To all our customers

Regarding the change of names mentioned in the document, such as Mitsubishi Electric and Mitsubishi XX, to Renesas Technology Corp.

The semiconductor operations of Hitachi and Mitsubishi Electric were transferred to Renesas Technology Corporation on April 1st 2003. These operations include microcomputer, logic, analog and discrete devices, and memory chips other than DRAMs (flash memory, SRAMs etc.) Accordingly, although Mitsubishi Electric, Mitsubishi Electric Corporation, Mitsubishi Semiconductors, and other Mitsubishi brand names are mentioned in the document, these names have in fact all been changed to Renesas Technology Corp. Thank you for your understanding. Except for our corporate trademark, logo and corporate statement, no changes whatsoever have been made to the contents of the document, and these changes do not constitute any alteration to the contents of the document itself.

Note : Mitsubishi Electric will continue the business operations of high frequency & optical devices and power devices.

Renesas Technology Corp.
Customer Support Dept.
April 1, 2003

MITSUBISHI 8-BIT SINGLE-CHIP MICROCOMPUTER
740 FAMILY / 7470 SERIES

7470/7471
7477/7478
Group

User's Manual

RENESAS
Renesas Technology Corp.

keep safety first in your circuit designs !

- Mitsubishi Electric 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 non-flammable 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 Mitsubishi semiconductor 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 Mitsubishi Electric Corporation or a third party.

- Mitsubishi Electric 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 or circuit application examples contained in these materials.

- All information contained in these materials, including product data, diagrams and charts, represent information on products at the time of publication of these materials, and are subject to change by Mitsubishi Electric Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor for the latest product information before purchasing a product listed herein.

- Mitsubishi Electric 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 Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor 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 Mitsubishi Electric 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 Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor for further details on these materials or the products contained therein.

Preface

This user's manual describes Mitsubishi's CMOS 8-bit microcomputers 7470/7471/7477/7478 group. After reading this manual, the user should have a through knowledge of the functions and features of the 7470/7471/7477/7478 group, and should be able to fully utilize the product. The manual starts with specifications and ends with application examples.
For detailes of software, refer to the "SERIES MELPS 740<SOFTWARE> USER'S MANUAL."
For details of development support tools, refer to the "DEVELOPMENT SUPPORT TOOLS FOR MICROCOMPUTERS" data book.

BEFORE USING THIS USER'S MANUAL

1. Organization

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.

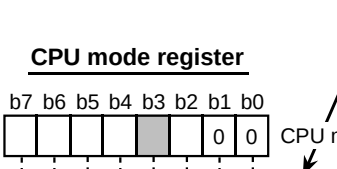
● **CHAPTER 1 HARDWARE**
This chapter describes features of the microcomputer and operation of each peripheral function.

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

● **CHAPTER 3 APPENDIX**
This chapter includes a list of registers, and necessary information for systems development using the microcomputer, the masking confirmation (mask ROM version), ROM programming confirmation, and mark specifications which are to be submitted when ordering.

2. Structure of Register

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



Bits

(Note 2)
Bit attributes

(Note 1)
Contents immediately after reset release

CPU mode register (CPUM) [Address: FB 16]

B	Name	Function	At reset	R	W
0, 1	Fix these bits to "0."		0	○	○
2	Stack page selection bit	0: In page 0 area 1: In page 1 area	0	○	○
3	Nothing is allocated for this bit. This is write enabled bit and is undefined at reading.		?	?	×
4	$P5_0$, $P5_1$/X_{CIN}, X_{COUT} selection bit	0: $P5_0$, $P5_1$ 1: X_{CIN}, X_{COUT}	0	○	○
5	X_{COUT} drive capacity selection bit	0: Low 1: High	0	○	○
6	Main clock (X_{IN}–X_{OUT}) stop bit	0: Oscillates 1: Stops	0	○	○
7	Internal system clock selection bit	0: X_{IN}–X_{OUT} selected (Ordinary mode) 1: X_{CIN}–X_{COUT} selected (low-speed mode)	0	○	○

: Bit in which nothing is arranged

Note 1. Contents immediately after reset release
0 ····· "0" at reset release
1 ····· "1" at reset release
? ····· Undefined at reset release

Note 2. Bit attributes
R ····· Read
○ ··· Read enabled
? ··· Undefined at reading
1 ··· "1" at reading
0 ··· "0" at reading

W ····· Write
○ ··· Write enabled
× ··· Write disabled
0 ··· Fix to "0".
✻ ··· Can be cleared to "0" by software but not set to "1".

Table of contents

CHAPTER 1. HARDWARE

CHAPTER 2. APPLICATION

CHAPTER 3. APPENDIX

List of tables

CHAPTER 1. HARDWARE

List of tables

CHAPTER 2. APPLICATION

List of figures

CHAPTER 1. HARDWARE

List of figures

List of figures

CHAPTER 2. APPLICATION

CHAPTER 3. APPENDIX

List of figures

CHAPTER 1

HARDWARE

1.1 Description

The 7470/7471/7477/7478 group is an 8-bit single-chip microcomputer which utilizes a silicon gate CMOS processing and has a simple instruction system of the 740 family using the same memory space for ROM, RAM and I/O.

1.2 Group expansion

The 7470/7471/7477/7478 group develops with the M37470M2-XXXSP as the base chip in the 7470 series. The classification of the 7470 series is as follows.

- 7470 series
  - 7470 group
  - 7471 group
  - 7477 group
  - 7478 group
  - 7480 group**
  - 7481 group**

**:Under development

In this manual, when multiple models are described collectively, their names are arranged by putting "/" among them for separation.
7470 group, 7471 group ⟶ 7470/7471 group
7477 group, 7478 group ⟶ 7477/7478 group
7470 group, 7477 group ⟶ 7470/7477 group
7471 group, 7478 group ⟶ 7471/7478 group

The 7470/7471/7477/7478 group permits group expansion as shown in Figure 1.2.1. This group expansion is all performed only by differences in memory type and capacity and the number of ports. This allows the user to select optimum elements according to the user's system.
The 7470/7471/7477/7478 group supports the following in addition to the mask ROM version.

(1) Support of One Time PROM version

The One Time PROM version is a programmable microcomputer and can perform a one-time write operation to the built-in programmable ROM (PROM).
For the details, refer to **"1.19 Built-in PROM version."**

(2) Support of EPROM version (with window)

The built-in EPROM version is a programmable microcomputer with window and can perform write and erase operations to the built-in EPROM.
For the details, refer to **"1.19 Built-in PROM version."**

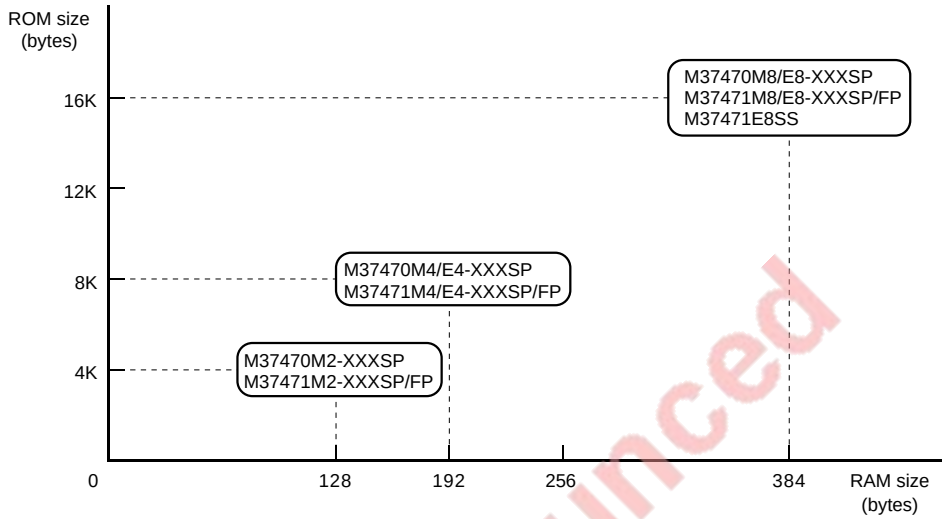
(3) Support of emulator MCU

The emulator MCU is a microcomputer designed for program development which facilitates program development and is an optimum element for system evaluation.
For the details, refer to**"1.20 Emulator MCU."**

Table 1.2.1 shows the products which the 7470/7471/7477/7478 group supports.

1.2 Group expansion

●Memory Expansion Plan of 7470/7471 group



●Memory Expansion Plan of 7477/7478 group

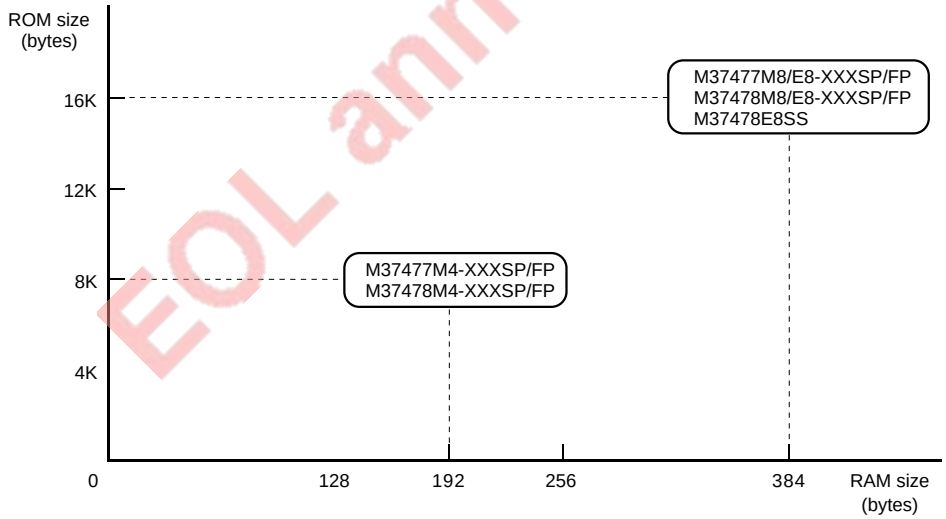


Fig. 1.2.1 Memory expansion plan of 7470/7471/7477/7478 group (As of Dec. 1997)

Table 1.2.1 List of supported products **(As of Dec. 1997)**

Product	ROM (bytes)	RAM (bytes)	I/O Port	Package	Remarks
M37470M2-XXXSP	4096	128	I/O ports: 22 (Including 4 analog input pins.) Input ports: 4	32P4B	Mask ROM version
M37470M4-XXXSP	8192	192			Mask ROM version
M37470E4-XXXSP					One Time PROM version
M37470M8-XXXSP	16384	384			Mask ROM version
M37470E8-XXXSP					One Time PROM version
M37471M2-XXXSP	4096	128	I/O ports: 28 (Including 8 analog input pins.) Input ports: 8	42P4B	Mask ROM version
M37471M2-XXXFP				56P6N-A	
M37471M4-XXXSP	8192	192		42P4B	
M37471M4-XXXFP				56P6N-A	
M37471E4-XXXSP				42P4B	One Time PROM version
M37471E4-XXXFP				56P6N-A	
M37471M8-XXXSP	16384	384		42P4B	Mask ROM version
M37471M8-XXXFP				56P6N-A	
M37471E8-XXXSP				42P4B	One Time PROM version
M37471E8-XXXFP				56P6N-A	
M37471E8SS				42S1B-A	EPROM version
M37471RSS	63.5K (Note)	384		42S1M	Emulator MCU
M37477M4-XXXSP	8192	192	I/O ports: 18 (Including 4 analog input pins.) Input ports: 8	32P4B	Mask ROM version
M37477M4-XXXFP				32P2W-A	
M37477M8-XXXSP	16384	384		32P4B	Mask ROM version
M37477M8-XXXFP				32P2W-A	
M37477E8-XXXSP				32P4B	One Time PROM version
M37477E8-XXXFP				32P2W-A	
M37478M4-XXXSP	8192	192	I/O ports: 20 (Including 8 analog input pins.) Input ports: 16	42P4B	Mask ROM version
M37478M4-XXXFP				56P6N-A	
M37478M8-XXXSP	16384	384		42P4B	Mask ROM version
M37478M8-XXXFP				56P6N-A	
M37478E8-XXXSP				42P4B	One Time PROM version
M37478E8-XXXFP				56P6N-A	
M37478E8SS	16384	384	I/O ports: 20 Input ports: 16 (Including 8 analog input pins.)	42S1B-A	EPROM version
M37478RSS	63.5K (Note)	384		42S1M	Emulator MCU

Note: Address space usable as a ROM area.

1.3 Performance overview

Tables 1.3.1 to 1.3.4 show the performance overview of 7470/7471/7477/7478 group.

Table 1.3.1 Performance overview of 7470 group

Parameter			Functions
Number of basic instructions			71 (69 basic instructions of 740 family and 2 multiplication and division instructions)
Instruction execution time			0.5 μs (the minimum instructions, at 8 MHz clock input oscillation frequency)
Clock input oscillation frequency			8 MHz (max.)
Memory size	ROM	M37470M2	4096 bytes
		M37470M4/E4	8192 bytes
		M37470M8/E8	16384 bytes
	RAM	M37470M2	128 bytes
		M37470M4/E4	192 bytes
		M37470M8/E8	384 bytes
Input/ Output port	I/O	P0	8-bit
		P1	8-bit
		P2	4-bit
		P4	2-bit
	Input	P3	4-bit
Serial I/O			8-bit × 1
Timers			8-bit timer × 4
PWM			1 (in common with 2 timer)
A-D converter			8-bit × 1 (4 channels)
Subroutine nesting		M37470M2	64 levels max.
		M37470M4/E4	96 levels max.
		M37470M8/E8	192 levels max.
Interrupt			5 external interrupts, 6 internal interrupts, 1 software interrupt
Clock generating circuit			Built-in circuit with internal feedback resistor (an external ceramic resonator or a quartz-crystal oscillator)
Power source voltage			2.7 V to 4.5 V (at (2.2 Vcc–2) MHz clock input oscillation frequency)
			4.5 V to 5.5 V (at 8 MHz clock input oscillation frequency)
Power dissipation			35 mW typ. (at 8 MHz clock input oscillation frequency)
Input/Output characteristics		Input/Output withstand voltage	5 V
		Output current	–5 mA to +10 mA (P0, P1, P2, P4: CMOS 3-state)
Operating temperature			–20 °C to +85 °C
Device structure			CMOS silicon gate
Package		M37470Mx/Ex-XXXSP	32-pin shrink plastic molded DIP

Table 1.3.2 Performance overview of 7471 group

Parameter			Functions
Number of basic instructions			71 (69 basic instructions of 740 family and 2 multiplication and division instructions)
Instruction execution time			0.5 μs (the minimum instructions, at 8 MHz clock input oscillation frequency)
Clock input oscillation frequency			8 MHz (max.)
Memory size	ROM	M37471M2	4096 bytes
		M37471M4/E4	8192 bytes
		M37471M8/E8	16384 bytes
	RAM	M37471M2	128 bytes
		M37471M4/E4	192 bytes
		M37471M8/E8	384 bytes
Input/ Output port	I/O	P0	8-bit
		P1	8-bit
		P2	8-bit
		P4	4-bit
	Input	P3	4-bit
		P5	4-bit
Serial I/O			8-bit × 1
Timers			8-bit timer × 4
PWM			1 (in common with 2 timer)
A-D converter			8-bit × 1 (8 channels)
Subroutine nesting		M37471M2	64 levels max.
		M37471M4/E4	96 levels max.
		M37471M8/E8	192 levels max.
Interrupt			5 external interrupts, 6 internal interrupts, 1 software interrupt
Clock generating circuit			Built-in circuit with internal feedback resistor (an external ceramic resonator or a quartz-crystal oscillator)
Sub-clock generating circuit			Built-in circuit with internal feedback resistor (a guartz-crystal oscillator)
Power source voltage			2.7 V to 4.5 V (at (2.2 VCC–2) MHz clock input oscillation frequency)
			4.5 V to 5.5 V (at 8 MHz clock input oscillation frequency)
Power dissipation			35 mW typ. (at 8 MHz clock input oscillation frequency)
Input/Output characteristics		Input/Output withstand voltage	5 V
		Output current	–5 mA to +10 mA (P0, P1, P2, P4: CMOS 3-state)
Operating temperature			–20 °C to +85 °C
Device structure			CMOS silicon gate
Package		M37471Mx/Ex-XXXSP	42-pin shrink plastic molded DIP
		M37471Mx/Ex-XXXFP	56-pin plastic molded QFP
		M37471E8SS	42-pin shrink ceramic DIP

1.3 Performance overview

Table 1.3.3 Performance overview of 7477 group

Parameter			Functions
Number of basic instructions			71 (69 basic instructions of 740 family and 2 multiplication and division instructions)
Instruction execution time			0.5 μs (the minimum instructions, at 8 MHz clock input oscillation frequency)
Clock input oscillation frequency			8 MHz (max.)
Memory size	ROM	M37477M4	8192 bytes
		M37477M8/E8	16384 bytes
	RAM	M37477M4	192 bytes
		M37477M8/E8	384 bytes
Input/ Output port	I/O	P0	8-bit
		P1	8-bit
		P4	2-bit
	Input	P2	4-bit
		P3	4-bit
Serial I/O			8-bit × 1 (operable in UART mode)
Timers			8-bit timer × 4
PWM			1 (in common with 2 timer)
A-D converter			8-bit × 1 (4 channels)
Subroutine nesting		M37477M4	96 level max.
		M37477M8/E8	192 level max.
Interrupt			5 external interrupts, 7 internal interrupts, 1 software interrupt
Clock generating circuit			Built-in circuit with internal feedback resistor (an external ceramic resonator or a quartz-crystal oscillator)
Power source voltage			2.7 V to 4.5 V (at (2.2 Vcc–2) MHz clock input oscillation frequency)
			4.5 V to 5.5 V (at 8 MHz clock input oscillation frequency)
Power dissipation			35 mW typ. (at 8 MHz clock input oscillation frequency)
Input/Output characteristics		Input/Output withstand voltage	5 V
		Output current	–5 mA to +10 mA (P0, P1, P4: CMOS 3-state)
Operating temperature			–20 °C to +85 °C
Device structure			CMOS silicon gate
Package		M37477Mx/E8-XXXSP	32-pin shrink plastic molded DIP
		M37477Mx/E8-XXXFP	32-pin plastic molded SOP

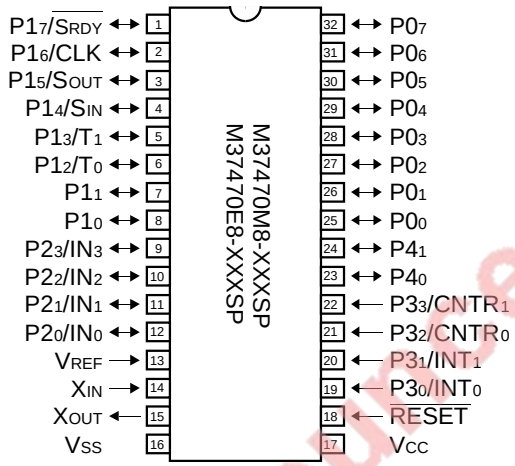
Table 1.3.4 Performance overview of 7478 group

Parameter			Functions
Number of basic instructions			71 (69 basic instructions of 740 family and 2 multiplication and division instructions)
Instruction execution time			0.5 μs (the minimum instructions, at 8 MHz clock input oscillation frequency)
Clock input oscillation frequency			8 MHz (max.)
Memory size	ROM	M37478M4	8192 bytes
		M37478M8/E8	16384 bytes
	RAM	M37478M4	192 bytes
		M37478M8/E8	384 bytes
Input/ Output port	I/O	P0	8-bit
		P1	8-bit
		P4	4-bit
	Input	P2	8-bit
		P3	4-bit
		P5	4-bit
Serial I/O			8-bit × 1 (operable in UART mode)
Timers			8-bit timer × 4
PWM			1 (in common with 2 timer)
A-D converter			8-bit × 1 (8 channels)
Subroutine nesting		M37478M4	96 level max.
		M37478M8/E8	192 level max.
Interrupt			5 external interrupts, 7 internal interrupts, 1 software interrupt
Clock generating circuit			Built-in circuit with internal feedback resistor (an external ceramic resonator or a quartz-crystal oscillator)
Sub-clock generating circuit			Built-in circuit with internal feedback resistor (a quartz-crystal oscillator)
Power source voltage			2.7 V to 4.5 V (at (2.2 VCC–2) MHz clock input oscillation frequency)
			4.5 V to 5.5 V (at 8 MHz clock input oscillation frequency)
Power dissipation			35 mW typ. (at 8 MHz clock input oscillation frequency)
Input/Output characteristics		Input/Output withstand voltage	5 V
		Output current	–5 mA to +10 mA (P0, P1, P4: CMOS 3-state)
Operating temperature			–20 °C to +85 °C
Device structure			CMOS silicon gate
Package		M37478Mx/E8-XXXSP	42-pin shrink plastic molded DIP
		M37478Mx/E8-XXXFP	56-pin plastic molded QFP
		M37478E8SS	42-pin shrink ceramic DIP

1.4 Pin configuration

Figures 1.4.1 to 1.4.4 show a pin configuration of "7470/7471/7477/7478 group."
For pin connections in the EPROM mode of the built-in programmable ROM version, refer to **"Figures 1.19.1 to 1.19.6 Pin connections in EPROM mode."**

PIN CONFIGURATION (TOP VIEW)

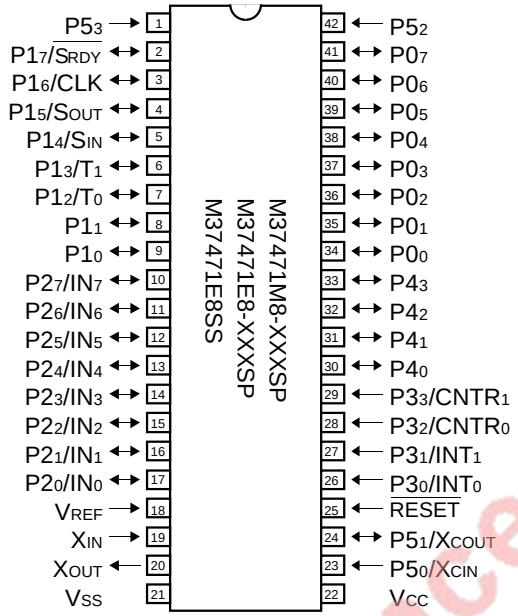


Outline 32P4B (Note)

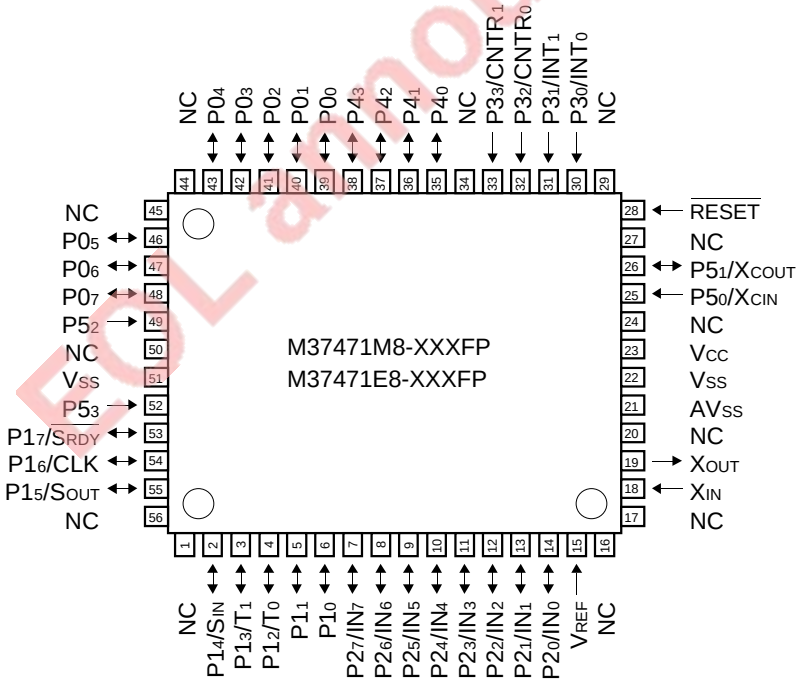
Note: The M37470M2-XXXSP and M37470M4/E4-XXXSP are included in the 32P4B package. All of these products are pin-compatible.

Fig. 1.4.1 Pin configuration of 7470 group

PIN CONFIGURATION (TOP VIEW)



Outline 42P4B (Note 1)
42S1B-A (M37471E8SS)



Outline 56P6N-A (Note 2)

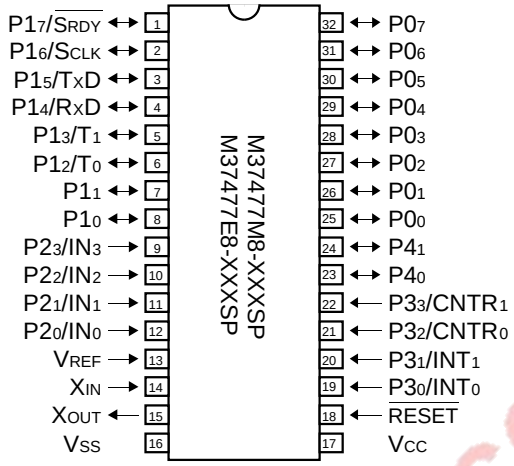
NC: No connection

Notes 1 : The M37471M2-XXXSP and M37471M4/E4-XXXSP are included in the 42P4B package. All of these products are pin-compatible.
2 : The M37471M2-XXXFP and M37471M4/E4-XXXFP are included in the 56P6N-A package. All of these products are pin-compatible.
3 : The only differences between the 42P4B package product and the 56P6N-A package product are package shape, absolute maximum ratings and the fact that the 56P6N-A package product has an AVSS pin.

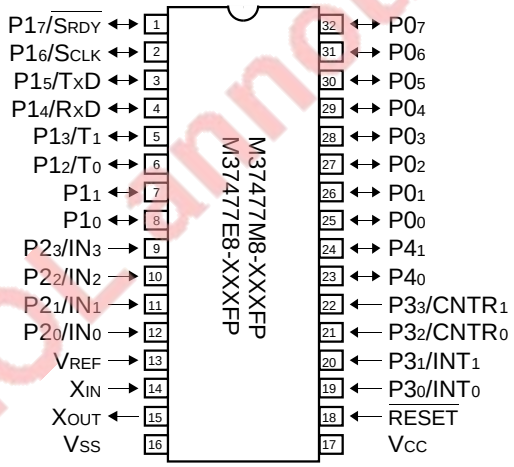
Fig. 1.4.2 Pin configuration of 7471 group

1.4 Pin configuration

PIN CONFIGURATION (TOP VIEW)



Outline 32P4B (Note 1)



Outline 32P2W-A (Note 2)

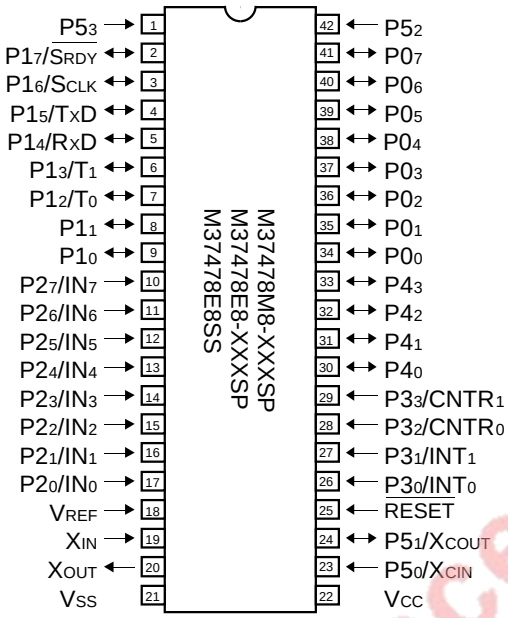
Notes 1 : The M37477M4-XXXSP is included in the 32P4B package.
These products are pin-compatible.

2 : The M37477M4-XXXFP is included in the 32P2W-A package.
These products are pin-compatible.

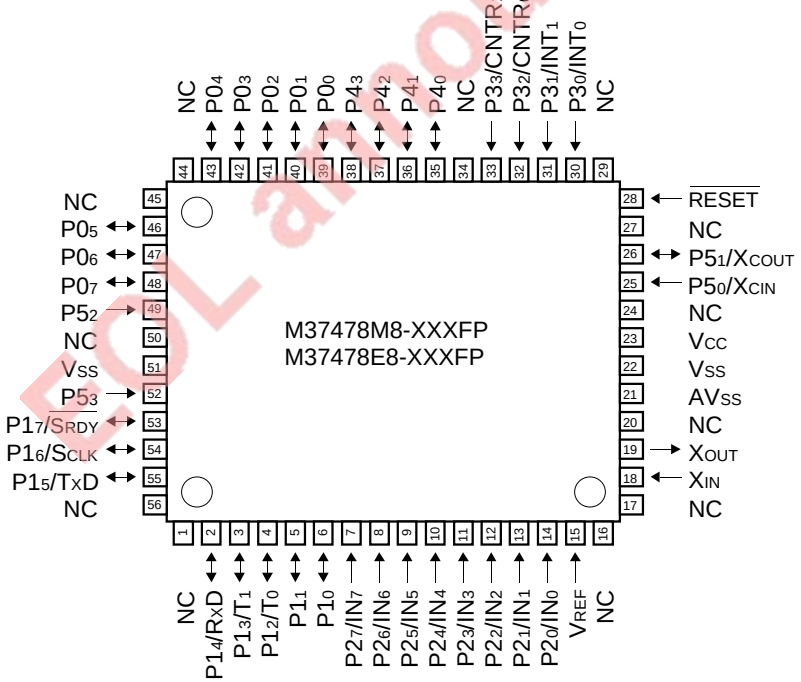
3 : The only differences between the 32P4B package product and the 32P2W-A package product are package shape and absolute maximum ratings.

Fig. 1.4.3 Pin configuration of 7477 group

PIN CONFIGURATION (TOP VIEW)



Outline 42P4B (Note 1)
42S1B-A (M37478E8SS)



Outline 56P6N-A (Note 2)

NC: No connection

Notes 1 : The M37478M4-XXXSP is included in the 42P4B package.
These products are pin-compatible
2 : The M37478M4-XXXFP is included in the 56P6N-A package.
These products are pin-compatible
3 : The only differences between the 42P4B package product and the 56P6N-A package product are package shape, absolute maximum ratings and the fact that the 56P6N-A package product has an AV_{SS} pin.

Fig. 1.4.4 Pin configuration of 7478 group

1.5 Pin description

Tables 1.5.1 to 1.5.3 show a pin description.
For pin functions in the EPROM mode of the built-in programmable ROM version, refer to **"1.19.2 Pin description."**

Table 1.5.1 Pin description (1)

Pin	Name	Input/ Output	Functions
V_{CC}, V_{SS}	Power source		• Apply the following voltage to the V_{CC} pin: 2.7 V to 4.5 V (at f(X_{IN}) = (2.2 V_{CC}–2) MHz clock input oscillation frequency) or 4.5 V to 5.5 V (at f(X_{IN}) = 8 MHz clock input oscillation frequency). • Apply 0 V to the V_{SS} pin.
AV_{SS}	Analog power source		• Ground level input pin for the A-D converter. • Apply the same voltage as V_{SS} pin to the AV_{SS} pin. **Note:** This pin is dedicated to 56P6N-A package products among the 7471/7478 group.
V_{REF}	Reference voltage input	Input	• Reference voltage input pin for the A-D converter. • When using the A-D converter, apply 0.5 V_{CC} (≧ 2) to V_{CC} [V]. • When not using the A-D converter, connect to V_{CC}.
$\overline{RESET}$	Reset input	Input	• Reset input pin • The microcomputer is put into a reset state by keeping the $\overline{RESET}$ pin at "L" for 2 μs or more, and the reset state is released by returning the $\overline{RESET}$ pin to "H."
X_{IN}	Clock input	Input	• An input pin and an output pin for the main clock generating circuit. • Connect a ceramic resonator or a quartz-crystal oscillator between pins X_{IN} and X_{OUT}. • A feedback resistor is incorporated between the X_{IN} and the X_{OUT} pins. • To use an external clock input, connect the clock oscillation source to the X_{IN} pin and leave the X_{OUT} pin open.
X_{OUT}	Clock output	Output	
$P0_0$–$P0_7$	I/O port P0	I/O	• Port P0 is an 8-bit I/O port. • The output structure is CMOS output. • In input mode, a pull-up transistor is connectable in units of one bit. • In input mode, a key-on wake up function is provided.

Table 1.5.2 Pin description (2)

Pin	Name	Input/ Output	Functions
$P1_0$–$P1_7$	I/O port P1	I/O	• Port P1 is an 8-bit I/O port. • The output structure is CMOS output. • In input mode, pull-up transistor can be connected in units of 4-bit. • Pins $P1_2$ and $P1_3$ are in common with timer output pins T_0, T_1 respectively. • In the case of the 7470/7471 group, $P1_4$–$P1_7$ are in common with serial I/O pins SIN, SOUT, CLK, $\overline{SRDY}$ respectirely. • In the case of the 7470/7471 group, the outputs of pins SOUT and the $\overline{SRDY}$ can be N-channel open drain outputs. • In the case of the 7477/7478 group, $P1_4$–$P1_7$ are in common with serial I/O pins RxD, TxD, SCLK, $\overline{SRDY}$, respectively.
$P2_0$–$P2_7$	I/O port P2 (7470/7471 group)	I/O	• Port P2 is an 8-bit I/O port. • The output structure is CMOS output. • In input mode, pull-up transistor can be connected in units of 4-bit. • Pins $P2_0$–$P2_7$ are in common with analog input pins IN_0–IN_7 respectively. **Note:** The 7470 group has only the 4 pins $P2_0$–$P2_3$ (IN_0–IN_3).
	Input port P2 (7477/7478 group)	Input	• Port P2 is an 8-bit input port. • It is impossible to connect a pull-up transistor. • Pins $P2_0$–$P2_7$ are in common with analog Input pins IN_0–IN_7 respectively. **Note:** The 7477 group has only the 4 pins $P2_0$–$P2_3$ (IN_0–IN_3).
$P3_0$–$P3_3$	Input port P3	Input	• Port P3 is a 4-bit input port. • Pins $P3_0$, $P3_1$ are in common with external interrupt input pins INT_0, INT_1 respectively. • Pins $P3_2$, $P3_3$ are in common with timer input pins $CNTR_0$, $CNTR_1$ respectively.
$P4_0$–$P4_3$	I/O port P4	I/O	• Port P4 is a 4-bit I/O port. • The output structure is CMOS output. • In input mode, pull-up transistor can be connected in units of 4-bit. **Note:** The 7470/7477 group has only 2 pins $P4_0$ and $P4_1$.

1.5 Pin description

Table 1.5.3 Pin description (3)

Pin	Name	Input/ Output	Functions
$P5_0$–$P5_3$	Input port P5	Input	• Port P5 is a 4-bit input port. • Pull-up transistor can be connected in units of 4-bit. • Pins $P5_0$, $P5_1$ are in common with input/output pins for sub-clock generating circuit XCIN, XCOUT respectively. • When using pins $P5_0$ and $P5_1$ as pins XCIN and XCOUT, connect a quartz-crystal oscillator between pins XCIN and XCOUT. • When using pins $P5_0$ and $P5_1$ as pins XCIN and XCOUT, a feedback resistor is connected between pins XCIN and XCOUT. • To use an external clock input, connect the clock oscillation source to the XCIN pin and leave the XCOUT pin open. **Note:** Only the 7471/7478 group has pins $P5_0$–$P5_3$.

1.6 Functional block diagram

The functional block diagram of 7470/7471/7477/7478 group is shown in Figure 1.6.1 to Figure 1.6.6.

