REJ09B0394-0200

Everywhere you imagine. RENESAS

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.

32

SH7101

Hardware Manual

Renesas 32-Bit RISC Microcomputer
SuperH™ RISC engine Family/SH7100 Series

SH7101 HD6437101

Hardware Manual

Rev.2.00
Revision date: Sep. 27, 2007

Renesas Technology
www.renesas.com

Notes regarding these materials

1. This document is provided for reference purposes only so that Renesas customers may select the appropriate Renesas products for their use. Renesas neither makes warranties or representations with respect to the accuracy or completeness of the information contained in this document nor grants any license to any intellectual property rights or any other rights of Renesas or any third party with respect to the information in this document.
2. Renesas shall have no liability for damages or infringement of any intellectual property or other rights arising out of the use of any information in this document, including, but not limited to, product data, diagrams, charts, programs, algorithms, and application circuit examples.
3. You should not use the products or the technology described in this document for the purpose of military applications such as the development of weapons of mass destruction or for the purpose of any other military use. When exporting the products or technology described herein, you should follow the applicable export control laws and regulations, and procedures required by such laws and regulations.
4. All information included in this document such as product data, diagrams, charts, programs, algorithms, and application circuit examples, is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas products listed in this document, please confirm the latest product information with a Renesas sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas such as that disclosed through our website. (http://www.renesas.com)
5. Renesas has used reasonable care in compiling the information included in this document, but Renesas assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the information included in this document.
6. When using or otherwise relying on the information in this document, you should evaluate the information in light of the total system before deciding about the applicability of such information to the intended application. Renesas makes no representations, warranties or guaranties regarding the suitability of its products for any particular application and specifically disclaims any liability arising out of the application and use of the information in this document or Renesas products.
7. With the exception of products specified by Renesas as suitable for automobile applications, Renesas products are not designed, manufactured or tested for applications or otherwise in systems the failure or malfunction of which may cause a direct threat to human life or create a risk of human injury or which require especially high quality and reliability such as safety systems, or equipment or systems for transportation and traffic, healthcare, combustion control, aerospace and aeronautics, nuclear power, or undersea communication transmission. If you are considering the use of our products for such purposes, please contact a Renesas sales office beforehand. Renesas shall have no liability for damages arising out of the uses set forth above.
8. Notwithstanding the preceding paragraph, you should not use Renesas products for the purposes listed below:
   (1) artificial life support devices or systems
   (2) surgical implantations
   (3) healthcare intervention (e.g., excision, administration of medication, etc.)
   (4) any other purposes that pose a direct threat to human life

   Renesas shall have no liability for damages arising out of the uses set forth in the above and purchasers who elect to use Renesas products in any of the foregoing applications shall indemnify and hold harmless Renesas Technology Corp., its affiliated companies and their officers, directors, and employees against any and all damages arising out of such applications.
9. You should use the products described herein within the range specified by Renesas, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas shall have no liability for malfunctions or damages arising out of the use of Renesas products beyond such specified ranges.
10. Although Renesas endeavors to improve the quality and reliability of its products, IC products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Please be sure to implement safety measures to guard against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas 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 applicable measures. Among others, since the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
11. In case Renesas products listed in this document are detached from the products to which the Renesas products are attached or affixed, the risk of accident such as swallowing by infants and small children is very high. You should implement safety measures so that Renesas products may not be easily detached from your products. Renesas shall have no liability for damages arising out of such detachment.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written approval from Renesas.
13. Please contact a Renesas sales office if you have any questions regarding the information contained in this document, Renesas semiconductor products, or if you have any other inquiries.

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

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

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

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

7. List of Registers
8. Electrical Characteristics
9. Appendix
10. Index

Preface

The SH7101 single-chip RISC (Reduced Instruction Set Computer) microcomputer includes a Renesas Technology-original RISC CPU as its core, and the peripheral functions required to configure a system.

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

Objective: This manual was written to explain the hardware functions and electrical characteristics of this LSI to the above users.
Refer to the SH-1/SH-2/SH-DSP Software Manual for a detailed description of the instruction set.

Notes on reading this manual:

- Product names

  The following products are covered in this manual.

Product Classifications and Abbreviations

Basic Classification	On-Chip ROM Classification	Part No.
SH7101 (80-pin version)	Mask ROM version (ROM: 32 kbytes)	HD6437101

  In this manual, the product abbreviations are used to distinguish products. For example, products are collectively referred to as the SH7101.

- 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 the user knows its name

  Read the index that is the final part of the manual to find the page number of the entry on the register. The addresses, bit names, and initial values of the registers are summarized in section 18, List of Registers.

Rules: Register name: The following notation is used for cases when the same or a similar function, e.g. serial communication, is implemented on more than one channel:
XXX_N (XXX is the register name and N is the channel number)

Bit order: The MSB (most significant bit) is on the left and the LSB (least significant bit) is on the right.

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/

SH7101 manuals:

Document Title	Document No.
SH7101 Hardware Manual	This manual
SH-1/SH-2/SH-DSP Software Manual	REJ09B0171-0500

Users manuals for development tools:

Document Title	Document No.
SuperH C/C++ Compiler, Assembler, Optimizing Linkage Editor User's Manual	REJ10B0047-0100
SuperH RISC engine Simulator/Debugger (for Windows) User's Manual	REJ10B0210-0300
High-performance Embedded Workshop User's Manual	REJ10J1554-0100

Application Notes:

Document Title	Document No.
SuperH RISC engine C/C++ Compiler Package Application Note	REJ05B0463-0400

Main Revisions for This Edition

<table>
<tr><th>Item</th><th>Page</th><th>Revision (See Manual for Details)</th></tr>
<tr><td>All</td><td>—</td><td>• Company name and brand names amended
(Before) Hitachi, Ltd. → (After) Renesas Technology Corp.</td></tr>
<tr><td>6.5 Interrupt Exception Processing Vectors Table

Table 6.2 Interrupt Exception Processing Vectors and Priorities</td><td>79</td><td>Table amended
<table>
<tr><th>Interrupt Source</th><th>Name</th></tr>
<tr><td>MTU channel 3</td><td>TGIA_3</td></tr>
<tr><td></td><td>TGIB_3</td></tr>
<tr><td></td><td>TGIC_3</td></tr>
<tr><td></td><td>TGID_3</td></tr>
<tr><td></td><td>TCIV_3</td></tr>
</table></td></tr>
<tr><td>7.5.1 Bus Control Register 1 (BCR1)</td><td>89</td><td>Bit table amended
<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>14</td><td>—</td><td>1</td><td>R</td><td>Reserved
These bits are always read as 1 and should always be written to 1.</td></tr>
</table></td></tr>
<tr><td>8.1 Features

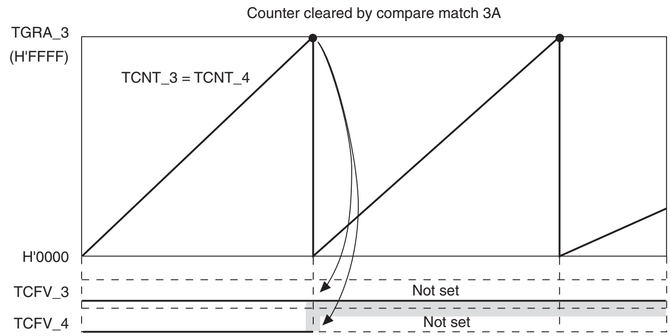
Figure 8.1 Block Diagram of MTU</td><td>94</td><td>Title and figure amended

Interrupt request signals
Channel 3: TGI3A
TGI3B
TGI3C
TGI3D
TCI3V
Channel 4: TGI4A
TGI4B
TGI4C
TGI4D
TCI4V</td></tr>
<tr><td>8.3.3 Timer I/O Control Register (TIOR)

Table 8.13 TIORL_0 (channel 0)</td><td>109</td><td>Table amended
<table>
<tr><th></th><th></th><th></th><th></th><th colspan="2">Description</th></tr>
<tr><th>Bit 3 IOC3</th><th>Bit 2 IOC2</th><th>Bit 1 IOC1</th><th>Bit 0 IOC0</th><th>TGRC_0 Function</th><th>TIOC0C Pin Function</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td rowspan="2">Output compare register*</td><td>Output disable</td></tr>
<tr><td></td><td></td><td></td><td>1</td><td>Initial output is 0
0 output at compare match</td></tr>
</table></td></tr>
<tr><td>8.4.4 Cascaded Operation

Table 8.30 Cascaded Combinations</td><td>149</td><td>Note amended
Note: When phase counting mode is set for channel 1 or 2, the counter clock setting is invalid and the counters operates independently in phase counting mode.</td></tr>
</table>

Item	Page	Revision (See Manual for Details)
8.4.4 Cascaded Operation Figure 8.18 Cascaded Operation Setting Procedure	149	Figure amended [1] Set bits TPSC2 to TPSC0 in the channel 1 TCR to B'111 to select TCNT_2 overflow/ underflow counting.
8.4.8 Complementary PWM Mode Example of Complementary PWM Mode Setting Procedure:	170	Description amended 10. Set enabling/disabling of PWM waveform output pin output in the timer output master enable register (TOER). 11. Set bits CST3 and CST4 in TSTR to 1 simultaneously to start the count operation.
Complementary PWM Mode Output Protection Function:	192	Description amended • 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 bit 13 in the bus controller's bus control register 1 (BCR1). Some registers in channels 3 and 4 concerned are listed below: total 21 registers of 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 the CPU to prevent miswriting due to the CPU runaway by disabling CPU access to the mode registers, control registers, and counters. In access disabled state, an undefined value is read from the registers concerned, and cannot be modified.
8.5.1 Interrupts and Priorities Underflow Interrupt:	194	Description amended 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 MTU has two underflow interrupts, one each for channels 1 and 2.

Item	Page	Revision (See Manual for Details)
8.7.15 Overflow Flags in Reset Sync PWM Mode Figure 8.81 Reset Sync PWM Mode Overflow Flag	216	Figure amended 
8.7.21 Simultaneous Input Capture of TCNT_1 and TCNT_2 in Cascade Connection	219	Description amended 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.

Item	Page	Revision (See Manual for Details)
8.9.5 Usage Note	262	Description added (1) Symptom (a) Regarding the POEnF*1 bits If setting of the POEnF bits in the input level control/status registers (ICSR1 and ICSR2) by the hardware*2 and reading from these bits occur simultaneously, "0" will be read, where "1" should be read. Furthermore, if clearing of these bits is attempted subsequent to the above condition, the clearing should be ignored*3 but it will be carried out. Notes: *1 For the SH7046-Series and SH7047-Series, n = 0 to 6; for the SH7144-Series, n = 0 to 3. *2 The POEnF bits are set when the signals input to the respective POEn pins satisfy the conditions that are specified by the POEnM1 and POEnM0 of the ICSR1 and ICSR2. *3 The correct operation is that clearing of the POEnF bits is only possible after "1" is read from them in order to prevent accidental clearing. (b) Regarding the OSF bit The same symptom applies to the OSF bits of the output level control/status register (OCSR). (2) To Avoid This Problem Please clear the POEnF bits or the OSF bit in these steps: first execute a read for ICSR1, ICSR2, or OCSR, then write "0" to the bits that had a read value of "1" to clear them while writing "1" to other bits. If this procedure is not followed, the POEnF bits and the OSF bit may be cleared unexpectedly if their setting by hardware and reading occur simultaneously.
10.3.2 Receive Data Register (RDR)	280	Description added ... RDR cannot be written to by the CPU. The initial value of RDR is H'00.
10.3.4 Transmit Data Register (TDR)	280	Description added ... Although TDR can be read or written to by the CPU at all times, to achieve reliable serial transmission, write transmit data to TDR for only once after confirming that the TDRE bit in SSR is set to 1. The initial value of TDR is H'FF.

Item	Page	Revision (See Manual for Details)
11.1 Features	325	Description added • Conversion time: 6.7 µs per channel (at Pϕ = 20-MHz operation) 5.4 µs per channel (at Pϕ = 25-MHz operation)
11.7.2 Permissible Signal Source Impedance	340	Description amended 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, or 3 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 or 3 kΩ, charging may be insufficient and it may not be possible to guarantee A/D conversion accuracy. ...
17.3.1 Sleep Mode Notes on Using Sleep Mode	395	Description added • There are 4 conditions to clear sleep mode. (1) Clearing by an interrupt (2) Clearing by DTC address error (3) Clearing by the power-on reset (4) Clearing by the manual reset When clearing sleep mode by (1) or (2), CPU may run out of control. Please clear sleep mode by (3) or (4), don't use (1) or (2). • Do not use DTC module or AUD module during sleep mode.

Contents

Figures

Section 9 Watchdog Timer

Section 10 Serial Communication Interface (SCI)

Section 11 A/D Converter

Tables

Section 1 Overview

The SH7101 single-chip RISC (Reduced Instruction Set Computer) microcomputer integrates a Renesas Technology-original RISC CPU core with peripheral functions required for system configuration.

The SH7101 CPU 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/high-functioning systems, even for applications that were previously impossible with microcomputers, such as real-time control, which demands high speeds.

In addition, the SH7101 includes on-chip peripheral functions necessary for system configuration, such as ROM and RAM, timers, a serial communication interface (SCI), an A/D converter, an interrupt controller (INTC), and I/O ports.

As the on-chip ROM, only mask ROM version is available. However, when F-ZTAT™ (Flexible Zero Turn Around Time) version is required, the SH7046F can be used.

1.1 Features

- 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 four cycles
  - C language-oriented 62 basic instructions
- Various peripheral functions
  - Multifunction timer/pulse unit (MTU)
  - Compare match timer (CMT)
  - Watchdog timer (WDT)
  - Asynchronous or clocked synchronous serial communication interface (SCI)
  - 10-bit A/D converter
  - Clock pulse generator

- On-chip memory

ROM	Model	ROM	RAM	Remarks
Mask ROM Version	HD6437101	32 kbytes	2 kbytes	—

- Maximum operating frequency and operating temperature range

Model	Maximum operating frequency (MHz) (system clock (ϕ) and peripheral clock (Pϕ))	Operating temperature range (°C)
HD6437101F40	(40, 40)	–20 to +75
HD6437101FW40	(40, 40)	–40 to +85

- I/O ports

Model	No. of I/O Pins	No. of Input-only Pins
HD6437101	42	12

- Supports various power-down states
- Compact package

Model	Package	Package (Code)	Body Size	Pin Pitch
HD6437101	QFP-80	FP-80Q	14.0 × 14.0 mm	0.65 mm

1.2 Internal Block Diagram

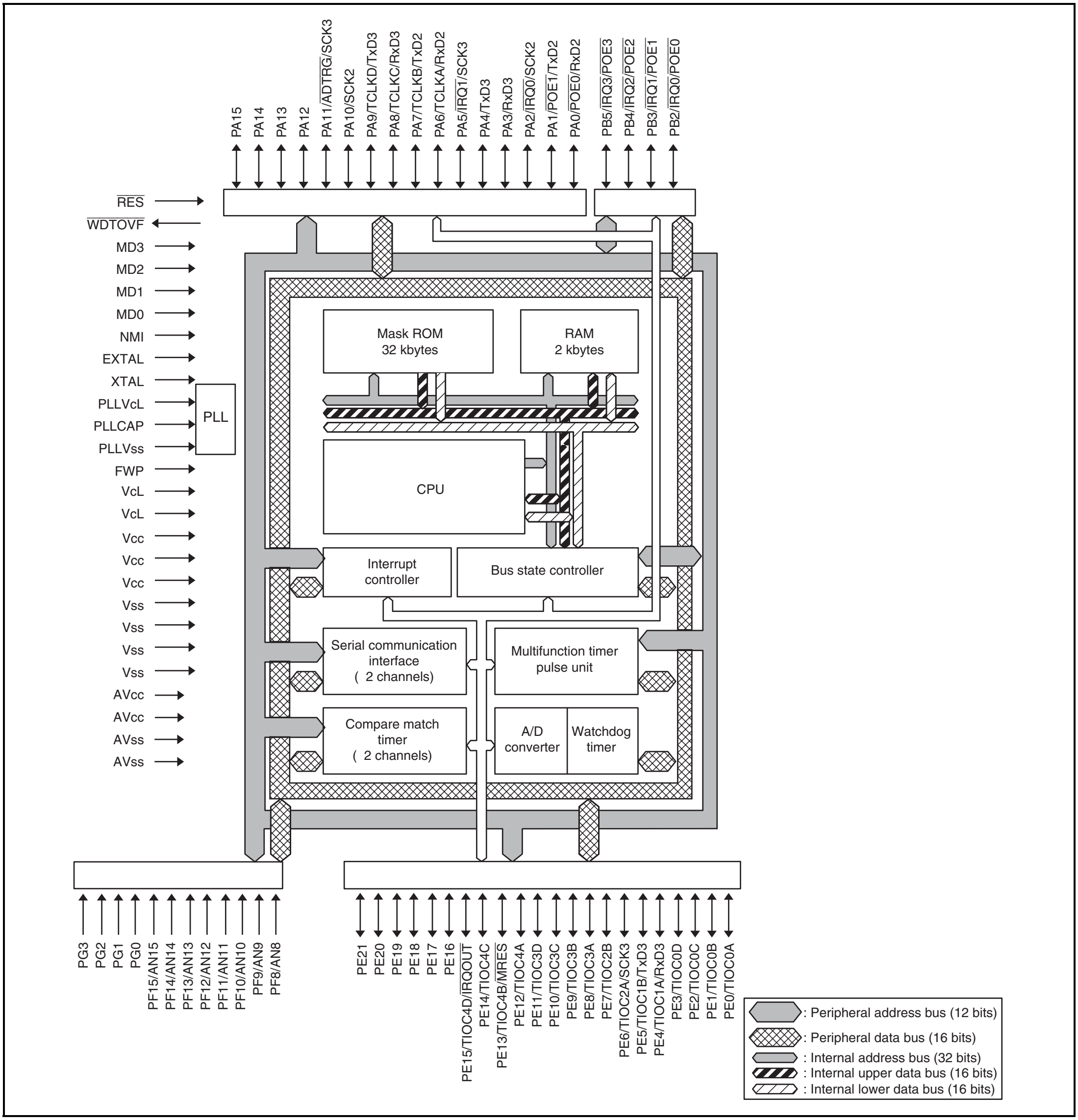
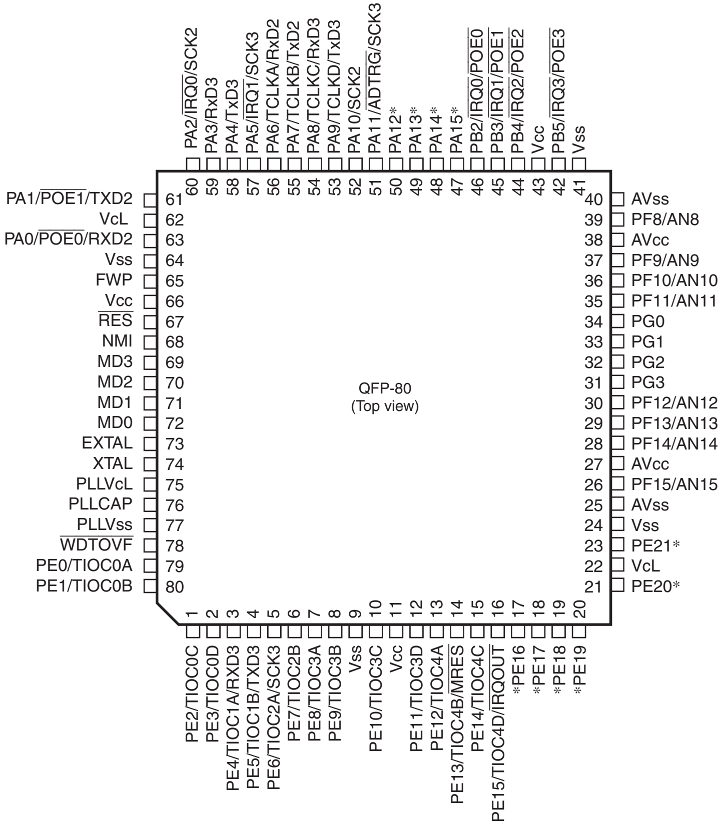


Figure 1.1 Internal Block Diagram of SH7101

1.3 Pin Arrangement



Note: * This pin has a pull-up MOS.

Figure 1.2 SH7101 Pin Arrangement

1.4 Pin Functions

Table 1.1 Pin Functions

Type	Symbol	I/O	Name	Function
Power Supply	VCC	Input	Power supply	Power supply pins. Connect all these pins to the system power supply. The chip does not operate normally when some of these pins are open.
	VSS	Input	Ground	Ground pins. Connect all these pins to the system power supply (0 V). The chip does not operate normally when some of these pins are open.
	VCL	Output	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 (–10%/+100%) capacitor (placed close to the pins).
Clock	PLLVCL	Output	Power supply for PLL	External capacitance pin for internal power-down power supply for an on-chip PLL oscillator. Connect this pin to PLLVSS via a 0.47 μF (–10%/+100%) capacitor (placed close to the pin).
	PLLVSS	Input	Ground for PLL	On-chip PLL oscillator ground pin.
	PLLCAP	Input	Capacitance for PLL	External capacitance pin for an on-chip PLL oscillator.
	EXTAL	Input	External clock	For connection to a crystal resonator. (An external clock can be supplied from the EXTAL pin.) For examples of crystal resonator connection and external clock input, see section 4, Clock Pulse Generator.
	XTAL	Input	Crystal	For connection to a crystal resonator. For examples of crystal resonator connection and external clock input, see section 4, Clock Pulse Generator.
Operating mode control	MD3 to MD0	Input	Set the mode	Set the operating mode. Inputs at these pins should not be changed during operation.

Type	Symbol	I/O	Name	Function
Operating mode control	FWP	Input	Protection against write operation into Flash memory	Pin for the flash memory. This pin is only used in the flash memory version. Writing or erasing of flash memory can be protected. This pin becomes the Vcc pin for the mask ROM version.
System control	$\overline{\text{RES}}$	Input	Power on reset	When this pin is driven low, the chip becomes to power on reset state.
	$\overline{\text{MRES}}$	Input	Manual reset	When this pin is driven low, the chip becomes to manual reset state.
	$\overline{\text{WDTOVF}}$	Output	Watchdog timer overflow	Output signal for the watchdog timer overflow. This pin should be pulled down with at least 1 MΩ resistor value.
Interrupts	NMI	Input	Non-maskable interrupt	Non-maskable interrupt pin. If this pin is not used, it should be fixed high or low.
	$\overline{\text{IRQ3}}$ to $\overline{\text{IRQ0}}$	Input	Interrupt request 3 to 0	These pins request a maskable interrupt. One of the level input or edge input can be selected. In case of the edge input, one of the rising edge, falling edge, or both can be selected.
	$\overline{\text{IRQOUT}}$	Output	Interrupt request output	Shows that an interrupt cause has occurred.
Multi function timer-pulse unit (MTU)	TCLKA TCLKB TCLKC TCLKD	Input	External clock input for MTU timer	These pins input an external clock.
	TIOC0A TIOC0B TIOC0C TIOC0D	Input/ Output	MTU input capture/output compare (channel 0)	The TGRA_0 to TGRD_0 input capture input/output compare output/PWM output pins.
	TIOC1A TIOC1B	Input/ Output	MTU input capture/output compare (channel 1)	The TGRA_1 to TGRB_1 input capture input/output compare output/PWM output pins.
	TIOC2A TIOC2B	Input/ Output	MTU input capture/output compare (channel 2)	The TGRA_2 to TGRB_2 input capture input/output compare output/PWM output pins.

Type	Symbol	I/O	Name	Function
Multi function timer-pulse unit (MTU)	TIOC3A TIOC3B TIOC3C TIOC3D	Input/ Output	MTU input capture/output compare (channel 3)	The TGRA_3 to TGRD_3 input capture input/output compare output/PWM output pins.
	TIOC4A TIOC4B TIOC4C TIOC4D	Input/ Output	MTU input capture/output compare (channel 4)	The TGRA_4 to TGRB_4 input capture input/output compare output/PWM output pins.
Serial com-munication Interface (SCI)	TxD2 TxD3	Output	Transmitted data	Data output pins.
	RxD2 RxD3	Input	Received data	Data input pins.
	SCK2 SCK3	Input/ Output	Serial clock	Clock input/output pins.
MTU output control	$\overline{\text{POE3}}$ to $\overline{\text{POE0}}$	Input	Port output control	Input pins for the signal to request the output pins of MTU waveform to become high impedance state.
A/D converter	AN15 to AN8	Input	Analog input pins	Analog input pins. 8 channels: AN15 to AN8
	$\overline{\text{ADTRG}}$	Input	Input of trigger for A/D conversion	Pin for input of an external trigger to start A/D conversion
	AVCC	Input	Analog power supply	Power supply pin for the A/D converter. When the A/D converter is not used, connect this pin to the system power supply (+5 V). Connect all AVCC pins to the power supply. The chip does not operate normally when some of these pins are open.
	AVSS	Input	Analog ground	The ground pin for the A/D converter. Connect this pin to the system power supply (0 V). Connect all AVSS pins to the system power supply. The chip does not operate normally when some of these pins are open.

Type	Symbol	I/O	Name	Function
I/O ports	PA15 to PA0	Input/ Output	General purpose port	16-bit general purpose input/output port pins. PA15 to PA12 have a pull-up MOS.
	PB5 to PB2	Input/ Output	General purpose port	4-bit general purpose input/output port pins
	PE21 to PE0	Input/ Output	General purpose port	22-bit general purpose input/output port pins. PE21 to PE16 have a pull-up MOS.
	PF15 to PF8	Input	General purpose port	8-bit general purpose input port pins
	PG3 to PG0	Input	General purpose port	4-bit general purpose input port pins

1.5 Differences from SH7046 Group

Table 1.2 Differences from SH7046 Group

Item	SH7046F	SH7101
DTC, MMT, UBC	Incorporated	Not incorporated
INT	Registers IPRA and IPRD to IPRK available.	Registers IPRJ and IPRK deleted.
BSC	Registers BCR1, BCR2, WCR1, and RAMER	Registers BCR2, WCR1, and RAMER deleted.
POE	$\overline{\text{POE6}}$ to $\overline{\text{POE0}}$	$\overline{\text{POE6}}$ to $\overline{\text{POE4}}$ for MMT deleted.
A/D Converter	4ch × 3 modules AN19 to AN8	4ch × 2 modules AN15 to AN8
I/O port PFC	PA2/$\overline{\text{IRQ0}}$/PCIO/SCK2	PA2/$\overline{\text{IRQ0}}$/SCK2
	PA3/$\overline{\text{POE4}}$/RXD3	PA3/RXD3
	PA4/$\overline{\text{POE5}}$/TXD3	PA4/TXD3
	PA5/$\overline{\text{IRQ1}}$/$\overline{\text{POE6}}$/SCK3	PA5/$\overline{\text{IRQ1}}$/SCK3
	PA8/TCLKC/$\overline{\text{IRQ2}}$/RXD3	PA8/TCLKC/RXD3
	PA9/TCLKD/$\overline{\text{IRQ3}}$/TXD3	PA9/TCLKD/TXD3
	PA12/$\overline{\text{UBCTRG}}$	PA12 (pulled up when used as general purpose input)
	PA13/$\overline{\text{POE4}}$	PA13 (pulled up when used as general purpose input)
	PA14/$\overline{\text{POE5}}$	PA14 (pulled up when used as general purpose input)
	PA15/$\overline{\text{POE6}}$	PA15 (pulled up when used as general purpose input)
	PE7/TIOC2B/RXD2	PE7/TIOC2B
	PE8/TIOC3A/SCK2	PE8/TIOC3A
	PE10/TIOC3C/TXD2	PE10/TIOC3C
	PE16/PUOA/$\overline{\text{UBCTRG}}$	PE16 (pulled up when used as general purpose input)*
	PE17/PVOA	PE17 (pulled up when used as general purpose input)*
	PE18/PWOA	PE18 (pulled up when used as general purpose input)*

Item	SH7046F	SH7101
I/O port PFC	PE19/PUOB	PE19 (pulled up when used as general purpose input)*
	PE20/PVOB	PE20 (pulled up when used as general purpose input)*
	PE21/PWOB	PE21 (pulled up when used as general purpose input)*
	PG0/AN16	PG0
	PG1/AN17	PG1
	PG2/AN18	PG2
	PG3/AN19	PG3
ROM	Flash 256 kbytes	Mask ROM 32 kbytes
RAM	12 kbytes	2 kbytes
Operating clock	4 to 50 MHz	10 to 40 MHz

Note: * Pins PE21 to PE16 also functioned as high-current function pins. Howwever, in the SH7101, pins PE21 to PE16 are exclusively for general input/output.

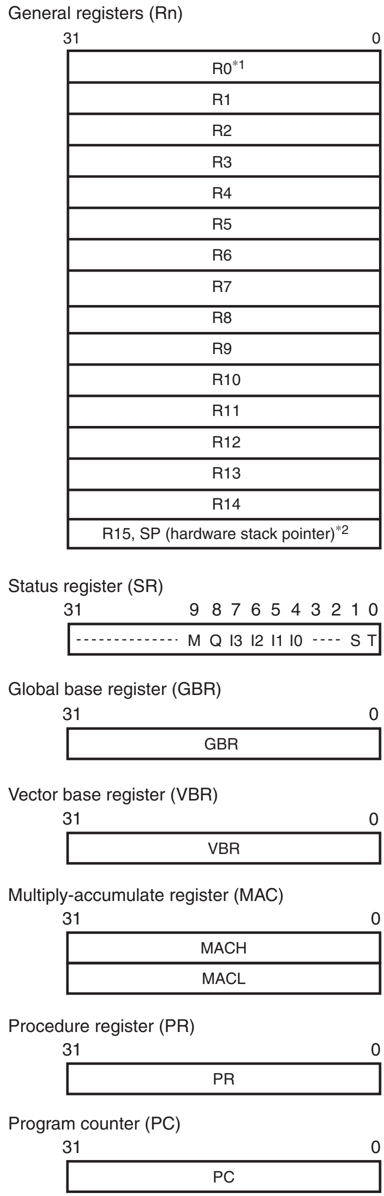
Section 2 CPU

2.1 Features

- General-register architecture
  - Sixteen 32-bit general registers
- Sixty-two basic instructions
- Eleven addressing modes
  - Register direct [Rn]
  - Register indirect [@Rn]
  - Register indirect with post-increment [@Rn+]
  - Register indirect with pre-decrement [@-Rn]
  - Register indirect with displacement [@disp:4,Rn]
  - Register indirect with index [@R0, Rn]
  - GBR indirect with displacement [@disp:8,GBR]
  - GBR indirect with index [@R0,GBR]
  - Program-counter relative with displacement [@disp:8,PC]
  - Program-counter relative [disp:8/disp:12/Rn]
  - Immediate [#imm:8]

2.2 Register Configuration

The register set consists of sixteen 32-bit general registers, three 32-bit control registers, and four 32-bit system registers.



Notes: 1. R0 functions as an index register in the indirect indexed register addressing mode and indirect indexed GBR addressing mode. In some instructions, R0 functions as a fixed source register or destination register.
2. R15 functions as a hardware stack pointer (SP) during exception processing.

Figure 2.1 CPU Internal Registers

2.2.1 General Registers (Rn)

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

2.2.2 Control Registers

The control registers consist of three 32-bit registers: status register (SR), global base register (GBR), and vector base register (VBR). The status register indicates processing states. The global base register functions as a base address for the indirect GBR addressing mode to transfer data to the registers of on-chip peripheral modules. The vector base register functions as the base address of the exception processing vector area (including interrupts).

Status Register (SR):

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R/W	Reserved This bit is 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	I3	1	R/W	Interrupt mask bits.
6	I2	1	R/W	
5	I1	1	R/W	
4	I0	1	R/W	
3, 2	—	All 0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
1	S	Undefined	R/W	S bit Used by the MAC instruction.

Bit	Bit Name	Initial Value	R/W	Description
0	T	Undefined	R/W	T bit The MOVT, CMP/cond, TAS, TST, BT (BT/S), BF (BF/S), SETT, and CLRT instructions use the T bit to indicate true (1) or false (0). The ADDV, ADDC, SUBV, SUBC, DIV0U, DIV0S, DIV1, NEGC, SHAR, SHAL, SHLR, SHLL, ROTR, ROTL, ROTCR, and ROTCL instructions also use the T bit to indicate carry/borrow or overflow/underflow.

Global Base Register (GBR): Indicates the base address of the indirect GBR addressing mode. The indirect GBR addressing mode is used in data transfer for on-chip peripheral modules register areas and in logic operations.

Vector Base Register (VBR): Indicates the base address of the exception processing vector area.

2.2.3 System Registers

System registers consist of four 32-bit registers: high and low multiply and accumulate registers (MACH and MACL), the procedure register (PR), and the program counter (PC).

Multiply-and-Accumulate Registers (MAC): Registers to store the results of multiply-and-accumulate operations.

Procedure Register (PR): Registers to store the return address from a subroutine procedure.

Program Counter (PC): Registers to indicate the sum of current instruction addresses and four, that is, the address of the second instruction after the current instruction.

2.2.4 Initial Values of Registers

Table 2.1 lists the values of the registers after reset.

Table 2.1 Initial Values of Registers

Classification	Register	Initial Value
General registers	R0 to R14	Undefined
	R15 (SP)	Value of the stack pointer in the vector address table
Control registers	SR	Bits I3 to I0 are 1111 (H'F), reserved bits are 0, and other bits are undefined
	GBR	Undefined
	VBR	H'00000000
System registers	MACH, MACL, PR	Undefined
	PC	Value of the program counter in the vector address table

2.3 Data Formats

2.3.1 Data Format in Registers

Register operands are always longwords (32 bits). If the size of memory operand is a byte (8 bits) or a word (16 bits), it is changed into a longword by expanding the sign-part when loaded into a register.

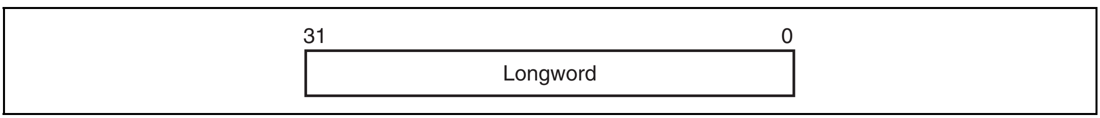


Figure 2.2 Data Format in Registers

2.3.2 Data Formats in Memory

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.

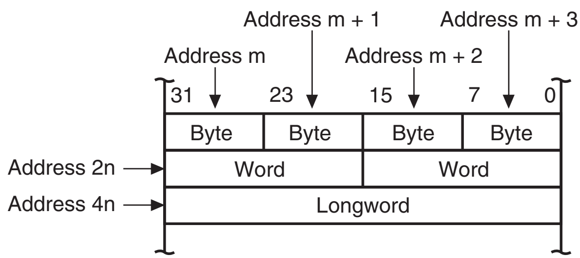


Figure 2.3 Data Formats in Memory

2.3.3 Immediate Data Format

Byte (8 bit) immediate data resides in an instruction code. Immediate data accessed by the MOV, ADD, and CMP/EQ instructions is sign-extended and handled in registers as longword data. Immediate data accessed by the TST, AND, OR, and XOR instructions is zero-extended and handled as longword data. Consequently, AND instructions with immediate data always clear the upper 24 bits of the destination register.

Word or longword immediate data is not located in the instruction code, but instead is stored in a memory table. An immediate data transfer instruction (MOV) accesses the memory table using the PC relative addressing mode with displacement.

2.4 Instruction Features

2.4.1 RISC-Type Instruction Set

All instructions are RISC type. This section details their functions.

16-Bit Fixed Length: All instructions are 16 bits long, increasing program code efficiency.

One Instruction per State: The microprocessor can execute basic instructions in one state using the pipeline system. One state is 25 ns at 40 MHz.

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

Table 2.2 Sign Extension of Word Data

CPU of This LSI	Description	Example of Conventional CPU

`MOV.W @(disp,PC),R1`
`ADD R1,R0`
`.........`
`.DATA.W H'1234`
	Data is sign-extended to 32 bits, and R1 becomes H'00001234. It is next operated upon by an ADD instruction.	`ADD.W #H'1234,R0`

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

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

Delayed Branch Instructions: Unconditional branch instructions are delayed branch instructions. With a delayed branch instruction, the branch is taken after execution of the instruction following the delayed branch instruction. This reduces the disturbance of the pipeline control in case of branch instructions. There are two types of conditional branch instructions: delayed branch instructions and ordinary branch instructions.

Table 2.3 Delayed Branch Instructions

CPU of This LSI	Description	Example of Conventional CPU
`BRA TRGET` `ADD R1,R0`	Executes the ADD before branching to TRGET.	`ADD.W R1,R0` `BRA TRGET`

Multiply/Multiply-and-Accumulate Operations: 16-bit × 16-bit → 32-bit multiply operations are executed in one to two states. 16-bit × 16-bit + 64-bit → 4-bit multiply-and-accumulate operations are executed in two to three states. 32-bit × 32-bit → 64-bit multiply and 32-bit × 32-bit + 64-bit → 64-bit multiply-and-accumulate operations are executed in two to four states.

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

Table 2.4 T Bit

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

Immediate Data: Byte (8-bit) immediate data is located in an instruction code. Word or longword immediate data is not located in instruction codes but in a memory table. An immediate data transfer instruction (MOV) accesses the memory table using the PC relative addressing mode with displacement.

Table 2.5 Immediate Data Accessing

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

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

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

Table 2.6 Absolute Address Accessing

Classification	CPU of This LSI		Example of Conventional CPU
Absolute address	MOV.L	@(disp,PC),R1	MOV.B @H'12345678,R0
	MOV.B	@R1,R0	
		....	
	.DATA.L	H'12345678	

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

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

Table 2.7 Displacement Accessing

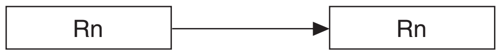
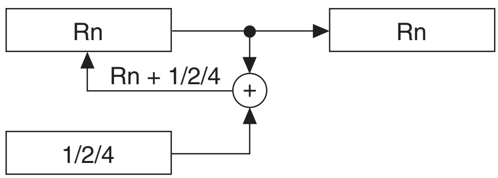
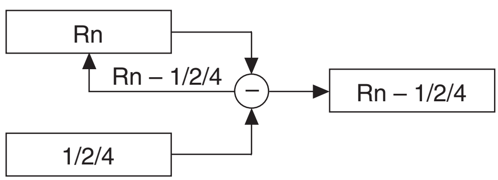
Classification	CPU of This LSI		Example of Conventional CPU
16-bit displacement	MOV.W	@(disp,PC),R0	MOV.W @(H'1234,R1),R2
	MOV.W	@(R0,R1),R2	
		....	
	.DATA.W	H'1234	

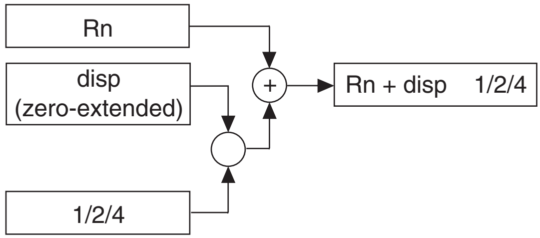
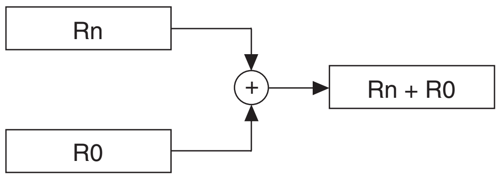
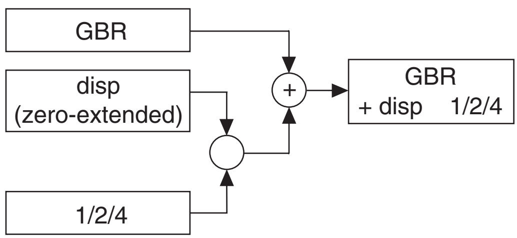
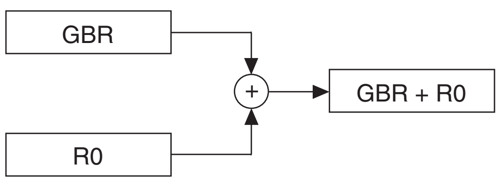
Note: @(disp, PC) accesses the immediate data.

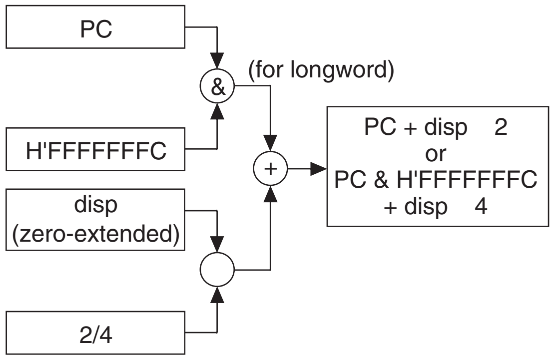
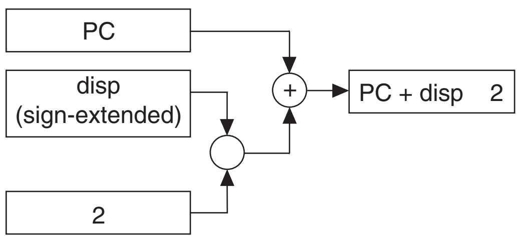
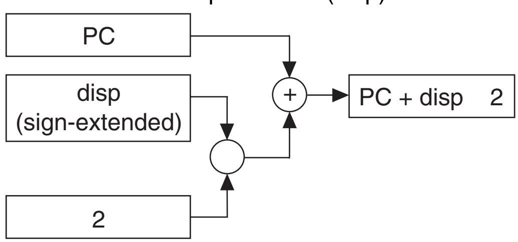
2.4.2 Addressing Modes

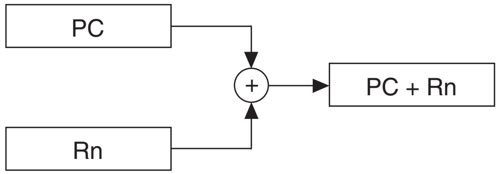
Table 2.8 describes addressing modes and effective address calculation.

Table 2.8 Addressing Modes and Effective Addresses

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

<table>
<tr><th>Addressing Mode</th><th>Instruction Format</th><th>Effective Address Calculation</th><th>Equation</th></tr>
<tr><td>Indirect register addressing with displacement</td><td><code>@(disp:4, Rn)</code></td><td>The effective address is the sum of Rn and a 4-bit displacement (disp). The value of disp is zero-extended, and remains unchanged for a byte operation, is doubled for a word operation, and is quadrupled for a longword operation.

</td><td>Byte:
Rn + disp
Word:
Rn + disp × 2
Longword:
Rn + disp × 4</td></tr>
<tr><td>Indirect indexed register addressing</td><td><code>@(R0, Rn)</code></td><td>The effective address is the sum of Rn and R0.

</td><td>Rn + R0</td></tr>
<tr><td>Indirect GBR addressing with displacement</td><td><code>@(disp:8, GBR)</code></td><td>The effective address is the sum of GBR value and an 8-bit displacement (disp). The value of disp is zero-extended, and remains unchanged for a byte operation, is doubled for a word operation, and is quadrupled for a longword operation.

</td><td>Byte:
GBR + disp
Word:
GBR + disp × 2
Longword:
GBR + disp × 4</td></tr>
<tr><td>Indirect indexed GBR addressing</td><td><code>@(R0, GBR)</code></td><td>The effective address is the sum of GBR value and R0.

</td><td>GBR + R0</td></tr>
</table>

Addressing Mode	Instruction Format	Effective Address Calculation	Equation
Indirect PC addressing with displacement	`@(disp:8, PC)`	The effective address is the sum of PC value and an 8-bit displacement (disp). The value of disp is zero-extended, and is doubled for a word operation, and quadrupled for a longword operation. For a longword operation, the lowest two bits of the PC value are masked.  	Word: PC + disp × 2 Longword: PC & H'FFFFFFFC + disp × 4
PC relative addressing	`disp:8`	The effective address is the sum of PC value and the value that is obtained by doubling the sign-extended 8-bit displacement (disp).  	PC + disp × 2
	`disp:12`	The effective address is the sum of PC value and the value that is obtained by doubling the sign-extended 12-bit displacement (disp).  	PC + disp × 2

Addressing Mode	Instruction Format	Effective Address Calculation	Equation
PC relative addressing	`Rn`	The effective address is the sum of the register PC and Rn. 	PC + Rn
Immediate addressing	`#imm:8`	The 8-bit immediate data (imm) for the TST, AND, OR, and XOR instructions is zero-extended.	—
	`#imm:8`	The 8-bit immediate data (imm) for the MOV, ADD, and CMP/EQ instructions is sign-extended.	—
	`#imm:8`	The 8-bit immediate data (imm) for the TRAPA instruction is zero-extended and then quadrupled.	—

2.4.3 Instruction Format

The instruction formats and the meaning of source and destination operand are described below. The meaning of the operand depends on the instruction code. The symbols used are as follows:

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

Table 2.9 Instruction Formats

Instruction Formats	Source Operand	Destination Operand	Example
0 format 15 (bit) … 0 (bit) xxxx xxxx xxxx xxxx	—	—	`NOP`
n format 15 (bit) … 0 (bit) xxxx | nnnn | xxxx xxxx	—	nnnn: Direct register	`MOVT Rn`
	Control register or system register	nnnn: Direct register	`STS MACH,Rn`
	Control register or system register	nnnn: Indirect pre-decrement register	`STC.L SR,@-Rn`
m format 15 (bit) … 0 (bit) xxxx | mmmm | xxxx xxxx	mmmm: Direct register	Control register or system register	`LDC Rm,SR`
	mmmm: Indirect post-increment register	Control register or system register	`LDC.L @Rm+,SR`
	mmmm: Indirect register	—	`JMP @Rm`
	mmmm: PC relative using Rm	—	`BRAF Rm`

Instruction Formats	Source Operand	Destination Operand	Example
nm format 15 (bit) … 0 (bit) [xxxx | nnnn | mmmm | xxxx]	mmmm: Direct register	nnnn: Direct register	`ADD Rm,Rn`
	mmmm: Direct register	nnnn: Indirect register	`MOV.L Rm,@Rn`
	mmmm: Indirect post-increment register (multiply-and-accumulate) nnnn*: Indirect post-increment register (multiply-and-accumulate)	MACH, MACL	`MAC.W @Rm+,@Rn+`
	mmmm: Indirect post-increment register	nnnn: Direct register	`MOV.L @Rm+,Rn`
	mmmm: Direct register	nnnn: Indirect pre-decrement register	`MOV.L Rm,@-Rn`
	mmmm: Direct register	nnnn: Indirect indexed register	`MOV.L Rm,@(R0,Rn)`
md format 15 (bit) … 0 (bit) [xxxx xxxx | mmmm | dddd]	mmmmdddd: Indirect register with displacement	R0 (Direct register)	`MOV.B @(disp,Rn),R0`
nd4 format 15 (bit) … 0 (bit) [xxxx | xxxx | nnnn | dddd]	R0 (Direct register)	nnnndddd: Indirect register with displacement	`MOV.B R0,@(disp,Rn)`
nmd format 15 (bit) … 0 (bit) [xxxx | nnnn | mmmm | dddd]	mmmm: Direct register	nnnndddd: Indirect register with displacement	`MOV.L Rm,@(disp,Rn)`
	mmmmdddd: Indirect register with displacement	nnnn: Direct register	`MOV.L @(disp,Rm),Rn`

Instruction Formats	Source Operand	Destination Operand	Example
d format 15 … 0 xxxx xxxx | dddd dddd	dddddddd: Indirect GBR with displacement	R0 (Direct register)	`MOV.L @(disp,GBR),R0`
	R0 (Direct register)	dddddddd: Indirect GBR with displacement	`MOV.L R0,@(disp,GBR)`
	dddddddd: PC relative with displacement	R0 (Direct register)	`MOVA @(disp,PC),R0`
	—	dddddddd: PC relative	`BF label`
d12 format 15 … 0 xxxx | dddd dddd dddd	—	dddddddddddd: PC relative	`BRA label` `(label = disp + PC)`
nd8 format 15 … 0 xxxx | nnnn | dddd dddd	dddddddd: PC relative with displacement	nnnn: Direct register	`MOV.L @(disp,PC),Rn`
i format 15 … 0 xxxx xxxx | i i i i i i i i	iiiiiiii: Immediate	Indirect indexed GBR	`AND.B #imm,@(R0,GBR)`
	iiiiiiii: Immediate	R0 (Direct register)	`AND #imm,R0`
	iiiiiiii: Immediate	—	`TRAPA #imm`
ni format 15 … 0 xxxx | nnnn | i i i i i i i i	iiiiiiii: Immediate	nnnn: Direct register	`ADD #imm,Rn`

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

2.5 Instruction Set

2.5.1 Instruction Set by Classification

Table 2.10 lists the instructions according to their classification.

Table 2.10 Classification of Instructions

Classification	Types	Operation Code	Function	No. of Instructions
Data transfer	5	MOV	Data transfer, immediate data transfer, peripheral module data transfer, structure data transfer	39
		MOVA	Effective address transfer	
		MOVT	T bit transfer	
		SWAP	Swap of upper and lower bytes	
		XTRCT	Extraction of the middle of registers connected	
Arithmetic operations	21	ADD	Binary addition	33
		ADDC	Binary addition with carry	
		ADDV	Binary addition with overflow check	
		CMP/cond	Comparison	
		DIV1	Division	
		DIV0S	Initialization of signed division	
		DIV0U	Initialization of unsigned division	
		DMULS	Signed double-length multiplication	
		DMULU	Unsigned double-length multiplication	
		DT	Decrement and test	
		EXTS	Sign extension	
		EXTU	Zero extension	
		MAC	Multiply-and-accumulate, double-length multiply-and-accumulate operation	
		MUL	Double-length multiply operation	
		MULS	Signed multiplication	
		MULU	Unsigned multiplication	
		NEG	Negation	
		NEGC	Negation with borrow	
		SUB	Binary subtraction	
		SUBC	Binary subtraction with borrow	
		SUBV	Binary subtraction with underflow	

Classification	Types	Operation Code	Function	No. of Instructions
Logic operations	6	AND	Logical AND	14
		NOT	Bit inversion	
		OR	Logical OR	
		TAS	Memory test and bit set	
		TST	Logical AND and T bit set	
		XOR	Exclusive OR	
Shift	10	ROTL	One-bit left rotation	14
		ROTR	One-bit right rotation	
		ROTCL	One-bit left rotation with T bit	
		ROTCR	One-bit right rotation with T bit	
		SHAL	One-bit arithmetic left shift	
		SHAR	One-bit arithmetic right shift	
		SHLL	One-bit logical left shift	
		SHLLn	n-bit logical left shift	
		SHLR	One-bit logical right shift	
		SHLRn	n-bit logical right shift	
Branch	9	BF	Conditional branch, conditional branch with delay (Branch when T = 0)	11
		BT	Conditional branch, conditional branch with delay (Branch when 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	

Classification	Types	Operation Code	Function	No. of Instructions
System control	11	CLRT	T bit clear	31
		CLRMAC	MAC register clear	
		LDC	Load to control register	
		LDS	Load to system register	
		NOP	No operation	
		RTE	Return from exception processing	
		SETT	T bit set	
		SLEEP	Transition to power-down mode	
		STC	Store control register data	
		STS	Store system register data	
		TRAPA	Trap exception handling	
Total:	62			142

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

Table 2.11 Symbols Used in Instruction Code, Operation, and Execution States Tables

Item	Format	Explanation
Instruction	Described in mnemonic. `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
Instruction code	Described in MSB ↔ LSB order	mmmm: Source register nnnn: Destination register 0000: R0 0001: R1 . . . 1111: R15 iiii: Immediate data dddd: Displacement
Outline of the Operation	→, ←	Direction of transfer
	(xx)	Memory operand
	M/Q/T	Flag bits in the SR
	&	Logical AND of each bit
			Logical OR of each bit
	^	Exclusive OR of each bit
	~	Logical NOT of each bit
	<<n	n-bit left shift
	>>n	n-bit right shift
Execution states	—	Value when no wait states are inserted*1
T bit	—	Value of T bit after instruction is executed. An em-dash (—) in the column means no change.

Notes: 1. Instruction execution states: The execution states shown in the table are minimums. The actual number of states may be increased when (1) contention occurs between instruction fetches and data access, or (2) when the destination register of the load instruction (memory → register) equals to the register used by the next instruction.

2. Depending on the operand size, displacement is scaled by ×1, ×2, or ×4. For details, refer the *SH-1/SH-2/SH-DSP Software Manual*.

Data Transfer Instructions

Table 2.12 Data Transfer Instructions

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

Instruction	Instruction Code	Operation	Execu-tion States	T Bit
MOV.L @(disp,Rm),Rn	0101nnnnmmmmdddd	(disp × 4 + Rm) → Rn	1	—
MOV.B Rm,@(R0,Rn)	0000nnnnmmmm0100	Rm → (R0 + Rn)	1	—
MOV.W Rm,@(R0,Rn)	0000nnnnmmmm0101	Rm → (R0 + Rn)	1	—
MOV.L Rm,@(R0,Rn)	0000nnnnmmmm0110	Rm → (R0 + Rn)	1	—
MOV.B @(R0,Rm),Rn	0000nnnnmmmm1100	(R0 + Rm) → Sign extension → Rn	1	—
MOV.W @(R0,Rm),Rn	0000nnnnmmmm1101	(R0 + Rm) → Sign extension → Rn	1	—
MOV.L @(R0,Rm),Rn	0000nnnnmmmm1110	(R0 + Rm) → Rn	1	—
MOV.B R0,@(disp,GBR)	11000000dddddddd	R0 → (disp + GBR)	1	—
MOV.W R0,@(disp,GBR)	11000001dddddddd	R0 → (disp × 2 + GBR)	1	—
MOV.L R0,@(disp,GBR)	11000010dddddddd	R0 → (disp × 4 + GBR)	1	—
MOV.B @(disp,GBR),R0	11000100dddddddd	(disp + GBR) → Sign extension → R0	1	—
MOV.W @(disp,GBR),R0	11000101dddddddd	(disp × 2 + GBR) → Sign extension → R0	1	—
MOV.L @(disp,GBR),R0	11000110dddddddd	(disp × 4 + GBR) → R0	1	—
MOVA @(disp,PC),R0	11000111dddddddd	disp × 4 + PC → R0	1	—
MOVT Rn	0000nnnn00101001	T → Rn	1	—
SWAP.B Rm,Rn	0110nnnnmmmm1000	Rm → Swap bottom two bytes → Rn	1	—
SWAP.W Rm,Rn	0110nnnnmmmm1001	Rm → Swap two consecutive words → Rn	1	—
XTRCT Rm,Rn	0010nnnnmmmm1101	Rm: Middle 32 bits of Rn → Rn	1	—

Arithmetic Operation Instructions

Table 2.13 Arithmetic Operation Instructions

Instruction		Instruction Code	Operation	Execu-tion States	T Bit
ADD	Rm,Rn	0011nnnnmmmm1100	Rn + Rm → Rn	1	—
ADD	#imm,Rn	0111nnnniiiiiiii	Rn + imm → Rn	1	—
ADDC	Rm,Rn	0011nnnnmmmm1110	Rn + Rm + T → Rn, Carry → T	1	Carry
ADDV	Rm,Rn	0011nnnnmmmm1111	Rn + Rm → Rn, Overflow → T	1	Overflow
CMP/EQ	#imm,R0	10001000iiiiiiii	If R0 = imm, 1 → T	1	Comparison result
CMP/EQ	Rm,Rn	0011nnnnmmmm0000	If Rn = Rm, 1 → T	1	Comparison result
CMP/HS	Rm,Rn	0011nnnnmmmm0010	If Rn ≥ Rm with unsigned data, 1 → T	1	Comparison result
CMP/GE	Rm,Rn	0011nnnnmmmm0011	If Rn ≥ Rm with signed data, 1 → T	1	Comparison result
CMP/HI	Rm,Rn	0011nnnnmmmm0110	If Rn > Rm with unsigned data, 1 → T	1	Comparison result
CMP/GT	Rm,Rn	0011nnnnmmmm0111	If Rn > Rm with signed data, 1 → T	1	Comparison result
CMP/PL	Rn	0100nnnn00010101	If Rn > 0, 1 → T	1	Comparison result
CMP/PZ	Rn	0100nnnn00010001	If Rn ≥ 0, 1 → T	1	Comparison result
CMP/STR	Rm,Rn	0010nnnnmmmm1100	If Rn and Rm have an equivalent byte, 1 → T	1	Comparison result
DIV1	Rm,Rn	0011nnnnmmmm0100	Single-step division (Rn ÷ Rm)	1	Calculation result
DIV0S	Rm,Rn	0010nnnnmmmm0111	MSB of Rn → Q, MSB of Rm → M, M ^ Q → T	1	Calculation result

Instruction		Instruction Code	Operation	Execution States	T Bit
DIV0U		0000000000011001	0 → M/Q/T	1	0
DMULS.L	Rm,Rn	0011nnnnmmmm1101	Signed operation of Rn × Rm → MACH, MACL 32 × 32 → 64 bits	2 to 4*	—
DMULU.L	Rm,Rn	0011nnnnmmmm0101	Unsigned operation of Rn × Rm → MACH, MACL 32 × 32 → 64 bits	2 to 4*	—
DT	Rn	0100nnnn00010000	Rn – 1 → Rn, when Rn is 0, 1 → T. When Rn is nonzero, 0 → T	1	Comparison result
EXTS.B	Rm,Rn	0110nnnnmmmm1110	Byte in Rm is sign-extended → Rn	1	—
EXTS.W	Rm,Rn	0110nnnnmmmm1111	Word in Rm is sign-extended → Rn	1	—
EXTU.B	Rm,Rn	0110nnnnmmmm1100	Byte in Rm is zero-extended → Rn	1	—
EXTU.W	Rm,Rn	0110nnnnmmmm1101	Word in Rm is zero-extended → Rn	1	—
MAC.L	@Rm+,@Rn+	0000nnnnmmmm1111	Signed operation of (Rn) × (Rm) + MAC → MAC 32 × 32 + 64 → 64 bits	3/ (2 to 4)*	—
MAC.W	@Rm+,@Rn+	0100nnnnmmmm1111	Signed operation of (Rn) × (Rm) + MAC → MAC 16 × 16 + 64 → 64 bits	3/(2)*	—
MUL.L	Rm,Rn	0000nnnnmmmm0111	Rn × Rm → MACL, 32 × 32 → 32 bits	2 to 4*	—
MULS.W	Rm,Rn	0010nnnnmmmm1111	Signed operation of Rn × Rm → MACL 16 × 16 → 32 bits	1 to 3*	—
MULU.W	Rm,Rn	0010nnnnmmmm1110	Unsigned operation of Rn × Rm → MACL 16 × 16 → 32 bits	1 to 3*	—
NEG	Rm,Rn	0110nnnnmmmm1011	0 – Rm → Rn	1	—
NEGC	Rm,Rn	0110nnnnmmmm1010	0 – Rm – T → Rn, Borrow → T	1	Borrow

Instruction		Instruction Code	Operation	Execu-tion States	T Bit
SUB	Rm,Rn	0011nnnnmmmm1000	Rn – Rm → Rn	1	—
SUBC	Rm,Rn	0011nnnnmmmm1010	Rn – Rm – T → Rn, Borrow → T	1	Borrow
SUBV	Rm,Rn	0011nnnnmmmm1011	Rn – Rm → Rn, Underflow → T	1	Overflow

Note: * The normal number of execution states is shown. (The number in parentheses is the number of states when there is contention with the preceding or following instructions.)

Logic Operation Instructions

Table 2.14 Logic Operation Instructions

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

Shift Instructions

Table 2.15 Shift Instructions

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

Branch Instructions

Table 2.16 Branch Instructions

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

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

System Control Instructions

Table 2.17 System Control Instructions

Instruction		Instruction Code	Operation	Execu- tion States	T Bit
CLRT		0000000000001000	0 → T	1	0
CLRMAC		0000000000101000	0 → MACH, MACL	1	—
LDC	Rm,SR	0100mmmm00001110	Rm → SR	1	LSB
LDC	Rm,GBR	0100mmmm00011110	Rm → GBR	1	—
LDC	Rm,VBR	0100mmmm00101110	Rm → VBR	1	—
LDC.L	@Rm+,SR	0100mmmm00000111	(Rm) → SR, Rm + 4 → Rm	3	LSB
LDC.L	@Rm+,GBR	0100mmmm00010111	(Rm) → GBR, Rm + 4 → Rm	3	—
LDC.L	@Rm+,VBR	0100mmmm00100111	(Rm) → VBR, Rm + 4 → Rm	3	—
LDS	Rm,MACH	0100mmmm00001010	Rm → MACH	1	—
LDS	Rm,MACL	0100mmmm00011010	Rm → MACL	1	—
LDS	Rm,PR	0100mmmm00101010	Rm → PR	1	—
LDS.L	@Rm+,MACH	0100mmmm00000110	(Rm) → MACH, Rm + 4 → Rm	1	—
LDS.L	@Rm+,MACL	0100mmmm00010110	(Rm) → MACL, Rm + 4 → Rm	1	—
LDS.L	@Rm+,PR	0100mmmm00100110	(Rm) → PR, Rm + 4 → Rm	1	—
NOP		0000000000001001	No operation	1	—
RTE		0000000000101011	Delayed branch, stack area → PC/SR	4	—
SETT		0000000000011000	1 → T	1	1
SLEEP		0000000000011011	Sleep	3*	—
STC	SR,Rn	0000nnnn00000010	SR → Rn	1	—
STC	GBR,Rn	0000nnnn00010010	GBR → Rn	1	—
STC	VBR,Rn	0000nnnn00100010	VBR → Rn	1	—
STC.L	SR,@-Rn	0100nnnn00000011	Rn – 4 → Rn, SR → (Rn)	2	—
STC.L	GBR,@-Rn	0100nnnn00010011	Rn – 4 → Rn, GBR → (Rn)	2	—
STC.L	VBR,@-Rn	0100nnnn00100011	Rn – 4 → Rn, VBR → (Rn)	2	—

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

Note: * The number of execution states before the chip enters sleep mode: The execution states shown in the table are minimums. The actual number of states may be increased when (1) contention occurs between instruction fetches and data access, or (2) when the destination register of the load instruction (memory → register) equals to the register used by the next instruction.

2.6 Processing States

2.6.1 State Transitions

The CPU has four processing states: reset, exception processing, program execution and power-down. Figure 2.4 shows the transitions between the states.

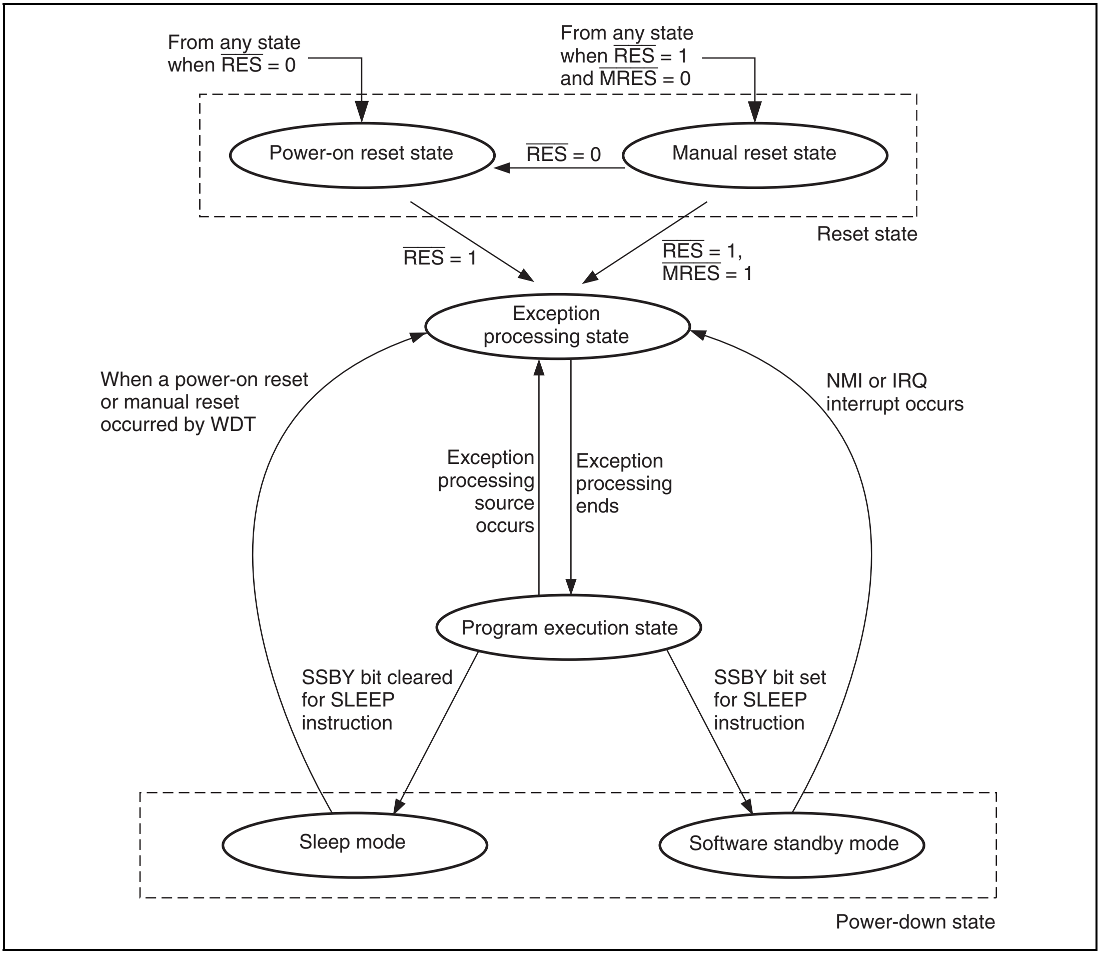


Figure 2.4 Transitions between Processing States

Reset State: The CPU resets in the reset state. When the $\overline{\text{RES}}$ pin level goes low, the power-on reset state is entered. When the $\overline{\text{RES}}$ pin is high and the $\overline{\text{MRES}}$ pin is low, the manual reset state is entered.

Exception Processing State: The exception processing state is a transient state that occurs when exception processing sources such as resets or interrupts alter the CPU's processing state flow.

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

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

Program Execution State: In the program execution state, the CPU sequentially executes the program.

Power-Down State: In the power-down state, the CPU operation halts and power consumption declines. The SLEEP instruction places the CPU in the sleep mode or the software standby mode.

Section 3 MCU Operating Modes

3.1 Selection of Operating Modes

This LSI has one operating mode and four clock modes. The operating mode is determined by the setting of MD3 to MD0, and FWP pins. Do not change these pins during LSI operation (while power is on). Do not set these pins in the other way than the combination shown in table 3.1. This LSI supports only mode 3.

Table 3.1 Selection of Operating Modes

Mode No.	Pin Setting					Mode Name	On-Chip ROM
	FWP	MD3	MD2	MD1	MD0		
Mode 3	1	x	x	1	1	Single chip mode	Active

Note: The symbol x means “Don't care.”

The clock mode is selected by the input of MD2 and MD3 pins.

Table 3.2 Maximum Operating Clock Frequency for Each Clock Mode

Pin Setting		Maximum Operating Clock Frequency
MD3	MD2	
0	0	10 MHz (Input clock × 1, maximum of input clock: 10 MHz)
0	1	20 MHz (Input clock × 2, maximum of input clock: 10 MHz)
1	0	40 MHz (Input clock × 4*, maximum of input clock: 10 MHz)
1	1	40 MHz (Input clock × 4 for system clock, Input clock × 2 for peripheral clock, maximum of input clock: 10 MHz)

Note: * The maximum of input clock is 10 MHz so that Pφ is lower or equal to 40 MHz.

3.2 Input/Output Pins

Table 3.3 describes the configuration of operating mode related pins.

Table 3.3 Operating Mode 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
MD2	Input	Designates clock mode through the level applied to this pin
MD3	Input	Designates clock mode through the level applied to this pin
FWP	Input	Pin for the hardware protection against writing/erasing the on-chip flash memory. In this LSI, conncet this pin to V_{CC}.

3.3 Explanation of Operating Modes

This LSI does not support modes 0 to 2 (MCU extension mode 0 to 2).

3.3.1 Mode 3 (Single chip mode)

All ports can be used in this mode, however the external address cannot be used. The SH7101 supports only this mode.

3.3.2 Clock Mode

The input waveform frequency can be used as is, doubled or quadrupled as system clock frequency.

3.4 Address Map

Figure 3.1 shows the address map.

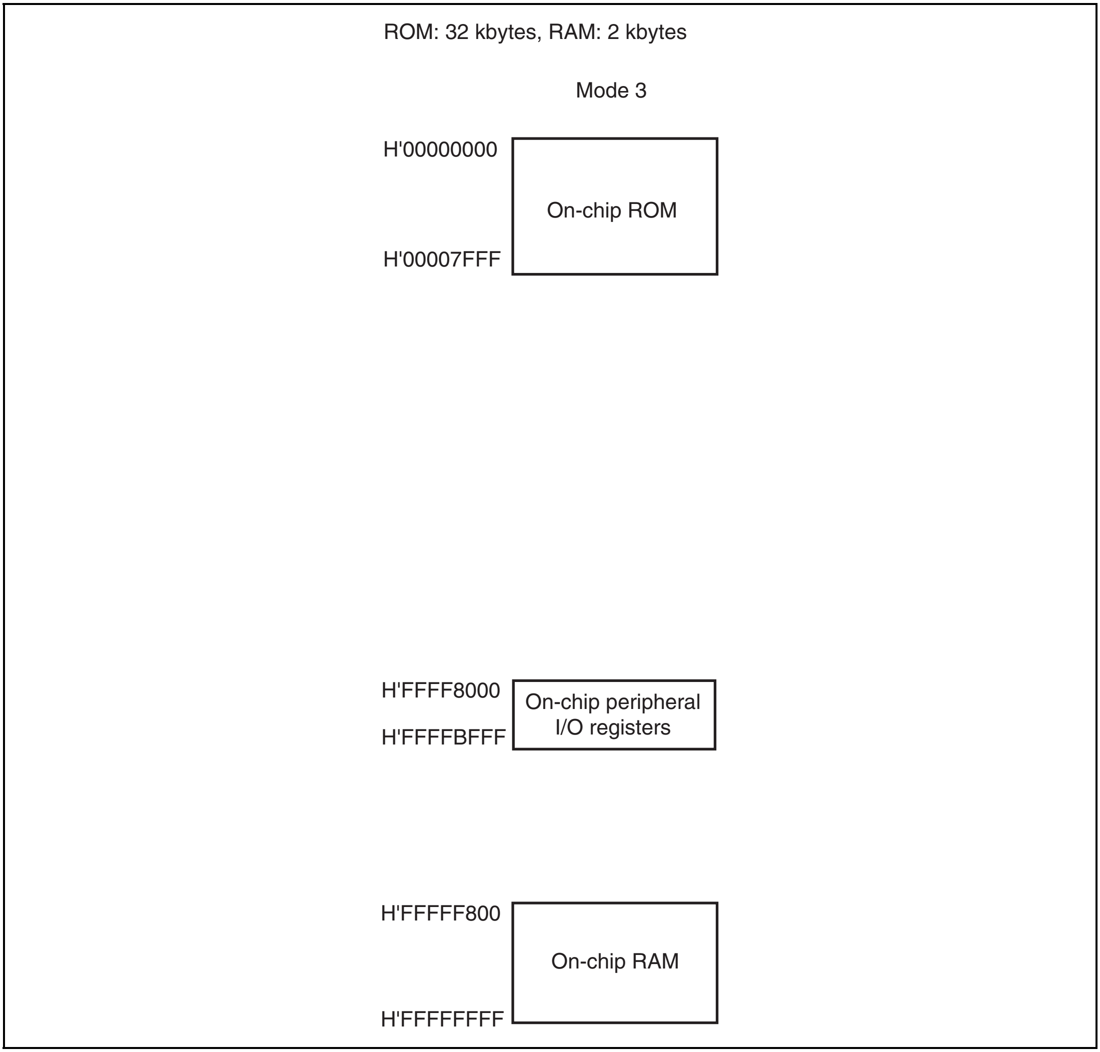


Figure 3.1 Address Map for SH7101 Mask ROM Version

3.5 Initial State of This LSI

To reduce power consumption, some modules are set to the module standby states in the initial state in this LSI. Therefore, the module standby states should be cancelled to activate these modules. For details, see section 17, Power-Down Modes.

Section 4 Clock Pulse Generator

This LSI has an on-chip clock pulse generator (CPG) that generates the system clock (ϕ), internal clock (ϕ/2 to ϕ/8192, Pϕ/2 to Pϕ/1024), and peripheral clock (Pϕ). The CPG consists of an oscillator, PLL circuit, and prescaler. A block diagram of the clock pulse generator is shown in figure 4.1. The frequency from the oscillator can be modified by the PLL circuit.

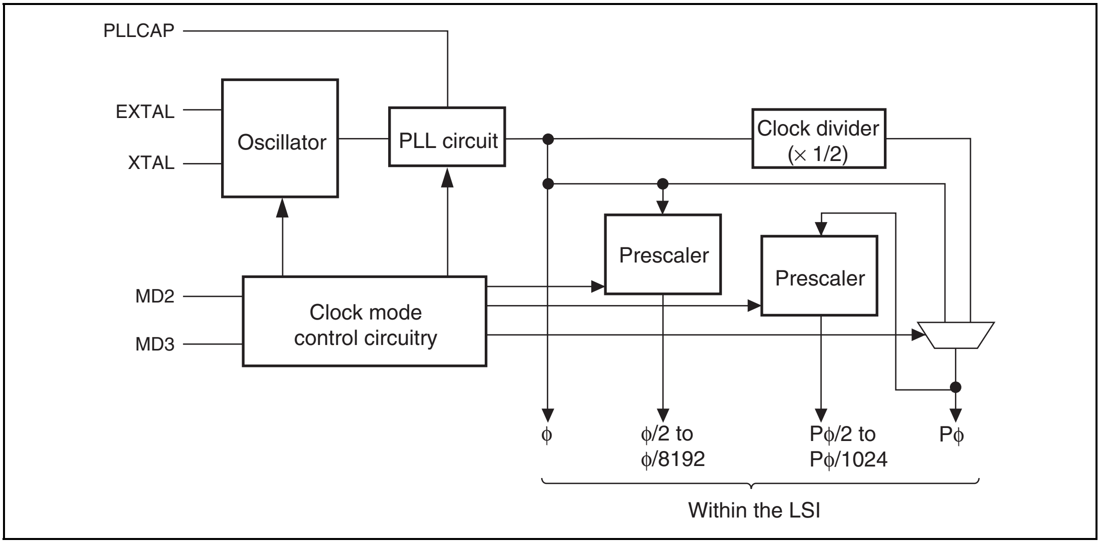


Figure 4.1 Block Diagram of Clock Pulse Generator

4.1 Oscillator

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

4.1.1 Connecting Crystal Resonator

Circuit Configuration: A crystal resonator can be connected as shown in figure 4.2. Use the damping resistance (Rd) listed in table 4.1. Use an AT-cut parallel-resonance type crystal resonator that has a resonance frequency of 4 to 10 MHz. It is recommended to consult crystal dealer concerning the compatibility of the crystal resonator and the LSI.

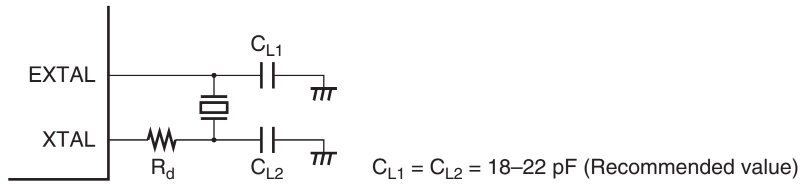


Figure 4.2 Connection of Crystal Resonator (Example)

Table 4.1 Damping Resistance Values

Frequency (MHz)	4	8	10
Rd (Ω)	500	200	0

Crystal Resonator: Figure 4.3 shows an equivalent circuit of the crystal resonator. Use a crystal resonator with the characteristics listed in table 4.2.

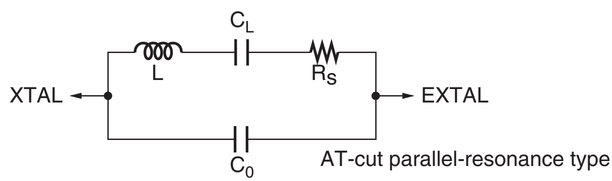


Figure 4.3 Crystal Resonator Equivalent Circuit

Table 4.2 Crystal Resonator Characteristics

Frequency (MHz)	4	8	10
Rs max (Ω)	120	80	60
C0 max (pF)	7	7	7

4.1.2 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 in standby mode. During operation, make the external input clock frequency 4 to 10 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 standby mode, in order to ensure the PLL stabilization time.

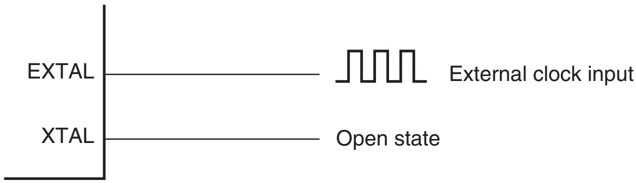


Figure 4.4 Example of External Clock Connection

4.2 Function for Detecting Oscillator Halt

This CPG can detect a clock halt and automatically cause the timer pins to become high-impedance when any system abnormality causes the oscillator to halt. That is, when a change of EXTAL has not been detected, the high-current six pins (PE9/TIOC3B, PE11/TIOC3D, PE12/TIOC4A, PE13/TIOC4B/$\overline{\text{MRES}}$, PE14/TIOC4C, and PE15/TIOC4D/$\overline{\text{IRQOUT}}$) are set to high-impedance regardless of PFC setting.

Even in standby mode, these six pins become high-impedance regardless of PFC setting. These pins enter the normal state after standby mode is released. When abnormalities that halt the oscillator occur except in standby mode, other LSI operations become undefined. In this case, LSI operations, including these six pins, become undefined even when the oscillator operation starts again.

4.3 Usage Notes

4.3.1 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 resonator circuit ratings will depend on the resonator and the floating capacitance of the mounting circuit, the ratings should be determined in consultation with the resonator manufacturer. Ensure that a voltage exceeding the maximum rating is not applied to the oscillator pin.

4.3.2 Notes on Board Design

Measures against radiation noise are taken in this LSI. If radiation noise needs to be further reduced, usage of a multi-layer printed circuit board with ground planes is recommended.

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.

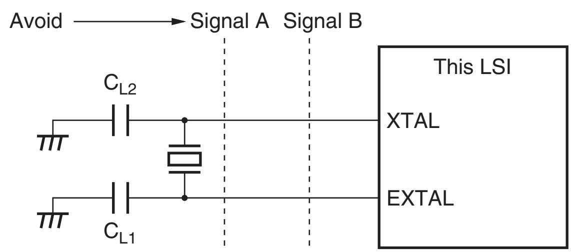
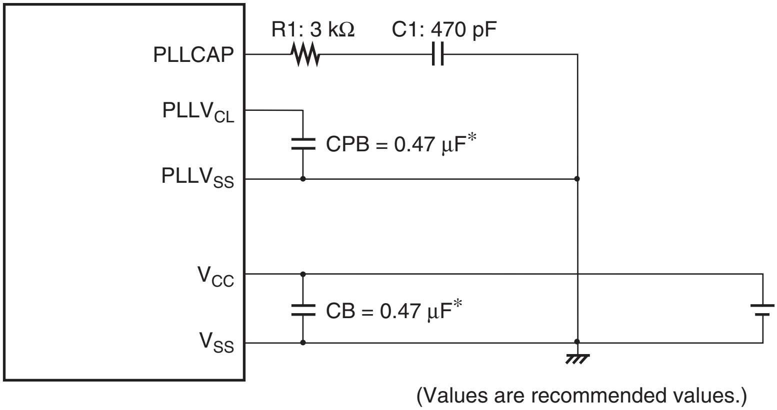


Figure 4.5 Cautions for Oscillator Circuit System Board Design

A circuitry shown in figure 4.6 is recommended as an external circuitry around the PLL. Place oscillation stabilization capacitor C1 close to the PLLCAP pin, and ensure that no other signal lines cross this line. Separate $PLLV_{CL}$, $PLLV_{SS}$, V_{CC}, and V_{SS} from the board power supply source, and be sure to insert bypass capacitors CB close to the pins.



Note: * CB and CPB are laminated ceramic type.

Figure 4.6 Recommended External Circuitry around PLL

Electromagnetic waves are radiated from an LSI in operation. This LSI has an electromagnetic peak in the harmonics band whose primary frequency is determined by the lower frequency between the system clock (ϕ) and peripheral clock (Pϕ). For example, when ϕ = 40 MHz and Pϕ = 40 MHz, the primary frequency is 40 MHz. If this LSI is used adjacent to a device sensitive to electromagnetic interference, e.g. FM/VHF band receiver, a printed circuit board of more than four layers with planes exclusively for system ground is recommended.

Section 5 Exception Processing

5.1 Overview

5.1.1 Types of Exception Processing and Priority

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

Table 5.1 Types of Exception Processing and Priority

Exception	Source		Priority
Reset	Power-on reset		High ↑
	Manual reset		
Address error	CPU address error		
Interrupt	NMI		
	IRQ		
	On-chip peripheral modules:	• Multifunction timer unit (MTU) • A/D converter 0 and 1 (A/D0, A/D1) • Compare match timer 0 and 1 (CMT0, CMT1) • Watchdog timer (WDT) • Input/output port (I/O) (MTU) • Serial communication interface 2 and 3 (SCI2, SCI3)	
Instructions	Trap instruction (TRAPA instruction)		
	General illegal instructions (undefined code)		
	Illegal slot instructions (undefined code placed directly after a delayed branch instruction*1 or instructions that rewrite the PC*2)		↓ Low

Notes: 1. Delayed branch instructions: JMP, JSR, BRA, BSR, RTS, RTE, BF/S, BT/S, BSRF, and BRAF.

2. Instructions that rewrite the PC: JMP, JSR, BRA, BSR, RTS, RTE, BT, BF, TRAPA, BF/S, BT/S, BSRF, and BRAF.

5.1.2 Exception Processing Operations

The exception processing sources are detected and the processing starts according to the timing shown in table 5.2.

Table 5.2 Timing for Exception Source Detection and Start of Exception Processing

Exception	Source	Timing of Source Detection and Start of Processing
Reset	Power-on reset	Starts when the $\overline{\text{RES}}$ pin changes from low to high or when WDT overflows.
	Manual reset	Starts when the $\overline{\text{MRES}}$ pin changes from low to high.
Address error		Detected when instruction is decoded and starts when the execution of the previous instruction is completed.
Interrupts		
Instructions	Trap instruction	Starts from the execution of a TRAPA instruction.
	General illegal instructions	Starts from the decoding of undefined code anytime except after a delayed branch instruction (delay slot).
	Illegal slot instructions	Starts from the decoding of undefined code placed in a delayed branch instruction (delay slot) or of instructions that rewrite the PC.

When exception processing starts, the CPU operates as follows:

1. Exception processing triggered by reset:

   The initial values of the program counter (PC) and stack pointer (SP) are fetched from the exception processing vector table (PC and SP are respectively the H'00000000 and H'00000004 addresses for power-on resets and the H'00000008 and H'0000000C addresses for manual resets). See section 5.1.3, Exception Processing Vector Table, for more information. 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) of the status register (SR). The program begins running from the PC address fetched from the exception processing vector table.
2. Exception processing triggered by address errors, interrupts and instructions:

   SR and PC are saved to the stack indicated by R15. For interrupt exception processing, the interrupt priority level is written to the SR's interrupt mask bits (I3 to I0). For address error and instruction exception processing, the I3 to I0 bits are not affected. The start address is then fetched from the exception processing vector table and the program begins running from that address.

5.1.3 Exception Processing Vector Table

Before exception processing begins running, the exception processing vector table must be set in memory. The exception processing vector table stores the start addresses of exception service routines. (The reset exception processing 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 processing, the start addresses of the exception service routines are fetched from the exception processing 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 Exception Processing Vector Table

Exception Sources		Vector Numbers	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 by system)		5	H'00000014 to H'00000017
Slot illegal instruction		6	H'00000018 to H'0000001B
(Reserved by system)		7	H'0000001C to H'0000001F
		8	H'00000020 to H'00000023
CPU address error		9	H'00000024 to H'00000027
(Reserved by system)		10	H'00000028 to H'0000002B
Interrupts	NMI	11	H'0000002C to H'0000002F
	(Reserved by system)	12	H'00000030 to H'00000033
(Reserved by system)		13	H'00000034 to H'00000037
(Reserved by system)		14	H'00000038 to H'0000003B
(Reserved by system)		15	H'0000003C to H'0000003F
		:	:
		31	H'0000007C to H'0000007F

Exception Sources		Vector Numbers	Vector Table Address Offset
Trap instruction (user vector)		32	H'00000080 to H'00000083
		:	:
		63	H'000000FC to H'000000FF
Interrupts	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
	Reserved by system	68	H'00000110 to H'00000113
	Reserved by system	69	H'00000114 to H'00000117
	Reserved by system	70	H'00000118 to H'0000011B
	Reserved by system	71	H'0000011C to H'0000011F
On-chip peripheral module *		72	H'00000120 to H'00000124
		:	:
		255	H'000003FC to H'000003FF

Note: * The vector numbers and vector table address offsets for each on-chip peripheral module interrupt are given in section 6, Interrupt Controller (INTC), and table 6.2, Interrupt Exception Processing Vectors and Priorities.

Table 5.4 Calculating Exception Processing 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 Reset

Resets have the highest priority of any exception source. There are two types of resets: manual resets and power-on 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 Status			Internal Status		
Type	**$\overline{\text{RES}}$**	**WDT Overflow**	**$\overline{\text{MRES}}$**	**CPU/INTC**	**On-Chip Peripheral Module**	**PFC, IO Port**
Power-on reset	Low	—	—	Initialized	Initialized	Initialized
	High	Overflow	High	Initialized	Initialized	Not initialized
Manual reset	High	—	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, the LSI becomes to be a power-on reset state. To reliably reset the LSI, the $\overline{\text{RES}}$ pin should be kept at low for at least the duration of the oscillation settling time when applying power or when in standby mode (when the clock circuit is halted) or at least 25 t_{cyc} when the clock circuit is running. During power-on reset, CPU internal status and all registers of on-chip peripheral modules are initialized. See appendix A, Pin States, for the status of individual pins during the power-on reset status.

In the power-on reset status, power-on reset exception processing starts when the $\overline{\text{RES}}$ pin is first driven low for a set period of time and then returned to high. The CPU will then operate as follows:

1. The initial value (execution start address) of the program counter (PC) is fetched from the exception processing vector table.
2. The initial value of the stack pointer (SP) is fetched from the exception processing 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 processing vector table are set in PC and SP, then the program begins executing.

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

Power-On Reset by WDT: When a setting is made for a power-on reset to be generated in the WDT's watchdog timer mode, and the WDT's TCNT overflows, the LSI becomes to be a power-on reset state.

The pin function controller (PFC) registers and I/O port registers are not initialized by the reset signal generated by the WDT (these registers are initialized only by a power-on reset from outside of the chip).

If reset caused by the input signal at the $\overline{\text{RES}}$ pin and a reset caused by WDT overflow occur simultaneously, the $\overline{\text{RES}}$ pin reset has priority, and the WOVF bit in RSTCSR is cleared to 0. When WDT-initiated power-on reset processing is started, the CPU operates as follows:

1. The initial value (execution start address) of the program counter (PC) is fetched from the exception processing vector table.
2. The initial value of the stack pointer (SP) is fetched from the exception processing vector table.
3. The vector base register (VBR) is cleared to H'00000000 and the interrupt mask bits (I3-I0) of the status register (SR) are set to H'F (B'1111).
4. The values fetched from the exception processing vector table are set in the PC and SP, then the program begins executing.

5.2.3 Manual Reset

When the $\overline{\text{RES}}$ pin is high and the $\overline{\text{MRES}}$ pin is driven low, the LSI enters 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 in standby mode (when the clock is halted) or at least 25 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 Cause of Address Error Exception

Address errors occur when instructions are fetched or data is read or written, 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 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 when in single chip mode	Address error occurs
Data read/write	CPU	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*	Address error occurs
		External memory space accessed when in single chip mode	Address error occurs

Note: * See section 7, Bus State Controller (BSC) for more information on the on-chip peripheral module space.

5.3.2 Address Error Exception Processing

When an address error occurs, the bus cycle in which the address error occurred ends, the current instruction finishes, and then address error exception processing 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 last executed instruction.
3. The start address of the exception service routine is fetched from the exception processing vector table that corresponds to the occurred address error, and the program starts executing from that address. The jump in this case is not a delayed branch.

5.4 Interrupts

5.4.1 Interrupt Sources

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

Table 5.7 Interrupt Sources

Type	Request Source	Number of Sources
NMI	NMI pin (external input)	1
IRQ	$\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$ pins (external input)	4
On-chip peripheral module	Multifunction timer unit	23
	Compare match timer	2
	A/D converter (A/D0 and A/D1)	2
	Serial communication interface	8
	Watchdog timer	1
	Input/output port	1

Each interrupt source is allocated a different vector number and vector table offset. See section 6, Interrupt Controller (INTC), and table 6.2, Interrupt Exception Processing Vectors and Priorities, for more information on vector numbers and vector table address offsets.

5.4.2 Interrupt Priority Level

The interrupt priority is predetermined. When multiple interrupts occur simultaneously (overlapped interruptions), the interrupt controller (INTC) determines their relative priorities and starts the exception processing 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. IRQ interrupts and on-chip peripheral module interrupt priority levels can be set freely using the INTC's interrupt priority registers A, D to I (IPRA, IPRD to IPRI) as shown in table 5.8. The priority levels that can be set are 0 to 15. Level 16 cannot be set. See section 6.3.4, Interrupt Priority Registers A, D to I (IPRA, IPRD to IPRI), for more information on IPRA, IPRD to IPRI.

Table 5.8 Interrupt Priority

Type	Priority Level	Comment
NMI	16	Fixed priority level. Cannot be masked.
IRQ	0 to 15	Set with interrupt priority registers A, D to I (IPRA, IPRD to IPRI).
On-chip peripheral module		

5.4.3 Interrupt Exception Processing

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 processing begins. In interrupt exception processing, the CPU saves SR and the program counter (PC) to the stack. The priority level value of the accepted interrupt is written to SR bits I3 to I0. For NMI, however, the priority level is 16, but the value set in I3 to I0 is H'F (level 15). Next, the start address of the exception service routine is fetched from the exception processing vector table for the accepted interrupt, that address is jumped to and execution begins. See section 6.6, Operation, for more information on the interrupt exception processing.

