE2A}}$, $\overline{\text{CE2B}}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{\text{BS}}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{\text{RASU}}$, $\overline{\text{RASL}}$	Z				O	Z	Z*2	O	Z*2	O	O
	$\overline{\text{CASU}}$, $\overline{\text{CASL}}$	Z				O	Z	Z*2	O	Z*2	O	O
	DQMUU (PA23/PA16), DQMUL, DQMLU, DQMLL	Z				O	Z	Z*3	O	Z	O	O
	DQMUU (PE14)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*3 (MZIZEL in HCPCR = 1)	O	Z	O*6	O
	$\overline{\text{AH}}$ (PA23/PA16)	Z				O	Z	Z*3	O	Z	O	O

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
Bus control	$\overline{AH}$ (PE14)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*3 (MZIZEL in HCPCR = 1)	O	Z	O*6	O
	$\overline{FRAME}$	Z				O	Z	Z*3	O	Z	O	O
	RDWR	Z				O	Z	Z*3	O	Z	O	O
	$\overline{RD}$	H		Z		O	Z	Z*3	O	Z	O	O
	$\overline{ICIORD}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{WRHH}$ (PA23), $\overline{WRHL}$	Z	H	Z		O	Z	Z*3	O	Z	O	O
	$\overline{WRHH}$ (PE14)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*3 (MZIZEL in HCPCR = 1)	O	Z	O*6	O
	$\overline{WRHH}$ (PA16)	Z				O	Z	Z*3	O	Z	O	O
	$\overline{WRH}$, $\overline{WRL}$	H		Z		O	Z	Z*3	O	Z	O	O
	$\overline{WE}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{ICIOWR}$ (PA23/PA16)	Z				O	Z	Z*3	O	Z	O	O
	$\overline{ICIOWR}$ (PE14)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*3 (MZIZEL in HCPCR = 1)	O	Z	O*6	O
	$\overline{IOIS16}$	Z				I	Z	Z	I	I	I	I

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
Bus control	CKE (PE15)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*2 (MZIZEL in HCPCR = 1)	O	Z*2	O*6	O
	CKE (PA9/PA16)	Z				O	Z	Z*2	O	Z*2	O	O
DMAC	DREQ0 (PD24), DREQ1 (PD25)	Z				I	Z	Z	I	I	I*4	I
	DREQ0 (PA2/PE0), DREQ1 (PA5/PE2), DREQ2, DREQ3	Z				I	Z	Z	I	I	I	I
	DACK0 (PD26), DACK1 (PD27)	Z				O	Z	Z (MZIZDH in HCPCR = 0) O*1 (MZIZDH in HCPCR = 1)	O	O	O*4	O
	DACK0 (PE14), DACK1 (PE15)	Z				O	Z	Z (MZIZEL in HCPCR = 0) O*1 (MZIZEL in HCPCR = 1)	O	O	O*6	O
	DACK2, DACK3	Z				O	Z	O*1	O	O	O	O
	TEND0, TEND1	Z				O	Z	O*1	O	O	O	O

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
MTU2	TCLKA to TCLKD	Z				I	Z	Z	I	I	I	I
	TIOC0A to TIOC0D	Z				I/O	Z	K*1	I/O	I/O	I/O	Z
	TIOC1A, TIOC1B	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC2A, TIOC2B	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC3A, TIOC3C	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC3B, TIOC3D	Z				I/O	Z	Z (MZIZEL in HCPCR = 0) K*1 (MZIZEL in HCPCR = 1)	I/O	I/O	I/O*6	Z
	TIOC4A to TIOC4D	Z				I/O	Z	Z (MZIZEL in HCPCR = 0) K*1 (MZIZEL in HCPCR = 1)	I/O	I/O	I/O*6	Z
	TIC5U, TIC5V, TIC5W	Z				I	Z	Z	I	I	I	I
MTU2S	TIOC3AS, TIOC3CS	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC3BS (PD9), TIOC3DS (PD11)	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K*1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O*5	Z

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
		Expansion without ROM		Expansion with ROM	Single-chip							
Type	Pin Name	16 bits	32 bits			Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
MTU2S	TIOC3BS (PD29), TIOC3DS (PD28)	Z				I/O	Z	Z (MZIZDH in HCPCR = 0) K*1 (MZIZDH in HCPCR = 1)	I/O	I/O	I/O*4	Z
	TIOC4AS (PD12), TIOC4BS (PD13), TIOC4CS (PD14), TIOC4DS (PD15)	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K*1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O*5	Z
	TIOC4AS (PD27), TIOC4BS (PD26), TIOC4CS (PD25), TIOC4DS (PD24)	Z				I/O	Z	Z (MZIZDH in HCPCR = 0) K*1 (MZIZDH in HCPCR = 1)	I/O	I/O	I/O*4	Z
	TIC5US, TIC5VS, TIC5WS	Z				I	Z	Z	I	I	I	I
POE	$\overline{\text{POE0}}$ to $\overline{\text{POE8}}$	Z				I	Z	Z	I	I	I	I
SCI	SCK0 to SCK2	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	RXD0 to RXD2	Z				I	Z	Z	I	I	I	I
	TXD0 to TXD2	Z				O	Z	O*1	O	O	O	O
SCIF	SCK3 (PE6)	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	SCK3 (PE9)	Z				I/O	Z	Z	I/O	I/O	I/O*6	I/O
	RXD3 (PE4)	Z				I	Z	Z	I	I	I	I
	RXD3 (PE11)	Z				I	Z	Z	I	I	I*6	I

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
SCIF	TXD3 (PE5)	Z				O	Z	O*1	O	O	O	O
	TXD3 (PE12)	Z				O	Z	Z (MZIZEL in HCPCR = 0) O*1 (MZIZEL in HCPCR = 1)	O	O	O*6	O
	$\overline{\text{RTS3}}$	Z				O	Z	Z (MZIZEL in HCPCR = 0) O*1 (MZIZEL in HCPCR = 1)	O	O	O*6	O
	$\overline{\text{CTS3}}$	Z				I	Z	Z	I	I	I*6	I
SSU	SSCK	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	$\overline{\text{SCS}}$	Z				I/O	Z	Z	I/O	I/O	I/O*6	I/O
	SSI	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	SSO	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
IIC2	SCL	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	SDA	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
UBC	$\overline{\text{UBCTRG}}$	Z				O	Z	O*1	O	O	O	O
A/D Converter	AN0 to AN7	Z				I	Z	Z	I	I	I	I
	$\overline{\text{ADTRG}}$	Z				I	Z	Z	I	I	I	I
I/O Port	PA0 to PA25	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	PB0 to PB9	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	PC0 to PC15	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	PD0 to PD8, PD10, PD16 to PD23, PD30, PD31	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
I/O Port	PD9, PD11 to PD15	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K*1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O*5	Z
	PD24 to PD29	Z				I/O	Z	Z (MZIZDH in HCPCR = 0) K*1 (MZIZDH in HCPCR = 1)	I/O	I/O	I/O*4	Z
	PE0 to PE3	Z				I/O	Z	K*1	I/O	I/O	I/O	Z
	PE4 to PE8, PE10	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	PE9, PE11 to PE15	Z				I/O	Z	Z (MZIZEL in HCPCR = 0) K*1 (MZIZEL in HCPCR = 1)	I/O	I/O	I/O*6	Z
	PF0 to PF7	Z				I	Z	Z	I	I	I	I

[Legend]

I: Input

O: Output

H: High-level output

L: Low-level output

Z: High-impedance

K: Input pins become high-impedance, and output pins retain their state.

Notes: 1. Output pins become high-impedance when the HIZ bit in standby control register 6 (STBCR6) is set to 1.
2. Becomes output when the HIZCNT bit in the common control register (CMNCR) is set to 1.
3. Becomes output when the HIZMEM bit in the common control register (CMNCR) is set to 1.
4. Becomes high-impedance when the MZIZDH bit in the high-current port control register (HCPCR) is cleared to 0.
5. Becomes high-impedance when the MZIZDL bit in the high-current port control register (HCPCR) is cleared to 0.
6. Becomes high-impedance when the MZIZEL bit in the high-current port control register (HCPCR) is cleared to 0.
7. Becomes input during a power-on reset. Pull-up to prevent erroneous operation. Pull-down with a resistance of at least 1 MΩ as required.
8. Pulled-up inside the LSI when there is no input.

Table A.4 Pin States (SH7086)

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
Clock	CK	O			Z	O	Z	H*1	O	O	O	O
	XTAL	O				O	L	L	O	O	O	O
	EXTAL	I				I	Z	I	I	I	I	I
System control	$\overline{RES}$	I				I	I	I	I	I	I	I
	$\overline{MRES}$	Z				I	Z	I*7	I	I	I*7	I
	$\overline{WDTOVF}$	O*8				O	O	O	O	O	O	O
	$\overline{BREQ}$	Z				I	Z	Z	I	I	I	I
	$\overline{BACK}$	Z				O	Z	Z	O	L	O	O
Operating mode control	MD0, MD1	I				I	I	I	I	I	I	I
	$\overline{ASEMD0}$	I*9				I*9	I*9	I*9	I*9	I*9	I*9	I*9
	FWE	I				I	I	I	I	I	I	I
Interrupt	NMI	I				I	I	I	I	I	I	I
	IRQ0 to IRQ7	Z				I	Z	I	I	I	I	I
	$\overline{IRQOUT}$ (PE15)	Z				O	Z	Z (MZIZEL in HCPCR = 0) H*1 (MZIZEL in HCPCR = 1)	O	O	O*7	O
	$\overline{IRQOUT}$ (PD30)	Z				O	Z	H*1	O	O	O	O
Address bus	A0 to A25	O		Z		O	Z	Z*3	O	Z	O	O
	A26 to A29	Z				O	Z	Z*3	O	Z	O	O
Data bus	D0 to D8, D10, D16 to D23, D30, D31	Z				I/O	Z	Z	I/O	Z	I/O	I/O
	D9, D11 to D15	Z				I/O	Z	Z	I/O	Z	I/O*5	I/O
	D24 to D29	Z				I/O	Z	Z	I/O	Z	I/O*4	I/O

Pin Function		Pin State										
		Reset State				Power-Down State						
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
Bus control	$\overline{WAIT}$	Z				I	Z	Z	I	Z	I	I
	$\overline{CS0}$, $\overline{CS1}$	H		Z		O	Z	Z*3	O	Z	O	O
	$\overline{CS2}$ (PA6), $\overline{CS3}$ (PA7), $\overline{CS4}$ to $\overline{CS7}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{CS2}$ (PD28), $\overline{CS3}$ (PD29)	Z				O	Z	Z (MZIZDH in HCPCR = 0) Z*3 (MZIZDH in HCPCR = 1)	O	Z	O*4	O
	$\overline{CS8}$	Z				O	Z	Z (MZIZEH in HCPCR = 0) Z*3 (MZIZEH in HCPCR = 1)	O	Z	O*6	O
	$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{BS}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{RASU}$, $\overline{RASL}$	Z				O	Z	Z*2	O	Z*2	O	O
	$\overline{CASU}$, $\overline{CASL}$	Z				O	Z	Z*2	O	Z*2	O	O
	DQMUU (PA23/PA16), DQMUL, DQMLU, DQMLL	Z				O	Z	Z*3	O	Z	O	O
	DQMUU (PE14)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*3 (MZIZEL in HCPCR = 1)	O	Z	O*7	O

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
Bus control	$\overline{AH}$ (PA23/PA16)	Z				O	Z	Z*3	O	Z	O	O
	$\overline{AH}$ (PE14)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*3 (MZIZEL in HCPCR = 1)	O	Z	O*7	O
	$\overline{FRAME}$	Z				O	Z	Z*3	O	Z	O	O
	RDWR	Z				O	Z	Z*3	O	Z	O	O
	$\overline{RD}$	H		Z		O	Z	Z*3	O	Z	O	O
	$\overline{ICIORD}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{WRHH}$ (PA23), $\overline{WRHL}$	Z	H	Z		O	Z	Z*3	O	Z	O	O
	$\overline{WRHH}$ (PE14)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*3 (MZIZEL in HCPCR = 1)	O	Z	O*7	O
	$\overline{WRHH}$ (PA16)	Z				O	Z	Z*3	O	Z	O	O
	$\overline{WRH}$, $\overline{WRL}$	H		Z		O	Z	Z*3	O	Z	O	O
	$\overline{WE}$	Z				O	Z	Z*3	O	Z	O	O
	$\overline{ICIOWR}$ (PA23/PA16)	Z				O	Z	Z*3	O	Z	O	O
	$\overline{ICIOWR}$ (PE14)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*3 (MZIZEL in HCPCR = 1)	O	Z	O*7	O
	$\overline{IOIS16}$	Z				I	Z	Z	I	I	I	I

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
Bus control	CKE (PE15)	Z				O	Z	Z (MZIZEL in HCPCR = 0) Z*2 (MZIZEL in HCPCR = 1)	O	Z*2	O*7	O
	CKE (PA9/PA16)	Z				O	Z	Z*2	O	Z*2	O	O
DMAC	DREQ0 (PD24), DREQ1 (PD25)	Z				I	Z	Z	I	I	I*4	I
	DREQ0 (PA2/PE0), DREQ1 (PA5/PE2), DREQ2, DREQ3	Z				I	Z	Z	I	I	I	I
	DACK0 (PD26), DACK1 (PD27)	Z				O	Z	Z (MZIZDH in HCPCR = 0) O*1 (MZIZDH in HCPCR = 1)	O	O	O*4	O
	DACK0 (PE14), DACK1 (PE15)	Z				O	Z	Z (MZIZEL in HCPCR = 0) O*1 (MZIZEL in HCPCR = 1)	O	O	O*7	O
	DACK2, DACK3	Z				O	Z	O*1	O	O	O	O
	TEND0, TEND1	Z				O	Z	O*1	O	O	O	O

Pin Function		Pin State										
		Reset State				Power-Down State						
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
MTU2	TCLKA to TCLKD	Z				I	Z	Z	I	I	I	I
	TIOC0A to TIOC0D	Z				I/O	Z	K*1	I/O	I/O	I/O	Z
	TIOC1A, TIOC1B	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC2A, TIOC2B	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC3A, TIOC3C	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC3B, TIOC3D	Z				I/O	Z	Z (MZIZEL in HCPCR = 0) K*1 (MZIZEL in HCPCR = 1)	I/O	I/O	I/O*7	Z
	TIOC4A to TIOC4D	Z				I/O	Z	Z (MZIZEL in HCPCR = 0) K*1 (MZIZEL in HCPCR = 1)	I/O	I/O	I/O*7	Z
	TIC5U, TIC5V, TIC5W	Z				I	Z	Z	I	I	I	I
MTU2S	TIOC3AS, TIOC3CS	Z				I/O	Z	K*1	I/O	I/O	I/O	I/O
	TIOC3BS (PD9), TIOC3DS (PD11)	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K*1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O*5	Z

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
MTU2S	TIOC3BS (PD29), TIOC3DS (PD28)	Z				I/O	Z	Z (MZIZDH in HCPCR = 0) K*1 (MZIZDH in HCPCR = 1)	I/O	I/O	I/O*4	Z
	TIOC3BS (PE16), TIOC3DS (PE17)	Z				I/O	Z	Z (MZIZEH in HCPCR = 0) K*1 (MZIZEH in HCPCR = 1)	I/O	I/O	I/O*6	Z
	TIOC4AS (PD12), TIOC4BS (PD13), TIOC4CS (PD14), TIOC4DS (PD15)	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K*1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O*5	Z
	TIOC4AS (PD27), TIOC4BS (PD26), TIOC4CS (PD25), TIOC4DS (PD24)	Z				I/O	Z	Z (MZIZDH in HCPCR = 0) K*1 (MZIZDH in HCPCR = 1)	I/O	I/O	I/O*4	Z
	TIOC4AS (PE18), TIOC4BS (PE19), TIOC4CS (PE20), TIOC4DS (PE21)	Z				I/O	Z	Z (MZIZEH in HCPCR = 0) K*1 (MZIZEH in HCPCR = 1)	I/O	I/O	I/O*6	Z

Pin Function		Pin State										
		Reset State				Power-Down State						
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
MTU2S	TIC5US, TIC5VS, TIC5WS	Z				I	Z	Z	I	I	I	I
POE	$\overline{\text{POE0}}$ to $\overline{\text{POE8}}$	Z				I	Z	Z	I	I	I	I
SCI	SCK0 to SCK2	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	RXD0 to RXD2	Z				I	Z	Z	I	I	I	I
	TXD0 to TXD2	Z				O	Z	O*1	O	O	O	O
SCIF	SCK3 (PE6)	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	SCK3 (PE9)	Z				I/O	Z	Z	I/O	I/O	I/O*7	I/O
	RXD3 (PE4)	Z				I	Z	Z	I	I	I	I
	RXD3 (PE11)	Z				I	Z	Z	I	I	I*7	I
	TXD3 (PE5)	Z				O	Z	O*1	O	O	O	O
	TXD3 (PE12)	Z				O	Z	Z (MZIZEL in HCPCR = 0) O*1 (MZIZEL in HCPCR = 1)	O	O	O*7	O
	$\overline{\text{RTS3}}$	Z				O	Z	Z (MZIZEL in HCPCR = 0) O*1 (MZIZEL in HCPCR = 1)	O	O	O*7	O
	$\overline{\text{CTS3}}$	Z				I	Z	Z	I	I	I*7	I
SSU	SSCK	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	$\overline{\text{SCS}}$	Z				I/O	Z	Z	I/O	I/O	I/O*7	I/O
	SSI	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	SSO	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
IIC2	SCL	Z				I/O	Z	Z	I/O	I/O	I/O	I/O
	SDA	Z				I/O	Z	Z	I/O	I/O	I/O	I/O

Pin Function		Pin State										
		Reset State				Power-Down State						
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Mastership Release	Oscillation Stop Detected	POE Function Used
UBC	$\overline{\text{UBCTRG}}$	Z				O	Z	O※1	O	O	O	O
A/D Converter	AN0 to AN15	Z				I	Z	Z	I	I	I	I
	$\overline{\text{ADTRG}}$	Z				I	Z	Z	I	I	I	I
I/O Port	PA0 to PA29	Z				I/O	Z	K※1	I/O	I/O	I/O	I/O
	PB0 to PB9	Z				I/O	Z	K※1	I/O	I/O	I/O	I/O
	PC0 to PC15, PC18 to PC25	Z				I/O	Z	K※1	I/O	I/O	I/O	I/O
	PD0 to PD8, PD10, PD16 to PD23, PD30, PD31	Z				I/O	Z	K※1	I/O	I/O	I/O	I/O
	PD9, PD11 to PD15	Z				I/O	Z	Z (MZIZDL in HCPCR = 0) K※1 (MZIZDL in HCPCR = 1)	I/O	I/O	I/O※5	Z
	PD24 to PD29	Z				I/O	Z	Z (MZIZDH in HCPCR = 0) K※1 (MZIZDH in HCPCR = 1)	I/O	I/O	I/O※4	Z
	PE0 to PE3	Z				I/O	Z	K※1	I/O	I/O	I/O	Z
	PE4 to PE8, PE10	Z				I/O	Z	K※1	I/O	I/O	I/O	I/O
	PE9, PE11 to PE15	Z				I/O	Z	Z (MZIZEL in HCPCR = 0) K※1 (MZIZEL in HCPCR = 1)	I/O	I/O	I/O※7	Z

Pin Function		Pin State										
		Reset State					Power-Down State					
		Power-On										
Type	Pin Name	Expansion without ROM 16 bits	Expansion without ROM 32 bits	Expansion with ROM	Single-chip	Manual	Deep Software Standby	Software Standby	Sleep	Bus Master-ship Release	Oscillation Stop Detected	POE Function Used
I/O Port	PE16 to PE21	Z				I/O	Z	Z (MZIZEH in HCPCR = 0) K*1 (MZIZEH in HCPCR = 1)	I/O	I/O	I/O*6	Z
	PF0 to PF15	Z				I	Z	Z	I	I	I	I

[Legend]

I: Input

O: Output

H: High-level output

L: Low-level output

Z: High-impedance

K: Input pins become high-impedance, and output pins retain their state.

Notes: 1. Output pins become high-impedance when the HIZ bit in standby control register 6 (STBCR6) is set to 1.

2. Becomes output when the HIZCNT bit in the common control register (CMNCR) is set to 1.

3. Becomes output when the HIZMEM bit in the common control register (CMNCR) is set to 1.

4. Becomes high-impedance when the MZIZDH bit in the high-current port control register (HCPCR) is cleared to 0.

5. Becomes high-impedance when the MZIZDL bit in the high-current port control register (HCPCR) is cleared to 0.

6. Becomes high-impedance when the MZIZEH bit in the high-current port control register (HCPCR) is cleared to 0.

7. Becomes high-impedance when the MZIZEL bit in the high-current port control register (HCPCR) is cleared to 0.

8. Becomes input during a power-on reset. Pull-up to prevent erroneous operation. Pull-down with a resistance of at least 1 MΩ as required.

9. Pulled-up inside the LSI when there is no input.

B. Processing of Unused Pins

Table B.1 Processing of Unused Pins

Pin	Processing
NMI	Fixed high-level (pull-up)
$\overline{\text{WDTOVF}}$	Open (If pull-down is necessary, use a resistor rated at 1 Ω or greater.)
AVref	AVref = Avcc
AVcc, AVss	AVcc = Vcc, AVss = Vss
$\overline{\text{ASEMD0}}$	Fixed high-level (pull-up)
PF0 to PF7 (PF0 to PF15 in SH7086)	Connect to AVcc or AVss via a resistor.
Input-only pins other than the above	Fixed (pull-up/pull-down)
I/O pins other than the above	Fixed at input pin setting (pull-up/pull-down) or set to output and left open
Output-only pins	Open

Notes: 1. For pull-up or pull-down, connect to Vcc or GND via a resistor.
2. When using the H-UDI, pin processing is according to the specifications of the emulator.

C. Pin States of Bus Related Signals

Table C.1 Pin States of Bus Related Signals (1)

Pin Name		On-chip ROM Space	On-chip RAM Space	On-chip Peripheral Module Space
$\overline{CS0}$ to $\overline{CS8}$		H	H	H
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$		H	H	H
$\overline{BS}$		H	H	H
$\overline{RASU}$, $\overline{RASL}$		H	H	H
$\overline{CASU}$, $\overline{CASL}$		H	H	H
DQMUU		H	H	H
DQMUL		H	H	H
DQMLU		H	H	H
DQMLL		H	H	H
$\overline{AH}$		L	L	L
$\overline{FRAME}$		H	H	H
RDWR	R	H	H	H
	W	—	H	H
$\overline{RD}$	R	H	H	H
	W	—	H	H
$\overline{ICIORD}$	R	H	H	H
	W	—	H	H
$\overline{WRHH}$	R	H	H	H
	W	—	H	H
$\overline{WRHL}$	R	H	H	H
	W	—	H	H
$\overline{WRH}$	R	H	H	H
	W	—	H	H
$\overline{WRL}$	R	H	H	H
	W	—	H	H
$\overline{WE}$	R	H	H	H
	W	—	H	H

Pin Name		On-chip ROM Space	On-chip RAM Space	On-chip Peripheral Module Space
$\overline{\text{ICIOWR}}$	R	H	H	H
	W	—	H	H
A29 to A0		Address*	Address*	Address*
D31 to D24		Hi-Z	Hi-Z	Hi-Z
D23 to D16		Hi-Z	Hi-Z	Hi-Z
D15 to D8		Hi-Z	Hi-Z	Hi-Z
D7 to D0		Hi-Z	Hi-Z	Hi-Z

[Legend]
R: Read
W: Write
Note: * Value of external space address that was previously accessed

Table C.1 Pin States of Bus Related Signals (2)

		External Space (Normal Space)			
			16-bit Space		
Pin Name		8-bit Space	Upper Byte	Lower Byte	Word/Longword
$\overline{\text{CS0}}$ to $\overline{\text{CS8}}$		Enabled	Enabled	Enabled	Enabled
$\overline{\text{CE1A}}$, $\overline{\text{CE1B}}$, $\overline{\text{CE2A}}$, $\overline{\text{CE2B}}$		H	H	H	H
$\overline{\text{BS}}$		L	L	L	L
$\overline{\text{RASU}}$, $\overline{\text{RASL}}$		H	H	H	H
$\overline{\text{CASU}}$, $\overline{\text{CASL}}$		H	H	H	H
DQMUU		H	H	H	H
DQMUL		H	H	H	H
DQMLU		H	H	H	H
DQMLL		H	H	H	H
$\overline{\text{AH}}$		L	L	L	L
$\overline{\text{FRAME}}$		H	H	H	H
RDWR	R	H	H	H	H
	W	L	L	L	L

		External Space (Normal Space)			
			16-bit Space		
Pin Name		**8-bit Space**	**Upper Byte**	**Lower Byte**	**Word/Longword**
$\overline{RD}$	R	L	L	L	L
	W	H	H	H	H
$\overline{ICIORD}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRHH}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRHL}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRH}$	R	H	H	H	H
	W	H	L	H	L
$\overline{WRL}$	R	H	H	H	H
	W	L	H	L	L
$\overline{WE}$	R	H	H	H	H
	W	H	H	H	H
$\overline{ICIOWR}$	R	H	H	H	H
	W	H	H	H	H
A29 to A0		Address	Address	Address	Address
D31 to D24		Hi-Z	Hi-Z	Hi-Z	Hi-Z
D23 to D16		Hi-Z	Hi-Z	Hi-Z	Hi-Z
D15 to D8		Hi-Z	Data	Hi-Z	Data
D7 to D0		Data	Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Enabled: Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

Table C.1 Pin States of Bus Related Signals (3)

		External Space (Normal Space)						
		32-bit Space						
Pin Name		Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{CS0}$ to $\overline{CS8}$		Enabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$		H	H	H	H	H	H	H
$\overline{BS}$		L	L	L	L	L	L	L
$\overline{RASU}$, $\overline{RASL}$		H	H	H	H	H	H	H
$\overline{CASU}$, $\overline{CASL}$		H	H	H	H	H	H	H
DQMUU		H	H	H	H	H	H	H
DQMUL		H	H	H	H	H	H	H
DQMLU		H	H	H	H	H	H	H
DQMLL		H	H	H	H	H	H	H
$\overline{AH}$		L	L	L	L	L	L	L
$\overline{FRAME}$		H	H	H	H	H	H	H
RDWR	R	H	H	H	H	H	H	H
	W	L	L	L	L	L	L	L
$\overline{RD}$	R	L	L	L	L	L	L	L
	W	H	H	H	H	H	H	H
$\overline{ICIORD}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{WRHH}$	R	H	H	H	H	H	H	H
	W	L	H	H	H	L	H	L
$\overline{WRHL}$	R	H	H	H	H	H	H	H
	W	H	L	H	H	L	H	L
$\overline{WRH}$	R	H	H	H	H	H	H	H
	W	H	H	L	H	H	L	L
$\overline{WRL}$	R	H	H	H	H	H	H	H
	W	H	H	H	L	H	L	L

		External Space (Normal Space)						
		32-bit Space						
Pin Name		Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{WE}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{ICIOWR}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
A29 to A0		Address	Address	Address	Address	Address	Address	Address
D31 to D24		Data	Hi-Z	Hi-Z	Hi-Z	Data	Hi-Z	Data
D23 to D16		Hi-Z	Data	Hi-Z	Hi-Z	Data	Hi-Z	Data
D15 to D8		Hi-Z	Hi-Z	Data	Hi-Z	Hi-Z	Data	Data
D7 to D0		Hi-Z	Hi-Z	Hi-Z	Data	Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Enabled: Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

Table C.1 Pin States of Bus Related Signals (4)

	External Space (SRAM with Byte Selection)		
	16-bit Space		
Pin Name	Upper Byte	Lower Byte	Word/Longword
$\overline{CS0}$ to $\overline{CS8}$	Enabled	Enabled	Enabled
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$	H	H	H
$\overline{BS}$	L	L	L
$\overline{RASU}$, $\overline{RASL}$	H	H	H
$\overline{CASU}$, $\overline{CASL}$	H	H	H
DQMUU	H	H	H
DQMUL	H	H	H
DQMLU	H	H	H
DQMLL	H	H	H

		External Space (SRAM with Byte Selection)		
		16-bit Space		
Pin Name		**Upper Byte**	**Lower Byte**	**Word/Longword**
$\overline{AH}$		L	L	L
$\overline{FRAME}$		H	H	H
RDWR	R	H	H	H
	W	L	L	L
$\overline{RD}$	R	L	L	L
	W	H	H	H
$\overline{ICIORD}$	R	H	H	H
	W	H	H	H
$\overline{WRHH}$	R	H	H	H
	W	H	H	H
$\overline{WRHL}$	R	H	H	H
	W	H	H	H
$\overline{WRH}$	R	L	H	L
	W	L	H	L
$\overline{WRL}$	R	H	L	L
	W	H	L	L
$\overline{WE}$	R	H	H	H
	W	H	H	H
$\overline{ICIOWR}$	R	H	H	H
	W	H	H	H
A29 to A0		Address	Address	Address
D31 to D24		Hi-Z	Hi-Z	Hi-Z
D23 to D16		Hi-Z	Hi-Z	Hi-Z
D15 to D8		Data	Hi-Z	Data
D7 to D0		Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Enabled: Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

Table C.1 Pin States of Bus Related Signals (5)

		External Space (SRAM with Byte Selection)						
		32-bit Space						
Pin Name		Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{CS0}$ to $\overline{CS8}$		Enabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$		H	H	H	H	H	H	H
$\overline{BS}$		L	L	L	L	L	L	L
$\overline{RASU}$, $\overline{RASL}$		H	H	H	H	H	H	H
$\overline{CASU}$, $\overline{CASL}$		H	H	H	H	H	H	H
DQMUU		H	H	H	H	H	H	H
DQMUL		H	H	H	H	H	H	H
DQMLU		H	H	H	H	H	H	H
DQMLL		H	H	H	H	H	H	H
$\overline{AH}$		L	L	L	L	L	L	L
$\overline{FRAME}$		H	H	H	H	H	H	H
RDWR	R	H	H	H	H	H	H	H
	W	L	L	L	L	L	L	L
$\overline{RD}$	R	L	L	L	L	L	L	L
	W	H	H	H	H	H	H	H
$\overline{ICIORD}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{WRHH}$	R	L	H	H	H	L	H	L
	W	L	H	H	H	L	H	L
$\overline{WRHL}$	R	H	L	H	H	L	H	L
	W	H	L	H	H	L	H	L
$\overline{WRH}$	R	H	H	L	H	H	L	L
	W	H	H	L	H	H	L	L
$\overline{WRL}$	R	H	H	H	L	H	L	L
	W	H	H	H	L	H	L	L

