REJ09B0018-0110Z

8

7540 Group

User's Manual

RENESAS 8-BIT CISC SINGLE-CHIP MICROCOMPUTER

740 FAMILY / 740 SERIES

User's Manual

Before using this material, please visit our website to confirm that this is the most current document available.

Rev. 1.10
Revision date: May. 28, 2003

Renesas Technology
www.renesas.com

Keep safety first in your circuit designs!

- Renesas Technology Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage. Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

- These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corporation product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corporation or a third party.
- Renesas Technology Corporation assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
- All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor for the latest product information before purchasing a product listed herein.
  The information described here may contain technical inaccuracies or typographical errors. Renesas Technology Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
  Please also pay attention to information published by Renesas Technology Corporation by various means, including the Renesas Technology Corporation Semiconductor home page (http://www.renesas.com).
- When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
- Renesas Technology Corporation semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
- The prior written approval of Renesas Technology Corporation is necessary to reprint or reproduce in whole or in part these materials.
- If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
  Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
- Please contact Renesas Technology Corporation for further details on these materials or t he products contained therein.

REVISION HISTORY	7540 Group User's Manual

Rev.	Date	Description	
		Page	Summary
1.00	Sep. 17, 2002	–	First edition issued
1.10	May 28, 2003	1-17	[Pull-up control register] PULL; Note added.
			Fig.15; Note 2 eliminated.
		1-19	Fig.17; (2) Ports $P0_1$,$P0_2$ revised.
		1-34	Fig.29; Port $P0_3$ direction register block, Port $P0_1$ direction register block and Port $P0_2$ direction register block revised.
		1-45	(3) RC oscillation revised.
		2-49	Fig.2.4.12; RTI → RTS
		2-52	Fig.2.4.16; Prescaler X 1/4 → 1/2, $CNTR_0$ pin output 4 MHz → 4 kHz
		2-53	Fig.2.4.17; The second CPUM setting $00000X00_2$ → $11000X00_2$
			Prescaler X 03_{16} → 01_{16}, Note 2 revised.
		2-61	Fig.2.4.26; The followings are revised.
			The second setting of $CNTR_0$ interrupt enable bit and Timer X interrupt enable bit.
			The second setting of timer X mode register.
		3-86	Fig.3.3.5; NOP added.

BEFORE USING THIS MANUAL

This user's manual consists of the following three chapters. Refer to the chapter appropriate to your conditions, such as hardware design or software development. Chapter 3 also includes necessary information for systems development. You must refer to that chapter.

1. Organization

● **CHAPTER 1 HARDWARE**
This chapter describes features of the microcomputer and operation of each peripheral function.

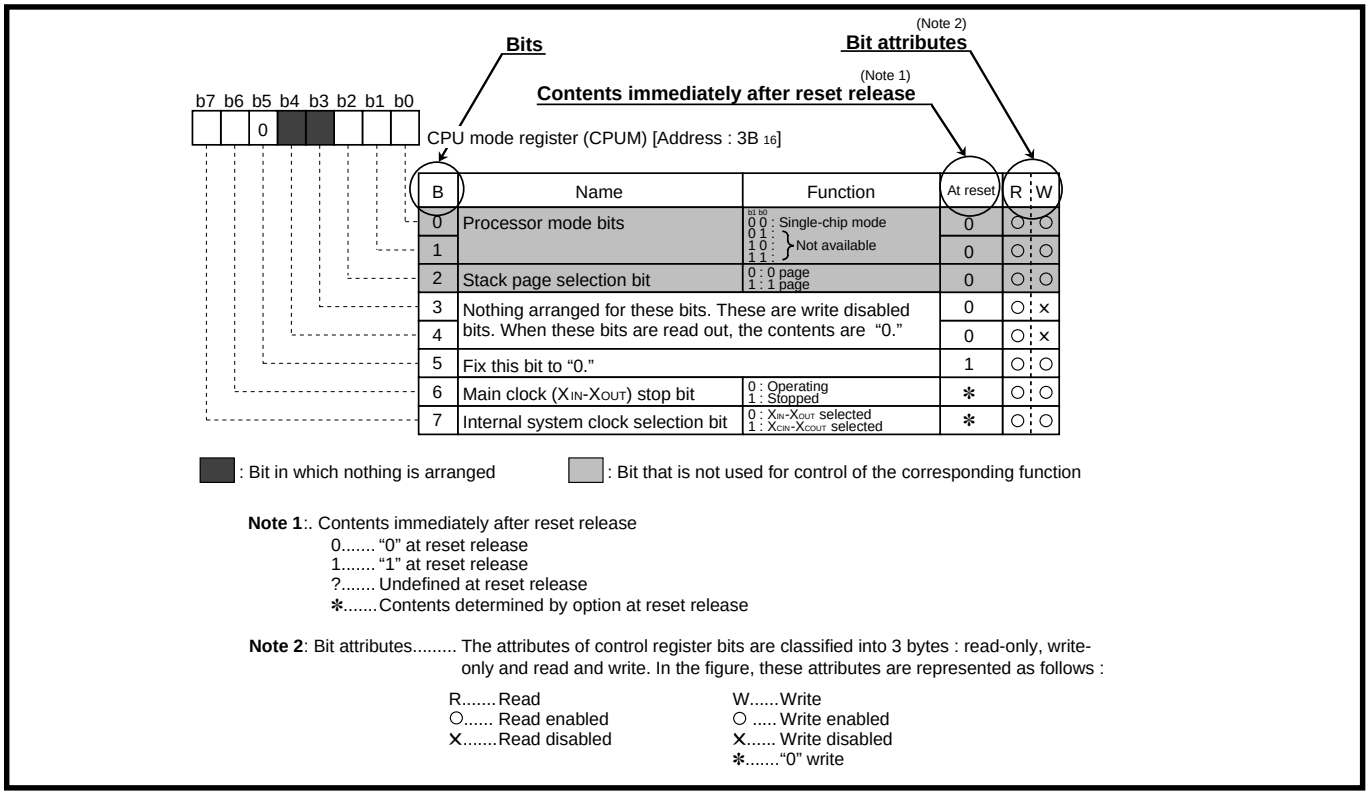
● **CHAPTER 2 APPLICATION**
This chapter describes usage and application examples of peripheral functions, based mainly on setting examples of relevant registers.

● **CHAPTER 3 APPENDIX**
This chapter includes necessary information for systems development using the microcomputer, such as the electrical characteristics, the list of registers.

As for the Mask ROM confirmation form, the ROM programming confirmation form, and the Mark specification form which are to be submitted when ordering, refer to the "Renesas Technology Corp" Homepage (http://www.renesas.com/en/rom).

2. Structure of register

The figure of each register structure describes its functions, contents at reset, and attributes as follows :



Note 1:. Contents immediately after reset release
0..... "0" at reset release
1..... "1" at reset release
?..... Undefined at reset release
✻..... Contents determined by option at reset release

Note 2: Bit attributes..... The attributes of control register bits are classified into 3 bytes : read-only, write-only and read and write. In the figure, these attributes are represented as follows :

R..... Read
○..... Read enabled
×..... Read disabled

W..... Write
○ ..... Write enabled
×..... Write disabled
✻..... "0" write

3. Supplementation

For details of software, refer to the "740 FAMILY SOFTWARE MANUAL."
For development tools, refer to the "Renesas Technology Corp" Homepage (http://www.renesas.com/en/tools).

Table of contents

CHAPTER 1 HARDWARE

Table of contents

CHAPTER 2 APPLICATION

CHAPTER 3 APPENDIX

List of figures

CHAPTER 1 HARDWARE

CHAPTER 2 APPLICATION

List of figures

CHAPTER 3 APPENDIX

List of figures

List of figures

List of tables

CHAPTER 1 HARDWARE

CHAPTER 2 APPLICATION

CHAPTER 3 APPENDIX

CHAPTER 1
HARDWARE

DESCRIPTION

The 7540 Group is the 8-bit microcomputer based on the 740 family core technology.

The 7540 Group has a serial I/O, 8-bit timers, a 16-bit timer, and an A-D converter, and is useful for control of home electric appliances and office automation equipment.

FEATURES

- Basic machine-language instructions 71
- The minimum instruction execution time ..... 0.34 µs
  (at 6 MHz oscillation frequency, double-speed mode for the shortest instruction)
- Memory size ROM ..... 8 K to 32 K bytes
  RAM ..... 384 to 768 bytes
- Programmable I/O ports ..... 29 (25 in 32-pin version)
- Interrupts ..... 15 sources, 15 vectors
  ..... (14 sources, 14 vectors for 32-pin version)
- Timers ..... 8-bit × 4
  ..... 16-bit × 1
- Serial I/O1 ..... 8-bit × 1 (UART or Clock-synchronized)
- Serial I/O2 (**Note 1**) ..... 8-bit × 1 (Clock-synchronized)
- A-D converter ..... 10-bit × 8 channels
  ..... (6 channels for 32-pin version)
- Clock generating circuit ..... Built-in type
  (low-power dissipation by a ring oscillator enabled)
  (connect to external ceramic resonator or quartz-crystal oscillator permitting RC oscillation)
- Watchdog timer ..... 16-bit × 1
- Power source voltage
  X_{IN} oscillation frequency at ceramic oscillation, in double-speed mode
  At 6 MHz ..... 4.5 to 5.5 V
  X_{IN} oscillation frequency at ceramic oscillation, in high-speed mode
  At 8 MHz ..... 4.0 to 5.5 V
  At 4 MHz ..... 2.4 to 5.5 V
  At 2 MHz ..... 2.2 to 5.5 V
  X_{IN} oscillation frequency at RC oscillation in high-speed mode or middle-speed mode
  At 4 MHz ..... 4.0 to 5.5 V
  At 2 MHz ..... 2.4 to 5.5 V
  At 1 MHz ..... 2.2 to 5.5 V
- Power dissipation
  Mask ROM version ..... 22.5 mW (standard)
  One Time PROM version ..... 30 mW (standard)
- Operating temperature range ..... –20 to 85 °C
  (–40 to 85 °C for extended operating temperature version)
  (–40 to 125 °C for extended operating temperature 125 °C version (**Note 2**))

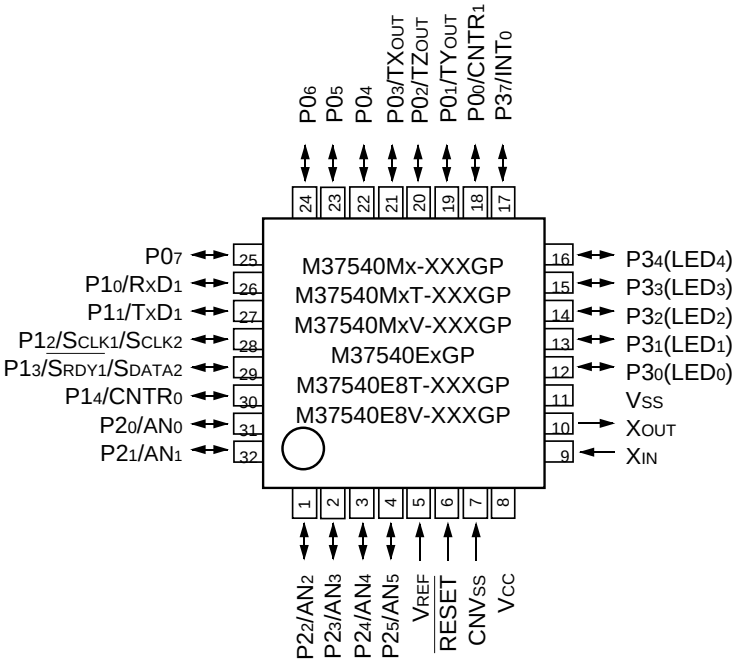
APPLICATION

Office automation equipment, factory automation equipment, home electric appliances, consumer electronics, car, etc.

Notes 1: Serial I/O2 can be used in the following cases;
(1) Serial I/O1 is not used,
(2) Serial I/O1 is used as UART and BRG output divided by 16 is selected as the synchronized clock.

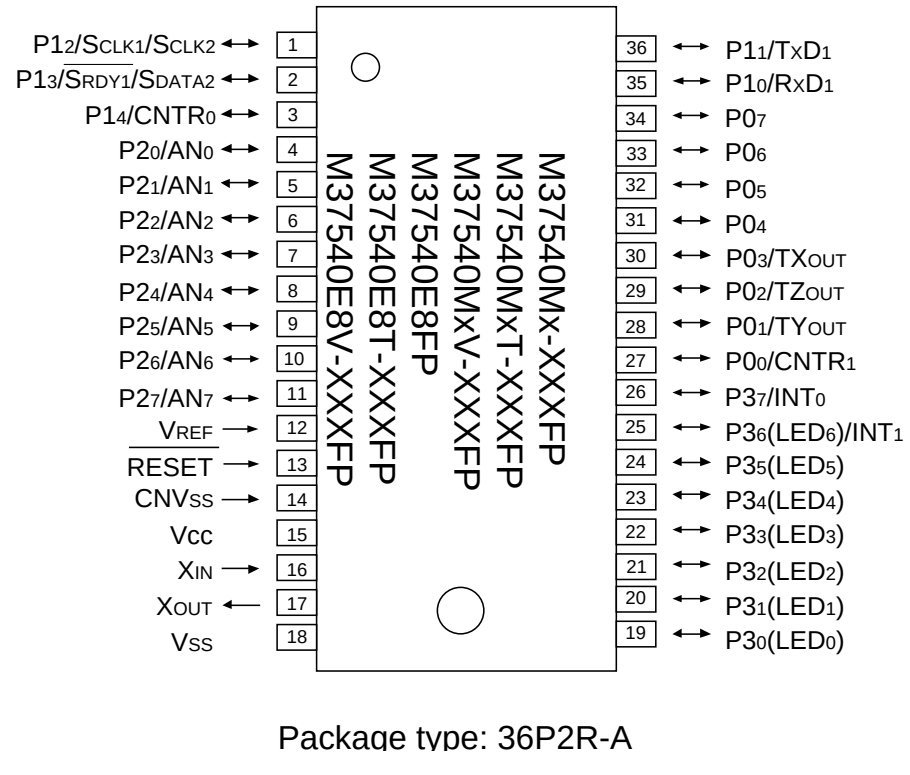
2: In this version, the operating temperature range and total time are limited as follows;
55 °C to 85 °C: within total 6000 hours,
85 °C to 125 °C: within total 1000 hours.

PIN CONFIGURATION (TOP VIEW)



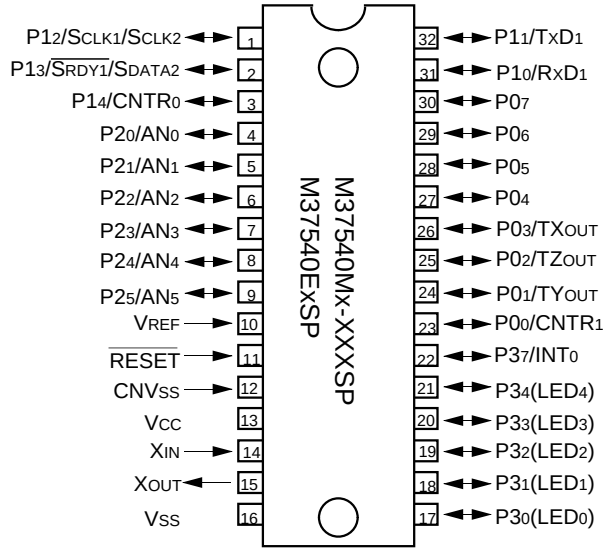
Package type: 32P6U-A

Fig. 1 Pin configuration (32P6U-A type)



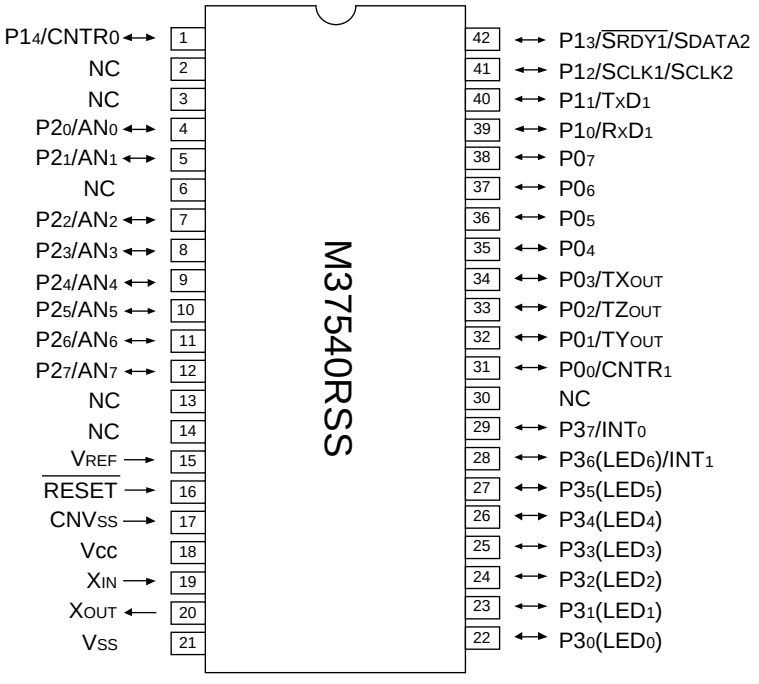
Package type: 36P2R-A

Fig. 2 Pin configuration (36P2R-A type)



Package type: 32P4B

Fig. 3 Pin configuration (32P4B-A type)



Outline 42S1M

Fig. 4 Pin configuration (42S1M type)

FUNCTIONAL BLOCK

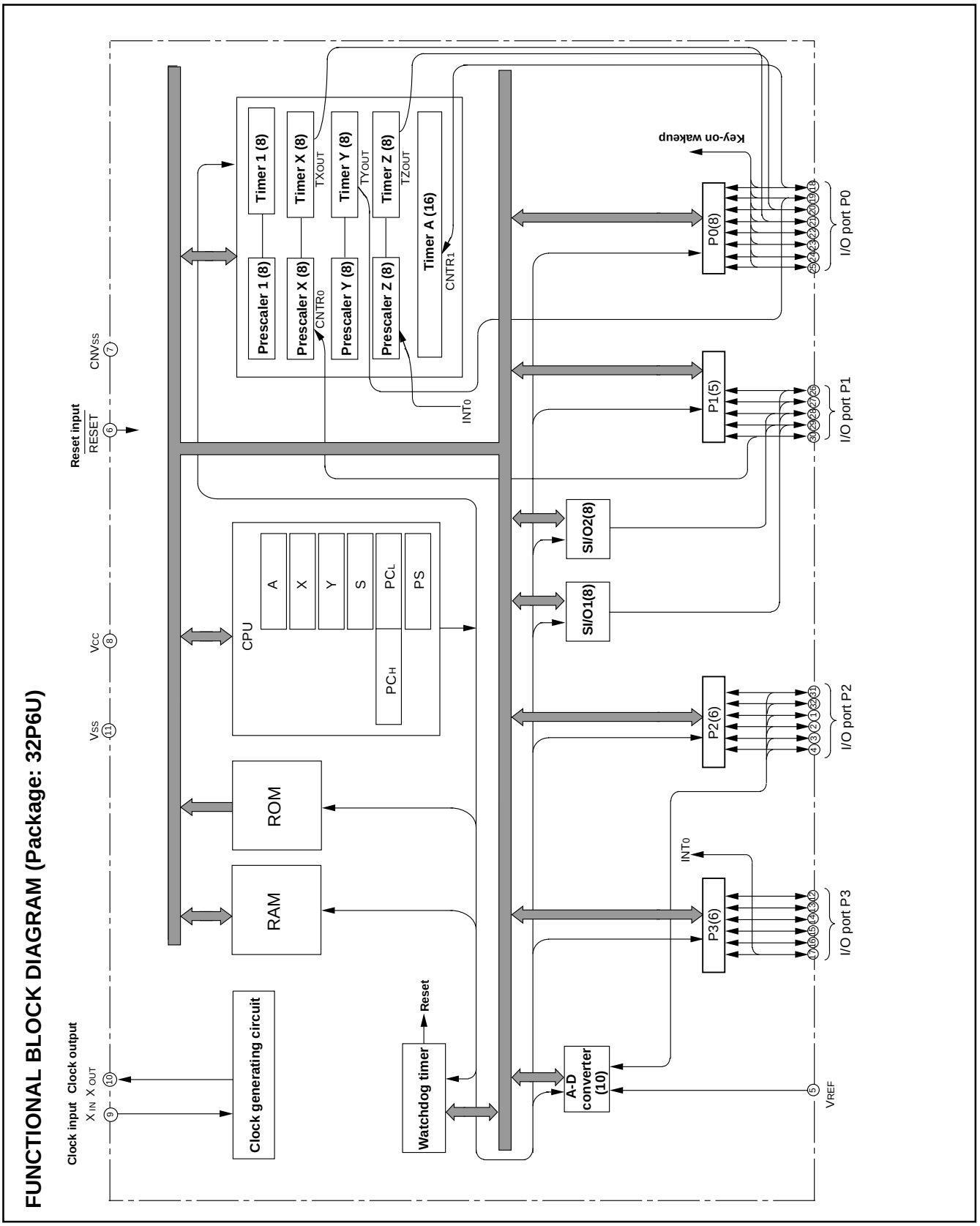


Fig. 5 Functional block diagram (32P6U package)

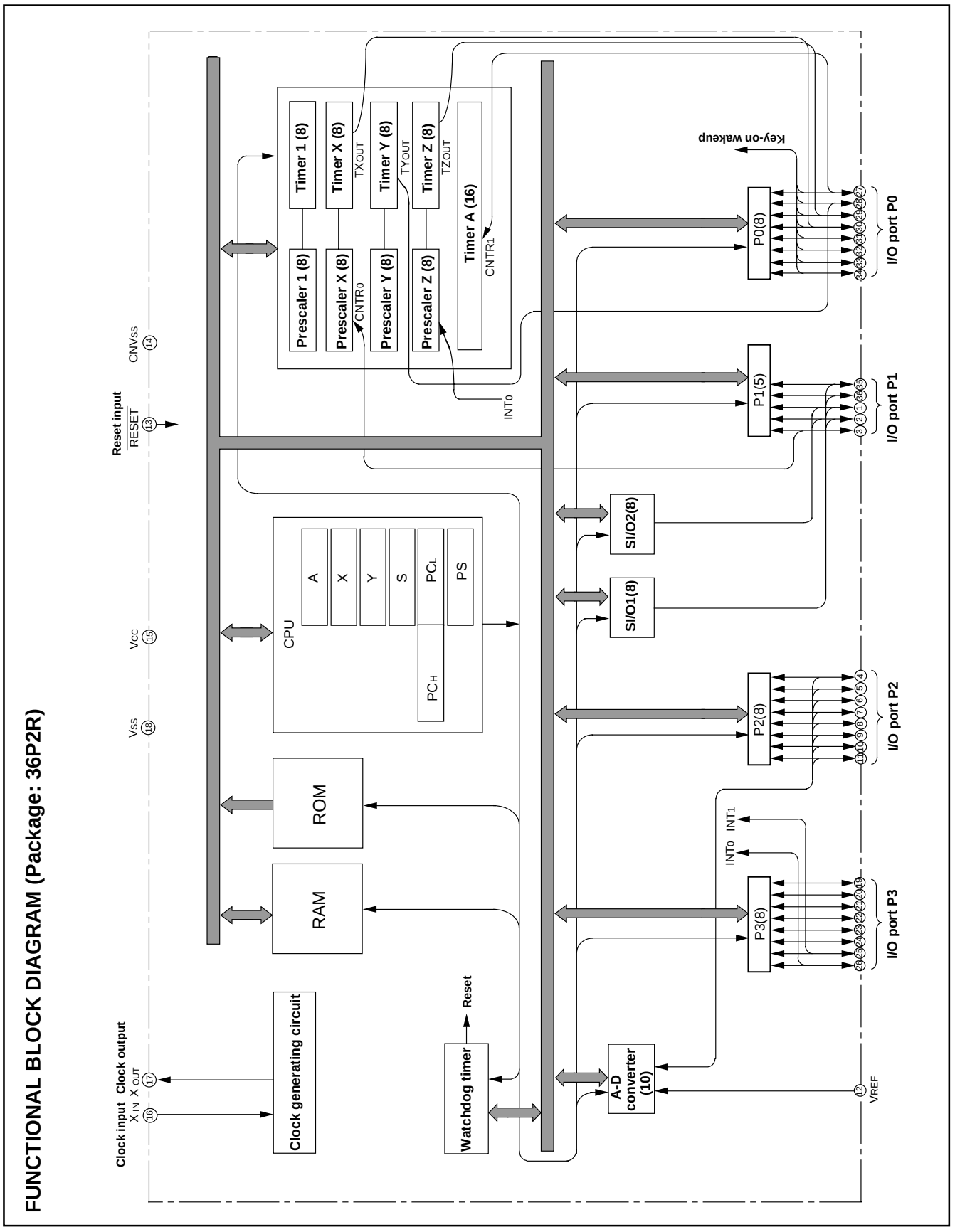


Fig. 6 Functional block diagram (36P2R package)

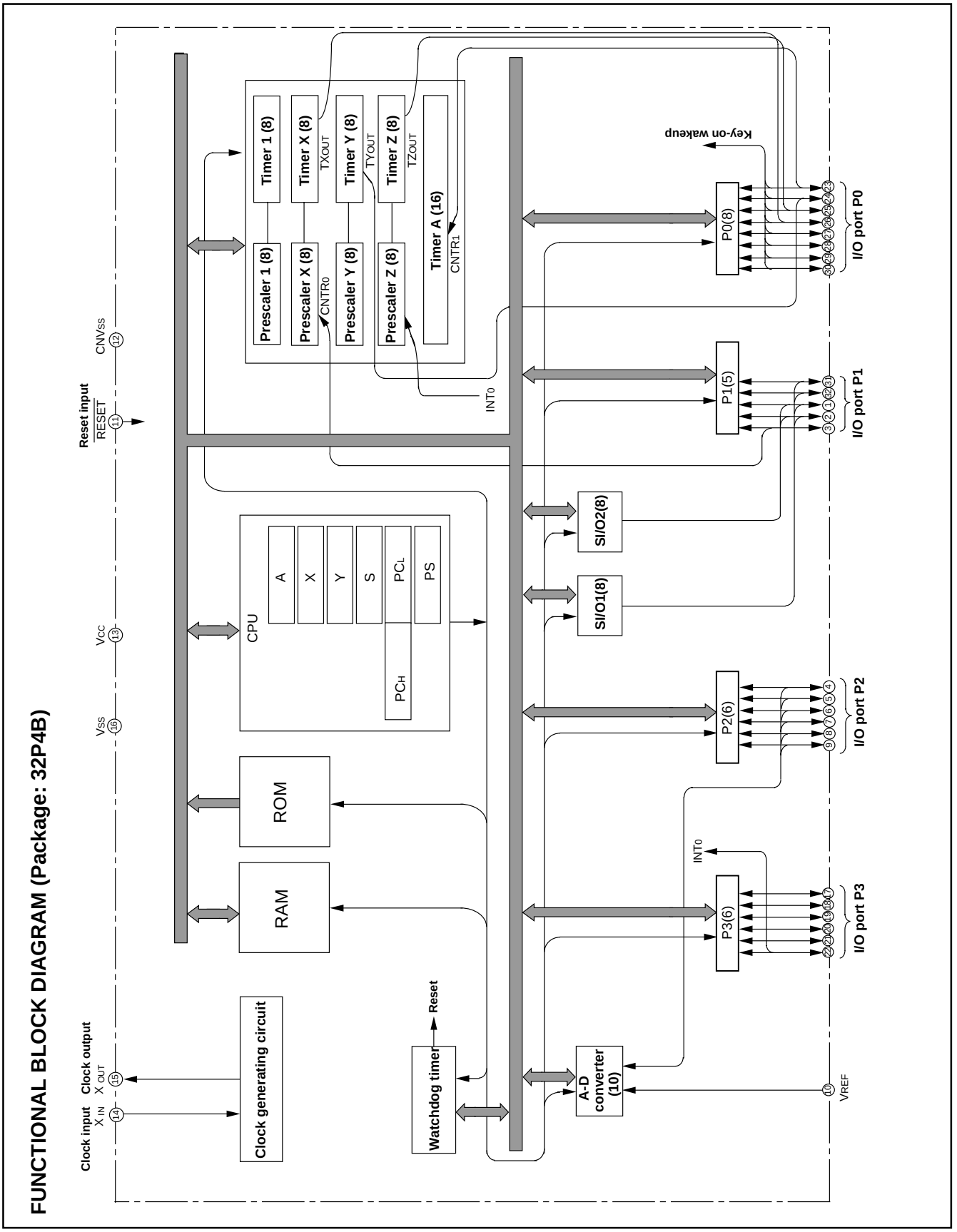


Fig. 7 Functional block diagram (32P4B package)

PIN DESCRIPTION

Table 1 Pin description

Pin	Name	Function	Function expect a port function
Vcc, Vss	Power source (Note 1)	•Apply voltage of 2.2 to 5.5 V to Vcc, and 0 V to Vss.	
V_{REF}	Analog reference voltage	•Reference voltage input pin for A-D converter	
CNVss	CNVss	•Chip operating mode control pin, which is always connected to Vss.	
$\overline{RESET}$	Reset input	•Reset input pin for active "L"	
X_{IN}	Clock input	•Input and output pins for main clock generating circuit •Connect a ceramic resonator or quartz crystal oscillator between the X_{IN} and X_{OUT} pins. •For using RC oscillator, short between the X_{IN} and X_{OUT} pins, and connect the capacitor and resistor. •If an external clock is used, connect the clock source to the X_{IN} pin and leave the X_{OUT} pin open. • When the ring oscillator is selected as the main clock, connect X_{IN} pin to V_{SS} and leave X_{OUT} open.	
X_{OUT}	Clock output		
$P0_0$/$CNTR_1$ $P0_1$/TY_{OUT} $P0_2$/TZ_{OUT} $P0_3$/TX_{OUT} $P0_4$–$P0_7$	I/O port P0	•8-bit I/O port. •I/O direction register allows each pin to be individually programmed as either input or output. •CMOS compatible input level •CMOS 3-state output structure •Whether a built-in pull-up resistor is to be used or not can be determined by program.	• Key-input (key-on wake up interrupt input) pins • Timer Y, timer Z, timer X and timer A function pin
$P1_0$/RxD_1 $P1_1$/TxD_1	I/O port P1	•5-bit I/O port •I/O direction register allows each pin to be individually programmed as either input or output. •CMOS compatible input level •CMOS 3-state output structure •CMOS/TTL level can be switched for $P1_0$, $P1_2$ and $P1_3$	• Serial I/O1 function pin
$P1_2$/$SCLK_1$/$SCLK_2$ $P1_3$/$\overline{SRDY_1}$/$SDATA_2$			• Serial I/O1 function pin • Serial I/O2 function pin
$P1_4$/$CNTR_0$			• Timer X function pin
$P2_0$/AN_0–$P2_7$/AN_7	I/O port P2 (Note 2)	•8-bit I/O port having almost the same function as P0 •CMOS compatible input level •CMOS 3-state output structure	• Input pins for A-D converter
$P3_0$–$P3_5$	I/O port P3 (Note 3)	•8-bit I/O port •I/O direction register allows each pin to be individually programmed as either input or output. •CMOS compatible input level (CMOS/TTL level can be switched for $P3_6$ and $P3_7$). •CMOS 3-state output structure •$P3_0$ to $P3_6$ can output a large current for driving LED.	
$P3_6$/INT_1 $P3_7$/INT_0		•Whether a built-in pull-up resistor is to be used or not can be determined by program.	• Interrupt input pins

Notes 1: V_{CC} = 2.4 to 5.5 V for the extended operating temperature version and the extended operating temperature 125 °C version.
2: $P2_6$/AN_6 and $P2_7$/AN_7 do not exist for the 32-pin version, so that Port P2 is a 6-bit I/O port.
3: $P3_5$ and $P3_6$/INT_1 do not exist for the 32-pin version, so that Port P3 is a 6-bit I/O port.

GROUP EXPANSION

We plan to expand the 7540 group as follow:

Memory type

Support for Mask ROM version, One Time PROM version, and Emulator MCU .

Memory size

ROM/PROM size ..... 8 K to 32 K bytes
RAM size ..... 384 to 768 bytes

Package

32P4B ..... 32-pin plastic molded SDIP
32P6U-A ..... 0.8 mm-pitch 32-pin plastic molded LQFP
36P2R-A ..... 0.8 mm-pitch 36-pin plastic molded SSOP
42S1M ..... 42-pin shrink ceramic PIGGY BACK

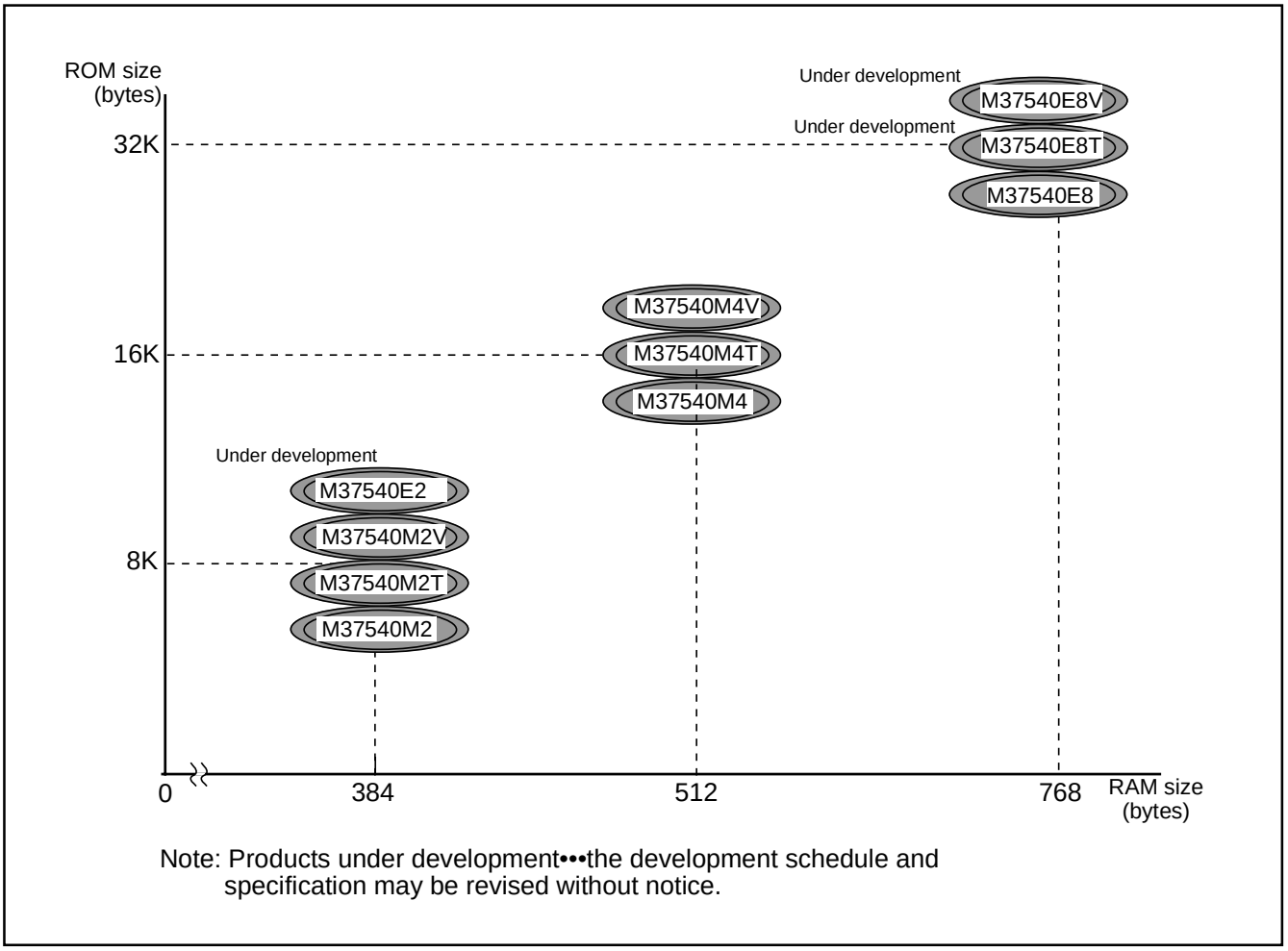


Fig. 8 Memory expansion plan

Currently supported products are listed below.

Table 2 List of supported products

Part Number	(P) ROM size (bytes) ROM size for User ()	RAM size (bytes)	Package	Remarks
M37540M2-XXXSP	8192 (8062)	384	32P4B	Mask ROM version
M37540M2-XXXFP			36P2R-A	Mask ROM version
M37540M2T-XXXFP				Mask ROM version (extended operating temperature version)
M37540M2V-XXXFP				Mask ROM version (extended operating temperature 125 °C version)
M37540M2-XXXGP			32P6U-A	Mask ROM version
M37540M2T-XXXGP				Mask ROM version (extended operating temperature version)
M37540M2V-XXXGP				Mask ROM version (extended operating temperature 125 °C version)
M37540M4-XXXSP	16384 (16254)	512	32P4B	Mask ROM version
M37540M4-XXXFP			36P2R-A	Mask ROM version
M37540M4T-XXXFP				Mask ROM version (extended operating temperature version)
M37540M4V-XXXFP				Mask ROM version (extended operating temperature 125 °C version)
M37540M4-XXXGP			32P6U-A	Mask ROM version
M37540M4T-XXXGP				Mask ROM version (extended operating temperature version)
M37540M4V-XXXGP				Mask ROM version (extended operating temperature 125 °C version)
M37540E2SP*	8192 (8062)	384	32P4B	One Time PROM version (blank)
M37540E2FP*			36P2R-A	One Time PROM version (blank)
M37540E2GP*			32P6U-A	One Time PROM version (blank)
M37540E8SP	32768 (32638)	768	32P4B	One Time PROM version (blank)
M37540E8FP			36P2R-A	One Time PROM version (blank)
M37540E8T-XXXFP*				One Time PROM version (shipped after programming, extended operating temperature version)
M37540E8V-XXXFP*				One Time PROM version (shipped after programming, extended operating temperature 125 °C version)
M37540E8GP			32P6U-A	One Time PROM version (blank)
M37540E8T-XXXGP*				One Time PROM version (shipped after programming, extended operating temperature version)
M37540E8V-XXXGP*				One Time PROM version (shipped after programming, extended operating temperature 125 °C version)
M37540RSS	————	768	42S1M	Emulator MCU

*: Under development

FUNCTIONAL DESCRIPTION

Central Processing Unit (CPU)

The MCU uses the standard 740 family instruction set. Refer to the table of 740 family addressing modes and machine-language instructions or the SERIES 740 <SOFTWARE> USER'S MANUAL for details on each instruction set.

Machine-resident 740 family instructions are as follows:

1. The FST and SLW instructions cannot be used.
2. The MUL and DIV instructions can be used.
3. The WIT instruction can be used.
4. The STP instruction can be used.

This instruction cannot be used while CPU operates by a ring oscillator.

Accumulator (A)

The accumulator is an 8-bit register. Data operations such as data transfer, etc., are executed mainly through the accumulator.

Index register X (X), Index register Y (Y)

Both index register X and index register Y are 8-bit registers. In the index addressing modes, the value of the OPERAND is added to the contents of register X or register Y and specifies the real address.

When the T flag in the processor status register is set to "1", the value contained in index register X becomes the address for the second OPERAND.

Stack pointer (S)

The stack pointer is an 8-bit register used during subroutine calls and interrupts. The stack is used to store the current address data and processor status when branching to subroutines or interrupt routines.

The lower eight bits of the stack address are determined by the contents of the stack pointer. The upper eight bits of the stack address are determined by the Stack Page Selection Bit. If the Stack Page Selection Bit is "0", then the RAM in the zero page is used as the stack area. If the Stack Page Selection Bit is "1", then RAM in page 1 is used as the stack area.

The Stack Page Selection Bit is located in the SFR area in the zero page. Note that the initial value of the Stack Page Selection Bit varies with each microcomputer type. Also some microcomputer types have no Stack Page Selection Bit and the upper eight bits of the stack address are fixed. The operations of pushing register contents onto the stack and popping them from the stack are shown in Fig. 9.

Program counter (PC)

The program counter is a 16-bit counter consisting of two 8-bit registers PC_H and PC_L. It is used to indicate the address of the next instruction to be executed.

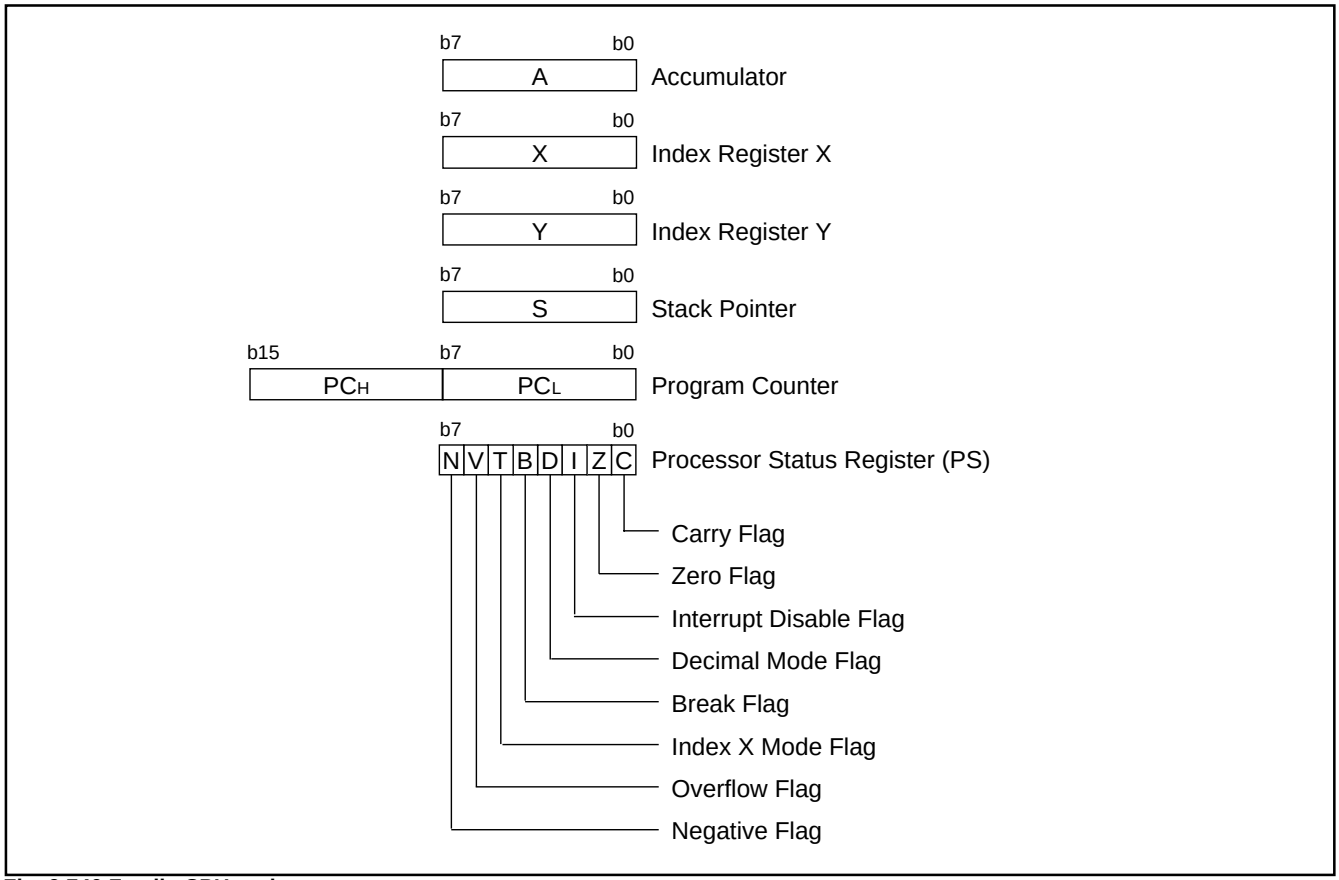


Fig. 9 740 Family CPU register structure

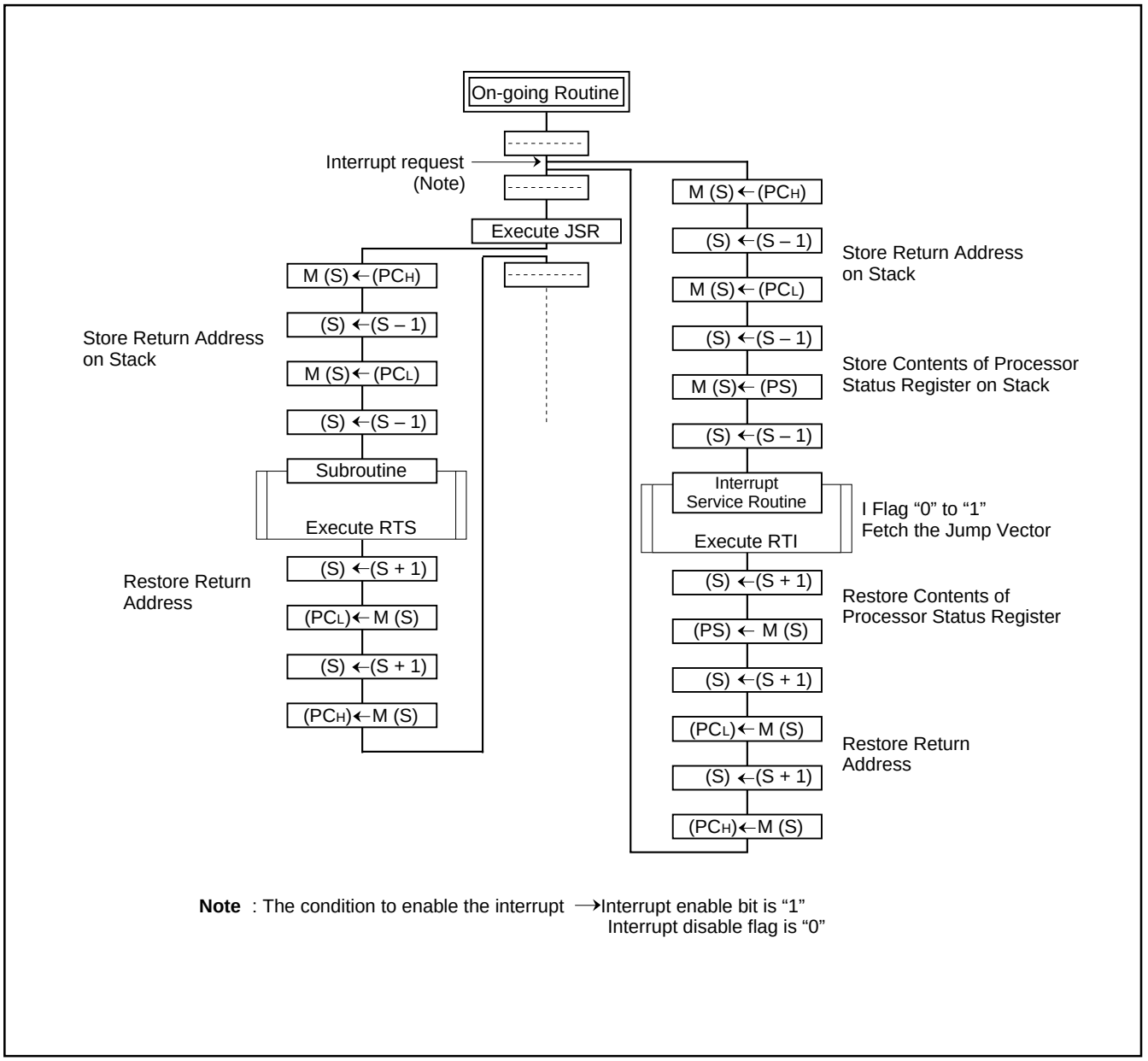


Fig. 10 Register push and pop at interrupt generation and subroutine call

Table 3 Push and pop instructions of accumulator or processor status register

	Push instruction to stack	Pop instruction from stack
Accumulator	PHA	PLA
Processor status register	PHP	PLP

Processor status register (PS)

The processor status register is an 8-bit register consisting of flags which indicate the status of the processor after an arithmetic operation. Branch operations can be performed by testing the Carry (C) flag, Zero (Z) flag, Overflow (V) flag, or the Negative (N) flag. In decimal mode, the Z, V, N flags are not valid.
After reset, the Interrupt disable (I) flag is set to "1", but all other flags are undefined. Since the Index X mode (T) and Decimal mode (D) flags directly affect arithmetic operations, they should be initialized in the beginning of a program.

(1) Carry flag (C)
The C flag contains a carry or borrow generated by the arithmetic logic unit (ALU) immediately after an arithmetic operation. It can also be changed by a shift or rotate instruction.

(2) Zero flag (Z)
The Z flag is set if the result of an immediate arithmetic operation or a data transfer is "0", and cleared if the result is anything other than "0".

(3) Interrupt disable flag (I)
The I flag disables all interrupts except for the interrupt generated by the BRK instruction. Interrupts are disabled when the I flag is "1".
When an interrupt occurs, this flag is automatically set to "1" to prevent other interrupts from interfering until the current interrupt is serviced.

(4) Decimal mode flag (D)
The D flag determines whether additions and subtractions are executed in binary or decimal. Binary arithmetic is executed when this flag is "0"; decimal arithmetic is executed when it is "1".
Decimal correction is automatic in decimal mode. Only the ADC and SBC instructions can be used for decimal arithmetic.

(5) Break flag (B)
The B flag is used to indicate that the current interrupt was generated by the BRK instruction. The BRK flag in the processor status register is always "0". When the BRK instruction is used to generate an interrupt, the processor status register is pushed onto the stack with the break flag set to "1". The saved processor status is the only place where the break flag is ever set.

(6) Index X mode flag (T)
When the T flag is "0", arithmetic operations are performed between accumulator and memory, e.g. the results of an operation between two memory locations is stored in the accumulator. When the T flag is "1", direct arithmetic operations and direct data transfers are enabled between memory locations, i.e. between memory and memory, memory and I/O, and I/O and I/O. In this case, the result of an arithmetic operation performed on data in memory location 1 and memory location 2 is stored in memory location 1. The address of memory location 1 is specified by index register X, and the address of memory location 2 is specified by normal addressing modes.

(7) Overflow flag (V)
The V flag is used during the addition or subtraction of one byte of signed data. It is set if the result exceeds +127 to -128. When the BIT instruction is executed, bit 6 of the memory location operated on by the BIT instruction is stored in the overflow flag.

(8) Negative flag (N)
The N flag is set if the result of an arithmetic operation or data transfer is negative. When the BIT instruction is executed, bit 7 of the memory location operated on by the BIT instruction is stored in the negative flag.

Table 4 Set and clear instructions of each bit of processor status register

	C flag	Z flag	I flag	D flag	B flag	T flag	V flag	N flag
Set instruction	SEC	–	SEI	SED	–	SET	–	–
Clear instruction	CLC	–	CLI	CLD	–	CLT	CLV	–

[CPU mode register] CPUM

The CPU mode register contains the stack page selection bit. This register is allocated at address 003B16.

Switching method of CPU mode register

Switch the CPU mode register (CPUM) at the head of program after releasing Reset in the following method.

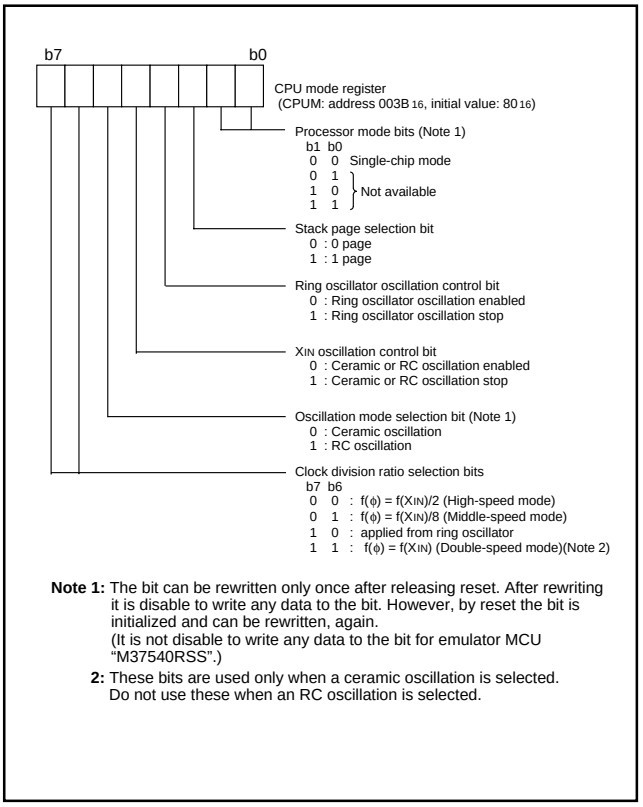


Fig. 11 Structure of CPU mode register

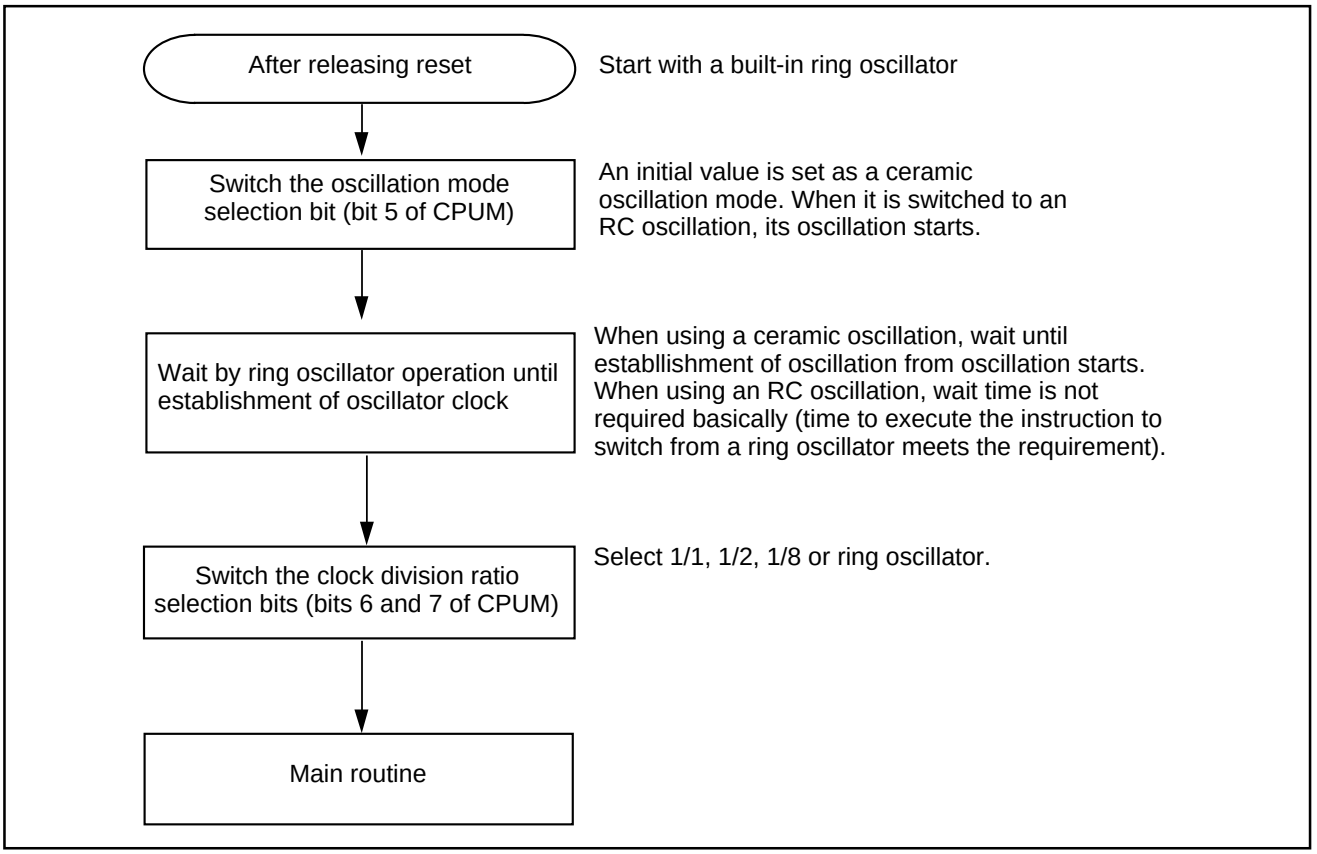


Fig. 12 Switching method of CPU mode register

Memory

Special function register (SFR) area

The SFR area in the zero page contains control registers such as I/O ports and timers.

RAM

RAM is used for data storage and for a stack area of subroutine calls and interrupts.

ROM

The first 128 bytes and the last 2 bytes of ROM are reserved for device testing and the rest is a user area for storing programs.

Interrupt vector area

The interrupt vector area contains reset and interrupt vectors.

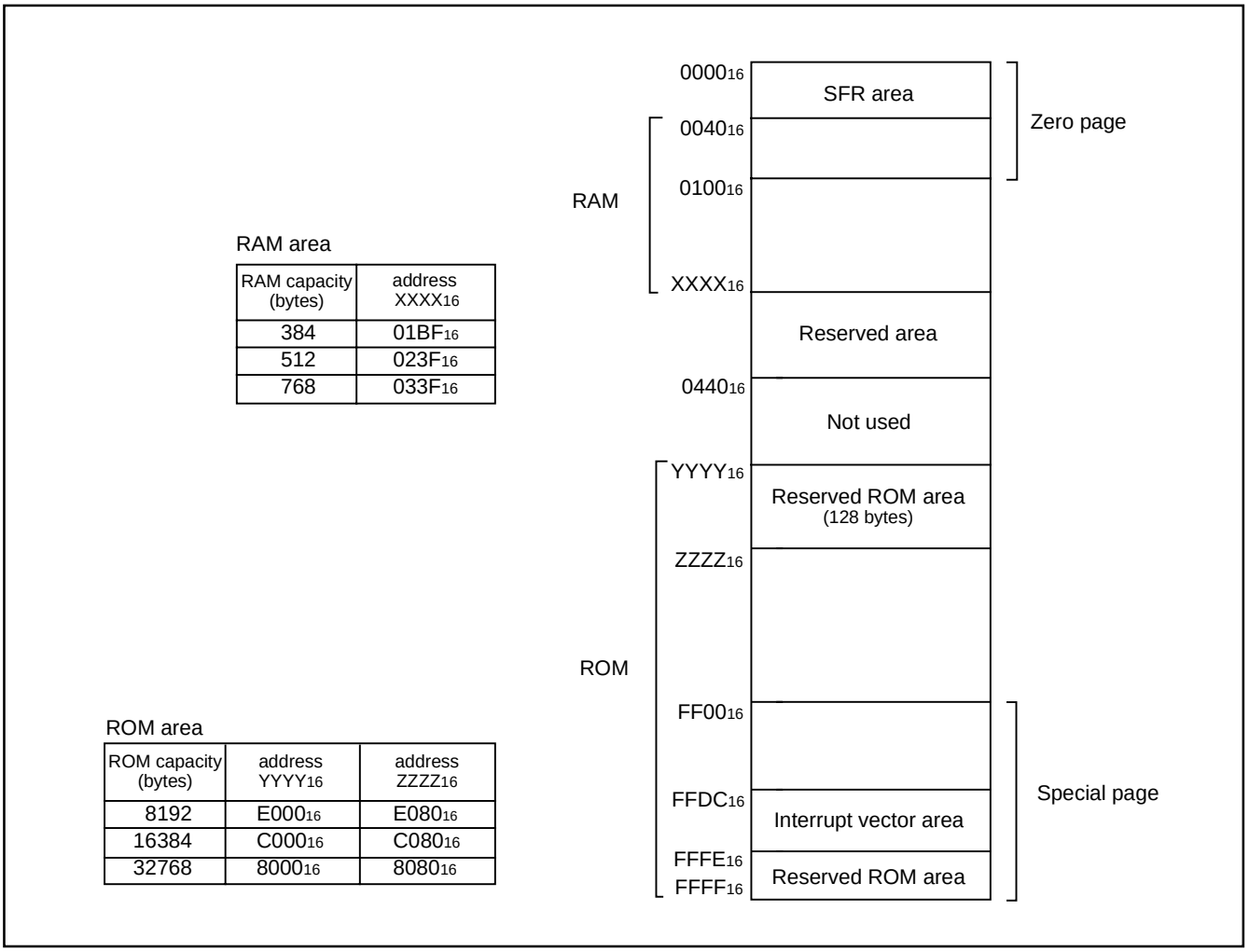
Zero page

The 256 bytes from addresses 0000_{16} to $00FF_{16}$ are called the zero page area. The internal RAM and the special function registers (SFR) are allocated to this area.

The zero page addressing mode can be used to specify memory and register addresses in the zero page area. Access to this area with only 2 bytes is possible in the zero page addressing mode.

Special page

The 256 bytes from addresses $FF00_{16}$ to $FFFF_{16}$ are called the special page area. The special page addressing mode can be used to specify memory addresses in the special page area. Access to this area with only 2 bytes is possible in the special page addressing mode.



RAM area

RAM capacity (bytes)	address XXXX16
384	$01BF_{16}$
512	$023F_{16}$
768	$033F_{16}$

ROM area

ROM capacity (bytes)	address YYYY16	address ZZZZ16
8192	$E000_{16}$	$E080_{16}$
16384	$C000_{16}$	$C080_{16}$
32768	8000_{16}	8080_{16}

Fig. 13 Memory map diagram

0000_{16}	Port P0 (P0)	0020_{16}	Timer Y, Z mode register (TYZM)
0001_{16}	Port P0 direction register (P0D)	0021_{16}	Prescaler Y (PREY)
0002_{16}	Port P1 (P1)	0022_{16}	Timer Y secondary (TYS)
0003_{16}	Port P1 direction register (P1D)	0023_{16}	Timer Y primary (TYP)
0004_{16}	Port P2 (P2)	0024_{16}	Timer Y, Z waveform output control register (PUM)
0005_{16}	Port P2 direction register (P2D)	0025_{16}	Prescaler Z (PREZ)
0006_{16}	Port P3 (P3)	0026_{16}	Timer Z secondary (TZS)
0007_{16}	Port P3 direction register (P3D)	0027_{16}	Timer Z primary (TZP)
0008_{16}		0028_{16}	Prescaler 1 (PRE1)
0009_{16}		0029_{16}	Timer 1 (T1)
$000A_{16}$		$002A_{16}$	One-shot start register (ONS)
$000B_{16}$		$002B_{16}$	Timer X mode register (TXM)
$000C_{16}$		$002C_{16}$	Prescaler X (PREX)
$000D_{16}$		$002D_{16}$	Timer X (TX)
$000E_{16}$		$002E_{16}$	Timer count source set register (TCSS)
$000F_{16}$		$002F_{16}$	
0010_{16}		0030_{16}	Serial I/O2 control register (SIO2CON)
0011_{16}		0031_{16}	Serial I/O2 register (SIO2)
0012_{16}		0032_{16}	
0013_{16}		0033_{16}	
0014_{16}		0034_{16}	A-D control register (ADCON)
0015_{16}		0035_{16}	A-D conversion register (low-order) (ADL)
0016_{16}	Pull-up control register (PULL)	0036_{16}	A-D conversion register (high-order) (ADH)
0017_{16}	Port P1P3 control register (P1P3C)	0037_{16}	
0018_{16}	Transmit/Receive buffer register (TB/RB)	0038_{16}	MISRG
0019_{16}	Serial I/O1 status register (SIO1STS)	0039_{16}	Watchdog timer control register (WDTCON)
$001A_{16}$	Serial I/O1 control register (SIO1CON)	$003A_{16}$	Interrupt edge selection register (INTEDGE)
$001B_{16}$	UART control register (UARTCON)	$003B_{16}$	CPU mode register (CPUM)
$001C_{16}$	Baud rate generator (BRG)	$003C_{16}$	Interrupt request register 1 (IREQ1)
$001D_{16}$	Timer A mode register (TAM)	$003D_{16}$	Interrupt request register 2 (IREQ2)
$001E_{16}$	Timer A (low-order) (TAL)	$003E_{16}$	Interrupt control register 1 (ICON1)
$001F_{16}$	Timer A (high-order) (TAH)	$003F_{16}$	Interrupt control register 2 (ICON2)

Note : Do not access to the SFR area including nothing.

Fig. 14 Memory map of special function register (SFR)

I/O Ports

[Direction registers] PiD

The I/O ports have direction registers which determine the input/output direction of each pin. Each bit in a direction register corresponds to one pin, and each pin can be set to be input or output.
When "1" is set to the bit corresponding to a pin, this pin becomes an output port. When "0" is set to the bit, the pin becomes an input port.
When data is read from a pin set to output, not the value of the pin itself but the value of port latch is read. Pins set to input are floating, and permit reading pin values.
If a pin set to input is written to, only the port latch is written to and the pin remains floating.

[Pull-up control register] PULL

By setting the pull-up control register (address 0016_{16}), ports P0 and P3 can exert pull-up control by program. However, pins set to output are disconnected from this control and cannot exert pull-up control.

Note: $P2_6/AN_6$, $P2_7/AN_7$, $P3_5$ and $P3_6$ do not exist for the 32-pin version.
Accordingly, the following settings are required;
• Set direction registers of ports $P2_6$ and $P2_7$ to output.
• Set direction registers of ports $P3_5$ and $P3_6$ to output.

[Port P1P3 control register] P1P3C

By setting the port P1P3 control register (address 0017_{16}), a CMOS input level or a TTL input level can be selected for ports $P1_0$, $P1_2$, $P1_3$, $P3_6$, and $P3_7$ by program.

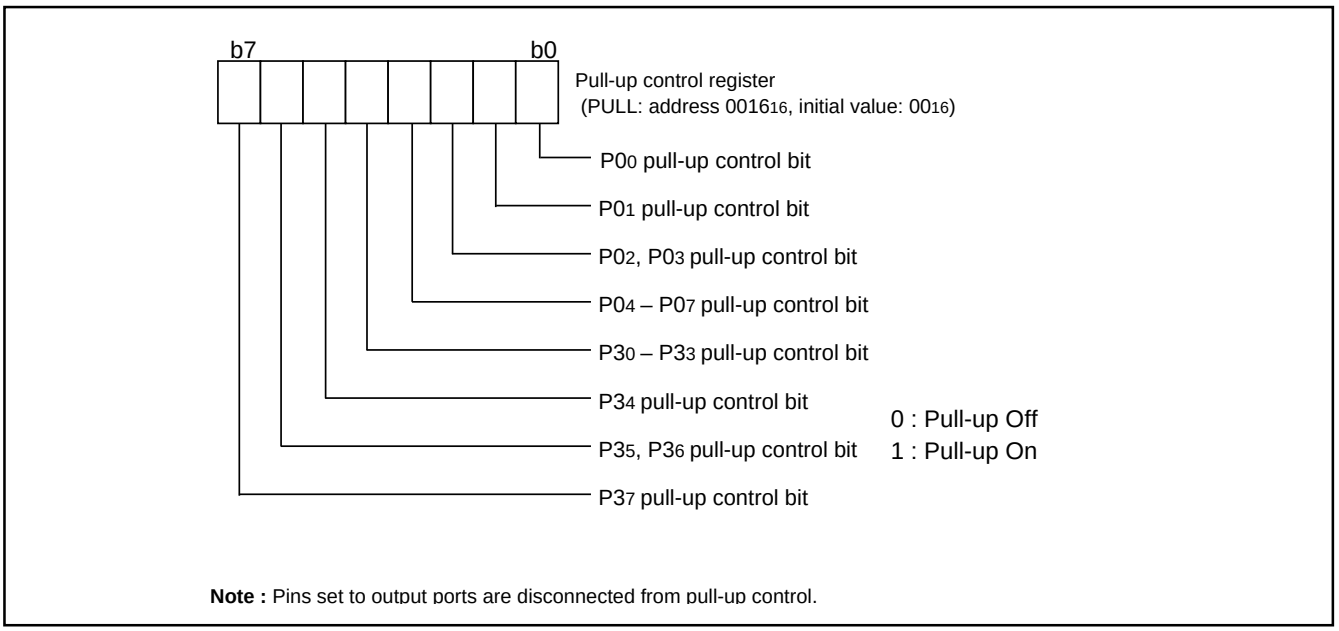


Fig. 15 Structure of pull-up control register

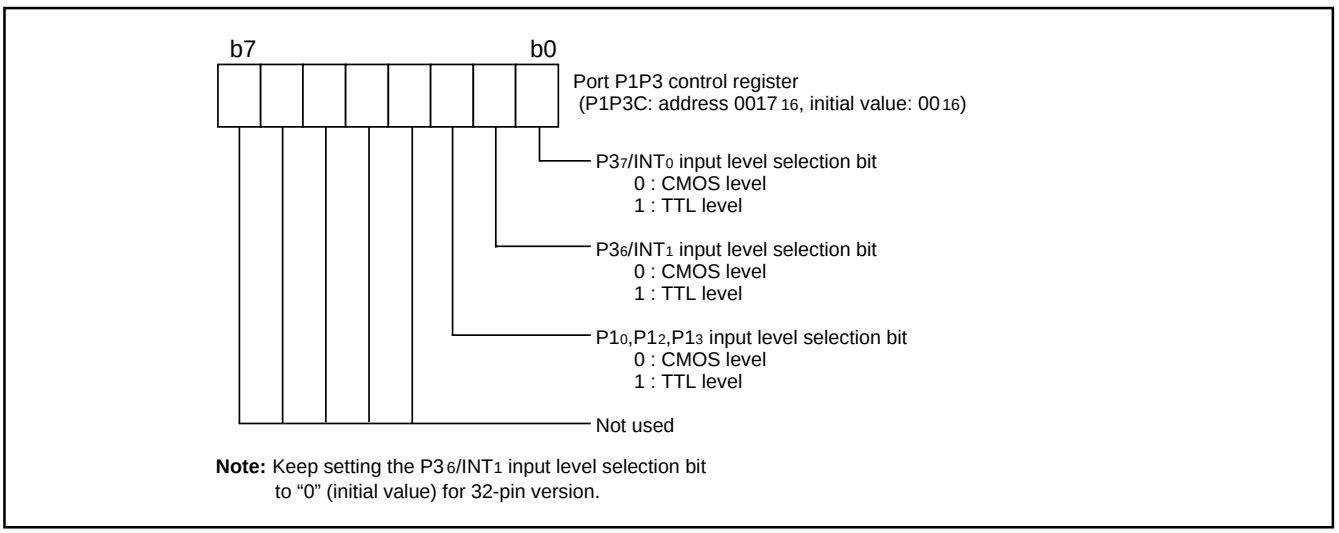
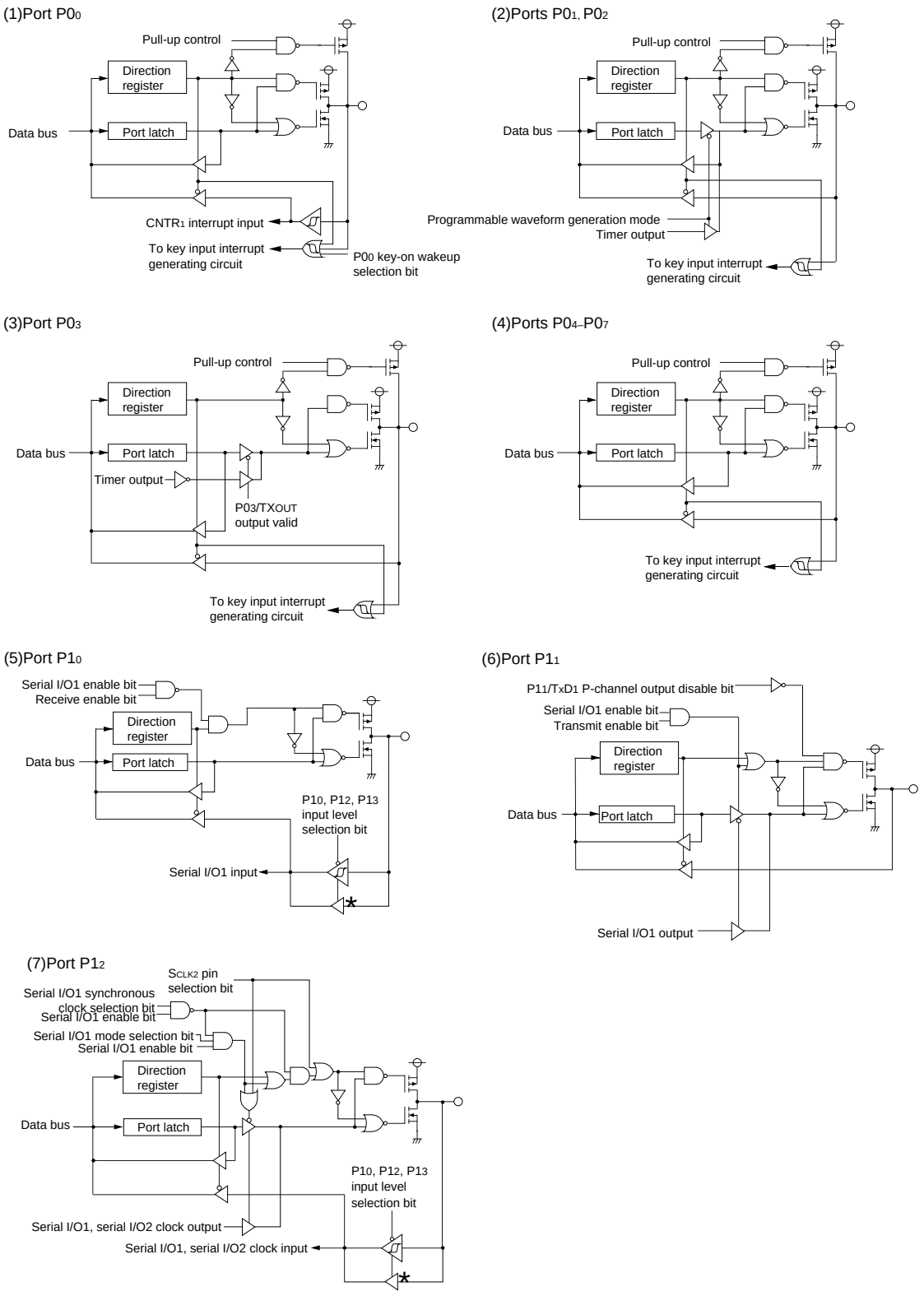


Fig. 16 Structure of port P1P3 control register

Table 5 I/O port function table

Pin	Name	Input/output	I/O format	Non-port function	Related SFRs	Diagram No.
$P0_0$/CNTR$_1$ $P0_1$/TY$_{OUT}$ $P0_2$/TZ$_{OUT}$ $P0_3$/TX$_{OUT}$ $P0_4$–$P0_7$	I/O port P0	I/O individual bits	•CMOS compatible input level •CMOS 3-state output (Note 1)	Key input interrupt Timer X function output Timer Y function output Timer Z function output Timer A function input	Pull-up control register Timer Y mode register Timer Z mode register Timer X mode register Timer Y,Z waveform output control register Timer A mode register	(1) (2) (3) (4)
$P1_0$/RxD$_1$ $P1_1$/TxD$_1$	I/O port P1			Serial I/O1 function input/output	Serial I/O1 control register	(5) (6)
$P1_2$/S$_{CLK1}$/S$_{CLK2}$ $P1_3$/S$_{RDY1}$/S$_{DATA2}$				Serial I/O2 function input/output	Serial I/O1 control register Serial I/O2 control register	(7) (8)
$P1_4$/CNTR$_0$				Timer X function input/output	Timer X mode register	(9)
$P2_0$/AN$_0$– $P2_7$/AN$_7$	I/O port P2 (Note 2)			A-D conversion input	A-D control register	(10)
$P3_0$–$P3_5$	I/O port P3					(11)
$P3_6$/INT$_1$ $P3_7$/INT$_0$	(Note 3)			External interrupt input	Interrupt edge selection register	(12)

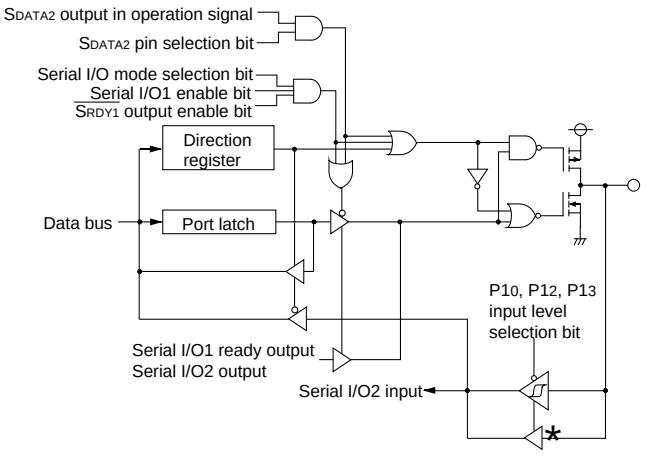
Notes 1: Ports $P1_0$, $P1_2$, $P1_3$, $P3_6$, and $P3_7$ are CMOS/TTL level.
2: $P2_6$/AN$_6$ and $P2_7$/AN$_7$ do not exist for the 32-pin version.
3: $P3_5$ and $P3_6$/INT$_1$ do not exist for the 32-pin version.



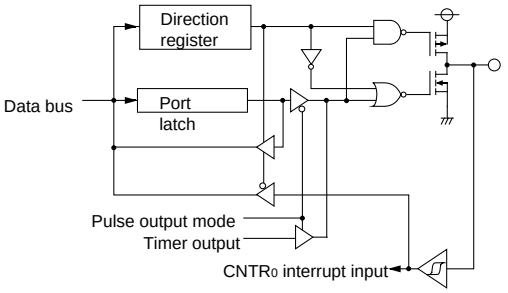
* P10, P12, P13, P36, and P37 input level are switched to the CMOS/TTL level by the port P1P3 control register. When the TTL level is selected, there is no hysteresis characteristics.

Fig. 17 Block diagram of ports (1)

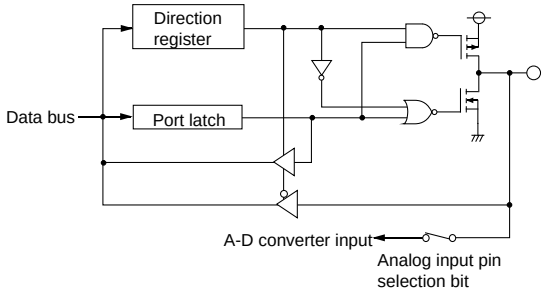
(8) Port P13



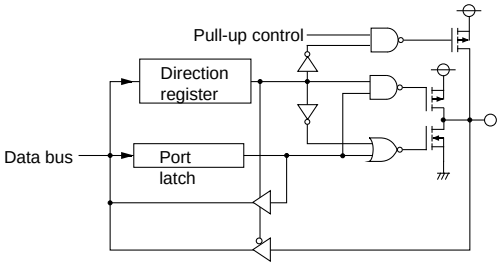
(9) Port P14



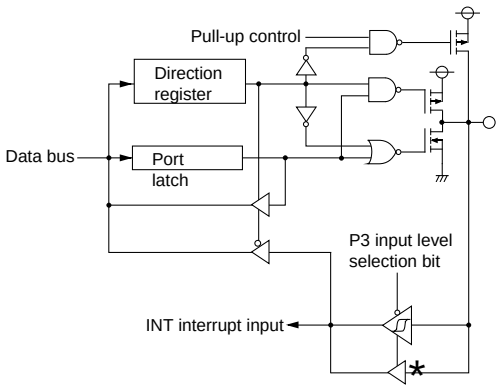
(10) Ports P20–P27



(11) Ports P30–P35



(12) Ports P36, P37



* P10, P12, P13, P36, and P37 input level are switched to the CMOS/TTL level by the port P1P3 control register. When the TTL level is selected, there is no hysteresis characteristics.

Fig. 18 Block diagram of ports (2)

Interrupts

Interrupts occur by 15 different sources : 5 external sources, 9 internal sources and 1 software source.

Interrupt control

All interrupts except the BRK instruction interrupt have an interrupt request bit and an interrupt enable bit, and they are controlled by the interrupt disable flag. When the interrupt enable bit and the interrupt request bit are set to "1" and the interrupt disable flag is set to "0", an interrupt is accepted.

The interrupt request bit can be cleared by program but not be set.

The interrupt enable bit can be set and cleared by program.

The reset and BRK instruction interrupt can never be disabled with any flag or bit. All interrupts except these are disabled when the interrupt disable flag is set.

When several interrupts occur at the same time, the interrupts are received according to priority.

Interrupt operation

Upon acceptance of an interrupt the following operations are automatically performed:

1. The processing being executed is stopped.
2. The contents of the program counter and processor status register are automatically pushed onto the stack.
3. The interrupt disable flag is set and the corresponding interrupt request bit is cleared.
4. Concurrently with the push operation, the interrupt destination address is read from the vector table into the program counter.

■ Notes on use

When setting the followings, the interrupt request bit may be set to "1".

•When switching external interrupt active edge

Related register: Interrupt edge selection register (address $003A_{16}$)

Timer X mode register (address $2B_{16}$)

Timer A mode register (address $1D_{16}$)

When not requiring the interrupt occurrence synchronized with these setting, take the following sequence.

① Set the corresponding interrupt enable bit to "0" (disabled).

② Set the interrupt edge select bit (active edge switch bit).

③ Set the corresponding interrupt request bit to "0" after 1 or more instructions have been executed.

④ Set the corresponding interrupt enable bit to "1" (enabled).

Table 6 Interrupt vector address and priority

Interrupt source	Priority	Vector addresses (Note 1)		Interrupt request generating conditions	Remarks
		High-order	Low-order		
Reset (Note 2)	1	$FFFD_{16}$	$FFFC_{16}$	At reset input	Non-maskable
Serial I/O1 receive	2	$FFFB_{16}$	$FFFA_{16}$	At completion of serial I/O1 data receive	Valid only when serial I/O1 is selected
Serial I/O1 transmit	3	$FFF9_{16}$	$FFF8_{16}$	At completion of serial I/O1 transmit shift or when transmit buffer is empty	Valid only when serial I/O1 is selected
INT_0	4	$FFF7_{16}$	$FFF6_{16}$	At detection of either rising or falling edge of INT_0 input	External interrupt (active edge selectable)
INT_1 (Note 3)	5	$FFF5_{16}$	$FFF4_{16}$	At detection of either rising or falling edge of INT_1 input	External interrupt (active edge selectable)
Key-on wake-up	6	$FFF3_{16}$	$FFF2_{16}$	At falling of conjunction of input logical level for port P0 (at input)	External interrupt (valid at falling)
$CNTR_0$	7	$FFF1_{16}$	$FFF0_{16}$	At detection of either rising or falling edge of $CNTR_0$ input	External interrupt (active edge selectable)
$CNTR_1$	8	$FFEF_{16}$	$FFEE_{16}$	At detection of either rising or falling edge of $CNTR_1$ input	External interrupt (active edge selectable)
Timer X	9	$FFED_{16}$	$FFEC_{16}$	At timer X underflow	
Timer Y	10	$FFEB_{16}$	$FFEA_{16}$	At timer Y underflow	
Timer Z	11	$FFE9_{16}$	$FFE8_{16}$	At timer Z underflow	
Timer A	12	$FFE7_{16}$	$FFE6_{16}$	At timer A underflow	
Serial I/O2	13	$FFE5_{16}$	$FFE4_{16}$	At completion of transmit/receive shift	
A-D conversion	14	$FFE3_{16}$	$FFE2_{16}$	At completion of A-D conversion	
Timer 1	15	$FFE1_{16}$	$FFE0_{16}$	At timer 1 underflow	STP release timer underflow
Reserved area	16	$FFDF_{16}$	$FFDE_{16}$	Not available	
BRK instruction	17	$FFDD_{16}$	$FFDC_{16}$	At BRK instruction execution	Non-maskable software interrupt

Note 1: Vector addressed contain internal jump destination addresses.

2: Reset function in the same way as an interrupt with the highest priority.

3: It is an interrupt which can use only for 36 pin version.

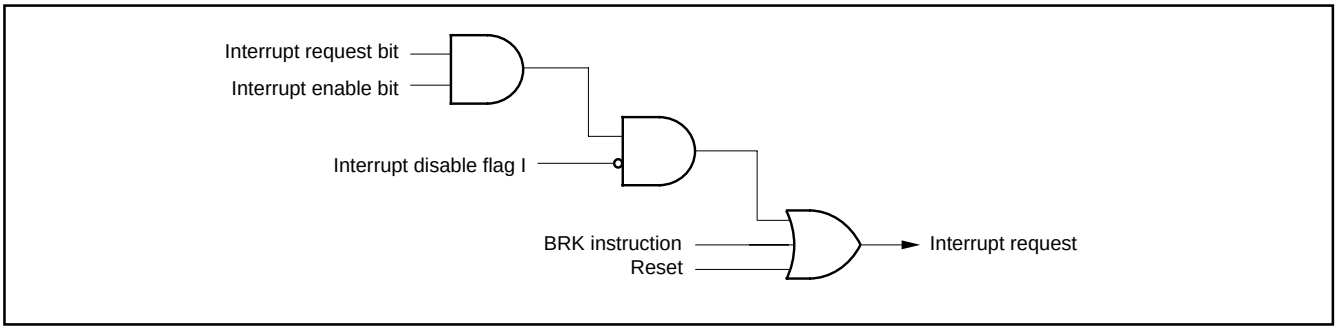


Fig. 19 Interrupt control

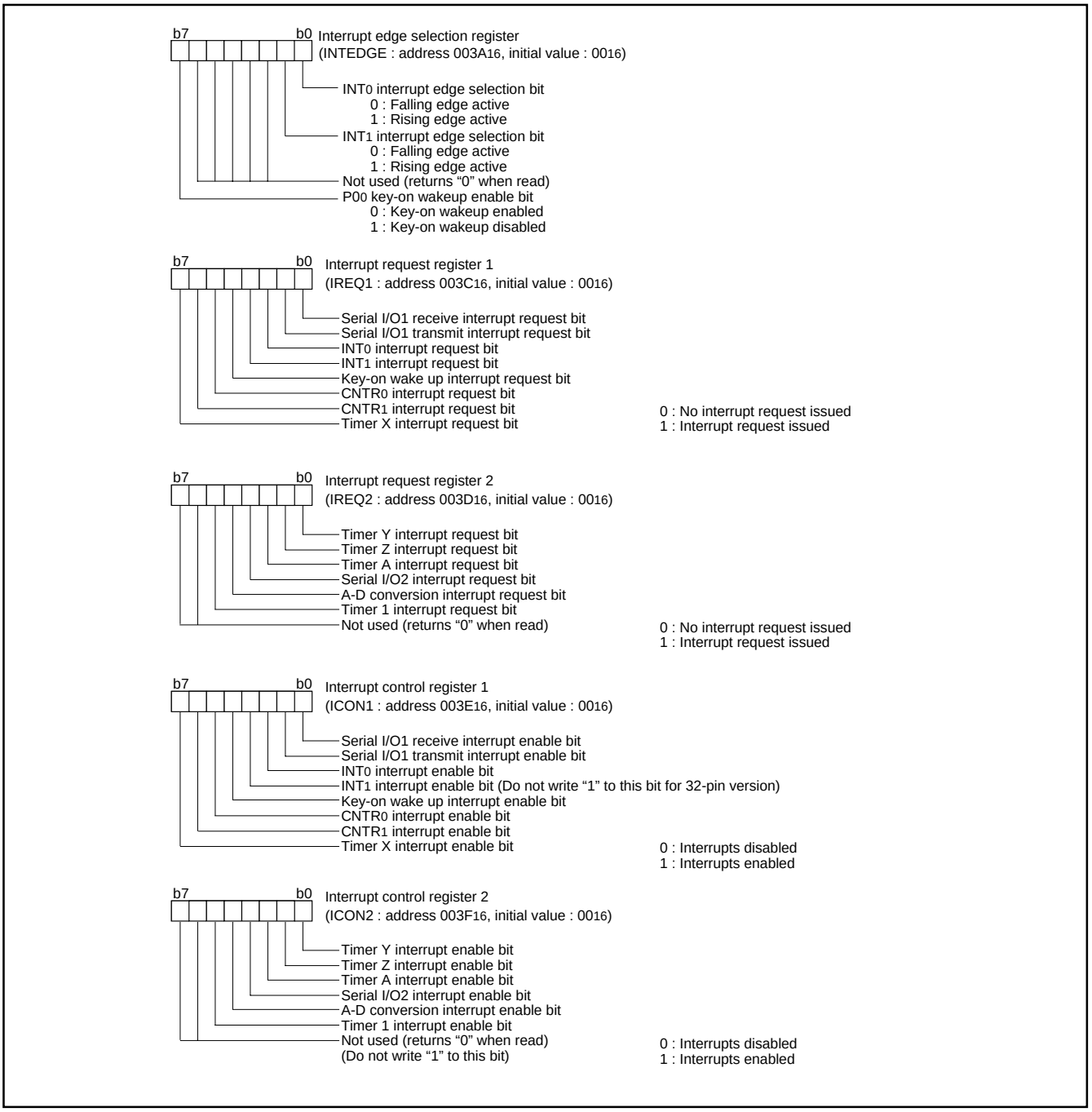


Fig. 20 Structure of Interrupt-related registers

Key Input Interrupt (Key-On Wake-Up)

A key-on wake-up interrupt request is generated by applying "L" level to any pin of port P0 that has been set to input mode.
In other words, it is generated when the AND of input level goes from "1" to "0". An example of using a key input interrupt is shown in Figure 21, where an interrupt request is generated by pressing one of the keys provided as an active-low key matrix which uses ports P00 to P03 as input ports.

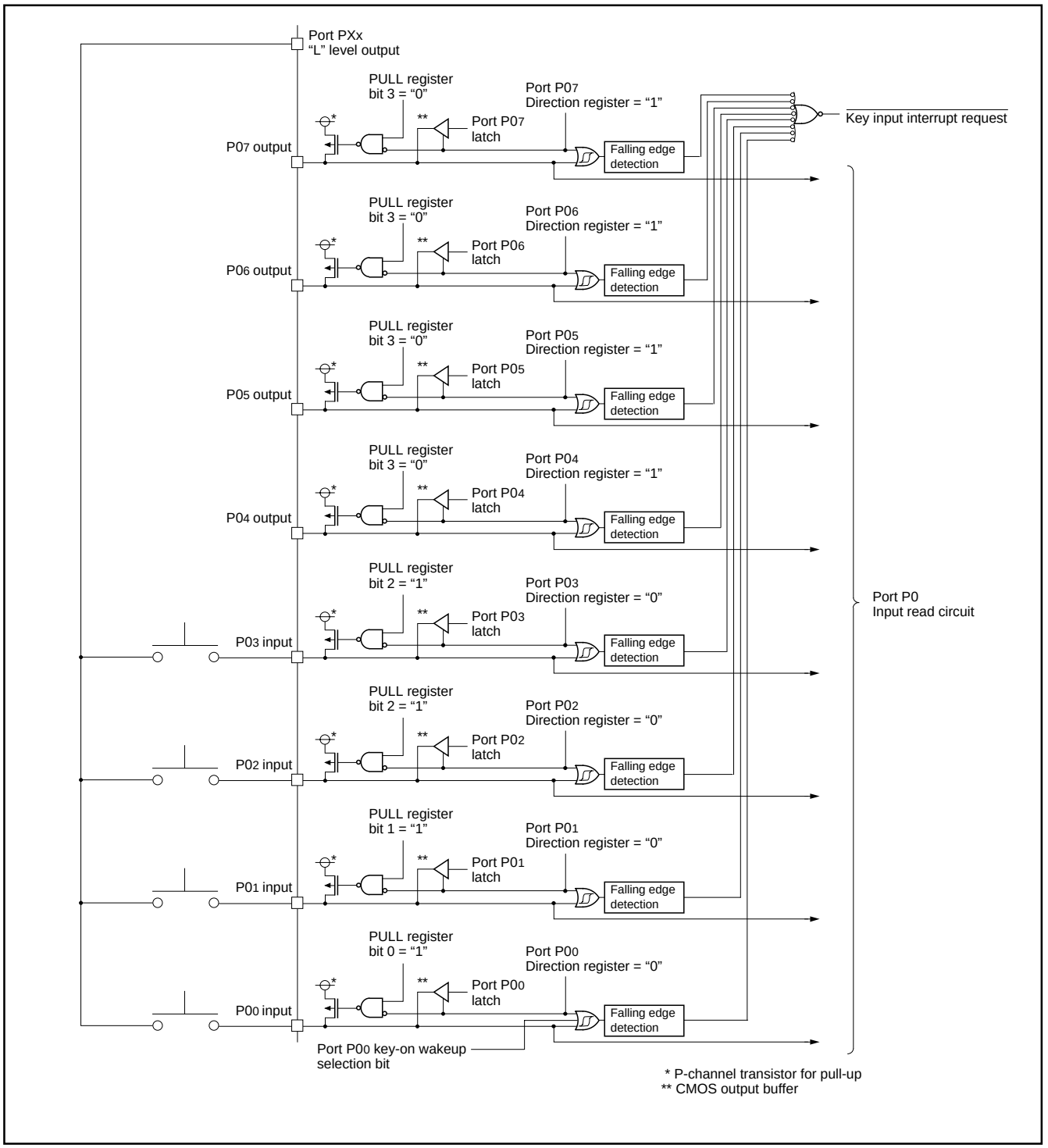


Fig. 21 Connection example when using key input interrupt and port P0 block diagram

Timers

The 7540 Group has 5 timers: timer 1, timer A, timer X, timer Y and timer Z.

The division ratio of every timer and prescaler is 1/(n+1) provided that the value of the timer latch or prescaler is n.

All the timers are down count timers. When a timer reaches "0", an underflow occurs at the next count pulse, and the corresponding timer latch is reloaded into the timer. When a timer underflows, the interrupt request bit corresponding to each timer is set to "1".

●Timer 1

Timer 1 is an 8-bit timer and counts the prescaler output.

When Timer 1 underflows, the timer 1 interrupt request bit is set to "1".

Prescaler 1 is an 8-bit prescaler and counts the signal which is the oscillation frequency divided by 16.

Prescaler 1 and Timer 1 have the prescaler 1 latch and the timer 1 latch to retain the reload value, respectively. The value of prescaler 1 latch is set to Prescaler 1 when Prescaler 1 underflows.The value of timer 1 latch is set to Timer 1 when Timer 1 underflows.

When writing to Prescaler 1 (PRE1) is executed, the value is written to both the prescaler 1 latch and Prescaler 1.

When writing to Timer 1 (T1) is executed, the value is written to both the timer 1 latch and Timer 1.

When reading from Prescaler 1 (PRE1) and Timer 1 (T1) is executed, each count value is read out.

Timer 1 always operates in the timer mode.

Prescaler 1 counts the signal which is the oscillation frequency divided by 16. Each time the count clock is input, the contents of Prescaler 1 is decremented by 1. When the contents of Prescaler 1 reach "00_{16}", an underflow occurs at the next count clock, and the prescaler 1 latch is reloaded into Prescaler 1 and count continues. The division ratio of Prescaler 1 is 1/(n+1) provided that the value of Prescaler 1 is n.

The contents of Timer 1 is decremented by 1 each time the underflow signal of Prescaler 1 is input. When the contents of Timer 1 reach "00_{16}", an underflow occurs at the next count clock, and the timer 1 latch is reloaded into Timer 1 and count continues. The division ratio of Timer 1 is 1/(m+1) provided that the value of Timer 1 is m. Accordingly, the division ratio of Prescaler 1 and Timer 1 is 1/((n+1)×(m+1)) provided that the value of Prescaler 1 is n and the value of Timer 1 is m.

Timer 1 cannot stop counting by software.

●Timer A

Timer A is a 16-bit timer and counts the signal which is the oscillation frequency divided by 16. When Timer A underflows, the timer A interrupt request bit is set to "1".

Timer A consists of the low-order of Timer A (TAL) and the high-order of Timer A (TAH).

Timer A has the timer A latch to retain the reload value. The value of timer A latch is set to Timer A at the timing shown below.

• When Timer A undeflows.

• When an active edge is input from $CNTR_1$ pin (valid only when period measurement mode and pulse width HL continuously measurement mode).

When writing to both the low-order of Timer A (TAL) and the high-order of Timer A (TAH) is executed, the value is written to both the timer A latch and Timer A.

When reading from the low-order of Timer A (TAL) and the high-order of Timer A (TAH) is executed, the following values are read out according to the operating mode.

• In timer mode, event counter mode:

The count value of Timer A is read out.

• In period measurement mode, pulse width HL continuously measurement mode:

The measured value is read out.

Be sure to write to/read out the low-order of Timer A (TAL) and the high-order of Timer A (TAH) in the following order;

Read

Read the high-order of Timer A (TAH) first, and the low-order of Timer A (TAL) next and be sure to read out both TAH and TAL.

Write

Write to the low-order of Timer A (TAL) first, and the high-order of Timer A (TAH) next and be sure to write to both TAL and TAH.

Timer A can be selected in one of 4 operating modes by setting the timer A mode register.

(1) Timer mode

Timer A counts the oscillation frequency divided by 16. Each time the count clock is input, the contents of Timer A is decremented by 1. When the contents of Timer A reach "0000_{16}", an underflow occurs at the next count clock, and the timer A latch is reloaded into Timer A. The division ratio of Timer A is 1/(n+1) provided that the value of Timer A is n.

(2) Period measurement mode

In the period measurement mode, the pulse period input from the $P0_0$/$CNTR_1$ pin is measured.

$CNTR_1$ interrupt request is generated at rising/falling edge of $CNTR_1$ pin input singal. Simultaneousuly, the value in the timer A latch is reloaded inTimer A and count continues. The active edge of $CNTR_1$ pin input signal can be selected from rising or falling by the $CNTR_1$ active edge switch bit .The count value when trigger input from $CNTR_1$ pin is accepted is retained until Timer A is read once.

(3) Event counter mode

Timer A counts signals input from the P00/CNTR1 pin.

Except for this, the operation in event counter mode is the same as in timer mode.

The active edge of CNTR1 pin input signal can be selected from rising or falling by the CNTR1 active edge switch bit .

(4) Pulse width HL continuously measurement mode

In the pulse width HL continuously measurement mode, the pulse width ("H" and "L" levels) input to the P00/CNTR1 pin is measured. CNTR1 interrupt request is generated at both rising and falling edges of CNTR1 pin input signal. Except for this, the operation in pulse width HL continuously measurement mode is the same as in period measurement mode.

The count value when trigger input from the CNTR1 pin is accepted is retained until Timer A is read once.

Timer A can stop counting by setting "1" to the timer A count stop bit in any mode.

Also, when Timer A underflows, the timer A interrupt request bit is set to "1".

Note on Timer A is described below;

■ Note on Timer A

CNTR1 interrupt active edge selection

CNTR1 interrupt active edge depends on the CNTR1 active edge switch bit.

When this bit is "0", the CNTR1 interrupt request bit is set to "1" at the falling edge of the CNTR1 pin input signal. When this bit is "1", the CNTR1 interrupt request bit is set to "1" at the rising edge of the CNTR1 pin input signal.

However, in the pulse width HL continuously measurement mode, CNTR1 interrupt request is generated at both rising and falling edges of CNTR1 pin input signal regardless of the setting of CNTR1 active edge switch bit.

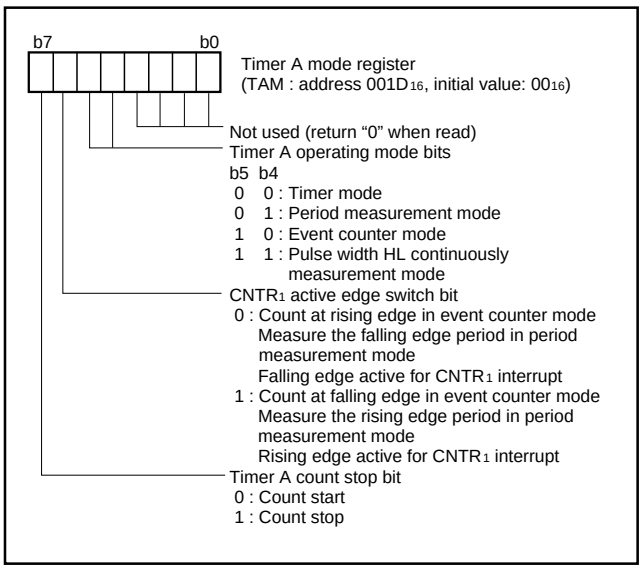


Fig. 22 Structure of timer A mode register

●Timer X

Timer X is an 8-bit timer and counts the prescaler X output.

When Timer X underflows, the timer X interrupt request bit is set to "1".

Prescaler X is an 8-bit prescaler and counts the signal selected by the timer X count source selection bit.

Prescaler X and Timer X have the prescaler X latch and the timer X latch to retain the reload value, respectively. The value of prescaler X latch is set to Prescaler X when Prescaler X underflows.The value of timer X latch is set to Timer X when Timer X underflows.

When writing to Prescaler X (PREX) is executed, the value is written to both the prescaler X latch and Prescaler X.

When writing to Timer X (TX) is executed, the value is written to both the timer X latch and Timer X.

When reading from Prescaler X (PREX) and Timer X (TX) is executed, each count value is read out.

Timer X can can be selected in one of 4 operating modes by setting the timer X operating mode bits of the timer X mode register.

(1) Timer mode

Prescaler X counts the count source selected by the timer X count source selection bits. Each time the count clock is input, the contents of Prescaler X is decremented by 1. When the contents of Prescaler X reach "0016", an underflow occurs at the next count clock, and the prescaler X latch is reloaded into Prescaler X and count continues. The division ratio of Prescaler X is 1/(n+1) provided that the value of Prescaler X is n.

The contents of Timer X is decremented by 1 each time the underflow signal of Prescaler X is input. When the contents of Timer X reach "0016", an underflow occurs at the next count clock, and the timer X latch is reloaded into Timer X and count continues. The division ratio of Timer X is 1/(m+1) provided that the value of Timer X is m. Accordingly, the division ratio of Prescaler X and Timer X is 1/((n+1)×(m+1)) provided that the value of Prescaler X is n and the value of Timer X is m.

(2) Pulse output mode

In the pulse output mode, the waveform whose polarity is inverted each time timer X underflows is output from the CNTR0 pin.

The output level of CNTR0 pin can be selected by the CNTR0 active edge switch bit. When the CNTR0 active edge switch bit is "0", the output of CNTR0 pin is started at "H" level. When this bit is "1", the output is started at "L" level.

Also, the inverted waveform of pulse output from CNTR0 pin can be output from TXOUT pin by setting "1" to the P03/TXOUT output valid bit.

When using a timer in this mode, set the port P14 and P03 direction registers to output mode.

(3) Event counter mode

The timer A counts signals input from the P14/CNTR0 pin.

Except for this, the operation in event counter mode is the same as in timer mode.

The active edge of CNTR0 pin input signal can be selected from rising or falling by the CNTR0 active edge switch bit .

(4) Pulse width measurement mode

In the pulse width measurement mode, the pulse width of the signal input to P14/CNTR0 pin is measured.

The operation of Timer X can be controlled by the level of the signal input from the CNTR0 pin.

When the CNTR0 active edge switch bit is “0”, the signal selected by the timer X count source selection bit is counted while the input signal level of CNTR0 pin is “H”. The count is stopped while the pin is “L”. Also, when the CNTR0 active edge switch bit is “1”, the signal selected by the timer X count source selection bit is counted while the input signal level of CNTR0 pin is “L”. The count is stopped while the pin is “H”.

Timer X can stop counting by setting “1” to the timer X count stop bit in any mode.

Also, when Timer X underflows, the timer X interrupt request bit is set to “1”.

Note on Timer X is described below;

■ Note on Timer X

CNTR0 interrupt active edge selection

CNTR0 interrupt active edge depends on the CNTR0 active edge switch bit.

When this bit is “0”, the CNTR0 interrupt request bit is set to “1” at the falling edge of CNTR0 pin input signal. When this bit is “1”, the CNTR0 interrupt request bit is set to “1” at the rising edge of CNTR0 pin input signal.

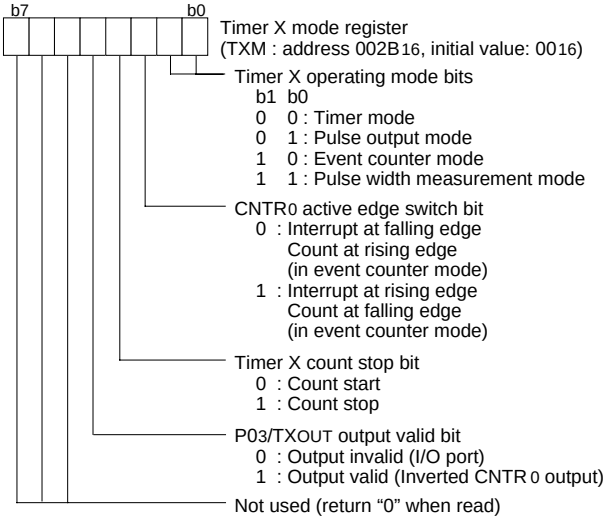
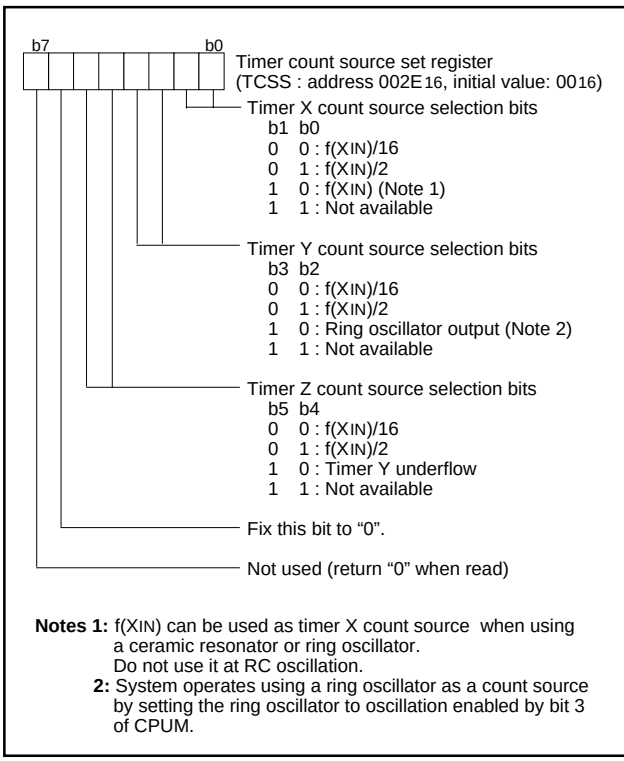


Fig. 23 Structure of timer X mode register



Notes 1: f(XIN) can be used as timer X count source when using a ceramic resonator or ring oscillator. Do not use it at RC oscillation.
2: System operates using a ring oscillator as a count source by setting the ring oscillator to oscillation enabled by bit 3 of CPUM.

Fig. 24 Timer count source set register

●Timer Y

Timer Y is an 8-bit timer and counts the prescaler Y output.
When Timer Y underflows, the timer Y interrupt request bit is set to "1".
Prescaler Y is an 8-bit prescaler and counts the signal selected by the timer Y count source selection bit.
Prescaler Y has the prescaler Y latch to retain the reload value. Timer Y has the timer Y primary latch and timer Y secondary latch to retain the reload value.
The value of prescaler Y latch is set to Prescaler Y when Prescaler Y underflows.The value of timer Y primary latch or timer Y secondary latch are set to Timer Y when Timer Y underflows.
As for the value to transfer to Timer Y, either of timer Y primary or timer Y secondary is selected depending on the timer Y operating mode.

When writing to Prescaler Y (PREY), timer Y primary (TYP) or timer Y secondary (TYS) is executed, writing to "latch only" or "latch and prescaler (timer)" can be selected by the setting value of the timer Y write control bit. Be sure to set the timer Y write control bit because there are some notes according to the operating mode.
When reading from Prescaler Y (PREY) is executed, the count value of Prescaler Y is read out. When reading from timer Y primary (TYP) is executed, the count value of Timer Y is read out. The count value of Timer Y can be read out by reading from the timer Y primary (TYP) even when the value of timer Y primary latch or timer Y secondary latch is counted. When reading the timer Y secondary (TYS) is executed, the undefined value is read out.

Timer Y can be selected in one of 2 operating modes by setting the timer Y operating mode bits of the timer Y, Z mode register.

(1) Timer mode

Prescaler Y counts the count source selected by the timer Y count source selection bits. Each time the count clock is input, the contents of Prescaler Y is decremented by 1. When the contents of Prescaler Y reach "00_{16}", an underflow occurs at the next count clock, and the prescaler Y latch is reloaded into Prescaler Y. The division ratio of Prescaler Y is 1/(n+1) provided that the value of Prescaler Y is n.
The contents of Timer Y is decremented by 1 each time the underflow signal of Prescaler Y is input. When the contents of Timer Y reach "00_{16}", an underflow occurs at the next count clock, and the timer Y primary latch is reloaded into Timer Y and count continues. (In the timer mode, the contents of timer Y primary latch is counted. Timer Y secondary latch is not used in this mode.)
The division ratio of Timer Y is 1/(m+1) provided that the value of Timer Y is m. Accordingly, the division ratio of Prescaler Y and Timer Y is 1/((n+1)×(m+1)) provided that the value of Prescaler Y is n and the value of Timer Y is m.
In the timer mode, writing to "latch only" or "latches and Prescaler Y and timer Y primary" can be selected by the setting value of the timer Y write control bit.

(2) Programmable waveform generation mode

In the programmable waveform generation mode, timer counts the setting value of timer Y primary and the setting value of timer Y secondary alternately, the waveform inverted each time Timer Y underflows is output from TYOUT pin.
When using this mode, be sure to set "1" to the timer Y write control bit to select "write to latch only". Also, set the port $P0_1$ direction registers to output mode.
The active edge of output waveform is set by the timer Y output level latch (b5) of the timer Y, Z waveform output control register (PUM). When "0" is set to b5 of PUM, "H" interval by the setting value of TYP or "L" interval by the setting value of TYS is output alternately. When "1" is set to b5 of PUM, "L" interval by the setting value of TYP or "H" interval by the setting value of TYS is output alternately.
Also, in this mode, the primary interval and the secondary interval of the output waveform can be extended respectively for 0.5 cycle of timer count source clock by setting the timer Y primary waveform extension control bit (b2) and the timer Y secondary waveform extension control bit (b3) of PUM to "1". As a result, the waveforms of more accurate resolution can be output.
When b2 and b3 of PUM are used, the frequency and duty of the output waveform are as follows;

Waveform frequency:

$$FYOUT=\frac{2\times TMYCL}{2\times(TYP+1)+2\times(TYS+1)+(EXPYP+EXPYS)}$$

Duty:

$$DYOUT=\frac{2\times(TYP+1)+EXPYP}{(2\times(TYP+1)+EXPYP)+(2\times(TYS+1)+EXPYS)}$$

TMYCL: Timer Y count source (frequency)
TYP: Timer Y primary (8bit)
TYS: Timer Y secondary (8bit)
EXPYP: Timer Y primary waveform extension control bit (1bit)
EXPYS: Timer Y secondary waveform extension control bit (1bit)

In the programmable waveform generation mode, when values of the TYP, TYS, EXPYP and EXPYS are changed, the output waveform is changed at the beginning (timer Y primary waveform interval) of waveform period.
When the count values are changed, set values to the TYS, EXPYP and EXPYS first. After then, set the value to TYP. The values are set all at once at the beginning of the next waveform period when the value is set to TYP. (When writing at timer stop is executed, writing to TYP at last is required.)

Notes on programmable waveform generation mode is described below;

■ **Notes on programmable generation waveform mode**

• Count set value

In the programmable waveform generation mode, values of TYS, EXPYP, and EXPYS are valid by writing to TYP because the setting to them is executed all at once by writing to TYP. Even when changing TYP is not required, write the same value again.

• Write timing to TYP

In the programmable waveform generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TYP and the timing of timer underflow during the secondary interval simultanesously.

• Usage of waveform extension function

The waveform extension function by the timer Y waveform extension control bit can be used only when "00_{16}" is set to Prescaler Y. When the value other than "00_{16}" is set to Prescaler Y, be sure to set "0" to EXPYP and EXPYS.

• Timer Y write mode

When using this mode, be sure to set "1" to the timer Y write control bit to select "write to latch only".

Timer Y can stop counting by setting "1" to the timer Y count stop bit in any mode.
Also, when Timer Y underflows, the timer Y interrupt request bit is set to "1".

Timer Y reloads the value of latch when counting is stopped by the timer Y count stop bit. (When timer is read out while timer is stopped, the value of latch is read. The value of timer can be read out only while timer is operating.)

●Timer Z

Timer Z is an 8-bit timer and counts the prescaler Z output.
When Timer Z underflows, the timer Z interrupt request bit is set to "1".
Prescaler Z is an 8-bit prescaler and counts the signal selected by the timer Z count source selection bit.
Prescaler Z has the prescaler Z latch to retain the reload value. Timer Z has the timer Z primary latch and timer Z secondary latch to retain the reload value.
The value of prescaler Z latch is set to Prescaler Z when Prescaler Z underflows.The value of timer Z primary latch or timer Z secondary latch are set to Timer Z when Timer Z underflows.
As for the value to transfer to Timer Z, either of timer Z primary or timer Z secondary is selected depending on the timer Z operating mode.

When writing to Prescaler Z (PREZ), timer Z primary (TZP) or timer Z secondary (TZS) is executed, writing to "latch only" or "latches and Prescaler Z and Timer Z" can be selected by the setting value of the timer Z write control bit. Be sure to set the write control bit because there are some notes according to the operating mode.
When reading from Prescaler Z (PREZ) is executed, the count value of Prescaler Z is read out. When reading from timer Z primary (TZP) is executed, the count value of Timer Z is read out. The count value of Timer Z can be read out by reading from the timer Z primary (TZP) even when the value of timer Z primary latch or timer Z secondary latch is counted. When reading the timer Z secondary (TZS) is executed, the undefined value is read out.

Timer Z can be selected in one of 4 operating modes by setting the timer Z operating mode bits of the timer Y, Z mode register.

(1) Timer mode

Prescaler Z counts the count source selected by the timer Z count source selection bits. Each time the count clock is input, the contents of Prescaler Z is decremented by 1. When the contents of Prescaler Z reach "00_{16}", an underflow occurs at the next count clock, and the prescaler Z latch is reloaded into Prescaler Z. The division ratio of Prescaler Z is 1/(n+1) provided that the value of Prescaler Z is n.
The contents of Timer Z is decremented by 1 each time the underflow signal of Prescaler Z is input. When the contents of Timer Z reach "00_{16}", an underflow occurs at the next count clock, and the timer Z primary latch is reloaded into Timer Z and count continues. (In the timer mode, the contents of timer Z primary latch is counted. Timer Z secondary latch is not used in this mode.)
The division ratio of Timer Z is 1/(m+1) provided that the value of Timer Z is m. Accordingly, the division ratio of Prescaler Z and Timer Z is 1/((n+1)×(m+1)) provided that the value of Prescaler Z is n and the value of Timer Z is m.
In the timer mode, writing to "latch only" or "latches and Prescaler Z and timer Z primary" can be selected by the setting value of the timer Z write control bit.

(2) Programmable waveform generation mode

In the programmable waveform generation mode, timer counts the setting value of timer Z primary and the setting value of timer Z secondary alternately, the waveform inverted each time Timer Z underflows is output from TZ_{OUT} pin.
When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only". Also, set the port $P0_2$ direction registers to output mode.
The active edge of output waveform is set by the timer Z output level latch (b4) of the timer Y, Z waveform output control register (PUM). When "0" is set to b4 of PUM, "H" interval by the setting value of TZP or "L" interval by the setting value of TZS is output alternately. When "1" is set to b4 of PUM, "L" interval by the setting value of TZP or "H" interval by the setting value of TZS is output alternately.
Also, in this mode, the primary interval and the secondary interval of the output waveform can be extended respectively for 0.5 cycle of timer count source clock by setting the timer Z primary waveform extension control bit (b0) and the timer Z secondary waveform extension control bit (b1) of PUM to "1". As a result, the waveforms of more accurate resolution can be output.
When b0 and b1 of PUM are used, the frequency and duty of the output waveform are as follows;

Waveform frequency:

$$FZOUT=\frac{2\times TMZCL}{2\times(TZP+1)+2\times(TZS+1)+(EXPZP+EXPZS)}$$

Duty:

$$DZOUT=\frac{2\times(TZP+1)+EXPZP}{(2\times(TZP+1)+EXPZP)+(2\times(TZS+1)+EXPZS}$$

TMZCL: Timer Z count source (frequency)
TZP: Timer Z primary (8bit)
TZS: Timer Z secondary (8bit)
EXPZP: Timer Z primary waveform extension control bit (1bit)
EXPZS: Timer Z secondary waveform extension control bit (1bit)

In the programmable waveform generation mode, when values of the TZP, TZS, EXPZP and EXPZS are changed, the output waveform is changed at the beginning (timer Z primary waveform interval) of waveform period.
When the count values are changed, set values to the TZS, EXPZP and EXPZS first. After then, set the value to TZP. The values are set all at once at the beginning of the next waveform period when the value is set to TZP. (When writing at timer stop is executed, writing to TZP at last is required.)

Notes on the programmable waveform generation mode are described below;

■ **Notes on programmable waveform generation mode**

• Count set value

In the programmable waveform generation mode, values of TZS, EXPZP, and EXPZS are valid by writing to TZP because the setting to them is executed all at once by writing to TZP. Even when changing TZP is not required, write the same value again.

• Write timing to TZP

In the programmable waveform generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow during the secondary interval simultanesously.

• Usage of waveform extension function

The waveform extension function by the timer Z waveform extension control bit can be used only when "00_{16}" is set to Prescaler Z. When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the count source, the waveform extension function cannot be used.

• Timer Z write mode

When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only".

(3) Programmable one-shot generation mode

In the programmable one-shot generation mode, the one-shot pulse by the setting value of timer Z primary can be output from TZ_{OUT} pin by software or external trigger. When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only". Also, set the port $P0_2$ direction registers to output mode. In this mode, TZS is not used.

The active edge of output waveform is set by the timer Z output level latch (b5) of the timer Y, Z waveform output control register (PUM). When "0" is set to b5 of PUM, "H" pulse during the interval of the TZP setting value is output. When "1" is set to b5 of PUM, "L" pulse during the interval of the TZP setting value is output.

Also, in this mode, the interval of the one-shot pulse output can be extended for 0.5 cycle of timer count source clock by setting the timer Z primary waveform extension control bit (b2) of PUM to "1". As a result, the waveforms of more accurate resolution can be output.

In the programmable one-shot generation mode, the trigger by software or the external INT_0 pin can be accepted by writing "0" to the timer Z count stop bit after the count value is set. (At the time when "0" is written to the timer Z count stop bit, Timer Z stops.)

By writing "1" to the timer Z one-shot start bit, or by inputting the valid trigger to the INT_0 pin after the trigger to the INT_0 pin becomes valid by writing "1" to the INT_0 pin one-shot trigger control bit, Timer Z starts counting, at the same time, the output of TZ_{OUT} pin is inverted. When Timer Z underflows, the output of TZ_{OUT} pin is inverted again and Timer Z stops. When also the trigger of INT_0 pin is accepted, the contents of the one-shot start bit is changed to "1" by hardware.

The falling or rising can be selected as the edge of the valid trigger of INT_0 pin by the INT_0 pin one-shot trigger edge selection bit.

During the one-shot pulse output interval, the one-shot pulse output can be stopped forcibly by writing "0" to the timer Z one-shot start bit.

In the programmable one-shot generation mode, when the count values are changed, set value to the EXPZP first. After then, set the value to TZP. The values are set all at once at the beginning of the next one-shot pulse when the value is set to TZP. (When writing at timer stop is executed, writing to TZP at last is required.)

Notes on the programmable one-shot generation mode are described below;

■ **Notes on programmable one-shot generation mode**

• Count set value

In the programmable one-shot generation mode, the value of EXPZP becomes valid by writing to TZP. Even when changing TZP is not required, write the same value again.

• Write timing to TZP

In the programmable one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow simultanesously.

• Usage of waveform extension function

The waveform extension function by the timer Z waveform extension control bit can be used only when "00_{16}" is set to Prescaler Z. When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP. Also, when the timer Y underflow is selected as the count source, the waveform extension function cannot be used.

• Timer Z write mode

When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only".

(4) Programmable wait one-shot generation mode

In the programmable wait one-shot generation mode, the one-shot pulse by the setting value of timer Z secondary can be output from TZ_{OUT} pin by software or external trigger to INT_0 pin after the wait by the setting value of the timer Z primary. When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only". Also, set the port $P0_2$ direction registers to output mode.

The active edge of output waveform is set by the timer Z output level latch (b5) of the timer Y, Z waveform output control register (PUM). When "0" is set to b5 of PUM, after the wait during the interval of the TZP setting value, "H" pulse during the interval of the TZS setting value is output. When "1" is set to b5 of PUM, after the wait during the interval of the TZP setting value, "L" pulse during the interval of the TZS setting value is output.

Also, in this mode, the intervals of the wait and the one-shot pulse output can be extended for 0.5 cycle of timer count source clock by setting EXPZP and EXPZS of PUM to "1". As a result, the waveforms of more accurate resolution can be output.

In the programmable one-shot generation mode, the trigger by software or the external INT_0 pin can be accepted by writing "0" to the timer Z count stop bit after the count value is set. (At the time when "0" is written to the timer Z count stop bit, Timer Z stops.)

By writing "1" to the timer Z one-shot start bit, or by inputting the valid trigger to the INT_0 pin after the trigger to the INT_0 pin becomes valid by writing "1" to the INT_0 pin one-shot trigger control bit, Timer Z starts counting.

While Timer Z counts the TZP, the initial value of the TZ_{OUT} pin output is retained. When Timer Z underflows, the value of TZS is reloaded, at the same time, the output of TZ_{OUT} pin is inverted.

When Timer Z underflows, the output of TZ_{OUT} pin is inverted again and Timer Z stops. When also the trigger of INT_0 pin is accepted, the contents of the one-shot start bit is changed to "1" by hardware.

The falling or rising can be selected as the edge of the valid trigger of INT_0 pin by the INT_0 pin one-shot trigger edge selection bit.

During the wait interval and the one-shot pulse output interval, the one-shot pulse output can be stopped forcibly by writing "0" to the timer Z one-shot start bit.

In the programmable wait one-shot generation mode, when the count values are changed, set values to the TZS, EXPZP and EXPZS first. After then, set the value to TZP. The values are set all at once at the beginning of the next wait interval when the value is set to TZP. (When writing at timer stop is executed, writing to TZP at last is required.)

Notes on the programmable wait one-shot generation mode are described below;

■ **Notes on programmable wait one-shot generation mode**

• Count set value

In the programmable wait one-shot generation mode, values of TZS, EXPZP and EXPZS are valid by writing to TZP. Even when changing TZP is not required, write the same value again.

• Write timing to TZP

In the programmable wait one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow during the secondary interval simultanesously.

• Usage of waveform extension function

The waveform extension function by the timer Z waveform extension control bit can be used only when "00_{16}" is set to Prescaler Z. When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the count source, the waveform extension function cannot be used.

• Timer Z write mode

When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only".

Timer Z can stop counting by setting "1" to the timer Z count stop bit in any mode.

Also, when Timer Z underflows, the timer Z interrupt request bit is set to "1".

Timer Z reloads the value of latch when counting is stopped by the timer Z count stop bit. (When timer is read out while timer is stopped, the value of latch is read. The value of timer can be read out only while timer is operating.)

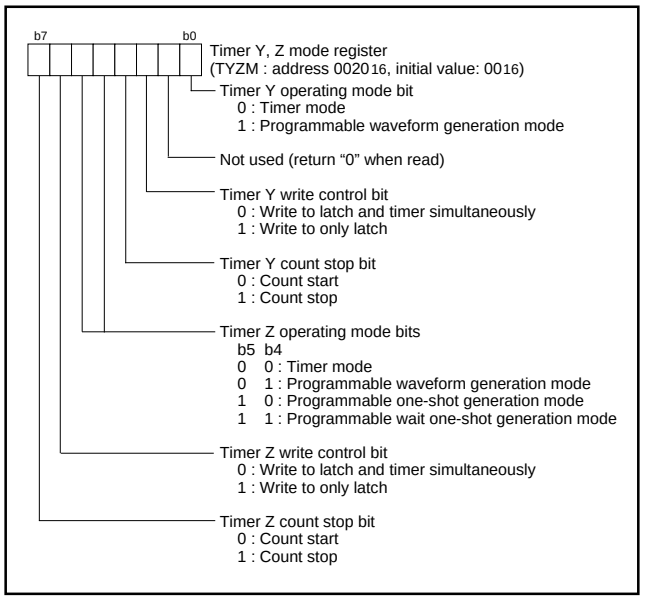


Fig. 25 Structure of timer Y, Z mode register

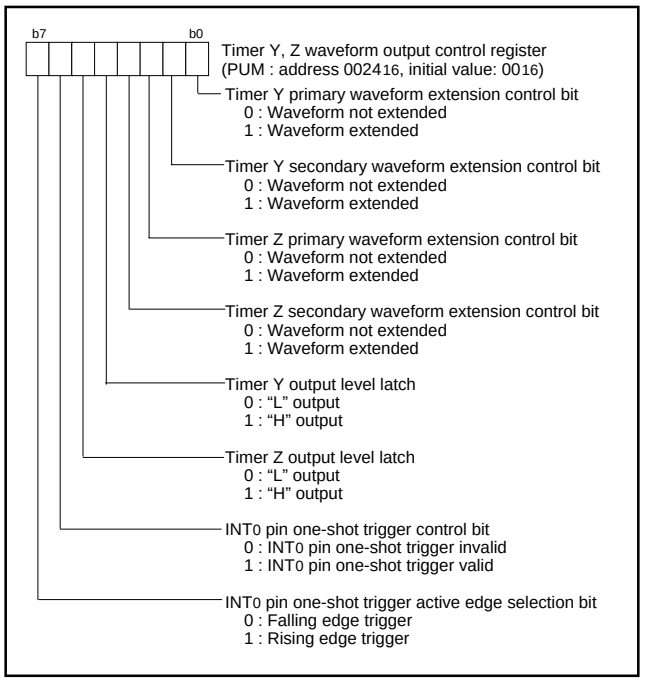


Fig. 26 Structure of timer Y, Z waveform output control register

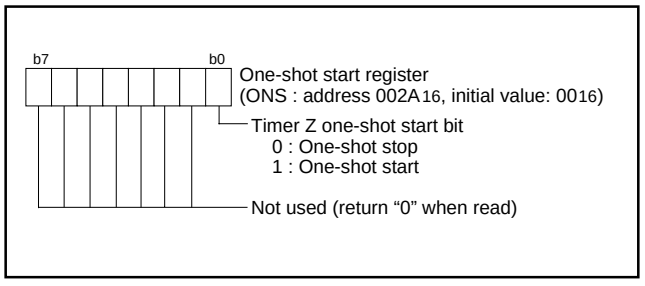


Fig. 27 Structure of one-shot start register

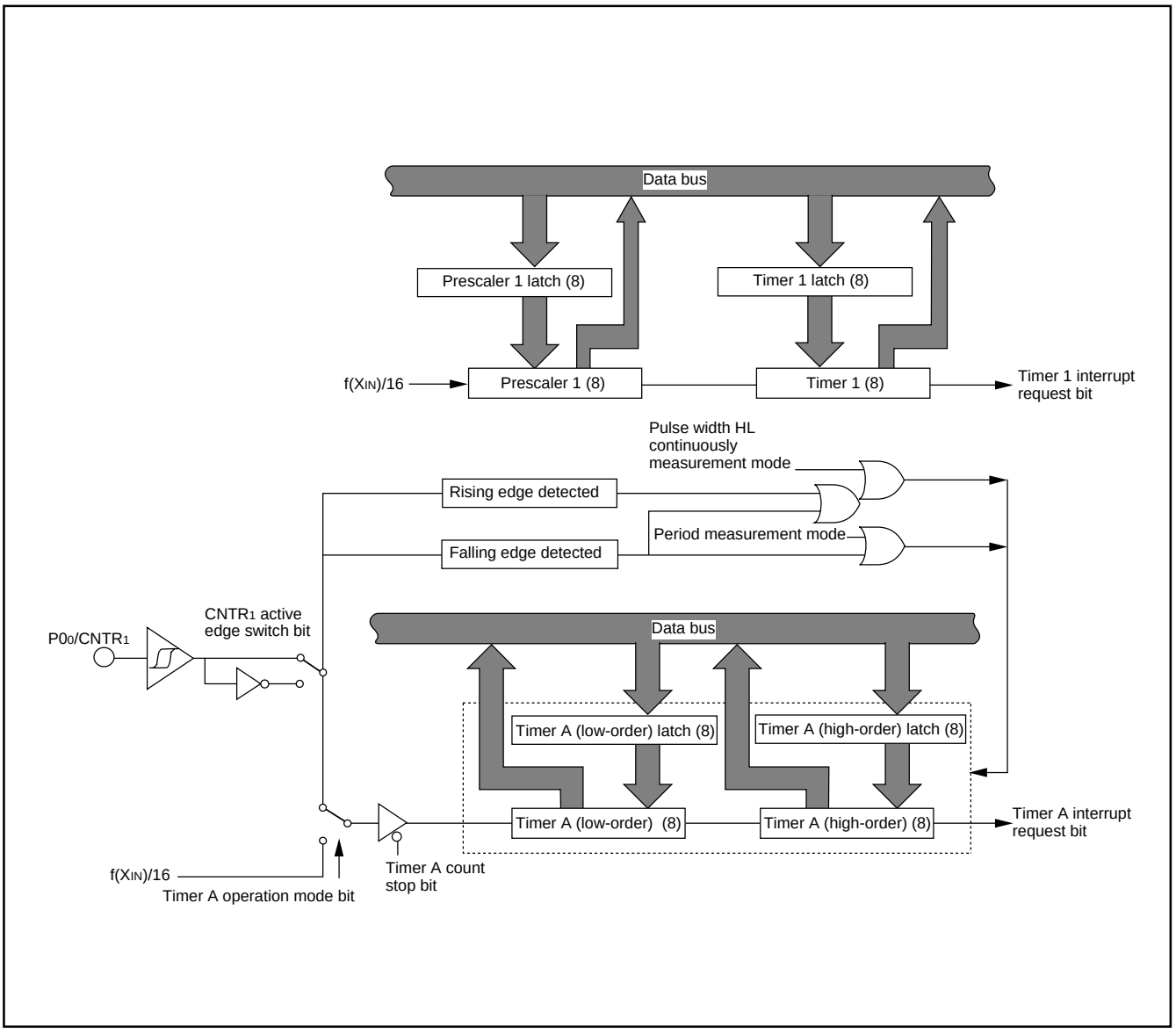


Fig. 28 Block diagram of timer 1 and timer A

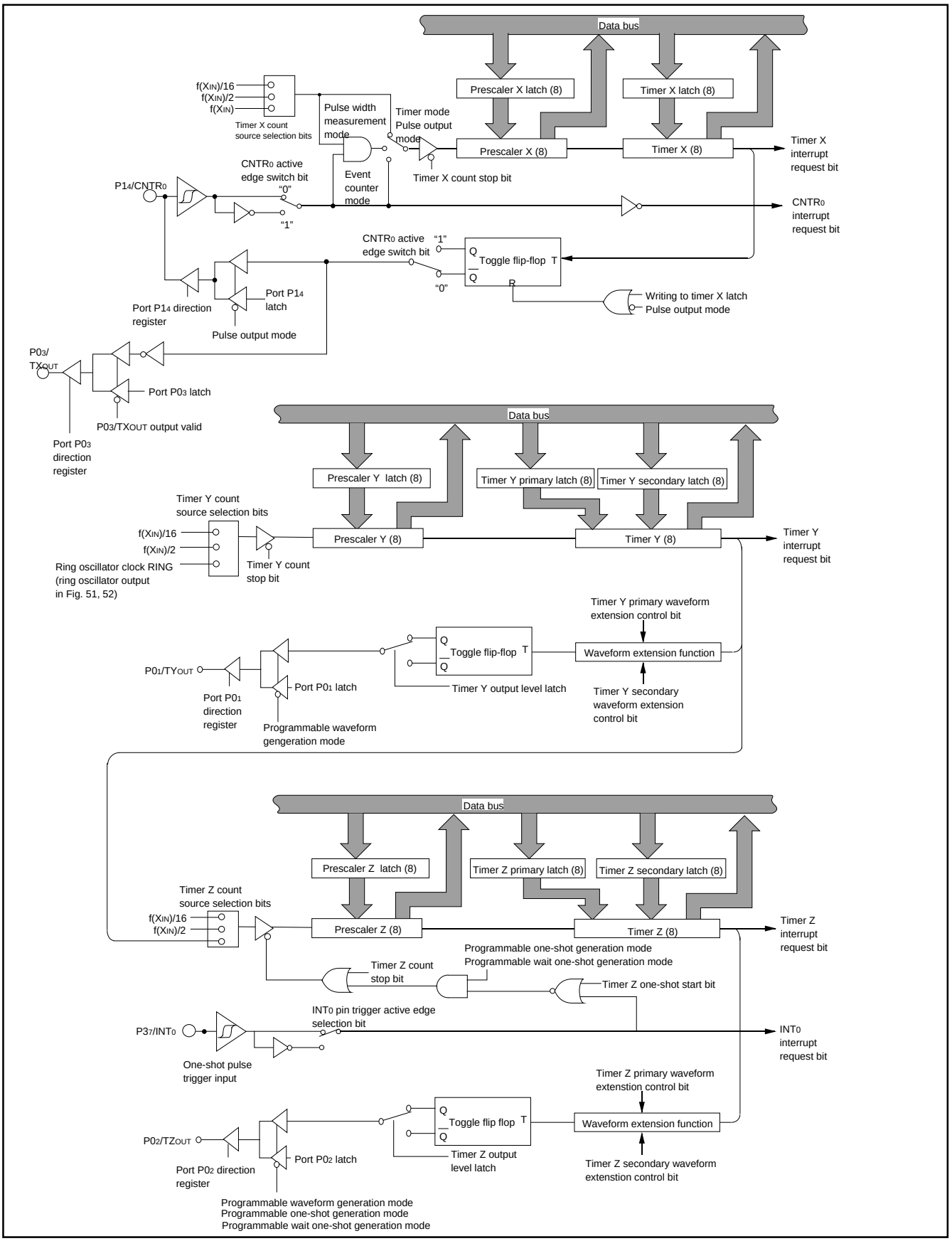


Fig. 29 Block diagram of timer X, timer Y and timer Z

Serial I/O
●Serial I/O1

Serial I/O1 can be used as either clock synchronous or asynchronous (UART) serial I/O. A dedicated timer is also provided for baud rate generation.

(1) Clock Synchronous Serial I/O Mode

Clock synchronous serial I/O1 mode can be selected by setting the serial I/O1 mode selection bit of the serial I/O1 control register (bit 6) to "1".

For clock synchronous serial I/O1, the transmitter and the receiver must use the same clock. If an internal clock is used, transfer is started by a write signal to the TB/RB.