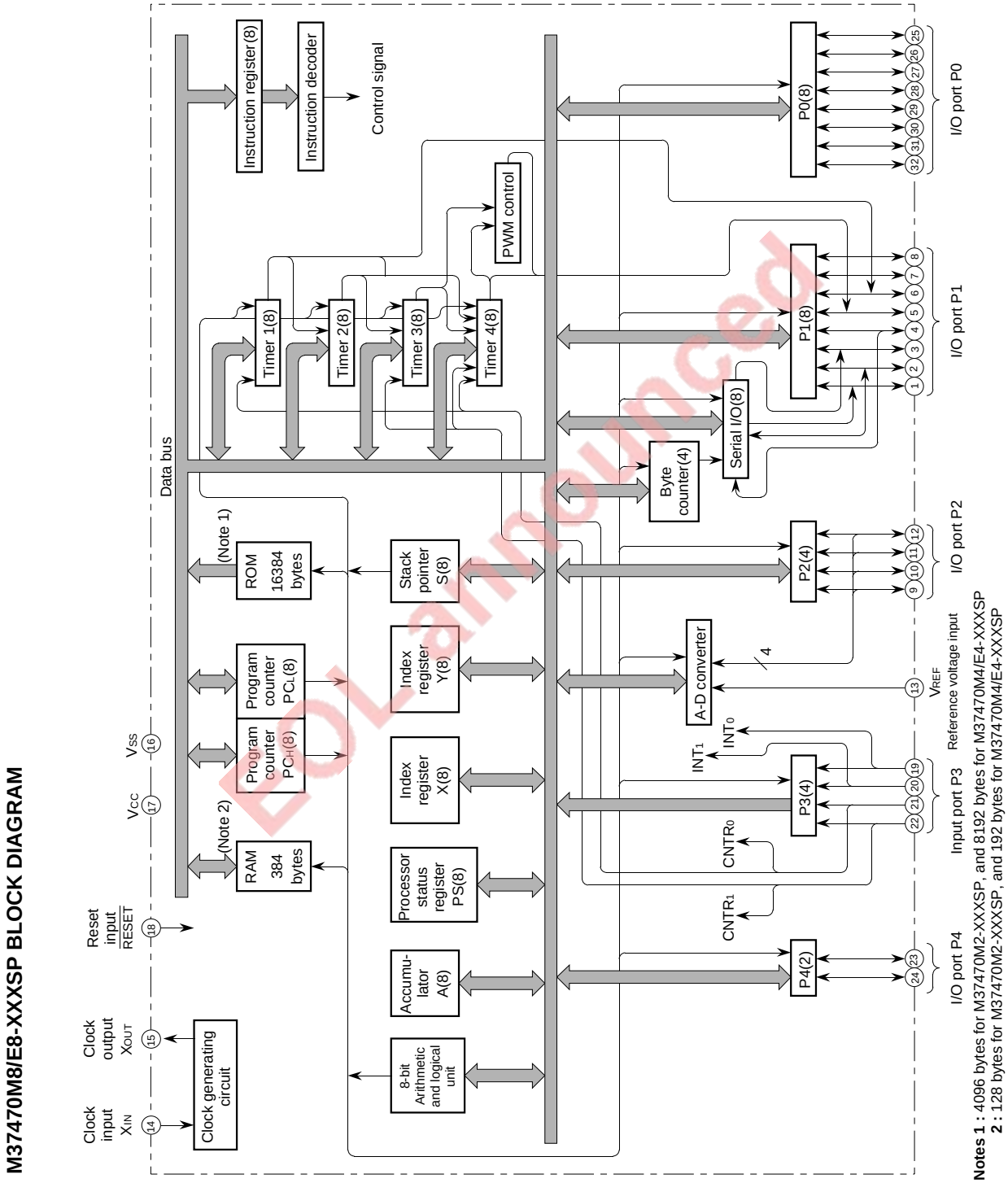


Fig. 1.6.1 M37470Mx/Ex-XXXSP functional block diagram

1.6 Functional block diagram

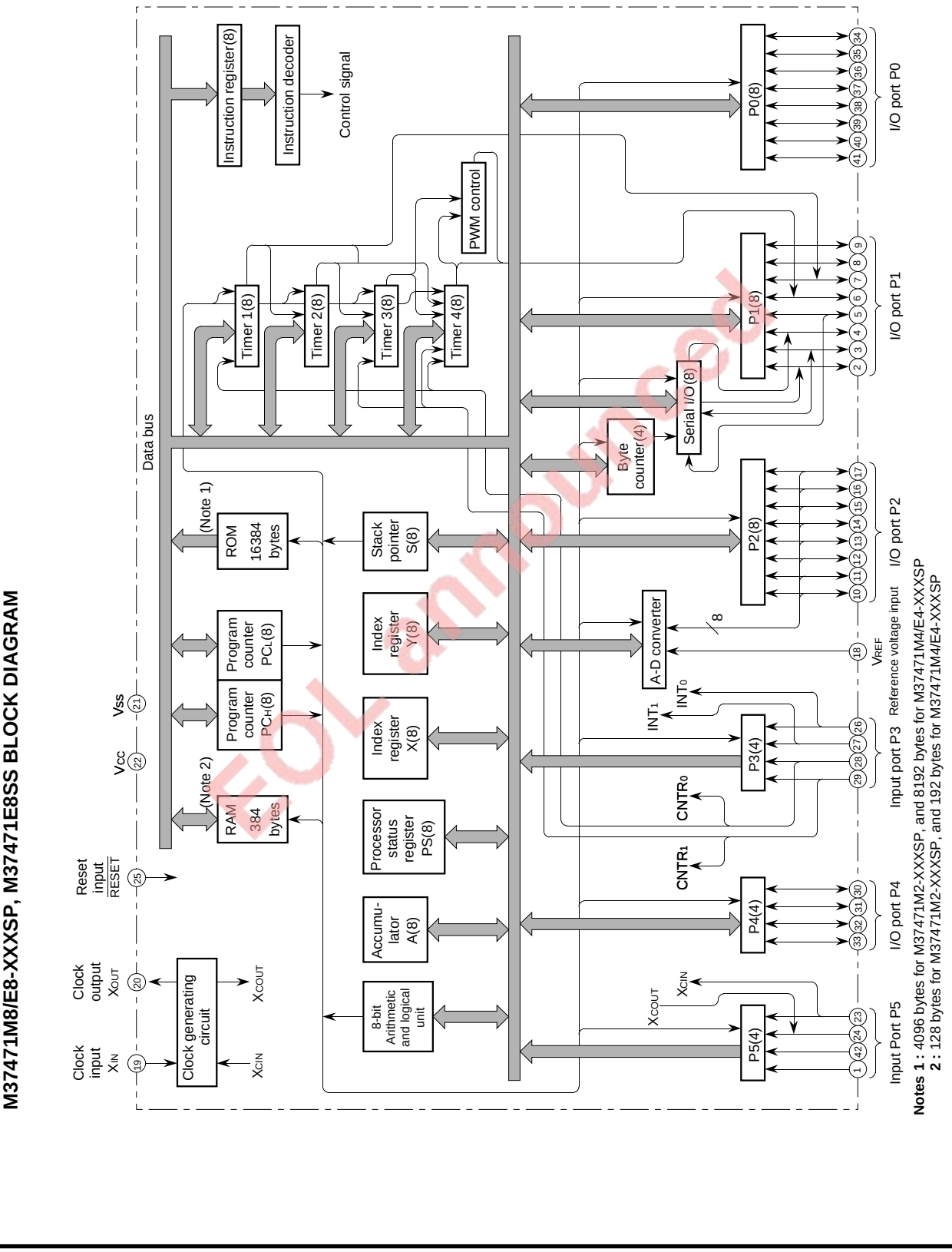


Fig. 1.6.2 M37471Mx/Ex-XXXSP, M37471E8SS functional block diagram

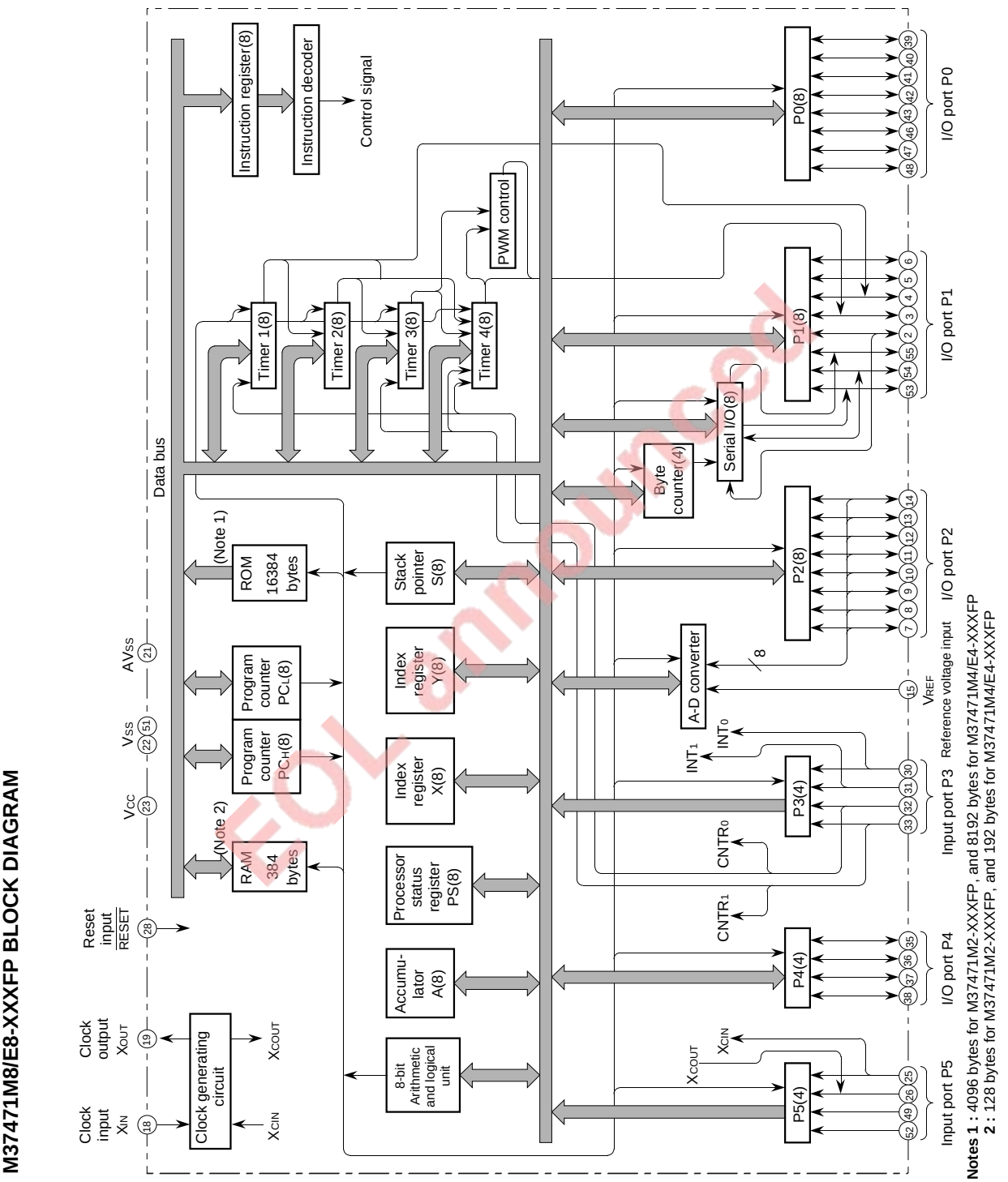


Fig. 1.6.3 M37471Mx/Ex-XXXFP functional block diagram

1.6 Functional block diagram

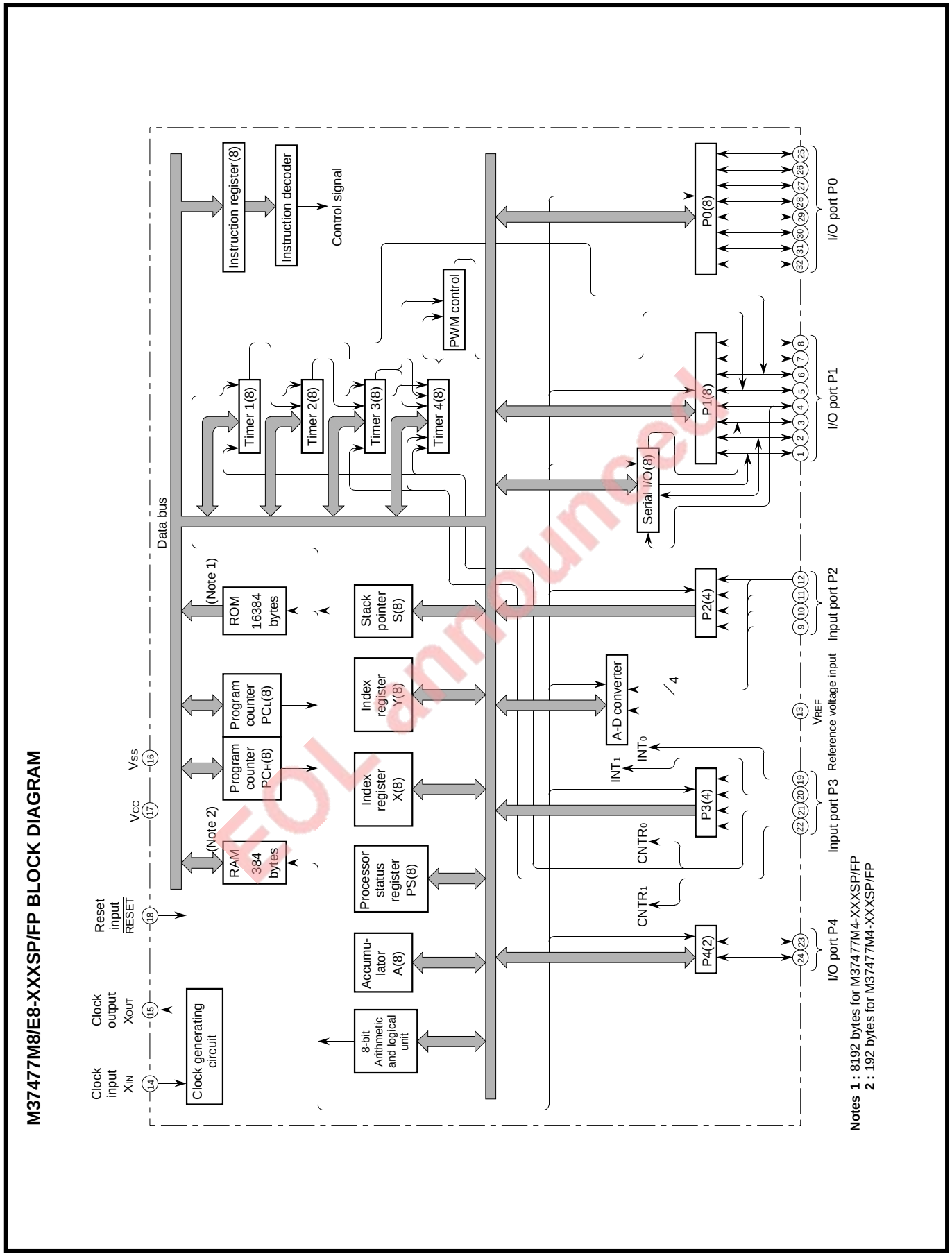


Fig. 1.6.4 M37477Mx/E8-XXXSP/FP functional block diagram

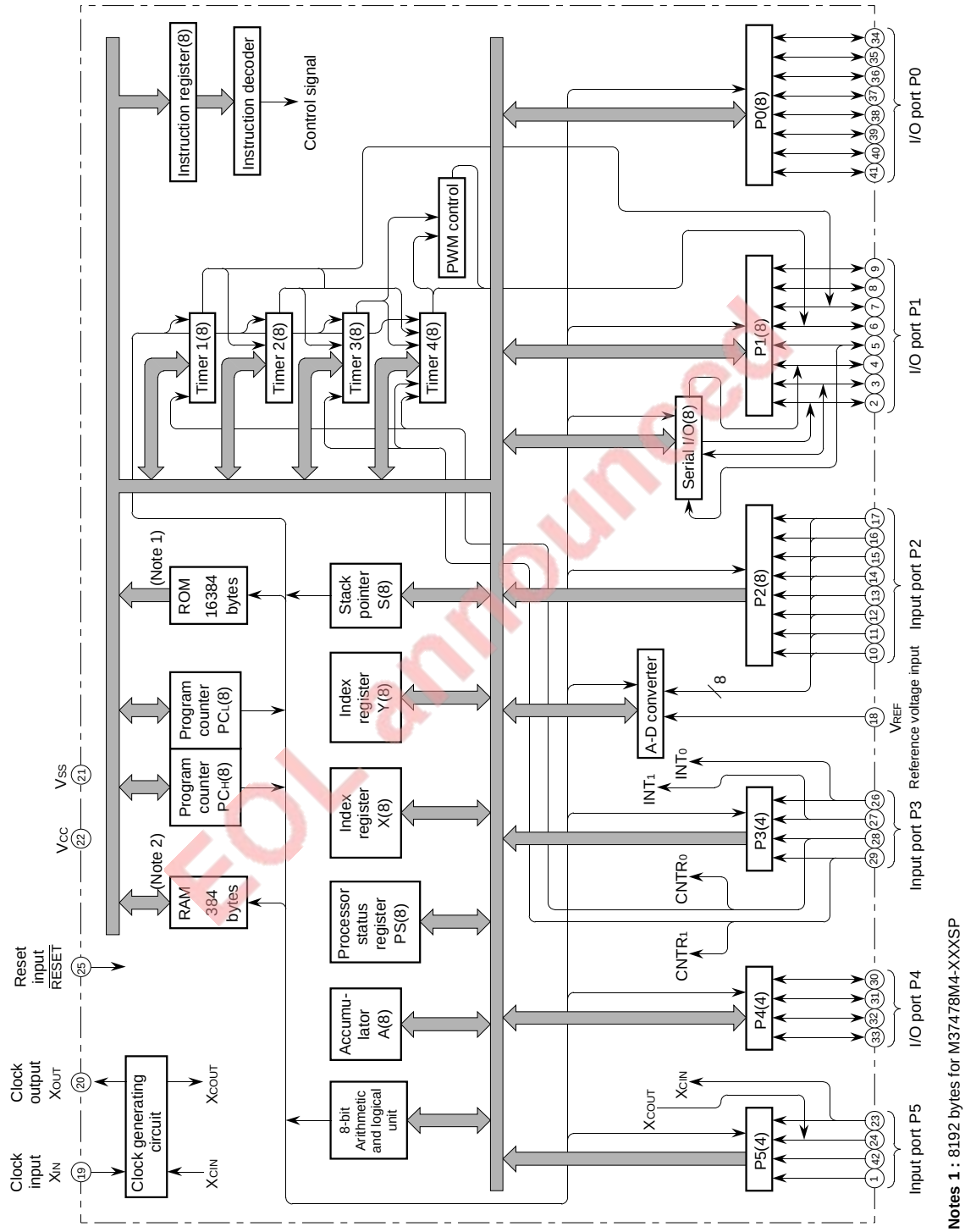


Fig. 1.6.5 M37478Mx/E8-XXXSP, M37478E8SS functional block diagram

1.6 Functional block diagram

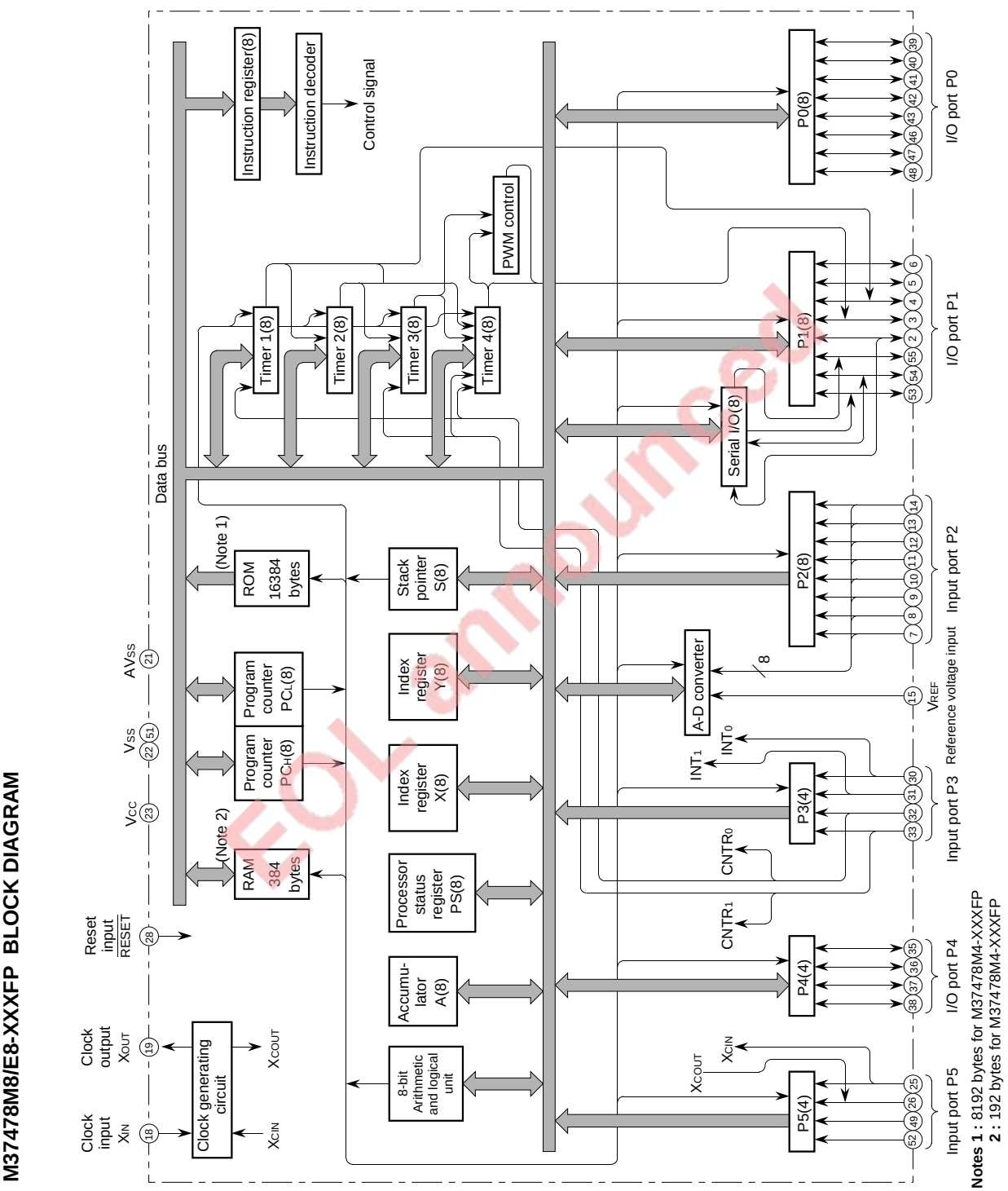


Fig. 1.6.6 M37478Mx/E8-XXXFP functional block diagram

1.7 Central processing unit (CPU)

The CPU of 7470/7471/7477/7478 group has the following 6 registers (referred as “CPU registers”).

- Accumulator (A) 8-bit
- Index register X (X) 8-bit
- Index register Y (Y) 8-bit
- Stack pointer (S) 8-bit
- Processor status register (PS) 8-bit
- Program counter (PC) 16-bit
  - high-order (PC_H) 8-bit
  - low-order (PC_L) 8-bit

Figure 1.7.1 shows a structure of CPU registers.

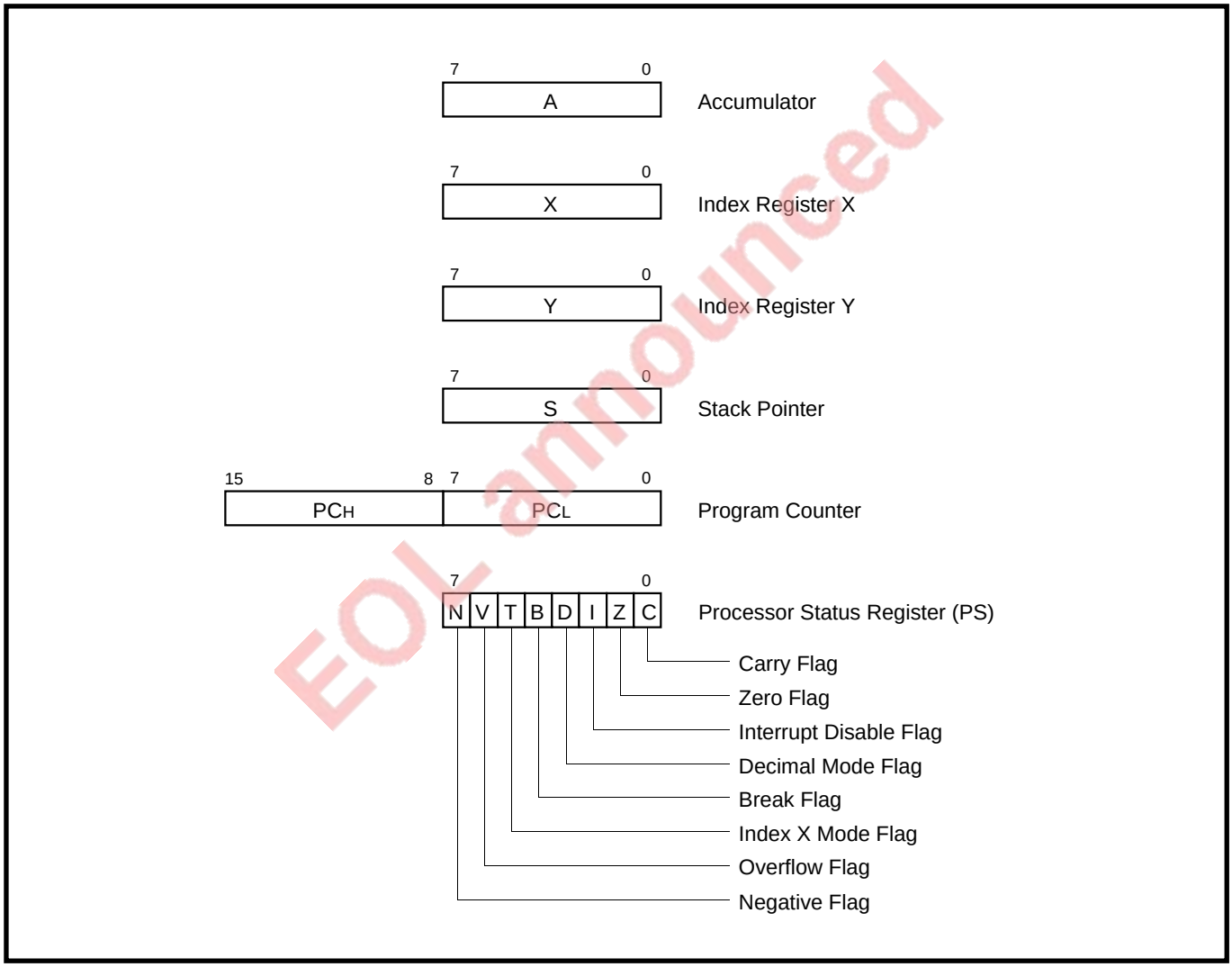


Fig. 1.7.1 Structure of CPU registers

1.7 Central processing unit (CPU)

The CPU register states provided immediately after hardware reset are described below.

- The interrupt disable flag (I) of the Processor status register (PS) is set to "1."
- The high-order 8 bits (PC_H) of the Program counter (PC) become the contents of address $FFFF_{16}$ and the low-order 8 bits (PC_L) become the contents of address $FFFE_{16}$.

  The contents of the other CPU registers are undefined, so be sure to initialize the CPU registers with the program.

1.7.1 Accumulator (A)

The Accumulator is the central of microcomputer and is an 8-bit register. This accumulator is used for arithmetic operations, data transfer, temporary storage, condition judgment, and is a general-purpose register with the highest frequency of use.

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

The Index register X and the Index register Y are 8-bit registers.
In the addressing mode using these Index registers, a value resulting from adding the contents of this register to the operand becomes a real specified address. This addressing mode is used to make reference to a subroutine table or a memory table. The Index registers are provided with increment, decrement, comparison and data transfer functions and can also be used as a simplified accumulator.
In the Index register X, when the index X mode flag (T) of the Processor status register is "1," the contents of the Index register become an operand address.

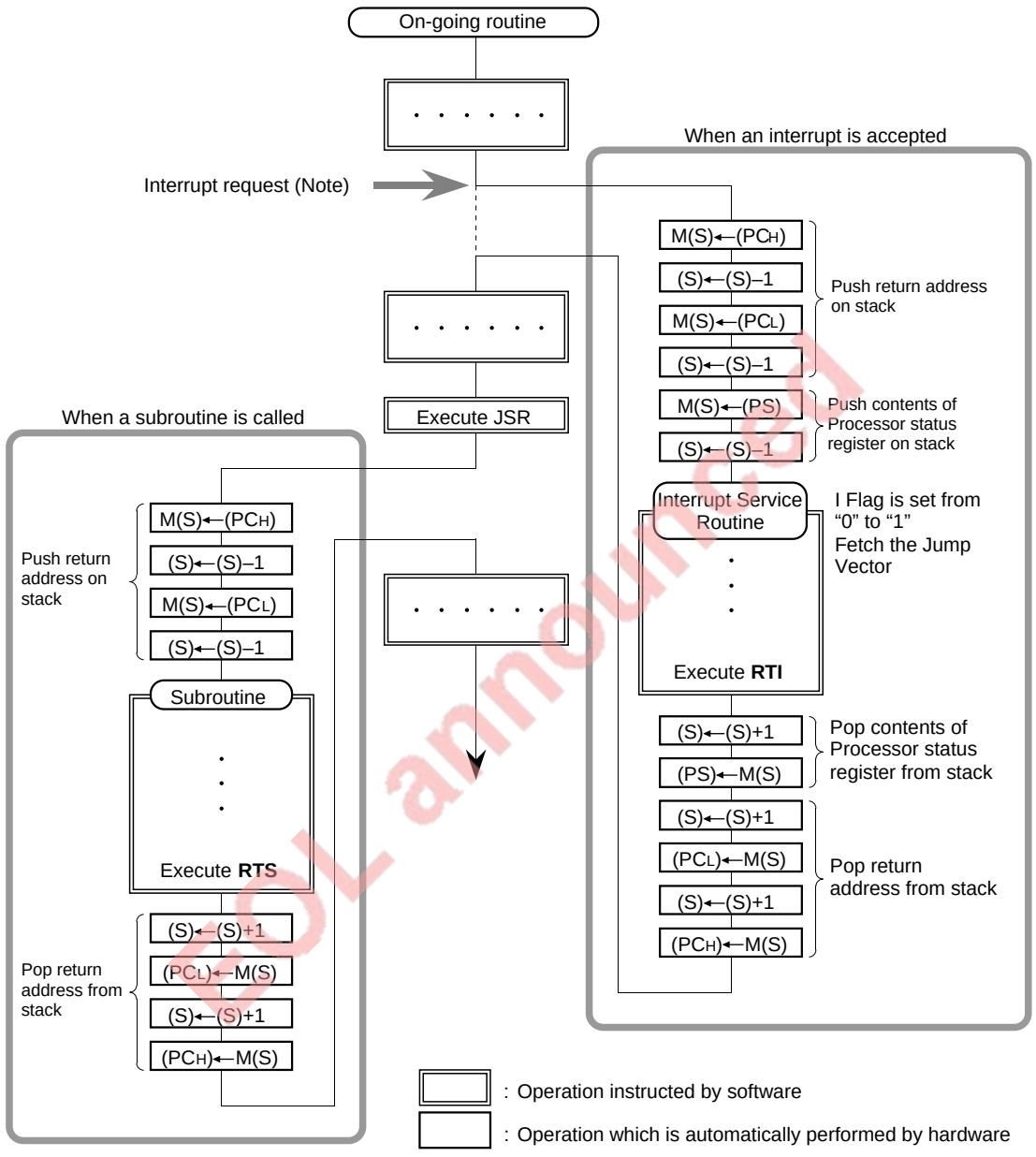
1.7.3 Stack pointer (S)

The Stack pointer is an 8-bit register which is used to call a subroutine or generate an interrupt.
For a branch from a routine being executed to a subroutine or an interrupt processing routine, it is necessary to temporarily store (push) in memory the return address at the termination of this processing. Usually, the internal RAM is used as the push destination, and this area is called a stack area. The stack pointer indicates an address in the stack area to which the data will be pushed next.

Figure 1.7.2 shows a push operation to the stack area of the register and a pop operation from the Stack area of the register.

The Program counter and registers other than the Processor status register are not automatically pushed. Accordingly, be sure to push necessary registers with the program.
The PHA instruction and the PLA instruction are used for push and pop operations of the Accumulator and the PHP instruction and the PLP instruction are used for push and pop operations of the Processor status register.
In the 7470/7471/7477/7478 group, the RAM in 0 page or 1 page is available as a stack area. Select it by the stack page bit (bit 2) of the CPU mode register (address $00FB_{16}$), which will be described later ("0" for 0 page or "1" for 1 page). In some products whose RAM capacity is 192 bytes or less, RAM does not exist on 1 page, so be sure to set this bit to "0."

The stack pointer is in an undefined state immediately after hardware reset. Be sure to initialize so as not to destroy the data arranged in the RAM area.



Note : Condition for acceptance of an interrupt · · · { Interrupt disable flag is "0" (enable state)
Interrupt enable bit is "1" (enable state)

Fig. 1.7.2 Register push and pop at interrupt generation and subroutine call

1.7 Central processing unit (CPU)

1.7.4 Program counter (PC)

The Program counter is a 16-bit counter consisting of an 8-bit register PC_H and an 8-bit register PC_L. This counter indicates the address at which the next instruction to be executed is stored.
The contents of this counter are automatically pushed to the stack when a subroutine is called or an interrupt occurs.
The high-order 8 bits (PC_H) of the program counter become the contents of address $FFFF_{16}$ and the low-order 8 bits (PC_L) become the contents of address $FFFE_{16}$ immediately after hardware reset.

1.7.5 Processor status register (PS)

The Processor status register is an 8-bit register consisting of 5 flags to indicate the state immediately after arithmetic processing and 3 flags to determine an operation for the CPU.
Each bit of the Processor status register is described below.

(1) Carry flag (C) ..... Bit 0

The carry flag holds the carry or borrow from the arithmetic logical unit after arithmetic processing. This flag is also changed by the Shift instruction or Rotate instruction.
This flag is set to "1" by the SEC instruction and cleared to "0" by the CLC instruction.

(2) Zero flag (Z) ..... Bit 1

The zero flag is set to "1" when the arithmetic processing or data transfer result is "0" and cleared to "0" in all other cases. In the decimal operation mode, this flag is invalidated.
There is no instruction to change the contents of this flag.

(3) Interrupt disable flag (I) ..... Bit 2

The interrupt request flag disables all instructions (except an interrupt by the BRK instruction). When this flag is "1," the interrupt disable state is provided. This flag is set to "1" by accepting an interrupt, thereby disabling a multi-interrupt.
This flag is set to "1" by the SEI instruction and cleared to "0" by the CLI instruction.
This flag is set to "1" (interrupt disable state) immediately after hardware reset.

(4) Decimal mode flag (D) ..... Bit 3

The decimal mode flag determines whether addition and subtraction should be performed in binary or decimal notation. When the contents of this flag are "0," an ordinary binary operation is performed. When they are "1," an arithmetic operation is performed assuming that one word is a 2-digit decimal number. In a decimal operation, decimal compensation is automatically performed (decimal operation can be performed only by the ADC instruction and the SBC instruction).
This flag is set to "1" by the SED instruction and cleared to "0" by the CLD instruction.
This flag is put in an undefined state immediately after hardware reset. As this flag directly affects arithmetic operations, be sure to initialize it.

(5) Break flag (B) ..... Bit 4

The break flag identifies whether or not an interrupt has been caused by the BRK instruction. The BRK instruction is used for program debugging and performs the same operation as an interrupt is performed by executing the BRK instruction.
The Processor status register is pushed to the stack, after the B flag is automatically set to "1" in case of the BRK instruction interrupt, or after the B flag is automatically cleared to "0" in case of the other interrupts.
There is no instruction to change the contents of this flag.

(6) Index X mode flag (T) .....Bit 5

When the Index X mode flag is “0,” arithmetic operations are performed between the Accumulator and the memory. When this flag is “1,” direct arithmetic operations and direct data transfer between one memory and another, between a memory and an I/O, or between one I/O and another without passing through the accumulator. An arithmetic operation result between memory 1 directly specified by the Index register X and memory 2 specified by an operand is stored into memory 1.

① When the T flag is “0” A ⟵ A ✻ M_2
② When the T flag is “1” M_1 ⟵ M_1 ✻ M_2

✻ : Denotes an arithmetic operation
A : Content of accumulataor
M_1 : Contents of memory 1 directly specified by the Index register X
M_2 : Contents of memory 2 specified by the operand

This flag is set to "1" by the SET instruction and cleared to "0" by the CLT instruction. This flag is in the undefined state immediately after hardware resetting. This flag has a direct effect on arithmetic operations. Accordingly, be sure to initialize it.

(7) Overflow flag (V).....Bit 6

The contents of the overflow flag have significance when addition and subtraction are performed assuming that one word is a signed binary number. When an addition or subtraction result exceeds the range of +127 to –128, this flag is set to “1.” When the BIT instruction is executed for other cases, the contents of bit 6 of the executed memory are put into the overflow flag.
This flag is cleared to “0” by the CLV instruction, but there is no instruction to set this flag to “1.” In the decimal operation mode, this flag is invalidated.

(8) Negative flag (N).....Bit 7

The negative flag is set to “1” when an arithmetic processing or data transfer result is negative (bit 7 is “1”). The contents of bit 7 of the executed memory are put into this flag when the BIT instruction is executed.
There is no instruction to change the contents of this flag.
In the decimal operation mode, this flag is invalidated.

1.8 Access area

In the 7470/7471/7477/7478 group, all ROM, RAM and I/O and the various control registers are located in the same memory area. Accordingly, the same instructions are used for data transfer and arithmetic operations without discriminating between a memory and an I/O.

The Program counter consists of 16 bits and the access space is 64K-byte of memory area: addresses 0000_{16} to $FFFF_{16}$.

The area of the least significant 256 bytes (addresses 0000_{16} to $00FF_{16}$) is called the “zero page,” and memories with a high frequency of use such as internal RAM, I/O ports and timers are located here. The area of the most significant 256 bytes (addresses $FF00_{16}$ to $FFFF_{16}$) is called the “special page,” and an internal ROM and interrupt vectors are located here.

The zero page and the special page can be accessed with 2 bytes by using each special addressing mode.

Figure 1.8.1 shows an outline of accsess area.

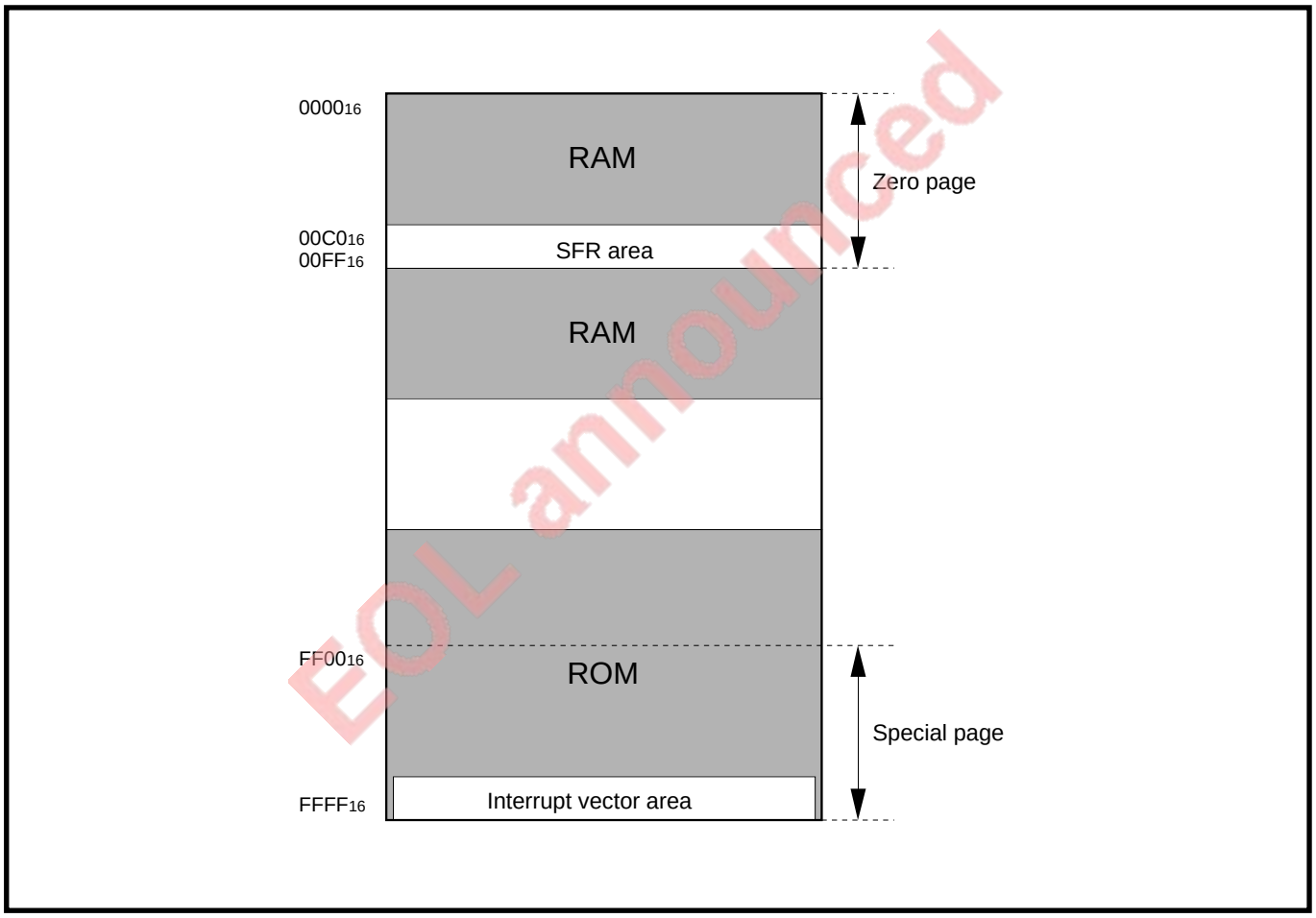


Fig. 1.8.1 Access area

1.8.1 Zero page (Addresses 0000_{16} to $00FF_{16}$)

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

To specify a memory or a register in this area, use the addressing mode shown in Table 1.8.1. In this area, especially, it is possible to access this area in a shorter instruction cycle by using the zero addressing mode.

1.8.2 Special page (Addresses $FF00_{16}$ to $FFFF_{16}$)

The area of 256 bytes from addresses $FF00_{16}$ to $FFFF_{16}$ is called the special page. The internal ROM and the interrupt vector area are located in this area.

To specify a memory or subroutine in this area, use the addressing mode shown in Table 1.8.1. In this area, especially, it is possible to jump to this area in a shorter instruction cycle by using the special page addressing mode.

Ordinary, subroutines with high frequency of use are located in this area.

Table 1.8.1 Addressing mode accessible to each area

Addressing mode (bytes required)	Zero page reference	Special page reference	Other area reference
Zero page (2)	○	—	—
Zero page indirect (2)	○	—	—
Zero page X (2)	○	—	—
Zero page Y (2)	○	—	—
Zero page bit (2)	○	—	—
Zero page bit relative (3)	○	—	—
Absolute (3)	○	○	○
Absolute X (3)	○	○	○
Absolute Y (3)	○	○	○
Relative (2)	○	○	○
Indirect (3)	○	○	○
Indirect X (2)	○	○	○
Indirect Y (2)	○	○	○
Special page (2)	—	○	—

1.9 Memory allocation

Figure 1.9.1 and Figure 1.9.2 show the memory allocation of 7470/7471/7477/7478 group.
The memories, I/Os and others located in the access area are explained below.

● **RAM**

An internal RAM is located in each area shown in Table 1.9.1. The internal RAM is used as a data storage area and a stack area for subroutine call and interrupt occurrence.
When the RAM is used as a stack area, be careful about subroutine nesting depth and interrupt levels so that the data in the RAM is not destroyed.

● **Special function register (SFR) (Addresses $00C0_{16}$ to $00FF_{16}$)**

The area from addresses $00C0_{16}$ to $00FF_{16}$ is assigned to the SFR (Special Function Register).
Various control registers such as I/O ports, timers, serial I/Os, A-D converters and interrupts are located in this SFR.

Figure 1.9.3 shows the special function register (SFR)memory map.

● **ROM**

An internal ROM is located in each area shown in Table 1.9.2.
The internal ROM is used to store data tables and programs. In the internal ROM, a vector area to store jump destination addresses upon a reset or occurrence of interrupt are assigned to addresses $FFEA_{16}$ to $FFFF_{16}$ in the 7470/7471 group and to addresses $FFE8_{16}$ to $FFFF_{16}$ in the 7477/7478 group.

Figure 1.9.4 shows the interrupt vector memory map.

Table 1.9.1 RAM area

Product	Range	Memory size
M3747xM2	Addresses 0000_{16} to $007F_{16}$	128 X 8-bit
M3747xM4/E4	Addresses 0000_{16} to $00BF_{16}$	192 X 8-bit
M3747xM8/E8	Addresses 0000_{16} to $00BF_{16}$, Addresses 0100_{16} to $01BF_{16}$	384 X 8-bit

Table 1.9.2 ROM area

Product	Memory type	Range	Memory size
M3747xM2	Mask ROM	Addresses $F000_{16}$ to $FFFF_{16}$	4K X 8-bit
M3747xM4	Mask ROM	Addresses $E000_{16}$ to $FFFF_{16}$	8K X 8-bit
M3747xE4	Programmable ROM		
M3747xM8	Mask ROM	Addresses $C000_{16}$ to $FFFF_{16}$	16K X 8-bit
M3747xE8	Programmable ROM		

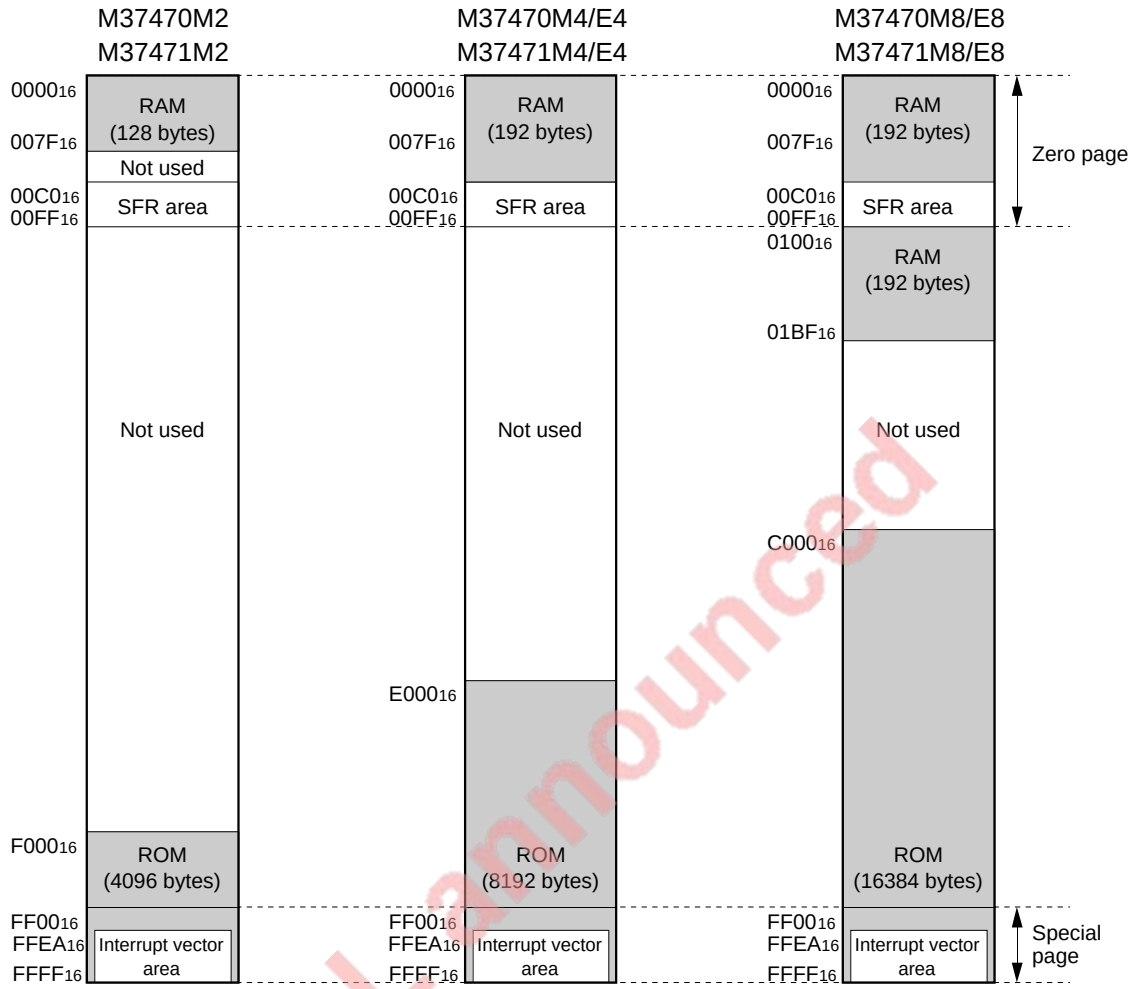


Fig. 1.9.1 Memory allocation of 7470/7471 group

1.9 Memory allocation

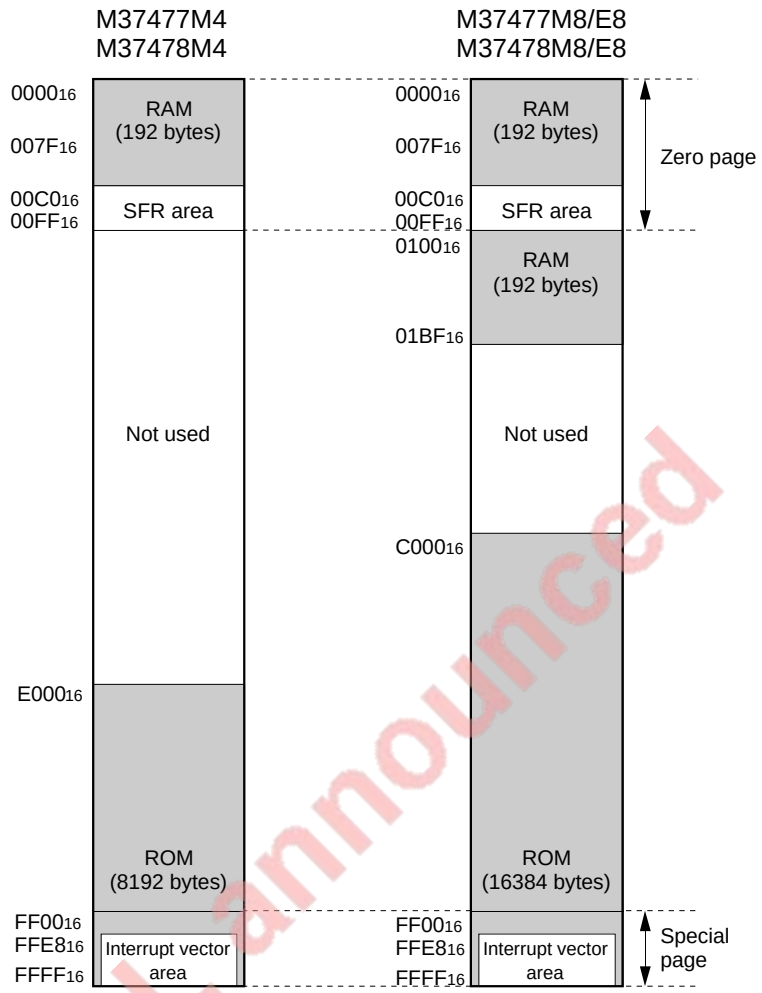


Fig. 1.9.2 Memory allocation of 7477/7478 group

00C0₁₆	Port P0
00C1₁₆	Port P0 direction register
00C2₁₆	Port P1
00C3₁₆	Port P1 direction register
00C4₁₆	Port P2
00C5₁₆	Port P2 direction register (Note 1)
00C6₁₆	Port P3
00C7₁₆	
00C8₁₆	Port P4
00C9₁₆	Port P4 direction register
00CA₁₆	Port P5 (Note 2)
00CB₁₆	
00CC₁₆	
00CD₁₆	
00CE₁₆	
00CF₁₆	
00D0₁₆	Port P0 pull-up control register
00D1₁₆	Port P1-P5 pull-up control register (Note 3)
00D2₁₆	
00D3₁₆	
00D4₁₆	Edge polarity selection register
00D5₁₆	
00D6₁₆	Input latch register
00D7₁₆	
00D8₁₆	
00D9₁₆	A-D control register
00DA₁₆	A-D conversion register
00DB₁₆	
00DC₁₆	Serial I/O mode register
00DD₁₆	Serial I/O register
00DE₁₆	Serial I/O counter ¦ Byte counter
00DF₁₆	

(Note 4)

00E0₁₆	Transmit/receive buffer register
00E1₁₆	Serial I/O status register
00E2₁₆	Serial I/O control register
00E3₁₆	UART control register
00E4₁₆	Baud rate generator
00E5₁₆	
00E6₁₆	
00E7₁₆	
00E8₁₆	
00E9₁₆	
00EA₁₆	
00EB₁₆	
00EC₁₆	
00ED₁₆	
00EE₁₆	
00EF₁₆	
00F0₁₆	Timer 1
00F1₁₆	Timer 2
00F2₁₆	Timer 3
00F3₁₆	Timer 4
00F4₁₆	
00F5₁₆	
00F6₁₆	
00F7₁₆	Timer FF register
00F8₁₆	Timer 12 mode register
00F9₁₆	Timer 34 mode register
00FA₁₆	Timer mode register 2
00FB₁₆	CPU mode register
00FC₁₆	Interrupt request register 1
00FD₁₆	Interrupt request register 2
00FE₁₆	Interrupt control register 1
00FF₁₆	Interrupt control register 2

(Note 5)

Notes 1: In the 7477/7478 group, this register is not located.
2: In the 7470/7477 group, this register is not located.
3: This address is allocated P1-P4 pull-up control register for the 7470/7477 group.
4: In the 7477/7478 group, this register is not located.
5: In the 7470/7471 group, this register is not located.

Fig. 1.9.3 Special function register (SFR) memory map

1.9 Memory allocation

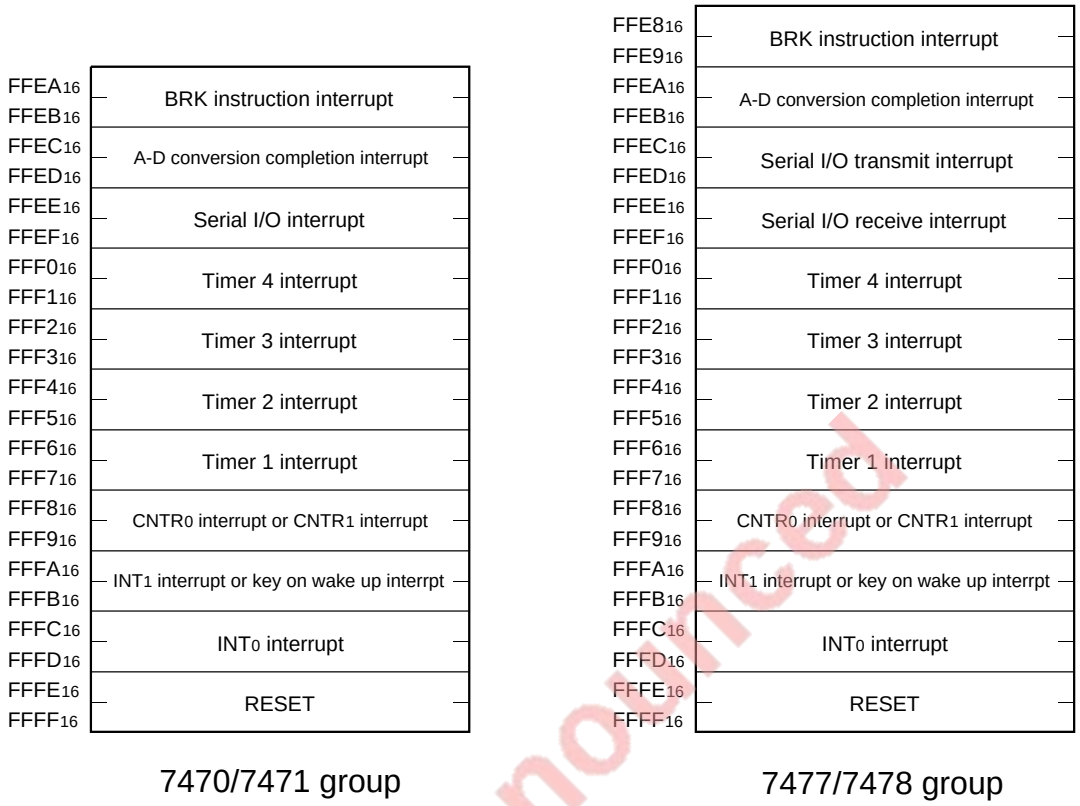


Fig. 1.9.4 Interrupt vector memory map

1.10 I/O pins

The 7470/7471/7477/7478 group is provided with the following I/O pins.

- I/O port (P0 to P5)
- Reset input ($\overline{\text{RESET}}$)
- Clock input/output (XIN, XOUT, XCIN, XCOUT)
- A-D convesion reference voltage input (VREF)
- Power supply voltage input (VCC, VSS, AVSS)

Notes 1: The 7470/7477 group is not provided with port P5 and pins XCIN and XCOUT.
2: The AVSS pin is dedicated to the 56P6N-A package product.

For an outline of each pin, refer to **"1.5 Pin description."**

1.10.1. I/O port

(1) I/O port writing and reading

■ **The input-only pin and the programmable I/O port set as input port**

The values (pin states) which input to the input-only pin and to the programmable I/O port set as input port can be read in by reading the Port register corresponding to each port.
When data is written into the Port register corresponding to each port, it can be only written in the Port register and has no effect on the pin state.

■ **The programmable I/O port set as an output port**

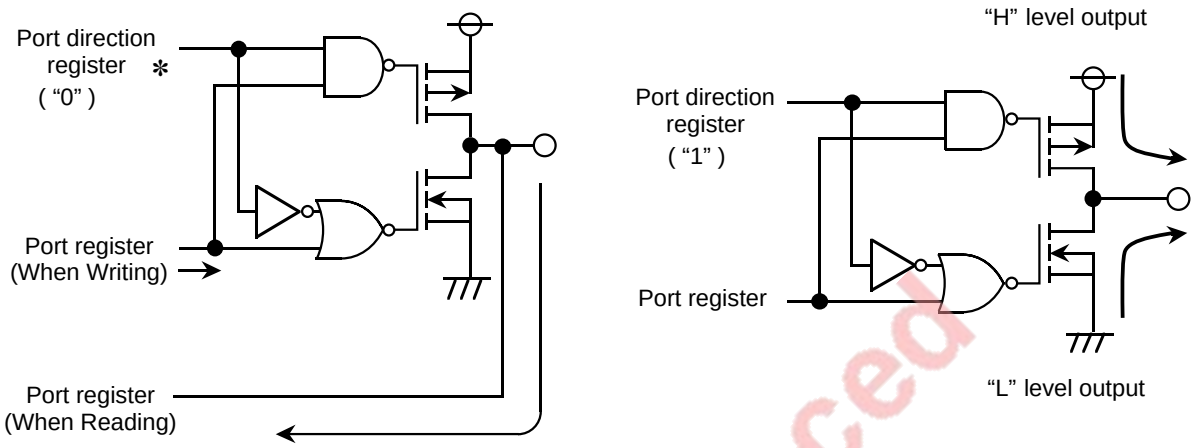
The value written into the Port register corresponding to the programmable I/O port set as an output port is output to the outside by way of a transistor.
When the Port register corresponding to each port has been read, each pin state is not read in but the value written into the Port register is read. Accordingly, if the output "H" voltage has been reduced or the output "L" voltage has been increased by an external load, the previous output value can be correctly read.