		External Space (Normal Space)						
		32-bit Space						
Pin Name		Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{WE}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{ICIOWR}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
A29 to A0		Address	Address	Address	Address	Address	Address	Address
D31 to D24		Data	Hi-Z	Hi-Z	Hi-Z	Data	Hi-Z	Data
D23 to D16		Hi-Z	Data	Hi-Z	Hi-Z	Data	Hi-Z	Data
D15 to D8		Hi-Z	Hi-Z	Data	Hi-Z	Hi-Z	Data	Data
D7 to D0		Hi-Z	Hi-Z	Hi-Z	Data	Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Enabled: Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

Table C.1 Pin States of Bus Related Signals (6)

	External Space (Burst ROM (Clock Asynchronous))			
		16-bit Space		
Pin Name	8-bit Space	Upper Byte	Lower Byte	Word/Longword
$\overline{CS0}$ to $\overline{CS8}$	Enabled	Enabled	Enabled	Enabled
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$	H	H	H	H
$\overline{BS}$	L	L	L	L
$\overline{RASU}$, $\overline{RASL}$	H	H	H	H
$\overline{CASU}$, $\overline{CASL}$	H	H	H	H
DQMUU	H	H	H	H
DQMUL	H	H	H	H
DQMLU	H	H	H	H
DQMLL	H	H	H	H

		External Space (Burst ROM (Clock Asynchronous))			
			16-bit Space		
Pin Name		**8-bit Space**	**Upper Byte**	**Lower Byte**	**Word/Longword**
$\overline{AH}$		L	L	L	L
$\overline{FRAME}$		H	H	H	H
RDWR	R	H	H	H	H
	W	—	—	—	—
$\overline{RD}$	R	L	L	L	L
	W	—	—	—	—
$\overline{ICIORD}$	R	H	H	H	H
	W	—	—	—	—
$\overline{WRHH}$	R	H	H	H	H
	W	—	—	—	—
$\overline{WRHL}$	R	H	H	H	H
	W	—	—	—	—
$\overline{WRH}$	R	H	H	H	H
	W	—	—	—	—
$\overline{WRL}$	R	H	H	H	H
	W	—	—	—	—
$\overline{WE}$	R	H	H	H	H
	W	—	—	—	—
$\overline{ICIOWR}$	R	H	H	H	H
	W	—	—	—	—
A29 to A0		Address	Address	Address	Address
D31 to D24		Hi-Z	Hi-Z	Hi-Z	Hi-Z
D23 to D16		Hi-Z	Hi-Z	Hi-Z	Hi-Z
D15 to D8		Hi-Z	Data	Hi-Z	Data
D7 to D0		Data	Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Enabled: Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

Table C.1 Pin States of Bus Related Signals (7)

		External Space (Burst ROM (Clock Asynchronous))						
		32-bit Space						
Pin Name		Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{CS0}$ to $\overline{CS8}$		Enabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$		H	H	H	H	H	H	H
$\overline{BS}$		L	L	L	L	L	L	L
$\overline{RASU}$, $\overline{RASL}$		H	H	H	H	H	H	H
$\overline{CASU}$, $\overline{CASL}$		H	H	H	H	H	H	H
DQMUU		H	H	H	H	H	H	H
DQMUL		H	H	H	H	H	H	H
DQMLU		H	H	H	H	H	H	H
DQMLL		H	H	H	H	H	H	H
$\overline{AH}$		L	L	L	L	L	L	L
$\overline{FRAME}$		H	H	H	H	H	H	H
RDWR	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{RD}$	R	L	L	L	L	L	L	L
	W	—	—	—	—	—	—	—
$\overline{ICIORD}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{WRHH}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{WRHL}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{WRH}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{WRL}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—

		External Space (Burst ROM (Clock Asynchronous))						
		32-bit Space						
Pin Name		Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{WE}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{ICIOWR}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
A29 to A0		Address	Address	Address	Address	Address	Address	Address
D31 to D24		Data	Hi-Z	Hi-Z	Hi-Z	Data	Hi-Z	Data
D23 to D16		Hi-Z	Data	Hi-Z	Hi-Z	Data	Hi-Z	Data
D15 to D8		Hi-Z	Hi-Z	Data	Hi-Z	Hi-Z	Data	Data
D7 to D0		Hi-Z	Hi-Z	Hi-Z	Data	Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Enabled: Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

Table C.1 Pin States of Bus Related Signals (8)

	External Space (Burst ROM (Clock Synchronous))		
	16-bit Space		
Pin Name	Upper Byte	Lower Byte	Word/Longword
$\overline{CS0}$ to $\overline{CS8}$	Enabled	Enabled	Enabled
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$	H	H	H
$\overline{BS}$	L	L	L
$\overline{RASU}$, $\overline{RASL}$	H	H	H
$\overline{CASU}$, $\overline{CASL}$	H	H	H
DQMUU	H	H	H
DQMUL	H	H	H
DQMLU	H	H	H
DQMLL	H	H	H

		External Space (Burst ROM (Clock Synchronous))		
		16-bit Space		
Pin Name		**Upper Byte**	**Lower Byte**	**Word/Longword**
$\overline{AH}$		L	L	L
$\overline{FRAME}$		H	H	H
RDWR	R	H	H	H
	W	—	—	—
$\overline{RD}$	R	L	L	L
	W	—	—	—
$\overline{ICIORD}$	R	H	H	H
	W	—	—	—
$\overline{WRHH}$	R	H	H	H
	W	—	—	—
$\overline{WRHL}$	R	H	H	H
	W	—	—	—
$\overline{WRH}$	R	H	H	H
	W	—	—	—
$\overline{WRL}$	R	H	H	H
	W	—	—	—
$\overline{WE}$	R	H	H	H
	W	—	—	—
$\overline{ICIOWR}$	R	H	H	H
	W	—	—	—
A29 to A0		Address	Address	Address
D31 to D24		Hi-Z	Hi-Z	Hi-Z
D23 to D16		Hi-Z	Hi-Z	Hi-Z
D15 to D8		Data	Hi-Z	Data
D7 to D0		Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Enabled: Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

Table C.1 Pin States of Bus Related Signals (9)

		External Space (Burst ROM (Clock Synchronous))						
		32-bit Space						
Pin Name		Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{CS0}$ to $\overline{CS8}$		Enabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$		H	H	H	H	H	H	H
$\overline{BS}$		L	L	L	L	L	L	L
$\overline{RASU}$, $\overline{RASL}$		H	H	H	H	H	H	H
$\overline{CASU}$, $\overline{CASL}$		H	H	H	H	H	H	H
DQMUU		H	H	H	H	H	H	H
DQMUL		H	H	H	H	H	H	H
DQMLU		H	H	H	H	H	H	H
DQMLL		H	H	H	H	H	H	H
$\overline{AH}$		L	L	L	L	L	L	L
$\overline{FRAME}$		H	H	H	H	H	H	H
RDWR	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{RD}$	R	L	L	L	L	L	L	L
	W	—	—	—	—	—	—	—
$\overline{ICIORD}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{WRHH}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{WRHL}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{WRH}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{WRL}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—

		External Space (Burst ROM (Clock Synchronous))						
		32-bit Space						
Pin Name		Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{WE}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
$\overline{ICIOWR}$	R	H	H	H	H	H	H	H
	W	—	—	—	—	—	—	—
A29 to A0		Address	Address	Address	Address	Address	Address	Address
D31 to D24		Data	Hi-Z	Hi-Z	Hi-Z	Data	Hi-Z	Data
D23 to D16		Hi-Z	Data	Hi-Z	Hi-Z	Data	Hi-Z	Data
D15 to D8		Hi-Z	Hi-Z	Data	Hi-Z	Hi-Z	Data	Data
D7 to D0		Hi-Z	Hi-Z	Hi-Z	Data	Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Enabled: Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

Table C.1 Pin States of Bus Related Signals (10)

	External Space (SDRAM)		
	16-bit Space		
Pin Name	Upper Byte	Lower Byte	Word/Longword
$\overline{CS0}$ to $\overline{CS8}$	Enabled*[1]	Enabled*[1]	Enabled*[1]
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$	H	H	H
$\overline{BS}$	L	L	L
$\overline{RASU}$, $\overline{RASL}$	Enabled*[2]	Enabled*[2]	Enabled*[2]
$\overline{CASU}$, $\overline{CASL}$	Enabled*[2]	Enabled*[2]	Enabled*[2]
DQMUU	H	H	H
DQMUL	H	H	H
DQMLU	L	H	L
DQMLL	H	L	L

		External Space (SDRAM)		
		16-bit Space		
Pin Name		**Upper Byte**	**Lower Byte**	**Word/Longword**
$\overline{AH}$		L	L	L
$\overline{FRAME}$		H	H	H
RDWR	R	H	H	H
	W	L	L	L
$\overline{RD}$	R	H	H	H
	W	H	H	H
$\overline{ICIORD}$	R	H	H	H
	W	H	H	H
$\overline{WRHH}$	R	H	H	H
	W	H	H	H
$\overline{WRHL}$	R	H	H	H
	W	H	H	H
$\overline{WRH}$	R	H	H	H
	W	H	H	H
$\overline{WRL}$	R	H	H	H
	W	H	H	H
$\overline{WE}$	R	H	H	H
	W	H	H	H
$\overline{ICIOWR}$	R	H	H	H
	W	H	H	H
A29 to A0		Address	Address	Address
D31 to D24		Hi-Z	Hi-Z	Hi-Z
D23 to D16		Hi-Z	Hi-Z	Hi-Z
D15 to D8		Data	Hi-Z	Data
D7 to D0		Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Notes: 1. Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

2. $\overline{RASL}/\overline{CASL}$ = Low when address wherein A25 = 0 is accessed. $\overline{RASU}/\overline{CASU}$ = Low when address wherein A25 = 1 is accessed.

Table C.1 Pin States of Bus Related Signals (11)

		External Space (SDRAM)						
		32-bit Space						
Pin Name		Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{CS0}$ to $\overline{CS8}$		Enabled*1	Enabled*1	Enabled*1	Enabled*1	Enabled*1	Enabled*1	Enabled*1
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$		H	H	H	H	H	H	H
$\overline{BS}$		L	L	L	L	L	L	L
$\overline{RASU}$, $\overline{RASL}$		Enabled*2	Enabled*2	Enabled*2	Enabled*2	Enabled*2	Enabled*2	Enabled*2
$\overline{CASU}$, $\overline{CASL}$		Enabled*2	Enabled*2	Enabled*2	Enabled*2	Enabled*2	Enabled*2	Enabled*2
DQMUU		L	H	H	H	L	H	L
DQMUL		H	L	H	H	L	H	L
DQMLU		H	H	L	H	H	L	L
DQMLL		H	H	H	L	H	L	L
$\overline{AH}$		L	L	L	L	L	L	L
$\overline{FRAME}$		H	H	H	H	H	H	H
RDWR	R	H	H	H	H	H	H	H
	W	L	L	L	L	L	L	L
$\overline{RD}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{ICIORD}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{WRHH}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{WRHL}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{WRH}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{WRL}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H

		External Space (Normal Space)						
		32-bit Space						
Pin Name		Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{WE}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{ICIOWR}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
A29 to A0		Address	Address	Address	Address	Address	Address	Address
D31 to D24		Data	Hi-Z	Hi-Z	Hi-Z	Data	Hi-Z	Data
D23 to D16		Hi-Z	Data	Hi-Z	Hi-Z	Data	Hi-Z	Data
D15 to D8		Hi-Z	Hi-Z	Data	Hi-Z	Hi-Z	Data	Data
D7 to D0		Hi-Z	Hi-Z	Hi-Z	Data	Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Notes: 1. Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

2. $\overline{RASL}/\overline{CASL}$ = Low when address wherein A25 = 0 is accessed. $\overline{RASU}/\overline{CASU}$ = Low when address wherein A25 = 1 is accessed.

Table C.1 Pin States of Bus Related Signals (12)

		External Space (MPX-I/O)			
			16-bit Space		
Pin Name		**8-bit Space**	**Upper Byte**	**Lower Byte**	**Word/Longword**
$\overline{CS0}$ to $\overline{CS8}$		Enabled	Enabled	Enabled	Enabled
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$		H	H	H	H
$\overline{BS}$		L	L	L	L
$\overline{RASU}$, $\overline{RASL}$		H	H	H	H
$\overline{CASU}$, $\overline{CASL}$		H	H	H	H
DQMUU		H	H	H	H
DQMUL		H	H	H	H
DQMLU		H	H	H	H
DQMLL		H	H	H	H
$\overline{AH}$		H	H	H	H
$\overline{FRAME}$		H	H	H	H
RDWR	R	H	H	H	H
	W	L	L	L	L
$\overline{RD}$	R	L	L	L	L
	W	H	H	H	H
$\overline{ICIORD}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRHH}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRHL}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRH}$	R	H	H	H	H
	W	H	L	H	L
$\overline{WRL}$	R	H	H	H	H
	W	L	H	L	L
$\overline{WE}$	R	H	H	H	H
	W	H	H	H	H

		External Space (MPX-I/O)			
			16-bit Space		
Pin Name		8-bit Space	Upper Byte	Lower Byte	Word/Longword
$\overline{\text{ICIOWR}}$	R	H	H	H	H
	W	H	H	H	H
A29 to A0		Address	Address	Address	Address
D31 to D24		Hi-Z	Hi-Z	Hi-Z	Hi-Z
D23 to D16		Hi-Z	Hi-Z	Hi-Z	Hi-Z
D15 to D8		Hi-Z	Address/Data	Address	Address/Data
D7 to D0		Address/Data	Address	Address/Data	Address/Data

[Legend]

R: Read

W: Write

Enabled: Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

Table C.1 Pin States of Bus Related Signals (13)

	External Space (Burst MPX-I/O)						
	32-bit Space						
Pin Name	Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{\text{CS0}}$ to $\overline{\text{CS8}}$	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
$\overline{\text{CE1A}}$, $\overline{\text{CE1B}}$, $\overline{\text{CE2A}}$, $\overline{\text{CE2B}}$	H	H	H	H	H	H	H
$\overline{\text{BS}}$	L	L	L	L	L	L	L
$\overline{\text{RASU}}$, $\overline{\text{RASL}}$	H	H	H	H	H	H	H
$\overline{\text{CASU}}$, $\overline{\text{CASL}}$	H	H	H	H	H	H	H
DQMUU	H	H	H	H	H	H	H
DQMUL	H	H	H	H	H	H	H
DQMLU	H	H	H	H	H	H	H
DQMLL	H	H	H	H	H	H	H

		External Space (Burst MPX-I/O)						
		32-bit Space						
Pin Name		Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
$\overline{AH}$		L	L	L	L	L	L	L
$\overline{FRAME}$		L	L	L	L	L	L	L
RDWR	R	H	H	H	H	H	H	H
	W	L	L	L	L	L	L	L
$\overline{RD}$	R	L	L	L	L	L	L	L
	W	H	H	H	H	H	H	H
$\overline{ICIORD}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{WRHH}$	R	H	H	H	H	H	H	H
	W	L	H	H	H	L	H	L
$\overline{WRHL}$	R	H	H	H	H	H	H	H
	W	H	L	H	H	L	H	L
$\overline{WRH}$	R	H	H	H	H	H	H	H
	W	H	H	L	H	H	L	L
$\overline{WRL}$	R	H	H	H	H	H	H	H
	W	H	H	H	L	H	L	L
$\overline{WE}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
$\overline{ICIOWR}$	R	H	H	H	H	H	H	H
	W	H	H	H	H	H	H	H
A29 to A0		Address	Address	Address	Address	Address	Address	Address
D31 to D24		Address/ Data	Address	Address	Address	Address/ Data	Address	Address/ Data
D23 to D16		Address	Address/ Data	Address	Address	Address/ Data	Address	Address/ Data

	External Space (Burst MPX-I/O)						
	32-bit Space						
Pin Name	Most Significant Byte	Second Byte	Third Byte	Least Significant Byte	Upper Word	Lower Word	Longword
D15 to D8	Address	Address	Address/ Data	Address	Address	Address/ Data	Address/ Data
D7 to D0	Address	Address	Address	Address/ Data	Address	Address/ Data	Address/ Data

[Legend]

R: Read

W: Write

Enabled: Chip select signals corresponding to accessed areas = Low.
The other chip select signals = High.

Table C.1 Pin States of Bus Related Signals (14)

		External Space (PCMCIA Memory Card Interface)			
			16-bit Space		
Pin Name		8-bit Space	Upper Byte	Lower Byte	Word/Longword
$\overline{CS0}$ to $\overline{CS8}$		H	H	H	H
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$		Enabled	Enabled	Enabled	Enabled
$\overline{BS}$		L	L	L	L
$\overline{RASU}$, $\overline{RASL}$		H	H	H	H
$\overline{CASU}$, $\overline{CASL}$		H	H	H	H
DQMUU		H	H	H	H
DQMUL		H	H	H	H
DQMLU		H	H	H	H
DQMLL		H	H	H	H
$\overline{AH}$		L	L	L	L
$\overline{FRAME}$		H	H	H	H
RDWR	R	H	H	H	H
	W	L	L	L	L

		External Space (PCMCIA Memory Card Interface)			
			16-bit Space		
Pin Name		**8-bit Space**	**Upper Byte**	**Lower Byte**	**Word/Longword**
$\overline{RD}$	R	L	L	L	L
	W	H	H	H	H
$\overline{ICIORD}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRHH}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRHL}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRH}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRL}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WE}$	R	H	H	H	H
	W	L	L	L	L
$\overline{ICIOWR}$	R	H	H	H	H
	W	H	H	H	H
A29 to A0		Address	Address	Address	Address
D31 to D24		Hi-Z	Hi-Z	Hi-Z	Hi-Z
D23 to D16		Hi-Z	Hi-Z	Hi-Z	Hi-Z
D15 to D8		Hi-Z	Data	Hi-Z	Data
D7 to D0		Data	Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Enabled: Card enable signals corresponding to accessed areas = Low.
The other card enable signals = High.

Table C.1 Pin States of Bus Related Signals (15)

		External Space (PCMCIA I/O Card Interface)			
			16-bit Space		
Pin Name		**8-bit Space**	**Upper Byte**	**Lower Byte**	**Word/Longword**
$\overline{CS0}$ to $\overline{CS8}$		H	H	H	H
$\overline{CE1A}$, $\overline{CE1B}$, $\overline{CE2A}$, $\overline{CE2B}$		Enabled	Enabled	Enabled	Enabled
$\overline{BS}$		L	L	L	L
$\overline{RASU}$, $\overline{RASL}$		H	H	H	H
$\overline{CASU}$, $\overline{CASL}$		H	H	H	H
DQMUU		H	H	H	H
DQMUL		H	H	H	H
DQMLU		H	H	H	H
DQMLL		H	H	H	H
$\overline{AH}$		L	L	L	L
$\overline{FRAME}$		H	H	H	H
RDWR	R	H	H	H	H
	W	L	L	L	L
$\overline{RD}$	R	H	H	H	H
	W	H	H	H	H
$\overline{ICIORD}$	R	L	L	L	L
	W	H	H	H	H
$\overline{WRHH}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRHL}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRH}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WRL}$	R	H	H	H	H
	W	H	H	H	H
$\overline{WE}$	R	H	H	H	H
	W	H	H	H	H

		External Space (PCMCIA I/O Card Interface)			
			16-bit Space		
Pin Name		**8-bit Space**	**Upper Byte**	**Lower Byte**	**Word/Longword**
$\overline{\text{ICIOWR}}$	R	H	H	H	H
	W	L	L	L	L
A29 to A0		Address	Address	Address	Address
D31 to D24		Hi-Z	Hi-Z	Hi-Z	Hi-Z
D23 to D16		Hi-Z	Hi-Z	Hi-Z	Hi-Z
D15 to D8		Hi-Z	Data	Hi-Z	Data
D7 to D0		Data	Hi-Z	Data	Data

[Legend]

R: Read

W: Write

Enabled: Card enable signals corresponding to accessed areas = Low.
The other card enable signals = High.

D. Product Code Lineup

Table D.1 Product Code Lineup

Product Type							
Product Name	Classification	ROM Capacity	RAM Capacity	Application	Operating temperature	Product Code	Package (Package Code)
SH7083	F-ZTAT version	256 kbytes	16 kbytes	Consumer application	–20 to +85°C	R5F70834AN80FTV	TQFP1414-100 (TFP-100BV)
				Industrial application	–40 to +85°C	R5F70834AD80FTV	
		512 kbytes	32 kbytes	Consumer application	–20 to +85°C	R5F70835AN80FTV	
				Industrial application	–40 to +85°C	R5F70835AD80FTV	
		256 kbytes	16 kbytes	Consumer application	–20 to +85°C	R5F70834AN80BGV	P-LFBGA-112 (BP-112V)
				Industrial application	–40 to +85°C	R5F70834AD80BGV	
		512 kbytes	32 kbytes	Consumer application	–20 to +85°C	R5F70835AN80BGV	
				Industrial application	–40 to +85°C	R5F70835AD80BGV	
	Mask ROM version	256 kbytes	16 kbytes	Consumer application	–20 to +85°C	R5M70834ANXXXFTV※2	TQFP1414-100 (TFP-100BV)
				Industrial application	–40 to +85°C	R5M70834ADXXXFTV※2	
		256 kbytes	16 kbytes	Consumer application	–20 to +85°C	R5M70834ANXXXBGV※2	P-LFBGA-112 (BP-112V)
				Industrial application	–40 to +85°C	R5M70834ADXXXBGV※2	
	ROM-less version	0 kbyte	16 kbytes	Consumer application	–20 to +85°C	R5S70830AN80FTV	TQFP1414-100 (TFP-100BV)
				Industrial application	–40 to +85°C	R5S70830AD80FTV	
		0 kbyte	16 kbytes	Consumer application	–20 to +85°C	R5S70830AN80BGV	P-LFBGA-112 (BP-112V)
				Industrial application	–40 to +85°C	R5S70830AD80BGV	
	F-ZTAT version supporting full functions of E10A※1	512 kbytes	32 kbytes	For system development only※1	0 to +50°C	R5E70835RN80FTV	TQFP1414-100 (TFP-100BV)
SH7084	F-ZTAT version	256 kbytes	16 kbytes	Consumer application	–20 to +85°C	R5F70844AN80FPV	LQFP2020-112 (FP-112EV)
				Industrial application	–40 to +85°C	R5F70844AD80FPV	
		512 kbytes	32 kbytes	Consumer application	–20 to +85°C	R5F70845AN80FPV	
				Industrial application	–40 to +85°C	R5F70845AD80FPV	
	Mask ROM version	256 kbytes	16 kbytes	Consumer application	–20 to +85°C	R5M70844ANXXXFPV※2	
				Industrial application	–40 to +85°C	R5M70844ADXXXFPV※2	
	ROM-less version	0 kbyte	16 kbytes	Consumer application	–20 to +85°C	R5S70840AN80FPV	
				Industrial application	–40 to +85°C	R5S70840AD80FPV	
	F-ZTAT version supporting full functions of E10A※1	512 kbytes	32 kbytes	For system development only※1	0 to +50°C	R5E70845RN80FPV	

Product Type							
Product Name	**Classification**	**ROM Capacity**	**RAM Capacity**	**Application**	**Operating temperature**	**Product Code**	**Package (Package Code)**
SH7085	F-ZTAT version	256 kbytes	16 kbytes	Consumer application	−20 to +85°C	R5F70854AN80FPV	LQFP2020-144 (FP-144LV)
				Industrial application	−40 to +85°C	R5F70854AD80FPV	
		512 kbytes	32 kbytes	Consumer application	−20 to +85°C	R5F70855AN80FPV	
				Industrial application	−40 to +85°C	R5F70855AD80FPV	
	Mask ROM version	256 kbytes	16 kbytes	Consumer application	−20 to +85°C	R5M70854ANXXXFPV※2	
				Industrial application	−40 to +85°C	R5M70854ADXXXFPV※2	
	ROM-less version	0 kbyte	16 kbytes	Consumer application	−20 to +85°C	R5S70850AN80FPV	
				Industrial application	−40 to +85°C	R5S70850AD80FPV	
	F-ZTAT version supporting full functions of E10A※1	512 kbytes	32 kbytes	For system development only※1	0 to +50°C	R5E70855RN80FPV	
SH7086	F-ZTAT version	512 kbytes	32 kbytes	Consumer application	−20 to +85°C	R5F70865AN80FPV	LQFP2424-176 (FP-176EV)
				Industrial application	−40 to +85°C	R5F70865AD80FPV	
	F-ZTAT version supporting full functions of E10A※1			For system development only※1	0 to +50°C	R5E70865RN80FPV	

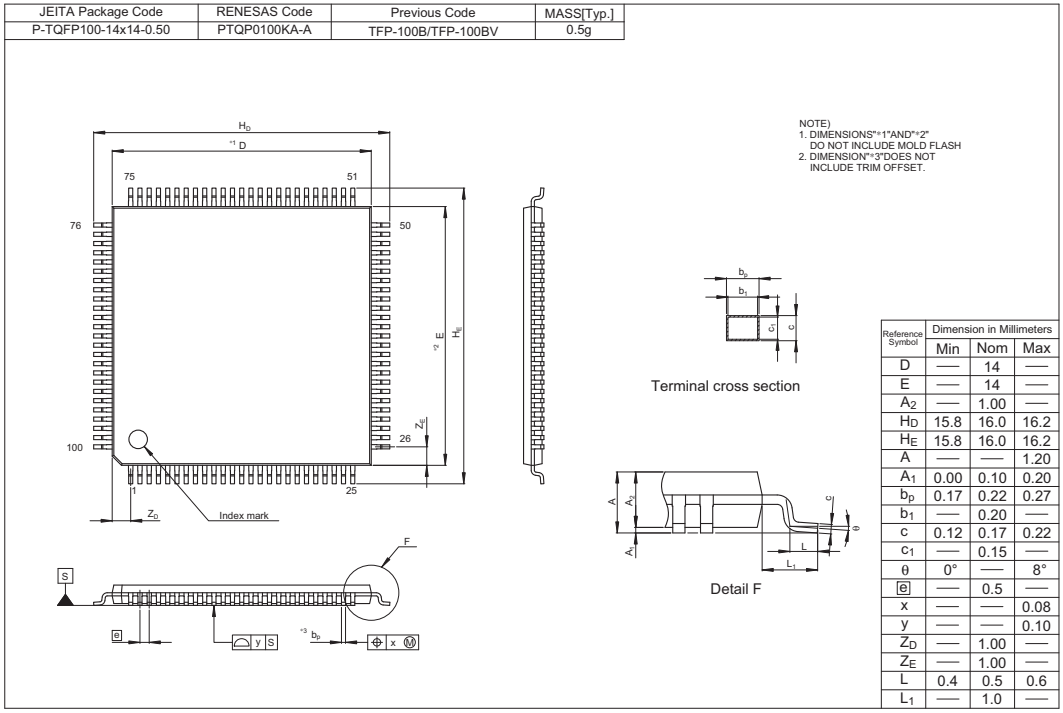
Notes: 1. These products are only used for system development by the customer, and E10A internal bus trace function and AUD function are available. However, normal F-ZTAT version or mask ROM version must be used in mass production.
In normal F-ZTAT version, the E10A internal bus trace function and AUD function are not available. The reliability is not is not guaranteed in the F-ZTAT version supporting full functions of E10A.

2. XXX indicates the ROM code.

E. Package Dimensions

The package dimension that is shown in the Renesas Semiconductor Package Data Book has Priority.

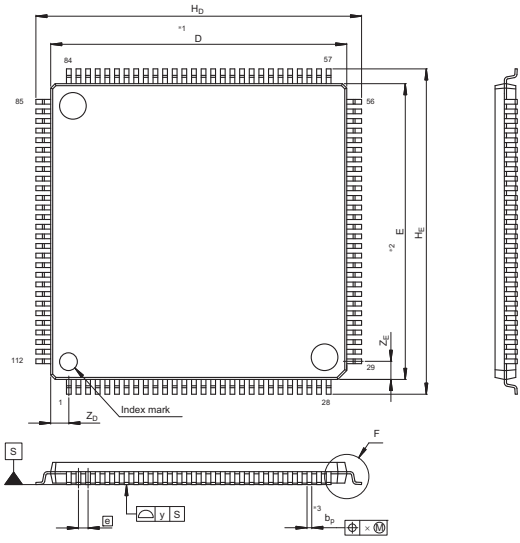
JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-TQFP100-14x14-0.50	PTQP0100KA-A	TFP-100B/TFP-100BV	0.5g



Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	—	14	—
E	—	14	—
A_2	—	1.00	—
H_D	15.8	16.0	16.2
H_E	15.8	16.0	16.2
A	—	—	1.20
A_1	0.00	0.10	0.20
b_p	0.17	0.22	0.27
b_1	—	0.20	—
c	0.12	0.17	0.22
c_1	—	0.15	—
θ	0°	—	8°
e	—	0.5	—
x	—	—	0.08
y	—	—	0.10
Z_D	—	1.00	—
Z_E	—	1.00	—
L	0.4	0.5	0.6
L_1	—	1.0	—

Figure E.1 TFP-100BV

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LQFP112-20x20-0.65	PLQP0112JA-A	FP-112E/FP-112EV	1.2g

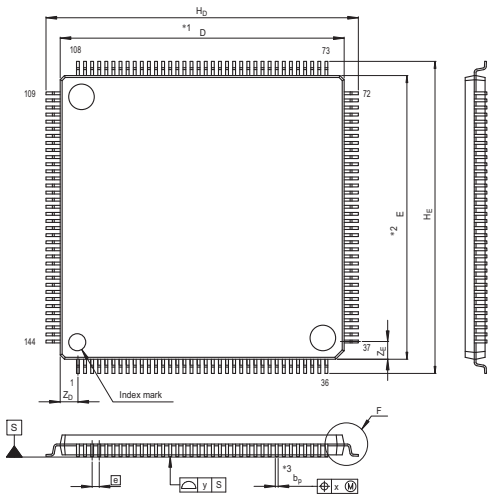


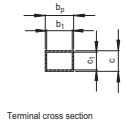
NOTE)
1. DIMENSIONS "*1" AND "*2" DO NOT INCLUDE MOLD FLASH.
2. DIMENSION "*3" DOES NOT INCLUDE TRIM OFFSET.

Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	19.9	20.0	20.1
E	19.9	20.0	20.1
A_2	—	1.4	—
H_D	21.8	22.0	22.2
H_E	21.8	22.0	22.2
A	—	—	1.7
A_1	0.05	0.1	0.15
b_p	0.27	0.32	0.37
b_1	—	0.30	—
c	0.09	0.145	0.20
c_1	—	0.125	—
θ	0°	—	8°
e	—	0.65	—
x	—	—	0.13
y	—	—	0.10
Z_D	—	1.225	—
Z_E	—	1.225	—
L	0.35	0.5	0.65
L_1	—	1.0	—

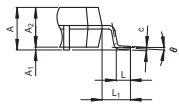
Figure E.2 FP-112EV

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LQFP144-20x20-0.50	PLQP0144KA-A	144P6Q-A / FP-144L / FP-144LV	1.2g





Terminal cross section



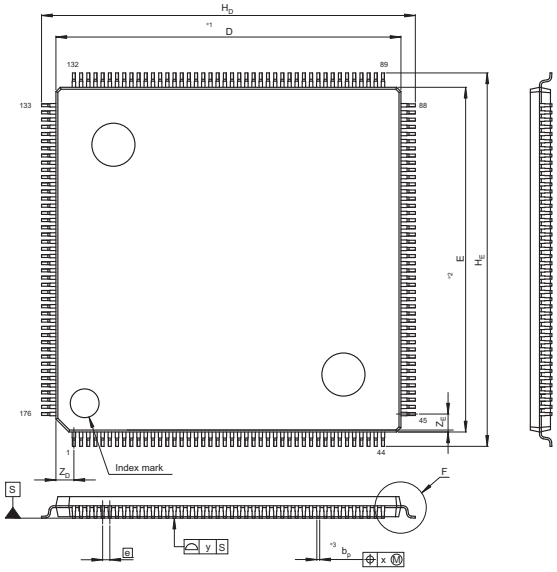
Detail F

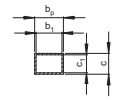
NOTE)
1. DIMENSIONS "*1" AND "*2" DO NOT INCLUDE MOLD FLASH.
2. DIMENSION "*3" DOES NOT INCLUDE TRIM OFFSET.

Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	19.9	20.0	20.1
E	19.9	20.0	20.1
A_2	—	1.4	—
H_D	21.8	22.0	22.2
H_E	21.8	22.0	22.2
A	—	—	1.7
A_1	0.05	0.1	0.15
b_p	0.17	0.22	0.27
b_1	—	0.20	—
c	0.09	0.145	0.20
c_1	—	0.125	—
θ	0°	—	8°
e	—	0.5	—
x	—	—	0.08
y	—	—	0.10
Z_D	—	1.25	—
Z_E	—	1.25	—
L	0.35	0.5	0.65
L_1	—	1.0	—

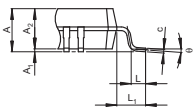
Figure E.3 FP-144LV

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LQFP176-24x24-0.50	PLQP0176KB-A	176P6Q-A/FP-176E/FP-176EV	1.8g





Terminal cross section



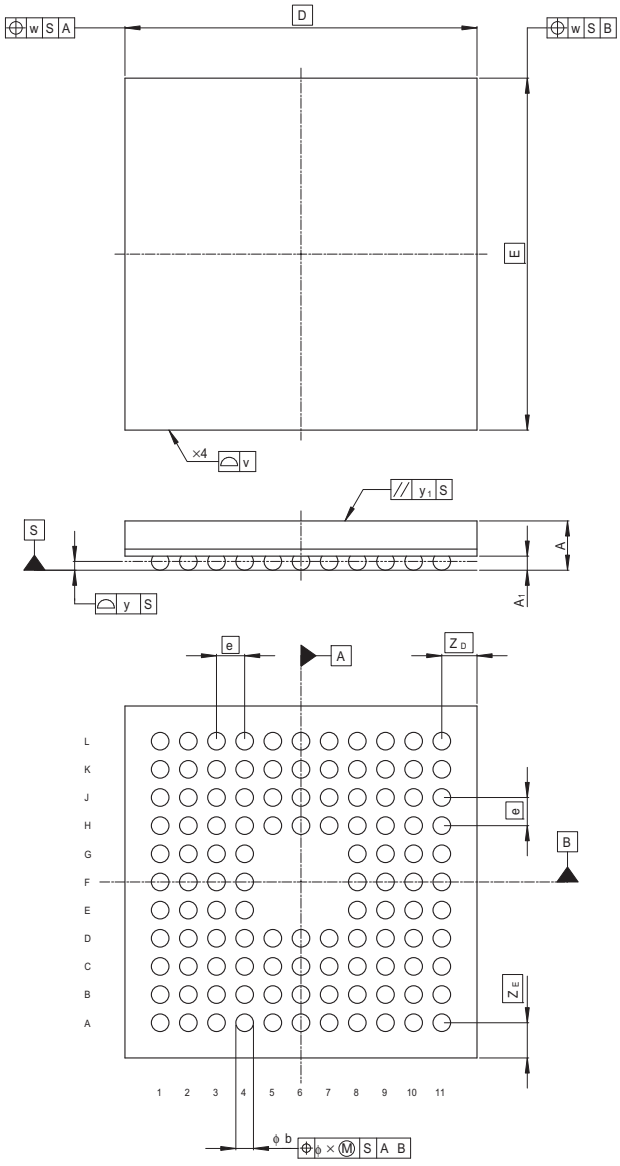
Detail F

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	23.9	24.0	24.1
E	23.9	24.0	24.1
A_2	—	1.4	—
H_D	25.8	26.0	26.2
H_E	25.8	26.0	26.2
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	—	—	0.10
Z_D	—	1.25	—
Z_E	—	1.25	—
L	0.35	0.5	0.65
L_1	—	1.0	—

Figure E.4 FP-176EV

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LFBGA112-10x10-0.80	PLBG0112GA-A	BP-112/BP-112V	0.3g



Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	—	10.00	—
E	—	10.00	—
v	—	—	0.15
w	—	—	0.20
A	—	—	1.40
A_1	0.35	0.40	0.45
e	—	0.80	—
b	0.45	0.50	0.55
x	—	—	0.08
y	—	—	0.10
y_1	—	—	0.2
S_D	—	—	—
S_E	—	—	—
Z_D	—	1.00	—
Z_E	—	1.00	—

Figure E.5 BP-112V

Main Revisions for This Edition

Item	Page	Revision (See Manual for Details)
Overall	—	"I²C2" changed to "IIC2"
3.1 Selection of Operating Modes Table 3.1 Selection of Operating Modes*1	55	Table amended

	Pin Setting					Bus Width of CS0 Space			
Mode No.	FWE	MD1	MD0	Mode Name	On-Chip ROM	SH7083	SH7084	SH7085	SH7086
Mode 0	0	0	0	MCU extension mode 0	Not active	8	8	16	16
Mode 1	0	0	1	MCU extension mode 1	Not active	16	16	32	32
Mode 2	0	1	0	MCU extension mode 2	Active	Set by CS0BCR in BSC			
Mode 3	0	1	1	Single chip mode	Active	—			
Mode 4*2	1	0	0	Boot mode	Active	—			
Mode 5*2	1	0	1	User boot mode	Active	Set by CS0BCR in BSC			
Mode 6*2	1	1	0	User programming mode	Active	Set by CS0BCR in BSC			
Mode 7*2	1	1	1			—			

Notes: 1. Do not input a low-level signal to $\overline{\text{ASEMD0}}$ when the MCU is not connected to the E10A. Operation cannot be guaranteed if a low-level signal is input to $\overline{\text{ASEMD0}}$ when the E10A is not connected. For information on connecting the E10A, see SuperH™ Family E10A-USB Emulator, Additional Document for User's Manual: Supplementary Information on Using the SH7083, SH7084, SH7085, and SH7086.

2. These are flash memory programming modes.

5.8.4 Notes on Slot Illegal Instruction Exception Handling	103	Description deleted 3. Others The slot illegal instruction exception handling may be generated in this LSI in a case where the instruction is described in assembler or when the middleware of the object is introduced.
8.2.2 DTC Mode Register B (MRB)	176	Table amended

Bit	Bit Name	Initial Value	R/W	Description
5	DISEL	Undefined	—	DTC Interrupt Select When this bit is set to 1, an interrupt request is generated to the CPU every time a data transfer or a block transfer ends. When this bit is set to 0, a CPU interrupt request is only generated when the specified number of data transfers end. Note: This bit should be cleared to 0 when the IIC2 is selected as the activation source.

Item | **Page** | **Revision (See Manual for Details)**

9.4.3 CSn Space Wait Control Register (CSnWCR) (n = 0 to 8)

(2) MPX-I/O

254, 256

Table amended

Bit	Bit Name	Initial Value	R/W	Description
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from the End of the Address Cycles (Ta3) to $\overline{RD}$ and $\overline{WRxx}$ Assertion Specify the number of delay cycles from the end of the address cycles (Ta3) to $\overline{RD}$ and $\overline{WRxx}$ assertion. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles
1, 0	HW[1:0]	00	R/W	Delay Cycles from $\overline{RD}$ and $\overline{WRxx}$ Negation to $\overline{CSn}$ Negation Specify the number of delay cycles from $\overline{RD}$ and $\overline{WRxx}$ negation to $\overline{CSn}$ negation. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

9.5.5 MPX-I/O Interface

297

Description amended

The data cycle is the same as that in a normal space access.

The delay cycle of SW[1:0] is inserted between Ta3 and T1 cycle.

The delay cycle of HW[1:0] is added after T2 cycle.

Timing charts are shown in figures 9.11 to 9.14.

Figure 9.12 Access Timing for MPX Space (Address Cycle No Wait, Assert Period Expansion 1.5, Data Cycle No Wait, Negation Period Expansion 1.5)

299

Figure added

9.5.16 Access to On-Chip Peripheral I/O Registers by CPU

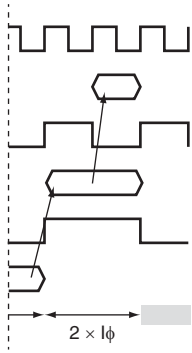
Table 9.35 Number of Cycles for Access to On-Chip Peripheral I/O Registers

379

Table and note amended

		Number of Access Cycles[*1,*2]
Pϕ reference	Write	(3 + n) × Iϕ + (1 + m) × Bϕ + 2 × Pϕ[*3]
	Read	(3 + n) × Iϕ + (1 + m) × Bϕ + 2 × Pϕ[*3] + 2 × Iϕ
Bϕ reference	Write	(3 + n) × Iϕ + 3 × Bϕ[*4]
	Read	(3 + n) × Iϕ + 3 × Bϕ[*4,*5] + 2Iϕ

Notes: 1. When Iϕ:Bϕ = 8:1, n = 0 to 7.
...
2. The clock ratio of MIϕ and MPϕ does not affect the number of access cycles.
3. The access cycle count is 5 × Pϕ for all flash registers other than RAMER.
4. The access cycle count of the RAMER flash register is 5 × Bϕ.
5. The access cycle count is 1 × Bϕ for all BSC registers other than BSCEHR.

Item	Page	Revision (See Manual for Details)
9.5.16 Access to On-Chip Peripheral I/O Registers by CPU Figure 9.54 Timing of Read Access to On-Chip Peripheral I/O Registers When Iϕ:Bϕ:Pϕ = 4:2:1	381	Figure amended 
11.1 Features	429	Description added • Dead time compensation counter available in channel 5 • External pulse width measurement available in channel 5
Table 11.1 MTU2 Functions	431	Table amended Item: External pulse width measurement Channel 0: — Channel 1: — Channel 2: — Channel 3: — Channel 4: — Channel 5: √
11.3.1 Timer Control Register (TCR) • TCRU_5, TCRV_5, TCRW_5	444	Section added
11.3.2 Timer Mode Register (TMDR)	446	Table note added Notes: 1. To enable buffer operation in reset-synchronized PWM mode, set the BFB or BFA bit to 1 on channel 3 and clear the BFB or BFA bit to 0 on channel 4. Buffer operation on channel 4 takes place according to the settings for channel 3. 2. To enable buffer operation in complementary PWM mode, set the BFB or BFA bit to 1 on channel 3. The settings of the BFB or BFA bit on channel 4 are ignored. Buffer operation on channel 4 takes place according to the settings for channel 3.

Item	Page	Revision (See Manual for Details)
11.3.5 Timer Interrupt Enable Register (TIER)	469, 470	Table amended

Bit	Bit Name	Initial Value	R/W	Description
3	TGIED	0	R/W	TGR Interrupt Enable D Enables or disables interrupt requests (TGID) by the TGFD bit when the TGFD bit in TSR is set to 1 in channels 0, 3, and 4. In complementary PWM mode, clear the TGIED bit to 0 on channels 3 and 4. In channels 1 and 2, bit 3 is reserved. It is always read as 0 and the write value should always be 0. 0: Interrupt requests (TGID) by TGFD bit disabled 1: Interrupt requests (TGID) by TGFD bit enabled
2	TGIEC	0	R/W	TGR Interrupt Enable C Enables or disables interrupt requests (TGIC) by the TGFC bit when the TGFC bit in TSR is set to 1 in channels 0, 3, and 4. In complementary PWM mode, clear the TGIEC bit to 0 on channel 4. In channels 1 and 2, bit 2 is reserved. It is always read as 0 and the write value should always be 0. 0: Interrupt requests (TGIC) by TGFC bit disabled 1: Interrupt requests (TGIC) by TGFC bit enabled

11.3.6 Timer Status Register (TSR)	476	Table note added Notes: 3. TGFC or TGFD may be set to 1 when a compare match occurs during the Tb interval even when the BFB and BFA bits in TMDR have been set to 1 to make TGRC and TGRD operate as buffers in complementary PWM mode.
11.3.9 Timer Synchronous Clear Register (TSYCR)	484	Description amended TSYCR is an 8-bit readable/writable register that specifies conditions for clearing TCNT_3S and TCNT_4S in the MTU2S in synchronization with the MTU2. The MTU2S has one TSYCRS in channel 3 but the MTU2 has no TSYCR.
11.3.10 Timer A/D Converter Start Request Control Register (TADCR)	486	Bit table note amended Notes: Accessing TADCR in 8-bit units is prohibited. Always access TADCR in 16-bit units. * Set to 0 when complementary PWM mode is not selected.

Item	Page	Revision (See Manual for Details)
11.3.10 Timer A/D Converter Start Request Control Register (TADCR)	486	Table amended

Bit	Bit Name	Initial Value	R/W	Description
7	UT4AE	0	R/W	Up-Count TRG4AN Enable Enables or disables A/D converter start requests (TRG4AN) during TCNT_4 up-count operation. 0: A/D converter start requests (TRG4AN) are disabled during TCNT_4 up-count operation 1: A/D converter start requests (TRG4AN) are enabled during TCNT_4 up-count operation
6	DT4AE	0*1	R/W	Down-Count TRG4AN Enable Enables or disables A/D converter start requests (TRG4AN) during TCNT_4 down-count operation. 0: A/D converter start requests (TRG4AN) are disabled during TCNT_4 down-count operation 1: A/D converter start requests (TRG4AN) are enabled during TCNT_4 down-count operation

	487	Table amended		

Bit	Bit Name	Initial Value	R/W	Description
5	UT4BE	0	R/W	Up-Count TRG4BN Enable Enables or disables A/D converter start requests (TRG4BN) during TCNT_4 up-count operation. 0: A/D converter start requests (TRG4BN) are disabled during TCNT_4 up-count operation 1: A/D converter start requests (TRG4BN) are enabled during TCNT_4 up-count operation
4	DT4BE	0*1	R/W	Down-Count TRG4BN Enable Enables or disables A/D converter start requests (TRG4BN) during TCNT_4 down-count operation. 0: A/D converter start requests (TRG4BN) are disabled during TCNT_4 down-count operation 1: A/D converter start requests (TRG4BN) are enabled during TCNT_4 down-count operation
3	ITA3AE	0*1*2*3	R/W	TGIA_3 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4AN) with TGIA_3 interrupt skipping operation. 0: Does not link with TGIA_3 interrupt skipping operation 1: Links with TGIA_3 interrupt skipping operation
2	ITA4VE	0*1*2*3	R/W	TCIV_4 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4AN) with TCIV_4 interrupt skipping operation. 0: Does not link with TCIV_4 interrupt skipping operation 1: Links with TCIV_4 interrupt skipping operation
1	ITB3AE	0*1*2*3	R/W	TGIA_3 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4BN) with TGIA_3 interrupt skipping operation. 0: Does not link with TGIA_3 interrupt skipping operation 1: Links with TGIA_3 interrupt skipping operation

Item	Page	Revision (See Manual for Details)
11.3.10 Timer A/D Converter Start Request Control Register (TADCR)	488	Table amended

Bit	Bit Name	Initial Value	R/W	Description
0	ITB4VE	0*1*2*3	R/W	TCIV_4 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4BN) with TCIV_4 interrupt skipping operation. 0: Does not link with TCIV_4 interrupt skipping operation 1: Links with TCIV_4 interrupt skipping operation

Table note amended

Notes: 1. Set to 0 when complementary PWM mode is not selected.

2. Clear this bit to 0 when interrupt skipping is disabled (when the T3AEN or T4VEN bit in the timer interrupt skipping set register (TITCR) is cleared to 0 or when the skipping count set bit (3ACOR or 4VCOR) in TITCR is cleared to 0).

3. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D converter start requests will not be issued.

Item	Page	Revision (See Manual for Details)
11.3.10 Timer A/D Converter Start Request Control Register (TADCR) Table 11.29 Setting of Transfer Timing by BF[1:0] Bits	489, 490	Table amended

Bit 15	Bit 14	Description	
BF1	BF0	Complementary PWM Mode	Reset-Synchronized PWM Mode
0	0	Does not transfer data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4).	Does not transfer data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4).
0	1	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) at the crest of TCNT_4.	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) when a compare match occurs between TCNT_3 and TGRA_3.
1	0	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) at the trough of TCNT_4.	Setting prohibited
1	1	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) at the crest and trough of TCNT_4.	Setting prohibited

Bit 15	Bit 14	Description	
BF1	BF0	PWM Mode 1	Normal Mode
0	0	Does not transfer data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4).	Does not transfer data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4).
0	1	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) when a compare match occurs between TCNT_4 and TGRA_4.	Transfers data from the cycle set buffer register (TADCOBRA_4/TADCOBRB_4) to the cycle set register (TADCORA_4/TADCORB_4) when a compare match occurs between TCNT_4 and TGRA_4.
1	0	Setting prohibited	Setting prohibited
1	1	Setting prohibited	Setting prohibited

11.3.19 Timer Output Master Enable Register (TOER)	501	Description added TOER is an 8-bit readable/writable register that enables/disables output settings for output pins TIOC4D, TIOC4C, TIOC3D, TIOC4B, TIOC4A, and TIOC3B. ... Set TOER when count operation of TCNT channels 3 and 4 is halted (See figures 11.35 and 11.38).

Item	Page	Revision (See Manual for Details)
11.3.20 Timer Output Control Register 1 (TOCR1)	503	Table note added Notes: 1. This bit can be set to 1 only once after a power-on 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 TOCS0 bit to 0 makes this bit setting valid. 4. The inverse-phase output is the exact inverse of the positive-phase output unless dead time is generated. When no dead time is generated, only the OLSP setting is valid
11.3.21 Timer Output Control Register 2 (TOCR2)	506	Table note added Notes: 1. Setting the TOCS bit in TOCR1 to 1 makes this bit setting valid. 2. The inverse-phase output is the exact inverse of the positive-phase output unless dead time is generated. When no dead time is generated, only the OLSiP setting is valid.

Item	Page	Revision (See Manual for Details)
11.3.23 Timer Gate Control Register (TGCR)	509, 510	Table and table note amended

Bit	Bit Name	Initial value	R/W	Description
6	BDC	0	R/W	Brushless DC Motor This bit selects whether to make the functions of this register (TGCR) effective or ineffective. 0: The function of this register is disabled. 1: The function of this register is enabled.
5	N	0	R/W	Reverse Phase Output (N) Control This bit selects between level output by the output level select function (table 11.39) and reset-synchronized PWM/complementary PWM output when a reverse pin (TIOC3D, TIOC4C, or TIOC3D) is in the on state. 0: Level output 1: Reset synchronized PWM/complementary PWM output
4	P	0	R/W	Positive Phase Output (P) Control This bit selects between level output by the output level select function (table 11.39) and reset-synchronized PWM/complementary PWM output when a positive pin (TIOC3B, TIOC4A, and TIOC4B) is in the on state. 0: Level output 1: Reset synchronized PWM/complementary PWM output
3	FB*	0	R/W	External Feedback Signal Enable This bit selects whether the switching of the output of the positive/reverse phase is carried out automatically with the TIOC0A, TIOC0B, TIOC0C input signals or the UF, VF, and WF bits in TGCR. 0: Output switching is external input (TIOC0A, TIOC0B, TIOC0C) 1: Output switching is carried out by software (UF, VF, WF settings).
2	WF	0	R/W	Output Phase Switch
1	VF	0	R/W	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. See table 11.39.
0	UF	0	R/W	

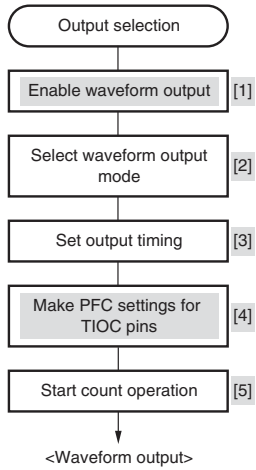
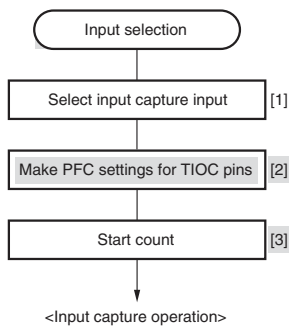
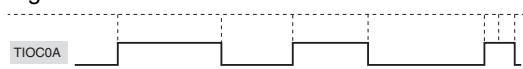
Note: * When the MTU2S is used to set the BDC bit to 1, set the FB bit to 1.

Table 11.39 Output level Select Function	511	Table amended

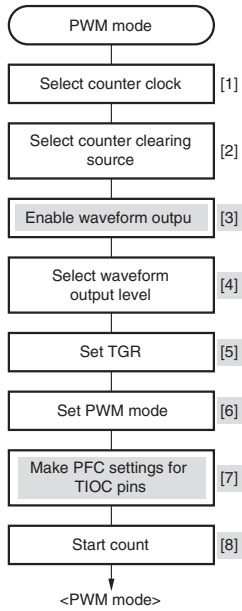
Bit 2	Bit 1	Bit 0
WF	VF	UF
(TIOC0C)	(TIOC0B)	(TIOC0A)

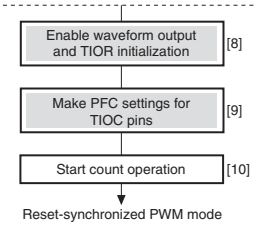
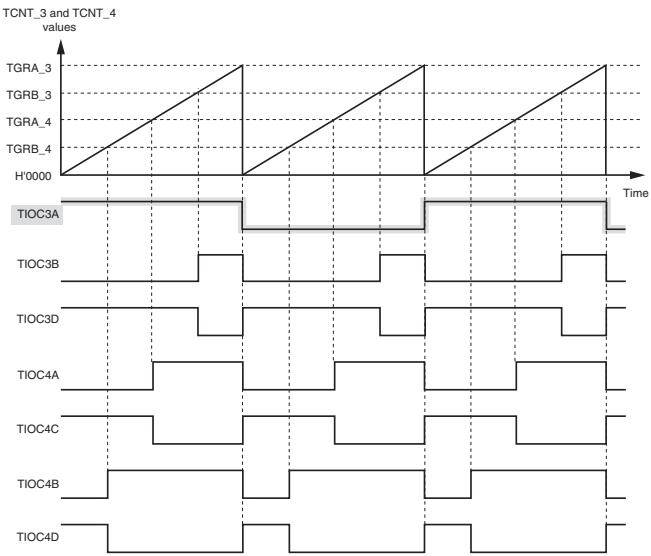
11.3.26 Timer Cycle Data Register (TCDR)	512	Description added TCDR is a 16-bit register used only in complementary PWM mode. Set half the PWM carrier sync value (note that this value should be at least double the value specified in TDDR + 3) as the TCDR register value.

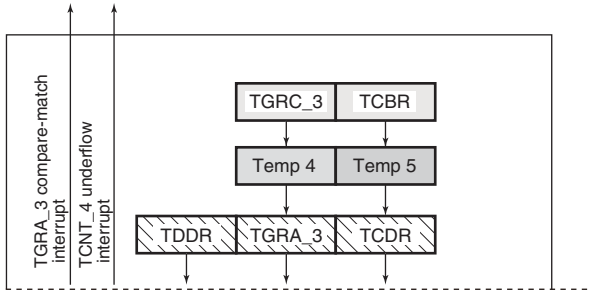
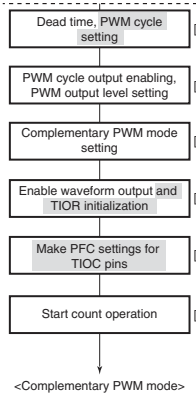
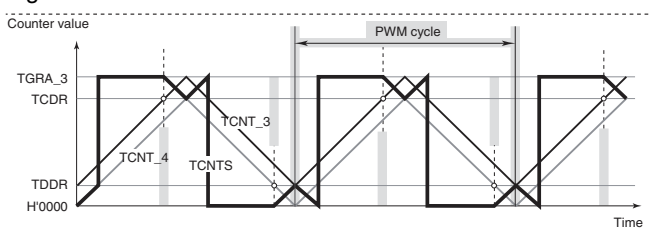
Item	Page	Revision (See Manual for Details)
11.3.30 Timer Buffer Transfer Set Register (TBTER) Table 11.42 Setting of Bits BTE[1:0]	517	Table and note amended **Bit 1 / Bit 0** **BTE1 / BTE0 — Description** 0 0 — Enables transfer from the buffer registers to the temporary registers*1 and does not link the transfer with interrupt skipping operation. 0 1 — Disables transfer from the buffer registers to the temporary registers.*1 1 0 — Links transfer from the buffer registers to the temporary registers with interrupt skipping operation.*1*2 1 1 — Setting prohibited Notes: 1. Transfers from the temporary registers to the compare registers take place in accordance with the setting of the MD[3:0] bit field in TMDR, regardless of the setting of the BTE[1:0] bit field. For details, refer to section 11.4.8, Complementary PWM Mode.
11.3.32 Timer Waveform Control Register (TWCR)	519	Description amended TWCR is an 8-bit readable/writable register. It controls the output waveform when synchronous counter clearing of TCNT_3 and TCNT_4 occurs in complementary PWM mode, specifies the MTU2-MTU2S counter synchronous clearing setting, and specifies whether or not counter clearing occurs at TGRA_3 compare match. The CCE bit and WRE bit in TWCR must be modified only while TCNT stops. Bit table note amended Note: * Set to 0 when complementary PWM mode is not selected.
	520	Table amended **Bit / Bit Name / Initial Value / R/W / Description** 1 / SCC / 0 / R/(W) / Synchronous Clearing Control (only in MTU2S) Specifies whether to clear TCNT_3S and TCNT_4S in the MTU2S when synchronous counter clearing between the MTU2 and MTU2S occurs in complementary PWM mode. When using this control, place the MTU2S in complementary PWM mode. When modifying the SCC bit while the counters are operating, do not modify the CCE or WRE bits. Counter clearing synchronized with the MTU2 is disabled by the SCC bit setting only when synchronous clearing occurs outside the Tb interval at the trough. When synchronous clearing occurs in the Tb interval at the trough including the period immediately after TCNT_3S and TCNT_4S start operation, TCNT_3S and TCNT_4S in the MTU2S are cleared. For the Tb interval at the trough in complementary PWM mode, see figure 11.40. In the MTU2, this bit is reserved. It is always read as 0 and the write value should always be 0. 0: Enables clearing of TCNT_3S and TCNT_4S in the MTU2S by MTU2–MTU2S synchronous clearing operation 1: Disables clearing of TCNT_3S and TCNT_4S in the MTU2S by MTU2–MTU2S synchronous clearing operation [Setting condition] • When 1 is written to SCC after reading SCC = 0

Item	Page	Revision (See Manual for Details)
11.3.32 Timer Waveform Control Register (TWCR)	521	Table note amended Note: * Set to 0 when complementary PWM mode is not selected.
11.4.1 Basic Functions Figure 11.7 Example of Setting Procedure for Waveform Output by Compare Match	524	Figure amended  [1] Enable waveform output on the TIOC pins of channels 3 and 4 in TOER. Make settings to TOER before TIOR. [2] Select initial value 0 output or 1 output, and compare match output value 0 output, 1 output, or toggle output, by means of TIOR. The set initial value is output at the TIOC pin until the first compare match occurs. [3] Set the timing for compare match generation in TGR. [4] Make settings to the PFC (port control register and port I/O register) for the TIOC pins to be used for waveform output. Set the corresponding IOR bit in the port I/O register to 1 to set each TIOC pin to output. [5] Set the CST bit in TSTR to 1 to start the count operation.
Figure 11.10 Example of Input Capture Operation Setting Procedure	526	Figure amended  [1] Designate TGR as an input capture register by means of TIOR, and select rising edge, falling edge, or both edges as the input capture source and input signal edge. [2] Make settings to the PFC (port control register and port I/O register) for the TIOC pins to be used for input capture input. Clear the corresponding IOR bit in the port I/O register to 0 to set each TIOC pin to input. [3] Set the CST bit in TSTR to 1 to start the count operation.
11.4.3 Buffer Operation Figure 11.17 Example of Buffer Operation (1)	532	Figure amended 
Figure 11.19 Example of Buffer Operation When TCNT_0 Clearing Is Selected for TGRC_0 to TGRA_0 Transfer Timing	534	Figure amended 

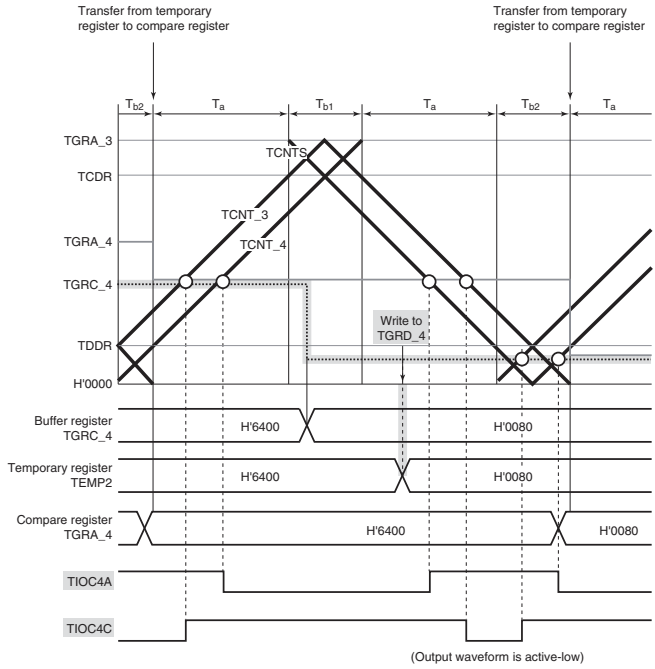
Item	Page	Revision (See Manual for Details)
11.4.4 Cascaded Operation	534, 535	Description amended For simultaneous input capture of TCNT_1 and TCNT_2 during cascaded operation, additional input capture input pins can be specified by the input capture control register (TICCR). Edge detection as the condition for input capture is the detection of edges in the signal produced by taking the logical OR of the signals on the main and additional pins. For details, refer to (4), Cascaded Operation Example (c). For input capture in cascade connection, refer to section 11.7.22, Simultaneous Capture of TCNT_1 and TCNT_2 in Cascade Connection.
	535	Description amended **Cascaded Operation Example (a):** Figure 11.21 illustrates the operation when TCNT_2 overflow/underflow counting has been set for TCNT_1 and phase counting mode 1 has been designated for channel 2.
Figure 11.23 Cascaded Operation Example (c)	537	Figure note added Note: * When either of the input pins is at the high level, an edge on the other input pin does not act as an input capture condition.
11.4.5 PWM Modes	539	Description amended 2. PWM mode 2 PWM output is generated using one TGR as the cycle register and the others as duty registers. The output specified in TIOR is performed by means of compare matches. Upon counter clearing by a cycle register compare match, the output value of each pin is the initial value set in TIOR.