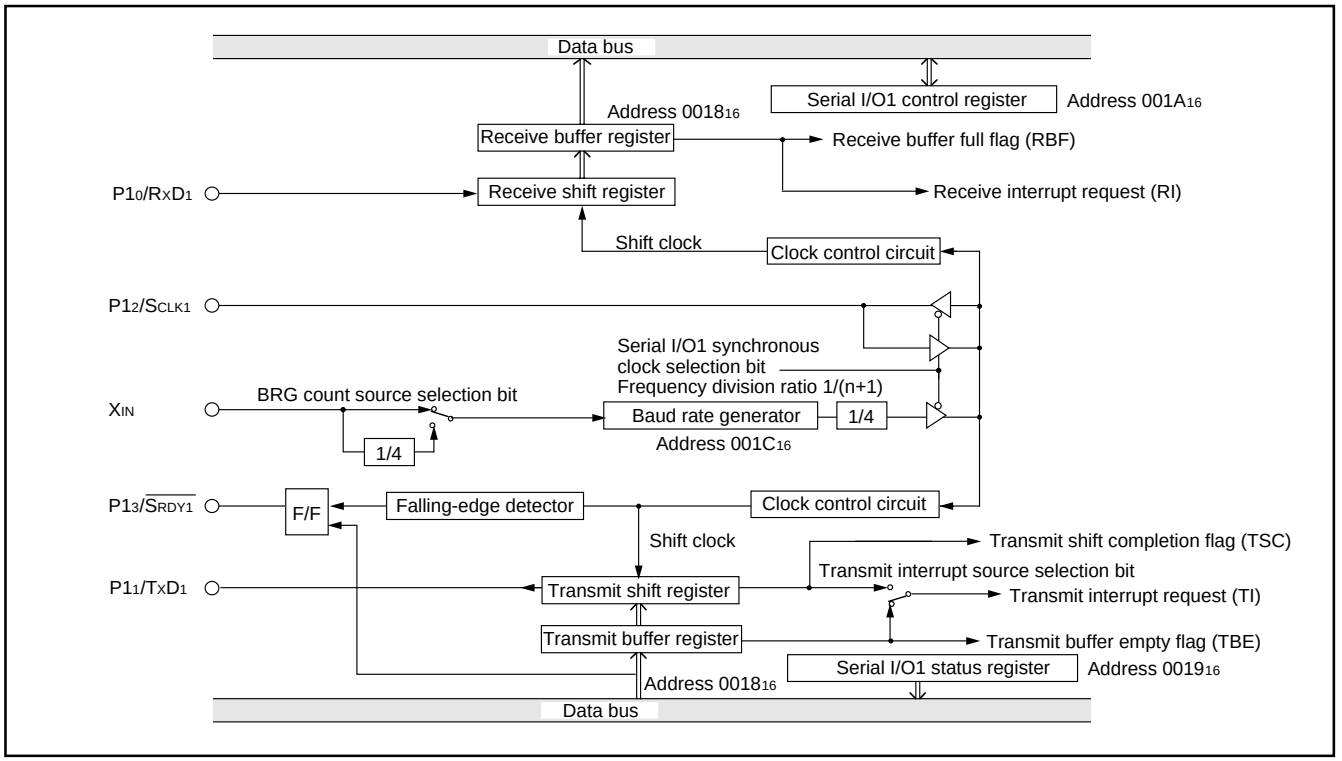
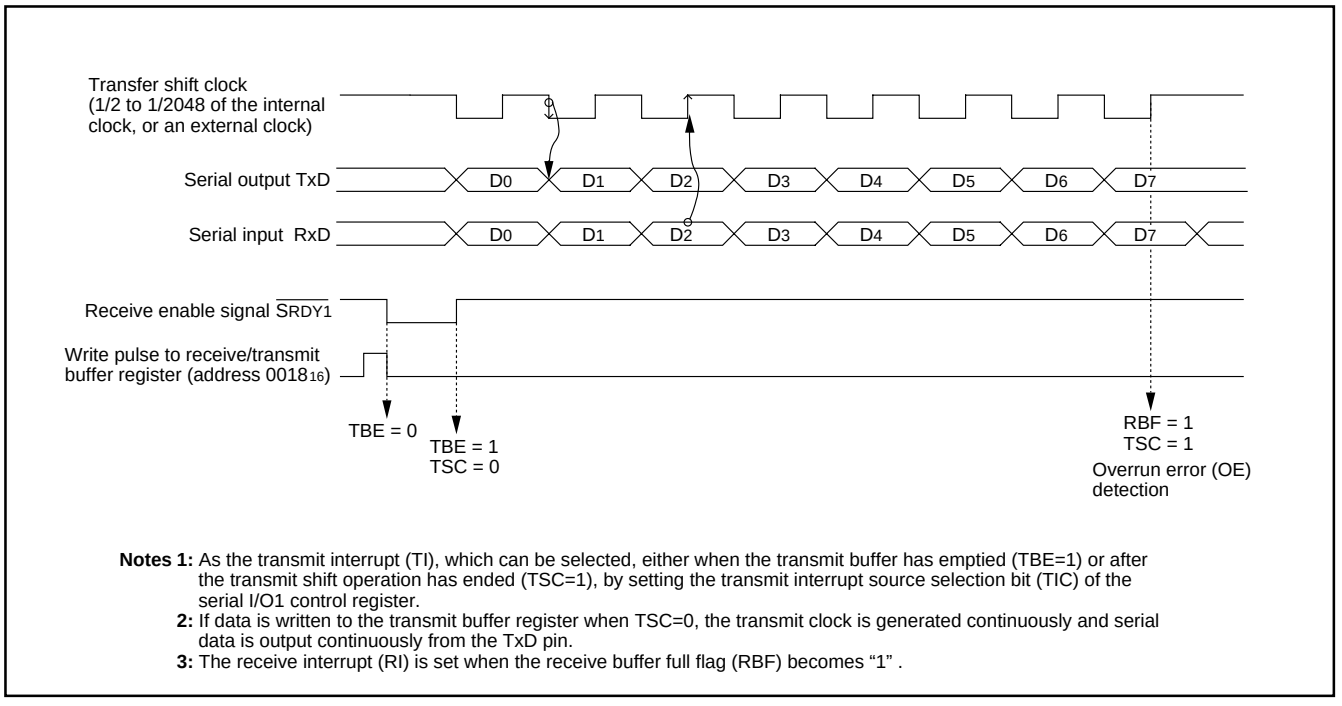


Fig. 30 Block diagram of clock synchronous serial I/O1



Notes 1: As the transmit interrupt (TI), which can be selected, either when the transmit buffer has emptied (TBE=1) or after the transmit shift operation has ended (TSC=1), by setting the transmit interrupt source selection bit (TIC) of the serial I/O1 control register.
2: If data is written to the transmit buffer register when TSC=0, the transmit clock is generated continuously and serial data is output continuously from the TxD pin.
3: The receive interrupt (RI) is set when the receive buffer full flag (RBF) becomes "1" .

Fig. 31 Operation of clock synchronous serial I/O1 function

(2) Asynchronous Serial I/O (UART) Mode

Clock asynchronous serial I/O mode (UART) can be selected by clearing the serial I/O1 mode selection bit of the serial I/O1 control register to "0".

Eight serial data transfer formats can be selected, and the transfer formats used by a transmitter and receiver must be identical.

The transmit and receive shift registers each have a buffer, but the two buffers have the same address in memory. Since the shift register cannot be written to or read from directly, transmit data is written to the transmit buffer register, and receive data is read from the receive buffer register.

The transmit buffer register can also hold the next data to be transmitted, and the receive buffer register can hold a character while the next character is being received.

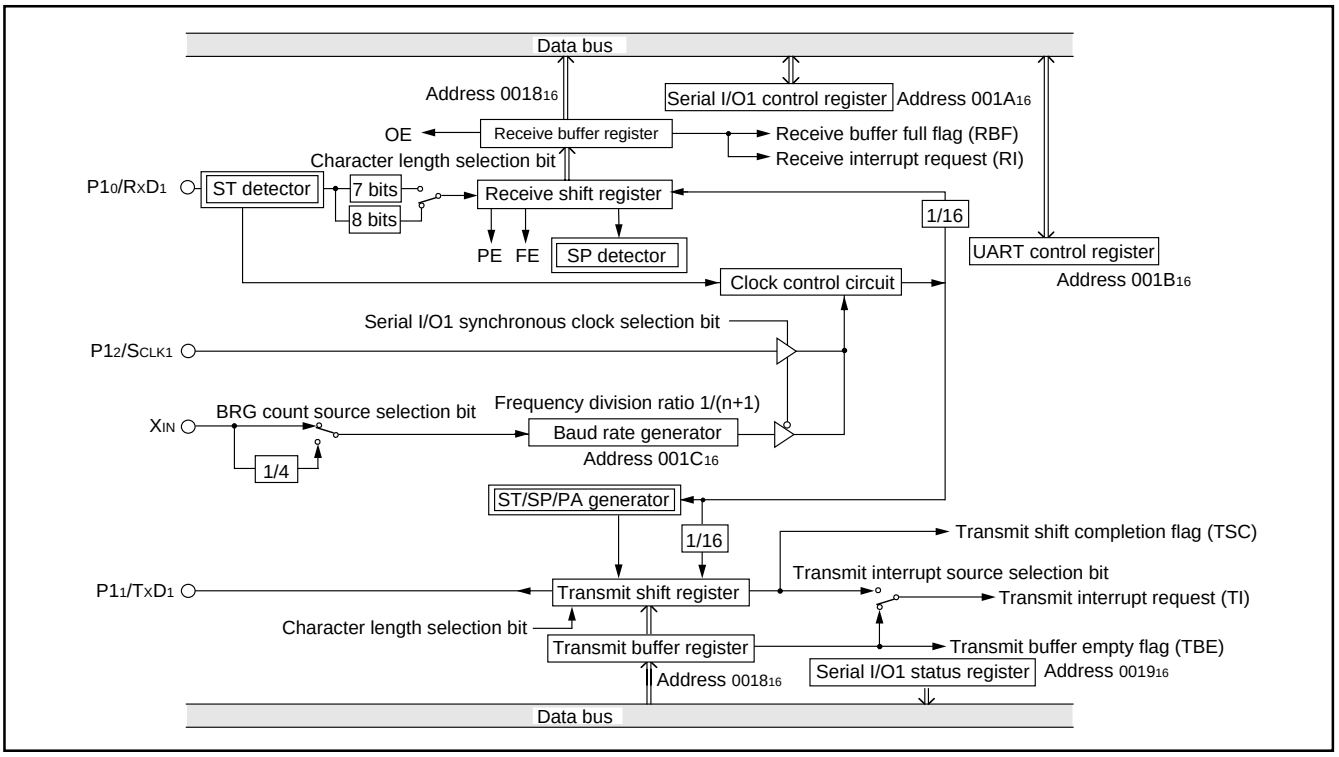
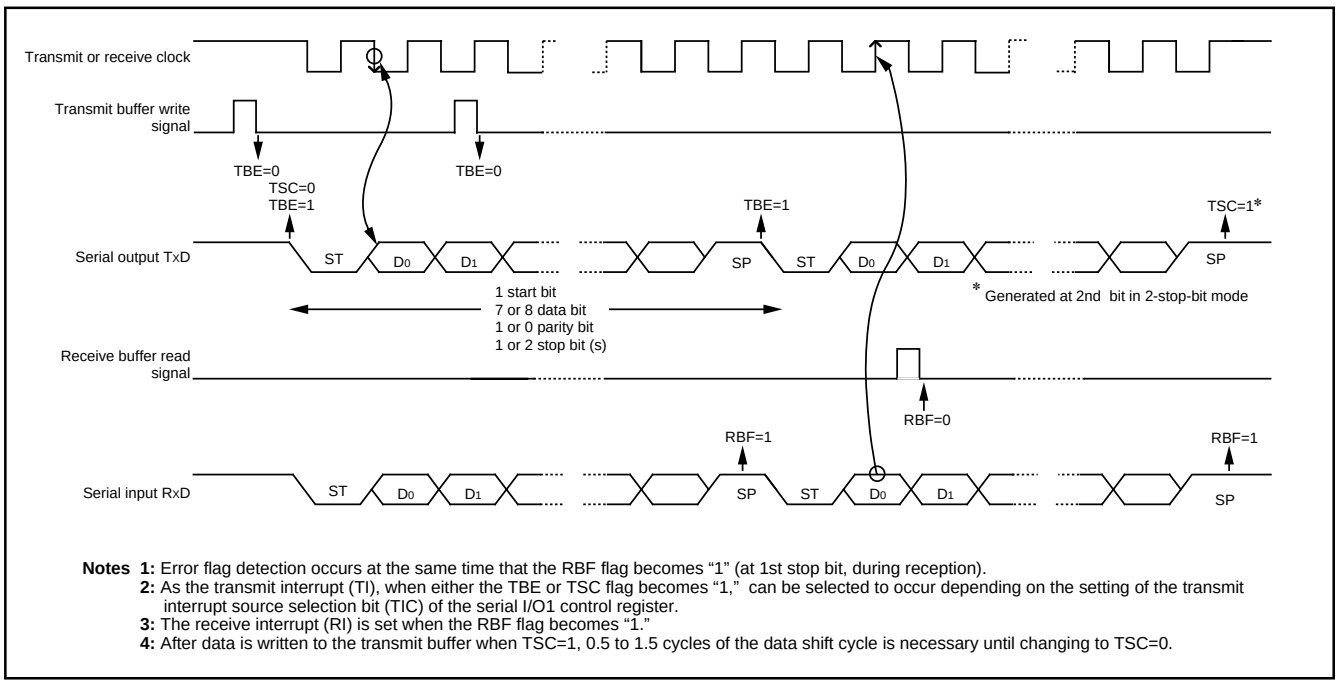


Fig. 32 Block diagram of UART serial I/O1



Notes 1: Error flag detection occurs at the same time that the RBF flag becomes "1" (at 1st stop bit, during reception).
2: As the transmit interrupt (TI), when either the TBE or TSC flag becomes "1," can be selected to occur depending on the setting of the transmit interrupt source selection bit (TIC) of the serial I/O1 control register.
3: The receive interrupt (RI) is set when the RBF flag becomes "1."
4: After data is written to the transmit buffer when TSC=1, 0.5 to 1.5 cycles of the data shift cycle is necessary until changing to TSC=0.

Fig. 33 Operation of UART serial I/O1 function

[Transmit buffer register/receive buffer register (TB/RB)] 0018_{16}

The transmit buffer register and the receive buffer register are located at the same address. The transmit buffer is write-only and the receive buffer is read-only. If a character bit length is 7 bits, the MSB of data stored in the receive buffer is "0".

[Serial I/O1 status register (SIO1STS)] 0019_{16}

The read-only serial I/O1 status register consists of seven flags (bits 0 to 6) which indicate the operating status of the serial I/O1 function and various errors.

Three of the flags (bits 4 to 6) are valid only in UART mode.

The receive buffer full flag (bit 1) is cleared to "0" when the receive buffer register is read.

If there is an error, it is detected at the same time that data is transferred from the receive shift register to the receive buffer register, and the receive buffer full flag is set. A write to the serial I/O1 status register clears all the error flags OE, PE, FE, and SE (bit 3 to bit 6, respectively). Writing "0" to the serial I/O1 enable bit SIOE (bit 7 of the serial I/O1 control register) also clears all the status flags, including the error flags.

Bits 0 to 6 of the serial I/O1 status register are initialized to "0" at reset, but if the transmit enable bit of the serial I/O1 control register has been set to "1", the transmit shift completion flag (bit 2) and the transmit buffer empty flag (bit 0) become "1".

[Serial I/O1 control register (SIO1CON)] $001A_{16}$

The serial I/O1 control register consists of eight control bits for the serial I/O1 function.

[UART control register (UARTCON)] $001B_{16}$

The UART control register consists of four control bits (bits 0 to 3) which are valid when asynchronous serial I/O is selected and set the data format of an data transfer and one bit (bit 4) which is always valid and sets the output structure of the $P1_1$/TXD_1 pin.

[Baud rate generator (BRG)] $001C_{16}$

The baud rate generator determines the baud rate for serial transfer. The baud rate generator divides the frequency of the count source by 1/(n + 1), where n is the value written to the baud rate generator.

■ Notes on serial I/O

• Serial I/O interrupt

When setting the transmit enable bit to "1", the serial I/O transmit interrupt request bit is automatically set to "1". When not requiring the interrupt occurrence synchronized with the transmission enabled, take the following sequence.

① Set the serial I/O transmit interrupt enable bit to "0" (disabled).
② Set the transmit enable bit to "1".
③ Set the serial I/O transmit interrupt request bit to "0" after 1 or more instructions have been executed.
⑤ Set the serial I/O transmit interrupt enable bit to "1" (enabled).

• I/O pin function when serial I/O1 is enabled.

The functions of $P1_2$ and $P1_3$ are switched with the setting values of a serial I/O1 mode selection bit and a serial I/O1 synchronous clock selection bit as follows.

(1) Serial I/O1 mode selection bit → "1" :
Clock synchronous type serial I/O is selected.
Setup of a serial I/O1 synchronous clock selection bit
"0" : $P1_2$ pin turns into an output pin of a synchronous clock.
"1" : $P1_2$ pin turns into an input pin of a synchronous clock.
Setup of a SRDY1 output enable bit (SRDY)
"0" : $P1_3$ pin can be used as a normal I/O pin.
"1" : $P1_3$ pin turns into a SRDY output pin.

(2) Serial I/O1 mode selection bit → "0" :
Clock asynchronous (UART) type serial I/O is selected.
Setup of a serial I/O1 synchronous clock selection bit
"0": $P1_2$ pin can be used as a normal I/O pin.
"1": $P1_2$ pin turns into an input pin of an external clock.
When clock asynchronous (UART) type serial I/O is selected, it is $P1_3$ pin. It can be used as a normal I/O pin.

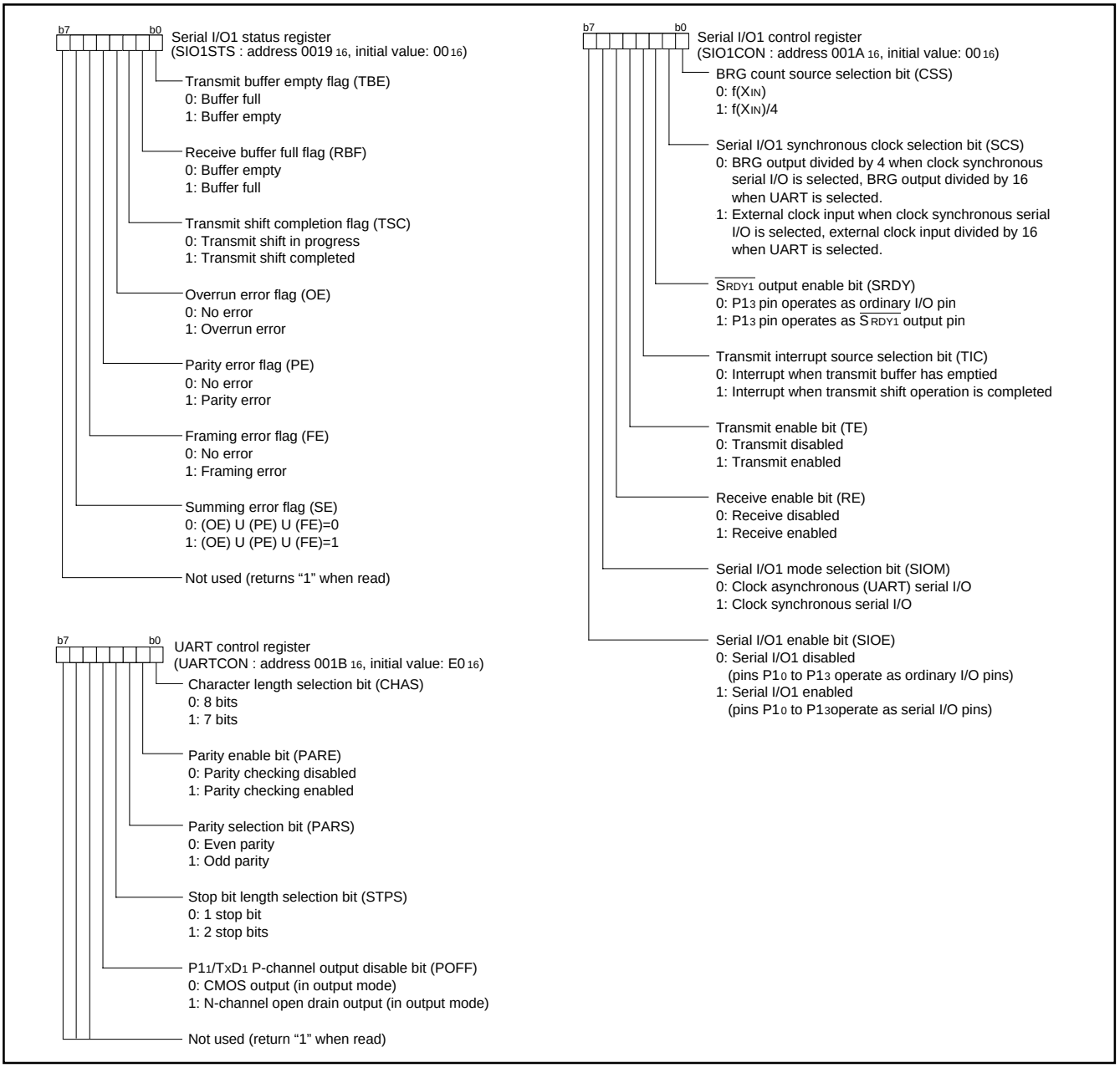


Fig. 34 Structure of serial I/O1-related registers

●Serial I/O2

The serial I/O2 function can be used only for clock synchronous serial I/O.

For clock synchronous serial I/O2 the transmitter and the receiver must use the same clock. When the internal clock is used, transfer is started by a write signal to the serial I/O2 register.

Note: Serial I/O2 can be used in the following cases;

(1) Serial I/O1 is not used,

(2) Serial I/O1 is used as UART and BRG output divided by 16 is selected as the synchronized clock.

[Serial I/O2 control register] SIO2CON

The serial I/O2 control register contains 8 bits which control various serial I/O functions.

• Set “0” to bit 3 to receive.

• At reception, clear bit 7 to “0” by writing a dummy data to the serial I/O2 register after completion of shift.

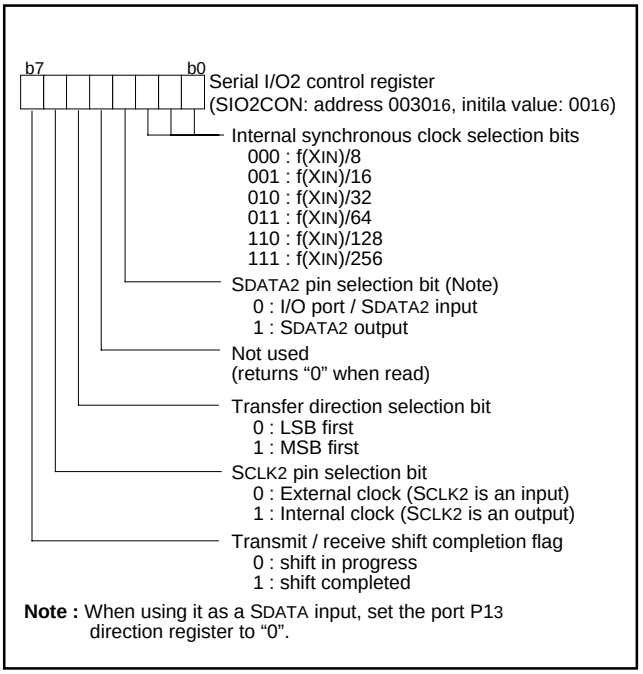


Fig. 35 Structure of serial I/O2 control registers

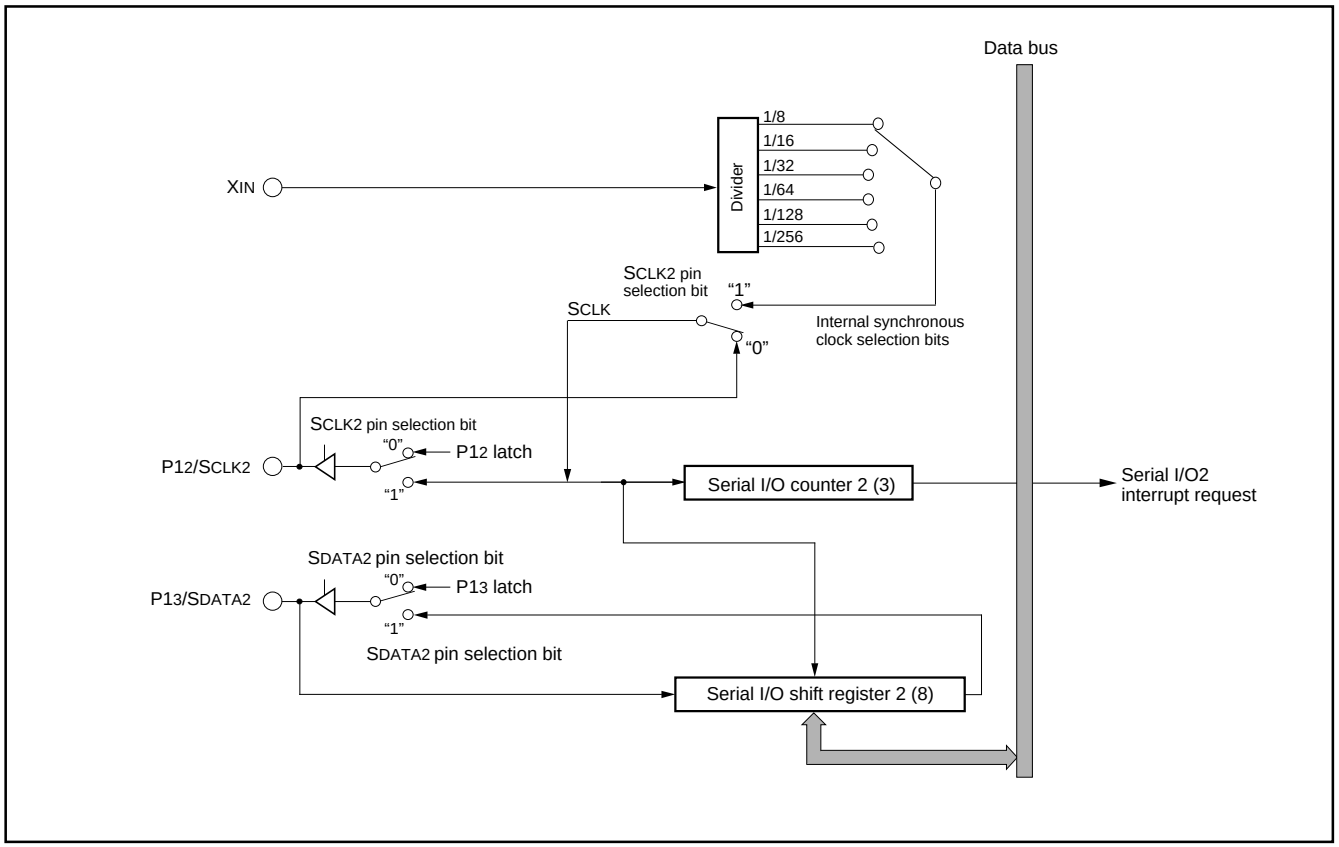


Fig. 36 Block diagram of serial I/O2

Serial I/O2 operation

By writing to the serial I/O2 register (address 0031_{16}) the serial I/O2 counter is set to “7”.

After writing, the SDATA2 pin outputs data every time the transfer clock shifts from “H” to “L”. And, as the transfer clock shifts from “L” to “H”, the SDATA2 pin reads data, and at the same time the contents of the serial I/O2 register are shifted by 1 bit.

When the internal clock is selected as the transfer clock source, the following operations execute as the transfer clock counts up to 8.

- Serial I/O2 counter is cleared to “0”.
- Transfer clock stops at an “H” level.
- Interrupt request bit is set.
- Shift completion flag is set.

Also, the SDATA2 pin is in a high impedance state after the data transfer is completed (refer to Fig.37).

When the external clock is selected as the transfer clock source, the interrupt request bit is set as the transfer clock counts up to 8, but external control of the clock is required since it does not stop. Notice that the SDATA2 pin is not in a high impedance state on the completion of data transfer.

Also, after the receive operation is completed, the transmit/receive shift completion flag is cleared by reading the serial I/O2 register. At transmit, the transmit/receive shift completion flag is cleared and the transmit operation is started by writing to serial I/O2 register.

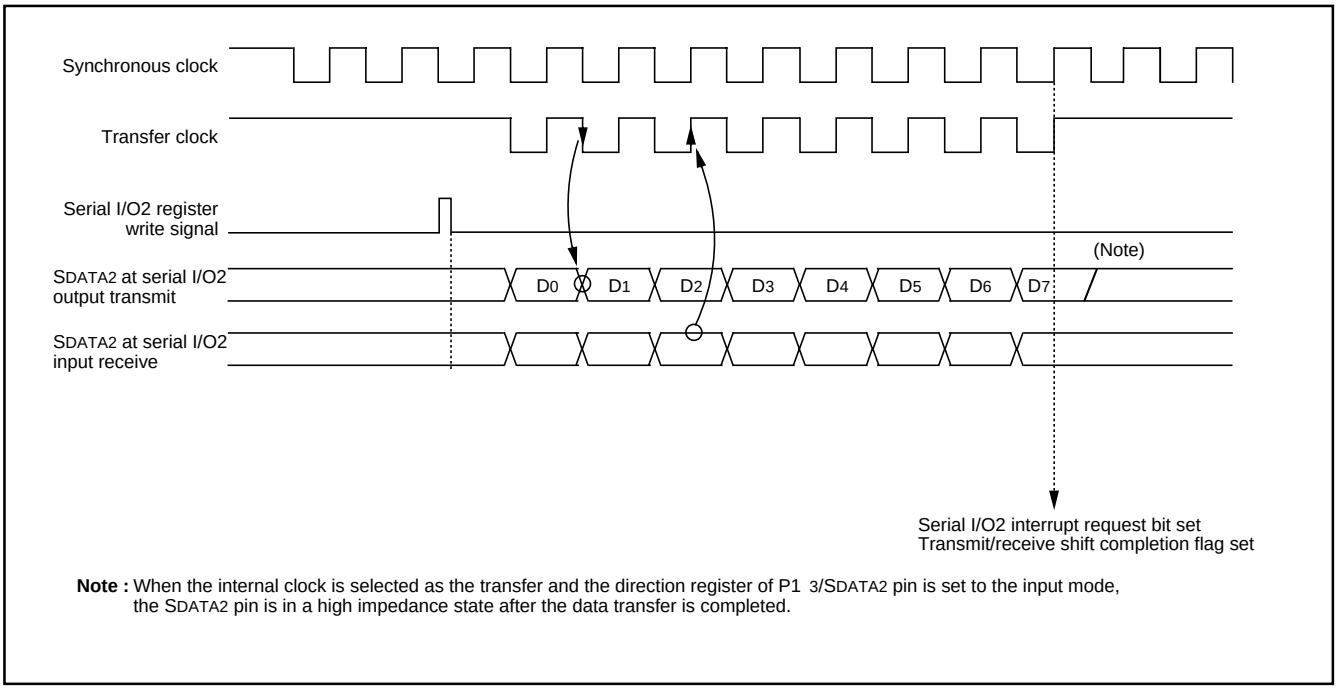


Fig. 37 Serial I/O2 timing (LSB first)

A-D Converter

The functional blocks of the A-D converter are described below.

[A-D conversion register] AD

The A-D conversion register is a read-only register that stores the result of A-D conversion. Do not read out this register during an A-D conversion.

[A-D control register] ADCON

The A-D control register controls the A-D converter. Bit 2 to 0 are analog input pin selection bits. Bit 4 is the AD conversion completion bit. The value of this bit remains at "0" during A-D conversion, and changes to "1" at completion of A-D conversion.
A-D conversion is started by setting this bit to "0".

[Comparison voltage generator]

The comparison voltage generator divides the voltage between AVSS and VREF by 1024, and outputs the divided voltages.

[Channel selector]

The channel selector selects one of ports P27/AN7 to P20/AN0, and inputs the voltage to the comparator.

[Comparator and control circuit]

The comparator and control circuit compares an analog input voltage with the comparison voltage and stores its result into the A-D conversion register. When A-D conversion is completed, the control circuit sets the AD conversion completion bit and the AD interrupt request bit to "1". Because the comparator is constructed linked to a capacitor, set f(XIN) to 500 kHz or more during A-D conversion.

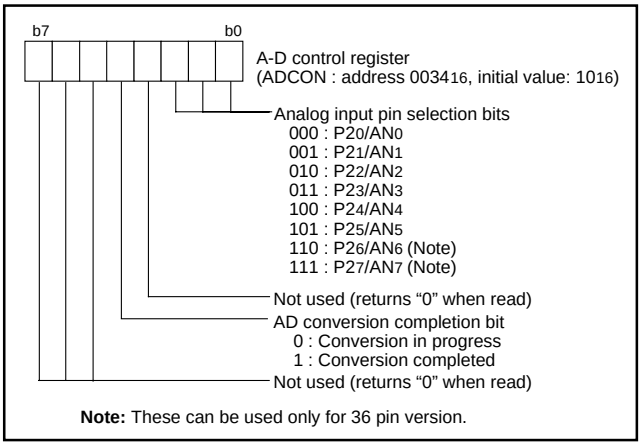


Fig. 38 Structure of A-D control register

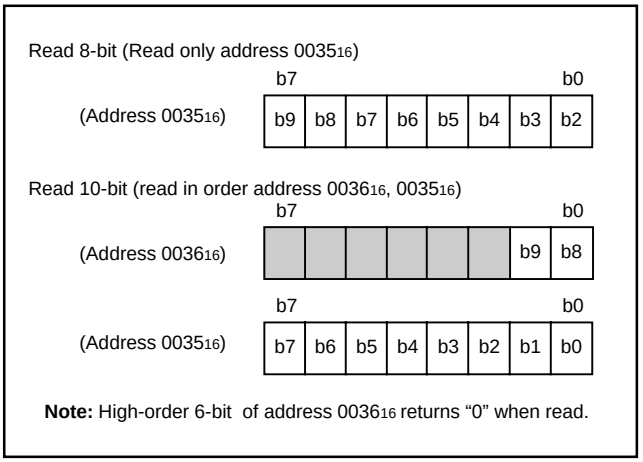


Fig. 39 Structure of A-D conversion register

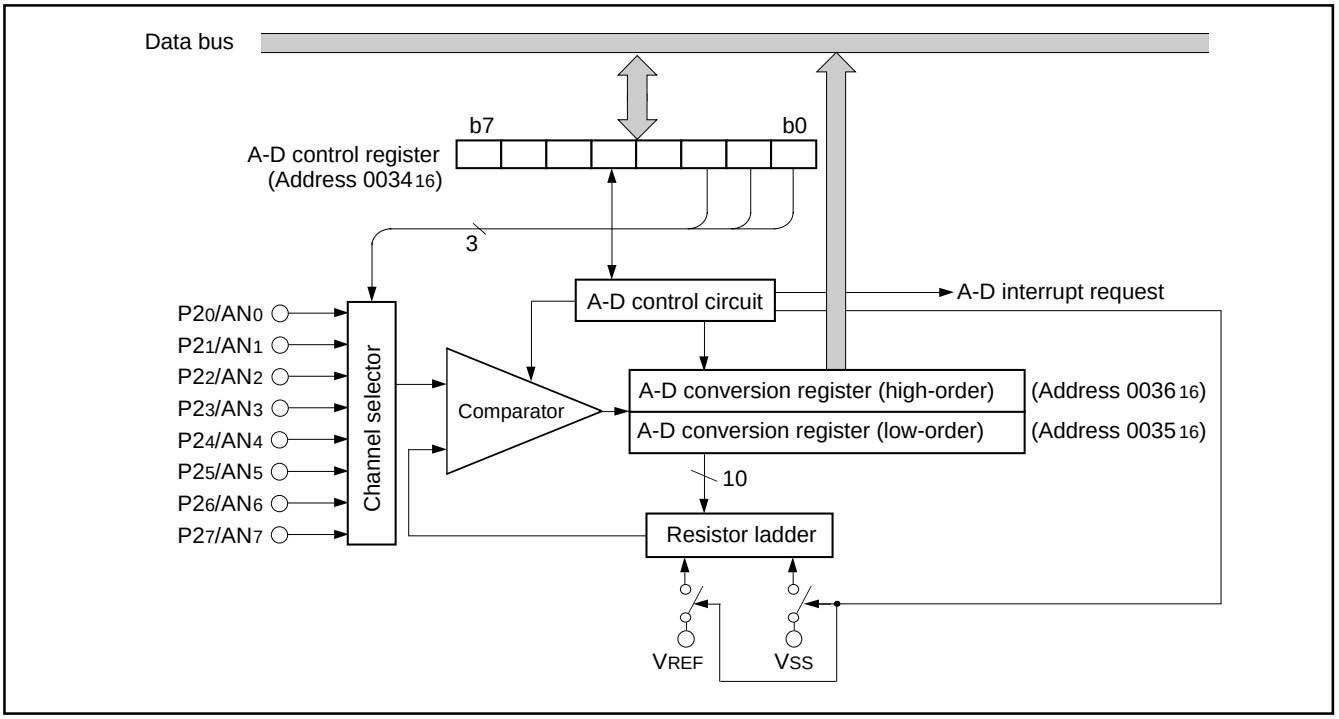


Fig. 40 Block diagram of A-D converter

Watchdog Timer

The watchdog timer gives a means for returning to a reset status when the program fails to run on its normal loop due to a runaway. The watchdog timer consists of an 8-bit watchdog timer H and an 8-bit watchdog timer L, being a 16-bit counter.

Standard operation of watchdog timer

The watchdog timer stops when the watchdog timer control register (address 0039_{16}) is not set after reset. Writing an optional value to the watchdog timer control register (address 0039_{16}) causes the watchdog timer to start to count down. When the watchdog timer H underflows, an internal reset occurs. Accordingly, it is programmed that the watchdog timer control register (address 0039_{16}) can be set before an underflow occurs.

When the watchdog timer control register (address 0039_{16}) is read, the values of the high-order 6-bit of the watchdog timer H, STP instruction disable bit and watchdog timer H count source selection bit are read.

Initial value of watchdog timer

By a reset or writing to the watchdog timer control register (address 0039_{16}), the watchdog timer H is set to "FF_{16}" and the watchdog timer L is set to "FF_{16}".

Operation of watchdog timer H count source selection bit

A watchdog timer H count source can be selected by bit 7 of the watchdog timer control register (address 0039_{16}). When this bit is "0", the count source becomes a watchdog timer L underflow signal. The detection time is 131.072 ms at f(XIN)=8 MHz.

When this bit is "1", the count source becomes f(XIN)/16. In this case, the detection time is 512 μs at f(XIN)=8 MHz.

This bit is cleared to "0" after reset.

Operation of STP instruction disable bit

When the watchdog timer is in operation, the STP instruction can be disabled by bit 6 of the watchdog timer control register (address 0039_{16}).

When this bit is "0", the STP instruction is enabled.

When this bit is "1", the STP instruction is disabled, and an internal reset occurs if the STP instruction is executed.

Once this bit is set to "1", it cannot be changed to "0" by program.

This bit is cleared to "0" after reset.

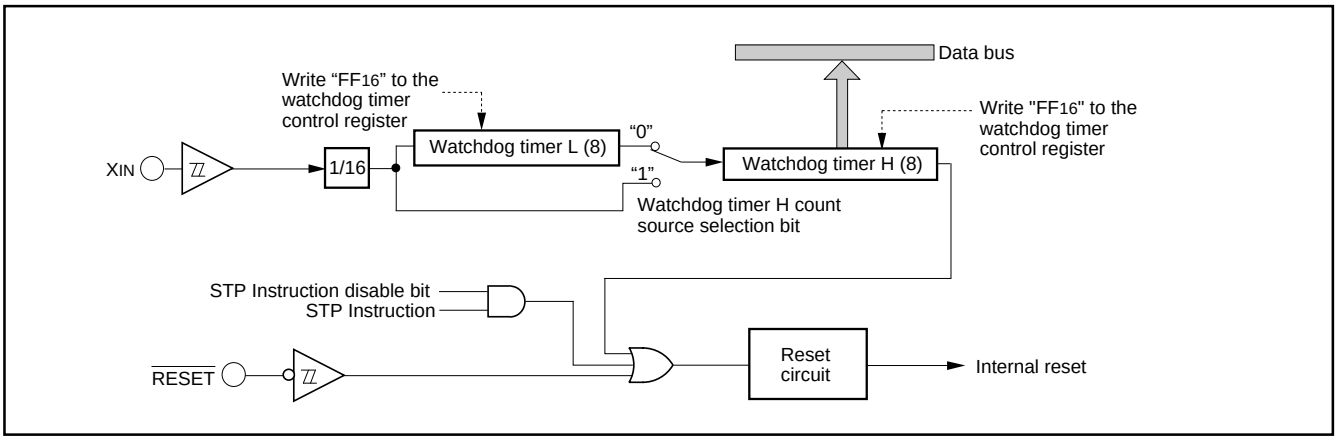


Fig. 41 Block diagram of watchdog timer

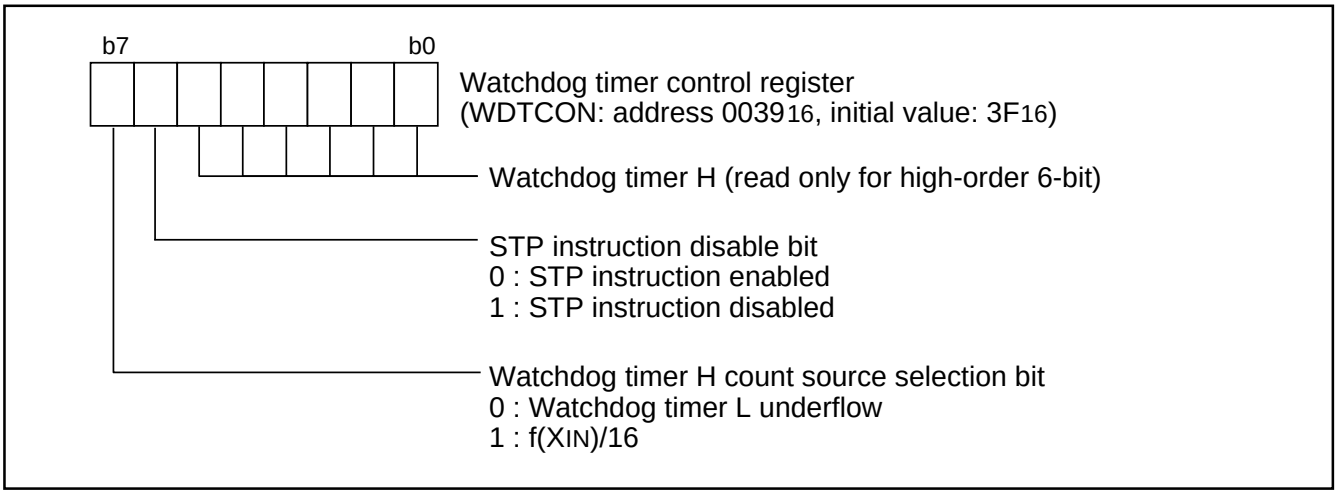


Fig. 42 Structure of watchdog timer control register

Reset Circuit

The microcomputer is put into a reset status by holding the $\overline{\text{RESET}}$ pin at the "L" level for 2 μs or more when the power source voltage is 2.2 to 5.5 V and X_{IN} is in stable oscillation.

After that, this reset status is released by returning the $\overline{\text{RESET}}$ pin to the "H" level. The program starts from the address having the contents of address $FFFD_{16}$ as high-order address and the contents of address $FFFC_{16}$ as low-order address.

In the case of f(ϕ) ≤ 6 MHz, the reset input voltage must be 0.9 V or less when the power source voltage passes 4.5 V.

In the case of f(ϕ) ≤ 4 MHz, the reset input voltage must be 0.8 V or less when the power source voltage passes 4.0 V.

In the case of f(ϕ) ≤ 2 MHz, the reset input voltage must be 0.48 V or less when the power source voltage passes 2.4 V.

In the case of f(ϕ) ≤ 1 MHz, the reset input voltage must be 0.44 V or less when the power source voltage passes 2.2 V.

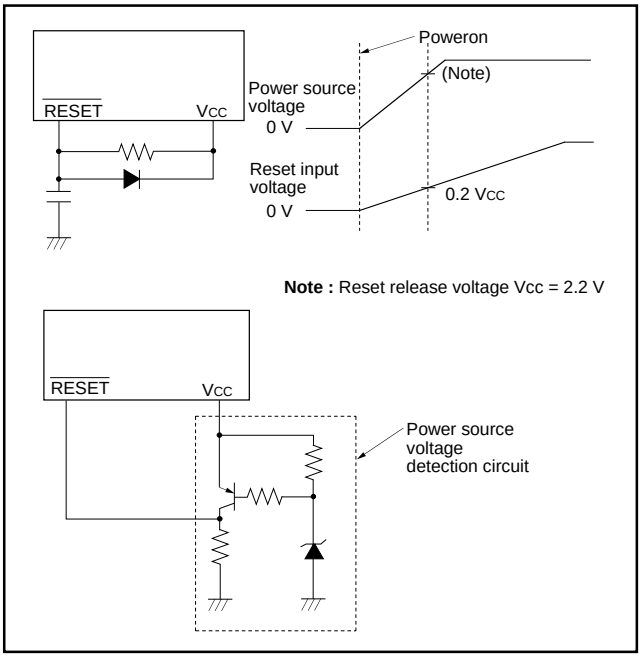


Fig. 43 Example of reset circuit

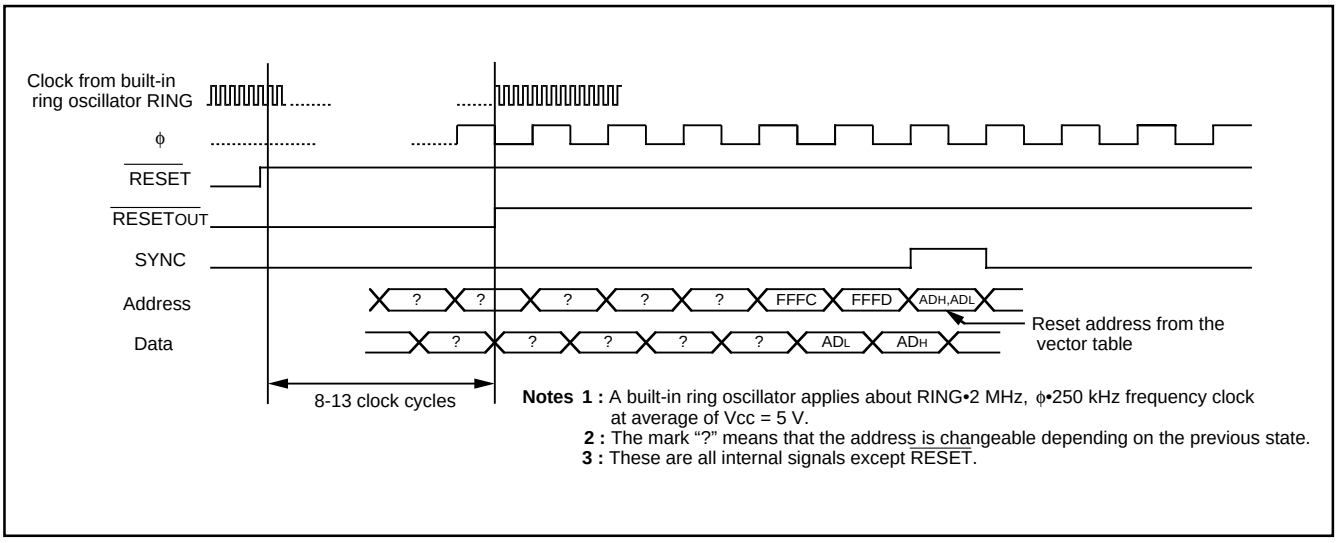


Fig. 44 Timing diagram at reset

	Address	Register contents
(1) Port P0 direction register	0001_{16}	00_{16}
(2) Port P1 direction register	0003_{16}	X X X 0 0 0 0 0
(3) Port P2 direction register	0005_{16}	00_{16}
(4) Port P3 direction register	0007_{16}	00_{16}
(5) Pull-up control register	0016_{16}	00_{16}
(6) Port P1P3 control register	0017_{16}	00_{16}
(7) Serial I/O1 status register	0019_{16}	1 0 0 0 0 0 0 0
(8) Serial I/O1 control register	$001A_{16}$	00_{16}
(9) UART control register	$001B_{16}$	1 1 1 0 0 0 0 0
(10) Timer A mode register	$001D_{16}$	00_{16}
(11) Timer A (low-order)	$001E_{16}$	FF_{16}
(12) Timer A (high-order)	$001F_{16}$	FF_{16}
(13) Timer Y, Z mode register	0020_{16}	00_{16}
(14) Prescaler Y	0021_{16}	FF_{16}
(15) Timer Y secondary	0022_{16}	FF_{16}
(16) Timer Y primary	0023_{16}	FF_{16}
(17) Timer Y, Z waveform output control register	0024_{16}	00_{16}
(18) Prescaler Z	0025_{16}	FF_{16}
(19) Timer Z secondary	0026_{16}	FF_{16}
(20) Timer Z primary	0027_{16}	FF_{16}
(21) Prescaler 1	0028_{16}	FF_{16}
(22) Timer 1	0029_{16}	01_{16}
(23) One-shot start register	$002A_{16}$	00_{16}
(24) Timer X mode register	$002B_{16}$	00_{16}
(25) Prescaler X	$002C_{16}$	FF_{16}
(26) Timer X	$002D_{16}$	FF_{16}
(27) Timer count source set register	$002E_{16}$	00_{16}
(28) Serial I/O2 control register	0030_{16}	00_{16}
(29) Serial I/O2 register	0031_{16}	00_{16}
(30) A-D control register	0034_{16}	10_{16}
(31) MISRG	0038_{16}	00_{16}
(32) Watchdog timer control register	0039_{16}	0 0 1 1 1 1 1 1
(33) Interrupt edge selection register	$003A_{16}$	00_{16}
(34) CPU mode register	$003B_{16}$	1 0 0 0 0 0 0 0
(35) Interrupt request register 1	$003C_{16}$	00_{16}
(36) Interrupt request register 2	$003D_{16}$	00_{16}
(37) Interrupt control register 1	$003E_{16}$	00_{16}
(38) Interrupt control register 2	$003F_{16}$	00_{16}
(39) Processor status register	(PS)	X X X X X 1 X X
(40) Program counter	(PC_H)	Contents of address $FFFD_{16}$
	(PC_L)	Contents of address $FFFC_{16}$

Note X : Undefined

Fig. 45 Internal status of microcomputer at reset

Clock Generating Circuit

An oscillation circuit can be formed by connecting a resonator between XIN and XOUT, and an RC oscillation circuit can be formed by connecting a resistor and a capacitor.
Use the circuit constants in accordance with the resonator manufacturer's recommended values.

(1) Ring oscillator operation

When the MCU operates by the ring oscillator for the main clock, connect XIN pin to VSS and leave XOUT pin open.
The clock frequency of the ring oscillator depends on the supply voltage and the operation temperature range.
Be careful that variable frequencies when designing application products.

(2) Ceramic resonator

When the ceramic resonator is used for the main clock, connect the ceramic resonator and the external circuit to pins XIN and XOUT at the shortest distance. A feedback resistor is built in between pins XIN and XOUT.

(3) RC oscillation

When the RC oscillation is used for the main clock, connect the XIN pin and XOUT pin to the external circuit of resistor R and the capacitor C at the shortest distance.
The frequency is affected by a capacitor, a resistor and a microcomputer.
So, set the constants within the range of the frequency limits.

(4) External clock

When the external signal clock is used for the main clock, connect the XIN pin to the clock source and leave XOUT pin open.
Select “ceramic resonance” by setting “0” to the Oscillation mode selection bit of CPU mode register (address 003B16).

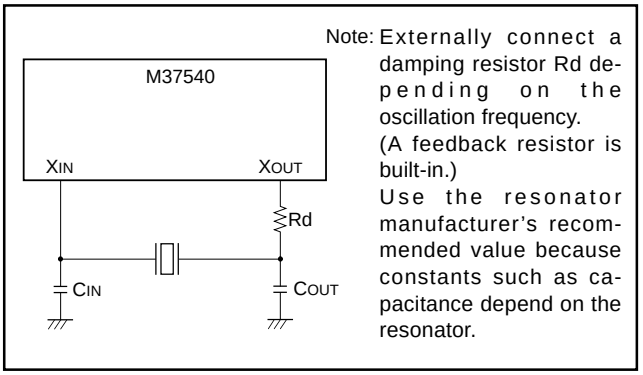


Fig. 46 External circuit of ceramic resonator

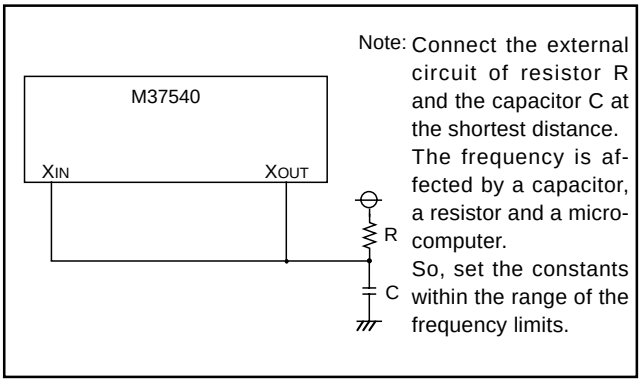


Fig. 47 External circuit of RC oscillation

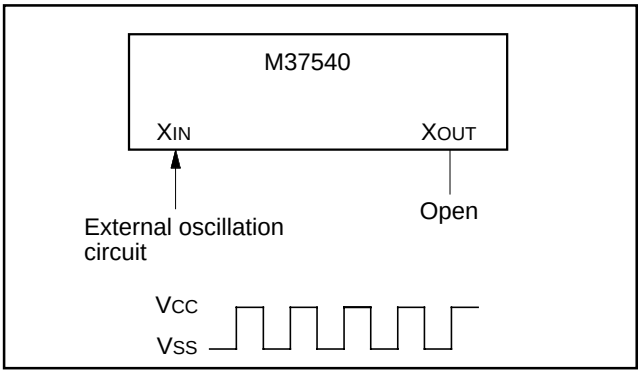


Fig. 48 External clock input circuit

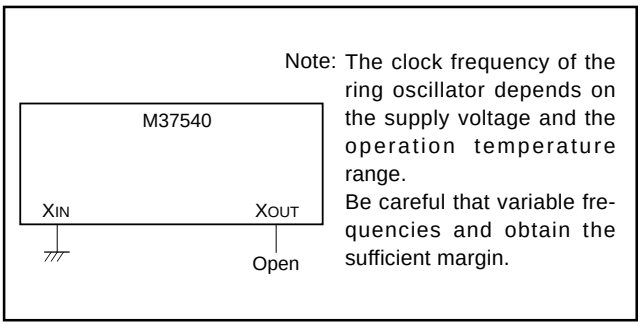


Fig. 49 Processing of XIN and XOUT pins at ring oscillator operation

(1) Oscillation control

• Stop mode

When the STP instruction is executed, the internal clock φ stops at an "H" level and the XIN oscillator stops. At this time, timer 1 is set to "01$_{16}$" and prescaler 1 is set to "FF$_{16}$" when the oscillation stabilization time set bit after release of the STP instruction is "0". On the other hand, timer 1 and prescaler 1 are not set when the above bit is "1". Accordingly, set the wait time fit for the oscillation stabilization time of the oscillator to be used. f(XIN)/16 is forcibly connected to the input of prescaler 1. When an external interrupt is accepted, oscillation is restarted but the internal clock φ remains at "H" until timer 1 underflows. As soon as timer 1 underflows, the internal clock φ is supplied. This is because when a ceramic oscillator is used, some time is required until a start of oscillation. In case oscillation is restarted by reset, no wait time is generated. So apply an "L" level to the RESET pin while oscillation becomes stable.

Also, the STP instruction cannot be used while CPU is operating by a ring oscillator.

• Wait mode

If the WIT instruction is executed, the internal clock φ stops at an "H" level, but the oscillator does not stop. The internal clock restarts if a reset occurs or when an interrupt is received. Since the oscillator does not stop, normal operation can be started immediately after the clock is restarted. To ensure that interrupts will be received to release the STP or WIT state, interrupt enable bits must be set to "1" before the STP or WIT instruction is executed.

■ Notes on clock generating circuit

For use with the oscillation stabilization set bit after release of the STP instruction set to "1", set values in timer 1 and prescaler 1 after fully appreciating the oscillation stabilization time of the oscillator to be used.

• Switch of ceramic and RC oscillations

After releasing reset the operation starts by starting a built-in ring oscillator. Then, a ceramic oscillation or an RC oscillation is selected by setting bit 5 of the CPU mode register.

• Double-speed mode

When a ceramic oscillation is selected, a double-speed mode can be used. Do not use it when an RC oscillation is selected.

• CPU mode register

Bits 5, 1 and 0 of CPU mode register are used to select oscillation mode and to control operation modes of the microcomputer. In order to prevent the dead-lock by error-writing (ex. program run-away), these bits can be rewritten only once after releasing reset. After rewriting it is disable to write any data to the bit. (The emulator MCU "M37540RSS" is excluded.)

Also, when the read-modify-write instructions (SEB, CLB) are executed to bits 2 to 4, 6 and 7, bits 5, 1 and 0 are locked.

• Clock division ratio, XIN oscillation control, ring oscillator control

The state transition shown in Fig. 52 can be performed by setting the clock division ratio selection bits (bits 7 and 6), XIN oscillation control bit (bit 4), ring oscillator oscillation control bit (bit 3) of CPU mode register. Be careful of notes on use in Fig. 52.

● Oscillation stop detection circuit (Note)

The oscillation stop detection circuit is used for reset occurrence when a ceramic resonator or an oscillation circuit stops by disconnection. When internal reset occurs, reset because of oscillation stop can be detected by setting "1" to the oscillation stop detection status bit.

Also, when using the oscillation stop detection circuit, a built-in ring oscillator is required.

Figure 53 shows the state transition.

Note: The oscillation stop detection circuit is not included in the emulator MCU "M37540RSS".

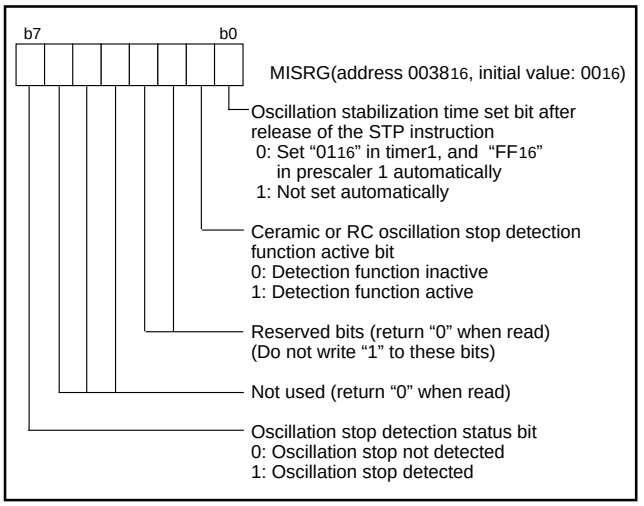


Fig. 50 Structure of MISRG

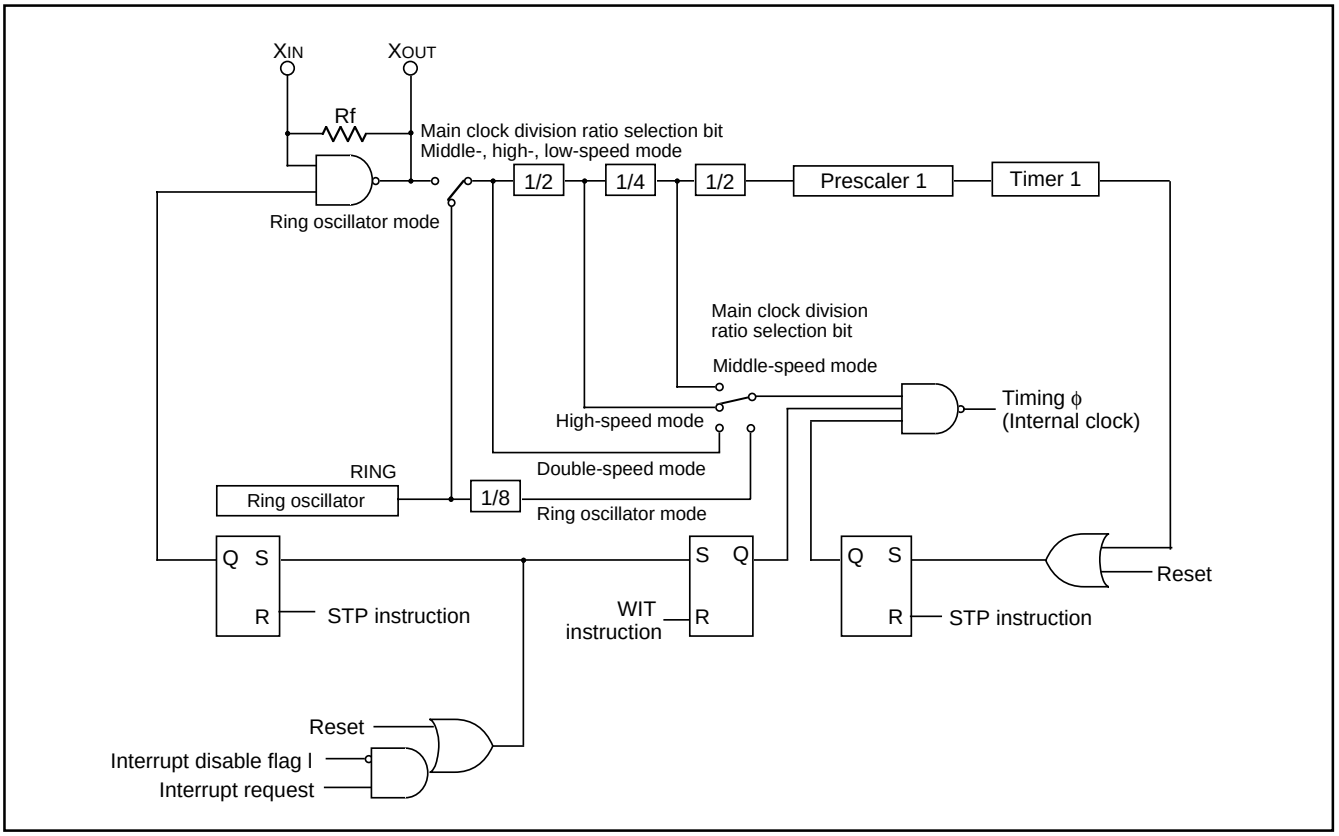


Fig. 51 Block diagram of internal clock generating circuit (for ceramic resonator)

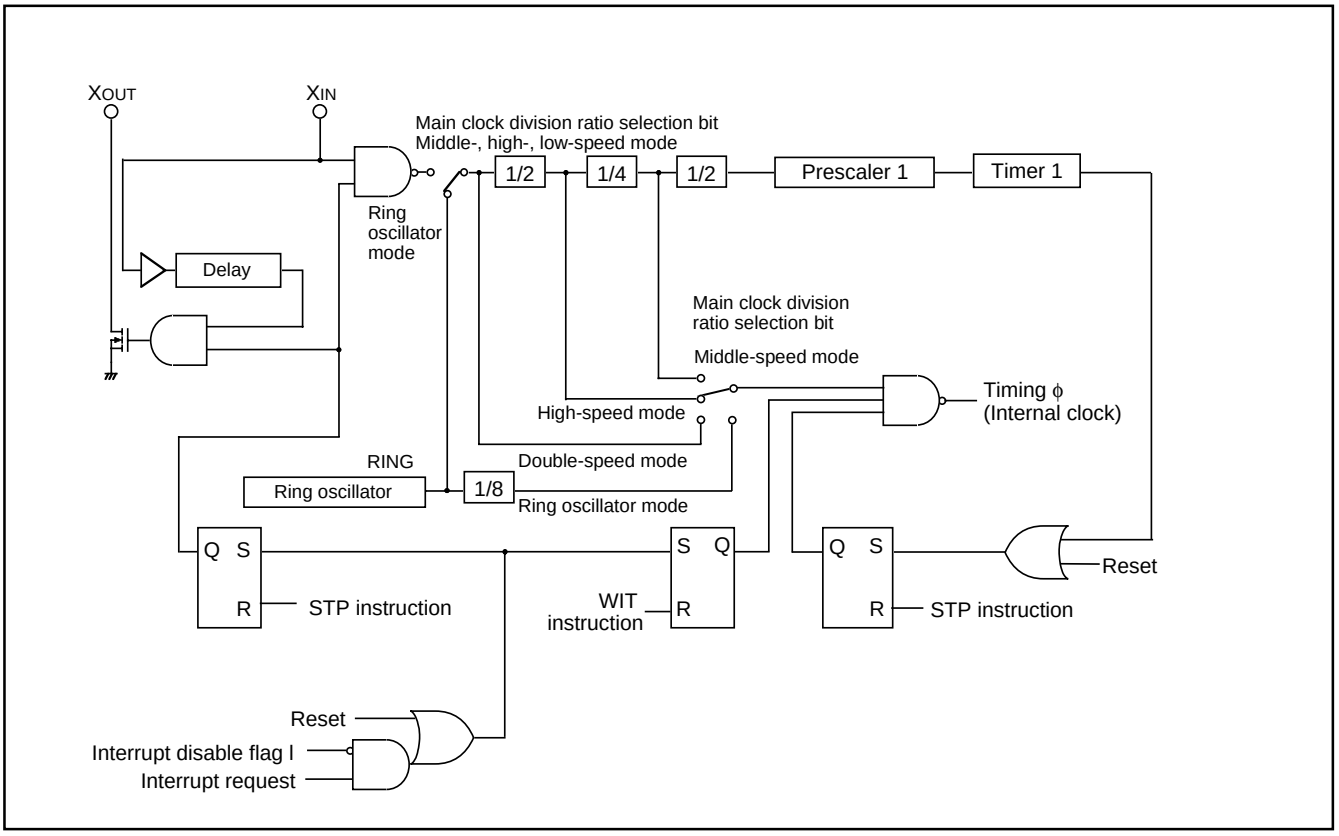


Fig. 52 Block diagram of internal clock generating circuit (for RC oscillation)

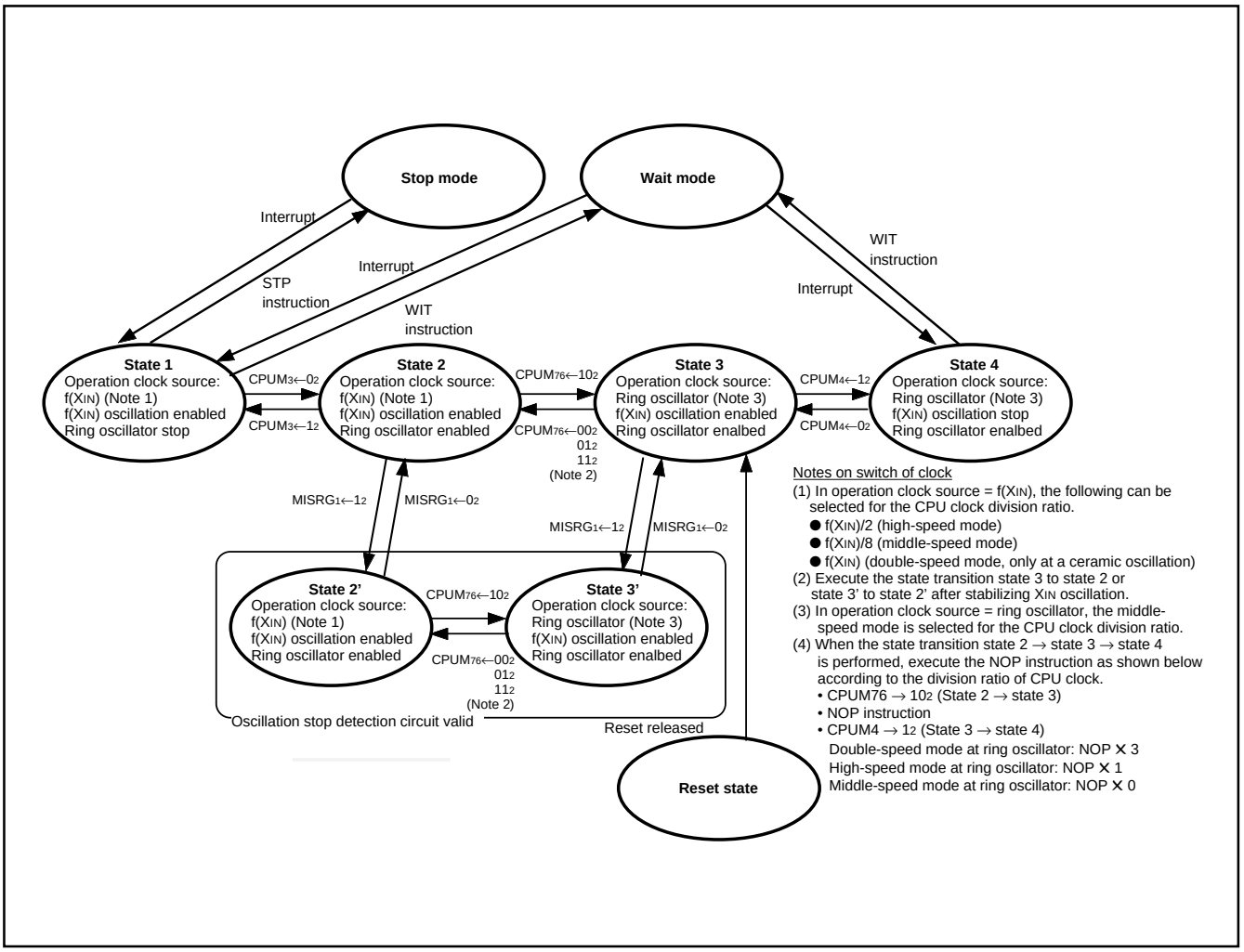


Fig. 53 State transition

NOTES ON PROGRAMMING

Processor Status Register

The contents of the processor status register (PS) after reset are undefined except for the interrupt disable flag I which is "1". After reset, initialize flags which affect program execution. In particular, it is essential to initialize the T flag and the D flag because of their effect on calculations.

Interrupts

The contents of the interrupt request bit do not change even if the BBC or BBS instruction is executed immediately after they are changed by program because this instruction is executed for the previous contents. For executing the instruction for the changed contents, execute one instruction before executing the BBC or BBS instruction.

Decimal Calculations

• For calculations in decimal notation, set the decimal mode flag D to "1", then execute the ADC instruction or SBC instruction. In this case, execute SEC instruction, CLC instruction or CLD instruction after executing one instruction before the ADC instruction or SBC instruction.

• In the decimal mode, the values of the N (negative), V (overflow) and Z (zero) flags are invalid.

Ports

• The values of the port direction registers cannot be read.

That is, it is impossible to use the LDA instruction, memory operation instruction when the T flag is "1", addressing mode using direction register values as qualifiers, and bit test instructions such as BBC and BBS.

It is also impossible to use bit operation instructions such as CLB and SEB and read/modify/write instructions of direction registers for calculations such as ROR.

For setting direction registers, use the LDM instruction, STA instruction, etc.

A-D Conversion

Do not execute the STP instruction during A-D conversion.

Instruction Execution Timing

The instruction execution time can be obtained by multiplying the frequency of the internal clock ϕ by the number of cycles mentioned in the machine-language instruction table.

The frequency of the internal clock ϕ is the same as that of the X_{IN} in double-speed mode, twice the X_{IN} cycle in high-speed mode and 8 times the X_{IN} cycle in middle-speed mode.

CPU Mode Register

The oscillation mode selection bit and processor mode bits can be rewritten only once after releasing reset. However, after rewriting it is disable to write any value to the bit. (Emulator MCU is excluded.)

When a ceramic oscillation is selected, a double-speed mode of the clock division ratio selection bits can be used. Do not use it when an RC oscillation is selected.

State transition

Do not stop the clock selected as the operation clock because of setting of CM3, 4.

NOTES ON HARDWARE

Handling of Power Source Pin

In order to avoid a latch-up occurrence, connect a capacitor suitable for high frequencies as bypass capacitor between power source pin (Vcc pin) and GND pin (Vss pin). Besides, connect the capacitor to as close as possible. For bypass capacitor which should not be located too far from the pins to be connected, a ceramic capacitor of 0.01 μF to 0.1 μF is recommended.

One Time PROM Version

The CNVss pin is connected to the internal memory circuit block by a low-ohmic resistance, since it has the multiplexed function to be a programmable power source pin (V_{PP} pin) as well.

To improve the noise reduction, connect a track between CNVss pin and Vss pin with 1 to 10 kΩ resistance.

The mask ROM version track of CNVss pin has no operational interference even if it is connected via a resistor.

NOTES ON PERIPHERAL FUNCTIONS

■ Interrupt

When setting the followings, the interrupt request bit may be set to "1".

•When switching external interrupt active edge

Related register: Interrupt edge selection register (address $003A_{16}$)

Timer X mode register (address $2B_{16}$)

Timer A mode register (address $1D_{16}$)

When not requiring the interrupt occurrence synchronized with these setting, take the following sequence.

① Set the corresponding interrupt enable bit to "0" (disabled).

② Set the interrupt edge select bit (active edge switch bit).

③ Set the corresponding interrupt request bit to "0" after 1 or more instructions have been executed.

④ Set the corresponding interrupt enable bit to "1" (enabled).

■ Timers

• When n (0 to 255) is written to a timer latch, the frequency division ratio is 1/(n+1).

• When a count source of timer X, timer Y or timer Z is switched, stop a count of timer X.

■ Timer A

$CNTR_1$ interrupt active edge selection

$CNTR_1$ interrupt active edge depends on the $CNTR_1$ active edge switch bit.

When this bit is "0", the $CNTR_1$ interrupt request bit is set to "1" at the falling edge of the $CNTR_1$ pin input signal. When this bit is "1", the $CNTR_1$ interrupt request bit is set to "1" at the rising edge of the $CNTR_1$ pin input signal.

However, in the pulse width HL continuously measurement mode, $CNTR_1$ interrupt request is generated at both rising and falling edges of $CNTR_1$ pin input signal regardless of the setting of $CNTR_1$ active edge switch bit.

■ Timer X

$CNTR_0$ interrupt active edge selection

$CNTR_0$ interrupt active edge depends on the $CNTR_0$ active edge switch bit.

When this bit is "0", the $CNTR_0$ interrupt request bit is set to "1" at the falling edge of $CNTR_0$ pin input signal. When this bit is "1", the $CNTR_0$ interrupt request bit is set to "1" at the rising edge of $CNTR_0$ pin input signal.

■ Timer Y: Programmable Generation Waveform Mode

• Count set value

In the programmable waveform generation mode, values of TYS, EXPYP, and EXPYS are valid by writing to TYP because the setting to them is executed all at once by writing to TYP. Even when changing TYP is not required, write the same value again.

• Write timing to TYP

In the programmable waveform generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TYP and the timing of timer underflow during the secondary interval simultanesously.

• Usage of waveform extension function

The waveform extension function by the timer Y waveform extension control bit can be used only when "00_{16}" is set to Prescaler Y. When the value other than "00_{16}" is set to Prescaler Y, be sure to set "0" to EXPYP and EXPYS.

• Timer Y write mode

When using this mode, be sure to set "1" to the timer Y write control bit to select "write to latch only".

Timer Y can stop counting by setting "1" to the timer Y count stop bit in any mode.

Also, when Timer Y underflows, the timer Y interrupt request bit is set to "1".

Timer Y reloads the value of latch when counting is stopped by the timer Y count stop bit. (When timer is read out while timer is stopped, the value of latch is read. The value of timer can be read out only while timer is operating.)

■ Timer Z: Programmable Waveform Generation Mode

• Count set value

In the programmable waveform generation mode, values of TZS, EXPZP, and EXPZS are valid by writing to TZP because the setting to them is executed all at once by writing to TZP. Even when changing TZP is not required, write the same value again.

• Write timing to TZP

In the programmable waveform generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow during the secondary interval simultanesously.

• Usage of waveform extension function

The waveform extension function by the timer Z waveform extension control bit can be used only when "00_{16}" is set to Prescaler Z. When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the count source, the waveform extension function cannot be used.

• Timer Z write mode

When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only".

■ Timer Z: Programmable One-shot Generation Mode

• Count set value

In the programmable one-shot generation mode, the value of EXPZP becomes valid by writing to TZP. Even when changing TZP is not required, write the same value again.

• Write timing to TZP

In the programmable one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow simultanesously.

• Usage of waveform extension function

The waveform extension function by the timer Z waveform extension control bit can be used only when "00_{16}" is set to Prescaler Z. When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP. Also, when the timer Y underflow is selected as the count source, the waveform extension function cannot be used.

• Timer Z write mode

When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only".

■ Timer Z: Programmable Wait One-shot Generation Mode

• Count set value

In the programmable wait one-shot generation mode, values of TZS, EXPZP and EXPZS are valid by writing to TZP. Even when changing TZP is not required, write the same value again.

• Write timing to TZP

In the programmable wait one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow during the secondary interval simultanesously.

• Usage of waveform extension function

The waveform extension function by the timer Z waveform extension control bit can be used only when "00_{16}" is set to Prescaler Z. When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the count source, the waveform extension function cannot be used.

• Timer Z write mode

When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only".

Timer Z can stop counting by setting "1" to the timer Z count stop bit in any mode.

Also, when Timer Z underflows, the timer Z interrupt request bit is set to "1".

Timer Z reloads the value of latch when counting is stopped by the timer Z count stop bit. (When timer is read out while timer is stopped, the value of latch is read. The value of timer can be read out only while timer is operating.)

■ Serial I/O

• Serial I/O interrupt

When setting the transmit enable bit to "1", the serial I/O transmit interrupt request bit is automatically set to "1". When not requiring the interrupt occurrence synchronized with the transmission enabled, take the following sequence.

① Set the serial I/O transmit interrupt enable bit to "0" (disabled).
② Set the transmit enable bit to "1".
③ Set the serial I/O transmit interrupt request bit to "0" after 1 or more instructions have been executed.
⑤ Set the serial I/O transmit interrupt enable bit to "1" (enabled).

• I/O pin function when serial I/O1 is enabled.

The functions of $P1_2$ and $P1_3$ are switched with the setting values of a serial I/O1 mode selection bit and a serial I/O1 synchronous clock selection bit as follows.

(1) Serial I/O1 mode selection bit → "1" :
Clock synchronous type serial I/O is selected.
Setup of a serial I/O1 synchronous clock selection bit
"0" : $P1_2$ pin turns into an output pin of a synchronous clock.
"1" : $P1_2$ pin turns into an input pin of a synchronous clock.
Setup of a SRDY1 output enable bit (SRDY)
"0" : $P1_3$ pin can be used as a normal I/O pin.
"1" : $P1_3$ pin turns into a SRDY output pin.

(2) Serial I/O1 mode selection bit → "0" :
Clock asynchronous (UART) type serial I/O is selected.
Setup of a serial I/O1 synchronous clock selection bit
"0": $P1_2$ pin can be used as a normal I/O pin.
"1": $P1_2$ pin turns into an input pin of an external clock.
When clock asynchronous (UART) type serial I/O is selected, it is $P1_3$ pin. It can be used as a normal I/O pin.

■ A-D Converter

The comparator uses internal capacitors whose charge will be lost if the clock frequency is too low.

Make sure that f(X_{IN}) is 500kHz or more during A-D conversion.

■ Notes on clock generating circuit

For use with the oscillation stabilization set bit after release of the STP instruction set to "1", set values in timer 1 and prescaler 1 after fully appreciating the oscillation stabilization time of the oscillator to be used.

• Switch of ceramic and RC oscillations

After releasing reset the operation starts by starting a built-in ring oscillator. Then, a ceramic oscillation or an RC oscillation is selected by setting bit 5 of the CPU mode register.

• Double-speed mode

When a ceramic oscillation is selected, a double-speed mode can be used. Do not use it when an RC oscillation is selected.

• CPU mode register

Bits 5, 1 and 0 of CPU mode register are used to select oscillation mode and to control operation modes of the microcomputer. In order to prevent the dead-lock by error-writing (ex. program run-away), these bits can be rewritten only once after releasing reset. After rewriting it is disable to write any data to the bit. (The emulator MCU "M37540RSS" is excluded.)

Also, when the read-modify-write instructions (SEB, CLB) are executed to bits 2 to 4, 6 and 7, bits 5, 1 and 0 are locked.

• Clock division ratio, X_{IN} oscillation control, ring oscillator control

The state transition shown in Fig. 53 can be performed by setting the clock division ratio selection bits (bits 7 and 6), X_{IN} oscillation control bit (bit 4), ring oscillator oscillation control bit (bit 3) of CPU mode register. Be careful of notes on use in Fig. 53.

DATA REQUIRED FOR MASK ORDERS

The following are necessary when ordering a mask ROM production:

1.Mask ROM Order Confirmation Form *
2.Mark Specification Form *
3.Data to be written to ROM, in EPROM form (three identical copies) or one floppy disk.

DATA REQUIRED FOR ROM PROGRAMMING ORDERS

The following are necessary when ordering a One Time PROM production:

1.ROM Programming Order Confirmation Form *
2.Mark Specification Form *
3.Data to be written to ROM, in EPROM form (three identical copies) or one floppy disk.

* For the mask ROM confirmation ROM programming order confirmation and the mark specifications,
refer to the “Renesas Technology Corp” Homepage
(http://www.renesas.com/en/rom).

ROM PROGRAMMING METHOD

The built-in PROM of the blank One Time PROM version can be read or programmed with a general-purpose PROM programmer using a special programming adapter. Set the address of PROM programmer in the user ROM area.

Table 7 Special programming adapter

Package	Name of Programming Adapter
32P4B	PCA7435SPG02
32P6U-A	PCA7435GPG03
36P2R-A	PCA7435FPG02

The PROM of the blank One Time PROM version is not tested or screened in the assembly process and following processes. To ensure proper operation after programming, the procedure shown in Figure 54 is recommended to verify programming.

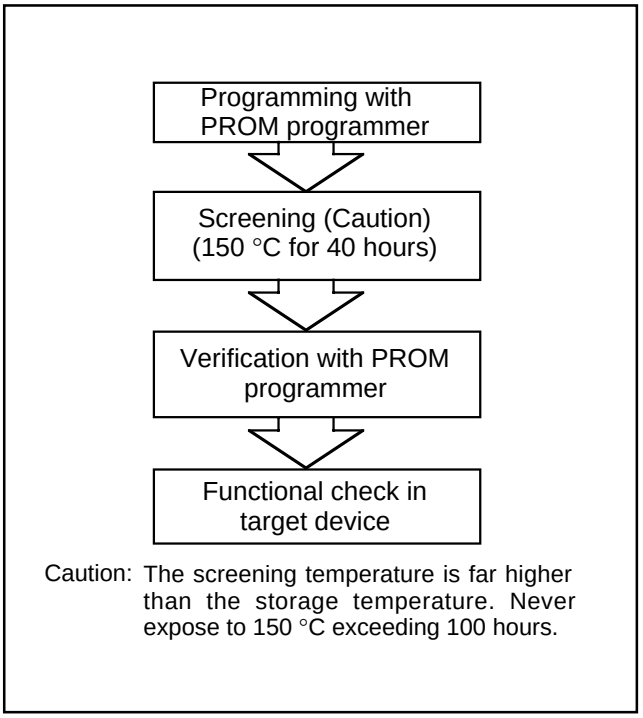


Fig. 54 Programming and testing of One Time PROM version

FUNCTIONAL DESCRIPTION SUPPLEMENT

Interrupt

7540 group permits interrupts on the 14 sources for 42-pin version, 13 sources for 36-pin version and 12 sources for 32-pin version. It is vector interrupts with a fixed priority system. Accordingly, when two or more interrupt requests occur during the same sampling, the higher-priority interrupt is accepted first. This priority is determined by hardware, but variety of priority processing can be performed by software, using an interrupt enable bit and an interrupt disable flag.

For interrupt sources, vector addresses and interrupt priority, refer to **"Table 8."**

Table 8 Interrupt sources, vector addresses and interrupt priority

Interrupt source	Priority	Vector addresses (Note 1)		Interrupt request generating conditions	Remarks
		High-order	Low-order		
Reset (Note 2)	1	$FFFD_{16}$	$FFFC_{16}$	At reset input	Non-maskable
Serial I/O1 receive	2	$FFFB_{16}$	$FFFA_{16}$	At completion of serial I/O1 data receive	Valid only when serial I/O1 is selected
Serial I/O1 transmit	3	$FFF9_{16}$	$FFF8_{16}$	At completion of serial I/O1 transmit shift or when transmit buffer is empty	Valid only when serial I/O1 is selected
INT_0	4	$FFF7_{16}$	$FFF6_{16}$	At detection of either rising or falling edge of INT_0 input	External interrupt (active edge selectable)
INT_1 (Note 3)	5	$FFF5_{16}$	$FFF4_{16}$	At detection of either rising or falling edge of INT_1 input	External interrupt (active edge selectable)
Key-on wake-up	6	$FFF3_{16}$	$FFF2_{16}$	At falling of conjunction of input logical level for port P0 (at input)	External interrupt (valid at falling)
$CNTR_0$	7	$FFF1_{16}$	$FFF0_{16}$	At detection of either rising or falling edge of $CNTR_0$ input	External interrupt (active edge selectable)
$CNTR_1$	8	$FFEF_{16}$	$FFEE_{16}$	At detection of either rising or falling edge of $CNTR_1$ input	External interrupt (active edge selectable)
Timer X	9	$FFED_{16}$	$FFEC_{16}$	At timer X underflow	
Timer Y	10	$FFEB_{16}$	$FFEA_{16}$	At timer Y underflow	
Timer Z	11	$FFE9_{16}$	$FFE8_{16}$	At timer Z underflow	
Timer A	12	$FFE7_{16}$	$FFE6_{16}$	At timer A underflow	
Serial I/O2	13	$FFE5_{16}$	$FFE4_{16}$	At completion of transmit/receive shift	
A-D conversion	14	$FFE3_{16}$	$FFE2_{16}$	At completion of A-D conversion	
Timer 1	15	$FFE1_{16}$	$FFE0_{16}$	At timer 1 underflow	STP release timer underflow
Reserved area	16	$FFDF_{16}$	$FFDE_{16}$	Not available	
BRK instruction	17	$FFDD_{16}$	$FFDC_{16}$	At BRK instruction execution	Non-maskable software interrupt

Note 1: Vector addressed contain internal jump destination addresses.
2: Reset function in the same way as an interrupt with the highest priority.
3: It is an interrupt which can use only for 36 pin version.

Timing After Interrupt

The interrupt processing routine begins with the machine cycle following the completion of the instruction that is currently in execution.
Figure 55 shows a timing chart after an interrupt occurs, and Figure 56 shows the time up to execution of the interrupt processing routine.

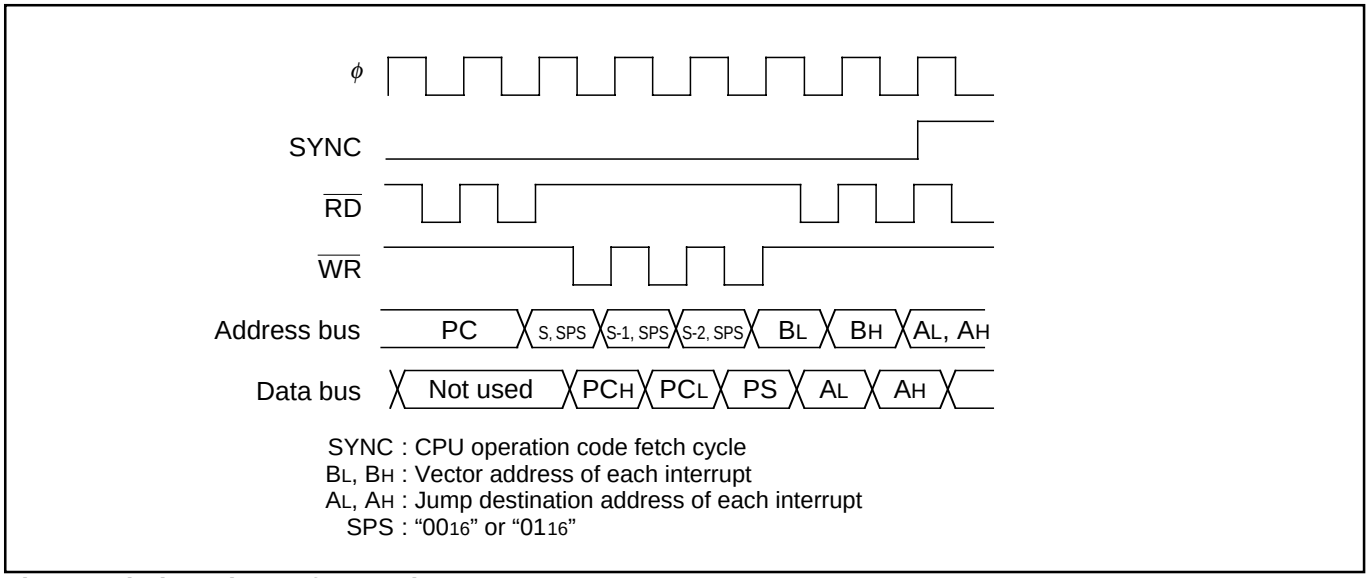


Fig. 55 Timing chart after an interrupt occurs

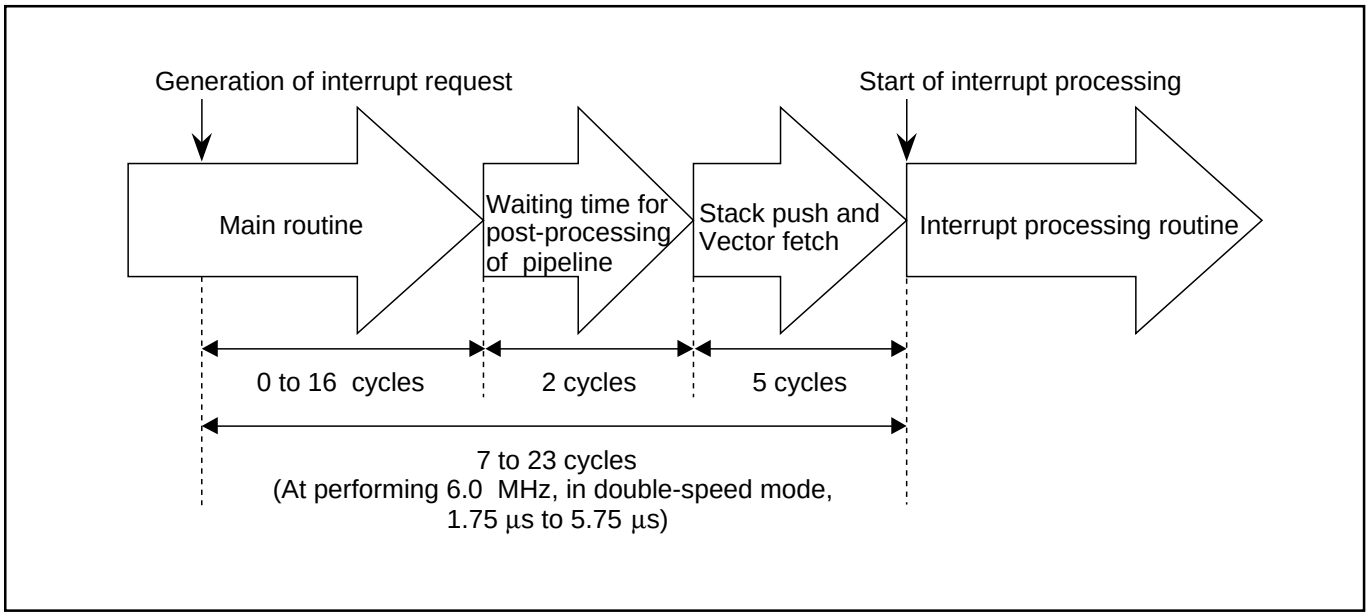


Fig. 56 Time up to execution of the interrupt processing routine

A-D Converter

A-D conversion is started by setting AD conversion completion bit to "0." During A-D conversion, internal operations are performed as follows.

1. After the start of A-D conversion, A-D conversion register goes to "00_{16}."
2. The highest-order bit of A-D conversion register is set to "1," and the comparison voltage Vref is input to the comparator. Then, Vref is compared with analog input voltage V_{IN}.
3. As a result of comparison, when Vref < V_{IN}, the highest-order bit of A-D conversion register becomes "1." When Vref > V_{IN}, the highest-order bit becomes "0."

By repeating the above operations up to the lowest-order bit of the A-D conversion register, an analog value converts into a digital value.

A-D conversion completes at 122 clock cycles (20.34 µs at f(X_{IN}) = 6.0 MHz) after it is started, and the result of the conversion is stored into the A-D conversion register.

Concurrently with the completion of A-D conversion, A-D conversion interrupt request occurs, so that the AD conversion interrupt request bit is set to "1."

Relative formula for a reference voltage V_{REF} of A-D converter and Vref

When n = 0 Vref = 0

When n = 1 to 1023 $Vref = \frac{V_{REF}}{1024} \times n$

n : the value of A-D converter (decimal numeral)

Table 9 Change of A-D conversion register during A-D conversion

	Change of A-D conversion register	Value of comparison voltage (Vref)
At start of conversion	0 0 0 0 0 0 0 0 0 0	0
First comparison	1 0 0 0 0 0 0 0 0 0	$\frac{V_{REF}}{2}$
Second comparison	*1 1 0 0 0 0 0 0 0 0	$\frac{V_{REF}}{2} \pm \frac{V_{REF}}{4}$
Third comparison	*1 *2 1 0 0 0 0 0 0 0	$\frac{V_{REF}}{2} \pm \frac{V_{REF}}{4} \pm \frac{V_{REF}}{8}$
⋮	⋮	⋮
After completion of tenth comparison	A result of A-D conversion *1 *2 *3 *4 *5 *6 *7 *8 *9 *10	$\frac{V_{REF}}{2} \pm \frac{V_{REF}}{4} \pm \cdots \pm \frac{V_{REF}}{1024}$

∗1–∗10: A result of the first to tenth comparison

Figure 56 shows A-D conversion equivalent circuit, and Figure 57 shows A-D conversion timing chart.

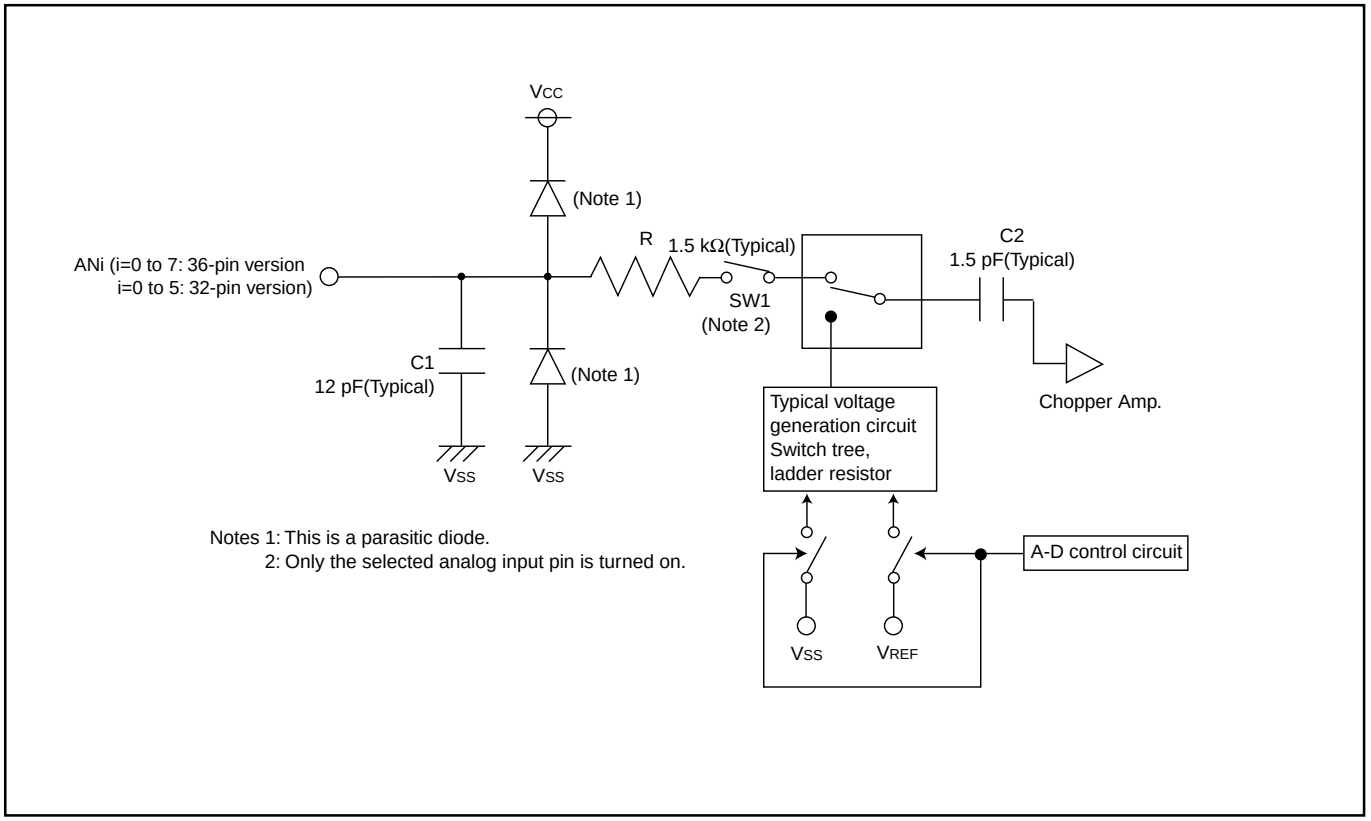


Fig. 57 A-D conversion equivalent circuit

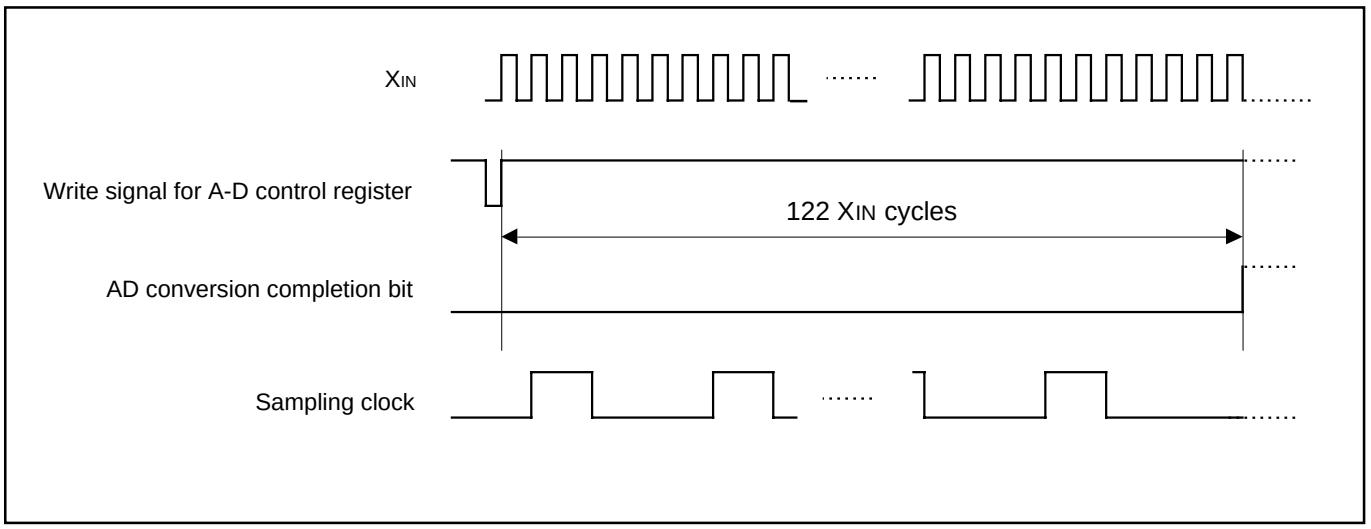


Fig. 58 A-D conversion timing chart

CHAPTER 2

APPLICATION

2.1 I/O port

This paragraph explains the registers setting method and the notes relevant to the I/O ports.

2.1.1 Memory map

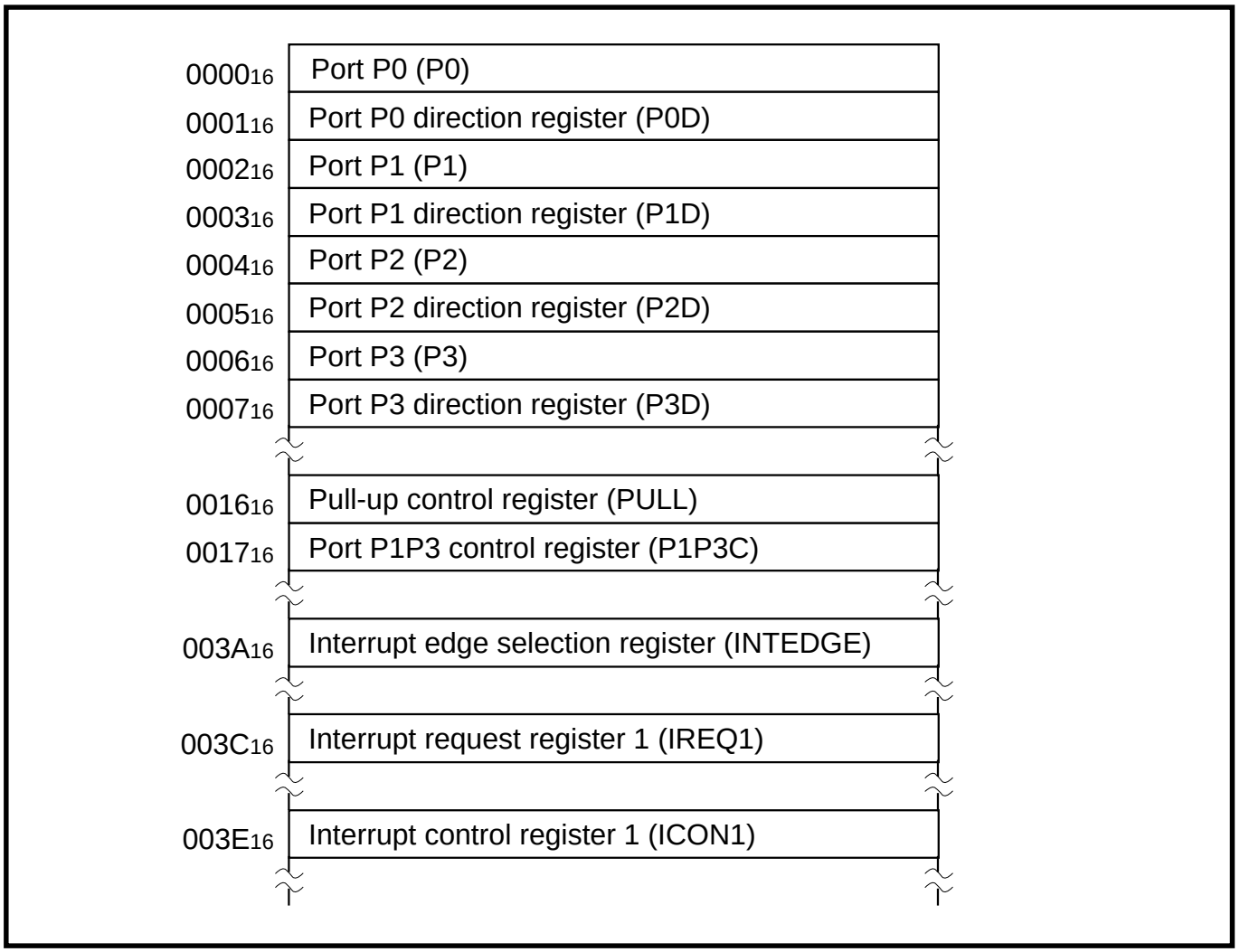


Fig. 2.1.1 Memory map of registers relevant to I/O port

2.1.2 Relevant registers

Port Pi

b7 b6 b5 b4 b3 b2 b1 b0

Port Pi (Pi) (i = 0, 2, 3) [Address : 00_{16}, 04_{16}, 06_{16}]

B	Name	Function	At reset	R	W
0	Port Pi_0	• In output mode Write } Port latch Read } • In input mode Write : Port latch Read : Value of pins	?	○	○
1	Port Pi_1		?	○	○
2	Port Pi_2		?	○	○
3	Port Pi_3		?	○	○
4	Port Pi_4		?	○	○
5	Port Pi_5		?	○	○
6	Port Pi_6		?	○	○
7	Port Pi_7		?	○	○

Note: The 32-pin package versions have nothing to be allocated for the following:
•Bits 6 and 7 of port P2
•Bits 5 and 6 of port P3.

Fig. 2.1.2 Structure of Port Pi (i = 0, 2, 3)

Port P1

b7 b6 b5 b4 b3 b2 b1 b0

Port P1 (P1) [Address : 02_{16}]

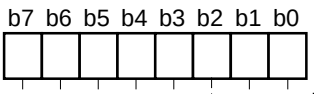
B	Name	Function	At reset	R	W
0	Port $P1_0$	• In output mode Write } Port latch Read } • In input mode Write : Port latch Read : Value of pins	?	○	○
1	Port $P1_1$		?	○	○
2	Port $P1_2$		?	○	○
3	Port $P1_3$		?	○	○
4	Port $P1_4$		?	○	○
5	Nothing is allocated for these bits. When these bits are read out, the values are undefined.		?	×	×
6			?	×	×
7			?	×	×

Fig. 2.1.3 Structure of Port P1

APPLICATION

2.1 I/O port

Port Pi direction register



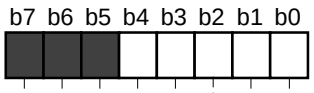
Port Pi direction register (PiD) (i = 0, 2, 3) [Address : 01 16, 0516, 0716]

B	Name	Function	At reset	R	W
0	Port Pi direction register	0 : Port Pi_0 input mode 1 : Port Pi_0 output mode	0	×	○
1		0 : Port Pi_1 input mode 1 : Port Pi_1 output mode	0	×	○
2		0 : Port Pi_2 input mode 1 : Port Pi_2 output mode	0	×	○
3		0 : Port Pi_3 input mode 1 : Port Pi_3 output mode	0	×	○
4		0 : Port Pi_4 input mode 1 : Port Pi_4 output mode	0	×	○
5		0 : Port Pi_5 input mode 1 : Port Pi_5 output mode	0	×	○
6		0 : Port Pi_6 input mode 1 : Port Pi_6 output mode	0	×	○
7		0 : Port Pi_7 input mode 1 : Port Pi_7 output mode	0	×	○

Note: The 32-pin package versions have nothing to be allocated for the following:
- Bits 6 and 7 of P2D
- Bits 5 and 6 of P3D.

Fig. 2.1.4 Structure of Port Pi direction register (i = 0, 2, 3)

Port P1 direction register

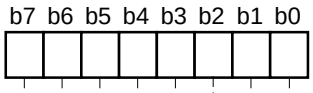


Port P1 direction register (P1D) [Address : 03 16]

B	Name	Function	At reset	R	W
0	Port P1 direction register	0 : Port $P1_0$ input mode 1 : Port $P1_0$ output mode	0	×	○
1		0 : Port $P1_1$ input mode 1 : Port $P1_1$ output mode	0	×	○
2		0 : Port $P1_2$ input mode 1 : Port $P1_2$ output mode	0	×	○
3		0 : Port $P1_3$ input mode 1 : Port $P1_3$ output mode	0	×	○
4		0 : Port $P1_4$ input mode 1 : Port $P1_4$ output mode	0	×	○
5	Nothing is allocated for these bits. When these bits are read out, the values are undefined.		?	×	×
6			?	×	×
7			?	×	×

Fig. 2.1.5 Structure of Port P1 direction register

Pull-up control register



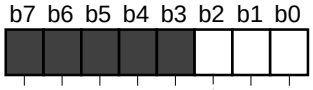
Pull-up control register (PULL) [Address : 16_{16}]

B	Name	Function	At reset	R	W
0	$P0_0$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
1	$P0_1$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
2	$P0_2$, $P0_3$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
3	$P0_4$ – $P0_7$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
4	$P3_0$ – $P3_3$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
5	$P3_4$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
6	$P3_5$, $P3_6$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
7	$P3_7$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○

Notes 1: Pins set to output are disconnected from the pull-up control.
2: Keep setting the $P3_5$, $P3_6$ pull-up control bit to "1" (initial value: 0) for the 32-pin package versions.

Fig. 2.1.6 Structure of Pull-up control register

Port P1P3 control register



Port P1P3 control register (P1P3C) [Address : 17_{16}]

B	Name	Function	At reset	R	W
0	$P3_7$/INT_0 input level selection bit	0 : CMOS level 1 : TTL level	0	○	○
1	$P3_6$/INT_1 input level selection bit (**Note**)	0 : CMOS level 1 : TTL level	0	○	○
2	$P1_0$, $P1_2$,$P1_3$ input level selection bit	0 : CMOS level 1 : TTL level	0	○	○
3	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Note: Keep setting the $P3_6$/INT_1 input level selection bit to "0" (initial value) for the 32-pin package version.

Fig. 2.1.7 Structure of Port P1P3 control register

APPLICATION

2.1 I/O port

Interrupt edge selection register

b7 b6 b5 b4 b3 b2 b1 b0

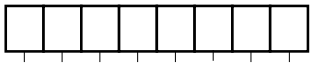
Interrupt edge selection register (INTEDGE) [Address : 3A $_{16}$]

B	Name	Function	At reset	R	W
0	INT_0 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
1	INT_1 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
2	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
3			0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7	$P0_0$ key-on wakeup enable bit	0 : Key-on wakeup enabled 1 : Key-on wakeup disabled	0	○	○

Fig. 2.1.8 Structure of Interrupt edge selection register

Interrupt request register 1

b7 b6 b5 b4 b3 b2 b1 b0



Interrupt request register 1 (IREQ1) [Address : 3C $_{16}$]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	*
1	Serial I/O1 transmit interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	*
2	INT_0 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	*
3	INT_1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	*
4	Key-on wake up interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	*
5	$CNTR_0$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	*
6	$CNTR_1$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	*
7	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	*

***:** These bits can be cleared to "0" by program, but cannot be set to "1".

Fig. 2.1.9 Structure of Interrupt request register 1

Interrupt control register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 1 (ICON1) [Address : 3E 16]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Serial I/O1 transmit interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	INT_0 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	INT_1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Key-on wake up interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	$CNTR_0$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	$CNTR_1$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Fig. 2.1.10 Structure of Interrupt control register 1

2.1.3 Application example of key-on wake up (1)

Outline: The built-in pull-up resistor is used.

Specifications: System is returned from the wait mode when the key-on wakeup interrupt occurs by input of the falling edge to port P0i.

Note: Only the falling edge is active for the key-on wakeup interrupt.

Figure 2.1.11 shows an example of application circuit, and Figure 2.1.12 shows an example of control procedure.

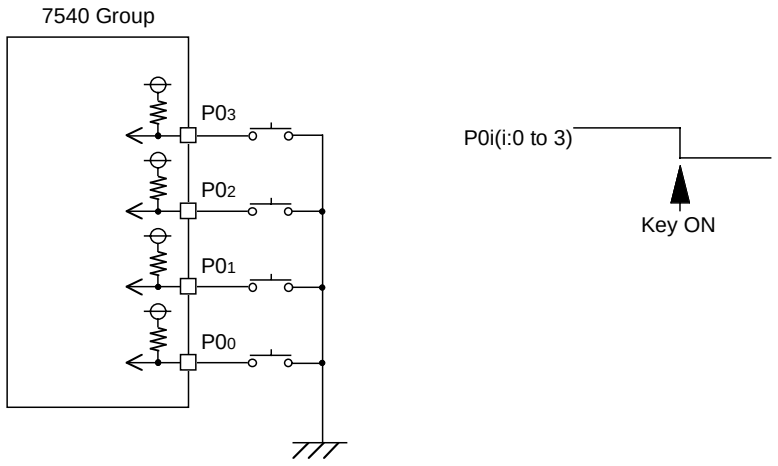


Fig. 2.1.11 Example of application circuit

2.1 I/O port

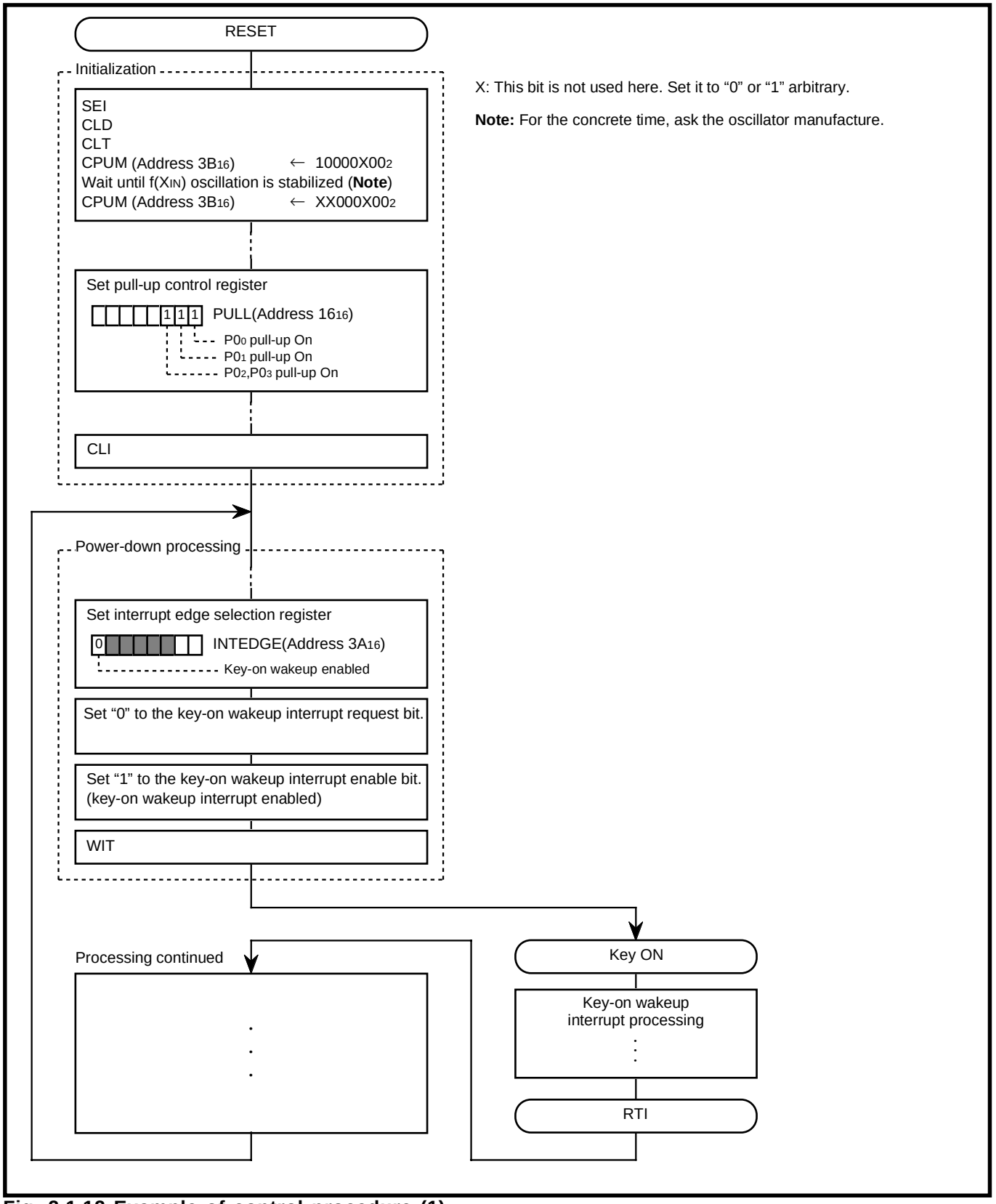


Fig. 2.1.12 Example of control procedure (1)

2.1.4 Application example of key-on wake up (2)

Outline: The key-on wakeup interrupt is used as the normal external interrupt.

Specifications: The key-on wakeup interrupt occurs by input of the falling edge to port P0i. If necessary, the built-in pull-up resistor is used.

Note: Only the falling edge is active for the key-on wakeup interrupt.

Figure 2.1.13 shows an example of control procedure.

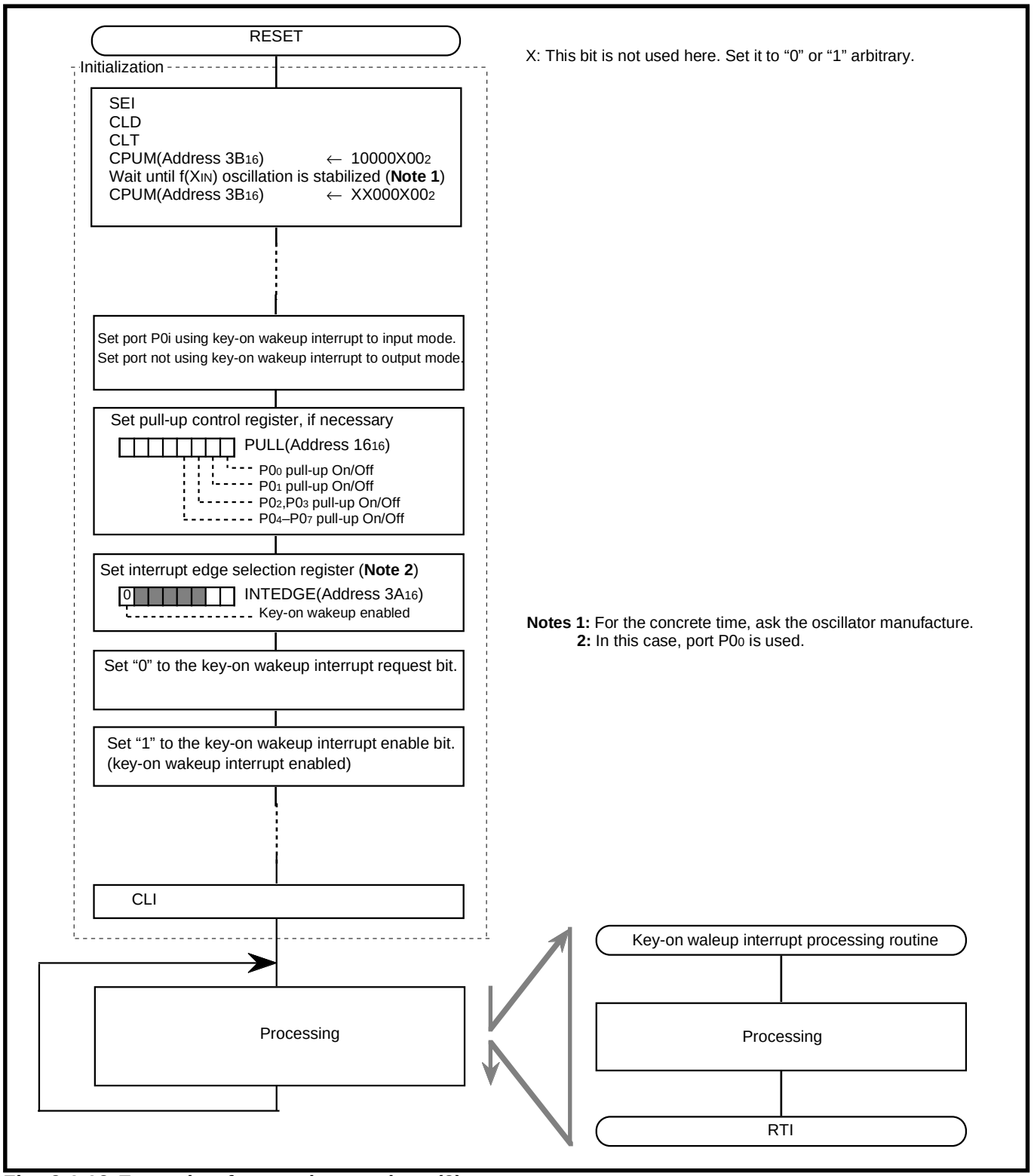


Fig. 2.1.13 Example of control procedure (2)

2.1.5 Handling of unused pins

Table 2.1.1 Handling of unused pins

Pins/Ports name	Handling
P0, P1, P2, P3	•Set to the input mode and connect each to Vcc or Vss through a resistor of 1 kΩ to 10 kΩ. •Set to the output mode and open at “L” or “H” level.
V_{REF}	•Connect to Vss (GND).
X_{IN}	•Connect to V_{SS} (GND) when using a ring oscillator for main clock.
X_{OUT}	•Open when using an external clock. •Open when using a ring oscillator for main clock.

2.1.6 Notes on input and output ports

Notes on using input and output ports are described below.

(1) Notes in stand-by state

In stand-by state*1 for low-power dissipation, do not make input levels of an input port and an I/O port "undefined".

Pull-up (connect the port to V_{CC}) or pull-down (connect the port to V_{SS}) these ports through a resistor.

When determining a resistance value, note the following points:

- External circuit
- Variation of output levels during the ordinary operation

When using a built-in pull-up resistor, note on varied current values:

- When setting as an input port : Fix its input level
- When setting as an output port : Prevent current from flowing out to external.

● **Reason**

The output transistor becomes the OFF state, which causes the ports to be the high-impedance state. Note that the level becomes "undefined" depending on external circuits.

Accordingly, the potential which is input to the input buffer in a microcomputer is unstable in the state that input levels of a input port and an I/O port are "undefined". This may cause power source current.

*1 stand-by state : the stop mode by executing the **STP** instruction
the wait mode by executing the **WIT** instruction

(2) Modifying output data with bit managing instruction

When the port latch of an I/O port is modified with the bit managing instruction*2, the value of the unspecified bit may be changed.

● **Reason**

The bit managing instructions are read-modify-write form instructions for reading and writing data by a byte unit. Accordingly, when these instructions are executed on a bit of the port latch of an I/O port, the following is executed to all bits of the port latch.

- As for a bit which is set for an input port :
  The pin state is read in the CPU, and is written to this bit after bit managing.
- As for a bit which is set for an output port :
  The bit value of the port latch is read in the CPU, and is written to this bit after bit managing.

Note the following :

- Even when a port which is set as an output port is changed for an input port, its port latch holds the output data.
- As for a bit of the port latch which is set for an input port, its value may be changed even when not specified with a bit managing instruction in case where the pin state differs from its port latch contents.

*2 bit managing instructions : **SEB**, and **CLB** instructions

(3) Usage for the 32-pin version

① Fix the $P3_5$, $P3_6$ pull-up control bit of the pull-up control register to "1".
② Keep the $P3_6/INT_1$ input level selection bit of the port P1P3 control register "0" (initial state).

2.1 I/O port

2.1.7 Termination of unused pins

(1) Terminate unused pins

① I/O ports :
- Set the I/O ports for the input mode and connect them to V_{CC} or V_{SS} through each resistor of 1 kΩ to 10 kΩ.
  Ports that permit the selecting of a built-in pull-up resistor can also use this resistor. Set the I/O ports for the output mode and open them at "L" or "H".
- When opening them in the output mode, the input mode of the initial status remains until the mode of the ports is switched over to the output mode by the program after reset. Thus, the potential at these pins is undefined and the power source current may increase in the input mode. With regard to an effects on the system, thoroughly perform system evaluation on the user side.
- Since the direction register setup may be changed because of a program runaway or noise, set direction registers by program periodically to increase the reliability of program.

(2) Termination remarks

① Input ports and I/O ports :
Do not open in the input mode.

● **Reason**
- The power source current may increase depending on the first-stage circuit.
- An effect due to noise may be easily produced as compared with proper termination ② and ③ shown on the above.

② I/O ports :
When setting for the input mode, do not connect to V_{CC} or V_{SS} directly.

● **Reason**
If the direction register setup changes for the output mode because of a program runaway or noise, a short circuit may occur between a port and V_{CC} (or V_{SS}).

③ I/O ports :
When setting for the input mode, do not connect multiple ports in a lump to V_{CC} or V_{SS} through a resistor.

● **Reason**
If the direction register setup changes for the output mode because of a program runaway or noise, a short circuit may occur between ports.

- At the termination of unused pins, perform wiring at the shortest possible distance (20 mm or less) from microcomputer pins.

2.2 Timer A

This paragraph explains the registers setting method and the notes relevant to the timer A.

2.2.1 Memory map

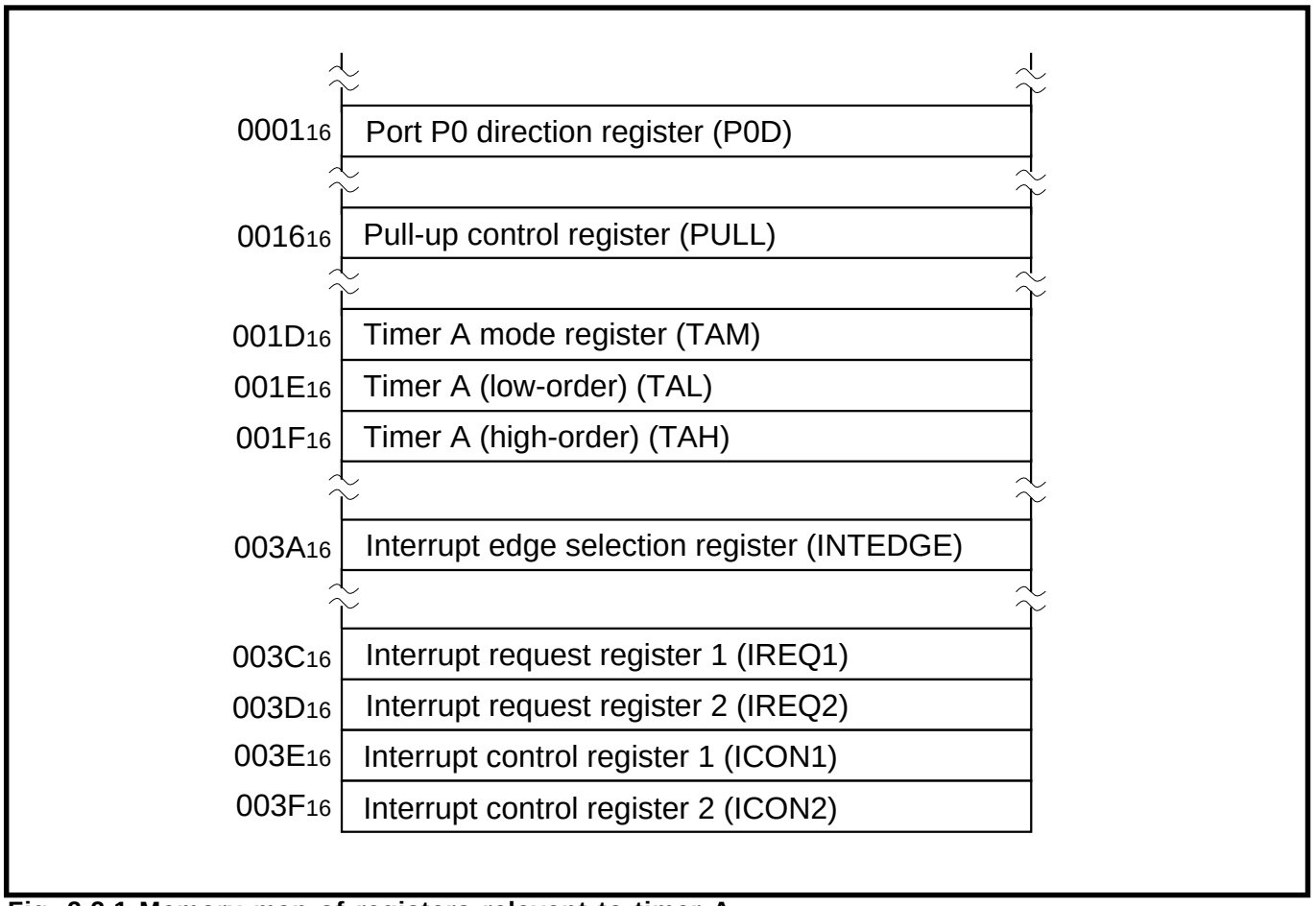


Fig. 2.2.1 Memory map of registers relevant to timer A

2.2.2 Relevant registers

Port P0 direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port P0 direction register (P0D) [Address : 01 16]

B	Name	Function	At reset	R	W
0	Port P0 direction register	0 : Port Pi_0 input mode 1 : Port Pi_0 output mode	0	×	○
1		0 : Port Pi_1 input mode 1 : Port Pi_1 output mode	0	×	○
2		0 : Port Pi_2 input mode 1 : Port Pi_2 output mode	0	×	○
3		0 : Port Pi_3 input mode 1 : Port Pi_3 output mode	0	×	○
4		0 : Port Pi_4 input mode 1 : Port Pi_4 output mode	0	×	○
5		0 : Port Pi_5 input mode 1 : Port Pi_5 output mode	0	×	○
6		0 : Port Pi_6 input mode 1 : Port Pi_6 output mode	0	×	○
7		0 : Port Pi_7 input mode 1 : Port Pi_7 output mode	0	×	○

Fig. 2.2.2 Structure of Port P0 direction register

Pull-up control register

b7 b6 b5 b4 b3 b2 b1 b0

Pull-up control register (PULL) [Address : 16 16]

B	Name	Function	At reset	R	W
0	$P0_0$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
1	$P0_1$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
2	$P0_2$, $P0_3$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
3	$P0_4$ – $P0_7$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
4	$P3_0$ – $P3_3$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
5	$P3_4$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
6	$P3_5$, $P3_6$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
7	$P3_7$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○

Notes 1: Pins set to output are disconnected from the pull-up control.

2: Keep setting the $P3_5$, $P3_6$ pull-up control bit to “1” (initial value: 0) for the 3 2-pin package versions.

Fig. 2.2.3 Structure of Pull-up control register

Timer A mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer A mode register (TAM) [Address : 1D 16]

B	Name	Function	At reset	R	W
0	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
1			0	○	×
2			0	○	×
3			0	○	×
4	Timer A operating mode bits	b5 b4 0 0 : Timer mode 0 1 : Period measurement mode 1 0 : Event counter mode 1 1 : Pulse width HL continuously measurement mode	0	○	○
5			0	○	○
6	$CNTR_1$ active edge switch bit	The function depends on the operating mode. (Refer to Table 2.2.1)	0	○	○
7	Timer A count stop bit	0 : Count start 1 : Count stop	0	○	○

Fig. 2.2.4 Structure of Timer A mode register

Table 2.2.1 $CNTR_1$ active edge switch bit function

Timer A operating modes	$CNTR_1$ active edge switch bit (bit 6 of address $1D_{16}$) contents	
Timer mode	"0"	$CNTR_1$ interrupt request occurrence: Falling edge ; No influence to timer count
	"1"	$CNTR_1$ interrupt request occurrence: Rising edge ; No influence to timer count
Period measurement mode	"0"	Period measurement: Falling period measurement $CNTR_1$ interrupt request occurrence: Falling edge
	"1"	Period measurement: Rising period measurement $CNTR_1$ interrupt request occurrence: Rising edge
Event counter mode	"0"	Timer A: Rising edge count $CNTR_1$ interrupt request occurrence: Falling edge
	"1"	Timer A: Falling edge count $CNTR_1$ interrupt request occurrence: Rising edge
Pulse width HL continuously measurement mode	"0"	$CNTR_1$ interrupt request occurrence: Rising edge and Falling edge
	"1"	$CNTR_1$ interrupt request occurrence: Rising edge and Falling edge

Timer A register (low-order, high-order)

b7 b6 b5 b4 b3 b2 b1 b0

Timer A register (low-order, high-order) (TAL, TAH) [Address : 1E 16, 1F16]

B	Function	At reset	R	W
0	•Set a count value of timer A. •The value set in this register is written to both timer A and timer A latch at the same time. •When this register is read out, the timer A's count value is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Notes 1: Be sure to write to/read out both the low-order of timer A (TAL) and the high-order of timer A (TAH).
2: Read the high-order of timer A (TAH) first, and the high-order of timer A (TAL) next.
3: Write to the low-order of timer A (TAL) first, and the high-order of timer A (TAH) next.
4: Do not write to them during read, and do not read out them during write.

Fig. 2.2.5 Structure of Timer A register

Interrupt edge selection register

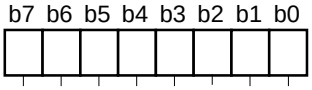
b7 b6 b5 b4 b3 b2 b1 b0

Interrupt edge selection register (INTEDGE) [Address : 3A 16]

B	Name	Function	At reset	R	W
0	INT_0 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
1	INT_1 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
2	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
3			0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7	$P0_0$ key-on wakeup enable bit	0 : Key-on wakeup enabled 1 : Key-on wakeup disabled	0	○	○

Fig. 2.2.6 Structure of Interrupt edge selection register

Interrupt request register 1



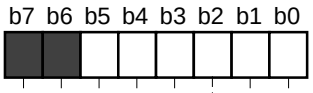
Interrupt request register 1 (IREQ1) [Address : 3C 16]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Serial I/O1 transmit interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	INT_0 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	INT_1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	Key-on wake up interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	$CNTR_0$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	$CNTR_1$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻

✻: These bits can be cleared to “0” by program, but cannot be set to “1”.

Fig. 2.2.7 Structure of Interrupt request register 1

Interrupt request register 2



Interrupt request register 2 (IREQ2) [Address : 3D 16]

B	Name	Function	At reset	R	W
0	Timer Y interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Timer Z interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	Timer A interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	AD converter interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are “0”.		0	○	×
7			0	○	×

✻: These bits can be cleared to “0” by program, but cannot be set to “1”.

Fig. 2.2.8 Structure of Interrupt request register 2

Interrupt control register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 1 (ICON1) [Address : 3E 16]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Serial I/O1 transmit interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	INT_0 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	INT_1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Key-on wake up interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	$CNTR_0$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	$CNTR_1$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Fig. 2.2.9 Structure of Interrupt control register 1

Interrupt control register 2

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 2 (ICON2) [Address : 3F 16]

B	Name	Function	At reset	R	W
0	Timer Y interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer Z interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer A interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	AD conversion interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
7			0	○	×

Fig. 2.2.10 Structure of Interrupt control register 2

2.2.3 Timer mode

(1) Operation description

Timer A counts the oscillation frequency divided by 16. Each time the count clock is input, the contents of Timer A is decremented by 1. When the contents of Timer A reach "0000_{16}", an underflow occurs at the next count clock, and the timer A latch is reloaded into Timer A. The division ratio of Timer A is 1/(n+1) provided that the value of Timer A is n.

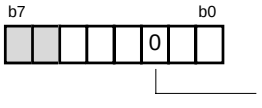
Timer A can stop counting by setting "1" to the timer A count stop bit.

Also, when Timer A underflows, the timer A interrupt request bit is set to "1".

(2) Timer mode setting method

Figure 2.2.11 shows the setting method for timer mode of timer A.

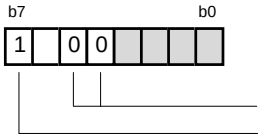
Process 1: Disable timer A interrupt.



Interrupt control register 2 (ICON2) [Address $3F_{16}$]

Timer A interrupt disabled

Process 2: Set timer A mode register.

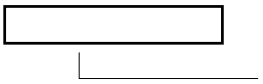


Timer A mode register (TAM) [Address $1D_{16}$]

Timer mode
Timer A count stop

Process 3: Set the count value to Timer A (**Note**).

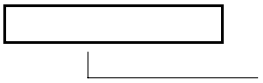
• Set the count value to timer A (low-order)



Timer A (low-order) (TAL) (Address $1E_{16}$)

Count value

• Set the count value to timer A (high-order)

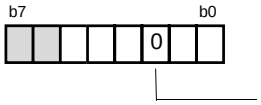


Timer A (high-order) (TAH) (Address $1F_{16}$)

Count value

Note: Write both registers in order of timer X (low-order) and timer X (high-order) following, certainly.

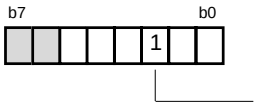
Process 4: In order not to execute the no requested interrupt processing, set “0” (no requested) to the timer A interrupt request bit.



Interrupt request register 2 (IREQ2) [Address $3D_{16}$]

No timer A interrupt request issued

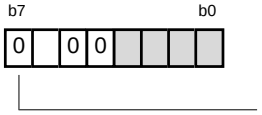
Process 5: When Timer A interrupt is used, set “1” (interrupt enabled) to the timer A interrupt enable bit.



Interrupt control register 2 (ICON2) [Address $3F_{16}$]

Timer A interrupt enabled

Process 6: Start counting of Timer A.



Timer A mode register (TAM) [Address $1D_{16}$]

Timer A count start

Fig. 2.2.11 Setting method for timer mode

(3) Application example of timer mode

Outline: The input clock is divided by the timer so that the period processing is executed every 25 ms intervals.

Specifications: •The f(X_{IN}) = 8 MHz is divided by timer A to detect 25 ms.
•The timer A interrupt request is confirmed in the main routine. When 25 ms has elapsed, the period processing is executed in the timer A interrupt processing routine.
• Operation clock: f(X_{IN}) = 8 MHz, high-speed mode

Figure 2.2.12 shows an example of control procedure.

