PanaXSeries

The One to Watch for Constant Innovation-Making the Future Come Alive

MICROCOMPUTER MN101C

MN101C 539
LSI User's Manual

Pub.No.21453-012E

Panasonic

Request for your special attention and precautions in using the technical informaition and semiconductors described in this book

(1) An export permit needs to be obtained from the competent authorities of the Japanese Government if any of the products or technologies described in this book and controlled under the "Foreign Exchange and Foreign Trade Law" is to be exported or taken out of Japan.

(2) The contents of this book are subject to change without notice in matters of improved function.When finalizing your design, therefore, ask for the most up-to-date version in advance in order to check for any changes.

(3) We are not liable for any damage arising out of the use of the contents of this book, or for any infringement of patents or any other rights owned by a third party.

(4) No part of this book may be reprinted or reproduced by any means without written permission from our company.

(5) This book deals with standard specification. Ask for the latest individual Product Standards or Specifications in advance for more detailsd infomation required for your design, purchasing and applications.

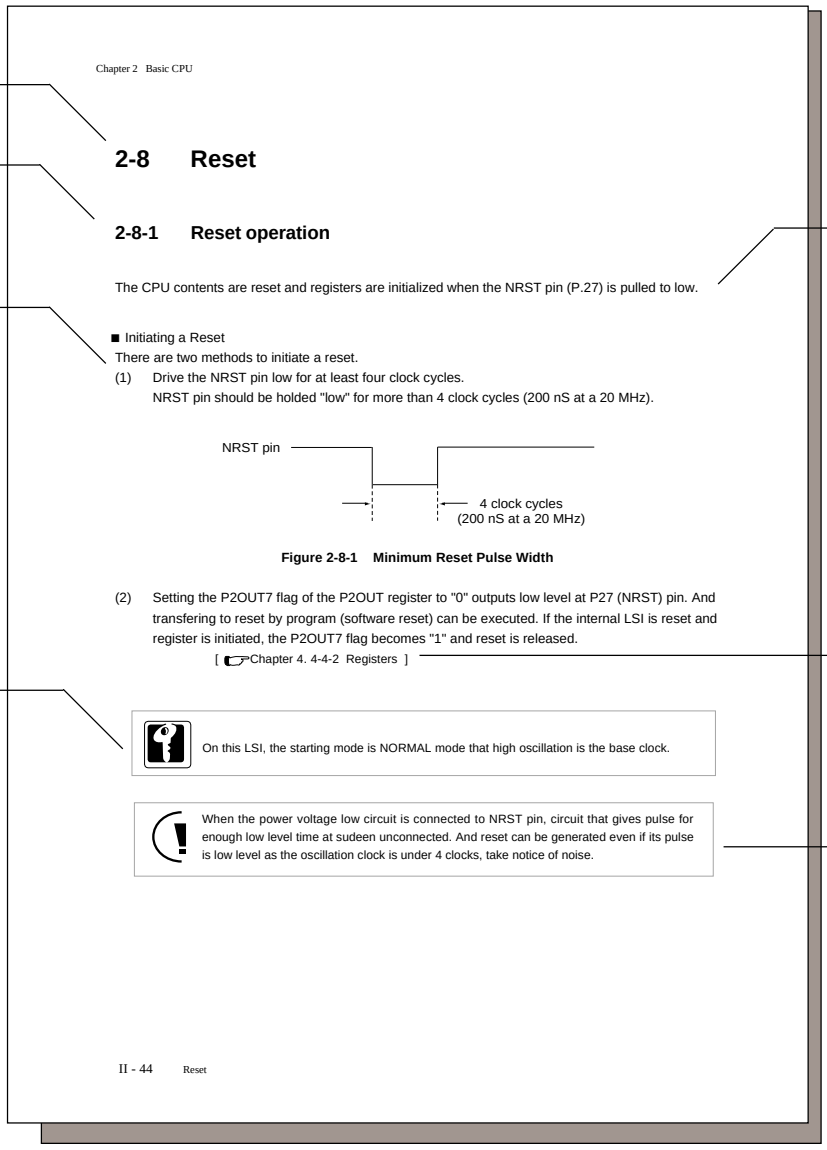
If you have any inquiries or questions about this book or our semiconductors, please contact one of our sales offices listed at the back of this book.

About This Manual

In this LSI manual, this LSI functions are presented in the following order : overview, basic CPU functions, interrupt functions, port functions, timer functions, serial functions, and other peripheral hardware functions.
Each section contains overview of function, block diagram, control register, operation, and setting example.

■Manual Configuration

Each section of this manual consists of a title, summary, main text, key information, precautions and warnings, and references.
The layout and definition of each section are shown below.



Subtitle

Sub-subtitle
The smallest block in this manual.

Main text

Summary
Introduction to the section.

Key information
Important information from the text.

References
References for the main text.

Precautions and warnings
Precautions are listed in case.
Be sure to read these of lost functionality or damage.

■Finding Desired Information

This manual provides three methods for finding desired information quickly and easily.

(1) Consult the index at the front of the manual to locate the beginning of each section.
(2) Consult the table of contents at the front of the manual to locate desired titles.
(3) Chapter names are located at the top outer corner of each page, and section titles are located at the bottom outer corner of each page.

■Related Manuals

Note that the following related documents are available.

"MN101C Series LSI user's Manual"
<Describes the device hardware>
"MN101C Series Instruction Manual"
<Describes the instruction set.>
"MN101C Series C Compiler User's Manual: Usage Guide"
<Describes the installation, the commands, and options of the C Compiler.>
"MN101C Series C Compiler User's Manual: Language Description"
<Describes the syntax of the C Compiler.>
"MN101C Series C Compiler User's Manual: Library Reference"
<Describes the standard library of the C Compiler.>
"MN101C Series Cross-assembler User's Manual"
<Describes the assembler syntax and notation.>
"MN101C Series C Source Code Debugger User's Manual"
<Describes the use of C source code debugger.>
"MN101C Series PanaX Series Installation Manual"
<Describes the installation of C compiler, cross-assembler and C source code debugger and the procedure for bringing up the in-circuit emulator.>

Contents

Chapter 1 Overview

Chapter 2 CPU Basics

Chapter 3 Interrupts

Chapter 4 I/O Ports

Chapter 5 Prescaler

Chapter 6 8-bit Timers

Chapter 7 16-bit Timer

Chapter 8 Time Base Timer / 8-bit Free-running Timer

Chapter 9 Watchdog Timer

Chapter 10 Buzzer

Chapter 11 Serial Interface 0

Chapter 12 A/D Converter

Chapter 13 Appendices

Chapter 1 Overview

1-1 Overview

1-1-1 Overview

The MN101C series of 8-bit single-chip microcontroller incorporate multiple types of peripheral functions. This chip series is well suited for camera, VCR, MD, TV, CD, LD, printer, telephone, home automation, pager, air conditioner, PPC remote control, fax machine, musical instrument, and other applications.

This LSI brings to embedded microcontroller applications flexible, optimized hardware configurations and a simple efficient instruction set. This LSI has an internal 24 KB of ROM and 512 bytes of RAM. Peripheral functions include 4 external interrupts, 10 internal interrupts including NMI, 5 timer counters, 1 set of serial interfaces, A/D converter, watchdog timer, buzzer output, and remote control output. The configuration of this microcontroller is well suited for application such as a system controller in a camera, VCR selection timer, CD player, or MD.

With two oscillation systems (max.20 MHz/32.768 kHz) contained on the chip, the system clock can be switched to high speed oscillation (**NORMAL mode**), or to low speed oscillation (**SLOW mode**). The system clock is generated by dividing the oscillation clock. The best operation clock for the system can be selected by switching its frequency by software.

When the oscillation source(fosc) is 8 MHz, **minimum instructions execution time** is for 238 ns, and when fosc is 20 MHz, it is 100 ns. The package is a 48-pin TQFP.

1-1-2 Product Summary

This manual describes the following models of the MN101C539 series. These products have same peripheral functions. MN101C539 is main in this manual. Differences between MN101C539 and MN101CP539 are shown in table 13-1-1 "Differences between MASK ROM version and internal EPROM version".

Table 1-1-1 Product Summary

Model	ROM Size	RAM Size	Classification
MN101C539	24 KB	512 bytes	Mask ROM version
MN101CP539	24 KB	512 bytes	EPROM version

1-2 Hardware Functions

CPU Core **MN101C Core**

- LOAD-STORE architecture (3-stage pipeline)
- Half-byte instruction set / Handy addressing
- Memory addressing space is 256 KB
- System clock switching function (at reset fs=fosc/64)
  System clock fs=fosc/2, fosc/4, fosc/8, fosc/64 (NORMAL mode)
  =fx/2, fx4 (SLOW mode)
- Minimum instructions execution time
  High speed oscillation
  0.100 µs / 20 MHz (4.5 V to 5.5 V)
  0.238 µs / 8.39 MHz (2.7 V to 5.5 V)
  1.000 µs / 4 MHz (2.0 V to 5.5 V) *1
  Low speed oscillation 61.04 µs / 32.768 kHz (2.0 V to 5.5 V) *1
  *1 : Minimum rating for EPROM vers. is 2.7 V to 5.5 V.
- Operation modes
  NORMAL mode (High speed oscillation)
  SLOW mode (Low speed oscillation)
  HALT mode
  STOP mode
  (The operation clock can be switched in each mode.)

Internal memory **ROM 24 KB**
RAM 512 bytes

Interrupts **10 Internal interrupts**
<Non-maskable interrupt (NMI)>
- Incorrect code execution interrupt and Watchdog timer interrupt
< Timer interrupts >
- Timer 2 interrupt
- Timer 3 interrupt
- Timer 6 interrupt
- Time base interrupt
- Timer 7 interrupt
- Match interrupt for Timer 7 compare register 2
< Serial interface interrupts >
- Serial interface 0 interrupt
- Serial interface 0 UART reception interrupt
< A/D interrupt >
- A/D converter interrupt

4 External interrupts

- IRQ0 : Edge selectable. With/Without noise filter.
  Both edges interrupt (STOP/HALT : can be recovered at the both edges)
- IRQ1 : Edge selectable. With/Without noise filter. AC zero cross detector.
  Both edges interrupt (STOP/HALT : can be recovered at the both edges)
- IRQ2 : Edge selectable. Serial interface 0 clock interrupt
- IRQ3 : Edge selectable. Key interrupt.

Timers

5 timers (4 can be operated independently)

- 8-bit timer for general use (UART baud rate timer)	2 sets
- 8-bit free-running timer	1 set
Time base timer	1 set
- 16-bit timer for general use	1 set

Timer 2 (8-bit timer for general use or UART baud rate timer)

- Square wave output (Timer pulse output), PWM output, Event count, Simple pulse width measurement, Serial interface transfer clock
- Clock source
  fosc, fosc/4, fosc/16, fosc/32, fosc/64, fs/2, fs/4, fx, external clock

Timer 3 (8-bit timer for general use or UART baud rate timer)

- Square wave output (Timer pulse output), Event count, Serial transfer clock, 16-bit cascade connection function (connect to timer 2), Remote control carrier output
- Clock source
  fosc, fosc/4, fosc/16, fosc/64, fosc/128, fs/2, fs/8, fx, external clock

Timer 6 (8-bit free-running timer, Time base timer)

❑ **8-bit free-running timer**

- Clock source
  fosc, fosc/2^{12}, fosc/2^{13}, fs, fx, fx/2^{12}, fx/2^{13}

❑ **Time base timer**

- Interrupt generation cycle
  fosc/2^{7}, fosc/2^{8}, fosc/2^{9}, fosc/2^{10}, fosc/2^{13}, fosc/2^{15},
  fx/2^{7}, fx/2^{8}, fx/2^{9}, fx/2^{10}, fx/2^{13}, fx/2^{15}

at 32.768 kHz for low speed oscillation input can be set to measure one minute intervals.

Timer 7 (16-bit timer for general use)

- Clock source

  fosc, fosc/2, fosc/4, fosc/16, fs, fs/2, fs/4, fs/16,
  1/1, 1/2, 1/4, 1/16 of the external clock

- Hardware organization

Compare register with double buffer	2 sets
Input capture register	1 set
Timer interrupt	2 vectors

- Timer functions

  Square wave output (Timer pulse output), Event count,
  High precision PWM output (Cycle / Duty continuous changeable),
  Timer synchronous output, Input capture function (Both edges can be operated), Remote control carrier output.

Watchdog timer

- Watchdog timer frequency can be selected from $fs/2^{16}$, $fs/2^{18}$ or $fs/2^{20}$.

Oscillation Stabilization wait timer

- Oscillation Stabilization wait time can be selected from among $2^2/fs$, $2^5/fs$, $2^{10}/fs$ and $2^{14}/fs$ (at reset $2^5/fs$).

Remote control output

Based on timer 3 pulse output and timer 7 PWM output, a remote control carrier with duty cycle of 1/2 or 1/3 can be output.
Timer 7 can be activated by generation of timer 2 interrupt.

Buzzer output Output frequency can be selected from $fosc/2^9$, $fosc/2^{10}$, $fosc/2^{11}$, $fosc/2^{12}$, $fosc/2^{13}$, $fosc/2^{14}$, $fx/2^3$, $fx/2^4$.

A/D converter **10 bits X 8 channels input**

Serial interface **1 type**

Serial interface 0 (Duplex UART / Synchronous serial interface)

❑ **Synchronous serial interface**

- Transfer clock source
  fosc/2, fosc/4, fosx/16, fosc/64, fs/2, fs/4
  UART baud rate timer (timers 2 and 3) output
- MSB/LSB can be selected as the first bit to be transferred. Any transfer size from 1 to 8 bits can be selected.
- Sequence transmission, sequence reception or both are available.
- SBO0 output control after transmission of last data (can be selected from H output, L output and maintaining of last data).
- Slave communitation in standby mode is available (can be recovered by an interrupt when communication is completed).

❑ **Duplex UART (Baud rate timer : Timers 2 and 3)**

- Parity check, Overrun error, Framing error detection
- Transfer size 7 to 8 bits can be selected.
- At UART communication, transmission / reception complete interrupts are available.

LED driver **8 pins**

Port	**I/O ports**	**36 pins**
	- LED (large current) driver pin	8 pins
	- External memory I/F pin	11 pins
	- D/A output pin	8 pins
	Input ports	**5 pins**
	- External interrupt pin	4 pins
	Special pins	
	- Operation mode input pin	1 pin
	- Reset input pin	1 pin
	- Power pin	2 pins
	- Oscillation pin	4 pins

Package **48-pin TQFP (7 mm square / 0.5 mm pitch)**
code name : TQFP048-P-0707B

1-3 Pin Description

1-3-1 Pin Configuration

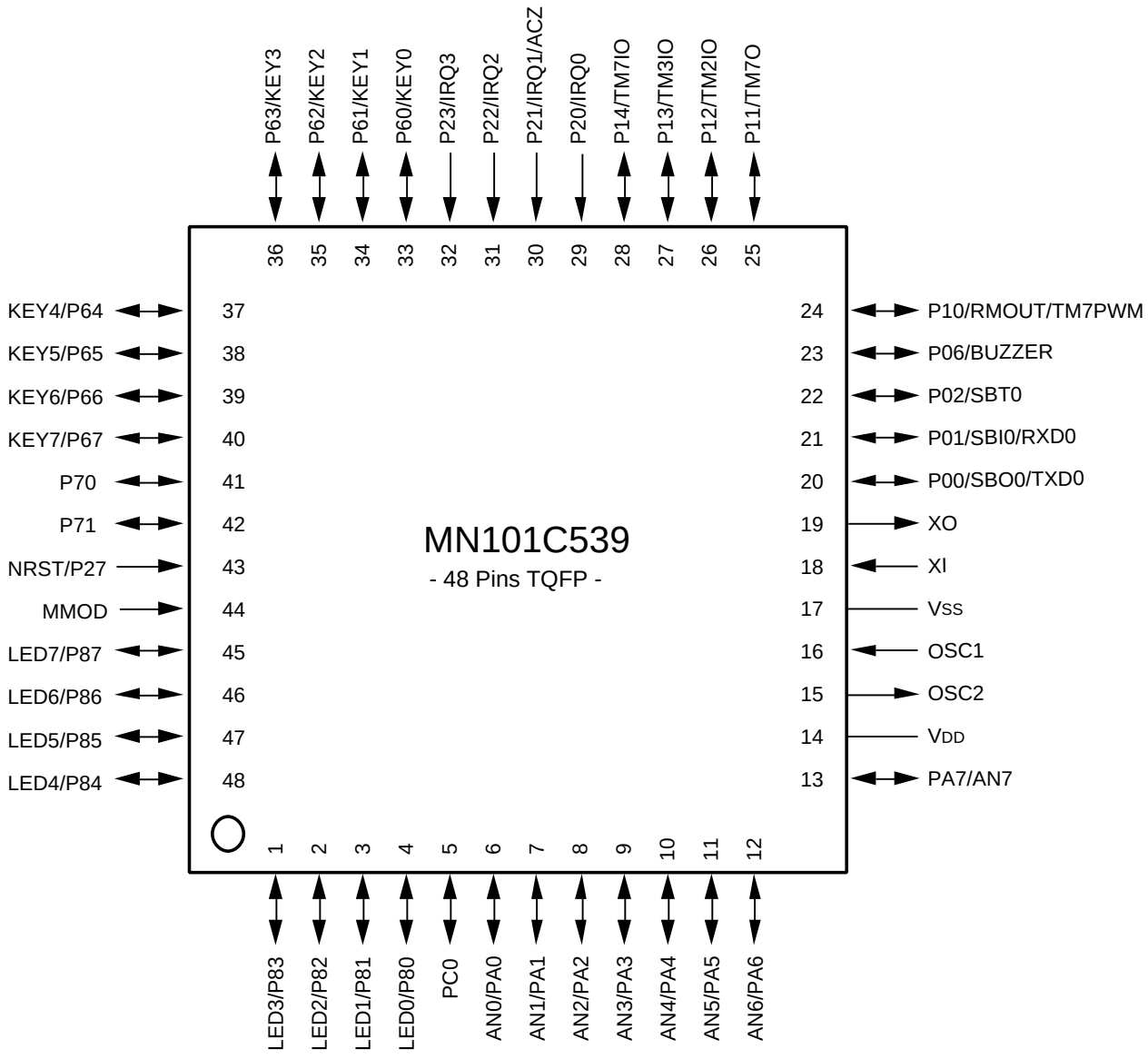


Figure 1-3-1 Pin Configuration (48 TQFP : Top view)

1-3-2 Pin Specification

Table 1-3-1 Pin Specification (1/2)

Pins	Special Functions	I/O	Direction Control	Pin Control	Functions Description
P00	SBO0/TXD0	in/out	P0DIR0	P0PUL0	SBO0 : Serial interface 0 transmission data output TXD0 : UART0 transmission data output
P01	SBI0/RXD0	in/out	P0DIR1	P0PUL1	SBI0 : Serial interface 0 reception data input RXD0 : UART0 reception data input
P02	SBT0	in/out	P0DIR2	P0PUL2	SBT0 : Serial interface 0 clock I/O
P06	BUZZER	in/out	P0DIR6	P0PUL6	BUZZER : Buzzer output
P10	RMOUT TM7PWM	in/out	P1DIR0	P1PUL0	RMOUT : Remote control carrier output TM7PWM : Timer 7PWM output
P11	TM7O	in/out	P1DIR1	P1PUL1	TM7O : Timer 7 output
P12	TM2IO	in/out	P1DIR2	P1PUL2	TM2IO : Timer 2 I/O
P13	TM3IO	in/out	P1DIR3	P1PUL3	TM3IO : Timer 3 I/O
P14	TM7IO	in/out	P1DIR4	P1PUL4	TM7IO : Timer 7 I/O
P20	IRQ0	in	-	P2PUL0	IRQ0 : External interrupt 0
P21	IRQ1 ACZ	in	-	P2PUL1	IRQ1 : External interrupt 1 ACZ : Zero-cross input
P22	IRQ2	in	-	P2PUL2	IRQ2 : External interrupt 2
P23	IRQ3	in	-	P2PUL3	IRQ3 : External interrupt 3
P27	NRST	in	-	-	NRST : Reset
P60	KEY0	in/out	P6DIR0	P6PUL0	KEY0 : Key interrupt input 0
P61	KEY1	in/out	P6DIR1	P6PUL1	KEY1 : Key interrupt input 1
P62	KEY2	in/out	P6DIR2	P6PUL2	KEY2 : Key interrupt input 2
P63	KEY3	in/out	P6DIR3	P6PUL3	KEY3 : Key interrupt input 3
P64	KEY4	in/out	P6DIR4	P6PUL4	KEY4 : Key interrupt input 4
P65	KEY5	in/out	P6DIR5	P6PUL5	KEY5 : Key interrupt input 5
P66	KEY6	in/out	P6DIR6	P6PUL6	KEY6 : Key interrupt input 6
P67	KEY7	in/out	P6DIR7	P6PUL7	KEY7 : Key interrupt input 7
P70		in/out	P7DIR0	P7PULD0	
P71		in/out	P7DIR1	P7PULD1	
P80	LED0	in/out	P8DIR0	P8PUL0	LED0 : LED driver pin 0
P81	LED1	in/out	P8DIR1	P8PUL1	LED1 : LED driver pin 1
P82	LED2	in/out	P8DIR2	P8PUL2	LED2 : LED driver pin 2
P83	LED3	in/out	P8DIR3	P8PUL3	LED3 : LED driver pin 3
P84	LED4	in/out	P8DIR4	P8PUL4	LED4 : LED driver pin 4
P85	LED5	in/out	P8DIR5	P8PUL5	LED5 : LED driver pin 5
P86	LED6	in/out	P8DIR6	P8PUL6	LED6 : LED driver pin 6
P87	LED7	in/out	P8DIR7	P8PUL7	LED7 : LED driver pin 7
PA0	AN0	in/out	PADIR0	PAPULD0	AN0 : Analog 0 input
PA1	AN1	in/out	PADIR1	PAPULD1	AN1 : Analog 1 input
PA2	AN2	in/out	PADIR2	PAPULD2	AN2 : Analog 2 input
PA3	AN3	in/out	PADIR3	PAPULD3	AN3 : Analog 3 input
PA4	AN4	in/out	PADIR4	PAPULD4	AN4 : Analog 4 input
PA5	AN5	in/out	PADIR5	PAPULD5	AN5 : Analog 5 input
PA6	AN6	in/out	PADIR6	PAPULD6	AN6 : Analog 6 input
PA7	AN7	in/out	PADIR7	PAPULD7	AN7 : Analog 7 input
PC0		in/out	PCDIR0	PCPUL0	

1-3-3 Pin Functions

Table 1-3-2 Pin Function Summary (1/5)

Name	No.	I/O	Other Function	Function	Description
V_{SS} V_{DD}	17 14			Power supply pin	Supply 2.0 V to 5.5 V to V_{DD} and 0 V to V_{SS}.
OSC1 OSC2	16 15	Input Output		Clock input pin Clock output pin	Connect these oscillation pins to ceramic or crystal oscillators for high-frequency clock operation. If the clock is an external input, connect it to OSC1 and leave OSC2 open. The chip will not operate with an external clock when using either the STOP or SLOW modes.
XI XO	18 19	Input Output		Clock input pin Clock output pin	Connect these oscillation pins to crystal oscillators for low-frequency clock operation. If the clock is an external input, connect it to XI and leave XO open. The chip will not operate with an external clock when using the STOP mode. If these pins are not used, connect XI to V_{SS} and leave XO open.
NRST	43	Input	P27	Reset pin [Active low]	This pin resets the chip when power is turned on, is allocated as P27 and contains an internal pull-up resistor. Setting this pin low initializes the internal state of the device. Thereafter, setting the input to high releases the reset. The hardware waits for the system clock to stabilize, then processes the reset interrupt. Also, if "0" is written to P27 and the reset is initiated by software, a low level will be output. The output has an n-channel open-drain configuration. If a capacitor is to be inserted between NRST and V_{DD}, it is recommended that a discharge diode be placed between NRST and V_{DD}.
P00 P01 P02 P06	20 21 22 23	I/O	SBO0, TXD0 SBI0, RXD0 SBT0 BUZZER	I/O port 0	4-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by the P0DIR register. A pull-up resistor for each bit can be selected individually by the P0PLU register. At reset, the input mode is selected and pull-up resistors are disabled (high impedance output).
P10 P11 P12 P13 P14	24 25 26 27 28	I/O	RMOUT, TM7PWM TM7O TM2IO TM3IO TM7IO	I/O port 1	5-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by the P1DIR register. A pull-up resistor for each bit can be selected individually by the P1PLU register. At reset, P11 to P14 are set to input mode and pull-up resistors are disabled (high impedance output), and P10 is set to ourput mode and pull-up resistors are disabled (output "L").

Table 1-3-3 Pin Function Summary (2/5)

Name	No.	I/O	Other Function	Function	Description
P20 P21 P22 P23	29 30 31 32	Input	IRQ0 IRQ1, ACZ IRQ2 IRQ3	Input port 2	4-bit input port. A pull-up resistor for each bit can be selected individually by the P2PLU register. At reset, pull-up resistors are disabled.
P27	43	Input	NRST	I/O port 2	P27 has an n-channel open-drain configuration. When "0" is written and the reset is initiated by software, a low level will be output.
P60 P61 P62 P63 P64 P65 P66 P67	33 34 35 36 37 38 39 40	I/O	KEY0 KEY1 KEY2 KEY3 KEY4 KEY5 KEY6 KEY7	I/O port 6	8-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by the P6DIR register. A pull-up resistor for each bit can be selected individually by the P6PLU register. At reset, P60 to P67 input mode is selected and pull-up resistors are disabled (high impedance output).
P70 P71	41 42	I/O		I/O port 7	2-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by the P7DIR register. A pull-up or pull-down resistor for each bit can be selected individually by the P7PLUD register. However, pull-up and pull-down resistors cannot be mixed. At reset, P70 to P77 input mode is selected and pull-up resistors for P70 to P77 are disabled (high impedance output).
P80 P81 P82 P83 P84 P85 P86 P87	4 3 2 1 48 47 46 45	I/O	LED0 LED1 LED2 LED3 LED4 LED5 LED6 LED7	I/O port 8	8-bit CMOS tri-state I/O port. Each individual bit can be switched to an input or output by the P8DIR register. A pull-up resistor for each bit can be selected individually by the P8PLU register. When configured as outputs, these pins can drive LEDs directly. At reset, P80 to P87 input mode is selected and pull-up resistors for P80 to P87 are disabled (high impedance output).
PA0 PA1 PA2 PA3 PA4 PA5 PA6 PA7	6 7 8 9 10 11 12 13	I/O	A0 A1 A2 A3 A4 A5 A6 A7	input port A	8-bit input port. Each individual bit can be switched to an input or output by the PAPLU resister. A pull-up or pull-down resistor for each bit can be selected individually by the PAPLUD resister. However, pull-up and pull-down resistors cannot be mixed. At reset, the PA0 to PA7 input mode is selected and pull- up resistors are disabled (high impedance output).

Table 1-3-4 Pin Function Summary (3/5)

Name	No.	I/O	Other Function	Function	Description
PC0	5	I/O		I/O port C	1-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by the PCDIR register. A pull-up resistor for each bit can be selected individually by the PCPLU register. At reset, the input mode is selected and pull-up resistors are disabled (high impedance output).
SBO0	20	Output	P00, TXD0	Serial interface transmission data output pins	Transmission data output pins for serial interfaces 0. The output configuration, either CMOS push-pull or n-channel open-drain can be selected. Pull-up resistors can be selected by the P0PLU register. Select output mode by the P0DIR register, and serial data output mode by serial mode register 1 (SC0MD1). This pin can be used as normal I/O pins when the serial interface is not used.
SBI0	21	Input	P01, RXD0	Serial interface reception data input pins	Reception data input pins for serial interfaces 0. Pull-up resistors can be selected by the P0PLU register. Select input mode by the P0DIR register and serial input mode by the serial mode register 1 (SC0MD1). This pin can be used as normal I/O pins when the serial interface is not used.
SBT0	22	I/O	P02	Serial interface clock I/O pins	Clock I/O pins for serial interfaces 0. The output configuration, either CMOS push-pull or n-channel open-drain can be selected. Pull-up resistors can be selected by the P0PLU resister. Select clock I/O for each communication mode by the P0DIR register, and serial mode register 1 (SC0MD1). This pin can be used as normal I/O pins when the serial interface is not used.
TXD0	20	Output	SBO0, P00	UART transmission data output pins	In the serial interface in UART mode, these pins are configured as the transmission data output pins. The output configuration, either CMOS push-pull or n-channel open-drain can be selected. Pull-up resistors can be selected by the P0PLU register. Select output mode by the P0DIR register, and serial data output by serial interface 1 mode register 1 (SC0MD1). This pin can be used as normal I/O pins when the serial interface is not used.

Table 1-3-5 Pin Function Summary (4/5)

Name	No.	I/O	Ohter Function	Function	Description
RXD0	21	Input	SBI0, P01	UART reception data input pin	In the serial interface in UART mode, these pins are configured as the received data input pin. Pull-up resistors can be selected by the P0PLU register. Set this pin to the input mode by the P0DIR register, and to the serial input mode by the serial interface 1 mode register 1 (SC0MD1). This pin can be used as normal I/O pin when the serial interface is not used.
TM2IO TM3IO	26 27	I/O	P12 P13	Timer I/O pins	Event counter clock input pins, timer output and PWM signal output pins for 8-bit timers 2, 3. To use these pins as event clock inputs, configure them as inputs by the P1DIR register. When the pins are used as inputs, pull-up resistors can be specified by the P1PLU register. For timer output, PWM signal output, select the special function pin by the port 1 output mode register (P1OMD) and set to the output mode by the P1DIR register. When not used for timer I/O, these can be used as normal I/O pins.
RMOUT	24	Output	TM7PWM, P10	Remote control transmission signal output pin	Output pin for remote control transmission signal with a carrier signal. For remote control carrier output, select the special function pin by the port 1 output mode register (P1OMD) and set to the output mode by the P1DIR register. Also, set to the remote control carrier output by the remote control carrier output control register (RMCTR). This can be used as a normal I/O pin when remote control is not used.
TM7O	25	Output	P11	Timer output pin	Timer output and PWM signal output pin for 16-bit timer 7. For timer output, PWM signal output, select the special function pin by the port 1 output mode register (P1OMD), and set to the output mode by the P1DIR register. When not used for timer I/O, this can be used as normal I/O pin.
TM7PWM	24	Output	RMOUT, P10	Timer output pin	PWM signal output pin for 16-bit timer 7. For PWM signal output, select the special function pin by the port 1 output mode register (P1OMD), and set to the output mode by the P1DIR register. At the same time, select timer 7 output with the remote comtrol carrier output control register (RMCTR). When not used for timer 7 PWM output pin, this can be used as normal I/O pin.
BUZZER	23	Output	P06	Buzzer output	Piezoelectric buzzer driver pin. The driving frequency can be selected by the DLYCTR register. Select output mode by the P0DIR register and select P06 buzzer output by the DLYCTR register. When not used for buzzer output, this pin can be used as a normal I/O pin.

Table 1-3-6 Pin Function Summary (5/5)

Name	No.	I/O	Other Function	Function	Description
TM7IO	28	I/O	P14	Timer I/O pin	Event counter clock input pin, timer output and PWM signal output pin for 16-bit timer 7. To use this pin as event clock input, configure this as input by the P1DIR register. In the input mode, pull-up resistors can be selected by the P1PLU register. For timer output, PWM signal output, select the special function pin by the port 1 output mode register (P1OMD), and set to the output mode by the P1DIR register. When not used for timer I/O, this can be used as normal I/O pin.
AN0 AN1 AN2 AN3 AN4 AN5 AN6 AN7	6 7 8 9 10 11 12 13	Input	PA0 PA1 PA2 PA3 PA4 PA5 PA6 PA7	Analog input pins	Analog input pins for an 8-channel, 10-bit A/D converter. When not used for analog input, these pins can be used as normal input pins.
IRQ0 IRQ1 IRQ2 IRQ3	29 30 31 32	Input	P20 P21, ACZ P22 P23	External interrupt input pins	External interrupt input pins. The valid edge for IRQ0 to 3 can be selected with the IRQnICR register. IRQ1 is an external interrupt pin that is able to determine AC zero crossings. Both edge for IRQ0, 1 are valid for interrupt. When these are not used for interrupts, these can be used as normal input pins.
ACZ	30	Input	P21, IRQ1	AC zero-cross detection input pin	An input pin for an AC zero-cross detection circuit. The AC zero-cross detection circuit outputs a high level when the input is at an intermediate level. It outputs a low level at all other times. ACZ input signal is connected to the P21 input circuit and the IQR1 interrupt circuit. When the AC zero-cross detection circuit is not used, this pin can be used as a normal P21 input.
KEY0 KEY1 KEY2 KEY3 KEY4 KEY5 KEY6 KEY7	33 34 35 36 37 38 39 40	Input	P60 P61 P62 P63 P64 P65 P66 P67	Key interrupt input pins	Input pins for interrupt based on ORed result of pin inputs. Key input pin for each bits can be selected individually by the key interrupt control register (P6IMD). When not used for KEY input, these pins can be used as normal I/O pins.
MMOD	44	Input		Memory mode switch input pin	Set this pin always to "L" for use. Do not change the setup after reset.

1-4 Block Diagram

1-4-1 Block Diagram

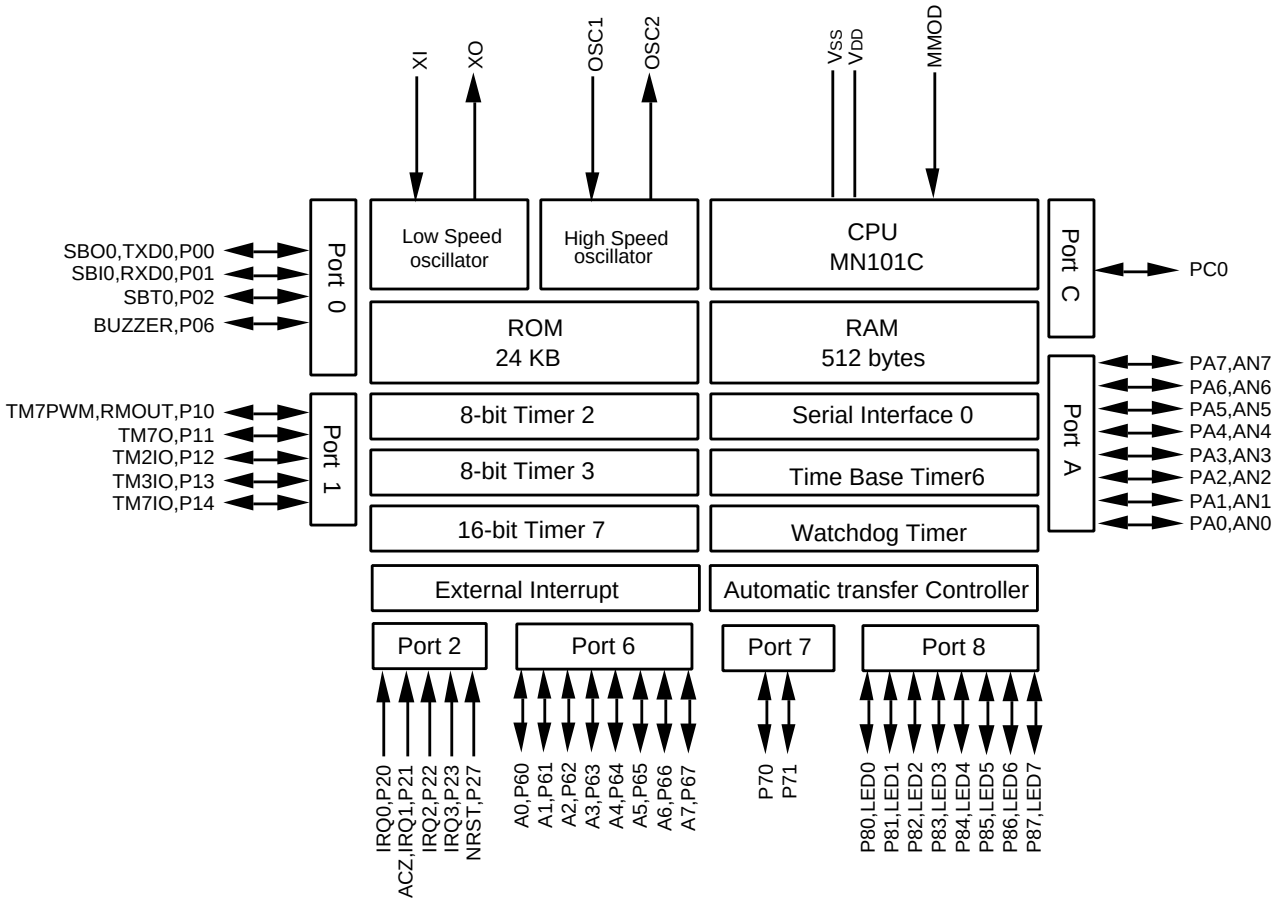


Figure 1-4-1 Block Diagram

1-5 Electrical Characteristics

This LSI user's manual describes the standard specification. System clock (fs) is 1/2 of high speed oscillation at NORMAL mode, or 1/4 of low speed oscillation at SLOW mode. Please ask our sales offices for its own product specifications.

Contents \ Model	MN101C539
Structure	CMOS integrated circuit
Application	General purpose
Function	8-bit single-chip microcontroller

1-5-1 Absolute Maximum Ratings*2,*3

No.	Parameter		Symbol	Rating	Unit
1	Power supply voltage		V_{DD}	- 0.3 to +7.0	V
2	Input clamp current (ACZ)		I_C	- 0.5 to +0.5	mA
3	Input pin voltage		V_I	- 0.3 to V_{DD} +0.3	V
4	Output pin voltage		V_O	- 0.3 to V_{DD} +0.3	
5	I/O pin voltage		V_{IO1}	- 0.3 to V_{DD} +0.3 (except ACZ)	
6	Peak output current	Port 8	I_{OL1} (peak)	30	mA
7		Other than Port 8	I_{OL2} (peak)	20	
8		All pins	I_{OH} (peak)	- 10	
9	Average output current *1	Port 8	I_{OL1} (avg)	20	
10		Other than Port 8	I_{OL2} (avg)	15	
11		All pins	I_{OH} (avg)	- 5	
12	Power dissipation		P_D	400 (T_a=85 °C)	mW
13	Operating ambient temperature		T_{opr}	- 40 to +85 (EPROM version (-20 to +85))	°C
14	Storage temperature		T_{stg}	- 55 to +125	

*1 Applied to any 100 ms period.

*2 Connect at least one bypass capacitor of 0.1 µF or larger between the power supply pin and the ground for latch-up prevention.

*3 The absolute maximum ratings are the limit values beyond which the LSI may be damaged and proper operation is not assured.

1-5-2 Operating Conditions [NORMAL mode : fs=fosc/2, SLOW mode : fs=fx/4]

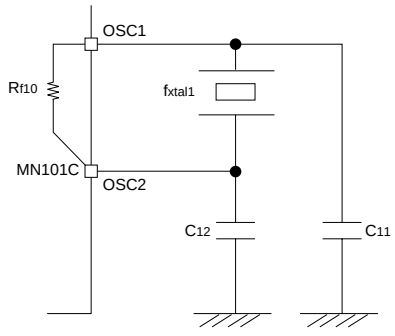
Ta=-40 °C to +85 °C (-20 °C to +85 °C) V_{DD}=2.0 V (2.7 V) to 5.5 V V_{SS}=0 V
EPROM vers. is in ( ).

	Parameter	Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
Power supply voltage							
1	Power supply voltage	V_{DD1}	fosc≤20.0 MHz	4.5	-	5.5	V
2		V_{DD2}	fosc≤8.39 MHz	2.7	-	5.5	
3		V_{DD3}	fosc≤4.00 MHz fs=fosc/4	2.0 (2.7)	-	5.5	
4		V_{DD4}	fx=32.768 kHz	2.0 (2.7)	-	5.5	
5	Voltage to maintain RAM data	V_{DD5}	During STOP mode	1.8	-	5.5	
Operation speed *1							
6	Minimum instruction execution time	tc1	V_{DD}=4.5 V to 5.5 V	0.100	-	-	µs
7		tc2	V_{DD}=2.7 V to 5.5V	0.238	-	-	
8		tc3	V_{DD}=2.0 V (2.7 V) to 5.5 V	1.00	-	-	
9		tc4	V_{DD}=2.0 V (2.7 V) to 5.5 V	-	125	-	

*1 tc1, tc2, tc3 : 1/2 of high speed oscillation at NORMAL mode
tc4 : 1/4 of low speed oscillation at SLOW mode

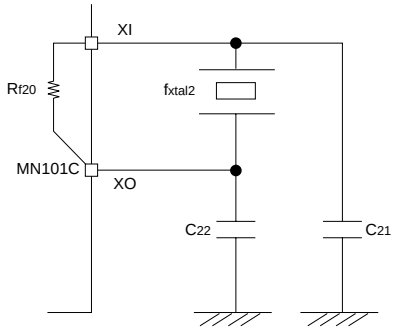
Ta=-40 °C to +85 °C (-20 °C to +85 °C) V_{DD}=2.0 V (2.7 V) to 5.5 V V_{SS}=0 V
EPROM vers. is in ( ).

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
Crystal oscillator 1 Fig. 1-5-1							
10	Crystal frequency	f_{xtal1}	V_{DD}=4.5 V to 5.5 V	1.0	-	20.0	MHz
11	External capasitors	C_{11}		-	20	-	pF
12		C_{12}		-	20	-	
13	Internal feedback resistor	R_{f10}		-	700	-	kΩ
Crystal oscillator 2 Fig. 1-5-2							
14	Crystal frequency	f_{xtal2}		-	32.768	-	kHz
15	External capasitors	C_{21}		-	20	-	pF
16		C_{22}		-	20	-	
17	Internal feedback resistor	R_{f20}		-	4.0	-	MΩ



The feedback resistor is built-in.

Figure 1-5-1 Crystal Oscillator 1



The feedback resistor is built-in.

Figure 1-5-2 Crystal Oscillator 2

Connect external capacitors that suits the used pin. When crystal oscillator or ceramic oscillator is used, the frequency is changed depending on the condenser rate. Therefore, consult the manufacturer of the pin for the appropreate external capacitor.

Ta=-40 °C to +85 °C (-20 °C to +85 °C) V_{DD}=2.0 V (2.7 V) to 5.5 V V_{SS}=0 V
EPROM vers. is in ( ).

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
External clock input 1 OSC1 (OSC2 is unconnected)							
18	Clock frequency	fosc		1.0	-	20.0	MHz
19	High level pulse width	twh1	*2 Fig. 1-5-3	20.0	-	-	ns
20	Low level pulse width	twl1		20.0	-	-	
21	Rising time	twr1	Fig. 1-5-3	-	-	5.0	
22	Falling time	twf1		-	-	5.0	
External clock input 2 XI (XO is unconnected)							
23	Clock frequency	fx		32.768	-	100	kHz
24	High level pulse width	twh2	*2 Fig. 1-5-4	3.5	-	-	µs
25	Low level pulse width	twl2		3.5	-	-	
26	Rising time	twr2	Fig. 1-5-4	-	-	20	ns
27	Falling time	twf2		-	-	20	

*2 The clock duty rate in the standard mode should be 45% to 55%.

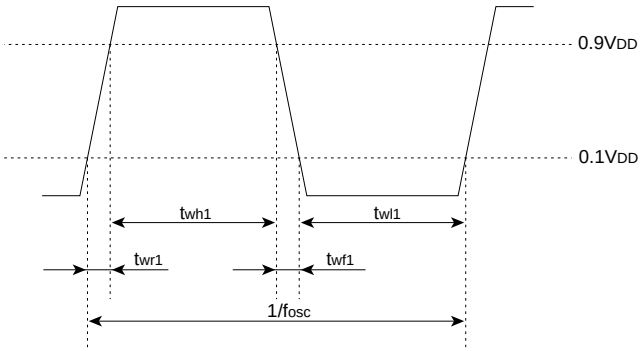


Figure 1-5-3 OSC1 Timing Chart

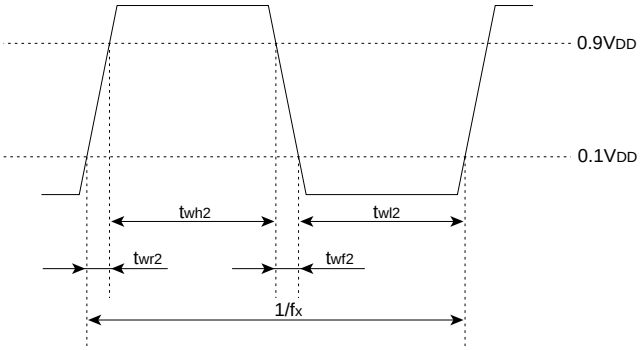


Figure 1-5-4 XI Timing Chart

1-5-3 DC Characteristics

Ta=-40 °C to +85 °C (-20 °C to +85 °C) V_{DD}=2.0 V (2.7 V) to 5.5 V V_{SS}=0 V
EPROM vers. is in ( ).

	Parameter	Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
Power supply current (no load at output pin) *1							
1	Power supply current	I_{DD1}	fosc=20.0 MHz V_{DD}=5 V fs=fosc/2	-	20	50	mA
2		I_{DD2}	fosc=8.39 MHz V_{DD}=5 V fs=fosc/2	-	10	20	
3		I_{DD3}	fx=32.768 kHz V_{DD}=3 V fs=fx/4	-	20	70	µA
4	Supply current during HALT mode	I_{DD4}	fx=32.768 kHz V_{DD}=3 V Ta=25 °C	-	2	6	
5		I_{DD5}	fx=32.768 kHz V_{DD}=3 V Ta=-40 °C to +85 °C (Ta=--20 °C to +70 °C)	-	-	15	
6	Supply current during STOP mode	I_{DD6}	V_{DD}=5 V Ta=25 °C	-	0	2	
7		I_{DD7}	V_{DD}=5 V Ta=-40 °C to +85 °C (Ta=--20 °C to +85 °C)	-	-	20	

*1 Measured under conditions of no load.

- The supply current during operation, I_{DD1}(I_{DD2}), is measured under the following conditions : After all I/O pins are set to input mode and the oscillation is set to <NORMAL mode>, the MMOD pin is at V_{SS} level, the input pins are at V_{DD} level, and a 20 MHz (8.39 MHz) square wave of V_{DD} and V_{SS} amplitudes is input to the OSC1 pin.
- The supply current during operation, I_{DD3}, is measured under the following conditions : After all I/O pins are set to input mode and the oscillation is set to <SLOW mode>, the MMOD pin is at V_{SS} level, the input pins are at V_{DD} level, and a 32.768 kHz square wave of V_{DD} and V_{SS} amplitudes is input to the XI pin.
- The supply current during HALT mode, I_{DD4}(I_{DD5}), is measured under the following conditions : After all I/O pins are set to input mode and the oscillation is set to <HALT mode>, the MMOD pin is at V_{SS} level, the input pins are at V_{DD} level, and an 32.768 kHz square wave of V_{DD} and V_{SS} amplitudes is input to the XI pin.
- The supply current during STOP mode, I_{DD6}(I_{DD7}), is measured under the following conditions : After the oscillation is set to <STOP mode>, the MMOD pin is at V_{SS} level, the input pins are at V_{DD} level, and the OSC1 and XI pins are unconnected.

Ta=-40 °C to +85 °C (-20 °C to +85 °C) V_{DD}=2.0 V (2.7 V) to 5.5 V V_{SS}=0 V
EPROM vers. is in ( ).

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
Input pin 1 MMOD (Schmitt trigger input)							
8	Input high voltage	V_{IH1}		0.8 V_{DD}	-	V_{DD}	V
9	Input low voltage	V_{IL1}		V_{SS}	-	0.2 V_{DD}	
10	Input leakage current	I_{LK1}	V_I=0 V to V_{DD}	-	± 0.01	± 2.0	µA
Input pin 2 P20, P22, P23 (Schmitt trigger input)							
11	Input high voltage	V_{IH2}		0.8 V_{DD}	-	V_{DD}	V
12	Input low voltage	V_{IL2}		V_{SS}	-	0.2 V_{DD}	
13	Input leakage current	I_{LK2}	V_I=0 V to V_{DD}	-	± 0.01	± 2.0	µA
14	Input high current	I_{IH2}	V_{DD}=5.0 V V_I=1.5 V Pull-up resistor ON	-50	-140	-200	
Input pin 3-1 P21							
15	Input high voltage	V_{IH3}		0.8 V_{DD}	-	V_{DD}	V
16	Input low voltage	V_{IL3}		V_{SS}	-	0.2 V_{DD}	
17	Input leakage current	I_{LK3}	V_I=0 V to V_{DD}	-	± 0.01	± 2.0	µA
18	Input high current	I_{LK3}	V_{DD}=5.0 V V_I=1.5 V Pull-up resistor ON	-50	-140	-200	
Input pin 3-2 P21(at used as ACZ)							
19	High detection voltage	V_{DLH}	Fig. 1-5-5	V_{SS}	-	V_{DD}-1.5	V
20		V_{DHL}		1.5	-	V_{DD}	
21	Low detection voltage	V_{DHH}		V_{DD}-0.5	-	V_{DD}	
22		V_{DLL}		V_{SS}	-	0.5	
23	Input leakage current	I_{LK10}	V_I=0 V to V_{DD}	-	± 0.01	± 2.0	µA
24	Input clamp current	I_{C10}	V_I>V_{DD} V_I<0 V	-	-	± 500	
ACZ pins							
25	Rising time	trs	Fig. 1-5-5	30	-	-	µs
26	Falling time	tfs		30	-	-	

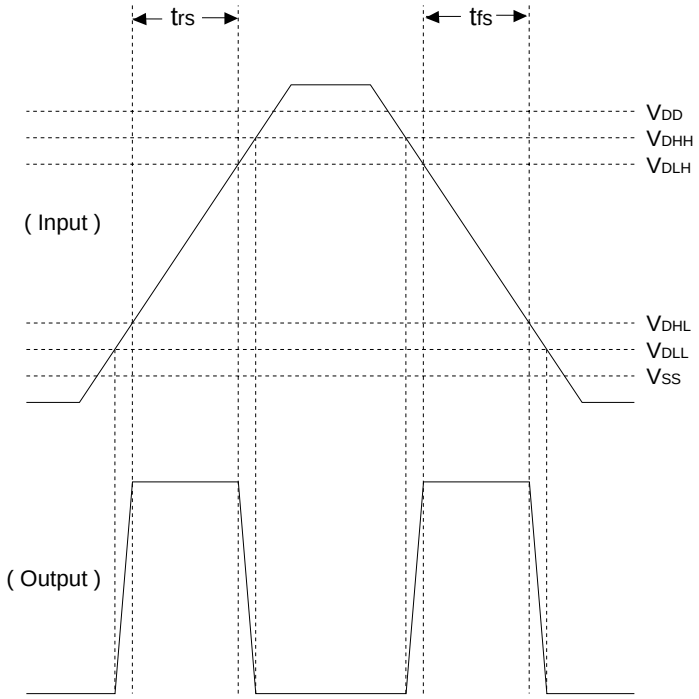


Figure 1-5-5 AC Zero-Cross Detector

Ta=-40 °C to +85 °C (-20 °C to +85 °C) V_{DD}=2.0 V (2.7 V) to 5.5 V V_{SS}=0 V
EPROM vers. is in ( ).

Parameter		Symbol	Conditions	ting			Unit
				MIN	TYP	MAX	
Input pin 4 P27 (NRST)							
27	Input high voltage	V_{IH4}		0.8 V_{DD}	-	V_{DD}	V
28	Input low voltage	V_{IL4}		V_{SS}	-	0.2 V_{DD}	
29	Input low current	I_{IL4}	V_{DD}=5.0 V V_I =1.5 V Pull-up resistor ON	-50	-140	-200	µA
I/O pin 5 P00 to P02, P06, P11 to P14, PC0 (Schmitt trigger input)							
30	Input high voltage	V_{IH5}		0.8 V_{DD}	-	V_{DD}	V
31	Input low voltage	V_{IL5}		V_{SS}	-	0.2 V_{DD}	
32	Input leakage current	I_{LK5}	V_I =0 V to V_{DD}	-	± 0.01	± 2.0	µA
33	Input high current	I_{IH5}	V_{DD}=5.0 V V_I =1.5 V Pull-up resistor ON	-50	-140	-200	
34	Output high voltage	V_{OH5}	V_{DD}=5.0 V I_{OH}=-0.5 mA	4.5	-	-	V
35	Output low voltage	V_{OL5}	V_{DD}=5.0 V I_{OL}=1.0 mA	-	0.1	0.5	
I/O pin 6 P60 to P67 (Schmitt trigger input)							
36	Input high voltage	V_{IH6}		0.8 V_{DD}	-	V_{DD}	V
37	Input low voltage	V_{IL6}		V_{SS}	-	0.2 V_{DD}	
38	Input leakage current	I_{LK6}	V_I =0 V to V_{DD}	-	± 0.01	± 2.0	µA
39	Input high current	I_{IH6}	V_{DD}=5.0 V V_I =1.5 V Pull-up resistor ON	-50	-140	-200	
40	Output high voltage	V_{OH6}	V_{DD}=5.0 V I_{OH}=-0.5 mA	4.5	-	-	V
41	Output low voltage	V_{OL6}	V_{DD}=5.0 V I_{OL}=1.0 mA	-	0.1	0.5	
I/O pin 7 P70 to P71, PA0 to PA7 (Schmitt trigger input)							
42	Input high voltage	V_{IH7}		0.8 V_{DD}	-	V_{DD}	V
43	Input low voltage	V_{IL7}		V_{SS}	-	0.2 V_{DD}	
44	Input leakage current	I_{LK7}	V_I =0 V to V_{DD}	-	± 0.01	± 2.0	µA
45	Input high current	I_{IH7}	V_{DD}=5.0 V V_I =1.5 V Pull-up resistor ON	-50	-140	-200	
46	Input low current	I_{IL7}	V_{DD}=5.0 V V_I =1.5 V Pull-up resistor ON	30	100	300	V
47	Output high voltage	V_{OH7}	V_{DD}=5.0 V I_{OH}=-0.5 mA	4.5	-	-	
48	Output low voltage	V_{OL7}	V_{DD}=5.0 V I_{OL}=1.0 mA	-	0.1	0.5	

Ta=-40 °C to +85 °C (-20 °C to +85 °C) V_{DD}=2.0 V (2.7 V) to 5.5 V V_{SS}=0 V
EPROM vers. is in ( ).

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
Input pin 8 P80 to P87							
49	Input high voltage 1	V_{IH8}		0.8 V_{DD}	-	V_{DD}	V
50	Input high voltage 2	V_{IH9}	V_{DD}=4.5 V to 5.5V	0.7 V_{DD}	-	V_{DD}	
51	Input low voltage 1	V_{IL8}		V_{SS}	-	0.2 V_{DD}	
52	Input low voltage 1	V_{IL9}	V_{DD}=4.5 V to 5.5V	V_{SS}	-	0.3 V_{DD}	
53	Input leakage current	I_{LK8}	V_I =0 V to V_{DD}	-	± 0.01	± 2.0	µA
54	Input high current	I_{IH8}	V_{DD}=5.0 V V_I =1.5 V Pull-up resistor ON	-50	-140	-200	
55	Output high voltage	V_{OH8}	V_{DD}=5.0 V I_{OH}=-0.5 mA	4.5	-	-	V
56	Output low voltage	V_{OL8}	V_{DD}=5.0 V I_{OL}=15 mA	-	0.4	1.0	
I/O pin 9 P10							
57	Input high voltage	V_{IH10}		0.8 V_{DD}	-	V_{DD}	V
58	Input low voltage	V_{IL10}		V_{SS}	-	0.2 V_{DD}	
59	Input leakage current	I_{LK9}	V_I =0 V to V_{DD}	-	± 0.01	± 2.0	µA
60	Input high current	I_{IH9}	V_{DD}=5.0 V V_I =1.5 V Pull-up resistor ON	-50	-140	-200	
61	Output high voltage	V_{OH9}	V_{DD}=5.0 V I_{OH}=-3.3 mA	4.5	-	-	V
62	Output low voltage	V_{OL9}	V_{DD}=5.0 V I_{OL}=3.5 mA	-	-	0.5	

1-5-4 A/D Converter Characteristics

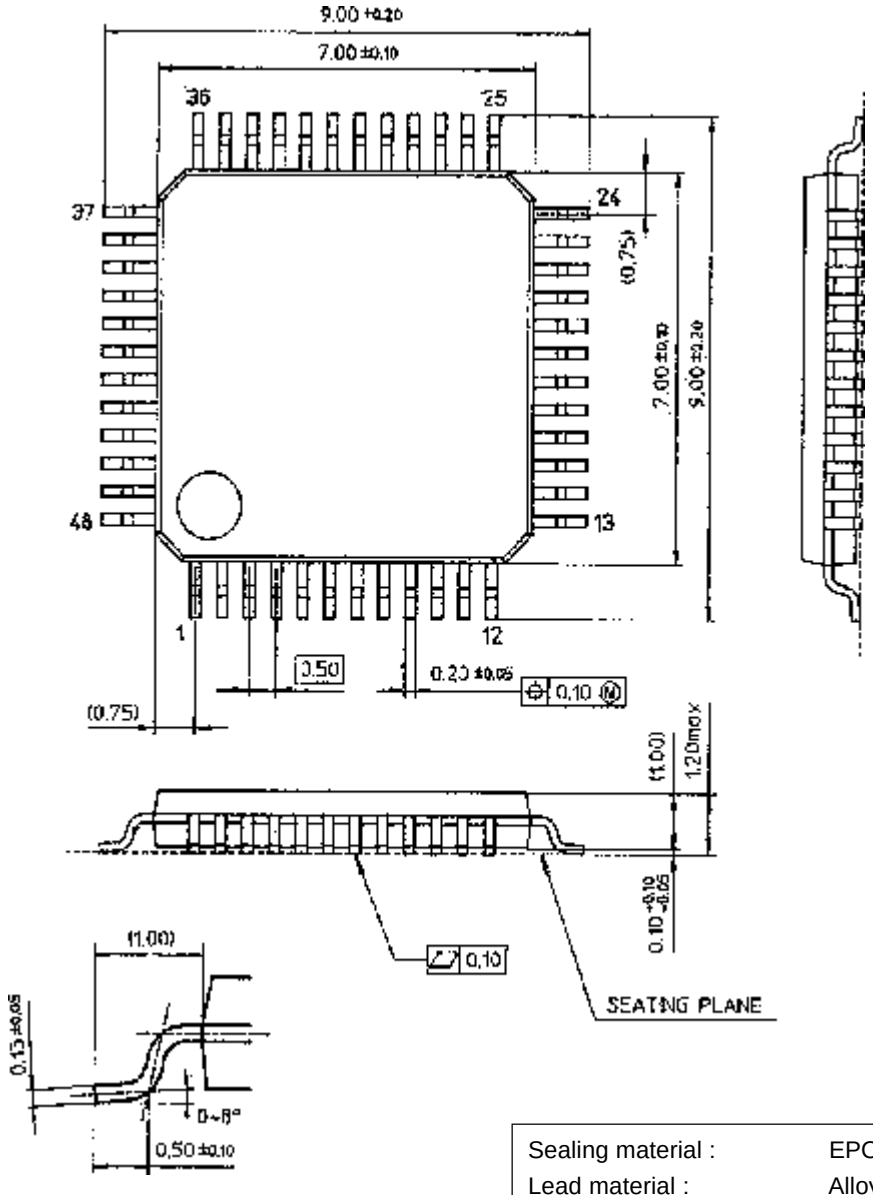
Ta=-40 °C to +85 °C (-20 °C to +85 °C) V_{DD}=2.0 V (2.7 V) to 5.5 V V_{SS}=0 V
EPROM vers. is in ( ).

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
1	Resolution			-	-	10	Bits
2	Non-linearity error 1		V_{DD} =5.0 V V_{SS}=0 V T_{AD}=800 ns	-	-	± 3	LSB
3	Differential non-linearity error 1			-	-	± 3	
4	Non-linearity error 2		V_{DD} =5.0 V V_{SS}=0 V f_{OSC}=32.768 kHz	-	-	± 5	
5	Differential non-linearity error 2			-	-	± 5	
6	Zero transition voltage	V_{DT}	V_{DD} =5.0 V V_{SS}=0 V T_{AD}=800 ns	-	30	100	mV
7	Full-scale transition voltage	V_{FST}		-	30	100	
8	A/D conversion time		T_{AD}=800 ns	9.6	-	-	µs
9			fx=32.768 kHz	-	-	183	
10	Sampling time		f_{OSC}=8 MHz	1.0	-	36	
11			fx=32.768 kHz	-	30.5	-	
12	Analog input voltage	V_{IA}		V_{SS}	-	V_{DD}	V
13	Analog input leakage current	I_{LIA}	V_{IA}=V_{SS} to V_{DD} unselected channel	-	± 0.01	± 2.0	µA

1-6 Package Dimension

Package Code : TQFP048-P-0707B

Units : mm



Sealing material :	EPOXY resin
Lead material :	Alloy of Cu
Lead surface processing	: Pd plate

Figure 1-6-1 48-Pin TQFP

> The package dimension is subjected to change. Before using this product, please obtain product specifications from the sales office.

1-7 Precautions

1-7-1 General Usage

■Connection of V_{DD} pin, and V_{SS} pin

All V_{DD} pins should be connected directly to the power supply and all Vss pins should be connected to ground in the external. Please consider the LSI chip orientation before mounting it on the printed circuit board. Incorrect connection may lead a fusion and break a micro controller.

■Cautions for Operation

(1) If you install the product close to high-field emissions (under the cathode ray tube, etc), shield the package surface to ensure normal performance.

(2) Each model has different operating condition,

-Operation temperature should be well considered. For example, if temperature is over the operating condition, its operation may be executed wrongly.

-Operation voltage should be also well considered. If the operation voltage is over the operation range, it can be shortened the length of its life. If the operation voltage is below the operating range, its operation may be executed wrongly.

1-7-2 Unused Pins

■Unused Pins (only for input)

Insert 10 kΩ to 100 kΩ resistor to unused pins (only for input) for pull-up or pull-down. If the input is unstable, Pch transistor and Nch transistor of input inverter are on, and through current goes to the input circuit. That increases current consumption and causes power supply noise.

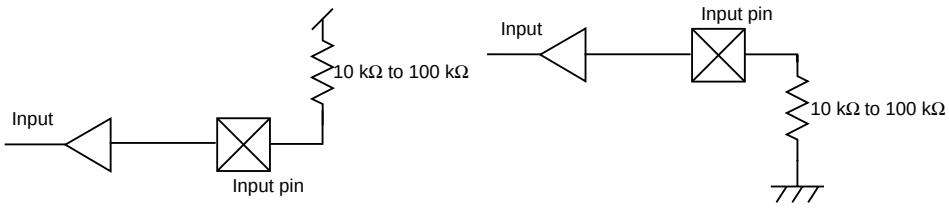


Figure 1-7-1 Unused Pins (only for input)

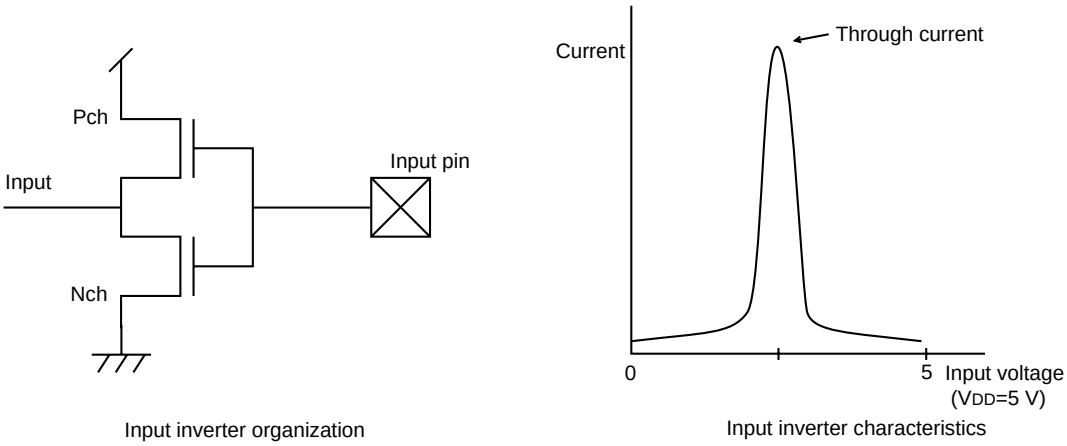


Figure 1-7-2 Input Inverter Organization and Characteristics

■Unused pins (for I/O)

Unused I/O pins should be set according to pins' condition at reset. If the output is high impedance (Pch / Nch transistor : output off) at reset, to stabilize input, set 10 kΩ to 100 kΩ resistor to be pull-up or pull-down. If the output is on at reset, set them open.

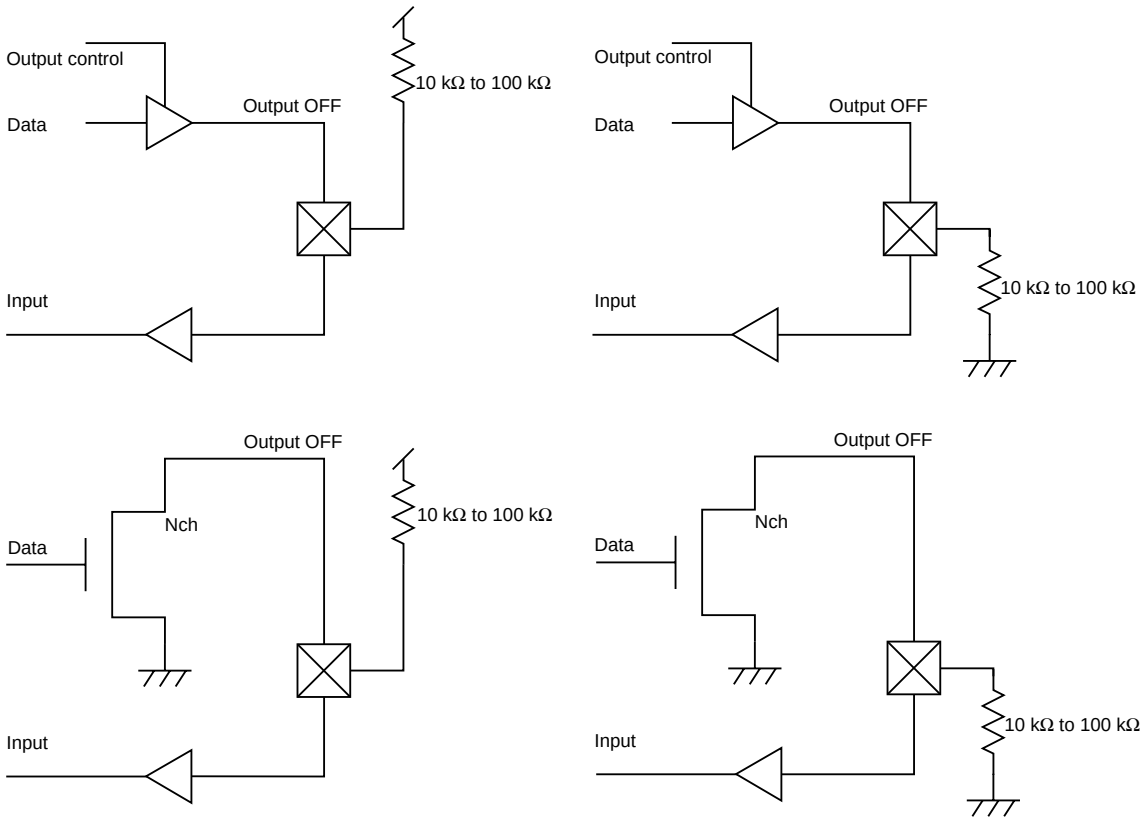


Figure 1-7-3 Unused I/O pins (high impedance output at reset)

1-7-3 Power Supply

■The Relation between Power Supply and Input Pin Voltage

Input pin voltage should be supplied only after power supply is on. If the input pin voltage is applied supplies before power supply is on, a latch up occurs and causes the destruction of micro controller by a large current flow.

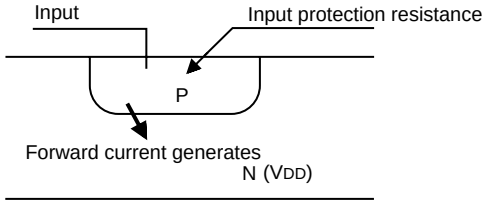
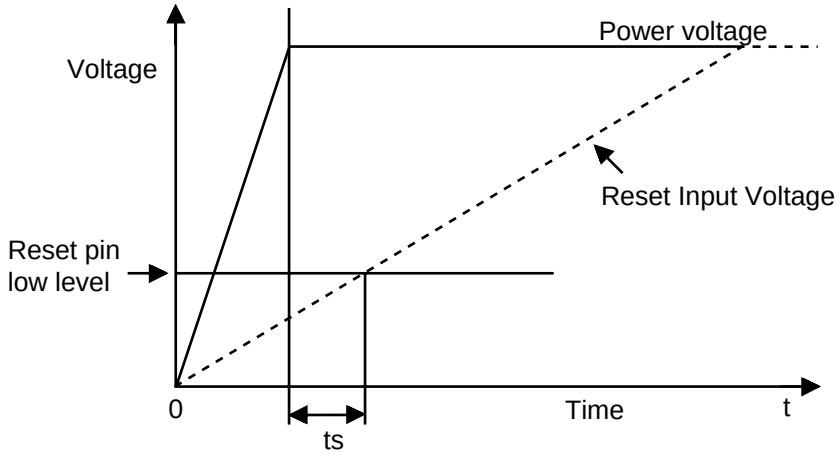


Figure 1-7-4 Power Supply and Input Pin Voltage

■The Relation between Power Supply and Reset Input Voltage

After power supply is on, reset pin voltage should be low for sufficient time, ts, before rising , in order to be recognized as a reset signal.



[☞ Chapter 2. 2-6-1 Reset Operation]

Figure 1-7-5 Power Supply and Reset Input Voltage

1-7-4 Power Supply Circuit

■Cautions for Setting Power Supply Circuit

The CMOS logic microcontroller is high speed and high density. So, the power circuit should be designed, taking into consideration of AC line noise, ripple caused by LED driver. Figure 1-7-6 shows an example for emitter follower type power supply circuit.

■An example for Emitter Follower Type Power Supply Circuit

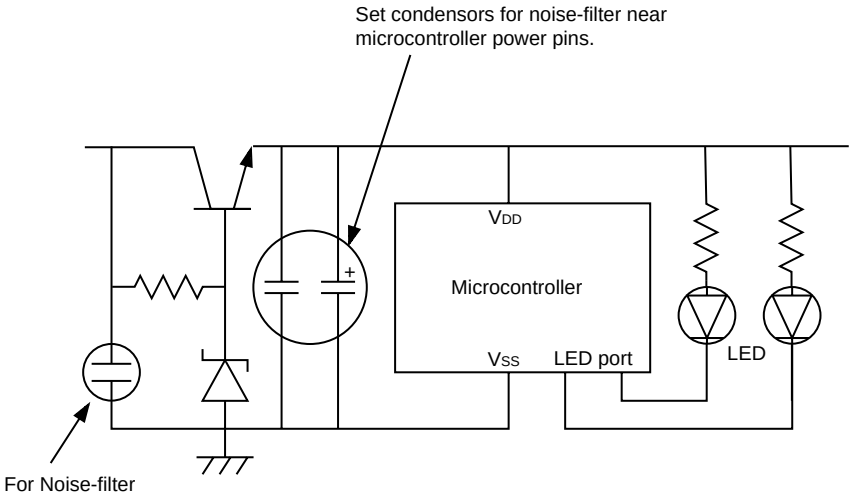


Figure 1-7-6 An Example for Emitter follower type Power Supply Circuit

1-7-5 Oscillators

This LSI's oscillation clock can be used with a ceramic or crystal oscillator.

■Recomended Oscillators
Figure 1-7-7 shows basic configuration connected with a ceramic oscillator.

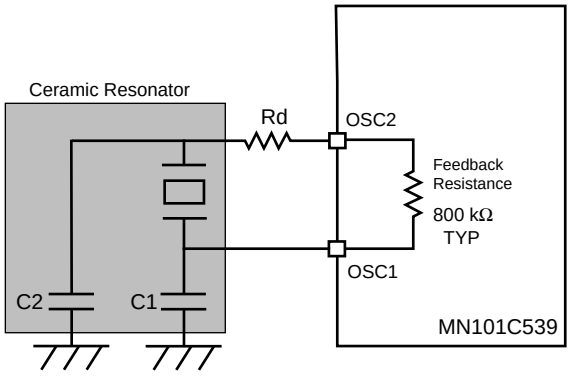


Figure 1-7-7 Basic Configuration Connected with a Ceramic Oscillator

After evaluate the actual oscillating on the target board, dumping resistance may be set, if necessary.

We do not evaluate oscillating of crystal oscillator on this LSI. Set the circuit constant as is recommendation of the oscillator manufacturer.

Circuit constant of each ceramic or crystal oscillator, which is connected to OSC1/OSC2 or XI/XO, differs depending on stray capacitance of the oscillator or the mounting circuit. So consult the oscillator manufacturer for the appropreate circuit constant.

Chapter 2 CPU Basics

2-1 Overview

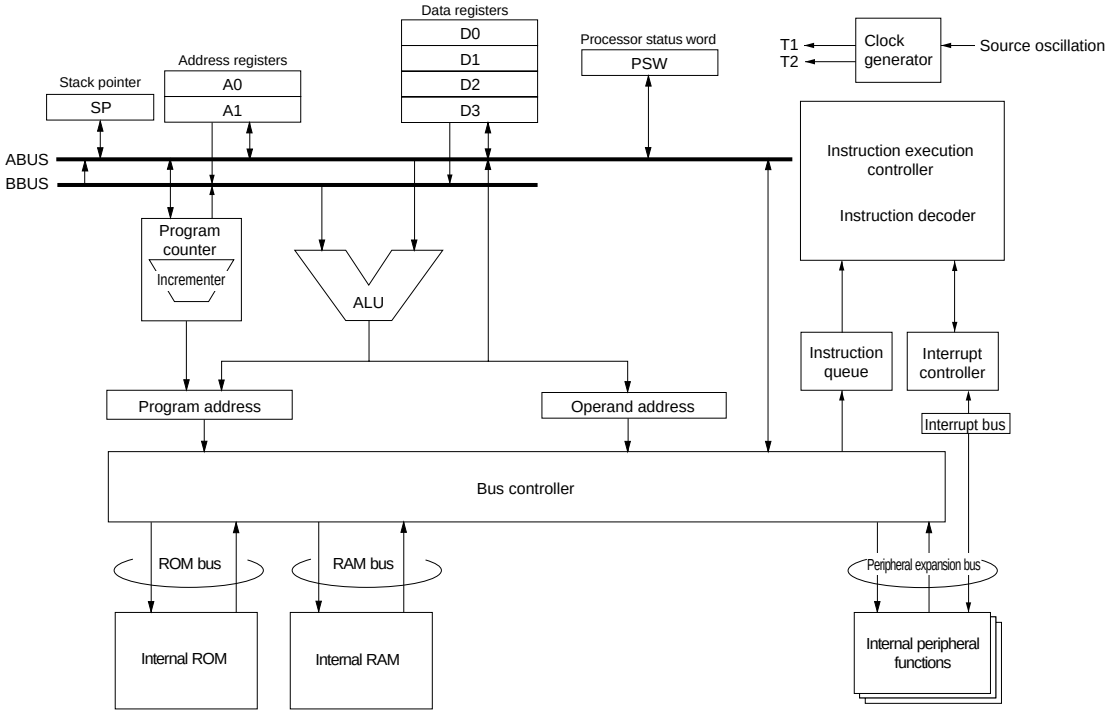
The MN101C CPU has a flexible optimized hardware configuration. It is a high speed CPU with a simple and efficient instruction set. Specific features are as follows:

1. Minimized code sizes with instruction lengths based on 4-bit increments
   The series keeps code sizes down by adopting a basic instruction length of one byte and variable instruction lengths based on 4-bit increments.
2. Minimum execution instruction time is one system clock cycle.
3. Minimized register set that simplifies the architecture and supports C language
   The instruction set has been determined, depending on the size and capacity of hardware, after an analysis of embedded application programing code and creation code by C language compiler. Therefore, the set is simple instruction using the minimal register set required for C language compiler. [☞" MN101C LSI User's Manual" (Architecture Instructions)]

Table 2-1-1 Basic Specifications

Structure	Load / store architecture	
	Six registers	Data : 8-bit x 4 Address : 16-bit x 2
	Other	PC : 19-bit PSW : 8-bit SP : 16-bit
Instructions	Number of instructions	37
	Addressing modes	9
	Instruction length	Basic portion : 1 byte (min.) Extended portion : 0.5-byte x n (0≤n≤9)
Basic performance	Internal operation frequency (max)	10 MHz
	Instruction execution	Min. 1 cycle
	Inter-register operation	Min. 2 cycles
	Load / store	Min. 2 cycles
	Conditional branch	2 to 3 cycles
Pipeline	3-stage (instruction fetch, decode, execution)	
Address space	256 KB (max. 64 KB for data)	[☞ 2-2 Memory space]
External bus	Address	18-bit (max.)
	Data	8-bit
	Minimum bus cycle	1 system clock cycle
Interrupt	Vector interrupt	3 interrupt levels
Low-power dissipation mode	STOP mode	
	HALT mode	

2-1-1 Block Diagram



Clock generator	Uses a clock oscillator circuit driven by an external crystal or ceramic resonator to supply clock signals to CPU blocks.
Program counter	Generates addresses for the instructions to be inserted into the instruction queue. Normally incremented by sequencer indication, but may be set to branch destination address or ALU operation result when branch instructions or interrupts occur.
Instruction queue	Stores up to 2 bytes of pre-fetched instructions.
Instruction decoder	Decodes the instruction queue, sequentially generates the control signals needed for instruction execution, and executes the instruction by controlling the blocks within the chip.
Instruction execution controller	Controls CPU block operations in response to the result decoded by the instruction decoder and interrupt requests.
ALU	Executes arithmetic operations, logic operations, shift operations, and calculates operand addresses for register relative indirect addressing mode.
Internal ROM, RAM	Assigned to the execution program, data and stack region.
Address register	Stores the addresses specifying memory for data transfer. Stores the base address for register relative indirect addressing mode.
Data register	Holds data for operations. Two 8-bit registers can be connected to form a 16-bit register.
Interrupt controller	Detects interrupt requests from peripheral functions and requests CPU shift to interrupt processing.
Bus controller	Controls connection of CPU internal bus and CPU external bus. Includes bus usage arbitration function.
Internal peripheral functions	Includes peripheral functions (timer, serial interface, A/D converter, D/A converter, etc.). Peripheral functions vary with model.

Figure 2-1-1 Block Diagram and Function

2-1-2 CPU Control Registers

This LSI locates the peripheral circuit registers in memory space (x'03F00' to x'03FFF') with memory-mapped I/O. CPU control registers are also located in this memory space.

Table 2-1-2 CPU Control Registers

Registers	Address	R/W	Function	Pages
CPUM	x'03F00'	R/W *1	CPU mode control register	II - 19,23
MEMCTR	x'03F01'	R/W	Memory control register	II - 16
OSCMD	x'03F0D'	R/W	Oscillation frequency control register	II - 23
Reserved	x'03FE0'	-	For debugger	-
NMICR	x'03FE1'	R/W	Non - maskable interrupt control register [☞ Chapter 3]	III - 16
xxxICR	x'03FE2' to x'03FFA'	R/W	Maskable interrupt control register [☞ Chapter 3]	III - 17 to 29
Reserved	x'03FFF'	-	Reserved (For reading interrupt vector data on interrupt process)	-

R/W : Readable / Writable

*1 a part of register is only readable

2-1-3 Instruction Execution Controller

The instruction execution controller consists of four blocks: memory, instruction queue, instruction registers, and instruction decoder.

Instructions are fetched in 1-byte units, and temporarily stored in the 2-byte instruction queue. Transfer is made in 1-byte or half-byte units from the instruction queue to the instruction register to be decoded by the instruction decoder.

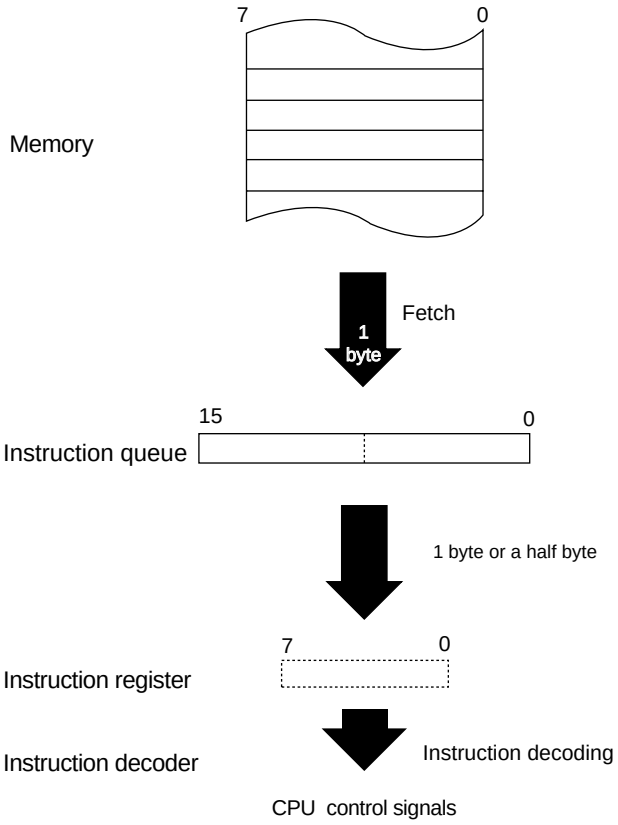


Figure 2-1-2 Instruction Execution Controller Configuration

2-1-4 Pipeline Process

Pipeline process means that reading and decoding are executed at the same time on different instructions, then instructions are executed without stopping. Pipeline process makes instruction execution continual and speedy. This process is executed with instruction queue and instruction decoder.

Instruction queue is buffer that fetches the second instruction in advance. That is controlled to fetch the next instruction when instruction queue is empty at each cycle on execution. At the last cycle of instruction execution, the first word (operation code) of executed instruction is stored to instruction register. At that time, the next operand or operation code is fetched to instruction queue, so that the next instruction can be executed immediately, even if register direct (da) or immediate (imm) is needed at the first cycle of the next instruction execution. But on some other instruction such as branch instruction, instruction queue becomes empty on the time that the next operation code to be executed is stored to instruction register at the last cycle. Therefore, only when instruction queue is empty, and direct address (da) or immediate data (imm) are needed, instruction queue keeps waiting for a cycle.

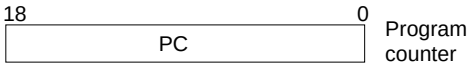
Instruction queue is controlled automatically by hardware so that there is no need to be controlled by software. But when instruction execution time is estimated, operation of instruction queue should be into consideration. Instruction decoder generates control signal at each cycle of instruction execution by micro program control. Instruction decoder uses pipeline process to decode instruction queue at one cycle before control signal is needed.

2-1-5 Registers for Address

Registers for address include program counter (PC), address registers (A0, A1), and stack pointer (SP).

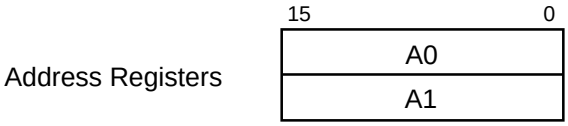
■Program Counter (PC)

This register gives the address of the currently executing instruction. It is 19 bits wide to provide access to a 256 KB address space in half byte(4-bit increments). The LSB of the program counter is used to indicate half byte instruction. The program counter after reset is stored from the value of vector table at the address of 4000.



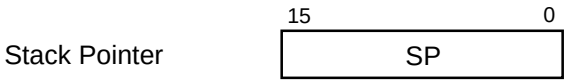
■Address Registers (A0, A1)

These registers are used as address pointers specifying data locations in memory. They support the operations involved in address calculations (i.e. addition, subtraction and comparison). Those pointers are 2 bytes data. Transfers between these registers and memory are always in 16-bit units. Either odd or even address can be transferred. At reset, the value of address register is undefined.



■Stack Pointer (SP)

This register gives the address of the byte at the top of the stack. It is decremented during push operations and incremented during pop operations. Ar reset, the value of SP is undefined.



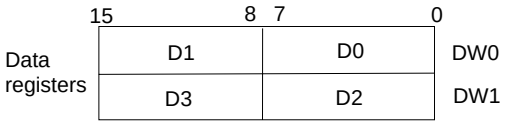
2-1-6 Registers for Data

Registers for data include four data registers (D0, D1, D2, D3).

■Data Registers (D0, D1, D2, D3)

Data registers D0 to D3 are 8-bit general-purpose registers that support all arithmetic, logical and shift operations. All registers can be used for data transfers with memory.

The four data registers may be paired to form the 16-bit data registers DW0 (D0+D1) and DW1 (D2+D3). At reset, the value of Dn is undefined.



2-1-7 Processor Status Word

Processor status word (PSW) is an 8-bit register that stores flags for operation results, interrupt mask level, and maskable interrupt enable. PSW is automatically pushed onto the stack when an interrupt occurs and is automatically popped when return from the interrupt service routine.

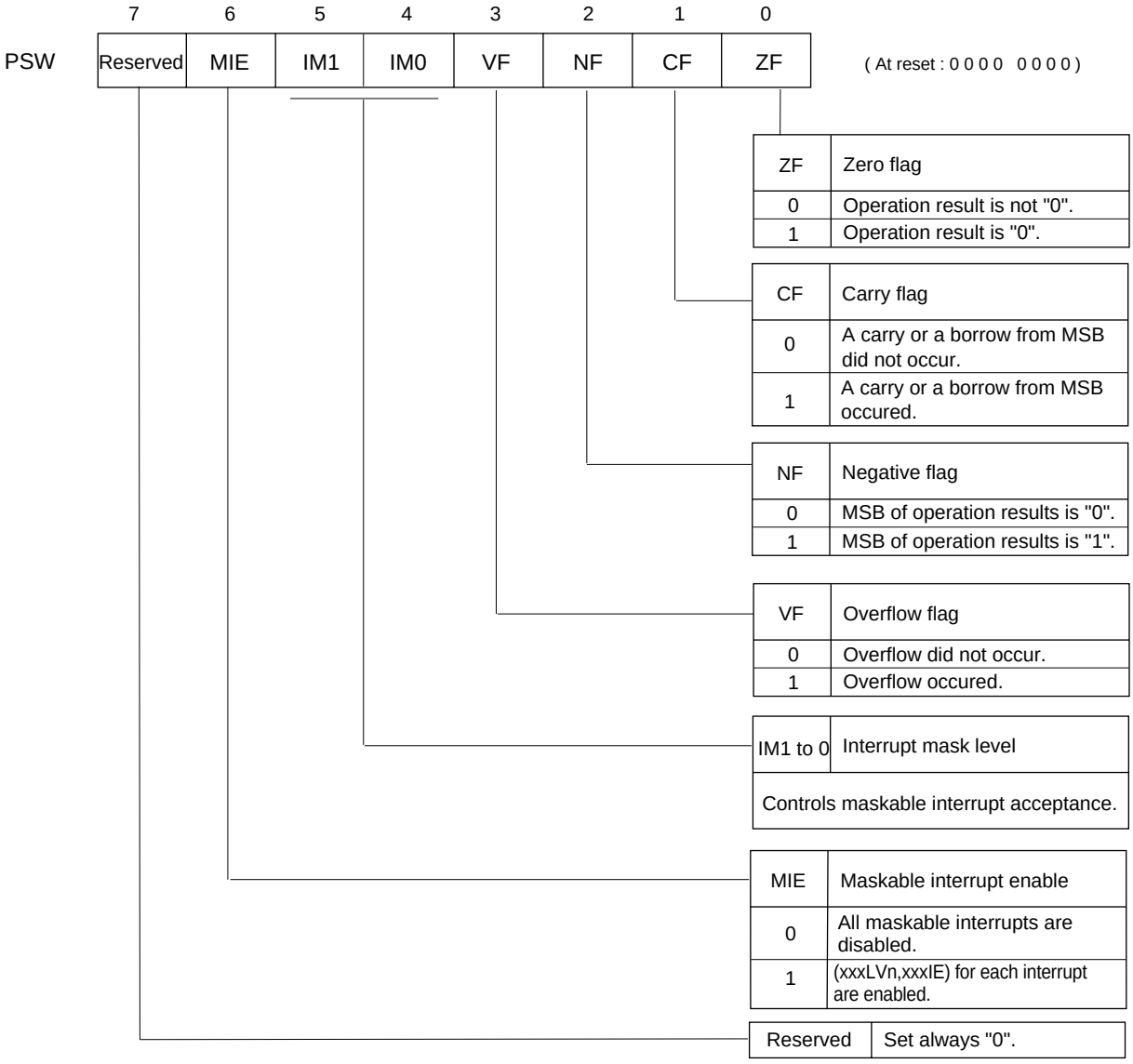


Figure 2-1-3 Processor Status Word(PSW)

■Zero Flag (ZF)
Zero flag (ZF) is set to "1", when all bits are '0' in the operation result. Otherwise, zero flag is cleared to "0".

■Carry Flag (CF)
Carry flag (CF) is set to "1", when a carry from or a borrow to the MSB occurs. Carry flag is cleared to "0", when no carry or borrow occurs.

■Negative Flag (NF)
Negative flag (NF) is set to "1" when MSB is '1' and reset to "0" when MSB is '0'. Negative flag is used to handle a signed value.

■Overflow Flag (VF)
Overflow flag (VF) is set to "1", when the arithmetic operation results overflow as a signed value. Otherwise, overflow flag is cleared to "0".
Overflow flag is used to handle a signed value.

■Interrupt Mask Level (IM1 and IM0)
Interrupt mask level (IM1 and IM0) controls the maskable interrupt acceptance in accordance with the interrupt factor interrupt priority for the interrupt control circuit in the CPU. The two-bit control flag defines levels '0' to '3'. Level 0 is the highest mask level. The interrupt request will be accepted only when the level set in the interrupt level flag (xxxLVn) of the interrupt control register (xxxICR) is higher than the interrupt mask level. When the interrupt is accepted, the level is reset to IM1-IM0, and interrupts whose mask levels are the same or lower are rejected during the accepted interrupt processing.

Table 2-1-3 Interrupt Mask Level and Interrupt Acceptance

	Interrupt mask level		Priority	Acceptable interrupt levels
	IM1	IM0		
Mask level 0	0	0	High	Non-maskable interrupt (NMI) only
Mask level 1	0	1	.	NMI, Level 0
Mask level 2	1	0	.	NMI, Level 0 to 1
Mask level 3	1	1	Low	NMI, Level 0 to 2

■Maskable Interrupt Enable (MIE)
Maskable interrupt enable flag (MIE) enables/disables acceptance of maskable interrupts by the CPU's internal interrupt acceptance circuit. A '1' enables maskable interrupts; a '0' disables all maskable interrupts regardless of the interrupt mask level (IM1-IM0) setting in PSW.
This flag is not changed by interrupts.

2-1-8 Addressing Modes

This LSI supports the nine addressing modes.

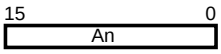
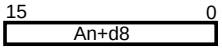
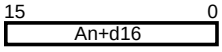
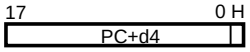
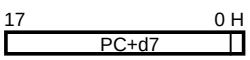
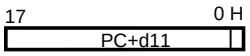
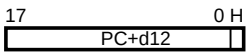
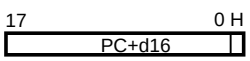
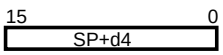
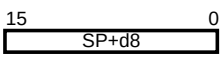
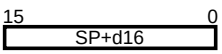
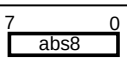
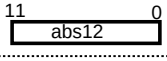
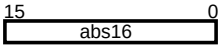
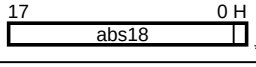
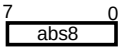
Each instruction uses a combination of the following addressing modes.

1) Register direct
2) Immediate
3) Register indirect
4) Register relative indirect
5) Stack relative indirect
6) Absolute
7) RAM short
8) I/O short
9) Handy

These addressing modes are well-suited for C language compilers. All of the addressing modes can be used for data transfer instructions. In modes that allow half-byte addressing, the relative value can be specified in half-byte (4-bit) increments, so that instruction length can be shorter. Handy addressing reuses the last memory address accessed and is only available with the MOV and MOVW instructions. Combining handy addresssing with absolute addressing reduces code size. For transfer data between memory, 7 addressing modes ; register indirect, register relative indirect, stack relative indirect, absolute, RAM short, I/O short, handy can be used. For operation instruction, register direct and immediate can be used. Refer to instruction's manual for the MN101C series.

This LSI is designed for 8-bit data access. It is possible to tranfer data in 16-bit increments with odd or all even addresses.

Table 2-1-4 Addressing Modes

Addressing mode		Effective address	Explanation
Register direct	Dn/DWn An/SP PSW	-	Directly specifies the register. Only internal registers can be specified.
Immediate	imm4/imm8 imm16	-	Directly specifies the operand or mask value appended to the instruction code.
Register indirect	(An)		Specifies the address using an address register.
Register relative indirect	(d8, An)		Specifies the address using an address register with 8-bit displacement.
	(d16, An)		Specifies the address using an address register with 16-bit displacement.
	(d4, PC) (branch instructions only)	 * 1	Specifies the address using the program counter with 4-bit displacement and H bit.
	(d7, PC) (branch instructions only)	 * 1	Specifies the address using the program counter with 7-bit displacement and H bit.
	(d11, PC) (branch instructions only)	 * 1	Specifies the address using the program counter with 11-bit displacement and H bit.
	(d12, PC) (branch instructions only)	 * 1	Specifies the address using the program counter with 12-bit displacement and H bit.
	(d16, PC) (branch instructions only)	 * 1	Specifies the address using the program counter with 16-bit displacement and H bit.
Stack relative indirect	(d4, SP)		Specifies the address using the stack pointer with 4-bit displacement.
	(d8, SP)		Specifies the address using the stack pointer with 8-bit displacement.
	(d16, SP)		Specifies the address using the stack pointer with 16-bit displacement.
Absolute	(abs8)		Specifies the address using the operand value appended to the instruction code. Optimum operand length can be used to specify the address.
	(abs12)		
	(abs16)		
	(abs18) (branch instructions only)	 * 1	
RAM short	(abs8)		Specifies an 8-bit offset from the address x'00000'.
I/O short	(io8)		Specifies an 8-bit offset from the top address (x'03F00') of the special function register area.
Handy	(HA)	-	Reuses the last memory address accessed and is only available with the MOV and MOVW instructions. Combined use with absolute addressing reduces code size.

* 1 H: half-byte bit

2-2 Memory Space

2-2-1 Memory Mode

ROM is the read only area and RAM is the memory area which contains readable/writable data. In addition to these, peripheral resources such as memory-mapped special registers are allocated. This LSI supports one memory mode (single chip mode) in its memory model.

In single chip mode, the system consists of only internal memory.
Settings for this mode are as follows ;

Table 2-2-1 Memory Mode Setup

Memory mode	MMOD pin	EXMEM flag (MEMCTR register)
Single chip mode	L	0

MMOD pin should be fixed to "L" level, or "H" level. Do not change the setup of MMOD pin after reset.

2-2-2 Single-chip Mode

In single-chip mode, the system consists of only internal memory. This is the optimized memory mode and allows construction of systems with the highest performance.

The single-chip mode uses only internal ROM and internal RAM. The MN101C series devices offer up to 12 KB of RAM and up to 240 KB of ROM. This LSI offers 512 bytes of RAM and 24 KB of ROM.

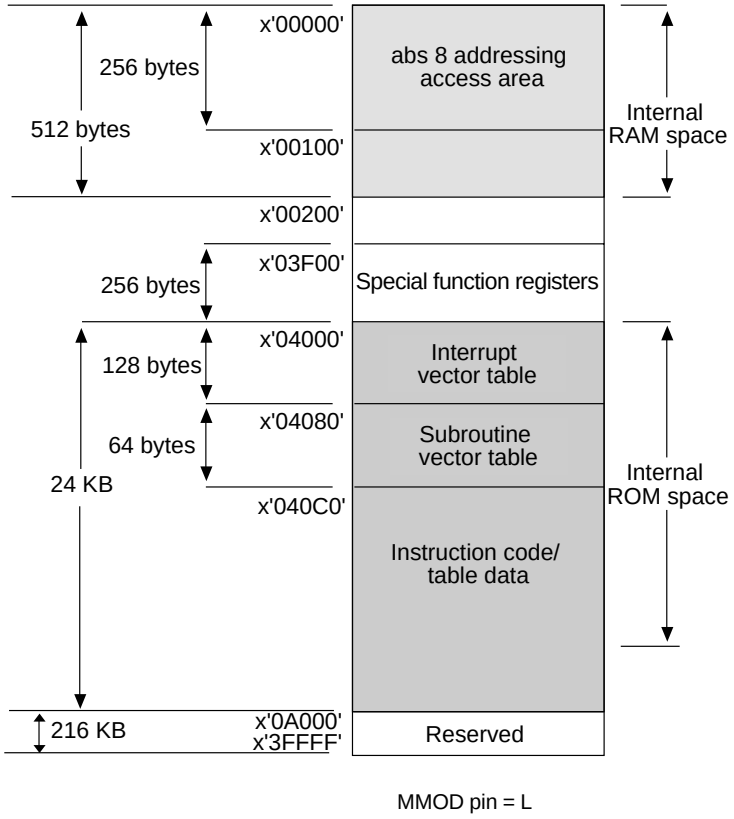


Figure 2-2-1 Single-chip Mode

The value of internal RAM is uncertain when power is applied to it.
It needs to be initialized before it is used.

2-2-3 Special Function Registers

The MN101C series locates the special function registers (I/O spaces) at the addresses x'03F00' to x'03FFF' in memory space. The special function registers of this LSI are located as shown below.

Table 2-2-2 Register Map

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03F0X	CPUM	MEMCTR	WDCTR	DLYCTR	Reserved						Reserved	Reserved		OSCMD		Reserved
03F1X	P0OUT	P1OUT	P2OUT				P6OUT	P7OUT	P8OUT		PAOUT		PCOUT			
03F2X	P0IN	P1IN	P2IN				P6IN	P7IN	P8IN		PAIN		PCIN		FLOAT	P1OMD
03F3X	P0DIR	P1DIR					P6DIR	P7DIR	P8DIR		PAIMD		PCDIR			PADIR
03F4X	P0PLU	P1PLU	P2PLU				P6PLU	P7PLUD	P8PLU		PAPLUD		PCPLU		P6IMD	PAODC
03F5X									TM2BC	TM3BC	TM2OC	TM3OC	TM2MD	TM3MD	CK2MD	CK3MD
03F6X									TM6BC	TM6OC	TM6MD	TBCLR			RMCTR	PSCMD
03F7X	TM7BCL	TM7BCH	TM7OC1L	TM7OC1H	TM7PR1L	TM7PR1H	TM7ICL	TM7ICH	TM7MD1	TM7MD2	TM7OC2L	TM7OC2H	TM7PR2L	TM7PR2H		
03F8X															NFCTR	EDGDT
03F9X	SC0MD0	SC0MD1	SC0MD2	SC0STR	RXBUF0	TXBUF0	SC0ODC	SC0CKS								
03FAX																
03FBX	ANCTR0	ANCTR1	ANCTR2	ANBUF0	ANBUF1											
03FCX																
03FDX																
03FEX	Reserved	NMICR	IRQ0ICR	IRQ1ICR	IRQ2ICR	IRQ3ICR						TM2ICR	TM3ICR			TM6ICR
03FFX	TBICR	TM7ICR	T7OC2 ICR			SC0RICR	SC0TICR				ADICR					Reserved

CPU mode, memory control

I/O ports: Port output; Port input; I/O mode control; Resistor control

Timer control

Interrupt I/F control

Serial I/F control

Analog I/F control

Interrupt control

2-3 Bus Interface

2-3-1 Bus Controller

The MN101C series provides separate buses to the internal memory and internal peripheral circuits to reduce bus line loads and thus realize faster operation.

There are three such buses: ROM bus, RAM bus, and peripheral expansion bus (I/O bus). They connect to the internal ROM, internal RAM, and internal peripheral circuits respectively. The bus control block controls the parallel operation of instruction read and data access. A functional block diagram of the bus controller is given below.