Figure 1.10.1 shows the I/O port writing and reading and Table 1.10.1 shows the port register address allocation.

1.10 I/O pins

At input : A write operation is enabled to the Port register.
Each pin state can be read in by reading the Port register.

At output : An output value can be set by writing to the Port register. The Port register can be read out.



✻ :The P channel transistor and the N channel transistor are in a cut-off state.

Fig. 1.10.1 I/O port writing and reading

Table 1.10.1 Port register address allocation

Port register	Address
P0	$00C0_{16}$
P1	$00C2_{16}$
P2	$00C4_{16}$
P3	$00C6_{16}$
P4	$00C8_{16}$
P5 (Note)	$00CA_{16}$

Note: The 7470/7477 group is not provided with P5.

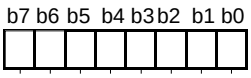
(2) Input/output selection of the programmable I/O ports

An input/output selection of the programmable I/O ports is made by the Port direction register corresponding to each port.

Figure 1.10.2 shows a structure of Port Pi (i = 0, 1, 2, 4) direction register.

Note: Each direction register is initialized into "00_{16}" at reset, so that the I/O ports are put into an input state.

Port Pi direction register



Port Pi direction register (PiD) (i = 0,1,2,4)
[Address $00C1_{16}$, $00C3_{16}$, $00C5_{16}$, $00C9_{16}$]

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	○	○

Notes 1: The 7477/7478 group is not provided with the port P2 direction register (input only).
2: The Port P4 is provided as below:
•7470/7477 group has 2 bits of $P4_0$ and $P4_1$.
•7471/7478 group has 4 bits of $P4_0$ to $P4_3$.

Fig. 1.10.2 Structure of Port Pi direction register (i=0, 1, 2, 4)

1.10 I/O pins

(3) Pull-up control

When input has been selected by the Port direction register, pull-up control can be exerted in bit units shown in Table 1.10.1 by the Port P0 pull-up control register (address $00D0_{16}$) or the Port P1–P5 pull-up control register* (address $00D1_{16}$). At this time, control is exerted by turning on and off the pull-up transistor.

*: The Port P1–P4 pull-up control register is arranged in the 7470/7477 group.

Note: Ports other than P0 cannot be controlled in one-bit units. For example, when $P1_0$ is pulled up at P1 (pull-up control in units of 4 bits), $P1_1$ to $P1_3$ are also pulled up.

Figure 1.10.3 shows a structure of Port P0 pull-up control register, and Figure 1.10.4 shows a structure of Port P1–P5 pull-up control register.

Port P0 pull-up control register

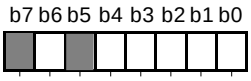
b7 b6 b5 b4 b3 b2 b1 b0

Port P0 pull-up control register [Address $00D0_{16}$]

B	Name	Function	At reset	R	W
0	Port $P0_0$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
1	Port $P0_1$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
2	Port $P0_2$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
3	Port $P0_3$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
4	Port $P0_4$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
5	Port $P0_5$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
6	Port $P0_6$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
7	Port $P0_7$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○

Fig. 1.10.3 Structure of Port P0 pull-up control register

Ports P1 to P5 pull-up control register



Ports P1 to P5 pull-up control register [Address $00D1_{16}$]

b	Name	Function	At reset	R	W
0	Ports $P1_0$ to $P1_3$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
1	Ports $P1_4$ to $P1_7$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
2	Ports $P2_0$ to $P2_3$ pull-up control bit (Note 2)	0 : No pull-up 1 : Pull-up	0	○	○
3	Ports $P2_4$ to $P2_7$ pull-up control bit (Notes 2, 3)	0 : No pull-up 1 : Pull-up	0	○	○
4	Ports $P4_0$ to $P4_3$ pull-up control bit (Note 4)	0 : No pull-up 1 : Pull-up	0	○	○
5	Nothing is allocated for this bit. This is write disabled bit and is undefined at reading.		?	?	×
6	Ports $P5_0$ to $P5_3$ pull-up control bit (Note 3)	0 : No pull-up 1 : Pull-up	0	○	○
7	Nothing is allocated for this bit. This is write disabled bit and is undefined at reading.		?	?	×

Notes 1 : In the 7470/7477 group, the P1 to P4 Pull-up control register is provided.
2 : In the 7477/7478 group, nothing is allocated to these bits. They are undefined at reading.
3 : In the 7470/7477 group, nothing is allocated to these bits. They are undefined at reading.
4 : The 7470/7477 group is provided with only $P4_0$ and $P4_1$.

Fig. 1.10.4 Structure of Ports P1 to P5 pull-up control register

1.10 I/O pins

1.10.2 Port block diagram

Figure 1.10.5 to Figure 1.10.9 show the block diagram of I/O ports.

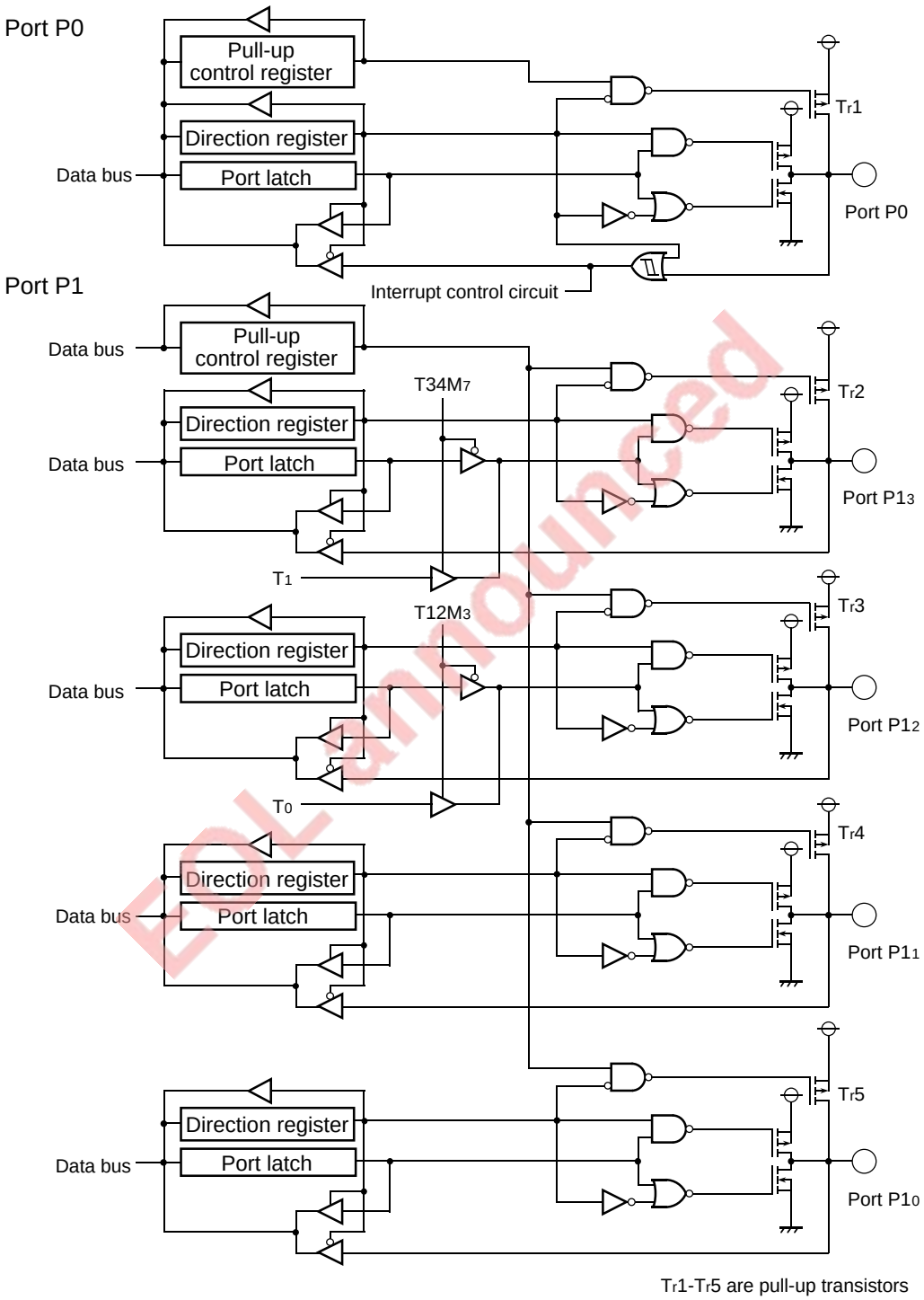


Fig. 1.10.5 Block diagram of Ports P0, P10 to P13

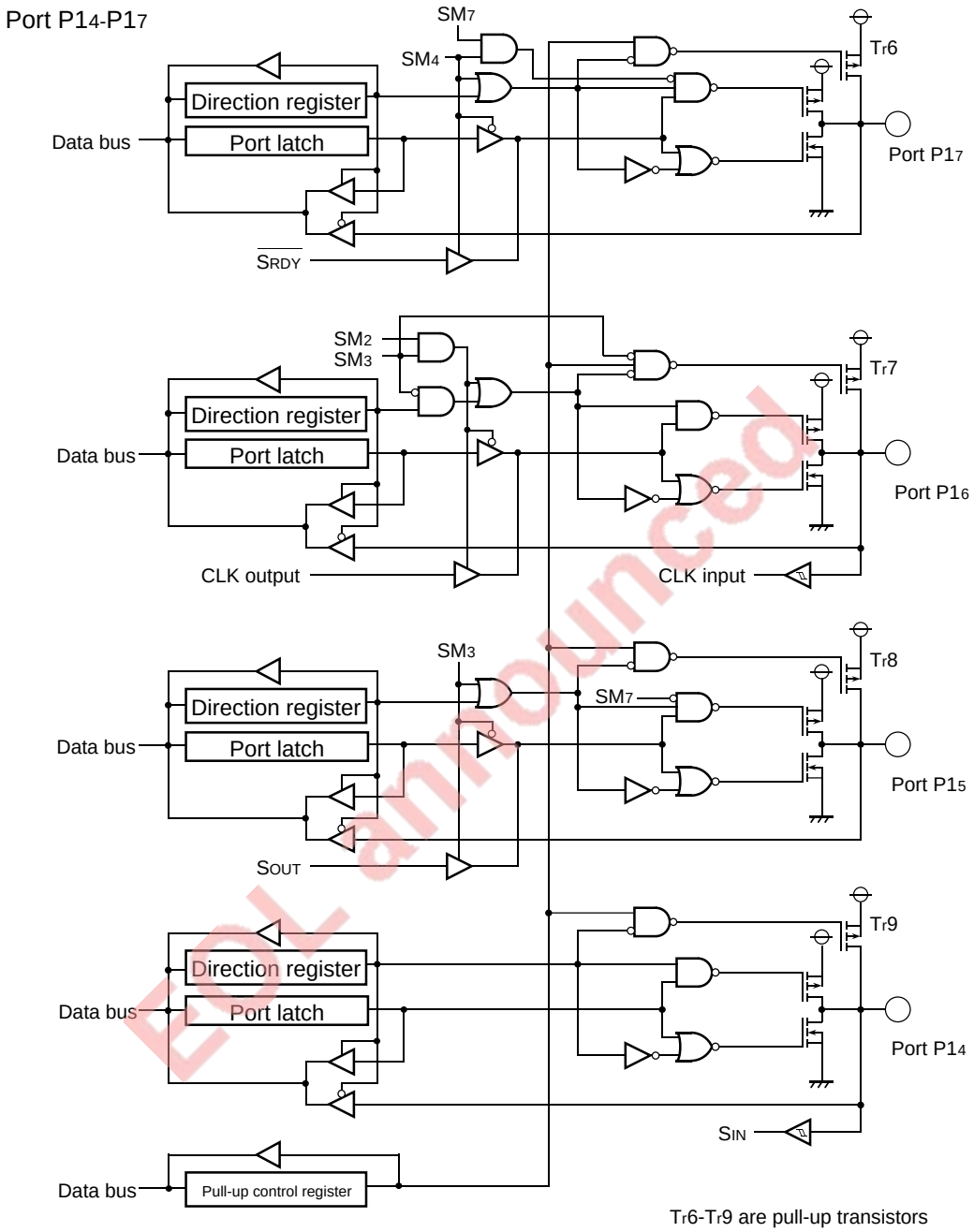


Fig. 1.10.6 Block diagram of Ports P14 to P17 (7470/7471 group)

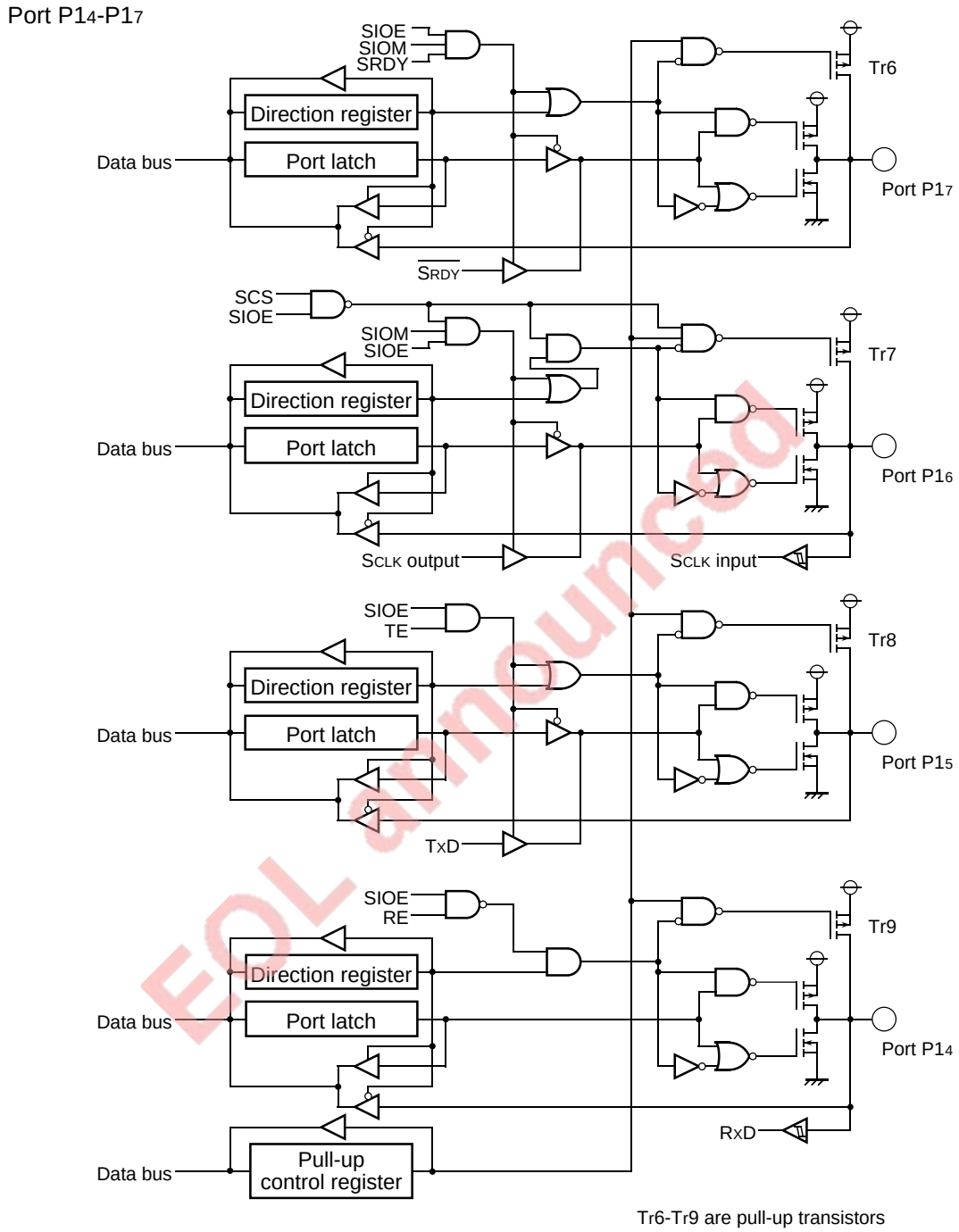
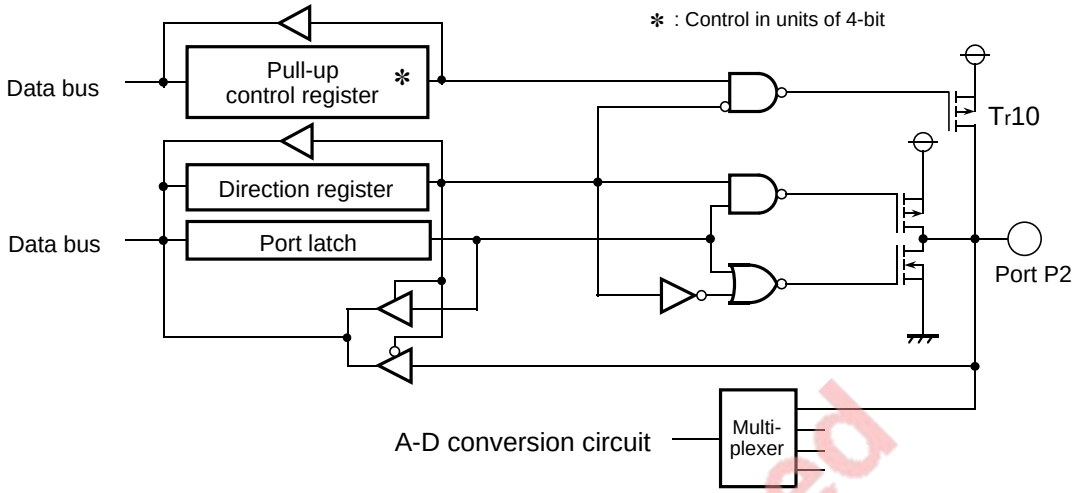
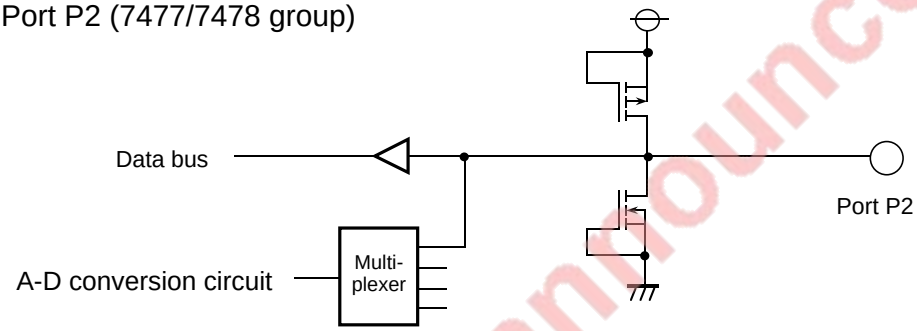


Fig. 1.10.7 Block diagram of Ports P14 to P17 (7477/7478 group)

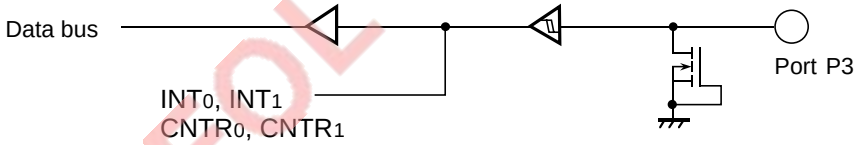
Port P2 (7470/7471 group)



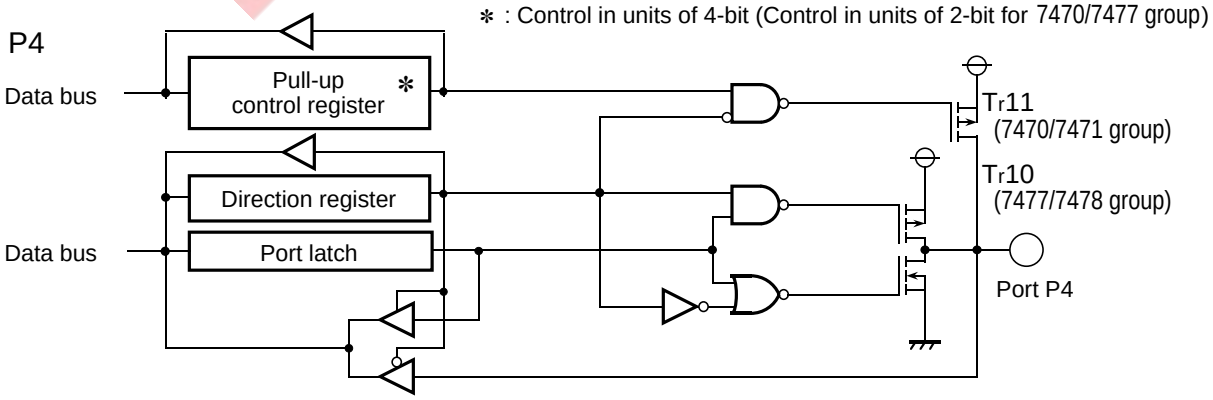
Port P2 (7477/7478 group)



Port P3



Port P4



Tr10 and Tr11 are pull-up transistors

Fig. 1.10.8 Block diagram of Ports P2 to P4

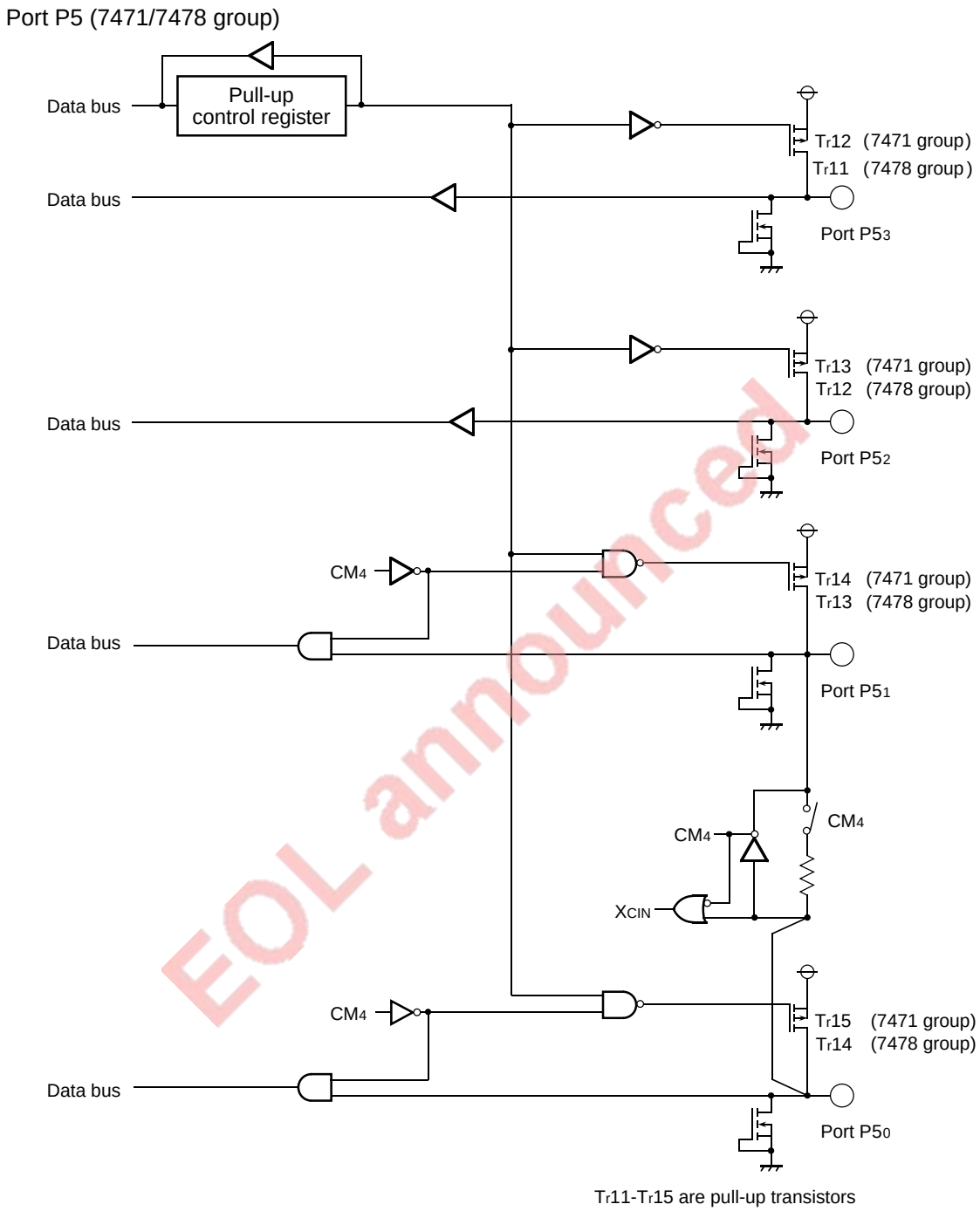


Fig. 1.10.9 Block diagram of Port P5

1.10.3 Notes on use

When using I/O ports, note the following.

(1) Modify of the content of I/O port latch

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

Reason

The bit managing instruction is read-modify-write instruction for reading and writing data by a byte unit. Accordingly, when this instruction is executed on one 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 as 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 as an output port: The bit value is read in the CPU, and is written to this bit after bit managing.

Make sure 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.
- Even when a bit of a port latch which is set as an input port is not specified with a bit managing instruction, its value may be changed in case where content of the pin differs from a content of the port latch.

∗ bit managing instructions: SEB and CLB instruction

(2) Pull-up control

To pull-up ports by software, note the following.

- When P1 is used in the serial I/O mode, the pull-up settings corresponding to $P1_4$ to $P1_7$ are invalidated (pull-up is impossible).
  Refer to the port block diagram for details.
- When a port is set in the output mode, the pull-up setting corresponding to the port is invalidated (pull-up is impossible).
- Ports other than P0 cannot be controlled in one-bit units. For example, when $P1_0$ is pulled up at P1 (pull-up control in units of 4 bits), $P1_1$ to $P1_3$ are also pulled up.

1.10 I/O pins

(3) Fix of a port input level in stand-by state

Fix input levels of an input and an I/O port for getting effect of low-power dissipation in stand-by state*, especially for the I/O ports of the N-channel open-drain.

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, make sure the following:

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

*: "Stand-by state": The stop mode by execution of the STP instruction or the wait mode by execution of the WIT instruction:

Reason

Even when setting as an output port with its direction register, in the following state:

- N-channel .....when the content of the port latch is "1"

the transistor becomes the OFF state, which causes the ports to be the high-impedance state. Make sure 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 an input and an I/O port are "undefined." This may cause power source current.

(4) Termination of unused pins

Table 1.10.2 shows a termination of unused pins.

Table 1.10.2 Termination of unused pins

Port	Terminations				
	Open	Pull-up (connect to V_{CC}) ports through a resistor (Note 2)	Pull down (connect to V_{SS}) ports through a resistor (Note 2)	Connect to V_{CC}	Connect to V_{SS}
P0 $P1_0$ to $P1_3$ $P1_5$, $P1_7$ P2 (7470/7471 group) P4	○ (Note 1)	○ (Note 4)	○ (Note 5)	×	×
$P1_4$, $P1_6$	○ (Note 3)	○ (Note 4)	○ (Note 5)	×	×
P2 (7477/7478 group)	○ (Note 1)	○ (Note 6)	○ (Note 6)	○ (Note 6)	○ (Note 6)
$P3_0$ to $P3_3$	×	○ (Note 6)	○ (Note 6)	○ (Note 6)	○ (Note 6)
P5 (Note 7)	○	○	○ (Note 8)	×	×
V_{REF}	×	×	×	○	×
AV_{SS}	×	×	×	×	○

Notes 1: A pin that can be opened at the unused time has a circuit that does not allow a current to flow into itself unless any read signal is internally input even if a medium-level input is applied at the open state.

2: For programmable I/O ports, do not connect two or more ports together through a resistor to V_{CC} or V_{SS}.

3: Note the following when setting them to the output mode and making the pins open.
- The ports function as input ports in the period from reset release till switching the ports to the output mode by software. Accordingly, the power source current may be increased depending on the input levels of the pins.
- If the Port direction register has been changed into the input mode by runaway or noise, re-set the Port direction register to the output mode periodically by software.

4: To pull up a pin, set the Port direction register and the Port latch so that this pin may be into the input mode or "H" output state.

5: To pull down a pin, set the Port direction register, the Port pull-up control register and the Port latch so that this pin may be put in the no pull-up transistor state in the input mode or in the "L" output state.

6: These pins are connect to the V_{CC} or V_{SS} without a resistor when the wiring is the shortest. However, they are connect to the V_{CC} or V_{SS} through a resistor. In addition, the $P3_3$ pin of the built-in programmable ROM version is used in common with the V_{PP} pin, insert a resistor of about 5 k in series and connect by the shortest wiring.

7: When using neither the $P5_0$ pin nor the $P5_1$ pin (used in common with the X_{CIN} and X_{COUT} pin), set bit 4 of the CPU mode register to "0" ($P5_0$ and $P5_1$ functions).

8: To pull down a pin, set the port pull-up control register so that a pull-up transistor will not be provided for this pin.

1.11 Interrupts

Interrupts are used in the following cases.

- When it is requested to execute higher-priority processing than the processing routine being executed.
- When it is necessary to observe any timing for processing.

The 7470/7471 group can generate interrupts from 12 sources and the 7477/7478 group can generate interrupts from 13 sources.

1.11.1 Description of interrupt source

■ Priority of interrupt

The interrupts are vector interrupts with a fixed priority sequence. When two or more interrupt requests occur at the same sampling time, they are accepted starting with the highest-priority interrupt. This priority is determined by hardware. However, a variety of priority processing can be executed by software when the interrupt control flags (interrupt enable bit and interrupt disable flag) are used.

■ Acceptance of interrupt

The corresponding interrupt request bit is set to "1" upon occurrence of an interrupt. When the following conditions are satisfied in this state, this interrupt is accepted.
For the details, refer to **"1.11.3 Interrupt control."**

① When the interrupt disable flag is cleared to "0" (interrupt enable state)
② When the interrupt enable bit is set to "1" (interrupt enable state)

Table 1.11.1 shows an interrupt priority, interrupt sources and vector addresses.

Table 1.11.1 Interrupt sources and priority

Priority	Interrupt source		Vector address		Remark
	7470/7471 group	7477/7478 group	High	Lower	
1	Reset (Note)		$FFFF_{16}$	$FFFE_{16}$	Non-maskable
2	INT_0 interrupt		$FFFD_{16}$	$FFFC_{16}$	Polarity programmable
3	INT_1 interrupt or key-on wake up interrupt		$FFFB_{16}$	$FFFA_{16}$	INT_1: polarity programmable
4	$CNTR_0$ interrupt or $CNTR_1$ interrupt		$FFF9_{16}$	$FFF8_{16}$	Polarity programmable
5	Timer 1 interrupt		$FFF7_{16}$	$FFF6_{16}$	
6	Timer 2 interrupt		$FFF5_{16}$	$FFF4_{16}$	
7	Timer 3 interrupt		$FFF3_{16}$	$FFF2_{16}$	
8	Timer 4 interrupt		$FFF1_{16}$	$FFF0_{16}$	
9	Serial I/O interrupt	Serial I/O receive interrupt	$FFEF_{16}$	$FFEE_{16}$	
10	A-D conversion completion interrupt	Serial I/O transmit interrupt	$FFED_{16}$	$FFEC_{16}$	
11	BRK instruction interrupt	A-D conversion completion interrupt	$FFEB_{16}$	$FFEA_{16}$	BRK instruction interrupt is non-maskable software interrupt
12	————	BRK instruction interrupt	$FFE9_{16}$	$FFE8_{16}$	

Note: A reset operation is performed in the same way as an interrupt, so it is described in the table.

(1) INT interrupt

When detecting a rising edge or a falling edge of each INT pin (INT_0, INT_1), the microcomputer generates an INT interrupt request.These polarity is selected by the edge polarity selection register (EG: Address $00D4_{16}$).

■ **$P3_0$, $P3_1$ pins**

The INT_0 and INT_1 pins are used in common with the $P3_0$ and $P3_1$ pins and always detect the levels of $P3_0$ and $P3_1$.

■ **In the stop mode/wait mode**

When bit 5 of the Edge polarity selection register is "0," a restoration can be attained by the INT interrupt from the stop mode/wait mode state provided by the STP/WIT instruction. For the details, refer to **"1.17 Low-power dissipation function.**"

■ **After reset**

At reset release, the Edge polarity selection register is cleared to "00_{16}," so the INT_0 and INT_1 interrupts generate the interrupt request by detecting a falling edge. At reset release, however, the Interrupt control register is put into the interrupt disable state, so any interrupt is not accepted.

Note: The INT_0 and INT_1 pins are used in common with input port $P3_0$ and $P3_1$, however, there is no register for switching between the INT pins and the ports, so the active edges of $P3_0$ and $P3_1$ are always detected. When these pins are used as ports, put the corresponding INT interrupt into the disable state.

In the INT interrupt enable state, the INT interrupt is generated by a pin level change, thereby causing a program run away.

(2) Key-on wake up interrupt

When bit 5 of the Edge polarity selection register is "1," the key-on wake up interrupt request is generated by applying the "L" level to any pin of P0 being an input port in the stop mode/wait mode provided by the STP/WIT instruction, so that a recovery can be attained from the stop mode/wait mode.

■ **After reset**

At reset release, bit 5 of the Edge polarity selection register is cleared to "0" so that the key-on wake up interrupt request does not occur in the stop mode/wait mode.

Notes 1: In modes other than the stop mode/wait mode, the key-on wake up interrupt is disabled.

2: To select the stop mode/wait mode by the STP/WIT instruction when the interrupt disable flag is cleared to "0" and bit 5 of the Edge polarity selection register is set to "1," set every input to P0 to "H."

1.11 Interrupts

Figure 1.11.1 shows a block diagram of interrupt input and key-on wake up circuit.

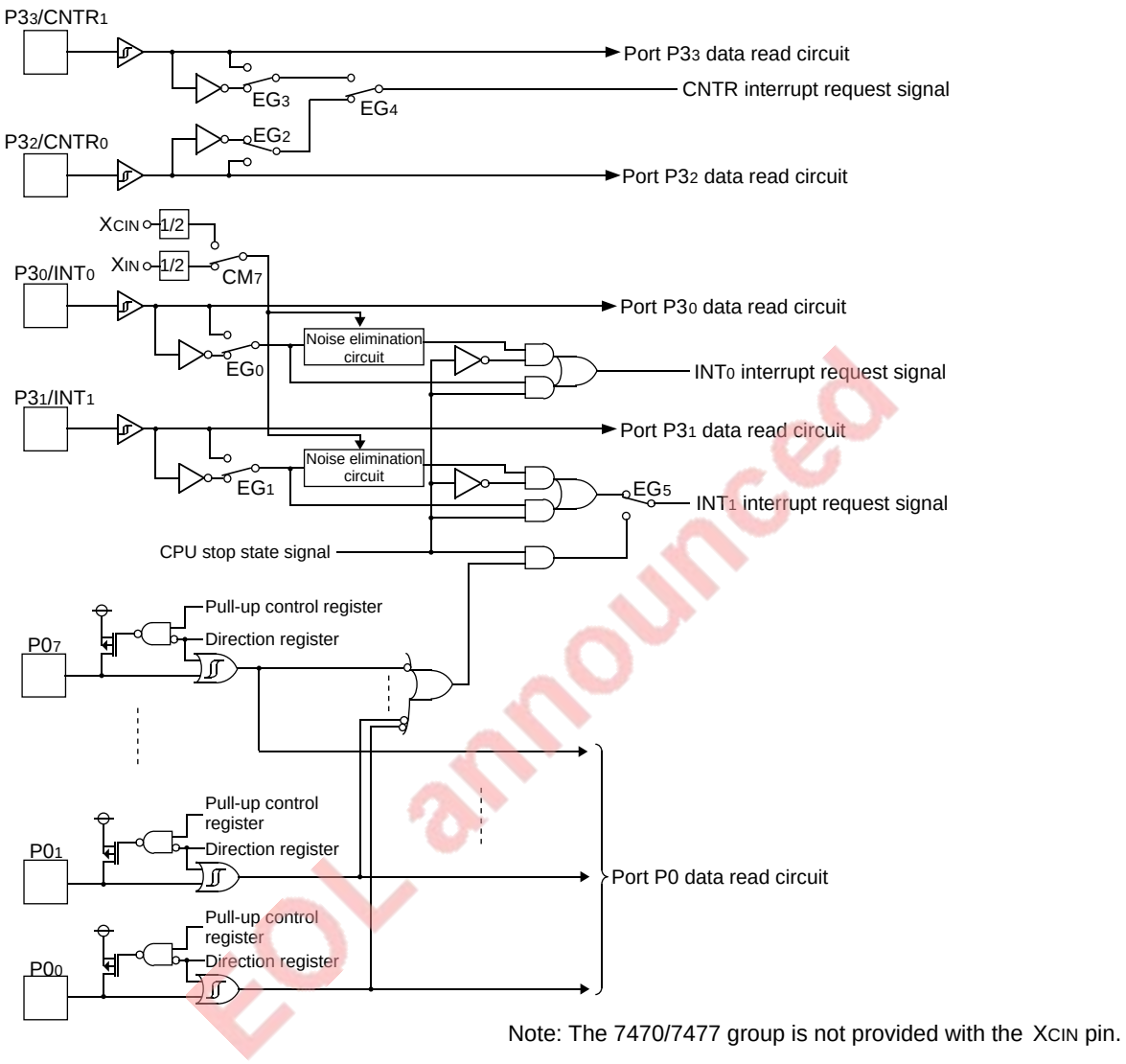


Fig. 1.11.1 Block diagram of interrupt input and key-on wake up circuit

(3) CNTR interrupt

When detecting a rising edge or a falling edge of each CNTR pin ($CNTR_0$, $CNTR_1$), the microcomputer generates an CNTR interrupt. For selecting the active edge of interrupt and the $CNTR_0$/$CNTR_1$ pin, the Edge polarity selection register (EG) is used.

■ **After reset**

At reset release, the Edge polarity selection register is cleared to "00_{16}," so the $CNTR_0$ interrupts generate the interrupt request by detecting a falling edge. At reset release, however, the Interrupt control register is put into the interrupt disable state, so any interrupt is not accepted.

Note: The $CNTR_0$ and $CNTR_1$ pins are used in common with input port $P3_2$ and $P3_3$, however there is no register for switching between the CNTR pins and the ports, so the active edges of $P3_2$ and $P3_3$ are always detected. When these pins are used as ports, put the corresponding CNTR interrupt into the disable state. In the CNTR interrupt enable state, the CNTR interrupt is generated by a pin level change, thereby causing a program run away.

(4) Timer interrupt

The microcomputer generates the interrupt request at the rise of the next count source after the respective timer overflows.

For the details of the timer interrupt, refer to **"1.12 Timers."**

(5) Serial I/O interrupt

There is a difference in the serial I/O interrupt between the 7470/7471 group and the 7477/7478 group.

■ **Serial I/O interrupt of 7470/7471 group**

An interrupt request is generated upon termination of the serial I/O transmit/receive.

■ **Serial I/O interrupt of 7477/7478 group**

The serial I/O transmit interrupt and the serial I/O receive interrupt are available.

● **Serial I/O transmit interrupt**

For the Serial I/O transmit interrupt, interrupt request generation timing can be selected by bit 3 of the Serial I/O control register (SIOCON: Address $00E2_{16}$) as shown below.

0: The data written in the Transmit buffer is transferred to the Transmit shift register, and when the Transmit buffer becomes empty, the interrupt request is generated.

1: The interrupt request is generated when a shift operation of the Transmit shift register terminates.

Note: When the transmit enable bit is set to the enable state, the Transmit buffer becomes empty and the transmit shift terminates. Accordingly, the interrupt request can be generated by selecting one of these sources. To use the transmit interrupt, set the transmit enable bit to "1," clear the transmit interrupt request bit to "0," and then set the transmit interrupt enable bit to the enable state.

● **Serial I/O receive interrupt**

When all data has been put in the Receive shift register and the contents of the shift register have been transferred to the Receive buffer, the interrupt request is generated.

For the details of the serial I/O interrupt, refer to **"1.13 Serial I/O."**

(6) A-D conversion completion interrupt

As soon as A-D conversion terminates, the interrupt request is generated.

For the details of the A-D conversion completion interrupt, refer to **"1.14 A-D Converter."**

(7) BRK instruction interrupt

This is the lowest-priority software interrupt without any corresponding interrupt enable flag, and not affected by the interrupt disable flag. (Non maskable)

For the details, refer to **"SERIES 740 SOFTWARE USER'S MANUAL."**

1.11 Interrupts

1.11.2 Operation description

(1) Interrupt operation

After an interrupt is accepted, the contents of the register shown below are automatically pushed to the stack area in sequence in the order of ①, ② and ③.

① Program counter high-order (PC_H)
② Program counter low-order (PC_L)
③ Processor status register (PS)

After the above register is pushed, a branch is made to the vector address of the accepted interrupt. When the RTI instruction is executed at the end of the interrupt processing routine, the contents of the above register which were pushed onto the stack area are popped to the respective registers in sequence in the order of ③, ② and ①, and the processing precedent to the acceptance of the interrupt is restarted.

Figure 1.11.2 shows the interrupt operation.

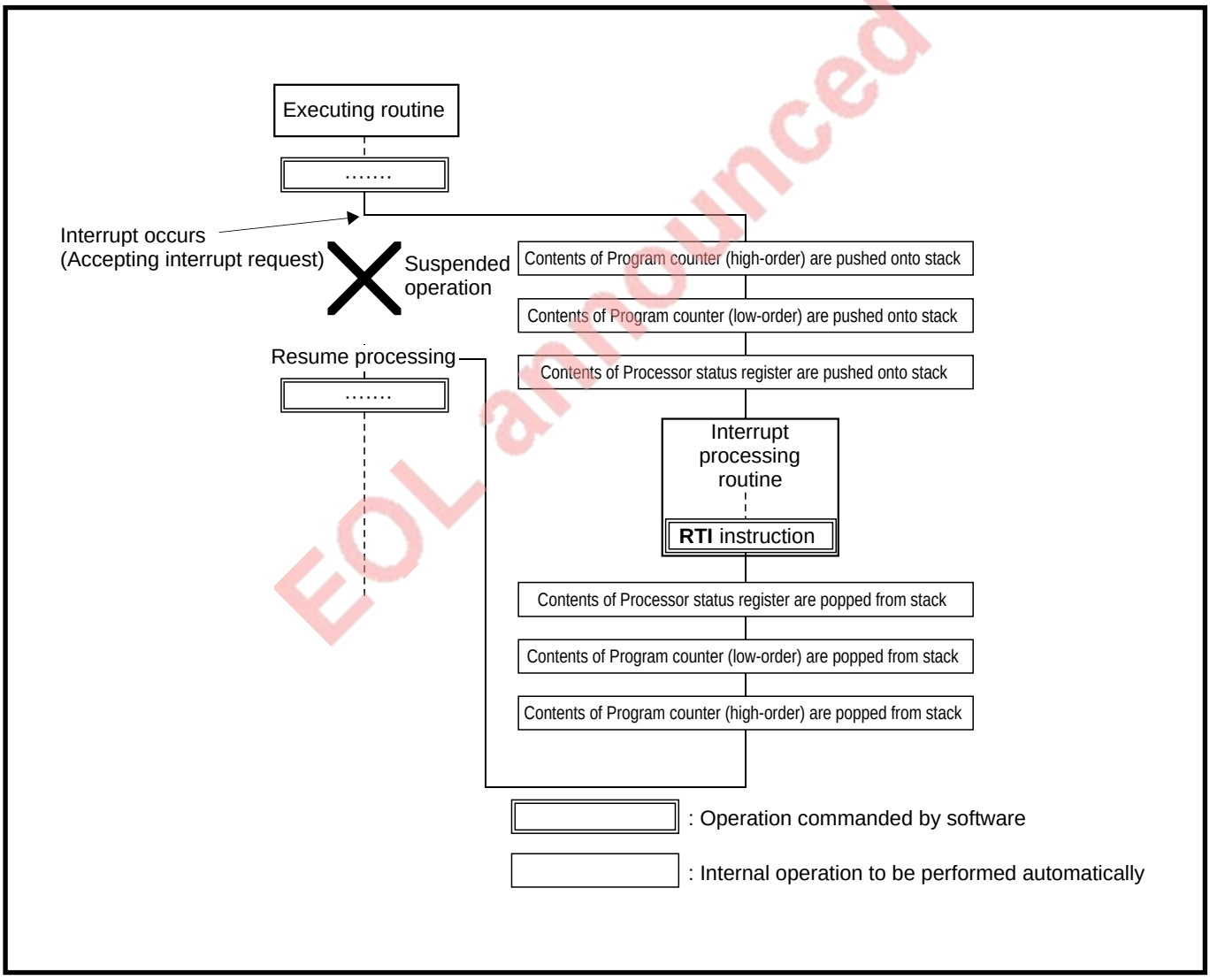


Fig. 1.11.2 Interrupt operation

(2) Processing upon acceptance of interrupt

When an interrupt is accepted, the following operations are automatically performed.

① The processing being executed is interrupted.

② The contents of the Program counter and the Processor status register are pushed to the stack area.
Figure 1.11.3 shows a change of the contents of the Program counter and the Stack pointer upon acceptance of the interrupt.

③ The vector address (start address of the interrupt processing routine) stored in the vector area corresponding to the generated interrupt concurrently with pushing is set in the Program counter and the interrupt processing routine is executed.

④ After the interrupt processing routine is started, the corresponding interrupt request bit is automatically cleared to "0." The interrupt disable flag is set to "1," thereby disabling a multi-interrupt.

To execute the interrupt processing routine, it is necessary to set a vector address in the vector area corresponding to each interrupt beforehand.