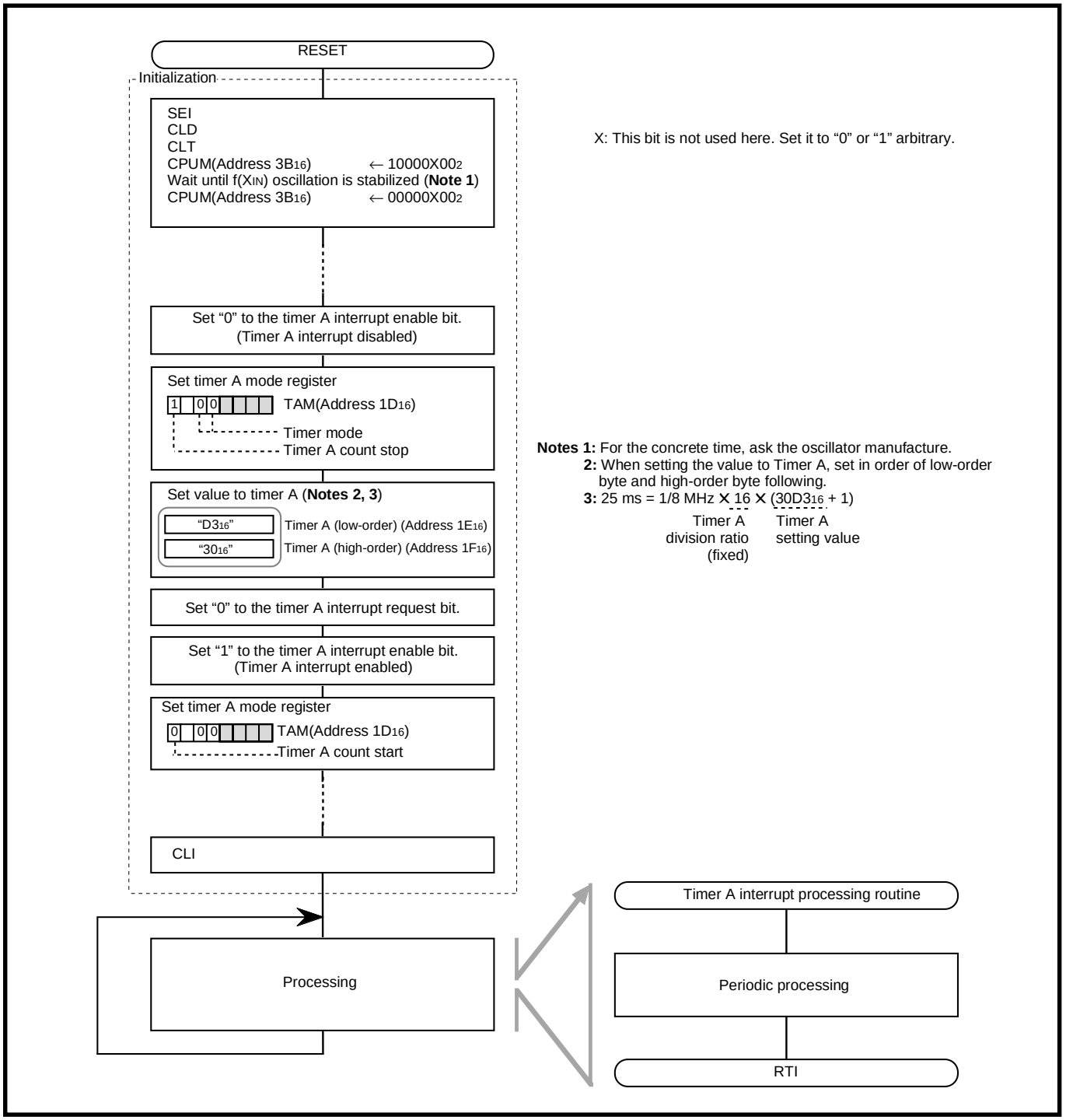


Fig. 2.2.12 Example of control procedure

2.2 Timer A

2.2.4 Period measurement mode

(1) Operation description

In the period measurement mode, the pulse period input from the $P0_0/CNTR_1$ pin is measured. $CNTR_1$ interrupt request is generated at rising/falling edge of $CNTR_1$ pin input signal. Simultaneously, the value in the timer A latch is reloaded in Timer A and count continues. The active edge of $CNTR_1$ pin input signal can be selected from rising or falling by the $CNTR_1$ active edge switch bit. The count value when trigger input from $CNTR_1$ pin is accepted is retained until Timer A is read once.
Timer A can stop counting by setting "1" to the timer A count stop bit.
Also, when Timer A underflows, the timer A interrupt request bit is set to "1".

(2) Period measurement mode setting method

Figure 2.2.13 and Figure 2.2.14 show the setting method for period measurement mode of timer A.

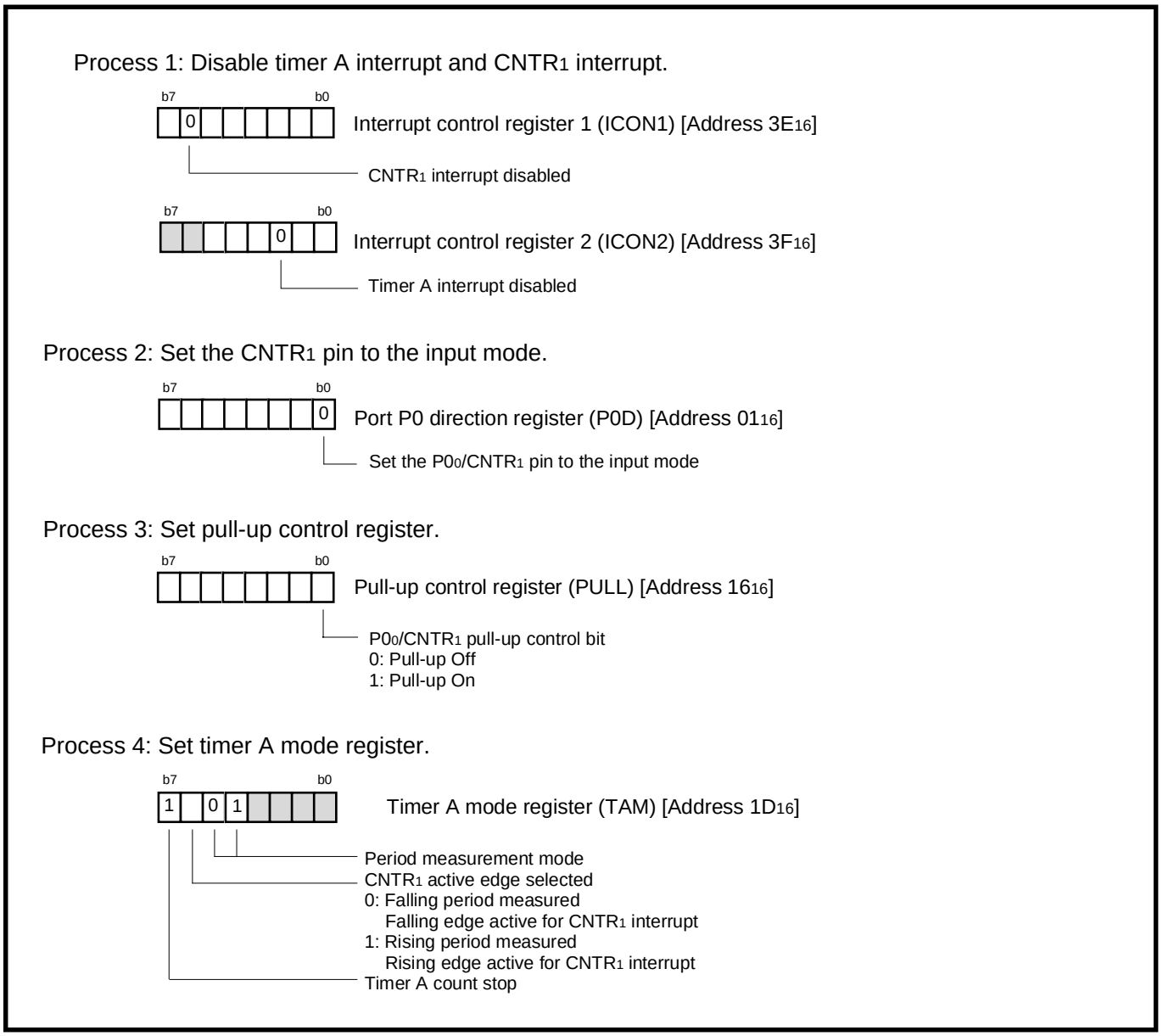
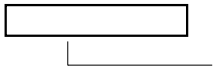


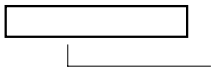
Fig. 2.2.13 Setting method for period measurement mode (1)

Process 5: Set the count value to timer A (**Note**).

• Set the initial value to timer A (low-order)

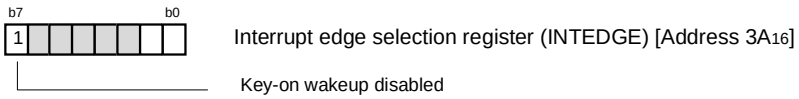


• Set the initial value to timer A (high-order)

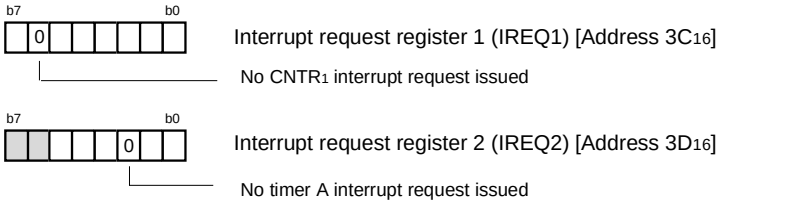


Note: Write both registers in order of timer X (low-order) and timer X (high-order) following, certainly.

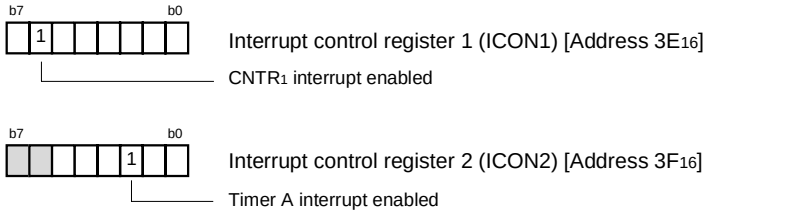
Process 6: In order to use the $CNTR_1$ pin function of the $P0_0/CNTR_1$ pin, disable the $P0_0$ key-on wakeup function.



Process 7: In order not to execute the no requested interrupt processing, set “0” (no requested) to the timer A interrupt request bit and $CNTR_1$ interrupt request bit.



Process 8: When the interrupt is used, set “1” (interrupt enabled) to the timer A interrupt enable bit or $CNTR_1$ interrupt enable bit.



Process 9: Start counting of timer A.

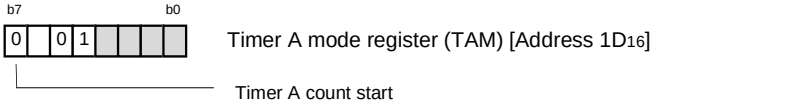


Fig. 2.2.14 Setting method for period measurement mode (2)

(3) Application example of period measurement mode

Outline: The phase control signal is adjusted by using the period measurement mode.

Specifications: • The phase control signal is output to a load, and that controls the phase of a load.
• The period of the pulse input to the $P0_0/CNTR_1$ pin from the load as a feedback signal is measured. The correct of the phase control signal to the load is executed using this result. The input pulse period is set to be less than the period of timer A. When timer A underflows, the period is recognized as not corrected, and error processing is executed in the timer A interrupt processing routine.
• Operation clock: $f(X_{IN})$ = 8 MHz, high-speed mode

Figure 2.2.15 shows an example of a peripheral circuit, and Figure 2.2.16 shows an example of control procedure.

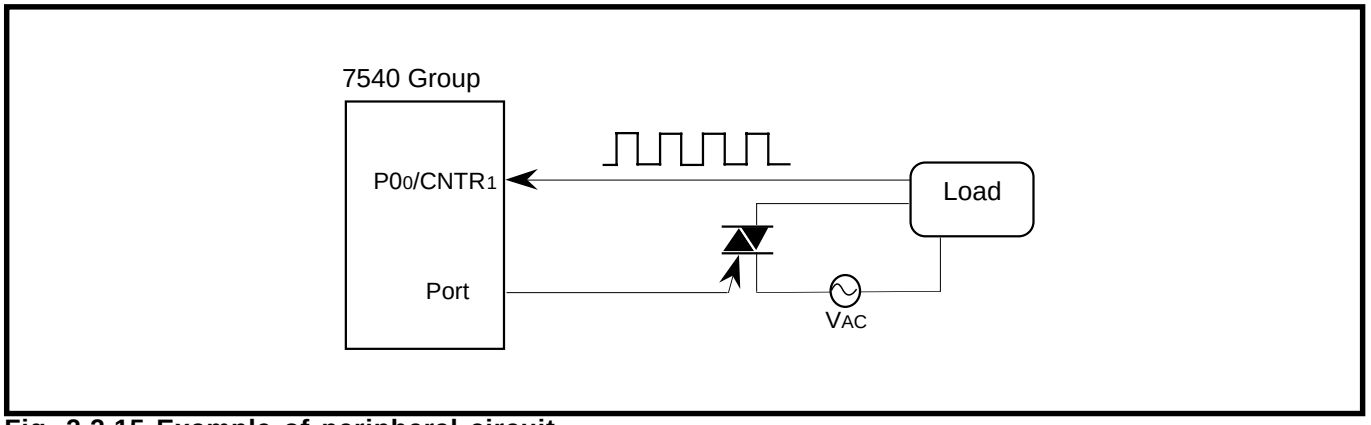
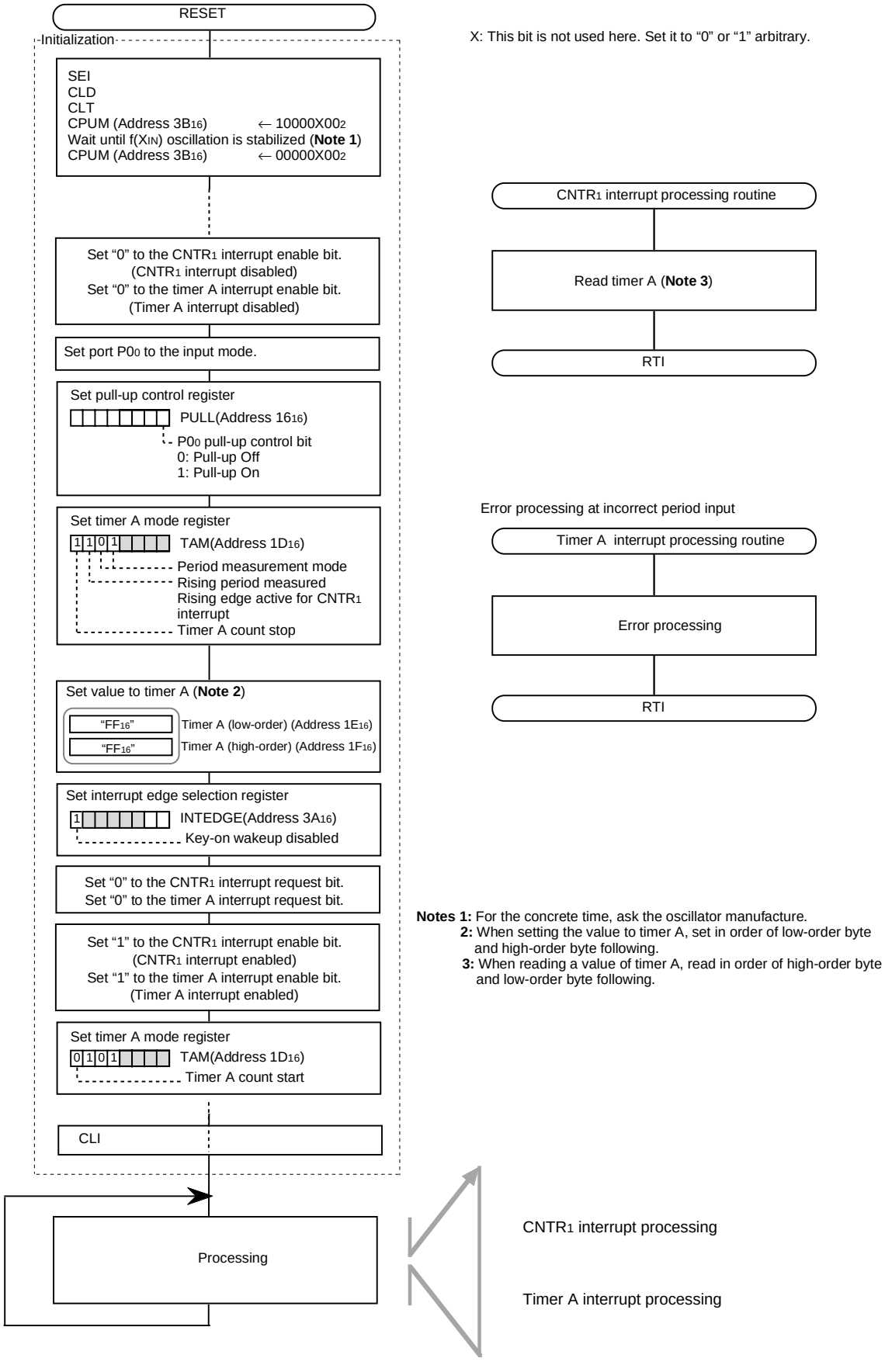


Fig. 2.2.15 Example of peripheral circuit



Notes 1: For the concrete time, ask the oscillator manufacture.
2: When setting the value to timer A, set in order of low-order byte and high-order byte following.
3: When reading a value of timer A, read in order of high-order byte and low-order byte following.

Fig. 2.2.16 Example of control procedure

2.2.5 Event counter mode

(1) Operation description

Timer A counts signals input from the $P0_0/CNTR_1$ pin.

Except for this, the operation in event counter mode is the same as in timer mode.

The active edge of $CNTR_1$ pin input signal can be selected from rising or falling by the $CNTR_1$ active edge switch bit.

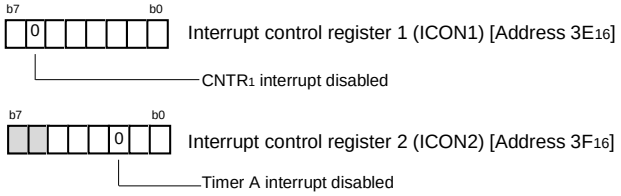
Timer A can stop counting by setting "1" to the timer A count stop bit.

Also, when Timer A underflows, the timer A interrupt request bit is set to "1".

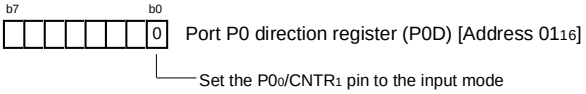
(2) Event counter mode setting method

Figure 2.2.17 and Figure 2.2.18 show the setting method for event counter mode of timer A.

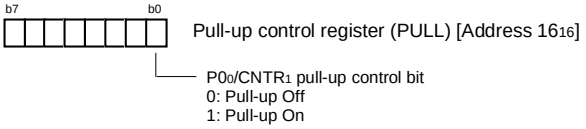
Process 1: Disable timer A interrupt and $CNTR_1$ interrupt.



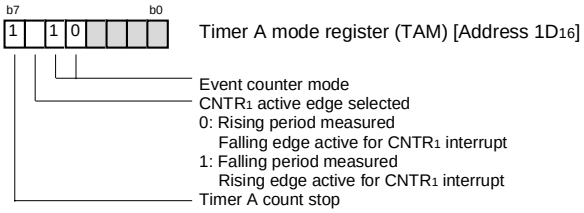
Process 2: Set the $CNTR_1$ pin to the input mode.



Process 3: Set pull-up control register.

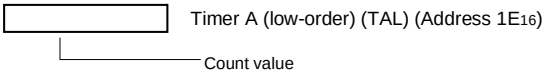


Process 4: Set timer A mode register.

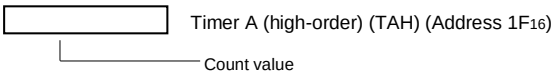


Process 5: Set the count value to timer A (**Note**).

• Set the count value to timer A (low-order)



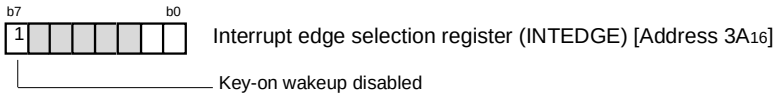
• Set the count value to timer A (high-order)



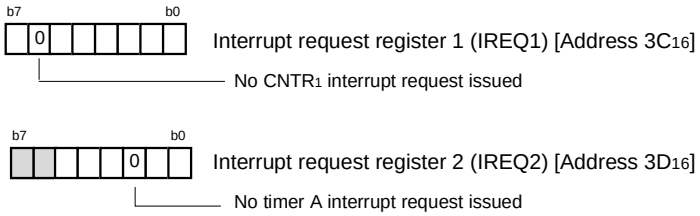
Note: Write both registers in order of timer X (low-order) and the timer X (high-order) following, certainly.

Fig. 2.2.17 Setting method for event counter mode (1)

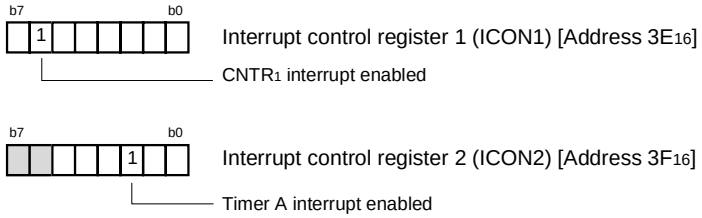
Process 6: In order to use the $CNTR_1$ pin function of the $P0_0$/$CNTR_1$ pin, disable the $P0_0$ key-on wakeup function.



Process 7: In order not to execute the no requested interrupt processing, set "0" (no requested) to the timer A interrupt request bit and $CNTR_1$ interrupt request bit.



Process 8: When the interrupt is used, set "1" (interrupt enabled) to the timer A interrupt enable bit or $CNTR_1$ interrupt enable bit.



Process 9: Start counting of timer A.

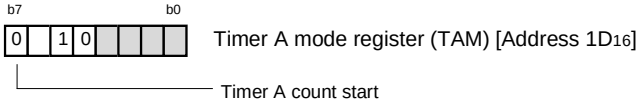


Fig. 2.2.18 Setting method for event counter mode (2)

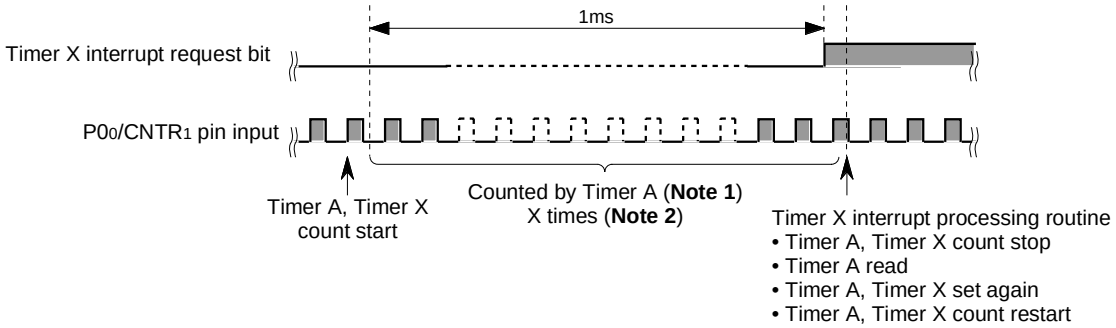
(3) Application example of event counter mode

Outline: The frequency of the pulse which is input to the $P0_0/CNTR_1$ pin ("H" active) is measured by the number of events in a certain period.

Specifications: The count source of timer A is input from the $P0_0/CNTR_1$ pin, and the timer A starts counting the count source. Clock ($f(X_{IN})$ = 8 MHz) is divided by timer X to detect 1 ms. The frequency of the pulse input to the $P0_0/CNTR_1$ pin is calculated by the number of events counted within 1 ms.

Operation clock: $f(X_{IN})$ = 8 MHz, high-speed mode

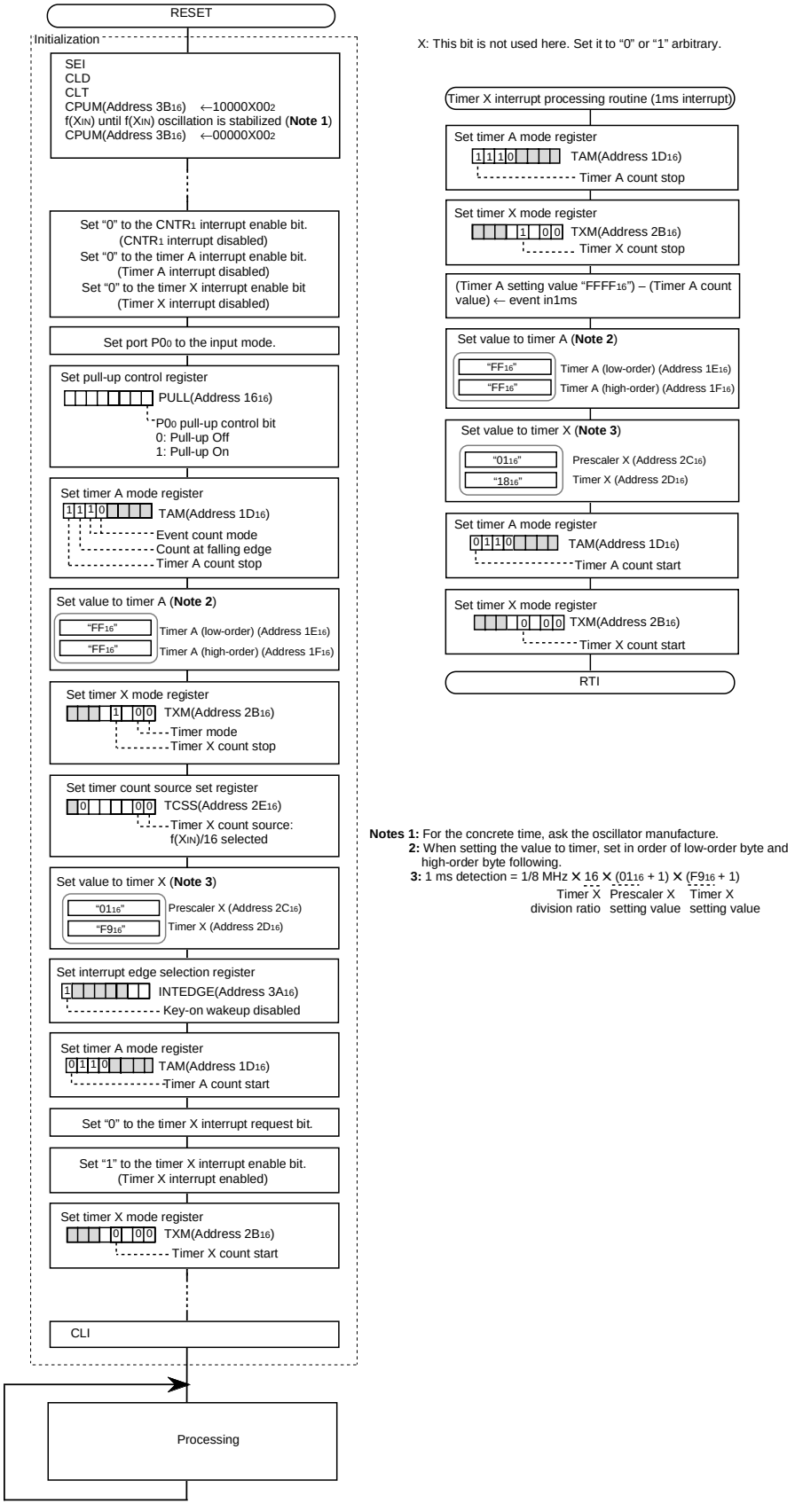
Figure 2.2.19 shows an example of measurement method of frequency, and Figure 2.2.20 shows an example of control procedure.



Notes 1: Counted at falling edge.

2: Frequency of pulse input from $P0_0/CNTR_1$ pin: $\frac{\text{X times}}{\text{1ms}}$ kHz

Fig. 2.2.19 Example of measurement method of frequency



Notes 1: For the concrete time, ask the oscillator manufacture.
2: When setting the value to timer, set in order of low-order byte and high-order byte following.
3: 1 ms detection = 1/8 MHz × 16 × (01_{16} + 1) × ($F9_{16}$ + 1)
Timer X division ratio; Prescaler X setting value; Timer X setting value

Fig. 2.2.20 Example of control procedure

2.2.6 Pulse width HL continuously measurement mode

(1) Operation description

In the pulse width HL continuously measurement mode, the pulse width ("H" and "L" levels) input to the $P0_0/CNTR_1$ pin is measured.

$CNTR_1$ interrupt request is generated at both rising and falling edges of $CNTR_1$ pin input signal. Except for this, the operation in pulse width HL continuously measurement mode is the same as in period measurement mode.

The count value when trigger input from the $CNTR_1$ pin is accepted is retained until Timer A is read once. Timer A can stop counting by setting "1" to the timer A count stop bit.

Also, when Timer A underflows, the timer A interrupt request bit is set to "1".

(2) Pulse width HL continuously measurement mode setting method

Figure 2.2.21 and Figure 2.2.22 show the setting method for pulse width HL continuously measurement mode of timer A.

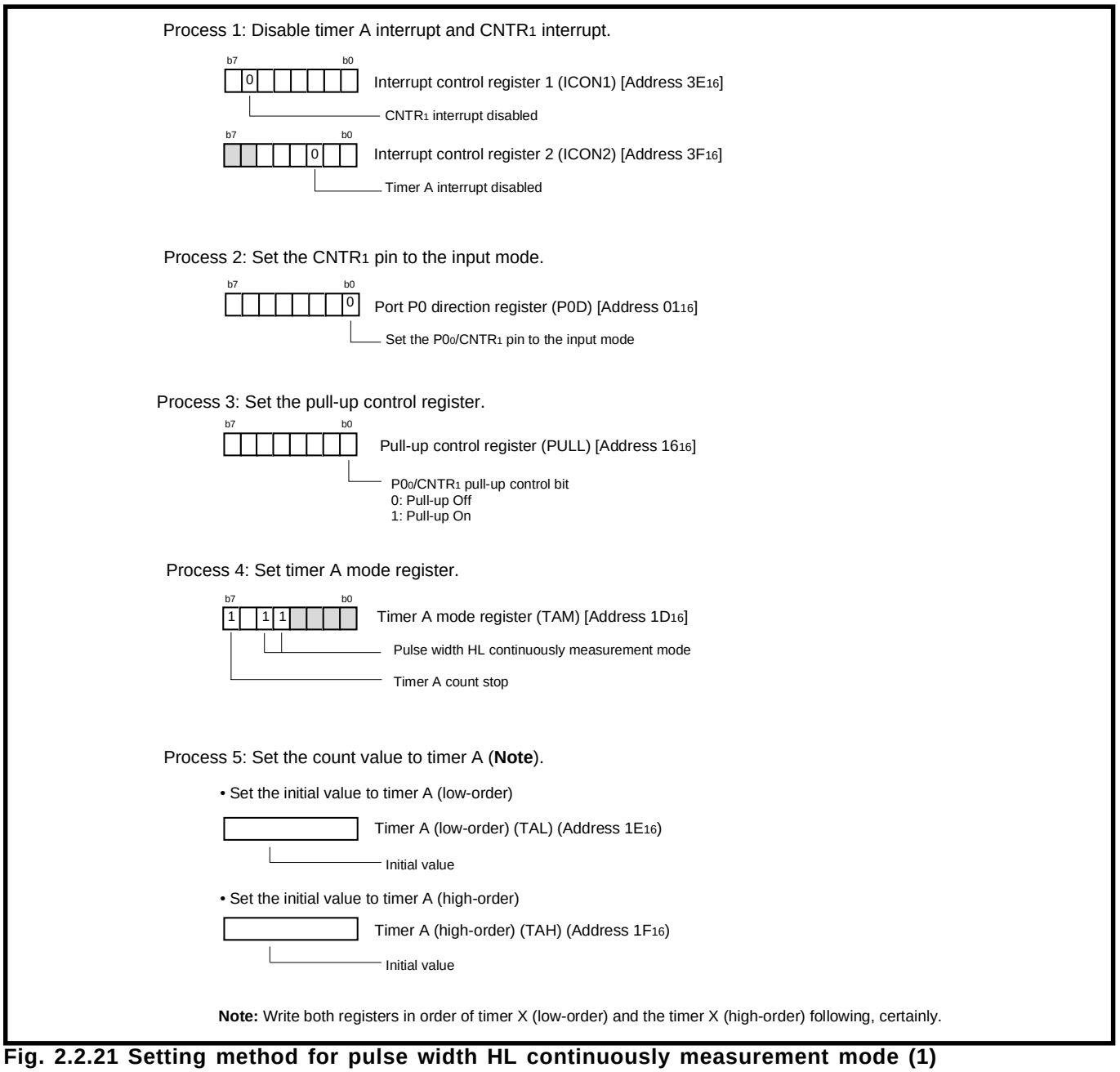
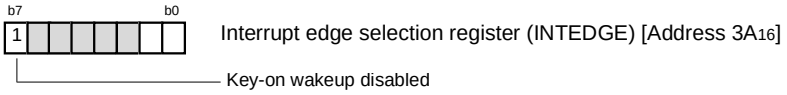
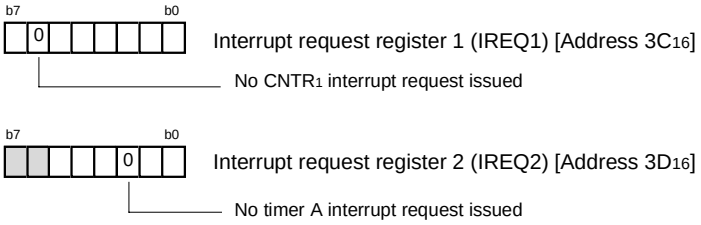


Fig. 2.2.21 Setting method for pulse width HL continuously measurement mode (1)

Process 6: In order to use the $CNTR_1$ pin function of the $P0_0/CNTR_1$ pin, disable the $P0_0$ key-on wakeup function.

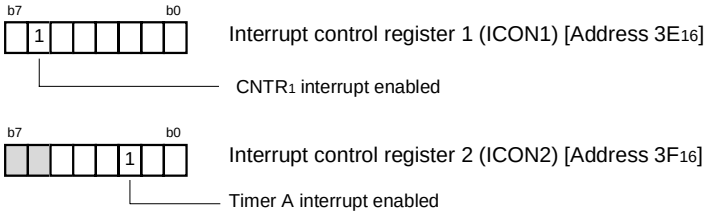


Process 7: In order not to execute the no requested interrupt processing, set "0" (no requested) to the timer A interrupt request bit and $CNTR_1$ interrupt request bit (**Note**).



Note: In the pulse width HL continuously measurement mode, the $CNTR_1$ interrupt request occurs at the rising edge and falling edge of the $P0_0/CNTR_1$ pin regardless of the value of the $P0_0/CNTR_1$ active edge switch bit of the timer A mode register.

Process 8: When the interrupt is used, set "1" (interrupt enabled) to the timer A interrupt enable bit or $CNTR_1$ interrupt enable bit.



Process 9: Start counting of timer A.

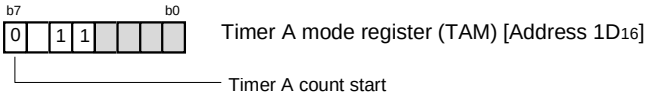


Fig. 2.2.22 Setting method for pulse width HL continuously measurement mode (2)

(3) Application example of pulse width HL continuously measurement mode

Outline: A telephone ringing (calling) pulse* is detected by using the pulse width HL continuously measurement mode.

* Signal which is sent by turning on/off (make/break) the telephone line.
Each country has a different standard. In this case, Japanese domestic standard is adopted as an example.

Specifications: Whether a telephone call exists or not is judged by measuring a pulse width output from the ringing signal detection circuit.
$f(X_{IN})/16$ ($f(X_{IN})$ = 6.4 MHz) is used as the count source, and "H" and "L" pulse width of the ringing pulse are measured by using the pulse width HL continuously measurement mode. When the following conditions are satisfied, it is recognized as a normal value. When the following conditions are not satisfied, it is recognized as an unusual value.
200 ms ≤ "H" pulse width of ringing pulse < 1.2 s
600 ms ≤ "L" pulse width of ringing pulse < 2.2 s
1.0 s ≤ one period ("H" pulse width + "L" pulse width) < 3.0 s
Operation clock: $f(X_{IN})$ = 6.4 MHz, high-speed mode

Figure 2.2.23 shows an example of a peripheral circuit, and Figure 2.2.24 shows an operation timing when a ringing pulse is input. Figures 2.2.25 and 2.2.26 show an example of control procedure.

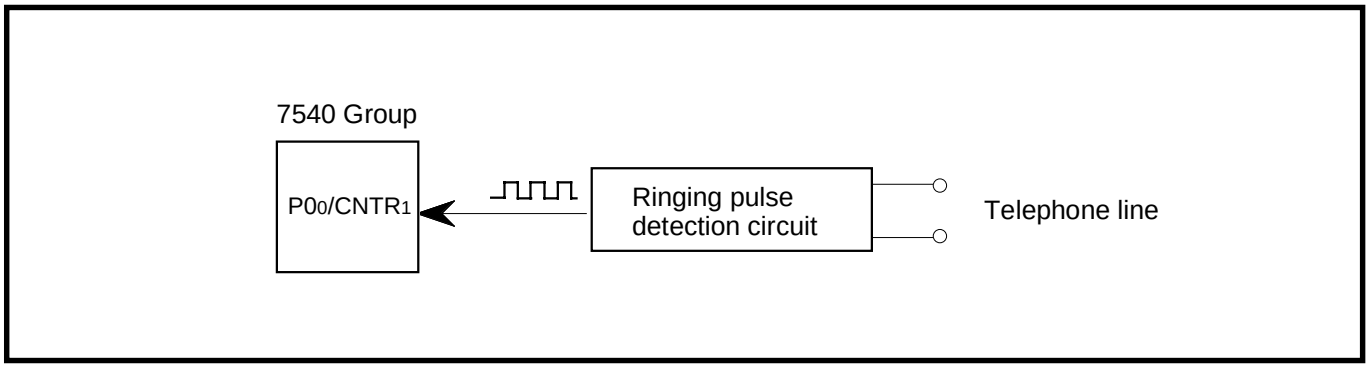


Fig. 2.2.23 Example of peripheral circuit

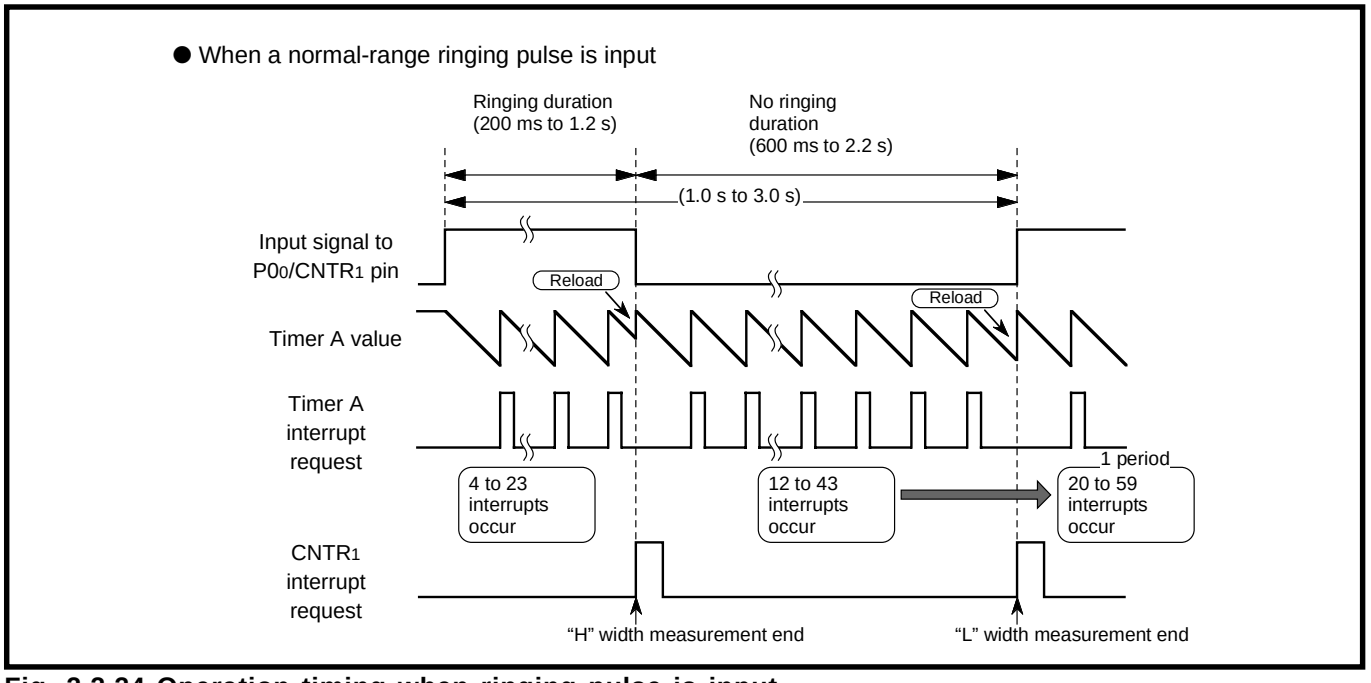


Fig. 2.2.24 Operation timing when ringing pulse is input

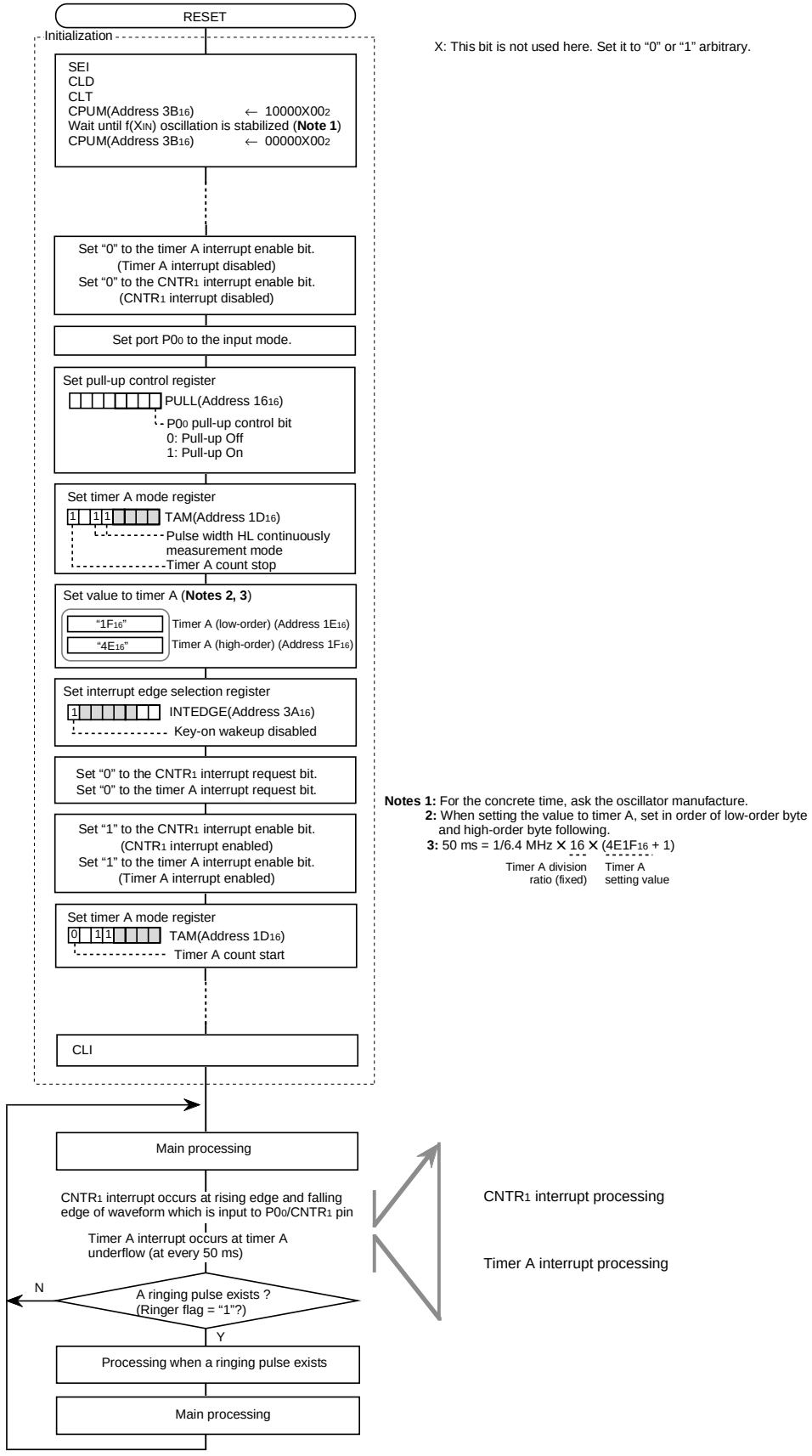


Fig. 2.2.25 Example of control procedure (1)

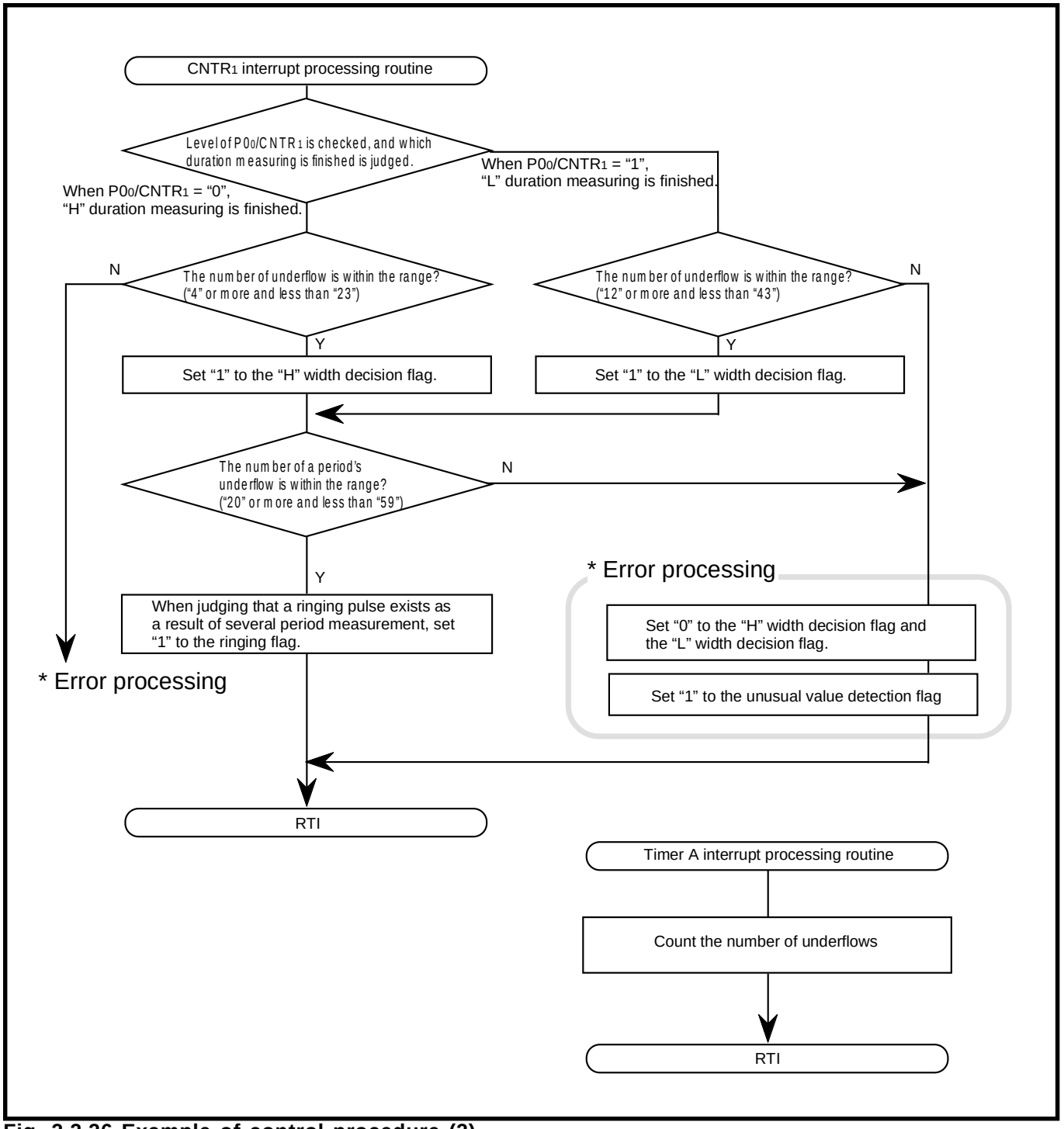


Fig. 2.2.26 Example of control procedure (2)

2.2.7 Notes on timer A

Notes on using timer A are described below.

(1) Common to all modes

① When reading timer A (high-order) (TAH) and timer A (low-order) (TAL), the contents of timer A is read out. Read both registers in order of TAH and TAL following, certainly.
TAH and TAL keep the values until they are read out.
Also, do not write to them during read. In this case, unexpected operation may occur.

② When writing data to TAL and TAH when timer A is operating or stopped, the data are set to timer A and timer A latch simultaneously. Write both registers in order of TAL and TAH following, certainly.
Also, do not read them during write. In this case, unexpected operation may occur.

(2) Period measurement mode, event counter mode, and pulse width HL continuously measurement mode

① In order to use $CNTR_1$ pin, set "0" to bit 0 of the port P0 direction register (input mode).

② In order to use $CNTR_1$ pin, set "1" to bit 7 of the interrupt control register to disable the $P0_0$ key-on wakeup function.

③ $CNTR_1$ interrupt active edge depends on the $CNTR_1$ active edge switch bit. When this bit is "0", the $CNTR_1$ interrupt request bit is set to "1" at the falling edge of the $CNTR_1$ pin input signal. When this bit is "1", the $CNTR_1$ interrupt request bit is set to "1" at the rising edge of the $CNTR_1$ pin input signal.
However, in the pulse width HL continuously measurement mode, $CNTR_1$ interrupt request is generated at both rising and falling edges of $CNTR_1$ pin input signal regardless of the setting of $CNTR_1$ active edge switch bit.

2.3 Timer 1

This paragraph explains the registers setting method and the notes relevant to the timer 1.

2.3.1 Memory map

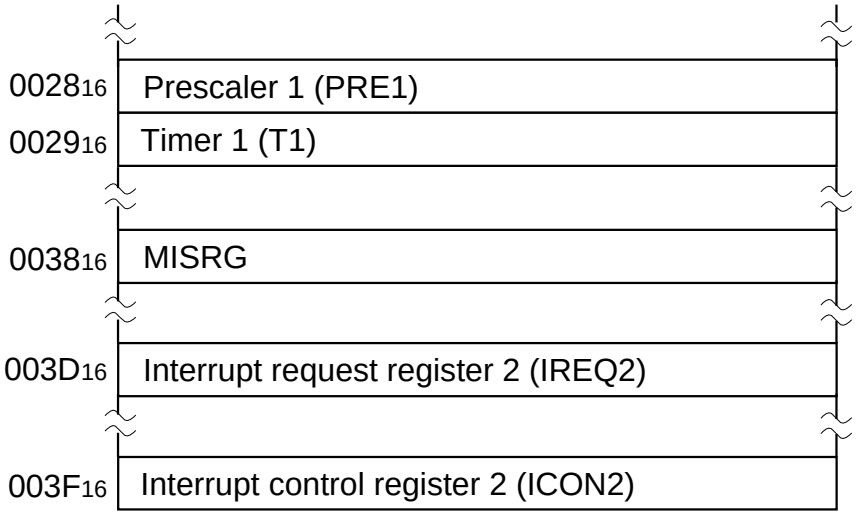


Fig. 2.3.1 Memory map of registers relevant to timer 1

2.3.2 Relevant registers

Prescaler 1

b7	b6	b5	b4	b3	b2	b1	b0

Prescaler 1 (PRE1) [Address : 28 16]

B	Function	At reset	R	W
0	•Set a count value of prescaler 1. •The value set in this register is written to both prescaler 1 and the prescaler 1 latch at the same time. •When this register is read out, the count value of the prescaler 1 latch is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Fig. 2.3.2 Structure of Prescaler 1

Timer 1

b7 b6 b5 b4 b3 b2 b1 b0

Timer 1 (T1) [Address : 29_{16}]

B	Function	At reset	R	W
0	•Set a count value of timer 1. •The value set in this register is written to both timer 1 and timer 1 latch at the same time. •When this register is read out, the timer 1's count value is read out.	1	○	○
1		0	○	○
2		0	○	○
3		0	○	○
4		0	○	○
5		0	○	○
6		0	○	○
7		0	○	○

Fig. 2.3.3 Structure of Timer 1

MISRG

b7 b6 b5 b4 b3 b2 b1 b0

MISRG [Address : 38_{16}]

B	Name	Function	At reset	R	W
0	Oscillation stabilization time set bit after release of the STP instruction	0 : Set "01_{16}" in timer 1, and "FF_{16}" in prescaler 1 automatically 1 : Not set automatically	0	○	○
1	Ceramic or RC oscillation stop detection function active bit	0 : Detection function inactive 1 : Detection function active	0	○	○
2	These are reserved bits. Do not write "1" to these bits.		0	○	×
3			0	○	×
4	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
5			0	○	×
6			0	○	×
7	Oscillation stop detection status bit	0 : Oscillation stop not detected 1 : Oscillation stop detected	(**Note**)	○	×

Note: "0" at normal reset
"1" at reset by detecting the oscillation stop

Fig. 2.3.4 Structure of MISRG

Interrupt request register 2

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt request register 2 (IREQ2) [Address : 3D 16]

B	Name	Function	At reset	R	W
0	Timer Y interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Timer Z interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	Timer A interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	AD converter interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
7			0	○	×

✻: These bits can be cleared to "0" by program, but cannot be set to "1".

Fig. 2.3.5 Structure of Interrupt request register 2

Interrupt control register 2

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 2 (ICON2) [Address : 3F 16]

B	Name	Function	At reset	R	W
0	Timer Y interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer Z interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer A interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	AD conversion interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
7			0	○	×

Fig. 2.3.6 Structure of Interrupt control register 2

2.3.3 Timer 1 operation description

Timer 1 always operates in the timer mode.

Prescaler 1 counts the selected count source. Each time the count clock is input, the contents of Prescaler 1 is decremented by 1.

When the contents of Prescaler 1 reach "00_{16}", an underflow occurs at the next count clock, and the prescaler 1 latch is reloaded into Prescaler 1 and count continues. The division ratio of Prescaler 1 is 1/(n+1) provided that the value of Prescaler 1 is n.

The contents of Timer 1 is decremented by 1 each time the underflow signal of Prescaler 1 is input. When the contents of Timer 1 reach "00_{16}", an underflow occurs at the next count clock, and the timer 1 latch is reloaded into Timer 1 and count continues. The division ratio of Timer 1 is 1/(m+1) provided that the value of Timer 1 is m. Accordingly, the division ratio of Prescaler 1 and Timer 1 is provided as follows that the value of Prescaler 1 is n and the value of Timer 1 is m.

$$\text{Division ratio} = \frac{1}{(n+1) \times (m+1)}$$

Timer 1 cannot stop counting by software.

Also, when timer 1 underflows, the timer 1 interrupt request bit is set to "1".

2.3.4 Notes on timer 1

Note on using timer 1 is described below.

(1) Notes on set of the oscillation stabilizing time

Timer 1 can be used to set the oscillation stabilizing time after release of the **STP** instruction. The oscillation stabilizing time after release of **STP** instruction can be selected from "set automatically"/ "not set automatically" by the oscillation stabilizing time set bit after release of the **STP** instruction of MISRG. When "0" is set to this bit, "01_{16}" is set to timer 1 and "FF_{16}" is set to prescaler 1 automatically. When "1" is set to this bit, nothing is set to timer 1 and prescaler 1. Therefore, set the wait time according to the oscillation stabilizing time of the oscillation. Also, when timer 1 is used, set values again to timer 1 and prescaler 1 after system is returned from the stop mode.

2.4 Timer X

This paragraph explains the registers setting method and the notes relevant to the timer X.

2.4.1 Memory map

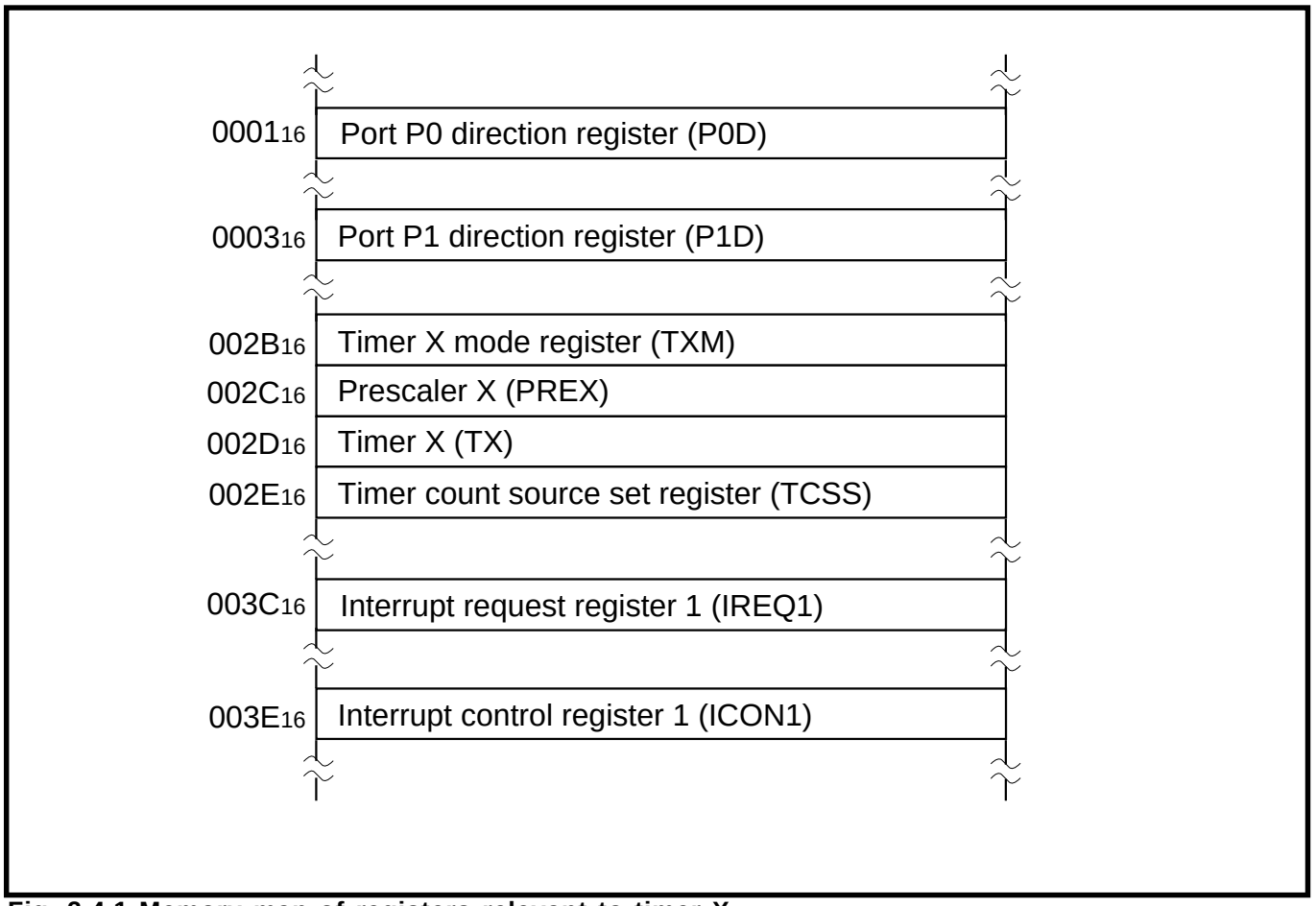


Fig. 2.4.1 Memory map of registers relevant to timer X

2.4.2 Relevant registers

Port P0 direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port P0 direction register (P0D) [Address : 01 16]

B	Name	Function	At reset	R	W
0	Port P0 direction register	0 : Port Pi_0 input mode 1 : Port Pi_0 output mode	0	×	○
1		0 : Port Pi_1 input mode 1 : Port Pi_1 output mode	0	×	○
2		0 : Port Pi_2 input mode 1 : Port Pi_2 output mode	0	×	○
3		0 : Port Pi_3 input mode 1 : Port Pi_3 output mode	0	×	○
4		0 : Port Pi_4 input mode 1 : Port Pi_4 output mode	0	×	○
5		0 : Port Pi_5 input mode 1 : Port Pi_5 output mode	0	×	○
6		0 : Port Pi_6 input mode 1 : Port Pi_6 output mode	0	×	○
7		0 : Port Pi_7 input mode 1 : Port Pi_7 output mode	0	×	○

Fig. 2.4.2 Structure of Port P0 direction register

Port P1 direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port P1 direction register (P1D) [Address : 03 16]

B	Name	Function	At reset	R	W
0	Port P1 direction register	0 : Port $P1_0$ input mode 1 : Port $P1_0$ output mode	0	×	○
1		0 : Port $P1_1$ input mode 1 : Port $P1_1$ output mode	0	×	○
2		0 : Port $P1_2$ input mode 1 : Port $P1_2$ output mode	0	×	○
3		0 : Port $P1_3$ input mode 1 : Port $P1_3$ output mode	0	×	○
4		0 : Port $P1_4$ input mode 1 : Port $P1_4$ output mode	0	×	○
5	Nothing is allocated for these bits. When these bits are read out, the values are undefined.		?	×	×
6			?	×	×
7			?	×	×

Fig. 2.4.3 Structure of Port P1 direction register

Timer X mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer X mode register (TXM) [Address : 2B 16]

B	Name	Function	At reset	R	W
0	Timer X operating mode bits	b1 b0 0 0 : Timer mode 0 1 : Pulse output mode 1 0 : Event counter mode 1 1 : Pulse width measurement mode	0	○	○
1			0	○	○
2	$CNTR_0$ active edge switch bit	The function depends on the operating mode. (Refer to Table 2.4.1)	0	○	○
3	Timer X count stop bit	0 : Count start 1 : Count stop	0	○	○
4	$P0_3$/TX_{OUT} output valid bit	0 : Output invalid (I/O port) 1 : Output valid (Inverted $CNTR_0$ output)	0	○	○
5	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
6			0	○	×
7			0	○	×

Fig. 2.4.4 Structure of Timer X mode register

Table 2.4.1 $CNTR_0$ active edge switch bit function

Timer X operating modes	$CNTR_0$ active edge switch bit (bit 2 of address $2B_{16}$) contents	
Timer mode	"0"	$CNTR_0$ interrupt request occurrence: Falling edge ; No influence to timer count
	"1"	$CNTR_0$ interrupt request occurrence: Rising edge ; No influence to timer count
Pulse output mode	"0"	Pulse output start: Beginning at "H" level $CNTR_0$ interrupt request occurrence: Falling edge
	"1"	Pulse output start: Beginning at "L" level $CNTR_0$ interrupt request occurrence: Rising edge
Event counter mode	"0"	Timer X: Rising edge count $CNTR_0$ interrupt request occurrence: Falling edge
	"1"	Timer X: Falling edge count $CNTR_0$ interrupt request occurrence: Rising edge
Pulse width measurement mode	"0"	Timer X: "H" level width measurement $CNTR_0$ interrupt request occurrence: Falling edge
	"1"	Timer X: "L" level width measurement $CNTR_0$ interrupt request occurrence: Rising edge

Prescaler X

b7	b6	b5	b4	b3	b2	b1	b0

Prescaler X (PREX) [Address : $2C_{16}$]

B	Function	At reset	R	W
0	•Set a count value of prescaler X. •The value set in this register is written to both prescaler X and the prescaler X latch at the same time. •When this register is read out, the count value of the prescaler X is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Fig. 2.4.5 Structure of Prescaler X

Timer X

b7	b6	b5	b4	b3	b2	b1	b0

Timer X (TX) [Address : $2D_{16}$]

B	Function	At reset	R	W
0	•Set a count value of timer X. •The value set in this register is written to both timer X and timer X latch at the same time. •When this register is read out, the timer X's count value is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Fig. 2.4.6 Structure of Timer X

Timer count source set register

b7	b6	b5	b4	b3	b2	b1	b0
	0						

Timer count source set register (TCSS) [Address : 2E 16]

B	Name	Function	At reset	R	W
0	Timer X count source selection bits	b1 b0 0 0 : f(XIN)/16 0 1 : f(XIN)/2 1 0 : f(XIN) (**Note 1**) 1 1 : Not available	0	○	○
1			0	○	○
2	Timer Y count source selection bits	b3 b2 0 0 : f(XIN)/16 0 1 : f(XIN)/2 1 0 : Ring oscillator output (**Note 2**) 1 1 : Not available	0	○	○
3			0	○	○
4	Timer Z count source selection bits	b5 b4 0 0 : f(XIN)/16 0 1 : f(XIN)/2 1 0 : Timer Y underflow 1 1 : Not available	0	○	○
5			0	○	○
6	Fix this bit to “0”.		0	○	○
7	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are “0”.		0	○	×

Notes 1: f(XIN) can be used as timer X count source only when using a ceramic oscillator or ring oscillator.
Do not use it at RC oscillation.
2: System operates using a ring oscillator as a count source by setting the ring oscillator to oscillation enabled by bit 3 of CPUM.

Fig. 2.4.7 Structure of Timer count source set register

Interrupt request register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt request register 1 (IREQ1) [Address : 3C $_{16}$]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Serial I/O1 transmit interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	INT_0 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	INT_1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	Key-on wake up interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	$CNTR_0$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	$CNTR_1$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻

✻: These bits can be cleared to "0" by program, but cannot be set to "1".

Fig. 2.4.8 Structure of Interrupt request register 1

Interrupt control register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 1 (ICON1) [Address : 3E $_{16}$]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Serial I/O1 transmit interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	INT_0 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	INT_1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Key-on wake up interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	$CNTR_0$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	$CNTR_1$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Fig. 2.4.9 Structure of Interrupt control register 1

2.4 Timer X

2.4.3 Timer mode

(1) Operation description

Prescaler X counts the selected count source by the timer X count source selection bits. Each time the count clock is input, the contents of Prescaler X is decremented by 1. When the contents of Prescaler X reach "00_{16}", an underflow occurs at the next count clock, and the prescaler X latch is reloaded into Prescaler X and count continues. The division ratio of Prescaler X is 1/(n+1) provided that the value of Prescaler X is n.

The contents of Timer X is decremented by 1 each time the underflow signal of Prescaler X is input. When the contents of Timer X reach "00_{16}", an underflow occurs at the next count clock, and the timer X latch is reloaded into Timer X and count continues. The division ratio of Timer X is 1/(m+1) provided that the value of Timer X is m. Accordingly, the division ratio of Prescaler X and Timer X is provided as follows that the value of Prescaler X is n and the value of Timer X is m.

$$\text{Division ratio} = \frac{1}{(n+1) \times (m+1)}$$

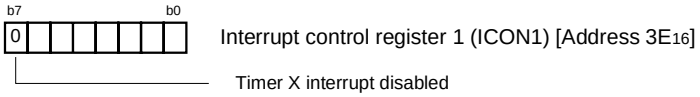
Timer X can stop counting by setting "1" to the timer X count stop bit.

Also, when Timer X underflows, the timer X interrupt request bit is set to "1".

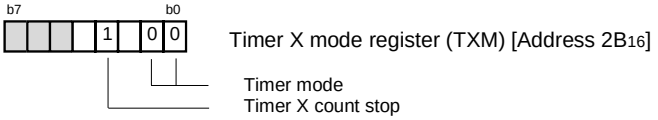
(2) Timer mode setting method

Figure 2.4.10 shows the setting method for timer mode of timer X.

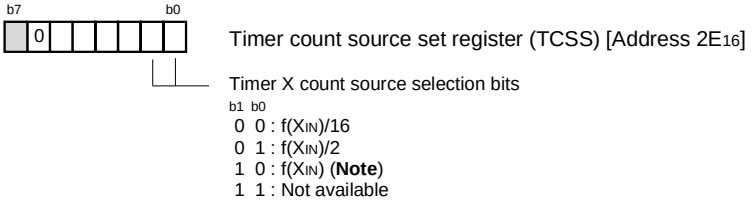
Process 1: Disable timer X interrupt.



Process 2: Set timer X mode register.



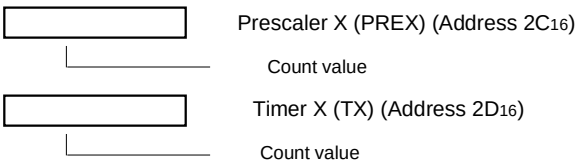
Process 3: Set timer count source set register.



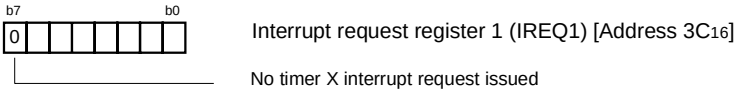
Note: f(XIN) can be used only when a ceramic resonator or a ring oscillator is used. Do not use f(XIN) at RC oscillation.

Process 4: Set the count value to timer X.

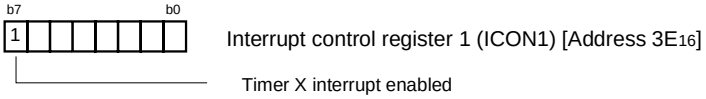
• Set the count value to prescaler X and timer X



Process 5: In order not to execute the no requested interrupt processing, set “0” (no requested) to the timer X interrupt request bit.



Process 6: When timer X interrupt is used, set “1” (interrupt enabled) to the timer X interrupt enable bit.



Process 7: Start counting of timer X.

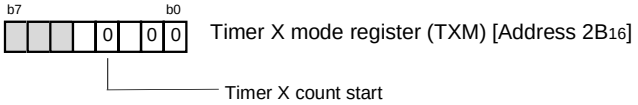


Fig. 2.4.10 Setting method for timer mode

(3) Application example of timer mode

Outline: The input clock is divided by the timer so that the clock is counted up every 250 ms intervals.

Specifications: •The f(X_{IN}) = 4.19 MHz (2^{22} Hz) is divided by timer X.
•The clock is counted up in the timer X interrupt processing routine (timer X interrupt occurs every 250 ms).
• Operation clock: f(X_{IN}) = 4.19 MHz, high-speed mode

Figure 2.4.11 shows the connection of timer and setting of division ratio and Figure 2.4.12 shows an example of control procedure.

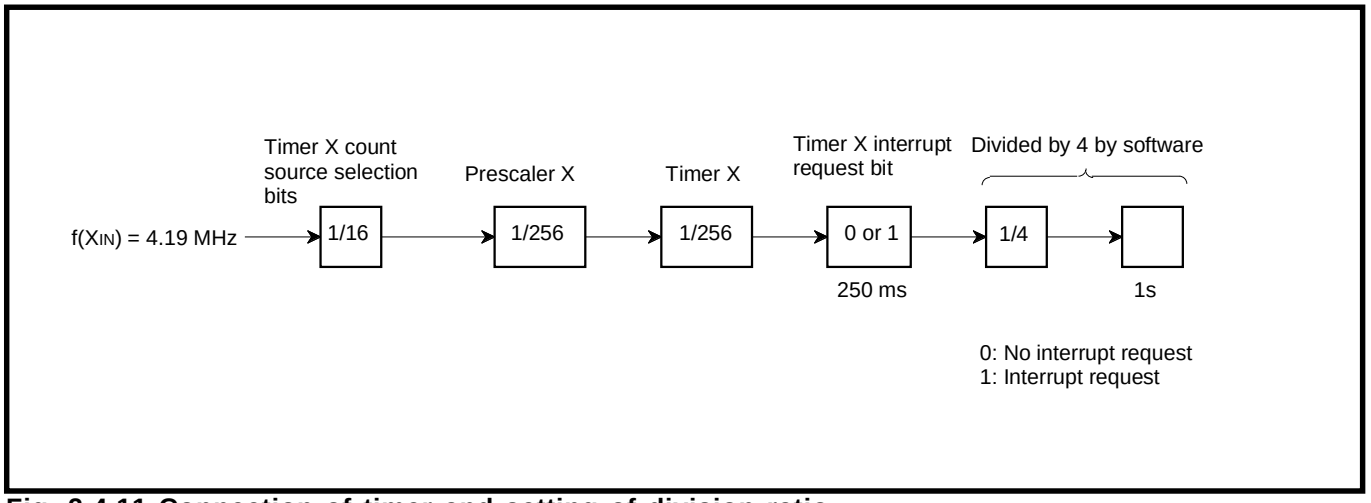


Fig. 2.4.11 Connection of timer and setting of division ratio

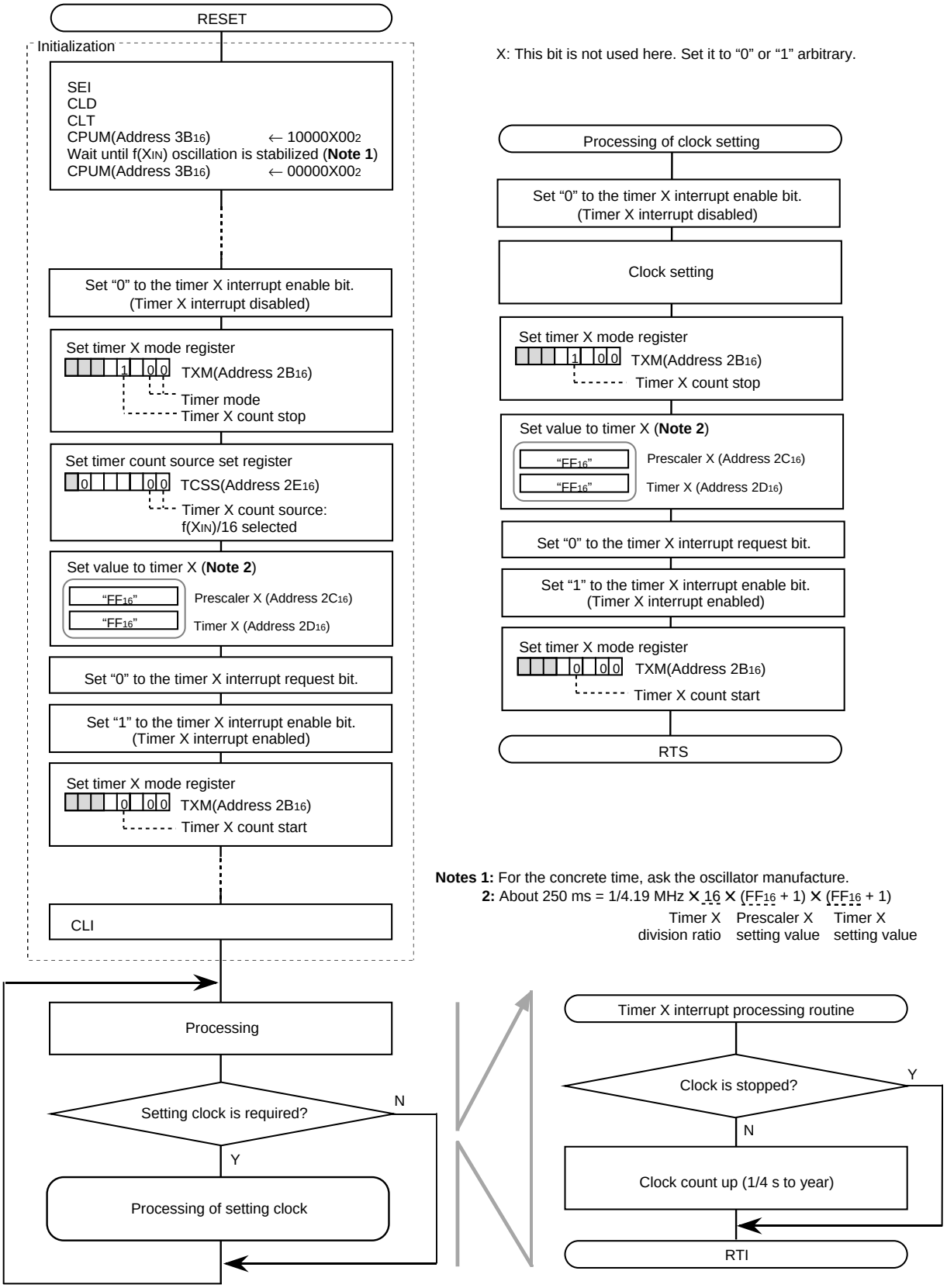


Fig. 2.4.12 Example of control procedure

2.4.4 Pulse output mode

(1) Operation description

In the pulse output mode, the waveform whose polarity is inverted each time timer X underflows is output from the $P1_4/CNTR_0$ pin.

The output level of $CNTR_0$ pin can be selected by the $CNTR_0$ active edge switch bit. When the $CNTR_0$ active edge switch bit is "0", the output of $CNTR_0$ pin is started at "H" level. When this bit is "1", the output is started at "L" level.

Also, the inverted waveform of pulse output from $CNTR_0$ pin can be output from TX_{OUT} pin by setting "1" to the $P0_3/TX_{OUT}$ output valid bit.

When using a timer in this mode, set the port $P1_4$ and $P0_3$ direction registers to output mode.

Timer X can stop counting by setting "1" to the timer X count stop bit.

Also, when Timer X underflows, the timer X interrupt request bit is set to "1".

(2) Pulse output mode setting method

Figure 2.4.13 and Figure 2.4.14 show the setting method for pulse output mode of timer X.

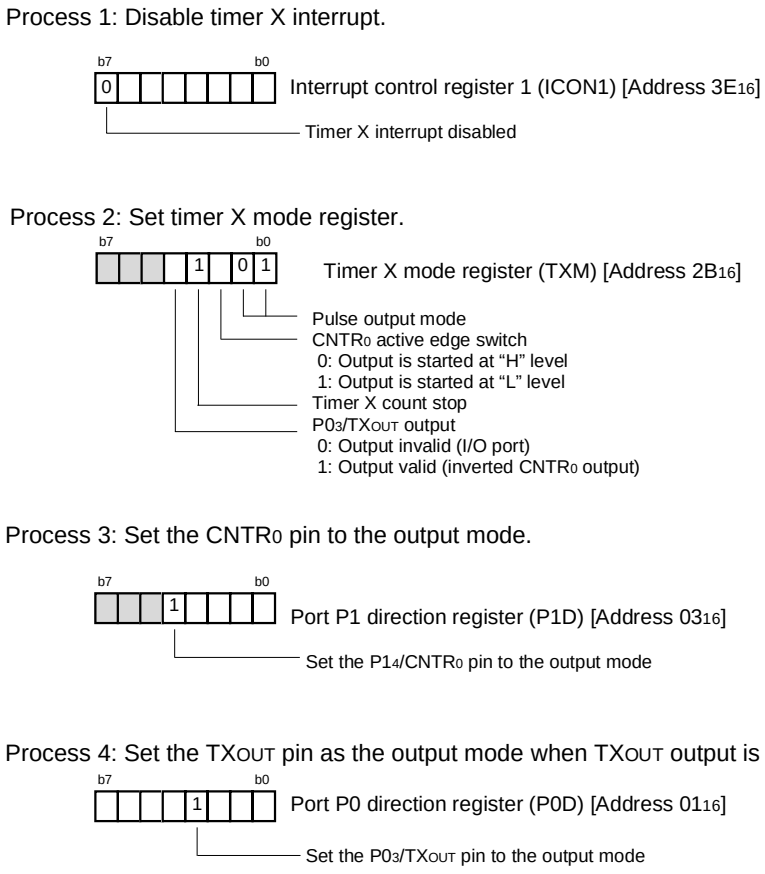
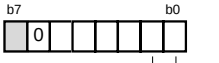


Fig. 2.4.13 Setting method for pulse output mode (1)

Process 5: Set timer count source set register.



Timer count source set register (TCSS) [Address $2E_{16}$]

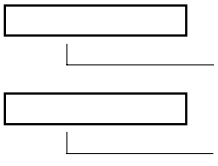
Timer X count source selection bits

b1 b0
0 0 : f(X_{IN})/16
0 1 : f(X_{IN})/2
1 0 : f(X_{IN}) (**Note)**
1 1 : Not available

Note: f(X_{IN}) can be used only when a ceramic resonator or a ring oscillator is used. Do not use f(X_{IN}) at RC oscillation.

Process 6: Set the count value to timer X.

• Set the count value to prescaler X and timer X



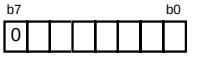
Prescaler X (PREX) (Address $2C_{16}$)

Count value

Timer X (TX) (Address $2D_{16}$)

Count value

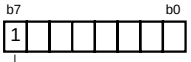
Process 7: In order not to execute the no requested interrupt processing, set “0” (no requested) to the timer X interrupt request bit.



Interrupt request register 1 (IREQ1) [Address $3C_{16}$]

No timer X interrupt request issued

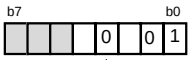
Process 8: When the interrupt is used, set “1” (interrupt enabled) to the timer X interrupt enable bit.



Interrupt control register 1 (ICON1) [Address $3E_{16}$]

Timer X interrupt enabled

Process 9: Start counting of timer X.



Timer X mode register (TXM) [Address $2B_{16}$]

Timer X count start

Fig. 2.4.14 Setting method for pulse output mode (2)

(3) Application example of pulse output mode

Outline: The pulse output mode of timer X is used for a piezoelectric buzzer output.

Specifications: The rectangular waveform which is clock f(X_{IN}) = 4 MHz divided up to 4 kHZ is output from the $P1_4$/$CNTR_0$ pin.

The level of the $P1_4$/$CNTR_0$ pin is fixed to "H" while a piezoelectric buzzer output is stopped.

Operation clock: f(X_{IN}) = 4 MHz, double-speed mode

Figure 2.4.15 shows an example of a peripheral circuit, Figure 2.4.16 shows the connection of timer and setting of the division ratio, and Figure 2.4.17 shows an example of control procedure.

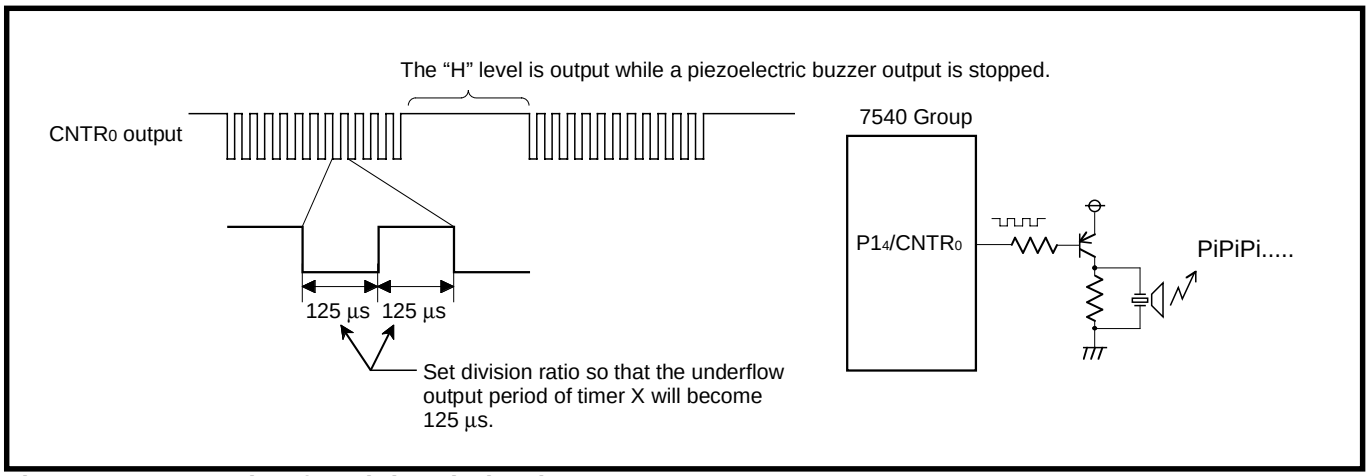


Fig. 2.4.15 Example of peripheral circuit

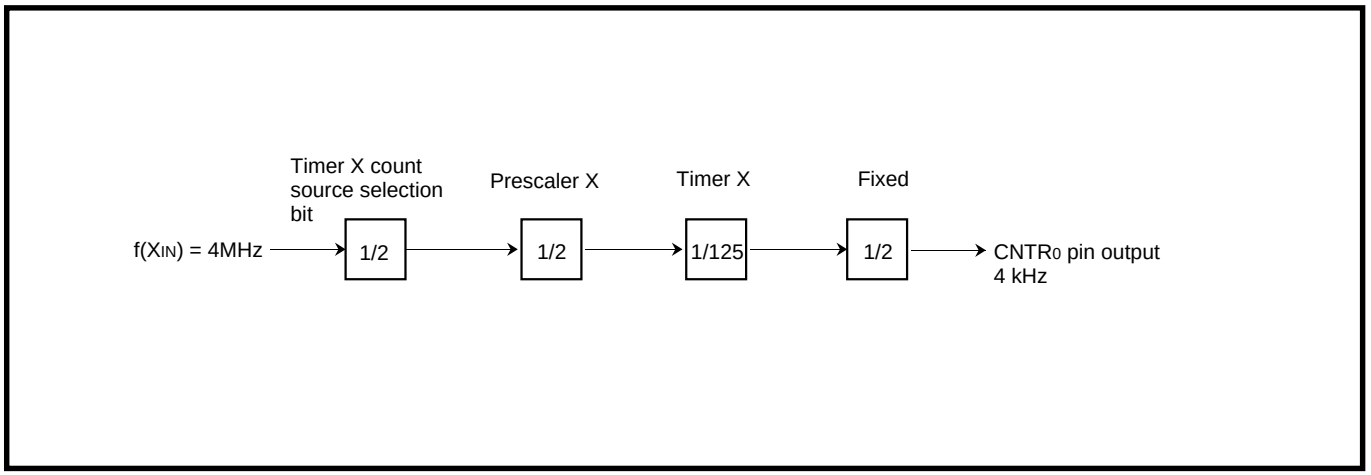
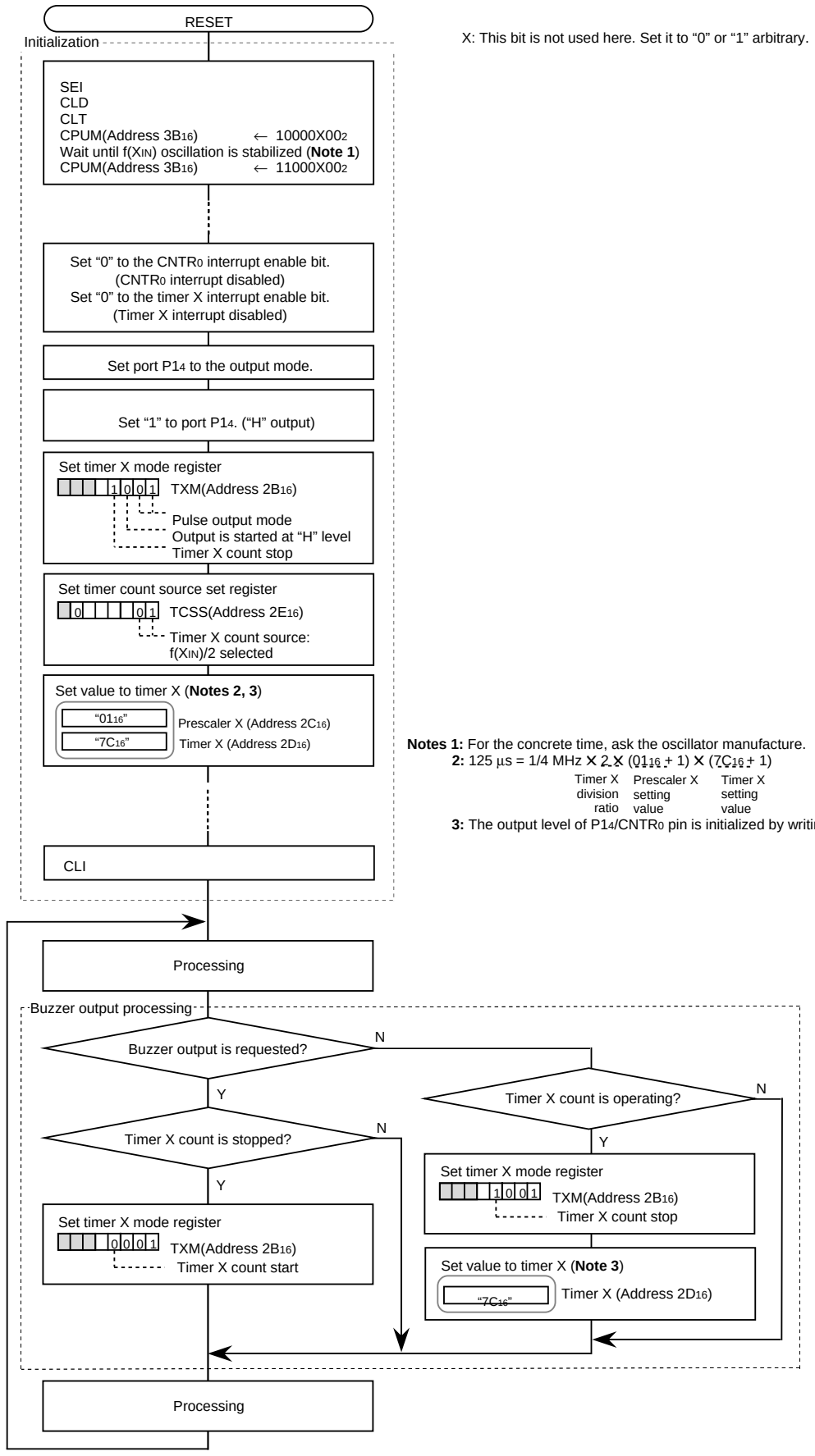


Fig. 2.4.16 Connection of timer and setting of division ratio



X: This bit is not used here. Set it to "0" or "1" arbitrary.

Notes 1: For the concrete time, ask the oscillator manufacture.

2: 125 μs = 1/4 MHz × 2 × (01_{16} + 1) × ($7C_{16}$ + 1)

Timer X division ratio | Prescaler X setting value | Timer X setting value

3: The output level of P14/CNTR0 pin is initialized by writing to timer X.

Fig. 2.4.17 Example of control procedure

2.4.5 Event counter mode

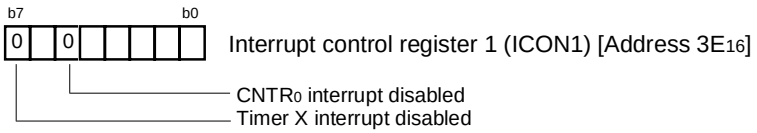
(1) Operation description

The timer X counts signals input from the $P1_4/CNTR_0$ pin.
Except for this, the operation in event counter mode is the same as in timer mode.
The active edge of $CNTR_0$ pin input signal can be selected from rising or falling by the $CNTR_0$ active edge switch bit.
Timer X can stop counting by setting "1" to the timer X count stop bit.
Also, when Timer X underflows, the timer X interrupt request bit is set to "1".

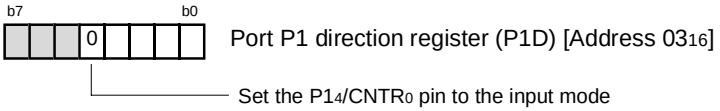
(2) Event counter mode setting method

Figure 2.4.18 and Figure 2.4.19 show the setting method for event counter mode of timer X.

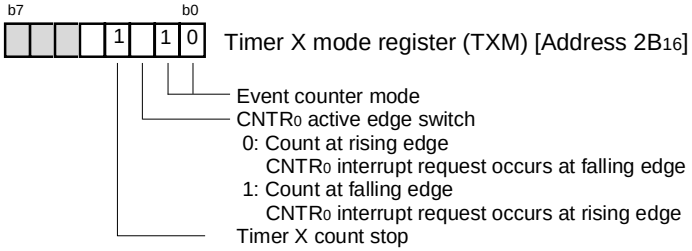
Process 1: Disable timer X interrupt and $CNTR_0$ interrupt.



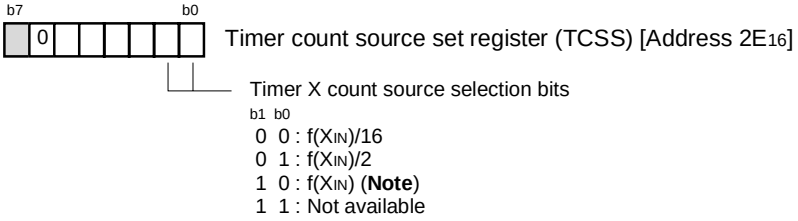
Process 2: Set the $CNTR_0$ pin to the input mode.



Process 3: Set timer X mode register.



Process 4: Set timer count source set register.

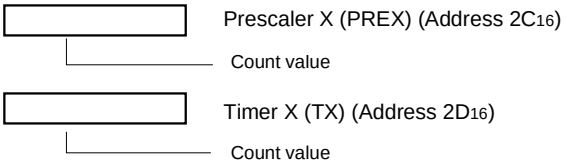


Note: $f(X_{IN})$ can be used only when a ceramic oscillator or a ring oscillator is used.
Do not use $f(X_{IN})$ at RC oscillation.

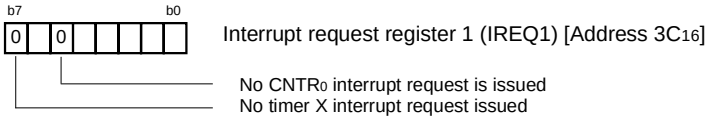
Fig. 2.4.18 Setting method for event counter mode (1)

Process 5: Set the count value to timer X.

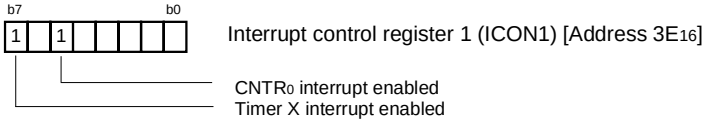
• Set the count value to prescaler X and timer X



Process 6: In order not to execute the no requested interrupt processing, set “0” (no requested) to the timer X interrupt request bit and CNTR1 interrupt request bit.



Process 7: When the interrupt is used, set “1” (interrupt enabled) to the timer X interrupt enable bit.



Process 8: Start counting of timer X.

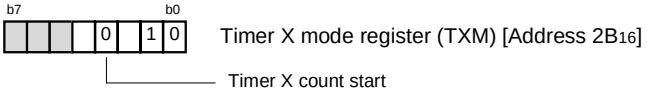


Fig. 2.4.19 Setting method for event counter mode (2)

(3) Application example of event counter mode

Outline: Pulses generated corresponding to the water flow rate are counted for a fixed period (100 ms), and the water flow rate during this period is calculated.

Specifications: Pulses generated corresponding to the water flow rate are input to the $P1_4/CNTR_0$ pin and counted using timer X.

The contents of timer X are read in the timer Y interrupt processing routine generated after 100 ms from the start of counting pulses, and the water flow rate during 100 ms is calculated.

Operation clock: $f(X_{IN})$ = 8 MHz, high-speed mode

Figure 2.4.20 shows an example of peripheral circuit, Figure 2.4.21 shows the method of measuring water flow rate, and Figure 2.4.21 shows an example of control procedure.

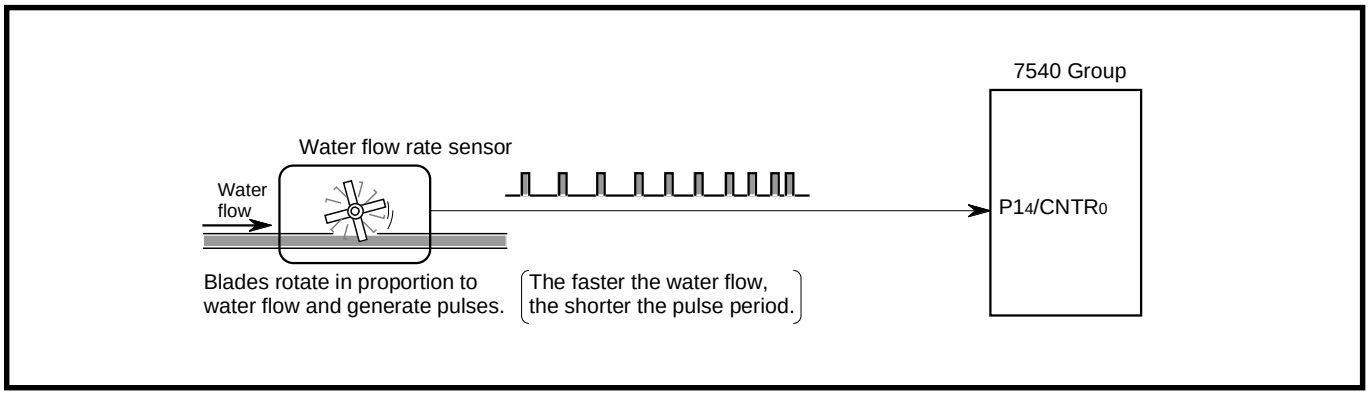


Fig. 2.4.20 Example of peripheral circuit

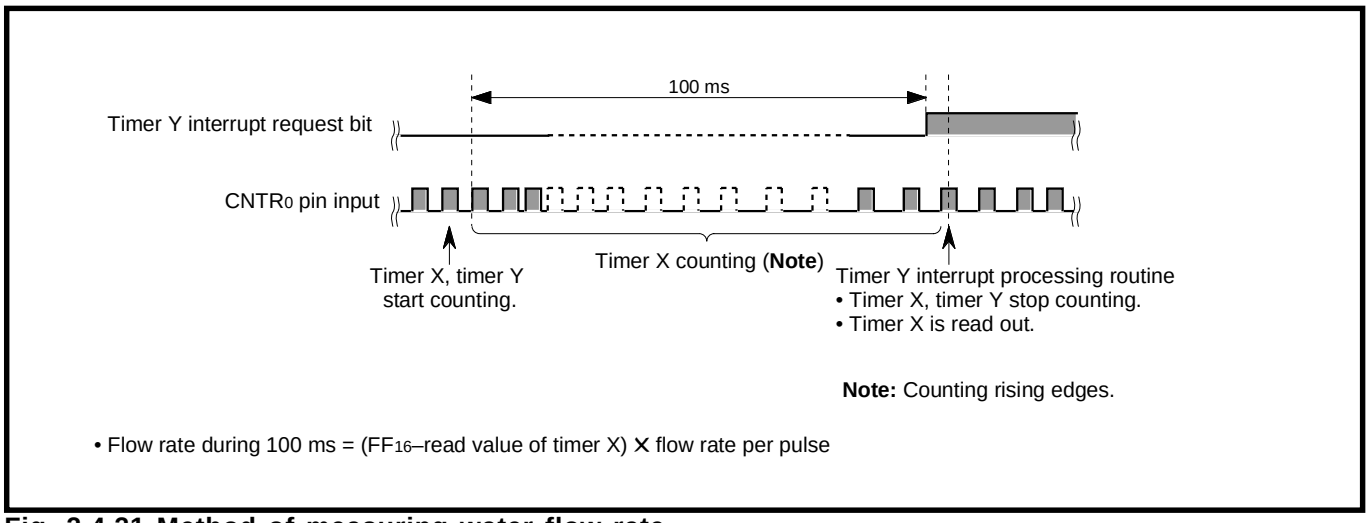


Fig. 2.4.21 Method of measuring water flow rate

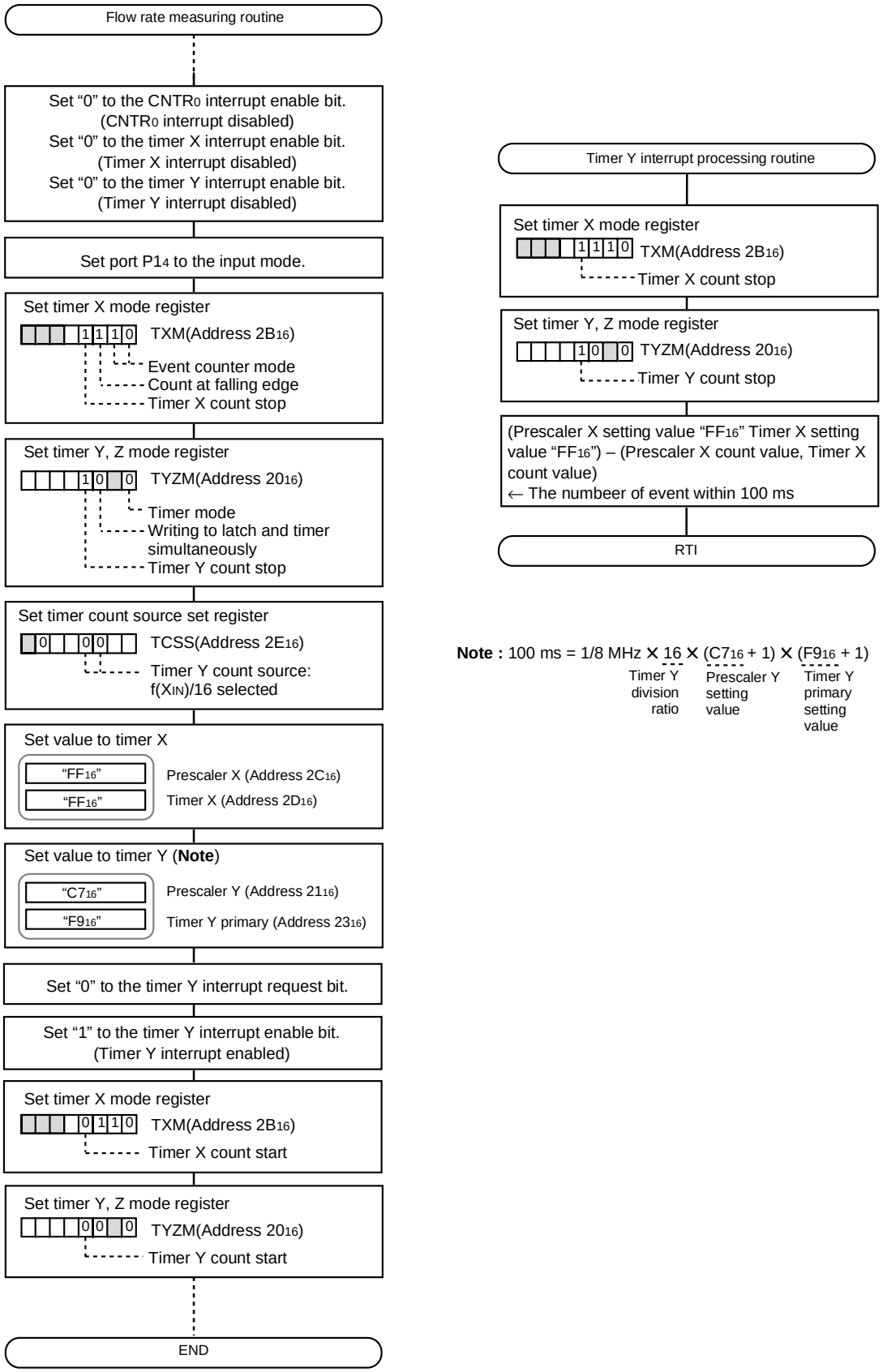


Fig. 2.4.22 Example of control procedure

2.4.6 Pulse width measurement mode

(1) Operation description

In the pulse width measurement mode, the pulse width of the signal input to $P1_4/CNTR_0$ pin is measured.

The operation of Timer X can be controlled by the level of the signal input from the $CNTR_0$ pin.

When the $CNTR_0$ active edge switch bit is "0", the signal selected by the timer X count source selection bit is counted while the input signal level of $CNTR_0$ pin is "H". The count is stopped while the pin is "L". Also,

when the $CNTR_0$ active edge switch bit is "1", the signal selected by the timer X count source selection bit is counted while the input signal level of $CNTR_0$ pin is "L". The count is stopped while the pin is "H".

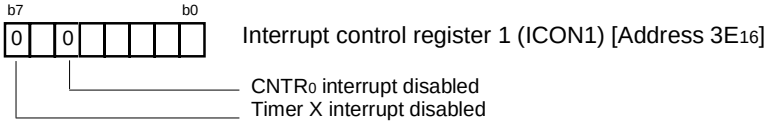
Timer X can stop counting by setting "1" to the timer X count stop bit.

Also, when Timer X underflows, the timer X interrupt request bit is set to "1".

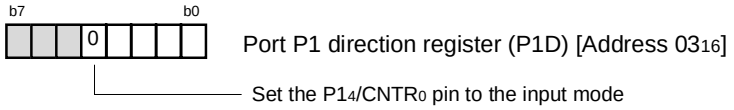
(2) Pulse width HL continuously measurement mode setting method

Figure 2.4.23 and Figure 2.4.24 show the setting method for pulse width measurement mode of timer X.

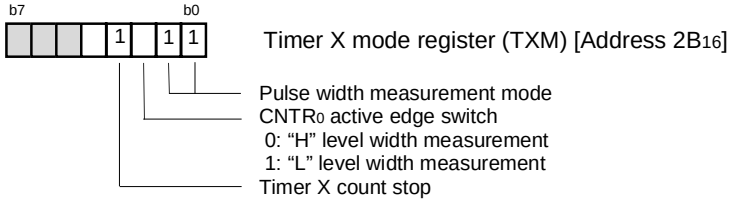
Process 1: Disable timer X interrupt and $CNTR_0$ interrupt.



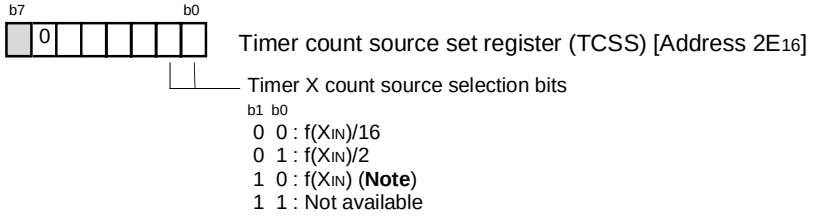
Process 2: Set the $CNTR_0$ pin to the input mode.



Process 3: Set timer X mode register.



Process 4: Set timer X count source.

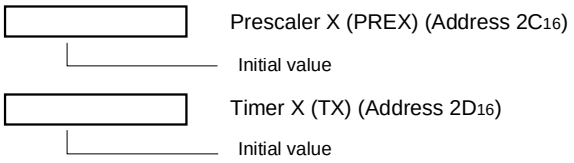


Note: $f(X_{IN})$ can be used only when a ceramic resonator or a ring oscillator is used. Do not use $f(X_{IN})$ at RC oscillation.

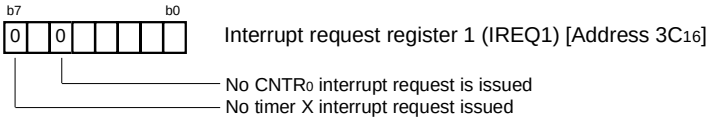
Fig. 2.4.23 Setting method for pulse width measurement mode (1)

Process 5: Set the count value to timer X.

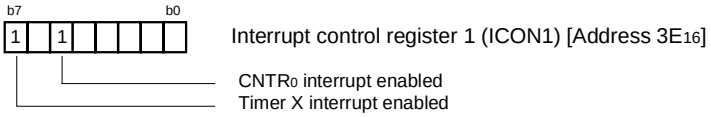
• Set the initial value to prescaler X and timer X



Process 6: In order not to execute the no requested interrupt processing, set “0” (no requested) to the timer X interrupt request bit and CNTR0 interrupt request bit.



Process 7: When the interrupt is used, set “1” (interrupt enabled) to the timer X interrupt enable bit.



Process 8: Start counting of timer X.

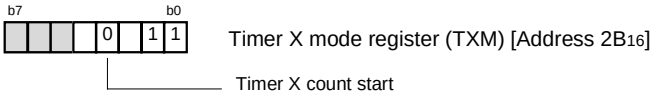


Fig. 2.4.24 Setting method for pulse width measurement mode (2)

(3) Application example of pulse width measurement mode

Outline: "H" level width of pulse input to $P1_4/CNTR_0$ pin is counted.

Specifications: The "H" level width of a FG pulse input to the $P1_4/CNTR_0$ pin is counted. An underflow is detected by the timer X interrupt. The completion of "H" level of input pulse is detected by the $CNTR_0$ interrupt.
Operation clock: $f(X_{IN})$ = 4.19 MHz, high-speed mode

Example: When $f(X_{IN})$ = 4.19 MHz, the count source becomes 3.8 μs divided by 16. Measurement can be made up to 250 ms in the range of "$FFFF_{16}$" to "0000_{16}".

Figure 2.4.25 shows a connection of the timer and setting of the division ratio. Figure 2.4.26 shows an example of control procedure.

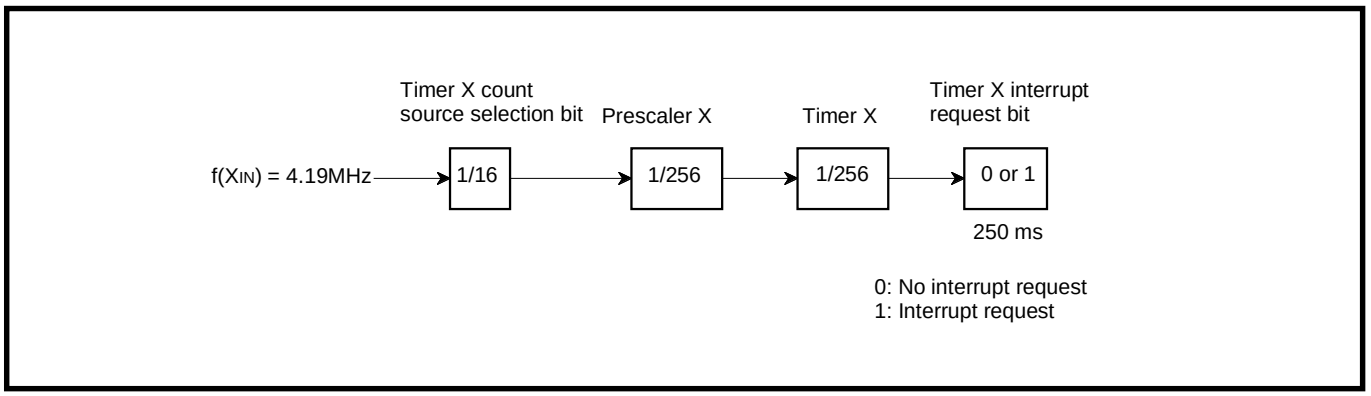
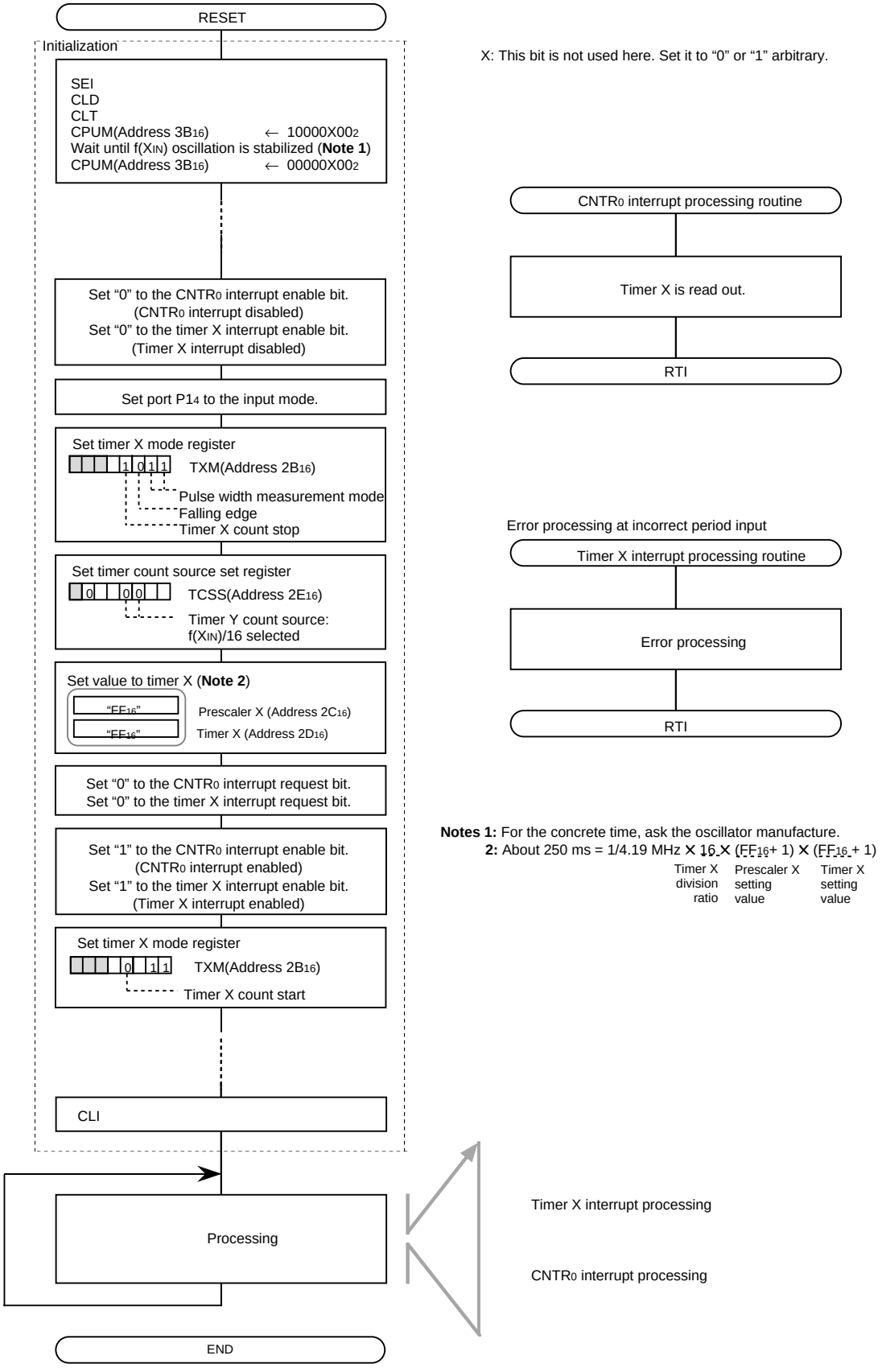


Fig. 2.4.25 Connection of timer and setting of division ratio



Notes 1: For the concrete time, ask the oscillator manufacture.

2: About 250 ms = 1/4.19 MHz × 16 × (FF_{16} + 1) × (FF_{16} + 1)

Timer X division ratio | Prescaler X setting value | Timer X setting value

Fig. 2.4.26 Example of control procedure

2.4.7 Notes on timer X

Notes on using each mode of timer X are described below.

(1) Count source

① f(X_{IN}) can be used only when a ceramic oscillator or a ring oscillator is used. Do not use f(X_{IN}) at RC oscillation.

(2) Pulse output mode

① In order to use $CNTR_0$ pin, set "1" to bit 4 of the port P1 direction register (output mode).

② In order to use TX_{OUT} pin, set "1" to bit 3 of the port P0 direction register (output mode).

③ $CNTR_0$ interrupt active edge depends on the $CNTR_0$ active edge switch bit. When this bit is "0", the $CNTR_0$ interrupt request bit is set to "1" at the falling edge of $CNTR_0$ pin input signal. When this bit is "1", the $CNTR_0$ interrupt request bit is set to "1" at the rising edge of $CNTR_0$ pin input signal.

(3) Pulse width measurement mode

① In order to use $CNTR_0$ pin, set "1" to bit 4 of the port P1 direction register (output mode).

② $CNTR_0$ interrupt active edge depends on the $CNTR_0$ active edge switch bit. When this bit is "0", the $CNTR_0$ interrupt request bit is set to "1" at the falling edge of $CNTR_0$ pin input signal. When this bit is "1", the $CNTR_0$ interrupt request bit is set to "1" at the rising edge of $CNTR_0$ pin input signal.

2.5 Timer Y and timer Z

This paragraph explains the registers setting method and the notes relevant to the timer Y and timer Z.

2.5.1 Memory map

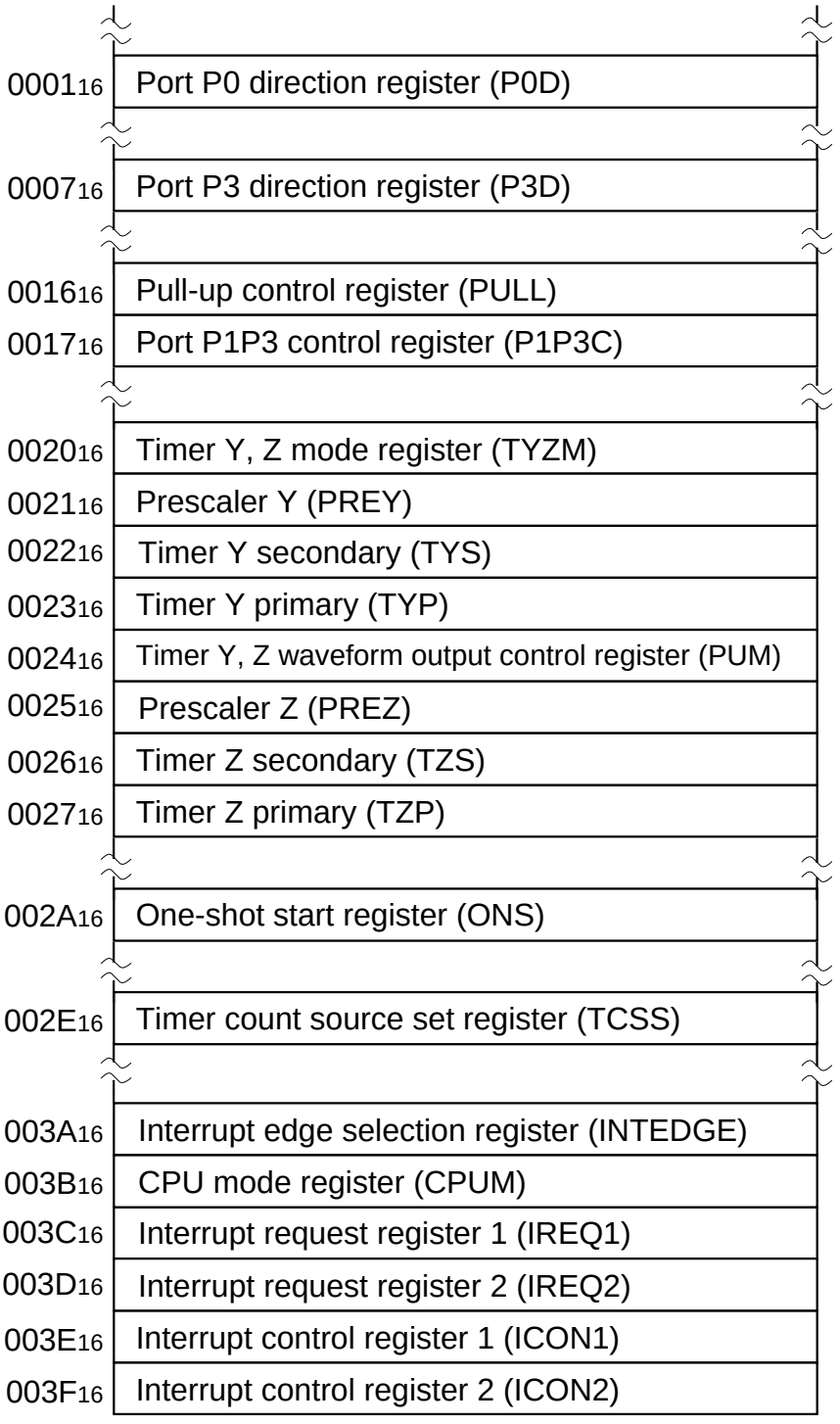


Fig. 2.5.1 Memory map of registers relevant to timer Y and timer Z

2.5.2 Relevant registers

Port P0 direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port P0 direction register (P0D) [Address : 01 16]

B	Name	Function	At reset	R	W
0	Port P0 direction register	0 : Port Pi_0 input mode 1 : Port Pi_0 output mode	0	×	○
1		0 : Port Pi_1 input mode 1 : Port Pi_1 output mode	0	×	○
2		0 : Port Pi_2 input mode 1 : Port Pi_2 output mode	0	×	○
3		0 : Port Pi_3 input mode 1 : Port Pi_3 output mode	0	×	○
4		0 : Port Pi_4 input mode 1 : Port Pi_4 output mode	0	×	○
5		0 : Port Pi_5 input mode 1 : Port Pi_5 output mode	0	×	○
6		0 : Port Pi_6 input mode 1 : Port Pi_6 output mode	0	×	○
7		0 : Port Pi_7 input mode 1 : Port Pi_7 output mode	0	×	○

Fig. 2.5.2 Structure of Port P0 direction register

Port P3 direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port P3 direction register (P3D) [Address : 07 16]

B	Name	Function	At reset	R	W
0	Port P3 direction register	0 : Port Pi_0 input mode 1 : Port Pi_0 output mode	0	×	○
1		0 : Port Pi_1 input mode 1 : Port Pi_1 output mode	0	×	○
2		0 : Port Pi_2 input mode 1 : Port Pi_2 output mode	0	×	○
3		0 : Port Pi_3 input mode 1 : Port Pi_3 output mode	0	×	○
4		0 : Port Pi_4 input mode 1 : Port Pi_4 output mode	0	×	○
5		0 : Port Pi_5 input mode 1 : Port Pi_5 output mode	0	×	○
6		0 : Port Pi_6 input mode 1 : Port Pi_6 output mode	0	×	○
7		0 : Port Pi_7 input mode 1 : Port Pi_7 output mode	0	×	○

Note: The 32-pin package versions have nothing to be allocated for the following:
•Bits 5 and 6 of P3D.

Fig. 2.5.3 Structure of Port P3 direction register

Pull-up control register

b7 b6 b5 b4 b3 b2 b1 b0

Pull-up control register (PULL) [Address : 16 16]

B	Name	Function	At reset	R	W
0	$P0_0$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
1	$P0_1$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
2	$P0_2$, $P0_3$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
3	$P0_4$ – $P0_7$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
4	$P3_0$ – $P3_3$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
5	$P3_4$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
6	$P3_5$, $P3_6$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
7	$P3_7$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○

Notes 1: Pins set to output are disconnected from the pull-up control.
2: Keep setting the $P3_5$, $P3_6$ pull-up control bit to "1" (initial value: 0) for the 32-pin package versions.

Fig. 2.5.4 Structure of Pull-up control register

Port P1P3 control register

b7 b6 b5 b4 b3 b2 b1 b0

Port P1P3 control register (P1P3C) [Address : 17 16]

B	Name	Function	At reset	R	W
0	$P3_7$/INT_0 input level selection bit	0 : CMOS level 1 : TTL level	0	○	○
1	$P3_6$/INT_1 input level selection bit (**Note**)	0 : CMOS level 1 : TTL level	0	○	○
2	$P1_0$, $P1_2$,$P1_3$ input level selection bit	0 : CMOS level 1 : TTL level	0	○	○
3	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Note: Keep setting the $P3_6$/INT_1 input level selection bit to "0" (initial value) for the 32-pin package version.

Fig. 2.5.5 Structure of Port P1P3 control register

Timer Y, Z mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer Y, Z mode register (TYZM) [Address : 20 16)

B	Name	Function	At reset	R	W
0	Timer Y operating mode bit	0 : Timer mode 1 : Programmable waveform generation mode	0	○	○
1	Nothing is allocated for this bit. This is a write disabled bit. When this bit is read out, the value is "0".		0	○	×
2	Timer Y write control bit (**Note**)	0 : Write to latch and timer simultaneously 1 : Write to only latch	0	○	○
3	Timer Y count stop bit	0 : Count start 1 : Count stop	0	○	○
4	Timer Z operating mode bits	b5 b4 0 0 : Timer mode 0 1 : Programmable waveform generation mode 1 0 : Programmable one-shot generation mode 1 1 : Programmable wait one-shot generation mode	0	○	○
5			0	○	○
6	Timer Z write control bit (**Note**)	0 : Write to latch and timer simultaneously 1 : Write to only latch	0	○	○
7	Timer Z count stop bit	0 : Count start 1 : Count stop	0	○	○

Note: When modes other than the timer mode, set these bits to "1".

Fig. 2.5.6 Structure of Timer Y, Z mode register

Prescaler Y, Prescaler Z

b7 b6 b5 b4 b3 b2 b1 b0

Prescaler Y (PREY) [Address : 21 16]
Prescaler Z (PREZ) [Address : 25 16]

B	Function	At reset	R	W
0	•Set a count value of each prescaler. •The value set in this register is written to both each prescaler and the corresponding prescaler latch at the same time. •When this register is read out, the count value of the corresponding prescaler is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Fig. 2.5.7 Structure of Prescaler Y, Prescaler Z

Timer Y secondary, Timer Z secondary

b7 b6 b5 b4 b3 b2 b1 b0

Timer Y secondary, Timer Z secondary (TYS, TZS) [Address : 22 16, 2616]

B	Function	At reset	R	W
0	•Set a count value of the corresponding timer. •The value set in this register is written to the corresponding secondary latch at the same time. •These are read disabled bits. When these bits are read out, the values are undefined.	1	×	○
1		1	×	○
2		1	×	○
3		1	×	○
4		1	×	○
5		1	×	○
6		1	×	○
7		1	×	○

Fig. 2.5.8 Structure of Timer Y secondary, Timer Z secondary

Timer Y primary, Timer Z primary

b7 b6 b5 b4 b3 b2 b1 b0

Timer Y primary, Timer Z primary (TYP, TZP) [Address : 23 16, 2716]

B	Function	At reset	R	W
0	•Set a count value of the corresponding timer. •When the corresponding timer is stopped, the value set in this register is written to both the corresponding primary latch and the corresponding timer at the same time. •When the corresponding timer is operating, the value set in this register is written as follows; timer write control bit = 0: the value is written to both the corresponding primary latch and the corresponding timer at the same time. timer write control bit = 1: the value is written to the corresponding primary latch. •When these bits are read out, the count value of the corresponding timer is read out (Note).	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Note: The primary count value is read out at the primary interval, the secondary count value is read out at the secondary interval.

Fig. 2.5.9 Structure of Timer Y primary, Timer Z primary

Timer Y, Z waveform output control register

b7 b6 b5 b4 b3 b2 b1 b0

Timer Y, Z waveform output control register (PUM) [Address : 24 16]

B	Name	Function	At reset	R	W
0	Timer Y primary waveform extension control bit	0 : Waveform not extended 1 : Waveform extended	0	○	○
1	Timer Y secondary waveform extension control bit	0 : Waveform not extended 1 : Waveform extended	0	○	○
2	Timer Z primary waveform extension control bit	0 : Waveform not extended 1 : Waveform extended	0	○	○
3	Timer Z secondary waveform extension control bit	0 : Waveform not extended 1 : Waveform extended	0	○	○
4	Timer Y output level latch	0 : "L" output 1 : "H" output	0	○	○
5	Timer Z output level latch	0 : "L" output 1 : "H" output	0	○	○
6	INT_0 pin one-shot trigger control bit (**Note**)	0 : INT_0 pin one-shot trigger invalid 1 : INT_0 pin one-shot trigger valid	0	○	○
7	INT_0 pin one-shot trigger active edge selection bit (**Note**)	0 : Falling edge trigger 1 : Rising edge trigger	0	○	○

Note: Stop timer Z to change the values of these bits.

Fig. 2.5.10 Structure of Timer Y, Z waveform output control register

One-shot start register

b7 b6 b5 b4 b3 b2 b1 b0

One-shot start register (ONS) [Address : 2A 16]

B	Name	Function	At reset	R	W
0	Timer Z one-shot start bit	0 : One-shot stop 1 : One-shot start	0	○	○
1	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
2			0	○	×
3			0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Fig. 2.5.11 Structure of One-shot start register

Timer count source set register

b7	b6	b5	b4	b3	b2	b1	b0
	0						

Timer count source set register (TCSS) [Address : 2E 16]

B	Name	Function	At reset	R	W
0	Timer X count source selection bits	b1 b0 0 0 : f(XIN)/16 0 1 : f(XIN)/2 1 0 : f(XIN) (**Note 1**) 1 1 : Not available	0	○	○
1			0	○	○
2	Timer Y count source selection bits	b3 b2 0 0 : f(XIN)/16 0 1 : f(XIN)/2 1 0 : Ring oscillator output (**Note 2**) 1 1 : Not available	0	○	○
3			0	○	○
4	Timer Z count source selection bits	b5 b4 0 0 : f(XIN)/16 0 1 : f(XIN)/2 1 0 : Timer Y underflow 1 1 : Not available	0	○	○
5			0	○	○
6	Fix this bit to "0".		0	○	○
7	Nothing is allocated for this bit. This is a write disabled bit. When this bit is read out, the value is "0".		0	○	×

Notes 1: f(XIN) can be used as timer X count source only when using a ceramic oscillator or ring oscillator.
Do not use it at RC oscillation.
2: System operates using a ring oscillator as a count source by setting the ring oscillator to oscillation enabled by bit 3 of CPUM.

Fig. 2.5.12 Structure of Timer count source set register

Interrupt edge selection register

b7	b6	b5	b4	b3	b2	b1	b0

Interrupt edge selection register (INTEDGE) [Address : 3A 16]

B	Name	Function	At reset	R	W
0	INT0 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
1	INT1 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
2	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
3			0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7	P00 key-on wakeup enable bit	0 : Key-on wakeup enabled 1 : Key-on wakeup disabled	0	○	○

Fig. 2.5.13 Structure of Interrupt edge selection register

CPU mode register

b7 b6 b5 b4 b3 b2 b1 b0

CPU mode register (CPUM) [Address : 3B 16]

B	Name	Function	At reset	R	W
0	Processor mode bits (**Note 1**)	b1 b0 0 0 : Single-chip mode 0 1 : Not available 1 0 : Not available 1 1 : Not available	0	○	○
1			0	○	○
2	Stack page selection bit	0 : 0 page 1 : 1 page	0	○	○
3	Ring oscillator oscillation control bit	0 : Ring oscillator oscillation enabled 1 : Ring oscillator oscillation stop	0	○	○
4	X_{IN} oscillation control bit	0 : Ceramic or RC oscillation enabled 1 : Ceramic or RC oscillation stop	0	○	○
5	Oscillation mode selection bit (**Note 1**)	0 : Ceramic oscillation 1 : RC oscillation	0	○	○
6	Clock division ratio selection bits	b7 b6 0 0 : $\phi = f(X_{IN})/2$ (high-speed mode) 0 1 : $\phi = f(X_{IN})/8$ (middle-speed mode) 1 0 : Applied from ring oscillator 1 1 : $\phi = f(X_{IN})$ (double-speed mode) (**Note 2**)	0	○	○
7			1	○	○

Notes 1: The bit can be rewritten only once after releasing reset. After rewriting it is disable to write any data to the bit. However, by reset the bit is initialized and can be rewritten, again.
(It is not disable to write any data to the bit for emulator MCU "M37540RSS".)

2: These bits are used only when a ceramic oscillation is selected.
Do not use these when an RC oscillation is selected.

Fig. 2.5.14 Structure of CPU mode register

Interrupt request register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt request register 1 (IREQ1) [Address : 3C $_{16}$]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Serial I/O1 transmit interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	INT_0 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	INT_1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	Key-on wake up interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	$CNTR_0$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	$CNTR_1$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻

✻: These bits can be cleared to “0” by program, but cannot be set to “1”.

Fig. 2.5.15 Structure of Interrupt request register 1

Interrupt request register 2

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt request register 2 (IREQ2) [Address : 3D $_{16}$]

B	Name	Function	At reset	R	W
0	Timer Y interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Timer Z interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	Timer A interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	AD converter interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are “0”.		0	○	×
7			0	○	×

✻: These bits can be cleared to “0” by program, but cannot be set to “1”.

Fig. 2.5.16 Structure of Interrupt request register 2

2.5 Timer Y and timer Z

Interrupt control register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 1 (ICON1) [Address : 3E 16]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Serial I/O1 transmit interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	INT_0 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	INT_1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Key-on wake up interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	$CNTR_0$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	$CNTR_1$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Fig. 2.5.17 Structure of Interrupt control register 1

Interrupt control register 2

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 2 (ICON2) [Address : 3F 16]

B	Name	Function	At reset	R	W
0	Timer Y interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer Z interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer A interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	AD conversion interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
7			0	○	×

Fig. 2.5.18 Structure of Interrupt control register 2

2.5.3 Timer mode (timer Y and timer Z)

The basic operation of Timer Y and Timer Z are the same. In this section, Timer Y is explained.

(1) Operation description

Prescaler Y counts the count source selected by the timer Y count source selection bits. Each time the count clock is input, the contents of Prescaler Y is decremented by 1.

When the contents of Prescaler Y reach "00_{16}", an underflow occurs at the next count clock, and the prescaler Y latch is reloaded into Prescaler Y and count continues. The division ratio of Prescaler Y is 1/(n+1) provided that the value of Prescaler Y is n.

The contents of Timer Y is decremented by 1 each time the underflow signal of Prescaler Y is input. When the contents of Timer Y reach "00_{16}", an underflow occurs at the next count clock, and the timer Y primary latch is reloaded into Timer Y and count continues.

(In the timer mode, the contents of timer Y primary latch is counted. Timer Y secondary latch is not used in this mode.)

The division ratio of Timer Y is 1/(m+1) provided that the value of Timer Y is m. Accordingly, the division ratio of Prescaler Y and Timer Y is provided as follows that the value of Prescaler Y is n and the value of Timer Y is m.

$$\text{Division ratio} = \frac{1}{(n+1) \times (m+1)}$$

In the timer mode, writing to "latch only" or "latches and Prescaler Y and timer Y primary" can be selected by the setting value of the timer Y write control bit.

Timer Y can stop counting by setting "1" to the timer Y count stop bit.

Also, when timer Y underflows, the timer Y interrupt request bit is set to "1".

Timer Y reloads the value of latch when counting is stopped by the timer count stop bit.

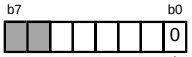
(When timer is read out while timer is stopped, the value of latch is read. The value of timer can be read out only while timer is operating.)

(2) Timer mode setting method

Figure 2.5.19 shows the setting method for timer mode of Timer Y.

When Timer Z is used, registers are set by the same method.

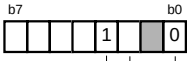
Process 1: Disable timer Y interrupt.



Interrupt control register 2 (ICON2) [Address $3F_{16}$]

Timer Y interrupt disabled

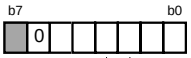
Process 2: Set timer Y, Z mode register.



Timer Y, Z mode register (TYZM) [Address 20_{16}]

Timer mode
Timer Y write control bit
0: Write to latch and timer simultaneously
1: Write to only latch
Timer Y count stop

Process 3: Set timer Y count source (**Note 1**)



Timer count source set register (TCSS) [Address $2E_{16}$]

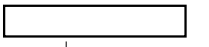
Timer Y count source selection bits

b3 b2
0 0 : f(X_{IN})/16
0 1 : f(X_{IN})/2
1 0 : Ring oscillator output (**Note 2**)
1 1 : Not available

Notes 1: For timer Z, f(X_{IN}), f(X_{IN})/2, or timer Y underflow can be selected.
2: Set the ring oscillator oscillation to be enabled by bit 3 (ring oscillator oscillation control bit) of CPU mode register.

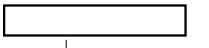
Process 4: Set the count value to timer Y.

• Set the count value to prescaler Y and timer Y primary



Prescaler Y (PREY) (Address 21_{16})

Count value

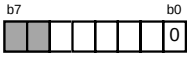


Timer Y primary (TYP) (Address 23_{16})

Count value

Note: In the timer mode, the timer Y secondary is not used.

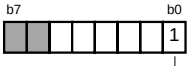
Process 5: In order not to execute the no requested interrupt processing, set “0” (no requested) to the timer Y interrupt request bit.



Interrupt request register 2 (IREQ2) [Address $3D_{16}$]

No timer Y interrupt request issued

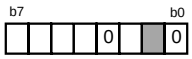
Process 6: When timer Y interrupt is used, set “1” (interrupt enabled) to the timer Y interrupt enable bit.



Interrupt control register 2 (ICON2) [Address $3F_{16}$]

Timer Y interrupt enabled

Process 7: Start counting of timer Y.



Timer Y, Z mode register (TYZM) [Address 20_{16}]

Timer Y count start

Fig. 2.5.19 Setting method for timer mode

(3) Application example of timer mode

Outline: Pulses generated corresponding to the water flow rate are counted for a fixed period (100 ms), and the water flow rate during this period is calculated.

Specifications: Pulses generated corresponding to the water flow rate are input to the $P1_4/CNTR_1$ pin and counted using timer X.

The contents of timer X are read in the timer Y interrupt processing routine generated after 100 ms from the start of counting pulses, and the water flow rate during 100 ms is calculated.

Operation clock: $f(X_{IN})$ = 8 MHz, high-speed mode

Figure 2.5.20 shows an example of peripheral circuit, Figure 2.5.21 shows the method of measuring water flow rate, and Figure 2.5.21 shows an example of control procedure.