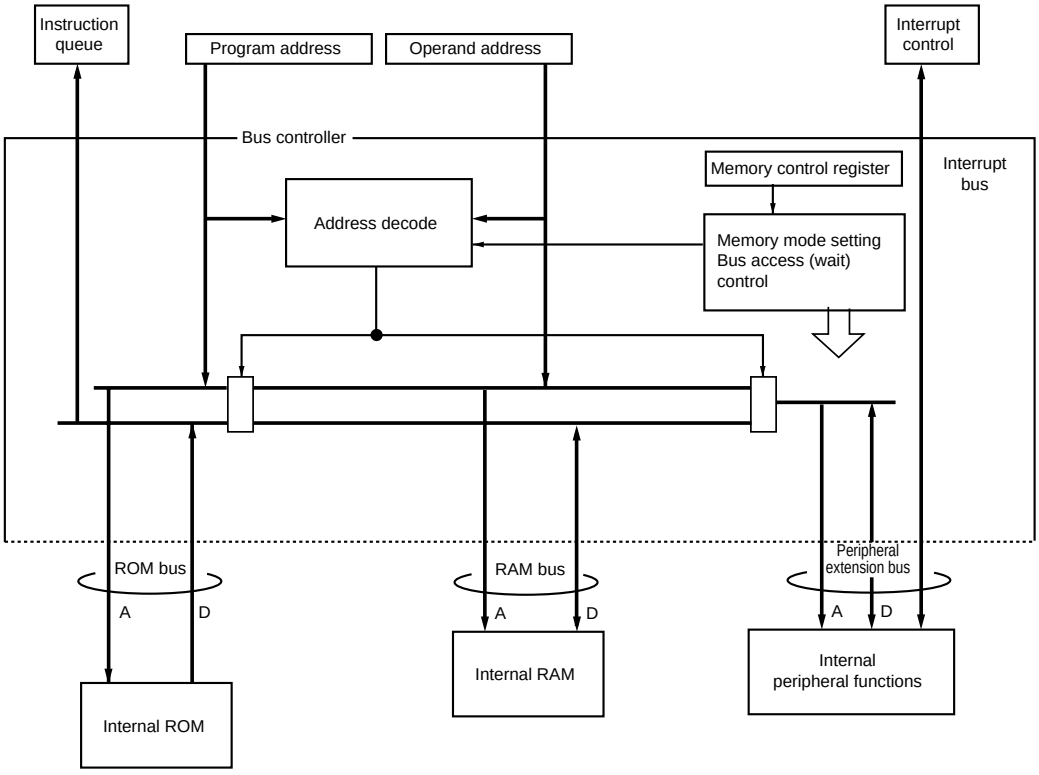


Figure 2-3-1 Functional Block Diagram of the Bus Controller

2-3-2 Control Registers

Bus interface is controlled by the memory control register (MEMCTR).

■Memory Control Register (MEMCTR)

	7	6	5	4	3	2	1	0	
MEMCTR	IOW1	IOW0	IVBM	EXMEM	-	IRWE	-	-	(At reset : 1 1 0 0 - 0 - -)

IRWE	Software write enable flag for interrupt request flag
0	Software write disable Even if data is written to each interrupt control register (xxxICR), the state of the interrupt request flag (xxxIR) will not change.
1	Software write enable

EXMEM	Set always to "0".

IVBM	Base address setting for interrupt vector table
0	Interrupt vector base = x'04000'
1	Interrupt vector base = x'00100'

IOW1 to 0	Wait cycles when accessing special register area	Bus cycle at 20 MHz oscillation
0 0	No wait cycles	100 ns
0 1	1 wait cycle	100 ns
1 0	2 wait cycles	200 ns
1 1	3 wait cycles	250 ns

Figure 2-3-2 Memory Control Register (MEMCTR: x'03F01' R/W)

The IOW1-IOW0 wait settings affect accesses to the special registers located at the addresses x'3F00'-x'3FFF'. After reset, MEMCTR specifies the fixed wait cycle mode with three wait cycles. Wait setting of IOW is a function, which CPU supports for special use, for example, when special function register or I/O is expanded to external. For this LSI, wait cycle setting is not always necessary. Select "no-wait cycle" for high performance system construction.

2-4 Standby Function

2-4-1 Overview

This LSI has two sets of system clock oscillator (high speed oscillation, low speed oscillation) for two CPU operating modes (NORMAL and SLOW), each with two standby modes (HALT and STOP). Power consumption can be decreased with using those modes.

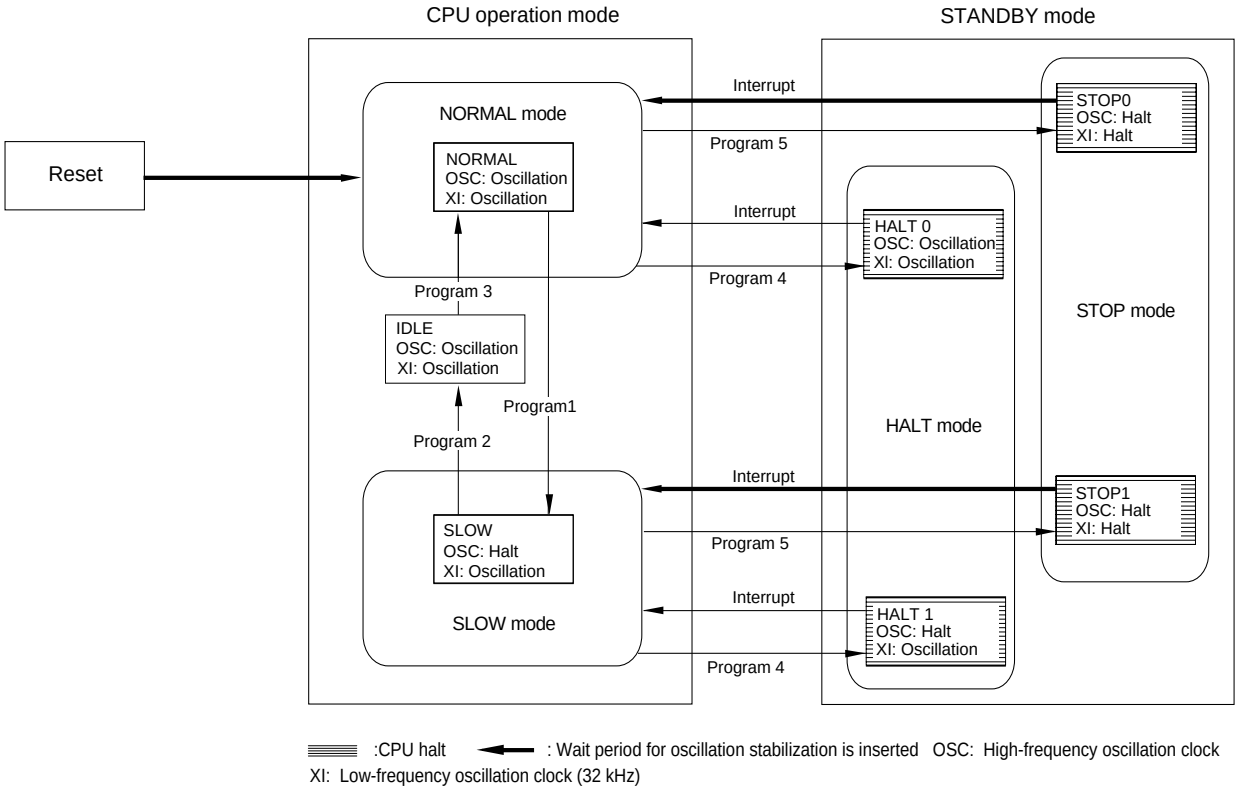


Figure 2-4-1 Transition Between Operation Modes

■HALT Modes (HALT0, HALT1)
– The CPU stops operating. But both of the oscillators remain operational in HALT0 and only the high-frequency oscillator stops operating in HALT1.
– An interrupt returns the CPU to the previous CPU operating mode that is, to NORMAL from HALT0 or to SLOW from HALT1.

■STOP Modes (STOP0, STOP1)
– The CPU and both of the oscillators stop operating.
– An interrupt restarts the oscillators and, after allowing time for them to stabilize, returns the CPU to the previous CPU operating mode - that is, to NORMAL from STOP0 or to SLOW from STOP1.

■SLOW Mode
– This mode executes the software using the low-frequency clock. Since the high-frequency oscillator is turned off, the device consumes less power while executing the software.

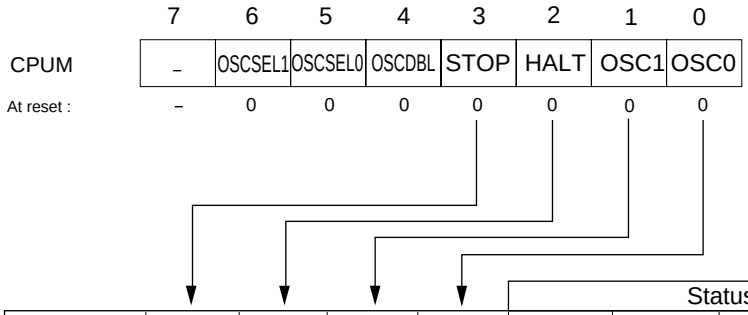
■IDLE Mode
– This mode allows time for the high-frequency oscillator to stabilize when the software is changing from SLOW to NORMAL mode.

To reduce power dissipation in STOP and HALT modes, it is necessary to check the stability of both the output current from pins and port level of input pins. For output pins, the output level should match the external level or direction control should be changed to input mode. For input pins, the external level should be fixed.
This LSI has two system clock oscillation circuits. OSC is for high-frequency operation (NORMAL mode) and XI is for low-frequency operation (SLOW mode). Transition between NORMAL and SLOW modes or to standby mode is controlled by the CPU mode control register (CPUM). Reset and interrupts are the return factors from standby mode. A wait period is inserted for oscillation stabilization at reset and when returning from STOP mode, but not when returning from HALT mode. High/low-frequency oscillation mode is automatically returned to the same state as existed before entering standby mode.

To stabilize the synchronization at the moment of switching clock speed between high speed oscillation (fosc) and low speed oscillation (fx), fosc should be set to 2.5 times or higher frequency than fx.

2-4-2 CPU Mode Control Register

Transition from one mode to another mode is controlled by the CPU mode control register (CPUM).



Operation mode	STOP	HALT	OSC1	OSC0	Status: OSC1 /OSC0	Status: XI/XO	Status: System clock	Status: CPU
NORMAL	0	0	0	0	Oscillation	Oscillation	OSCI	Operating
IDLE	0	0	0	1	Oscillation	Oscillation	XI	Operating
SLOW	0	0	1	1	Halt	Oscillation	XI	Operating
HALT0	0	1	0	0	Oscillation	Oscillation	OSCI	Halt
HALT1	0	1	1	1	Halt	Oscillation	XI	Halt
STOP0	1	0	0	0	Halt	Halt	Halt	Halt
STOP1	1	0	1	1	Halt	Halt	Halt	Halt

Figure 2-4-2 Operating Mode and Clock Oscillation (CPUM : x'3F00', R/W)

The procedure for transition from NORMAL to HALT or STOP mode is given below.

(1) If the return factor is a maskable interrupt, set the MIE flag in the PSW to "1" and set the interrupt mask (IM) to a level permitting acceptance of the interrupt.
(2) Clear the interrupt request flag (xxxIR) in the maskable interrupt control register (xxxICR) , set the interrupt enable flag (xxxIE) for the return factor, and set the IE flag in the PSW.
(3) Set CPUM to HALT or STOP mode.

Set the IRWE flag of the memory control register (MEMCTR) to clear interrupt request flag by software.

System clock (fs) is changed depending on CPU operation mode.
In NORMAL mode, HALT0 mode, fs is based on fosc (high speed oscillation). In SLOW mode, IDLE mode, HALT1 mode, fs is based on fx (low speed oscillation).
[☞ Chapter 2. 2-5 Clock Switching]

2-4-3 Transition between SLOW and NORMAL

This LSI has two CPU operating modes, NORMAL and SLOW. Transition from SLOW to NORMAL requires passing through IDLE mode.

A sample program for transition from NORMAL to SLOW mode is given below.

```
Program 1
        MOV   x'3', D0          ; Set SLOW mode.
        MOV   D0, (CPUM)
```

Transition from NORMAL to SLOW mode, when the low-frequency clock has fully stabilized, can be done by writing to the CPU mode control register. In this case, transition through IDLE is not needed.

For transition from SLOW to NORMAL mode, the program must maintain the idle state until high-frequency clock oscillation is fully stable. In IDLE mode, the CPU operates on the low-frequency clock.

For transition from SLOW to NORMAL, oscillation stabilization waiting time is required same as that after reset. Software must count that time.
We recommend selecting the oscillation stabilization time after consulting with oscillator manufacturers.

Sample program for transition from SLOW to NORMAL mode is given below.

```
Program 2
                MOV         x'01', D0          ; Set IDLE mode.
                MOV         D0, (CPUM)
```

```
Program 3
        MOV   x'0B', D0       ; A loop to keep approx. 6.7ms with low-frequency clock (32.768 kHz)
LOOP    ADD   -1, D0          ; operation when changed to high-frequency clock (20 MHz).
        BNE   LOOP            ;
        SUB   D0, D0          ;
        MOV   D0, (CPUM)      ;  Set NORMAL mode.
```

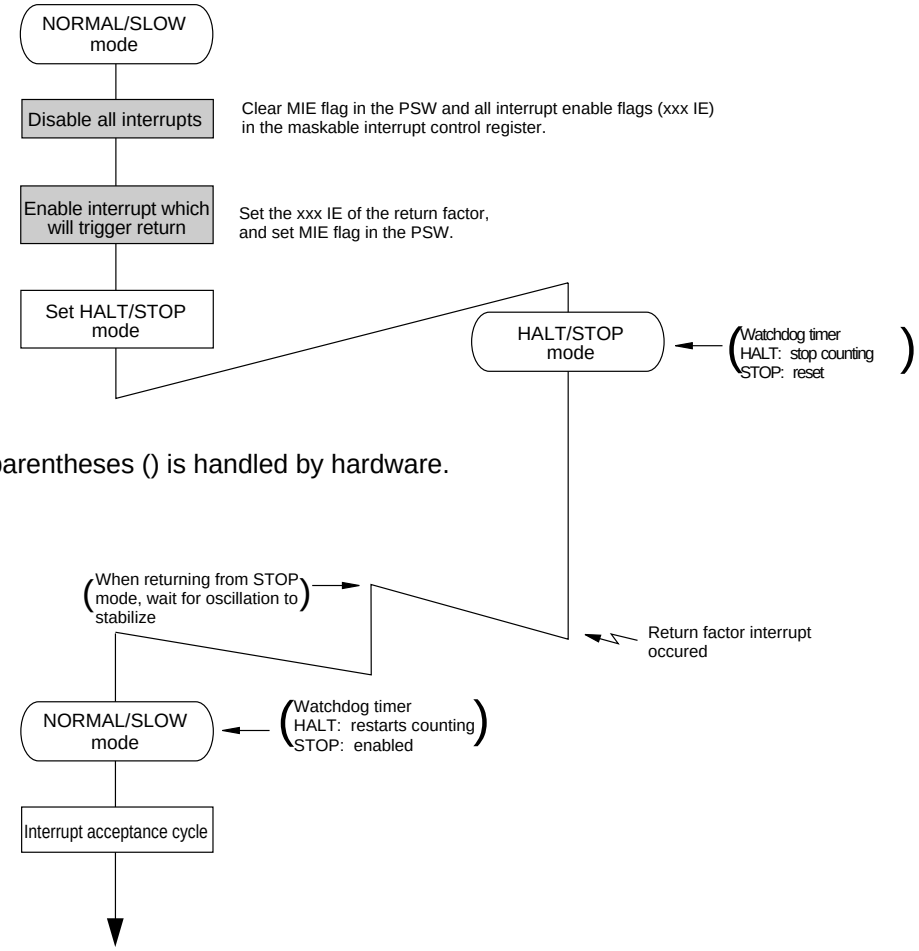
2-4-4 Transition to STANDBY Modes

The program initiates transitions from a CPU operating mode to the corresponding STANDBY (HALT/STOP) modes by specifying the new mode in the CPU mode control register (CPUM). Interrupts initiate the return to the former CPU operating mode.

Before initiating a transition to a STANDBY mode, however, the program must

(1) Set the maskable interrupt enable flag (MIE) in the processor status word (PSW) to '0' to disable all maskable interrupts temporarily.

(2) Set the interrupt enable flags (xxxIE) in the interrupt control registers (xxxICR) to '1' or '0' to specify which interrupts do and do not initiate the return from the STANDBY mode. Set MIE '1' to enable those maskable interrupts.



Processing inside parentheses () is handled by hardware.

Figure 2-4-3 Transition to/from STANDBY Mode

> **!** If the interrupt is enabled but interrupt priority level of the interrupt to be used is not equal to or higher than the mask level in PSW before transition to HALT or STOP mode, it is impossible to return to CPU operation mode by maskable interrupt.

■Transition to HALT modes

The system transfers from NORMAL mode to HALT0 mode, and from SLOW mode to HALT1 mode. The CPU stops operating, but the oscillators remain operational. There are two ways to leave a HALT mode: a reset or an interrupt. A reset produces a normal reset; an interrupt, an immediate return to the CPU state prior to the transition to the HALT mode. The watchdog timer, if enabled, resumes counting.

```
Program 4
        MOV     x'4',  D0      ; Set HALT mode.
        MOV     D0,  (CPUM)
        NOP                    ; After written in CPUM, some NOP
        NOP                    ; instructions (three or less) are
        NOP                    ; executed.
```

```
Program 4
        MOV     x'7',  D0      ; Set HALT mode.
        MOV     D0,  (CPUM)
        NOP                    ; After written in CPUM, some NOP
        NOP                    ; instructions (three or less) are
        NOP                    ; executed.
```

■Transition to STOP mode

The system transfers from NORMAL mode to STOP0 mode, and from SLOW mode to STOP1 mode. In both cases, oscillation and the CPU are both halted. There are two ways to leave a STOP mode: a reset or an interrupt.

```
Program 5
        MOV     x'8',  D0      ; Set STOP mode
        MOV     D0,  (CPUM)
        NOP                    ; After written in CPUM, some NOP
        NOP                    ; instructions (three or less) are
        NOP                    ; executed.
```

```
Program 5
        MOV     x'B',  D0      ; Set STOP mode
        MOV     D0,  (CPUM)
        NOP                    ; After written in CPUM, some NOP
        NOP                    ; instructions (three or less) are
        NOP                    ; executed.
```

Right after the instruction of the transition to HALT, STOP mode, NOP instruction should be inserted 3 times.

2-5 Clock Switching

This LSI can select the best operation clock for system by switching clock cycle division factor by program. Division factor is determined by both flags of the CPU mode control register (CPUM) and the Oscillator frequency control register (OSCMD).

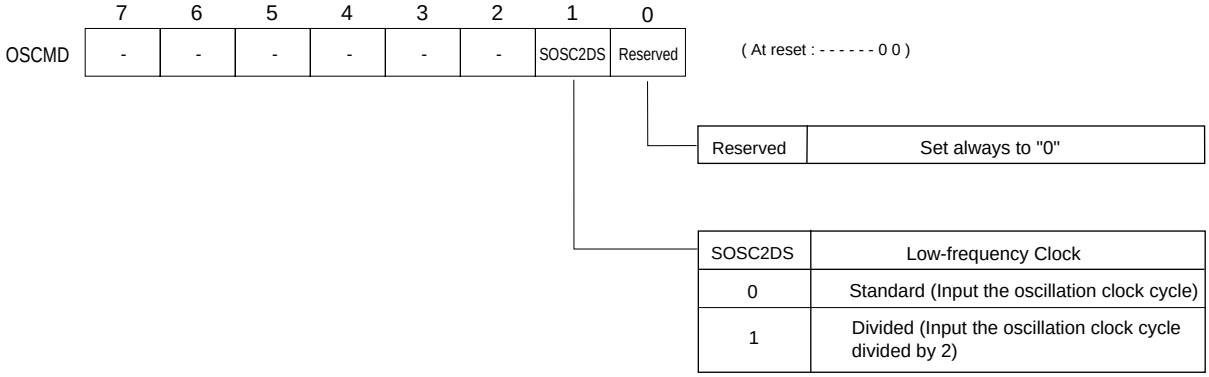


Figure 2-5-1 Oscillator Frequency Control Register (OSCMD : x'03F0D', R/W)

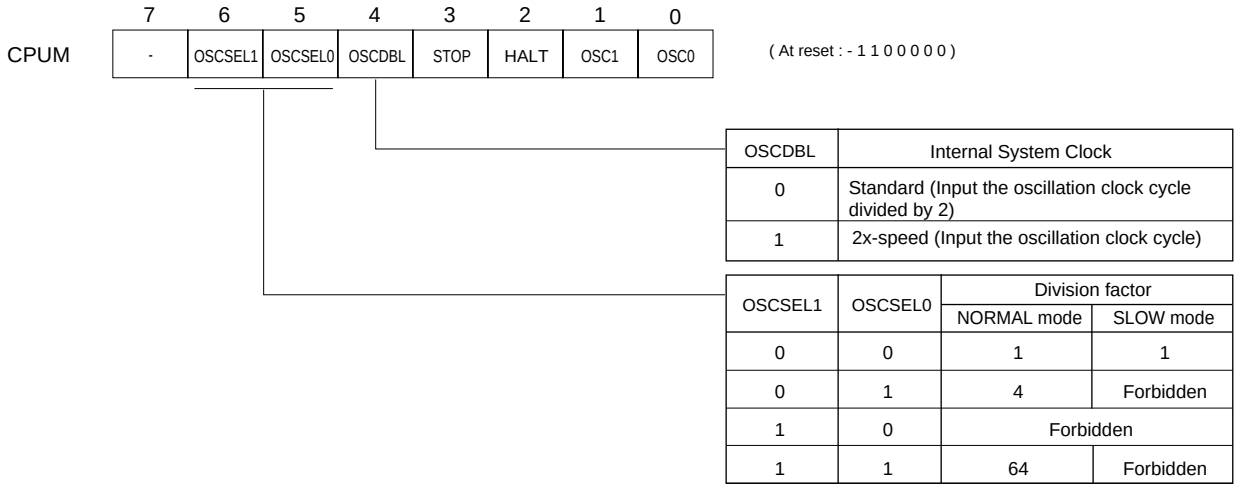


Figure 2-5-2 CPU Mode Control Register (CPUM : x'03F00', R/W)

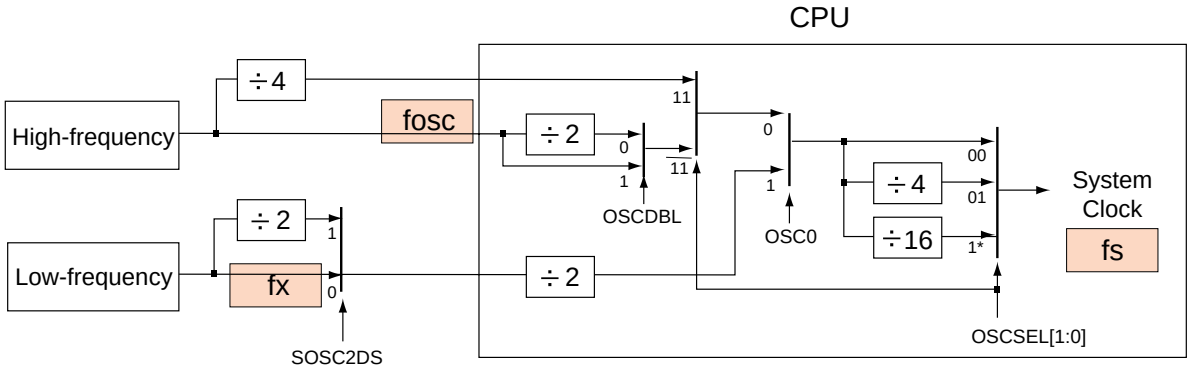


Figure 2-5-3 Clock Switching Circuit

OSCSEL1	OSCSEL0	OSCDBL	Division factor for High-frequency(OSC) Input (NORMAL mode)
0	0	0	2
0	1	1	4
0	1	0	8
1	1	0	64

Do not use other than this setup

Figure 2-5-4 Setting Division Factor at NORMAL mode by combination of OSCSEL and OSCDBL

OSCSEL1	OSCSEL0	SOSC2DS	Division factor for Low-frequency(XI/XO) Input (SLOW mode)
0	0	0	2
0	0	1	4

Do not use other than this setup

Figure 2-5-5 Setting Division Factor at SLOW mode by combination of OSCSEL and SOSC2DS

fs=fosc/64 after reset is released.

■Transition in clock switching

Clock switching in NORMAL mode should be operated in the order of following arrows.

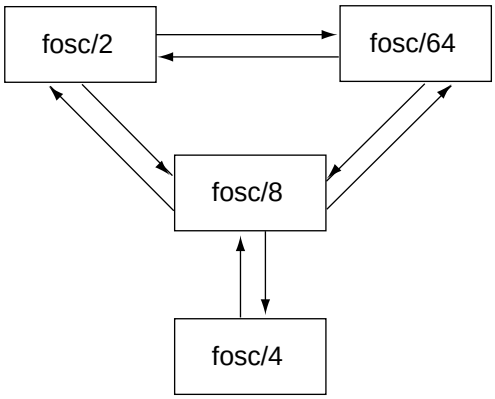


Figure 2-5-6 Transition in clock switching

For clock switching, set OSCDBL flag, OSCSEL flag and OSC0 flag separately.
Even two flags mapped in same special function register need to be set individually.

2-6 Reset

2-6-1 Reset operation

The CPU contents are reset and registers are initialized when the NRST pin (P27) is pulled to low.

■Initiating a Reset

There are two methods to initiate a reset.

(1) Drive the NRST pin low.
NRST pin should be held "low" for more than OSC 4 clock cycles (200 ns at a 20 MHz).

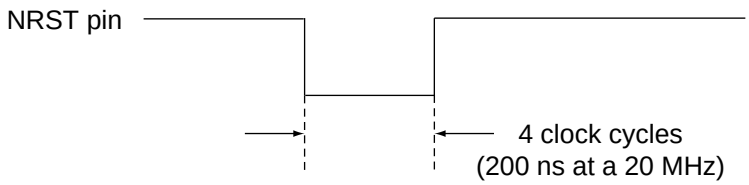


Figure 2-6-1 Minimum Reset Pulse Width

(2) Setting the P2OUT7 flag of the P2OUT register to "0" outputs low level at P27 (NRST) pin. And transferring to reset by program (software reset) can be executed. If the internal LSI is reset and register is initiated, the P2OUT7 flag becomes "1" and reset is released.

[☞ Chapter 4. 4-4-2 Registers]

On this LSI, the starting mode is NORMAL mode that high oscillation is the base clock.

When NRST pin is connected to low power voltage circuit that gives pulse for enough low level time at sudden unconnection. And reset can be generated even if NRST pin is held "low" for less than OSC 4 clock cycles, take notice of noise.

■Sequence at Reset

(1) When reset pin comes to high level from low level, the innternal 14-bit counter (It can be used as watchdog timer, too.) starts its operation by system clock. The period from starting its count from its overflow is called oscillation stabilization wait time.

(2) During reset, internal register and special function register are initiated.

(3) After oscillation stabilization wait time, internal reset is released and program is started from the address written at address X '4000' at interrupt rector table.

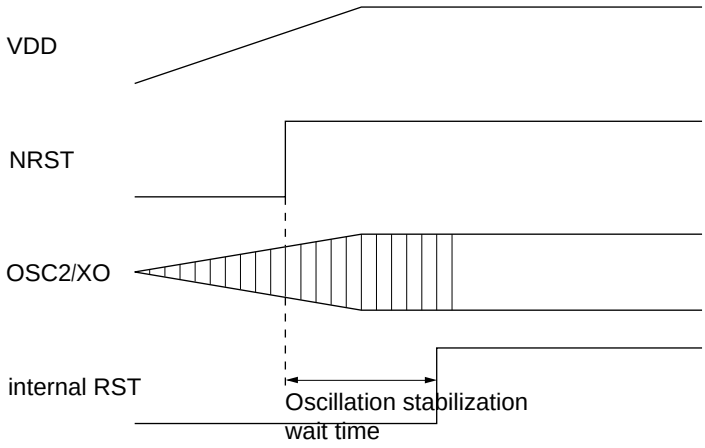


Figure 2-6-2 Reset Released Sequence

2-6-2 Oscillation Stabilization Wait time

Oscillation stabilization wait time is the period from the stop of oscillation circuit to the stablization for oscillation. Oscillation stabilization wait time is automatically inserted at releasing from reset and at recovering from STOP mode. At recovering from STOP mode the oscillation stabilization wait time control register (DLYCTR) is set to select the oscillation stabilization wait time. At releasing from reset, oscillation stabilization wait time is fixed.

The timer that counts oscillation stabilization wait time is also used as a watchdog timer. That is used as a runaway detective timer at anytime except at releasing from reset and at recovering from STOP mode. Watchdog timer is initiated at reset and at STOP mode and starts counting from the initialize value (x'0000') when system clock (fs) is as clock source. After oscillation stabilization wait time, it continues counting as a watchdog timer. [☞ Chapter 9 Watchdog timer]

■Block Diagram of Oscillation Stabilization Wait Time (watchdog timer)

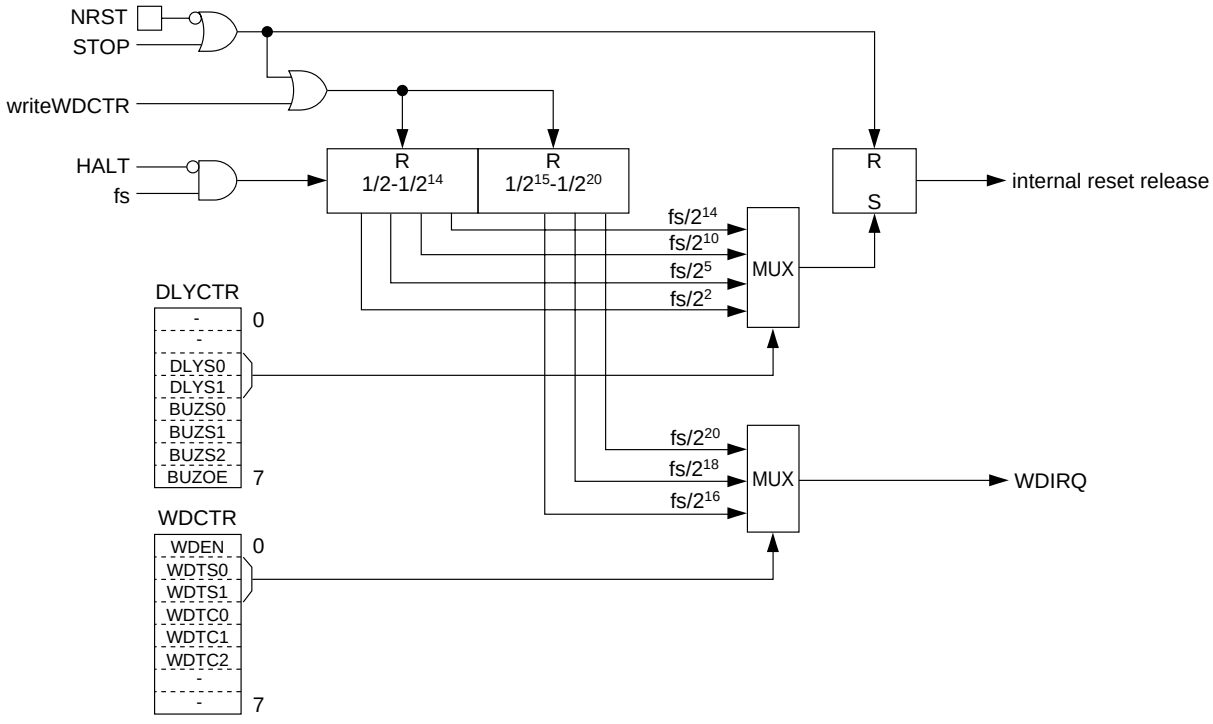
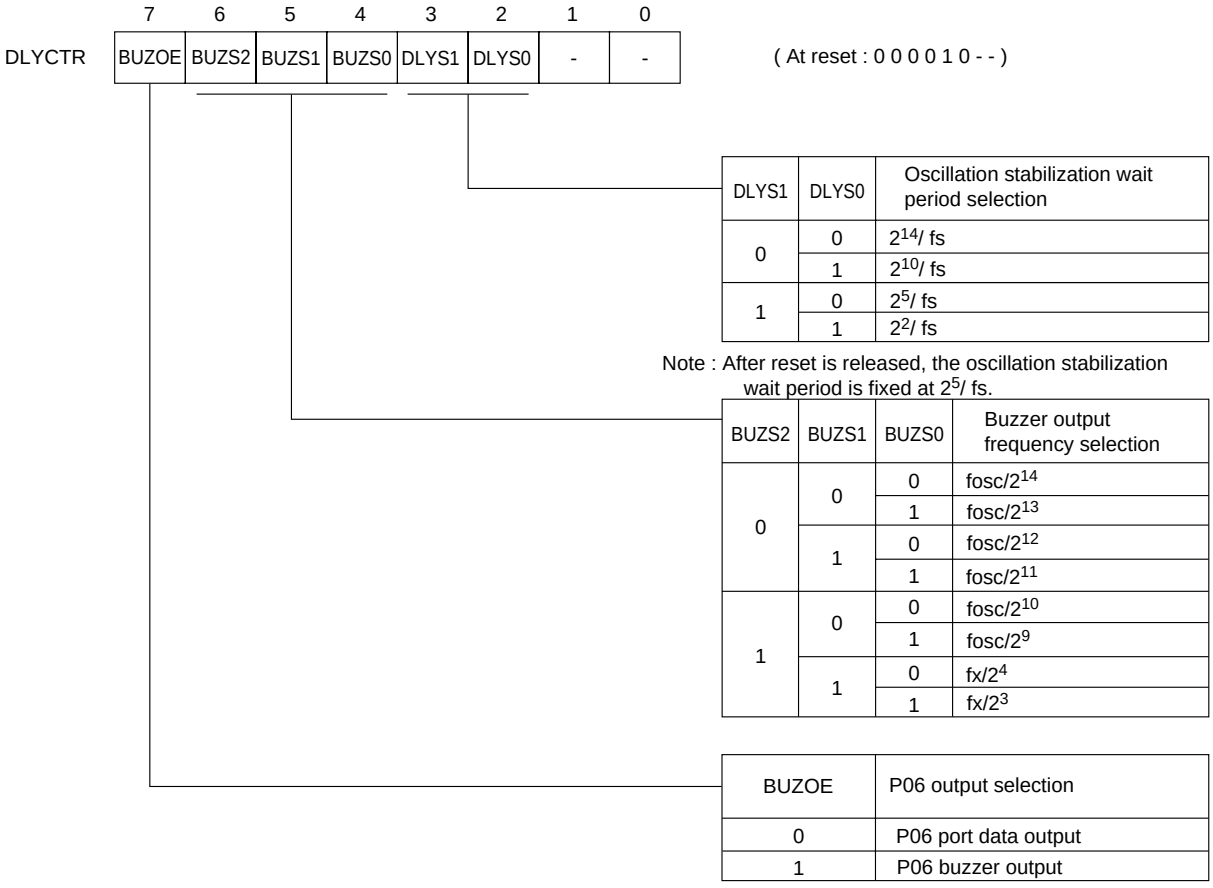


Figure 2-6-3 Block Diagram of Osillation Stabilization Wait Time (watchdog timer)

■Oscillation Stabilization Wait Time Control Register



DLYS1	DLYS0	Oscillation stabilization wait period selection
0	0	2^{14}/ fs
	1	2^{10}/ fs
1	0	2^{5}/ fs
	1	2^{2}/ fs

Note : After reset is released, the oscillation stabilization wait period is fixed at 2^{5}/ fs.

BUZS2	BUZS1	BUZS0	Buzzer output frequency selection
0	0	0	fosc/2^{14}
		1	fosc/2^{13}
	1	0	fosc/2^{12}
		1	fosc/2^{11}
1	0	0	fosc/2^{10}
		1	fosc/2^{9}
	1	0	fx/2^{4}
		1	fx/2^{3}

BUZOE	P06 output selection
0	P06 port data output
1	P06 buzzer output

Figure 2-6-4 Oscillation Stabilization Wait Time Control Register (DLYCTR : x'03F03', R/W)

System clock is fs=fosc/64 after reset release. If system clock is switched to fs=fosc/2 at oscillation stabilization wait time 2^{5}/fs with the first instruction, the time T required for MCU to start operation at fosc/2 (including the oscillation stabilization wait time) in single chip mode is;

T = 47.5/fs = 3040/fosc (at OSC8 MHz T=380 μs)

To extend the oscillation stabilizaion wait time, operate MCU at fosc/64. Select the oscillation cycle in consideration of matching with your oscillator.

■Control the Oscillation Stabilization Wait Time

At recovering from STOP mode, the bit 3-2 (DLYS1, DLYS0) of the oscillation stabilization wait time control register can be set to select the oscillation stabilization wait time from 2^{14}, 2^{10}, 2^5, 2^2 x system clock. The DLYCTR register is also used for controlling of buzzer functions.

[☞ Chapter 10 Buzzer]

At releasing from reset, the oscillation stabilization wait time is fixed to "2^5 x system clock". System clock is determined by the CPU mode control register (CPUM).

Table 2-6-1 Oscillation Stabilization Wait Time

DLYS1	DLYS0	Oscillation stabilization wait time
0	0	2^{14} x System clock
0	1	2^{10}x System clock
1	0	2^5 x System clock
1	1	2^2 x System clock

After reset release, if system clock is switched to fs=fosc/2, time T, required for CPU operation at fosc/2, including oscillation stablization wait time, in single chip mode is ;

T=47.5/fs=3040/fosc (at fosc 8 MHz, T=380μs)

Chapter 3 Interrupts

3

3-1 Overview

This LSI speeds up interrupt response with circuitry that automatically loads the branch address to the corresponding interrupt service routine from an interrupt vector table : reset, non-maskable interrupts (NMI), 9 maskable peripheral interrupts, and 4 external interrupts.

For interrupts other than reset, the interrupt processing sequence consists of interrupt request, interrupt acceptance, and hardware processing. After the interrupt is accepted, the program counter (PC) and processor status word (PSW) and handy addressing data (HA) are saved onto the stack. And an interrupts handler ends by restoring, using the POP instruction and other means, the contents of any registers used during processing and then executing the return from interrupt (RTI) instruction to return to the point at which execution was interrupted. Max.12 machine cycles before execution, and max 11 machine cycles after execution.

Each interrupt has an interrupt control register, which controls the interrupts. Interrupt control register consists of the interrupt level field (LV1-0), interrupt enable flag (IE), and interrupt request flag (IR).

Interrupt request flag (IR) is set to "1" by an interrupt request, and cleared to "0" by the interrupt acceptance. This flag is managed by hardware, but can be rewritten by software.

Interrupt enable flag (IE) is the flag that enables interrupts in the group. There is no interrupt enable flag in non-maskable interrupt (NMI). Once this interrupt request flag is set, it is accepted without any conditions. Interrupt enable flag is set in maskable interrupt. Interrupt enable flag (IE) of each maskable interrupt is valid when the maskable interrupt enable flag (MIE flag) of PSW is "1".

Maskable interrupts have had vector numbers by hardware, but their priority can be changed by setting interrupts level field. There are three hierarchical interrupt levels. If multiple interrupts have the same priority, the one with the lowest vector number takes priority. Maskable interrupts are accepted when its level is higher than the interrupt mask level (IM1-0) of PSW. Non-maskable interrupts are always accepted, regardless of the interrupt mask level.

3-1-1 Functions

Table 3-1-1 Interrupt Functions

Interrupt type	Reset (interrupt)	Non-maskable interrupt	Maskable interrupt
Vector number	0	1	2 to 26
Table address	x'04000'	x'04004'	x'04008' to x'04068
Starting address	Address specified by vector address		
Interrupt level	-	-	Level 0 to 2 (set by software)
Interrupt factor	External RST pin input	Errors detection, PI interrupt	External pin input Internal peripheral function
Generated operation	Direct input to CPU core	Input to CPU core from non-maskable interrupt control register (NMICR)	Input interrupt request level set in interrupt level flag (xxxL Vn) of maskable interrupt control register (xxxICR) to CPU core.
Accept operation	Always accepts	Always accepts	Acceptance only by the interrupt control of the register (xxxICR) and the interrupt mask level in PSW.
Machine cycles until acceptance	12	12	12
PSW status after acceptance	All flags are cleared to "0".	The interrupt mask level flag in PSW is cleared to "00".	Values of the interrupt level flag (xxxLVn) are set to the interrupt mask level (masking all interrupt requests with the same or the lower priority).

3-1-2 Block Diagram

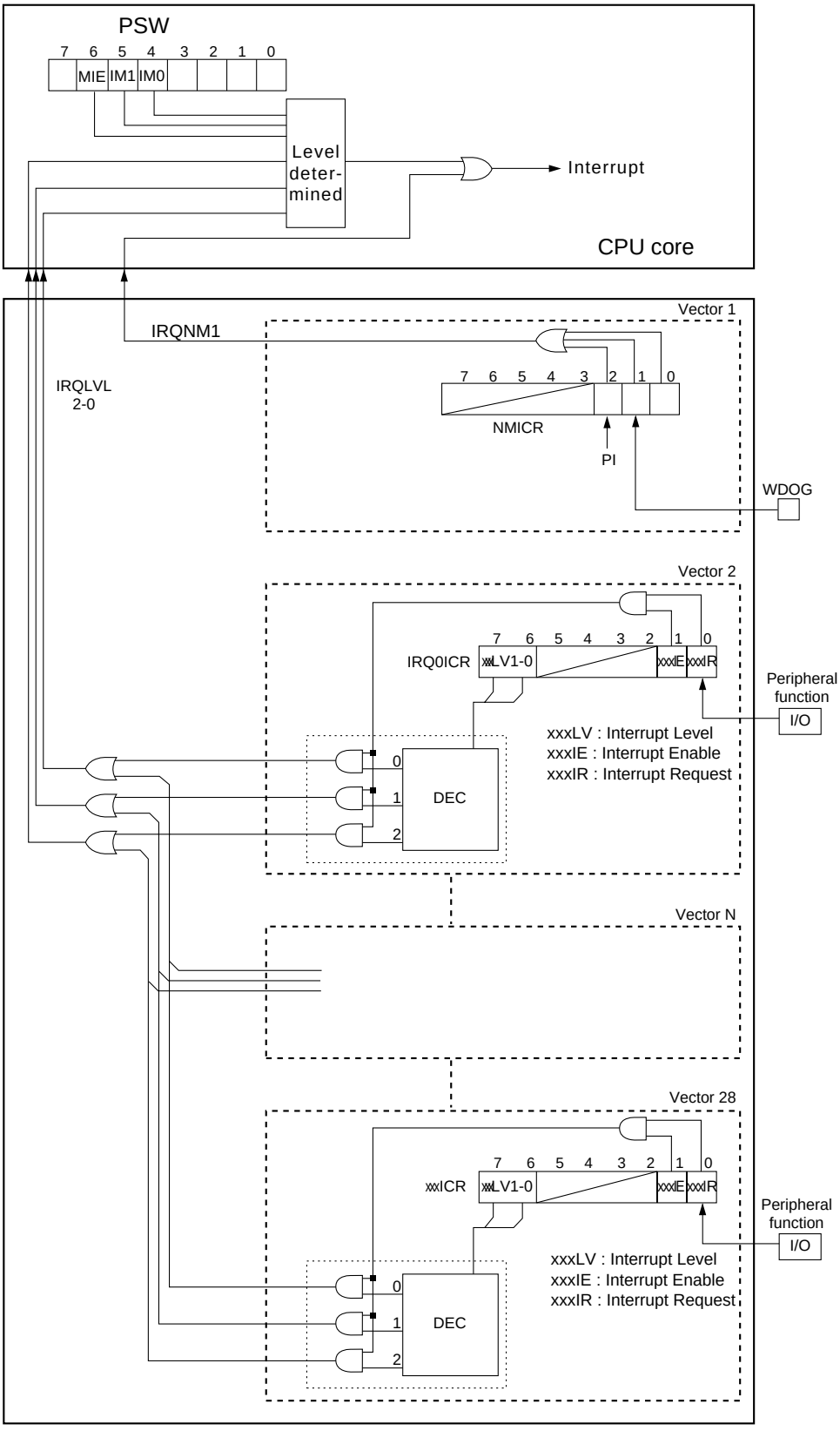


Figure 3-1-1 Interrupt Block Diagram

3-1-3 Operation

■Interrupt Processing Sequence

For interrupts other than reset, the interrupt processing sequence consists of interrupt request, interrupt acceptance, and hardware processing. The program counter (PC) and processor status word (PSW) and handy addressing data (HA) are saved onto the stack, and execution branches to the address specified by the corresponding interrupt vector.

An interrupt handler ends by restoring the contents of any registers used during processing and then executing the return from interrupt (RTI) instruction to return to the point at which execution was interrupted.

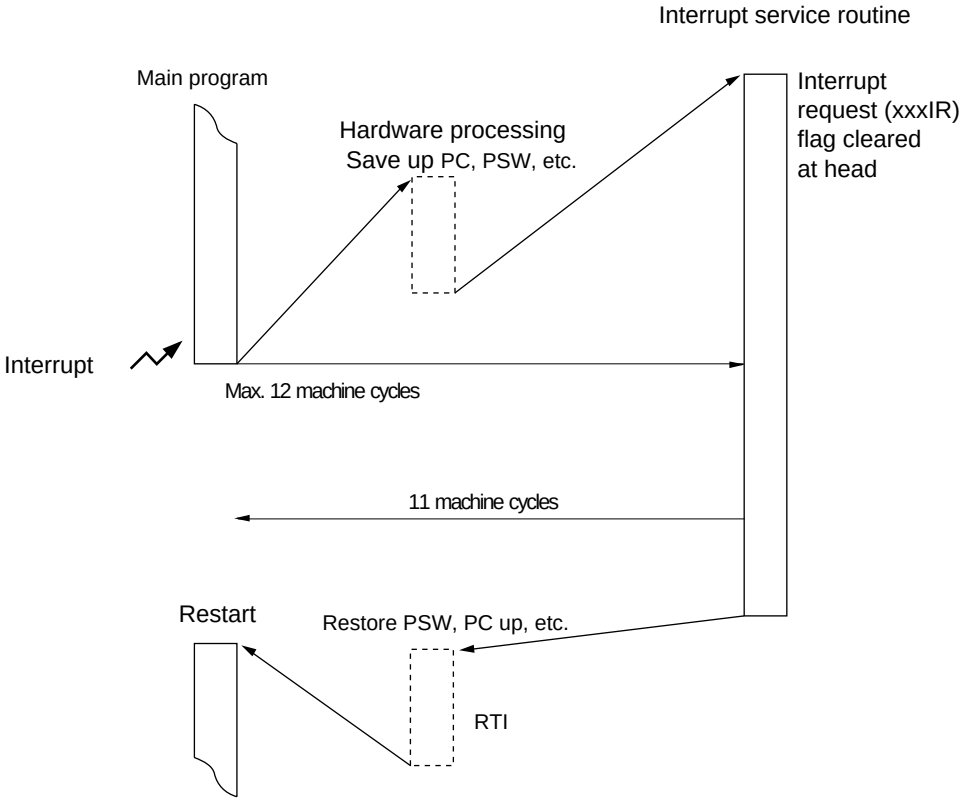


Figure 3-1-2 Interrupt Processing Sequence (maskable interrupts)

■Interrupt Sources and Vector Addresses

Here is the list of interrupt vector address and interrupt group.

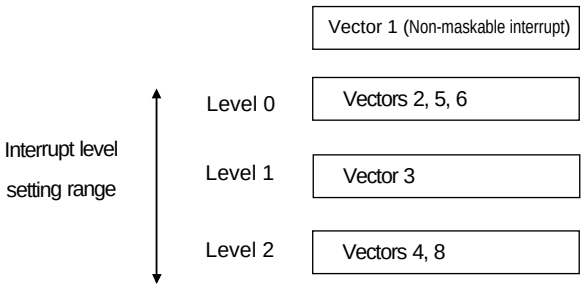
Table 3-1-2 Interrupt Vector Address and Interrupt Group

Vector Number	Vector Address	Interrupt group (Interrupt source)		Control Register (address)	
0	x'04000'	Reset	-	-	-
1	x'04004'	Non-maskable interrupt	NMI	NMICR	x'03FE1'
2	x'04008'	External interrupt 0	IRQ0	IRQ0ICR	x'03FE2'
3	x'0400C'	External interrupt 1	IRQ1	IRQ1ICR	x'03FE3'
4	x'04010'	External interrupt 2	IRQ2	IRQ2ICR	x'03FE4'
5	x'04014'	External interrupt 3	IRQ3	IRQ3ICR	x'03FE5'
6	x'04018'	Reserved	-	-	-
7	x'0401C'	Reserved	-	-	-
8	x'04020'	Reserved	-	-	-
9	x'04024'	Reserved	-	-	-
10	x'04028'	Reserved	-	-	-
11	x'0402C'	Timer 2 interrupt	TM2IRQ	TM2ICR	x'03FEB'
12	x'04030'	Timer 3 interrupt	TM3IRQ	TM3ICR	x'03FEC'
13	x'04034'	Reserved	-	-	-
14	x'04038'	Reserved	-	-	-
15	x'0403C'	Timer 6 interrupt	TM6IRQ	TM6ICR	x'03FEF'
16	x'04040'	Time base interrupt	TBIRQ	TBICR	x'03FF0'
17	x'04044'	Timer 7 interrupt	TM7IRQ	TM7ICR	x'03FF1'
18	x'04048'	Timer 7 compare2-match interrupt	T7OC2IRQ	T7OC2ICR	x'03FF2'
19	x'0404C'	Reserved	-	-	-
20	x'04050'	Reserved	-	-	-
21	x'04054'	Serial interface 0 UART reception interrupt	SC0RIRQ	SC0RICR	x'03FF5'
22	x'04058'	Serial interface 0 interrupt	SC0TIRQ	SC0TICR	x'03FF6'
23	x'0405C'	Reserved	-	-	-
24	x'04060'	Reserved	-	-	-
25	x'04064'	Reserved	-	-	-
26	x'04068'	A/D converter interrupt	ADIRQ	ADICR	x'03FFA'
27	x'0406C'	Reserved	-	-	-
28	x'04070'	Reserved	-	-	-
29	x'04074'	Reserved	-	-	-
30	x'04078'	Reserved	-	-	-

For unused interrupts and reserved interrupts, set the address on which the RTI instruction is described to the corresponded address.

■Interrupt Level and Priority

This LSI allocated vector numbers and interrupt control registers (except reset interrupt) to each interrupt. The interrupt level (except reset interrupt, non-maskable interrupt) can be set by software, per each interrupt group. There are three hierarchical interrupt levels. If multiple interrupts have the same priority, the one with the lowest vector number takes priority. For example, if a vector 3 set to level 1 and a vector 4 set to level 2 request interrupts simultaneously, vector 3 will be accepted.



Priority	Interrupt vector No.
1	Vector 1
2	Vector 2
3	Vector 5
4	Vector 6
5	Vector 3
6	Vector 4
7	Vector 8

Figure 3-1-3 Interrupt Priority Outline

■Determination of Interrupt Acceptance

The following is the procedure from interrupt request input to acceptance.

(1) The interrupt request flag (xxxIR) in the corresponding external interrupt control register(IRQnICR) or internal interrupt control register (xxxICR) is set to '1'.

(2) An interrupt request is input to the CPU, If the interrupt enable flag (xxxIE) in the same register is '1'.

(3) The interrupt level (IL) is set for each interrupt. The interrupt level (IL) is input to the CPU.

(4) The interrupt request is accepted, if IL has higher priority than IM and MIE is '1
[☞ Chapter 2. 2-1-7 Processor Status Word]

(5) After the interrupt is accepted, the hardware resets the interrupt request flag (xxxIR) in the interrupt control register (xxxICR) to '0'.

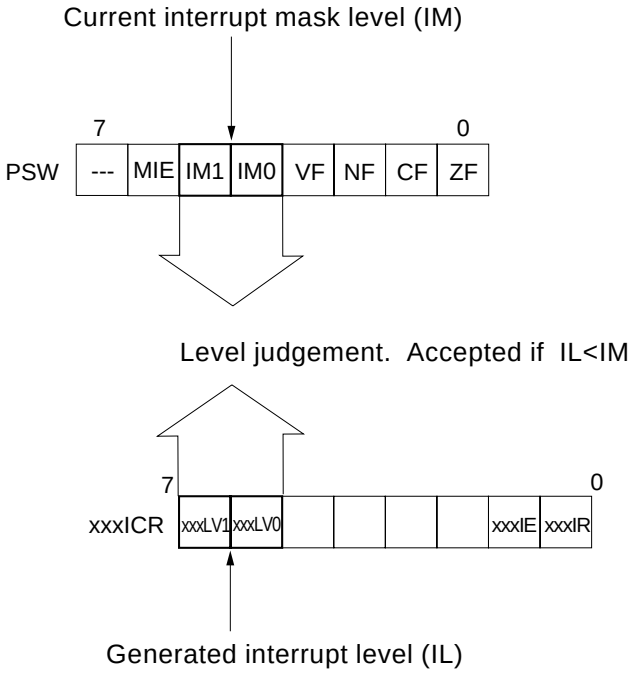


Figure 3-1-4 Determination of Interrupt Acceptance

The corresponding interrupt enable flag (xxxIE) is not cleared to "0", even if the interrupt is accepted.

When the setting is as xxxLV=1, XXXLV0=1, the interrupt of that vector is disabled, regardless of the value of xxxIE, xxxIR.

MIE='0' and interrupts are disabled when:

- MIE in the PSW is reset to '0' by a program
- Reset is detected

MIE='1' and interrupts are enabled when:

- MIE in the PSW is set to '1' by a program

The interrupt mask level (IM=IM1 - IM0) in the processor status word (PSW) changes when:

- The program alters it directly,
- A reset initializes it to 0 (00b),
- The hardware accepts and thus switches to the interrupt level (IL) for a maskable interrupt, or
- Execution of the RTI instruction at the end of an interrupt service routine restores the processor status word (PSW) and thus the previous interrupt mask level.

The maskable interrupt enable (MIE) flag in the processor status word (PSW) is not cleared to "0".

Non-maskable interrupts have priority over maskable ones.

■Interrupt Acceptance Operation

When accepting an interrupt, this LSI hardware saves the handy address register, the return address from the program counter, and the processor status word (PSW) to the stack and branches to the interrupt handler using the starting address in the vector table.

The following is the hardware processing sequence after by interrupt acceptance.

1. The stack pointer (SP) is updated.
   (SP-6 → SP)
2. The contents of the handy address register (HA) are saved to the stack.
   Upper half of HA → (SP+5)
   Lower half of HA → (SP+4)
3. The contents of the program counter (PC), the return address, are saved to the stack.
   PC bits 18, 17, and 0 → (SP+3)
   PC bits 16-9 → (SP+2)
   PC bits 8-1 → (SP+1)
4. The contents of the PSW are saved to the stack.
   PSW → (SP)
5. The interrupt level (xxxLVn) for the interrupt is copied to the interrupt mask (IMn) in the PSW.
   Interrupt level (xxxLVn) → IMn
6. The hardware branches to the address in the vector table.

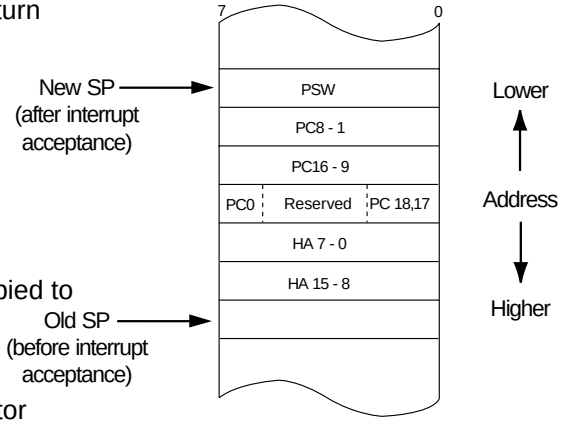


Figure 3-1-5 Stack Operation during interrupt acceptance

■Interrupt Return Operation

An interrupt handler ends by restoring, using the POP instruction and other means, the contents of any registers used during processing and then executing the return from interrupt (RTI) instruction to return to the point at which execution was interrupted.

The following is the processing sequence after the RTI instruction.

1. The contents of the PSW are restored from the stack. (SP)
2. The contents of the program counter (PC), the return address, are restored from the stack. (SP+1 to SP+3)
3. The contents of the handy address register (HA) are restored from the stack. (SP+4, SP+5)
4. The stack pointer is updated. (SP+6 → SP)
5. Execution branches to the address in the program counter.

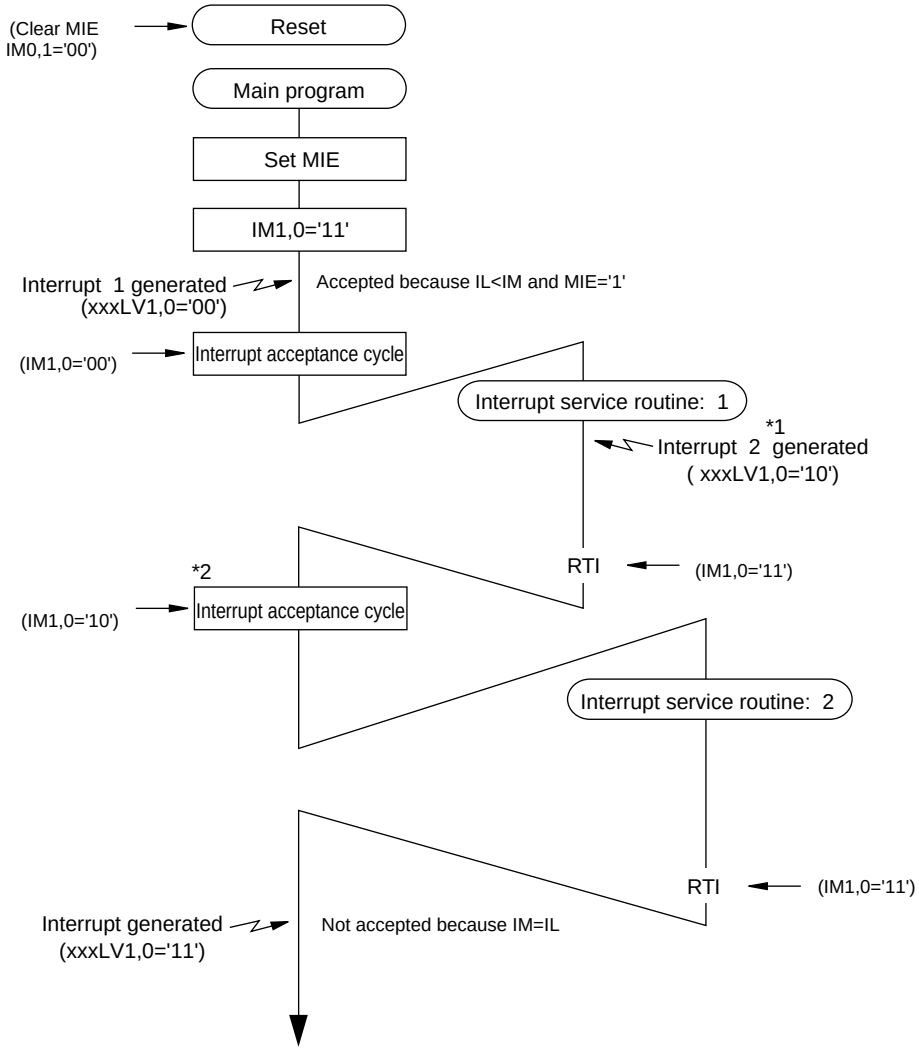
The handy address register is an internal register used by the handy addressing function. The hardware saves its contents to the stack to prevent the interrupt from interfering with operation of the function.

Registers such as data register, or address register are not saved, so that PUSH instruction should be used to save data register or address register onto the stack, if neccessary.

The address bp6 to bp2, when program counter (PC) are saved to the stack, are reserved. Do not change by program.

■Maskable Interrupt

Figure 3-1-6 shows the processing flow when a second interrupt with a lower priority level (xxxLV1-xxxLV0='10') arrives during the processing of one with a higher priority level (xxxLV1-xxxLV0='00').



Parentheses ( ) indicate hardware processing.

*1 If during the processing of the first interrupt, an interrupt request with an interrupt level (IL) numerically lower than the interrupt mask (IM) arrives, it is accepted as a nested interrupt. If IL ≥ IM, however, the interrupt is not accepted.

*2 The second interrupt, postponed because its interrupt level (IL) was numerically greater than the interrupt mask (IM) for the first interrupt service routine, is accepted when the first interrupt handler returns.

Figure 3-1-6 Processing Sequence for Maskable Interrupts

■Multiplex Interrupt

When an MN101C539 series device accepts an interrupt, it automatically disables acceptance of subsequent interrupts with the same or lower priority level. When the hardware accepts an interrupt, it copies the interrupt level (xxxLVn) for the interrupt to the interrupt mask (IM) in the PSW. As a result, subsequent interrupts with the same or lower priority levels are automatically masked. Only interrupts with higher priority levels are accepted. The net result is that interrupts are normally processed in decreasing order of priority. It is, however, possible to alter this arrangement.

1. To disable interrupt nesting
   - Reset the MIE bit in the PSW to "0."
   - Raise the priority level of the interrupt mask (IM) in the PSW.
2. To enable interrupts with lower priority than the currently accepted interrupt
   - Lower the priority level of the interrupt mask (IM) in the PSW.

Multiplex interrupts are only enabled for interrupts with levels higher than the PSW interrupt mask level (IM).

It is possible to forcibly rewrite IM to accept an interrupt with a priority lower than the interrupt being processed, but be careful of stack overflow.

Do not operate the maskable interrupt control register (xxxICR) when multiple interrupts are enabled. If operation is necessary, first clear the PSW MIE flag to disable interrupts.

Figure 3-1-7 shows the processing flow for multiple interrupts (interrupt 1: xxxLV1-xxxLV0='10', and interrupt 2: xxxLV1-xxxLV0='00').

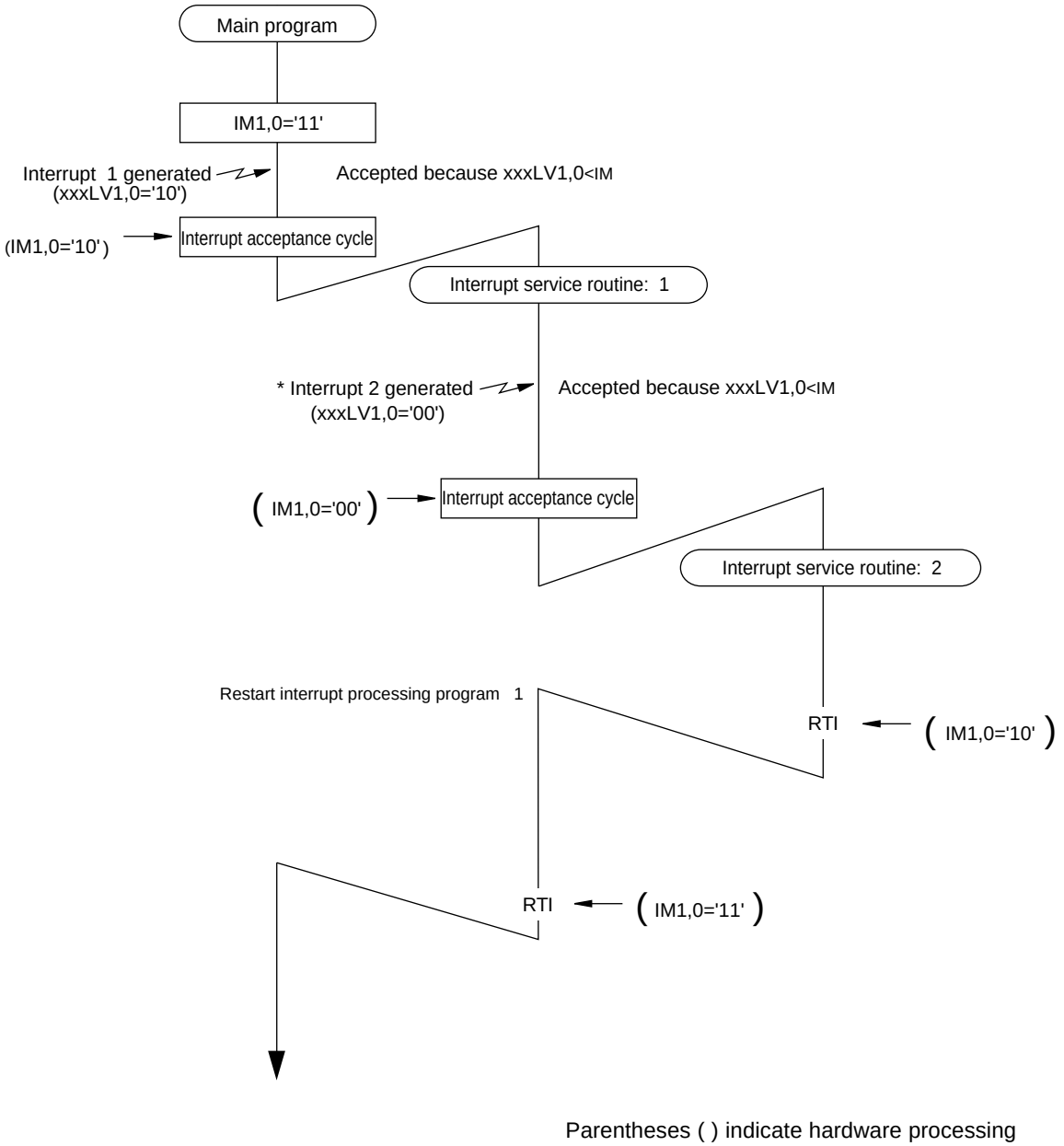


Figure 3-1-7 Processing Sequence with Multiple Interrupts Enabled

3-1-4 Interrupt Flag Setup

■ Interrupt request flag (IR) setup by the software

The interrupt request flag is operated by the hardware. That is set to "1" when any interrupt factor is generated, and cleared to "0" when the interrupt is accepted. If you want to operate it by the software, the IRWE flag of MEMCTR should be set to "1".

■ Interrupt flag setup procedure

A setup procedure of the interrupt request flag set by the hardware and the software shows as follows ;

Setup Procedure	Description
(1) Disable all maskable interrupts. PSW bp6 : MIE = 0	(1) Clear the MIE flag of PSW to disable all maskable interrupts. This is necessary, especially when the interrupt control register is changed.
(2) Select the interrupt factor.	(2) Select the interrupt factor such as interrupt edge selection, or timer interrupt cycle change.
(3) Enable the interrupt request flag to be rewritten. MEMCTR (x'3F01') bp2 : IRWE = 1	(3) Set the IRWE flag of MEMCTR to enable the interrupt request flag to be rewritten. This is necessary only when the interrupt request flag is changed by the software.
(4) Rewrite the interrupt request flag. xxxICR bp0 : xxxIR	(4) Rewrite the interrupt request flag (xxxIR) of the interrupt control register (xxxICR).
(5) Disable the interrupt request flag to be rewritten. MEMCTR (x'3F01') bp2 : IRWE = 0	(5) Clear the IRWE flag so that interrupt request flag can not be rewritten by the software.
(6) Set the interrupt level. xxxICR bp7-6 : xxxLV1-0 PSW bp5-4 : IM1-0	(6) Set the interrupt level by the xxxLV1-0 flag of the interrupt control register (xxxICR). Set the IM1-0 flag of PSW when the interrupt acceptance level of CPU should be changed.
(7) Enable the interrupt. xxxICR bp1 : xxxIE = 1	(7) Set the xxxIE flag of the interrupt control register (xxxICR) to enable the interrupt.
(8) Enable all maskable interrupts. PSW bp6 : MIE = 1	(8) Set the MIE flag of PSW to enable maskable interrupts.

3-2 Control Registers

3-2-1 Registers List

Table 3-2-1 Interrupt Control Registers

Register	Address	R/W	Functions	Page
NMICR	x'03FE1'	R/W	Non-maskable interrupt control register	III - 16
IRQ0ICR	x'03FE2'	R/W	External interrupt 0 control register	III - 17
IRQ1ICR	x'03FE3'	R/W	External interrupt 1 control register	III - 18
IRQ2ICR	x'03FE4'	R/W	External interrupt 2 control register	III - 19
IRQ3ICR	x'03FE5'	R/W	External interrupt 3 control register	III - 20
TM2ICR	x'03FEB'	R/W	Timer 2 interrupt control register (Timer 2 compare match)	III - 21
TM3ICR	x'03FEC'	R/W	Timer 3 interrupt control register (Timer 3 compare match)	III - 22
TM6ICR	x'03FEF'	R/W	Timer 6 interrupt control register (Timer 6 compare match)	III - 23
TBICR	x'03FF0'	R/W	Time base interrupt control register (Time base period)	III - 24
TM7ICR	x'03FF1'	R/W	Timer 7 interrupt control register (Timer 7 interrupt)	III - 25
T7OC2ICR	x'03FF2'	R/W	Timer 7 compare register 2-match interrupt control register	III - 26
SC0RICR	x'03FF5'	R/W	Serial interface 0 UART reception interrupt control register (Serial interface 0 UART reception interrupt)	III - 27
SC0TICR	x'03FF6'	R/W	Serial interface 0 interrupt control register (Serial interface 0 interrupt)	III - 28
ADICR	x'03FFA'	R/W	A/D conversion interrupt control register (A/D converter interrupt)	III - 29

Writing to the interrupt control register should be done after that all maskable interrupts are set to be disabled by the MIE flag of the PSW register.

If the interrupt level flag (xxxLVn) is set to "level 3", its vector is disabled, regardless of interrupt enable flag and interrupt request flag.

3-2-2 Interrupt Control Registers

The interrupt control registers include the maskable interrupt control registers (xxxICR) and the non-maskable interrupt control register (NMICR).

■Non-Maskable Interrupt Control Register (NMICR address: x'03FE1')

The non-maskable interrupt control register (NMICR) stores the non maskable interrupt request. When the non-maskable interrupt request is generated, the interrupt is accepted regardless of the interrupt mask level (IMn) of PSW. The hardware then branches to the address stored at location x'04004' in the interrupt vector table. The watchdog timer overflow interrupt request flag (WDIR) is set to "1" when the watchdog timer overflows. The program interrupt request flag (PIR) is set to "1" when the undefined instruction is executed.

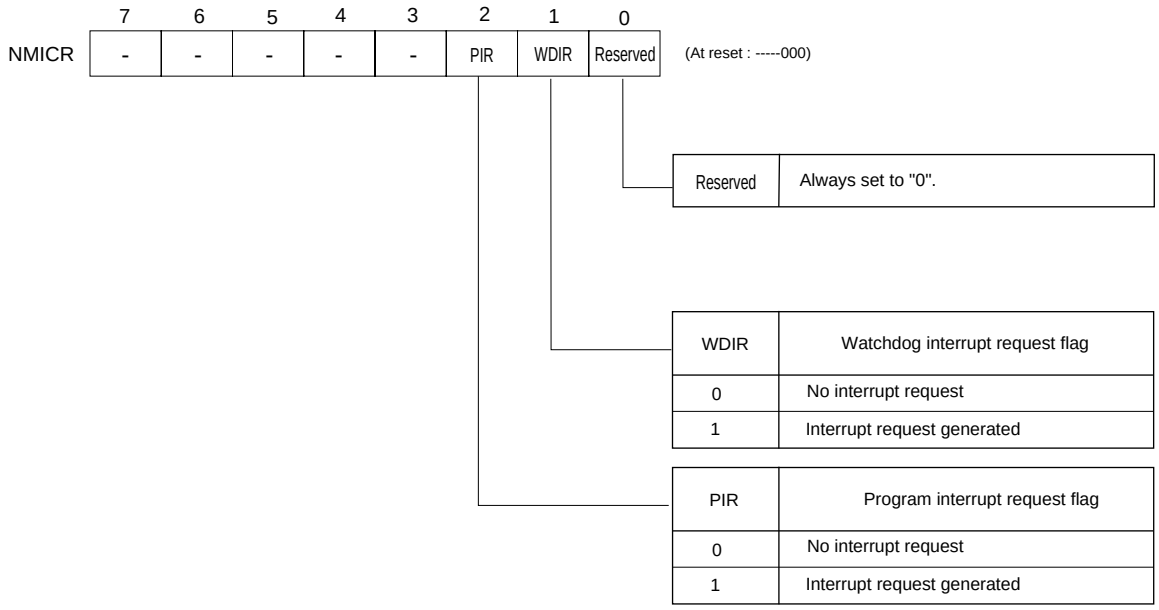


Figure 3-2-1 Non-Maskable Interrupt Control Register (NMICR:x'03FE1', R/W)

On this LSI, when undefined instruction is decoded, the program interrupt request flag (PIR) is set to "1", and the non-maskable interrupt is generated.
If the PIR flag setup is confirmed by the non-maskable interrupt service routine, the reset via the software is recommended, When software reset, the reset pin (p27) outputs "0".

Once the WDIR flag becomes "1" after non-maskable interrupt happens, only the program can clear it to "0".

■External Interrupt 0 Control Register (IRQ0ICR)

The external interrupt 0 control register (IRQ0ICR) controls interrupt level of the external interrupt 0, active edge, interrupt enable and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

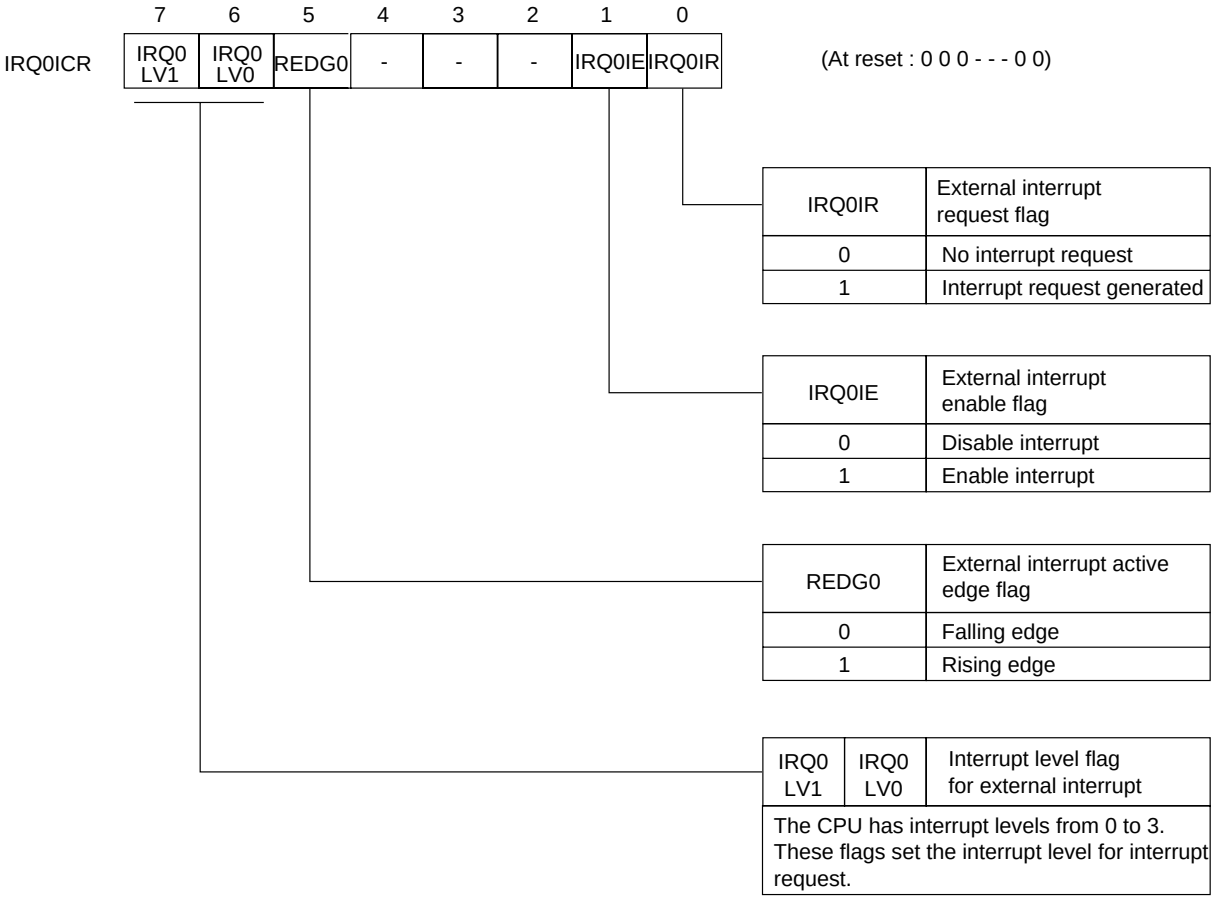


Figure 3-2-2 External Interrupt 0 Control Register (IRQ0ICR : x'03FE2', R/W)

■External Interrupt 1 Control Register (IRQ1ICR)

The external interrupt 1 control register (IRQ1ICR) controls interrupt level of external interrupt 1, active edge, interrupt enable and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

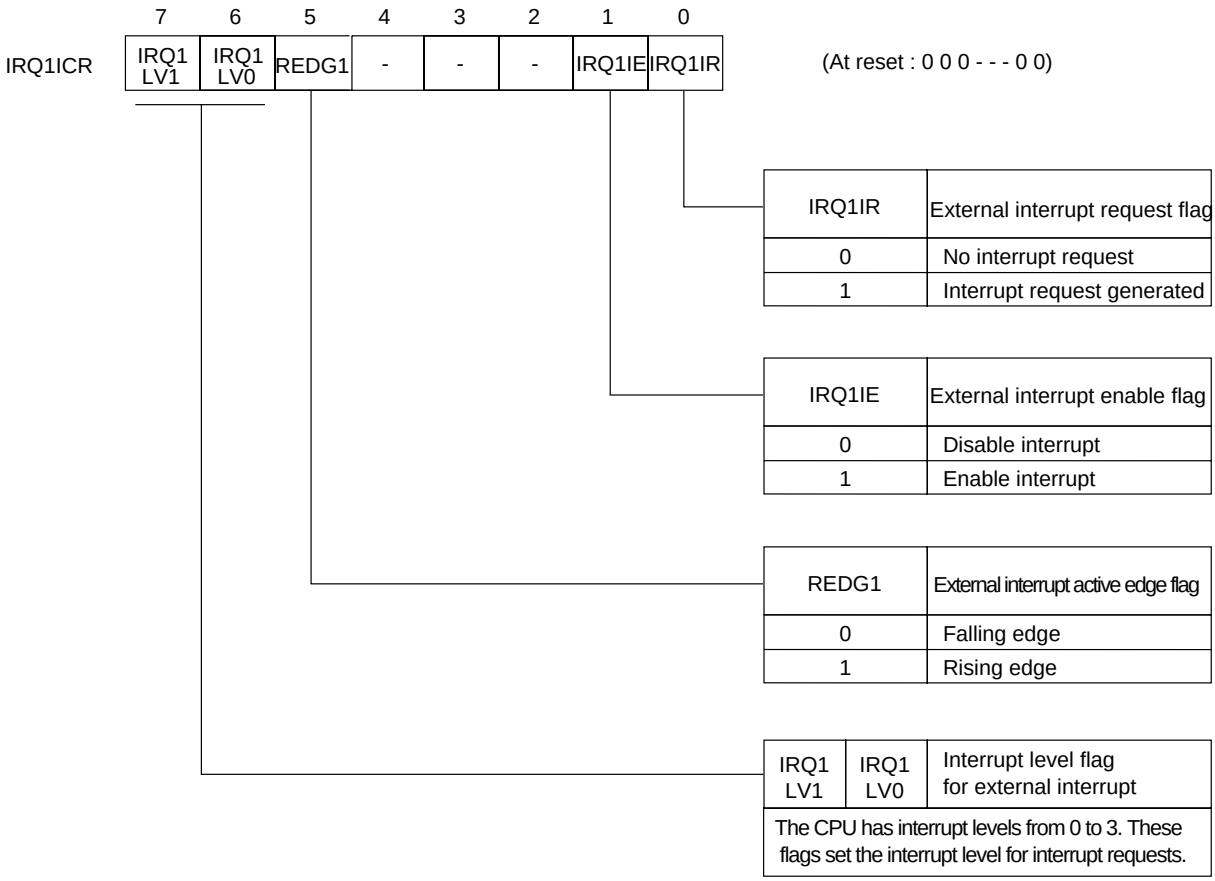


Figure 3-2-3 External Interrupt 1 Control Register (IRQ1ICR : x'03FE3', R/W)

■External Interrupt 2 Control Register (IRQ2ICR)

The external interrupt 2 control register (IRQ2ICR) controls interrupt level of external interrupt 2, active edge, interrupt enable and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

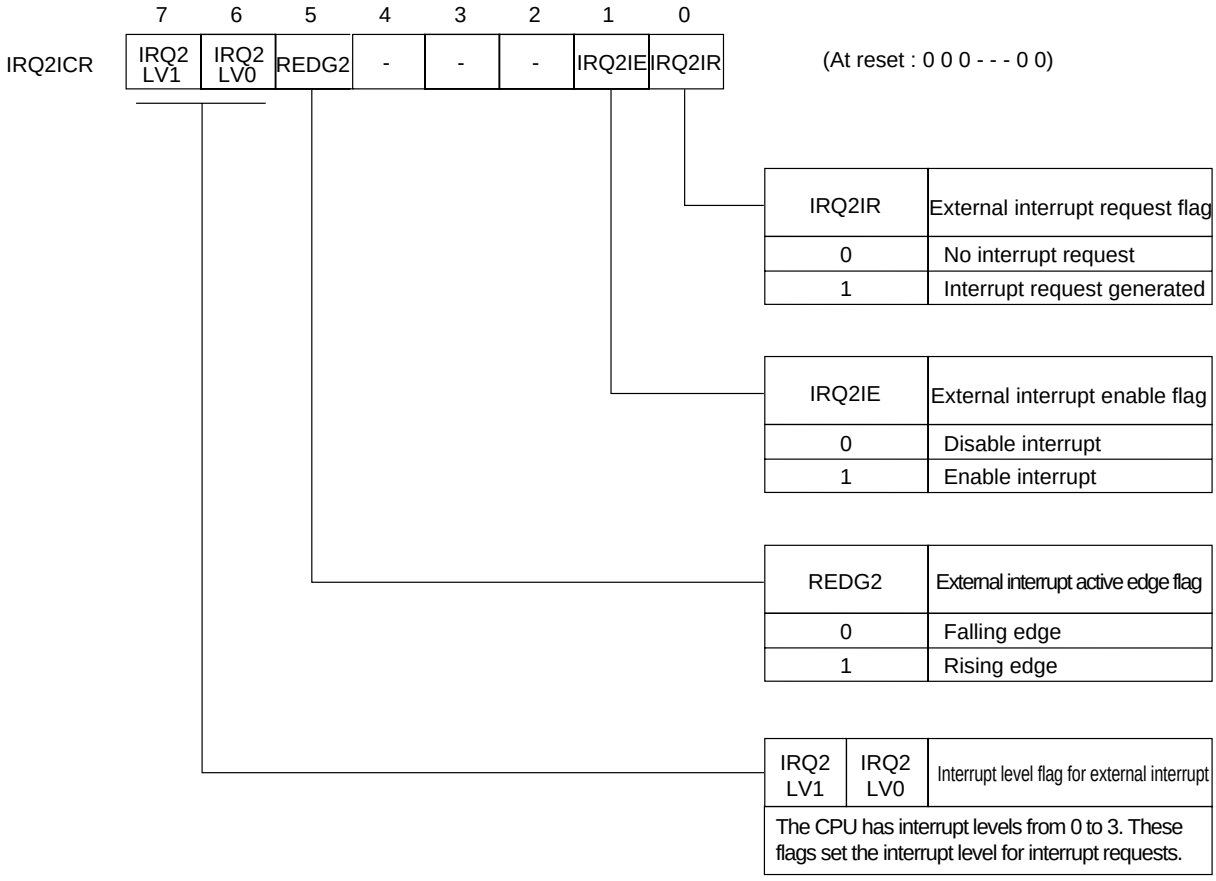


Figure 3-2-4 External Interrupt 2 Control Register (IRQ2ICR : x'03FE4', R/W)

■External Interrupt 3 Control Register (IRQ3ICR)

The external interrupt 3 control register (IRQ3ICR) controls interrupt level of external interrupt 3, active edge, interrupt enable flag and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

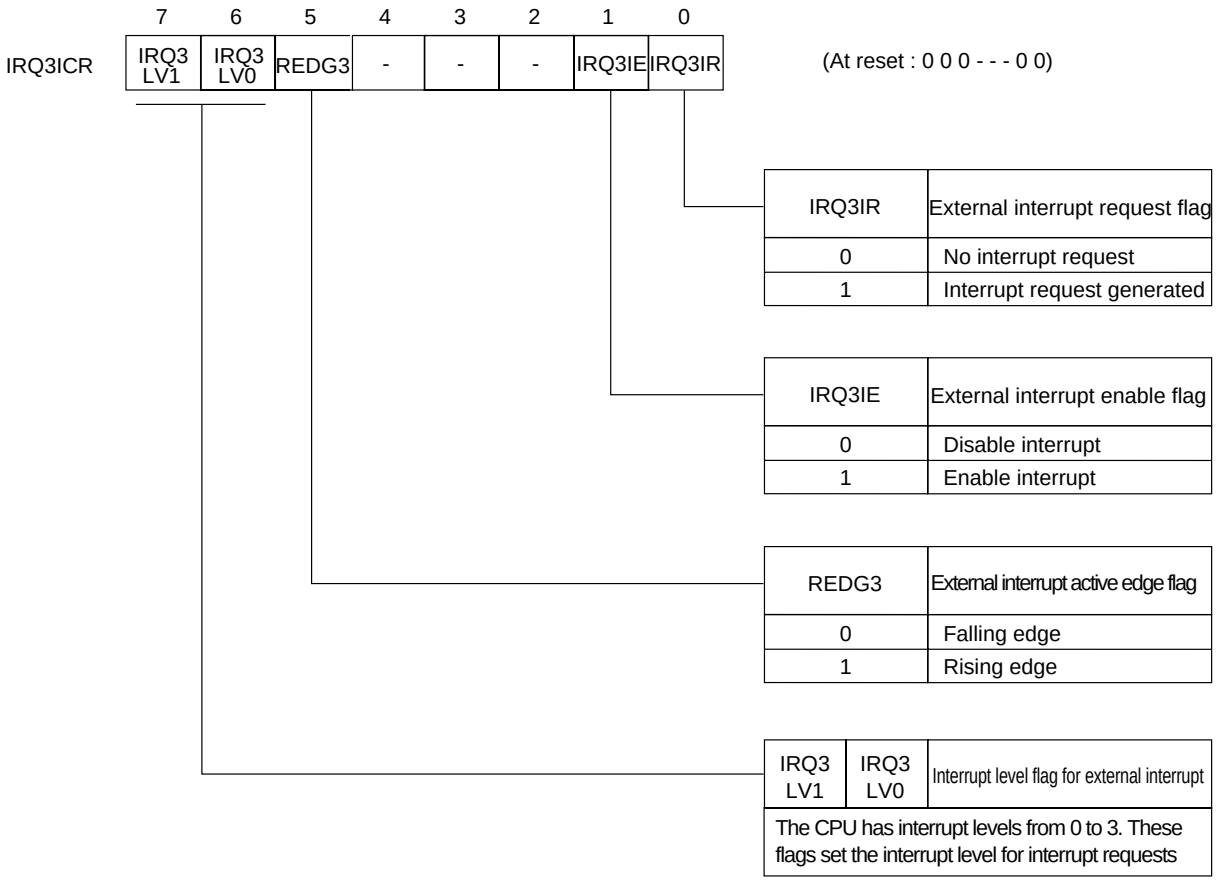


Figure 3-2-5 External Interrupt 3 Control Register (IRQ3ICR : x'03FE5', R/W)

■Timer 2 Interrupt Control Register (TM2ICR)

The timer 2 interrupt control register (TM2ICR) controls interrupt level of timer 2 interrupt, interrupt enable flag and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

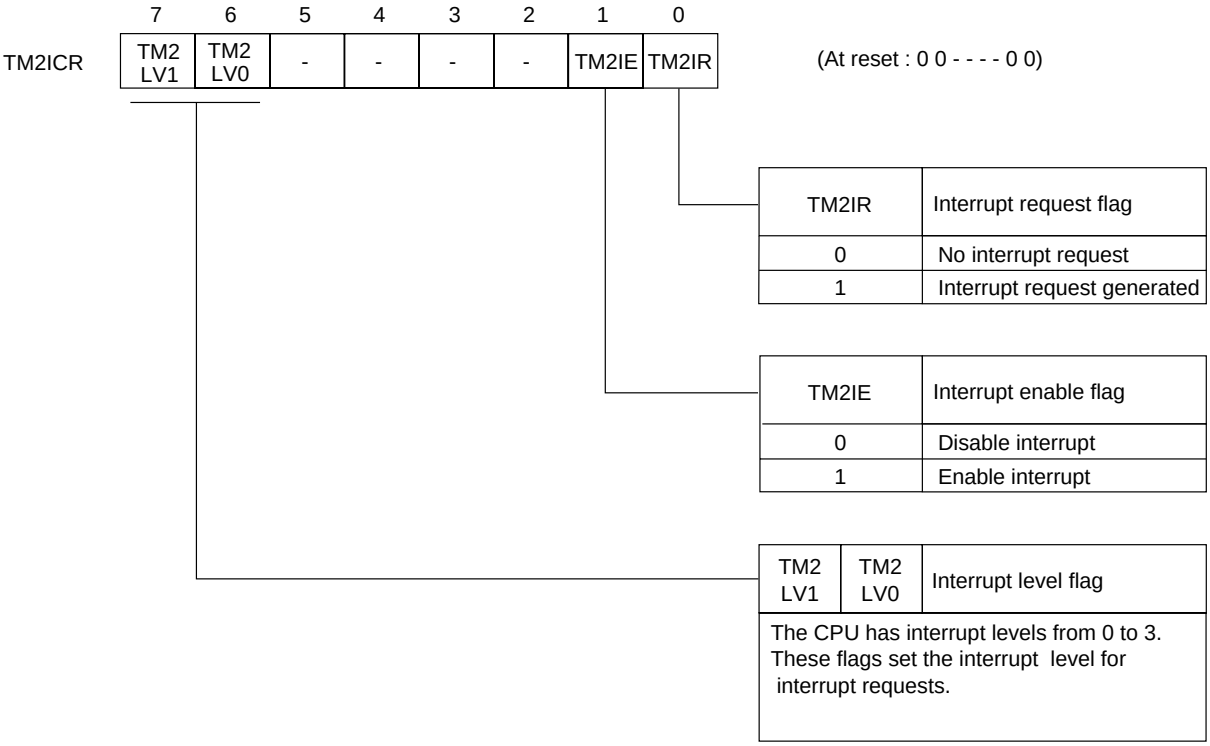


Figure 3-2-6 Timer 2 Interrupt Control Register (TM2ICR : x'03FEB', R/W)

■Timer 3 Interrupt Control Register (TM3ICR)

The timer 3 interrupt control register (TM3ICR) controls interrupt level of timer 3 interrupt, interrupt enable flag and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

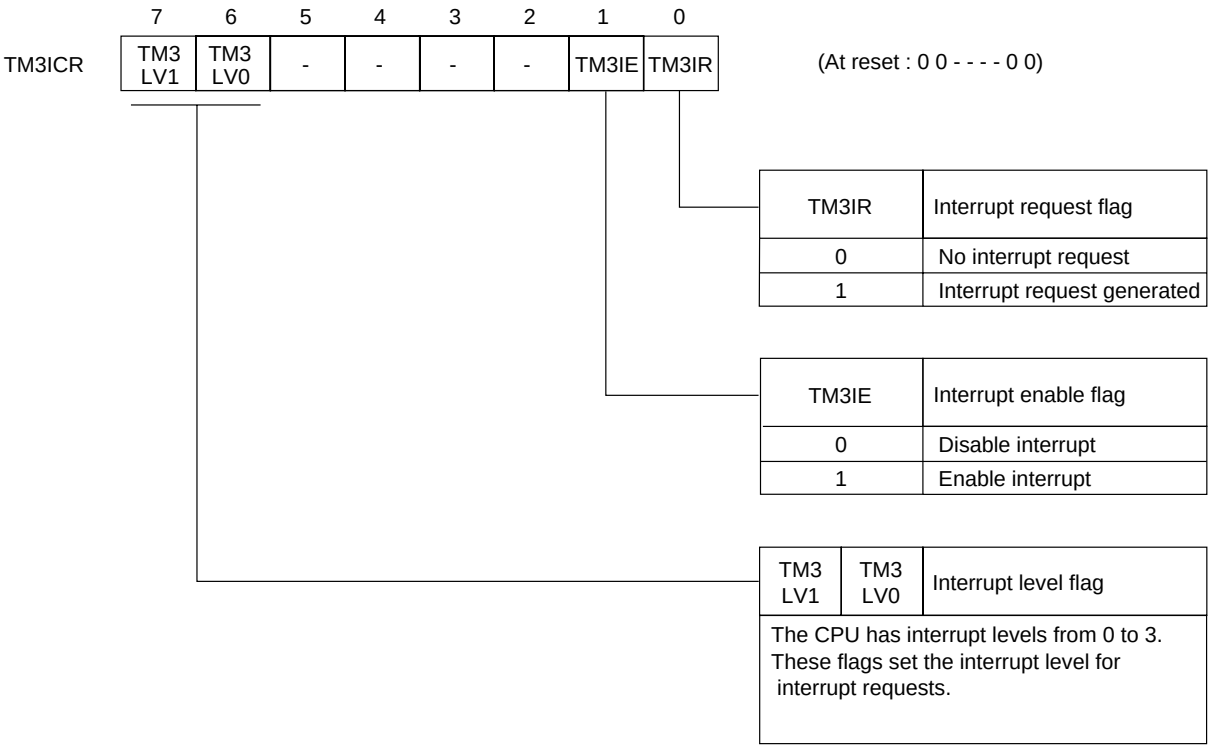


Figure 3-2-7 Timer 3 Interrupt Control Register (TM3ICR : x'03FEC', R/W)

■Timer 6 Interrupt Control Register (TM6ICR)

The timer 6 interrupt control register (TM6ICR) controls interrupt level of timer 6 interrupt, interrupt enable flag and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

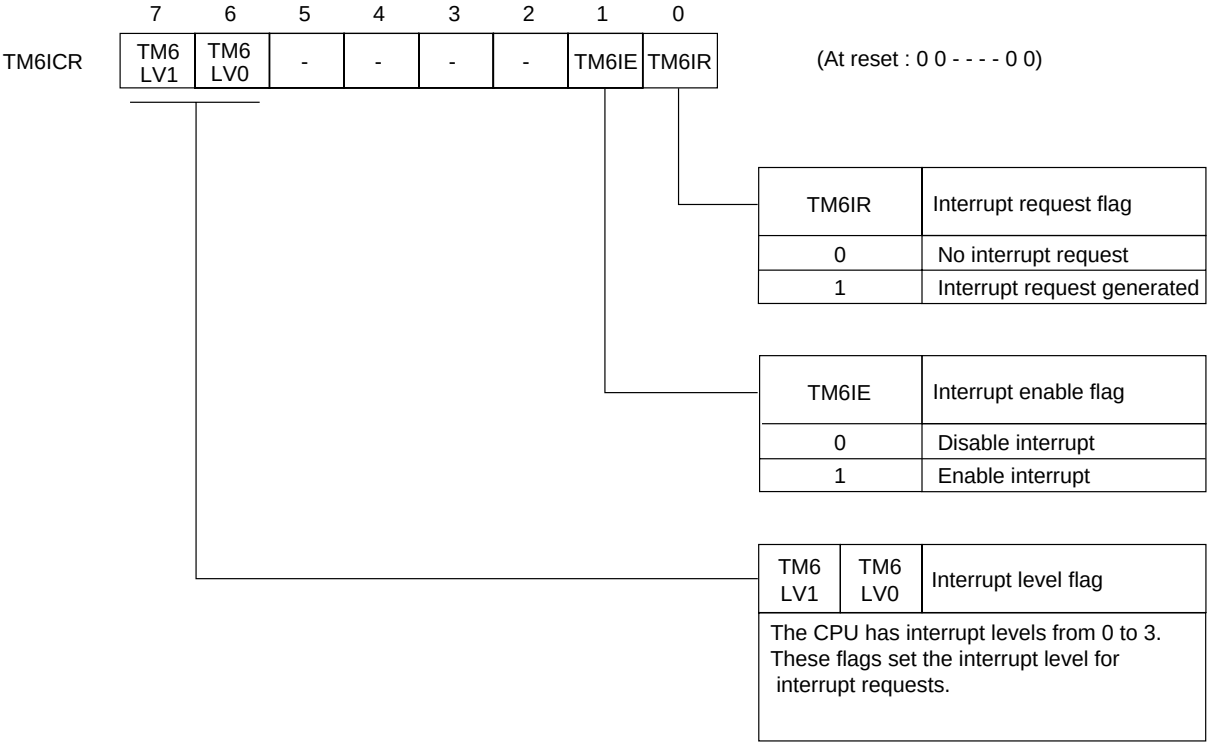


Figure 3-2-8 Timer 6 Interrupt Control Register (TM6ICR : x'03FEF', R/W)

■Time Base Interrupt Control Register (TBICR)

The time base interrupt control register (TBICR) controls interrupt level of time base interrupt, interrupt enable flag and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

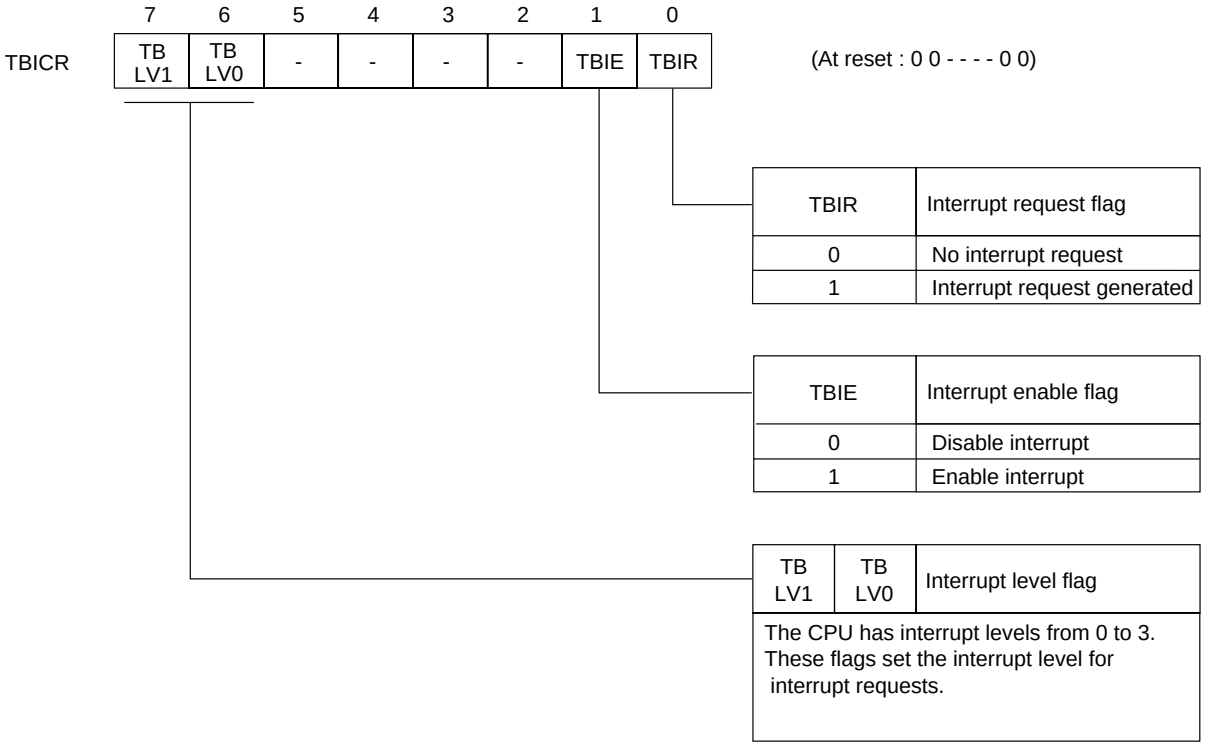


Figure 3-2-9 Time Base Interrupt Control Register (TBICR : x'03FF0', R/W)

■Timer 7 Interrupt Control Register (TM7ICR)

The timer 7 interrupt control register (TM7ICR) controls interrupt level of timer 7 interrupt, interrupt enable flag and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

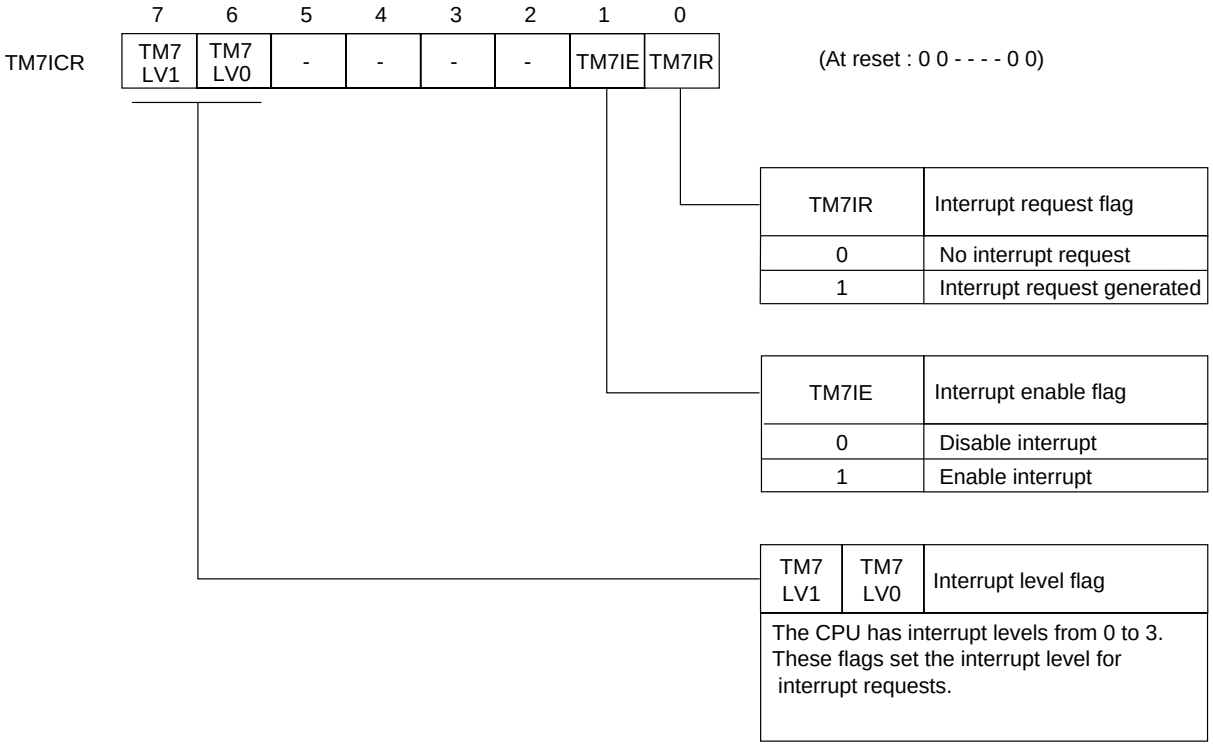


Figure 3-2-10 Timer 7 Interrupt Control Register (TM7ICR : x'03FF1', R/W)

■Timer 7 Compare Register 2-match Interrupt Control Register (TOC2ICR)

The timer 7 compare register 2-match interrupt control register (TOC2ICR) controls interrupt level of timer 7 compare register 2-match interrupt , interrupt enable flag and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

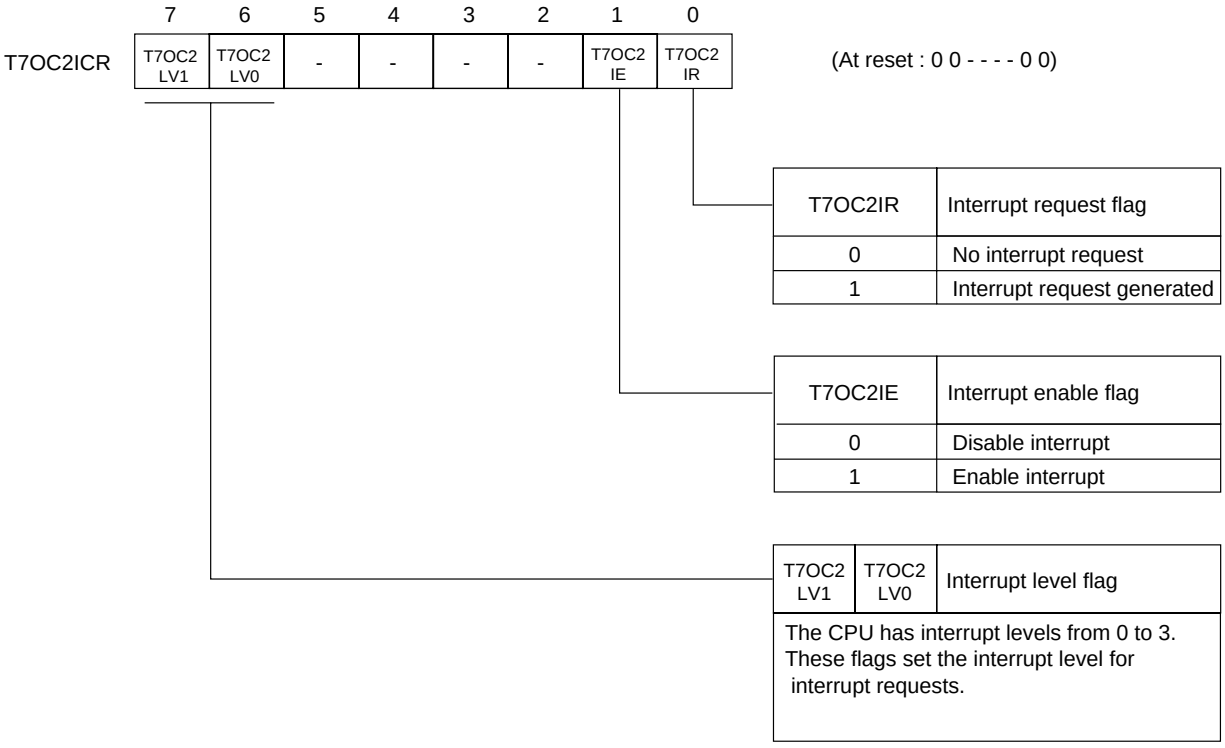


Figure 3-2-11 Timer 7 Compare Register 2-match Interrupt Control Register (TMOC2ICR : x'03FF2', R/W)

■Serial Interface 0 UART Interrupt Control Register (SC0RICR)

The serial Interface 0 UART reception interrupt control register (SC0RICR) controls interrupt level of serial Interface 0 UART reception interrupt, interrupt enable flag and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

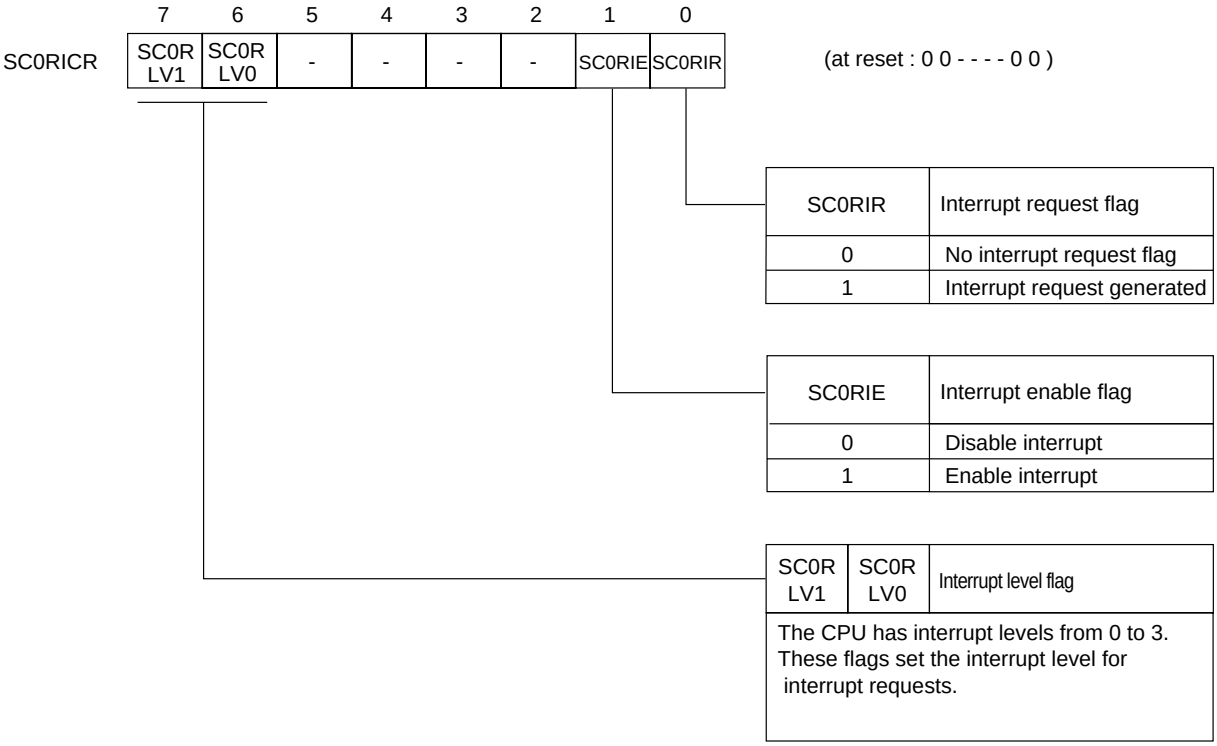


Figure 3-2-12 Serial Interface 0 UART Reception Interrupt Control register (SC0RICR:x'03FF5', R/W)

■Serial Interface 0 Interrupt Control Register (SC0TICR)

The serial Interface 0 interrupt control register (SC0TICR) controls interrupt level of serial Iinterface 0 interrupt, interrupt enable flag and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

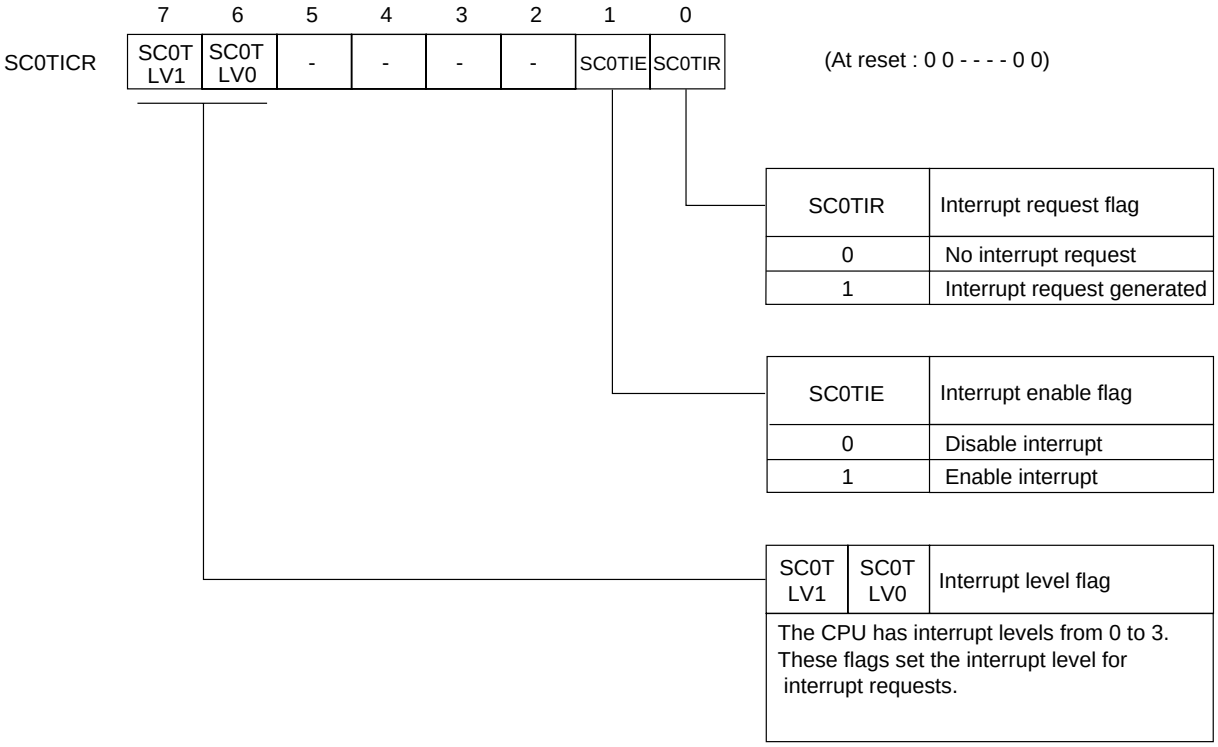


Figure 3-2-13 Serial Interface 0 Interrupt Control Register (SC0TICR : x'03FF6', R/W)

■A/D Converter Interrupt Control Register (ADICR)

The A/D converter interrupt control register (ADICR) controls interrupt level of A/D converter interrupt, interrupt enable flag and interrupt request. Interrupt control register should be operated when the maskable interrupt enable flag (MIE) of PSW is "0".