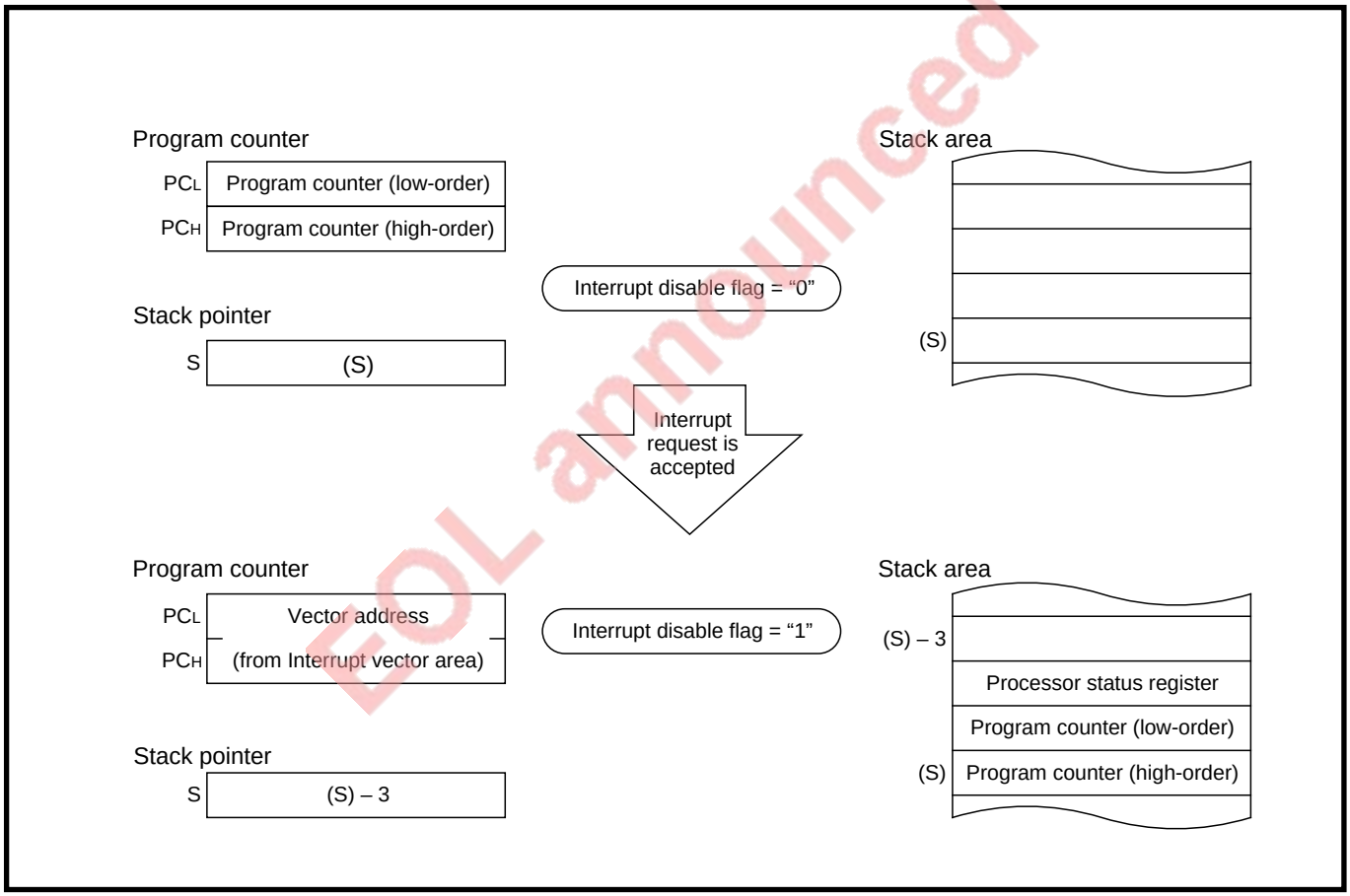


Fig. 1.11.3 Changes of contents of Program counter and Stack pointer upon acceptance of interrupt

1.11 Interrupts

(3) Timing after acceptance of interrupt

The interrupt processing routine starts with the machine cycle after termination of the instruction being executed.

Figure 1.11.4 shows a processing time up to the execution of the interrupt processing routine and Figure 1.11.5 shows a timing after acceptance of the interrupt.

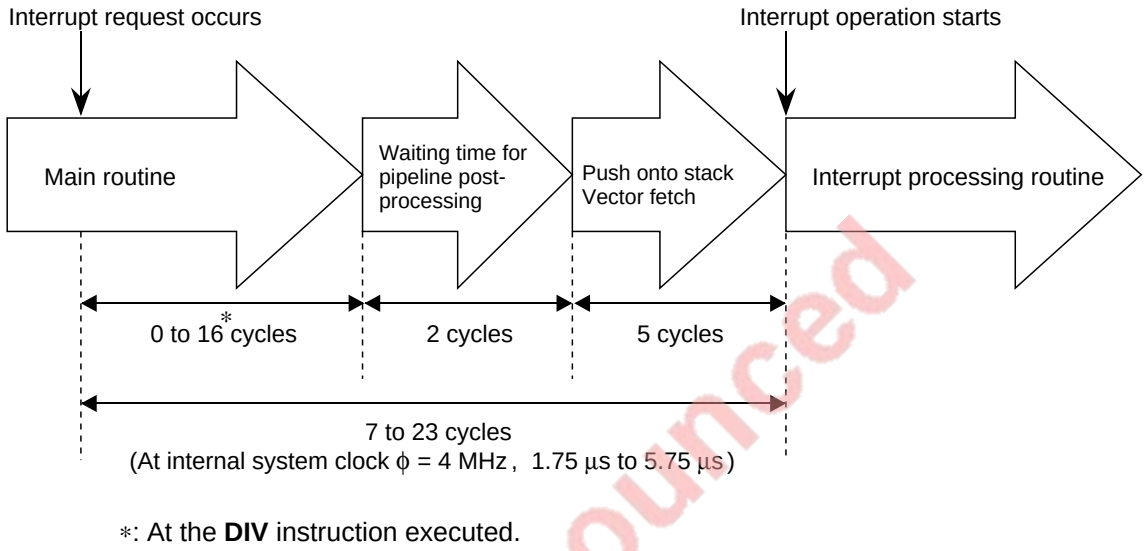


Fig. 1.11.4 Processing time up to the execution of interrupt processing routine

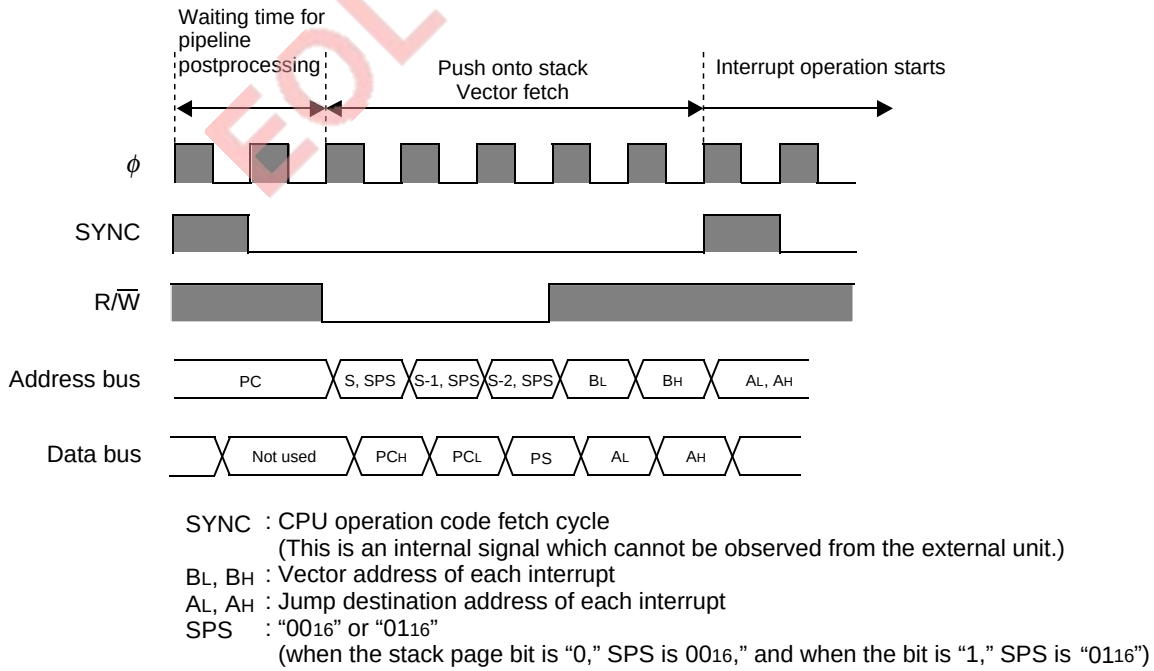


Fig. 1.11.5 Timing after acceptance of interrupt

1.11.3 Interrupt control

Regarding interrupts other than the BRK instruction, the acceptance of them can be controlled by the interrupt request bit, the interrupt enable bit and the interrupt disable flag. This section describes interrupt control other than the BRK instruction.

Figure 1.11.6 shows a interrupt control diagram.

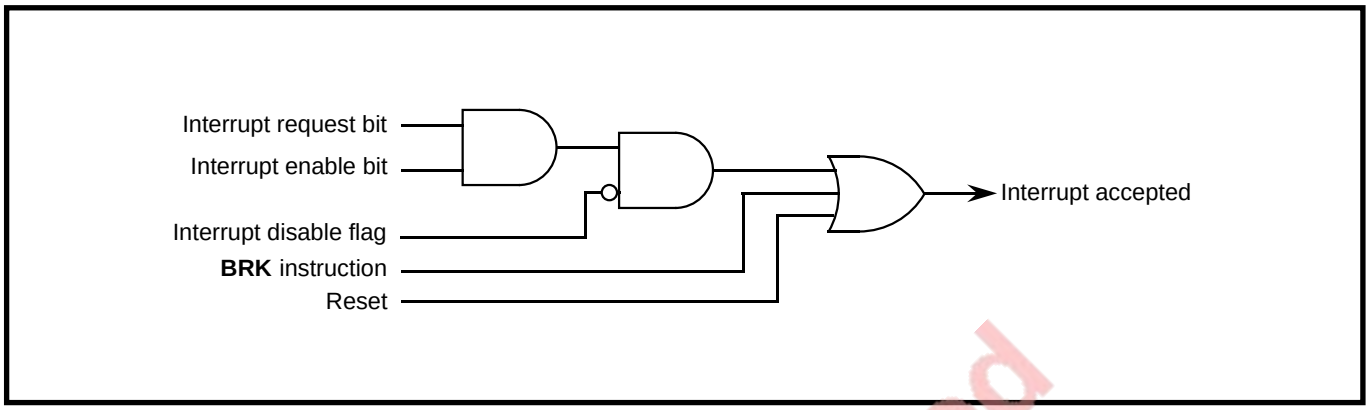


Fig. 1.11.6 Interrupt control diagram

The interrupt request bit, the interrupt enable bit and the interrupt disable flag function independently and do not affect one another. An interrupt is accepted when all the following conditions are satisfied.

- Interrupt request bit ..... "1"
- Interrupt enable bit ..... "1"
- Interrupt disable flag ..... "0"

The priority is determined by hardware. However, a variety of priority processing can be executed by software when the above flag and bits are used.

Table 1.11.2 shows a interrupt control bits for individual interrupt sources.

Table 1.11.2 Interrupt control bits for individual interrupt sources

Interrupt source	Interrupt request bits		Interrupt enable bits	
	Address	Bits	Address	Bits
Timer 1	$00FC_{16}$	b0	$00FE_{16}$	b0
Timer 2	$00FC_{16}$	b1	$00FE_{16}$	b1
Timer 3	$00FC_{16}$	b2	$00FE_{16}$	b2
Timer 4	$00FC_{16}$	b3	$00FE_{16}$	b3
Serial I/O (7470/7471 group)	$00FC_{16}$	b6 (Note 1)	$00FE_{16}$	b6 (Note 1)
Serial I/O receive (7477/7478 group)	$00FC_{16}$	b5 (Note 2)	$00FE_{16}$	b5 (Note 2)
Serial I/O transmit (7477/7478 group)	$00FC_{16}$	b6 (Note 2)	$00FE_{16}$	b6 (Note 2)
A-D conversion	$00FC_{16}$	b7	$00FE_{16}$	b7
INT_0	$00FD_{16}$	b0	$00FF_{16}$	b0
INT_1	$00FD_{16}$	b1	$00FF_{16}$	b1
$CNTR_0$/$CNTR_1$	$00FD_{16}$	b2	$00FF_{16}$	b2

Notes **1:** This bit is not provided in the 7477/7478 group.
2: This bit is not provided in the 7470/7471 group.

1.11 Interrupts

(1) Interrupt request bit

The interrupt request bits are assigned to each bit of the Interrupt request register 1(IR1: Address $00FC_{16}$) and the Interrupt request register 2(IR2: Address $00FD_{16}$).
If an interrupt request occurs, the corresponding interrupt request bit is set to "1."
The interrupt request bit is held in the "1" state until the interrupt is accepted. After it is accepted, this bit is automatically cleared to "0."
The interrupt request bit can be cleared to "0" by software but cannot be set to "1" by software.

(2) Interrupt enable bit

The interrupt enable bits are assigned to each bit of the Interrupt control register 1 (IE1: Address $00FE_{16}$) and the Interrupt control register 2 (IE2: Address $00FF_{16}$).

The interrupt enable bit controls the acceptance of the corresponding interrupt. When the interrupt enable bit is "0," the acceptance of the corresponding interrupt is disabled. If an interrupt request occurs when this bit is "0," the corresponding interrupt request bit is set to "1," but this interrupt is not accepted. In this case, the interrupt request bit is cleared to "0" by software or remains in the "1" state until the interrupt enable bit is set to "1."
When an interrupt enable bit is "1," the corresponding interrupt is enabled. If an interrupt request occurs when this bit is "1," this interrupt is accepted. (However, the interrupt disable flag that will be described later must be "0.") The interrupt enable bit can be cleared to "0" or set to "1" by software.

(3) Interrupt disable flag

The interrupt disable flag controls the acceptance of the interrupt, and is assigned to bit 2 of the Processor status register (PS).
When this flag is "1," the interrupt disable state is provided. When this flag is "0," the acceptance of interrupt is enable state.
This flag is set to "1" by the SEI instruction and cleared to "0" by the CLI instruction.
This flag is set to "1" (interrupt disable state) automatically after the interrupt processing routine. To use a multi-interrupt, set this flag to "0" by using the CLI instruction in the interrupt processing routine.

■ **Interrupt setting**

Set an interrupt according to the procedure shown below.

①The interrupt disable flag is set to "1."
②The interrupt enable bit is cleared to "0."
③For the INT interrupt or the CNTR interrupt, set the active edge in the Edge polarity selection register.
Select one of the above interrupts in bit 4 of the Edge polarity selection register because the $CNTR_0$ interrupt and the $CNTR_1$ interrupt can not be used simultaneously. (0: $CNTR_0$, 1: $CNTR_1$)
④The request bit of interrupt used is cleared to "0." (Refer to **"Table 1.11.2."**)
⑤The enable bit of interrupt used is set to "1." (Refer to **"Table 1.11.2."**)
⑥The interrupt disable flag is cleared to "0."

1.11.4 Notes on use

(1) When using $P3_0$ to $P3_3$ as input ports, put the corresponding INT interrupt or the CNTR interrupt into a disable state.

(2) Set the interrupt request bit and the interrupt enable bit for preparations for an interrupt in the following order.

① Clear the interrupt request bit to "0." (No interrupt request)

② Set the interrupt enable bit to "1." (Interrupt enabled)

When using the INT interrupt or the CNTR interrupt, first set the interrupt detection edge and then set the above items ① and ②. (Refer to (4) that will be described later.)

(3) An interrupt request bit can be cleared to "0" by software, but is still remained at the value precedent to a change immediately after execution of the clear instruction. For this reason, when executing the BBC or BBS instruction after changing an interrupt request bit, first execute the interrupt request bit change instruction and then execute the BBC or BBS instruction after one instruction or more.

(4) When the detection edge of the INT interrupt or that of the CNTR interrupt is switched, the corresponding interrupt request bit may be set to "1." Accordingly, perform setting referring to the register setting example shown in Figure 1.11.7.

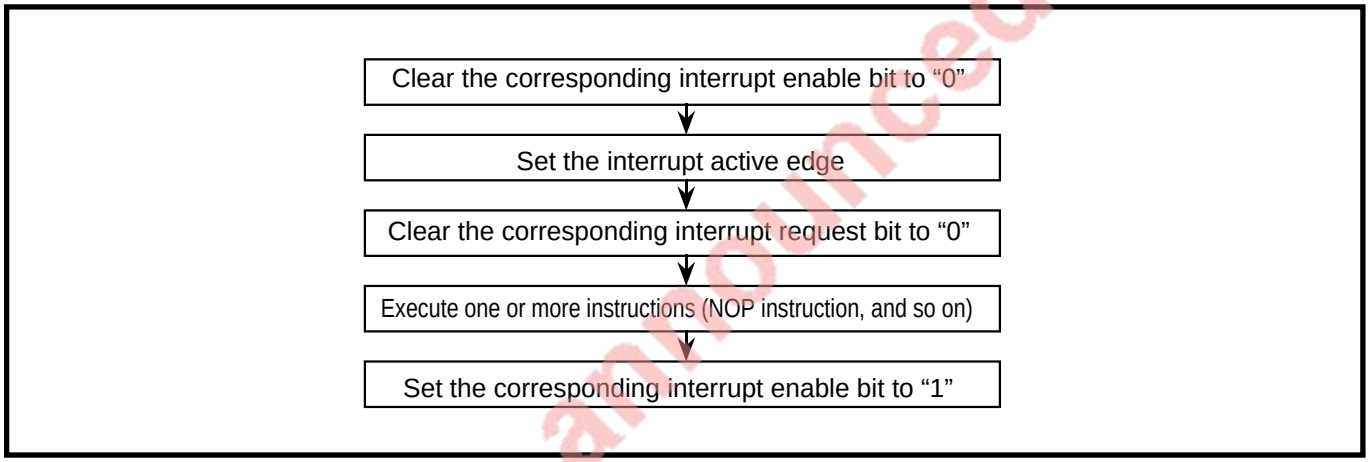


Fig. 1.11.7 Example of register setting

(5) Whether an interrupt is caused by the BRK instruction or not can be judged by the contents of the break flag of the Processor status register pushed on the stack area.

- Break flag = "1" : An interrupt has been caused by the BRK instruction
- Break flag = "0" : In case of the other interrupts

Note: Make this judgment in the interrupt processing routine.

(6) When an INT interrupt request is generated by executing the STP/WIT instruction in one of the following states, the stop mode/wait mode is released.

- When the active edge of the INT interrupt is a rising edge and the INT pin input level is "H"
- When the active edge of the INT interrupt is a falling edge and the INT pin input level is "L"

Accordingly, when executing the STP/WIT instruction, it is necessary to consider the input level of the INT pin and the polarity of the INT edge. Examples of countermeasures for it are shown below.

1. An example of a countermeasure for the case where the stop mode/wait mode is released at the rising edge of the INT pin input level

Point: To release the stop mode/wait mode normally, perform mode release processing in the INT interrupt processing routine only when the STP/WIT instruction was executed at the "L" INT pin input level.

< Main routine >

In the main routine, set the INT edge polarity according to the INT pin input level just precedent to execution of the STP/WIT instruction.

① INT interrupt disable
② ● Select the falling edge when the INT pin input level is "H."
● Select the rising edge when the INT pin input level is "L."
③ Clear the INT interrupt request bit to "0" and enable an INT interrupt after one instruction or more.
④ Clear the interrupt disable flag to "0."
⑤ Execute the STP/WIT instruction

< INT interrupt processing routine >

In the INT interrupt processing routine, change the active edge of the INT interrupt without performing release processing and proceed to the stop mode/wait mode in the case where the stop mode/wait mode is released by detecting a falling edge.

- When the INT pin input level is "H" (when a rising edge is detected)
  Processing for releasing the stop mode/wait mode
- When the INT pin input level is "L" (when a falling edge is detected)
  [1] Select the rising edge
  [2] Clear the INT interrupt request bit to "0"
  [3] Pop from stack
  [4] Perform ④ and ⑤ processing of the main routine.

2. An example of a countermeasure for the case where the stop mode/wait mode is released at the rising edge of the INT_0 pin input level or the falling edge of the INT_1 pin input level after the same signal is input to the INT_0 pin and the INT_1 pin.

Point: Select the INT interrupt, by the main routine, that becomes a source of release of the stop mode/wait mode according to the INT pin input level just precedent to execution of the STP/WIT instruction.

< Main routine >

① INT_0 and INT_1 interrupt disable
② Select the rising edge for the active edge of the INT_0 interrupt.
Select the falling edge for the active edge of the INT_1 interrupt.
③ ● When the INT pin input level is "H"
Clear the INT_1 interrupt request bit to "0" and enable the INT_1 interrupt after one instruction or more.
● When the INT pin input level is "L"
Clear the INT_0 interrupt request bit to "0" and enable the INT_0 interrupt after one instruction or more.
④ Clear the interrupt disable flag to "0."
⑤ Execute the STP/WIT instruction.

(7) In ordinary operation, if the pulse width of the INT input signal is 2 internal clocks ϕ ($f(X_{IN})/2$) or more by the built-in noise elimination circuit, it is accepted as an interrupt input. Input the INT input signal with a pulse width of 100 ns or more in the stop mode and the wait mode.

Reference: As a hardware-level means to prevent incorrect interrupt processing due to noise, a noise elimination circuit is incorporated in the INT_0 and INT_1 pins so that no interrupt can be generated by an "H" pulse (when the rising edge is selected) or an "L" pulse (when a falling edge is selected) of one machine cycle or less in modes other than the stop mode and the wait mode. As a software-level means, the levels of the INT_0 and INT_1 pins are judged at the beginning of the interrupt processing routine.

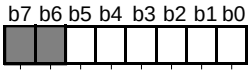
1.11.5 Related registers

(1) Edge polarity selection register (EG: Address 00D4$_{16}$)

The Edge polarity selection register selects an active edge of each INT interrupt and each CNTR interrupt selects a source of interrupt.

Figure 1.11.8 shows a structure of Edge polarity selection register.

Edge polarity selection register



Edge polarity selection register (EG) [Address 00D4$_{16}$]

B	Name	Function	At reset	R	W
0	INT$_0$ edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
1	INT$_1$ edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
2	CNTR$_0$ edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
3	CNTR$_1$ edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
4	CNTR$_0$/CNTR$_1$ interrupt selection bit	0 : CNTR$_0$ 1 : CNTR$_1$	0	○	○
5	INT$_1$ source selection bit (at STP or WIT instruction execution)	0 : P3$_1$/INT$_1$ 1 : P0$_0$ to P0$_7$ "L" level input (for key-on wake-up)	0	○	○
6, 7	Nothing is allocated for these bits. These are write disabled bits and are undefined at reading.		?	?	×

Fig. 1.11.8 Structure of Edge polarity selection register

1.11 Interrupts

(2) Interrupt request register 1 (IR1: Address 00FC16)
Interrupt request register 2 (IR2: Address 00FD16)

The Interrupt request register 1 and the Interrupt request register 2 consist of bits that indicate whether an interrupt request exists or not.

Figure 1.11.9 shows a structure of the Interrupt request register 1 and Figure 1.11.10 shows a structure of the Interrupt request register 2.

Interrupt request register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt request register 1 (IR1) [Address 00FC16]

B	Name	Function	At reset	R	W
0	Timer 1 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
1	Timer 2 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
2	Timer 3 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
3	Timer 4 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
4	Nothing is allocated for this bit. This is write disabled bit and is undefined at reading.		?	?	×
5	Serial I/O receive interrupt request bit (7477/7478 group)(Note)	0 : No interrupt request 1 : Interrupt requested	0	○	✻
6	Serial I/O interrupt request bit (7470/7471group) Serial I/O transmit interrupt request bit (7477/7478 group)	0 : No interrupt request 1 : Interrupt requested	0	○	✻
7	A-D conversion completion interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻

Note: In the 7470/7471group, nothing is allocated for bit 5. This is write disabled bit and is undefined at reading.
✻ : "0" is set by software, but not "1."

Fig. 1.11.9 Structure of Interrupt request register 1

Interrupt request register 2

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt request register 2 (IR2) [Address 00FD16]

B	Name	Function	At reset	R	W
0	INT0 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
1	INT1 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
2	CNTR0 or CNTR1 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
3 to 7	Nothing is allocated for these bits. There are write disabled bits and are undefined at reading.		?	?	×

✻ : "0" is set by software, but not "1."

Fig. 1.11.10 Structure of Interrupt request register 2

(3) Interrupt control register 1 (IE1: Address 00FE16)
Interrupt control register 2 (IE2: Address 00FF16)

The Interrupt control register 1 and the Interrupt control register 2 control the acceptance of interrupt by source.

Figure 1.11.11 shows a structure of the Interrupt control register 1 and Figure 1.11.12 shows a structure of an Interrupt control register 2.

Interrupt control register 1

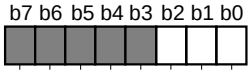
Interrupt control register 1 (IE1) [Address 00FE16]

B	Name	Function	At reset	R	W
0	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer 2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer 3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Timer 4 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Nothing is allocated for this bit. This is write disabled bit and is undefined at reading.		?	?	×
5	Serial I/O receive interrupt enable bit (7477/7478 group) (Note)	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Serial I/O interrupt enable bit (7470/7471 group) Serial I/O transmit interrupt enable bit (7477/7478 group)	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	A-D conversion completion interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Note: In the 7470/7471 group, Nothing is allocated for bit 5.
This is write disabled bit and undefined at reading.

Fig. 1.11.11 Structure of Interrupt control register 1

Interrupt control register 2



Interrupt control register 2 (IE2) [Address 00FF16]

B	Name	Function	At reset	R	W
0	INT0 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	INT1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	CNTR0 or CNTR1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3 to 7	Nothing is allocated for these bits. There are write disabled bits and are undefined at reading.		?	?	×

Fig. 1.11.12 Structure of Interrupt control register 2

1.12 Timers

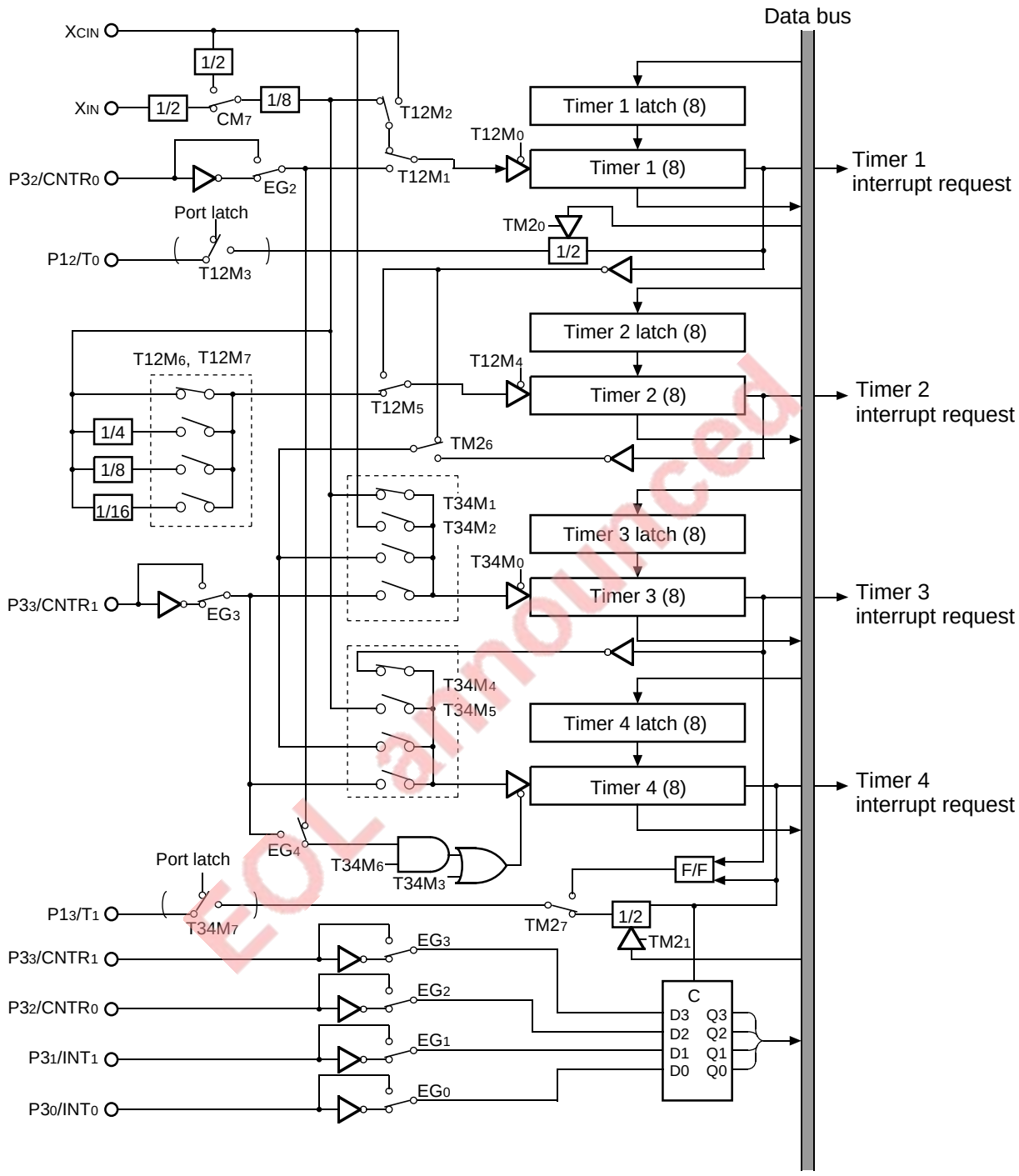
The 7470/7471/7477/7478 group has four 8-bit timers (Timer 1, Timer 2, Timer 3 and Timer 4) with an 8-bit timer latch. The division ratio of the timer is 1/(n+1) when the contents of the timer latch are n (n:0 to 255).
For the timer, the following modes can be selected by software setting.

- Timer mode
- Event counter mode
- Pulse output mode
- External pulse width measurement mode
- PWM mode

Table 1.12.1 shows the modes of each timer and Figure 1.12.1 shows a timer block diagram.

Table 1.12.1 Modes of each timer

Mode / Timer	Timer mode	Event counter mode	Pulse output mode	External pulse width measurement mode	PWM mode
Timer 1	○	○	○	×	×
Timer 2	○	×	×	×	×
Timer 3	○	○	×	×	○
Timer 4	○	○	○	○	



Notes 1: The 7470/7477 group is not provided with the XCIN pin.

2: The number (ex. EG3) described the right side of the register represents the bit number of the register.

Fig. 1.12.1 Timer block diagram

1.12 Timers

1.12.1 Operation description

In a write operation, the timer latch is specified at the same time when the timer is specified. If the timer is set to n_{16}, the timer latch is also set to n_{16} (n: 00_{16} to FF_{16}). After the timer starts to count,

① the timer value is counted down as $n_{16} \rightarrow (n\text{-}1)_{16} \rightarrow (n\text{-}2)_{16} \rightarrow \ldots \rightarrow 1_{16} \rightarrow 0_{16} \rightarrow FF_{16}$ at each rise of the count source.

② At the next rise of the count source after "FF_{16},"

- $(n\text{-}1)_{16}$ resulting from decrementing 1 from the timer latch value is set (reloaded) in the timer and then the timer continues to count.
- When an overflow occurs, the interrupt request bit is set to "1."

Note: When the interrupt is accepted, the interrupt request bit changes from "1" to "0." It can be clearned to "0" but cannot be set to "1" by software.

Figure 1.12.2 shows a timer count timing.

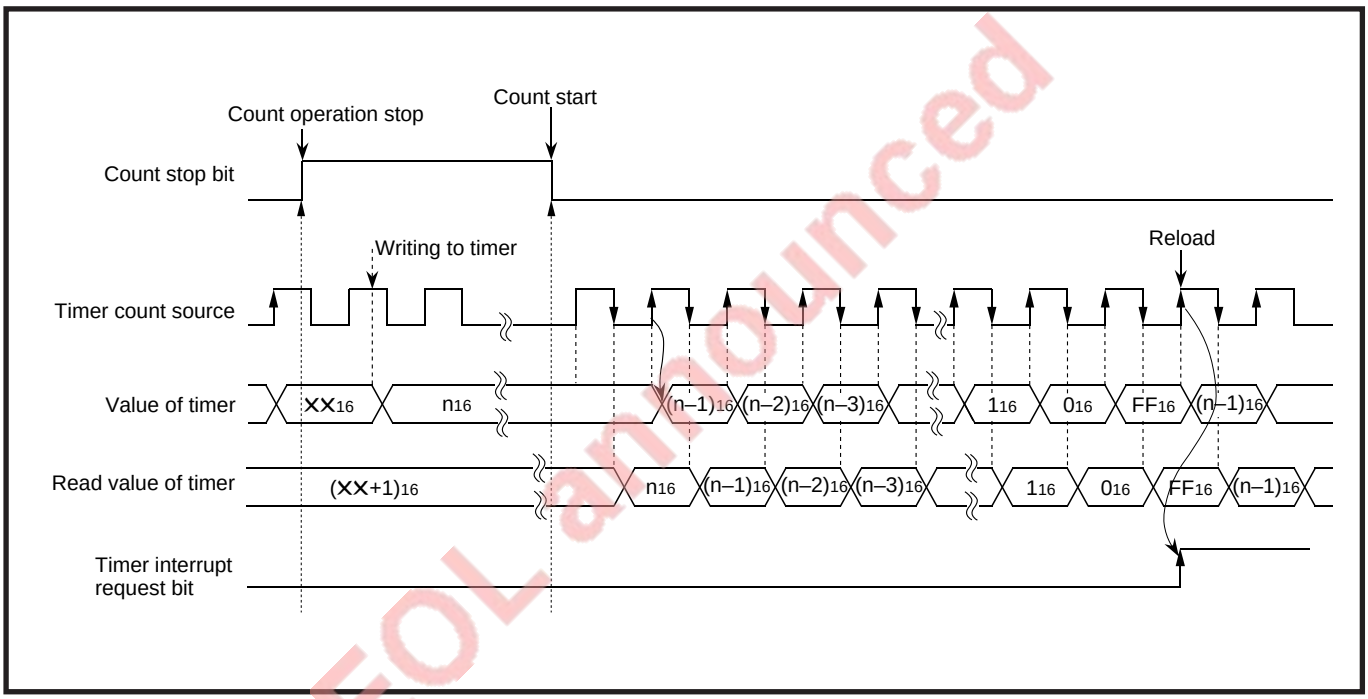


Fig. 1.12.2 Timer count timing

1.12.2 Description of modes

(1) Timer mode

The operations of the timer modes are as explained below.

① Start of count operation

When the count stop bit is cleared to "0," a count operation starts. When there is a count source input, the contents of the timer are decremented by 1.

Note: Because the count stop bit is "0" immediately after reset release, the count operation is automatically started after reset release.

② Reload operation

When the timer overflows, the value resulting from decrementing 1 from the contents of the timer latch is transferred (reloaded) to the timer.

③ Interrupt operation

■Timer interrupt

When the timer overflows, an interrupt request occurs, so that the interrupt request bit is set to "1." The acceptance of interrupt is controlled by the interrupt enable bit of each timer.

④ Stop of count operation

When the counter stop bit is set to "1" by software, the count operation stops. (The count operation continues until the count stop bit is set to "1.")

Figure 1.12.3 shows an example of timer mode operation.

Count period

Count period T(s) = 1 ÷ count source frequency × (the timer initial value + 1)

Timer mode operation example

- OF: Overflow
- RL: Reload
- n: Timer initial value

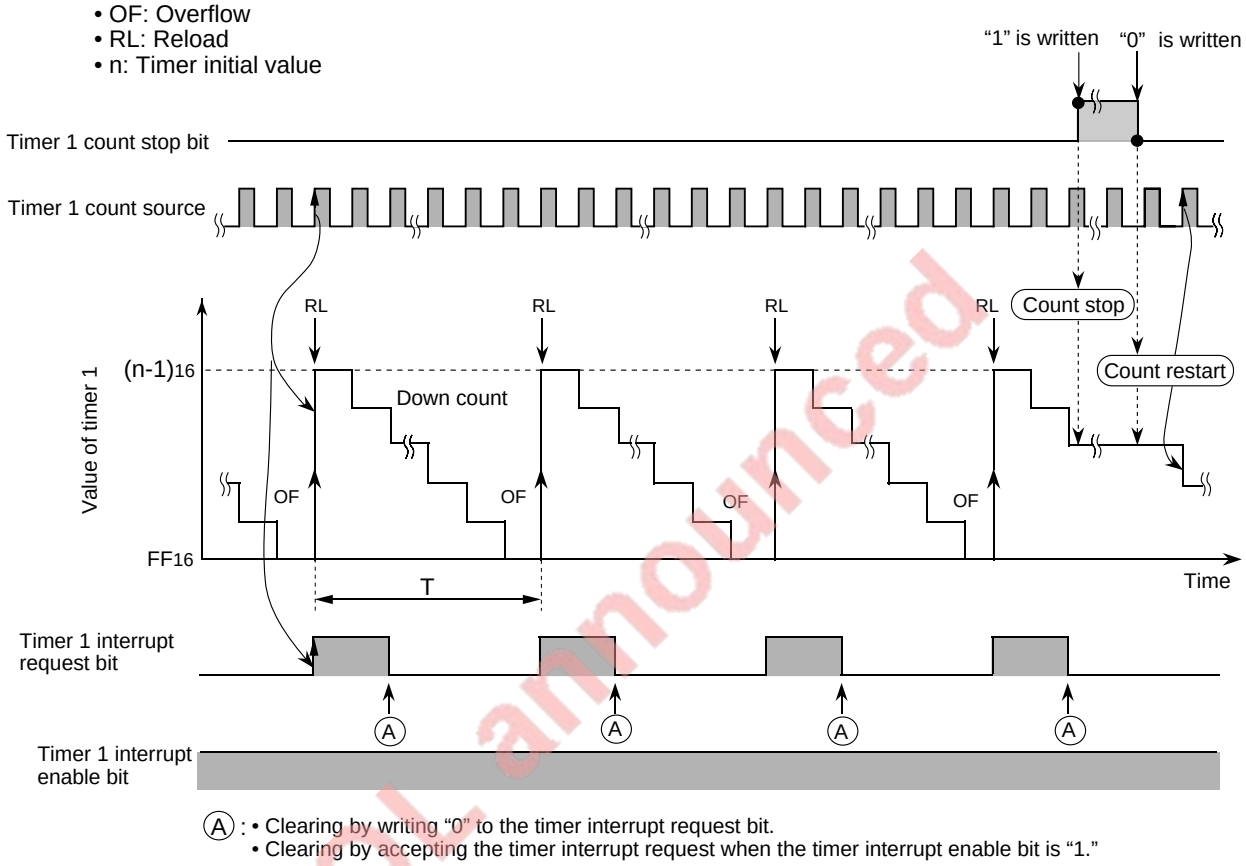


Fig. 1.12.3 Example of timer mode operation

[Setting method]

① Set a value to be used according to the count source setting for timers. The count operation of the timer is stopped. Refer to **"Table 1.12.2 Setting for count stop."**

Table 1.12.2 Setting for count stop

Setting item / Timer	Timer 12 mode register (T12M: Address $00F8_{16}$)		Timer 34 mode register (T34M: Address $00F9_{16}$)	
	b4	b0	b3	b0
Timer 1	–	1	–	–
Timer 2	1	–		
Timer 3	–	–	–	1
Timer 4			1	–

② Count source selecting

Select a count source according to the count source setting for timers shown in Table 1.12.3 to Table 1.12.6. Note that selectable count sources are different among timers.

Because the 7470/7477 group is not provided with an XCIN pin, do not select f(XCIN) as a count source.

Table 1.12.3 Setting for timer 1 count source

Setting item / Count source to be selected	CPU mode register (CM: Address $00FB_{16}$)	Timer 12 mode register (T12M: Address $00F8_{16}$)	
	b7	b2	b1
f(XIN)/16	0	0	0
f(XCIN)	– (Note)	1	
f(XCIN)/16	1	0	
External clock input from CNTR0 pin.	–	–	1

Note: When f(XCIN) is selected as a timer count source, f(XIN) or f(XCIN) can be selected as a system clock.

Table 1.12.4 Setting for timer 2 count source

Setting item / Count source to be selected	CPU mode register (CM: Address $00FB_{16}$)	Timer 12 mode register (T12M: Address $00F8_{16}$)		
	b7	b7	b6	b5
f(XIN)/16	0	0	0	0
f(XIN)/64		0	1	
f(XIN)/128		1	0	
f(XIN)/256		1	1	
f(XCIN)/16	1	0	0	
f(XCIN)/64		0	1	
f(XCIN)/128		1	0	
f(XCIN)/256		1	1	
Timer 1 overflow signal	–	–	–	1

1.12 Timers

Table 1.12.5 Setting for Timer 3 count source

Setting item / Count source to be selected	CPU mode register (CM: Address 00FB16)	Timer mode register 2 (TM2: Address 00FA16)	Timer 34 mode register (T34M: Address 00F916)	
	b7	b6	b2	b1
f(XIN)/16	0	–	0	0
f(XCIN)	– (Note)		0	1
f(XCIN)/16	1		0	0
Timer 1 overflow signal	–	0	1	0
Timer 2 overflow signal		1		
External clock input from CNTR1 pin	–	–	1	1

Note: When f(XCIN) is selected as a timer count source, f(XIN) or f(XCIN) can be selected as a system clock.

Table 1.12.6 Setting for Timer 4 count source

Setting item / Count source to be selected	CPU mode register (CM: Address 00FB16)	Timer mode register 2 (TM2: Address 00FA16)	Timer 34 mode register (T34M: Address 00F916)		
	b7	b6	b6	b5	b4
f(XIN)/16	0	–	0	0	1
f(XCIN)/16	1				
Timer 1 overflow signal	–	0		1	0
Timer 2 overflow signal		1		(Note)	(Note)
Timer 3 overflow signal		–		0	0
External clock input from CNTR1 pin				1	1

Note: If the Timer 1 overflow signal is selected as a Timer 2 count source at [b5,b4] = [1,0], the Timer 4 count source becomes the Timer 1 overflow signal regardless of bit 6 of Timer mode register 2.

③ Set a count value in the timer.
Refer to **"Table 1.12.7 Address allocation for timer."**

Table 1.12.7 Address allocation of timer

Timer	Address
Timer 1(T1)	00F016
Timer 2(T2)	00F116
Timer 3(T3)	00F216
Timer 4(T4)	00F316

④ When a value is set according to the count start setting shown in Table 1.12.8, the timer starts to count.

Table 1.12.8 Count start setting

Setting item / Timer	Timer 12 mode register (T12M: Address 00F816)		Timer 34 mode register (T34M: Address 00F916)	
	b4	b0	b3	b0
Timer 1	–	0	–	–
Timer 2	0	–		
Timer 3	–	–	–	0
Timer 4			0	–

(2) Event counter mode

In the event counter mode, the same operations as those in the timer mode are performed, with the exception that the signal input from the $CNTR_0$ pin becomes a count source of Timer 1 and the signal input from the $CNTR_1$ pin becomes a count source of Timer 3 and Timer 4.
The operation in the event counter mode are described below.

① Start of count operation

After the count stop bit is cleared to "0," a count operation starts. Each time a count source is input, the contents of the timer are decremented by 1.
For the active edge of count source, a rise or a fall can be selected by the edge polarity selection register (address $00D4_{16}$).

Note: Because the count stop bit is "0" immediately after reset release, the count operation is automatically started after reset release but the count source is not a CNTR pin input (operates as the timer mode).

② Reload operation

When the timer overflows, the value resulting from decrementing 1 from the contents of the timer latch is transferred (reloaded) to the timer.

③ Interrupt operation

■Timer interrupt

When the timer overflows, an interrupt request occurs, so that the interrupt request bit is set to "1." The acceptance of interrupt is controlled by the interrupt enable bit of each timer.

■CNTR interrupt

An interrupt request is generated from the edge of the count source input from the $CNTR_0$ pin or the $CNTR_1$ pin, so that the interrupt request is set to "1." The acceptance of interrupt is controlled by the interrupt enable bit of each timer.
The edge polarity selection register selects an active edge of count source and a $CNTR_0$/$CNTR_1$ interrupt.

④ Stop of count operation

When the counter stop bit is set to "1" by software, the count operation stops. (The count operation continues until "1" is set in the count stop bit.)

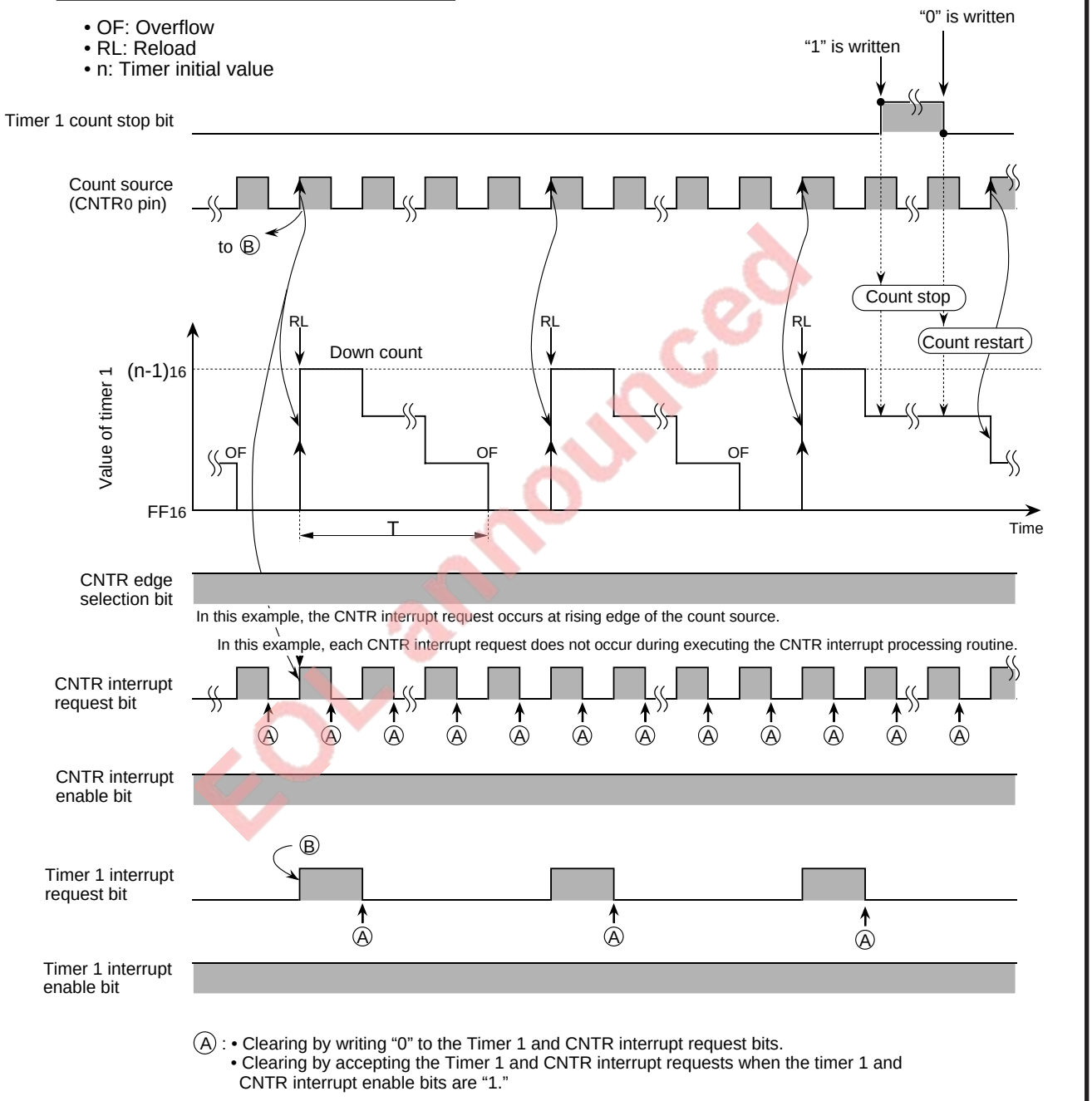
Figure 1.12.4 shows an example of event counter mode operation.

Count period

Count period T(s) = 1 ÷ count source frequency × (the timer initial value + 1)

Event counter mode operation example

- OF: Overflow
- RL: Reload
- n: Timer initial value



Ⓐ : • Clearing by writing "0" to the Timer 1 and CNTR interrupt request bits.
• Clearing by accepting the Timer 1 and CNTR interrupt requests when the timer 1 and CNTR interrupt enable bits are "1."

Fig. 1.12.4 Example of event counter mode operation

[Setting method]

① The count operation of a timer to be used is stopped.
Refer to **"Table 1.12.2 Setting for count stop."**

② Select a count source according to the event counter mode setting shown in Table1.12.9.

③ Set a count value in the timer.
Refer to **"Table 1.12.7 Address allocation of timer."**

④ Start a count operation of timer to be used.
Refer to **"Table 1.12.8 Count start setting."**

Table. 1.12.9 Event counter mode setting

Timer to be used	Count source	Edge polarity selection register (EG: Address $00D4_{16}$)		Timer 12 mode register (T12M: Address $00F8_{16}$)	Timer 34 mode register (T34M: Address $00F9_{16}$)				
		b3	b2	b1	b6	b5	b4	b2	b1
Timer 1	$CNTR_0$	–	Select (Note)	1	–				
Timer 3	$CNTR_1$	Select (Note)	–	–	–			1	1
Timer 4					0	1	1	–	

Note: 0: The falling edge (An input, when it is inverted, becomes a count source).
1: The rising edge (An input itself becomes a count source).