5.5 Exceptions Triggered by Instructions

5.5.1 Types of Exceptions Triggered by Instructions

Exception processing can be triggered by 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 rewrite the PC	Delayed branch instructions: JMP, JSR, BRA, BSR, RTS, RTE, BF/S, BT/S, BSRF, BRAF Instructions that rewrite the PC: JMP, JSR, BRA, BSR, RTS, RTE, BT, BF, TRAPA, BF/S, BT/S, BSRF, BRAF
General illegal instructions	Undefined code anywhere besides in a delay slot	—

5.5.2 Trap Instructions

When a TRAPA instruction is executed, trap instruction exception processing 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 service routine from the exception processing vector table that corresponds to the vector number specified in the TRAPA instruction, jumps to that address and starts executing the program. This jump 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 processing starts after the undefined code is decoded. Illegal slot exception processing also starts when an instruction that rewrites the program counter (PC) 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 service routine is fetched from the exception processing vector table that corresponds to the exception that occurred. That address is jumped to and the program starts executing. The jump in this case is not a delayed branch.

5.5.4 General Illegal Instructions

When undefined code placed anywhere other than immediately after a delayed branch instruction (i.e., in a delay slot) is decoded, general illegal instruction exception processing 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 Exception Sources are Not Accepted

When an address error or interrupt is generated directly after a delayed branch instruction or interrupt-disabled instruction, it is sometimes not accepted immediately but stored instead, as shown in table 5.10. In this case, it will be accepted when an instruction that can accept the exception is decoded.

Table 5.10 Generation of Exception Sources Immediately after Delayed Branch Instruction or Interrupt-Disabled Instruction

	Exception Source	
Point of Occurrence	**Address Error**	**Interrupt**
Immediately after a delayed branch instruction*[1]	Not accepted	Not accepted
Immediately after an interrupt-disabled instruction*[2]	Accepted	Not accepted

Notes: 1. Delayed branch instructions: JMP, JSR, BRA, BSR, RTS, RTE, BF/S, BT/S, BSRF, and BRAF

2. Interrupt-disabled instructions: LDC, LDC.L, STC, STC.L, LDS, LDS.L, STS, and STS.L

5.6.1 Immediately after Delayed Branch Instruction

When an instruction placed immediately after a delayed branch instruction (delay slot) is decoded, neither address errors nor interrupts are accepted. The delayed branch instruction and the instruction placed immediately after it (delay slot) are always executed consecutively, so no exception processing occurs during this period.

5.6.2 Immediately after Interrupt-Disabled Instruction

When an instruction placed immediately after an interrupt-disabled instruction is decoded, interrupts are not accepted. Address errors can be accepted.

5.7 Stack Status after Exception Processing Ends

The status of the stack after exception processing ends is shown in table 5.11.

Table 5.11 Stack Status after Exception Processing Ends

Types	Stack Status
Address error	SP → Address of instruction after executed instruction (32 bits) SR (32 bits)
Trap instruction	SP → Address of instruction after TRAPA instruction (32 bits) SR (32 bits)
General illegal instruction	SP → Address of instruction after general illegal instruction (32 bits) SR (32 bits)
Interrupt	SP → Address of instruction after executed instruction (32 bits) SR (32 bits)
Illegal slot instruction	SP → Jump destination address of delay branch instruction (32 bits) SR (32 bits)

5.8 Usage Notes

5.8.1 Value of Stack Pointer (SP)

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

5.8.2 Value of Vector Base Register (VBR)

The value of the vector base register must always be a multiple of four. If it is not, an address error will occur when the stack is accessed during exception processing.

5.8.3 Address Errors Caused by Stacking of Address Error Exception Processing

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

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

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 canceler function
- Occurrence of interrupt can be reported externally ($\overline{\text{IRQOUT}}$ pin)

Figure 6.1 shows a block diagram of the INTC.

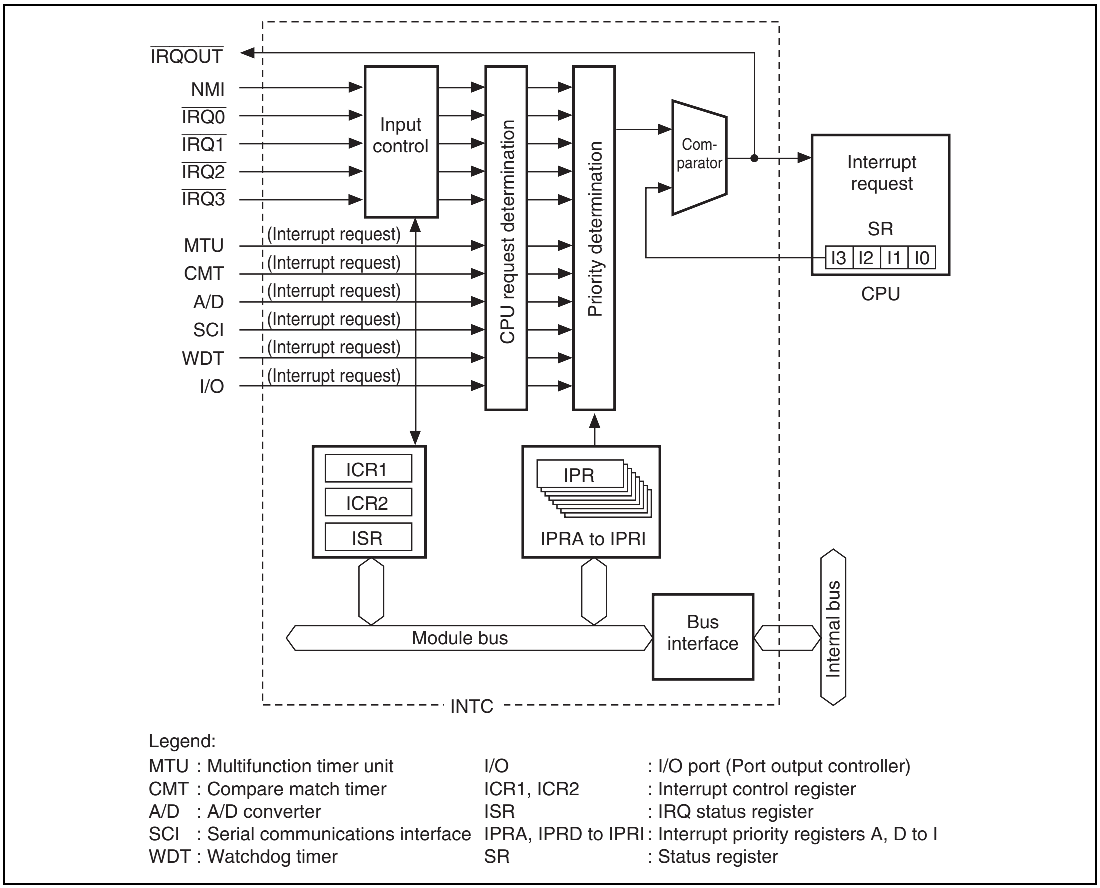


Figure 6.1 INTC Block Diagram

6.2 Input/Output Pins

Table 6.1 shows the INTC pin configuration.

Table 6.1 Pin Configuration

Name	Abbreviation	I/O	Function
Non-maskable interrupt input pin	NMI	I	Input of non-maskable interrupt request signal
Interrupt request input pins	$\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$	I	Input of maskable interrupt request signals
Interrupt request output pin	$\overline{\text{IRQOUT}}$	O	Output of notification signal when an interrupt has occurred

6.3 Register Descriptions

The interrupt controller has the following registers. For details on register addresses and register states during each processing, refer to section 18, List of Registers.

- Interrupt control register 1 (ICR1)
- Interrupt control register 2 (ICR2)
- IRQ status register (ISR)
- Interrupt priority register A (IPRA)
- Interrupt priority register D (IPRD)
- Interrupt priority register E (IPRE)
- Interrupt priority register F (IPRF)
- Interrupt priority register G (IPRG)
- Interrupt priority register H (IPRH)
- Interrupt priority register I (IPRI)

6.3.1 Interrupt Control Register 1 (ICR1)

ICR1 is a 16-bit register that sets the input signal detection mode of the external interrupt input pins NMI and $\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$ and indicates the input signal level at the NMI pin.

Bit	Bit Name	Initial Value	R/W	Description
15	NMIL	1/0	R	NMI Input Level Sets the level 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: NMI input level is low 1: NMI input level 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 falling edge of NMI input (Initial value) 1: Interrupt request is detected on rising edge of NMI input
7	IRQ0S	0	R/W	IRQ0 Sense Select This bit sets the IRQ0 interrupt request detection mode. 0: Interrupt request is detected on low level of IRQ0 input 1: Interrupt request is detected on edge of IRQ0 input (edge direction is selected by ICR2)
6	IRQ1S	0	R/W	IRQ1 Sense Select This bit sets the IRQ1 interrupt request detection mode. 0: Interrupt request is detected on low level of IRQ1 input 1: Interrupt request is detected on edge of IRQ1 input (edge direction is selected by ICR2)
5	IRQ2S	0	R/W	IRQ2 Sense Select This bit sets the IRQ2 interrupt request detection mode. 0: Interrupt request is detected on low level of IRQ2 input 1: Interrupt request is detected on edge of IRQ2 input (edge direction is selected by ICR2)

Bit	Bit Name	Initial Value	R/W	Description
4	IRQ3S	0	R/W	IRQ3 Sense Select This bit sets the IRQ3 interrupt request detection mode. 0: Interrupt request is detected on low level of IRQ3 input 1: Interrupt request is detected on edge of IRQ3 input (edge direction is selected by ICR2)
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

6.3.2 Interrupt Control Register 2 (ICR2)

ICR2 is a 16-bit register that sets the edge detection mode of the external interrupt input pins $\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$. ICR2 is, however, valid only when IRQ interrupt request detection mode is set to the edge detection mode by the sense select bits of IRQ0 to IRQ 3 in Interrupt control register 1 (ICR1). If the IRQ interrupt request detection mode has been set to low level detection mode, the setting of ICR2 is ignored.

Bit	Bit Name	Initial Value	R/W	Description
15 14	IRQ0ES1 IRQ0ES0	0 0	R/W R/W	This bit sets the IRQ0 interrupt request edge detection mode. 00: Interrupt request is detected on falling edge of $\overline{\text{IRQ0}}$ input 01: Interrupt request is detected on rising edge of $\overline{\text{IRQ0}}$ input 10: Interrupt request is detected on both of falling and rising edge of $\overline{\text{IRQ0}}$ input 11: Cannot be set

Bit	Bit Name	Initial Value	R/W	Description
13 12	IRQ1ES1 IRQ1ES0	0 0	R/W R/W	This bit sets the IRQ1 interrupt request edge detection mode. 00: Interrupt request is detected on falling edge of $\overline{\text{IRQ1}}$ input 01: Interrupt request is detected on rising edge of $\overline{\text{IRQ1}}$ input 10: Interrupt request is detected on both of falling and rising edge of $\overline{\text{IRQ1}}$ input 11: Cannot be set
11 10	IRQ2ES1 IRQ2ES0	0 0	R/W R/W	This bit sets the IRQ2 interrupt request edge detection mode. 00: Interrupt request is detected on falling edge of $\overline{\text{IRQ2}}$ input 01: Interrupt request is detected on rising edge of $\overline{\text{IRQ2}}$ input 10: Interrupt request is detected on both of falling and rising edge of $\overline{\text{IRQ2}}$ input 11: Cannot be set
9 8	IRQ3ES1 IRQ3ES0	0 0	R/W R/W	This bit sets the IRQ3 interrupt request edge detection mode. 00: Interrupt request is detected on falling edge of $\overline{\text{IRQ3}}$ input 01: Interrupt request is detected on rising edge of $\overline{\text{IRQ3}}$ input 10: Interrupt request is detected on both of falling and rising edge of $\overline{\text{IRQ3}}$ input 11: Cannot be set
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

6.3.3 IRQ Status Register (ISR)

ISR is a 16-bit register that indicates the interrupt request status of the external interrupt input pins $\overline{IRQ0}$ to $\overline{IRQ3}$. When IRQ interrupts are set to edge detection, held interrupt requests can be withdrawn by writing 0 to IRQnF after reading IRQnF = 1.

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 6 5 4	IRQ0F IRQ1F IRQ2F IRQ3F	0 0 0 0	R/W R/W R/W R/W	IRQ0 to IRQ3 Flags These bits display the IRQ0 to IRQ3 interrupt request status. [Setting condition] • When interrupt source that is selected by ICR1 and ICR2 has occurred. [Clearing conditions] • When 0 is written after reading IRQnF = 1 • When interrupt exception processing has been executed at high level of $\overline{IRQn}$ input under the low level detection mode. • When IRQn interrupt exception processing has been executed under the edge detection mode of falling edge, rising edge or both of falling and rising edge.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

6.3.4 Interrupt Priority Registers A, D to I (IPRA, IPRD to IPRI)

Interrupt priority registers are seven 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 6.2 Interrupt Exception Processing Vectors and Priorities. 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	Bit Name	Initial Value	R/W	Description
15	IPR15	0	R/W	These bits set priority levels for the corresponding interrupt source.
14	IPR14	0	R/W	
13	IPR13	0	R/W	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)
12	IPR12	0	R/W	

Bit	Bit Name	Initial Value	R/W	Description
11 10 9 8	IPR11 IPR10 IPR9 IPR8	0 0 0 0	R/W R/W R/W R/W	These bits 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)
7 6 5 4	IPR7 IPR6 IPR5 IPR4	0 0 0 0	R/W R/W R/W R/W	These bits 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
3	IPR3	0	R/W	These bits set priority levels for the corresponding interrupt source.
2	IPR2	0	R/W	0000: Priority level 0 (lowest)
1	IPR1	0	R/W	0001: Priority level 1
0	IPR0	0	R/W	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 three types of interrupt sources: NMI, IRQ, and on-chip peripheral modules. Each interrupt has a priority expressed as a priority level (0 to 16, with 0 the lowest and 16 the highest). Giving an interrupt a priority level of 0 masks it.

NMI Interrupts: The NMI interrupt has priority 16 and is always accepted. Input at the NMI pin is detected by edge. Use the NMI edge select bit (NMIE) in the interrupt control register 1 (ICR1) to select either the rising or falling edge. NMI interrupt exception processing sets the interrupt mask level bits (I3 to I0) in the status register (SR) to level 15.

IRQ3 to IRQ0 Interrupts: IRQ interrupts are requested by input from pins $\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$. Set the IRQ sense select bits (IRQ0S to IRQ3S) of the interrupt control register 1 (ICR1) and IRQ edge select bit (IRQ0ES[1:0] to IRQ3ES[1:0]) of the interrupt control register 2 (ICR2) to select low level detection, falling edge detection, or rising edge detection for each pin. The priority level can be set from 0 to 15 for each pin using the interrupt priority register A (IPRA).

When IRQ interrupts are set to low level detection, an interrupt request signal is sent to the INTC during the period the IRQ pin is low level. Interrupt request signals are not sent to the INTC when the IRQ pin becomes high level. Interrupt request levels can be confirmed by reading the IRQ flags (IRQ0F to IRQ3F) of the IRQ status register (ISR).

When IRQ interrupts are set to falling edge detection, interrupt request signals are sent to the INTC upon detecting a change on the IRQ pin from high to low level. The results of detection for IRQ interrupt request are maintained until the interrupt request is accepted. It is possible to confirm that IRQ interrupt requests have been detected by reading the IRQ flags (IRQ0F to IRQ3F) of the IRQ status register (ISR), and by writing a 0 after reading a 1, IRQ interrupt request detection results can be withdrawn.

In IRQ interrupt exception processing, the interrupt mask bits (I3 to I0) of the status register (SR) are set to the priority level value of the accepted IRQ interrupt. Figure 6.2 shows the block diagram of this IRQ3 to IRQ0 interrupts.

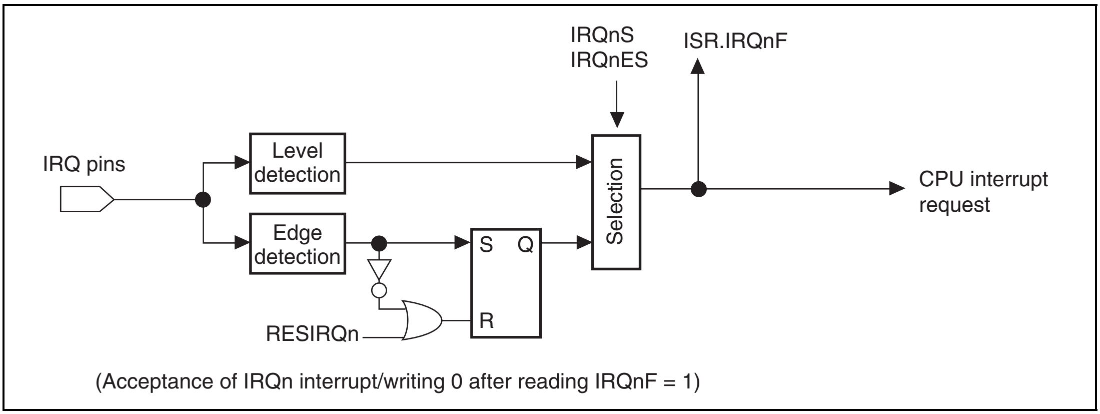


Figure 6.2 Block Diagram of IRQ3 to IRQ0 Interrupts Control

6.4.2 On-Chip Peripheral Module Interrupts

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

As a different interrupt vector is assigned to each interrupt source, the exception service routine does not have to decide which interrupt has occurred. Priority levels between 0 and 15 can be assigned to individual on-chip peripheral modules in interrupt priority registers A, D to I (IPRA, IPRD to IPRI). On-chip peripheral module interrupt exception processing 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.

6.5 Interrupt Exception Processing Vectors Table

Table 6.2 lists interrupt sources and their vector numbers, vector table address offsets and interrupt priorities.

Each interrupt source is allocated a different vector number and vector table address offset. Vector table addresses are calculated from the vector numbers and address offsets. In interrupt exception processing, the exception service routine start address is fetched from the vector table indicated by the vector table address. For the details of calculation of vector table address, see table 5.4, Calculating Exception Processing Vector Table Addresses in the section 5 Exception Processing.

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, D to I (IPRA, IPRD to IPRI). However, the smaller vector number has interrupt source, the higher priority ranking is assigned among two or more interrupt sources specified by the same IPR, and the priority ranking cannot be changed. A power-on reset assigns priority level 0 to IRQ interrupts and on-chip peripheral module interrupts. If the same priority level is assigned to two or more interrupt sources and interrupts from those sources occur simultaneously, they are processed by the default priority order indicated in table 6.2.

Table 6.2 Interrupt Exception Processing Vectors and Priorities

Interrupt Source	Name	Vector No.	Vector Table Starting Address	IPR	Default Priority
External pin	NMI	11	H'0000002C	—	High
—	Reserved by system	12	H'00000030	—	↑
—	Reserved by system	14	H'00000038	—	
—	Reserved by system	15	H'0000003C	—	
Interrupts	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	
	Reserved by system	68	H'00000110	—	
	Reserved by system	69	H'00000114	—	
	Reserved by system	70	H'00000118	—	
	Reserved by system	71	H'0000011C	—	
—	Reserved by system	72	H'00000120	—	
	Reserved by system	76	H'00000130	—	
	Reserved by system	80	H'00000140	—	
	Reserved by system	84	H'00000150	—	
MTU channel 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	
MTU channel 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		Low

Interrupt Source	Name	Vector No.	Vector Table Starting Address	IPR	Default Priority
MTU channel 2	TGIA_2	104	H'000001A0	IPRE15 to IPRE12	High
	TGIB_2	105	H'000001A4		↑
	TCIV_2	108	H'000001B0	IPRE11 to IPRE8	
	TCIU_2	109	H'000001B4		
MTU channel 3	TGIA_3	112	H'000001C0	IPRE7 to IPRE4	
	TGIB_3	113	H'000001C4		
	TGIC_3	114	H'000001C8		
	TGID_3	115	H'000001CC		
	TCIV_3	116	H'000001D0	IPRE3 to IPRE0	
MTU channel 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	
—	Reserved by system	128 to 135	H'00000200 to H'0000021C	—	
A/D	ADI0	136	H'00000220	IPRG15 to IPRG12	
	ADI1	137	H'00000224		
—	Reserved by system	140	H'00000230	—	
CMT	CMI0	144	H'00000240	IPRG7 to IPRG4	
	CMI1	148	H'00000250	IPRG3 to IPRG0	
Watchdog timer	ITI	152	H'00000260	IPRH15 to IPRH12	
—	Reserved by system	153	H'00000264	—	
I/O (MTU)	MTUPOE	156	H'00000270	IPRH11 to IPRH8	
—	Reserved by system	160 to 167	H'00000290 to H'0000029C	—	
SCI channel 2	ERI_2	168	H'000002A0	IPRI15 to IPRI12	
	RXI_2	169	H'000002A4		
	TXI_2	170	H'000002A8		
	TEI_2	171	H'000002AC		Low

Interrupt Source	Name	Vector No.	Vector Table Starting Address	IPR	Default Priority
SCI channel 3	ERI_3	172	H'000002B0	IPRI11 to IPRI8	High ↑
	RXI_3	173	H'000002B4		
	TXI_3	174	H'000002B8		
	TEI_3	175	H'000002BC		
—	Reserved by system	176	H'000002C0	—	
	Reserved by system	177	H'000002C4		
	Reserved by system	178	H'000002C8		
	Reserved by system	179	H'000002CC		
—	Reserved by system	180	H'000002D0	—	
	Reserved by system	181	H'000002D4		
—	Reserved by system	184	H'000002E0	—	
—	Reserved by system	188 to 196	H'000002F0 to H'00000310	—	
—	Reserved by system	200	H'00000320	—	
—	Reserved by system	204	H'00000330	—	
—	Reserved by system	208	H'00000340	—	
	Reserved by system	209	H'00000344		
	Reserved by system	210	H'00000348		
	Reserved by system	211	H'0000034C		
—	Reserved by system	212	H'00000350 to H'000003DC	—	Low

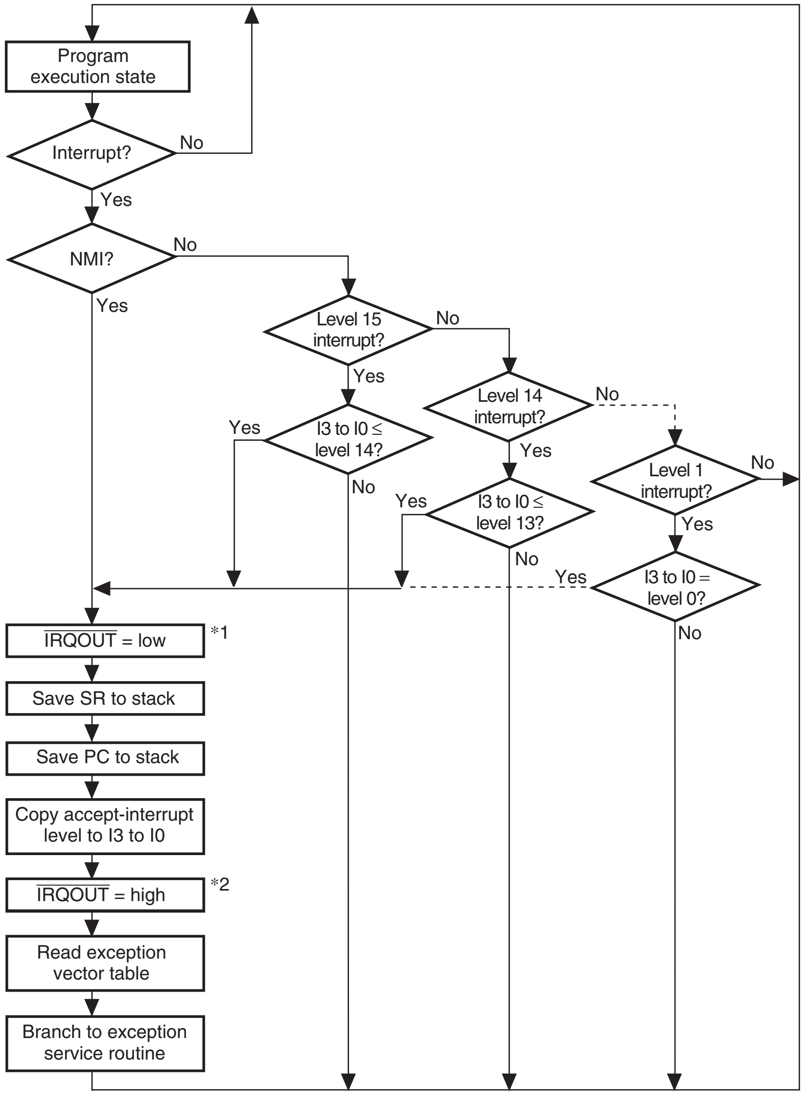
6.6 Operation

6.6.1 Interrupt Sequence

The sequence of interrupt operations is explained below.

1. The interrupt request sources send interrupt request signals to the interrupt controller.
2. The interrupt controller selects the highest priority interrupt in the interrupt requests sent, according to the priority levels set in interrupt priority registers A, D to I (IPRA, IPRD to IPRI). 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 order indicated in table 6.2.
3. The interrupt controller compares the priority level of the selected interrupt request with the interrupt mask bits (I3 to I0) in the CPU's status register (SR). If the request priority level is equal to or less than the level set in I3 to I0, the request is ignored. If the request priority level is higher than the level in bits I3 to I0, the interrupt controller accepts the interrupt 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 when CPU decodes the instruction to be executed. Instead of executing the decoded instruction, the CPU starts interrupt exception processing (figure 6.5).
6. SR and PC are saved onto the stack.
7. The priority level of the accepted interrupt is copied to the interrupt mask level bits (I3 to I0) in the status register (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 accepting, the $\overline{\text{IRQOUT}}$ pin holds low level.
9. The CPU reads the start address of the exception service routine from the exception vector table for the accepted interrupt, jumps to that address, and starts executing the program. This jump is not a delay branch.

Note: * 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 (ISR). Interrupts held pending due to edge detection are cleared by a power-on reset or a manual reset.



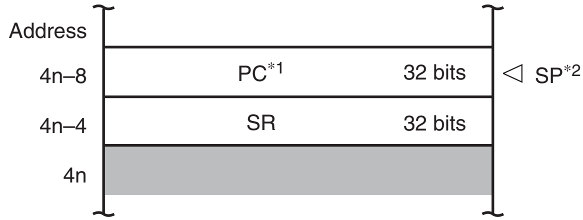
Notes: I3 to I0 are Interrupt mask bits of status register (SR) in the CPU

1. $\overline{\text{IRQOUT}}$ is the same signal as 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.

Figure 6.3 Interrupt Sequence Flowchart

6.6.2 Stack after Interrupt Exception Processing

Figure 6.4 shows the stack after interrupt exception processing.



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

Figure 6.4 Stack after Interrupt Exception Processing

6.7 Interrupt Response Time

Table 6.3 lists the interrupt response time, which is the time from the occurrence of an interrupt request until the interrupt exception processing starts and fetching of the first instruction of the interrupt service routine begins. Figure 6.5 shows an example of the pipeline operation when an IRQ interrupt is accepted.

Table 6.3 Interrupt Response Time

Item		Number of States: NMI, Peripheral Module	Number of States: IRQ	Remarks
Idle cycle		0 or 1	1	
Interrupt priority judgment and comparison with SR mask bits		2	3	
Wait for completion of sequence currently being executed by CPU		X (≥ 0)	X (≥ 0)	The longest sequence is for interrupt or address-error exception processing (X = 4 + m1 + m2 + m3 + m4). If an interrupt-masking instruction follows, however, the time may be even longer.
Time from start of interrupt exception processing until fetch of first instruction of exception service routine starts		5 + m1 + m2 + m3	5 + m1 + m2 + m3	Performs the saving PC and SR, and vector address fetch.
Interrupt response time	Total:	(7 or 8) + m1 + m2 + m3 + X	9 + m1 + m2 + m3 + X	
	Minimum:	10	12	0.25 to 0.28 μs
	Maximum:	12 + 2 (m1 + m2 + m3) + m4	13 + 2 (m1 + m2 + m3) + m4	0.48 μs*

Note: * 0.48 μs at 40 MHz is the value in the case that m1 = m2 = m3 = m4 = 1.

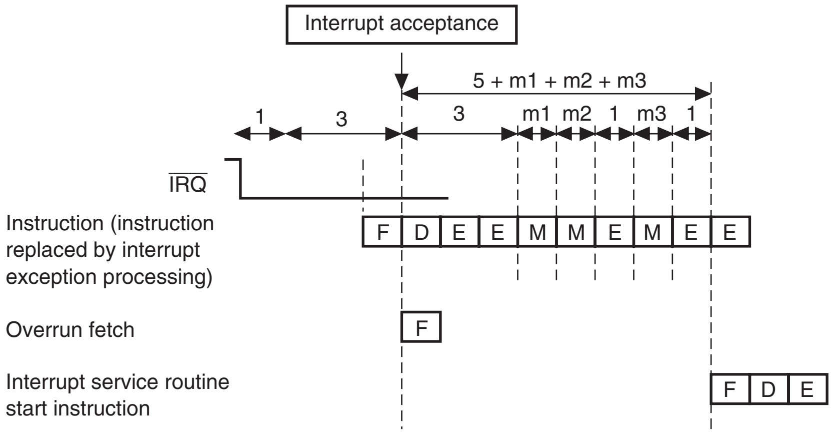
m1 to m4 are the number of states needed for the following memory accesses.

m1: SR save (longword write)

m2: PC save (longword write)

m3: Vector address read (longword read)

m4: Fetch first instruction of interrupt service routine



Legend:
F: Instruction fetch (instruction fetched from memory where program is stored).
D: Instruction decoding (fetched instruction is decoded).
E: Instruction execution (data operation and address calculation is performed according to the results of decoding).
M: Memory access (data in memory is accessed).

Figure 6.5 Example of the Pipeline Operation when an IRQ Interrupt is Accepted

Section 7 Bus State Controller (BSC)

The bus state controller (BSC) controls accesses to the on-chip ROM, RAM, and peripheral module registers.

7.1 Features

The BSC has the following features:

- On-chip ROM and RAM interfaces
  - On-chip ROM and RAM access of 32 bits in 1 state
- Accesses to on-chip peripheral module registers

7.2 Input/output Pin

There are no pins corresponding to this function.

7.3 Register

The BSC has the following register. For details on these register addresses and register states in each processing states, refer to section 18, List of Registers.

- Bus control register 1 (BCR1)

7.4 Address Map

Table 7.1 shows the address map.

Table 7.1 Address Map

On-chip ROM enabled mode

Address	Space	Memory	Size	Bus Width
H'0000 0000 to H'0000 7FFF	On-chip ROM	On-chip ROM	32 kbytes	32 bits
H'0000 8000 to H'0001 FFFF			Reserved	32 bits
H'0002 0000 to H'0003 FFFF			Reserved	32 bits
H'0004 0000 to H'FFFF 7FFF	Reserved	Reserved	Reserved	
H'FFFF 8000 to H'FFFF BFFF	On-chip peripheral module	On-chip peripheral module	16 kbytes	8, 16 bits
H'FFFF C000 to H'FFFF CFFF	Reserved	Reserved		
H'FFFF D000 to H'FFFF DFFF	On-chip RAM	On-chip RAM	Reserved	32 bits
H'FFFF E000 to H'FFFF F7FF			Reserved	32 bits
H'FFFF F800 to H'FFFF FFFF			2 kbytes	32 bits

Note: Reserved area should not be accessed, or operation cannot be guaranteed.

7.5 Register Description

7.5.1 Bus Control Register 1 (BCR1)

BCR1 is a 16-bit readable/writable register that enables access to the MTU control registers.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved These bits are always read as 0 and should always be written to 0.
14	—	1	R	Reserved These bits are always read as 1 and should always be written to 1.
13	MTURWE	1	R/W	MTU Read/Write Enable This bit enables MTU control register access. For details, refer to MTU section. 0: MTU control register access is disabled 1: MTU control register access is enabled
12 to 8	—	All 0	R	Reserved These bits are always read as 0 and the write value should always be 0.
7 to 4	—	All 0	R	Reserved These bits are always read as 0 and the write value should always be 0.
3 to 0	—	All 1	R	Reserved These bits are always read as 1 and the write value should always be 1.

7.6 On-chip Peripheral I/O Register Access

On-chip peripheral I/O registers are accessed from the bus state controller, as shown in table 7.2.

Table 7.2 On-chip Peripheral I/O Register Access

On-chip Peripheral Module	SCI	MTU, POE	INTC	PFC, PORT	CMT	A/D	WDT
Connected bus width	8 bits	16 bits	16 bits	16 bits	16 bits	8 bits	16 bits
Access cycle	2 cycles*1	2 cycles*1	2 cycles*2	2 cycles*2	2 cycles*1	3 cycles*1	3 cycles*2

Notes: 1. In terms of the peripheral clock value
2. In terms of the system clock value

Section 8 Multi-Function Timer Pulse Unit (MTU)

This LSI has an on-chip multi-function timer pulse unit (MTU) that comprises five 16-bit timer channels.

The block diagram is shown in figure 8.1.

8.1 Features

- Maximum 16-pulse input/output
- Selection of 8 counter input clocks for each channel
- The following operations can be set for each channel:
  - 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
- 23 interrupt sources
- Automatic transfer of register data
- A/D converter conversion start trigger can be generated
- Module standby mode can be set
- Positive and negative 3-phase waveforms (6-phase waveforms in total) can be output in complementary or reset synchronous PWM mode by combining channels 3 and 4.
- AC synchronous motor (brushless DC motor) can be driven in complementary or reset synchronous PWM mode by combining channels 0, 3, and 4. Chopping or level output can be selected as drive waveform output.

Table 8.1 MTU Functions

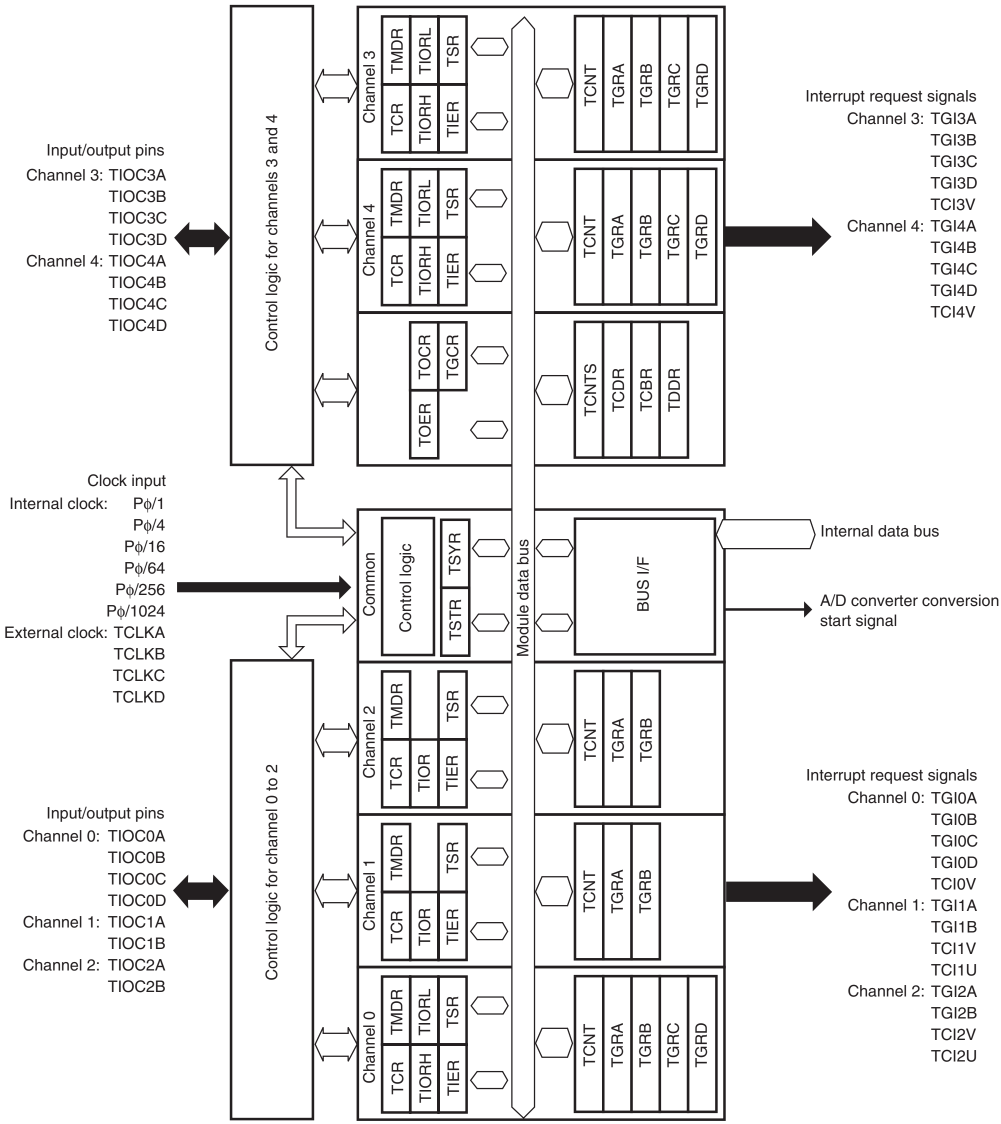
Item		Channel 0	Channel 1	Channel 2	Channel 3	Channel 4
Count clock		Pφ/1 Pφ/4 Pφ/16 Pφ/64 TCLKA TCLKB TCLKC TCLKD	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/256 TCLKA TCLKB	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/1024 TCLKA TCLKB TCLKC	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/256 Pφ/1024 TCLKA TCLKB	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/256 Pφ/1024 TCLKA TCLKB
General registers		TGRA_0 TGRB_0	TGRA_1 TGRB_1	TGRA_2 TGRB_2	TGRA_3 TGRB_3	TGRA_4 TGRB_4
General registers/ buffer registers		TGRC_0 TGRD_0	—	—	TGRC_3 TGRD_3	TGRC_4 TGRD_4
I/O pins		TIOC0A TIOC0B TIOC0C TIOC0D	TIOC1A TIOC1B	TIOC2A TIOC2B	TIOC3A TIOC3B TIOC3C TIOC3D	TIOC4A TIOC4B TIOC4C TIOC4D
Counter clear function		TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture
Compare match output	0 output	○	○	○	○	○
	1 output	○	○	○	○	○
	Toggle output	○	○	○	○	○
Input capture function		○	○	○	○	○
Synchronous operation		○	○	○	○	○
PWM mode 1		○	○	○	○	○
PWM mode 2		○	○	○	—	—
Complementary PWM mode		—	—	—	○	○
Reset synchronous PWM mode		—	—	—	○	○
AC synchronous motor drive mode		○	—	—	○	○

Item	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4
Phase counting mode	—	○	○	—	—
Buffer operation	○	—	—	○	○
A/D converter start trigger	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
Interrupt sources	5 sources • Compare match or input capture 0A • Compare match or input capture 0B • Compare match or input capture 0C • Compare match or input capture 0D • Overflow	4 sources • Compare match or input capture 1A • Compare match or input capture 1B • Overflow • Underflow	4 sources • Compare match or input capture 2A • Compare match or input capture 2B • Overflow • Underflow	5 sources • Compare match or input capture 3A • Compare match or input capture 3B • Compare match or input capture 3C • Compare match or input capture 3D • Overflow	5 sources • Compare match or input capture 4A • Compare match or input capture 4B • Compare match or input capture 4C • Compare match or input capture 4D • Underflow/ Overflow

Legend:

○ : Possible

— : Not possible



Legend:

TSTR:	Timer start register	TIER:	Timer interrupt enable register
TSYR:	Timer synchro register	TSR:	Timer status register
TCR:	Timer control register	TCNT:	Timer counter
TMDR:	Timer mode register	TGR (A, B, C, D):	Timer general registers (A, B, C, D)
TIOR (H, L):	Timer I/O control registers (H, L)		

Figure 8.1 Block Diagram of MTU

8.2 Input/Output Pins

Table 8.2 Pin configuration

Channel	Symbol	I/O	Function
All	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

8.3 Register Descriptions

The MTU has the following registers. For details on register addresses and register states during each process, refer to section 18, 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.

- Timer control register_0 (TCR_0)
- Timer mode register_0 (TMDR_0)
- Timer I/O control register H_0 (TIORH_0)
- Timer I/O control register L_0 (TIORL_0)
- Timer interrupt enable register_0 (TIER_0)
- Timer status register_0 (TSR_0)
- Timer counter_0 (TCNT_0)
- Timer general register A_0 (TGRA_0)
- Timer general register B_0 (TGRB_0)
- Timer general register C_0 (TGRC_0)
- Timer general register D_0 (TGRD_0)
- Timer control register_1 (TCR_1)
- Timer mode register_1 (TMDR_1)
- Timer I/O control register _1 (TIOR_1)
- Timer interrupt enable register_1 (TIER_1)
- Timer status register_1 (TSR_1)
- Timer counter_1 (TCNT_1)
- Timer general register A_1 (TGRA_1)
- Timer general register B_1 (TGRB_1)
- Timer control register_2 (TCR_2)
- Timer mode register_2 (TMDR_2)
- Timer I/O control register_2 (TIOR_2)
- Timer interrupt enable register_2 (TIER_2)
- Timer status register_2 (TSR_2)
- Timer counter_2 (TCNT_2)
- Timer general register A_2 (TGRA_2)
- Timer general register B_2 (TGRB_2)
- Timer control register_3 (TCR_3)

- Timer mode register_3 (TMDR_3)
- Timer I/O control register H_3 (TIORH_3)
- Timer I/O control register L_3 (TIORL_3)
- Timer interrupt enable register_3 (TIER_3)
- Timer status register_3 (TSR_3)
- Timer counter_3 (TCNT_3)
- Timer general register A_3 (TGRA_3)
- Timer general register B_3 (TGRB_3)
- Timer general register C_3 (TGRC_3)
- Timer general register D_3 (TGRD_3)
- Timer control register_4 (TCR_4)
- Timer mode register_4 (TMDR_4)
- Timer I/O control register H_4 (TIORH_4)
- Timer I/O control register L_4 (TIORL_4)
- Timer interrupt enable register_4 (TIER_4)
- Timer status register_4 (TSR_4)
- Timer counter_4 (TCNT_4)
- Timer general register A_4 (TGRA_4)
- Timer general register B_4 (TGRB_4)
- Timer general register C_4 (TGRC_4)
- Timer general register D_4 (TGRD_4)

Common Registers

- Timer start register (TSTR)
- Timer synchro register (TSYR)

Common Registers for timers 3 and 4

- Timer output master enable register (TOER)
- Timer output control enable register (TOCR)
- Timer gate control register (TGCR)
- Timer cycle data register (TCDR)
- Timer dead time data register (TDDR)
- Timer subcounter (TCNTS)
- Timer cycle buffer register (TCBR)

8.3.1 Timer Control Register (TCR)

The TCR registers are 8-bit readable/writable registers that control the TCNT operation for each channel. The MTU has a total of five TCR registers, one for each channel (channel 0 to 4). TCR register settings should be conducted only when TCNT operation is stopped.

Bit	Bit Name	Initial value	R/W	Description
7	CCLR2	0	R/W	Counter Clear 0 to 2
6	CCLR1	0	R/W	These bits select the TCNT counter clearing source. See tables 8.3 and 8.4 for details.
5	CCLR0	0	R/W	
4	CKEG1	0	R/W	Clock Edge 0 and 1
3	CKEG0	0	R/W	These bits select the input clock edge. When the input clock is counted using both edges, the input clock period is halved (e.g. Pϕ /4 both edges = ϕ /2 rising edge). If phase counting mode is used on channels 1 and 2, this setting is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the input clock is Pϕ /4 or slower. When Pϕ /1, or the overflow/underflow of another channel is selected for the input clock, although values can be written, counter operation compiles with the initial value. 00: Count at rising edge 01: Count at falling edge 1X: Count at both edges Legend: X: Don't care
2	TPSC2	0	R/W	Time Prescaler 0 to 2
1	TPSC1	0	R/W	These bits select the TCNT counter clock. The clock source can be selected independently for each channel. See tables 8.5 to 8.8 for details.
0	TPSC0	0	R/W	

Table 8.3 CCLR0 to CCLR2 (channels 0, 3, and 4)

Channel	Bit 7 CCLR2	Bit 6 CCLR1	Bit 5 CCLR0	Description
0, 3, 4	0	0	0	TCNT clearing disabled
			1	TCNT cleared by TGRA compare match/input capture
		1	0	TCNT cleared by TGRB compare match/input capture
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/ synchronous operation*[1]
	1	0	0	TCNT clearing disabled
			1	TCNT cleared by TGRC compare match/input capture*[2]
		1	0	TCNT cleared by TGRD compare match/input capture*[2]
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/ synchronous operation*[1]

Notes: 1. Synchronous operation is set by setting the SYNC bit in TSYR to 1.
2. When TGRC or TGRD is used as a buffer register, TCNT is not cleared because the buffer register setting has priority, and compare match/input capture does not occur.

Table 8.4 CCLR0 to CCLR2 (channels 1 and 2)

Channel	Bit 7 Reserved*[2]	Bit 6 CCLR1	Bit 5 CCLR0	Description
1, 2	0	0	0	TCNT clearing disabled
			1	TCNT cleared by TGRA compare match/input capture
		1	0	TCNT cleared by TGRB compare match/input capture
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/ synchronous operation*[1]

Notes: 1. Synchronous operation is selected by setting the SYNC bit in TSYR to 1.
2. Bit 7 is reserved in channels 1 and 2. It is always read as 0.
Writing is ignored.

Table 8.5 TPSC0 to TPSC2 (channel 0)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
0	0	0	0	Internal clock: counts on Pϕ/1
			1	Internal clock: counts on Pϕ/4
		1	0	Internal clock: counts on Pϕ/16
			1	Internal clock: counts on Pϕ/64
	1	0	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input
		1	0	External clock: counts on TCLKC pin input
			1	External clock: counts on TCLKD pin input

Table 8.6 TPSC0 to TPSC2 (channel 1)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
1	0	0	0	Internal clock: counts on Pϕ/1
			1	Internal clock: counts on Pϕ/4
		1	0	Internal clock: counts on Pϕ/16
			1	Internal clock: counts on Pϕ/64
	1	0	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input
		1	0	Internal clock: counts on Pϕ/256
			1	Counts on TCNT_2 overflow/underflow

Note: This setting is ignored when channel 1 is in phase counting mode.

Table 8.7 TPSC0 to TPSC2 (channel 2)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
2	0	0	0	Internal clock: counts on Pϕ/1
			1	Internal clock: counts on Pϕ/4
		1	0	Internal clock: counts on Pϕ/16
			1	Internal clock: counts on Pϕ/64
	1	0	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input
		1	0	External clock: counts on TCLKC pin input
			1	Internal clock: counts on Pϕ/1024

Note: This setting is ignored when channel 2 is in phase counting mode.

Table 8.8 TPSC0 to TPSC2 (channels 3 and 4)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
3, 4	0	0	0	Internal clock: counts on Pϕ/1
			1	Internal clock: counts on Pϕ/4
		1	0	Internal clock: counts on Pϕ/16
			1	Internal clock: counts on Pϕ/64
	1	0	0	Internal clock: counts on Pϕ/256
			1	Internal clock: counts on Pϕ/1024
		1	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input

8.3.2 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 MTU has five TMDR registers, one for each channel. TMDR register settings should be changed only when TCNT operation is stopped.

Bit	Bit Name	Initial value	R/W	Description
7, 6	—	All 1	—	Reserved These bits are always read as 1, and should only be written with 1.
5	BFB	0	R/W	Buffer Operation B Specifies whether TGRB is to operate in the normal way, or TGRB and TGRD are to be used together for buffer operation. When TGRD is used as a buffer register, TGRD input capture/output compare is not generated. In channels 1 and 2, which have no TGRD, bit 5 is reserved. It is always read as 0, and the write value should always be 0. 0: TGRB and TGRD operate normally 1: TGRB and TGRD used together for buffer operation
4	BFA	0	R/W	Buffer Operation A Specifies whether TGRA is to operate in the normal way, or TGRA and TGRC are to be used together for buffer operation. When TGRC is used as a buffer register, TGRC input capture/output compare is not generated. In channels 1 and 2, which have no TGRC, bit 4 is reserved. It is always read as 0, and the write value should always be 0. 0: TGRA and TGRD operate normally 1: TGRA and TGRC used together for buffer operation
3	MD3	0	R/W	Modes 0 to 3
2	MD2	0	R/W	These bits are used to set the timer operating mode.
1	MD1	0	R/W	See table 8.9 for details.
0	MD0	0	R/W	

Table 8.9 MD0 to MD3

Bit 3 MD3	Bit 2 MD2	Bit 1 MD1	Bit 0 MD0	Description
0	0	0	0	Normal operation
			1	Reserved (do not set)
		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	Reserved (do not set)
		1	X	Reserved (do not set)
	1	0	0	Reserved (do not set)
			1	Complementary PWM mode 1 (transmit at peak)*[3]
		1	0	Complementary PWM mode 2 (transmit at valley)*[3]
			1	Complementary PWM mode 2 (transmit at peak and valley)*[3]

Legend:

X: Don't care

Notes: 1. PWM mode 2 can not be set for channels 3, 4.

2. Phase counting mode can not be set for channels 0, 3, and 4.

3. Reset synchronous PWM mode, complementary PWM mode can only be set for channel 3. When channel 3 is set to reset synchronous PWM mode or complementary PWM mode, the channel 4 settings become ineffective and automatically conform to the channel 3 settings. However, do not set channel 4 to reset synchronous PWM mode or complementary PWM mode. Reset synchronous PWM mode and complementary PWM mode can not be set for channels 0, 1, and 2.

8.3.3 Timer I/O Control Register (TIOR)

The TIOR registers are 8-bit readable/writable registers that control the TGR registers. The MTU has eight TIOR registers, two each for channels 0, 3, and 4, and one each for channels 1 and 2.

Care is required as TIOR is affected by the TMDR setting. 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	IOB3	0	R/W	I/O Control B0 to B3
6	IOB2	0	R/W	Specify the function of TGRB.
5	IOB1	0	R/W	See the following tables.
4	IOB0	0	R/W	TIORH_0: Table 8.10 TIOR_1: Table 8.14 TIOR_2: Table 8.16 TIORH_3: Table 8.18 TIORH_4: Table 8.22
3	IOA3	0	R/W	I/O Control A0 to A3
2	IOA2	0	R/W	Specify the function of TGRA.
1	IOA1	0	R/W	See the following tables.
0	IOA0	0	R/W	TIORH_0: Table 8.11 TIOR_1: Table 8.15 TIOR_2: Table 8.17 TIORH_3: Table 8.19 TIORH_4: Table 8.23

TIORL_0, TIORL_3, TIORL_4

Bit	Bit Name	Initial value	R/W	Description
7	IOD3	0	R/W	I/O Control D0 to D3
6	IOD2	0	R/W	Specify the function of TGRD.
5	IOD1	0	R/W	When the TGRD is used as a buffer register of the TGRB, this setting is invalid and input capture/output compare is not generated.
4	IOD0	0	R/W	See the following tables. TIORL_0: Table 8.12 TIORL_3: Table 8.20 TIORL_4: Table 8.24
3	IOC3	0	R/W	I/O Control C0 to C3
2	IOC2	0	R/W	Specify the function of TGRC.
1	IOC1	0	R/W	When the TGRC is used as a buffer register of the TGRA, this setting is invalid and input capture/output compare is not generated.
0	IOC0	0	R/W	See the following tables. TIORL_0: Table 8.13 TIORL_3: Table 8.21 TIORL_4: Table 8.25

Table 8.10 TIORH_0 (channel 0)

Bit 7 IOB3	Bit 6 IOB2	Bit 5 IOB1	Bit 4 IOB0	Description: TGRB_0 Function	Description: TIOC0B Pin Function
0	0	0	0	Output compare register	Output disabled
			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 disabled
			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

Table 8.11 TIORH_0 (channel 0)

Bit 3 IOA3	Bit 2 IOA2	Bit 1 IOA1	Bit 0 IOA0	Description: TGRA_0 Function	Description: TIOC0A Pin Function
0	0	0	0	Output compare register	Output disabled
			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 disabled
			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

Table 8.12 TIORL_0 (channel 0)

				Description	
Bit 7 IOD3	**Bit 6 IOD2**	**Bit 5 IOD1**	**Bit 4 IOD0**	**TGRD_0 Function**	**TIOC0D Pin Function**
0	0	0	0	Output compare register*	Output disabled
			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 disabled
			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: * 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 8.13 TIORL_0 (channel 0)

Bit 3 IOC3	Bit 2 IOC2	Bit 1 IOC1	Bit 0 IOC0	Description: TGRC_0 Function	Description: TIOC0C Pin Function
0	0	0	0	Output compare register*	Output disable
			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 disabled
			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: * 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 8.14 TIOR_1 (channel 1)

Bit 7 IOB3	Bit 6 IOB2	Bit 5 IOB1	Bit 4 IOB0	Description: TGRB_1 Function	Description: TIOC1B Pin Function
0	0	0	0	Output compare register	Output disabled
			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 disabled
			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

Table 8.15 TIOR_1 (channel 1)

Bit 3 IOA3	Bit 2 IOA2	Bit 1 IOA1	Bit 0 IOA0	Description: TGRA_1 Function	Description: TIOC1A Pin Function
0	0	0	0	Output compare register	Output disabled
			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 disabled
			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

Table 8.16 TIOR_2 (channel 2)

Bit 7 IOB3	Bit 6 IOB2	Bit 5 IOB1	Bit 4 IOB0	Description: TGRB_2 Function	Description: TIOC2B Pin Function
0	0	0	0	Output compare register	Output disabled
			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 disabled
			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

Table 8.17 TIOR_2 (channel 2)

Bit 3 IOA3	Bit 2 IOA2	Bit 1 IOA1	Bit 0 IOA0	Description: TGRA_2 Function	Description: TIOC2A Pin Function
0	0	0	0	Output compare register	Output disabled
			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 disabled
			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

Table 8.18 TIORH_3 (channel 3)

Bit 7 IOB3	Bit 6 IOB2	Bit 5 IOB1	Bit 4 IOB0	Description: TGRB_3 Function	Description: TIOC3B Pin Function
0	0	0	0	Output compare register	Output disabled
			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 disabled
			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

Table 8.19 TIORH_3 (channel 3)

Bit 3 IOA3	Bit 2 IOA2	Bit 1 IOA1	Bit 0 IOA0	Description: TGRA_3 Function	Description: TIOC3A Pin Function
0	0	0	0	Output compare register	Output disabled
			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 disabled
			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

Table 8.20 TIORL_3 (channel 3)

Bit 7 IOD3	Bit 6 IOD2	Bit 5 IOD1	Bit 4 IOD0	Description: TGRD_3 Function	Description: TIOC3D Pin Function
0	0	0	0	Output compare register*	Output disabled
			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 disabled
			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: * 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 8.21 TIORL_3 (channel 3)

Bit 3 IOC3	Bit 2 IOC2	Bit 1 IOC1	Bit 0 IOC0	Description: TGRC_3 Function	Description: TIOC3C Pin Function
0	0	0	0	Output compare register*	Output disabled
			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 disabled
			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: * 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 8.22 TIORH_4 (channel 4)

Bit 7 IOB3	Bit 6 IOB2	Bit 5 IOB1	Bit 4 IOB0	Description: TGRB_4 Function	Description: TIOC4B Pin Function
0	0	0	0	Output compare register	Output disabled
			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 disabled
			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

Table 8.23 TIORH_4 (channel 4)

Bit 3 IOA3	Bit 2 IOA2	Bit 1 IOA1	Bit 0 IOA0	Description: TGRA_4 Function	Description: TIOC4A Pin Function
0	0	0	0	Output compare register	Output disabled
			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 disabled
			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

Table 8.24 TIORL_4 (channel 4)

Bit 7 IOD3	Bit 6 IOD2	Bit 5 IOD1	Bit 4 IOD0	Description: TGRD_4 Function	Description: TIOC4B Pin Function
0	0	0	0	Output compare register*	Output disabled
			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 disabled
			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: * 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 8.25 TIORL_4 (channel 4)

Bit 3 IOC3	Bit 2 IOC2	Bit 1 IOC1	Bit 0 IOC0	Description: TGRC_4 Function	Description: TIOC4C Pin Function
0	0	0	0	Output compare register*	Output disabled
			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 disabled
			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: * 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.

8.3.4 Timer Interrupt Enable Register (TIER)

The TIER registers are 8-bit readable/writable registers that control enabling or disabling of interrupt requests for each channel. The MTU has five TIER registers, one for each channel.

Bit	Bit Name	Initial value	R/W	Description
7	TTGE	0	R/W	A/D Conversion Start Request Enable Enables or disables generation of A/D conversion start requests by TGRA input capture/compare match. 0: A/D conversion start request generation disabled 1: A/D conversion start request generation enabled
6	—	1	R	Reserved This bit is always read as 1, and should only be written with 1.
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 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 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

8.3.5 Timer Status Register (TSR)

The TSR registers are 8-bit readable/writable registers that indicate the status of each channel. The MTU has five TSR registers, one for each channel.

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, 2, 3, and 4. In channel 0, bit 7 is reserved. It is always read as 1, and should only be written with 1. 0: TCNT counts down 1: TCNT counts up
6	—	1	R	Reserved This bit is always read as 1, and should only be written with 1.
5	TCFU	0	R/(W)	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
4	TCFV	0	R/(W)	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 TCNT_4 is underflowed (changes from H'0000 to H'0001) in complementary PWM mode. [Clearing condition] • When 0 is written to TCFV after reading TCFV = 1

Bit	Bit Name	Initial value	R/W	Description
3	TGFD	0	R/(W)	Input Capture/Output Compare Flag D Status flag that indicates the occurrence of TGRD input capture or compare match in channels 0, 3, and 4. Only 0 can be written, for flag clearing. In channels 1 and 2, bit 3 is reserved. It is always read as 0, and the write value should always be 0. [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 condition] • When 0 is written to TGFD after reading TGFD = 1
2	TGFC	0	R/(W)	Input Capture/Output Compare Flag C Status flag that indicates the occurrence of TGRC input capture or compare match in channels 0, 3, and 4. Only 0 can be written, for flag clearing. In channels 1 and 2, bit 2 is reserved. It is always read as 0, and the write value should always be 0. [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 condition] • When 0 is written to TGFC after reading TGFC = 1

Bit	Bit Name	Initial value	R/W	Description
1	TGFB	0	R/(W)	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 condition] • When 0 is written to TGFB after reading TGFB = 1
0	TGFA	0	R/(W)	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 condition] • When 0 is written to TGFA after reading TGFA = 1

8.3.6 Timer Counter (TCNT)

The TCNT registers are 16-bit readable/writable counters. The MTU has five TCNT counters, one for each channel. The initial value is H'0000.

The TCNT counters cannot be accessed in 8-bit units; they must always be accessed as a 16-bit unit.

8.3.7 Timer General Register (TGR)

The TGR registers are dual function 16-bit readable/writable registers, functioning as either output compare or input capture registers. The MTU has 16 TGR registers, four each for channels 0, 3, and 4 and two each for channels 1 and 2. TGRC and TGRD for channels 0, 3, and 4 can also be designated for operation as buffer registers. The TGR registers cannot be accessed in 8-bit units; they must always be accessed as a 16-bit unit. TGR buffer register combinations are TGRA-TGRC and TGRB-TGRD. The initial value is H'FFFF.

8.3.8 Timer Start Register (TSTR)

TSTR is an 8-bit readable/writable register that selects operation/stoppage for channels 0 to 4. When setting the operating mode in TMDR or setting the count clock in TCR, first stop the TCNT counter.

Bit	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. Only 0 should be written to these bits.
2	CST2	0	R/W	Counter Start 2 to 0
1	CST1	0	R/W	These bits select operation or stoppage for TCNT.
0	CST0	0	R/W	If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. If TIOR is written to when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value. 0: TCNT_2 and TCNT_0 count operation is stopped 1: TCNT_2 and TCNT_0 performs count operation

8.3.9 Timer Synchro 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	Bit Name	Initial value	R/W	Description
7 6	SYNC4 SYNC3	0 0	R/W R/W	Timer Synchro 4 and 3 These bits are used to select whether operation is independent of or synchronized with other channels. When synchronous operation is selected, the TCNT synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channel, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits CCLR0 to CCLR2 in TCR. 0: TCNT_4 and TCNT_3 operate independently (TCNT presetting/clearing is unrelated to other channels) 1: TCNT_4 and TCNT_3 performs synchronous operation TCNT synchronous presetting/synchronous clearing is possible
5 to 3	—	All 0	R	Reserved These bits are always read as 0. Only 0 should be written to these bits.

Bit	Bit Name	Initial value	R/W	Description
2	SYNC2	0	R/W	Timer Synchro 2 to 0
1	SYNC1	0	R/W	These bits are used to select whether operation is independent of or synchronized with other channels.
0	SYNC0	0	R/W	When synchronous operation is selected, the TCNT synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channel, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits CCLR0 to CCLR2 in TCR. 0: TCNT_2 to TCNT_0 operates independently (TCNT presetting /clearing is unrelated to other channels) 1: TCNT_2 to TCNT_0 performs synchronous operation TCNT synchronous presetting/synchronous clearing is possible

8.3.10 Timer Output Master Enable Register (TOER)

TOER is an 8-bit readable/writable register that enables/disables output settings for output pins TIOC4D, TIOC4C, TIOC3D, TIOC4B, TIOC4A, and TIOC3B. These pins do not output correctly if the TOER bits have not been set. Set TOER of CH3 and CH4 prior to setting TIOR of CH3 and CH4.

Bit	Bit Name	Initial value	R/W	Description
7, 6	—	All 1	R	Reserved These bits are always read as 1. Only 1 should be written to these bits.
5	OE4D	0	R/W	Master Enable TIOC4D This bit enables/disables the TIOC4D pin MTU output. 0: MTU output is disabled 1: MTU output is enabled
4	OE4C	0	R/W	Master Enable TIOC4C This bit enables/disables the TIOC4C pin MTU output. 0: MTU output is disabled 1: MTU output is enabled
3	OE3D	0	R/W	Master Enable TIOC3D This bit enables/disables the TIOC3D pin MTU output. 0: MTU output is disabled 1: MTU output is enabled
2	OE4B	0	R/W	Master Enable TIOC4B This bit enables/disables the TIOC4B pin MTU output. 0: MTU output is disabled 1: MTU output is enabled
1	OE4A	0	R/W	Master Enable TIOC4A This bit enables/disables the TIOC4A pin MTU output. 0: MTU output is disabled 1: MTU output is enabled
0	OE3B	0	R/W	Master Enable TIOC3B This bit enables/disables the TIOC3B pin MTU output. 0: MTU output is disabled 1: MTU output is enabled

8.3.11 Timer Output Control Register (TOCR)

TOCR 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	Bit Name	Initial value	R/W	Description
7	—	0	R	Reserved These bits are always read as 0. Only 0 should be written to this bit.
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 to 2	—	All 0	R	Reserved These bits are always read as 0. Only 0 should be written to this bit.
1	OLSN	0	R/W	Output Level Select N This bit selects the reverse phase output level in reset-synchronized PWM mode/complementary PWM mode. See table 8.26
0	OLSP	0	R/W	Output Level Select P This bit selects the positive phase output level in reset-synchronized PWM mode/complementary PWM mode. See table 8.27

Table 8.26 Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLSN	Initial Output	Active Level	Increment Count	Decrement 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 8.27 Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLSP	Initial Output	Active Level	Increment Count	Decrement Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Figure 8.2 shows an example of complementary PWM mode output (1 phase) when OLSN = 1, OLSP = 1.

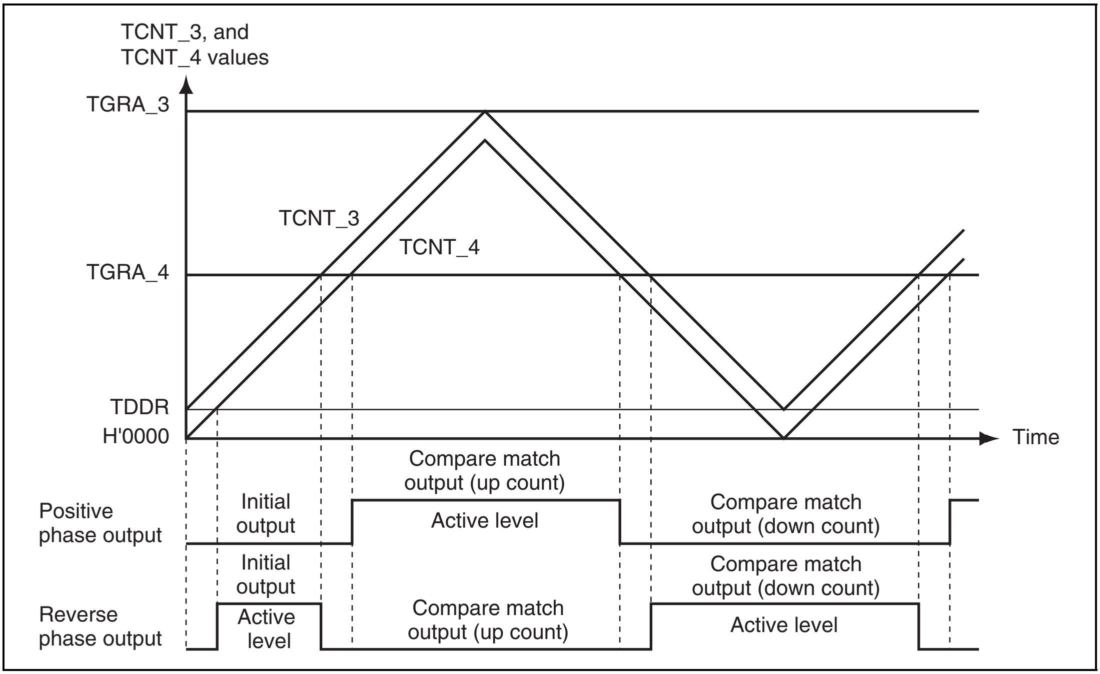


Figure 8.2 Complementary PWM Mode Output Level Example

8.3.12 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	Bit Name	Initial value	R/W	Description
7	—	0	R	Reserved This bit is always read as 1. Only 1 should be written to this bit.
6	BDC	0	R/W	Brushless DC Motor This bit selects whether to make the functions of this register (TGCR) effective or ineffective. 0: Ordinary output 1: Functions of this register are made effective
5	N	0	R/W	Reverse Phase Output (N) Control This bit selects whether the level output or the reset-synchronized PWM/complementary PWM output while the reverse pins (TIOC3D, TIOC4C, and TIOC4D) are on-output. 0: Level output 1: Reset synchronized PWM/complementary PWM output
4	P	0	R/W	Positive Phase Output (P) Control This bit selects whether the level output or the reset-synchronized PWM/complementary PWM output while the positive pin (TIOC3B, TIOC4A, and TIOC4B) are on-output. 0: Level output 1: Reset synchronized PWM/complementary PWM output

Bit	Bit Name	Initial value	R/W	Description
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 MTU/channel 0 TGRA, TGRB, TGRC input capture signals or by writing 0 or 1 to bits 2 to 0 in TGCR. 0: Output switching is carried out by external input (Input sources are channel 0 TGRA, TGRB, TGRC input capture signal) 1: Output switching is carried out by software (TGCR's UF, VF, WF settings).
2 1 0	WF VF UF	0 0 0	R/W R/W R/W	Output Phase Switch 2 to 0 These bits set the positive phase/negative phase output phase on or off state. The setting of these bits is valid only when the FB bit in this register is set to 1. In this case, the setting of bits 2 to 0 is a substitute for external input. See table 8.28.

Table 8.28 Output level Select Function

			Function					
Bit 2	Bit 1	Bit 0	TIOC3B	TIOC4A	TIOC4B	TIOC3D	TIOC4C	TIOC4D
WF	VF	UF	U Phase	V Phase	W Phase	U Phase	V Phase	W Phase
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
		1	ON	OFF	OFF	OFF	OFF	ON
	1	0	OFF	ON	OFF	ON	OFF	OFF
		1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
		1	ON	OFF	OFF	OFF	ON	OFF
	1	0	OFF	OFF	ON	ON	OFF	OFF
		1	OFF	OFF	OFF	OFF	OFF	OFF

8.3.13 Timer Subcounter (TCNTS)

TCNTS is a 16-bit read-only counter that is used only in complementary PWM mode. The initial value is H'0000.

Note: Accessing the TCNTS in 8-bit units is prohibited. Always access in 16-bit units.

8.3.14 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 is H'FFFF.

Note: Accessing the TDDR in 8-bit units is prohibited. Always access in 16-bit units.

8.3.15 Timer Period Data Register (TCDR)

TCDR is a 16-bit register used only in complementary PWM mode. Set half the PWM carrier sync value as the TCDR register value. This register is constantly compared with the TCNTS counter in complementary PWM mode, and when a match occurs, the TCNTS counter switches direction (decrement to increment). The initial value is H'FFFF.

Note: Accessing the TCDR in 8-bit units is prohibited. Always access in 16-bit units.

8.3.16 Timer Period Buffer Register (TCBR)

The timer period buffer register (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 is H'FFFF.

Note: Accessing the TCBR in 8-bit units is prohibited. Always access in 16-bit units.

8.3.17 Bus Master Interface

The timer counters (TCNT), general registers (TGR), timer subcounter (TCNTS), timer period buffer register (TCBR), and timer dead time data register (TDDR), and timer period data register (TCDR) 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.

8.4 Operation

8.4.1 Basic Functions

Each channel has a TCNT and TGR register. TCNT performs up-counting, and is also capable of free-running operation, synchronous counting, and external event counting.

Each TGR can be used as an input capture register or output compare register.

Always set the MTU external pins function using the pin function controller (PFC).

Counter Operation

When one of bits CST0 to CST4 is set to 1 in TSTR, the TCNT counter for the corresponding channel begins counting. TCNT can operate as a free-running counter, periodic counter, for example.

Example of Count Operation Setting Procedure: Figure 8.3 shows an example of the count operation setting procedure.

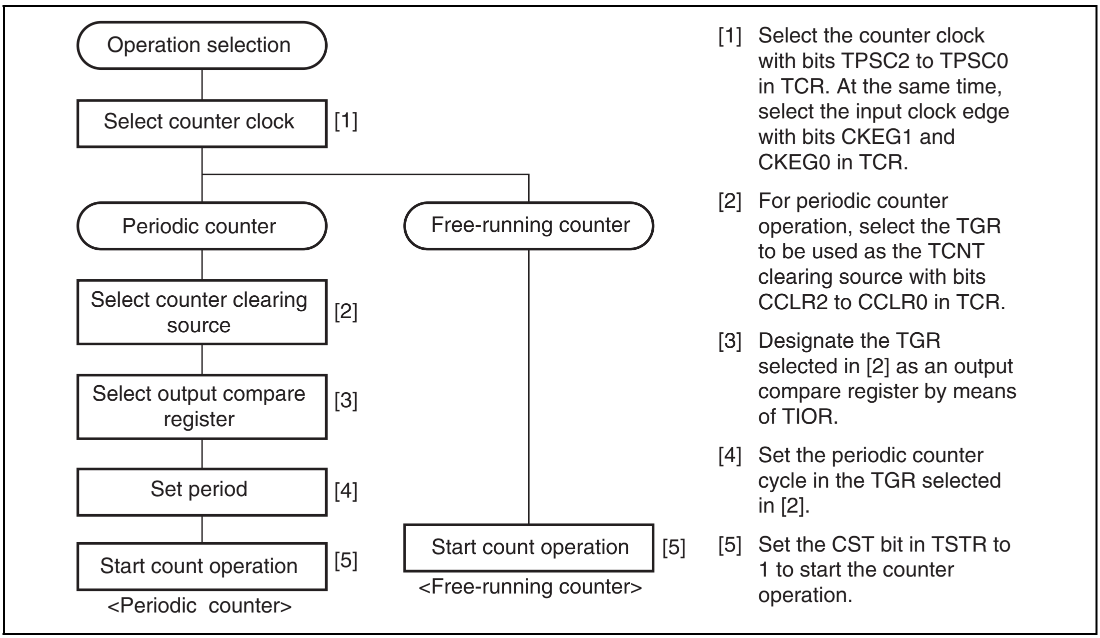


Figure 8.3 Example of Counter Operation Setting Procedure

Free-Running Count Operation and Periodic Count Operation: Immediately after a reset, the MTU'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 MTU requests an interrupt. After overflow, TCNT starts counting up again from H'0000.

Figure 8.4 illustrates free-running counter operation.

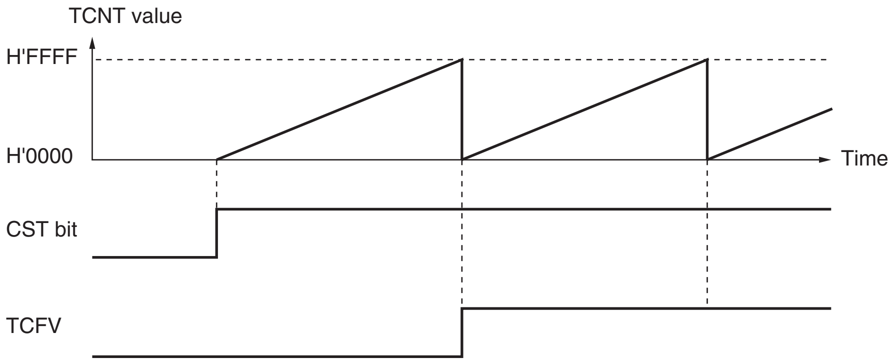


Figure 8.4 Free-Running Counter Operation

When compare match is selected as the TCNT clearing source, the TCNT counter for the relevant channel performs periodic count operation. The TGR register for setting the period is designated as an output compare register, and counter clearing by compare match is selected by means of bits CCLR0 to CCLR2 in TCR. After the settings have been made, TCNT starts up-count operation as a periodic counter when the corresponding bit in TSTR is set to 1. When the count value matches the value in TGR, the TGF bit in TSR is set to 1 and TCNT is cleared to H'0000.

If the value of the corresponding TGIE bit in TIER is 1 at this point, the TPU requests an interrupt. After a compare match, TCNT starts counting up again from H'0000.

Figure 8.5 illustrates periodic counter operation.

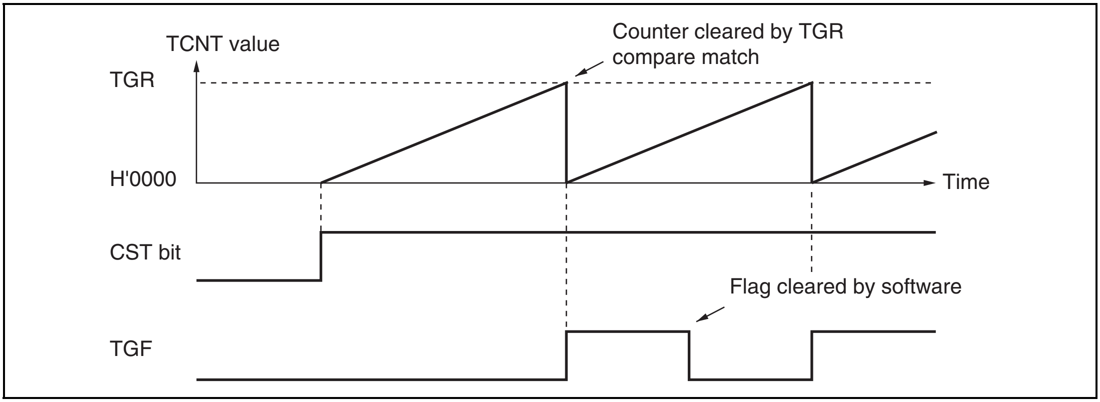


Figure 8.5 Periodic Counter Operation

Waveform Output by Compare Match

The MTU can perform 0, 1, or toggle output from the corresponding output pin using compare match.

Example of Setting Procedure for Waveform Output by Compare Match: Figure 8.6 shows an example of the setting procedure for waveform output by compare match.

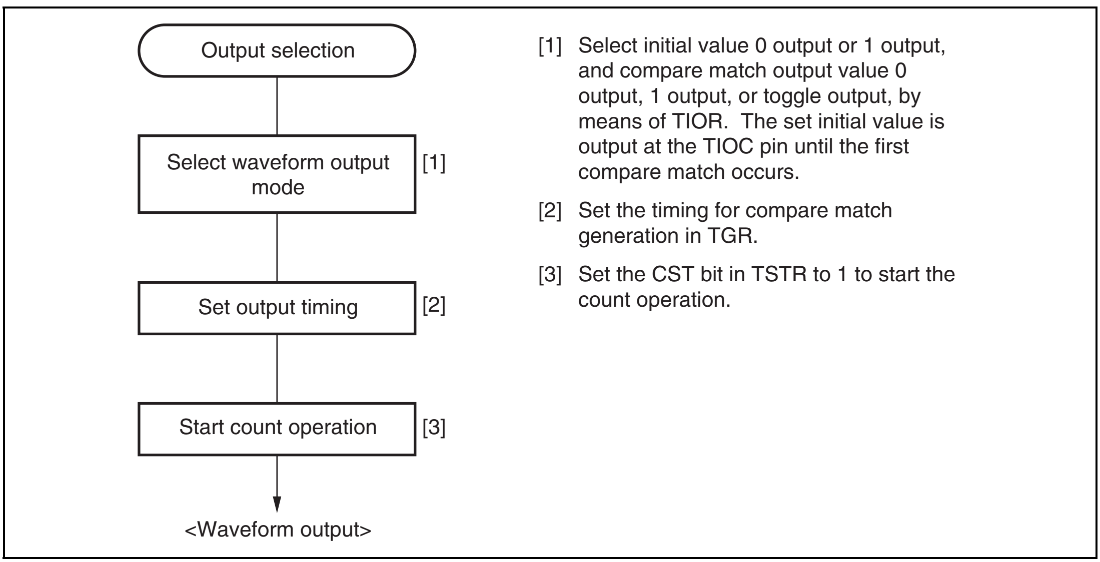


Figure 8.6 Example of Setting Procedure for Waveform Output by Compare Match

Examples of Waveform Output Operation: Figure 8.7 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.

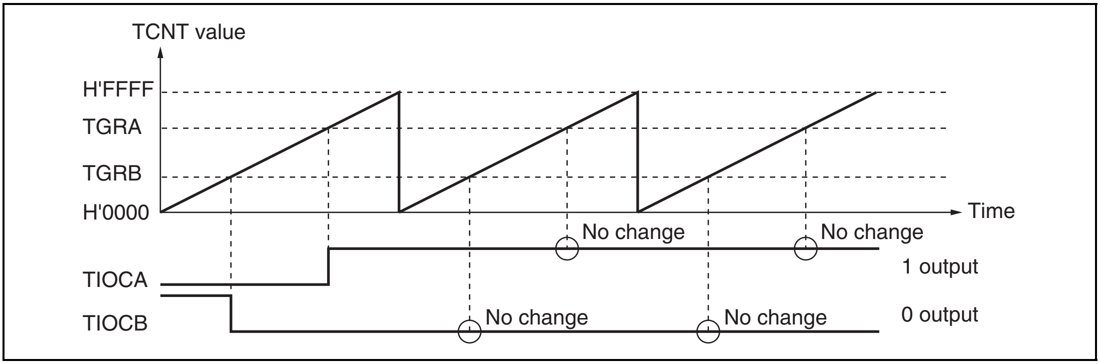


Figure 8.7 Example of 0 Output/1 Output Operation

Figure 8.8 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.

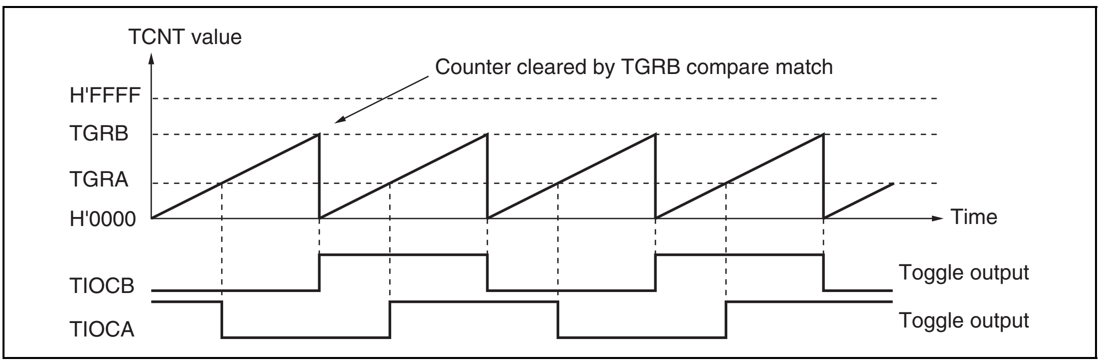


Figure 8.8 Example of Toggle Output Operation

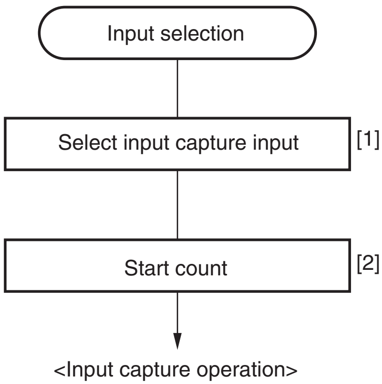
Input Capture Function

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, ϕ /1 should not be selected as the counter input clock used for input capture input. Input capture will not be generated if ϕ /1 is selected.

Example of Input Capture Operation Setting Procedure: Figure 8.9 shows an example of the input capture operation setting procedure.



[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] Set the CST bit in TSTR to 1 to start the count operation.

Figure 8.9 Example of Input Capture Operation Setting Procedure

Example of Input Capture Operation: Figure 8.10 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.

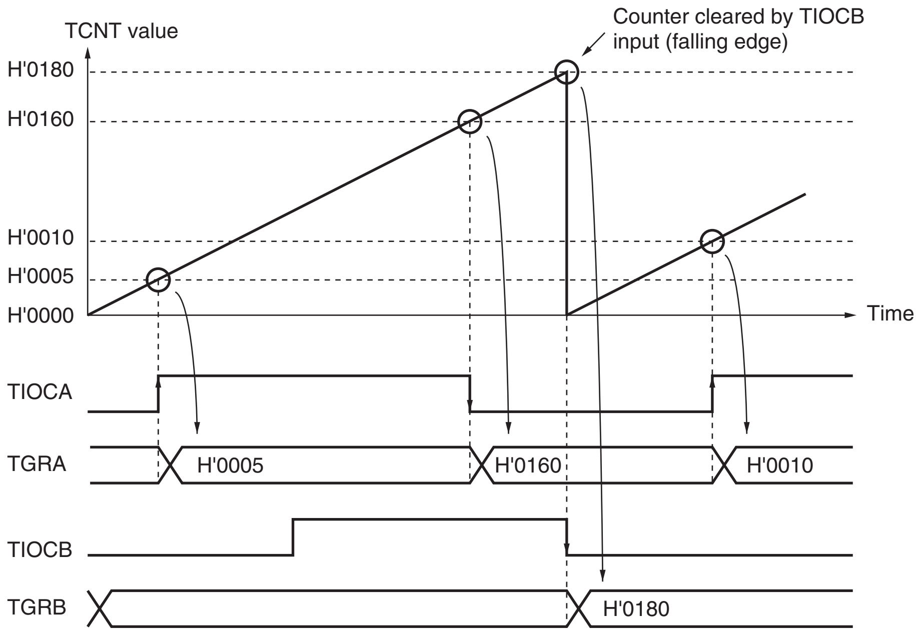


Figure 8.10 Example of Input Capture Operation

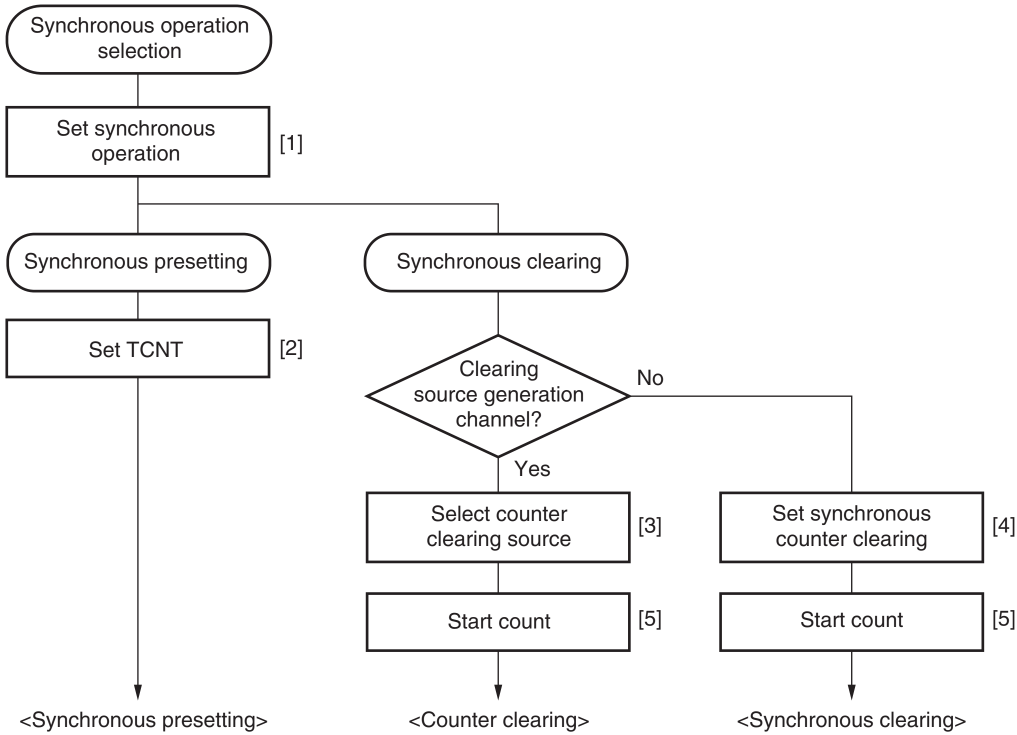
8.4.2 Synchronous Operation

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.

Example of Synchronous Operation Setting Procedure: Figure 8.11 shows an example of the synchronous operation setting procedure.



[1] Set to 1 the SYNC bits in TSYR corresponding to the channels to be designated for synchronous operation.
[2] When the TCNT counter of any of the channels designated for synchronous operation is written to, the same value is simultaneously written to the other TCNT counters.
[3] Use bits CCLR2 to CCLR0 in TCR to specify TCNT clearing by input capture/output compare, etc.
[4] Use bits CCLR2 to CCLR0 in TCR to designate synchronous clearing for the counter clearing source.
[5] Set to 1 the CST bits in TSTR for the relevant channels, to start the count operation.

Figure 8.11 Example of Synchronous Operation Setting Procedure

Example of Synchronous Operation: Figure 8.12 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 8.4.5, PWM Modes.

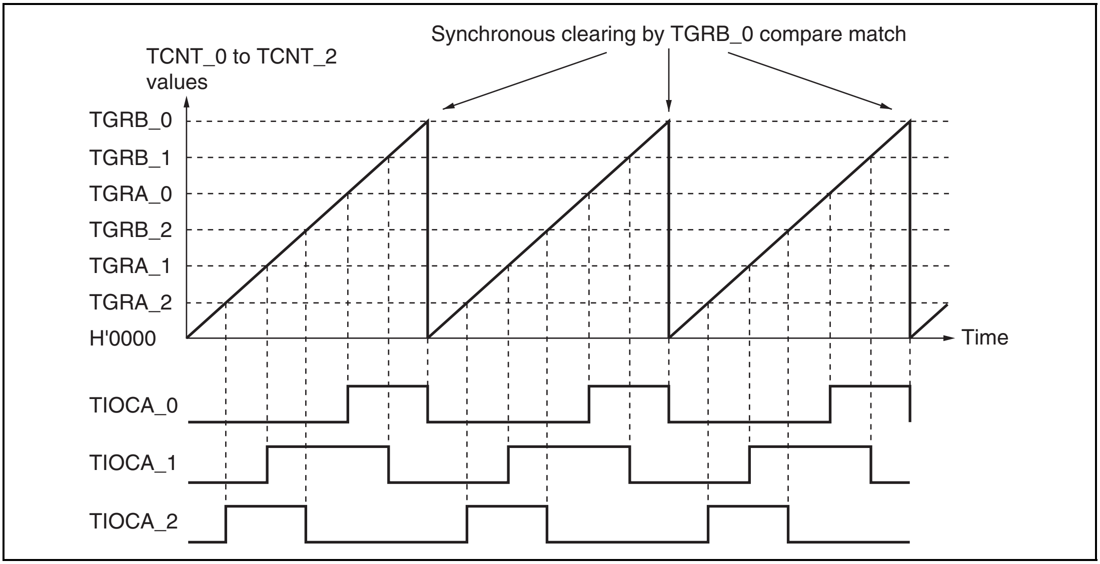


Figure 8.12 Example of Synchronous Operation

8.4.3 Buffer Operation

Buffer operation, provided for channels 0, 3, and 4, enables TGRC and TGRD to be used as buffer registers.

Buffer operation differs depending on whether TGR has been designated as an input capture register or as a compare match register.

Table 8.29 shows the register combinations used in buffer operation.

Table 8.29 Register Combinations in Buffer Operation

Channel	Timer General Register	Buffer Register
0	TGRA_0	TGRC_0
	TGRB_0	TGRD_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 8.13.

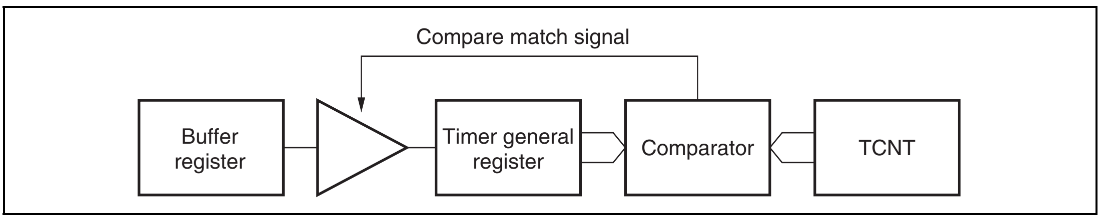


Figure 8.13 Compare Match Buffer Operation

- 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 8.14.