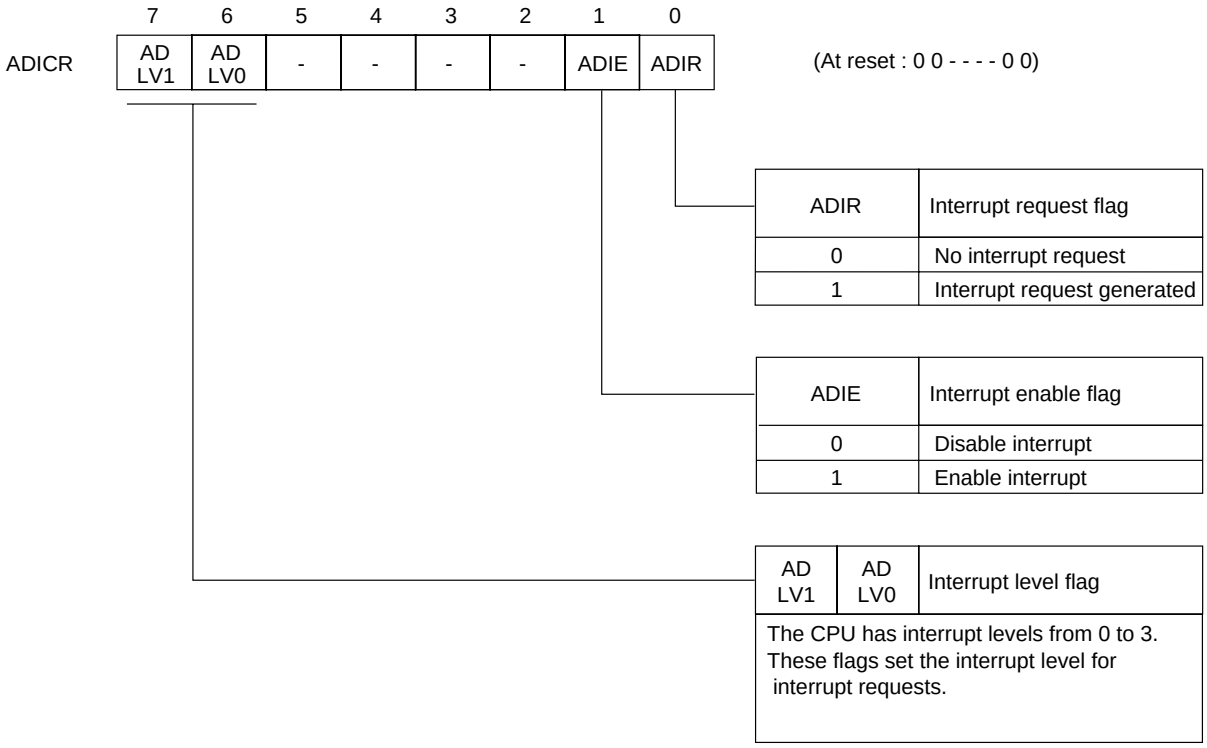


Figure 3-2-14 A/D Converter Interrupt Control Register (ADICR : x'03FFA', R/W)

3-3 External Interrupts

There are 4 external interrupts in this LSI. The circuit (external interrupt interface) for the external interrupt input signal, is built-in between the external interrupt input pin and the interrupt controller block. This external interrupt interface can manage to do with any kind of external interrupts.

3-3-1 Overview

Table 3-3-1 shows the list for functions which external interrupts 0 to 3 can be used.

Table 3-3-1 External Interrupt Functions

	External interrupt 0 (IRQ0)	External interrupt 1 (IRQ1)	External interrupt 2 (IRQ2)	External interrupt 3 (IRQ3)
External interrupt input pin	P20	P21	P22, P02	P23, P60 to P67
Programmable active edge interrupt	√	√	√ (P22)	√ (P23)
Both edges interrupt	√	√	-	-
Key input interrupt	-	-	-	√ (P60 to P67)
Noise filter built-in	√	√	-	-
AC zero cross detection	-	√	-	-

3-3-2 Block Diagram

■External Interrupt 0 Interface, External Interrupt 1 Interface, Block Diagram

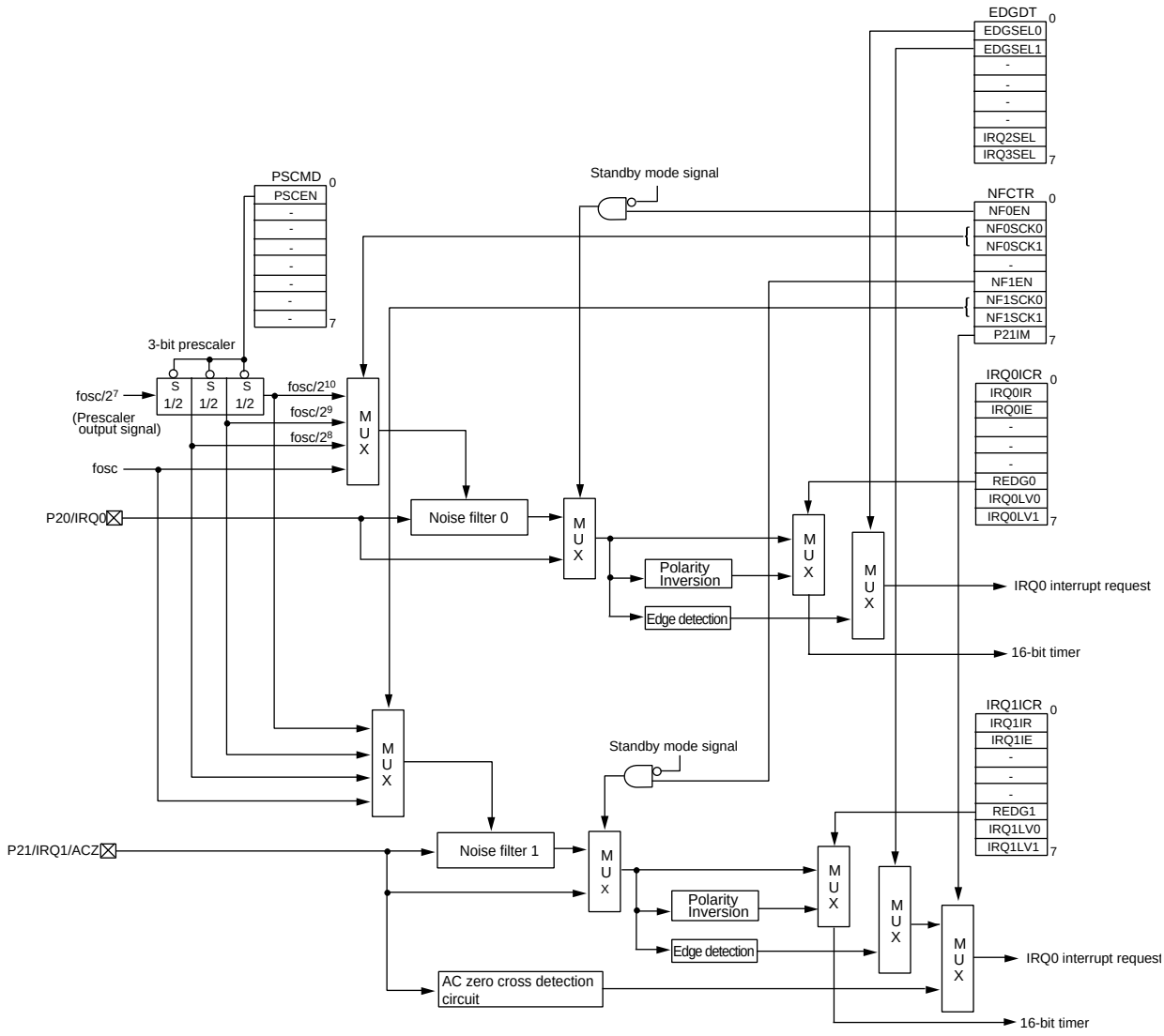


Figure 3-3-1 External Interrupt 0 Interface and External Interrupt 1 Interface Block Diagram

■External Interrupt 2 Interface Block Diagram

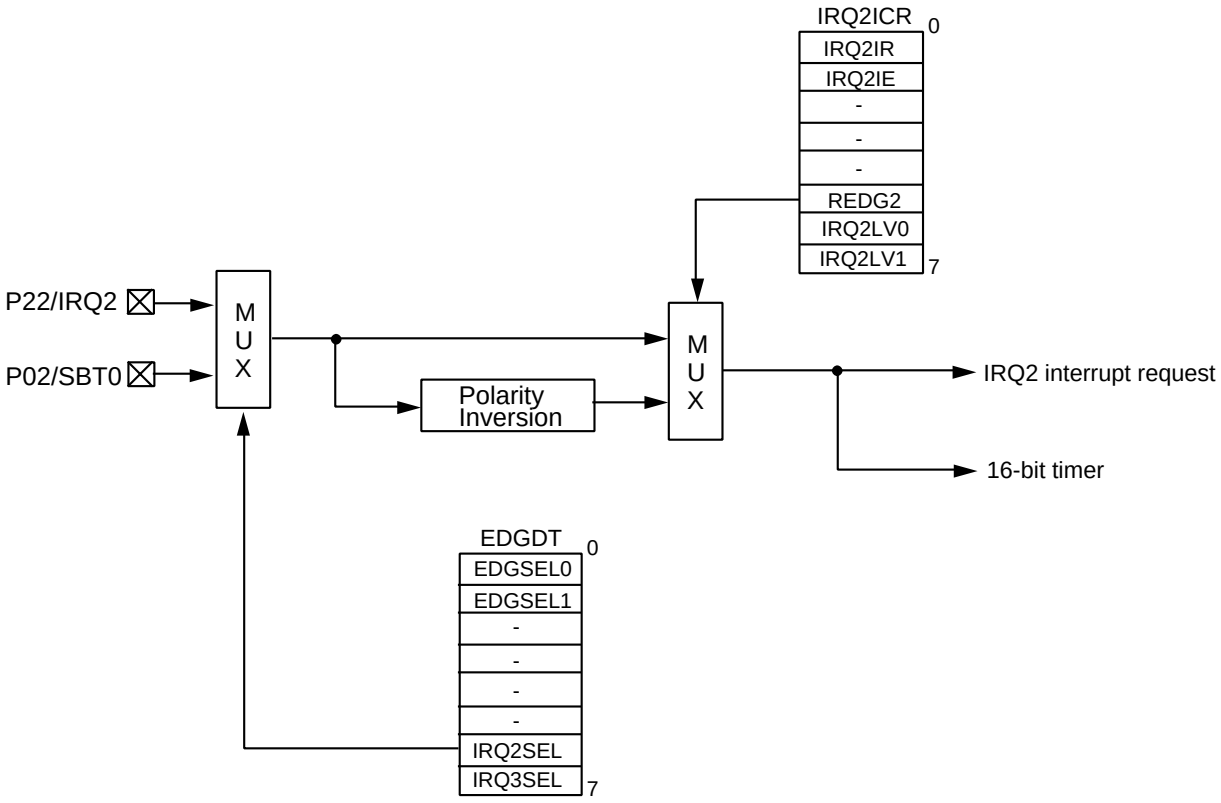


Figure 3-3-2 External Interrupt 2 Interface Block Diagram

■External Interrupt 3 Interface Block Diagram

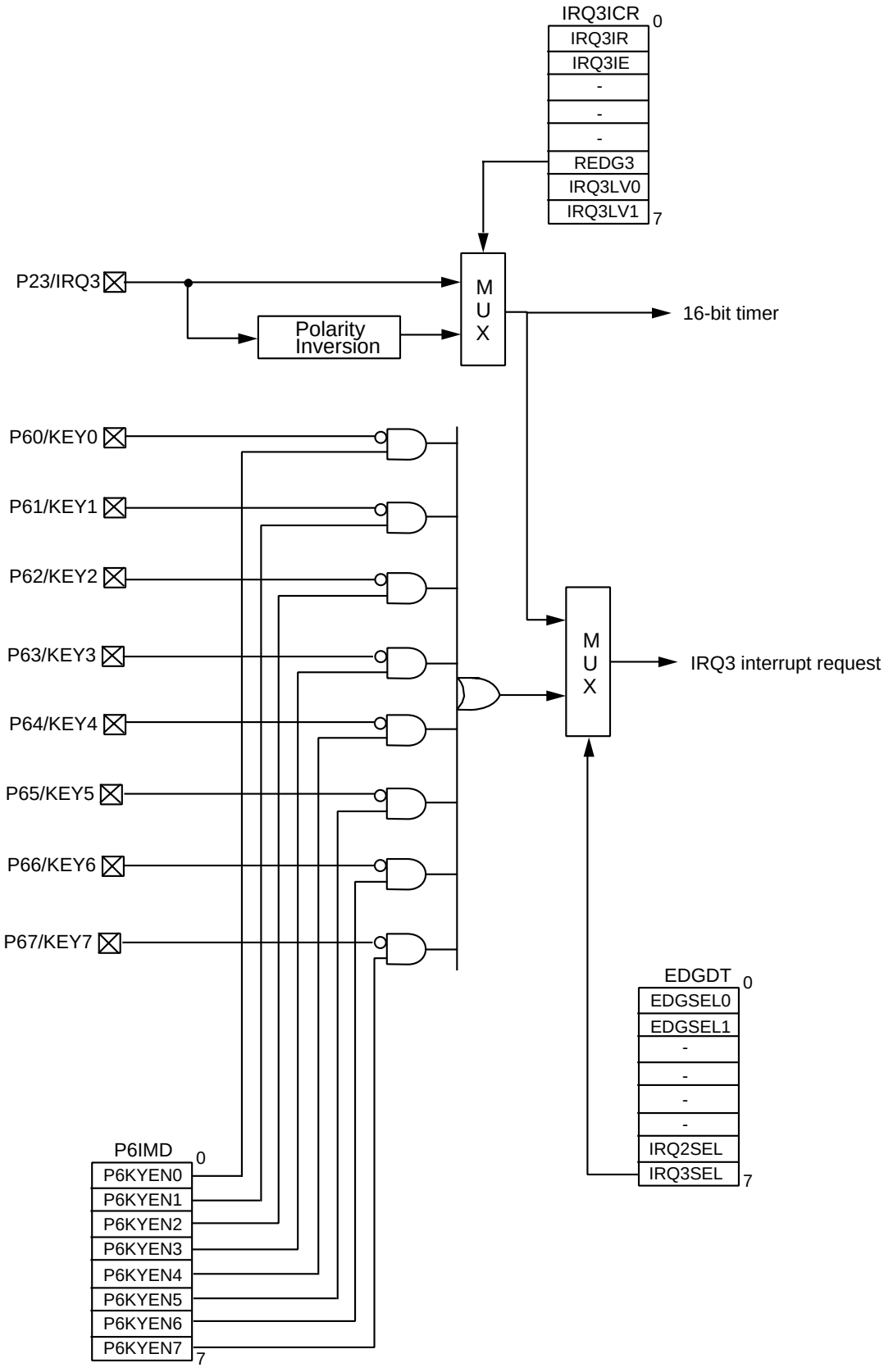


Figure 3-3-3 External Interrupt 4 Interface Block Diagram

3-3-3 Control Registers

The external interrupt input signal, which operated in each external interrupt 0 to 3 interface generate interrupt requests.

External interrupt 0 to 3 interface are controlled by the external interrupt control register (IRQnICR). And external interrupt interface 0 to 1 are controlled by the noise filter control register (NFCTR) and the both edges interrupt control register (EDGDT), external interrupt interface 3 is controlled by the port 6 key interrupt control register (P6IMD).

Table 3-3-2 shows the list of registers, control external interrupt 0 to 3.

Table 3-3-2 External Interrupt Control Register

External Interrupt	Register	Address	R/W	Function	Page
External interrupt 0	IRQ0ICR	x'03FE2'	R/W	External interrupt 0 control register	III -17
	NFCTR	x'03F8E'	R/W	Noise filter control register	III -35
	EDGDT	x'03F8F'	R/W	Both edges interrupt control register	III -36
External interrupt 1	IRQ1ICR	x'03FE3'	R/W	External interrupt 1 control register	III -18
	NFCTR	x'03F8E'	R/W	Noise filter control register	III -35
	EDGDT	x'03F8F'	R/W	Both edges interrupt control register	III -36
External interrupt 2	IRQ2ICR	x'03FE4'	R/W	External interrupt 2 control register	III -19
External interrupt 3	IRQ3ICR	x'03FE5'	R/W	External interrupt 3 control register	III -20
	P6IMD	x'03F4E'	R/W	Port6 key interrupt control register	III -37

R/W : Readable / Writable.

■Noise Filter Control Register (NFCTR)

The noise filter control register (NFCTR) sets the noise remove function for IRQ0 and IRQ1 and also selects the sampling cycle of noise remove function. And this register also set the AC zero cross detection function for IRQ1.

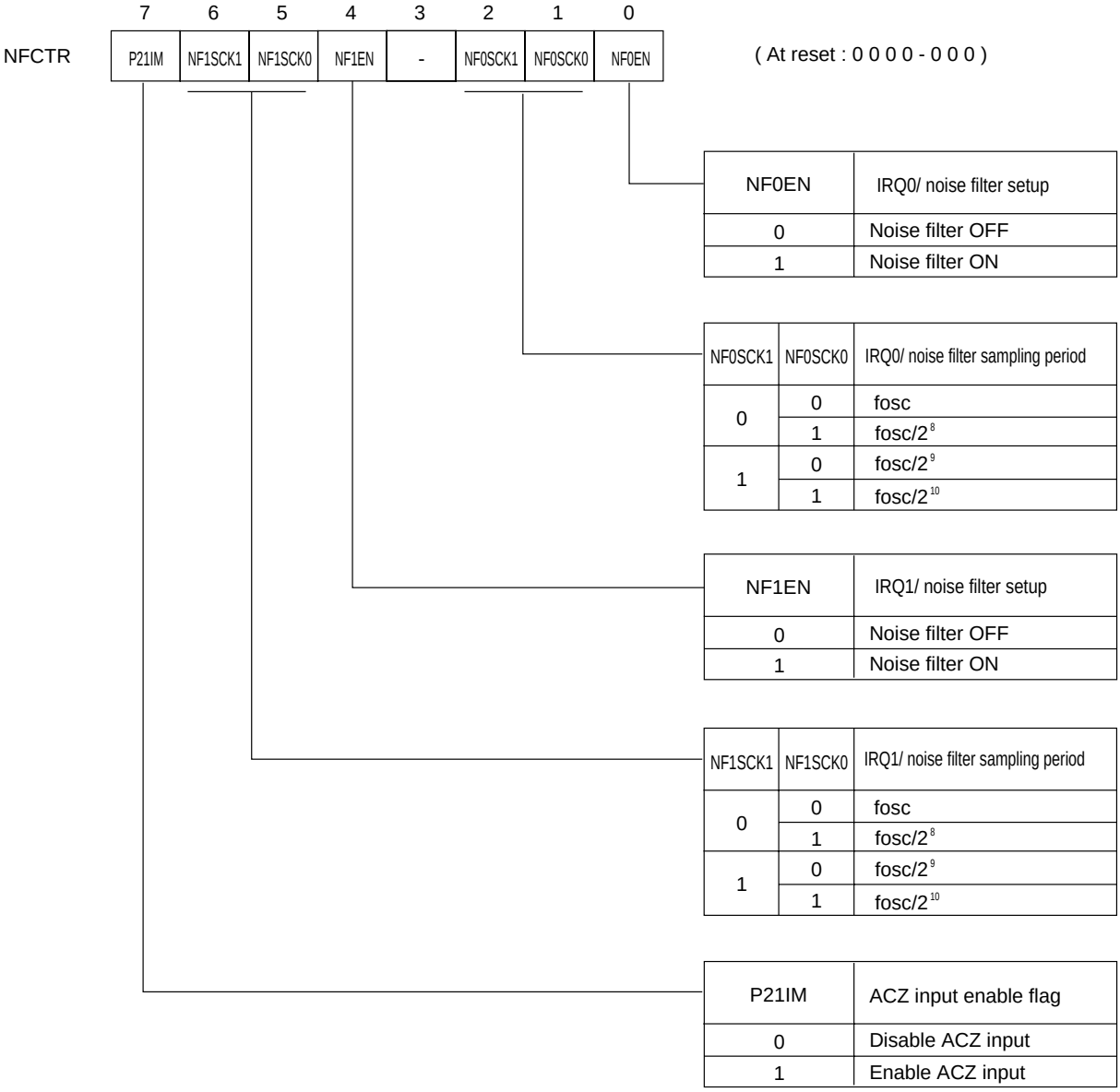


Figure 3-3-4 Noise Filter Control Register (NFCTR : x'03F8E', R/W)

■Both Edges Interrupt Control Register (EDGDT)

The both edges interrupt control register (EDGDT) selects interrupt edges of IRQ0 and IRQ1. Interrupts are generated at both edges, or at single edge. The external interrupt control register (IRQ0ICR, IRQ1ICR) specifies whether interrupts are generated. This register also selects the interrupt trigger factosr of IRQ2 and IRQ3.

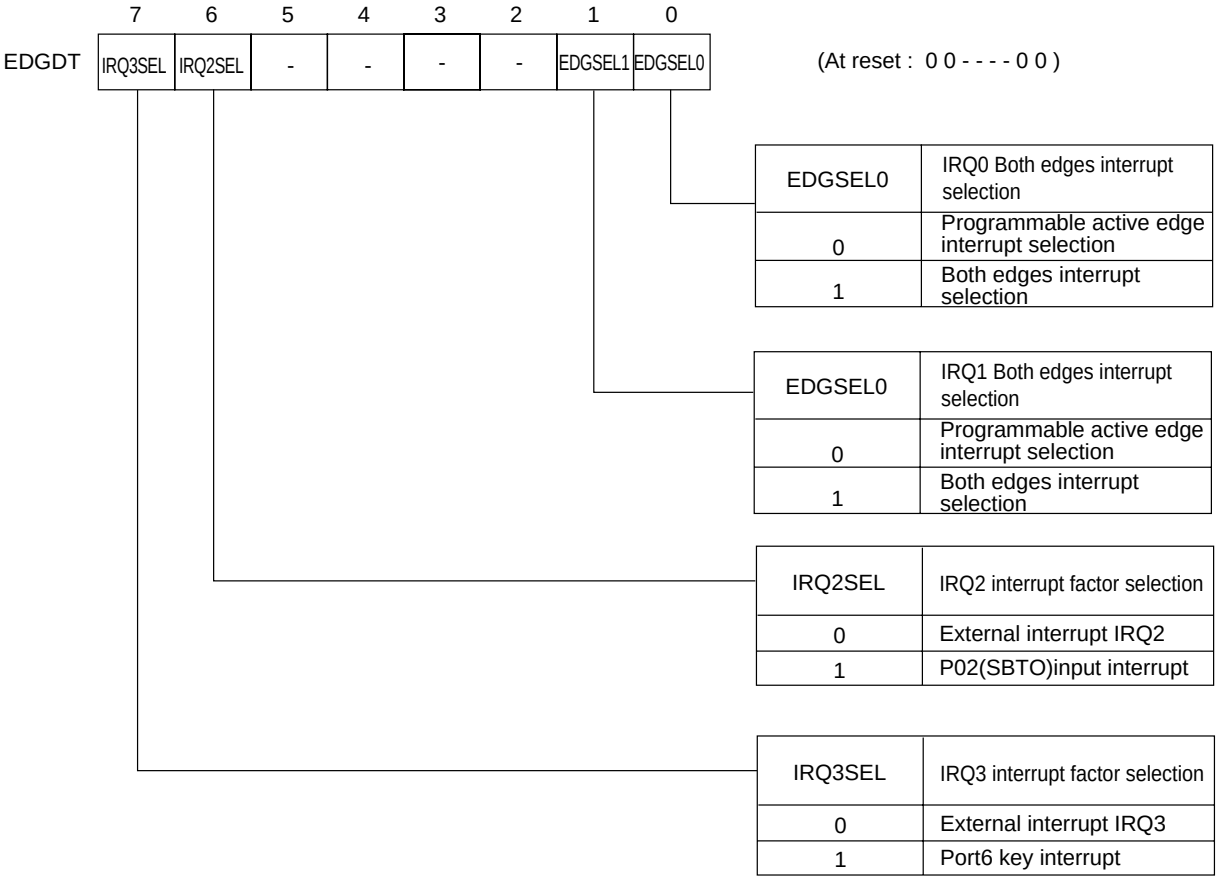


Figure 3-3-5 Both Edges Interrupt Control Register (EDGDT : x'03F8F', R/W)

■Port 6 Key Interrupt Control Register (P6IMD)

The port 6 key interrupt control register (P6IMD) selects which pin on port 6 approved key interrupt in each bit.

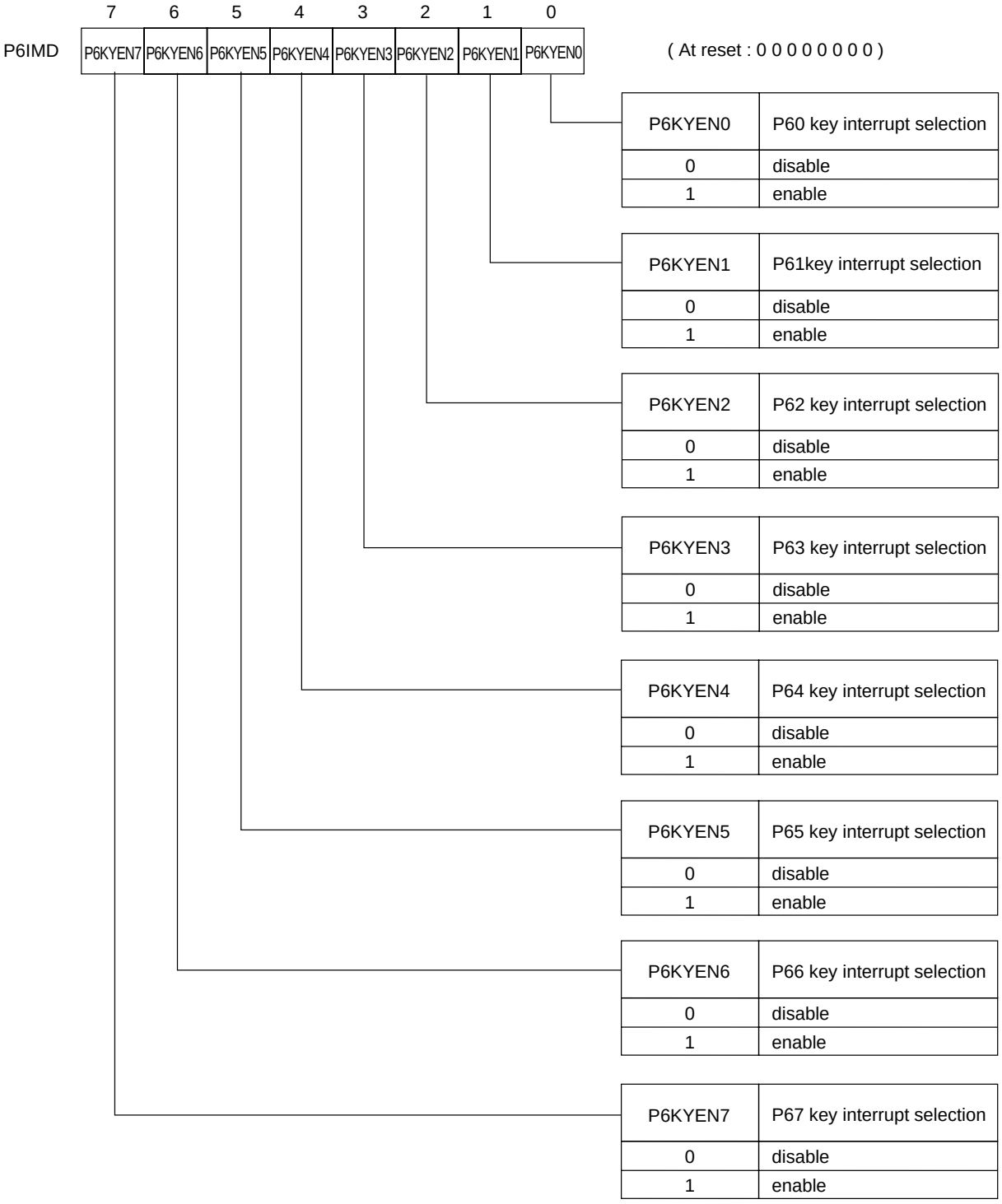


Figure 3-3-6 Port 6 Key Interrupt Control Register (P6IMD : x'03F4E', R/W)

3-3-4 Programmable Active Edge Interrupt

■Programmable Active Edge Interrupts (External interrupts 0 to 3)
Through register settings, external interrupts 0 to 3 can generate interrupt at the selected edge either rising or falling edge.

■Programmable Active Edge Interrupt Setup Example (External interrupt 0 to 3)
External interrupt 2 (IRQ2) is generated at the rising edge of the input signal from P22.
The table below provides a setup example for IRQ2.

Setup Procedure	Description
(1) Specify the interrupt active edge. IRQ2ICR (x'3FE4') bp5 : REDG2 = 1	(1) Set the REDG2 flag of the external interrupt 2 control register (IRQ2ICR) to "1" to specify the rising edge as the active edge for interrupts.
(2) Set the interrupt level. IRQ2ICR (x'3FE4') bp7-6 : IRQ2LV1-0= 10	(2) Set the interrupt priority level in the IRQ2LV1-0 flag of the IRQ2ICR register. If the interrupt request flag has been already set, clear it. [☞ Chapter 3. 3-1-4 Interrupt flag setup]
(3) Enable the interrupt. IRQ2ICR (x'3FE4') bp1 : IRQ2IE = 1	(3) Set the IRQ2IE flag of the IRQ2ICR register to "1" to enable the interrupt.

External interrupt 2 is generated at the rising edge of the input signal from P22.

The Interrupt request flag may be set to "1" at switching the interrupt edge, so clear the interrupt request flag before interrupt is accepted. Also specify the interrupt active edge before the interrupt acceptance.

The external interrupt pin is recommended to be pull-up in advance.

When the programmable active edge interrupt is specified for external interrupt 0, 1(IRQ0, IRQ1), set the EDGSELn flag of the both edge interrupt control register (EDGDT) to "0".

3-3-5 Both Edges Interrupt

■Both Edges Interrupt (External interrupts 0 and 1)

Both edges interrupt can generate interrupt at both the falling edge and the rising edge by the input signal from external input pins. CPU also can be returned from standby mode by both edges interrupt.

■Both Edges Interrupt Setup Example (External interrupts 0 and 1)

External interrupt 0 (IRQ0) is generated at the both edges of the input signal from P20 pin.
An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Select the both edges interrupt. EDGDT (x'3F8F') bp0 : EDGSEL0 = 1	(1) Set the EDGSEL0 flag of the both edges interrupt control register (EDGDT) to "1" to select the both edges interrupt.
(2) Set the interrupt level. IRQ0ICR (x'3FE2') bp7-6 : IRQ0LV1- 0 = 10	(2) Set the interrupt level by the IRQ0LV1-0 flag of the IRQ0ICR register. The interrupt request flag of the IRQ0ICR register may be set, so make sure to clear the interrupt request flag (IRQ0IR). [☞ Chapter 3 3-1-4 Interrupt flag setup]
(3) Enable the interrupt. IRQ0ICR (x'3FE2') bp1 : IRQ0IE = 1	(3) Set the IRQ0IE flag of the IRQ0ICR register to "1" to enable the interrupt.

At the both edge of the input signal from P20 pin, an external interrupt 0 is generated .

When the both edge interrupt is selected, the interrupt request generates at the both edge, regardless of the REDGn flag of the external interrupt control register (IRQnICR).

The interrupt request flag may be set to "1" at switching the interrupt edge. So, clear the interrupt request flag before the interrupt enable. Also, select the both edge interrupt before the interrupt enable.

The external interrupt pin is recommended to be pull-up, in advance.

3-3-6 Key Input Interrupt

■Key Input Interrupt (External interrupt 3)

This LSI can set port 6 pin (P60 to P67) by 2 bits to key input pin. Key input interrupt can generate an interrupt at the falling edge, if at least 1 key input pin outputs low level.

Key input pin should be pull-up in advance.

When key input interrupt is used, set the IRQ3SEL flag of the both edges interrupt control register (EDGDT) to "1".

■Key Input Interrupt Setup Example (External interrupt 3)

After P60 to P63 of port 6 are set to key input pins and key is input (low level), the external interrupt 3 (IRQ3) is generated. An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Set the key input pin to input. P6DIR (x'3F36') bp3-0 : P6DIR3-0 = 0000	(1) Set the P6DIR3-0 flag of the port 6 direction control register (P6DIR) to "0000" to set P60 to P63 pins to input pins.
(2) Set the pull-up resistance. P6PLU (x'3F46') bp3-0 : P6PLU3-0 = 1111	(2) Set the P6PLU3-0 flag of the port 6 pull-up/ down resistor control register (P6PLUs) to "1111" to add the pull-up resistance to P60 to P63 pins.
(3) Select the key input interrupt. EDGDT (x'3F8F') bp7 : IRQ3SEL = 1	(3) Set the IRQ3SEL flag of the both edges interrupt control register (EDGDT) to "1" to select the external interrupt 3 source to the port 6 key interrupt.
(4) Select the key input pin. P6IMD (x'3F4E') bp3-0 : P6KYEN3-0= 11	(4) Set the P6KYEN3-1 flag of the port 6 key interrupt control register (P6IMD) to "1111" to set P60 to P63 pins to key input pins.
(5) Set the interrupt level. IRQ3ICR (x'3FE5') bp7-6 : IRQ3LV1-0= 10	(5) Set the interrupt level by the IRQ3LV1-0 flag of the IRQ3ICR register. If the interrupt request flag has been already set, clear the it. [☞ Chapter 3 3-1-4. Interrupt flag setup]
(6) Enable the interrupt. IRQ3ICR (x'3FE5') bp1 : IRQ3IE = 1	(6) Set the IRQ3IE flag of the IRQ3ICR register to "1" to enable the interrupt.

Note : The above (3) and (4), (5) and (6) are set at the same time.

If there is at least one input signal, from the P60 to P63 pins, shows low level, the external interrupt 3 is generated at the falling edge.

The setup of the key input should be done before the interrupt is accepted.

3-3-7 Noise Filter

■Noise Filter (External interrupts 0 to1)
Noise filter reduce noise by sampling the input waveform from the external interrupt pins (IRQ0, IRQ1). Its sampling cycle can be selected from 4 types (fosc, fosc/2^8, fosc/2^9, fosc/2^{10}).

■Noise Remove Selection (External interrupts 0 to 1)
Noise remove function can be used by setting the NFnEN flag of the noise filter control register (NFCTR) to "1".

Table 3-3-3 Noise Remove Function

NFnEN	IRQ0 input (P20)	IRQ1 input (P21)
0	IRQ0 Noise filter OFF	IRQ1 Noise filter OFF
1	IRQ0 Noise filter ON	IRQ1 Noise filter ON

■Sampling Cycle Setup (External interrupts 0 and 1)
The sampling cycle of noise remove function can be set by the NFnSCK 1- 0 flag of the NFCTR register.

Table 3-3-4 Sampling Cycle / Time of Noise Remove Function

NFnCKS1	NFnCKS0	Sampling cycle	High-Speed oscillation			
			fosc=20 MHz		fosc=8 MHz	
0	0	fosc	20 MHz	50 ns	8 MHz	125 ns
	1	fosc/2^8	78.13 kHz	12.80 µs	31.25 kHz	32 µs
1	0	fosc/2^9	39.06 kHz	25.60 µs	15.62 kHz	64 µs
	1	fosc/2^{10}	19.53 kHz	51.20 µs	7.81 kHz	128 µs

To select fosc/2^8, fosc/2^9 or fosc2/^{10} as a sampling cycle, set "1" to the PSCEN flag of the prescaler control register (PSCMD) to activate prescaler function in advance.

■Noise Remove Function Operation (External interrupts 0 to 1)

After sampling the input signal to the external interrupt pins (IRQ0, IRQ1) by the set sampling time, if the same level comes continuously three times, that level is sent to the inside of LSI. If the same level does not come continuously three times, the previous level is sent. It means that only the signal with the width of more than " Sampling time X 3 sampling clock " can pass through the noise filter, and other much narrower signals are removed, because those are regarded as noise.

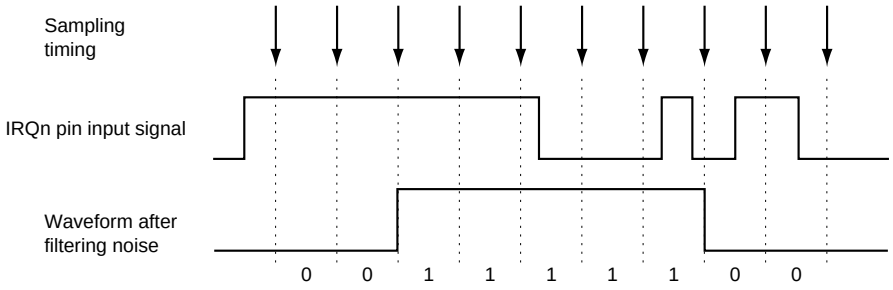


Figure 3-3-7 Noise Remove Function Operation

Noise filter can not be used at STOP mode, HALT mode and SLOW mode.

■Noise Filter Setup Example (External interrupt 0 and 1)

Noise remove function is added to the input signal from P20 pin to generate the external interrupt 0 (IRQ0) at the rising edge. The sampling clock is set to fosc, and the operation state is fosc = 20 MHz. An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Specify the interrupt active edge. IRQ0ICR (x'3FE2') bp5 : REDG0 = 1	(1) Set the REDG0 flag of the external interrupt 0 control register (IRQ0ICR) to "1" to specify the interrupt active edge to the rising edge.
(2) Select the sampling clock. NFCTR (x'3F8E') bp2-1 : NF0SCK1-0 = 00	(2) Select the sampling clock to fosc by the NF0SCK 1-0 flag of the noise filter control register (NFCTR).
(3) Set the noise filter operation. NFCTR (x'3F8E') bp0 : NF0EN = 1	(3) Set the NF0EN flag of the NFCTR register to "1" to add the noise filter operation.
(4) Set the interrupt level. IRQ0ICR (x'3FE2') bp7-6 : IRQ0LV1-0= 10	(4) Set the interrupt level by the IRQ0LV 1- 0 flag of the IRQ0ICR register. If the interrupt request flag has been already set, clear the request flag. [☞Chapter 3 3-1-4. Interrupt flag setup]
(5) Enable the interrupt. IRQ0ICR (x'3FE2') bp1 : IRQ0IE = 1	(5) Set the IRQ0IE flag of the IRQ0ICR register to "1" to enable the interrupt.

Note : The above (2) and (3) are set at the same time.

The input signal from the P20 pin generates the external interrupt 0 at the rising edge of the signal, after passing through the noise filter.

The setup of the noise filter should be done before the interrupt is enabled.

The external interrupt pins are recommended to be pull-up in advance.

3-3-8 AC Zero-Cross Detector

This LSI has AC zero-cross detector circuit. The P21 / ACZ pin is the input pin of AC zero-cross detector circuit. AC zero-cross detector circuit output the high level when the input level is at the middle, and outputs the low level at other level.

■AC Zero-Cross Detector (External interrupt 1)
AC zero-cross detector sets the IRQ1 pin to the high level when the input signal (P21/ACZ pin) is at intermediate range. At the other level, IRQ1 pin is set to the low level. AC zero-cross can be detected by setting the P21IM flag of the noise filter control register (NFCTR) to "1".

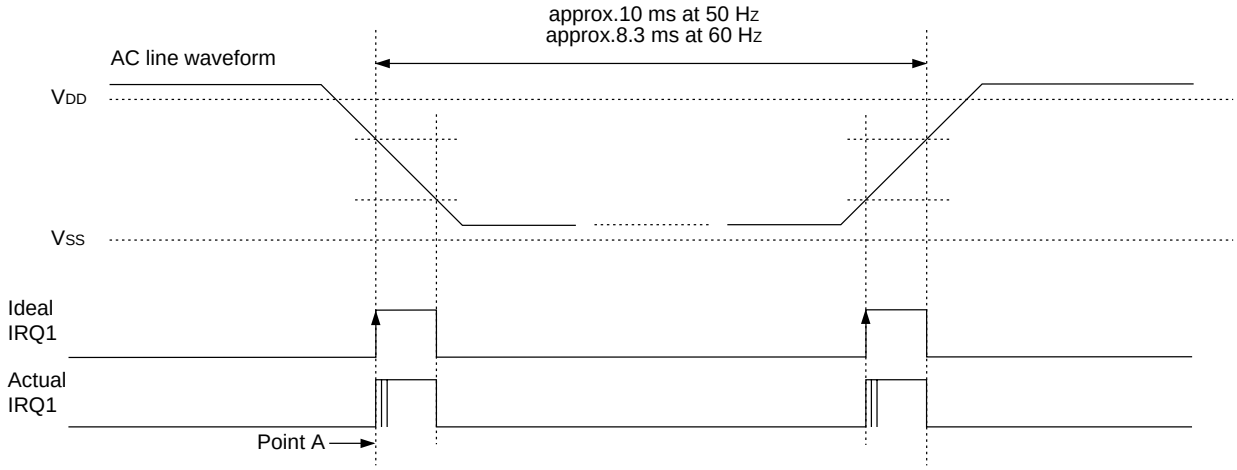


Figure 3-3-8 AC Line Waveform and IRQ1 Generation Timing

Actual IRQ1 interrupt request is generated several times at crossing the 1/2 V_{DD} of AC line waveform. So, the filtering operation by the program is needed.

The interrupt request is generated at the rising edge of the AC zero-cross detector signal.

■AC Zero-Cross Detector Setup Example (External interrupt 1)

AC zero-cross detector generates the external interrupt 1 (IRQ1) by using P21/ACZ pin.
An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Select the AC zero-cross detector signal. NFCTR (x'3F8E') bp7 : P21IM = 1	(1) Set the P21IM flag of the noise filter control register (NFCTR) to "1" to select the AC zero-cross detector signal as the external interrupt 1 generation factor.
(2) Set the interrupt level. IRQ1ICR (x'3FE3') bp7-6 : IRQ1LV1-0= 10	(2) Set the interrupt level by the IRQ1LV 1-0 flag of the IRQ1ICR register. If the interrupt request flag has been already set, clear the interrupt flag. [☞ Chapter 3 3-1-4. Interrupt flag setup]
(3) Enable the interrupt. IRQ1ICR (x'3FE3') bp1 : IRQ1IE = 1	(3) Set the IRQ1IE flag of the IRQ1ICR register to "1" to enable the interrupt.

When the input signal level from P21/ACZ pin crosses 1/2 V_{DD}, the external interrupt 1 is generated.

3-3-9 P02 (SBT0) interrupt

■P02(SBT0) interrupt (External interrupts 2)
Through register settings, signals I/O from P02/SBT0 pin can generate interrupt at the selected edge either rising or falling edge. Recovery from standby mode is available by generation of serial interface 0 clock.

■P02(SBT0) interrupt Setup Example (External interrupts 2)
External interrupt 2 (IRQ2) is generated at the rising edge of the input signal from P02(SBT0).
The table below provides a setup example for IRQ2.

Setup Procedure	Description
(1) Select the P02(SBT0) interrupt EDGDT (x'3F8F') bp6 : IRQ2SEL = 1	(1) Set the IRQ2SEL flag of the both edges interrupt register (EDGDT) to "1" to select the serial interface 0SBT0 interrupt.
(2) Select the interrupt valid edge. IRQ2ICR (x'3FE4') bp5 : REDG2 = 1	(2) Select the interrupt valid edge by the REDG2 flag of the external interrupt 2 control register (IRQ2ICR)
(3) Set the interrupt level. IRQ2ICR (x'3FE4') bp7-6 : IRQ2LV-0 = 10	(3) Set the interrupt level by IRQ2LV1-0 flag of the IRQ2ICR register. If the interrupt request flag has been already set, clear it. [☞ Chapter 3 3-1-4. Interrupt flag setup]
(4) Enable the interrupt. IRQ2ICR (x'3FE4') bp1 : IRQ2IE = 1	(4) Set the IRQ2IE flag of the IRQ2ICR register to "1" to enable the interrupt.

External interrupt 2 is generated at the rising edge of the signal input from the P02(SBT0) pin.

The Interrupt request flag may be set to "1" at switching the interrupt edge, so clear the interrupt request flag before interrupt is accepted. Also specify the interrupt active edge before the interrupt acceptance.

Chapter 4 I/O Ports

4

4-1 Overview

4-1-1 I/O Port Diagram

A total of 41 pins on this LSI, including those shared with special function pins, are allocated for the 8 I/O ports of ports 0 to 2, ports 6 to 8, port A and port C. Each I/O port is assigned to its corresponding special function register area in memory. I/O ports are operated in byte or bit units in the same way as RAM.

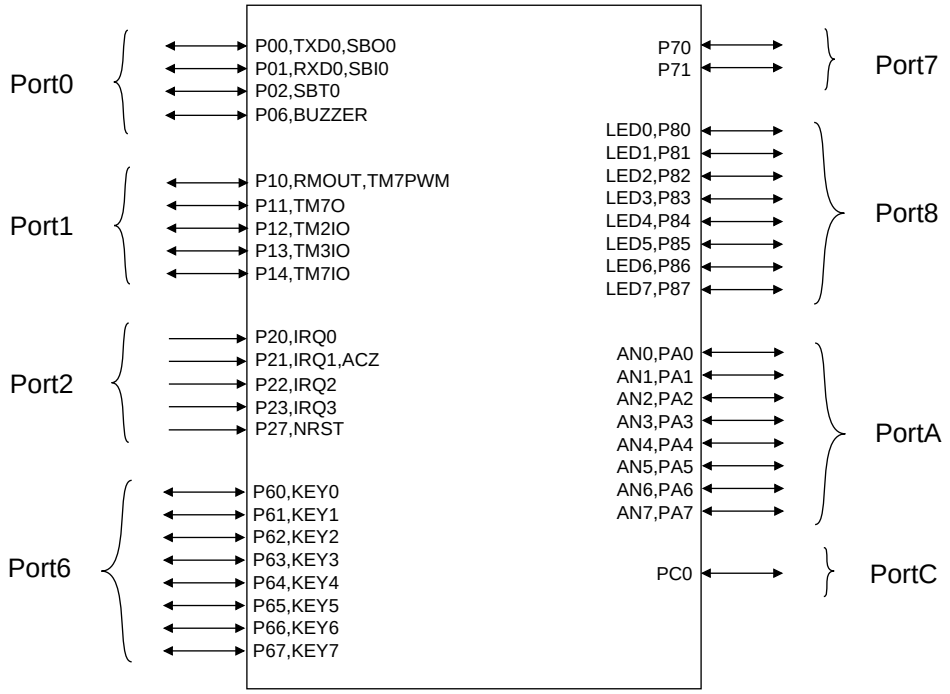


Figure 4-1-1 I/O Port Functions

4-1-2 I/O Port Status at Reset

Table 4-1-1 I/O Port Status at Reset (Single chip mode)

Port Name	I/O mode	Pull-up / Pull-down resistor	I/O port, special functions
Port 0	Input mode	No pull-up resistor	I/O port
Port 1	Input mode*	No pull-up resistor	I/O port
Port 2	Input mode	P27 : Pull-up resistor Others : No pull-up resistor	I/O port
Port 6	Input mode	No pull-up resistor	I/O port
Port 7	Input mode	No pull-up / pull-down resistor	I/O port
Port 8	Input mode	No pull-up resistor	I/O port
Port A	Input mode	No pull-up / pull-down resistor	I/O port
Port C	Input mode	No pull-up resistor	I/O port

* P10 is output mode

4-1-3 Control Registers

Ports 0 to 2, ports 6 to 8, port A and port C are controlled by the data output register (PnOUT), the data input register (PnIN), the I/O direction control register (PnDIR), the pull-up resistor control register (PnPLU) and the pull-up / pull-down resistor control resister (PnPLUD) and registers (P1OMD, PAIMD, SC0ODC, PA0DC, FLOAT) that control special function pin.

Table 4-1-3 shows the registers to control ports 0 to 2, ports 6 to 8, port A and port C ;

Table 4-1-3 I/O Port Control Registers List (1/2)

	Register	Address	R/W	Function	Page
Port 0	P0OUT	x'03F10'	R/W	Port 0 output register	Ⅳ-7
	P0IN	x'03F20'	R	Port 0 input register	Ⅳ-7
	P0DIR	x'03F30'	R/W	Port 0 direction control register	Ⅳ-7
	P0PLU	x'03F40'	R/W	Port 0 pull-up resistor control register	Ⅳ-7
	SC0ODC	x'03F96'	R/W	Serial interface 0 port control register	Ⅳ-8
Port 1	P1OUT	x'03F11'	R/W	Port 1 output register	Ⅳ-12
	P1IN	x'03F21'	R	Port 1 input register	Ⅳ-12
	P1DIR	x'03F31'	R/W	Port 1 direction control register	Ⅳ-12
	P1PLU	x'03F41'	R/W	Port 1 pull-up resistor control register	Ⅳ-12
	P1OMD	x'03F2F'	R/W	Port 1 output mode register	Ⅳ-13
Port 2	P2OUT	x'03F12'	R/W	Port 2 output register	Ⅳ-17
	P2IN	x'03F22'	R	Port 2 input register	Ⅳ-17
	P2PLU	x'03F42'	R/W	Port 2 pull-up resistor control register	Ⅳ-17
Port 6	P6OUT	x'03F16'	R/W	Port 6 output register	Ⅳ-20
	P6IN	x'03F26'	R	Port 6 input register	Ⅳ-20
	P6DIR	x'03F36'	R/W	Port 6 direction control register	Ⅳ-20
	P6PLU	x'03F46'	R/W	Port 6 pull-up resistor control register	Ⅳ-20
Port 7	P7OUT	x'03F17'	R/W	Port 7 output register	Ⅳ-23
	P7IN	x'03F27'	R	Port 7 input register	Ⅳ-23
	P7DIR	x'03F37'	R/W	Port 7 direction control register	Ⅳ-23
	P7PLUD	x'03F47'	R/W	Port 7 pull-up / pull-down resistor control register	Ⅳ-23
Port 8	P8OUT	x'03F18'	R/W	Port 8 output register	Ⅳ-27
	P8IN	x'03F28'	R	Port 8 input register	Ⅳ-27
	P8DIR	x'03F38'	R/W	Port 8 direction control register	Ⅳ-27
	P8PLU	x'03F48'	R/W	Port 8 pull-up resistor control register	Ⅳ-27

Table 4-1-4 I/O Port Control Registers List (2/2)

	Register	Address	R/W	Function	Page
Port A	PAOUT	x'03F1A'	R/W	Port A output register	IV-30
	PAIN	x'03F2A'	R	Port A input register	IV-30
	PADIR	x'03F3F'	R/W	Port A direction control register	IV-30
	PAIMD	x'03F3A'	R/W	Port A input mode register	IV-30
	PAODC	x'03F4F'	R/W	Port A output mode control register	IV-30
	PAPLUD	x'03F4A'	R/W	Port A pull-up / pull-down resistor control register	IV-31
Port C	PCOUT	x'03F1C'	R/W	Port C output register	IV-34
	PCIN	x'03F2C'	R	Port C input register	IV-34
	PCDIR	x'03F3C'	R/W	Port C direction control register	IV-34
	PCPLU	x'03F4C'	R/W	Port C pull-up resistor control register	IV-34
Pin control	FLOAT	x'03F2E'	R/W	Pull-up / Pull-down resistor selection, pin control register	IV-24, IV-31

R/W : Both Readable/Writable

R : Read only

4-2 Port 0

4-2-1 Description

■General Port Setup
Each bit of the port 0 control I/O direction register (P0DIR) can be set individually to set each pin as input or output. The control flag of the port 0 direction control register (P0DIR) should be set to "1" for output mode, and "0" for input mode.

To read input data of pin, set the control flag of the port 0 direction control register (P0DIR) to "0" and read the value of the port 0 input register (P0IN).

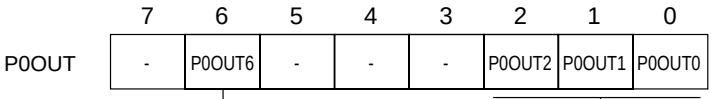
To output data to pin, set the control flag of the port 0 direction control register (P0DIR) to "1" and write the value of the port 0 output register (P0OUT).

Each pin can be set individually if pull-up resistor is added or not, by the port 0 pull-up resistor control register (P0PLU). Set the control flag of the port 0 pull-up resistor control register (P0PLU) to "1" to add pull-up resistor.

■Special Function Pin Setup
P00 to P02 are used as I/O pin for serial interface 0, as well. P00 is output pin of the serial interface 0 transmission data, and UART 0 transmission data. When the SC0SBOS flag of the serial interface 0 mode register 1 (SC0MD1) is "1", P00 is serial data output pin. P01 is the input pin of the serial interface 0 reception data, and UART 0 transmission data. P02 is I/O pin of the serial interface 0 clock. When the SC0SBTS flag of serial interface 0 mode register 1 (SC0MD1) is "1", P02 is serial clock output pin.
P00 and P02 can be selected as either an push-pull output or Nch open-drain output by the serial interface 0 port control register (SC0ODC).
[☞ Chapter 11 11-2. Control registers]

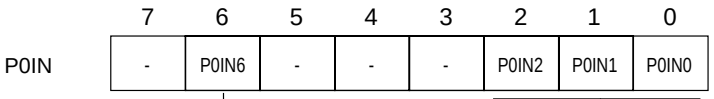
P06 is used as a buzzer output pin, as well. When the bp7 of the oscillation stabilization control register (DLYCTR) is "1", buzzer output is enabled.

4-2-2 Registers



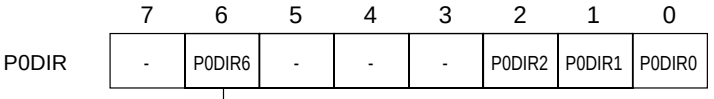
P0OUT	Output data
0	L(VSS level)
1	H(VDD level)

Port 0 output register (P0OUT : x'03F10', R/W)



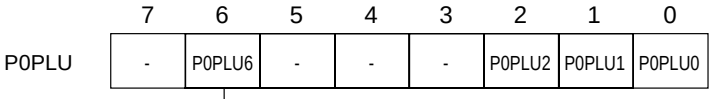
P0IN	Input data
0	Pin is low(VSS level).
1	Pin is high(VDD level).

Port 0 input register (P0IN : x'03F20', R)



P0DIR	I/O mode selection
0	Input mode
1	Output mode

Port 0 direction control register (P0DIR : x'03F30', R/W)



P0PLU	Pull-up resistor selection
0	No pull-up resistor
1	Pull-up resistor

Port 0 pull-up resistor control register (P0PLU : x'03F40', R/W)

Figure 4-2-1 Port 0 Registers (1/2)

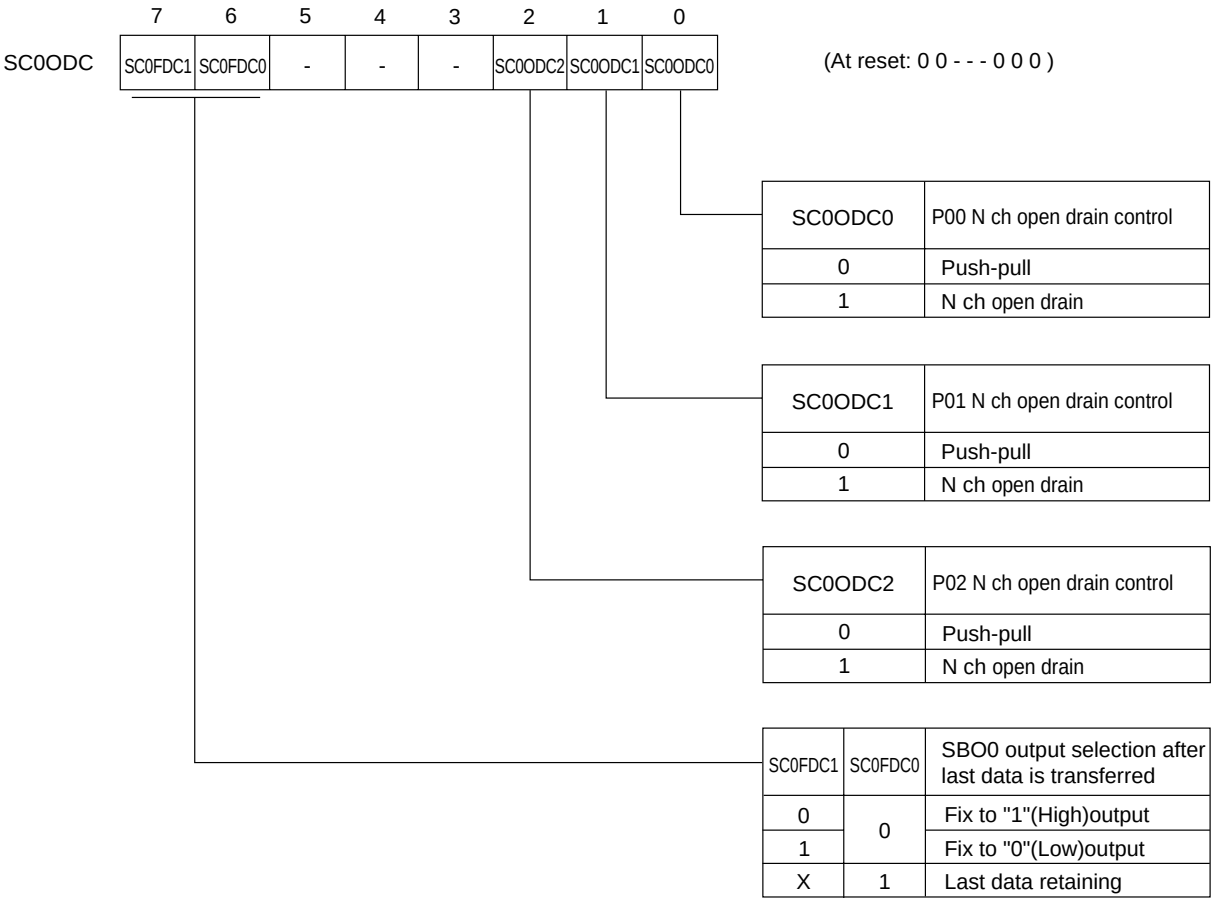


Figure 4-2-2 Port 0 Registers (2/2)

4-2-3 Block Diagram

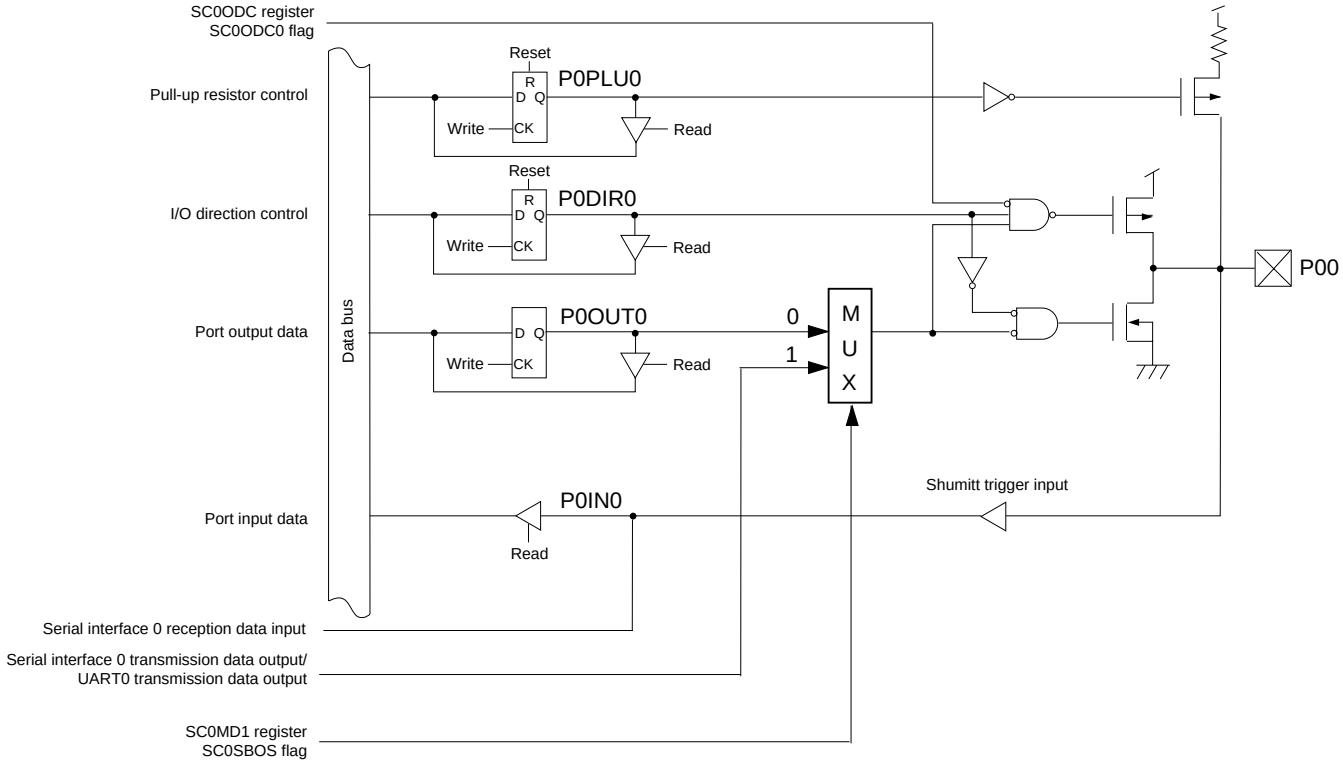


Figure 4-2-3 Block diagram (P00)

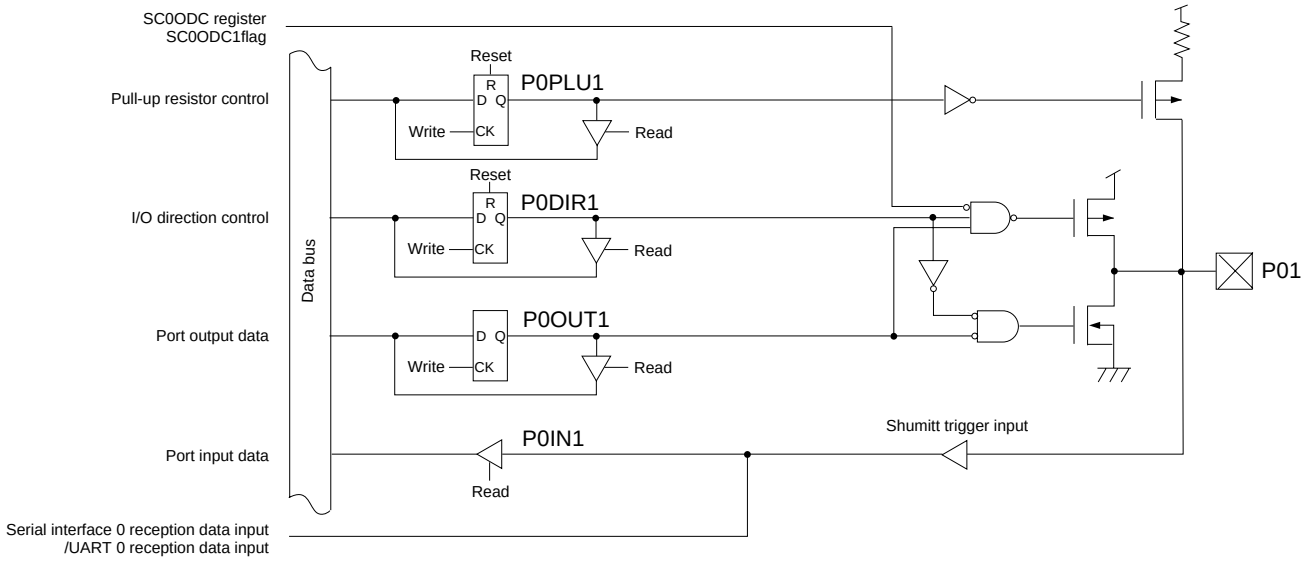


Figure 4-2-4 Block diagram (P01)

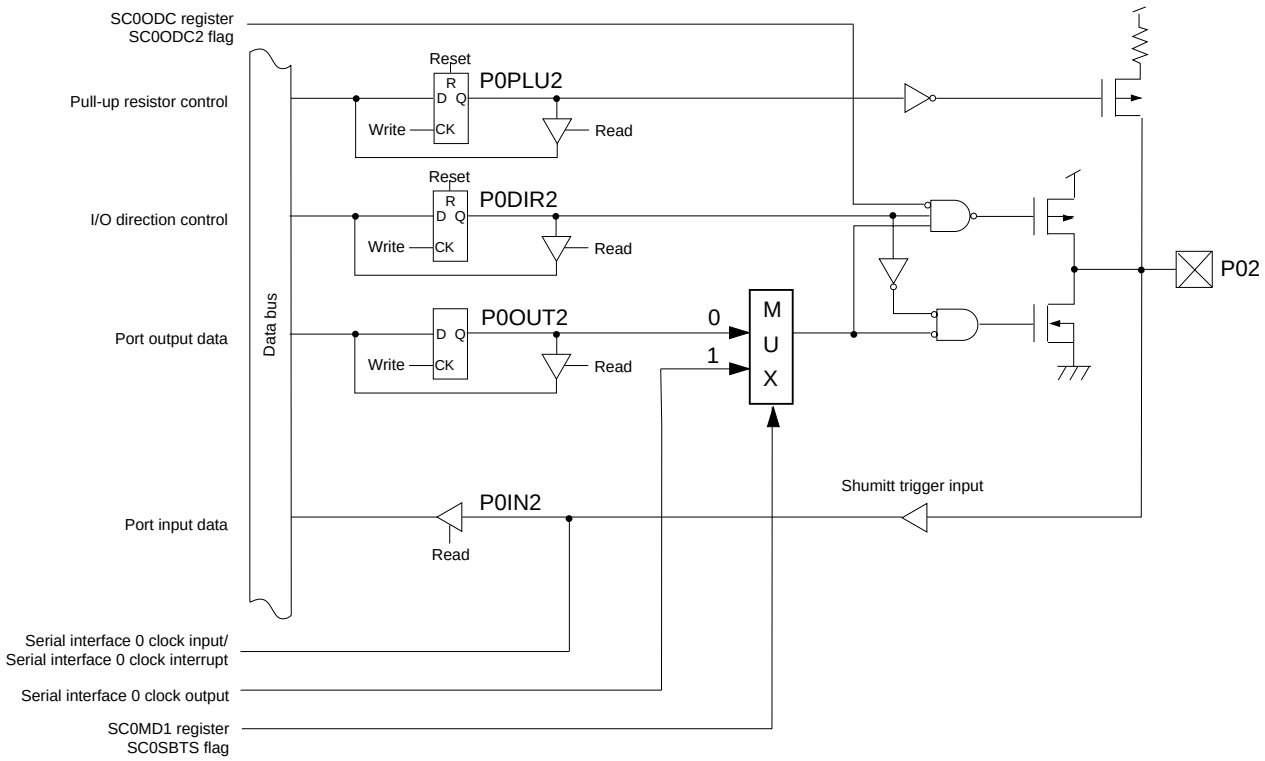


Figure 4-2-5 Block diagram (P02)

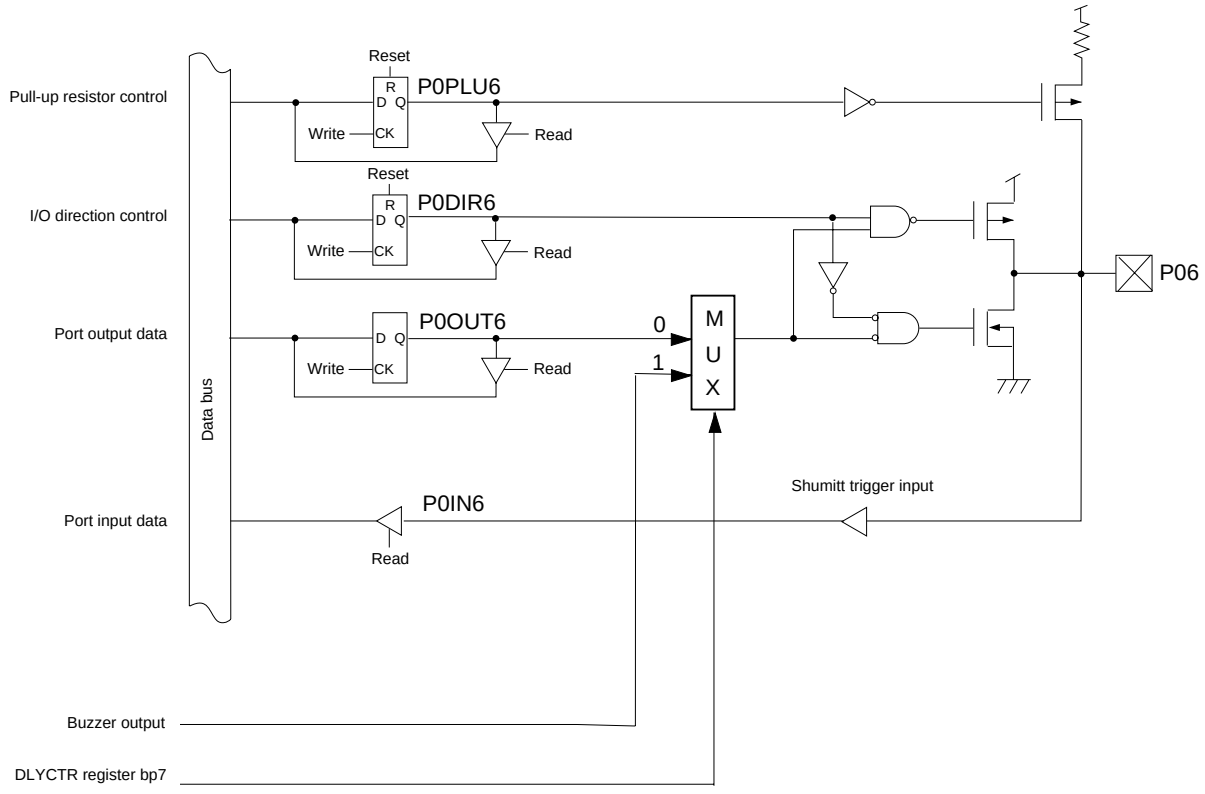


Figure 4-2-6 Block diagram (P06)

4-3 Port 1

4-3-1 Description

■General Port Setup

Each bit of the port 1 control I/O direction register (P1DIR) can be set individually to set pins as input or output. The control flag of the port 1 direction control register (P1DIR) should be set to "1" for output mode, and "0" for input mode.

To read input data of pin, set the control flag of the port 1 direction control register (P1DIR) to "0" and read the value of the port 1 input register (P1IN).

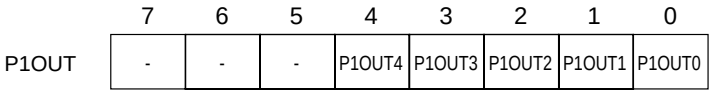
To output data to pin, set the control flag of the port 1 direction control register (P1DIR) to "1" and write the value of the port 1 output register (P1OUT).

Each pin can be set individually if pull-up resistor is added or not, by the port 1 pull-up resistor control register (P1PLU). Set the control flag of the port 1 pull-up resistor control register (P1PLU) to "1" to add pull-up resistor.

■Special Function Pin Setup

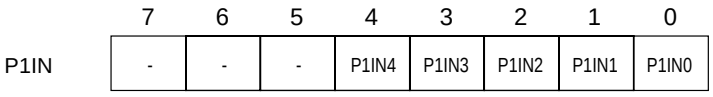
P12 to P14 are used as timer I/O pin, as well. P10 is used as remote control carrier output pin or PWM output pin of timer 7, as well. P11 is used as output pin of timer 7. The port 1 output mode register (P1OMD) can select P10 to P14, P16 output mode by each bit. When the port 1 output mode register (P1OMD) is "1", special function data is output, and when it is "0", they are used as general port.

4-3-2 Registers



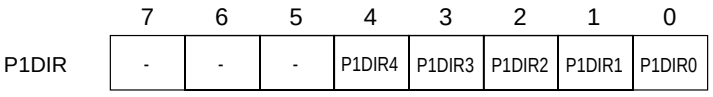
P1OUT	Output data
0	Low (VSS level)
1	High (VDD level)

Port 1 output register (P1OUT : x'03F11', R/W)



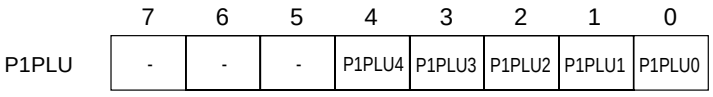
P1IN	Input data
0	Pin is low(VSS level).
1	Pin is high(VDD level).

Port 1 input register (P1IN : x'03F21', R)



P1DIR	I/O mode selection
0	Input mode
1	Output mode

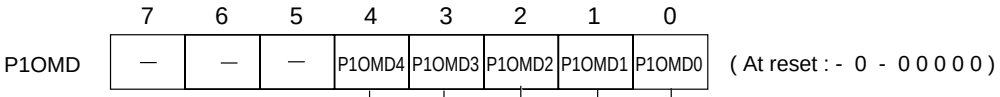
Port 1 direction control register (P1DIR : x'03F31', R/W)



P1PLU	Pull-up resistor selection
0	No pull-up resistor
1	Pull-up resistor

Port 1 pull-up resistor control register (P1PLU : x'03F41', R/W)

Figure 4-3-1 Port 1 Registers (1/2)



P1OMD0	I/O port, timer 0 output selection
0	I/O port
1	Timer 0 output

P1OMD1	I/O port, timer 1 output selection
0	I/O port
1	Timer 1 output

P1OMD2	I/O port, timer 2 output selection
0	I/O port
1	Timer 2 output

P1OMD3	I/O port, timer 3 output selection
0	I/O port
1	Timer 3 output

P1OMD4	I/O port, timer 7 output selection
0	I/O port
1	Timer 7 output

Port 1 output mode register (P1OMD : x'03F2F', R/W)

Figure 4-3-2 Port 1 Registers (2/2)

4-3-3 Block Diagram

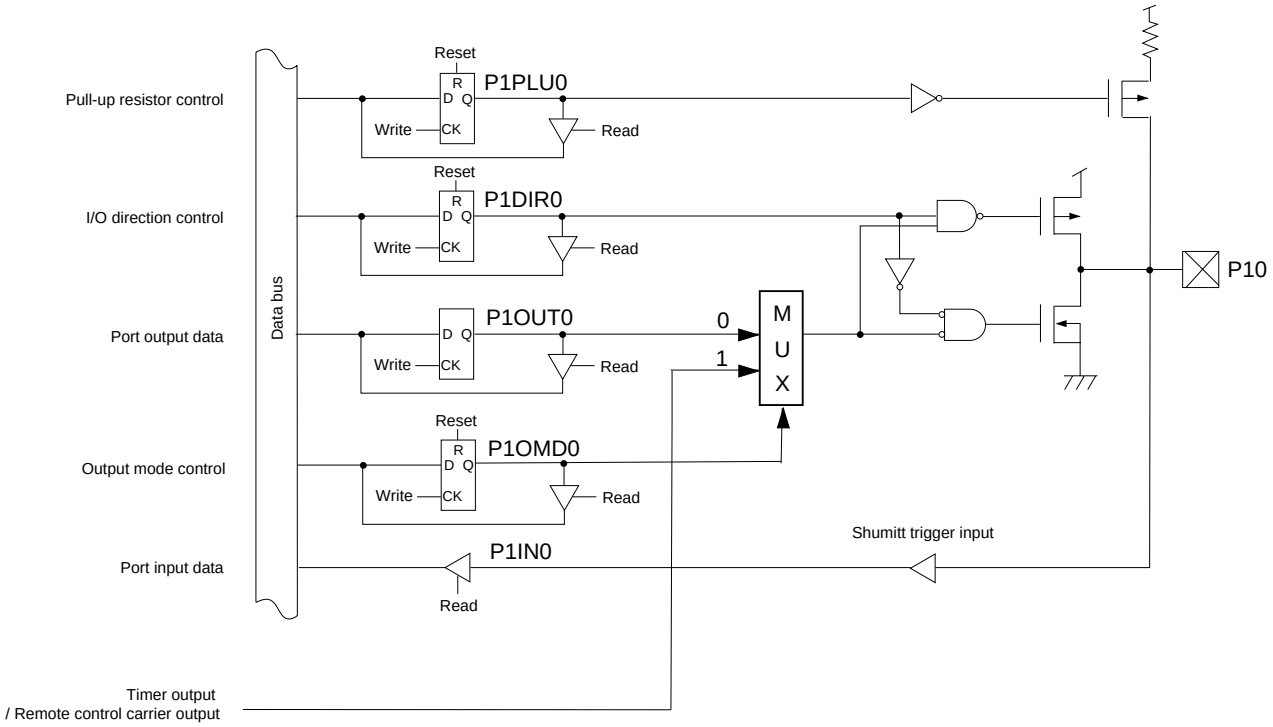


Figure 4-3-3 Block Diagram (P10)

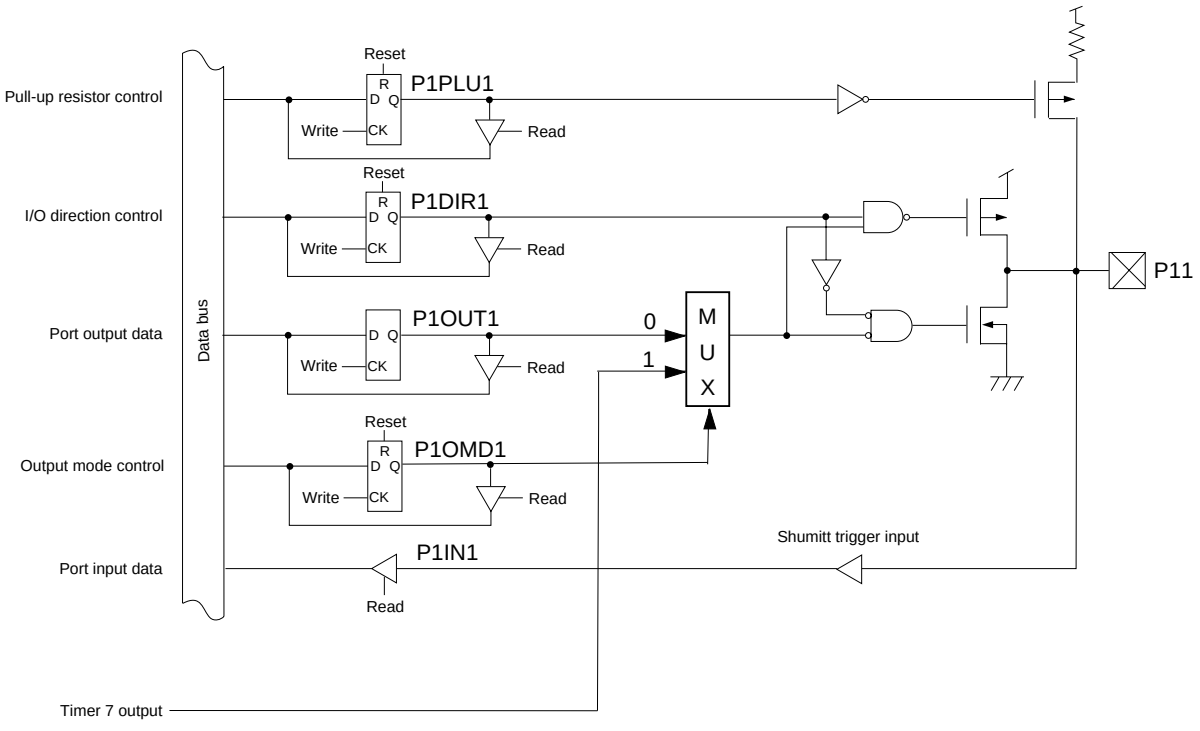


Figure 4-3-5 Block Diagram (P11)

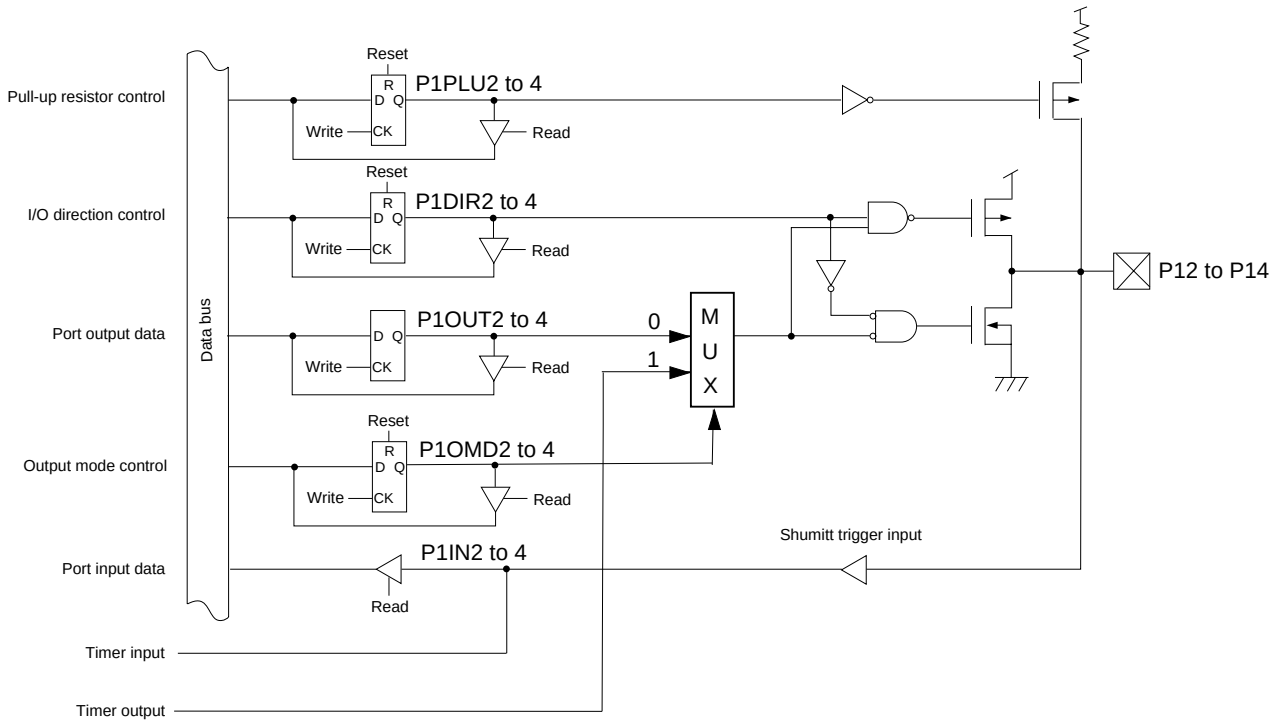


Figure 4-3-5 Block Diagram (P12 to P14)

4-4 Port 2

4-4-1 Description

■General Port Setup
Port 2 is input port, except P27. To read input data of pin, read out the value of the port 2 input register (P2IN).

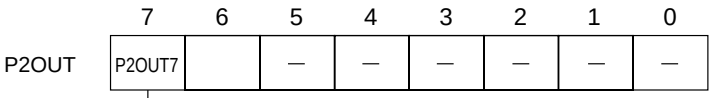
P27 is reset pin. When the software is reset, write the bp7 of the port 2 output register (P2OUT) to "0".

The port 2 pull-up resistor control register (P2PLU) can select if port 2 is added pull-up resistor or not, by each bit. When the control flag of the port 2 pull-up resistor control register (P2PLU) is set to "1", pull-up resistor is added. P27 is always added pull-up resistor.

■Special Function Pin Setup
P20, P22 to P23 are used as external interrupt pins, as well.

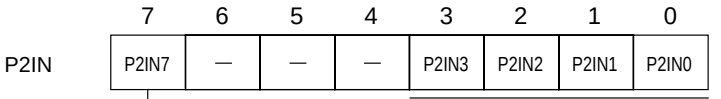
P21 is used as an input pin for external interrupt and AC zero-cross. To read data of AC zero-cross, set the bp7 of the noise filter control register (NFCTR) to "1" and read the value of the port 2 input register (P2IN).

4-4-2 Registers



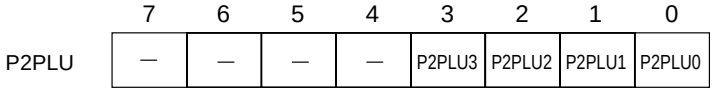
P2OUT	Output data
0	Low (V_{SS} level)
1	High (V_{DD} level)

Port 2 output register(P2OUT : x'03F12', R/W)



P2IN	Input data
0	Pin is Low (V_{SS} level).
1	Pin is High (V_{DD} level).