(3) Pulse output mode

The pulse output mode is a mode resulting from adding a pulse output operation to a timer mode operation. In this mode, a pulse whose polarity is inverted at each overflow is output from the T_0 (Timer 1 overflow signal/2) pin and the T_1 (Timer 4 overflow signal/2) pin.
The operations in the pulse output mode are described below.

① Start of count operation
After the count stop bit is set to "0," a count operation starts. Each time a count source is input, the contents of the timer are decremented by 1.
Note: Because the count stop bit is "0" immediately after reset release, the count operation is automatically started immediately after reset release but no pulse is output.

② Reload operation
When the timer overflows, the value resulting from decrementing 1 from the contents of the timer latch is transferred (reloaded) to the timer.

③ Pulse is output
- A pulse whose polarity is inverted at each overflow is output from the T_0 pin and the T_1 pin.
- "H" or "L" can be selected as a level for a start of pulse output by each division flip-flop.
- A pulse output is started from the moment when the T_0 or T_1 pin output is selected by the Timer 12 mode register or the Timer 34 mode register.

④ Interrupt operation
■ Timer interrupt
When the timer overflows, an interrupt request occurs, so that the interrupt request bit is set. The acceptance of interrupt is controlled by the interrupt enable bit of each timer.

⑤ Stop of count operation
When "1" is set in the counter stop bit by software, the count operation stops. (The count operation continues until "1" is set in the count stop bit.)

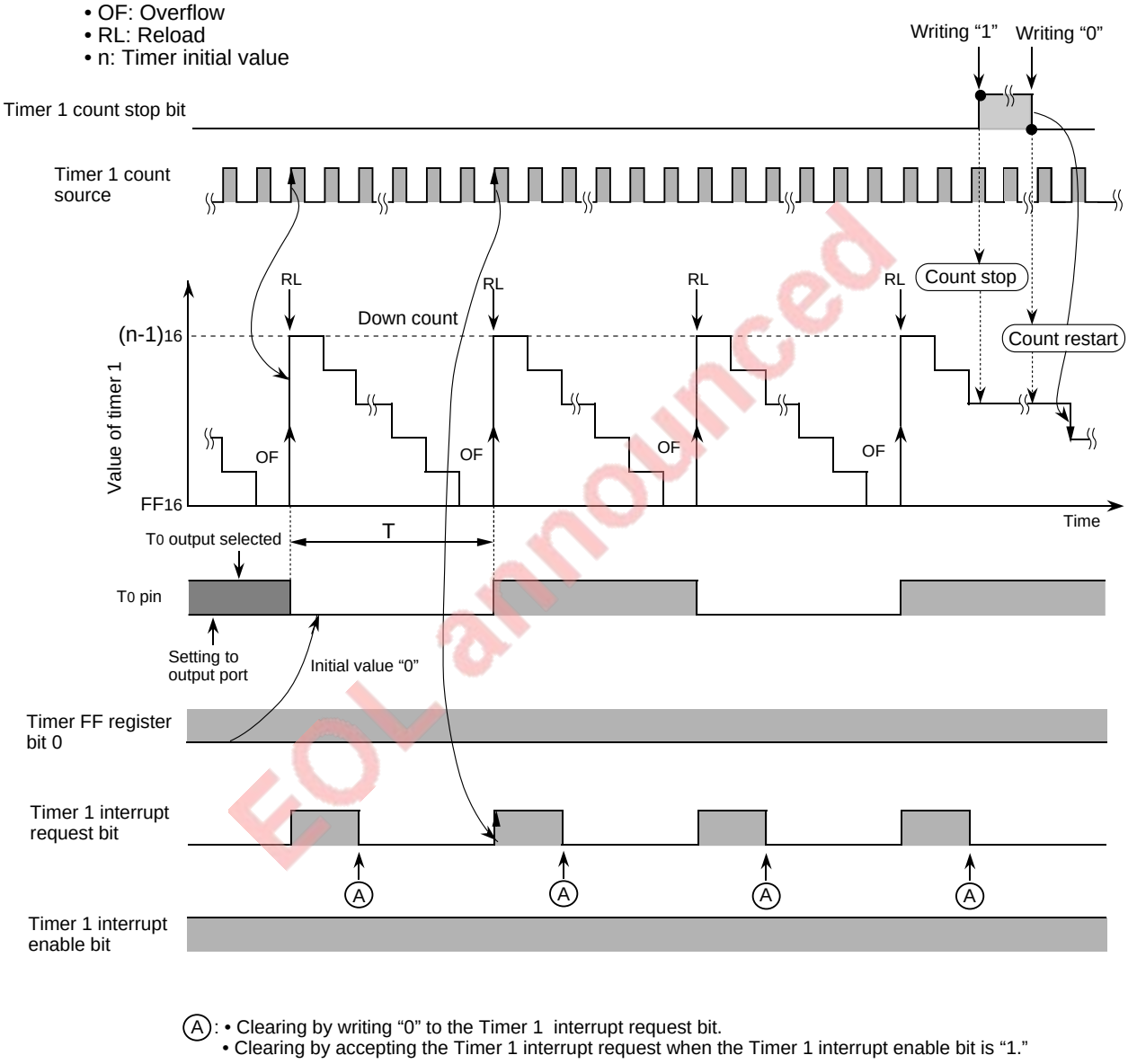
Figure 1.12.5 shows a example of pulse output mode operation.

Count period

Count period T(s) = 1 ÷ count source frequency × (the timer initial value + 1)

Pulse output mode operation example

- OF: Overflow
- RL: Reload
- n: Timer initial value



A: • Clearing by writing "0" to the Timer 1 interrupt request bit.
• Clearing by accepting the Timer 1 interrupt request when the Timer 1 interrupt enable bit is "1."

Fig. 1.12.5 Example of pulse output mode operation

[Setting method]

① The count operation of a timer to be used is stopped.
Refer to **"Table 1.12.2 Setting for count stop."**

② Set the pulse output mode according to the pulse output mode initial value setting shown in Table 1.12.10. However, set the timer FF register after setting timer mode register 2.

Table 1.12.10 Pulse output mode initial value setting

Setting item / Timer to be used	Timer mode register 2 (TM2: Address 00FA16)		Timer FF mode register (TF: Address 00F716)	
	b1	b0	b1	b0
Timer 1	–	1	–	Select (Note)
Timer 4	1	–	Select (Note)	–

Note: 0: The initial value becomes "0."
1: The initial value becomes "1."

③ Set the pulse output mode according to the pulse output mode setting shown in Table 1.12.11. The T_0 output port and the T_1 output port are used in common with $P1_2$ and $P1_3$, respectively. Accordingly, set the bit 2 and bit 3 of the port P1 direction register to the output mode.

Table 1.12.11 Pulse output mode setting

Timer	Output pin	Timer 12 mode register (T12M: Address 00F816)	Timer 34 mode register (T34M: Address 00F916)	
		b3	b7	b6
Timer 1	T_0 Timer 1 overflow signal / 2	1	–	
Timer 4	T_1 Timer 4 overflow signal / 2	–	1	0

④ Set a count value in Timer 1 (T_0 output) and Timer 4 (T_1 output).

⑤ When a value is set according to the count start setting shown in Table 1.12.8, the timer starts to count.

Note: When resetting a value in the Timer FF register, be sure to observe the setting methods of the above items ① to ⑤.

(4) External pulse width measurement mode

The external pulse width measurement mode is used to measure a pulse width ("H" or "L") input from the $CNTR_0$ or $CNTR_1$ pin.

The operations in the external pulse width measuring mode are described below.

① Start of count operation

After the count stop bit is cleared to "0," a count operation starts. Each time a count source is input, the contents of the timer are decremented by 1.

Note: Because the count stop bit is "0" immediately after reset release, the count operation is automatically started after reset release but the count source is not a CNTR pin input. At this time, the external pulse width measurement mode is not provided.

② Reload operation

When the timer overflows, the value resulting from decrementing 1 from the contents of the timer latch is transferred (reloaded) to the timer.

③ External pulse width measurement mode

- The "H" or "L" level of a pulse can be selected as a pulse measuring period by the Edge polarity selection register.
- The difference between the initial value of the timer and the counter value at a count stop becomes a measured pulse width.
- A reload operation by reading the count value is not performed automatically. To perform measurement continuously, re-set the initial value by software.

④ Interrupt operation

■Timer interrupt

When the timer overflows, an interrupt request occurs, so that the interrupt request bit is set to "1." The acceptance of interrupt is controlled by the interrupt enable bit of each timer.

■CNTR interrupt

An interrupt request is generated from the edge of the pulse input from the $CNTR_0$ pin or the $CNTR_1$ pin, so that the interrupt request bit is set to "1." Interrupt acceptance is controlled by the interrupt enable bit.

The pulse active edge and the $CNTR_0$/$CNTR_1$ interrupt are selected by the Edge polarity selection register.

⑤ Stop of count operation

The count operation terminates at the falling edge (at "H" level pulse width measurement) or the falling edge ("L" level pulse width measurement) of the CNTR pin input. This operation is also terminated by setting "1" in the count stop bit by software.

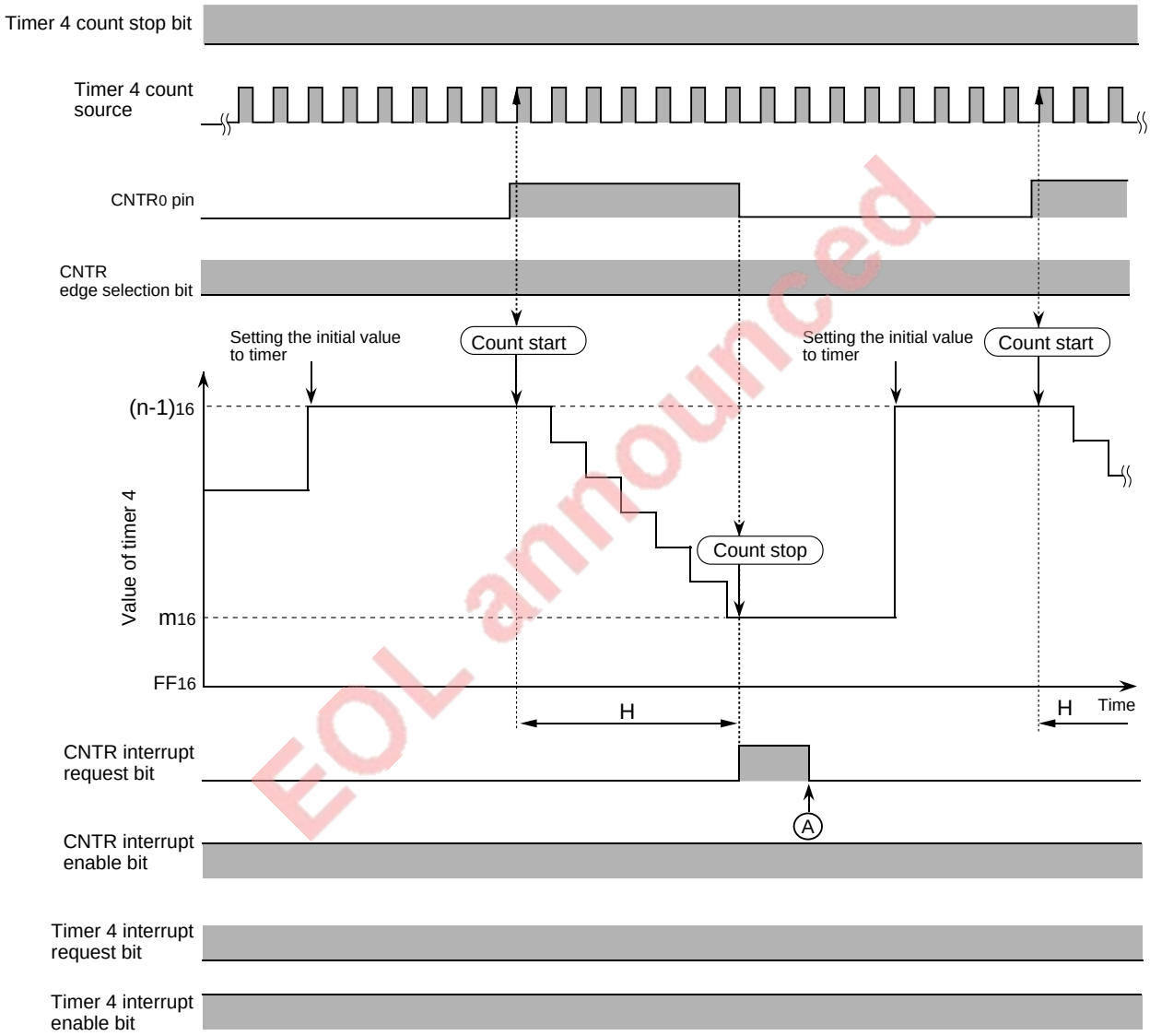
Figure 1.12.6 shows an example of the operation of the external pulse width measurement mode.

Pulse width

Pulse width H(s) = 1 ÷ count source frequency × (the timer initial value - count value at count stop)

External pulse width measurement mode operation example

- n: Timer initial value
- m: Count value at count stop



Ⓐ : • Clearing by writing "0" to the CNTR interrupt request bit.
• Clearing by accepting the CNTR interrupt request when the CNTR interrupt enable bit is "1."
✻ : When the CNTR edge selection bit is "0," "H" level width of the input pulse is measured.

Fig. 1.12.6 Example of operation of external pulse width measurement mode

1.12 Timers

[Setting method]

① The count operation of Timer 4 is stopped by setting bit 3 of the Timer 34 mode register to "1." Refer to **"Table 1.12.2 Setting for count stop."**

② Set each register according to the external pulse width measuring mode setting shown in Table 1.12.12.

③ Set a value in the timer.
Refer to **"Table 1.12.7 Address allocation of timer."**

④ Clear bit 3 of the Timer 34 mode register to "0" and start the count of timer 4.
Refer to **"Table 1.12.8 Count start setting."**

Table 1.12.12 External pulse width measurement mode setting

Timer to be used	Measuring pulse	Timer 34 mode register (T34M: Address $00F8_{16}$)			Edge polarity selection register (EG: Address $00D4_{16}$)		
		b6	b5	b4	b4	b3	b2
Timer 4	CNTR0	1	b5 b4 0 0 : Timer 3 overflow signal 0 1 : f (X_{IN})/16 or f (X_{CIN})/16 1 0 : Timer 1 or Timer 2 overflow signal 1 1 : CNTR1 pin (Note 2)		0	–	Select (Note 1)
	CNTR1				1	Select (Note 1)	–

Notes 1: 0: The count source is counted while the external pulse is "H."
1: The count source is counted while the external pulse is "L."

2: When the measured pulse is the $CNTR_1$ pin, do not select the $CNTR_1$ as the count source of Timer 4.

(5) PWM mode

In the PWM mode, a PWM waveform is output from the T_1 pin by using Timer 3 and Timer 4. The operations in the PWM mode are described below.

① Start of count operation

After the count stop bit is cleared to "0," a count operation starts. Each time a count source is input, the contents of the timer are decremented by 1.

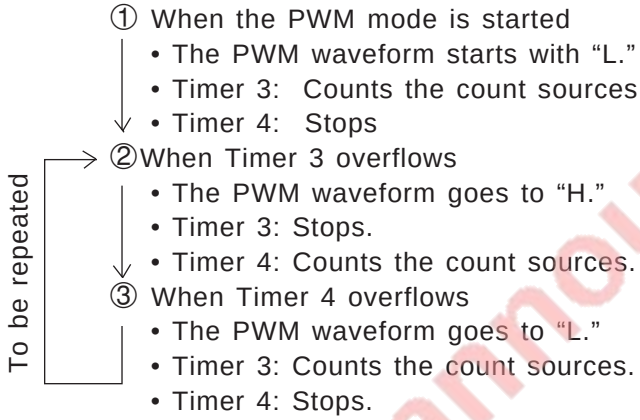
Note: The count stop bit is "0" immediately after reset release. Accordingly, a count operation is automatically started immediately after reset release but no PWM waveform is output because the PWM mode is not provided.

② Reload operation

When the timer overflows, the value resulting from decrementing 1 from the contents of the timer latch is transferred (reloaded) to the timer.

③ PWM output

In the PWM mode, the following operations are performed.



The "L" width of the PWM waveform is set in Timer 3 and the "H" width is set in Timer 4.

④ Interrupt operation

■Timer interrupt

When the timer overflows, an interrupt request occurs, so that the interrupt request bit is set to "1." The acceptance of interrupt is controlled by the interrupt enable bit of each timer.

⑤ Stop of count operation

When the counter stop bit is set to "1" by software, the count operation stops. (The count operation continues until the count stop bit is set to "1.")

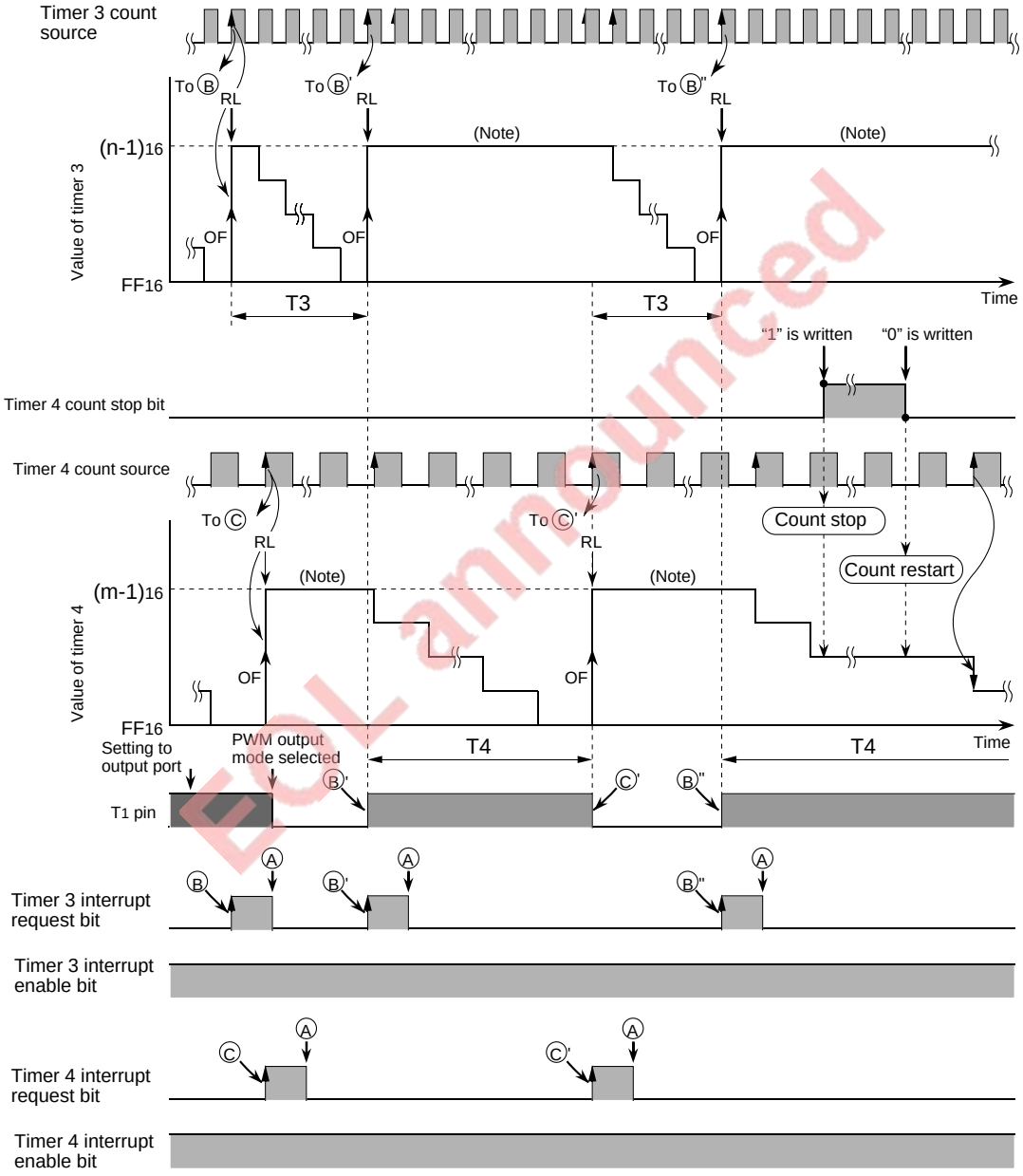
Figure 1.12.7 shows an example of the operation of the PWM output mode.

Count period

Timer 3 count period T3(s) = 1 ÷ timer 3 count source frequency × (the timer 3 initial value + 1)
Timer 4 count period T4(s) = 1 ÷ timer 4 count source frequency × (the timer 4 initial value + 1)

PWM mode operation example

- OF: Overflow
- RL: Reload
- n: Timer 3 initial value
- m: Timer 4 initial value



Ⓐ : • Clearing by writing "0" to the Timer 3 and Timer 4 interrupt request bits.
• Clearing by accepting the interrupt request when the Timer 3 and Timer 4 interrupt enable bits are "1."

Note: Timer 3 and Timer 4 do not accept count sources in the period from an overflow of the respective timer till an overflow of the other timer. Because the timer read value changes at the fall of a count source, the read value in this period remains "FF16".

Fig.1.12.7 Example of operation of PWM output mode

[Setting method]

① The count operation of a timer to be used is stopped.
Refer to **"Table 1.12.2 Setting for count stop."**

② Port $P1_3$ is put into the output mode by setting bit 3 of the port P1 direction register (address $00C1_{16}$) to "1."

③ Select a count source of Timer 3 and Timer 4. However, don't select the Timer 3 overflow signal as a count source of Timer 4.
Refer to **"Table 1.12.5 Setting for Timer 3 count source"** and **"Table 1.12.6 Setting for Timer 4 count source."**

④ Set the bit 7 of the Timer 34 mode register to "1."

⑤ Set a value in the timer.
Refer to **"Table 1.12.7 Address allocation of timer."**

⑥ Set the bit 7 of the Timer mode register 2 to "1."

⑦ The count operation of a timer to be used is started.
Refer to **"Table 1.12.8 Count start setting."**

Note: When the PWM mode is started from another mode, the PWM waveform starts with the "L" state.

1.12.3 Input latch function

There is a function which latches the levels of the INT_0, INT_1, $CNTR_0$ and $CNTR_1$ pins to the input latch register when Timer 4 overflows. Using this function permits knowing the level of each pin accurately the moment when a Timer 4 overflow occurs. The polarity of each pin is selected by the edge polarity selection register, the level or the reverse level of each pin are latched to the Input latch register.

Table 1.12.13 shows the Edge polarity selection register setting related to the input latch.

Table 1.12.13 Edge polarity selection register setting

Input latch register (ILR: Address $00D6_{16}$)	Latched contents	Edge polarity selection register (EG: Address $00D4_{16}$)			
		b3	b2	b1	b0
b0	INT_0 pin level	–	–	–	1
	Reverse level on INT_0 pin				0
b1	INT_1 pin level	–	–	1	–
	Reverse level on INT_1 pin			0	
b2	$CNTR_0$ pin level	–	1	–	–
	Reverse level on $CNTR_0$ pin		0		
b3	$CNTR_1$ pin level	1	–	–	–
	Reverse level on $CNTR_1$ pin	0			

1.12 Timers

1.12.4 Updating of contents of Timer and Timer latch

After data is written to the Timer, the contents of the Timer and the Timer latch are updated.

■ Timer 1 and Timer 2

When data is written to the Timer, this data is set in the Timer and the Timer latch at the same time. As a result, after data is written to the Timer which is in count operation, the count period becomes invalid.

Figure 1.12.8 shows an example of updating of the contents of Timer 1, Time 2 and Timer latch.

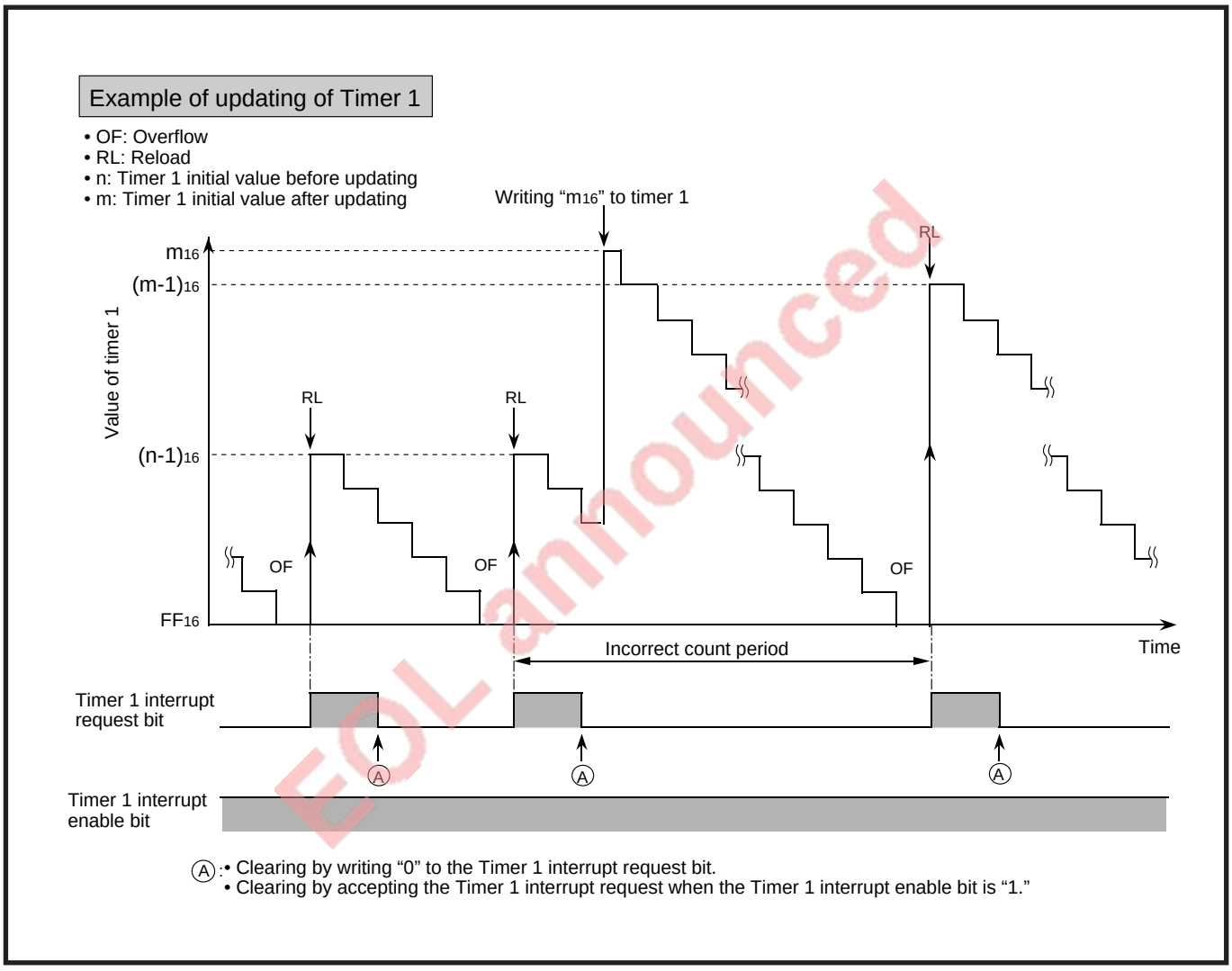


Fig. 1.12.8 Example of updating of Timer 1, Timer 2 and Timer latch

■ Timer 3 and Timer 4

● At PWM mode

After data is written to the Timer which is in count operation, the written data is set only in the Timer latch but not in the Timer. After that, when the Timer overflows, a value resulting from decrementing 1 from the value of the Timer latch is written in the Timer.
When data is written to the Timer being at a stop, the written data is set in both Timer and Timer latch.

● In modes other than the PWM mode

The same operation as **"Timer 1 and Timer 2"** described in the previous item is performed.

Figure 1.12.9 shows an example of updating of the contents of Timer 3, Time 4 and Timer latch in PWM mode.

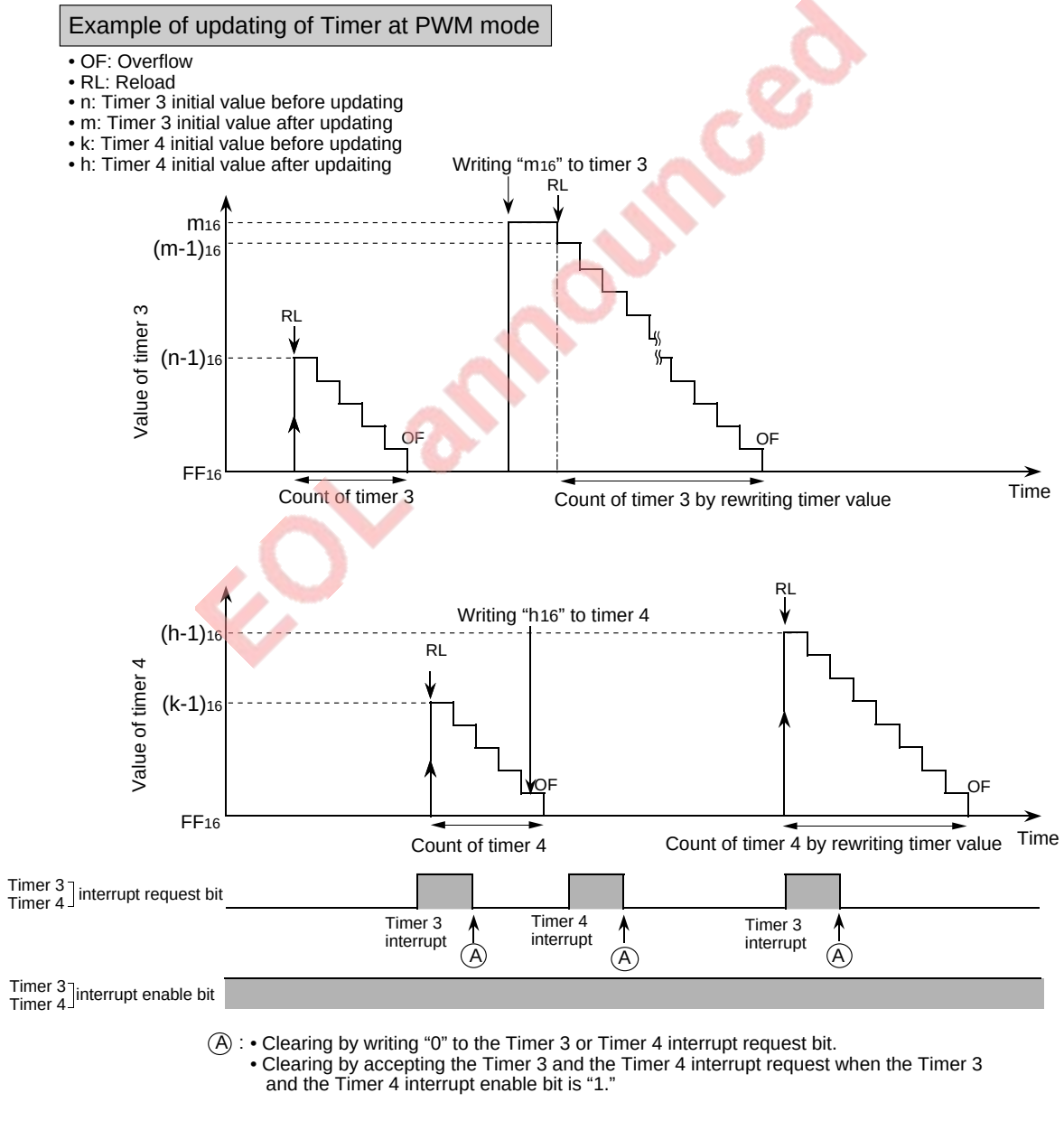


Fig. 1.12.9 Example of updating of Timer 3, Timer 4 and Timer latch in PWM mode

1.12 Timers

1.12.5 Notes on use

(1) The contents of the Timer 12 mode register (T12M: Address $00F8_{16}$) and the Timer 34 mode register (T34M: Address $00F9_{16}$) become "00_{16}" at reset and each timer performs a count operation.

(2) Figure 1.12.10 shows the relation between Timer value change timing and read value change timing. The Timer value changes at the rise of the count source, while the read value changes at the fall of the count source. Accordingly, the read value may be larger than the real timer value by "1."

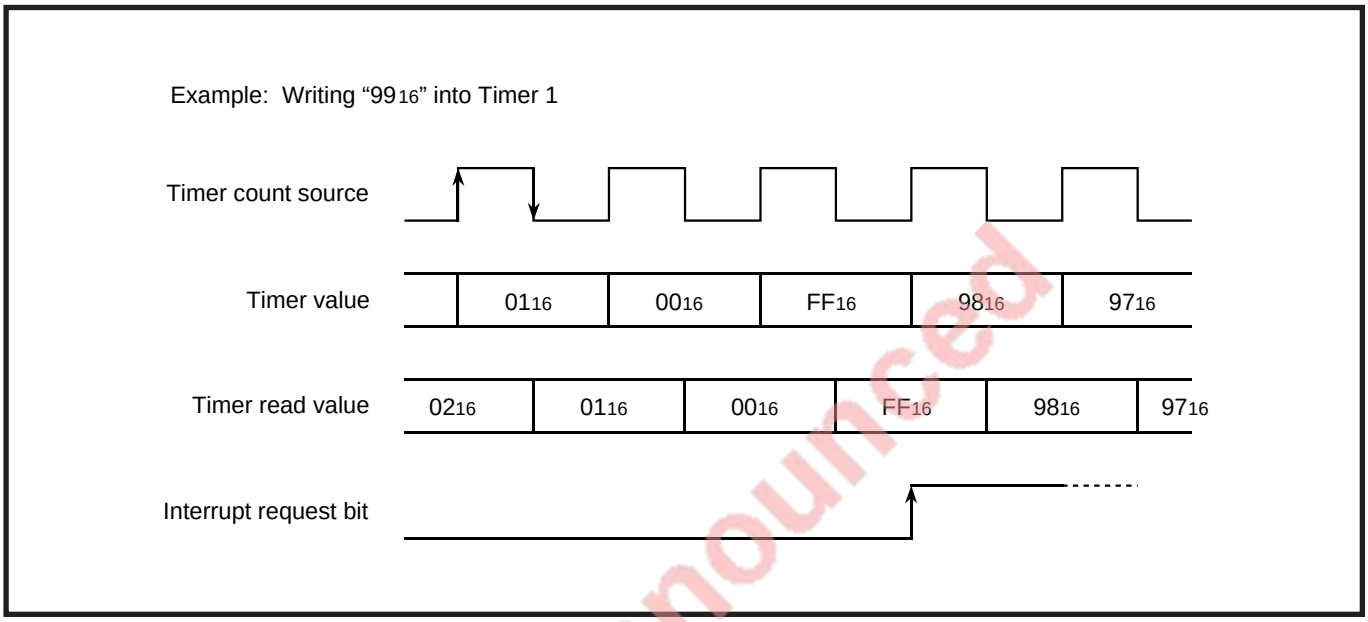


Fig. 1.12.10 Relation between timer value change timing and read value change timing

(3) To select the CNTR pin input as Timer count source, the frequency of the CNTR count source should satisfy the condition shown in Table 1.12.14.

Table 1.12.14 Frequency of CNTR

Main clock	Frequency of the CNTR count source	
	Upper boundary	Lower boundary
4MHz	1MHz	There is no special restriction.
8MHz	2MHz	

1.12.6 Related registers

(1) Timer 1, Timer 2, Timer 3, Timer 4 (T1 to T4: Address $00F0_{16}$ to $00F3_{16}$)

Each Timer is a register consisting of 8 bits.

Read

The contents (count value) of the Timer are read by reading the timer.

Write

When data is written to the Timer, this data is set in the Timer and the Timer latch at the same time.

When data is written into the Timer being in count operation in the PWM mode, the written data is set only in the Timer latch.

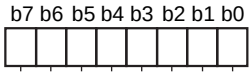
Refer to **"1.12.4 Updating of contents of Timer and Timer latch."**

✻Timer latch

The Timer latch is a register that holds a value to be automatically transferred (reloaded) to the Timer as its initial value when the Timer overflows. It is impossible to read the contents of the Timer latch.

Figure 1.12.11 shows a structure of Timer.

Timers 1 to 4



Timer 1, Timer 2, Timer 3, Timer 4 (T1, T2, T3, T4)
[Address $00F0_{16}$, $00F1_{16}$, $00F2_{16}$, $00F3_{16}$]

B	Function	At reset	R	W
0 to 7	•Set "00_{16} to FF_{16}." •The value is decremented by 1 each time a count source is input. •Each Timer values are set to the respective counter. •The count values are read out by reading the respective timer.	(Note)	○	○

Note : Timers 1 and 2 are undefined.
Timer 3 is "FF_{16}."
Timer 4 is "07_{16}."

Fig. 1.12.11 Structure of Timers 1 to 4

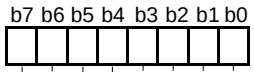
1.12 Timers

(2) Timer 12 mode register (T12M: Address 00F8$_{16}$)

The Timer 12 mode register is a register consisting of bits that control a Timer count source and a count operation.

Figure 1.12.12 shows a structure of Timer 12 mode register.

Timer 12 mode register



Timer 12 mode register (T12M) [Address 00F8$_{16}$]

B	Name	Function	At reset	R	W
0	Timer 1 count stop bit	0 : Count start 1 : Count stop	0	○	○
1	Timer 1 count source selection bit	0 : Internal clock (Note 1) 1 : P3$_2$/CNTR$_0$ external clock	0	○	○
2	Timer 1 internal clock source selection bit	0 : f(X$_{IN}$)/16 or f(X$_{CIN}$)/16 1 : f(X$_{CIN}$) (Note 2)	0	○	○
3	P1$_2$/T$_0$ port output selection bit	0 : P1$_2$ port output 1 : T$_0$(Timer 1 overflow divided by 2)	0	○	○
4	Timer 2 count stop bit	0 : Count start 1 : Count stop	0	○	○
5	Timer 2 count source selection bit	0 : Internal clock (Note 1) 1 : Timer 1 overflow signal	0	○	○
6, 7	Timer 2 internal clock source selection bits	b7 b6 0 0 : f(X$_{IN}$)/16 or f(X$_{CIN}$)/16 0 1 : f(X$_{IN}$)/64 or f(X$_{CIN}$)/64 1 0 : f(X$_{IN}$)/128 or f(X$_{CIN}$)/128 1 1 : f(X$_{IN}$)/256 or f(X$_{CIN}$)/256 (Note 3)	0	○	○

Notes 1: In the 7470/7477 group, the internal clock is f(X$_{IN}$)/16.
2: Since the 7470/7477 group is not provided the sub-clock generating circuit, f(X$_{CIN}$) cannot be used. Fix this bit to "0."
3: Since the 7470/7477 group is not provided the sub-clock generating circuit, f(X$_{CIN}$) cannot be used.

Fig. 1.12.12 Structure of Timer 12 mode register

(3) Timer 34 mode register (T34M: Address 00F9$_{16}$)

The Timer 34 mode register is a register consisting of bits that control a timer count source and a count operation.

Figure 1.12.13 shows a structure of Timer 34 mode register.

Timer 34 mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer 34 mode register (T34M) [Address 00F9$_{16}$]

B	Name	Function	At reset	R	W
0	Timer 3 count stop bit	0 : Count start 1 : Count stop	0	○	○
1, 2	Timer 3 count source selection bits	b2 b1 0 0 : f(XIN)/16 or f(XCIN)/16 0 1 : f(XCIN) 1 0 : Timer 1 overflow or Timer 2 overflow 1 1 : P33/CNTR1 external clock (Note 2)	0	○	○
3	Timer 4 count stop bit	0 : Count start 1 : Count stop	0	○	○
4, 5	Timer 4 count source selection bits	b4 b3 0 0 : Timer 3 overflow 0 1 : f(XIN)/16 or f(XCIN)/16 1 0 : Timer 1 overflow or Timer 2 overflow 1 1 : P33/CNTR1 external clock (Notes 1, 2)	0	○	○
6	Timer 4 pulse width measurement mode selection bit	0 : Timer mode 1 : External pulse width measurement mode	0	○	○
7	P13/T1 port output selection bit	0 : P13 port 1 : T1(Timer 4 overflow divided by 2 or PWM output)	0	○	○

Notes 1: When Timer 1 overflow is selected as a Timer 2 count source, the Timer 4 count source is the Timer 1 overflow regardless of the value of bit 6 of the Timer mode register 2.

2: Since the 7470/7477 group is not provided the sub-clock generating circuit, f(XCIN) cannot be used.

Fig. 1.12.13 Structure of Timer 34 mode register

(4) Timer mode register 2 (TM2: Address 00FA16)

The Timer mode register 2 consists of bits that control a mode selection and a count source selection.

Figure 1.12.14 shows a structure of Timer mode register 2.

Timer mode register 2

b7 b6 b5 b4 b3 b2 b1 b0

Timer mode register 2 (TM2) [Address 00FA16]

B	Name	Function	At reset	R	W
0	Timer 1 overflow FF set enable bit	0 : Set disable 1 : Set enable	0	○	○
1	Timer 4 overflow FF set enable bit	0 : Set disable 1 : Set enable	0	○	○
2 to 5	Nothing is allocated for these bits. These are write disabled bits and are undefined at reading.		?	?	×
6	Timer 3, timer 4 count overflow signal selection bit	0 : Timer 1 overflow 1 : Timer 2 overflow	0	○	○
7	Timer 3, timer 4 function selection bit	0 : Ordinary mode 1 : PWM mode	0	○	○

Fig. 1.12.14 Structure of Timer mode register 2

(5) Timer FF register (TF: Address 00F716)

The Timer FF register consists of bits that are used for initialization in the pulse output mode.

Figure 1.12.15 shows a structure of Timer FF register.

Timer FF register

b7 b6 b5 b4 b3 b2 b1 b0

Timer FF register (TF) [Address 00F716]

B	Name	Function	At reset	R	W
0	Timer 1 division flip-flop	0 : Initial value is “0” 1 : Initial value is “1”	0	○	○
1	Timer 4 division flip-flop	0 : Initial value is “0” 1 : Initial value is “1”	0	○	○
2 to 7	Nothing is allocated for these bits. These are write disabled bits and are undefined at reading.		?	?	×

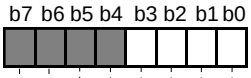
Fig. 1.12.15 Structure of Timer FF register

(6) Input latch register (ILR: Address 00D6$_{16}$)

The Input latch register consists of bits that latch the levels of the INT$_0$, INT$_1$, CNTR$_0$ and CNTR$_1$ pins when the Timer 4 overflows.

Figure 1.12.16 shows a structure of Input latch register.

Input latch register



Input latch register (ILR) [Address 00D6$_{16}$]

B	Name	Function	At reset	R	W
0	P3$_0$/INT$_0$ latch bit	When b0 of EG (Note) is "0": reverse level on INT$_0$ pin When b0 of EG (Note) is "1": level on INT$_0$ pin	?	○	×
1	P3$_1$/INT$_1$ latch bit	When b1 of EG (Note) is "0": reverse level on INT$_1$ pin When b1 of EG (Note) is "1": level on INT$_1$ pin	?	○	×
2	P3$_2$/CNTR$_0$ latch bit	When b2 of EG (Note) is "0": reverse level on CNTR$_0$ pin When b2 of EG (Note) is "1": level on CNTR$_0$ pin	?	○	×
3	P3$_3$/CNTR$_1$ latch bit	When b3 of EG (Note) is "0": reverse level on CNTR$_1$ pin When b3 of EG (Note) is "1": level on CNTR$_1$ pin	?	○	×
4 to 7	Nothing is allocated for these bits. These are write disabled bits and are undefined at reading.		?	?	×

Note: EG is the Edge polarity selection register.

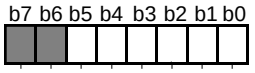
Fig. 1.12.16 Structure of Input latch register

(7) Edge polarity selection register (EG: Address 00D4$_{16}$)

The Edge polarity selection register consists of bits that control a polarity selection of each of the INT_0, INT_1, $CNTR_0$ and $CNTR_1$ pins and an INT_1 interrupt or key on wake-up interrupt selection at stop mode or wait mode.

Figure 1.12.17 shows a structure of Edge polarity selection register.

Edge polarity selection register



Edge polarity selection register (EG) [Address 00D4$_{16}$]

B	Name	Function	At reset	R	W
0	INT_0 edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
1	INT_1 edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
2	$CNTR_0$ edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
3	$CNTR_1$ edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
4	$CNTR_0$/$CNTR_1$ interrupt selection bit	0 : $CNTR_0$ 1 : $CNTR_1$	0	○	○
5	INT_1 source selection bit (at STP or WIT instruction execution)	0 : $P3_1$/INT_1 1 : $P0_0$ to $P0_7$ "L" level input (for key-on wake-up)	0	○	○
6, 7	Nothing is allocated for these bits. These are write disabled bits and are undefined at reading.		?	?	×

Fig. 1.12.17 Structure of Edge polarity selection register

1.13 Serial I/O

The 7470/7471/7477/7478 group can transmit or receive 8-bit data in series by Serial I/O.
The Serial I/O transmit/receive method is shown below.

- In the 7470/7471 group, only the clock synchronous is available.
- In the 7477/7478 group, either clock synchronous or clock asynchronous (UART) can be selected.

There are differences in circuit configuration and applicable registers between them.

This section explains each of them as **"1.13A 7470/7471 group part"** and **"1.13B 7477/7478 group part."** Table 1.13.1 shows defferences between 7470/7471 group and 7477/7478 group.

Table 1.13.1 7470/7471 group vs. 7477/7478 group serial I/O

		7470/7471 group	7477/7478 group
Serial I/O transmit/ receive method	Clock synchronous	○	○
	Clock asynchronous	×	○
$\overline{\text{SRDY}}$ signal output*1		○	○
SARDY signal output*1		○	×
Byte specification mode*2		○	×

∗1 $\overline{\text{SRDY}}$ and SARDY signal : Signal that indicates a Serial I/O transfer ready state
∗2 Byte specification mode : Mode for transmitting or receiving 1-byte data of a specific cycle out of multiple-byte data to be transmitted or received.

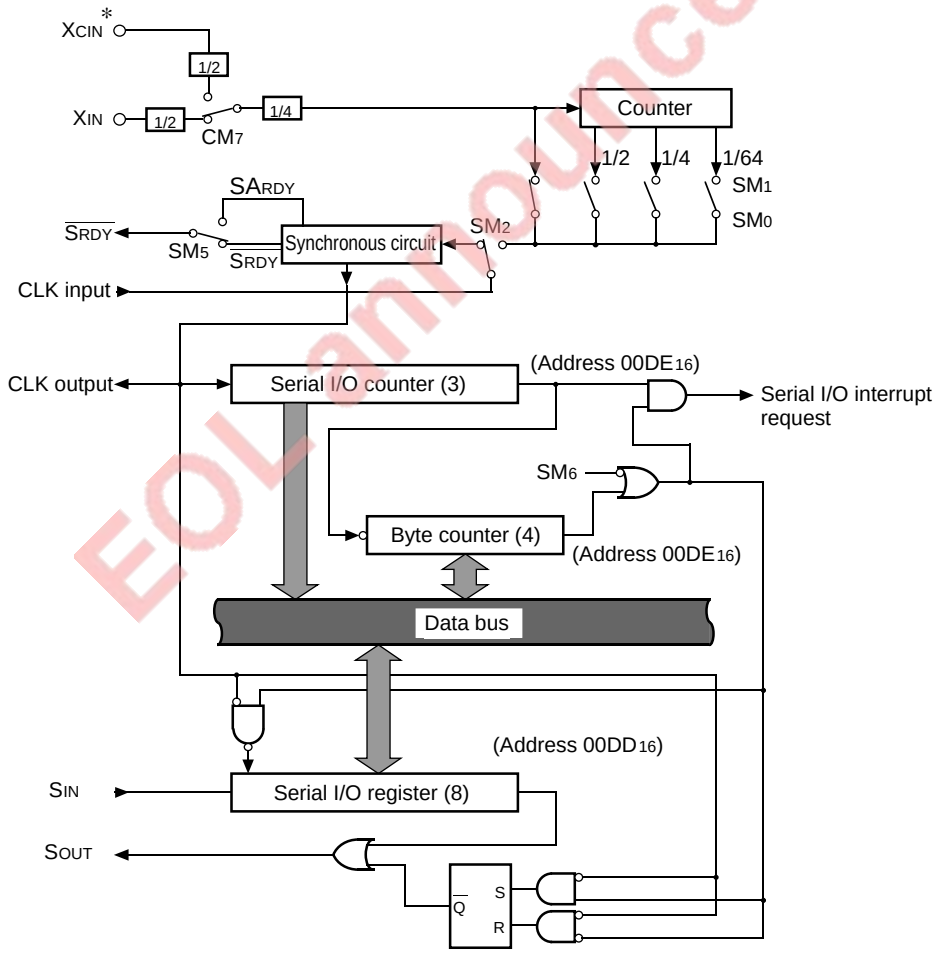
1.13A 7470/7471 group part

1.13A.1 Operation description

The 7470/7471 group incorporates a clock synchronous Serial I/O.
The 8 shift clocks obtained by the clock control circuit are used as synchronous clocks for transmitting or receiving data. The transmit operation of the transmit side and the receive operation of the receive side are simultaneously executed in synchronization with these shift clocks.