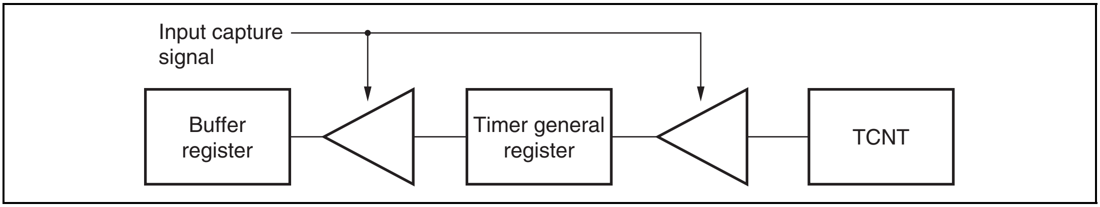


Figure 8.14 Input Capture Buffer Operation

Example of Buffer Operation Setting Procedure: Figure 8.15 shows an example of the buffer operation setting procedure.

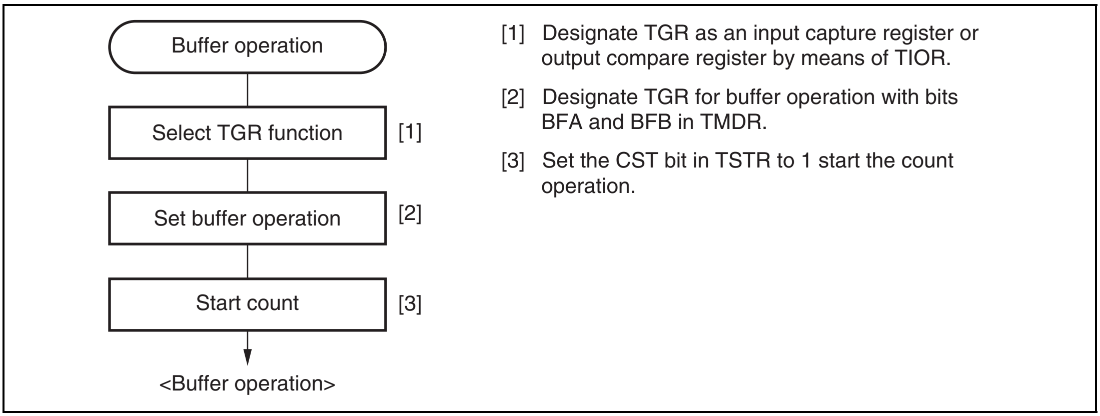


Figure 8.15 Example of Buffer Operation Setting Procedure

Examples of Buffer Operation:

- When TGR is an output compare register

  Figure 8.16 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.

  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 8.4.5, PWM Modes.

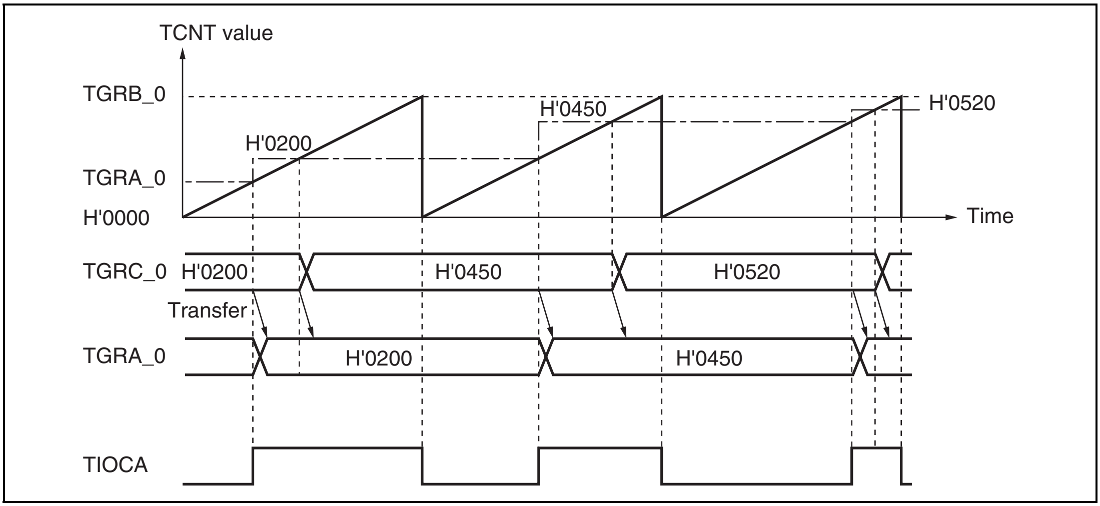


Figure 8.16 Example of Buffer Operation (1)

- When TGR is an input capture register

  Figure 8.17 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.

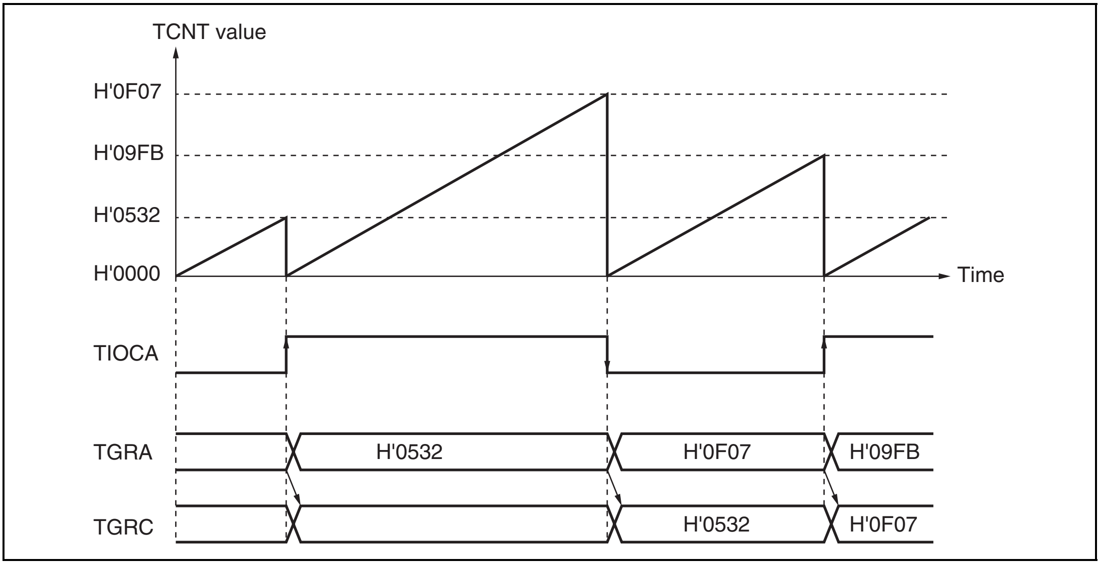


Figure 8.17 Example of Buffer Operation (2)

8.4.4 Cascaded Operation

In cascaded operation, two 16-bit counters for different channels are used together as a 32-bit counter.

This function works by counting the channel 1 counter clock upon overflow/underflow of TCNT_2 as set in bits TPSC0 to TPSC2 in TCR.

Underflow occurs only when the lower 16-bit TCNT is in phase-counting mode.

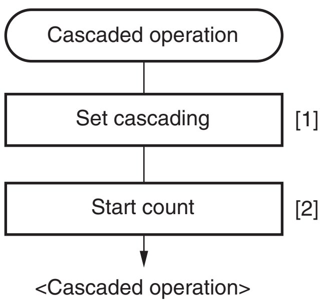
Table 8.30 shows the register combinations used in cascaded operation.

Table 8.30 Cascaded Combinations

Combination	Upper 16 Bits	Lower 16 Bits
Channels 1 and 2	TCNT_1	TCNT_2

Note: When phase counting mode is set for channel 1 or 2, the counter clock setting is invalid and the counters operates independently in phase counting mode.

Example of Cascaded Operation Setting Procedure: Figure 8.18 shows an example of the setting procedure for cascaded operation.



[1] Set bits TPSC2 to TPSC0 in the channel 1 TCR to B'111 to select TCNT_2 overflow/ underflow counting.

[2] Set the CST bit in TSTR for the upper and lower channel to 1 to start the count operation.

Figure 8.18 Cascaded Operation Setting Procedure

Examples of Cascaded Operation: Figure 8.19 illustrates the operation when TCNT_2 overflow/underflow counting has been set for TCNT_1 and phase counting mode has been designated for channel 2.

TCNT_1 is incremented by TCNT_2 overflow and decremented by TCNT_2 underflow.

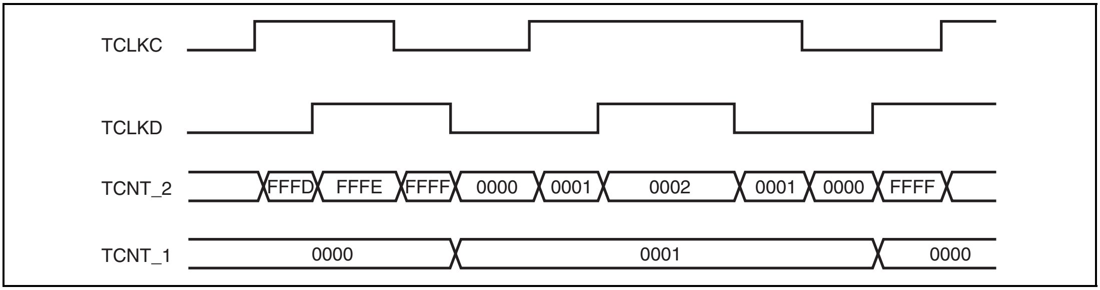


Figure 8.19 Example of Cascaded Operation

8.4.5 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 cycle.

Designating TGR compare match as the counter clearing source enables the period to be set in that register. All channels can be designated for PWM mode independently. Synchronous operation is also possible.

There are two PWM modes, as described below.

- PWM mode 1

  PWM output is generated from the TIOCA and TIOCC pins by pairing TGRA with TGRB and TGRC with TGRD. The output specified by bits IOA0 to IOA3 and IOC0 to IOC3 in TIOR is output from the TIOCA and TIOCC pins at compare matches A and C, and the output specified by bits IOB0 to IOB3 and IOD0 to IOD3 in TIOR is output at compare matches B and D. The initial output value is the value set in TGRA or TGRC. If the set values of paired TGRs are identical, the output value does not change when a compare match occurs.

  In PWM mode 1, a maximum 8-phase PWM output is possible.

- PWM mode 2

  PWM output is generated using one TGR as the cycle register and the others as duty cycle registers. The output specified in TIOR is performed by means of compare matches. Upon counter clearing by a synchronization register compare match, the output value of each pin is the initial value set in TIOR. If the set values of the cycle and duty cycle 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.

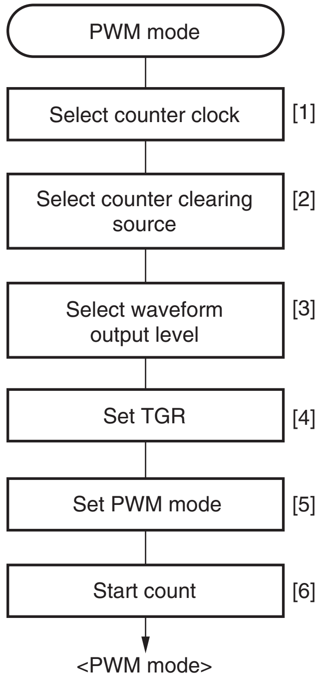
The correspondence between PWM output pins and registers is shown in table 8.31.

Table 8.31 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 8.20 shows an example of the PWM mode setting procedure.



[1] Select the counter clock with bits TPSC2 to TPSC0 in TCR. At the same time, select the input clock edge with bits CKEG1 and CKEG0 in TCR.

[2] Use bits CCLR2 to CCLR0 in TCR to select the TGR to be used as the TCNT clearing source.

[3] Use TIOR to designate the TGR as an output compare register, and select the initial value and output value.

[4] Set the cycle in the TGR selected in [2], and set the duty in the other TGR.

[5] Select the PWM mode with bits MD3 to MD0 in TMDR.

[6] Set the CST bit in TSTR to 1 to start the count operation.

Figure 8.20 Example of PWM Mode Setting Procedure

Examples of PWM Mode Operation: Figure 8.21 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 cycle.

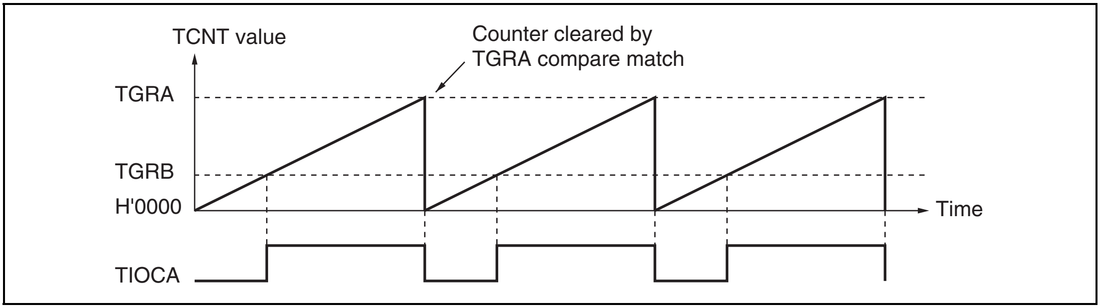


Figure 8.21 Example of PWM Mode Operation (1)

Figure 8.22 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 cycle levels.

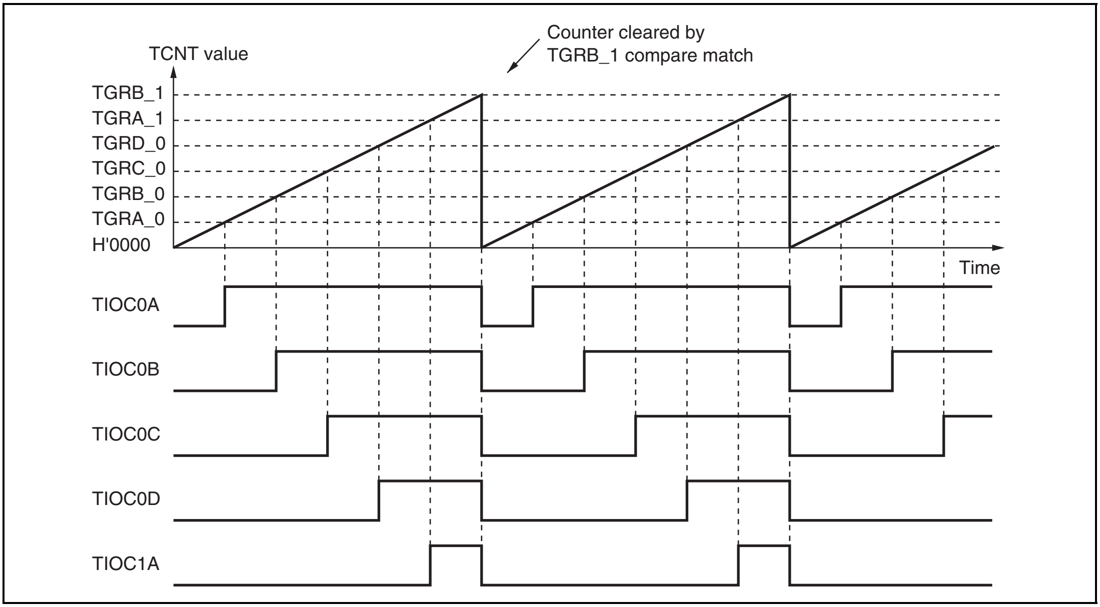


Figure 8.22 Example of PWM Mode Operation (2)

Figure 8.23 shows examples of PWM waveform output with 0% duty cycle and 100% duty cycle in PWM mode.

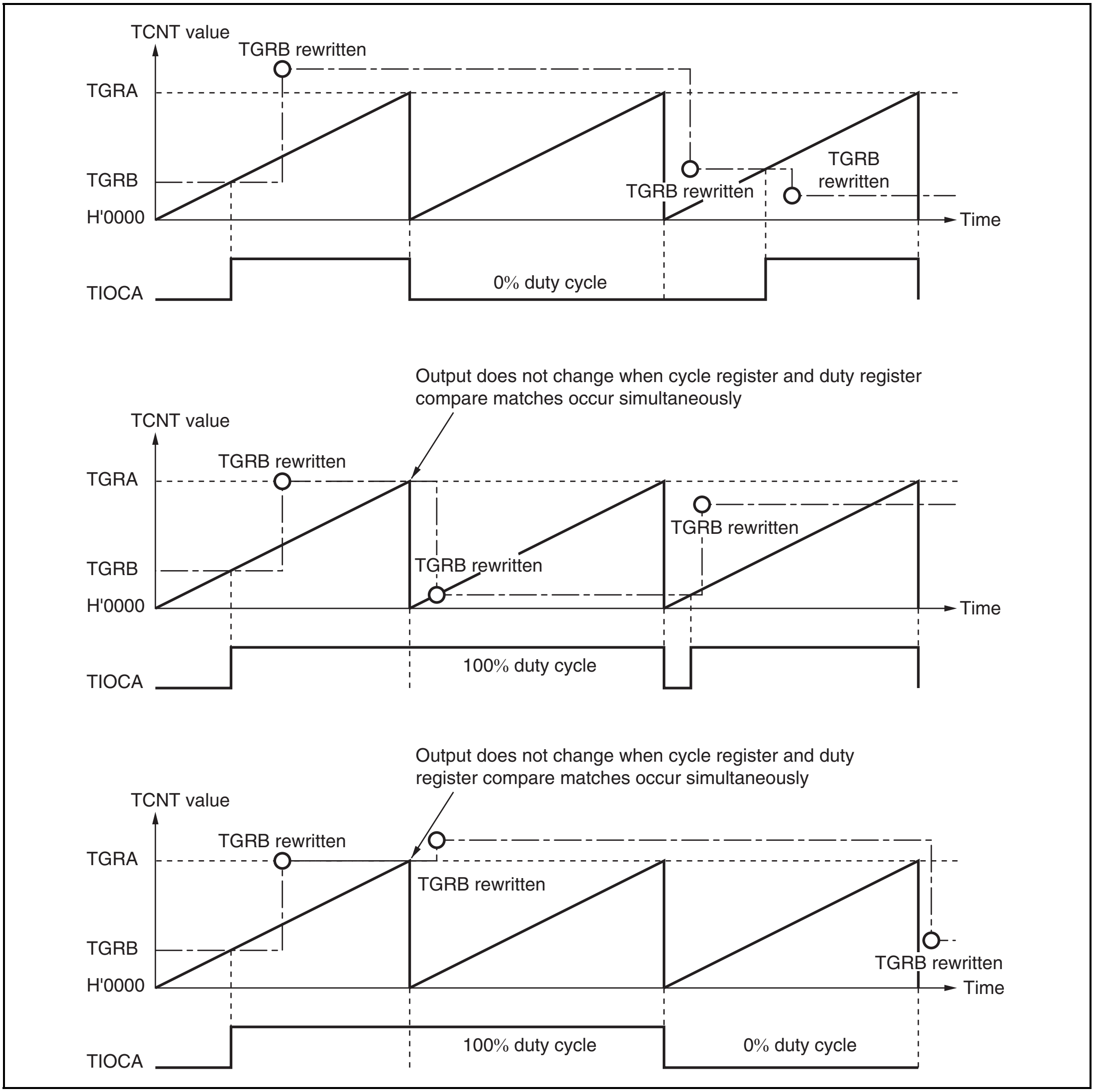


Figure 8.23 Example of PWM Mode Operation (3)

8.4.6 Phase Counting Mode

In phase counting mode, the phase difference between two external clock inputs is detected and TCNT counts up or down accordingly. This mode can be set for channels 1 and 2.

When phase counting mode is set, an external clock is selected as the counter input clock and TCNT operates as an up/down-counter regardless of the setting of bits TPSC0 to TPSC2 and bits CKEG0 and CKEG1 in TCR. However, the functions of bits CCLR0 and CCLR1 in TCR, and of TIOR, TIER, and TGR, are valid, and input capture/compare match and interrupt functions can be used.

This can be used for two-phase encoder pulse input.

If overflow occurs when TCNT is counting up, the TCFV flag in TSR is set; if underflow occurs when TCNT is counting down, the TCFU flag is set.

The TCFD bit in TSR is the count direction flag. Reading the TCFD flag reveals whether TCNT is counting up or down.

Table 8.32 shows the correspondence between external clock pins and channels.

Table 8.32 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 8.24 shows an example of the phase counting mode setting procedure.

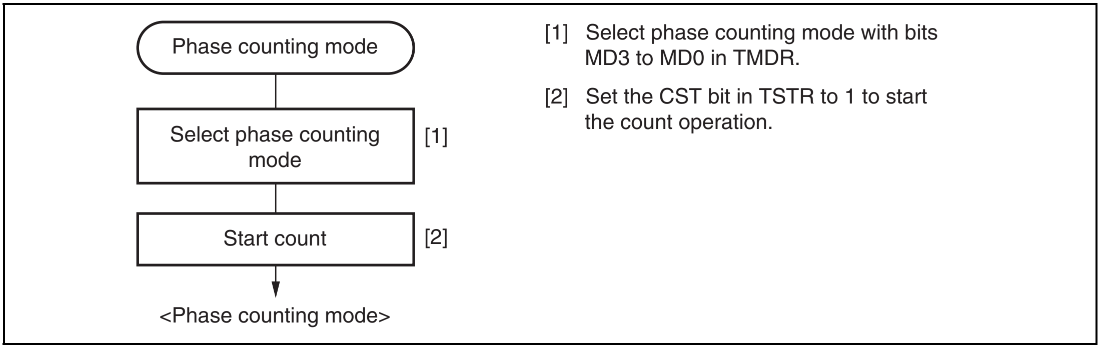


Figure 8.24 Example of Phase Counting Mode Setting Procedure

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.

- Phase counting mode 1

  Figure 8.25 shows an example of phase counting mode 1 operation, and table 8.33 summarizes the TCNT up/down-count conditions.

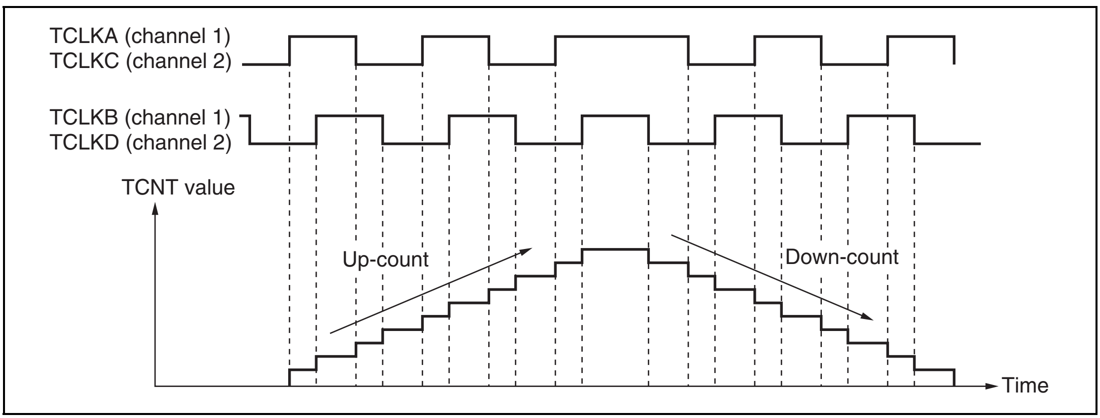


Figure 8.25 Example of Phase Counting Mode 1 Operation

Table 8.33 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

- Phase counting mode 2

  Figure 8.26 shows an example of phase counting mode 2 operation, and table 8.34 summarizes the TCNT up/down-count conditions.

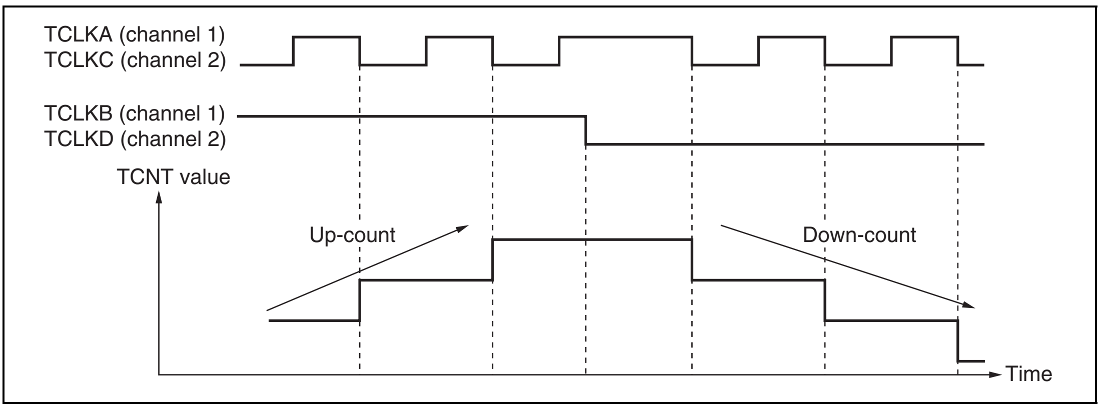


Figure 8.26 Example of Phase Counting Mode 2 Operation

Table 8.34 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	↓	
↑	Low level	
↓	High level	Up-count
High level	↓	Don't care
Low level	↑	
↑	High level	
↓	Low level	Down-count

Legend:

↑ : Rising edge

↓ : Falling edge

- Phase counting mode 3

  Figure 8.27 shows an example of phase counting mode 3 operation, and table 8.35 summarizes the TCNT up/down-count conditions.

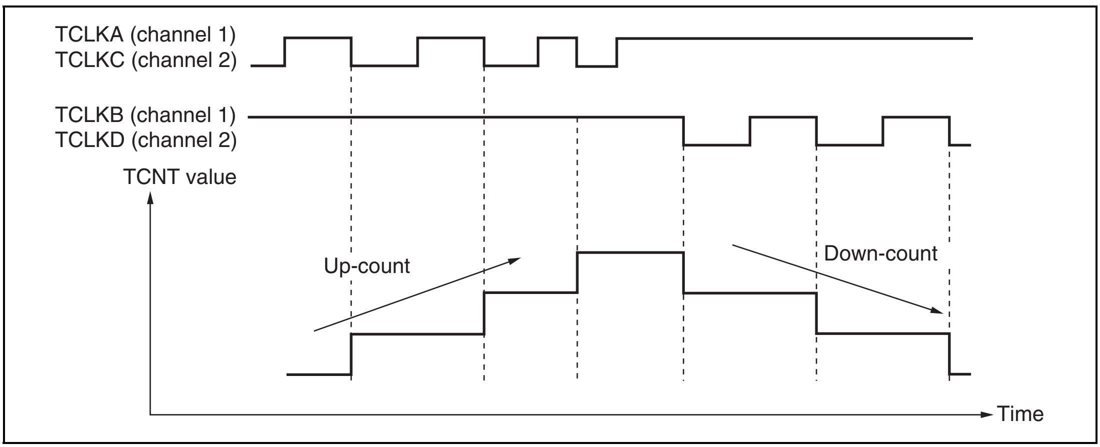


Figure 8.27 Example of Phase Counting Mode 3 Operation

Table 8.35 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	↓	
↑	Low level	
↓	High level	Up-count
High level	↓	Down-count
Low level	↑	Don't care
↑	High level	
↓	Low level	

Legend:

↑ : Rising edge

↓ : Falling edge

- Phase counting mode 4

  Figure 8.28 shows an example of phase counting mode 4 operation, and table 8.36 summarizes the TCNT up/down-count conditions.

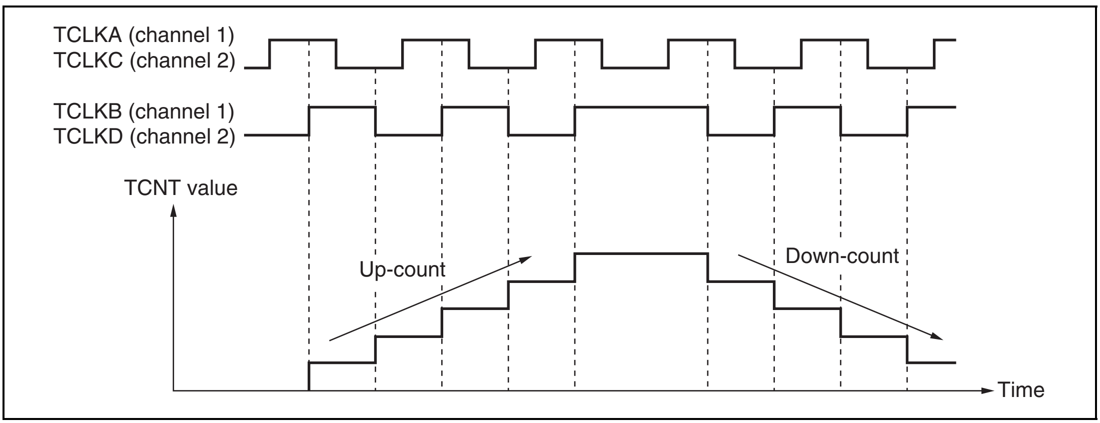


Figure 8.28 Example of Phase Counting Mode 4 Operation

Table 8.36 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 8.29 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.

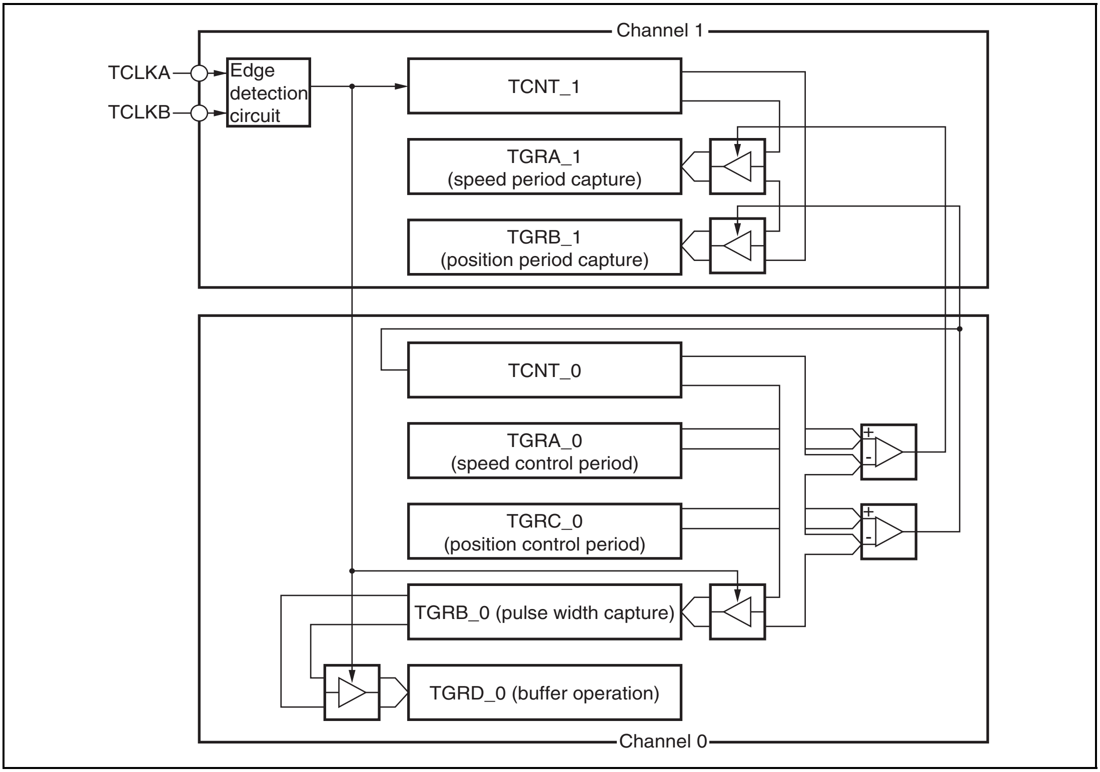


Figure 8.29 Phase Counting Mode Application Example

8.4.7 Reset-Synchronized PWM Mode

In the reset-synchronized PWM mode, three-phase output of positive and negative PWM waveforms that share a common wave transition point can be obtained by combining channels 3 and 4.

When set for reset-synchronized PWM mode, the TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, and TIOC4D pins function as PWM output pins and TCNT3 functions as an upcounter.

Table 8.37 shows the PWM output pins used. Table 8.38 shows the settings of the registers.

Table 8.37 Output Pins for Reset-Synchronized PWM Mode

Channel	Output Pin	Description
3	TIOC3B	PWM output pin 1
	TIOC3D	PWM output pin 1' (negative-phase waveform of PWM output 1)
4	TIOC4A	PWM output pin 2
	TIOC4C	PWM output pin 2' (negative-phase waveform of PWM output 2)
	TIOC4B	PWM output pin 3
	TIOC4D	PWM output pin 3' (negative-phase waveform of PWM output 3)

Table 8.38 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 8.30 shows an example of procedure for selecting the reset synchronized PWM mode.

1. Clear the CST3 and CST4 bits in the TSTR to 0 to halt the counting of TCNT. The reset-synchronized PWM mode must be set up while TCNT_3 and TCNT_4 are halted.
2. Set bits TPSC2 to TPSC0 and CKEG1 and CKEG0 in the TCR_3 to select the counter clock and clock edge for channel 3. Set bits CCLR2 to CCLR0 in the TCR_3 to select TGRA compare-match as a counter clear source.
3. When performing brushless DC motor control, set bit BDC in the timer gate control register (TGCR) and set the feedback signal input source and output chopping or gate signal direct output.
4. Reset TCNT_3 and TCNT_4 to H'0000.
5. TGRA_3 is the period register. Set the waveform period value in TGRA_3. Set the transition timing of the PWM output waveforms in TGRB_3, TGRA_4, and TGRB_4. Set times within the compare-match range of TCNT_3.

   X ≤ TGRA_3 (X: set value).
6. Select enabling/disabling of toggle output synchronized with the PMW cycle using bit PSYE in the timer output control register (TOCR), and set the PWM output level with bits OLSP and OLSN.
7. Set bits MD3 to MD0 in TMDR_3 to B'1000 to select the reset-synchronized PWM mode. TIOC3A, TIOC3B, TIOC3D, TIOC4A, TIOC4B, TIOC4C and TIOC4D function as PWM output pins*. Do not set to TMDR_4.
8. Set the enabling/disabling of the PWM waveform output pin in TOER.
9. Set the CST3 bit in the TSTR to 1 to start the count operation.

Notes: The output waveform starts to toggle operation at the point of TCNT_3 = TGRA_3 = X by setting X = TGRA, i.e., cycle = duty cycle.

* PFC registers should be specified before this procedure.

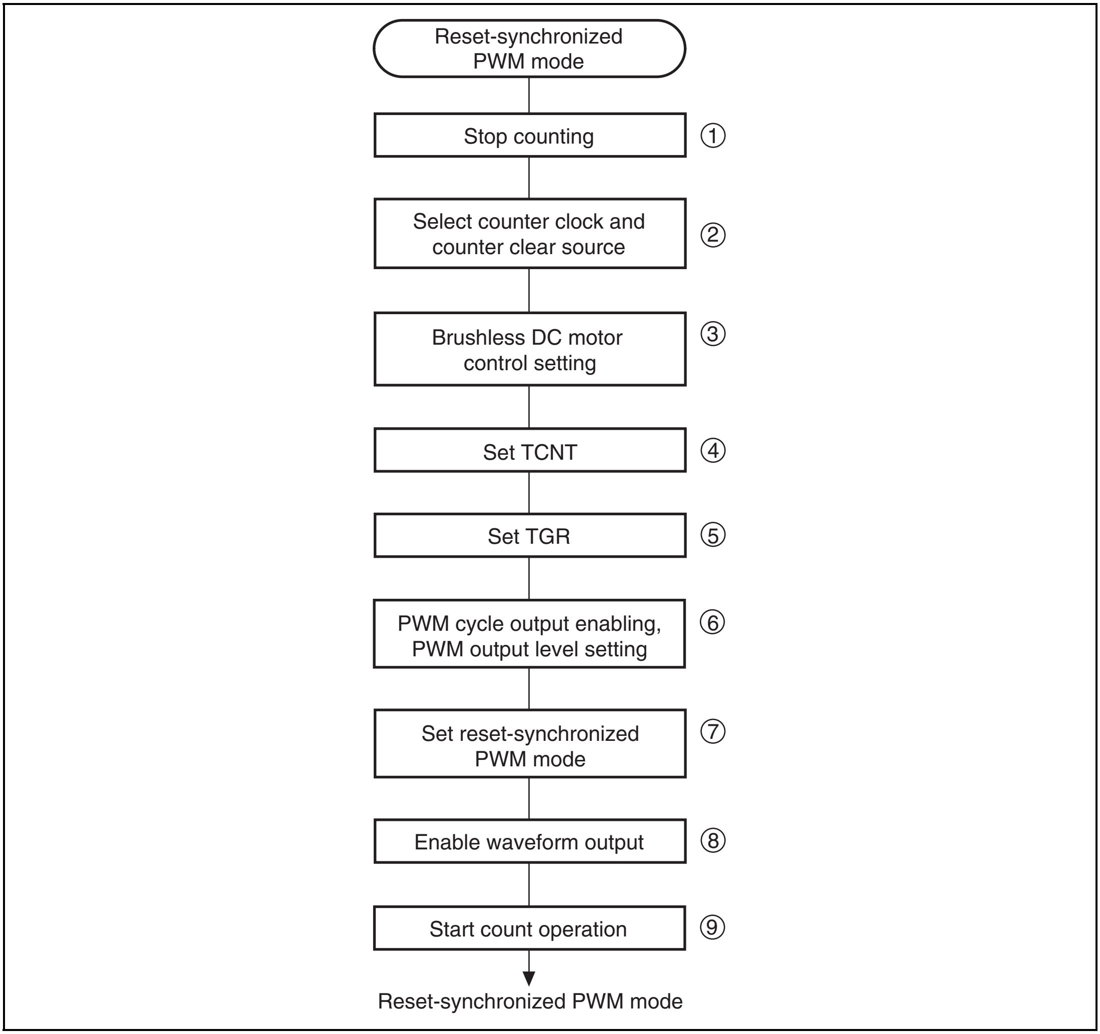


Figure 8.30 Procedure for Selecting the Reset-Synchronized PWM Mode

Reset-Synchronized PWM Mode Operation: Figure 8.31 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 counting up 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.

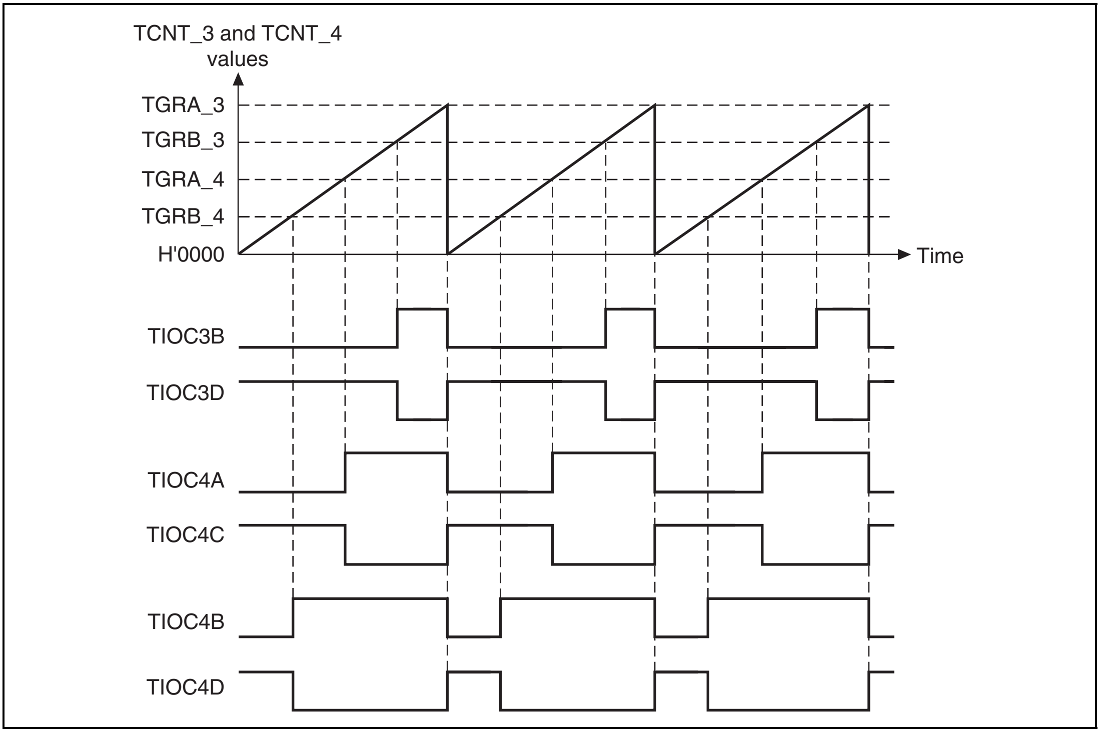


Figure 8.31 Reset-Synchronized PWM Mode Operation Example (When the TOCR's OLSN = 1 and OLSP = 1)

8.4.8 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.

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 increment/decrement counters.

Table 8.39 shows the PWM output pins used. Table 8.40 shows the settings of the registers used.

A function to directly cut off the PWM output by using an external signal is supported as a port function.

Table 8.39 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)
4	TIOC4A	PWM output pin 2
	TIOC4C	PWM output pin 2' (non-overlapping negative-phase waveform of PWM output 2)
	TIOC4B	PWM output pin 3
	TIOC4D	PWM output pin 3' (non-overlapping negative-phase waveform of PWM output 3)

Note: * Avoid setting the TIOC3C pin as a timer I/O pin in the complementary PWM mode.

Table 8.40 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 BSC/BCR1 setting*
	TGRA_3	Set TCNT_3 upper limit value (1/2 carrier cycle + dead time)	Maskable by BSC/BCR1 setting*
	TGRB_3	PWM output 1 compare register	Maskable by BSC/BCR1 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 BSC/BCR1 setting*
	TGRA_4	PWM output 2 compare register	Maskable by BSC/BCR1 setting*
	TGRB_4	PWM output 3 compare register	Maskable by BSC/BCR1 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 BSC/BCR1 setting*
Timer cycle data register (TCDR)		Set TCNT_4 upper limit value (1/2 carrier cycle)	Maskable by BSC/BCR1 setting*
Timer cycle buffer register (TCBR)		TCDR buffer register	Always readable/writable
Subcounter (TCNTS)		Subcounter for dead time generation	Read-only
Temporary register 1 (TEMP1)		PWM output 1/TGRB_3 temporary register	Not readable/writable
Temporary register 2 (TEMP2)		PWM output 2/TGRA_4 temporary register	Not readable/writable
Temporary register 3 (TEMP3)		PWM output 3/TGRB_4 temporary register	Not readable/writable

Note: * Access can be enabled or disabled according to the setting of bit 13 (MTURWE) in BSC/BCR1 (bus controller/bus control register 1).

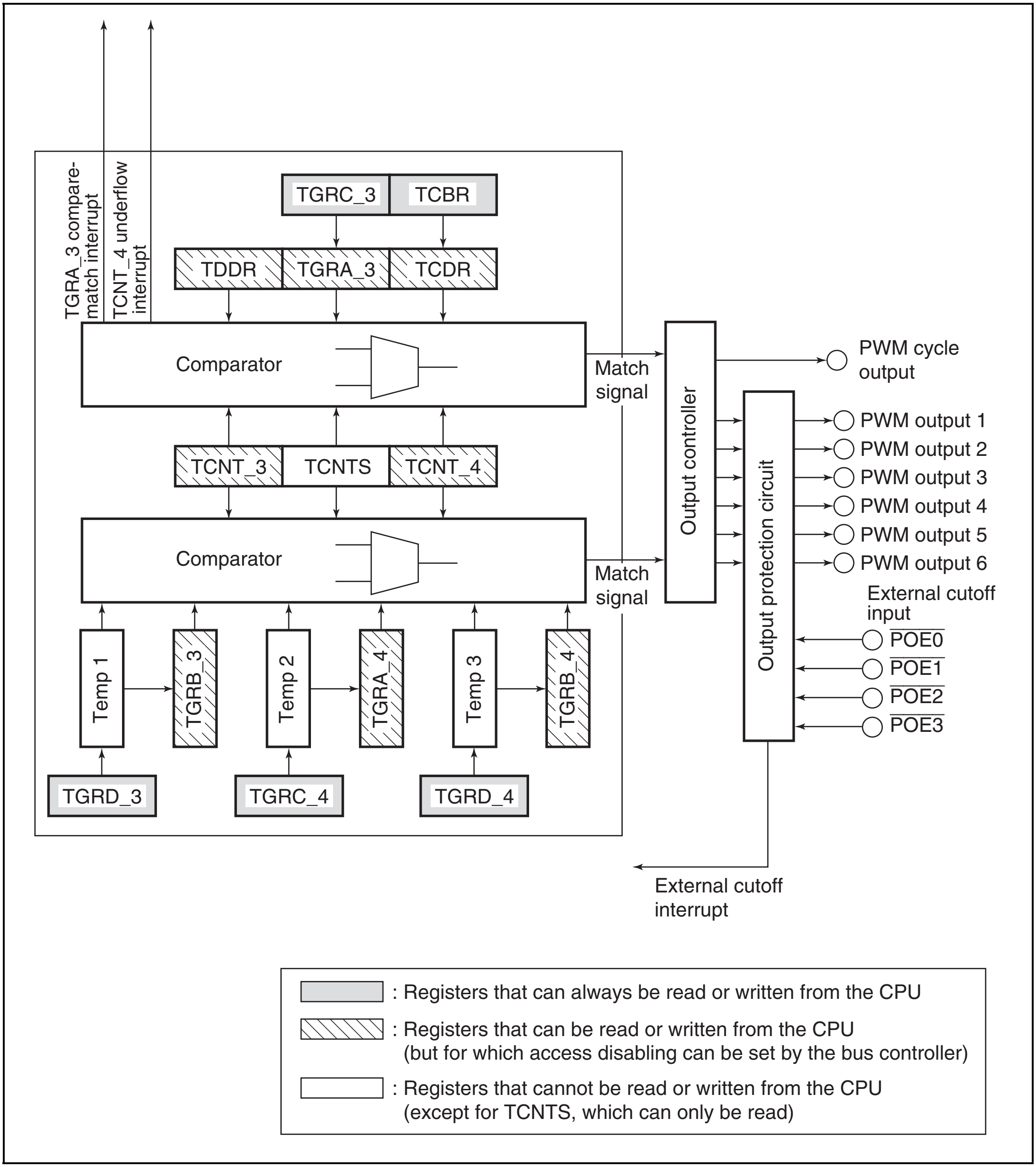


Figure 8.32 Block Diagram of Channels 3 and 4 in Complementary PWM Mode

Example of Complementary PWM Mode Setting Procedure: An example of the complementary PWM mode setting procedure is shown in figure 8.33.

1. Clear bits CST3 and CST4 in the timer start register (TSTR) to 0, and halt timer counter (TCNT) operation. Perform complementary PWM mode setting when TCNT_3 and TCNT_4 are stopped.
2. Set the same counter clock and clock edge for channels 3 and 4 with bits TPSC2 to TPSC0 and bits CKEG1 and CKEG0 in the timer control register (TCR). Use bits CCLR2 to CCLR0 to set synchronous clearing only when restarting by a synchronous clear from another channel during complementary PWM mode operation.
3. When performing brushless DC motor control, set bit BDC in the timer gate control register (TGCR) and set the feedback signal input source and output chopping or gate signal direct output.
4. Set the dead time in TCNT_3. Set TCNT_4 to H'0000.
5. Set only when restarting by a synchronous clear from another channel during complementary PWM mode operation. In this case, synchronize the channel generating the synchronous clear with channels 3 and 4 using the timer synchro register (TSYR).
6. Set the output PWM duty cycle in the duty cycle 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. Set the dead time in the dead time register (TDDR), 1/2 the carrier cycle in the carrier cycle data register (TCDR) and carrier cycle buffer register (TCBR), and 1/2 the carrier cycle plus the dead time in TGRA_3 and TGRC_3.
8. Select enabling/disabling of toggle output synchronized with the PWM cycle using bit PSYE in the timer output control register (TOCR), and set the PWM output level with bits OLSP and OLSN.
9. Select complementary PWM mode in timer mode register 3 (TMDR_3). Pins TIOC3A, TIOC3B, TIOC3D, TIOC4A, TIOC4B, TIOC4C, and TIOC4D function as output pins*. Do not set in TMDR_4.
10. Set enabling/disabling of PWM waveform output pin output in the timer output master enable register (TOER).
11. Set bits CST3 and CST4 in TSTR to 1 simultaneously to start the count operation.

Note: * PFC registers should be specified before this procedure.

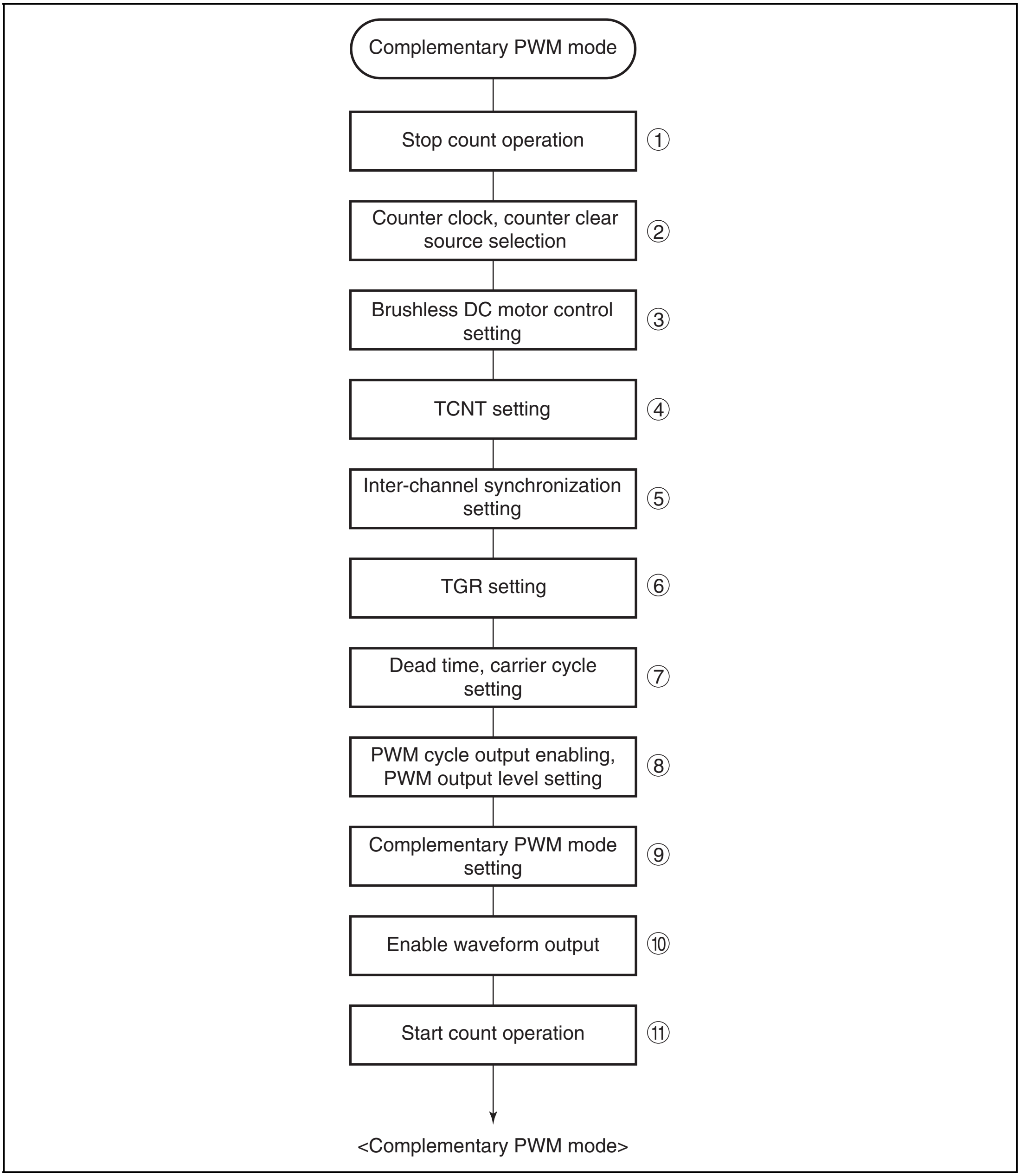


Figure 8.33 Example of Complementary PWM Mode Setting Procedure

Outline of Complementary PWM Mode Operation

In complementary PWM mode, 6-phase PWM output is possible. Figure 8.34 illustrates counter operation in complementary PWM mode, and figure 8.35 shows an example of complementary PWM mode operation.

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 TCNT_3 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, TCNT_4 counts up in synchronization with TCNT_3, and switches to down-counting when it matches TCDR. On reaching H'0000, TCNT_4 switches to up-counting, and the operation is repeated in this way.

TCNTS is a read-only counter. It need not be initialized.

When TCNT_3 matches TCDR during TCNT_3 and TCNT_4 up/down-counting, down-counting is started, and when TCNTS matches TCDR, the operation switches to up-counting. When TCNTS matches TGRA_3, it is cleared to H'0000.

When TCNT_4 matches TDDR during TCNT_3 and TCNT_4 down-counting, up-counting is started, and when TCNTS matches TDDR, the operation switches to down-counting. When TCNTS reaches H'0000, it is set with the value in TGRA_3.

TCNTS is compared with the compare register and temporary register in which the PWM duty cycle is set during the count operation only.

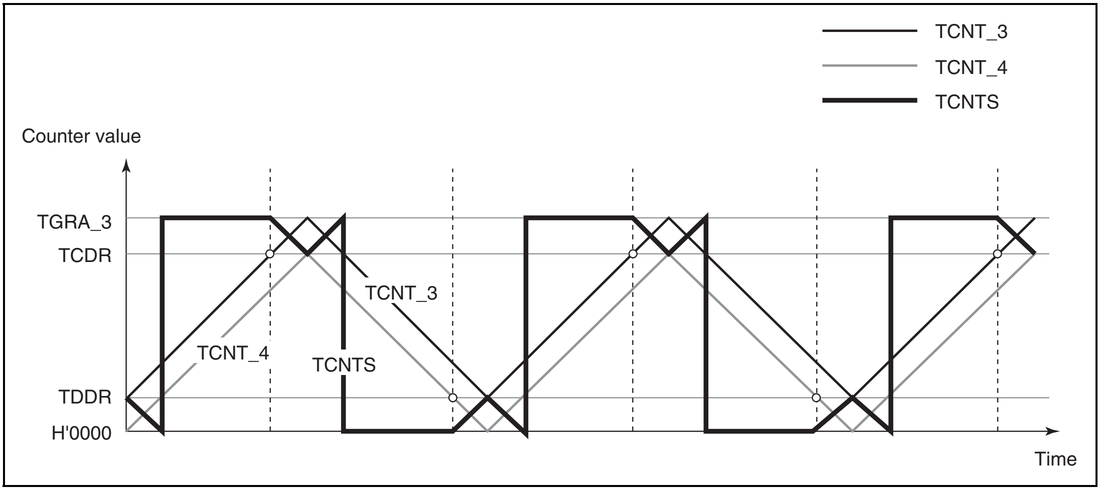


Figure 8.34 Complementary PWM Mode Counter Operation

Register Operation: In complementary PWM mode, nine registers are used, comprising compare registers, buffer registers, and temporary registers. Figure 8.35 shows an example of complementary PWM mode operation.

The registers which are constantly compared with the counters to perform PWM output are TGRB_3, TGRA_4, and TGRB_4. When these registers match the counter, the value set in bits OLSN and OLSP in the timer output control register (TOCR) is output.

The buffer registers for these compare registers are TGRD_3, TGRC_4, and TGRD_4.

Between a buffer register and compare register there is a temporary register. The temporary registers cannot be accessed by the CPU.

Data in a compare register is changed by writing the new data to the corresponding buffer register. The buffer registers can be read or written at any time.

The data written to a buffer register is constantly transferred to the temporary register in the Ta interval. Data is not transferred to the temporary register in the Tb interval. Data written to a buffer register in this interval is transferred to the temporary register at the end of the Tb interval.

The value transferred to a temporary register is transferred to the compare register when TCNTS for which the Tb interval ends matches TGRA_3 when counting up, or H'0000 when counting down. The timing for transfer from the temporary register to the compare register can be selected with bits MD3 to MD0 in the timer mode register (TMDR). Figure 8.35 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 8.35) 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.

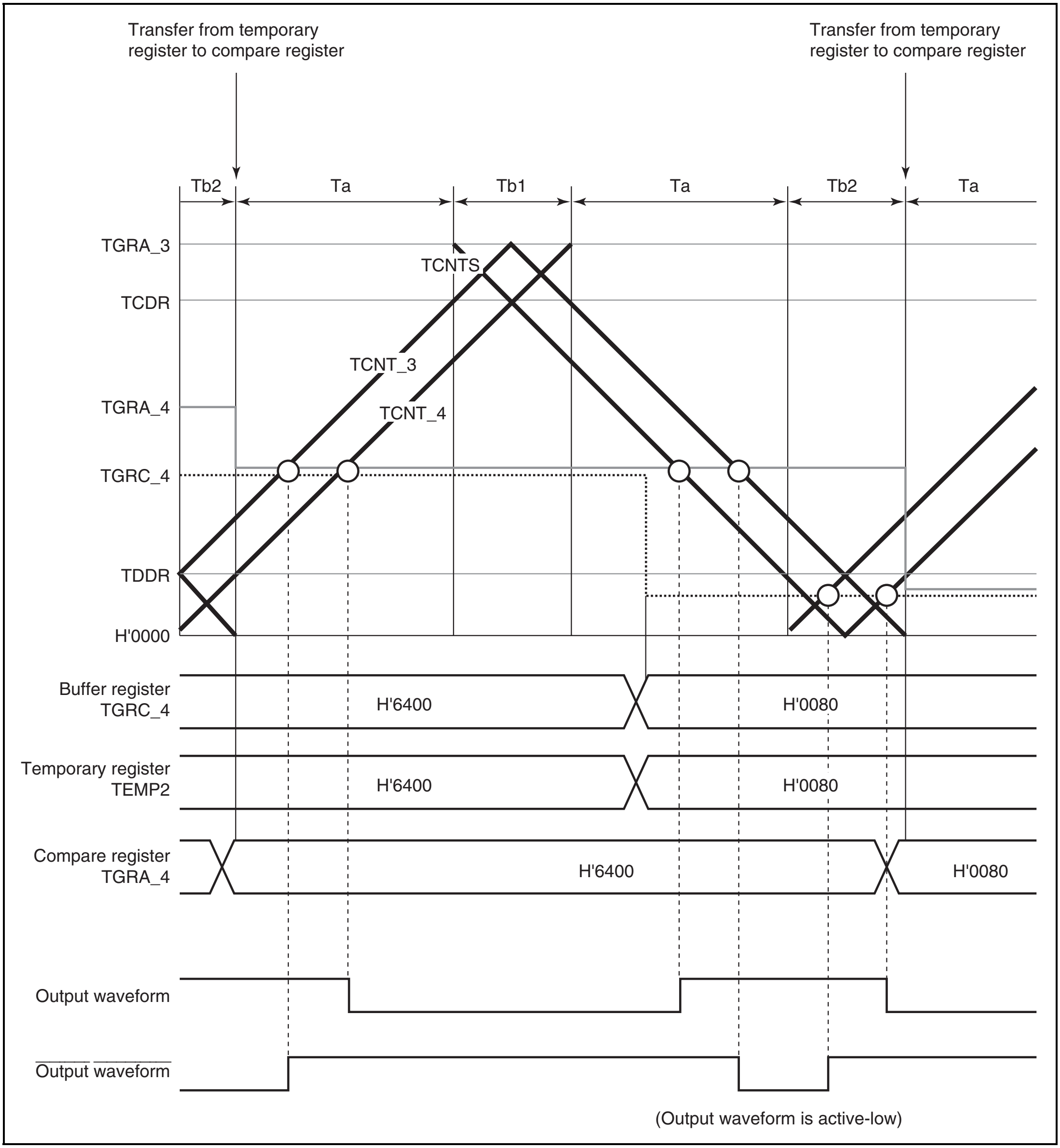


Figure 8.35 Example of Complementary PWM Mode Operation

Initialization: In complementary PWM mode, there are six registers that must be initialized.

Before setting complementary PWM mode with bits MD3 to MD0 in the timer mode register (TMDR), the following initial register values must be set.

TGRC_3 operates as the buffer register for TGRA_3, and should be set with 1/2 the PWM carrier cycle + dead time Td. The timer cycle buffer register (TCBR) operates as the buffer register for the timer cycle data register (TCDR), and should be set with 1/2 the PWM carrier cycle. Set dead time Td in the timer dead time data register (TDDR).

Set the respective initial PWM duty cycle 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 8.41 Registers and Counters Requiring Initialization

Register/Counter	Set Value
TGRC_3	1/2 PWM carrier cycle + dead time Td
TDDR	Dead time Td
TCBR	1/2 PWM carrier cycle
TGRD_3, TGRC_4, TGRD_4	Initial PWM duty cycle value for each phase
TCNT_4	H'0000

Note: The TGRC_3 set value must be the sum of 1/2 the PWM carrier cycle set in TCBR and dead time Td set in TDDR.

PWM Output Level Setting: In complementary PWM mode, the PWM pulse output level is set with bits OLSN and OLSP in the timer output control register (TOCR).

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.

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.

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:

TGRA_3 set value = TCDR set value + TDDR set value

The TGRA_3 and TCDR settings are made by setting the values in buffer registers TGRC_3 and TCBR. The values set in TGRC_3 and TCBR are transferred simultaneously to TGRA_3 and TCDR in accordance with the transfer timing selected with bits MD3 to MD0 in the timer mode register (TMDR).

The updated PWM cycle is reflected from the next cycle when the data update is performed at the crest, and from the current cycle when performed in the trough. Figure 8.36 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.

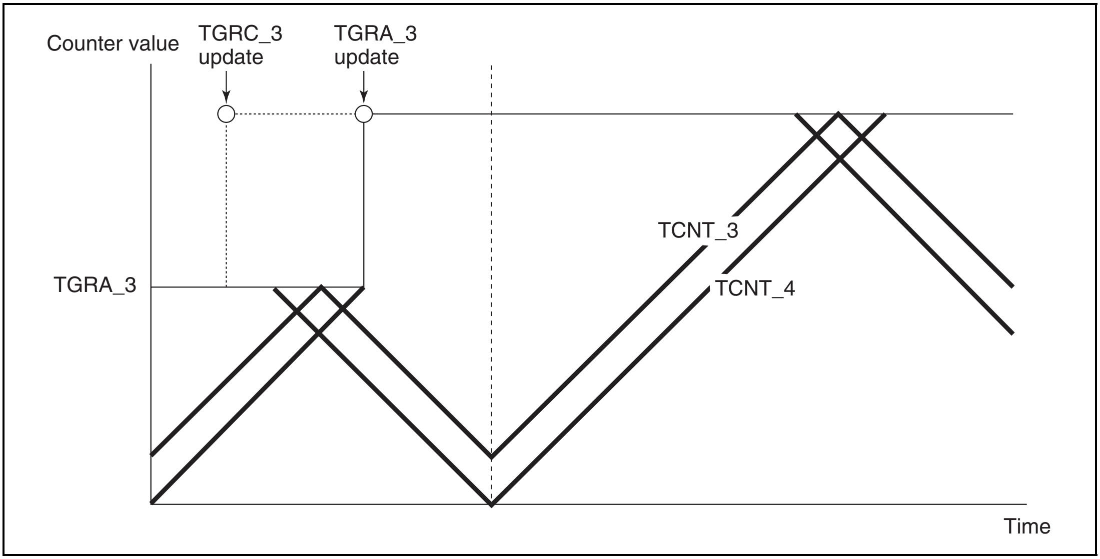


Figure 8.36 Example of PWM Cycle Updating

Register Data Updating: In complementary PWM mode, the buffer register is used to update the data in a compare register. The update data can be written to the buffer register at any time. There are five PWM duty cycle and carrier cycle registers that have buffer registers and can be updated during operation.

There is a temporary register between each of these registers and its buffer register. When subcounter TCNTS is not counting, if buffer register data is updated, the temporary register value is also rewritten. Transfer is not performed from buffer registers to temporary registers when TCNTS is counting; in this case, the value written to a buffer register is transferred after TCNTS halts.

The temporary register value is transferred to the compare register at the data update timing set with bits MD3 to MD0 in the timer mode register (TMDR). Figure 8.37 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.

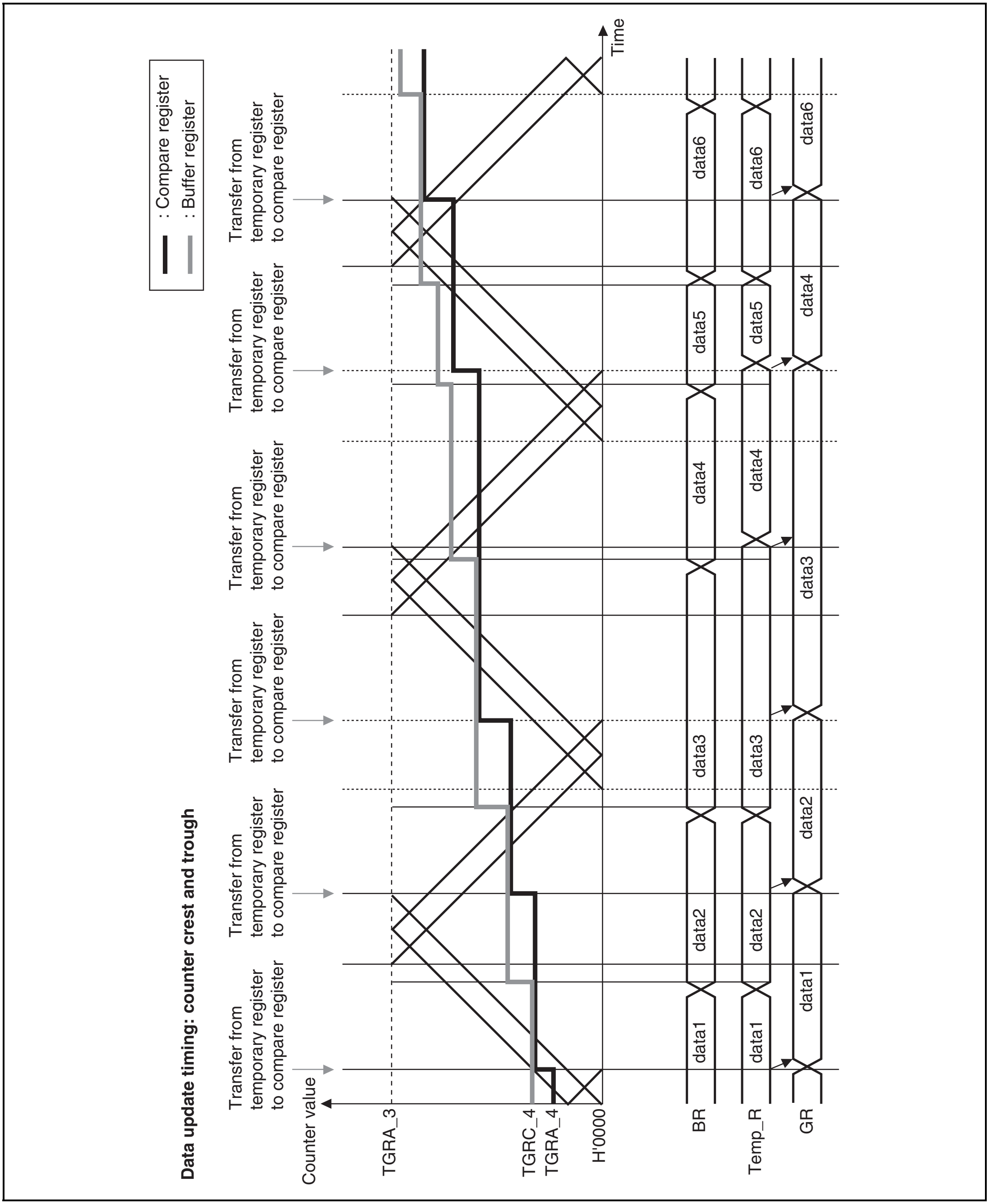


Figure 8.37 Example of Data Update in Complementary PWM Mode

Initial Output in Complementary PWM Mode: In complementary PWM mode, the initial output is determined by the setting of bits OLSN and OLSP in the timer output control register (TOCR).

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 8.38 shows an example of the initial output in complementary PWM mode.

An example of the waveform when the initial PWM duty cycle value is smaller than the TDDR value is shown in figure 8.39.

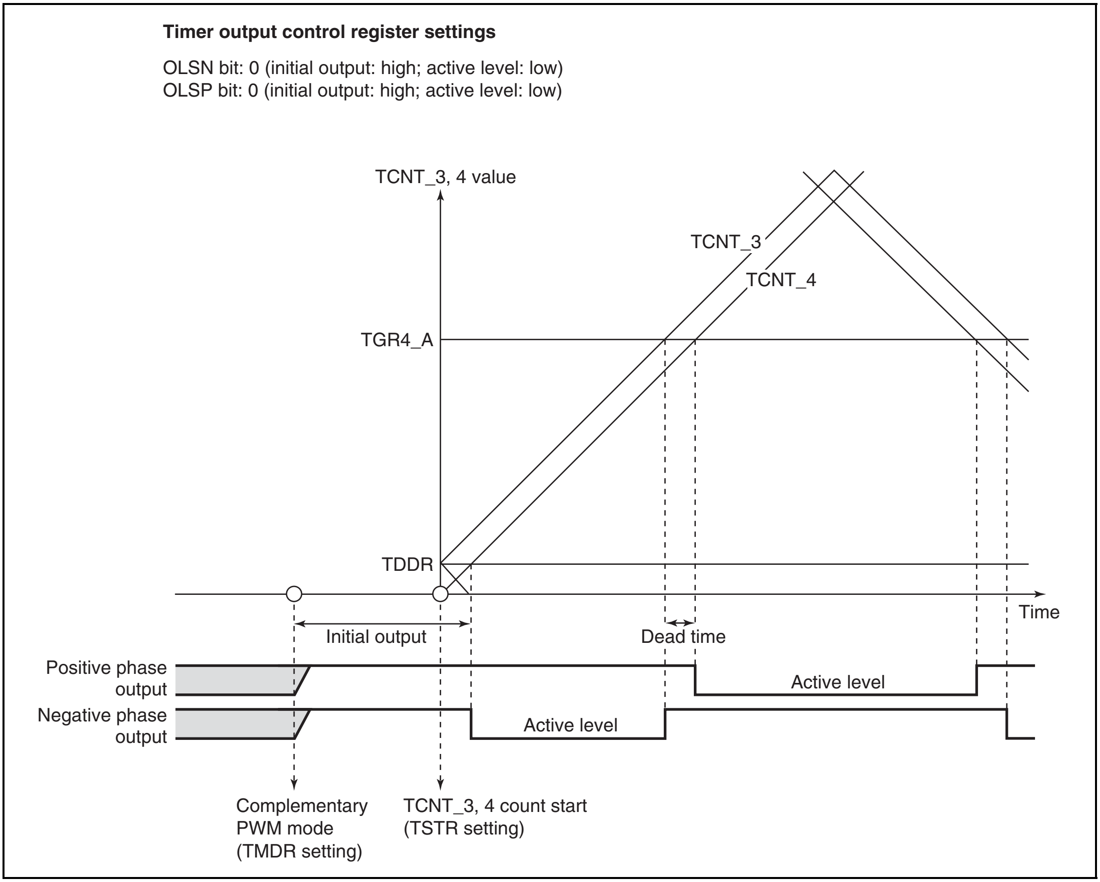


Figure 8.38 Example of Initial Output in Complementary PWM Mode (1)

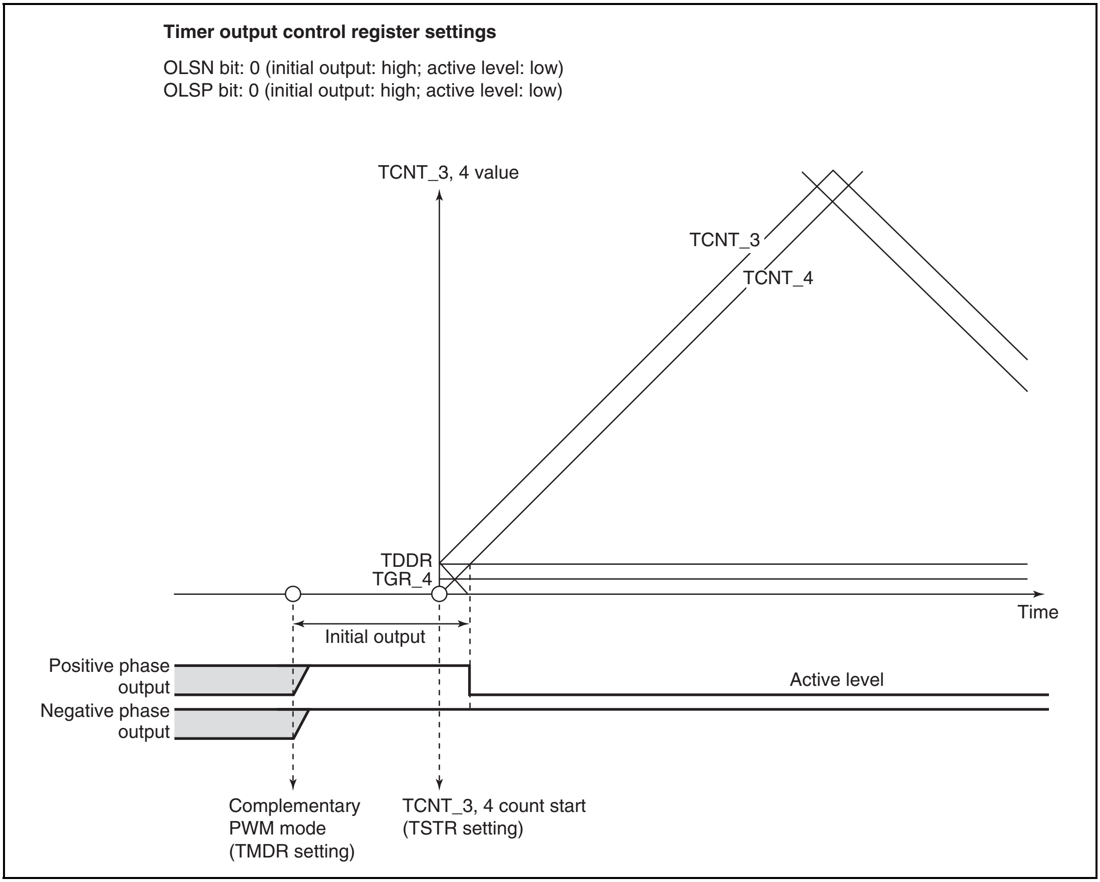


Figure 8.39 Example of Initial Output in Complementary PWM Mode (2)

Complementary PWM Mode PWM Output Generation Method: In complementary PWM mode, 3-phase output is performed of PWM waveforms with a non-overlap time between the positive and negative phases. This non-overlap time is called the dead time.

A PWM waveform is generated by output of the output level selected in the timer output control register in the event of a compare-match between a counter and data register. While TCNTS is counting, data register and temporary register values are simultaneously compared to create consecutive PWM pulses from 0 to 100%. The relative timing of on and off compare-match occurrence may vary, but the compare-match that turns off each phase takes precedence to secure the dead time and ensure that the positive phase and negative phase on times do not overlap. Figures 8.40 to 8.42 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 8.40.

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 8.41, compare-match **b** is ignored, and the negative phase is turned off by compare-match **d**. This is because turning off of the positive phase has priority due to the occurrence of compare-match **c** (positive phase off timing) before compare-match **b** (positive phase on timing) (consequently, the waveform does not change since the positive phase goes from off to off).

Similarly, in the example in figure 8.42, 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 positive 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.

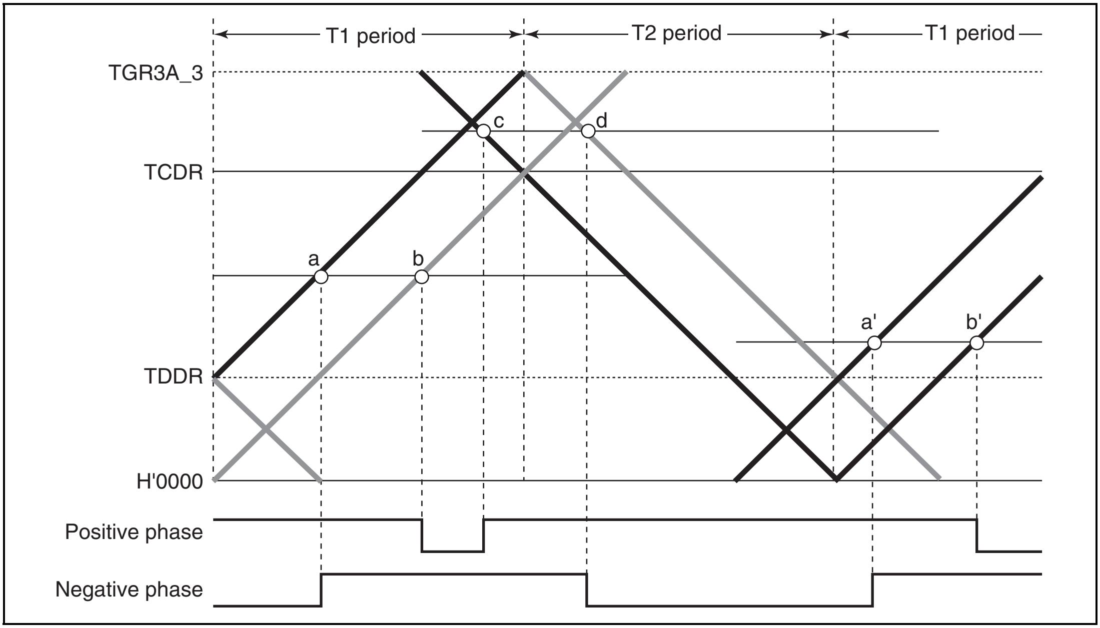


Figure 8.40 Example of Complementary PWM Mode Waveform Output (1)

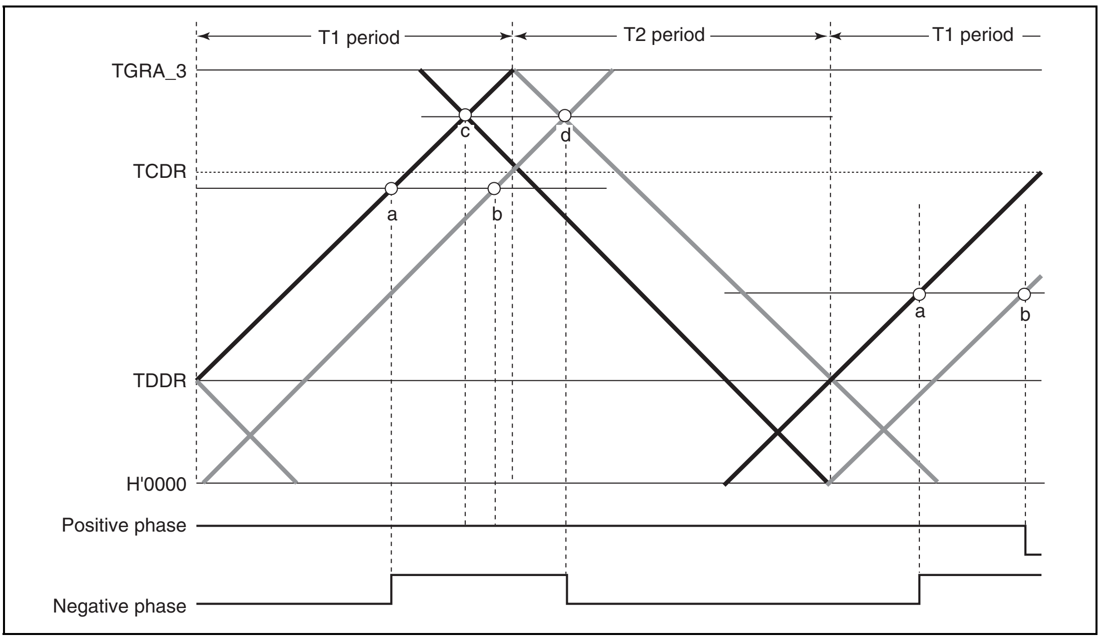


Figure 8.41 Example of Complementary PWM Mode Waveform Output (2)

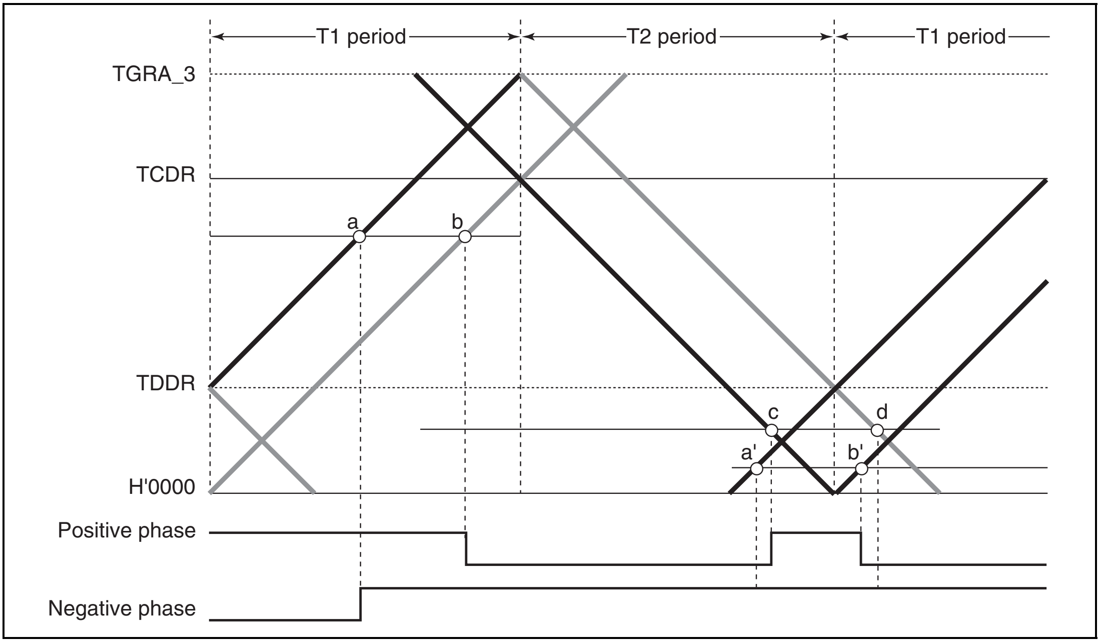


Figure 8.42 Example of Complementary PWM Mode Waveform Output (3)

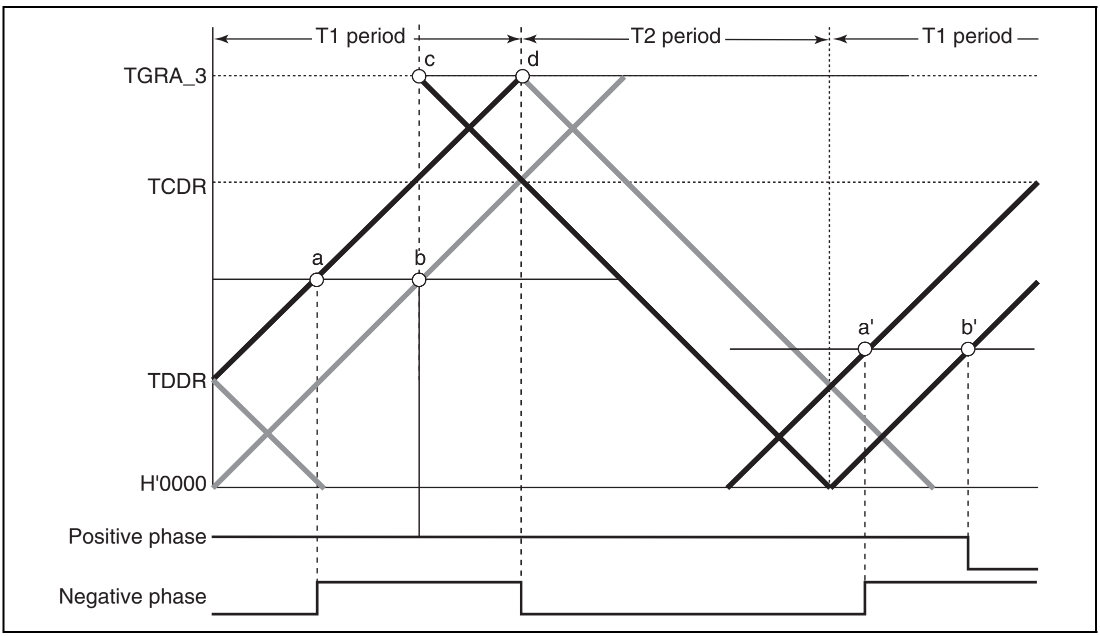


Figure 8.43 Example of Complementary PWM Mode 0% and 100% Waveform Output (1)

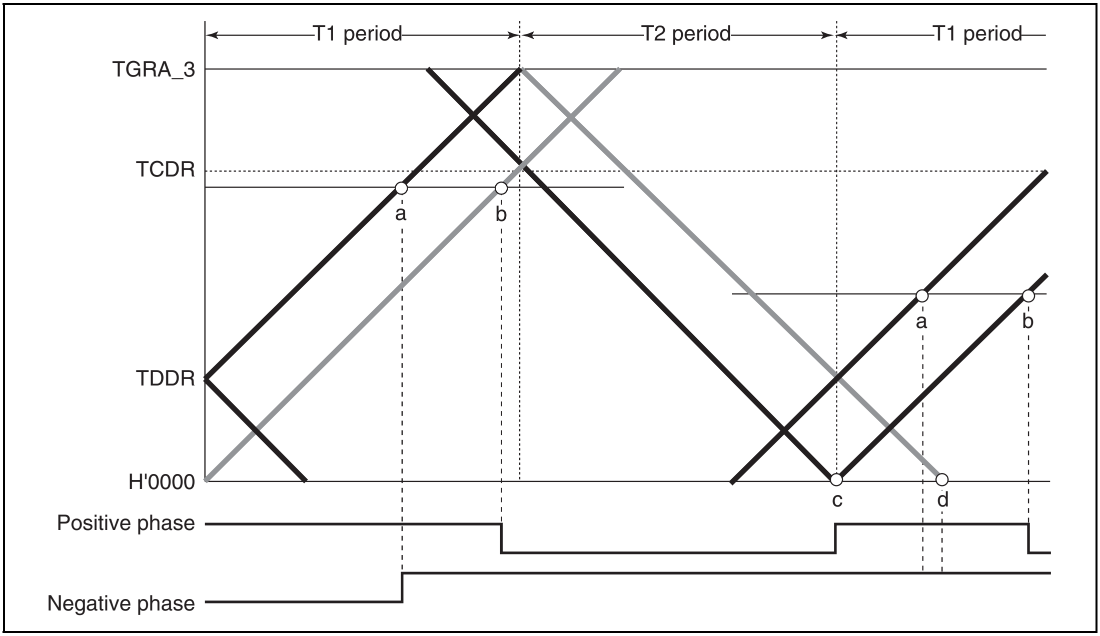


Figure 8.44 Example of Complementary PWM Mode 0% and 100% Waveform Output (2)

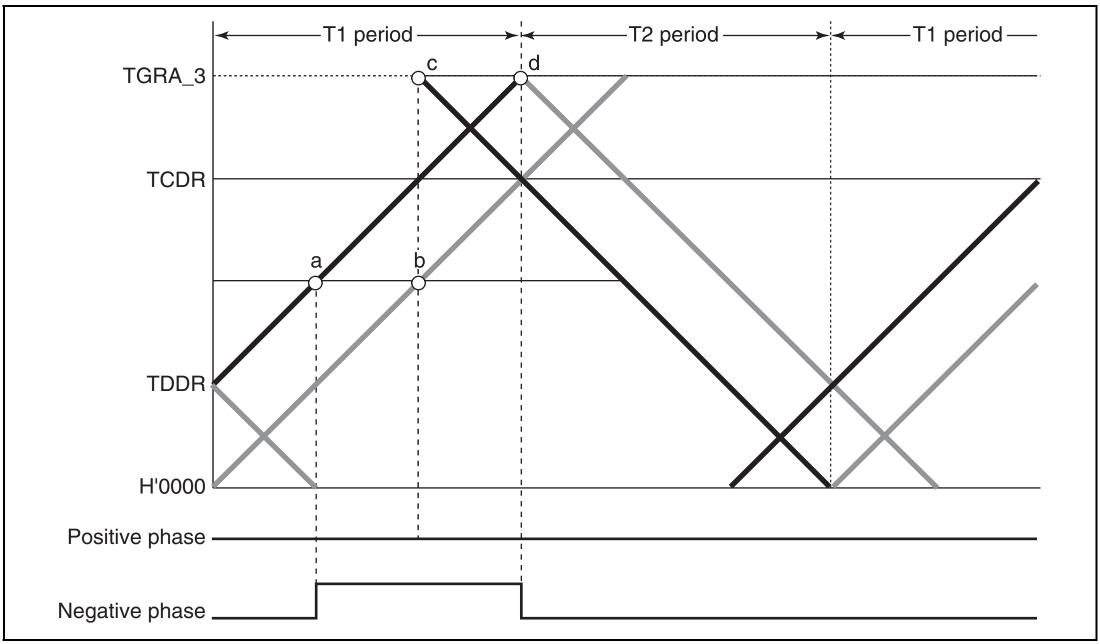


Figure 8.45 Example of Complementary PWM Mode 0% and 100% Waveform Output (3)

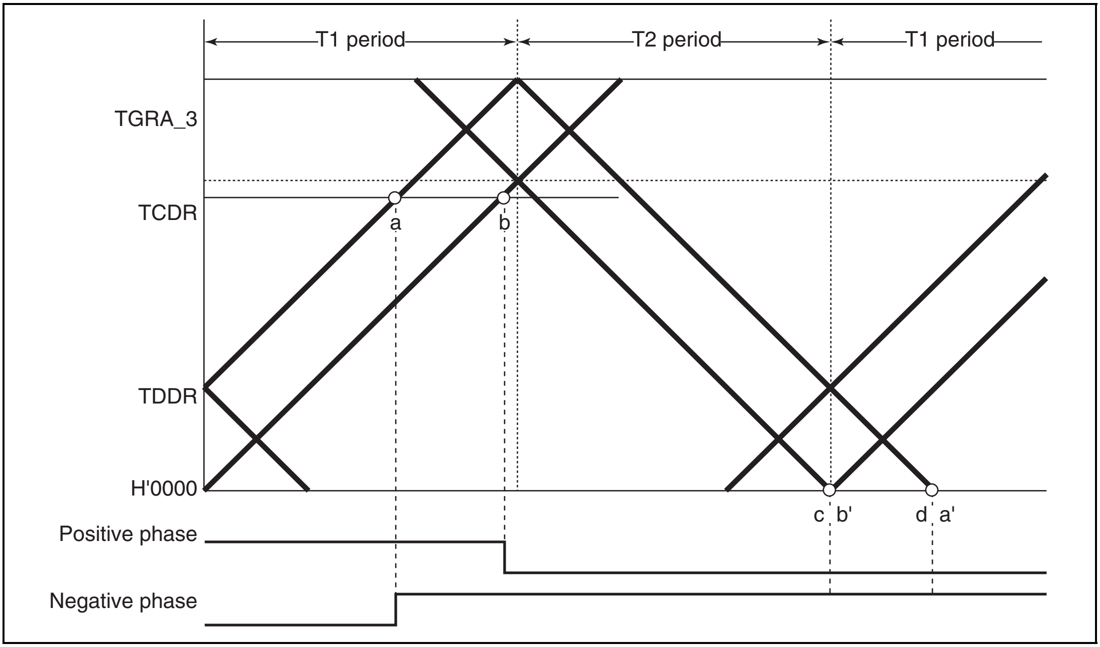


Figure 8.46 Example of Complementary PWM Mode 0% and 100% Waveform Output (4)

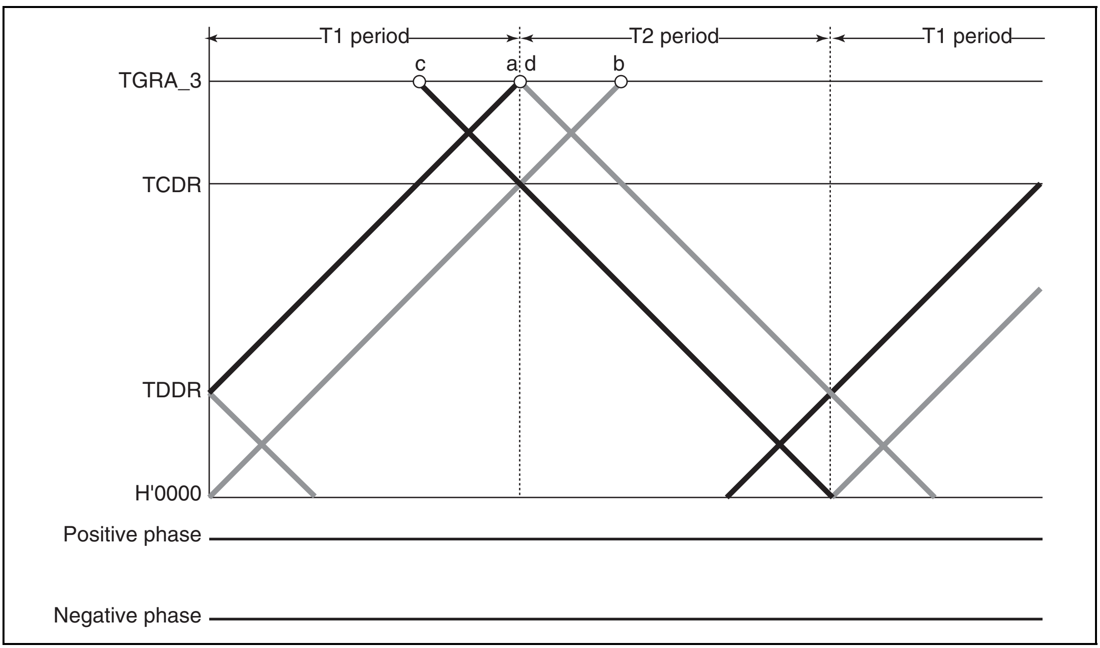


Figure 8.47 Example of Complementary PWM Mode 0% and 100% Waveform Output (5)

Complementary PWM Mode 0% and 100% Duty Cycle Output: In complementary PWM mode, 0% and 100% duty cycles can be output as required. Figures 8.43 to 8.47 show output examples.