Port 2 input register(P2IN : x'03F22', R)



P2PLU	Pull-up resistor selection
0	No pull-up resistor
1	Pull-up resistor

Port 2 pull-up resistor control register(P2PLU : x'03F42', R/W)

Figure 4-4-1 Port 2 Registers

4-4-3 Block Diagram

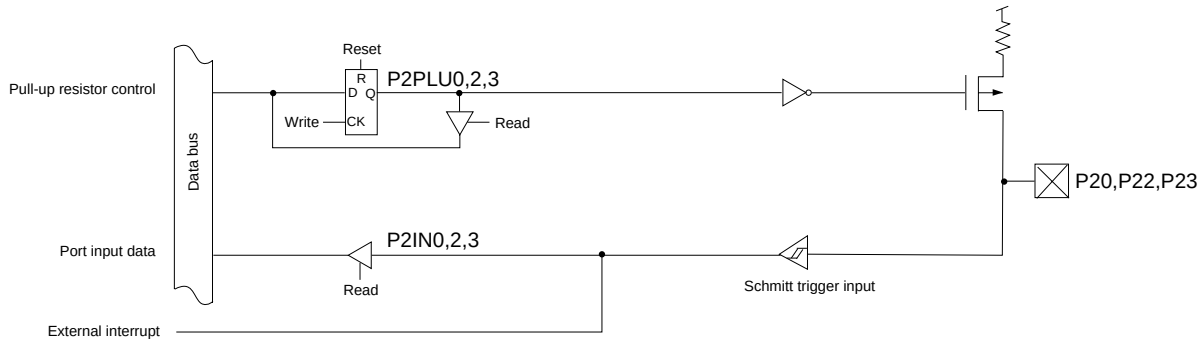


Figure 4-4-2 Block Diagram (P20, P22, P23)

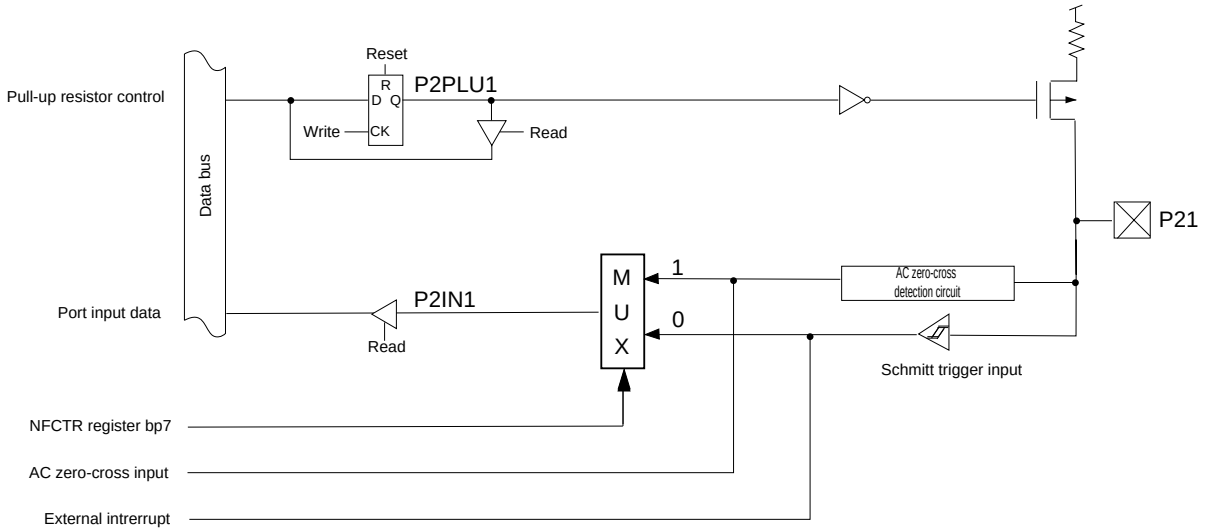


Figure 4-4-3 Block Diagram (P21)

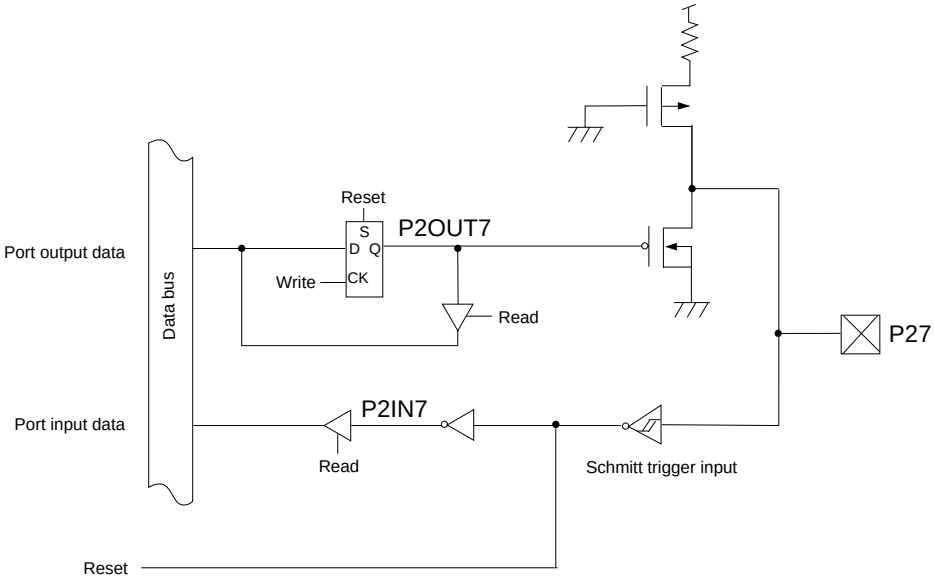


Figure 4-4-4 Block Diagram (P27)

4-5 Port 6

4-5-1 Description

■General port Setup
Each bit of the port 6 control I/O direction register (P6DIR) can be set individually to set pins as input or output. The control flag of the port 6 direction control register (P6DIR) is set to "1" for output mode, and "0" for input mode.

To read input data of pin, set the control flag of the port 6 direction control register (P6DIR) to "0" and read the value of the port 6 input register (P6IN).

To output data to pin, set the control flag of the port 6 direction control register (P6DIR) to "1" and write the value of the port 6 output register (P6OUT).

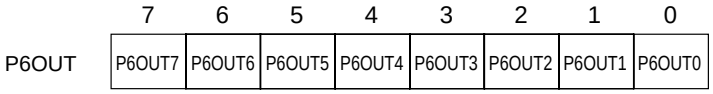
Each pin can be set individually if pull-up resistor is added or not, by the port 6 pull-up resistor control register (P6PLU). Set the control flag of the port 6 pull-up resistor control register (P6PLU) to "1" to add pull-up resistor.

■Special Function Pin Setup
P60 to P67 are used as input pins of key interrupt, as well.

Key input pins should be set to pull-up in advance.

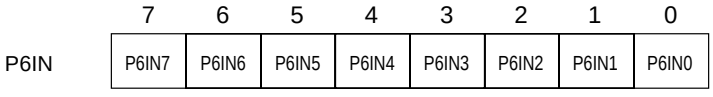
Write "0" (falling edge) to the REDG flag of the external interrupt 3 control register (IRQ3ICR) when key interrupt is used.

4-5-2 Registers



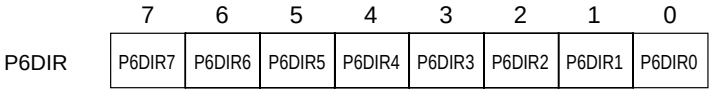
P6OUT	Output data
0	Low (V_{SS} level)
1	High (V_{DD} level)

Port 6 output register (P6OUT : x'03F16', R/W)



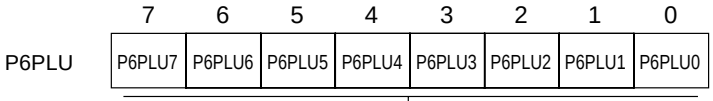
P6IN	Input data
0	Pin is Low (V_{SS} level).
1	Pin is High (V_{DD} level).

Port 6 intput register (P6IN : x'03F26', R)



P6DIR	I/O mode selection
0	Input mode
1	Output mode

Port 6 direction control register (P6DIR : x'03F36', R/W)



P6PLU	Pull-up resistor selection
0	No pull-up resistor
1	Pull-up resistor

Port 6 pull-up resistor control register (P6PLU : x'03F46', R/W)

Figure 4-5-1 Port 6 Registers

4-5-3 Block Diagram

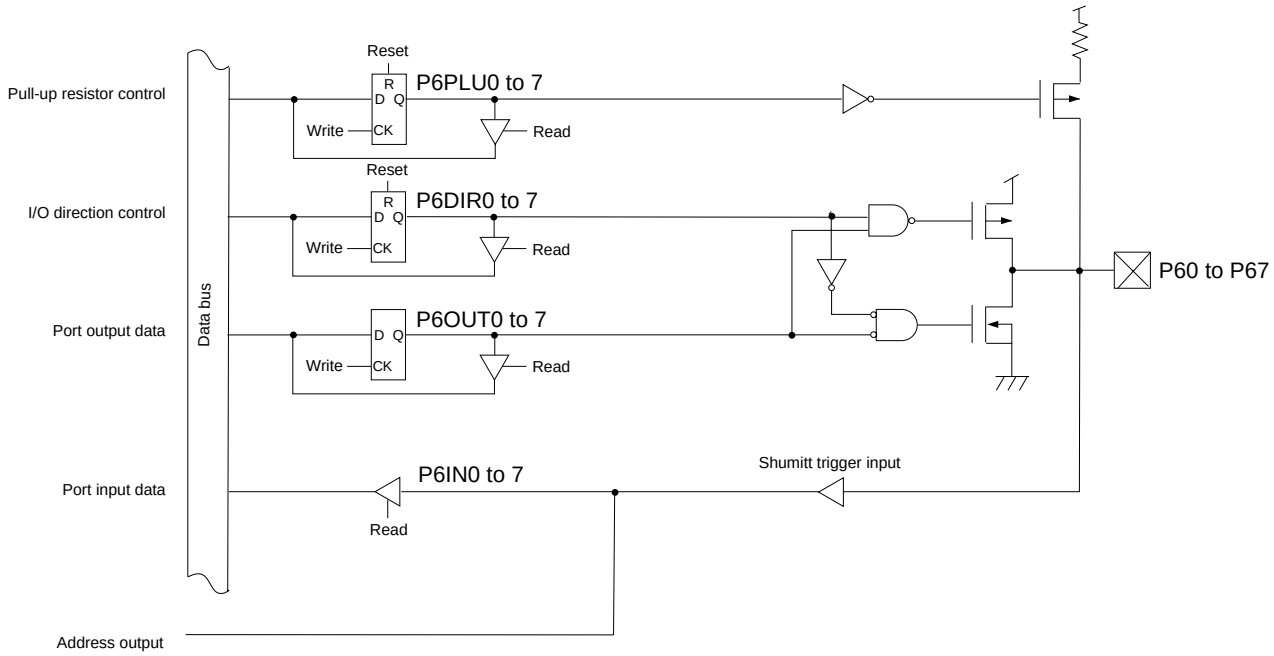


Figure 4-5-2 Block Diagram (P60 to P67)

4-6 Port 7

4-6-1 Description

■General Port Setup

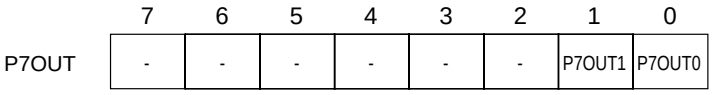
Each bit of the port 7 control I/O direction register (P7DIR) can be set individually to set pins as input or output. The control flag of the port 5 direction control register (P7DIR) is set to "1" for output mode, and "0" for input mode.

To read input data of pin, set the control flag of the port 7 direction control register (P7DIR) to "0" and read the value of the port 7 input register (P7IN).

To output data to pin, set the control flag of the port 7 direction control register (P7DIR) to "1" and write the value of the port 7 output register (P7OUT).

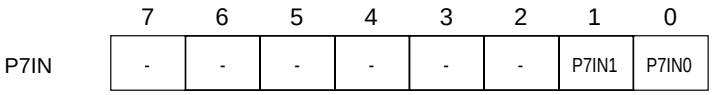
Each pin can be set individually if pull-up / pull-down resistor is added or not, by the port 7 pull-up / pull-down resistor control register (P7PLUD). Set the control flag of the port 7 pull-up / pull-down resistor control register (P7PLUD) to "1" to add pull-up or pull-down resistor. The pull-up / pull-down resistor selection register (FLOAT) select if pull-up resistor or pull-down resistor is added. The bp4 of the pull-up / pull-down resistor control register (FLOAT) is set to "1" for pull-down resistor, set to "0" for pull-up resistor.

4-6-2 Registers



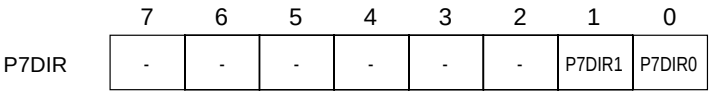
P7OUT	Output data
0	Low (VSS level)
1	High (VDD level)

Port 7 output register (P7OUT : x'03F17', R/W)



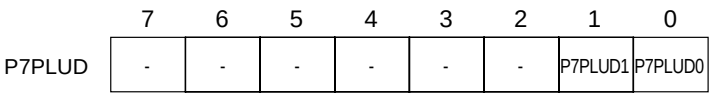
P7IN	Input data
0	Pin is Low (VSS level).
1	Pin is High (VDD level).

Port 7 input register (P7IN : x'03F27', R)



P7DIR	I/O mode selection
0	Input mode
1	Output mode

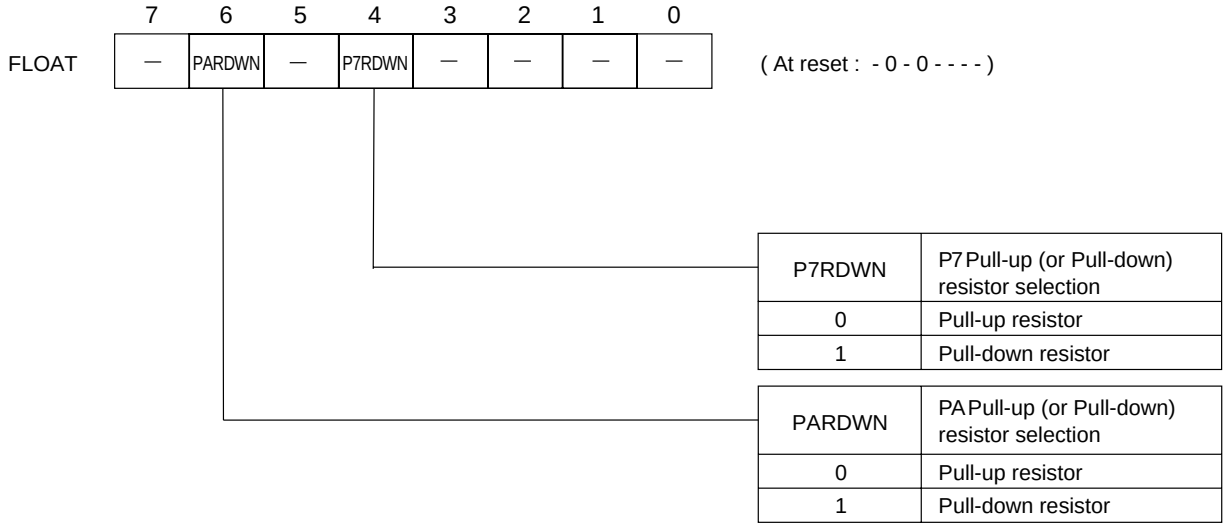
Port 7 direction control register (P7DIR : x'03F37', R/W)



P7PLUD	Pull-up (or Pull-down) resistor selection
0	No pull-up (or pull-down) resistor
1	Pull-up (or Pull-down) resistor

Port 7 pull-up / pull-down resistor control register (P7PLUD : x'03F47', R/W)

Figure 4-6-1 Port 7 Registers (1/2)



Pull-up/pull-down resistor selection, pin control resister (FLOAT : X'03F2E', R/W)

Figure 4-6-2 Port 7 Registers (2/2)

4-6-3 Block Diagram

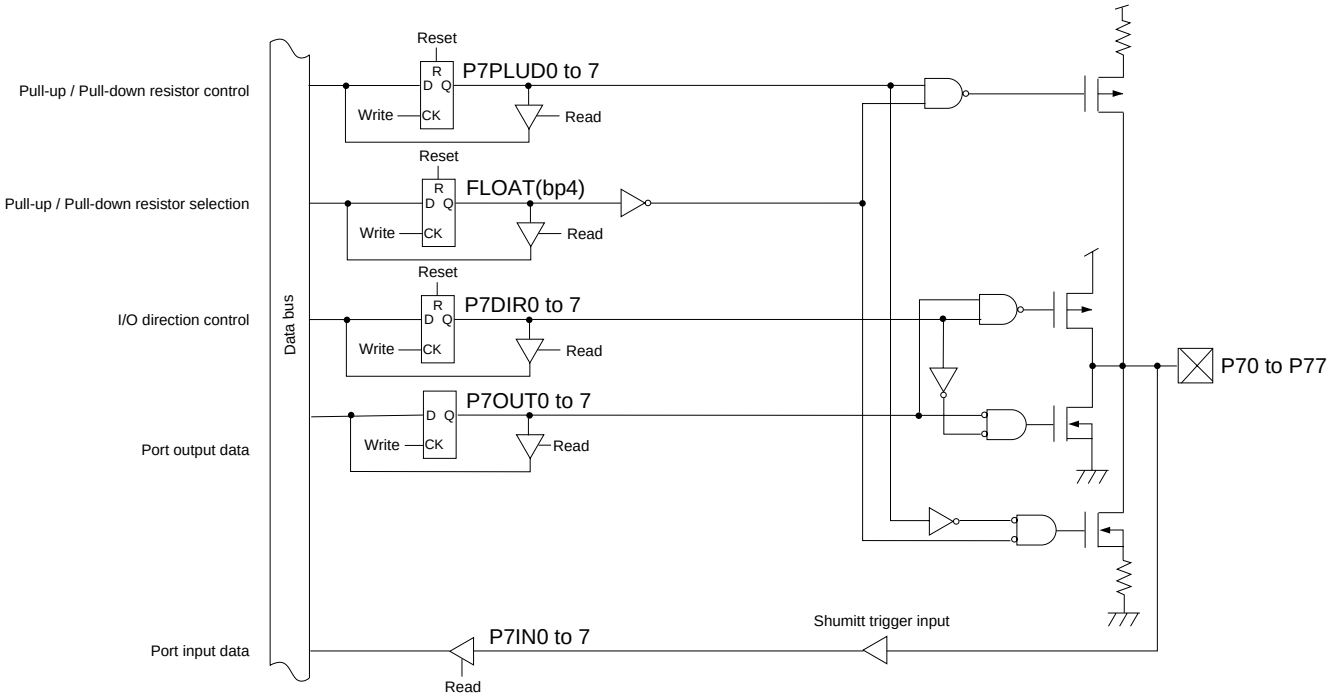


Figure 4-6-3 Block Diagram (P70 to P77)

4-7 Port 8

4-7-1 Description

■General Port Setup
Each bit of the port 8 control I/O direction register (P8DIR) can be set individually to set each pin as input or output. The control flag of the port 8 direction control register (P8DIR) is set to "1" for output mode, and "0" for input mode.

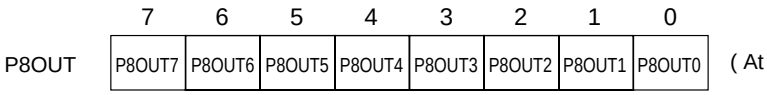
To read input data of pin, set the control flag of the port 8 direction control register (P8DIR) to "0" and read the value of the port 8 input register (P8IN).

To output data to pin, set the control flag of the port 8 direction control register (P8DIR) to "1" and write the value of the port 8 output register (P8OUT).

Each pin can be set individually if pull-up resistor is added or not, by the port 8 pull-up resistor control register (P8PLU). Set the control flag of the port 8 pull-up resistor control register (P8PLU) to "1" to add pull-up resistor.

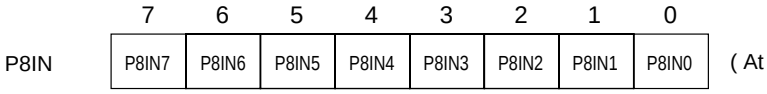
■Special Function Pin Setup
P80 to P87 are used as LED driving pins, as well.

4-7-2 Registers



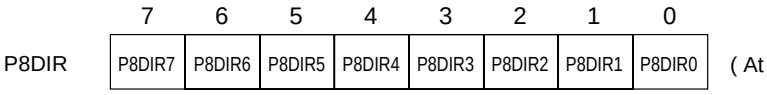
P8OUT	Output data
0	Low (V_{SS} level)
1	High (V_{DD} level)

Port 8 output register (P8OUT : x'03F18', R/W)



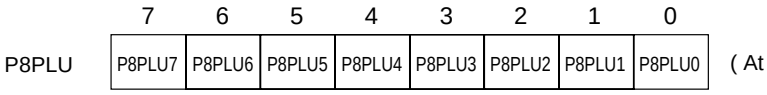
P8IN	Input data
0	Pin is Low (V_{SS} level).
1	Pin is High (V_{DD} level).

Port 8 input register (P8IN : x'03F28', R)



P8DIR	I/O mode selection
0	Input mode
1	Output mode

Port 8 direction control register (P8DIR : x'03F38', R/W)



P8PLU	Pull-up resistor selection
0	No pull-up resistor
1	Pull-up resistor

Port 8 pull-up resistor control register (P8PLU : x'03F48', R/W)

Figure 4-7-1 Port 8 Registers

4-7-3 Block Diagram

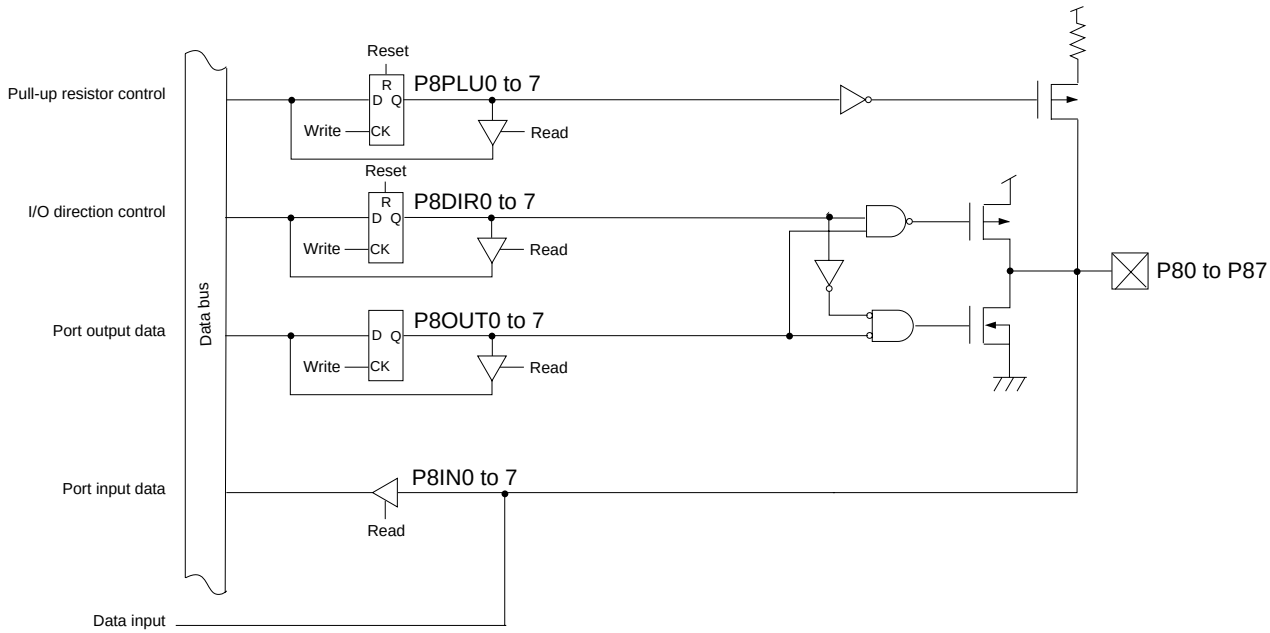


Figure 4-7-2 Block Diagram (P80 to P87)

4-8 Port A

4-8-1 Description

■General Port Setup

Port A direction control register (PADIR) controls I/O direction of each bit. When "1" is set to the control flag of port A direction control register (PADIR), output mode is set, and when "0" is set to there, input mode is set.

To read input data of pin, set "0" to the control flag of the port A direction control register (PADIR) and read out the value of port A output register (PAOUT).

To output data to pin, set "1" to the control flag of the port A direction control register (PADIR) and write data to port A output register (PAOUT).

Each bit can be set individually if pull-up / pull-down resistor is added or not, by the port A pull-up / pull-down resistor control register (PAPLUD). Set the control flag of the port A pull-up / pull-down resistor control register (PAPLUD) to "1" to add pull-up or pull-down resistor. The pull-up / pull-down resistor selection register (FLOAT) select if pull-up resistor or pull-down resistor is added. The bp6 of the pull-up / pull-down resistor control register (FLOAT) is set to "1" for pull-down resistor, set to "0" for pull-up resistor.

Either push-pull output or Nch open drain output can be selected with the port A output mode control register (PAODC).

■Special Function Pin Setup

PA0 to PA7 are used as input pins for analog. Each bit can be set individually as an input by the port A input mode register (PAIMD). When they are used as analog input pins, set the port A input mode register (PAIMD) to "1". Then, the value of the port A input register (PAIN) is read out "1".

By setting the control flag of the PAIMD register to "1", the through current is not occurred when input voltage is at intermediate level.

4-8-2 Registers

	7	6	5	4	3	2	1	0
PAOUT	PAOUT7	PAOUT6	PAOUT5	PAOUT4	PAOUT3	PAOUT2	PAOUT1	PAOUT0

(At reset : X X X X X X X X)

PAOUT	Output data
0	Output L level(Vss level)
1	Output H level(VDD level)

PortA output register (PAOUT : X'03F1A', R/W)

	7	6	5	4	3	2	1	0
PAIN	PAIN7	PAIN6	PAIN5	PAIN4	PAIN3	PAIN2	PAIN1	PAIN0

(At reset : X X X X X X X X)

PAIN	Input data
0	Pin is Low (VSS level).
1	Pin is High (VDD level).

PortA input register (PAIN : X'03F2A', R)

	7	6	5	4	3	2	1	0
PADIR	PADIR7	PADIR6	PADIR5	PADIR4	PADIR3	PADIR2	PADIR1	PADIR0

(At reset : 0 0 0 0 0 0 0 0)

PADIR	I/O mode selection
0	Inout mode
1	Output mode

PortA direction control register (P6DIR : X'03F3F', R/W)

	7	6	5	4	3	2	1	0
PAIMD	PAIMD7	PAIMD6	PAIMD5	PAIMD4	PAIMD3	PAIMD2	PAIMD1	PAIMD0

(At reset : 0 0 0 0 0 0 0 0)

PAIMD	I/O port, analog n input pin selection
0	I/O port
1	Analog n input pin

PortA input control register (PAIMD : X'03F3A', R/W)

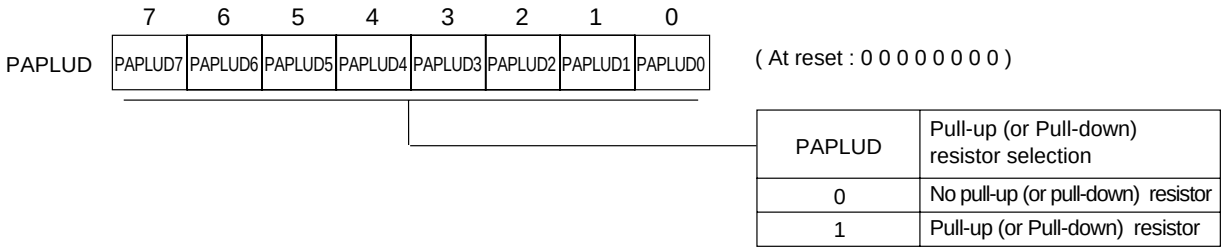
	7	6	5	4	3	2	1	0
PAODC	PAODC7	PAODC6	PAODC5	PAODC4	PAODC3	PAODC2	PAODC1	PAODC0

(At reset : 0 0 0 0 0 0 0 0)

PAODC	Nch open drain selection
0	Push-pull
1	Nch open drain

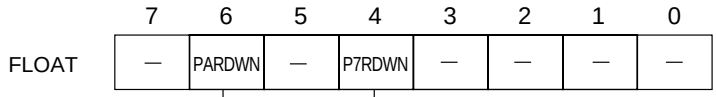
PortA output mode control register (PAODC : X'03F4F', R/W)

Figure 4-8-1 Port A Registers (1/2)



PAPLUD	Pull-up (or Pull-down) resistor selection
0	No pull-up (or pull-down) resistor
1	Pull-up (or Pull-down) resistor

PortA pull-up/pull-down resistor control resister (PAPLUD : X'03F4A', R/W)



P7RDWN	P7 Pull-up (or Pull-down) resistor selection
0	Pull-up resistor
1	Pull-down resistor

PARDWN	PA Pull-up (or Pull-down) resistor selection
0	Pull-up resistor
1	Pull-down resistor

Pull-up/pull-down resistor selection, pin control resister (FLOAT : X'03F2E', R/W)

Figure 4-8-2 Port A Registers (2/2)

4-8-3 Block Diagram

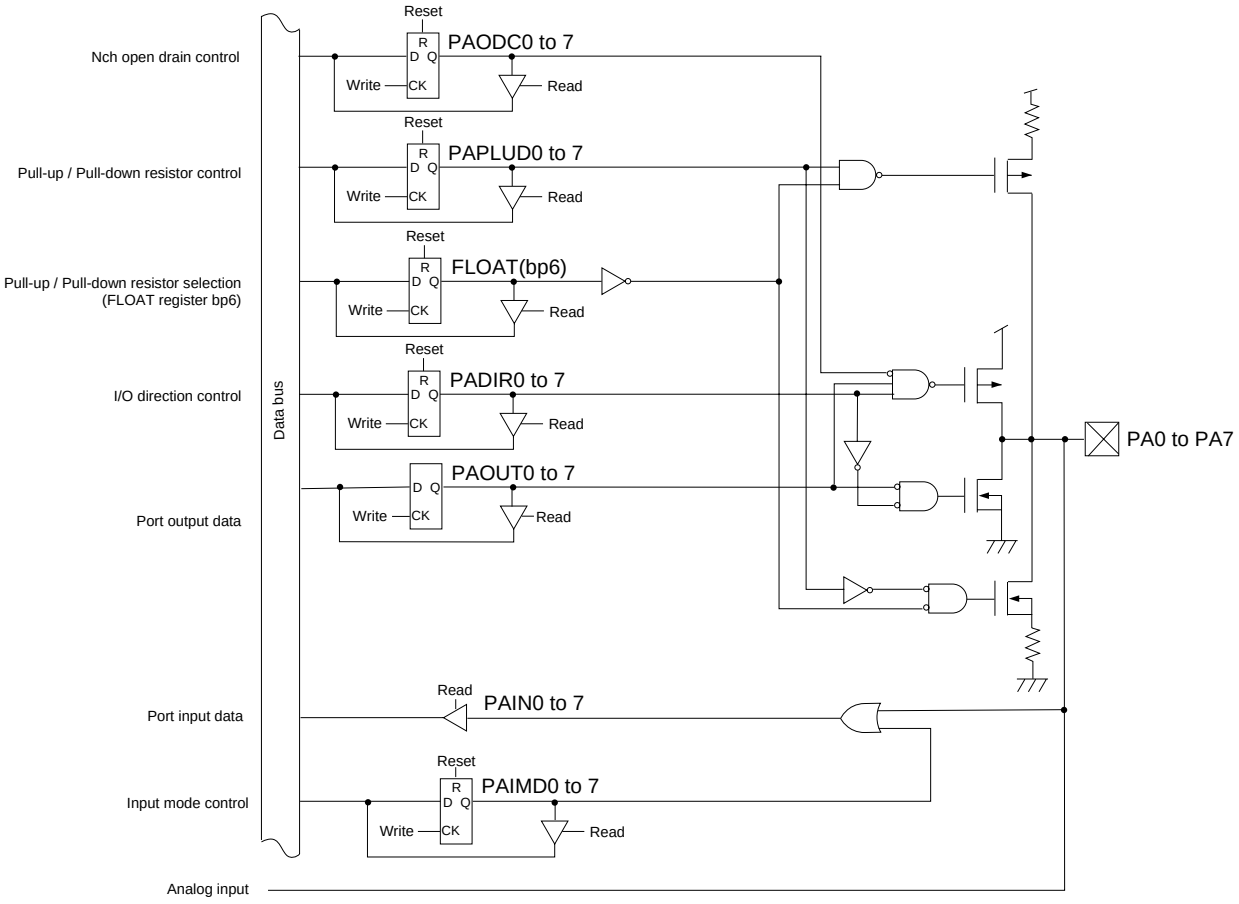


Figure 4-8-3 Block Diagram (PA0 to PA7)

4-9 Port C

4-9-1 Description

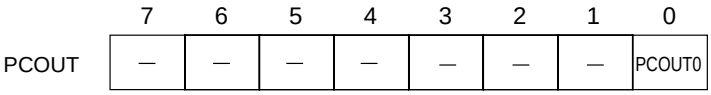
■General Port Setup
Each bit of the port C control I/O direction register (PCDIR) can be set individually to set pins as input or output. The control flag of the port C direction control register (PCDIR) is set to "1" for output mode, and "0" for input mode.

To read input data of pin, set the control flag of the port C direction control register (PCDIR) to "0" and read the value of the port C input register (PCIN).

To output data to pin, set the control flag of the port C direction control register (PCDIR) to "1" and write the value of the port C output register (PCOUT).

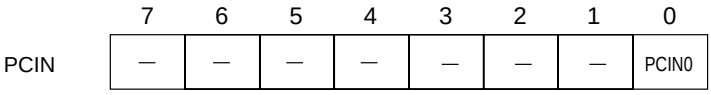
Each pin can be set individually if pull-up resistor is added or not, by the port C pull-up resistor control register (PCPLU). Set the control flag of the port C pull-up resistor control register (PCPLU) to "1" to add pull-up resistor.

4-9-2 Registers



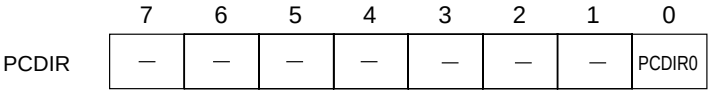
PCOUT	Output data
0	Low (V_{SS} level)
1	High (V_{DD} level)

Port C output register (PCOUT : x'03F1C', R/W)



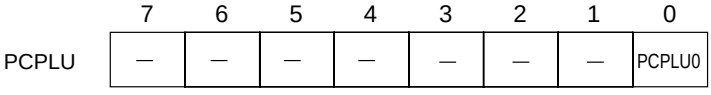
PCIN	Input data
0	Pin is Low (V_{SS} level).
1	Pin is High (V_{DD} level).

Port C input register (PCIN : x'03F2C', R)



PCDIR	I/O mode selection
0	Input mode
1	Output mode

Port C direction control register (PCDIR : x'03F3C', R/W)



PCPLU	Pull-up resistor selection
0	No pull-up resistor
1	Pull-up resistor

Port C pull-up resistor control resister (PCPLU : x'03F4C', R/W)

Figure 4-9-1 Port C Registers

4-9-3 Block Diagram

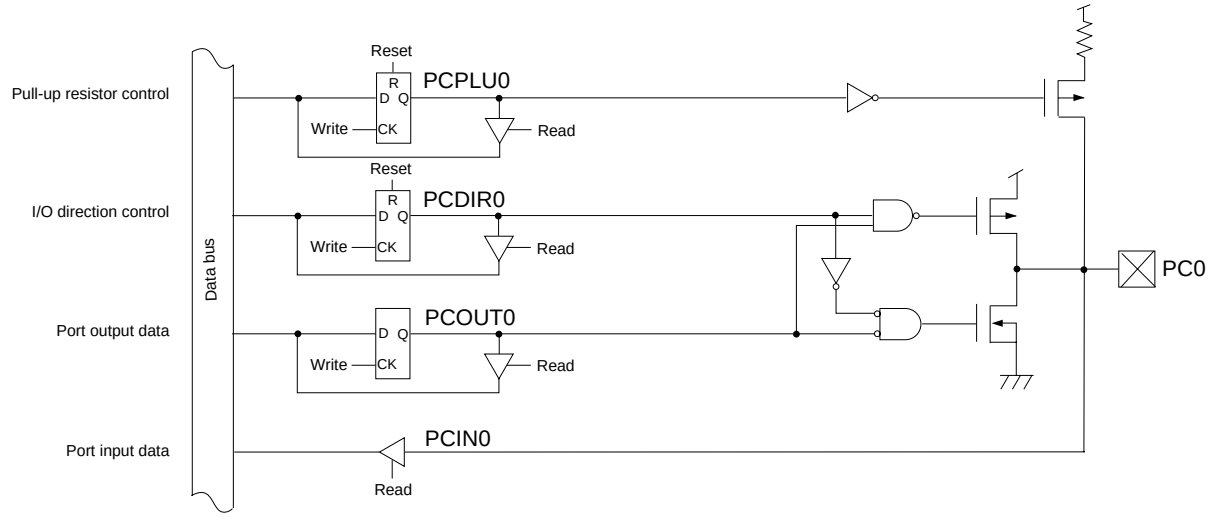


Figure 4-9-2 Block Diagram (PC0 to PC3)

Chapter 5 Prescaler

5-1 Overview

This LSI has 2 prescalers that can be used by its peripheral functions at the same time. Each of them count with fosc or fs as a base clock. Its hardware is constructed as follows ;

Prescaler 0 (fosc count)	7 bits prescaler
Prescaler 1 (fs count)	3 bits prescaler

Prescaler 0 outputs fosc/2, fosc/4, fosc/16, fosc/32, fosc/64, fosc/128 as cycle clock. Prescaler 1 outputs fs/2, fs/4, fs/8 as cycle clock. Prescaler is used when cycle clock based fosc and fs is used on the following peripheral functions ;

External interrupt 0 interface (with noise filter)
External interrupt 1 interface (with noise filter)
Timer 2 (8-bit timer counter)
Timer 3 (8-bit timer counter)
Serial interface 0 (Clock synchronous / Duplex UART)

About fosc, fs, refer to chapter 2. 2-5 Clock Switching [p.II-23].

5-1-1 Peripheral Functions

Table 5-1-1 shows several kinds of clock source that can be selected by each peripheral functions from prescaler output.

Table 5-1-1 Peripheral Functions Used with Prescaler Output

Clock source selection	Peripheral functions				
	External interrupt 0	External interrupt 1	Timer 2	Timer 3	Serial interface 0
fosc/2	-	-	-	-	√
fosc/4	-	-	√	√	√
fosc/16	-	-	√	√	√
fosc/32	-	-	√	-	-
fosc/64	-	-	√	√	√
fosc/128	√	√	-	√	-
fs/2	-	-	√	√	√
fs/4	-	-	√	-	√
fs/8	-	-	-	√	-
Timer 2 output	-	-	-	-	√
Timer 3 output	-	-	-	-	√

5-1-2 Block Diagram

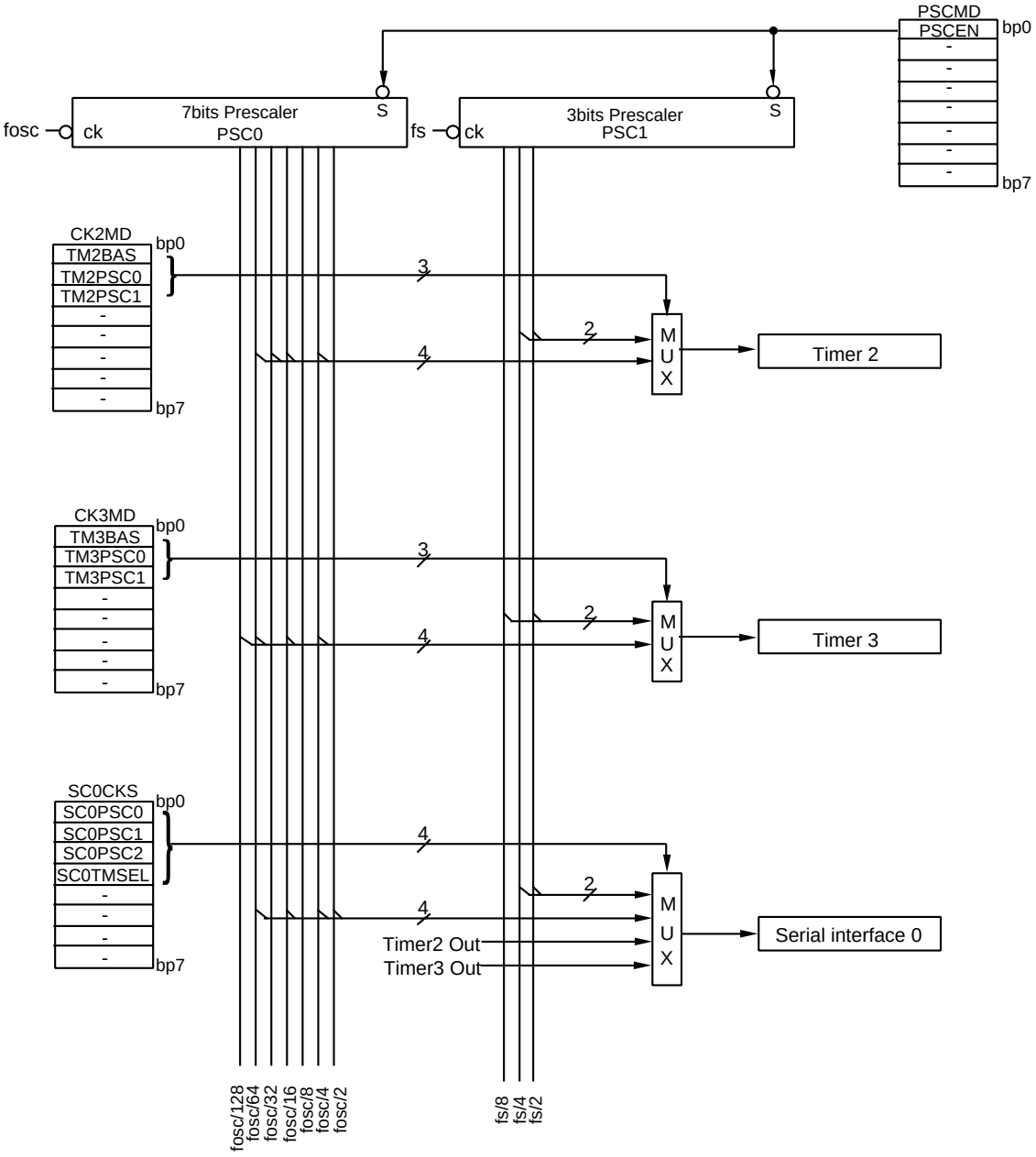


Figure 5-1-1 Prescaler Block Diagram

5-2 Control Register

5-2-1 Registers List

Table 5-2-1 shows registers to control prescaler.

Table 5-2-1 Prescaler Control Registers

Register	Address	R/W	Function	Page
PSCMD	x'03F6F'	R/W	Prescaler control register	V-6
CK2MD	x'03F5E'	R/W	Timer 2 prescaler selection register	V-7
CK3MD	x'03F5F'	R/W	Timer 3 prescaler selection register	V-7
SC0CKS	x'03F97'	R/W	Serial interface 0 transfer clock selection register	V-8

R/W : Readable/Writable

5-2-2 Control Registers

Registers that select prescaler outputs cycle clock and prescaler operation control, consists of the prescaler control register (PSCMD), the timer prescaler selection register (CKnMD) and the serial transfer clock selection register (SCnCKS).

The prescaler control register controls if counting of prescaler is permitted or not.

■Prescaler Control Register (PSCMD)

	7	6	5	4	3	2	1	0	
PSCMD	—	—	—	—	—	—	—	PSCEN	(At reset : - - - - - - - 0)

PSCEN	Prescaler 0, 1 count control
0	Disable the count
1	Enable the count

Figure 5-2-1 Prescaler Control Register (PSCMD : x'03F6F', R/W)

The timer prescaler selection register selects the count clock that used in 8-bit timer.

■Timer 2 Prescaler Selection Register (CK2MD)

	7	6	5	4	3	2	1	0	
CK2MD	—	—	—	—	—	TM2PSC1	TM2PSC0	TM2BAS	(At reset : - - - - - X X X)

TM2PSC1	TM2PSC0	TM2BAS	Clock source selection
0	0	0	fosc/4
	1		fosc/16
1	0		fosc/32
	1		fosc/64
-	0	1	fs/2
	1		fs/4

Figure 5-2-2 Timer 2 Prescaler Selection Register (CK2MD : x'03F5E', R/W)

■Timer 3 prescaler selection register (CK3MD)

	7	6	5	4	3	2	1	0	
CK3MD	—	—	—	—	—	TM3PSC1	TM3PSC0	TM3BAS	(At reset : - - - - - X X X)

TM3PSC1	TM3PSC0	TM3BAS	Clock source selection
0	0	0	fosc/4
	1		fosc/16
1	0		fosc/64
	1		fosc/128
-	0	1	fs/2
	1		fs/8

Figure 5-2-3 Timer 3 Prescaler Selection Register (CK3MD : x'03F5F', R/W)

The serial interface transfer clock selection register (SCnCKS) selects the transfer clock used for serial data transfer.

■Serial Interface 0 Transfer Clock Selection Register (SC0CKS)

	7	6	5	4	3	2	1	0	
SC0CKS	—	—	—	—	SC0TMSEL	SC0PSC2	SC0PSC1	SC0PSC0	(At reset : - - - - X X X X)

SC0TMSEL	SC0PSC2	SC0PSC1	SC0PSC0	Clock source selection
-	0	0	0	fosc/2
			1	fosc/4
		1	0	fosc/16
			1	fosc/64
	1	0	0	fs/2
			1	fs/4
0	1	1	-	Timer 2 output
1			-	Timer 3 output

Figure 5-2-4 Serial Interface 0 Transfer Clock Selection Register (SC0CKS : x'03F97', R/W)

5-3 Operation

5-3-1 Operation

■Prescaler Operation (Prescaler 0 to 1)
Prescaler 0 is a 7-bit and prescaler 1 is a 3-bit free-running counter that divides the base clock. This prescaler can be started or stopped by the PSCEN flag of the prescaler control register (PSCMD).

■Count Timing of Prescaler Operation (Prescalers 0 and 1)
Prescaler 0 counts up at the falling edge of fosc.
Prescaler 1 counts up at the falling edge of fs.

■Peripheral Functions with Prescaler Output Cycle Clock
Table 5-3-1 shows the prescaler output clock source that the peripheral functions can be used, and the registers that control the clock source selection.

Table 5-3-1 Peripheral Functions Used with Prescaler Output Cycle Clock

Peripheral functions		Control register
External interrupt 0	Noise filter sampling clock	-
External interrupt 1	Noise filter sampling clock	-
Timer 2	Count clock	CK2MD
Timer 3	Count clock	CK3MD
Serial 0	Transfer clock	SC0CKS

When the prescaler output clock source is used, counting of prescaler should be enabled before starting the peripheral functions.

5-3-2 Setup Example

■Prescaler Setup Example (Timer 2 count clock)
Select the clock of fosc/16 that is output from the prescaler 0, to the count clock of the timer 2.
An example setup procedure , with a description of each step is shown below.

Setup Procedure	Description
(1) Select the prescaler output. CK2MD (x'3F5E') bp2-1 : TM2PSC1-0 = 01 bp0 : TM2BAS = 0 (2) Enable the prescaler output. PSCMD (x'3F6F') bp0 : PSCEN = 1	(1) Select the prescaler output to fosc/16 by the TM2PSC1-0, TM2BAS flag of the timer 2 prescaler selection register (CK2MD). (2) Enable the prescaler counting by setting the PSCEN flag of the prescaler control register (PSCMD) to "1".

Enable the prescaler counting by the PSCEN flag of the prescaler control register (PSCMD). The prescaler counting is started after it is enabled.
Start the timer operation after the prescaler is set. Also, the selection of the prescaler output should be set by the timer mode register.

Chapter 6 8-bit Timers

6

6-1 Overview

This LSI contains two 8-bit timers (Timers 2 and 3) that can be also used as baud rate timer. Timers 2 and 3 can be used as 16-bit timers with cascade connection.

Fosc or fs can be selected as the clock source for each timer by using the prescaler. Also, remote control output circuit is built in.

6-1-1 Functions

Table 6-1-1 shows functions of each timer.

Table 6-1-1 Timer Functions

	Timer 2 (8 bit)	Timer 3 (8 bit)
Interrupt source	TM2IRQ	TM3IRQ
Timer operation	√	√
Event count	√	√
Timer pulse output	√	√
PWM output	√	-
Serial transfer clock output	√	√
Pulse width measurement	√	-
Cascade connection	√	
Remote control carrier output	-	√
Clock source	fosc fosc/4 fosc/16 fosc/32 fosc/64 fs/2 fs/4 fx TM2IO input	fosc fosc/4 fosc/16 fosc/64 fosc/128 fs/2 fs/8 fx TM3IO input
fosc : Machine clock (High speed oscillation) fx : Machine clock (Low speed oscillation) fs : System clock [☞ Chapter 2 2-5 Clock Switching] - When timers 2 and 4 are used as a baud rate timer for serial interface I function, it is not used as a general timer.		

6-1-2 Block Diagram

■Timers 2 and 3 Block Diagram

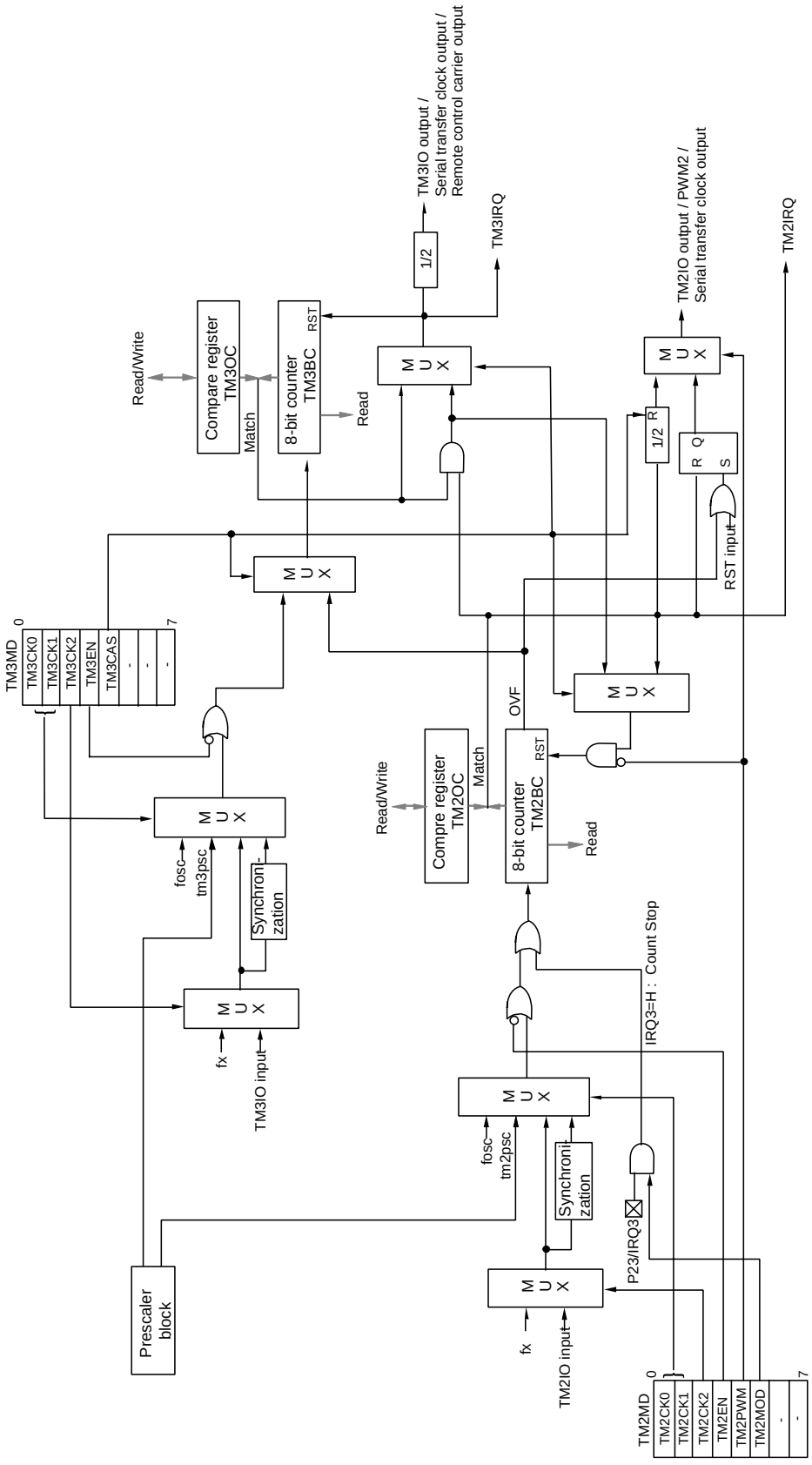


Figure 6-1-1 Timers 2 and 3 Block Diagram

■Remote Control Carrier Output Block Diagram

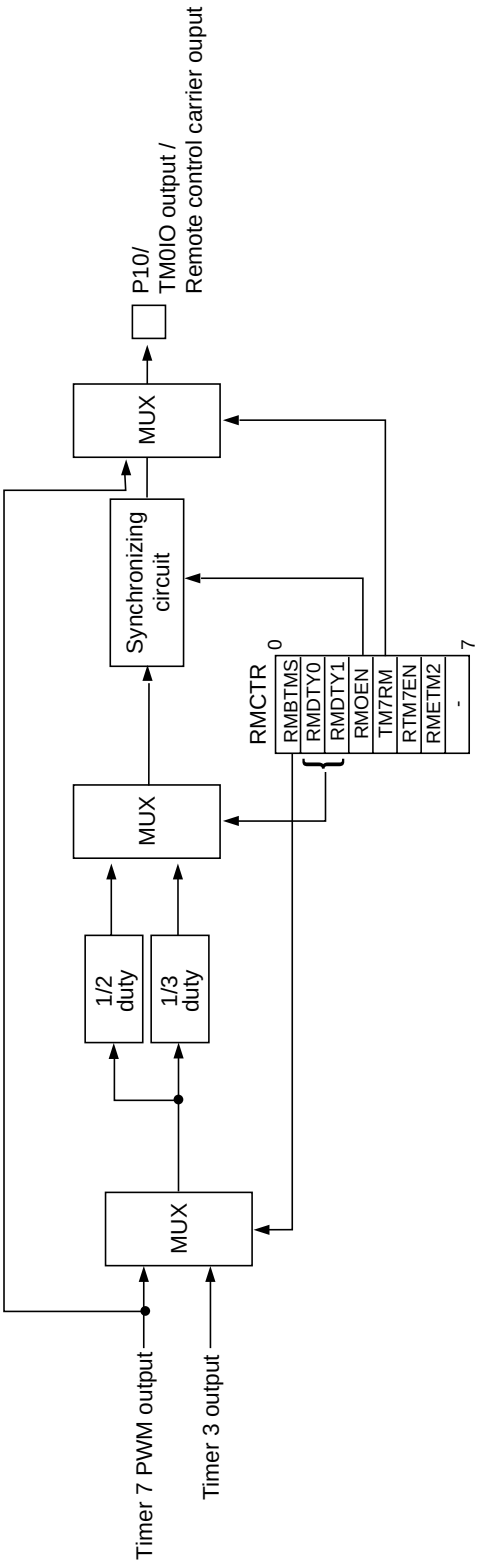


Figure 6-1-2 Remote Control Carrier Output Block Diagram

6-2 Control Registers

Timers 2 and 3 consist of the binary counter (TMnBC) and the compare register (TMnOC). And they are controlled by the mode register (TMnMD).
When the prescaler output is selected as the count clock source of timers 2 and 3, they should be controlled by the prescaler control register (PSCMD) and the prescaler selection register (CKnMD).
Remote control carrier output is controlled by the remote control carrier output control register (RMCTR).

6-2-1 Registers

Table 6-2-1 shows registers that control timers 2 and 3 and remote control carrier output

Table 6-2-1 8-bit Timer Control Registers

	Register	Address	R/W	Function	Page
Timer 2	TM2BC	x'03F58'	R	Timer 2 binary counter	VI-6
	TM2OC	x'03F5A'	R/W	Timer 2 compare register	VI-6
	TM2MD	x'03F5C'	R/W	Timer 2 mode register	VI-7
	CK2MD	x'03F5E'	R/W	Timer 2 prescaler selection register	V-7
	PSCMD	x'03F6F'	R/W	Prescaler control register	V-6
	TM2ICR	x'03FEB'	R/W	Timer 2 interrupt control register	III-21
	P1OMD	x'03F2F'	R/W	Port 1 output mode register	IV-13
	P1DIR	x'03F31'	R/W	Port 1 direction control register	IV-12
Timer 3	TM3BC	x'03F59'	R	Timer 3 binary counter	VI-6
	TM3OC	x'03F5B'	R/W	Timer 3 compare register	VI-6
	TM3MD	x'03F5D'	R/W	Timer 3 mode register	VI-8
	CK3MD	x'03F5F'	R/W	Timer 3 prescaler selection register	V-7
	PSCMD	x'03F6F'	R/W	Prescaler control register	V-6
	TM3ICR	x'03FEC'	R/W	Timer 3 interrupt control register	III-22
	P1OMD	x'03F2F'	R/W	Port 1 output mode register	IV-13
	P1DIR	x'03F31'	R/W	Port 1 direction control register	IV-12
Remote control carrier output	RMCTR	x'03F6E'	R/W	Remote control carrier output control register	VI-9

R/W : Readable / Writable
R : Readable only

6-2-2 Programmable Timer Registers

Each of timers 2 and 3 has 8-bit programmable timer registers. Programmable timer register consists of compare register and binary counter.

Compare register is 8-bit register which stores the value to be compared to binary counter.

■Timer 2 Compare Register (TM2OC)

	7	6	5	4	3	2	1	0	
TM2OC	TM2OC7	TM2OC6	TM2OC5	TM2OC4	TM2OC3	TM2OC2	TM2OC1	TM2OC0	(At reset : X X X X X X X X)

Figure 6-2-1 Timer 2 Compare Register (TM2OC : x'03F5A', R/W)

■Timer 3 Compare Register (TM3OC)

	7	6	5	4	3	2	1	0	
TM3OC	TM3OC7	TM3OC6	TM3OC5	TM3OC4	TM3OC3	TM3OC2	TM3OC1	TM3OC0	(At reset : X X X X X X X X)

Figure 6-2-2 Timer 3 Compare Register (TM3OC : x'03F5B', R/W)

Binary counter is 8-bit up counter. If any data is written to compare register during counting is stopped, binary counter is cleared to x'00'.

■Timer 2 Binary Counter (TM2BC)

	7	6	5	4	3	2	1	0	
TM2BC	TM2BC7	TM2BC6	TM2BC5	TM2BC4	TM2BC3	TM2BC2	TM2BC1	TM2BC0	(At reset : X X X X X X X X)

Figure 6-2-3 Timer 2 Binary Counter (TM2BC : x'03F58', R)

■Timer 3 Binary Counter (TM3BC)

	7	6	5	4	3	2	1	0	
TM3BC	TM3BC7	TM3BC6	TM3BC5	TM3BC4	TM3BC3	TM3BC2	TM3BC1	TM3BC0	(At reset : X X X X X X X X)

Figure 6-2-4 Timer 3 Binary Counter (TM3BC : x'03F59', R)

6-2-3 Timer Mode Registers

Timer mode register is readable/writable register that controls timers 2 and 3.

■Timer 2 Mode Register (TM2MD)

	7	6	5	4	3	2	1	0
TM2MD	-	-	TM2MOD	TM2PWM	TM2EN	TM2CK2	TM2CK1	TM2CK0

(At reset : - - 0 0 0 0 0 0)

TM2CK2	TM2CK1	TM2CK0	Clock source
-	0	0	fosc
		1	tm2psc(Prescaler output)
0	1	0	fx
		1	Synchronous fx
1		0	TM2IO input
		1	Synchronous TM2IOinput

TM2EN	Timer 2 count control
0	Disable the count
1	Enable the count

TM2PWM	Timer 2 operation mode
0	Normal timer operation
1	PWM operation

TM2MOD	Pulse width measurement control
0	Normal timer operation
1	Measure the pulse width of P23/IRQ3

Figure 6-2-5 Timer 2 Mode Register (TM2MD : x'03F5C', R/W)

■Timer 3 Mode Register (TM3MD)

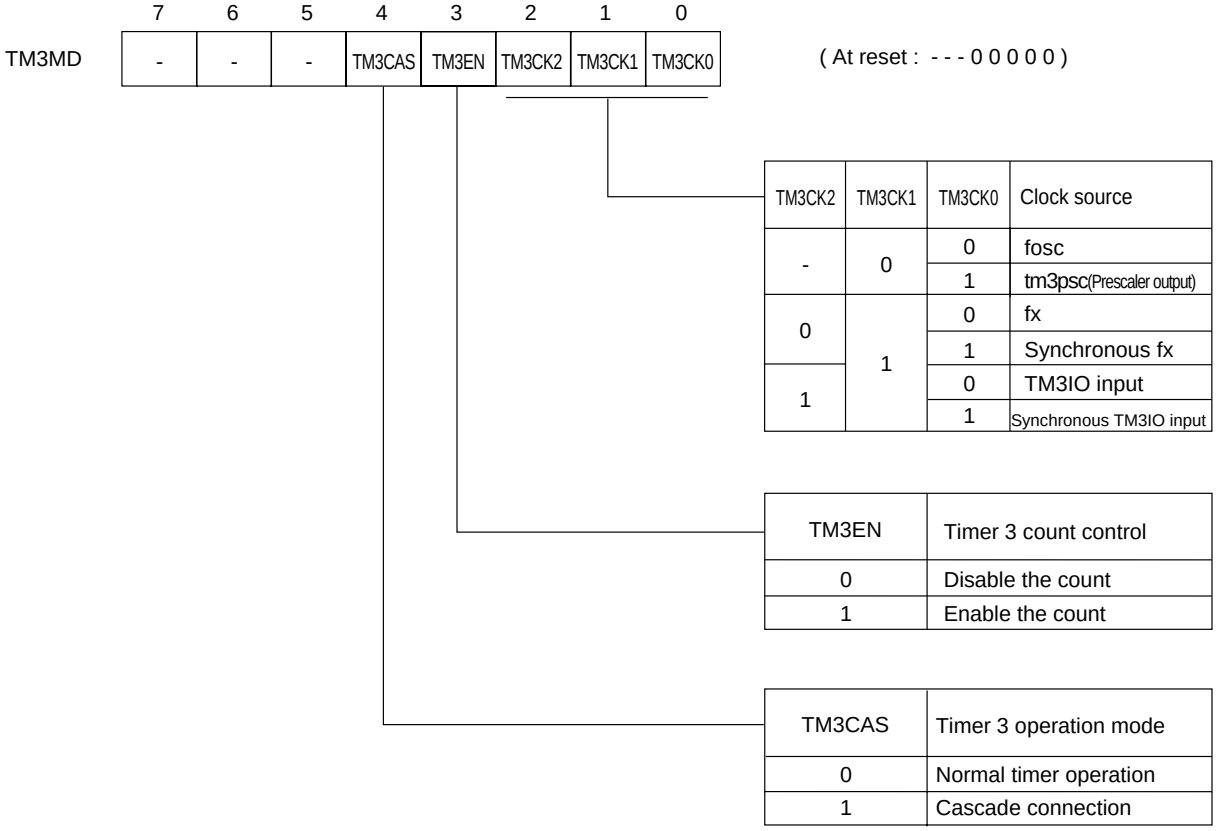
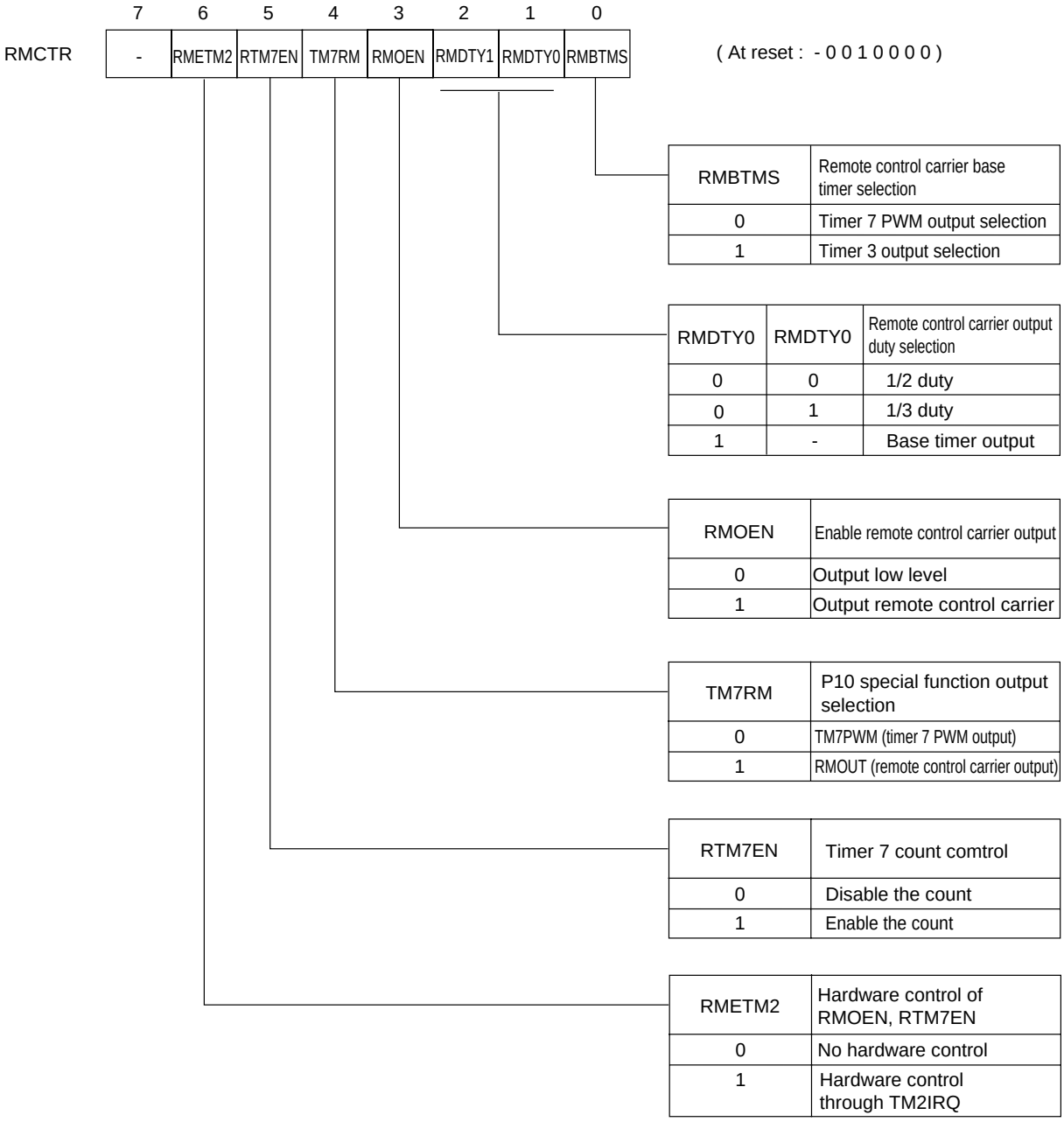


Figure 6-2-6 Timer 3 Mode Register (TM3MD : x'03F5D', R/W)

■Remote Control Carrier Output Control Register (RMCTR)



RMBTMS	Remote control carrier base timer selection
0	Timer 7 PWM output selection
1	Timer 3 output selection

RMDTY0	RMDTY0	Remote control carrier output duty selection
0	0	1/2 duty
0	1	1/3 duty
1	-	Base timer output

RMOEN	Enable remote control carrier output
0	Output low level
1	Output remote control carrier

TM7RM	P10 special function output selection
0	TM7PWM (timer 7 PWM output)
1	RMOUT (remote control carrier output)

RTM7EN	Timer 7 count comtrol
0	Disable the count
1	Enable the count

RMETM2	Hardware control of RMOEN, RTM7EN
0	No hardware control
1	Hardware control through TM2IRQ

Figure 6-2-7 Remote Control Carrier Output Control Register (RMCTR : x'03F6E', R/W)

If RMETM2 flag is set to "1" during timer 2 operation, RMOEN and RTM7EN flags are set every time timer 2 interrupt request (TM2IRQ) is generated.
Note that if timer 2 interrupt is enable, interrupt acceptance is occurred everytime TM2IRQ is generated.

6-3 8-bit Timer Count

6-3-1 Operation

The timer operation can constantly generate interrupts.

■8-bit Timer Operation (Timers 2 and 3)
The generation cycle of timer interrupts is set by the clock source selection and the setting value of the compare register (TMnOC), in advance. If the binary counter (TMnBC) reaches the setting value of the compare register, an interrupt is generated at the next count clock, then binary counter is cleared and counting is restarted from x'00'.
Table 6-3-1 shows clock source that can be selected.

Table 6-3-1 Clock Source (Timers 2 and 3) at Timer Operation

Clock source	1 count time	Timer 2 (8-bit)	Timer 3 (8-bit)
fosc	50 ns	√	√
fosc/4	200 ns	√	√
fosc/16	800 ns	√	√
fosc/32	1.6 µs	√	-
fosc/64	3.2 µs	√	√
fosc/128	6.4 µs	-	√
fs/2	200 ns	√	√
fs/4	400 ns	√	-
fs/8	800 ns	-	√
fx	30.5 µs	√	√
Notes : as fosc = 20 MHz fx = 32.768 kHz fs = fosc/2 = 10 MHz			

■Count Timing of Timer Operation (Timers 2 and 3)

Binary counter counts up with selected clock source as a count clock.

The basic operation of the whole function of 8-bit timer is as follows ;

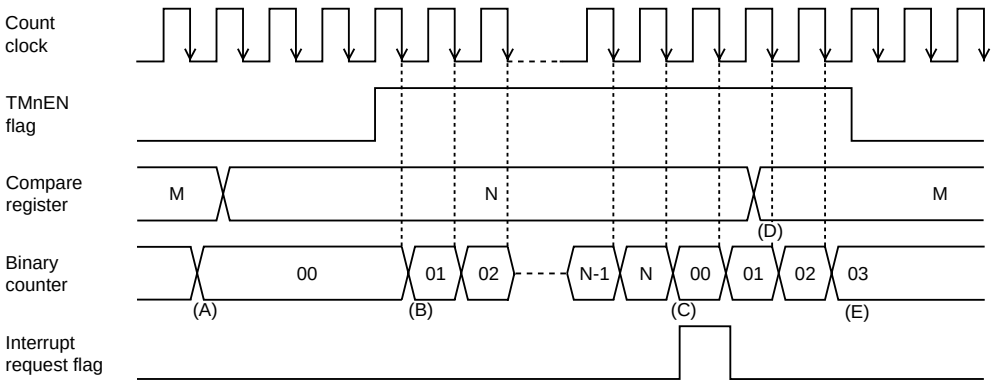


Figure 6-3-1 Count Timing of Timer Operation (Timers 2 and 3)

(A) If the value is written to the compare register during the TMnEN flag is "0", the binary counter is cleared to x'00', at the writing cycle.

(B) If the TMnEN flag is "1", the binary counter is started to count.
The counter starts to count up at the falling edge of the count clock.
But the binary counter doesn't count up at the first falling edge of the count clock.

(C) If the binary counter reaches the value of the compare register, the interrupt request flag is set at the next count clock, then the binary counter is cleared to x'00' and the counting is restarted.

(D) Even if the compare register is rewritten during the TMnEN flag is "1", the binary counter is not changed.

(E) If the TMnEN flag is "0", the binary counter is stopped.

When the binary counter reaches the value in the compare register, the interrupt request flag is set and the binary counter is cleared, at the next count clock. So set the compare register as:

Compare register setting = (count till the interrupt request - 1)

If the compare register is set the smaller than the binary counter during the count operation, the binary counter counts up to the overflow, at first.

If the interrupt is enabled, the timer interrupt request flag should be cleared before timer operation is started.

The timer n interrupt request generation (at TMnOC = x'00') has the same waveform at TMnOC = x'01'.

6-3-2 Setup Example

■Timer Operation Setup Example (Timers 2 and 3)

Timer function can be set by using timer 0 that generates the constant interrupt. By selecting fs/4 (at fosc = 20 MHz) as a clock source, interrupt is generated every 250 clock cycles (100 µs).

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter. TM2MD (x'3F5C') bp3 :TM2EN = 0	(1) Set the TM2EN flag of the timer 2 mode register (TM2MD) to "0" to stop the counting of timer 2.
(2) Select the normal timer operation. TM0MD (x'3F5C') bp4 :TM2PWM = 0 bp5 :TM2MOD = 0	(2) Set the TM2PWM flag and TM2MOD flag of the TM2MD register to "0" to select the normal timer operation.
(3) Select the count clock source. TM2MD (x'3F5C') bp2-0 :TM0CK2-0 = 001	(3) Select the prescaler output to the clock source by the TM2CK2-0 flag of the TM2MD register.
(4) Select the prescaler output and enable the counting. CK2MD (x'3F56') bp2-1 :TM2PSC1-0 = 01 bp0 :TM2BAS = 1 PSCMD (x'3F6F') bp0 :PSCEN = 1	(4) Select fs/4 to the prescaler output by the TM2PSC1-0, TM2BAS flag of the timer 2 prescaler selection register (CK2MD). Also, set the PSCEN flag of the prescaler control register (PSCMD) to "1" to enable the counting of the prescaler.
(5) Set the cycle of the interrupt generation. TM2OC (x'3F5A') = x'F9'	(5) Set the value of the interrupt generation cycle to the timer 2 compare register (TM2OC). The cycle is 250, so that the setting value is set to 249 (x'F9'). At that time, the timer 2 binary counter (TM2BC) is initialized to x'00'.
(6) Set the interrupt level. TM2ICR (x'3FEB') bp7-6 :TM2LV1-0 = 10	(6) Set the interrupt level by the TM2LV1-0 flag of the timer 2 interrupt control register (TM2ICR). If the interrupt request flag may be already set, clear the request flag. [☞ Chapter 3 3-1-4. Interrupt flag setting]

Setup Procedure	Description
(7) Enable the interrupt. TM2ICR (x'3FEB') bp1 :TM2IE = 1	(7) Set the TM2IE flag of the TM2ICR register to "1" to enable the interrupt.
(8) Start the timer operation. TM2MD (x'3F5C') bp3 :TM2EN = 1	(8) Set the TM2EN flag of the TM2MD register to "1" to start the timer 0.

The TM2BC starts to count up from 'x00'. When the TM2BC reaches the setting value of the TM2OC register, the timer 0 interrupt request flag is set at the next count clock, then the value of the TM2BC becomes x'00' and restart to count up.

When the TMnEN flag of the TMnMD register is changed at the same time with other bit, binary counter may start to count up by the switching operation.

If fx is selected as the count clock source, when the binary counter is read at operation, uncertain value on counting up may be read. To prevent this, select the synchronous fx as the count clock source.
In this case the timer n counter counts up in synchronization with system clock, therefore the correct value is always read.
But, if the synchronous fx is selected as the count clock source, CPU mode cannot return from STOP/HALT mode.

6-4 8-bit Event Count

6-4-1 Operation

Event count operation has 2 types ; TMnIO input and synchronous TMnIO input can be selected as the count clock.

■8-bit Event Count Operation

Event count means that the binary counter (TMnBC) counts the input signal from external to the TMnIO pin. If the value of the binary counter reaches the setting value of the compare register (TMnOC), interrupts can be generated at the next count clock.

Table 6-4-1 Event Count Input Clock

	Timer 2	Timer 3
Event input	TM2IO input (P12)	TM3IO input (P13)
	Synchronous TM2IO input	Synchronous TM3IO input

■Count Timing of TMnIO Input (Timers 2 and 3)

When TMnIO input is selected, TMnIO input signal is directly input to the count clock of the timer n. The binary counter counts up at the falling edge of the TMnIO input signal.

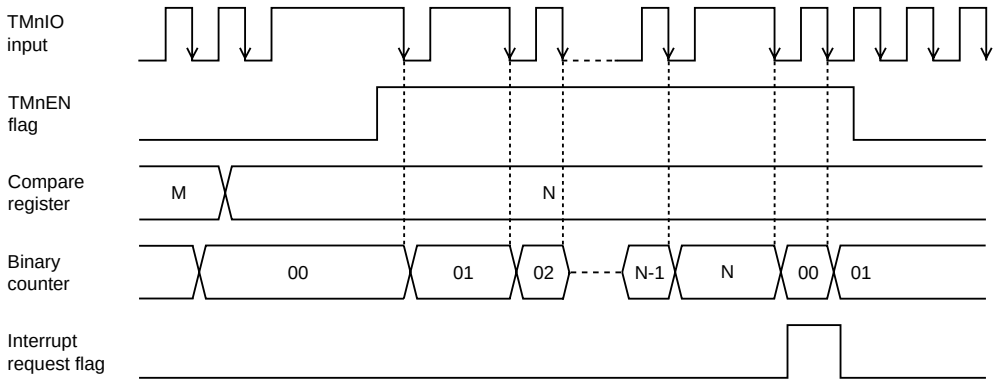


Figure 6-4-1 Count Timing of TMnIO Input (Timers 2 and 3)

When the TMnIO input is selected for count clock source and the value of the timer n binary counter is read during operation, incorrect value at count up may be read out. To prevent this, use the event count by synchronous TMnIO input, as the following page.

■Count Timing of Synchronous TMnIO Input (Timers 2 and 3)

If the synchronous TMnIO input is selected, the synchronizing circuit output signal is input to the timer n count clock. The synchronizing circuit output signal is changed at the falling edge of the system clock after TMnIO input signal is changed.

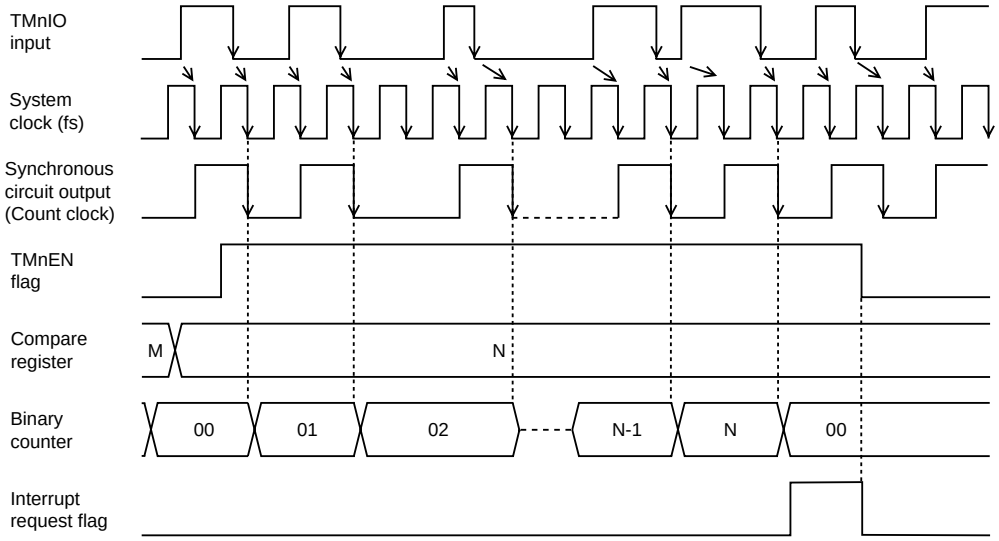


Figure 6-4-2 Count Timing of Synchronous TMnIO Input (Timers 2 and 3)

When the synchronous TMnIO input is selected as the count clock source, the timer n counter counts up in synchronization with system clock, therefore the correct value is always read.

But, if the synchronous TMnIO is selected as the count clock source, CPU mode cannot return from STOP/HALT mode.

6-4-2 Setup Example

■Event Count Setup Example (Timers 2 and 3)

If the falling edge of the TM2IO input pin signal is detected 5 times with using timer 2, an interrupt is generated.

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter. TM2MD (x'3F5C') bp3 :TM2EN = 0	(1) Set the TM2EN flag of the timer 2 mode register (TM2MD) to "0" to stop timer 2 counting.
(2) Set the special function pin to input. P1DIR (x'3F31') bp2 :P1DIR2 = 0	(2) Set the P1DIR2 flag of the port 1 direction control register (P1DIR) to "0" to set P12 pin to input mode. If it needs, pull up resistor should be added. [☞ Chapter 4. I/O Ports]
(3) Select the normal timer operation. TM0MD (x'3F5C') bp4 :TM2PWM = 0 bp5 :TM2MOD = 0	(3) Set the TM2PWM flag and TM2MOD flag of the TM2MD register to "0" to select the normal timer operation.
(4) Select the count clock source. TM2MD (x'3F5C) bp2-0 :TM2CK2-0 = 110	(4) Select the clock source to TM2IO input by the TM2CK2-0 flag of the TM2MD register.
(5) Set the interrupt generation cycle. TM2OC (x'3F5A') = x'04'	(5) Set the timer 2 compare register (TM2OC) the interrupt generation cycle. Counting is 5, so the setting value should be 4. At that time, the timer 2 binary counter (TM2BC) is initialized to x'00'.
(6) Set the interrupt level. TM2ICR (x'3FEB') bp7-6 :TM2LV1-0 = 10	(6) Set the interrupt level by the TM2LV1-0 flag of the timer 2 interrupt control register (TM2ICR). If the interrupt request flag may be already set, clear all existing interrupt requests. [☞ Chapter 3 3-1-4. Interrupt Flag Setup]

Setup Procedure	Description
(7) Enable the interrupt. TM2ICR (x'3FEB') bp1 :TM2IE = 1	(7) Set the TM2IE flag of the TM2ICR register to "1" to enable the interrupt.
(8) Start the event counting. TM2MD (x'3F5C') bp3 :TM0EN = 1	(8) Set the TM2EN flag of the TM2MD register to start timer 2.

Every time TM2BC detects the falling edge of TM2IO input , TM2BC counts up from 'x00'. When TM2BC reaches the setting value of theTM2OC register, the timer 2 interrupt request flag is set at the next count clock, then the value of TM2BC becomes x'00' and counting up is restarted.

6-5 8-bit Timer Pulse Output

6-5-1 Operation

The TMnIO pin can output a pulse signal with any cycle.

■Operation of Timer Pulse Output (Timers 2 and 3)
The timers can output 2 x cycle signal, compared to the setting value in compare register (TMnOC). Output pins are as follows ;

Table 6-5-1 Timer Pulse Output Pins

	Timer 2	Timer 3
Pulse output pin	TM2IO output (P12)	TM3IO output (P13)

■Count Timing of Timer Pulse Output (Timers 2 and 3)

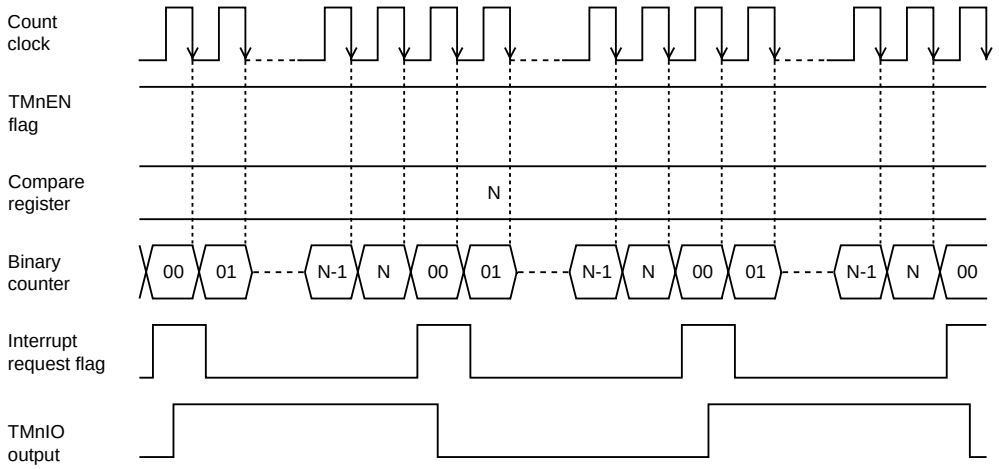


Figure 6-5-1 Count Timing of Timer Pulse Output (Timers 2 and 3)

The TMnIO pin outputs 2 x cycle, compared to the value in the compare register. If the binary counter reaches the compare register, and the binary counter is cleared to x'00', TMnIO output is inverted. The inversion of the timer output is changed at the rising edge of the count clock. This is happened to form waveform inside to correct the output cycle.

6-5-2 Setup Example

■Timer Pulse Output Setup Example (Timers 2 and 3)
TM2IO pin outputs 50 kHz pulse by using timer 2. For this, select fosc as clock source, and set a 1/2 cycle (100 kHz) for the timer 0 compare register (at fosc=20 MHz).
An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter. TM2MD (x'3F5C') bp3 :TM2EN = 0	(1) Set the TM2EN flag of the timer 2 mode register (TM2MD) to "0" to stop timer 2 counting.
(2) Set the special function pin to the output mode. P1OMD (x'3F2F') bp2 :P1OMD2 = 1 P1DIR (x'3F31') bp2 :P1DIR2 = 1	(2) Set the P1OMD2 flag of the port 1 output mode register (P1OMD) to "1" to set P12 the special function pin. Set the P1DIR2 flag of the port 1 direction control register (P1DIR) to "1" to set output mode. If it needs, pull-up resistor should be added. [☞ Chapter 4. I/O Ports]
(3) Select the normal timer operation. TM0MD (x'3F5C') bp4 :TM2PWM = 0 bp5 :TM2MOD = 0	(3) Set the TM2PWM flag and TM2MOD flag of the TM2MD register to "0" to select the normal timer operation.
(4) Select the count clock source. TM2MD (x'3F5C') bp2-0 :TM2CK2-0 = 000	(4) Select fosc for the clock source by the TM2CK2-0 flag of the TM2MD register.
(5) Set the timer pulse output cycle. TM2OC (x'3F5A') = x'C7'	(5) Set the timer 2 compare register (TM2OC) to the 1/2 of the timer pulse output cycle. The setting value should be 200-1=199(x'C7'), because 100 kHz is divided by 20 MHz. At that time, the timer 2 binary counter (TM2BC) is initialized to x'00'.
(6) Start the timer operation. TM2MD (x'3F5C') bp3 :TM2EN = 1	(6) Set the TM2EN flag of the TM2MD register to "1" to start timer 2.

TM2BC counts up from x'00'. If TM2BC reaches the setting value of the TM2OC register, then TM2BC is cleared to x'00', TM2IO output signal is inverted and TM2BC restarts to count up from x'00'.

At TMnOC = x'00', timer pulse output has the same waveform to at x'01'.

If any data is written to compare register binary counter is stopped, timer output is reset to "L".

Set the compare register value as follows.

$$\text{The compare register value} = \frac{\text{The timer pulse output cycle}}{\text{The count clock cycle x 2}} - 1$$

6-6 8-bit PWM Output

The TMnIO pin outputs the PWM waveform, which is determined by the match timing for the compare register and the overflow timing of the binary counter.

6-6-1 Operation

■Operation of 8-bit PWM Output (Timers 2)

The PWM waveform with any duty cycle is generated by setting the duty cycle of PWM "H" period to the compare register (TMnOC). The cycle is the period from the full count to the overflow of the 8-bit timer. Table 6-6-1 shows PWM output pins ;

Table 6-6-1 Output Pins of PWM Output

	Timer 2
PWM output pin	TM2IO output pin (P12)

■Count Timing of PWM Output (at normal) (Timers 2)

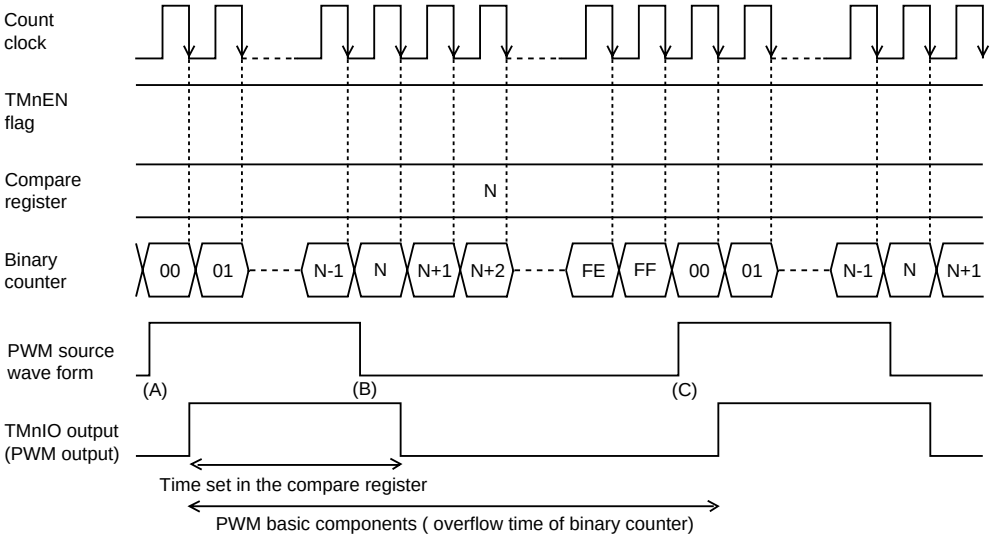


Figure 6-6-1 Count Timing of PWM Output (at Normal)

PWM source waveform,

(A) is "H" while counting up from x'00' to the value stored in the compare register.

(B) is "L" after the match to the value in the compare register, then the binary counter continues counting up till the overflow.

(C) is "H" again, if the binary counter overflow.

The PWM outputs the PWM source waveform with 1 count clock delay. This is happened, because the waveform is created inside to correct the output cycle.

■Count Timing of PWM Output (when the compare register is x'00') (Timers 2)
Here is the count timing when the compare register is set to x'00' ;

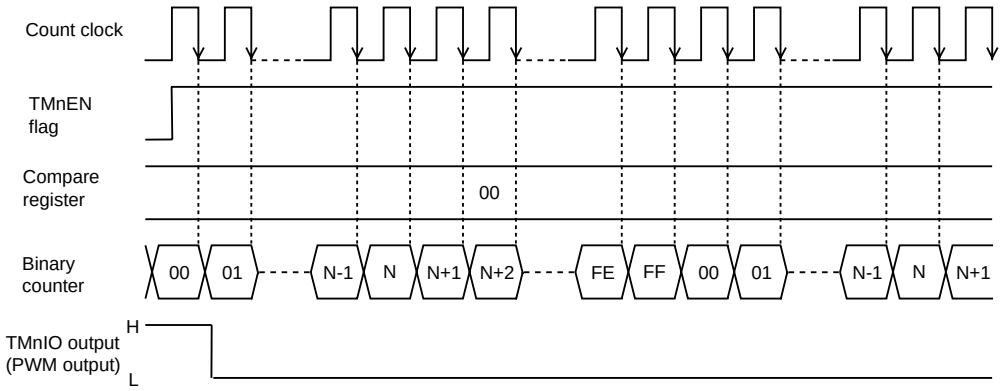


Figure 6-6-2 Count Timing of PWM Output (when compare register is x'00')

When TMnEN flag is stopped ("0") PWM output is "H".

■Count Timing of PWM Output (when the compare register is x'FF') (Timers 2)
Here is the count timing when the compare register is set to x'FF' ;

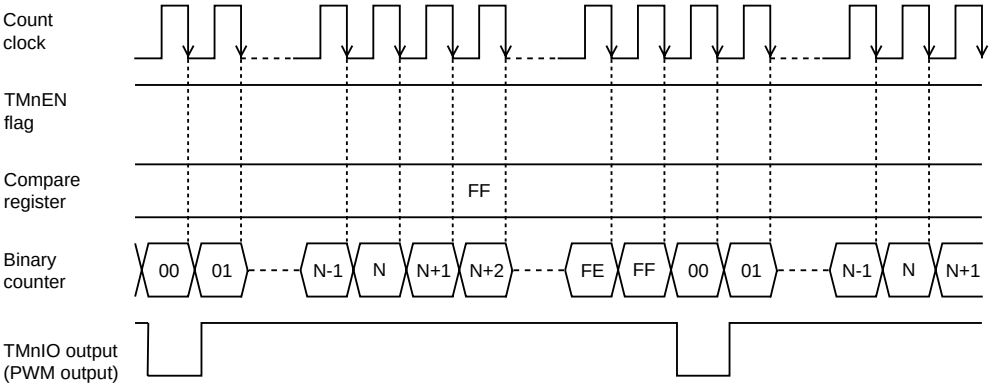


Figure 6-6-3 Count Timing of PWM Output (when compare register is x'FF')

6-6-2 Setup Example

■PWM Output Setup Example (Timers 2)

The 1/4 duty cycle PWM output waveform is output from the TM2IO output pin at 128 Hz by using timer 2 (at fx=32.768 kHz). Cycle period of PWM output waveform is decided by the overflow of the binary counter. "H" period of the PWM output waveform is decided by the setting value of the compare register. An example setup procedure, with a description of each step is shown below.

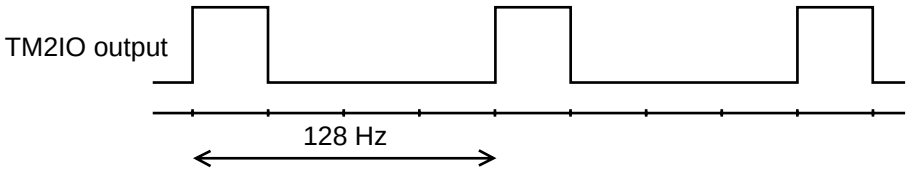


Figure 6-6-4 Output Waveform of TM0IO Output Pin

Setup Procedure	Description
(1) Stop the counter. TM2MD (x'3F5C') bp3 :TM2EN = 0	(1) Set the TM2EN flag of the timer 2 mode register (TM2MD) to "0" to stop the timer 2 counting.
(2) Set the special function pin to the output mode. P1OMD (x'3F2F') bp2 :P1OMD2 = 1 P1DIR (x'3F31') bp2 :P1DIR2 = 1	(2) Set the P1OMD2 flag of the port 1 output mode register (P1OMD) to "1" to set P12 pin to the special function pin. Set the P1DIR2 flag of the port 1 direction control register (P1DIR) to "1" for the output mode. If it needs, pull up resistor should be added. [☞ Chapter 4. I/O Ports]
(3) Select the PWM operation. TM2MD (x'3F5C') bp4 :TM2PWM = 1 bp5 :TM2MOD = 0	(3) Set the TM2PWM flag of the TM2MD register to "1", the TM2MOD flag to "0" to select the PWM operation.
(4) Select the count clock source. TM2MD (x'3F5C') bp2-0 :TM2CK2-0 = 010	(4) Select "fx" for the clock source by the TM2CK2-0 flag of the TM2MD register.

Setup Procedure	Description
(5) Set the period of PWM "H" output. TM2OC (x'3F5A') = x'40'	(5) Set the "H" period of PWM output to the timer 2 compare register (TM2OC). The setting value is set to 256 / 4 = 64 (x'40'), because it should be the 1/4 duty of the full count (256). At that time, the timer 2 binary counter (TM2BC) is initialized to x'00'.
(6) Start the timer operation. TM2MD (x'3F5C') bp3 :TM2EN = 1	(6) Set the TM2EN flag of the TM2MD register to "1" to operate timer 2.

TM2BC counts up from x'00'. PWM source waveform outputs "H" till TM2BC reaches the setting value of the TM2OC register, and outputs "L" after that. Then, TM2BC continues counting up, and PWM source waveform outputs "H" again, once overflow happens, and TM2BC restarts counting up from x'00'. TM2IO pin outputs the PWM source waveform with 1 count clock delay.

The initial setting of PWM output is changed from "L" output to "H" output at the selection of PWM operation by the TMnPWM flag of the TMnMD register.

6-7 Serial Interface Transfer Clock Output

6-7-1 Operation

Serial interface transfer clock can be created by using the timer output signal.

■Serial InterfaceTransfer Clock Operation by 8-bit Timer (Timers 2 and 3)
Timer 2 output and timer 3 output can be used as a transfer clock source for serial interface 0.

Table 6-7-1 Timer for Serial Interface Transfer Clock

Serial transfer clock	Timer 2	Timer 3
Serial interface 0	√	-

When timer output is selected as serial interface transfer clock, select fosc as a clock source of the timer. If other clock is selected, normal transfer of serial interface data is not guaranteed.

■Timing of Serial Interface Transfer Clock (Timers 2 and 3)

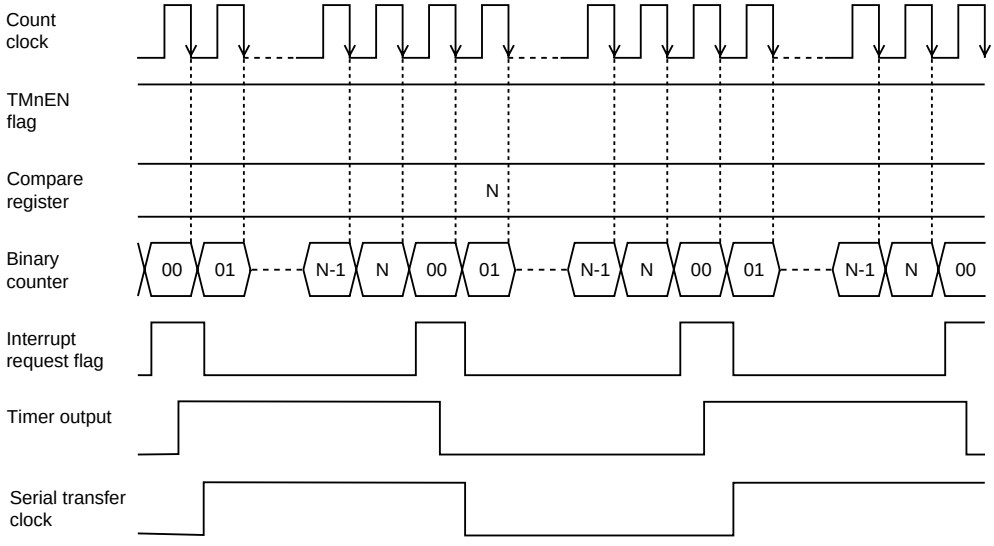


Figure 6-7-1 Timing of Serial Interface Transfer Clock (Timers 2 and 3)

The timer output is synchronized to the serial transfer clock by the timer count clock, and its frequency is 1/2 of the set frequency by the compare register.
Other count timings are same to the timing of timer operation. For the baud rate calculation and the serial interface setup, refer to chapter 11. Serial Interface 0.

6-7-2 Setup Example

■Serial Interface Transfer Clock Setup Example (Timer 2)

How to create a transfer clock for full duplex UART (Serial interface 0) using with timer 2 is shown below. The baud rate is selected to be 300 bps, the source clock of timer 2 is selected to be fs/4 (at fosc=8 MHz).

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter. TM2MD (x'3F5C') bp3 :TM2EN = 0	(1) Set the TM2EN flag of the timer 2 mode register (TM2MD) to "0" to stop timer 2 counting.
(2) Select the normal timer operation. TM2MD (x'3F5C') bp4 :TM2PWM = 0 bp5 :TM2MOD = 0	(2) Set the TM2PWM flag and TM2MOD flag of the TM2MD register to "0" to select the normal timer operation.
(3) Select the count clock source. TM2MD (x'3F5C') bp2-0 :TM2CK2-0 = 001	(3) Select the clock source to prescaler output by the TM2CK2-0 flag of the TM2MD register.
(4) Select the prescaler output and enable counting. CK2MD (x'3F5E') bp2-1 :TM2PSC1-0 = 01 bp0 :TM2BAS = 1 PSCMD (x'3F6F') bp0 :PSCEN = 1	(4) Select the prescaler output to fs/4 by the TM2PSC1-0, TM2BAS flag of the timer 2 prescaler selection register (CK2MD). Also, set the PSCEN flag of the prescaler control register (PSCMD) to "1" to enable the prescaler counting.
(5) Set the baud rate. TM2OC (x'3F5A') = x'CF'	(5) Set the timer 2 compare register (TM2OC) to the value that baud rate comes to 300 bps. [☞ Chapter 11. Table 11-3-18] At that time, the timer 2 binary counter (TM2BC) is initialized to x'00'.
(6) Start the timer operation TM2MD (x'3F5C') bp3 :TM2EN = 1	(6) Set the TM2EN flag of the TM2MD register to "1" to start timer 2.

TM2BC counts up from x'00'. Timer 2 output is the clock of the serial interface 0 at transmission and reception.

For the compare register setup value and the serial operation setup, refer to chapter 11. Serial Interface 0.

6-8 Simple Pulse Width Measurement

6-8-1 Operation

Timer measures the "L" duration of the pulse signal input from the external interrupt pin.

■Simple Pulse Width Measurement Operation by 8-bit Timer (Timers 2)
During the input signal of the external interrupt pin (simple pulse width) is "L", the binary counter of the timer counts up. Pulse width "L" period can be measured by reading the count of timer. 8-bit timers that have the simple pulse width measurement function is timers 2.

Table 6-8-1 Simple Pulse Width Measurement Able Pins (Timers 2)

	Timer 2
Simple pulse width measurement enable pin	External interrupt 3 (P23/IRQ3)

■Count Timing of Simple Pulse Width Measurement (Timer 2)

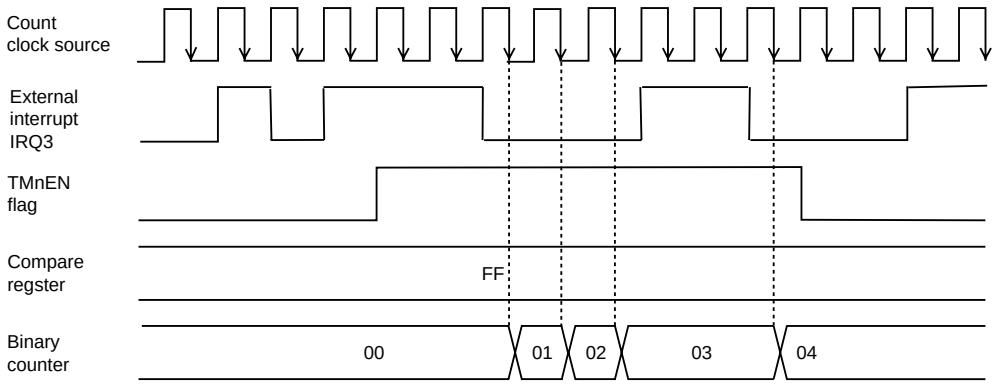


Figure 6-8-1 Count Timing at Measurement of Simple Pulse Width (Timer 2)

During the input signal of the external interrupt pin for simple pulse width measurement is "L" at TMnEN flag operation ("1"), timer counts up.