- The transmit side transmits data 1 bit by 1 bit from the P15/SOUT pin in synchronization with the fall of each shift clock.
- The receive side receives data 1 bit by 1 bit from the P14/SIN pin in synchronization with the rise of each shift clock.

Figure 1.13A.1 shows a Serial I/O block diagram.



*: The 7470 group is not provided with the XCIN pin.

Fig. 1.13A.1 Serial I/O block diagram

Communication format

The half-duplex data communication or the full-duplex data communication are available.

Synchronous clock

The internal clock or the external clock can be selected as a synchronous clock by bit 2 of the Serial I/O mode register (SM: address 00DC16).

● The synchronous clock for the case where the internal clock is selected is shown below.

- f(XIN)/8
- f(XIN)/16
- f(XIN)/32
- f(XIN)/512

} When the system clock is f(XIN)

- f(XCIN)/8
- f(XCIN)/16
- f(XCIN)/32
- f(XCIN)/512

} When the system clock is f(XCIN)

Notes 1: In the 7470 group, f(XCIN) is not available.
2: When selecting a divided signal of f(XCIN), set the system clock to the low-speed mode by bit 7 of the CPU mode register.

● When the external clock is selected, the synchronous clock is an external clock input from the P16/CLK pin.

Notes on external clock selection

● When writing data into the Serial I/O register, perform a write operation while the synchronous clock is at "H."
● The shift operation of the Serial I/O register is continued while the synchronous clock is input to the Serial I/O circuit. When the external clock is selected, stop the synchronous clock at the end of 8 cycles. (When the internal clock is selected, the synchronous clock stops automatically.)
● Set the "H" and "L" widths (TWH, TWL) of the pulse used as the external clock source to TWH, TWL [s] ≧ 2/(system clock frequency [Hz]). For example, when the system clock is 8 MHz, use a clock of 2 MHz or less (duty ratio 50 %).

Shift clock

Usually, when a clock synchronous transfer is performed between 2 microcomputers, one microcomputer selects the internal clock and outputs the 8 shift clock pulses generated by a start of transfer operation from the P16/CLK pin. The other microcomputer selects the external clock and uses the clock input from the CLK pin as a synchronous clock.

$\overline{\text{SRDY}}$ signal, SARDY signal

A Serial I/O transfer ready status can be known to the outside by outputting the $\overline{\text{SRDY}}$ signal and the SARDY signal.

1.13 Serial I/O

■ Transmit operation of Serial I/O

The transmit operation of the Serial I/O is described below.

● Start of transmit operation

Transmit operation begins by writing transmit data into the Serial I/O register[*2] in the transmit enable state.[*1] At the time when this data has been written, "7" is set in the Serial I/O counter (address 00DE16, bit 4 – 6), so that the synchronous clock is forced to go to "H."

∗1: State in which the register for transmit operation has been initialized. Refer to **"[Transmit setting method]"** which will be described later.

∗2: When the external clock is selected, perform a write operation while the synchronous clock is at "H."

● Transmit operation

①The transmit data written in the Serial I/O register is output from the P15/SOUT pin in synchronization with the fall of the synchronous clock. At this time, the Serial I/O counter is decremented by 1.

②Transmit data is output starting with the least significant bit of the Serial I/O register. Each time one bit is output, the contents of the Serial I/O register are shifted by 1 in the direction of the least significant bit.

③After the transmit shift operation is completed, an interrupt request occurs at the rise of the last cycle of the synchronous clock, so that the Serial I/O interrupt request bit is set[*3] to "1."

∗3: When the internal clock is selected as a synchronous clock, the shift clock supply to the Serial I/O register is automatically stopped after 8-bit data is transmitted (the Serial I/O counter overflows). When the external clock is selected, the contents of the Serial I/O register are continuously sifted while the synchronous clock is input. Accordingly, stop it externally.

● When using the $\overline{\text{SRDY}}$ output

At the time when transmit data has been written, the level of the $\overline{\text{SRDY}}$ signal changes from "H" to "L" and the level of the SARDY signal changes from "L" to "H," by which a receive ready state can be known externally. The $\overline{\text{SRDY}}$ signal goes to "H" at the first fall of the synchronous clock and the SARDY signal goes to "L" at the rise of the last cycle of the synchronous clock.

Figure 1.13A.2 shows a transmit operation and Figure 1.13A.3 shows a transmit timing chart.

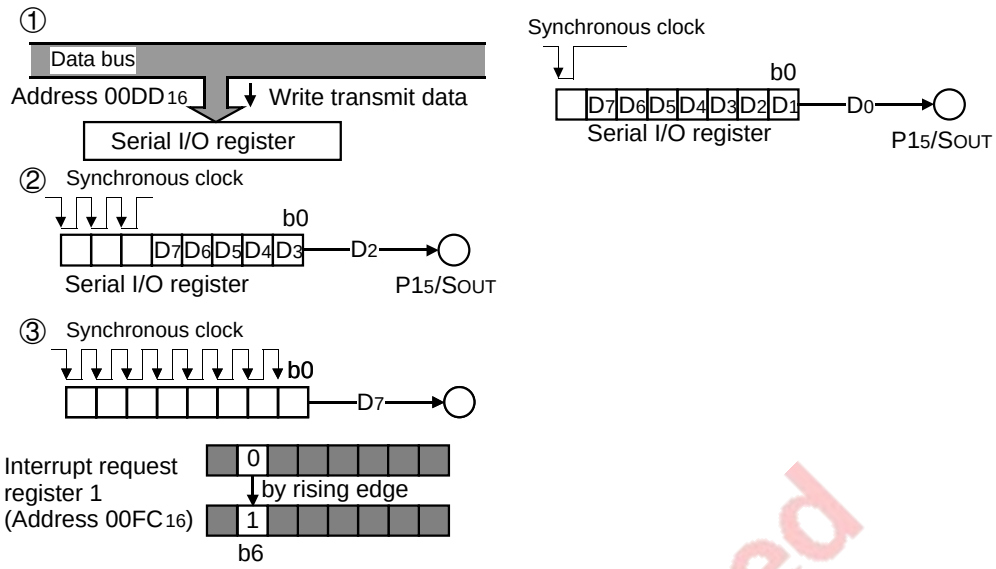


Fig. 1.13A.2 Serial I/O transmit operation

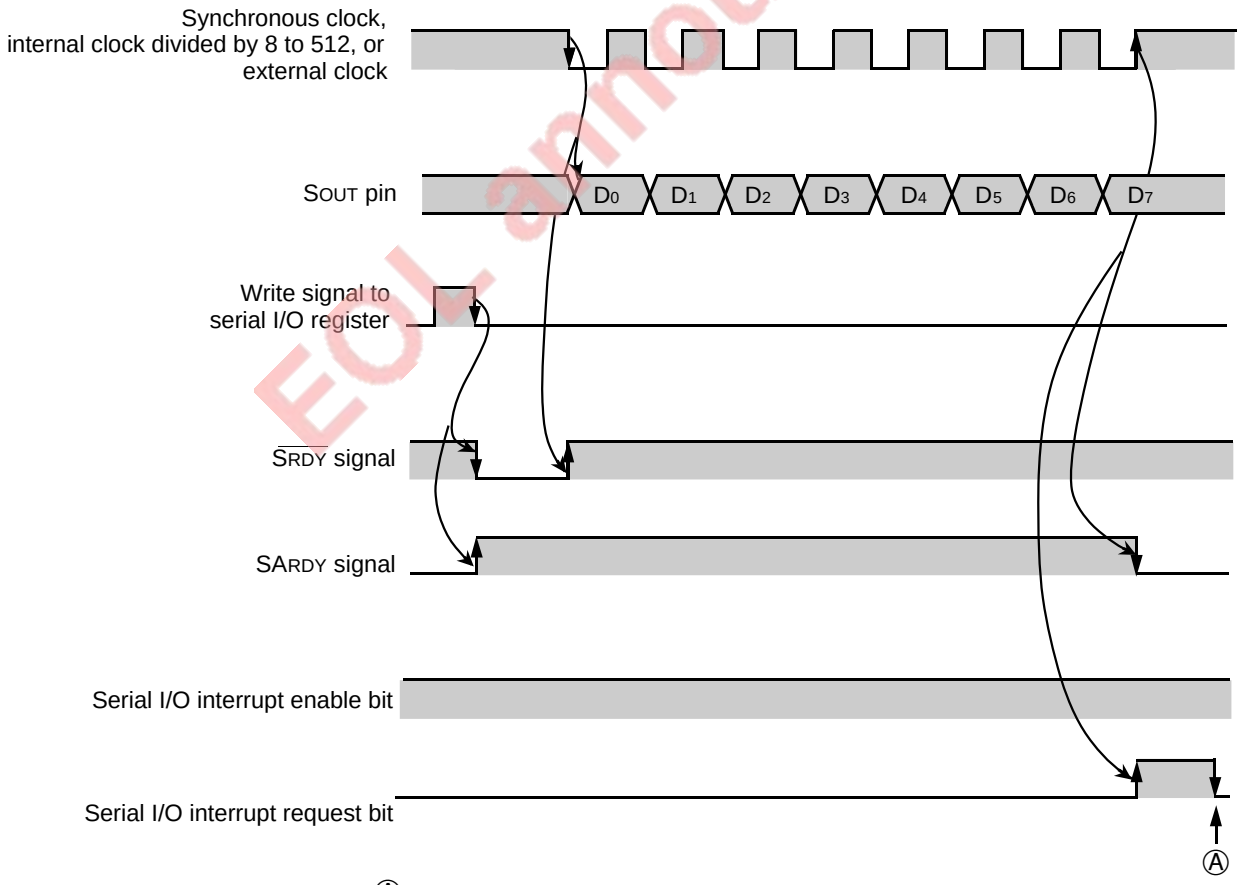


Fig. 1.13A.3 Serial I/O transmit timing chart

1.13 Serial I/O

[Transmit setting method]

①Clear the Serial I/O interrupt enable bit (bit 6 of the Interrupt control register 1) to "0."
②Set the Serial I/O mode register according to "**Table 1.13A.1.**"
③When using the Serial I/O interrupt,
[1] Clear the Serial I/O interrupt request bit (bit 6 of the Interrupt request register 1) to "0."
[2] Set the Serial I/O interrupt enable bit to "1."
④Write transmit data into the Serial I/O register (address $00DD_{16}$).

Note: When the external clock is selected, perform a write operation while the synchronous clock is at "H."

Table 1.13A.1 shows a Serial I/O transmit setting.

Table 1.13A.1 Serial I/O transmit setting

Item \ Register to be used		Serial I/O mode register (SM: Address $00DC_{16}$)	
		Bit	Setting value
Synchronous clock (at internal clock selection) (Note 1)	f(XIN)/8 f(XCIN)/8	b1 • b0	00
	f(XIN)/16 f(XCIN)/16		01
	f(XIN)/32 f(XCIN)/32		10
	f(XIN)/512 f(XCIN)/512		11
Synchronous clock selection	External clock	b2	0
	Internal clock		1
Serial I/O port using	Serial I/O port (SOUT, CLK) (Note 2)	b3	1
$\overline{\text{SRDY}}$ signal output selection	Ordinary port	b4	0
	$\overline{\text{SRDY}}$ signal output		1
$\overline{\text{SRDY}}$ signal selection	$\overline{\text{SRDY}}$ signal	b5	0
	SARDY signal		1
Serial I/O Byte specification mode selection	Ordinary mode	b6	0
	Byte specification mode		1
P15/SOUT, $\overline{\text{SRDY}}$ pin output format (Note 3)	CMOS output	b7	0
	N channel open drain output		1

Notes 1: Select the internal clock as a synchronous clock in the following condition. In the 7470 group, however, f(XCIN) is not available.
- When a divided signal of f(XIN) is selected, the system clock is f(XIN).
- When a divided signal of f(XCIN) is selected, the system clock is f(XCIN).

Select a system clock state by bit 7 of the CPU mode register.

2: When the ordinary port is switched over to the Serial I/O port, the Serial I/O interrupt request bit may be set to "1." Clear the Serial I/O interrupt request bit to "0" after one instruction or more after switching the ordinary port over to the Serial I/O port.

3: When ordinary P17 is selected by bit 4 of the Serial I/O mode register, the CMOS output is provided regardless of the set value of bit 7.

■ Receive operation of Serial I/O

The receive operation of the Serial I/O is described below.

● Start of receive operation

Receive operation begins by writing the following data into the Serial I/O register (SIO: address $00DD_{16}$)*2 in the receive enable state.*1

- Transmit data in the full-duplex data communication
- Arbitrary dummy data in the half-duplex data communication

At the time when this data has been written, "7" is set in the Serial I/O counter (address $00DE_{16}$, bit 4 – 6), so that the synchronous clock is forced to go to "H."

∗1: State in which the register for receive operation has been initialized. Refer to **"[Receive setting method]"** which will be described later.

∗2: When the external clock is selected, perform a write operation while the synchronous clock is at "H."

● Receive operation

①Receive data is input from the P14/SIN pin to the Serial I/O register in synchronization with the rise of the synchronous clock. At this time, the Serial I/O counter is decremented by 1.

②Receive data is input starting into the most significant bit of the Serial I/O register. Each time one bit is input, the contents of the Serial I/O register are shifted by 1 in the direction of the least significant bit.

③After the receive shift operation is completed, an interrupt request occurs at the rise of the last cycle of the synchronous clock, so that the Serial I/O interrupt request bit is set to "1".*3

∗3: When the internal clock is selected as a synchronous clock, the shift clock supply to the Serial I/O register is automatically stopped after 8-bit data is transmitted (the Serial I/O counter overflows). When the external clock is selected, the contents of the Serial I/O register are continuously sifted while the synchronous clock is input. Accordingly, stop it externally.

● When using the $\overline{\text{SRDY}}$ output

At the time when data has been written into the Serial I/O register, the level of the $\overline{\text{SRDY}}$ signal changes from "H" to "L" and the level of the SARDY signal changes from "L" to "H," by which a receive ready state can be known externally. The $\overline{\text{SRDY}}$ signal goes to "H" at the first fall of the synchronous clock and the SARDY signal goes to "L" at the rise of the last cycle of the synchronous clock.

Figure 1.13A.4 shows a receive operation and Figure 1.13A.5 shows a receive timing chart.

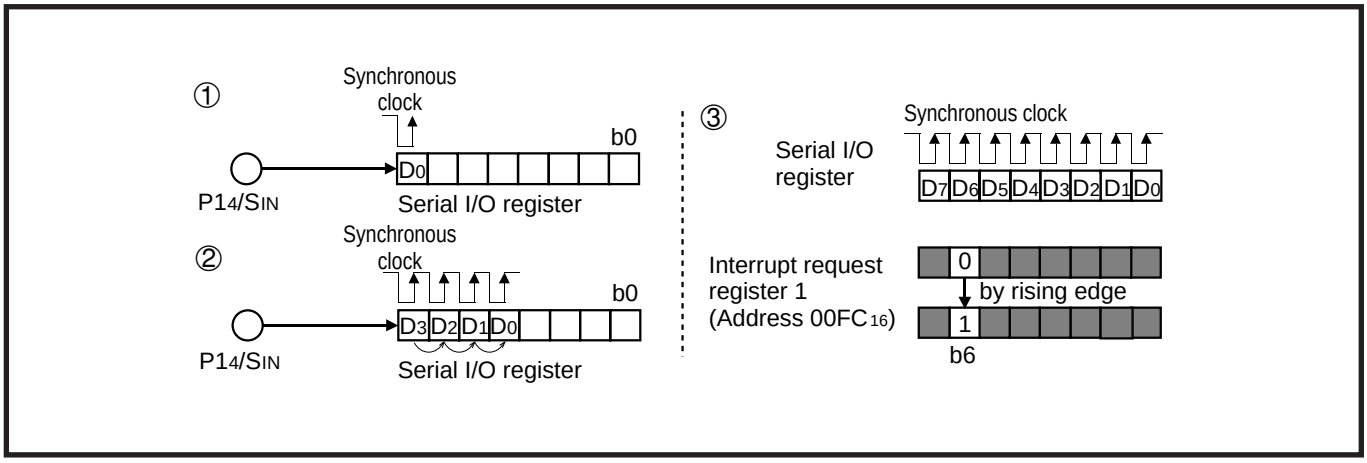


Fig. 1.13A.4 Serial I/O receive operation

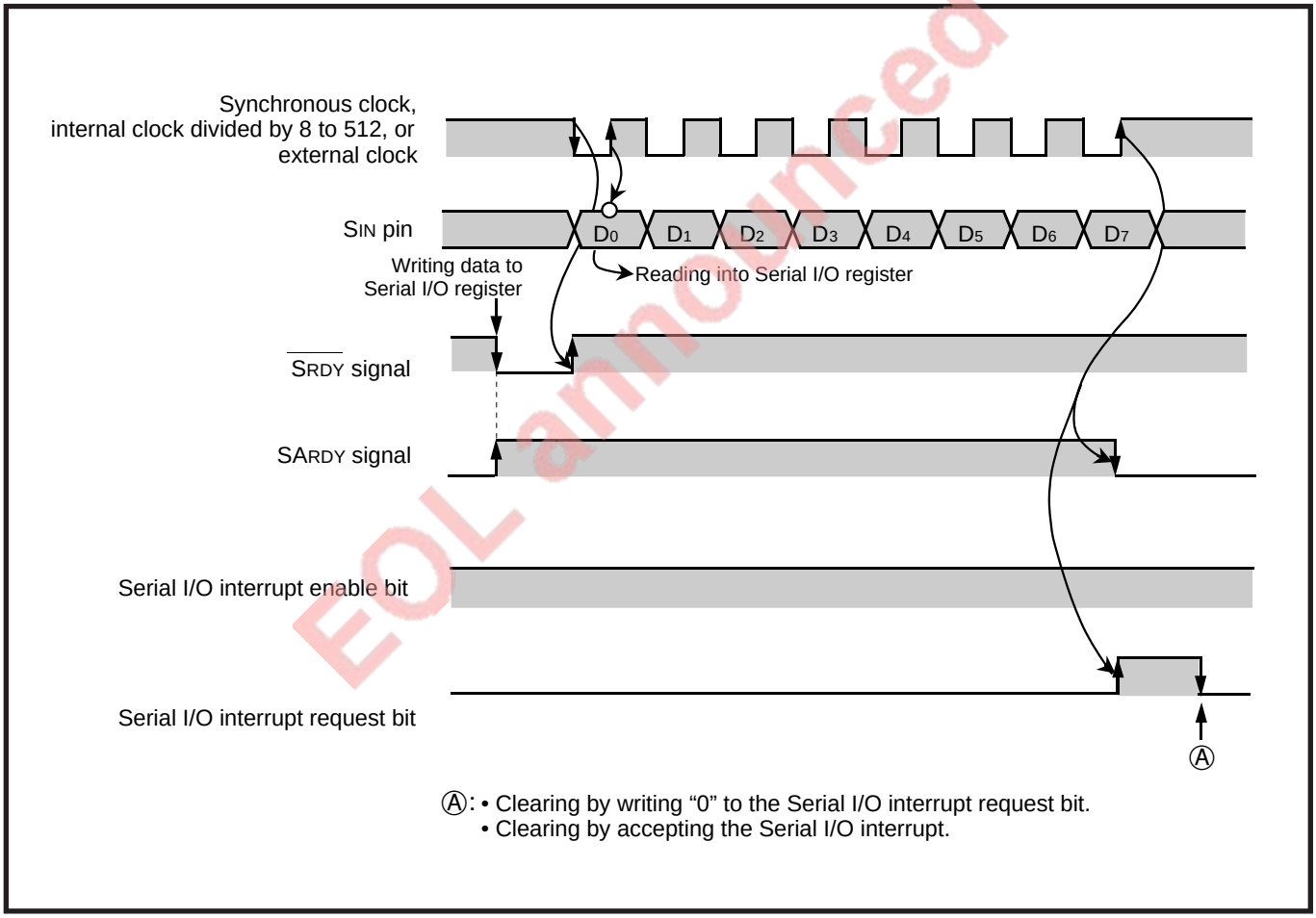


Fig. 1.13A.5 Serial I/O receive timing chart

[Receive setting method]

①Clear the Serial I/O interrupt enable bit (bit 6 of the Interrupt control register 1) to "0."
②Clear the Port P14 direction register to "0" to set it to the input mode.
③Clear the Serial I/O mode register according to "**Table 1.13A.2.**"
④When using the Serial I/O interrupt,
[1] Clear the Serial I/O interrupt request bit (bit 6 of the Interrupt request register 1) to "0."
[2] Set the Serial I/O interrupt enable bit to "1."
⑤Write the following data into the Serial I/O register.
- Transmit data in the full-duplex data communication
- Arbitrary dummy data in the half-duplex data communication

Note: When the external clock is selected, write data into the Serial I/O register while the synchronous clock is at "H."

Table 1.13A.2 shows a Serial I/O receive setting.

Table 1.13A.2 Serial I/O receive setting

Item \ Register to be used		Serial I/O mode register (SM: Address 00DC16)	
		Bit	Setting value
Synchronous clock (at internal clock selection) (Note 1)	f(XIN)/8 f(XCIN)/8	b1 • b0	00
	f(XIN)/16 f(XCIN)/16		01
	f(XIN)/32 f(XCIN)/32		10
	f(XIN)/512 f(XCIN)/512		11
Synchronous clock selection	External clock	b2	0
	Internal clock		1
Serial I/O port using	Ordinary port (P15, P16) (Note 2)	b3	0
	Serial I/O port (SOUT, CLK) (Note 3)		1
$\overline{SRDY}$ signal output selection	Ordinary port	b4	0
	$\overline{SRDY}$ signal output		1
$\overline{SRDY}$ signal selection	$\overline{SRDY}$ signal	b5	0
	SARDY signal		1
Serial I/O Byte specification mode selection	Ordinary mode	b6	0
	Byte specification mode		1
P15/SOUT, $\overline{SRDY}$ pin output format (Note 4, Note 5)	CMOS output	b7	0
	N channel open drain output		1

Notes 1: Select the internal clock as a synchronous clock in the following condition. In the 7470 group, however, f(XCIN) is not available.
- When a divided signal of f(XIN) is selected, the system clock is f(XIN).
- When a divided signal of f(XCIN) is selected, the system clock is f(XCIN).

Select a system clock state by bit 7 of the CPU mode register.

2: When the external clock is selected, the P16/CLK pin becomes clock input pin CLK regardless of the set value of bit 3 of the Serial I/O mode register. For this reason, only P15 is available as an ordinary port.

3: When the ordinary port is switched over to the Serial I/O port, the Serial I/O interrupt request bit may be set to "1." Clear the Serial I/O interrupt request bit to "0" after one instruction or more after switching the ordinary port over to the Serial I/O port.

4: When ordinary P17 is selected by bit 4 of the Serial I/O mode register, the CMOS output is provided regardless of the set value of bit 7.

5: When SOUT is selected by bit 3 of the Serial I/O mode register, the data written in the Serial I/O register is output from the SOUT pin in synchronization with the fall of the synchronous clock.

1.13A.2 Byte specification mode

The Serial I/O of the 7470/7471 group has the byte specification mode.
This mode permits transmitting or receiving specific one-byte data out of multiple-byte data transmitted or received through the Serial I/O bus.

■ **Byte counter (Address 00DE16)**

The Byte counter is located at the same address as that of the Serial I/O counter but the Serial I/O counter is a read-only type. So this counter is not affected by any write operation to the Byte counter. Because the Byte counter is not provided with a reload function, re-set a value to transmit or receive data continuously.

■ **SARDY**

When the SARDY signal is selected in the byte specification mode, the N channel open drain is selected as its output type, and the $\overline{\text{SRDY}}$ pins of multiple microcomputers are connected, the SARDY signal goes to "H" only when all the microcomputers have become ready for data transfer.

■ **Operations in the byte specification mode**

After setting the Serial I/O mode register, specify a byte corresponding to the clock to be used for a Serial I/O transmit/receive in the Byte counter. Where the value written in the Byte counter is n, a Serial I/O transmit/receive is performed by the clock of the (n + 1)-th byte.

● **Start of transmit/receive operation**

A transfer operation is started by writing the data (arbitrary dummy data in the half-duplex data communication)[*1] to be transmitted to the Serial I/O register.

∗1: When the external clock is selected, write data into the Serial I/O register when the synchronous clock is at "H." However, if the Byte counter value is a value other than "0," writing data is enabled even if the synchronous clock is at "L."

● **Transmit/receive operation**

①Each time the synchronous clock is input in 8 cycles, the Byte counter value is decremented by 1.
②With the synchronous clock of the next 8 cycles after the Byte counter value becomes "0," a Serial I/O transmit/receive is performed as in the ordinary mode.[*2] After completion of the 8-bit data output, an interrupt occurs at the rise of the last cycle of the synchronous clock, so that bit 6 of the Interrupt request register (address 00FC16) is set to "1."
③When the Byte counter overflows, the Serial I/O transfer stops.

∗2: When the Byte counter value is a value other than 0, the output of the SOUT pin goes to "H" at the fall of the first synchronous clock. If the N channel open drain is selected as the output type of the SOUT pin, the output is put into a high-impedance state, so the SOUT pin can be connected to the SOUT pin of another microcomputer.

Figure 1.13A.6 shows a transmit/receive operation in the byte specification mode.

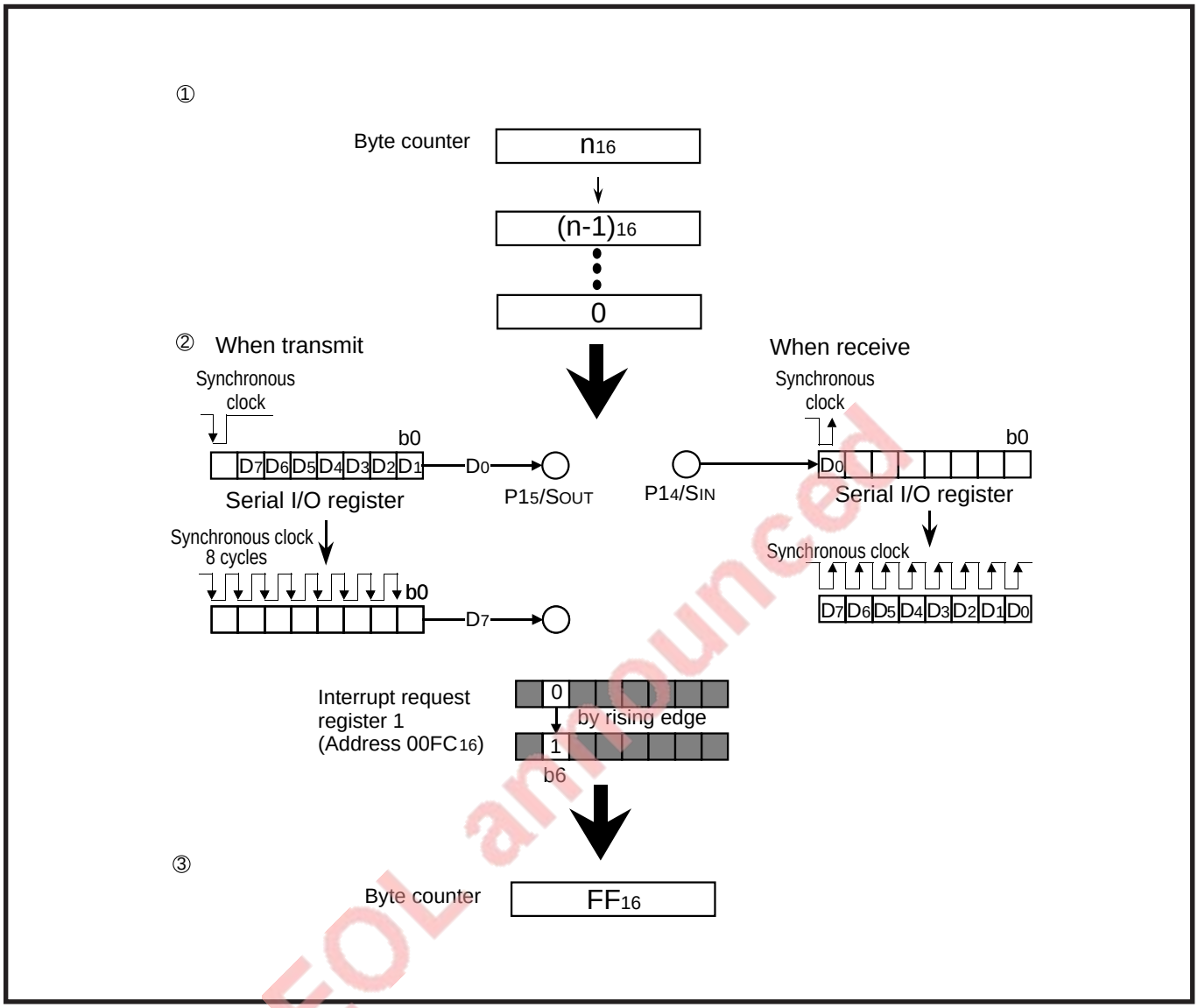


Fig. 1.13A.6 Transmit/receive operation in byte specification mode

[Byte specification mode setting]

For a Serial I/O transfer in the byte specification mode, refer to the setting method in the ordinary mode in **"1.13A.1 Operation description,"** and also take the following into consideration.

- Select the byte specification mode. (Set bit 6 of the Serial I/O register to "1.")
- Be sure to select the external clock as the synchronous clock. (Clear bit 2 of the Serial I/O mode register to "0.")
  - [1] When data is received, the ordinary port can be selected by the Serial I/O port selection bit (bit 3 of the Serial I/O mode register). $P1_6$ pin is used as a external clock input pin CLK. Only $P1_5$ is available as an ordinary port.
  - [2] Write data into the Serial I/O register when the synchronous clock is at "H." However, if the Byte counter value is a value other than "0," writing data is enabled even if the synchronous clock is at "L."
- When performing a Serial I/O transmit/receive at the n–th byte, write (n – 1) in the Byte counter.

Note: Because the Byte counter is not provided with a reload function, re-set a value to transmit or receive data continuously.

1.13A.3 Pins

The 7470/7471 group uses 4 pins for data transmit, data receive, shift clock transmit/receive and receive ready signal output. All these pins are used in common with P1. A function selection is made by the Serial I/O port selection bit (bit 3) and the $\overline{\text{SRDY}}$ signal output selection bit (bit 4) of the Serial I/O mode register (SM : Address $00DC_{16}$).
The function of each pin is explained below.

(1) Data transmit pin [SOUT]

Transmit data is output bit by bit. This pin is used in common with $P1_5$. When the Serial I/O port selection bit (bit 3) of the Serial I/O mode register is set to "1," this pin becomes a Serial I/O data output pin.

(2) Data receive pin [SIN]

Data is input bit by bit. This pin is used in common with $P1_4$. When the port $P1_4$ direction register is put into the input mode, this pin becomes a Serial I/O data input pin.

(3) Shift clock transmit/receive pin [CLK]

This pin inputs (receives from the outside) or outputs (supplies to the outside) the shift clock for data transmit/receive. This pin is used in common with $P1_6$.
The internal clock or the external clock can be selected by bit 2 of the Serial I/O mode register.

(4) Receive enable signal output pins [$\overline{\text{SRDY}}$], [SARDY]

This pin informs the outside of a receive ready state. This pin is used in common with $P1_7$.

● **$\overline{\text{SRDY}}$ signal**

- $\overline{\text{SRDY}}$ signal output selection bit (bit 4) of Serial I/O mode register is set to "1."
- $\overline{\text{SRDY}}$ signal selection bit (bit 5) of Serial I/O mode register is cleared to "0."

When the above 2 conditions are satisfied, the level of the pin changes from "H" to "L" at the timing at which data is written into the Serial I/O register, informing the outside of a receive ready state.

● **SARDY signal**

- $\overline{\text{SRDY}}$ signal output selection bit (bit 4) of Serial I/O mode register is set to "1."
- $\overline{\text{SRDY}}$ signal selection bit (bit 5) of Serial I/O mode register is set to "1."

When the above 2 conditions are satisfied, the level of the pin changes from "L" to "H" at the timing at which data is written into the Serial I/O register, informing the outside of a receive ready state.

1.13A.4 Notes on use

When the external clock is selected, take the following points into consideration.

①When writing data into the Serial I/O register, perform a write operation while the synchronous clock is at "H."
②The shift operation of the Serial I/O register is continued while the synchronous clock is input to the Serial I/O circuit. When the external clock is selected, stop the synchronous clock at the end of 8 cycles. (When the internal clock is selected, the synchronous clock stops automatically at the end of 8 cycles.)
③Set the "H" and "L" widths (T_{WH}, T_{WL}) of the pulse used as the external clock source to T_{WH}, T_{WL} [s] ≧ 2/(system clock frequency [Hz]). For example, when the system clock is 8 MHz, use a clock of 2 MHz or less (duty ratio 50 %).

1.13A.5 Related registers

(1) Serial I/O register (SIO: Address 00DD16)

The Serial I/O register is written Serial I/O transmit data or is read receive data.

- When transmitting data, write transmit data into this register.
- Receive data can be obtained by reading this register.

Figure 1.13A.7 shows a structure of the Serial I/O register.

Serial I/O register (7470/7471 group)

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O register (SIO) [Address 00DD16]

B	Function	At reset	R	W
0 to 7	At transmit: A value of "0016" to "FF16" can be set as transmit data. At the transmit, data is transmitted one bit at a time starting with the least significant bit. At receive: At the receive, data is received one bit at a time starting with the most significant bit.	?	○	○

Fig. 1.13A.7 Structure of Serial I/O register

(2) Serial I/O counter, Byte counter (Address 00DE16)

The Serial I/O counter and the Byte counter are located at the same address.

● **Serial I/O counter (bit 4 – bit 6)**

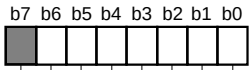
The Serial I/O counter is set to "7" by writing transmit data into the Serial I/O register and counts the synchronous clock of the Serial I/O eight times. The Serial I/O counter is a read-only type and not affected by any write operation to the Byte counter.

● **Byte counter (bit 0 – bit 3)**

In the Serial I/O byte specification mode, the value written in the Byte counter is counted down at 8 cycles of the synchronous clock. When the value becomes "0," a Serial I/O transmit/receive is performed by the synchronous clock of the next 8 cycles. Because a reload function is not available, re-set a value to transfer data continuously in the byte specification mode.

Figure 1.13A.8 shows a structure of the Serial I/O counter and the Byte counter.

Serial I/O counter and Byte counter (7470/7471 group)



Serial I/O counter and Byte counter [Address 00DE16]

B	Function	At reset	R	W
0 to 3	Byte counter When using the byte specification mode, set a value of "0016" to "0F16." Supposing that the value to be written into the byte counter is "n," a Serial transmit/receive is performed with the clock of the "n + 1"-th byte.	?	○	○
4 to 6	Serial I/O counter When the internal clock is selected as a synchronous clock, this counter generates 8 shift clocks. When transmit data is written into the Serial I/O register, "0716" is set in the Serial I/O counter.	?	○	×
7	Nothing is allocated for this bit. This is write disabled bit and is undefined at reading.	?	?	×

Fig. 1.13A.8 Structure of Serial I/O counter and Byte counter

(3) Serial I/O mode register (SM: Address 00DC16)

The Serial I/O mode register selects a state of the clock or port to be used for a data transfer.

Figure 1.13A.9 shows a structure of the Serial I/O mode register.

Serial I/O mode register (7470/7471 group)

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O mode register (SM) [Address 00DC16]

B	Name	Function	At reset	R	W
0, 1	Internal clock selection bits	b1 b0 0 0 : f(XIN)/8 or f(XCIN)/8 0 1 : f(XIN)/16 or f(XCIN)/16 1 0 : f(XIN)/32 or f(XCIN)/32 1 1 : f(XIN)/512 or f(XCIN)/512 (Note)	0	○	○
2	Synchronous clock selection bit	0 : External clock 1 : Internal clock	0	○	○
3	Serial I/O port selection bit	0 : Ordinary I/O port (P15, P16) 1 : Serial I/O port (SOUT, CLK pin)	0	○	○
4	$\overline{\text{SRDY}}$ signal output selection bit	0 : Ordinary I/O port(P17) 1 : $\overline{\text{SRDY}}$ signal output pin	0	○	○
5	$\overline{\text{SRDY}}$ signal selection bit	0 : $\overline{\text{SRDY}}$ signal 1 : SARDY signal	0	○	○
6	Serial I/O byte specify mode selection bit	0 : Ordinary mode 1 : Byte specify mode	0	○	○
7	P15/SOUT, $\overline{\text{SRDY}}$ output structure selection bit	0 : CMOS output 1 : N-channel open-drain output	0	○	○

Note: Since the 7470 group is not provided with the sub-clock generating circuit, do not select f(XCIN).

Fig. 1.13A.9 Structure of Serial I/O mode register

1.13B 7477/7478 group part

1.13B.1 Operation description

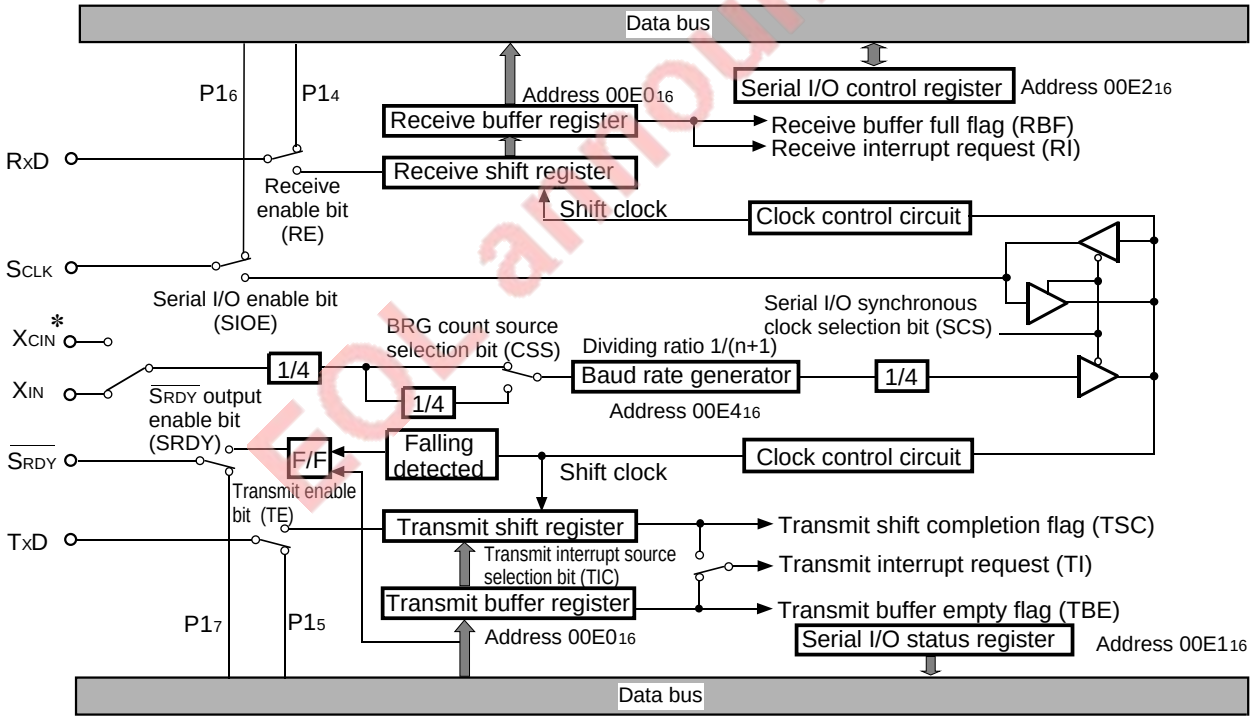
The 7477/7478 group incorporates a Serial I/O that permits selecting one of the clock synchronous and the clock asynchronous. This section describes the operation in each of the clock synchronous serial I/O and the clock asynchronous Serial I/O (UART).

(1) Clock synchronous Serial I/O

In the clock synchronous Serial I/O, the 8 shift clocks obtained by the clock control circuit are used as synchronous clocks for transmitting or receiving. The transmit operation of the transmit side and the receive operation of the receive side are simultaneously executed in synchronization with these shift clocks.

- The transmit side transmits data bit by bit from the $P1_5$/TxD pin in synchronization with the fall of each shift clock.
- The receive side receives data bit by bit from the $P1_4$/RxD pin in synchronization with the rise of each shift clock.

Figure 1.13B.1 shows a clock synchronous Serial I/O block diagram.



*: The 7477 group is not provided with the XCIN pin.

Fig. 1.13B.1 Clock synchronous Serial I/O block diagram

1.13 Serial I/O

■ Communication format

The half-duplex data communication or the full-duplex data communication are available for communication.

■ Synchronous clock

The following can be selected as a synchronous clock by bit 1 of the Serial I/O control register (SIOCON: address $00E2_{16}$).

- "0" : Baud rate generator (BRG) output divided by 4
- "1" : External clock input from the SCLK pin

The BRG output is set by the baud rate generator (BRG: address $00E4_{16}$), which is an 8-bit counter dedicated to the Serial I/O. As an input clock to the BRG, f(XIN)/4 (at "0"), f(XIN)/16 (at "1") can be selected by bit 0 of the Serial I/O control register. In the 7478 Group, f(XCIN)/4 (at "0"), f(XCIN)/16 (at "1") can be also selected.

Notes on external clock selection

- When setting the transmit enable bit to "1" or writing data into the Transmit buffer register, perform a write operation while the synchronous clock is at "H."
- The shift operation of the Transmit shift register or the Receive shift register is continued while the synchronous clock is input to the Serial I/O circuit. When the external clock is selected, stop the synchronous clock at the end of 8 cycles. (When the internal clock is selected, the synchronous clock stops automatically at the end of 8 cycles.)
- Set the "H" and "L" widths (TWH, TWL) of the pulse used as the external clock source to TWH, TWL [s] ≧ 8/(system clock [Hz]). For example, when a system clock is 8 MHz, use a clock of 500 kHz or less (duty ratio 50 %).

■ Shift clock

Usually, when a clock synchronous transfer is performed between 2 microcomputers, one microcomputer selects the internal clock and outputs the 8 shift clock pulses generated by a start of transmit operation from the P16/SCLK pin. The other microcomputer selects the external clock and uses the clock input from the P16/SCLK pin as a synchronous clock.

■ Data transfer rate (baud rate)

In the clock synchronous Serial I/O, the expression for calculating a data transfer rate (baud rate), which is the frequency of the synchronous clock is shown below.

● When the internal clock is selected (using the BRG)

$$\text{Baud rate [bps]} = \frac{f(X_{IN})^{*3}}{\text{Division ratio}^{*1} \times (\text{BRG set value}^{*2} + 1) \times 4}$$

∗1 Division ratio : Select "4" or "16" by the BRG count source selection bit.
∗2 BRG set value : 0 – 255 (00_{16} – FF_{16})
∗3 In the 7478 Group, f(XCIN) can be also used.

● When the external clock is selected

Baud rate [bps] = Input clock frequency to SCLK pin

The BRG is an 8-bit counter dedicated to the Serial I/O, having a reload register, and divides the count source by (n + 1) by setting the value n. As a count source, f(XIN)/4 (at "0"), f(XIN)/16 or (at "1") can be selected by bit 0 of the Serial I/O control register.
In the 7478 Group, f(XCIN)/4 (at "0"), f(XCIN)/16 (at "1") can be also selected.

■ $\overline{\text{SRDY}}$ signal

The clock synchronous Serial I/O can inform the outside that a serial transfer has become ready, by outputting the $\overline{\text{SRDY}}$ signal.

■ Transmit operation of the clock synchronous Serial I/O

The transmit operation of the clock synchronous Serial I/O is described below.

● Start of transmit operation

Transmit data is transmitted by writing it into the Transmit buffer register (TB: address $00E0_{16}$)[*2] in the transmit enable state.[*1] When the internal clock is selected as a synchronous clock, 8 shift clocks are generated at the time when this set value has been written.

● Transmit operation

①After transmit data is written into the Transmit buffer register,[*2] the transmit buffer empty flag (bit 0) of the Serial I/O status register is cleared to "0."

②The transmit data written in the Transmit buffer register is transferred to the Transmit shift register.[*3]

③When the data transfer from the Transmit buffer register to the Transmit shift register is completed, the transmit buffer empty flag is set to "1."[*4]

④The transmit data transferred to the Transmit shift register is output from the P15/TxD pin in synchronization with the fall of the synchronous clock.

⑤When a transmit shift operation is started, the transmit shift completion flag (b2) of the Serial I/O status register is cleared to "0."[*5]

⑥Data is output starting with the least significant bit of the Transmit shift register. Each time one-bit data is output, the contents of the Transmit shift register are shifted by 1 bit in the direction of the least significant bit.

⑦At the time when the transmit shift operation has been completed, the Transmit shift register shift completion flag is set to "1."[*3] [*5]

∗1: Status in which the register for transmit operation has been completed. Refer to **"[Clock synchronous Serial I/O setting method]"** which will be described later.

∗2: When the external clock is selected, write data into the Transmit buffer register when the synchronous clock is at "H."

∗3: A transmit interrupt request occurs immediately after the transfer of ② when the transmit interrupt source bit (bit 3) of the Serial I/O control register (SIOCON) is "0," or at the time of ⑦ when the said bit is "1."

∗4: While the transmit buffer empty flag is "1," the next transmit data can be written into the Transmit buffer register.

∗5: When the internal clock is used as a synchronous clock, the shift clock supply to the Transmit shift register is automatically stopped after 8-bit data is transmitted. However, if the next transmit data is written to the Transmit buffer register while the Transmit shift register shift completion flag is "0," the shift clock supply is continued and serial data is continuously output from the TxD pin.

1.13 Serial I/O

● **When using the $\overline{\text{SRDY}}$ output**

At the time when data has been written into the Transmit buffer register, the $\overline{\text{SRDY}}$ pin changes from "H" to "L," informing the outside of a receive ready state. The $\overline{\text{SRDY}}$ pin is restored to "H" at the first fall of the synchronous clock.

● **Transmit interrupt operation (valid when the Serial I/O is selected)**

Regarding a transmit interrupt, interrupt request generating timing can be selected by bit 3 of the Serial I/O control register (SIOCON).

0: When the Transmit buffer register becomes empty after the data written in the Transmit buffer register is transferred to the Transmit shift register, an interrupt request is generated.

1: When the shift operation of the Transmit shift register is completed, an interrupt request is generated.