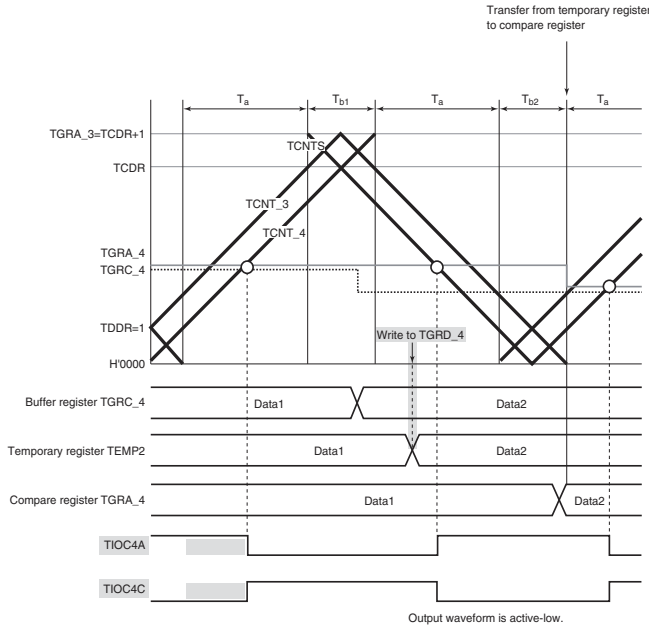
Item	Page	Revision (See Manual for Details)
11.4.5 PWM Modes Figure 11.25 Example of PWM Mode Setting Procedure	541	Figure amended  [1] Select the counter clock with bits TPSC[2:0] in TCR. At the same time, select the input clock edge with bits CKEG[1:0] in TCR. [2] Use bits CCLR[2:0] in TCR to select the TGR to be used as the TCNT clearing source. [3] For PWM output on the TIOC pins of channel 4, enable output in TOER. Set the value in TOER before setting TIOR. [4] Use TIOR to designate the TGR as an output compare register, and select the initial value and output value. [5] Set the cycle in the TGR selected in [2], and set the duty in the other TGR. [6] Select the PWM mode with bits MD[3:0] in TMDR. [7] Make settings to the PFC (port control register and port I/O register) for the TIOC pins to be used for PWM output. Set the corresponding IOR bit in the port I/O register to 1 to set each TIOC pin to output. [8] Set the CST bit in TSTR to 1 to start the count operation.
Figure 11.28 Example of PWM Mode Operation (PWM Waveform Output with 0% Duty and 100% Duty)	543	Figure title amended
11.4.6 Phase Counting Mode	544	Description amended The TCFD bit in TSR is the count direction flag. Reading the TCFD flag reveals whether TCNT is counting up or down. In phase counting mode, the external clock pins TCLKA, TCLKB, TCLKC, and TCLKD can be used as two-phase encoder pulse inputs. Table 11.47 shows the correspondence between external clock pins and channels.
11.4.7 Reset-Synchronized PWM Mode	552	Description amended When reset-synchronized PWM mode is selected, the TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, and TIOC4D pins function as PWM output pins, and the TIOC3A pin can be set as a toggle output synchronized with the PWM period. In addition, TCNT_3 and TCNT_4 function as up-counters.

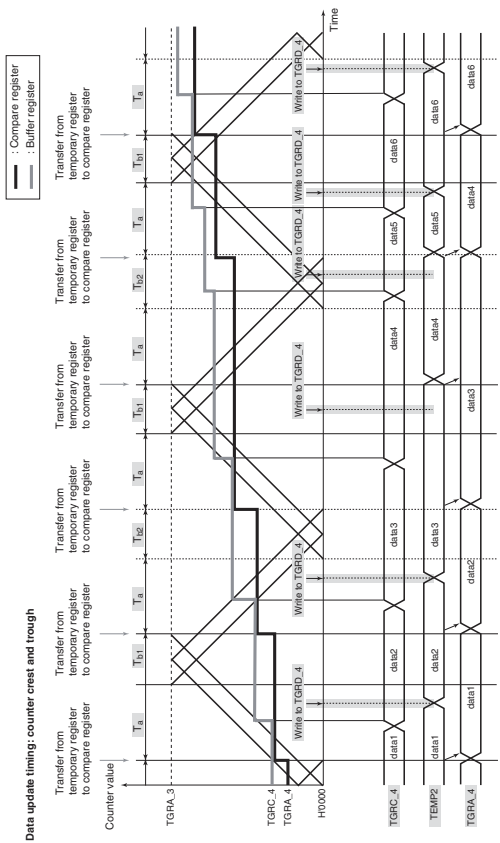
Item	Page	Revision (See Manual for Details)
11.4.7 Reset-Synchronized PWM Mode Table 11.52 Output Pins for Reset-Synchronized PWM Mode	552	Table amended **Channel** / **Output Pin** / **Description** 3 / TIOC3A / Toggle output synchronized with the PWM period (or I/O port)
Figure 11.35 Procedure for Selecting Reset-Synchronized PWM Mode	553	Figure amended  [7] Set bits MD[3:0] in TMDR_3 to B'1000 to select the reset-synchronized PWM mode. Do not set to TMDR_4. [8] Set the enabling/disabling of the PWM waveform output pin in TOER. Next, set TIORH_3, TIORL_3, TIORH_4, and TIORL_4 to H'00. [9] Make settings to the PFC (port control register and port I/O register) for the TIOC pins to be used for PWM input. Set the corresponding IOR bit in the port I/O register to 1 to set each TIOC pin to output. [10] Set the CST3 bit in the TSTR to 1 to start the count operation.
Figure 11.36 Reset-Synchronized PWM Mode Operation Example (When TOCR's OLSN = 1 and OLSP = 1)	554	Figure and note amended  Note: Do not change TOER settings when count operation is in progress.
11.4.8 Complementary PWM Mode	555	Description amended Table 11.54 shows the PWM output pins used. Table 11.55 shows the settings of the registers used. Figure 11.37 describes a block diagram of channels 3 and 4 in complementary PWM mode.

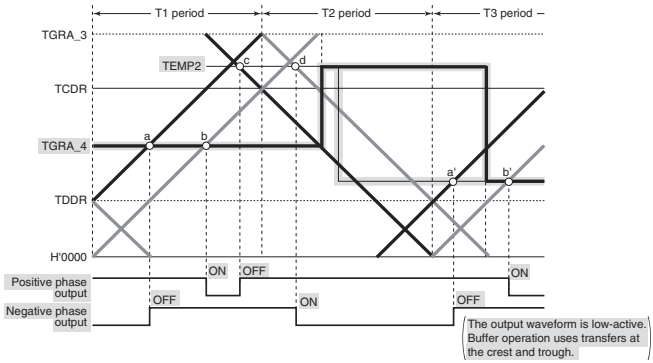
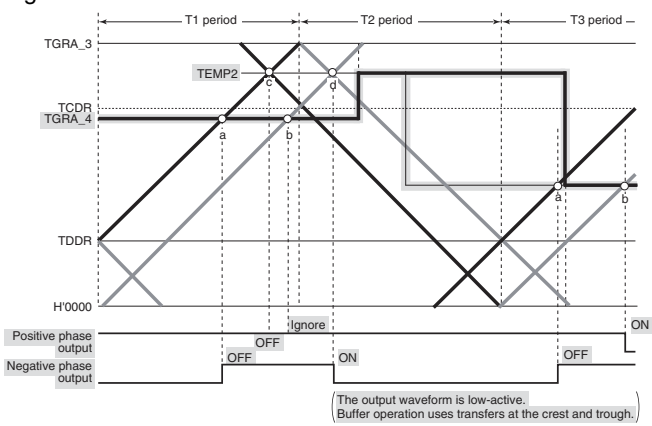
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode Table 11.55 Register Settings for Complementary PWM Mode	556	Figure amended **Channel / Counter/Register / Description / Read/Write from CPU** Temporary register 4 (TEMP4) / TGRA_3 temporary register / Not readable/writable Temporary register 5 (TEMP5) / TCDR temporary register / Not readable/writable
Figure 11.37 Block Diagram of Channels 3 and 4 in Complementary PWM Mode	557	Figure amended 
(1) Example of Complementary PWM Mode Setting Procedure Figure 11.38 Example of Complementary PWM Mode Setting Procedure	558	Figure amended  [7] This setting is necessary only when no dead time should be generated. Make appropriate settings in the timer dead time enable register (TDER) so that no dead time is generated. [8] Set the dead time in the dead time register (TDDR), 1/2 the PWM cycle in the timer cycle data register (TCDR) and timer cycle buffer register (TCBR), and 1/2 the PWM cycle plus the dead time in TGRA_3 and TGRC_3. When no dead time generation is selected, set 1 in TDDR and 1/2 the PWM cycle + 1 in TGRA_3 and TGRC_3. [9] Select enabling/disabling of toggle output synchronized with the PWM cycle using bit PSYE in the timer output control register 1 (TOCR1), and set the PWM output level with bits OLSP and OLSN. When specifying the PWM output level by using TOLBR as a buffer for TOCR_2, see figure 11.3. [10] Select complementary PWM mode in timer mode register 3 (TMDR_3). Do not set in TMDR_4. [11] Set enabling/disabling of PWM waveform output pin output in the timer output master enable register (TOER). Next, set TIORH_3, TIORL_3, TIORH_4, and TIORL_4 to H'00. [12] Make settings to the PFC (port control register and port I/O register) for the TIOC pins to be used for PWM input. Set the corresponding IOR bit in the port I/O register to 1 to set each TIOC pin to output. [13] Set bits CST3 and CST4 in TSTR to 1 simultaneously to start the count operation.
(2) Outline of Complementary PWM Mode Operation (a) Counter Operation Figure 11.39 Complementary PWM Mode Counter Operation	560	Figure amended 

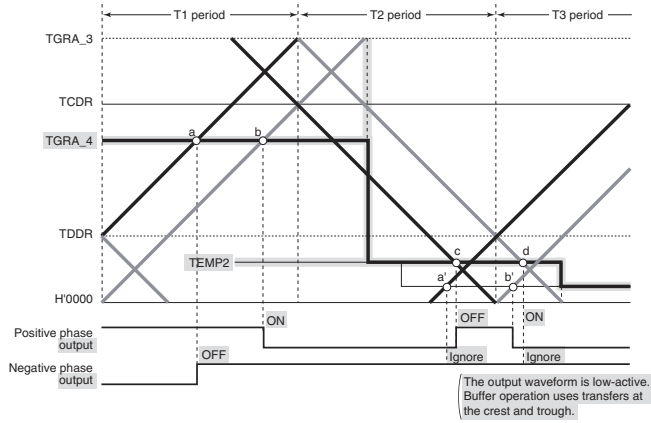
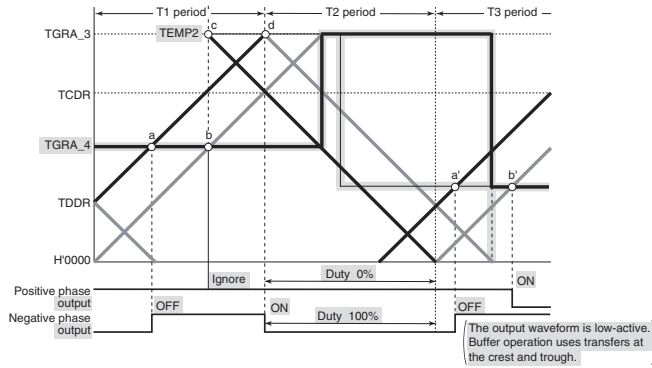
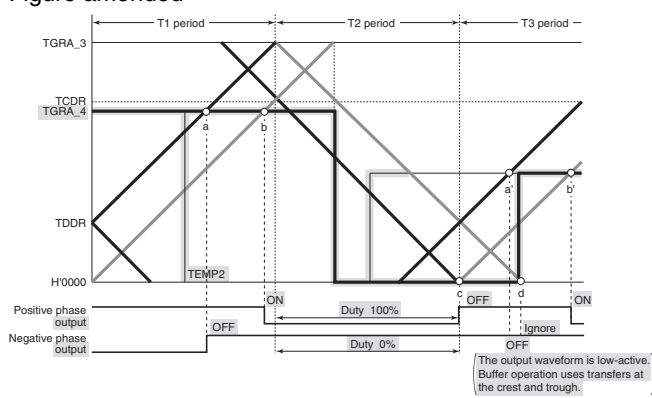
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (b) Register Operation	560, 561	Description amended In complementary PWM mode, nine registers, comprising compare registers, buffer registers, and temporary registers, are used to control the PWM duty. Figure 11.40 shows an example of complementary PWM mode operation. The registers which are constantly compared with the counters to perform PWM output are TGRB_3, TGRA_4, and TGRB_4. When one of these registers matches the counter, the level set in the corresponding timer output control register (TOCR1 or TOCR2) is output on the PWM pin. The buffer registers for these compare registers are TGRD_3, TGRC_4, and TGRD_4. Between a buffer register and compare register there is a temporary register. The temporary registers cannot be accessed by the CPU. Data in a compare register is changed by writing the new data to the corresponding buffer register. The buffer registers can be read or written at any time. When overwriting the data in the buffer registers, always write to TGRD_4 last to enable data transfer from the buffer registers to the temporary registers. At this time, transfer is also enabled from the timer cycle register buffer registers (TGRA_3 and TCBR) to the temporary registers. All five temporary registers can be used simultaneously for transfers. When transfer is enabled during the Ta interval, data written to the buffer register is transferred immediately to the temporary register. Transfer to the temporary register does not take place in the Tb1 or Tb2 interval. Data for which transfer is enabled during either of these intervals is transferred to the temporary register after the interval ends. The value transferred to a temporary register is transferred to the compare register either when the Tb1 interval ends (when TGRA_3 is matched if TCNTS is counting up) or when the Tb2 interval ends (when H'0000 is matched if TCNTS is counting down). The timing for transfer from the temporary register to the compare register can be selected with bits MD[3:0] in the timer mode register (TMDR). Figure 11.40 shows an example in which the mode is selected in which the change is made in the trough.

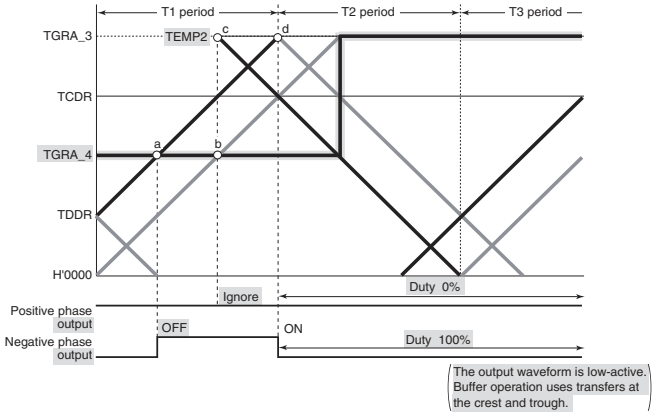
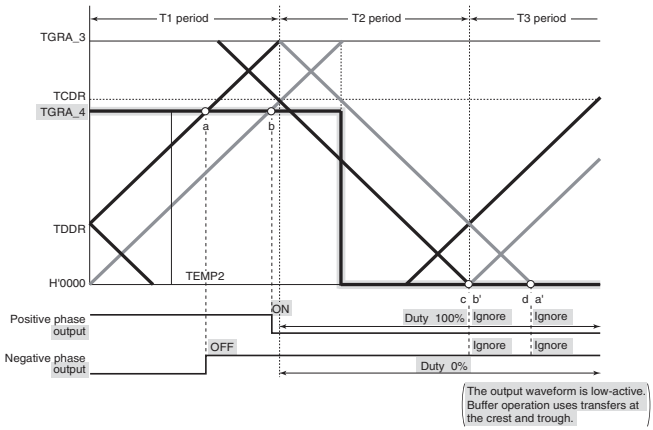
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (b) Register Operation Figure 11.40 Example of Complementary PWM Mode Operation	562	Figure amended 
(c) Initialization	563	Description amended In complementary PWM mode, there are nine 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).
Table 11.56 Registers and Counters Requiring Initialization		Table and note amended **Register/Counter** — **Set Value** TOCR1, TOCR2 — PWM output level setting Note: The TGRC_3 set value must be the sum of 1/2 the PWM cycle set in TCBR and dead time Td set in TDDR. When dead time generation is disabled by TDER, TGRC_3 must be set to 1/2 the PWM cycle + 1.

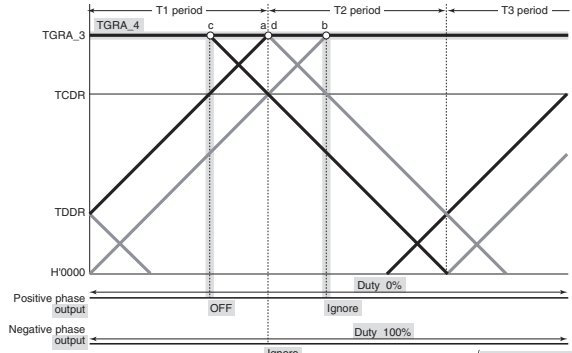
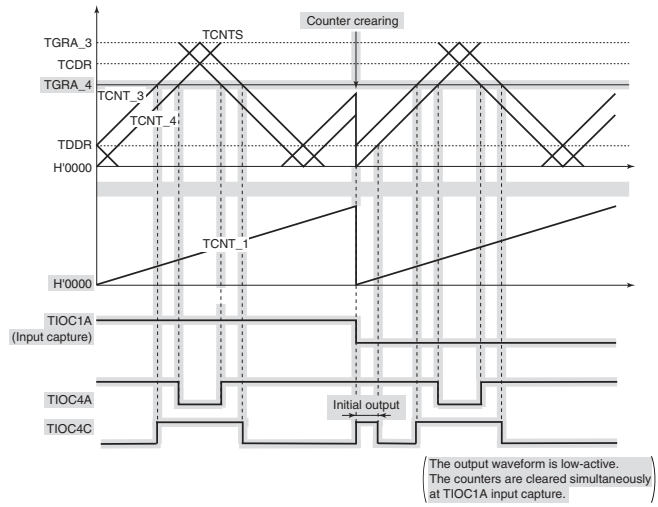
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (f) Dead Time Suppressing Figure 11.41 Example of Operation without Dead Time	565	Figure amended 
(g) PWM Cycle Setting	566	Description amended With dead time: TGRA_3 set value = TCDR set value + TDDR set value TCDR set value > Double the TDDR set value + 2 Without dead time: TGRA_3 set value = TCDR set value + 1 TCDR set value > 4 The TGRA_3 and TCDR settings are made by setting the values in buffer registers TGRC_3 and TCBR. When a write is performed to TGRD_4 and transfer is enabled, the values set in TGRC_3 and TCBR are transferred simultaneously to TGRA_3 and TCDR in accordance with the transfer timing selected by bits MD[3:0] in the timer mode register.
Figure 11.42 Example of PWM Cycle Updating	567	Figure replaced

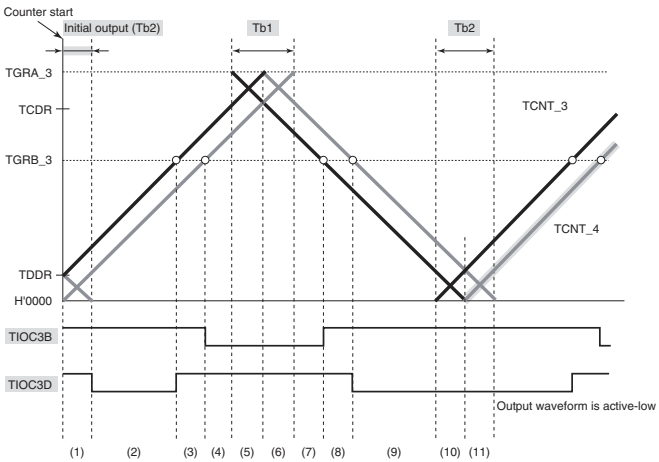
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (h) Register Data Updating	568	Description amended In complementary PWM mode, the buffer register is used to update the data in a compare register and timer cycle register. The update data can be written to the buffer register at any time. There are five PWM duty and PWM cycle registers that have buffer registers and can be updated during operation. There is a temporary register between each of these registers and its buffer register. When subcounter TCNTS is not counting, the temporary register value is also rewritten in the Ta interval when buffer register data is updated. Transfer is not performed from buffer registers to temporary registers in the Tb interval when TCNTS is counting; in this case, the value written to the buffer register is transferred after TCNTS halts.
Figure 11.43 Example of Data Update in Complementary PWM Mode	569	Figure amended 

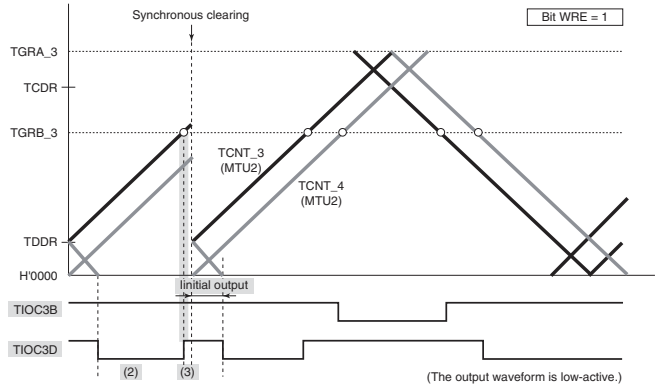
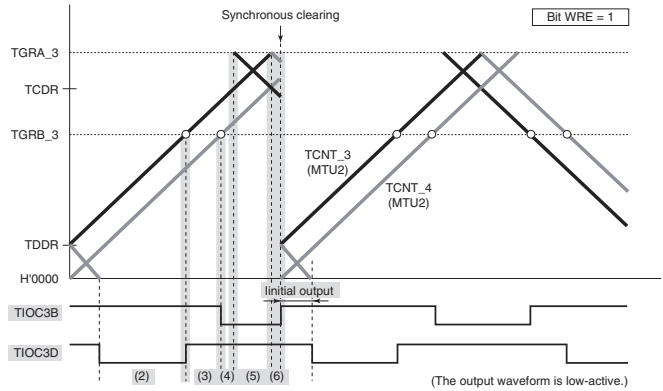
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (j) Complementary PWM Mode PWM Output Generation Method	572	Description amended A PWM waveform is generated by output of the output level selected in the timer output control register in the event of a compare-match between a counter and compare register. While TCNTS is counting, compare register and temporary register values are simultaneously compared to create consecutive PWM pulses from 0 to 100%. ... If compare-match **c** occurs first following compare-match **a**, as shown in figure 11.47, compare-match **b** is ignored, and the negative phase is turned on by compare-match **d**. ...
Figure 11.46 Example of Complementary PWM Mode Waveform Output (1)	573	Figure amended 
Figure 11.47 Example of Complementary PWM Mode Waveform Output (2)		Figure amended 

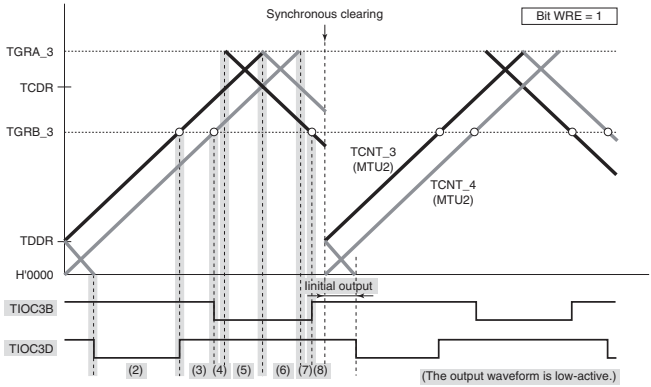
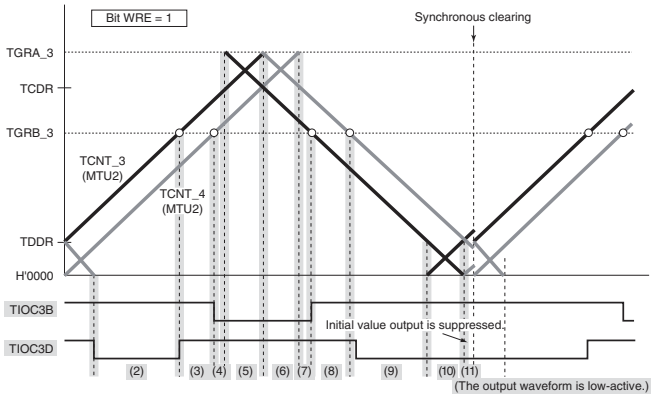
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (j) Complementary PWM Mode PWM Output Generation Method Figure 11.48 Example of Complementary PWM Mode Waveform Output (3)	574	Figure amended 
(k) Complementary PWM Mode 0% and 100% Duty Output Figure 11.49 Example of Complementary PWM Mode 0% and 100% Waveform Output (1)	575	Figure amended 
Figure 11.50 Example of Complementary PWM Mode 0% and 100% Waveform Output (2)		Figure amended 

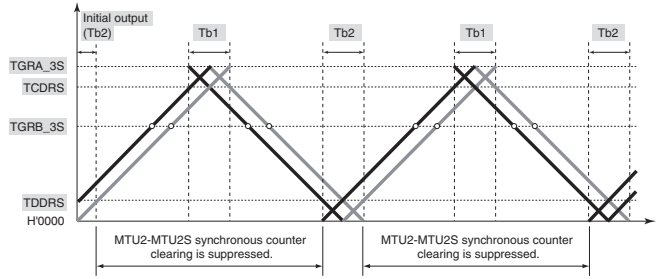
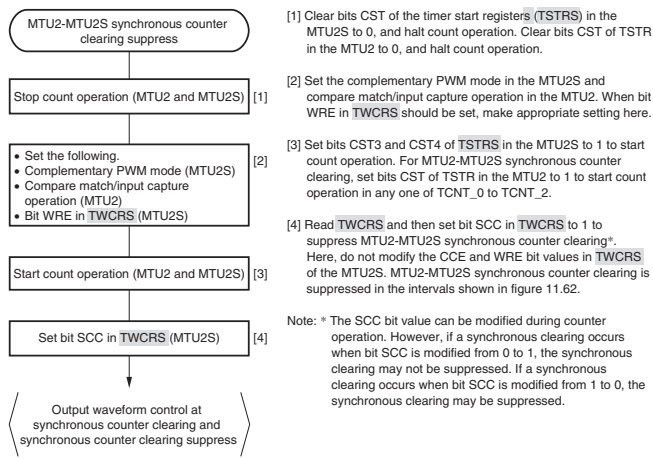
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (k) Complementary PWM Mode 0% and 100% Duty Output Figure 11.51 Example of Complementary PWM Mode 0% and 100% Waveform Output (3)	576	Figure amended 
Figure 11.52 Example of Complementary PWM Mode 0% and 100% Waveform Output (4)		Figure amended 

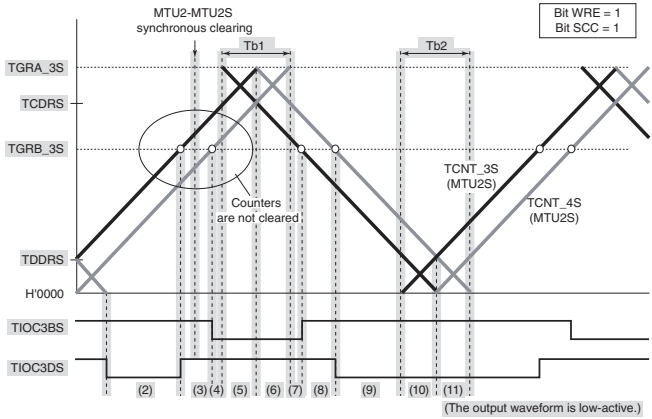
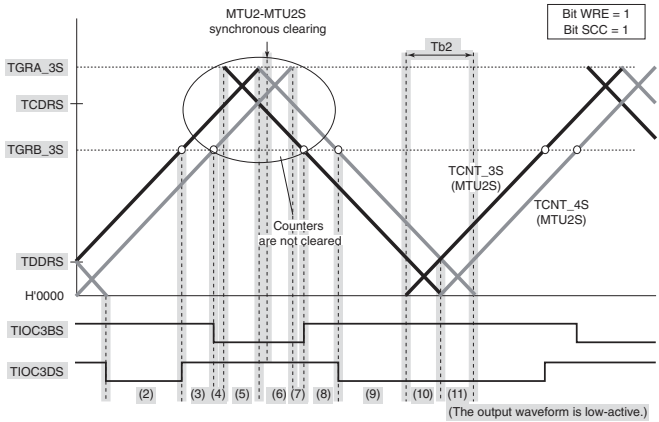
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (k) Complementary PWM Mode 0% and 100% Duty Output Figure 11.53 Example of Complementary PWM Mode 0% and 100% Waveform Output (5)	577	Figure amended 
(m) Counter Clearing by Another Channel Figure 11.55 Counter Clearing Synchronized with Another Channel	579	Figure amended 

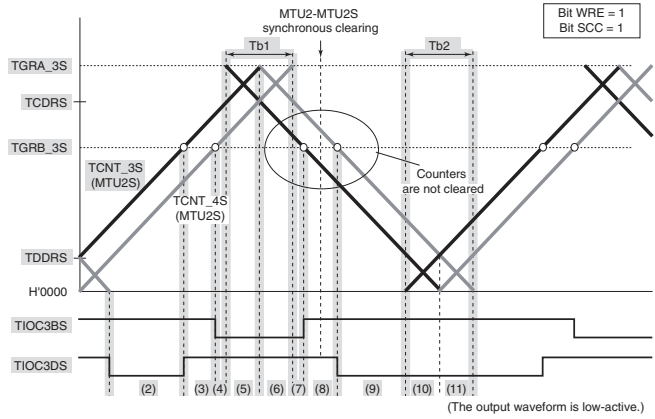
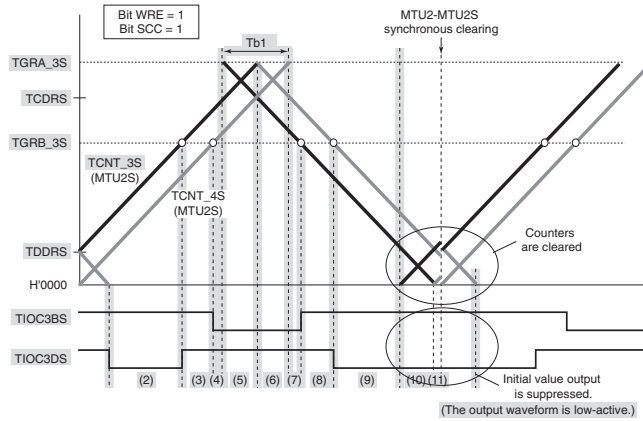
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (n) Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode	580	Description amended Setting the WRE bit in TWCR to 1 suppresses initial output when synchronous counter clearing occurs in the Tb2 interval 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 Tb2 interval as indicated by (10) or (11) in figure 11.56. When synchronous clearing occurs outside that interval, the initial value specified by the OLS bits in TOCR is output. Even in the Tb2 interval, if synchronous clearing occurs in the initial value output period (indicated by (1) in figure 11.56) immediately after the counters start operation, initial value output is not suppressed.
Figure 11.56 Timing for Synchronous Counter Clearing	581	Figure amended 
• Examples of Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode	582	Description amended In the MTU2S, these examples are equivalent to the cases when the MTU2S operates 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 TWCRS.