6-8-2 Setup Example

■Set up Example of Simple Pulse Width Measurement by 8-bit Timer (Timers 2)
The pulse width of 'L" period of the external interrupt 3 (IRQ3) input signal is measured by timer 2. The clock source of timer 2 is selected to fosc.
An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter. TM2MD (x'3F5C') bp3 :TM2EN = 0	(1) Set the TM2EN flag of the timer 2 mode register (TM2MD) to stop timer 2 counting.
(2) Set the pulse width measurement operation. TM2MD (x'3F5C') bp4 :TM2PWM = 0 bp5 :TM2MOD = 1	(2) Set the TM2PWM flag of the TM2MD register to "0" and TM2MOD flag to "1" to enable the timer operation during "L" period to be measured.
(3) Select the count clock source. TM2MD (x'3F5C') bp2-0 : TM2CK2-0 = 000	(3) Set the clock source to fosc by the TM2CK2-0 flag of the TM2MD register.
(4) Set the compare register. TM2OC (X'3F5A') = x'FF'	(4) Set the timer 2 compare register (TM2OC) to the bigger value than ("L"period of measured pulse width / the cycle of fosc). At that time, the timer 2 binary counter (TM2BC) is initialized to x'00'.
(5) Set the interrupt level IRQ3ICR (x'3FE5') bp7-6 :IRQ3LV1-0 = 10	(5) Set the interrupt level by the IRQ3LV1-0 flag of the external interrupt 3 control register (IRQ3ICR). If interrupt request flag is already set, clear all interrupt request flags. [☞ Chapter 3. 3-1-4 Interrupt Flag Setup]
(6) Set the interrupt valid edge. IRQ3ICR (x'3FE5') bp5 :REDG3 = 1	(6) Set the REDG3 flag of the IRQ3ICR register to "1" to specify the interrupt valid edge to the rising edge.

Setup Procedure	Description
(7) Enable the interrupt. IRQ3ICR (x'3FE5') bp1 :IRQ3IE = 1	(7) Set the IRQ2IE flag of the IRQ3ICR register to "1" to enable the interrupt.
(8) Enable the timer operation. TM2MD (x'3F5C') bp3 :TM2EN = 1	(8) Set the TM2EN flag of the TM2MD register to "1" to enable timer 2 operation.

TM2BC starts to count up with negative edge of the external interrupt 3 (IRQ3) input as a trigger. Timer 2 continues to count up during "L" period of IRQ3 input, then stop the counting with positive edge of IRQ3 input as a trigger. At the same time, reading the value of TM2BC by interrupt handling can detects "L" period.

6-9 Cascade Connection

6-9-1 Operation

Cascading timers 2 and 3 form a 16-bit timer.

■8-bit Timer Cascade Connection Operation (Timer 2 + Timer 3)
Timer 2 and timer 3 are combined to be a 16-bit timer. Cascading timer is operated at clock source of timer 2 which is lower 8 bits.

Table 6-9-1 Timer Functions at Cascade Connection

	Timer 2 + Timer 3 (16 bit)
Interrupt source	TM3IRQ
Timer operation	√
Event count	√ (TM2IO input)
Timer pulse output	√ (TM3IO output)
PWM output	-
Serial Interface transfer clock output	√ (TM3IO output)
Pulse width measurement	√
Remote control carrier output	√
Clock source	fosc fosc/4 fosc/16 fosc/32 fosc/64 fs/2 fs/4 fx TM2IO input
fosc : Machine clock (High speed oscillation) fx : Machine clock (Low speed oscillation) fs : System clock [☞ Chapter 2 2-5 Clock Switching]	

At cascade connection, the binary counter and the compare register are operated as a 16 bit register. At operation, set the TMnEN flag of the upper and lower 8-bit timers to "1" to be operated.
Also, the clock source is the one which is selected in the lower 8-bit timer.
Other setup and count timing is the same to the 8-bit timer at independently operation.

When timer 2 and timer 3 are used in cascade connection, timer 3 interrupt request flag is used. Timer pulse output of timer 2 is "L" fixed output.
An interrupt request of timer 2 is not generated, and the timer 2 interrupt should be disabled.

At the cascade connection, if the binary counter should be cleared by rewriting the compare register, the TMnEN flags of the lower and upper 8 bits timers mode registers should be set to "0" to stop the counting, then rewrite the compare register.

6-9-2 Setup Example

■Cascade Connection Timer Setup Example (Timer 2 + Timer 3)

Setting example of timer function that an interrupt is constantly generated by cascade connection of timer 2 and timer 3, as a 16-bit timer is shown. An interrupt is generated in every 2500 cycles (1 ms) by selecting source clock to fs/4 (fosc=20 MHz at operation).

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter. TM2MD (x'3F5C') bp3 :TM0EN = 0 TM3MD (x'3F5D') bp3 :TM3EN = 0	(1) Set the TM2EN flag of the timer 2 mode register (TM2MD) to "0", the TM3EN flag of the timer 3 mode register to "0" to stop timer 2 and timer 3 counting.
(2) Select the normal operation lower timer. TM2MD (x'3F5C') bp4 :TM2PWM = 0 bp5 :TM2MOD = 0	(2) Set both of the TM2PWM flag and TM2MOD flag of the TM2MD register to "0" to select the normal operation of timer 2.
(3) Set the cascade connection. TM3MD (x'3F5D') bp4 :TM3CAS = 1	(3) Set the TM3CAS flag of the TM3MD register to "1" to connect timer 3 and timer 2 in cascade connection.
(4) Select the count clock source. TM2MD (x'3F5C') bp2-0 :TM02CK2-0= 001	(4) Set the clock source to prescaler output by the TM2CK2-0 flag of the TM2MD register.
(5) Select the prescaler output and enable counting. CK2MD (x'3F5E') bp2-1 :TM2PSC1-0= 01 bp0 :TM2BAS = 1 PSCMD (x'3F6F') bp0 :PSCEN = 1	(5) Set the prescaler output to fs/4 by the TM2PSC1-0, TM2BAS flag of the timer 2 prescaler selection register (CK2MD). Also, set the PSCEN flag of the prescaler control register (PSCMD) to "1" to enable the prescaler counting.
(6) Set the interrupt generation cycle TMnOC(x'3F5B', x'3F5A')=x'09C3'	(6) Set the timer 3 compare register + timer 2 compare register (TM3OC + TM2OC) to the interrupt generation cycle (x'09C3' : 2500 cycles - 1). At that time, timer 3 binary counter + timer 2 binary counter (TM3BC + TM2BC) are initialized to x'0000'.

Setup Procedure	Description
(7) Disable the lower timer interrupt. TM2ICR (x'3FEB') bp1 :TM2IE = 0	(7) Set the TM2IE flag of the timer 2 interrupt control register (TM2ICR) to "0" to disable the interrupt.
(8) Set the level of the upper timer interrupt. TM3ICR (x'3FEC') bp7-6 :TM3LV1-0 = 10	(8) Set the interrupt level by the TM3LV1-0 flag of the timer 3 interrupt control register (TM3ICR). If any interrupt request flag may be already set, clear all request flags. [☞ Chapter 3 3-1-4. Interrupt Flag Setup]
(9) Enable the upper timer interrupt. TM3ICR (x'3FEC') bp1 :TM3IE = 1	(9) Set the TM3IE flag of the TM3ICR register to "1" to enable the interrupt.
(10) Start the upper timer operation. TM3MD (x'3F5D') bp3 :TM3EN = 1	(10) Set the TM3EN flag of the TM3MD register to "1" to start timer 3.
(11) Start the lower timer operation. TM2MD (x'3F5C') bp3 :TM2EN = 1	(11) Set the TM2EN flag of the TM2MD register to "1" to start timer 2.

TM3BC + TM2BC counts up from x'0000' as a 16-bit timer. When TM3BC + TM2BC reaches the set value of TM3OC + TM2OC register, the timer 3 interrupt request flag is set to "1" at the next count clock, and the value of TM3BC + TM2BC becomes x'0000' and counting up is restarted.

Use a 16-bit access instruction to set the (TM3OC + TM2OC) register.

Start the upper timer operation before the lower timer operation.

6-10 Remote Control Carrier Output

6-10-1 Operation

Carrier pulse for remote control can be generated.

■Operation of Remote Control Carrier Output (Timer 3)

Remote control carrier pulse is based on output signal of timer 3. Duty cycle is selected from 1/2, 1/3 and base timer output. RMOUT (P10) outputs remote control carrier output signal.

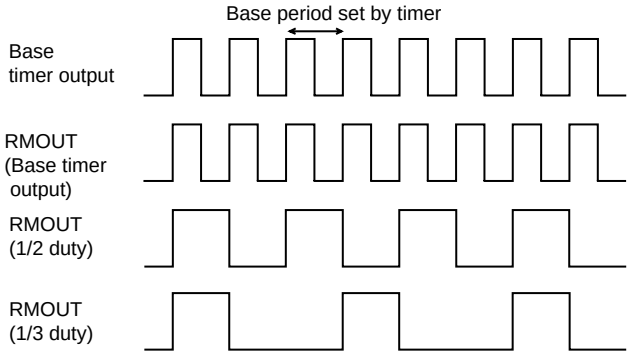


Figure 6-10-1 Duty Cycle of Remote Control Carrier Output Signal

■Count Timing of Remote Control Carrier Output (Timer 3)

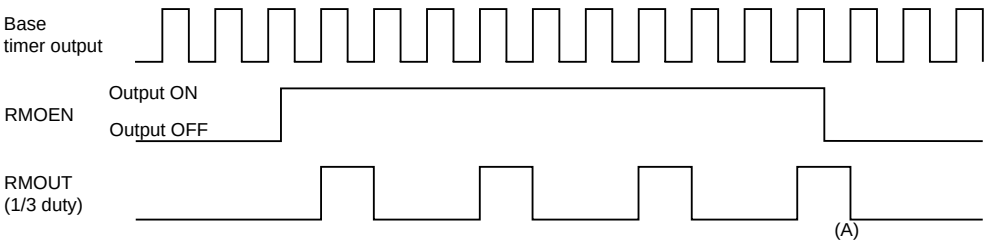


Figure 6-10-2 Count Timing of Remote Control Carrier Output Function (Timer 3)

(A) Even if the RMOEN flag is off when the carrier output is high, the carrier waveform is held by the synchronizing circuit.

If RMETM2 flag of Remote control carrier output control register (RMCTR) is set to "1" during timer 2 operation, RMOEN and RTM7EN flag are set every time timer 2 interrupt request is generated.
Also note that if timer 2 interrupt is enabled, interrupt is accepted every time TM2IRQ is generated.

Before the RMOEN flag is switched to on, set the P1OMD0 flag of the P1OMD register to "1". After it is switched to off, set it to "0".

When the RMOEN flag is changed, do not change the base cycle and its duty at the same time. If they are changed at the same time, the carrier wave form is not output properly.

6-10-2 Setup Example

■Remote Control Carrier Output Setup Example (Timer 3)
Here is the setting example that the RMOUT pin outputs the 1/3 duty carrier pulse signal with "H" period of 36.7 kHz, by using timer 3. The source clock of timer 3 is set to fosc (at 8 MHz).
An example setup procedure, with a description of each step is shown below.

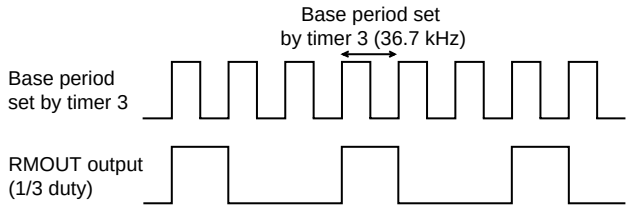


Figure 6-10-3 Output Wave Form of RMOUT Output Pin

Setup Procedure	Description
(1) Disable the remote control carrier output. RMCTR (x'3F6E') bp3 : RMOEN = 0	(1) Set the RMOEN flag of the remote control carrier output control register (RMCTR) to "0" to disable the remote control carrier output.
(2) Select the base cycle setting timer. RMCTR (x'3F6E') bp0 : RMBTMS = 0	(2) Set the RMBTMS flag of the RMCTR register to "0" to set the timer as a base cycle setting timer.
(3) Select the carrier output duty. RMCTR (x'3F6E') bp2,1 : RMDTY1,0= 0,1	(3) Set the RMDTY1,0 flag of the RMCTR register to "0,1" to select 1/3 duty.
(4) Stop the counter. TM3MD (x'3F5D') bp3 : TM3EN = 0	(4) Set the TM3EN flag of the timer 3 mode register (TM3MD) to stop the timer 3 counting.
(5) Set the remote control carrier output of the special function pin. P1OMD (x'3F2F') bp0 : P1OMD0 = 1 P1DIR (x'3F31') bp0 : P1DIR0 = 1 RMCTR (x'3F6E') bp4 :TM7RM = 1	(5) Set the P1OMD0 flag of the port 1 output mode register (P1OMD) to "1" to set P10 pin as a special function pin. Set the P1DIR0 flag of the port 1 direction control register (P1DIR) to "1" for output mode. Set the TM7RM flag of the RMCTR register to "1" to select the remote control carrier output.

Setup Procedure	Description
(6) Select the normal timer operation. TM3MD (x'3F5D') bp4 : TM3PWM = 0 bp5 : TM30MOD = 0	(6) Set both of the TM3MOD flag and TM3PWM flag of the TM3MD register to "0" to select normal timer operation.
(7) Select the count clock source. TM3MD (x'3F5D') bp2-0 : TM3CK2-0 = 000	(7) Select fosc to clock source by the TM3CK2-0 flag of the TM3MD register.
(8) Set the base cycle of remote control carrier. TM3OC (x'3F5B') = x'6C'	(8) Set the base cycle of remote control carrier by writing x'6C' to the timer 3 compare register (TM3OC). The set value should be (8 MHz/ 73.4 kHz) - 1 = 108(x'6C') 8 MHz is divided to be 73.4 kHz, 2 times 36.7 kHz.
(9) Start the timer operation. TM3MD (x'3F5D') bp3 : TM3EN = 1	(9) Set the TM3EN flag of the TM3MD register to "1" to stop the timer 3 counting.
(10) Enable the remote control carrier output. RMCTR (x'3F6E') bp3 : RMOEN = 1	(10) Set the RMOEN flag of the RMCTR register to "1" to enable the remote control carrier output.

TM3BC counts up from x'00'. Timer 3 outputs the base cycle pulse set in TM3OC. Then, the 1/3 duty remote control carrier pulse signal is output. If the RMOEN flag of the RMCTR register is set to "0", the remote control carrier pulse signal output is stopped.

Chapter 7 16-bit Timer

7

7-1 Overview

This LSI contains a general-purpose 16-bit timer (Timer 7). Its compare register is double buffer type. Timer 7 (high function 16-bit timer) has 2 sets of compare registers with double buffering. Also, as an independent interrupt it has a timer 7 interrupt and a timer 7 compare register 2 match interrupt.

7-1-1 Functions

Table 7-1-1 shows the functions of timer 7.

Table 7-1-1 16-bit Timer Functions

	Timer 7 (High precision 16-bit timer)
Interrupt source	TM7IRQ T7OC2IRQ
Timer operation	√
Event count	P14
Timer pulse output	P14, P11
PWM output (duty is changeable)	P14, P11, P10
High precision PWM output (duty and cycle are changeable)	P14, P11, P10
Capture function	√
Pulse width measurement	√
Remote control carrier output	√
Clock source	fosc fosc/2 fosc/4 fosc/16 fs fs/2 fs/4 fs/16 TM7IO input TM7IO input/2 TM7IO input/4 TM7IO input/16
fosc : Machine clock (High speed oscillation) fs : System clock [☞ Chapter 2 2-5 Clock Switching]	

7-1-2 Block Diagram

■Timer 7 Block Diagram

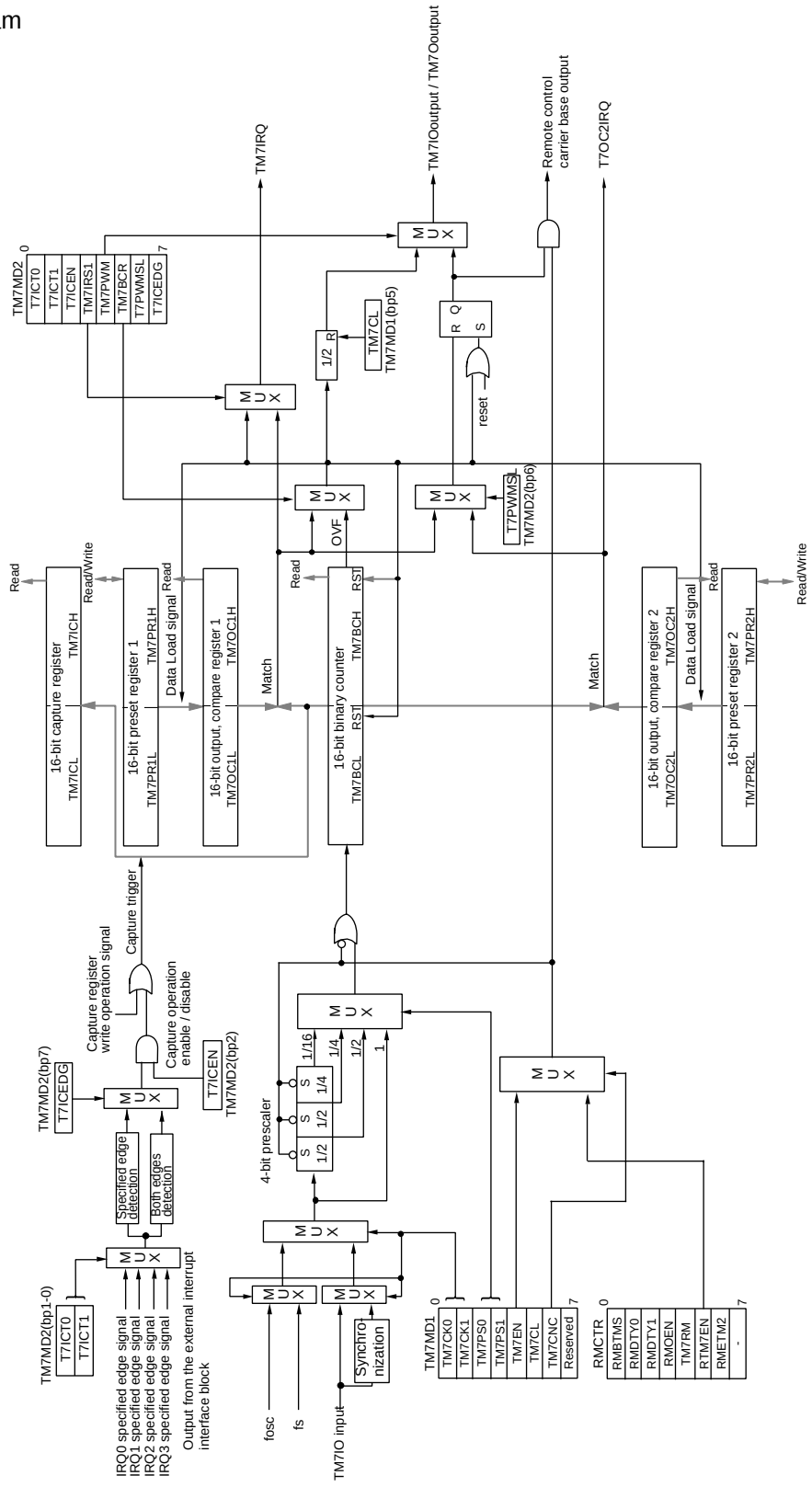


Figure 7-1-1 Timer 7 Block Diagram

■Remote Control Carrier Output Block Diagram

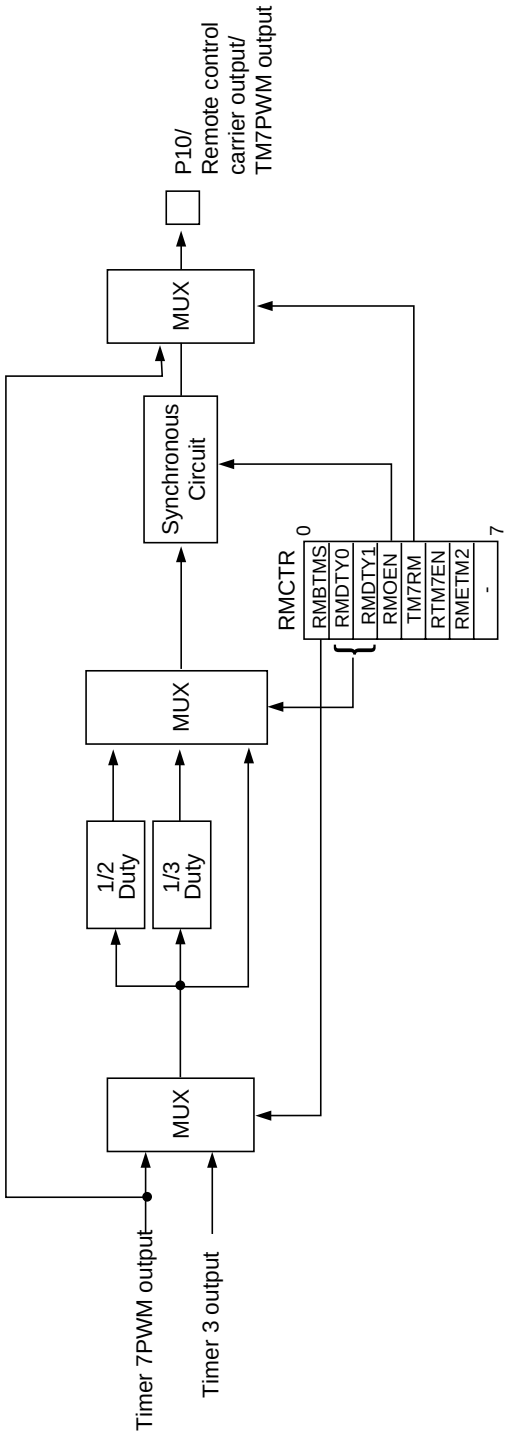


Figure 7-1-2 Remote Control Carrier Output Block Diagram

7-2 Control Registers

Timer 7 contains the binary counter (TM7BC), the compare register 1 (TM7OC1), and its double buffer preset register (TM7PR1), the compare register 2 (TM7OC2) and its double buffer preset register 2 (TM7PR2), the capture register (TM7IC). The mode register 1 (TM7MD1) and the mode register 2 (TM7MD2) controls timer 7. Remote control carrier output control register (RMCTR) controls remote control function.

7-2-1 Registers

Table 7-2-1 shows the registers that control timer 7.

Table 7-2-1 16-bit Timer Control Registers

	Register	Address	R/W	Function	Page
Timer 7	TM7BCL	x'03F70'	R	Timer 7 binary counter (lower 8 bits)	VII - 8
	TM7BCH	x'03F71'	R	Timer 7 binary counter (upper 8 bits)	VII - 8
	TM7OC1L	x'03F72'	R	Timer 7 compare register 1 (lower 8 bits)	VII - 6
	TM7OC1H	x'03F73'	R	Timer 7 compare register 1 (upper 8 bits)	VII - 6
	TM7PR1L	x'03F74'	R/W	Timer 7 preset register 1 (lower 8 bits)	VII - 7
	TM7PR1H	x'03F75'	R/W	Timer 7 preset register 1 (upper 8 bits)	VII - 7
	TM7ICL	x'03F76'	R	Timer 7 capture regsiter (lower 8 bits)	VII - 8
	TM7ICH	x'03F77'	R	Timer 7 capture register (upper 8 bits)	VII - 8
	TM7MD1	x'03F78'	R/W	Timer 7 mode register 1	VII - 9
	TM7MD2	x'03F79'	R/W	Timer 7 mode register 2	VII - 10
	TM7OC2L	x'03F7A'	R	Timer 7 compare register 2 (lower 8 bits)	VII - 6
	TM7OC2H	x'03F7B'	R	Timer 7 compare register 2 (upper 8 bits)	VII - 6
	TM7PR2L	x'03F7C'	R/W	Timer 7 preset register 2 (lower 8 bits)	VII - 7
	TM7PR2H	x'03F7D'	R/W	Timer 7 preset register 2 (upper 8 bits)	VII - 7
	TM7ICR	x'03FF1'	R/W	Timer 7 interrupt control register	III - 25
	T7OC2ICR	x'03FF2'	R/W	Timer 7 compare register 2 match interrupt control register	III - 26
	P1OMD	x'03F2F'	R/W	Port 1 output mode register	IV - 13
	P1DIR	x'03F31'	R/W	Port 1 direction control register	IV - 12
Remote control	RMCTR	x'03F6E'	R/W	Remote control carrier output control register	VII - 11

R/W : Readable/Writable

R : Readable only

7-2-2 Programmable Timer Registers

Timer 7 has a 16-bit programmable timer register. It contains a compare register, a preset register, a binary counter and a capture register. Each register has 2 sets of 8-bit register. Operate by 16-bit access.

Compare register is a 16-bit register stores the value that compared to binary counter. The compared value that written to the preset register in advance is loaded.

■Timer 7 Compare Register 1 (TM7OC1)

	7	6	5	4	3	2	1	0	
TM7OC1L	TM7OC1L7	TM7OC1L6	TM7OC1L5	TM7OC1L4	TM7OC1L3	TM7OC1L2	TM7OC1L1	TM7OC1L0	(At reset : X X X X X X X X)

Figure 7-2-1 Timer 7 Compare Register 1 Lower 8 bits (TM7OC1L : x'03F72', R)

	7	6	5	4	3	2	1	0	
TM7OC1H	TM7OC1H7	TM7OC1H6	TM7OC1H5	TM7OC1H4	TM7OC1H3	TM7OC1H2	TM7OC1H1	TM7OC1H0	(At reset : X X X X X X X X)

Figure 7-2-2 Timer 7 Compare Register 1 Upper 8 bits (TM7OC1H : x'03F73', R)

■Timer 7 Compare Register 2 (TM7OC2)

	7	6	5	4	3	2	1	0	
TM7OC2L	TM7OC2L7	TM7OC2L6	TM7OC2L5	TM7OC2L4	TM7OC2L3	TM7OC2L2	TM7OC2L1	TM7OC2L0	(At reset : X X X X X X X X)

Figure 7-2-3 Timer 7 Compare Register 2 Lower 8 bits (TM7OC2L : x'03F7A', R)

	7	6	5	4	3	2	1	0	
TM7OC2H	TM7OC2H7	TM7OC2H6	TM7OC2H5	TM7OC2H4	TM7OC2H3	TM7OC2H2	TM7OC2H1	TM7OC2H0	(At reset : X X X X X X X X)

Figure 7-2-4 Timer 7 Compare Register 2 Upper 8 bits (TM7OC2H : x'03F7B', R)

The timer 7 preset register 1 and 2 are buffer registers of the timer 7 compare register 1 and 2. If the set value is written to the timer 7 preset register 1 and 2 when the counting is stopped, the same set value is loaded to the timer 7 compare register 1 and 2. If the set value is written to the timer 7 preset register 1 and 2 when the counting is operated, the set value of the timer 7 preset register 1 and 2 is loaded to the timer 7 compare register 1 and 2 at the timing that the timer 7 binary counter is cleared.

■Timer 7 Preset Register 1 (TM7PR1)

	7	6	5	4	3	2	1	0	
TM7PR1L	TM7PR1L7	TM7PR1L6	TM7PR1L5	TM7PR1L4	TM7PR1L3	TM7PR1L2	TM7PR1L1	TM7PR1L0	(At reset : X X X X X X X X)

Figure 7-2-5 Timer 7 Preset Register 1 Lower 8 bits (TM7PR1L : x'03F74', R/W)

	7	6	5	4	3	2	1	0	
TM7PR1H	TM7PR1H7	TM7PR1H6	TM7PR1H5	TM7PR1H4	TM7PR1H3	TM7PR1H2	TM7PR1H1	TM7PR1H0	(At reset : X X X X X X X X)

Figure 7-2-6 Timer 7 Preset Register 1 Upper 8 bits (TM7PR1H : x'03F75', R/W)

■Timer 7 Preset Register 2 (TM7PR2)

	7	6	5	4	3	2	1	0	
TM7PR2L	TM7PR2L7	TM7PR2L6	TM7PR2L5	TM7PR2L4	TM7PR2L3	TM7PR2L2	TM7PR2L1	TM7PR2L0	(At reset : X X X X X X X X)

Figure 7-2-7 Timer 7 Preset Register 2 Lower 8 bits (TM7PR2L : x'03F7C', R/W)

	7	6	5	4	3	2	1	0	
TM7PR2H	TM7PR2H7	TM7PR2H6	TM7PR2H5	TM7PR2H4	TM7PR2H3	TM7PR2H2	TM7PR2H1	TM7PR2H0	(At reset : X X X X X X X X)

Figure 7-2-8 Timer 7 Preset Register 2 Upper 8 bits (TM7PR2H : x'03F7D', R/W)

Binary counter is a 16-bit up counter. If any data is written to a preset register when the counting is stopped, the binary counter is cleared to x'0000'.

■Timer 7 Binary Counter (TM7BC)

	7	6	5	4	3	2	1	0	
TM7BCL	TM7BCL7	TM7BCL6	TM7BCL5	TM7BCL4	TM7BCL3	TM7BCL2	TM7BCL1	TM7BCL0	(At reset : X X X X X X X X)

Figure 7-2-9 Timer 7 Binary Counter Lower 8 bits (TM7BCL : x'03F70', R)

	7	6	5	4	3	2	1	0	
TM7BCH	TM7BCH7	TM7BCH6	TM7BCH5	TM7BCH4	TM7BCH3	TM7BCH2	TM7BCH1	TM7BCH0	(At reset : X X X X X X X X)

Figure 7-2-10 Timer 7 Binary Counter Upper 8 bits (TM7BCH : x'03F71', R)

Input capture register is a register that holds the value loaded from a binary counter by capture trigger. Capture trigger is generated by an input signal from an external interrupt pin, and when an arbitrary value is written to an input capture register (Directly writing to the register by program is disable.).

■Timer 7 Input Capture Register (TM7IC)

	7	6	5	4	3	2	1	0	
TM7ICL	TM7ICL7	TM7ICL6	TM7ICL5	TM7ICL4	TM7ICL3	TM7ICL2	TM7ICL1	TM7ICL0	(At reset : X X X X X X X X)

Figure 7-2-11 Timer 7 Input Capture Register Lower 8 bits (TM7ICL : x'03F76', R)

	7	6	5	4	3	2	1	0	
TM7ICH	TM7ICH7	TM7ICH6	TM7ICH5	TM7ICH4	TM7ICH3	TM7ICH2	TM7ICH1	TM7ICH0	(At reset : X X X X X X X X)

Figure 7-2-12 Timer 7 Input Capture Register Upper 8 bits (TM7ICH : x'03F77', R)

7-2-3 Timer Mode Registers

This is a readable / writable register that controls timer 7.

■Timer 7 Mode Register 1 (TM7MD1)

	7	6	5	4	3	2	1	0
TM7MD1	RESERVED	TMCNC	TM7CL	TM7EN	TM7PS1	TM7PS0	TM7CK1	TM7CK0

(At reset : 0 0 1 0 0 0 0 0)

TM7CK1	TM7CK0	Clock source selection
0	0	fosc
	1	fs
1	0	TM7IO input
	1	Synchronous TM7IO input

TM7PS1	TM7PS0	Count clock selection
0	0	1/1 of clock
	1	1/2 of clock
1	0	1/4 of clock
	1	1/16 of clock

TM7EN	Timer 7 count control
0	Disable the count
1	Enable the count

TM7CL	Timer output reset control
0	Operate timer output
1	Disable timer output (reset)

TM7CNC	Timer 7 count control sewitching
0	Operate timer output
1	Disable timer output (reset)

RESERVED	Set always "0".

Figure 7-2-13 Timer 7 Mode Register 1 (TM7MD1 : x'03F78', R/W)

■Timer 7 Mode Register 2 (TM7MD2)

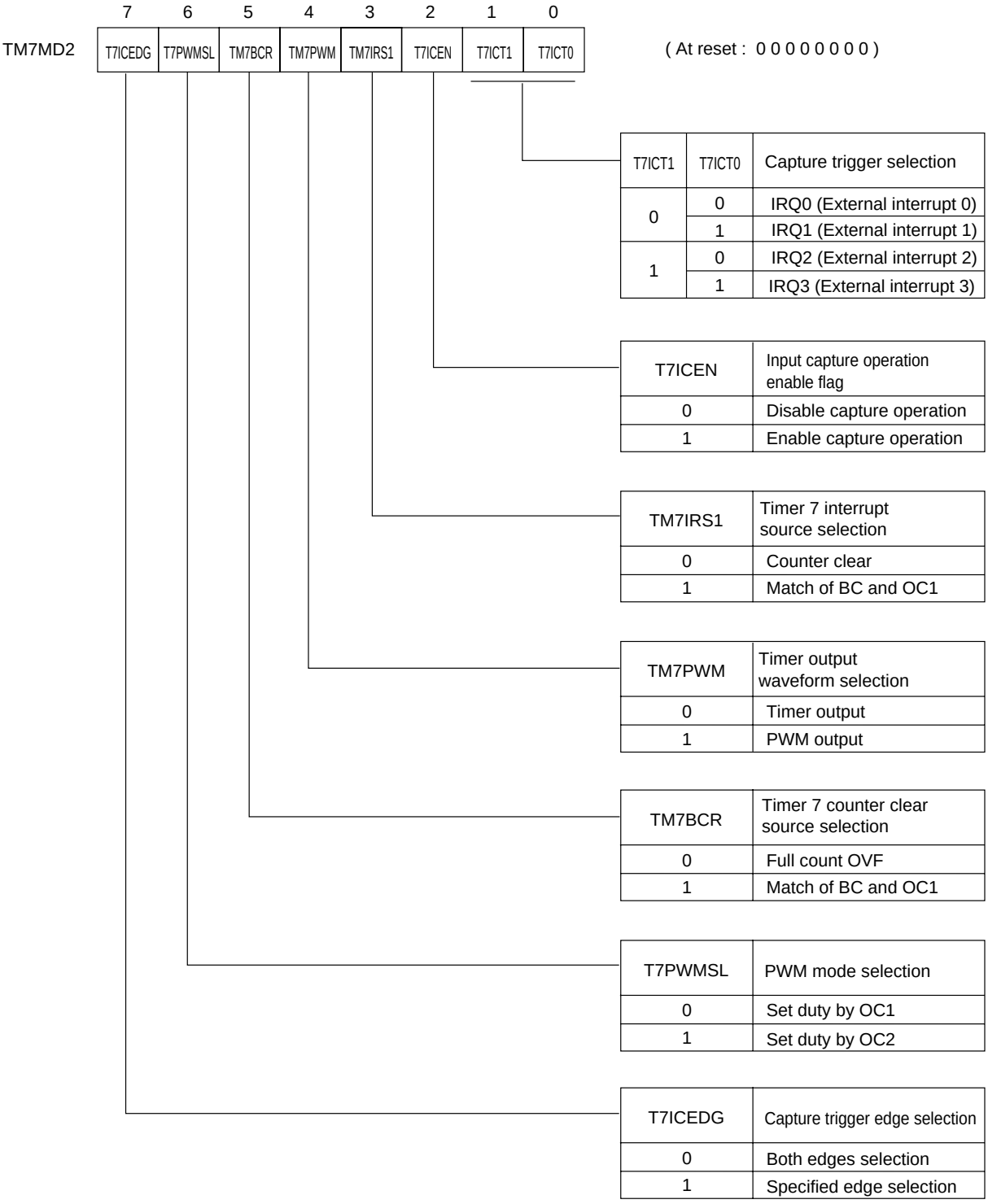
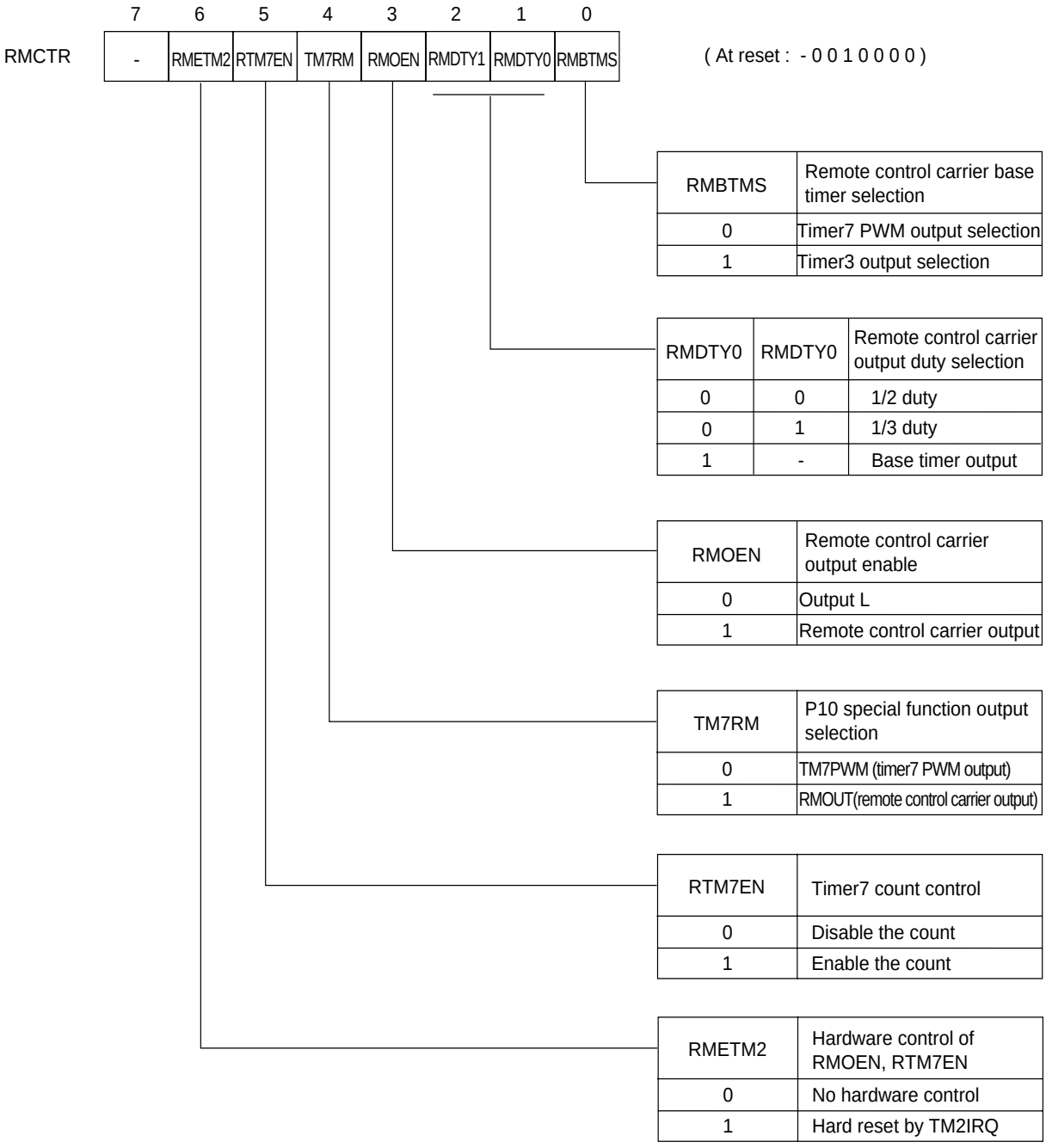


Figure 7-2-14 Timer 7 Mode Register 2 (TM7MD2 : x'03F79', R/W)

■Remote Control Carrier Output Control Register (RMCTR)



RMBTMS	Remote control carrier base timer selection
0	Timer7 PWM output selection
1	Timer3 output selection

RMDTY0	RMDTY0	Remote control carrier output duty selection
0	0	1/2 duty
0	1	1/3 duty
1	-	Base timer output

RMOEN	Remote control carrier output enable
0	Output L
1	Remote control carrier output

TM7RM	P10 special function output selection
0	TM7PWM (timer7 PWM output)
1	RMOUT(remote control carrier output)

RTM7EN	Timer7 count control
0	Disable the count
1	Enable the count

RMETM2	Hardware control of RMOEN, RTM7EN
0	No hardware control
1	Hard reset by TM2IRQ

Figure 7-2-15 Remote Control Carrier Output Control Register
(RMCTR : x'03F6E', R/W)

If RMETM2 flag of Remote control carrier output control register (RMCTR) is set to "1" during timer 2 operation, RMOEN and RTM7EN flag are set every time timer 2 interrupt request is generated.
Also note that if timer 2 interrupt is enabled, interrupt is accepted every time TM2IRQ is generated.

7-3 16-bit Timer Count

7-3-1 Operation

The timer operation can constantly generate interrupts.

■16-bit Timer Operation (Timer 7)
The generation cycle of an timer interrupt is set by the clock source selection and the set value of the compare register 1 (TM7OC1), in advance. When the binary counter (TM7BC) reaches the set value of the compare register 1, the timer 7 interrupt request is generated at the next count clock. There are 2 sources ; the TM7OC1 compare match or the full count over flow, to be selected to clear the binary counter. After the binary counter is cleared to x'0000, the counting up is restarted from x'0000'.

Table 7-3-1 16-bit Timer Interrupt Source and Binary Counter Clear Source (Timer 7)

TM7MD2 register		Interrupt source	Binary counter clear source
TM7IRS1 flag	TM7BCR flag		
1	1	TM7OC1 compare match	TM7OC1 compare match
0	1	TM7OC1 compare match	TM7OC1 compare match
1	0	TM7OC1 compare match	full count over flow
0	0	full count over flow	full count over flow

Timer 7 can generate another set of an independent interrupt (Timer 7 compare register 2 match interrupt) by the set value of the timer 7 compare register (TM7OC2). At that timer, the binary counter is cleared as the above setup.

The compare register is double buffer type. So, when the value of the preset register is changed during the counting, the changed value is stored to the compare register as the binary counter is cleared. This function can change its value of the compare register constantly, without disturbing the cycle during timer operation (Reload function).

When the CPU reads the 16-bit binary counter (TM7BC), the read data is treated as 8-bits unit data even if it is a 16-bit MOVW instruction. As a result, it will read the data incorrectly if a carry from the lower 8 bits to the upper 8 bits occurs during counting.

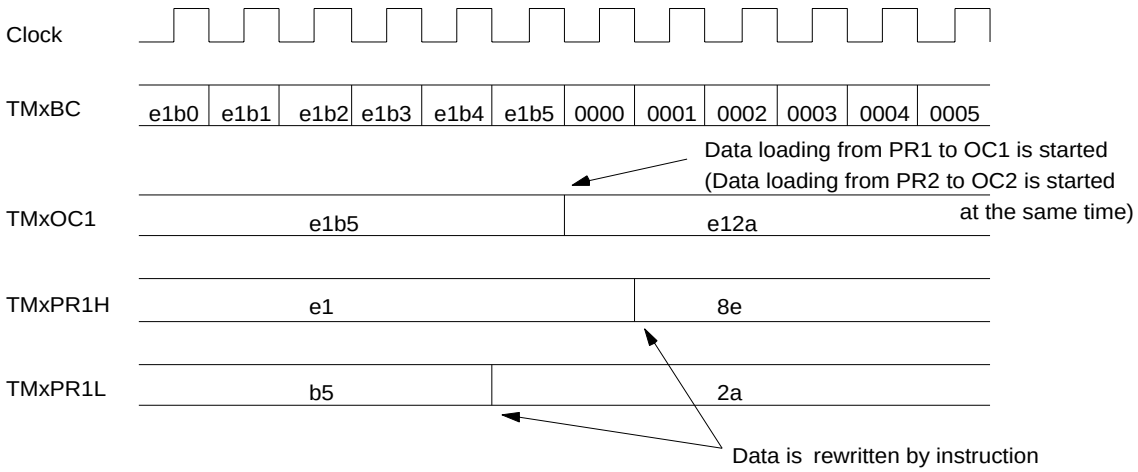
To read the correct value of the 16-bit counting (TM7BC), use the writing program function to the input capture register (TM7IC). By writing to the TM7IC, the counting data of TM7BC can be stored to TM7IC to read out the correct counting data during operation.

[☞ Chapter 7-9-1. Operation (p.VII-33)]

When a data is written to 16-bit timer preset register (TM7PR1, TM7PR2), it is recognized as a 8-bit unit data inside LSI even if it is a 16-bit access MOVW instruction. After lower 8 bits of preset register is written, if data loading from preset register to compare register is started before the upper 8 bits is written, data which is not rewritten is loaded to the upper 8 bits and rewritten data is loaded to the lower 8 bits.
Therefore, writing data to the preset register (TM7PR1, TM7PR2) need to be completed before data loading from the preset register to the compare register is started.

Shown below is timing chart of TM7PR1 and TM7OC1 data rewriting and the data loading. When data is written to TM7PR2 wrong data could be loaded due to the same problem.

TM7BC and TM7OC1 compare match and load timing of TM7PR1



Data e12a is loaded to OC1 as PR1rewrting (e1b5 -> 8e2a) and loading to OC1 are operated at the same time.

Table 7-3-2 shows the clock source that can be selected.

Table 7-3-2 Clock Source at Timer Operation(Timer 7)

Clock source	1 count time
fosc	50 ns
fosc/2	100 ns
fosc/4	200 ns
fosc/16	800 ns
fs	100 ns
fs/2	200 ns
fs/4	400 ns
fs/16	1.6 µs
as fosc = 20 MHz, fs = fosc/2 = 10 MHz	

■Count Timing of Timer Operation (Timer 7)

The binary counter counts up with the selected clock source as the count clock.

The basic operation of the whole function of 16-bit timer is as follows ;

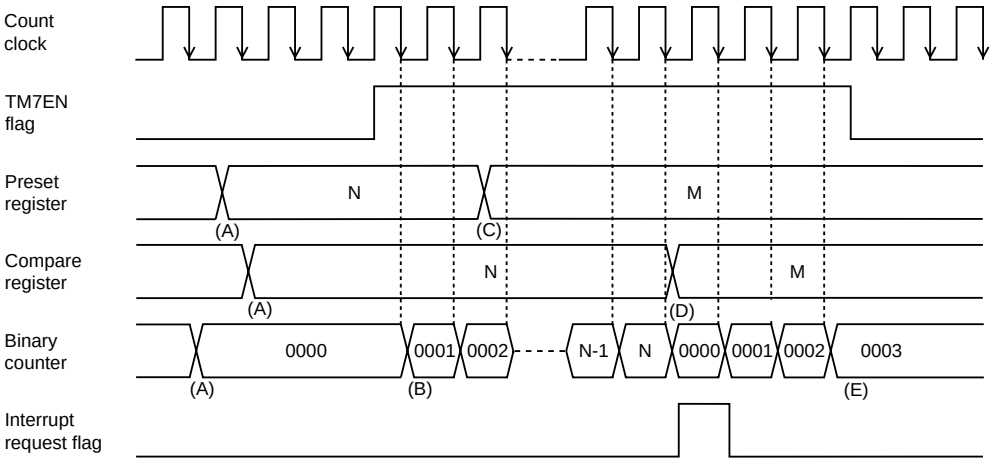


Figure 7-3-1 Count Timing of Timer Operation (Timer 7)

(A) When any data is written to the preset register as the TM7EN flag is stopped ("0"), the same value is loaded at the writing cycle and the binary counter is cleared to x'0000'.

(B) If the TM7EN flag is "1", the binary counter starts counting. The counting is happened at the falling edge of the count clock.

(C) Even if the preset register is rewritten as the TM7EN flag is "1", the binary counter is not changed.

(D) If the binary counter reaches the value of the compare register 1, the set value of the preset register is loaded to the compare register at the next count clock. And the interrupt request flag is set at the next count clock, and the binary counter is cleared to x'0000' to restart counting up.

(E) If the TM7EN flag is"0", the binary counter is stopped.

When the binary counter reaches the value of the compare register, the interrupt request flag is set and the binary counter is cleared, at the next count clock. So, set the compare register as; (the set value of the compare register) = (count till the interrupt request - 1)

When the timer 7 compare register 2 match interrupt is generated and the TM7OC1 compare match is selected as a binary counter clear source, the set value of the compare register 2 should be smaller than the set value of the compare register 1.

If the interrupt is enabled, the timer interrupt request flag should be cleared before timer operation is started.

At TM7OC=x'0000', x'0001', the timer n interrupt request generation has the same waveform.

When more than 2 waits is set at access to the special register area by the IOW1, IOW0 flag of the MEMCTR register, write the same value 2 times at setup of the preset register as the timer is stopped. When 1 wait or no wait is set, there is no need to do this.
(This is for all functions of a 16-bit timer.) [☞ Chapter 2 2-3-2. Control Registers]

7-3-2 Setup Example

■Timer Operation Setup Example (Timer 7)

Timer 7 generates an interrupt constantly for timer function. Fosc/2 (fosc=20 MHz) is selected as a clock source to generate an interrupt every 1000 cycles (100 µs).

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter. TM7MD1 (x'3F78') bp4 : TM7EN = 0	(1) Set the TM7EN flag of the timer 7 mode register 1 (TM7MD1) to "0" to stop timer 7 counting.
(2) Select the timer clear source. TM7MD2 (x'3F79') bp5 : TM7BCR = 1	(2) Set the TM7BCR flag of the timer 7 mode register 2 (TM7MD2) to "1" to select the compare match as a binary counter clear source.
(3) Select the count clock source. TM7MD1 (x'3F78') bp1-0 : TM7CK1-0 = 00 bp3-2 : TM7PS1-0 = 01	(3) Select fosc as a clock source by the TM7CK1-0 flag of the TM7MD1 register. Also select 1/2 fosc as a count clock source by TM7PS1-0 flag.
(4) Set the count control flag TM7MD1 (x'3F78') bp6 : TM7CNC = 0	(4) Select the TM7EN flag to the timer 7 count control with the TM7CNC flag of the TM7MD1 register.
(5) Set the interrupt generation cycle TM7PR1 (x'3F75', x'3F74')=x'03E7	(5) Set the interrupt generation cycle to the timer 7 preset register 1 (TM7PR1). The cycle is 1000. The set value should be 1000-1=999(x'03E7'). At that time, the same value is loaded to the timer 7 compare register 1 (TM7OC1), and the timer 7 binary counter (TM7BC) is initialized to x'0000'.
(6) Set the interrupt level. TM7ICR (x'3FF1') bp7-6 : TM7LV1-0 = 10	(6) Set the interrupt level by the TM7LV1-0 flag of the timer 7 interrupt control register (TM7ICR). If the interrupt request flag may be already set, clear the request flag. [☞ Chapter 3 3-1-4. Interrupt Flag Setup]
(7) Enable the interrupt. TM7ICR (x'3FF1') bp1 : TM7IE = 1	(7) Set the TM7IE flag of the TM7ICR register to "1" to enable the interrupt.

Setup Procedure	Description
(8) Start the timer operation. TM7MD1 (x'3F78') bp4 : TM7EN = 1	(8) Set the TM7EN flag of the TM7MD1 register to "1" to start timer 7.

TM7BC counts up from x'0000'. When TM7BC reaches the set value of the TM7OC1 register, the timer 7 interrupt request flag is set to "1" at the next count clock and the TM7BC becomes x'0000' and counts up, again.

When the TM7EN flag of the TM7MD register is changed at the same time to other bits, the binary counter may count up by the switching operation.

7-4 16-bit Event Count

7-4-1 Operation

Event count operation has 2 types ; TM7IO input and synchronous TM7IO input can be selected as the count clock. Each type can select 1/1, 1/2, 1/4 or 1/6 as a count clock source.

■16-bit Event Count Operation (Timer 7)
Event count means that the binary counter (TM7BC) counts the input signal from external to the TM7IO pin. If the value of the binary counter reaches the setting value of the compare register (TM7OC), interrupts can be generated at the next count clock.

Table 7-4-1 Event Count Input Clock

	Timer 7
Event input	TM7IO input (P14)
	Synchronous TM7IO input

As an actual count clock, a signal divided 1, 2, 4, or 16 is selected.

■Count Timing of TM7IO Input (Timer 7)
When TM7IO input is selected, TM7IO input signal is directly input to the count clock of the timer 7. The binary counter counts up at the falling edge of the TM7IO input signal or at the falling edge of the TM7IO input signal that passed the divider.

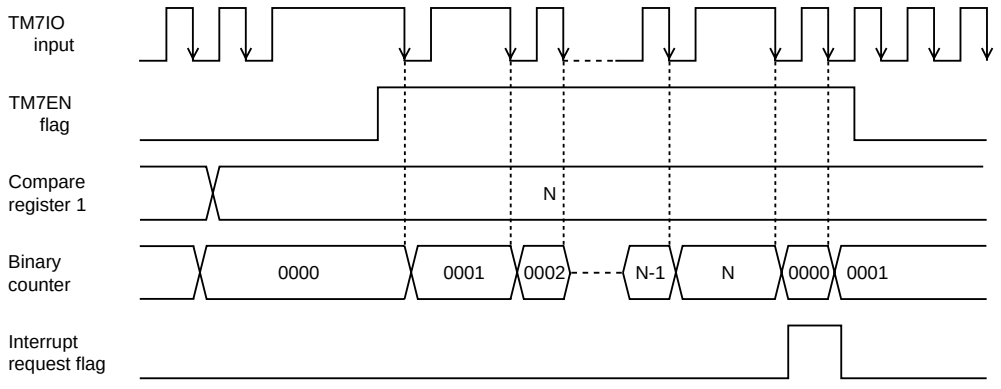


Figure 7-4-1 Count Timing TM7IO Input (Timer 7)

If the binary counter is read out at operation, incorrect data at counting up may be read. To prevent this, use the event count by the synchronous TM7IO input as the following page.

■Count Timing of Synchronous TM7IO Input (Timer 7)

If the synchronous TM7IO input is selected, the synchronizing circuit output signal is input to the count clock. The synchronizing circuit output signal is changed at the falling edge of the system clock after the TM7IO input signal is changed. The binary counter counts up at the falling edge of the synchronizing circuit output signal or the synchronizing circuit output signal that passed through the divide-by circuit.

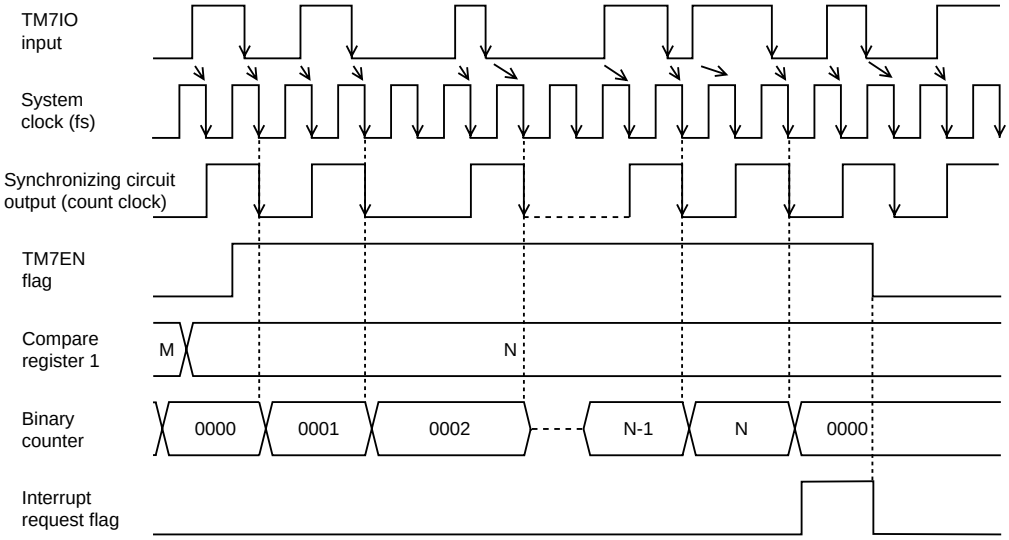


Figure 7-4-2 Count Timing of Synchronous TM7IO Input (Timer 7)

When the synchronous TM7IO input is selected as the count clok source, the timer 7 counter counts up in synchronization with the system clock. Therefore, the correct value is always read. But, if the synchronous TM7IO is selected as the count clock source, CPU mode cannot return from STOP/HALT mode.

7-4-2 Setup Example

■Event Count Setup Example (Timer 7)

If the falling edge of the TM7IO input pin signal is detected 5 times with using timer 7, an interrupt is generated. An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter. TM7MD1 (x'3F78') bp4 : TM7EN = 0	(1) Set the TM7EN flag of the timer 7 mode register 1 (TM7MD1) to "0" to stop timer 7 counting.
(2) Set the special function pin to input mode. P1DIR (x'3F31') bp4 : P1DIR4 = 0	(2) Set the P1DIR4 flag of the port 1 direction control register (P1DIR) to "0" to set P14 pin to input mode. If it needs, pull-up resistor should be added. [☞ Chapter 4 I/O Ports]
(3) Select the condition for timer clear. TM7MD2 (x'3F79') bp5 : TM7BCR = 1	(3) Set the TM7BCR flag of the timer 7 mode register 2 (TM7MD2) to "1" to select the compare match as a clear source of binary counter.
(4) Select the count clock source. TM7MD1 (x'3F78') bp1-0 : TM7CK1-0 = 10 bp3-2 : TM7PS1-0 = 00	(4) Select the TM7IO input as a clock source by the TM7CK1-0 flag of the TM7MD1 register. Also, select 1/1(no division) as a count clock source by the TM7PS1-0 flag.
(5) Set the count control flag TM7MD1 (x'3F78') bp6 : TM7CNC = 0	(5) Select the TM7EN flag to the timer 7 count control with the TM7CNC flag of the TM7MD1 register.
(6) Set the interrupt generation cycle. TM7PR1 (x'3F75', x'3F74')=x'0004'	(6) Set the interrupt generation cycle to the timer 7 preset register 1 (TM7PR1). The set value should be 4, because the counting is 5 times. At that time, the same value is loaded to the timer 7 compare register 1 (TM7OC1), and the timer 7 binary counter (TM7BC) is initialized to x'0000'.

Setup Procedure	Description
(7) Set the interrupt level. TM7ICR (x'3FF1') bp7-6 :TM7LV1-0 = 10	(7) Set the interrupt level by the TM7LV1-0 flag of the timer 7 interrupt control register (TM7ICR). If any interrupt request flag may be already set, clear those request flags. [☞ Chapter 3 3-1-4. Interrupt Flag Setup]
(8) Enable the interrupt. TM7ICR (x'3FF1') bp1 : TM7IE = 1	(8) Set the TM7IE flag of the TM7ICR register to "1" to enable interrupt.
(9) Start the event count. TM7MD1 (x'3F78') bp4 : TM7EN = 1	(9) Set the TM7EN flag of the TM7MD 1 register to "1" to start timer 7.

Every time TM7BC detects the falling edge of the TM7IO input, TM7BC counts up from x'0000'. When the TM7BC reaches the setting value of the TM7OC1 register, the timer 7 interrupt request flag is set at the next count clock, then the value of TM7BC becomes x'0000' and counting up is restarted.

7-5 16-bit Timer Pulse Output

7-5-1 Operation

TM7IO pin can output a pulse signal with an arbitrary frequency.

■16-bit Timer Pulse Output Operation (Timer 7)
The timers can output 2 x cycle signal, compared to the setting value to the compare register 1 (TM7OC1) or 1/2 the frequency of the 16-bit full count.
Output pin are as follows.

Table 7-5-1 Timer Pulse Output Pin

	Timer 7	
Pulse output pin	TM7IO output (P14)	TM7O output (P11)

Table 7-5-2 shows the timer interrupt generation sources and the flags that control the timer pulse output cycle.

Table 7-5-2 16-bit Timer Interrupt Generation Source and Timer Pulse Output Cycle (Timer 7)

TM7MD2 register		Interrupt source	Timer pulse output cycle
TM7IRS1 flag	TM7BCR flag		
1	1	TM7OC1 compare match	set value of TM7OC1 x 2
0	1	TM7OC1 compare match	set value of TM7OC1 x 2
1	0	TM7OC1 compare match	full count of TM7BC x 2
0	0	full count over flow	full count of TM7BC x 2

■Count Timing of Timer Pulse Output (Timer 7)

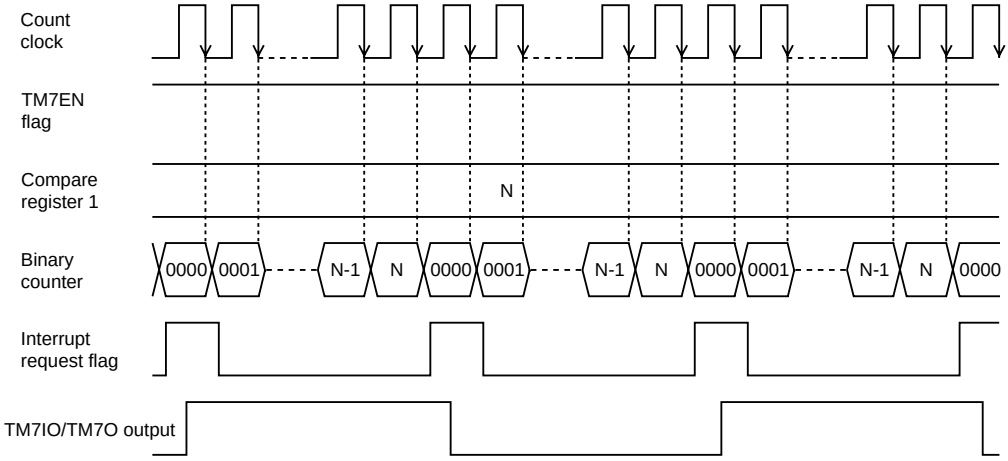


Figure 7-5-1 Count Timing of Timer Pulse Output (Timer 7)

The TM7IO/TM7O pin outputs 2 x cycle, compared to the value in the compare register 1. If the binary counter reaches the compare register, and the binary counter is cleared to x'0000' or the full count overflow, the TM7IO/TM7O output (timer output) is inverted. The inversion of the timer output is changed at the rising edge of the count clock. This is happened to form the waveform inside to correct the output cycle.

In the initial state after releasing reset, the timer pulse output is reset, and low output is fixed. Therefore, release the reset of the timer pulse output by setting the TM7CL flag of the TM7MD1 register to "0".

7-5-2 Setup Example

■Timer Pulse Output Setup Example (Timer 7)
TM7IO pin (P14) outputs 50 kHz pulse by using timer 7. For this, select fosc as clock source, and set a 1/2 cycle (100 kHz) for the timer 7 compare register (at fosc=20 MHz).
An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter. TM7MD1 (x'3F78') bp4 : TM7EN = 0	(1) Set the TM7EN flag of the timer 7 mode register 1 (TM7MD1) to "0" to stop timer 7 counting.
(2) Set the special function pin to output mode. P1OMD (x'3F2F') bp4 : P1OMD4 = 1 P1DIR (x'3F31') bp4 : P1DIR4 = 1	(2) Set the P1OMD4 flag of the port 1 output mode register (P1OMD) to "1" to set P14 pin as the special function pin. Set the P1DIR4 flag of the port 1 direction control register (P1DIR) to "1" to set output mode. If it needs, pull-up resistor should be added. [☞ Chapter 4 I/O Ports]
(3) Set the timer pulse output. TM7MD2 (x'3F79') bp4 : TM7PWM = 0	(3) Set the TM7PWM flag of the timer 7 mode register 2 (TM7MD2) to "0" to select the timer pulse output.
(4) Select the condition for timer clear. TM7MD2 (x'3F79') bp5 : TM7BCR = 1	(4) Set the TM7BCR flag of the TM7MD2 register to "1" to select the compare match as a clear source of a binary counter .
(5) Select the count clock source. TM7MD1 (x'3F78') bp1-0 : TM7CK1-0 = 00 bp3-2 : TM7PS1-0 = 00	(5) Select fosc as an clock source by the TM7CK1-0 flag of the TM7MD1 register. Also, select 1/1 frequency as an count clock source by the TM7PS1-0 flag.
(6) Set the count control flag TM7MD1 (x'3F78') bp6 : TM7CNC = 0	(6) Select the TM7EN flag to the timer 7 count control with the TM7CNC flag of the TM7MD1 register.

Setup Procedure	Description
(7) Set the timer pulse output cycle. TM7PR1 (X'3F75', X'3F74')=x'00C7'	(7) Set the 1/2 frequency of the timer pulse output cycle to the timer 7 preset register 1 (TM7PR1). To obtain 100 kHz by dividing 20 MHz, set as follows ; 200 - 1 = 199 (x'C7') At that time, the same value is loaded to the timer 7 compare register 1 (TM7OC1) and the timer 7 binary counter (TM7BC) is initialized to x'0000'.
(8) Release the reset of the timer pulse output. TM7MD1 (x'3F78') bp5 : TM7CL = 0	(8) Set the TN7CL flag of the TM7MD 1 register to "0" to enable the timer pulse output.
(9) Start the timer operation. TM7MD1 (x'3F78') bp4 : TM7EN = 1	(9) Set the TM7EN flag of the TM7MD1 register to "1" to start timer 7.

TM7BC counts up from x'0000'. If TM7BC reaches the set value of the TM7OC1 register and TM7BC is cleared to x'0000', the signal of the TM7IO output is inverted and TM7BC counts up from x'0000', again.

At TM7OC1 = x'0000' and x'0001', the timer pulse output has the same waveform.

Either binary counter stops or operates, the timer output is "L", when the TM7CL flag of the TM7MD2 register is set to "1".

Set the compare register value as follows.

$$\text{The compare register value} = \frac{\text{The timer pulse output cycle}}{\text{The count clock cycle x 2}} - 1$$

7-6 16-bit Standard PWM Output

(Only duty can be changed consecutively)

The TM7IO/TM7O/TM7PWM pins outputs the standard PWM output, which is determined by the over flow timing of the binary counter, and the match timing of the timer binary counter and the compare register.

7-6-1 Operation

■16-bit Standard PWM Output (Timer 7)

PWM waveform with an arbitrary duty is generated by setting a duty of PWM "H" period to the compare register 1 (TM7OC1). Its cycle is the time of the 16-bit timer full count over flow.

Table 7-6-1 shows the PWM output pin.

Table 7-6-1 PWM Output Pin

	Timer 7		
PWM output pin	TM7IO output pin (P14)	TM7O output pin (P11)	TM7PWM output pin (P10)

■Count Timing of Standard PWM Output (at Normal)(Timer 7)

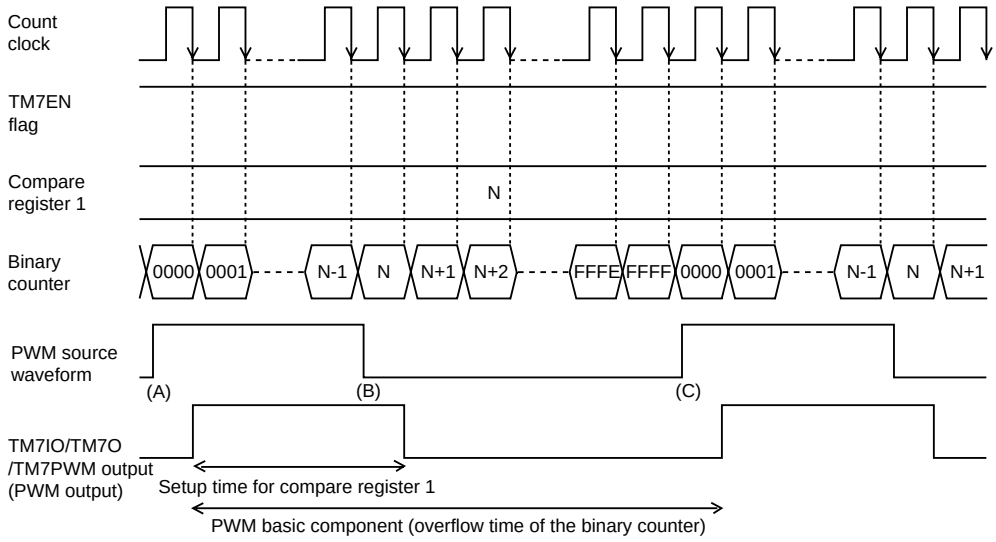


Figure 7-6-1 Count Timing of Standard PWM Output (at Normal)

PWM source waveform,

(A) shows "H" till the binary counter reaches the compare register from x'0000'.

(B) shows "L" after the compare match, then the binary counter counts up till the over flow.

(C) shows "H", again if the binary counter becomes overflow.

The PWM output form pins is 1 count clock delay of PWM source waveform. This is happened to correct the output cycle.

■Count Timing of Standard PWM Output (when Compare Register 1 is x'0000')(Timer 7)
Here is the count timing at setting x'0000' to the compare register 1.

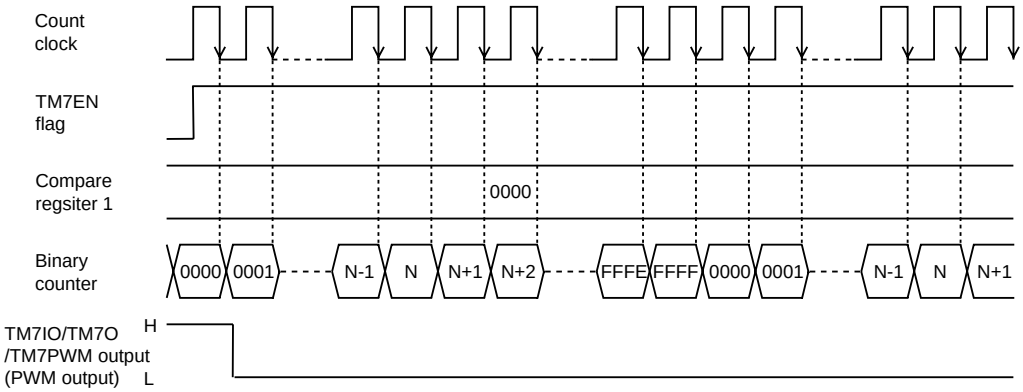


Figure 7-6-2 Count Timing of Standard PWM Output (when Compare Register 1 is x'0000')

PWM output shows "H ", when TM7EN flag is stopped (at "0").

■Count Timing of Standard PWM Output (when Compare Register 1 is x'FFFF')(Timer 7)
Here is the count timing at setting x'FFFF' to the compare register 1.

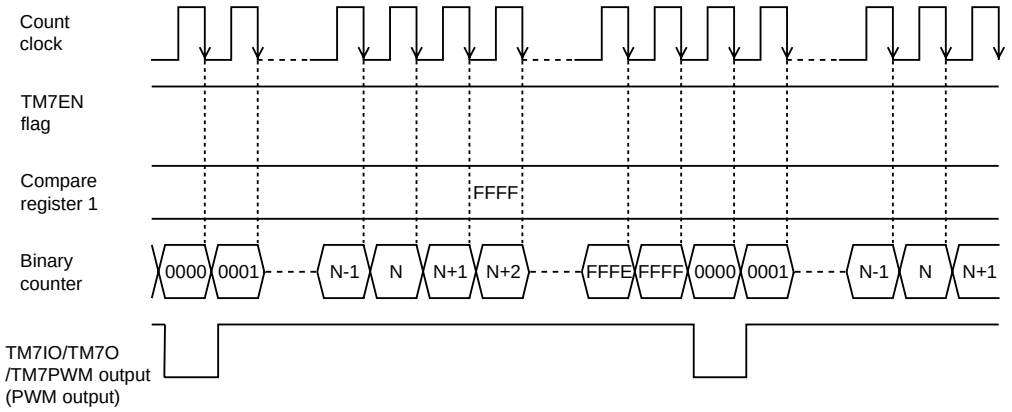


Figure 7-6-3 Count Timing of Standard PWM Output (when Compare Register 1 is x'FFFF')

When the standard PWM output is operated, set the TM7BCR flag of the TM7MD2 register to "0" to select the full count over flow as a binary counter clear source and a PWM output setup ("H" output) source.

By setting the T7PWMSL flag of the TM7MD2 register, the TM7OC1 compare match or the TM7OC2 compare match can be selected as a PWM output reset ("L" output) source.

7-6-2 Setup Example

■Standard PWM Output Setup Example (Timer 7)

The TM7IO output pin (P14) outputs the 1/4 duty PWM output waveform at 305.18 Hz with timer 7. The high frequency oscillation (fosc) is set to be operated at 20 MHz. One cycle of the PWM output waveform is decided by the overflow of a binary counter. "H" period of the PWM output waveform is decided by the set value of a compare register 1.

An example setup procedure, with a description of each step is shown below.

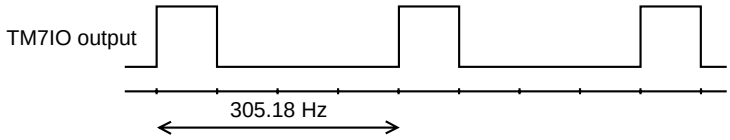


Figure 7-6-4 Output Waveform of TM7IO Output Pin

Setup Procedure	Description
(1) Stop the counter. TM7MD1 (x'3F78') bp4 : TM7EN = 0	(1) Set the TM7EN flag of the timer 7 mode register 1 (TM7MD1) to "0" to stop timer 7 counting.
(2) Set the special function pin to output mode. P1OMD (x'3F2F') bp4 : P1OMD4 = 1 P1DIR (x'3F31') bp4 : P1DIR4 = 1	(2) Set the P1OMD4 flag of the port 1 output mode register (P1OMD) to "1" to set the P14 pin as a special function pin. Set the P1DIR4 flag of the port 1 direction control register (P1DIR) to "1" to set output mode. Add pull-up resistor, if it necessary. [☞ Chapter 4 I/O Ports]
(3) Set the PWM output. TM7MD2 (x'3F79') bp4 : TM7PWM = 1	(3) Set the TM7PWM flag of the timer 7 mode register 2 (TM7MD2) to "1" to select the PWM output.
(4) Set the standard PWM output operation. TM7MD2 (x'3F79') bp5 : TM7BCR = 0	(4) Set the TM7BCR flag of the TM7MD2 register to "0" to select the full count over flow as a binary counter clear source.

Setup Procedure	Description
(5) Select the count clock source. TM7MD1 (x'3F78') bp1-0 : TM7CK1-0 = 00 bp3-2 : TM7PS1-0 = 00	(5) Select fosc at clock source by the TM7CK1-0 flag of the TM7MD1 register. Also, select 1/1 frequency (no division) at count clock source by the TM7PS1-0 flag.
(6) Set "H" period of the PWM output. TM7PR1 (x'3F75', x'3F74')=x'4000'	(6) Set "H" period of the PWM output to the timer 7 preset register 1 (TM7PR1). To be a 1/4 duty of the full count (65536), set as follows ; 65536 / 4 = 16384 (x'4000') At that time, the same value is loaded to the timer 7 compare register 1 (TM7OC1) and the timer 7 binary counter (TM7BC) is initialized to x'0000'.
(7) Start the timer operation. TM7MD1 (x'3F78') bp4 : TM7EN = 1	(7) Set the TM7EN flag of the TM7MD1 register to "1" to start timer 7.

TM7BC counts up from x'0000'. The PWM source waveform outputs "H" until TM7BC reaches the set value of the TM7OC1 register, then, after the match it outputs "L". After that, TM7BC continues to count up, once overflow happens, the PWM source waveform outputs "H" again, and TM7BC counts up from x'0000', again. TM7IO pin outputs one count clock delay of the PWM source waveform.

In the initial state of the PWM output, it is changed to "H" output from "L" output as the PWM operation is selected by the TM7PWM flag of the TM7MD2 register.

7-7 16-bit High Precision PWM Output

(Cycle/Duty can be changed consecutively)

The TM7IO/TM7O/TM7PWM pins output high precision PWM output, which is determined by the match timing of the timer binary counter and the compare register 1 and the match timing of the binary counter and the compare register 2.

7-7-1 Operation

■16-bit High Precision PWM Output Operation (Timer 7)

The PWM waveform with any cycle/duty is generated by setting the cycle of PWM to the compare register 1 (TM7OC1) and setting the duty of the "H" period to the compare register 2 (TM7OC2). The 16-bit timer that high precision PWM output operation function can be used is timer 7.

■Count Timing of High Precision PWM Output (at Normal) (Timer 7)

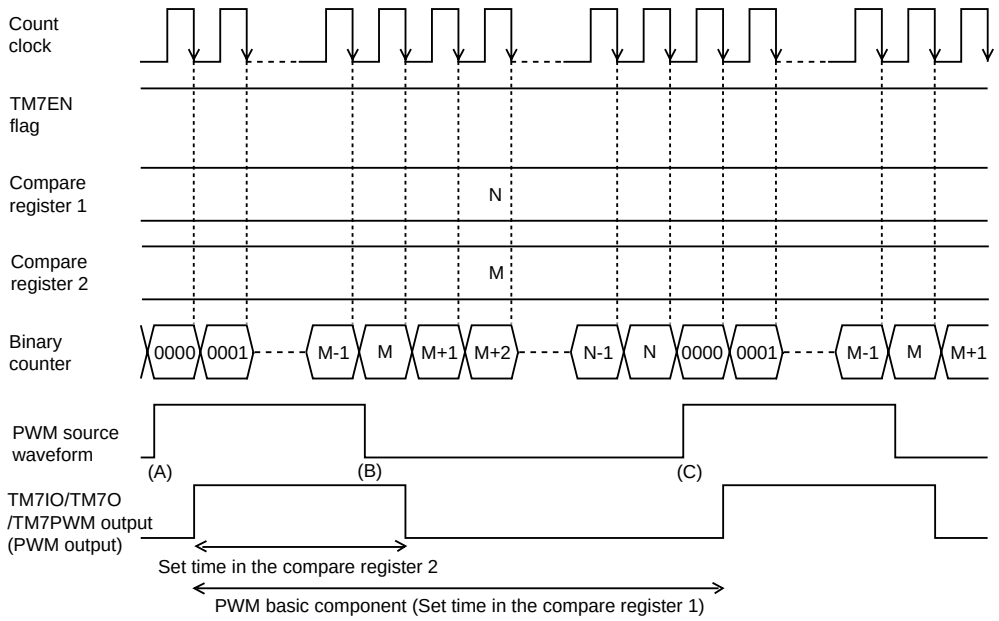


Figure 7-7-1 Count Timing of High Precision PWM Output (at Normal)

PWM source waveform,

(A) is "H" until the binary counter reaches the compare register from x'0000'.

(B) is "L" after the TM7OC2 compare match, then the binary counter counts up till the binary counter reaches the TM7OC1 compare register to be cleared.

(C) is "H", again if the binary counter is cleared.

The PWM output from pin is 1 count clock delay of PWM source waveform. This is happened to form inside to correct the output cycle.

■Count Timing of High Precision PWM Output (When compare register 2 is x'0000'l) (Timer 7)
Here is the count timing as the compare register 2 is set to x'0000' ;

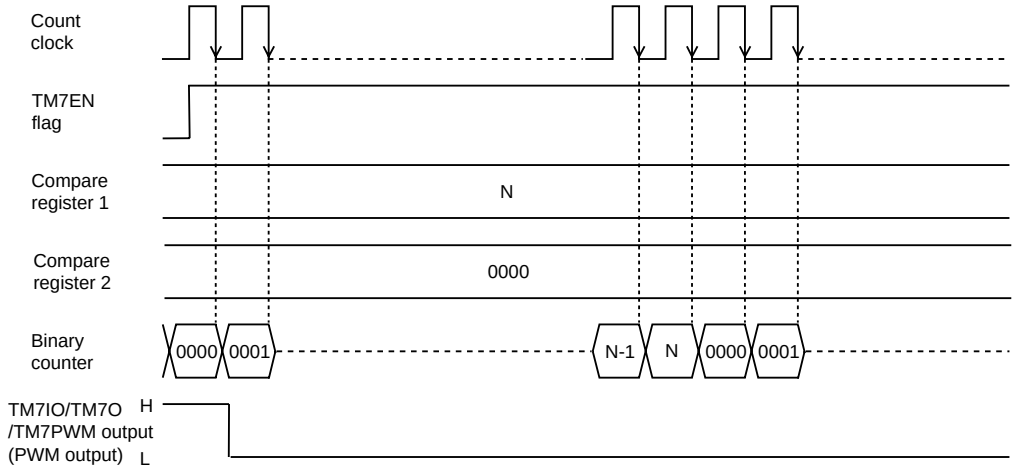


Figure 7-7-2 Count Timing of High Precision PWM Output (When compare register 2 is x'0000')

When the TM7EN flag is stopped (at "0"), the PWM output signal is "H".

■Count Timing of High Precision PWM Output (at compare register 2 = compare register 1) (Timer 7)
Here is the count timing as the compare register 2 is set the same value to the compare register 1 ;

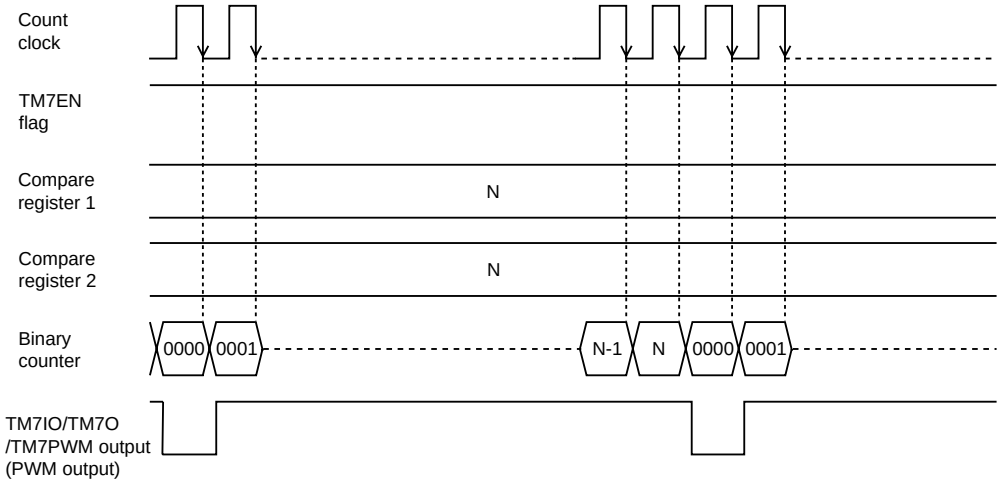


Figure 7-7-3 Count Timing of High Precision PWM Output (at compare register 2=compare register 1)

For the high precision PWM output, set the TMBCR flag of the TM7MD2 register to "1" to select the TM7OC1 compare match as a clear source of the binary counter and as a setup ("H" output) source of the PWM output. Also, set the T7PWMSL flag to "1" to select the TM7OC2 compare match as a reset ("L" output) source of the PWM output.

7-7-2 Setup Example

■High Precision PWM Output Setup Example (Timer 7)

The TM7IO output pin (P14) outputs the 1/4 duty PWM output waveform at 400 Hz with timer 7. Select fosc/2 (at fosc = 20 MHz) as a clock source. One cycle of the PWM output waveform is decided by the set value of a compare register 1. "H" period of the PWM output waveform is decided by the set value of a compare register 2.

An example setup procedure, with a description of each step is shown below.

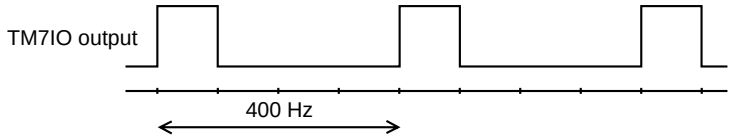


Figure 7-7-4 Output Waveform of TM7IO Output Pin

Setup Procedure	Description
(1) Stop the counter. TM7MD1 (x'3F78') bp4 : TM7EN = 0	(1) Set the TM7EN flag of the timer 7 mode register 1 (TM7MD1) to "0" to stop timer 7 counting.
(2) Set the special function pin to output mode. P1OMD (x'3F2F') bp4 :P1OMD4 = 1 P4DIR (x'3F31') bp4 : P1DIR4 = 1	(2) Set the P1OMD4 flag of the port 1 output mode register (P1OMD) to "1" to set the P14 pin as a special function pin. Set the P1DIR4 flag of the port 1 direction control register (P1DIR) to "1" for output mode. Add pull-up resistor, if it necessary. [☞ Chapter 4 I/O Ports]
(3) Set the PWM output. TM7MD2 (x'3F79') bp4 : TM7PWM = 1	(3) Set the TM7PWM flag of the timer 7 mode register 2 (TM7MD2) to "1" to select the PWM output.
(4) Set the high precision PWM output operation. TM7MD2 (x'3F79') bp5 : TM7BCR = 1 bp6 : T7PWMSL = 1	(4) Set the TM7BCR flag of the TM7MD2 register to "1" to select the TM7OC1 compare match as a clear source of binary counter. Also, set the T7PWMSL flag to "1" to select the TM7OC2 compare match as a duty decision source of the PWM output.

Setup Procedure	Description
(5) Select the count clock source. TM7MD1 (x'3F78') bp1-0 : TM7CK1-0 = 00 bp3-2 : TM7PS1-0 = 01	(5) Select fosc as clock source by the TM7CK1-0 flag of the TM7MD1 register. Also, select 1/2 dividing as count clock source by the TM7PS1-0 flag.
(6) Set the count control flag TM7MD1 (x'3F78') bp6 : TM7CNC = 0	(6) Select the TM7EN flag to the timer 7 count control with the TM7CNC flag of the TM7MD1 register.
(7) Set the PWM output cycle. TM7PR1 (x'3F75',x'3F74') = x'61a7'	(7) Set the PWM output cycle to the timer 7 preset register 1 (TM7PR1). To obtain 400 Hz by dividing 10 MHz, set as follows : 25000 - 1 = 24999 (x'61a7') At that time, the same value is loaded to the timer 7 compare register 1 (TM7OC1), and the timer 7 binary counter (TM7BC) is initialized to x'0000'.
(8) Set the "H" period of the PWM output. TM7PR2 (x'3F7D',x'3F7C')=x'186a'	(8) Set the "H" period of the PWM output to the timer 7 preset register 2 (TM7PR2). To be a 1/4 duty of 25000 dividing, set as follows ; 25000 / 4 = 6250 (x'186a') At that time, the same value is loaded to the timer 7 compare register 2 (TM7OC2).
(9) Start the timer operation. TM7MD1 (x'3F78') bp4 : TM7EN = 1	(9) Set the TM7EN flag of the TM7MD1 register to "1" to start timer 7.

TM7BC counts up from x'0000'. The PWM source waveform outputs "H" until TM7BC matches the set value of the TM7OC2 register. Once they matches, it outputs "L". After that, TM7BC continues to count up, once TM7BC matches the TM7OC1 register to be cleared, the PWM source waveform outputs "H" again and TM7BC counts up from x'0000' again. TM7IO pin outputs one count clock delay of the PWM source waveform.

In the initial state of the PWM output, it is changed from "L" output to "H" output as the PWM output is selected by the TM7PWM flag of the TM7MD register.

Set as the set value of TM7OC2 ≤ the set value of TM7OC1. If it is set as the set value of TM7OC2 > the set value of TM7OC1, the PWM output is a "H" fixed output.

7-8 16-bit Timer Capture

7-8-1 Operation

The value of a binary counter is stored to register at the timing of the external interrupt input signal, or the timing of writing operation with an arbitrary value to the capture register.

■Capture Operation with External Interrupt Signal as a Trigger (Timer 7)
Capture trigger of input capture function is generated at the external interrupt signal that passed through the external interrupt interface block. The capture trigger is selected by the timer 7 mode register 2 (TM7MD2) and the external interrupt control register (IRQ0ICR, IRQ1ICR, IRQ2ICR, IRQ3ICR).
Here are the capture trigger to be selected and the interrupt flag setup.

Table 7-8-1 Capture Trigger

Capture trigger source	Timer 7 mode register 2		External interrupt n control register (IRQnICR)	Both edges interrupt control register (EDGDT)		Interrupt starting edge of external interrupt n
	T7ICT1-0	T7ICEDG	REDGn (bp5)	EDGSEL1	EDGSEL0	
IRQ0 falling edge	00(IRQ0)	1	0	-	0	IRQ0 falling edge
				-	1	IRQ0 both edges
IRQ0 rising edge	00(IRQ0)	1	1	-	0	IRQ0 rising edge
				-	1	IRQ0 both edges
IRQ0 both edges	00(IRQ0)	0	0	-	0	IRQ0 falling edge
			1	-	0	IRQ0 rising edge
			-	-	1	IRQ0 both edges
IRQ1 falling edge	01(IRQ1)	1	0	0	-	IRQ1 falling edge
				1	-	IRQ1 both edges
IRQ1 rising edge	01(IRQ1)	1	1	0	-	IRQ1 rising edge
				1	-	IRQ1 both edges
IRQ1 both edges	01(IRQ1)	0	0	0	-	IRQ1 falling edge
			1	0	-	IRQ1 rising edges
			-	1	-	IRQ1 both edges
IRQ2 falling edge	10(IRQ2)	1	0	-	-	IRQ2 falling edge
IRQ2 rising edge	10(IRQ2)	1	1	-	-	IRQ2 rising edge
IRQ2 both edges	10(IRQ2)	0	0	-	-	IRQ2 falling edge
			1	-	-	IRQ2 rising edge
IRQ3 falling edge	11(IRQ3)	1	0	-	-	IRQ3 falling edge
IRQ3 rising edge	11(IRQ3)	1	1	-	-	IRQ3 rising edge
IRQ3 both edges	11(IRQ3)	0	0	-	-	IRQ3 falling edge
			1	-	-	IRQ3 rising edge

An interrupt request and a capture trigger are generated at switching the valid edge of an external interrupt by program, when the setup is as follows ;

(1) at switching the valid edge from the falling to the rising, when the interrupt pin is "H" level.

(2) at switching the valid edge from the rising to the falling, when the interrupt pin is "L" level.

This is not happened, if the interrupt edge is switched after the generation of an valid edge interrupt set in advance. But when the both edges interrupt function is used, this may be happened. Be sure to consider the noise influence for operation of the interrupt flag on program.

[☞ Chapter 3 3-3-4. Programmable active Edge Interrupt]

■Capture Count Timing at a Both Edges of External Interrupt Signal is selected as a Trigger (Timer 7)

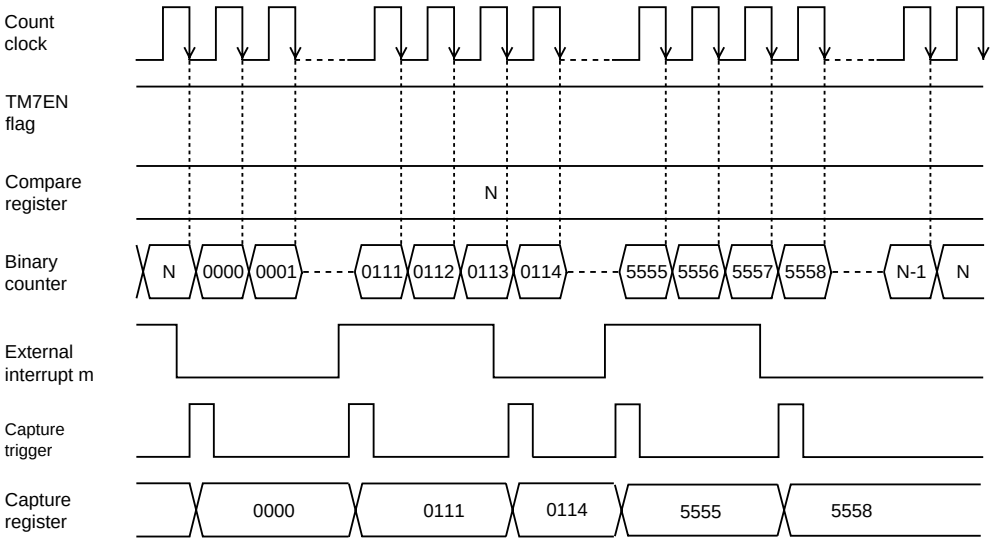


Figure 7-8-1 Capture Count Timing at an External Interrupt Signal is selected as a Trigger (Timer 7)

A capture trigger is generated at the both edges of the external interrupt m input signal. At the same timing, the value of a binary counter is stored to the input capture register. That value is decided by the value of a binary counter at the falling edge of a capture trigger. When the specified edge is selected as a capture trigger generation source, a capture trigger is generated at the interrupt generation specified edge, only. The other count timing is same to the count timing of the timer operation.

When the binary counter is used as a free counter that counts x'0000' to x'FFFF', set the compare register 1 to x'FFFF', or set the TM7BCR flag of the TM7MD2 register to "0".

Even if a capture trigger is generated before the value of the input capture register is read out, the value of the input capture register can be rewritten.

In the initial state after releasing the reset, the generation of trigger by the external interrupt signal is disabled. Set the T7ICEN flag of the TM7MD2 register to "1" to enable the trigger generation.

■Capture Operation that the writing to program is selected as a Trigger (Timer 7)

A capture trigger can be generated by writing an arbitrary value to the input capture register (TM7IC), and at the same timing, the value of the binary counter can be stored to the input capture register.

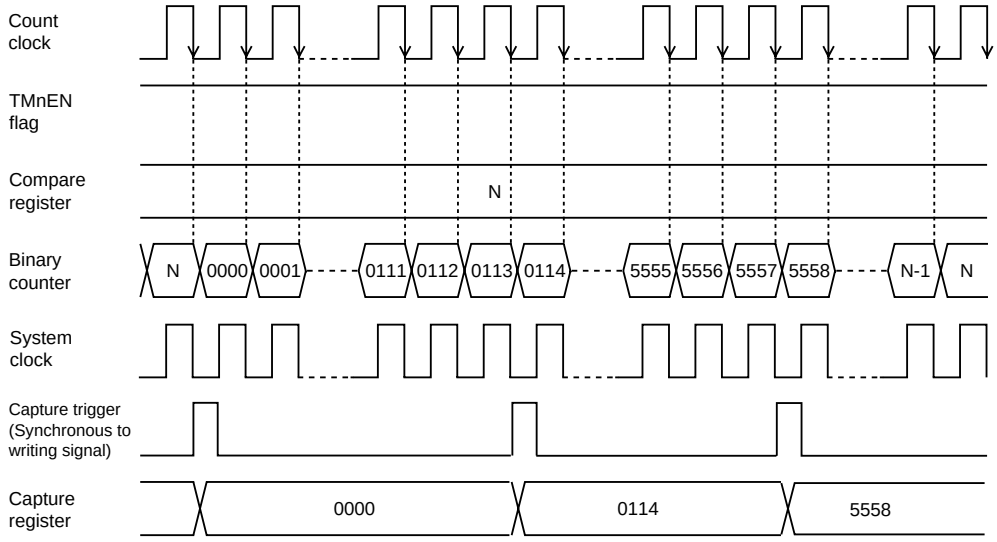


Figure 7-8-2 Capture Count Timing with a Writing Signal to Program as a Trigger (Timer 7)

A capture trigger is generated at the writing signal to the input capture register. The writing signal is generated at the last cycle of the writing instruction. At this timing, the value of the binary counter is stored to the input capture register. That value is decided by the value of the binary counter at the falling edge of the capture trigger. The other timing is same to the timer operation.

The writing to the input capture register to generate a capture trigger should be done with a 8-bit access instruction to the TM7ICL register or the TM7ICH register.
At this time, data is not actually written to the TM7IC register.

On hardware, there is no flag to disable the capture operation with the writing operation to the software as a trigger. Capture operation is enabled, regardless of the T7ICEN flag of the TM7MD2 register.

7-8-2 Setup Example

■Capture Function Setup Example (Timer 7)

Pulse width measurement is enabled by storing the value of the binary counter to the capture register at the interrupt generation edge of the external interrupt 0 input signal with timer 7. The interrupt generation edge is specified to be the rising edge.

An example setup procedure, with a description of each step is shown below.

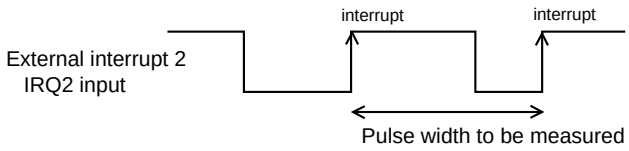


Figure 7-9-3 Pulse Width Measurement of External Interrupt 0

Setup Procedure	Description
(1) Stop the counter. TM7MD1 (x'3F78') bp4 : TM7EN = 0	(1) Set the TM7EN flag of the timer 7 mode register 1 (TM7MD1) to "0" to stop timer 7 counting.
(2) Select the condition for timer clear. TM7MD2 (x'3F79') bp5 : TM7BCR = 1	(2) Set the TM7BCR flag of the timer 7 mode register 2 (TM7MD2) to "1" to select the compare match as a clear source of binary counter.
(3) Select the count clock source. TM7MD1 (x'3F78') bp1-0 : TM7CK1-0 = 00 bp3-2 : TM7PS1-0 = 00	(3) Select fosc as clock source by the TM7CK1-0 flag of the TM7MD1 register. And select 1/1 (no dividing) of fosc as count clock source by the TM7PS1-0 flag.
(4) Set the count control flag TM7MD1 (x'3F78') bp6 : TM7CNC = 0	(4) Select the TM7EN flag to the timer 7 count control with the TM7CNC flag of the TM7MD1 register.
(5) Select the capture trigger generation interrupt source. TM7MD2 (x'3F79') bp1-0 : T7ICT1-0 = 00	(5) Select the external interrupt 0 (IRQ0) input as a generation source of capture trigger by the T7ICT1-0 flag of the TM7MD2 register.
(6) Select the interrupt generation valid edge. IRQ2ICR (x'3FE4') bp5 : REDG2 = 1	(6) Set the REDG2 flag of the external interrupt 2 control register (IRQ2ICR) to "1" to select the rising edge as the interrupt generation valid edge.

Setup Procedure	Description
(7) Select the capture trigger generation edge. TM7MD2 (x'3F79') bp7 : T7ICEDG = 1	(7) Set the T7ICEDG flag of the TM7MD2 register to "1" to select the external interrupt valid edge as a generation source of capture trigger.
(8) Set the compare register. TM7PR1(x'3F75',x'3F74') = x'FFFF'	(8) Set the timer 7 preset register 1 (TM7PR1) to x'FFFF'. At that time, the same value is loaded to the timer 7 compare register 1 (TM7OC1), and the timer 7 binary counter (TM7BC) is initialized to x'0000'.
(9) Set the interrupt level. IRQ2ICR (x'3FE4') bp7-6 : IRQ2LV1-0= 10	(9) Set the interrupt level by the IRQ2LV1-0 flag of the IRQ2ICR register. If any interrupt request flag may be set already, clear them. [☞ Chapter 3 3-1-4. Interrupt Flag Setup]
(10) Enable the interrupt. IRQ2ICR (x'3FE4') bp1 : IRQ2IE = 1	(10) Enable the interrupt by setting the IRQ2IE flag of the IRQ2ICR register to "1".
(11) Enable the capture trigger generation. TM7MD2 (x'3F79') bp2 : T7ICEN = 1	(11) Enable the capture trigger generation by setting the T7ICEN flag of the TM7MD2 register to "1".
(12) Start the timer operation. TM7MD1 (x'3F78') bp4 : TM7EN = 1	(12) Set the TM7EN flag of the TM7MD1 register to "1" to start timer 7.

TM7BC counts up from x'0000'. At the timing of the rising edge of the external interrupt 2 input signal, the value of TM7BC is stored to the TM7IC register. And at that time, the pulse width between rising edges of the external interrupt input signal can be measured by reading the value of TM7IC register by the interrupt service routine, and by calculating the margin of the capture values (the values of the TM7IC register).

7-9 Remote Control Carrier Output

7-9-1 Operation

Carrier pulse for remote control can be generated.

■Operation of Remote Control Carrier Output (Timer 7)
Remote control carrier pulse is based on PWM output signal of timer 7. Duty cycle is selected from 1/2, 1/3 and base timer output. RMOUT (P10) outputs remote control carrier output signal.
There are two ways for remote control output; 1. use program 2. use generation of timer 2 interrupt request to enable timer 7 activation and remote control carrier output by hardware.