100% duty cycle output is performed when the data register value is set to H'0000. The waveform in this case has a positive phase with a 100% on-state. 0% duty cycle output is performed when the data register value is set to the same value as TGRA_3. The waveform in this case has a positive phase with a 100% off-state.

On and off compare-matches occur simultaneously, but if a turn-on compare-match and turn-off compare-match for the same phase occur simultaneously, both compare-matches are ignored and the waveform does not change.

Toggle Output Synchronized with PWM Cycle: In complementary PWM mode, toggle output can be performed in synchronization with the PWM carrier cycle by setting the PSYE bit to 1 in the timer output control register (TOCR). An example of a toggle output waveform is shown in figure 8.48.

This output is toggled by a compare-match between TCNT_3 and TGRA_3 and a compare-match between TCNT4 and H'0000.

The output pin for this toggle output is the TIOC3A pin. The initial output is 1.

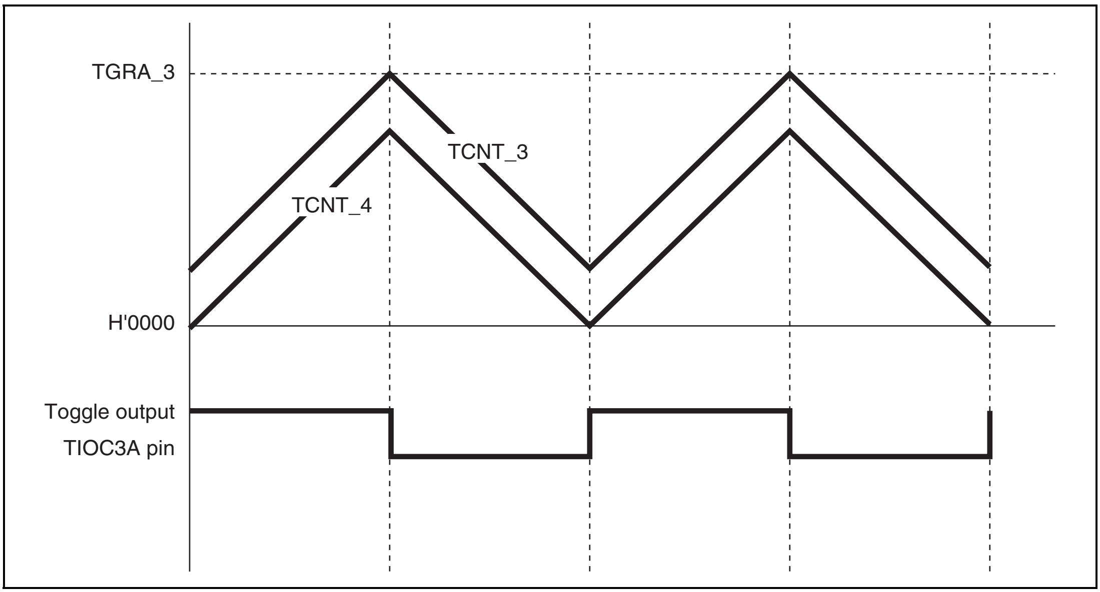


Figure 8.48 Example of Toggle Output Waveform Synchronized with PWM Output

Counter Clearing by another Channel: In complementary PWM mode, by setting a mode for synchronization with another channel by means of the timer synchro register (TSYR), and selecting synchronous clearing with bits CCLR2 to CCLR0 in the timer control register (TCR), it is possible to have TCNT_3, TCNT_4, and TCNTS cleared by another channel.

Figure 8.49 illustrates the operation.

Use of this function enables counter clearing and restarting to be performed by means of an external signal.

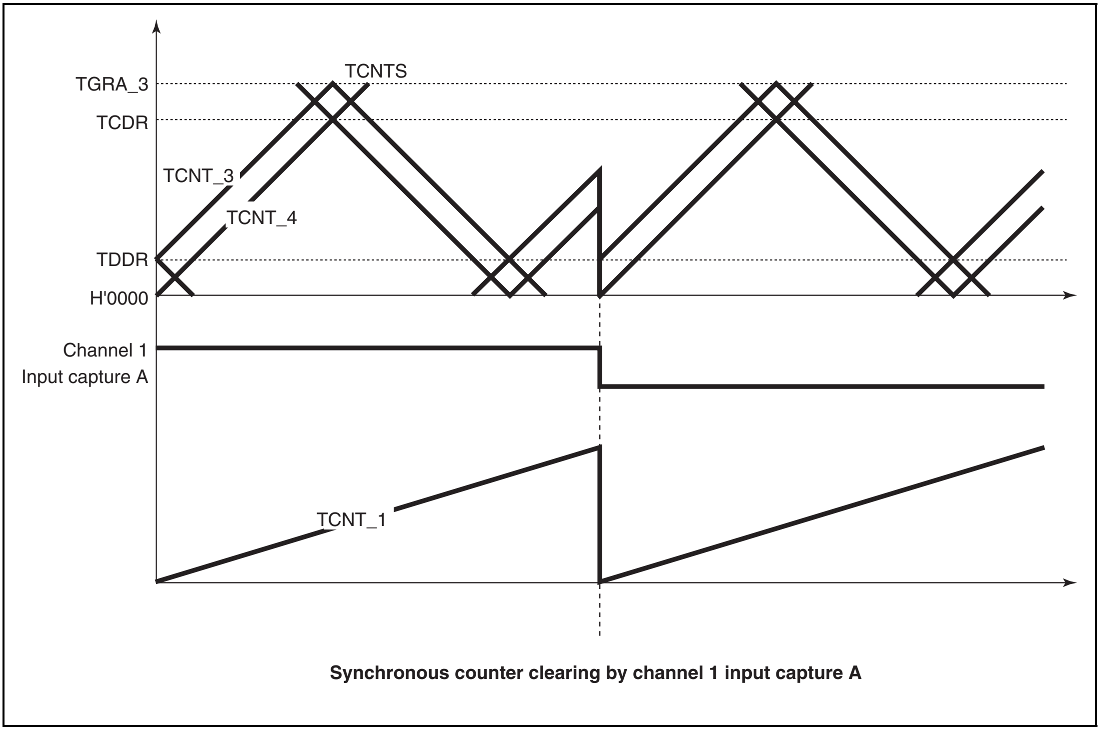


Figure 8.49 Counter Clearing Synchronized with Another Channel

Example of AC Synchronous Motor (Brushless DC Motor) Drive Waveform Output: In complementary PWM mode, a brushless DC motor can easily be controlled using the timer gate control register (TGCR). Figures 8.50 to 8.53 show examples of brushless DC motor drive waveforms created using TGCR.

When output phase switching for a 3-phase brushless DC motor is performed by means of external signals detected with a Hall element, etc., clear the FB bit in TGCR to 0. In this case, the external signals indicating the polarity position are input to channel 0 timer input pins TIOC0A, TIOC0B, and TIOC0C (set with PFC). When an edge is detected at pin TIOC0A, TIOC0B, or TIOC0C, the output on/off state is switched automatically.

When the FB bit is 1, the output on/off state is switched when the UF, VF, or WF bit in TGCR is cleared to 0 or set to 1.

The drive waveforms are output from the complementary PWM mode 6-phase output pins. With this 6-phase output, in the case of on output, it is possible to use complementary PWM mode output and perform chopping output by setting the N bit or P bit to 1. When the N bit or P bit is 0, level output is selected.

The 6-phase output active level (on output level) can be set with the OLSN and OLSP bits in the timer output control register (TOCR) regardless of the setting of the N and P bits.

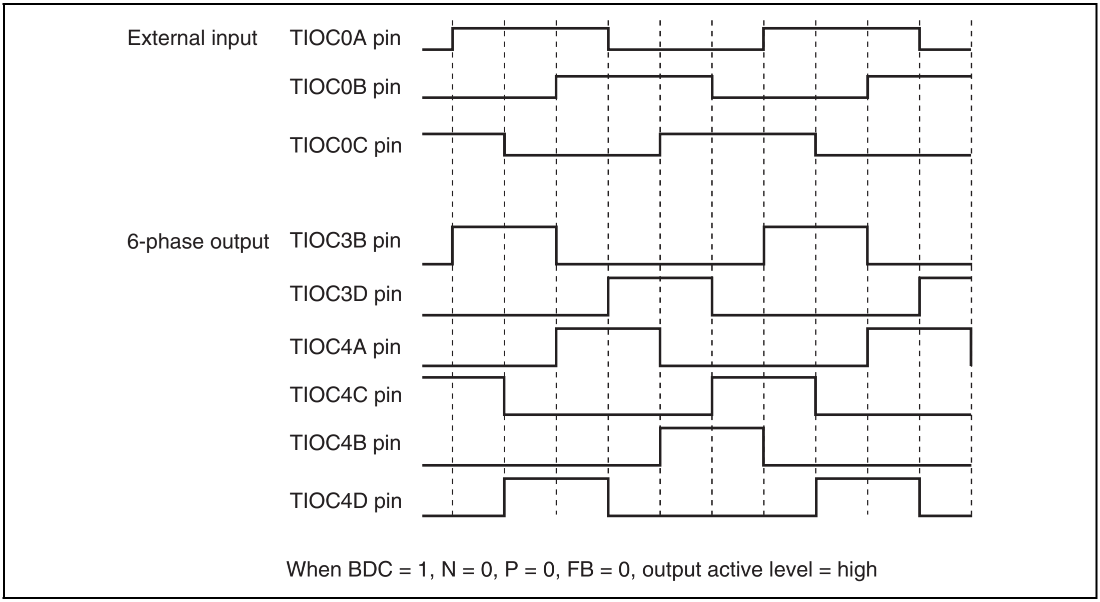


Figure 8.50 Example of Output Phase Switching by External Input (1)

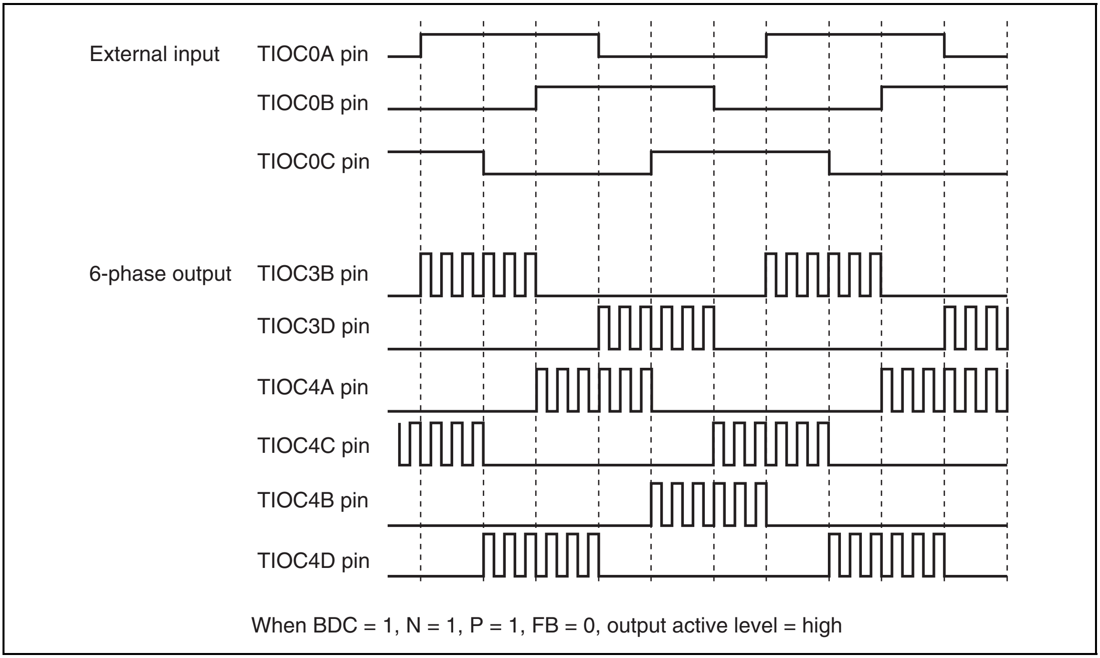


Figure 8.51 Example of Output Phase Switching by External Input (2)

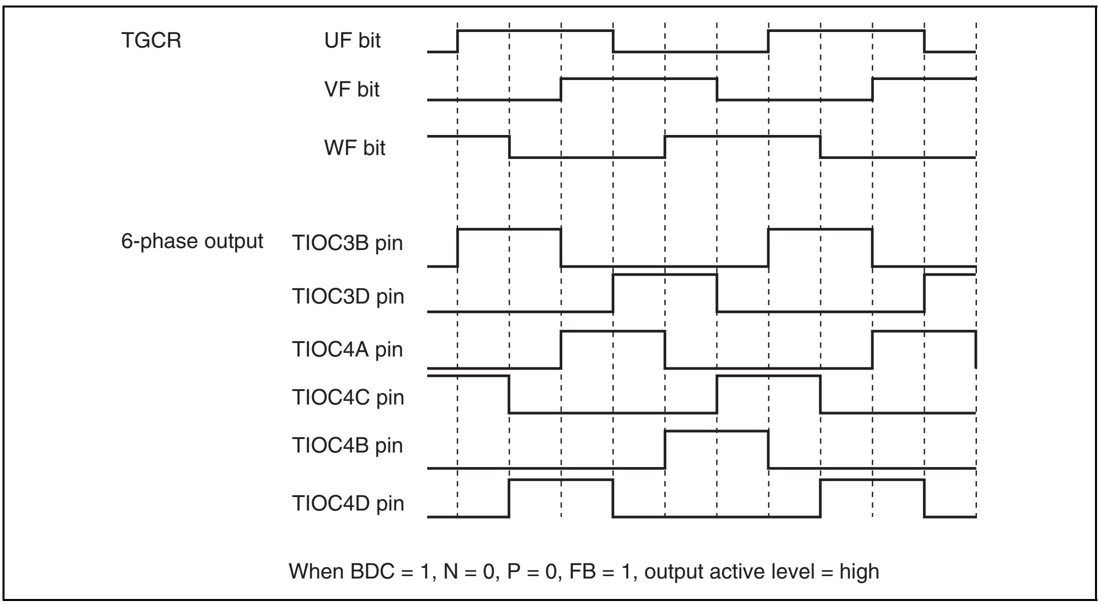


Figure 8.52 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (1)

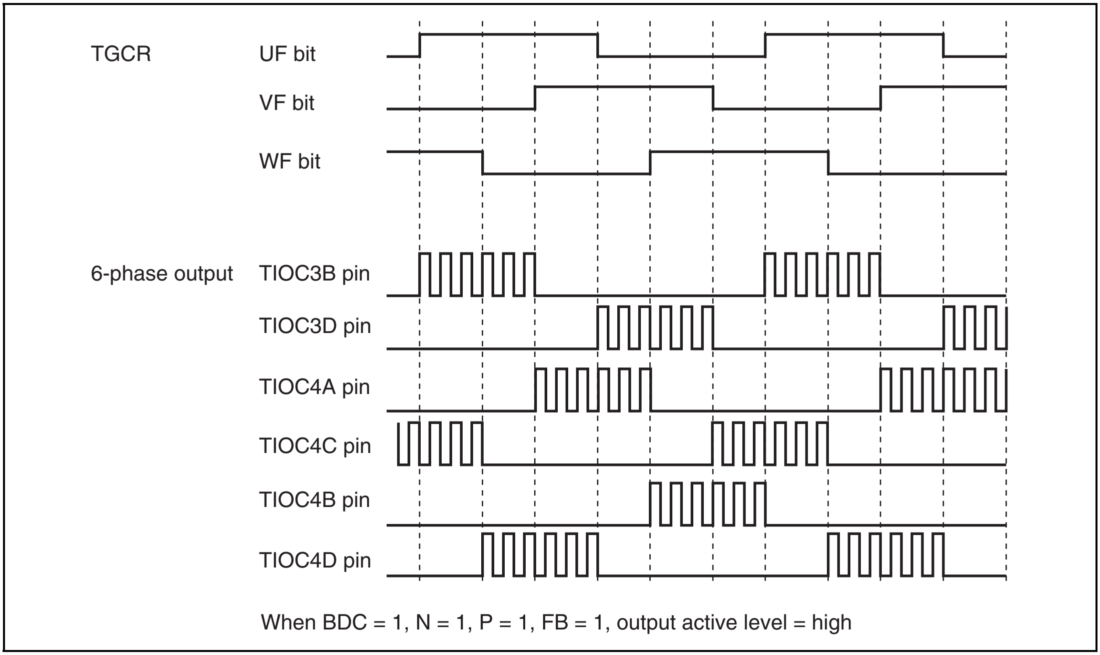


Figure 8.53 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (2)

A/D Conversion Start Request Setting: In complementary PWM mode, an A/D conversion start request can be set using a TGRA_3 compare-match or a compare-match on a channel other than channels 3 and 4.

When start requests using a TGRA_3 compare-match are set, A/D conversion can be started at the center of the PWM pulse.

A/D conversion start requests can be set by setting the TTGE bit to 1 in the timer interrupt enable register (TIER).

Complementary PWM Mode Output Protection Function

Complementary PWM mode output has the following protection functions.

- 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 bit 13 in the bus controller's bus control register 1 (BCR1). Some registers in channels 3 and 4 concerned are listed below: total 21 registers of 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 the CPU to prevent miswriting due to the CPU runaway by disabling CPU access to the mode registers, control registers, and counters. In access disabled state, an undefined value is read from the registers concerned, and cannot be modified.
- 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 8.9, Port Output Enable (POE), for details.

- 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.

  For details, see section 4.2, Function for Detecting Oscillator Halt.

8.5 Interrupt Sources

8.5.1 Interrupts and Priorities

There are three kinds of MTU 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 8.42 lists the MTU interrupt sources.

Table 8.42 MTU Interrupts

Channel	Name	Interrupt Source	Interrupt Flag	Priority
0	TGI0A	TGRA_0 input capture/compare match	TGFA_0	High
	TGI0B	TGRB_0 input capture/compare match	TGFB_0	↑
	TGI0C	TGRC_0 input capture/compare match	TGFC_0	
	TGI0D	TGRD_0 input capture/compare match	TGFD_0	
	TCI0V	TCNT_0 overflow	TCFV_0	
1	TGI1A	TGRA_1 input capture/compare match	TGFA_1	
	TGI1B	TGRB_1 input capture/compare match	TGFB_1	
	TCI1V	TCNT_1 overflow	TCFV_1	
	TCI1U	TCNT_1 underflow	TCFU_1	
2	TGI2A	TGRA_2 input capture/compare match	TGFA_2	
	TGI2B	TGRB_2 input capture/compare match	TGFB_2	
	TCI2V	TCNT_2 overflow	TCFV_2	
	TCI2U	TCNT_2 underflow	TCFU_2	
3	TGI3A	TGRA_3 input capture/compare match	TGFA_3	
	TGI3B	TGRB_3 input capture/compare match	TGFB_3	
	TGI3C	TGRC_3 input capture/compare match	TGFC_3	
	TGI3D	TGRD_3 input capture/compare match	TGFD_3	
	TCI3V	TCNT_3 overflow	TCFV_3	
4	TGI4A	TGRA_4 input capture/compare match	TGFA_4	
	TGI4B	TGRB_4 input capture/compare match	TGFB_4	
	TGI4C	TGRC_4 input capture/compare match	TGFC_4	
	TGI4D	TGRD_4 input capture/compare match	TGFD_4	↓
	TCI4V	TCNT_4 overflow	TCFV_4	Low

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

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 MTU has 16 input capture/compare match interrupts, four each for channels 0, 3, and 4, and two each for channels 1 and 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 MTU has five overflow interrupts, one for each channel.

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 MTU has two underflow interrupts, one each for channels 1 and 2.

8.5.2 A/D Converter Activation

The A/D converter can be activated by the TGRA input capture/compare match in each channel.

If the TTGE bit in TIER is set to 1 when the TGFA flag in TSR is set to 1 by the occurrence of a TGRA input capture/compare match on a particular channel, a request to start A/D conversion is sent to the A/D converter. If the MTU conversion start trigger has been selected on the A/D converter at this time, A/D conversion starts.

In the MTU, a total of five TGRA input capture/compare match interrupts can be used as A/D converter conversion start sources, one for each channel.

8.6 Operation Timing

8.6.1 Input/Output Timing

TCNT Count Timing: Figure 8.54 shows TCNT count timing in internal clock operation, and figure 8.55 shows TCNT count timing in external clock operation (normal mode), and figure 8.56 shows TCNT count timing in external clock operation (phase counting mode).

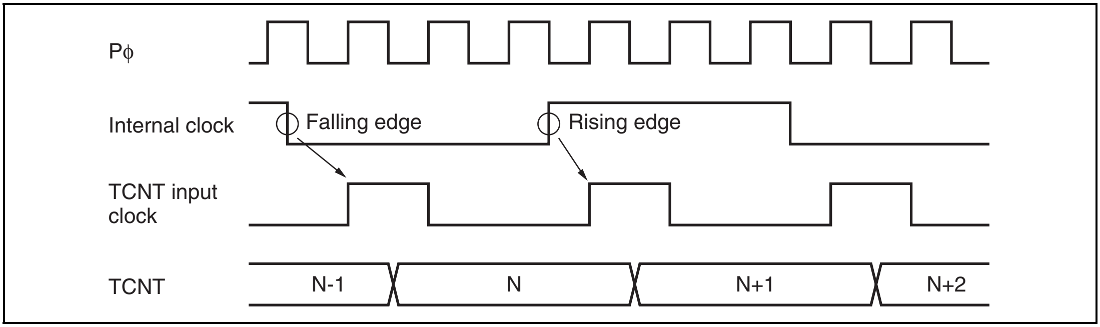


Figure 8.54 Count Timing in Internal Clock Operation

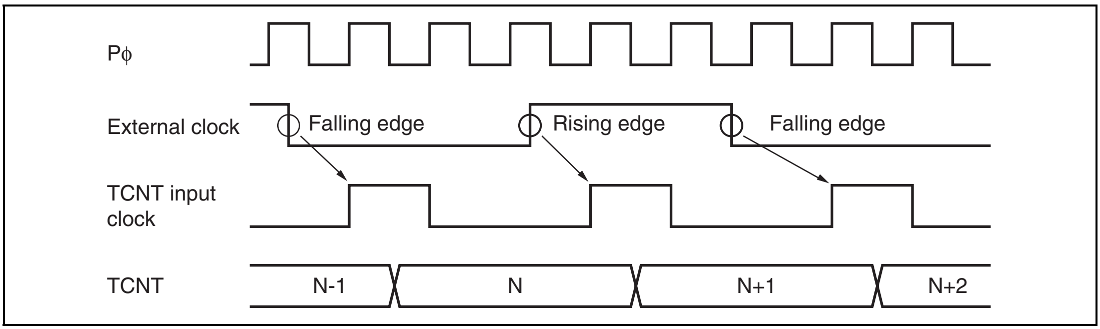


Figure 8.55 Count Timing in External Clock Operation

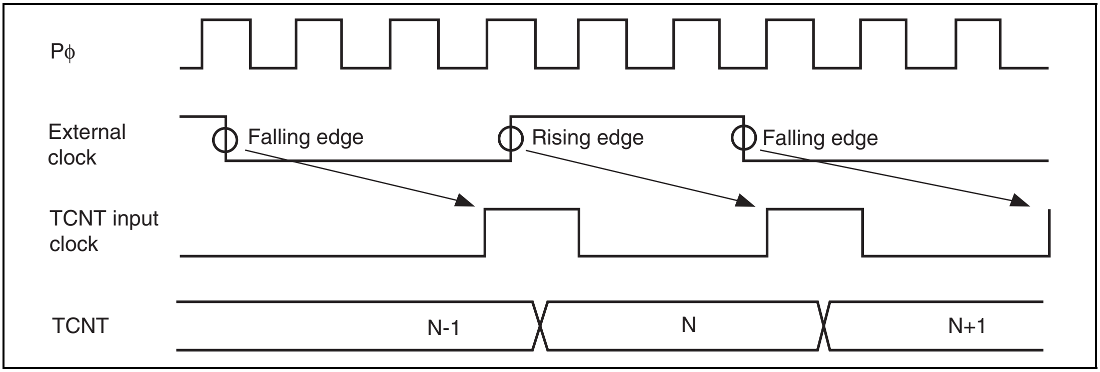


Figure 8.56 Count Timing in External Clock Operation (Phase Counting Mode)

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 8.57 shows output compare output timing (normal mode and PWM mode) and figure 8.58 shows output compare output timing (complementary PWM mode and reset synchronous PWM mode).

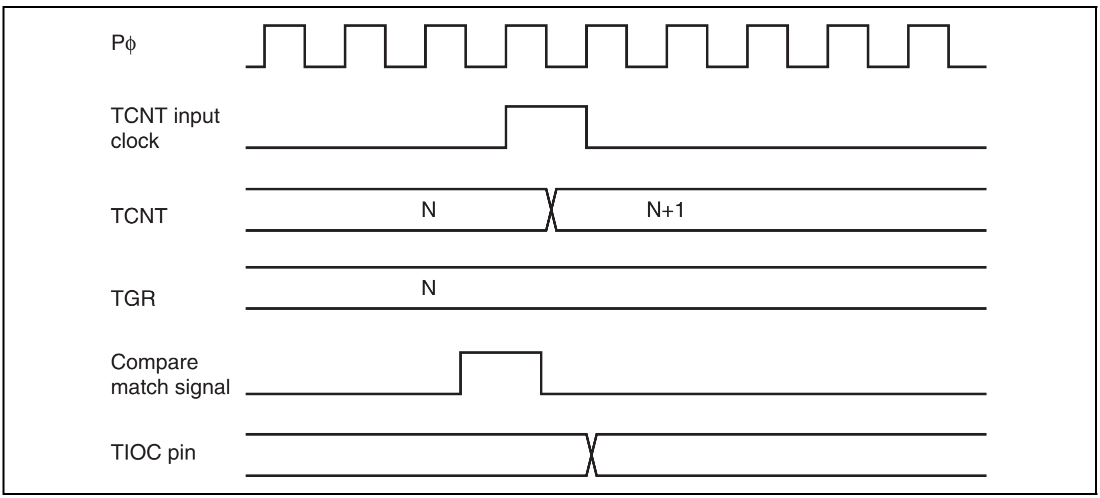


Figure 8.57 Output Compare Output Timing (Normal Mode/PWM Mode)

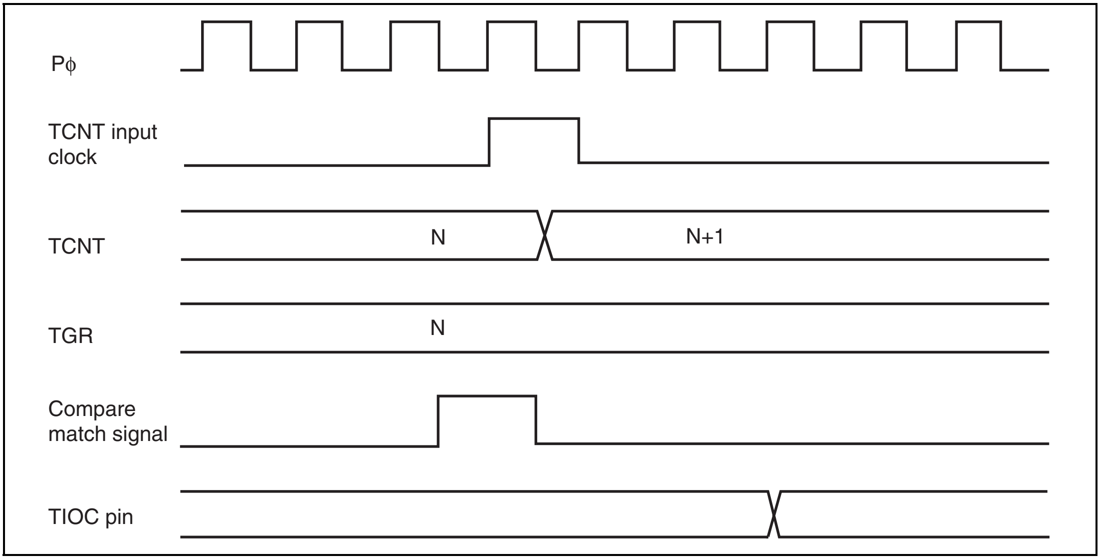


Figure 8.58 Output Compare Output Timing (Complementary PWM Mode/Reset Synchronous PWM Mode)

Input Capture Signal Timing: Figure 8.59 shows input capture signal timing.

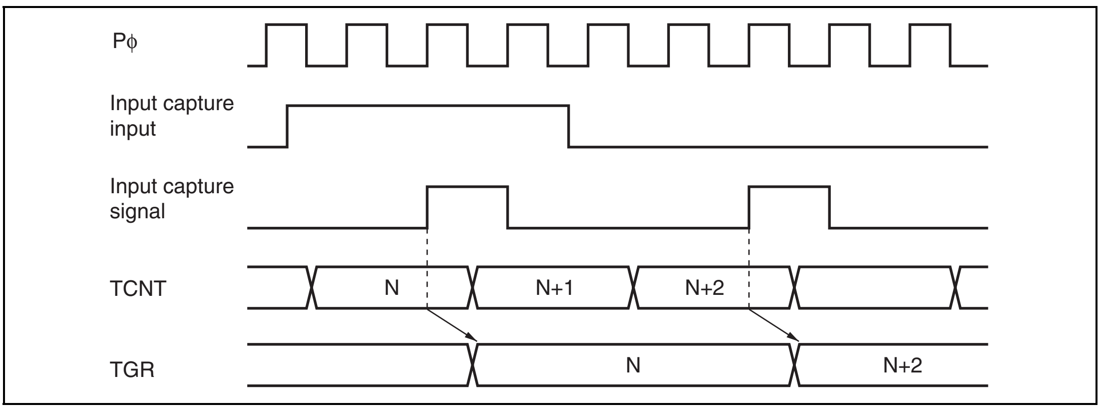


Figure 8.59 Input Capture Input Signal Timing

Timing for Counter Clearing by Compare Match/Input Capture: Figure 8.60 shows the timing when counter clearing on compare match is specified, and figure 8.61 shows the timing when counter clearing on input capture is specified.

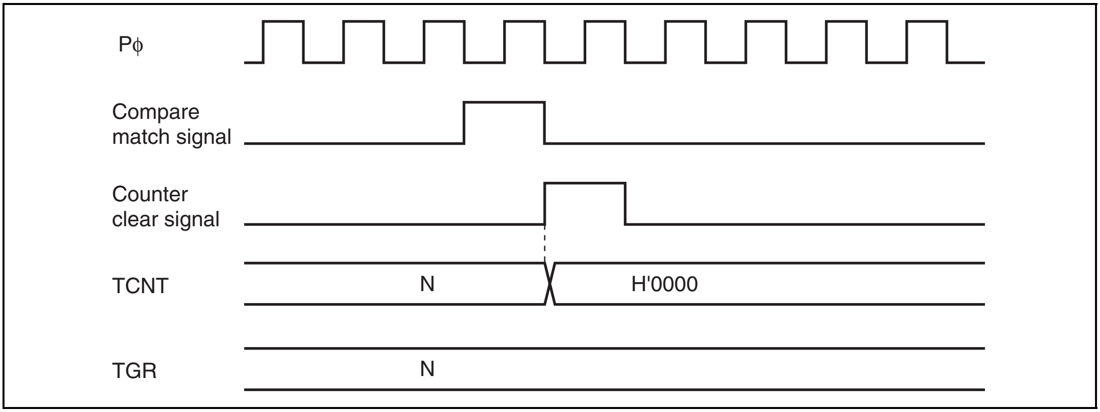


Figure 8.60 Counter Clear Timing (Compare Match)

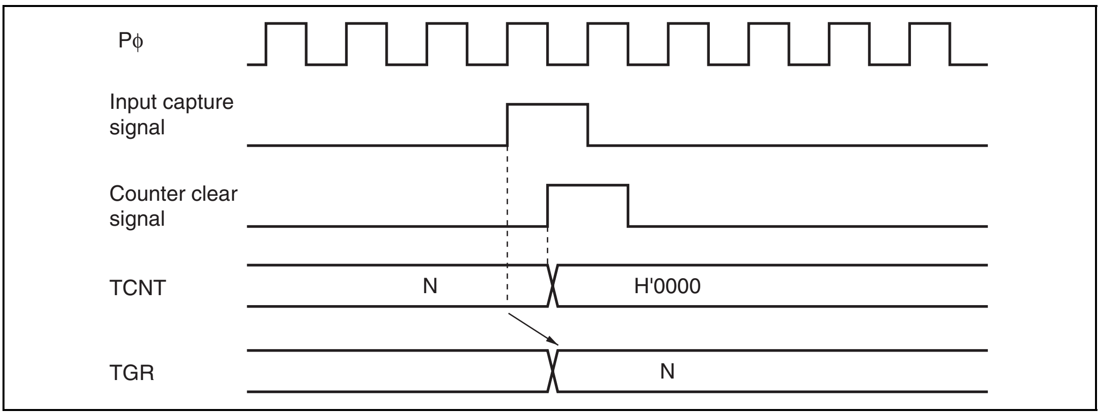


Figure 8.61 Counter Clear Timing (Input Capture)

Buffer Operation Timing: Figures 8.62 and 8.63 show the timing in buffer operation.

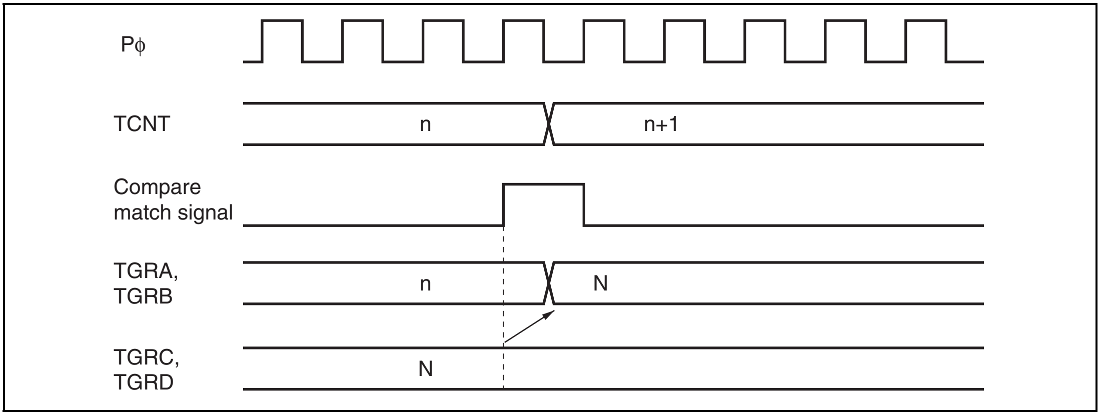


Figure 8.62 Buffer Operation Timing (Compare Match)

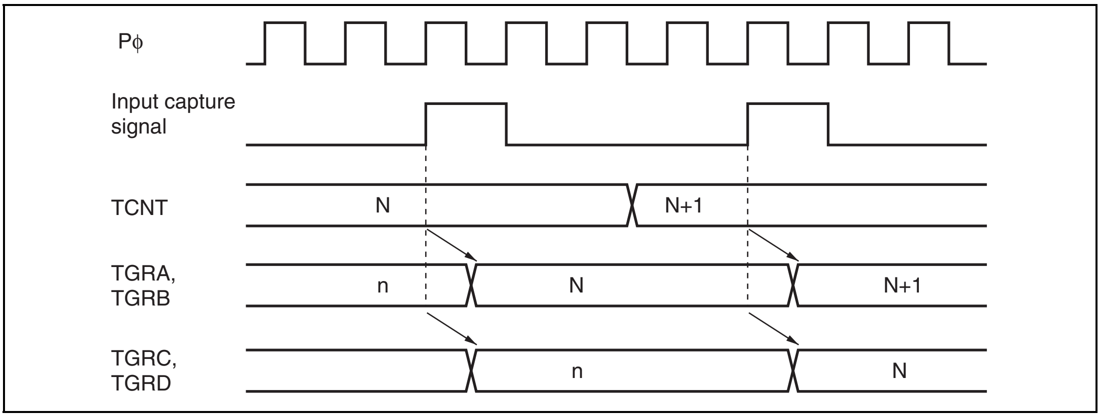


Figure 8.63 Buffer Operation Timing (Input Capture)

8.6.2 Interrupt Signal Timing

TGF Flag Setting Timing in Case of Compare Match: Figure 8.64 shows the timing for setting of the TGF flag in TSR on compare match, and TGI interrupt request signal timing.

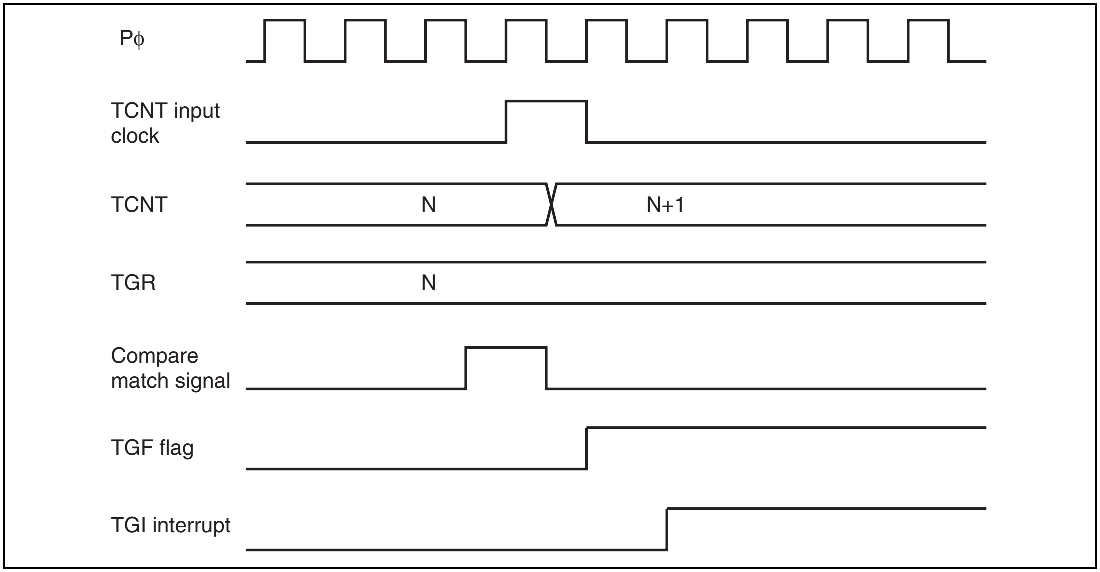


Figure 8.64 TGI Interrupt Timing (Compare Match)

TGF Flag Setting Timing in Case of Input Capture: Figure 8.65 shows the timing for setting of the TGF flag in TSR on input capture, and TGI interrupt request signal timing.

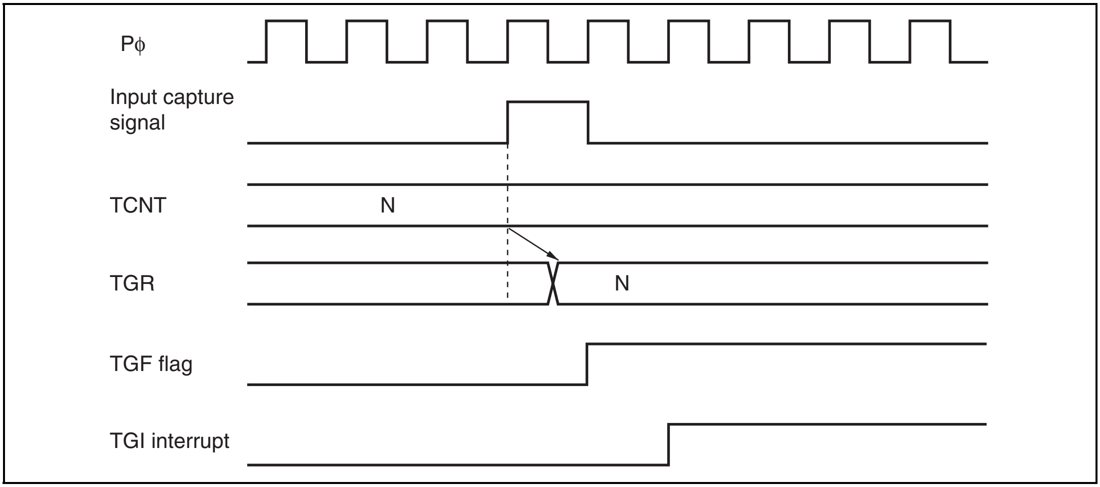


Figure 8.65 TGI Interrupt Timing (Input Capture)

TCFV Flag/TCFU Flag Setting Timing: Figure 8.66 shows the timing for setting of the TCFV flag in TSR on overflow, and TCIV interrupt request signal timing.

Figure 8.67 shows the timing for setting of the TCFU flag in TSR on underflow, and TCIU interrupt request signal timing.

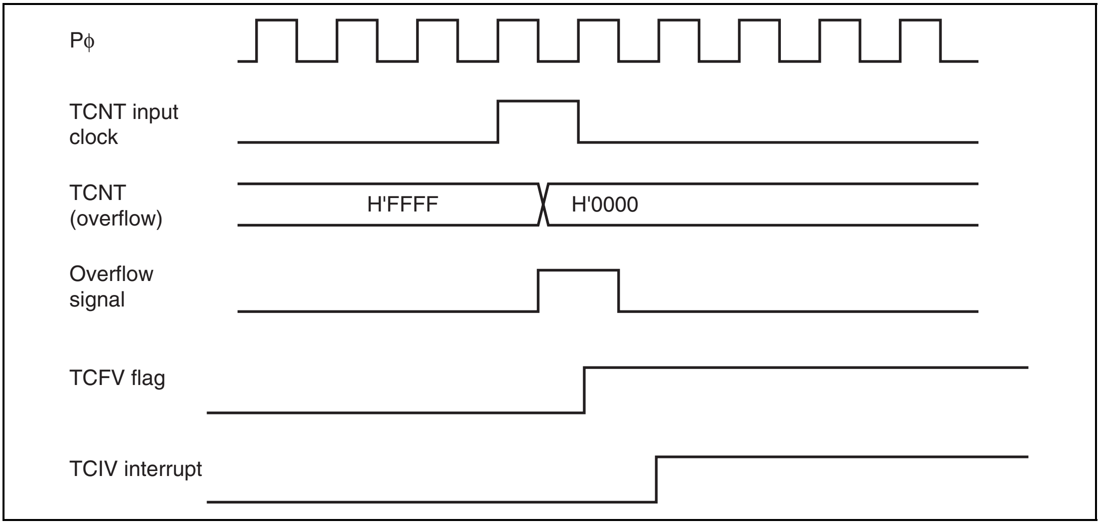


Figure 8.66 TCIV Interrupt Setting Timing

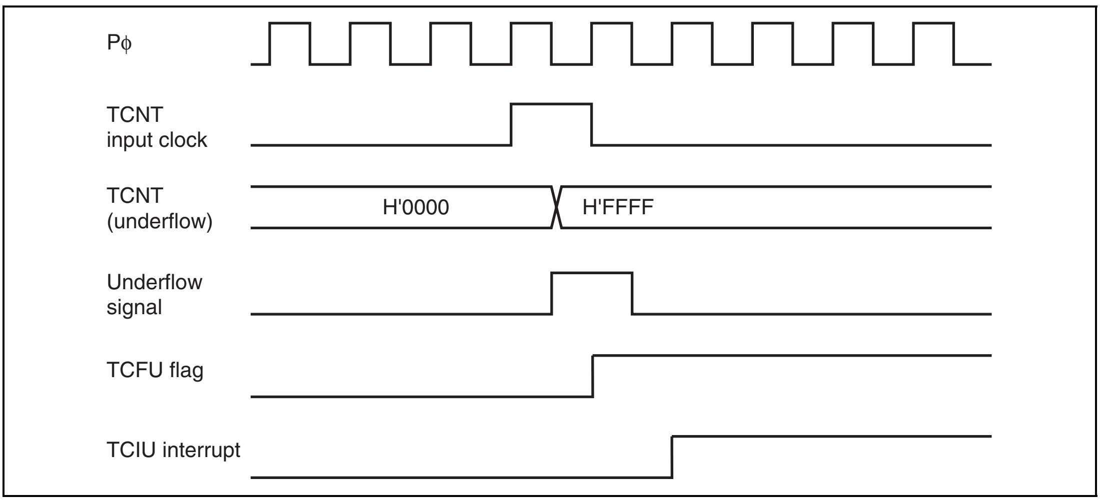


Figure 8.67 TCIU Interrupt Setting Timing

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 is activated, the flag is cleared automatically. Figure 8.68 shows the timing for status flag clearing by the CPU.

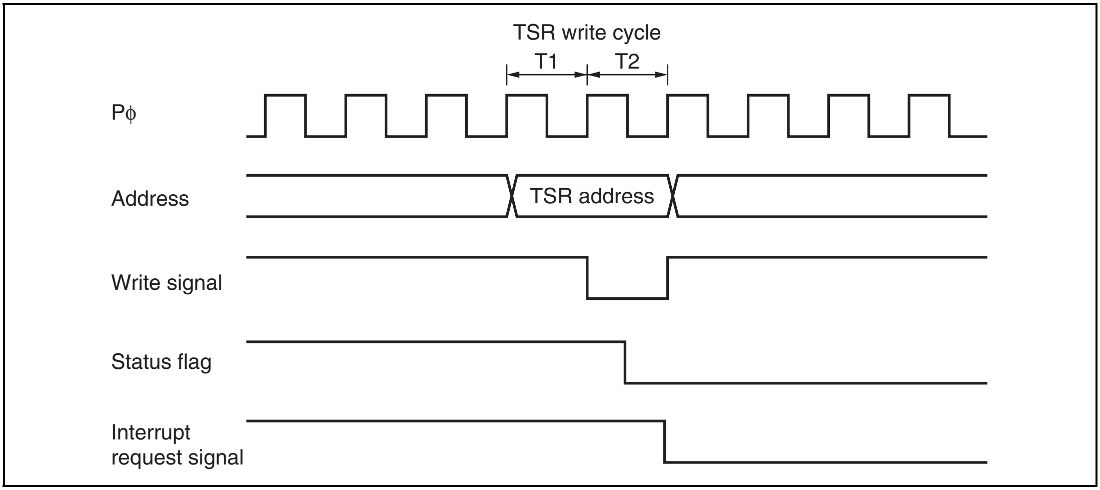


Figure 8.68 Timing for Status Flag Clearing by the CPU

8.7 Usage Notes

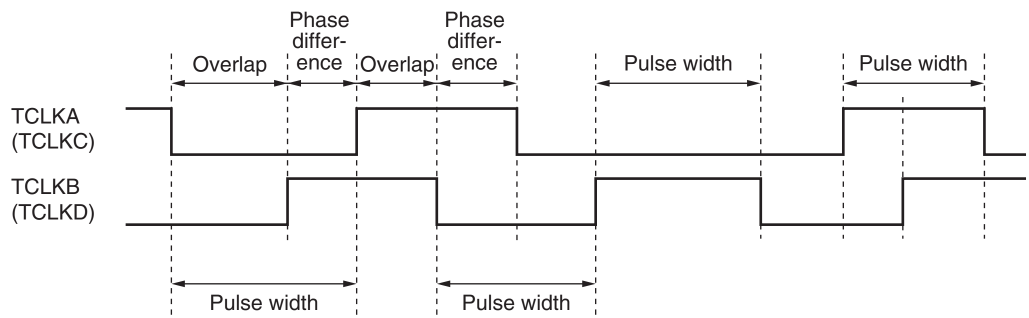
8.7.1 Module Standby Mode Setting

MTU operation can be disabled or enabled using the module standby register. The initial setting is for MTU operation to be halted. Register access is enabled by clearing module standby mode. For details, refer to section 17, Power-Down Modes.

8.7.2 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 TPU 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 8.69 shows the input clock conditions in phase counting mode.



Notes: Phase difference and overlap : 1.5 states or more
Pulse width : 2.5 states or more

Figure 8.69 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

8.7.3 Caution on Period Setting

When counter clearing on compare match is set, TCNT is cleared in the final state in which it matches the TGR value (the point at which the count value matched by TCNT is updated). Consequently, the actual counter frequency is given by the following formula:

$$f = \frac{P\phi}{(N + 1)}$$

Where f : Counter frequency
Pϕ : Peripheral clock operating frequency
N : TGR set value

8.7.4 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 8.70 shows the timing in this case.

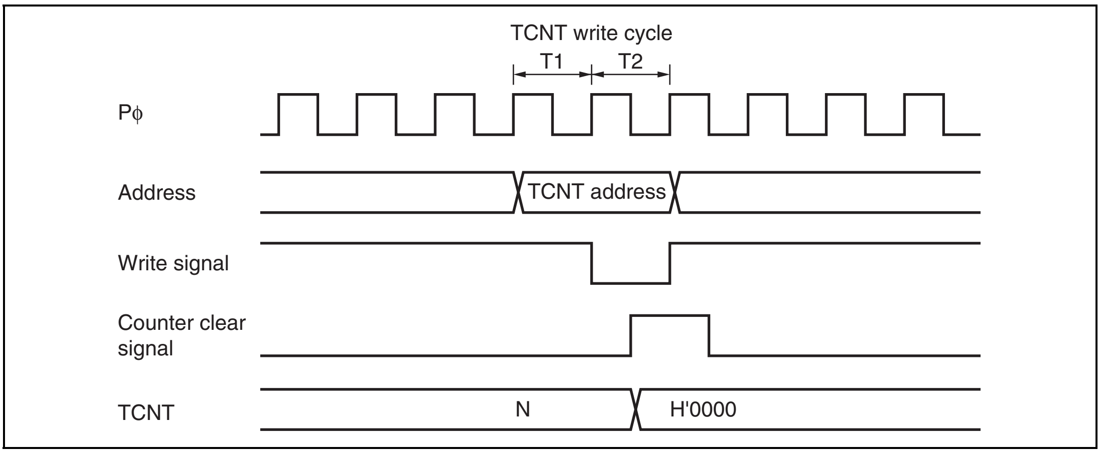


Figure 8.70 Contention between TCNT Write and Clear Operations

8.7.5 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 8.71 shows the timing in this case.

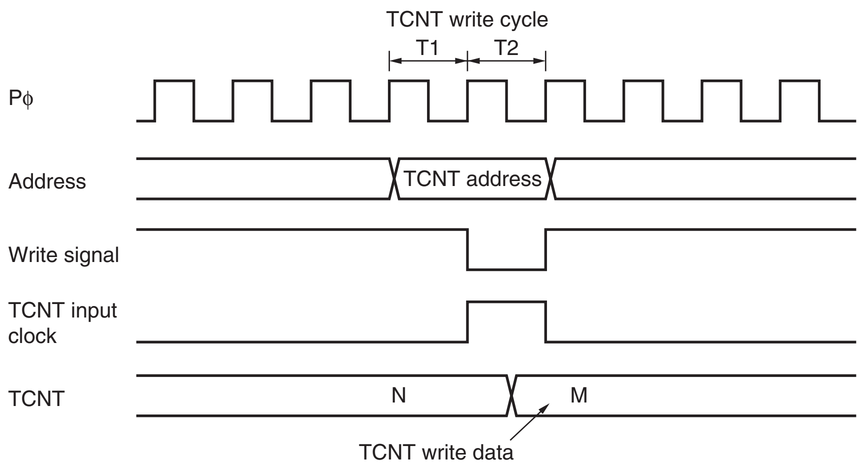


Figure 8.71 Contention between TCNT Write and Increment Operations

8.7.6 Contention between TGR Write and Compare Match

When a compare match occurs in the T2 state of a TGR write cycle, the TGR write is executed and the compare match signal is generated.

Figure 8.72 shows the timing in this case.

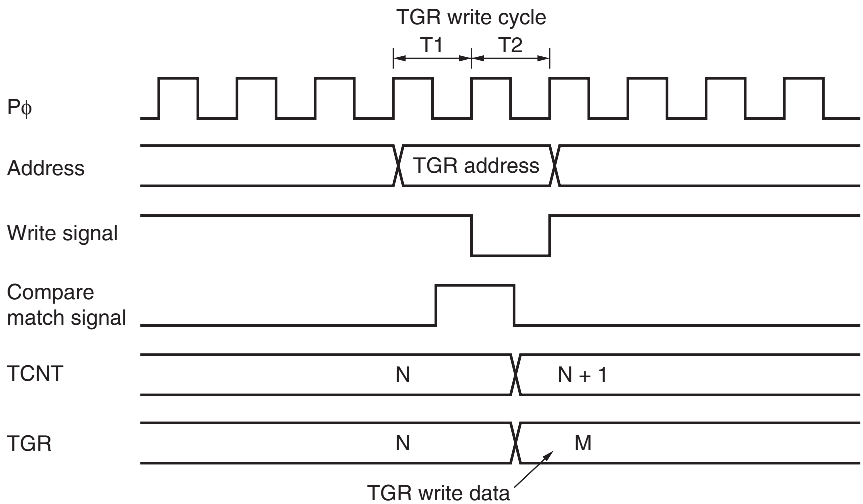


Figure 8.72 Contention between TGR Write and Compare Match

8.7.7 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 differs depending on channel 0 and channels 3 and 4: data on channel 0 is that after write, and on channels 3 and 4, before write.

Figures 8.73 and 8.74 show the timing in this case.

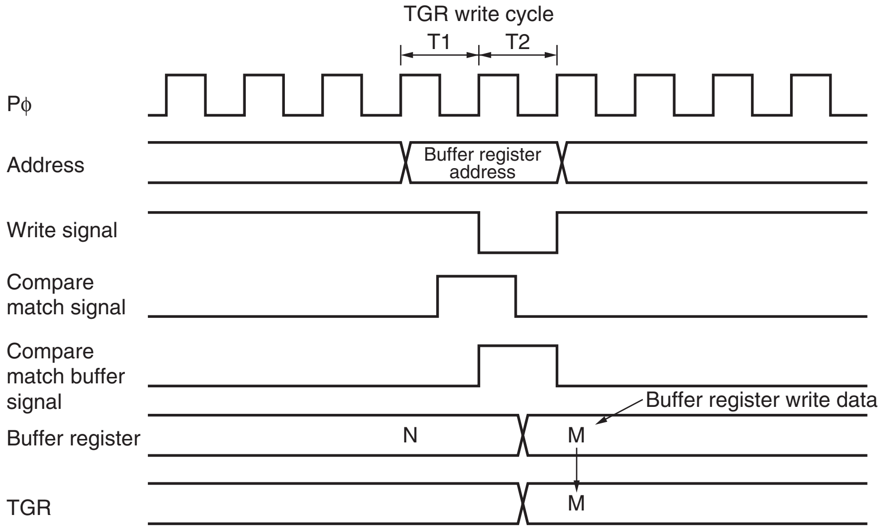


Figure 8.73 Contention between Buffer Register Write and Compare Match (Channel 0)

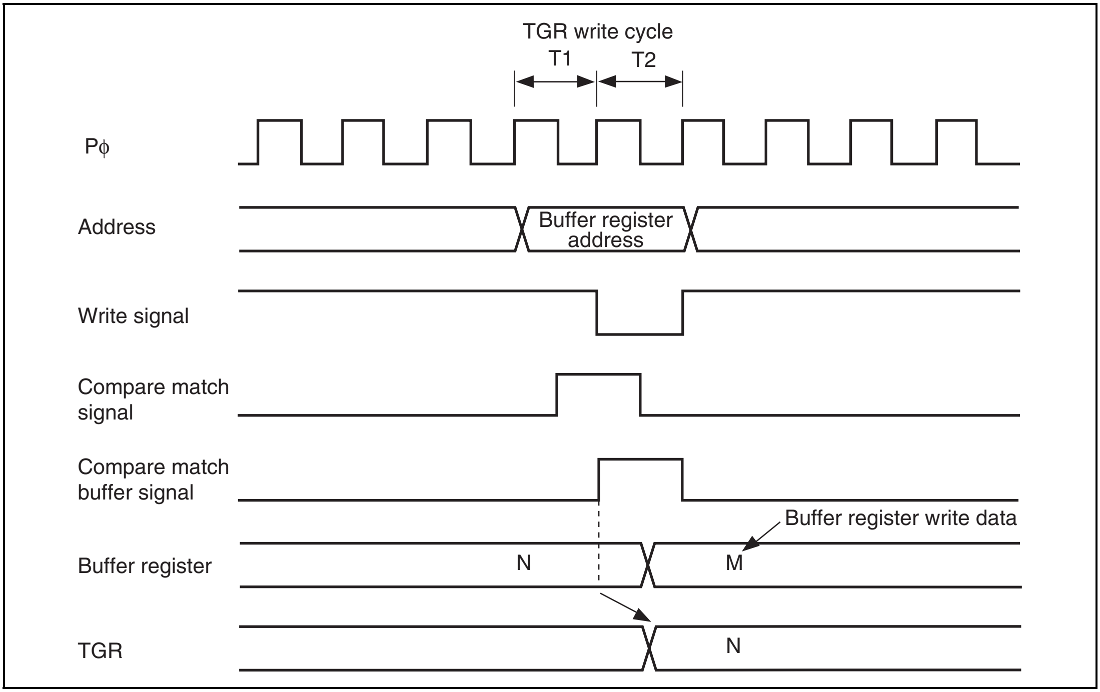


Figure 8.74 Contention between Buffer Register Write and Compare Match (Channels 3 and 4)

8.7.8 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 that in the buffer after input capture transfer.

Figure 8.75 shows the timing in this case.

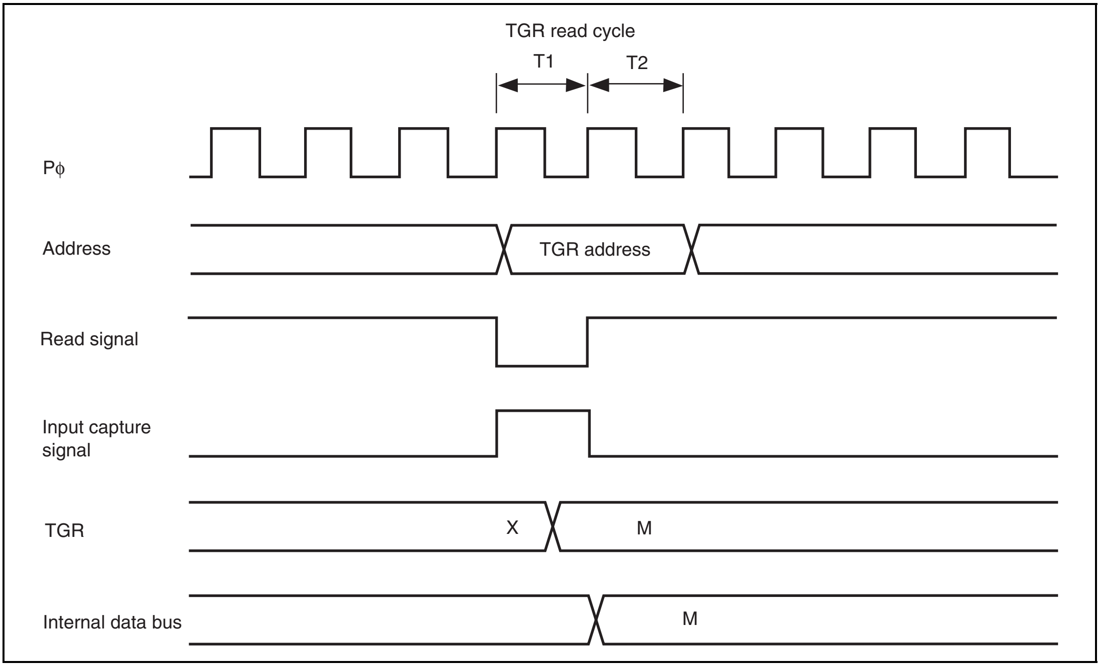


Figure 8.75 Contention between TGR Read and Input Capture

8.7.9 Contention between TGR Write and Input Capture

If an input capture signal is generated in the T2 state of a TGR write cycle, the input capture operation takes precedence and the write to TGR is not performed.

Figure 8.76 shows the timing in this case.

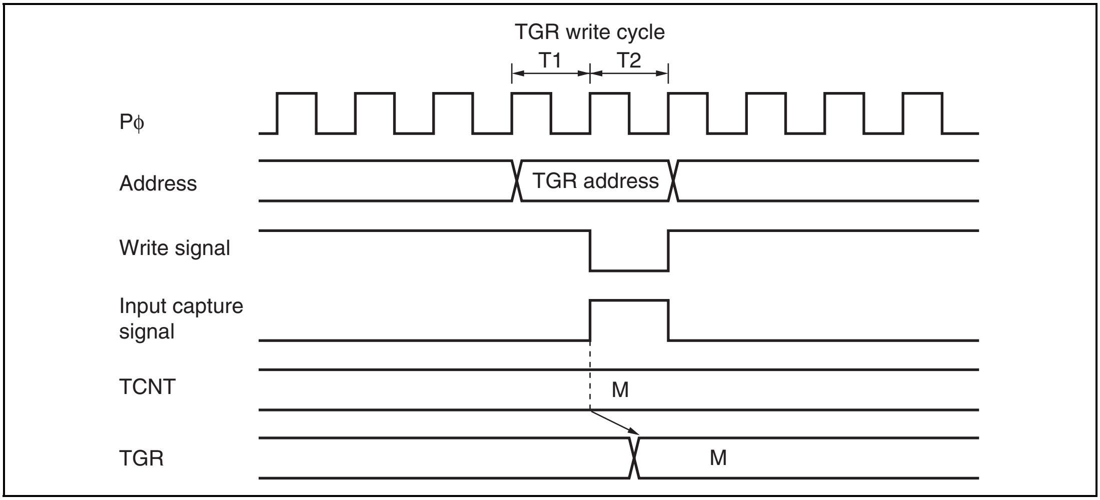


Figure 8.76 Contention between TGR Write and Input Capture

8.7.10 Contention between Buffer Register Write and Input Capture

If an input capture signal is generated in the T2 state of a buffer register write cycle, the buffer operation takes precedence and the write to the buffer register is not performed.

Figure 8.77 shows the timing in this case.

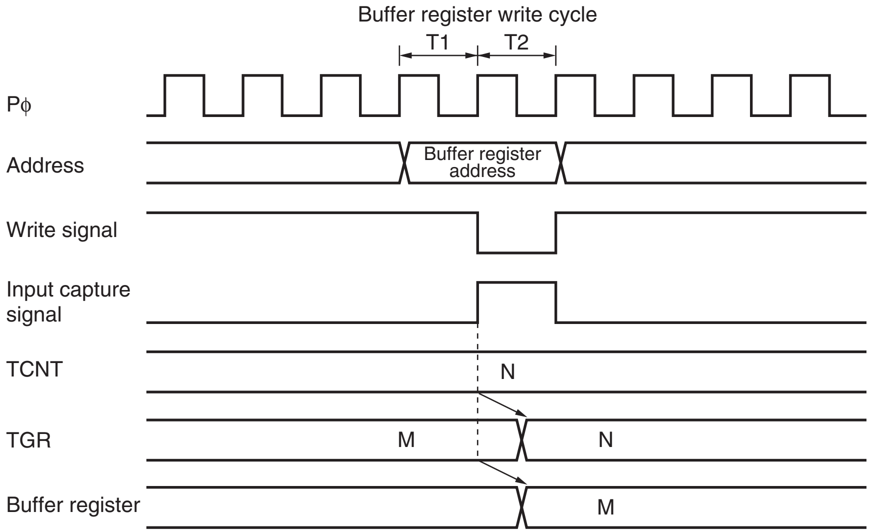


Figure 8.77 Contention between Buffer Register Write and Input Capture

8.7.11 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 D_0 carry out the input capture operation. In addition, when the compare match/input capture is selected as the input capture source of TGRB_1, TGRB_1 carries out input capture operation. The timing is shown in figure 8.78.

For cascade connections, be sure to synchronize settings for channels 1 and 2 when setting TCNT clearing.

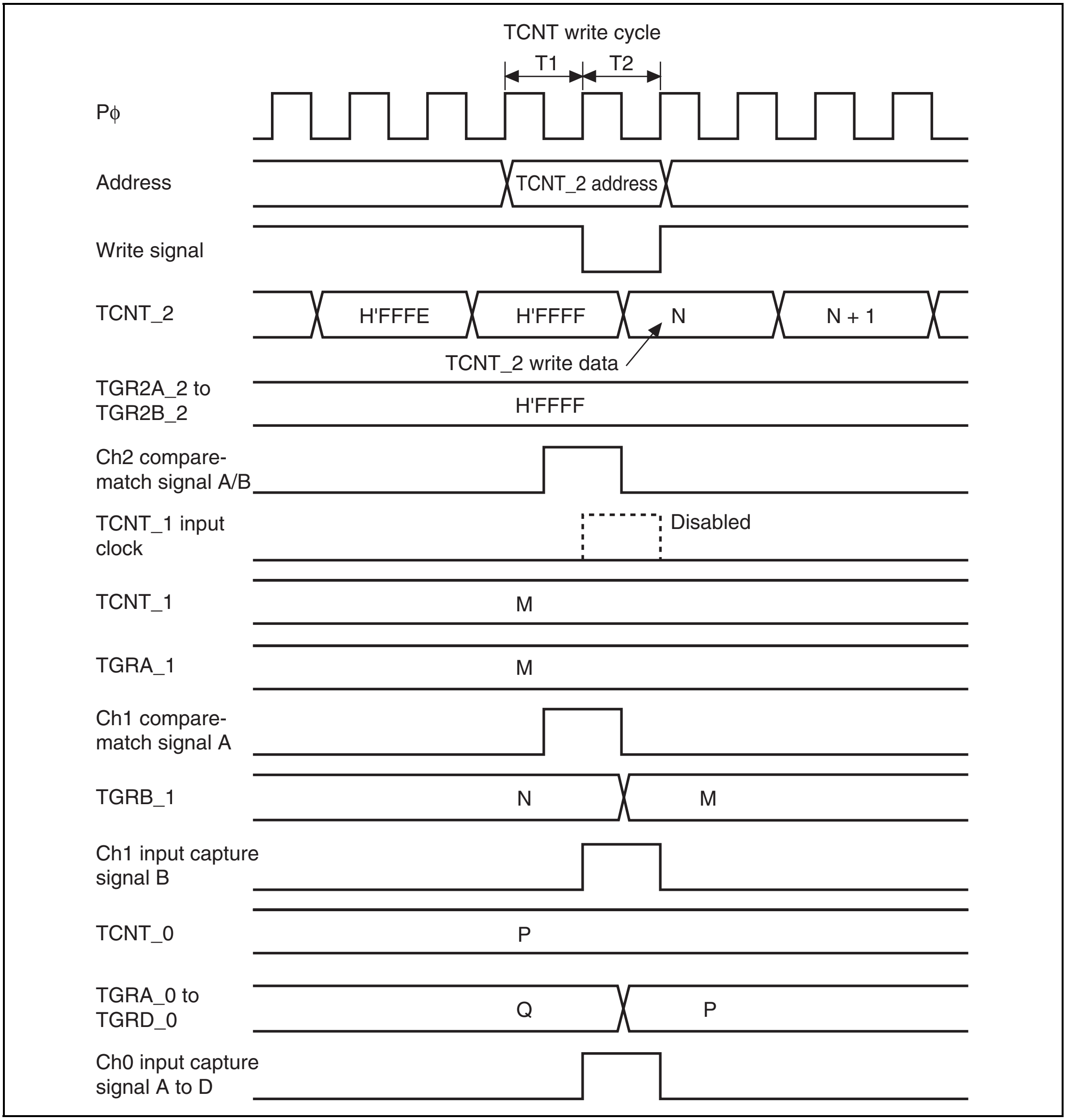


Figure 8.78 TCNT_2 Write and Overflow/Underflow Contention with Cascade Connection

8.7.12 Counter Value during Complementary PWM Mode Stop

When counting operation is stopped with TCNT_3 and TCNT_4 in complementary PWM mode, TCNT_3 has the timer dead time register (TDDR) value, and TCNT_4 is set to H'0000.

When restarting complementary PWM mode, counting begins automatically from the initialized state. This explanatory diagram is shown in figure 8.79.

When counting begins in another operating mode, be sure that TCNT_3 and TCNT_4 are set to the initial values.

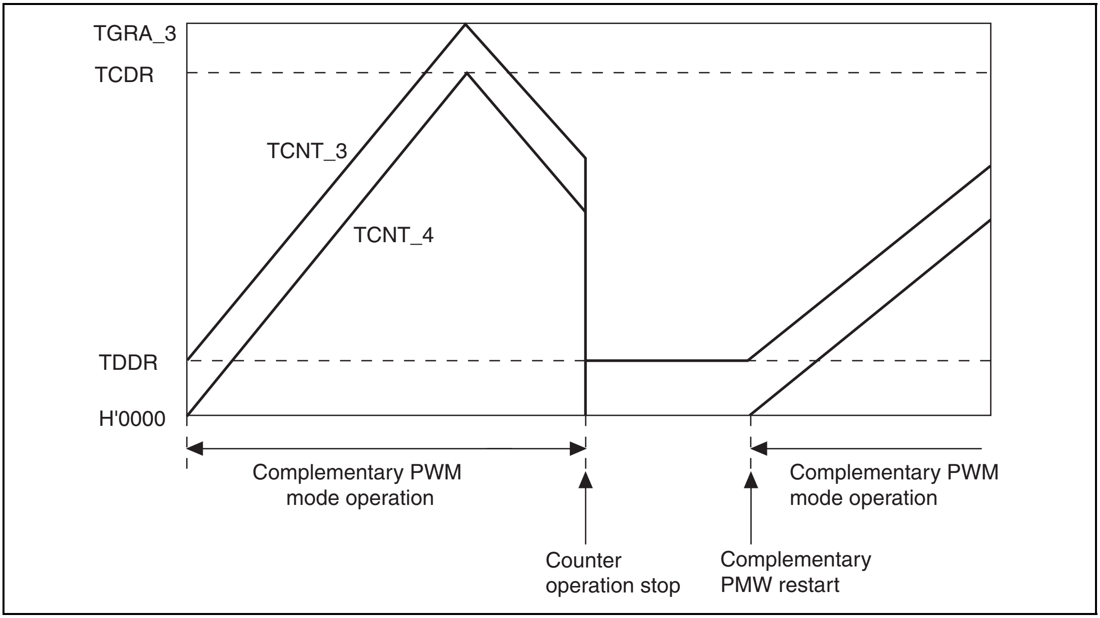


Figure 8.79 Counter Value during Complementary PWM Mode Stop

8.7.13 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 cycle setting registers (TGRB_3, TRGA_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 TRGA_4, while the TCBR functions as the TCDR's buffer register.

8.7.14 Reset Sync PWM Mode Buffer Operation and Compare Match Flag

When setting buffer operation for reset sync PWM mode, set the BFA and BFB bits of TMDR_4 to 0. The TIOC4C pin will be unable to produce its waveform output if the BFA bit of TMDR_4 is set to 1.

In reset sync PWM mode, the channel 3 and channel 4 buffers operate in accordance with the BFA and BFB bit settings of TMDR_3. For example, if the BFA bit of TMDR_3 is set to 1, TGRC_3 functions as the buffer register for TGRA_3. At the same time, TGRC_4 functions as the buffer register for TRGA_4.

The TGFC bit and TGFD bit of TSR_3 and TSR_4 are not set when TGRC_3 and TGRD_3 are operating as buffer registers.

Figure 8.80 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.

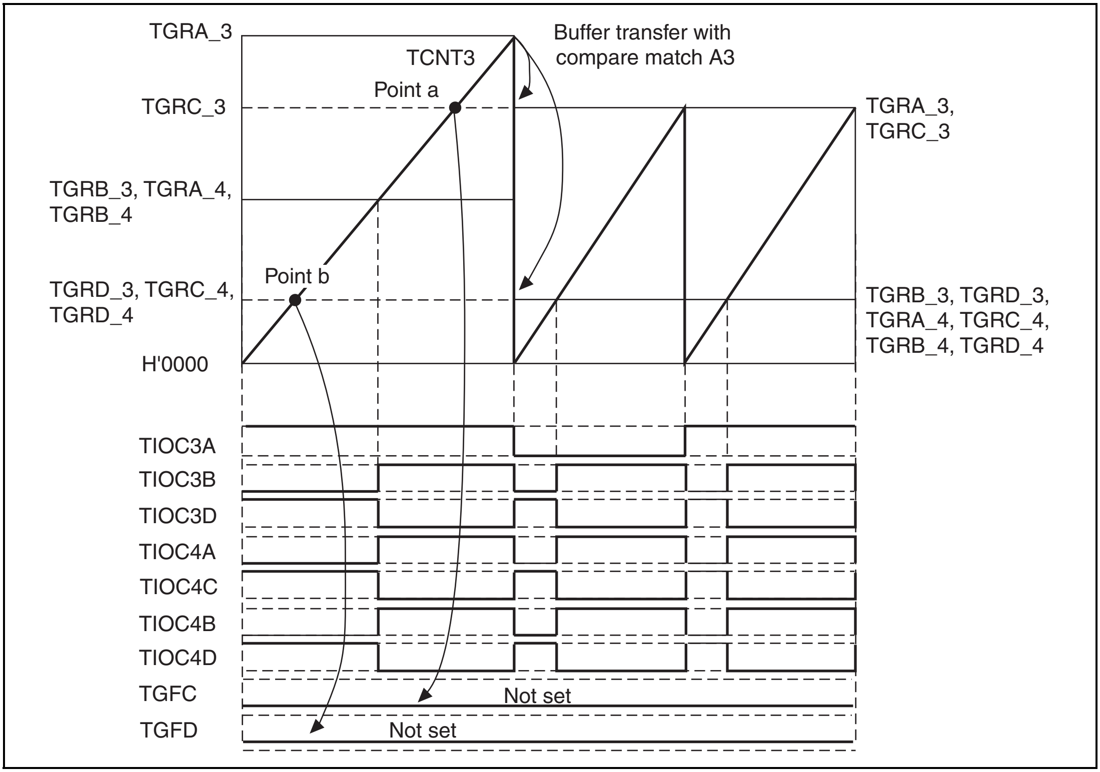


Figure 8.80 Buffer Operation and Compare-Match Flags in Reset Sync PWM Mode

8.7.15 Overflow Flags in Reset Sync PWM Mode

When set to reset sync 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 sync 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 8.81 shows a TCFV bit operation example in reset sync 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.

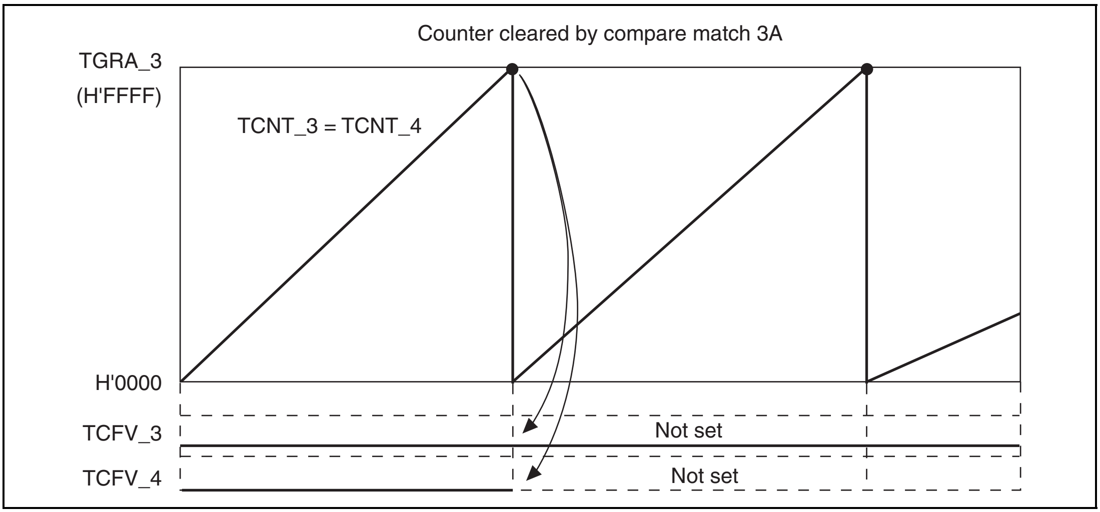


Figure 8.81 Reset Sync PWM Mode Overflow Flag

8.7.16 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 8.82 shows the operation timing when a TGR compare match is specified as the clearing source, and when H'FFFF is set in TGR.

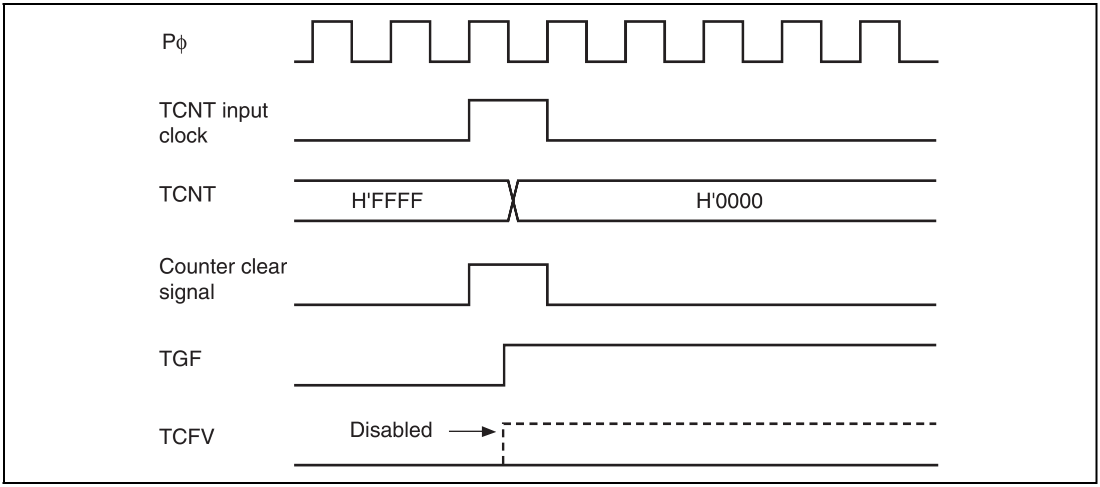


Figure 8.82 Contention between Overflow and Counter Clearing

8.7.17 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 8.83 shows the operation timing when there is contention between TCNT write and overflow.

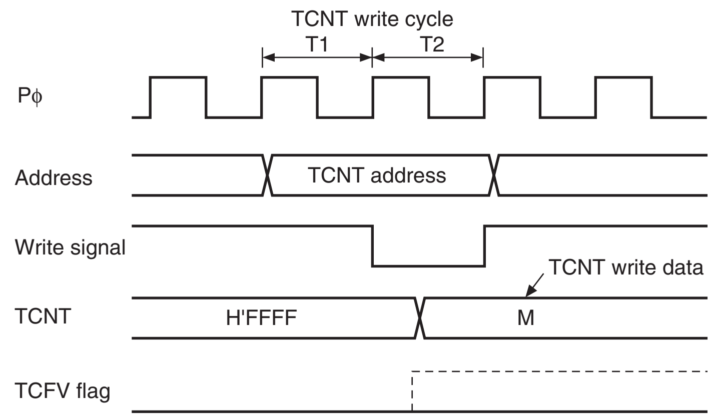


Figure 8.83 Contention between TCNT Write and Overflow

8.7.18 Cautions on Transition from Normal Operation or PWM Mode 1 to Reset-Synchronous PWM Mode

When making a transition from channel 3 or 4 normal operation or PWM mode 1 to reset-synchronous PWM mode, if the counter is halted with the output pins (TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, TIOC4D) in the high-impedance state, followed by the transition to reset-synchronous PWM mode and operation in that mode, the initial pin output will not be correct.

When making a transition from normal operation to reset-synchronous PWM mode, write H'11 to registers TIOR3_H, TIOR3_L, TIOR4_H, and TIOR4_L 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-synchronous 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-synchronous PWM mode.

8.7.19 Output Level in Complementary PWM Mode and Reset-Synchronous PWM Mode

When channels 3 and 4 are in complementary PWM mode or reset-synchronous 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-synchronous PWM mode, TIOR should be set to H'00.

8.7.20 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. Interrupts should therefore be disabled before entering module standby mode.

8.7.21 Simultaneous Input 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.

8.8 MTU Output Pin Initialization

8.8.1 Operating Modes

The MTU 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-synchronous PWM mode (channels 3 and 4)

The MTU output pin initialization method for each of these modes is described in this section.

8.8.2 Reset Start Operation

The MTU output pins (TIOC*) are initialized low by a reset or in standby mode. Since MTU pin function selection is performed by the pin function controller (PFC), when the PFC is set, the MTU pin states at that point are output to the ports. When MTU output is selected by the PFC immediately after a reset, the MTU 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 MTU output pins is completed.

Note: Channel number and port notation are substituted for *.

8.8.3 Operation in Case of Re-Setting Due to Error During Operation, Etc.

If an error occurs during MTU operation, MTU 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 MTU 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 8.43.

Table 8.43 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-synchronous PWM mode

The above abbreviations are used in some places in following descriptions.

8.8.4 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, and 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 8.43. The active level is assumed to be low.

(1) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Normal Mode: Figure 8.84 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in normal mode after re-setting.

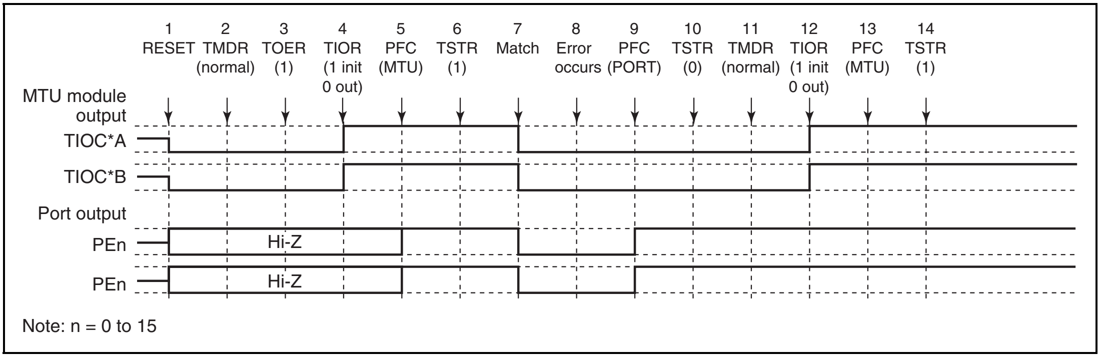


Figure 8.84 Error Occurrence in Normal Mode, Recovery in Normal Mode

1. After a reset, MTU 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 MTU 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 MTU output with the PFC.
14. Operation is restarted by TSTR.

(2) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in PWM Mode 1: Figure 8.85 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.

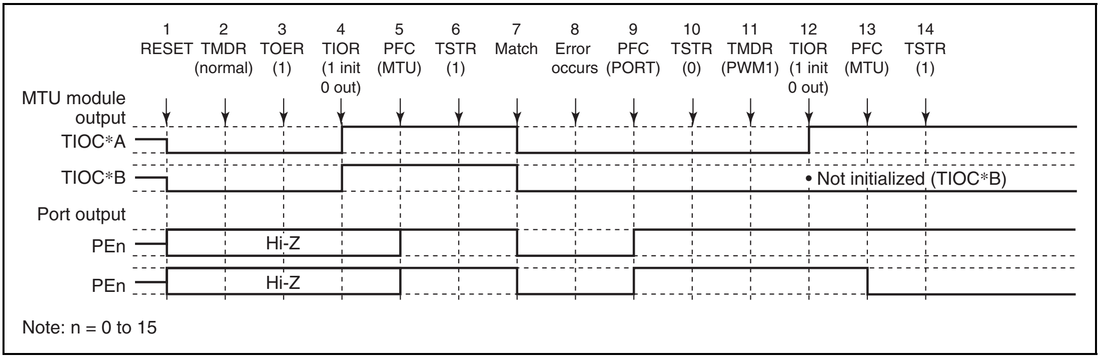


Figure 8.85 Error Occurrence in Normal Mode, Recovery in PWM Mode 1

1 to 10 are the same as in figure 8.84.

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 MTU output with the PFC.
14. Operation is restarted by TSTR.

(3) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in PWM Mode 2: Figure 8.86 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.

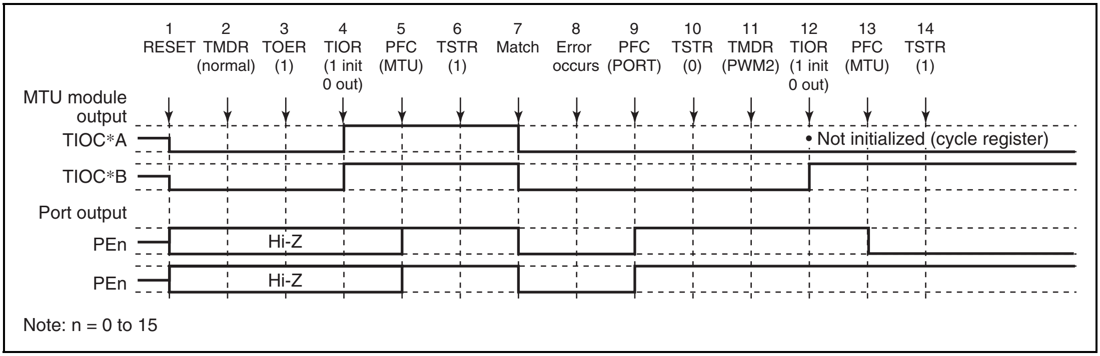


Figure 8.86 Error Occurrence in Normal Mode, Recovery in PWM Mode 2

1 to 10 are the same as in figure 8.84.

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 MTU 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.

(4) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Phase Counting Mode: Figure 8.87 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.