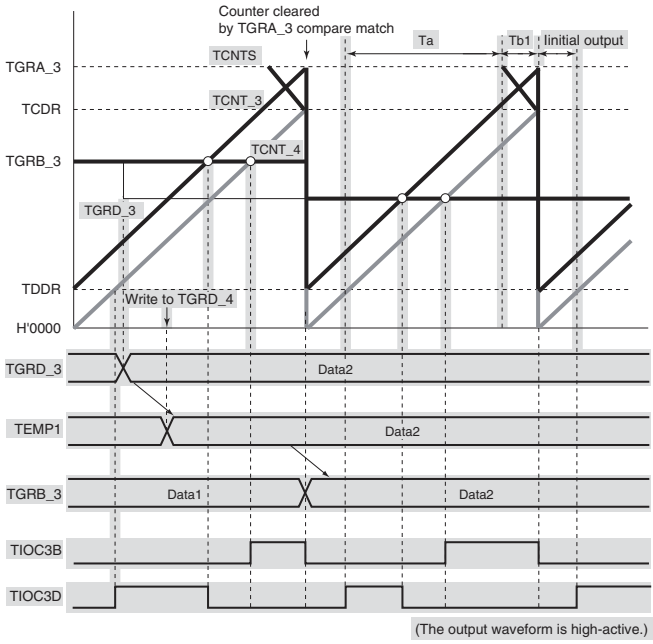
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (n) Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode Figure 11.58 Example of Synchronous Clearing in Dead Time during Up-Counting (Timing (3) in Figure 11.56; Bit WRE of TWCR in MTU2 Is 1)	583	Figure amended 
Figure 11.59 Example of Synchronous Clearing in Interval Tb1 (Timing (6) in Figure 11.56; Bit WRE of TWCR in MTU2 Is 1)		Figure amended 

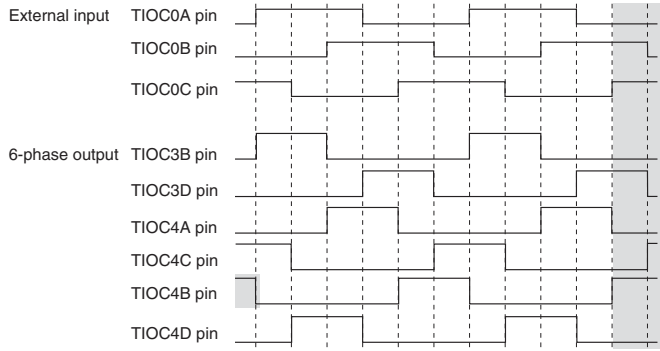
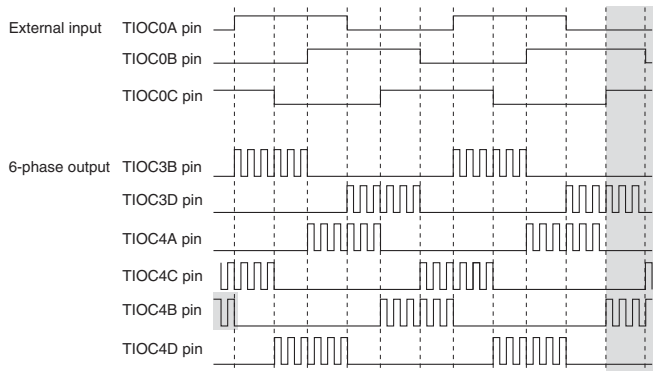
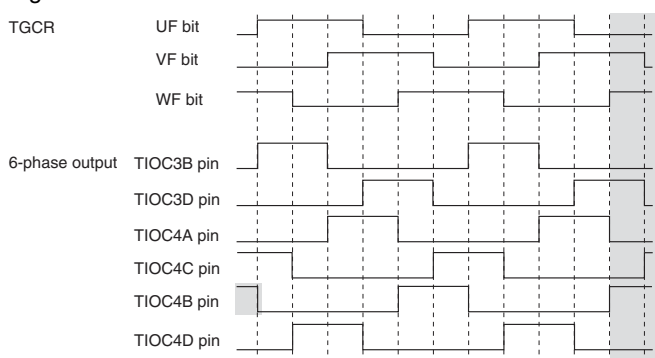
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (n) Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode Figure 11.60 Example of Synchronous Clearing in Dead Time during Down-Counting (Timing (8) in Figure 11.56; Bit WRE of TWCR Is 1)	584	Figure amended 
Figure 11.61 Example of Synchronous Clearing in Interval Tb2 (Timing (11) in Figure 11.56; Bit WRE of TWCR Is 1)		Figure amended 
(o) Suppressing MTU2–MTU2S Synchronous Counter Clearing	585	Description amended In the MTU2S, setting the SCC bit in TWCRS to 1 suppresses synchronous counter clearing caused by the MTU2.

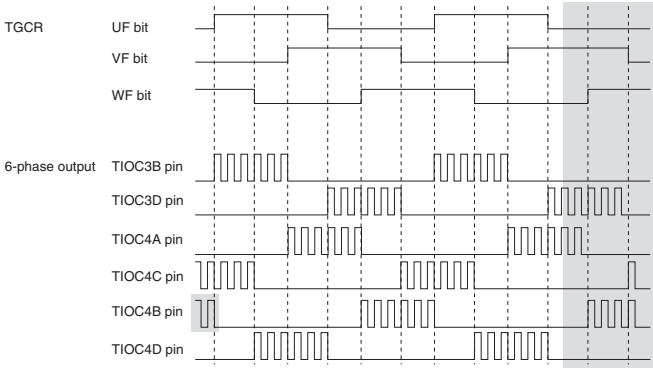
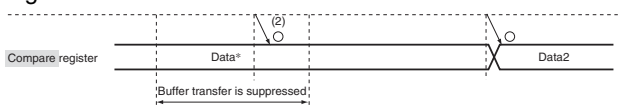
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (o) Suppressing MTU2–MTU2S Synchronous Counter Clearing Figure 11.62 MTU2–MTU2S Synchronous Clearing-Suppressed Interval Specified by SCC Bit in TWCRS	585	Figure and title amended 
Figure 11.63 Example of Procedure for Suppressing MTU2–MTU2S Synchronous Counter Clearing	586	Figure amended 
• Examples of Suppression of MTU2–MTU2S Synchronous Counter Clearing		Description amended Figures 11.64 to 11.67 show examples of operation in which the MTU2S operates in complementary PWM mode and MTU2–MTU2S synchronous counter clearing is suppressed by setting the SCC bit in TWCRS in the MTU2S to 1. In the examples shown in figures 11.64 to 11.67, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in figure 11.56, respectively. In these examples, the WRE bit in TWCRS of the MTU2S is set to 1.

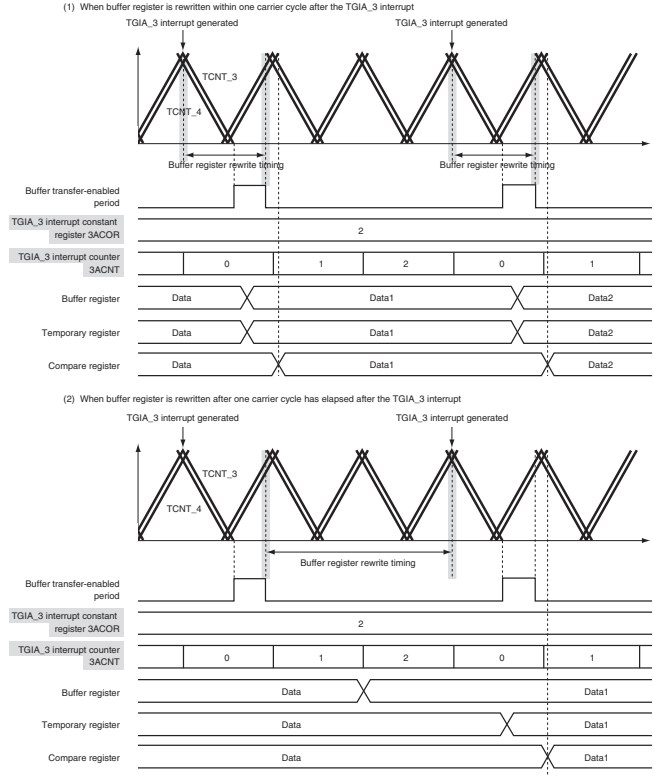
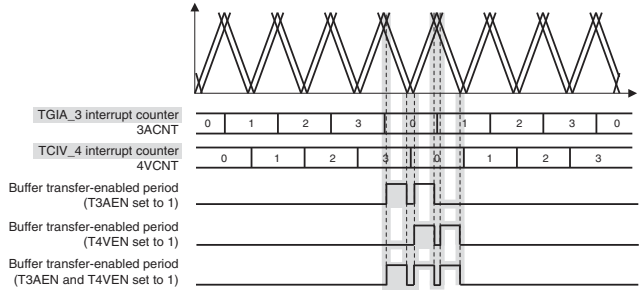
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (o) Suppressing MTU2–MTU2S Synchronous Counter Clearing Figure 11.64 Example of Synchronous Clearing in Dead Time during Up-Counting (Timing (3) in Figure 11.56; Bit WRE Is 1 and Bit SCC Is 1 in TWCRS of MTU2S)	587	Figure and title amended 
Figure 11.65 Example of Synchronous Clearing in Interval Tb1 (Timing (6) in Figure 11.56; Bit WRE Is 1 and Bit SCC Is 1 in TWCRS of MTU2S)	588	Figure and title amended 

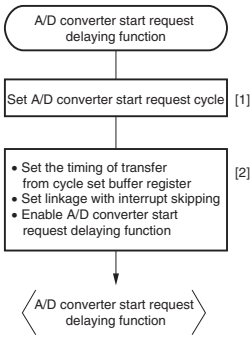
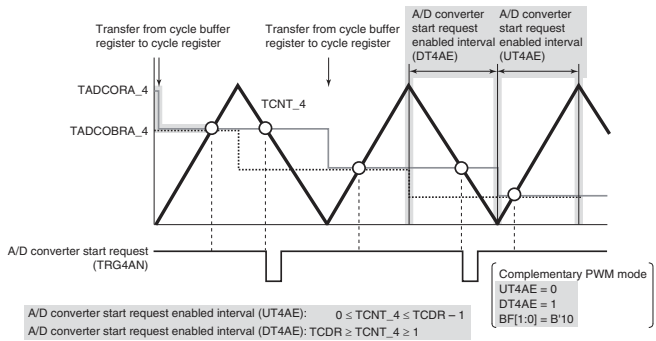
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (o) Suppressing MTU2–MTU2S Synchronous Counter Clearing Figure 11.66 Example of Synchronous Clearing in Dead Time during Down-Counting (Timing (8) in Figure 11.56; Bit WRE Is 1 and Bit SCC Is 1 in TWCRS of MTU2S)	589	Figure and title amended 
Figure 11.67 Example of Synchronous Clearing in Interval Tb2 (Timing (11) in Figure 11.56; Bit WRE Is 1 and Bit SCC Is 1 in TWCRS of MTU2S)	590	Figure and title amended 

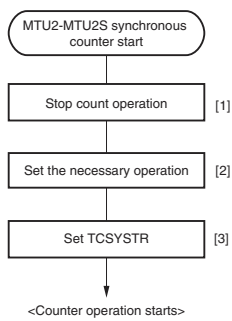
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (p) Counter Clearing by TGRA_3 Compare Match Figure 11.68 Example of Counter Clearing Operation by TGRA_3 Compare Match	591	Figure and title amended 
(q) AC Synchronous Motor (Brushless DC Motor) Drive Waveform Output	592	Description amended To perform output phase switching for a 3-phase brushless DC motor by means of external signals detected with a Hall element, etc., clear the FB bit in TGCR to 0 and input the external signals indicating the polarity position to channel 0 timer input pins TIOC0A, TIOC0B, and TIOC0C (clear to 0 the corresponding IOR bits in the PFC's I/O register). The output is switched on and off automatically according to the signals input on pins TIOC0A, TIOC0B, and TIOC0C. When the FB bit is set to 1, output is switched on and off according to the settings of the UF, VF, and WF bits in TGCR (table 11.39). ... The 6-phase output active level (on output level) can be set with the OLSN and OLSP bits in the timer output control register (TOCR1, TOCR2) regardless of the setting of the N and P bits.

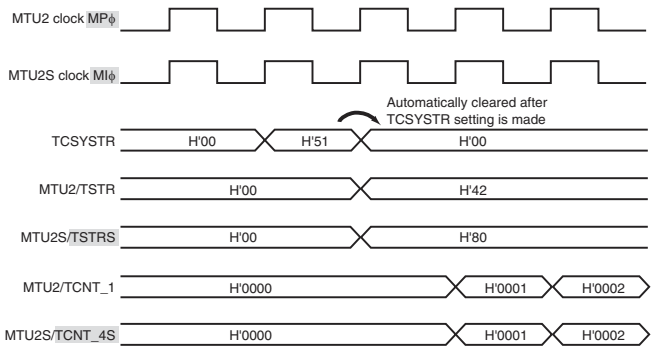
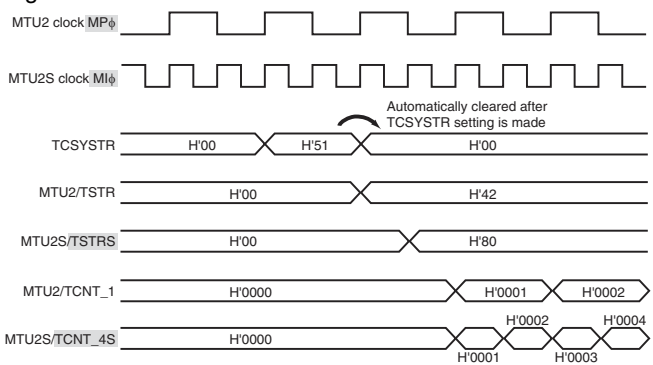
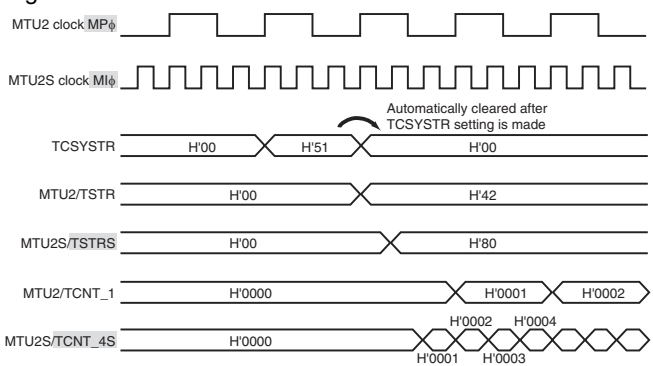
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (q) AC Synchronous Motor (Brushless DC Motor) Drive Waveform Output Figure 11.69 Example of Output Phase Switching by External Input (1)	592	Figure amended  **When BDC = 1, N = 0, P = 0, FB = 0, output active level = high**
Figure 11.70 Example of Output Phase Switching by External Input (2)	593	Figure amended  **When BDC = 1, N = 1, P = 1, FB = 0, output active level = high**
Figure 11.71 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (1)		Figure amended  **When BDC = 1, N = 0, P = 0, FB = 1, output active level = high**

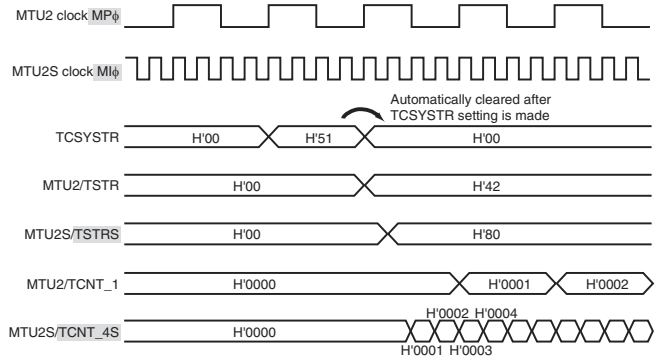
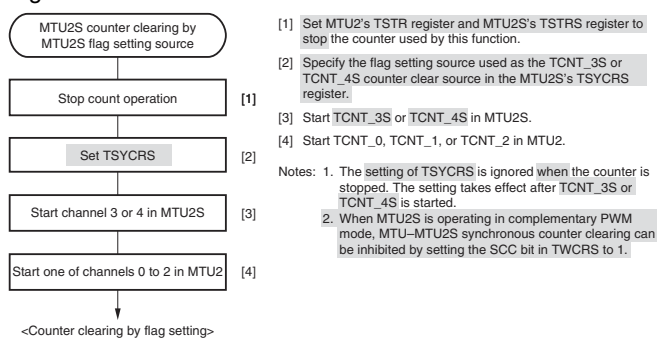
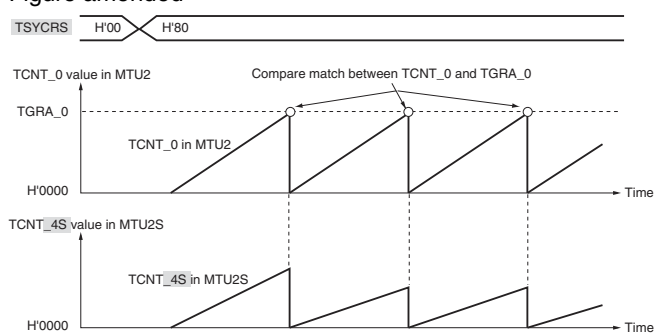
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (2) Outline of Complementary PWM Mode Operation (q) AC Synchronous Motor (Brushless DC Motor) Drive Waveform Output Figure 11.72 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (2)	594	Figure amended  **When BDC = 1, N = 1, P = 1, FB = 1, output active level = high**
(r) A/D Converter Start Request Setting		Description amended In complementary PWM mode, an A/D converter start request can be issued using a TGRA_3 compare-match, or TCNT_4 underflow (trough) . ... For information on the A/D converter start request delaying function, see 11.4.9, A/D Converter Start Request Delaying Function.
(3) Interrupt Skipping in Complementary PWM Mode (c) Buffer Transfer Control Linked with Interrupt Skipping	597	Description amended Figure 11.77 shows an example of operation when buffer transfer is linked with interrupt skipping (BTE[1:0] = B'10). While this setting is valid, data is not transferred from the buffer register outside the buffer transfer-enabled period. Depending on the timing of interrupt generation and writing to the buffer register, the timing of transfer from the buffer register to the temporary register and from the temporary register to the compare register is one of two types.
Figure 11.76 Example of Operation when Buffer Transfer Is Suppressed (BTE[1:0] = B'01)	598	Figure amended  [Legend] (1) No data is transferred from the buffer register to the temporary register in the buffer transfer-disabled period (bits BTE[1:0] in TBTER are set to B'01, respectively). (2) Data is transferred from the temporary register to the compare 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.

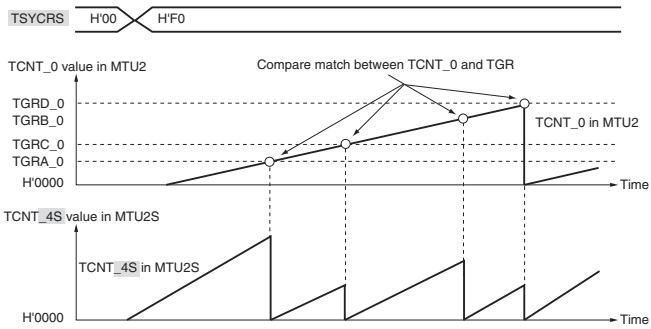
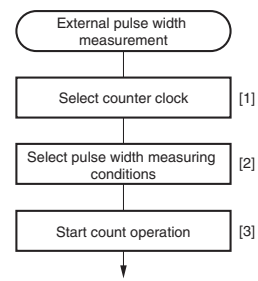
Item	Page	Revision (See Manual for Details)
11.4.8 Complementary PWM Mode (3) Interrupt Skipping in Complementary PWM Mode (c) Buffer Transfer Control Linked with Interrupt Skipping Figure 11.77 Example of Operation when Buffer Transfer Is Linked with Interrupt Skipping (BTE[1:0] = B'10)	599	Figure amended  Note: Bits MD[3:0] in TMDR_3 are set to 1101, selecting buffer transfer at the crest. The skipping count is set to two. T3AEN is set to 1, and T4VEN is cleared to 0.
Figure 11.78 Relationship between Bits T3AEN and T4VEN in Timer Interrupt Skipping Set Register (TITCR) and Buffer Transfer-Enabled Period	600	Figure amended  Note: Bits MD[3:0] in TMDR_3 are set to 1111, selecting buffer transfer at the crest and trough. The skipping count is set to three. T3AEN and T4VEN are set to 1.

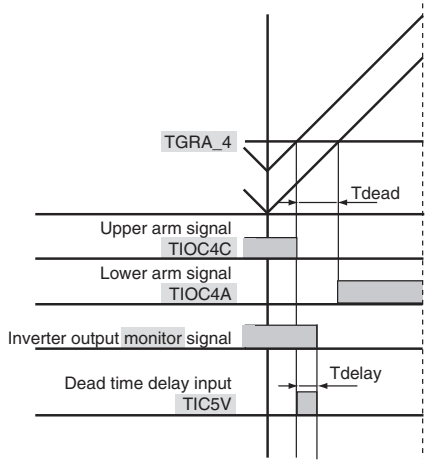
Item	Page	Revision (See Manual for Details)
11.4.9 A/D Converter Start Request Delaying Function (a) Example of Procedure for Specifying A/D Converter Start Request Delaying Function Figure 11.79 Example of Procedure for Specifying A/D Converter Start Request Delaying Function	602	Figure amended  [1] Set the cycle in the timer A/D converter start request cycle buffer register (TADCOBRA_4 or TADCOBRB_4) and timer A/D converter start request cycle register (TADCORA_4 or TADCORB_4). (The same initial value must be specified in the cycle buffer register and cycle register.) [2] Use bits BF[1:0] in the timer A/D converter start request control register (TADCR) to specify the timing of transfer from the timer A/D converter start request cycle buffer register to A/D converter start request cycle register. • Specify whether to link with interrupt skipping through bits ITA3AE, ITA4VE, ITB3AE, and ITB4VE. • Use bits TU4AE, DT4AE, UT4BE, and DT4BE to enable A/D conversion start requests (TRG4AN or TRG4BN). Notes: 1. Perform TADCR setting while TCNT_4 is stopped. 2. Set BF1 bit to 0 when complementary PWM mode is not selected. 3. Set ITA3AE, ITA4VE, ITB3AE, ITB4VE, DT4AE, or DT4BE bits to 0 when complementary PWM mode is not selected. 4. Clear the ITA3AE, ITA4VE, ITB3AE, or ITB4VE bit to 0 to disable interrupt skipping.
(b) Basic Operation Example of A/D Converter Start Request Delaying Function Figure 11.80 Basic Example of A/D Converter Start Request Signal (TRG4AN) Operation	603	Figure amended 
(c) A/D Converter Start Request Enabled Interval		Section added
(d) Buffer Transfer	604	Description added When using buffer transfer in complementary PWM mode, exercise care regarding the buffer transfer timing. For details, see 11.7.24, Notes on Using the A/D Converter Start Request Delaying Function in Complementary PWM Mode. Also, clear the BF1 bit to 0 when not in complementary PWM mode.

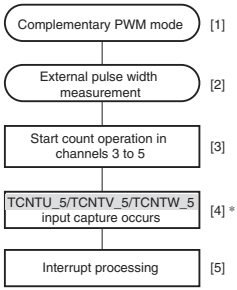
Item	Page	Revision (See Manual for Details)
11.4.9 A/D Converter Start Request Delaying Function (e) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping	604	Description and note added In complementary PWM mode, A/D converter start requests (TRG4AN and TRG4BN) can be issued in coordination with interrupt skipping by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the timer A/D converter start request control register (TADCR). ... The A/D converter start request delaying function linked to the interrupt skipping function cannot be used when not in complementary PWM mode. In this case, clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in TADCR to 0. Note: This function must be used in combination with interrupt skipping. ... Furthermore, when this function is to be used, set TADCORA_4 and TADCORB_4 to a value between H'0002 and the TCDR setting minus two.
11.4.10 MTU2–MTU2S Synchronous Operation (1) MTU2–MTU2S Synchronous Counter Start (a) Example of MTU2–MTU2S Synchronous Counter Start Setting Procedure Figure 11.83 Example of Synchronous Counter Start Setting Procedure	607	Figure amended  [1] Use TSTR registers in the MTU2 and MTU2S and halt the counters used for synchronous start operation. [2] Specify necessary operation with appropriate registers such as TCR and TMDR. [3] In TCSYSTR in the MTU2, set the bits corresponding to the counters to be started synchronously to 1. The TSTRs are automatically set appropriately and the counters start synchronously. Notes: 1. Even if a bit in TCSYSTR corresponding to an operating counter is cleared to 0, the counter will not stop. To stop the counter, clear the corresponding bits in TSTR and TSTRS to 0 directly. 2. To start channels 3 and 4 in reset-synchronized PWM mode or complementary PWM mode, make appropriate settings in TCYSTR according to the TSTR setting for the respective mode. For details, refer to section 11.4.7, Reset-Synchronized PWM Mode, and section 11.4.8, Complementary PWM Mode.
(b) Examples of Synchronous Counter Start Operation	608	Description amended In these examples, the count clock settings are MPϕ/1 (MTU2) and MIϕ/1 (MTU2S).

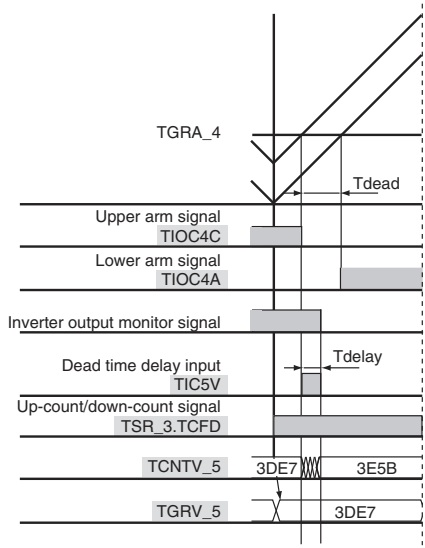
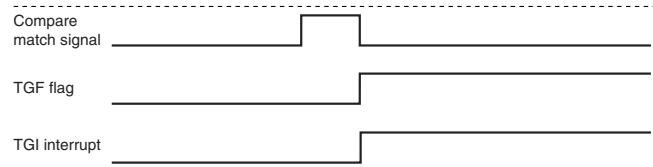
Item	Page	Revision (See Manual for Details)
11.4.10 MTU2–MTU2S Synchronous Operation (1) MTU2–MTU2S Synchronous Counter Start (b) Examples of Synchronous Counter Start Operation Figure 11.84 (1) Example of Synchronous Counter Start Operation (MTU2-to-MTU2S Clock Frequency Ratio = 1:1)	608	Figure amended 
Figure 11.84 (2) Example of Synchronous Counter Start Operation (MTU2-to-MTU2S Clock Frequency Ratio = 1:2)	609	Figure amended 
Figure 11.84 (3) Example of Synchronous Counter Start Operation (MTU2-to-MTU2S Clock Frequency Ratio = 1:3)		Figure amended 

Item	Page	Revision (See Manual for Details)
11.4.10 MTU2–MTU2S Synchronous Operation (1) MTU2–MTU2S Synchronous Counter Start (b) Examples of Synchronous Counter Start Operation Figure 11.84 (4) Example of Synchronous Counter Start Operation (MTU2-to-MTU2S Clock Frequency Ratio = 1:4)	610	Figure amended 
(2) MTU2S Counter Clearing Caused by MTU2 Flag Setting Source (MTU2–MTU2S Synchronous Counter Clearing) (a) Example of Procedure for Specifying MTU2S Counter Clearing by MTU2 Flag Setting Source Figure 11.85 Example of Procedure for Specifying MTU2S Counter Clearing by MTU2 Flag Setting Source	611	Figure amended 
(b) Examples of MTU2S Counter Clearing Caused by MTU2 Flag Setting Source Figure 11.86 (1) Example of MTU2S Counter Clearing Caused by MTU2 Flag Setting Source (1)	612	Figure amended 

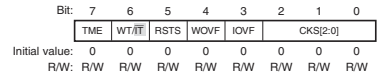
Item	Page	Revision (See Manual for Details)
11.4.10 MTU2–MTU2S Synchronous Operation (2) MTU2S Counter Clearing Caused by MTU2 Flag Setting Source (MTU2–MTU2S Synchronous Counter Clearing) (b) Examples of MTU2S Counter Clearing Caused by MTU2 Flag Setting Source Figure 11.86 (2) Example of MTU2S Counter Clearing Caused by MTU2 Flag Setting Source (2)	612	Figure amended 
11.4.11 External Pulse Width Measurement (1) Example of External Pulse Width Measurement Setting Procedure	613	Title deleted Description amended The pulse widths of up to three external input lines can be measured in channel 5. If pulse width measurement is specified by the setting of the IOC[4:0] bit field in TIORU_5/TIORV_5/TIORW_5, the pulse width of the signal input on TIC5U/TIC5V/TIC5W is measured. TCNTU_5/TCNTV_5/TCNTW_5 is incremented when the input level is that specified by the IOC[4:0] bit field.
Figure 11.87 Example of External Pulse Width Measurement Setting Procedure		Figure amended  [1] Select the counter clock by means of the TPSC[1:0] bit field in TCRU_5/TCRV_5/TCRW_5. [2] Specify high-level or low-level as the pulse width measurement condition in TIORU_5/TIORV_5/TIORW_5. [3] Set the CSTU_5/CSTV_5/CSTW_5 bit in TSTR_5 to 1 to start counter operation. Notes: If channels 3 and 4 are not operating in complementary PWM mode, take note of 1 to 3 below: 1. Clear the CMPCLR5U/CMPCLR5V/CMPCLR5W bit in TCNTCMPCLR to 0. 2. Clear the TGIE5U/TGIE5V/TGIE5W bit in TIER_5 to 0. 3. The value of TCNTU_5/TCNTV_5/TCNTW_5 is not captured to TGRU_5/TGRV_5/TGRW_5.