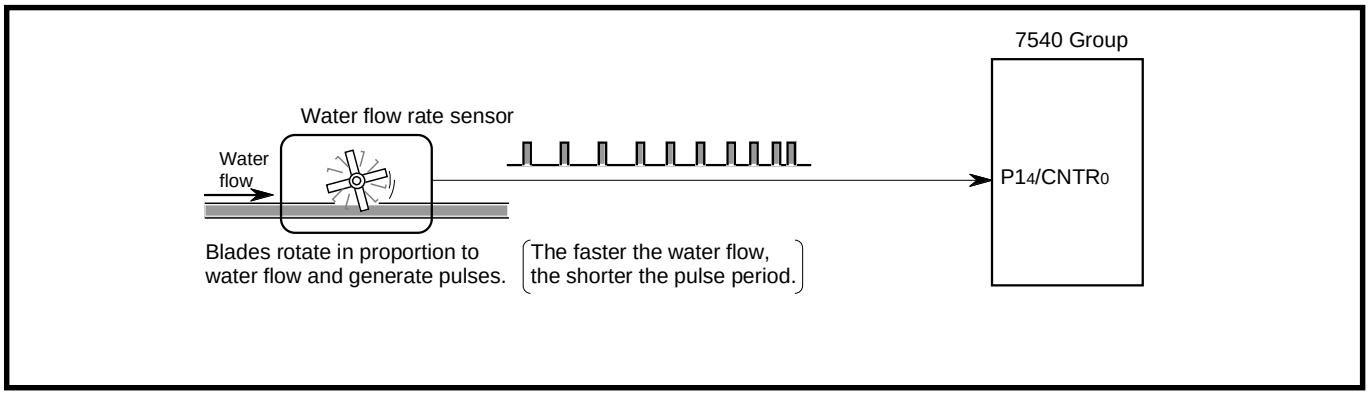


Fig. 2.5.20 Example of peripheral circuit

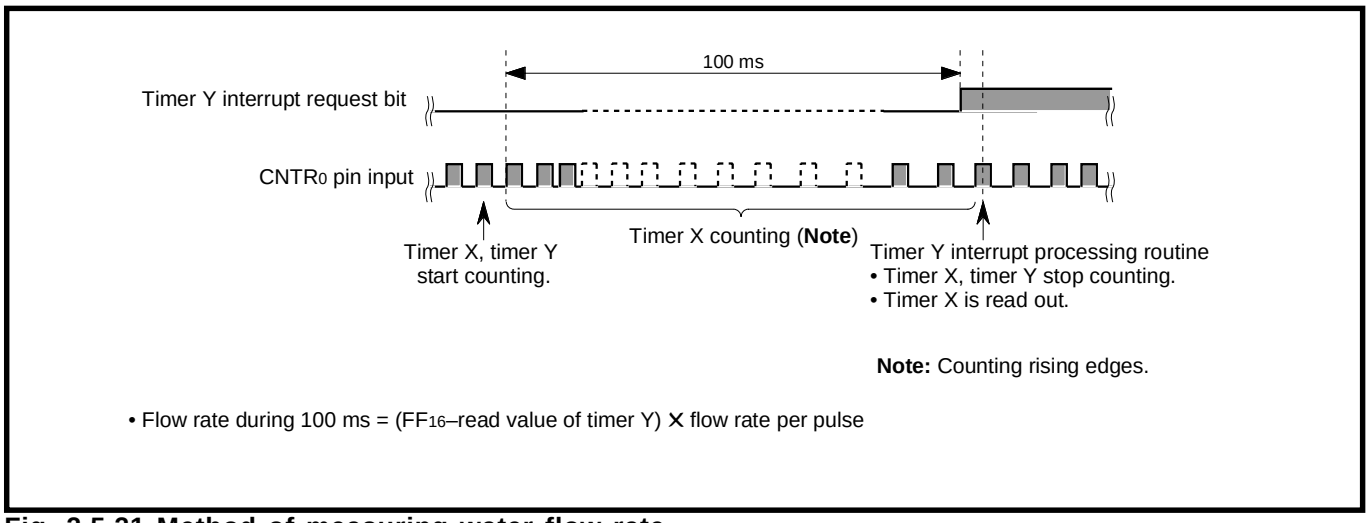


Fig. 2.5.21 Method of measuring water flow rate

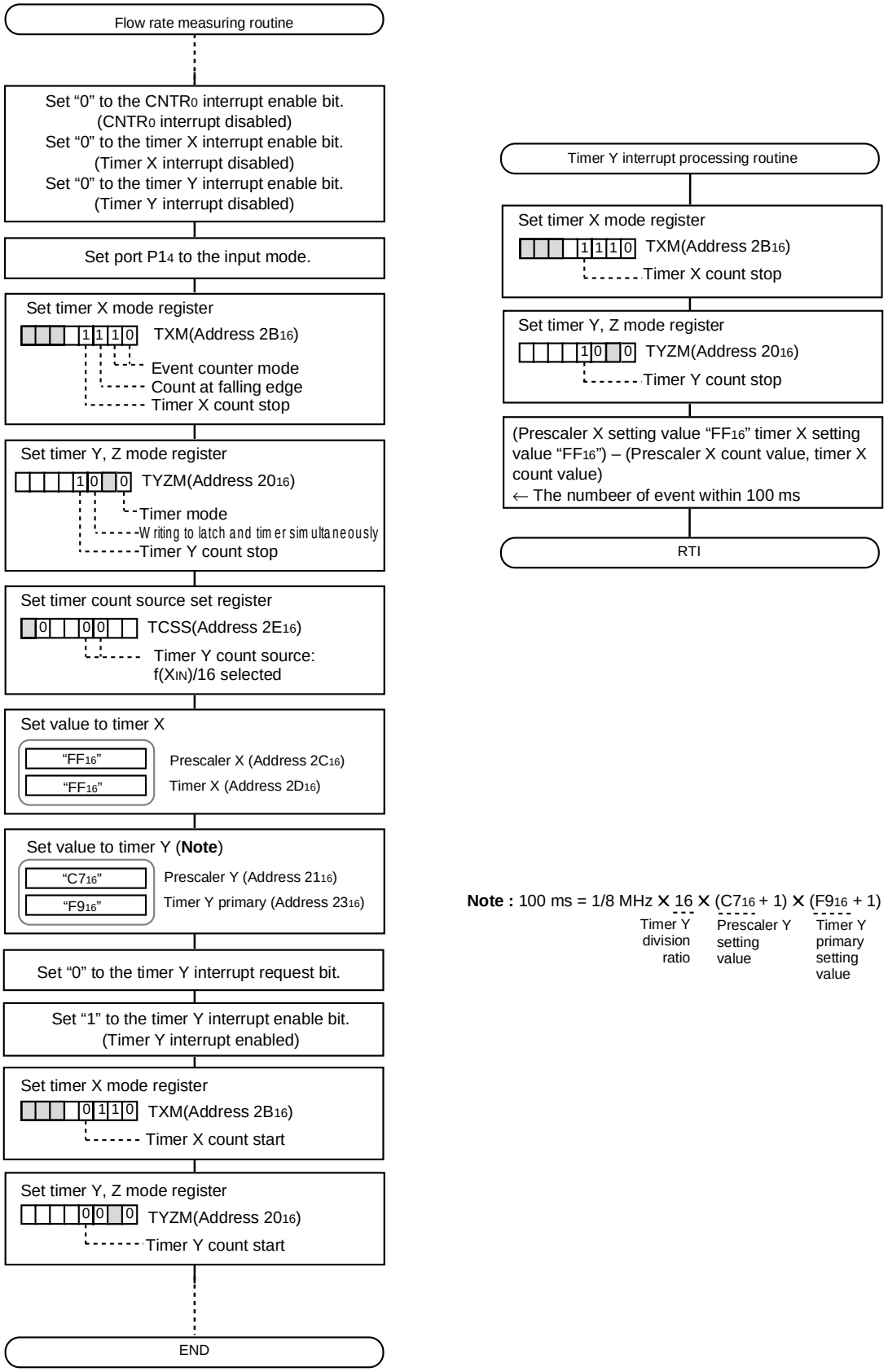


Fig. 2.5.22 Example of control procedure

2.5.4 Programmable waveform generation mode (timer Y and timer Z)

The basic operation of Timer Y and Timer Z are the same. In this section, Timer Y is explained.

(1) Operation description

In the programmable waveform generation mode, timer counts the setting value of timer Y primary (TYP) and the setting value of timer Y secondary (TYS) alternately, the waveform whose polarity is inverted each time Timer Y underflows is output from $P0_1/TY_{OUT}$ pin.

When using this mode, be sure to set "1" to the timer Y write control bit to select "write to latch only". Also, set the port $P0_1$ direction registers to output mode.

The active edge of output waveform is set by the timer Y output level latch. When "0" is set to the timer Y output level latch, "H" interval by the setting value of TYP or "L" interval by the setting value of TYS is output alternately. When "1" is set to the timer Y output level latch, "L" interval by the setting value of TYP or "H" interval by the setting value of TYS is output alternately.

Also, in this mode, the primary interval and the secondary interval of the output waveform can be extended respectively for 0.5 cycle of timer count source clock by setting the timer Y primary waveform extension control bit (b2) and the timer Y secondary waveform extension control bit (b3) of PUM to "1". As a result, the waveforms of more accurate resolution can be output.

When b2 and b3 of PUM are used, the frequency and duty of the output waveform are as follows;

$$\text{Waveform frequency: FYOUT} = \frac{2 \times (\text{TMYCL})}{(2 \times (\text{TYP}+1)+\text{EXPYP})+(2 \times (\text{TYS}+1)+\text{EXPYS}))}$$

$$\text{Duty: DYOUT} = \frac{2 \times (\text{TYP}+1)+\text{EXPYP}}{(2 \times (\text{TYP}+1)+\text{EXPYP})+(2 \times (\text{TYS}+1)+\text{EXPYS}))}$$

TMYCL: Timer Y count source (frequency)
TYP: Timer Y primary
TYS: Timer Y secondary
EXPYP (1 bit): Timer Y primary waveform extension control bit
EXPYS (1 bit): Timer Y secondary waveform extension control bit

In the programmable waveform generation mode, when values of the TYP, TYS, EXPYP and EXPYS are changed, the output waveform is changed at the beginning (timer Y primary waveform interval) of waveform period.

When the count values are changed, set values to the TYS, EXPYP and EXPYS first. After then, set the value to TYP. The values are set all at once at the beginning of the next waveform period when the value is set to TYP. (When writing at timer stop is executed, writing to TYP at last is required.)

Timer Y can stop counting by setting "1" to the timer Y count stop bit.

Also, when timer Y underflows, the timer Y interrupt request bit is set to "1".

Timer Y reloads the value of latch when counting is stopped by the timer Y count stop bit. (When timer is read out while timer is stopped, the value of latch is read. The value of timer can be read out only while timer is operating.)

Notes 1: In the programmable waveform generation mode, values of TYS, EXPYP, and EXPYS are valid by writing to TYP because the setting to them is executed all at once by writing to TYP. Even when changing TYP is not required, write the same value again.

2: In the programmable waveform generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TYP and the timing of timer underflow during the secondary interval simultaneously.
An example of a measurement is shown below.
ex.) The underflow by the primary and the underflow by secondary are stored by polling etc. using timer Y interrupt.
Writing is performed in by judging that there is no problem if the underflow by secondary is completed with reference to primary write operation before.
(Depending on a primary and a secondary setting values, and primary write timing, it may be impossible.)

3: The waveform extension function by the timer Y waveform extension control bits can be used only when "00_{16}" is set to Prescaler Y.
When the value other than "00_{16}" is set to Prescaler Y, be sure to set "0" to EXPYP and EXPYS.
The waveform extension function by the timer Z waveform extension control bits can be used only when "00_{16}" is set to Prescaler Z. When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.

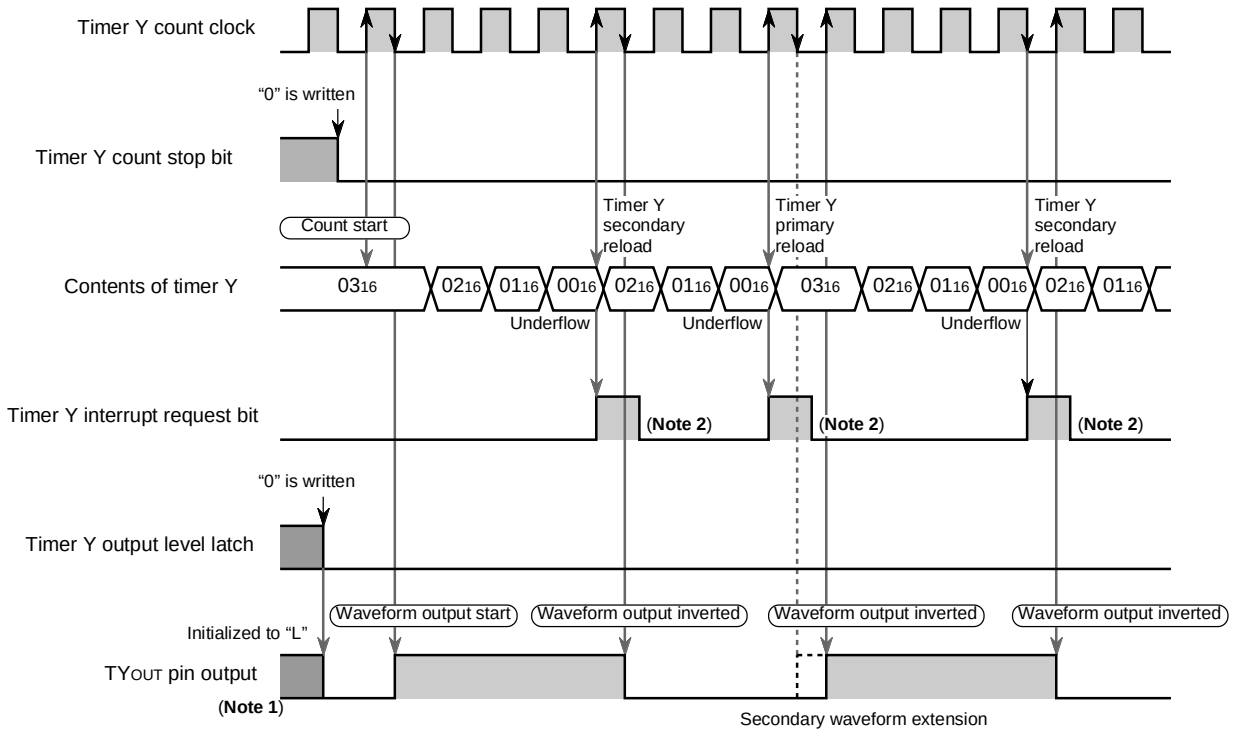
4: When using this mode, be sure to set "1" to the timer Y write control bit to select "write to latch only".

5: When TYS is read out, the undefined value is read out. However, while timer Y counts the setting value of TYS, the count value during the secondary interval can be obtained by reading the timer Y primary.

6: In order to use TY_{OUT} pin, set "1" to bit 1 of the port P0 direction register (output mode).

Figure 2.5.23 shows the timing diagram of the programmable waveform generation mode.

● When "03_{16}" is set to TYP and "02_{16}" is set to TYS.



Notes 1: In this case, timer Y primary waveform is not extended, timer Y secondary waveform is extended.
2: In this time, "0" is written to the timer Y interrupt request bit or the timer Y interrupt request bit is cleared to "0" by accepting the timer Y interrupt request.

Fig. 2.5.23 Timing diagram of programmable waveform generation mode

(2) Programmable waveform generation mode setting method

Figure 2.5.24 and Figure 2.5.25 show the setting method for programmable waveform generation mode of timer Y.

When timer Z is used, registers are set by the same method.

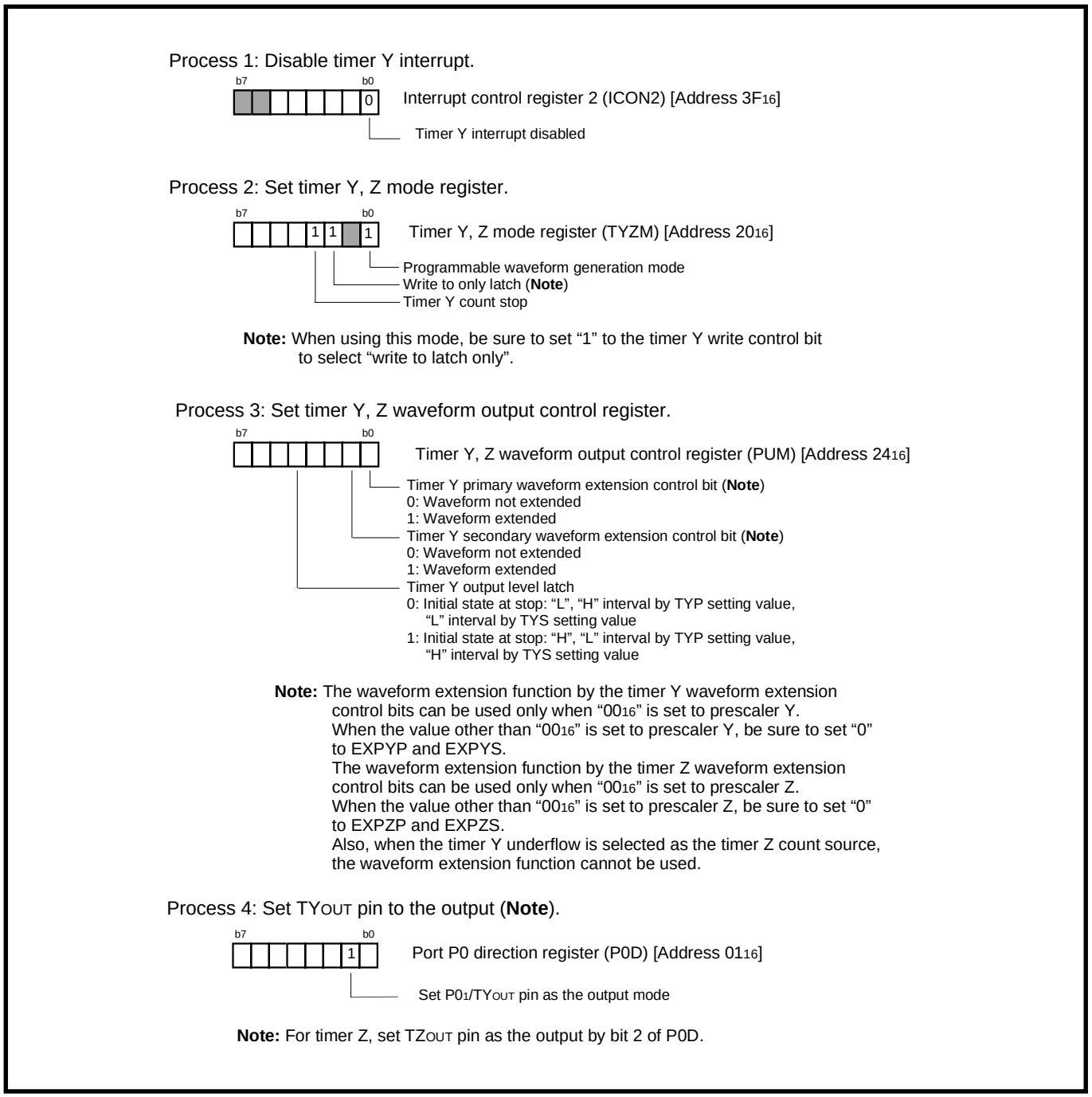
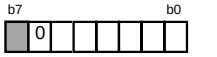


Fig. 2.5.24 Setting method for programmable waveform generation mode (1)

Process 5: Set timer Y count source (**Note 1**).



Timer count source set register (TCSS) [Address $2E_{16}$]

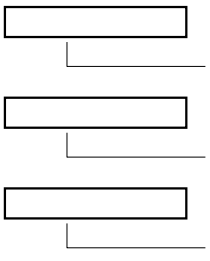
Timer Y count source selection bits

b3 b2
0 0 : f(X_{IN})/16
0 1 : f(X_{IN})/2
1 0 : Ring oscillator output (**Note 2**)
1 1 : Not available

Notes 1: For Timer Z, f(X_{IN})/16, f(X_{IN})/2, or timer Y underflow can be selected. However, when the timer Z waveform expansion function is used, do not select the timer Y underflow for the timer Z count source.
2: Set the ring oscillator oscillation to be enabled by bit 3 (ring oscillator oscillation control bit) of CPU mode register.

Process 6: Set the count value to timer Y (**Note 1**).

• Set the count value to prescaler Y, timer Y secondary and timer Y primary



Prescaler Y (PREY) (Address 21_{16}) (**Note 2**)

Count value

Timer Y secondary (TYS) (Address 22_{16})

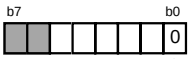
Count value

Timer Y primary (TYP) (Address 23_{16}) (**Notes 3, 4**)

Count value

Notes 1: In the programmable waveform generation mode, values of TYS, EXPYP, and EXPYS are valid by writing to TYP. Even when changing TYP is not required, write the same value again.
2: When the timer Y waveform extension function is used, be sure to set "00_{16}" to prescaler Y.
3: In the programmable waveform generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TYP and the timing of timer Y underflow during the secondary interval simultanesously.
4: Count values of the primary interval and secondary interval can be checked by reading the TYP (TYS is undefined at read).

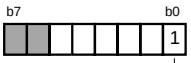
Process 7: In order not to execute the no requested interrupt processing, set "0" (no requested) to the timer Y interrupt request bit.



Interrupt request register 2 (IREQ2) [Address $3D_{16}$]

No timer Y interrupt request issued

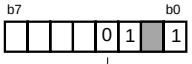
Process 8: When Timer Y interrupt is used, set "1" (interrupt enabled) to the timer Y interrupt enable bit.



Interrupt control register 2 (ICON2) [Address $3F_{16}$]

Timer Y interrupt enabled

Process 9: Start counting of timer Y.



Timer Y, Z mode register (TYZM) [Address 20_{16}]

Timer Y count start

Fig. 2.5.25 Setting method for programmable waveform generation mode (2)

(3) Application example of programmable waveform generation mode

Outline: The waveform extension function is used and the waveform output is executed.

Specifications: The "H" width generated by TYP and the "L" width generated by TYS are output. Set each waveform extension function to be valid, and set the duty ratio to be 2:1. The frequency is 40 kHz.
Operation clock: $f(X_{IN})$ = 8 MHz, high-speed mode

Figure 2.5.26 shows an example of waveform output and Figure 2.5.27 shows an example of control procedure.

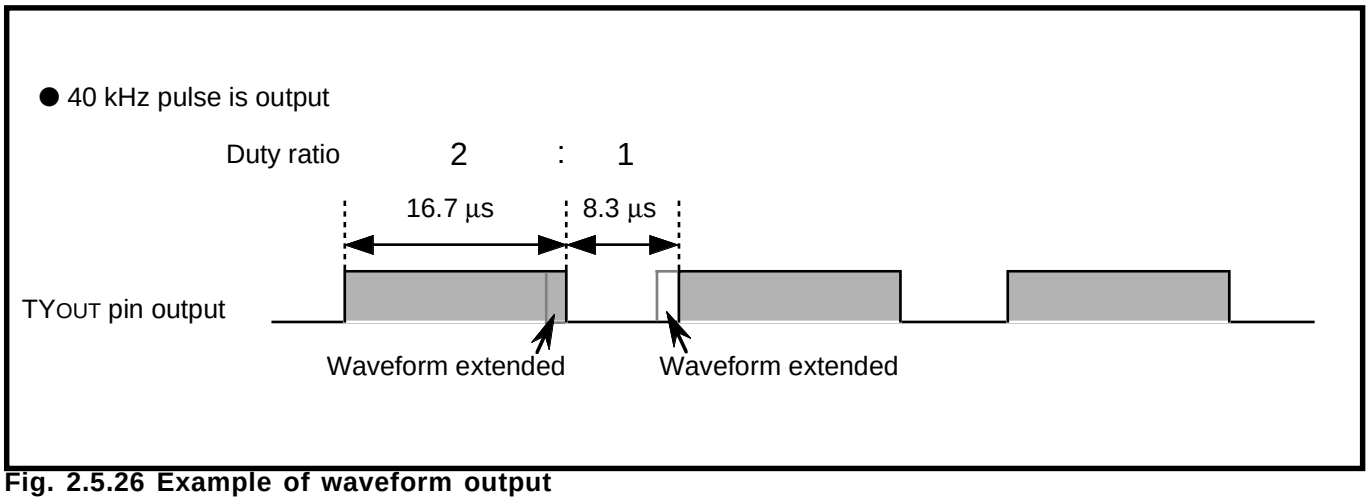
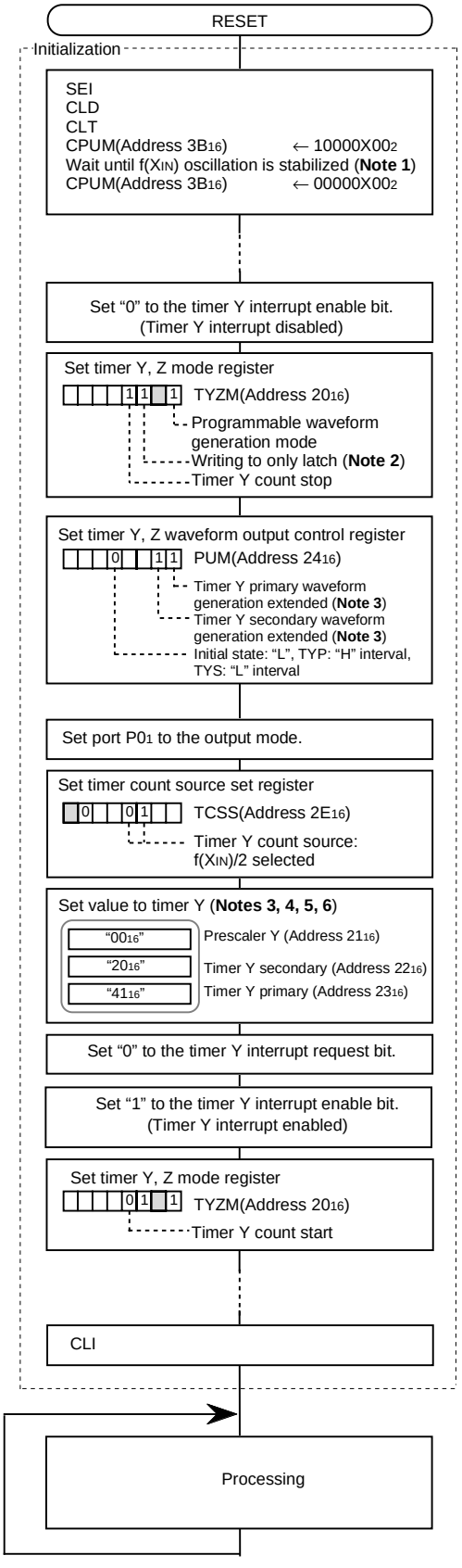


Fig. 2.5.26 Example of waveform output



X: This bit is not used here. Set it to "0" or "1" arbitrary.

Notes 1: For the concrete time, ask the oscillator manufacture.
2: When using this mode, be sure to select "write to latch only".
3: The waveform extension function by the timer Y waveform extension control bits can be used only when "0016" is set to prescaler Y.
When the value other than "0016" is set to prescaler Y, be sure to set "0" to EXPYP and EXPYS.
4: In the programmable waveform generation mode, values of TYS, EXPYP, and EXPYS are valid by writing to TYP. Even when changing TYP is not required, write the same value again.
5: In the programmable waveform generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TYP and the timing of timer Y underflow during the secondary interval simultanesously.
6: Count values of the primary interval and secondary interval can be checked by reading TYP (TYS is undefined at read).

Fig. 2.5.27 Example of control procedure

2.5.5 Programmable one-shot generation mode (timer Z)

(1) Operation description

In the programmable one-shot generation mode, the one-shot pulse by the setting value of timer Z primary can be output from $P0_2/TZ_{OUT}$ pin by software or external trigger to the $P3_7/INT_0$ pin. When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only". Also, set the port $P0_2$ direction registers to output mode. In this mode, the timer Z secondary (TZS) is not used.

The active edge of output waveform is set by the timer Z output level latch. When "0" is set to the timer Z output level latch, "H" pulse during the interval of the timer Z primary (TZP) setting value is output. When "1" is set to the timer Z output level latch, "L" pulse during the interval of the TZP setting value is output.

Also, in this mode, the interval of the one-shot pulse output can be extended for 0.5 cycle of timer count source clock by setting the timer Z primary waveform extension control bit (EXPZP) to "1". As a result, the waveforms of more accurate resolution can be output.

During the one-shot pulse output interval, the one-shot pulse output can be stopped forcibly by writing "0" to the timer Z one-shot start bit.

In the programmable one-shot generation mode, when the count values are changed, set value to the EXPZP first. After then, set the value to TZP. The values are set all at once at the beginning of the next one-shot pulse when the value is set to TZP. (Even when writing at timer stop is executed, writing to TZP at last is required.)

Timer Z can stop counting by setting "1" to the timer Z count stop bit.

Also, when timer Z underflows, the timer Z interrupt request bit is set to "1".

Timer Z reloads the value of latch when counting is stopped by the timer Z count stop bit.

(When timer is read out while timer is stopped, the value of latch is read. The value of timer can be read out only while timer is operating.)

Notes 1: In the programmable one-shot generation mode, the value of EXPZP becomes valid by writing to TZP. Even when changing TZP is not required, write the same value again.

2: In the programmable one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow simultaneously.

An example of a measurement is shown below.

ex.) The underflow of timer is stored by polling etc. using timer Z interrupt.

Writing to primary is performed in by judging that there is no problem if the underflow by secondary is completed with reference to primary write operation before.

(Depending on a primary setting value, primary write timing, software and timing of external trigger to INT_0 pin, it may be impossible.)

3: The waveform extension function by the timer Z waveform extension control bits can be used only when "00_{16}" is set to Prescaler Z.

When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.

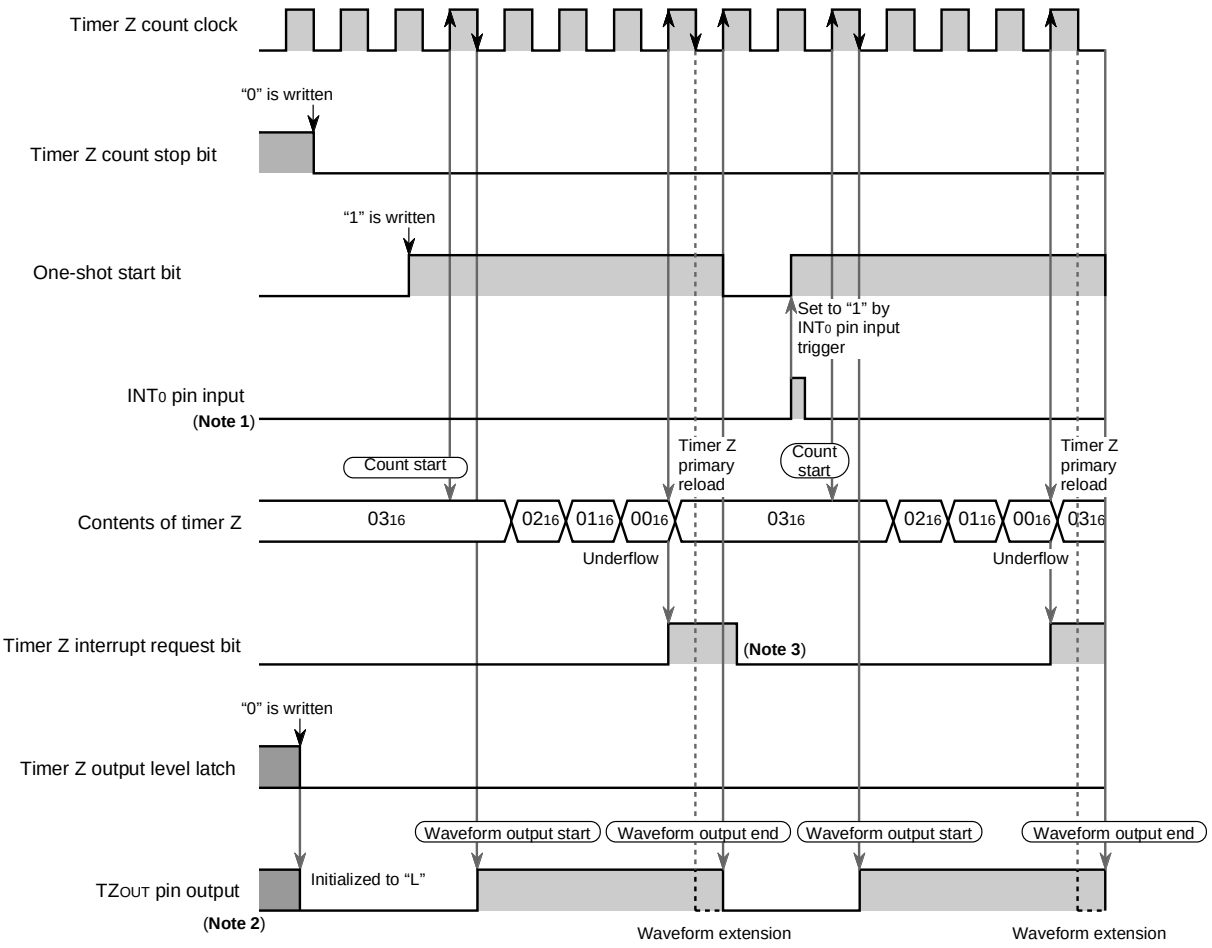
4: When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only".

5: In order to use TZ_{OUT} pin, set "1" to bit 2 of the port P0 direction register (output mode).

6: Stop Timer Z to change the INT_0 pin one-shot trigger control bit and INT_0 pin one-shot trigger active edge selection bit.

Figure 2.5.28 shows the timing diagram of the programmable one-shot generation mode.

● When "03_{16}" is set to TZP



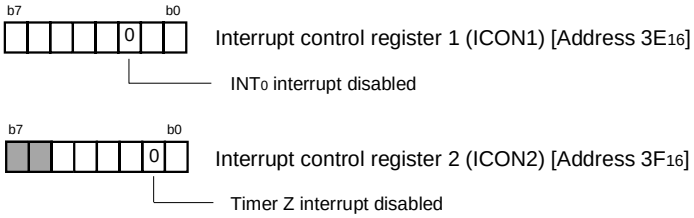
Notes 1: In this case, INT0 pin one-shot trigger valid.
2: In this case, timer Z primary waveform is extended.
3: In this time, "0" is written to the timer Z interrupt request bit or the timer Z interrupt request bit is cleared to "0" by accepting the timer Z interrupt request.

Fig. 2.5.28 Timing diagram of programmable one-shot generation mode

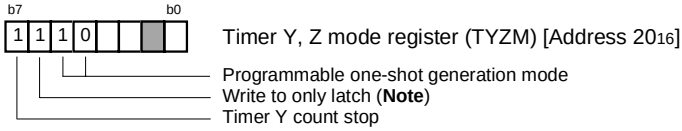
(2) Event counter mode setting method

Figure 2.5.29 to Figure 2.5.31 show the setting method for programmable one-shot generation mode of timer Z.

Process 1: Disable the interrupt.

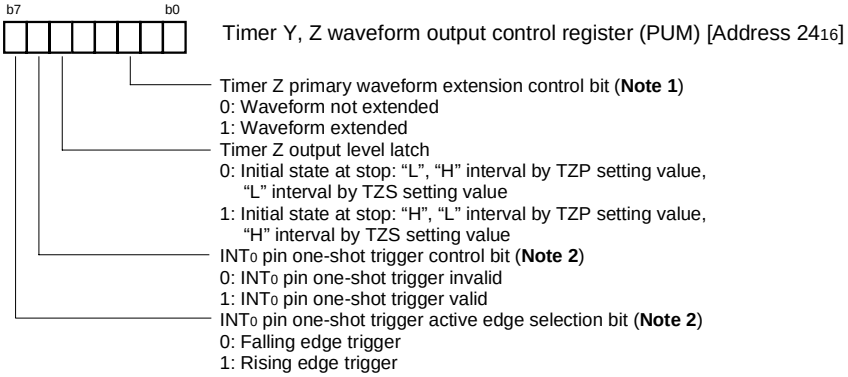


Process 2: Set timer Y, Z mode register.



Note: When using this mode, be sure to select "write to latch only".

Process 3: Set timer Y, Z waveform output control register.



Notes 1: The waveform extension function by the timer Z waveform extension control bits can be used only when "0016" is set to prescaler Z.
When the value other than "0016" is set to prescaler Z, be sure to set "0" to EXPZP.
Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.
2: Stop timer Z to change these bits.

Process 4: Set TZ_{OUT} pin to the output (**Note**).

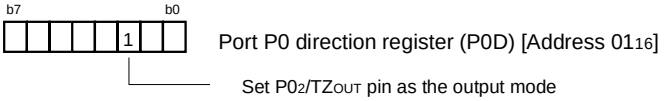
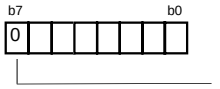


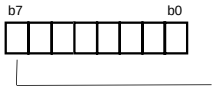
Fig. 2.5.29 Setting method for programmable one-shot generation mode (1)

Process 5: When the trigger by INT_0 pin input is selected:
Set port P3 direction register, pull-up control register and port P1P3 control register



Port P3 direction register (P3D) [Address 07_{16}]

Set $P3_7/INT_0$ pin as the input mode



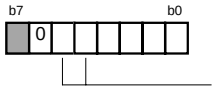
Pull-up control register (PULL) [Address 16_{16}]

$P3_7$ pull-up control bit
0: Pull-up Off
1: Pull-up On

Port P1P3 control register (P1P3C) [Address 17_{16}]

$P3_7/INT_0$ input level selection bit
0: CMOS level
1: TTL level

Process 6: Set the timer Z count source.



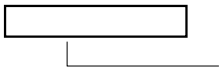
Timer count source set register (TCSS) [Address $2E_{16}$]

Timer Z count source selection bits
b5 b4
0 0 : $f(X_{IN})/16$
0 1 : $f(X_{IN})/2$
1 0 : Timer Y underflow (**Note)**
1 1 : Not available

Note: When the timer Z waveform extension function is used,
do not select the timer Y underflow as the timer Z count source.

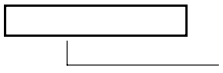
Process 7: Set the one-shot pulse width (**Note 1**).

• Set the count value to prescaler Z and timer Z primary



Prescaler Z (PREZ) [Address 25_{16}] (**Note 2**)

Count value

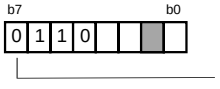


Timer Z primary (TZP) [Address 27_{16}] (**Note 3**)

Count value

Notes 1: In the programmable one-shot generation mode, TZS is not used.
When the count setting value is changed, value of EXPZP is valid by writing to TZP. Even when changing TZP is not required, write the same value again.
2: When the timer Z waveform extension function is used, be sure to set "00_{16}" to prescaler Z.
3: In the programmable one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer Z underflow simultanesously.

Process 8: Set the standby state to accept the one-shot start trigger (**Note**).



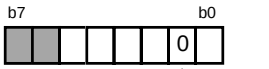
Timer Y, Z mode register (TYZM) [Address 20_{16}]

Timer Z count start

Note: When the INT_0 pin one-shot trigger control bit of PUM is set to"valid",
timer Z counting is started by the input of trigger to INT_0 pin after this setting.

Fig. 2.5.30 Setting method for programmable one-shot generation mode (2)

Process 9: In order not to execute the no requested interrupt processing, set “0” (no requested) to the timer Z interrupt request bit.



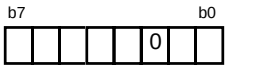
Interrupt request register 2 (IREQ2) [Address $3D_{16}$]

No Timer Z interrupt request issued

- When the INT_0 pin one-shot trigger control bit of PUM is set to “valid” and the INT_0 interrupt is used, set the following;

Interrupt edge selection register (INTEDGE) [Address $3A_{16}$]

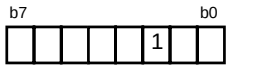
INT_0 interrupt edge selection bit
0: Falling edge active
1: Rising edge active



Interrupt request register 1 (IREQ1) [Address $3C_{16}$]

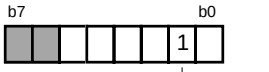
No INT_0 interrupt request issued

Process 10: When the interrupt is used, set “1” (interrupt enabled) to the corresponding interrupt enable bit.



Interrupt control register 1 (ICON1) [Address $3E_{16}$]

INT_0 interrupt enabled (**Note**)



Interrupt control register 2 (ICON2) [Address $3F_{16}$]

Timer Y interrupt enabled

Note: When the INT_0 pin one-shot trigger control bit is set to “valid”, the INT_0 interrupt can be accepted after this setting.

Process 11: Start counting of timer Z.

One-shot start register (ONS) [Address $2A_{16}$]

Timer Z one-shot start (**Note**)

- When the INT_0 pin one-shot trigger control bit of PUM is set to “valid”, the timer Z count is started by input of trigger to the INT_0 pin.

Note: Pulse is output from TZ_{OUT} pin. After output, this bit is initialized to “0”.

Fig. 2.5.31 Setting method for programmable one-shot generation mode (3)

(3) Application example of programmable one-shot generation mode

Outline: The phase control signal to the load is output by using the programmable one-shot generation mode of Timer Z.

Specifications: The phase control signal to the load is output from the $P0_2/TZ_{OUT}$ pin using the programmable one-shot generation mode of timer Z.

- Count source: $f(X_{IN})/16$
- Rising edges of the signal input to the $P3_7/INT_0$ pin from the trigger detection circuit are detected.
- A triac is turned on at the "H" level.
  The period of the feedback signal input from the load is measured, analyzed, and used to adjust the phase control signal.
  Operation clock: $f(X_{IN})$ = 8 MHz, high-speed mode

For the measurement of the period of the feedback signal, refer to the period measurement mode of the using timer.

Figure 2.5.32 shows an example of peripheral circuit, Figure 2.5.33 shows an example of an operation timing, and Figure 2.5.34 shows an example of a control procedure.

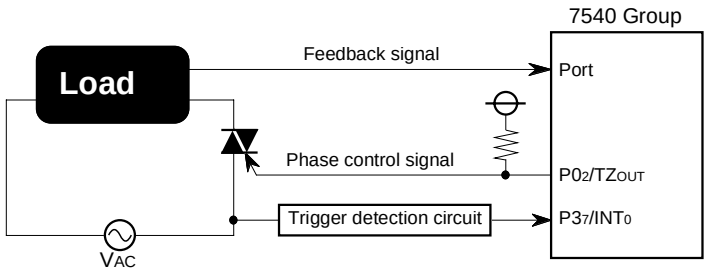


Fig. 2.5.32 Example of peripheral circuit

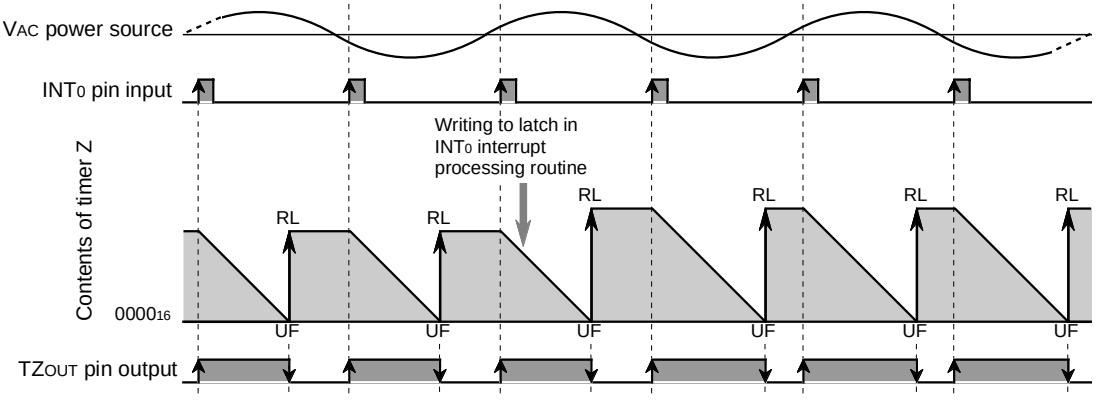
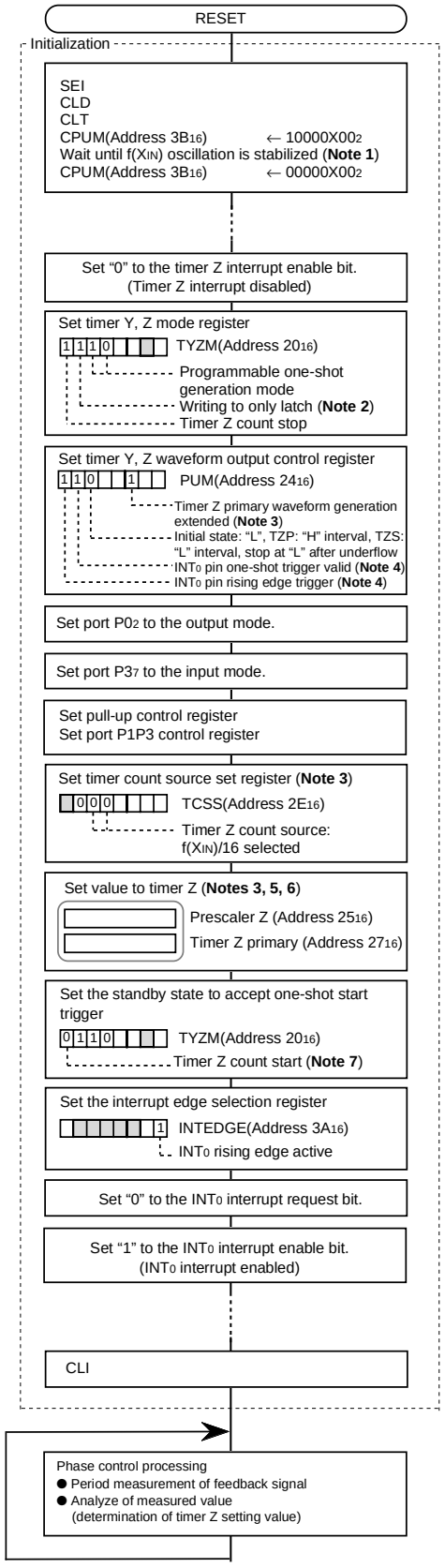
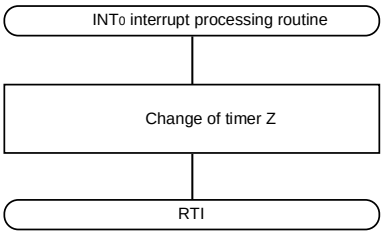


Fig. 2.5.33 Example of operation timing



X: This bit is not used here. Set it to "0" or "1" arbitrary.



Notes 1: For the concrete time, ask the oscillator manufacture.
2: When using this mode, be sure to select "write to latch only".
3: The waveform extension function by the timer Z waveform extension control bits can be used only when "0016" is set to prescaler Z.
When the value other than "0016" is set to prescaler Z, be sure to set "0" to EXPZP. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.
4: Stop timer Z to change the INT0 pin one-shot trigger control bit and INT0 one-shot trigger active edge selection bit.
5: In the programmable one-shot generation mode, the value of EXPZP is valid by writing to TZP. Even when changing TZP is not required, write the same value again.
6: In the programmable one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer Z underflow simultanesously.
7: In this state, timer count is not started.

Fig. 2.5.34 Example of control procedure

2.5.6 Programmable wait one-shot generation mode (timer Z)

(1) Operation description

In the programmable wait one-shot generation mode, the one-shot pulse by the setting value of timer Z secondary (TZS) can be output from $P0_2/TZ_{OUT}$ pin by software or external trigger to $P3_7/INT_0$ pin after the wait by the setting value of the timer Z primary (TZP). When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only". Also, set the port $P0_2$ direction registers to output mode.

The active edge of output waveform is set by the timer Z output level latch. When "0" is set to the timer Z output level latch, after the wait during the interval of the TZP setting value, "H" pulse during the interval of the TZS setting value is output. When "1" is set to the timer Z output level latch, after the wait during the interval of the TZP setting value, "L" pulse during the interval of the TZS setting value is output.

Also, in this mode, the intervals of the wait and the one-shot pulse output can be extended for 0.5 cycle of timer count source clock by setting the timer Z primary waveform extension control bit (EXPZP) and the timer Z secondary waveform extension control bit (EXPZS) to "1". As a result, the waveforms of more accurate resolution can be output.

In the programmable wait one-shot generation mode, the trigger by software or the external INT_0 pin can be accepted by writing "0" to the timer Z count stop bit after the count value is set. (At the time when "0" is written to the timer Z count stop bit, Timer Z stops.)

By writing "1" to the timer Z one-shot start bit, or by inputting the valid trigger to the INT_0 pin after the trigger to the INT_0 pin becomes valid by writing "1" to the INT_0 pin one-shot trigger control bit, Timer Z starts counting.

While Timer Z counts the TZP, the initial value of the TZ_{OUT} pin output is retained. When Timer Z underflows, the value of TZS is reloaded, at the same time, the output of TZ_{OUT} pin is inverted.

When Timer Z underflows, the output of TZ_{OUT} pin is inverted again and Timer Z stops. When also the trigger of INT_0 pin is accepted, the contents of the one-shot start bit is changed to "1" by hardware.

The falling or rising can be selected as the edge of the valid trigger of INT_0 pin by the INT_0 pin one-shot trigger edge selection bit.

During the wait interval and the one-shot pulse output interval, the one-shot pulse output can be stopped forcibly by writing "0" to the timer Z one-shot start bit.

In the programmable wait one-shot generation mode, when the count values are changed, set values to the TZS, EXPZP and EXPZS first. After then, set the value to TZP. The values are set all at once at the beginning of the next wait interval when the value is set to TZP. (When writing at timer stop is executed, writing to TZP at last is required.)

Timer Z can stop counting by setting "1" to the timer Z count stop bit.

Also, when timer Z underflows, the timer Z interrupt request bit is set to "1".

Timer Z reloads the value of latch when counting is stopped by the timer Z count stop bit.

(When timer is read out while timer is stopped, the value of latch is read. The value of timer can be read out only while timer is operating.)

Notes 1: In the programmable wait one-shot generation mode, values of TZS, EXPZP and EXPZS are valid by writing to TZP. Even when changing TZP is not required, write the same value again.

2: In the programmable wait one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow during the secondary interval simultaneously.
An example of a measurement is shown below.
ex.) The underflow by the primary and the underflow by secondary are stored by polling etc. using timer Z interrupt.
Writing to primary is performed in by judging that there is no problem if the underflow by secondary is completed with reference to primary write operation before.
(Depending on a primary setting value, primary write timing, software and timing of external trigger to INT_0 pin, it may be impossible.)

3: The waveform extension function by the timer Z waveform extension control bits can be used only when "00_{16}" is set to Prescaler Z.
When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.

4: When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only".

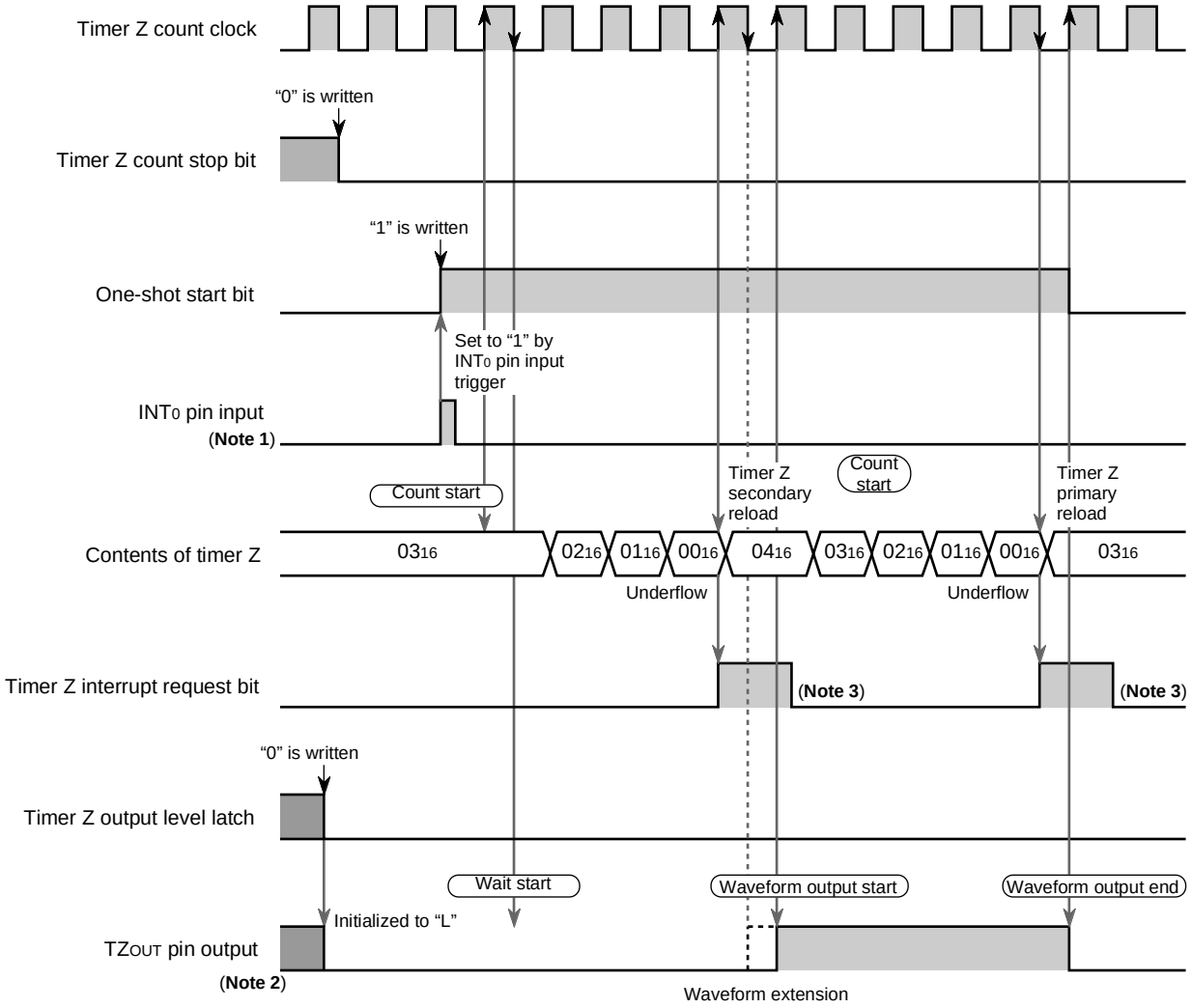
5: When TZS is read out, the undefined value is read out. However, while Timer Z counts the setting value of TZS (during one-shot output), the count value during the secondary interval can be obtained by reading TZP.

6: In order to use TZ_{OUT} pin, set "1" to bit 2 of the port P0 direction register (output mode).

7: Stop Timer Z to change the INT_0 pin one-shot trigger control bit and INT_0 pin one-shot trigger active edge selection bit.

Figure 2.5.35 shows the timing diagram of the programmable wait one-shot generation mode.

● When "03_{16}" is set to TZP and "04_{16}" is set to TZS.



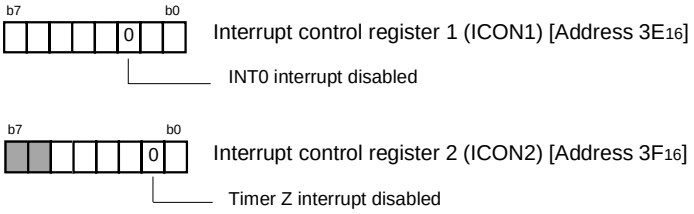
Notes 1: In this case, INT_0 pin one-shot trigger valid (rising edge trigger selected).
2: In this case, timer Z primary waveform is extended, timer Z secondary waveform is not extended.
3: In this time, "0" is written to the timer Z interrupt request bit or the timer Z interrupt request bit is cleared to "0" by accepting the timer Z interrupt request.

Fig. 2.5.35 Timing diagram of programmable wait one-shot generation mode

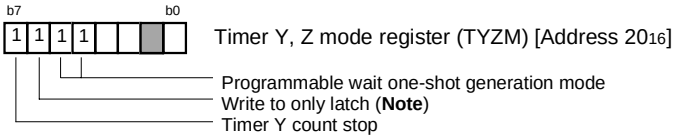
(2) Programmable wait one-shot generation mode setting method

Figure 2.5.36 to Figure 2.5.38 show the setting method for programmable wait one-shot generation mode of Timer Z.

Process 1: Disable interrupt.

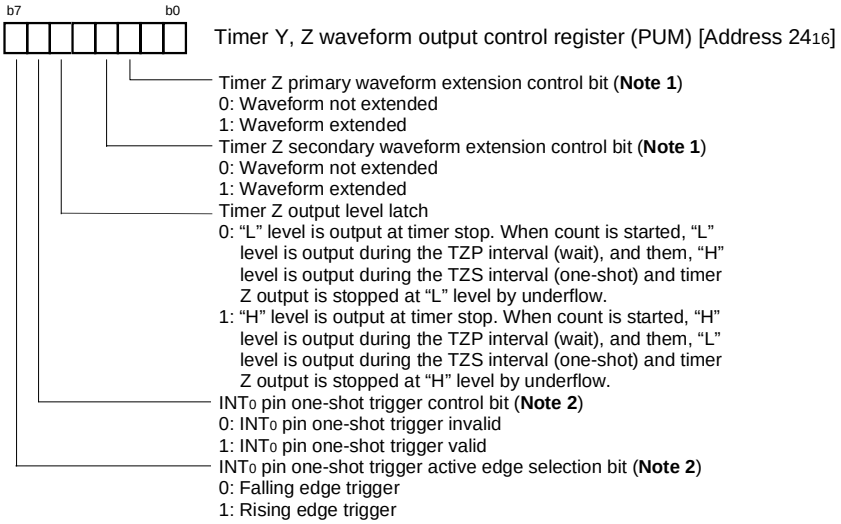


Process 2: Set timer Y, Z mode register.



Note: When using this mode, be sure to select "write to latch only".

Process 3: Set timer Y, Z waveform output control register.



Notes 1: The waveform extension function by the timer Z waveform extension control bits can be used only when "0016" is set to prescaler Z.
When the value other than "0016" is set to prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.
2: Stop Timer Z to change these bits.

Process 4: Set TZ_{OUT} pin to the output.

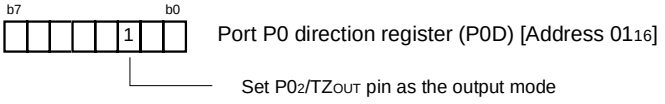
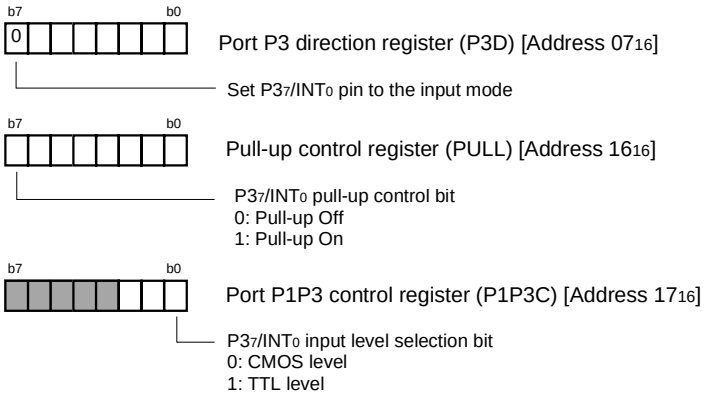
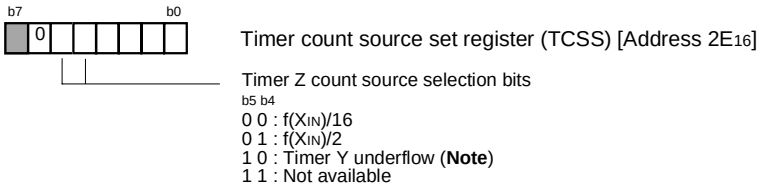


Fig. 2.5.36 Setting method for programmable wait one-shot generation mode (1)

Process 5: When the trigger by INT_0 pin input is selected:
Set port P3 direction register, pull-up control register and port P1P3 control register



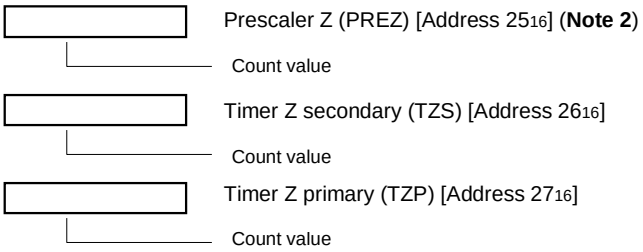
Process 6: Set the timer Z count source.



Note: When the timer Z waveform extension function is used,
do not select the timer Y underflow for the timer Z count source.

Process 7: Set the wait interval, one-shot pulse width (**Note 1**).

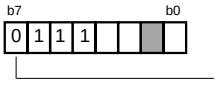
• Set the wait interval to the timer Z primary, and one-shot pulse width to the timer Z secondary.



Notes 1: In the programmable wait one-shot generation mode, values of TZS, EXPZP, and EXPZS are valid by writing to TZP. Even when changing TZP is not required, write the same value again.
2: When the timer Z waveform extension function is used, be sure to set "00₁₆" to prescaler Z.
3: In the programmable wait one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer Z underflow during the secondary interval simultanesously.
4: Count values of the primary interval (during wait) and secondary interval (during one-shot output) can be checked by reading TZP (TZS is undefined at read).

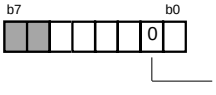
Fig. 2.5.37 Setting method for programmable wait one-shot generation mode (2)

Process 8: Set the standby state to accept the one-shot start trigger (**Note**).

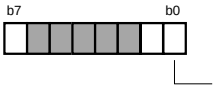


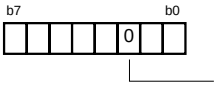
Note: When the INT0 pin one-shot trigger control bit of PUM is set to "valid", timer Z counting is started by the input of trigger to INT0 pin after this setting.

Process 9: In order not to execute the no requested interrupt processing, set "0" (no requested) to the timer Z interrupt request bit.

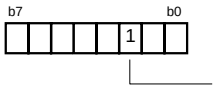


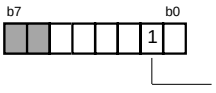
● When the INT0 pin one-shot trigger control bit is set to "valid" and the INT0 interrupt is used, set the following;





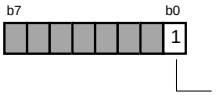
Process 10: When the interrupt is used, set "1" (interrupt enabled) to the corresponding interrupt enable bit.





Note: When the INT0 pin one-shot trigger control bit is set to "valid", the INT0 interrupt can be accepted after this setting.

Process 11: Start counting of timer Z.



● When the INT0 pin one-shot trigger control bit of PUM is set to "valid", the timer Z count is started by input of trigger to the INT0 pin.

Note: Pulse is output from TZOUT pin. After output, this bit is initialized to "0".

Fig. 2.5.38 Setting method for programmable wait one-shot generation mode (3)

(3) Application example of programmable wait one-shot generation mode

Outline: The wait one-shot pulse synchronized with the PWM waveform output from the $P0_1/TY_{OUT}$ pin is generated from Timer Z by using the programmable waveform generation mode of Timer Y.

Specifications: TY_{OUT} pin is connected to the $P3_7/INT_0$ pin. The wait one-shot pulse is output by the INT_0 pin input as trigger.
Operation clock: $f(X_{IN})$ = 8 MHz, high-speed mode

As for the usage of Timer Y, refer to the above mentioned programmable waveform generation mode. Figure 2.5.39 shows an example of waveform generation and peripheral circuit. Figure 2.5.40 shows an example of control procedure.

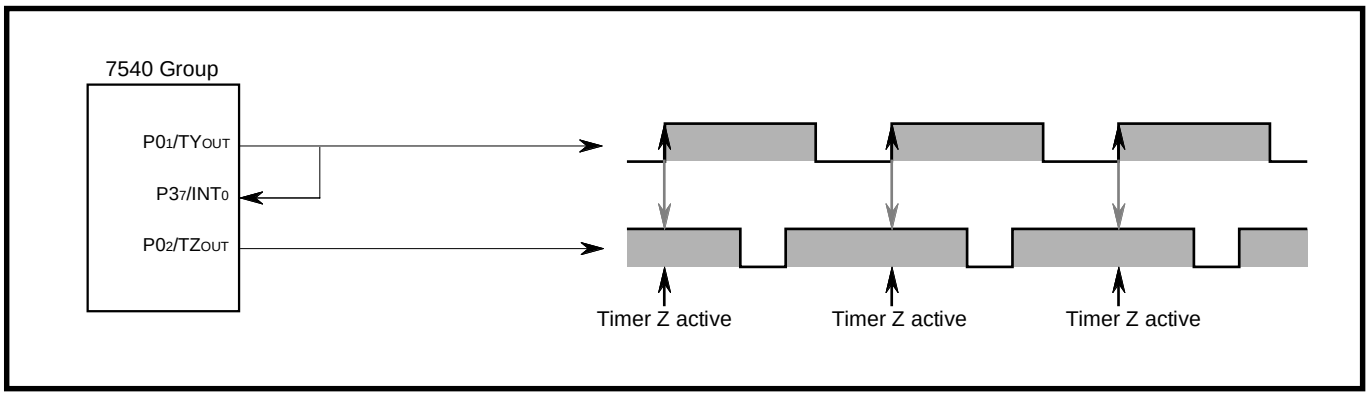
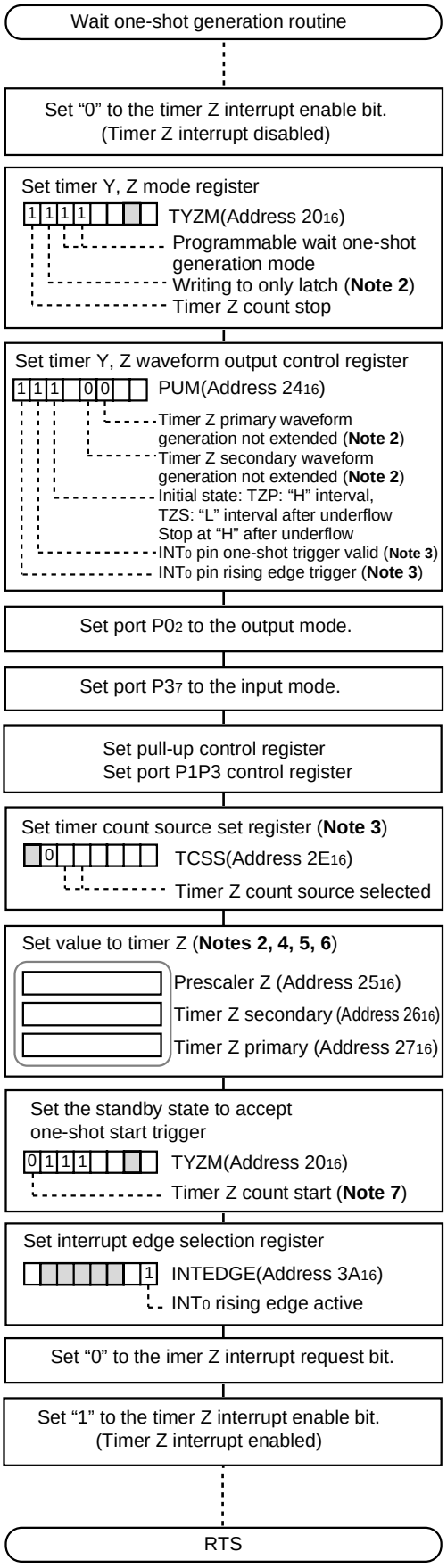


Fig. 2.5.39 Example of waveform generation and peripheral circuit



Notes 1: When using this mode, be sure to select "write to latch only".

2: The waveform extension function by the timer Z waveform extension control bits can be used only when "00₁₆" is set to prescaler Z.
When the value other than "00₁₆" is set to prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.

3: Stop timer Z to change the INT0 pin one-shot trigger control bit and INT0 one-shot trigger active edge selection bit.

4: In the programmable wait one-shot generation mode, values of TZS, EXPZP, and EXPZS are valid by writing to TZP. Even when changing TZP is not required, write the same value again.

5: In the programmable wait one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer Z underflow simultanesously.

6: Count values of the primary interval (during wait) and secondary interval (during one-shot output) can be checked by reading TZP (TZS is undefined at read).

7: In this state, timer count is not started.

Fig. 2.5.40 Example of control procedure

2.5.7 Notes on timer Y and timer Z

Notes on using each mode of Timer Y and Timer Z are described below.

(1) Timer mode (timer Y and timer Z)

① In the timer mode, TYP and TYS is not used.

(2) Programmable waveform generation mode (timer Y and timer Z)

① In the programmable waveform generation mode, values of TYS, EXPYP, and EXPYS are valid by writing to TYP because the setting to them is executed all at once by writing to TYP. Even when changing TYP is not required, write the same value again.

② In the programmable waveform generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TYP and the timing of timer underflow during the secondary interval simultaneously.
An example of a measurement is shown below.
ex.) The underflow by the primary and the underflow by secondary are stored by polling etc. using timer Y interrupt.
Writing to primary is performed in by judging that there is no problem if the underflow by secondary is completed with reference to primary write operation before.
(Depending on a primary and a secondary setting values, and primary write timing, it may be impossible.)

③ The waveform extension function by the timer Y waveform extension control bits can be used only when "00_{16}" is set to Prescaler Y.
When the value other than "00_{16}" is set to Prescaler Y, be sure to set "0" to EXPYP and EXPYS. The waveform extension function by the timer Z waveform extension control bits can be used only when "00_{16}" is set to Prescaler Z. When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.

④ When using this mode, be sure to set "1" to the timer Y write control bit to select "write to latch only".

⑤ When TYS is read out, the undefined value is read out. However, while timer Y counts the setting value of TYS, the count value during the secondary interval can be obtained by reading the timer Y primary.

⑥ In order to use TY_{OUT} pin, set "1" to bit 1 of the port P0 direction register (output mode).

(3) Programmable one-shot generation mode (timer Z)

① In the programmable one-shot generation mode, the value of EXPZP becomes valid by writing to TZP. Even when changing TZP is not required, write the same value again.

② In the programmable one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow simultaneously.

③ The waveform extension function by the timer Z waveform extension control bits can be used only when "00_{16}" is set to Prescaler Z.
When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.
An example of a measurement is shown below.
ex.) The underflow of timer is stored by polling etc. using timer Z interrupt.
Writing to primary is performed in by judging that there is no problem if the underflow by secondary is completed with reference to primary write operation before.
(Depending on a primary setting value, primary write timing, software and timing of external trigger to INT_0 pin, it may be impossible.)

④ When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only".

⑤ In order to use TZ_{OUT} pin, set "1" to bit 2 of the port P0 direction register (output mode).

⑥ Stop Timer Z to change the INT_0 pin one-shot trigger control bit and INT_0 pin one-shot trigger active edge selection bit.

(4) Programmable wait one-shot generation mode (timer Z)

① In the programmable wait one-shot generation mode, values of TZS, EXPZP and EXPZS are valid by writing to TZP. Even when changing TZP is not required, write the same value again.
An example of a measurement is shown below.
ex.) The underflow by the primary and the underflow by secondary are stored by polling etc. using timer Z interrupt.
Writing to primary is performed in by judging that there is no problem if the underflow by secondary is completed with reference to primary write operation before.
(Depending on a primary setting value, primary write timing, software and timing of external trigger to INT_0 pin, it may be impossible.)

② In the programmable wait one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow during the secondary interval simultaneously.

③ The waveform extension function by the timer Z waveform extension control bit can be used only when "00_{16}" is set to Prescaler Z.
When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.

④ When using this mode, be sure to set "1" to the timer Z write control bits to select "write to latch only".

⑤ When TZS is read out, the undefined value is read out. However, while Timer Z counts the setting value of TZS (during one-shot output), the count value during the secondary interval can be obtained by reading TZP.

⑥ In order to use TZ_{OUT} pin, set "1" to bit 2 of the port P0 direction register (output mode).

⑦ Stop Timer Z to change the INT_0 pin one-shot trigger control bit and INT_0 pin one-shot trigger active edge selection bit.

2.6 Serial I/O1

This paragraph explains the registers setting method and the notes relevant to the serial I/O.

2.6.1 Memory map

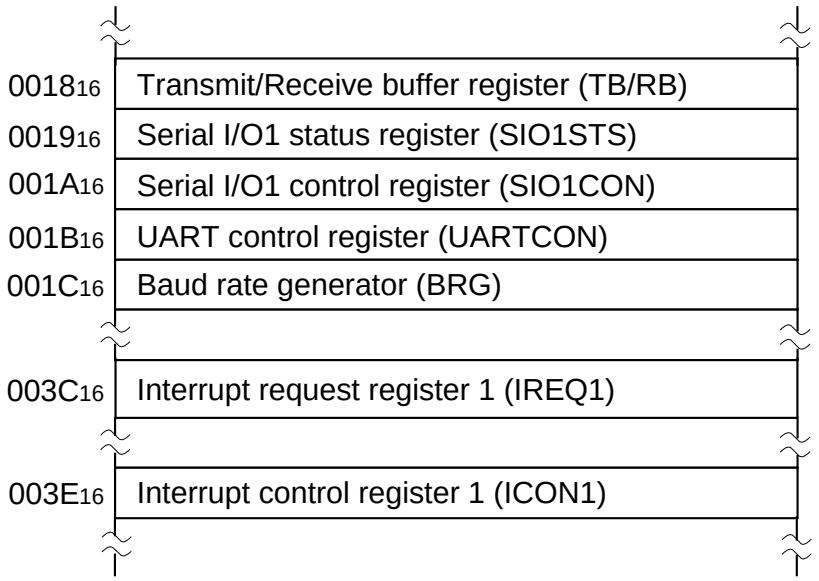
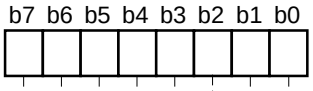


Fig. 2.6.1 Memory map of registers relevant to serial I/O

2.6.2 Relevant registers

Transmit/Receive buffer register



Transmit/Receive buffer register (TB/RB) [Address : 18 16]

B	Function	At reset	R	W
0	The transmission data is written to or the receive data is read out from this buffer register. • At writing: A data is written to the transmit buffer register. • At reading: The contents of the receive buffer register are read out.	?	○	○
1		?	○	○
2		?	○	○
3		?	○	○
4		?	○	○
5		?	○	○
6		?	○	○
7		?	○	○

Note: The contents of transmit buffer register cannot be read out.
The data cannot be written to the receive buffer register.

Fig. 2.6.2 Structure of Transmit/Receive buffer register

Serial I/O1 status register

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O1 status register (SIO1STS) [Address : 19 16]

B	Name	Function	At reset	R	W
0	Transmit buffer empty flag (TBE)	0 : Buffer full 1 : Buffer empty	0	○	×
1	Receive buffer full flag (RBF)	0 : Buffer empty 1 : Buffer full	0	○	×
2	Transmit shift register shift completion flag (TSC)	0 : Transmit shift in progress 1 : Transmit shift completed	0	○	×
3	Overrun error flag (OE)	0 : No error 1 : Overrun error	0	○	×
4	Parity error flag (PE)	0 : No error 1 : Parity error	0	○	×
5	Framing error flag (FE)	0 : No error 1 : Framing error	0	○	×
6	Summing error flag (SE)	0 : (OE) ∪ (PE) ∪ (FE) = 0 1 : (OE) ∪ (PE) ∪ (FE) = 1	0	○	×
7	Nothing is allocated for this bit. This is a write disabled bit. When this bit is read out, the value is "1".		1	○	×

Fig. 2.6.3 Structure of Serial I/O1 status register

Serial I/O1 control register

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O1 control register (SIO1CON) [Address : 1A 16]

B	Name	Function	At reset	R	W
0	BRG count source selection bit (CSS)	0 : f(XIN) 1 : f(XIN)/4	0	○	○
1	Serial I/O1 synchronous clock selection bit (SCS)	When clock synchronous serial I/O is selected; 0: BRG output divided by 4 1: External clock input When UART is selected; 0: BRG output divided by 16 1: External clock input divided by 16	0	○	○
2	$\overline{SRDY1}$ output enable bit (SRDY)	0: P13 pin 1: $\overline{SRDY1}$ output pin	0	○	○
3	Transmit interrupt source selection bit (TIC)	0 : Interrupt when transmit buffer has emptied 1 : Interrupt when transmit shift operation is completed	0	○	○
4	Transmit enable bit (TE)	0 : Transmit disabled 1 : Transmit enabled	0	○	○
5	Receive enable bit (RE)	0 : Receive disabled 1 : Receive enabled	0	○	○
6	Serial I/O1 mode selection bit (SIOM)	0: Clock asynchronous (UART) serial I/O 1: Clock synchronous serial I/O	0	○	○
7	Serial I/O1 enable bit (SIOE)	0: Serial I/O1 disabled 1: Serial I/O1 enabled	0	○	○

Fig. 2.6.4 Structure of Serial I/O1 control register

UART control register

b7 b6 b5 b4 b3 b2 b1 b0

UART control register (UARTCON) [Address : $1B_{16}$]

B	Name	Function	At reset	R	W
0	Character length selection bit (CHAS)	0 : 8 bits 1 : 7 bits	0	○	○
1	Parity enable bit (PARE)	0 : Parity checking disabled 1 : Parity checking enabled	0	○	○
2	Parity selection bit (PARS)	0 : Even parity 1 : Odd parity	0	○	○
3	Stop bit length selection bit (STPS)	0 : 1 stop bit 1 : 2 stop bits	0	○	○
4	$P11$/TxD P-channel output disable bit (POFF)	In output mode 0 : CMOS output 1 : N-channel open-drain output	0	○	○
5	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "1".		1	○	×
6			1	○	×
7			1	○	×

Fig. 2.6.5 Structure of UART control register

Baud rate generator

b7 b6 b5 b4 b3 b2 b1 b0

Baud rate generator (BRG) [Address : $1C_{16}$]

B	Function	At reset	R	W
0	Set a count value of baud rate generator.	?	○	○
1		?	○	○
2		?	○	○
3		?	○	○
4		?	○	○
5		?	○	○
6		?	○	○
7		?	○	○

Fig. 2.6.6 Structure of Baud rate generator

Interrupt request register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt request register 1 (IREQ1) [Address : 3C 16]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Serial I/O1 transmit interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	INT_0 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	INT_1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	Key-on wake up interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	$CNTR_0$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	$CNTR_1$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻

✻: These bits can be cleared to "0" by program, but cannot be set to "1".

Fig. 2.6.7 Structure of Interrupt request register 1

Interrupt control register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 1 (ICON1) [Address : 3E 16]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Serial I/O1 transmit interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	INT_0 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	INT_1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Key-on wake up interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	$CNTR_0$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	$CNTR_1$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Fig. 2.6.8 Structure of Interrupt control register 1

2.6.3 Serial I/O1 transfer data format

Figure 2.6.9 shows the serial I/O1 transfer data format.

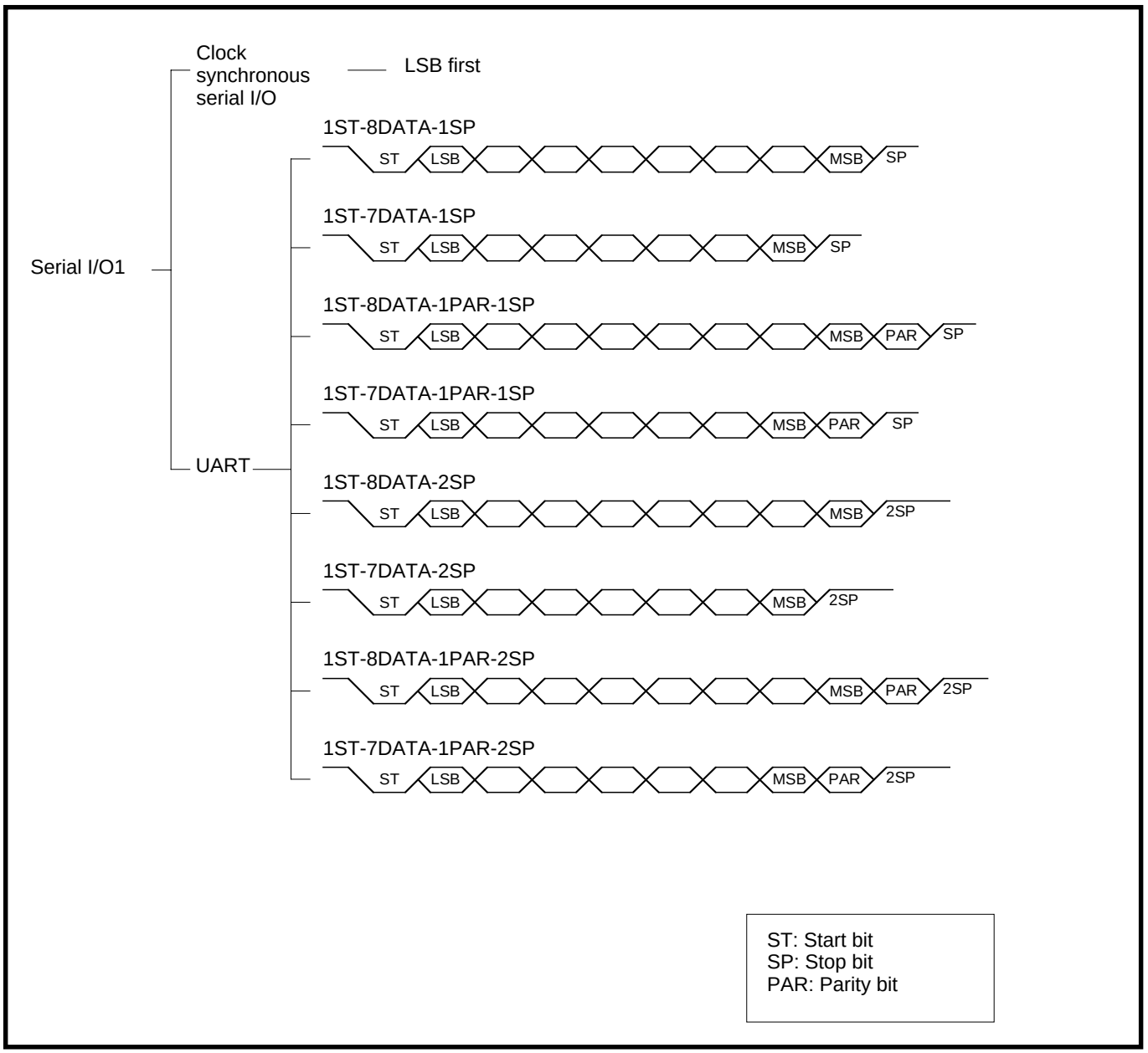


Fig. 2.6.9 Serial I/O1 transfer data format

2.6.4 Application example of clock synchronous serial I/O1

For clock synchronous serial I/O1, the transmitter and the receiver use the same clock. Synchronizing with this clock, the transmit operation of the transmitter and the receive operation of the receiver are executed at the same time. If an internal clock is used as the operation clock, transfer is started by a write signal to the TB/RB.

(1) Data transfer rate

The synchronous clock frequency is calculated by the following formula;

● When the internal clock is selected (when baud rate generator is used)

$$\text{Synchronous clock frequency [Hz]} = \frac{f(X_{IN})}{\text{Division ratio } *1 \times (\text{BRG setting value } *2 + 1) \times 4}$$

Division ratio*1 : "1" or "4" is selected (set by bit 0 of serial I/O1 control register)
BRG setting value*2 : 0 to 255 (00_{16} to FF_{16}) is set

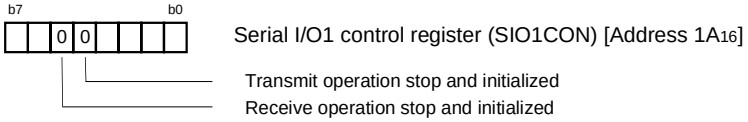
● When the external clock is selected

Synchronous clock frequency [Hz] = Clock input to S_{CLK1} pin

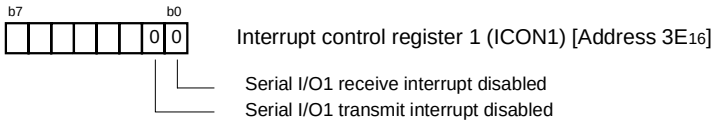
(2) Clock synchronous serial I/O setting method

Figure 2.6.10 and Figure 2.6.11 show the setting method for the clock synchronous serial I/O1.

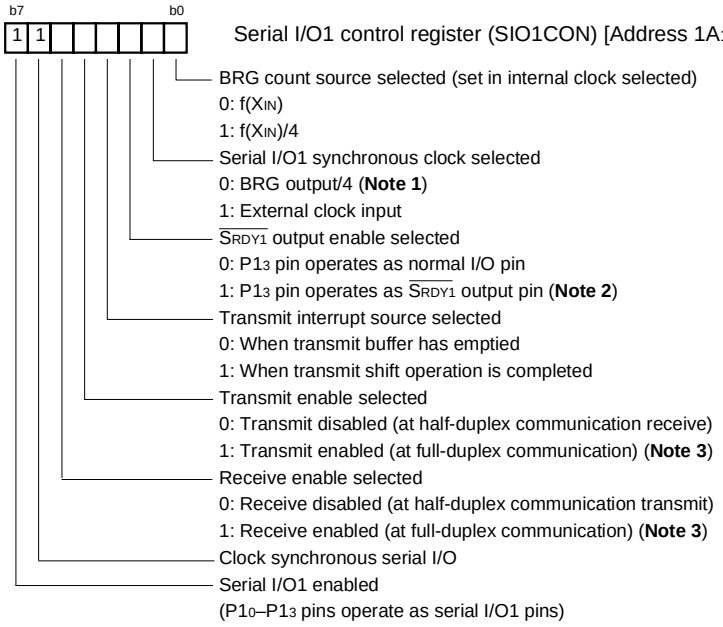
Process 1: Stop and initialize serial I/O.



Process 2: Disable serial I/O1 transmit/receive interrupt.



Process 3: Set serial I/O1 control register.



Notes **1:** Setting of serial I/O1 synchronous selection bit is as follows:
"0": P12 pin is set to be an output pin of the synchronous clock.
"1": P12 pin is set to be an input pin of the synchronous clock.

2: When an external clock input is selected as the synchronous clock, and the receiver performs the $\overline{S_{RDY1}}$ output, set "1" to the transmit enable bit in addition to the receive enable bit and $\overline{S_{RDY1}}$ output enable bit.

3: When data transmission is executed at the state that an external clock input is selected as the synchronous clock, set "1" to the transmit enable bit while the S_{CLK1} is "H" state.

Process 4: When BRG output/4 is selected as synchronous clock, set value to baud rate generator.

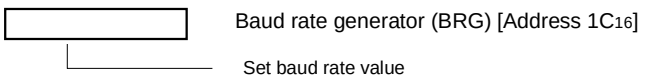
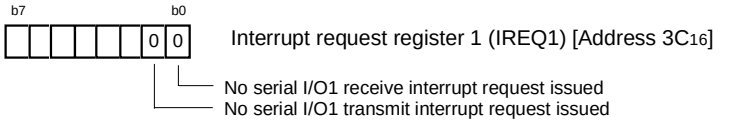
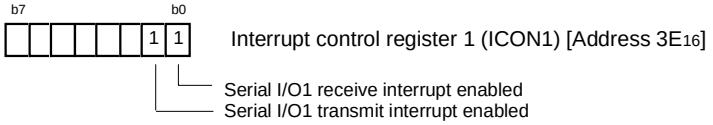


Fig. 2.6.10 Setting method for clock synchronous serial I/O1 (1)

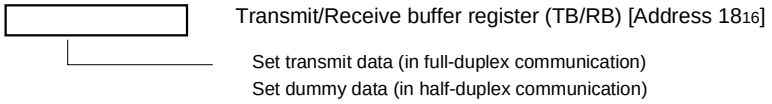
Process 5: In order not to execute the no requested interrupt processing, set “0” (no requested) to the serial I/O1 transmit/receive interrupt request bit.



Process 6: When the interrupt is used, set “1” (interrupt enabled) to the serial I/O transmit/receive interrupt enable bit.



Process 7: Transmit/Receive of serial data (**Notes 1, 2**).



Notes 1: When data transmission is executed at the state that an external clock input is selected as the synchronous clock, set the transmit data while the SCLK is “H” state.
2: When inputting the $\overline{\text{SRDY1}}$ signal, set used pins to to the input mode before transmitting data.

Fig. 2.6.11 Setting method for clock synchronous serial I/O1 (2)

(3) Communication using clock synchronous serial I/O1 (transmit/receive)

Outline : 2-byte data is transmitted and received, using the clock synchronous serial I/O1. $\overline{SRDY1}$ signal is used for communication control.

Specifications : •The serial I/O1 (clock synchronous serial I/O selected) is used.
•Synchronous clock frequency : 125 kHz; f(X_{IN}) = 4 MHz divided by 32
•The receiver outputs the $\overline{SRDY1}$ signal at 2 ms intervals which the timer generates, and 2-byte data is transferred from the transmitter to the receiver.

Figure 2.6.12 shows a connection diagram, Figure 2.6.13 shows a timing chart, Figure 2.6.14 shows the control procedure of transmitter, and Figure 2.6.15 shows an example of control procedure of receiver.

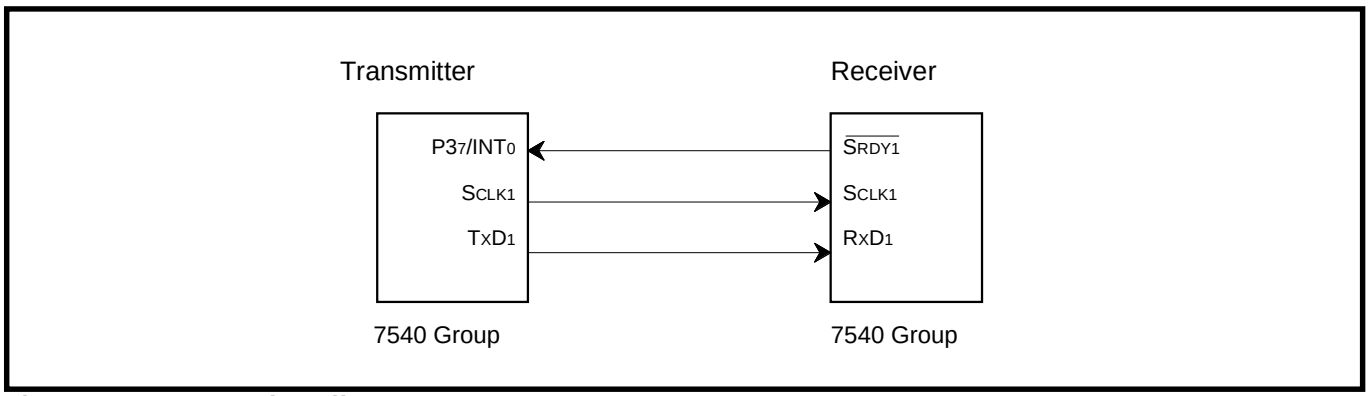


Fig. 2.6.12 Connection diagram

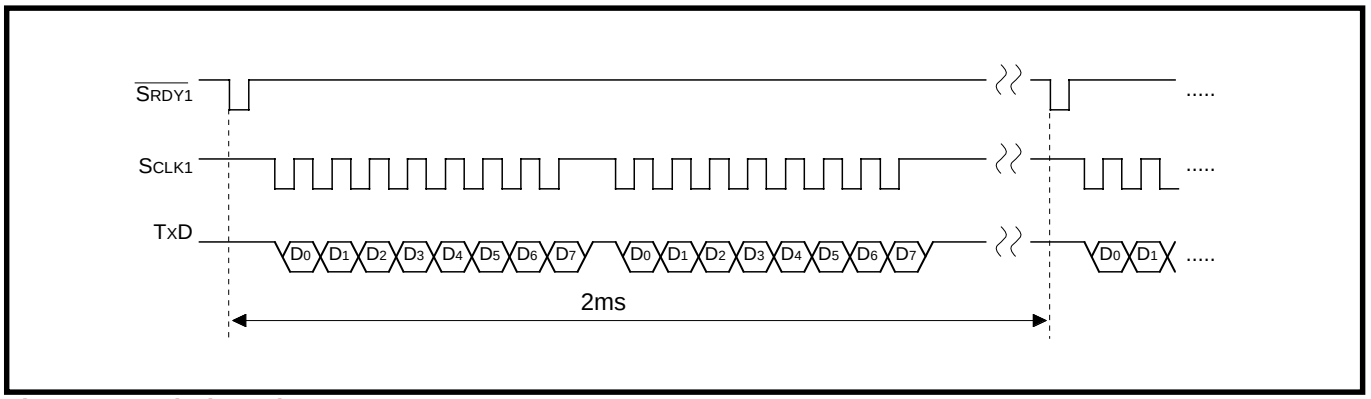


Fig. 2.6.13 Timing chart

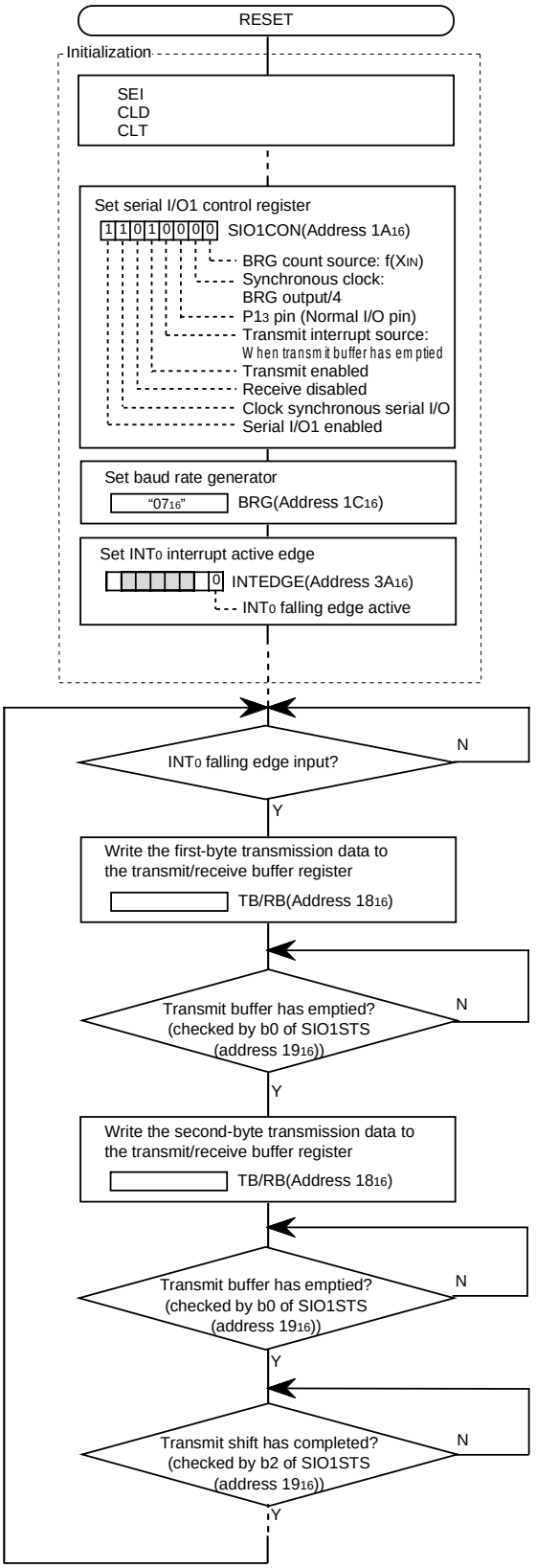


Fig. 2.6.14 Control procedure of transmitter

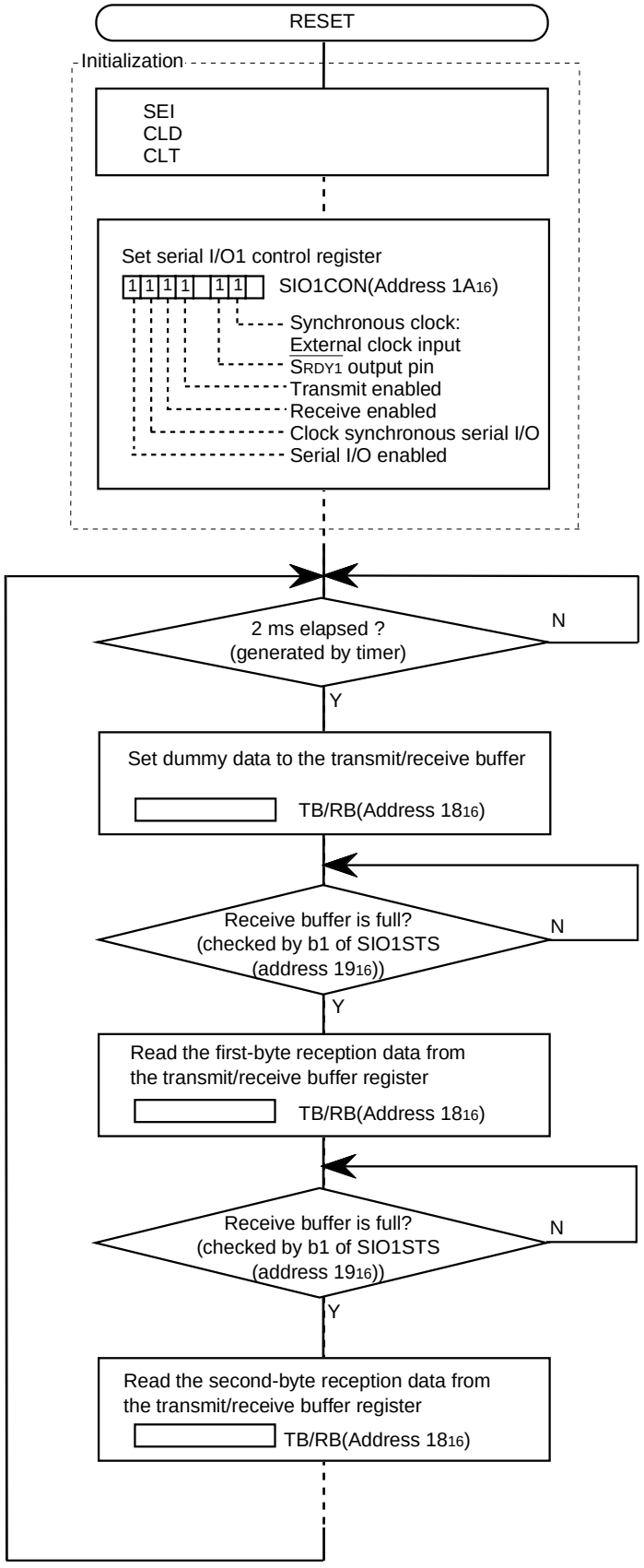


Fig. 2.6.15 Control procedure of receiver

2.6.5 Application example of clock asynchronous serial I/O1

For clock asynchronous serial I/O1 (UART), the transfer formats used by a transmitter and receiver must be identical.
In the 7540 Group, eight serial data transfer formats can be selected.

(1) Data transfer rate

The transfer bit rate is calculated by the following formula;

● When the internal clock is selected (when baud rate generator is used)

$$\text{Transfer bit rate [bps]} = \frac{f(X_{IN})}{\text{Division ratio}^{*1} \times (\text{BRG setting value}^{*2} + 1) \times 16}$$

Division ratio*1 : "1" or "4" is selected (set by bit 0 of serial I/O1 control register)
BRG setting value*2 : 0 to 255 (00_{16} to FF_{16}) is set

● When the external clock is selected

Transfer bit rate [bps] = Clock input to S_{CLK1} pin/16

Table 2.6.1 shows the setting example of baud rate generator and transfer bit rate values.

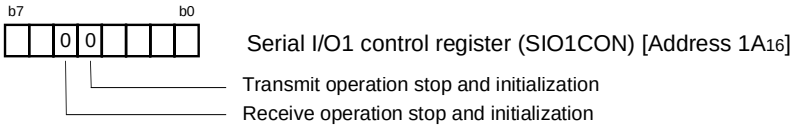
Table 2.6.1 Setting example of baud rate generator (BRG) and transfer bit rate values

BRG count source	BRG set value	Transfer bit rate (bps)	
		At $f(X_{IN})$ = 4.9152 MHz	At $f(X_{IN})$ = 8 MHz
$f(X_{IN})$ / 4	255 (FF_{16})	300	488.28125
$f(X_{IN})$ / 4	127 ($7F_{16}$)	600	976.5625
$f(X_{IN})$ / 4	63 ($3F_{16}$)	1200	1953.125
$f(X_{IN})$ / 4	31 ($1F_{16}$)	2400	3906.25
$f(X_{IN})$ / 4	15 ($0F_{16}$)	4800	7812.5
$f(X_{IN})$ / 4	7 (07_{16})	9600	15625
$f(X_{IN})$ / 4	3 (03_{16})	19200	31250
$f(X_{IN})$ / 4	1 (01_{16})	38400	62500
$f(X_{IN})$	3 (03_{16})	76800	125000
$f(X_{IN})$	1 (01_{16})	153600	250000
$f(X_{IN})$	0 (00_{16})	307200	500000

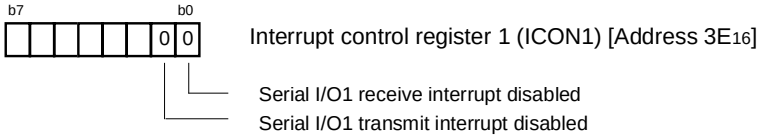
(2) UART setting method

Figure 2.6.16 and Figure 2.6.17 show the setting method for UART of serial I/O1.

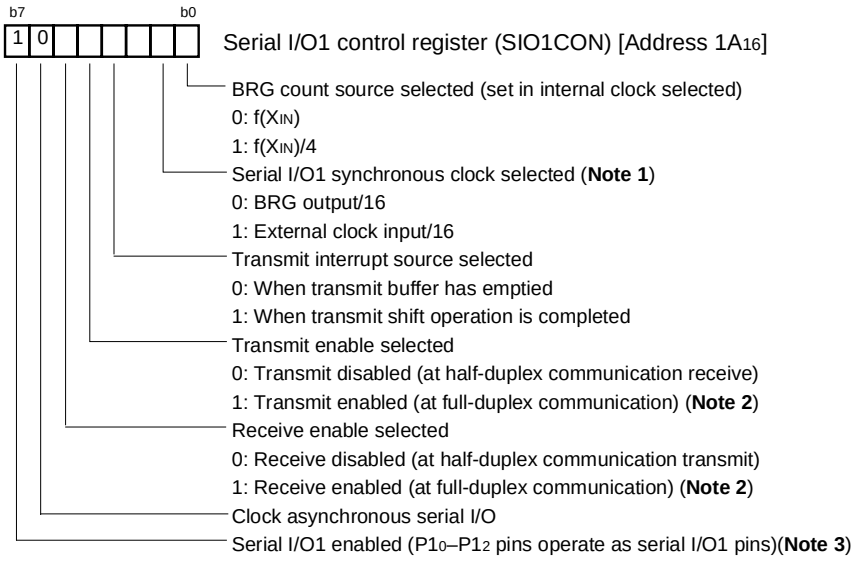
Process 1: Stop and initialize serial I/O.



Process 2: Disable serial I/O1 transmit/receive interrupt.



Process 3: Set serial I/O1 control register.



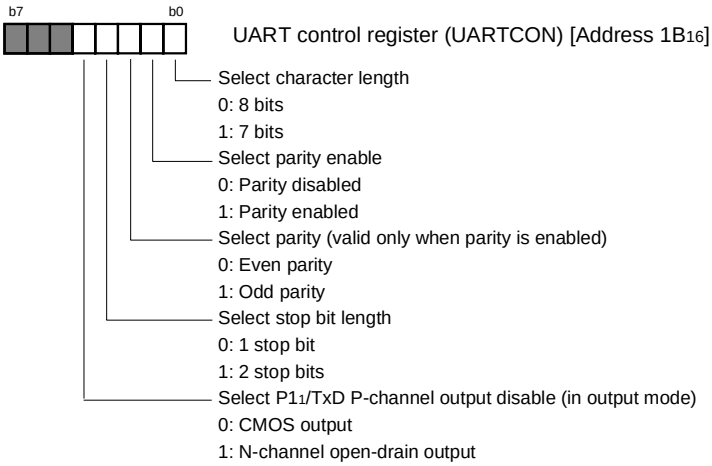
Note 1: Setting of serial I/O1 synchronous clock selection bit is as follows;
"0": P12 pin can be used as a normal I/O pin
"1": P12 pin is used as an input pin for an external clock.

2: When data transmission is executed at the state that an external clock input is selected as the synchronous clock, set "1" to the transmit enable bit while the SCLK1 is "H" state.

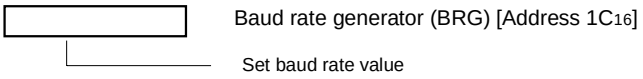
3: When clock asynchronous (UART) serial I/O is selected, P13 pin can be used as a normal I/O pin.

Fig. 2.6.16 Setting method for UART of serial I/O1 (1)

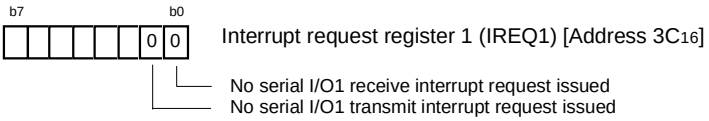
Process 4: Set UART control register.



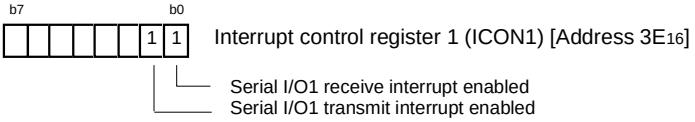
Process 5: When BRG output/16 is selected as synchronous clock, set value to baud rate generator.



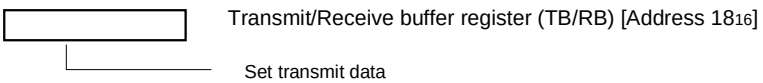
Process 6: In order not to execute the no requested interrupt processing, set “0” (no requested) to the serial I/O1 transmit/receive interrupt request bit.



Process 7: When the interrupt is used, set “1” (interrupt enabled) to the serial I/O1 transmit/receive interrupt enable bit.



Process 8: When transmitting, start serial data transmission (**Note**).



Note: When data transmission is executed at the state that an external clock input is selected as the synchronous clock, set the transmit data while the S_{CLK1} is “H” state.

Fig. 2.6.17 Setting method for UART of serial I/O1 (2)

(3) Communication using UART of serial I/O (transmit/receive)

Outline : 2-byte data is transmitted and received, using UART. Port $P0_0$ is used for communication control.

Specifications : •The Serial I/O1 (UART selected) is used.
•Transfer bit rate : 9600 bps ($f(X_{IN})$ = 4.9152 MHz divided by 512)
•Communication control using port $P0_0$ (output level of port $P0_0$ is controlled by software)
2-byte data is transferred from the transmitter to the receiver at 10 ms intervals which the timer generates.

Figure 2.6.18 shows a connection diagram, Figure 2.6.19 shows a timing chart, Figure 2.6.20 shows the control procedure of transmitter, and Figure 2.6.21 shows an example of control procedure of receiver.

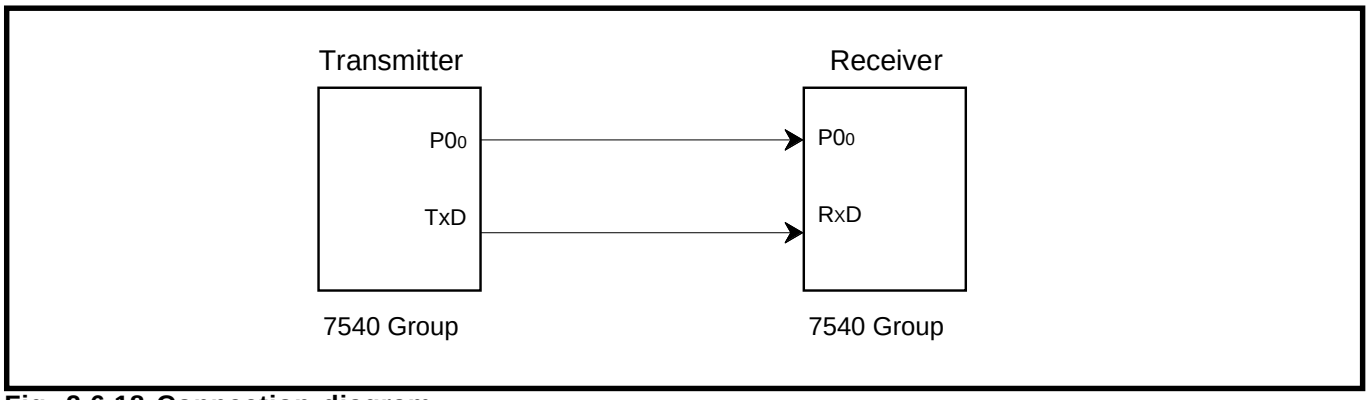


Fig. 2.6.18 Connection diagram

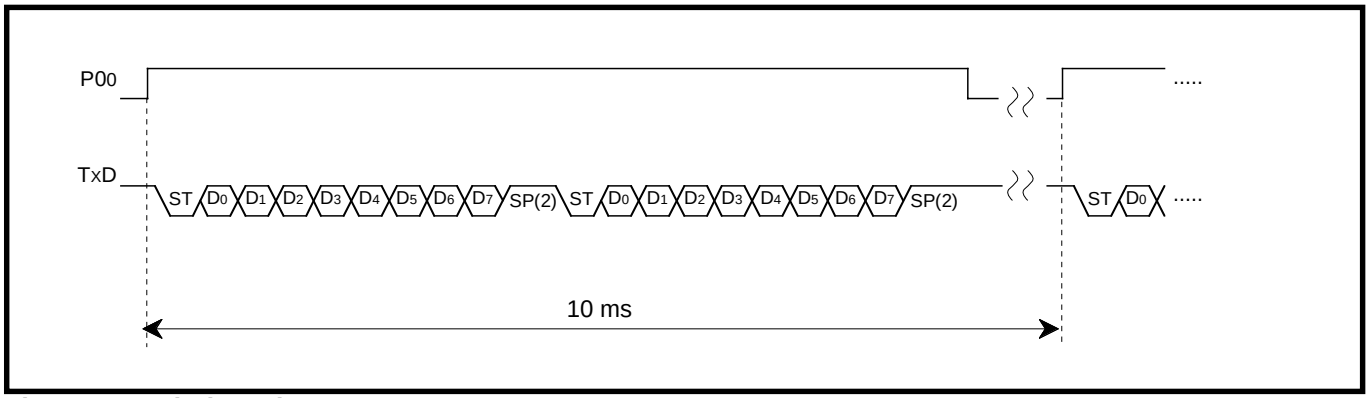


Fig. 2.6.19 Timing chart

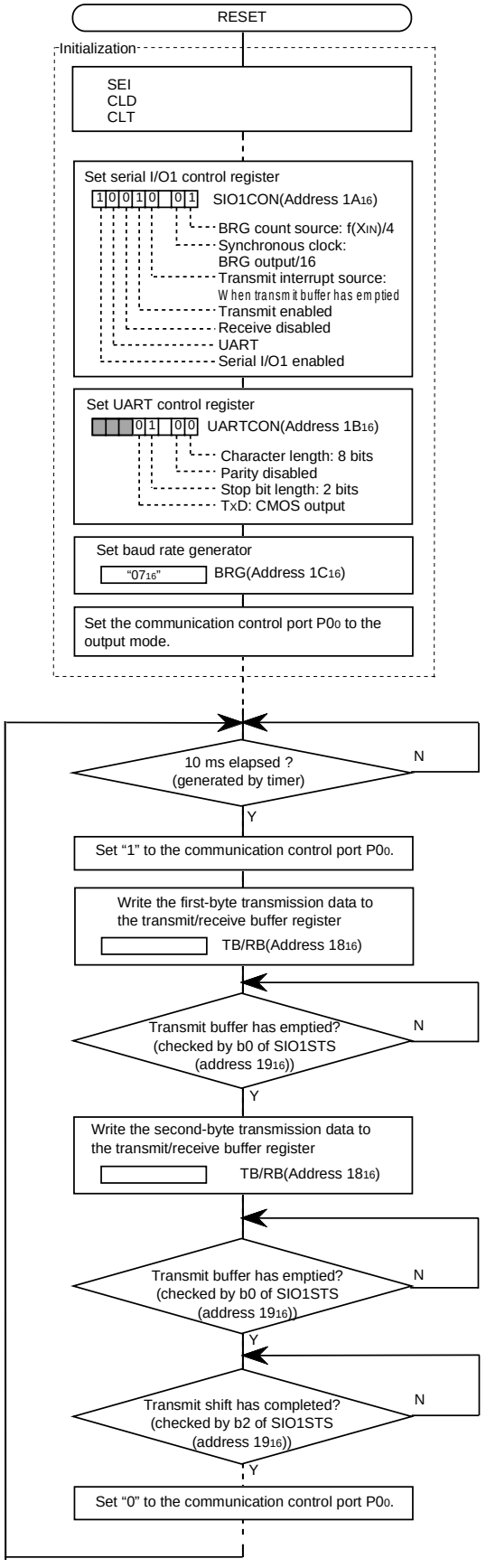


Fig. 2.6.20 Control procedure of transmitter

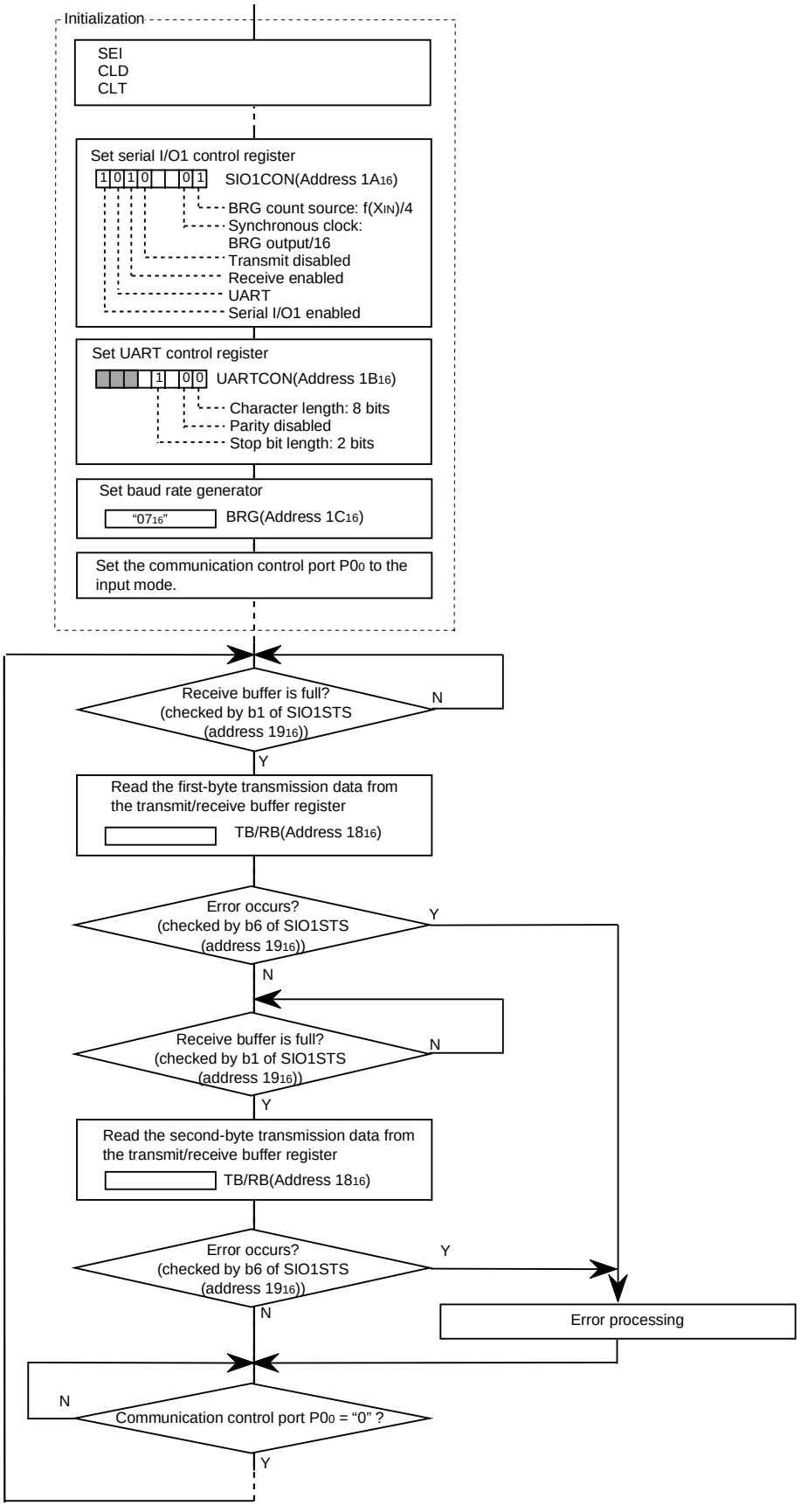


Fig. 2.6.21 Control procedure of receiver

2.6.6 Notes on Serial I/O1

Notes on using serial I/O1 are described below.

(1) Notes when selecting clock synchronous serial I/O

① When the clock synchronous serial I/O1 is used, serial I/O2 cannot be used.

② When the transmit operation is stopped, clear the serial I/O1 enable bit and the transmit enable bit to "0" (serial I/O1 and transmit disabled).

● **Reason**

Since transmission is not stopped and the transmission circuit is not initialized even if only the serial I/O1 enable bit is cleared to "0" (serial I/O1 disabled), the internal transmission is running (in this case, since pins TxD_1, RxD_1, S_{CLK1}, and $\overline{S_{RDY1}}$ function as I/O ports, the transmission data is not output). When data is written to the transmit buffer register in this state, data starts to be shifted to the transmit shift register. When the serial I/O1 enable bit is set to "1" at this time, the data during internally shifting is output to the TxD_1 pin and an operation failure occurs.

③ When the receive operation is stopped, clear the receive enable bit to "0" (receive disabled), or clear the serial I/O1 enable bit to "0" (serial I/O1 disabled).

④ When the transmit/receive operation is stopped, clear both the transmit enable bit and receive enable bit to "0" (transmit and receive disabled) simultaneously. (any one of data transmission and reception cannot be stopped.)

● **Reason**

In the clock synchronous serial I/O mode, the same clock is used for transmission and reception. If any one of transmission and reception is disabled, a bit error occurs because transmission and reception cannot be synchronized.

In this mode, the clock circuit of the transmission circuit also operates for data reception. Accordingly, the transmission circuit does not stop by clearing only the transmit enable bit to "0" (transmit disabled). Also, the transmission circuit cannot be initialized even if the serial I/O1 enable bit is cleared to "0" (serial I/O1 disabled) (same as ②).

⑤ When signals are output from the $\overline{S_{RDY1}}$ pin on the reception side by using an external clock, set all of the receive enable bit, the $\overline{S_{RDY1}}$ output enable bit, and the transmit enable bit to "1".

⑥ When the S_{RDY1} signal input is used, set the using pin to the input mode before data is written to the transmit/receive buffer register.

⑦ Setup of a serial I/O1 synchronous clock selection bit when a clock synchronous serial I/O is selected;

"0" : $P1_2$ pin turns into an output pin of a synchronous clock.

"1" : $P1_2$ pin turns into an input pin of a synchronous clock.

Setup of a $\overline{S_{RDY1}}$ output enable bit ($\overline{S_{RDY1}}$) when a clock synchronous serial I/O1 is selected;

"0" : $P1_3$ pin can be used as a normal I/O pin.

"1" : $P1_3$ pin turns into a $\overline{S_{RDY1}}$ output pin.

(2) Notes when selecting UART

① When the clock asynchronous serial I/O1 (UART) is used, serial I/O2 can be used only when BRG output divided by 16 is selected as the synchronous clock.

② When the transmit operation is stopped, clear the transmit enable bit to "0" (transmit disabled).

● **Reason**
Same as (1) ②.

③ When the receive operation is stopped, clear the receive enable bit to "0" (receive disabled).

④ When the transmit/receive operation is stopped, clear the transmit enable bit to "0" (transmit disabled) and receive enable bit to "0" (receive disabled).

⑤ Setup of a serial I/O1 synchronous clock selection bit when a clock asynchronous (UART) serial I/O is selected;
"0": $P1_2$ pin can be used as a normal I/O pin.
"1": $P1_2$ pin turns into an input pin of an external clock.
When clock asynchronous (UART) type serial I/O is selected, it is $P1_3$ pin. It can be used as a normal I/O pin.

(3) Notes common to clock synchronous serial I/O and UART

① Set the serial I/O control register again after the transmission and the reception circuits are reset by clearing both the transmit enable bit and the receive enable bit to "0."

② The transmit shift completion flag changes from "1" to "0" with a delay of 0.5 to 1.5 shift clocks. When data transmission is controlled with referring to the flag after writing the data to the transmit buffer register, note the delay.

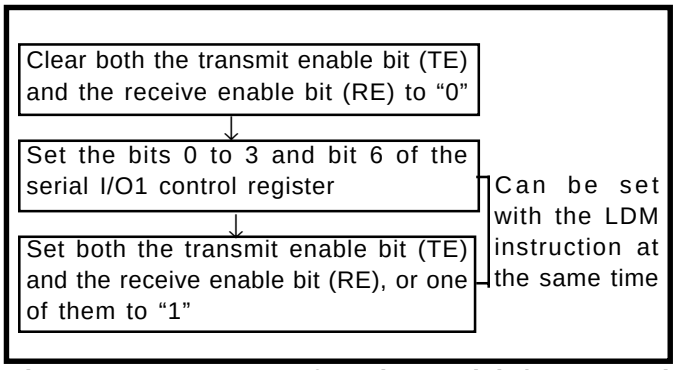


Fig. 2.6.22 Sequence of setting serial I/O1 control register again

③ When data transmission is executed at the state that an external clock input is selected as the synchronous clock, set "1" to the transmit enable bit while the S_{CLK1} is "H" state. Also, write to the transmit buffer register while the S_{CLK1} is "H" state.

④ When the transmit interrupt is used, set as the following sequence.
❶ Serial I/O1 transmit interrupt enable bit is set to "0" (disabled).
❷ Serial I/O1 transmit enable bit is set to "1".
❸ Serial I/O1 transmit interrupt request bit is set to "0".
❹ Serial I/O1 transmit interrupt enable bit is set to "1" (enabled).

● **Reason**
When the transmit enable bit is set to "1", the transmit buffer empty flag and transmit shift completion flag are set to "1".
Accordingly, even if the timing when any of the above flags is set to "1" is selected for the transmit interrupt source, interrupt request occurs and the transmit interrupt request bit is set.

⑤ Write to the baud rate generator (BRG) while the transmit/receive operation is stopped.

2.7 Serial I/O2

This paragraph explains the registers setting method and the notes relevant to the serial I/O.

2.7.1 Memory map

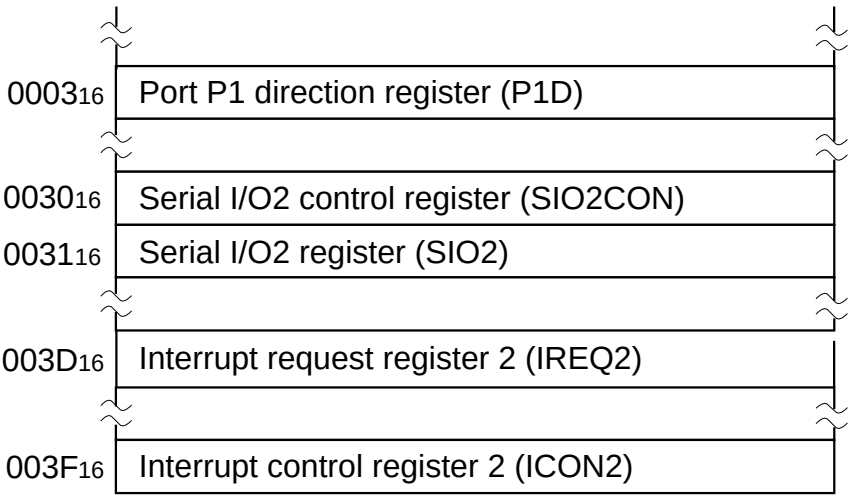


Fig. 2.7.1 Memory map of registers relevant to serial I/O2

2.7.2 Relevant registers

Port P1 direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port P1 direction register (P1D) [Address : 03 16]

B	Name	Function	At reset	R	W
0	Port P1 direction register	0 : Port P1$_0$ input mode 1 : Port P1$_0$ output mode	0	×	○
1		0 : Port P1$_1$ input mode 1 : Port P1$_1$ output mode	0	×	○
2		0 : Port P1$_2$ input mode 1 : Port P1$_2$ output mode	0	×	○
3		0 : Port P1$_3$ input mode 1 : Port P1$_3$ output mode	0	×	○
4		0 : Port P1$_4$ input mode 1 : Port P1$_4$ output mode	0	×	○
5	Nothing is allocated for these bits. When these bits are read out, the values are undefined.		?	×	×
6			?	×	×
7			?	×	×

Fig. 2.7.2 Structure of Port P1 direction register

Serial I/O2 control register

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O2 control register (SIO2CON) [Address : 30_{16}]

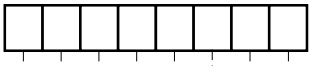
B	Name	Function	At reset	R	W
0	Internal synchronous clock selection bits	b2 b1 b0 0 0 0 : f(X_{IN})/8 0 0 1 : f(X_{IN})/16 0 1 0 : f(X_{IN})/32 0 1 1 : f(X_{IN})/64 1 1 0 : f(X_{IN})/128 1 1 1 : f(X_{IN})/256	0	○	○
1			0	○	○
2			0	○	○
3	S_{DATA2} pin selection bit (**Note**)	0 : I/O port / S_{DATA2} input 1 : S_{DATA2} output	0	○	○
4	Nothing is allocated for this bit. This is a write disabled bit. When this bit is read out, the value is "0".		0	○	×
5	Transfer direction selection bit	0 : LSB first 1 : MSB first	0	○	○
6	S_{CLK2} pin selection bit	0 : External clock (S_{CLK2} is input) 1 : Internal clock (S_{CLK2} is output)	0	○	○
7	Transmit / receive shift completion flag	0 : shift in progress 1 : shift completed	0	○	×

Note: When using it as a S_{DATA} input, set the port $P1_3$ direction register bit to "0".

Fig. 2.7.3 Structure of Serial I/O2 control register

Serial I/O2 register

b7 b6 b5 b4 b3 b2 b1 b0



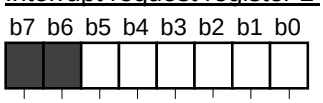
Serial I/O2 register (SIO2) [Address : 31_{16}]

B	Function	At reset	R	W
0	A shift register for serial transmission and reception. • At transmitting : Set a transmission data. • At receiving : A reception data is stored.	?	○	○
1		?	○	○
2		?	○	○
3		?	○	○
4		?	○	○
5		?	○	○
6		?	○	○
7		?	○	○

Fig. 2.7.4 Structure of Serial I/O2 register

2.7 Serial I/O2

Interrupt request register 2



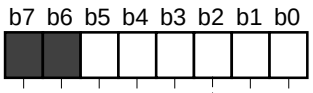
Interrupt request register 2 (IREQ2) [Address : 3D 16]

B	Name	Function	At reset	R	W
0	Timer Y interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Timer Z interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	Timer A interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	AD converter interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are “0”.		0	○	×
7			0	○	×

✻: These bits can be cleared to “0” by program, but cannot be set to “1”.

Fig. 2.7.5 Structure of Interrupt request register 2

Interrupt control register 2



Interrupt control register 2 (ICON2) [Address : 3F 16]

B	Name	Function	At reset	R	W
0	Timer Y interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer Z interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer A interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	AD conversion interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are “0”.		0	○	×
7			0	○	×

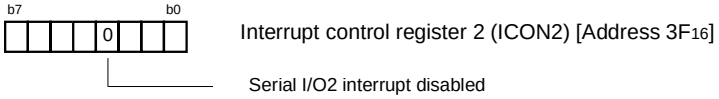
Fig. 2.7.6 Structure of Interrupt control register 2

2.7.3 Application example of serial I/O2

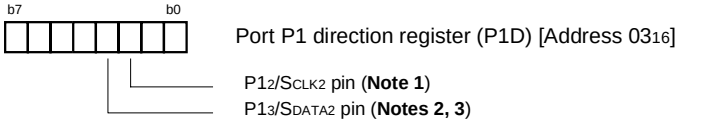
(1) Serial I/O2 setting method

Figure 2.7.7 and Figure 2.7.8 show the setting method for the serial I/O2.

Process 1: Disable serial I/O2 transmit/receive interrupt.

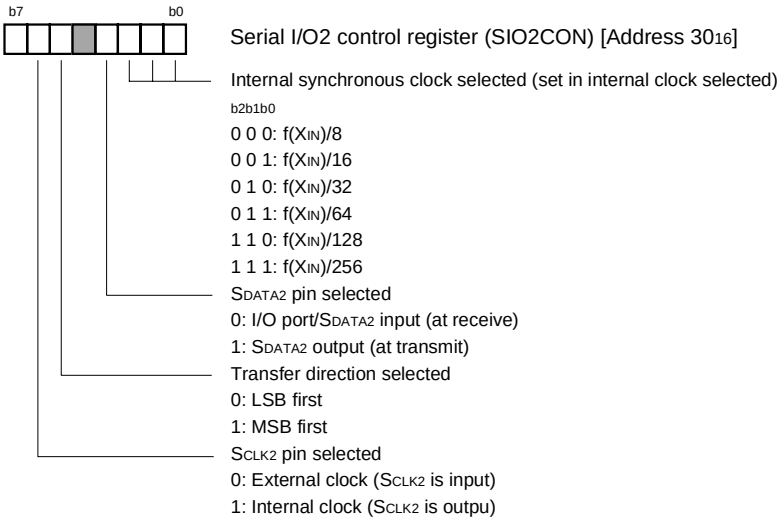


Process 2: Set port P1 according to the usage condition.



Notes 1: When an external clock input is selected, set P12/SCLK2 pin to the input mode.
2: When P13/SDATA2 pin is used as the P13 pin, set this bit to "0".
3: When this bit is set to "0" at transmit and the internal clock is selected for SCLK2, the SDATA2 pin is in a high impedance state after the data transfer is completed.

Process 3: Set serial I/O2 control register.



Process 4: In order not to execute the no requested interrupt processing, set "0" (no requested) to the serial I/O2 interrupt request bit.

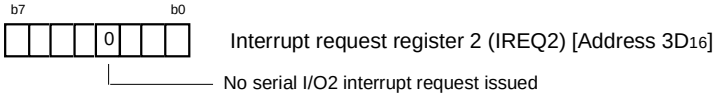
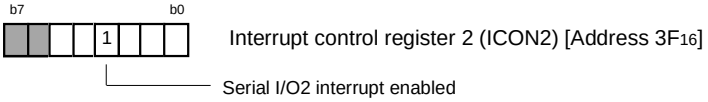


Fig. 2.7.7 Setting method for serial I/O2

2.7 Serial I/O2

Process 5: When the interrupt is used, set “1” (interrupt enabled) to the serial I/O2 interrupt. enable bit.



Process 6: When transmitting, start serial data transmission.

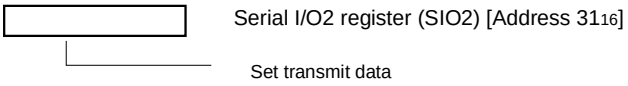


Fig. 2.7.8 Setting method for serial I/O2

(2) Communication using serial I/O2 (transmit/receive)

Outline: 2-byte data is transmitted and received, using the serial I/O2. Port $P0_0$ is used for communication control and outputs the quasi-$\overline{S_{RDY}}$ signal.

Specifications:
- The serial I/O2, clock synchronous serial I/O, is used.
- Synchronous clock frequency : 125 kHz; f(X_{IN}) = 8 MHz divided by 64
- Transfer direction : LSB first
- The receiver outputs the quasi-$\overline{S_{RDY}}$ signal at 2 ms intervals which the timer generates, and 2-byte data is transferred from the transmitter to the receiver.

Figure 2.7.9 shows a connection diagram, Figure 2.7.10 shows a timing chart, Figure 2.7.11 shows the control procedure of transmitter, and Figure 2.7.12 shows an example of control procedure of receiver.