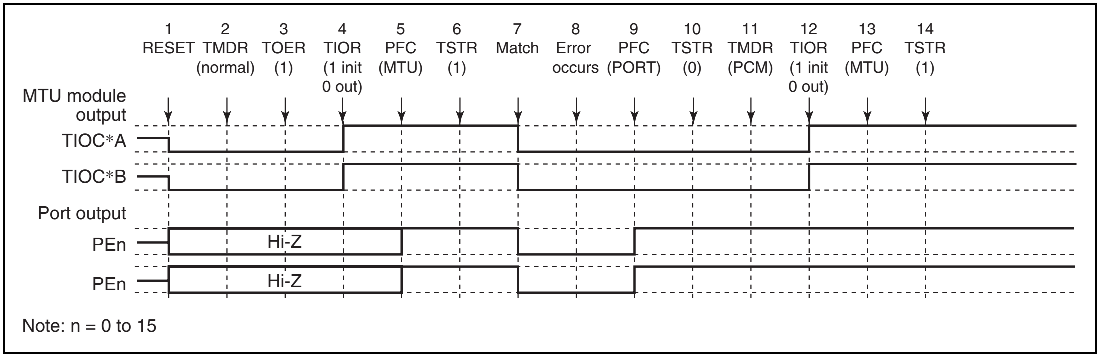


Figure 8.87 Error Occurrence in Normal Mode, Recovery in Phase Counting Mode

1 to 10 are the same as in figure 8.84.

11. Set phase counting mode.
12. Initialize the pins with TIOR.
13. Set MTU 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.

(5) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Complementary PWM Mode: Figure 8.88 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.

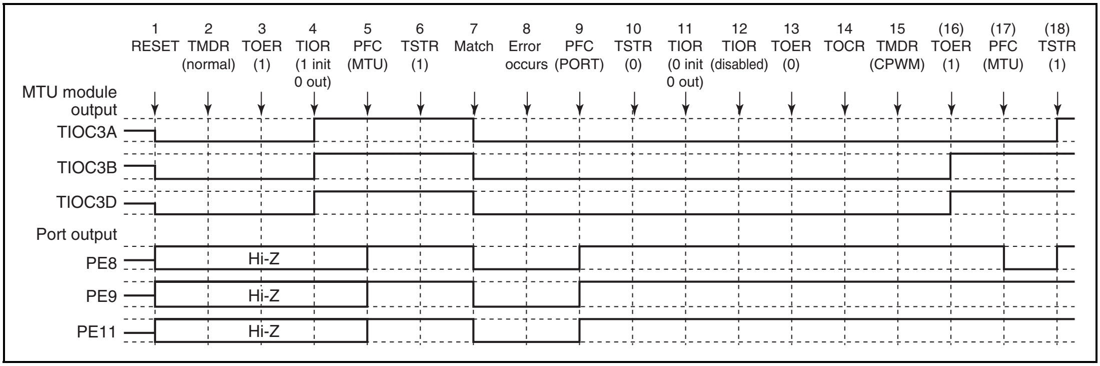


Figure 8.88 Error Occurrence in Normal Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 8.84.

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 MTU output with the PFC.
18. Operation is restarted by TSTR.

(6) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Reset-Synchronous PWM Mode: Figure 8.89 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in reset-synchronous PWM mode after re-setting.

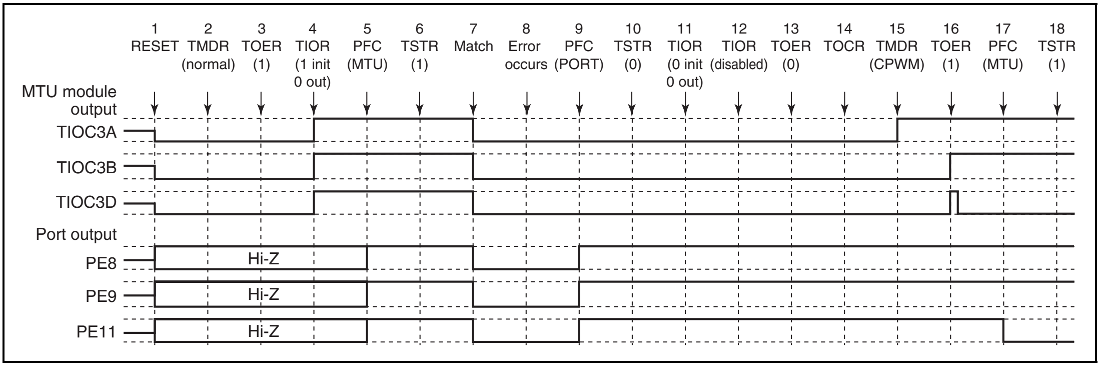


Figure 8.89 Error Occurrence in Normal Mode, Recovery in Reset-Synchronous PWM Mode

1 to 13 are the same as in figure 8.88.

14. Select the reset-synchronous PWM output level and cyclic output enabling/disabling with TOCR.
15. Set reset-synchronous PWM.
16. Enable channel 3 and 4 output with TOER.
17. Set MTU output with the PFC.
18. Operation is restarted by TSTR.

(7) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Normal Mode: Figure 8.90 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.

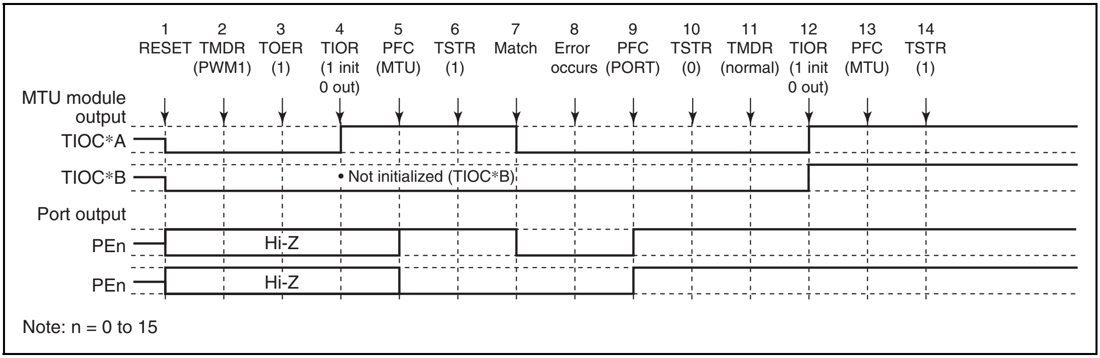


Figure 8.90 Error Occurrence in PWM Mode 1, Recovery in Normal Mode

1. After a reset, MTU 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 MTU 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 MTU output with the PFC.
14. Operation is restarted by TSTR.

(8) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in PWM Mode 1: Figure 8.91 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.

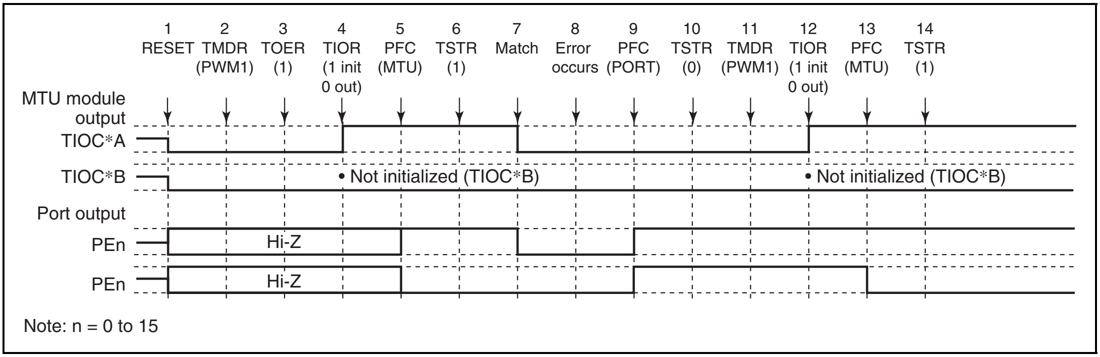


Figure 8.91 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 1

1 to 10 are the same as in figure 8.90.

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 MTU output with the PFC.
14. Operation is restarted by TSTR.

(9) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in PWM Mode 2: Figure 8.92 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.

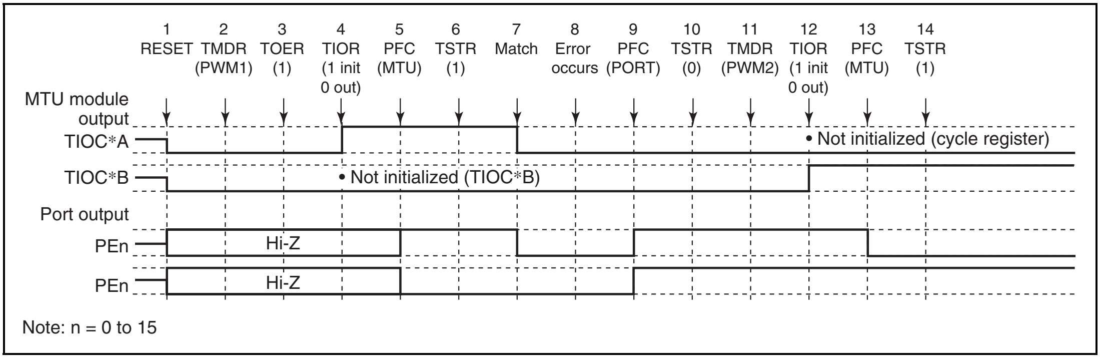


Figure 8.92 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 2

1 to 10 are the same as in figure 8.90.

11. Set PWM mode 2.
12. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
13. Set MTU 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.

(10) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Phase Counting Mode: Figure 8.93 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.

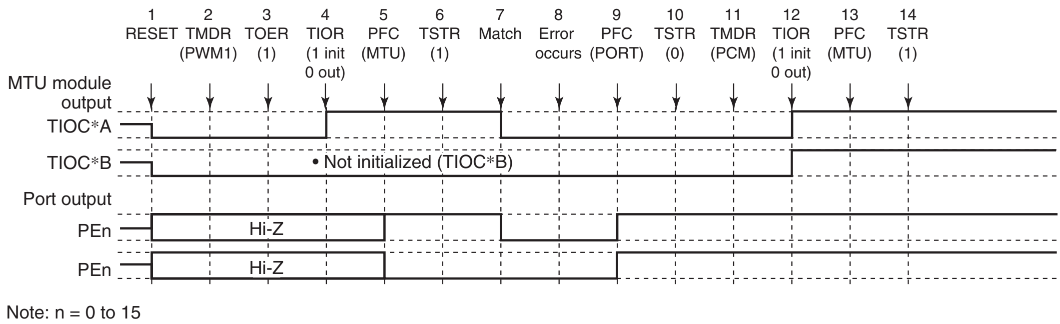


Figure 8.93 Error Occurrence in PWM Mode 1, Recovery in Phase Counting Mode

1 to 10 are the same as in figure 8.90.

11. Set phase counting mode.
12. Initialize the pins with TIOR.
13. Set MTU 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.

(11) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Complementary PWM Mode: Figure 8.94 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.

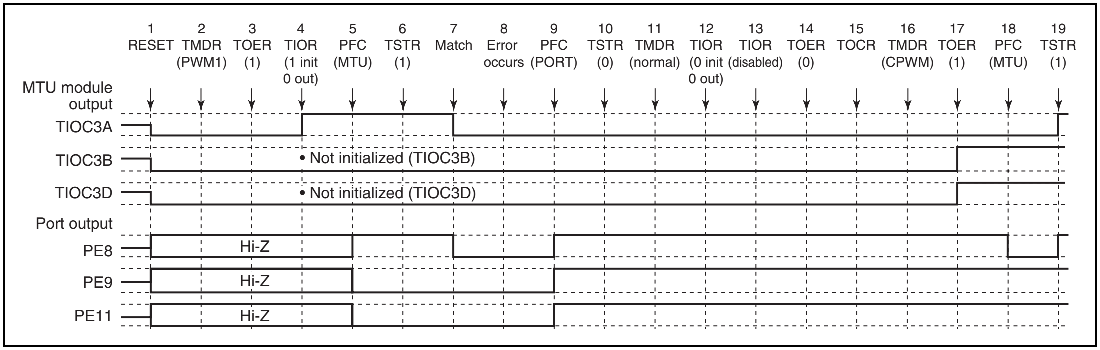


Figure 8.94 Error Occurrence in PWM Mode 1, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 8.90.

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 MTU output with the PFC.
19. Operation is restarted by TSTR.

(12) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Reset-Synchronous PWM Mode: Figure 8.95 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in reset-synchronous PWM mode after re-setting.

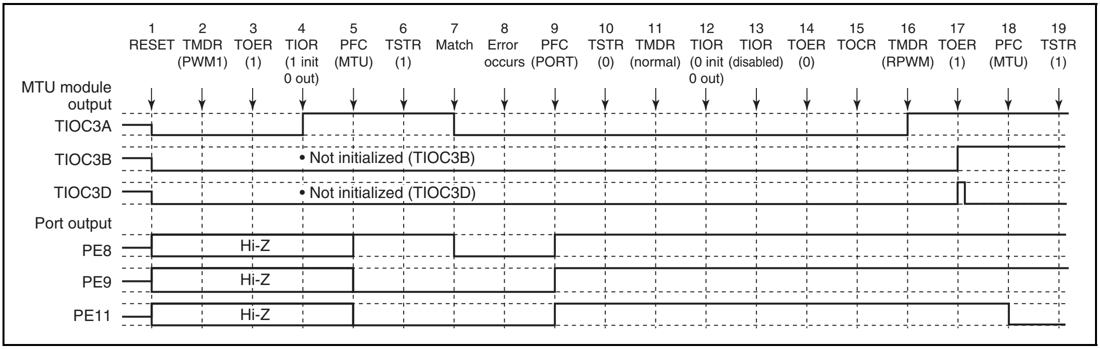


Figure 8.95 Error Occurrence in PWM Mode 1, Recovery in Reset-Synchronous PWM Mode

1 to 14 are the same as in figure 8.90.

15. Select the reset-synchronous PWM output level and cyclic output enabling/disabling with TOCR.
16. Set reset-synchronous PWM.
17. Enable channel 3 and 4 output with TOER.
18. Set MTU output with the PFC.
19. Operation is restarted by TSTR.

(13) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in Normal Mode: Figure 8.96 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.

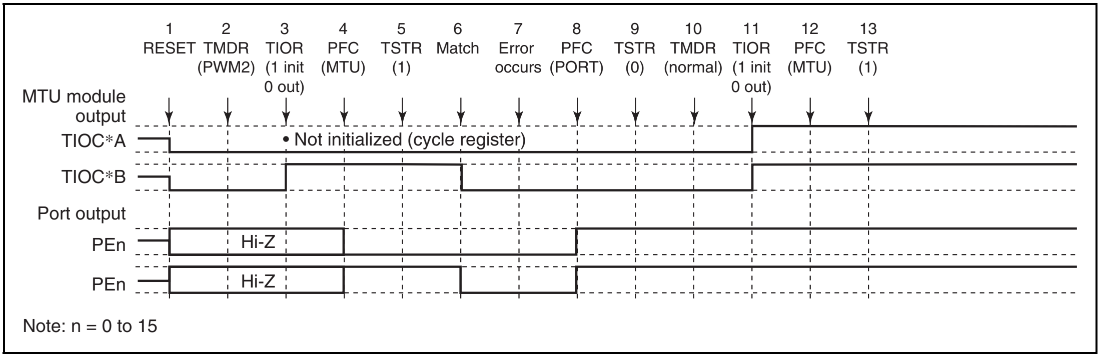


Figure 8.96 Error Occurrence in PWM Mode 2, Recovery in Normal Mode

1. After a reset, MTU 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 MTU 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 MTU output with the PFC.
13. Operation is restarted by TSTR.

(14) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in PWM Mode 1: Figure 8.97 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.

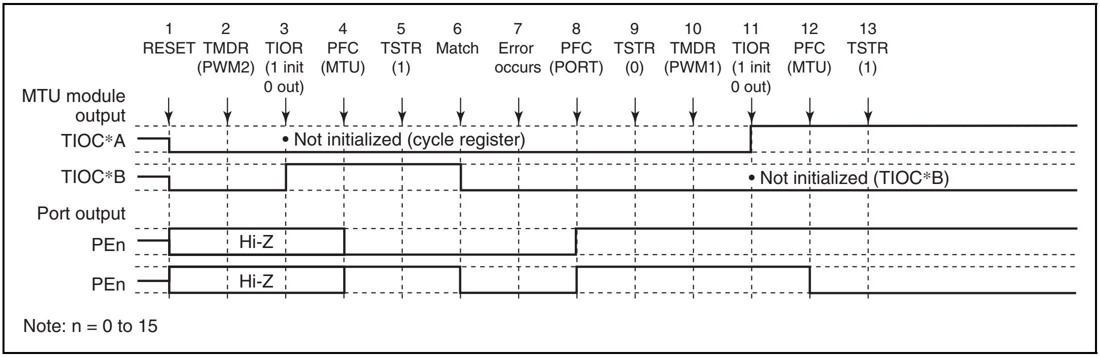


Figure 8.97 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 1

1 to 9 are the same as in figure 8.96.

10. Set PWM mode 1.
11. Initialize the pins with TIOR. (In PWM mode 1, the TIOC*B side is not initialized.)
12. Set MTU output with the PFC.
13. Operation is restarted by TSTR.

(15) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in PWM Mode 2: Figure 8.98 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.

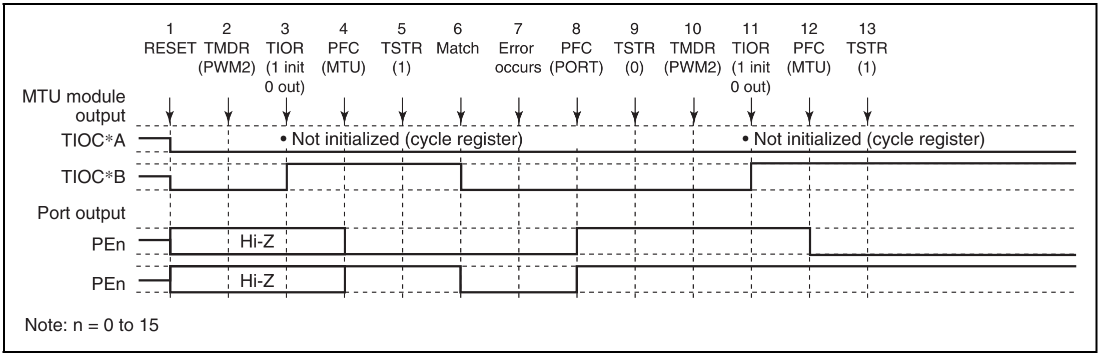


Figure 8.98 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 2

1 to 9 are the same as in figure 8.96.

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 MTU output with the PFC.
13. Operation is restarted by TSTR.

(16) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in Phase Counting Mode: Figure 8.99 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.

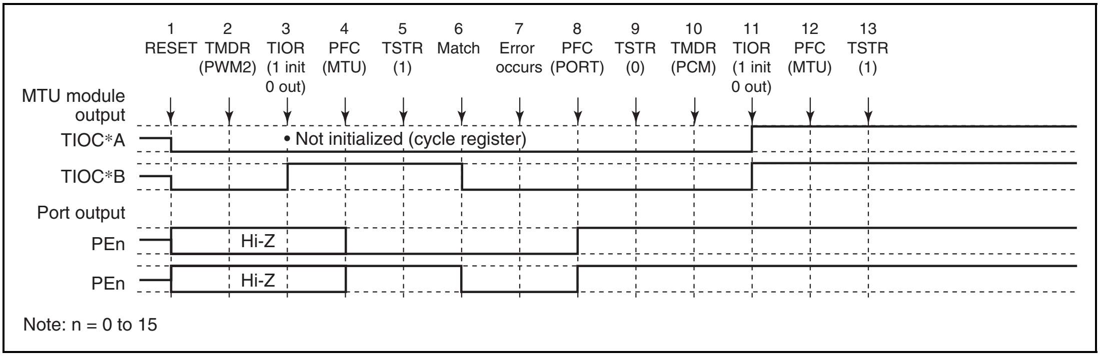


Figure 8.99 Error Occurrence in PWM Mode 2, Recovery in Phase Counting Mode

1 to 9 are the same as in figure 8.96.

10. Set phase counting mode.
11. Initialize the pins with TIOR.
12. Set MTU output with the PFC.
13. Operation is restarted by TSTR.

(17) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in Normal Mode: Figure 8.100 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.

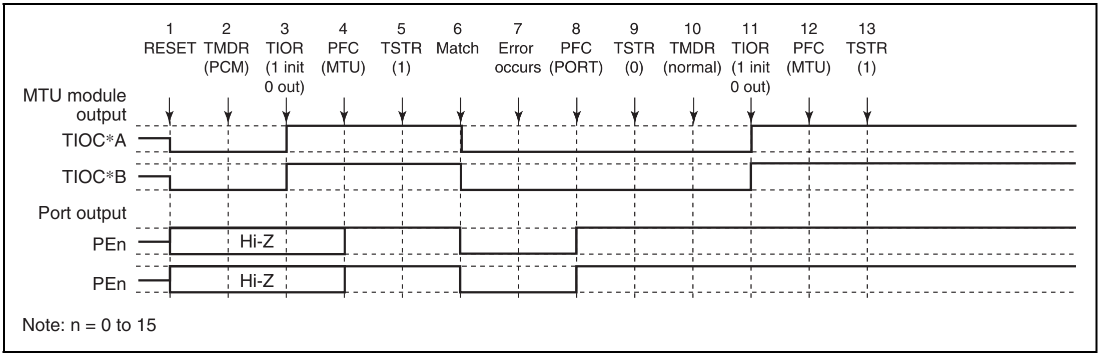


Figure 8.100 Error Occurrence in Phase Counting Mode, Recovery in Normal Mode

1. After a reset, MTU 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 MTU 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 MTU output with the PFC.
13. Operation is restarted by TSTR.

(18) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in PWM Mode 1: Figure 8.101 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.

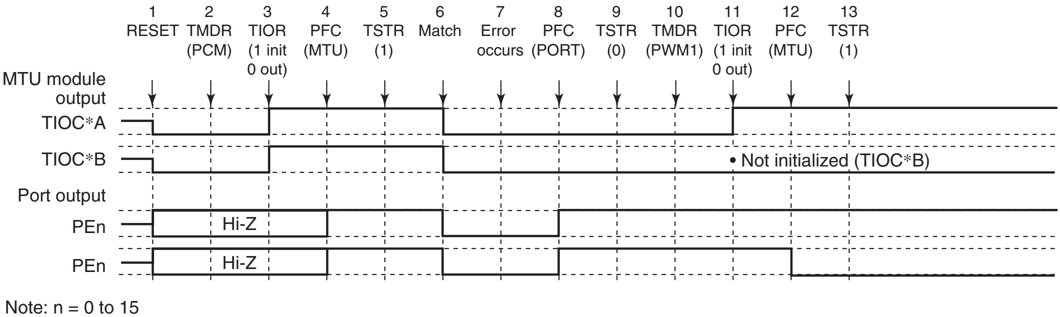


Figure 8.101 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 1

1 to 9 are the same as in figure 8.100.

10. Set PWM mode 1.
11. Initialize the pins with TIOR. (In PWM mode 1, the TIOC *B side is not initialized.)
12. Set MTU output with the PFC.
13. Operation is restarted by TSTR.

(19) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in PWM Mode 2: Figure 8.102 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.

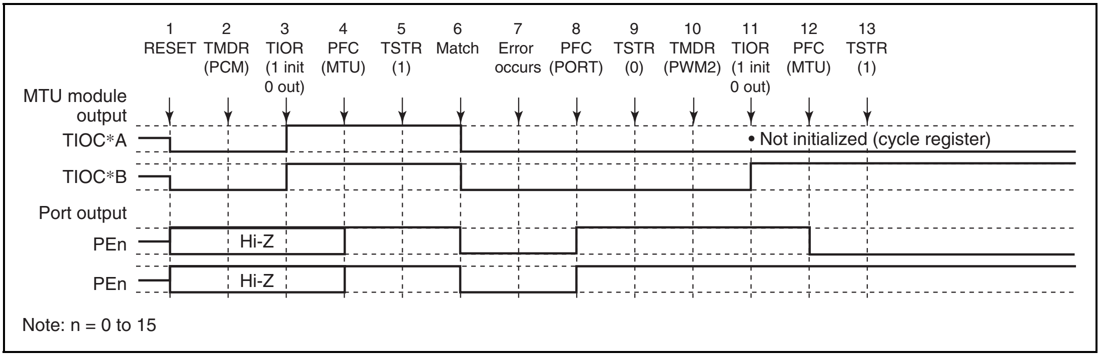


Figure 8.102 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 2

1 to 9 are the same as in figure 8.100.

10. Set PWM mode 2.
11. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
12. Set MTU output with the PFC.
13. Operation is restarted by TSTR.

(20) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in Phase Counting Mode: Figure 8.103 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.

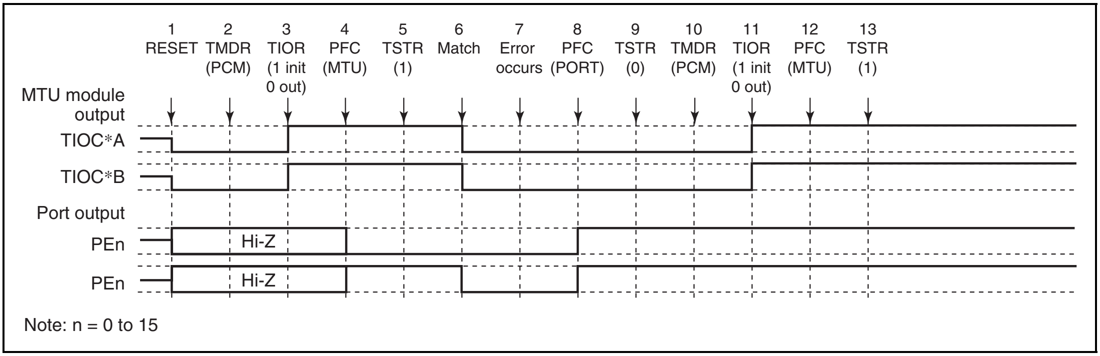


Figure 8.103 Error Occurrence in Phase Counting Mode, Recovery in Phase Counting Mode

1 to 9 are the same as in figure 8.100.

10. Not necessary when restarting in phase counting mode.
11. Initialize the pins with TIOR.
12. Set MTU output with the PFC.
13. Operation is restarted by TSTR.

(21) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Normal Mode: Figure 8.104 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.

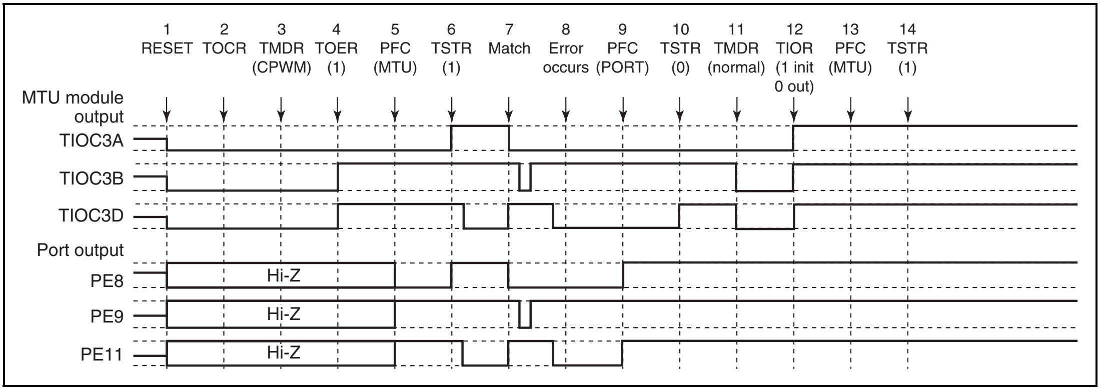


Figure 8.104 Error Occurrence in Complementary PWM Mode, Recovery in Normal Mode

1. After a reset, MTU 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 MTU 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. (MTU output becomes the complementary PWM output initial value.)
11. Set normal mode. (MTU output goes low.)
12. Initialize the pins with TIOR.
13. Set MTU output with the PFC.
14. Operation is restarted by TSTR.

(22) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in PWM Mode 1: Figure 8.105 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.

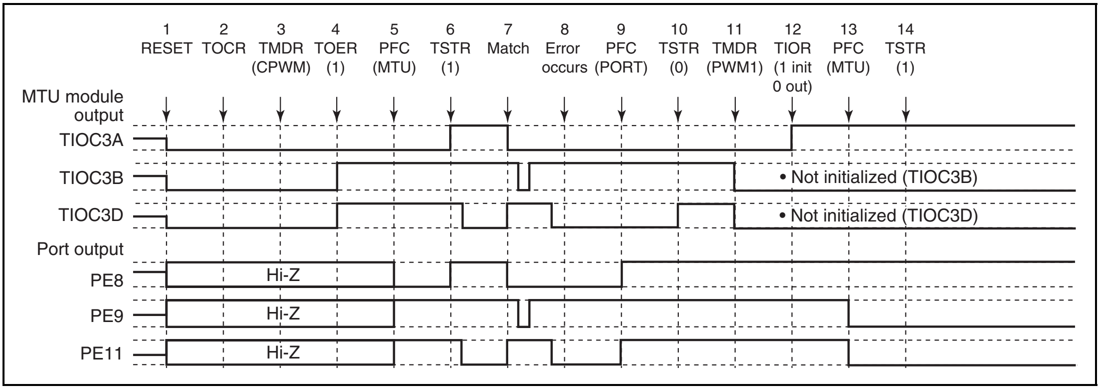


Figure 8.105 Error Occurrence in Complementary PWM Mode, Recovery in PWM Mode 1

1 to 10 are the same as in figure 8.104.

11. Set PWM mode 1. (MTU output goes low.)
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC *B side is not initialized.)
13. Set MTU output with the PFC.
14. Operation is restarted by TSTR.

(23) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode: Figure 8.106 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 cycle settings at the time the counter was stopped).

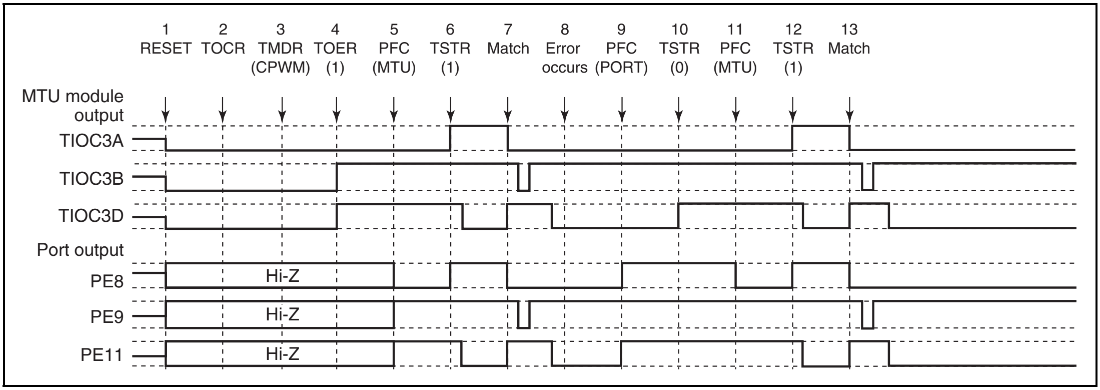


Figure 8.106 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 8.104.

11. Set MTU output with the PFC.
12. Operation is restarted by TSTR.
13. The complementary PWM waveform is output on compare-match occurrence.

(24) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode: Figure 8.107 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 cycle settings).

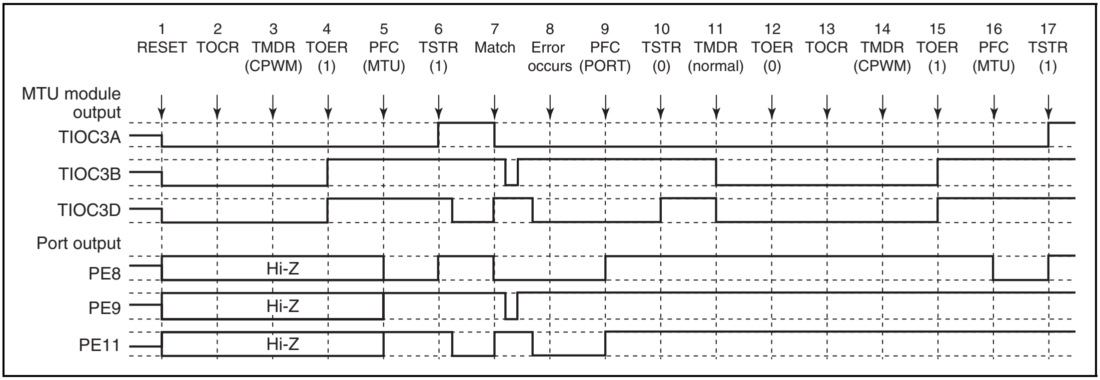


Figure 8.107 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 8.104.

11. Set normal mode and make new settings. (MTU 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 MTU output with the PFC.
17. Operation is restarted by TSTR.

(25) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Reset-Synchronous PWM Mode: Figure 8.108 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in reset-synchronous PWM mode.

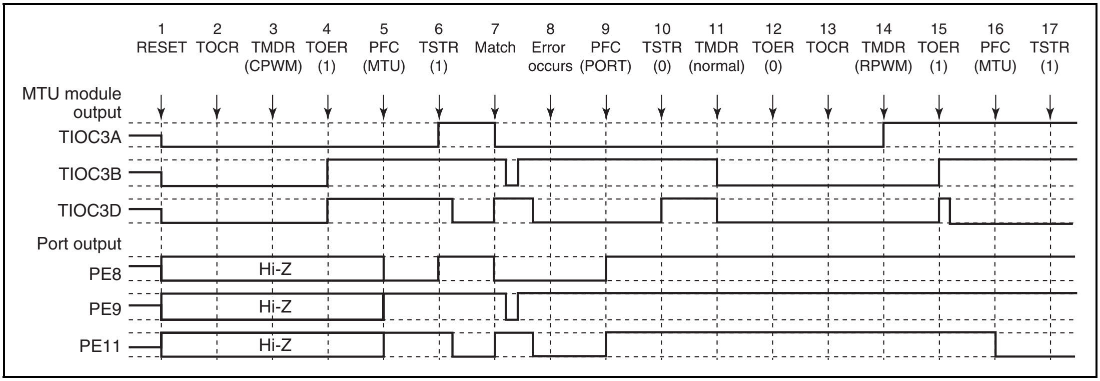


Figure 8.108 Error Occurrence in Complementary PWM Mode, Recovery in Reset-Synchronous PWM Mode

1 to 10 are the same as in figure 8.104.

11. Set normal mode. (MTU output goes low.)
12. Disable channel 3 and 4 output with TOER.
13. Select the reset-synchronous PWM mode output level and cyclic output enabling/disabling with TOCR.
14. Set reset-synchronous PWM.
15. Enable channel 3 and 4 output with TOER.
16. Set MTU output with the PFC.
17. Operation is restarted by TSTR.

(26) Operation when Error Occurs during Reset-Synchronous PWM Mode Operation, and Operation is Restarted in Normal Mode: Figure 8.109 shows an explanatory diagram of the case where an error occurs in reset-synchronous PWM mode and operation is restarted in normal mode after re-setting.

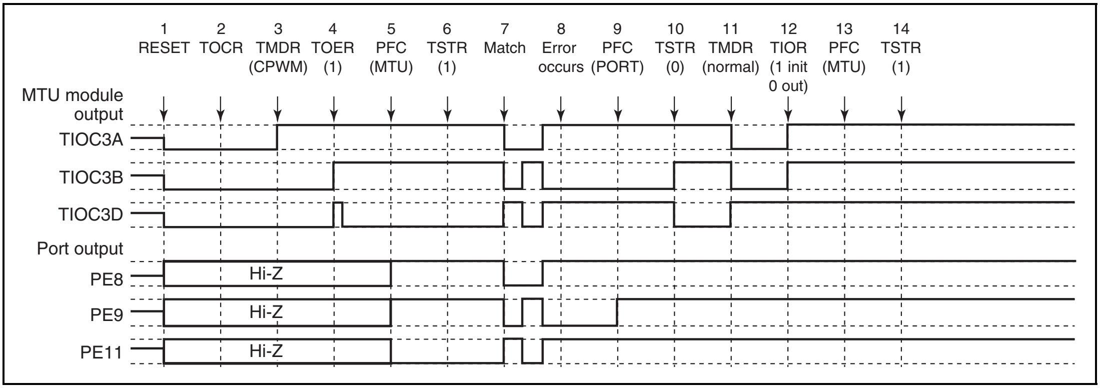


Figure 8.109 Error Occurrence in Reset-Synchronous PWM Mode, Recovery in Normal Mode

1. After a reset, MTU output is low and ports are in the high-impedance state.
2. Select the reset-synchronous PWM output level and cyclic output enabling/disabling with TOCR.
3. Set reset-synchronous PWM.
4. Enable channel 3 and 4 output with TOER.
5. Set MTU output with the PFC.
6. The count operation is started by TSTR.
7. The reset-synchronous 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. (MTU output becomes the reset-synchronous PWM output initial value.)
11. Set normal mode. (MTU positive phase output is low, and negative phase output is high.)
12. Initialize the pins with TIOR.
13. Set MTU output with the PFC.
14. Operation is restarted by TSTR.

(27) Operation when Error Occurs during Reset-Synchronous PWM Mode Operation, and Operation is Restarted in PWM Mode 1: Figure 8.110 shows an explanatory diagram of the case where an error occurs in reset-synchronous PWM mode and operation is restarted in PWM mode 1 after re-setting.

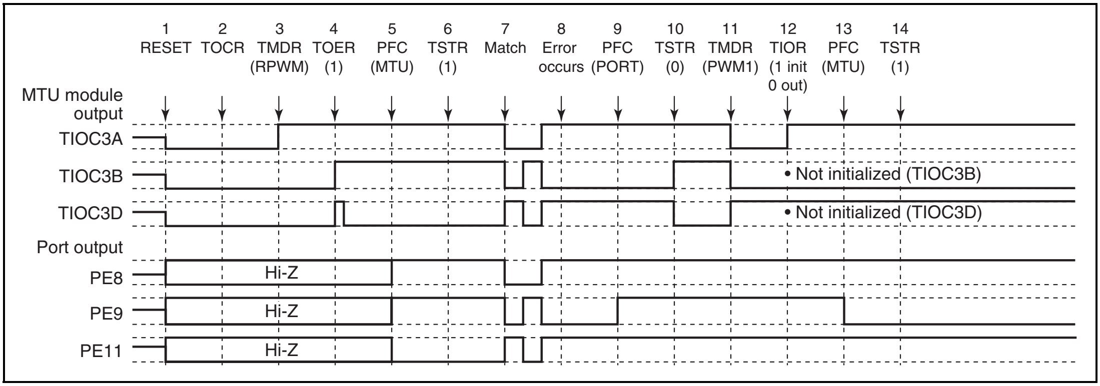


Figure 8.110 Error Occurrence in Reset-Synchronous PWM Mode, Recovery in PWM Mode 1

1 to 10 are the same as in figure 8.109.

11. Set PWM mode 1. (MTU 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 MTU output with the PFC.
14. Operation is restarted by TSTR.

(28) Operation when Error Occurs during Reset-Synchronous PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode: Figure 8.111 shows an explanatory diagram of the case where an error occurs in reset-synchronous PWM mode and operation is restarted in complementary PWM mode after re-setting.

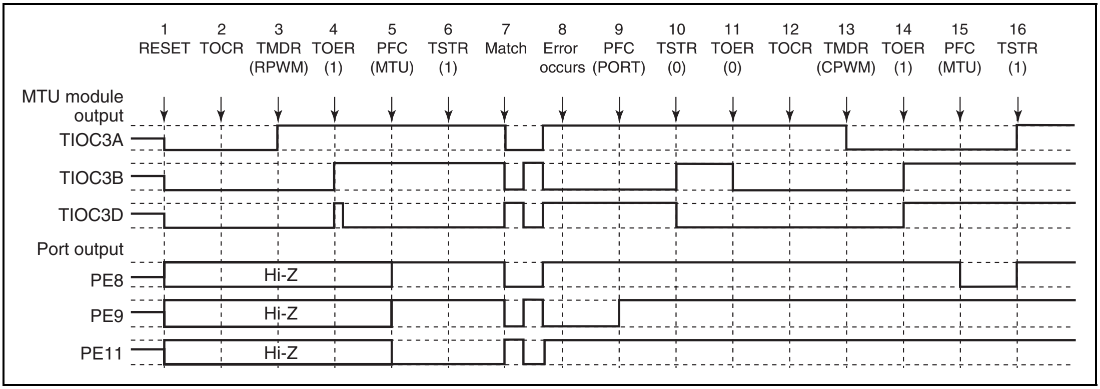


Figure 8.111 Error Occurrence in Reset-Synchronous PWM Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 8.109.

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 MTU cyclic output pin goes low.)
14. Enable channel 3 and 4 output with TOER.
15. Set MTU output with the PFC.
16. Operation is restarted by TSTR.

(29) Operation when Error Occurs during Reset-Synchronous PWM Mode Operation, and Operation is Restarted in Reset-Synchronous PWM Mode: Figure 8.112 shows an explanatory diagram of the case where an error occurs in reset-synchronous PWM mode and operation is restarted in reset-synchronous PWM mode after re-setting.

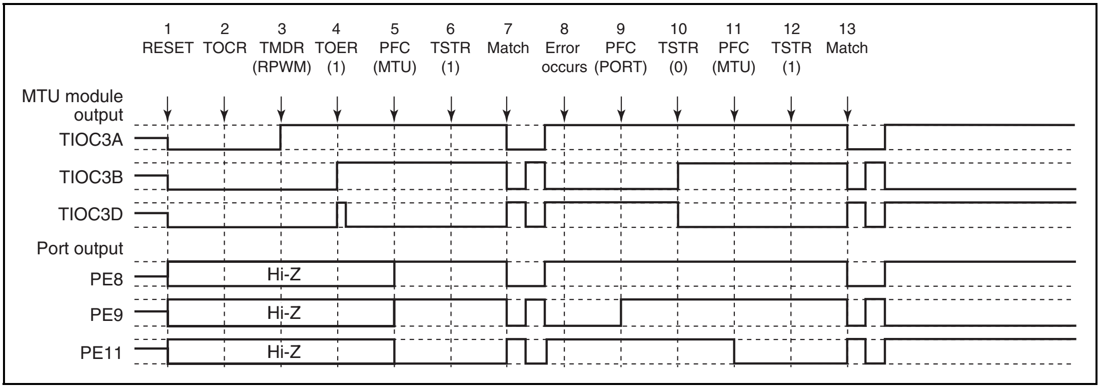


Figure 8.112 Error Occurrence in Reset-Synchronous PWM Mode, Recovery in Reset-Synchronous PWM Mode

1 to 10 are the same as in figure 8.109.

11. Set MTU output with the PFC.
12. Operation is restarted by TSTR.
13. The reset-synchronous PWM waveform is output on compare-match occurrence.

8.9 Port Output Enable (POE)

The port output enable (POE) can be used to establish a high-impedance state for high-current pins, by changing the $\overline{\text{POE0}}$ to $\overline{\text{POE3}}$ pin input, depending on the output status of the high-current pins (PE9/TIOC3B, PE11/TIOC3D, PE12/TIOC4A, PE13/TIOC4B/$\overline{\text{MRES}}$, PE14/TIOC4C, PE15/TIOC4D/$\overline{\text{IRQOUT}}$). It can also simultaneously generate interrupt requests.

The high-current pins also become high-impedance regardless of whether these pin functions are selected in cases such as when the oscillator stops or in standby mode.

8.9.1 Features

- Each of the $\overline{\text{POE0}}$ to $\overline{\text{POE3}}$ input pins can be set for falling edge, $P\phi/8 \times 16$, $P\phi/16 \times 16$, or $P\phi/128 \times 16$ low-level sampling.
- High-current pins can be set to high-impedance state by $\overline{\text{POE0}}$ to $\overline{\text{POE3}}$ pin falling-edge or low-level sampling.
- High-current pins can be set to high-impedance state when the high-current pin output levels are compared and simultaneous low-level output continues for one cycle or more.
- Interrupts can be generated by input-level sampling or output-level comparison results.

The POE has input-level detection circuitry and output-level detection circuitry, as shown in the block diagram of figure 8.113.

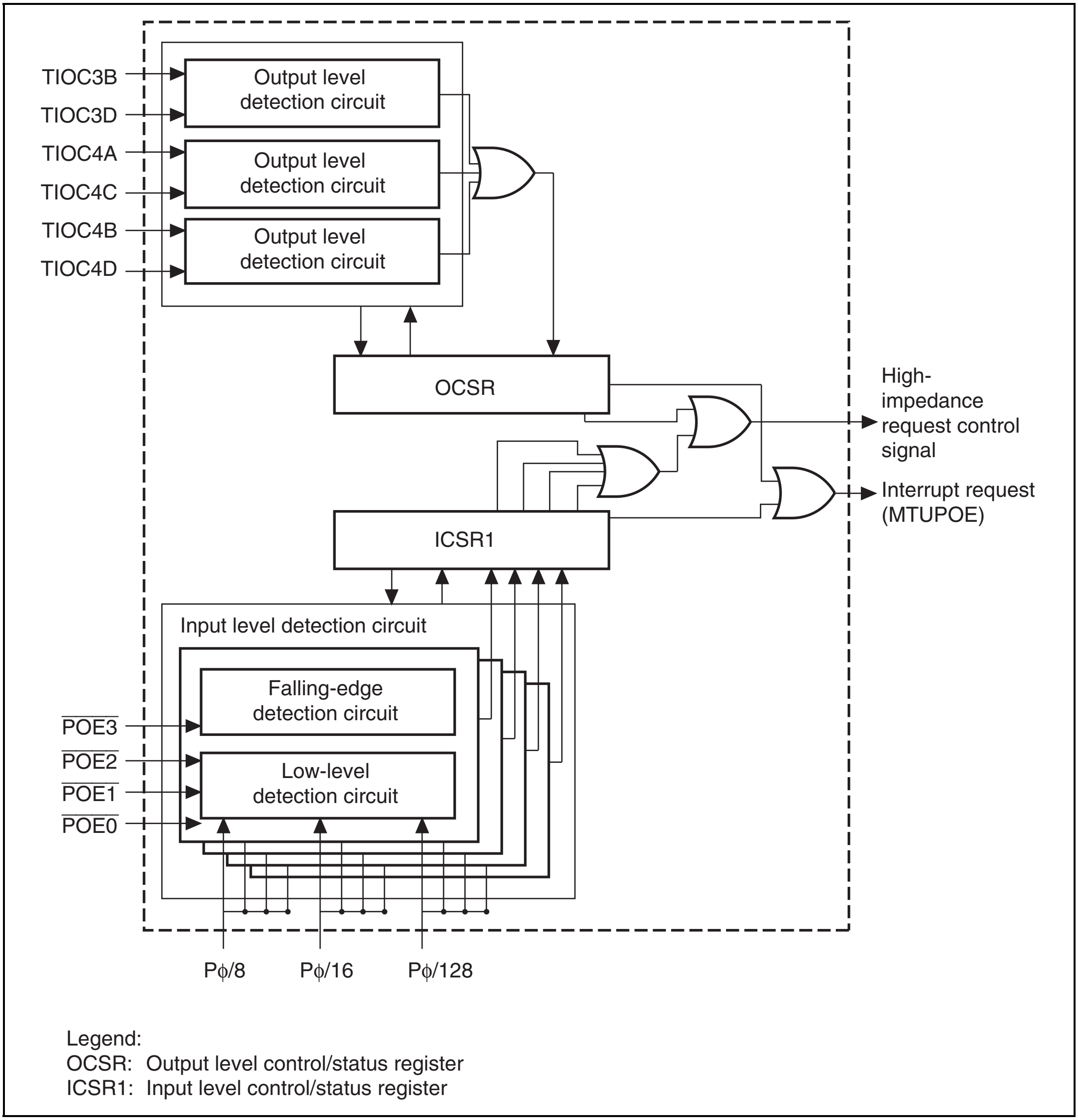


Figure 8.113 POE Block Diagram

8.9.2 Pin Configuration

Table 8.44 Pin Configuration

Name	Abbreviation	I/O	Description
Port output enable input pins	POE0 to POE3	Input	Input request signals to make high-current pins high-impedance state

Table 8.45 shows output-level comparisons with pin combinations.

Table 8.45 Pin Combinations

Pin Combination	I/O	Description
PE9/TIOC3B and PE11/TIOC3D	Output	All high-current pins are made high-impedance state when the pins simultaneously output low-level for longer than 1 cycle.
PE12/TIOC4A and PE14/TIOC4C	Output	All high-current pins are made high-impedance state when the pins simultaneously output low-level for longer than 1 cycle.
PE13/TIOC4B/MRES and PE15/TIOC4D/IRQOUT	Output	All high-current pins are made high-impedance state when the pins simultaneously output low-level for longer than 1 cycle.

8.9.3 Register Configuration

The POE has the two registers. The input level control/status register 1 (ICSR1) controls both POE0 to POE3 pin input signal detection and interrupts. The output level control/status register (OCSR) controls both the enable/disable of output comparison and interrupts.

Input Level Control/Status Register 1 (ICSR1): The input level control/status register (ICSR1) is a 16-bit readable/writable register that selects the POE0 to POE3 pin input modes, controls the enable/disable of interrupts, and indicates status.

Bit	Bit Name	Initial value	R/W	Description
15	POE3F	0	R/(W)*	POE3 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE3}}$ pin [Clearing condition] • By writing 0 to POE3F after reading a POE3F = 1 [Setting condition] • When the input set by ICSR1 bits 7 and 6 occurs at the $\overline{\text{POE3}}$ pin
14	POE2F	0	R/(W)*	POE2 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE2}}$ pin [Clearing condition] • By writing 0 to POE2F after reading a POE2F = 1 [Setting condition] • When the input set by ICSR1 bits 5 and 4 occurs at the $\overline{\text{POE2}}$ pin
13	POE1F	0	R/(W)*	POE1 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE1}}$ pin [Clearing condition] • By writing 0 to POE1F after reading a POE1F = 1 [Setting condition] • When the input set by ICSR1 bits 3 and 2 occurs at the $\overline{\text{POE1}}$ pin
12	POE0F	0	R/(W)*	POE0 Flag This flag indicates that a high impedance request has been input to the $\overline{\text{POE0}}$ pin [Clearing condition] • By writing 0 to POE0F after reading a POE0F = 1 [Setting condition] • When the input set by ICSR1 bits 1 and 0 occurs at the $\overline{\text{POE0}}$ pin

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	PIE	0	R/W	Port Interrupt Enable This bit enables/disables interrupt requests when any of the POE0F to POE3F bits of the ICSR1 are set to 1 0: Interrupt requests disabled 1: Interrupt requests enabled
7 6	POE3M1 POE3M0	0 0	R/W R/W	POE3 mode 1, 0 These bits select the input mode of the $\overline{POE3}$ pin 00: Accept request on falling edge of $\overline{POE3}$ input 01: Accept request when $\overline{POE3}$ input has been sampled for 16 $P\phi/8$ clock pulses, and all are low level. 10: Accept request when $\overline{POE3}$ input has been sampled for 16 $P\phi/16$ clock pulses, and all are low level. 11: Accept request when $\overline{POE3}$ input has been sampled for 16 $P\phi/128$ clock pulses, and all are low level.
5 4	POE2M1 POE2M0	0 0	R/W R/W	POE2 mode 1, 0 These bits select the input mode of the $\overline{POE2}$ pin 00: Accept request on falling edge of $\overline{POE2}$ input 01: Accept request when $\overline{POE2}$ input has been sampled for 16 $P\phi/8$ clock pulses, and all are low level. 10: Accept request when $\overline{POE2}$ input has been sampled for 16 $P\phi/16$ clock pulses, and all are low level. 11: Accept request when $\overline{POE2}$ input has been sampled for 16 $P\phi/128$ clock pulses, and all are low level.
3 2	POE1M1 POE1M0	0 0	R/W R/W	POE1 mode 1, 0 These bits select the input mode of the $\overline{POE1}$ pin 00: Accept request on falling edge of $\overline{POE1}$ input 01: Accept request when $\overline{POE1}$ input has been sampled for 16 $P\phi/8$ clock pulses, and all are low level. 10: Accept request when $\overline{POE1}$ input has been sampled for 16 $P\phi/16$ clock pulses, and all are low level. 11: Accept request when $\overline{POE1}$ input has been sampled for 16 $P\phi/128$ clock pulses, and all are low level.

Bit	Bit Name	Initial value	R/W	Description
1	POE0M1	0	R/W	POE0 mode 1, 0
0	POE0M0	0	R/W	These bits select the input mode of the $\overline{\text{POE0}}$ pin 00: Accept request on falling edge of $\overline{\text{POE0}}$ input 01: Accept request when $\overline{\text{POE0}}$ input has been sampled for 16 Pϕ/8 clock pulses, and all are low level. 10: Accept request when $\overline{\text{POE0}}$ input has been sampled for 16 Pϕ/16 clock pulses, and all are low level. 11: Accept request when $\overline{\text{POE0}}$ input has been sampled for 16 Pϕ/128 clock pulses, and all are low level.

Note: * The write value should always be 0.

Output Level Control/Status Register (OCSR): The output level control/status register (OCSR) is a 16-bit readable/writable register that controls the enable/disable of both output level comparison and interrupts, and indicates status. If the OSF bit is set to 1, the high current pins become high impedance.

Bit	Bit Name	Initial value	R/W	Description
15	OSF	0	R/(W)*	Output Short Flag This flag indicates that any one pair of the three pairs of 2 phase outputs compared have simultaneously become low level outputs. [Clearing condition] • By writing 0 to OSF after reading an OSF = 1 [Setting condition] • When any one pair of the three 2-phase outputs simultaneously become low level
14 to 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	OCE	0	R/W	Output Level Compare Enable This bit enables the start of output level comparisons. When setting this bit to 1, pay attention to the output pin combinations shown in table 8.43, Mode Transition Combinations. When 0 is output, the OSF bit is set to 1 at the same time when this bit is set, and output goes to high impedance. Accordingly, bits 15 to 11 and bit 9 of the port E data register (PEDR) are set to 1. For the MTU output comparison, set the bit to 1 after setting the MTU's output pins with the PFC. Set this bit only when using pins as outputs. When the OCE bit is set to 1, if OIE = 0 a high-impedance request will not be issued even if OSF is set to 1. Therefore, in order to have a high-impedance request issued according to the result of the output level comparison, the OIE bit must be set to 1. When OCE = 1 and OIE = 1, an interrupt request will be generated at the same time as the high-impedance request: however, this interrupt can be masked by means of an interrupt controller (INTC) setting. 0: Output level compare disabled 1: Output level compare enabled; makes an output high impedance request when OSF = 1.
8	OIE	0	R/W	Output Short Interrupt Enable This bit makes interrupt requests when the OSF bit of the OCSR is set. 0: Interrupt requests disabled 1: Interrupt request enabled
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: * The write value should always be 0.

8.9.4 Operation

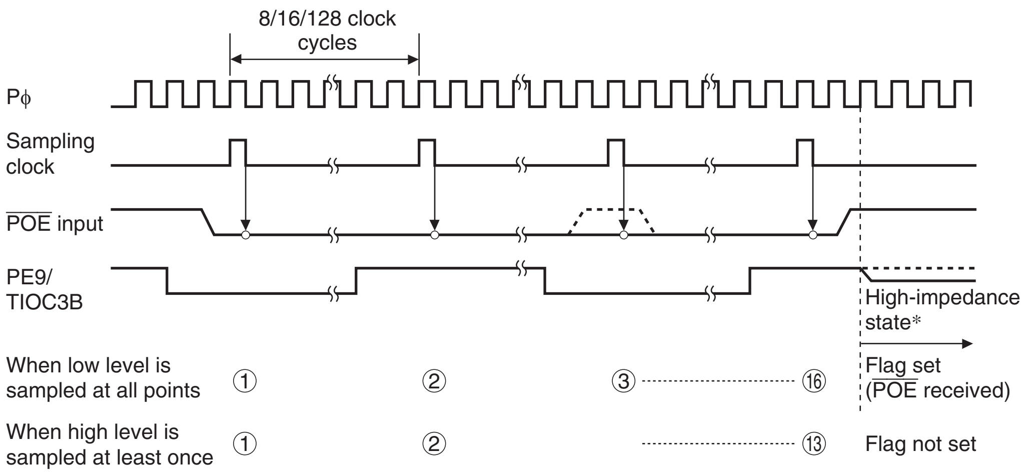
Input Level Detection Operation

If the input conditions set by the ICSR1 occur on any of the $\overline{\text{POE}}$ pins, all high-current pins become high-impedance state. However, only when the general input/output function or MTU function is selected, the large-current pin is in the high-impedance state.

Falling Edge Detection: When a change from high to low level is input to the $\overline{\text{POE}}$ pins.

Low-Level Detection: Figure 8.114 shows the low-level detection operation. Sixteen continuous low levels are sampled with the sampling clock established by the ICSR1. If even one high level is detected during this interval, the low level is not accepted.

Furthermore, the timing when the large-current pins enter the high-impedance state from the sampling clock is the same in both falling-edge detection and in low-level detection.



Note: * Other large-current pins (PE11/TIOC3D, PE12/TIOC4A, PE13/TIOC4B/$\overline{\text{MRES}}$, PE14/TIOC4C, PE15/TIOC4D/$\overline{\text{IRQOUT}}$) also go to the high-impedance state at the same timing.

Figure 8.114 Low-Level Detection Operation

Output-Level Compare Operation

Figure 8.115 shows an example of the output-level compare operation for the combination of PE9/TIOC3B and PE11/TIOC3D. The operation is the same for the other pin combinations.

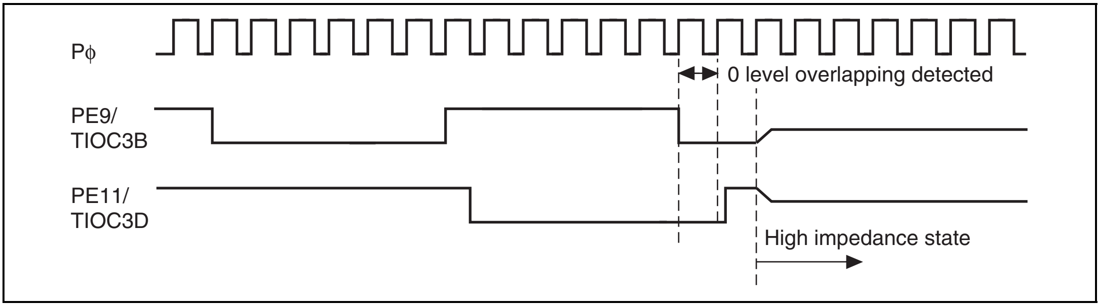


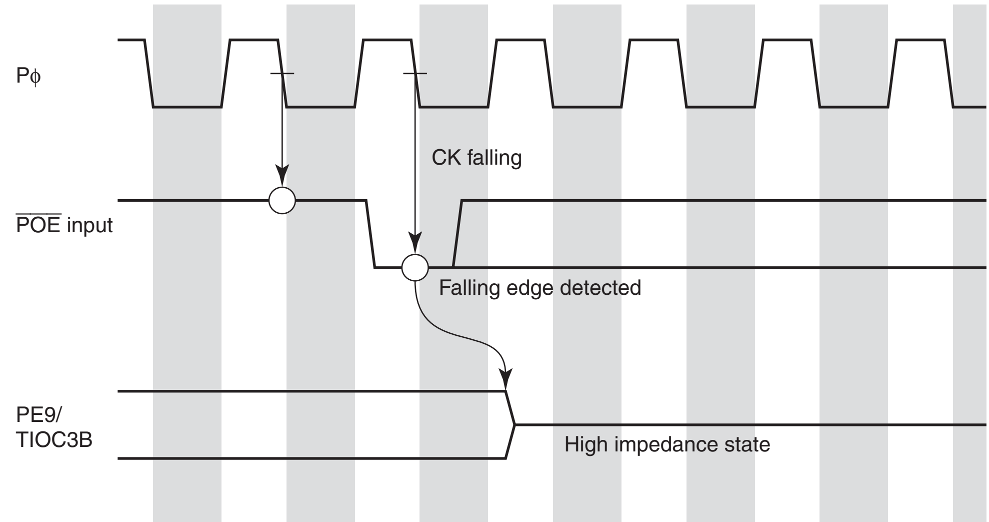
Figure 8.115 Output-Level Detection Operation

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 bit 12 to 15 (POE0F to POE3F) flags of the ICSR1. High-current pins that have become high-impedance due to output-level detection can be released either by returning them to their initial state with a power-on reset, or by first clearing bit 9 (OCE) of the OCSR to disable output-level compares, then clearing the bit 15 (OSF) flag. However, when returning from high-impedance state by clearing the OSF flag, always do so only after outputting a high level from the high-current pins (TIOC3B, TIOC3D, TIOC4A, TIOC4B, TIOC4C, and TIOC4D). High-level outputs can be achieved by setting the MTU internal registers.

POE Timing

Figure 8.116 shows an example of timing from $\overline{\text{POE}}$ input to high impedance of pin.



Note: Other large-current pins (PE11/TICO3D, PE12/TIOC4A, PE13/TIOC4B/$\overline{\text{MRES}}$, PE14/TIOC4C, PE15/TIOC4D/$\overline{\text{IRQOUT}}$) also go to the high impedance state at the same timing

Figure 8.116 Falling Edge Detection Operation

8.9.5 Usage Note

To set the POE pin as a level detection pin, a high level signal must be firstly input to the POE pin.

(1) Symptom

(a) Regarding the POEnF*1 bits

If setting of the POEnF bits in the input level control/status registers (ICSR1 and ICSR2) by the hardware*2 and reading from these bits occur simultaneously, "0" will be read, where "1" should be read.

Furthermore, if clearing of these bits is attempted subsequent to the above condition, the clearing should be ignored*3 but it will be carried out.

Notes: *1 For the SH7046-Series and SH7047-Series, n = 0 to 6; for the SH7144-Series, n = 0 to 3.
*2 The POEnF bits are set when the signals input to the respective POEn pins satisfy the conditions that are specified by the POEnM1 and POEnM0 of the ICSR1 and ICSR2.
*3 The correct operation is that clearing of the POEnF bits is only possible after "1" is read from them in order to prevent accidental clearing.

(b) Regarding the OSF bit

The same symptom applies to the OSF bits of the output level control/status register (OCSR).

(2) To Avoid This Problem

Please clear the POEnF bits or the OSF bit in these steps: first execute a read for ICSR1, ICSR2, or OCSR, then write "0" to the bits that had a read value of "1" to clear them while writing "1" to other bits. If this procedure is not followed, the POEnF bits and the OSF bit may be cleared unexpectedly if their setting by hardware and reading occur simultaneously.

Section 9 Watchdog Timer

The watchdog timer (WDT) is an 8-bit timer that can reset this LSI internally if the counter overflows without rewriting the counter value due to a system crash or the like.

When this watchdog function is not needed, the WDT can be used as an interval timer. In interval timer operation, an interval timer interrupt is generated each time the counter overflows.

The block diagram of the WDT is shown in figure 9.1.

9.1 Features

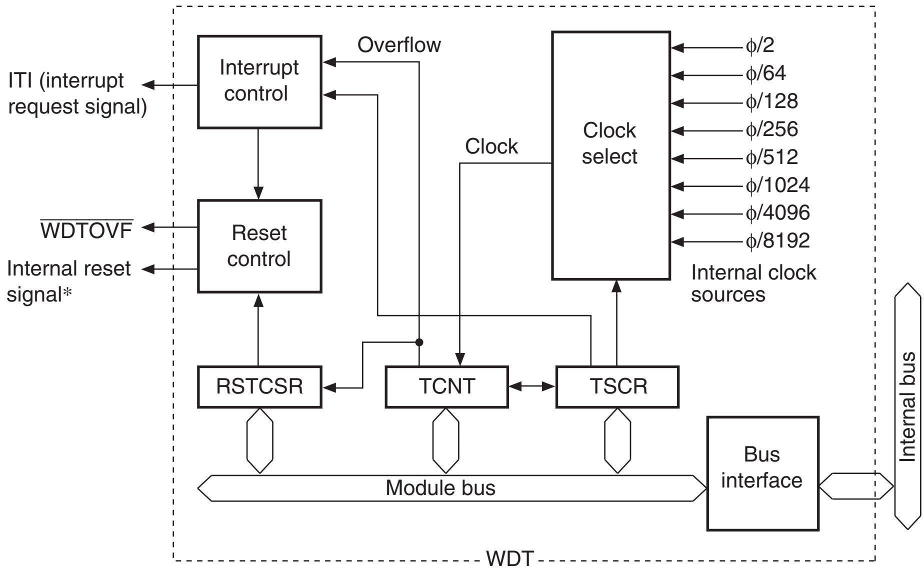
- Selectable from eight counter input clocks.
- Switchable between watchdog timer mode and interval timer mode
- Clears software standby mode

In watchdog timer mode

- Output $\overline{\text{WDTOVF}}$ signal
- If the counter overflows, it is possible to select whether this LSI is internally reset or not.

In interval timer mode

- If the counter overflows, the WDT generates an interval timer interrupt (ITI).



Legend:
TCSR: Timer control/status register
TCNT: Timer counter
RSTCSR: Reset control/status register

Note: * The internal reset signal can be generated by making a register setting. Power-on reset or manual reset can be selected.

Figure 9.1 Block Diagram of WDT

9.2 Input/Output Pin

Table 9.1 shows the pin configuration of the watchdog timer.

Table 9.1 Pin Configuration

Pin	Abbreviation	I/O	Function
Watchdog timer overflow	$\overline{\text{WDTOVF}}$	O	Outputs the counter overflow signal in watchdog timer mode

Note: The $\overline{\text{WDTOVF}}$ pin should not be pulled down. However, if it is necessary to pull this pin down, a resistance of 1 MΩ or higher should be used.

9.3 Register Descriptions

The WDT has the following three registers. For details, refer to section 18, List of Registers. To prevent accidental overwriting, TCSR, TCNT, and RSTCSR have to be written to in a method different from normal registers. For details, refer to section 9.6.1, Notes on Register Access.

- Timer control/status register (TCSR)
- Timer counter (TCNT)
- Reset control/status register (RSTCSR)

9.3.1 Timer Counter (TCNT)

TCNT is an 8-bit readable/writable upcounter. When the timer enable bit (TME) in the timer control/status register (TCSR) is set to 1, TCNT starts counting pulses of an internal clock selected by clock select bits 2 to 0 (CKS2 to CKS0) in TCSR. When the value of TCNT overflows (changes from H'FF to H'00), a watchdog timer overflow signal ($\overline{\text{WDTOVF}}$) or interval timer interrupt (ITI) is generated, depending on the mode selected in the WT/$\overline{\text{IT}}$ bit of TCSR.

9.3.2 Timer Control/Status Register (TCSR)

TCSR is an 8-bit readable/writable register. Its functions include selecting the clock source to be input to TCNT, and the timer mode.

Bit	Bit Name	Initial Value	R/W	Description
7	OVF	0	R/(W)*[1]	Overflow Flag Indicates that TCNT has overflowed in interval timer mode. Only a write of 0 is permitted, to clear the flag. This flag is not set in watchdog timer mode. [Setting condition] • When TCNT overflows in interval timer mode. [Clearing conditions] • Cleared by reading OVF • When 0 is written to the TME bit in interval timer mode
6	WT/$\overline{\text{IT}}$	0	R/W	Timer Mode Select Selects whether the WDT is used as a watchdog timer or interval timer. When TCNT overflows, the WDT either generates an interval timer interrupt (ITI) or generates a $\overline{\text{WDTOVF}}$ signal, depending on the mode selected. 0: Interval timer mode Interval timer interrupt (ITI) request to the CPU when TCNT overflows 1: Watchdog timer mode $\overline{\text{WDTOVF}}$ signal output externally when TCNT overflows*[2].
5	TME	0	R/W	Timer Enable Enables or disables the timer. 0: Timer disabled TCNT is initialized to H'00 and count-up stops 1: Timer enabled TCNT starts counting. A $\overline{\text{WDTOVF}}$ signal or interrupt is generated when TCNT overflows.
4, 3	—	All 1	R	Reserved This bit is always read as 1, and should only be written with 1.

Bit	Bit Name	Initial Value	R/W	Description
2	CKS2	0	R/W	Clock Select 2 to 0
1	CKS1	0	R/W	Select one of eight internal clock sources for input to TCNT. The clock signals are obtained by dividing the frequency of the system clock (φ). The overflow frequency for φ = 40 MHz is enclosed in parentheses*3.
0	CKS0	0	R/W	000: Clock φ/2 (frequency: 12.8 μs) 001: Clock φ/64 (frequency: 409.6 μs) 010: Clock φ/128 (frequency: 0.8 ms) 011: Clock φ/256 (frequency: 1.6 ms) 100: Clock φ/512 (frequency: 3.3 ms) 101: Clock φ/1024 (frequency: 6.6 ms) 110: Clock φ/4096 (frequency: 26.2 ms) 111: Clock φ/8192 (frequency: 52.4 ms)

Notes: 1. Only a 0 can be written after reading 1.

2. Section 9.3.3, Reset Control/Status Register (RSTCSR), describes in detail what happens when TCNT overflows in watchdog timer mode.

3. The overflow interval listed is the time from when the TCNT begins counting at H'00 until an overflow occurs.

9.3.3 Reset Control/Status Register (RSTCSR)

RSTCSR is an 8-bit readable/writable register that controls the generation of the internal reset signal when TCNT overflows, and selects the type of internal reset signal.

Bit	Bit Name	Initial Value	R/W	Description
7	WOVF	0	R/(W)*	Watchdog Overflow Flag This bit is set when TCNT overflows in watchdog timer mode. This bit cannot be set in interval timer mode. [Setting condition] • Set when TCNT overflows in watchdog timer mode [Clearing condition] • Cleared by reading WOVF, and then writing 0 to WOVF
6	RSTE	0	R/W	Reset Enable Specifies whether or not a reset signal is generated in the chip if TCNT overflows in watchdog timer mode. 0: Reset signal is not generated even if TCNT overflows (Though this LSI is not reset, TCNT and TCSR in WDT are reset) 1: Reset signal is generated if TCNT overflows
5	RSTS	0	R/W	Reset Select Selects the type of internal reset generated if TCNT overflows in watchdog timer mode. 0: Power-on reset 1: Manual reset
4 to 0	—	All 1	R	Reserved These bits are always read as 1, and should only be written with 1.

Note: * Only 0 can be written, for flag clearing.

9.4 Operation

9.4.1 Watchdog Timer Mode

To use the WDT as a watchdog timer, set the WT/$\overline{\text{IT}}$ and TME bits of TCSR to 1. Software must prevent TCNT overflow by rewriting the TCNT value (normally by writing H'00) before overflow occurs. No TCNT overflows will occur while the system is operating normally, but if TCNT fails to be rewritten and overflows occur due to a system crash or the like, a $\overline{\text{WDTOVF}}$ signal is output externally. The $\overline{\text{WDTOVF}}$ signal can be used to reset the system. The $\overline{\text{WDTOVF}}$ signal is output for 128 ϕ clock cycles.

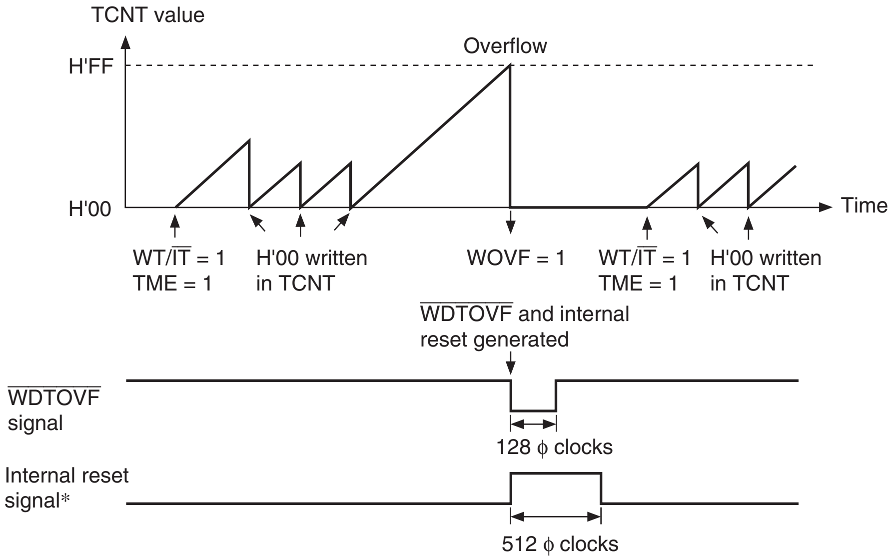
If the RSTE bit in RSTCSR is set to 1, a signal to reset the chip will be generated internally simultaneous to the $\overline{\text{WDTOVF}}$ signal when TCNT overflows. Either a power-on reset or a manual reset can be selected by the RSTS bit in RSTCSR. The internal reset signal is output for 512 ϕ clock cycles.

When a WDT overflow reset is generated simultaneously with a reset input at the $\overline{\text{RES}}$ pin, the $\overline{\text{RES}}$ reset takes priority, and the WOVF bit in RSTCSR is cleared to 0.

The following are not initialized by a WDT reset signal:

- POE (port output enable) of MTU registers
- PFC (pin function controller) registers
- I/O port registers

These registers are initialized only by an external power-on reset.



Legend:
WT/IT: Timer mode select bit
TME: Timer enable bit

Note: * Internal reset signal occurs only when the RSTE bit is set to 1.

Figure 9.2 Operation in Watchdog Timer Mode

9.4.2 Interval Timer Mode

To use the WDT as an interval timer, clear WT/$\overline{\text{IT}}$ to 0 and set TME to 1 in TCSR. An interval timer interrupt (ITI) is generated each time the timer counter (TCNT) overflows. This function can be used to generate interval timer interrupts at regular intervals.

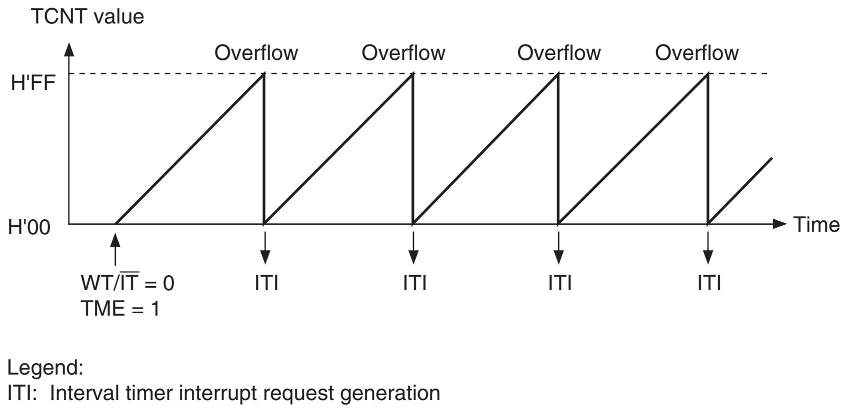


Figure 9.3 Operation in Interval Timer Mode

9.4.3 Clearing Software Standby Mode

The watchdog timer has a special function to clear software standby mode with an NMI interrupt or IRQ0 to IRQ3 interrupts. When using software standby mode, set the WDT as described below.

Before Transition to Software Standby Mode: The TME bit in TCSR must be cleared to 0 to stop the watchdog timer counter before entering software standby mode. The chip cannot enter software standby mode while the TME bit is set to 1. Set bits CKS2 to CKS0 in TCSR so that the counter overflow interval is equal to or longer than the oscillation settling time. See section 19.3, AC Characteristics, for the oscillation settling time.

Recovery from Software Standby Mode: When an NMI signal or $\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$ signals are received in software standby mode, the clock oscillator starts running and TCNT starts incrementing at the rate selected by bits CKS2 to CKS0 before software standby mode was entered. When TCNT overflows (changes from H'FF to H'00), the clock is presumed to be stable and usable; clock signals are supplied to the entire chip and software standby mode ends.

For details on software standby mode, see section 17, Power-Down Modes.

9.4.4 Timing of Setting the Overflow Flag (OVF)

In interval timer mode, when TCNT overflows, the OVF bit of TCSR is set to 1 and an interval timer interrupt (ITI) is simultaneously requested. Figure 9.4 shows this timing.

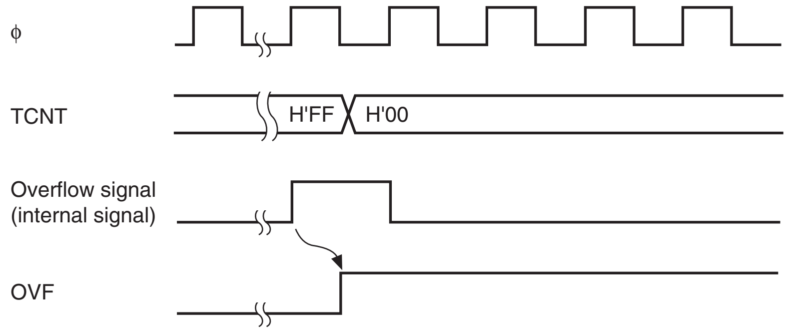


Figure 9.4 Timing of Setting OVF

9.4.5 Timing of Setting the Watchdog Timer Overflow Flag (WOVF)

When TCNT overflows in watchdog timer mode, the WOVF bit of RSTCSR is set to 1 and a $\overline{\text{WDTOVF}}$ signal is output. When the RSTE bit in RSTCSR is set to 1, TCNT overflow enables an internal reset signal to be generated for the entire chip. Figure 9.5 shows this timing.

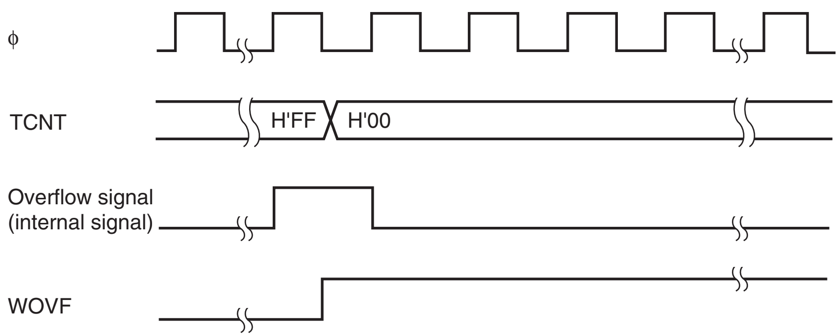


Figure 9.5 Timing of Setting WOVF

9.5 Interrupt Source

During interval timer mode operation, an overflow generates an interval timer interrupt (ITI). The interval timer interrupt is requested whenever the OVF flag is set to 1 in TCSR. OVF must be cleared to 0 in the interrupt handling routine.

Table 9.2 WDT Interrupt Source (in Interval Timer Mode)

Name	Interrupt Source	Interrupt Flag
ITI	TCNT overflow	OVF

9.6 Usage Notes

9.6.1 Notes on Register Access

The watchdog timer's TCNT, TCSR, and RSTCSR registers differ from other registers in being more difficult to write to. The procedures for writing to and reading these registers are given below.

Writing to TCNT and TCSR: These registers must be written by a word transfer instruction. They cannot be written by byte transfer instructions.

TCNT and TCSR both have the same write address. The write data must be contained in the lower byte of the written word. The upper byte must be H'5A (for TCNT) or H'A5 (for TCSR) (figure 9.6). This transfers the write data from the lower byte to TCNT or TCSR.

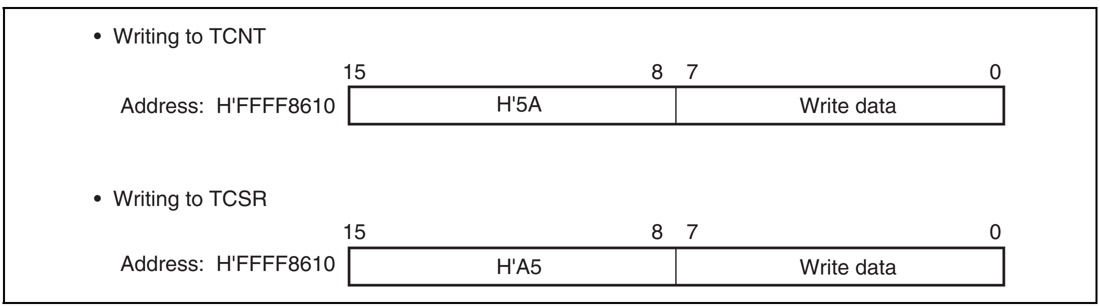


Figure 9.6 Writing to TCNT and TCSR

Writing to RSTCSR: RSTCSR must be written by a word access to address H'FFFF8612. It cannot be written by byte transfer instructions.

Procedures for writing 0 to WOVF (bit 7) and for writing to RSTE (bit 6) and RSTS (bit 5) are different, as shown in figure 9.7.

To write 0 to the WOVF bit, the write data must be H'A5 in the upper byte and H'00 in the lower byte. This clears the WOVF bit to 0. The RSTE and RSTS bits are not affected. To write to the RSTE and RSTS bits, the upper byte must be H'5A and the lower byte must be the write data. The values of bits 6 and 5 of the lower byte are transferred to the RSTE and RSTS bits, respectively. The WOVF bit is not affected.

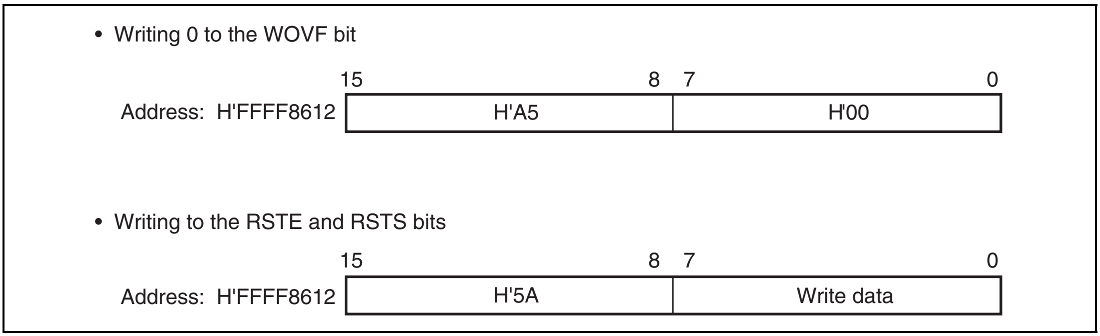


Figure 9.7 Writing to RSTCSR

Reading from TCNT, TCSR, and RSTCSR: TCNT, TCSR, and RSTCSR are read like other registers. Use byte transfer instructions. The read addresses are H'FFFF8610 for TCSR, H'FFFF8611 for TCNT, and H'FFFF8613 for RSTCSR.

9.6.2 TCNT Write and Increment Contention

If a timer counter increment clock pulse is generated during the T3 state of a write cycle to TCNT, the write takes priority and the timer counter is not incremented. Figure 9.8 shows this operation.

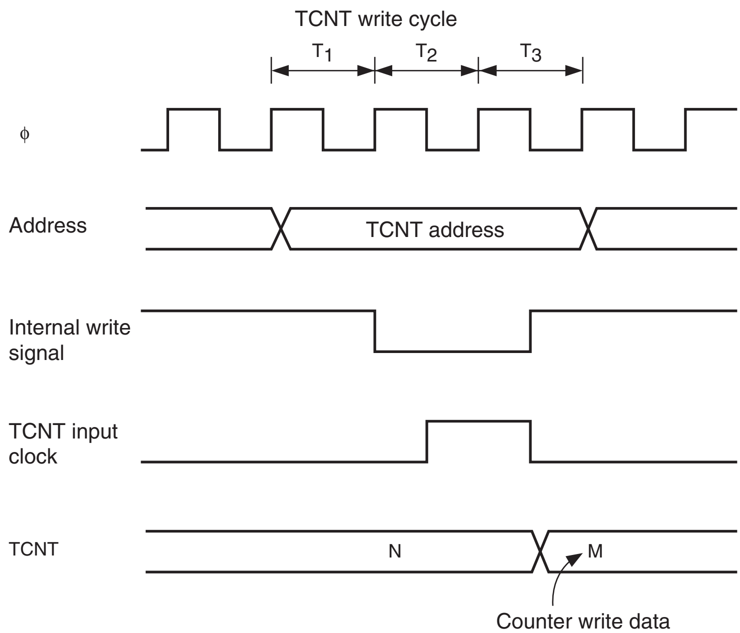


Figure 9.8 Contention between TCNT Write and Increment

9.6.3 Changing CKS2 to CKS0 Bit Values

If the values of bits CKS2 to CKS0 in the timer control/status register (TCSR) are rewritten while the WDT is running, the count may not increment correctly. Always stop the watchdog timer (by clearing the TME bit to 0) before rewriting the values of bits CKS2 to CKS0.

9.6.4 Changing between Watchdog Timer/Interval Timer Modes

To prevent incorrect operation, always stop the watchdog timer (by clearing the TME bit to 0) before switching between interval timer mode and watchdog timer mode.

9.6.5 System Reset by $\overline{\text{WDTOVF}}$ Signal

If a $\overline{\text{WDTOVF}}$ output signal is input to the $\overline{\text{RES}}$ pin, the chip cannot initialize correctly.

Avoid logical input of the $\overline{\text{WDTOVF}}$ signal to the $\overline{\text{RES}}$ input pin. To reset the entire system with the $\overline{\text{WDTOVF}}$ signal, use the circuit shown in figure 9.9.

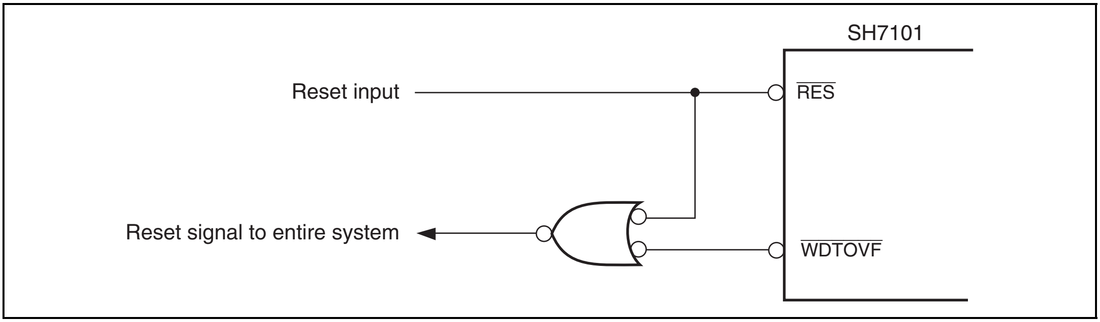


Figure 9.9 Example of System Reset Circuit Using $\overline{\text{WDTOVF}}$ Signal

9.6.6 Internal Reset in Watchdog Timer Mode

If the RSTE bit is cleared to 0 in watchdog timer mode, the chip will not be reset internally when a TCNT overflow occurs, but TCNT and TCSR in the WDT will be reset.

9.6.7 Manual Reset in Watchdog Timer Mode

When an internal reset is effected by TCNT overflow in watchdog timer mode, the processor waits until the end of the bus cycle at the time of manual reset generation before making the transition to manual reset exception processing.

9.6.8 Notes on Using $\overline{\text{WDTOVF}}$ pin

The $\overline{\text{WDTOVF}}$ pin should not be pulled down. However, if it is necessary to pull this pin down, a resistance of 1 MΩ or higher should be used.

Section 10 Serial Communication Interface (SCI)

This LSI has two independent serial communication interface (SCI) channels. The SCI can handle both asynchronous and clocked 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).

10.1 Features

- Choice of asynchronous or clocked synchronous serial communication mode
- Full-duplex communication capability

  The transmitter and receiver are mutually independent, enabling transmission and reception to be executed simultaneously.

  Double-buffering is used in both the transmitter and the receiver, enabling continuous transmission and continuous reception of serial data.
- On-chip baud rate generator allows any bit rate to be selected

  External clock can be selected as a transfer clock source.
- Choice of LSB-first or MSB-first transfer* (except in the case of asynchronous mode 7-bit data)
- Four interrupt sources

  Four interrupt sources — transmit-end, transmit-data-empty, receive-data-full, and receive error — that can issue requests.
- Module standby mode can be set

Asynchronous mode

- Data length: 7 or 8 bits
- Stop bit length: 1 or 2 bits
- Parity: Even, odd, or none
- Multiprocessor bit: 1 or 0
- Receive error detection: Parity, overrun, and framing errors
- Break detection: Break can be detected by reading the RxD pin level directly in case of a framing error

Clocked Synchronous mode

- Data length: 8 bits
- Receive error detection: Overrun errors detected

Note: * The description in this section are based on LSB-first transfer.

Figure 10.1 shows a block diagram of the SCI.

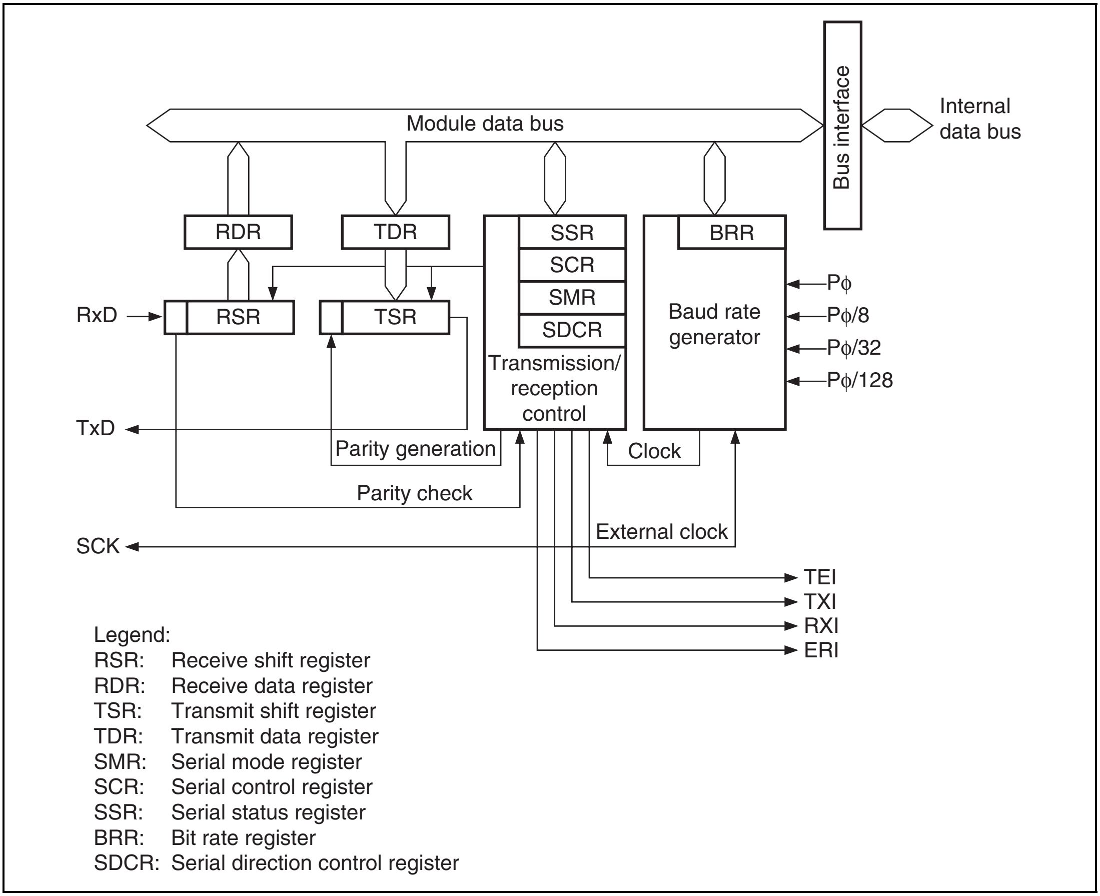


Figure 10.1 Block Diagram of SCI

10.2 Input/Output Pins

Table 10.1 shows the SCI pin configuration.