Item	Page	Revision (See Manual for Details)
11.4.11 External Pulse Width Measurement (2) Example of External Pulse Width Measurement	613	Title deleted
11.4.12 Dead Time Compensation	614	Description amended A motor control circuit (figure 11.89) is provided that feeds back to channel 5 the dead time delay (delay between the complementary PWM output and the inverter output). Dead time compensation can be applied to the PWM output waveform by using the external pulse width measurement function on channel 5 to measure the dead time delay and adjusting the PWM duty accordingly (figure 11.90). Figure 11.91 shows an example of the procedure for setting the dead time compensation when using channel 5. For details the operation of channel 5, see 11.4.13, TCNTU_5/TCNTV_5/TCNTW_5 Capture at Crest and/or Trough in Complementary PWM Operation.
Figure 11.90 Delay in Dead Time in Complementary PWM Operation	615	Figure amended 

Item	Page	Revision (See Manual for Details)
11.4.12 Dead Time Compensation Figure 11.91 Example of Dead Time Compensation Setting Procedure	615	Figure amended  [1] Place channels 3 and 4 in complementary PWM mode. For details, refer to section 11.4.8, Complementary PWM Mode. [2] Specify the external pulse width measurement function for the target TIOR in channel 5. For details, refer to section 11.4.11, External Pulse Width Measurement. [3] Set bits CST3 and CST4 in TSTR and bit CSTU_5/CSTV_5/CSTW_5 in TSTR_5 to 1 to start count operation. [4] When the capture condition specified in TIORU_5/TIORV_5/TIORW_5 is satisfied, the TCNTU_5/TCNTV_5/TCNTW_5 value is captured in TGRU_5/TGRV_5/TGRW_5. [5] For U-phase dead time compensation, when an interrupt is generated at the crest (TGIA_3) or trough (TCIV_4) in complementary PWM mode, read the TGRU_5 value, calculate the difference in time in TGRB_3, and write the corrected value to TGRD_3 in the interrupt processing. For the V phase and W phase, read the TGRV_5 and TGRW_5 values and write the corrected values to TGRC_4 and TGRD_4, respectively, in the same way as for U-phase compensation. The TCNTU_5/TCNTV_5/TCNTW_5 value should be cleared through the TCNTCMPCLR setting or by software. Notes: The PFC settings must be completed in advance. * As an interrupt flag is set under the capture condition specified in TIOR, do not enable interrupt requests in TIER_5.
11.4.13 TCNTU_5/TCNTV_5/TCNTW_5 Capture at Crest and/or Trough in Complementary PWM Operation	616	Title and Description amended The external pulse width measurement function on channel 5 captures the value of TCNTU_5/TCNTV_5/TCNTW_5 at the crest, the trough, or both the crest and trough, and stores it in TGRU_5/TGRV_5/TGRW_5, during complementary PWM operation. The timing for capturing to TGRU_5/TGRV_5/TGRW_5 is selected by TIORU_5/TIORV_5/TIORW_5. Also, setting to 1 the CMPCLRU/CMPCLRV/CMPCLRW bit in TCNTCMPCLR causes TCNTU_5/TCNTV_5/TCNTW_5 to be cleared when the capture takes place. Figure 11.92 is an operating example in which TCNTU_5/TCNTV_5/TCNTW_5 is used as a free-running counter without being cleared, and the TCNTU_5/TCNTV_5/TCNTW_5 value is captured to TGRU_5/TGRV_5/TGRW_5 at the crest and trough during complementary PWM operation.

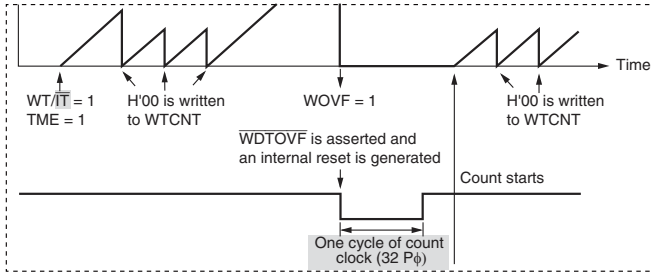
Item	Page	Revision (See Manual for Details)
11.4.13 TCNTU_5/TCNTV_5/TCNTW_5 Capture at Crest and/or Trough in Complementary PWM Operation Figure 11.92 TCNTU_5/TCNTV_5/TCNTW_5 Capturing at Crest and/or Trough in Complementary PWM Operation	616	Figure and title amended 
11.6.2 Interrupt Signal Timing Figure 11.110 TGI Interrupt Timing (Compare Match) (Channel 5)	629	Figure amended  Note: The compare match is generated even though TCNT is stopped.
11.7.24 Notes on Using the A/D Converter Start Request Delaying Function in Complementary PWM Mode	651	Section added

Item	Page	Revision (See Manual for Details)
14.3.2 Watchdog Timer Control/Status Register (WTCSR)	731	Figure and bit table amended



Bit	Bit Name	Initial Value	R/W	Description
7	TME	0	R/W	Timer Enable Starts and stops timer operation. Clear this bit to 0 when using the WDT to revoke software standby mode. 0: Timer disabled: Count-up stops and WTCNT value is retained 1: Timer enabled
6	WT/$\overline{IT}$	0	R/W	Timer Mode Select Selects whether to use the WDT as a watchdog timer or an interval timer. 0: Interval timer mode 1: Watchdog timer mode Note: When the WTCNT overflows in watchdog timer mode, the $\overline{WDTOVF}$ signal is output externally. If this bit is modified when the WDT is running, the up-count may not be performed correctly.

14.3.3 Notes on Register Access	733	Description amended The watchdog timer counter (WTCNT) and watchdog timer control/status register (WTCSR) are more difficult to write to than other registers. The procedures for reading or writing to these registers are given below.
(1) Writing to WTCNT and WTCSR Figure 14.2 Writing to WTCNT and WTCSR		Figure moved
(2) Reading from WTCNT and WTCSR		Description replaced
14.4.1 Revoking Software Standbys	734	Description amended The WDT can be used to revoke software standby mode with an NMI interrupt or external interrupt (IRQ). The procedure is described below. (The WDT does not run when resets are used for revoking, so keep the $\overline{RES}$ or $\overline{MRES}$ pin low until the clock stabilizes.) … 3. After setting the STBY bit to 1 in standby control register 1 (STBCR1: see section 26, Power-Down Modes) and the STBYMD bit to 1 in standby control register 6 (STBCR6: see section 26, Power-Down Modes), execute a SLEEP instruction to transition to software standby mode and stop the clock.

Item	Page	Revision (See Manual for Details)
14.4.2 Using Watchdog Timer Mode	734	Description amended 1. Set the WT/$\overline{IT}$ bit in WTCSR to 1, set the reset type in the RSTS bit, set the type of count clock in the CKS2 to CKS0 bits, and set the initial value of the counter in the WTCNT counter.
Figure 14.3 Operation in Watchdog Timer Mode (When WTCNT Count Clock Is Specified to Pϕ/32 by CKS2 to CKS0)	735	Figure amended 
14.4.3 Using Interval Timer Mode	736	Description amended 1. Clear the WT/$\overline{IT}$ bit in WTCSR to 0, set the type of count clock in the CKS2 to CKS0 bits, and set the initial value of the counter in the WTCNT counter.
Figure 14.4 Operation in Interval Timer Mode		Figure added
14.6 Usage Note	737	Description added Pay attention to the following points when using the WDT in either the interval timer or watchdog timer mode.
14.6.1 WTCNT Setting Value		Description amended If the timer is stopped and WTCNT is set to H'FF in interval timer mode, an overflow does not occur when WTCNT changes from H'FF to H'00 after one cycle of the count clock, but an overflow does occur when WTCNT changes from H'FE to H'FF after 256 cycles of the count clock. If the timer is operating and WTCNT is set to H'FF, an interval timer interrupt is generated immediately. Do not set WTCNT to H'FF in watchdog timer mode. If WTCNT is set to H'FF, a WDT reset is generated immediately, regardless of the clock selected by bits CKS[2:0]. In this case, the assertion periods of the $\overline{WDTOVF}$ signal and internal reset signal are shortened.
14.6.2 Timer Variation		Section added

Item | **Page** | **Revision (See Manual for Details)**

14.6.3 System Reset by WDTOVF Signal
14.6.4 Manual Reset in Watchdog Timer Mode
14.6.5 Internal Reset in Watchdog Timer Mode

738 — Section added

15.3 Register Descriptions

Table 15.2 Register Configuration

742 — Table amended

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	Transmit data register_0	SCTDR_0	R/W	H'xx	H'FFFFC006	8
	Serial status register_0	SCSSR_0	R/W	H'84	H'FFFFC008	8
	Receive data register_0	SCRDR_0	R	H'xx	H'FFFFC00A	8
1	Transmit data register_1	SCTDR_1	R/W	H'xx	H'FFFFC086	8
	Serial status register_1	SCSSR_1	R/W	H'84	H'FFFFC088	8
	Receive data register_1	SCRDR_1	R	H'xx	H'FFFFC08A	8
2	Transmit data register_2	SCTDR_2	R/W	H'xx	H'FFFFC106	8
	Serial status register_2	SCSSR_2	R/W	H'84	H'FFFFC108	8
	Receive data register_2	SCRDR_2	R	H'xx	H'FFFFC10A	8

15.3.1 Receive Shift Register (SCRSR)

743 — Description amended

SCRSR receives serial data. The SCI converts serial data input to SCRSR via the RXD pin to parallel data.

15.3.2 Receive Data Register (SCRDR)

Description amended

SCRDR is a read-only register and SCRDR can be read but not written to by the CPU.

15.3.3 Transmit Shift Register (SCTSR)

Description amended

SCTSR transmits serial data. The SCI loads transmit data from the transmit data register (SCTDR) into SCTSR, then transmits the data serially from the TXD pin .

15.3.6 Serial Control Register (SCSCR)

749 — Table amended

Bit	Bit Name	Initial value	R/W	Description
1, 0	CKE[1:0]	00	R/W	Clock Enable Select the SCI clock source and enable or disable clock output from the SCK pin. Depending on the combination of CKE[1:0], the SCK pin can be used for serial clock output or serial clock input. When selecting the clock output in clock synchronous mode, set the C/$\overline{A}$ bit in SCSMR to 1 and then set bits CKE[1:0]. For details on clock source selection, refer to table 15.11. • Asynchronous mode 00: Internal clock, SCK pin used for input pin (The input signal is ignored.) The state of the SCK pin depends on the settings of bits SPB1IO and SPB1DT in SCSPTR.

Item | **Page** | **Revision (See Manual for Details)**

15.3.7 Serial Status Register (SCSSR)

Page: 750, 751

Table amended

Bit	Bit Name	Initial value	R/W	Description
7	TDRE	1	R/(W)*	Transmit Data Register Empty ... • When the DMAC is activated by a TXI interrupt and transmit data is written to SCTDR • When the DTC is activated by a TXI interrupt and transmit data is transferred to SCTDR while the DISEL bit of MRB in the DTC is 0 (except when the transfer counter value of the DTC is H'0000)
6	RDRF	0	R/(W)*	Receive Data Register Full ... • When the DMAC is activated by an RXI interrupt and data is transferred from SCRDR • When the DTC is activated by an RXI interrupt and the received data is transferred from SCRDR while the DISEL bit of MRB in the DTC is 0 (except when the transfer counter value of the DTC is H'0000)

15.3.8 Serial Port Register (SCSPTR)

Page: 756

Description and bit table amended

SCSPTR is an 8-bit register that controls input/output and data for the ports multiplexed with the SCI function pins. It controls break signals during serial transmission and reception when writing of output data to the TXD pin is enabled by the settings of bits SPB0IO and SPB0DT. Also, bits SPB1IO and SPB1DT can be used to write output data to the SCK pin. The EIO bit enables or disables RXI interrupts.

Bit:	7	6	5	4	3	2	1	0
	EIO	-	-	-	SPB1IO	SPB1DT	SPB0IO	SPB0DT
Initial value:	0	0	0	0	0	-	0	1
R/W:	R/W	-	-	-	R/W	W	R/W	W

Table amended

Bit	Bit Name	Initial value	R/W	Description
7	EIO	0	R/W	Error Interrupt Only Enables or disables RXI interrupts. While the EIO bit is set to 1, the SCI does not request an RXI interrupt to the CPU even if the RIE bit is set to 1. 0: While the RIE bit is 1, RXI and ERI interrupts are sent to the INTC. 1: While the RIE bit is 1, only the ERI interrupt is sent to the INTC.
3	SPB1IO	0	R/W	Clock Port Input/Output in Serial Port Controls the SCK pin in combination with the SPB1DT bit, the C/$\overline{A}$ bit in SCSMR, and the CKE[1:0] bits in SCSCR.

Item	Page	Revision (See Manual for Details)
15.3.8 Serial Port Register (SCSPTR)	757	Table amended

Bit	Bit Name	Initial value	R/W	Description
2	SPB1DT	Undefined	W	Clock Port Data in Serial Port Controls the SCK pin in combination with the SPB1IO bit, the C/$\overline{A}$ bit in SCSMR, and the CKE[1:0] bits in SCSCR. Note that the SCK pin function needs to have been selected with the pin function controller (PFC). Also, this bit is write-only. When read its value is undefined.

C/$\overline{A}$	CKE1	CKE0	SPB 1IO	SPB 1DT	SCK pin state
0	0	0	0	×	SCK pin functions as input pin.
0	0	0	1	0	Low-level output
0	0	0	1	1	High-level output
0	0	1	×	×	SCK pin functions as clock output.
0	1	0	×	×	SCK pin functions as clock input.
0	1	1	×	×	SCK pin functions as clock input.
1	0	0	×	×	SCK pin functions as sync clock output.
1	0	1	×	×	SCK pin functions as sync clock output.
1	1	0	×	×	SCK pin functions as sync clock input.
1	1	1	×	×	SCK pin functions as sync clock input.

Note: ×: Don't care

Bit	Bit Name	Initial value	R/W	Description
1	SPB0IO	0	R/W	Serial Port Break Output Together with the SPB0DT bit and the TE bit in SCSCR, controls the TXD pin.

	758	Table amended		

Bit	Bit Name	Initial value	R/W	Description
0	SPB0DT	1	W	Serial Port Break Data ... Also, this bit is write-only. When read its value is undefined.

TE bit setting in SCSCR	SPB0IO bit setting	SPB0DT bit setting	State of TXD pin
1	*	*	Transmit data output

Item	Page	Revision (See Manual for Details)
15.3.10 Bit Rate Register (SCBRR) Table 15.4 Bit Rates and SCBRR Settings in Asynchronous Mode (1)	762	Table amended

Pφ (MHz)

Bit Rate	10			12			14			16			18			20		
(bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
115200	0	2	-9.58	0	2	8.51	0	3	-5.06	0	3	8.51	0	4	-2.34	0	4	8.51
500000	0	0*	-37.5	0	0*	-25.0	0	0*	-12.5	0	0*	0.00	0	0*	12.5	0	0*	25.0

Table 15.4 Bit Rates and SCBRR Settings in Asynchronous Mode (2)	763	Table amended

Pφ (MHz)

Bit Rate	22			24			26			28			30			32		
(bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
115200	0	5	-0.54	0	6	-6.99	0	6	0.76	0	7	-5.06	0	7	1.73	0	8	-3.55
500000	0	0*	37.5	0	1	-25.0	0	1	-18.8	0	1	-12.5	0	1	-6.25	0	1	0.00

Table 15.4 Bit Rates and SCBRR Settings in Asynchronous Mode (3)	764	Table amended

Pφ (MHz)

Bit Rate	34			36			38			40		
(bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
115200	0	8	2.48	0	9	-2.34	0	9	3.08	0	10	-1.36
500000	0	1	6.25	0	1	12.5	0	1	18.8	0	2	-16.7

	768	Description amended …Table 15.7 lists the maximum bit rates in clock-synchronous mode when the baud rate generator is used.
Table 15.6 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (Asynchronous Mode)	768	Table amended

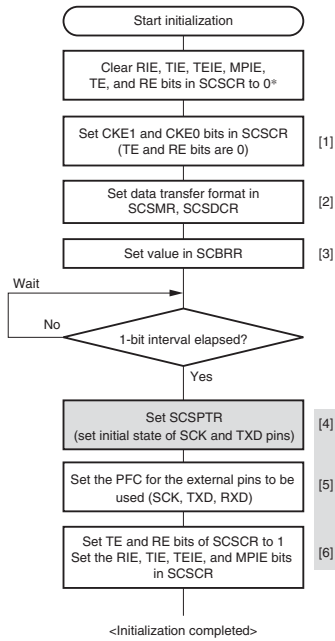
	Discontinuous Transmission/Reception			Continuous Transmission/Reception		
		Settings			Settings	
Pφ (MHz)	Maximum Bit Rate (bits/s)	n	N	Maximum Bit Rate (bits/s)	n	N
10	312500	0	0	156250	0	1
12	375000	0	0	187500	0	1
14	437500	0	0	218750	0	1
16	500000	0	0	250000	0	1
18	562500	0	0	281250	0	1
20	625000	0	0	312500	0	1
22	687500	0	0	343750	0	1
24	750000	0	0	375000	0	1
26	812500	0	0	406250	0	1
28	875000	0	0	437500	0	1
30	937500	0	0	468750	0	1
32	1000000	0	0	500000	0	1
34	1062500	0	0	531250	0	1
36	1125000	0	0	562500	0	1
38	1187500	0	0	593750	0	1
40	1250000	0	0	625000	0	1

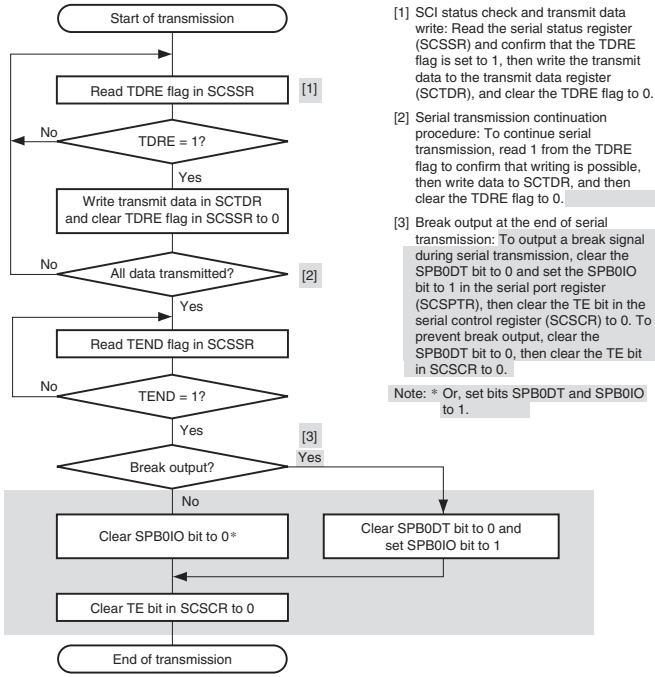
Item	Page	Revision (See Manual for Details)
15.3.10 Bit Rate Register (SCBRR) Table 15.7 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (Clock Synchronous Mode)	769	Table replaced
15.3.10 Bit Rate Register (SCBRR) Table 15.8 Maximum Bit Rates with External Clock Input (Asynchronous Mode)	770	Table amended

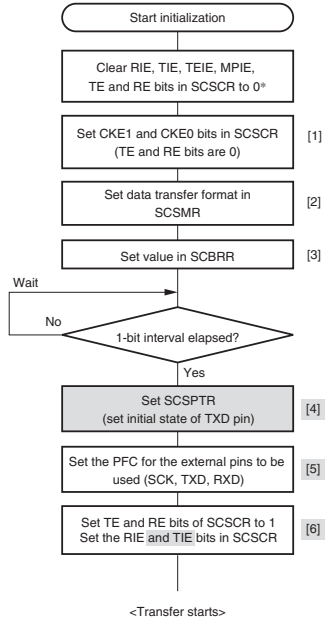
Pφ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bits/s)
10	2.5	156250
12	3.0	187500
14	3.5	218750
16	4.0	250000
18	4.5	281250
20	5.0	312500
22	5.5	343750
24	6.0	375000
26	6.5	406250
28	7.0	437500
30	7.5	468750
32	8.0	500000
34	8.5	531250
36	9.0	562500
38	9.5	593750
40	10.0	625000

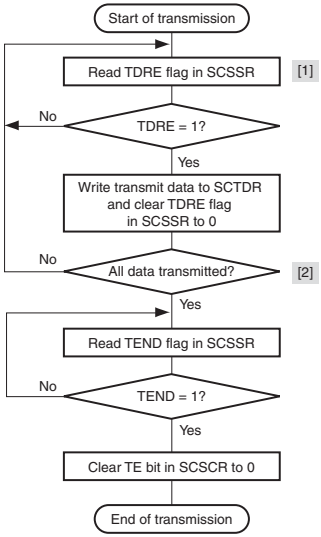
Table 15.9 Maximum Bit Rates with External Clock Input (Clock Synchronous Mode)	771	Table amended

Pφ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bits/s)
10	1.6667	1666666
12	2.0000	2000000
14	2.3333	2333333
16	2.6667	2666666
18	3.0000	3000000
20	3.3333	3333333
22	3.6667	3666666
24	4.0000	4000000
26	4.3333	4333333
28	4.6667	4666666
30	5.0000	5000000
32	5.3333	5333333
34	5.6667	5666666
36	6.0000	6000000
38	6.3333	6333333
40	6.6667	6666666

Item	Page	Revision (See Manual for Details)
15.4.2 Operation in Asynchronous Mode	774	Description amended …One serial character consists of a start bit (low), data (LSB first when LSB-first transfer is selected), parity bit (high or low), and stop bit (high), in that order.
Figure 15.2 Example of Data Format in Asynchronous Communication (8-Bit Data with Parity and Two Stop Bits when LSB-First Transfer is Selected)		Figure title amended
(3) Transmitting and Receiving Data Figure 15.3 Sample Flowchart for SCI Initialization (Asynchronous Mode)	777	Figure and title amended  [1] Set the clock selection in SCSCR. [2] Set the data transfer format in SCSMR and SCSDCR. [3] Write a value corresponding to the bit rate to SCBRR. Not necessary if an external clock is used. [4] Set SCSPTR to specify the initial state of the SCK and TXD pins. If it is not necessary to specify an initial state, clear bits SPB1IO and SPB0IO to 0. [5] Make PFC settings for the external pins to be used. If a clock output setting has been specified, clock output from the SCK pin starts at this point in time. [6] Set the TE bit or RE bit in SCSCR to 1.* Also make settings of the RIE, TIE, TEIE, and MPIE bits. At this time, the TXD, RXD, and SCK pins are ready to be used. The TXD pin is in a mark state during transmitting, and RXD pin is in an idle state for waiting the start bit during receiving.

Item	Page	Revision (See Manual for Details)
15.4.2 Operation in Asynchronous Mode (3) Transmitting and Receiving Data Figure 15.4 Sample Flowchart for Transmitting Serial Data (Asynchronous Mode)	778	Figure and title amended 
	779	Description amended B. Transmit data: 8-bit or 7-bit data is output in LSB-first order (when LSB-first transfer is selected).
Figure 15.5 Example of Transmission in Asynchronous Mode (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)	780	Figure title amended
Figure 15.6 Sample Flowchart for Receiving Serial Data (Asynchronous Mode) (1)	781	Figure and title amended [3] Serial reception continuation procedure: To continue serial reception, clear the RDRF flag to 0 before the stop bit for the current frame is received.
Figure 15.6 Sample Flowchart for Receiving Serial Data (Asynchronous Mode) (2)	782	Figure title amended

Item	Page	Revision (See Manual for Details)
15.4.2 Operation in Asynchronous Mode Figure 15.7 Example of SCI Receive Operation (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)	784	Figure title amended
15.4.3 Clock Synchronous Mode Figure 15.8 Data Format in Clock Synchronous Communication (when LSB-First Transfer is Selected)		
	785	Description amended ... After output of the MSB, the communication line remains in the state of the MSB (when LSB-first transfer is selected).
Figure 15.9 Sample Flowchart for SCI Initialization (Clock Synchronous Mode)	786	Figure and title amended 

Item	Page	Revision (See Manual for Details)
15.4.3 Clock Synchronous Mode Figure 15.10 Sample Flowchart for Transmitting Serial Data (Clock Synchronous Mode)	787	Figure and title amended  [1] SCI status check and transmit data write: Read SCSSR and check that the TDRE flag is set to 1, then write transmit data to SCTDR, and clear the TDRE flag to 0. [2] Serial transmission continuation procedure: To continue serial transmission, read 1 from the TDRE flag to confirm that writing is possible, then write data to SCTDR, and then clear the TDRE flag to 0.
	788	Description amended 2. ...Data is output from the TXD pin in order from the LSB (bit 0) to the MSB (bit 7) (when LSB-first transfer is selected). 3. The SCI checks the TDRE flag at the timing for sending the last bit. If the TDRE flag is 0, the data is transferred from SCTDR to SCTSR and serial transmission of the next frame is started. If the TDRE flag is 1, the TEND flag in SCSSR is set to 1, the last bit is sent, and then the TXD pin holds the states.
Figure 15.11 Example of SCI Transmit Operation (when LSB-First Transfer is Selected)		Figure title amended
Figure 15.12 Sample Flowchart for Receiving Serial Data (Clock Synchronous Mode) (1)	789	Figure and title amended [3] Serial reception continuation procedure: To continue serial reception, read the receive data register (SCRDR) and clear the RDRF flag to 0 before the MSB (bit 7) of the current frame is received.

Item	Page	Revision (See Manual for Details)
15.4.3 Clock Synchronous Mode Figure 15.12 Sample Flowchart for Receiving Serial Data (Clock Synchronous Mode) (2)	790	Figure title and description amended Description amended 2. Receive data is shifted into SCRSR in order from the LSB to the MSB (when LSB-first transfer is selected).
Figure 15.13 Example of SCI Receive Operation (when LSB-First Transfer is Selected)	791	Figure title amended
Figure 15.14 Sample Flowchart for Transmitting/Receiving Serial Data (Clock Synchronous Mode)	792	Figure and title amended [4] Serial transmission/reception continuation procedure: To continue serial transmission/reception, before the last bit of the current frame is received, finish reading the RDRF flag, reading SCRDR, and clearing the RDRF flag to 0. Also, before the last bit of the current frame is transmitted, read 1 from the TDRE flag to confirm that writing is possible. Then write data to SCTDR and clear the TDRE flag to 0.
15.4.5 Multiprocessor Serial Data Transmission	795	Description amended Figure 15.16 shows a sample flowchart for multiprocessor serial data transmission. For an ID transmission cycle, set the MPBT bit in SCSSR to 1 before transmission, and maintain the MPBT value at 1 until the ID transmission actually completes. For a data transmission cycle, ...
Figure 15.16 Sample Multiprocessor Serial Transmission Flowchart	796	Figure replaced
15.4.6 Multiprocessor Serial Data Reception Figure 15.17 Example of SCI Operation in Reception (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit, LSB-first)	797	Figure title amended
Figure 15.18 Sample Multiprocessor Serial Reception Flowchart (1)	798	Figure replaced

Item	Page	Revision (See Manual for Details)
15.5 Interrupt Sources and DMAC/DTC	800	Title and description amended Table 15.14 shows the interrupt sources and priority. … When the TDRE flag in the serial status register (SCSSR) is set to 1, a TXI interrupt request is generated. The TXI interrupt request can be used to activate the direct memory access controller (DMAC) or the data transfer controller (DTC) to transfer data. When the DMAC is activated to transfer data, the TDRE flag is cleared to 0 automatically when data is written to the transmit data register (SCTDR), and no TXI interrupt request to the CPU is generated. When the DTC is activated to transfer data, and if the value of DTC's DISEL bit is 0 and the transfer counter value is other than 0, the TDRE flag is cleared to 0 automatically when data is written to SCTDR, and no TXI interrupt request to the CPU is generated. However, if the value of the DISEL bit is 0 and the transfer counter value is 0, or if the value of the DISEL bit is 1, the TDRE flag is not cleared to 0 automatically when data is written to SCTDR, and a TXI interrupt request to the CPU is generated after the data write to SCTDR. When the RDRF flag in SCSSR is set to 1, an RXI interrupt request is generated. The RXI interrupt request can be used to activate the DMAC or DTC to transfer data. When the DMAC is activated to transfer data, the RDRF flag is cleared to 0 automatically when data is read from the receive data register (SCRDR), and no RXI interrupt request to the CPU is generated. When the DTC is activated to transfer data, and if the value of DTC's DISEL bit is 0 and the transfer counter value is other than 0, the RDRF flag is cleared to 0 automatically when data is read from the receive data register (SCRDR), and no RXI interrupt request to the CPU is generated. However, if the value of the DISEL bit is 0 and the transfer counter value is 0, or if the value of the DISEL bit is 1, the RDRF flag is not cleared to 0 automatically when data is read from SCRDR, and an RXI interrupt request to the CPU is generated after the data write to SCRDR. When the ORER, FER, or PER flag in SCSSR is set to 1, an ERI interrupt request is generated. This request cannot be used to activate the DMAC or DTC. … Note that setting the EIO bit to 1 will prevent the DMAC or DTC from transferring received data because no RXI interrupt requests are generated.