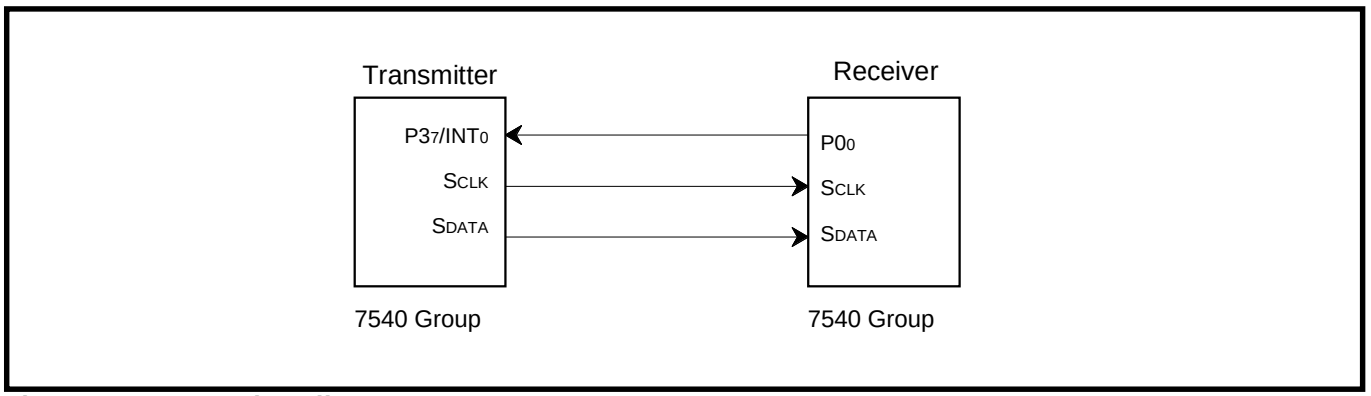


Fig. 2.7.9 Connection diagram

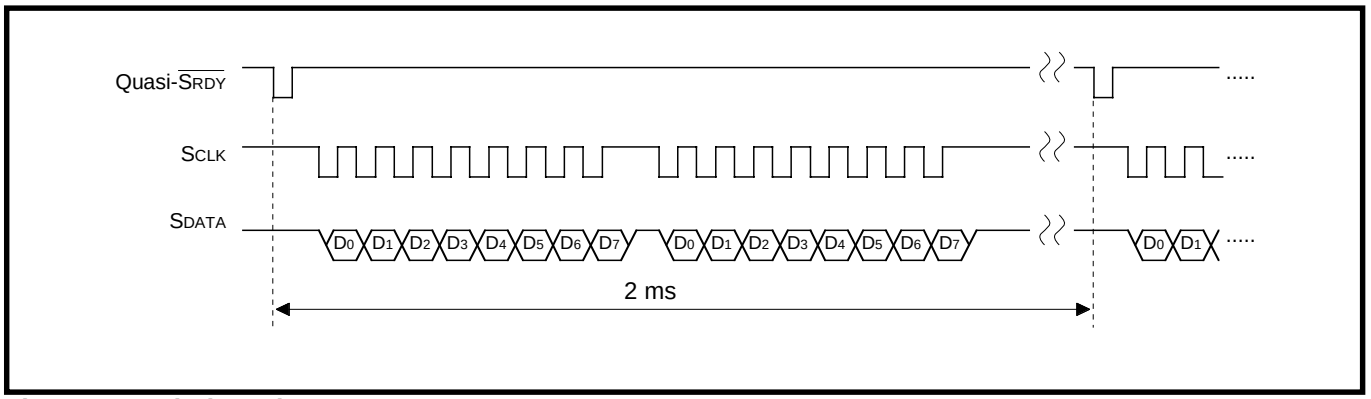
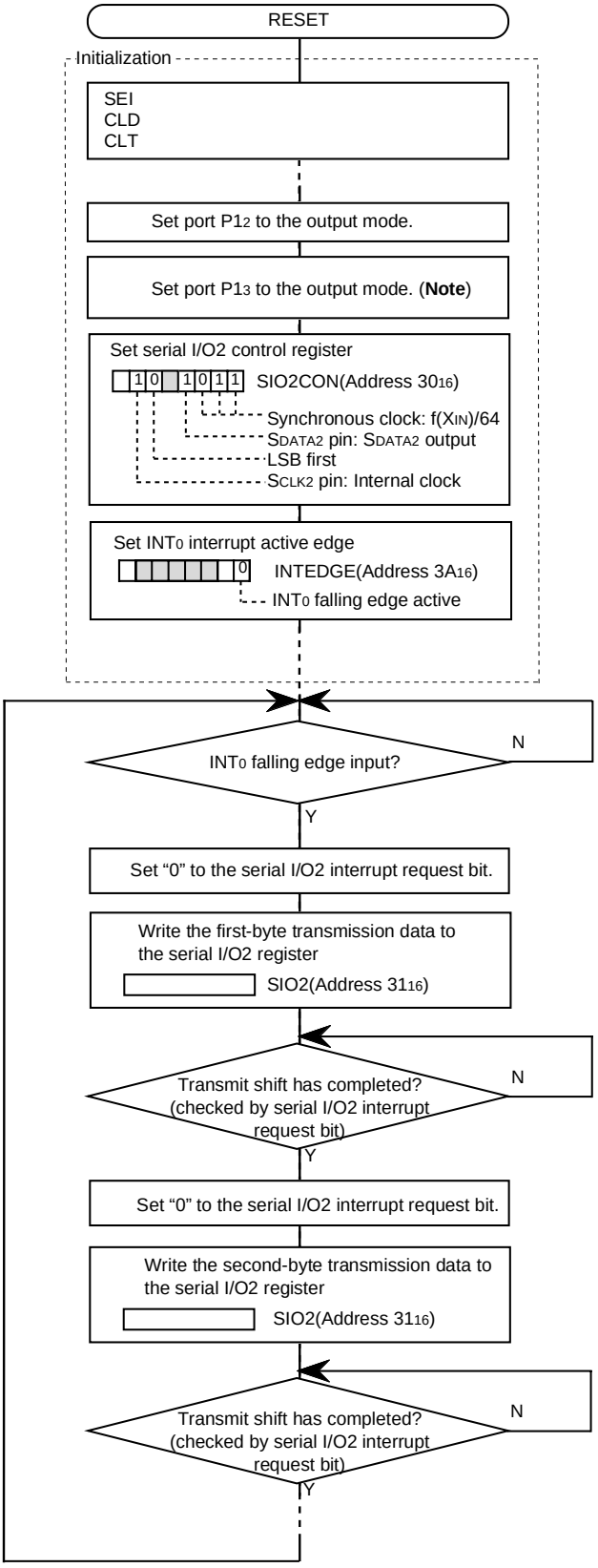
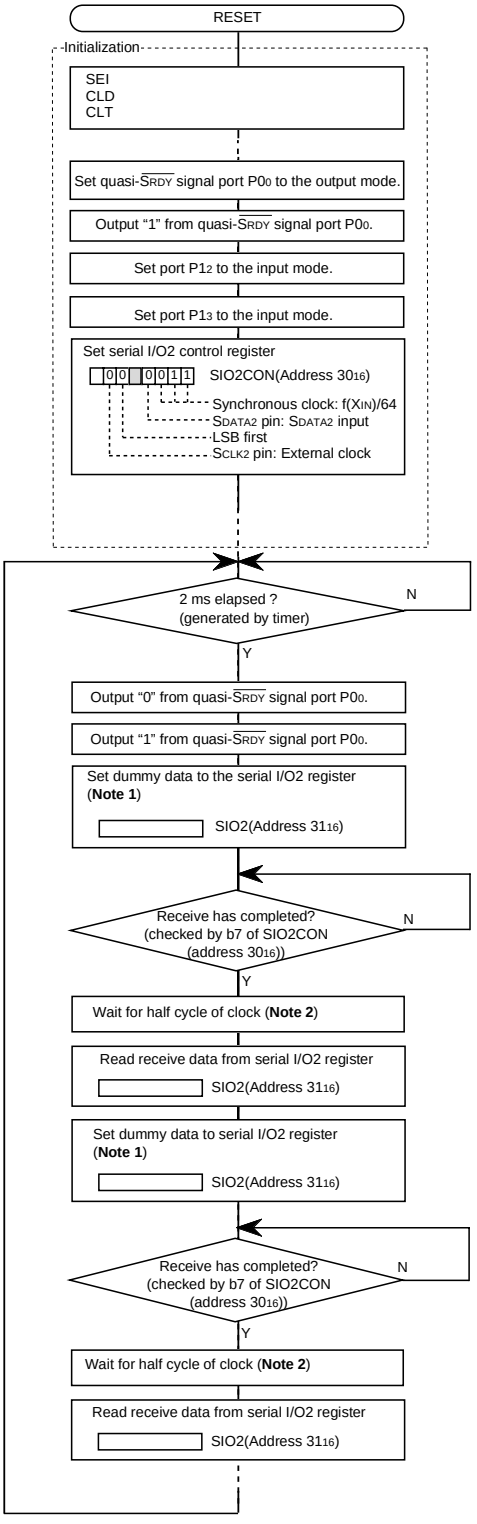


Fig. 2.7.10 Timing chart



Note: When direction register of P13/SDATA2 pin is set to the input mode and the internal clock is selected, the SDATA2 pin is in a high impedance state after the data transfer is completed.

Fig. 2.7.11 Control procedure of transmission side



Notes 1: The transmit/receive shift completion flag of the serial I/O2 control register is "1" after transmit/receive shift is completed. In order to set "0" to this flag, set data (dummy data at receive) to the serial I/O2 register by program.

2: Bit 7 (transmit/receive shift completion flag) of the serial I/O2 control register is set earlier than the completion of the actual shift operation for a half cycle of shift clock. Accordingly, when the shift completion is checked by using this bit, read/write the serial I/O2 register after a half or more cycle of clock from the setting "1" to this bit is checked.

Fig. 2.7.12 Control procedure of reception side

2.7.4 Notes on serial I/O2

Notes on using serial I/O2 are described below.

(1) Note on serial I/O1

Serial I/O2 can be used only when serial I/O1 is not used or serial I/O1 is used as UART and the BRG output divided by 16 is selected as the synchronous clock.

(2) Note on S_{CLK2} pin

When an external clock is selected, set "0" to bit 2 of the port P1 direction register (input mode).

(3) Note on S_{DATA2} pin

When $P1_3/S_{RDY1}/S_{DATA2}$ pin is used as the S_{DATA} input, set "0" to bit 3 of the port P1 direction register (input mode).

When the internal clock is selected as the transfer and $P1_3/S_{DATA2}$ pin is set to the input mode, the S_{DATA2} pin is in a high-impedance state after the data transfer is completed.

(4) Notes on serial I/O2 transmit/receive shift completion flag

① The transmit/receive shift completion flag of the serial I/O2 control register is "1" after transmit/receive shift is completed. In order to set "0" to this flag, set data (dummy data at receive) to the serial I/O2 register by program.

② Bit 7 (transmit/receive shift completion flag) of the serial I/O2 control register is set earlier than the completion of the actual shift operation for a half cycle of shift clock. Accordingly, when the shift completion is checked by using this bit, read/write the serial I/O2 register after a half or more cycle of clock from the setting "1" to this bit is checked.

2.8 A-D converter

This paragraph explains the registers setting method and the notes relevant to the A-D converter.

2.8.1 Memory map

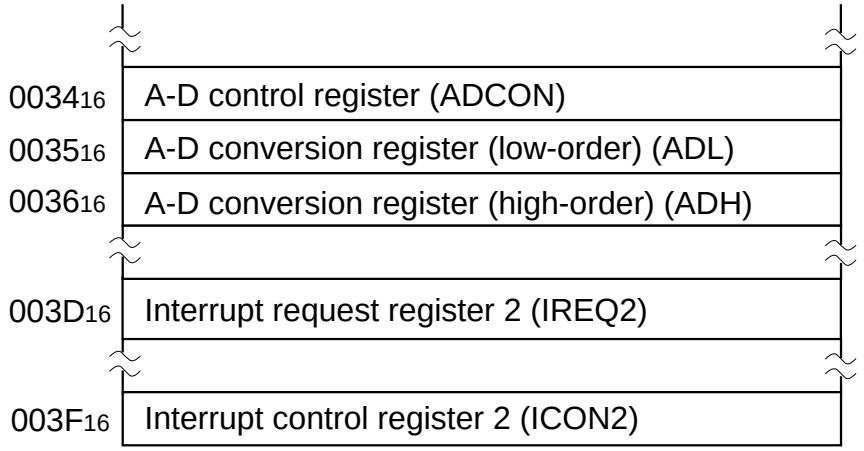


Fig. 2.8.1 Memory map of registers relevant to A-D converter

2.8.2 Relevant registers

A-D control register

b7 b6 b5 b4 b3 b2 b1 b0

A-D control register (ADCON) [Address : 34 16]

B	Name	Function	At reset	R	W
0	Analog input pin selection bits	b2 b1 b0 0 0 0 : P2₀/AN₀ 0 0 1 : P2₁/AN₁ 0 1 0 : P2₂/AN₂ 0 1 1 : P2₃/AN₃ 1 0 0 : P2₄/AN₄ 1 0 1 : P2₅/AN₅ 1 1 0 : P2₆/AN₆ (**Note**) 1 1 1 : P2₇/AN₇ (**Note**)	0	○	○
1			0	○	○
2			0	○	○
3	Nothing is allocated for this bit. This is a write disabled bit. When this bit is read out, the value is "0".		0	○	×
4	AD conversion completion bit	0 : Conversion in progress 1 : Conversion completed	1	○	*
5	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
6			0	○	×
7			0	○	×

Note: These can be used only for the 36-pin package versions.
***:** This bit can be cleared to "0" by program, but cannot be set to "1".

Fig. 2.8.2 Structure of A-D control register

2.8 A-D converter

A-D conversion register (low-order)

b7 b6 b5 b4 b3 b2 b1 b0

A-D conversion register (low-order) (ADL) [Address : 35_{16}]

B	Function	At reset	R	W
0	The read-only register in which the A-D conversion's results are stored. < 8-bit read> b7 b0 b9 b8 b7 b6 b5 b4 b3 b2 < 10-bit read> b7 b0 b7 b6 b5 b4 b3 b2 b1 b0	?	○	×
1		?	○	×
2		?	○	×
3		?	○	×
4		?	○	×
5		?	○	×
6		?	○	×
7		?	○	×

Fig. 2.8.3 Structure of A-D conversion register (low-order)

A-D conversion register (high-order)

b7 b6 b5 b4 b3 b2 b1 b0

A-D conversion register (high-order) (ADH) [Address : 36_{16}]

B	Function	At reset	R	W
0	The read-only register in which the A-D conversion's results are stored.	?	○	×
1	< 10-bit read> b7 b0 b9 b8	?	○	×
2	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".	?	○	×
3		?	○	×
4		?	○	×
5		?	○	×
6		?	○	×
7		?	○	×

Fig. 2.8.4 Structure of A-D conversion register (high-order)

Interrupt request register 2

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt request register 2 (IREQ2) [Address : 3D 16]

B	Name	Function	At reset	R	W
0	Timer Y interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Timer Z interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	Timer A interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	AD converter interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
7			0	○	×

✻: These bits can be cleared to "0" by program, but cannot be set to "1".

Fig. 2.8.5 Structure of Interrupt request register 2

Interrupt control register 2

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 2 (ICON2) [Address : 3F 16]

B	Name	Function	At reset	R	W
0	Timer Y interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer Z interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer A interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	AD conversion interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
7			0	○	×

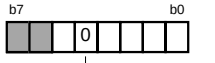
Fig. 2.8.6 Structure of Interrupt control register 2

2.8.3 A-D converter application examples

(1) Setting of A-D converter

Figure 2.8.7 shows the relevant registers setting.

Process 1: Disable A-D conversion interrupt.



Interrupt control register 2 (ICON2) [Address $3F_{16}$]

A-D conversion interrupt disabled

Process 2: Set A-D control register.

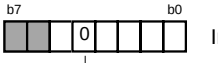
A-D control register (ADCON) [Address 34_{16}]

Analog input pins selected

b2b1b0
0 0 0: $P2_0/AN_0$
0 0 1: $P2_1/AN_1$
0 1 0: $P2_2/AN_2$
0 1 1: $P2_3/AN_3$
1 0 0: $P2_4/AN_4$
1 0 1: $P2_5/AN_5$
1 1 0: $P2_6/AN_6$ (**Note**)
1 1 1: $P2_7/AN_7$ (**Note**)

Note: These can be used only for 36-pin version.

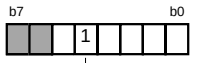
Process 3: In order not to execute the no requested interrupt processing, set “0” (no requested) to the A-D conversion interrupt request bit.



Interrupt request register 2 (IREQ2) [Address $3D_{16}$]

No A-D conversion interrupt request issued

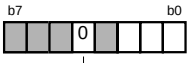
Process 4: When the interrupt is used, set “1” (interrupt enabled) to the A-D conversion interrupt enable bit.



Interrupt control register 2 (ICON2) [Address $3F_{16}$]

A-D conversion interrupt enabled

Process 5: Start A-D conversion.



A-D control register (ADCON) [Address 34_{16}]

Start A-D conversion

Fig. 2.8.7 Relevant registers setting

(2) Control procedure

Outline : The analog input voltage input from a sensor is converted to digital values.

Specifications : •The analog input voltage input from a sensor is converted to digital values.
•$P2_0/AN_0$ pin is used as an analog input pin.

Figure 2.8.8 shows a connection diagram, and Figure 2.8.9 shows an example of control procedure.

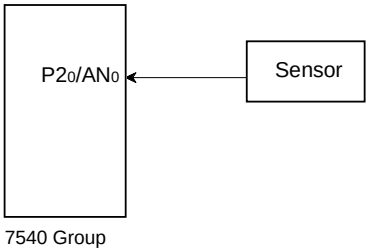
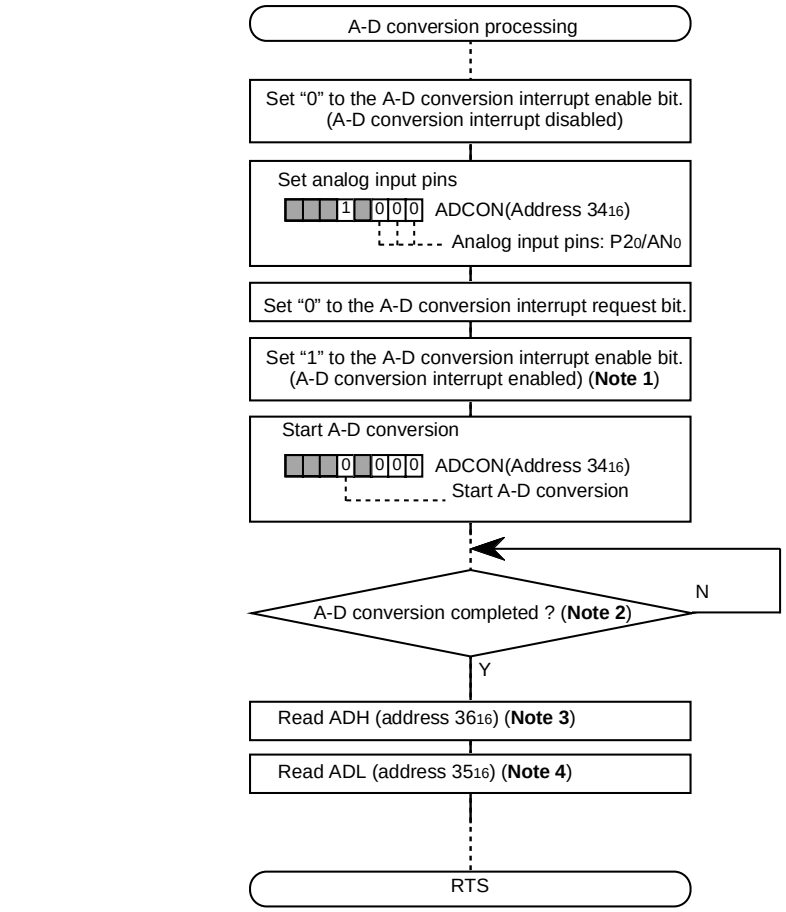


Fig. 2.8.8 Connection diagram



Notes 1: In this case, the A-D conversion interrupt is used.

2: The completion of the A-D conversion is checked by the following;
- The A-D conversion completion bit of the A-D control register is "1".
- The A-D conversion interrupt request bit of the interrupt request register 2 is "1".
- Branch to the A-D conversion interrupt processing routine is executed. (In this time, the A-D conversion interrupt is enabled.)

3: At 10-bit read: the conversion result of the high-order 2 bits (b9, b8) can be read.
At 8-bit read: Not used.

4: At 10-bit read: the conversion result of the low-order 8 bits (b7 to b0) can be read.
At 8-bit read: the conversion result of b7 to b0 can be read.

Fig. 2.8.9 Control procedure

2.8.4 Notes on A-D converter

Notes on A-D converter are described below.

(1) Analog input pin

Figure 2.8.10 shows the internal equivalent circuit of an analog input. In order to execute the A-D conversion correctly, to complete the charge to an internal capacitor within the specified time is required. The maximum output impedance of the analog input source required to complete the charge to a capacitor within the specified time is as follows;

About 35 kΩ (at $f(X_{IN})$ = 8 MHz)

When the maximum output impedance exceeds the above value, equip an analog input pin with an external capacitor of 0.01μF to 1μF between an analog input pin and V_{SS}.
Further, be sure to verify the operation of application products on the user side.

● **Reason**

An analog input pin includes the capacitor for analog voltage comparison. Accordingly, when signals from signal source with high impedance are input to an analog input pin, charge and discharge noise generates. This may cause the A-D conversion/comparison precision to be worse.

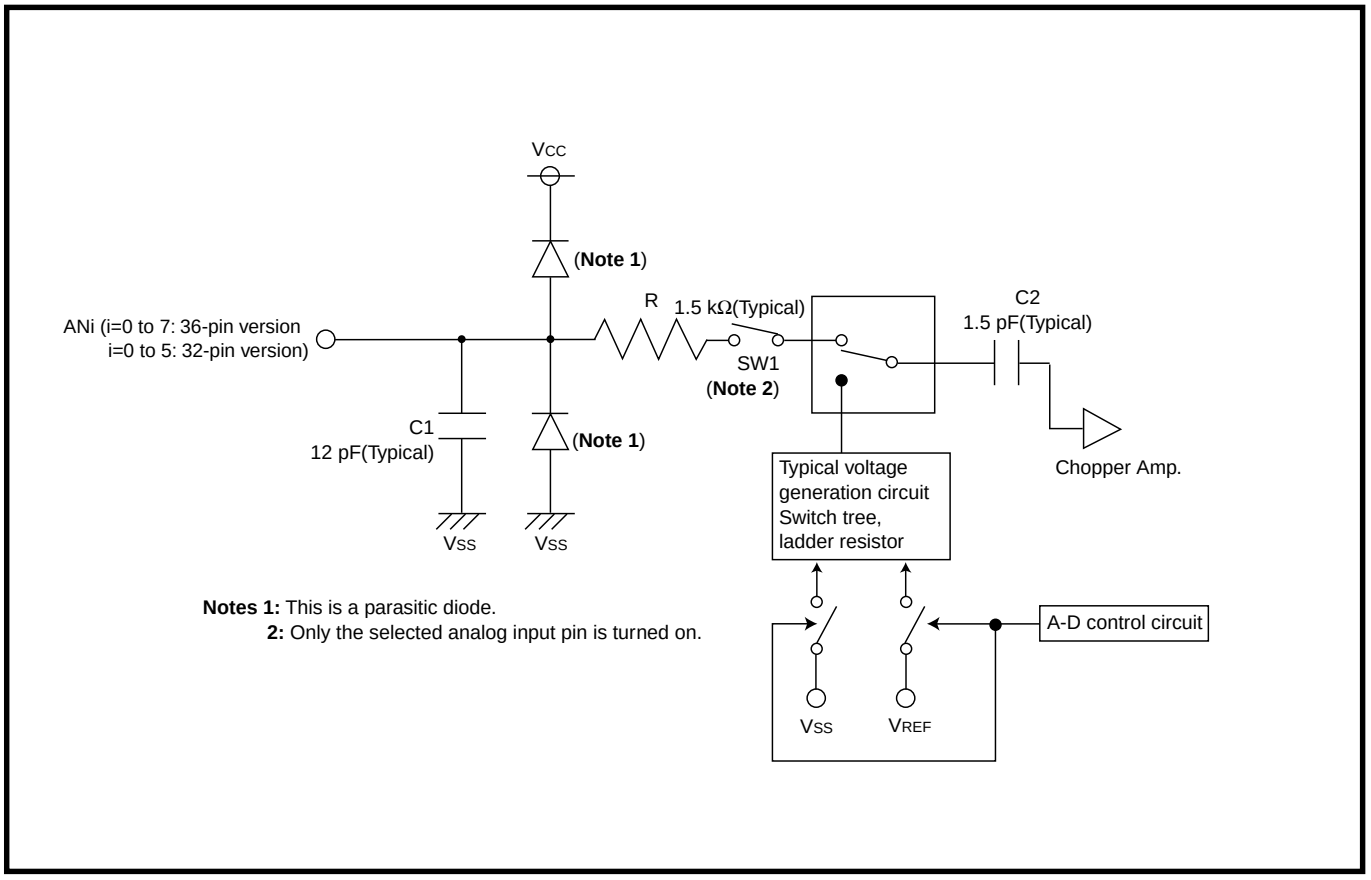


Fig. 2.8.10 Connection diagram

(2) Clock frequency during A-D conversion

The comparator consists of a capacity coupling, and a charge of the capacity will be lost if the clock frequency is too low. Thus, make sure the following during an A-D conversion.

- $f(X_{IN})$ is 500 kHz or more
- Do not execute the **STP** instruction

2.9 Oscillation control

This paragraph explains the registers setting method and the notes relevant to the oscillation control.

2.9.1 Memory map

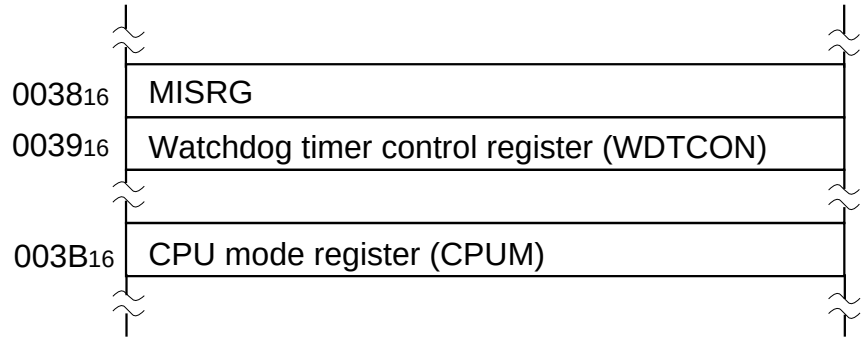


Fig. 2.9.1 Memory map of registers relevant to oscillation control

2.9.2 Relevant registers

MISRG

b7 b6 b5 b4 b3 b2 b1 b0

MISRG [Address : 38_{16}]

B	Name	Function	At reset	R	W
0	Oscillation stabilization time set bit after release of the STP instruction	0 : Set "01_{16}" in timer 1, and "FF_{16}" in prescaler 1 automatically 1 : Not set automatically	0	○	○
1	Ceramic or RC oscillation stop detection function active bit	0 : Detection function inactive 1 : Detection function active	0	○	○
2	These are reserved bits. Do not write "1" to these bits.		0	○	×
3			0	○	×
4	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
5			0	○	×
6			0	○	×
7	Oscillation stop detection status bit	0 : Oscillation stop not detected 1 : Oscillation stop detected	(**Note**)	○	×

Note: "0" at normal reset
"1" at reset by detecting the oscillation stop

Fig. 2.9.2 Structure of MISRG

2.9 Oscillation control

Watchdog timer control register

b7 b6 b5 b4 b3 b2 b1 b0

Watchdog timer control register (WDTCON) [Address : 39 16]

B	Name	Function	At reset	R	W
0	Watchdog timer H (The high-order 6 bits are read-only bits.)		1	○	×
1			1	○	×
2			1	○	×
3			1	○	×
4			1	○	×
5			1	○	×
6	STP instruction disable bit	0 : STP instruction enabled 1 : STP instruction disabled	0	○	○
7	Watchdog timer H count source selection bit	0 : Watchdog timer L underflow 1 : f(XIN)/16	0	○	○

Fig. 2.9.3 Structure of Watchdog timer control register

CPU mode register

b7 b6 b5 b4 b3 b2 b1 b0

CPU mode register (CPUM) [Address : 3B 16]

B	Name	Function	At reset	R	W
0	Processor mode bits (**Note 1**)	b1 b0 0 0 : Single-chip mode 0 1 : Not available 1 0 : Not available 1 1 : Not available	0	○	○
1			0	○	○
2	Stack page selection bit	0 : 0 page 1 : 1 page	0	○	○
3	Ring oscillator oscillation control bit	0 : Ring oscillator oscillation enabled 1 : Ring oscillator oscillation stop	0	○	○
4	XIN oscillation control bit	0 : Ceramic or RC oscillation enabled 1 : Ceramic or RC oscillation stop	0	○	○
5	Oscillation mode selection bit (**Note 1**)	0 : Ceramic oscillation 1 : RC oscillation	0	○	○
6	Clock division ratio selection bits	b7 b6 0 0 : ϕ = f(XIN)/2 (high-speed mode) 0 1 : ϕ = f(XIN)/8 (middle-speed mode) 1 0 : Applied from ring oscillator 1 1 : ϕ = f(XIN) (double-speed mode) (**Note 2**)	0	○	○
7			1	○	○

Notes 1: The bit can be rewritten only once after releasing reset. After rewriting it is disable to write any data to the bit. However, by reset the bit is initialized and can be rewritten, again.
(It is not disable to write any data to the bit for emulator MCU "M37540RSS".)
2: These bits are used only when a ceramic oscillation is selected.
Do not use these when an RC oscillation is selected.

Fig. 2.9.4 Structure of CPU mode register

2.9.3 Application example of ring oscillator

The ring oscillator is the oscillation circuit which is equipped with the 7540 Group. External circuits can be eliminated by using this oscillator as the operation clock or by using this oscillator with a ceramic or RC oscillation circuit. When this oscillator is used as the operation clock, all peripheral functions can be used. In this section, the setting method and application example are explained.

Note: The 7540 Group starts operation by the ring oscillator.

(1) Setting method

Figure 2.9.5 shows the setting method when the ring oscillator is used as the operation clock.

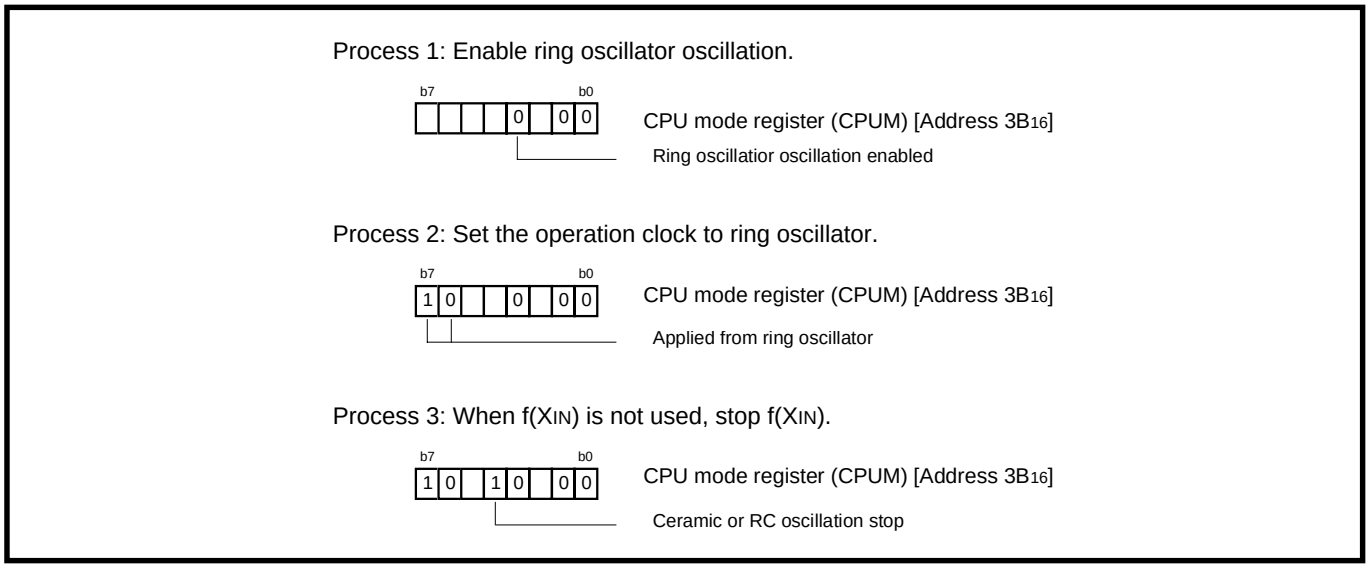


Fig. 2.9.5 Setting method when the ring oscillator is used as the operation clock

(2) Example of control procedure

Outline: The frequency of the ring oscillator is measured, and an error by the power source voltage or temperature is confirmed.

Specifications: • The f(X_{IN}) = 4 MHz is divided by timer Z and 10 ms is detected. The ring oscillator is divided by timer Y.
• The count value of timer Y is read out in the timer Z interrupt processing routine which occurs every 10 ms, and an error from f(X_{IN}) is confirmed.

Figure 2.9.6 shows an example of control procedure.

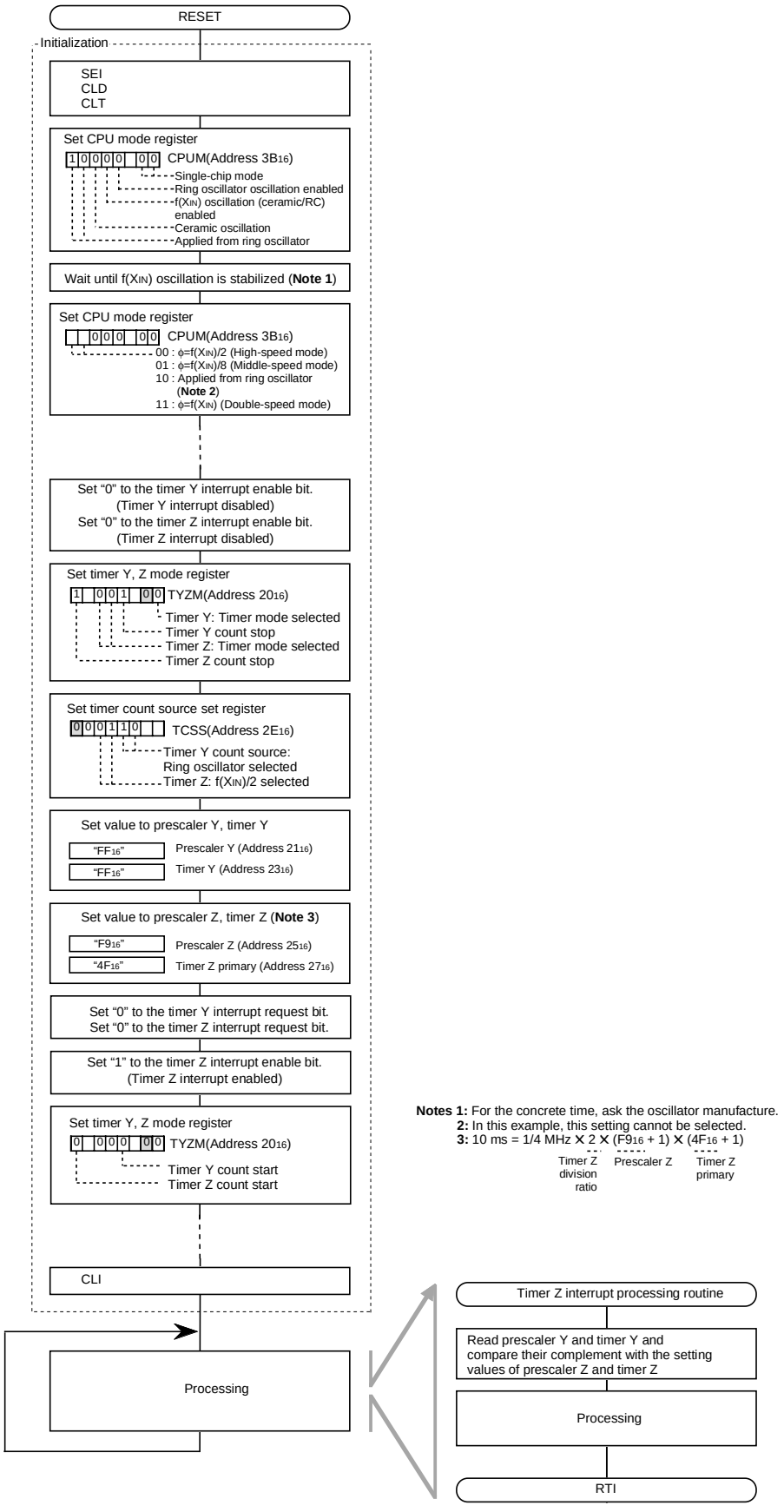


Fig. 2.9.6 Control procedure

2.9.4 Oscillation stop detection circuit

The oscillation stop detection circuit can be used to detect the stop by some failure or disconnection of an external ceramic oscillation circuit.
In this section, the setting method and application example.

(1) Operation description

When the stop of an external oscillation circuit is detected by the oscillation stop detection circuit, the oscillation stop detection status bit of MISRG is set to “1” and the internal reset occurs.
The 7540 Group starts operation by the built-in ring oscillator after system is released from reset. Accordingly, error of the external oscillation circuit can be detected by checking the oscillation stop detection status bit after system starts operation.

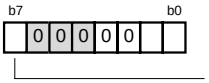
Notes 1: When the stop mode is used, set the oscillation stop detection function to “invalid”.
2: When f(X_{IN}) oscillation is stopped, set the oscillation stop detection function to “invalid”.

(2) Setting method

Figure 2.9.7 shows the initial setting method oscillation stop detection circuit.
Figure 2.9.8 shows the setting method for the oscillation stop detection circuit in the main processing.

* Execute the following set at the beginning of program after system is released from reset.

Process 1: Check that reset by oscillation stop detection is executed by referring the oscillation stop detection status bit.

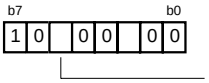


MISRG (MISRG) [Address 38_{16}]

Oscillation stop detection status bit
0: Oscillation stop not detected
1: Oscillation stop detected

- Oscillation stop is detected
  Some error occus in the oscillation circuit.
  Do not switch the operation clock and execute the processing when some error occurs.
- Oscillation stop is not detected
  Execute the Process 2.

Process 2: Select oscillation mode.



CPU mode register (CPUM) [Address $3B_{16}$]

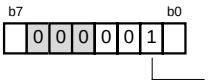
Oscillation mode selection bit (**Note**)
0: Ceramic oscillation
1: RC oscillation

Note: The bit can be rewritten only once after releasing reset. After rewriting it is disable to write any data to the bit. However, by reset the bit is initialized and can be rewritten, again. (It is not disable to write any data to the bit for emulator MCU "M37540RSS").

Process 3: Wait oscillation stabilizing (**Note**).

Note: This process can be eliminated when the RC oscillation is selected. For the oscillation stabilizing time, ask the oscillator manufacture.

Process 4: Set the ceramic or RC oscillation stop detection function active bit.

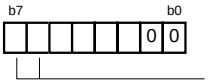


MISRG (MISRG) [Address 38_{16}]

Detection function active (**Note**)

Note: When some error occurs in the oscillation circuit, system is released from reset after setting of Process 4 is executed.

Process 5: Select clock division ratio.



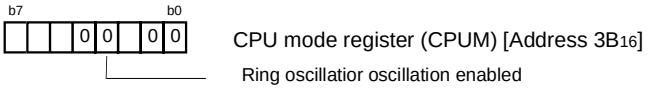
CPU mode register (CPUM) [Address $3B_{16}$]

Clock division ratio selection bits
b7b6
0 0: f(ϕ) = f(X_{IN})/2 (high-speed mode)
0 1: f(ϕ)= f(X_{IN})/8 (middle-speed mode)
1 0: Applied from ring oscillator
1 1: f(ϕ)=f(X_{IN}) (double-speed mode) (**Note**)

Note: These bits are used only when a ceramic oscillation is selected. Do not use these when an RC oscillation is selected.

Fig. 2.9.7 Initial setting method for the oscillation stop detection circuit

Process: Start ring oscillation when it is stopped.



Process 2: Set ceramic or RC oscillation stop detection function active bit.

Note: When some error occurs in the oscillation circuit, system is released from reset after setting of Process 2 is executed.

Fig. 2.9.8 Setting method for the oscillation stop detection circuit in main processing

2.9.5 State transition

In the 7540 Group, the operation clock is selected from the following 4 types.

- f(X_{IN})/2 (high-speed mode)
- f(X_{IN})/8 (middle-speed mode)
- Ring oscillator
- f(X_{IN}) (double-speed mode) (**Note 1**)

Note 1: f(X_{IN}) can be used only at the ceramic oscillation. Do not use f(X_{IN}) at RC oscillation.

Also, in the 7540 Group, the function to stop CPU operation by software and to keep CPU wait in the following 2-type low power dissipation.

- Stop mode with the **STP** instruction (**Notes 2, 3, 4, 5, 6, 7**)
- Wait mode with the **WIT** instruction (**Note 8**)

Notes 2: When the stop mode is used, set the oscillation stop detection function to "invalid".

3: When the stop mode is used, set "0" (**STP** instruction enabled) the **STP** instruction disable bit of the watchdog timer control register.

4: Timer 1 can be used to set the oscillation stabilizing time after release of the **STP** instruction. The oscillation stabilizing time after release of **STP** instruction can be selected from "set automatically"/ "not set automatically" by the oscillation stabilizing time set bit after release of the **STP** instruction of MISRG. When "0" is set to this bit, "01_{16}" is set to timer 1 and "FF_{16}" is set to prescaler 1 automatically. When "1" is set to this bit, nothing is set to timer 1 and prescaler 1. Therefore, set the wait time according to the oscillation stabilizing time of the oscillation. Also, when timer 1 is used, set values again to timer 1 and prescaler 1 after system is returned from the stop mode.

5: The **STP** instruction cannot be used during CPU is operating by the ring oscillator.

6: When the stop mode is used, stop the ring oscillator oscillation.

7: Do not execute the **STP** instruction during the A-D conversion.

8: When the wait mode is used, stop the clock except the operation clock source.

Figure 2.9.9 shows the state transition.

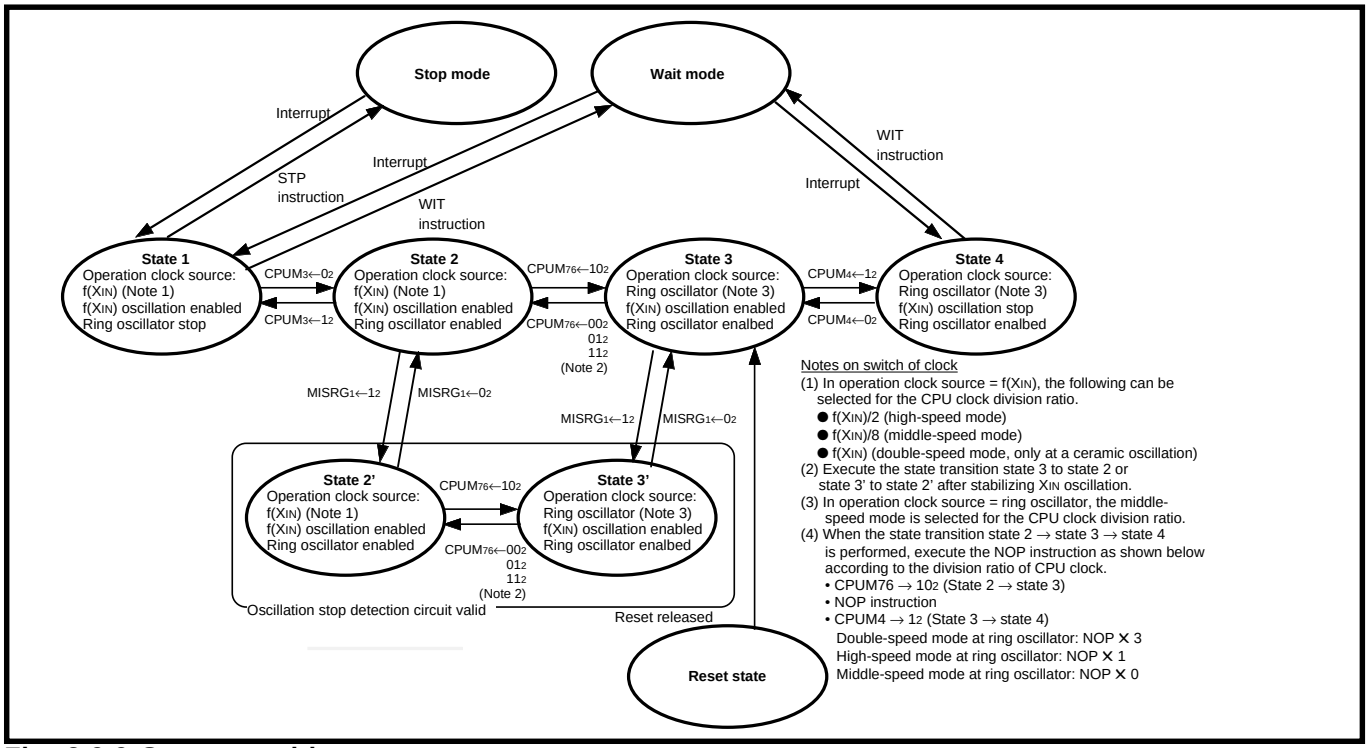


Fig. 2.9.9 State transition

(1) Example of control procedure

Outline: The ring oscillator is used, and the intermittent operation for the low-power dissipation can be realized.

Specifications: A mode is selected from the following modes 1 to 4 according to the usage condition. The return from mode 1 is executed by the timer A interrupt request which occurs every 0.5 s.

Mode 1: Wait mode by the ring oscillator oscillation
Operation clock source: Ring oscillator
CPU stop, ceramic oscillation stop, ring oscillator oscillation

Mode 2: Middle-speed mode by the ring oscillator oscillation
Operation clock source: Ring oscillator
CPU operation, ceramic oscillation stop, ring oscillator oscillation

Mode 3: Middle-speed mode by the ceramic oscillation
Operation clock source: Ceramic oscillation
CPU operation, ceramic oscillation, ring oscillator oscillation

Mode 4: Double-speed mode by the ceramic oscillation
Operation clock source: Ceramic oscillation
CPU operation, ceramic oscillation, ring oscillator oscillation

Figure 2.9.10 shows an example of mode transition and Figure 2.9.11 shows an example of control procedure.

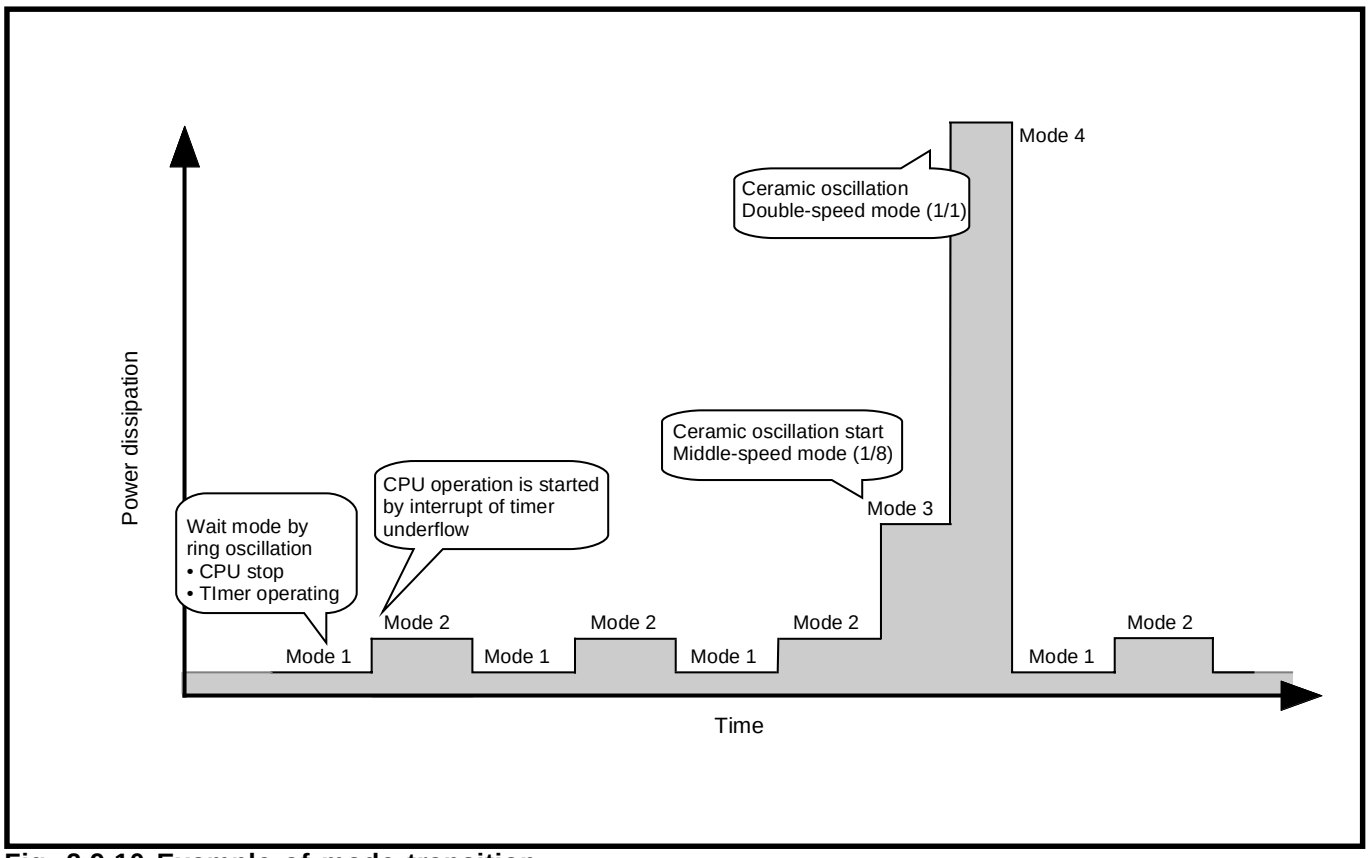
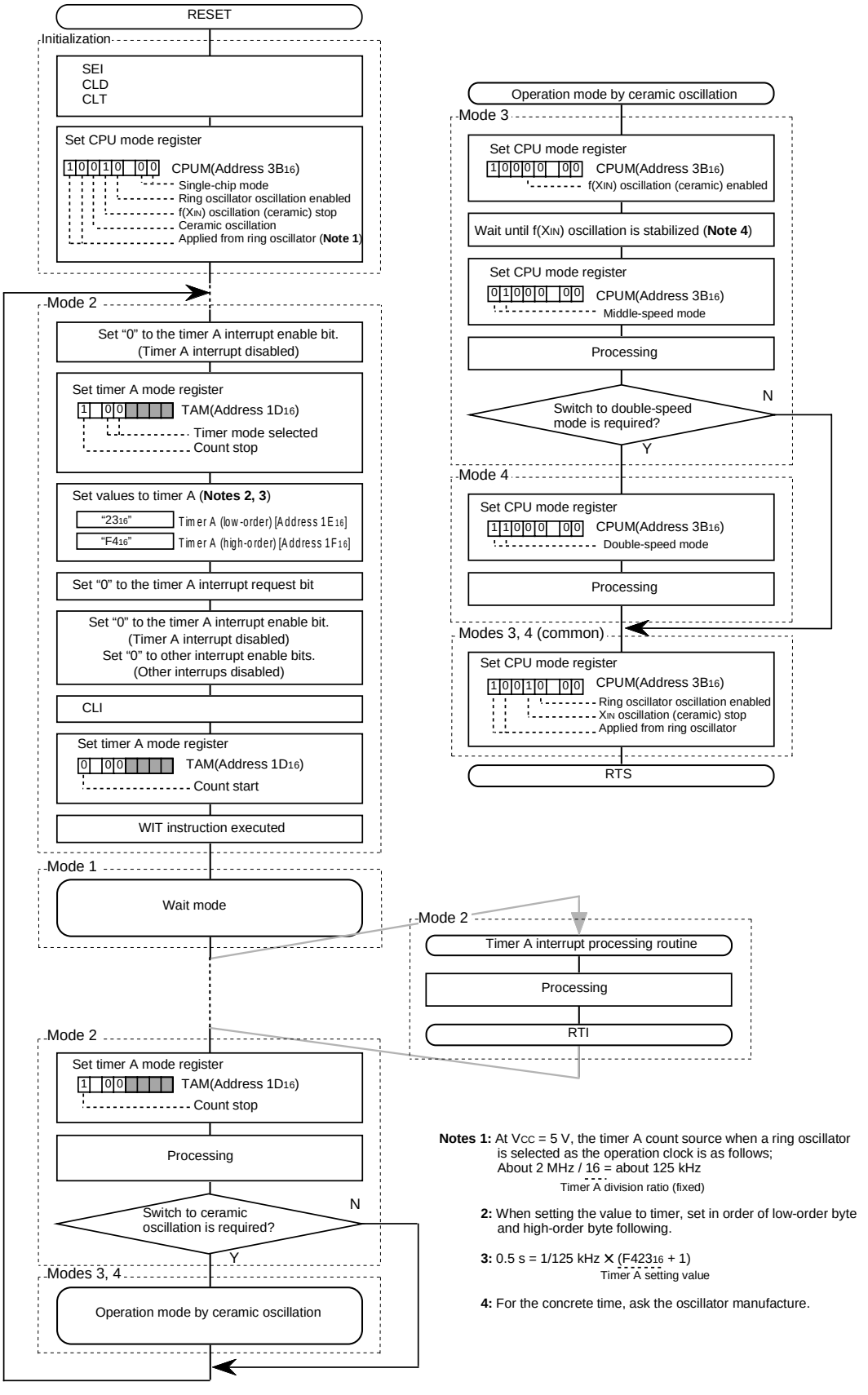


Fig. 2.9.10 Example of mode transition



Notes 1: At V_{CC} = 5 V, the timer A count source when a ring oscillator is selected as the operation clock is as follows;
About 2 MHz / 16 = about 125 kHz
(Timer A division ratio (fixed))

2: When setting the value to timer, set in order of low-order byte and high-order byte following.

3: 0.5 s = 1/125 kHz × ($F423_{16}$ + 1)
(Timer A setting value)

4: For the concrete time, ask the oscillator manufacture.

Fig. 2.9.11 Control procedure

2.9.6 Notes on oscillation stop detection circuit

Notes on using oscillation stop detection circuit are described below.

(1) Note on ring oscillator

① The 7540 Group starts operation by the ring oscillator.

(2) Notes on oscillation circuit stop detection circuit

① When the stop mode is used, set the oscillation stop detection function to “invalid”.

② When f(X_{IN}) oscillation is stopped, set the oscillation stop detection function to “invalid”.

③ The oscillation stop detection circuit is not included in the emulator MCU “M37540RSS”.

(3) Notes on stop mode

① When the stop mode is used, set the oscillation stop detection function to “invalid”.

② When the stop mode is used, set “0” (**STP** instruction enabled) to the **STP** instruction disable bit of the watchdog timer control register.

③ Timer 1 can be used to set the oscillation stabilizing time after release of the **STP** instruction. The oscillation stabilizing time after release of **STP** instruction can be selected from “set automatically”/ “not set automatically” by the oscillation stabilizing time set bit after release of the **STP** instruction of MISRG. When “0” is set to this bit, “01_{16}” is set to timer 1 and “FF_{16}” is set to prescaler 1 automatically. When “1” is set to this bit, nothing is set to timer 1 and prescaler 1. Therefore, set the wait time according to the oscillation stabilizing time of the oscillation. Also, when timer 1 is used, set values again to timer 1 and prescaler 1 after system is returned from the stop mode.

④ The **STP** instruction cannot be used during CPU is operating by the ring oscillator.

⑤ When the stop mode is used, stop the ring oscillator oscillation.

⑥ Do not execute the **STP** instruction during the A-D conversion.

(4) Note on wait mode

① When the wait mode is used, stop the clock except the operation clock source.

(5) Notes on state transition

① When the operation clock source is f(X_{IN}), the CPU clock division ratio can be selected from the following;
- f(X_{IN})/2 (high-speed mode)
- f(X_{IN})/8 (middle-speed mode)
- f(X_{IN}) (double-speed mode)

The double-speed mode can be used only at ceramic oscillation.
Do not use the mode at RC oscillation.

② Stabilize the f(X_{IN}) oscillation to change the operation clock source from the ring oscillator to f(X_{IN}).

③ When the ring oscillation is used as the operation clock, the CPU clock division ratio is the middle-speed mode.

④ When the state transition state 2→state 3→state 4 is performed, execute the NOP instruction as shown below according to the division ratio of CPU clock.
- $CPUM_{76} \rightarrow 10_2$ (State 2→state 3)
- NOP instruction
- $CPUM_4 \rightarrow 1_2$ (State 3→state 4)

Double-speed mode at ring oscillator: NOP×3
High-speed mode at ring oscillator: NOP×1
Middle-speed mode at ring oscillator: NOP×0

CHAPTER 3

APPENDIX

3.1 Electrical characteristics

3.1.1 7540 Group (General purpose)

Applied to: M37540M2-XXXFP/SP/GP, M37540M4-XXXFP/SP/GP, M37540E2FP/SP/GP, M37540E8FP/SP/GP

(1) Absolute Maximum Ratings (General purpose)

Table 3.1.1 Absolute maximum ratings

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Power source voltage	All voltages are based on V_{SS}. Output transistors are cut off.	–0.3 to 6.5 (**Note 1**)	V
V_I	Input voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, V_{REF}		–0.3 to V_{CC} + 0.3	V
V_I	Input voltage $\overline{RESET}$, X_{IN}		–0.3 to V_{CC} + 0.3	V
V_I	Input voltage CNV_{SS} (**Note 2**)		–0.3 to 13	V
V_O	Output voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, X_{OUT}		–0.3 to V_{CC} + 0.3	V
P_d	Power dissipation	Ta = 25°C	300 (**Note 3**)	mW
T_{opr}	Operating temperature		–20 to 85	°C
T_{stg}	Storage temperature		–40 to 125	°C

Notes 1: This is the rating value for the Mask ROM version.
The rating value for the One Time PROM version is –0.3 to 7.0 V.
2: It is a rating only for the One Time PROM version. Connect to V_{SS} for the mask ROM version.
3: 200 mW for the 32P6U package product.

(2) Recommended Operating Conditions (General purpose)

Table 3.1.2 Recommended operating conditions (1)
(V_{CC} = 2.2 to 5.5 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
V_{CC}	Power source voltage (ceramic)	f(X_{IN}) = 8 MHz (High-, Middle-speed mode)	4.0	5.0	5.5	V
		f(X_{IN}) = 4 MHz (High-, Middle-speed mode)	2.4	5.0	5.5	V
		f(X_{IN}) = 2 MHz (High-, Middle-speed mode)	2.2	5.0	5.5	V
		f(X_{IN}) = 6 MHz (Double-speed mode)	4.5	5.0	5.5	V
		f(X_{IN}) = 4 MHz (Double-speed mode)	4.0	5.0	5.5	V
		f(X_{IN}) = 2 MHz (Double-speed mode)	2.4	5.0	5.5	V
		f(X_{IN}) = 1 MHz (Double-speed mode)	2.2	5.0	5.5	V
	Power source voltage (RC)	f(X_{IN}) = 4 MHz (High-, Middle-speed mode)	4.0	5.0	5.5	V
		f(X_{IN}) = 2 MHz (High-, Middle-speed mode)	2.4	5.0	5.5	V
		f(X_{IN}) = 1 MHz (High-, Middle-speed mode)	2.2	5.0	5.5	V
V_{SS}	Power source voltage			0		V
V_{REF}	Analog reference voltage		2.0		V_{CC}	V
V_{IH}	"H" input voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$		0.8V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage (TTL input level selected) $P1_0$, $P1_2$, $P1_3$, $P3_6$, $P3_7$ (Note 1)		2.0		V_{CC}	V
V_{IH}	"H" input voltage $\overline{RESET}$, X_{IN}		0.8V_{CC}		V_{CC}	V
V_{IL}	"L" input voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$		0		0.3V_{CC}	V
V_{IL}	"L" input voltage (TTL input level selected) $P1_0$, $P1_2$, $P1_3$, $P3_6$, $P3_7$ (Note 1)		0		0.8	V
V_{IL}	"L" input voltage $\overline{RESET}$, CNV_{SS}		0		0.2V_{CC}	V
V_{IL}	"L" input voltage X_{IN}		0		0.16V_{CC}	V
$\Sigma I_{OH(peak)}$	"H" total peak output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$				–80	mA
$\Sigma I_{OL(peak)}$	"L" total peak output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$				80	mA
$\Sigma I_{OL(peak)}$	"L" total peak output current (Note 2) $P3_0$–$P3_6$				60	mA
$\Sigma I_{OH(avg)}$	"H" total average output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$				–40	mA
$\Sigma I_{OL(avg)}$	"L" total average output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$				40	mA
$\Sigma I_{OL(avg)}$	"L" total average output current (Note 2) $P3_0$–$P3_6$				30	mA

Note 1: Vcc = 4.0 to 5.5V

2: The total output current is the sum of all the currents flowing through all the applicable ports. The total average current is an average value measured over 100 ms. The total peak current is the peak value of all the currents.

Table 3.1.3 Recommended operating conditions (2)
(V_{CC} = 2.2 to 5.5 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
$I_{OH(peak)}$	"H" peak output current (Note 1)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$			–10	mA
$I_{OL(peak)}$	"L" peak output current (Note 1)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$			10	mA
$I_{OL(peak)}$	"L" peak output current (Note 1)	$P3_0$–$P3_6$			30	mA
$I_{OH(avg)}$	"H" average output current (Note 2)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$			–5	mA
$I_{OL(avg)}$	"L" average output current (Note 2)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$			5	mA
$I_{OL(avg)}$	"L" average output current (Note 2)	$P3_0$–$P3_6$			15	mA
f(X_{IN})	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 4.5 to 5.5 V Double-speed mode			6	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 4.0 to 5.5 V Double-speed mode			4	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 2.4 to 5.5 V Double-speed mode			2	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 2.2 to 5.5 V Double-speed mode			1	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 4.0 to 5.5 V High-, Middle-speed mode			8	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 2.4 to 5.5 V High-, Middle-speed mode			4	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 2.2 to 5.5 V High-, Middle-speed mode			2	MHz
	Internal clock oscillation frequency (Note 3) at RC oscillation	V_{CC} = 4.0 to 5.5 V High-, Middle-speed mode			4	MHz
	Internal clock oscillation frequency (Note 3) at RC oscillation	V_{CC} = 2.4 to 5.5 V High-, Middle-speed mode			2	MHz
	Internal clock oscillation frequency (Note 3) at RC oscillation	V_{CC} = 2.2 to 5.5 V High-, Middle-speed mode			1	MHz

Notes 1: The peak output current is the peak current flowing in each port.
2: The average output current I_{OL} (avg), I_{OH} (avg) in an average value measured over 100 ms.
3: When the oscillation frequency has a duty cycle of 50 %.

(3) Electrical Characteristics (General purpose)

Table 3.1.4 Electrical characteristics (1)
V_{CC} = 2.2 to 5.5 V, V_{SS} = 0 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
V_{OH}	"H" output voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$ (Note 1)	I_{OH} = –5 mA V_{CC} = 4.0 to 5.5 V	V_{CC}–1.5			V
		I_{OH} = –1.0 mA V_{CC} = 2.2 to 5.5 V	V_{CC}–1.0			V
V_{OL}	"L" output voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$	I_{OL} = 5 mA V_{CC} = 4.0 to 5.5 V			1.5	V
		I_{OL} = 1.5 mA V_{CC} = 4.0 to 5.5 V			0.3	V
		I_{OL} = 1.0 mA V_{CC} = 2.2 to 5.5 V			1.0	V
V_{OL}	"L" output voltage $P3_0$–$P3_6$	I_{OL} = 15 mA V_{CC} = 4.0 to 5.5 V			2.0	V
		I_{OL} = 1.5 mA V_{CC} = 4.0 to 5.5 V			0.3	V
		I_{OL} = 10 mA V_{CC} = 2.2 to 5.5 V			1.0	V
V_{T+}–V_{T-}	Hysteresis $CNTR_0$, $CNTR_1$, INT_0, INT_1(Note 2) $P0_0$–$P0_7$ (Note 3)			0.4		V
V_{T+}–V_{T-}	Hysteresis RxD, S_{CLK1}, S_{CLK2}, S_{DATA2} (Note 2)			0.5		V
V_{T+}–V_{T-}	Hysteresis $\overline{RESET}$			0.5		V
I_{IH}	"H" input current $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$	V_I = V_{CC} (Pin floating. Pull up transistors "off")			5.0	µA
I_{IH}	"H" input current $\overline{RESET}$	V_I = V_{CC}			5.0	µA
I_{IH}	"H" input current X_{IN}	V_I = V_{CC}		4.0		µA
I_{IL}	"L" input current $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$	V_I = V_{SS} (Pin floating. Pull up transistors "off")			–5.0	µA
I_{IL}	"L" input current $\overline{RESET}$, CNV_{SS}	V_I = V_{SS}			–5.0	µA
I_{IL}	"L" input current X_{IN}	V_I = V_{SS}		–4.0		µA
I_{IL}	"L" input current $P0_0$–$P0_7$, $P3_0$–$P3_7$	V_I = V_{SS} (Pull up transistors "on")		–0.2	–0.5	mA
V_{RAM}	RAM hold voltage	When clock stopped	2.0		5.5	V
R_{OSC}	Ring oscillator oscillation frequency	V_{CC} = 5.0 V, Ta = 25 °C	1000	2000	3000	kHz
D_{OSC}	Oscillation stop detection circuit detection frequency	V_{CC} = 5.0 V, Ta = 25 °C	62.5	125	187.5	kHz

Notes 1: $P1_1$ is measured when the $P1_1$/TxD_1 P-channel output disable bit of the UART control register (bit 4 of address $001B_{16}$) is "0".
2: RxD_1, S_{CLK1}, S_{CLK2}, S_{DATA2}, INT_0, and INT_1 have hysteresises only when bits 0 to 2 of the port P1P3 control register are set to "0" (CMOS level).
3: It is available only when operating key-on wake up.

APPENDIX

3.1 Electrical characteristics

Table 3.1.5 Electrical characteristics (2)
(V_{CC} = 2.2 to 5.5 V, V_{SS} = 0 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions		Limits			Unit
				Min.	Typ.	Max.	
I_{CC}	Power source current	One Time PROM version	High-speed mode, f(X_{IN}) = 8 MHz Output transistors “off”		5.0	8.0	mA
			High-speed mode, f(X_{IN}) = 2 MHz, V_{CC} = 2.2 V Output transistors “off”		0.5	1.5	mA
			Double-speed mode, f(X_{IN}) = 6 MHz Output transistors “off”		6.0	10.0	mA
			Middle-speed mode, f(X_{IN}) = 8 MHz Output transistors “off”		2.0	5.0	mA
			Ring oscillator operation mode, V_{CC} = 5 V Output transistors “off”		350	1000	µA
			f(X_{IN}) = 8 MHz (in WIT state), functions except timer 1 disabled, Output transistors “off”		1.6	3.2	mA
			f(X_{IN}) = 2 MHz, V_{CC} = 2.2 V (in WIT state), functions except timer 1 disabled, Output transistors “off”		0.2		mA
			Ring oscillator operation mode, V_{CC} = 5V Output transistors “off”		150	450	µA
			Increment when A-D conversion is executed f(X_{IN}) = 8 MHz, V_{CC} = 5 V		0.5		mA
			All oscillation stopped (in STP state) Output transistors “off” — Ta = 25 °C		0.1	1.0	µA
			All oscillation stopped (in STP state) Output transistors “off” — Ta = 85 °C			10	µA
		Mask ROM version	High-speed mode, f(X_{IN}) = 8 MHz Output transistors “off”		3.5	6.5	mA
			High-speed mode, f(X_{IN}) = 2 MHz, V_{CC} = 2.2 V Output transistors “off”		0.4	1.2	mA
			Double-speed mode, f(X_{IN}) = 6 MHz Output transistors “off”		4.5	8.0	mA
			Middle-speed mode, f(X_{IN}) = 8 MHz Output transistors “off”		2.0	5.0	mA
			Ring oscillator operation mode, V_{CC} = 5 V Output transistors “off”		300	900	µA
			f(X_{IN}) = 8 MHz (in WIT state), functions except timer 1 disabled, Output transistors “off”		1.6	3.2	mA
			f(X_{IN}) = 2 MHz, V_{CC} = 2.2 V (in WIT state), functions except timer 1 disabled, Output transistors “off”		0.2		mA
			Ring oscillator operation mode, V_{CC} = 5V Output transistors “off”		150	450	µA
			Increment when A-D conversion is executed f(X_{IN}) = 8 MHz, V_{CC} = 5 V		0.5		mA
			All oscillation stopped (in STP state) Output transistors “off” — Ta = 25 °C		0.1	1.0	µA
			All oscillation stopped (in STP state) Output transistors “off” — Ta = 85 °C			10	µA

(4) A-D Converter Characteristics (General purpose)

Table 3.1.6 A-D Converter characteristics
(V_{CC} = 2.7 to 5.5 V, V_{SS} = 0 V, Ta = –20 to 85 °C, unless otherwise noted)

	Symbol	Parameter	Test conditions	Limits			Unit
				Min.	Typ.	Max.	
One Time PROM version	—	Resolution				10	Bits
	—	Linearity error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±3	LSB
	—	Differential nonlinear error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±0.9	LSB
	V_{OT}	Zero transition voltage	V_{CC} = V_{REF} = 5.12 V	0	5	20	mV
			V_{CC} = V_{REF} = 3.072 V	0	3	15	mV
	V_{FST}	Full scale transition voltage	V_{CC} = V_{REF} = 5.12 V	5105	5115	5125	mV
			V_{CC} = V_{REF} = 3.072 V	3060	3069	3075	mV
	t_{CONV}	Conversion time				122	$t_{c(XIN)}$
	R_{LADDER}	Ladder resistor			55		kΩ
	I_{VREF}	Reference power source input current	V_{REF} = 5.0 V	50	150	200	µA
			V_{REF} = 3.0 V	50	70	120	
	$I_{I(AD)}$	A-D port input current				5.0	µA
Mask ROM version	—	Resolution				10	Bits
	—	Linearity error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±3	LSB
	—	Differential nonlinear error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±1.5	LSB
	V_{OT}	Zero transition voltage	V_{CC} = V_{REF} = 5.12 V	0	15	35	mV
			V_{CC} = V_{REF} = 3.072 V	0	9	21	mV
	V_{FST}	Full scale transition voltage	V_{CC} = V_{REF} = 5.12 V	5105	5125	5150	mV
			V_{CC} = V_{REF} = 3.072 V	3060	3075	3090	mV
	t_{CONV}	Conversion time				122	$t_{c(XIN)}$
	R_{LADDER}	Ladder resistor			55		kΩ
	I_{VREF}	Reference power source input current	V_{REF} = 5.0 V	50	150	200	µA
			V_{REF} = 3.0 V	50	70	120	
	$I_{I(AD)}$	A-D port input current				5.0	µA

(5) Timing Requirements (General purpose)

Table 3.1.7 Timing requirements (1)
(V_{CC} = 4.0 to 5.5 V, V_{SS} = 0 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{W(\overline{RESET})}$	Reset input “L” pulse width	2			µs
$t_{C(XIN)}$	External clock input cycle time	125			ns
$t_{WH(XIN)}$	External clock input “H” pulse width	50			ns
$t_{WL(XIN)}$	External clock input “L” pulse width	50			ns
$t_{C(CNTR0)}$	CNTR0 input cycle time	200			ns
$t_{WH(CNTR0)}$	CNTR0, INT0, INT1, input “H” pulse width	80			ns
$t_{WL(CNTR0)}$	CNTR0, INT0, INT1, input “L” pulse width	80			ns
$t_{C(CNTR1)}$	CNTR1 input cycle time	2000			ns
$t_{WH(CNTR1)}$	CNTR1 input “H” pulse width	800			ns
$t_{WL(CNTR1)}$	CNTR1 input “L” pulse width	800			ns
$t_{C(SCLK1)}$	Serial I/O1 clock input cycle time (Note)	800			ns
$t_{WH(SCLK1)}$	Serial I/O1 clock input “H” pulse width (Note)	370			ns
$t_{WL(SCLK1)}$	Serial I/O1 clock input “L” pulse width (Note)	370			ns
$t_{su(RxD1–SCLK1)}$	Serial I/O1 input set up time	220			ns
$t_{h(SCLK1–RxD1)}$	Serial I/O1 input hold time	100			ns
$t_{C(SCLK2)}$	Serial I/O2 clock input cycle time	1000			ns
$t_{WH(SCLK2)}$	Serial I/O2 clock input “H” pulse width	400			ns
$t_{WL(SCLK2)}$	Serial I/O2 clock input “L” pulse width	400			ns
$t_{su(SDATA2–SCLK2)}$	Serial I/O2 input set up time	200			ns
$t_{h(SCLK2–SDATA2)}$	Serial I/O2 input hold time	200			ns

Note: In this time, bit 6 of the serial I/O1 control register (address $001A_{16}$) is set to “1” (clock synchronous serial I/O1 is selected).
When bit 6 of the serial I/O1 control register is “0” (clock asynchronous serial I/O1 is selected), the rating values are divided by 4.

Table 3.1.8 Timing requirements (2)
(V_{CC} = 2.4 to 5.5 V, V_{SS} = 0 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{W(\overline{RESET})}$	Reset input “L” pulse width	2			µs
$t_{C(XIN)}$	External clock input cycle time	250			ns
$t_{WH(XIN)}$	External clock input “H” pulse width	100			ns
$t_{WL(XIN)}$	External clock input “L” pulse width	100			ns
$t_{C(CNTR0)}$	CNTR0 input cycle time	500			ns
$t_{WH(CNTR0)}$	CNTR0, INT0, INT1, input “H” pulse width	230			ns
$t_{WL(CNTR0)}$	CNTR0, INT0, INT1, input “L” pulse width	230			ns
$t_{C(CNTR1)}$	CNTR1 input cycle time	4000			ns
$t_{WH(CNTR1)}$	CNTR1 input “H” pulse width	1600			ns
$t_{WL(CNTR1)}$	CNTR1 input “L” pulse width	1600			ns
$t_{C(SCLK1)}$	Serial I/O1 clock input cycle time (Note)	2000			ns
$t_{WH(SCLK1)}$	Serial I/O1 clock input “H” pulse width (Note)	950			ns
$t_{WL(SCLK1)}$	Serial I/O1 clock input “L” pulse width (Note)	950			ns
$t_{su(RxD1–SCLK1)}$	Serial I/O1 input set up time	400			ns
$t_{h(SCLK1–RxD1)}$	Serial I/O1 input hold time	200			ns
$t_{C(SCLK2)}$	Serial I/O2 clock input cycle time	2000			ns
$t_{WH(SCLK2)}$	Serial I/O2 clock input “H” pulse width	950			ns
$t_{WL(SCLK2)}$	Serial I/O2 clock input “L” pulse width	950			ns
$t_{su(SDATA2–SCLK2)}$	Serial I/O2 input set up time	400			ns
$t_{h(SCLK2–SDATA2)}$	Serial I/O2 input hold time	400			ns

Note: In this time, bit 6 of the serial I/O1 control register (address $001A_{16}$) is set to “1” (clock synchronous serial I/O1 is selected).
When bit 6 of the serial I/O1 control register is “0” (clock asynchronous serial I/O1 is selected), the rating values are divided by 4.

Table 3.1.9 Timing requirements (3)
(V_{CC} = 2.2 to 5.5 V, V_{SS} = 0 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{W(\overline{RESET})}$	Reset input “L” pulse width	2			µs
$t_{C(XIN)}$	External clock input cycle time	500			ns
$t_{WH(XIN)}$	External clock input “H” pulse width	200			ns
$t_{WL(XIN)}$	External clock input “L” pulse width	200			ns
$t_{C(CNTR0)}$	CNTR0 input cycle time	1000			ns
$t_{WH(CNTR0)}$	CNTR0, INT0, INT1, input “H” pulse width	460			ns
$t_{WL(CNTR0)}$	CNTR0, INT0, INT1, input “L” pulse width	460			ns
$t_{C(CNTR1)}$	CNTR1 input cycle time	8000			ns
$t_{WH(CNTR1)}$	CNTR1 input “H” pulse width	3200			ns
$t_{WL(CNTR1)}$	CNTR1 input “L” pulse width	3200			ns
$t_{C(SCLK1)}$	Serial I/O1 clock input cycle time (Note)	4000			ns
$t_{WH(SCLK1)}$	Serial I/O1 clock input “H” pulse width (Note)	1900			ns
$t_{WL(SCLK1)}$	Serial I/O1 clock input “L” pulse width (Note)	1900			ns
$t_{su(RxD1–SCLK1)}$	Serial I/O1 input set up time	800			ns
$t_{h(SCLK1–RxD1)}$	Serial I/O1 input hold time	400			ns
$t_{C(SCLK2)}$	Serial I/O2 clock input cycle time	4000			ns
$t_{WH(SCLK2)}$	Serial I/O2 clock input “H” pulse width	1900			ns
$t_{WL(SCLK2)}$	Serial I/O2 clock input “L” pulse width	1900			ns
$t_{su(SDATA2–SCLK2)}$	Serial I/O2 input set up time	800			ns
$t_{h(SCLK2–SDATA2)}$	Serial I/O2 input hold time	800			ns

Note: In this time, bit 6 of the serial I/O1 control register (address $001A_{16}$) is set to “1” (clock synchronous serial I/O1 is selected).
When bit 6 of the serial I/O1 control register is “0” (clock asynchronous serial I/O1 is selected), the rating values are divided by 4.

(6) Switching Characteristics (General purpose)

Table 3.1.10 Switching characteristics (1)
(V_{CC} = 4.0 to 5.5 V, V_{SS} = 0 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
t_{WH}(SCLK1)	Serial I/O1 clock output “H” pulse width	t_C(SCLK1)/2–30			ns
t_{WL}(SCLK1)	Serial I/O1 clock output “L” pulse width	t_C(SCLK1)/2–30			ns
t_d(SCLK1–TxD1)	Serial I/O1 output delay time			140	ns
t_v(SCLK1–TxD1)	Serial I/O1 output valid time	–30			ns
t_r(SCLK1)	Serial I/O1 clock output rising time			30	ns
t_f(SCLK1)	Serial I/O1 clock output falling time			30	ns
t_{WH}(SCLK2)	Serial I/O2 clock output “H” pulse width	t_C(SCLK2)/2–30			ns
t_{WL}(SCLK2)	Serial I/O2 clock output “L” pulse width	t_C(SCLK2)/2–30			ns
t_d(SCLK2–SDATA2)	Serial I/O2 output delay time			140	ns
t_v(SCLK2–SDATA2)	Serial I/O2 output valid time	0			ns
t_r(SCLK2)	Serial I/O2 clock output rising time			30	ns
t_f(SCLK2)	Serial I/O2 clock output falling time			30	ns
t_r(CMOS)	CMOS output rising time (Note 1)		10	30	ns
t_f(CMOS)	CMOS output falling time (Note 1)		10	30	ns

Note 1: Pin XOUT is excluded.

Table 3.1.11 Switching characteristics (2)
(V_{CC} = 2.4 to 5.5 V, V_{SS} = 0 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
t_{WH}(SCLK1)	Serial I/O1 clock output “H” pulse width	t_C(SCLK1)/2–50			ns
t_{WL}(SCLK1)	Serial I/O1 clock output “L” pulse width	t_C(SCLK1)/2–50			ns
t_d(SCLK1–TxD1)	Serial I/O1 output delay time			350	ns
t_v(SCLK1–TxD1)	Serial I/O1 output valid time	–30			ns
t_r(SCLK1)	Serial I/O1 clock output rising time			50	ns
t_f(SCLK1)	Serial I/O1 clock output falling time			50	ns
t_{WH}(SCLK2)	Serial I/O2 clock output “H” pulse width	t_C(SCLK2)/2–50			ns
t_{WL}(SCLK2)	Serial I/O2 clock output “L” pulse width	t_C(SCLK2)/2–50			ns
t_d(SCLK2–SDATA2)	Serial I/O2 output delay time			350	ns
t_v(SCLK2–SDATA2)	Serial I/O2 output valid time	0			ns
t_r(SCLK2)	Serial I/O2 clock output rising time			50	ns
t_f(SCLK2)	Serial I/O2 clock output falling time			50	ns
t_r(CMOS)	CMOS output rising time (Note 1)		20	50	ns
t_f(CMOS)	CMOS output falling time (Note 1)		20	50	ns

Note 1: Pin XOUT is excluded.

Table 3.1.12 Switching characteristics (3)
(V_{CC} = 2.2 to 5.5 V, V_{SS} = 0 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
t_{WH}(SCLK1)	Serial I/O1 clock output “H” pulse width	t_C(SCLK1)/2–70			ns
t_{WL}(SCLK1)	Serial I/O1 clock output “L” pulse width	t_C(SCLK1)/2–70			ns
t_d(SCLK1–TxD1)	Serial I/O1 output delay time			450	ns
t_v(SCLK1–TxD1)	Serial I/O1 output valid time	–30			ns
t_r(SCLK1)	Serial I/O1 clock output rising time			70	ns
t_f(SCLK1)	Serial I/O1 clock output falling time			70	ns
t_{WH}(SCLK2)	Serial I/O2 clock output “H” pulse width	t_C(SCLK2)/2–70			ns
t_{WL}(SCLK2)	Serial I/O2 clock output “L” pulse width	t_C(SCLK2)/2–70			ns
t_d(SCLK2–SDATA2)	Serial I/O2 output delay time			450	ns
t_v(SCLK2–SDATA2)	Serial I/O2 output valid time	0			ns
t_r(SCLK2)	Serial I/O2 clock output rising time			70	ns
t_f(SCLK2)	Serial I/O2 clock output falling time			70	ns
t_r(CMOS)	CMOS output rising time (Note 1)		25	70	ns
t_f(CMOS)	CMOS output falling time (Note 1)		25	70	ns

Note 1: Pin XOUT is excluded.

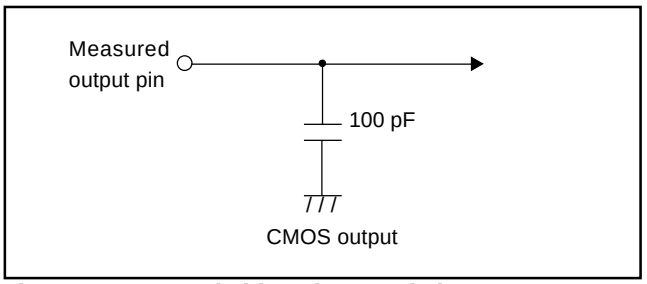


Fig. 3.1.1 Switching characteristics measurement circuit diagram (General purpose)

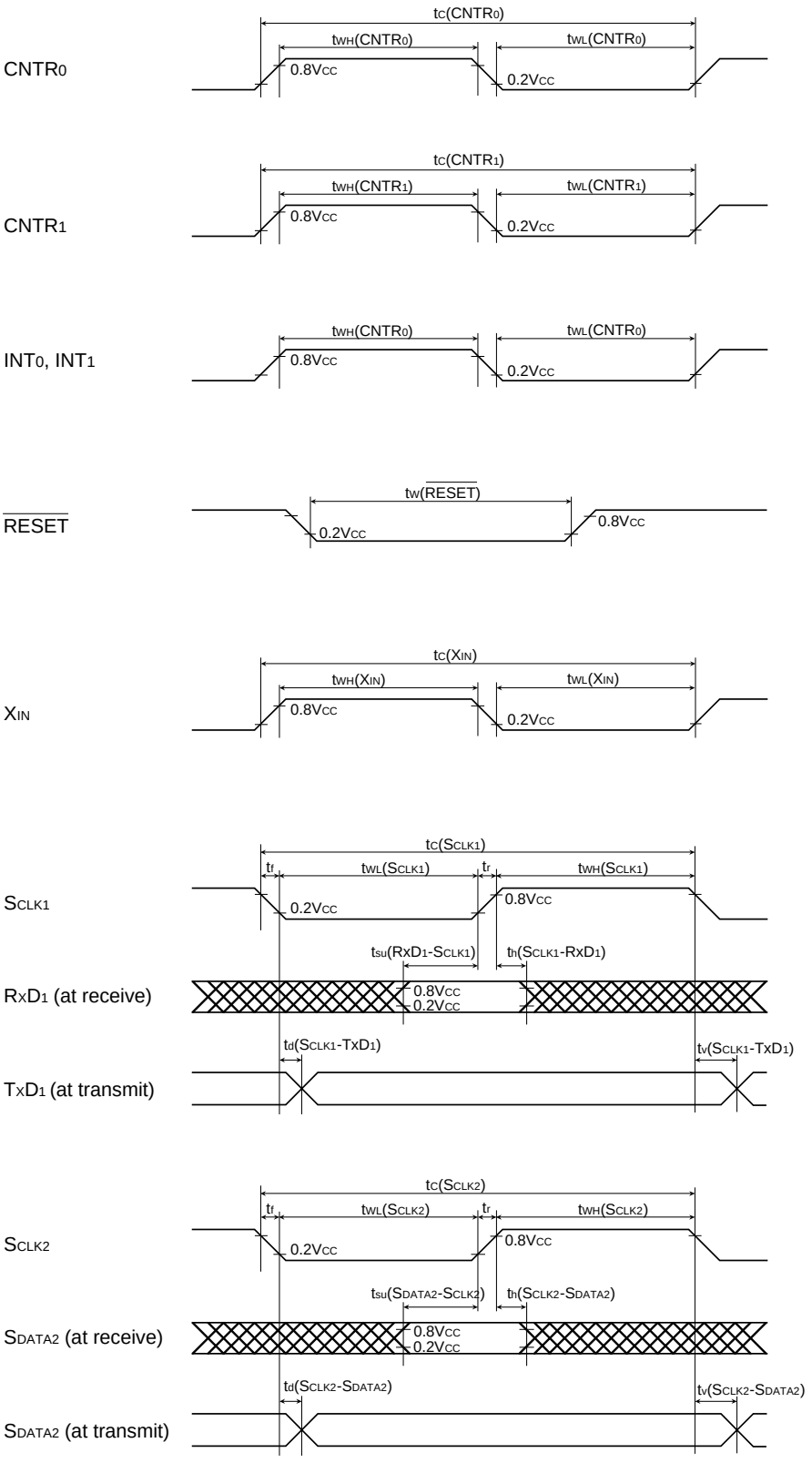


Fig. 3.1.2 Timing chart (General purpose)

3.1.2 7540Group (Extended operating temperature version)

Applied to: M37540M2T-XXXFP/GP, M37540M4T-XXXFP/GP, M37540E8T-XXXFP/GP

(2) Absolute Maximum Ratings (Extended operating temperature version)

Table 3.1.13 Absolute maximum ratings

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Power source voltage	All voltages are based on V_{SS}. Output transistors are cut off.	–0.3 to 6.5 (**Note 1**)	V
V_I	Input voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, V_{REF}		–0.3 to V_{CC} + 0.3	V
V_I	Input voltage $\overline{RESET}$, X_{IN}, CNV_{SS}		–0.3 to V_{CC} + 0.3	V
V_O	Output voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, X_{OUT}		–0.3 to V_{CC} + 0.3	V
P_d	Power dissipation	Ta = 25°C	300 (**Note 2**)	mW
T_{opr}	Operating temperature		–40 to 85	°C
T_{stg}	Storage temperature		–65 to 150	°C

Notes 1: This is the rating value for the Mask ROM version.
The rating value for the One Time PROM version is –0.3 to 7.0 V.
2: 200 mW for the 32P6U package product.

(2) Recommended Operating Conditions (Extended operating temperature version)

Table 3.1.14 Recommended operating conditions (1)
(V_{CC} = 2.4 to 5.5 V, Ta = –40 to 85 °C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
V_{CC}	Power source voltage (ceramic)	f(X_{IN}) = 8 MHz (High-, Middle-speed mode)	4.0	5.0	5.5	V
		f(X_{IN}) = 4 MHz (High-, Middle-speed mode)	2.4	5.0	5.5	V
		f(X_{IN}) = 6 MHz (Double-speed mode)	4.5	5.0	5.5	V
		f(X_{IN}) = 4 MHz (Double-speed mode)	4.0	5.0	5.5	V
		f(X_{IN}) = 2 MHz (Double-speed mode)	2.4	5.0	5.5	V
	Power source voltage (RC)	f(X_{IN}) = 4 MHz (High-, Middle-speed mode)	4.0	5.0	5.5	V
		f(X_{IN}) = 2 MHz (High-, Middle-speed mode)	2.4	5.0	5.5	V
V_{SS}	Power source voltage			0		V
V_{REF}	Analog reference voltage		2.0		V_{CC}	V
V_{IH}	“H” input voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$		0.8V_{CC}		V_{CC}	V
V_{IH}	“H” input voltage (TTL input level selected) $P1_0$, $P1_2$, $P1_3$, $P3_6$, $P3_7$ (Note 1)		2.0		V_{CC}	V
V_{IH}	“H” input voltage $\overline{RESET}$, X_{IN}		0.8V_{CC}		V_{CC}	V
V_{IL}	“L” input voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$		0		0.3V_{CC}	V
V_{IL}	“L” input voltage (TTL input level selected) $P1_0$, $P1_2$, $P1_3$, $P3_6$, $P3_7$ (Note 1)		0		0.8	V
V_{IL}	“L” input voltage $\overline{RESET}$, CNV_{SS}		0		0.2V_{CC}	V
V_{IL}	“L” input voltage X_{IN}		0		0.16V_{CC}	V
$\Sigma I_{OH(peak)}$	“H” total peak output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$				–80	mA
$\Sigma I_{OL(peak)}$	“L” total peak output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$				80	mA
$\Sigma I_{OL(peak)}$	“L” total peak output current (Note 2) $P3_0$–$P3_6$				60	mA
$\Sigma I_{OH(avg)}$	“H” total average output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$				–40	mA
$\Sigma I_{OL(avg)}$	“L” total average output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$				40	mA
$\Sigma I_{OL(avg)}$	“L” total average output current (Note 2) $P3_0$–$P3_6$				30	mA

Note 1: Vcc = 4.0 to 5.5V

2: The total output current is the sum of all the currents flowing through all the applicable ports. The total average current is an average value measured over 100 ms. The total peak current is the peak value of all the currents.

Table 3.1.15 Recommended operating conditions (2)
(V_{CC} = 2.4 to 5.5 V, Ta = –40 to 85 °C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
$I_{OH(peak)}$	"H" peak output current (Note 1)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$			–10	mA
$I_{OL(peak)}$	"L" peak output current (Note 1)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$			10	mA
$I_{OL(peak)}$	"L" peak output current (Note 1)	$P3_0$–$P3_6$			30	mA
$I_{OH(avg)}$	"H" average output current (Note 2)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$			–5	mA
$I_{OL(avg)}$	"L" average output current (Note 2)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$			5	mA
$I_{OL(avg)}$	"L" average output current (Note 2)	$P3_0$–$P3_6$			15	mA
$f(X_{IN})$	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 4.5 to 5.5 V Double-speed mode			6	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 4.0 to 5.5 V Double-speed mode			4	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 2.4 to 5.5 V Double-speed mode			2	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 4.0 to 5.5 V High-, Middle-speed mode			8	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 2.4 to 5.5 V High-, Middle-speed mode			4	MHz
	Internal clock oscillation frequency (Note 3) at RC oscillation	V_{CC} = 4.0 to 5.5 V High-, Middle-speed mode			4	MHz
	Internal clock oscillation frequency (Note 3) at RC oscillation	V_{CC} = 2.4 to 5.5 V High-, Middle-speed mode			2	MHz

Notes 1: The peak output current is the peak current flowing in each port.
2: The average output current I_{OL} (avg), I_{OH} (avg) in an average value measured over 100 ms.
3: When the oscillation frequency has a duty cycle of 50 %.

APPENDIX

3.1 Electrical characteristics

(3) Electrical Characteristics (Extended operating temperature version)

Table 3.1.16 Electrical characteristics (1)
(V_{CC} = 2.4 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
V_{OH}	"H" output voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$ (Note 1)	I_{OH} = –5 mA V_{CC} = 4.0 to 5.5 V	V_{CC}–1.5			V
		I_{OH} = –1.0 mA V_{CC} = 2.4 to 5.5 V	V_{CC}–1.0			V
V_{OL}	"L" output voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$	I_{OL} = 5 mA V_{CC} = 4.0 to 5.5 V			1.5	V
		I_{OL} = 1.5 mA V_{CC} = 4.0 to 5.5 V			0.3	V
		I_{OL} = 1.0 mA V_{CC} = 2.4 to 5.5 V			1.0	V
V_{OL}	"L" output voltage $P3_0$–$P3_6$	I_{OL} = 15 mA V_{CC} = 4.0 to 5.5 V			2.0	V
		I_{OL} = 1.5 mA V_{CC} = 4.0 to 5.5 V			0.3	V
		I_{OL} = 10 mA V_{CC} = 2.4 to 5.5 V			1.0	V
V_{T+}–V_{T-}	Hysteresis $CNTR_0$, $CNTR_1$, INT_0, INT_1(Note 2) $P0_0$–$P0_7$ (Note 3)			0.4		V
V_{T+}–V_{T-}	Hysteresis RxD, S_{CLK1}, S_{CLK2}, S_{DATA2} (Note 2)			0.5		V
V_{T+}–V_{T-}	Hysteresis $\overline{RESET}$			0.5		V
I_{IH}	"H" input current $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$	V_I = V_{CC} (Pin floating. Pull up transistors "off")			5.0	µA
I_{IH}	"H" input current $\overline{RESET}$	V_I = V_{CC}			5.0	µA
I_{IH}	"H" input current X_{IN}	V_I = V_{CC}		4.0		µA
I_{IL}	"L" input current $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$	V_I = V_{SS} (Pin floating. Pull up transistors "off")			–5.0	µA
I_{IL}	"L" input current $\overline{RESET}$, CNV_{SS}	V_I = V_{SS}			–5.0	µA
I_{IL}	"L" input current X_{IN}	V_I = V_{SS}		–4.0		µA
I_{IL}	"L" input current $P0_0$–$P0_7$, $P3_0$–$P3_7$	V_I = V_{SS} (Pull up transistors "on")		–0.2	–0.5	mA
V_{RAM}	RAM hold voltage	When clock stopped	2.0		5.5	V
R_{OSC}	Ring oscillator oscillation frequency	V_{CC} = 5.0 V, Ta = 25 °C	1000	2000	3000	kHz
D_{OSC}	Oscillation stop detection circuit detection frequency	V_{CC} = 5.0 V, Ta = 25 °C	62.5	125	187.5	kHz

Notes 1: $P1_1$ is measured when the $P1_1$/TxD1 P-channel output disable bit of the UART control register (bit 4 of address $001B_{16}$) is "0".
2: RxD1, S_{CLK1}, S_{CLK2}, S_{DATA2}, INT_0, and INT_1 have hysteresises only when bits 0 to 2 of the port P1P3 control register are set to "0" (CMOS level).
3: It is available only when operating key-on wake up.

Table 3.1.17 Electrical characteristics (2)
(V_{CC} = 2.4 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 85 °C, unless otherwise noted)

Symbol	Test conditions			Limits			Unit
				Min.	Typ.	Max.	
I_{CC}	One Time PROM version	High-speed mode, f(X_{IN}) = 8 MHz Output transistors “off”			5.0	8.0	mA
		High-speed mode, f(X_{IN}) = 2 MHz, V_{CC} = 2.4 V Output transistors “off”			0.5	1.5	mA
		Double-speed mode, f(X_{IN}) = 6 MHz, Output transistors “off”			6.0	10.0	mA
		Middle-speed mode, f(X_{IN}) = 8 MHz, Output transistors “off”			2.0	5.0	mA
		Ring oscillator operation mode, V_{CC} = 5 V Output transistors “off”			350	1000	µA
		f(X_{IN}) = 8 MHz (in WIT state), functions except timer 1 disabled, Output transistors “off”			1.6	3.2	mA
		f(X_{IN}) = 2 MHz, V_{CC} = 2.4 V (in WIT state), functions except timer 1 disabled, Output transistors “off”			0.2		mA
		Ring oscillator operation mode, V_{CC} = 5V (in WIT state), functions except timer 1 disabled, Output transistors “off”			150	450	µA
		Increment when A-D conversion is executed f(X_{IN}) = 8 MHz, V_{CC} = 5 V			0.5		mA
		All oscillation stopped (in STP state) Output transistors “off”	Ta = 25 °C		0.1	1.0	µA
			Ta = 85 °C			10	µA
	Mask ROM version	High-speed mode, f(X_{IN}) = 8 MHz Output transistors “off”			3.5	6.5	mA
		High-speed mode, f(X_{IN}) = 2 MHz, V_{CC} = 2.4 V Output transistors “off”			0.4	1.2	mA
		Double-speed mode, f(X_{IN}) = 6 MHz Output transistors “off”			4.5	8.0	mA
		Middle-speed mode, f(X_{IN}) = 8 MHz Output transistors “off”			2.0	5.0	mA
		Ring oscillator operation mode, V_{CC} = 5 V Output transistors “off”			300	900	µA
		f(X_{IN}) = 8 MHz (in WIT state), functions except timer 1 disabled, Output transistors “off”			1.6	3.2	mA
		f(X_{IN}) = 2 MHz, V_{CC} = 2.4 V (in WIT state), functions except timer 1 disabled, Output transistors “off”			0.2		mA
		Ring oscillator operation mode, V_{CC} = 5V (in WIT state), functions except timer 1 disabled, Output transistors “off”			150	450	µA
		Increment when A-D conversion is executed f(X_{IN}) = 8 MHz, V_{CC} = 5 V			0.5		mA
		All oscillation stopped (in STP state) Output transistors “off”	Ta = 25 °C		0.1	1.0	µA
			Ta = 85 °C			10	µA

(4) A-D Converter Characteristics (Extended operating temperature version)

Table 3.1.18 A-D Converter characteristics
(V_{CC} = 2.7 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 85 °C, unless otherwise noted)

	Symbol	Parameter	Test conditions	Limits			Unit
				Min.	Typ.	Max.	
One Time PROM version	—	Resolution				10	Bits
	—	Linearity error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±3	LSB
	—	Differential nonlinear error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±0.9	LSB
	V_{OT}	Zero transition voltage	V_{CC} = V_{REF} = 5.12 V	0	5	20	mV
			V_{CC} = V_{REF} = 3.072 V	0	3	15	mV
	V_{FST}	Full scale transition voltage	V_{CC} = V_{REF} = 5.12 V	5105	5115	5125	mV
			V_{CC} = V_{REF} = 3.072 V	3060	3069	3075	mV
	t_{CONV}	Conversion time				122	tc(X_{IN})
	R_{LADDER}	Ladder resistor			55		kΩ
	I_{VREF}	Reference power source input current	V_{REF} = 5.0 V	50	150	200	µA
			V_{REF} = 3.0 V	50	70	120	
	$I_{I(AD)}$	A-D port input current				5.0	µA
Mask ROM version	—	Resolution				10	Bits
	—	Linearity error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±3	LSB
	—	Differential nonlinear error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±1.5	LSB
	V_{OT}	Zero transition voltage	V_{CC} = V_{REF} = 5.12 V	0	15	35	mV
			V_{CC} = V_{REF} = 3.072 V	0	9	21	mV
	V_{FST}	Full scale transition voltage	V_{CC} = V_{REF} = 5.12 V	5105	5125	5150	mV
			V_{CC} = V_{REF} = 3.072 V	3060	3075	3090	mV
	t_{CONV}	Conversion time				122	tc(X_{IN})
	R_{LADDER}	Ladder resistor			55		kΩ
	I_{VREF}	Reference power source input current	V_{REF} = 5.0 V	50	150	200	µA
			V_{REF} = 3.0 V	30	70	120	
	$I_{I(AD)}$	A-D port input current				5.0	µA

(5) Timing Requirements (Extended operating temperature version)

Table 3.1.19 Timing requirements (1)
(V_{CC} = 4.0 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{W(\overline{RESET})}$	Reset input "L" pulse width	2			µs
$t_{C(XIN)}$	External clock input cycle time	125			ns
$t_{WH(XIN)}$	External clock input "H" pulse width	50			ns
$t_{WL(XIN)}$	External clock input "L" pulse width	50			ns
$t_{C(CNTR0)}$	CNTR0 input cycle time	200			ns
$t_{WH(CNTR0)}$	CNTR0, INT0, INT1, input "H" pulse width	80			ns
$t_{WL(CNTR0)}$	CNTR0, INT0, INT1, input "L" pulse width	80			ns
$t_{C(CNTR1)}$	CNTR1 input cycle time	2000			ns
$t_{WH(CNTR1)}$	CNTR1 input "H" pulse width	800			ns
$t_{WL(CNTR1)}$	CNTR1 input "L" pulse width	800			ns
$t_{C(SCLK1)}$	Serial I/O1 clock input cycle time (Note)	800			ns
$t_{WH(SCLK1)}$	Serial I/O1 clock input "H" pulse width (Note)	370			ns
$t_{WL(SCLK1)}$	Serial I/O1 clock input "L" pulse width (Note)	370			ns
$t_{su(RxD1–SCLK1)}$	Serial I/O1 input set up time	220			ns
$t_{h(SCLK1–RxD1)}$	Serial I/O1 input hold time	100			ns
$t_{C(SCLK2)}$	Serial I/O2 clock input cycle time	1000			ns
$t_{WH(SCLK2)}$	Serial I/O2 clock input "H" pulse width	400			ns
$t_{WL(SCLK2)}$	Serial I/O2 clock input "L" pulse width	400			ns
$t_{su(SDATA2–SCLK2)}$	Serial I/O2 input set up time	200			ns
$t_{h(SCLK2–SDATA2)}$	Serial I/O2 input hold time	200			ns

Note: In this time, bit 6 of the serial I/O1 control register (address $001A_{16}$) is set to "1" (clock synchronous serial I/O1 is selected).
When bit 6 of the serial I/O1 control register is "0" (clock asynchronous serial I/O1 is selected), the rating values are divided by 4.

Table 3.1.20 Timing requirements (2)
(V_{CC} = 2.4 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{W(\overline{RESET})}$	Reset input "L" pulse width	2			µs
$t_{C(XIN)}$	External clock input cycle time	250			ns
$t_{WH(XIN)}$	External clock input "H" pulse width	100			ns
$t_{WL(XIN)}$	External clock input "L" pulse width	100			ns
$t_{C(CNTR0)}$	CNTR0 input cycle time	500			ns
$t_{WH(CNTR0)}$	CNTR0, INT0, INT1, input "H" pulse width	230			ns
$t_{WL(CNTR0)}$	CNTR0, INT0, INT1, input "L" pulse width	230			ns
$t_{C(CNTR1)}$	CNTR1 input cycle time	4000			ns
$t_{WH(CNTR1)}$	CNTR1 input "H" pulse width	1600			ns
$t_{WL(CNTR1)}$	CNTR1 input "L" pulse width	1600			ns
$t_{C(SCLK1)}$	Serial I/O1 clock input cycle time (Note)	2000			ns
$t_{WH(SCLK1)}$	Serial I/O1 clock input "H" pulse width (Note)	950			ns
$t_{WL(SCLK1)}$	Serial I/O1 clock input "L" pulse width (Note)	950			ns
$t_{su(RxD1–SCLK1)}$	Serial I/O1 input set up time	400			ns
$t_{h(SCLK1–RxD1)}$	Serial I/O1 input hold time	200			ns
$t_{C(SCLK2)}$	Serial I/O2 clock input cycle time	2000			ns
$t_{WH(SCLK2)}$	Serial I/O2 clock input "H" pulse width	950			ns
$t_{WL(SCLK2)}$	Serial I/O2 clock input "L" pulse width	950			ns
$t_{su(SDATA2–SCLK2)}$	Serial I/O2 input set up time	400			ns
$t_{h(SCLK2–SDATA2)}$	Serial I/O2 input hold time	400			ns

Note: In this time, bit 6 of the serial I/O1 control register (address $001A_{16}$) is set to "1" (clock synchronous serial I/O1 is selected).
When bit 6 of the serial I/O1 control register is "0" (clock asynchronous serial I/O1 is selected), the rating values are divided by 4.

(6) Switching Characteristics (Extended operating temperature version)

Table 3.1.21 Switching characteristics (1)
(V_{CC} = 4.0 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
t_{WH}(SCLK1)	Serial I/O1 clock output “H” pulse width	t_C(SCLK1)/2–30			ns
t_{WL}(SCLK1)	Serial I/O1 clock output “L” pulse width	t_C(SCLK1)/2–30			ns
t_d(SCLK1–TxD1)	Serial I/O1 output delay time			140	ns
t_v(SCLK1–TxD1)	Serial I/O1 output valid time	–30			ns
t_r(SCLK1)	Serial I/O1 clock output rising time			30	ns
t_f(SCLK1)	Serial I/O1 clock output falling time			30	ns
t_{WH}(SCLK2)	Serial I/O2 clock output “H” pulse width	t_C(SCLK2)/2–30			ns
t_{WL}(SCLK2)	Serial I/O2 clock output “L” pulse width	t_C(SCLK2)/2–30			ns
t_d(SCLK2–SDATA2)	Serial I/O2 output delay time			140	ns
t_v(SCLK2–SDATA2)	Serial I/O2 output valid time	0			ns
t_r(SCLK2)	Serial I/O2 clock output rising time			30	ns
t_f(SCLK2)	Serial I/O2 clock output falling time			30	ns
t_r(CMOS)	CMOS output rising time (Note 1)		10	30	ns
t_f(CMOS)	CMOS output falling time (Note 1)		10	30	ns

Note 1: Pin XOUT is excluded.

Table 3.1.22 Switching characteristics (2)
(V_{CC} = 2.4 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
t_{WH}(SCLK1)	Serial I/O1 clock output “H” pulse width	t_C(SCLK1)/2–50			ns
t_{WL}(SCLK1)	Serial I/O1 clock output “L” pulse width	t_C(SCLK1)/2–50			ns
t_d(SCLK1–TxD1)	Serial I/O1 output delay time			350	ns
t_v(SCLK1–TxD1)	Serial I/O1 output valid time	–30			ns
t_r(SCLK1)	Serial I/O1 clock output rising time			50	ns
t_f(SCLK1)	Serial I/O1 clock output falling time			50	ns
t_{WH}(SCLK2)	Serial I/O2 clock output “H” pulse width	t_C(SCLK2)/2–50			ns
t_{WL}(SCLK2)	Serial I/O2 clock output “L” pulse width	t_C(SCLK2)/2–50			ns
t_d(SCLK2–SDATA2)	Serial I/O2 output delay time			350	ns
t_v(SCLK2–SDATA2)	Serial I/O2 output valid time	0			ns
t_r(SCLK2)	Serial I/O2 clock output rising time			50	ns
t_f(SCLK2)	Serial I/O2 clock output falling time			50	ns
t_r(CMOS)	CMOS output rising time (Note 1)		20	50	ns
t_f(CMOS)	CMOS output falling time (Note 1)		20	50	ns

Note 1: Pin XOUT is excluded.

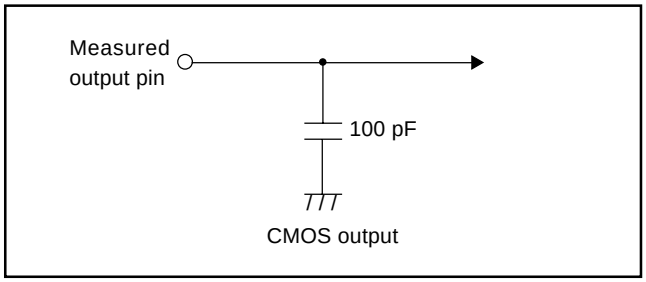


Fig. 3.1.3 Switching characteristics measurement circuit diagram (Extended operating temperature)

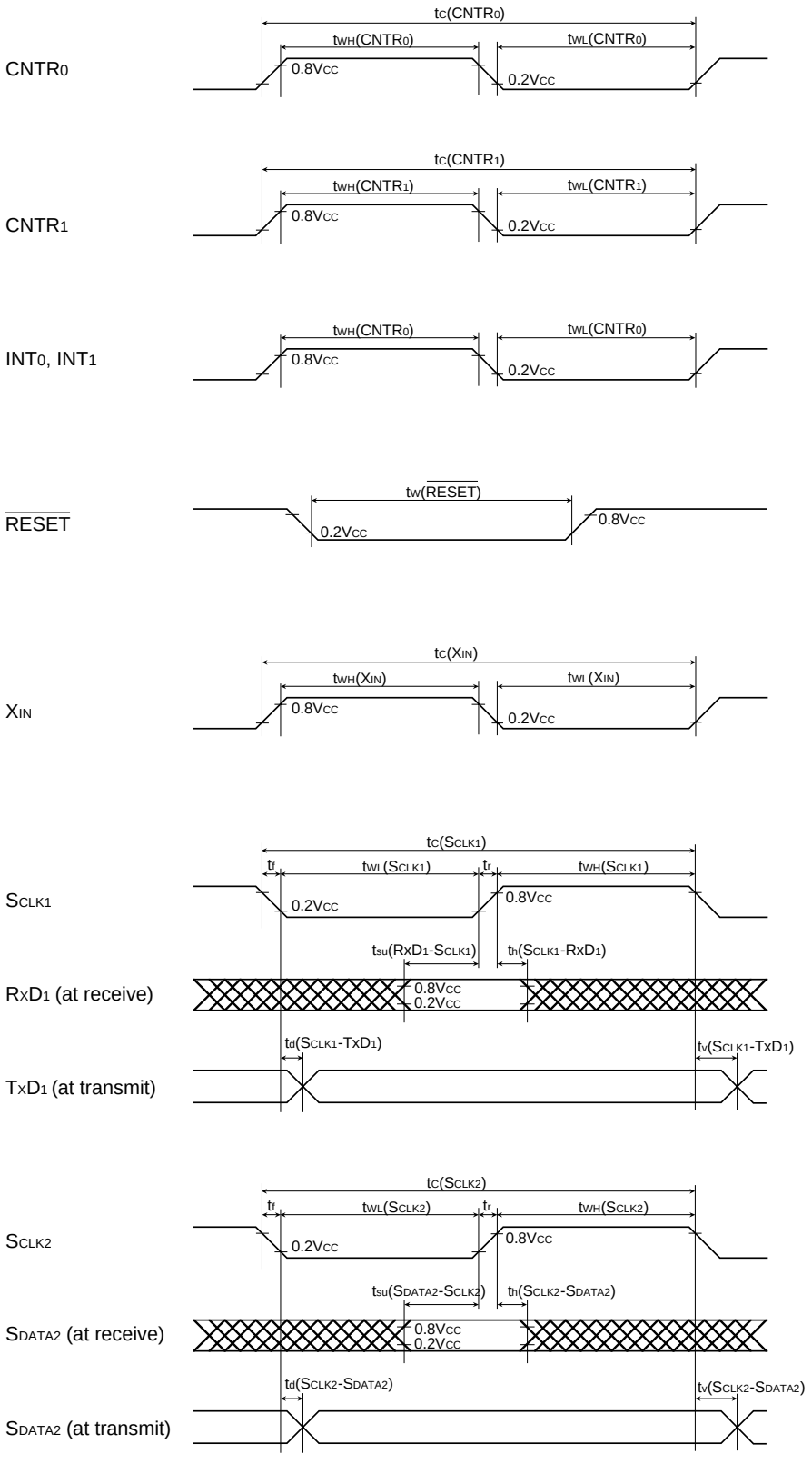


Fig. 3.1.4 Timing chart (Extended operating temperature version)

3.1.3 7540Group (Extended operating temperature 125 °C version)

Applied to: M37540M2V-XXXFP/GP, M37540M4V-XXXFP/GP, M37540E8V-XXXFP/GP

(1) Absolute Maximum Ratings (Extended operating temperature 125 °C version)

Table 3.1.23 Absolute maximum ratings

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Power source voltage	All voltages are based on V_{SS}. Output transistors are cut off.	–0.3 to 6.5 (**Note 1**)	V
V_I	Input voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, V_{REF}		–0.3 to V_{CC} + 0.3	V
V_I	Input voltage $\overline{RESET}$, X_{IN}, CNV_{SS}		–0.3 to V_{CC} + 0.3	V
V_O	Output voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, X_{OUT}		–0.3 to V_{CC} + 0.3	V
P_d	Power dissipation	Ta = 25°C	300 (**Note 2**)	mW
T_{opr}	Operating temperature		–40 to 125 (**Note 3**)	°C
T_{stg}	Storage temperature		–65 to 150	°C

Notes 1: This is the rating value for the Mask ROM version.
The rating value for the One Time PROM version is –0.3 to 7.0 V.
2: 200 mW for the 32P6U package product.
3: In this version, the operating temperature range and total time are limited as follows;
55 °C to 85 °C: within total 6000 hours,
85 °C to 125 °C: within total 1000 hours.

(2) Recommended Operating Conditions (Extended operating temperature 125 °C version)

Table 3.1.24 Recommended operating conditions (1)
(V_{CC} = 2.4 to 5.5 V, Ta = –40 to 125 °C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
V_{CC}	Power source voltage (ceramic)	f(X_{IN}) = 8 MHz (High-, Middle-speed mode)	4.0	5.0	5.5	V
		f(X_{IN}) = 4 MHz (High-, Middle-speed mode)	2.4	5.0	5.5	V
		f(X_{IN}) = 4 MHz (Double-speed mode)	4.0	5.0	5.5	V
		f(X_{IN}) = 2 MHz (Double-speed mode)	2.4	5.0	5.5	V
	Power source voltage (RC)	f(X_{IN}) = 4 MHz (High-, Middle-speed mode)	4.0	5.0	5.5	V
		f(X_{IN}) = 2 MHz (High-, Middle-speed mode)	2.4	5.0	5.5	V
V_{SS}	Power source voltage			0		V
V_{REF}	Analog reference voltage		2.0		V_{CC}	V
V_{IH}	"H" input voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$		0.8V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage (TTL input level selected) $P1_0$, $P1_2$, $P1_3$, $P3_6$, $P3_7$ (Note 1)		2.0		V_{CC}	V
V_{IH}	"H" input voltage $\overline{RESET}$, X_{IN}		0.8V_{CC}		V_{CC}	V
V_{IL}	"L" input voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$		0		0.3V_{CC}	V
V_{IL}	"L" input voltage (TTL input level selected) $P1_0$, $P1_2$, $P1_3$, $P3_6$, $P3_7$ (Note 1)		0		0.8	V
V_{IL}	"L" input voltage $\overline{RESET}$, CNV_{SS}		0		0.2V_{CC}	V
V_{IL}	"L" input voltage X_{IN}		0		0.16V_{CC}	V
$\Sigma I_{OH(peak)}$	"H" total peak output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$				–80	mA
$\Sigma I_{OL(peak)}$	"L" total peak output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$				80	mA
$\Sigma I_{OL(peak)}$	"L" total peak output current (Note 2) $P3_0$–$P3_6$				60	mA
$\Sigma I_{OH(avg)}$	"H" total average output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$				–40	mA
$\Sigma I_{OL(avg)}$	"L" total average output current (Note 2) $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$				40	mA
$\Sigma I_{OL(avg)}$	"L" total average output current (Note 2) $P3_0$–$P3_6$				30	mA

Note 1: Vcc = 4.0 to 5.5V

2: The total output current is the sum of all the currents flowing through all the applicable ports. The total average current is an average value measured over 100 ms. The total peak current is the peak value of all the currents.

APPENDIX

3.1 Electrical characteristics

Table 3.1.25 Recommended operating conditions (2)
(V_{CC} = 2.4 to 5.5 V, Ta = –40 to 125 °C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
$I_{OH(peak)}$	"H" peak output current (Note 1)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$			–10	mA
$I_{OL(peak)}$	"L" peak output current (Note 1)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$			10	mA
$I_{OL(peak)}$	"L" peak output current (Note 1)	$P3_0$–$P3_6$			30	mA
$I_{OH(avg)}$	"H" average output current (Note 2)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$			–5	mA
$I_{OL(avg)}$	"L" average output current (Note 2)	$P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$			5	mA
$I_{OL(avg)}$	"L" average output current (Note 2)	$P3_0$–$P3_6$			15	mA
$f(X_{IN})$	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 4.0 to 5.5 V Double-speed mode			4	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 2.4 to 5.5 V Double-speed mode			2	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 4.0 to 5.5 V High-, Middle-speed mode			8	MHz
	Internal clock oscillation frequency (Note 3) at ceramic oscillation or external clock input	V_{CC} = 2.4 to 5.5 V High-, Middle-speed mode			4	MHz
	Internal clock oscillation frequency (Note 3) at RC oscillation	V_{CC} = 4.0 to 5.5 V High-, Middle-speed mode			4	MHz
	Internal clock oscillation frequency (Note 3) at RC oscillation	V_{CC} = 2.4 to 5.5 V High-, Middle-speed mode			2	MHz

Notes 1: The peak output current is the peak current flowing in each port.
2: The average output current I_{OL} (avg), I_{OH} (avg) in an average value measured over 100 ms.
3: When the oscillation frequency has a duty cycle of 50 %.

(3) Electrical Characteristics (Extended operating temperature 125 °C version)

Table 3.1.26 Electrical characteristics (1)
(V_{CC} = 2.4 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 125 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
V_{OH}	"H" output voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$ (Note 1)	I_{OH} = –5 mA V_{CC} = 4.0 to 5.5 V	V_{CC}–1.5			V
		I_{OH} = –1.0 mA V_{CC} = 2.4 to 5.5 V	V_{CC}–1.0			V
V_{OL}	"L" output voltage $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_7$	I_{OL} = 5 mA V_{CC} = 4.0 to 5.5 V			1.5	V
		I_{OL} = 1.5 mA V_{CC} = 4.0 to 5.5 V			0.3	V
		I_{OL} = 1.0 mA V_{CC} = 2.4 to 5.5 V			1.0	V
V_{OL}	"L" output voltage $P3_0$–$P3_6$	I_{OL} = 15 mA V_{CC} = 4.0 to 5.5 V			2.0	V
		I_{OL} = 1.5 mA V_{CC} = 4.0 to 5.5 V			0.3	V
		I_{OL} = 10 mA V_{CC} = 2.4 to 5.5 V			1.0	V
V_{T+}–V_{T-}	Hysteresis $CNTR_0$, $CNTR_1$, INT_0, INT_1(Note 2) $P0_0$–$P0_7$ (Note 3)			0.4		V
V_{T+}–V_{T-}	Hysteresis RxD, S_{CLK1}, S_{CLK2}, S_{DATA2} (Note 2)			0.5		V
V_{T+}–V_{T-}	Hysteresis $\overline{RESET}$			0.5		V
I_{IH}	"H" input current $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$	V_I = V_{CC} (Pin floating. Pull up transistors "off")			5.0	µA
I_{IH}	"H" input current $\overline{RESET}$	V_I = V_{CC}			5.0	µA
I_{IH}	"H" input current X_{IN}	V_I = V_{CC}		4.0		µA
I_{IL}	"L" input current $P0_0$–$P0_7$, $P1_0$–$P1_4$, $P2_0$–$P2_7$, $P3_0$–$P3_7$	V_I = V_{SS} (Pin floating. Pull up transistors "off")			–5.0	µA
I_{IL}	"L" input current $\overline{RESET}$, CNV_{SS}	V_I = V_{SS}			–5.0	µA
I_{IL}	"L" input current X_{IN}	V_I = V_{SS}		–4.0		µA
I_{IL}	"L" input current $P0_0$–$P0_7$, $P3_0$–$P3_7$	V_I = V_{SS} (Pull up transistors "on")		–0.2	–0.5	mA
V_{RAM}	RAM hold voltage	When clock stopped	2.0		5.5	V
R_{OSC}	Ring oscillator oscillation frequency	V_{CC} = 5.0 V, Ta = 25 °C	1000	2000	3000	kHz
D_{OSC}	Oscillation stop detection circuit detection frequency	V_{CC} = 5.0 V, Ta = 25 °C	62.5	125	187.5	kHz

Notes 1: $P1_1$ is measured when the $P1_1$/T_xD_1 P-channel output disable bit of the UART control register (bit 4 of address $001B_{16}$) is "0".
2: R_xD1, S_{CLK1}, S_{CLK2}, S_{DATA2}, INT_0, and INT_1 have hysteresises only when bits 0 to 2 of the port P1P3 control register are set to "0" (CMOS level).
3: It is available only when operating key-on wake up.

Table 3.1.27 Electrical characteristics (2)
(V_{CC} = 2.4 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 125 °C, unless otherwise noted)

Symbol	Test conditions			Limits			Unit
				Min.	Typ.	Max.	
I_{CC}	One Time PROM version	High-speed mode, f(X_{IN}) = 8 MHz Output transistors “off”			5.0	8.0	mA
		High-speed mode, f(X_{IN}) = 2 MHz, V_{CC} = 2.4 V Output transistors “off”			0.5	1.5	mA
		Middle-speed mode, f(X_{IN}) = 8 MHz, Output transistors “off”			2.0	5.0	mA
		Ring oscillator operation mode, V_{CC} = 5 V Output transistors “off”			350	1000	µA
		f(X_{IN}) = 8 MHz (in WIT state), functions except timer 1 disabled, Output transistors “off”			1.6	3.2	mA
		f(X_{IN}) = 2 MHz, V_{CC} = 2.4 V (in WIT state), functions except timer 1 disabled, Output transistors “off”			0.2		mA
		Ring oscillator operation mode, V_{CC} = 5V (in WIT state), functions except timer 1 disabled, Output transistors “off”			150	450	µA
		Increment when A-D conversion is executed f(X_{IN}) = 8 MHz, V_{CC} = 5 V			0.5		mA
		All oscillation stopped (in STP state) Output transistors “off”	Ta = 25 °C		0.1	1.0	µA
			Ta = 125 °C			50	µA
	Mask ROM version	High-speed mode, f(X_{IN}) = 8 MHz Output transistors “off”			3.5	6.5	mA
		High-speed mode, f(X_{IN}) = 2 MHz, V_{CC} = 2.4 V Output transistors “off”			0.4	1.2	mA
		Middle-speed mode, f(X_{IN}) = 8 MHz, Output transistors “off”			2.0	5.0	mA
		Ring oscillator operation mode, V_{CC} = 5 V Output transistors “off”			300	900	µA
		f(X_{IN}) = 8 MHz (in WIT state), functions except timer 1 disabled, Output transistors “off”			1.6	3.2	mA
		f(X_{IN}) = 2 MHz, V_{CC} = 2.4 V (in WIT state), functions except timer 1 disabled, Output transistors “off”			0.2		mA
		Ring oscillator operation mode, V_{CC} = 5V (in WIT state), functions except timer 1 disabled, Output transistors “off”			150	450	µA
		Increment when A-D conversion is executed f(X_{IN}) = 8 MHz, V_{CC} = 5 V			0.5		mA
		All oscillation stopped (in STP state) Output transistors “off”	Ta = 25 °C		0.1	1.0	µA
			Ta = 125 °C			50	µA

(4) A-D Converter Characteristics (Extended operating temperature 125 °C version)

Table 3.1.28 A-D Converter characteristics
(V_{CC} = 2.7 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 125 °C, unless otherwise noted)

	Symbol	Parameter	Test conditions	Limits			Unit
				Min.	Typ.	Max.	
One Time PROM version	—	Resolution				10	Bits
	—	Linearity error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±3	LSB
	—	Differential nonlinear error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±0.9	LSB
	V_{OT}	Zero transition voltage	V_{CC} = V_{REF} = 5.12 V	0	5	20	mV
			V_{CC} = V_{REF} = 3.072 V	0	3	15	mV
	V_{FST}	Full scale transition voltage	V_{CC} = V_{REF} = 5.12 V	5105	5115	5125	mV
			V_{CC} = V_{REF} = 3.072 V	3060	3069	3075	mV
	t_{CONV}	Conversion time				122	tc(X_{IN})
	R_{LADDER}	Ladder resistor			55		kΩ
	I_{VREF}	Reference power source input current	V_{REF} = 5.0 V	50	150	200	µA
			V_{REF} = 3.0 V	30	70	120	
	$I_{I(AD)}$	A-D port input current				7.0	µA
Mask ROM version	—	Resolution				10	Bits
	—	Linearity error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±3	LSB
	—	Differential nonlinear error	V_{CC} = 2.7 to 5.5 V Ta = 25 °C			±1.5	LSB
	V_{OT}	Zero transition voltage	V_{CC} = V_{REF} = 5.12 V	0	15	35	mV
			V_{CC} = V_{REF} = 3.072 V	0	9	21	mV
	V_{FST}	Full scale transition voltage	V_{CC} = V_{REF} = 5.12 V	5105	5125	5150	mV
			V_{CC} = V_{REF} = 3.072 V	3060	3075	3090	mV
	t_{CONV}	Conversion time				122	tc(X_{IN})
	R_{LADDER}	Ladder resistor			55		kΩ
	I_{VREF}	Reference power source input current	V_{REF} = 5.0 V	50	150	200	µA
			V_{REF} = 3.0 V	30	70	120	
	$I_{I(AD)}$	A-D port input current				7.0	µA

(5) Timing Requirements (Extended operating temperature 125 °C version)

Table 3.1.29 Timing requirements (1)
(V_{CC} = 4.0 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 125 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{W(\overline{RESET})}$	Reset input “L” pulse width	2			µs
$t_{C(XIN)}$	External clock input cycle time	125			ns
$t_{WH(XIN)}$	External clock input “H” pulse width	50			ns
$t_{WL(XIN)}$	External clock input “L” pulse width	50			ns
$t_{C(CNTR0)}$	CNTR0 input cycle time	200			ns
$t_{WH(CNTR0)}$	CNTR0, INT0, INT1, input “H” pulse width	80			ns
$t_{WL(CNTR0)}$	CNTR0, INT0, INT1, input “L” pulse width	80			ns
$t_{C(CNTR1)}$	CNTR1 input cycle time	2000			ns
$t_{WH(CNTR1)}$	CNTR1 input “H” pulse width	800			ns
$t_{WL(CNTR1)}$	CNTR1 input “L” pulse width	800			ns
$t_{C(SCLK1)}$	Serial I/O1 clock input cycle time (Note)	800			ns
$t_{WH(SCLK1)}$	Serial I/O1 clock input “H” pulse width (Note)	370			ns
$t_{WL(SCLK1)}$	Serial I/O1 clock input “L” pulse width (Note)	370			ns
$t_{su(RxD1–SCLK1)}$	Serial I/O1 input set up time	220			ns
$t_{h(SCLK1–RxD1)}$	Serial I/O1 input hold time	100			ns
$t_{C(SCLK2)}$	Serial I/O2 clock input cycle time	1000			ns
$t_{WH(SCLK2)}$	Serial I/O2 clock input “H” pulse width	400			ns
$t_{WL(SCLK2)}$	Serial I/O2 clock input “L” pulse width	400			ns
$t_{su(SDATA2–SCLK2)}$	Serial I/O2 input set up time	200			ns
$t_{h(SCLK2–SDATA2)}$	Serial I/O2 input hold time	200			ns

Note: In this time, bit 6 of the serial I/O1 control register (address $001A_{16}$) is set to “1” (clock synchronous serial I/O1 is selected).
When bit 6 of the serial I/O1 control register is “0” (clock asynchronous serial I/O1 is selected), the rating values are divided by 4.

Table 3.1.30 Timing requirements (2)
(V_{CC} = 2.4 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 125 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{W(\overline{RESET})}$	Reset input “L” pulse width	2			µs
$t_{C(XIN)}$	External clock input cycle time	250			ns
$t_{WH(XIN)}$	External clock input “H” pulse width	100			ns
$t_{WL(XIN)}$	External clock input “L” pulse width	100			ns
$t_{C(CNTR0)}$	CNTR0 input cycle time	500			ns
$t_{WH(CNTR0)}$	CNTR0, INT0, INT1, input “H” pulse width	230			ns
$t_{WL(CNTR0)}$	CNTR0, INT0, INT1, input “L” pulse width	230			ns
$t_{C(CNTR1)}$	CNTR1 input cycle time	4000			ns
$t_{WH(CNTR1)}$	CNTR1 input “H” pulse width	1600			ns
$t_{WL(CNTR1)}$	CNTR1 input “L” pulse width	1600			ns
$t_{C(SCLK1)}$	Serial I/O1 clock input cycle time (Note)	2000			ns
$t_{WH(SCLK1)}$	Serial I/O1 clock input “H” pulse width (Note)	950			ns
$t_{WL(SCLK1)}$	Serial I/O1 clock input “L” pulse width (Note)	950			ns
$t_{su(RxD1–SCLK1)}$	Serial I/O1 input set up time	400			ns
$t_{h(SCLK1–RxD1)}$	Serial I/O1 input hold time	200			ns
$t_{C(SCLK2)}$	Serial I/O2 clock input cycle time	2000			ns
$t_{WH(SCLK2)}$	Serial I/O2 clock input “H” pulse width	950			ns
$t_{WL(SCLK2)}$	Serial I/O2 clock input “L” pulse width	950			ns
$t_{su(SDATA2–SCLK2)}$	Serial I/O2 input set up time	400			ns
$t_{h(SCLK2–SDATA2)}$	Serial I/O2 input hold time	400			ns

Note: In this time, bit 6 of the serial I/O1 control register (address $001A_{16}$) is set to “1” (clock synchronous serial I/O1 is selected).
When bit 6 of the serial I/O1 control register is “0” (clock asynchronous serial I/O1 is selected), the rating values are divided by 4.

(6) Switching Characteristics (Extended operating temperature 125 °C version)

Table 3.1.30 Switching characteristics (1)
(V_{CC} = 4.0 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 125 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
t_{WH}(SCLK1)	Serial I/O1 clock output “H” pulse width	t_C(SCLK1)/2–50			ns
t_{WL}(SCLK1)	Serial I/O1 clock output “L” pulse width	t_C(SCLK1)/2–50			ns
t_d(SCLK1–TxD1)	Serial I/O1 output delay time			140	ns
t_v(SCLK1–TxD1)	Serial I/O1 output valid time	–30			ns
t_r(SCLK1)	Serial I/O1 clock output rising time			30	ns
t_f(SCLK1)	Serial I/O1 clock output falling time			30	ns
t_{WH}(SCLK2)	Serial I/O2 clock output “H” pulse width	t_C(SCLK2)/2–50			ns
t_{WL}(SCLK2)	Serial I/O2 clock output “L” pulse width	t_C(SCLK2)/2–50			ns
t_d(SCLK2–SDATA2)	Serial I/O2 output delay time			140	ns
t_v(SCLK2–SDATA2)	Serial I/O2 output valid time	0			ns
t_r(SCLK2)	Serial I/O2 clock output rising time			30	ns
t_f(SCLK2)	Serial I/O2 clock output falling time			30	ns
t_r(CMOS)	CMOS output rising time (Note 1)		10	30	ns
t_f(CMOS)	CMOS output falling time (Note 1)		10	30	ns

Note 1: Pin XOUT is excluded.

Table 3.1.31 Switching characteristics (2)
(V_{CC} = 2.4 to 5.5 V, V_{SS} = 0 V, Ta = –40 to 125 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
t_{WH}(SCLK1)	Serial I/O1 clock output “H” pulse width	t_C(SCLK1)/2–80			ns
t_{WL}(SCLK1)	Serial I/O1 clock output “L” pulse width	t_C(SCLK1)/2–80			ns
t_d(SCLK1–TxD1)	Serial I/O1 output delay time			350	ns
t_v(SCLK1–TxD1)	Serial I/O1 output valid time	–30			ns
t_r(SCLK1)	Serial I/O1 clock output rising time			50	ns
t_f(SCLK1)	Serial I/O1 clock output falling time			50	ns
t_{WH}(SCLK2)	Serial I/O2 clock output “H” pulse width	t_C(SCLK2)/2–80			ns
t_{WL}(SCLK2)	Serial I/O2 clock output “L” pulse width	t_C(SCLK2)/2–80			ns
t_d(SCLK2–SDATA2)	Serial I/O2 output delay time			350	ns
t_v(SCLK2–SDATA2)	Serial I/O2 output valid time	0			ns
t_r(SCLK2)	Serial I/O2 clock output rising time			50	ns
t_f(SCLK2)	Serial I/O2 clock output falling time			50	ns
t_r(CMOS)	CMOS output rising time (Note 1)		20	50	ns
t_f(CMOS)	CMOS output falling time (Note 1)		20	50	ns

Note 1: Pin XOUT is excluded.

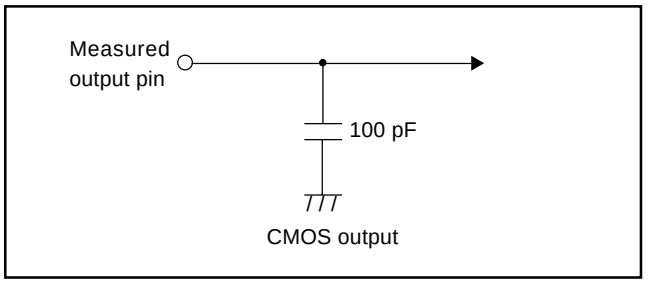


Fig. 3.1.5 Switching characteristics measurement circuit diagram (Extended operating temperature 125 °C version)

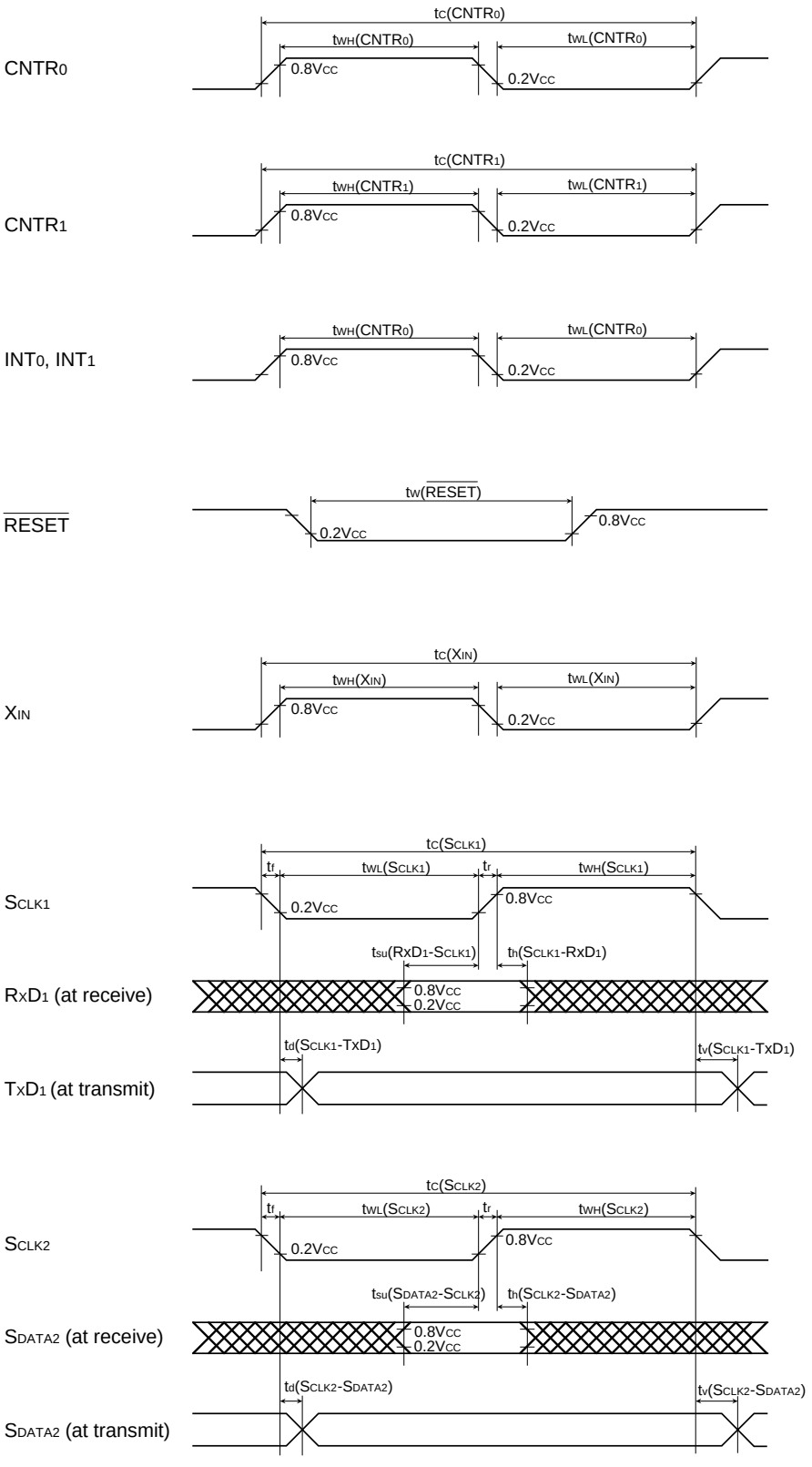


Fig. 3.1.6 Timing chart (Extended operating temperature 125 °C version)

3.2 Typical characteristics

Standard characteristics described below are just examples of the 7540 Group's characteristics and are not guaranteed. For rated values, refer to "3.1 Electrical characteristics".