Table 10.1 Pin Configuration

Channel	Pin Name*	I/O	Function
2	SCK2	I/O	SCI2 clock input/output
	RxD2	Input	SCI2 receive data input
	TxD2	Output	SCI2 transmit data output
3	SCK3	I/O	SCI3 clock input/output
	RxD3	Input	SCI3 receive data input
	TxD3	Output	SCI3 transmit data output

Note: * Pin names SCK, RxD, and TxD are used in the text for all channels, omitting the channel designation.

10.3 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 18, List of Registers.

Channel 2

- Serial Mode Register_2 (SMR_2)
- Bit Rate Register_2 (BRR_2)
- Serial Control Register_2 (SCR_2)
- Transmit Data Register_2 (TDR_2)
- Serial Status Register_2 (SSR_2)
- Receive Data Register_2 (RDR_2)
- Serial Direction Control Register_2 (SDCR_2)

Channel 3

- Serial Mode Register_3 (SMR_3)
- Bit Rate Register_3 (BRR_3)
- Serial Control Register_3 (SCR_3)
- Transmit Data Register_3 (TDR_3)
- Serial Status Register_3 (SSR_3)

- Receive Data Register_3 (RDR_3)
- Serial Direction Control Register_3 (SDCR_3)

10.3.1 Receive Shift Register (RSR)

RSR is a shift register used to receive serial data that is input to the RxD pin and convert it into parallel data. When one byte of data has been received, it is transferred to RDR automatically. RSR cannot be directly read or written to by the CPU.

10.3.2 Receive Data Register (RDR)

RDR is an 8-bit register that stores receive data. When the SCI has received one byte of serial data, it transfers the received serial data from RSR to RDR where it is stored. After this, RSR is receive-enabled. Since RSR and RDR function as a double buffer in this way, enables continuous receive operations to be performed. After confirming that the RDRF bit in SSR is set to 1, read RDR for only once. RDR cannot be written to by the CPU. The initial value of RDR is H'00.

10.3.3 Transmit Shift Register (TSR)

TSR is a shift register that transmits serial data. To perform serial data transmission, the SCI first transfers transmit data from TDR to TSR, then sends the data to the TxD pin. TSR cannot be directly accessed by the CPU.

10.3.4 Transmit Data Register (TDR)

TDR is an 8-bit register that stores transmit data. When the SCI detects that TSR is empty, it transfers the transmit data written in TDR to TSR and starts transmission. The double-buffered structures of TDR and TSR enables continuous serial transmission. If the next transmit data has already been written to TDR during serial transmission, the SCI transfers the written data to TSR to continue transmission. Although TDR can be read or written to by the CPU at all times, to achieve reliable serial transmission, write transmit data to TDR for only once after confirming that the TDRE bit in SSR is set to 1. The initial value of TDR is H'FF.

10.3.5 Serial Mode Register (SMR)

SMR is used to set the SCI's serial transfer format and select the baud rate generator clock source.

Bit	Bit Name	Initial Value	R/W	Description
7	C/$\overline{A}$	0	R/W	Communication Mode 0: Asynchronous mode 1: Clocked synchronous mode
6	CHR	0	R/W	Character Length (enabled only in asynchronous mode) 0: Selects 8 bits as the data length. 1: Selects 7 bits as the data length. LSB-first is fixed and the MSB (bit 7) of TDR is not transmitted in transmission. In clocked synchronous mode, a fixed data length of 8 bits is used.
5	PE	0	R/W	Parity Enable (enabled only in asynchronous mode) When this bit is set to 1, the parity bit is added to transmit data before transmission, and the parity bit is checked in reception. For a multiprocessor format, parity bit addition and checking are not performed regardless of the PE bit setting.
4	O/$\overline{E}$	0	R/W	Parity Mode (enabled only when the PE bit is 1 in asynchronous mode) 0: Selects even parity. 1: Selects odd parity.
3	STOP	0	R/W	Stop Bit Length (enabled only in asynchronous mode) Selects the stop bit length in transmission. 0: 1 stop bit 1: 2 stop bits In reception, only the first stop bit is checked. If the second stop bit is 0, it is treated as the start bit of the next transmit character.
2	MP	0	R/W	Multiprocessor Mode (enabled only in asynchronous mode) When this bit is set to 1, the multiprocessor communication function is enabled. The PE bit and O/$\overline{E}$ bit settings are invalid in multiprocessor mode.

Bit	Bit Name	Initial Value	R/W	Description
1	CKS1	0	R/W	Clock Select 1 and 0
0	CKS0	0	R/W	These bits select the clock source for the baud rate generator. 00: Pϕ clock (n = 0) 01: Pϕ/8 clock (n = 1) 10: Pϕ/32 clock (n = 2) 11: Pϕ/128 clock (n = 3) For the relation between the bit rate register setting and the baud rate, see section 10.3.9, Bit Rate Register (BRR). n is the decimal display of the value of n in BRR (see section 10.3.9, Bit Rate Register (BRR)).

10.3.6 Serial Control Register (SCR)

SCR is a register that performs enabling or disabling of SCI transfer operations and interrupt requests, and selection of the transfer clock source. For details on interrupt requests, refer to section 10.7, Interrupts Sources.

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable When this bit is set to 1, TXI interrupt request is enabled.
6	RIE	0	R/W	Receive Interrupt Enable When this bit is set to 1, RXI and ERI interrupt requests are enabled.
5	TE	0	R/W	Transmit Enable When this bit is set to 1, transmission is enabled.
4	RE	0	R/W	Receive Enable When this bit is set to 1, reception is enabled.
3	MPIE	0	R/W	Multiprocessor Interrupt Enable (enabled only when the MP bit in SMR is 1 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 SSR is prohibited. On receiving data in which the multiprocessor bit is 1, this bit is automatically cleared and normal reception is resumed. For details, refer to section 10.5, Multiprocessor Communication Function.
2	TEIE	0	R/W	Transmit End Interrupt Enable This bit is set to 1, TEI interrupt request is enabled.

Bit	Bit Name	Initial Value	R/W	Description
1	CKE1	0	R/W	Clock Enable 1 and 0
0	CKE0	0	R/W	Selects the clock source and SCK pin function. Asynchronous mode: 00: Internal clock, SCK pin used for input pin (input signal is ignored) or output pin (output level is undefined) 01: Internal clock, SCK pin used for clock output (The output clock frequency is the same as the bit rate) 10: External clock, SCK pin used for clock input (The input clock frequency is 16 times the bit rate) 11: External clock, SCK pin used for clock input (The input clock frequency is 16 times the bit rate) Clocked 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

10.3.7 Serial Status Register (SSR)

SSR is a register containing status flags of the SCI and multiprocessor bits for transfer. 1 cannot be written to flags TDRE, RDRF, ORER, PER, and FER; they can only be cleared.

Bit	Bit Name	Initial Value	R/W	Description
7	TDRE	1	R/(W)*	Transmit Data Register Empty Displays whether TDR contains transmit data. [Setting conditions] • Power-on reset or software standby mode • When the TE bit in SCR is 0 • When data is transferred from TDR to TSR and data can be written to TDR [Clearing condition] • When 0 is written to TDRE after reading TDRE = 1
6	RDRF	0	R/(W)*	Receive Data Register Full Indicates that the received data is stored in RDR. [Setting condition] • When serial reception ends normally and receive data is transferred from RSR to RDR [Clearing conditions] • Power-on reset or software standby mode • When 0 is written to RDRF after reading RDRF = 1 The RDRF flag is not affected and retains their previous values when the RE bit in SCR is cleared to 0.
5	ORER	0	R/(W)*	Overrun Error [Setting condition] • When the next serial reception is completed while RDRF = 1 [Clearing conditions] • Power-on reset or software standby mode • When 0 is written to ORER after reading ORER = 1 The ORER flag is not affected and retains their previous values when the RE bit in SCR is cleared to 0.

Bit	Bit Name	Initial Value	R/W	Description
4	FER	0	R/(W)*	Framing Error [Setting condition] • When the stop bit is 0 [Clearing conditions] • Power-on reset or software standby mode • When 0 is written to FER after reading FER = 1 In 2-stop-bit mode, only the first stop bit is checked. The FER flag is not affected and retains their previous values when the RE bit in SCR is cleared to 0.
3	PER	0	R/(W)*	Parity Error [Setting condition] • When a parity error is detected during reception [Clearing conditions] • Power-on reset or software standby mode • When 0 is written to PER after reading PER = 1 The PER flag is not affected and retains their previous values when the RE bit in SCR is cleared to 0.
2	TEND	1	R	Transmit End [Setting conditions] • Power-on reset or software standby mode • When the TE bit in SCR is 0 • When TDRE = 1 at transmission of the last bit of a 1-byte serial transmit character [Clearing condition] • When 0 is written to TDRE after reading TDRE = 1
1	MPB	0	R	Multiprocessor Bit MPB stores the multiprocessor bit in the receive data. When the RE bit in SCR is cleared to 0 its previous state is retained.
0	MPBT	0	R/W	Multiprocessor Bit Transfer MPBT sets the multiprocessor bit value to be added to the transmit data.

Note: * Only 0 can be written, for flag clearing.

10.3.8 Serial Direction Control Register (SDCR)

The DIR bit in the serial direction control register (SDCR) 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. With a 7-bit data length, LSB-first transfer must be selected. The description in this section assumes LSB-first transfer.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 1	R	Reserved The write value must always be 1. Operation cannot be guaranteed if 0 is written.
3	DIR	0	R/W	Data Transfer Direction Selects the serial/parallel conversion format. Valid for an 8-bit transmit/receive format. 0: TDR contents are transmitted in LSB-first order Receive data is stored in RDR in LSB-first 1: TDR contents are transmitted in MSB-first order Receive data is stored in RDR in MSB-first
2	—	0	R	Reserved The write value must always be 0. Operation cannot be guaranteed if 1 is written.
1	—	1	R	Reserved This bit is always read as 1, and cannot be modified.
0	—	0	R	Reserved The write value must always be 0. Operation cannot be guaranteed if 1 is written.

10.3.9 Bit Rate Register (BRR)

BRR is an 8-bit register that adjusts the bit rate. As the SCI performs baud rate generator control independently for each channel, different bit rates can be set for each channel. Table 10.2 shows the relationships between the N setting in BRR and the effective bit rate B_0 for asynchronous and clocked synchronous modes. The initial value of BRR is H'FF, and it can be read or written to by the CPU at all times.

Table 10.2 Relationships between N Setting in BRR and Effective Bit Rate B_0

Mode	Bit Rate	Error
Asynchronous mode (n = 0)	$B_0 = \dfrac{P\phi \times 10^6}{32 \times 2^{2n} \times (N + 1)}$	$\text{Error (\%)} = \left(\dfrac{B_0}{B_1} - 1\right) \times 100$
Asynchronous mode (n = 1 to 3)	$B_0 = \dfrac{P\phi \times 10^6}{32 \times 2^{2n+1} \times (N + 1)}$	$\text{Error (\%)} = \left(\dfrac{B_0}{B_1} - 1\right) \times 100$
Clocked synchronous mode (n = 0)	$B_0 = \dfrac{P\phi \times 10^6}{4 \times 2^{2n} \times (N + 1)}$	—
Clocked synchronous mode (n = 1 to 3)	$B_0 = \dfrac{P\phi \times 10^6}{4 \times 2^{2n+1} \times (N + 1)}$	—

Notes: B_0: Effective bit rate (bit/s) Actual transfer speed according to the register settings
B_1: Logical bit rate (bit/s) Specified transfer speed of the target system
N: BRR setting for baud rate generator ($0 \leq N \leq 255$)
Pϕ: Peripheral clock operating frequency (MHz)
n : Determined by the SMR settings shown in the following tables.

SMR Setting		
CKS1	**CKS0**	**n**
0	0	0
0	1	1
1	0	2
1	1	3

Table 10.3 shows sample N settings in BRR in normal asynchronous mode. Table 10.4 shows the maximum bit rate for each frequency in normal asynchronous mode. Table 10.6 shows sample N settings in BRR in clocked synchronous mode. For details, refer to section 10.4.2, Receive Data Sampling Timing and Reception Margin in Asynchronous Mode. Tables 10.5 and 10.7 show the maximum bit rates with external clock input.

Table 10.3 BRR Settings for Various Bit Rates (Asynchronous Mode)

Logical Bit Rate (bit/s)	Operating Frequency Pϕ (MHz) 4			6			8			10			12		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	1	140	0.74	1	212	0.03	2	70	0.03	2	88	–0.25	2	106	–0.44
150	1	103	0.16	1	155	0.16	2	51	0.16	2	64	0.16	2	77	0.16
300	1	51	0.16	1	77	0.16	2	25	0.16	1	129	0.16	2	38	0.16
600	1	25	0.16	1	38	0.16	2	12	0.16	1	64	0.16	1	77	0.16
1200	1	12	0.16	0	155	0.16	1	25	0.16	1	32	–1.36	1	38	0.16
2400	0	51	0.16	0	77	0.16	1	12	0.16	0	129	0.16	0	155	0.16
4800	0	25	0.16	0	38	0.16	0	51	0.16	0	64	0.16	0	77	0.16
9600	0	12	0.16	0	19	–2.34	0	25	0.16	0	32	–1.36	0	38	0.16
14400	0	8	–3.55	0	12	0.16	0	16	2.12	0	21	–1.36	0	25	0.16
19200	0	6	–6.99	0	9	–2.34	0	12	0.16	0	15	1.73	0	19	–2.34
28800	0	3	8.51	0	6	–6.99	0	8	–3.55	0	10	–1.36	0	12	0.16
31250	0	3	0.00	0	5	0.00	0	7	0.00	0	9	0.00	0	11	0.00
38400	0	2	8.51	0	4	–2.34	0	6	–6.99	0	7	1.73	0	9	–2.34

Logical Bit Rate (bit/s)	Operating Frequency Pϕ (MHz) 14			16			18			20			22		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	123	0.23	2	141	0.03	2	159	–0.12	2	177	–0.25	2	194	0.16
150	2	90	0.16	2	103	0.16	2	116	0.16	2	129	0.16	2	142	0.16
300	2	45	–0.93	2	51	0.16	2	58	–0.69	2	64	0.16	2	71	–0.54
600	2	22	–0.93	1	103	0.16	1	116	0.16	1	129	0.16	1	142	0.16
1200	1	45	–0.93	1	51	0.16	1	58	–0.69	1	64	0.16	1	71	–0.54
2400	1	22	–0.93	0	207	0.16	0	233	0.16	1	32	–1.36	1	35	–0.54
4800	0	90	0.16	0	103	0.16	0	116	0.16	0	129	0.16	0	142	0.16
9600	0	45	–0.93	0	51	0.16	0	58	–0.69	0	64	0.16	0	71	–0.54
14400	0	29	1.27	0	34	–0.79	0	38	0.16	0	42	0.94	0	47	–0.54
19200	0	22	–0.93	0	25	0.16	0	28	1.02	0	32	–1.36	0	35	–0.54
28800	0	14	1.27	0	16	2.12	0	19	–2.34	0	21	–1.36	0	23	–0.54
31250	0	13	0.00	0	15	0.00	0	17	0.00	0	19	0.00	0	21	0.00
38400	0	10	3.57	0	12	0.16	0	14	–2.34	0	15	1.73	0	17	–0.54

Logical Bit Rate (bit/s)	Operating Frequency Pφ (MHz) 24			25			26			28			30		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	212	0.03	2	221	–0.02	2	230	–0.08	2	248	–0.17	3	66	–0.62
150	2	155	0.16	2	162	–0.15	2	168	0.16	2	181	0.16	2	194	0.16
300	2	77	0.16	2	80	0.47	2	84	–0.43	2	90	0.16	2	97	–0.35
600	1	155	0.16	1	162	–0.15	1	168	0.16	1	181	0.16	2	48	–0.35
1200	1	77	0.16	1	80	0.47	1	84	–0.43	1	90	0.16	1	97	–0.35
2400	1	38	0.16	1	40	–0.76	1	41	0.76	1	45	–0.93	1	48	–0.35
4800	0	155	0.16	0	162	–0.15	0	168	0.16	0	181	0.16	0	194	0.16
9600	0	77	0.16	0	80	0.47	0	84	–0.43	0	90	0.16	0	97	–0.35
14400	0	51	0.16	0	53	0.47	0	55	0.76	0	60	–0.39	0	64	0.16
19200	0	38	0.16	0	40	–0.76	0	41	0.76	0	45	–0.93	0	48	–0.35
28800	0	25	0.16	0	26	0.47	0	27	0.76	0	29	1.27	0	32	–1.36
31250	0	23	0.00	0	24	0.00	0	25	0.00	0	27	0.00	0	29	0.00
38400	0	19	–2.34	0	19	1.73	0	20	0.76	0	22	–0.93	0	23	1.73

Logical Bit Rate (bit/s)	Operating Frequency Pφ (MHz) 32			34			36			38			40		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	3	70	0.03	3	74	0.62	3	79	–0.12	3	83	0.40	3	88	–0.25
150	2	207	0.16	2	220	0.16	2	233	0.16	2	246	0.16	3	64	0.16
300	2	103	0.16	2	110	–0.29	2	116	0.16	2	123	–0.24	2	129	0.16
600	2	51	0.16	2	54	0.62	2	58	–0.69	2	61	–0.24	2	64	0.16
1200	1	103	0.16	1	110	–0.29	1	116	0.16	1	123	–0.24	1	129	0.16
2400	1	51	0.16	1	51	6.42	1	58	–0.69	1	61	–0.24	1	64	0.16
4800	0	207	0.16	0	220	0.16	0	234	–0.27	0	246	0.16	1	32	–1.36
9600	0	103	0.16	0	110	–0.29	0	116	0.16	0	123	–0.24	0	129	0.16
14400	0	68	0.64	0	73	–0.29	0	77	0.16	0	81	0.57	0	86	–0.22
19200	0	51	0.16	0	54	0.62	0	58	–0.69	0	61	–0.24	0	64	0.16
28800	0	34	–0.79	0	36	–0.29	0	38	0.16	0	40	0.57	0	42	0.94
31250	0	31	0.00	0	33	0.00	0	35	0.00	0	37	0.00	0	39	0.00
38400	0	25	0.16	0	27	–1.18	0	28	1.02	0	30	–0.24	0	32	–1.36

Table 10.4 Maximum Bit Rate for Each Frequency when Using Baud Rate Generator (Asynchronous Mode)

Pφ (MHz)	n	N	Maximum Bit Rate (bit/s)
4	0	0	125000
8	0	0	250000
10	0	0	312500
12	0	0	375000
14	0	0	437500
16	0	0	500000
18	0	0	562500
20	0	0	625000
22	0	0	687500
24	0	0	750000
25	0	0	781250
26	0	0	812500
28	0	0	875000
30	0	0	937500
32	0	0	1000000
34	0	0	1062500
36	0	0	1125000
38	0	0	1187500
40	0	0	1250000

Table 10.5 Maximum Bit Rate with External Clock Input (Asynchronous Mode)

Pφ (MHz)	External Clock (MHz)	Maximum Bit Rate (bit/s)
4	1.0000	62500
6	1.5000	93750
8	2.0000	125000
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
25	6.2500	390625
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 10.6 BRR Settings for Various Bit Rates (Clocked Synchronous Mode)

Logical Bit Rate (bit/s)	Operating Frequency Pϕ (MHz) 4		6		8		10		12	
	n	N	n	N	n	N	n	N	n	N
250	2	124	2	187	2	249	—	—	—	—
500	1	249	—	—	2	124	2	155	2	187
1000	1	124	1	187	1	249	—	—	—	—
2500	1	49	1	74	1	99	1	124	1	149
5000	1	24	—	—	1	49	—	—	1	74
10000	0	99	0	149	1	24	0	249	—	—
25000	0	39	0	59	1	9	0	99	1	14
50000	0	19	0	29	1	4	0	49	0	59
100000	0	9	0	14	0	19	0	24	0	29
250000	0	3	0	5	0	7	0	9	0	11
500000	0	1	0	2	0	3	0	4	0	5
1000000	0	0	—	—	0	1	—	—	0	2
2500000	—	—	—	—	—	—	0	0	—	—
5000000	—	—	—	—	—	—	—	—	—	—

Logical Bit Rate (bit/s)	Operating Frequency Pϕ (MHz) 14		16		18		20		22	
	n	N	n	N	n	N	n	N	n	N
250	3	108	3	124	3	140	3	155	3	171
500	2	218	2	249	—	—	—	—	—	—
1000	2	108	2	124	2	140	2	155	3	42
2500	1	174	2	49	1	224	1	249	—	—
5000	—	—	2	24	1	112	1	124	1	137
10000	—	—	1	49	—	—	—	—	—	—
25000	0	139	1	19	0	179	1	24	0	219
50000	0	69	1	9	0	89	0	99	0	109
100000	0	34	1	4	0	44	0	49	0	54
250000	0	13	1	1	0	17	0	19	0	21
500000	0	6	1	0	0	8	0	9	0	10
1000000	—	—	0	3	—	—	0	4	—	—
2500000	—	—	—	—	—	—	0	1	—	—
5000000	—	—	—	—	—	—	0	0	—	—

Logical Bit Rate (bit/s)	Operating Frequency Pϕ (MHz) 24		25		26		28		30	
	n	N	n	N	n	N	n	N	n	N
250	3	187	3	194	3	202	3	218	3	233
500	—	—	—	—	3	101	3	108	3	116
1000	2	187	2	194	2	202	2	218	2	233
2500	2	74	—	—	—	—	—	—	—	—
5000	1	149	1	155	1	162	1	174	1	187
10000	1	74	—	—	—	—	—	—	—	—
25000	1	29	0	249	—	—	1	34	—	—
50000	1	14	0	124	0	129	0	139	0	149
100000	0	59	—	—	0	64	0	69	0	74
250000	0	23	0	24	0	25	0	27	0	29
500000	0	11	—	—	0	12	0	13	0	14
1000000	0	5	—	—	—	—	0	6	—	—
2500000	—	—	—	—	—	—	—	—	0	2
5000000	—	—	—	—	—	—	—	—	—	—

Logical Bit Rate (bit/s)	Operating Frequency Pϕ (MHz) 32		34		36		38		40	
	n	N	n	N	n	N	n	N	n	N
250	3	249	—	—	—	—	—	—	—	—
500	3	124	3	132	3	140	3	147	3	155
1000	2	249	—	—	—	—	—	—	—	—
2500	2	99	2	105	2	112	2	118	2	124
5000	2	49	1	212	1	224	1	237	1	249
10000	2	24	1	105	1	112	1	118	1	124
25000	2	9	—	—	1	44	—	—	1	49
50000	2	4	0	169	0	179	0	189	1	24
100000	1	9	0	84	0	89	0	94	0	99
250000	1	3	0	33	0	35	0	37	0	39
500000	1	1	0	16	0	17	0	18	0	19
1000000	1	0	—	—	0	8	—	—	0	9
2500000	—	—	—	—	—	—	—	—	0	3
5000000	—	—	—	—	—	—	—	—	0	1

Table 10.7 Maximum Bit Rate with External Clock Input (Clocked Synchronous Mode)

Pφ (MHz)	External Clock (MHz)	Maximum Bit Rate (bit/s)
4	0.6667	666666.7
6	1.0000	1000000.0
8	1.3333	1333333.3
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
25	4.1667	4166666.7
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

Legend:

— : Can be set, but there will be a degree of error.

* : Continuous transfer is not possible.

Note: Settings with an error of 1% or less are recommended.

10.4 Operation in Asynchronous Mode

Figure 10.2 shows the general format for asynchronous serial communication. One frame consists of a start bit (low level), followed by data, a parity bit, and finally stop bits (high level). In asynchronous serial communication, the transmission line is usually held in the mark state (high level). The SCI monitors the communication line, and when it goes to the space state (low level), recognizes a start bit and starts serial communication. Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communication. Both the transmitter and the receiver also have a double-buffered structure, so that data can be read or written during transmission or reception, enabling continuous data transfer.

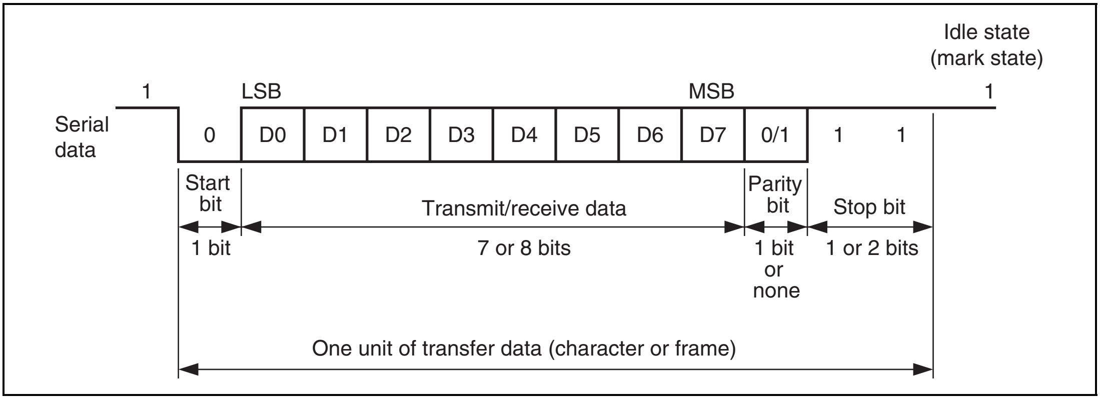


Figure 10.2 Data Format in Asynchronous Communication (Example with 8-Bit Data, Parity, Two Stop Bits)

10.4.1 Data Transfer Format

Table 10.8 shows the data transfer formats that can be used in asynchronous mode. Any of 12 transfer formats can be selected according to the SMR setting. For details on the multiprocessor bit, refer to section 10.5, Multiprocessor Communication Function.

Table 10.8 Serial Transfer Formats (Asynchronous Mode)

SMR 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

10.4.2 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCI operates on a basic clock with a frequency of 16 times the bit rate. In reception, the SCI samples the falling edge of the start bit using the basic clock, and performs internal synchronization. Receive data is latched internally at the rising edge of the 8th pulse of the basic clock as shown in figure 10.3. Thus the reception margin in asynchronous mode is given by formula (1) below.

$$M = \left\{ \left(0.5 - \frac{1}{2N}\right) - \frac{(D - 0.5)}{N} - (L - 0.5)\,F \right\} \times 100\% \quad \text{.......................... Formula (1)}$$

Where M: Reception margin (%)
N: Ratio of bit rate to clock (N = 16)
D: Clock duty cycle (D = 0 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute value of clock rate deviation

Assuming values of F = 0 and D = 0.5 in formula (1), a reception margin is given by formula below.

$$M = \{0.5 - 1/(2 \times 16)\} \times 100\ [\%] = 46.875\%$$

However, this is only the computed value, and a margin of 20% to 30% should be allowed in system design.

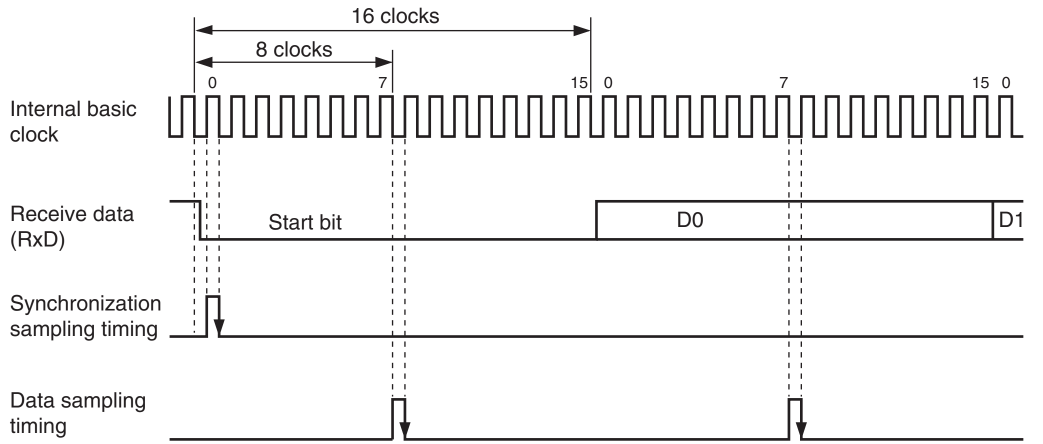


Figure 10.3 Receive Data Sampling Timing in Asynchronous Mode

10.4.3 Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock input at the SCK pin can be selected as the SCI's serial clock, according to the setting of the C/$\overline{A}$ bit in SMR and the CKE1 and CKE0 bits in SCR. When an external clock is input at the SCK pin, the clock frequency should be 16 times the bit rate used.

When the SCI is operated on an internal clock, the clock can be output from the SCK pin. The frequency of the clock output in this case is equal to the bit rate, and the phase is such that the rising edge of the clock is in the middle of the transmit data, as shown in figure 10.4.

The clock must not be stopped during operation.

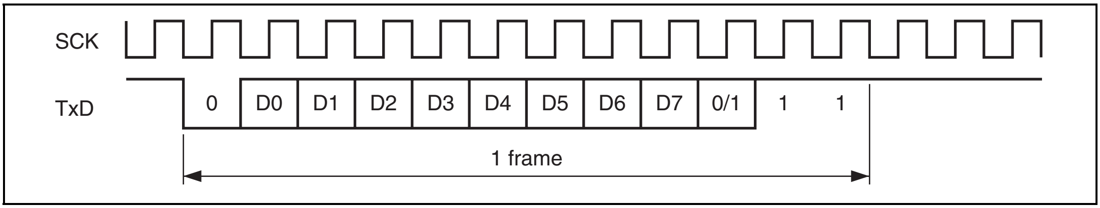
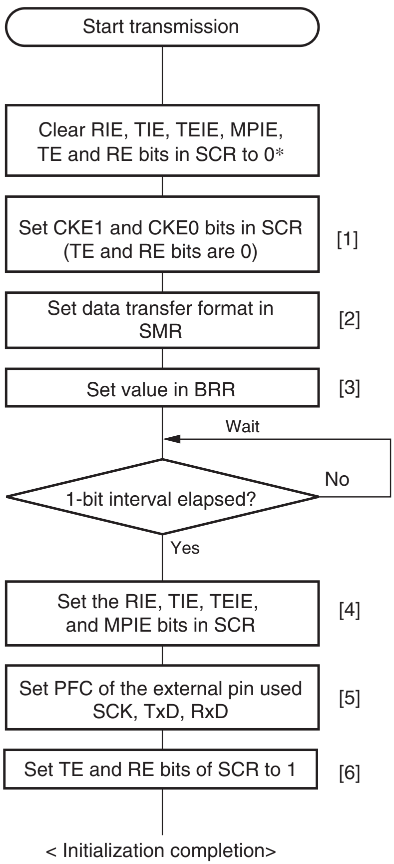


Figure 10.4 Relation between Output Clock and Transmit Data Phase (Asynchronous Mode)

10.4.4 SCI Initialization (Asynchronous Mode)

Before transmitting and receiving data, you should first clear the TE and RE bits in SCR to 0, then initialize the SCI as described below. When the operating mode, transfer format, etc., is changed, the TE and RE bits must be cleared to 0 before making the change using the following procedure. When the TE bit is cleared to 0, the TDRE flag is set to 1. Note that clearing the RE bit to 0 does not initialize the contents of the RDRF, PER, FER, and ORER flags, or the contents of RDR. When the external clock is used in asynchronous mode, the clock must be supplied even during initialization.



[1] Set the clock selection in SCR.
[2] Set the data transfer format in SMR and SCMR.
[3] Write a value corresponding to the bit rate to BRR. Not necessary if an external colck is used.
[4] Set the RIE, TIE, TEIE, and MPIE bits.
[5] Set PFC of the external pin used. Set RxD input during receiving and TxD output during transmitting. Set SCK input/output according to contents set by CKE1 and CKE0. When CKE1 and CKE0 are 0 in asynchronous mode, setting the SCK pin is unnecessary. Outputting clocks from the SCK pin starts at synchronous clock output setting.
[6] Wait at least one bit interval, then set the TE bit or RE bit in SCR to 1.* At this time, the TxD, RxD, and SCK pins can 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 10.5 Sample SCI Initialization Flowchart

10.4.5 Data Transmission (Asynchronous Mode)

Figure 10.6 shows an example of the operation for transmission in asynchronous mode. In transmission, the SCI operates as described below.

1. The SCI monitors the TDRE flag in SSR, and if is cleared to 0, recognizes that data has been written to TDR, and transfers the data from TDR to TSR.
2. After transferring data from TDR to TSR, the SCI sets the TDRE flag to 1 and starts transmission. If the TIE bit is set to 1 at this time, a transmit data empty interrupt request (TXI) is generated. Because the TXI interrupt routine writes the next transmit data to TDR before transmission of the current transmit data has finished, continuous transmission can be enabled.
3. Data is sent from the TxD pin in the following order: start bit, transmit data, parity bit or multiprocessor bit (may be omitted depending on the format), and stop bit.
4. The SCI checks the TDRE flag at the timing for sending the stop bit.
5. If the TDRE flag is 0, the data is transferred from TDR to TSR, the stop bit is sent, and then serial transmission of the next frame is started.
6. If the TDRE flag is 1, the TEND flag in SSR 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 SCR is set to 1 at this time, a TEI interrupt request is generated.

Figure 10.7 shows a sample flowchart for transmission in asynchronous mode.

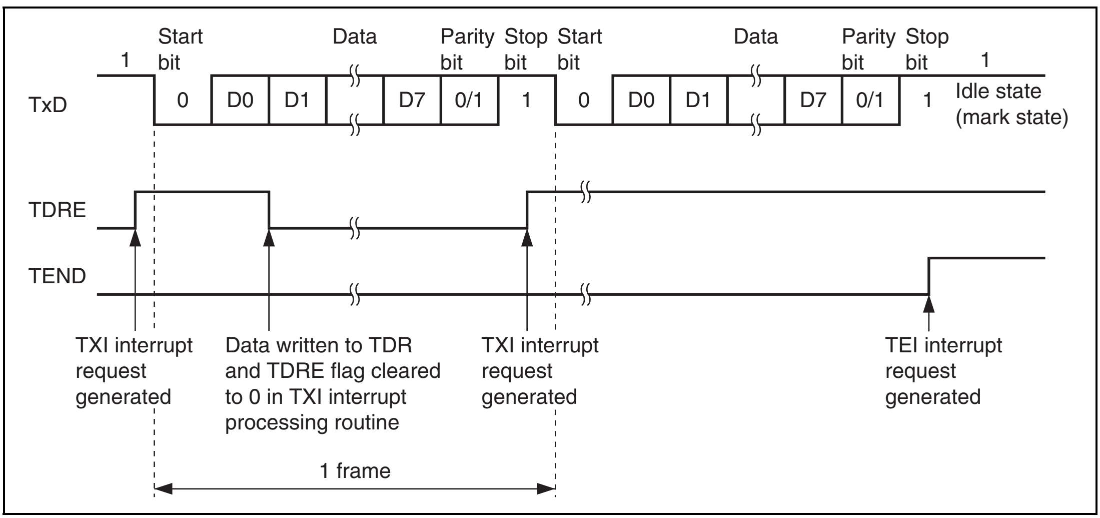
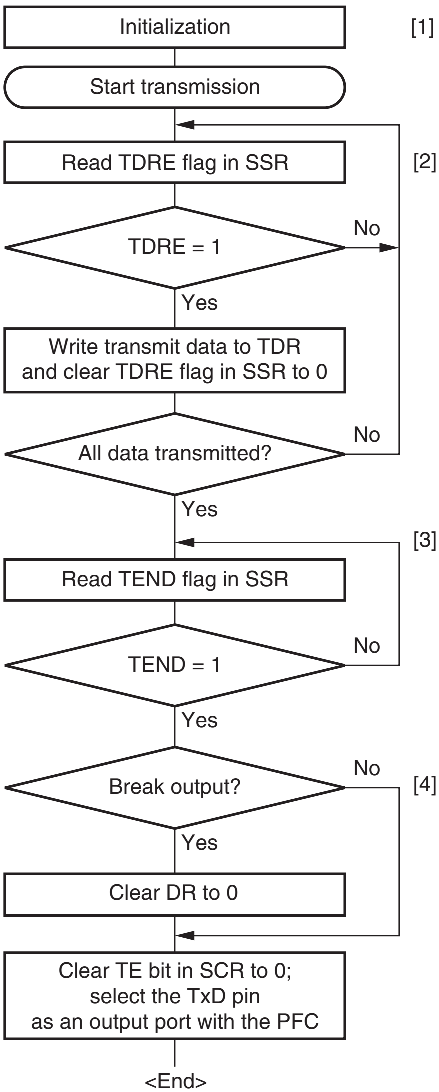


Figure 10.6 Example of Operation in Transmission in Asynchronous Mode (Example with 8-Bit Data, Parity, One Stop Bit)



[1] SCI initialization:
Set the TxD pin using the PFC. After the TE bit is set to 1, 1 is output for one frame, and transmission is enabled. However, data is not transmitted.

[2] SCI status check and transmit data write:
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR and clear the TDRE flag to 0.

[3] Serial transmission continuation procedure:

To continue serial transmission, read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR, and then clear the TDRE flag to 0.

[4] Break output at the end of serial transmission:
To output a break in serial transmission, first clear the port data register (DR) to 0, then clear the TE bit to 0 in SCR and use the PFC to select the TxD pin as an output port.

Figure 10.7 Sample Serial Transmission Flowchart

10.4.6 Serial Data Reception (Asynchronous Mode)

Figure 10.8 shows an example of the operation for reception in asynchronous mode. In serial reception, the SCI operates as described below.

1. The SCI monitors the communication line, and if a start bit is detected, performs internal synchronization, receives receive data in RSR, and checks the parity bit and stop bit.
2. If an overrun error (when reception of the next data is completed while the RDRF flag is still set to 1) occurs, the OER bit in SSR is set to 1. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR. The RDRF flag remains to be set to 1.
3. If a parity error is detected, the PER bit in SSR is set to 1 and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated.
4. If a framing error (when the stop bit is 0) is detected, the FER bit in SSR is set to 1 and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated.
5. If reception finishes successfully, the RDRF bit in SSR is set to 1, and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an RXI interrupt request is generated. Because the RXI interrupt processing routine reads the receive data transferred to RDR before reception of the next receive data has finished, continuous reception can be enabled.

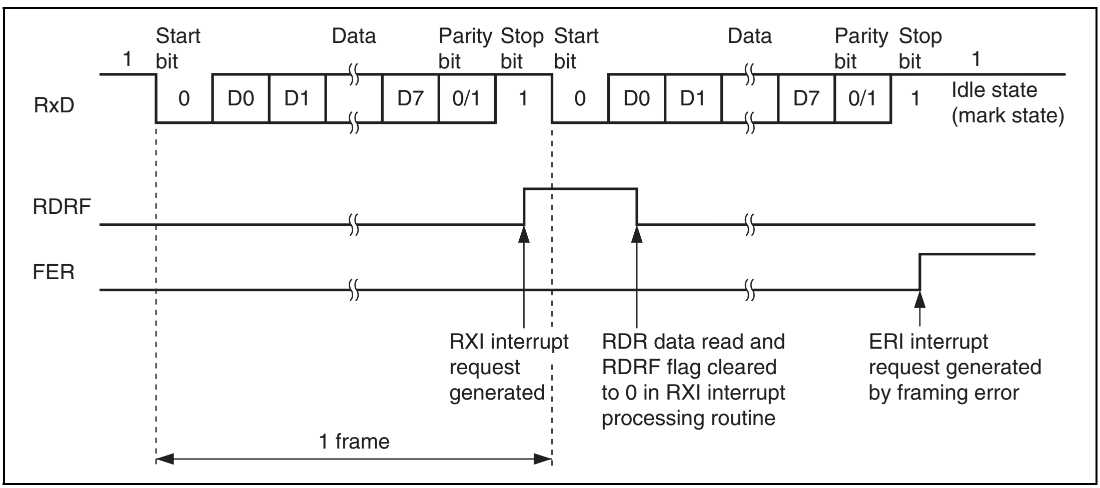


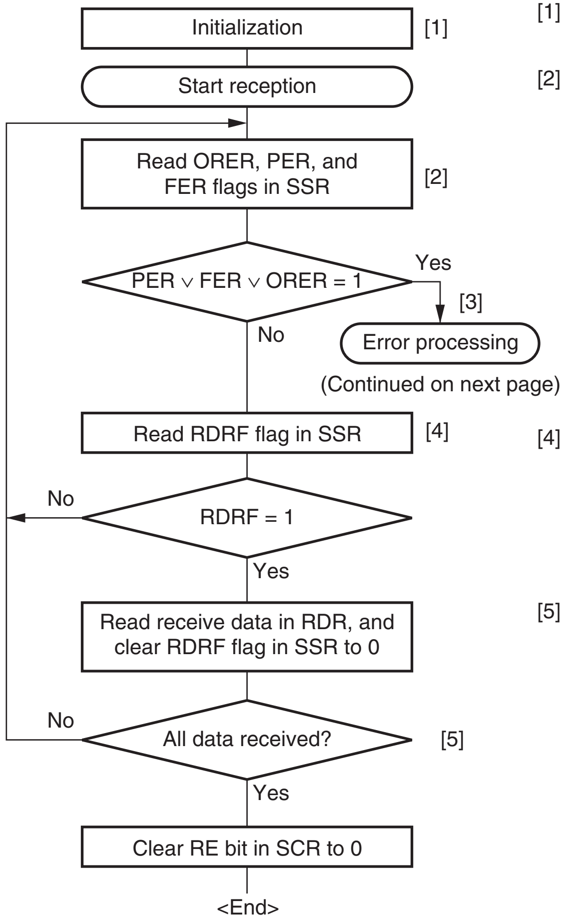
Figure 10.8 Example of SCI Operation in Reception (Example with 8-Bit Data, Parity, One Stop Bit)

Table 10.9 shows the states of the SSR status flags and receive data handling when a receive error is detected. If a receive error is detected, the RDRF flag retains its state before receiving data. Reception cannot be resumed while a receive error flag is set to 1. Accordingly, clear the OER, FER, PER, and RDRF bits to 0 before resuming reception. Figure 10.9 shows a sample flow chart for serial data reception.

Table 10.9 SSR Status Flags and Receive Data Handling

SSR Status Flag					
RDRF*	**OER**	**FER**	**PER**	**Receive Data**	**Receive Error Type**
1	1	0	0	Lost	Overrun error
0	0	1	0	Transferred to RDR	Framing error
0	0	0	1	Transferred to RDR	Parity error
1	1	1	0	Lost	Overrun error + framing error
1	1	0	1	Lost	Overrun error + parity error
0	0	1	1	Transferred to RDR	Framing error + parity error
1	1	1	1	Lost	Overrun error + framing error + parity error

Note: * The RDRF flag retains its state before data reception.



[1] SCI initialization:
Set the RxD pin using the PFC.

[2] [3] Receive error processing and break detection:
If a receive error occurs, read the ORER, PER, and FER flags in SSR to identify the error. After performing the appropriate error processing, ensure that the ORER, PER, and FER flags are all cleared to 0. Reception cannot be resumed if any of these flags are set to 1. In the case of a framing error, a break can be detected by reading the value of the input port corresponding to the RxD pin.

[4] SCI status check and receive data read:
Read SSR and check that RDRF = 1, then read the receive data in RDR and clear the RDRF flag to 0. Transition of the RDRF flag from 0 to 1 can also be identified by an RXI interrupt.

[5] Serial reception continuation procedure:
To continue serial reception, before the stop bit for the current frame is received, read the RDRF flag, read RDR, and clear the RDRF flag to 0.

Figure 10.9 Sample Serial Reception Data Flowchart (1)

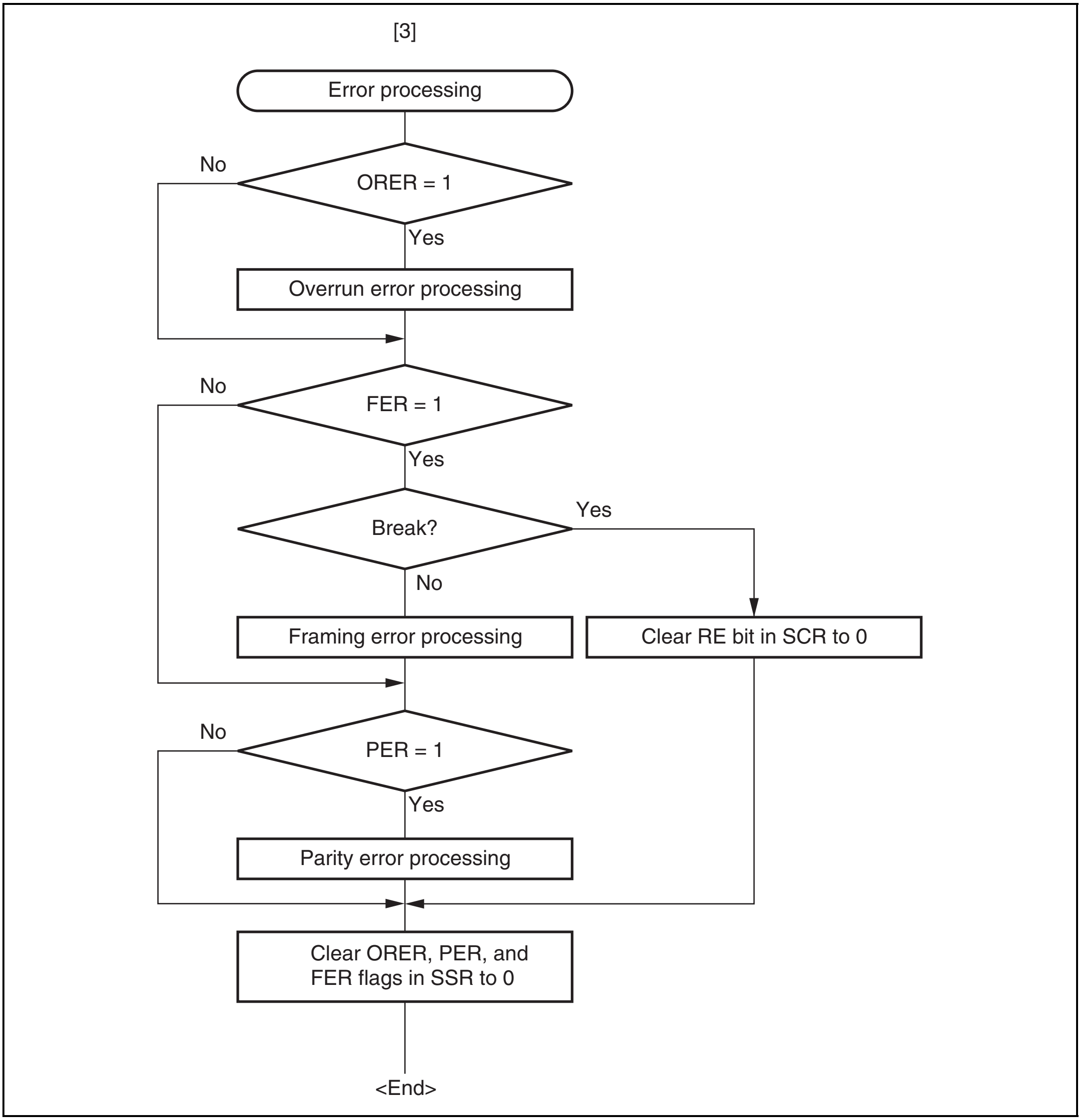


Figure 10.9 Sample Serial Reception Data Flowchart (2)

10.5 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 10.10 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 SCR to implement this function. When the MPIE bit is set to 1, transfer of receive data from RSR to RDR, error flag detection, and setting the SSR 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 SSR is set to 1 and the MPIE bit is automatically cleared, thus normal reception is resumed. If the RIE bit in SCR 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.

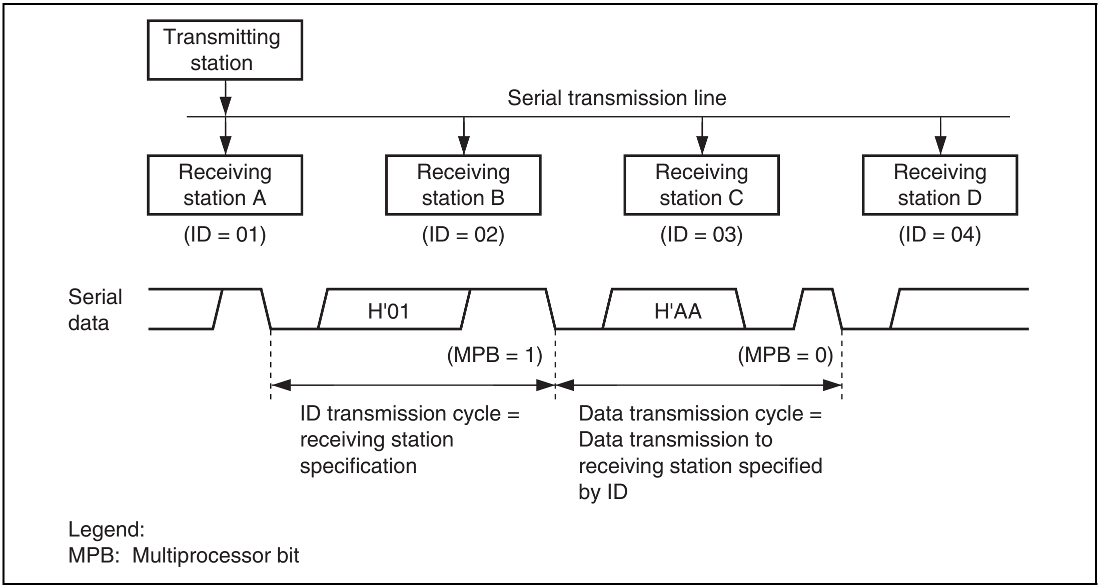
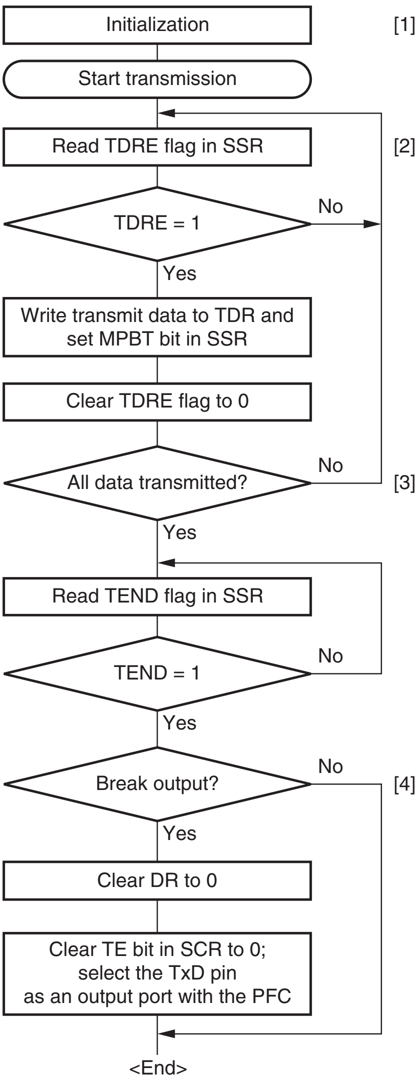


Figure 10.10 Example of Communication Using Multiprocessor Format (Transmission of Data H'AA to Receiving Station A)

10.5.1 Multiprocessor Serial Data Transmission

Figure 10.11 shows a sample flowchart for multiprocessor serial data transmission. For an ID transmission cycle, set the MPBT bit in SSR to 1 before transmission. For a data transmission cycle, clear the MPBT bit in SSR to 0 before transmission. All other SCI operations are the same as those in asynchronous mode.



[1] SCI initialization:
Set the TxD pin using the PFC. After the TE bit is set to 1, 1 is output for one frame, and transmission is enabled. However, data is not transmitted.

[2] SCI status check and transmit data write:
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR. Set the MPBT bit in SSR to 0 or 1. Finally, clear the TDRE flag to 0.

[3] Serial transmission continuation procedure:
To continue serial transmission, be sure to read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR, and then clear the TDRE flag to 0.

[4] Break output at the end of serial transmission:
To output a break in serial transmission, first clear the port data register (DR) to 0, then clear the TE bit to 0 in SCR and use the PFC to select the TxD pin as an output port.

Figure 10.11 Sample Multiprocessor Serial Transmission Flowchart

10.5.2 Multiprocessor Serial Data Reception

Figure 10.13 shows a sample flowchart for multiprocessor serial data reception. If the MPIE bit in SCR 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 RDR. An RXI interrupt request is generated at this time. All other SCI operations are the same as in asynchronous mode. Figure 10.12 shows an example of SCI operation for multiprocessor format reception.

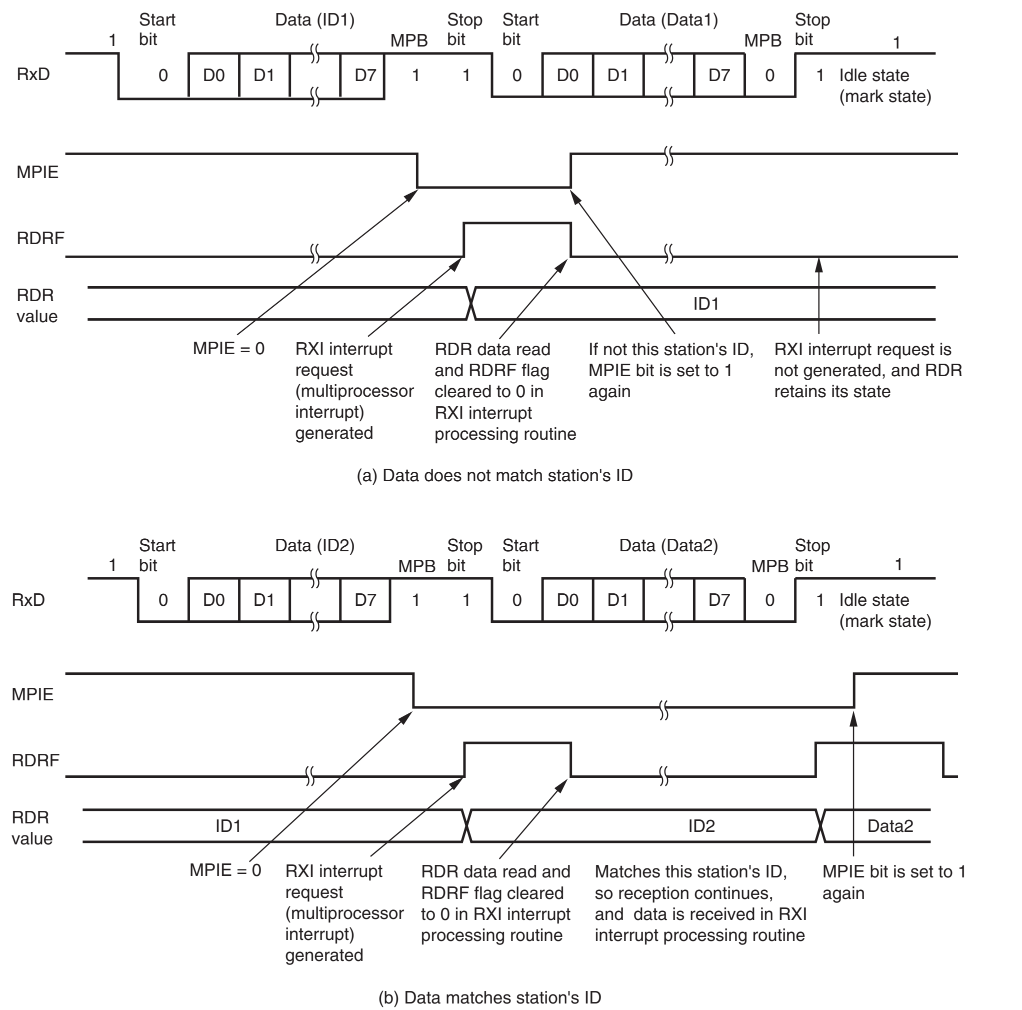
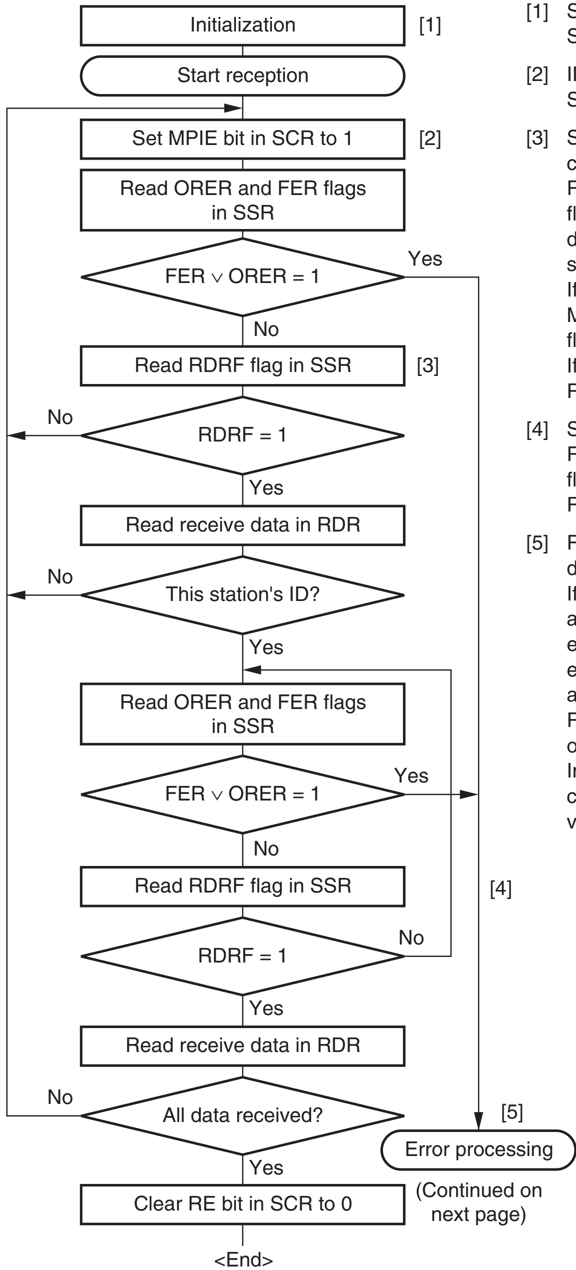


Figure 10.12 Example of SCI Operation in Reception (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit)



[1] SCI initialization:
Set the RxD pin using the PFC.

[2] ID reception cycle:
Set the MPIE bit in SCR to 1.

[3] SCI status check, ID reception and comparison:
Read SSR and check that the RDRF flag is set to 1, then read the receive data in RDR and compare it with this station's ID.
If the data is not this station's ID, set the MPIE bit to 1 again, and clear the RDRF flag to 0.
If the data is this station's ID, clear the RDRF flag to 0.

[4] SCI status check and data reception:
Read SSR and check that the RDRF flag is set to 1, then read the data in RDR.

[5] Receive error processing and break detection:
If a receive error occurs, read the ORER and FER flags in SSR to identify the error. After performing the appropriate error processing, ensure that the ORER and FER flags are all cleared to 0.
Reception cannot be resumed if either of these flags is set to 1.
In the case of a framing error, a break can be detected by reading the RxD pin value.

Figure 10.13 Sample Multiprocessor Serial Reception Flowchart (1)

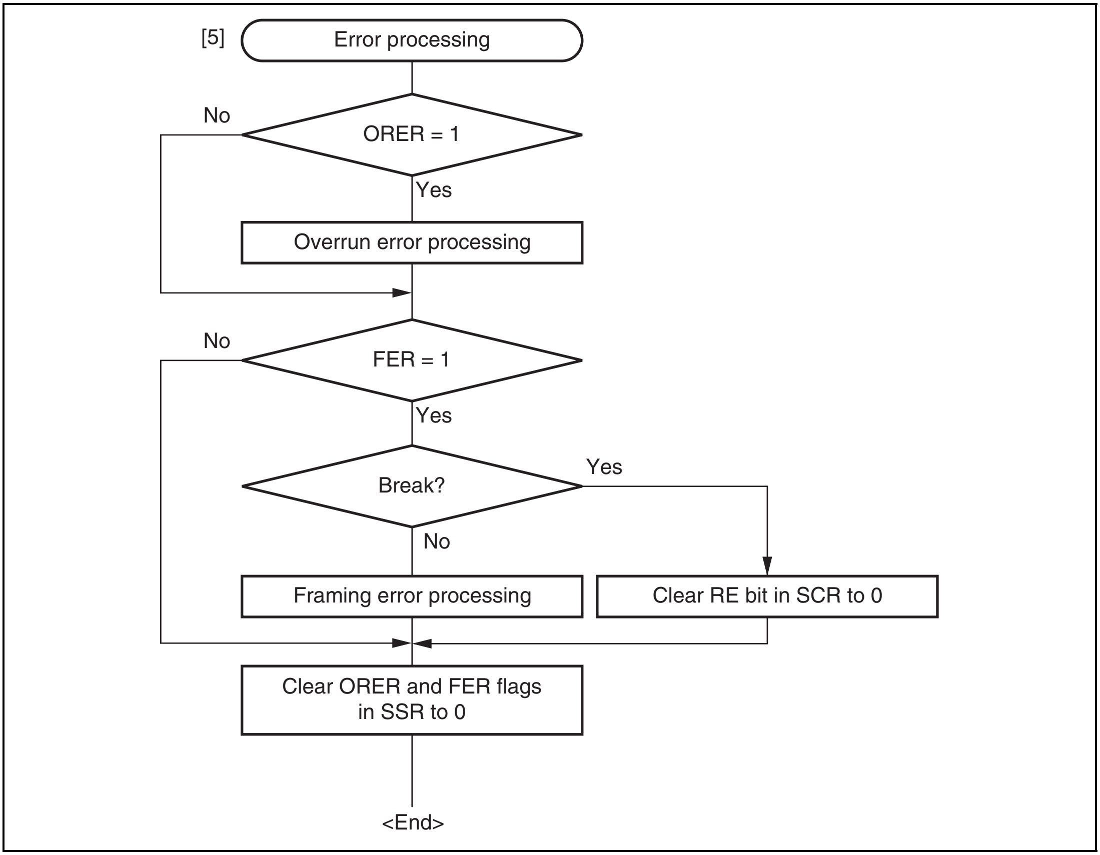


Figure 10.13 Sample Multiprocessor Serial Reception Flowchart (2)

10.6 Operation in Clocked Synchronous Mode

Figure 10.14 shows the general format for clocked synchronous communication. In clocked synchronous mode, data is transmitted or received in synchronization with clock pulses. Data is transferred in 8-bit units. In clocked synchronous serial communication, data on the transmission line is output from one falling edge of the serial clock to the next. In clocked synchronous mode, the SCI receives data in synchronization with the rising edge of the serial clock. After 8-bit data is output, the transmission line holds the MSB state. In clocked synchronous mode, no parity or multiprocessor bit is added. Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communication by use of a common clock. Both the transmitter and the receiver also have a double-buffered structure, so that data can be read or written during transmission or reception, enabling continuous data transfer.

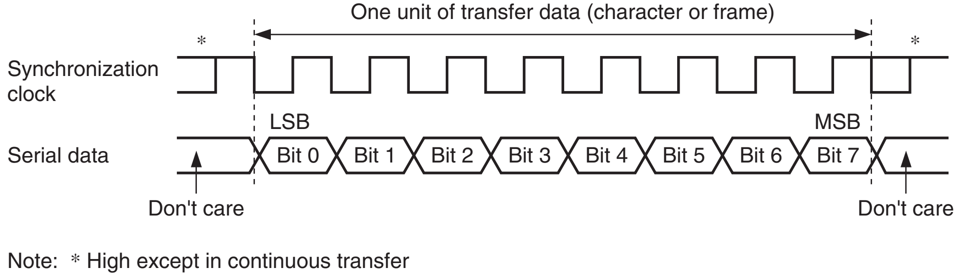


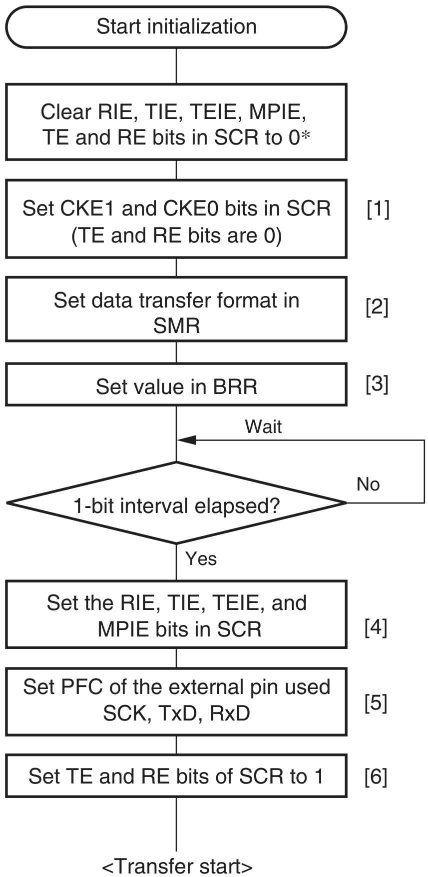
Figure 10.14 Data Format in Clocked Synchronous Communication (For LSB-First)

10.6.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external synchronization clock input at the SCK pin can be selected, according to the setting of CKE1 and CKE0 bits in SCR. When the SCI is operated on an internal clock, the serial clock is output from the SCK pin. Eight serial clock pulses are output in the transfer of one character, and when no transfer is performed, the clock is fixed high.

10.6.2 SCI Initialization (Clocked Synchronous Mode)

Before transmitting and receiving data, you should first clear the TE and RE bits in SCR to 0, then initialize the SCI as described in a sample flowchart in figure 10.15. When the operating mode, transfer format, etc., is changed, the TE and RE bits must be cleared to 0 before making the change using the following procedure. When the TE bit is cleared to 0, the TDRE flag is set to 1. Note that clearing the RE bit to 0 does not initialize the RDRF, PER, FER, and ORER flags, or the contents of RDR.



[1] Set the clock selection in SCR.

[2] Set the data transfer format in SMR.

[3] Write a value corresponding to the bit rate to BRR. Not necessary if an external clock is used.

[4] Set the RIE, TIE TEIE, and MPIE bits.

[5] Set PFC of the external pin used. Set RxD input during receiving and TxD output during transmitting. Set SCK input/output according to contents set by CKE1 and CKE0.

[6] Wait at least one bit interval, then set the TE bit or RE bit in SCR to 1.* At this time, the TxD, RxD, and SCK pins can be used. The TxD pin is in a mark state during transmitting. When synchronous clock output (clock master) is set during receiving in synchronous mode, outputting clocks from the SCK pin starts.

Note: * In simultaneous transmit and receive operations, the TE and RE bits should both be cleared to 0 or set to 1 simultaneously.

Figure 10.15 Sample SCI Initialization Flowchart

10.6.3 Serial Data Transmission (Clocked Synchronous Mode)

Figure 10.16 shows an example of SCI operation for transmission in clocked synchronous mode. In serial transmission, the SCI operates as described below.

1. The SCI monitors the TDRE flag in SSR, and if it is cleared to 0, recognizes that data has been written to TDR, and transfers the data from TDR to TSR.
2. After transferring data from TDR to TSR, the SCI sets the TDRE flag to 1 and starts transmission. If the TIE bit in SCR is set to 1 at this time, a transmit data empty (TXI) interrupt request is generated. Because the TXI interrupt routine writes the next transmit data to TDR before transmission of the current transmit data has finished, continuous transmission can be enabled.
3. 8-bit data is sent from the TxD pin synchronized with the output clock when output clock mode has been specified and synchronized with the input clock when use of an external clock has been specified.
4. The SCI checks the TDRE flag at the timing for sending the MSB (bit 7).
5. If the TDRE flag is cleared to 0, data is transferred from TDR to TSR, and serial transmission of the next frame is started.
6. If the TDRE flag is set to 1, the TEND flag in SSR is set to 1, and the TxD pin maintains the output state of the last bit. If the TEIE bit in SCR is set to 1 at this time, a TEI interrupt request is generated. The SCK pin is fixed high.

Figure 10.17 shows a sample flowchart for serial data transmission. Even if the TDRE flag is cleared to 0, transmission will not start while a receive error flag (ORER, FER, or PER) is set to 1. Make sure to clear the receive error flags to 0 before starting transmission. Note that clearing the RE bit to 0 does not clear the receive error flags.

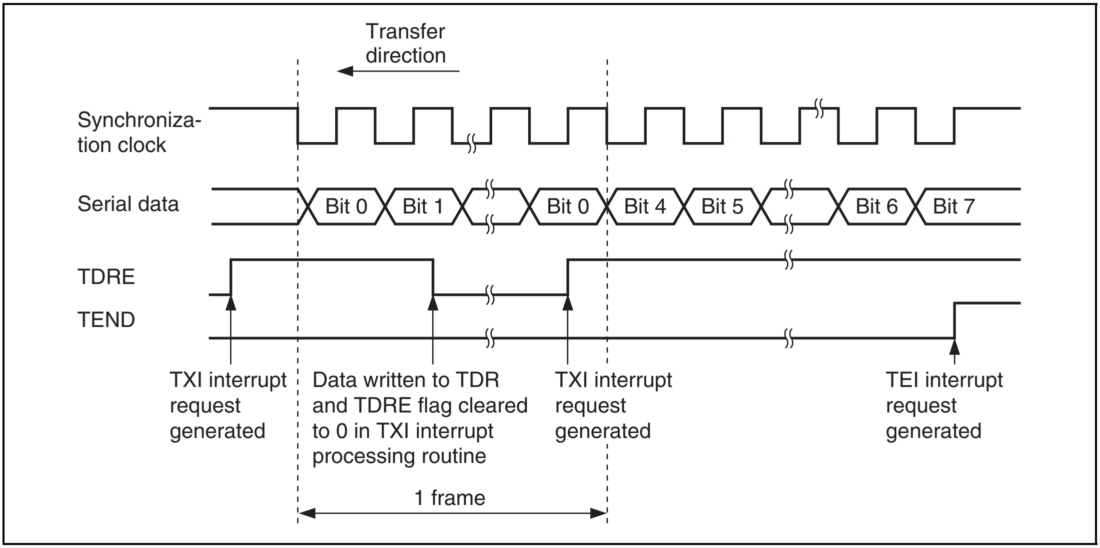
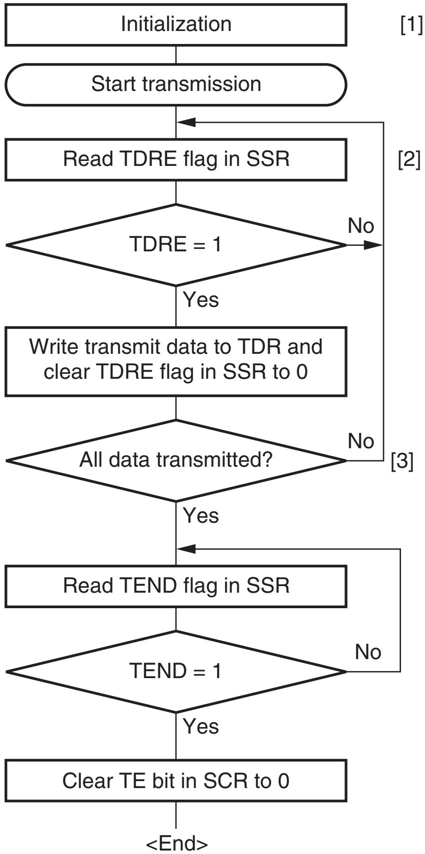


Figure 10.16 Sample SCI Transmission Operation in Clocked Synchronous Mode



[1] SCI initialization:
Set the TxD pin using the PFC.

[2] SCI status check and transmit data write:
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR and clear the TDRE flag to 0.

[3] Serial transmission continuation procedure:
To continue serial transmission, be sure to read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR, and then clear the TDRE flag to 0.

Figure 10.17 Sample Serial Transmission Flowchart

10.6.4 Serial Data Reception (Clocked Synchronous Mode)

Figure 10.18 shows an example of SCI operation for reception in clocked synchronous mode. In serial reception, the SCI operates as described below.

1. The SCI performs internal initialization in synchronization with a synchronization clock input or output, starts receiving data, and stores the received data in RSR.
2. If an overrun error (when reception of the next data is completed while the RDRF flag is still set to 1) occurs, the ORER bit in SSR is set to 1. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR. The RDRF flag remains to be set to 1.
3. If reception finishes successfully, the RDRF bit in SSR is set to 1, and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an RXI interrupt request is generated. Because the RXI interrupt processing routine reads the receive data transferred to RDR before reception of the next receive data has finished, continuous reception can be enabled.

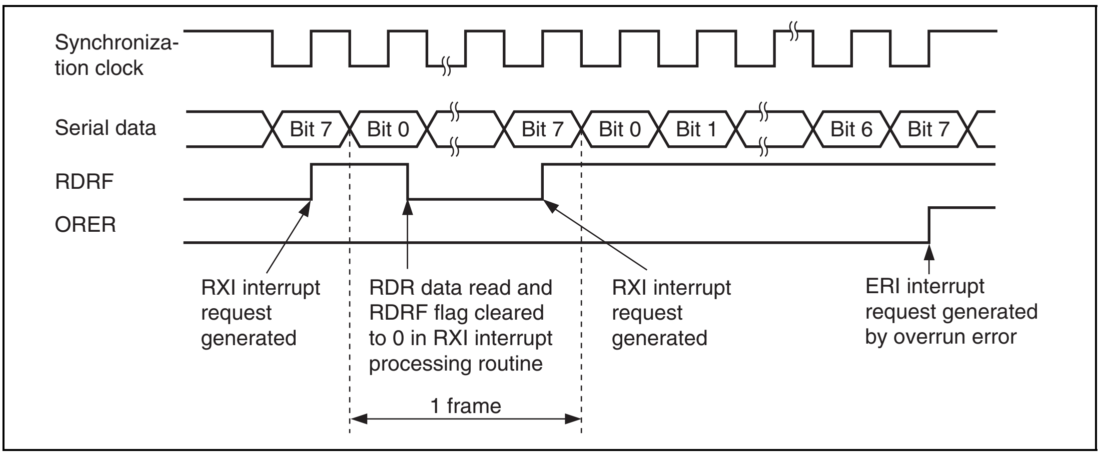
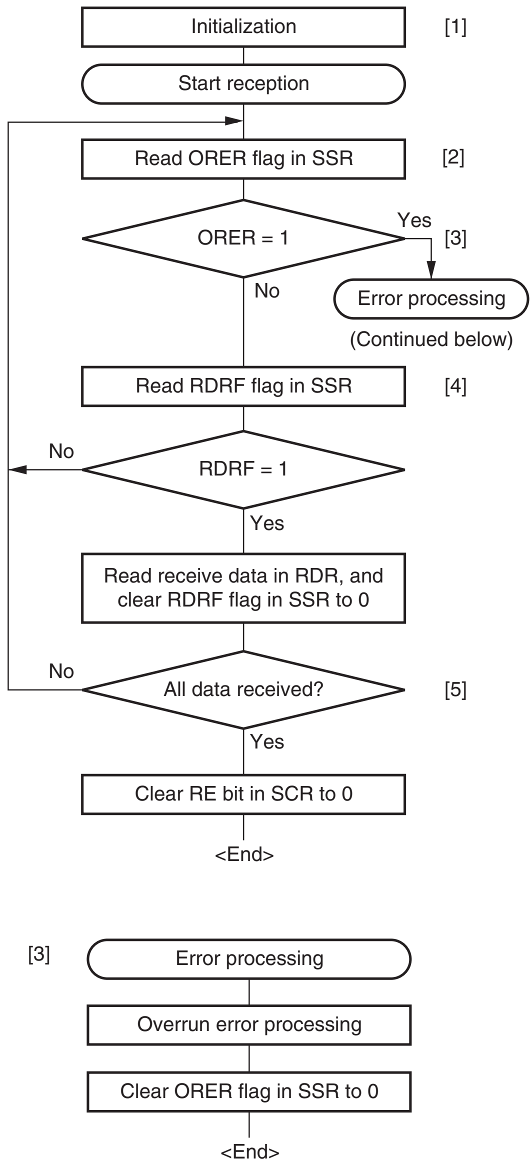


Figure 10.18 Example of SCI Operation in Reception

Reception cannot be resumed while a receive error flag is set to 1. Accordingly, clear the ORER, FER, PER, and RDRF bits to 0 before resuming reception. Figure 10.19 shows a sample flowchart for serial data reception.



[1] SCI initialization:
Set the RxD pin using the PFC.

[2] [3] Receive error processing:
If a receive error occurs, read the ORER flag in SSR, and after performing the appropriate error processing, clear the ORER flag to 0. Transfer cannot be resumed if the ORER flag is set to 1.

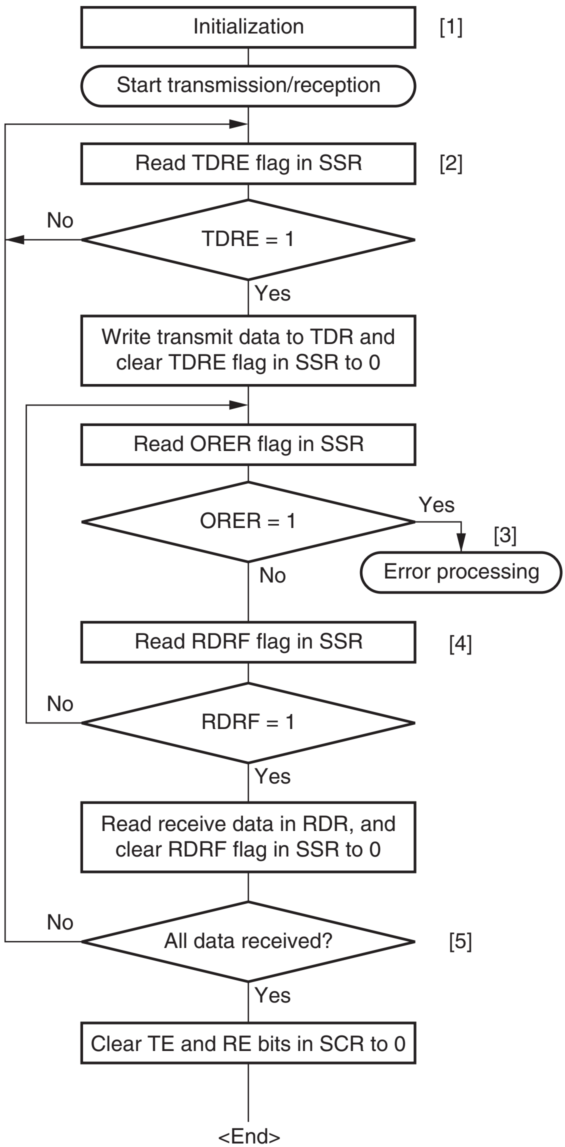
[4] SCI status check and receive data read:
Read SSR and check that the RDRF flag is set to 1, then read the receive data in RDR and clear the RDRF flag to 0.
Transition of the RDRF flag from 0 to 1 can also be identified by an RXI interrupt.

[5] Serial reception continuation procedure:
To continue serial reception, before the MSB (bit 7) of the current frame is received, reading the RDRF flag, reading RDR, and clearing the RDRF flag to 0 should be finished.

Figure 10.19 Sample Serial Reception Flowchart

10.6.5 Simultaneous Serial Data Transmission and Reception (Clocked Synchronous Mode)

Figure 10.20 shows a sample flowchart for simultaneous serial transmit and receive operations. The following procedure should be used for simultaneous serial data transmit and receive operations after the SCI initialization. To switch from transmit mode to simultaneous transmit and receive mode, after checking that the SCI has finished transmission and the TDRE and TEND flags are set to 1, clear TE to 0. Then simultaneously set TE and RE to 1 with a single instruction. To switch from receive mode to simultaneous transmit and receive mode, after checking that the SCI has finished reception, clear RE to 0. Then after checking that the RDRF and receive error flags (ORER, FER, and PER) are cleared to 0, simultaneously set TE and RE to 1 with a single instruction.



[1] SCI initialization:
Set the TxD and RxD pins using the PFC.

[2] SCI status check and transmit data write:
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR and clear the TDRE flag to 0.
Transition of the TDRE flag from 0 to 1 can also be identified by a TXI interrupt.

[3] Receive error processing:
If a receive error occurs, read the ORER flag in SSR, and after performing the appropriate error processing, clear the ORER flag to 0. Transmission/reception cannot be resumed if the ORER flag is set to 1.

[4] SCI status check and receive data read:
Read SSR and check that the RDRF flag is set to 1, then read the receive data in RDR and clear the RDRF flag to 0.
Transition of the RDRF flag from 0 to 1 can also be identified by an RXI interrupt.

[5] Serial transmission/reception continuation procedure:
To continue serial transmission/reception, before the MSB (bit 7) of the current frame is received, finish reading the RDRF flag, reading RDR, and clearing the RDRF flag to 0. Also, before the MSB (bit 7) of the current frame is transmitted, read 1 from the TDRE flag to confirm that writing is possible. Then write data to TDR 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 10.20 Sample Flowchart of Simultaneous Serial Transmit and Receive Operations

10.7 Interrupts Sources

10.7.1 Interrupts in Normal Serial Communication Interface Mode

Table 10.10 shows the interrupt sources in normal serial communication interface mode. A different interrupt vector is assigned to each interrupt source, and individual interrupt sources can be enabled or disabled using the enable bits in SCR.

When the TDRE flag in SSR is set to 1, a TXI interrupt request is generated. When the TEND flag in SSR is set to 1, a TEI interrupt request is generated.

When the RDRF flag in SSR is set to 1, an RXI interrupt request is generated. When the ORER, PER, or FER flag in SSR is set to 1, an ERI interrupt request is generated.

A TEI interrupt is generated when the TEND flag is set to 1 while the TEIE bit is set to 1. If a TEI interrupt and a TXI interrupt are generated simultaneously, the TXI interrupt has priority for acceptance. However, note that if the TDRE and TEND flags are cleared simultaneously by the TXI interrupt routine, the SCI cannot branch to the TEI interrupt routine later.

Table 10.10 SCI Interrupt Sources

Channel	Name	Interrupt Source	Interrupt Flag
2	ERI_2	Receive Error	ORER, FER, PER
	RXI_2	Receive Data Full	RDRF
	TXI_2	Transmit Data Empty	TDRE
	TEI_2	Transmission End	TEND
3	ERI_3	Receive Error	ORER, FER, PER
	RXI_3	Receive Data Full	RDRF
	TXI_3	Transmit Data Empty	TDRE
	TEI_3	Transmission End	TEND

10.8 Usage Notes

10.8.1 TDR Write and TDRE Flag

The TDRE bit in the serial status register (SSR) is a status flag indicating transferring of transmit data from TDR into TSR. The SCI sets the TDRE bit to 1 when it transfers data from TDR to TSR.

Data can be written to TDR regardless of the TDRE bit status.

If new data is written in TDR when TDRE is 0, however, the old data stored in TDR will be lost because the data has not yet been transferred to TSR. Before writing transmit data to TDR, be sure to check that the TDRE bit is set to 1.

10.8.2 Module Standby Mode Setting

SCI operation can be disabled or enabled using the module 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 17, Power-Down Modes.

10.8.3 Break Detection and Processing (Asynchronous Mode Only)

When framing error detection is performed, a break can be detected by reading the RxD pin value directly. In a break, the input from the RxD pin becomes all 0s, and so the FER flag is set, and the PER flag may also be set. Note that, since the SCI continues the receive operation after receiving a break, even if the FER flag is cleared to 0, it will be set to 1 again.

10.8.4 Sending a Break Signal (Asynchronous Mode Only)

The TxD pin becomes of the I/O port general I/O pin with the I/O direction and level determined by the port data register (DR) and the port I/O register (IOR) of the pin function controller (PFC). These conditions allow break signals to be sent.

The DR value is substituted for the marking status until the PFC is set. Consequently, the output port is set to initially output a 1.

To send a break in serial transmission, first clear the DR to 0, then establish the TxD pin as an output port using the PFC.

When the TE bit is cleared to 0, the transmission section is initialized regardless of the present transmission status.

10.8.5 Receive Error Flags and Transmit Operations (Clocked Synchronous Mode Only)

Transmission cannot be started when a receive error flag (ORER, PER, or FER) is set to 1, even if the TDRE flag is cleared to 0. Be sure to clear the receive error flags to 0 before starting transmission. Note also that receive error flags cannot be cleared to 0 even if the RE bit is cleared to 0.

10.8.6 Cautions on Clocked Synchronous External Clock Mode

1. Set TE = RE = 1 only when external clock SCK is 1.
2. Do not set TE = RE = 1 until at least four Pϕ clocks after external clock SCK has changed from 0 to 1.
3. When receiving, RDRF is 1 when RE is cleared to 0 after 2.5–3.5 Pϕ clocks from the rising edge of the RxD D7 bit SCK input, but copying to RDR is not possible.

10.8.7 Caution on Clocked Synchronous Internal Clock Mode

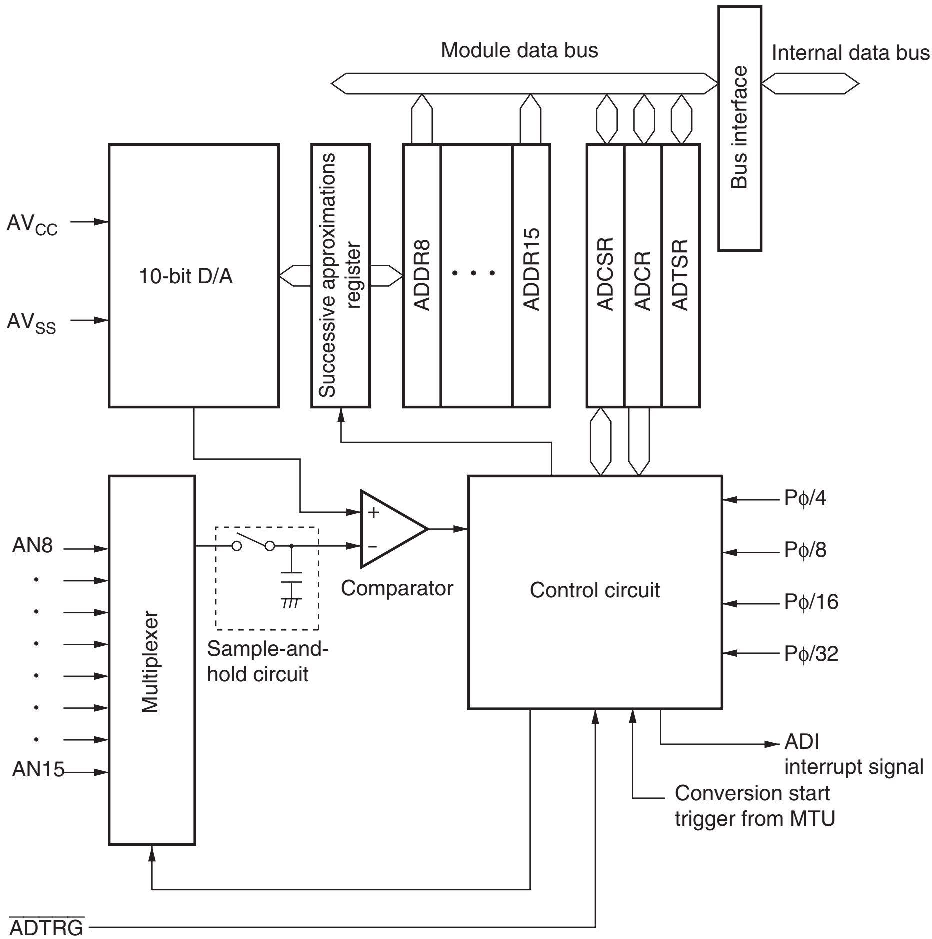
When receiving, RDRF is 1 when RE is cleared to 0 after 1.5 Pϕ clocks from the rising edge of the RxD D7 bit SCK output, but copying to RDR is not possible.

Section 11 A/D Converter

This LSI includes a successive approximation type 10-bit A/D converter. The block diagram of the A/D converter is shown in figure 11.1.

11.1 Features

- 10-bit resolution
- Input channels
  - 8 channels (two independent A/D conversion modules)
- Conversion time: 6.7 µs per channel (at Pϕ = 20-MHz operation)
  5.4 µs per channel (at Pϕ = 25-MHz operation)
- Three operating modes
  - Single mode: Single-channel A/D conversion
  - Continuous scan mode: 1 to 4 channels
  - Single-cycle scan mode: 1 to 4 channels
- 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 (MTU)
  - External trigger signal
- Interrupt source
  - An A/D conversion end interrupt request (ADI) can be generated
- Module standby mode can be set



Legend:
ADCR: A/D control register
ADCSR: A/D control/status register
ADTSR: A/D trigger select register
ADDR8–ADDR15: A/D data register 8 to 15

Note: The register number corresponds to the channel number of the module.

Figure 11.1 Block Diagram of A/D Converter (For One Module)

11.2 Input/Output Pins

Table 11.1 summarizes the input pins used by the A/D converter. This LSI has two A/D conversion modules, each of which can be operated independently. The input channels are divided into four channel sets.

Table 11.1 Pin Configuration

Module Type	Pin Name	I/O	Function	
Common	AV_{CC}	Input	Analog block power supply and 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/D0)	AN8	Input	Analog input pin 8	Group 1
	AN9	Input	Analog input pin 9	
	AN10	Input	Analog input pin 10	
	AN11	Input	Analog input pin 11	
A/D module 1 (A/D1)	AN12	Input	Analog input pin 12	Group 1
	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.