Item	Page	Revision (See Manual for Details)
15.5 Interrupt Sources and DMAC/DTC Table 15.14 SCI Interrupt Sources	801	Table amended

Interrupt Source	Description	Interrupt Enable Bit	DMAC/DTC Activation	Priority
ERI	Interrupt caused by receive error (ORER, FER, or PER)	RIE = 1	Not possible	High
RXI	Interrupt caused by receive data full (RDRF)	RIE = 1 and EIO = 0	Possible	
TXI	Interrupt caused by transmit data empty (TDRE)	TIE = 1	Possible	
TEI	Interrupt caused by transmit end (TEND)	TEIE = 1	Not possible	Low

15.6 Serial Port Register (SCSPTR) and SCI Pins Figure 15.19 SPB1IO Bit, SPB1DT bit, and SCK Pin	801	Figure and title amended

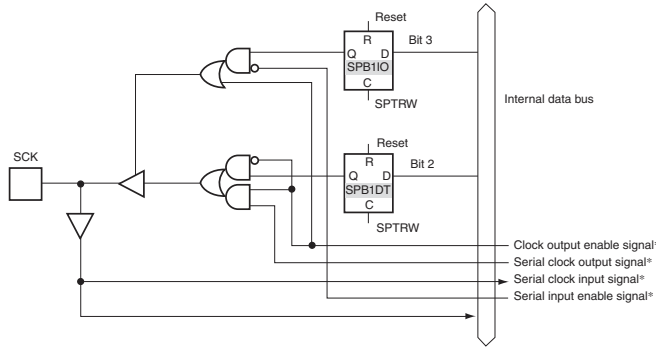
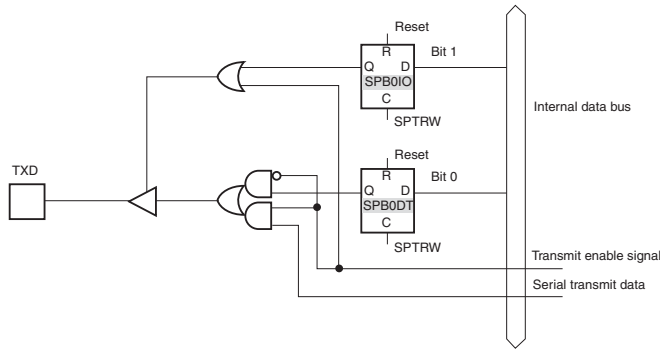


Figure 15.20 SPB0IO Bit, SPB0DT bit, and TXD Pin	802	Figure and title amended



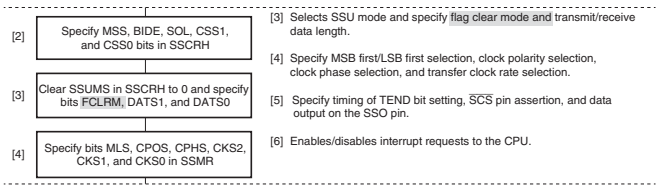
Item	Page	Revision (See Manual for Details)
15.7.3 Break Detection and Processing	804	Description amended Break signals can be detected by reading the RXD pin directly when a framing error (FER) is detected. In the break state the input from the RXD pin consists of all 0s, so the FER flag is set and the parity error flag (PER) may also be set. After a break is received, the SCI halts receive operation. At this time not only transfer of receive data from SCRSR to SCRDR stops; setting in SCRSR of serial data input on the RXD pin stops as well. To restart receive operation, input a high-level signal on the RXD pin, and clear the overrun error (ORER), FER, and PER flags.
15.7.4 Sending a Break Signal		Description amended To send a break signal during serial transmission, set the SPB0IO bit to 1 and clear the SPB0DT bit to 0 (low level), then clear the TE bit to 0 (halting transmission). When the TE bit is cleared to 0, the transmitter is initialized regardless of the current transmission state, and the low level is output from the TXD pin.
16.7.3 Break Detection and Processing	873	Description amended When data containing a framing error is received, and then space 0 (low level) is input for more than one frame length, a break (BRK) is detected. When a break is detected, not only the transfer of receive data (H'00) to SCFRDR but also the setting in SCRSR of serial data input on the RXD pin is stopped. If the RIE or REIE bit in SCSCR is set to 1, a break interrupt request (BRI) is issued. Reception resumes when the break ends and the receive signal is mark 1 (high level). It is also possible to perform break detection by reading the value of the RXD pin directly when a framing error (FER) is detected. Use the port register to read the value of the RXD pin. In the break state the input from the RXD pin consists of all 0s, so the FER flag is set and the parity error flag (PER) may also be set.

Item	Page	Revision (See Manual for Details)
17.3.5 SS Status Register (SSSR)	888 to 890	Table and note amended

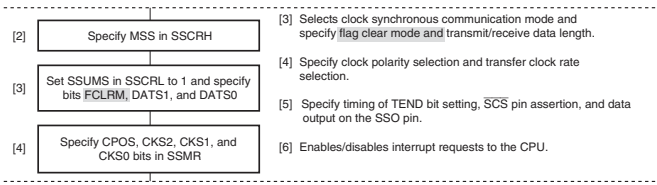
Bit	Bit Name	Initial Value	R/W	Description
3	TEND	0	R/W	Transmit End ... [Clearing conditions] • When writing 0 after reading TEND = 1 • When writing data to SSTDR with FCLRM = 1 • When the DTC is activated by an SSTXI interrupt and transmit data is written to SSTDR while the DISEL bit in MRB of the DTC is 0 (except when DTC transfer counter value is H'0000)*1
2	TDRE	1	R/W	Transmit Data Empty ... • When the DTC is activated by an SSTXI interrupt and transmit data is written to SSTDR while the DISEL bit in MRB of the DTC is 0 (except when DTC transfer counter value is H'0000)*1
1	RDRF	0	R/W	Receive Data Register Full ... • When the DTC is activated by an SSRXI interrupt and receive data is read into SSRDR while the DISEL bit in MRB of the DTC is 0 (except when DTC transfer counter value is H'0000)*1

Note: * Clearing the FCLRM bit to 0 enables transfer by the DTC.

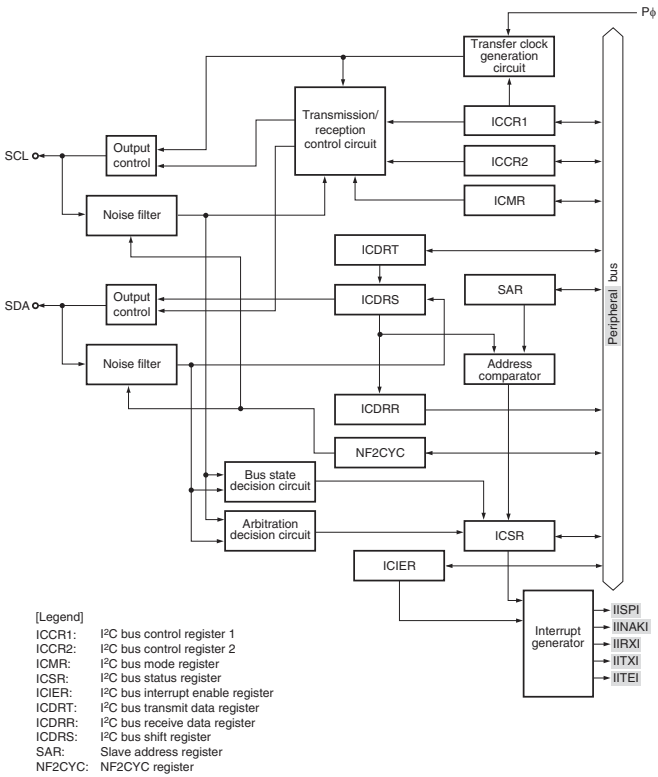
17.4.5 SSU Mode (1) Initial Settings in SSU Mode Figure 17.4 Example of Initial Settings in SSU Mode	901	Figure amended



17.4.7 Clock Synchronous Communication Mode (1) Initial Settings in Clock Synchronous Communication Mode Figure 17.12 Example of Initial Settings in Clock Synchronous Communication Mode	911	Figure amended



17.6.5 Note on Master Transmission and Master Transmission/Reception Operations in SSU Mode	919	Title and description amended To perform master transmission or transmission/reception in SSU mode, perform one of the following operations:

Item	Page	Revision (See Manual for Details)
Section 18 I^2C Bus Interface 2 (IIC2)	921	Description amended The I^2C bus interface 2 conforms to and provides a subset of the Philips I^2C (Inter-IC) bus interface functions. However, the configuration of the registers that control the I^2C bus differs partly from the Philips register configuration.
18.1 Features	922	Description added Figure 18.1 shows a block diagram of the I_2C bus interface.
Figure 18.1 Block Diagram of I^2C Bus Interface 2		Figure amended 
18.2 Input/Output Pins	923	Description added Figure 18.2 shows an example of I/O pin connections to external circuits.

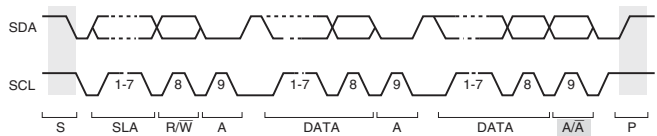
Item	Page	Revision (See Manual for Details)
18.3 Register Descriptions Table 18.2 Register Configuration	924	Table amended Register Name: I²C bus receive data register — Abbreviation: ICDRR — R/W: R — Initial value: H'FF — Address: H'FFFFCD87 — Access Size: 8 Register Name: NF2CYC register — Abbreviation: NF2CYC — R/W: R/W — Initial value: H'00 — Address: H'FFFFCD88 — Access Size: 8
18.3.1 I²C Bus Control Register 1 (ICCR1)		Description added ICCR1 is initialized to H'00 by a power-on reset.
18.3.2 I²C Bus Control Register 2 (ICCR2)	927	Description added ICCR2 is initialized to H'7D by a power-on reset.
	928	Table amended Bit: 1 — Bit Name: IICRST — Initial Value: 0 — R/W: R/W — Description: IIC Control Part Reset IICRST resets the BC[2:0] bits in the ICMR register and the internal circuits of IIC2. If the MCU hangs because of a communication failure while the I²C bus is operating, the BC[2:0] bits in the ICMR register and the internal circuits of IIC2 can be reset by setting the IICRST bit to 1.
	928, 929	Table note added Notes: When 1 is written to the IICRST bit in ICCR2, the state becomes as follows. • The SDAO and SCLO bits in ICCR2 are set to 1. • If the module is in master transmit mode or slave transmit mode, the TDRE bit in ICSR is set to 1. • Writing to the BBSY, SCP, and SDAO bits in ICCR2 is invalid while a reset is being applied by writing 1 to IICRST. • Writing 1 to IICRST does not clear the BBSY bit in ICCR2 to 0. However, if the states of the SCL and SDA pins lead to the generation of a stop condition, (rising edge on SDA while SCL is at the high level), the BBSY bit may be cleared to 0 as a result. This can also affect other bits in the same way. • Data transfer stops while a reset is being applied by writing 1 to IICRST. However, functions for detecting start conditions, stop conditions, and failure in bus contention continue to operate. Signals input to the SCL and SDA pins may alter the states of ICCR1, ICCR2, and ICSR.

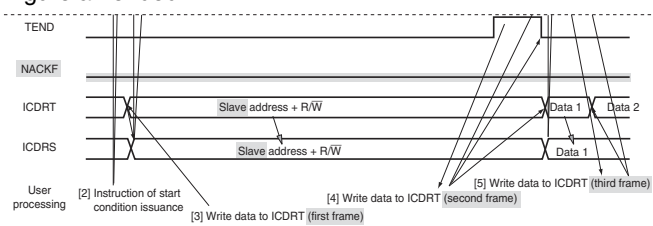
Item	Page	Revision (See Manual for Details)
18.3.3 I^2C Bus Mode Register (ICMR)	929	Description added ICMR is initialized to H'38 by a power-on reset. Bits BC[2:0] are initialized to B'000 by the IICRST bit in ICCR2.
	930	Table amended **Bit** / **Bit Name** / **Initial Value** / **R/W** / **Description** 2 to 0 / BC[2:0] / 000 / R/W / Bit Counter These bits specify the number of bits to be transferred next. When read, the remaining number of transfer bits is indicated. With the I^2C bus format, the data is transferred with one addition acknowledge bit. Should be made between transfer frames. If these bits are set to a value other than B'000, the setting should be made while the SCL pin is low. The value returns to B'000 at the end of a data transfer, including the acknowledge bit. These bits are automatically set to B'111 after a stop condition is detected. These bits are cleared by a power-on reset, software standby mode, and module standby mode. These bits are also cleared by setting IICRST of ICCR2 to 1. With the clock synchronous serial format, these bits should not be modified.
18.3.4 I^2C Bus Interrupt Enable Register (ICIER)	931	Description added ICIER is initialized to H'00 by a power-on reset.
	932	Table amended **Bit** / **Bit Name** / **Initial Value** / **R/W** / **Description** 4 / NAKIE / 0 / R/W / NACK Receive Interrupt Enable NAKIE enables or disables the NACK detection and arbitration lost/overrun error interrupt request (IINAKI) when the NACKF or AL/OVE bit in ICSR is set to 1. IINAKI can be canceled by clearing the NACKF, AL/OVE, or NAKIE bit to 0. 0: NACK detection and arbitration lost/overrun error interrupt request (IINAKI) is disabled. 1: NACK detection and arbitration lost/overrun error interrupt request (IINAKI) is enabled.
18.3.5 I^2C Bus Status Register (ICSR)	933	Description added ICSR is initialized to H'00 by a power-on reset.
		Bit table amended

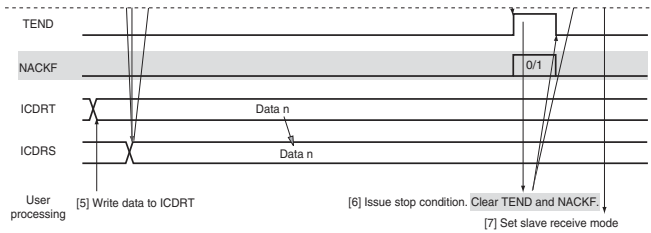
Bit:	7	6	5	4	3	2	1	0
	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

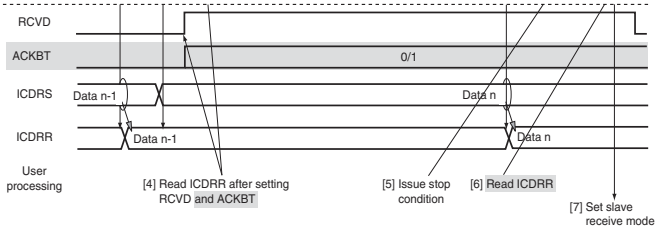
Note: * These flags can be cleared only by writing 0 after reading the flag's value as 1.

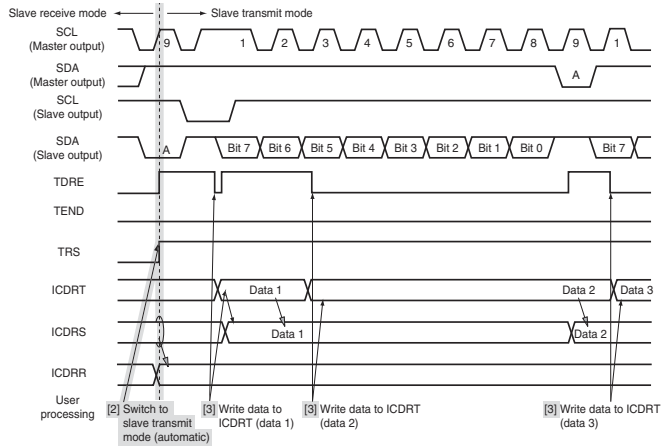
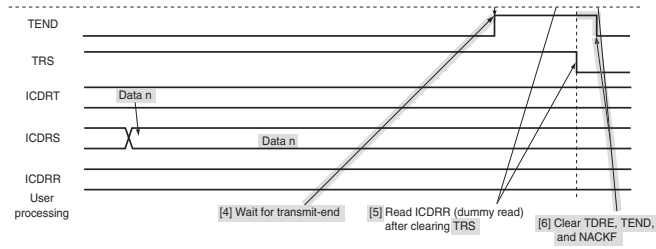
	936	Table note amended Notes: 1. These flags can be cleared only by writing 0 after reading the flag's value as 1. 2. When NACKF = 1 is detected, be sure to clear NACKF in the transfer end processing. Until the flag is cleared, next transmission or reception cannot be started.

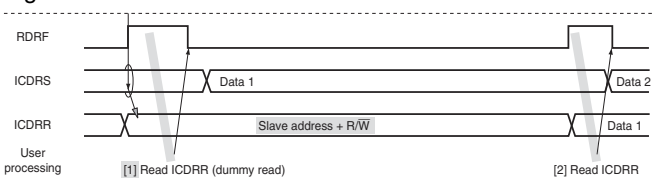
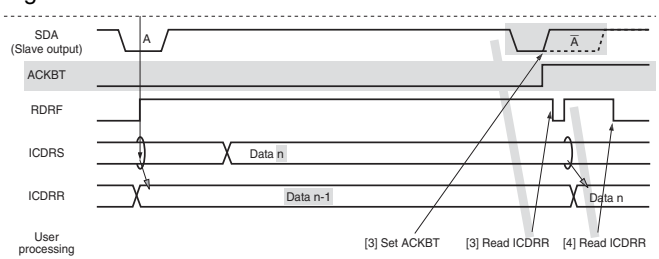
Item	Page	Revision (See Manual for Details)
18.3.6 I^2C Bus Slave Address Register (SAR)	937	Description added SAR is initialized to H'00 by a power-on reset.
18.3.7 I^2C Bus Transmit Data Register (ICDRT)	938	Description amended ICDRT is an 8-bit readable/writable register that stores the transmit data. When ICDRT detects the space in the shift register (ICDRS), it transfers the transmit data which is written in ICDRT to ICDRS and starts transferring data. If the next transfer data is written to ICDRT during transferring data of ICDRS, continuous transfer is possible. If ICDRT is read while the LSB-first setting is enabled (MLS bit in ICMR set to 1), data is read with the MSB–LSB order reversed relative to the value that was written to ICDRT. ICDRT is initialized to H'FF.
18.3.8 I^2C Bus Receive Data Register (ICDRR)	938	Description and bit table amended ICDRR is initialized to H'FF by a power-on reset. Bit: 7 6 5 4 3 2 1 0 Initial value: 1 1 1 1 1 1 1 1 R/W: R R R R R R R R
18.3.10 NF2CYC Register (NF2CYC)	939	Description added NF2CYC is initialized to H'00 by a power-on reset.
18.4.1 I^2C Bus Format Figure 18.4 I^2C Bus Timing	940	Figure amended  Description amended R/$\overline{W}$: Indicates the direction of data transfer: from the slave device to the master device when R/$\overline{W}$ is high, or from the master device to the slave device when R/$\overline{W}$ is low.

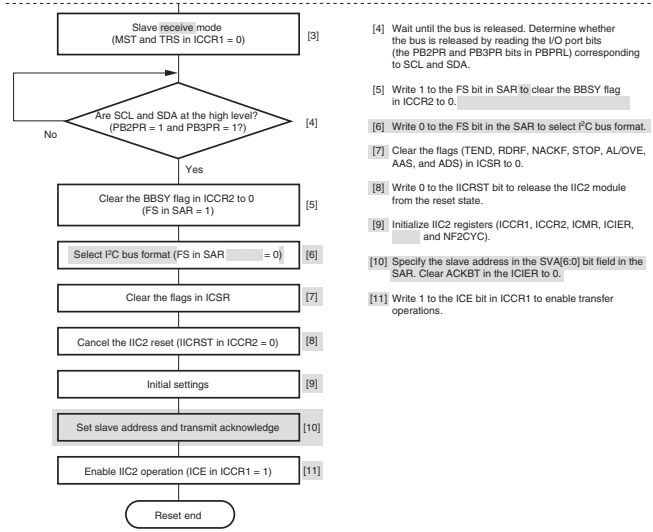
Item	Page	Revision (See Manual for Details)
18.4.2 Master Transmit Operation	941	Description amended 1. Initialize IIC2 (figure 18.7). After initialization, set the ICE bit in ICCR1 to 1. 2. ... 3. After issuing the start condition, write the transmit data to ICDRT (the 1st frame consists of data indicating the slave address and the R/W bit). At this time, TDRE is automatically cleared to 0, and data is transferred from ICDRT to ICDRS. TDRE is set again. 4. When transmission of first frame data is completed, TEND in ICSR is set to 1 at the rise of the 9th transmit clock pulse. Read the ACKBR bit in ICIER, and confirm that the slave device has been selected. Then, write second frame data to ICDRT. When ACKBR is 1, the slave device has not been acknowledged, so issue the stop condition. To issue the stop condition, write 0 to BBSY and to SCP after waiting for 0 to be read from SCLO in ICCR2. SCL is fixed low until the transmit data is prepared or the stop condition is issued. 5. The transmit data after the second frame is written to ICDRT every time TDRE is set. 6. Write the last transmit data to ICDRT, then wait until the TEND bit is set to 1 (the end of the last byte of data transmission). Alternately, wait for a NACK (NACKF in ICSR = 1) from the receive device while the ACKE bit in ICIER is set to 1. Once TEND or NACKF is set to 1, wait for 0 to be read from SCLO in ICCR2. Next, clear TEND and NACK, and issue the stop condition. 7. When the STOP bit in ICSR is set to 1, clear MST and TRS to return to slave receive mode.
Figure 18.5 Master Transmit Mode Operation Timing (1)	942	Figure amended 

Item	Page	Revision (See Manual for Details)
18.4.2 Master Transmit Operation Figure 18.6 Master Transmit Mode Operation Timing (2)	942	Figure amended 
Figure 18.7 Flowchart of Initialization of I²C Bus Interface 2	943	Figure added
18.4.3 Master Receive Operation	944	Description amended The reception procedure and operations in master receive mode are shown below. For operation up to transmission of the 1st frame (slave address + R/$\overline{W}$), see 18.4.2, Master Transmit Operation. ... 3. After the reception of first frame data is completed, the RDRF bit in ICSR is set to 1 at the rise of 9th receive clock pulse. At this time, the receive data is read by reading ICDRR, and RDRF is cleared to 0. The continuous reception is performed by reading ICDRR every time RDRF is set. If reading of ICDRR cannot take place before the rising edge of the 8th clock pulse of SCL, set RCVD in ICCR1 to 1 and perform communication one byte at a time. 4. If next frame is the last receive data, set the RCVD bit in ICCR1 and ACKBT bit in ICIER to 1 before reading ICDRR. This enables the issuance of the stop condition after the next reception. 5. When RDRF is set to 1 at the 9th receive clock pulse, wait until 0 is read from SCLO in ICCR2. Then issue the stop condition. 6. When STOP is set to 1, read the final receive data from ICDRR. 7. Clear RCVD and MST to 0 to return to slave receive mode.

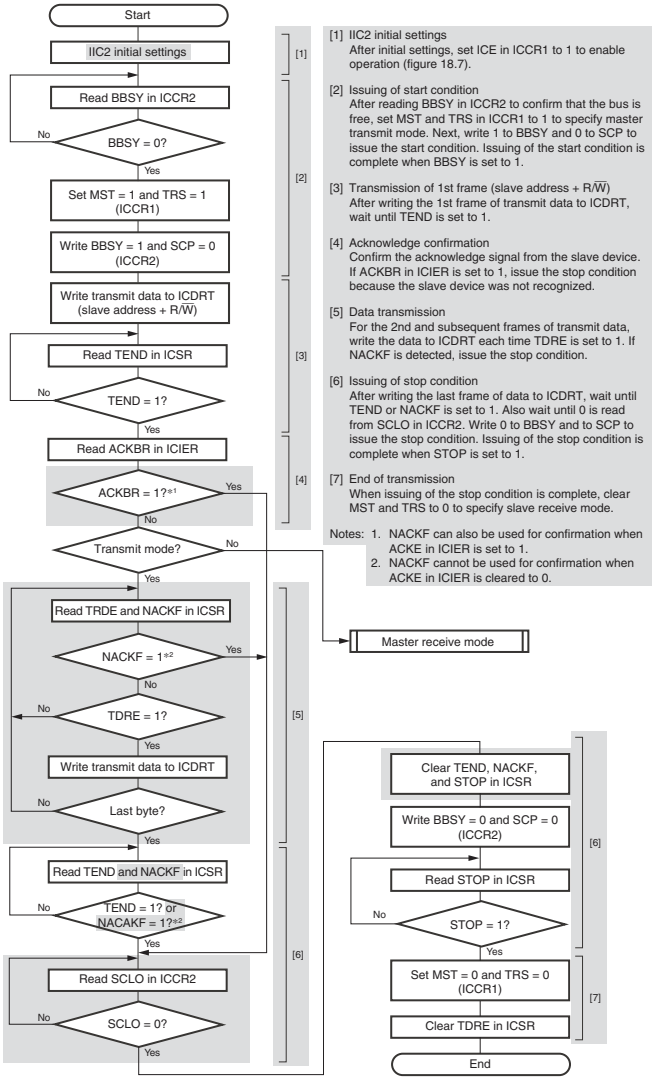
Item	Page	Revision (See Manual for Details)
18.4.3 Master Receive Operation Figure 18.9 Master Receive Mode Operation Timing (2)	946	Figure amended 
18.4.4 Slave Transmit Operation	947	Description amended 1. Initialize IIC2 (figure 18.7). After initialization, set the ICE bit in ICCR1 to 1. Set the MST and TRS bits in ICCR1 to select slave receive mode, and wait until the slave address matches. 2. When the slave address matches in the first frame following detection of the start condition, the slave device outputs the level specified by ACKBT in ICIER to SDA, at the rise of the 9th clock pulse. At this time, if the 8th bit data (R/$\overline{W}$) is high, the TRS bit in ICCR1 and the TDRE bit in ICSR are set to 1, and the mode changes to slave transmit mode automatically. If the 8th bit of data is low-level, operation continues in slave receive mode. 3. Write the transmit data to ICDRT. At this time, TDRE is automatically cleared to 0, the data is transferred from ICDRT to ICDRS, and TDRE is set to 1 once again. Write the subsequent transmit data to ICDRT each time TDRE is set to 1. 4. Write the last transmit data to ICDRT, wait until TEND is set to 1 (final frame transmit-end). Alternately, wait for a NACK from the receive device (NACKF in ICSR = 1) with ACKE in ICIER set to 1. 5. Clear TRS for the end processing, and read ICDRR (dummy read). SCL is free. 6. Clear TDRE, TEND, and NACKF.

Item	Page	Revision (See Manual for Details)
18.4.4 Slave Transmit Operation Figure 18.10 Slave Transmit Mode Operation Timing (1)	948	Figure amended 
Figure 18.11 Slave Transmit Mode Operation Timing (2)	949	Figure amended 

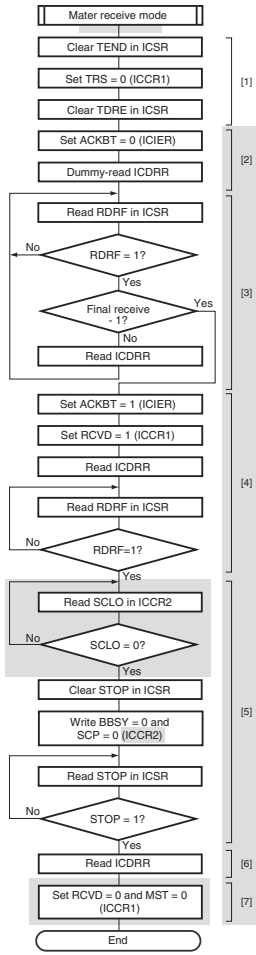
Item	Page	Revision (See Manual for Details)
18.4.5 Slave Receive Operation	950	Description amended The reception procedure and operations in slave receive mode are described below. For operation up to reception of the 1st frame (slave address + R/$\overline{W}$), see 18.4.4, Slave Transmit Operation. 1. Perform a dummy read of ICDRR. (The read data indicates slave address + R/$\overline{W}$, and is therefore unnecessary.) The slave device outputs to SDA the level indicated by the setting of ACKBT in ICIER at the 9th clock pulse of the receive clock. 2. After reception of one frame of data finishes, RDRF in ICSR is set to 1 at the rising edge of the 9th receive clock pulse. At this time, the received data can be read by reading ICDRR, and RDRF is cleared to 0 simultaneously. Continuous receive operation can be accomplished by reading ICDRR each time RDRF is set to 1. If 8th receive clock pulse falls while RDRF is 1, SCL is fixed low until ICDRR is read. 3. If the next receive operation is the final frame, set ACKBT in ICIER to 1 before reading ICDRR. 4. When RDRF in ICSR is set to 1, read the final receive data from ICDRR.
Figure 18.12 Slave Receive Mode Operation Timing (1)		Figure amended 
Figure 18.13 Slave Receive Mode Operation Timing (2)	951	Figure amended 

Item	Page	Revision (See Manual for Details)
18.4.6 Clock Synchronous Serial Format (1) Data Transfer Format Figure 18.14 Clock Synchronous Serial Transfer Format (LSB-First Operation)	951	Figure title amended
(2) Transmit Operation Figure 18.15 Transmit Mode Operation Timing (LSB-First Operation)	952	Figure title amended
(3) Receive Operation Figure 18.16 Receive Mode Operation Timing (LSB-First Operation)	954	Figure title amended
Figure 18.17 Operation Timing For Receiving One Byte (LSB-First Operation)		Figure title amended
18.4.8 Using the IICRST Bit to Reset I²C Bus Interface 2 Figure 18.19 Sequence for Using the IICRST Bit to Reset I²C Bus Interface 2	956	Figure amended 

Item	Page	Revision (See Manual for Details)
18.4.9 Example of Use Figure 18.20 Sample Flowchart for Master Transmit Mode	957	Figure amended



Item	Page	Revision (See Manual for Details)
18.4.9 Example of Use Figure 18.21 Sample Flowchart for Master Receive Mode	958	Figure amended



[1] Specification of master receive mode*1
After clearing TEND in ICSR to 0, clear TRS in ICCR1 to 0 to switch from master transmit mode to master receive mode. Then clear TDRE to 0.
See figure 18.20 for the processing up to transmission of the 1st frame (slave address + R/W).