3.2.1 Mask ROM version

(1) Power source current characteristic example (V_{CC}-I_{CC} characteristics)

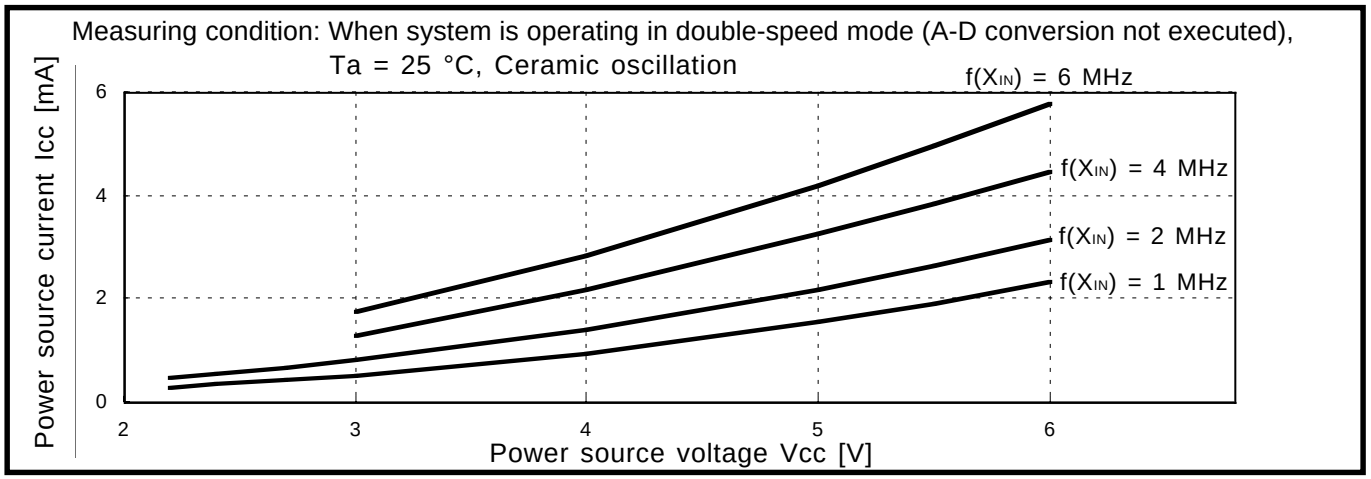


Fig. 3.2.1 V_{CC}-I_{CC} characteristics (in double-speed mode: Mask ROM version)

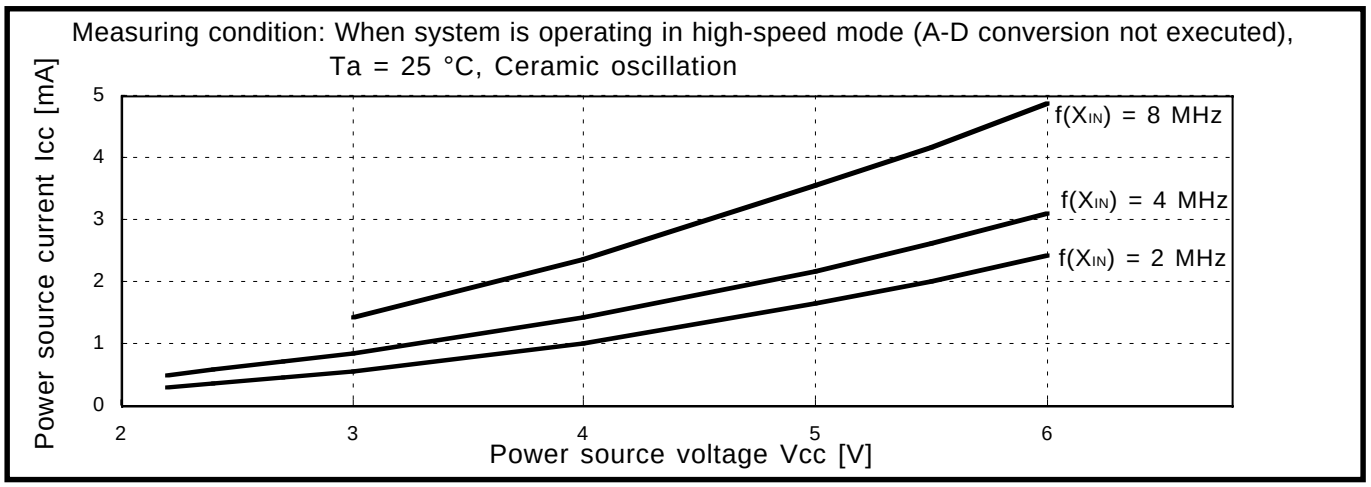


Fig. 3.2.2 V_{CC}-I_{CC} characteristics (in high-speed mode: Mask ROM version)

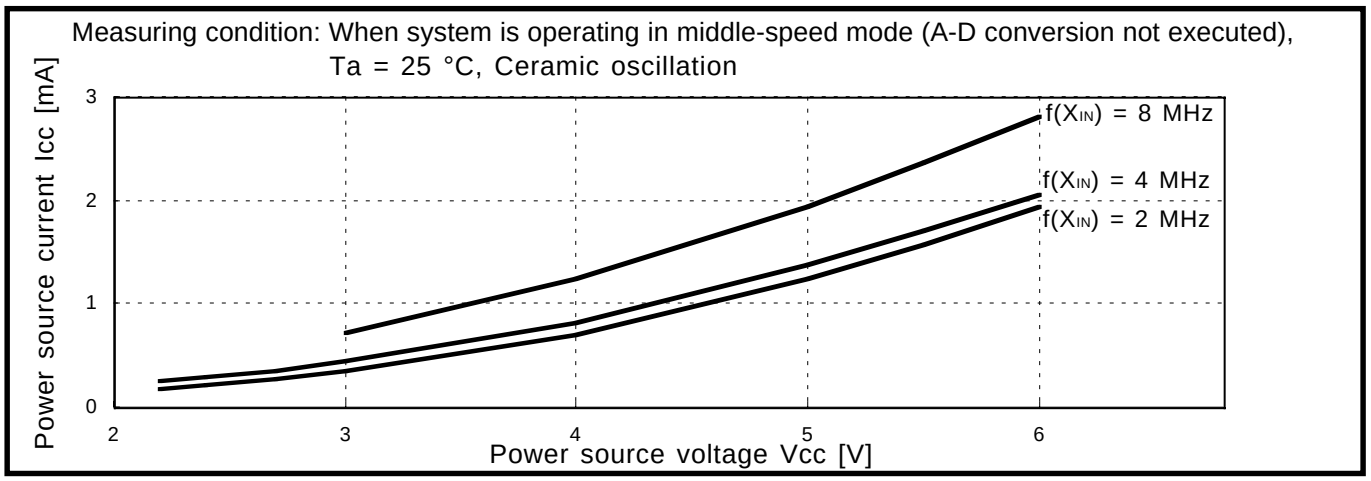


Fig. 3.2.3 V_{CC}-I_{CC} characteristics (in middle-speed mode: Mask ROM version)

APPENDIX

3.2 Typical characteristics

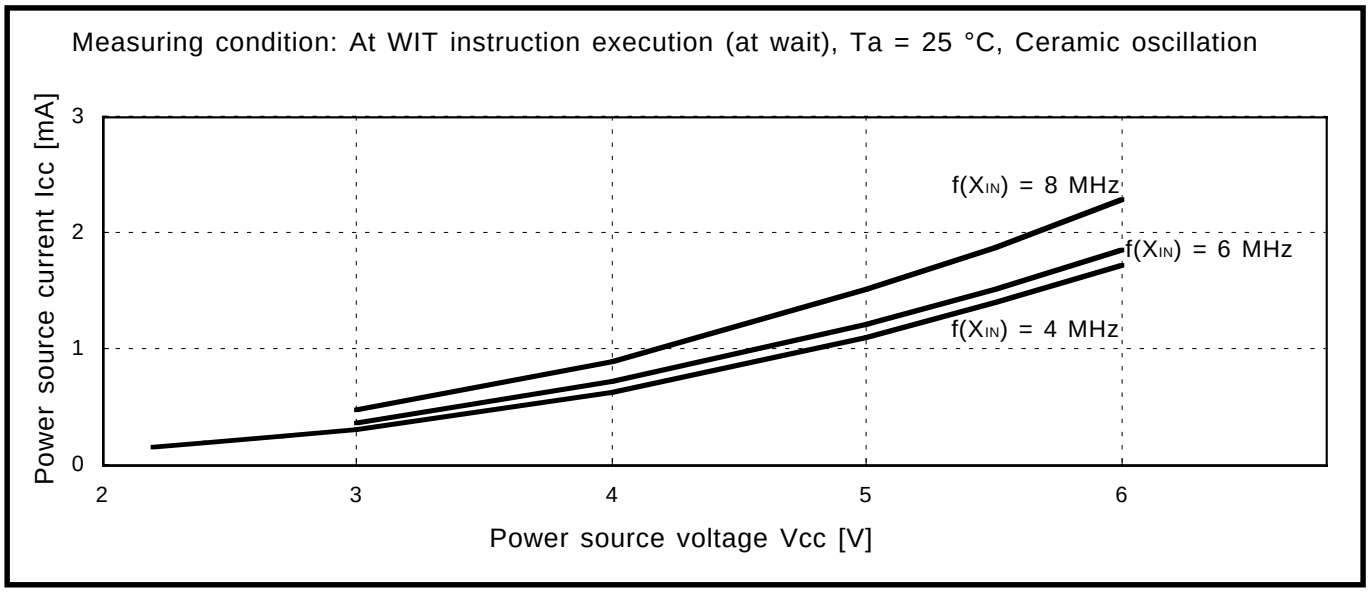


Fig. 3.2.4 V_{CC}-I_{CC} characteristics (at WIT instruction execution: Mask ROM version)

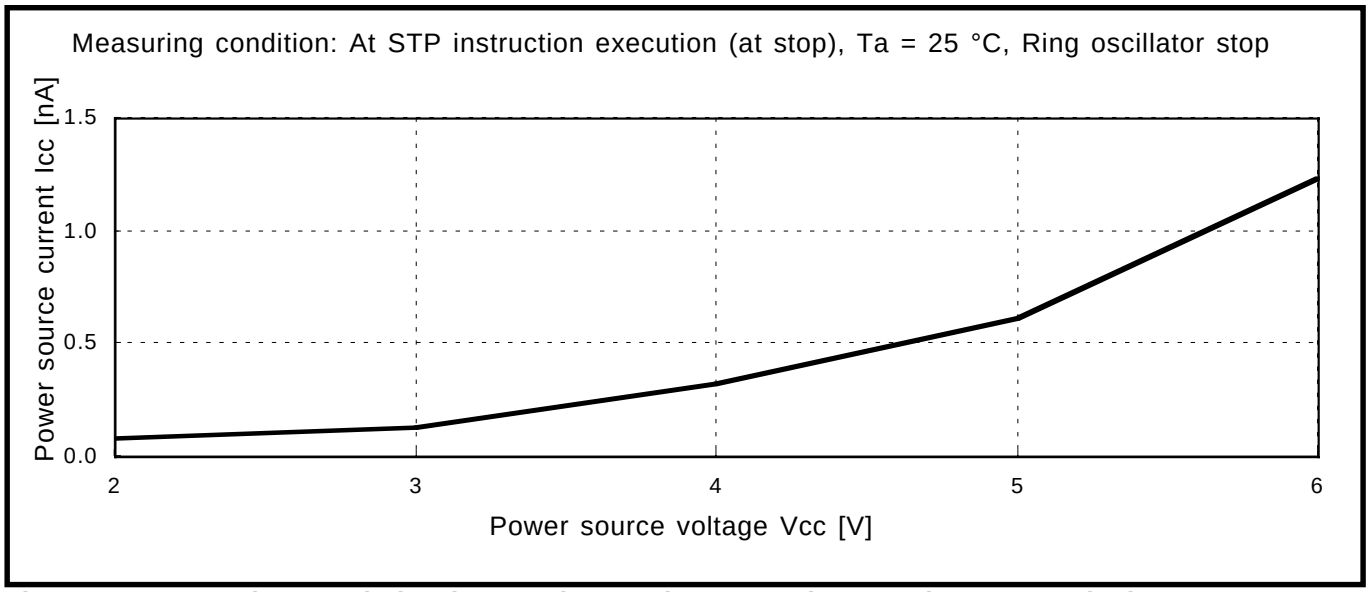


Fig. 3.2.5 V_{CC}-I_{CC} characteristics (at STP instruction execution: Mask ROM version)

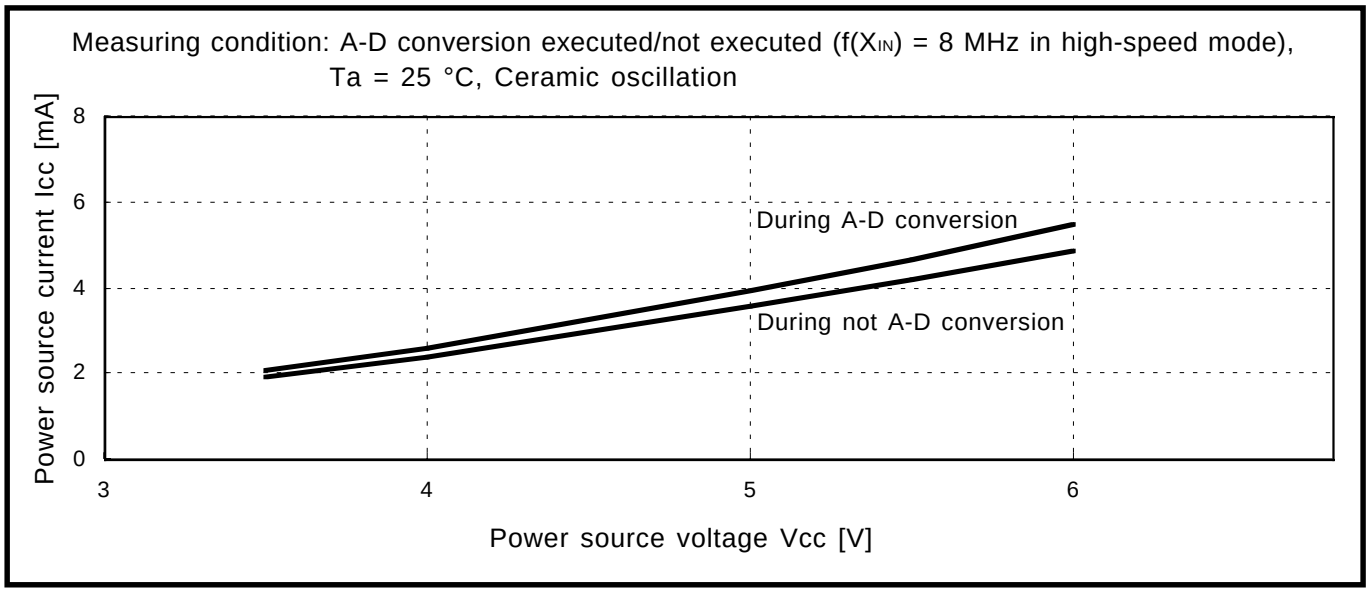


Fig. 3.2.6 V_{CC}-I_{CC} characteristics (addition when operating A-D conversion, f(X_{IN}) = 8 MHz in high-speed mode: Mask ROM version)

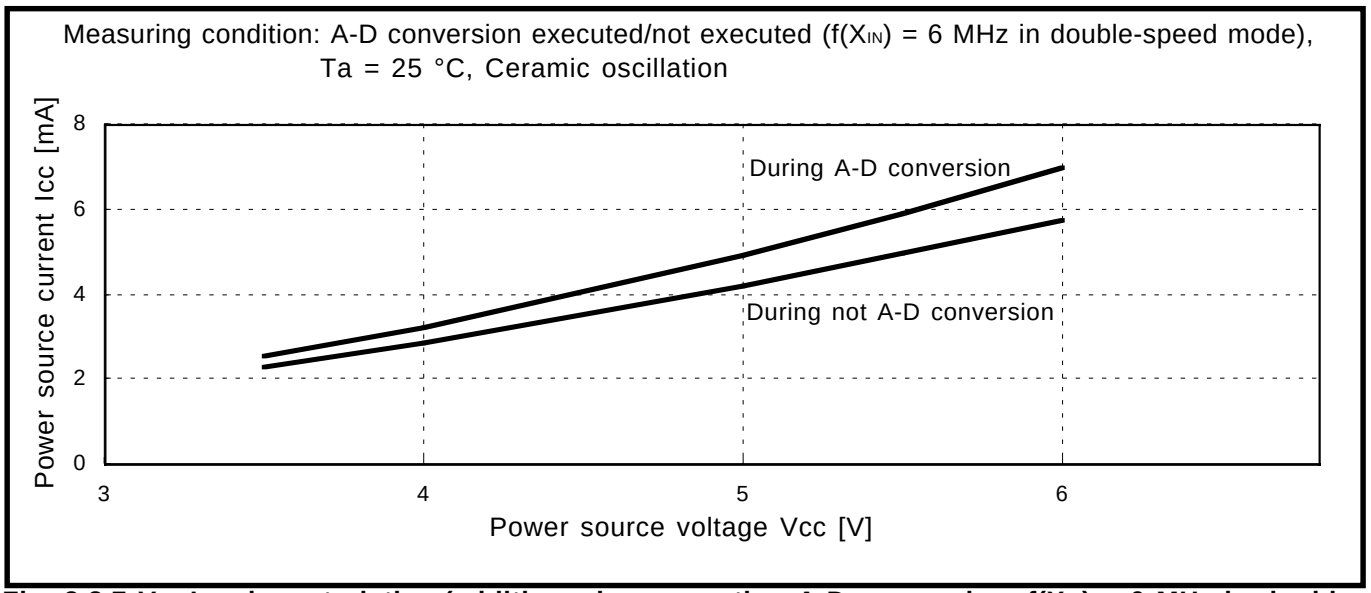


Fig. 3.2.7 V_{CC}-I_{CC} characteristics (addition when operating A-D conversion, f(X_{IN}) = 6 MHz in double-speed mode: Mask ROM version)

APPENDIX

3.2 Typical characteristics

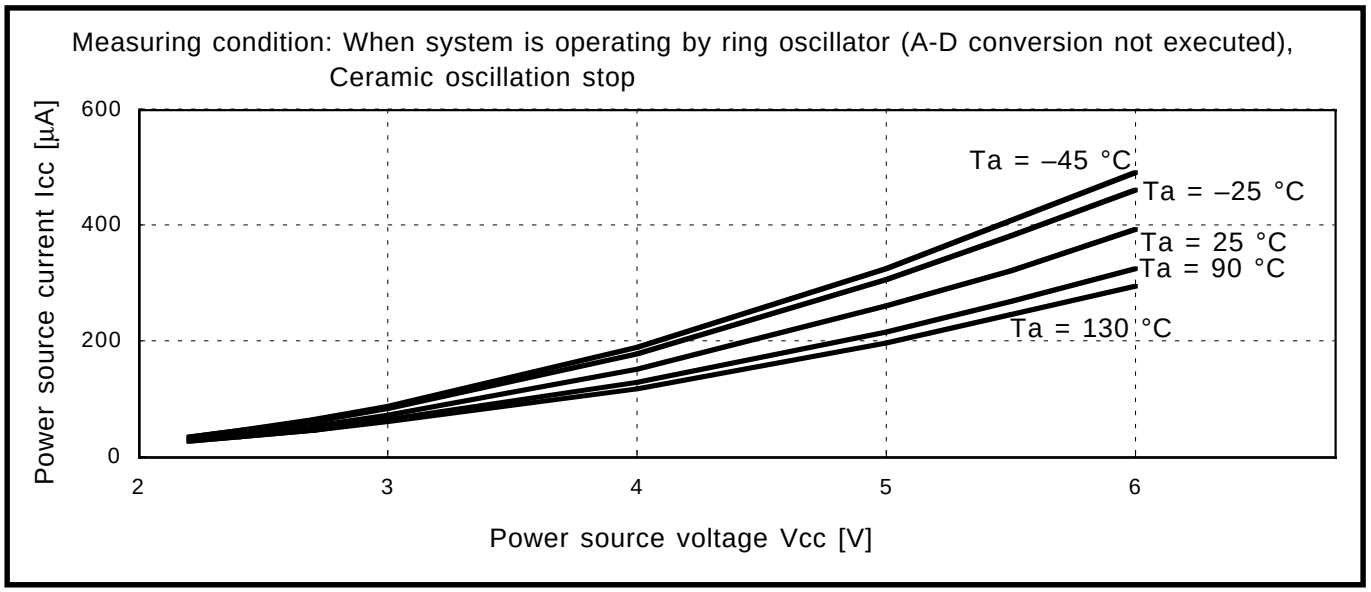


Fig. 3.2.8 V_{CC}-I_{CC} characteristics (When system is operating by ring oscillator, Ceramic oscillation stop: Mask ROM version)

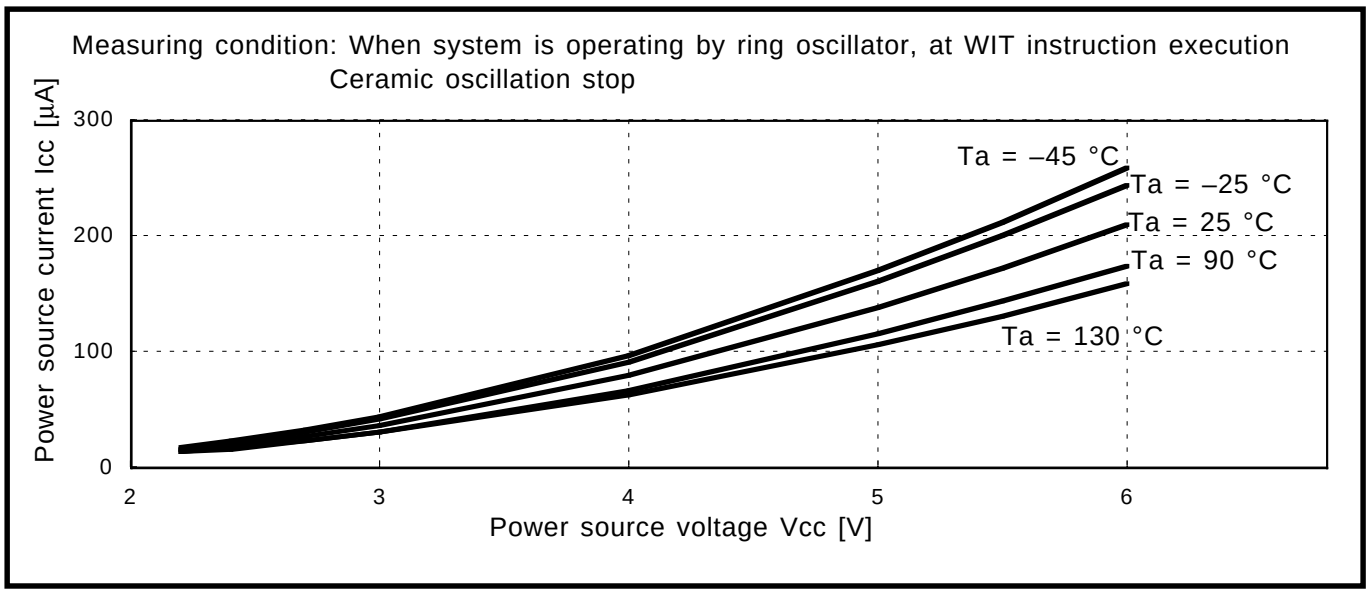


Fig. 3.2.9 V_{CC}-I_{CC} characteristics (When system is operating by ring oscillator, at WIT instruction execution, Ceramic oscillation stop: Mask ROM version)

(2) Power source current characteristic example ($f(X_{IN})$-I_{CC} characteristics)

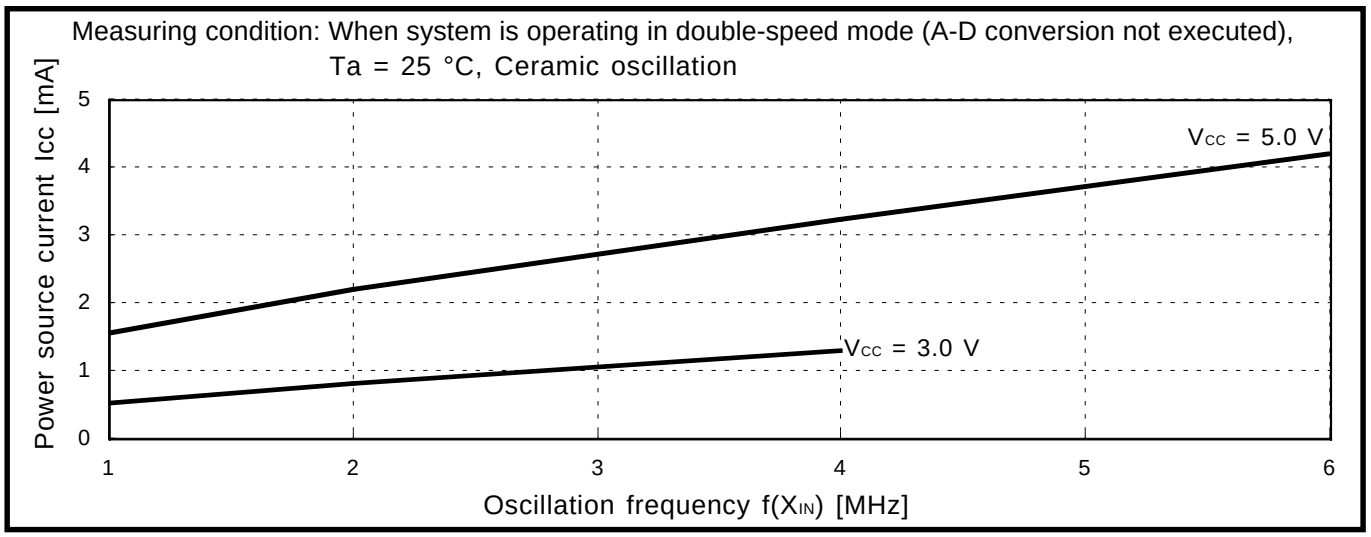


Fig. 3.2.10 $f(X_{IN})$-I_{CC} characteristics (in double-speed mode: Mask ROM version)

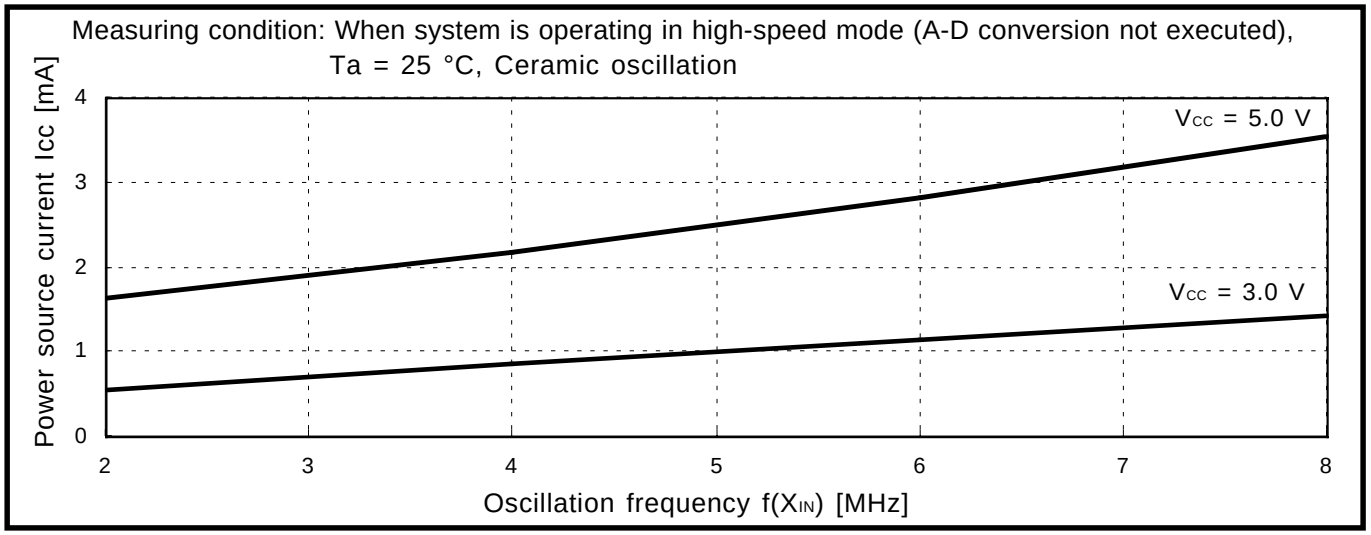


Fig. 3.2.11 $f(X_{IN})$-I_{CC} characteristics (in high-speed mode: Mask ROM version)

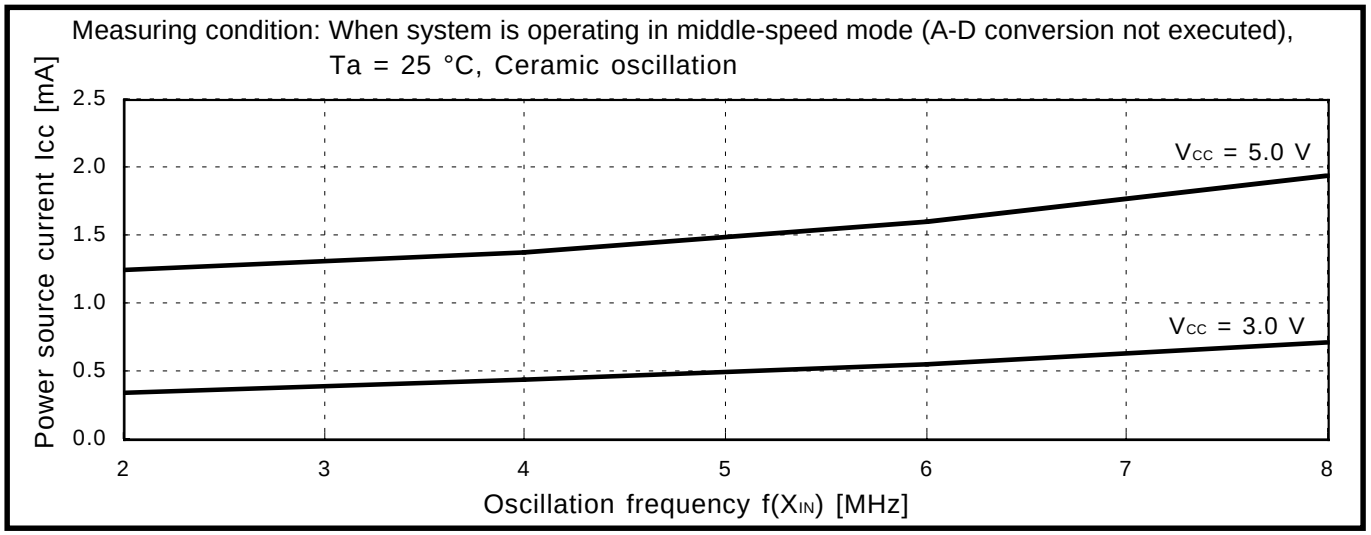


Fig. 3.2.12 $f(X_{IN})$-I_{CC} characteristics (in middle-speed mode: Mask ROM version)

3.2 Typical characteristics

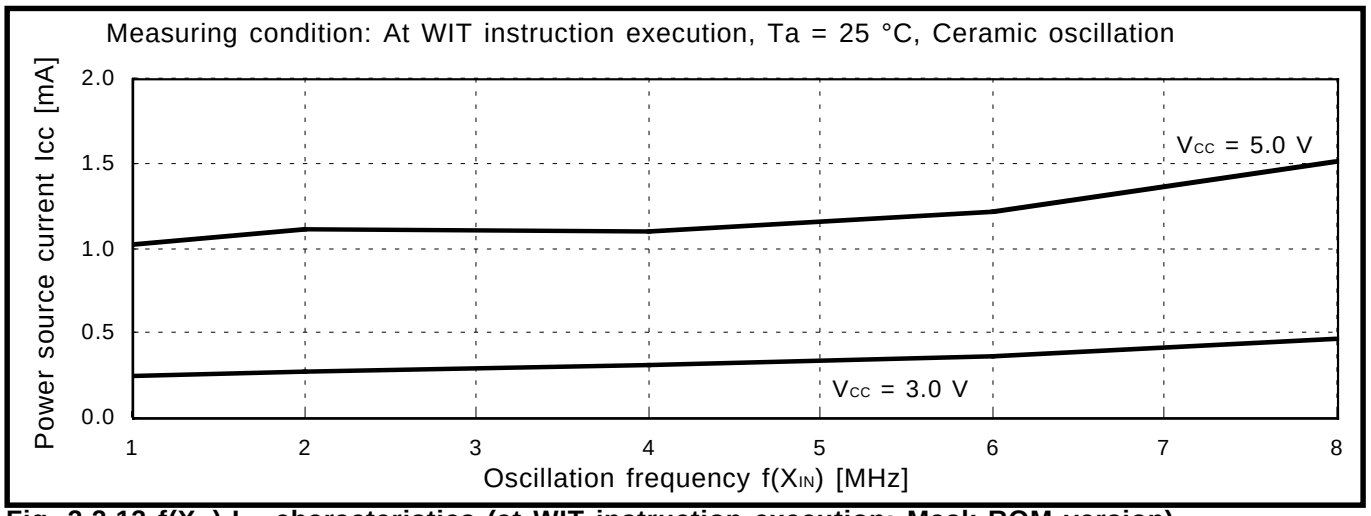


Fig. 3.2.13 f(X_{IN})-I_{CC} characteristics (at WIT instruction execution: Mask ROM version)

(3) Power source current characteristic example (Ta-I_{CC} characteristics)

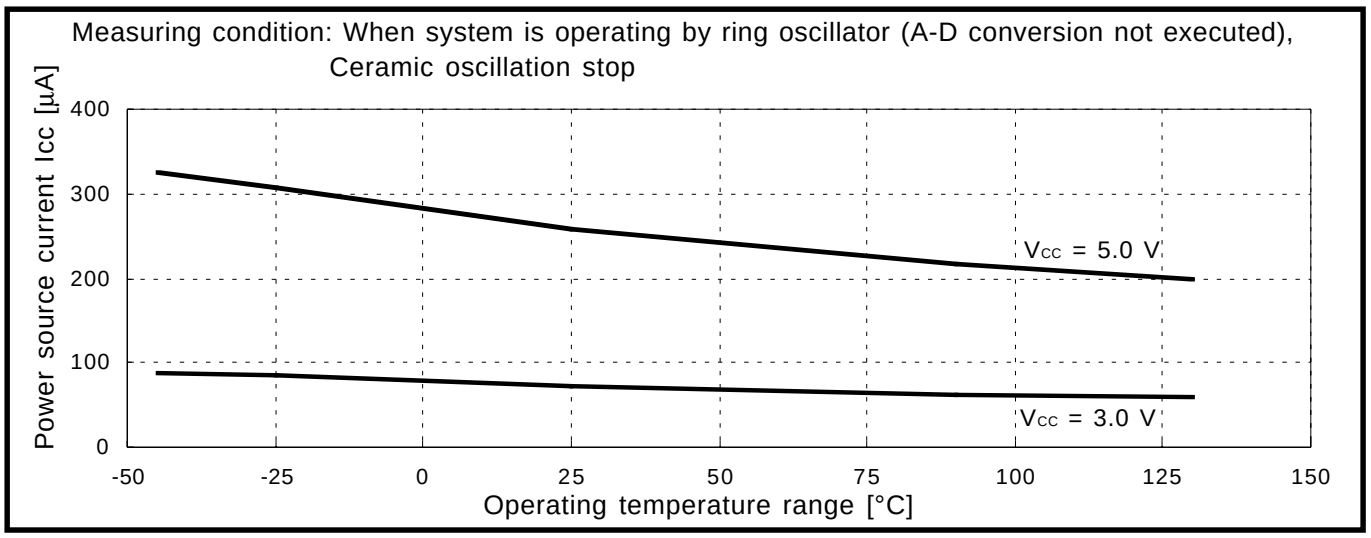


Fig. 3.2.14 Ta-I_{CC} characteristics (When system is operating by ring oscillator, Ceramic oscillation stop: Mask ROM version)

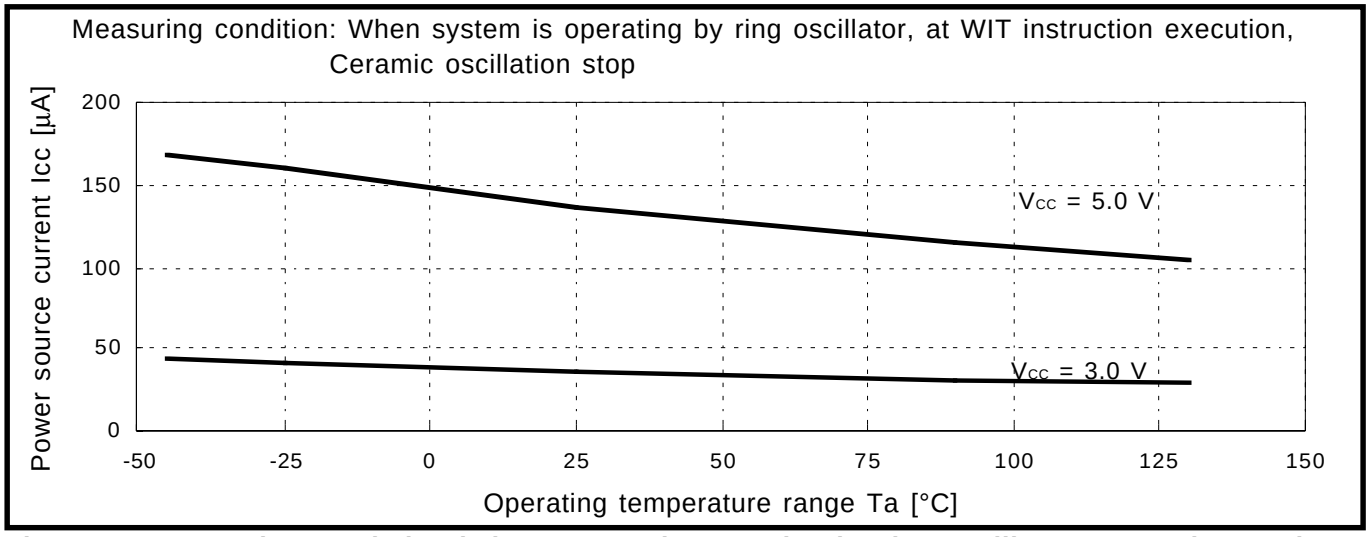


Fig. 3.2.15 Ta-I_{CC} characteristics (When system is operating by ring oscillator, at WIT instruction execution, Ceramic oscillation stop: Mask ROM version)

(4) Port typical characteristic example (V_{CC}-V_{IHL} characteristics)

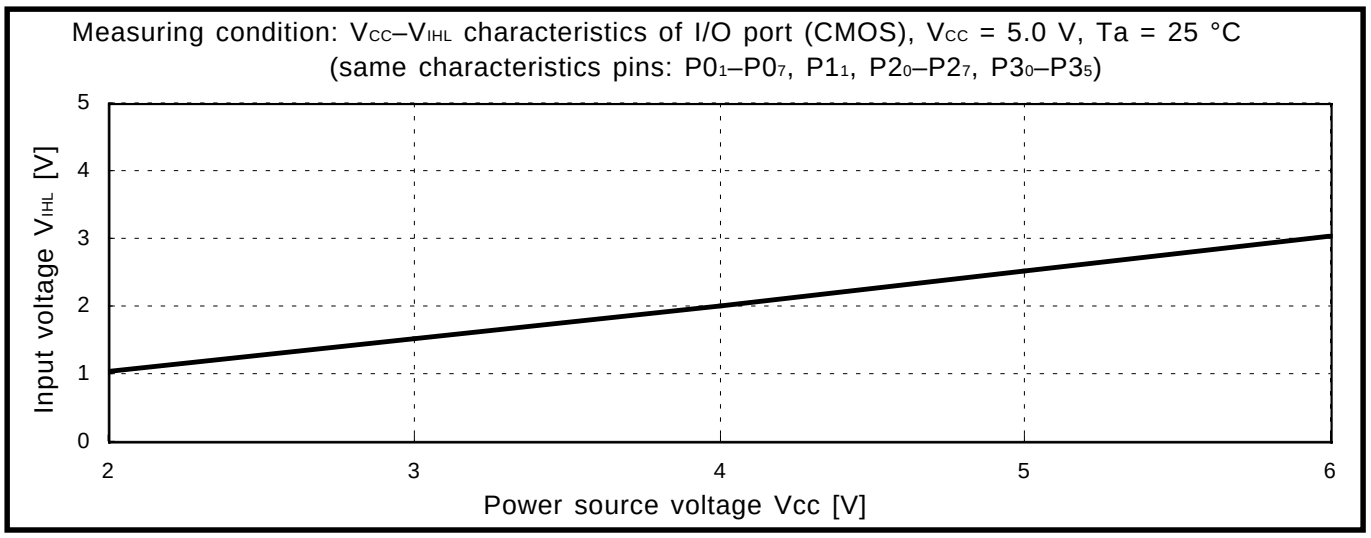


Fig. 3.2.16 V_{CC}-V_{IHL} characteristics (I/O port (CMOS): Mask ROM version)

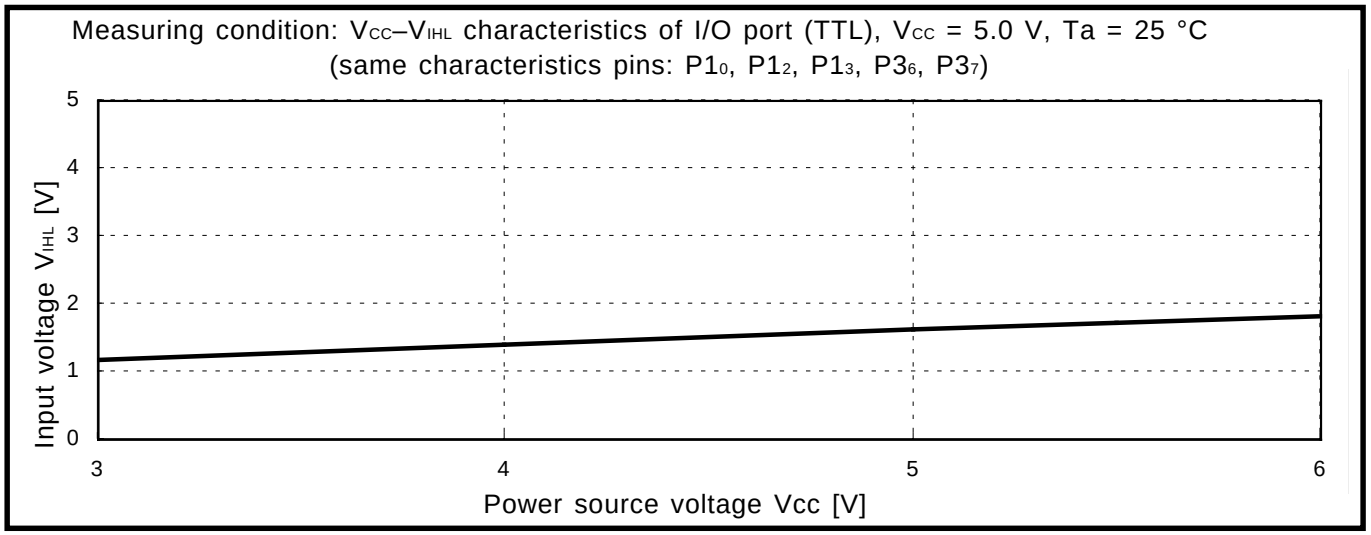


Fig. 3.2.17 V_{CC}-V_{IHL} characteristics (I/O port (TTL): Mask ROM version)

3.2 Typical characteristics

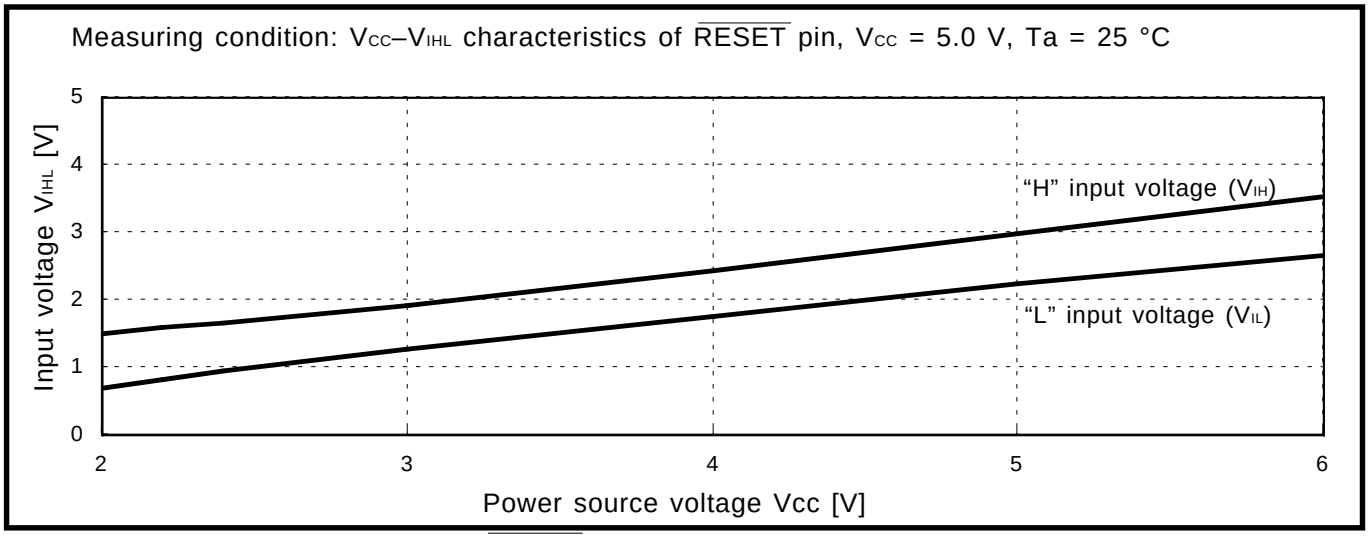


Fig. 3.2.18 V_{CC}-V_{IHL} characteristics ($\overline{\text{RESET}}$ pin: Mask ROM version)

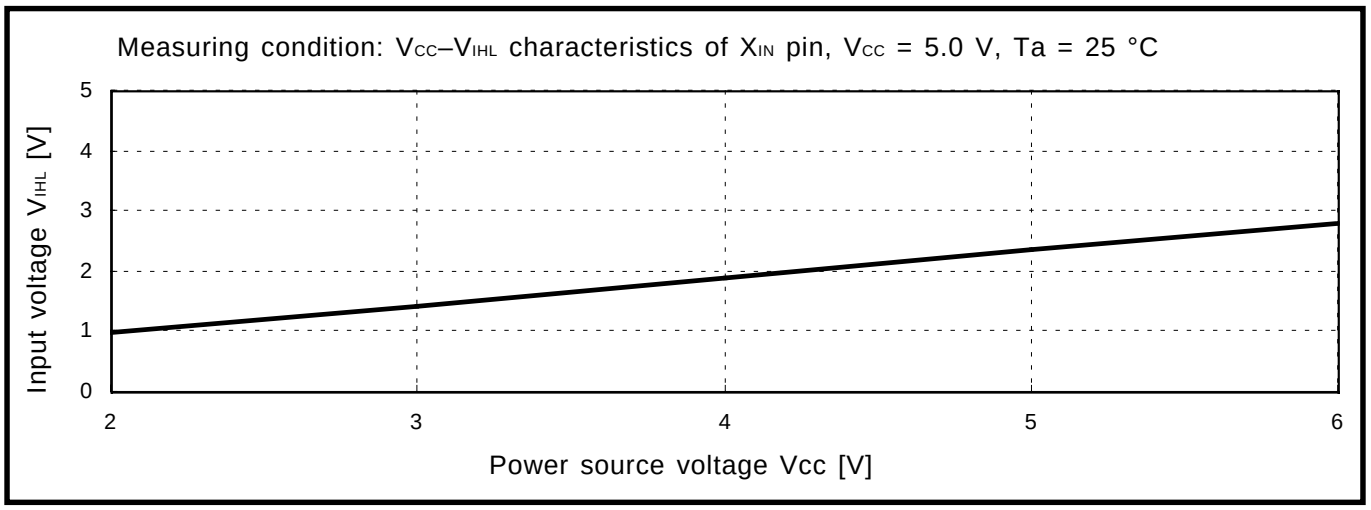


Fig. 3.2.19 V_{CC}-V_{IHL} characteristics (X_{IN} pin: Mask ROM version)

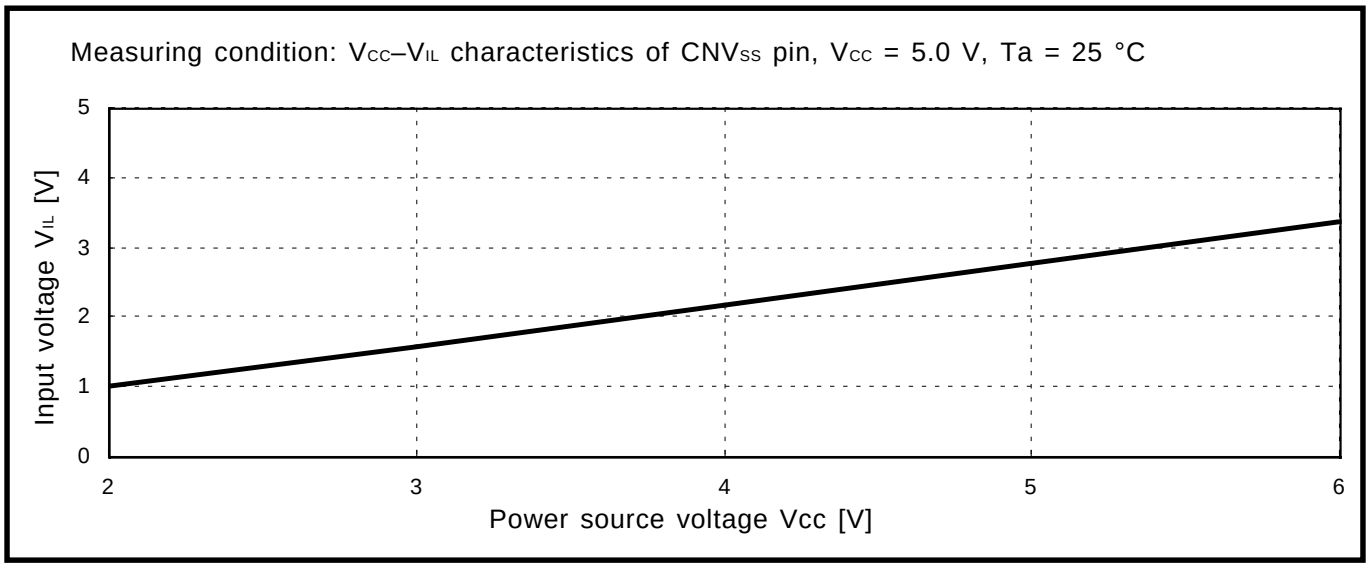


Fig. 3.2.20 V_{CC}-V_{IL} characteristics (CNV_{SS} pin: Mask ROM version)

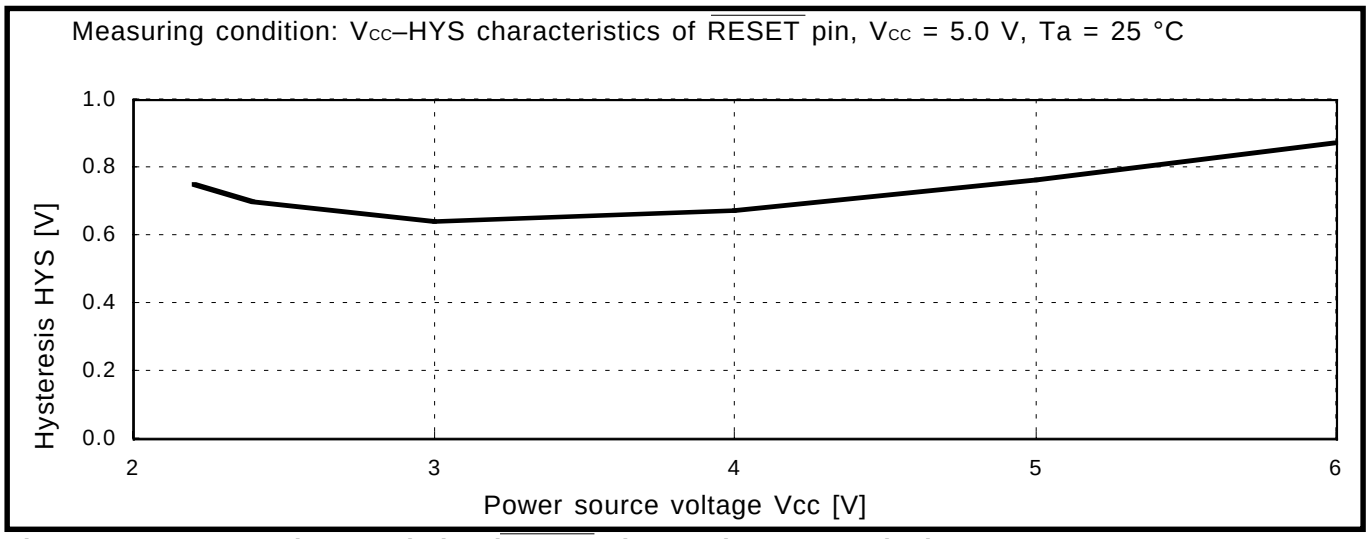


Fig. 3.2.21 Vcc-HYS characteristics (RESET pin: Mask ROM version)

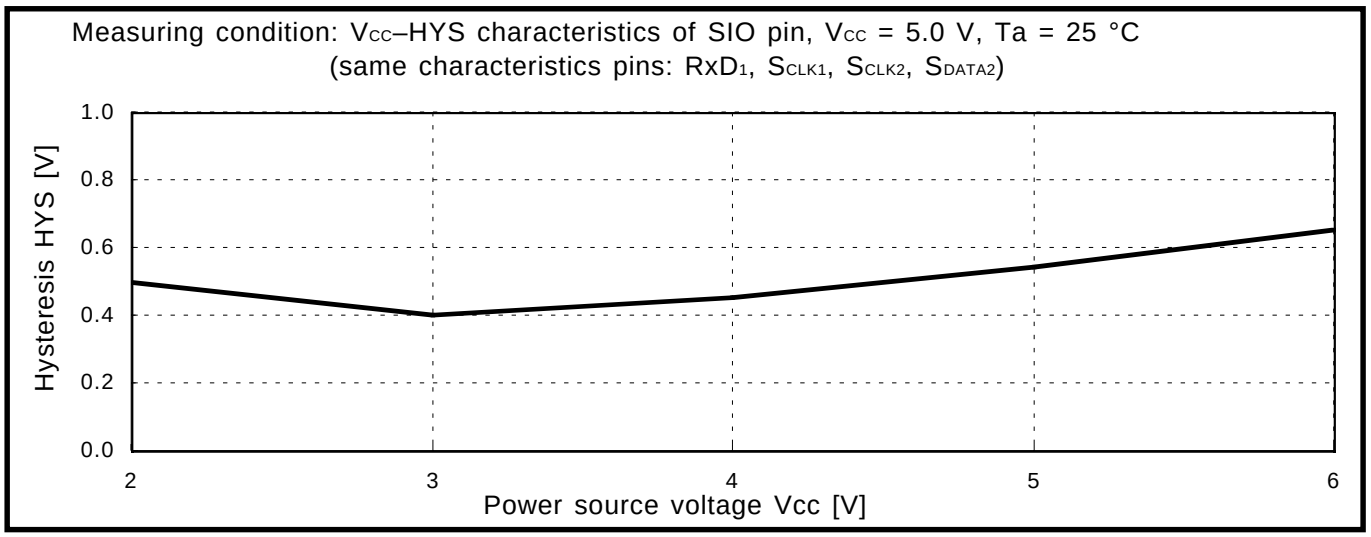


Fig. 3.2.22 Vcc-HYS characteristics (SIO pin: Mask ROM version)

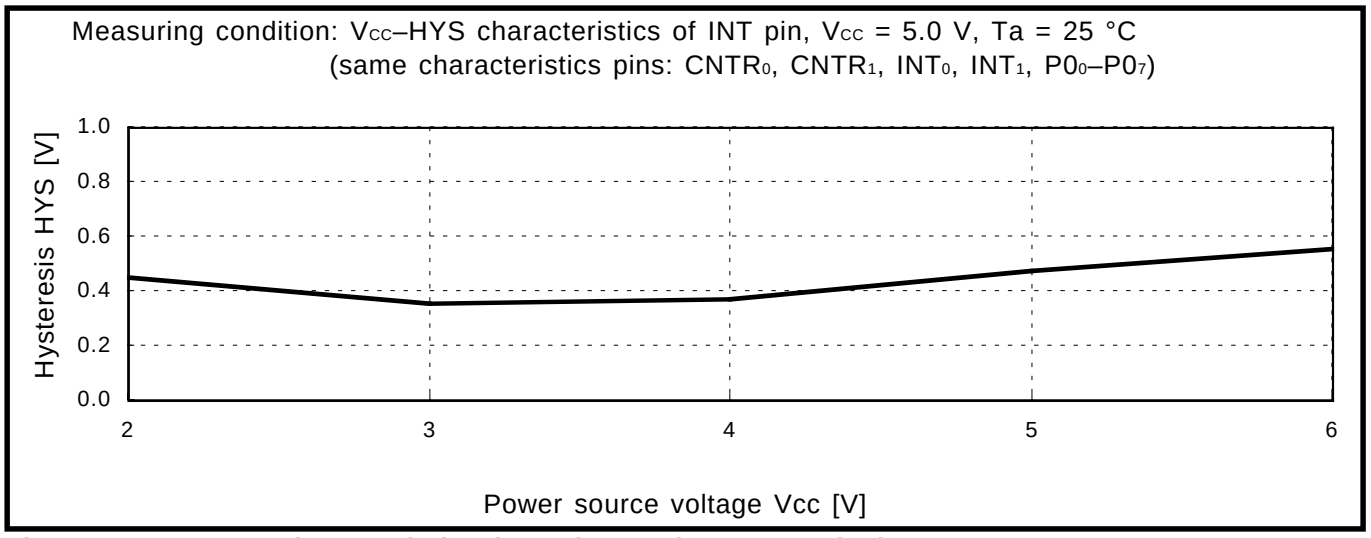


Fig. 3.2.23 Vcc-HYS characteristics (INT pin: Mask ROM version)

(5) Port typical characteristic example (V_{OH}-I_{OH} characteristics)

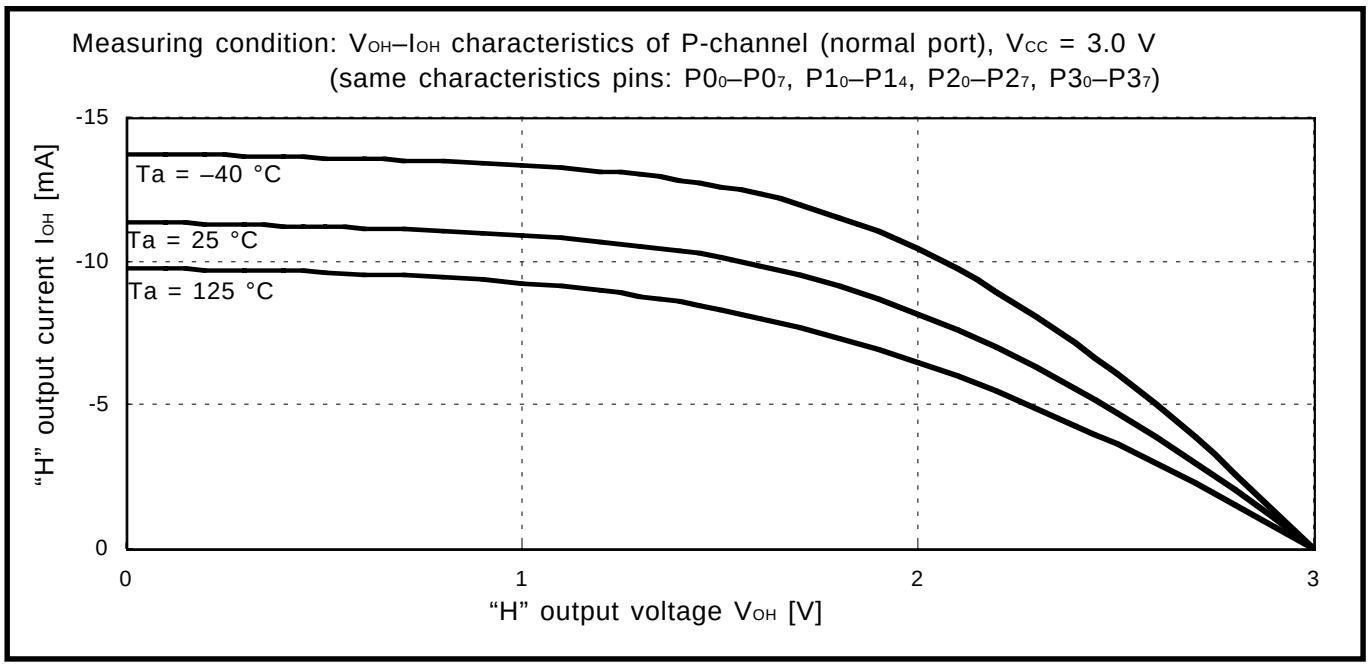


Fig. 3.2.24 V_{OH}-I_{OH} characteristics of P-channel (V_{CC} = 3.0 V, normal port: Mask ROM version)

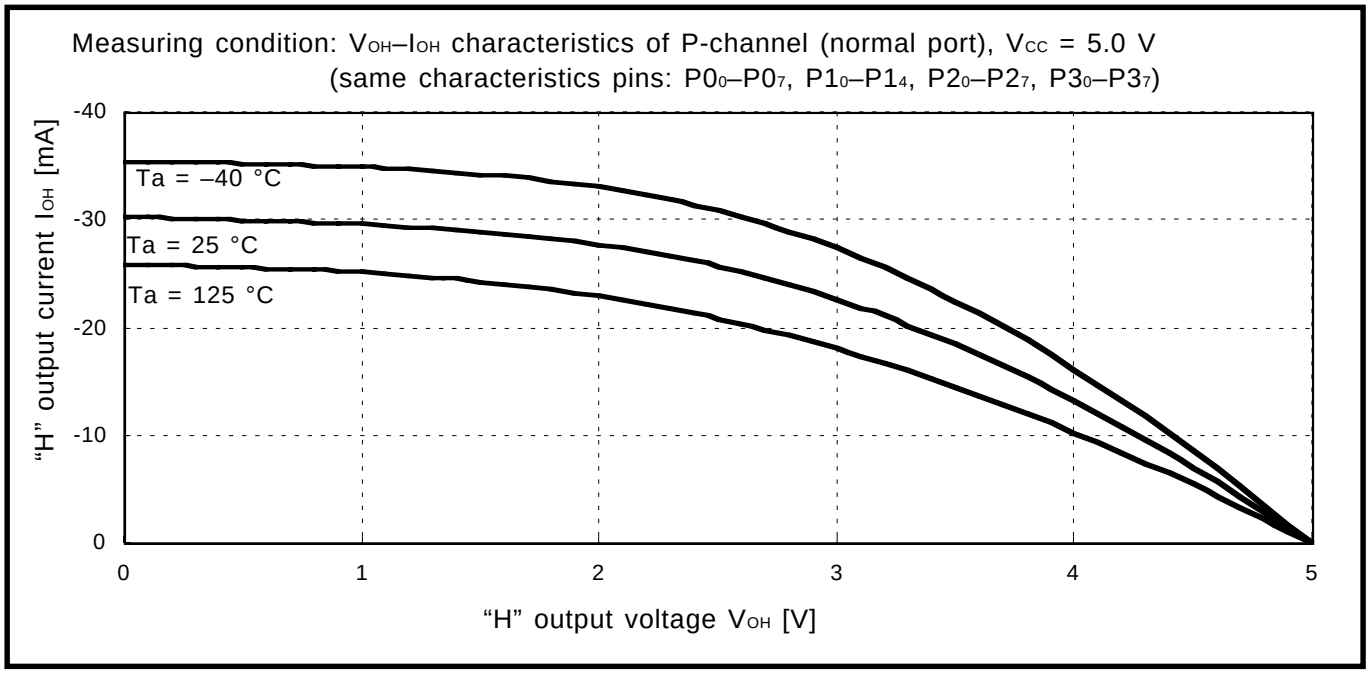


Fig. 3.2.25 V_{OH}-I_{OH} characteristics of P-channel (V_{CC} = 5.0 V, normal port: Mask ROM version)

(6) Port typical characteristic example (V_{OL}-I_{OL} characteristics)

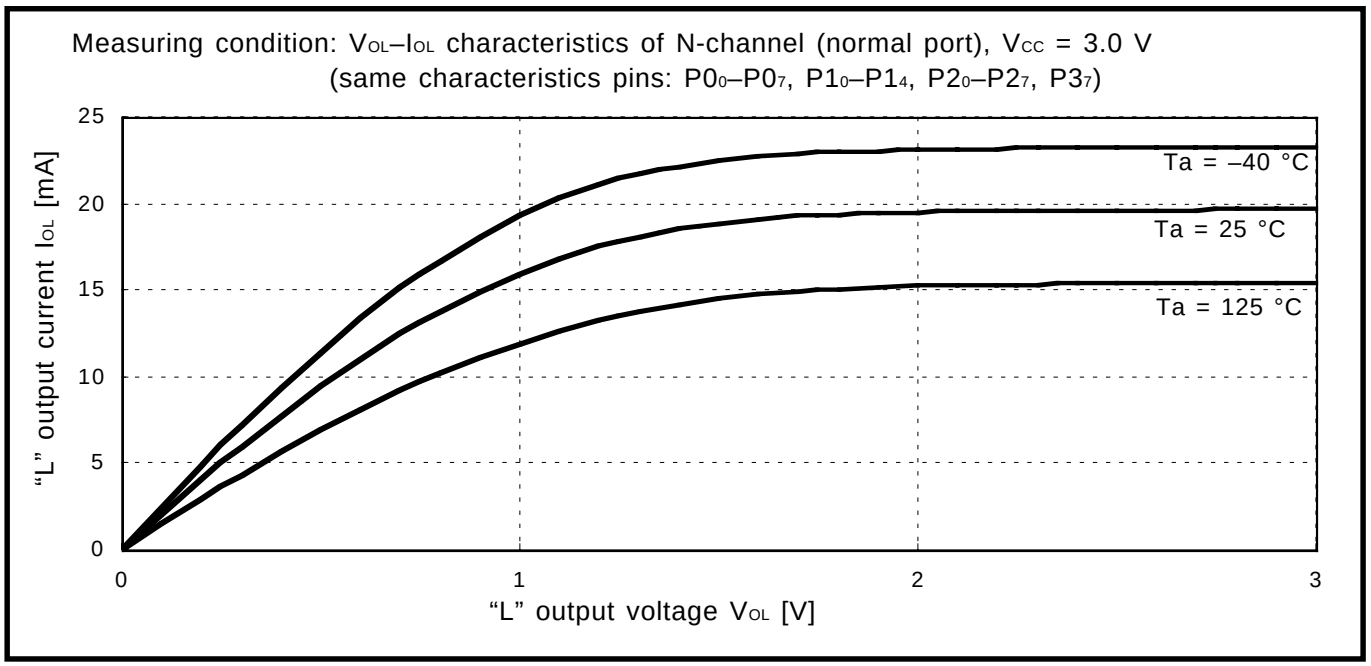


Fig. 3.2.26 V_{OL}-I_{OL} characteristics of N-channel (V_{CC} = 3.0 V, normal port: Mask ROM version)

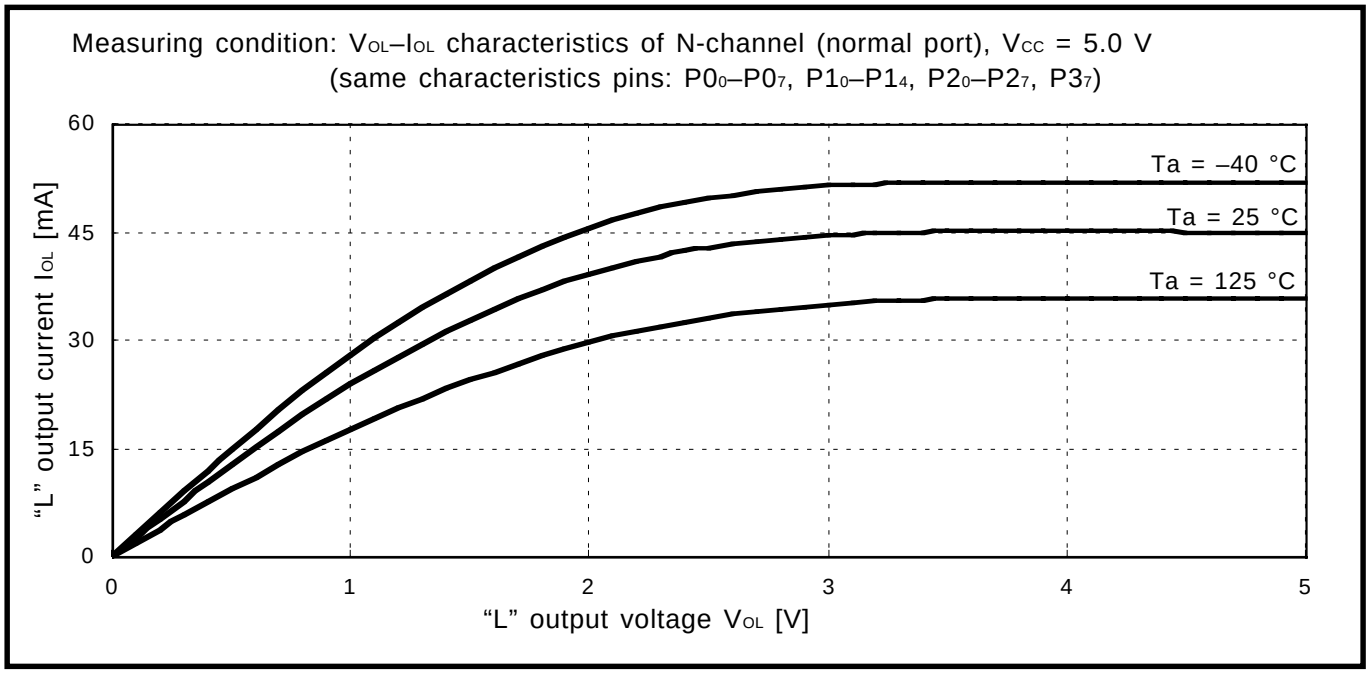


Fig. 3.2.27 V_{OL}-I_{OL} characteristics of N-channel (V_{CC} = 5.0 V, normal port: Mask ROM version)

APPENDIX

3.2 Typical characteristics

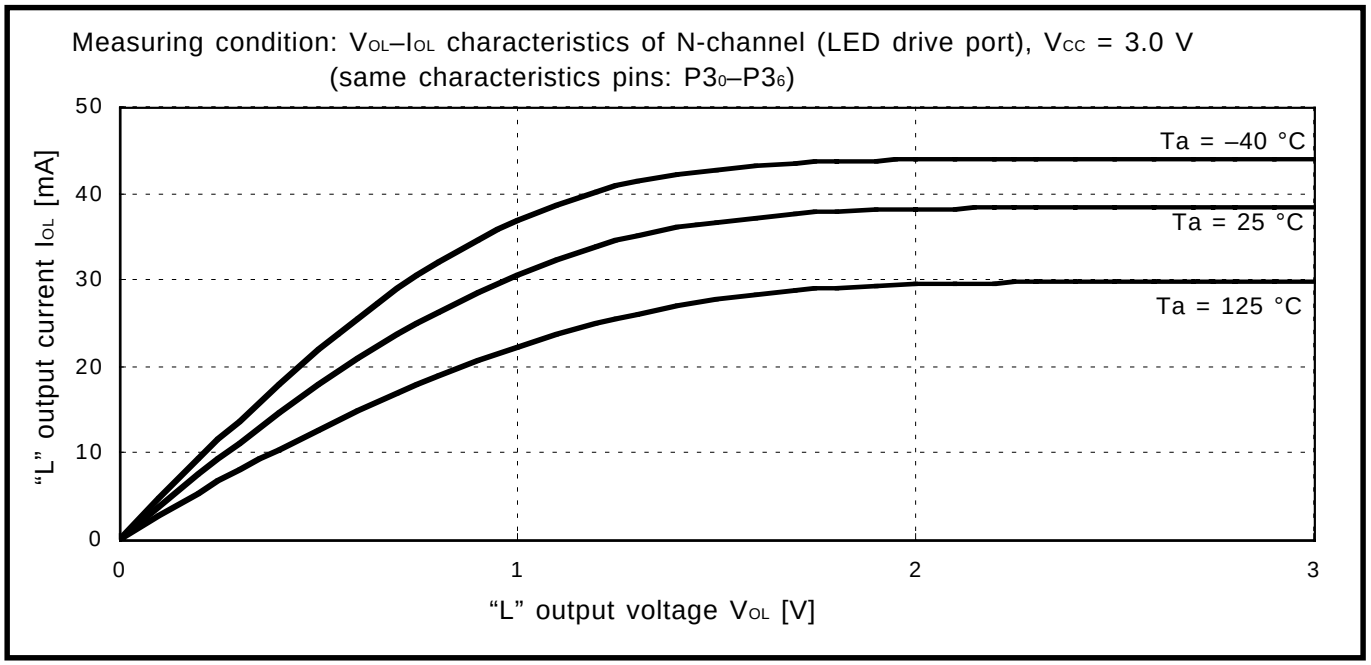


Fig. 3.2.28 V_{OL}-I_{OL} characteristics of N-channel (V_{CC} = 3.0 V, LED drive port: Mask ROM version)

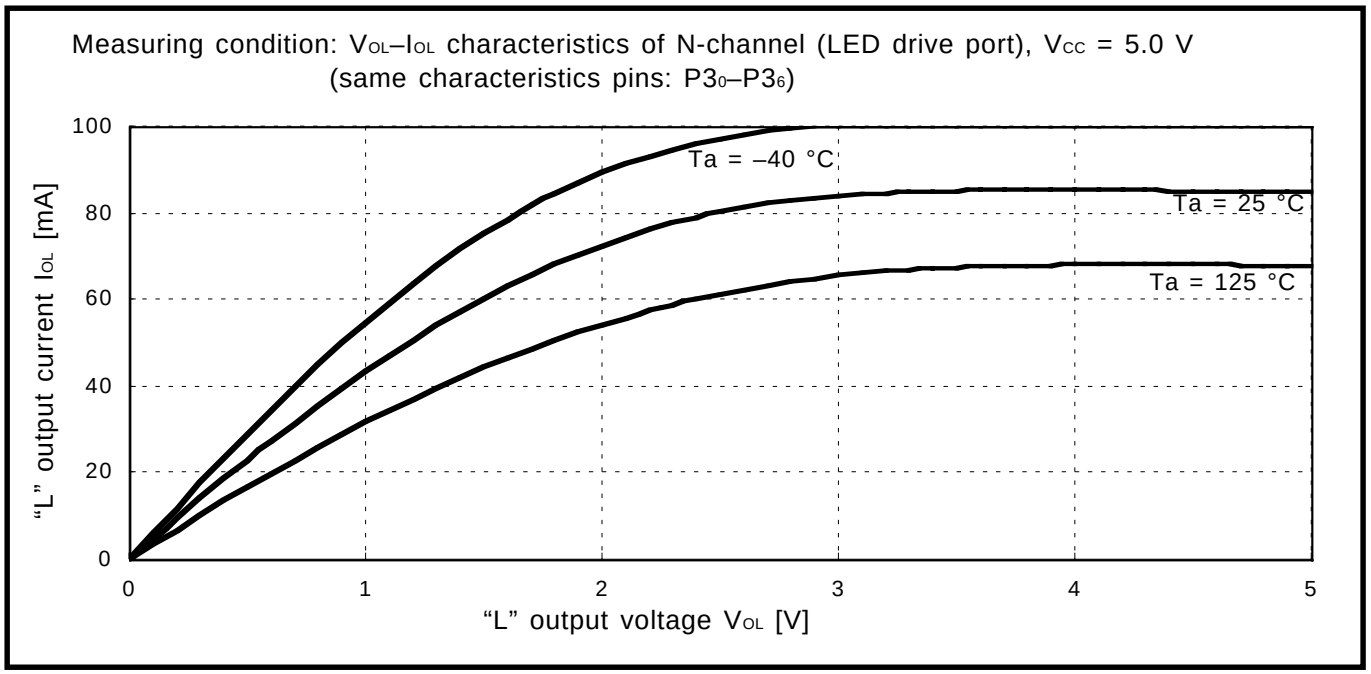


Fig. 3.2.29 V_{OL}-I_{OL} characteristics of N-channel (V_{CC} = 5.0 V, LED drive port: Mask ROM version)

(7) Port typical characteristic example (V_{CC}-IIL characteristics)

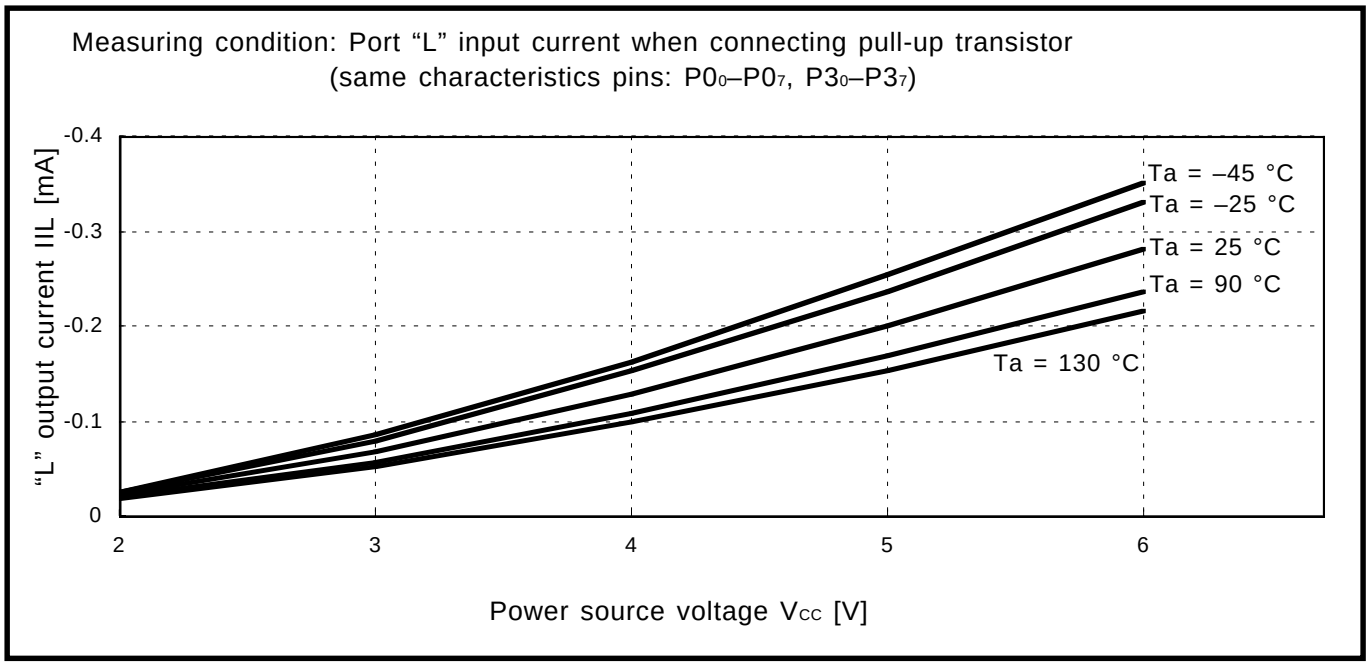


Fig. 3.2.30 V_{CC}-IIL characteristics (Port "L" input current when connecting pull-up transistor: Mask ROM version)

(8) Port typical characteristic example (V_{IN}-II(AD) characteristics)

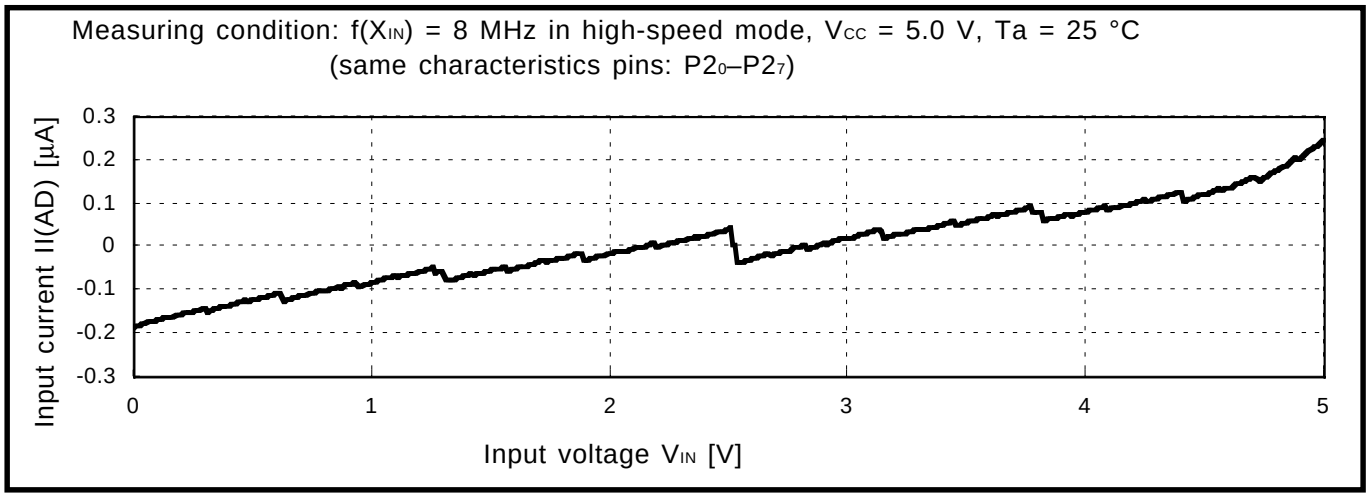


Fig. 3.2.31 V_{IN}-II(AD) characteristics (A-D port input current during A-D conversion, f(X_{IN}) = 8 MHz in high-speed mode: Mask ROM version)

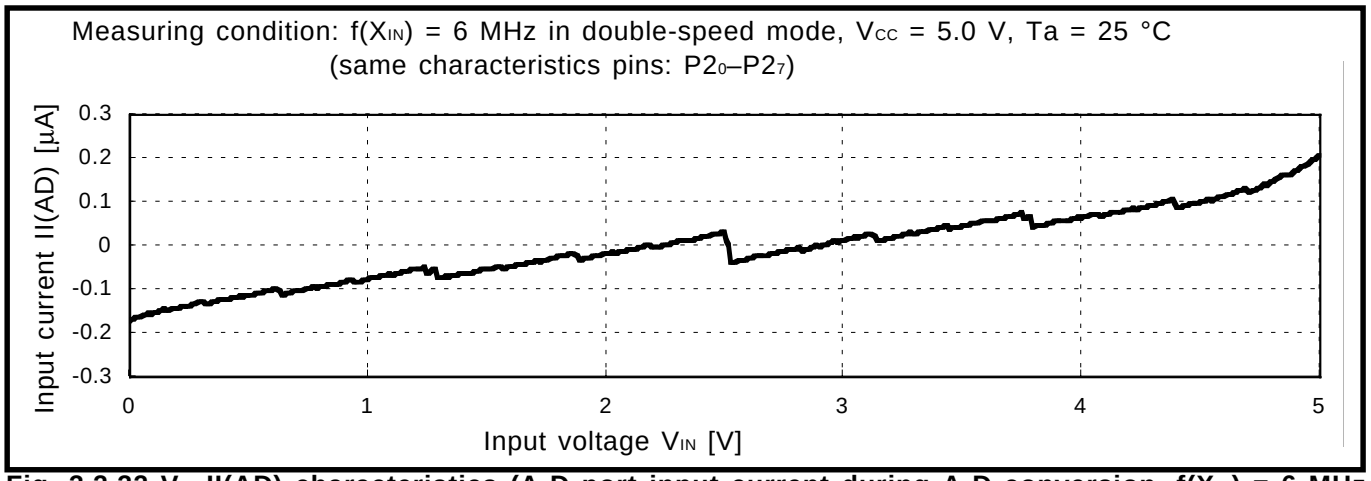


Fig. 3.2.32 V_{IN}-II(AD) characteristics (A-D port input current during A-D conversion, f(X_{IN}) = 6 MHz in double-speed mode: Mask ROM version)

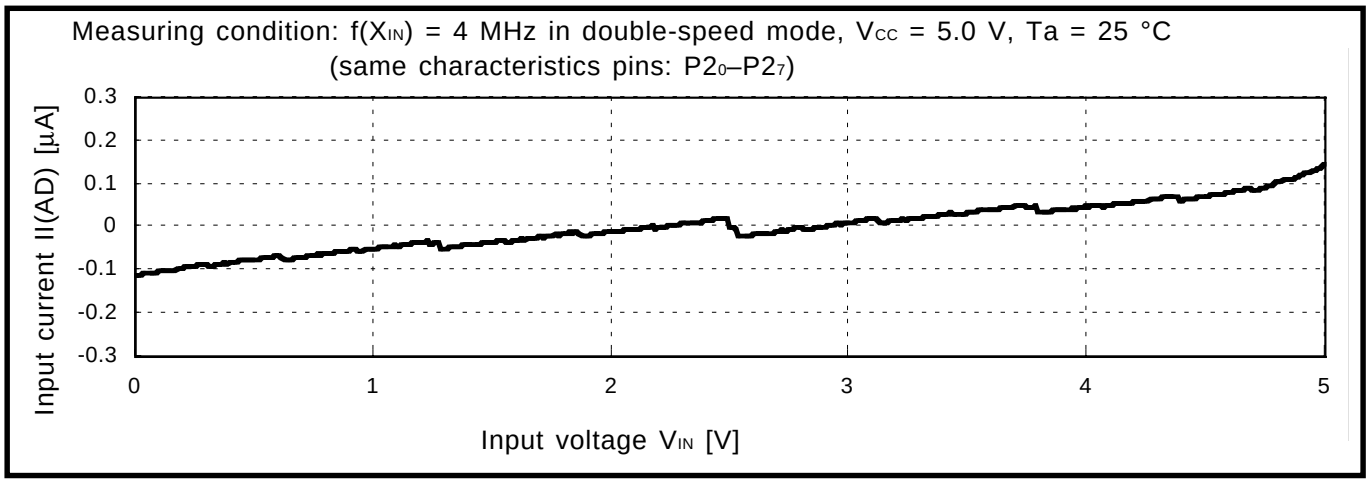


Fig. 3.2.33 V_{IN}-II(AD) characteristics (A-D port input current during A-D conversion, f(X_{IN}) = 4 MHz in double-speed mode: Mask ROM version)

(9) Ring oscillator frequency typical characteristic example

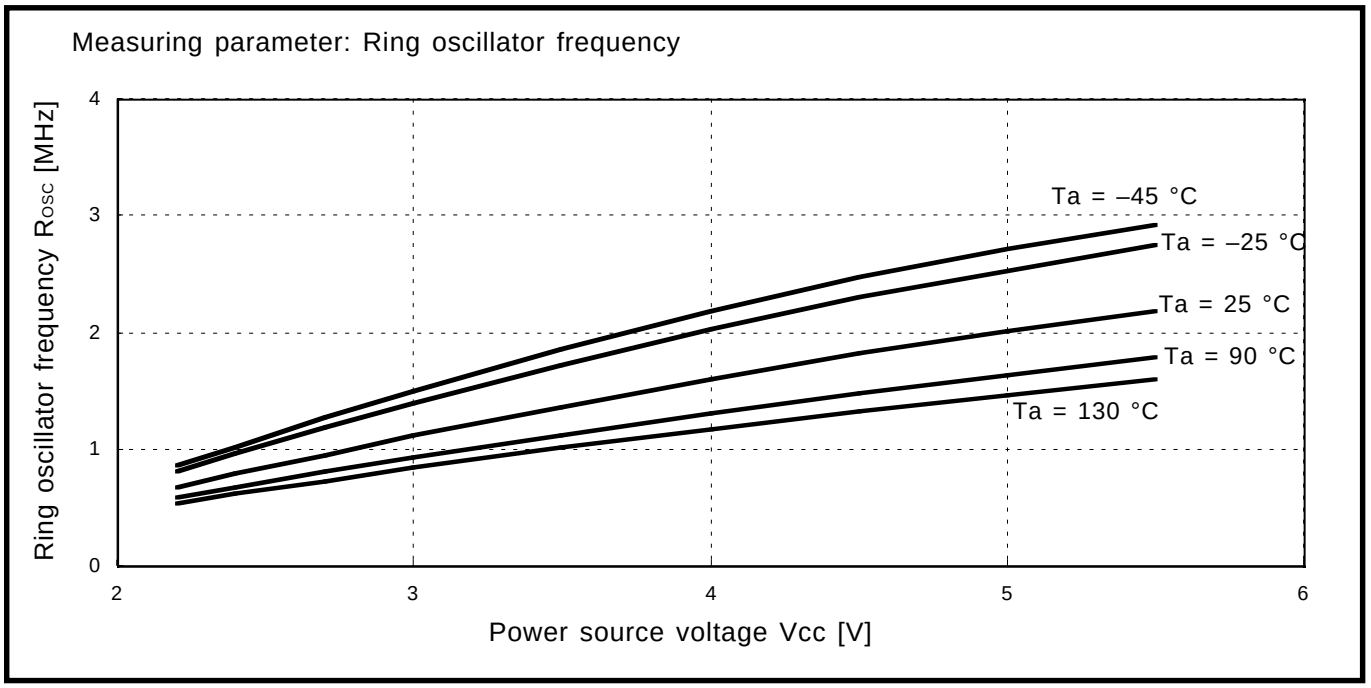


Fig. 3.2.34 V_{CC}-R_{OSC} characteristics (ring oscillator frequency: Mask ROM version)

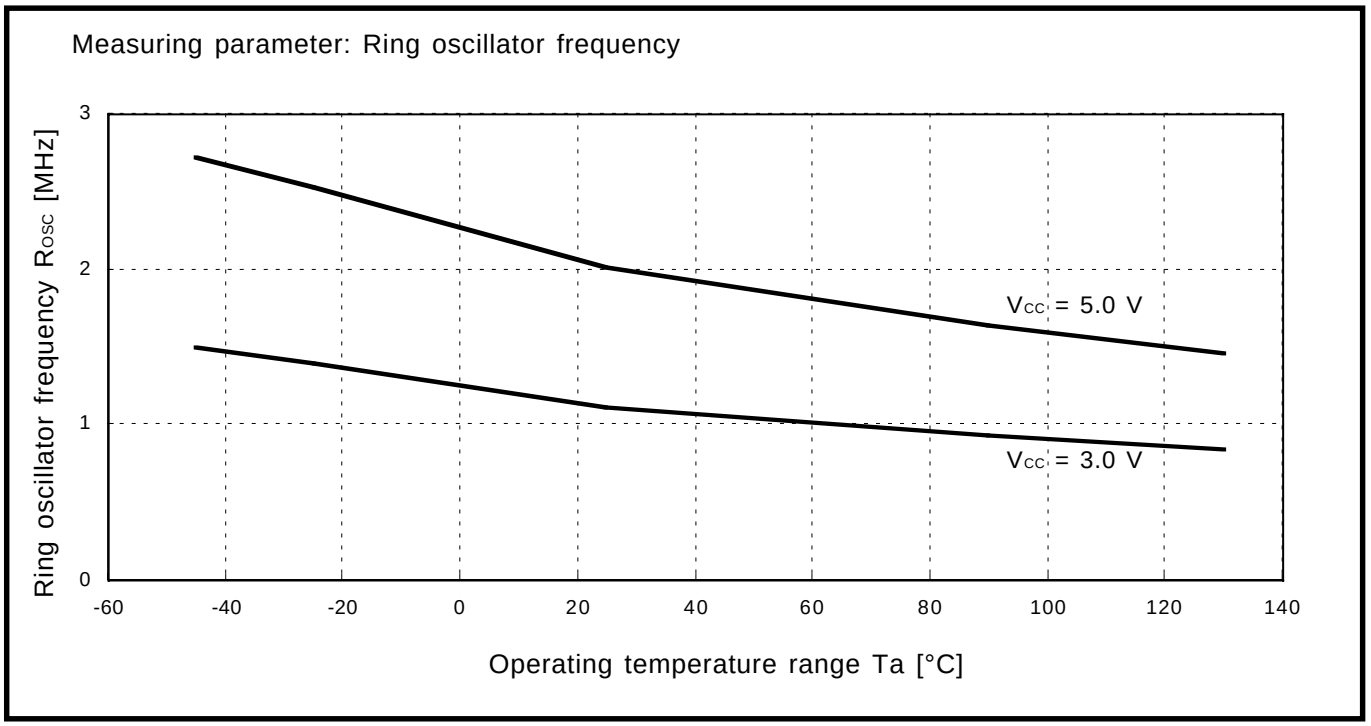


Fig. 3.2.35 Ta-R_{OSC} characteristics (ring oscillator frequency: Mask ROM version)

(10) RC oscillation frequency typical characteristic example

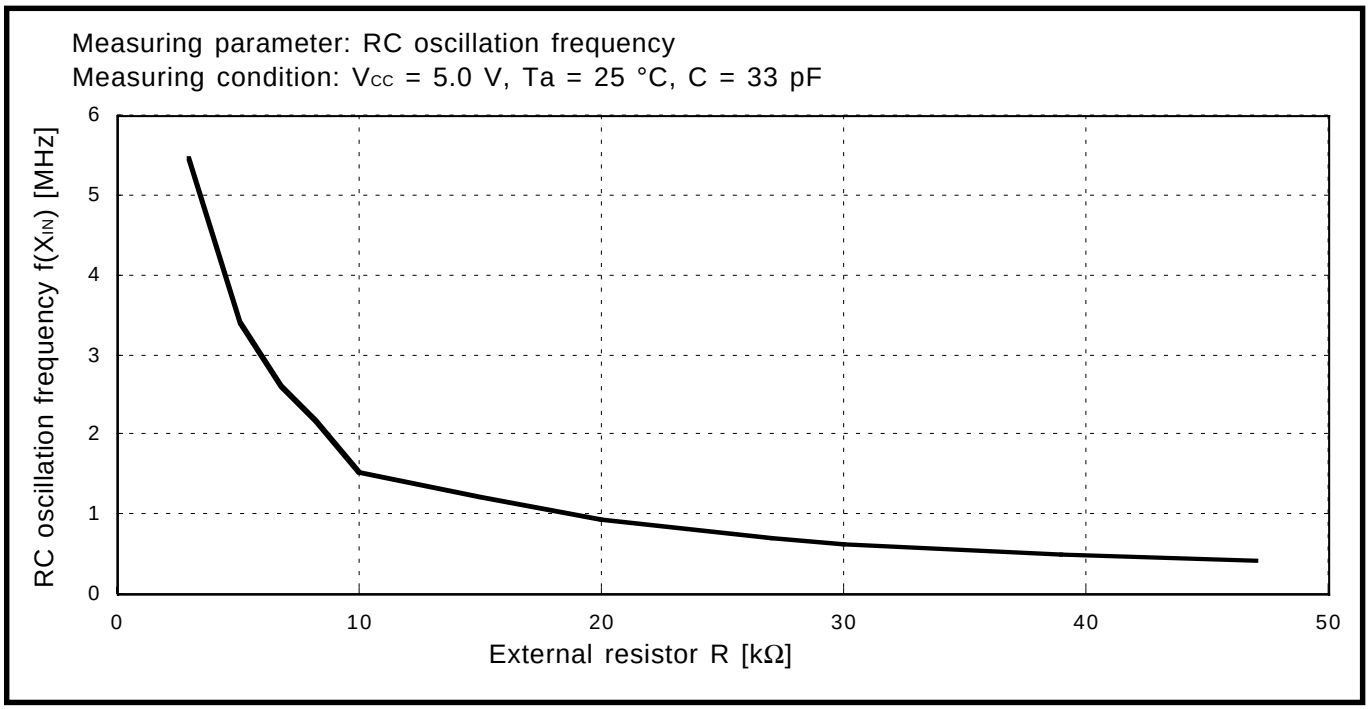


Fig. 3.2.36 R-f(X_{IN}) characteristics (RC oscillation frequency: Mask ROM version)

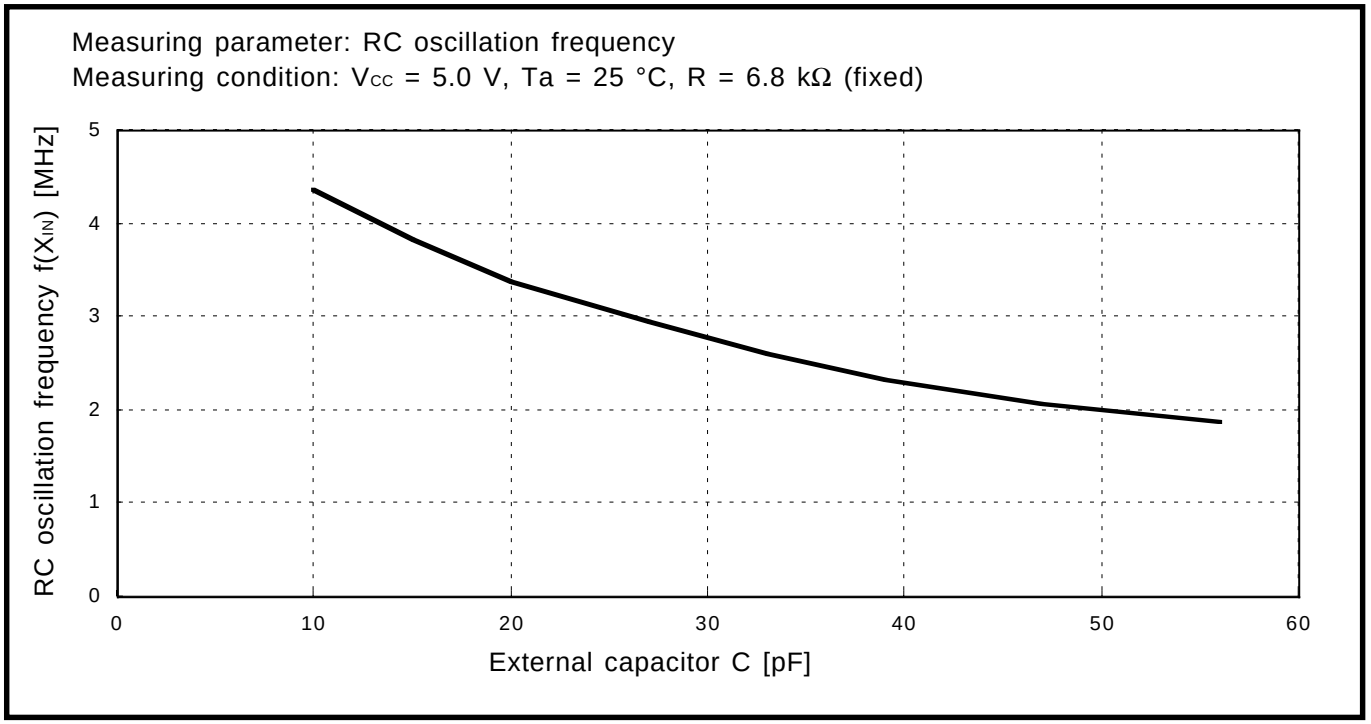


Fig. 3.2.37 C-f(X_{IN}) characteristics (RC oscillation frequency: Mask ROM version)

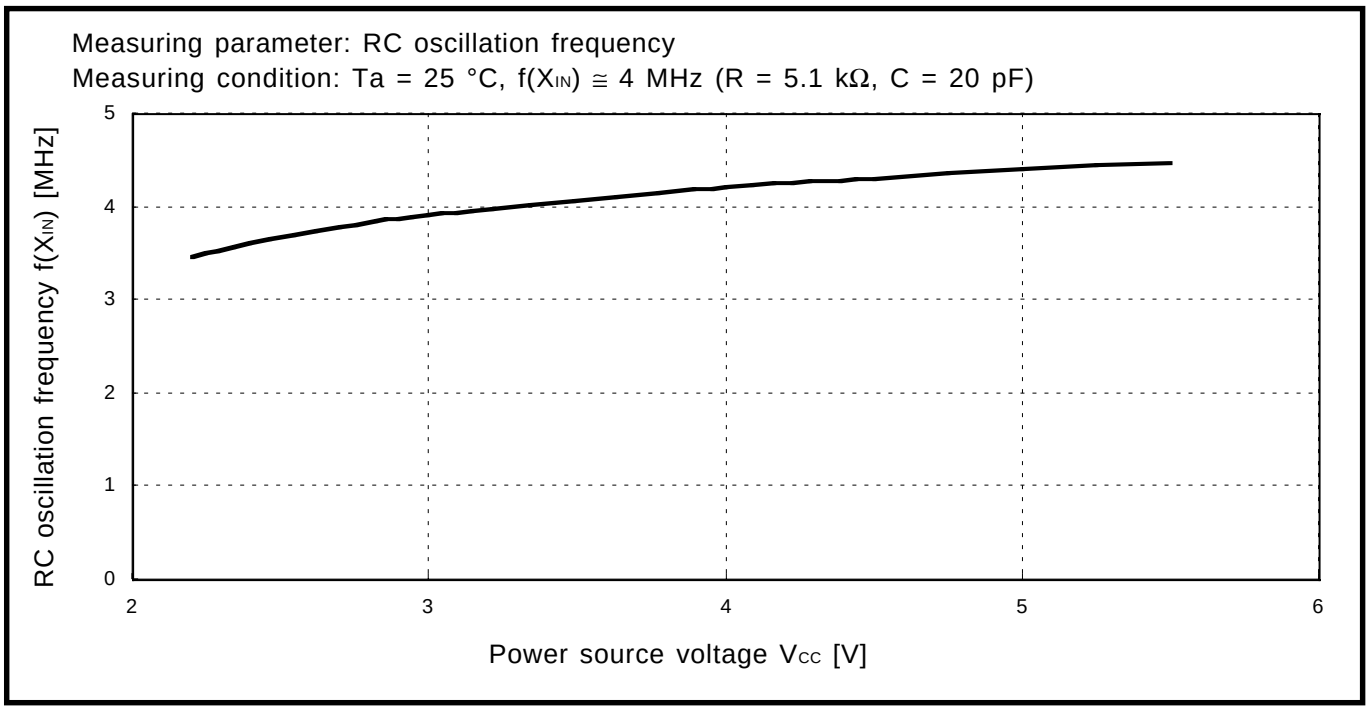


Fig. 3.2.38 V_{CC}-f(X_{IN}) characteristics (RC oscillation frequency: Mask ROM version)

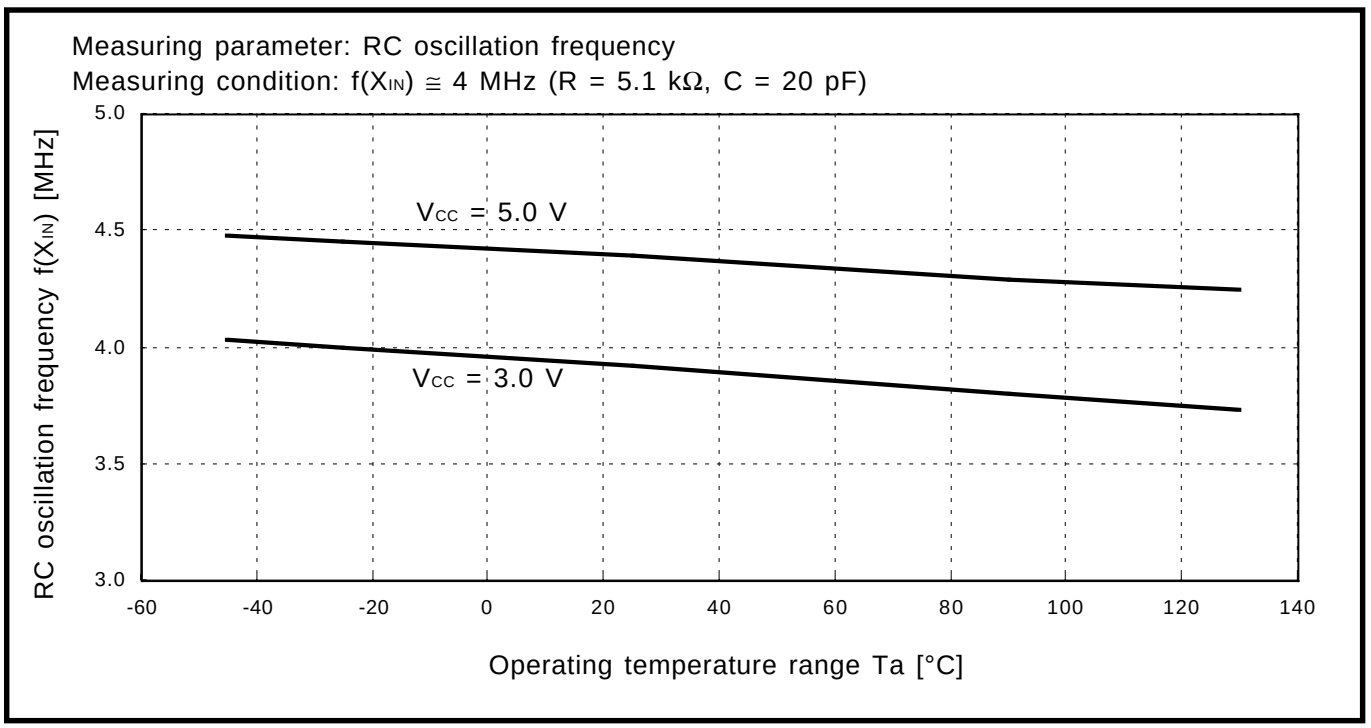


Fig. 3.2.39 Ta-f(X_{IN}) characteristics (RC oscillation frequency: Mask ROM version)

(11) A-D conversion typical characteristics example

① Definition of A-D conversion accuracy

The A-D conversion accuracy is defined below (refer to Fig. 3.2.40).

●Relative accuracy

- Zero transition voltage (V_{OT})
  This means an analog input voltage when the actual A-D conversion output data changes from "0" to "1."
- Full-scale transition voltage (V_{FST})
  This means an analog input voltage when the actual A-D conversion output data changes from "1023" to "1022."
- Non-linearity error
  This means a deviation from the line between V_{OT} and V_{FST} of a converted value between V_{OT} and V_{FST}.
- Differential non-linearity error
  This means a deviation from the input potential difference required to change a converted value between V_{OT} and V_{FST} by 1 LSB of the 1 LSB at the relative accuracy.

●Absolute accuracy

This means a deviation from the ideal characteristics between 0 to V_{REF} of actual A-D conversion characteristics.

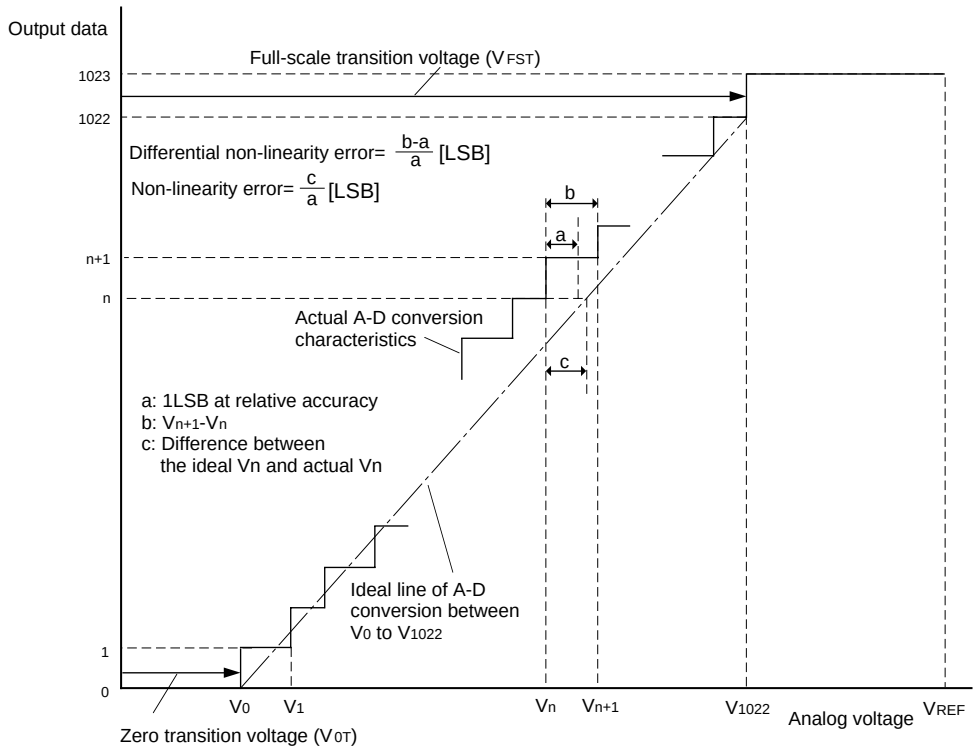


Fig. 3.2.40 Definition of A-D conversion accuracy

Vn: Analog input voltage when the output data changes from "n" to "n + 1" (n = 0 to 1022)

- 1 LSB at relative accuracy $\rightarrow \dfrac{V_{FST} - V_{OT}}{1022}$ (V)
- 1 LSB at absolute accuracy $\rightarrow \dfrac{V_{REF}}{1024}$ (V)

② **A-D conversion accuracy typical characteristics-1**

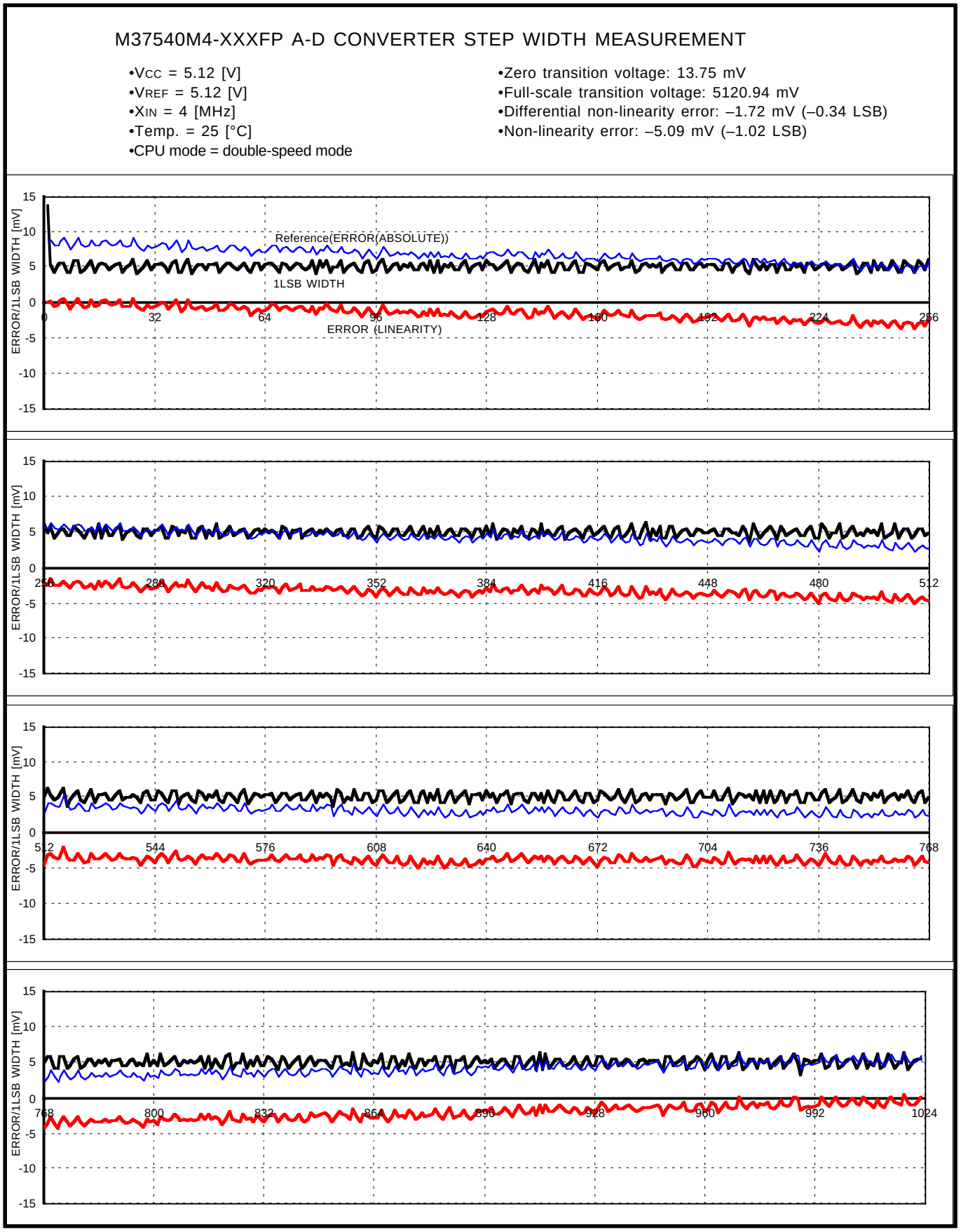


Fig. 3.2.41 A-D conversion accuracy typical characteristic example-1 (Mask ROM version)

③ A-D conversion accuracy typical characteristics-2

M37540M4-XXXFP A-D CONVERTER STEP WIDTH MEASUREMENT

- V_{CC} = 5.12 [V]
- V_{REF} = 5.12 [V]
- X_{IN} = 6 [MHz]
- Temp. = 25 [°C]
- CPU mode = double-speed mode
- Zero transition voltage: 14.38 mV
- Full-scale transition voltage: 5121.88 mV
- Differential non-linearity error: 1.41 mV (0.28 LSB)
- Non-linearity error: –4.23 mV (–0.85 LSB)

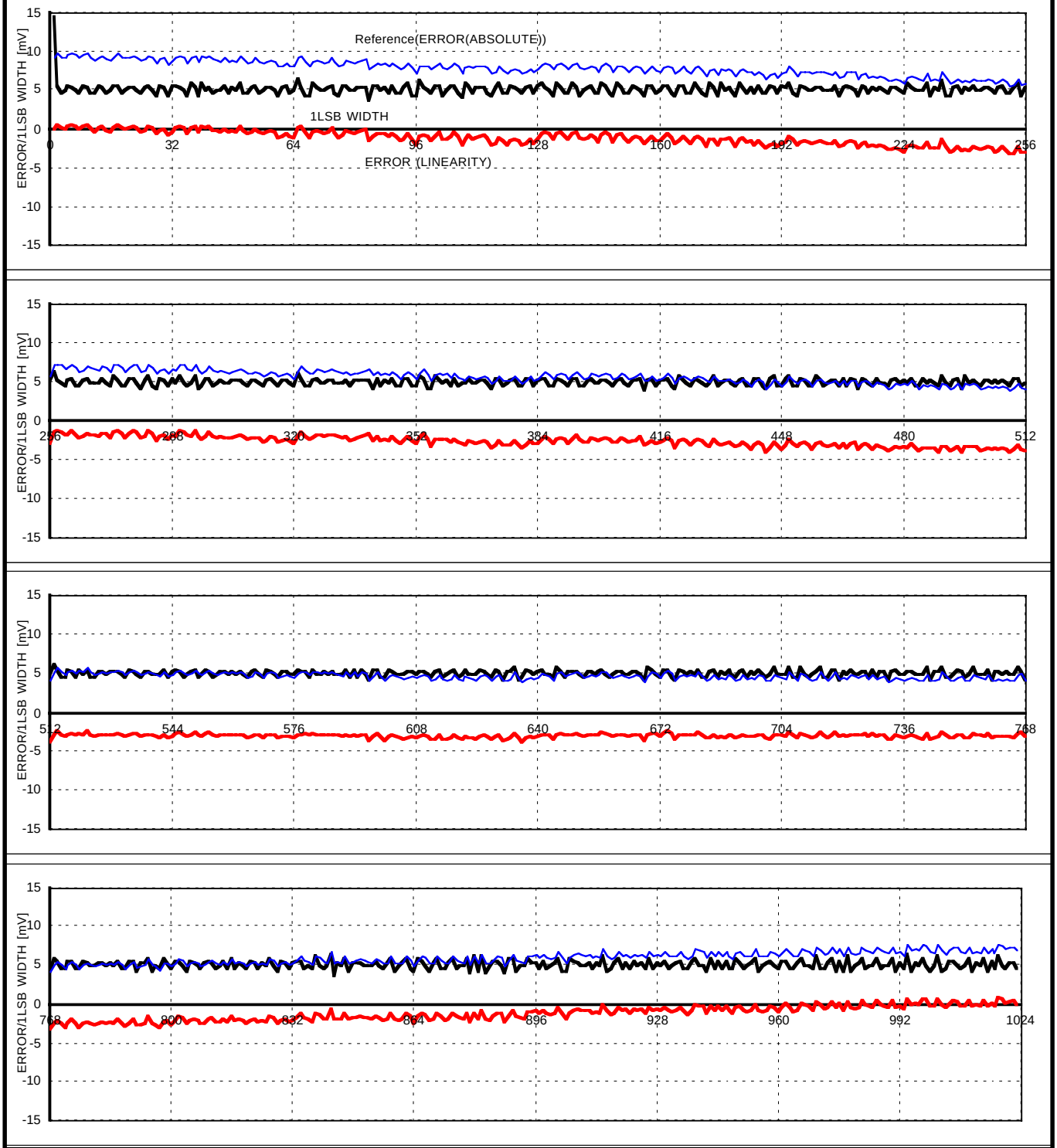


Fig. 3.2.42 A-D conversion accuracy typical characteristic example-2 (Mask ROM version)

④ A-D conversion accuracy typical characteristics-3

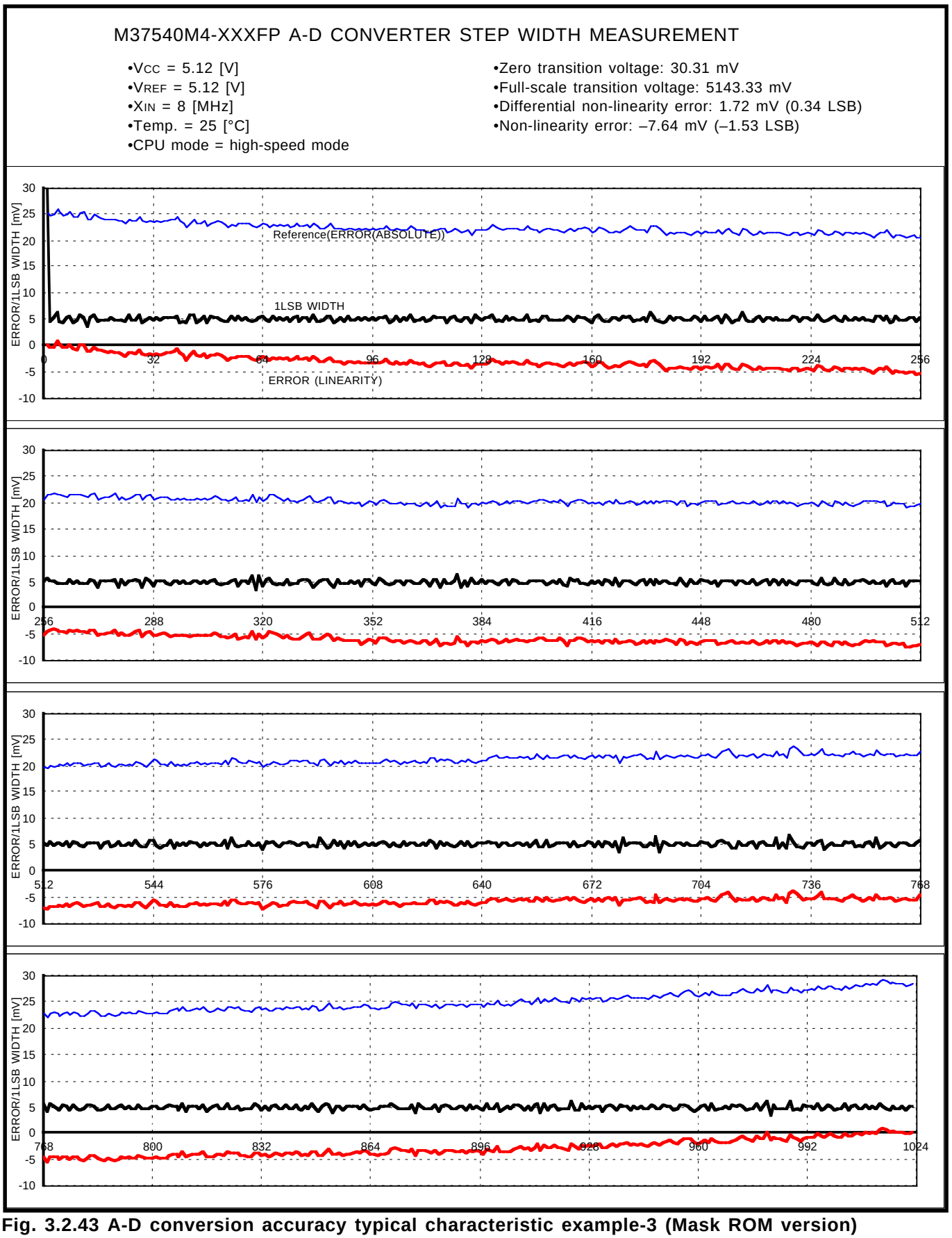


Fig. 3.2.43 A-D conversion accuracy typical characteristic example-3 (Mask ROM version)

3.2.2 One Time PROM version

(1) Power source current characteristic example (V_{CC}-I_{CC} characteristics)

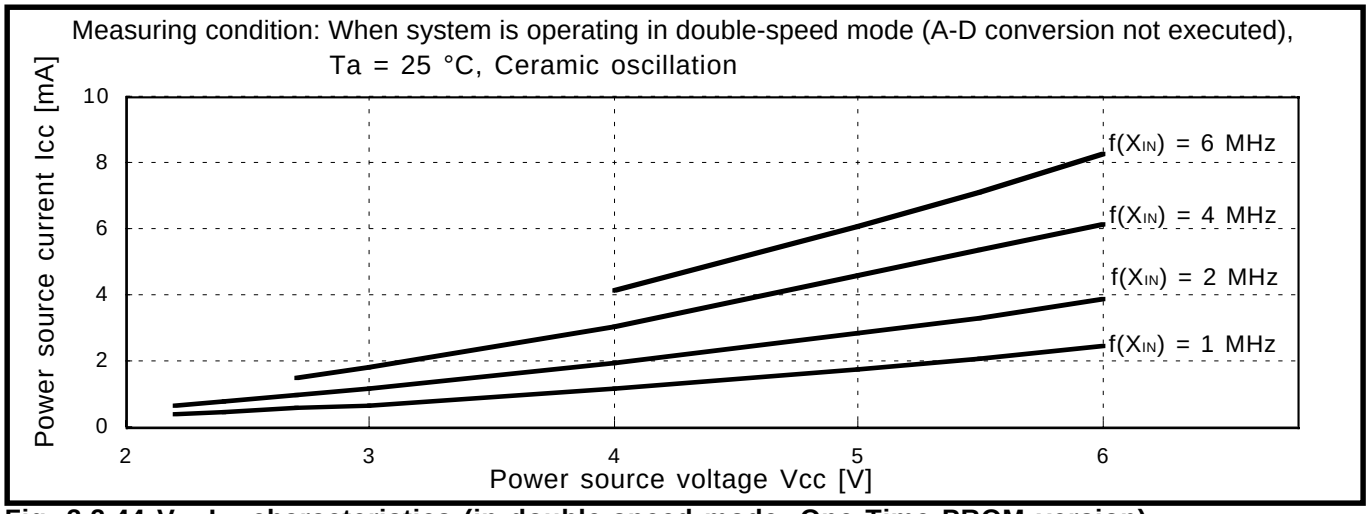


Fig. 3.2.44 V_{CC}-I_{CC} characteristics (in double-speed mode: One Time PROM version)

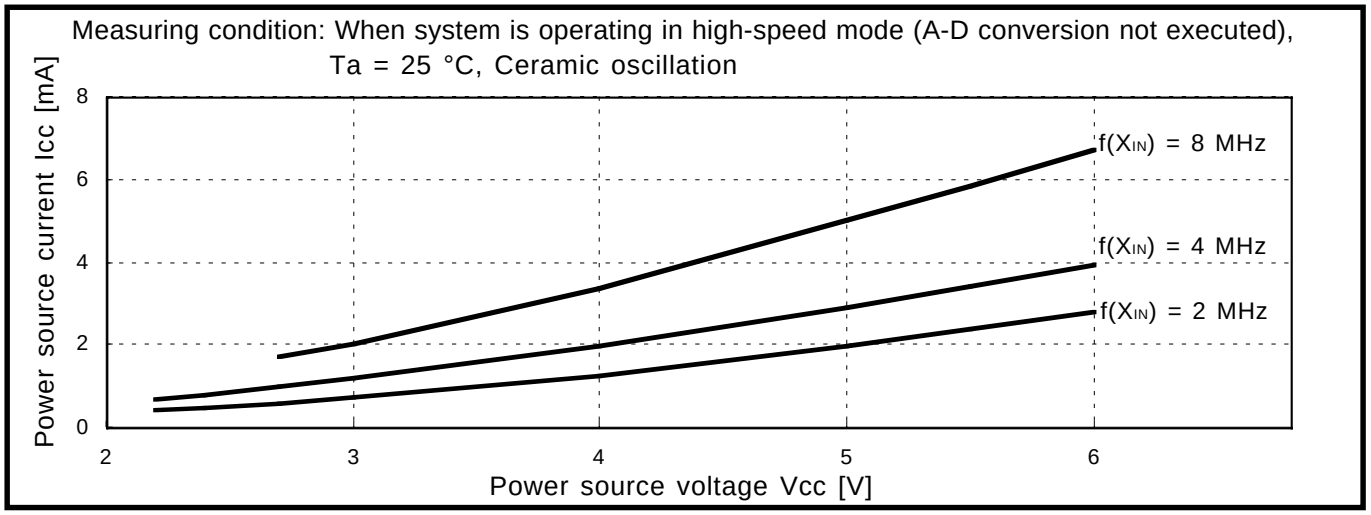


Fig. 3.2.45 V_{CC}-I_{CC} characteristics (in high-speed mode: One Time PROM version)

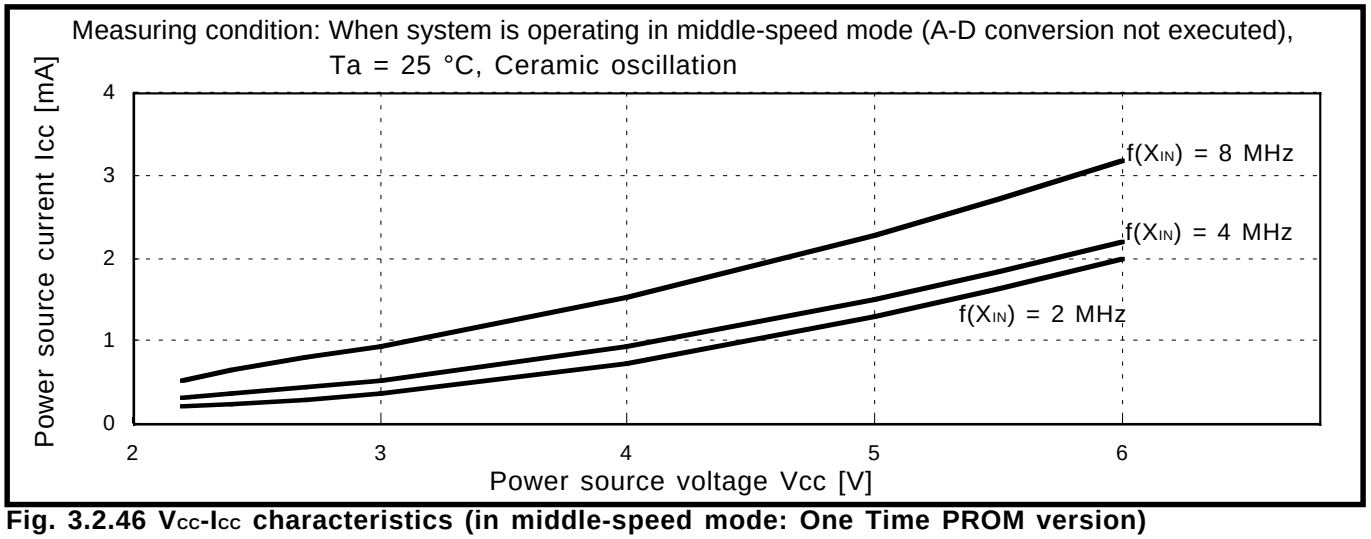


Fig. 3.2.46 V_{CC}-I_{CC} characteristics (in middle-speed mode: One Time PROM version)

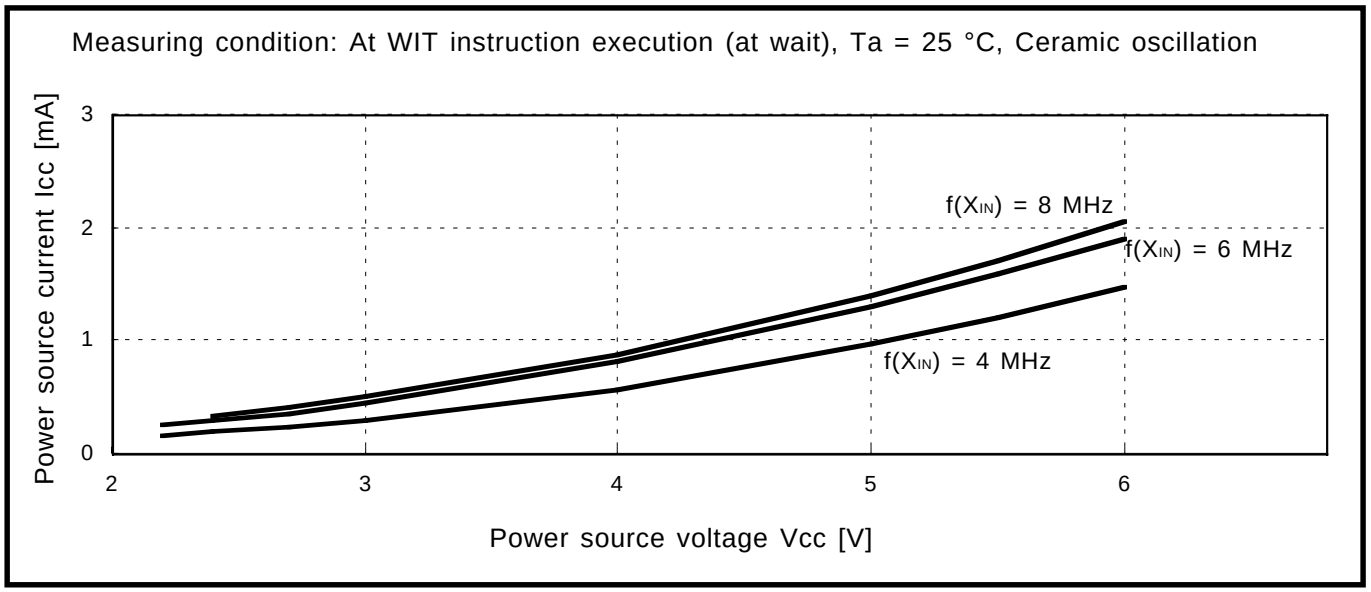


Fig. 3.2.47 V_{CC}-I_{CC} characteristics (at WIT instruction execution: One Time PROM version)

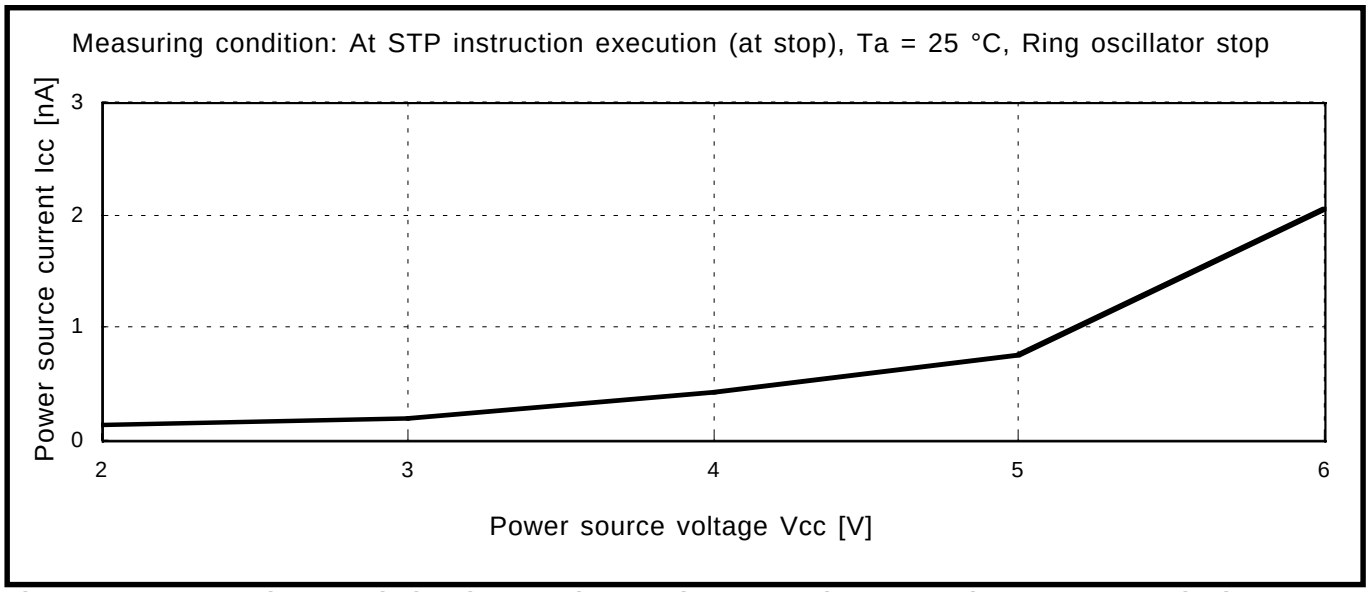


Fig. 3.2.48 V_{CC}-I_{CC} characteristics (at STP instruction execution: One Time PROM version)

APPENDIX

3.2 Typical characteristics

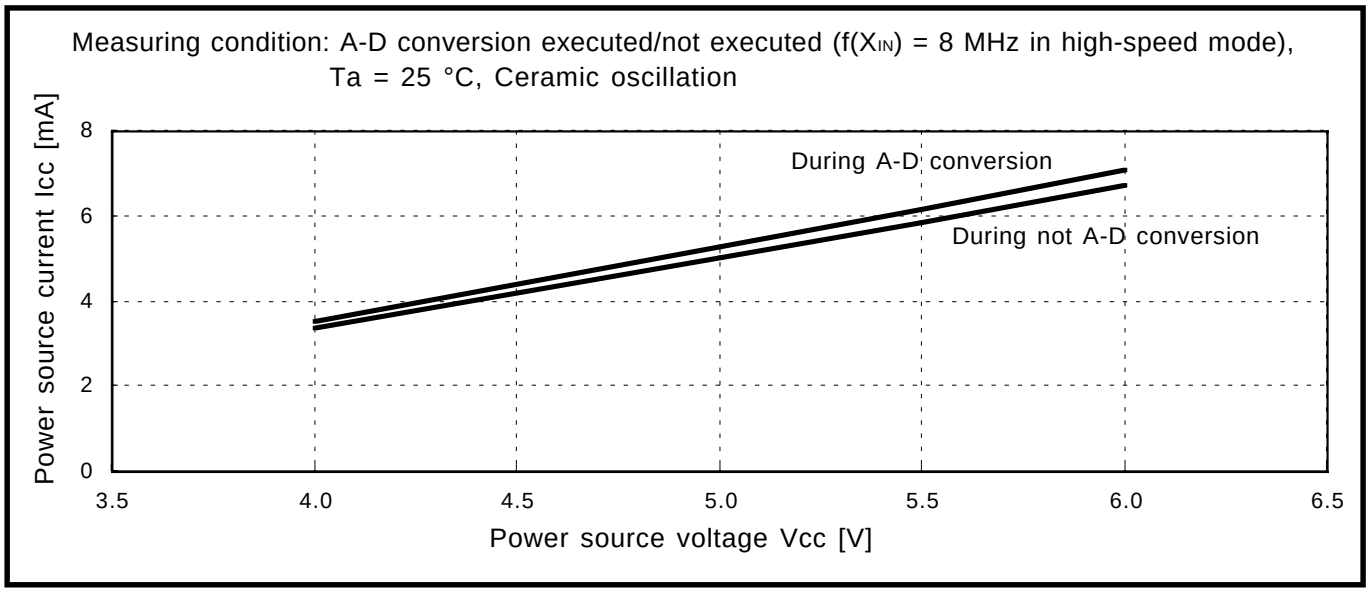


Fig. 3.2.49 V_{CC}-I_{CC} characteristics (addition when operating A-D conversion, f(X_{IN}) = 8 MHz in high-speed mode: One Time PROM version)

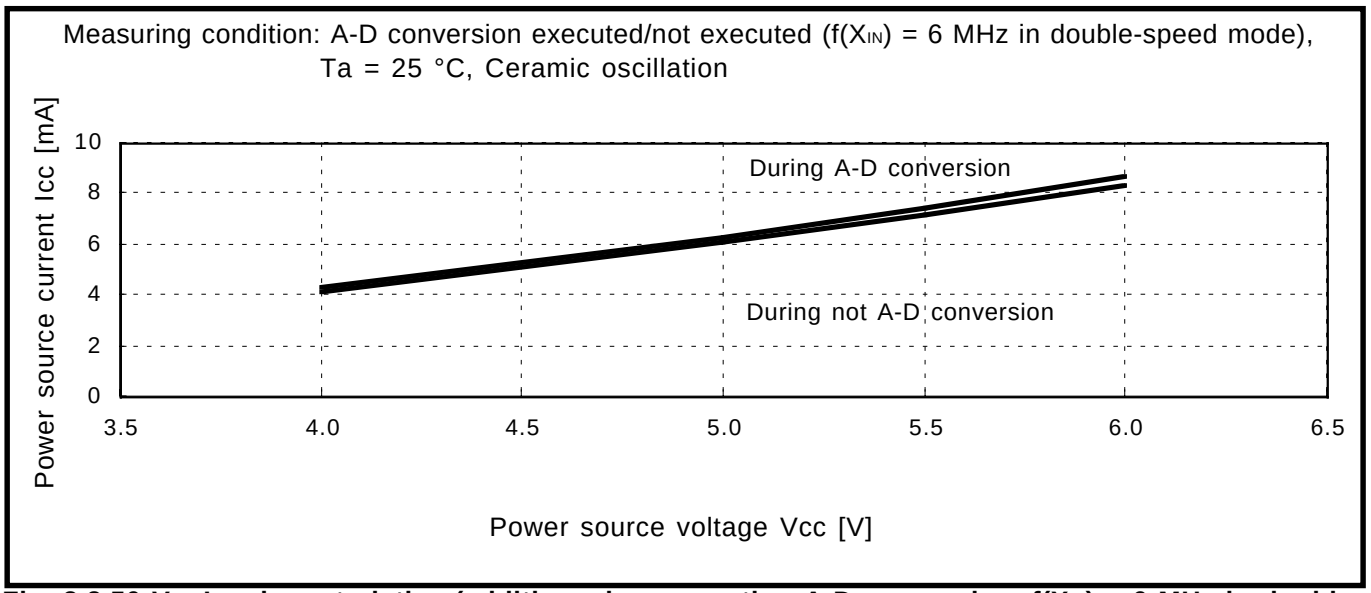


Fig. 3.2.50 V_{CC}-I_{CC} characteristics (addition when operating A-D conversion, f(X_{IN}) = 6 MHz in double-speed mode: One Time PROM version)