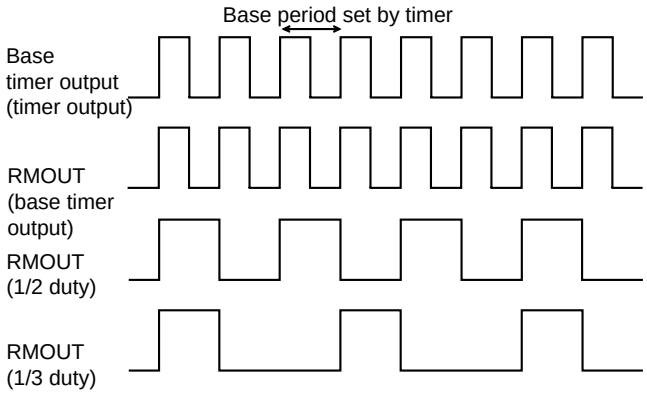


Figure 7-9-1 Duty Cycle of Remote Control Carrier Output Signal

■Count Timing of Remote Control Carrier Output by Program (Timer 7)

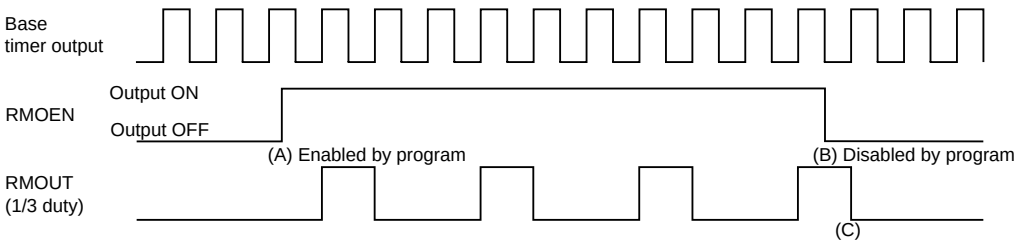


Figure 7-9-2 Count Timing of Remote Control Carrier Output by Program (Timer 7)

(A) Set "1" to RMOEN flag by program to enable remote control carrier output.
(B) Set "0" to RMOEN flag by program to disable remote control carrier output.
(C) Even if the RMOEN flag is off when the carrier output is high, the carrier waveform is held by the synchronizing circuit.

Before the RMOEN flag is switched to on, set the P1OMD0 flag of the P1OMD register to "1". After it is switched to off, set it to "0".

When the RMOEN flag is changed, do not change the base cycle and its duty at the same time. If they are changed at the same time, the carrier wave form is not output properly.

■Count Timing of Remote Control Carrier Output by Hardware (1/3 duty) (timer 7)

Set out below are how to enable timer 7 activation and remote control carrier output with generation of timer 2 interrupt request using hardware and select 1/3 duty for remote control output.

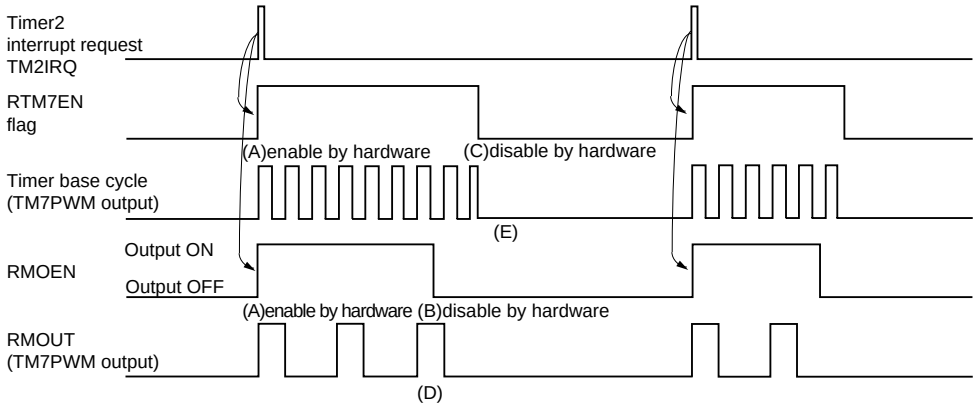


Figure 7-9-3 Count Timing of Remote Control Carrier Output by Hardware (1/3 duty) (timer 7)

(A) When TM2IRQ interrupt request flag is set RTM7EN and RMOEN flags of TM7MD1 and RMCTR registers are set to "1" and timer 7 counting is started and remote carrier output becomes ON.
But the first "H" period of carrier output is shorter for half clock of timer 7 count clock source. For example, it becomes shorter for 31.25 ns at count clock source fosc=16 MHz.

(B) Set "0" to RMOEN flag by program to OFF the remote control carrier output.

(C) Set "0" to RTM7EN flag by program to stop timer 7 counting after carrier output becomes "L".

(D) Even if the RMOEN flag is off when the carrier output is high, the carrier waveform is held by the synchronizing ci

(E) Clear timer 7 binary counter before next TM2IRQ interrupt request is set.

■Count Timing of Remote Control Carrier Output by Hardware (any duty) (timer 7)

Set out below are how to enable timer 7 activation and remote control carrier output with generation of timer 2 interrupt request using hardware and select base timer output as remote control output duty to output given duty of timer 7 PWM output as a remote control carrier output.

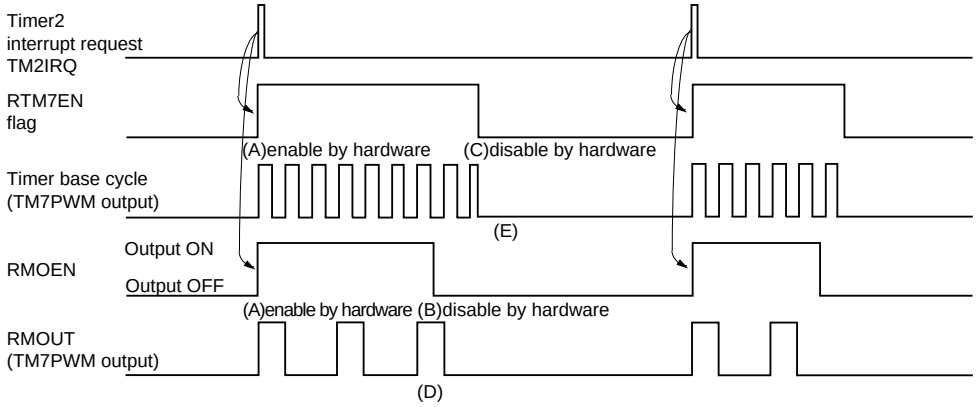


Figure 7-9-3 Count Timing of Remote Control Carrier Output by Hardware (any duty) (timer 7)

(A) When TM2IRQ interrupt request flag is set RTM7EN and RMOEN flags of TM7MD1 and RMCTR registers are set to "1" by hardware and timer 7 counting is started and remote carrier output becomes ON.
But the first "H" period of carrier output is shorter for half clock of timer 7 count clock source. For example, it becomes shorter for 31.25 ns at count clock source fosc=16 MHz.

(B) Set "0" to RMOEN flag by program to OFF the remote control carrier output.

(C) Set "0" to RTM7EN flag by program to stop timer 7 counting after carrier output becomes "L".

(D) Even if the RMOEN flag is off when the carrier output is high, the carrier waveform is held by the synchronizing ci

(E) Clear timer 7 binary counter before next TM2IRQ interrupt request is set.

If RMETN2 flag of Remote control carrier output control register (RMCTR) is set to "1" during timer 2 operation, RMOEN and RTM7EN flag are set every time timer 2 interrupt request is generated.
Also note that if timer 2 interrupt is enabled, interrupt is accepted every time TM2IRQ is generated.

7-9-2 Setup Example

■Remote Control Carrier Output Setup Example Using Program (Timer 7)
Here is the setting example that the RMOUT pin outputs the 1/3 duty carrier pulse signal with "H" period of 27.2 µs (36.7 kHz), by using timer7 PWM output. The source clock of timer 7 is set to fosc (at 8 MHz). An example setup procedure, with a description of each step is shown below.

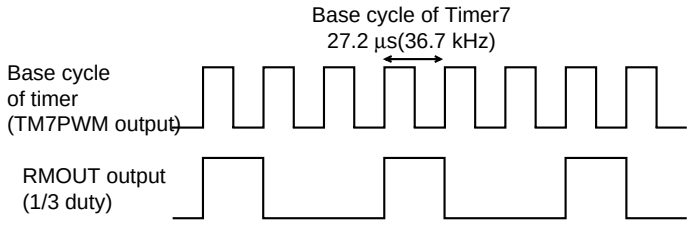


Figure 7-9-4 Output Wave Form of RMOUT Output Pin

Setup Procedure	Description
(1) Disable the remote control carrier output. RMCTR (x'3F6E') bp3 : RMOEN = 0	(1) Set the RMOEN flag of the remote control carrier output control register (RMCTR) to "0" to disable the remote control carrier output.
(2) Select the base cycle setting timer. RMCTR (x'3F6E') bp0 : RMBTMS = 0	(2) Set the RMBTMS flag of the RMCTR register to "0" to set the timer as a base cycle setting timer.
(3) Select the carrier output duty. RMCTR (x'3F6E') bp2,1 : RMDTY1,0 = 1	(3) Set the RMDTY1, 0 flag of the RMCTR register to "0,1" to select 1/3 duty.
(4) Stop the counter. TM7MD1 (x'3F78') bp4 : TM7EN = 0	(4) Set the TM7EN flag of the timer 7 mode register (TM7MD1) to stop the timer 7 count ing.
(5) Set the remote control carrier output of the special function pin. P1OMD (x'3F2F') bp0 : P1OMD0 = 1 P1DIR (x'3F31') bp0 : P1DIR0 = 1 RMCTR (x'3F6E') bp4 :TM7RM = 1	(5) Set the P1OMD0 flag of the port 1 output mode register (P1OMD) to "1" to set P10 pin as a special function pin. Set the P1DIR0 flag of the port 1 direction control register (P1DIR) to "1" for output mode. Set the TM0RM flag of the RMCTR register to "1" to select the remote control carrier output.

Setup Procedure	Description
(6) Set the timer 7. Set the PWM output. TM7MD2 (x'3F79') bp4 : TM7PWM = 1	(6) Set the TM7PWM flag of the timer 7 mode register 2 (TM7MD2) to "0" to select PWM output.
Set the high precision output. TM7MD2 (x'3F79') bp5 : TM7BCR = 1 bp6 : T7PWMSL = 1	Set the TM7BCR flag of the TM7MD2 register to "1" to select TM7OC1 compare match as a clear source of the binary counter. Also set "1" to the T7PWMSL flag to select TM7OC2 compare match as a duty determination source of PWM output.
Select count clock souece TM7MD1 (x'3F78') bp1-0 : TM7CK1-0 = 00 bp3-2 : T7PS1-0 = 00	Select fosc as a clock source with TM7CK1-0 flag of the TM7MD1 register. Also select 1/1 frequency (no division) as a count clock source with TM7PS1-0 flag.
Sey the PWM output cycle. TM7PR1 (x'3F75', x'3F74')= x'00D9'	Set the base cycle of remote control carrier by writing x'00D9' to the timer 7 preset register 1 (TM7PR1). To obtain 36.7 kHz by dividing 8 MHz, set as; (8 MHx/36.7 kHz)-1 = 217 (x'D9) At this time, the same value is loaded to timer 7 compare regoster 1 (TM7OC1) to initialize timer 7 binary counter (TM7BC) to x'0000'.
Set the "H" period of the PWM output. TM7PR2 (x'3F7D', x'3F7C')= x'0010'	Set the value of the "H" period of PWM output to timer 7 preset register 2 (TM7PR2). Though this duty setting of PWM output does not affect remote control carrier output, set the TM7PR2 register any smaller value than the set value of the TM7PR1 register (x'0010' in this setup example).
(7) Start the timer operation. TM7MD1 (x'3F78') bp4 : TM7EN = 1	(7) Set "1" to the TM7EN flag of the TM7MD1 register to start timer 7.

Setup Procedure	Description
(8) Enable the remote control carrier output. RMCTR (x'3F6E') bp3 : RMOEN = 1	(8) Set the RMOEN flag of the RMCTR register to "1" to enable the remote control carrier output.

TM7BC counts up from x'00'. Timer 7 outputs the base cycle pulse set in TM7PR1. Then, the 1/3 duty remote control carrier pulse signal is output. If the RMOEN flag of the RMCTR register is set to "0", the remote control carrier pulse signal output is stopped.

■Remote Control Carrier Output Setup Example Using Hardware (Timer 7)

Here is the setting example that the RMOUT pin outputs the 2/3 duty carrier pulse signal with "H" period of 1.25 μs (800 kHz), by using timer7 PWM output.

Timer 2 interrupt is used for the acceptance of the remote control carrier output. Hardware accepts carrier output in every 20 μs (50 kHz) cycle, which is setup in timer 2.

The source clock of timer 2 and 7 is set to fosc (at 8 MHz).

An example setup procedure, with a description of each step is shown below.

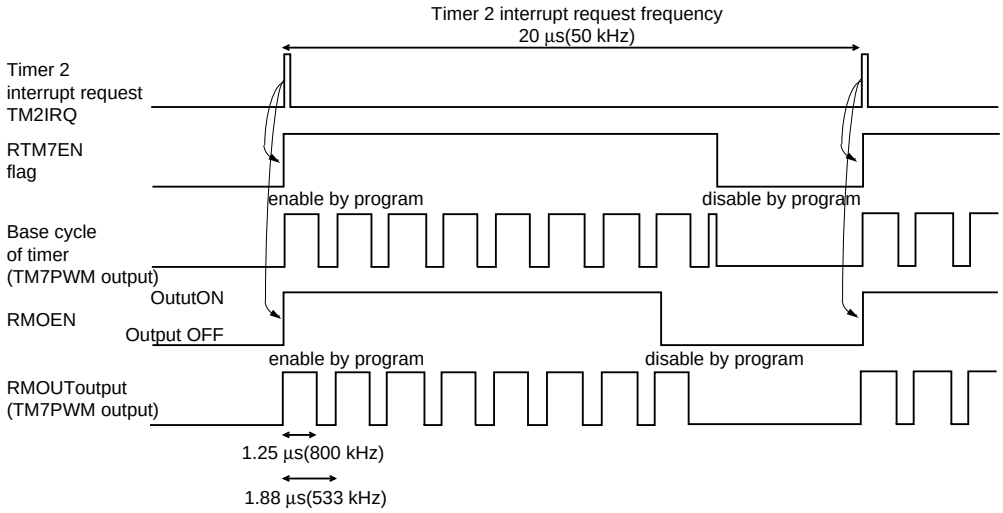


Figure 7-9-5 Output Wave Form of RMOUT Output Pin

Setup Procedure	Description
(1) Disable the remote control carrier output. RMCTR (x'3F6E') bp3 : RMOEN = 0	(1) Set the RMOEN flag of the remote control carrier output control register (RMCTR) to "0" to disable the remote control carrier output.
(2) Select the base cycle setting timer. RMCTR (x'3F6E') bp0 : RMBTMS = 0	(2) Set the RMBTMS flag of the RMCTR register to "0" to set the timer as a base cycle setting timer.
(3) Select the carrier output duty. RMCTR (x'3F6E') bp2,1 : RMDTY1,0 = 1	(3) Set the RMDTY1, 0 flag of the RMCTR register to "0,1" to select 1/3 duty.
(4) Stop the counter. TM7MD1 (x'3F78') bp4 : TM7EN = 0 RMCTR (x'3F6E') bp5 : RTM7EN = 0 TM2MD (x'3F5C') bp3 : TM2EN = 0	(4) Set the TM7EN flag of the timer 7 mode register (TM7MD1) to stop the timer 7 count ing. Also, set "0" to the TM2EN flag of the timer 2 mode register (TM2MD) to halt the timer 2 counting.

Setup Procedure	Description
(5) Select the timer 7 count control factor. TM7MD1 (x'3F78') bp6 : TM7CNC = 1	(5) Set the TM7CNC flag of the timer 7 mode register 1 (TM7MD1) to "1" to select RTM7EN as a count control factor of timer 7.
(6) Set the special function pins to remote control carrier output. P1OMD (x'3F2F') bp0 : P1OMD0 = 1 P1DIR (x'3F31') bp0 : P1DIR0 = 1 RMCTR (x'3F6E') bp4 : TM7RM = 1	(5) Set the P1OMD0 flag of the port 1 output mode register (P1OMD) to "1" to set P10 pin as a special function pin. Set the P1DIR0 flag of the port 1 direction control register (P1DIR) to "1" for output mode. Set the TM7RM flag of the RMCTR register to "1" to select the remote control carrier output.
(7) Set the timer 2. Select normal timer operation. TM2MD (x'3F5C') bp4 : TM2PWM = 0 bp5 : TM2MOD = 0	(7) Set the TM2PWM and TM2MOD flags of the TM2MD register to "0" to select normal timer operation.
Select the count clock source. TM2MD (x'3F5C') bp2-0 : TM2CK2-0 = 000	Set the TM2CK2-0 flag of the TM2MD register to select fosc as a clock source.
Set the interrupt generation cycle. TM2OC (x'3F5A') = x'9F'	Set the value of the interrupt generation cycle to the timer 2 compare register (TM2OC). The set value is 59 (x'9F) for 160 dividing. At the same time, the timer 2 binary counter (TM2BC) is initialized to x'00'.
Disable the interrupts TM2ICR (x'3FEB') bp1 : TM2IE = 0	Set the TM2IE flag of the timer 2 interrupt control register (TM2ICR) to "0" to disable interrupts.
(8) Set the timer 7. Set the PWM output. TM7MD2 (x'3F79') bp4 : TM7PWM = 1	(8) Set "1" to the TM7PWM flag of the timer 7 mode register 2 (TM7MD2) to select PWM output.

Setup Procedure	Description
Set the high precision output. TM7MD2 (x'3F79') bp5 : TM7BCR = 1 bp6 : T7PWMSL = 1	Set the TM7BCR flag of the TM7MD2 register to "1" to select TM7OC1 compare match as a clear source of the binary counter. Also set "1" to the T7PWMSL flag to select TM7OC2 compare match as a duty determination source of PWM output.
Select count clock souece TM7MD1 (x'3F78') bp1-0 : TM7CK1-0 = 00 bp3-2 : T7PS1-0 = 00	Select fosc as a clock source with TM7CK1-0 flag of the TM7MD1 register. Also select 1/1 frequency (no division) as a count clock source with TM7PS1-0 flag.
Set the PWM output cycle. TM7PR1 (x'3F75', x'3F74')= x'000E'	Set the value of the PWM output cycle to the timer 7 preset register 1(TM7PR1). To obtain 533 kHz by dividing 8 MHz, set as; 15-1=14(x'000E') At this time, the same value is loaded to timer 7 compare regoster 1 (TM7OC1) to initialize timer 7 bynary counter (TM7BC) to x'0000'.
Set the "H" period of the PWM output. TM7PR2 (x'3F7D', x'3F7C')= x'000A'	Set the value of the "H" period of PWM output to timer 7 preset register 2 (TM7PR2). To set 2/3 duty of 15 dividing set as; 15x2/3=10(x'000A') At this time, the same value is loaded to timer 7 compare regoster 2 (TM7OC2).
(9) Set the TM2IRQ by hardware RMCTR (x'3F6E') bp6 : RMETM2 = 1	(9) Set the RMETM2 flag of the RMCTR register to "1" to select that the RMOEN and RTM7EN flags are set by TM2IRQ through hardware.
(10) Start the timer 2 operation. TM2MD (x'3F5C') bp3 : TM2EN = 1	(10) Set "1" to the TM2EN flag of the TM2MD register to start timer 2.

When timer 2 interrupt request TM2IRQ is generated, the RMOEN and RTM7EN flags are set to "1". This accepts the remote control carrier output and TM7BC counts up from x'00'. Timer 7 outputs the the PWM output with duty set in TM7PR1 and with cycle set in TM7PR2, and then the PWM output is output as remote control carrier pulse output.
After remote control carrier output with aimed pulse, following two setups should be done by program.

(1) Clear the RMOEN flag to "0" to stop the remote control carrier output.
(2) Afrer confirming that the remote control carrier output became "L", clear the RTM7EN flag to "0" to stop the timer 7 count operation.
(3) Write to the TM7PR1 register again to clear the timer 7 binary counter.

When next TM2IRQ is generated, RMOEN and RTN7EN flags are set to "1" by the hardware, and remote control carrier output is started again.

Chapter 8 Time Base Timer / 8-bit Free-running Timer

8-1 Overview

This LSI has a time base timer and a 8-bit free-running timer (timer 6).
Time base timer is a 15-bit timer counter. These timers can stop the timer counting only at stand-by mode (STOP mode).

8-1-1 Functions

Table 8-1-1 shows the clock sources and the interrupt generation cycles used for timer 6 and time base timer.

Table 8-1-1 Clock Source and Generation Cycle

	Time base timer	Timer 6 (8-bit free-running timer)
8-bit timer operation	-	√
Interrupts / source	TBIRQ	TM6IRQ
Clock source	fosc fx	fosc fx fs fosc X $1/2^{12}$ (*1) fosc X $1/2^{13}$ (*1) fx X $1/2^{12}$ (*2) fx X $1/2^{13}$ (*2)
Interrupt generation cycle	fosc X $1/2^{7}$ (*1) fosc X $1/2^{8}$ (*1) fosc X $1/2^{9}$ (*1) fosc X $1/2^{10}$ (*1) fosc X $1/2^{13}$ (*1) fosc X $1/2^{15}$ (*1) fx X $1/2^{7}$ (*2) fx X $1/2^{8}$ (*2) fx X $1/2^{9}$ (*2) fx X $1/2^{10}$ (*2) fx X $1/2^{13}$ (*2) fx X $1/2^{15}$ (*2)	The interrupt generation cycle is decided by the arbitrary value written to TM6OC.
fosc : Machine clock (High speed oscillation) fx : Machine clock (Low speed oscillation) fs : System clock [☞ Chapter 2 2-5. Clock Switching] - *1 can be used as a clock source of time base timer is selected to 'fosc'. - *2 can be used as a clock source of time base timer is selected to 'fx'. - Time base timer and timer 6 cannot stop timer 6 counting.		

8-1-2 Block Diagram

■Timer 6, Time Base Timer Block Diagram

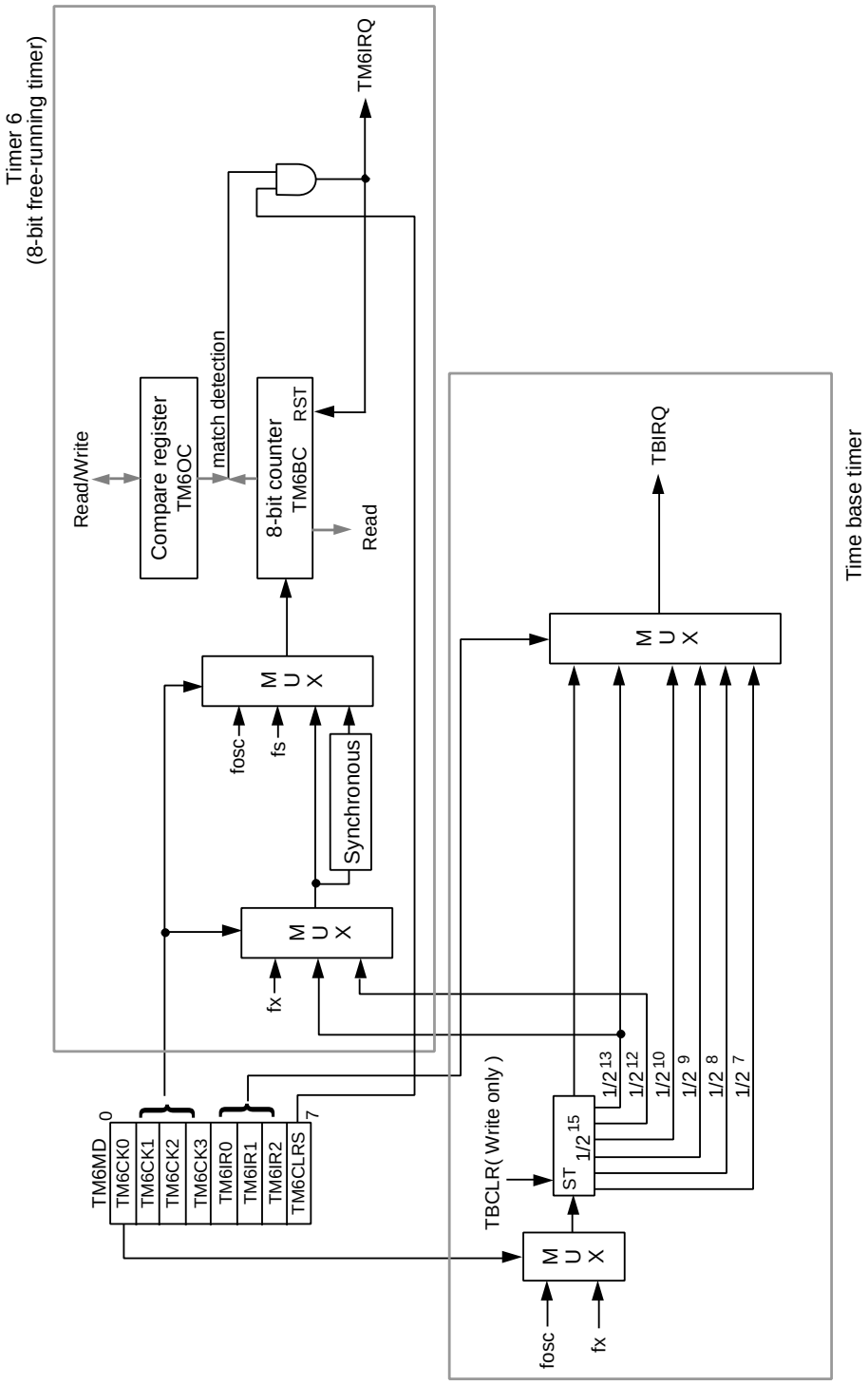


Figure 8-1-1 Block Diagram (Timer 6, Time Base Timer)

8-2 Control Registers

Timer 6 consists of binary counter (TM6BC), compare register (TM6OC), and is controlled by mode register (TM6MD). Time base timer is controlled by mode register (TM6MD) and time base timer clear register (TBCLR), too.

8-2-1 Control Registers

Table 8-2-1 shows the registers that control timer 6, time base timer.

Table 8-2-1 Control Registers

	Register	Address	R/W	Function	Page
Timer 6	TM6BC	x'03F68'	R	Timer 6 binary counter	VIII - 5
	TM6OC	x'03F69'	R/W	Timer 6 compare register	VIII - 5
	TM6MD	x'03F6A'	R/W	Timer 6 mode register	VIII - 6
	TM6ICR	x'03FEF'	R/W	Timer 6 interrupt control register	III - 23
Timer base timer	TM6MD	x'03F6A'	R/W	Timer 6 mode register	VIII - 6
	TBCLR	x'03F6B'	W	Time base timer clear control register	VIII - 5
	TBICR	x'03FF0'	R/W	Time base interrupt control register	III - 24

R/W : Readable / Writable
R : Readable only
W : Writable only

8-2-2 Programmable Timer Registers

Timer 6 is a 8-bit programmable counter.
Programmable counter consists of compare register (TM6OC) and binary counter (TM6BC).

Binary counter is a 8-bit up counter. When the TM6CLRS flag of the timer 6 mode register (TM6MD) is "0" and the interrupt cycle data is written to the compare register (TM6OC), the timer 6 binary counter (TM6BC) is cleared to x'00'.

■Timer 6 Binary Counter (TM6BC)

	7	6	5	4	3	2	1	0	
TM6BC	TM6BC7	TM6BC6	TM6BC5	TM6BC4	TM6BC3	TM6BC2	TM6BC1	TM6BC0	(At reset : X X X X X X X X)

Figure 8-2-1 Timer 6 Binary Counter (TM6BC : x'03F68', R)

■Timer 6 Compare Register (TM6OC)

	7	6	5	4	3	2	1	0	
TM6OC	TM6OC7	TM6OC6	TM6OC5	TM6OC4	TM6OC3	TM6OC2	TM6OC1	TM6OC0	(At reset : X X X X X X X X)

Figure 8-2-2 Timer 6 Compare Register (TM6OC : x'03F69', R/W)

Time base timer cannot stop counting but the software can reset its operation. Time base timer can be cleared by writing an arbitrary value to the time base timer clear control register (TBCLR).

■Time Base Timer Clear Control Register (TBCLR)

	7	6	5	4	3	2	1	0	
TBCLR	-	-	-	-	-	-	-	-	(For writing only)

Figure 8-2-3 Time Base Timer Clear Control Register (TBCLR : x'03F6B')

8-2-3 Timer Mode Registers

This is a readable / writable register that controls timer 6 and time base timer.

■Timer 6 Mode Register (TM6MD)

	7	6	5	4	3	2	1	0
TM6MD	TM6CLRS	TM6IR2	TM6IR1	TM6IR0	TM6CK3	TM6CK2	TM6CK1	TM6CK0

(At reset : 0 0 0 0 0 0 0 0)

TM6CKS0	Time base timer clock source
0	fosc
1	fx

TM6CK3	TM6CK2	TM6CK1	Timer 6 clock source
0	0	0	fosc
		1	fs
	1	0	fx
		1	Synchronous fx
1	0	0	Time base selection clock × $1/2^{13}$
		1	Synchronous time base selection clock × $1/2^{13}$
	1	0	Time base selection clock ×$1/2^{12}$
		1	Synchronous time base selection clock × $1/2^{12}$

TM6IR2	TM6IR1	TM6IR0	Time base timer interrupt cycle selection
0	0	0	Time base selection clock × $1/2^{7}$
		1	Time base selection clock × $1/2^{8}$
	1	0	Time base selection clock × $1/2^{9}$
		1	Time base selection clock × $1/2^{10}$
1	0	-	Time base selection clock × $1/2^{13}$
	1		Time base selection clock × $1/2^{15}$

TM6CLRS	Timer 6 binary counter clear
0	Enable the initialization of TM6BC as TM6OC is written
1	Disable the initialization of TM6BC as TM6OC is written

* TM6IRQ is disabled as TM6CLRS = 0,
TM6IRQ is enabled as TM6CLRS = 1.

Figure 8-2-4 Timer 6 Mode Register (TM6MD : x'03F6A', R/W)

8-3 8-bit Free-running Timer

8-3-1 Operation

■8-bit Free-running Timer (Timer 6)

The generation cycle of the timer interrupt is set by the clock source selection and the setting value of the compare register (TM6OC), in advance. If the binary counter (TM6BC) reaches the setting value of the compare register, an interrupt is generated at the next count clock, then the binary counter is cleared and counting is restarted from x'00'.

Table 8-3-1 shows clock source that can be selected.

Table 8-3-1 Clock Source at Timer Operation (Timer 6)

Clock source	1count time
fosc	50 ns
fx	30.5 µs
fs	100 ns
fosc X $1/2^{12}$	204.8 µs
fosc X $1/2^{13}$	409.6 µs
fx X $1/2^{12}$	125 ms
fx X $1/2^{13}$	250 ms
Notes : as fosc = 20(MHz) fx = 32.768(kHz) fs = fosc/2 = 10 MHz	

Timer 6 cannot stop its timer counting except at stanby mode (STOP mode).

■8-bit Free-running Timer as a 1 minute-timer, a 1 second-timer

Table 8-3-2 shows the clock source selection and the TM6OC register setup, when a 8-bit free-running timer is used as a 1 minute-timer, a 1 second-timer.

Table 8-3-2 1 minute-timer, 1 second-timer Setup (Timer 6)

Interrupt Generation Cycle	Clock Source	TM6OC Register
1 min	fx x $1/2^{13}$	X'EF'
1 s	fx x $1/2^{12}$	X'07'
	fx x $1/2^{13}$	X'03'
fx = 32.768(kHz)		

When the 1 minute-timer (1 min.) is set on Table 8-3-2, the bp1 waveform frequency (cycle) of the TM6BC register is 1 Hz (1 s). So, that can be used for adjusting the seconds.

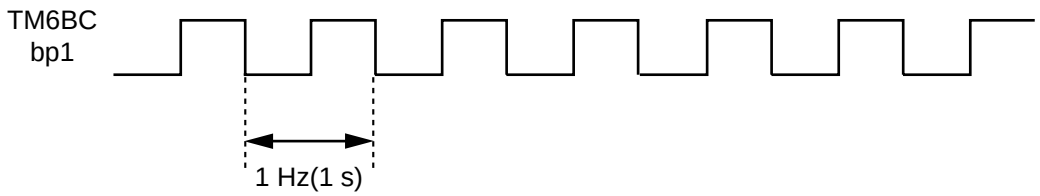


Figure 8-3-1 Waveform of TM6BC Register bp1 (Timer 6)

■Count Timing of Timer Operation (Timer 6)

Binary counter counts up with the selected clock source as a count clock.

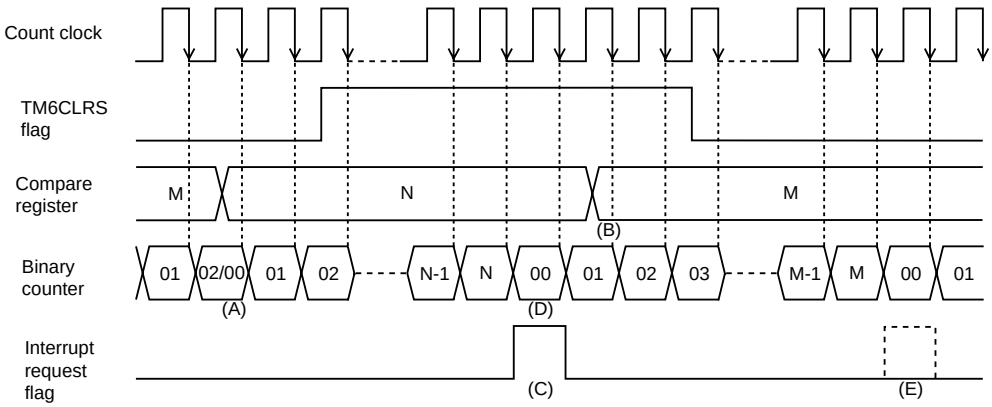


Figure 8-3-2 Count Timing of Timer Operation (Timer 6)

(A) When any data is written to the compare register as the TM6CLRS flag is "0", the binary counter is cleared to x'00'.

(B) Even if any data is written to the compare register as the TM6CLRS flag is "1", the binary counter is not changed.

(C) When the binary counter reaches the value of the compare register as the TM6CLRS flag is "1", an interrupt request flag is set at the next count clock.

(D) When an interrupt request flag is set, the binary counter is cleared to x'00' and restarts the counting.

(E) Even if the binary counter reaches the value of the compare register as the TM6CLRS flag is "0", no interrupt request flag is set.

When the binary counter reaches the value in the compare register, the inrerrupt request flag is set and the binary counter is cleared, at the next count clock.
So, set the compare register as :

Compare register setting = (count till the interrupt request - 1)

If fx is selected as the count clock source in timer 6, when the binary counter is read at operation, uncertain value on counting up may be read. To prevent this, select the synchronous fx as the count clock source.
But if the synchronous fx is selected as the count clock source, CPU mode cannot return from STOP/HALT mode.

If the compare register is set the smaller than the binary counter during the count operation, the binary counter counts up to the overflow, at first.

8-3-2 Setup Example

■Timer Operation Setup (Timer 6)

Timer 6 generates an interrupt constantly for timer function. Fs(fosc = 16 MHz) is selected as a clock source to generate an interrupt every 200 cycles (25 µs).

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Enable the binary counter initialization. TM6MD (x'3F6A') bp7 : TM6CLRS = 0	(1) Set the TM6LRS flag of the timer 6 mode register (TM6MD) to "0". At that time, the initialization of the timer 6 binary counter (TM6BC) is enabled.
(2) Select the clock source. TM6MD (x'3F6A') bp3-1 : TM6CK3-1 = 001	(2) Clock source can be selected by the TM6CK3-1 flag of the TM6MD register. Actually, fs is selected.
(3) Set the interrupt generation cycle. TM6OC (X'3F69') = x'F9'	(3) Set the interrupt generation cycle to the timer 6 compare register (TM6OC). At that timer, TM6BC is initialized to x'00'.
(4) Enable the interrupt request generation. TM6MD (x'3F6A') bp7 : TM6CLRS = 1	(4) Set the TM6CLRS flag of the TM6MD register to "1" to enable the interrupt request generation.
(5) Set the interrupt level. TM6ICR (x'3FEF') bp7-6 : TM6LV1-0 = 01	(5) Set the interrupt level by the TM6LV1-0 flag of the timer 6 interrupt control register (TM6ICR). If the interrupt request flag may be already set, clear them. [☞ Chapter 3 3-1-4. Interrupt Flag Setup]
(6) Enable the interrupt. TM6ICR (x'3FEF') bp1 : TM6IE = 1	(6) Set the TM6IE flag of the TM6ICR register to "1" to enable the interrupt.

* the above steps (1), (2) can be set at once.

As TM6OC is set, TM6BC is initialized to x'00' to count up.
When TM6BC matches TM6OC, the timer 6 interrupt request flag is set to "1" at the next count clock and TM6BC is cleared to x'00' to restart counting.

If the TM6CLRS flag of the TM6MD register is set to "0", TM6BC can be initialized in every rewriting of TM6OC register, but in that state the timer 6 interrupt is disabled. If the timer 6 interrupt should be enabled, set the TM6CLRS flag to "1" after rewriting the TM6OC register.

On the timer 6 clock source selection, either the time base timer output or the time base timer synchronous output is selected, the clock setup of time base timer is needed.

8-4 Time Base Timer

8-4-1 Operation

■Time Base Timer (Time Base Timer)
The Interrupt is constantly generated.
Table 8-4-1 shows the interrupt generation cycle in combination with the clock source ;

Table 8-4-1 Time Base Timer Interrupt Generation Cycle

Selected clock source	Interrupt generation cycle	
fosc	fosc X $1/2^7$	6.4 µs
	fosc X $1/2^8$	12.8 µs
	fosc X $1/2^9$	25.6 µs
	fosc X $1/2^{10}$	51.2 µs
	fosc X $1/2^{13}$	409.6 µs
	fosc X $1/2^{15}$	1.64 ms
fx	fx X $1/2^7$	3.9 ms
	fx X $1/2^8$	7.8 ms
	fx X $1/2^9$	15.6 ms
	fx X $1/2^{10}$	31.2 ms
	fx X $1/2^{13}$	250 ms
	fx X $1/2^{15}$	1 s
fosc = 20(MHz) fx = 32.768(kHz)		

■Count Timing of Timer Operation (Time Base Timer)

The counter counts up with the selected clock source as a count clock.

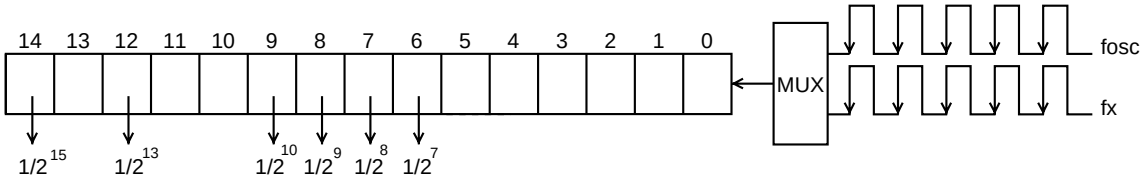


Figure 8-4-1 Count Timing of Timer Operation (Time Base Timer)

When the selected interrupt cycle has passed, the interrupt request flag of the time base interrupt control register (TBICR) is set to "1".

An interrupt may be generated at switching of the clock source. Enable interrupt after switching the clock source.

Time base timer cannot stop the operation.
The initialization can be done by writing an arbitrary value to the time base timer clear control register (TBCLR).

8-4-2 Setup Example

■Timer Operation Setup (Time Base Timer)

An interrupt can be generated constantly with time base timer in the selected interrupt cycle. The interrupt generation cycle is as fosc × $1/2^{13}$ (as 0.977 ms : fosc = 8.38 MHz) for generation interrupts.
An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Select the clock source. TM6MD (x'3F6A') bp0 : TM6CK0 = 0	(1) Select fosc as a clock source by the TM6CK0 flag of the timer 6 mode register (TM6MD).
(2) Select the interrupt generation cycle. TM6MD (x'3F6A') bp6-4 : TM6IR2-0 = 100	(2) Select the selected clock × $1/2^{13}$ as an interrupt generation cycle by the TM6IR2-0 flag of the TM6MD register.
(3) Initialize the time base timer. TBCLR (x'3F6B') = x'00'	(3) Write value to the time base timer clear control register (TBCLR) to initialize the time base timer. That makes the time base timer initialize.
(4) Set the interrupt level. TBICR (x'3FF0') bp7-6 : TBLV1-0 = 01	(4) Set the interrupt level by the TBLV1-0 flag of the time base interrupt control register (TBICR). If the interrupt request flag had already been set, clear it. [☞ Chapter 3 3-1-4. Interrupt Flag Setup]
(5) Enable the interrupt. TBICR (x'3FF0') bp1 : TBIE = 1	(5) Set the TBIE flag of the TBICR register to "1" to enable the interrupt.

* the above steps (1), (2) can be set at once.

When the selected interrupt generation cycle has passed, the interrupt request flag of the time base interrupt control register (TBICR) is set to "1".

Chapter 9 Watchdog Timer

9-1 Overview

This LSI has a watchdog timer. This timer is used to detect software processing errors. It is controlled by the watchdog timer control register (WDCTR). And, once an overflow of watchdog timer is generated, a watchdog interrupt (WDIRQ) is generated. If the watchdog interrupt is generated twice, consecutively, it is regarded to be an indication that the software cannot execute in the intended sequence; thus, a system reset is initiated by the hardware.

9-1-1 Block Diagram

■Watchdog Timer Block Diagram

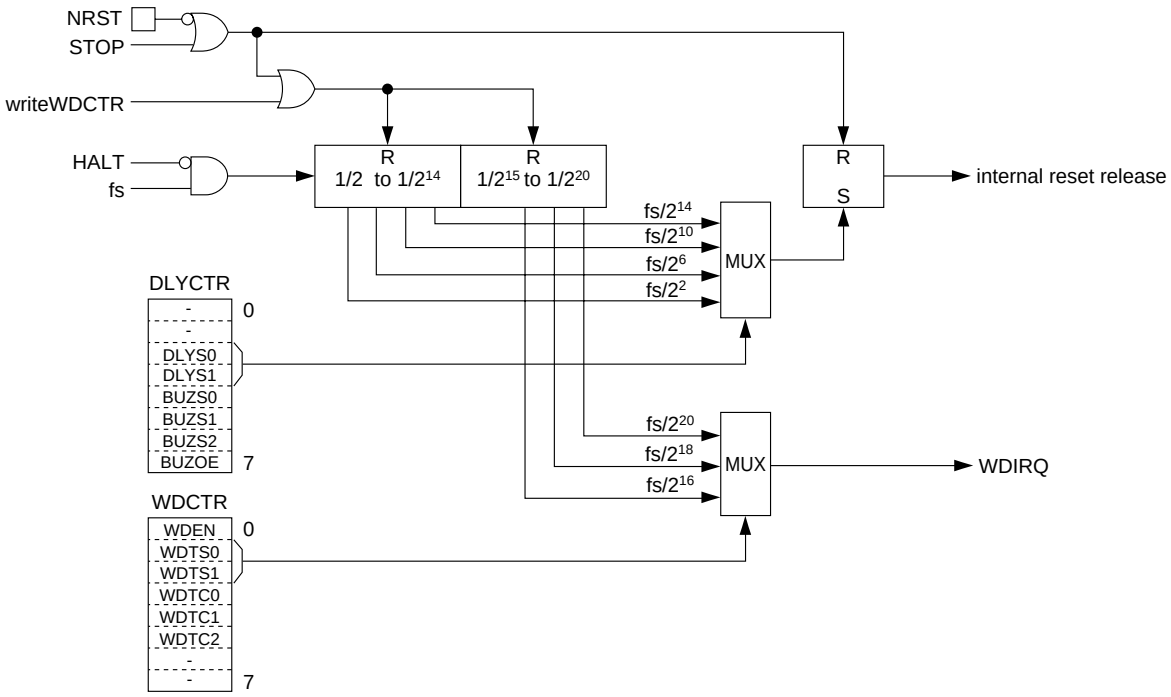


Figure 9-1-1 Block Diagram (Watchdog Timer)

The watchdog timer is also used as a timer to count the oscillation stabilization wait time. This is used as a watchdog timer except at recovering from STOP mode and at reset releasing.
The watchdog timer is initialized at reset or at STOP mode, and counts system clock (fs) as a clock source from the initial value (x'0000'). The oscillation stabilization wait time is set by the oscillation stabilization control register (DLYCTR). After the oscillation stabilization wait, counting is continued as a watchdog timer. [☞ Chapter 2 2-6. Reset]

9-2 Control Registers

The watchdog timer is controlled by the watchdog timer control register (WDCTR).

■Watchdog Timer Control Register (WDCTR)

	7	6	5	4	3	2	1	0
WDCTR	-	-	WDTC2	WDTC1	WDTC0	WDTS1	WDTS0	WDEN

(At reset : - - 0 0 0 1 1 0)

WDEN	Watchdog timer
0	Watchdog timer is disabled/cleared
1	Watchdog timer is enebled

WDTS1	WDTS0	Watchdog time-out period setup
0	0	2^{16} of system clock
	1	2^{18} of system clock
1	×	2^{20} of system clock

WDTC2	WDTC1	WDTC0	Watchdog timer can be cleared at the following cycle or more
0	0	0	can be cleared anytime
		1	2^{7} of system clock
	1	0	2^{9} of system clock
		1	2^{11} of system clock
1	0	0	2^{13} of system clock
		1	2^{15} of system clock
	1	0	2^{17} of system clock
		1	2^{19} of system clock

Figure 9-2-1 Watchdog Timer Control Register (WDCTR : x'03F02', R/W)

9-3 Operation

9-3-1 Operation

The watchdog timer counts system clock (fs) as a clock source. If the watchdog timer is overflowes, the watchdog interrupt (WDIRQ) is generated as an non maskable interrupt (NMI). At reset, the watchdog timer is stopped, but once the operation is enabled, it cannot be stopped except at reset. The watchdog timer control register (WDCTR) sets when the watchdog timer is released or how long the time-out period should be.

This watchdog timer can detect such that the watchdog timer clear is repeated in short cycle. If the watchdog timer clear is repeated in shorter cycle than the set time (the lowest value of watchdog timer clear possible), it is regarded as an error and the watchdog interrupt (WDIRQ) is generated.

If the watchdog interrupt (WDIRQ) is generated twice consecutively, it is regarded to be an indication that the software cannot execute in the intended sequence; thus, a system reset is initiated by the hardware.

The watchdog timer cannot stop, once it starts operation.

■Usage of Watchdog Timer

When the watchdog timer is used, constant clear in program is needed to prevent an overflow of the watchdog timer. As a result of the software failure, the software cannot execute in the intended sequence, thus the watchdog timer overflows and error is detected.

Programming of the watchdog timer is generally done in the last step of its programming.

■How to Detect Incorrect Code Execution

The watchdog timer is executed to be cleared in the certain cycle on the correct code execution. On this LSI, the watchdog timer detects errors when,

(1) the watchdog timer overflows.
(2) the watchdog timer clear happens in the shorter cycle than the watchdog timer clear possible lowest value, set in the watchdog timer control register (WDCTR).

When the watchdog timer detects any error, the watchdog interrupt (WDIRQ) is generated as a non maskable interrupt (NMI).

■How to Clear Watchdog Timer

The watchdog timer can be cleared by writing to the watchdog timer control register (WDCTR). The watchdog timer can be cleared regardless of the writing data to the register. The bit-set (BSET) that does not change the value is recommended.

■Watchdog Timer Period

The watchdog timer period is decided by the bp2, 1 (WDTS1-0) of the watchdog timer control register (WDCTR) and the system clock (fs). If the watchdog timer is not cleared till the set period of watchdog timer, that is regarded as an error and the watchdog interrupt (WDIRQ) of the non-maskable interrupt (NMI) is generated.

Table 9-3-1 Watchdog Timer Period

WDTS1	WDTS0	Watchdog time-out period
0	0	2^{16} X system clock
0	1	2^{18} X system clock
1	X	2^{20} X system clock

System clock is decided by the CPU mode control register (CPUM).

[☞ Chapter 2 2-5. Clock Switching]

The watchdog timer period is generally decided from the execution time for main routine of program. That should be set the longer period than the value of the execution time for main routine divided by natural number (1, 2, , ,). And insert the instruction of the watchdog timer clear to the main routine as that value makes the same cycle.

■The Lowest Value for Watchdog Timer Clear

The lowest value for watchdog timer clear is decided by the bp5, 4, 3 (WDTC2, WDTC1, WDTC0) of the watchdog timer control register (WDCTR).

Table 9-3-2 The Lowest Value for Watchdog Timer Clear

WDTC2	WDTC1	WATC0	Watchdog timer can be cleared at the following cycle or more
0	0	0	no limit
0	0	1	2^{7} X system clock
0	1	0	2^{9} X system clock
0	1	1	2^{11} X system clock
1	0	0	2^{13} X system clock
1	0	1	2^{15} X system clock
1	1	0	2^{17} X system clock
1	1	1	2^{19} X system clock

■Watchdog Timer and CPU Mode

The relation between this watchdog timer and CPU mode features are as follows ;

(1) In NORMAL, IDLE, SLOW mode, the system clock is counted.
(2) The counting is continued regardless of switching at NORMAL, IDLE, SLOW mode.
(3) In HALT mode, the watchdog timer is stopped.
(4) In STOP mode, the watchdog timer is cleared automatically by hardware.
(5) In STOP mode, the watchdog interrupt cannot be generated.
(6) After releasing reset or recovering from STOP, the counting is executed for the duration of the oscillation stabilization wait time.

Generally, in the system used STOP mode, if the STOP mode is done or not is divided on the program execution, but, in this case, the counting value of the watchdog timer differs. So, the watchdog interrupt should be prevented by setting the lowest value for watchdog timer clear.

9-3-2 Setup Example

The watchdog timer detects errors. On the following example, the watchdog timer period is set to $2^{18} \times$ system clock, the lowest value for watchdog timer clear is set to $2^9 \times$ system clock.
An example setup procedure, with a description of each step is shown below.

■Initial Setup Program (Watchdog Timer Initial Setup Example)

Setup Procedure	Description
(1) Set the time-out period. WDCTR (x'03F02') bp2-1 : WDTS1-0 = 01	(1) Set the WDTS1-0 flag of the watchdog timer control register (WDCTR) to "01" to select the time-out period to $2^{18} \times$ system clock.
(2) Set the lowest value for clear. WDCTR (x'03F02') bp5-3 : WDTC2-0 = 010	(2) Set the WDTC2-0 flag of the WDCTR register to "010" to select the lowest value for clear to $2^9 \times$ system clock.
(3) Start the watchdog timer operation. WDCTR (x'03F02') bp0 : WDEN = 1	(3) Set the WDEN flag of the WDCTR register to start the watchdog timer operation.

The command of setting the WDEN flag to "1" should be done on the last step of the initial setting. If the watchdog control register (WDCTR) is changed after starting the operation, the watchdog interrupt may be generated depending on the setting of the lowest value for clear.

■Main Routine Program (Watchdog Timer Constant Clear Setup Example)

Setup Procedure	Description
(1) Set the constant watchdog timer clear. Writing to WDCTR (x'03F02') (cf.) BSET (WDCTR) WDEN (bp0 : WDEN = 1)	(1) Clear the watchdog timer by the cycle from $2^9 \times$ system clock up to $2^{18} \times$ system clock. The watchdog timer clear should be inserted in the main routine, with the same cycle, and to be the set cycle. The recommended instruction is the bit-set (BSET), does not change value, for clear.

■Interrupt Service Routine Setup

Setup Procedure	Description
(1) Set the watchdog interrupt service routine. NMICR (x'03FE1') TBNZ (NMICR) WDIR, WDPRO	(1) If the watchdog timer overflows, the non maskable interrupt is generated. Confirm that the WDIR flag of the non maskable interrupt control register (NMICR) is "1" on the interrupt service routine, and manage the suitable execution.

The operation, just before the WDOG interrupt may be executed wrongly. Therefore, if the WDOG interrupt is generated, initialize the system.

Chapter 10 Buzzer

10

10-1 Overview

This LSI has a buzzer. It can output the square wave, having a frequency $1/2^9$ to $1/2^{14}$ of the high speed oscillation clock, or by $1/2^3$ to $1/2^4$ of the low speed oscillation clock.

10-1-1 Block Diagram

■Buzzer Block Diagram

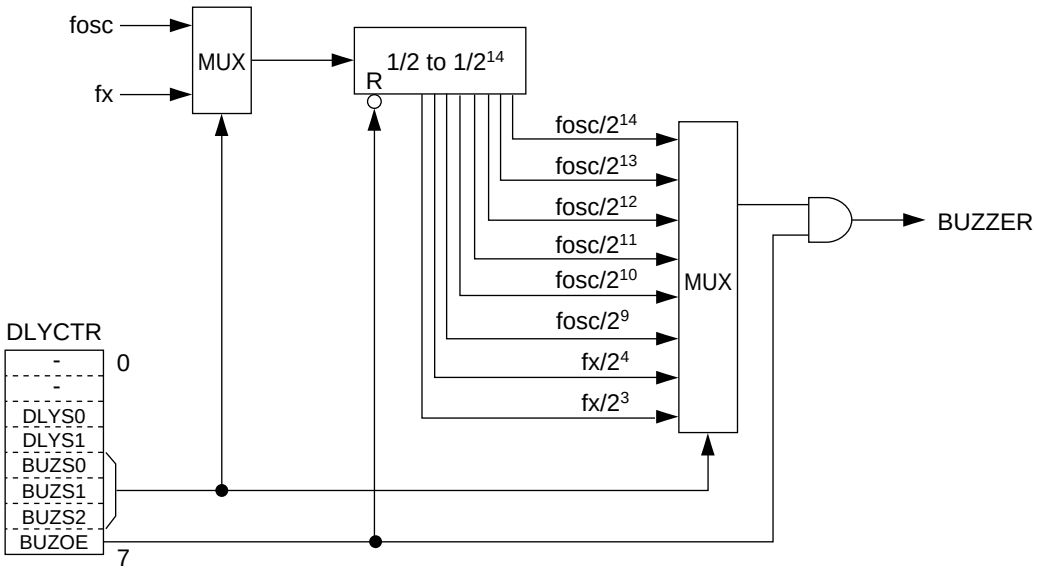
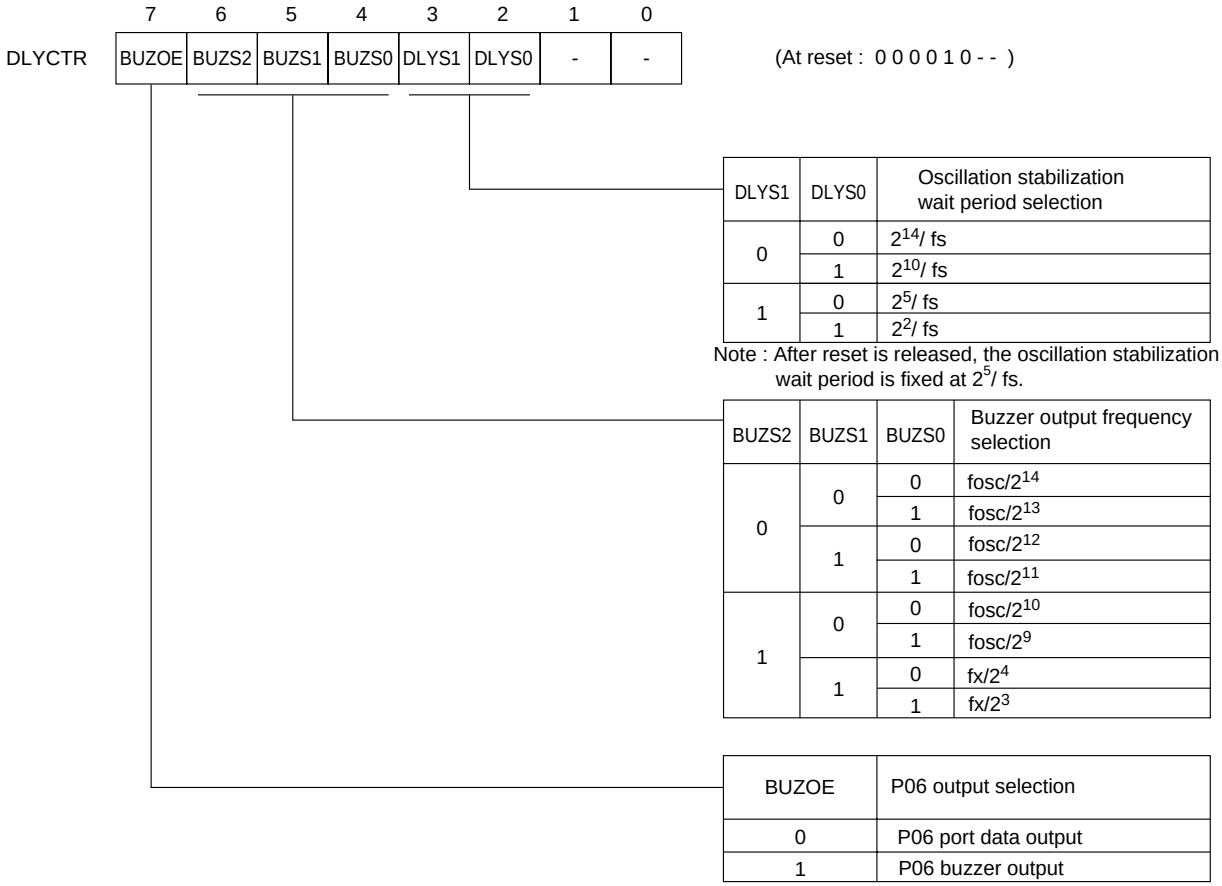


Figure 10-1-1 Block Diagram (Buzzer)

10-2 Control Register

■Oscillation Stabilization Wait Timer Control Register



DLYS1	DLYS0	Oscillation stabilization wait period selection
0	0	2^{14}/ fs
	1	2^{10}/ fs
1	0	2^{5}/ fs
	1	2^{2}/ fs

Note : After reset is released, the oscillation stabilization wait period is fixed at 2^{5}/ fs.

BUZS2	BUZS1	BUZS0	Buzzer output frequency selection
0	0	0	fosc/2^{14}
		1	fosc/2^{13}
	1	0	fosc/2^{12}
		1	fosc/2^{11}
1	0	0	fosc/2^{10}
		1	fosc/2^{9}
	1	0	fx/2^{4}
		1	fx/2^{3}

BUZOE	P06 output selection
0	P06 port data output
1	P06 buzzer output

Figure 10-2-1 Oscillation Stabilization Wait Time Control Register (DLYCTR : x'03F03', R/W)

10-3 Operation

10-3-1 Operation

■Buzzer

Buzzer outputs the square wave, having a frequency $1/2^9$ to $1/2^{14}$ of the high speed oscillation clock (fosc), or by $1/2^3$ to $1/2^4$ of the low speed oscillation clock (fx). The BUZS 2, 1, 0 flag of the oscillation stabilization wait control register (DLYCTR) set the frequency of buzzer output. The BUZOE flag of the oscillation stabilization wait control register (DLYCTR) sets buzzer output ON / OFF.

■Buzzer Output Frequency

The frequency of buzzer output is decided by the frequency of the high oscillation clock (fosc) or the low oscillation clock (fx) and the bit 6, 5, 4 (BUZS2, BUZS1, BUZS0) of the oscillation stabilization wait control register (DLYCTR).

Table 10-3-1 Buzzer Output Frequency

fosc	fx	BUZS2	BUZS1	BUZS0	Buzzer output frequency
20 MHz	-	0	0	0	1.22 kHz
20 MHz	-	0	0	1	2.44 kHz
20 MHz	-	0	1	0	4.88 kHz
8.38 MHz	-	0	1	0	2.05 kHz
8.38 MHz	-	0	1	1	4.09 kHz
2 MHz	-	1	0	0	1.95 kHz
2 MHz	-	1	0	1	3.91 kHz
-	32 kHz	1	1	0	2 kHz
-	32 kHz	1	1	1	4 kHz

10-3-2 Setup Example

Buzzer outputs the square wave of 2 kHz from P06 pin. It is used 8.38 MHz as the high oscillation clock (fosc).
An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Set the buzzer frequency. DLYCTR (x'3F03') bp6-4 : BUZS2-0 = 010	(1) Set the BUZS2-0 flag of the oscillation stabilization wait control register (DLYCTR) to "010" to select fosc/2^{12} to the buzzer frequency. When the high oscillation clock fosc is 8.38 MHz, the buzzer output frequency is 2.05 kHz.
(2) Set P06 pin. P0OUT (x'3F10') bp6 : P0OUT6 = 0 P0DIR (x'3F30') bp6 : P0DIR6 = 1	(2) Set the output data P0OUT6 of P06 pin to "0", and set the direction control P0DIR6 of P06 pin to "1" to select output mode. Port 06 pin outputs low level.
(3) Buzzer output ON. DLYCTR (x'3F03') bp7 : BUZOE = 1	(3) Set the BUZOE flag of the oscillation stabilization wait control register (DLYCTR) to "1" to output the square wave of the buzzer output frequency set by P06 pin.
(4) Buzzer output OFF. DLYCTR (x'3F03') bp7 : BUZOE = 0	(4) Set the BUZOE flag of the oscillation stabilization wait control register (DLYCTR) to "0" to clear, and P06 pin outputs low level.

Chapter 11 Serial Interface 0

11-1 Overview

This LSI contains a serial interface 0 that can be used for both communication types of clock synchronous and UART (duplex).

11-1-1 Functions

Table 11-1-1 shows functions of serial interface 0.

Table 11-1-1 Serial Interface 0 Functions

Communication style	clock synchronous	UART (duplex)
Interrupt	SC0TIRQ	SC0TIRQ (on transmission completion) SC0RIRQ (on reception completion)
Used pins	SBO0,SBI0,SBT0	TXD0,RXD0
3 channels type	√	-
2 channels type	√ (SBO0, SBT0)	√
1 channel type	-	√ (TXD0)
Specification of transfer bit count / Frame selection	1 to 8 bits	7 bits + 1stop 7 bits + 2stops 8 bits + 1stop 8 bits + 2stops
Selection of parity bit	-	√
Parity bit control	-	0 parity 1 parity odd parity even parity
Selection of start condition	√	only "enable start condition" is available
Specification of the first transfer bit	√	√
Specification of input edge / output edge	√	-
SBO0 output control after the transmission of last data	hold H/L/last data	-
Communicative function at STANDBY mode	only slave reception is available	-
Continuous operation	√	√
Internal clock 1/8 dividing	√	only 1/8 dividing is available
Clock source	fosc/2 fosc/4 fosc/16 fosc/64 fs/2 fs/4 Timer 2 output Timer 3 output External clock	fosc/2 fosc/4 fosc/16 fosc/64 fs/2 fs/4 Timer 2 output Timer 3 output
Maximum transfer rate	2.5 MHz	300 kbps (standard 300 bps to 38.4 kbps) (timer 4 output)
fosc : Machine clock (High speed oscillation) fs : System clock [☞ Chapter 2 2-5. Clock Switching]		

11-1-2 Block Diagram

■Serial Interface 0 Block Diagram

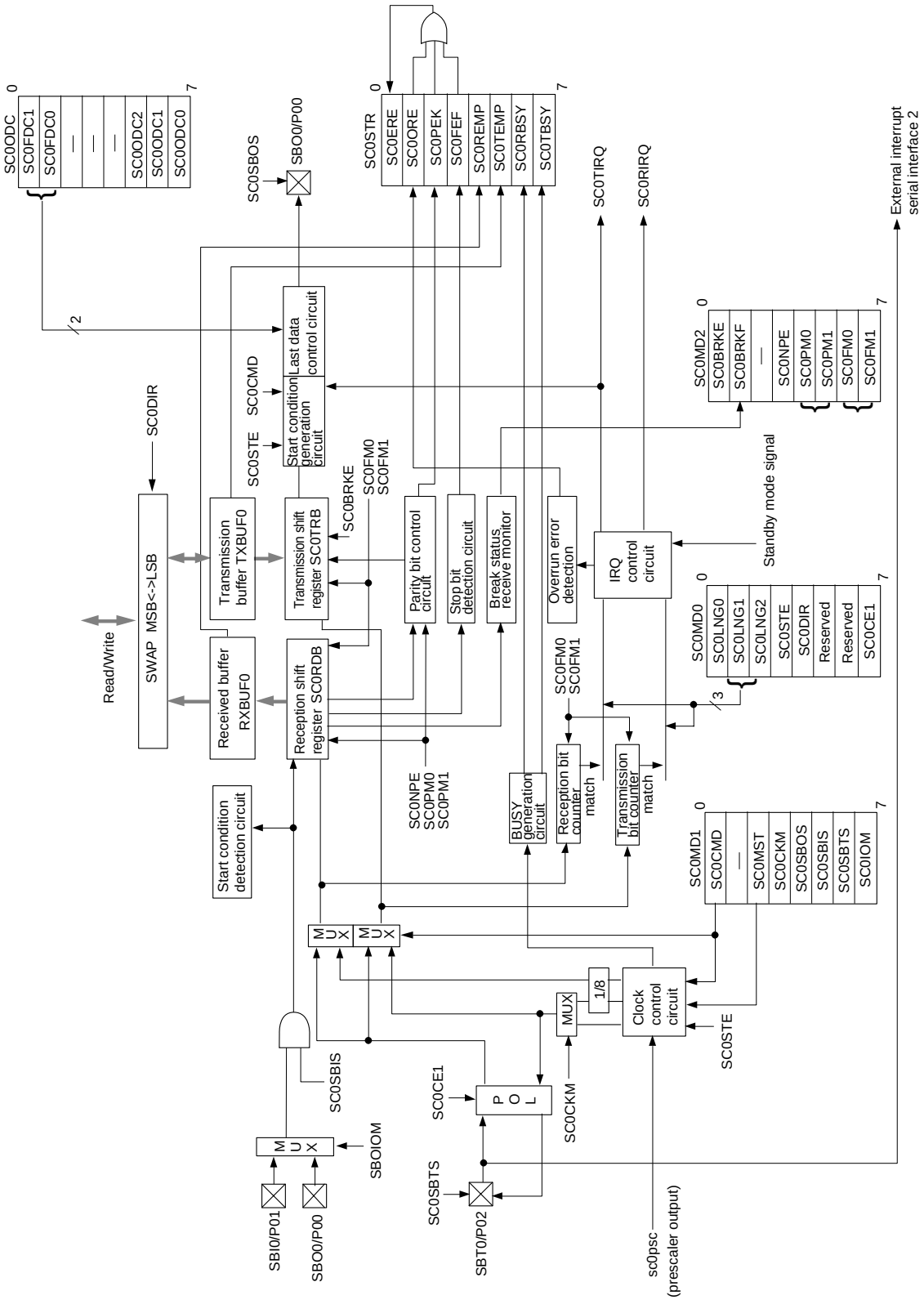


Figure 11-1-1 Serial Interface 0 Block Diagram

11-2 Control Registers

11-2-1 Registers

Table 11-2-1 shows registers to control serial interface 0.

Table 11-2-1 Serial Interface 0 Control Registers

	Register	Address	R/W	Function	Page
Serial interface 0	SC0MD0	x'03F90'	R/W	Serial interface 0 mode register 0	XI - 6
	SC0MD1	x'03F91'	R/W	Serial interface 0 mode register 1	XI - 7
	SC0MD2	x'03F92'	R/W	Serial interface 0 mode register 2	XI - 8
	SC0STR	x'03F93'	R	Serial interface 0 state register	XI - 9
	RXBUF0	x'03F94'	R	Serial interface 0 reception data buffer	XI - 5
	TXBUF0	x'03F95'	R/W	Serial interface 0 transmission data buffer	XI - 5
	SC0ODC	x'03F96'	R/W	Serial interface 0 port control register	XI - 10
	SC0CKS	x'03F97'	R/W	Serial interface 0 transfer clock selection register	XI - 11
	PSCMD	x'03F6F'	R/W	Prescaler control register	V - 6
	P0DIR	x'03F30'	R/W	Port 0 direction control register	IV - 7
	P0PLU	x'03F40'	R/W	Port 0 pull-up control register	IV - 7
	SC0RICR	x'03FF5'	R/W	Serial interface 0 UART reception interrupt control register	III - 27
	SC0TICR	x'03FF6'	R/W	Serial interface 0 interrupt control register	III - 28

R/W : Readable / Writable
R : Readable only

11-2-2 Data Buffer Registers

Serial Interface 0 has each 8-bit data buffer register for transmission, and for reception.

■Serial Interface 0 Reception Data Buffer (RXBUF0)

	7	6	5	4	3	2	1	0	
RXBUF0	RXBUF07	RXBUF06	RXBUF05	RXBUF04	RXBUF03	RXBUF02	RXBUF01	RXBUF00	(At reset : X X X X X X X X)

Figure 11-2-1 Serial Interface 0 Reception Data Buffer (RXBUF0 : x'03F94', R)

■Serial Interface 0 Transmissin Data Buffer (TXBUF0)

	7	6	5	4	3	2	1	0	
TXBUF0	TXBUF07	TXBUF06	TXBUF05	TXBUF04	TXBUF03	TXBUF02	TXBUF01	TXBUF00	(At reset : X X X X X X X X)

Figure 11-2-2 Serial Interface 0 Transmission Data Buffer (TXBUF0 : x'03F95', R/W)

11-2-3 Mode Registers

■Serial Interface 0 Mode Register 0 (SC0MD0)

	7	6	5	4	3	2	1	0
SC0MD0	SC0CE1	RESERVED	RESERVED	SC0DIR	SC0STE	SC0LNG2	SC0LNG1	SC0LNG0

(At reset : 0 0 0 0 0 1 1 1)

SC0LNG2	SC0LNG1	SC0LNG0	Synchronous serial transfer bit count
0	0	0	1 bit
		1	2 bits
	1	0	3 bits
		1	4 bits
1	0	0	5 bits
		1	6 bits
	1	0	7 bits
		1	8 bits

SC0STE	Synchronous serial transfer start condition
0	Disable start condition
1	Enable start condition

SC0DIR	First bit to be transferred
0	MSB first
1	LSB first

RESERVED	Set always to "0".

RESERVED	Set always to "0".

SC1CE1	Transmission data output edge	Reception data input edge
0	falling	rising
1	rising	falling

Figure 11-2-3 Serial Interface 0 Mode Register 0 (SC0MD0 : x'03F90', R/W)

■Serial Interface 0 Mode Register 1 (SC0MD1)

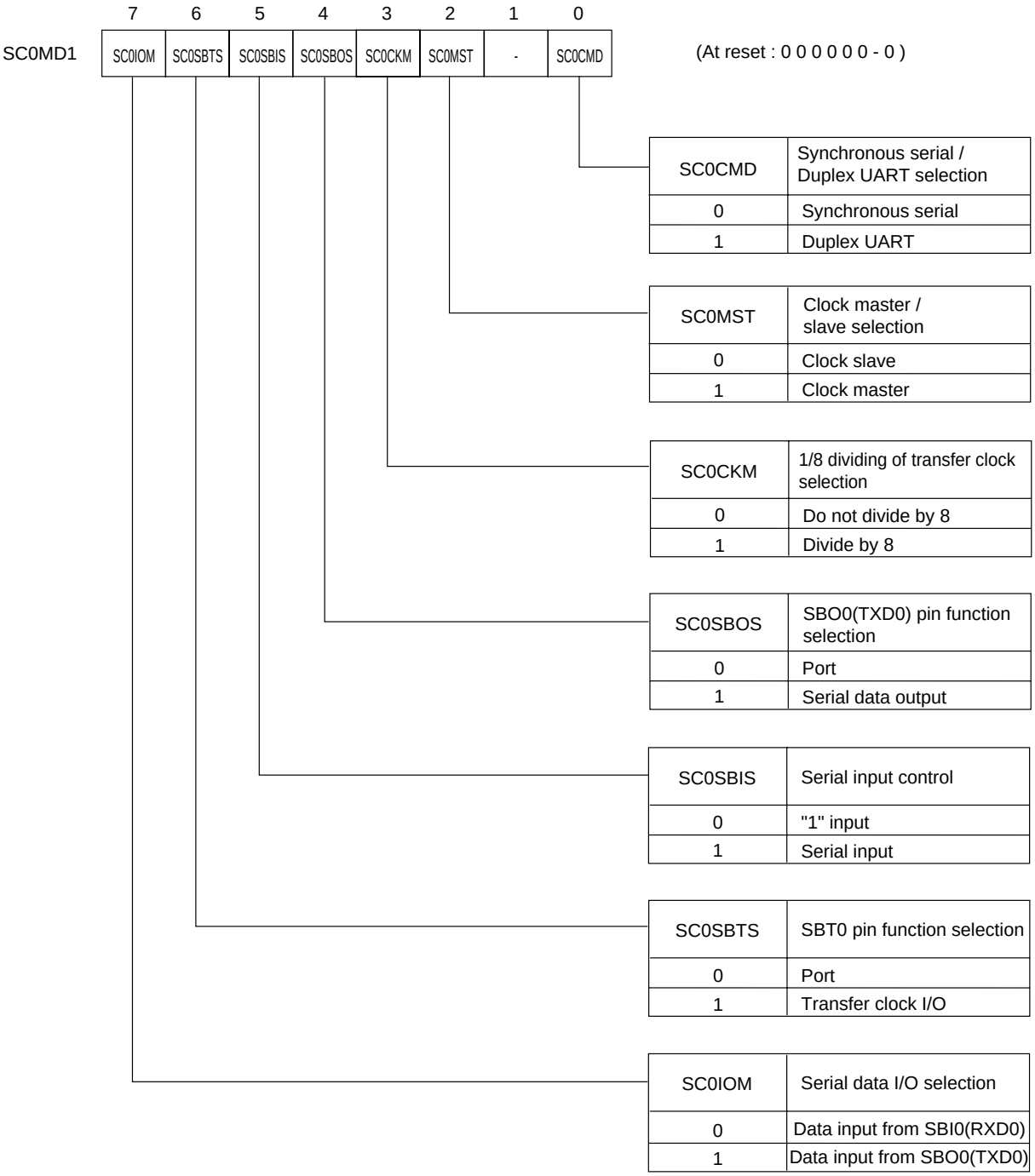
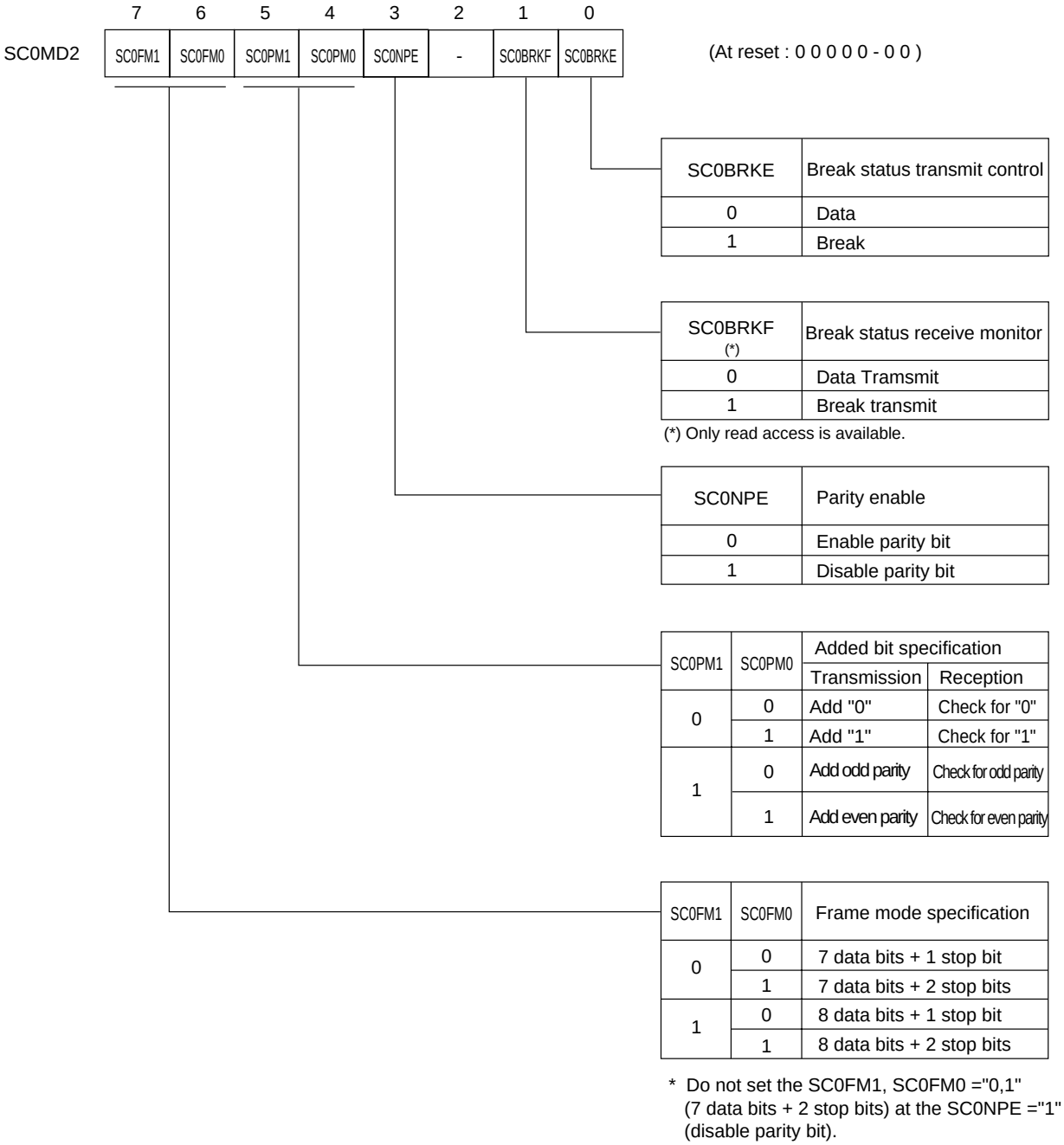


Figure 11-2-4 Serial Interface 0 Mode Register 1 (SC0MD1 : x'03F91', R/W)

■Serial Interface 0 Mode Register 2 (SC0MD2)

SC0BRKF flag is only for reading.



SC0BRKE	Break status transmit control
0	Data
1	Break

SC0BRKF (*)	Break status receive monitor
0	Data Tramsmit
1	Break transmit

(*) Only read access is available.

SC0NPE	Parity enable
0	Enable parity bit
1	Disable parity bit

SC0PM1	SC0PM0	Added bit specification	
		Transmission	Reception
0	0	Add "0"	Check for "0"
	1	Add "1"	Check for "1"
1	0	Add odd parity	Check for odd parity
	1	Add even parity	Check for even parity

SC0FM1	SC0FM0	Frame mode specification
0	0	7 data bits + 1 stop bit
	1	7 data bits + 2 stop bits
1	0	8 data bits + 1 stop bit
	1	8 data bits + 2 stop bits

* Do not set the SC0FM1, SC0FM0 ="0,1" (7 data bits + 2 stop bits) at the SC0NPE ="1" (disable parity bit).

Figure 11-2-5 Serial Interface 0 Mode Register 2 (SC0MD2 : x'03F92', R/W)

■Serial Interface 0 State Register (SC0STR)

All flags are only for reading.

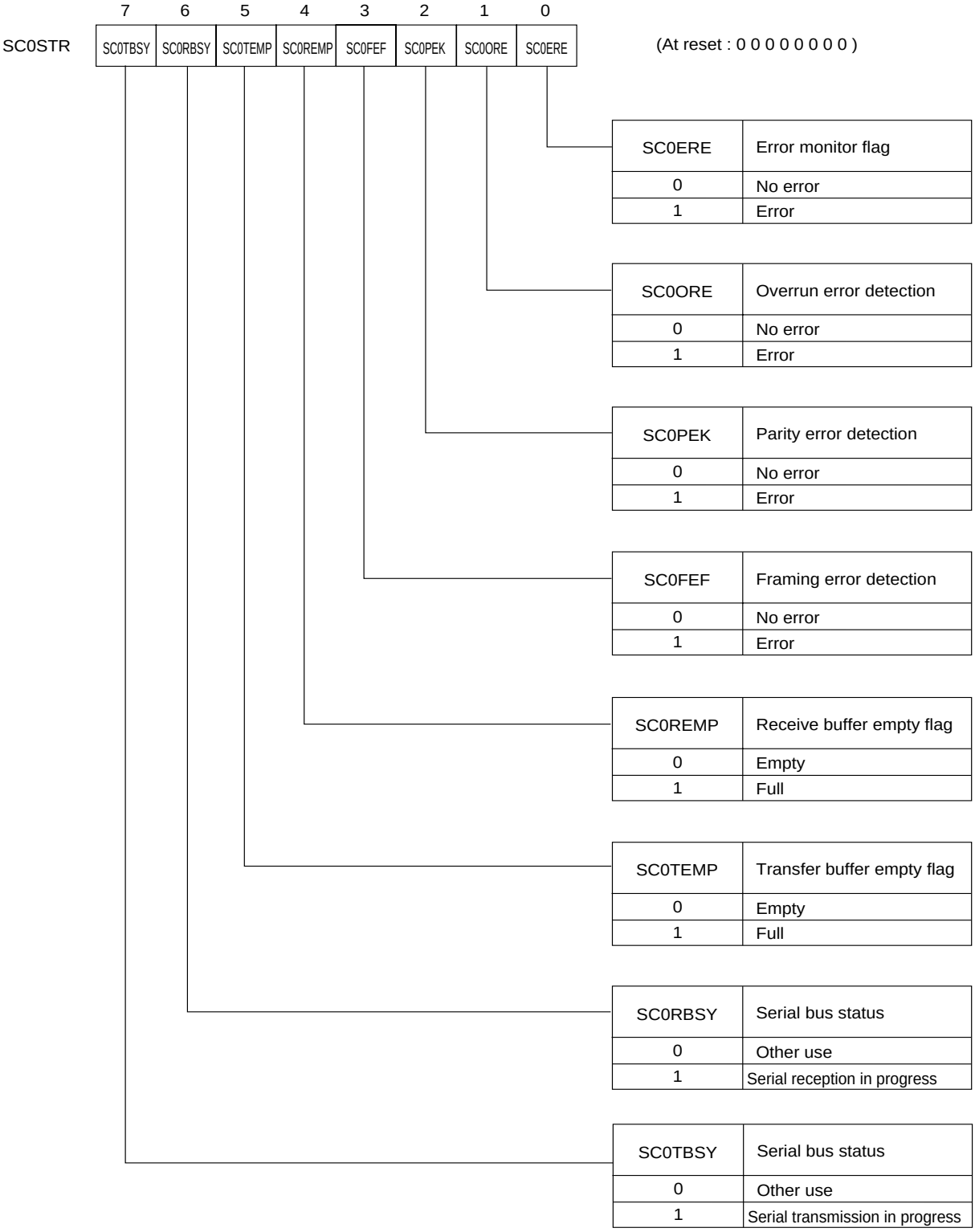


Figure 11-2-6 Serial Interface 0 State Register (SC0STR : x'03F93', R)

■Serial Interface 0 Port Control Register (SC0ODC)

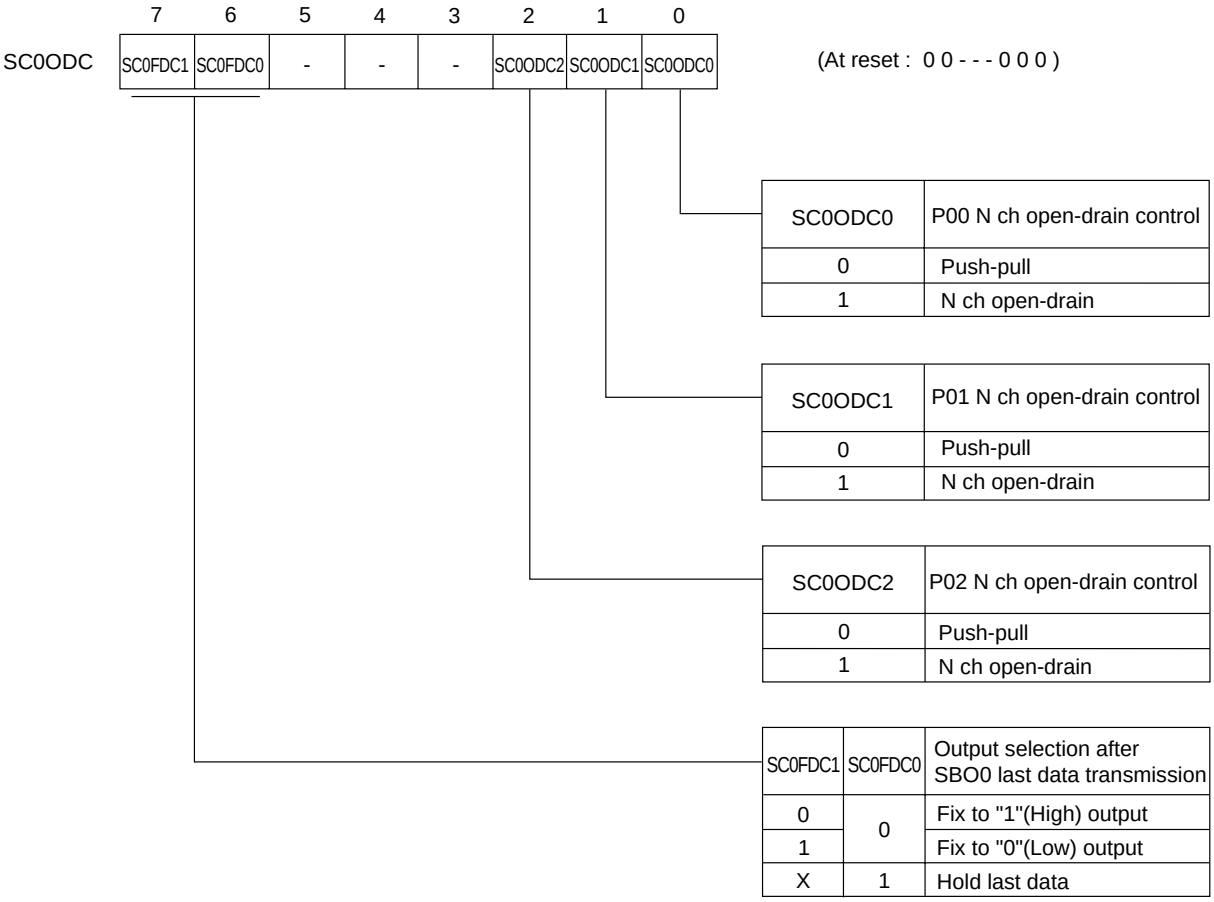


Figure 11-2-7 Serial Interface 0 Port Control Register (SC0ODC : x'03F96', R/W)

■Serial Interface 0 Transfer Clock Selection Register (SC0CKS)

	7	6	5	4	3	2	1	0
SC0CKS	-	-	-	-	SC0TMSEL	SC0PSC2	SC0PSC1	SC0PSC0

(At reset: - - - - X X X X)

SC0TMSEL	SC0PSC2	SC0PSC1	SC0PSC0	Selected clock
-	0	0	0	fosc/2
			1	fosc/4
		1	0	fosc/16
			1	fosc/64
	1	0	0	fs/2
			1	fs/4
0		1	-	Timer 2 output
1			-	Timer 3 output

Figure 11-2-8 Serial Interface 0 Tranfer Clock Selection Register (SC0CKS : x'03F97', R/W)

When timer output is selected as serial interface transfer clock, select fosc as a clock source of the timer. If other clock is selected, normal transfer of serial interface data is not guaranteed.

11-3 Operation

Serial Interface 0 can be used for both clock synchronous and duplex UART.

11-3-1 Clock Synchronous Serial Interface

■Activation Factor for Communication

Table 11-3-1 shows activation factors for communication. At master, the transfer clock is generated by setting data to the transmission data buffer TXBUF0, or by receiving a start condition. Except during communication, the input signal from SBT0 pin is masked to prevent errors by noise or so. This mask can be released automatically by setting a data to TXBUF0(access to the TXBUF0 register), or by inputting a start condition to the data input pin. Therefore, at slave, set data to TXBUF0, or input an external clock after a start condition is input.

Table 11-3-1 Synchronous Serial Interface Activation Factor

	Activation factor	
	Transmission	Reception
at master	Set transmission data	Set dummy data
		Input start condition
at slave	Input clock after transmission data is set	Input clock after dummy data is set
		Input clock after start condition is input

■Transfer Bit Setup

The transfer bit count is selected from 1 bit to 8 bits. Set them by the SC0LNG 2 to 0 flag of the SC0MD0 register (at reset : 111). The SC0LNG 2 to 0 flag holds the former set value until it is set again.

Except during communication, SBT0 pin is masked to prevent errors by noise. At slave communication, set data to TXBUF0 or input a clock to SBT0 pin after a start condition is input.

■Start Condition Setup

The SC0STE flag of the SC0MD0 register sets if a start condition is enabled or not. If a start condition is enabled, and received at communication, a bit counter is cleared to restart the communication. The start condition, if the SC0CE1 flag of the SC0MD0 register is set to "0", is regarded when a data line (SBI0 pin (with 3 channels) or SBO0 pin (with 2 channels) is changed from "H" to "L" as a clock line (SBT0 pin) is "H". Also, the start condition, if the SC0CE1 flag of the SC0MD0 register is set to "1", is regarded when a data line (SBI0 pin (with 3 channels) or SBO0 pin (with 2 channels) is changed from "H" to "L" as a clock line (SBT0 pin) is "L". Both the SC0SBOS flag and the SC0SBIS flag of the SC0MD1 register should be set to "0", before the start condition setup is changed

■First Transfer Bit Setup

The SC0DIR flag of the SC0MD0 register can set the first transfer bit. MSB first or LSB first can be selected.

■Transmission Data Buffer

The transmission data buffer, TXBUF0 is the sub buffer that stores data to load the internal shift register. Data to be transfered should be set to the transmission data buffer, TXBUF0 to load to the internal shift register automatically. The first data loading to the internal shift register is done at the same timing of the data setting to TXBUF0.

■Received Data Buffer

The received data buffer RXBUF0 is the sub buffer that pushed the received data in the internal shift register. After the communication complete interrupt SC0IRQ is generated, data stored in the internal shift register is stored to the received data buffer RXBUF0 automatically. RXBUF0 can store data up to 1 byte. RXBUF0 is rewritten in every communication complete, so read out data of RXBUF0 till the next receive complete. The received data buffer empty flag SC0REMP is set to "1" at the same time SC0TIRQ is generated. SC0REMP is cleared to "0" after RXBUF0 is read.

If a start condition is input to restart during communication, the transmission data is not valid. If the transmission should be operated again, set the transmission data to TXBUF0, again.

Start condition should be switched after both the SC0SBOS and the SC0SBIS flags of the SC0MD1 register are set to "0". If they are not set to "0", the switching is not valid.

RXBUF0 is rewritten in every communication complete. At continuous communication, data of RXBUF0 should be read out till the next reception complete.

■Tranfer Bit Count and First Transfer Bit

When the transfer bit is 1 bit to 7 bits, the data storing method to the transmission data buffer TXBUF0 is different, depending on the first transfer bit selection. At MSB first, use the upper bits of TXBUF0 for storing. When there are 6 bits to be transfered, as shown on figure 11-3-1, if data "A" to "F" are stored to bp2 to bp7 of TXBUF0, the transmission is operated from "F" to "A". At LSB first, use the lower bits of TXBUF0 for storing. When there are 6 bits to be transfered, as shown on figure 11-3-2, if data "A" to "F" are stored to bp0 to bp5 of TXBUF0, the transmission is operated from "A" to "F".

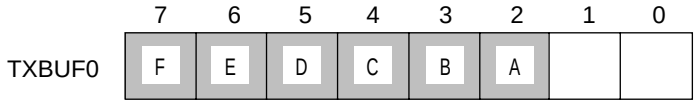


Figure 11-3-1 Transfer Bit Count and First Transfer Bit (starting with MSB)

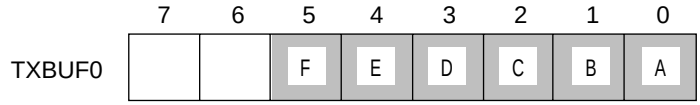


Figure 11-3-2 Transfer Bit Count and First Transfer Bit (starting with LSB)

■Receive Bit Count and First Transfer Bit

When the transfer bit count is 1 bit to 7 bits, the data storing method to the received data buffer RXBUF0 is different depending on the first transfer bit selection. At MSB first, data are stored to the lower bits of RXBUF0. When there are 6 bits to be transfered, as shown on figure 11-3-3, if data "F" to "A" are stored to bp0 to bp5 of RXBUF0. At LSB first, data are stored to the upper bits of RXBUF0. When there are 6 bits to be transfered, as shown on figure 11-3-4, if data "A" to "F" are stored to bp2 to bp7 of RXBUF0.

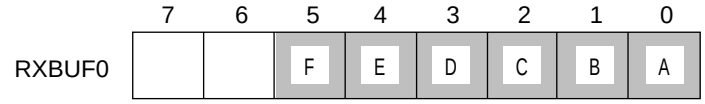


Figure 11-3-3 Receive Bit Count and Transfer First Bit (starting with MSB bit)

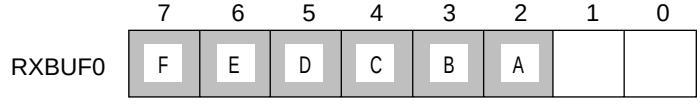


Figure 11-3-4 Receive Bit Count and Transfer First Bit (starting with LSB bit)

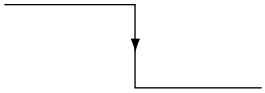
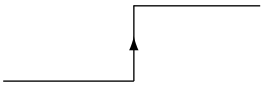
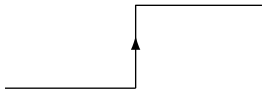
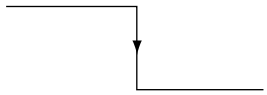
■Continuous Communication

This serial has a function for continuous communication. If data is set to the transmission data buffer TXBUF0 during communication, the transmission buffer empty flag SC0TEMP is automatically set to communicate continuously. Data setup to TXBUF0 should be done till the communication complete interrupt SC0IRQ is generated after the former data is set. At master communication, there is a suspension of communication for 3 transfer clocks till the next transmission clock is output after the SC0IRQ generation.

■Input Edge / Output Edge Setup

The SC0CE 1 to 0 flag of the SC0MD0 register set an output edge of the transmission data, an input edge of the received data. As the SC0CE1 flag = "0", the transmission data is output at the falling edge, and as "1", output at the rising edge. As SC0CE1="0", the received data is received at the inversion edge to the output edge of transmission data, and as "1", stored at the same edge.

Table 11-3-2 Transmission Data Output Edge and Received Data Input Edge

SC0CE1	Transmission data output edge	Received data input edge
0		
1		

■Clock Setup

The SC0CKS register selects a clock source from the special prescaler and timer output. The special prescaler starts its operation after the PSCMD (x'03F6F') register selects "prescaler operation". The SC0MST flag of the SC0MD1 register can select the internal clock (clock master), or the external clock (clock slave). Even if the external clock is selected, set the internal clock that has the same clock cycle or below to the external clock, by the SC0CKS register. That is happened, because the interrupt flag SC0TIRQ is generated by the internal clock. Here is the internal clock source that can be set by the SC0CKS register. Also, the SC0CKM flag of the SC0MD1 register can divide the internal clock by 8.

Table 11-3-3 Synchronous Serial Interface Internal Clock Source

Internal clock	fosc/2
	fosc/4
	fosc/16
	fosc/64
	fs/2
	fs/4
	Timer 2 output
	Timer 3 output

When timer output is selected as serial interface transfer clock, select fosc as a clock source of the timer. If other clock is selected, normal transfer of serial interface data is not guaranteed.

■Data Input Pin Setup

3 channels type (clock pin (SBT0 pin), data output pin (SBO0 pin), data input pin (SBI0 pin)) or 2 channels type (clock pin (SBT0 pin), data I/O pin (SBO0 pin)) can be selected as the communication. SBI0 pin can be used for only serial data input. SBO0 pin can be used for serial data input or output. The SC0IOM flag of the SC0MD1 register can select if the serial data is input from SBI0 pin or SBO0 pin. When "data input from SBO0 pin" is selected to set the 2 channels type, the P0DIR0 flag of the P0DIR register controls direction of SBO0 pin to switch transmission / reception. At that time, SBI0 pin is free to be used as a general port.

The transfer speed should be up to 2.5 MHz. If the transfer clock is over 2.5 MHz, the transmission data may not be sent correctly.

At reception, if SC0IOM of the SC0MD1 register is set to "1" and "serial data input from SBO0" is selected, SBI0 pin is used as a general port.

■Received Buffer Empty Flag

When the reception is completed (the last data reception edge of the clock is input), data is stored to RXBUF0 from the internal shift register, automatically. If data is stored to the shift register RXBUF0, the received buffer empty flag SC0REMP of the SC0STR register is set to "1". That indicates that the received data is going to be read. SC0REMP is cleared to "0" by reading out the data of RXBUF0.

■Transmission Buffer Empty Flag

If any data is set to TXBUF0 again, during communication (after setting data to TXBUF0 before generating the communication complete interrupt SC0IRQ), the transmission buffer empty flag SC0TEMP of the SC0STR register is set to "1". That indicates that the next transmission data is going to load. Data is loaded to the inside shift register from TXBUF0 by generation of SC0TIRQ, and the next transfer is started as SC0TEMP is cleared to "0".

■Overrun Error and Error Monitor Flag

If, after reception complete, the next data has been already received before reading out the data of the received data buffer RXBUF0, overrun error is generated and the SC0ORE flag of the SC0STR register is set to "1". And at the same time, the error monitor flag SC0ERE is set to indicate that something wrong on reception. The SC0ORE flag holds the status unless the data of RXBUF0 is read out. SC0ERE is cleared as SC0ORE flag is cleared. These error flags are nothing to do with communication operation.

■Reception BUSY Flag

When any data is set to TXBUF0 or when the SC0SBIS flag of the SC0MD1 register is "1" as start condition is input, the SC0RBSY flag of the SC0STR register is set to "1". And, on the generation of the communication complete interrupt SC0TIRQ, the flag is cleared to "0". And, during continuous communication, the SC0RBSY falg is always set. If the transmission buffer empty flag SC0TEMP is cleared to "0" as the communication complete interrupt SC0TIRQ is generated, SC0RBSY is cleared to "0". If the SC0SBIS flag is set to "0" during communication, the SC0RBSY flag is cleared to "0".

■Transmission BUSY Flag

When any data is set to TXBUF0 or when the SC0SBOS flag of the SC0MD1 register is "1" as start condition is input, the SC0TBSY flag of the SC0STR register is set to "1". And, on the generation of the communication complete interrupt SC0TIRQ, the flag is cleared to "0". And, during continuous communication, the SC0TBSY flag is always set. If the transmission buffer empty flag SC0TEMP is cleared to "0" as the communication complete interrupt SC0TIRQ is generated, SC0TBSY is cleared to "0". If the SC0SBOS flag is set to "0" during communication, the SC0TBSY flag is cleared to "0".

■Emergency Reset

It is possible to shut down communication. For a forced reset, the SC0SBOS flag and the SC0SBIOS flag of the SC0MD1 register should be set to "0" (SBO0 pin : port, input data : "1" input). At forced reset, the status registers (the SC0BRKF flag of the SC0MD2 register, all flags of the SC0STR register) are initialized as they are set at reset, but the control register holds the setting value.

■Last Bit of Transfer Data

Table 11-3-4 shows the data output holding period of the last bit at transmission, and the minimum data input period of the last bit at reception.

Table 11-3-4 Last Bit Data Length of Transfer Data

	The last bit data holding period at transmission	The last data input period at reception
At master	1 bit data length	1 bit data length (Minimum)
At slave	[1 bit data length of external clock x 1/2] + [Internal clock frequency x (1/2 to1)]	

After the last bit data holding period at transmission, SBO0 output without start condition (SC0STE flag=0) can be set as table 11-3-5 with SC0FDC1-0 flag of the SC0ODC register.
After reset release, its output becomes "H" regardless of the set value of the SC0FDC1-0 flag.
When start condition is added to it, it outputs "H" regardless of the set value of the SC0FDC1-0 flag.

Table 11-3-5 The last bit data holding period at transmission, SBO0 output without start condition (SC0STE flag=0)

SC0FDC1 flag	SC0FDC0 flag	The last bit data holding period at transmission, SBO0 output
0	0	Fixed to "1"(High) output
1	0	Fixed to "0"(Low) output
X	1	Last bit data holding

■Other Control Flag Setup

Table 11-3-6 shows flags that are not used at clock synchronous communication. So, they are not needed to set or monitor.