Figure 1.13B.2 shows a transmit operation of clock synchronous Serial I/O and Figure 1.13B.3 shows a transmit timing chart of clock synchronous Serial I/O.

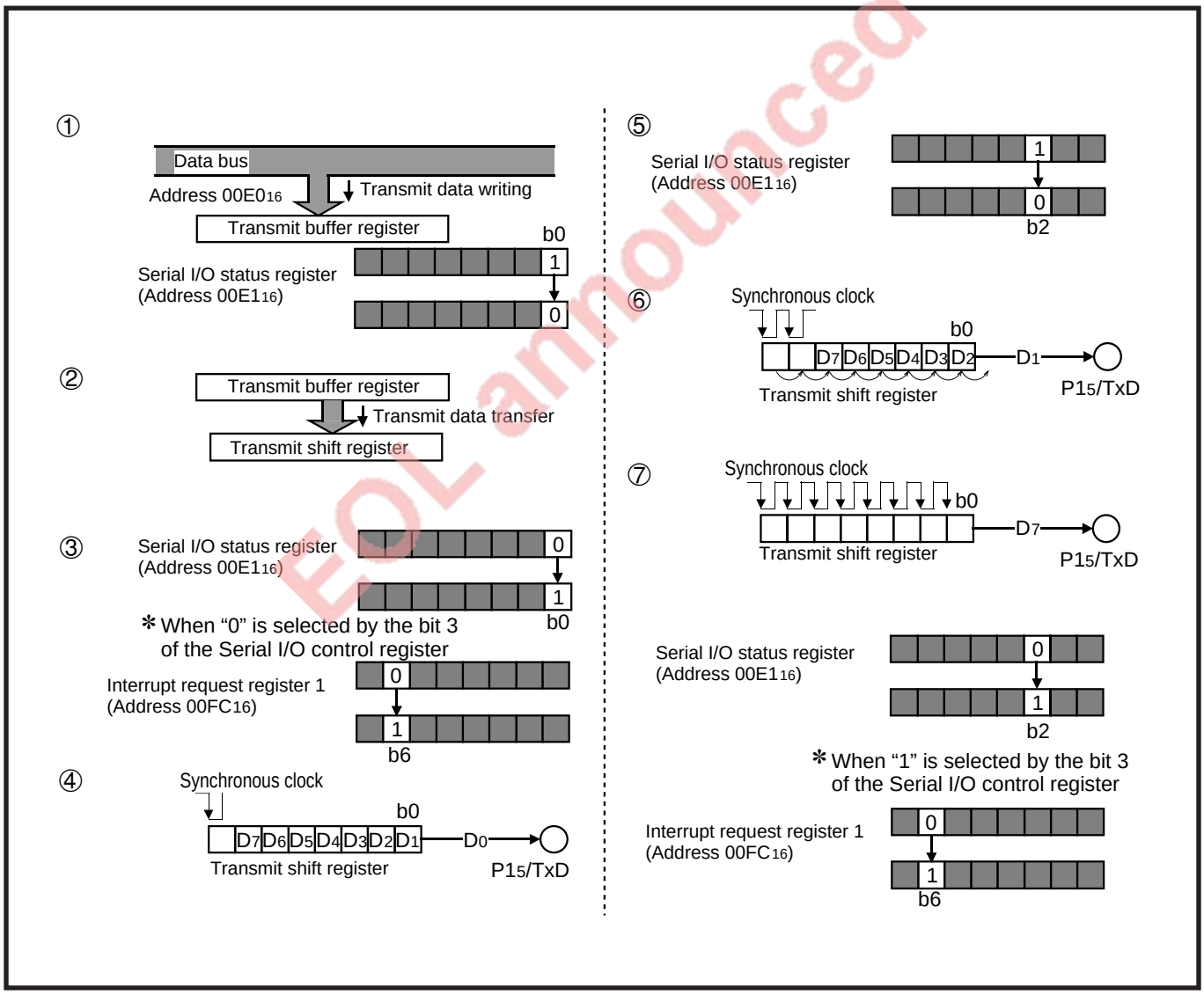
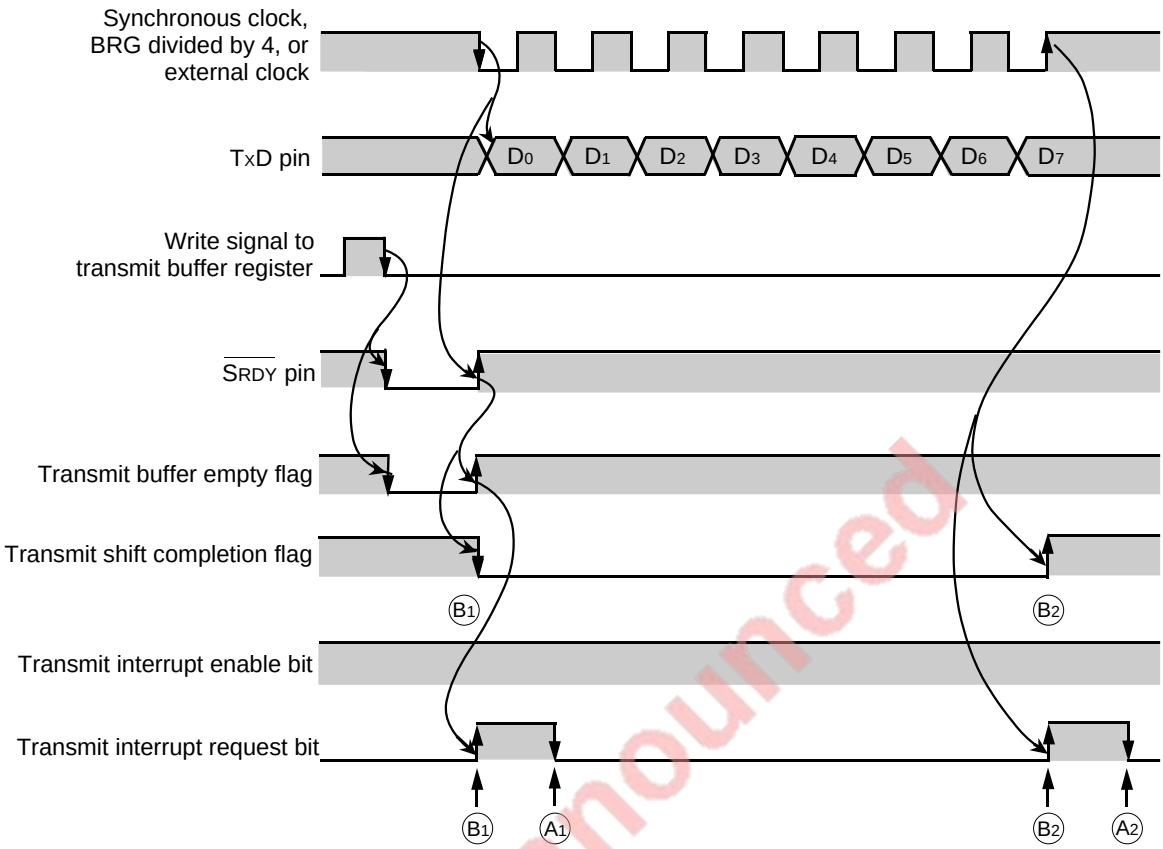


Fig. 1.13B.2 Transmit operation of clock synchronous Serial I/O



(A1) (A2) : • Clearing by writing "0" to the transmit interrupt request bit.
• Clearing by accepting the transmit interrupt.

(B1) : When interrupt request generation is selected, when the Transmit buffer register becomes empty by clearing the transmit interrupt source selection bit to "0".

(B2) : When interrupt request generation is selected, when the shift operation of the transmit shift register is completed by setting the transmit interrupt source selection bit to "1".

Fig. 1.13B.3 Transmit timing chart of clock synchronous Serial I/O

[Clock synchronous Serial I/O transmit setting method]

①Clear the Serial I/O transmit interrupt enable bit (bit 6 of Interrupt control register 1) to "0."

②When selecting the internal clock, set the BRG value.

③Set the Serial I/O control register according to Table 1.13B.1.

④When using a Serial I/O transmit interrupt

[1] Clear the Serial I/O transmit interrupt request bit (bit 6 of Interrupt request register 1) to "0."

Note: When the ordinary port is switched over to the Serial I/O port, the Serial I/O transmit interrupt request bit may be set to "1." Clear the Serial I/O transmit interrupt request bit to "0" after one instruction or more after switching the ordinary port over to the Serial I/O port.

[2] Set the Serial I/O transmit interrupt enable bit to "1."

⑤Write transmit data into the Transmit buffer register (TB: address $00E0_{16}$).

Note: When the external clock is selected, perform a write operation while the synchronous clock is at "H."

Table 1.13B.1 Clock synchronous Serial I/O transmit setting

Item \ Register to be used		Serial I/O control register (SIOCON: Address $00E2_{16}$)	
		Bit	Setting value
BRG count source selection (Note 1)	$f(X_{IN})/4$	b0	0
	$f(X_{IN})/16$		1
Synchronous clock selection	BRG output divided by 4	b1	0
	External clock input		1
$\overline{SRDY}$ signal output selection	Ordinary port	b2	0
	$\overline{SRDY}$ signal output		1
Transmit interrupt request selection	Transmit buffer empty	b3	0
	When the transmit shift operation is completed		1
Transmit enable selection	Transmit enable	b4	1(Note 2)
Receive enable selection	Disable (half-duplex data communication)	b5	0
	Enable (full-duplex data communication)		1
Clock synchronous selection	Clock synchronous	b6	1
Serial I/O enable selection	$P1_4$ to $P1_7$ function as serial I/O pins	b7	1

Notes 1 : $f(X_{CIN})/4$ (setting value : 0), $f(X_{CIN})/16$ (setting value : 1) can be selected in the 7478 Group.

2 : When the external clock is selected, write "1" in bit 4 (transmit enable bit) while the synchronous clock is at "H."

Receive operation of clock synchronous Serial I/O

The receive operation of the clock synchronous Serial I/O is described below.

● Start of receive operation

Receive operation begins by writing data into the Transmit buffer register (TB: address $00E0_{16}$)[*2] in the receive enable state.[*1]

- Transmit data in the full-duplex data communication
- Arbitrary dummy data in the half-duplex data communication

● Receive operation

①Receive data is input bit by bit from the P14/RxD pin to the Receive shift register in synchronization with the rise of the synchronous clock.

②Receive data is input starting with the most significant bit of the Receive shift register. Each time one bit is input, the contents of the Receive shift register are shifted by 1 in the direction of the least significant bit.

③After one-byte data is completely input to the Receive shift register, the contents of the Receive shift register are transferred to the receive buffer register (RB).[*3]

④When receive data has been transferred to the receive buffer register, the receive buffer full flag (b1) of the Serial I/O status register (SIOSTS) is set to "1,"[*4] so that a receive interrupt request is generated.

∗1: Status in which the register for receive operation has been completed. Refer to **"[Clock synchronous Serial I/O receive setting method]"** which will be described later.

∗2: When the external clock is selected, write data into the Transmit buffer register when the synchronous clock is at "H."

∗3: If receive data is further input to the Receive shift register when data remains (when the receive buffer full flag is "1") without reading out the contents of the Receive buffer register, the overrun error flag of the Serial I/O status register is set to "1." At this time, the data of the Receive shift register is not transferred to the Receive buffer register and the original data of the Receive buffer register is held.

∗4: The receive buffer full flag is cleared to "0" by reading out the Receive buffer register.

● When using the $\overline{\text{SRDY}}$ output

At the time when data has been written into the Transmit buffer register, the level of the $\overline{\text{SRDY}}$ signal changes from "H" to "L" by which a receive ready state can be known externally. The $\overline{\text{SRDY}}$ signal goes to "H" at the first fall of the synchronous clock of the synchronous clock.

● Receive interrupt operation (Serial I/O select only)

When receive data is transferred from the receive shift register to the Receive buffer register after one-byte data is all input to the Receive shift register, an interrupt request is generated.

Figure 1.13B.4 shows a receive operation of the clock synchronous Serial I/O and Figure 1.13B.5 shows a receive timing chart of the clock synchronous Serial I/O.

1.13 Serial I/O

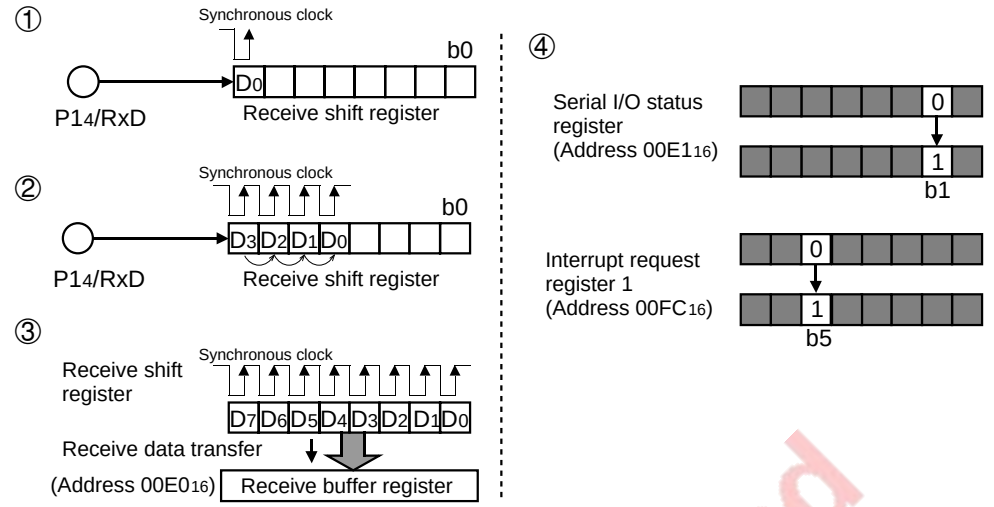


Fig. 1.13B.4 Receive operation of clock synchronous serial I/O

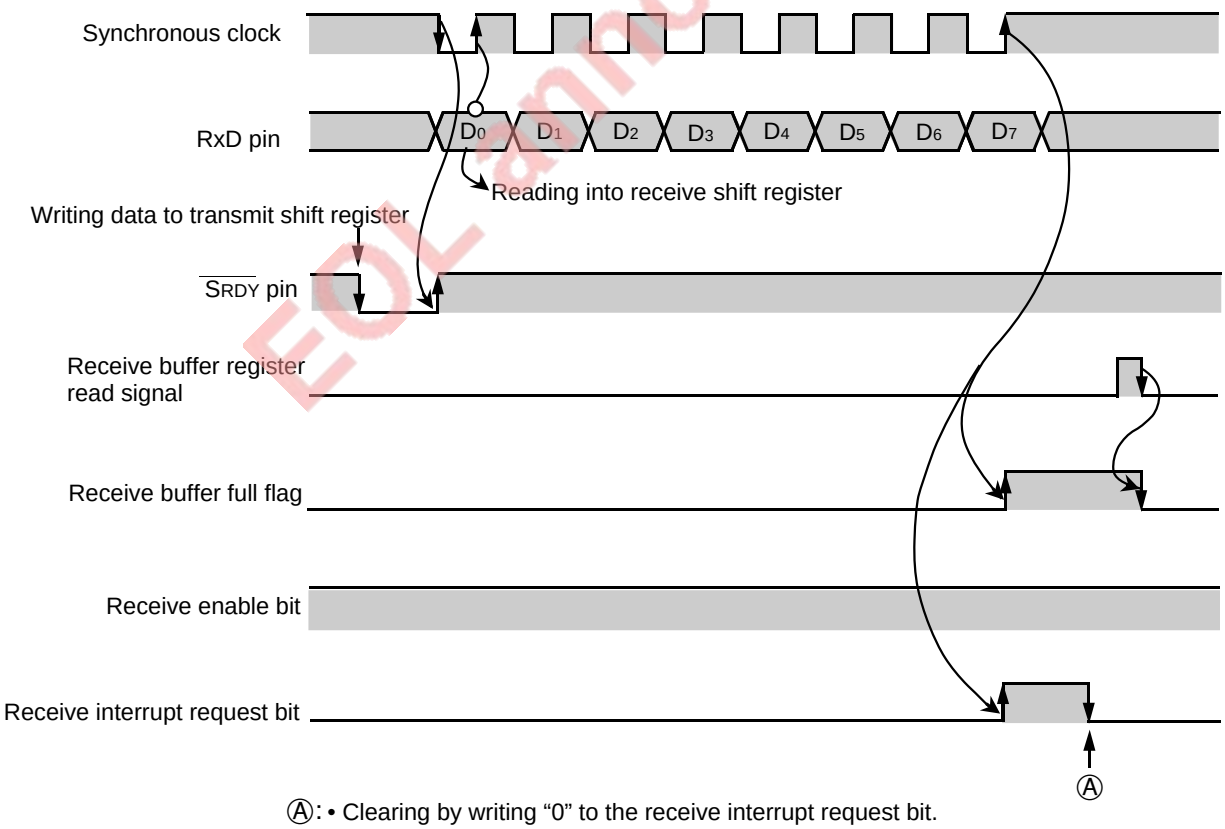


Fig. 1.13B.5 Receive timing chart of clock synchronous serial I/O

[Clock synchronous Serial I/O receive setting method]

①Clear the Serial I/O receive interrupt enable bit (bit 5 of interrupt control register 1) to "0."
②When selecting the internal clock, set the BRG value.
③Set the Serial I/O control register according to Table 1.13B.2.
④When using a Serial I/O receive interrupt
[1] Clear the Serial I/O receive interrupt request bit (bit 5 of interrupt request register 1) to "0."
Note: When the ordinary port is switched over to the Serial I/O port, the Serial I/O receive interrupt request bit may be set. Clear the Serial I/O receive interrupt request bit to "0" after one instruction or more after switching the ordinary port over to the Serial I/O port.
[2] Set the Serial I/O receive interrupt enable bit to "1."
⑤Set the following data into the Transmit buffer register (TB).
• Transmit data in the full-duplex data communication
• Arbitrary dummy data in the half-duplex data communication
Note: When the external clock is selected, perform a write operation while the synchronous clock is at "H."

Table 1.13B.2 Clock synchronous Serial I/O receive setting

Item \ Register to be used		Serial I/O control register (SIOCON: Address 00E2$_{16}$)	
		Bit	Setting value
BRG count source selection (Note 1)	f(XIN)/4	b0	0
	f(XIN)/16		1
Synchronous clock selection	BRG output divided by 4	b1	0
	External clock input		1
$\overline{\text{SRDY}}$ signal output selection	Ordinary port	b2	0
	$\overline{\text{SRDY}}$ signal output (Note 2)		1
Transmit enable selection	Disable (half-duplex data communication)	b4	0
	Enable (full-duplex data communication)		1(Note 3)
Receive enable selection	Receive enable	b5	1
Clock synchronous selection	Clock synchronous	b6	1
Serial I/O enable selection	P14 to P17 function as Serial I/O pins	b7	1

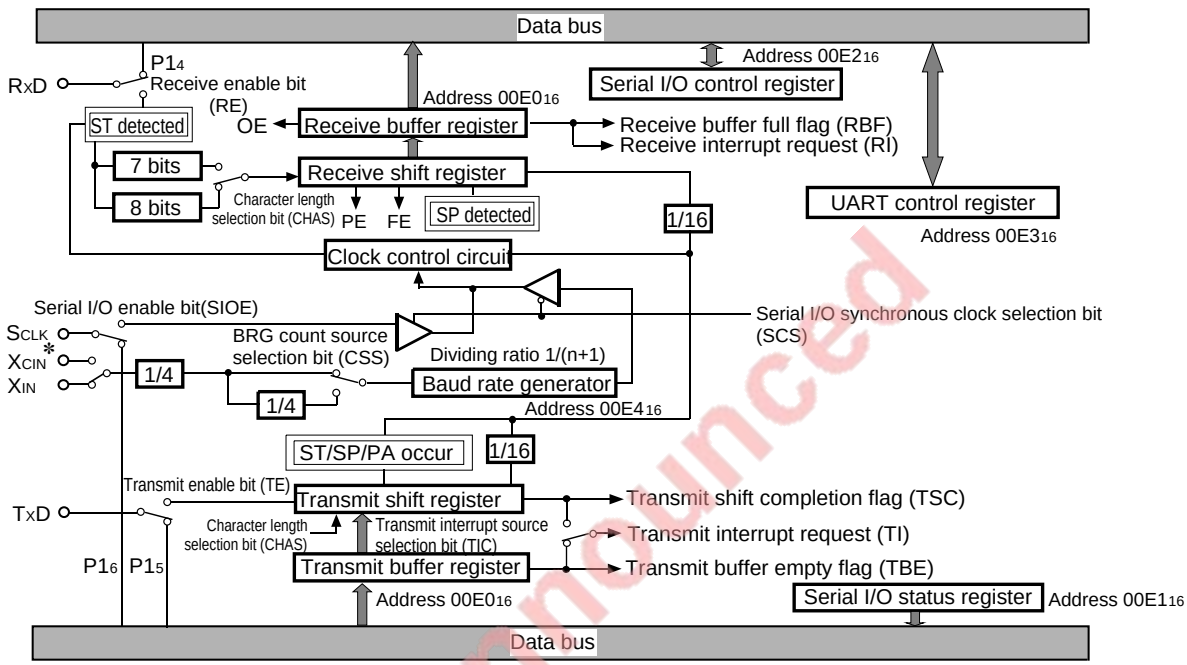
Notes 1: f(XCIN)/4 (setting value : 0), f(XCIN)/16 (setting value : 1) can be selected in the 7478 Group.
2: When the receive side performs an $\overline{\text{SRDY}}$ output by using an external clock, set the receive enable bit, the $\overline{\text{SRDY}}$ output enable bit, and the transmit enable bit to "1" (transmit enable).
3: When the external clock is selected, write "1" in bit 4 (transmit enable bit) while the synchronous clock is at "H."

1.13 Serial I/O

(2) Clock asynchronous Serial I/O

In case of the clock asynchronous Serial I/O (UART), the transmit operation of the transmit side and the receive operation of the receive side are simultaneously executed by unifying the baud rate and the transfer data format between both transmit side and receive side.

Figure 1.13B.6 shows a UART block diagram.



✻: The 7477 group is not provided with the XCIN pin.

Fig. 1.13B.6 UART block diagram

■ Synchronous clock

The following can be selected as a synchronous clock by bit 1 of the Serial I/O control register (SIOCON: address 00E2$_{16}$).

- "0" : Baud rate generator (BRG) output divided by 16
- "1" : External clock input from the SCLK pin divided by 16

The BRG output is set by the baud rate generator (BRG: address 00E4$_{16}$), which is an 8-bit counter dedicated to the Serial I/O. As an input clock to the BRG, f(XIN)/4 (at "0"), f(XIN)/16 (at "1") can be selected by bit 0 of the Serial I/O control register. In the 7478 Group, f(XCIN)/4 (at "0"), f(XCIN)/16 (at "1"), can be also selected.

Precaution on internal clock selection

In the UART, when the internal clock is selected as a synchronous clock, the P16/SCLK pin can be used as port P16.

Notes on external clock selection

- Set the "H" and "L" widths (TWH, TWL) of the pulse used as the external clock source to TWH, TWL [s] ≧ 2/(f(XIN) [Hz]). For example, when f(XIN) = 8 MHz, use a clock of 2 MHz or less (duty ratio 50 %).

■ Data transfer speed (Baud rate)

In the UART, the expression for calculating a data transfer speed (baud rate), which is the frequency of the synchronous clock is shown below.

● When the internal clock is selected (using the BRG)

$$\text{Baud rate [bps]} = \frac{f(X_{IN})^{*3}}{\text{Division ratio}^{*1} \times (\text{BRG set value}^{*2} + 1) \times 16}$$

∗1 Division ratio : Select "4" or "16" by the BRG count source selection bit.
∗2 BRG set value : 0 – 255 (00$_{16}$ – FF$_{16}$)
∗3 In the 7478 Group, f(XCIN) can be also used.

● When the external clock is selected

$$\text{Baud rate [bps]} = \frac{\text{Input clock oscillation frequency to SCLK pin}}{16}$$

1.13 Serial I/O

The BRG is an 8-bit counter dedicated to the Serial I/O, having a reload register, and divides the count source by (n + 1) by setting the value n. As a count source, f(XIN)/4 (at "0"), f(XIN)/16 (at "1") can be selected by bit 0 of the Serial I/O control register. In the 7478 Group, f(XCIN)/4 (at "0"), f(XCIN)/16 (at "1"), can be also selected.

Table 1.13B.3 shows a baud rate reference value.

Table 1.13B.3 Baud rate reference value

Baud rate [bps]	At f(XIN) = 7.9872 MHz		At f(XIN) = 3.9936 MHz	
	Count source	BRG set value	Count source	BRG set value
300	f(XIN)/16	103(67_{16})	f(XIN)/16	51(33_{16})
600	f(XIN)/16	51(33_{16})	f(XIN)/16	25(19_{16})
1200	f(XIN)/16	25(19_{16})	f(XIN)/16	12($0C_{16}$)
2400	f(XIN)/16	12($0C_{16}$)	f(XIN)/4	25(19_{16})
4800	f(XIN)/4	25(19_{16})	f(XIN)/4	12($0C_{16}$)
9600	f(XIN)/4	12($0C_{16}$)		
15600	f(XIN)/4	7(07_{16})		
31200	f(XIN)/4	3(03_{16})		
41600	f(XIN)/4	2(02_{16})		

■ Transmit/receive data format

A transmit/receive data format can be selected by the bits of the UART control register (UARTCON).

- Start bit (ST) : 1-bit
- Data bit (DATA) : 7-bit or 8-bit
- Parity bit (PA) : Non or 1-bit
- Stop bit (SP) : 1-bit or 2-bit

Figure 1.13B.7 shows a transmit/receive data format, Table 1.13B.4 shows a function of each bit of transmit data, and Figure 1.13B.8 shows all data formats.

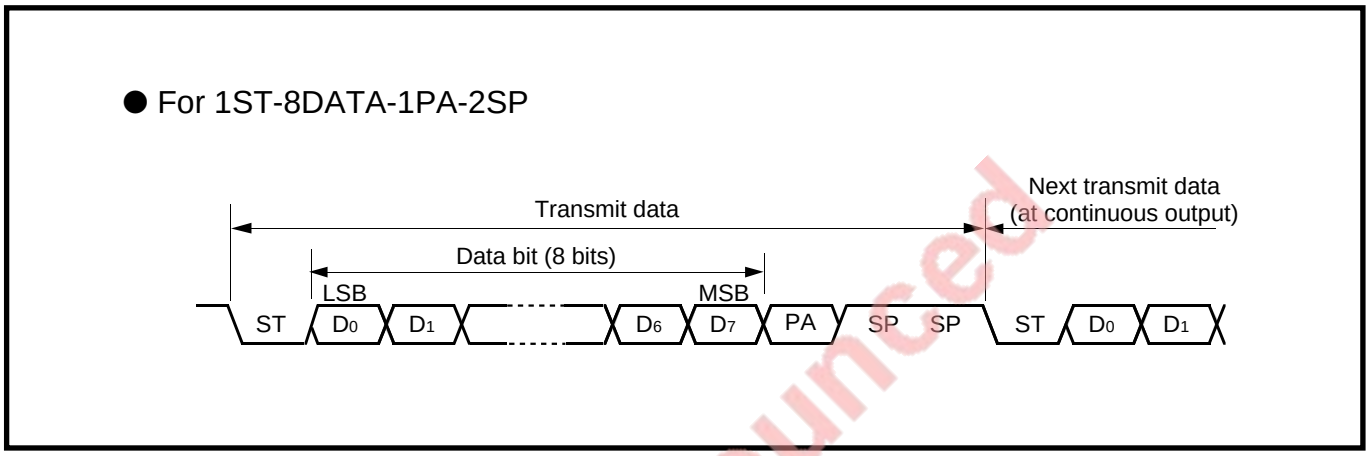


Fig. 1.13B.7 UART data format

Table 1.13B.4 Each bit function of UART transmit data

Name	Function
Start bit ST	The “L” signal for 1 bit is added by the bit indicating a start of data transmission immediately before the transmit data.
Data bit DATA	This bit indicates the transmit data written in the UART transmit buffer register. The “0” data is an “L” signal and the “1” data is an “H” signal.
Parity bit PA	This bit is added immediately after the data bit for improvement of data reliability. The contents of this bit change according to the contents of the parity selection bit so that the number of “1”s in the transmit data including the parity bit may always be even or odd.
Stop bit SP	This bit indicates that data has been transmitted, and is added immediately after the data bit (immediately after the parity bit when the parity is valid). The “H” signal for 1 bit or 2 bits is output.

1.13 Serial I/O

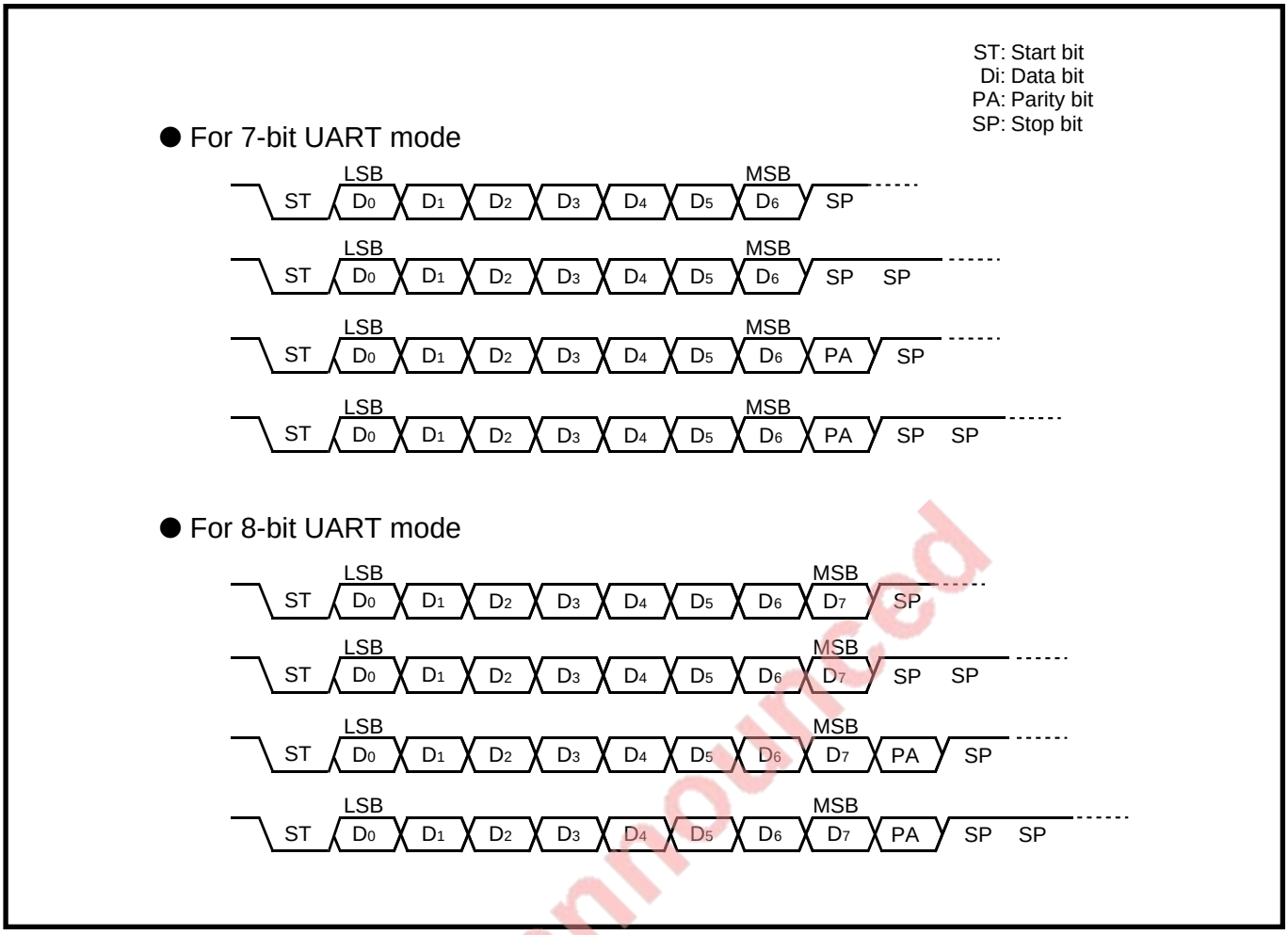


Fig. 1.13B.8 Transmit/receive format of UART

■ Transmit operation of UART

The Transmit operation of the UART is described below.

● Start of Transmit operation

Transmit data is transmitted by writing it into the Transmit buffer register (TB: address $00E0_{16}$) in the Transmit enable state.*1

● Transmit operation

① After transmit data is written into the Transmit buffer register, the transmit buffer empty flag (bit 0) of the Serial I/O status register is cleared to "0."

② The transmit data written in the Transmit buffer register is transferred to the Transmit shift register. When the data transfer from the Transmit buffer register to the Transmit shift register is completed, the transmit buffer empty flag is set to "1."*2
When the transmit interrupt source bit (bit 3) of the Serial I/O control register (SIOCON) is "0," the interrupt request bit is set to "1," then a transmit interrupt request occurs.

③ The transmit data transferred to the transmit shift register is output from the $P1_5$/TxD pin in synchronization with the fall of the synchronous clock starting with the start bit. The start bit, the parity bit and the stop bit are automatically generated and output according to the contents of setting of the UART control register.

④ When a transmit shift operation is started, the transmit shift completion flag (b2) of the Serial I/O status register is cleared to "0."

⑤ Data is output starting with the least significant bit of the Transmit shift register. Each time one-bit data is output, the contents of the Transmit shift register are shifted by 1 bit in the direction of the least significant bit.

⑥ After one-half a cycle of the synchronous clock*3 after a start of stop bit transmission, the transmit shift completion flag is set to "1."
When the bit 3 of the Serial I/O control register is "1" (transmit shift operation is completed), at the time the interrupt request bit is set to "1" and the Transmit interrupt request occurs.

∗1: Status in which the register for transmit operation has been completed. Refer to the **"[UART transmit setting method]"** which will be described later.

∗2: While the transmit buffer empty flag is "1," the next transmit data can be written into the Transmit buffer register.

∗3: In case of two stop bits, the stop bit output period is that of the 2nd bit.

● **Transmit interrupt operation** (valid when the Serial I/O is selected)

Regarding a transmit interrupt, interrupt request generating timing can be selected by bit 3 of the Serial I/O control register (SIOCON).

0: When the Transmit buffer register becomes empty after the data written in the Transmit buffer register is transferred to the Transmit shift register, an interrupt request is generated.

1: When the shift operation of the Transmit shift register is completed, an interrupt request is generated.

※ In case of the UART, an interrupt operation is performed in the same way as when the synchronous clock is selected.

Figure 1.13B.9 shows a transmit operation of UART and Figure 1.13B.10 shows a transmit timing of UART.

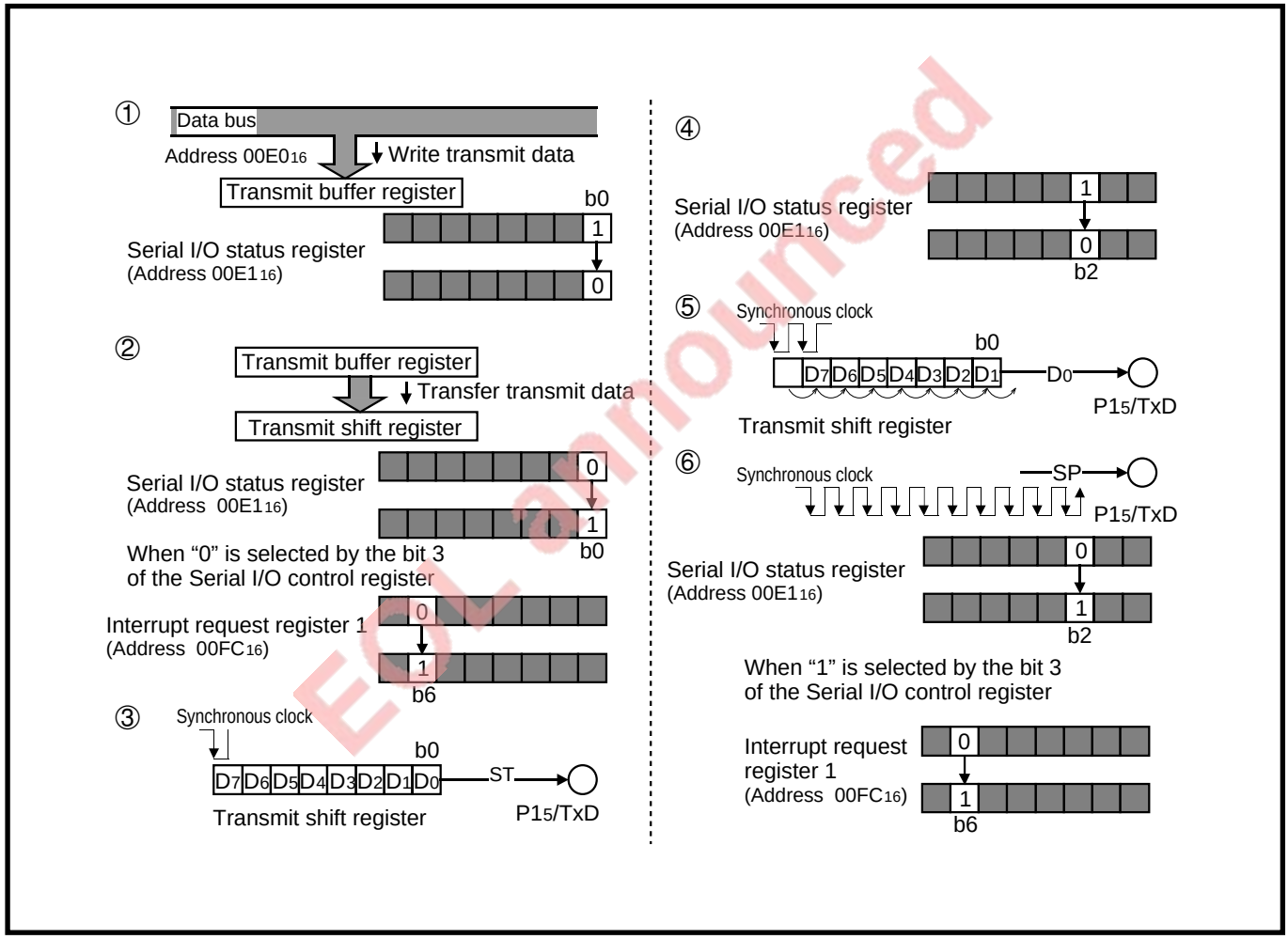
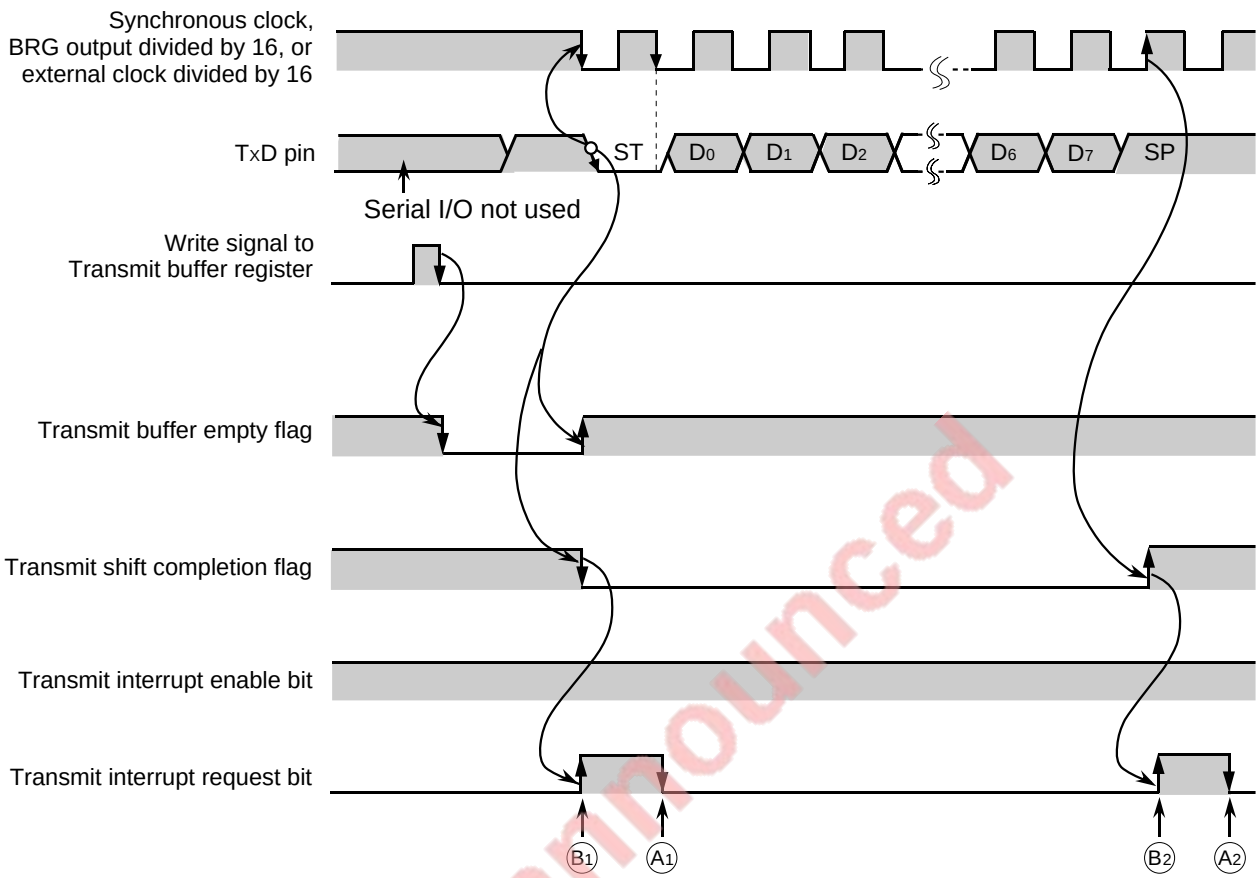


Fig. 1.13B.9 Transmit operation of UART



(A1) (A2) : • Clearing by writing "0" to the transmit interrupt request bit.
• Clearing by accepting the transmit interrupt.

(B1) : When interrupt request generation is selected, when the Transmit buffer register becomes empty by clearing the transmit interrupt source selection bit to "0".

(B2) : When interrupt request generation is selected, when the shift operation of the Transmit shift register is completed by setting the transmit interrupt source selection bit to "1".

Fig. 1.13B.10 Transmit timing chart of UART

[UART transmit setting method]

① Clear the Serial I/O transmit interrupt enable bit (bit 6 of Interrupt control register 1) to "0."
② When selecting the internal clock, set the BRG value.
③ Set the Serial I/O control register according to Table 1.13B.5.
④ Set the data format according to Table 1.13B.6.
⑤ When using a Serial I/O transmit interrupt
[1] Clear the Serial I/O transmit interrupt request bit (bit 6 of Interrupt request register 1) to "0."
Note: When the ordinary port is switched over to the Serial I/O port, the Serial I/O transmit interrupt request may be set to "1." Clear the Serial I/O transmit interrupt request bit to "0" after one instruction or more after switching the ordinary port over to the Serial I/O port.
[2] Set the Serial I/O transmit interrupt enable bit to "1."
⑥ Write transmit data into the Transmit buffer register.

Table 1.13B.5 UART transmit setting

Item \ Register to be used		Serial I/O control register (SIOCON: Address 00E2$_{16}$)	
		bit	setting value
BRG count source selection (Note 1)	f (X_{IN})/4	b0	0
	f (X_{IN})/16		1
Synchronous clock selection	BRG output divided by 16	b1	0
	External clock input divided by 16		1
$\overline{SRDY}$ signal output selection		b2	(Note 2)
Transmit interrupt request selection	Transmit buffer empty	b3	0
	When the transmit shift operation is completed		1
Transmit enable selection	Transmit enable	b4	1
Receive enable selection	Disable (Half-duplex data communication)	b5	0
	Enable (Full-duplex data communication)		1
Clock asynchronous selection	Clock asynchronization	b6	0
Serial I/O enable selection	P1$_4$ to P1$_7$ function as Serial I/O pins (Note 3)	b7	1

Notes 1: f(X_{CIN})/4 (setting value : 0), f(X_{CIN})/16 (setting value : 1) can be selected in the 7478 Group.
2: When the UART is selected, this bit does not function.
3: When the internal clock is selected, the P1$_6$/S_{CLK} pin can be used as port P1$_6$.

Table 1.13B.6 Set value of UART control register

Serial data transfer format	UART control register (UARTCON: Address 00E3$_{16}$)			
	b3	b2	b1	b0
1ST-8DATA-1SP	0	–	0	0
1ST-7DATA-1SP	0		0	1
1ST-8DATA-1PA-1SP	0	Selection (Note)	1	0
1ST-7DATA-1PA-1SP	0		1	1
1ST-8DATA-2SP	1	–	0	0
1ST-7DATA-2SP	1		0	1
1ST-8DATA-1PA-2SP	1	Selection (Note)	1	0
1ST-7DATA-1PA-2SP	1		1	1

Note: 0: Even parity
1: Odd parity

■ Receive operation of UART

The receive operation of UART is described below.

● Start of receive operation

Set the receive enable bit (bit 5) of the Serial I/O control register (SIOCON) to the enable state ("1") in the receive enable state.*1 With this operation, the start bit is detected and serial data is received.

● Receive operation

① After a fall of the P14/RxD pin is detected, the level of the P14/RxD pin is checked after one-half a cycle of the synchronous clock. If its level is "L," the bit is judged as a start bit. When its level is "H," it is judged that noise is generated, so that the the receive operation is stopped and the UART waits for the start bit.

② Receive data is input bit by bit from the P14/RxD pin to the Receive shift register in synchronization with the rise of the synchronous clock.

③ Data, immediately after the start bit, is input starting with the most significant bit of the Receive shift register. Each time one bit is received, the contents of the Receive shift register are shifted by 1 bit in the direction of the least significant bit.

④ When the specified number of bits are all input in the Receive shift register, the contents of the Receive shift register are transferred to the Receive buffer register (RB).*2,*3

⑤ After 1/2 cycle of the shift clock after a start of stop bit reception, the receive buffer full flag (bit 1) of the Serial I/O status register (SIOSTS) is set to "1"*4 and a receive interrupt request is generated.

⑥ Error flag detection is started concurrently with the occurrence of the receive interrupt request.

∗1: Status in which the register for receive operation has been completed. Refer to the **"[UART receive setting method]"** which will be described later.

∗2: When the data bit length is 7 bits, the contents of the Receive buffer register consist of receive data of bits 0 to 6 and "0" of bit 7 (MSB).

∗3: If receive data is further input to the Receive shift register when data remains (when the receive buffer full flag is "1") without reading out the contents of the Receive buffer register, the overrun error flag of the Serial I/O status register is set to "1." At this time, the data of the Receive shift register is not transferred to the Receive buffer register and the original data of the Receive buffer register is held.

∗4: The receive buffer full flag is cleared to "0" by reading out the Receive buffer register.

● **Receive interrupt operation** (valid when the Serial I/O is selected)
When receive data is transferred from the Receive shift register to the Receive buffer register after one-byte data is all input to the Receive shift register, an interrupt request is generated.

※ In case of the UART, an interrupt operation is performed in the same way as when the synchronous clock is selected.

Figure 1.13B.11 shows a receive operation of UART and Figure 1.13B.12 shows a receive timing of UART.