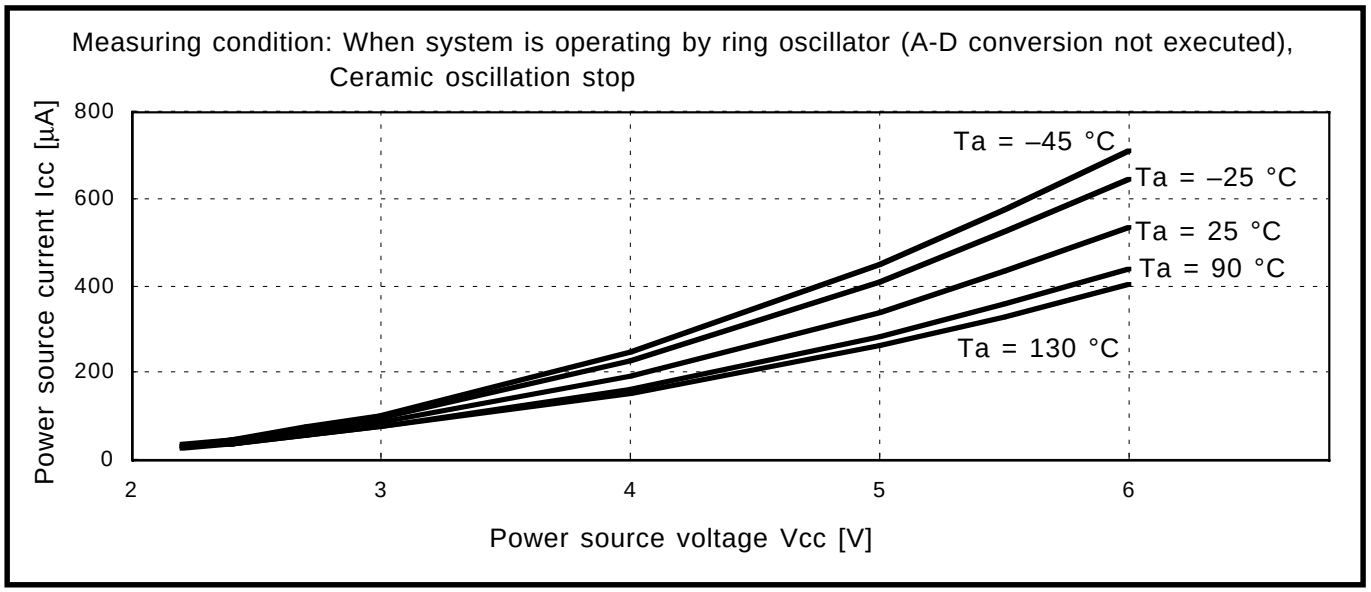


Fig. 3.2.51 V_{CC}-I_{CC} characteristics (When system is operating by ring oscillator, Ceramic oscillation stop: One Time PROM version)

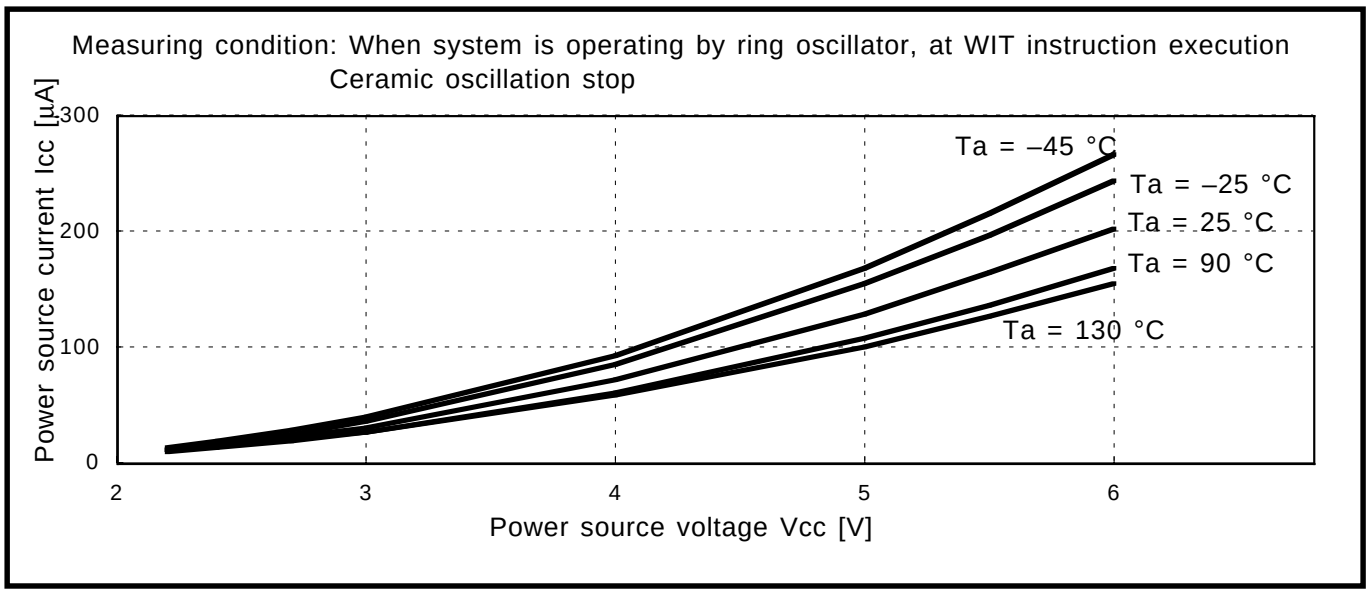


Fig. 3.2.52 V_{CC}-I_{CC} characteristics (When system is operating by ring oscillator, at WIT instruction execution, Ceramic oscillation stop: One Time PROM version)

(2) Power source current characteristic example ($f(X_{IN})$-I_{CC} characteristics)

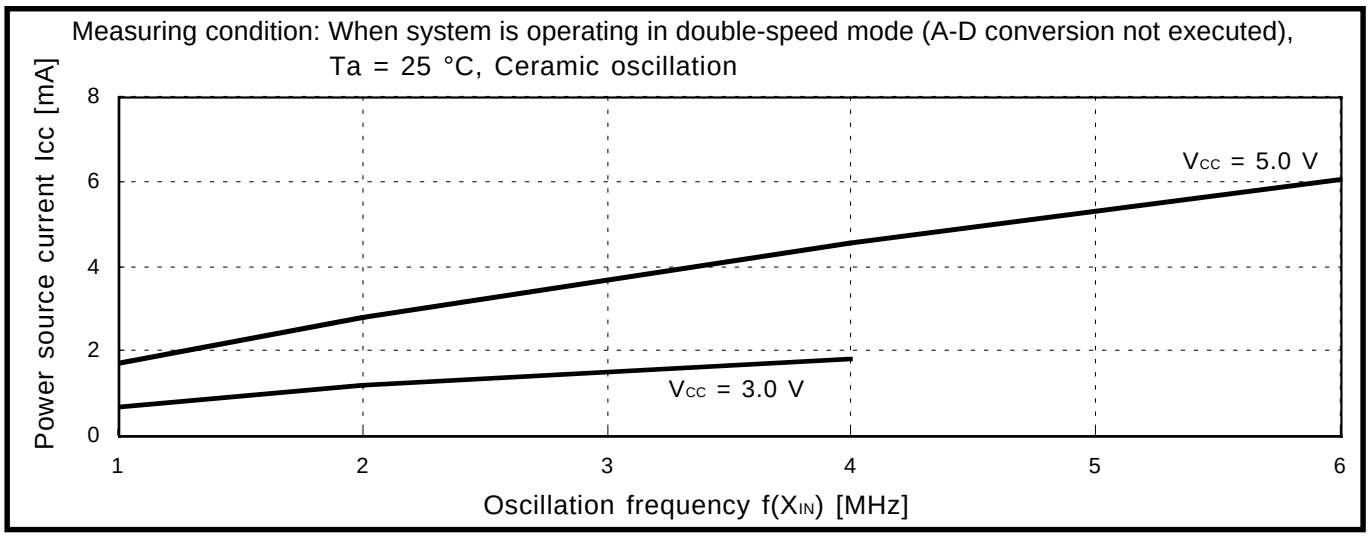


Fig. 3.2.53 $f(X_{IN})$-I_{CC} characteristics (in double-speed mode: One Time PROM version)

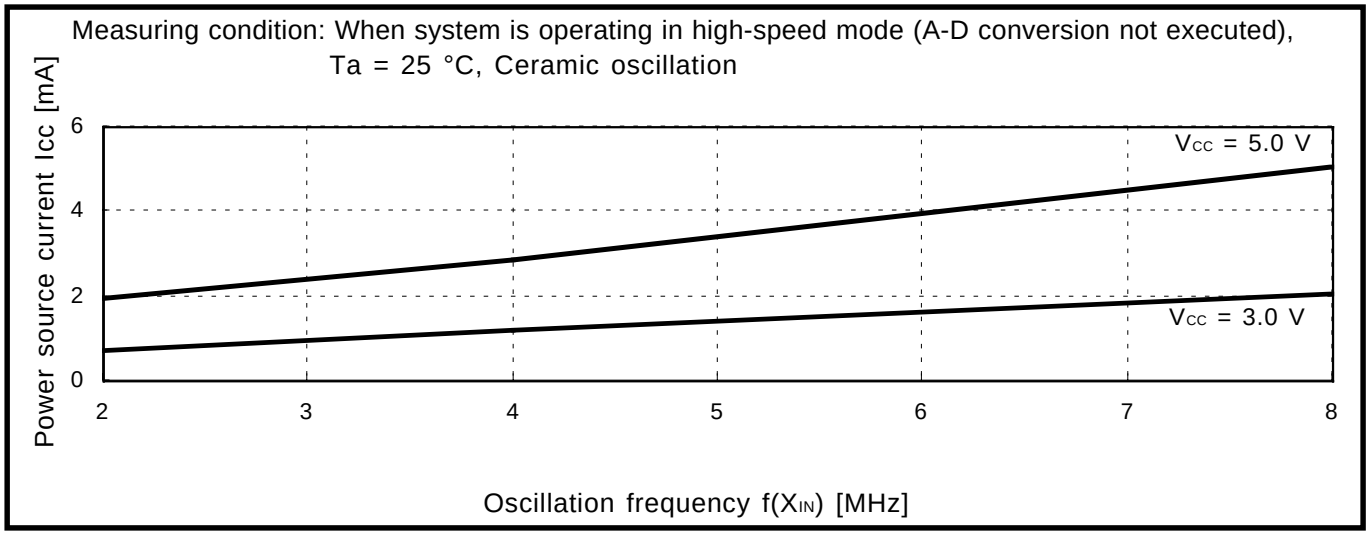


Fig. 3.2.54 $f(X_{IN})$-I_{CC} characteristics (in high-speed mode: One Time PROM version)

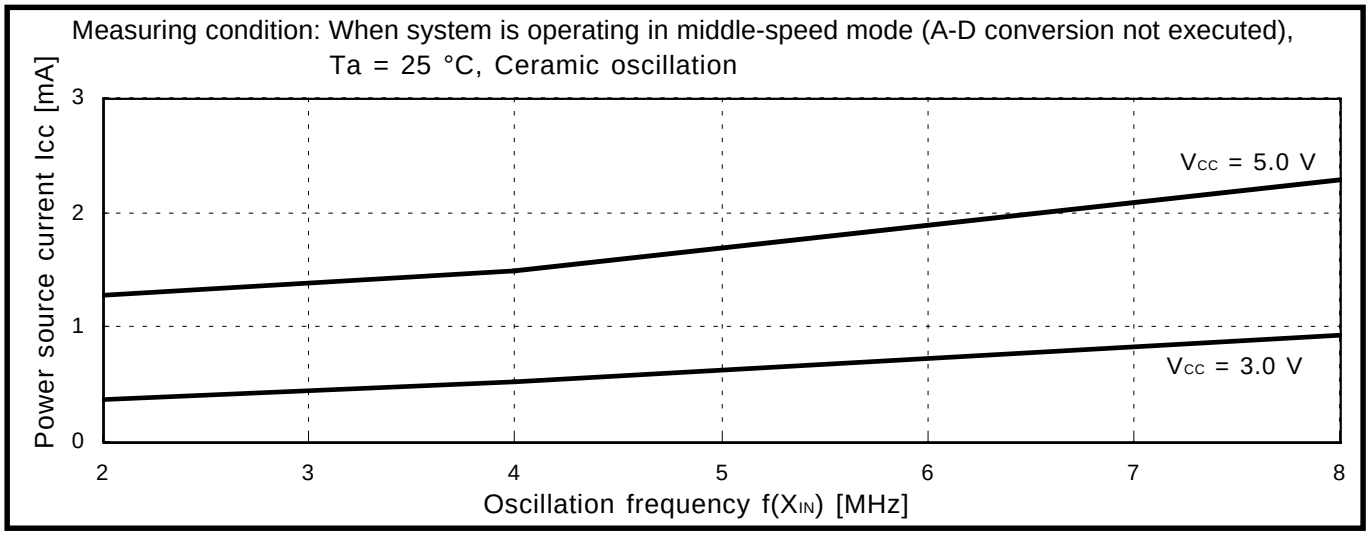


Fig. 3.2.55 $f(X_{IN})$-I_{CC} characteristics (in middle-speed mode: One Time PROM version)

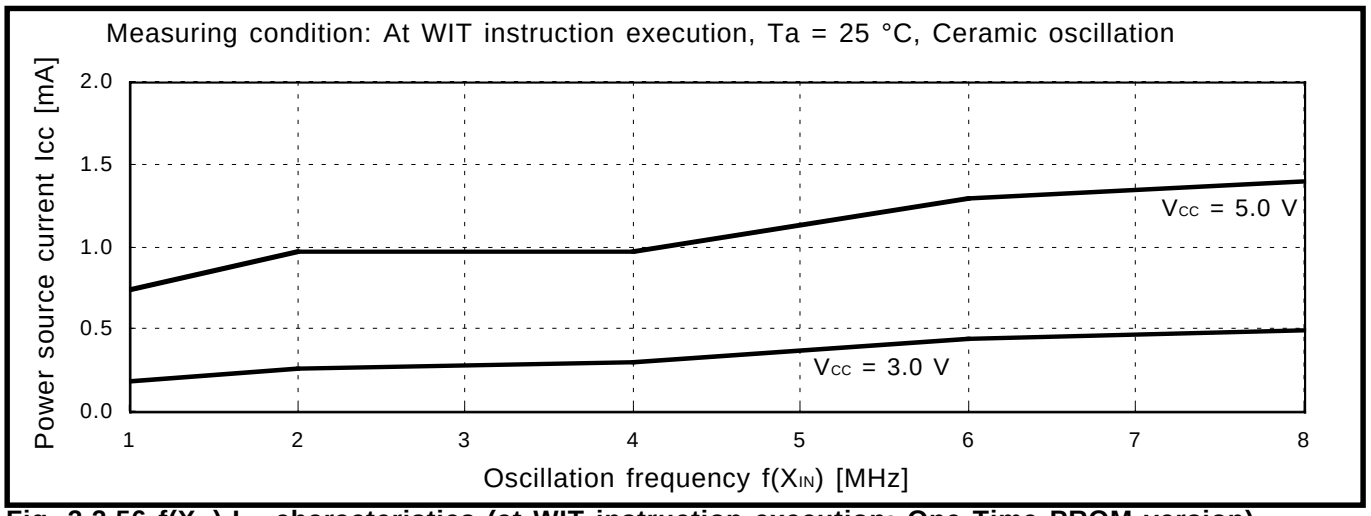


Fig. 3.2.56 f(X_{IN})-I_{CC} characteristics (at WIT instruction execution: One Time PROM version)

(3) Power source current characteristic example (Ta-I_{CC} characteristics)

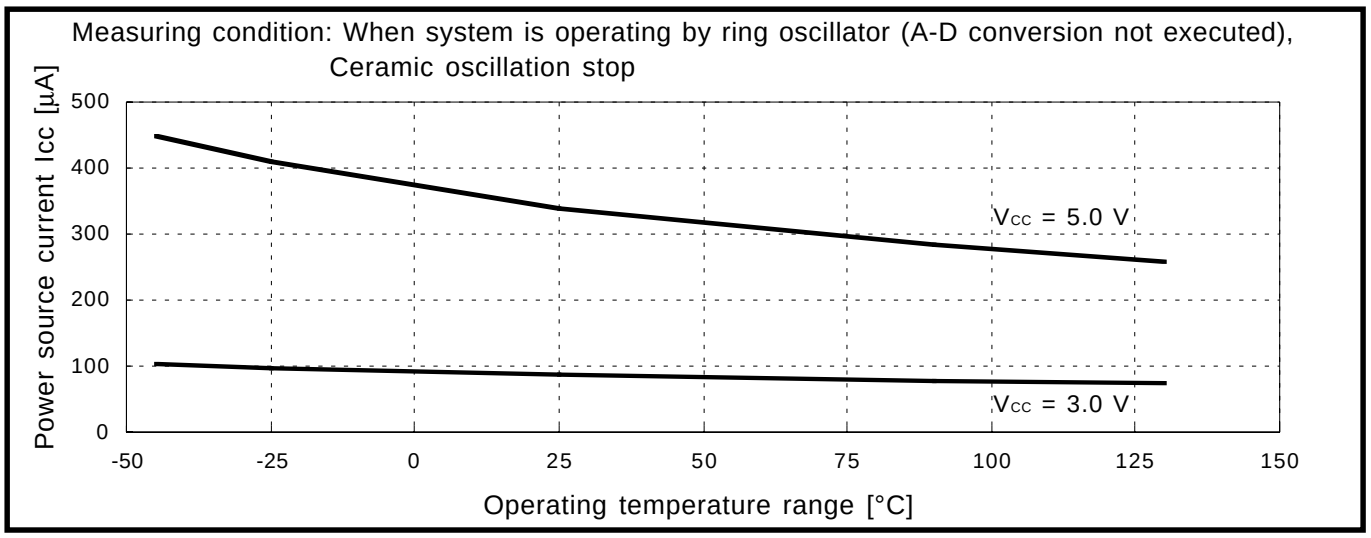


Fig. 3.2.57 Ta-I_{CC} characteristics (When system is operating by ring oscillator, Ceramic oscillation stop: One Time PROM version)

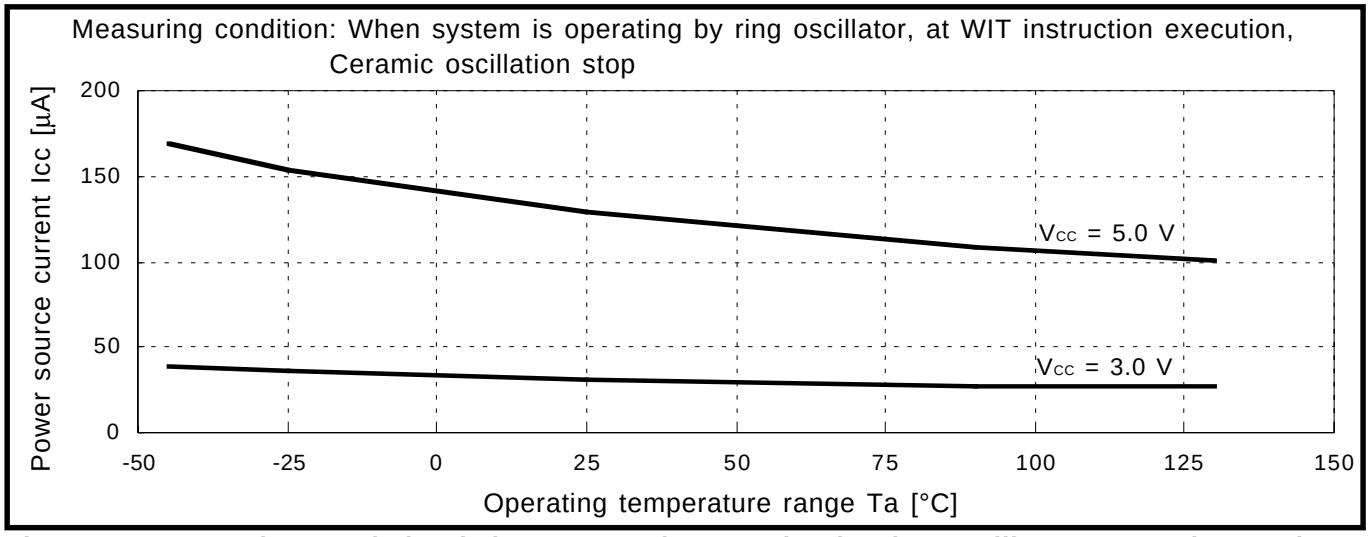


Fig. 3.2.58 Ta-I_{CC} characteristics (When system is operating by ring oscillator, at WIT instruction execution, Ceramic oscillation stop: One Time PROM version)

(4) Port typical characteristic example (V_{CC}-V_{IHL} characteristics)

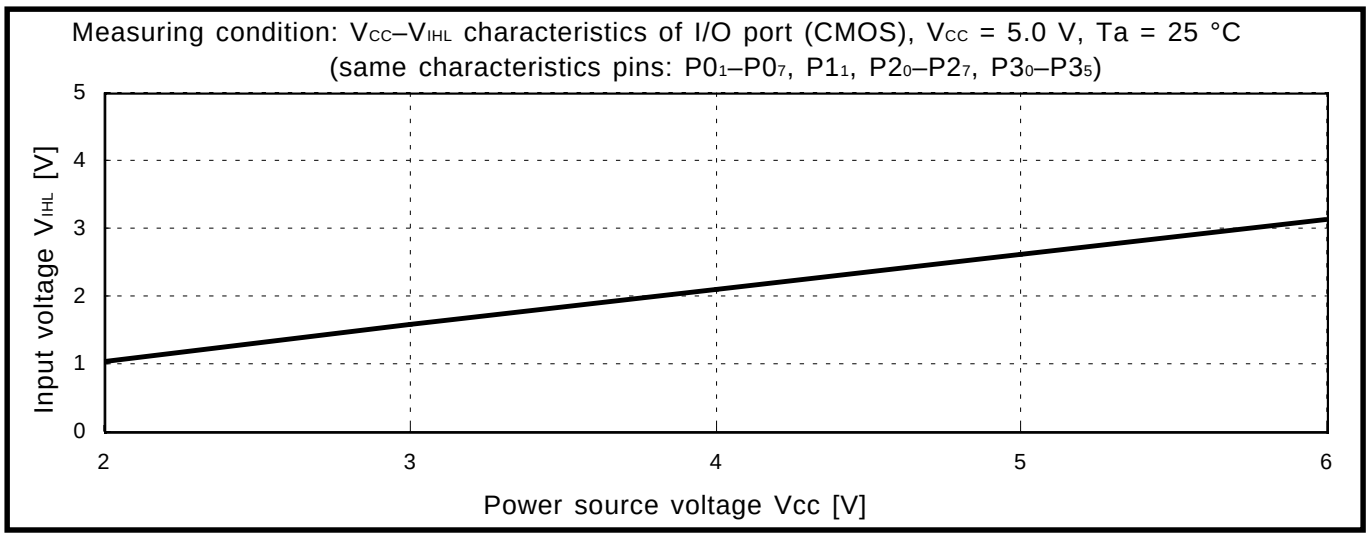


Fig. 3.2.59 V_{CC}-V_{IHL} characteristics (I/O port (CMOS): One Time PROM version)

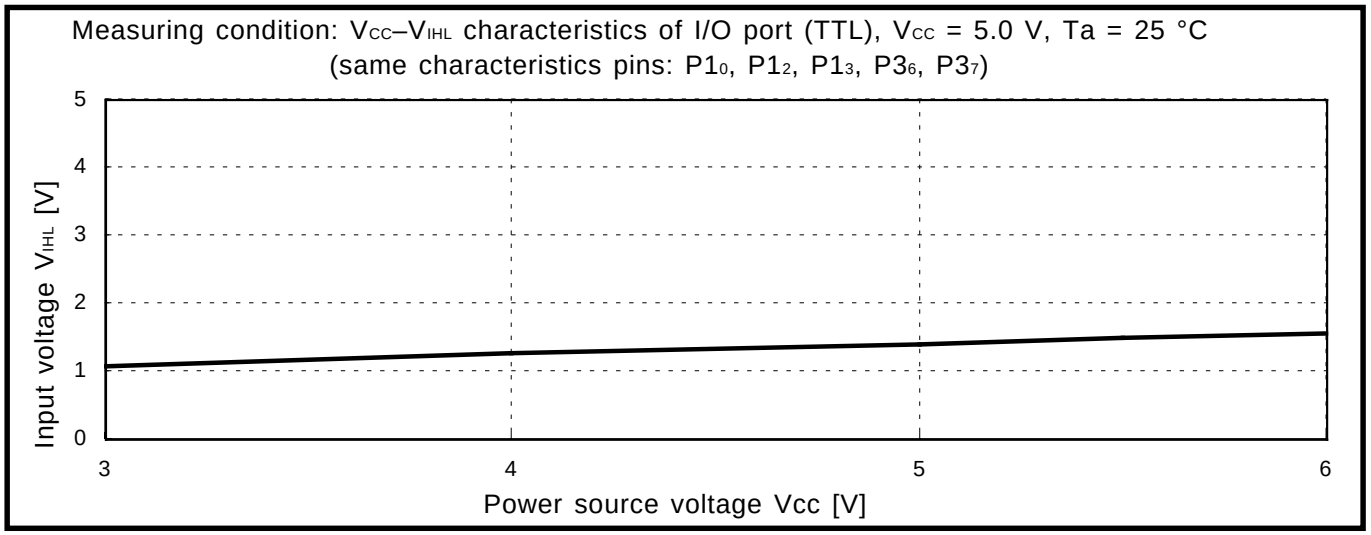


Fig. 3.2.60 V_{CC}-V_{IHL} characteristics (I/O port (TTL): One Time PROM version)

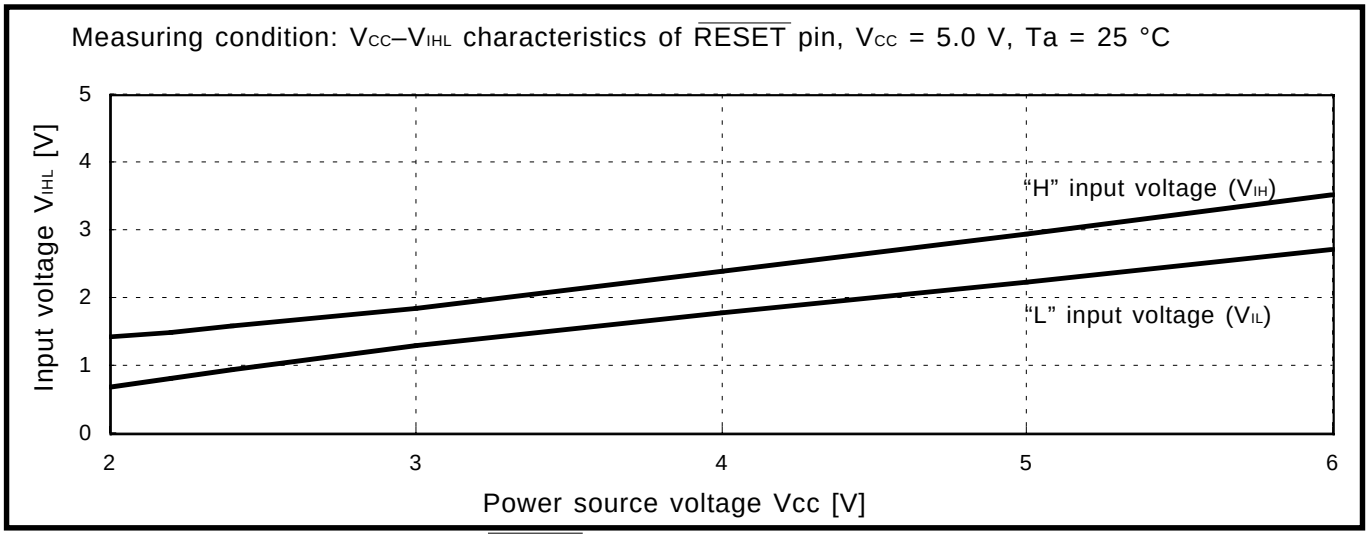


Fig. 3.2.61 V_{CC}-V_{IHL} characteristics ($\overline{\text{RESET}}$ pin: One Time PROM version)

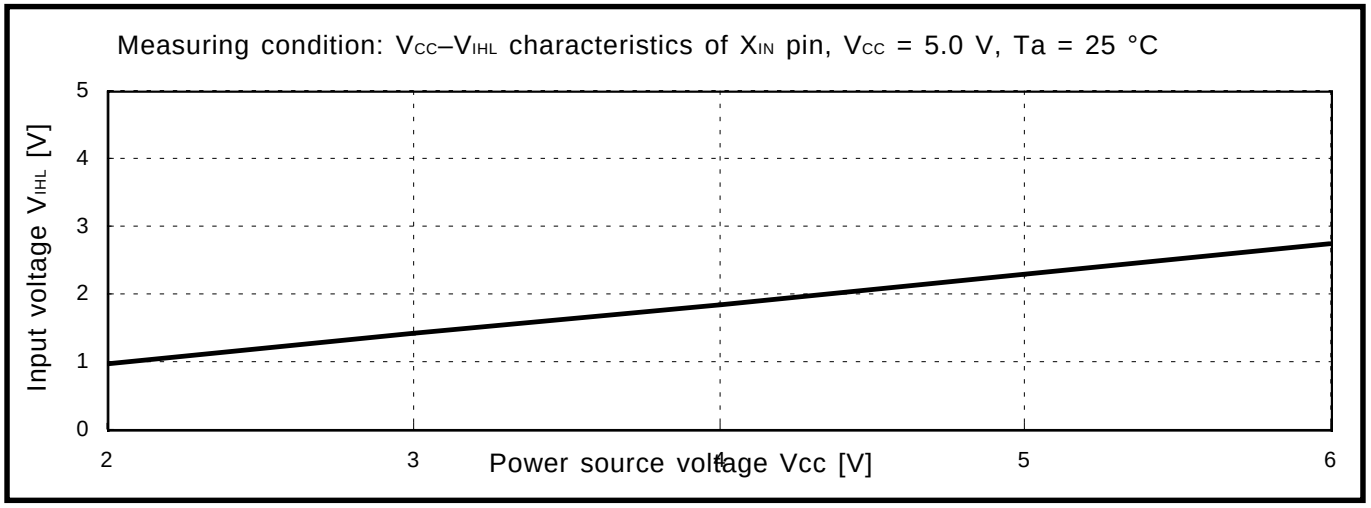


Fig. 3.2.62 V_{CC}-V_{IHL} characteristics (X_{IN} pin: One Time PROM version)

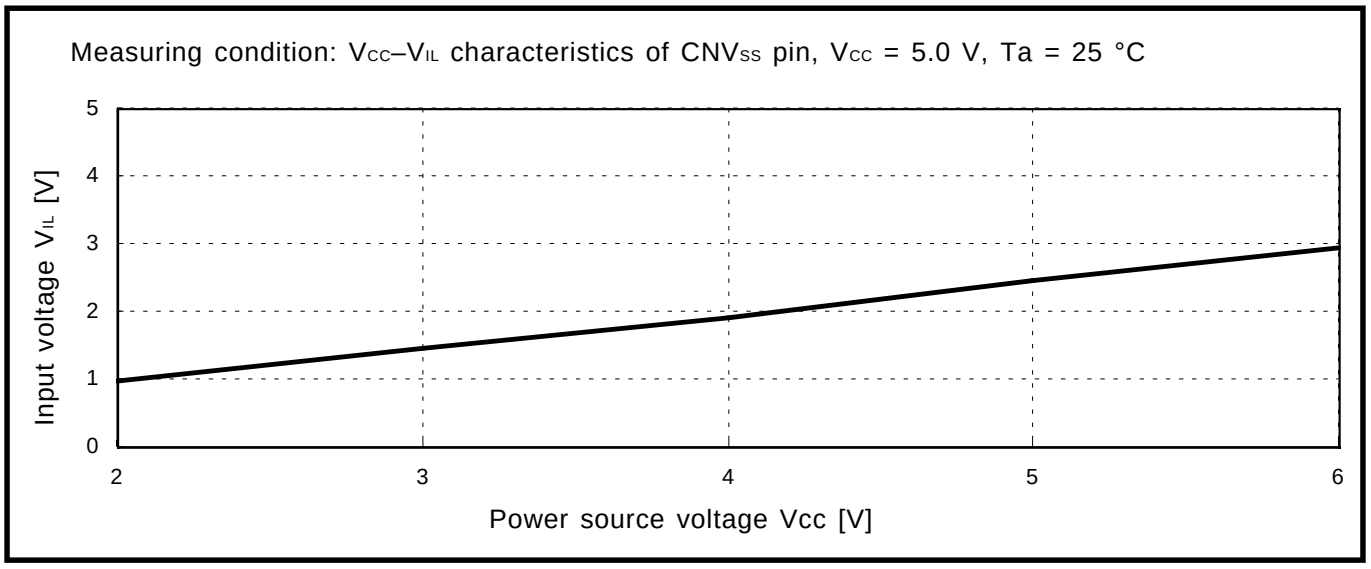


Fig. 3.2.63 V_{CC}-V_{IL} characteristics (CNV_{SS} pin: One Time PROM version)

APPENDIX

3.2 Typical characteristics

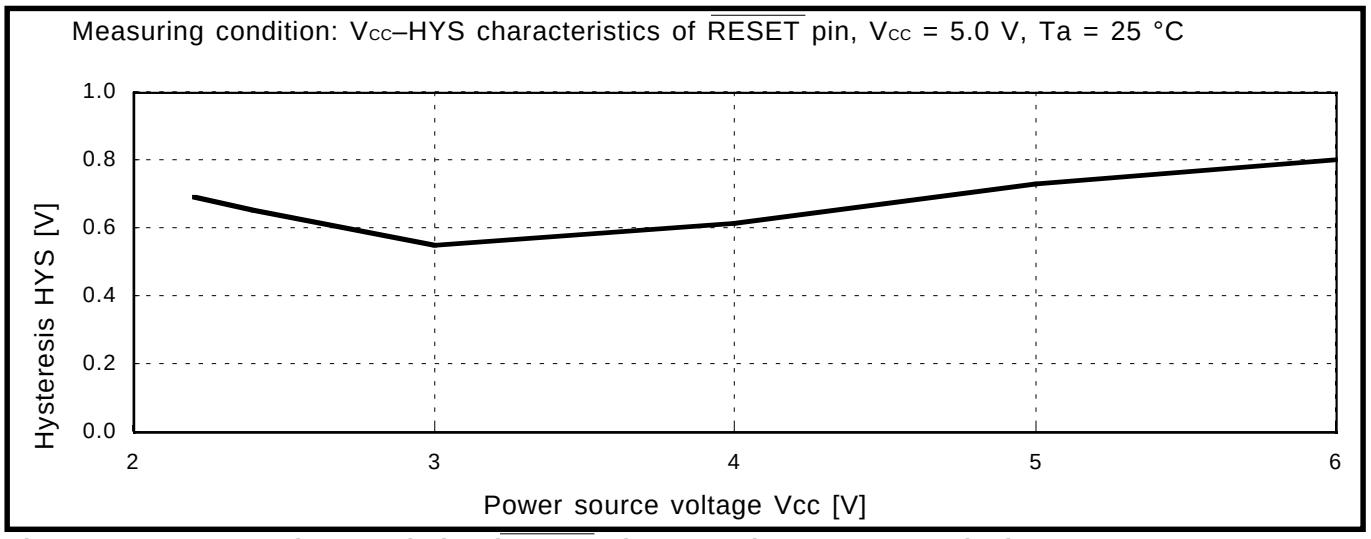


Fig. 3.2.64 Vcc-HYS characteristics (RESET pin: One Time PROM version)

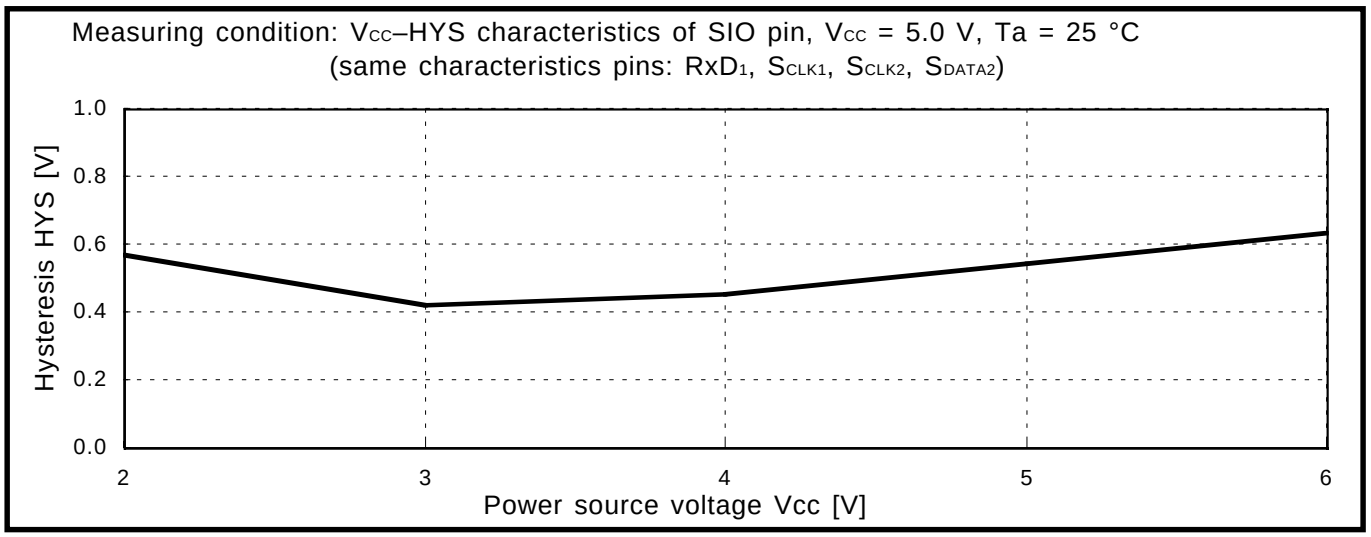


Fig. 3.2.65 Vcc-HYS characteristics (SIO pin: One Time PROM version)

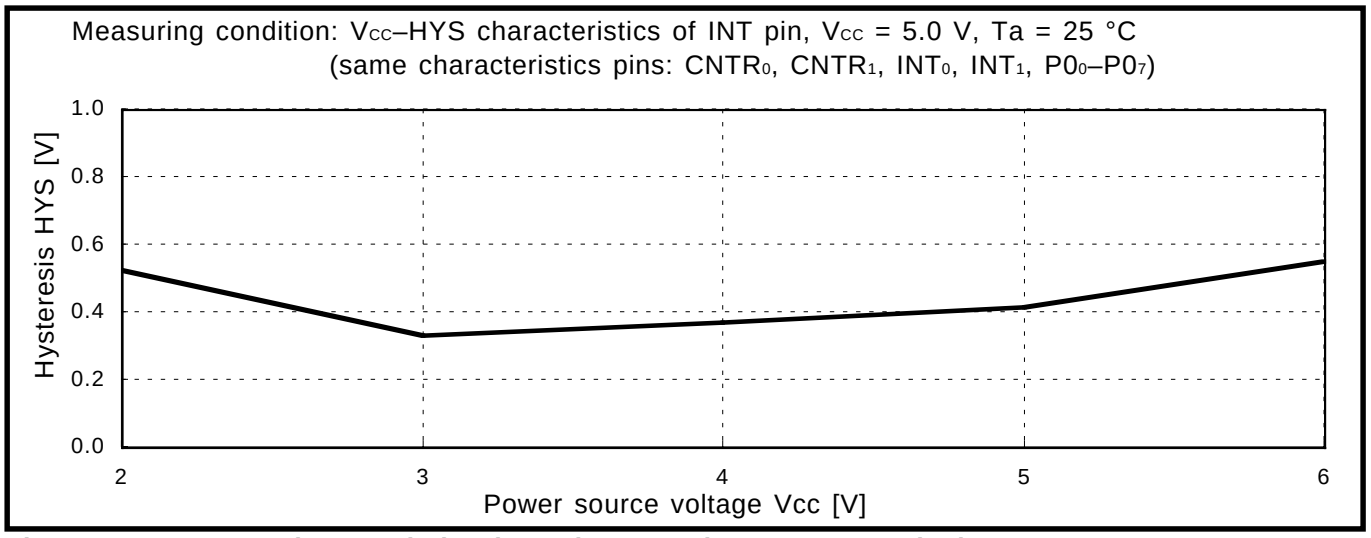


Fig. 3.2.66 Vcc-HYS characteristics (INT pin: One Time PROM version)

(5) Port typical characteristic example (V_{OH}-I_{OH} characteristics)

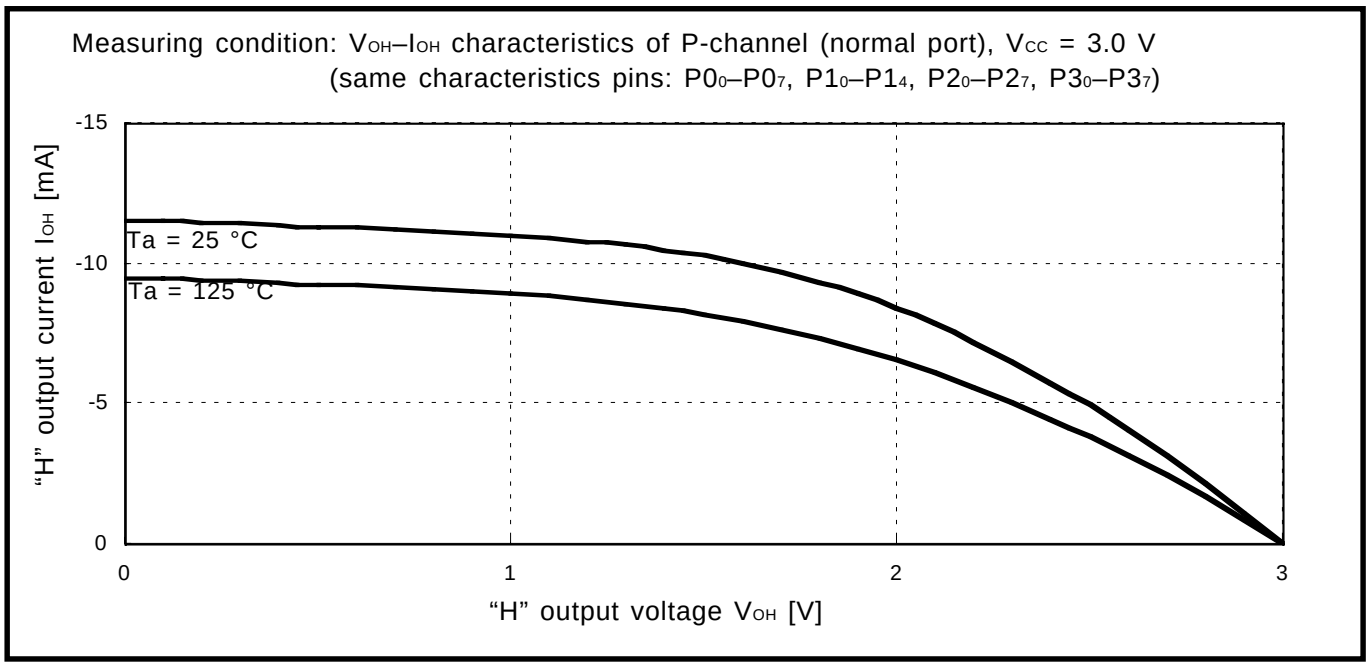


Fig. 3.2.67 V_{OH}-I_{OH} characteristics of P-channel (V_{CC} = 3.0 V, normal port: One Time PROM version)

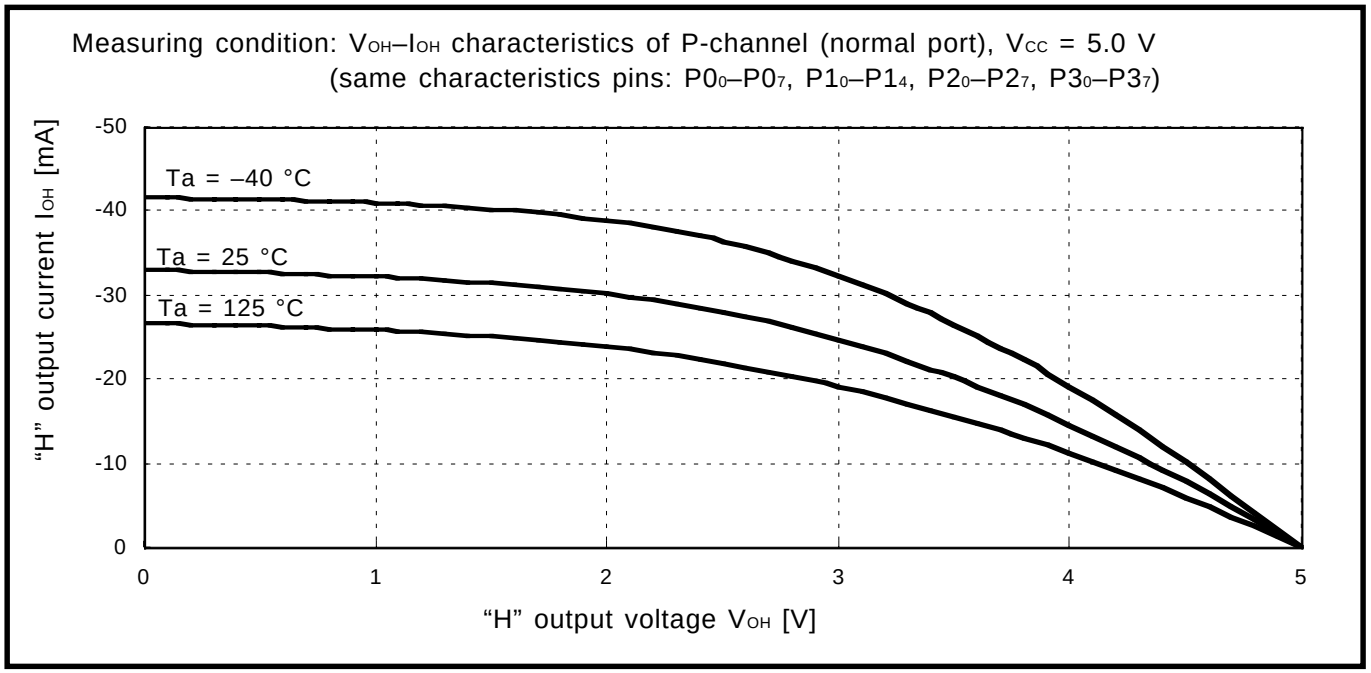


Fig. 3.2.68 V_{OH}-I_{OH} characteristics of P-channel (V_{CC} = 5.0 V, normal port: One Time PROM version)

(6) Port typical characteristic example (V_{OL}-I_{OL} characteristics)

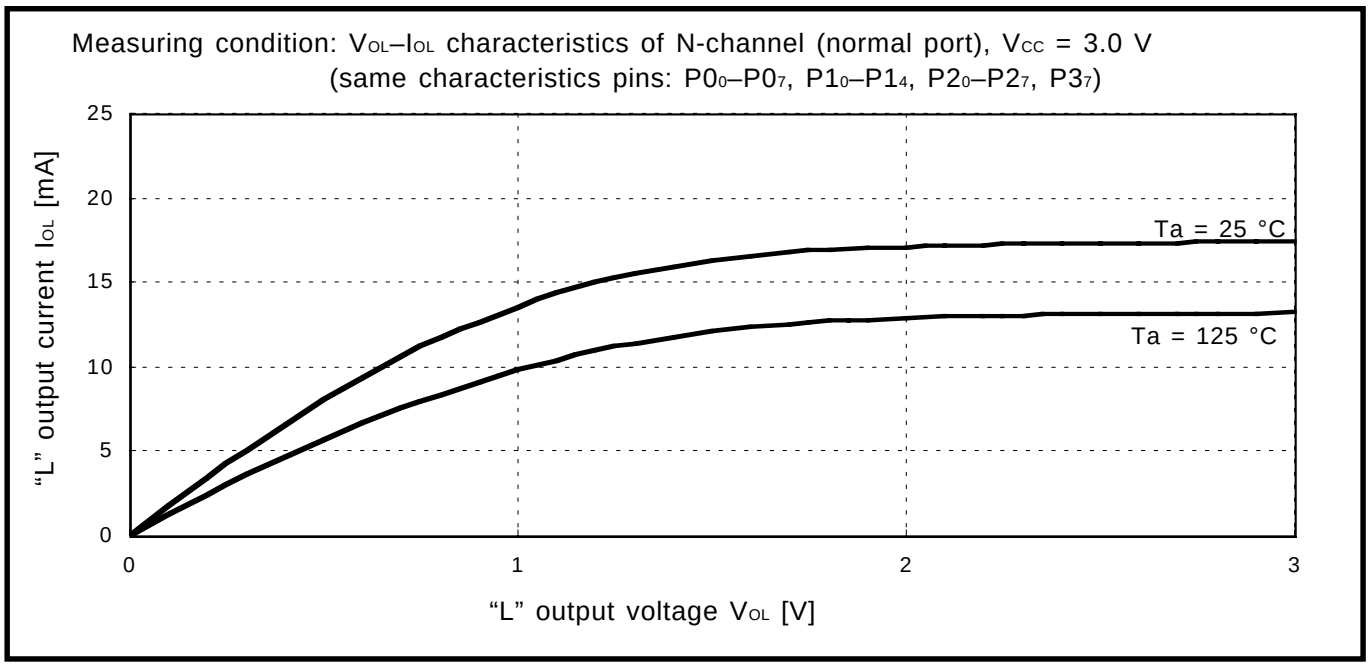


Fig. 3.2.69 V_{OL}-I_{OL} characteristics of N-channel (V_{CC} = 3.0 V, normal port: One Time PROM version)

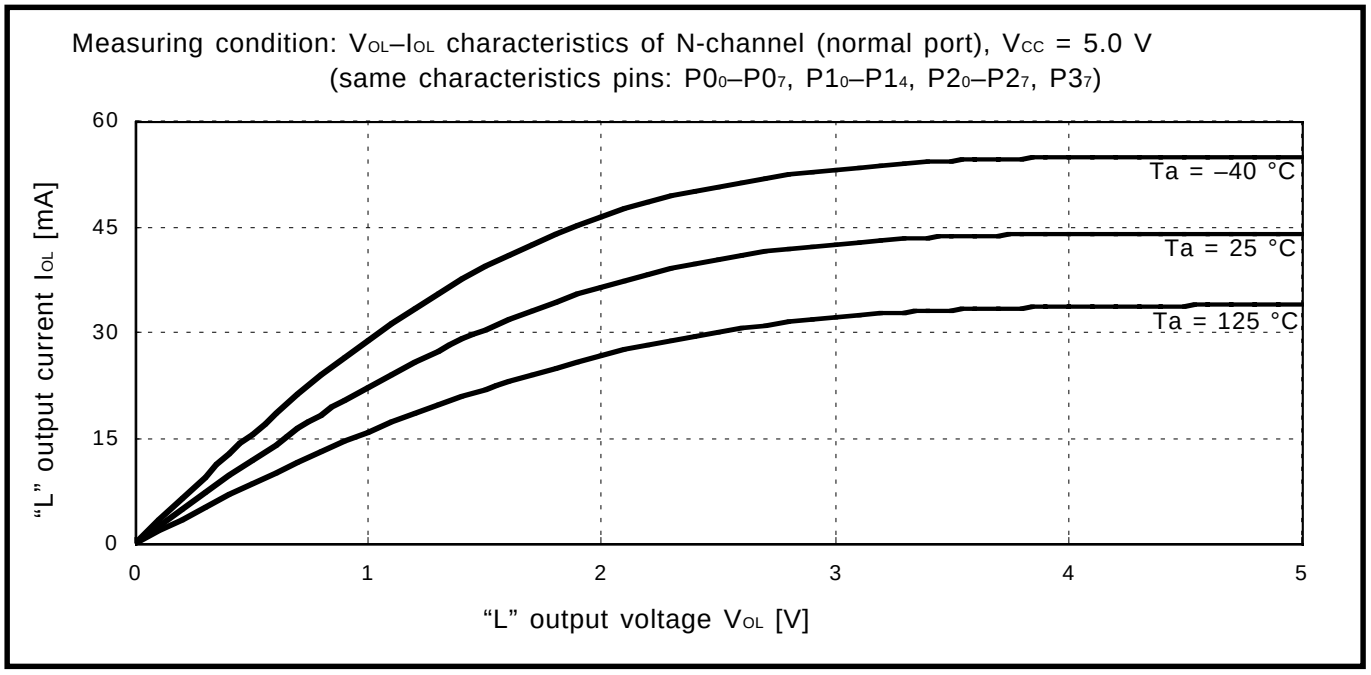


Fig. 3.2.70 V_{OL}-I_{OL} characteristics of N-channel (V_{CC} = 5.0 V, normal port: One Time PROM version)

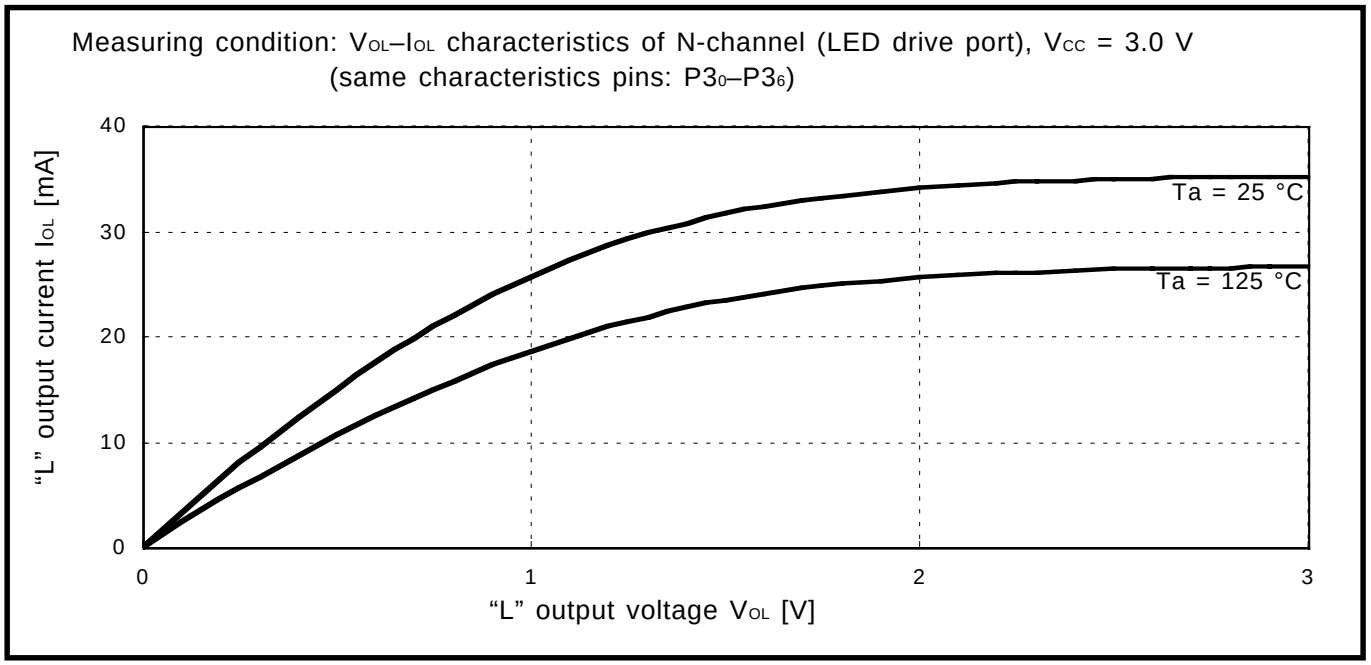


Fig. 3.2.71 V_{OL}-I_{OL} characteristics of N-channel (V_{CC} = 3.0 V, LED drive port: One Time PROM version)

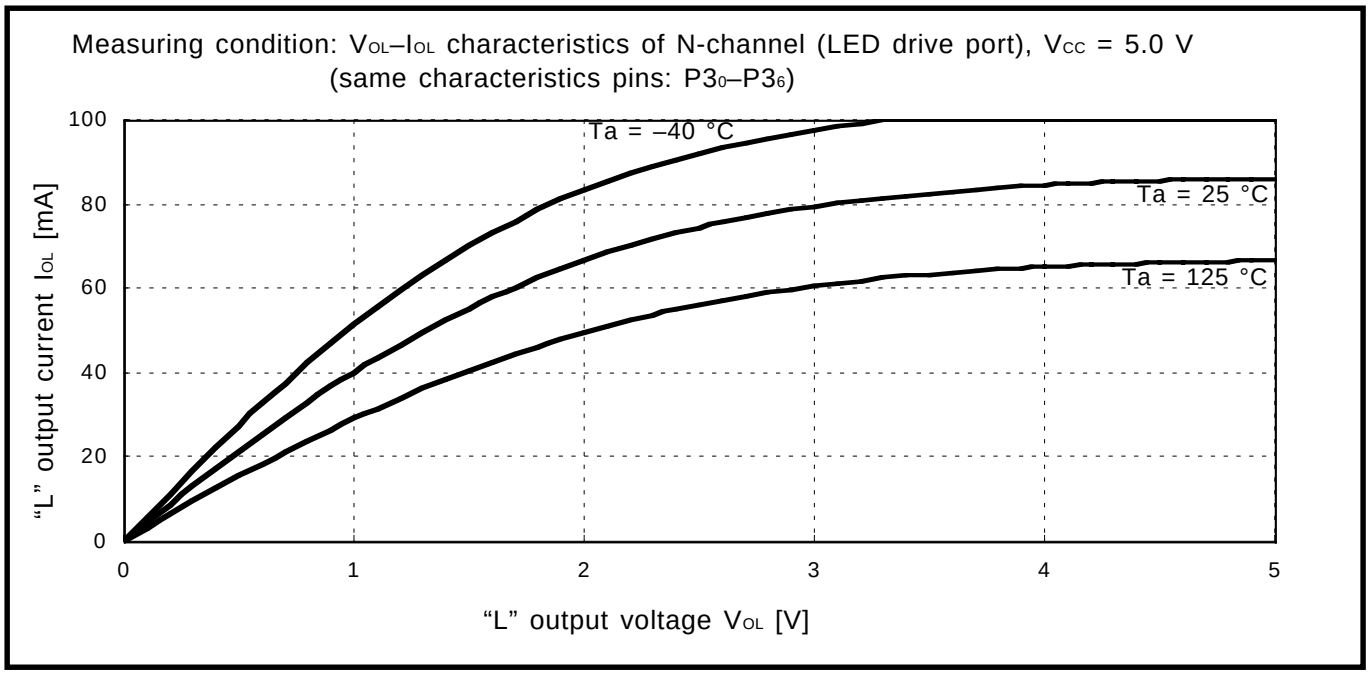


Fig. 3.2.72 V_{OL}-I_{OL} characteristics of N-channel (V_{CC} = 5.0 V, LED drive port: One Time PROM version)

(7) Port typical characteristic example (V_{CC}-IIL characteristics)

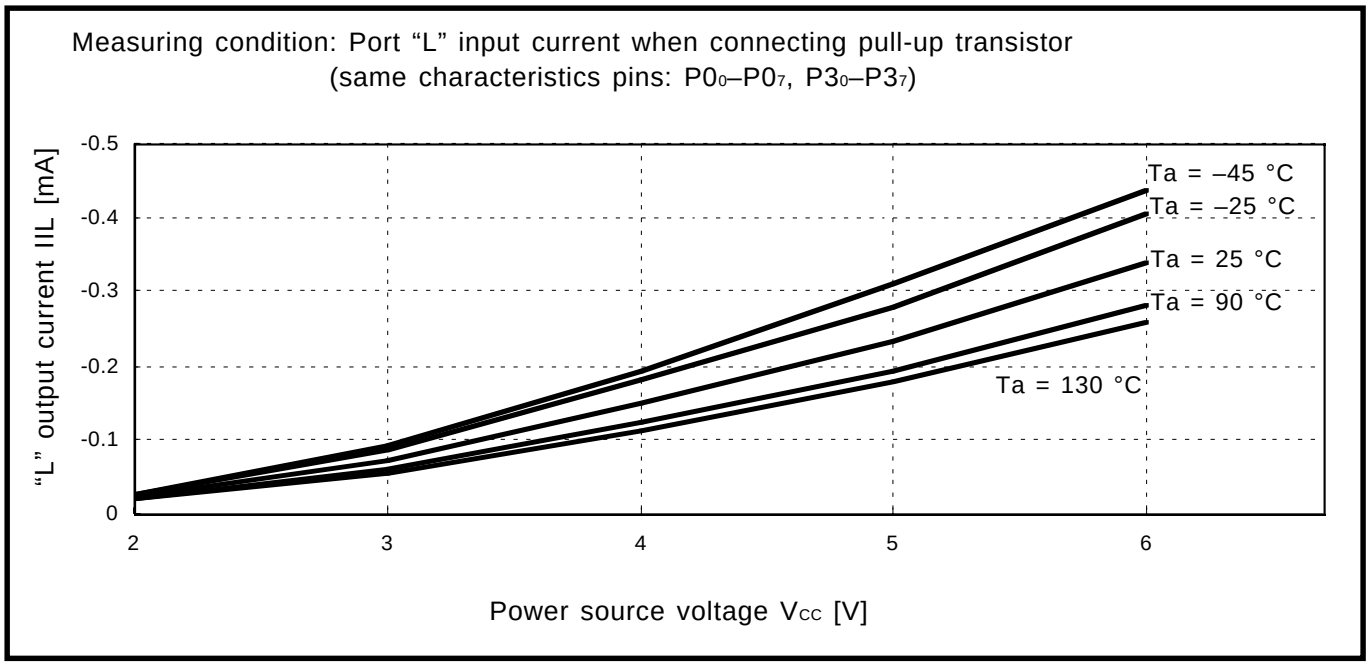


Fig. 3.2.73 V_{CC}-IIL characteristics (Port “L” input current when connecting pull-up transistor: One Time PROM version)

(8) Port typical characteristic example (V_{IN}-II(AD) characteristics)

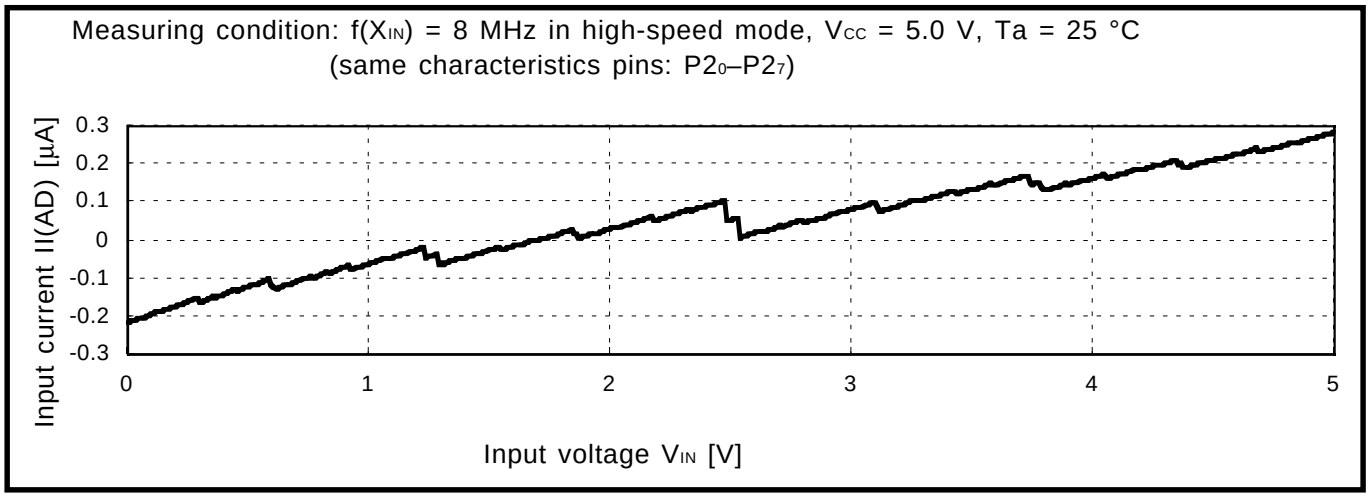


Fig. 3.2.74 V_{IN}-II(AD) characteristics (A-D port input current during A-D conversion, f(X_{IN}) = 8 MHz in high-speed mode: One Time PROM version)

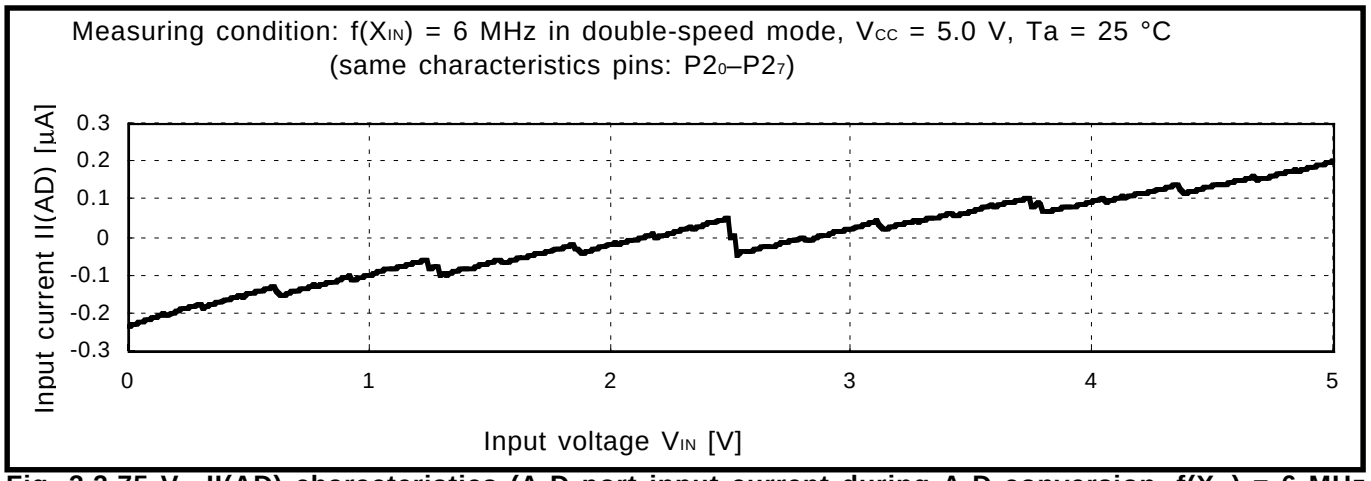


Fig. 3.2.75 V_{IN}-II(AD) characteristics (A-D port input current during A-D conversion, f(X_{IN}) = 6 MHz in double-speed mode: One Time PROM version)

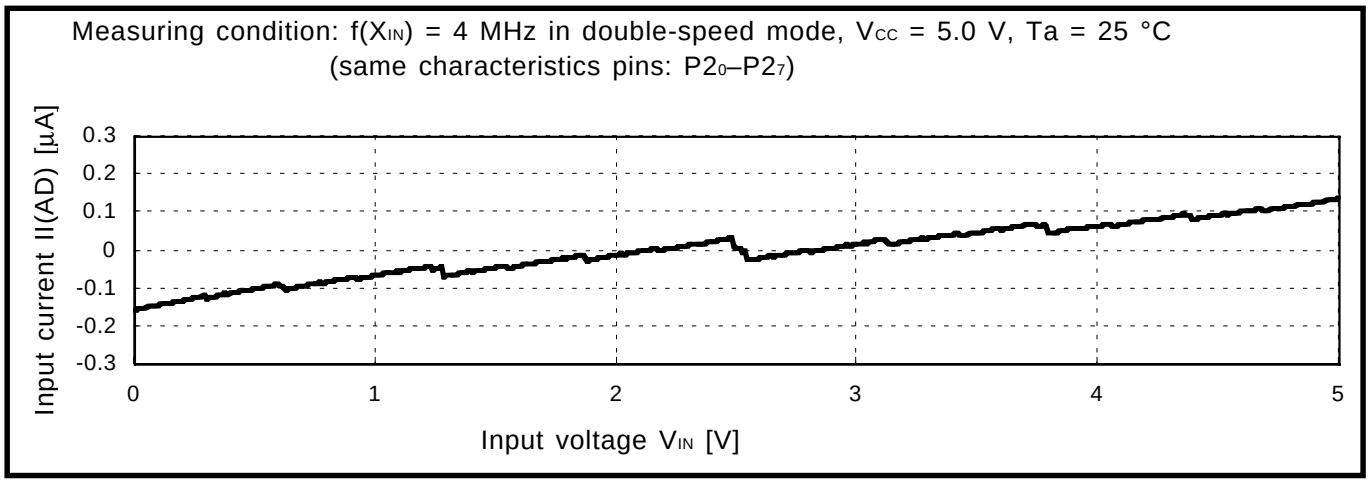


Fig. 3.2.76 V_{IN}-II(AD) characteristics (A-D port input current during A-D conversion, f(X_{IN}) = 4 MHz in double-speed mode: One Time PROM version)

(9) Ring oscillator frequency typical characteristic example

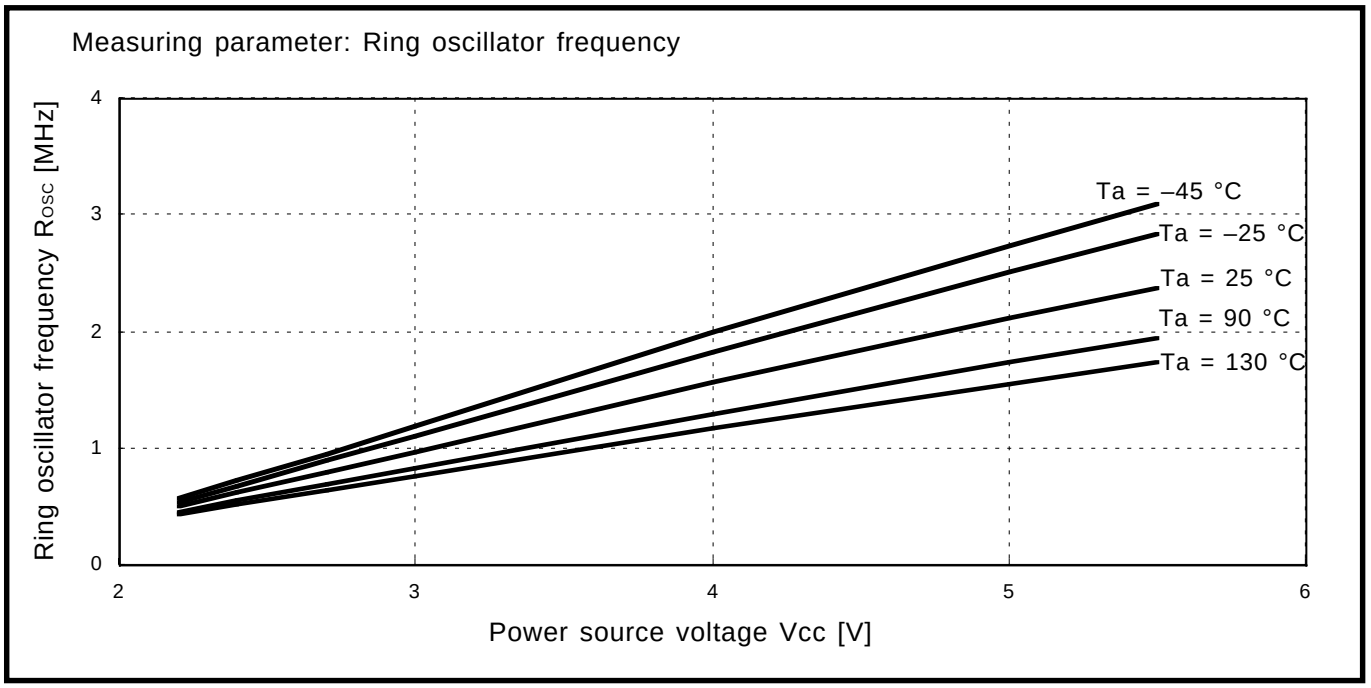


Fig. 3.2.77 V_{CC}-R_{OSC} characteristics (ring oscillator frequency: One Time PROM version)

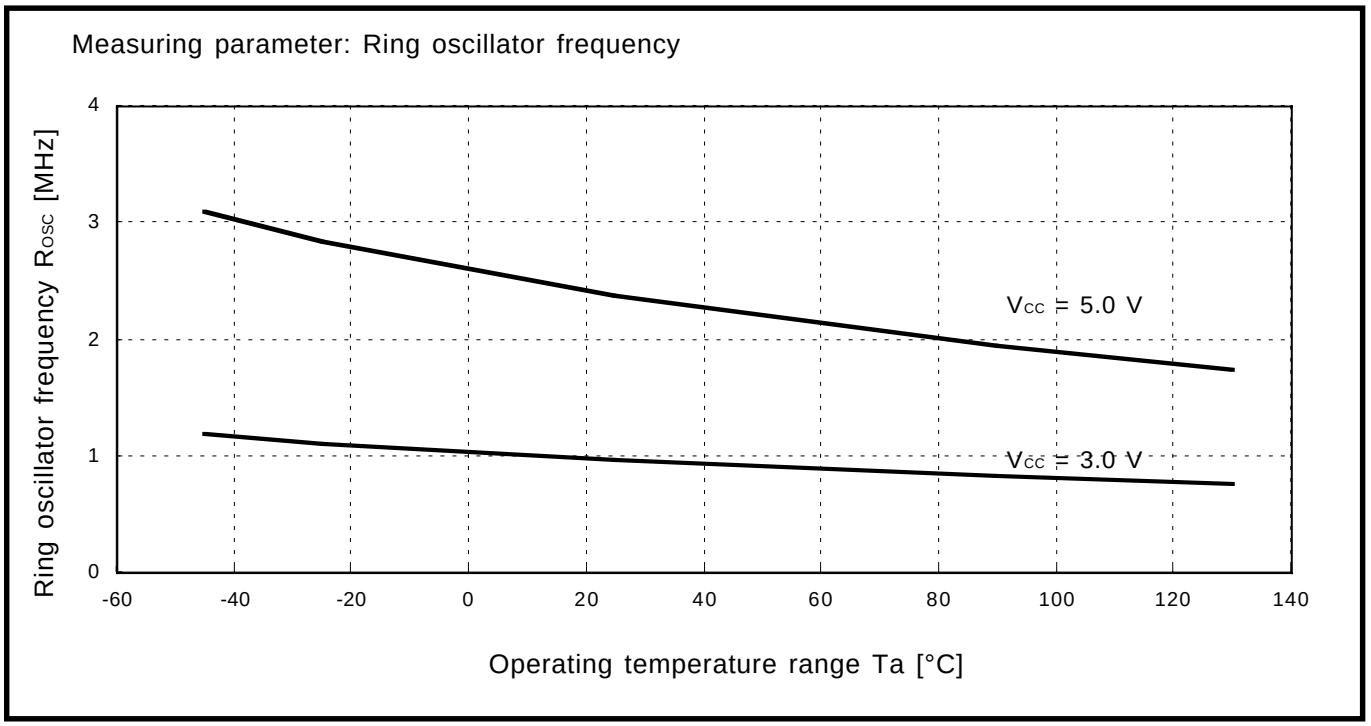


Fig. 3.2.78 Ta-R_{OSC} characteristics (ring oscillator frequency: One Time PROM version)

(10) RC oscillation frequency typical characteristic example

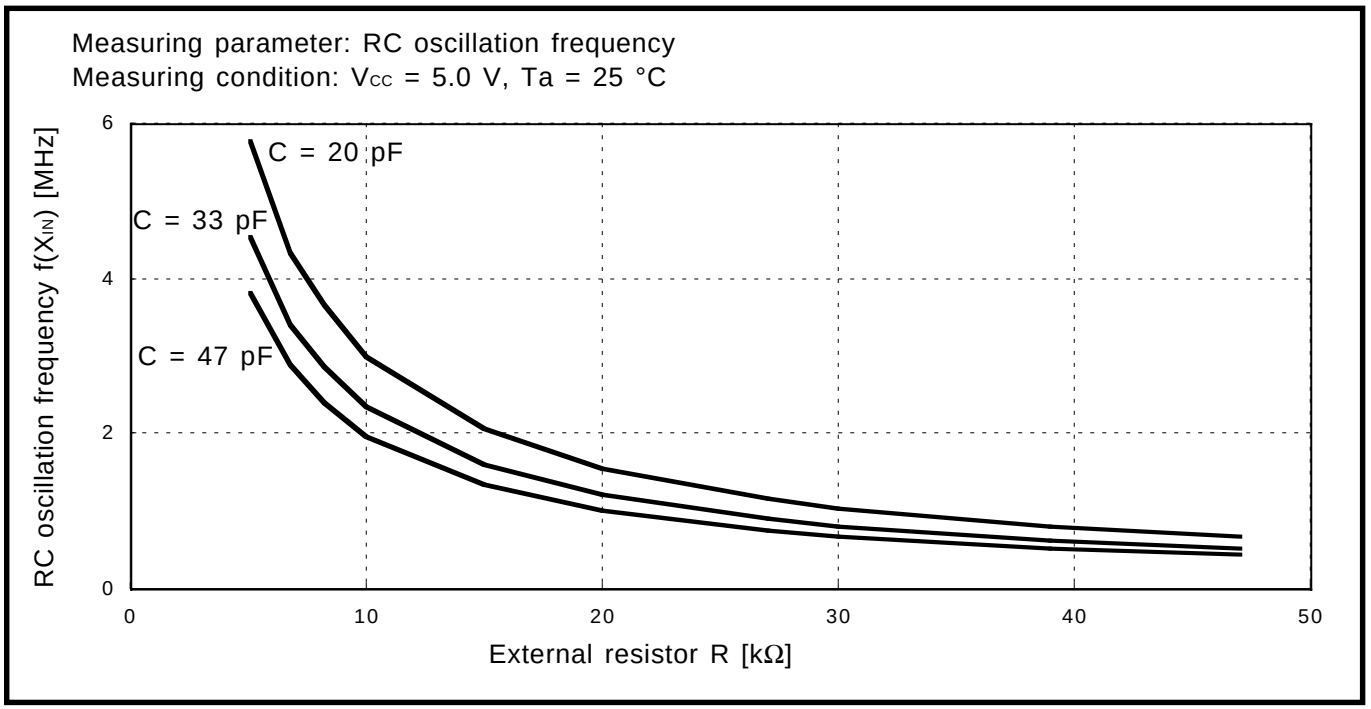


Fig. 3.2.79 R-f(X_{IN}) characteristics (RC oscillation frequency: One Time PROM version)

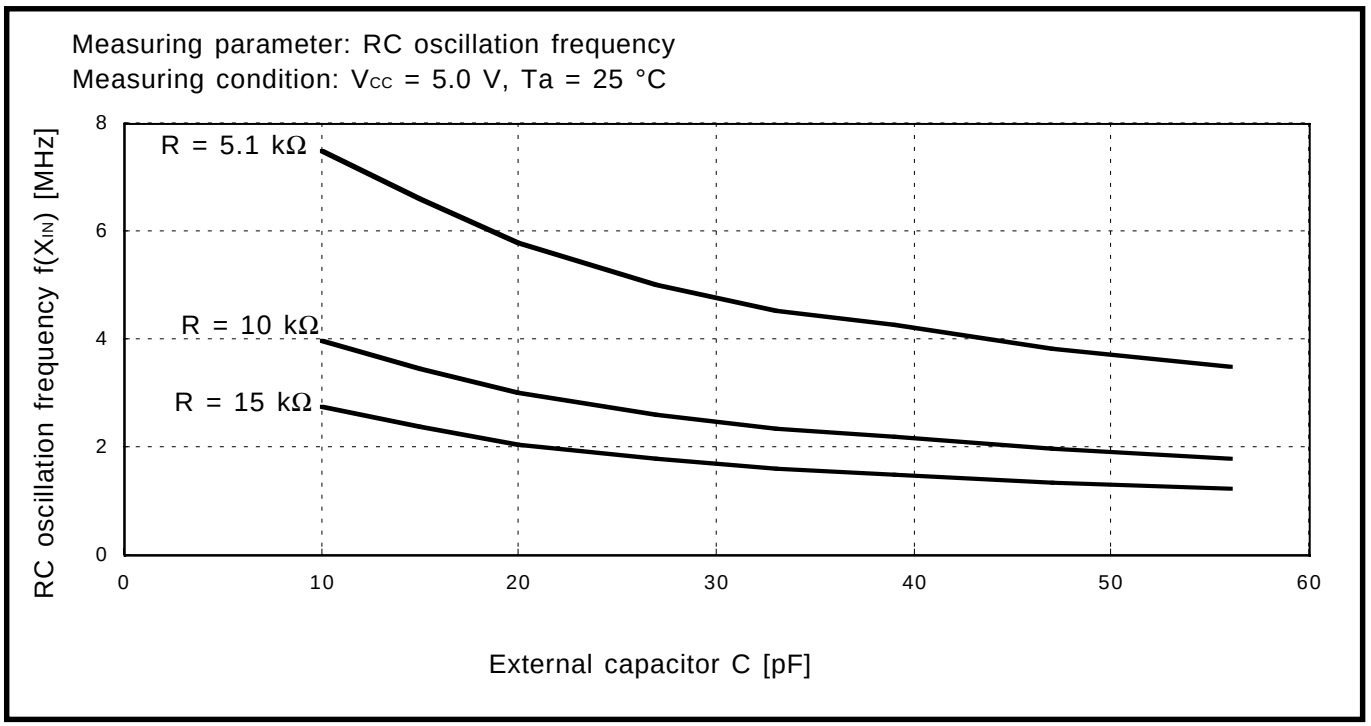


Fig. 3.2.80 C-f(X_{IN}) characteristics (RC oscillation frequency: One Time PROM version)

APPENDIX

3.2 Typical characteristics

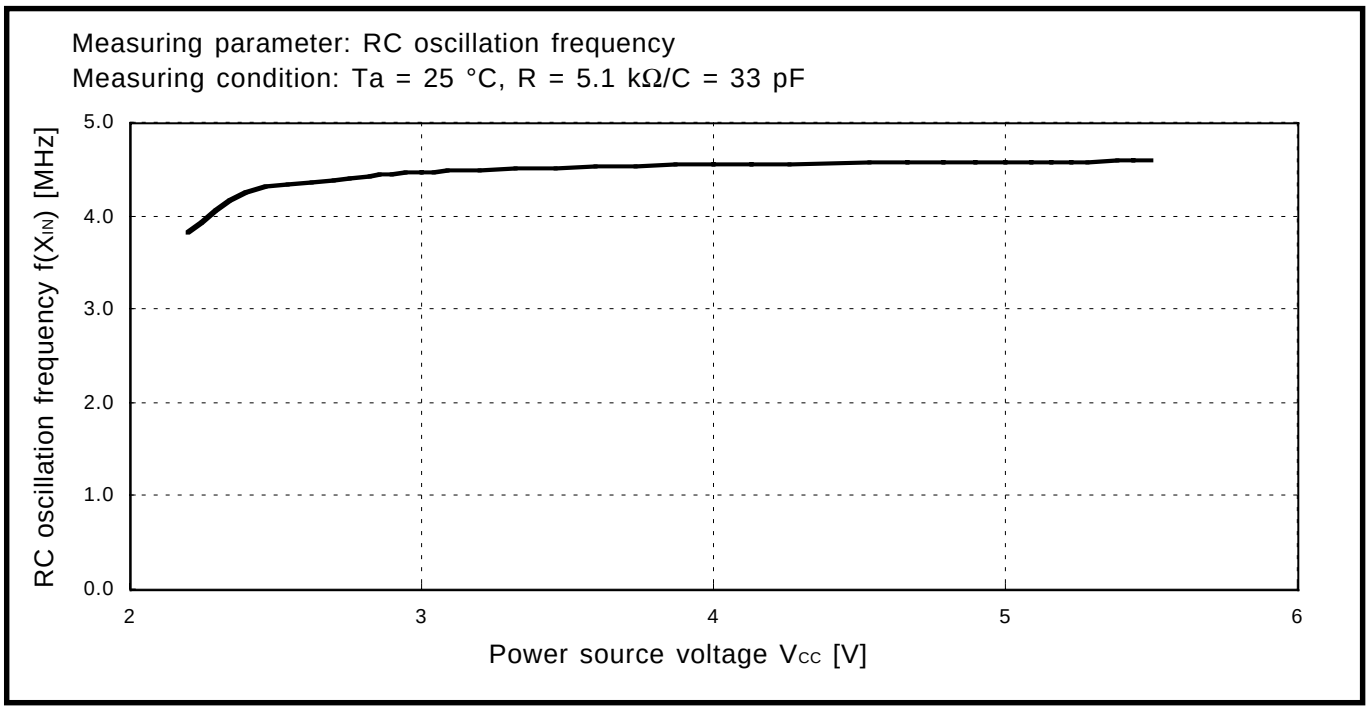


Fig. 3.2.81 V_{CC}-f(X_{IN}) characteristics (RC oscillation frequency: One Time PROM version)

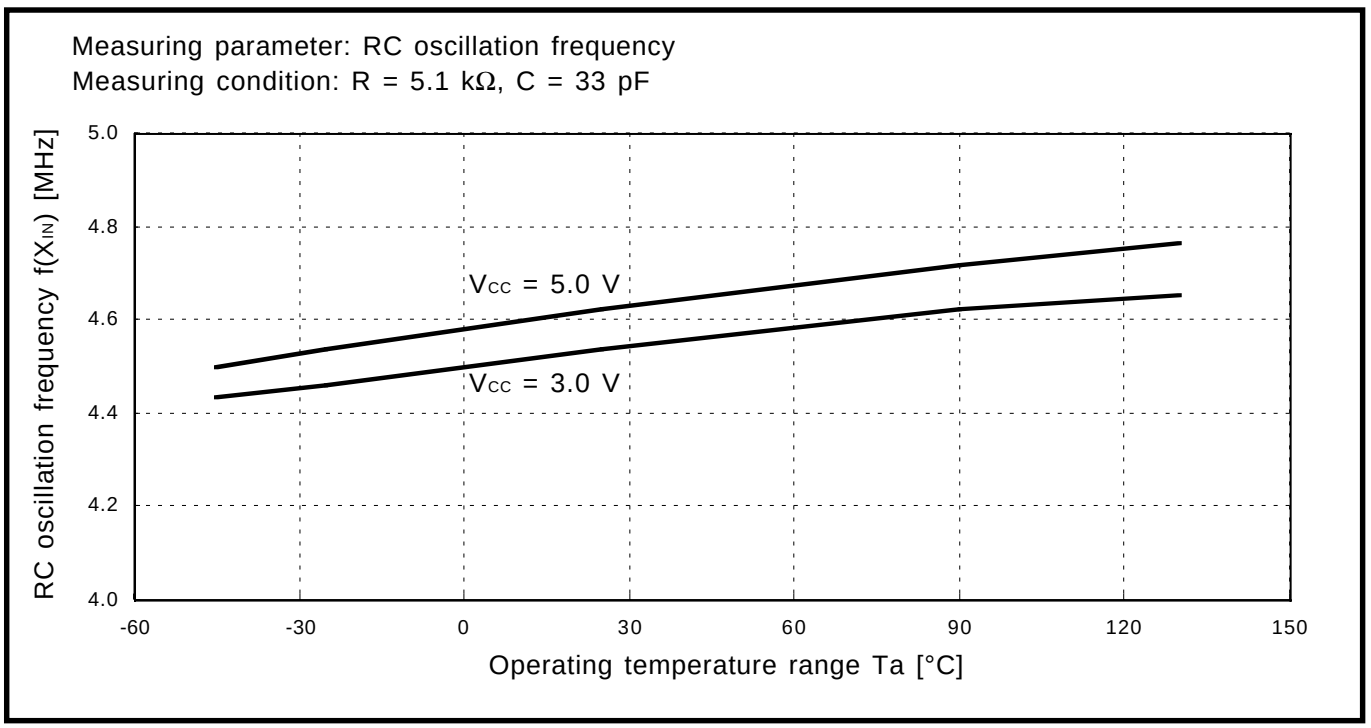


Fig. 3.2.82 Ta-f(X_{IN}) characteristics (RC oscillation frequency: One Time PROM version)

(11) A-D conversion typical characteristics example

① Definition of A-D conversion accuracy

The A-D conversion accuracy is defined below (refer to Fig. 3.2.83).

●Relative accuracy

- Zero transition voltage (V_{OT})
  This means an analog input voltage when the actual A-D conversion output data changes from "0" to "1."
- Full-scale transition voltage (V_{FST})
  This means an analog input voltage when the actual A-D conversion output data changes from "1023" to "1022."
- Non-linearity error
  This means a deviation from the line between V_{OT} and V_{FST} of a converted value between V_{OT} and V_{FST}.
- Differential non-linearity error
  This means a deviation from the input potential difference required to change a converted value between V_{OT} and V_{FST} by 1 LSB of the 1 LSB at the relative accuracy.

●Absolute accuracy

This means a deviation from the ideal characteristics between 0 to V_{REF} of actual A-D conversion characteristics.

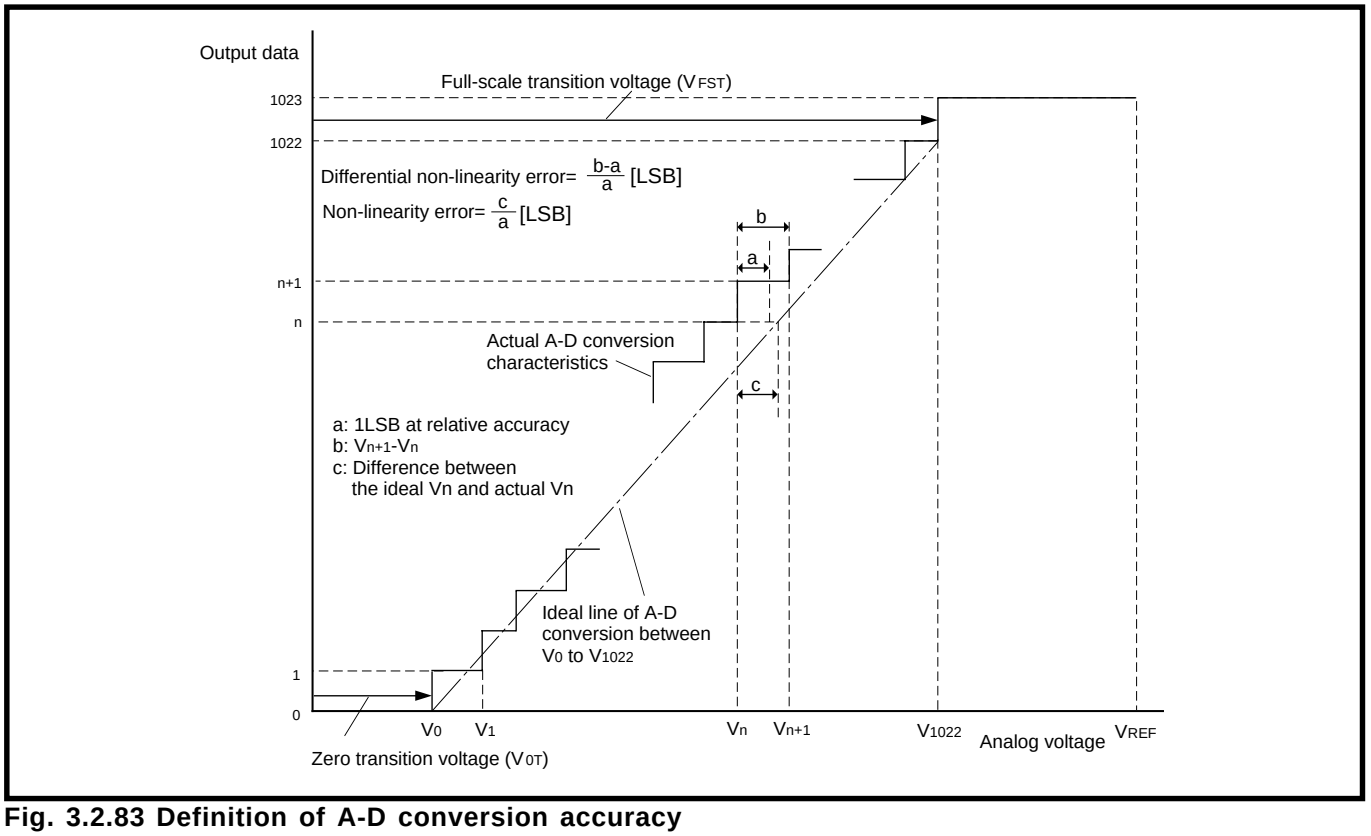


Fig. 3.2.83 Definition of A-D conversion accuracy

Vn: Analog input voltage when the output data changes from "n" to "n + 1" (n = 0 to 1022)

- 1 LSB at relative accuracy $\rightarrow \dfrac{V_{FST} - V_{OT}}{1022}$ (V)
- 1 LSB at absolute accuracy $\rightarrow \dfrac{V_{REF}}{1024}$ (V)

② A-D conversion accuracy typical characteristics-1

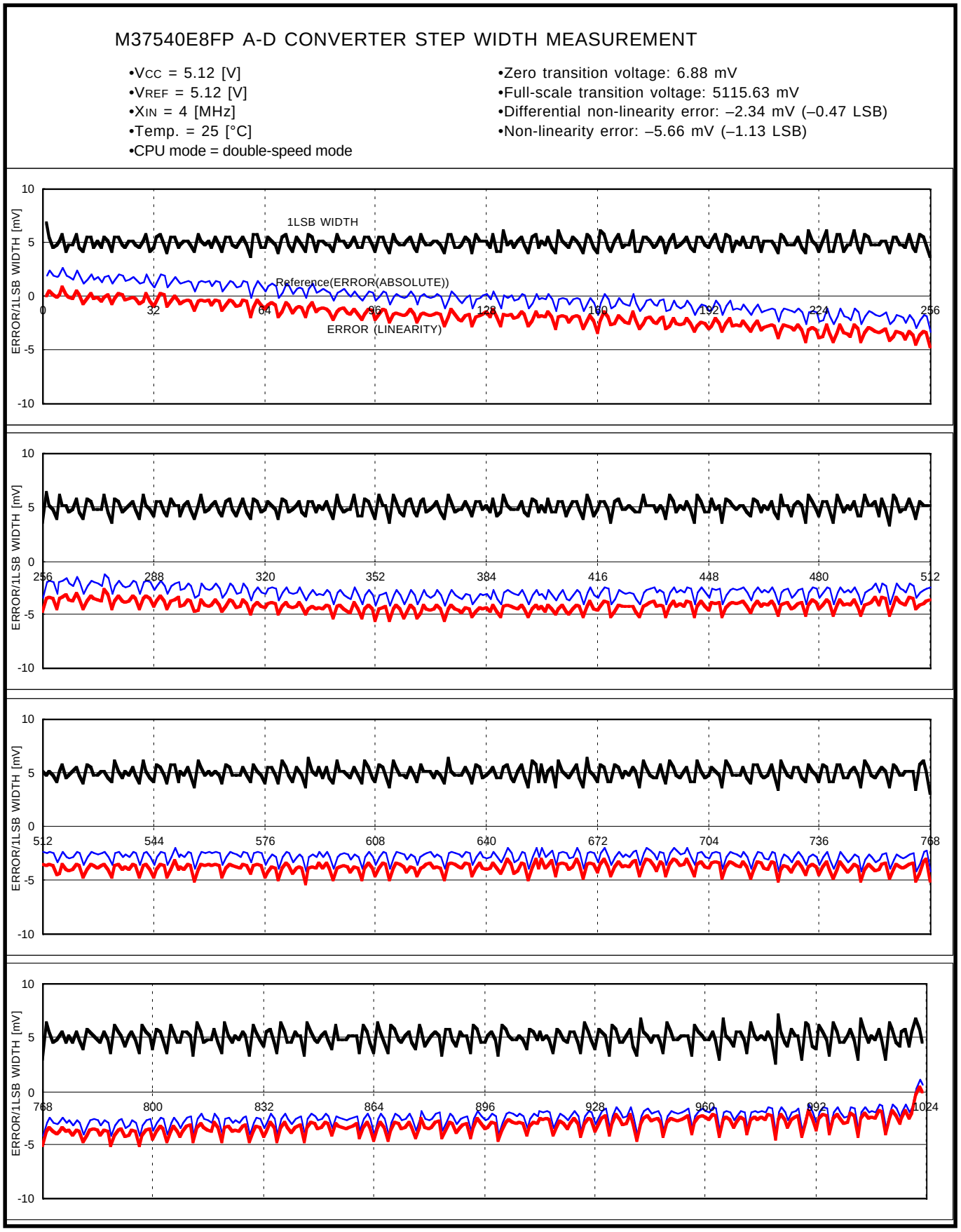


Fig. 3.2.84 A-D conversion accuracy typical characteristic example-1 (One Time PROM version)

③ A-D conversion accuracy typical characteristics-2

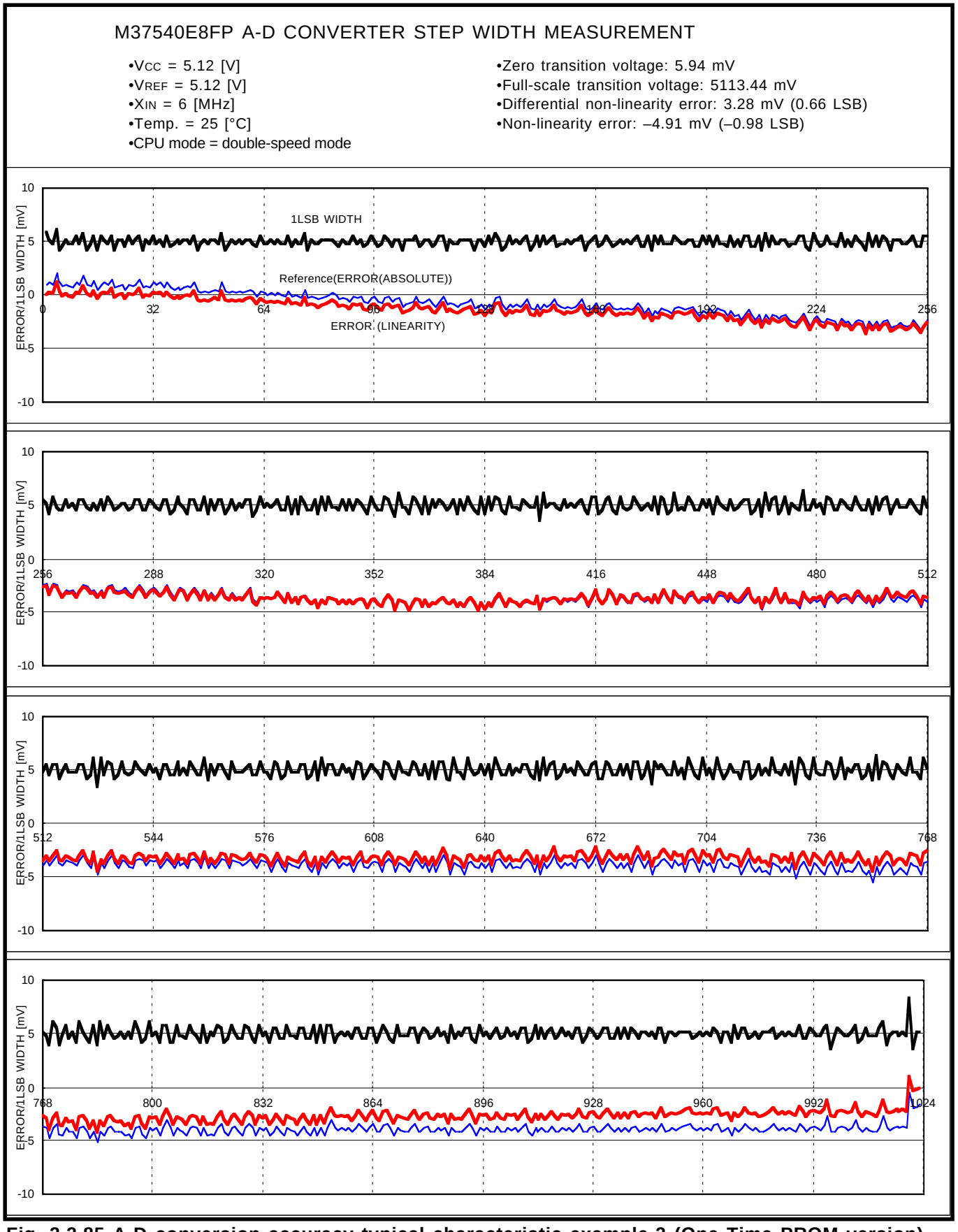


Fig. 3.2.85 A-D conversion accuracy typical characteristic example-2 (One Time PROM version)

④ A-D conversion accuracy typical characteristics-3

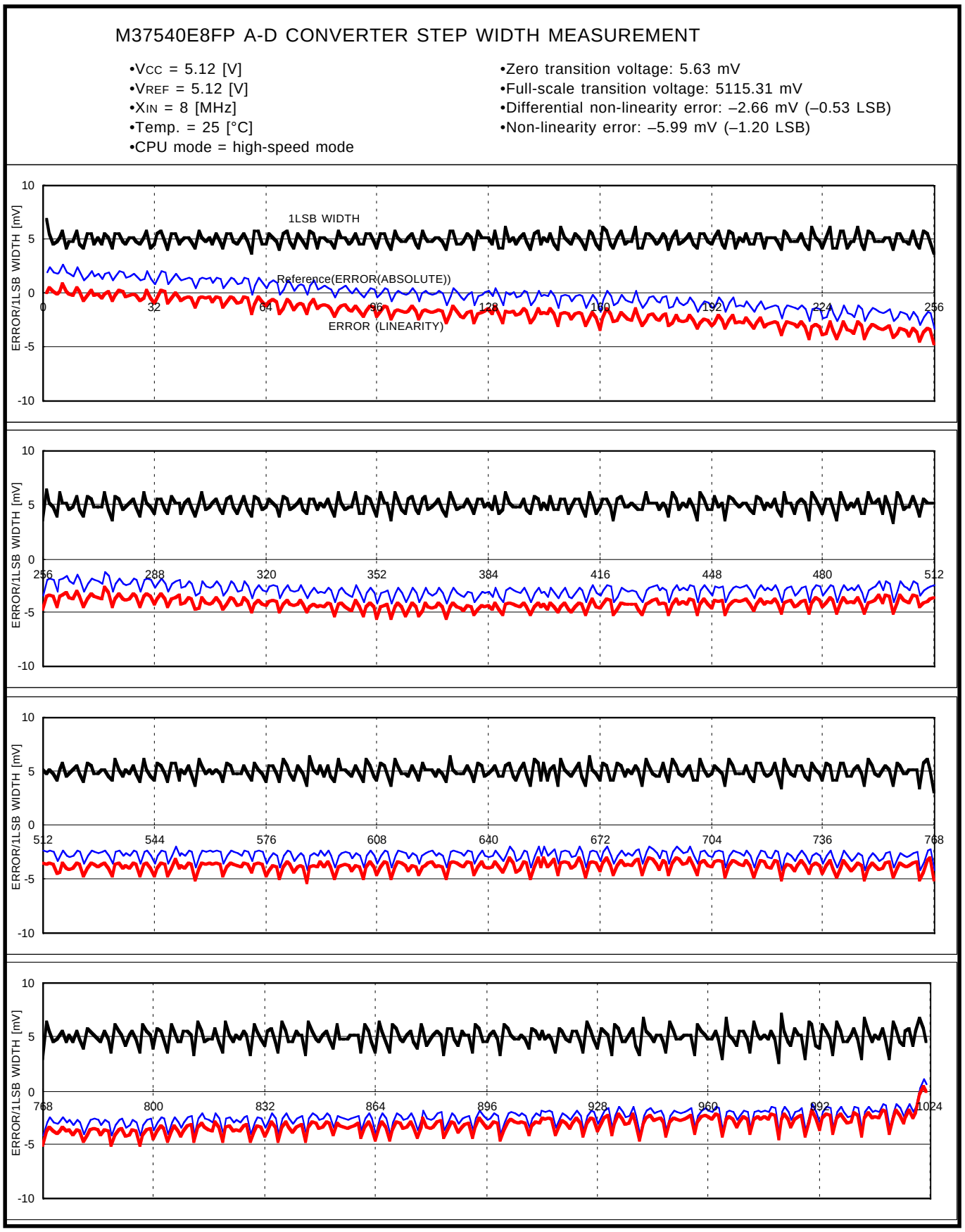


Fig. 3.2.86 A-D conversion accuracy typical characteristic example-3 (One Time PROM version)

3.3 Notes on use

3.3.1 Notes on input and output ports

Notes on using input and output ports are described below.

(1) Notes in stand-by state

In stand-by state*1 for low-power dissipation, do not make input levels of an input port and an I/O port "undefined".

Pull-up (connect the port to V_{CC}) or pull-down (connect the port to V_{SS}) these ports through a resistor.

When determining a resistance value, note the following points:

- External circuit
- Variation of output levels during the ordinary operation

When using a built-in pull-up resistor, note on varied current values:

- When setting as an input port : Fix its input level
- When setting as an output port : Prevent current from flowing out to external.

● **Reason**

The output transistor becomes the OFF state, which causes the ports to be the high-impedance state. Note that the level becomes "undefined" depending on external circuits.

Accordingly, the potential which is input to the input buffer in a microcomputer is unstable in the state that input levels of a input port and an I/O port are "undefined". This may cause power source current.

*1 stand-by state : the stop mode by executing the **STP** instruction
the wait mode by executing the **WIT** instruction

(2) Modifying output data with bit managing instruction

When the port latch of an I/O port is modified with the bit managing instruction*2, the value of the unspecified bit may be changed.

● **Reason**

The bit managing instructions are read-modify-write form instructions for reading and writing data by a byte unit. Accordingly, when these instructions are executed on a bit of the port latch of an I/O port, the following is executed to all bits of the port latch.

- As for a bit which is set for an input port :
  The pin state is read in the CPU, and is written to this bit after bit managing.
- As for a bit which is set for an output port :
  The bit value of the port latch is read in the CPU, and is written to this bit after bit managing.

Note the following :

- Even when a port which is set as an output port is changed for an input port, its port latch holds the output data.
- As for a bit of the port latch which is set for an input port, its value may be changed even when not specified with a bit managing instruction in case where the pin state differs from its port latch contents.

*2 bit managing instructions : **SEB**, and **CLB** instructions

(3) Usage for the 32-pin version

① Fix the $P3_5$, $P3_6$ pull-up control bit of the pull-up control register to "1".
② Keep the $P3_6/INT_1$ input level selection bit of the port P1P3 control register "0" (initial state).

3.3 Notes on use

3.3.2 Termination of unused pins

(1) Terminate unused pins

① I/O ports :
- Set the I/O ports for the input mode and connect them to V_{CC} or V_{SS} through each resistor of 1 kΩ to 10 kΩ.
  Ports that permit the selecting of a built-in pull-up resistor can also use this resistor. Set the I/O ports for the output mode and open them at "L" or "H".
- When opening them in the output mode, the input mode of the initial status remains until the mode of the ports is switched over to the output mode by the program after reset. Thus, the potential at these pins is undefined and the power source current may increase in the input mode. With regard to an effects on the system, thoroughly perform system evaluation on the user side.
- Since the direction register setup may be changed because of a program runaway or noise, set direction registers by program periodically to increase the reliability of program.

(2) Termination remarks

① Input ports and I/O ports :
Do not open in the input mode.

● **Reason**
- The power source current may increase depending on the first-stage circuit.
- An effect due to noise may be easily produced as compared with proper termination ② and ③ shown on the above.

② I/O ports :
When setting for the input mode, do not connect to V_{CC} or V_{SS} directly.

● **Reason**
If the direction register setup changes for the output mode because of a program runaway or noise, a short circuit may occur between a port and V_{CC} (or V_{SS}).

③ I/O ports :
When setting for the input mode, do not connect multiple ports in a lump to V_{CC} or V_{SS} through a resistor.

● **Reason**
If the direction register setup changes for the output mode because of a program runaway or noise, a short circuit may occur between ports.

- At the termination of unused pins, perform wiring at the shortest possible distance (20 mm or less) from microcomputer pins.

3.3.3 Notes on Timer

- When n (0 to 255) is written to a timer latch, the frequency division ratio is 1/(n+1).
- When a count source of timer X, timer Y or timer Z is switched, stop a count of timer X.

3.3.4 Notes on Timer A

Notes on using timer A are described below.

(1) Common to all modes

① When reading timer A (high-order) (TAH) and timer A (low-order) (TAL), the contents of timer A is read out. Read both registers in order of TAH and TAL following, certainly.
TAH and TAL keep the values until they are read out.
Also, do not write to them during read. In this case, unexpected operation may occur.

② When writing data to TAL and TAH even when timer A is operating or stopped, the data are set to timer A and timer A latch simultaneously. Write both registers in order of TAL and TAH following, certainly.
Also, do not read them during write. In this case, unexpected operation may occur.

(2) Period measurement mode, event counter mode, and pulse width HL continuously measurement mode

① In order to use $CNTR_1$ pin, set "0" to bit 0 of the port P0 direction register (input mode).

② In order to use $CNTR_1$ pin, set "1" to bit 7 of the interrupt control register to disable the $P0_0$ key-on wakeup function.

③ $CNTR_1$ interrupt active edge depends on the $CNTR_1$ active edge switch bit. When this bit is "0", the $CNTR_1$ interrupt request bit is set to "1" at the falling edge of the $CNTR_1$ pin input signal. When this bit is "1", the $CNTR_1$ interrupt request bit is set to "1" at the rising edge of the $CNTR_1$ pin input signal.
However, in the pulse width HL continuously measurement mode, $CNTR_1$ interrupt request is generated at both rising and falling edges of $CNTR_1$ pin input signal regardless of the setting of $CNTR_1$ active edge switch bit.

3.3.5 Notes on timer 1

Note on timer 1 is described below.

(1) Notes on set of the oscillation stabilizing time

Timer 1 can be used to set the oscillation stabilizing time after release of the **STP** instruction. The oscillation stabilizing time after release of STP instruction can be selected from "set automatically"/ "not set automatically" by the oscillation stabilizing time set bit after release of the **STP** instruction of MISRG. When "0" is set to this bit, "01_{16}" is set to timer 1 and "FF_{16}" is set to prescaler 1 automatically. When "1" is set to this bit, nothing is set to timer 1 and prescaler 1. Therefore, set the wait time according to the oscillation stabilizing time of the oscillation. Also, when timer 1 is used, set values again to timer 1 and prescaler 1 after system is returned from the stop mode.

3.3 Notes on use

3.3.6 Notes on Timer X

Notes on using each mode of timer X are described below.

(1) Count source

① f(X_{IN}) can be used only when a ceramic oscillator or a ring oscillator is used. Do not use f(X_{IN}) at RC oscillation.

(2) Pulse output mode

① In order to use $CNTR_0$ pin, set "1" to bit 4 of the port P1 direction register (output mode).

② In order to use TX_{OUT} pin, set "1" to bit 3 of the port P0 direction register (output mode).

③ $CNTR_0$ interrupt active edge depends on the $CNTR_0$ active edge switch bit. When this bit is "0", the $CNTR_0$ interrupt request bit is set to "1" at the falling edge of $CNTR_0$ pin input signal. When this bit is "1", the $CNTR_0$ interrupt request bit is set to "1" at the rising edge of $CNTR_0$ pin input signal.

(3) Pulse width measurement mode

① In order to use $CNTR_0$ pin, set "1" to bit 4 of the port P1 direction register (output mode).

② $CNTR_0$ interrupt active edge depends on the $CNTR_0$ active edge switch bit. When this bit is "0", the $CNTR_0$ interrupt request bit is set to "1" at the falling edge of $CNTR_0$ pin input signal. When this bit is "1", the $CNTR_0$ interrupt request bit is set to "1" at the rising edge of $CNTR_0$ pin input signal.

3.3.7 Notes on timer Y and timer Z

Notes on using each mode of Timer Y and Timer Z are described below.

(1) Timer mode (timer Y and timer Z)

① In the timer mode, TYP and TYS is not used.

(2) Programmable waveform generation mode (timer Y and timer Z)

① In the programmable waveform generation mode, values of TYS, EXPYP, and EXPYS are valid by writing to TYP because the setting to them is executed all at once by writing to TYP. Even when changing TYP is not required, write the same value again.

② In the programmable waveform generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TYP and the timing of timer underflow during the secondary interval simultaneously.
An example of a measurement is shown below.
ex.) The underflow by the primary and the underflow by secondary are stored by polling etc. using timer Y interrupt.
Writing to primary is performed in by judging that there is no problem if the underflow by secondary is completed with reference to primary write operation before.
(Depending on a primary and a secondary setting values, and primary write timing, it may be impossible.)

③ The waveform extension function by the timer Y waveform extension control bits can be used only when "00_{16}" is set to Prescaler Y.
When the value other than "00_{16}" is set to Prescaler Y, be sure to set "0" to EXPYP and EXPYS.
The waveform extension function by the timer Z waveform extension control bits can be used only when "00_{16}" is set to Prescaler Z. When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.

④ When using this mode, be sure to set "1" to the timer Y write control bit to select "write to latch only".

⑤ When TYS is read out, the undefined value is read out. However, while timer Y counts the setting value of TYS, the count value during the secondary interval can be obtained by reading the timer Y primary.

⑥ In order to use TY_{OUT} pin, set "1" to bit 1 of the port P0 direction register (output mode).

(3) Programmable one-shot generation mode (timer Z)

① In the programmable one-shot generation mode, the value of EXPZP becomes valid by writing to TZP. Even when changing TZP is not required, write the same value again.

② In the programmable one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow simultaneously.

③ The waveform extension function by the timer Z waveform extension control bits can be used only when "00_{16}" is set to Prescaler Z.
When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.
An example of a measurement is shown below.
ex.) The underflow of timer is stored by polling etc. using timer Z interrupt.
Writing to primary is performed in by judging that there is no problem if the underflow by secondary is completed with reference to primary write operation before.
(Depending on a primary setting value, primary write timing, software and timing of external trigger to INT_0 pin, it may be impossible.)

④ When using this mode, be sure to set "1" to the timer Z write control bit to select "write to latch only".

⑤ In order to use TZ_{OUT} pin, set "1" to bit 2 of the port P0 direction register (output mode).

⑥ Stop Timer Z to change the INT_0 pin one-shot trigger control bit and INT_0 pin one-shot trigger active edge selection bit.

(4) Programmable wait one-shot generation mode (timer Z)

① In the programmable wait one-shot generation mode, values of TZS, EXPZP and EXPZS are valid by writing to TZP. Even when changing TZP is not required, write the same value again.
An example of a measurement is shown below.
ex.) The underflow by the primary and the underflow by secondary are stored by polling etc. using timer Z interrupt.
Writing is performed in by judging that there is no problem if the underflow by secondary is completed with reference to primary write operation before.
(Depending on a primary setting value, primary write timing, software and timing of external trigger to INT_0 pin, it may be impossible.)

② In the programmable wait one-shot generation mode, when the setting value is changed while the waveform is output, set by software in order not to execute the writing to TZP and the timing of timer underflow during the secondary interval simultaneously.

③ The waveform extension function by the timer Z waveform extension control bit can be used only when "00_{16}" is set to Prescaler Z.
When the value other than "00_{16}" is set to Prescaler Z, be sure to set "0" to EXPZP and EXPZS. Also, when the timer Y underflow is selected as the timer Z count source, the waveform extension function cannot be used.

④ When using this mode, be sure to set "1" to the timer Z write control bits to select "write to latch only".

⑤ When TZS is read out, the undefined value is read out. However, while Timer Z counts the setting value of TZS (during one-shot output), the count value during the secondary interval can be obtained by reading TZP.

⑥ In order to use TZ_{OUT} pin, set "1" to bit 2 of the port P0 direction register (output mode).

⑦ Stop Timer Z to change the INT_0 pin one-shot trigger control bit and INT_0 pin one-shot trigger active edge selection bit.

(5) Common to all modes (timer Y and timer Z)

Timer Y can stop counting by setting "1" to the timer Y count stop bit in any mode.
Also, when Timer Y underflows, the timer Y interrupt request bit is set to "1".
Timer Y reloads the value of latch when counting is stopped by the timer Y count stop bit. (When timer is read out while timer is stopped, the value of latch is read. The value of timer can be read out only while timer is operating.)

3.3.8 Notes on Serial I/O1

Notes on using serial I/O1 are described below.

(1) Notes when selecting clock synchronous serial I/O

① When the clock synchronous serial I/O1 is used, serial I/O2 cannot be used.

② When the transmit operation is stopped, clear the serial I/O1 enable bit and the transmit enable bit to "0" (serial I/O1 and transmit disabled).

● **Reason**

Since transmission is not stopped and the transmission circuit is not initialized even if only the serial I/O1 enable bit is cleared to "0" (serial I/O1 disabled), the internal transmission is running (in this case, since pins TxD_1, RxD_1, S_{CLK1}, and $\overline{S_{RDY1}}$ function as I/O ports, the transmission data is not output). When data is written to the transmit buffer register in this state, data starts to be shifted to the transmit shift register. When the serial I/O1 enable bit is set to "1" at this time, the data during internally shifting is output to the TxD_1 pin and an operation failure occurs.

③ When the receive operation is stopped, clear the receive enable bit to "0" (receive disabled), or clear the serial I/O1 enable bit to "0" (serial I/O1 disabled).

④ When the transmit/receive operation is stopped, clear both the transmit enable bit and receive enable bit to "0" (transmit and receive disabled) simultaneously. (any one of data transmission and reception cannot be stopped.)

● **Reason**

In the clock synchronous serial I/O mode, the same clock is used for transmission and reception. If any one of transmission and reception is disabled, a bit error occurs because transmission and reception cannot be synchronized.
In this mode, the clock circuit of the transmission circuit also operates for data reception. Accordingly, the transmission circuit does not stop by clearing only the transmit enable bit to "0" (transmit disabled). Also, the transmission circuit cannot be initialized even if the serial I/O1 enable bit is cleared to "0" (serial I/O1 disabled) (same as ②).

⑤ When signals are output from the $\overline{S_{RDY1}}$ pin on the reception side by using an external clock, set all of the receive enable bit, the $\overline{S_{RDY1}}$ output enable bit, and the transmit enable bit to "1".

⑥ When the S_{RDY1} signal input is used, set the using pin to the input mode before data is written to the transmit/receive buffer register.

⑦ Setup of a serial I/O1 synchronous clock selection bit when a clock synchronous serial I/O is selected;
"0" : $P1_2$ pin turns into an output pin of a synchronous clock.
"1" : $P1_2$ pin turns into an input pin of a synchronous clock.
Setup of a $\overline{S_{RDY1}}$ output enable bit ($\overline{S_{RDY1}}$) when a clock synchronous serial I/O1 is selected;
"0" : $P1_3$ pin can be used as a normal I/O pin.
"1" : $P1_3$ pin turns into a $\overline{S_{RDY1}}$ output pin.

3.3 Notes on use

(2) Notes when selecting UART

① When the clock asynchronous serial I/O1 (UART) is used, serial I/O2 can be used only when BRG output divided by 16 is selected as the synchronous clock.

② When the transmit operation is stopped, clear the transmit enable bit to “0” (transmit disabled).

● **Reason**
Same as (1) ②.

③ When the receive operation is stopped, clear the receive enable bit to “0” (receive disabled).

④ When the transmit/receive operation is stopped, clear the transmit enable bit to “0” (transmit disabled) and receive enable bit to “0” (receive disabled).

⑤ Setup of a serial I/O1 synchronous clock selection bit when a clock asynchronous (UART) serial I/O is selected;
“0”: $P1_2$ pin can be used as a normal I/O pin.
“1”: $P1_2$ pin turns into an input pin of an external clock.
When clock asynchronous (UART) type serial I/O is selected, it is $P1_3$ pin. It can be used as a normal I/O pin.

(3) Notes common to clock synchronous serial I/O and UART

① Set the serial I/O control register again after the transmission and the reception circuits are reset by clearing both the transmit enable bit and the receive enable bit to “0.”

② The transmit shift completion flag changes from “1” to “0” with a delay of 0.5 to 1.5 shift clocks. When data transmission is controlled with referring to the flag after writing the data to the transmit buffer register, note the delay.

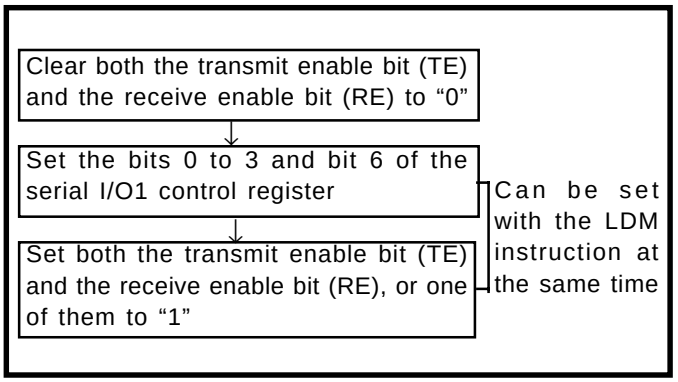


Fig. 3.3.1 Sequence of setting serial I/O1 control register again

③ When data transmission is executed at the state that an external clock input is selected as the synchronous clock, set “1” to the transmit enable bit while the S_{CLK1} is “H” state. Also, write to the transmit buffer register while the S_{CLK1} is “H” state.

④ When the transmit interrupt is used, set as the following sequence.
❶ Serial I/O1 transmit interrupt enable bit is set to “0” (disabled).
❷ Serial I/O1 transmit enable bit is set to “1”.
❸ Serial I/O1 transmit interrupt request bit is set to “0”.
❹ Serial I/O1 transmit interrupt enable bit is set to “1” (enabled).

● **Reason**
When the transmit enable bit is set to “1”, the transmit buffer empty flag and transmit shift completion flag are set to “1”.
Accordingly, even if the timing when any of the above flags is set to “1” is selected for the transmit interrupt source, interrupt request occurs and the transmit interrupt request bit is set.

⑤ Write to the baud rate generator (BRG) while the transmit/receive operation is stopped.

3.3.9 Notes on serial I/O2

Notes on using serial I/O2 are described below.

(1) Note on serial I/O1

Serial I/O2 can be used only when serial I/O1 is not used or serial I/O1 is used as UART and the BRG output divided by 16 is selected as the synchronous clock.

(2) Note on S_{CLK2} pin

When an external clock is selected, set "0" to bit 2 of the port P1 direction register (input mode).

(3) Note on S_{DATA2} pin

When $P1_3/S_{RDY1}/S_{DATA2}$ pin is used as the S_{DATA} input, set "0" to bit 3 of the port P1 direction register (input mode).

When the internal clock is selected as the transfer and $P1_3/S_{DATA2}$ pin is set to the input mode, the S_{DATA2} pin is in a high-impedance state after the data transfer is completed.

(4) Notes on serial I/O2 transmit/receive shift completion flag

① The transmit/receive shift completion flag of the serial I/O2 control register is "1" after transmit/receive shift is completed. In order to set "0" to this flag, set data (dummy data at receive) to the serial I/O2 register by program.

② Bit 7 (transmit/receive shift completion flag) of the serial I/O2 control register is set earlier than the completion of the actual shift operation for a half cycle of shift clock. Accordingly, when the shift completion is checked by using this bit, read/write the serial I/O2 register after a half or more cycle of clock from the setting "1" to this bit is checked.

3.3 Notes on use

3.3.10 Notes on A-D converter

Notes on A-D converter are described below.

(1) Analog input pin

Figure 3.3.2 shows the internal equivalent circuit of an analog input. In order to execute the A-D conversion correctly, to complete the charge to an internal capacitor within the specified time is required. The maximum output impedance of the analog input source required to complete the charge to a capacitor within the specified time is as follows;

About 35 kΩ (at $f(X_{IN})$ = 8 MHz)

When the maximum output impedance exceeds the above value, equip an analog input pin with an external capacitor of 0.01μF to 1μF between an analog input pin and V_{SS}.
Further, be sure to verify the operation of application products on the user side.

● **Reason**

An analog input pin includes the capacitor for analog voltage comparison. Accordingly, when signals from signal source with high impedance are input to an analog input pin, charge and discharge noise generates. This may cause the A-D conversion/comparison precision to be worse.

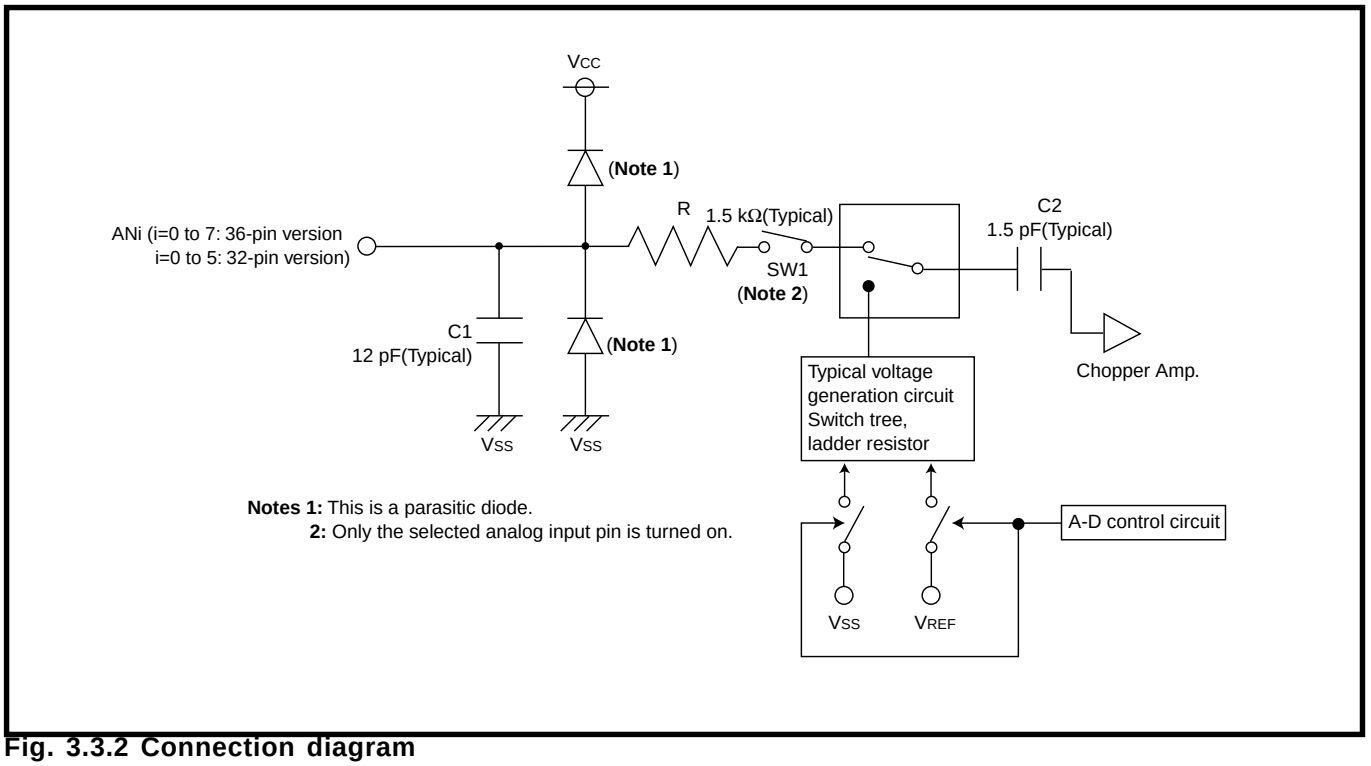


Fig. 3.3.2 Connection diagram

(2) Clock frequency during A-D conversion

The comparator consists of a capacity coupling, and a charge of the capacity will be lost if the clock frequency is too low. Thus, make sure the following during an A-D conversion.

- $f(X_{IN})$ is 500 kHz or more
- Do not execute the **STP** instruction

3.3.11 Notes on oscillation stop detection circuit

Notes on using oscillation stop detection circuit are described below.

(1) Note on ring oscillator

① The 7540 Group starts operation by the ring oscillator.

(2) Notes on oscillation circuit stop detection circuit

① When the stop mode is used, set the oscillation stop detection function to “invalid”.

② When f(X_{IN}) oscillation is stopped, set the oscillation stop detection function to “invalid”.

③ The oscillation stop detection circuit is not included in the emulator MCU “M37540RSS”.

(3) Notes on stop mode

① When the stop mode is used, set the oscillation stop detection function to “invalid”.

② When the stop mode is used, set “0” (**STP** instruction enabled) to the **STP** instruction disable bit of the watchdog timer control register.

③ Timer 1 can be used to set the oscillation stabilizing time after release of the **STP** instruction. The oscillation stabilizing time after release of **STP** instruction can be selected from “set automatically”/ “not set automatically” by the oscillation stabilizing time set bit after release of the **STP** instruction of MISRG. When “0” is set to this bit, “01_{16}” is set to timer 1 and “FF_{16}” is set to prescaler 1 automatically. When “1” is set to this bit, nothing is set to timer 1 and prescaler 1. Therefore, set the wait time according to the oscillation stabilizing time of the oscillation. Also, when timer 1 is used, set values again to timer 1 and prescaler 1 after system is returned from the stop mode.

④ The **STP** instruction cannot be used during CPU is operating by the ring oscillator.

⑤ When the stop mode is used, stop the ring oscillator oscillation.

⑥ Do not execute the **STP** instruction during the A-D conversion.

(4) Note on wait mode

① When the wait mode is used, stop the clock except the operation clock source.

(5) Notes on state transition

① When the operation clock source is f(X_{IN}), the CPU clock division ratio can be selected from the following;
- f(X_{IN})/2 (high-speed mode)
- f(X_{IN})/8 (middle-speed mode)
- f(X_{IN}) (double-speed mode)

The double-speed mode can be used only at ceramic oscillation.
Do not use the mode at RC oscillation.

② Stabilize the f(X_{IN}) oscillation to change the operation clock source from the ring oscillator to f(X_{IN}).

③ When the ring oscillation is used as the operation clock, the CPU clock division ratio is the middle-speed mode.

④ When the state transition state 2→state 3→state 4 is performed, execute the NOP instruction as shown below according to the division ratio of CPU clock.
- $CPUM_{76} \rightarrow 10_2$ (State 2→state 3)
- NOP instruction
- $CPUM_4 \rightarrow 1_2$ (State 3→state 4)

Double-speed mode at ring oscillator: NOP×3
High-speed mode at ring oscillator: NOP×1
Middle-speed mode at ring oscillator: NOP×0

(6) Switch of ceramic and RC oscillations

After releasing reset the operation starts by starting a built-in ring oscillator. Then, a ceramic oscillation or an RC oscillation is selected by setting bit 5 of the CPU mode register.

(7) Double-speed mode

When a ceramic oscillation is selected, a double-speed mode can be used. Do not use it when an RC oscillation is selected.

(8) Clock division ratio, X_{IN} oscillation control, ring oscillator control

The state transition shown in Figure 3.3.3 can be performed by setting the clock division ratio selection bits (bits 7 and 6), X_{IN} oscillation control bit (bit 4), ring oscillator oscillation control bit (bit 3) of CPU mode register. Be careful of notes on use in Figure 3.3.3.

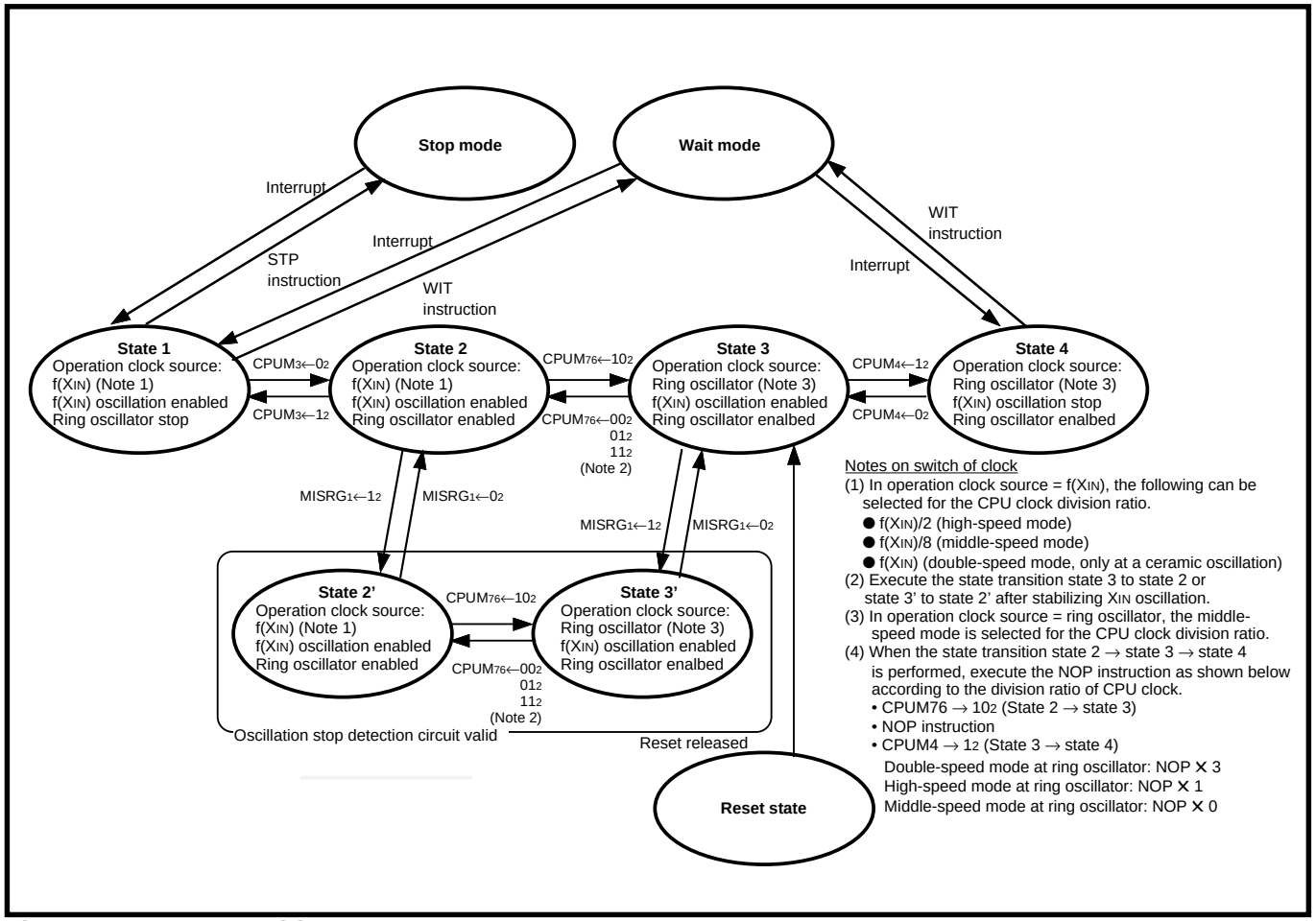


Fig. 3.3.3 State transition

3.3.12 Notes on CPU mode register

(1) Switching method of CPU mode register after releasing reset

Switch the CPU mode register (CPUM) at the head of program after releasing reset in the following method.