Table 11-3-6 Other Control Flag

Register	Flag	Detail
SC0MD2	SC0BRKF	Brake status reception monitor
	SC0NPE	Parity is enabled
	SC0PM1 to 0	Added bit specification
	SC0FM1 to 0	Frame mode specification
SC0STR	SC0PEK	Parity error detection
	SC0FEF	Frame error detection

■Trasnmission Timing

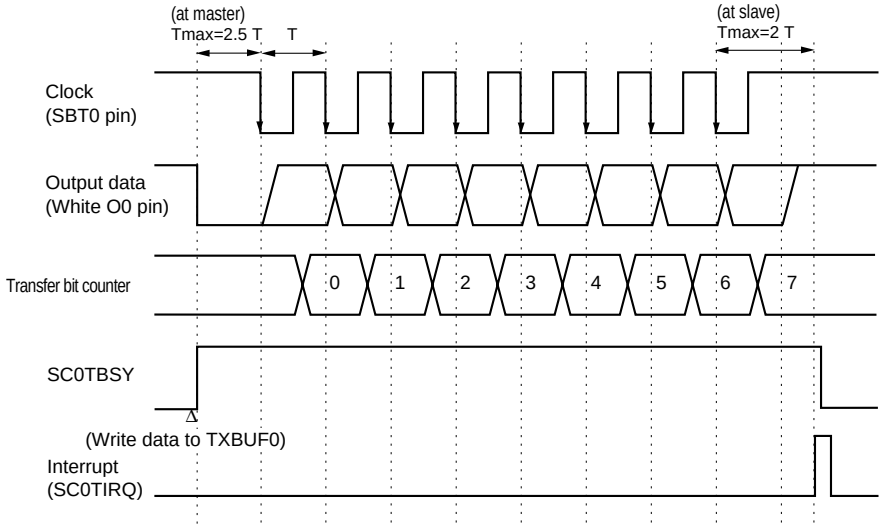


Figure 11-3-5 Transmission Timing (falling edge, start condition is enabled)

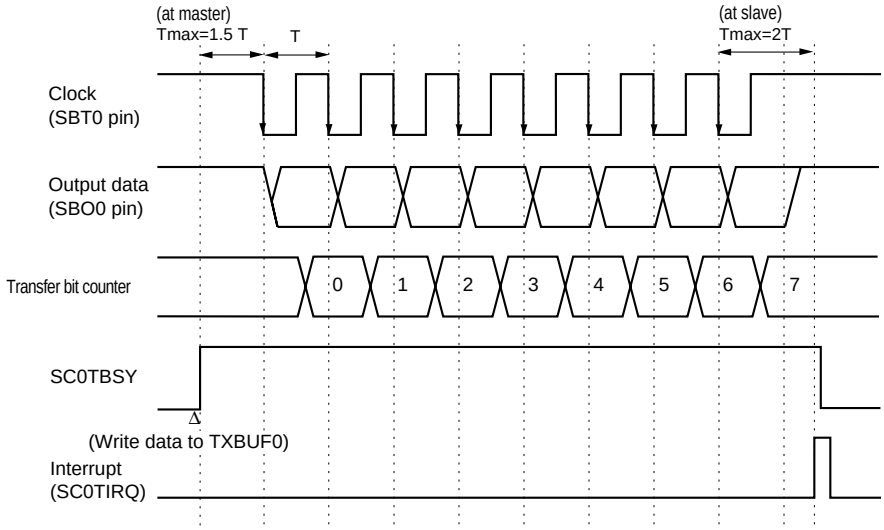


Figure 11-3-6 Transmission Timing (falling edge, start condition is disabled)

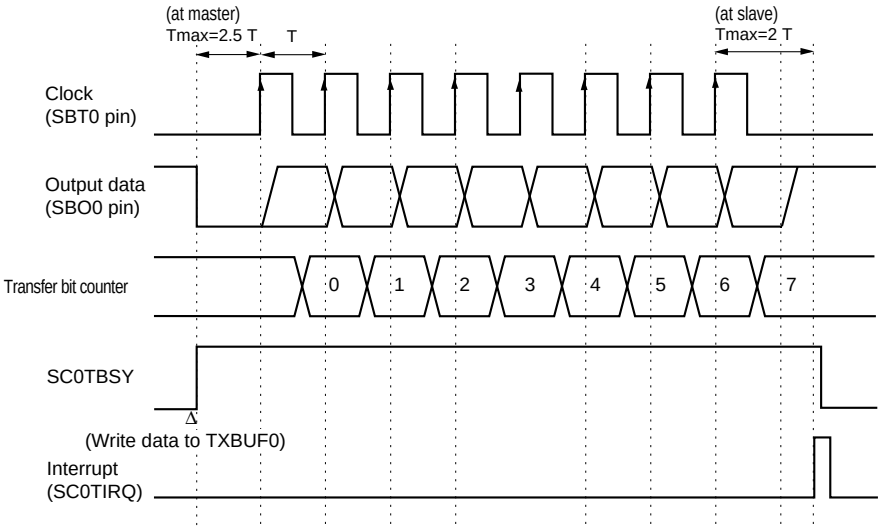


Figure 11-3-7 Transmission Timing (rising edge, start condition is enabled)

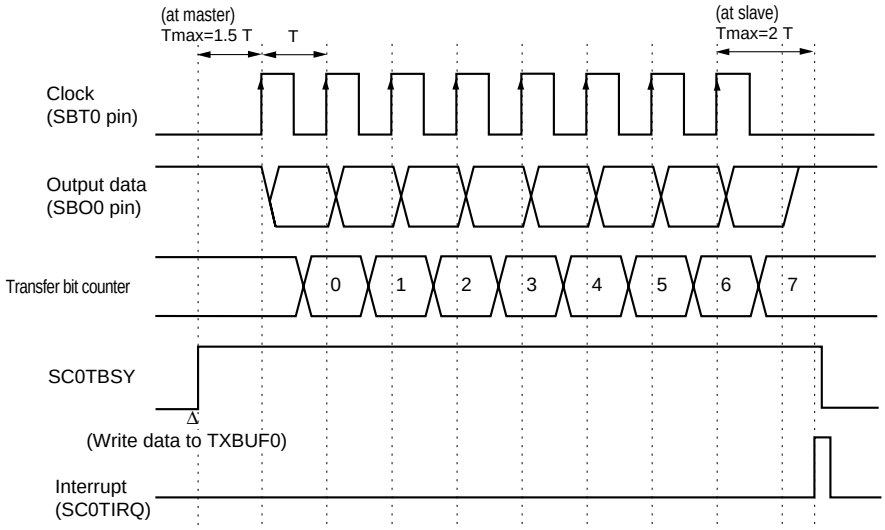


Figure 11-3-8 Transmission Timing (rising edge, start condition is disabled)

■Reception Timing

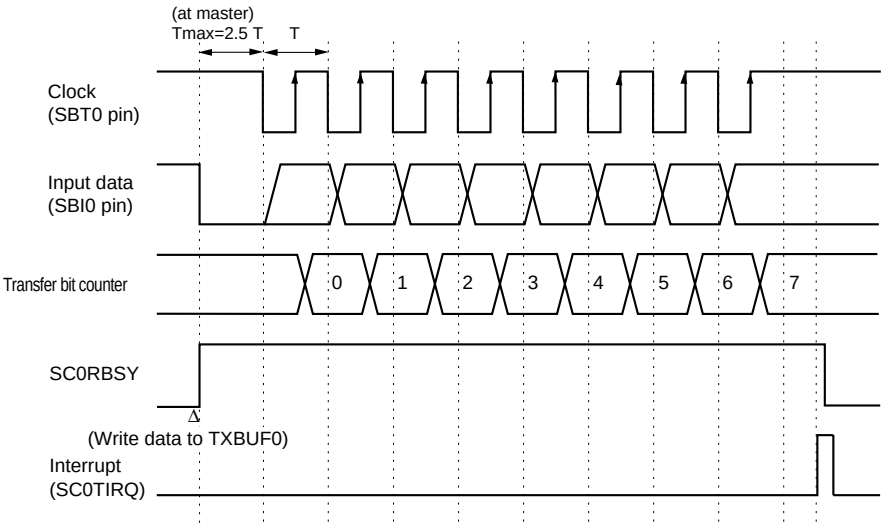


Figure 11-3-9 Reception Timing (rising edge, start condition is enabled)

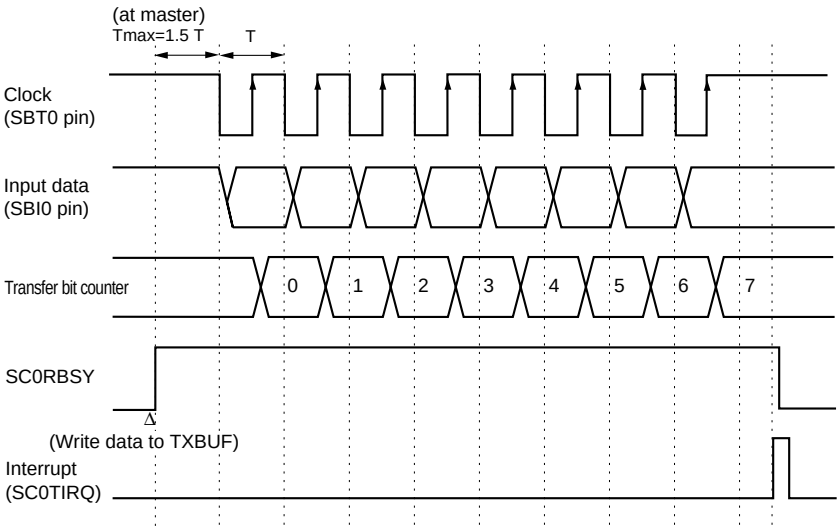


Figure 11-3-10 Reception Timing (rising edge, start condition is disabled)

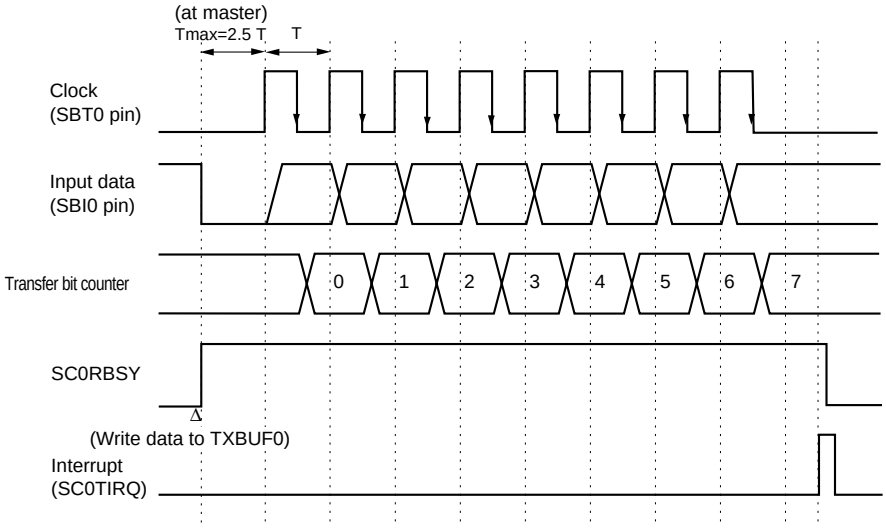


Figure 11-3-11 Reception Timing (falling edge, start condition is enabled)

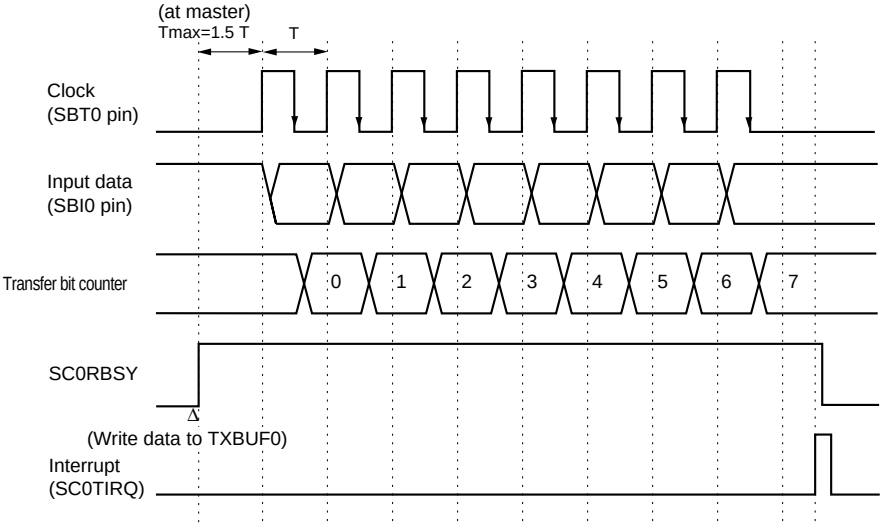


Figure 11-3-12 Reception Timing (falling edge, start condition is disabled)

■Transmission / Reception Timing

When transmission and reception are operated at the same time, set the SC0CE1 flag of the SC0MD0 register to "0" or "1". Data is received at the opposite edge of the transmission clock, so that the reception clock should be the opposite edge of the transmission clock from the other side.

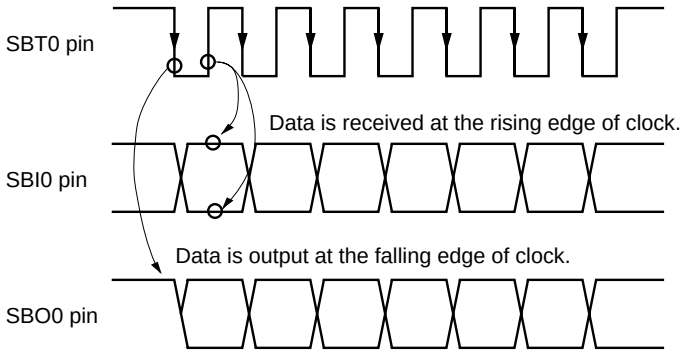


Figure 11-3-13 Transmission / Reception Timing (Reception : rising edge, Transmission : falling edge)

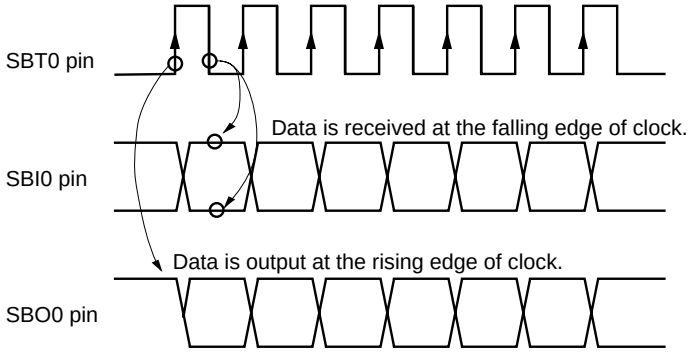


Figure 11-3-14 Transmission / Reception Timing (Reception : falling edge, Transmission : rising edge)

■At STANDBY mode

On serial interface, there are two ways to return from STANDBY mode.

(1) Slave reception at STANDBY mode is available. Operation mode of CPU can be recovered from STANDBY mode to NORMAL mode with communication complete interrupt SC0TIRQ, which is generated after the slave reception.

STANDBY mode does not accept next data once the transfer bit data set with the SC0LNG2-0 flag of the SC0MDO register is received, so this makes continuous reception impossible. Received data should be read out from the reception data buffer RXBUF after the recovery to NORMAL mode.

Communication with the start condition with reception during STANDBY mode cannot be operated, and the setup should be without start condition. Also, set a dummy data to the transmission data buffer TXBFU0 before the transition to STANBY mode.

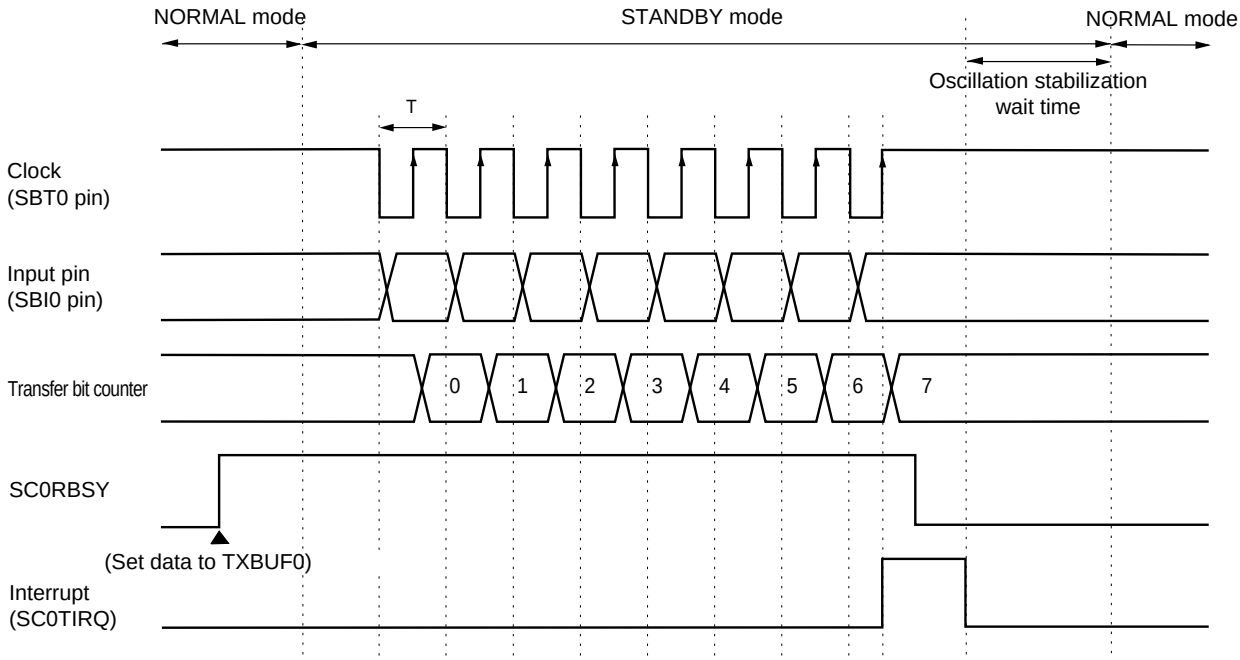


Figure 11-3-15 Reception Timing at STANDBY mode (Reception : rising edge, without start condition)

Select always fs/2 as a internal clock by SC0CKS for slave reception at STANDBY mode. Otherwise, normal reception is not guaranteed.

(2) When the level of serial interface 0 clock line (SBT0 pin) is changed by the generation of serial interface 0 clock, external interrupt 2IRQ2 is generated and this makes CPU operation mode return from STANDBY mode to NORMAL mode.

[☞ Chapter 3 3-3-9. P02(SBT0) interrupt]

■Pins Setup (3 channels, at transmission)

Table 11-3-7 shows the setup for synchronous serial interface pin with 3 channels (SBO0 pin, SBI0 pin, SBT0 pin) at transmission.

Table 11-3-7 Setup for Synchronous Serial Interface Pin (3 channels, at transmission)

Setup item	Data output pin	Data input pin	Clock I/O pin	
	SBO0 pin	SBI0 pin	SBT0 pin	
			Internal clock	External clock
Pin	P00	P01	P02	
SBI0 / SBO0 pin	SBI0 / SBO0 independent		-	
	SC0MD1(SC1IOM)			
Function	Serial data output	"1" input	Serial clock I/O	Serial clock I/O
	SC0MD1(SC0SBOS)	SC0MD1(SC0SBIS)	SC0MD1(SC0SBTS)	
Style	Push-pull / Nch open-drain	-	Push-pull / Nch open-drain	Push-pull / Nch open-drain
	SC0ODC(SC0ODC0)		SC0ODC(SC0ODC1)	
I/O	Output mode	-	Output mode	Input mode
	P0DIR(P0DIR0)		P0DIR(P0DIR2)	
Pull-up	Added / Not added	-	Added / Not added	Added / Not added
	P0PLU(P0PLU0)		P0PLU(P0PLU2)	

■Pins Setup (3 channels, at reception)

Table 11-3-8 shows the setup for synchronous serial interface pin with 3 channels (SBO0 pin, SBI0 pin, SBT0 pin) at reception.

Table 11-3-8 Setup for Synchronous Serial Interface Pin (3 channels, at reception)

Setup item	Data output pin	Data input pin	Clock I/O pin	
	SBO0 pin	SBI0 pin	SBT0 pin	
			Internal clock	External clock
Pin	P00	P01	P02	
SBI0 / SBO0 pin	SBI0 / SBO0 independent		-	
	SC0MD1(SC0IOM)			
Function	Port	Serial data input	Serial clock I/O	Serial clock I/O
	SC0MD1(SC0SBOS)	SC0MD1(SC0SBIS)	SC0MD1(SC0SBTS)	
Style	-	-	Push-pull / Nch open-drain	Push-pull / Nch open-drain
			SC0ODC(SC0ODC1)	
I/O	-	Input mode	Output mode	Input mode
		P0DIR(P0DIR1)	P0DIR(P0DIR2)	
Pull-up	-	-	Added / Not added	Added / Not added
			P0PLU(P0PLU2)	

■Pins Setup (3 channels, at transmission / reception)

Table 11-3-9 shows the setup for synchronous serial interface pin with 3 lines (SBO0 pin, SBI0 pin, SBT0 pin) at transmission / reception.

Table 11-3-9 Setup for Synchronous Serial Interface Pin (3 channels, at transmission / reception)

<table>
<tr><td rowspan="2">Setup item</td><td>Data output pin</td><td>Data input pin</td><td colspan="2">Clock I/O pin</td></tr>
<tr><td rowspan="2">SBO0 pin</td><td rowspan="2">SBI0 pin</td><td colspan="2">SBT0 pin</td></tr>
<tr><td>Internal clock</td><td>External clock</td></tr>
<tr><td>Pin</td><td>P00</td><td>P01</td><td colspan="2">P02</td></tr>
<tr><td rowspan="2">SBI0 / SBO0 pin</td><td colspan="2">SBI0 / SBO0 independent</td><td colspan="2" rowspan="2">-</td></tr>
<tr><td colspan="2">SC0MD1(SC0IOM)</td></tr>
<tr><td rowspan="2">Function</td><td>Serial data output</td><td>Serial data input</td><td>Serial clock I/O</td><td>Serial clock I/O</td></tr>
<tr><td>SC0MD1(SC0SBOS)</td><td>SC0MD1(SC0SBIS)</td><td colspan="2">SC0MD1(SC0SBTS)</td></tr>
<tr><td rowspan="2">Style</td><td>Push-pull /
Nch open-drain</td><td rowspan="2">-</td><td>Push-pull /
Nch open-drain</td><td>Push-pull /
Nch open-drain</td></tr>
<tr><td>SC0ODC(SC0ODC0)</td><td colspan="2">SC0ODC(SC0ODC1)</td></tr>
<tr><td rowspan="2">I/O</td><td>Output mode</td><td>Input mode</td><td>Output mode</td><td>Input mode</td></tr>
<tr><td>P0DIR(P0DIR0)</td><td>P0DIR(P0DIR1)</td><td colspan="2">P0DIR(P0DIR2)</td></tr>
<tr><td rowspan="2">Pull-up</td><td>Added / Not added</td><td rowspan="2">-</td><td>Added / Not added</td><td>Added / Not added</td></tr>
<tr><td>P0PLU(P0PLU0)</td><td colspan="2">P0PLU(P0PLU2)</td></tr>
</table>

■Pins Setup (2 channels, at transmission)

Table 11-3-10 shows the setup for synchronous serial interface pin with 2 channels (SBO0 pin, SBT0 pin) at transmission. SBI0 pin can be used as a general port.

Table 11-3-10 Setup for Synchronous Serial Interface Pin (2 channels, at transmission)

Setup item	Data output pin		Clock I/O pin	
	SBO0 pin	SBI0 pin	SBT0 pin	
			Internal clock	External clock
Pin	P00		P02	
SBI0 / SBO0 pin	SBI0/SBO0 connected		-	
	SC0MD1(SC0IOM)			
Function	Serial data output	"1" input	Serial clock I/O	Serial clock I/O
	SC0MD1(SC0SBOS)	SC0MD1(SC0SBIS)	SC0MD1(SC0SBTS)	
Style	Push-pull / Nch open-drain	-	Push-pull / Nch open-drain	Push-pull / Nch open-drain
	SC0ODC(SC0ODC0)		SC0ODC(SC0ODC1)	
I/O	Output mode	-	Output mode	Input mode
	P0DIR(P0DIR0)		P0DIR(P0DIR2)	
Pull-up	Added / Not added	-	Added / Not added	Added / Not added
	P0PLU(P0PLU0)		P0PLU(P0PLU2)	

■Pins Setup (2 channels, at reception)

Table 11-3-11 shows the setup for synchronous serial interface pin with 2 channels (SBO0 pin, SBT0 pin) at reception. SBI0 pin can be used as a general port.

Table 11-3-11 Setup for Synchronous Serial Interface Pin (2 channels, at reception)

Setup item	Data input pin		Clock I/O pin	
	SBO0 pin	SBI0 pin	SBT0 pin	
			Internal clock	External clock
Pin	P00		P02	
SBI0 / SBO0 pin	SBI0 / SBO0 connected		-	
	SC0MD1(SC0IOM)			
Function	Port	Serial data input	Serial clock I/O	Serial clock I/O
	SC0MD1(SC0SBOS)	SC0MD1(SC0SBIS)	SC0MD1(SC0SBTS)	
Style	-	-	Push-pull / Nch open-drain	Push-pull / Nch open-drain
			SC0ODC(SC0ODC1)	
I/O	Input mode	-	Output mode	Input mode
	P0DIR(P0DIR0)		P0DIR(P0DIR2)	
Pull-up	-	-	Added / Not added	Added / Not added
			P0PLU(P0PLU2)	

11-3-2 Setup Example

■Transmission / Reception Setup Example

The setup example for clock synchronous serial communication with serial 0 is shown. Table 11-3-12 shows the conditions at transmission / reception.

Table 11-3-12 Setup Examples for Synchronous Serial Interface Transmission / Reception

Setup item	set to	Setup item	set to
SBI0 / SBO0 pin	Independent (with 3 channels)	Clock source	fs/2
Transfer bit count	8 bits	Clock source 1/8 dividing	divided by 8
Start condition	none	SBT0 / SBO0 pin style	Nch open-drain
First transfer bit	MSB	SBT0 pin pull-up resistor	Added
Input clock edge	falling edge	SBO0 pin pull-up resistor	Added
Output clock edge	rising edge	Serial 0 communication complete interrupt	Enable
Clock	Internal clock	SBO0 output after last data transmission	Fixed to "1" (H)

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Select the prescaler operation. PSCMD (x'3F6F') bp0 : PSCEN = 1	(1) Set the PSCEN flag of the PSCMD register to "1" to select "prescaler operation".
(2) Select the clock source. SC0CKS (x'3F97') bp2-0 : SC0PSC2-0 = 100 bp3 : SC0TMSEL = 0	(2) Select the clcok source by the SC0CKS register. Set bp3-0 to "0100" to select "fs/2".
(3) Control the pin type. SC0ODC (x'3F96') bp2, 0 : SC0ODC1-0 = 1, 1 P0PLU (x'3F40') bp2, 0 : P3PLU2, 0 = 1, 1	(3) Set the SC0ODC2, 0 flag of the SC0ODC register to "11" to select "N-ch open drain" to the SBO0/SBT0 pin. Set the P0PLU2, 0 flag of the P0PLU register to "1, 1" to add pull-up resistor.
(4) SBO0 output control after the last data transmission SC0ODC(x'3F96') bp7, 8 : SC0FDC1, 0= 1, 1	(4) Set "0, 0" to the SC0FDC1, 0 flag of the SC0ODC register to select "1" (High) fixed output for SBO0 output after the last data transmission.
(5) Control the pin direction. P0DIR (x'3F30') bp2-0 : P0DIR2-0 = 101	(5) Set the P0DIR2-0 flag of the port 0 pin's direction control register (P0DIR) to "101" to set P00, P02 "output mode", and to set P01 "input mode".

Setup Procedure	Description
(6) Select the transfer bit count. SC0MD0 (x'3F90')	(6) Set the SC0LNG2-0 flag of the serial 0 mode register (SC0MD0) to "111" to set the transfer bit count "8 bits".
(7) Select the start condition. SC0MD0 (x'3F90') bp3 : SC0STE = 0	(7) Set the SC0STE flag of the SC0MD0 register to "0" to set the start condition.
(8) Select the first bit to be transfered. SC0MD0 (x'3F90') bp4 : SC0DIR = 0	(8) Set the SC0DIR flag of the SC0MD0 register to "0" to set MSB as the first transfer bit.
(9) Select the transfer edge. SC0MD0 (x'3F90') bp7 : SC0CE1 = 1	(9) Set the SC0CE1 flag of the SC0MD0 register to "1" to set the transmission data output edge "rising" and the received data input edge "falling".
(10) Control the output data. SC0MD2 (x'3F92') bp0 : SC0BRKE = 0	(10) Set the SC0BRKE flag of the SC0MD2 register to "0" to select "serial data transmission".
(11) Set other mode registers. SC0MD2 (x'3F92') bp7-3	(11) No need at synchronous serial communication.
(12) Select the communication type. SC0MD1 (x'3F91') bp0 : SC0CMD = 0	(12) Set the SC0CMD flag of the SC0MD1 register to "0" to select "synchronous serial".
(13) Select the transfer clock. SC0MD1 (x'3F91') bp2 : SC0MST = 1 bp3 : SC0CKM = 1	(13) Set the SC0MST flag of the SC0MD1 register to "1" to select clock master (inside clock). Set the SC0CKM flag to "1" to select "divide by 8" for source clock.
(14) Control the pin function. SC0MD1 (x'3F91') bp4 : SC0SBOS = 1 bp5 : SC0SBIS = 1 bp6 : SC0SBTS = 1 bp7 : SC0IOM = 0	(14) Set the SC0SBOS, SC0SBIS, SC0SBTS flag of the SC0MD1 register to "1" to set SBO0 pin "serial data output", SBI0 pin "serial data input", and SBT0 pin "serial clock I/O". Set the SC0IOM flag "0" to set serial data input from SBI0 pin.

Setup Procedure	Description
(15) Set the interrupt level. SC0TICR (x'3FF6') bp7-6 : SC0TLV1-0 = 10	(15) Set the interrupt level by the SC0TLV1-0 flag of the serial 0 transmission interrupt control register (SC0TICR). (Set level 2.)
(16) Enable the interrupt. SC0TICR (x'3FF6') bp1 : SC0TIE = 1	(16) Set the SC0TIE flag of the SC0TICR register to "1" to enable interrupts. If any interrupt request flag (SC0TIR of the SC0TICR register) is already set, clear SC0TIR before an interrupt is enabled. [☞ Chapter 3 3-1-4. Interrupt Flag Setup]
(17) Start serial transmission. Transmission data→TXBUF0 (x'3F95') Received data→input to SBI0 pin.	(17) Set the transmission data to the serial transmission data buffer TXBUF0. Then, an internal clock is generated to start transmission / reception. After the transmission is finished, serial0 transmission interrupt SC0TIRQ is generated.

Note : above (3) to (4), (6) to (9), (10) to (11) and (12) to (14), each settings can be set at once.

When only reception with 3 channels is operated, set SC0SBOS of the SC0MD1 register to "0" and select a port. The SBO0 pin can be used as a general port.

When SBO0 / SBI0 pin are connected for communication with 2 lines, the SBO0 pin inputs / outputs serial data. The port direction control register P0DIR switches I/O. At reception, set SC0SBIS of the SC0MD1 register to "1", always, to select "serial data input". The SBI0 pin can be used as a general port.

It is possible to shut down communication. If the communication should be stopped by force, set SC0SBOS and SC0SBIS of the SC0MD1 register to "0".

Each flag should be set as the procedure in order. Activation for communication should be operated after all control registers (except Table 11-2-1 : TXBUF0, RXBUF0) are set.

Transfer rate of transfer clock that set by SC0CKS register should be under 2.5 MHz.

When timer output is selected as serial interface transfer clock, select fosc as a clock source of the timer. If other clock is selected, normal transfer of serial interface data is not guaranteed.

■Transmission / reception Setup (at STANDBY mode)

Table 11-3-13 shows the setup for reception with serial interface 0 in clock synchronous serial interface communication in STANBY mode.

Table 11-3-13 Setup for Synchronous Serial Interface Reception in STANDBY mode

	Setup			Setup
SBI0/SBO0 pin	Independant (with 2 channels)		Clock source	fs/2
Transfer bit	8 bits		Clock source 1/8 dividing	No 1/8 dividing
Start condition	-		Style of SBT0/SBI0 pins	Push-pull
First transfer bit	MSB		Pull-up resistor for SBT0 pin	-
Input edge	Falling		Pull-up resistor for SBI0 pin	-
Clock	External clock		Serial interface 0 communication complete interrupt	Accept
Operation mode	STOP mode			

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Select the prescaler operation. PSCMD (x'3F6F') bp0 : PSCEN = 1	(1) Set the PSCEN flag of the PSCMD register to "1" to select "prescaler operation".
(2) Select the clock source. SC0CKS (x'3F97') bp2-0 : SC0PSC2-0 = 100 bp3 : SC0TMSEL = 0	(2) Select the clcok source by the SC0CKS register. Set bp3-0 to "0100" to select "fs/2".
(3) Control the pin type. SC0ODC (x'3F96') bp2, 1 : SC0ODC1-0 = 0, 0 P0PLU (x'3F40') bp2, 1 : P3PLU2, 0 = 0, 0	(3) Set the SC0ODC2, 1 flag of the SC0ODC register to "0, 0" to select "Push-pull" to the SBO0/SBT0 pin. Set the P0PLU2, 1 flag of the P0PLU register to "1, 1" to add pull-up resistor.
(4) Control the pin direction. P0DIR(x'3F30') bp2, 1 : P0DIR2, 1 = 0, 0	(4) Set "0, 0" to the P0DIR2, 1 flag of the Port 0 pin direction control register (P0DIR) to "0, 0" to set P00 and P01 to input mode.

Setup Procedure	Description
(5) Select the transfer bit count. SC0MD0 (x'3F90')	(5) Set the SC0LNG2-0 flag of the serial 0 mode register (SC0MD0) to "111" to set the transfer bit count "8 bits".
(6) Select the start condition. SC0MD0 (x'3F90') bp3 : SC0STE = 0	(6) Set the SC0STE flag of the SC0MD0 register to "0" to set the start condition.
(7) Select the first bit to be transfered. SC0MD0 (x'3F90') bp4 : SC0DIR = 0	(7) Set the SC0DIR flag of the SC0MD0 register to "0" to set MSB as the first transfer bit.
(8) Select the transfer edge. SC0MD0 (x'3F90') bp7 : SC0CE1 = 1	(8) Set the SC0CE1 flag of the SC0MD0 register to "1" to set the transmission data output edge "rising" and the received data input edge "falling".
(9) Control the output data. SC0MD2 (x'3F92') bp0 : SC0BRKE = 0	(9) Set the SC0BRKE flag of the SC0MD2 register to "0" to select "serial data transmission".
(10) Set other mode registers. SC0MD2 (x'3F92') bp7-3	(10) No need at synchronous serial communication.
(11) Select the communication type. SC0MD1 (x'3F91') bp0 : SC0CMD = 0	(11) Set the SC0CMD flag of the SC0MD1 register to "0" to select "synchronous serial".
(12) Select the transfer clock. SC0MD1 (x'3F91') bp2 : SC0MST = 1 bp3 : SC0CKM = 1	(12) Set the SC0MST flag of the SC0MD1 register to "1" to select clock master (inside clock). Set the SC0CKM flag to "1" to select "divide by 8" for source clock.
(13) Control the pin function. SC0MD1 (x'3F91') bp4 : SC0SBOS = 0 bp5 : SC0SBIS = 1 bp6 : SC0SBTS = 1 bp7 : SC0IOM = 0	(13) Set the SC0SBOS, SC0SBIS, SC0SBTS flag of the SC0MD1 register to "0" to set SBO0 pin "normal port", SBI0 pin "serial data input", and SBT0 pin "serial clock I/O". Set the SC0IOM flag "0" to set serial data input from SBI0 pin.

Setup Procedure	Description
(14) Set the interrupt level. SC0TICR (x'3FF6') bp7-6 : SC0TLV1-0 = 10	(15) Set the interrupt level by the SC0TLV1-0 flag of the serial 0 transmission interrupt control register (SC0TICR). (Set level 2.)
(15) Enable the interrupt. SC0TICR (x'3FF6') bp1 : SC0TIE = 1	(16) Set the SC0TIE flag of the SC0TICR register to "1" to enable interrupts. If any interrupt request flag (SC0TIR of the SC0TICR register) is already set, clear SC0TIR before an interrupt is enabled. [☞ Chapter 3 3-1-4. Interrupt Flag Setup]
(16) Set the serial interface communication activation factor. Set dummy data to TXBUF0 (x'3F95')	(16) Set dummy data to serial interface transmission data buffer TXBUF0
(17) Transition to STOP mode CPUM (x'3F00') bp3 : STOP = 0	(17) Set the STOP flag of the CPUM register to "1" to transition to STOP mode
(18) Start serial transmission. Transmission data→TXBUF0 (x'3F95') Received data→input to SBI0 pin.	(18) Input the transfer clock to the SBT0 pin, and input transition data to the SBI0 pin.
(19) Return from STANDBY mode	(19) Serial interface 0 interrupt SC0TIRQ is generated at the same time of reception of data of 8th bit. Then after the oscillation stabilization wait time, CPU returns from STOP mode to NORMAL mode.

Note : above (5) to (8), (9) to (10) and (11) to (13), each settings can be set at once.

Setup for slave reception in STANDBY mode should be without start condition.
Otherwise, proper reception is not guaranteed.

Each flag should be set as the procedure in order. Activation for communication should be operated after all control registers (except Table 11-2-1 : TXBUF0, RXBUF0) are set.

Select always fs/2 as a internal clock by SC0CKS for slave reception at STANDBY mode.
Otherwise, normal reception is not guaranteed.

11-3-3 UART Serial Interface

Serial 0 can be used for duplex UART communication. Table 11-3-13 shows UART serial interface functions.

Table 11-3-13 UART Serial Interface Functions

Communication style	UART(duplex)
Interrupt	SC0TIRQ(transmission), SC0RIRQ(reception)
Used pins	TXD0(output, input) RXD0(input)
Specification the first transfer bit	MSB / LSB
Selection of parity bit	√
Parity bit control	0 parity 1 parity odd parity even parity
Frame selection	7 bits + 1 stop 7 bits + 2 stops 8 bits + 1 stop 8 bits + 2 stops
Continuous operation	√
Maximum transfer rate	300 kbps (standard 300 bps to 38.4 kbps) (with baud rate timer)

■Activation Factor for Communication

At transmission, if any data is written to the transmission data buffer TXBUF0, a start condition is generated to start transfer. At reception, if a start condition is received, communication is started. At reception, if the data length of "L" for start bit is longer than 0.5 bit, that can be regarded as a start condition.

■Transmission

Data transfer is automatically started by writing data to the transmission data buffer TXBUF0. When the transmission has completed, the serial 0 transmission interrupt SC0TIRQ is generated.

■Reception

Once a start condition is received, reception is started after the transfer bit counter that counts transfer bit is cleared. When the reception is completed, the serial 0 reception interrupt SC0RIRQ is generated.

■Duplex communication

Duplex communication, that the transmission and reception can be operated independently at the same time is available. On duplex communication, the frame mode and parity bit of the used data on transmission / reception should have the same polarity.

■Transfer Bit Count Setup

The transfer bit count is automatically set after the frame mode is specified by the SC0FM1 to 0 flag of the SC0MD2 register. If the SC0CMD flag of the SC0MD1 register is set to "1", and UART communication is selected, the setup by the synchronous serial data transfer bit count selection flag SC0LNG2 to 0 is no more valid.

■Data Input Pin Setup

The communication mode can be selected from with 2 channels (data output pin (TXD0 pin), data input pin (RXD0 pin)), or with 1 channel (data I/O pin TXD0 pin). The RXD0 pin can be used only for serial data input. The TXD0 pin can be used for serial data input or output. The SC0IOM flag of the SC0MD1 register can specify which pin, RXD0 or TXD0 to input the serial data. "Data input from TXD0 pin" is selected to be with 1 channel communication, transmission / reception is switched by controlling TXD0 pin's direction by the P0DIR0 flag of the P0DIR register. At that time, the RXD0 pin can be used as a general port.

■Received Buffer Empty Flag

When the communication complete interrupt SC0RIRQ is generated, data is stored to RXBUF0 from the internal shift register, automatically. If data is stored to the shift register RXBUF0, the recieved buffer empty flag SC0REMP of the SC0STR register is set to "1". This indicates that the reception data is going to be read. SC0REMP is cleared to "0" by reading data in RXBUF0.

■Reception BUSY flag

When the start condition is reagarded, the SC0RBSY flag of the SC0STR register is set to "1". That is cleared to "0" by the generation of the reception complete interrupt SC0RIRQ. If, during reception, the SC0SBIS flag is set to "0", the SC0RBSY flag is reset to "0".

■Transmission BUSY flag

When any data is set to TXBUF0, the SC0TBSY flag of the SC0STR register is set to "1". That is cleared to "0" by the generation of the transmission complete interrupt SC0TIRQ. During continuous communication the SC0TBSY flag is always set. If the transmission buffer empty flag S0TEMP is set to "0" as the transmission complete interrupt SC0TIRQ is generated, the SC0TBSY is cleared to "0". If the SC0SBOS flag is set to "0", the SC0TBSY flag is reset to "0".

■Frame Mode and Parity Check Setup

Figure 11-3-16 shows the data format at UART communication.

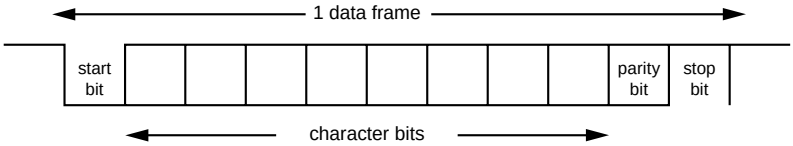


Figure 11-3-16 UART Serial Interface Transmission / Reception Data Format

The transmission / reception data consists of start bit, character bit, parity bit and stop bit.
Table 11-3-16 shows its kinds to be set.

Table 11-3-16 UART Serial Interface Transmission / Reception Data

Start bit	1 bit
Character bit	7, 8 bits
Parity bit	fixed to 0, fixed to 1, even, odd, none
Stop bit	1, 2 bits

The SC0FM1 to 0 flag of the SC0MD2 register sets the frame mode. Table 11-3-15 is shown the UART Serial Interface Frame Mode setting. If the SC0CMD flag of the SC0MD1 register is set to "1", and UART communication is selected, the transfer bit count on the SC0LNG2 to 0 flag of the SC0MD0 register is no more valid.

Table 11-3-15 UART Serial Interface Frame Mode

SC0MD2 register		Frame mode
SC0FM1	SC0FM0	
0	0	Character bit 7 bits + Stop bit 1 bit
0	1	Character bit 7 bits + Stop bit 2 bits
1	0	Character bit 8 bits + Stop bit 1 bit
1	1	Character bit 8 bits + Stop bit 2 bits

Parity bit is to detect wrong bits with transmission / reception data.
Table 11-3-16 shows kinds of parity bit. The SC0NPE, SC0PM1 to 0 flag of the SC0MD2 register set parity bit.

Table 11-3-16 Parity Bit of UART Serial Interface

SC0MD2 register			Parity bit	Setup
SC0NPE	SC0PM1	SC0PM0		
0	0	0	fixed to 0	Set parity bit to "0".
0	0	1	fixed to 1	Set parity bit to "1".
0	1	0	odd parity	Control that the total of "1" of parity bit and character bit should be odd.
0	1	1	even parity	Control that the total of "1" of parity bit and character bit should be even.
1	-	-	none	Do not add parity bit.

In case the SC0NPE flag is "1" and disable parity bit is selected, do not set character bit 7 bits + stop bit 2 bits of frame mode.

■Break Status Transmission Control Setup
The SC0BRKE flag of the SC0MD2 register generates the break status. If SC0BRKE is set to "1" to select the break transmission, all bits from start bits to stop bits transfer "0".

■Reception Error

At reception , there are 3 types of error ; overrun error, parity error and framing error. Reception error can be determined by the SC0ORE, SC0PEK, SC0FEF flag of the SC0STR register. Even one of those errors is detected, the SC0ERE flag of the SC0STR register is set to "1". The SC0PEK, the SC0FEF flags in recepption error flag are renewed at generation of the reception complete interrupt SC0RIRQ. The SC0ORE flag is holded the status unless data of RXBUF0 is read out. The judgements of the received error flag should be operated until the next communication is finished. The communication operation does not have any effect on those error flags . Table 11-3-17 shows the list of reception error source.

Table 11-3-17 Reception Error Source of UART Serial Interface

Flag	Error	Error source	
SC0ORE	Overrun error	Next data is received before reading the receive buffer.	
SC0PEK	Parity error	at fixed to 0	when parity bit is "1"
		at fixed to 1	when parity bit is "0"
		odd parity	The total of "1" of parity bit and character bit is even.
		even parity	The total of "1" of parity bit and character bit is odd.
SC0FEF	Framing error	Stop bit is not detected.	

■Judgement of Break Status Reception

Reception at break status can be judged. If all received data from start bit to stop bit is "0", the SC0BRKF flag of the SC0MD2 register is set and regard the break status. The SC0BRKF flag is set at generation of the reception complete interrupt SC0RIRQ.

■Sequence Communication

It is possible to transfer continuously. If data is set to the transmission data buffer TXBUF0 during communication, the transmission buffer empty flag SC0TEMP is set to continue the communication, automatically. In this case, there is no pause on communication. Data should be set to TXBUF0 after data is loaded to the inside shift register before the communication complete interrupt SC0TIRQ is generated.

■Clock Setup

At UART communication, the transfer clock is not needed, but the clock setup should be needed to decide the timing of the data transmission / reception in the serial interface.

Select the timer to be used as a baud rate timer, by the SC0CKS register, and set the SC0MST flag of the SC0MD1 register to "1" to select the internal clock (clock master).

At UART communication, set the SC0MST flag of the SC0MD1 register to "1". If that is set to "0", the communication is impossible.

When timer output is selected as serial interface transfer clock, select fosc as a clock source of the timer. If other clock is selected, normal transfer of serial interface data is not guaranteed.

■Tranfer Bit Count and First Transfer Bit

When the transfer bit is 7 bits, the data storing method to the transmission data buffer TXBUF0 is different, depending on the first transfer bit selection. At MSB first, use the upper bits of TXBUF0 for storing. When there are 7 bits to be transfered, as shown on figure 11-3-17, if data "A" to "G" are stored to bp1 to bp7 of TXBUF0, the transmission is operated from "G" to "A". At LSB first, use the lower bits of TXBUF0 for storing. When there are 7 bits to be transfered, as shown on figure 11-3-18, if data "A" to "G" are stored to bp0 to bp6 of TXBUF0, the transmission is operated from "A" to "G".

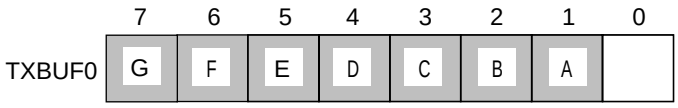


Figure 11-3-17 Transfer Bit Count and First Transfer Bit (starting with MSB)

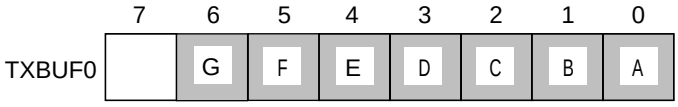


Figure 11-3-18 Transfer Bit Count and First Transfer Bit (starting with LSB)

■Receive Bit Count and First Transfer Bit

When the transfer bit count is 7 bits, the data storing method to the received data buffer RXBUF0 is different depending on the first transfer bit selection. At MSB first, data are stored to the upper bits of RXBUF0. When there are 7 bits to be transfered, as shown on figure 11-3-19, if data "G" to "A" are stored to bp7 to bp1 of RXBUF0. At LSB first, data are stored to the lower bits of RXBUF0. When there are 7 bits to be transfered, as shown on figure 11-3-20, if data "A" to "G" are stored to bp0 to bp6 of RXBUF0.

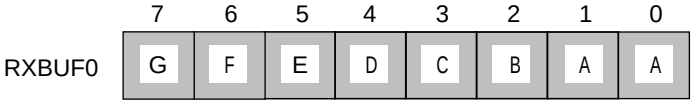


Figure 11-3-19 Receive Bit Count and Transfer First Bit (starting with MSB bit)

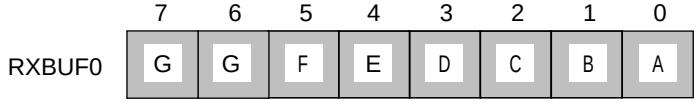


Figure 11-3-20 Receive Bit Count and Transfer First Bit (starting with LSB bit)

The following items are same to clock synchronous serial.
Reference as follows ;

■First Transfer Bit Setup
Refer to : XI-13

■Transmission Data Buffer
Refer to : XI-13

■Received Data Buffer
Refer to : XI-13

■Transmission Buffer Empty Flag
Refer to : XI-17

■Emergency Reset
Refer to : XI-18

■Transmission Timing

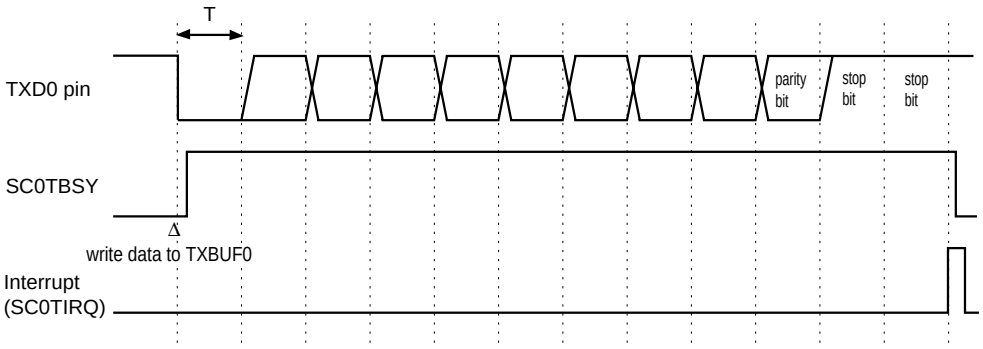


Figure 11-3-21 Transmission Timing (parity bit is enabled)

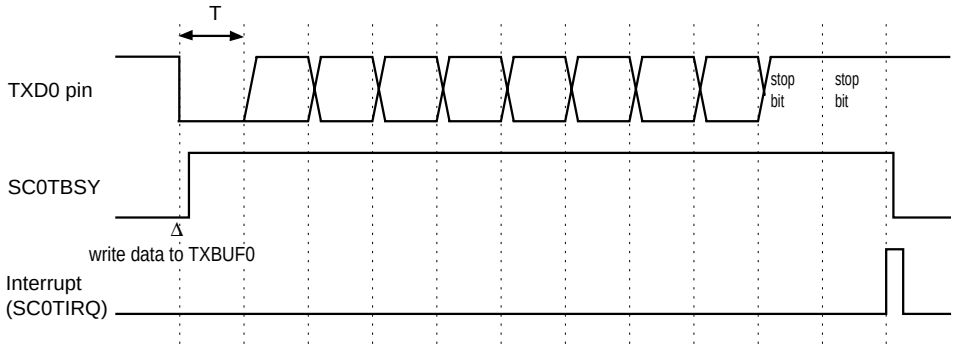


Figure 11-3-22 Transmission Timing (parity bit is disabled)

■Reception Timing

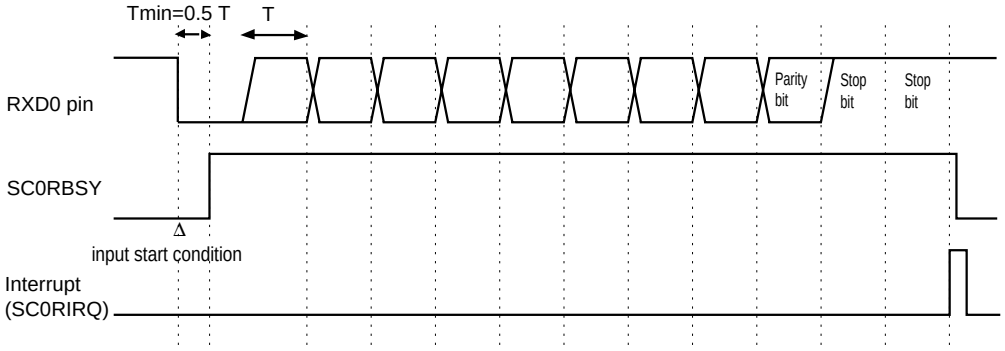


Figure 11-3-23 Reception Timing (parity bit is enabled)

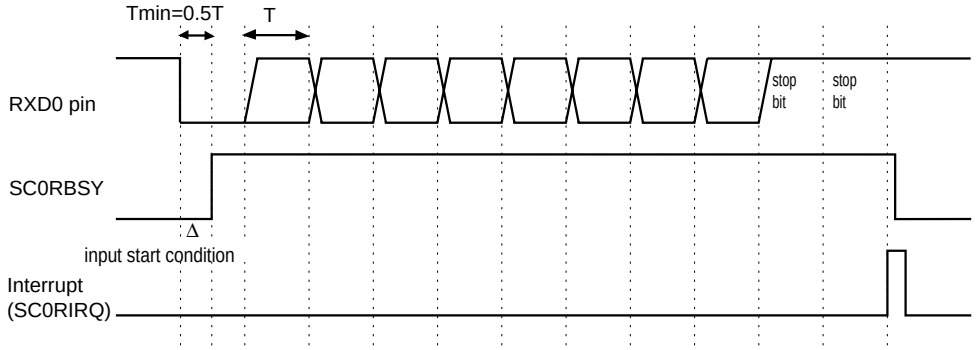


Figure 11-3-24 Reception Timing (parity bit is disabled)

■Transfer Rate

Baud rate timer (timer 2 and timer 3) can set any transfer rate.

Table 11-3-18 shows the setup example of the transfer rate. For detail of the baud rate timer setup, refer to chapter 6. 6-7 serial transfer cock output operation.

Table 11-3-18 UART Serial Interface Transfer Rate Setup Register

Setup	Register	Page
Serial 0 clock source (timer 2, 3 output)	SC0CKS	XI - 11
Timer 2 clock source	TM2MD	VI -7
Timer 2 compare register	TM2OC	VI - 6
Timer 3 clock source	TM3MD	VI - 8
Timer 3 compare register	TM3OC	VI - 6

Timer 4 compare register is set as follows ;

overflow cycle = (set value of compare register + 1) x timer clock cycle

baud rate = 1 / (overflow cycle x 2 x 8) ("8" means that clock source is divided by 8)

therefore,

set value of compare register = timer clock frequency / (baud rate x 2 x 8) - 1

For example, if baud rate should be 300 bps at timer clock source fs/4 (fosc = 8 MHz, fs = fosc/2), set value should be as follows ;

Set value of comapre register = $(8 \times 10^6 / 2 / 4) / (300 \times 2 \times 8) - 1$

= 207

= x'CF'

Timer clock source and the set values of timer compare register at the standard rate are shown on the following page.

Transfer rate should be selected under 300 kbps.

When timer output is selected as serial interface transfer clock, select fosc as a clock source of the timer. If other clock is selected, normal transfer of serial interface data is not guaranteed.

Table 11-3-19-1 UART Serial Interface Transfer Rate (decimal)

fosc (MHz)	Clock source (timer)	Transfer rate (bps)									
		300		960		1200		2400		4800	
		Set value	Calculated Value	Set value	Calculated Value	Set value	Calculated Value	Set value	Calculated Value	Set value	Calculated Value
4.00	fosc	-	-	-	-	207	1202	103	2404	51	4808
4.19	fosc	-	-	-	-	217	1201	108	2403	54	4761
8.00	fosc	-	-	-	-	-	-	207	2404	103	4808
8.38	fosc	-	-	-	-	-	-	217	2403	108	4805
12.00	fosc	-	-	-	-	-	-	-	-	155	4808
16.00	fosc	-	-	-	-	-	-	-	-	207	4808
16.76	fosc	-	-	-	-	-	-	-	-	-	-
20.00	fosc	-	-	-	-	-	-	-	-	-	-

Table 11-3-19-2 UART Serial Interface Transfer Rate (decimal)

fosc (MHz)	Clock source (timer)	Transfer rate (bps)									
		9600		19200		28800		31250		38400	
		Set value	Calculated Value	Set value	Calculated Value	Set value	Calculated Value	Set value	Calculated Value	Set value	Calculated Value
4.00	fosc	25	9615	12	19231	-	-	7	31250	-	-
4.19	fosc	26	9699	-	-	-	-	-	-	-	-
8.00	fosc	51	9615	25	19231	-	-	15	31250	12	38462
8.38	fosc	54	9523	26	19398	-	-	-	-	-	-
12.00	fosc	77	9615	38	19231	25	28846	23	31250	-	-
16.00	fosc	103	9615	51	19231	-	-	31	31250	25	38462
16.76	fosc	108	9610	54	19045	-	-	-	-	-	-
20.00	fosc	129	9615	64	19231	-	-	39	31250	-	-

■Pin Setup (1, 2 channels, at transmission)

Table 11-3-20 shows the pins setup at UART serial interface transmission. The pins setup is common to the TXD0 pin, RXD0 pin, regardless of those pins are independent / connected.

Table 11-3-20 UART Serial Interface Pin Setup (1, 2 channels, at transmission)

Setup item	Data output pin	Data input pin
	TXD0 pin	RXD0 pin
Pin	P00	P01
TXD0 / RXD0 pins	TXD0 / RXD0 pins connected or independent	
	SC0MD1(SC0IOM)	
Function	Serial data output	"1" input
	SC0MD1(SC0SBOS)	SC0MD1(SC0SBIS)
Style	Push-pull / Nch open-drain	-
	SC0ODC(SC0ODC0)	
I/O	Output mode	-
	P0DIR(P0DIR0)	
Pul-up	Added / Not added	-
	P0PLU(P0PLU0)	

■Pin Setup (2 channels, at reception)

Table 11-3-21 shows the pins setup at UART serial interface reception with 2 channels (TXD0 pin, RXD0 pin).

Table 11-3-21 UART Serial Interface Pin Setup (2 channels, at reception)

Setup item	Data output pin	Data input pin
	TXD0 pin	RXD0 pin
Pin	P00	P01
TXD0 / RXD0 pin	TXD0 / RXD0 pins connected or independent	
	SC0MD1(SC0IOM)	
Function	port	serial data input
	SC0MD1(SC0SBOS)	SC0MD1(SC0SBIS)
Style	-	-
I/O	-	input mode
	-	P0DIR(P0DIR1)
Pull-up	-	-

■Pin Setup (1 channel, at reception)

Table 11-3-22 shows the pin setup at UART serial interface reception with 1 channel (TXD0 pin). The RXD0 pin is not used, so can be used as a port.

Table 11-3-22 UART Serial Interface Pin Setup (1 channel, at reception)

Setup item	Data output pin	Data input pin
	TXD0 pin	RXD0 pin
Pin	P00	P01
TXD0 / RXD0 pin	TXD0 / RXD0 pins connected	
	SC0MD1(SC0IOM)	
Function	Port	Serial data input
	SC0MD1(SC0SBOS)	SC0MD1(SC0SBIS)
Style	-	-
I/O	Input mode	-
	P0DIR(P0DIR0)	-
Pull-up	-	-

■Pin Setup (2 channels, at transmission / reception)

Table 11-3-23 shows the pin setup at UART serial interface transmission / reception with 2 channels (TXD0 pin, RXD0).

Table 11-3-23 UART Serial Interface Pin Setup (2 channels, at transmission / reception)

Setup item	Data output pin	Data input pin
	TXD0 pin	RXD0 pin
Pin	P00	P01
TXD0 / RXD0 pins	TDX0 / RXD0 pins independent	
	SC0MD1(SC0IOM)	
Function	Serial data output	Serial data input
	SC0MD1(SC0SBOS)	SC0MD1(SC0SBIS)
Style	Push-pull / Nch open-drain	-
	SC0ODC(SC0ODC0)	
I/O	Output mode	Input mode
	P0DIR(P0DIR0)	P0DIR(P0DIR1)
Pull-up	Added / Not added	-
	P0PLU(P0PLU0)	

11-3-4 Setup Example

■Transmission / Reception Setup

The setup example at UART transmission / reception with serial 0 is shown.

Table 11-3-24 shows the conditions at transmission / reception.

Table 11-3-24 UART Interface Transmision Reception Setup

Setup item	set to
TXD0 / RXD0 pin	independent (with 2 channels)
Frame mode specification	8 bits + 2 stop bits
First transfer bit	MSB
Clock source	timer 3
TXD0 / RXD0 pin type	Nch open-drain
Pull-up resistor of TXD0 pin	added
Parity bit add / check	"0"add / check
Serial interface 0 transmission complete interrupt	Enable.
Serial interface 0 reception complete interrupt	Enable.

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Select prescaler operation. PSCMD (x'3F6F') bp0 : PSCEN = 1	(1) Set the PSCEN flag of the PSCMD register to "1" to select prescaler operation.
(2) Select the clock source. SC0CKS (x'3F97') bp2-0 : SC0PSC2-0 = 110 bp3 : SC0TMSEL = 1	(2) Set the bp3-0 flag of the SC0CKS register to "1110" to select timer 3 output as a clock source.
(3) Control the pin type. SC0ODC (x'3F96') bp0 : SC0ODC0 = 1 P0PLU (x'3F40') bp0 : P0PLU0 = 1	(3) Set the SC0ODC0 flag of the SC0ODC register to "1" to select N-ch open drain for the TXD0 pin. Set the P0PLU0 flag of the P0PLU register to "1" to add pull-up resistor.
(4) Control the pin direction. P0DIR (x'3F30') bp1-0 : P0DIR1-0 = 01	(4) Set the P0DIR1-0 flag of the port 0 pin direction control register (P0DIR) to "01" to set P00 to output mode, and P01 to input mode.
(5) Select the start condition. SC0MD0 (x'3F90') bp3 : SC0STE = 1	(5) Set the SC0STE flag of the SC0MD0 register to "1" to enable start condition.

Setup Procedure	Description
(6) Select the first bit to be transfered. SC0MD0 (x'3F90') bp4 : SC0DIR = 0	(6) Set the SC0DIR flag of the SC0MD0 register to "0" to select MSB as first transfer bit.
(7) Control the output data. SC0MD2 (x'3F92') bp0 : SC0BRKE = 0	(7) Set the SC0BRKE flag of the SC0MD2 register to "0" to select serial data transmission.
(8) Select the added parity bit. SC0MD2 (x'3F92') bp3 : SC0NPE = 0 bp5-4 : SC0PM1-0 = 00	(8) Set the SC0PM1-0 flag of the SC0MD2 register to "00" to select 0 parity, and set the SC0NPE flag to "0" to add parity bit.
(9) Specify the frame mode. SC0MD2 (x'3F92') bp7-6 : SC0FM1-0 = 11	(9) Set the SC0FM1-0 flag of the SC0MD2 register to "11" to select 8 bits + 2 stop bits at the frame mode.
(10) Select the communication type. SC0MD1 (x'3F91') bp0 : SC0CMD = 1	(10) Set the SC0CMD flag of the SC0MD1 register to "1" to select duplex UART.
(11) Select the clock frequency. SC0MD1 (x'3F91') bp3 : SC0CKM = 1 bp2 : SC0MST = 1	(11) Set the SC0CKM flag of the SC0MD1 register to "1" to select "divided by 8" at source clock. And, the SC0MST flag should be always set to "1" to select colck master.
(12) Control the pin function. SC0MD1 (x'3F91') bp4 : SC0SBOS = 1 bp5 : SC0SBIS = 1 bp7 : SC0IOM = 0	(12) Set the SC0SBOS, SC0SBIS flag of the SC0MD1 register to "1" to set the TXD0 pin to serial data output and the RXD0 pin to serial data input.
(13) Enable the interrupt. SC0RICR (x'3FF5') bp1 : SC0RIE = 1 SC0TICR (x'3FF6') bp1 : SC0TIE = 1	(13) Set the SC0RIE flag of the SC0RICR register to "1", and set the SC0TIE flag ot the SC0TICR register to "1" to enable the interrupt request. If any interrupt request flag is already set, clear them. [☞ Chapter 3. 3-1-4 Interrupt Flag Setup]

Setup Procedure	Description
(15) Set the baud rate timer.	(15) Set the baud rate timer by the TM3MD register, the TM3OC register. Set the TM3EN flag to "1" to start timer 3. [☞ Chapter 6. 6-7 Serial Transfer Clock]
(16) Start serial communication. The transmission data → TXBUF0 (x'3F95') The received data → input to RXD0	(16) The transmission is started by setting the transmission data to the serial transmission data buffer (TXBUF0). When the transmission has finished, the serial 0 transmission interrupt (SC0TIRQ) is generated. After the serial data is input from the RXD0 pin and the start condition is recognized, the received data is stored. When the reception has finished, the received data is stored to the serial received data buffer RXBUF0 and the serial 0 reception data buffer interrupt SC0RICR is generated.

Note : (5) to (6), (7) to (9), (10) to (12) can be set at once.

When the TXD0 / RXD0 pin are connected for communication with 1 channel, the TXD0 pin inputs / outputs serial data. The port direction control register P0DIR switches I/O. At reception, set SC0SBIOS of the SC0MD1 register to "1" to select serial data input. The RXD0 pin can be used as a general port.

It is possible to shut down the communication. If the communication should be stopped by force, set SC0SBOS and SC0SBIS of the SC0MD1 register to "0".

Each flag should be set as its procedure in order. Activation for communication should be operated after all control registers (except Table 11-2-1 : TXBUF0, RXBUF0) are set.

Only timer 2 and timer4 can be used as a baud rate timer.
For baud rate setup, refer to Chapter 6. 6-7 Serial Transfer Clock Output.

When timer output is selected as serial interface transfer clock, select fosc as a clock source of the timer. If other clock is selected, normal transfer of serial interface data is not guaranteed.

Chapter 12 A/D Converter

12-1 Overview

This LSI has an A/D converter with 10 bits resolution. That has a built-in sample hold circuit, and software can switch channel 0 to 7 (AN0 to AN7) to analog input. As A/D converter is stopped, the power consumption can be reduced by a built-in ladder resistance. A/D converter is activated by 2 factors : a register setup or an external interrupt.

12-1-1 Functions

Table 12-1-1 shows the A/D converter functions.

Table 12-1-1 A/D Converter Functions

A/D Input Pins	8 pins
Pins	AN7 to AN0
Interrupt	ADIRQ
Resolution	10 bits
Conversion Time (Min.)	9.6 µs(T_{AD} = as 800 ns)
Input range	V_{REF-} to V_{REF+}
Power Consumption	Built-in Ladder Resistance (ON/OFF)

12-1-2 Block Diagram

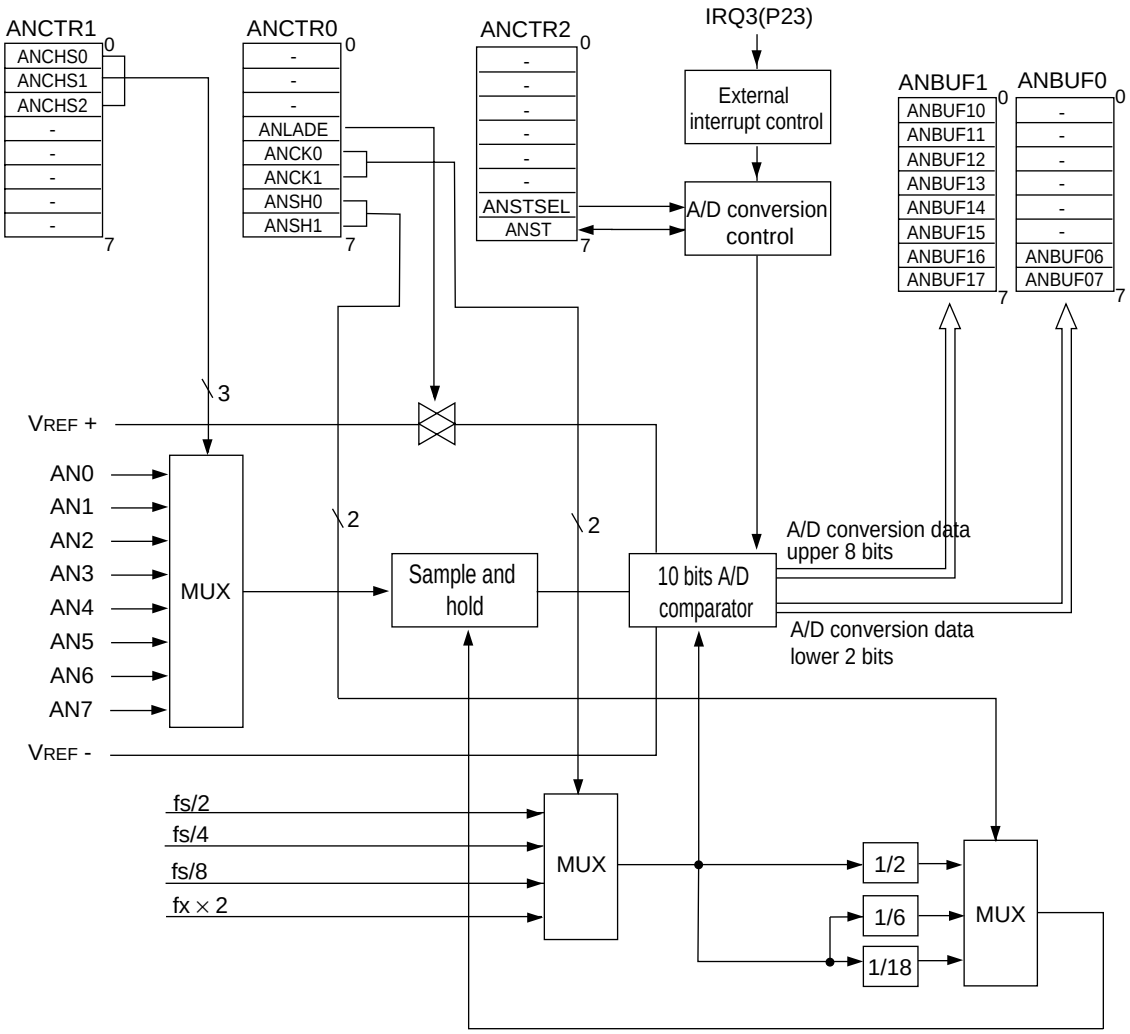


Figure 12-1-1 A/D Converter Block Diagram

12-2 Control Registers

A/D converter consists of the control register (ANCTRn) and the data storage buffer (ANBUFn).

12-2-1 Registers

Table 12-2-1 shows the registers used to control A/D converter.

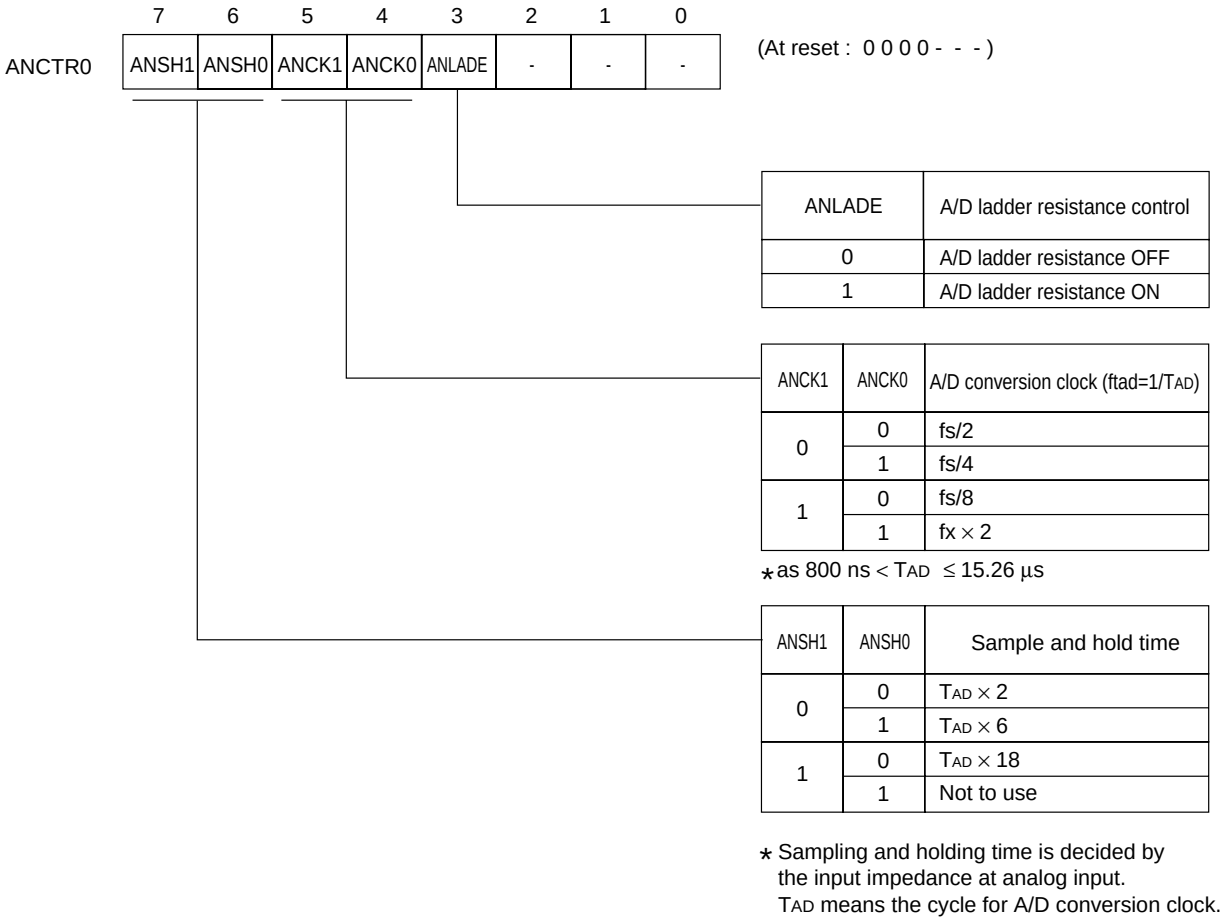
Table 12-2-1 A/D Converter Control Registers

Register	Address	R/W	Function	Page
ANCTR0	x'03FB0'	R/W	A/D converter control register 0	XVI - 5
ANCTR1	x'03FB1'	R/W	A/D converter control register 1	XVI - 6
ANCTR2	x'03FB2'	R/W	A/D converter control register 2	XVI - 6
ANBUF0	x'03FB3'	R	A/D converter data storage buffer 0	XVI - 7
ANBUF1	x'03FB4'	R	A/D converter data storage buffer 1	XVI - 7
ADICR	x'03FFB'	R/W	A/D +converter interrupt control register	III - 29
IRQ3ICR	x'03FE5'	R/W	External interrupt 3 control register	III - 20
EDGDT	x'03F8F'	R/W	Both edges interrupt control register	III - 36
PAIMD	x'03F3A'	R/W	Port A input mode register	IV - 30
PAPLUD	x'03F4A'	R/W	Port A pull-up/pull-down resistance control register	IV - 31

R/W : Readable/Writable
R : Readable only

12-2-2 Control Registers

■A/D Converter Control Register 0 (ANCTR0)



ANLADE	A/D ladder resistance control
0	A/D ladder resistance OFF
1	A/D ladder resistance ON

ANCK1	ANCK0	A/D conversion clock (ftad=1/TAD)
0	0	fs/2
	1	fs/4
1	0	fs/8
	1	fx × 2

* as 800 ns < TAD ≤ 15.26 µs

ANSH1	ANSH0	Sample and hold time
0	0	TAD × 2
	1	TAD × 6
1	0	TAD × 18
	1	Not to use

* Sampling and holding time is decided by the input impedance at analog input.
TAD means the cycle for A/D conversion clock.

Figure 12-2-1 A/D Converter Control Register 0 (ANCTR0 : x'03FB0', R/W)

■A/D Converter Control Register 1 (ANCTR1)

	7	6	5	4	3	2	1	0
ANCTR1	-	-	-	-	RESERVED	ANCHS2	ANCHS1	ANCHS0

(At reset : - - - - 0 0 0 0)

ANCHS2	ANCHS1	ANCHS0	Analog Input Channel
0	0	0	AN0 (PA0)
		1	AN1 (PA1)
	1	0	AN2 (PA2)
		1	AN3 (PA3)
1	0	0	AN4 (PA4)
		1	AN5 (PA5)
	1	0	AN6 (PA6)
		1	AN7 (PA7)

RESERVED	Set always "0".

Figure 12-2-2 A/D Converter Control Register 1 (ANCTR1 : x'03FB1', R/W)

■A/D Converter Control Register 2 (ANCTR2)

	7	6	5	4	3	2	1	0
ANCTR2	ANST	ANSTSEL	-	-	-	-	-	-

(At reset : 0 0 - - - - - -)

ANSTSEL	A/D conversion activation factor selection
0	Set the ANST flag to "1".
1	Set the ANST flag, or the external interrupt 3 to "1".

ANST	A/D conversion status
0	A/D conversion is completed, stopped
1	A/D conversion is started, in progress

Figure 12-2-3 A/D Converter Control Register 2 (ANCTR2 : x'03FB2', R/W)

12-2-3 Data Buffers

■A/D Conversion Data Storage Buffer 0 (ANBUF0)

The lower 2 bits from the result of A/D conversion are stored to this register.

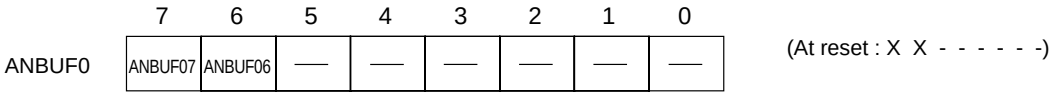


Figure 12-2-4 A/D Conversion Data Buffer 0 (ANBUF0 : x'03FB3', R)

■A/D Conversion Data Storage Buffer 1 (ANBUF1)

The upper 8 bits from the result of A/D conversion are stored to this register.

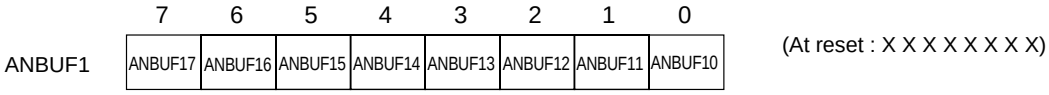


Figure 12-2-5 A/D Conversion Data Buffer 1 (ANBUF1 : x'03FB4', R)

12-3 Operation

Here is a description of A/D converter circuit setup procedure.

(1) Set the analog pins.
Set the analog input pin, set in (2), to "special function pin" by the port A input mode register (PAIMD).
* Setup for the port A input mode register should be done before analog voltage is put to pins.

(2) Select the analog input pin.
Select the analog input pin from AN7 to AN0 (PA7 to PA0) by the ANCHS2 to ANCHS0 flag of the A/D converter control register 1 (ANCTR1).

(3) Select the A/D converter clock.
Select the A/D converter clock by the ANCK1, ANCK0 flag of the A/D converter control register 0 (ANCTR0).
Setup should be in such a way that converter clock is not below 800 ns with any resonator.

(4) Set the sample hold time.
Set the sample hold time by the ANSH1, ANSH0 flag of the A/D converter control register 0 (ANCTR0). The sample hold time should be based on analog input impedance.

(5) Set the A/D ladder resistance.
Set the ANLADE flag of the A/D converter control register 0 (ANCTR0) to "1", and a current flow through the ladder resistance and A/D converter goes into the waiting.
* (2) to (5) are not in order. (3), (4) and (5) can be operated simultaneously.

(6) Select the A/D converter activation factor, then start A/D conversion.
Set the ANST flag of the A/D converter control register 2 (ANCTR2) to "1" to start A/D converter, or set the ANSTSEL flag of the A/D converter control register 2 (ANCTR2) to "1" to start A/D conversion by the external interrupt IRQ3.
* Specify the valid edge by the REDG3 flag of the external interrupt 3 control register (IRQ3ICR).

(7) A/D conversion
Each bit of the A/D buffer 0,1 is generated after sampling with the sample and hold time set in (3). Each bit is generated in sequence from MSB to LSB.

(8) Complete the A/D conversion.
When A/D conversion is finished, the ANST flag is cleared to "0", and the result of the conversion is stored to the A/D buffer (ANBUF0, 1). At the same time, the A/D complete interrupt request (ADIRQ) is generated.

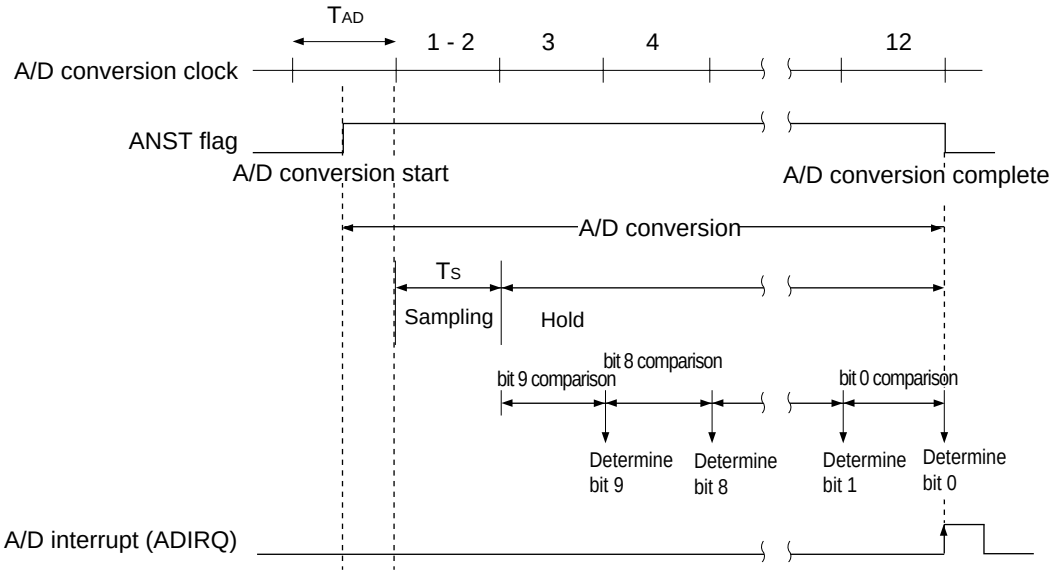


Figure 12-3-1 Operation of A/D Conversion

To read the value of the A/D conversion, A/D conversion should be done several times to prevent noise error by confirming the match of level by program, or by using the average value.

12-3-1 Setup

■Input Pins of A/D Converter Setup

Input pins for A/D converter is selected by the ANCH2 to 0 flag of the ANCTR1 register.

Table 12-3-1 Input Pins of A/D Converter Setup

ANCHS2	ANCHS1	ANCHS0	A/D pin
0	0	0	AN0 pin
		1	AN1 pin
	1	0	AN2 pin
		1	AN3 pin
1	0	0	AN4 pin
		1	AN5 pin
	1	0	AN6 pin
		1	AN7 pin

■Clock of A/D Converter Setup

The A/D converter clock is set by the ANCK1 to 0 flag of the ANCTR0 register. Set the A/D converter clock (TAD) more than 800 ns and less than 15.26 µs. Table 12-3-2 shows the machine clock (fosc, fx, fs) and the A/D converter clock (TAD). (calculated as fs = fosc/2, fx/4)

Table 12-3-2 A/D Conversion Clock and A/D Conversion Cycle

ANCK1	ANCK0	A/D conversion clock	A/D conversion cycle (TAD)		
			at oscillation for high speed		at oscillation for low speed
			at fosc=20 MHz	at fosc=8.38 MHz	at fx=32.768 kHz
0	0	fs/2	200.00 ns (no usable)	477.33 ns (no usable)	244.14 µs (no usable)
	1	fs/4	400.00ns (no usable)	954.65ns	488.28 µs (no usable)
1	0	fs/8	800.00 ns	1.91 µs	976.56 µs (no usable)
	1	fx x 2	15.26 µs	15.26 µs	15.26 µs

For the system clock (fs), refer to Chapter 2. 2-5 Clock Switching.

■Sampling Time (Ts) of A/D Converter Setup

The sampling time of A/D converter is set by the ANSH1 to 0 flag of the ANCTR0 register. The sampling time of A/D converter depends on external circuit, so set the right value by analog input impedance.

Table 12-3-3 Sampling Time of A/D Conversion and A/D Conversion Time

ANSH1	ANSH0	Sampling time (Ts)	A/D conversion time			
			at TAD=800 ns	at TAD=954.65 ns	at TAD=1.91 µs	at TAD=15.26 µs
0	0	TAD x 2	9.60 µs	11.46 µs	22.92 µs	183.12 µs
	1	TAD x 6	12.80 µs	15.27 µs	30.56 µs	244.16 µs
1	0	TAD x 18	22.40 µs	26.73 µs	53.48 µs	427.28 µs
	1	Reserved	-	-	-	-

■Built-in Ladder Resistor Control

The ANLADE flag of the ANCTR0 register is set to "1" to send a current to the ladder resistance for A/D conversion. As A/D converter is stopped, the ANLADE flag of the ANCTR0 register is set to "0" to save the power consumption.

Table 12-3-4 A/D Ladder Resistor Control

ANLADE	A/D ladder resistance control
0	A/D ladder resistance OFF (A/D conversion stopped)
1	A/D ladder resistance ON (A/D conversion operated)

■A/D Conversion Activation Factor Selection Setup

The A/D conversion activation factor is set by the ANSTSEL flag of the ANCTR2 register. The ANSTSEL flag of the ANCTR2 register is set to "1" to start A/D conversion by the external interrupt 3. And if the ANST flag of the ANCTR2 register is set to "1", A/D conversion can be started.

Table 12-3-5 A/D Conversion Activation Factor Selection

ANSTSEL	A/D conversion activation factor
1	The external interrupt 3, or set "1" to the ANST flag
0	Set the ANST flag to "1".

If the external interrupt 3 is selected as the A/D conversion activation factor, specify the valid edge by the REDG3 flag of the external interrupt 3 control register (IRQ3ICR), and the EDGSEL3 flag of the both edges interrupt control register (EDGDT).

[☞Chapter 3. 3-3 External Interrupts]

Specify the interrupt valid edge before the external interrupt 3 is selected as the A/D conversion activation factor.

■A/D Conversion Starting Setup

A/D conversion starting is set by the ANST flag of the ANCTR2 register. The ANST flag of the ANCTR2 register is set to "1" to start A/D conversion. When the external interrupt 3 is selected as the A/D conversion activation factor, the ANST flag of the ANCTR2 register is set to "1" to start A/D conversion, as the external interrupt 3 is generated. Also, the ANST flag of the ANCTR2 register is set to "1" during A/D conversion, then cleared to "0" as the A/D conversion complete interrupt is generated.

Table 12-3-6 A/D Conversion Starting

ANST	A/D conversion status
1	A/D conversion started or in progress.
0	A/D conversion completed or stopped

12-3-2 Setup Example

■A/D Converter Setup Example by Registers

A/D conversion is started by setting registers. The analog input pins are set to AN0, the converter clock is set to fs/4, and the sampling hold time is set to TAD x 6. Then, A/D conversion complete interrupt is generated.

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Set the analog input pin. PAIMD (x'3F3A') bp0 : PAIMD0 = 1 PAPLUD (x'3F4A') bp0 : PAPLUD0 = 0	(1) Set the analog input pin, set in (2), to the special function pin by the port A input mode register (PAIMD). Also, set no pull-up/pull-down resistance by the port A pull-up/pull-down resistance control register (PAPLUD).
(2) Select the analog input pin. ANCTR1 (x'3FB1') bp2-0 : ANCHS2-0 = 000	(2) Select the analog input pin from AN7-0 (PA7-0) by the ANCHS2-0 flag of the A/D converter control register 1 (ANCTR1).
(3) Select the A/D converter clock. ANCTR0 (x'3FB0') bp5-4 : ANCK1-0 = 01	(3) Select the A/D converter clock by the ANCK1, ANCK0 flag of the A/D converter control register 0 (ANCTR0).
(4) Set the sample and hold time. ANCTR0 (x'3FB0') bp7-6 : ANSH1-0 = 01	(4) Set the sample and hold time by the ANSH1, ANSH0 flag of the A/D converter control register 0 (ANCTR0).
(5) Set the interrupt level. ADICR (x'3FFA') bp7-6 : ADLV1-0 = 00	(5) Set the interrupt level by the ADLV1-0 flag of the A/D conversion complete interrupt control register (ADICR). If any interrupt request flag is already set, clear them. [☞ Chapter 3. 3-1-4 Interrupt Flag Setting]
(6) Enable the interrupt. ADICR (x'3FFA') bp1 : ADIE = 1	(6) Enable the interrupt by setting the ADIE flag the ADICR register to "1".
(7) Set the A/D ladder resistance. ANCTR0 (x'3FB0') bp3 : ANLADE = 1	(7) Set the ANLADE flag of the A/D converter control register 0 (ANCTR0) to "1" to send a current to the ladder resistance for the A/D conversion.

Setup Procedure	Description
(8) Start the A/D conversion. ANCTR2 (x'3FB2') bp6 : ANSTSEL = 0	(8) Set the ANSTSEL flag of the A/D converter control register 2 (ANCTR2) to "0", and select "writing to the ANST flag of the A/D converter control register 2 (ANCTR2)"as the A/D converter activation factor.
(9) Start the A/D conversion operation. ANCTR2 (x'3FB2') bp7 : ANST = 1	(9) Set the ANST flag of the A/D converter control register 2 (ANCTR2) to "1" to start the A/D conversion.
(10) Complete the A/D conversion. ANBUF0 (x'3FB3') ANBUF1 (x'3FB4')	(10) When the A/D conversion is finished, the A/D conversion complete interrupt is generated and the ANST flag of the A/D converter control register 2 (ANCTR2) is cleared to "0". The result of the conversion is stored to the A/D converter buffer (ANBUF0, 1).

Note : The above (3) to (4) can be set at once.

■A/D Conversion Setup Example by External Interrupt 3

The A/D conversion is started by the external interrupt 3. The analog input pin is set to AN0, the converter clock is set to fs/4, and the sample hold time is set to TAD x 6. Then, the A/D conversion complete interrupt is generated.

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Set the analog input pin. PAIMD (x'3F3A') bp0 : PAIMD0 = 1 PAPLUD (x'3F4A') bp0 : PAPLUD0 = 0	(1) Set the analog input pin that set in (2), to the special function pin by the port A input mode register (PAIMD). Also, set no pull-up/pull-down resistance by the port A pull-up/pull-down resistance control register (PAPLUD).
(2) Select the analog input pin. ANCTR1 (x'3FB1') bp2-0 : ANCH2-0 = 000	(2) Select the analog input pin from AN7-0 (PA7-0) by the ANCHS2-0 flag of the A/D converter control register 1 (ANCTR1).
(3) Select the A/D converter clock. ANCTR0 (x'3FB0') bp5-4 : ANCK1-0 = 01	(3) Select the A/D converter clock by the ANCK1, ANCK0 flag of the A/D converter control register 0 (ANCTR0).
(4) Set the sample hold time. ANCTR0 (x'3FB0') bp7-6 : ANSH1-0 = 01	(4) Set the sample hold time by the ANSH1, ANSH0 flag of the A/D converter control register 0 (ANCTR0).
(5) Specify the external interrupt 3 valid edge. IRQ3ICR (x'3FE5') bp5 : REDG3 = 1	(5) Specify the valid edge by the REDG3 flag of the external interrupt 3 control register (IRQ3ICR).
(6) Set the interrupt level. ADICR (x'3FFA') bp7-6 : ADLV1-0 = 10	(6) Set the interrupt level by the ADLV1-0 flag of the A/D conversion complete interrupt control register (ADICR). If any interrupt request flag is already set, clear them. [☞ Chapter 3. 3-1-4 Interrupt Flag Setup]
(7) Enable the interrupt. ADICR (x'3FFA') bp1 : ADIE = 1	(7) Enable the interrupt by setting the ADIE flag of the ADICR register to "1".

Setup Procedure	Description
(8) Set the A/D ladder resistance. ANCTR0 (x'3FB0') bp3 : ANLADE = 1	(8) Set the ANLADE flag of the A/D converter control register 0 (ANCTR0) to "1" to send a current to the ladder resistance for the A/D conversion.
(9) Select the A/D converter activation factor. ANCTR2 (x'3FB2') bp6 : ANSTSEL = 1	(9) Set the ANSTSEL flag of the A/D converter control register 2 (ANCTR2) to "1", and select "writing to the ANST flag of the A/D converter control register 3 (ANCTR3), the external interrupt 3"as the A/D converter activation factor.
(10) Start the A/D conversion. ANCTR2 (x'3FB2') bp7 : ANST = 1	(10) When the external interrupt 3, set in (5) is generated, the ANST flag of the A/D converter control register 2 (ANCTR2) is set to "1" to start the A/D conversion. And even if the external interrupt 3 is not generated, the A/D conversion is started by setting the ANST flag of the A/D converter control register 3 (ANCTR3) to "1".
(11) Complete the A/D conversion.	(11) When the A/D conversion is finished, the A/D conversion complete interrupt is generated, and the ANST flag of the A/D converter control register 2 (ANCTR2) is cleared to "0". The result of the conversion is stored to the A/D converter buffer (ANBUF0, 1).

Note : The above (3) to (4) can be set at once.

Even if the external interrupt 3 is generated during A/D conversion, the A/D converter is operated in normal.
Also, once the A/D conversion is finished, it is never started again.

12-3-3 Cautions

A/D conversion can be damaged by noise easily, hence anti-noise transaction should be operated.

■Anti-noise transaction
For A/D input (analog input pin), add condenser near the VSS pins of microcontroller.

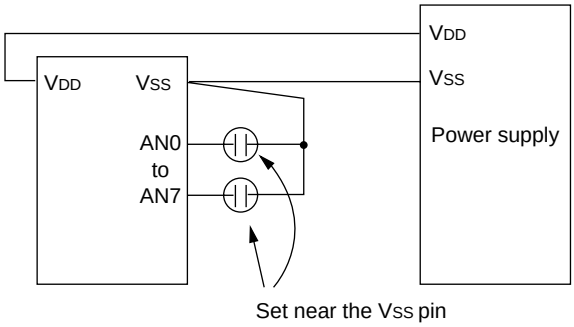
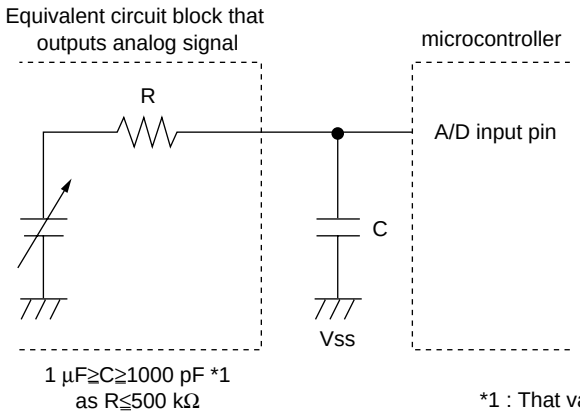


Figure 12-3-2 A/D Converter Recommended Example 1

For high precision of A/D conversion, the following cautions on A/D converter should be kept.

1. The input impedance R of A/D input pin should be under 500 kΩ[*1], and the external capacitor C (more than 1000 pF, under 1 μF)[*1].
2. The A/D conversion frequency should be set with consideration of R, C time constant.
3. At the A/D conversion, if the input level of micro controller is changed, or the peripheral added circuit is switched to ON/OFF, the A/D conversion may work wrongly, because the analog input pins and power pins does not fix. At the check of the setup, confirm the wave form of analog input pins.



Recommended Connection with A/D Converter

Chapter 13 Appendices

13-1 EPROM Version

13-1-1 Overview

EPROM version is microcomputer which was replaced the mask ROM of the MN101C539 with an electronically programmable 24 KB.

The MN101CP539HT is sealed in plastic. Once data is written to the internal PROM, it cannot be erased. We offer a 48-pin flat package of plastic.

Setting the EPROM version to EPROM mode, functions as a microcomputer are halted, and the internal EPROM can be programmed. For EPROM mode pin connection, refer to figure 13-1-2. Programming Adapter Connection.

The specification for writing to the internal EPROM are the same as for a general-purpose 256 K-bit EPROM (V_{PP}=12.5 V, tpw=1.0 ms). Therefore, by using a dedicated programming adapter (supplied by Panasonic) which can convert the 48 pin of EPROM version to 28 pin, having the same configuration as a normal EPROM, a general-purpose EPROM writer can be used to perform read and write operations.

The EPROM Version is described on the following items :

- Cautions on use of the internal EPROM
- Differences between mask ROM vers. and EPROM vers.
- Writing to the Microcomputer with internal EPROM
- Cautions on handling a ROM writer
- Programming adaptor connection

13-1-2 Cautions on Use

EPROM Version differs from the MN101C539 series mask ROM version in some of its electrical characteristics. Cautions on use of this version are as follows :

(1) Because of device characteristics of the MN101CP539HT, a writing test cannot be performed on all bits. Therefore, the reliability of data writing may not be 100% ensured.

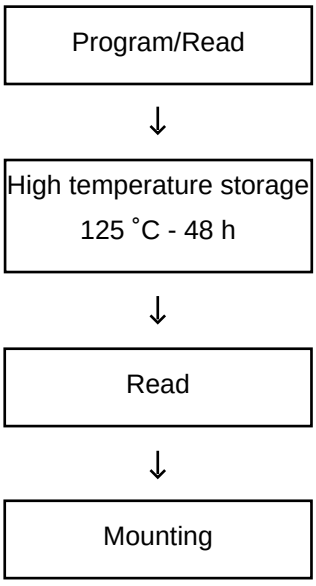
(2) When a program is written, be sure that V_{DD} power supply (6 V) is connected before applying the V_{PP} power supply (12.5 V). Disconnect the V_{PP} supply before disconnecting the V_{DD} supply.

(3) V_{PP} should never exceed 13.5 V including overshoot.

(4) If a device is removed while a V_{PP} of +12.5 V is applied, device reliability may be damaged.

(5) At NCE=V_{IL}, do not change Vpp from V_{IL} to +12.5 V or from +12.5 V to V_{IL}.

(6) After a program is written, screening at a high temperature storage is recommended before mounting.



13-1-3 Differences between Mask ROM version and EPROM version

The differences between the 8-bit microcomputer MN101C539 (Mask ROM vers.) and MN101CP539 (internal EPROM version) are as follows ;

Table 13-1-1 Differences between Mask ROM version and internal EPROM version

	MN101C539 (Mask ROM version)	MN101CP539 (EPROM version)
Operating ambient temperature	- 40 ˚C to 85 ˚C	- 20 ˚C to 85 ˚C
Operating voltage	2.0 V to 5.5 V (1.00 µs / at 2 MHz) 2.0 V to 5.5 V (62.5 µs / at 32 kHz)	2.7 V to 5.5 V (1.00 µs / at 2 MHz) 2.7 V to 5.5 V (62.5 µs / at 32.768 kHz)
Pin DC Characteristics	Output current, input current and input judge level are the same.	
Oscillation Characteristics	Matching evaluation of each oscillator is necessary when these versions are rotated for mass production	
Noise Characteristics	Matching evaluation of each oscillator is necessary when these versions are rotated for mass production	

There are no other functional differences.

13-1-4 Writing to Microcomputer with Internal EPROM

The device type that set by each ROM writer should be selected the mode for writing 256 K-bit EPROM. Set the writing voltage to 12.5 V.

■Mounting the device in the programming adapter and the position of the No.1 pin.