11.3 Register Descriptions

The A/D converter has the following registers. For details on register addresses and register states in each operating mode, refer to section 18, List of Registers.

- A/D data register 8 (H/L) (ADDR8)
- A/D data register 9 (H/L) (ADDR9)
- A/D data register 10 (H/L) (ADDR10)
- A/D data register 11 (H/L) (ADDR11)
- A/D data register 12 (H/L) (ADDR12)
- A/D data register 13 (H/L) (ADDR13)
- A/D data register 14 (H/L) (ADDR14)
- A/D data register 15 (H/L) (ADDR15)
- A/D control/status register_0 (ADCSR_0)
- A/D control/status register_1 (ADCSR_1)
- A/D control register_0 (ADCR_0)
- A/D control register_1 (ADCR_1)
- A/D trigger select register (ADTSR)

11.3.1 A/D Data Registers 8 to 15 (ADDR8 to ADDR15)

ADDR 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 AN8 is stored in ADDR8.)

The converted 10-bit data is stored in bits 6 to 15. The lower 6 bits are always read as 0.

The data bus between the CPU and the A/D converter is 8 bits wide. The upper byte can be read directly from the CPU, however the lower byte should be read via a temporary register. The temporary register contents are transferred from the ADDR when the upper byte data is read. When reading ADDR, read only the upper byte, or read in word unit. ADDR are initialized to H'0000.

11.3.2 A/D Control/Status Registers_0 and _1 (ADCSR_0 and ADCSR_1)

ADCSR for each module controls A/D conversion operations.

Bit	Bit Name	Initial Value	R/W	Description
7	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 condition] • When 0 is written after reading ADF = 1
6	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 in the A/D control registers (ADCR) to 0.
5 4	ADM1 ADM0	0 0	R/W R/W	A/D Mode 1 and 0 Select the A/D conversion mode. 00: Single mode 01: 4-channel scan mode 10: Setting prohibited 11: Setting prohibited When changing the operating mode, first clear the ADST bit in the A/D control registers (ADCR) to 0.
3	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
2 1 0	CH2 CH1 CH0	0 0 0	R/W R/W R/W	Channel Select 2 to 0 Select analog input channels. See table 11.2. When changing the operating mode, first clear the ADST bit in the A/D control registers (ADCR) to 0.

Note: * Only 0 can be written to clear the flag.

Table 11.2 Channel Select List

Bit 2	Bit 1	Bit 0	Analog Input Channels			
			Single Mode		4-Channel Scan Mode*	
CH2	CH1	CH0	A/D0	A/D1	A/D0	A/D1
1	0	0	AN8	AN12	AN8	AN12
		1	AN9	AN13	AN8, AN9	AN12, AN13
	1	0	AN10	AN14	AN8 to AN10	AN12 to AN14
		1	AN11	AN15	AN8 to AN11	AN12 to AN15

Note: * Continuous scan mode or single-cycle scan mode can be selected with the ADCS bit.

11.3.3 A/D Control Registers_0 and _1 (ADCR_0 and ADCR_1)

ADCR for each module controls A/D conversion started by an external trigger signal and selects the operating clock.

Bit	Bit Name	Initial Value	R/W	Description
7	TRGE	0	R/W	Trigger Enable Enables or disables triggering of A/D conversion by $\overline{\text{ADTRG}}$ or an MTU trigger. 0: A/D conversion triggering is disabled 1: A/D conversion triggering is enabled
6	CKS1	0	R/W	Clock Select 0 and 1
5	CKS0	0	R/W	Select the A/D conversion time. 00: Pϕ/32 01: Pϕ/16 10: Pϕ/8 11: Pϕ/4 When changing the A/D conversion time, first clear the ADST bit in the A/D control registers (ADCR) to 0. CKS [1,0] = b'11 can be set while P$\phi \leq$ 25 MHz.

Bit	Bit Name	Initial Value	R/W	Description
4	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 a software, reset, or in software standby mode, or module standby mode.
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 in the A/D control registers (ADCR) to 0.
2 to 0	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.

11.3.4 A/D Trigger Select Register (ADTSR)

ADTSR enables an A/D conversion started by an external trigger signal.

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	TRG1S1	0	R/W	AD Trigger 1 Select 1 and 0
2	TRG1S0	0	R/W	Enable the start of A/D conversion by A/D1 with a trigger signal. 00: A/D conversion start by external trigger pin ($\overline{\text{ADTRG}}$) or MTU trigger is enabled 01: A/D conversion start by external trigger pin ($\overline{\text{ADTRG}}$) is enabled 10: A/D conversion start by MTU trigger is enabled 11: Setting prohibited When changing the operating mode, first clear the TRGE and ADST bits in the A/D control registers (ADCR) to 0.
1	TRG0S1	0	R/W	AD Trigger 0 Select 1 and 0
0	TRG0S0	0	R/W	Enable the start of A/D conversion by A/D0 with a trigger signal. 00: A/D conversion start by external trigger pin ($\overline{\text{ADTRG}}$) or MTU trigger is enabled 01: A/D conversion start by external trigger pin ($\overline{\text{ADTRG}}$) is enabled 10: A/D conversion start by MTU trigger is enabled 11: Setting prohibited When changing the operating mode, first clear the TRGE and ADST bits in the A/D control registers (ADCR) to 0.

11.4 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. The ADST bit can be set at the same time when the operating mode or analog input channel is changed.

11.4.1 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, MTU, 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.

11.4.2 Continuous Scan Mode

In continuous scan mode, A/D conversion is to be performed sequentially on the specified channels (four channels maximum). The operations are as follows.

1. When the ADST bit in ADCR is set to 1 by software, MTU, or external trigger input, A/D conversion starts on the channel with the lowest number in the group (AN8, AN9, ..., AN11).
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.

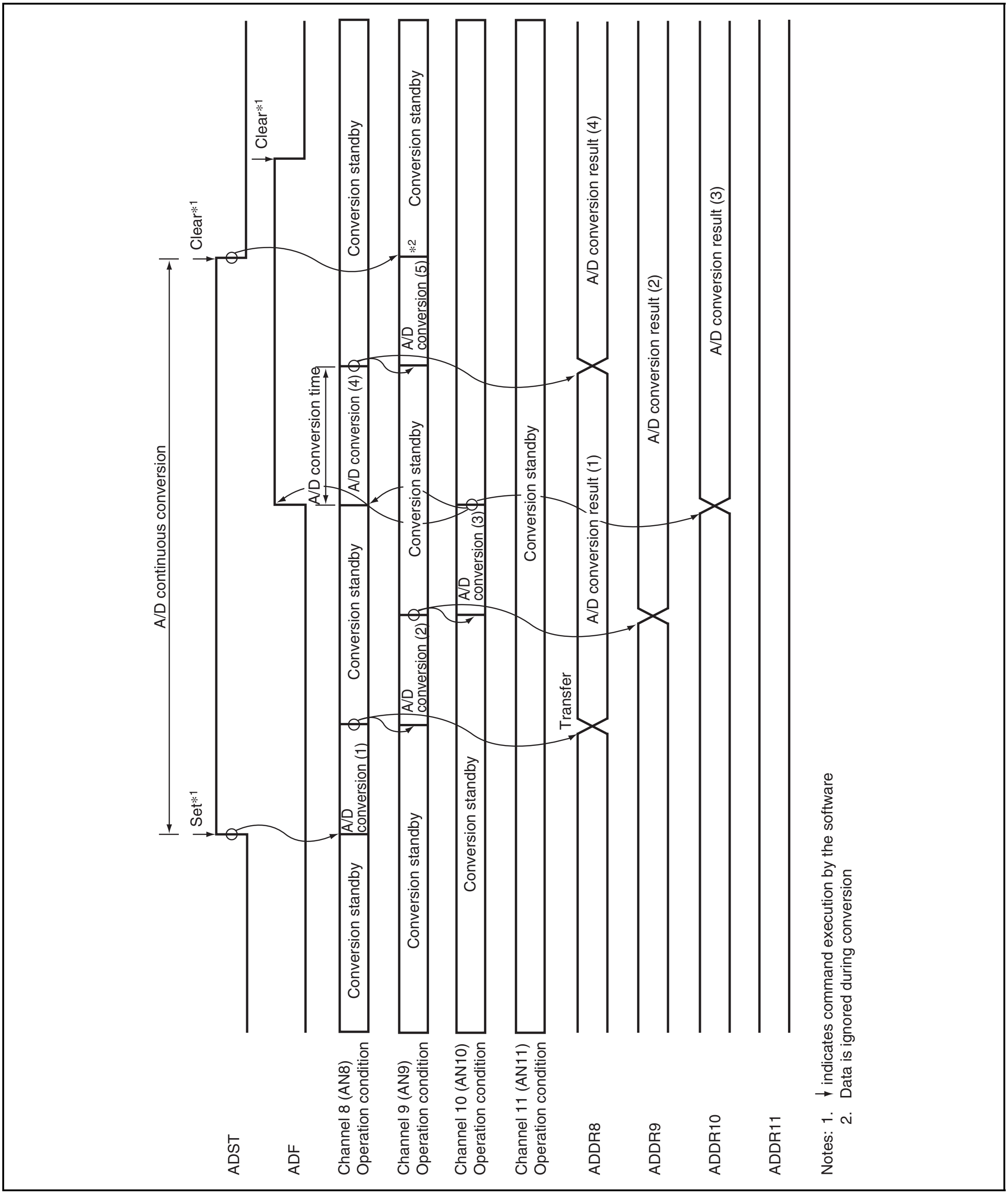


Figure 11.2 Operation Example in Continuous Scan Mode (Three Channels Selected) (AN8 to AN10)

11.4.3 Single-Cycle Scan Mode

In single-cycle scan mode, A/D conversion is to be performed once on the specified channels (four channels maximum). Operations are as follows.

1. When the ADST bit in ADCR is set to 1 by a software, MTU, or external trigger input, A/D conversion starts on the channel with the lowest number in the group (AN8, AN9, ..., AN11).
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.

11.4.4 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 11.3 shows the A/D conversion timing. Table 11.3 shows the A/D conversion time.

As indicated in figure 11.3, 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 11.3.

In scan mode, the values given in table 11.3 apply to the first conversion time. The values given in table 11.4 apply to the second and subsequent conversions.

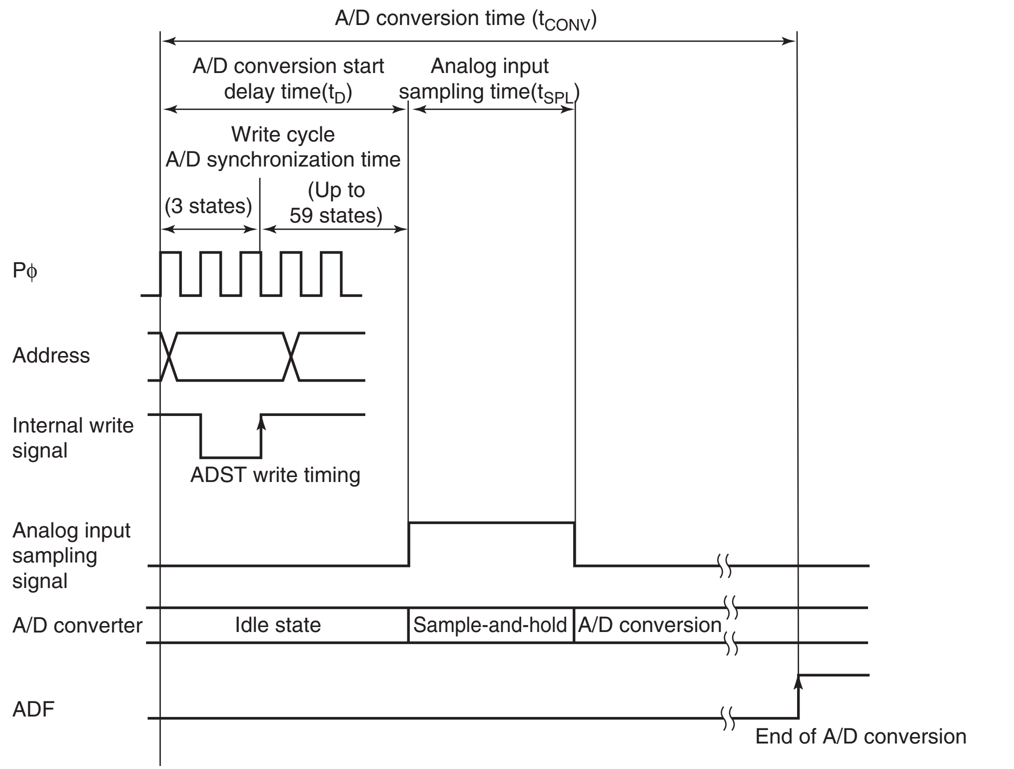


Figure 11.3 A/D Conversion Timing

Table 11.3 A/D Conversion Time (Single Mode)

		CKS1 = 0						CKS1 = 1					
		CKS0 = 0			CKS0 = 1			CKS0 = 0			CKS0 = 1		
Item	**Symbol**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**
A/D conversion start delay time	t_D	31	—	62	15	—	30	7	—	14	3	—	6
Input sampling time	t_{SPL}	—	256	—	—	128	—	—	64	—	—	32	—
A/D conversion time	t_{CONV}	1024	—	1055	515	—	530	259	—	266	131	—	134

Note: All values represent the number of states for Pφ.

Table 11.4 A/D Conversion Time (Scan Mode)

CKS1	CKS0	Conversion Time (State)
0	0	1024 (Fixed)
	1	512 (Fixed)
1	0	256 (Fixed)
	1	128 (Fixed)

11.4.5 A/D Converter Activation by MTU

The A/D converter can be independently activated by an A/D conversion request from the interval timer of the MTU.

To activate the A/D converter by the MTU, set the A/D trigger select register (ADTSR). When the TRGS1 and TRGS0 bits in ADTSR are set to 00 or 01, if an A/D conversion request from the interval timer of the MTU occurs, the ADST bit in ADCR is automatically set to 1. 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.

11.4.6 External Trigger Input Timing

A/D conversion can be externally triggered. When the TRGS0 and TRGS1 bits are set to 00 or 01 in ADTSR, 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 11.4 shows the timing.

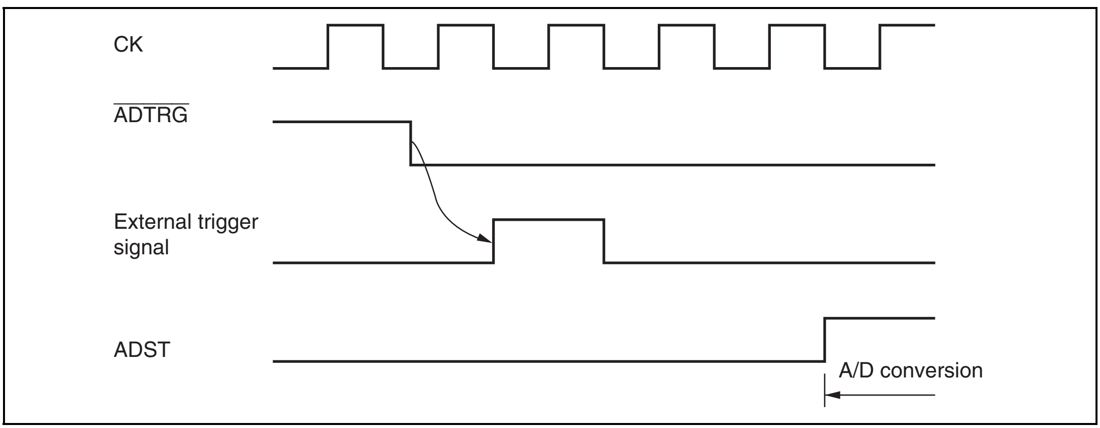


Figure 11.4 External Trigger Input Timing

11.5 Interrupt Sources

The A/D converter generates an A/D conversion end interrupt (ADI) upon the completion of A/D conversion. ADI interrupt requests are enabled when the ADIE bit is set to 1 while the ADF bit in ADCSR is set to 1 after A/D conversion is completed.

The A/D converter can generate an A/D conversion end interrupt request. 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.

Table 11.5 A/D Converter Interrupt Source

Name	Interrupt Source	Interrupt Source Flag
ADI	A/D conversion completed	ADF

11.6 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 11.5).
- 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'00) to B'0000000001 (H'01) (see figure 11.6).
- 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 11.6).
- 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 11.6).
- Absolute accuracy

  The deviation between the digital value and the analog input value. Includes offset error, full-scale error, quantization error, and nonlinearity error.

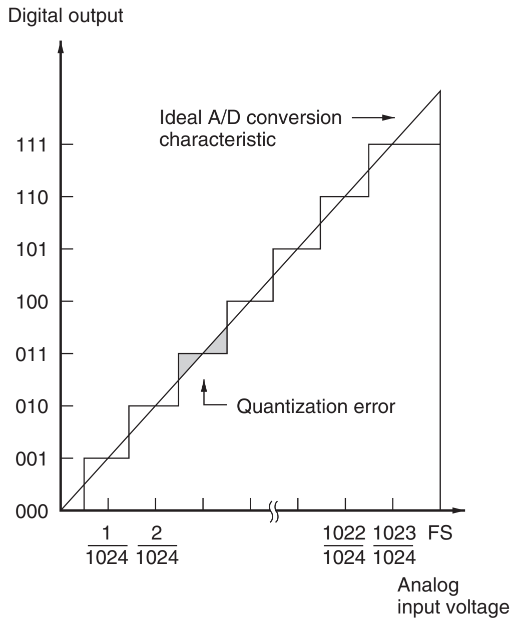


Figure 11.5 Definitions of A/D Conversion Accuracy

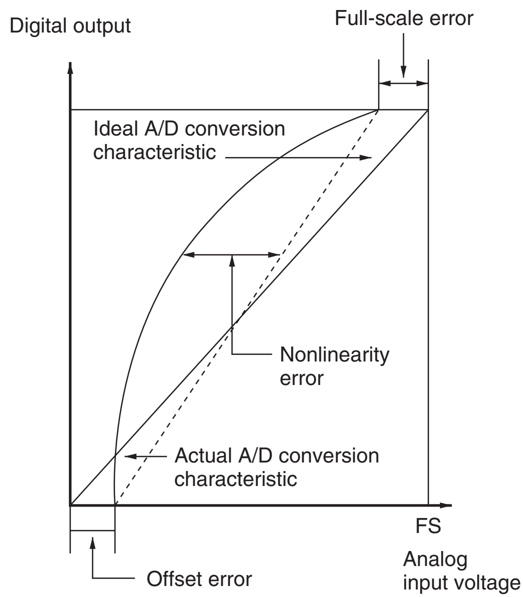


Figure 11.6 Definitions of A/D Conversion Accuracy

11.7 Usage Notes

11.7.1 Module Standby Mode Setting

Operation of the A/D converter can be disabled or enabled using the module 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 17, Power-Down Modes.

11.7.2 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, or 3 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 or 3 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 11.7). When converting a high-speed analog signal or converting in scan mode, a low-impedance buffer should be inserted.

11.7.3 Influences on Absolute Accuracy

Adding capacitance results in coupling with GND, and therefore noise in GND may adversely affect absolute accuracy. 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 communicate with digital signals on the mounting board (i.e., acting as antennas).

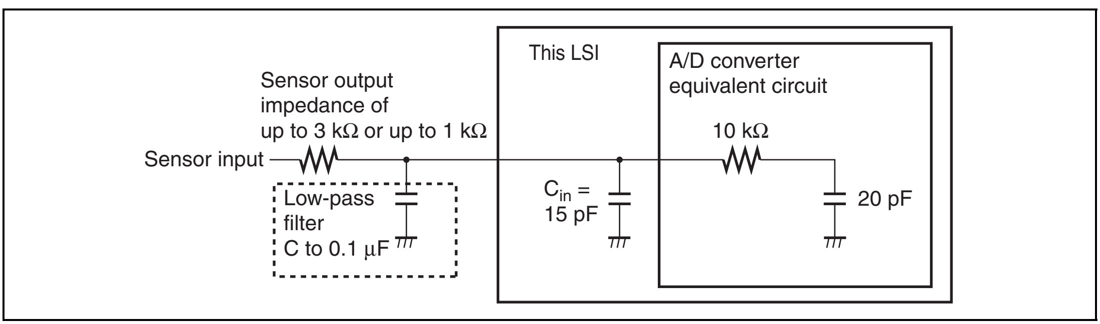


Figure 11.7 Example of Analog Input Circuit

11.7.4 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 ≤ AVcc.
- 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.

11.7.5 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 (AN8 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.

11.7.6 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 (AN8 to AN15), between AVcc and AVss, as shown in figure 11.8. Also, the bypass capacitors connected to AVcc and the filter capacitor connected to AN8 to AN15 must be connected to AVss.

If a filter capacitor is connected, the input currents at the analog input pins (AN8 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.

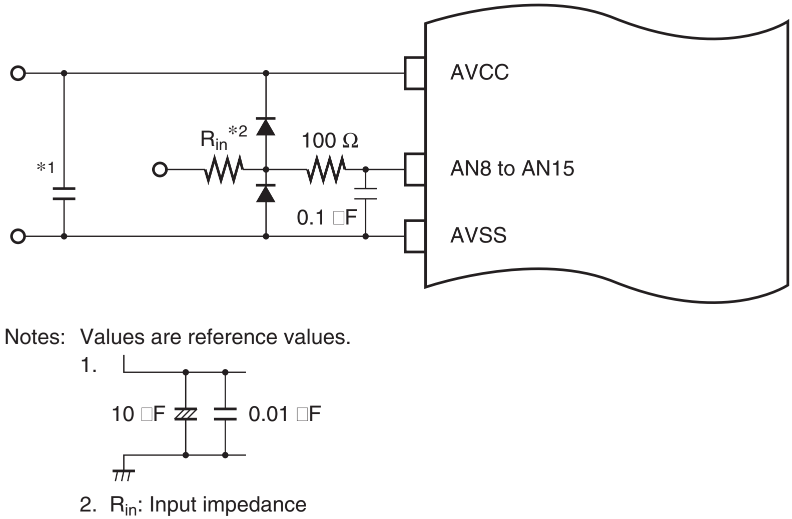


Figure 11.8 Example of Analog Input Protection Circuit

Table 11.6 Analog Pin Specifications

Item	Min	Max	Unit	Measurement Condition
Analog input capacitance	—	20	pF	
Permissible signal source impedance	—	3	kΩ	$p\phi \leq 20$ MHz
	—	1	kΩ	20 MHz $< P\phi \leq$ 25 MHz

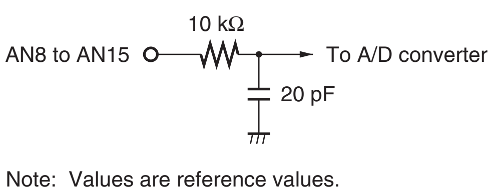


Figure 11.9 Analog Input Pin Equivalent Circuit

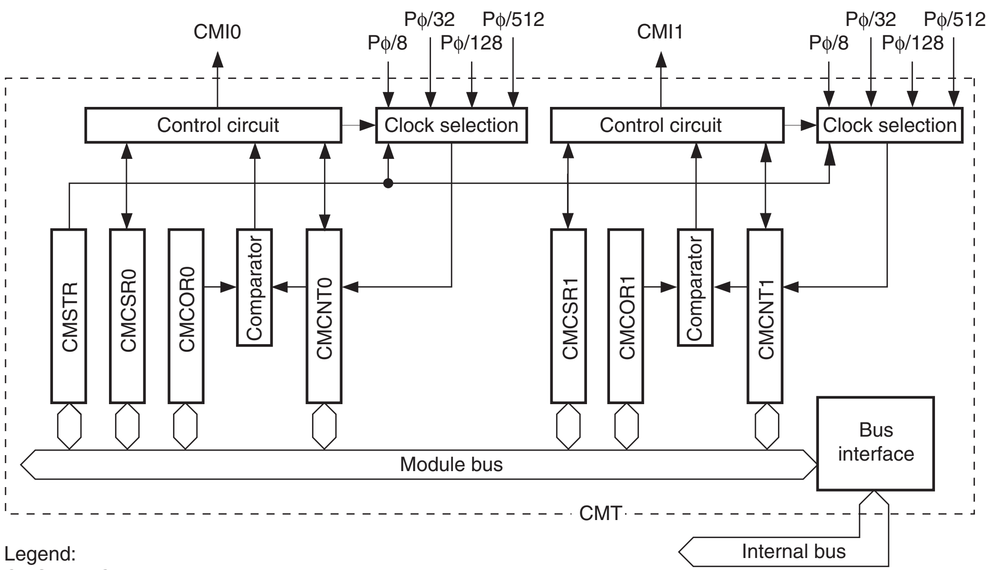
Section 12 Compare Match Timer (CMT)

This LSI has an on-chip compare match timer (CMT) comprising two 16-bit timer channels. The CMT has 16-bit counters and can generate interrupts at set intervals.

12.1 Features

- Four types of counter input clock can be selected
  — One of four internal clocks (Pϕ/8, Pϕ/32, Pϕ/128, Pϕ/512) can be selected independently for each channel.
- Interrupt sources
  — A compare match interrupt can be requested independently for each channel.
- Module standby mode can be set

Figure 12.1 shows a block diagram of the CMT.



Legend:
CMSTR: Compare match timer start register
CMCSR: Compare match timer control/status register
CMCOR: Compare match timer constant register
CMCNT: Compare match timer counter
CMI: Compare match interrupt

Figure 12.1 CMT Block Diagram

12.2 Register Descriptions

The CMT has the following registers. For details on register addresses and register states during each processing, refer to section 18, List of Registers.

- Compare Match Timer Start Register (CMSTR)
- Compare Match Timer Control/Status Register_0 (CMCSR_0)
- Compare Match Timer Counter_0 (CMCNT_0)
- Compare Match Timer Constant Register_0 (CMCOR_0)
- Compare Match Timer Control/Status Register_1 (CMCSR_1)
- Compare Match Timer Counter_1 (CMCNT_1)
- Compare Match Timer Constant Register_1 (CMCOR_1)

12.2.1 Compare Match Timer Start Register (CMSTR)

CMSTR is a 16-bit register that selects whether to operate or halt the channel 0 and channel 1 counters (CMCNT).

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 This bit selects whether to operate or halt compare match timer counter_1. 0: CMCNT_1 count operation halted 1: CMCNT_1 count operation
0	STR0	0	R/W	Count Start 0 This bit selects whether to operate or halt compare match timer counter_0. 0: CMCNT_0 count operation halted 1: CMCNT_0 count operation

12.2.2 Compare Match Timer Control/Status Register_0 and _1 (CMCSR_0, CMCSR_1)

CMCSR is a 16-bit register that indicates the occurrence of compare matches, sets the enable/disable status of interrupts, and establishes the clock used for incrementation.

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)*	Compare Match Flag This flag indicates whether or not the CMCNT and CMCOR values have matched. 0: CMCNT and CMCOR values have not matched [Clearing condition] • Write 0 to CMF after reading 1 from it 1: CMCNT and CMCOR values have matched
6	CMIE	0	R/W	Compare Match Interrupt Enable This bit selects whether to enable or disable a compare match interrupt (CMI) when the CMCNT and CMCOR values have matched (CMF = 1). 0: Compare match interrupt (CMI) disabled 1: Compare match interrupt (CMI) enabled
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1 0	CKS1 CKS0	0 0	R/W R/W	These bits select the clock input to CMCNT among the four internal clocks obtained by dividing the peripheral clock (Pϕ). When the STR bit in CMSTR is set to 1, CMCNT begins incrementing with the clock selected by CKS1 and CKS0. 00: Pϕ/8 01: Pϕ/32 10: Pϕ/128 11: Pϕ/512

Note: * Only 0 can be written, for flag clearing.

12.2.3 Compare Match Timer Counter_0 and _1 (CMCNT_0, CMCNT_1)

CMCNT is a 16-bit register used as an up-counter for generating interrupt requests. CMCNT is initialized to H'0000.

12.2.4 Compare Match Timer Constant Register_0 and _1 (CMCOR_0, CMCOR_1)

CMCOR is a 16-bit register that sets the period for compare match with CMCNT. CMCOR is initialized to H'FFFF.

12.3 Operation

12.3.1 Cyclic Count Operation

When an internal clock is selected with the CKS1, CKS0 bits in CMCSR and the STR bit in CMSTR is set to 1, CMCNT begins incrementing with the selected clock. When the CMCNT counter value matches that of the compare match constant register (CMCOR), the CMCNT counter is cleared to H'0000 and the CMF flag in CMCSR is set to 1. If the CMIE bit in CMCSR is set to 1 at this time, a compare match interrupt (CMI) is requested. The CMCNT counter begins counting up again from H'0000.

Figure 12.2 shows the compare match counter operation.

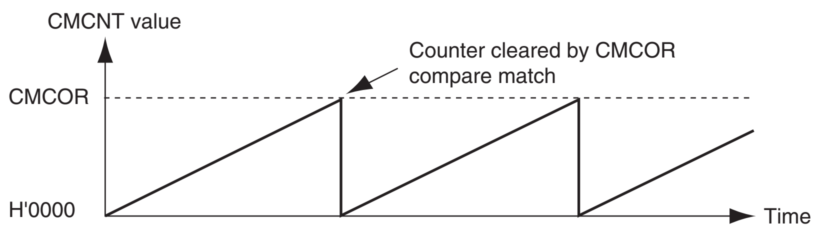


Figure 12.2 Counter Operation

12.3.2 CMCNT Count Timing

One of four internal clocks (Pϕ/8, Pϕ/32, Pϕ/128, Pϕ/512) obtained by dividing the peripheral clock (Pϕ) can be selected by the CKS1 and CKS0 bits in CMCSR. Figure 12.3 shows the timing.

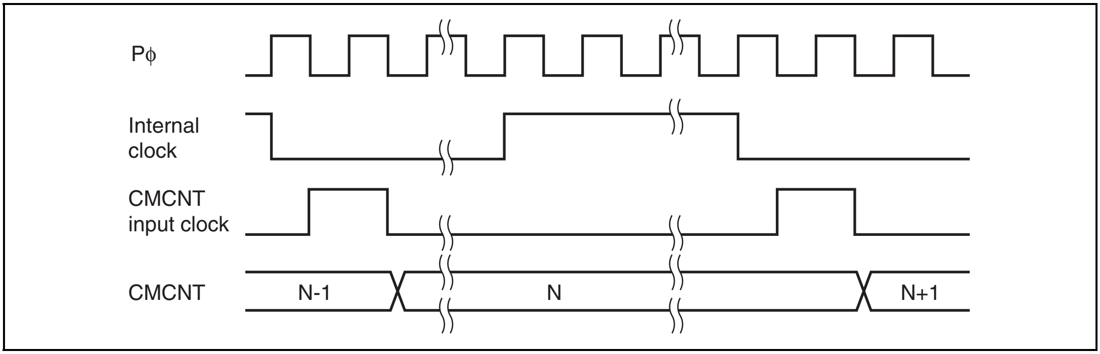


Figure 12.3 Count Timing

12.4 Interrupts

12.4.1 Interrupt Sources

The CMT has a compare match interrupt for each channel, with independent vector addresses allocated to each of them. The corresponding interrupt request is output when interrupt request flag CMF is set to 1 and interrupt enable bit CMIE has also been set to 1.

When activating CPU interrupts by interrupt request, the priority between the channels can be changed by means of interrupt controller settings. See section 6, Interrupt Controller (INTC), for details.

12.4.2 Compare Match Flag Set Timing

The CMF bit in CMCSR is set to 1 by the compare match signal generated when the CMCOR register and the CMCNT counter match. The compare match signal is generated upon the final state of the match (timing at which the CMCNT counter matching count value is updated). Consequently, after the CMCOR register and the CMCNT counter match, a compare match signal will not be generated until a CMCNT counter input clock occurs. Figure 12.4 shows the CMF bit set timing.

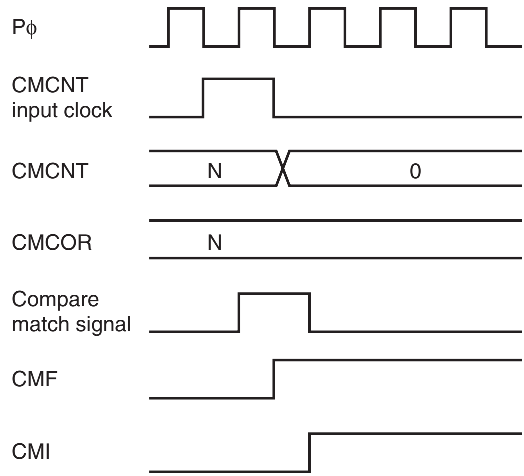


Figure 12.4 CMF Set Timing

12.4.3 Compare Match Flag Clear Timing

The CMF bit in CMCSR is cleared by writing 0 to it after reading 1. Figure 12.5 shows the timing when the CMF bit is cleared by the CPU.

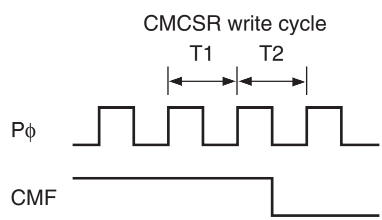


Figure 12.5 Timing of CMF Clear by CPU

12.5 Usage Notes

12.5.1 Contention between CMCNT Write and Compare Match

If a compare match signal is generated during the T2 state of the CMCNT counter write cycle, the CMCNT counter clear has priority, so the write to the CMCNT counter is not performed. Figure 12.6 shows the timing.

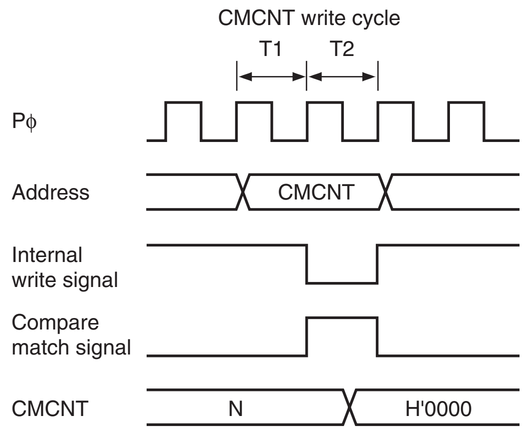


Figure 12.6 CMCNT Write and Compare Match Contention

12.5.2 Contention between CMCNT Word Write and Incrementation

If an increment occurs during the T2 state of the CMCNT counter word write cycle, the counter write has priority, so no increment occurs. Figure 12.7 shows the timing.

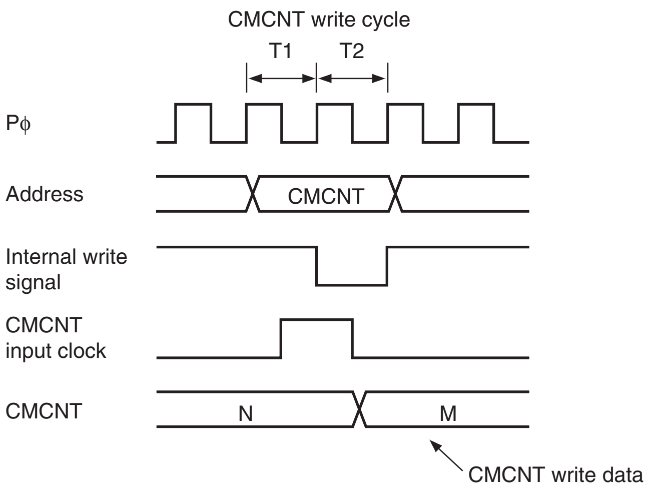


Figure 12.7 CMCNT Word Write and Increment Contention

12.5.3 Contention between CMCNT Byte Write and Incrementation

If an increment occurs during the T2 state of the CMCNT byte write cycle, the counter write has priority, so no increment of the write data results on the side on which the write was performed. The byte data on the side on which writing was not performed is also not incremented, so the contents are those before the write.

Figure 12.8 shows the timing when an increment occurs during the T2 state of the CMCNTH write cycle.

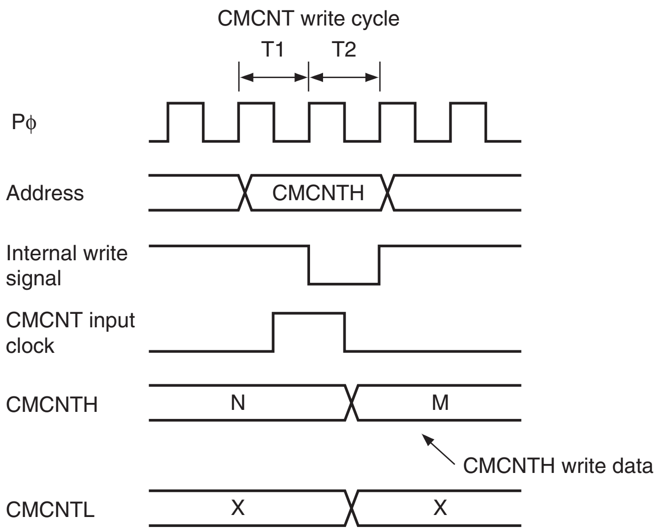


Figure 12.8 CMCNT Byte Write and Increment Contention

Section 13 Pin Function Controller (PFC)

The pin function controller (PFC) is composed of those registers that are used to select the functions of multiplexed pins and assign pins to be inputs or outputs. Tables 13.1 to 13.5 list the multiplexed pins of this LSI.

Tables 13.6 lists the pin functions in each operating mode.

Table 13.1 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)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)
A	PA0 I/O (port)	—	—	—	—	POE0 input (port)	RXD2 input (SCI)	—
	PA1 I/O (port)	—	—	—	—	POE1 input (port)	TXD2 output (SCI)	—
	PA2 I/O (port)	—	—	IRQ0 input (INTC)	—	—	SCK2 I/O (SCI)	—
	PA3 I/O (port)	—	—	—	—	—	RXD3 input (SCI)	—
	PA4 I/O (port)	—	—	—	—	—	TXD3 output (SCI)	—
	PA5 I/O (port)	—	—	IRQ1 input (INTC)	—	—	SCK3 I/O (SCI)	—
	PA6 I/O (port)	TCLKA input (MTU)	—	—	—	RXD2 input (SCI)	—	—
	PA7 I/O (port)	TCLKB input (MTU)	—	—	—	TXD2 output (SCI)	—	—
	PA8 I/O (port)	TCLKC input (MTU)	—	—	—	RXD3 input (SCI)	—	—
	PA9 I/O (port)	TCLKD input (MTU)	—	—	—	TXD3 output (SCI)	—	—
	PA10 I/O (port)	—	—	—	—	SCK2 I/O (SCI)	—	—
	PA11 I/O (port)	—	ADTRG input (A/D)	—	—	SCK3 I/O (SCI)	—	—
	PA12 I/O (port)	—	—	—	—	—	—	—
	PA13 I/O (port)	—	—	—	—	—	—	—

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)	Function 7 (Related Module)	Function 8 (Related Module)
A	PA14 I/O (port)	—	—	—	—	—	—	—
	PA15 I/O (port)	—	—	—	—	—	—	—

Table 13.2 Multiplexed Pins (Port B)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)
B	PB2 I/O (port)	$\overline{\text{IRQ0}}$ input (INTC)	$\overline{\text{POE0}}$ input (port)	—
	PB3 I/O (port)	$\overline{\text{IRQ1}}$ input (INTC)	$\overline{\text{POE1}}$ input (port)	—
	PB4 I/O (port)	$\overline{\text{IRQ2}}$ input (INTC)	$\overline{\text{POE2}}$ input (port)	—
	PB5 I/O (port)	$\overline{\text{IRQ3}}$ input (INTC)	$\overline{\text{POE3}}$ input (port)	—

Table 13.3 Multiplexed Pins (Port E)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)
E	PE0 I/O (port)	TIOC0A I/O (MTU)	—	—
	PE1 I/O (port)	TIOC0B I/O (MTU)	—	—
	PE2 I/O (port)	TIOC0C I/O (MTU)	—	—
	PE3 I/O (port)	TIOC0D I/O (MTU)	—	—
	PE4 I/O (port)	TIOC1A I/O (MTU)	RXD3 input (SCI)	—
	PE5 I/O (port)	TIOC1B I/O (MTU)	TXD3 output (SCI)	—
	PE6 I/O (port)	TIOC2A I/O (MTU)	SCK3 I/O (SCI)	—
	PE7 I/O (port)	TIOC2B I/O (MTU)	—	—
	PE8 I/O (port)	TIOC3A I/O (MTU)	—	—
	PE9 I/O (port)	TIOC3B I/O (MTU)	—	—
	PE10 I/O (port)	TIOC3C I/O (MTU)	—	—
	PE11 I/O (port)	TIOC3D I/O (MTU)	—	—
	PE12 I/O (port)	TIOC4A I/O (MTU)	—	—
	PE13 I/O (port)	TIOC4B I/O (MTU)	$\overline{\text{MRES}}$ input (INTC)	—
	PE14 I/O (port)	TIOC4C I/O (MTU)	—	—
	PE15 I/O (port)	TIOC4D I/O (MTU)	—	$\overline{\text{IRQOUT}}$ output (INTC)
	PE16 I/O (port)	—	—	—
	PE17 I/O (port)	—	—	—
	PE18 I/O (port)	—	—	—
	PE19 I/O (port)	—	—	—
	PE20 I/O (port)	—	—	—
	PE21 I/O (port)	—	—	—

Table 13.4 Multiplexed Pins (Port F)

Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)
F	PF8 input (port)	AN8 input (A/D0)	—	—
	PF9 input (port)	AN9 input (A/D0)	—	—
	PF10 input (port)	AN10 input (A/D0)	—	—
	PF11 input (port)	AN11 input (A/D0)	—	—
	PF12 input (port)	AN12 input (A/D1)	—	—
	PF13 input (port)	AN13 input (A/D1)	—	—
	PF14 input (port)	AN14 input (A/D1)	—	—
	PF15 input (port)	AN15 input (A/D1)	—	—

Table 13.5 Multiplexed Pins (Port G)

Port	Function 1 (Related Module)
G	PG0 input (port)
	PG1 input (port)
	PG2 input (port)
	PG3 input (port)

Table 13.6 Pin Functions in Each Operating Mode

	Pin Name	
	Single Chip Mode	
Pin No.	**Initial Function**	**PFC Selected Function Possibilities**
11, 43, 66	Vcc	Vcc
9, 24, 41, 64	Vss	Vss
22, 62	VCL	VCL
27, 38	AVcc	AVcc
25, 40	AVss	AVss
1	PE2	PE2/TIOC0C
2	PE3	PE3/TIOC0D
3	PE4	PE4/TIOC1A/RxD3
4	PE5	PE5/TIOC1B/TxD3
5	PE6	PE6/TIOC2A/SCK3
6	PE7	PE7/TIOC2B
7	PE8	PE8/TIOC3A
8	PE9	PE9/TIOC3B
10	PE10	PE10/TIOC3C
12	PE11	PE11/TIOC3D
13	PE12	PE12/TIOC4A
14	PE13	PE13/TIOC4B/$\overline{\text{MRES}}$
15	PE14	PE14/TIOC4C
16	PE15	PE15/TIOC4D/$\overline{\text{IRQOUT}}$
17	PE16	PE16
18	PE17	PE17
19	PE18	PE18
20	PE19	PE19
21	PE20	PE20
23	PE21	PE21
26	PF15/AN15	PF15/AN15
28	PF14/AN14	PF14/AN14
29	PF13/AN13	PF13/AN13
30	PF12/AN12	PF12/AN12
31	PG3	PG3

	Pin Name	
	Single Chip Mode	
Pin No.	Initial Function	PFC Selected Function Possibilities
32	PG2	PG2
33	PG1	PG1
34	PG0	PG0
35	PF11/AN11	PF11/AN11
36	PF10/AN10	PF10/AN10
37	PF9/AN9	PF9/AN9
39	PF8/AN8	PF8/AN8
42	PB5	PB5/$\overline{IRQ3}$/$\overline{POE3}$
44	PB4	PB4/$\overline{IRQ2}$/$\overline{POE2}$
45	PB3	PB3/$\overline{IRQ1}$/$\overline{POE1}$
46	PB2	PB2/$\overline{IRQ0}$/$\overline{POE0}$
47	PA15	PA15
48	PA14	PA14
49	PA13	PA13
50	PA12	PA12
51	PA11	PA11/$\overline{ADTRG}$/SCK3
52	PA10	PA10/SCK2
53	PA9	PA9/TCLKD/TXD3
54	PA8	PA8/TCLKC/RXD3
55	PA7	PA7/TCLKB/TXD2
56	PA6	PA6/TCLKA/RXD2
57	PA5	PA5/$\overline{IRQ1}$/SCK3
58	PA4	PA4/TXD3
59	PA3	PA3/RXD3
60	PA2	PA2/$\overline{IRQ0}$/SCK2
61	PA1	PA1/$\overline{POE1}$/TXD2
63	PA0	PA0/$\overline{POE0}$/RXD2
65	FWP	FWP
67	$\overline{RES}$	$\overline{RES}$
68	NMI	NMI
69	MD3	MD3

	Pin Name	
	Single Chip Mode	
Pin No.	Initial Function	PFC Selected Function Possibilities
70	MD2	MD2
71	MD1	MD1
72	MD0	MD0
73	EXTAL	EXTAL
74	XTAL	XTAL
75	PLLVCL	PLLVCL
76	PLLCAP	PLLCAP
77	PLLVss	PLLVss
78	$\overline{\text{WDTOVF}}$	$\overline{\text{WDTOVF}}$
79	PE0	PE0/TIOC0A
80	PE1	PE1/TIOC0B

Note: In single chip mode, do not set functions other than those that can be set by PFC listed in this table.

13.1 Register Descriptions

The PFC has the following registers. For details on the addresses of the registers and their states during each process, see section 18, List of Registers.

- Port A I/O register L (PAIORL)
- Port A control register L3 (PACRL3)
- Port A control register L2 (PACRL2)
- Port A control register L1 (PACRL1)
- Port B I/O register (PBIOR)
- Port B control register 1 (PBCR1)
- Port B control register 2 (PBCR2)
- Port E I/O register H (PEIORH)
- Port E I/O register L (PEIORL)
- Port E control register H (PECRH)
- Port E control register L1 (PECRL1)
- Port E control register L2 (PECRL2)

13.1.1 Port A I/O Register L (PAIORL)

PAIORL is a 16-bit readable/writable register that is used to set the pins on port A as inputs or outputs. Bits PA15IOR to PA0IOR correspond to pins PA15 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), and SCK2 and SCK3 pins are functioning as inputs/outputs of SCI. In other states, PAIORL is disabled.

A given pin on port A will be an output pin if the corresponding bit in PAIORL is set to 1, and an input pin if the bit is cleared to 0.

PAIORL is initialized to H'0000.

13.1.2 Port A Control Registers L3 to L1 (PACRL3 to PACRL1)

PACRL3 to PACRL1 are 16-bit readable/writable registers that are used to select the functions of the multiplexed pins on port A.

Port A Control Registers L3 to L1 (PACRL3 to PACRL1)

Register	Bit	Bit Name	Initial Value	R/W	Description	
PACRL3	15	PA15MD2	0	R/W	PA15 Mode	
PACRL1	15	PA15MD1	0	R/W	Select the function of the PA15 pin.	
PACRL1	14	PA15MD0	0	R/W	000: PA15 I/O (port)	011: Setting prohibited
					001: Setting prohibited	1xx: Setting prohibited
					010: Setting prohibited	
PACRL3	14	PA14MD2	0	R/W	PA14 Mode	
PACRL1	13	PA14MD1	0	R/W	Select the function of the PA14 pin.	
PACRL1	12	PA14MD0	0	R/W	000: PA14 I/O (port)	011: Setting prohibited
					001: Setting prohibited	1xx: Setting prohibited
					010: Setting prohibited	
PACRL3	13	PA13MD2	0	R/W	PA13 Mode	
PACRL1	11	PA13MD1	0	R/W	Select the function of the PA13 pin.	
PACRL1	10	PA13MD0	0	R/W	000: PA13 I/O (port)	011: Setting prohibited
					001: Setting prohibited	1xx: Setting prohibited
					010: Setting prohibited	
PACRL3	12	PA12MD2	0	R/W	PA12 Mode	
PACRL1	9	PA12MD1	0	R/W	Select the function of the PA12 pin.	
PACRL1	8	PA12MD0	0	R/W	000: PA12 I/O (port)	011: Setting prohibited
					001: Setting prohibited	1xx: Setting prohibited
					010: Setting prohibited	
PACRL3	11	PA11MD2	0	R/W	PA11 Mode	
PACRL1	7	PA11MD1	0	R/W	Select the function of the PA11/$\overline{\text{ADTRG}}$/SCK3 pin.	
PACRL1	6	PA11MD0	0	R/W	000: PA11 I/O (port)	100: Setting prohibited
					001: Setting prohibited	101: SCK3 I/O (SCI)
					010: $\overline{\text{ADTRG}}$ input (A/D)	110: Setting prohibited
					011: Setting prohibited	111: Setting prohibited

Register	Bit	Bit Name	Initial Value	R/W	Description	
PACRL3	10	PA10MD2	0	R/W	PA10 Mode	
PACRL1	5	PA10MD1	0	R/W	Select the function of the PA10/SCK2 pin.	
PACRL1	4	PA10MD0	0	R/W	000: PA10 I/O (port)	100: Setting prohibited
					001: Setting prohibited	101: SCK2 I/O (SCI)
					010: Setting prohibited	110: Setting prohibited
					011: Setting prohibited	111: Setting prohibited
PACRL3	9	PA9MD2	0	R/W	PA9 Mode	
PACRL1	3	PA9MD1	0	R/W	Select the function of the PA9/TCLKD/TXD3 pin.	
PACRL1	2	PA9MD0	0	R/W	000: PA9 I/O (port)	100: Setting prohibited
					001: TCLKD input (MTU)	101: TXD3 output (SCI)
					010: Setting prohibited	110: Setting prohibited
					011: Setting prohibited	111: Setting prohibited
PACRL3	8	PA8MD2	0	R/W	PA8 Mode	
PACRL1	1	PA8MD1	0	R/W	Select the function of the PA8/TCLKC/RXD3 pin.	
PACRL1	0	PA8MD0	0	R/W	000: PA8 I/O (port)	100: Setting prohibited
					001: TCLKC input (MTU)	101: RXD3 input (SCI)
					010: Setting prohibited	110: Setting prohibited
					011: Setting prohibited	111: Setting prohibited
PACRL3	7	PA7MD2	0	R/W	PA7 Mode	
PACRL2	15	PA7MD1	0	R/W	Select the function of the PA7/TCLKB/TXD2 pin.	
PACRL2	14	PA7MD0	0	R/W	000: PA7 I/O (port)	100: Setting prohibited
					001: TCLKB input (MTU)	101: TXD2 output (SCI)
					010: Setting prohibited	110: Setting prohibited
					011: Setting prohibited	111: Setting prohibited
PACRL3	6	PA6MD2	0	R/W	PA6 Mode	
PACRL2	13	PA6MD1	0	R/W	Select the function of the PA6/TCLKA/RXD2 pin.	
PACRL2	12	PA6MD0	0	R/W	000: PA6 I/O (port)	100: Setting prohibited
					001: TCLKA input (MTU)	101: RXD2 input (SCI)
					010: Setting prohibited	110: Setting prohibited
					011: Setting prohibited	111: Setting prohibited
PACRL3	5	PA5MD2	0	R/W	PA5 Mode	
PACRL2	11	PA5MD1	0	R/W	Select the function of the PA5/$\overline{\text{IRQ1}}$/SCK3 pin.	
PACRL2	10	PA5MD0	0	R/W	000: PA5 I/O (port)	100: Setting prohibited
					001: Setting prohibited	101: Setting prohibited
					010: Setting prohibited	110: SCK3 I/O (SCI)
					011: $\overline{\text{IRQ1}}$ input (INTC)	111: Setting prohibited

Register	Bit	Bit Name	Initial Value	R/W	Description	
PACRL3	4	PA4MD2	0	R/W	PA4 Mode	
PACRL2	9	PA4MD1	0	R/W	Select the function of the PA4/TXD3 pin.	
PACRL2	8	PA4MD0	0	R/W	000: PA4 I/O (port)	100: Setting prohibited
					001: Setting prohibited	101: Setting prohibited
					010: Setting prohibited	110: TXD3 output (SCI)
					011: Setting prohibited	111: Setting prohibited
PACRL3	3	PA3MD2	0	R/W	PA3 Mode	
PACRL2	7	PA3MD1	0	R/W	Select the function of the PA3/RXD3 pin.	
PACRL2	6	PA3MD0	0	R/W	000: PA3 I/O (port)	100: Setting prohibited
					001: Setting prohibited	101: Setting prohibited
					010: Setting prohibited	110: RXD3 input (SCI)
					011: Setting prohibited	111: Setting prohibited
PACRL3	2	PA2MD2	0	R/W	PA2 Mode	
PACRL2	5	PA2MD1	0	R/W	Select the function of the PA2/$\overline{\text{IRQ0}}$/SCK2 pin.	
PACRL2	4	PA2MD0	0	R/W	000: PA2 I/O (port)	100: Setting prohibited
					001: Setting prohibited	101: Setting prohibited
					010: Setting prohibited	110: SCK2 I/O (SCI)
					011: $\overline{\text{IRQ0}}$ input (INTC)	111: Setting prohibited
PACRL3	1	PA1MD2	0	R/W	PA1 Mode	
PACRL2	3	PA1MD1	0	R/W	Select the function of the PA1/$\overline{\text{POE1}}$/TXD2 pin.	
PACRL2	2	PA1MD0	0	R/W	000: PA1 I/O (port)	100: Setting prohibited
					001: Setting prohibited	101: $\overline{\text{POE1}}$ input (port)
					010: Setting prohibited	110: TXD2 output (SCI)
					011: Setting prohibited	111: Setting prohibited
PACRL3	0	PA0MD2	0	R/W	PA0 Mode	
PACRL2	1	PA0MD1	0	R/W	Select the function of the PA0/$\overline{\text{POE0}}$/RXD2 pin.	
PACRL2	0	PA0MD0	0	R/W	000: PA0 I/O (port)	100: Setting prohibited
					001: Setting prohibited	101: $\overline{\text{POE0}}$ input (port)
					010: Setting prohibited	110: RXD2 input (SCI)
					011: Setting prohibited	111: Setting prohibited

Note: x means “don't care”.

13.1.3 Port B I/O Register (PBIOR)

PBIOR is a 16-bit readable/writable register that is used to set the pins on port B as inputs or outputs. Bits PB5IOR to PB2IOR correspond to pins PB5 to PB2 (names of multiplexed pins are here given as port names and pin numbers alone). PBIOR is enabled when port B pins are functioning as general-purpose inputs/outputs (PB5 to PB2). In other states, PBIOR is disabled.

A given pin on port B will be an output pin if the corresponding bit in PBIOR is set to 1, and an input pin if the bit is cleared to 0.

Bits 15 to 6, 1, and 0 are reserved. These bits are always read as 0. The write value should always be 0.

PBIOR is initialized to H'0000.

13.1.4 Port B Control Registers 1 and 2 (PBCR1 and PBCR2)

PBCR1 and PBCR2 are 16-bit readable/writable registers that are used to select the multiplexed pin function of the pins on port B.

Port B Control Registers 1 and 2 (PBCR1 and PBCR2)

Register	Bit	Bit Name	Initial Value	R/W	Description
PBCR1	15, 14	—	All 0	R	Reserved
PBCR1	9 to 0	—	All 0	R	These bits are always read as 0. The write value should always be 0.
PBCR2	15 to 12	—	All 0	R	
PBCR2	3 to 0	—	All 0	R	
PBCR1	13	PB5MD2	0	R/W	PB5 Mode
PBCR2	11	PB5MD1	0	R/W	Select the function of the PB5/$\overline{\text{IRQ3}}$/$\overline{\text{POE3}}$ pin.
PBCR2	10	PB5MD0	0	R/W	000: PB5 I/O (port) 001: $\overline{\text{IRQ3}}$ input (INTC) 010: $\overline{\text{POE3}}$ input (port) 011: Setting prohibited 1xx: Setting prohibited
PBCR1	12	PB4MD2	0	R/W	PB4 Mode
PBCR2	9	PB4MD1	0	R/W	Select the function of the PB4/$\overline{\text{IRQ2}}$/$\overline{\text{POE2}}$ pin.
PBCR2	8	PB4MD0	0	R/W	000: PB4 I/O (port) 001: $\overline{\text{IRQ2}}$ input (INTC) 010: $\overline{\text{POE2}}$ input (port) 011: Setting prohibited 1xx: Setting prohibited
PBCR1	11	PB3MD2	0	R/W	PB3 Mode
PBCR2	7	PB3MD1	0	R/W	Select the function of the PB3/$\overline{\text{IRQ1}}$/$\overline{\text{POE1}}$ pin.
PBCR2	6	PB3MD0	0	R/W	000: PB3 I/O (port) 001: $\overline{\text{IRQ1}}$ input (INTC) 010: $\overline{\text{POE1}}$ input (port) 011: Setting prohibited 1xx: Setting prohibited
PBCR1	10	PB2MD2	0	R/W	PB2 Mode
PBCR2	5	PB2MD1	0	R/W	Select the function of the PB2/$\overline{\text{IRQ0}}$/$\overline{\text{POE0}}$ pin.
PBCR2	4	PB2MD0	0	R/W	000: PB2 I/O (port) 001: $\overline{\text{IRQ0}}$ input (INTC) 010: $\overline{\text{POE0}}$ input (port) 011: Setting prohibited 1xx: Setting prohibited

Note: x means “don't care”.

13.1.5 Port E I/O Registers L and H (PEIORL and PEIORH)

PEIORL and PEIORH are 16-bit readable/writable registers that are used to set the pins on port E as inputs or outputs. Bits 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), TIOC pins are functioning as inputs/outputs of MTU, and SCK3 pins are functioning as inputs/outputs of SCI. In other states, PEIORL is disabled. PEIORH is enabled when the port E pins are functioning as general-purpose inputs/outputs (PE21 to PE16). In other states, PEIORH is disabled.

A given pin on port E will be an output pin if the corresponding PEIORL or PEIORH bit is set to 1, and an input pin if the bit is cleared to 0.

Bits 15 to 6 in PEIORH are reserved. These bits are always read as 0. The write value should always be 0.

PEIORL and PEIORH are initialized to H'0000.

13.1.6 Port E Control Registers L1, L2, and H (PECRL1, PECRL2, and PECRH)

PECRL1, PECRL2 and PECRH are 16-bit readable/writable registers that are used to select the multiplexed pin function of the pins on port E.

Port E Control Registers L1, L2, and H (PECRL1, PECRL2, and PECRH)

Register	Bit	Bit Name	Initial Value	R/W	Description
PECRH	15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
PECRH	11	PE21MD1	0	R/W	PE21 Mode
PECRH	10	PE21MD0	0	R/W	Select the function of the PE21 pin. 00: PE21 I/O (port) 01: Setting prohibited 10: Setting prohibited 11: Setting prohibited
PECRH	9	PE20MD1	0	R/W	PE20 Mode
PECRH	8	PE20MD0	0	R/W	Select the function of the PE20 pin. 00: PE20 I/O (port) 01: Setting prohibited 10: Setting prohibited 11: Setting prohibited
PECRH	7	PE19MD1	0	R/W	PE19 Mode
PECRH	6	PE19MD0	0	R/W	Select the function of the PE19 pin. 00: PE19 I/O (port) 01: Setting prohibited 10: Setting prohibited 11: Setting prohibited
PECRH	5	PE18MD1	0	R/W	PE18 Mode
PECRH	4	PE18MD0	0	R/W	Select the function of the PE18 pin. 00: PE18 I/O (port) 01: Setting prohibited 10: Setting prohibited 11: Setting prohibited
PECRH	3	PE17MD1	0	R/W	PE17 Mode
PECRH	2	PE17MD0	0	R/W	Select the function of the PE17 pin. 00: PE17 I/O (port) 01: Setting prohibited 10: Setting prohibited 11: Setting prohibited
PECRH	1	PE16MD1	0	R/W	PE16 Mode
PECRH	0	PE16MD0	0	R/W	Select the function of the PE16 pin. 00: PE16 I/O (port) 01: Setting prohibited 10: Setting prohibited 11: Setting prohibited
PECRL1	15	PE15MD1	0	R/W	PE15 Mode
PECRL1	14	PE15MD0	0	R/W	Select the function of the PE15/TIOC4D/$\overline{\text{IRQOUT}}$ pin. 00: PE15 I/O (port) 01: TIOC4D I/O (MTU) 10: Setting prohibited 11: $\overline{\text{IRQOUT}}$ output (INTC)

Register	Bit	Bit Name	Initial Value	R/W	Description
PECRL1	13	PE14MD1	0	R/W	PE14 Mode
PECRL1	12	PE14MD0	0	R/W	Select the function of the PE14/TIOC4C pin. 00: PE14 I/O (port) 10: Setting prohibited 01: TIOC4C I/O (MTU) 11: Setting prohibited
PECRL1	11	PE13MD1	0	R/W	PE13 Mode
PECRL1	10	PE13MD0	0	R/W	Select the function of the PE13/TIOC4B/$\overline{\text{MRES}}$ pin. 00: PE13 I/O (port) 10: $\overline{\text{MRES}}$ input (INTC) 01: TIOC4B I/O (MTU) 11: Setting prohibited
PECRL1	9	PE12MD1	0	R/W	PE12 Mode
PECRL1	8	PE12MD0	0	R/W	Select the function of the PE12/TIOC4A pin. 00: PE12 I/O (port) 10: Setting prohibited 01: TIOC4A I/O (MTU) 11: Setting prohibited
PECRL1	7	PE11MD1	0	R/W	PE11 Mode
PECRL1	6	PE11MD0	0	R/W	Select the function of the PE11/TIOC3D pin. 00: PE11 I/O (port) 10: Setting prohibited 01: TIOC3D I/O (MTU) 11: Setting prohibited
PECRL1	5	PE10MD1	0	R/W	PE10 Mode
PECRL1	4	PE10MD0	0	R/W	Select the function of the PE10/TIOC3C pin. 00: PE10 I/O (port) 10: Setting prohibited 01: TIOC3C I/O (MTU) 11: Setting prohibited
PECRL1	3	PE9MD1	0	R/W	PE9 Mode
PECRL1	2	PE9MD0	0	R/W	Select the function of the PE9/TIOC3B pin. 00: PE9 I/O (port) 10: Setting prohibited 01: TIOC3B I/O (MTU) 11: Setting prohibited
PECRL1	1	PE8MD1	0	R/W	PE8 Mode
PECRL1	0	PE8MD0	0	R/W	Select the function of the PE8/TIOC3A pin. 00: PE8 I/O (port) 10: Setting prohibited 01: TIOC3A I/O (MTU) 11: Setting prohibited
PECRL2	15	PE7MD1	0	R/W	PE7 Mode
PECRL2	14	PE7MD0	0	R/W	Select the function of the PE7/TIOC2B pin. 00: PE7 I/O (port) 10: Setting prohibited 01: TIOC2B I/O (MTU) 11: Setting prohibited

Register	Bit	Bit Name	Initial Value	R/W	Description
PECRL2	13	PE6MD1	0	R/W	PE6 Mode
PECRL2	12	PE6MD0	0	R/W	Select the function of the PE6/TIOC2A/SCK3 pin.
					00: PE6 I/O (port) 10: SCK3 I/O (SCI)
					01: TIOC2A I/O (MTU) 11: Setting prohibited
PECRL2	11	PE5MD1	0	R/W	PE5 Mode
PECRL2	10	PE5MD0	0	R/W	Select the function of the PE5/TIOC1B/TXD3 pin.
					00: PE5 I/O (port) 10: TXD3 output (SCI)
					01: TIOC1B I/O (MTU) 11: Setting prohibited
PECRL2	9	PE4MD1	0	R/W	PE4 Mode
PECRL2	8	PE4MD0	0	R/W	Select the function of the PE4/TIOC1A/RXD3 pin.
					00: PE4 I/O (port) 10: RXD3 input (SCI)
					01: TIOC1A I/O (MTU) 11: Setting prohibited
PECRL2	7	PE3MD1	0	R/W	PE3 Mode
PECRL2	6	PE3MD0	0	R/W	Select the function of the PE3/TIOC0D pin.
					00: PE3 I/O (port) 10: Setting prohibited
					01: TIOC0D I/O (MTU) 11: Setting prohibited
PECRL2	5	PE2MD1	0	R/W	PE2 Mode
PECRL2	4	PE2MD0	0	R/W	Select the function of the PE2/TIOC0C pin.
					00: PE2 I/O (port) 10: Setting prohibited
					01: TIOC0C I/O (MTU) 11: Setting prohibited
PECRL2	3	PE1MD1	0	R/W	PE1 Mode
PECRL2	2	PE1MD0	0	R/W	Select the function of the PE1/TIOC0B pin.
					00: PE1 I/O (port) 10: Setting prohibited
					01: TIOC0B I/O (MTU) 11: Setting prohibited
PECRL2	1	PE0MD1	0	R/W	PE0 Mode
PECRL2	0	PE0MD0	0	R/W	Select the function of the PE0/TIOC0A pin.
					00: PE0 I/O (port) 10: Setting prohibited
					01: TIOC0A I/O (MTU) 11: Setting prohibited

13.2 Usage Notes

13.2.1 Note on PFC Setting

In this LSI, individual functions are available as multiplexed functions on multiple pins. This approach is intended to increase the number of selectable pin functions and to allow the easier design of boards.

When the pin function controller (PFC) is used to select a function, only a single pin can be specified for each function. If one function is specified for two or more pins, the function will not work properly.

13.2.2 Note on PFC Setting Order

When a pin function is selected, the port I/O registers (PAIORL and PBIORL) must be set after setting the port control registers (PACRL3, PACRL2, PACRL1, PBCR2, and PBCR1.

When a pin function which is multiplexed with the port E is selected, do not care about the setting order of the port control registers (PECRH, PECRL1, and PECRL2) and the port I/O registers (PEIORH and PEIORL).

Section 14 I/O Ports

This LSI has five ports: A, B, E, F, and G. Port A is a 16-bit port, port B is a 4-bit port, and port E is a 22-bit port, all supporting both input and output. Port F is an 8-bit port and port G is a 4-bit port, both for input-only.

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.

14.1 Port A

Port A is an input/output port with the 16 pins shown in figure 14.1.

PA15 to PA12 have an input-pull up MOS.

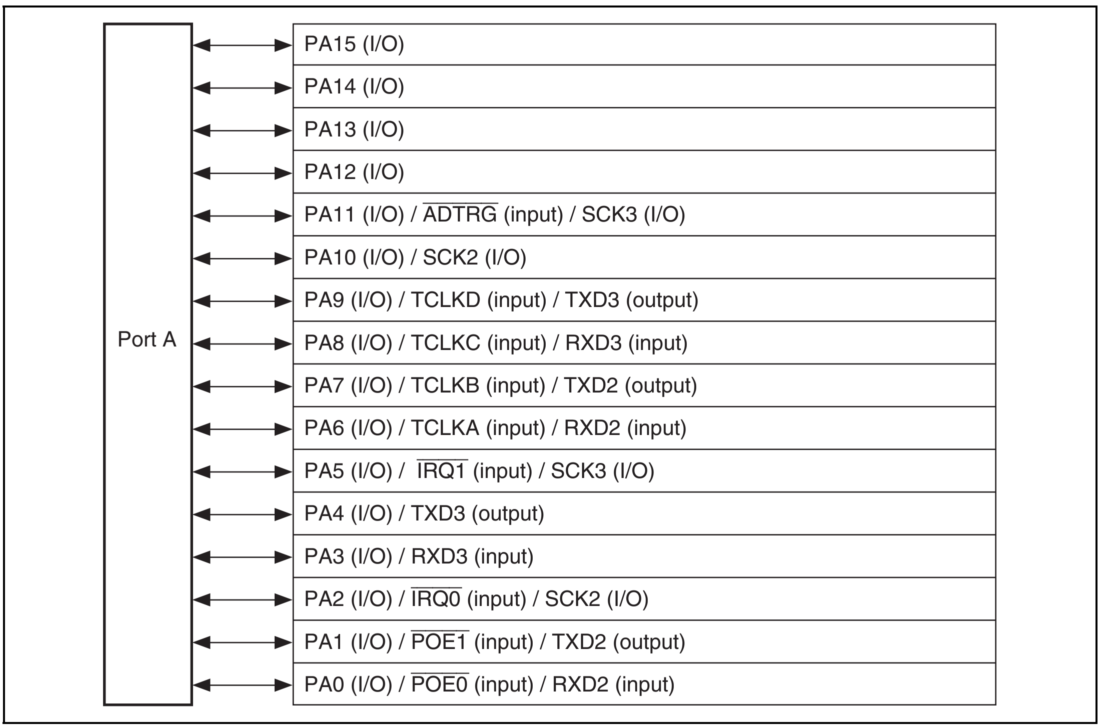


Figure 14.1 Port A

14.1.1 Register Description

Port A is a 16-bit input/output port. Port A has the following register. For details on register addresses and register states during each processing, refer to section 18, List of Registers.

- Port A data register L (PADRL)

14.1.2 Port A Data Register L (PADRL)

PADRL is a 16-bit readable/writable register that stores port A data. Bits PA15DR to PA0DR correspond to pins PA15 to PA0 (multiplexed functions omitted here).

When a pin functions is a general output, if a value is written to PADRL, that value is output directly from the pin, and if PADRL is read, the register value is returned directly regardless of the pin state.

When a pin functions is a general input, if PADRL is read, the pin state, not the register value, is returned directly. If a value is written to PADRL, although that value is written into PADRL, it does not affect the pin state. Table 14.1 summarizes port A data register L read/write operations.

Bit	Bit Name	Initial Value	R/W	Description
15	PA15DR	0	R/W	See table 14.1
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 14.1 Port A Data Register L (PADRL) Read/Write Operations

Bits 15 to 0:

PAIORL	Pin Function	Read	Write
0	General input	Pin state	Can write to PADRL, but it has no effect on pin state
	Other than general input	Pin state	Can write to PADRL, but it has no effect on pin state
1	General output	PADRL value	Value written is output from pin
	Other than general output	PADRL value	Can write to PADRL, but it has no effect on pin state

14.2 Port B

Port B is an input/output port with the four pins shown in figure 14.2.

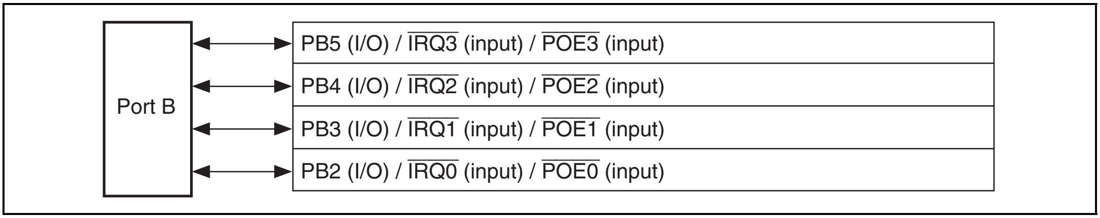


Figure 14.2 Port B

14.2.1 Register Description

Port B is a 4-bit input/output port. Port B has the following register. For details on register addresses and register states during each processing, refer to section 18, List of Registers.

- Port B data register (PBDR)

14.2.2 Port B Data Register (PBDR)

PBDR is a 16-bit readable/writable register that stores port B data. Bits PB5DR to PB2DR correspond to pins PB5 to PB2 (multiplexed functions omitted here).

When a pin functions is a general output, if a value is written to PBDR, that value is output directly from the pin, and if PBDR is read, the register value is returned directly regardless of the pin state.

When a pin functions is a general input, if PBDR is read, the pin state, not the register value, is returned directly. If a value is written to PBDR, although that value is written into PBDR, it does not affect the pin state. Table 14.2 summarizes port B data register read/write operations.

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	PB5DR	0	R/W	See table 14.2
4	PB4DR	0	R/W	
3	PB3DR	0	R/W	
2	PB2DR	0	R/W	
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Table 14.2 Port B Data Register (PBDR) Read/Write Operations

Bits 5 to 2:

PBIOR	Pin Function	Read	Write
0	General input	Pin state	Can write to PBDR, but it has no effect on pin state
	Other than general input	Pin state	Can write to PBDR, but it has no effect on pin state
1	General output	PBDR value	Value written is output from pin
	Other than general output	PBDR value	Can write to PBDR, but it has no effect on pin state

14.3 Port E

Port E is an input/output port with the 22 pins shown in figure 14.3.

PE21 to PE16 have an input-pull up MOS.

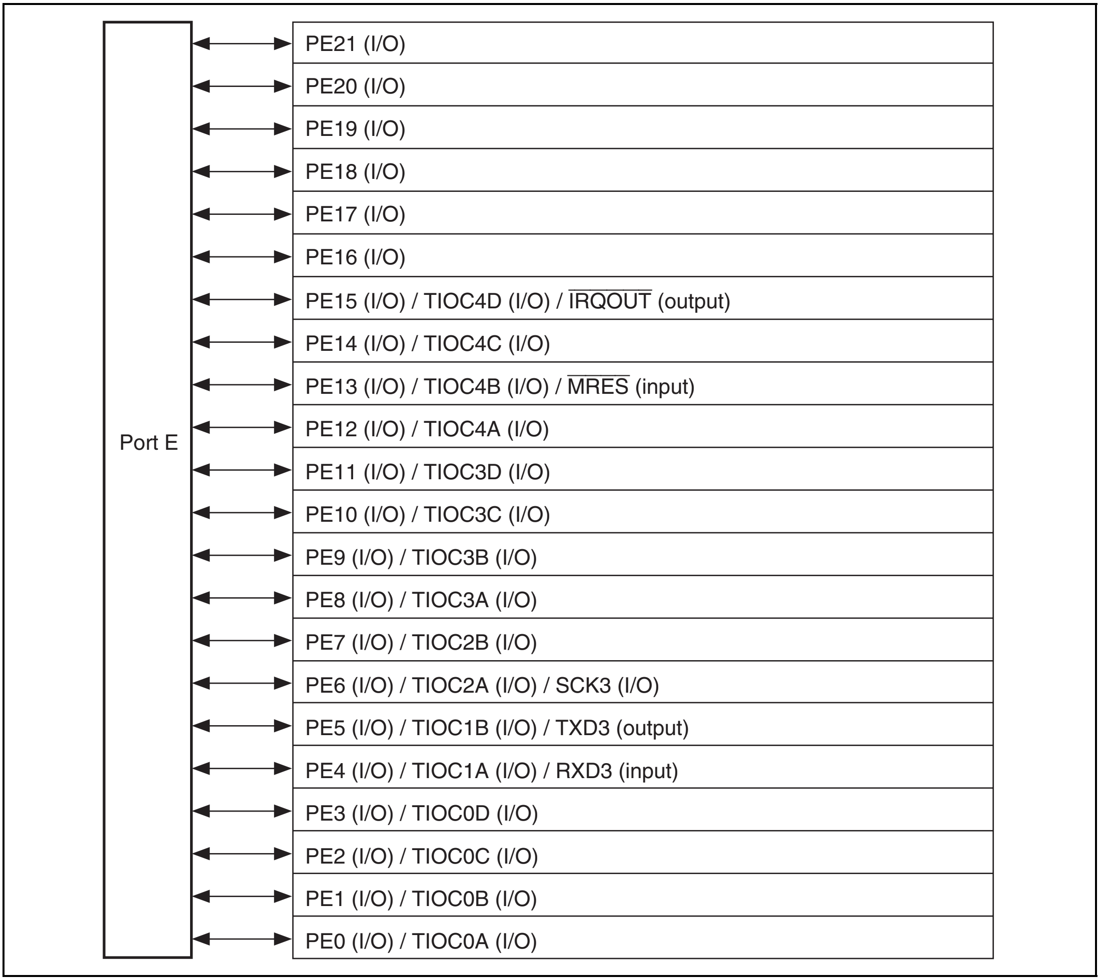


Figure 14.3 Port E

14.3.1 Register Descriptions

Port E has the following registers. For details on register addresses and register states during each processing, refer to section 18, List of Registers.

- Port E data register H (PEDRH)
- Port E data register L (PEDRL)

14.3.2 Port E Data Registers H and L (PEDRH and PEDRL)

PEDRH and PEDRL are 16-bit readable/writable registers that store port E data. Bits PE21DR to PE0DR correspond to pins PE21 to PE0 (multiplexed functions omitted here).

When a pin functions is a 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 functions is a 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 14.3 summarizes port E data register read/write operations.

PEDRH:

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 14.3.
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:

Bit	Bit Name	Initial Value	R/W	Description
15	PE15DR	0	R/W	See table 14.3.
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 14.3 Port E Data Registers H and L (PEDRH and PEDRL) Read/Write Operations

Bits 5 to 0 in PEDRH and bits 15 to 0 in PEDRL:

PEIOR	Pin Function	Read	Write
0	General input	Pin state	Can write to PEDRH or PEDRL, but it has no effect on pin state
	Other than general input	Pin state	Can write to PEDRH or PEDRL, but it has no effect on pin state
1	General output	PEDRH or PEDRL value	Value written is output from pin ($\overline{POE}$ pin = high)*
			High impedance regardless of PEDRH or PEDRL value ($\overline{POE}$ pin = low)*
	Other than general output	PEDRH or PEDRL value	Can write to PEDRH or PEDRL, but it has no effect on pin state

Note: * Control by the $\overline{POE}$ pin is only available for large current-output pins (PE9 and PE11 to PE15).

14.4 Port F

Port F is an input-only port with the eight pins shown in figure 14.4.

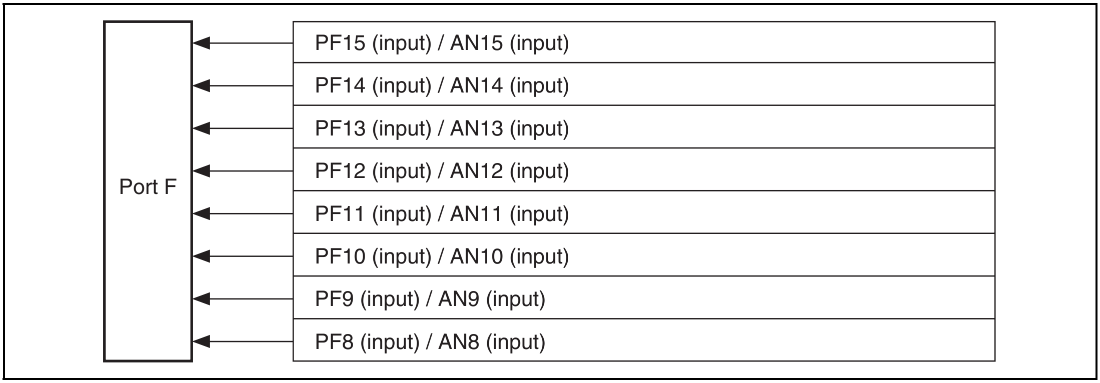


Figure 14.4 Port F

14.4.1 Register Description

Port F is an 8-bit input-only port. Port F has the following register. For details on register addresses and register states during each processing, refer to section 18, List of Registers.

- Port F data register (PFDR)

14.4.2 Port F Data Register (PFDR)

PFDR is a 16-bit read-only register that stores port F data.

Bits PF15DR to PF8DR correspond to pins PF15 to PF8 (multiplexed functions omitted here).

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 14.4 summarizes port F data register read operation.

Bit	Bit Name	Initial Value	R/W	Description
15	PF15DR	0/1*	R	See table 14.4.
14	PF14DR	0/1*	R	
13	PF13DR	0/1*	R	
12	PF12DR	0/1*	R	
11	PF11DR	0/1*	R	
10	PF10DR	0/1*	R	
9	PF9DR	0/1*	R	
8	PF8DR	0/1*	R	
7 to 0	—	All 0	R	Reserved These bits are always read as 0.

Note: * Initial values are dependent on the state of the external pins.

Table 14.4 Port F Data Register (PFDR) Read/Write Operations

Bits 15 to 8:

Pin Function	Read	Write
General input	Pin state	Ignored (no effect on pin state)
ANn input	1	Ignored (no effect on pin state)

14.5 Port G

Port G is an input-only port with the four pins shown in figure 14.5.

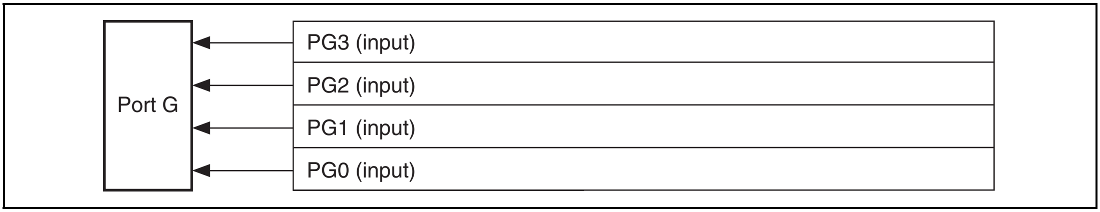


Figure 14.5 Port G

14.5.1 Register Description

Port G is a 4-bit input-only port. Port G has the following register. For details on register addresses and register states during each processing, refer to section 18, List of Registers.

- Port G data register (PGDR)

14.5.2 Port G Data Register (PGDR)

PGDR is an 8-bit read-only register that stores port G data.

Bits PG3DR to PG0DR correspond to pins PG3 to PG0.

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. Table 14.5 summarizes port G data register read operation.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0.
3	PG3DR	0/1*	R	See table 14.5.
2	PG2DR	0/1*	R	
1	PG1DR	0/1*	R	
0	PG0DR	0/1*	R	

Note: * Initial values are dependent on the state of the external pins.

Table 14.5 Port G Data Register (PGDR) Read/Write Operations

Bits 3 to 0:

Pin Function	Read	Write
General input	Pin state	Ignored (no effect on pin state)

Section 15 Mask ROM

This LSI is available with 32 kbytes of on-chip mask ROM. The on-chip ROM is connected to the CPU through a 32-bit data bus (figure 15.1). The CPU 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.

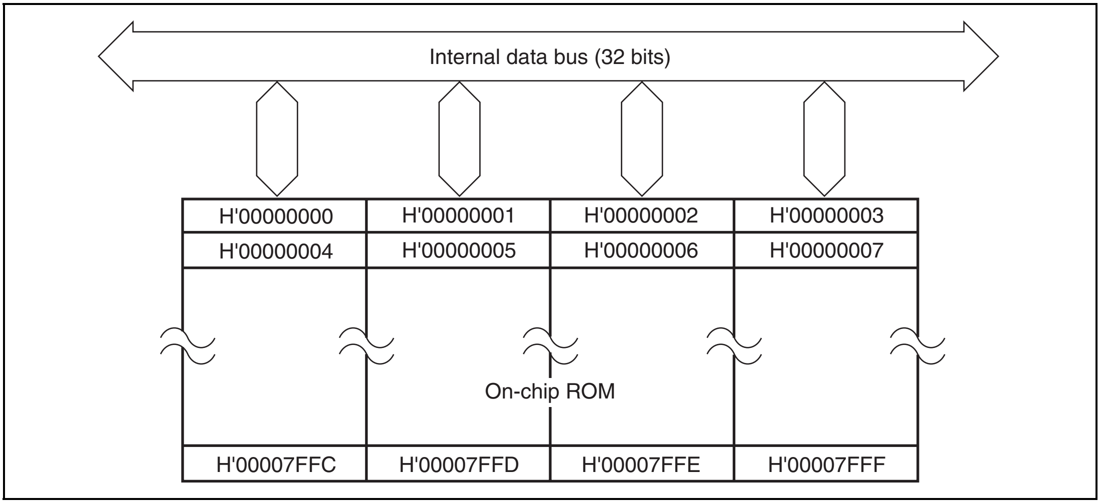


Figure 15.1 Mask ROM Block Diagram

The on-chip ROM is allocated to addresses H'00000000 to H'00007FFF.

15.1 Usage Note

- Setting module standby mode

  For mask ROM, this module can be disabled/enabled by the module standby control register. Mask ROM operation is enabled for the initial value. Accessing mask ROM is disabled by setting module standby mode. For details, see section 17, Power-Down Modes.

Section 16 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, enabling 8, 16, or 32-bit width access to data in the on-chip RAM. Data in the on-chip RAM can always be accessed in one cycle, providing high-speed access that makes this RAM ideal for use as a program area, stack area, or data area. The contents of the on-chip RAM are retained in both sleep and standby modes.

The on-chip RAM can be enabled or disabled by means of the RAME bit in the system control register (SYSCR). For details on the system control register (SYSCR), refer to section 17.2.2, System Control Register (SYSCR).

Product Type	Type of ROM	RAM Capacity	RAM Address
SH7101	Mask ROM	2 kbytes	H'FFFFF800 to H'FFFFFFFF

16.1 Usage Note

- Module Standby Mode Setting

  RAM can be enabled/disabled by the module standby control register. The initial value enables RAM operation. RAM access is disabled by setting the module standby mode. For details, see section 17, Power-Down Modes.

Section 17 Power-Down Modes

In addition to the normal program execution state, this LSI has three power-down modes in which operation of the CPU and oscillator is halted and power dissipation is reduced. Low-power operation can be achieved by individually controlling the CPU, on-chip peripheral functions, and so on.

This LSI's power-down modes are as follows:

(1) Sleep mode
(2) Software standby mode
(3) Module standby mode

Sleep mode indicates the state of the CPU, and module standby mode indicates the state of the on-chip peripheral function. Some of these states can be combined.

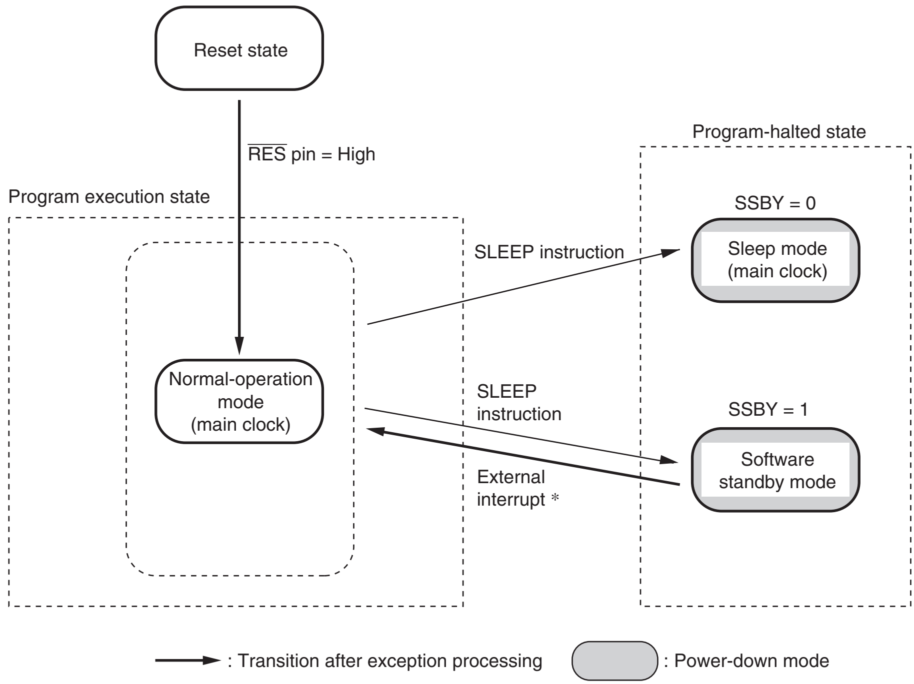
After a reset, the LSI is in normal-operation mode.

Table 17.1 lists internal operation states in each mode.

Table 17.1 Internal Operation States in Each Mode

Function		Normal operation	Sleep	Module Standby	Software Standby
System clock pulse generator		Functioning	Functioning	Functioning	Halted
CPU	Instructions	Functioning	Halted (retained)	Functioning	Halted (retained)
	Registers				
External interrupts	NMI	Functioning	Functioning	Functioning	Functioning
	$\overline{IRQ3}$ to $\overline{IRQ0}$				
Peripheral functions	I/O port	Functioning	Functioning	Functioning	Retained
	WDT	Functioning	Functioning	Functioning	Halted (retained)
	SCI	Functioning	Functioning	Halted (reset)	Halted (reset)
	A/D				
	MTU				
	CMT				
	ROM				
	RAM	Functioning	Functioning	Retained	Retained

Notes: 1. "Halted (retained)" means that the operation of the internal state is suspended, although internal register values are retained.
2. "Halted (reset)" means that internal register values and internal state are initialized.
3. In module standby mode, only modules for which a stop setting has been made are halted (reset or retained).
4. There are two types of on-chip peripheral module registers; ones which are initialized in software standby mode and module standby mode, and those not initialized those modes. For details, refer to section 18.3, Register States in Each Operating Mode.
5. The port high-impedance bit (HIZ) in SBYCR sets the state of the I/O port in software standby mode. For details on the setting, refer to section 17.2.1, Standby Control Register (SBYCR). For the state of pins, refer to appendix A, Pin States.



Notes: * NMI and IRQ
When a transition is made between modes by means of an interrupt, the transition cannot be made on interrupt source generation alone. Ensure that interrupt handling is performed after accepting the interrupt request.
In any state, a transition to the reset state occurs when $\overline{RES}$ is driven low.

Figure 17.1 Mode Transition Diagram

17.1 Input/Output Pins

Table 17.2 lists the pins relating to power-down mode.

Table 17.2 Pin Configuration

Pin Name	I/O	Function
$\overline{RES}$	Input	Power-on reset input pin
$\overline{MRES}$	Input	Manual reset input pin

17.2 Register Descriptions

Registers related to power down modes are shown below. For details on register addresses and register states during each process, refer to section 18, List of Registers.

- Standby control register (SBYCR)
- System control register (SYSCR)
- Module standby control register 1 (MSTCR1)
- Module standby control register 2 (MSTCR2)

17.2.1 Standby Control Register (SBYCR)

SBYCR is an 8-bit readable/writable register that performs software standby mode control.

Bit	Bit Name	Initial Value	R/W	Description
7	SSBY	0	R/W	Software Standby This bit specifies the transition mode after executing the SLEEP instruction. 0: Shifts to sleep mode after the SLEEP instruction has been executed 1: Shifts to software standby mode after the SLEEP instruction has been executed This bit cannot be set to 1 when the watchdog timer (WDT) is operating (when the TME bit in TCSR of the WDT is set to 1). When transferring to software standby mode, clear the TME bit to 0, stop the WDT, then set the SSBY bit to 1.
6	HIZ	0	R/W	Port High-Impedance In software standby mode, this bit selects whether the pin state of the I/O port 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. The HIZ bit cannot be set to 1 when the TME bit in TCSR of the WDT is set to 1. When changing the pin state of the I/O port to high-impedance, clear the TME bit to 0, then set the HIZ bit to 1.