[2] Start of reception*1*2
After setting ACKBT in ICIER to send an acknowledge signal to the transmit device, perform a dummy read of ICDRR. The dummy read causes clock output to start, and reception starts.

[3] Data reception*2
When reception of one frame finishes, RDRF in ICSR is set to 1. The received data can be read by reading ICDRR. Continuous receive operation can be accomplished by reading ICDRR each time RDRF is set to 1.

[4] Reception of final frame
If the next receive operation is the final frame, set RCVD and ACKBT to 1 before reading ICDRR. SCL is limited to low-level when receive finishes.

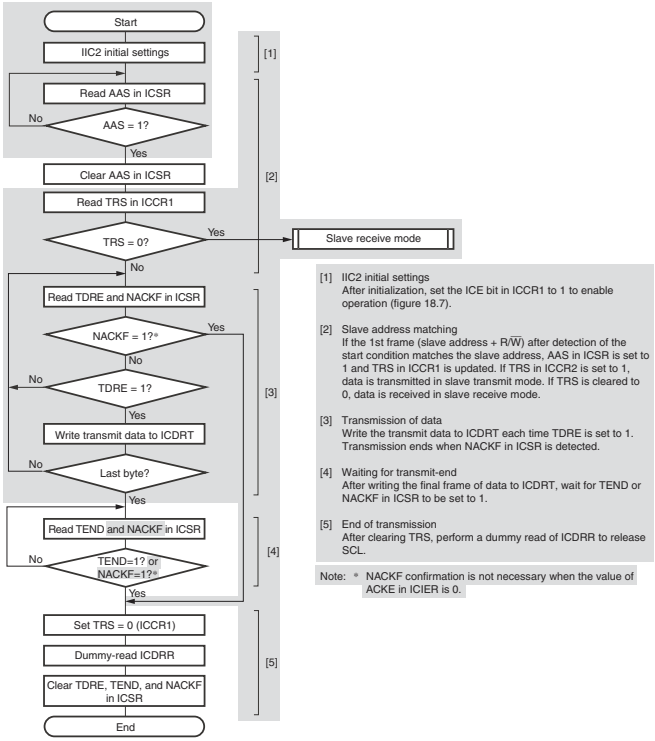
[5] Issuing of stop condition
When RDRF is set to 1, wait until 0 is read from SCLO in ICCR2. Next, write 0 to BBSY and to SCP to issue the stop condition. Issuing of the stop condition is complete when STOP in ICSR is set to 1.

[6] Reading of final receive data
When STOP is set to 1, read the final receive data from ICDRR.

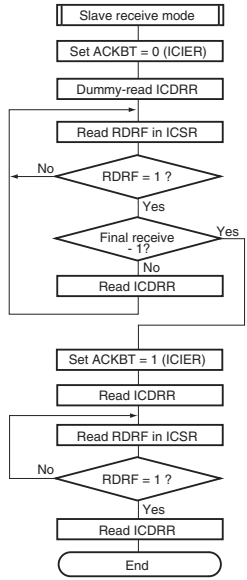
[7] End of reception
When issuing of the stop condition is complete, clear RCVD and MST to 0 to specify slave receive mode.

Notes: 1. Ensure that no interrupts occur during the processing of steps [1] and [2].
2. For one-byte reception, after processing step [1], steps [2] and [3] are omitted and processing jumps to step [4]. The reading of ICDRR in step [4] is a dummy read.

Item	Page	Revision (See Manual for Details)
18.4.9 Example of Use Figure 18.22 Sample Flowchart for Slave Transmit Mode	959	Figure amended



Item	Page	Revision (See Manual for Details)
18.4.9 Example of Use Figure 18.23 Sample Flowchart for Slave Receive Mode	960	Figure amended



[1] Start of reception*
After setting ACKBT in ICIER to send an acknowledge signal to the transmit device, perform a dummy read of ICDRR. (The read data indicates slave address + R/$\overline{W}$, and is therefore unnecessary.)

[2] Data reception*
When reception of one frame finishes, RDRF in ICSR is set to 1. The received data can be read by reading ICDRR. Continuous receive operation can be accomplished by reading ICDRR each time RDRF is set to 1.

[3] Reception of final frame
If the next receive operation is the final frame, set ACKBT to 1 before reading ICDRR.

[4] Reading of final receive data
When RDRF is set to 1, read the final receive data from ICDRR.

Note: * For one-byte reception, processing of steps [1] and [2] is omitted and processing jumps to step [3]. The reading of ICDRR in step [3] is a dummy read.

18.5 Interrupt Sources and DTC	961	Description replaced
Table 18.4 Interrupt Requests	962	Table amended

Interrupt Request	Abbreviation	Interrupt Condition	I^2C Bus Format	Clocked Synchronous Serial Format	DTC Activation	Priority
NACK detection	IINAKI*	{(NACKF = 1) + (AL/OVE = 1)} • (NAKIE = 1)	√	—	—	High
Arbitration lost/ overrun error			√	√	—	
Transmit end	IITEI	(TEND = 1) • (TEIE = 1)	√	√	—	
Stop condition detection	IISTPI	(STOP = 1) • (STIE = 1)	√	—	—	
Transmit data empty	IITXI	(TDRE = 1) • (TIE = 1)	√	√	√	
Receive data full	IIRXI	(RDRF = 1) • (RIE = 1)	√	√	√	Low

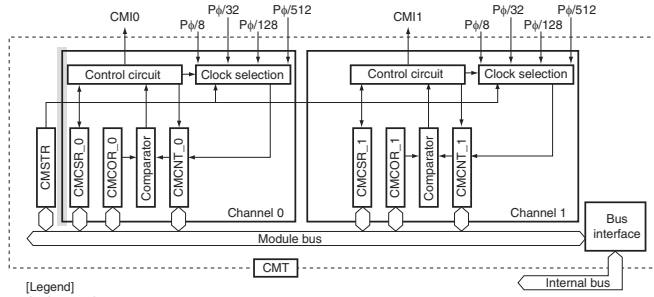
Note: * In the case of IINAKI, the IPR bit in the INTC, which determines the priority, is different. Depending on the setting of the IPR bit, the priority may be lower than that of IIRXI.

Item	Page	Revision (See Manual for Details)
18.6 Operation Using the DTC Table 18.5 Example of Processing Using DTC	963	Table amended and note added

Item	Master Transmit Mode	Master Receive Mode	Slave Transmit Mode	Slave Receive Mode
Slave address + R/$\overline{W}$ bit transmit/receive	Transmitted by DTC* (ICDRT writing)	Transmitted by CPU (ICDRT writing)	Received by CPU (ICDRR reading)	Received by CPU (ICDRR reading)
Dummy data read	—	Processed by CPU (ICDRR reading)	—	Processed by CPU (ICDRR reading)
Main data transmit/receive	Transmitted by DTC (ICDRT writing)	Received by DTC (ICDRR reading)	Transmitted by DTC (ICDRT writing)	Received by DTC (ICDRR reading)
Last frame processing	Not necessary	Received by CPU (ICDRR reading)	Not necessary	Received by CPU (ICDRR reading)
DTC transfer data frame count setting	Transmission: Actual data count + 1 (+1 is required for the slave address + R/$\overline{W}$ bit transfer)	Reception: Actual data count – 1 (–1 is required for processing of the last frame)	Transmission; Actual data count	Reception: Actual data count – 1 (–1 is required for processing of the last frame)

Note: * After issuing a start condition (writing 1 to BBSY and 0 to SCP), enable DTC transfers.

18.8.9 Notes on Switching from Master Transmit Mode to Master Receive Mode	967, 968	Section added
18.8.10 DTC Transfers Using the IIRXI Interrupt as the Source	968	Section added
18.8.11 DTC Transfers Using the IITXI Interrupt as the Source		Section added
20.1 Features	999	Description amended • Selection of four counter input clocks Any of four internal clocks (Pϕ/8, Pϕ/32, Pϕ/128, and Pϕ/512) can be selected independently for each channel. • Selection of DTC transfer request or interrupt request generation on compare match by DTC setting. • When not in use, the CMT can be stopped by halting its clock supply to reduce power consumption.

Item	Page	Revision (See Manual for Details)
20.1 Features Figure 20.1 Block Diagram of CMT	999	Figure amended  [Legend] CMSTR: Compare match timer start register CMCSR: Compare match timer control/status register CMCOR: Compare match timer constant register CMCNT: Compare match counter CMI: Compare match interrupt
20.2 Register Descriptions	1000	Description amended … For the states of these registers in each processing status, refer to section 27, List of Registers.
Table 20.1 Register Configuration		Table amended

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Common	Compare match timer start register	CMSTR	R/W	H'0000	H'FFFFCE00	8, 16, 32
0	Compare match timer control/ status register_0	CMCSR_0	R/W	H'0000	H'FFFFCE02	8, 16
	Compare match counter_0	CMCNT_0	R/W	H'0000	H'FFFFCE04	8, 16, 32
	Compare match constant register_0	CMCOR_0	R/W	H'FFFF	H'FFFFCE06	8, 16
1	Compare match timer control/ status register_1	CMCSR_1	R/W	H'0000	H'FFFFCE08	8, 16, 32
	Compare match counter_1	CMCNT_1	R/W	H'0000	H'FFFFCE0A	8, 16
	Compare match constant register_1	CMCOR_1	R/W	H'FFFF	H'FFFFCE0C	8, 16, 32

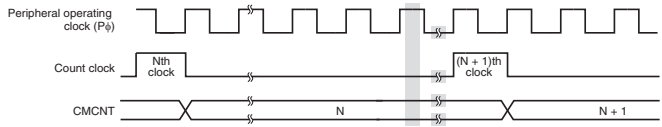
20.2.1 Compare Match Timer Start Register (CMSTR)	1001	Description added CMSTR is a 16-bit register that selects whether compare match counter (CMCNT) operates or is stopped. CMSTR is initialized to H'0000 when a power-on reset or a transition to standby mode occurs.

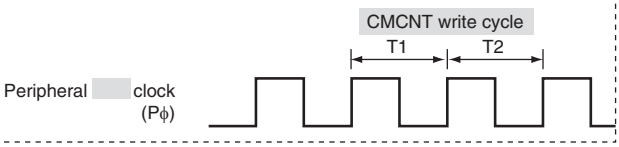
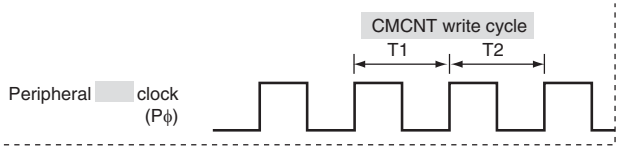
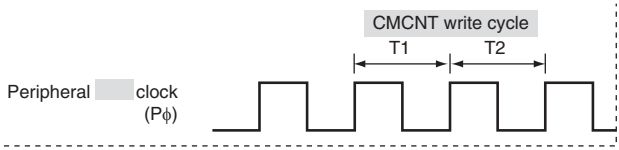
Item	Page	Revision (See Manual for Details)
20.2.2 Compare Match Timer Control/Status Register (CMCSR)	1002	Description and bit table note amended CMCSR is a 16-bit register that indicates compare match generation, enables/disables interrupts and selects the counter input clock. CMCSR is initialized to H'0000 when a power-on reset or a transition to standby mode occurs.  Note: * Only 0 can be written to clear the flag after 1 is read.
20.2.2 Compare Match Timer Control/Status Register (CMCSR)	1002, 1003	Table and note amended

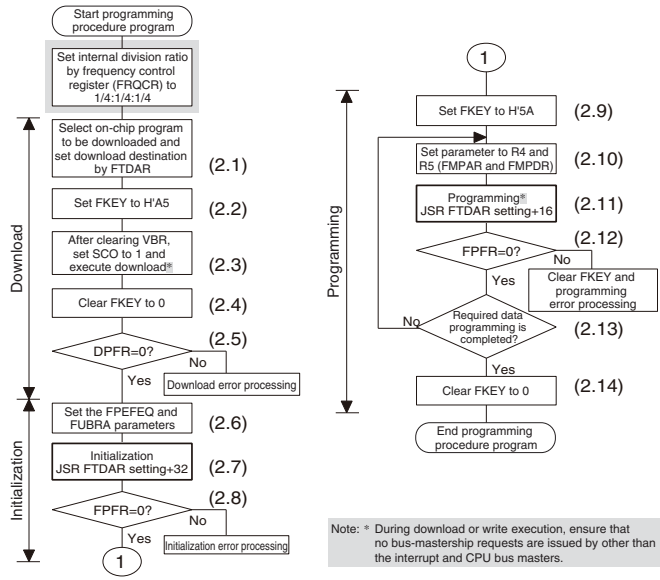
Bit	Bit Name	Initial value	R/W	Description
7	CMF	0	R/(W)*1	Compare Match Flag Indicates whether or not the values of CMCNT and CMCOR match. 0: CMCNT and CMCOR values do not match [Clearing conditions] • When a power-on reset or a transition to standby mode occurs • When 0 is written to this bit after reading CMF=1*2 • When CMT registers are accessed when the value of the DISEL bit of MRB in the DTC is 0 after activating the DTC by CMI interrupts (except when the DTC transfer counter value has become H'0000). 1: CMCNT and CMCOR values match [Setting condition] • When CMCNT and CMCOR values match
1, 0	CKS[1:0]	00	R/W	Clock Select Select the clock to be input to CMCNT from four internal clocks obtained by dividing the peripheral operating clock (Pϕ). When the STR bit in CMSTR is set to 1, CMCNT starts counting on the clock selected with CKS[1:0] bits. 00: Pϕ/8 01: Pϕ/32 10: Pϕ/128 11: Pϕ/512

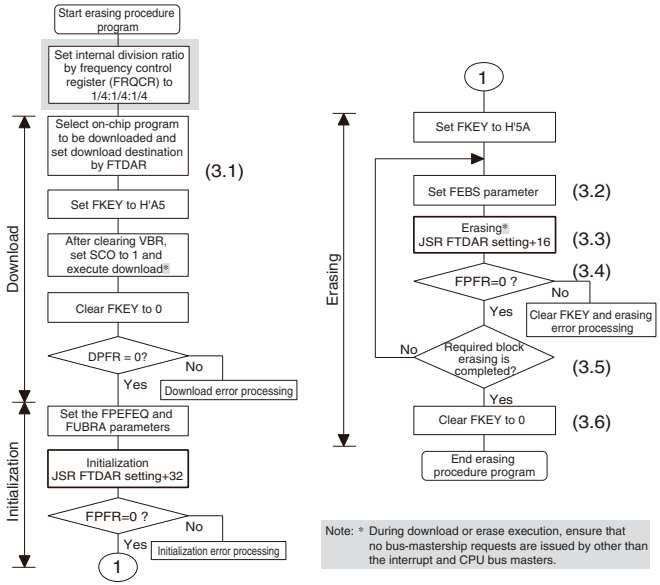
Notes: 1. Only 0 can be written to clear the flag after 1 is read.

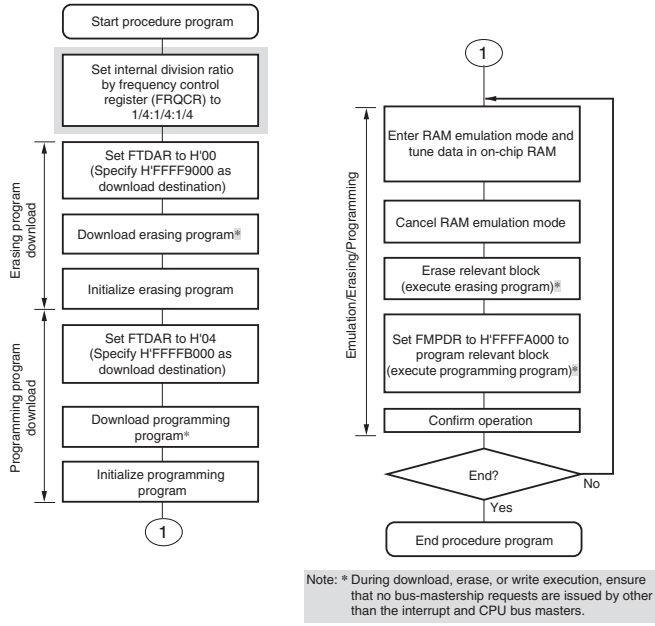
20.2.3 Compare Match Counter (CMCNT)	1003	Description amended When the value in CMCNT and the value in compare match constant register (CMCOR) match, CMCNT is cleared to H'0000 and the CMF flag in CMCSR is set to 1. CMCNT is initialized to H'0000 when a power-on reset or a transition to standby mode occurs.

Item	Page	Revision (See Manual for Details)
20.2.4 Compare Match Constant Register (CMCOR)	1003	Description amended CMCOR is a 16-bit register that sets the interval up to a compare match with CMCNT. CMCOR is initialized to H'FFFF when a power-on reset or a transition to standby mode occurs.
20.3.1 Interval Count Operation	1005	Description amended When an internal clock is selected with CKS[1:0] bits in CMCSR and the STR bit in CMSTR is set to 1, CMCNT starts counting up using the selected clock.
20.3.2 CMCNT Count Timing Figure 20.3 Count Timing	1005	Figure amended 
20.4.1 Interrupt Sources and DTC Transfer Requests	1006	Title and description amended The CMT has two channels, and each of them to which a different vector address is allocated has a compare match interrupt as shown in table 20.2. When both the interrupt request flag (CMF) and the interrupt enable bit (CMIE) are set to 1, the corresponding interrupt request is output. When the interrupt is used to activate a CPU interrupt, the priority of channels can be changed by the interrupt controller settings. For details, see section 6, Interrupt Controller (INTC). Clear the CMF bit to 0 from within the user exception handling routine. If this is not done, the interrupt will be generated again. If the next compare match sets the CMF flag before 0 is written to this flag after it has been read as 1, writing 0 to this flag will not clear it. Therefore, read this flag as 1 again before writing 0 to it. A compare match interrupt request can activate the data transfer controller (DTC). Since data transfer due to DTC activation automatically clears the flag while the DISEL bit of the DTC is 0 and the transfer counter value is not 0, an interrupt is not issued to the CPU in this case. However, if the DISEL bit and the transfer counter value are both 0, or the DISEL bit is 1, data transfer does not clear the flag, so an interrupt request for the CPU is generated after completion of the data transfer.

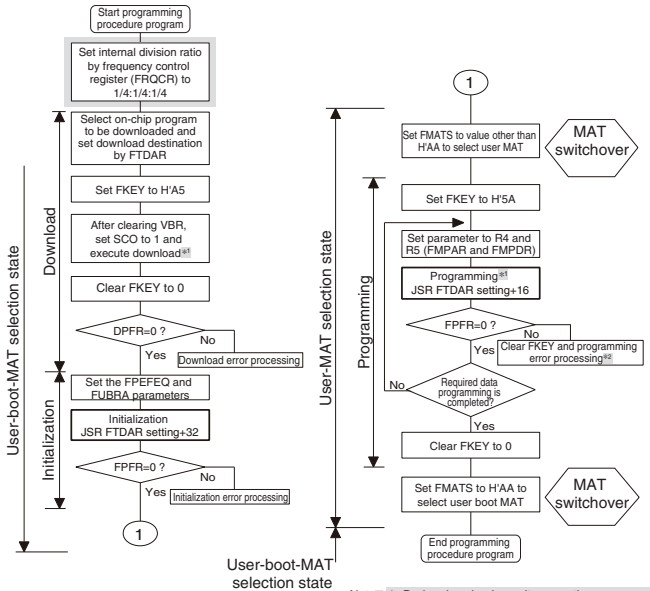
Item	Page	Revision (See Manual for Details)
20.4.1 Interrupt Sources and DTC Transfer Requests Table 20.2 Interrupt Sources	1006	Table amended Channel / Interrupt Source / Interrupt Enable Bit / Interrupt Flag / DTC Activation / Priority 0 / CMI_0 / CMIE of CMCSR_0 / CMF of CMCSR_0 / Possible / High 1 / CMI_1 / CMIE of CMCSR_1 / CMF of CMCSR_1 / Possible / Low
20.4.3 Timing of Clearing Compare Match Flag	1007	Description added The CMF bit in CMCSR is cleared by reading 1 from this bit, then writing 0. However, in the case of the DTC being activated, the CMF bit is automatically cleared to 0 when data is transferred by the DTC (except when the DTC's DISEL bit is cleared to 0 and the counter value is 0, or when the DISEL bit is set to 1).
20.5.2 Conflict between Write and Compare-Match Processes of CMCNT Figure 20.5 Conflict between Write and Compare-Match Processes of CMCNT	1008	Figure amended 
20.5.3 Conflict between Word-Write and Count-Up Processes of CMCNT	1009	Description amended Even when the count-up occurs in the T2 cycle while writing to CMCNT counter in words, the writing has priority over the count-up.
Figure 20.6 Conflict between Word-Write and Count-Up Processes of CMCNT		Figure amended 
20.5.4 Conflict between Byte-Write and Count-Up Processes of CMCNT Figure 20.7 Conflict between Byte-Write and Count-Up Processes of CMCNT	1010	Figure amended 

Item	Page	Revision (See Manual for Details)
23.2.6 Programming/ Erasing Interface (4) Programming/Erasing Execution	1250	Description amended The area to be programmed must be erased in advance when programming flash memory. Ensure that no interrupts, including NMI and IRQ, are generated during programming or erasure.
23.5.2 User Program Mode (2) Programming Procedure in User Program Mode Figure 23.11 Programming Procedure	1284	Figure amended 
	1286	Description amended In the download processing, the values of the general registers of the CPU are retained. During download processing, ensure that no bus-mastership requests are issued by other than the interrupt and CPU bus masters. For details, see section 23.8.2, Interrupts during Programming/Erasing. Since a stack area of maximum 128 bytes is used, an area of at least 128 bytes must be saved before setting the SCO bit to 1. (2.4) FKEY is cleared to H'00 for protection.

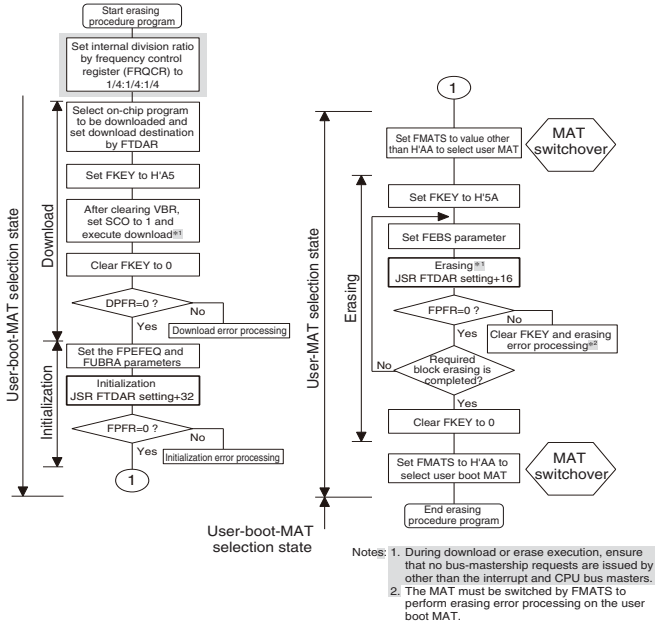
Item	Page	Revision (See Manual for Details)
23.5.2 User Program Mode (2) Programming Procedure in User Program Mode	1288	Description added — Since the stack area is used in the programming program, a stack area of maximum 128 bytes must be reserved in RAM. — During write processing, ensure that no bus-mastership requests are issued by other than the interrupt and CPU bus masters. For details, see section 23.8.2, Interrupts during Programming/Erasing.
Figure 23.12 Erasing Procedure	1289	Figure amended 
	1290	Description added — Since the stack area is used in the erasing program, a stack area of maximum 128 bytes must be reserved in RAM. — During erase processing, ensure that no bus-mastership requests are issued by other than the interrupt and CPU bus masters. For details, see section 23.8.2, Interrupts during Programming/Erasing.

Item	Page	Revision (See Manual for Details)
23.5.2 User Program Mode (2) Programming Procedure in User Program Mode Figure 23.13 Sample Procedure of Repeating RAM Emulation, Erasing, and Programming (Overview)	1292	Figure amended  Note: * During download, erase, or write execution, ensure that no bus-mastership requests are issued by other than the interrupt and CPU bus masters.

Item	Page	Revision (See Manual for Details)
23.5.3 User Boot Mode Figure 23.14 Procedure for Programming User MAT in User Boot Mode	1294	Figure amended



Notes: 1. During download or write execution, ensure that no bus-mastership requests are issued by other than the interrupt and CPU bus masters.
2. The MAT must be switched by FMATS to perform the programming error processing in the user boot MAT.

Item	Page	Revision (See Manual for Details)
23.5.3 User Boot Mode Figure 23.15 Procedure for Erasing User MAT in User Boot Mode	1296	Figure amended 
23.8.2 Interrupts during Programming/Erasing	1306	Description amended 2. Generation of interrupt requests during downloading Securing of bus-mastership by other than the interrupt and CPU bus masters (DMA transfer, DTC transfer, SDRAM refresh) is prohibited during SCO download execution. (2) Interrupts during programming/erasing Securing of bus-mastership by other than the interrupt and CPU bus masters (DMA transfer, DTC transfer, SDRAM refresh) is prohibited during programming or erase execution by a downloaded on-chip program.
23.9.2 Areas for Storage of the Procedural Program and Data for Programming	1340	Description amended 5. The flash memory is not accessible during programming/erasing operations. Therefore, the programming/erasing program must be downloaded to on-chip RAM in advance. Areas for executing each procedure program for initiating programming/erasing and the user program at the user branch destination for programming/erasing must be located in on-chip memory other than flash memory or the external address space.

Item	Page	Revision (See Manual for Details)
23.9.2 Areas for Storage of the Procedural Program and Data for Programming Table 23.18 (1) Usable Area for Programming in User Program Mode	1341	Table amended

	Storable/Executable Area			Selected MAT	
Item	On-Chip RAM	User MAT	External Space	User MAT	Embedded Program Storage MAT
Judging initialization result	√	√	√	√	
Initialization error processing	√	√	√	√	
Writing H'5A to key register	√	√	√	√	

Table 23.18 (2) Usable Area for Erasure in User Program Mode	1342	Table amended

	Storable/Executable Area			Selected MAT	
Item	On-Chip RAM	User MAT	External Space	User MAT	Embedded Program Storage MAT
Judging initialization result	√	√	√	√	
Initialization error processing	√	√	√	√	
Writing H'5A to key register	√	√	√	√	

Table 23.18 (3) Usable Area for Programming in User Boot Mode	1343	Table amended

	Storable/Executable Area			Selected MAT		
Item	On-Chip RAM	User Boot MAT	External Space	User MAT	User Boot MAT	Embedded Program Storage Area
Initialization error processing	√	√	√		√	
Switching MATs by FMATS	√	X	X	√		
Writing H'5A to Key Register	√	X	√	√		

Table 23.18 (4) Usable Area for Erasure in User Boot Mode	1345	Table amended

	Storable/Executable Area			Selected MAT		
Item	On-Chip RAM	User Boot MAT	External Space	User MAT	User Boot MAT	Embedded Program Storage Area
Judging initialization result	√	√	√		√	
Initialization error processing	√	√	√		√	
Switching MATs by FMATS	√	X	X		√	

Item	Page	Revision (See Manual for Details)
23.10 Programmer Mode	1346	Description amended ... Use a PROM programmer that supports the Renesas 256 or 512-kbyte flash memory on-chip MCU device type.
27.1 Register Address Table (In the Order from Lower Addresses)	1371 to 1376, 1383, 1384	Table amended

Register Name	Abbreviation	No. of Bits	Address	Module	Access Size	No. of Access States	Connected Bus Width
Timer control register_3	TCR_3	8	H'FFFFC200	MTU2	8, 16, 32	Pϕ reference	16 bits
Timer general register B_4	TGRB_4	16	H'FFFFC21E	MTU2	16	Pϕ reference	16 bits
Timer read/write enable register	TRWER	8	H'FFFFC284	MTU2	8	Pϕ reference	16 bits
Timer status register_2	TSR_2	8	H'FFFFC405	MTU2	8	Pϕ reference	16 bits
Timer control register_3S	TCR_3S	8	H'FFFFC600	MTU2S	8, 16, 32	Pϕ reference	16 bits
Timer gate control register S	TGCRS	8	H'FFFFC60D	MTU2S	8	Pϕ reference	16 bits
Timer A/D converter start request cycle set register B_4S	TADCORB_4S	16	H'FFFFC646	MTU2S	16	Pϕ reference B:2	16 bits
Common control register	CMNCR	32	H'FFFFF000	BSC	32	Bϕ reference	16 bits
CS0 space bus control register	CS0BCR	32	H'FFFFF004	*1: Read	32	L:1*1	
CS1 space bus control register	CS1BCR	32	H'FFFFF008	*2: Write	32	L:3*2	
RAM emulation register	RAMER	16	H'FFFFF108	FLASH	16	Bϕ reference W:1	16 bits
Break address register A	BARA	32	H'FFFFF300	UBC	32	Bϕ reference	16 bits
Break address mask register A	BAMRA	32	H'FFFFF304		32	W:3	
Break bus cycle register A	BBRA	16	H'FFFFF308		16	L:3	
Break data register A	BDRA	32	H'FFFFF310		32		
Break address register B	BARB	32	H'FFFFF320	UBC	32	Bϕ reference	16 bits
Break address mask register B	BAMRB	32	H'FFFFF324		32	W:3	
Break bus cycle register B	BBRB	16	H'FFFFF328		16	L:3	
Break data register B	BDRB	32	H'FFFFF330		32		

27.2 Register Bit List	1402	Table amended

Register Abbreviation	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0	Module
WTCNT									WDT
WTCSR	TME	WT/IT	RSTS	WOVF	IOVF	CKS[2:0]			
OSCCR	—	—	—	—	—	OSCSTOP	—	OSCERS	CPG

Index

D

E

F

G

H

I

S

T

U

V

W

Renesas 32-Bit RISC Microcomputer
SH7080 Group
User's Manual: Hardware

Publication Date: 1st Edition, March, 2005
Rev.6.00, October 16, 2014
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.
2801 Scott Boulevard Santa Clara, CA 95050-2549, 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: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

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: +852-2265-6688, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

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.
12F., 234 Teheran-ro, Gangnam-Ku, Seoul, 135-920, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 3.0

SH7080 Group
User's Manual: Hardware

Renesas Electronics Corporation

R01UH0198EJ0600