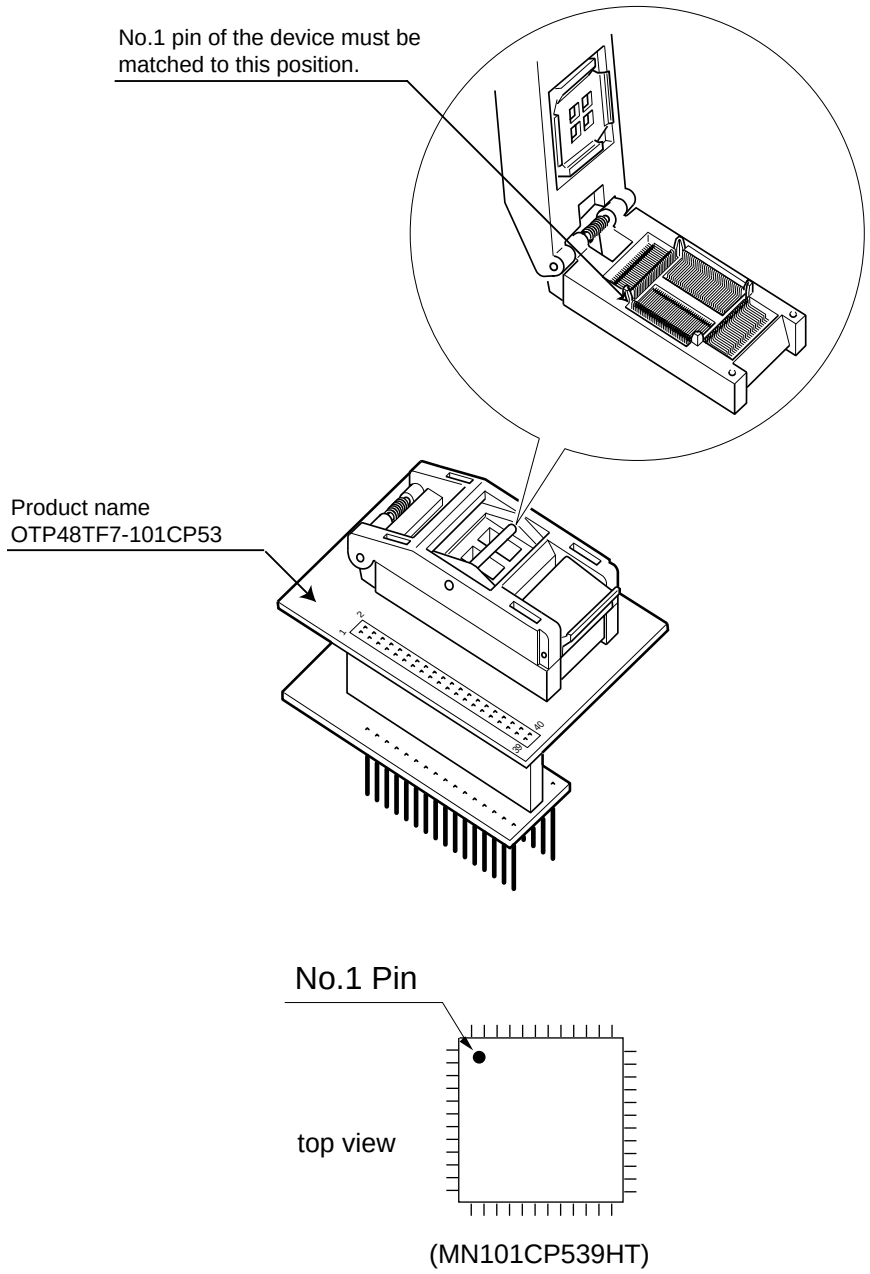


Figure 13-1-1 Mounting a Device in Programming Adapter and the Position of No.1 Pin

■ROM Writer Setup

The device types should be set up as listed below.

Table 13-1-2 Setup for Device Type

Equip. name	Vendor	Device type	Remarks
Pecker30	Aval Data	Mitsubishi 27C256	
R4945A	Advantest	Mitsubishi 27C256	
AF-9705	Ando Electronic	Mitsubishi 27C256	
LabSite	Data I/O	Mitsubishi 27C256	Do not run ID check and pin connection inspection.

The above table is based on the standard samples.

13-1-5 Cautions on Operation of ROM Writer

■Cautions on Handling the ROM writer

(1) The V_{PP} programming voltage for the EPROM versions is 12.5 V.
Programming with a 21 V ROM writer can lead to damage. The ROM writer specifications must match those for standard 256 K-bit EPROM : V_{PP}=12.5 V ; tpw=1.0 ms.

(2) Make sure that the socket adapter matches the ROM writer socket and that the chip is correctly mounted in the socket adapter. Faulty connections can damage the chip.

(3) After clearing all memory the ROM writer, load the program to the ROM writer.

(4) After confirming the device type, write the loaded program in (3) to this LSI address, from x'4000' to the final address of the internal ROM.

(5) Internal ROM of this LSI is programmed with 256 K-bit programming mode (X'0000' to X'7FFF') of the ROM writer. However, assigned address for the memory space of 24 KB EPROM is X'4000' to X'9FFF' of memory address on this LSI. Due to this address specification, not all the space can be written during OTP programming .
With regard to this problem, programming address of this LSI's hardware is assigned to X'0000' to X'5FFF'. Therefore, when OTP is programmed, address should be downshifted for what equivalent to X'4000'.

Example) To convert data X'4000' to X'9FFF' into X'0000' to X'5FFF' of Intel HEX format file.

EXCV101 -R4000, 9FFF -A0000 xxx.ex (EX format file)
*xxx.hex (Intel HEX format file)

On this LSI, program to be loaded should be converted to his LSI is from x'4000'.
[☞ MN101C series cross assembler User's manual 1.4
Activation of file conversion utility]

The internal ROM space of this LSI is from x'4000'.
[☞ Chapter 2 2-2. Memory Space]

This writer has no internal ID codes of "Silicon Signature" and "Intelligent Identifier" of the auto-device selection command of ROM writer. If the auto-device selection command is to be executed for this writer, the device is likely damaged. Therefore, never use this command.

Read the following cautions before ordering ROM.
Do not use the HEX file used for OTP programming for ordering ROM. Use the HEX file without downshifted address.
In downshifted address file, program data is stored in X'0000' to X'5FFF'.
In file without downshifted address, data X'FF is stored in X'0000' to X'3FFF', and program data is stored in X'4000' to X'9FFF'.
There is no problem in ordering ROM using EX format file.

■When the writing is disabled

When the writing is disabled, check the following points.

(1) Check that the device is mounted correctly on the socket (pin bending, connection failure).

(2) Check that the erase check result is no problem.

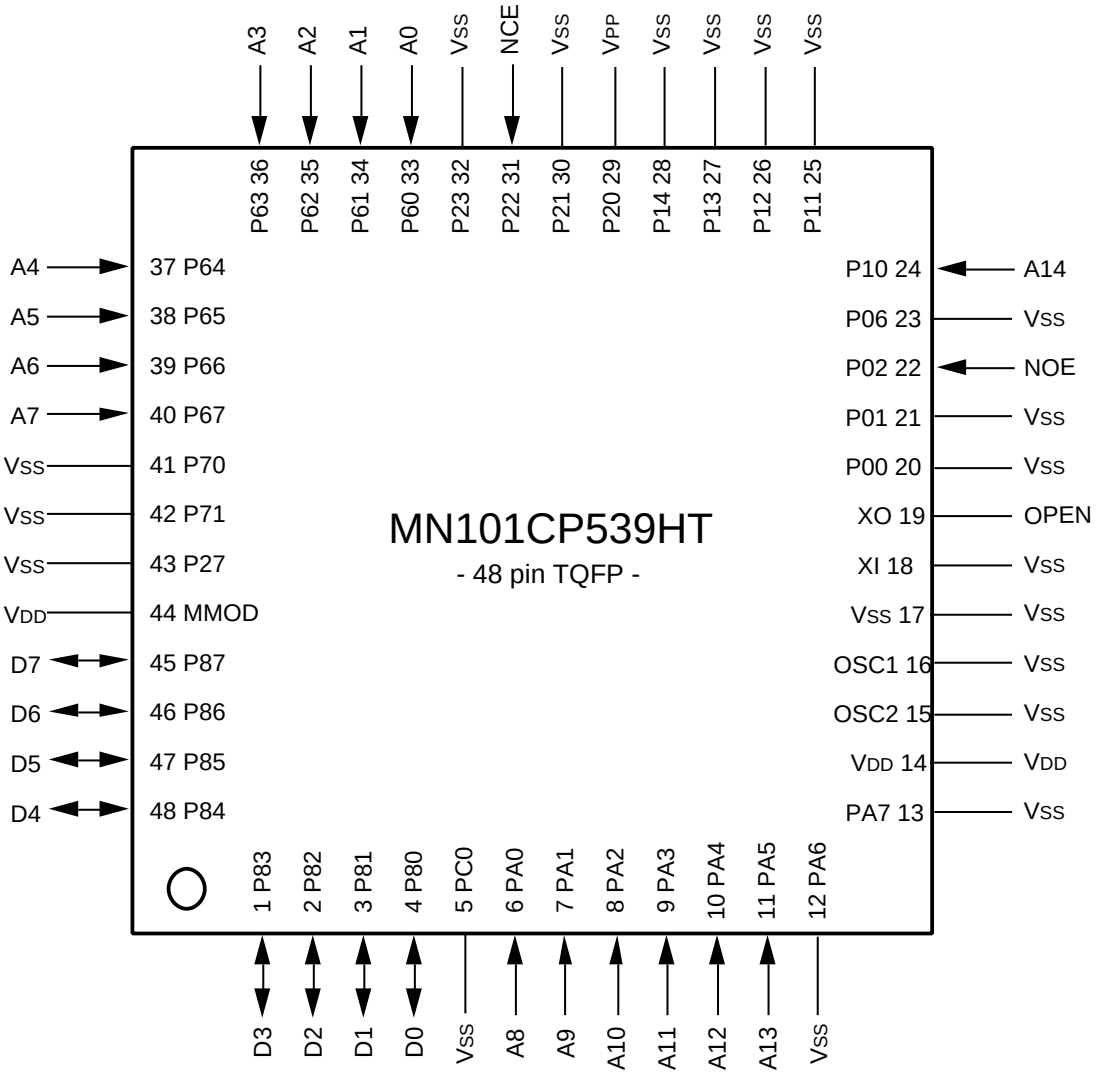
(3) Check that the adapter type is identical to the device name.

(4) Check that the writing mode is set correctly.

(5) Check that the data is correctly transferred to the ROM writer.

(6) Recheck the check points (1), (2) and (3) provided on the above paragraph of 'Cautions on Handling the ROM writer'.

13-1-6 Programming Adapter Connection



Package Code TQFP048-P-0707B

Figure 13-1-2 MN101CP539HT EPROM Programming Adapter Connection

Refer to the pin connection drawing of the 256 K-bit EPROM (27C256).

13-2 Probe Switches

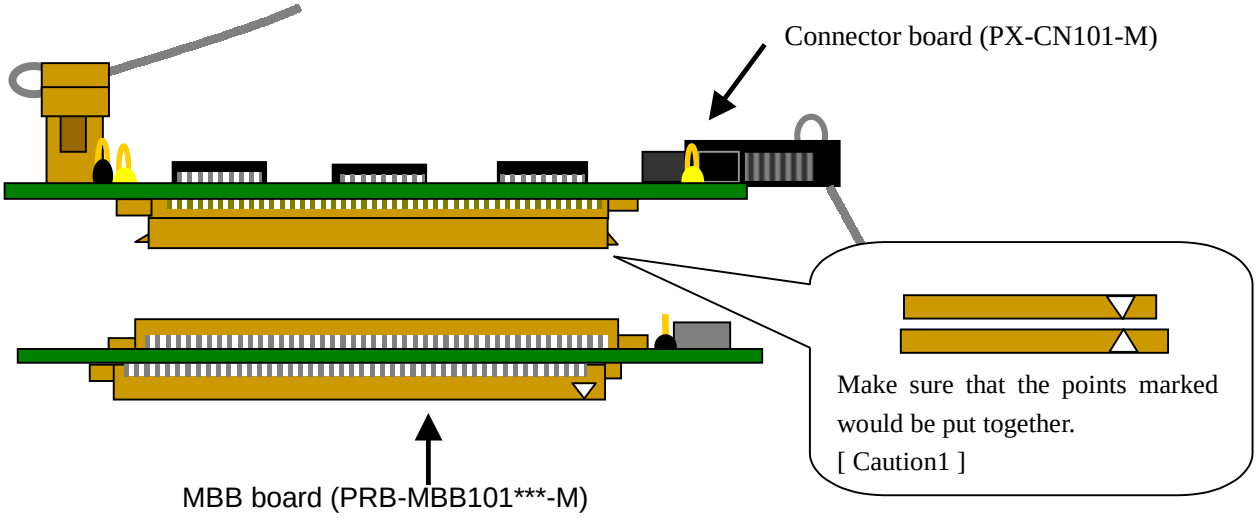
13-2-1 PX-CN101-M

This board can be used for any MBB models(product No.PRB-MBB101***-M) of MN101 series.
(Please visit our website for the latest information on the product.)

Figure1.PX-CN101-M Layout

< How to connect >

Figure2.Connecting a PX-CN101-M to a MBB board

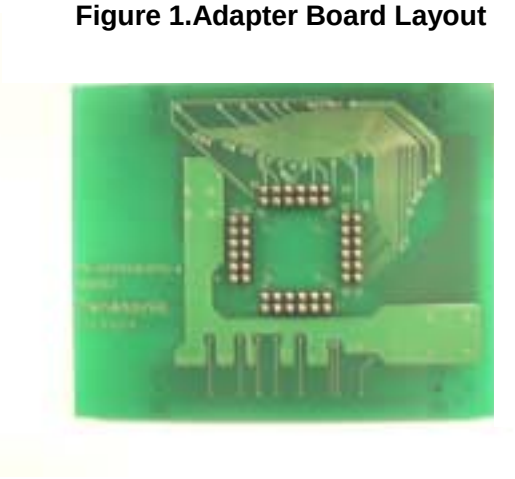


[Caution1]
Connect CNC of PX-CN101-M to CNC of PRB-MBB101***-M, and
CND of PX-CN101-M to CND of PRB-MBB101***-M.
When connecting the boards, make sure that they are connected without tilt.
If you put pressure on one side of the board, that may cause any damage to the pins.

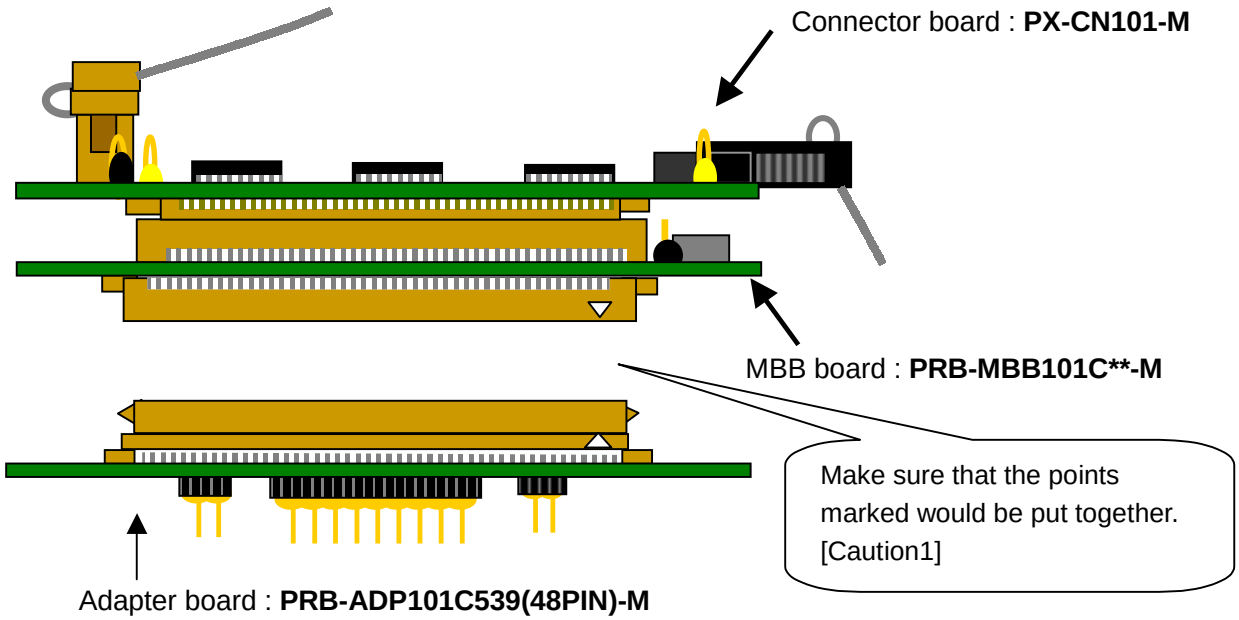
13-2-2 PRB-ADP101C539(48PIN)-M

This board can be used for only MN101C539 (48pin).
When connected to the target, use this board with PRB-MBB101C53-M
Improper matching may cause any damage to the ICE.

Figure 1.Adapter Board Layout



< How to connect >



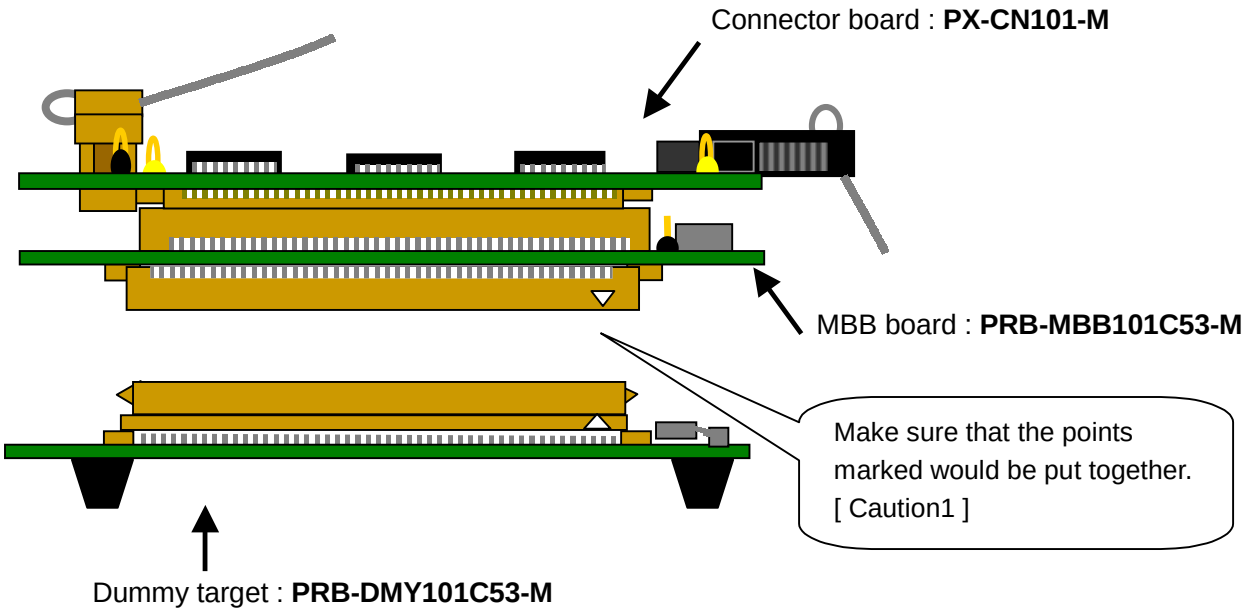
[Caution1]
Connect CNE of PRB-MBB101C**-M to CNE of PRB-ADP101C539(48PIN)-M, and CNF of PRB-MBB101C**-M to CNF of PRB-ADP101C539(48PIN)-M.
When connect the boards, make sure that they are connected without tilt.
If put pressure on one side of the board, that cause any damage to the pins.

13-2-3 PRB-DMY101C53-M

Dummy target boards differ depending upon the models. This board can be used for only 101C53.
When unconnected to the target, use this board with the PRB-MBB101C53-M.
Improper matching may cause any damage to the ICE

Figure 1.PRB-DMY101C53-M Layout

< How to connect >



[Caution1]
Connect CNE of PRB-MBB101C53-M to CNE of PRB-DMY101C53-M, and CNF of PRB-MBB101C53-M to CNF of PRB-DMY101C53-M.
When connect the boards, make sure that they are connected without tilt.
If put pressure on one side of the board, that cause any damage to the pins.

13-2-4 PRB-MBB101C53-M

- This probe must be used with the following boards.
  - Connector board : PX-CN101-M
  - MBB board : PRB-MBB101C53-M
  - Adapter board : PRB-ADP101C539(48PIN)-M
  - Dummy target : PRB-DMY101C53-M

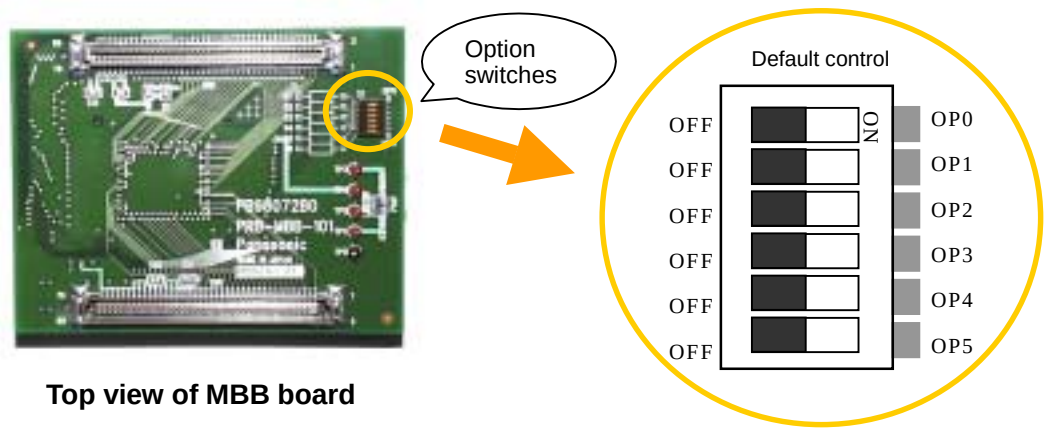
    The dummy target should be connected when ICE is operated independently, the adapter board should be connected at connection to the target.
- This probe is mounted the switches for mask option.

  The setting of the switches shown below.

Table 1.Table of setting option switches.

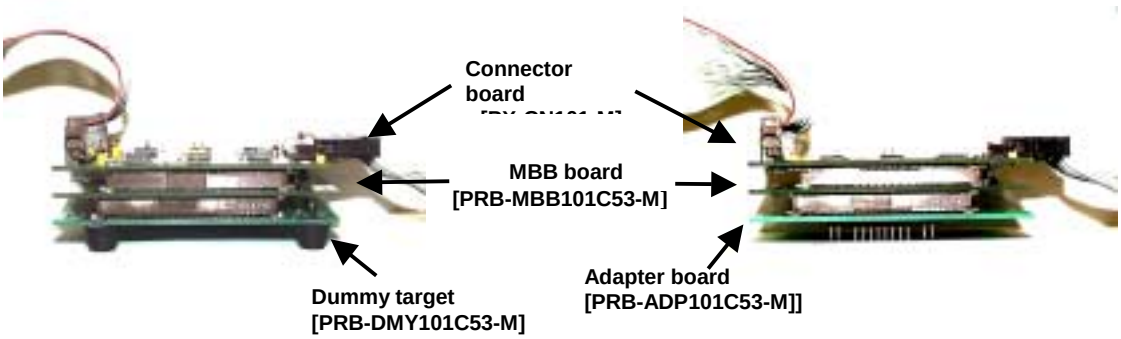
OP Number	ON	OFF
OP0	no operation	
OP1	Operate the watchdog timer at reset is released.	Stop the watchdog timer at reset is released.
OP2	Reset interruption occurs by overflow of watchdog timer of the 1st time.	NMI interruption occurs by overflow of watchdog timer of the 1st time. (If overflow of watchdog timer occurs by continuation twice, reset will occur.)
OP3	Reset by the hardware at watchdog timer count No2^14	Reset by the hardware at watchdog timer count No 2^15
OP4	no operation	
OP5	no operation	

Figure1.Layout of option switches



Top view of MBB board

Figure2.Composition with PRB-MBB101C53-M



- When ICE is operated independently.

- At connection to the target

13-3 Special Function Registers List

Address	Register	Bit Symbol / Initial Value / Description								Page
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
X'3F00'	CPUM	-	OSCSEL1	OSCSEL0	OSCDBL	STOP	HALT	OSC1	OSC0	II - 23
		-	1	1	0	0	0	0	0	II - 19
			Division Rate Setup		Internal System Clock Setup	STOP mode Setup	HALT mode Setup	Oscillation Control		
X'3F01'	MEMCTR	IOW1	IOW0	IVBM	EXMEM	-	IRWE	-	-	II - 16
		1	1	0	0	-	0	-	-	
		I/O Wait Setup		Interrupt Vector Address	Set always to "0"	Switch Wait	Software Writes Setup			
X'3F02'	WDCTR	-	-	WDTC2	WDTC1	WDTC0	WDTS1	WDTS0	WDEN	IX - 3
		-	-	0	0	0	1	1	0	
				The lowest value for clear Setup			Watchdog Time-out Period Setup		WDT Activation	
X'3F03'	DLYCTR	BUZOE	BUZS2	BUZS1	BUZS0	DLYS1	DLYS0	-	-	II - 30
		0	0	0	0	1	0	-	-	X - 3
		Enable Buzzer Output	Buzzer Output Frequency Setup			Oscillation Stabilization Wait Cycle Setup				
X'3F0D'	OSCMD	-	-	-	-	-	-	SOSC2DS	RESERVED	II - 23
		-	-	-	-	-	-	0	0	
								Low Frequency Divided by 2	Set always to "0"	
X'3F10'	P0OUT	-	P0OUT6	-	-	-	P0OUT2	P0OUT1	P0OUT0	IV - 7
		-	x	-	-	-	x	x	x	
			Port 0 Output Data				Port 0 Output Data			
X'3F11'	P1OUT	-	-	-	P1OUT4	P1OUT3	P1OUT2	P1OUT1	P1OUT0	IV - 12
		-	-	-	x	x	x	x	0	
					Port 1 Output Data					
X'3F12'	P2OUT	P2OUT7	-	-	-	-	-	-	-	IV - 17
		1	-	-	-	-	-	-	-	
		Port 2 Output Data								
X'3F16'	P6OUT	P6OUT7	P6OUT6	P6OUT5	P6OUT4	P6OUT3	P6OUT2	P6OUT1	P6OUT0	IV - 20
		x	x	x	x	x	x	x	x	
		Port 6 Output Data								
X'3F17'	P7OUT	-	-	-	-	-	-	P7OUT1	P7OUT0	IV - 23
		-	-	-	-	-	-	x	x	
								Port 7 Output Data		
X'3F18'	P8OUT	P8OUT7	P8OUT6	P8OUT5	P8OUT4	P8OUT3	P8OUT2	P8OUT1	P8OUT0	IV - 27
		x	x	x	x	x	x	x	x	
		Port 8 Output Data								
X'3F1A'	PAOUT	PAOUT7	PAOUT6	PAOUT5	PAOUT4	PAOUT3	PAOUT2	PAOUT1	PAOUT0	IV - 30
		x	x	x	x	x	x	x	x	
		Port A Output Data								
X'3F1C'	PCOUT	-	-	-	-	-	-	-	PCOUT0	IV - 34
		-	-	-	-	-	-	-	x	
									Port C Output Data	

Note) x : Initial value is unstable. - : No register is allocated.

Address	Register	Bit Symbol / Initial Value / Description								Page
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
X'3F20'	P0IN	-	P0IN6	-	-	-	P0IN2	P0IN1	P0IN0	IV - 7
		-	x	-	-	-	x	x	x	
			Port 0 Intput Data				Port 0 Intput Data			
X'3F21'	P1IN	-	-	-	P1IN4	P1IN3	P1IN2	P1IN1	P1IN0	IV - 12
		-	-	-	x	x	x	x	x	
					Port 1 Intput Data					
X'3F22'	P2IN	P2IN7	-	-	-	P2IN3	P2IN2	P2IN1	P2IN0	IV - 17
		1	-	-	-	x	x	x	x	
		Port 2 Intput Data				Port 2 Intput Data				
X'3F26'	P6IN	P6IN7	P6IN6	P6IN5	P6IN4	P6IN3	P6IN2	P6IN1	P6IN0	IV - 20
		x	x	x	x	x	x	x	x	
		Port 6 Intput Data								
X'3F27'	P7IN	-	-	-	-	-	-	P7IN1	P7IN0	IV - 23
		-	-	-	-	-	-	x	x	
								Port 7 Intput Data		
X'3F28'	P8IN	P8IN7	P8IN6	P8IN5	P8IN4	P8IN3	P8IN2	P8IN1	P8IN0	IV - 27
		x	x	x	x	x	x	x	x	
		Port 8 Intput Data								
X'3F2A'	PAIN	PAIN7	PAIN6	PAIN5	PAIN4	PAIN3	PAIN2	PAIN1	PAIN0	IV - 30
		x	x	x	x	x	x	x	x	
		Port A Intput Data								
X'3F2C'	PCIN	-	-	-	-	-	-	-	PCIN0	IV - 34
		-	-	-	-	-	-	-	x	
									Port C Intput Data	
X'3F2E'	FLOAT	-	PARDWN	-	P7RDWN	-	-	-	-	IV - 24,31
		-	0	-	0	-	-	-	-	
			PA Pull up/down Selection		P7 Pull up/down Selection					
X'3F2F'	P1OMD	-	-	-	P1OMD4	P1OMD3	P1OMD2	P1OMD1	P1OMD0	IV - 13
		-	-	-	0	0	0	0	0	
					Port 1 Special Function Pin Output Control					
X'3F30'	P0DIR	-	P0DIR6	-	-	-	P0DIR2	P0DIR1	P0DIR0	IV - 7
		-	0	-	-	-	0	0	0	
			Port 0 I/O Direction Control				Port 0 I/O Direction Control			
X'3F31'	P1DIR	-	-	-	P1DIR4	P1DIR3	P1DIR2	P1DIR1	P1DIR0	IV - 12
		-	-	-	0	0	0	0	1	
					Port 1 I/O Direction Control					
X'3F36'	P6DIR	P6DIR7	P6DIR6	P6DIR5	P6DIR4	P6DIR3	P6DIR2	P6DIR1	P6DIR0	IV - 20
		0	0	0	0	0	0	0	0	
		Port 6 I/O Direction Control								
X'3F37'	P7DIR	-	-	-	-	-	-	P7DIR1	P7DIR0	IV - 23
		-	-	-	-	-	-	0	0	
								Port 7 I/O Direction Control		

Note) x : Initial value is unstable. - : No register is allocated.

Address	Register	Bit Symbol / Initial Value / Description								Page
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
X'3F38'	P8DIR	P8DIR7	P8DIR6	P8DIR5	P8DIR4	P8DIR3	P8DIR2	P8DIR1	P8DIR0	IV - 27
		0	0	0	0	0	0	0	0	
		Port 8 I/O Direction Control								
X'3F3A'	PAIMD	PAIMD7	PAIMD6	PAIMD5	PAIMD4	PAIMD3	PAIMD2	PAIMD1	PAIMD0	IV - 30
		0	0	0	0	0	0	0	0	
		Port A Analog Input Selection								
X'3F3C'	PCDIR	-	-	-	-	-	-	-	PCDIR0	IV - 34
		-	-	-	-	-	-	-	0	
									Port C I/O Direction Control	
X'3F3F'	PADIR	PADIR7	PADIR6	PADIR5	PADIR4	PADIR3	PADIR2	PADIR1	PADIR0	IV - 30
		0	0	0	0	0	0	0	0	
		Port A I/O Direction Control								
X'3F40'	P0PLU	-	P0PLU6	-	-	-	P0PLU2	P0PLU1	P0PLU0	IV - 7
		-	0	-	-	-	0	0	0	
			Port 0 Pull-up Control				Port 0 Pull-up Control			
X'3F41'	P1PLU	-	-	-	P1PLU4	P1PLU3	P1PLU2	P1PLU1	P1PLU0	IV - 12
		-	-	-	0	0	0	0	0	
					Port 1 Pull-up Control					
X'3F42'	P2PLU	-	-	-	-	P2PLU3	P2PLU2	P2PLU1	P2PLU0	IV - 17
		-	-	-	-	0	0	0	0	
						Port 2 Pull-up Control				
X'3F46'	P6PLU	P6PLU7	P6PLU6	P6PLU5	P6PLU4	P6PLU3	P6PLU2	P6PLU1	P6PLU0	IV - 20
		0	0	0	0	0	0	0	0	
		Port 6 Pull-up Control								
X'3F47'	P7PLUD	-	-	-	-	-	-	P7PLUD1	P7PLUD0	IV - 23
		-	-	-	-	-	-	0	0	
								Port 7 Pull-up/Pull-down Control		
X'3F48'	P8PLU	P8PLU7	P8PLU6	P8PLU5	P8PLU4	P8PLU3	P8PLU2	P8PLU1	P8PLU0	IV - 27
		0	0	0	0	0	0	0	0	
		Port 8 Pull-up Control								
X'3F4A'	PAPLUD	PAPLUD7	PAPLUD6	PAPLUD5	PAPLUD4	PAPLUD3	PAPLUD2	PAPLUD1	PAPLUD0	IV - 31
		0	0	0	0	0	0	0	0	
		Port A Pull-up/Pull-down Control								
X'3F4C'	PCPLU	-	-	-	-	-	-	-	PCPLU0	IV - 34
		-	-	-	-	-	-	-	0	
									Port C Pull-up Control	
X'3F4E'	P6IMD	P6KYEN7	P6KYEN6	P6KYEN5	P6KYEN4	P6KYEN3	P6KYEN2	P6KYEN1	P6KYEN0	III - 37
		0	0	0	0	0	0	0	0	
		Port 6 Key Input Interrupt Pin Selection								
X'3F4F'	PAODC	PAODC7	PAODC6	PAODC5	PAODC4	PAODC3	PAODC2	PAODC1	PAODC0	IV - 30
		0	0	0	0	0	0	0	0	
		Port A Output Style Selection								

Note) x : Initial value is unstable. - : No data

Address	Register	Bit Symbol / Initial Value / Description								Page
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
X'3F58'	TM2BC	TM2BC7	TM2BC6	TM2BC5	TM2BC4	TM2BC3	TM2BC2	TM2BC1	TM2BC0	VI - 6
		x	x	x	x	x	x	x	x	
		Timer 2 Binary Counter								
X'3F59'	TM3BC	TM3BC7	TM3BC6	TM3BC5	TM3BC4	TM3BC3	TM3BC2	TM3BC1	TM3BC0	VI - 6
		x	x	x	x	x	x	x	x	
		Timer 3 Binary Counter								
X'3F5A'	TM2OC	TM2OC7	TM2OC6	TM2OC5	TM2OC4	TM2OC3	TM2OC2	TM2OC1	TM2OC0	VI - 6
		x	x	x	x	x	x	x	x	
		Timer 2 Output Compare Register								
X'3F5B'	TM3OC	TM3OC7	TM3OC6	TM3OC5	TM3OC4	TM3OC3	TM3OC2	TM3OC1	TM3OC0	VI - 6
		x	x	x	x	x	x	x	x	
		Timer 3 Output Compare Register								
X'3F5C'	TM2MD	-	-	TM2MOD	TM2PWM	TM2EN	TM2CK2	TM2CK1	TM2CK0	VI - 7
		-	-	0	0	0	0	0	0	
				Timer 2 Pulse Width Measurement	PWM Operation Selection	Timer 2 Count Control	Timer 2 Clock Source Selection			
X'3F5D'	TM3MD	-	-	-	TM3CAS	TM3EN	TM3CK2	TM3CK1	TM3CK0	VI - 8
		-	-	-	0	0	0	0	0	
					Cascade Selection	Timer 3 Count Control	Timer 3 Clock Source Selection			
X'3F5E'	CK2MD	-	-	-	-	-	TM2PSC1	TM2PSC0	TM2BAS	V - 7
		-	-	-	-	-	x	x	x	
							Timer 2 Count Clock Setting (Prescaler Output)			
X'3F5F'	CK3MD	-	-	-	-	-	TM3PSC1	TM3PSC0	TM3BAS	V - 7
		-	-	-	-	-	x	x	x	
							Timer 3 Count Clock Setting (Prescaler Output)			
X'3F68'	TM6BC	TM6BC7	TM6BC6	TM6BC5	TM6BC4	TM6BC3	TM6BC2	TM6BC1	TM6BC0	VIII - 5
		x	x	x	x	x	x	x	x	
		Timer 6 Binary Counter								
X'3F69'	TM6OC	TM6OC7	TM6OC6	TM6OC5	TM6OC4	TM6OC3	TM6OC2	TM6OC1	TM6OC0	VIII - 5
		x	x	x	x	x	x	x	x	
		Timer 6 Output Compare Register								
X'3F6A'	TM6MD	TM6CLRS	TM6IR2	TM6IR1	TM6IR0	TM6CK3	TM6CK2	TM6CK1	TM6CK0	VIII - 6
		0	0	0	0	0	0	0	0	
		Counter Clear Selection	Time Base Timer Interrupt Cycle Selection			Timer 6 Clock Source Selection			Time Base Timer Clock Source Selection	
X'3F6B'	TBCLR	-	-	-	-	-	-	-	-	VIII - 5
		-	-	-	-	-	-	-	-	
		Time Base Timer Clear Control Register (For Writing Only)								
X'3F6E'	RMCTR	-	RMETM2	RTM7EN	TM7RM	RMOEN	RMDTY1	RMDTY0	RMBTMS	VI - 9 VII - 11
		-	0	0	1	0	0	0	0	
			IRQ2 Hardware Control Selecrion	Timer 7 Count Control	P10 Special Function Output Selection	Remote Control Output Enable	Remote Control Duty Selection		Remote Control Base Timer Slection	
X'3F6F'	PSCMD	-	-	-	-	-	-	-	PSCEN	V - 6
		-	-	-	-	-	-	-	0	
									Prescaler Count Control	

Note) x : Initial value is unstable. - : No register is allocated.

Address	Register	Bit Symbol / Initial Value / Description								Page
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
X'3F70'	TM7BCL	TM7BCL7	TM7BCL6	TM7BCL5	TM7BCL4	TM7BCL3	TM7BCL2	TM7BCL1	TM7BCL0	VII - 8
		x	x	x	x	x	x	x	x	
		Timer 7 Binary Counter Lower 8 bits								
X'3F71'	TM7BCH	TM7BCH7	TM7BCH6	TM7BCH5	TM7BCH4	TM7BCH3	TM7BCH2	TM7BCH1	TM7BCH0	VII - 8
		x	x	x	x	x	x	x	x	
		Timer 7 Binary Counter Upper 8 bits								
X'3F72'	TM7OC1L	TM7OC1L7	TM7OC1L6	TM7OC1L5	TM7OC1L4	TM7OC1L3	TM7OC1L2	TM7OC1L1	TM7OC1L0	VII - 6
		x	x	x	x	x	x	x	x	
		Timer 7 Output Compare Register 1 Lower 8 bits								
X'3F73'	TM7OC1H	TM7OC1H7	TM7OC1H6	TM7OC1H5	TM7OC1H4	TM7OC1H3	TM7OC1H2	TM7OC1H1	TM7OC1H0	VII - 6
		x	x	x	x	x	x	x	x	
		Timer 7 Output Compare Register 1 Upper 8 bits								
X'3F74'	TM7PR1L	TM7PR1L7	TM7PR1L6	TM7PR1L5	TM7PR1L4	TM7PR1L3	TM7PR1L2	TM7PR1L1	TM7PR1L0	VII - 7
		x	x	x	x	x	x	x	x	
		Timer 7 Preset Register 1 Lower 8 bits								
X'3F75'	TM7PR1H	TM7PR1H7	TM7PR1H6	TM7PR1H5	TM7PR1H4	TM7PR1H3	TM7PR1H2	TM7PR1H1	TM7PR1H0	VII - 7
		x	x	x	x	x	x	x	x	
		Timer 7 Preset Register 1 Upper 8 bits								
X'3F76'	TM7ICL	TM7ICL7	TM7ICL6	TM7ICL5	TM7ICL4	TM7ICL3	TM7ICL2	TM7ICL1	TM7ICL0	VII - 8
		x	x	x	x	x	x	x	x	
		Timer 7 Input Capture Register Lower 8 bits								
X'3F77'	TM7ICH	TM7ICH7	TM7ICH6	TM7ICH5	TM7ICH4	TM7ICH3	TM7ICH2	TM7ICH1	TM7ICH0	VII - 8
		x	x	x	x	x	x	x	x	
		Timer 7 Input Capture Register Upper 8 bits								
X'3F78'	TM7MD1	RESERVED	TM7CNC	TM7CL	TM7EN	TM7PS1	TM7PS0	TM7CK1	TM7CK0	VII - 9
		0	0	1	0	0	0	0	0	
		Set always to "0"	Timer 7 Count Control Switching	Timer 7 Output Reset Control	Timer 7 Count Control	Timer 7 Count Clock Selection		Timer 7 Clock Source Selection		
X'3F79'	TM7MD2	T7ICEDG	T7PWMSL	TM7BCR	TM7PWM	TM7IRS1	T7ICEN	T7ICT1	T7ICT0	VII - 10
		0	0	0	0	0	0	0	0	
		Capture Trigger Edge Selection	PWM Mode Selection	Clear Factor Selection	Timer 7 Output Selection	Timer 7 Interrupt Factor Selection	Enable Capture Operation	Timer 7 Capture Trigger Selection		
X'3F7A'	TM7OC2L	TM7OC2L7	TM7OC2L6	TM7OC2L5	TM7OC2L4	TM7OC2L3	TM7OC2L2	TM7OC2L1	TM7OC2L0	VII - 6
		x	x	x	x	x	x	x	x	
		Timer 7 Output Compare Register 2 Lower 8 Bits								
X'3F7B'	TM7OC2H	TM7OC2H7	TM7OC2H6	TM7OC2H5	TM7OC2H4	TM7OC2H3	TM7OC2H2	TM7OC2H1	TM7OC2H0	VII - 6
		x	x	x	x	x	x	x	x	
		Timer 7 Output Compare Register 2 Upper 8 Bits								
X'3F7C'	TM7PR2L	TM7PR2L7	TM7PR2L6	TM7PR2L5	TM7PR2L4	TM7PR2L3	TM7PR2L2	TM7PR2L1	TM7PR2L0	VII - 7
		x	x	x	x	x	x	x	x	
		Timer 7 Preset Register 2 Lower 8 bits								
X'3F7D'	TM7PR2H	TM7PR2H7	TM7PR2H6	TM7PR2H5	TM7PR2H4	TM7PR2H3	TM7PR2H2	TM7PR2H1	TM7PR2H0	VII - 7
		x	x	x	x	x	x	x	x	
		Timer 7 Preset Register 2 Upper 8 bits								

Note) x : Initial value is unstable. - : No register is allocated.

Address	Register	Bit Symbol / Initial Value / Description								Page
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
X'3F8E'	NFCTR	P21IM	NF1SCK1	NF1SCK0	NF1EN	-	NF0SCK1	NF0SCK0	NF0EN	III - 35
		0	0	0	0	-	0	0	0	
		ACZ Input Enable Flag	IRQ1 Noise Filter Sampling Sampling Period Selection		IRQ1 Noise Filter Enable		IRQ0 Noise Filter Sampling Sampling Period Selection		IRQ0 Noise Filter Enable	
X'3F8F'	EDGDT	IRQ3SEL	IRQ2SEL	-	-	-	-	-	-	III - 36
		0	0	-	-	-	-	-	-	
		IRQ3 Interrupt Source Selecrion	IRQ2 Interrupt Source Selecrion					IRQ1 Both Edge Specification	IRQ0 Both Edge Specification	
X'3F90'	SC0MD0	SC0CE1	RESERVED	RESERVED	SC0DIR	SC0STE	SC0LNG2	SC0LNG1	SC0LNG0	XI - 6
		0	0	0	0	0	1	1	1	
		Transmission Data/ Recived Data Edge Selection	fixed to 0	fixed to 0	Specify First Bit to be Transfered	Start Condition Selection	Synchronous Serial Transfer Bit Count Selection			
X'3F91'	SC0MD1	SC0IOM	SC0SBTS	SC0SBIS	SC0SBOS	SC0CKM	SC0MST	-	SC0CMD	XI - 7
		0	0	0	0	0	0	-	0	
		Data Input Pin Selection	SBT Pin Func-tion Selection	Serial Input Control	SBO Pin Func-tion Selection	Clock Divided by 8 Selection	Master/Slave Selection		Synchronous/ UART Selection	
X'3F92'	SC0MD2	SC0FM1	SC0FM0	SC0PM1	SC0PM0	SC0NPE	-	SC0BRKF	SC0BRKE	XI - 8
		0	0	0	0	0	-	0	0	
		Specify Flame Mode		Added Bit Specification		Parity Enable		Break Status Receive Monitor	Break Status Transmit Monitor	
X'3F93'	SC0STR	SC0TBSY	SC0RBSY	SC0TEMP	SC0REMP	SC0FEF	SC0PEK	SC0ORE	SC0ERE	XI - 9
		0	0	0	0	0	0	0	0	
		Transmission flag	Reception flag	Transfer Buffer Empty Flag	Receive Buffer Empty Flag	Framing Error Detection	Parity Error Detection	Overrun Error Detection	Error Monitor Flag	
X'3F94'	RXBUF0	RXBUF07	RXBUF06	RXBUF05	RXBUF04	RXBUF03	RXBUF02	RXBUF01	RXBUF00	XI - 5
		x	x	x	x	x	x	x	x	
		Serial Interface 0 Receive Buffer								
X'3F95'	TXBUF0	TXBUF07	TXBUF06	TXBUF05	TXBUF04	TXBUF03	TXBUF02	TXBUF01	TXBUF00	XI - 5
		x	x	x	x	x	x	x	x	
		Serial Interface 0 Transfer Buffer								
X'3F96'	SC0ODC	SC0FDC1	SC0FDC0	-	-	-	SC0ODC2	SC0ODC1	SC0ODC0	IV - 8 XI - 10
		0	0	-	-	-	0	0	0	
		Output Selection After Last Bit Transmission					P02 Output Style Selection	P01 Output Style Selection	P00 Output Style Selection	
X'3F97'	SC0CKS	-	-	-	-	SC0TMSEL	SC0PSC2	SC0PSC1	SC0PSC0	V - 8 XI - 11
		-	-	-	-	x	x	x	x	
						Serial 0 Tranfer Clock Selection (Prescaler Output, Timer Output)				
X'3FB0'	ANCTR0	ANSH1	ANSH0	ANCK1	ANCK0	ANLADE	-	-	-	XII - 5
		0	0	0	0	0	-	-	-	
		A/D Sample Hold Time Setup		A/D Conversion Clock Selection		A/D Rudder Resistance Control				
X'3FB1'	ANCTR1	-	-	-	-	RESERVED	ANCHS2	ANCHS1	ANCHS0	XII - 6
		-	-	-	-	0	0	0	0	
						Set always to "0"	Analog Input Channel Selection			
X'3FB2'	ANCTR2	ANST	ANSTSEL	-	-	-	-	-	-	XII - 6
		0	0	-	-	-	-	-	-	
		A/D Conversion Status	A/D Conversion Start Source Selection							
X'3FB3'	ANBUF0	ANBUF07	ANBUF06	-	-	-	-	-	-	XII - 7
		x	x	-	-	-	-	-	-	
		A/D Conversion Data Storage Register (Lower 2 bits)								

Note) x : Initial value is unstable. - : No register is allocated.

Address	Register	Bit Symbol / Initial Value / Description								Page
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
X'3FB4'	ANBUF1	ANBUF17	ANBUF16	ANBUF15	ANBUF14	ANBUF13	ANBUF12	ANBUF11	ANBUF10	XII - 7
		x	x	x	x	x	x	x	x	
		A/D Conversion Data Storage Register (Upper 2 bits)								
X'3FE1'	NMICR	-	-	-	-	-	PIR	WDIR	RESERVED	III - 16
		-	-	-	-	-	0	0	0	
							Program Interrupt Request	Watchdog Timer Interrupt Request	Set Always to "0"	
X'3FE2'	IRQ0ICR	IRQ0LV1	IRQ0LV0	REDG0	-	-	-	IRQ0IE	IRQ0IR	III - 17
		0	0	0	-	-	-	0	0	
		Specify IRQ0 Interrupt Level		IRQ0 Interrupt Valid Edge				Enable IRQ0 Interrupt	Request IRQ0 Interrupt	
X'3FE3'	IRQ1ICR	IRQ1LV1	IRQ1LV0	REDG1	-	-	-	IRQ1IE	IRQ1IR	III - 18
		0	0	0	-	-	-	0	0	
		Specify IRQ1 Interrupt Level		IRQ1 Interrupt Valid Edge				Enable IRQ1 Interrupt	Request IRQ1 Interrupt	
X'3FE4'	IRQ2ICR	IRQ2LV1	IRQ2LV0	REDG2	-	-	-	IRQ2IE	IRQ2IR	III - 19
		0	0	0	-	-	-	0	0	
		Specify IRQ2 Interrupt Level		IRQ2 Interrupt Valid Edge				Enable IRQ2 Interrupt	Request IRQ2 Interrupt	
X'3FE5'	IRQ3ICR	IRQ3LV1	IRQ3LV0	REDG3	-	-	-	IRQ3IE	IRQ3IR	III - 20
		0	0	0	-	-	-	0	0	
		Specify IRQ3 Interrupt Level		IRQ3 Interrupt Valid Edge				Enable IRQ3 Interrupt	Request IRQ3 Interrupt	
X'3FEB'	TM2ICR	TM2LV1	TM2LV0	-	-	-	-	TM2IE	TM2IR	III - 21
		0	0	-	-	-	-	0	0	
		Specify TM2 Interrupt Level						Enable TM2 Interrupt	Request TM2 Interrupt	
X'3FEC'	TM3ICR	TM3LV1	TM3LV0	-	-	-	-	TM3IE	TM3IR	III - 22
		0	0	-	-	-	-	0	0	
		Specify TM3 Interrupt Level						Enable TM3 Interrupt	Request TM3 Interrupt	
X'3FEF'	TM6ICR	TM6LV1	TM6LV0	-	-	-	-	TM6IE	TM6IR	III - 23
		0	0	-	-	-	-	0	0	
		Specify TM6 Interrupt Level						Enable TM6 Interrupt	Request TM6 Interrupt	
X'3FF0'	TBICR	TBLV1	TBLV0	-	-	-	-	TBIE	TBIR	III - 24
		0	0	-	-	-	-	0	0	
		Specify TB Interrupt Level						Enable TB Interrupt	Request TB Interrupt	
X'3FF1'	TM7ICR	TM7LV1	TM7LV0	-	-	-	-	TM7IE	TM7IR	III - 25
		0	0	-	-	-	-	0	0	
		Specify TM7 Interrupt Level						Enable TM7 Interrupt	Request TM7 Interrupt	
X'3FF2'	T7OC2ICR	T7OC2LV1	T7OC2LV0	-	-	-	-	T7OC2IE	T7OC2IR	III - 26
		0	0	-	-	-	-	0	0	
		Specify T7OC2 Interrupt Level						Enable T7OC2 Interrupt	Request T7OC2 Interrupt	
X'3FF5'	SC0RICR	SC0RLV1	SC0RLV0	-	-	-	-	SC0RIE	SC0RIR	III - 27
		0	0	-	-	-	-	0	0	
		Specify SC0R Interrupt Level						Enable SC0R Interrupt	Request SC0R Interrupt	
X'3FF6'	SC0TICR	SC0TLV1	SC0TLV0	-	-	-	-	SC0TIE	SC0TIR	III - 28
		0	0	-	-	-	-	0	0	
		Specify SC0T Interrupt Level						Enable SC0T Interrupt	Request SC0T Interrupt	

Note) x : Initial value is unstable. - : No register is allocated.

Specify SC0T Interrupt Level Enable SC0T Interrupt Request SC0T Interrupt

Address	Register	Bit Symbol / Initial Value / Description								Page
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
X'3FFA'	ADICR	ADLV1	ADLV0	-	-	-	-	ADIE	ADIR	III - 29
		0	0	-	-	-	-	0	0	
		Specify AD Interrupt Level						Enable AD Interrupt	Request AD Interrupt	

Note) x : Initial value is unstable. - : No register is allocated.

13-4 Instruction Set

MN101C SERIES INSTRUCTION SET

Group	Mnemonic	Operation	Flag VF	Flag NF	Flag CF	Flag ZF	Code Size	Cycle	Re-peat	Machine Code Ext.	1	2	3	4	5	6	7	8	9	10	11	Notes
Data Move Instructions																						
MOV	MOV Dn,Dm	Dn→Dm	--	--	--	--	2	1		1010 DnDm												
	MOV imm8,Dm	imm8→Dm	--	--	--	--	4	2		1010 DmDm	<#8. ...>											
	MOV Dn,PSW	Dn→PSW	●	●	●	●	3	3		0010 1001 01Dn												
	MOV PSW,Dm	PSW→Dm	--	--	--	--	3	2		0010 0001 01Dm												
	MOV (An),Dm	mem8(An)→Dm	--	--	--	--	2	2		0100 1ADm												
	MOV (d8,An),Dm	mem8(d8+An)→Dm	--	--	--	--	4	2		0110 1ADm	<d8. ...>											*1
	MOV (d16,An),Dm	mem8(d16+An)→Dm	--	--	--	--	7	4		0010 0110 1ADm	<d16	....	...>									
	MOV (d4,SP),Dm	mem8(d4+SP)→Dm	--	--	--	--	3	2		0110 01Dm	<d4>											*2
	MOV (d8,SP),Dm	mem8(d8+SP)→Dm	--	--	--	--	5	3		0010 0110 01Dm	<d8. ...>											*3
	MOV (d16,SP),Dm	mem8(d16+SP)→Dm	--	--	--	--	7	4		0010 0110 00Dm	<d16	....	...>									
	MOV (io8),Dm	mem8(IOTOP+io8)→Dm	--	--	--	--	4	2		0110 00Dm	<io8 ...>											
	MOV (abs8),Dm	mem8(abs8)→Dm	--	--	--	--	4	2		0100 01Dm	<abs 8..>											
	MOV (abs12),Dm	mem8(abs12)→Dm	--	--	--	--	5	2		0100 00Dm	<abs 12.. ...>											
	MOV (abs16),Dm	mem8(abs16)→Dm	--	--	--	--	7	4		0010 1100 00Dm	<abs 16..	...>										
	MOV Dn,(Am)	Dn→mem8(Am)	--	--	--	--	2	2		0101 1aDn												
	MOV Dn,(d8,Am)	Dn→mem8(d8+Am)	--	--	--	--	4	2		0111 1aDn	<d8. ...>											*1
	MOV Dn,(d16,Am)	Dn→mem8(d16+Am)	--	--	--	--	7	4		0010 0111 1aDn	<d16	....	...>									
	MOV Dn,(d4,SP)	Dn→mem8(d4+SP)	--	--	--	--	3	2		0111 01Dn	<d4>											*2
	MOV Dn,(d8,SP)	Dn→mem8(d8+SP)	--	--	--	--	5	3		0010 0111 01Dn	<d8. ...>											*3
	MOV Dn,(d16,SP)	Dn→mem8(d16+SP)	--	--	--	--	7	4		0010 0111 00Dn	<d16	....	...>									
	MOV Dn,(io8)	Dn→mem8(IOTOP+io8)	--	--	--	--	4	2		0111 00Dn	<io8 ...>											
	MOV Dn,(abs8)	Dn→mem8(abs8)	--	--	--	--	4	2		0101 01Dn	<abs 8..>											
	MOV Dn,(abs12)	Dn→mem8(abs12)	--	--	--	--	5	2		0101 00Dn	<abs 12.. ...>											
	MOV Dn,(abs16)	Dn→mem8(abs16)	--	--	--	--	7	4		0010 1101 00Dn	<abs 16..	...>										
	MOV imm8,(io8)	imm8→mem8(IOTOP+io8)	--	--	--	--	6	3		0000 0010	<io8 ...>	<#8. ...>										
	MOV imm8,(abs8)	imm8→mem8(abs8)	--	--	--	--	6	3		0001 0100	<abs 8..>	<#8. ...>										
	MOV imm8,(abs12)	imm8→mem8(abs12)	--	--	--	--	7	3		0001 0101	<abs 12.. ...>	<#8. ...>										
	MOV imm8,(abs16)	imm8→mem8(abs16)	--	--	--	--	9	5		0011 1101 1001	<abs 16..	...>	<#8. ...>									
	MOV Dn,(HA)	Dn→mem8(HA)	--	--	--	--	2	2		1101 00Dn												
MOVW	MOVW (An),DWm	mem16(An)→DWm	--	--	--	--	2	3		1110 00Ad												
	MOVW (An),Am	mem16(An)→Am	--	--	--	--	3	4		0010 1110 10Aa												*4
	MOVW (d4,SP),DWm	mem16(d4+SP)→DWm	--	--	--	--	3	3		1110 011d	<d4>											*2
	MOVW (d4,SP),Am	mem16(d4+SP)→Am	--	--	--	--	3	3		1110 010a	<d4>											*2
	MOVW (d8,SP),DWm	mem16(d8+SP)→DWm	--	--	--	--	5	4		0010 1110 011d	<d8. ...>											*3
	MOVW (d8,SP),Am	mem16(d8+SP)→Am	--	--	--	--	5	4		0010 1110 010a	<d8. ...>											*3
	MOVW (d16,SP),DWm	mem16(d16+SP)→DWm	--	--	--	--	7	5		0010 1110 001d	<d16	....	...>									
	MOVW (d16,SP),Am	mem16(d16+SP)→Am	--	--	--	--	7	5		0010 1110 000a	<d16	....	...>									
	MOVW (abs8),DWm	mem16(abs8)→DWm	--	--	--	--	4	3		1100 011d	<abs 8..>											
	MOVW (abs8),Am	mem16(abs8)→Am	--	--	--	--	4	3		1100 010a	<abs 8..>											
	MOVW (abs16),DWm	mem16(abs16)→DWm	--	--	--	--	7	5		0010 1100 011d	<abs 16..	...>										
	MOVW (abs16),Am	mem16(abs16)→Am	--	--	--	--	7	5		0010 1100 010a	<abs 16..	...>										
	MOVW DWn,(Am)	DWn→mem16(Am)	--	--	--	--	2	3		1111 00aD												
	MOVW An,(Am)	An→mem16(Am)	--	--	--	--	3	4		0010 1111 10aA												*4
	MOVW DWn,(d4,SP)	DWn→mem16(d4+SP)	--	--	--	--	3	3		1111 011D	<d4>											*2
	MOVW An,(d4,SP)	An→mem16(d4+SP)	--	--	--	--	3	3		1111 010A	<d4>											*2
	MOVW DWn,(d8,SP)	DWn→mem16(d8+SP)	--	--	--	--	5	4		0010 1111 011D	<d8. ...>											*3
	MOVW An,(d8,SP)	An→mem16(d8+SP)	--	--	--	--	5	4		0010 1111 010A	<d8. ...>											*3
	MOVW DWn,(d16,SP)	DWn→mem16(d16+SP)	--	--	--	--	7	5		0010 1111 001D	<d16	....	...>									
	MOVW An,(d16,SP)	An→mem16(d16+SP)	--	--	--	--	7	5		0010 1111 000A	<d16	....	...>									
	MOVW DWn,(abs8)	DWn→mem16(abs8)	--	--	--	--	4	3		1101 011D	<abs 8..>											
	MOVW An,(abs8)	An→mem16(abs8)	--	--	--	--	4	3		1101 010A	<abs 8..>											
	MOVW DWn,(abs16)	DWn→mem16(abs16)	--	--	--	--	7	5		0010 1101 011D	<abs 16..	...>										
	MOVW An,(abs16)	An→mem16(abs16)	--	--	--	--	7	5		0010 1101 010A	<abs 16..	...>										
	MOVW DWn,(HA)	DWn→mem16(HA)	--	--	--	--	2	3		1001 010D												
	MOVW An,(HA)	An→mem16(HA)	--	--	--	--	2	3		1001 011A												
	MOVW imm8,DWm	sign(imm8)→DWm	--	--	--	--	4	2		0000 110d	<#8. ...>											*5
	MOVW imm8,Am	zero(imm8)→Am	--	--	--	--	4	2		0000 111a	<#8. ...>											*6
	MOVW imm16,DWm	imm16→DWm	--	--	--	--	6	3		1100 111d	<#16	....	...>									

*1 d8 sign-extension
*2 d4 zero-extension
*3 d8 zero-extension
*4 A=An, a=Am
*5 #8 sign-extension
*6 #8 zero-extension

MN101C SERIES INSTRUCTION SET

Group	Mnemonic	Operation	Flag VF	Flag NF	Flag CF	Flag ZF	Code Size	Cycle	Re-peat	extension	Machine Code 1	2	3	4	5	6	7	8	9	10	11	Notes
	MOVW imm16,Am	imm16→Am	--	--	--	--	6	3			1101	111a	<#16	....	....	...>						
	MOVW SP,Am	SP→Am	--	--	--	--	3	3			0010	0000	100a									
	MOVW An,SP	An→SP	--	--	--	--	3	3			0010	0000	101A									
	MOVW DWn,DWm	DWn→DWm	--	--	--	--	3	3			0010	1000	00Dd									*1
	MOVW DWn,Am	DWn→Am	--	--	--	--	3	3			0010	0100	11Da									
	MOVW An,DWm	An→DWm	--	--	--	--	3	3			0010	1100	11Ad									
	MOVW An,Am	An→Am	--	--	--	--	3	3			0010	0000	00Aa									*2
PUSH	PUSH Dn	SP-1→SP,Dn→mem8(SP)	--	--	--	--	2	3			1111	10Dn										
	PUSH An	SP-2→SP,An→mem16(SP)	--	--	--	--	2	5			0001	011A										
POP	POP Dn	mem8(SP)→Dn,SP+1→SP	--	--	--	--	2	3			1110	10Dn										
	POP An	mem16(SP)→An,SP+2→SP	--	--	--	--	2	4			0000	011A										
EXT	EXT Dn,DWm	sign(Dn)→DWm	--	--	--	--	3	3			0010	1001	000d									*3

Arithmetic manupulation instructions

ADD	ADD Dn,Dm	Dm+Dn→Dm	●	●	●	●	3	2			0011	0011	DnDm									
	ADD imm4,Dm	Dm+sign(imm4)→Dm	●	●	●	●	3	2			1000	00Dm	<#4>									*6
	ADD imm8,Dm	Dm+imm8→Dm	●	●	●	●	4	2			0000	10Dm	<#8.	...>								
ADDC	ADDC Dn,Dm	Dm+Dn+CF→Dm	●	●	●	●	3	2	○		0011	1011	DnDm									
ADDW	ADDW DWn,DWm	DWm+DWn→DWm	●	●	●	●	3	3	○		0010	0101	00Dd									*1
	ADDW DWn,Am	Am+DWn→Am	●	●	●	●	3	3	○		0010	0101	10Da									
	ADDW imm4,Am	Am+sign(imm4)→Am	●	●	●	●	3	2			1110	110a	<#4>									*6
	ADDW imm8,Am	Am+sign(imm8)→Am	●	●	●	●	5	3			0010	1110	110a	<#8.	...>							*7
	ADDW imm16,Am	Am+imm16→Am	●	●	●	●	7	4			0010	0101	011a	<#16	....	....	...>					
	ADDW imm4,SP	SP+sign(imm4)→SP	--	--	--	--	3	2			1111	1101	<#4>									*6
	ADDW imm8,SP	SP+sign(imm8)→SP	--	--	--	--	4	2			1111	1100	<#8.	...>								*7
	ADDW imm16,SP	SP+imm16→SP	--	--	--	--	7	4			0010	1111	1100	<#16	....	....	...>					
	ADDW imm16,DWm	DWm+imm16→DWm	●	●	●	●	7	4			0010	0101	010d	<#16	....	....	...>					
ADDUW	ADDUW Dn,Am	Am+zero(Dn)→Am	●	●	●	●	3	3	○		0010	1000	1aDn									*8
ADDSW	ADDSW Dn,Am	Am+sign(Dn)→Am	●	●	●	●	3	3	○		0010	1001	1aDn									
SUB	SUB Dn,Dm(when Dn≠Dm)	Dm-Dn→Dm	●	●	●	●	3	2	○		0010	1010	DnDm									
	SUB Dn,Dn	Dn-Dn→Dn	0	0	0	1	2	1			1000	01Dn										
	SUB imm8,Dm	Dm-imm8→Dm	●	●	●	●	5	3			0010	1010	DmDm	<#8.	...>							
SUBC	SUBC Dn,Dm	Dm-Dn-CF→Dm	●	●	●	●	3	2	○		0010	1011	DnDm									
SUBW	SUBW DWn,DWm	DWm-DWn→DWm	●	●	●	●	3	3			0010	0100	00Dd									*1
	SUBW DWn,Am	Am-DWn→Am	●	●	●	●	3	3			0010	0100	10Da									
	SUBW imm16,DWm	DWm-imm16→DWm	●	●	●	●	7	4			0010	0100	010d	<#16	....	....	...>					
	SUBW imm16,Am	Am-imm16→Am	●	●	●	●	7	4			0010	0100	011a	<#16	....	....	...>					
MULU	MULU Dn,Dm	Dm*Dn→DWk	0	●	●	●	3	8			0010	1111	111D									*4
DIVU	DIVU Dn,DWm	DWm/Dn→DWm-l...DWm-h	●	●	●	●	3	9			0010	1110	111d									*5
CMP	CMP Dn,Dm	Dm-Dn...PSW	●	●	●	●	3	2			0011	0010	DnDm									
	CMP imm8,Dm	Dm-imm8...PSW	●	●	●	●	4	2			1100	00Dm	<#8.	...>								
	CMP imm8,(abs8)	mem8(abs8)-imm8...PSW	●	●	●	●	6	3			0000	0100	<abs	8..>	<#8.	...>						
	CMP imm8,(abs12)	mem8(abs12)-imm8...PSW	●	●	●	●	7	3			0000	0101	<abs	12..	...>	<#8.	...>					
	CMP imm8,(abs16)	mem8(abs16)-imm8...PSW	●	●	●	●	9	5			0011	1101	1000	<abs	16..	....	...>	<#8.	...>			
CMPW	CMPW DWn,DWm	DWm-DWn...PSW	●	●	●	●	3	3			0010	1000	01Dd									*1
	CMPW DWn,Am	Am-DWn...PSW	●	●	●	●	3	3			0010	0101	11Da									
	CMPW An,Am	Am-An...PSW	●	●	●	●	3	3			0010	0000	01Aa									*2
	CMPW imm16,DWm	DWm-imm16...PSW	●	●	●	●	6	3			1100	110d	<#16	....	....	...>						
	CMPW imm16,Am	Am-imm16...PSW	●	●	●	●	6	3			1101	110a	<#16	....	....	...>						

Logical manipulation instructions

AND	AND Dn,Dm	Dm&Dn→Dm	0	●	0	●	3	2			0011	0111	DnDm									
	AND imm8,Dm	Dm&imm8→Dm	0	●	0	●	4	2			0001	11Dm	<#8.	...>								
	AND imm8,PSW	PSW&imm8→PSW	●	●	●	●	5	3			0010	1001	0010	<#8.	...>							
OR	OR Dn,Dm	DmIDn→Dm	0	●	0	●	3	2			0011	0110	DnDm									
	OR imm8,Dm	DmIimm8→Dm	0	●	0	●	4	2			0001	10Dm	<#8.	...>								
	OR imm8,PSW	PSWIimm8→PSW	●	●	●	●	5	3			0010	1001	0011	<#8.	...>							
XOR	XOR Dn,Dm	Dm^Dn→Dm	0	●	0	●	3	2			0011	1010	DnDm									*9
	XOR imm8,Dm	Dm^imm8→Dm	0	●	0	●	5	3			0011	1010	DmDm	<#8.	...>							

*1 D=DWn, d=DWm
*2 A=An, a=Am
*3 d=DWm
*4 D=DWk
*5 D=DWm
*6 #4 sign-extension
*7 #8 sign-extension
*8 Dn zero extension
*9 m≠n

MN101C SERIES INSTRUCTION SET

Group	Mnemonic	Operation	Flag VF	Flag NF	Flag CF	Flag ZF	Code Size	Cycle	Re-peat	Extension	Machine Code 1	2	3	4	5	6	7	8	9	10	11	Notes
NOT	NOT Dn	¯Dn→Dn=	0	●	0	●	3	2			0010	0010	10Dn									
ASR	ASR Dn	Dn.msb→temp,Dn.lsb→CF Dn>>1→Dn,temp→Dn.msb	0	--	●	●	3	2	○		0010	0011	10Dn									
LSR	LSR Dn	Dn.lsb→CF,Dn>>1→Dn 0→Dn.msb	0	0	●	●	3	2	○		0010	0011	11Dn									
ROR	ROR Dn	Dn.Isb→temp,Dn>>1→Dn CF→Dn.msb,temp→CF	0	●	●	●	3	2	○		0010	0010	11Dn									
Bit manipulation instructions																						
BSET	BSET (io8)bp	mem8(IOTOP+io8)&bpdata...PSW 1→mem8(IOTOP+io8)bp	0	●	0	●	5	5			0011	1000	0bp.	<io8	...>							
	BSET (abs8)bp	mem8(abs8)&bpdata...PSW 1→mem8(abs8)bp	0	●	0	●	4	4				1011	0bp.	<abs	8..>							
	BSET (abs16)bp	mem8(abs16)&bpdata...PSW 1→mem8(abs16)bp	0	●	0	●	7	6			0011	1100	0bp.	<abs	16..	....	...>					
BCLR	BCLR (io8)bp	mem8(IOTOP+io8)&bpdata...PSW 0→mem8(IOTOP+io8)bp	0	●	0	●	5	5			0011	1000	1bp.	<io8	...>							
	BCLR (abs8)bp	mem8(abs8)&bpdata...PSW 0→mem8(abs8)bp	0	●	0	●	4	4				1011	1bp.	<abs	8..>							
	BCLR (abs16)bp	mem8(abs16)&bpdata...PSW 0→mem8(abs16)bp	0	●	0	●	7	6			0011	1100	1bp.	<abs	16..	....	...>					
BTST	BTST imm8,Dm	Dm&imm8...PSW	0	●	0	●	5	3			0010	0000	11Dm	<#8.	...>							
	BTST (abs16)bp	mem8(abs16)&bpdata...PSW	0	●	0	●	7	5			0011	1101	0bp.	<abs	16..	....	...>					
Branch instructions																						
Bcc	BEQ label	if(ZF=1), PC+3+d4(label)+H→PC if(ZF=0), PC+3→PC	--	--	--	--	3	2/3				1001	000H	<d4>								*1
	BEQ label	if(ZF=1), PC+4+d7(label)+H→PC if(ZF=0), PC+4→PC	--	--	--	--	4	2/3				1000	1010	<d7.	...H							*2
	BEQ label	if(ZF=1), PC+5+d11(label)+H→PC if(ZF=0), PC+5→PC	--	--	--	--	5	2/3				1001	1010	<d11	....	...H						*3
	BNE label	if(ZF=0), PC+3+d4(label)+H→PC if(ZF=1), PC+3→PC	--	--	--	--	3	2/3				1001	001H	<d4>								1
	BNE label	if(ZF=0), PC+4+d7(label)+H→PC if(ZF=1), PC+4→PC	--	--	--	--	4	2/3				1000	1011	<d7.	...H							*2
	BNE label	if(ZF=0), PC+5+d11(label)+H→PC if(ZF=1), PC+5→PC	--	--	--	--	5	2/3				1001	1011	<d11	....	...H						*3
	BGE label	if((VF^NF)=0),PC+4+d7(label)+H→PC if((VF^NF)=1),PC+4→PC	--	--	--	--	4	2/3				1000	1000	<d7.	...H							*2
	BGE label	if((VF^NF)=0),PC+5+d11(label)+H→PC if((VF^NF)=1),PC+5→PC	--	--	--	--	5	2/3				1001	1000	<d11	....	...H						*3
	BCC label	if(CF=0),PC+4+d7(label)+H→PC if(CF=1), PC+4→PC	--	--	--	--	4	2/3				1000	1100	<d7.	...H							*2
	BCC label	if(CF=0), PC+5+d11(label)+H→PC if(CF=1), PC+5→PC	--	--	--	--	5	2/3				1001	1100	<d11	....	...H						*3
	BCS label	if(CF=1),PC+4+d7(label)+H→PC if(CF=0), PC+4→PC	--	--	--	--	4	2/3				1000	1101	<d7.	...H							*2
	BCS label	if(CF=1), PC+5+d11(label)+H→PC if(CF=0), PC+5→PC	--	--	--	--	5	2/3				1001	1101	<d11	....	...H						*3
	BLT label	if((VF^NF)=1),PC+4+d7(label)+H→PC if((VF^NF)=0),PC+4→PC	--	--	--	--	4	2/3				1000	1110	<d7.	...H							*2
	BLT label	if((VF^NF)=1),PC+5+d11(label)+H→PC if((VF^NF)=0),PC+5→PC	--	--	--	--	5	2/3				1001	1110	<d11	....	...H						*3
	BLE label	if((VF^NF)|ZF=1),PC+4+d7(label)+H→PC if((VF^NF)|ZF=0),PC+4→PC	--	--	--	--	4	2/3				1000	1111	<d7.	...H							*2
	BLE label	if((VF^NF)|ZF=1),PC+5+d11(label)+H→PC if((VF^NF)|ZF=0),PC+5→PC	--	--	--	--	5	2/3				1001	1111	<d11	....	...H						*3
	BGT label	if((VF^NF)|ZF=0),PC+5+d7(label)+H→PC if((VF^NF)|ZF=1),PC+5→PC	--	--	--	--	5	3/4			0010	0010	0001	<d7.	...H							*2

*1 d4 sign-extension
*2 d7 sign-extension
*3 d11 sign-extension

MN101C SERIES INSTRUCTION SET

Group	Mnemonic	Operation	Flag				Code Size	Cycle	Re-peat	Exten-sion	Machine Code 1	2	3	4	5	6	7	8	9	10	11	Notes
			VF	NF	CF	ZF																
Bcc	BGT label	if((VF^NF)|ZF=0),PC+6+d11(label)+H→PC if((VF^NF)|ZF=1),PC+6→PC	--	--	--	--	6	3/4			0010	0011	0001	<d11	....	...H						*3
	BHI label	if(CFIZF=0),PC+5+d7(label)+H→PC if(CFIZF=1), PC+5→PC	--	--	--	--	5	3/4			0010	0010	0010	<d7.	...H							*2
	BHI label	if(CFIZF=0),PC+6+d11(label)+H→PC if(CFIZF=1), PC+6→PC	--	--	--	--	6	3/4			0010	0011	0010	<d11	....	...H						*3
	BLS label	if(CFIZF=1),PC+5+d7(label)+H→PC if(CFIZF=0), PC+5→PC	--	--	--	--	5	3/4			0010	0010	0011	<d7.	...H							*2
	BLS label	if(CFIZF=1),PC+6+d11(label)+H→PC if(CFIZF=0), PC+6→PC	--	--	--	--	6	3/4			0010	0011	0011	<d11	....	...H						*3
	BNC label	if(NF=0),PC+5+d7(label)+H→PC if(NF=1),PC+5→PC	--	--	--	--	5	3/4			0010	0010	0100	<d7.	...H							*2
	BNC label	if(NF=0),PC+6+d11(label)+H→PC if(NF=1),PC+6→PC	--	--	--	--	6	3/4			0010	0011	0100	<d11	....	...H						*3
	BNS label	if(NF=1),PC+5+d7(label)+H→PC if(NF=0),PC+5→PC	--	--	--	--	5	3/4			0010	0010	0101	<d7.	...H							*2
	BNS label	if(NF=1),PC+6+d11(label)+H→PC if(NF=0),PC+6→PC	--	--	--	--	6	3/4			0010	0011	0101	<d11	....	...H						*3
	BVC label	if(VF=0),PC+5+d7(label)+H→PC if(VF=1),PC+5→PC	--	--	--	--	5	3/4			0010	0010	0110	<d7.	...H							*2
	BVC label	if(VF=0),PC+6+d11(label)+H→PC if(VF=1),PC+6→PC	--	--	--	--	6	3/4			0010	0011	0110	<d11	....	...H						*3
	BVS label	if(VF=1),PC+5+d7(label)+H→PC if(VF=0),PC+5→PC	--	--	--	--	5	3/4			0010	0010	0111	<d7.	...H							*2
	BVS label	if(VF=1),PC+6+d11(label)+H→PC if(VF=0),PC+6→PC	--	--	--	--	6	3/4			0010	0011	0111	<d11	....	...H						*3
	BRA label	PC+3+d4(label)+H→PC	--	--	--	--	3	3			1110	111H	<d4>									*1
	BRA label	PC+4+d7(label)+H→PC	--	--	--	--	4	3			1000	1001	<d7.	...H								*2
	BRA label	PC+5+d11(label)+H→PC	--	--	--	--	5	3			1001	1001	<d11	....	...H							*3
CBEQ	CBEQ imm8,Dm,label	if(Dm=imm8),PC+6+d7(label)+H→PC if(Dm≠imm8),PC+6→PC	●	●	●	●	6	3/4			1100	10Dm	<#8.	...>	<d7.	...H						*2
	CBEQ imm8,Dm,label	if(Dm=imm8),PC+8+d11(label)+H→PC if(Dm≠imm8),PC+8→PC	●	●	●	●	8	4/5			0010	1100	10Dm	<#8.	...>	<d11	....	...H				*3
	CBEQ imm8,(abs8),label	if(mem8(abs8)=imm8),PC+9+d7(label)+H→PC if(mem8(abs8)≠imm8),PC+9→PC	●	●	●	●	9	6/7			0010	1101	1100	<abs	8..>	<#8.	...>	<d7.	...H			*2
	CBEQ imm8,(abs8),label	if(mem8(abs8)=imm8),PC+10+d11(label)+H→PC if(mem8(abs8)≠imm8),PC+10→PC	●	●	●	●	10	6/7			0010	1101	1101	<abs	8..>	<#8.	...>	<d11	....	...H		*3
	CBEQ imm8,(abs16),label	if(mem8(abs16)=imm8),PC+11+d7(label)+H→PC if(mem8(abs16)≠imm8),PC+11→PC	●	●	●	●	11	7/8			0011	1101	1100	<abs	16..	....	...>	<#8.	...>	<d7.	...H	*2
	CBEQ imm8,(abs16),label	if(mem8(abs16)=imm8),PC+12+d11(label)+H→PC if(mem8(abs16)≠imm8),PC+12→PC	●	●	●	●	12	7/8			0011	1101	1101	<abs	16..	....	...>	<#8.	...>	<d11	....	...H	*3
CBNE	CBNE imm8,Dm,label	if(Dm≠imm8),PC+6+d7(label)+H→PC if(Dm=imm8),PC+6→PC	●	●	●	●	6	3/4			1101	10Dm	<#8.	...>	<d7.	..H>						*2
	CBNE imm8,Dm,label	if(Dm≠imm8),PC+8+d11(label)+H→PC if(Dm=imm8),PC+8→PC	●	●	●	●	8	4/5			0010	1101	10Dm	<#8.	...>	<d11	....	...H				*3
	CBNE imm8,(abs8),label	if(mem8(abs8)≠imm8),PC+9+d7(label)+H→PC if(mem8(abs8)=imm8),PC+9→PC	●	●	●	●	9	6/7			0010	1101	1110	<abs	8..>	<#8.	...>	<d7.	...H			*2
	CBNE imm8,(abs8),label	if(mem8(abs8)≠imm8),PC+10+d11(label)+H→PC if(mem8(abs8)=imm8),PC+10→PC	●	●	●	●	10	6/7			0010	1101	1111	<abs	8..>	<#8.	...>	<d11	....	...H		*3
	CBNE imm8,(abs16),label	if(mem8(abs16)≠imm8),PC+11+d7(label)+H→PC if(mem8(abs16)=imm8),PC+11→PC	●	●	●	●	11	7/8			0011	1101	1110	<abs	16..	....	...>	<#8.	...>	<d7.	...H	*2
	CBNE imm8,(abs16),label	if(mem8(abs16)≠imm8),PC+12+d11(label)+H→PC if(mem8(abs16)=imm8),PC+12→PC	●	●	●	●	12	7/8			0011	1101	1111	<abs	16..	....	...>	<#8.	...>	<d11	....	...H	*3
TBZ	TBZ (abs8)bp,label	if(mem8(abs8)bp=0),PC+7+d7(label)+H→PC if(mem8(abs8)bp=1),PC+7→PC	0	●	0	●	7	6/7			0011	0000	0bp.	<abs	8..>	<d7.	...H					*2
	TBZ (abs8)bp,label	if(mem8(abs8)bp=0),PC+8+d11(label)+H→PC if(mem8(abs8)bp=1),PC+8→PC	0	●	0	●	8	6/7			0011	0000	1bp.	<abs	8..>	<d11	....	...H				*3

*1 d4 sign-extension
*2 d7 sign-extension
*3 d11 sign-extension

MN101C SERIES INSTRUCTION SET

Group	Mnemonic	Operation	Flag VF	Flag NF	Flag CF	Flag ZF	Code Size	Cycle	Re-peat	Exten-sion	Machine Code 1	2	3	4	5	6	7	8	9	10	11	Notes
TBZ	TBZ (io8)bp,label	if(mem8(IOTOP+io8)bp=0),PC+7+d7(label)+H→PC if(mem8(IOTOP+io8)bp=1),PC+7→PC	0	●	0	●	7	6/7			0011	0100	0bp.	<io8 ...>	<d7. ...H							*1
	TBZ (io8)bp,label	if(mem8(IOTOP+io8)bp=0),PC+8+d11(label)+H→PC if(mem8(IOTOP+io8)bp=1),PC+8→PC	0	●	0	●	8	6/7			0011	0100	1bp.	<io8 ...>	<d11	...H						*2
	TBZ (abs16)bp,label	if(mem8(abs16)bp=0),PC+9+d7(label)+H→PC if(mem8(abs16)bp=1),PC+9→PC	0	●	0	●	9	7/8			0011	1110	0bp.	<abs 16..	...>	<d7. ...H						*1
	TBZ (abs16)bp,label	if(mem8(abs16)bp=0),PC+10+d11(label)+H→PC if(mem8(abs16)bp=1),PC+10→PC	0	●	0	●	10	7/8			0011	1110	1bp.	<abs 16..	...>	<d11	...H					*2
TBNZ	TBNZ (abs8)bp,label	if(mem8(abs8)bp=1),PC+7+d7(label)+H→PC if(mem8(abs8)bp=0),PC+7→PC	0	●	0	●	7	6/7			0011	0001	0bp.	<abs 8..>	<d7. ...H							*1
	TBNZ (abs8)bp,label	if(mem8(abs8)bp=1),PC+8+d11(label)+H→PC if(mem8(abs8)bp=0),PC+8→PC	0	●	0	●	8	6/7			0011	0001	1bp.	<abs 8..>	<d11	...H						*2
	TBNZ (io8)bp,label	if(mem8(io)bp=1),PC+7+d7(label)+H→PC if(mem8(io)bp=0),PC+7→PC	0	●	0	●	7	6/7			0011	0101	0bp.	<io8 ...>	<d7. ...H							*1
	TBNZ (io8)bp,label	if(mem8(io)bp=1),PC+8+d11(label)+H→PC if(mem8(io)bp=0),PC+8→PC	0	●	0	●	8	6/7			0011	0101	1bp.	<io8 ...>	<d11	...H						*2
	TBNZ (abs16)bp,label	if(mem8(abs16)bp=1),PC+9+d7(label)+H→PC if(mem8(abs16)bp=0),PC+9→PC	0	●	0	●	9	7/8			0011	1111	0bp.	<abs 16..	...>	<d7. ...H						*1
	TBNZ (abs16)bp,label	if(mem8(abs16)bp=1),PC+10+d11(label)+H→PC if(mem8(abs16)bp=0),PC+10→PC	0	●	0	●	10	7/8			0011	1111	1bp.	<abs 16..	...>	<d11	...H					*2
JMP	JMP (An)	0→PC.17-16,An→PC.15-0,0→PC.H	---	---	---	---	3	4			0010	0001	00A0									
	JMP label	abs18(label)+H→PC	---	---	---	---	7	5			0011	1001	0aaH	<abs 18.b	p15~ 0..>							*5
JSR	JSR (An)	SP-3→SP,(PC+3).bp7-0→mem8(SP) (PC+3).bp15-8→mem8(SP+1) (PC+3).H→mem8(SP+2).bp7, 0→mem8(SP+2).bp6-2, (PC+3).bp17-16→mem8(SP+2).bp1-0 0→PC.bp17-16 An→PC.bp15-0,0→PC.H	---	---	---	---	3	7			0010	0001	00A1									
	JSR label	SP-3→SP,(PC+5).bp7-0→mem8(SP) (PC+5).bp15-8→mem8(SP+1) (PC+5).H→mem8(SP+2).bp7, 0→mem8(SP+2).bp6-2, (PC+5).bp17-16→mem8(SP+2).bp1-0 PC+5+d12(label)+H→PC	---	---	---	---	5	6				0001	000H	<d12	...>							*3
	JSR label	SP-3→SP,(PC+6).bp7-0→mem8(SP) (PC+6).bp15-8→mem8(SP+1) (PC+6).H→mem8(SP+2).bp7, 0→mem8(SP+2).bp6-2, (PC+6).bp17-16→mem8(SP+2).bp1-0 PC+6+d16(label)+H→PC	---	---	---	---	6	7				0001	001H	<d16	....	...>						*4
	JSR label	SP-3→SP,(PC+7).bp7-0→mem8(SP) (PC+7).bp15-8→mem8(SP+1) (PC+7).H→mem8(SP+2).bp7, 0→mem8(SP+2).bp6-2, (PC+7).bp17-16→mem8(SP+2).bp1-0 abs18(label)+H→PC	---	---	---	---	7	8			0011	1001	1aaH	<abs 18.b	p15~ 0..>							*5
	JSRV (tbl4)	SP-3→SP,(PC+3).bp7-0→mem8(SP) (PC+3).bp15-8→mem8(SP+1) (PC+3).H→mem8(SP+2).bp7 0→mem8(SP+2).bp6-2, (PC+3).bp17-16→mem8(SP+2).bp1-0 mem8(x'004080+tbl4<<2)→PC.bp7-0 mem8(x'004080+tbl4<<2+1)→PC.bp15-8 mem8(x'004080+tbl4<<2+2).bp7→PC.H mem8(x'004080+tbl4<<2+2).bp1-0→ PC.bp17-16	---	---	---	---	3	9				1111	1110	<t4>								
NOP	NOP	PC+2→PC	---	---	---	---	2	1	○			0000	0000									

*1 d7 sign-extension
*2 d11 sign-extension
*3 d12 sign-extension
*4 d16 sign-extension
*5 aa=abs18.17 - 16

MN101C SERIES INSTRUCTION SET

Group	Mnemonic	Operation	Flag VF	Flag NF	Flag CF	Flag ZF	Code Size	Cycle	Re-peat	Exten-sion	Machine Code 1	2	3	4	5	6	7	8	9	10	11	Notes
RTS	RTS	mem8(SP)→(PC).bp7-0 mem8(SP+1)→(PC).bp15-8 mem8(SP+2).bp7→(PC).H mem8(SP+2).bp1-0→(PC).bp17-16 SP+3→SP	---	---	---	---	2	7			0000 0001											
RTI	RTI	mem8(SP)→PSW mem8(SP+1)→(PC).bp7-0 mem8(SP+2)→(PC).bp15-8 mem8(SP+3).bp7→(PC).H mem8(SP+3).bp1-0→(PC).bp17-16 mem8(SP+4)→HA-l mem8(SP+5)→HA-h SP+6→SP	●	●	●	●	2	11			0000 0011											
Contorl instructions																						
REP	REP imm3	imm3-1→RPC	---	---	---	---	3	2			0010 0001 1rep											*1

*1 no repeat whn imm3=0, (rep: imm3-1)

Other than the instruction of MN101C Series,the assembler of this Series has the following instructions as macro instructions.
The assembler will interpret the macro instructions below as the assembler instructions.

macro instructions		replaced instructions		remarks
INC	Dn	ADD	1,Dn	
DEC	Dn	ADD	-1,Dn	
INC	An	ADDW	1,An	
DEC	An	ADDW	-1,An	
INC2	An	ADDW	2,An	
DEC2	An	ADDW	-2,An	
CLR	Dn	SUB	Dn,Dm	n=m
ASL	Dn	ADD	Dn,Dm	n=m
LSL	Dn	ADD	Dn,Dm	n=m
ROL	Dn	ADDC	Dn,Dm	n=m
NEG	Dn	NOT ADD	Dn 1,Dn	
NOPL		MOVW	DWn,DWm	n=m
MOV	(SP),Dn	MOV	(0,SP),Dn	
MOV	Dn,(SP)	MOV	Dn,(0,SP)	
MOVW	(SP),DWn	MOVW	(0,SP),DWn	
MOVW	DWn,(SP)	MOVW	DWn,(0,SP)	
MOVW	(SP),An	MOVW	(0,SP),An	
MOVW	An,(SP)	MOVW	An,(0,SP)	

13-5 Instruction Map

MN101C SERIES INSTRUCTION MAP

1st nibble\2nd nibble

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	RTS	MOV #8,(io8)	RTI	CMP #8,(abs8)/(abs12)		POP An		ADD #8,Dm				MOVW #8,DWm		MOVW #8,Am	
1	JSR d12(label)		JSR d16(label)		MOV #8,(abs8)/(abs12)		PUSH An		OR #8,Dm				AND #8,Dm			
2	When the exension code is b'oo10'															
3	When the extension code is b'0011'															
4	MOV (abs12),Dm				MOV (abs8),Dm				MOV (An),Dm							
5	MOV Dn,(abs12)				MOV Dn,(abs8)				MOV Dn,(Am)							
6	MOV (io8),Dm				MOV (d4,SP),Dm				MOV (d8,An),Dm							
7	MOV Dn,(io8)				MOV Dn,(d4,SP)				MOV Dn,(d8,Am)							
8	ADD #4,Dm				SUB Dn,Dn				BGE d7	BRA d7	BEQ d7	BNE d7	BCC d7	BCS d7	BLT d7	BLE d7
9	BEQ d4		BNE d4		MOVW DWn,(HA)		MOVW An,(HA)		BGE d11	BRA d11	BEQ d11	BNE d11	BCC d11	BCS d11	BLT d11	BLE d11
A	MOV Dn,Dm / MOV #8,Dm															
B	BSET (abs8)bp								BCLR (abs8)bp							
C	CMP #8,Dm				MOVW (abs8),Am		MOVW (abs8),DWm		CBEQ #8,Dm,d7				CMPW #16,DWm		MOVW #16,DWm	
D	MOV Dn,(HA)				MOVW An,(abs8)		MOVW DWn,(abs8)		CBNE #8,Dm,d7				CMPW #16,Am		MOVW #16,Am	
E	MOVW (An),DWm				MOVW (d4,SP),Am		MOVW (d4,SP),DWm		POP Dn				ADDW #4,Am		BRA d4	
F	MOVW DWn,(Am)				MOVW An,(d4,SP)		MOVW DWn,(d4,SP)		PUSH Dn				ADDW #8,SP	ADDW #4,SP	JSRV (tbl4)	

Extension code: b'0010'

2nd nible\3rd nibble

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	MOVW An,Am				CMPW An,Am				MOVW SP,Am		MOVW An,SP		BTST #8,Dm			
1	JMP (A0)	JSR (A0)	JMP (A1)	JSR (A1)	MOV PSW,Dm				REP #3							
2		BGT d7	BHI d7	BLS d7	BNC d7	BNS d7	BVC d7	BVS d7	NOT Dn				ROR Dn			
3		BGT d11	BHI d11	BLS d11	BNC d11	BNS d11	BVC d11	BVS d11	ASR Dn				LSR Dn			
4	SUBW DWn,DWm				SUBW #16,DWm		SUBW #16,Am		SUBW DWn,Am				MOVW DWn,Am			
5	ADDW DWn,DWm				ADDW #16,DWm		ADDW #16,Am		ADDW DWn,Am				CMPW DWn,Am			
6	MOV (d16,SP),Dm				MOV (d8,SP),Dm				MOV (d16,An),Dm							
7	MOV Dn,(d16,SP)				MOV Dn,(d8,SP)				MOV Dn,(d16,Am)							
8	MOVW DWn,DWm (NOPL @n=m)				CMPW DWn,DWm				ADDUW Dn,Am							
9	EXT Dn,DWm		AND #8,PSW	OR #8,PSW	MOV Dn,PSW				ADDSW Dn,Am							
A	SUB Dn,Dm / SUB #8,Dm															
B	SUBC Dn,Dm															
C	MOV (abs16),Dm				MOVW (abs16),Am		MOVW (abs16),DWm		CBEQ #8,Dm,d12				MOVW An,DWm			
D	MOV Dn,(abs16)				MOVW An,(abs16)		MOVW DWn,(abs16)		CBNE #8,Dm,d12				CBEQ #8,(abs8),d7/d11		CBNE #8,(abs8),d7/d11	
E	MOVW (d16,SP),Am		MOVW (d16,SP),DWm		MOVW (d8,SP),Am		MOVW (d8,SP),DWm		MOVW (An),Am				ADDW #8,Am		DIVU	
F	MOVW An,(d16,SP)		MOVW DWn,(d16,SP)		MOVW An,(d8,SP)		MOVW DWn,(d8,SP)		MOVW An,(Am)				ADDW #16,SP		MULU	

Extension code: b'0011'

2nd nibble\ 3rd nibble	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	TBZ (abs8)bp,d7								TBZ (abs8)bp,d11							
1	TBNZ (abs8)bp,d7								TBNZ (abs8)bp,d11							
2	CMP Dn,Dm															
3	ADD Dn,Dm															
4	TBZ (io8)bp,d7								TBZ (io8)bp,d11							
5	TBNZ (io8)bp,d7								TBNZ (io8)bp,d11							
6	OR Dn,Dm															
7	AND Dn,Dm															
8	BSET (io8)bp								BCLR (io8)bp							
9	JMP abs18(label)								JSR abs18(label)							
A	XOR Dn,Dm / XOR #8,Dm															
B	ADDC Dn,Dm															
C	BSET (abs16)bp								BCLR (abs16)bp							
D	BTST (abs16)bp								cmp #8,(abs16)	mov #8,(abs16)			CBEQ #8,(abs16),d7/11		CBNE #8,(abs16),d7/11	
E	TBZ (abs16)bp,d7								TBZ (abs16)bp,d11							
F	TBNZ (abs16)bp,d7								TBNZ (abs16)bp,d11							

Ver2.1(2001.03.26)

MN101C 539
LSI User's Manual

July, 2002 1st Edition 2nd Printing

Issued by Matsushita Electric Industrial Co., Ltd.

Semiconductor Company, Matsushita Electric Industrial Co., Ltd.

Nagaokakyo, Kyoto 617-8520, Japan
Tel: (075) 951-8151
http://www.panasonic.co.jp/semicon/

SALES OFFICES

■ NORTH AMERICA

●U.S.A. Sales Office:

Panasonic Industrial Company [PIC]

• New Jersey Office:
Two Panasonic Way Secaucus, New Jersey 07094 U.S.A.
Tel: 1-201-348-5257 Fax:1-201-392-4652

• Chicago Office:
1707 N. Randall Road Elgin, Illinois 60123-7847 U.S.A.
Tel: 1-847-468-5720 Fax:1-847-468-5725

• Milpitas Office:
1600 McCandless Drive Milpitas, California 95035 U.S.A.
Tel: 1-408-942-2912 Fax:1-408-946-9063

• Atlanta Office:
1225 Northbrook Parkway Suite 1-151 Suwanee, GA 30024 U.S.A.
Tel: 1-770-338-6953 Fax:1-770-338-6849

• San Diego Office:
9444 Balboa Avenue, Suite 185, San Diego, California 92123 U.S.A.
Tel: 1-619-503-2903 Fax:1-858-715-5545

●Canada Sales Office:

Panasonic Canada Inc. [PCI]
5770 Ambler Drive 27 Mississauga, Ontario, L4W 2T3 CANADA
Tel: 1-905-238-2315 Fax:1-905-238-2414

■ LATIN AMERICA

●Mexico Sales Office:

Panasonic de Mexico, S.A. de C.V. [PANAMEX]
Amores 1120 Col. Del Valle Delegacion Benito Juarez C.P. 03100 Mexico, D.F. MEXICO
Tel: 52-5-488-1000 Fax:52-5-488-1073

• Guadalajara Office:
SUCURSAL GUADALAJARA
Av. Lazaro Cardenas 2305 Local G-102 Plaza Comercial Abastos; Col. Las Torres Guadalajara, Jal. 44920 MEXICO
Tel: 52-3-671-1205 Fax:52-3-671-1256

●Brazil Sales Office:

Panasonic do Brasil Ltda. [PANABRAS]
Caixa Postal 1641, Sao Jose dos Campos, Estado de Sao Paulo
Tel: 55-12-335-9000 Fax:55-12-331-3789

■ EUROPE

●Europe Sales Office:

Panasonic Industrial Europe GmbH [PIE]

• U.K. Sales Office:
Willoughby Road, Bracknell, Berks., RG12 8FP, THE UNITED KINGDOM
Tel: 44-1344-85-3671 Fax:44-1344-85-3853

• Germany Sales Office:
Hans-Pinsel-Strasse 2 85540 Haar, GERMANY
Tel: 49-89-46159-119 Fax:49-89-46159-195

■ ASIA

●Singapore Sales Office:

Panasonic Semiconductor of South Asia [PSSA]
300 Beach Road, #16-01, The Concourse, Singapore 199555 THE REPUBLIC OF SINGAPORE
Tel: 65-6390-3688 Fax:65-6390-3689

●Malaysia Sales Office:

Panasonic Industrial Company (M) Sdn. Bhd. [PICM]

• Head Office:
Tingkat 16B, Menara PKNS Petaling Jaya, No.17, Jalan Yong Shook Lin 46050 Petaling Jaya, Selangor Darul Ehsan, MALAYSIA
Tel: 60-3-7951-6601 Fax:60-3-7954-5968

• Penang Office:
Suite 20-07,20th Floor, MWE Plaza, No.8, Lebuh Farquhar,10200 Penang, MALAYSIA
Tel: 60-4-201-5113 Fax:60-4-261-9989

• Johore Sales Office:
Menara Pelangi, Suite8.3A, Level8, No.2, Jalan Kuning Taman Pelangi, 80400 Johor Bahru, Johor, MALAYSIA
Tel: 60-7-331-3822 Fax:60-7-355-3996

●Thailand Sales Office:

Panasonic Industrial (THAILAND) Ltd. [PICT]
252-133 Muang Thai-Phatra Complex Building, 31st Fl. Rachadaphisek Rd., Huaykwang, Bangkok 10320, THAILAND
Tel: 66-2-693-3428 Fax:66-2-693-3422

●Philippines Sales Office: [PISP]

Panasonic Indsutrial Sales Philippines Division of Matsushita Electric Philippines Corporation
102 Laguna Boulevard,Bo.Don Jose Laguna Technopark, Santa. Rosa, Laguna 4026 PHILIPPINES
Tel: 63-2-520-8615 Fax:63-2-520-8629

●India Sales Office:

National Panasonic India Ltd. [NPI]
E Block, 510, International Trade Tower Nehru Place, New Delhi_110019 INDIA
Tel: 91-11-629-2870 Fax:91-11-629-2877

●Indonesia Sales Office:

P.T.MET & Gobel [M&G]
JL. Dewi Sartika (Cawang 2) Jakarta 13630, INDONESIA
Tel: 62-21-801-5666 Fax:62-21-801-5675

●China Sales Office:

Panasonic Industrial (Shanghai) Co., Ltd. [PI(SH)]
Floor 6, Zhong Bao Mansion, 166 East Road Lujian Zui, PU Dong New District, Shanghai, 200120 CHINA
Tel: 86-21-5866-6114 Fax:86-21-5866-8000

Panasonic Industrial (Tianjin) Co., Ltd. [PI(TJ)]
Room No.1001, Tianjin International Building 75, Nanjin Road, Tianjin 300050, CHINA
Tel: 86-22-2313-9771 Fax:86-22-2313-9770

Panasonic SH Industrial Sales (Shenzhen) Co., Ltd. [PSI(SZ)]
7A-107, International Bussiness & Exhibition Centre, Futian Free Trade Zone, Shenzhen 518048, CHINA
Tel: 86-755-8359-8500 Fax:86-755-8359-8516

Panasonic Shun Hing Industrial Sales (Hong Kong) Co., Ltd. [PSI(HK)]
11th Floor, Great Eagle Center 23 Harbour Road, Wanchai, HONG KONG
Tel: 852-2529-7322 Fax:852-2865-3697

●Taiwan Sales Office:

Panasonic Industrial Sales (Taiwan) Co.,Ltd. [PIST]

• Head Office:
6F, 550, Sec. 4, Chung Hsiao E. RD. Taipei, 110, TAIWAN
Tel: 886-2-2757-1900 Fax:886-2-2757-1906

• Kaohsiung Office:
6th Floor, Hsin Kong Bldg. No.251, Chi Hsien 1st Road Kaohsiung 800, TAIWAN
Tel: 886-7-346-3815 Fax:886-7-236-8362

●Korea Sales Office:

Panasonic Industrial Korea Co., Ltd. [PIKL]
Kukje Center Bldg. 11th Fl., 191 Hangangro 2ga, Youngsan-ku, Seoul 140-702, KOREA
Tel: 82-2-795-9600 Fax:82-2-795-1542

050702

Printed in JAPAN