Bit	Bit Name	Initial Value	R/W	Description
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4 to 1	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
0	IRQEL	1	R/W	IRQ3 to IRQ0 Enable IRQ interrupts are enabled to clear software standby mode. 0: Software standby mode is cleared. 1: Software standby mode is not cleared.

17.2.2 System Control Register (SYSCR)

SYSCR is an 8-bit readable/writable register that enables/disables the access to the on-chip RAM.

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 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	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 to SYSCR. 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 SYSCR should be set next to the instruction to write to SYSCR. If an instruction to access the on-chip RAM is set next to the instruction to write to SYSCR, normal access is not guaranteed.

17.2.3 Module Standby Control Register 1 and 2 (MSTCR1 and MSTCR2)

MSTCR, comprising two 16-bit readable/writable registers, performs module standby mode control. Setting a bit to 1, the corresponding module enters module standby mode, while clearing the bit to 0 clears the module standby mode.

MSTCR1

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
11	MSTP27	0	R/W	On-chip RAM
10	MSTP26	0	R/W	On-chip ROM
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
4	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
3	MSTP19	1	R/W	Serial communication interface 3 (SCI_3)
2	MSTP18	1	R/W	Serial communication interface 2 (SCI_2)
1, 0	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.

MSTCR2

Bit	Bit Name	Initial Value	R/W	Description
15	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
14	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
13	MSTP13	1	R/W	Multi-function timer pulse unit (MTU)
12	MSTP12	1	R/W	Compare match timer (CMT)
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
7	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
6	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
5	MSTP5	1	R/W	A/D converter (A/D1)
4	MSTP4	1	R/W	A/D converter (A/D0)
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

17.3 Operation

17.3.1 Sleep Mode

Transition to Sleep Mode: If SLEEP instruction is executed while the SSBY bit in SBYCR = 0, the CPU enters sleep mode. In sleep mode, CPU operation stops, however the contents of the CPU's internal registers are retained. Peripheral functions except the CPU do not stop.

Clearing Sleep Mode: Sleep mode is cleared by the conditions below.

- Clearing by the power on reset
  When the $\overline{\text{RES}}$ pin is driven low, the CPU enters the reset state. When the $\overline{\text{RES}}$ pin is driven high after the elapse of the specified reset input period, the CPU starts the reset exception handling. Also, when the internal power on reset is occurred, sleep mode is cleared.
- Clearing by the manual reset
  When the $\overline{\text{MRES}}$ pin is driven low while the $\overline{\text{RES}}$ pin is high, the CPU shifts to the manual reset state and thus sleep mode is cleared. Also, when the internal manual reset is occurred, sleep mode is cleared.

Notes on Using Sleep Mode

- There are 4 conditions to clear sleep mode.
  (1) Clearing by an interrupt
  (2) Clearing by DTC address error
  (3) Clearing by the power-on reset
  (4) Clearing by the manual reset

  When clearing sleep mode by (1) or (2), CPU may run out of control. Please clear sleep mode by (3) or (4), don't use (1) or (2).
- Do not use DTC module or AUD module during sleep mode.

17.3.2 Software Standby Mode

Transition to Software Standby Mode: A transition is made to software standby mode if the SLEEP instruction is executed while the SSBY bit in SBYCR is set to 1. In this mode, the CPU, on-chip peripheral functions, and the oscillator, all stop.

However, the contents of the CPU's internal registers and on-chip RAM data (when the RAME bit in SYSCR is 0) are retained as long as the specified voltage is supplied. There are two types of on-chip peripheral module registers; ones which are initialized by software standby mode, and those not initialized by that mode. For details, refer to section 18.3, Register States in Each Operating Mode. The port high-impedance bit (HIZ) in SBYCR sets the state of the I/O port either to "retained" or "high-impedance". For the state of pins, refer to appendix A, Pin States. In software standby mode, the oscillator stops and thus power consumption is significantly reduced.

Clearing Software Standby Mode: Software standby mode is cleared by the condition below.

- Clearing by the NMI interrupt input

  When the falling edge or rising edge of the NMI pin (selected by the NMI edge select bit (NMIE) in ICR1 of the interrupt controller (INTC)) is detected, clock oscillation is started. This clock pulse is supplied only to the watchdog timer (WDT).

  After the elapse of the time set in the clock select bits (CKS2 to CKS0) in TCSR of the WDT before the transition to software standby mode, the WDT overflow occurs. Since this overflow indicates that the clock has been stabilized, clock pulse will be supplied to the entire chip after this overflow. Software standby mode is thus cleared and the NMI exception handling is started.

  When clearing software standby mode by the NMI interrupt, set CKS2 to CKS0 bits so that the WDT overflow period will be longer than the oscillation stabilization time.

  When software standby mode is cleared by the falling edge of the NMI pin, the NMI pin should be high when the CPU enters software standby mode (when the clock pulse stops) and should be low when the CPU returns from standby mode (when the clock is initiated after the oscillation stabilization). When software standby mode is cleared by the rising edge of the NMI pin, the NMI pin should be low when the CPU enters software standby mode (when the clock pulse stops) and should be high when the CPU returns from software standby mode (when the clock is initiated after the oscillation stabilization).

- Clearing by the $\overline{\text{RES}}$ pin

  When the $\overline{\text{RES}}$ pin is driven low, clock oscillation is started. At the same time as clock oscillation is started, clock pulse is supplied to the entire chip. Ensure that the $\overline{\text{RES}}$ pin is held low until clock oscillation stabilizes. When the $\overline{\text{RES}}$ pin is driven high, the CPU starts the reset exception handling.

- Clearing by the IRQ interrupt input

  When the IRQEL bit in the standby control register (SBYCR) is set to 1 and when the falling edge or rising edge of the IRQ pin (selected by the IRQ3S to IRQ0S bits in ICR1 of the interrupt controller (INTC) and the IRQ3ES [1:0] to IRQ0ES [1:0] bits in ICR2) is detected, clock oscillation is started.* This clock pulse is supplied only to the watchdog timer (WDT). The IRQ interrupt priority level should be higher than the interrupt mask level set in the status register (SR) of the CPU before the transition to software standby mode.

  After the elapse of the time set in the clock select bits (CKS2 to CKS0) in TCSR of the WDT before the transition to software standby mode, the WDT overflow occurs. Since this overflow indicates that the clock has been stabilized, clock pulse will be supplied to the entire chip after this overflow. Software standby mode is thus cleared and the IRQ exception handling is started.

  When clearing software standby mode by the IRQ interrupt, set CKS2 to CKS0 bits so that the WDT overflow period will be longer than the oscillation stabilization time.

  When software standby mode is cleared by the falling edge or both edges of the $\overline{\text{IRQ}}$ pin, the $\overline{\text{IRQ}}$ pin should be high when the CPU enters software standby mode (when the clock pulse stops) and should be low when the CPU returns from software standby mode (when the clock is initiated after the oscillation stabilization). When software standby mode is cleared by the rising edge of the $\overline{\text{IRQ}}$ pin, the $\overline{\text{IRQ}}$ pin should be low when the CPU enters software standby mode (when the clock pulse stops) and should be high when the CPU returns from software standby mode (when the clock is initiated after the oscillation stabilization).

Note: * When the $\overline{\text{IRQ}}$ pin is set to falling-edge detection or both-edge detection, clock oscillation starts at falling-edge detection. When the $\overline{\text{IRQ}}$ pin is set to rising-edge detection, clock oscillation starts at rising-edge detection. Do not set the $\overline{\text{IRQ}}$ pin to low-level detection.

Software Standby Mode Application Example: Figure 17.2 shows an example in which a transition is made to software standby mode at the falling edge of the NMI pin, and software standby mode is cleared at a rising edge of the NMI pin.

In this example, when the NMI pin is driven low while the NMI edge select bit (NMIE) in ICR1 is 0 (falling edge detection), an NMI interrupt is accepted. Then, the NMIE bit is set to 1 (rising edge detection) in the NMI exception service routine, the SSBY bit in SBYCR is set to 1, and a SLEEP instruction is executed to transfer to software standby mode.

Software standby mode is cleared by driving the NMI pin from low to high.

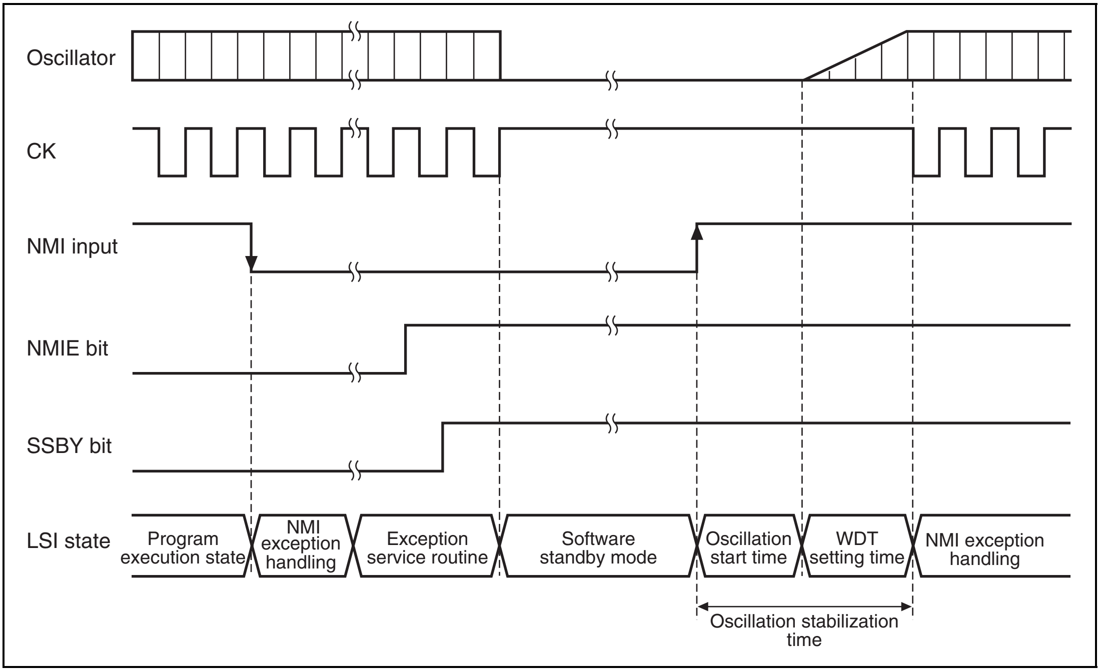


Figure 17.2 NMI Timing in Software Standby Mode

17.3.3 Module Standby Mode

Module standby mode can be set for individual on-chip peripheral functions.

When the corresponding MSTP bit in MSTCR is set to 1, module operation stops at the end of the bus cycle and a transition is made to module standby mode. The CPU continues operating independently.

When the corresponding MSTP bit is cleared to 0, module standby mode is cleared and the module starts operating at the end of the bus cycle. In module standby mode, the internal states of modules are initialized.

After reset clearing, the SCI, MTU, CMT, and A/D converter are in module standby mode.

When an on-chip peripheral module is in module standby mode, read/write access to its registers is disabled.

17.4 Usage Notes

17.4.1 I/O Port Status

When a transition is mode to software standby mode while the port high-impedance bit (HIZ) in SBYCR is 0, I/O port states are retained. Therefore, there is no reduction in current consumption for the output current when a high-level signal is output.

17.4.2 Current Consumption during Oscillation Stabilization Wait Period

Current consumption increases during the oscillation stabilization wait period.

17.4.3 On-Chip Peripheral Module Interrupt

Relevant interrupt operations cannot be performed in module standby mode. Consequently, if the CPU enters module standby mode while an interrupt has been requested, it will not be possible to clear the CPU interrupt source.

Interrupts should therefore be disabled before entering module standby mode.

17.4.4 Writing to MSTCR1 and MSTCR2

MSTCR1 and MSTCR2 should only be written to by the CPU.

Section 18 List of Registers

The column "Access Size" shows the number of bits.

The column "Access States" shows the number of access states, in units of cycles, of the specified reference clock. B, W, and L in the column represent 8-bit, 16-bit, and 32-bit access, respectively.

18.1 Register Addresses (Order of Address)

Register Name	Abbreviation	Bits	Address	Module	Access Size	Access States
—	—	—	H'FFFF8000 to H'FFFF81BF	—	—	—
Serial mode register_2	SMR_2	8	H'FFFF81C0	SCI (channel 2)	8, 16	In Pϕ cycles B: 2 W: 4
Bit rate register_2	BRR_2	8	H'FFFF81C1		8	
Serial control register_2	SCR_2	8	H'FFFF81C2		8, 16	
Transmit data register_2	TDR_2	8	H'FFFF81C3		8	
Serial status register_2	SSR_2	8	H'FFFF81C4		8, 16	
Receive data register_2	RDR_2	8	H'FFFF81C5		8	
Serial direction control register_2	SDCR_2	8	H'FFFF81C6		8	
—	—	—	H'FFFF81C7 to H'FFFF81CF	—		
Serial mode register_3	SMR_3	8	H'FFFF81D0	SCI (channel 3)	8, 16	
Bit rate register_3	BRR_3	8	H'FFFF81D1		8	
Serial control register_3	SCR_3	8	H'FFFF81D2		8, 16	
Transmit data register_3	TDR_3	8	H'FFFF81D3		8	
Serial status register_3	SSR_3	8	H'FFFF81D4		8, 16	
Receive data register_3	RDR_3	8	H'FFFF81D5		8	
Serial direction control register_3	SDCR_3	8	H'FFFF81D6		8	
—	—	—	H'FFFF81D7 to H'FFFF81FF	—		
Timer control register_3	TCR_3	8	H'FFFF8200	MTU (channels 3 and 4)	8, 16, 32	In Pϕ cycles B: 2 W: 2 L: 4
Timer control register_4	TCR_4	8	H'FFFF8201		8	
Timer mode register_3	TMDR_3	8	H'FFFF8202		8, 16	
Timer mode register_4	TMDR_4	8	H'FFFF8203		8	
Timer I/O control register H_3	TIORH_3	8	H'FFFF8204		8, 16, 32	
Timer I/O control register L_3	TIORL_3	8	H'FFFF8205		8	

Register Name	Abbreviation	Bits	Address	Module	Access Size	Access States
Timer I/O control register H_4	TIORH_4	8	H'FFFF8206	MTU (channels 3 and 4)	8, 16	In Pϕ cycles B: 2 W: 2 L: 4
Timer I/O control register L_4	TIORL_4	8	H'FFFF8207		8	
Timer interrupt enable register_3	TIER_3	8	H'FFFF8208		8, 16, 32	
Timer interrupt enable register_4	TIER_4	8	H'FFFF8209		8	
Timer output master enable register	TOER	8	H'FFFF820A		8, 16	
Timer output control register	TOCR	8	H'FFFF820B		8	
—	—	—	H'FFFF820C			
Timer gate control register	TGCR	8	H'FFFF820D		8	
—	—	—	H'FFFF820E			
—	—	—	H'FFFF820F			
Timer counter_3	TCNT_3	16	H'FFFF8210		16, 32	
Timer counter_4	TCNT_4	16	H'FFFF8212		16	
Timer period data register	TCDR	16	H'FFFF8214		16, 32	
Timer dead time data register	TDDR	16	H'FFFF8216		16	
Timer general register A_3	TGRA_3	16	H'FFFF8218		16, 32	
Timer general register B_3	TGRB_3	16	H'FFFF821A		16	
Timer general register A_4	TGRA_4	16	H'FFFF821C		16, 32	
Timer general register B_4	TGRB_4	16	H'FFFF821E		16	
Timer sub-counter	TCNTS	16	H'FFFF8220		16, 32	
Timer period buffer register	TCBR	16	H'FFFF8222		16	
Timer general register C_3	TGRC_3	16	H'FFFF8224		16, 32	
Timer general register D_3	TGRD_3	16	H'FFFF8226		16	
Timer general register C_4	TGRC_4	16	H'FFFF8228		16, 32	
Timer general register D_4	TGRD_4	16	H'FFFF822A		16	
Timer status register_3	TSR_3	8	H'FFFF822C		8, 16	
Timer status register_4	TSR_4	8	H'FFFF822D		8	
—	—	—	H'FFFF822E to H'FFFF823F			
Timer start register	TSTR	8	H'FFFF8240	MTU (common)	8, 16	In Pϕ cycles B: 2 W: 2
Timer synchro register	TSYR	8	H'FFFF8241		8	
—	—	—	H'FFFF8242 to H'FFFF825F			

Register Name	Abbreviation	Bits	Address	Module	Access Size	Access States
Timer control register_0	TCR_0	8	H'FFFF8260	MTU (channel 0)	8, 16, 32	In Pϕ cycles B: 2 W: 2 L: 4
Timer mode register_0	TMDR_0	8	H'FFFF8261		8	
Timer I/O control register H_0	TIORH_0	8	H'FFFF8262		8, 16	
Timer I/O control register L_0	TIORL_0	8	H'FFFF8263		8	
Timer interrupt enable register_0	TIER_0	8	H'FFFF8264		8, 16, 32	
Timer status register_0	TSR_0	8	H'FFFF8265		8	
Timer counter_0	TCNT_0	16	H'FFFF8266		16	
Timer general register A_0	TGRA_0	16	H'FFFF8268		16, 32	
Timer general register B_0	TGRB_0	16	H'FFFF826A		16	
Timer general register C_0	TGRC_0	16	H'FFFF826C		16, 32	
Timer general register D_0	TGRD_0	16	H'FFFF826E		16	
—	—	—	H'FFFF8270 to H'FFFF827F		—	
Timer control register_1	TCR_1	8	H'FFFF8280	MTU (channel 1)	8, 16	
Timer mode register_1	TMDR_1	8	H'FFFF8281		8	
Timer I/O control register_1	TIOR_1	8	H'FFFF8282		8	
—	—	—	H'FFFF8283		—	
Timer interrupt enable register_1	TIER_1	8	H'FFFF8284		8, 16, 32	
Timer status register_1	TSR_1	8	H'FFFF8285		8	
Timer counter_1	TCNT_1	16	H'FFFF8286		16	
Timer general register A_1	TGRA_1	16	H'FFFF8288		16, 32	
Timer general register B_1	TGRB_1	16	H'FFFF828A		16	
—	—	—	H'FFFF828C to H'FFFF829F		—	
Timer control register_2	TCR_2	8	H'FFFF82A0	MTU (channel 2)	8, 16	
Timer mode register_2	TMDR_2	8	H'FFFF82A1		8	
Timer I/O control register_2	TIOR_2	8	H'FFFF82A2		8	
—	—	—	H'FFFF82A3		—	
Timer interrupt enable register_2	TIER_2	8	H'FFFF82A4		8, 16, 32	
Timer status register_2	TSR_2	8	H'FFFF82A5		8	
Timer counter_2	TCNT_2	16	H'FFFF82A6		16	

Register Name	Abbreviation	Bits	Address	Module	Access Size	Access States
Timer general register A_2	TGRA_2	16	H'FFFF82A8	MTU (channel 2)	16, 32	In Pϕ cycles B: 2 W: 2 L: 4
Timer general register B_2	TGRB_2	16	H'FFFF82AA		16	
—	—	—	H'FFFF82AC to H'FFFF833F		—	
—	—	—	H'FFFF8340 to H'FFFF8347	INTC	—	In ϕ cycles B: 2 W: 2 L: 4
Interrupt priority register A	IPRA	16	H'FFFF8348		8, 16	
—	—	—	H'FFFF834A to H'FFFF834D		—	
Interrupt priority register D	IPRD	16	H'FFFF834E		8, 16	
Interrupt priority register E	IPRE	16	H'FFFF8350		8, 16, 32	
Interrupt priority register F	IPRF	16	H'FFFF8352		8, 16	
Interrupt priority register G	IPRG	16	H'FFFF8354		8, 16, 32	
Interrupt priority register H	IPRH	16	H'FFFF8356		8, 16	
Interrupt control register 1	ICR1	16	H'FFFF8358		8, 16, 32	
IRQ status register	ISR	16	H'FFFF835A		8, 16	
Interrupt priority register I	IPRI	16	H'FFFF835C		8, 16, 32	
—	—	—	H'FFFF835E to H'FFFF8365		—	
Interrupt control register 2	ICR2	8	H'FFFF8366		8, 16	
—	—	—	H'FFFF8368 to H'FFFF837F		—	
—	—	—	H'FFFF8380 to H'FFFF8381	—	—	—
Port A data register L	PADRL	16	H'FFFF8382	I/O	8, 16	In ϕ cycles B: 2 W: 2 L: 4
—	—	—	H'FFFF8384 to H'FFFF8385	—	—	
Port A I/O register L	PAIORL	16	H'FFFF8386	PFC	8, 16	
—	—	—	H'FFFF8388 to H'FFFF8389	—	—	
Port A control register L3	PACRL3	16	H'FFFF838A	PFC	8, 16	
Port A control register L1	PACRL1	16	H'FFFF838C		8, 16, 32	
Port A control register L2	PACRL2	16	H'FFFF838E		8, 16	
Port B data register	PBDR	16	H'FFFF8390	I/O	8, 16	
—	—	—	H'FFFF8392 to H'FFFF8393	—	—	

Register Name	Abbreviation	Bits	Address	Module	Access Size	Access States
Port B I/O register	PBIOR	16	H'FFFF8394	PFC	8, 16, 32	In ϕ cycles B: 2 W: 2 L: 4
—	—	—	H'FFFF8396 to H'FFFF8397	—	—	
Port B control register 1	PBCR1	16	H'FFFF8398	PFC	8, 16, 32	
Port B control register 2	PBCR2	16	H'FFFF839A		8, 16	
—	—	—	H'FFFF839C to H'FFFF83AE	—	—	
Port E data register L	PEDRL	16	H'FFFF83B0	I/O	8, 16, 32	
Port F data register	PFDR	16	H'FFFF83B2		8, 16	
Port E I/O register L	PEIORL	16	H'FFFF83B4	PFC	8, 16, 32	
Port E I/O register H	PEIORH	16	H'FFFF83B6		8, 16	
Port E control register L1	PECRL1	16	H'FFFF83B8		8, 16, 32	
Port E control register L2	PECRL2	16	H'FFFF83BA		8, 16	
Port E control register H	PECRH	16	H'FFFF83BC		8, 16, 32	
Port E data register H	PEDRH	16	H'FFFF83BE	I/O	8, 16	
Input control/status register 1	ICSR1	16	H'FFFF83C0	MTU	8, 16, 32	In ϕ cycles B: 2 W: 2 L: 4
Output control/status register	OCSR	16	H'FFFF83C2		8, 16	
—	—	—	H'FFFF83C4 to H'FFFF83CC	—	—	
Port G data register	PGDR	8	H'FFFF83CD	I/O	8	In Pϕ cycles B: 2 W: 2 L: 4
—	—	—	H'FFFF83CE to H'FFFF83CF	—	—	—
Compare match timer start register	CMSTR	16	H'FFFF83D0	CMT	8, 16, 32	In ϕ cycles B: 2 W: 2 L: 4
Compare match timer control/status register_0	CMCSR_0	16	H'FFFF83D2		8, 16	
Compare match timer counter_0	CMCNT_0	16	H'FFFF83D4		8, 16, 32	
Compare match timer constant register_0	CMCOR_0	16	H'FFFF83D6		8, 16	

Register Name	Abbreviation	Bits	Address	Module	Access Size	Access States
Compare match timer control/status register_1	CMCSR_1	16	H'FFFF83D8	CMT	8, 16, 32	In φ cycles B: 2 W: 2 L: 4
Compare match timer counter_1	CMCNT_1	16	H'FFFF83DA		8, 16	
Compare match timer constant register_1	CMCOR_1	16	H'FFFF83DC		8, 16	
—	—	—	H'FFFF83DE			
—	—	—	H'FFFF83E0 to H'FFFF842E	—	—	—
A/D data register 8	ADDR8	16	H'FFFF8430	A/D (channel 0)	8, 16	In Pφ cycles B: 3 W: 6
A/D data register 9	ADDR9	16	H'FFFF8432		8, 16	
A/D data register 10	ADDR10	16	H'FFFF8434		8, 16	
A/D data register 11	ADDR11	16	H'FFFF8436		8, 16	
A/D data register 12	ADDR12	16	H'FFFF8438	A/D (channel 1)	8, 16	
A/D data register 13	ADDR13	16	H'FFFF843A		8, 16	
A/D data register 14	ADDR14	16	H'FFFF843C		8, 16	
A/D data register 15	ADDR15	16	H'FFFF843E		8, 16	
—	—	—	H'FFFF8440 to H'FFFF847F	—	—	
A/D control/status register_0	ADCSR_0	8	H'FFFF8480	A/D	8, 16	
A/D control/status register_1	ADCSR_1	8	H'FFFF8481		8	
—	—	—	H'FFFF8482 to H'FFFF8487		—	
A/D control register_0	ADCR_0	8	H'FFFF8488		8, 16	
A/D control register_1	ADCR_1	8	H'FFFF8489		8	
—	—	—	H'FFFF848A to H'FFFF860F		—	—
Timer control/status register	TCSR	8	H'FFFF8610	WDT	8*2/16*1	In φ cycles B: 3 W: 3
Timer counter	TCNT*1	8	H'FFFF8610	*1: Write cycle	16	
Timer counter	TCNT*2	8	H'FFFF8611	*2: Read cycle	8	
Reset control/status register	RSTCSR*1	8	H'FFFF8612		16	
Reset control/status register	RSTCSR*2	8	H'FFFF8613		8	
Standby control register	SBYCR	8	H'FFFF8614	Power-down state	8	In φ cycles B: 3

Register Name	Abbreviation	Bits	Address	Module	Access Size	Access States
—	—	—	H'FFFF8615 to H'FFFF8617	—	—	—
System control register	SYSCR	8	H'FFFF8618	Power-down state	8	In Pφ cycles B: 3 W: 3 L: 6
—	—	—	H'FFFF8619 to H'FFFF861B		—	
Module standby control register 1	MSTCR1	16	H'FFFF861C		8, 16, 32	
Module standby control register 2	MSTCR2	16	H'FFFF861E		8, 16	
Bus control register 1	BCR1	16	H'FFFF8620	BSC	8, 16, 32	In φ cycles B: 3 W: 3 L: 6
—	—	—	H'FFFF8622 to H'FFFF8626		—	
	—	—	H'FFFF8628 to H'FFFF87F3		—	
AD trigger select register	ADTSR	8	H'FFFF87F4	A/D	8	In Pφ cycles B: 3
	—	—	H'FFFF87F5 to H'FFFF89FF		—	
	—	—	H'FFFF8A00 to H'FFFFB4F3		—	

18.2 Register Bits

On-chip peripheral module register addresses and bit names are shown in the following table.

16-bit and 32-bit registers are shown in two and four rows of 8 bits, respectively.

Register Abbreviation	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module
SMR_2	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS1	CKS0	SCI (channel 2)
BRR_2									
SCR_2	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
TDR_2									
SSR_2	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT	
RDR_2									
SDCR_2	—	—	—	—	DIR	—	—	—	
SMR_3	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS1	CKS0	SCI (channel 3)
BRR_3									
SCR_3	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
TDR_3									
SSR_3	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT	
RDR_3									
SDCR_3	—	—	—	—	DIR	—	—	—	
—	—	—	—	—	—	—	—	—	—
TCR_3	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	MTU (channels 3 and 4)
TCR_4	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	
TMDR_3	—	—	BFB	BFA	MD3	MD2	MD1	MD0	
TMDR_4	—	—	BFB	BFA	MD3	MD2	MD1	MD0	
TIORH_3	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0	
TIORL_3	IOD3	IOD2	IOD1	IOD0	IOC3	IOC2	IOC1	IOC0	
TIORH_4	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0	
TIORL_4	IOD3	IOD2	IOD1	IOD0	IOC3	IOC2	IOC1	IOC0	
TIER_3	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
TIER_4	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
TOER	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B	
TOCR	—	PSYE	—	—	—	—	OLSN	OLSP	
TGCR	—	BDC	N	P	FB	WF	VF	UF	
TCNT_3									

Register Abbreviation	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module
TCNT_4									MTU (channels 3 and 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	
TSTR	CST4	CST3	—	—	—	CST2	CST1	CST0	
TSYR	SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0	
TCR_0	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	MTU (channel 0)
TMDR_0	—	—	BFB	BFA	MD3	MD2	MD1	MD0	
TIORH_0	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0	
TIORL_0	IOD3	IOD2	IOD1	IOD0	IOC3	IOC2	IOC1	IOC0	
TIER_0	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
TSR_0	—	—	—	TCFV	TGFD	TGFC	TGFB	TGFA	

Register Abbreviation	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module
TCNT_0									MTU (channel 0)
TGRA_0									
TGRB_0									
TGRC_0									
TGRD_0									
TCR_1	—	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	MTU (channel 1)
TMDR_1	—	—	—	—	MD3	MD2	MD1	MD0	
TIOR_1	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0	
TIER_1	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA	
TSR_1	TCFD	—	TCFU	TCFV	—	—	TGFB	TGFA	
TCNT_1									
TGRA_1									
TGRB_1									
TCR_2	—	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	MTU (channel 2)
TMDR_2	—	—	—	—	MD3	MD2	MD1	MD0	
TIOR_2	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0	
TIER_2	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA	
TSR_2	TCFD	—	TCFU	TCFV	—	—	TGFB	TGFA	
TCNT_2									
TGRA_2									
TGRB_2									
—	—	—	—	—	—	—	—	—	—

Register Abbreviation	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module
IPRA	IRQ0	IRQ0	IRQ0	IRQ0	IRQ1	IRQ1	IRQ1	IRQ1	INTC
	IRQ2	IRQ2	IRQ2	IRQ2	IRQ3	IRQ3	IRQ3	IRQ3	
IPRD	MTU0	MTU0	MTU0	MTU0	MTU0	MTU0	MTU0	MTU0	
	MTU1	MTU1	MTU1	MTU1	MTU1	MTU1	MTU1	MTU1	
IPRE	MTU2	MTU2	MTU2	MTU2	MTU2	MTU2	MTU2	MTU2	
	MTU3	MTU3	MTU3	MTU3	MTU3	MTU3	MTU3	MTU3	
IPRF	MTU4	MTU4	MTU4	MTU4	MTU4	MTU4	MTU4	MTU4	
	—	—	—	—	—	—	—	—	
IPRG	A/D0,1	A/D0,1	A/D0,1	A/D0,1	—	—	—	—	
	CMT0	CMT0	CMT0	CMT0	CMT1	CMT1	CMT1	CMT1	
IPRH	WDT	WDT	WDT	WDT	I/O(MTU)	I/O(MTU)	I/O(MTU)	I/O(MTU)	
	—	—	—	—	—	—	—	—	
ICR1	NMIL	—	—	—	—	—	—	NMIE	
	IRQ0S	IRQ1S	IRQ2S	IRQ3S	—	—	—	—	
ISR	—	—	—	—	—	—	—	—	
	IRQ0F	IRQ1F	IRQ2F	IRQ3F	—	—	—	—	
IPRI	SCI2	SCI2	SCI2	SCI2	SCI3	SCI3	SCI3	SCI3	
	—	—	—	—	—	—	—	—	
ICR2	IRQ0ES1	IRQ0ES0	IRQ1ES1	IRQ1ES0	IRQ2ES1	IRQ2ES0	IRQ3ES1	IRQ3ES0	
	—	—	—	—	—	—	—	—	
—	—	—	—	—	—	—	—	—	—
PADRL	PA15DR	PA14DR	PA13DR	PA12DR	PA11DR	PA10DR	PA9DR	PA8DR	Port A
	PA7DR	PA6DR	PA5DR	PA4DR	PA3DR	PA2DR	PA1DR	PA0DR	
PAIORL	PA15IOR	PA14IOR	PA13IOR	PA12IOR	PA11IOR	PA10IOR	PA9IOR	PA8IOR	
	PA7IOR	PA6IOR	PA5IOR	PA4IOR	PA3IOR	PA2IOR	PA1IOR	PA0IOR	
PACRL3	PA15MD2	PA14MD2	PA13MD2	PA12MD2	PA11MD2	PA10MD2	PA9MD2	PA8MD2	
	PA7MD2	PA6MD2	PA5MD2	PA4MD2	PA3MD2	PA2MD2	PA1MD2	PA0MD2	
PACRL1	PA15MD1	PA15MD0	PA14MD1	PA14MD0	PA13MD1	PA13MD0	PA12MD1	PA12MD0	
	PA11MD1	PA11MD0	PA10MD1	PA10MD0	PA9MD1	PA9MD0	PA8MD1	PA8MD0	
PACRL2	PA7MD1	PA7MD0	PA6MD1	PA6MD0	PA5MD1	PA5MD0	PA4MD1	PA4MD0	
	PA3MD1	PA3MD0	PA2MD1	PA2MD0	PA1MD1	PA1MD0	PA0MD1	PA0MD0	

Register Abbreviation	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module
PBDR	—	—	—	—	—	—	—	—	Port B
	—	—	PB5DR	PB4DR	PB3DR	PB2DR	—	—	
PBIOR	—	—	—	—	—	—	—	—	
	—	—	PB5IOR	PB4 IOR	PB3 IOR	PB2 IOR	—	—	
PBCR1	—	—	PB5MD2	PB4MD2	PB3MD2	PB2MD2	—	—	
	—	—	—	—	—	—	—	—	
PBCR2	—	—	—	—	PB5MD1	PB5MD0	PB4MD1	PB4MD0	
	PB3MD1	PB3MD0	PB2MD1	PB2MD0	—	—	—	—	
PEDRL	PE15DR	PE14DR	PE13DR	PE12DR	PE11DR	PE10DR	PE9DR	PE8DR	Port E
	PE7DR	PE6DR	PE5DR	PE4DR	PE3DR	PE2DR	PE1DR	PE0DR	
PFDR	PF15DR	PF14DR	PF13DR	PF12DR	PF11DR	PF10DR	PF9DR	PF8DR	Port F
	PF7DR	PF6DR	PF5DR	PF4DR	PF3DR	PF2DR	PF1DR	PF0DR	
PEIORL	PE15IOR	PE14 IOR	PE13 IOR	PE12 IOR	PE11 IOR	PE10 IOR	PE9 IOR	PE8 IOR	Port E
	PE7IOR	PE6IOR	PE5IOR	PE4IOR	PE3IOR	PE2IOR	PE1IOR	PE0IOR	
PEIORH	—	—	—	—	—	—	—	—	
	—	—	PE21IOR	PE20IOR	PE19IOR	PE18IOR	PE17IOR	PE16IOR	
PECRL1	PE15MD1	PE15MD0	PE14MD1	PE14MD0	PE13MD1	PE13MD0	PE12MD1	PE12MD0	
	PE11MD1	PE11MD0	PE10MD1	PE10MD0	PE9MD1	PE9MD0	PE8MD1	PE8MD0	
PECRL2	PE7MD1	PE7MD0	PE6MD1	PE6MD0	PE5MD1	PE5MD0	PE4MD1	PE4MD0	
	PE3MD1	PE3MD0	PE2MD1	PE2MD0	PE1MD1	PE1MD0	PE0MD1	PE0MD0	
PECRH	—	—	—	—	PE21MD1	PE21MD0	PE20MD1	PE20MD0	
	PE19MD1	PE19MD0	PE18MD1	PE18MD0	PE17MD1	PE17MD0	PE16MD1	PE16MD0	
PEDRH	—	—	—	—	—	—	—	—	
	—	—	PE21DR	PE20DR	PE19DR	PE18DR	PE17DR	PE16DR	
—	—	—	—	—	—	—	—	—	—
ICSR1	POE3F	POE2F	POE1F	POE0F	—	—	—	PIE	MTU
	POE3M1	POE3M0	POE2M1	POE2M0	POE1M1	POE1M0	POE0M1	POE0M0	
OCSR	OSF	—	—	—	—	—	OCE	OIE	
	—	—	—	—	—	—	—	—	
—	—	—	—	—	—	—	—	—	—
PGDR	—	—	—	—	PG3DR	PG2DR	PG1DR	PG0DR	Port G
—	—	—	—	—	—	—	—	—	—

Register Abbreviation	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module
CMSTR	—	—	—	—	—	—	—	—	CMT
	—	—	—	—	—	—	STR1	STR0	
CMCSR_0	—	—	—	—	—	—	—	—	
	CMF	CMIE	—	—	—	—	CKS1	CKS0	
CMCNT_0									
CMCOR_0									
CMCSR_1	—	—	—	—	—	—	—	—	
	CMF	CMIE	—	—	—	—	CKS1	CKS0	
CMCNT_1									
CMCOR_1									
—	—	—	—	—	—	—	—	—	—
ADDR8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D
	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	—	—	—	—	—	—	
ADDR13	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	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_0	ADF	ADIE	ADM1	ADM0	—	CH2	CH1	CH0	
ADCSR_1	ADF	ADIE	ADM1	ADM0	—	CH2	CH1	CH0	
ADCR_0	TRGE	CKS1	CKS0	ADST	ADCS	—	—	—	
ADCR_1	TRGE	CKS1	CKS0	ADST	ADCS	—	—	—	
—	—	—	—	—	—	—	—	—	—

Register Abbreviation	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module
TCSR	OVF	WT/$\overline{IT}$	TME	—	—	CKS2	CKS1	CKS0	WDT
TCNT									
RSTCSR	WOVF	RSTE	RSTS	—	—	—	—	—	
—	—	—	—	—	—	—	—	—	—
SBYCR	SSBY	HIZ	—	—	—	—	—	IRQEL	Power-down state
SYSCR	—	—	—	—	—	—	—	RAME	
MSTCR1	—	—	—	—	MSTP27	MSTP26	—	—	
	—	—	—	—	MSTP19	MSTP18	—	—	
MSTCR2	—	—	MSTP13	MSTP12	—	—	—	—	
	—	—	MSTP5	MSTP4	—	—	—	—	
—	—	—	—	—	—	—	—	—	—
BCR1	—	—	MTURWE	—	—	—	—	—	BSC
	—	—	—	—	—	—	—	—	
—	—	—	—	—	—	—	—	—	—
ADTSR	—	—	—	—	TRG1S1	TRG1S0	TRG0S1	TRG0S0	A/D
—	—	—	—	—	—	—	—	—	—

18.3 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	Sleep	Module
SMR_2	Initialized	Held	Initialized	Initialized	Held	SCI (channel 2)
BRR_2	Initialized	Held	Initialized	Initialized	Held	
SCR_2	Initialized	Held	Initialized	Initialized	Held	
TDR_2	Initialized	Held	Initialized	Initialized	Held	
SSR_2	Initialized	Held	Initialized	Initialized	Held	
RDR_2	Initialized	Held	Initialized	Initialized	Held	
SDCR_2	Initialized	Held	Initialized	Initialized	Held	
SMR_3	Initialized	Held	Initialized	Initialized	Held	SCI (channel 3)
BRR_3	Initialized	Held	Initialized	Initialized	Held	
SCR_3	Initialized	Held	Initialized	Initialized	Held	
TDR_3	Initialized	Held	Initialized	Initialized	Held	
SSR_3	Initialized	Held	Initialized	Initialized	Held	
RDR_3	Initialized	Held	Initialized	Initialized	Held	
SDCR_3	Initialized	Held	Initialized	Initialized	Held	
TCR_3	Initialized	Held	Initialized	Initialized	Held	MTU (channels 3 and 4)
TCR_4	Initialized	Held	Initialized	Initialized	Held	
TMDR_3	Initialized	Held	Initialized	Initialized	Held	
TMDR_4	Initialized	Held	Initialized	Initialized	Held	
TIORH_3	Initialized	Held	Initialized	Initialized	Held	
TIORL_3	Initialized	Held	Initialized	Initialized	Held	
TIORH_4	Initialized	Held	Initialized	Initialized	Held	
TIORL_4	Initialized	Held	Initialized	Initialized	Held	
TIER_3	Initialized	Held	Initialized	Initialized	Held	
TIER_4	Initialized	Held	Initialized	Initialized	Held	
TOER	Initialized	Held	Initialized	Initialized	Held	
TOCR	Initialized	Held	Initialized	Initialized	Held	
TGCR	Initialized	Held	Initialized	Initialized	Held	
TCNT_3	Initialized	Held	Initialized	Initialized	Held	
TCNT_4	Initialized	Held	Initialized	Initialized	Held	
TCDR	Initialized	Held	Initialized	Initialized	Held	
TDDR	Initialized	Held	Initialized	Initialized	Held	

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	Sleep	Module
TGRA_3	Initialized	Held	Initialized	Initialized	Held	MTU (channels 3 and 4)
TGRB_3	Initialized	Held	Initialized	Initialized	Held	
TGRA_4	Initialized	Held	Initialized	Initialized	Held	
TGRB_4	Initialized	Held	Initialized	Initialized	Held	
TCNTS	Initialized	Held	Initialized	Initialized	Held	
TCBR	Initialized	Held	Initialized	Initialized	Held	
TGRC_3	Initialized	Held	Initialized	Initialized	Held	
TGRD_3	Initialized	Held	Initialized	Initialized	Held	
TGRC_4	Initialized	Held	Initialized	Initialized	Held	
TGRD_4	Initialized	Held	Initialized	Initialized	Held	
TSR_3	Initialized	Held	Initialized	Initialized	Held	
TSR_4	Initialized	Held	Initialized	Initialized	Held	
TSTR	Initialized	Held	Initialized	Initialized	Held	
TSYR	Initialized	Held	Initialized	Initialized	Held	
TCR_0	Initialized	Held	Initialized	Initialized	Held	MTU (channel 0)
TMDR_0	Initialized	Held	Initialized	Initialized	Held	
TIORH_0	Initialized	Held	Initialized	Initialized	Held	
TIORL_0	Initialized	Held	Initialized	Initialized	Held	
TIER_0	Initialized	Held	Initialized	Initialized	Held	
TSR_0	Initialized	Held	Initialized	Initialized	Held	
TCNT_0	Initialized	Held	Initialized	Initialized	Held	
TGRA_0	Initialized	Held	Initialized	Initialized	Held	
TGRB_0	Initialized	Held	Initialized	Initialized	Held	
TGRC_0	Initialized	Held	Initialized	Initialized	Held	
TGRD_0	Initialized	Held	Initialized	Initialized	Held	
TCR_1	Initialized	Held	Initialized	Initialized	Held	
TMDR_1	Initialized	Held	Initialized	Initialized	Held	
TIOR_1	Initialized	Held	Initialized	Initialized	Held	
TIER_1	Initialized	Held	Initialized	Initialized	Held	
TSR_1	Initialized	Held	Initialized	Initialized	Held	
TCNT_1	Initialized	Held	Initialized	Initialized	Held	

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	Sleep	Module
TGRA_1	Initialized	Held	Initialized	Initialized	Held	MTU (channel 2)
TGRB_1	Initialized	Held	Initialized	Initialized	Held	
TCR_2	Initialized	Held	Initialized	Initialized	Held	
TMDR_2	Initialized	Held	Initialized	Initialized	Held	
TIOR_2	Initialized	Held	Initialized	Initialized	Held	
TIER_2	Initialized	Held	Initialized	Initialized	Held	
TSR_2	Initialized	Held	Initialized	Initialized	Held	
TCNT_2	Initialized	Held	Initialized	Initialized	Held	
TGRA_2	Initialized	Held	Initialized	Initialized	Held	
TGRB_2	Initialized	Held	Initialized	Initialized	Held	
IPRA	Initialized	Initialized	Held	—	Held	INTC
IPRD	Initialized	Initialized	Held	—	Held	
IPRE	Initialized	Initialized	Held	—	Held	
IPRF	Initialized	Initialized	Held	—	Held	
IPRG	Initialized	Initialized	Held	—	Held	
IPRH	Initialized	Initialized	Held	—	Held	
ICR1	Initialized	Initialized	Held	—	Held	
ISR	Initialized	Initialized	Held	—	Held	
IPRI	Initialized	Initialized	Held	—	Held	
ICR2	Initialized	Initialized	Held	—	Held	
PADRL	Initialized	Held	Held	—	Held	Port A
PAIORL	Initialized	Held	Held	—	Held	
PACRL3	Initialized	Held	Held	—	Held	
PACRL1	Initialized	Held	Held	—	Held	
PACRL2	Initialized	Held	Held	—	Held	
PBDR	Initialized	Held	Held	—	Held	Port B
PBIOR	Initialized	Held	Held	—	Held	
PBCR1	Initialized	Held	Held	—	Held	
PBCR2	Initialized	Held	Held	—	Held	
PEDRL	Initialized	Held	Held	—	Held	Port E
PFDR	Held	Held	Held	—	Held	Port F

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	Sleep	Module
PEIORL	Initialized	Held	Held	—	Held	Port E
PEIORH	Initialized	Held	Held	—	Held	
PECRL1	Initialized	Held	Held	—	Held	
PECRL2	Initialized	Held	Held	—	Held	
PECRH	Initialized	Held	Held	—	Held	
PEDRH	Initialized	Held	Held	—	Held	
ICSR1	Initialized	Held	Held	Held	Held	MTU
OCSR	Initialized	Held	Held	Held	Held	
PGDR	Held	Held	Held	—	Held	Port G
CMSTR	Initialized	Held	Initialized	Initialized	Held	CMT
CMCSR_0	Initialized	Held	Initialized	Initialized	Held	
CMCNT_0	Initialized	Held	Initialized	Initialized	Held	
CMCOR_0	Initialized	Held	Initialized	Initialized	Held	
CMCSR_1	Initialized	Held	Initialized	Initialized	Held	
CMCNT_1	Initialized	Held	Initialized	Initialized	Held	
CMCOR_1	Initialized	Held	Initialized	Initialized	Held	
ADDR8	Initialized	Held	Initialized	Initialized	Held	A/D
ADDR9	Initialized	Held	Initialized	Initialized	Held	
ADDR10	Initialized	Held	Initialized	Initialized	Held	
ADDR11	Initialized	Held	Initialized	Initialized	Held	
ADDR12	Initialized	Held	Initialized	Initialized	Held	
ADDR13	Initialized	Held	Initialized	Initialized	Held	
ADDR14	Initialized	Held	Initialized	Initialized	Held	
ADDR15	Initialized	Held	Initialized	Initialized	Held	
ADCSR_0	Initialized	Held	Initialized	Initialized	Held	
ADCSR_1	Initialized	Held	Initialized	Initialized	Held	
ADCR_0	Initialized	Held	Initialized	Initialized	Held	
ADCR_1	Initialized	Held	Initialized	Initialized	Held	
TCSR	Initialized	Initialized	Held	—	Held	WDT
TCNT	Initialized	Initialized	Held	—	Held	
RSTCSR	Initialized	Held	Initialized	—	Held	

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	Sleep	Module
SBYCR	Initialized	Initialized	Held	—	Held	Power-down state
SYSCR	Initialized	Held	Held	—	Held	
MSTCR1	Initialized	Held	Held	—	Held	
MSTCR2	Initialized	Held	Held	—	Held	
BCR1	Initialized	Held	Held	—	Held	BSC
ADTSR	Initialized	Held	Held	—	Held	A/D

Section 19 Electrical Characteristics

19.1 Absolute Maximum Ratings

Table 19.1 shows the absolute maximum ratings.

Table 19.1 Absolute Maximum Ratings

Item		Symbol	Rating	Unit
Power supply voltage		V_{CC}	–0.3 to +7.0	V
Input voltage	EXTAL pin	Vin	–0.3 to V_{CC} +0.3	V
	All pins other than analog input and EXTAL pins	Vin	–0.3 to V_{CC} +0.3	V
Analog supply voltage		AV_{CC}	–0.3 to +7.0	V
Analog input voltage		V_{AN}	–0.3 to AV_{CC} +0.3	V
Operating temperature	Standard product*	T_{opr}	–20 to +75	°C
	Wide temperature-range product*		–40 to +85	
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.

Note: * See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

19.2 DC Characteristics

Table 19.2 DC Characteristics

Conditions: V_{CC} = 4.0 to 5.5 V, AV_{CC} = 4.0 to 5.5 V, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +75°C (Standard product)*, T_a = –40°C to +85°C (Wide temperature-range product)*

Item		Symbol	Min	Typ	Max	Unit	Measurement Conditions
Input high-level voltage (except Schmitt trigger input voltage)	$\overline{RES}$, $\overline{MRES}$, NMI, FWP, MD3 to MD0	V_{IH}	V_{CC}–0.7	—	V_{CC} +0.3	V	
	EXTAL		V_{CC}–0.7	—	V_{CC} +0.3	V	
	A/D port		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, FWP, MD3 to MD0, EXTAL	V_{IL}	–0.3	—	0.5	V	
	Other input pins		–0.3	—	0.8	V	
Schmitt trigger input voltage	$\overline{IRQ3}$ to $\overline{IRQ0}$, $\overline{POE3}$ to $\overline{POE0}$, TCLKA to TCLKD, TIOC0A to TIOC0D, TIOC1A, TIOC1B, TIOC2A, TIOC2B, TIOC3A to TIOC3D, TIOC4A to TIOC4D	V_{T+}	V_{CC} –0.5	—	V_{CC} +0.3	V	
		V_{T-}	–0.3	—	1.0	V	
		V_{T+}–V_{T-}	0.4	—	—	V	
Input leak current	$\overline{RES}$, $\overline{MRES}$, NMI, FWP, MD3 to MD0	$\| I_{in} \|$	—	—	1.0	μA	Vin = 0.5 to V_{CC} –0.5 V
	Ports F and G		—	—	1.0	μA	Vin = 0.5 to AV_{CC} –0.5 V
	Other input pins		—	—	1.0	μA	Vin = 0.5 to V_{CC} –0.5 V
Three-state leak current (while OFF)	Port A, B, E	$\| I_{tsi} \|$	—	—	1.0	μA	Vin = 0.5 to V_{CC} –0.5 V
Output high-level voltage	All output pins	V_{OH}	V_{CC} –0.5	—	—	V	I_{OH} = –200 μA
			V_{CC} –0.5	—	—	V	I_{OH} = –1 mA
Output low-level voltage	All output pins	V_{OL}	—	—	0.4	V	I_{OL} = 1.6 mA
	PE9, PE11 to PE15		—	—	1.5	V	I_{OL} = 15 mA
Pull-up resistor	PA15 to PA12, PE21 to PE16	R_{pull}	20	50	80	kΩ	

Item			Symbol	Min	Typ	Max	Unit	Measurement Conditions
Input capacitance	$\overline{RES}$		C_{in}	—	—	80	pF	Vin = 0 V
	NMI			—	—	50	pF	ϕ = 1 MHz
	All other input pins			—	—	20	pF	Ta = 25°C
Current consumption	Normal operation	Clock 1:1	I_{CC}	—	110	130	mA	ϕ = 40 MHz
		Clock 1:1/2		—	100	120	mA	ϕ = 40 MHz
	Sleep	Clock 1:1		—	70	90	mA	ϕ = 40 MHz
		Clock 1:1/2		—	60	80	mA	ϕ = 40 MHz
	Standby			—	1	10	µA	$T_a \leq 50°C$
				—	—	50	µA	$50°C < T_a$
Analog supply current	During A/D conversion, A/D converter idle state		AI_{CC}	—	3	5	mA	
	During standby			—	—	5	µA	
RAM standby voltage			V_{RAM}	2.0	—	—	V	V_{CC}

[Operating precautions]

1. When the A/D converter is not used, the AV_{CC}, and AV_{SS} pins should not be open.
2. The current consumption is measured when V_{IH}min = V_{CC} –0.5 V, V_{IL} = 0.5 V, with all output pins unloaded.

Note: * See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

Table 19.3 Permitted Output Current Values

Conditions: V_{CC} = 4.0 V to 0.5 V, AV_{CC} = 4.0 V to 5.5 V, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +75°C (Standard product)*[1], T_a = –40°C to +85°C (Wide temperature-range product)*[1]

Item	Symbol	Min	Typ	Max	Unit
Output low-level permissible current (per pin)	I_{OL}	—	—	2.0*[2]	mA
Output low-level permissible current (total)	$\Sigma\ I_{OL}$	—	—	110	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 this table.

Note: 1. See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

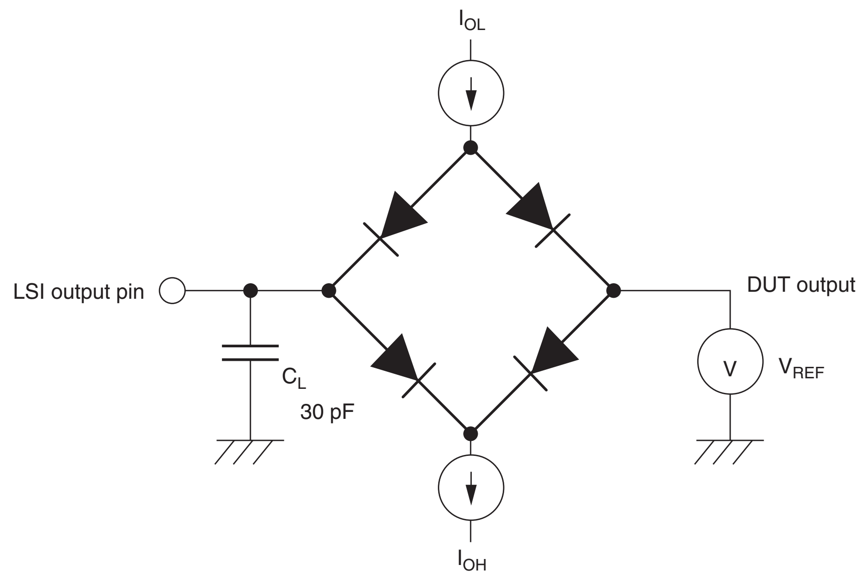
2. I_{OL}= 15 mA (max) about the pins PE9, PE11 to PE15. However, three pins at most are permitted to have simultaneously $I_{OL} > 2.0$ mA among these pins.

19.3 AC Characteristics

19.3.1 Test Conditions for the AC Characteristics

Input reference levels high level: V_{IH} minimum value, low level: V_{IL} maximum value

Output reference levels high level: 2.0 V, low level: 0.8 V



C_L is a total value that includes the capacitance of measurement equipment, and is set as follows:
30 pF: $\overline{IRQOUT}$
30 pF: Port output pins and peripheral module output pins other than the above
It is assumed that I_{OL} = 1.6 mA, I_{OH} = 200 A in the test conditions.

Figure 19.1 Output Load Circuit

19.3.2 Clock Timing

Table 19.4 shows the clock timing.

Table 19.4 Clock Timing

Conditions: V_{CC} = 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +75°C (Standard product)*, T_a = –40°C to +85°C (Wide temperature-range product)*

Item	Symbol	Min	Max	Unit	Figures
Operating frequency	f_{op}	10	40	MHz	Figure 19.2
Clock cycle time	t_{cyc}	25	250	ns	
Clock low-level pulse width	t_{CL}	4	—	ns	
Clock high-level pulse width	t_{CH}	4	—	ns	
Clock rise time	t_{CR}	—	5	ns	
Clock fall time	t_{CF}	—	5	ns	
EXTAL clock input frequency	f_{EX}	4	10.0	MHz	Figure 19.3
EXTAL clock input cycle time	t_{EXcyc}	100	250	ns	
EXTAL clock input low-level pulse width	t_{EXL}	45	—	ns	
EXTAL clock input high-level pulse width	t_{EXH}	45	—	ns	
EXTAL clock input rise time	t_{EXR}	—	5	ns	
EXTAL clock input fall time	t_{EXF}	—	5	ns	
Reset oscillation settling time	t_{OSC1}	10	—	ms	Figure 19.4
Standby return oscillation settling time	t_{OSC2}	10	—	ms	
Clock cycle time for on-chip peripheral modules	t_{pcyc}	25	100	ns	—

Note: * See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

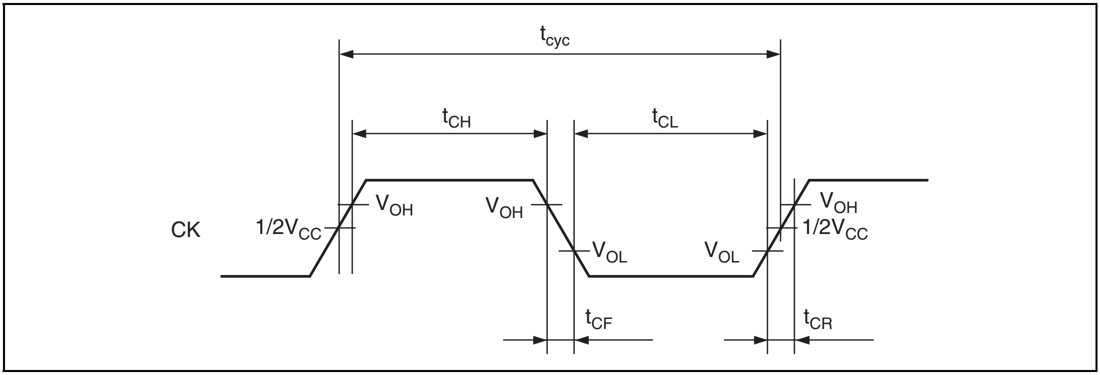


Figure 19.2 System Clock Timing

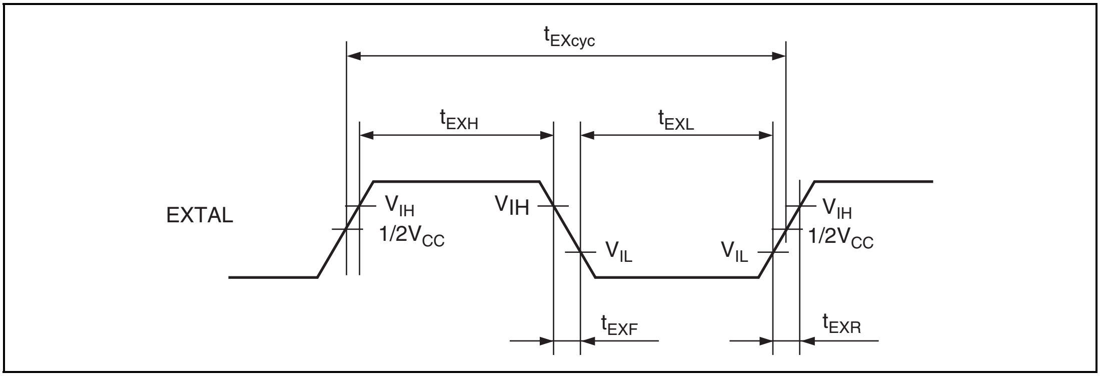


Figure 19.3 EXTAL Clock Input Timing

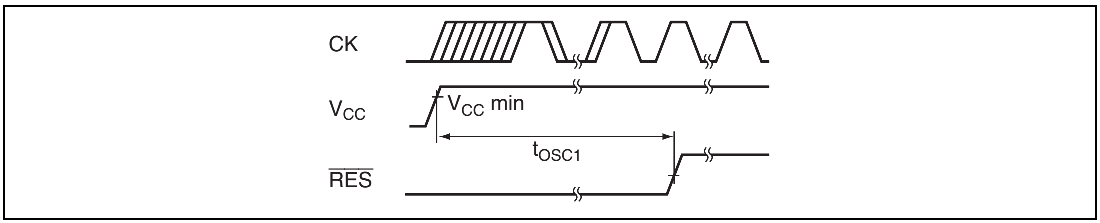


Figure 19.4 Oscillation Settling Time

19.3.3 Control Signal Timing

Table 19.5 shows control signal timing.

Table 19.5 Control Signal Timing

Conditions: V_{CC} = 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +75°C (Standard product)*[1], T_a = –40°C to +85°C (Wide temperature-range product)*[1]

Item	Symbol	Min	Max	Unit	Figures
$\overline{RES}$ rise time, fall time	t_{RESr}, t_{RESf}	—	200	ns	Figure 19.5
$\overline{RES}$ pulse width	t_{RESW}	25	—	t_{cyc}	Figure 19.6
$\overline{RES}$ setup time	t_{RESS}	19	—	ns	
$\overline{MRES}$ pulse width	t_{MRESW}	20	—	t_{cyc}	
$\overline{MRES}$ setup time	t_{MRESS}	19	—	ns	
MD3 to MD0, FWP setup time	t_{MDS}	20	—	t_{cyc}	
NMI rise time, fall time	t_{NMIr}, t_{NMIf}	—	200	ns	
NMI setup time	t_{NMIS}	19	—	ns	Figure 19.7
$\overline{IRQ3}$ to $\overline{IRQ0}$ setup time*[2] (edge detection)	t_{IRQES}	19	—	ns	
$\overline{IRQ3}$ to $\overline{IRQ0}$ setup time*[2] (level detection)	t_{IRQLS}	19	—	ns	
NMI hold time	t_{NMIH}	19	—	ns	
$\overline{IRQ3}$ to $\overline{IRQ0}$ hold time	t_{IRQEH}	19	—	ns	
$\overline{IRQOUT}$ output delay time	t_{IRQOD}	—	100	ns	Figure 19.8

[Operating precautions]

Notes: 1. See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

2. The $\overline{RES}$, $\overline{MRES}$, NMI and $\overline{IRQ3}$ to $\overline{IRQ0}$ signals are asynchronous inputs, but when the setup times shown here are observed, the signals are considered to have been changed at clock rise ($\overline{RES}$, $\overline{MRES}$) or fall (NMI and $\overline{IRQ3}$ to $\overline{IRQ0}$). If the setup times are not observed, detection of these signals may be delayed until the next clock rise or fall.

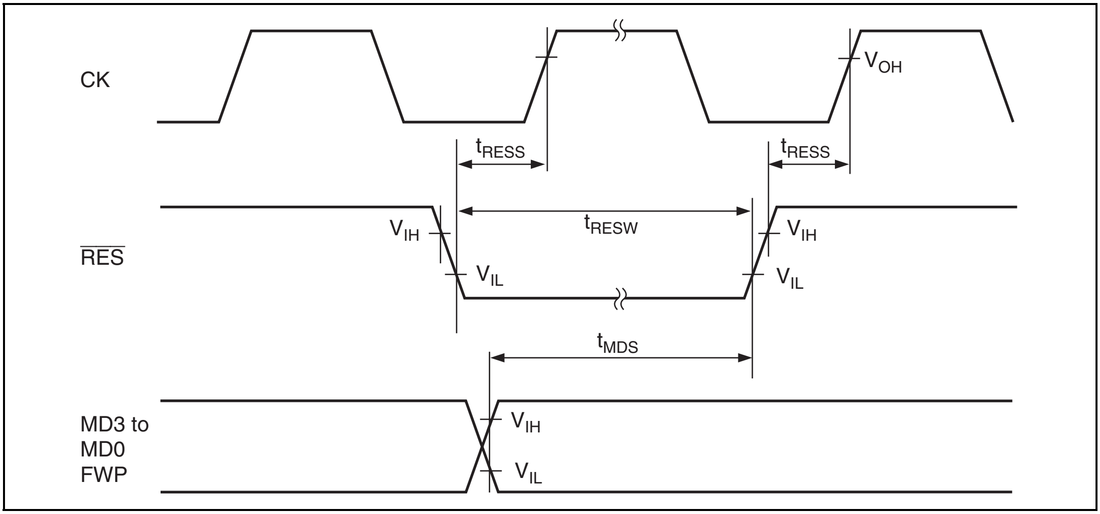


Figure 19.5 Reset Input Timing

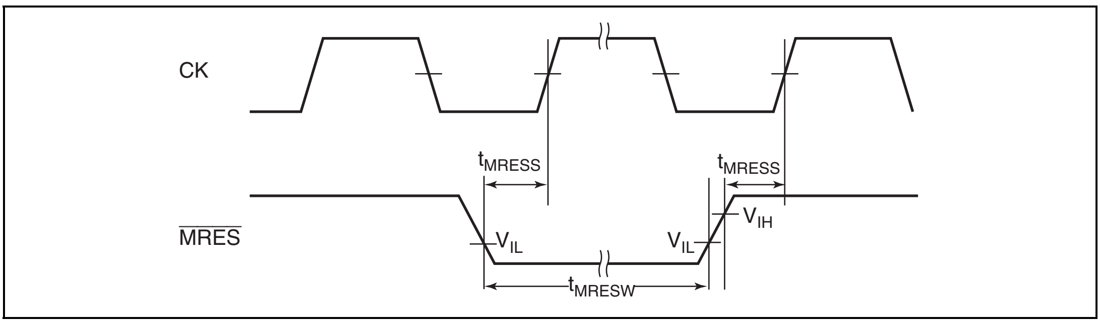


Figure 19.6 Reset Input Timing

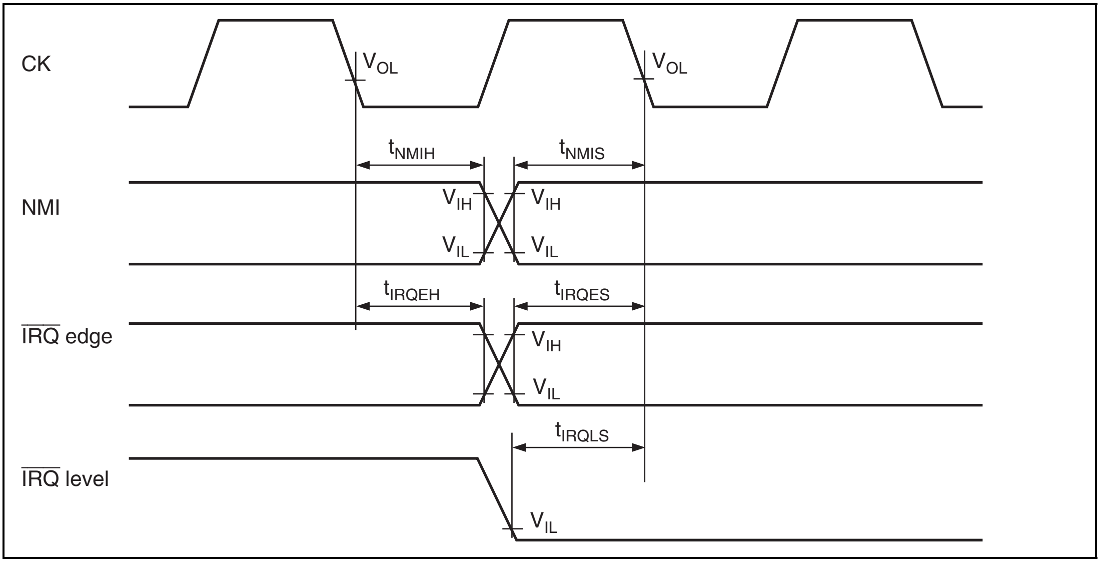


Figure 19.7 Interrupt Signal Input Timing

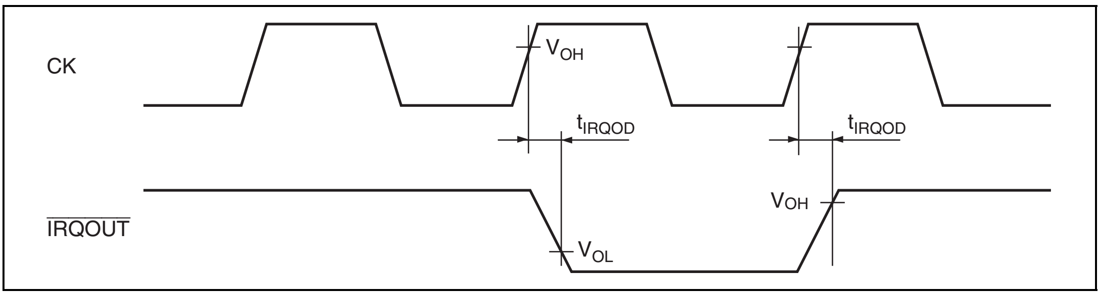


Figure 19.8 Interrupt Signal Output Timing

19.3.4 Multi-Function Timer Pulse Unit (MPU) Timing

Table 19.6 shows Multi-Function timer pulse unit timing.

Table 19.6 Multi-Function Timer Pulse Unit Timing

Conditions: V_{CC} = 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +75°C (Standard product*), T_a = –40°C to +85°C (Wide temperature-range product*)

Item	Symbol	Min	Max	Unit	Figures
Output compare output delay time	t_{TOCD}	—	100	ns	Figure 19.9
Input capture input setup time	t_{TICS}	19	—	ns	
Timer input setup time	t_{TCKS}	20	—	ns	Figure 19.10
Timer clock pulse width (single edge specified)	$t_{TCKWH/L}$	1.5	—	t_{pcyc}	
Timer clock pulse width (both edges specified)	$t_{TCKWH/L}$	2.5	—	t_{pcyc}	
Timer clock pulse width (phase count mode)	$t_{TCKWH/L}$	2.5	—	t_{pcyc}	

Note: * See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

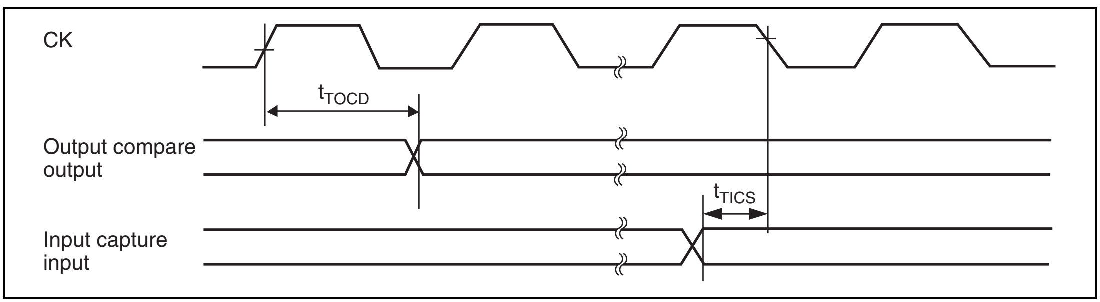


Figure 19.9 MTU Input/Output Timing

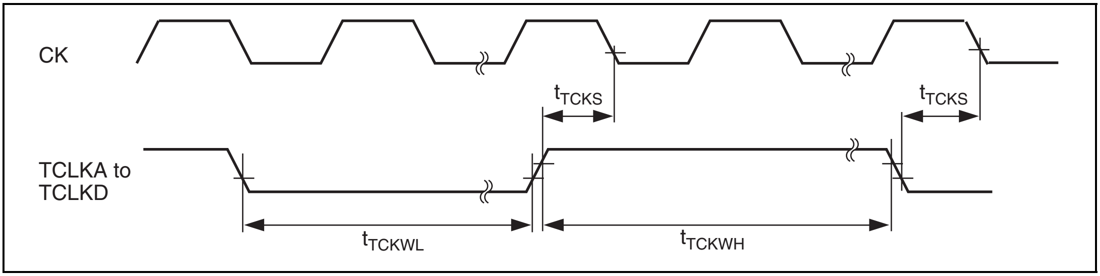


Figure 19.10 MTU Clock Input Timing

19.3.5 I/O Port Timing

Table 19.7 shows I/O port timing.

Table 19.7 I/O Port Timing

Conditions: V_{CC} = 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +75°C (Standard product*), T_a = –40°C to +85°C (Wide temperature-range product*)

Item	Symbol	Min	Max	Unit	Figures
Port output data delay time	t_{PWD}	—	100	ns	Figure 19.11
Port input hold time	t_{PRH}	19	—	ns	
Port input setup time	t_{PRS}	19	—	ns	

[Operating precautions]

The port input signals are asynchronous. They are, however, considered to have been changed at CK clock fall with two-state intervals shown in figure 19.11. If the setup times shown here are not observed, detection may be delayed until the clock fall two states after that timing.

Note: * See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

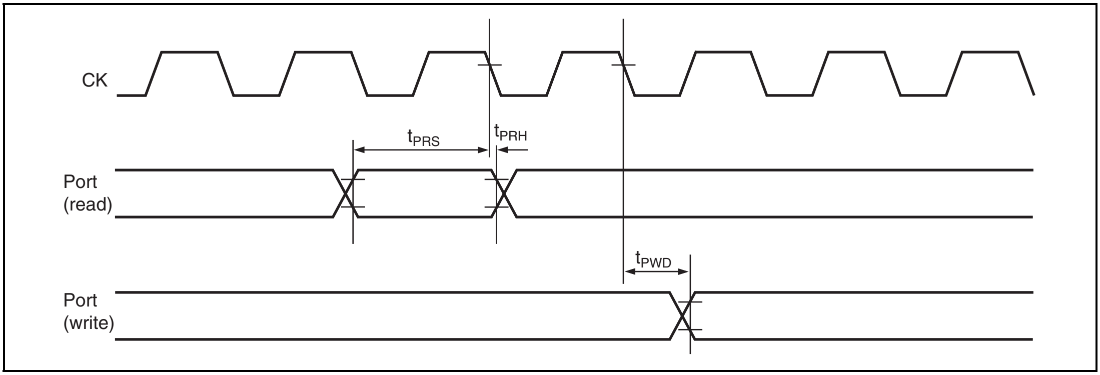


Figure 19.11 I/O Port Input/Output Timing

19.3.6 Watchdog Timer (WDT) Timing

Table 19.8 shows watchdog timer timing.

Table 19.8 Watchdog Timer Timing

Conditions: V_{CC} = 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +75°C (Standard product)*, T_a = –40°C to +85°C (Wide temperature-range product)*

Item	Symbol	Min	Max	Unit	Figures
$\overline{\text{WDTOVF}}$ delay time	t_{WOVD}	—	100	ns	Figure 19.12

Note: * See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

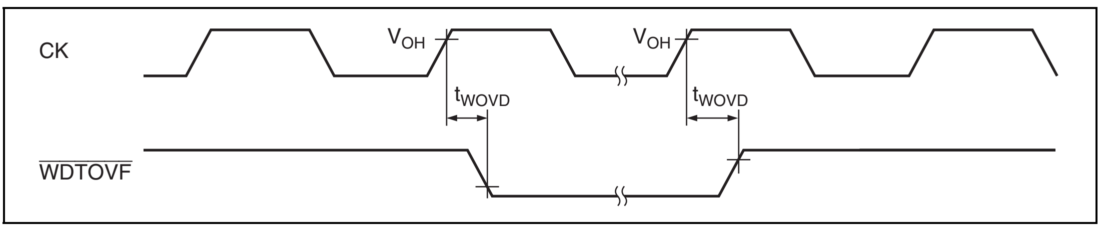


Figure 19.12 WDT Timing

19.3.7 Serial Communication Interface (SCI) Timing

Table 19.9 shows serial communication interface timing.

Table 19.9 Serial Communication Interface Timing

Conditions: V_{CC} = 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +75°C (Standard product)*, T_a = –40°C to +85°C (Wide temperature-range product)*

Item	Symbol	Min	Max	Unit	Figures
Input clock cycle	t_{scyc}	4	—	t_{pcyc}	Figure 19.13
Input clock cycle (clock sync)	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	t_{TxD}	—	100	ns	Figure 19.14
Received data setup time	t_{RxS}	100	—	ns	
Received data hold time	t_{RxH}	100	—	ns	

[Operating precautions]

The inputs and outputs are asynchronous in asynchronous mode, but as shown in figure 19.14, the received data is considered to have been changed at CK clock rise (two-clock intervals). The transmit signals change with a reference of CK clock rise (two-clock intervals).

Note: * See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

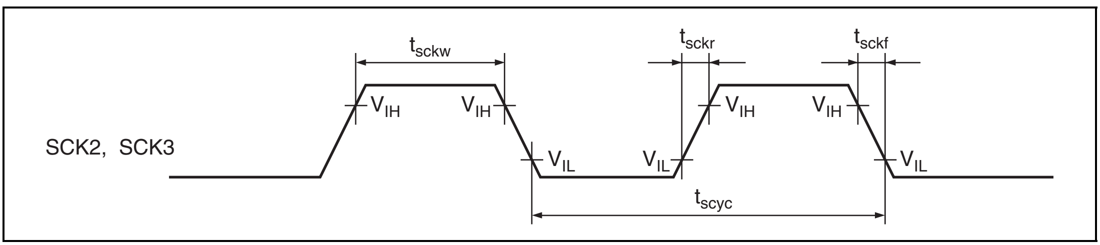


Figure 19.13 Input Clock Timing

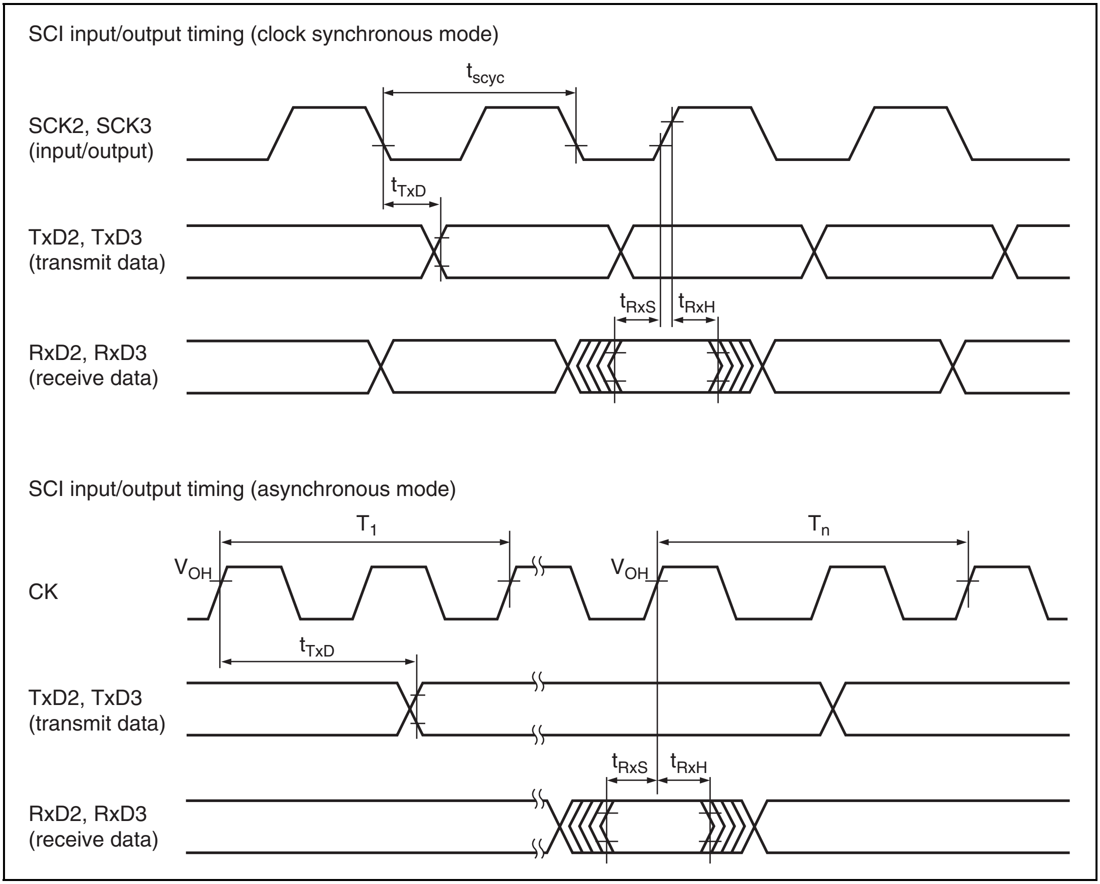


Figure 19.14 SCI Input/Output Timing

19.3.8 Output Enable (POE) Timing

Table 19.10 Output Enable Timing

Conditions: V_{CC} = 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +75°C (Standard product*), T_a = –40°C to +85°C (Wide temperature-range product*)

Item	Symbol	Min	Max	Unit	Figures
POE input setup time	t_{POES}	100	—	ns	Figure 19.15
POE input pulse width	t_{POEW}	1.5	—	t_{cyc}	

Note: * See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

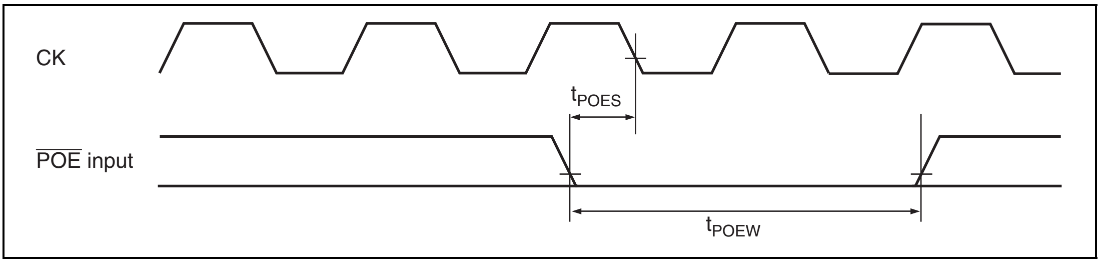


Figure 19.15 POE Input/Output Timing

19.3.9 A/D Converter Timing

Table 19.11 shows A/D converter timing.

Table 19.11 A/D Converter Timing

Conditions: V_{CC} = 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +75°C (Standard product*), T_a = –40°C to +85°C (Wide temperature-range product*)

Item	Symbol	Min	Typ	Max	Unit	Figure
External trigger input start delay time	t_{TRGS}	50	—	—	ns	Figure 19.16

Note: * See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.

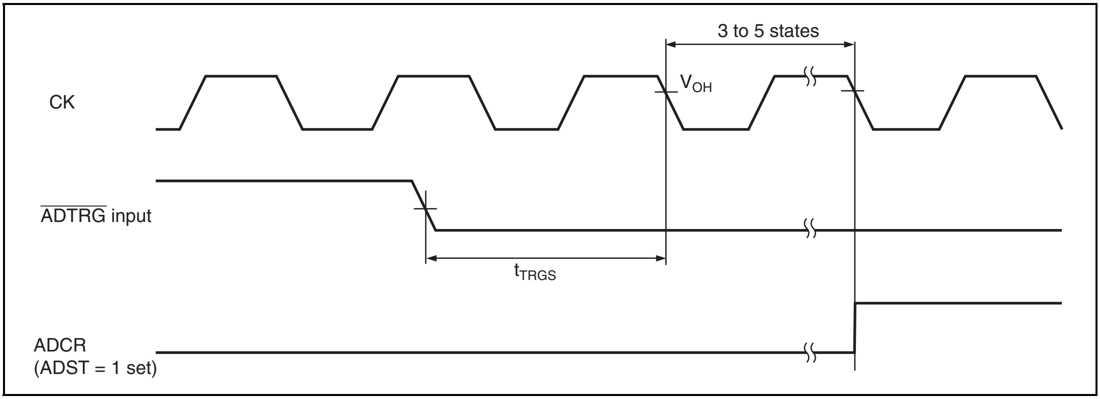


Figure 19.16 External Trigger Input Timing

19.4 A/D Converter Characteristics

Table 19.12 shows A/D converter characteristics.

Table 19.12 A/D Converter Characteristics

Conditions: V_{CC} = 4.0 V to 5.5 V, AV_{CC} = 4.0 V to 5.5 V, V_{SS} = $PLLV_{SS}$ = AV_{SS} = 0 V, T_a = –20°C to +75°C (Standard product)*[1], T_a = –40°C to +85°C (Wide temperature-range product)*

Item	Min	Typ	Max	Unit
Resolution	10	10	10/5.4	bit
A/D conversion time	—	—	6.7*[2]/5.4*[3]	μs
Analog input capacitance	—	—	20	pF
Permitted analog signal source impedance	—	—	3/1*[3]	kΩ
Non-linear error	—	—	±3.0*[2]/±5.0*[3]	LSB
Offset error	—	—	±3.0*[2]/±5.0*[3]	LSB
Full-scale error	—	—	±3.0*[2]/±5.0*[3]	LSB
Quantization error	—	—	±0.5	LSB
Absolute error	—	—	±4.0*[2]/±6.0*[3]	LSB

Notes: 1. See page 2 for correspondence of the standard product, wide temperature-range product, and product model name.
2. Value when (CKS1, 0) = (1, 1) and t_{pcyc} = 50 ns
3. Value when (CKS1, 0) = (1, 1) and t_{pcyc} = 40 ns

Appendix A Pin States

The initial values differ in each MCU operating mode. For details, refer to section 13, Pin Function Controller (PFC).

Table A.1 Pin States

Pin Function		Pin State			
		Reset State		Power-Down State	
Type	Pin Name	Power-On	Manual	Software Standby	Sleep
Clock	XTAL	O	O	L	O
	EXTAL	I	I	I	I
	PLLCAP	I	I	I	I
System Control	$\overline{\text{RES}}$	I	I	I	I
	$\overline{\text{MRES}}$	Z	I	Z*[2]	I
	$\overline{\text{WDTOVF}}$	O*[3]	O	O	O
Operation Mode Control	MD0 to MD3	I	I	I	I
	FWP	I	I	I	I
Interrupt	NMI	I	I	I	I
	$\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$	Z	I	Z*[4]	I
	$\overline{\text{IRQOUT}}$	Z	O	K*[1]	O
MTU	TCLKA to TCLKD	Z	I	Z	I
	TIOC0A to TIOC0D	Z	I/O	K*[1]	I/O
	TIOC1A, TIOC1B				
	TIOC2A, TIOC2B				
	TIOC3A, TIOC3C				
	TIOC3B, TIOC3D	Z	I/O	Z*[2]	I/O
	TIOC4A to TIOC4D				
Port control	$\overline{\text{POE0}}$ to $\overline{\text{POE3}}$	Z	I	Z	I
SCI	SCK2, SCK3	Z	I/O	Z	I/O
	RXD2, RXD3	Z	I	Z	I
	TXD2, TXD3	Z	O	O*[1]	O

Pin Function		Pin State			
		Reset State		Power-Down State	
Type	Pin Name	Power-On	Manual	Software Standby	Sleep
A/D converter	AN8 to AN15	Z	I	Z	I
	$\overline{ADTRG}$	Z	I	Z	I
I/O port	PA0 to PA15	Z	I/O	K*[1]	I/O
	PB2 to PB5				
	PE0 to PE8, PE10, PE16 to PE21				
	PE9, PE11 to PE15	Z	I/O	Z*[2]	I/O
	PF8 to PF15	Z	I	Z	I
	PG0 to PG3	Z	I	Z	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. When the HIZ bit in SBYCR is set to 1, the output pins enter their high-impedance state.

2. Those pins multiplexed with large-current pins (PE9, PE11 to PE15) unconditionally enter their high-impedance state.

3. This pin operates as an input pin during a power-on reset. This pin should be pulled up to avoid malfunction.

4. This pin operates as an input pin when the IRQEL bit in SBYCR is cleared to 0.

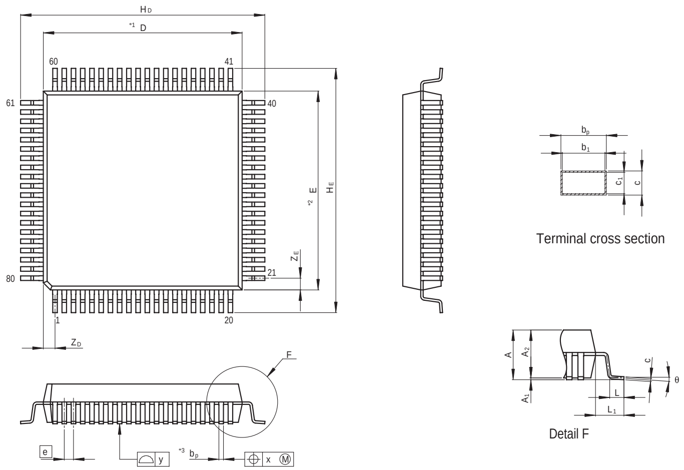
Appendix B Product Lineup

Product Type			Part No.	Package (Package Code)
SH7101	Mask ROM version	Standard product	HD6437101	QFP-80 (FP-80Q)

Appendix C 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-QFP80-14x14-0.65	PRQP0080JD-A	FP-80Q/FP-80QV	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	—	14	—
E	—	14	—
A_2	—	2.70	—
H_D	17.0	17.2	17.4
H_E	17.0	17.2	17.4
A	—	—	3.05
A_1	0.00	0.10	0.25
b_p	0.24	0.32	0.40
b_1	—	0.30	—
c	0.12	0.17	0.22
c_1	—	0.15	—
θ	0°	—	8°
e	—	0.65	—
x	—	—	0.12
y	—	—	0.10
Z_D	—	0.83	—
Z_E	—	0.83	—
L	0.6	0.8	1.0
L_1	—	1.6	—

Figure C.1 FP-80Q

Index

Renesas 32-Bit RISC Microcomputer
Hardware Manual
SH7101

Publication Date:	1st Edition, February, 2003
	Rev.2.00, September 27, 2007
Published by:	Sales Strategic Planning Div.
	Renesas Technology Corp.
Edited by:	Customer Support Department
	Global Strategic Communication Div.
	Renesas Solutions Corp.

Renesas Technology Corp. Sales Strategic Planning Div. Nippon Bldg., 2-6-2, Ohte-machi, Chiyoda-ku, Tokyo 100-0004, Japan

RENESAS SALES OFFICES

http://www.renesas.com

Refer to "**http://www.renesas.com/en/network**" for the latest and detailed information.

Renesas Technology America, Inc.
450 Holger Way, San Jose, CA 95134-1368, U.S.A
Tel: <1> (408) 382-7500, Fax: <1> (408) 382-7501

Renesas Technology Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: <44> (1628) 585-100, Fax: <44> (1628) 585-900

Renesas Technology (Shanghai) Co., Ltd.
Unit 204, 205, AZIACenter, No.1233 Lujiazui Ring Rd, Pudong District, Shanghai, China 200120
Tel: <86> (21) 5877-1818, Fax: <86> (21) 6887-7898

Renesas Technology Hong Kong Ltd.
7th Floor, North Tower, World Finance Centre, Harbour City, 1 Canton Road, Tsimshatsui, Kowloon, Hong Kong
Tel: <852> 2265-6688, Fax: <852> 2730-6071

Renesas Technology Taiwan Co., Ltd.
10th Floor, No.99, Fushing North Road, Taipei, Taiwan
Tel: <886> (2) 2715-2888, Fax: <886> (2) 2713-2999

Renesas Technology Singapore Pte. Ltd.
1 Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: <65> 6213-0200, Fax: <65> 6278-8001

Renesas Technology Korea Co., Ltd.
Kukje Center Bldg. 18th Fl., 191, 2-ka, Hangang-ro, Yongsan-ku, Seoul 140-702, Korea
Tel: <82> (2) 796-3115, Fax: <82> (2) 796-2145

Renesas Technology Malaysia Sdn. Bhd
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No.18, Jalan Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: <603> 7955-9390, Fax: <603> 7955-9510

Colophon 6.0

SH7101
Hardware Manual

Renesas Technology Corp.
2-6-2, Ote-machi, Chiyoda-ku, Tokyo, 100-0004, Japan