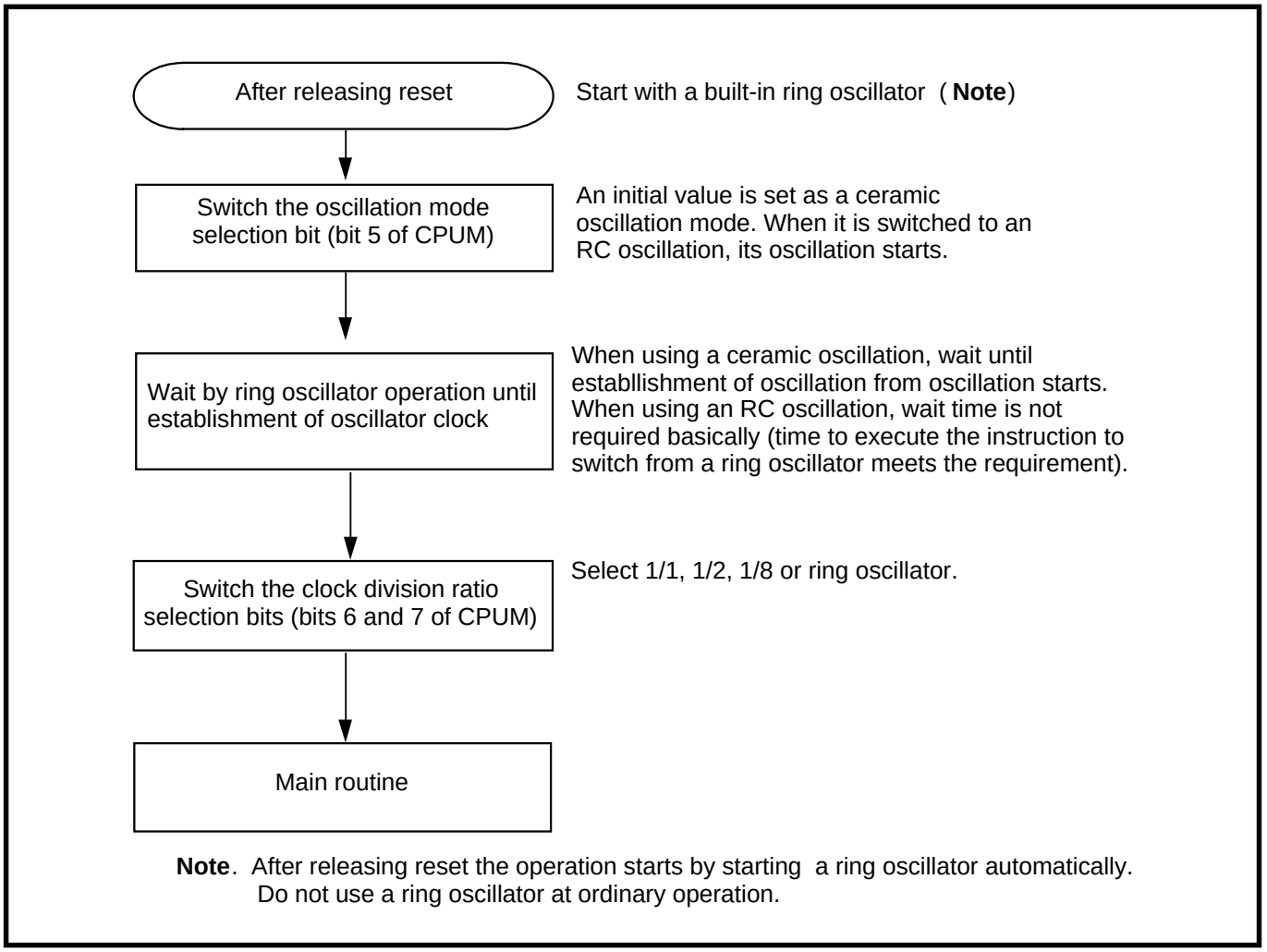


Fig. 3.3.4 Switching method of CPU mode register

(2) CPU mode register

Bits 5, 1 and 0 of CPU mode register are used to select oscillation mode and to control operation modes of the microcomputer. In order to prevent the dead-lock by error-writing (ex. program run-away), these bits can be rewritten only once after releasing reset. After rewriting it is disable to write any data to the bit. (The emulator MCU “M37540RSS” is excluded.)

Also, when the read-modify-write instructions (SEB, CLB) are executed to bits 2 to 4, 6 and 7, bits 5, 1 and 0 are locked.

3.3.13 Notes on interrupts

(1) Switching external interrupt detection edge

For the products able to switch the external interrupt detection edge, switch it as the following sequence.

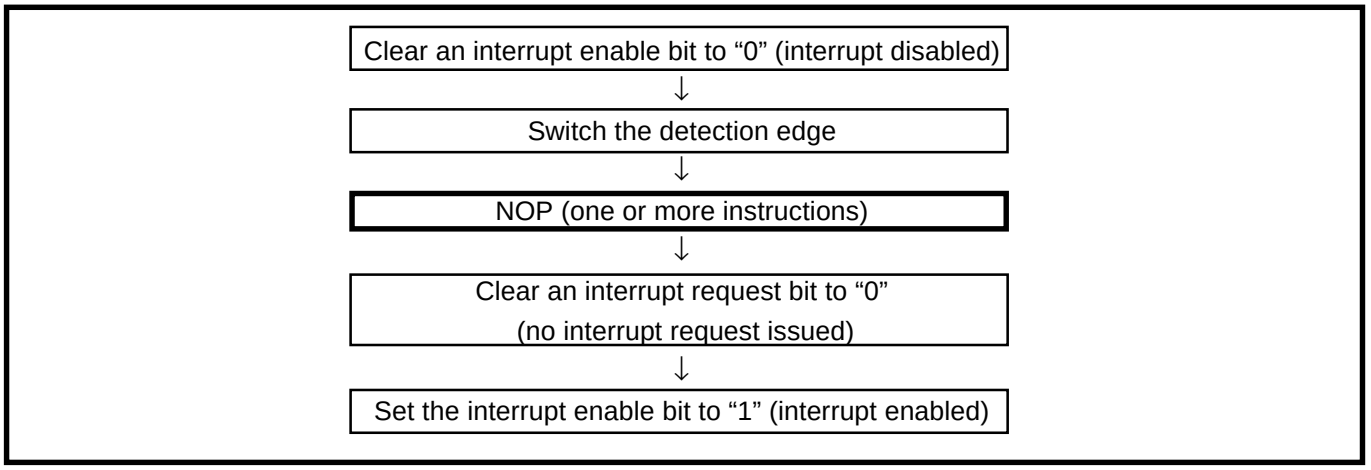


Fig. 3.3.5 Sequence of switch the detection edge

● **Reason**

The interrupt circuit recognizes the switching of the detection edge as the change of external input signals. This may cause an unnecessary interrupt.

(2) Check of interrupt request bit

When executing the **BBC** or **BBS** instruction to an interrupt request bit of an interrupt request register immediately after this bit is set to “0” by using a data transfer instruction, execute one or more instructions before executing the **BBC** or **BBS** instruction.

● **Reason**

If the **BBC** or **BBS** instruction is executed immediately after an interrupt request bit of an interrupt request register is cleared to “0”, the value of the interrupt request bit before being cleared to “0” is read.

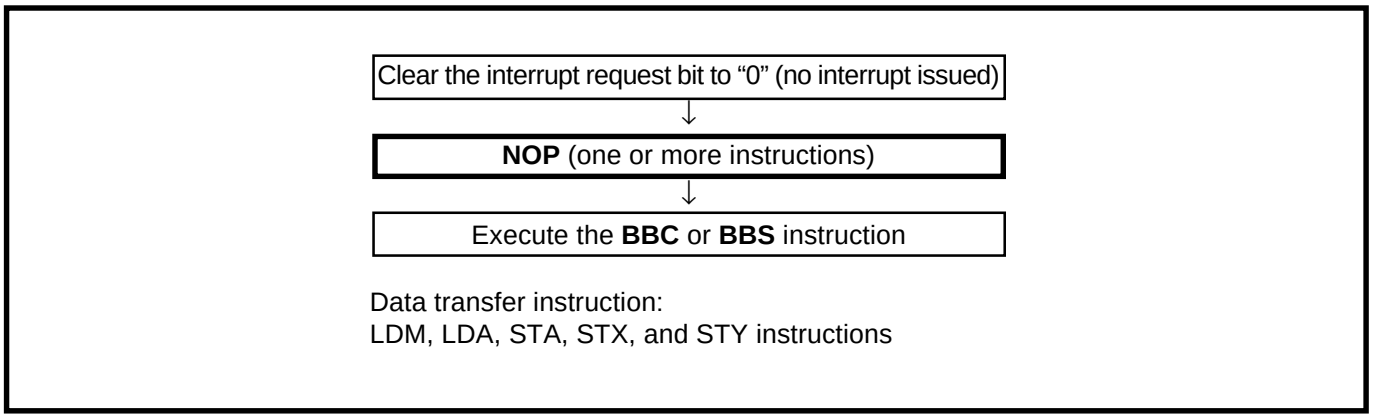


Fig. 3.3.6 Sequence of check of interrupt request bit

(3) Structure of interrupt control register 2

Fix the bit 7 of the interrupt control register 1 to “0”. Figure 3.3.7 shows the structure of the interrupt control register 2.

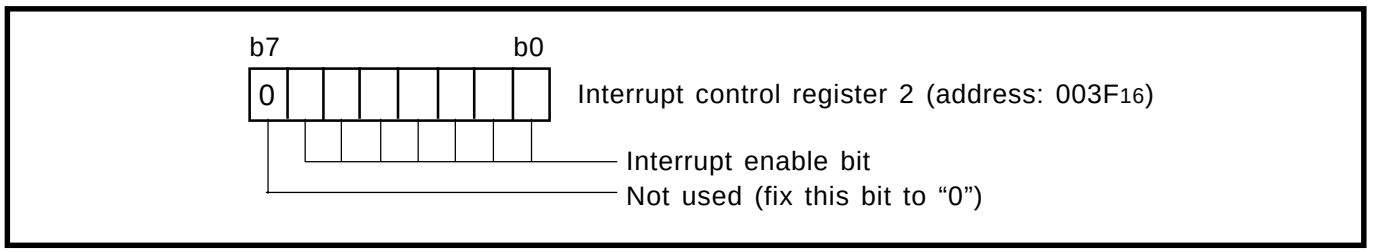


Fig. 3.3.7 Structure of interrupt control register 2

(4) Interrupt

When setting the followings, the interrupt request bit may be set to “1”.

•When switching external interrupt active edge

Related register: Interrupt edge selection register (address $003A_{16}$)

Timer X mode register (address $002B_{16}$)

Timer A mode register (address $001D_{16}$)

When not requiring the interrupt occurrence synchronized with these setting, take the following sequence.

① Set the corresponding interrupt enable bit to “0” (disabled).

② Set the interrupt edge select bit (active edge switch bit).

③ Set the corresponding interrupt request bit to “0” after 1 or more instructions have been executed.

④ Set the corresponding interrupt enable bit to “1” (enabled).

3.3.14 Notes on $\overline{\text{RESET}}$ pin

(1) Connecting capacitor

In case where the $\overline{\text{RESET}}$ signal rise time is long, connect a ceramic capacitor or others across the $\overline{\text{RESET}}$ pin and the V_{SS} pin. And use a 1000 pF or more capacitor for high frequency use. When connecting the capacitor, note the following :

• Make the length of the wiring which is connected to a capacitor as short as possible.

• Be sure to verify the operation of application products on the user side.

● **Reason**

If the several nanosecond or several ten nanosecond impulse noise enters the $\overline{\text{RESET}}$ pin, it may cause a microcomputer failure.

3.3.15 Notes on programming

(1) Processor status register

① **Initializing of processor status register**

Flags which affect program execution must be initialized after a reset.
In particular, it is essential to initialize the T and D flags because they have an important effect on calculations.

● **Reason**

After a reset, the contents of the processor status register (PS) are undefined except for the I flag which is “1”.

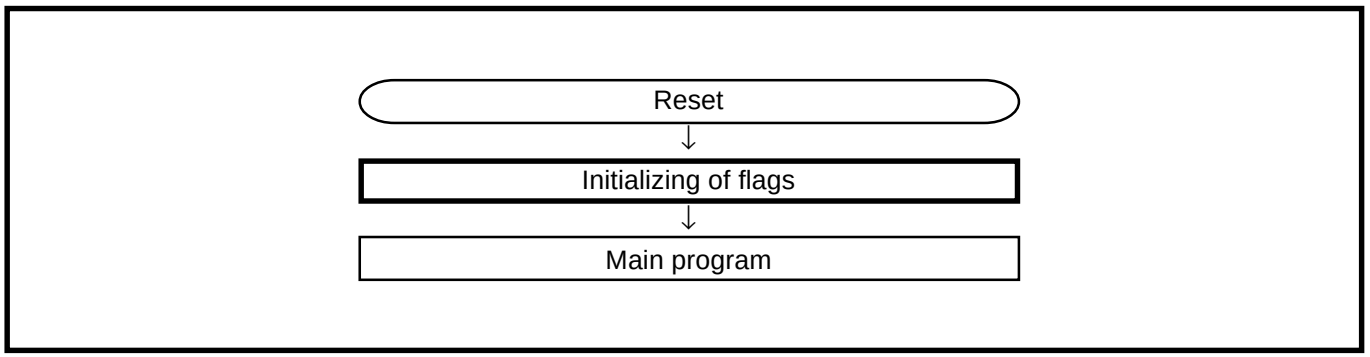


Fig. 3.3.8 Initialization of processor status register

② **How to reference the processor status register**

To reference the contents of the processor status register (PS), execute the **PHP** instruction once then read the contents of (S+1). If necessary, execute the **PLP** instruction to return the PS to its original status.
A **NOP** instruction should be executed after every **PLP** instruction.

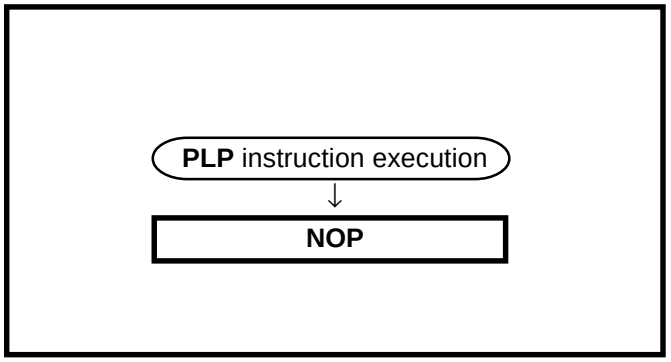


Fig. 3.3.9 Sequence of PLP instruction execution

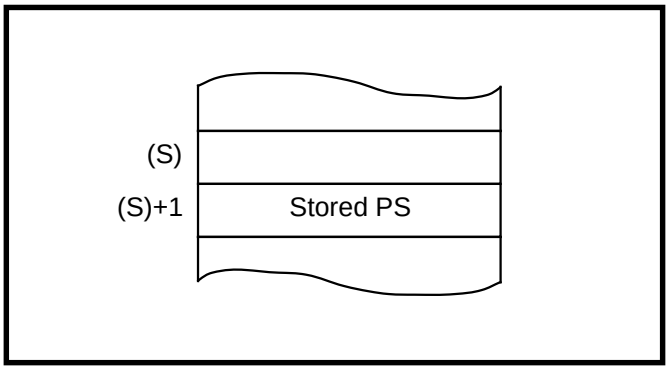


Fig. 3.3.10 Stack memory contents after PHP instruction execution

(2) Decimal calculations

① **Execution of decimal calculations**
The **ADC** and **SBC** are the only instructions which will yield proper decimal notation, set the decimal mode flag (D) to "1" with the **SED** instruction. After executing the **ADC** or **SBC** instruction, execute another instruction before executing the **SEC**, **CLC**, or **CLD** instruction.

② **Notes on status flag in decimal mode**
When decimal mode is selected, the values of three of the flags in the status register (the N, V, and Z flags) are invalid after a **ADC** or **SBC** instruction is executed.
The carry flag (C) is set to "1" if a carry is generated as a result of the calculation, or is cleared to "0" if a borrow is generated. To determine whether a calculation has generated a carry, the C flag must be initialized to "0" before each calculation. To check for a borrow, the C flag must be initialized to "1" before each calculation.

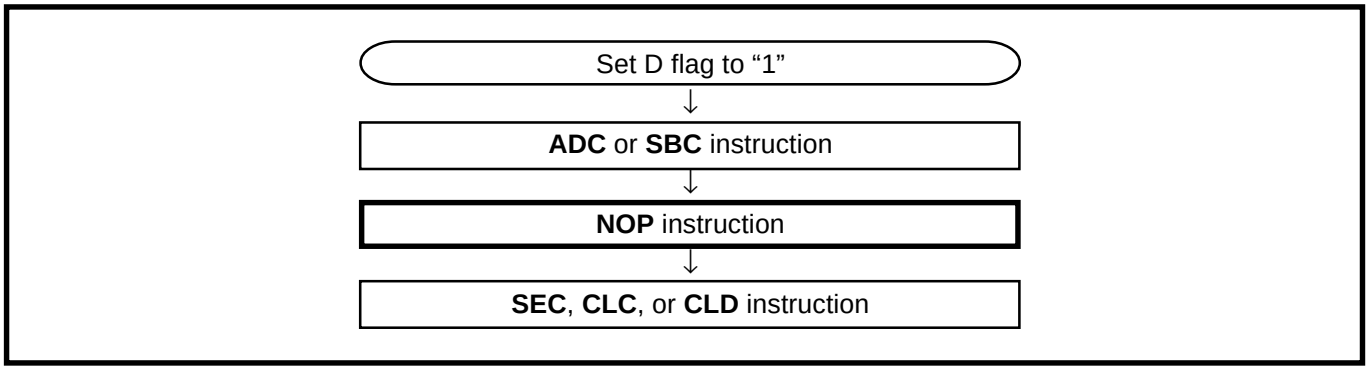


Fig. 3.3.11 Status flag at decimal calculations

(3) JMP instruction
When using the **JMP** instruction in indirect addressing mode, do not specify the last address on a page as an indirect address.

(4) Interrupts
The contents of the interrupt request bit do not change even if the BBC or BBS instruction is executed immediately after they are changed by program because this instruction is executed for the previous contents. For executing the instruction for the changed contents, execute one instruction before executing the BBC or BBS instruction.

(5) Ports
• The values of the port direction registers cannot be read.
That is, it is impossible to use the LDA instruction, memory operation instruction when the T flag is "1", addressing mode using direction register values as qualifiers, and bit test instructions such as BBC and BBS.
It is also impossible to use bit operation instructions such as CLB and SEB and read/modify/write instructions of direction registers for calculations such as ROR.
For setting direction registers, use the LDM instruction, STA instruction, etc.

(6) A-D Conversion
Do not execute the STP instruction during A-D conversion.

(7) Instruction Execution Timing

The instruction execution time can be obtained by multiplying the frequency of the internal clock f by the number of cycles mentioned in the machine-language instruction table.

The frequency of the internal clock f is the same as that of the X_{IN} in double-speed mode, twice the X_{IN} cycle in high-speed mode and 8 times the X_{IN} cycle in middle-speed mode.

(8) CPU Mode Register

The oscillation mode selection bit and processor mode bits can be rewritten only once after releasing reset. However, after rewriting it is disable to write any value to the bit. (Emulator MCU is excluded.) When a ceramic oscillation is selected, a double-speed mode of the clock division ratio selection bits can be used. Do not use it when an RC oscillation is selected.

3.3.16 Programming and test of built-in PROM version

As for in the One Time PROM version (shipped in blank), its built-in PROM can be read or programmed with a general-purpose PROM programmer using a special programming adapter.

The programming test and screening for PROM of the One Time PROM version (shipped in blank) are not performed in the assembly process and the following processes. To ensure reliability after programming, performing programming and test according to the Figure 3.3.12 before actual use are recommended.

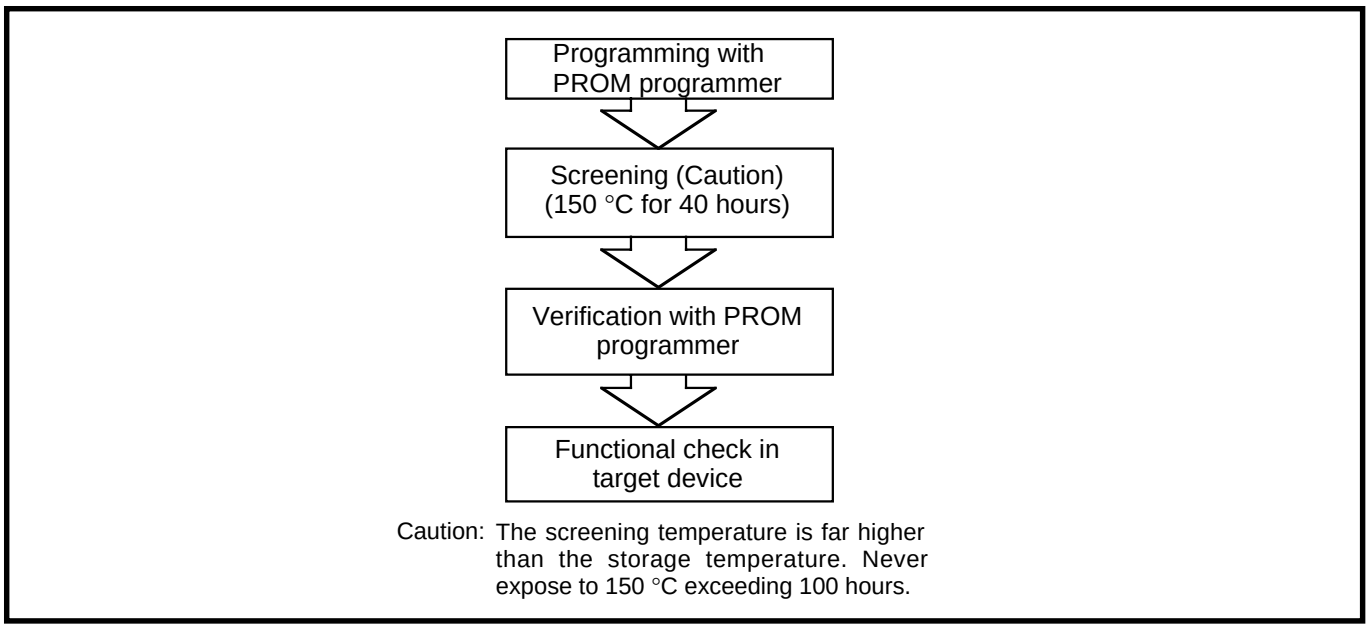


Fig. 3.3.12 Programming and testing of One Time PROM version

(1) One Time PROM Version

The CNVss pin is connected to the internal memory circuit block by a low-ohmic resistance, since it has the multiplexed function to be a programmable power source pin (V_{PP} pin) as well.

To improve the noise reduction, connect a track between CNVss pin and Vss pin with 1 to 10 kΩ resistance.

The mask ROM version track of CNVss pin has no operational interference even if it is connected via a resistor.

3.3.17 Handling of Power Source Pin

In order to avoid a latch-up occurrence, connect a capacitor suitable for high frequencies as bypass capacitor between power source pin (Vcc pin) and GND pin (Vss pin). Besides, connect the capacitor to as close as possible. For bypass capacitor which should not be located too far from the pins to be connected, a ceramic capacitor of 0.01 μF to 0.1 μF is recommended.

3.3.18 Notes on built-in PROM version

(1) Programming adapter

Use a special programming adapter shown in Table 3.3.1 and a general-purpose PROM programmer when reading from or programming to the built-in PROM in the built-in PROM version.

Table 3.3.1 Programming adapters

Part Number	Programming adapter
M37540E8GP (One Time PROM version shipped in blank)	PCA7435GPG03
M37540E8SP (One Time PROM version shipped in blank)	PCA7435SPG02
M37540E8FP (One Time PROM version shipped in blank)	PCA7435FPG02

(2) Programming/reading

In PROM mode, operation is the same as that of the M5M27C101AK, but programming conditions of PROM programmer are not set automatically because there are no internal device ID codes. Accurately set the following conditions for data programming /reading. Take care not to apply 21 V to V_{PP} pin (is also used as the CNV_{SS} pin), or the product may be permanently damaged.

- Programming voltage: 12.5 V
- Setting of PROM programmer switch: refer to Table 3.3.2.

Table 3.3.2 PROM programmer address setting

Part Number	PROM programmer start address	PROM programmer end address
M37540E8GP	Address 0808_{16} (**Note**)	Address $0FFFD_{16}$ (**Note**)
M37540E8SP		
M37540E8FP		

Note: Addersses 8080_{16} to $FFFD_{16}$ in the built-in PROM corresponds to addresses 08080_{16} to $0FFFD_{16}$ in the PROM programmer.

3.4 Countermeasures against noise

3.4.1 Shortest wiring length

(1) Package

Select the smallest possible package to make the total wiring length short.

● **Reason**

The wiring length depends on a microcomputer package. Use of a small package, for example QFP and not DIP, makes the total wiring length short to reduce influence of noise.

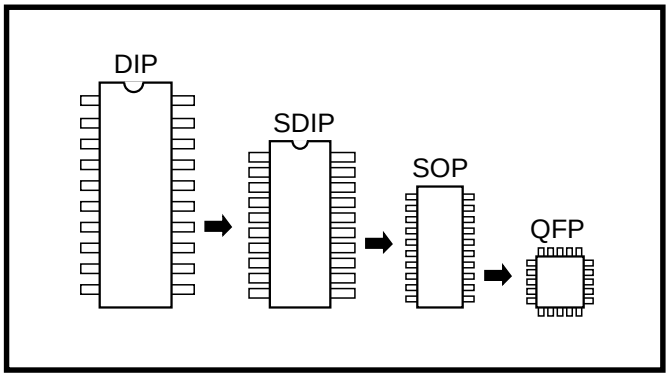


Fig. 3.4.1 Selection of packages

(2) Wiring for $\overline{\text{RESET}}$ pin

Make the length of wiring which is connected to the $\overline{\text{RESET}}$ pin as short as possible. Especially, connect a capacitor across the $\overline{\text{RESET}}$ pin and the V_{SS} pin with the shortest possible wiring (within 20mm).

● **Reason**

The width of a pulse input into the $\overline{\text{RESET}}$ pin is determined by the timing necessary conditions. If noise having a shorter pulse width than the standard is input to the $\overline{\text{RESET}}$ pin, the reset is released before the internal state of the microcomputer is completely initialized. This may cause a program runaway.

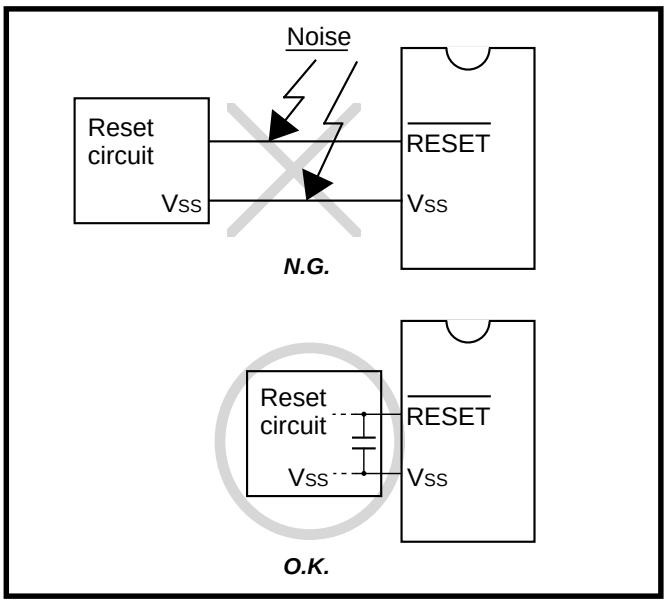


Fig. 3.4.2 Wiring for the $\overline{\text{RESET}}$ pin

(3) Wiring for clock input/output pins

- Make the length of wiring which is connected to clock I/O pins as short as possible.
- Make the length of wiring (within 20mm) across the grounding lead of a capacitor which is connected to an oscillator and the Vss pin of a microcomputer as short as possible.
- Separate the Vss pattern only for oscillation from other Vss patterns.

● **Reason**

If noise enters clock I/O pins, clock waveforms may be deformed. This may cause a program failure or program runaway. Also, if a potential difference is caused by the noise between the Vss level of a microcomputer and the Vss level of an oscillator, the correct clock will not be input in the microcomputer.

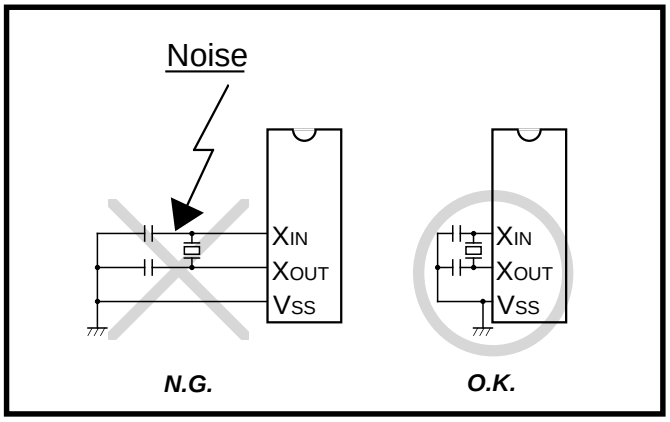


Fig. 3.4.3 Wiring for clock I/O pins

(4) Wiring to CNVss pin

Connect the CNVss pin to the Vss pin with the shortest possible wiring.

● **Reason**

The processor mode of a microcomputer is influenced by a potential at the CNVss pin. If a potential difference is caused by the noise between pins CNVss and Vss, the processor mode may become unstable. This may cause a microcomputer malfunction or a program runaway.

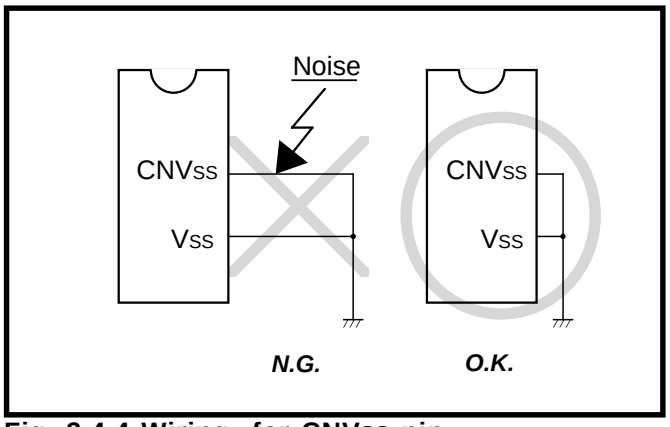


Fig. 3.4.4 Wiring for CNVss pin

3.4 Countermeasures against noise

(5) Wiring to V_{PP} pin of One Time PROM version

Connect an approximately 5 kΩ resistor to the V_{PP} pin the shortest possible in series and also to the V_{SS} pin. When not connecting the resistor, make the length of wiring between the V_{PP} pin and the V_{SS} pin the shortest possible.

Note: Even when a circuit which included an approximately 5 kΩ resistor is used in the Mask ROM version, the microcomputer operates correctly.

● **Reason**

The V_{PP} pin of the One Time PROM is the power source input pin for the built-in PROM. When programming in the built-in PROM, the impedance of the V_{PP} pin is low to allow the electric current for writing flow into the PROM. Because of this, noise can enter easily. If noise enters the V_{PP} pin, abnormal instruction codes or data are read from the built-in PROM, which may cause a program runaway.

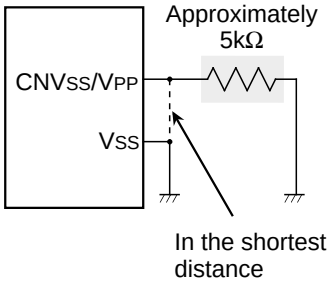


Fig. 3.4.5 Wiring for the V_{PP} pin of the One Time PROM

3.4.2 Connection of bypass capacitor across Vss line and Vcc line

Connect an approximately 0.1 μF bypass capacitor across the V_{SS} line and the V_{CC} line as follows:

- Connect a bypass capacitor across the V_{SS} pin and the V_{CC} pin at equal length.
- Connect a bypass capacitor across the V_{SS} pin and the V_{CC} pin with the shortest possible wiring.
- Use lines with a larger diameter than other signal lines for V_{SS} line and V_{CC} line.
- Connect the power source wiring via a bypass capacitor to the V_{SS} pin and the V_{CC} pin.

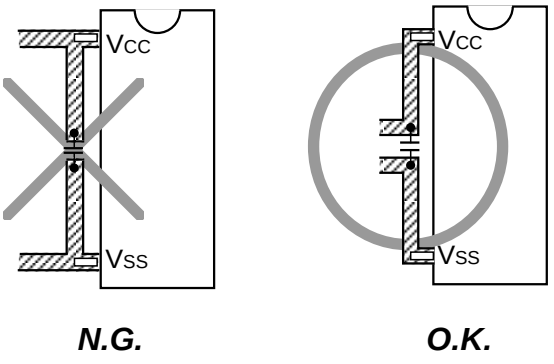


Fig. 3.4.6 Bypass capacitor across the Vss line and the Vcc line

3.4.3 Wiring to analog input pins

- Connect an approximately 100 Ω to 1 kΩ resistor to an analog signal line which is connected to an analog input pin in series. Besides, connect the resistor to the microcomputer as close as possible.
- Connect an approximately 1000 pF capacitor across the VSS pin and the analog input pin. Besides, connect the capacitor to the VSS pin as close as possible. Also, connect the capacitor across the analog input pin and the VSS pin at equal length.

● **Reason**

Signals which is input in an analog input pin (such as an A-D converter/comparator input pin) are usually output signals from sensor. The sensor which detects a change of event is installed far from the printed circuit board with a microcomputer, the wiring to an analog input pin is longer necessarily. This long wiring functions as an antenna which feeds noise into the microcomputer, which causes noise to an analog input pin.

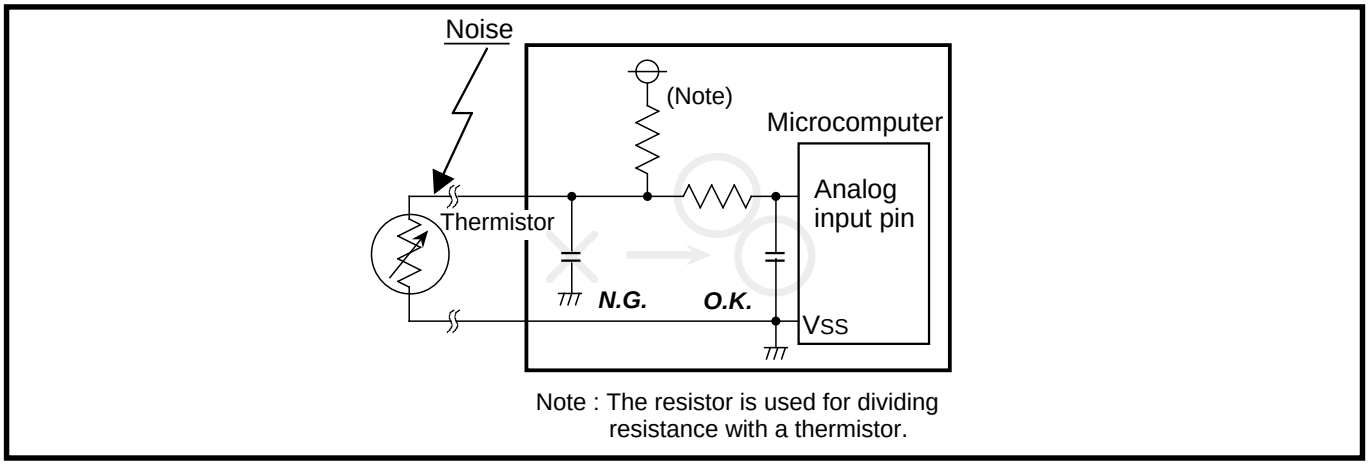


Fig. 3.4.7 Analog signal line and a resistor and a capacitor

3.4.4 Oscillator concerns

Take care to prevent an oscillator that generates clocks for a microcomputer operation from being affected by other signals.

(1) Keeping oscillator away from large current signal lines

Install a microcomputer (and especially an oscillator) as far as possible from signal lines where a current larger than the tolerance of current value flows.

● **Reason**

In the system using a microcomputer, there are signal lines for controlling motors, LEDs, and thermal heads or others. When a large current flows through those signal lines, strong noise occurs because of mutual inductance.

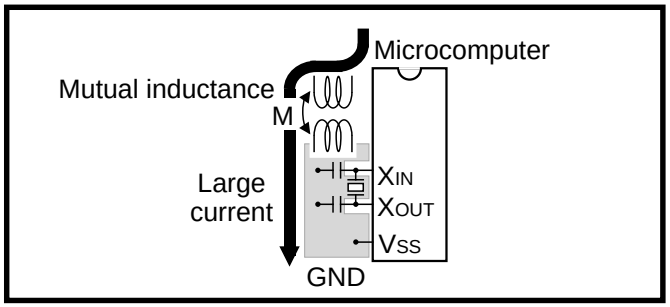


Fig. 3.4.8 Wiring for a large current signal line

3.4 Countermeasures against noise

(2) Installing oscillator away from signal lines where potential levels change frequently

Install an oscillator and a connecting pattern of an oscillator away from signal lines where potential levels change frequently. Also, do not cross such signal lines over the clock lines or the signal lines which are sensitive to noise.

● **Reason**

Signal lines where potential levels change frequently (such as the CNTR pin signal line) may affect other lines at signal rising edge or falling edge. If such lines cross over a clock line, clock waveforms may be deformed, which causes a microcomputer failure or a program runaway.

3) Oscillator protection using Vss pattern

As for a two-sided printed circuit board, print a VSS pattern on the underside (soldering side) of the position (on the component side) where an oscillator is mounted.

Connect the VSS pattern to the microcomputer VSS pin with the shortest possible wiring. Besides, separate this VSS pattern from other VSS patterns.

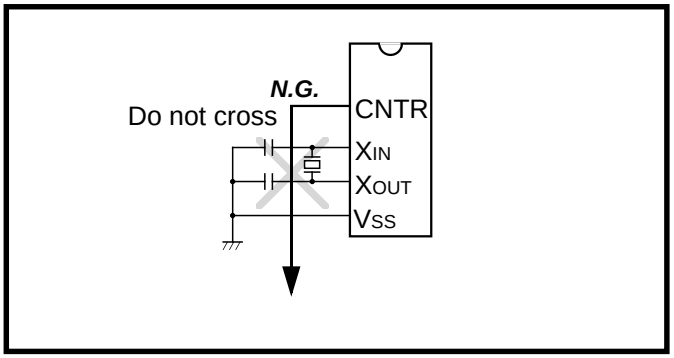


Fig. 3.4.9 Wiring of signal lines where potential levels change frequently

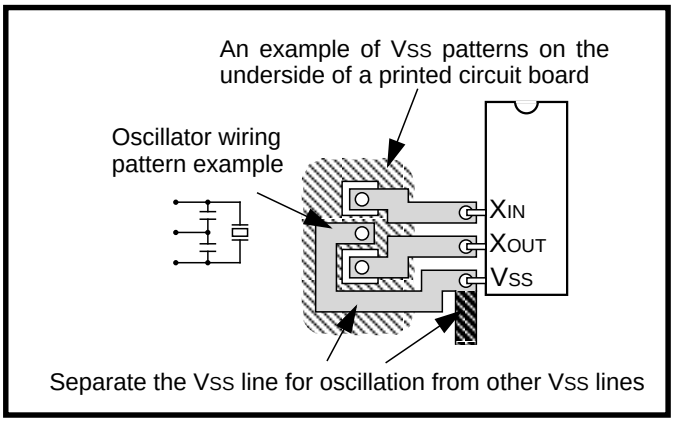


Fig. 3.4.10 VSS pattern on the underside of an oscillator

3.4.5 Setup for I/O ports

Setup I/O ports using hardware and software as follows:

<Hardware>

- Connect a resistor of 100 Ω or more to an I/O port in series.

<Software>

- As for an input port, read data several times by a program for checking whether input levels are equal or not.
- As for an output port, since the output data may reverse because of noise, rewrite data to its port latch at fixed periods.
- Rewrite data to direction registers and pull-up control registers at fixed periods.

Note: When a direction register is set for input port again at fixed periods, a several-nanosecond short pulse may be output from this port. If this is undesirable, connect a capacitor to this port to remove the noise pulse.

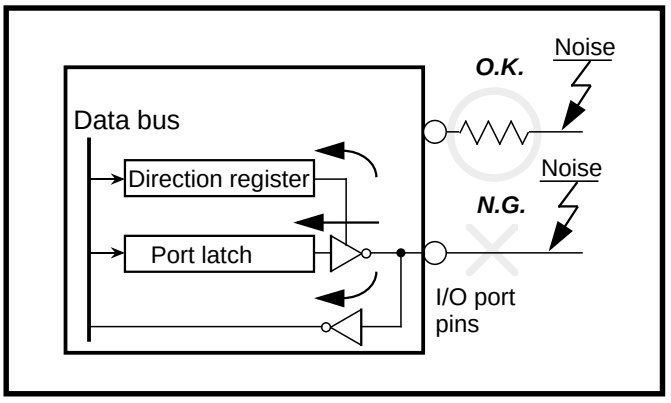


Fig. 3.4.11 Setup for I/O ports

3.4.6 Providing of watchdog timer function by software

If a microcomputer runs away because of noise or others, it can be detected by a software watchdog timer and the microcomputer can be reset to normal operation. This is equal to or more effective than program runaway detection by a hardware watchdog timer. The following shows an example of a watchdog timer provided by software.

In the following example, to reset a microcomputer to normal operation, the main routine detects errors of the interrupt processing routine and the interrupt processing routine detects errors of the main routine. This example assumes that interrupt processing is repeated multiple times in a single main routine processing.

<The main routine>

- Assigns a single byte of RAM to a software watchdog timer (SWDT) and writes the initial value N in the SWDT once at each execution of the main routine. The initial value N should satisfy the following condition:
  N+1 ≥ (Counts of interrupt processing executed in each main routine)
  As the main routine execution cycle may change because of an interrupt processing or others, the initial value N should have a margin.
- Watches the operation of the interrupt processing routine by comparing the SWDT contents with counts of interrupt processing after the initial value N has been set.
- Detects that the interrupt processing routine has failed and determines to branch to the program initialization routine for recovery processing in the following case:
  If the SWDT contents do not change after interrupt processing.

<The interrupt processing routine>

- Decrements the SWDT contents by 1 at each interrupt processing.
- Determines that the main routine operates normally when the SWDT contents are reset to the initial value N at almost fixed cycles (at the fixed interrupt processing count).
- Detects that the main routine has failed and determines to branch to the program initialization routine for recovery processing in the following case:
  If the SWDT contents are not initialized to the initial value N but continued to decrement and if they reach 0 or less.

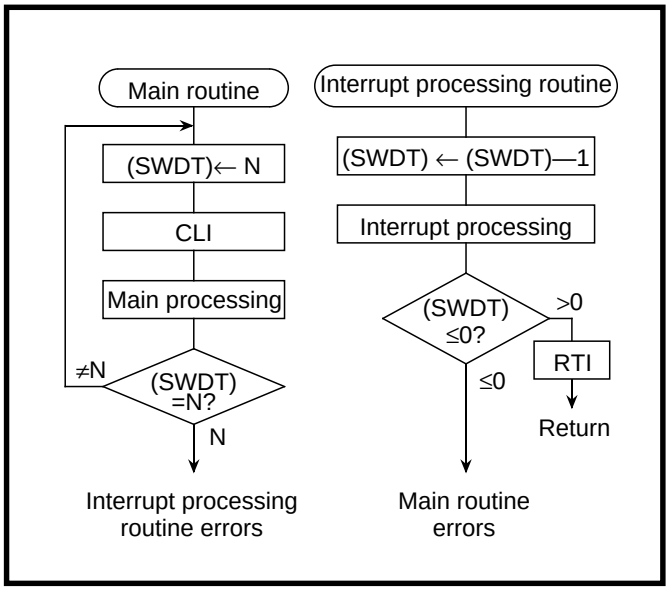


Fig. 3.4.12 Watchdog timer by software

3.5 List of registers

Port Pi

b7 b6 b5 b4 b3 b2 b1 b0

Port Pi (Pi) (i = 0, 2, 3) [Address : 00_{16}, 04_{16}, 06_{16}]

B	Name	Function	At reset	R	W
0	Port Pi_0	• In output mode Write } Port latch Read } • In input mode Write : Port latch Read : Value of pins	?	○	○
1	Port Pi_1		?	○	○
2	Port Pi_2		?	○	○
3	Port Pi_3		?	○	○
4	Port Pi_4		?	○	○
5	Port Pi_5		?	○	○
6	Port Pi_6		?	○	○
7	Port Pi_7		?	○	○

Note: The 32-pin package versions have nothing to be allocated for the following:
•Bits 6 and 7 of port P2
•Bits 5 and 6 of port P3.

Fig. 3.5.1 Structure of Port Pi (i = 0, 2, 3)

Port P1

b7 b6 b5 b4 b3 b2 b1 b0

Port P1 (P1) [Address : 02_{16}]

B	Name	Function	At reset	R	W
0	Port $P1_0$	• In output mode Write } Port latch Read } • In input mode Write : Port latch Read : Value of pins	?	○	○
1	Port $P1_1$		?	○	○
2	Port $P1_2$		?	○	○
3	Port $P1_3$		?	○	○
4	Port $P1_4$		?	○	○
5	Nothing is allocated for these bits. When these bits are read out, the values are undefined.		?	×	×
6			?	×	×
7			?	×	×

Fig. 3.5.2 Structure of Port P1

Port Pi direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port Pi direction register (PiD) (i = 0, 2, 3) [Address : 01 16, 05 16, 07 16]

B	Name	Function	At reset	R	W
0	Port Pi direction register	0 : Port Pi_0 input mode 1 : Port Pi_0 output mode	0	×	○
1		0 : Port Pi_1 input mode 1 : Port Pi_1 output mode	0	×	○
2		0 : Port Pi_2 input mode 1 : Port Pi_2 output mode	0	×	○
3		0 : Port Pi_3 input mode 1 : Port Pi_3 output mode	0	×	○
4		0 : Port Pi_4 input mode 1 : Port Pi_4 output mode	0	×	○
5		0 : Port Pi_5 input mode 1 : Port Pi_5 output mode	0	×	○
6		0 : Port Pi_6 input mode 1 : Port Pi_6 output mode	0	×	○
7		0 : Port Pi_7 input mode 1 : Port Pi_7 output mode	0	×	○

Note: The 32-pin package versions have nothing to be allocated for the following:
- Bits 6 and 7 of P2D
- Bits 5 and 6 of P3D.

Fig. 3.5.3 Structure of Port Pi direction register (i = 0, 2, 3)

Port P1 direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port P1 direction register (P1D) [Address : 03 16]

B	Name	Function	At reset	R	W
0	Port P1 direction register	0 : Port $P1_0$ input mode 1 : Port $P1_0$ output mode	0	×	○
1		0 : Port $P1_1$ input mode 1 : Port $P1_1$ output mode	0	×	○
2		0 : Port $P1_2$ input mode 1 : Port $P1_2$ output mode	0	×	○
3		0 : Port $P1_3$ input mode 1 : Port $P1_3$ output mode	0	×	○
4		0 : Port $P1_4$ input mode 1 : Port $P1_4$ output mode	0	×	○
5	Nothing is allocated for these bits. When these bits are read out, the values are undefined.		?	×	×
6			?	×	×
7			?	×	×

Fig. 3.5.4 Structure of Port P1 direction register

3.5 List of registers

Pull-up control register

b7 b6 b5 b4 b3 b2 b1 b0

Pull-up control register (PULL) [Address : 16_{16}]

B	Name	Function	At reset	R	W
0	$P0_0$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
1	$P0_1$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
2	$P0_2$, $P0_3$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
3	$P0_4$ – $P0_7$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
4	$P3_0$ – $P3_3$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
5	$P3_4$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
6	$P3_5$, $P3_6$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○
7	$P3_7$ pull-up control bit	0 : Pull-up Off 1 : Pull-up On	0	○	○

Notes 1: Pins set to output are disconnected from the pull-up control.
2: Keep setting the $P3_5$, $P3_6$ pull-up control bit to "1" (initial value: 0) for the 32-pin package versions.

Fig. 3.5.5 Structure of Pull-up control register

Port P1P3 control register

b7 b6 b5 b4 b3 b2 b1 b0

Port P1P3 control register (P1P3C) [Address : 17_{16}]

B	Name	Function	At reset	R	W
0	$P3_7$/INT_0 input level selection bit	0 : CMOS level 1 : TTL level	0	○	○
1	$P3_6$/INT_1 input level selection bit (**Note**)	0 : CMOS level 1 : TTL level	0	○	○
2	$P1_0$, $P1_2$,$P1_3$ input level selection bit	0 : CMOS level 1 : TTL level	0	○	○
3	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Note: Keep setting the $P3_6$/INT_1 input level selection bit to "0" (initial value) for the 32-pin package version.

Fig. 3.5.6 Structure of Port P1P3 control register

Transmit/Receive buffer register

b7 b6 b5 b4 b3 b2 b1 b0

Transmit/Receive buffer register (TB/RB) [Address : 18_{16}]

B	Function	At reset	R	W
0	The transmission data is written to or the receive data is read out from this buffer register. • At writing: A data is written to the transmit buffer register. • At reading: The contents of the receive buffer register are read out.	?	○	○
1		?	○	○
2		?	○	○
3		?	○	○
4		?	○	○
5		?	○	○
6		?	○	○
7		?	○	○

Note: The contents of transmit buffer register cannot be read out.
The data cannot be written to the receive buffer register.

Fig. 3.5.7 Structure of Transmit/Receive buffer register

Serial I/O1 status register

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O1 status register (SIO1STS) [Address : 19_{16}]

B	Name	Function	At reset	R	W
0	Transmit buffer empty flag (TBE)	0 : Buffer full 1 : Buffer empty	0	○	×
1	Receive buffer full flag (RBF)	0 : Buffer empty 1 : Buffer full	0	○	×
2	Transmit shift register shift completion flag (TSC)	0 : Transmit shift in progress 1 : Transmit shift completed	0	○	×
3	Overrun error flag (OE)	0 : No error 1 : Overrun error	0	○	×
4	Parity error flag (PE)	0 : No error 1 : Parity error	0	○	×
5	Framing error flag (FE)	0 : No error 1 : Framing error	0	○	×
6	Summing error flag (SE)	0 : (OE) ∪ (PE) ∪ (FE) = 0 1 : (OE) ∪ (PE) ∪ (FE) = 1	0	○	×
7	Nothing is allocated for this bit. This is a write disabled bit. When this bit is read out, the value is "1".		1	○	×

Fig. 3.5.8 Structure of Serial I/O1 status register

APPENDIX

3.5 List of registers

Serial I/O1 control register

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O1 control register (SIO1CON) [Address : 1A 16]

B	Name	Function	At reset	R	W
0	BRG count source selection bit (CSS)	0 : f(XIN) 1 : f(XIN)/4	0	○	○
1	Serial I/O1 synchronous clock selection bit (SCS)	When clock synchronous serial I/O is selected; 0: BRG output divided by 4 1: External clock input When UART is selected; 0: BRG output divided by 16 1: External clock input divided by 16	0	○	○
2	$\overline{SRDY1}$ output enable bit (SRDY)	0: P13 pin 1: $\overline{SRDY1}$ output pin	0	○	○
3	Transmit interrupt source selection bit (TIC)	0 : Interrupt when transmit buffer has emptied 1 : Interrupt when transmit shift operation is completed	0	○	○
4	Transmit enable bit (TE)	0 : Transmit disabled 1 : Transmit enabled	0	○	○
5	Receive enable bit (RE)	0 : Receive disabled 1 : Receive enabled	0	○	○
6	Serial I/O1 mode selection bit (SIOM)	0: Clock asynchronous (UART) serial I/O 1: Clock synchronous serial I/O	0	○	○
7	Serial I/O1 enable bit (SIOE)	0: Serial I/O1 disabled 1: Serial I/O1 enabled	0	○	○

Fig. 3.5.9 Structure of Serial I/O1 control register

UART control register

b7 b6 b5 b4 b3 b2 b1 b0

UART control register (UARTCON) [Address : 1B16]

B	Name	Function	At reset	R	W
0	Character length selection bit (CHAS)	0 : 8 bits 1 : 7 bits	0	○	○
1	Parity enable bit (PARE)	0 : Parity checking disabled 1 : Parity checking enabled	0	○	○
2	Parity selection bit (PARS)	0 : Even parity 1 : Odd parity	0	○	○
3	Stop bit length selection bit (STPS)	0 : 1 stop bit 1 : 2 stop bits	0	○	○
4	P11/TxD P-channel output disable bit (POFF)	In output mode 0 : CMOS output 1 : N-channel open-drain output	0	○	○
5	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "1".		1	○	×
6			1	○	×
7			1	○	×

Fig. 3.5.10 Structure of UART control register

Baud rate generator

b7	b6	b5	b4	b3	b2	b1	b0

Baud rate generator (BRG) [Address : 1C 16]

B	Function	At reset	R	W
0	Set a count value of baud rate generator.	?	○	○
1		?	○	○
2		?	○	○
3		?	○	○
4		?	○	○
5		?	○	○
6		?	○	○
7		?	○	○

Fig. 3.5.11 Structure of Baud rate generator

APPENDIX

3.5 List of registers

Timer A mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer A mode register (TAM) [Address : 1D $_{16}$]

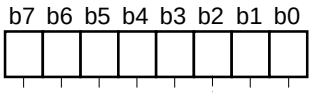
B	Name	Function	At reset	R	W
0	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
1			0	○	×
2			0	○	×
3			0	○	×
4	Timer A operating mode bits	b5 b4 0 0 : Timer mode 0 1 : Period measurement mode 1 0 : Event counter mode 1 1 : Pulse width HL continuously measurement mode	0	○	○
5			0	○	○
6	$CNTR_1$ active edge switch bit	The function depends on the operating mode. (Refer to Table 3.5.1)	0	○	○
7	Timer A count stop bit	0 : Count start 1 : Count stop	0	○	○

Fig. 3.5.12 Structure of Timer A mode register

Table 3.5.1 $CNTR_1$ active edge switch bit function

Timer A operating modes	$CNTR_1$ active edge switch bit	
Timer mode	"0"	$CNTR_1$ interrupt request occurrence: Falling edge ; No influence to timer A count
	"1"	$CNTR_1$ interrupt request occurrence: Rising edge ; No influence to timer A count
Period measurement mode	"0"	Pulse output start: Falling edge period measurement $CNTR_1$ interrupt request occurrence: Falling edge
	"1"	Pulse output start: Rising edge period measurement $CNTR_1$ interrupt request occurrence: Rising edge
Event counter mode	"0"	Timer A: Rising edge count $CNTR_1$ interrupt request occurrence: Falling edge
	"1"	Timer A: Falling edge count $CNTR_1$ interrupt request occurrence: Rising edge
Pulse width HL continuously measurement mode	"0"	$CNTR_1$ interrupt request occurrence: Falling edge and rising edge ; No influence to timer A count
	"1"	$CNTR_1$ interrupt request occurrence: Rising edge and falling edge ; No influence to timer A count

Timer A register (low-order, high-order)



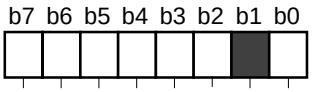
Timer A register (low-order, high-order) (TAL, TAH) [Address : 1E 16, 1F16]

B	Function	At reset	R	W
0	•Set a count value of timer A. •The value set in this register is written to both timer A and timer A latch at the same time. •When this register is read out, the timer A's count value is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Notes 1: Be sure to write to/read out both the low-order of timer A (TAL) and the high-order of timer A (TAH).
2: Read the high-order of timer A (TAH) first, and the high-order of timer A (TAL) next.
3: Write to the low-order of timer A (TAL) first, and the high-order of timer A (TAH) next.
4: Do not write to them during read, and do not read out them during write.

Fig. 3.5.13 Structure of Timer A register

Timer Y, Z mode register



Timer Y, Z mode register (TYZM) [Address : 20 16)

B	Name	Function	At reset	R	W
0	Timer Y operating mode bit	0 : Timer mode 1 : Programmable waveform generation mode	0	○	○
1	Nothing is allocated for this bit. This is a write disabled bit. When this bit is read out, the value is "0".		0	○	×
2	Timer Y write control bit (**Note**)	0 : Write to latch and timer simultaneously 1 : Write to only latch	0	○	○
3	Timer Y count stop bit	0 : Count start 1 : Count stop	0	○	○
4	Timer Z operating mode bits	b5 b4 0 0 : Timer mode 0 1 : Programmable waveform generation mode 1 0 : Programmable one-shot generation mode 1 1 : Programmable wait one-shot generation mode	0	○	○
5			0	○	○
6	Timer Z write control bit (**Note**)	0 : Write to latch and timer simultaneously 1 : Write to only latch	0	○	○
7	Timer Z count stop bit	0 : Count start 1 : Count stop	0	○	○

Note: When modes other than the timer mode, set these bits to "1".

Fig. 3.5.14 Structure of Timer Y, Z mode register

Prescaler Y, Prescaler Z

b7 b6 b5 b4 b3 b2 b1 b0

Prescaler Y (PREY) [Address : 21 16]
Prescaler Z (PREZ) [Address : 25 16]

B	Function	At reset	R	W
0	•Set a count value of each prescaler. •The value set in this register is written to both each prescaler and the corresponding prescaler latch at the same time. •When this register is read out, the count value of the corresponding prescaler is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Fig. 3.5.15 Structure of Prescaler Y, Prescaler Z

Timer Y secondary, Timer Z secondary

b7 b6 b5 b4 b3 b2 b1 b0

Timer Y secondary, Timer Z secondary (TYS, TZS) [Address : 22 16, 26 16]

B	Function	At reset	R	W
0	•Set a count value of the corresponding timer. •The value set in this register is written to the corresponding secondary latch at the same time. •These are read disabled bits. When these bits are read out, the values are undefined.	1	×	○
1		1	×	○
2		1	×	○
3		1	×	○
4		1	×	○
5		1	×	○
6		1	×	○
7		1	×	○

Fig. 3.5.16 Structure of Timer Y secondary, Timer Z secondary

Timer Y primary, Timer Z primary

b7 b6 b5 b4 b3 b2 b1 b0

Timer Y primary, Timer Z primary (TYP, TZP) [Address : 23 16, 2716]

B	Function	At reset	R	W
0	•Set a count value of the corresponding timer. •When the corresponding timer is stopped, the value set in this register is written to both the corresponding primary latch and the corresponding timer at the same time. •When the corresponding timer is operating, the value set in this register is written as follows; timer write control bit = 0: the value is written to both the corresponding primary latch and the corresponding timer at the same time. timer write control bit = 1: the value is written to the corresponding primary latch. •When these bits are read out, the count value of the corresponding timer is read out (Note).	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Note: The primary count value is read out at the primary interval, the secondary count value is read out at the secondary interval.

Fig. 3.5.17 Structure of Timer Y primary, Timer Z primary

Timer Y, Z waveform output control register

b7 b6 b5 b4 b3 b2 b1 b0

Timer Y, Z waveform output control register (PUM) [Address : 24 16]

B	Name	Function	At reset	R	W
0	Timer Y primary waveform extension control bit	0 : Waveform not extended 1 : Waveform extended	0	○	○
1	Timer Y secondary waveform extension control bit	0 : Waveform not extended 1 : Waveform extended	0	○	○
2	Timer Z primary waveform extension control bit	0 : Waveform not extended 1 : Waveform extended	0	○	○
3	Timer Z secondary waveform extension control bit	0 : Waveform not extended 1 : Waveform extended	0	○	○
4	Timer Y output level latch	0 : “L” output 1 : “H” output	0	○	○
5	Timer Z output level latch	0 : “L” output 1 : “H” output	0	○	○
6	INT_0 pin one-shot trigger control bit (**Note**)	0 : INT_0 pin one-shot trigger invalid 1 : INT_0 pin one-shot trigger valid	0	○	○
7	INT_0 pin one-shot trigger active edge selection bit (**Note**)	0 : Falling edge trigger 1 : Rising edge trigger	0	○	○

Note: Stop timer Z to change the values of these bits.

Fig. 3.5.18 Structure of Timer Y, Z waveform output control register

3.5 List of registers

Prescaler 1

b7 b6 b5 b4 b3 b2 b1 b0

Prescaler 1 (PRE1) [Address : 28_{16}]

B	Function	At reset	R	W
0	•Set a count value of prescaler 1. •The value set in this register is written to both prescaler 1 and the prescaler 1 latch at the same time. •When this register is read out, the count value of the prescaler 1 latch is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Fig. 3.5.19 Structure of Prescaler 1

Timer 1

b7 b6 b5 b4 b3 b2 b1 b0

Timer 1 (T1) [Address : 29_{16}]

B	Function	At reset	R	W
0	•Set a count value of timer 1. •The value set in this register is written to both timer 1 and timer 1 latch at the same time. •When this register is read out, the timer 1's count value is read out.	1	○	○
1		0	○	○
2		0	○	○
3		0	○	○
4		0	○	○
5		0	○	○
6		0	○	○
7		0	○	○

Fig. 3.5.20 Structure of Timer 1

One-shot start register

b7 b6 b5 b4 b3 b2 b1 b0

One-shot start register (ONS) [Address : 2A 16]

B	Name	Function	At reset	R	W
0	Timer Z one-shot start bit	0 : One-shot stop 1 : One-shot start	0	○	○
1	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
2			0	○	×
3			0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Fig. 3.5.21 Structure of One-shot start register

APPENDIX

3.5 List of registers

Timer X mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer X mode register (TXM) [Address : 2B 16]

B	Name	Function	At reset	R	W
0	Timer X operating mode bits	b1 b0 0 0 : Timer mode 0 1 : Pulse output mode 1 0 : Event counter mode 1 1 : Pulse width measurement mode	0	○	○
1			0	○	○
2	$CNTR_0$ active edge switch bit	The function depends on the operating mode. (Refer to Table 3.5.2)	0	○	○
3	Timer X count stop bit	0 : Count start 1 : Count stop	0	○	○
4	$P0_3/TX_{OUT}$ output valid bit	0 : Output invalid (I/O port) 1 : Output valid (Inverted $CNTR_0$ output)	0	○	○
5	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
6			0	○	×
7			0	○	×

Fig. 3.5.22 Structure of Timer X mode register

Table 3.5.2 $CNTR_0$ active edge switch bit function

Timer X operating modes	$CNTR_0$ active edge switch bit (bit 2 of address $2B_{16}$) contents	
Timer mode	"0"	$CNTR_0$ interrupt request occurrence: Falling edge ; No influence to timer count
	"1"	$CNTR_0$ interrupt request occurrence: Rising edge ; No influence to timer count
Pulse output mode	"0"	Pulse output start: Beginning at "H" level $CNTR_0$ interrupt request occurrence: Falling edge
	"1"	Pulse output start: Beginning at "L" level $CNTR_0$ interrupt request occurrence: Rising edge
Event counter mode	"0"	Timer X: Rising edge count $CNTR_0$ interrupt request occurrence: Falling edge
	"1"	Timer X: Falling edge count $CNTR_0$ interrupt request occurrence: Rising edge
Pulse width measurement mode	"0"	Timer X: "H" level width measurement $CNTR_0$ interrupt request occurrence: Falling edge
	"1"	Timer X: "L" level width measurement $CNTR_0$ interrupt request occurrence: Rising edge

Prescaler X

b7 b6 b5 b4 b3 b2 b1 b0

Prescaler X (PREX) [Address : 2C 16]

B	Function	At reset	R	W
0	•Set a count value of prescaler X. •The value set in this register is written to both prescaler X and the prescaler X latch at the same time. •When this register is read out, the count value of the prescaler X is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Fig. 3.5.23 Structure of Prescaler X

Timer X

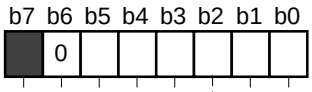
b7 b6 b5 b4 b3 b2 b1 b0

Timer X (TX) [Address : 2D 16]

B	Function	At reset	R	W
0	•Set a count value of timer X. •The value set in this register is written to both timer X and timer X latch at the same time. •When this register is read out, the timer X's count value is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Fig. 3.5.24 Structure of Timer X

Timer count source set register



Timer count source set register (TCSS) [Address : 2E 16]

B	Name	Function	At reset	R	W
0	Timer X count source selection bits	b1 b0 0 0 : f(XIN)/16 0 1 : f(XIN)/2 1 0 : f(XIN) (**Note 1**) 1 1 : Not available	0	○	○
1			0	○	○
2	Timer Y count source selection bits	b3 b2 0 0 : f(XIN)/16 0 1 : f(XIN)/2 1 0 : Ring oscillator output (**Note 2**) 1 1 : Not available	0	○	○
3			0	○	○
4	Timer Z count source selection bits	b5 b4 0 0 : f(XIN)/16 0 1 : f(XIN)/2 1 0 : Timer Y underflow 1 1 : Not available	0	○	○
5			0	○	○
6	Fix this bit to “0”.		0	○	○
7	Nothing is allocated for this bit. This is a write disabled bit. When this bit is read out, the value is “0”.		0	○	×

Notes 1: f(XIN) can be used as timer X count source only when using a ceramic oscillator or ring oscillator.
Do not use it at RC oscillation.
2: System operates using a ring oscillator as a count source by setting the ring oscillator to oscillation enabled by bit 3 of CPUM.

Fig. 3.5.25 Structure of Timer count source set register

Serial I/O2 control register

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O2 control register (SIO2CON) [Address : 30_{16}]

B	Name	Function	At reset	R	W
0	Internal synchronous clock selection bits	b2 b1 b0 0 0 0 : f(X_{IN})/8 0 0 1 : f(X_{IN})/16 0 1 0 : f(X_{IN})/32 0 1 1 : f(X_{IN})/64 1 1 0 : f(X_{IN})/128 1 1 1 : f(X_{IN})/256	0	○	○
1			0	○	○
2			0	○	○
3	S_{DATA2} pin selection bit (**Note**)	0 : I/O port / S_{DATA2} input 1 : S_{DATA2} output	0	○	○
4	Nothing is allocated for this bit. This is a write disabled bit. When this bit is read out, the value is "0".		0	○	×
5	Transfer direction selection bit	0 : LSB first 1 : MSB first	0	○	○
6	S_{CLK2} pin selection bit	0 : External clock (S_{CLK2} is input) 1 : Internal clock (S_{CLK2} is output)	0	○	○
7	Transmit / receive shift completion flag	0 : shift in progress 1 : shift completed	0	○	×

Note: When using it as a S_{DATA} input, set the port $P1_3$ direction register bit to "0".

Fig. 3.5.26 Structure of Serial I/O2 control register

Serial I/O2 register

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O2 register (SIO2) [Address : 31 $_{16}$]

B	Function	At reset	R	W
0	A shift register for serial transmission and reception. • At transmitting : Set a transmission data. • At receiving : A reception data is stored.	?	○	○
1		?	○	○
2		?	○	○
3		?	○	○
4		?	○	○
5		?	○	○
6		?	○	○
7		?	○	○

Fig. 3.5.27 Structure of Serial I/O2 register

A-D control register

b7 b6 b5 b4 b3 b2 b1 b0

A-D control register (ADCON) [Address : 34_{16}]

B	Name	Function	At reset	R	W
0	Analog input pin selection bits	b2 b1 b0 0 0 0 : $P2_0/AN_0$ 0 0 1 : $P2_1/AN_1$ 0 1 0 : $P2_2/AN_2$ 0 1 1 : $P2_3/AN_3$ 1 0 0 : $P2_4/AN_4$ 1 0 1 : $P2_5/AN_5$ 1 1 0 : $P2_6/AN_6$ (**Note**) 1 1 1 : $P2_7/AN_7$ (**Note**)	0	○	○
1			0	○	○
2			0	○	○
3	Nothing is allocated for this bit. This is a write disabled bit. When this bit is read out, the value is "0".		0	○	×
4	AD conversion completion bit	0 : Conversion in progress 1 : Conversion completed	1	○	*
5	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
6			0	○	×
7			0	○	×

Note: These can be used only for the 36-pin package versions.

*: This bit can be cleared to "0" by program, but cannot be set to "1".

Fig. 3.5.28 Structure of A-D control register

A-D conversion register (low-order)

b7 b6 b5 b4 b3 b2 b1 b0

A-D conversion register (low-order) (ADL) [Address : 35_{16}]

B	Function	At reset	R	W
0	The read-only register in which the A-D conversion's results are stored. < 8-bit read> b7 b0 b9 b8 b7 b6 b5 b4 b3 b2 < 10-bit read> b7 b0 b7 b6 b5 b4 b3 b2 b1 b0	?	○	×
1		?	○	×
2		?	○	×
3		?	○	×
4		?	○	×
5		?	○	×
6		?	○	×
7		?	○	×

Fig. 3.5.29 Structure of A-D conversion register (low-order)

A-D conversion register (high-order)

b7 b6 b5 b4 b3 b2 b1 b0

A-D conversion register (high-order) (ADH) [Address : 36 16]

B	Function	At reset	R	W
0	The read-only register in which the A-D conversion's results are stored. < 10-bit read> b7 b0 b9 b8	?	○	×
1		?	○	×
2	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".	?	○	×
3		?	○	×
4		?	○	×
5		?	○	×
6		?	○	×
7		?	○	×

Fig. 3.5.30 Structure of A-D conversion register (high-order)

MISRG

b7 b6 b5 b4 b3 b2 b1 b0

MISRG [Address : 38_{16}]

B	Name	Function	At reset	R	W
0	Oscillation stabilization time set bit after release of the STP instruction	0 : Set "01_{16}" in timer 1, and "FF_{16}" in prescaler 1 automatically 1 : Not set automatically	0	○	○
1	Ceramic or RC oscillation stop detection function active bit	0 : Detection function inactive 1 : Detection function active	0	○	○
2	These are reserved bits. Do not write "1" to these bits.		0	○	×
3			0	○	×
4	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
5			0	○	×
6			0	○	×
7	Oscillation stop detection status bit	0 : Oscillation stop not detected 1 : Oscillation stop detected	(**Note**)	○	×

Note: "0" at normal reset
"1" at reset by detecting the oscillation stop

Fig. 3.5.31 Structure of MISRG

Watchdog timer control register

b7 b6 b5 b4 b3 b2 b1 b0

Watchdog timer control register (WDTCON) [Address : 39 16]

B	Name	Function	At reset	R	W
0	Watchdog timer H (The high-order 6 bits are read-only bits.)		1	○	×
1			1	○	×
2			1	○	×
3			1	○	×
4			1	○	×
5			1	○	×
6	STP instruction disable bit	0 : STP instruction enabled 1 : STP instruction disabled	0	○	○
7	Watchdog timer H count source selection bit	0 : Watchdog timer L underflow 1 : f(X_{IN})/16	0	○	○

Fig. 3.5.32 Structure of Watchdog timer control register

Interrupt edge selection register

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt edge selection register (INTEDGE) [Address : 3A 16]

B	Name	Function	At reset	R	W
0	INT_0 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
1	INT_1 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
2	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
3			0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7	$P0_0$ key-on wakeup enable bit	0 : Key-on wakeup enabled 1 : Key-on wakeup disabled	0	○	○

Fig. 3.5.33 Structure of Interrupt edge selection register

CPU mode register

b7 b6 b5 b4 b3 b2 b1 b0

CPU mode register (CPUM) [Address : 3B 16]

B	Name	Function	At reset	R	W
0	Processor mode bits (**Note 1**)	b1 b0 0 0 : Single-chip mode 0 1 : Not available 1 0 : Not available 1 1 : Not available	0	○	○
1			0	○	○
2	Stack page selection bit	0 : 0 page 1 : 1 page	0	○	○
3	Ring oscillator oscillation control bit	0 : Ring oscillator oscillation enabled 1 : Ring oscillator oscillation stop	0	○	○
4	X_{IN} oscillation control bit	0 : Ceramic or RC oscillation enabled 1 : Ceramic or RC oscillation stop	0	○	○
5	Oscillation mode selection bit (**Note 1**)	0 : Ceramic oscillation 1 : RC oscillation	0	○	○
6	Clock division ratio selection bits	b7 b6 0 0 : $\phi = f(X_{IN})/2$ (high-speed mode) 0 1 : $\phi = f(X_{IN})/8$ (middle-speed mode) 1 0 : Applied from ring oscillator 1 1 : $\phi = f(X_{IN})$ (double-speed mode) (**Note 2**)	0	○	○
7			1	○	○

Notes 1: The bit can be rewritten only once after releasing reset. After rewriting it is disable to write any data to the bit. However, by reset the bit is initialized and can be rewritten, again.
(It is not disable to write any data to the bit for emulator MCU “M37540RSS”.)
2: These bits are used only when a ceramic oscillation is selected.
Do not use these when an RC oscillation is selected.

Fig. 3.5.34 Structure of CPU mode register

3.5 List of registers

Interrupt request register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt request register 1 (IREQ1) [Address : 3C 16]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Serial I/O1 transmit interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	INT_0 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	INT_1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	Key-on wake up interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	$CNTR_0$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	$CNTR_1$ interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻

✻: These bits can be cleared to "0" by program, but cannot be set to "1".

Fig. 3.5.35 Structure of Interrupt request register 1

Interrupt request register 2

b7 b6 b5 b4 b3 b2 b1 b0

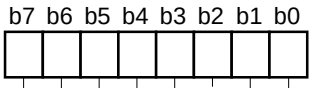
Interrupt request register 2 (IREQ2) [Address : 3D 16]

B	Name	Function	At reset	R	W
0	Timer Y interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Timer Z interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	Timer A interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	AD converter interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "0".		0	○	×
7			0	○	×

✻: These bits can be cleared to "0" by program, but cannot be set to "1".

Fig. 3.5.36 Structure of Interrupt request register 2

Interrupt control register 1

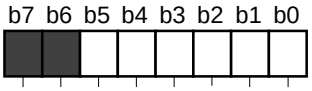


Interrupt control register 1 (ICON1) [Address : 3E 16]

B	Name	Function	At reset	R	W
0	Serial I/O1 receive interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Serial I/O1 transmit interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	INT_0 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	INT_1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Key-on wake up interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	$CNTR_0$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	$CNTR_1$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Fig. 3.5.37 Structure of Interrupt control register 1

Interrupt control register 2



Interrupt control register 2 (ICON2) [Address : 3F 16]

B	Name	Function	At reset	R	W
0	Timer Y interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer Z interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer A interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	AD conversion interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are “0”.		0	○	×
7			0	○	×

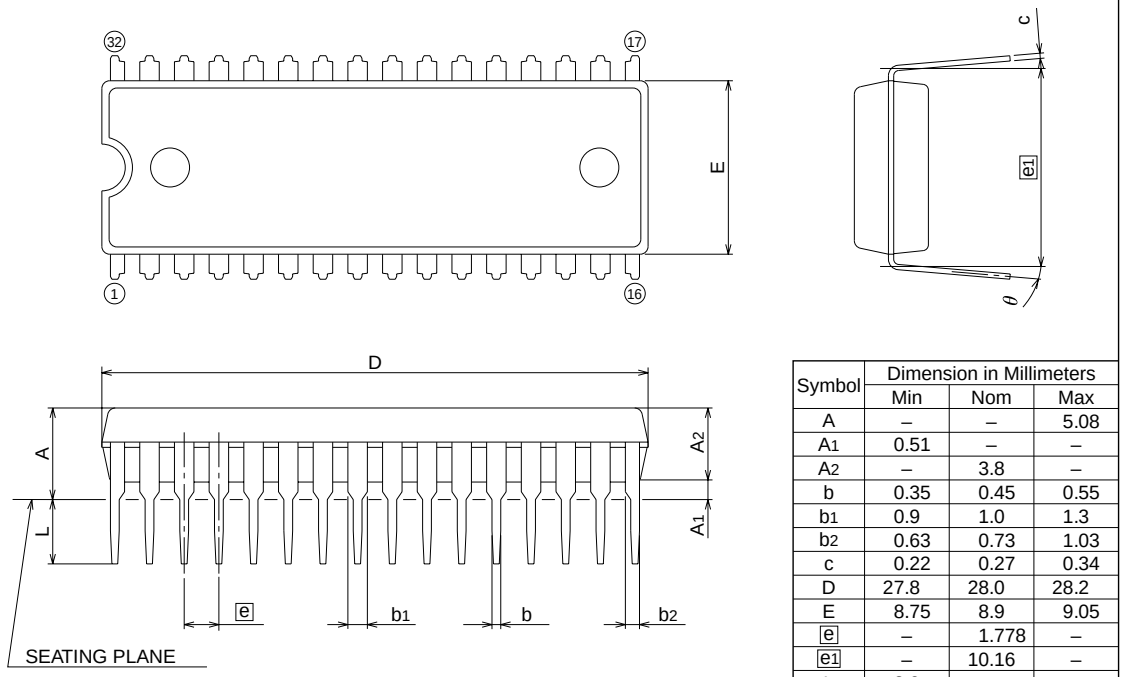
Fig. 3.5.38 Structure of Interrupt control register 2

3.6 Package outline

32P4B (MMP)

Plastic 32pin 400mil SDIP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
SDIP32-P-400-1.78	–	2.2	Alloy 42/Cu Alloy



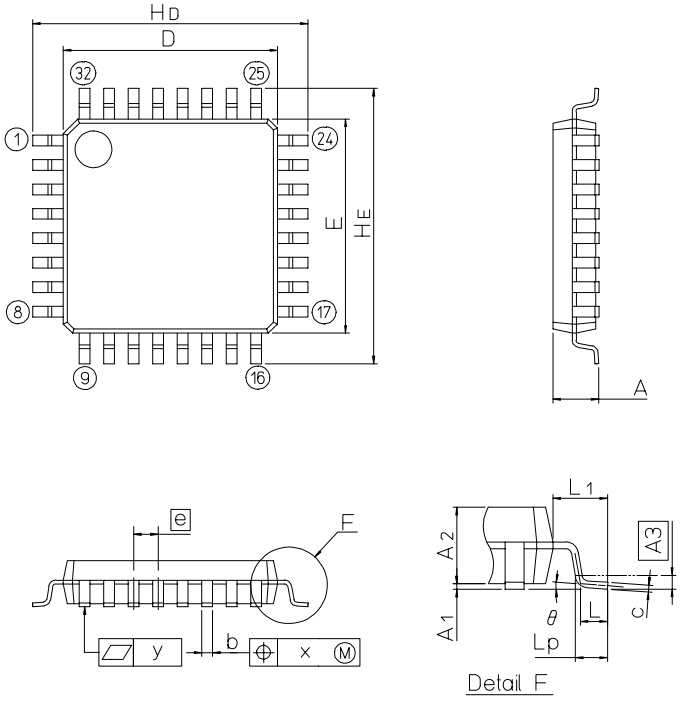
Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	–	–	5.08
A_1	0.51	–	–
A_2	–	3.8	–
b	0.35	0.45	0.55
b_1	0.9	1.0	1.3
b_2	0.63	0.73	1.03
c	0.22	0.27	0.34
D	27.8	28.0	28.2
E	8.75	8.9	9.05
e	–	1.778	–
e1	–	10.16	–
L	3.0	–	–
θ	0°	–	15°

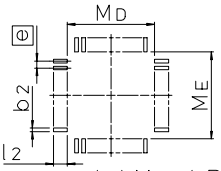
32P6U–A (MMP)

Plastic 32pin 7×7mm body LQFP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
LQFP32-P-0707-0.80	——		Cu Alloy

Scale: 4/1





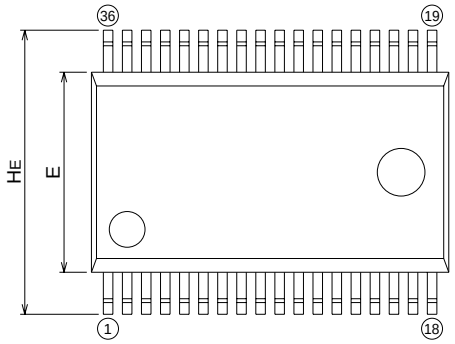
Recommended Mount Pad

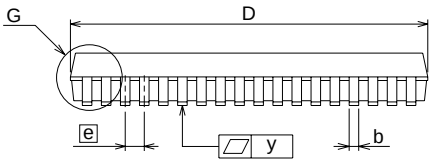
Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	—	—	1.7
A_1	0	0.1	0.2
A_2	—	1.4	—
b	0.32	0.37	0.45
c	0.105	0.125	0.175
D	6.9	7.0	7.1
E	6.9	7.0	7.1
e	—	0.8	—
H_D	8.8	9.0	9.2
H_E	8.8	9.0	9.2
L	0.3	0.5	0.7
L_1	—	1.0	—
Lp	0.45	0.6	0.75
A3	—	0.25	—
x	—	—	0.2
y	—	—	0.1
θ	0°	—	8°
b_2	—	0.5	—
l_2	1.0	—	—
M_D	—	7.4	—
M_E	—	7.4	—

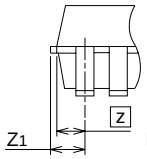
36P2R-A (MMP)

Plastic 36pin 450mil SSOP

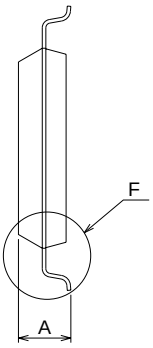
EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
SSOP36-P-450-0.80	–	0.53	Alloy 42

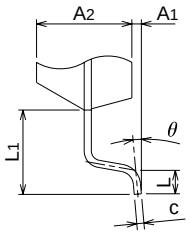




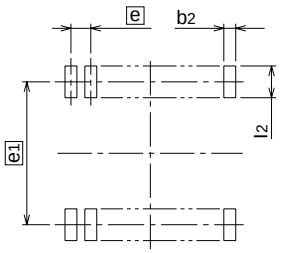


Detail G





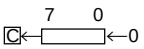
Detail F



Recommended Mount Pad

Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	–	–	2.4
A_1	0.05	–	–
A_2	–	2.0	–
b	0.35	0.4	0.5
c	0.13	0.15	0.2
D	14.8	15.0	15.2
E	8.2	8.4	8.6
e	–	0.8	–
H_E	11.63	11.93	12.23
L	0.3	0.5	0.7
L_1	–	1.765	–
z	–	0.7	–
Z_1	–	–	0.85
y	–	–	0.15
θ	0°	–	10°
b_2	–	0.5	–
e1	–	11.43	–
l_2	1.27	–	–

3.7 Machine instructions

Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT, A, R			ZP			BIT, ZP, R		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
ADC (Note 1) (Note 5)	When T = 0 A ← A + M + C When T = 1 M(X) ← M(X) + M + C	When T = 0, this instruction adds the contents M, C, and A; and stores the results in A and C. When T = 1, this instruction adds the contents of M(X), M and C; and stores the results in M(X) and C. When T=1, the contents of A remain unchanged, but the contents of status flags are changed. M(X) represents the contents of memory where is indicated by X.				69	2	2							65	3	2			
AND (Note 1)	When T = 0 A ← A ∧ M When T = 1 M(X) ← M(X) ∧ M	When T = 0, this instruction transfers the contents of A and M to the ALU which performs a bit-wise AND operation and stores the result back in A. When T = 1, this instruction transfers the contents M(X) and M to the ALU which performs a bit-wise AND operation and stores the results back in M(X). When T = 1, the contents of A remain unchanged, but status flags are changed. M(X) represents the contents of memory where is indicated by X.				29	2	2							25	3	2			
ASL		This instruction shifts the content of A or M by one bit to the left, with bit 0 always being set to 0 and bit 7 of A or M always being contained in C.							0A	2	1				06	5	2			
BBC (Note 4)	Ai or Mi = 0?	This instruction tests the designated bit i of M or A and takes a branch if the bit is 0. The branch address is specified by a relative address. If the bit is 1, next instruction is executed.										13 + 20i	4	2				17 + 20i	5	3
BBS (Note 4)	Ai or Mi = 1?	This instruction tests the designated bit i of the M or A and takes a branch if the bit is 1. The branch address is specified by a relative address. If the bit is 0, next instruction is executed.										03 + 20i	4	2				07 + 20i	5	3
BCC (Note 4)	C = 0?	This instruction takes a branch to the appointed address if C is 0. The branch address is specified by a relative address. If C is 1, the next instruction is executed.																		
BCS (Note 4)	C = 1?	This instruction takes a branch to the appointed address if C is 1. The branch address is specified by a relative address. If C is 0, the next instruction is executed.																		
BEQ (Note 4)	Z = 1?	This instruction takes a branch to the appointed address when Z is 1. The branch address is specified by a relative address. If Z is 0, the next instruction is executed.																		
BIT	A ∧ M	This instruction takes a bit-wise logical AND of A and M contents; however, the contents of A and M are not modified. The contents of N, V, Z are changed, but the contents of A, M remain unchanged.													24	3	2			
BMI (Note 4)	N = 1?	This instruction takes a branch to the appointed address when N is 1. The branch address is specified by a relative address. If N is 0, the next instruction is executed.																		
BNE (Note 4)	Z = 0?	This instruction takes a branch to the appointed address if Z is 0. The branch address is specified by a relative address. If Z is 1, the next instruction is executed.																		

Addressing mode																											Processor status register							
ZP, X			ZP, Y			ABS			ABS, X			ABS, Y			IND			ZP, IND			IND, X			IND, Y			REL			SP			7	6	5	4	3	2	1	0
OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	N	V	T	B	D	I	Z	C
75	4	2				6D	4	3	7D	5	3	79	5	3							61	6	2	71	6	2							N	V	•	•	•	•	Z	C
35	4	2				2D	4	3	3D	5	3	39	5	3							21	6	2	31	6	2							N	•	•	•	•	•	Z	•
16	6	2				0E	6	3	1E	7	3																						N	•	•	•	•	•	Z	C
																																	•	•	•	•	•	•	•	•
																																	•	•	•	•	•	•	•	•
																											90	2	2				•	•	•	•	•	•	•	•
																											B0	2	2				•	•	•	•	•	•	•	•
																											F0	2	2				•	•	•	•	•	•	•	•
						2C	4	3																									M7	M6	•	•	•	•	Z	•
																											30	2	2				•	•	•	•	•	•	•	•
																											D0	2	2				•	•	•	•	•	•	•	•

APPENDIX

3.7 Machine instructions

Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT, A			ZP			BIT, ZP		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
BPL (Note 4)	N = 0?	This instruction takes a branch to the appointed address if N is 0. The branch address is specified by a relative address. If N is 1, the next instruction is executed.																		
BRA	PC ← PC ± offset	This instruction branches to the appointed address. The branch address is specified by a relative address.																		
BRK	B ← 1 (PC) ← (PC) + 2 M(S) ← PC_H S ← S – 1 M(S) ← PC_L S ← S – 1 M(S) ← PS S ← S – 1 I← 1 PC_L ← AD_L PC_H ← AD_H	When the BRK instruction is executed, the CPU pushes the current PC contents onto the stack. The BADRS designated in the interrupt vector table is stored into the PC.	00	7	1															
BVC (Note 4)	V = 0?	This instruction takes a branch to the appointed address if V is 0. The branch address is specified by a relative address. If V is 1, the next instruction is executed.																		
BVS (Note 4)	V = 1?	This instruction takes a branch to the appointed address when V is 1. The branch address is specified by a relative address. When V is 0, the next instruction is executed.																		
CLB	Ai or Mi ← 0	This instruction clears the designated bit i of A or M.										1B + 20i	2	1				1F + 20i	5	2
CLC	C ← 0	This instruction clears C.	18	2	1															
CLD	D ← 0	This instruction clears D.	D8	2	1															
CLI	I ← 0	This instruction clears I.	58	2	1															
CLT	T ← 0	This instruction clears T.	12	2	1															
CLV	V ← 0	This instruction clears V.	B8	2	1															
CMP (Note 3)	When T = 0 A – M When T = 1 M(X) – M	When T = 0, this instruction subtracts the contents of M from the contents of A. The result is not stored and the contents of A or M are not modified. When T = 1, the CMP subtracts the contents of M from the contents of M(X). The result is not stored and the contents of X, M, and A are not modified. M(X) represents the contents of memory where is indicated by X.				C9	2	2							C5	3	2			
COM	M ← $\overline{M}$	This instruction takes the one's complement of the contents of M and stores the result in M.													44	5	2			
CPX	X – M	This instruction subtracts the contents of M from the contents of X. The result is not stored and the contents of X and M are not modified.				E0	2	2							E4	3	2			
CPY	Y – M	This instruction subtracts the contents of M from the contents of Y. The result is not stored and the contents of Y and M are not modified.				C0	2	2							C4	3	2			
DEC	A ← A – 1 or M ← M – 1	This instruction subtracts 1 from the contents of A or M.							1A	2	1				C6	5	2			

Addressing mode																												Processor status register							
ZP, X			ZP, Y			ABS			ABS, X			ABS, Y			IND			ZP, IND			IND, X			IND, Y			REL			SP			7	6	5	4	3	2	1	0
OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	N	V	T	B	D	I	Z	C
																											10	2	2				•	•	•	•	•	•	•	•
																											80	4	2				•	•	•	•	•	•	•	•
																																		•	•	•	1	•	1	•	•
																											50	2	2				•	•	•	•	•	•	•	•
																											70	2	2				•	•	•	•	•	•	•	•
																																		•	•	•	•	•	•	•	•
																																		•	•	•	•	•	•	•	0
																																		•	•	•	•	0	•	•	•
																																		•	•	•	•	•	0	•	•
																																		•	•	0	•	•	•	•	•
																																		•	0	•	•	•	•	•	•
D5	4	2				CD	4	3	DD	5	3	D9	5	3							C1	6	2	D1	6	2							N	•	•	•	•	•	Z	C
																																		N	•	•	•	•	•	Z	•
						EC	4	3																											N	•	•	•	•	•	Z	C
						CC	4	3																											N	•	•	•	•	•	Z	C
D6	6	2				CE	6	3	DE	7	3																							N	•	•	•	•	•	Z	•

APPENDIX

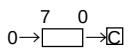
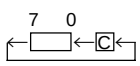
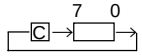
3.7 Machine instructions

Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT, A			ZP			BIT, ZP		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
DEX	X ← X – 1	This instruction subtracts one from the current contents of X.	CA	2	1															
DEY	Y ← Y – 1	This instruction subtracts one from the current contents of Y.	88	2	1															
DIV	A ← (M(zz + X + 1), M(zz + X)) / A M(S) ← one's complement of Remainder S ← S – 1	This instruction divides the 16-bit data in M(zz+(X)) (low-order byte) and M(zz+(X)+1) (high-order byte) by the contents of A. The quotient is stored in A and the one's complement of the remainder is pushed onto the stack.																		
EOR (Note 1)	When T = 0 A ← A ∀ M When T = 1 M(X) ← M(X) ∀ M	When T = 0, this instruction transfers the contents of the M and A to the ALU which performs a bit-wise Exclusive OR, and stores the result in A. When T = 1, the contents of M(X) and M are transferred to the ALU, which performs a bit-wise Exclusive OR and stores the results in M(X). The contents of A remain unchanged, but status flags are changed. M(X) represents the contents of memory where is indicated by X.				49	2	2							45	3	2			
INC	A ← A + 1 or M ← M + 1	This instruction adds one to the contents of A or M.							3A	2	1				E6	5	2			
INX	X ← X + 1	This instruction adds one to the contents of X.	E8	2	1															
INY	Y ← Y + 1	This instruction adds one to the contents of Y.	C8	2	1															
JMP	If addressing mode is ABS PC_L ← AD_L PC_H ← AD_H If addressing mode is IND PC_L ← M (AD_H, AD_L) PC_H ← M (AD_H, AD_L + 1) If addressing mode is ZP, IND PC_L ← M(00, AD_L) PC_H ← M(00, AD_L + 1)	This instruction jumps to the address designated by the following three addressing modes: Absolute Indirect Absolute Zero Page Indirect Absolute																		
JSR	M(S) ← PC_H S ← S – 1 M(S) ← PC_L S ← S – 1 After executing the above, if addressing mode is ABS, PC_L ← AD_L PC_H ← AD_H if addressing mode is SP, PC_L ← AD_L PC_H ← FF If addressing mode is ZP, IND, PC_L ← M(00, AD_L) PC_H ← M(00, AD_L + 1)	This instruction stores the contents of the PC in the stack, then jumps to the address designated by the following addressing modes: Absolute Special Page Zero Page Indirect Absolute																		
LDA (Note 2)	When T = 0 A ← M When T = 1 M(X) ← M	When T = 0, this instruction transfers the contents of M to A. When T = 1, this instruction transfers the contents of M to (M(X)). The contents of A remain unchanged, but status flags are changed. M(X) represents the contents of memory where is indicated by X.				A9	2	2							A5	3	2			
LDM	M ← nn	This instruction loads the immediate value in M.													3C	4	3			
LDX	X ← M	This instruction loads the contents of M in X.				A2	2	2							A6	3	2			
LDY	Y ← M	This instruction loads the contents of M in Y.				A0	2	2							A4	3	2			

Addressing mode																													Processor status register							
ZP, X			ZP, Y			ABS			ABS, X			ABS, Y			IND			ZP, IND			IND, X			IND, Y			REL			SP			7	6	5	4	3	2	1	0
OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	N	V	T	B	D	I	Z	C
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
E2	16	2																															•	•	•	•	•	•	•	•
55	4	2				4D	4	3	5D	5	3	59	5	3							41	6	2	51	6	2							N	•	•	•	•	•	Z	•
F6	6	2				EE	6	3	FE	7	3																						N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
						4C	3	3							6C	5	3	B2	4	2													•	•	•	•	•	•	•	•
						20	6	3										02	7	2										22	5	2	•	•	•	•	•	•	•	•
B5	4	2				AD	4	3	BD	5	3	B9	5	3							A1	6	2	B1	6	2							N	•	•	•	•	•	Z	•
																																	•	•	•	•	•	•	•	•
			B6	4	2	AE	4	3				BE	5	3																			N	•	•	•	•	•	Z	•
B4	4	2				AC	4	3	BC	5	3																						N	•	•	•	•	•	Z	•

APPENDIX

3.7 Machine instructions

Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT, A			ZP			BIT, ZP		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
LSR		This instruction shifts either A or M one bit to the right such that bit 7 of the result always is set to 0, and the bit 0 is stored in C.							4A	2	1				46	5	2			
MUL	M(S) • A ← A ✻ M(zz + X) S ← S – 1	This instruction multiply Accumulator with the memory specified by the Zero Page X address mode and stores the high-order byte of the result on the Stack and the low-order byte in A.																		
NOP	PC ← PC + 1	This instruction adds one to the PC but does no otheroperation.	EA	2	1															
ORA (Note 1)	When T = 0 A ← A V M When T = 1 M(X) ← M(X) V M	When T = 0, this instruction transfers the contents of A and M to the ALU which performs a bit-wise “OR”, and stores the result in A. When T = 1, this instruction transfers the contents of M(X) and the M to the ALU which performs a bit-wise OR, and stores the result in M(X). The contents of A remain unchanged, but status flags are changed. M(X) represents the contents of memory where is indicated by X.				09	2	2							05	3	2			
PHA	M(S) ← A S ← S – 1	This instruction pushes the contents of A to the memory location designated by S, and decrements the contents of S by one.	48	3	1															
PHP	M(S) ← PS S ← S – 1	This instruction pushes the contents of PS to the memory location designated by S and decrements the contents of S by one.	08	3	1															
PLA	S ← S + 1 A ← M(S)	This instruction increments S by one and stores the contents of the memory designated by S in A.	68	4	1															
PLP	S ← S + 1 PS ← M(S)	This instruction increments S by one and stores the contents of the memory location designated by S in PS.	28	4	1															
ROL		This instruction shifts either A or M one bit left through C. C is stored in bit 0 and bit 7 is stored in C.							2A	2	1				26	5	2			
ROR		This instruction shifts either A or M one bit right through C. C is stored in bit 7 and bit 0 is stored in C.							6A	2	1				66	5	2			
RRF		This instruction rotates 4 bits of the M content to the right.													82	8	2			
RTI	S ← S + 1 PS ← M(S) S ← S + 1 PCL ← M(S) S ← S + 1 PCH ← M(S)	This instruction increments S by one, and stores the contents of the memory location designated by S in PS. S is again incremented by one and stores the contents of the memory location designated by S in PCL. S is again incremented by one and stores the contents of memory location designated by S in PCH.	40	6	1															
RTS	S ← S + 1 PCL ← M(S) S ← S + 1 PCH ← M(S) (PC) ← (PC) + 1	This instruction increments S by one and stores the contents of the memory location designated by S in PCL. S is again incremented by one and the contents of the memory location is stored in PCH. PC is incremented by 1.	60	6	1															

Addressing mode																												Processor status register							
ZP, X			ZP, Y			ABS			ABS, X			ABS, Y			IND			ZP, IND			IND, X			IND, Y			REL			SP			7	6	5	4	3	2	1	0
OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	N	V	T	B	D	I	Z	C
56	6	2				4E	6	3	5E	7	3																						0	•	•	•	•	•	Z	C
62	15	2																															•	•	•	•	•	•	•	•
																																	•	•	•	•	•	•	•	•
15	4	2				0D	4	3	1D	5	3	19	5	3							01	6	2	11	6	2							N	•	•	•	•	•	Z	•
																																	•	•	•	•	•	•	•	•
																																	•	•	•	•	•	•	•	•
																																	N	•	•	•	•	•	Z	•
																																	(Value saved in stack)							
36	6	2				2E	6	3	3E	7	3																						N	•	•	•	•	•	Z	C
76	6	2				6E	6	3	7E	7	3																						N	•	•	•	•	•	Z	C
																																	•	•	•	•	•	•	•	•
																																	(Value saved in stack)							
																																	•	•	•	•	•	•	•	•

APPENDIX

3.7 Machine instructions

Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT, A			ZP			BIT, ZP		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
SBC (Note 1) (Note 5)	When T = 0 A ← A – M – C When T = 1 M(X) ← M(X) – M – C	When T = 0, this instruction subtracts the value of M and the complement of C from A, and stores the results in A and C. When T = 1, the instruction subtracts the contents of M and the complement of C from the contents of M(X), and stores the results in M(X) and C. A remain unchanged, but status flag are changed. M(X) represents the contents of memory where is indicated by X.				E9	2	2							E5	3	2			
SEB	Ai or Mi ← 1	This instruction sets the designated bit i of A or M.										0B + 20i	2	1				0F + 20i	5	2
SEC	C ← 1	This instruction sets C.	38	2	1															
SED	D ← 1	This instruction set D.	F8	2	1															
SEI	I ← 1	This instruction set I.	78	2	1															
SET	T ← 1	This instruction set T.	32	2	1															
STA	M ← A	This instruction stores the contents of A in M. The contents of A does not change.													85	4	2			
STP		This instruction resets the oscillation control F/F and the oscillation stops. Reset or interrupt input is needed to wake up from this mode.	42	2	1															
STX	M ← X	This instruction stores the contents of X in M. The contents of X does not change.													86	4	2			
STY	M ← Y	This instruction stores the contents of Y in M. The contents of Y does not change.													84	4	2			
TAX	X ← A	This instruction stores the contents of A in X. The contents of A does not change.	AA	2	1															
TAY	Y ← A	This instruction stores the contents of A in Y. The contents of A does not change.	A8	2	1															
TST	M = 0?	This instruction tests whether the contents of M are “0” or not and modifies the N and Z.													64	3	2			
TSX	X ← S	This instruction transfers the contents of S in X.	BA	2	1															
TXA	A ← X	This instruction stores the contents of X in A.	8A	2	1															
TXS	S ← X	This instruction stores the contents of X in S.	9A	2	1															
TYA	A ← Y	This instruction stores the contents of Y in A.	98	2	1															
WIT		The WIT instruction stops the internal clock but not the oscillation of the oscillation circuit is not stopped. CPU starts its function after the Timer X over flows (comes to the terminal count). All registers or internal memory contents except Timer X will not change during this mode. (Of course needs VDD).	C2	2	1															

Notes 1 : The number of cycles “n” is increased by 3 when T is 1.
2 : The number of cycles “n” is increased by 2 when T is 1.
3 : The number of cycles “n” is increased by 1 when T is 1.
4 : The number of cycles “n” is increased by 2 when branching has occurred.
5 : N, V, and Z flags are invalid in decimal operation mode.

Addressing mode																												Processor status register							
ZP, X			ZP, Y			ABS			ABS, X			ABS, Y			IND			ZP, IND			IND, X			IND, Y			REL			SP			7	6	5	4	3	2	1	0
OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	N	V	T	B	D	I	Z	C
F5	4	2				ED	4	3	FD	5	3	F9	5	3							E1	6	2	F1	6	2							N	V	•	•	•	•	Z	C
																																	•	•	•	•	•	•	•	•
																																	•	•	•	•	•	•	•	1
																																	•	•	•	•	1	•	•	•
																																	•	•	•	•	•	1	•	•
																																	•	•	1	•	•	•	•	•
95	5	2				8D	5	3	9D	6	3	99	6	3							81	7	2	91	7	2							•	•	•	•	•	•	•	•
																																	•	•	•	•	•	•	•	•
			96	5	2	8E	5	3																									•	•	•	•	•	•	•	•
94	5	2				8C	5	3																									•	•	•	•	•	•	•	•
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
																																	•	•	•	•	•	•	•	•
																																	N	•	•	•	•	•	Z	•
																																	•	•	•	•	•	•	•	•

APPENDIX

3.7 Machine instructions

Symbol	Contents	Symbol	Contents
IMP	Implied addressing mode	+	Addition
IMM	Immediate addressing mode	–	Subtraction
A	Accumulator or Accumulator addressing mode	✻	Multiplication
BIT, A	Accumulator bit addressing mode	/	Division
BIT, A, R	Accumulator bit relative addressing mode	Λ	Logical OR
ZP	Zero page addressing mode	V	Logical AND
BIT, ZP	Zero page bit addressing mode	¥	Logical exclusive OR
BIT, ZP, R	Zero page bit relative addressing mode	—	Negation
ZP, X	Zero page X addressing mode	←	Shows direction of data flow
ZP, Y	Zero page Y addressing mode	X	Index register X
ABS	Absolute addressing mode	Y	Index register Y
ABS, X	Absolute X addressing mode	S	Stack pointer
ABS, Y	Absolute Y addressing mode	PC	Program counter
IND	Indirect absolute addressing mode	PS	Processor status register
		PC_H	8 high-order bits of program counter
ZP, IND	Zero page indirect absolute addressing mode	PC_L	8 low-order bits of program counter
		AD_H	8 high-order bits of address
IND, X	Indirect X addressing mode	AD_L	8 low-order bits of address
IND, Y	Indirect Y addressing mode	FF	FF in Hexadecimal notation
REL	Relative addressing mode	nn	Immediate value
SP	Special page addressing mode	zz	Zero page address
C	Carry flag	M	Memory specified by address designation of any addressing mode
Z	Zero flag		
I	Interrupt disable flag	M(X)	Memory of address indicated by contents of index register X
D	Decimal mode flag		
B	Break flag	M(S)	Memory of address indicated by contents of stack pointer
T	X-modified arithmetic mode flag		
V	Overflow flag	$M(AD_H, AD_L)$	Contents of memory at address indicated by AD_H and AD_L, in AD_H is 8 high-order bits and AD_L is 8 low-order bits.
N	Negative flag		
		$M(00, AD_L)$	Contents of address indicated by zero page AD_L
		Ai	Bit i (i = 0 to 7) of accumulator
		Mi	Bit i (i = 0 to 7) of memory
		OP	Opcode
		n	Number of cycles
		#	Number of bytes

3.8 List of instruction code

D3 – D0 / Hexadecimal notation / D7 – D4		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	BRK	ORA IND, X	JSR ZP, IND	BBS 0, A	—	ORA ZP	ASL ZP	BBS 0, ZP	PHP	ORA IMM	ASL A	SEB 0, A	—	ORA ABS	ASL ABS	SEB 0, ZP
0001	1	BPL	ORA IND, Y	CLT	BBC 0, A	—	ORA ZP, X	ASL ZP, X	BBC 0, ZP	CLC	ORA ABS, Y	DEC A	CLB 0, A	—	ORA ABS, X	ASL ABS, X	CLB 0, ZP
0010	2	JSR ABS	AND IND, X	JSR SP	BBS 1, A	BIT ZP	AND ZP	ROL ZP	BBS 1, ZP	PLP	AND IMM	ROL A	SEB 1, A	BIT ABS	AND ABS	ROL ABS	SEB 1, ZP
0011	3	BMI	AND IND, Y	SET	BBC 1, A	—	AND ZP, X	ROL ZP, X	BBC 1, ZP	SEC	AND ABS, Y	INC A	CLB 1, A	LDM ZP	AND ABS, X	ROL ABS, X	CLB 1, ZP
0100	4	RTI	EOR IND, X	STP	BBS 2, A	COM ZP	EOR ZP	LSR ZP	BBS 2, ZP	PHA	EOR IMM	LSR A	SEB 2, A	JMP ABS	EOR ABS	LSR ABS	SEB 2, ZP
0101	5	BVC	EOR IND, Y	—	BBC 2, A	—	EOR ZP, X	LSR ZP, X	BBC 2, ZP	CLI	EOR ABS, Y	—	CLB 2, A	—	EOR ABS, X	LSR ABS, X	CLB 2, ZP
0110	6	RTS	ADC IND, X	MUL ZP, X	BBS 3, A	TST ZP	ADC ZP	ROR ZP	BBS 3, ZP	PLA	ADC IMM	ROR A	SEB 3, A	JMP IND	ADC ABS	ROR ABS	SEB 3, ZP
0111	7	BVS	ADC IND, Y	—	BBC 3, A	—	ADC ZP, X	ROR ZP, X	BBC 3, ZP	SEI	ADC ABS, Y	—	CLB 3, A	—	ADC ABS, X	ROR ABS, X	CLB 3, ZP
1000	8	BRA	STA IND, X	RRF ZP	BBS 4, A	STY ZP	STA ZP	STX ZP	BBS 4, ZP	DEY	—	TXA	SEB 4, A	STY ABS	STA ABS	STX ABS	SEB 4, ZP
1001	9	BCC	STA IND, Y	—	BBC 4, A	STY ZP, X	STA ZP, X	STX ZP, Y	BBC 4, ZP	TYA	STA ABS, Y	TXS	CLB 4, A	—	STA ABS, X	—	CLB 4, ZP
1010	A	LDY IMM	LDA IND, X	LDX IMM	BBS 5, A	LDY ZP	LDA ZP	LDX ZP	BBS 5, ZP	TAY	LDA IMM	TAX	SEB 5, A	LDY ABS	LDA ABS	LDX ABS	SEB 5, ZP
1011	B	BCS	LDA IND, Y	JMP ZP, IND	BBC 5, A	LDY ZP, X	LDA ZP, X	LDX ZP, Y	BBC 5, ZP	CLV	LDA ABS, Y	TSX	CLB 5, A	LDY ABS, X	LDA ABS, X	LDX ABS, Y	CLB 5, ZP
1100	C	CPY IMM	CMP IND, X	WIT	BBS 6, A	CPY ZP	CMP ZP	DEC ZP	BBS 6, ZP	INY	CMP IMM	DEX	SEB 6, A	CPY ABS	CMP ABS	DEC ABS	SEB 6, ZP
1101	D	BNE	CMP IND, Y	—	BBC 6, A	—	CMP ZP, X	DEC ZP, X	BBC 6, ZP	CLD	CMP ABS, Y	—	CLB 6, A	—	CMP ABS, X	DEC ABS, X	CLB 6, ZP
1110	E	CPX IMM	SBC IND, X	DIV ZP, X	BBS 7, A	CPX ZP	SBC ZP	INC ZP	BBS 7, ZP	INX	SBC IMM	NOP	SEB 7, A	CPX ABS	SBC ABS	INC ABS	SEB 7, ZP
1111	F	BEQ	SBC IND, Y	—	BBC 7, A	—	SBC ZP, X	INC ZP, X	BBC 7, ZP	SED	SBC ABS, Y	—	CLB 7, A	—	SBC ABS, X	INC ABS, X	CLB 7, ZP

: 3-byte instruction

: 2-byte instruction

: 1-byte instruction

3.9 SFR memory map

0000_{16}	Port P0 (P0)
0001_{16}	Port P0 direction register (P0D)
0002_{16}	Port P1 (P1)
0003_{16}	Port P1 direction register (P1D)
0004_{16}	Port P2 (P2)
0005_{16}	Port P2 direction register (P2D)
0006_{16}	Port P3 (P3)
0007_{16}	Port P3 direction register (P3D)
0008_{16}	
0009_{16}	
$000A_{16}$	
$000B_{16}$	
$000C_{16}$	
$000D_{16}$	
$000E_{16}$	
$000F_{16}$	
0010_{16}	
0011_{16}	
0012_{16}	
0013_{16}	
0014_{16}	
0015_{16}	
0016_{16}	Pull-up control register (PULL)
0017_{16}	Port P1P3 control register (P1P3C)
0018_{16}	Transmit/Receive buffer register (TB/RB)
0019_{16}	Serial I/O1 status register (SIO1STS)
$001A_{16}$	Serial I/O1 control register (SIO1CON)
$001B_{16}$	UART control register (UARTCON)
$001C_{16}$	Baud rate generator (BRG)
$001D_{16}$	Timer A mode register (TAM)
$001E_{16}$	Timer A (low-order) (TAL)
$001F_{16}$	Timer A (high-order) (TAH)

0020_{16}	Timer Y, Z mode register (TYZM)
0021_{16}	Prescaler Y (PREY)
0022_{16}	Timer Y secondary (TYS)
0023_{16}	Timer Y primary (TYP)
0024_{16}	Timer Y, Z waveform output control register (PUM)
0025_{16}	Prescaler Z (PREZ)
0026_{16}	Timer Z secondary (TZS)
0027_{16}	Timer Z primary (TZP)
0028_{16}	Prescaler 1 (PRE1)
0029_{16}	Timer 1 (T1)
$002A_{16}$	One-shot start register (ONS)
$002B_{16}$	Timer X mode register (TXM)
$002C_{16}$	Prescaler X (PREX)
$002D_{16}$	Timer X (TX)
$002E_{16}$	Timer count source set register (TCSS)
$002F_{16}$	
0030_{16}	Serial I/O2 control register (SIO2CON)
0031_{16}	Serial I/O2 register (SIO2)
0032_{16}	
0033_{16}	
0034_{16}	A-D control register (ADCON)
0035_{16}	A-D conversion register (low-order) (ADL)
0036_{16}	A-D conversion register (high-order) (ADH)
0037_{16}	
0038_{16}	MISRG
0039_{16}	Watchdog timer control register (WDTCON)
$003A_{16}$	Interrupt edge selection register (INTEDGE)
$003B_{16}$	CPU mode register (CPUM)
$003C_{16}$	Interrupt request register 1 (IREQ1)
$003D_{16}$	Interrupt request register 2 (IREQ2)
$003E_{16}$	Interrupt control register 1 (ICON1)
$003F_{16}$	Interrupt control register 2 (ICON2)

3.10 Pin configurations

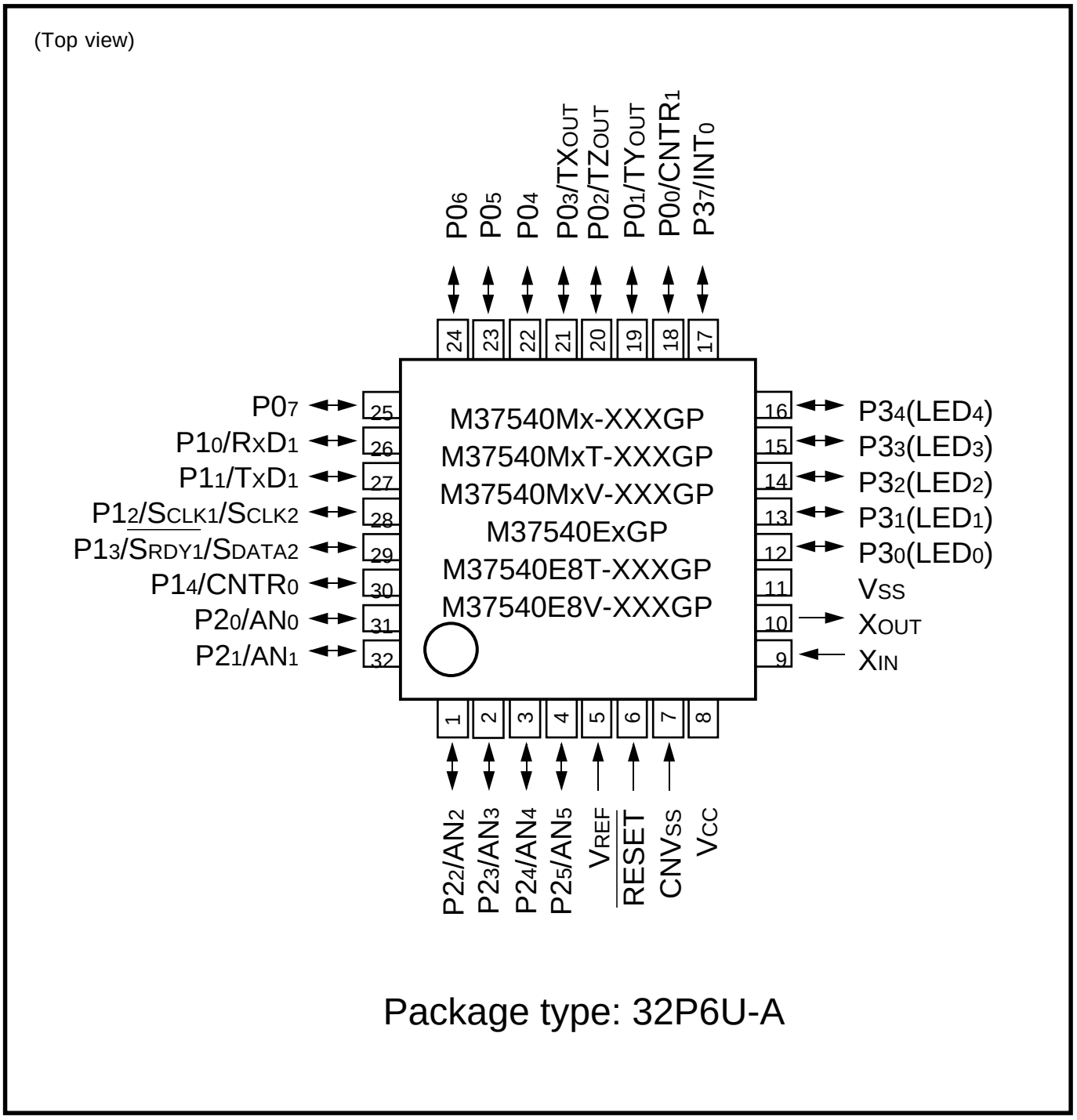


Fig. 3.10.1 32P6U-A package pin configuration

APPENDIX

3.10 Pin configurations

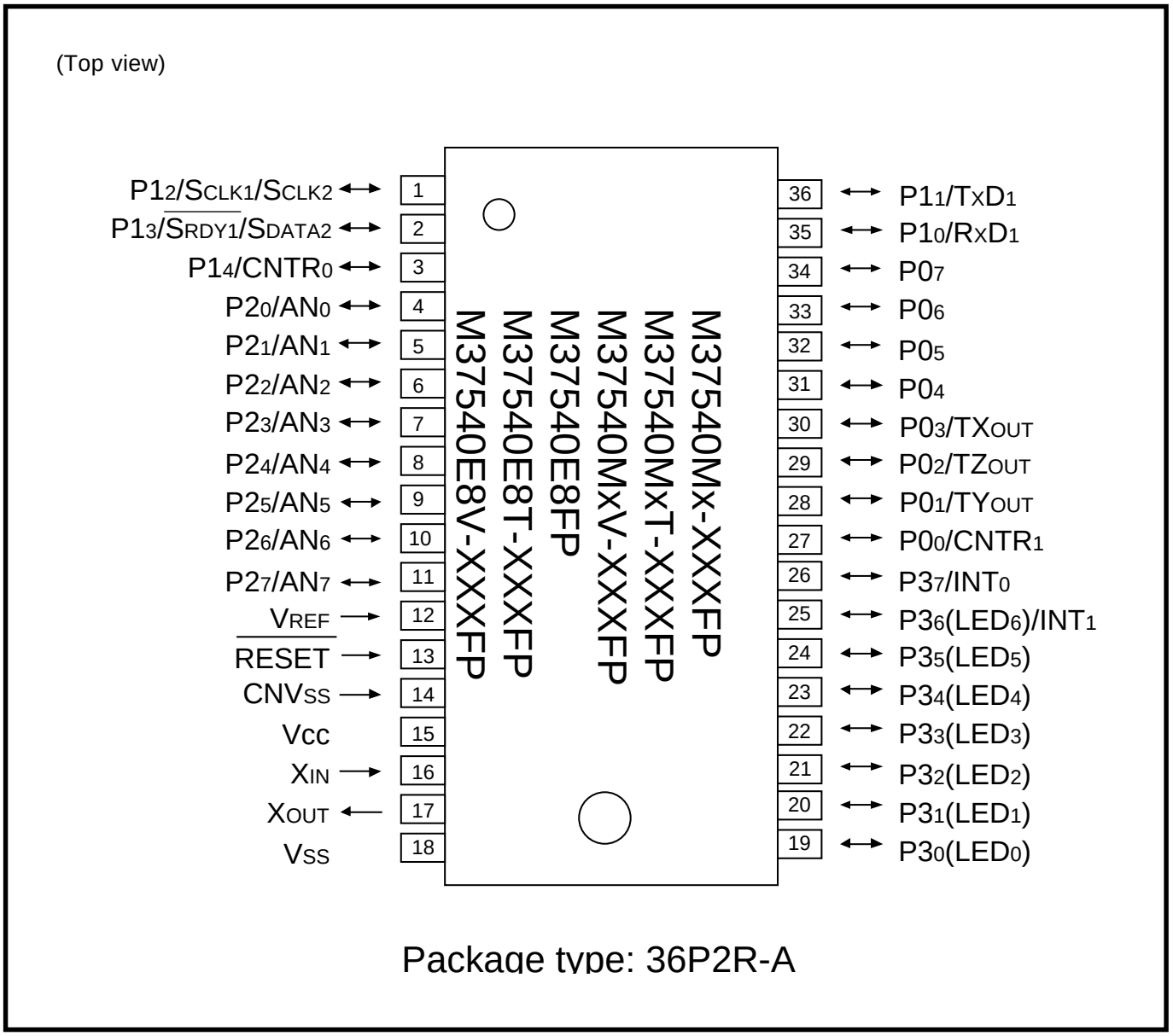
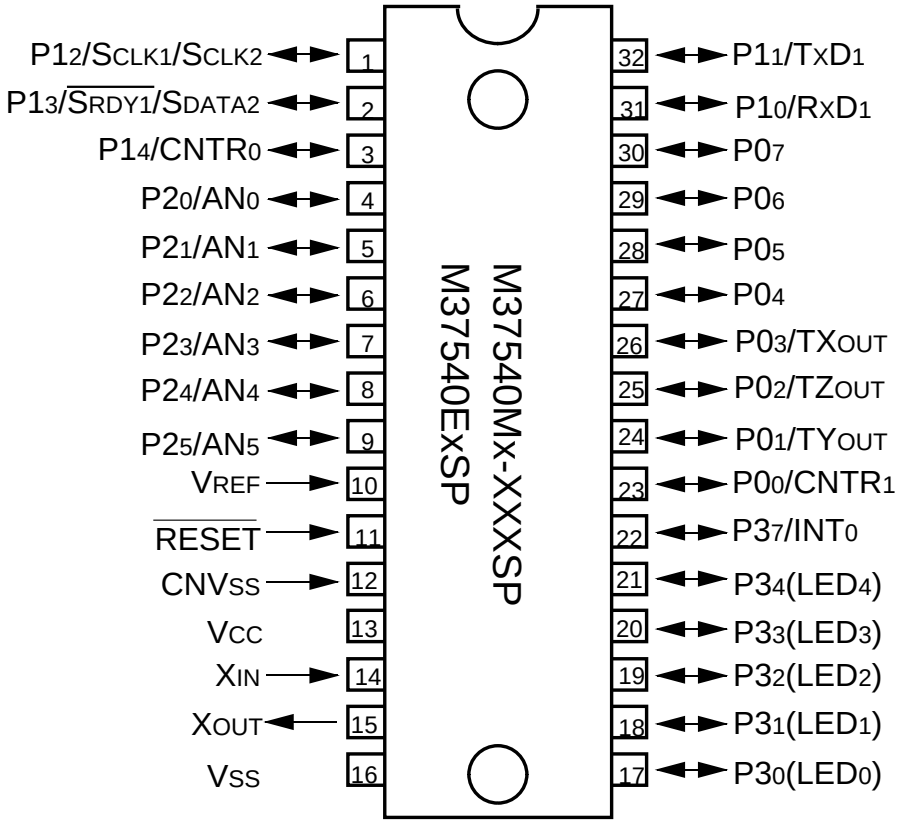


Fig. 3.10.2 36P2R-A package pin configuration

(Top view)

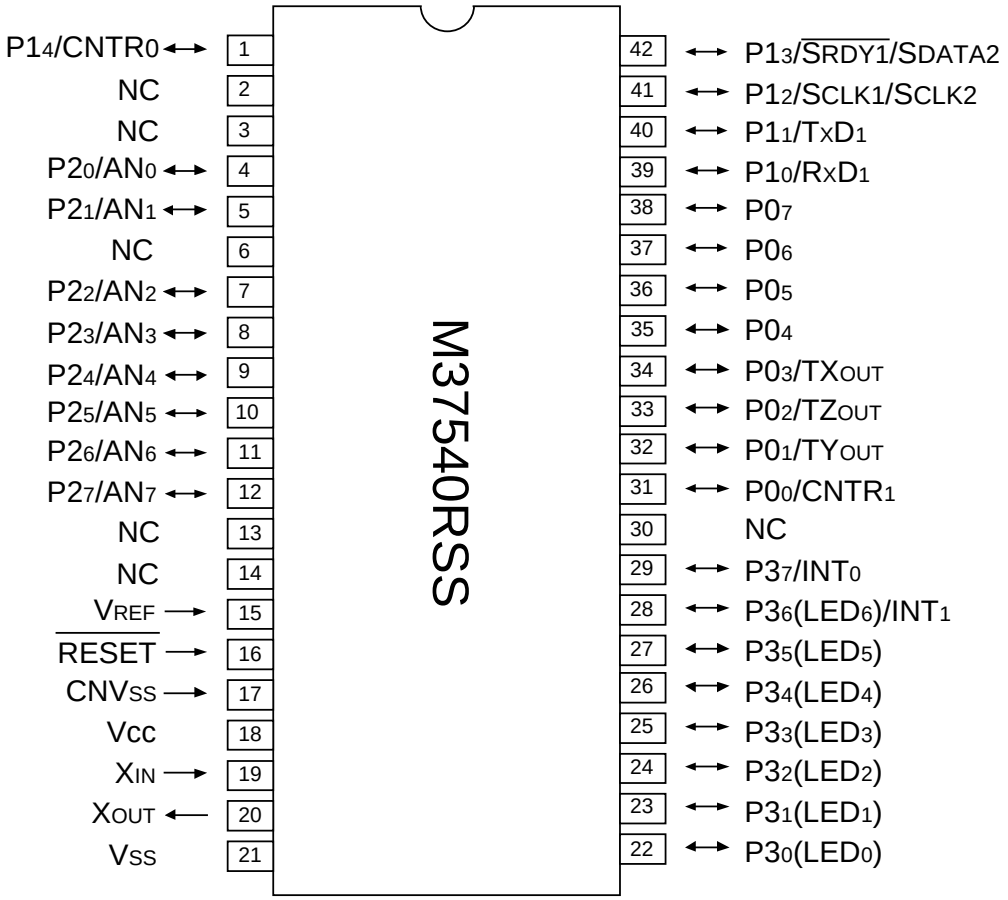


Package type: 32P4B

Fig. 3.10.3 32P4B package pin configuration

3.10 Pin configurations

(Top view)



Outline 42S1M

Fig. 3.10.4 42S1M package pin configuration

3.11 Differences between 7540 Group and 7531 Group

Table 3.11.1 shows the differences between 7540 Group and 7531 Group.

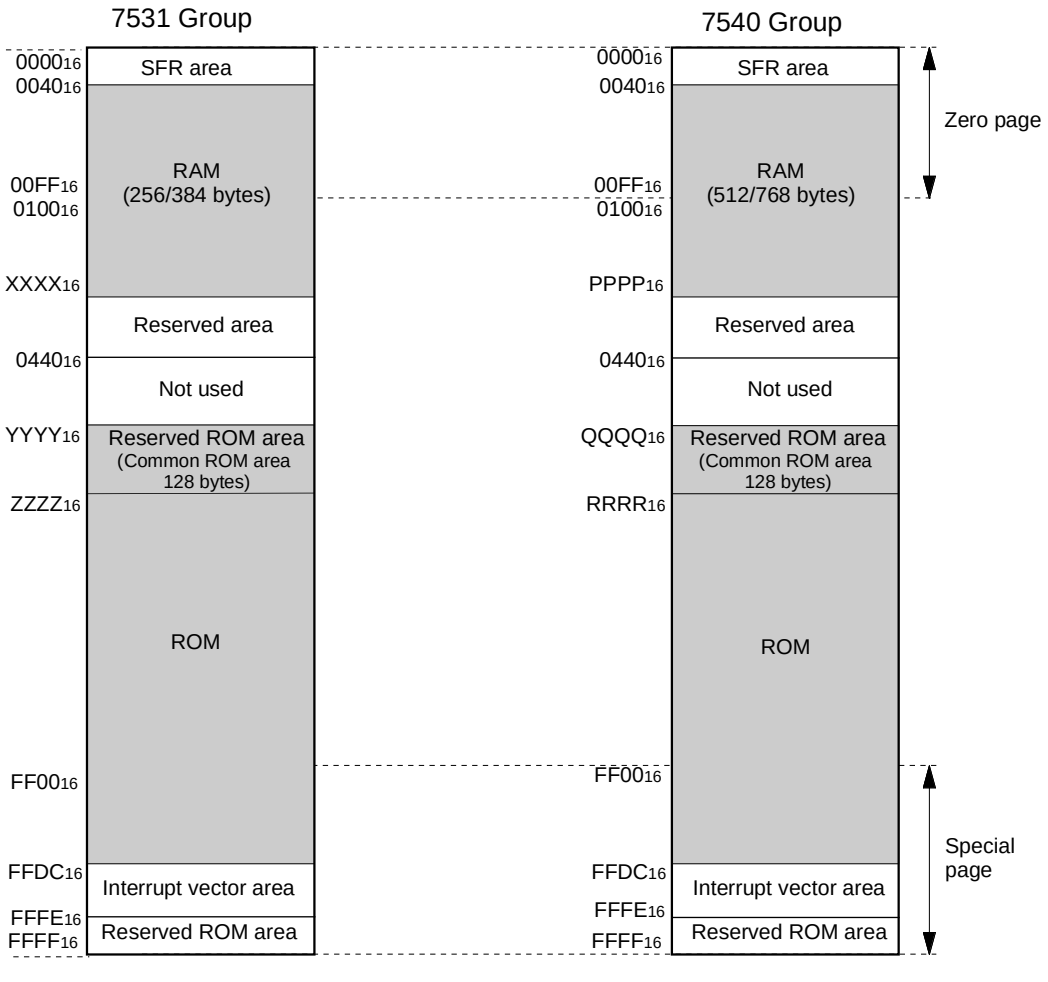
Table 3.11.1 Differences between 7540 Group and 7531 Group (Performance overview)

Parameter		7540 Group	7531 Group
Number of basic instructions		71 (DIV, MUL instruction added)	69
Memory sizes	ROM	16 to 32 K bytes	8 to 16 K bytes
	RAM	512 to 768 bytes	256 to 384 bytes
Input/Output ports		Initial value: 00_{16} (Ports P0 and P3 pull-up Off)	Initial value: FF_{16} (Ports P0 and P3 pull-up On)
Interrupt sources	32-pin version	14 sources, 14 vector (4 for external)	11 sources, 8 vector (3 for external)
	36-pin version	15 sources, 15 vector (5 for external)	12 sources, 8 vector (4 for external)
16-bit timer		1 (Timer A)	————
8-bit timer		3 (Timer 1, X, Y, Z)	3 (Timer 1, 2, X)
Serial I/O1		Clock synchronous/UART	UART only
Clock generation circuit		Cecamic oscillator/ Quartz-crystal oscillator/ RC oscillation/ Ring oscillator oscillation	Cecamic oscillator/ Quartz-crystal oscillator/ RC oscillation
Oscillation stop detection circuit		1	————

APPENDIX

3.11 Differences between 7540 Group and 7531 Group

Figure 3.11.1 shows the memory map of 7540 Group and 7531 Group.



RAM area

RAM capacity (bytes)	Address XXXX₁₆
256	013F₁₆
384	01BF₁₆

RAM area

RAM capacity (bytes)	Address PPPP₁₆
512	023F₁₆
768	033F₁₆

ROM area

ROM capacity (bytes)	Address YYYY₁₆	Address ZZZZ₁₆
8192	E000₁₆	E080₁₆
16384	C000₁₆	C080₁₆

ROM area

ROM capacity (bytes)	Address QQQQ₁₆	Address RRRR₁₆
16384	C000₁₆	C080₁₆
32768	8000₁₆	8080₁₆

Fig. 3.11.1 Memory map of 7540 Group and 7531 Group

Figure 3.11.2 shows the memory map of interrupt vector area of 7540 Group and 7531 Group.

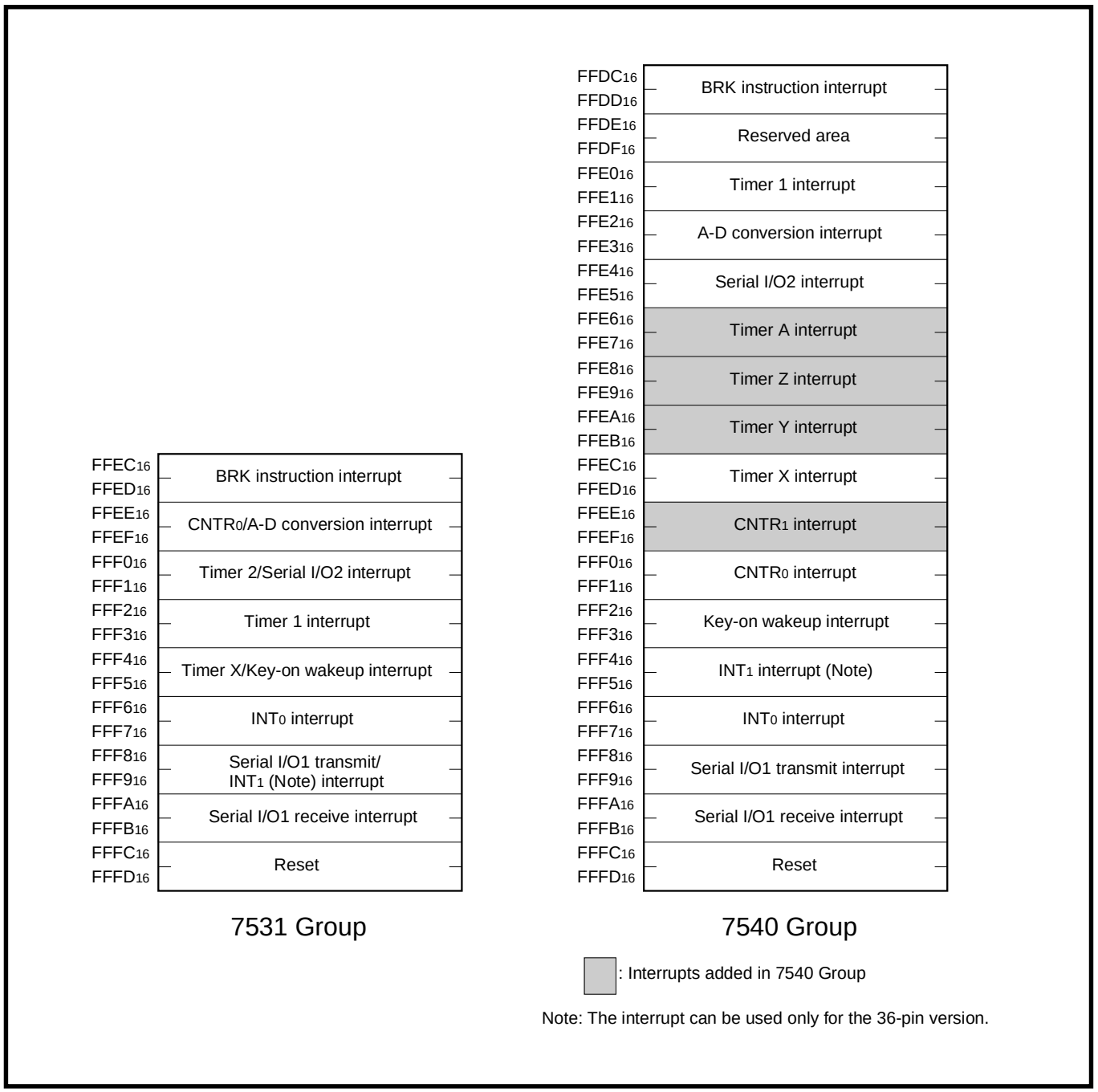


Fig. 3.11.2 Memory map of interrupt vector area of 7540 Group and 7531 Group

Figure 3.11.3 shows the timer function of 7540 Group and 7531 Group.

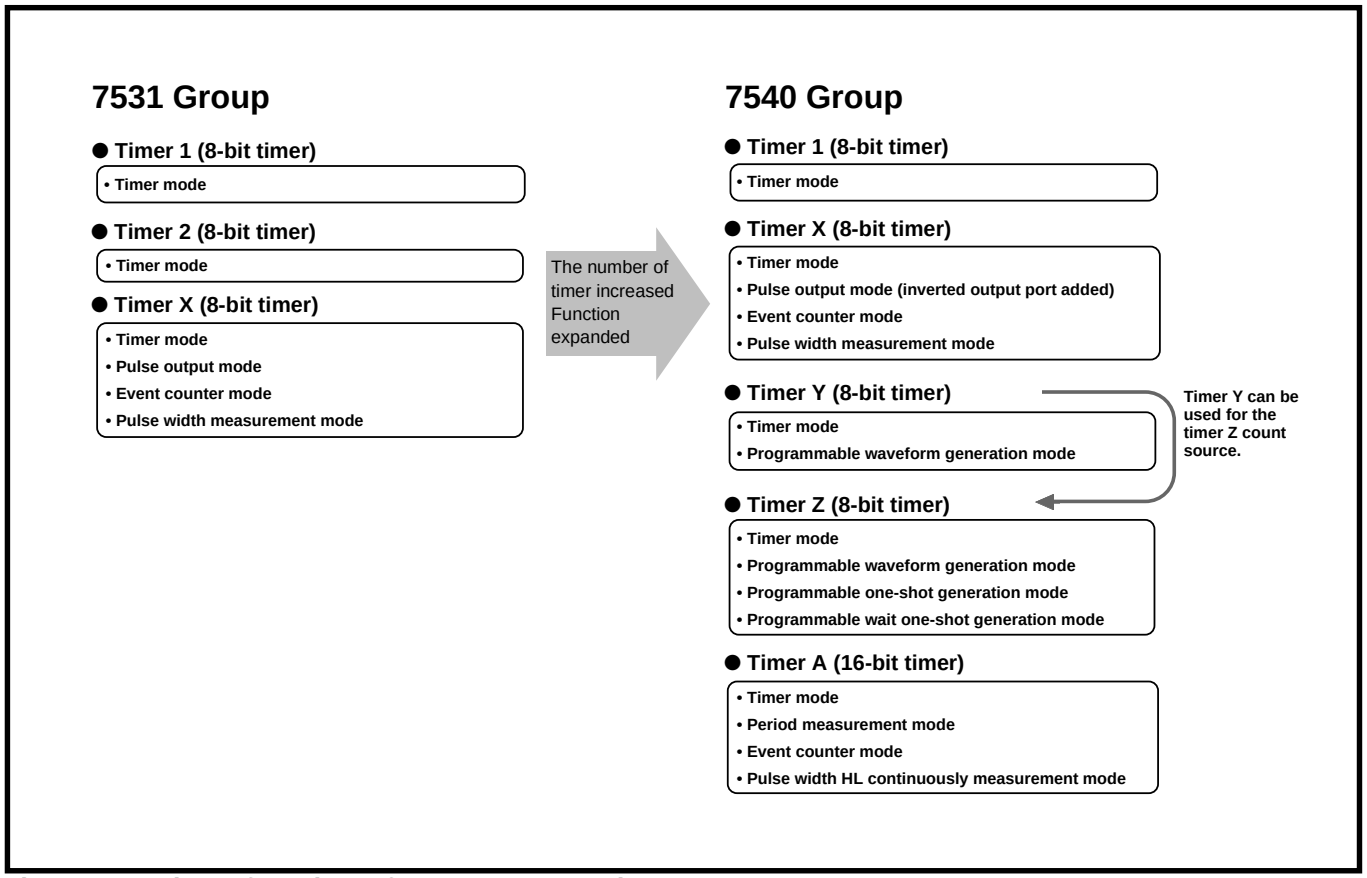
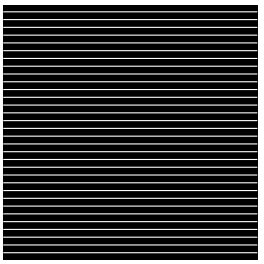


Fig. 3.11.3 Timer function of 7540 Group and 7531 Group

RENESAS 8-BIT CISC SINGLE-CHIP MICROCOMPUTER
USER'S MANUAL
7540 Group Rev.1.00

Editioned by
Committee of editing of RENESAS Semiconductor User's Manual

7540 Group
User's Manual



Renesas Technology Corp.
2-6-2, Ote-machi, Chiyoda-ku, Tokyo, 100-0004, Japan