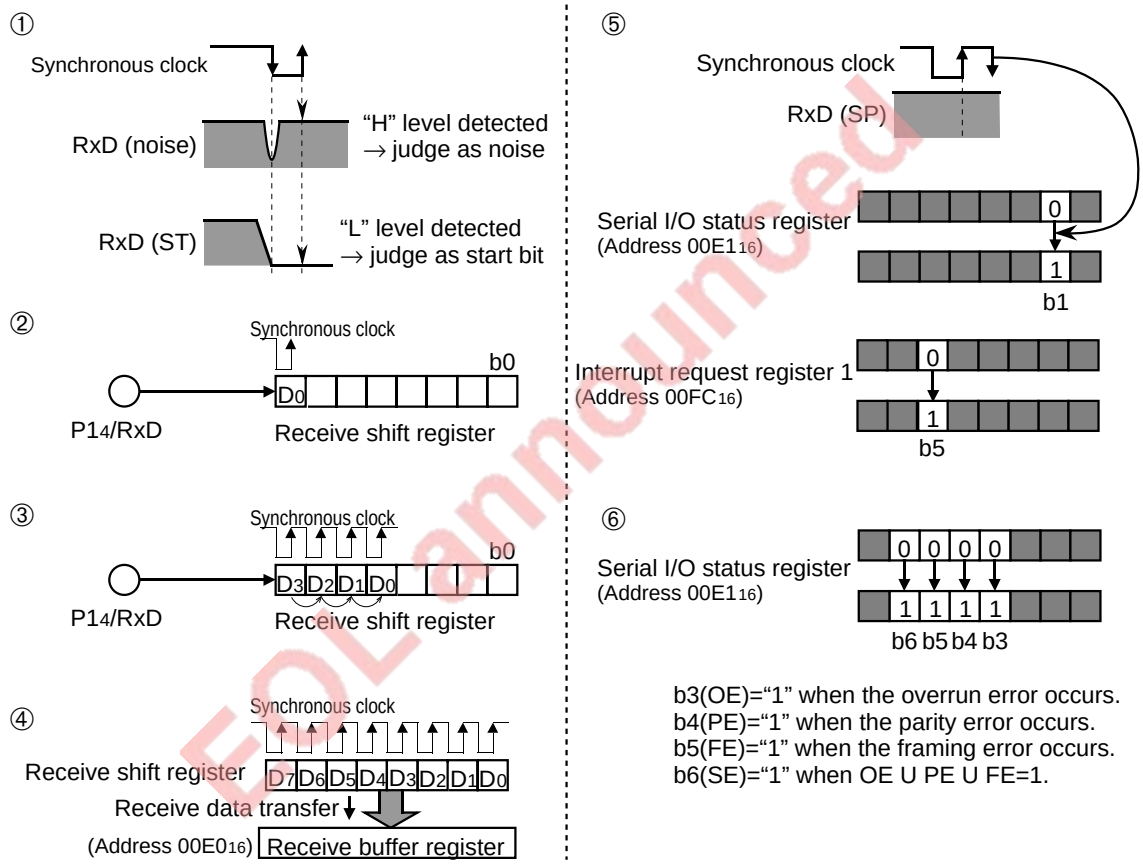


Fig. 1.13B.11 Receive operation of UART

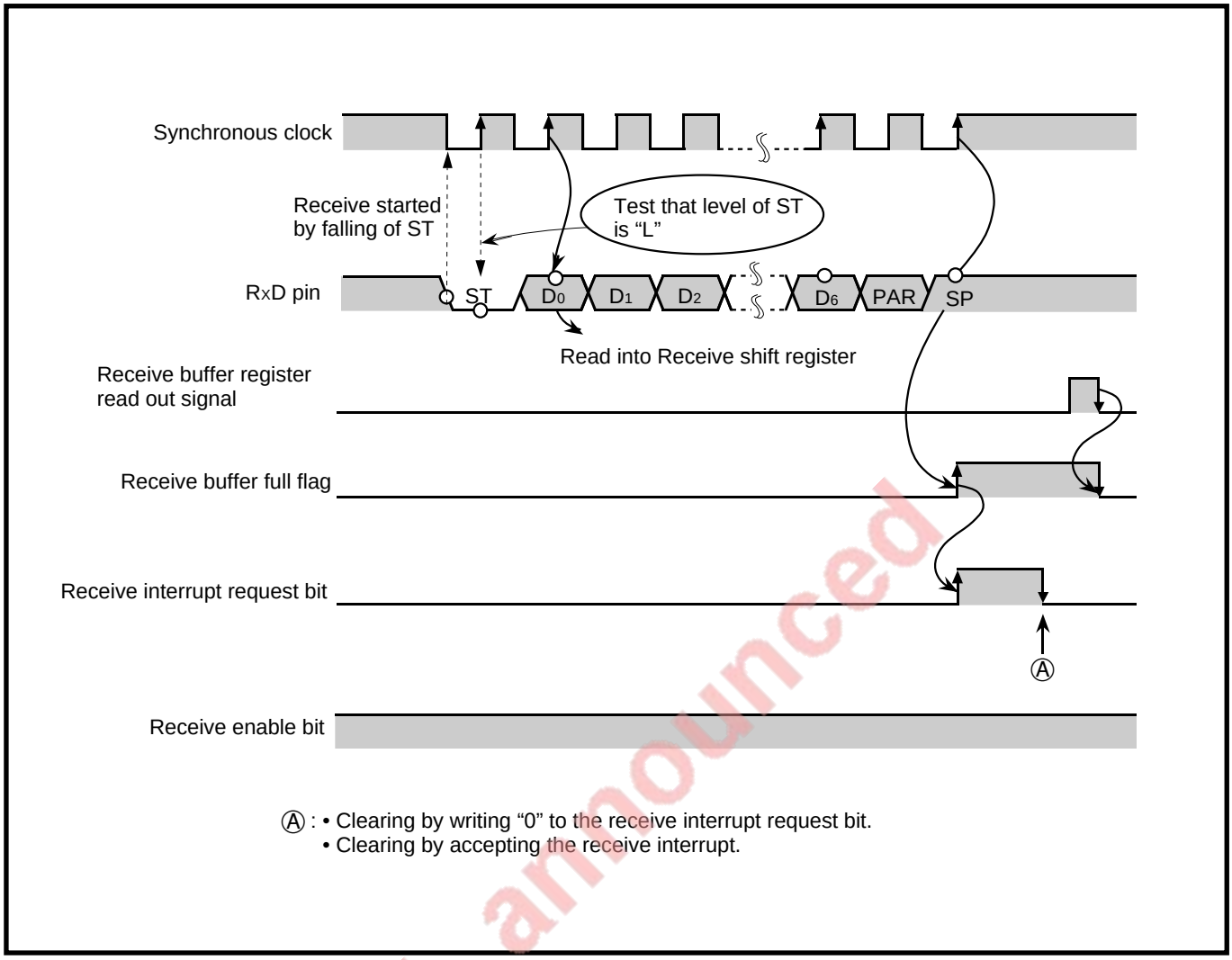


Fig. 1.13B.12 Receive timing chart of UART

[UART receive setting method]

① Clear the Serial I/O receive interrupt enable bit (bit 5 of Interrupt control register 1) to "0."
② When selecting the internal clock, set the BRG value.
③ Set the Serial I/O control register according to Table 1.13B.7.
④ Set the data format according to Table 1.13B.6.
⑤ When using a Serial I/O receive interrupt
[1] Clear the Serial I/O receive interrupt request bit (bit 5 of Interrupt request register 1) to "0."
Note: When the ordinary port is switched over to the Serial I/O port, the Serial I/O receive interrupt request may be set to "1." Clear the Serial I/O receive interrupt request bit to "0" after one instruction or more after switching the ordinary port over to the Serial I/O port.
[2] Set the Serial I/O receive interrupt enable bit to "1."
⑥ In the full-duplex data communication, set transmit data in the Transmit buffer register (TB).

Table 1.13B.7 UART receive setting

Item \ Register to be used		Serial I/O control register (SIOCON: Address 00E2$_{16}$)	
		bit	setting value
BRG count source selection (Note 1)	f (X_{IN})/4	b0	0
	f (X_{IN})/16		1
Synchronous clock selection	BRG output divided by 16	b1	0
	External clock input divided by 16		1
$\overline{SRDY}$ signal output selection		b2	(Note 2)
Transmit interrupt request selection	Transmit buffer empty	b3	0
	When the transmit shift operation is completed		1
Transmit enable selection	Disable (Half-duplex data communication)	b4	0
	Enable (Full-duplex data communication)		1
Receive enable selection	Receive enable	b5	1
Clock asynchronous selection	Clock asynchronous	b6	0
Serial I/O enable selection	P1$_4$ to P1$_7$ function as serial I/O pins (Note 3)	b7	1

Notes 1: f(X_{CIN})/4 (setting value : 0), f(X_{CIN})/16 (setting value : 1) can be selected in the 7478 Group.
2: When the UART is selected, this bit does not function.
3: When the internal clock is selected, the P1$_6$/S_{CLK} pin can be used as port P1$_6$.

1.13B.2 Pins

The 7477/7478 group uses 4 pins for data transmit, data receive, shift clock transmit/receive and serial I/O transfer ready signal output. All these pins are used in common with P1. A function selection is made by the serial I/O enable bit (bit 7) and the $\overline{\text{SRDY}}$ output enable bit (bit 2) of the Serial I/O control register. The function of each pin is explained below.

(1) Data transmit pin[TxD]

Transmit data is output bit by bit. This pin is used in common with $P1_5$. When the transfer enable bit and the serial I/O enable bit of the Serial I/O control register is set to "1," this pin becomes a serial I/O data output pin.

(2) Data receive pin [RxD]

Data is input bit by bit. This pin is used in common with $P1_4$. When the receive enable bit and the serial I/O enable bit of serial I/O control register are set to "1," this pin becomes a serial I/O data input pin.

(3) Shift clock transmit/receive pin [SCLK]

■ **Clock synchronous**

This pin inputs (receives from the outside) or outputs (supplies to the outside) the synchronous clock for data transmit/receive.

When the serial I/O synchronous clock selection bit (bit 1) of the Serial I/O control register is cleared to "0" (use of internal clock), the synchronous clock is output.

When the same bit is set to "1" (use of internal clock), the synchronous clock is input from the outside.

■ **Clock asynchronous (UART)**

When the serial I/O synchronous clock selection bit (bit 1) of the Serial I/O control register is set to "1" (use of external clock), the synchronous clock is supplied from the outside.

When the same bit is cleared to "0" (use of internal clock), this pin does not function.

Note: When the internal clock is selected, SCLK pin can be used as port $P1_6$.

(4) Serial transfer enable signal output pin [$\overline{\text{SRDY}}$]

This pin informs the outside of a receive enable state in the clock synchronous serial I/O.
In case of the UART, this pin does not function.

- $\overline{\text{SRDY}}$ signal output enable bit (bit 2) of Serial I/O control register is set to "1."
- Transmit enable bit (bit 4) of Serial I/O control register is set to "1."

When the above 2 conditions are satisfied, the level of the pin changes from "H" to "L" at the timing at which data was written into the Transmit buffer register, informing the outside of a serial transfer enable state.

1.13B.3 Notes on use

(1) Notes on external clock selection

In the 7477/7478 group, either the internal clock or external clock can be selected as the synchronous clock. When the external clock is selected as the synchronous clock, take the following points into consideration.

■ **Clock synchronous serial I/O**

① During data transmission, when setting the transmit enable bit to "1" or writing data into the Transmit buffer register, perform a write operation while the synchronous clock is at "H."

② The transmission or the shift operation of the Receive shift register is continued while the synchronous clock is input to the serial I/O circuit. When the external clock is selected, stop the synchronous clock at the end of 8 cycles. When the internal clock is selected, the synchronous clock stops automatically at the end of 8 cycles.

③ When the external clock is selected, set the "H" and "L" widths (T_{WH}, T_{WL}) of the pulse used as the external clock source to T_{WH}, T_{WL} [s] ≧ 8/(f(X_{IN}) [Hz]). For example, when f(X_{IN}) is 8 MHz, use a clock of 500 kHz or less (duty ratio 50 %).

■ **UART**

Set the "H" and "L" widths (T_{WH}, T_{WL}) of the pulse used as the external clock source to T_{WH}, T_{WL} [s] ≧ 2/(f(X_{IN}) [Hz]). For example, when f(X_{IN}) is 8 MHz, use a clock of 2 MHz or less (duty ratio 50 %).

(2) When the $\overline{S_{RDY}}$ output is performed in the clock synchronous serial I/O

When the receive side using the external clock performs an $\overline{S_{RDY}}$ output, set the receive enable bit, the $\overline{S_{RDY}}$ output enable and the transmit enable bit to "1" (transmit enable).

(3) When a serial I/O transmit interrupt or a serial I/O receive interrupt is caused

■ **When using a serial I/O transmit interrupt**

① Clear the serial I/O transmit interrupt request bit (bit 6 of IR1) to "0" after one instruction or more after setting a value in the Serial I/O control register.

② After setting in ①, set the serial I/O transmit interrupt enable bit (bit 6 of IE1) to "1."

■ **When using a serial I/O receive interrupt**

① Clear the serial I/O receive interrupt request bit (bit 5 of IR1) to "0" after one instruction or more after setting a value in the Serial I/O control register.

② After setting in ①, set the serial I/O receive interrupt enable bit (bit 5 of IE1) to "1."

(4) Transmit interrupt request in the transmit enable state

After the transmit enable bit is set to "1," the transmit buffer empty flag and the transmit shift completion flag are set to "1." Accordingly, even if a transmit buffer empty state is selected or a termination of shift operation of the Transmit shift register is selected as a transmit interrupt source, an interrupt request is generated and the transmit interrupt request bit is set to "1."

For this reason, when using a transmit interrupt, set the transmit enable bit to "1," clear the transmit interrupt request bit to "0," and then set the transmit interrupt enable bit to "1" (enable state).

(5) Disabling transmission after transmission of 1-byte data

In the 7477/7478 group, it is possible to make reference to the transmit shift register completion flag (TSC flag) to know that data has been transmitted.

The TSC flag is "0" during data transmission, and becomes "1" after data has been transmitted. Accordingly, if data transmission is disabled at the time of confirmation of a change of the TSC flag from "0" to "1," data transmission can be terminated after 1-byte data is transmitted. However, the TSC flag is also set to "1" when the serial I/O is enabled and does not become "0" until a synchronous clock is generated and transmitted. For this reason, if data transmission is disabled by making reference to the TSC flag at this time, data is not transmitted. Make reference to the TSC flag after a start of data transmission.

The change of the TSC flag from "1" to "0" has a delay of 0.5 to 1.5 cycles of the synchronous clock.

(6) Re-setting the Serial I/O control register (SIOCON)

Re-set the Serial I/O control register after setting both transmit enable bit and receive enable bit to "0" to re-set the transmit circuit and the receive circuit.

① Clear both transmit enable bit (TE) and receive enable bit (RE) to "0."
② Set the bit 0 to bit 3 and bit 6 of the Serial I/O control register.
③ Set both transmit enable bit (TE) and receive enable bit (RE) to "1."
(It is possible to set ② and ③ simultaneously with the LDM instruction.)

(7) Stopping data transmit/receive

■ In the following cases, clear the transmit enable bit to "0" (transmit disable).

- To stop the transmit operation when data is transmitted in the clock synchronous serial I/O
- To stop the transmit operation when UART data is transmitted
- To stop only the transmit operation when UART data is transferred

■ In the following cases, clear receive enable bit (receive disable) or serial I/O enable bit to "0" (serial I/O disable).

- To stop the receive operation when data is received in the clock synchronous serial I/O

■ In the following cases, clear the receive enable bit to "0."

- To stop the receive operation when UART data is received.
- To stop only the receive operation when UART data is transferred.

■ In the following cases, clear both transmit enable bit and receive enable bit to "0" (transfer disable) simultaneously.

- To stop the transmit operation and the receive operation when data is transferred in the clock synchronous serial I/O

Note: When data is transferred in the clock synchronous serial I/O, it is impossible to stop only the transmit operation or the receive operation.

(8) Processing upon occurrence of errors

■ When a parity error, a framing error or a summing error occurs

When a parity error, a framing error or a summing error occurs, the flag corresponding to each error in the Serial I/O status register is set to “1.” These flags are not cleared to “0” automatically. Clear them to “0” by software.

The parity error flag, the framing error flag and the summing error flag can be cleared to “0” by one of the following two methods.

- Clear the receive enable bit to “0.”
- Write arbitrary dummy data into the Serial I/O status register.

■ Processing upon occurrence of overrun error

An overrun error occurs when data has all been input to the Receive shift register while data is stored in the Receive buffer register.

When an overrun error occurs, the data of the Receive shift register is not transferred to the Receive buffer register and the data of the Receive buffer register is held. At this time, even if the data of the Receive buffer register is read out, the data of the Receive shift register is not transferred. Accordingly, the data of the Receive buffer register can be read out but the data of the Receive shift register cannot be read out and becomes invalid.

When an overrun error occurs, clear the overrun error flag of the Serial I/O status register to “0” and then make preparations for receiving data again.

The overrun error flag can be cleared by one of the following methods.

- Clear the serial I/O enable bit to “0.”
- Clear the receive enable bit to “0.”
- Write arbitrary dummy data into the Serial I/O status register.

1.13B.4 Related registers

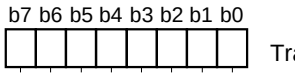
(1) Transmit/receive buffer register (TB/RB: Address 00E0$_{16}$)

The Transmit/receive buffer register is written serial I/O (used in common with the clock synchronous serial I/O and the UART) transmit data and is read out serial I/O receive data.

- To transmit data, write this transmit data into this register.
- Receive data can be obtained by reading this register.

Figure 1.13B.13 shows a structure of the Transmit/receive buffer register.

Transmit/receive buffer register (7477/7478 group)



Transmit/receive buffer register (TB/RB) [Address 00E0$_{16}$]

B	Function	At reset	R	W
0 to 7	At transmit: A value of "00$_{16}$" to "FF$_{16}$" can be set as transmit data. The transmit data is transferred automatically by writing the transmit data into the Transmit shift register. At receive: When all receive data has been input into the Receive shift register, the receive data is automatically transferred to the receive buffer register.	?	○	○

Fig. 1.13B.13 Structure of Transmit/receive buffer register

1.13 Serial I/O

(2) Serial I/O status register (SIOSTS: Address 00E1$_{16}$)

The Serial I/O status register consists of flags for representing the buffer/register state to be used for data transfer, and error flags.

- This register is a read-only type.
- Bit 7 is unused and “1” at a read operation.

Figure 1.13B.14 shows a structure of the Serial I/O status register.

Serial I/O status register (7477/7478 group)

b7 b6 b5 b4 b3 b2 b1 b0

1

Serial I/O status register (SIOSTS) [Address 00E1$_{16}$]

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 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) U (PE) U (FE) = 0 1 : (OE) U (PE) U (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	1	×

Fig. 1.13B.14 Structure of Serial I/O status register

Each bit of the Serial I/O status register is described below.

■ **Transmit buffer empty flag (TBE, bit 0)**

This flag indicates the state of the Transmit buffer register.
This bit is set to "1" after the data written in the Transmit buffer register is transferred to the Transmit shift register, and cleared to "0" after data is written into the Transmit buffer register.
This flag is valid in both clock synchronous serial I/O and UART.

■ **Receive buffer full flag (RBF, bit 1)**

This flag indicates the state of the Receive buffer register.
When 1-byte data has been all input to the Receive shift register and then the receive data has been transferred from the Receive shift register to the Receive buffer register, this flag is automatically set to "1." When the transferred data has been read out from the Receive buffer register, the flag is automatically cleared to "0."
If receive data is further input to the Receive shift register when the receive buffer full flag is "1" (without reading out the contents of the Receive buffer register), the overrun error flag is set to "1."
The receive buffer full flag is valid in both clock synchronous serial I/O and UART.

■ **Transmit shift completion flag (TSC, bit 2)**

This flag indicates the state of the transmit shift operation.
When transmit data has been transferred to the Transmit shift register and then a shift operation has been started with the synchronous clock (transmission of the 1st bit of the transmit data), this flag is cleared to "0." When the shift operation has been completed (completion of transmission the last bit of the transmit data), the flag is set to "1."
This flag is valid in both clock synchronous serial I/O and UART.

■ **Overrun error flag (OE, bit 3)**

This flag indicates the receive data read state.
If receive data is further input to the Receive shift register when the receive buffer full flag is "1" (without reading out the contents of the Receive buffer register), the overrun error flag is set to "1."
This flag is cleared to "0" by any operation shown in Table 1.13B.8.
This flag is valid in both clock synchronous Serial I/O and UART.

■ **Parity error flag (PE, bit 4)**

This flag indicates a hardware check result on the even parity or odd parity in the UART.
If there is a difference between the parity of received data and the set parity, the flag is set to "1."
This flag is cleared to "0" by any operation shown in Table 1.13B.8.
This flag is valid in the parity enable state in UART.

■ **Framing error flag (FE, bit 5)**

This flag judges a frame synchronization error in UART.
When the stop bit of receive data cannot be received at the set timing, the flag is set to "1."
At stop bit detection, only the 1st stop bit is detected but the 2nd stop bit is not checked.
This flag is cleared by any operation shown in Table 1.13B.8.
This flag is valid in UART only.

■ **Summing error flag (SE, bit 6)**

This flag judges a serial I/O error.
If one of an overrun error, a parity error and a framing error occurs, the flag is set to “1.”
This flag is cleared by any operation shown in Table 1.13B.8.
This flag is valid in both clock synchronous serial I/O and UART.

[Error flag clear method]

The error flags (bit 3 to bit 6) in the Serial I/O status register can be cleared to “0” by the error flag clear methods shown in Table 1.13B.8.

Table 1.13B.8 Error flag clear method

Error flag	Clear the serial I/O interrupt enable bit to “0.”	Clear the receive enable bit to “0.”	Write dummy data into the SIOSTS.
Overrun error flag	○	○	○
Parity error flag	×	○	○
Framing error flag	×	○	○
Summing error flag	×	○	○

(3) Serial I/O control register (SIOCON: Address 00E2$_{16}$)

The Serial I/O control register exerts various types of control over the serial I/O, for example, transfer mode, clocks and pin function selection. All the bits of this register can be read and written by software.

Figure 1.13B.15 shows a structure of the Serial I/O control register.

Serial I/O control register (7477/7478 group)

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O control register (SIOCON) [Address 00E2$_{16}$]

B	Name	Function	At reset	R	W
0	BRG count source selection bit (CSS)(Note 1)	0 : f(XIN)/4 1 : f(XIN)/16	0	○	○
1	Serial I/O synchronous clock selection bit (SCS)	•In clock synchronous mode 0 : BRG output divided by 4 1 : External clock input •In UART mode 0 : BRG output divided by 16 1 : External clock input divided by 16	0	○	○
2	$\overline{SRDY}$ output enable bit (SRDY) ✻In the UART mode, this bit is invalid.	0 : P17/$\overline{SRDY}$ pin operates as ordinary I/O pin 1 : P17/$\overline{SRDY}$ pin operates as $\overline{SRDY}$ output pin	0	○	○
3	Transmit interrupt source selection bit (TIC)	0 : When transmit buffer has emptied 1 : 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/O mode selection bit (SIOM)	0 : Clock asynchronous serial I/O (UART) 1 : Clock synchronous	0	○	○
7	Serial I/O enable bit (SIOE)	0 : Serial I/O disabled (pins operates as ordinary I/O pins P14–P17) 1 : Serial I/O enabled (pins operates as serial I/O pins RXD - $\overline{SRDY}$) (Note 2)	0	○	○

Notes 1: In the 7478 Group, f(XCIN)/4 (at "0"), f(XCIN)/16 (at "1") can be also selected.

2: Port P14–P17 are operates as the serial I/O pin only when the serial I/O enable bit is "1" (enable state).
At this time, Port P17 is also used as an ordinary I/O port.
In the UART mode, port P16 is used as an ordinary I/O port when the internal clock is selected.

Fig. 1.13B.15 Structure of Serial I/O control register

1.13 Serial I/O

Each bit of the Serial I/O control register is described below.

■ **BRG count source selection bit (CSS, bit 0)**
This bit selects a count source to be input to the BRG.
- "0": f(XIN)/4
- "1": f(XIN)/16

■ **Serial I/O synchronous clock selection bit (SCS, bit 1)**
This bit selects a synchronous clock to be used for the serial I/O.

● In the clock synchronous serial I/O
- "0": The BRG output divided by 4 becomes a shift clock.
- "1": The external clock (P16/SCLK pin input) becomes a synchronous clock.

● In the UART
- "0": The BRG output divided by 16 becomes a shift clock.
- "1": The external clock (P16/SCLK pin input) divided by 16 becomes a synchronous clock.

■ **$\overline{\text{SRDY}}$ output enable bit (SRDY, bit 2)**
This bit selects whether the P17/$\overline{\text{SRDY}}$ pin is used as P17 or as serial I/O pin $\overline{\text{SRDY}}$ or $\overline{\text{SRDY}}$ output disable.

● In the clock synchronous serial I/O
- "0": $\overline{\text{SRDY}}$ pin output disable (used as port P17)
- "1": $\overline{\text{SRDY}}$ pin output enable (used as serial I/O pin $\overline{\text{SRDY}}$)

● In the UART
The P17/$\overline{\text{SRDY}}$ pin is used as P17 regardless of the value of this bit.

■ **Transmit interrupt request selection bit (TIC, bit 3)**
This bit determines a source for generating a transmit interrupt request.
- "0": When the contents of the Transmit buffer register are transferred to the Transmit shift register, a transmit interrupt request is generated.
- "1": When the shift operation of the Transmit shift register terminates, a transmit interrupt request is generated.

■ **Transmit enable bit (TE, bit 4)**
This bit controls a transmit operation.

● When the serial I/O enable bit (bit 7) is "0" (serial I/O disable)
The transmit enable bit is invalid.

● When the serial I/O enable bit (bit 7) is "1" (serial I/O disable)
The control shown in Table 1.13B.9 is exerted.

Table 1.13B.9 Transmit enable bit function

Transmit enable bit	P15/TXD pin function	Transmit buffer empty flag*1	Transmit shift completion flag*2
0	Port P15	Cleared to "0"	Cleared to "0"
1	Serial I/O data transmit pin TXD	Flag function is valid.	Flag function is valid.

*1: Bit 0 of Serial I/O status register
*2: Bit 2 of Serial I/O status register

■ **Receive enable bit (bit 5)**

This bit controls the receive operation.

- When the serial I/O enable bit (bit 7) is "1" (serial I/O enable), the control shown in Table 1.13B.10 is exerted.
- When the serial I/O enable bit (bit 7) is "0" (serial I/O disable), this bit is invalid.

Table 1.13B.10 Receive enable bit function

Receive enable bit	P14/RxD pin function	Receive buffer full flag*1	Each error flag*2
0	Port P14	Cleared to "0"	Cleared to "0"
1	Serial I/O data transmit pin RxD	Flag function is valid.	Flag function is valid.

*1: Bit 1 of Serial I/O status register
*2: Bit 3,4,5 and 6 of Serial I/O status register

■ **Serial I/O mode selection bit (bit 6)**

This bit selects the clock synchronous serial I/O or the UART.

- "0": UART
- "1": Clock synchronous serial I/O

■ **Serial I/O enable bit (bit 7)**

This bit selects whether each of the P14/RxD, P15/TxD, P16/SCLK and P17/$\overline{\text{SRDY}}$ pins is used as a port or a serial I/O pin.

When using the serial I/O, set this bit to "1."

- "0": The respective pins become P14 to P17.
- "1": The respective pins become serial I/O pins, RxD, TxD, SCLK, $\overline{\text{SRDY}}$.

Note: In the UART, when the internal clock is selected, the P16/SCLK pin can be used as port P16. However, for the P17/$\overline{\text{SRDY}}$ pin, take the following points into consideration.

■ **In the clock asynchronous serial I/O**

When using the P17/$\overline{\text{SRDY}}$ pin as serial I/O pin $\overline{\text{SRDY}}$, set the $\overline{\text{SRDY}}$ output enable bit (bit 2) to "1."

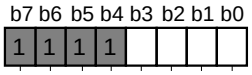
■ **In the UART**

The P17/$\overline{\text{SRDY}}$ pin is used as P17 regardless of the value of the serial I/O enable bit.

(4) UART control register (UARTCON: Address 00E3$_{16}$)

The UART control register controls the UART transfer data format.
Figure 1.13B.16 shows a structure of UART control register.

UART control register (7477/7478 group)



UART control register (UARTCON) [Address 00E3$_{16}$]

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 to 7	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "1."		1	1	×

Fig. 1.13B.16 Structure of UART control register

Each bit of the UART control register is described below.

■ **Character length selection bit (CHAS, bit 0)**

This bit selects a data bit length of the UART transfer data format.
- "0": 8-bit length
- "1": 7-bit length

■ **Parity enable bit (PARE, bit 1)**

This bit selects whether a parity check is made.
- "0": No parity check (Parity error flag is invalid.)
- "1": Parity check (Parity error flag is valid.)

■ **Parity selection bit (PARS, bit 2)**

This bit selects a parity type of the UART transfer data format.
- "0": Even
- "1": Odd

■ **Stop bit length selection bit (STPS, bit 3)**

This bit selects a stop bit length of the UART transfer data format.
- "0": 1-stop bit
- "1": 2-stop bit

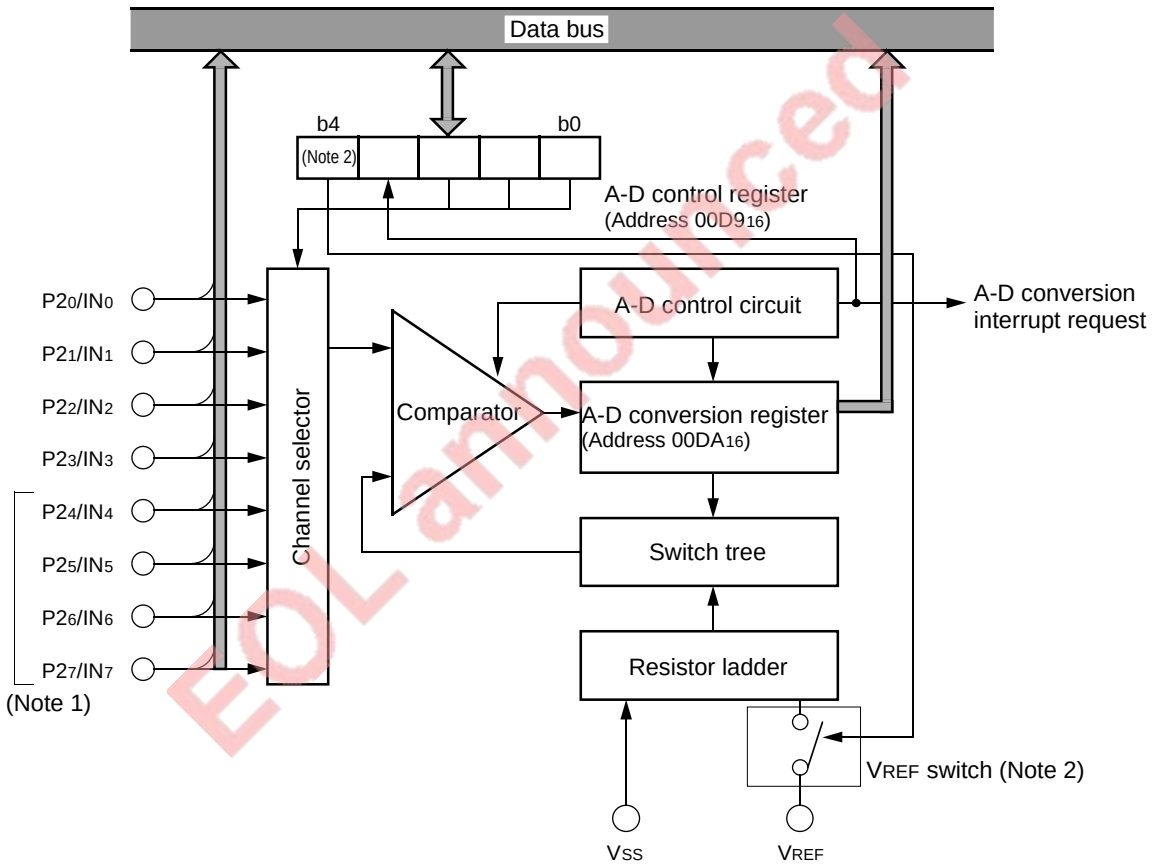
1.14 A-D converter

In the 7470/7471/7477/7478 group, the following A-D converter is incorporated.

- Analog input pins……7470/7477 group: 4 channels (in common with Port P2)
  7471/7478 group: 8 channels (in common with Port P2)
- Conversion method… Successive approximation comparison

In the 7470/7471 group, when the A-D converter is not used, power dissipation can be suppressed by the VREF switch. (The 7477/7478 group is not provided with this function.)

Figure 1.14.1 shows a block diagram of the A-D converter.



Notes 1. The 7470/7477 group is not provided with the P24/IN4–P27/IN7 pins.
2. The 7477/7478 group is not provided with the VREF switch and the VREF connection selection bit (bit 4) of the A-D control register.

Fig. 1.14.1 A-D converter block diagram

1.14 A-D converter

1.14.1 A-D conversion method

The A-D conversion method is of successive approximation comparison.
The reference voltage V_{ref} generated internally is compared with the analog input voltage V_{IN} which is input from the analog input pin ($P2_0/IN_0$–$P2_7/IN_7$), and its result is stored into each bit of the A-D conversion register (address $00DA_{16}$) successively to obtain a digital value.

[Internal operation]

After A-D conversion is started, the following operations are automatically performed.

① The contents of the A-D conversion register is set to "00_{16}."
② The most significant bit (bit 7) of the A-D conversion register is set to "1."
③ The reference voltage V_{ref} is input to the comparator. The reference voltage V_{ref} is specified by the contents n of the A-D conversion register and the reference voltage V_{REF} input from the V_{REF} pin.
An expression for the reference voltage V_{ref} is shown below.

Relational expression between V_{ref} and V_{REF}

When n = 0
$V_{ref} = 0$

When n = 1 to 255
$V_{ref} = V_{REF}/256 \times (n - 0.5)$
n: The values of A-D conversion register (decimal notation)

④ The reference voltage V_{ref} and the analog input voltage V_{IN} are compared with 8 times. Upon completion of each comparison, the comparison result is stored into the A-D conversion register. As the A-D conversion register changes, the reference voltage V_{ref} changes.

[1] Determination of the most significant bit (bit 7) of the A-D conversion register (in the 1st comparison)
The reference voltage V_{ref} and the analog input voltage V_{IN} are compared. Bit 7 is determined according to its result as follows.

If $V_{ref} < V_{IN}$, then bit 7 = "1."
If $V_{ref} > V_{IN}$, then bit 7 = "0."

[2] Determination of the bit 0 - 6 of the A-D conversion register (after the 2nd comparison).
First, bit 6 of the A-D conversion register is set to "1." Next, the reference voltage V_{ref} is compared with the analog input voltage V_{IN}. Bit 6 is determined according to its result as follows.

If $V_{ref} < V_{IN}$, then bit 6 = "1."
If $V_{ref} > V_{IN}$, then bit 6 = "0."

Likewise, bit 5 to bit 0 are determined according to comparison results of the 3rd to 8th comparisons.
A digital value (contents of the A-D conversion register) corresponding to the analog input voltage VIN is determined bit by bit by these operations.
Figure 1.14.2 shows the changes of the contents of the A-D conversion register and the reference voltage during A-D conversion.

⑤ After completion of A-D conversion, bit 3 of the A-D control register is set to "1" and an interrupt request is generated concurrently with the completion of A-D conversion.

Notes 1: An A-D conversion result can be obtained by reading the A-D conversion register after bit 3 of the A-D control register is set to "1."
2: The A-D conversion result is held in the A-D conversion register until bit 3 of the A-D control register is set to "1" again after completion of the next A-D conversion.

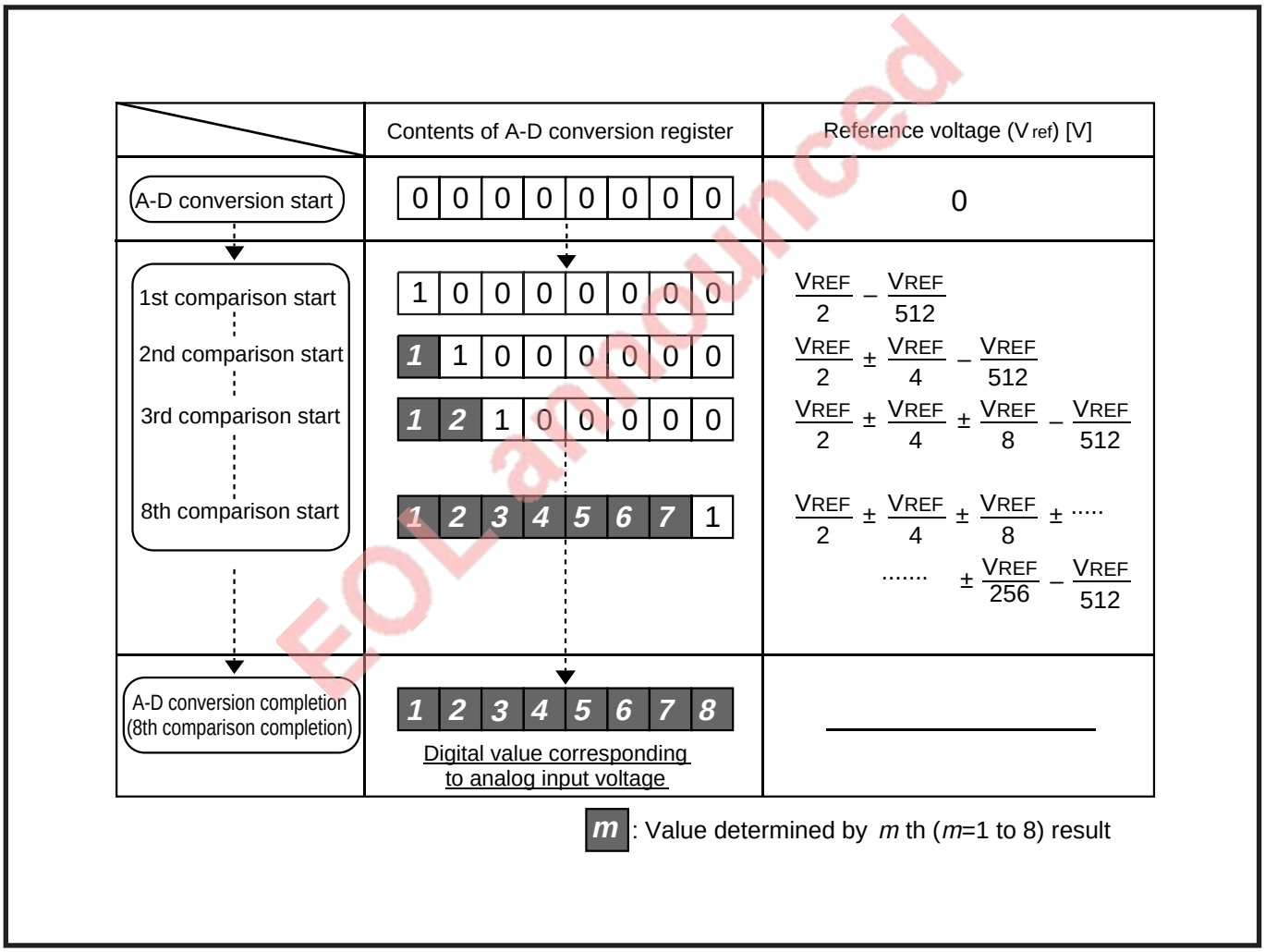


Fig 1.14.2 Contents of A-D conversion register and reference voltage during A-D conversion

1.14 A-D converter

■ Conversion time

- After a start of A-D conversion, this A-D conversion terminates after 50 cycles (12.5 μs at f(XIN) = 8 MHz).
- Main clock input oscillation frequency f(XIN)/2 is used as an operating clock for the A-D converter, so the A-D conversion time can be basically obtained by the following expression.

$$\text{A-D conversion time} = \frac{2}{f(X_{IN})} \times \text{conversion cycle (50: cycles)}$$

Note: Because the comparator is configurated by capacity coupling, use the A-D converter in the following condition.

$$f(X_{IN}) \geqq 1 \text{ MHz}$$

Accordingly, use the A-D converter in the condition that bit 7 of the CPU mode register (address 00FB16) is "0" (ordinary mode).

[Setting method]

In the 7470/7471 group

① Clear the bit of the Port P2 direction register corresponding to the used analog input pin to "0" (input mode).
② Clear the port pull-up control bit corresponding to the used analog input pin to "0" (no pull-up).
③ Clear the A-D conversion interrupt request bit of the Interrupt request register 1 to "0."
Note: After A-D conversion is started, the A-D conversion interrupt request bit is not cleared to "0" automatically.
④ When using an A-D conversion interrupt, set the A-D conversion interrupt enable bit to "1" to provide an interrupt enable state.
⑤ Set the A-D control register as follows.
- Select an analog input pin by the analog input pin selection bit.
- Set the VREF connection selection bit to "1" and connect VREF to a ladder resistor.

⑥ Wait for 1.0 μs or more as VREF stabilizing time.
⑦ Clear the A-D conversion end bit of the A-D control register to "0." (With this setting, A-D conversion is started.)

In the 7477/7478 group

① Clear the A-D conversion interrupt request bit of the Interrupt request register 1 to "0."
Note: After A-D conversion is started, the A-D conversion interrupt request bit is not cleared to "0" automatically.
② When using an A-D conversion interrupt, set the A-D conversion interrupt enable bit to "1" to provide an interrupt enable state.
③ Set the A-D control register as follows.
- Select an analog input pin by the analog input pin selection bit.
- Clear the A-D conversion end bit to "0." (With this setting, A-D conversion is started.)

Don't read the contents of the A-D conversion register during A-D conversion.
For register setting, refer to **"Table 1.14.1 Setting at A-D Conversion."**

Processing after conversion

① A termination of conversion can be verified by any of the following operations.
- State of A-D conversion end bit.
- State of A-D conversion interrupt request bit.
- Branch to A-D conversion interrupt routine.

② Read the A-D conversion register to obtain a conversion result.

Notes 1: Be sure to connect V_{REF} to a ladder resistor during A-D conversion (7470/7471 group).
2: A-D conversion is restarted at the time when the A-D conversion end bit (bit 3) of the A-D control register is cleared to "0" during A-D conversion.

Table 1.14.1 Setting at A-D conversion

Register	Bit	Value
① Port P2 direction register(P2D: Address $00C5_{16}$) (7470/7471 group)	b0 ≀ b7	Clear the bit corresponding to the used analog input pin (one of pins $P2_0$/IN_0 to $P2_7$/IN_7) to "0" (input mode). **Note:** In the 7470 group, only pins $P2_0$/IN_0 to $P2_3$/IN_3 are available.
② Port P1-P4 pull-up control register (7470 group) Port P1-P5 pull-up control register (7471 group) (Address $00D1_{16}$)	b2	0: $P2_0$ to $P2_3$ are not pulled up. **Note:** When one of pins $P2_0$/IN_0 to $P2_3$/IN_3 is used as an analog input pin
	b3	0: $P2_4$ to $P2_7$ are not pulled up. **Note:** When one of pins $P2_4$/IN_4 to $P2_7$/IN_7 is used as an analog input pin (7471 group)
③ Interrupt request register 1 (IR1: Address $00FC_{16}$)	b7	0: A-D conversion interrupt disabled
④ Interrupt control register 1 (IE1: Address $00FE_{16}$)	b7	1: A-D conversion interrupt enabled
⑤ A-D conversion control register (ADCON: Address $00D9_{16}$)	b2, b1, b0	• 000: $P2_0$/IN_0 • 001: $P2_1$/IN_1 • 010: $P2_2$/IN_2 • 011: $P2_3$/IN_3 • 100: $P2_4$/IN_4 • 101: $P2_5$/IN_5 • 110: $P2_6$/IN_6 • 111: $P2_7$/IN_7 Set the value corresponding to the used analog input pin. **Note:** In the 7470/7477 group, only pins $P2_0$/IN_0 to $P2_3$/IN_3 are available.
	b3	0: During conversion (A-D conversion is started.)
	b4	1: V_{REF} connection (7470/7471 group)
	b7	Fix this bit to "0."

1.14 A-D converter

1.14.2 Pins

The pins used in the A-D converter are described below.

(1) Analog input pin ($P2_0/IN_0$ to $P2_7/IN_7$) ($P2_0/IN_0$ to $P2_3/IN_3$ in the 7470/7477 group)

- The analog input pin is an input pin for analog voltage.
- Apply a voltage of V_{SS} (AV_{SS}) - V_{REF} to this pin.
- This pin is used in common with $P2_0$ to $P2_7$ ($P2_0$ to $P2_3$ in the 7470/7477 group).
  When using the A-D converter, select a pin to be used as an analog input pin by bit 2 to bit 0 of the A-D control register (ADCON: address $00D9_{16}$).
  Use the A-D converter in the following condition in the 7470/7471 group.
  - When the bit of the Port P2 direction register, which corresponds to the used analog input pin, is “0” (input mode)
  - When the bit of the Pull-up control register, which corresponds to the used analog input pin, is “0” (no pull-up)

(2) Reference voltage input pin (V_{REF})

- The reference voltage input pin is an input pin for reference voltage.
- Input a voltage of 0.5 V_{CC} (≧2) - V_{CC} [V].

(3) Analog power source input pin (AV_{SS})

- Analog power source input pin is an input pin for GND.
- Apply the same potential as the V_{SS} pin to this pin.
- This pin is dedicated to the 56P6N-A package product of the 7471/7478 group.

1.14.3 Notes on use

When using the A-D converter, take the following points into consideration.

■ The comparator is configured by capacity coupling, so the charge is lost if the clock input oscillation frequency is low.
- Set f(X_{IN}) at 1 MHz or more during A-D conversion.
- Don’t execute the STP instruction during A-D conversion.

■ Apply a voltage of 0.5 V_{CC} (≧2) - V_{CC} [V] to the reference voltage input pin V_{REF}.
Note that if the reference voltage is lowered below the above valve, the A-D conversion precision will be degraded.

■ Apply the same potential as that of the V_{SS} pin to the analog power supply voltage input pin AV_{SS}.
The AV_{SS} pin is dedicated to the 56P6N-A package product of the 7471/7478 group.

■ In the 7470/7471 group, clear the bit of the Port P2 direction register which corresponds to the used analog input pin to “0” (input mode).

■ In the 7470/7471 group, clear the bit of the Pull-up control register which corresponds to the used analog input pin to “0” (no pull-up).

1.14.4 References

■ Definition of A-D conversion precision
The definition of A-D conversion precision is described.
Refer to the definition of A-D conversion precision shown in Figure 1.14.3.

(1) Relative precision

● **Zero transition error (V_{OT})**
Deviation of the input voltage, at which A-D conversion output data changes from "0" to "1," from the ideal A-D conversion characteristics between 0 and V_{REF}

$$V_{OT} = (V_O - \frac{1}{2} \times \frac{V_{REF}}{256})/1LSB \text{ [LSB]}$$

● **Full-scale transition error (V_{FST})**
Deviation of the ideal A-D conversion characteristics between 0 and V_{REF} of the input voltage when the A-D conversion output data changes from "255" to "254".

$$V_{FST} = \{(V_{REF} - \frac{3}{2} \times \frac{V_{REF}}{256}) - V_{254}\}/1LSB \text{ [LSB]}$$

● **Non-linearity error**
Deviation of the real A-D conversion characteristics from the ideal characteristics between V_0 and V_{254}

$$\text{Non-linearity error} = \{V_n - (1LSB \times n + V_O)\}/1LSB \text{ [LSB]}$$

● **Differential non-linearity error**
Deviation of the input voltage required to change output data by "1" from the ideal characteristics between V_0 and V_{254}

$$\text{Differential non-linearity error} = \{(V_{n+1} - V_n) - 1LSB\}/1LSB \text{ [LSB]}$$

(2) Absolute precision

● **Absolute precision**
Deviation of the real A-D conversion characteristic from the ideal characteristics between 0 and V_{REF}.

$$\text{Absolute precision} = \{V_n - 1LSB \times (n + \frac{1}{2})\}/1LSB \text{ [LSB]}$$

1.14 A-D converter

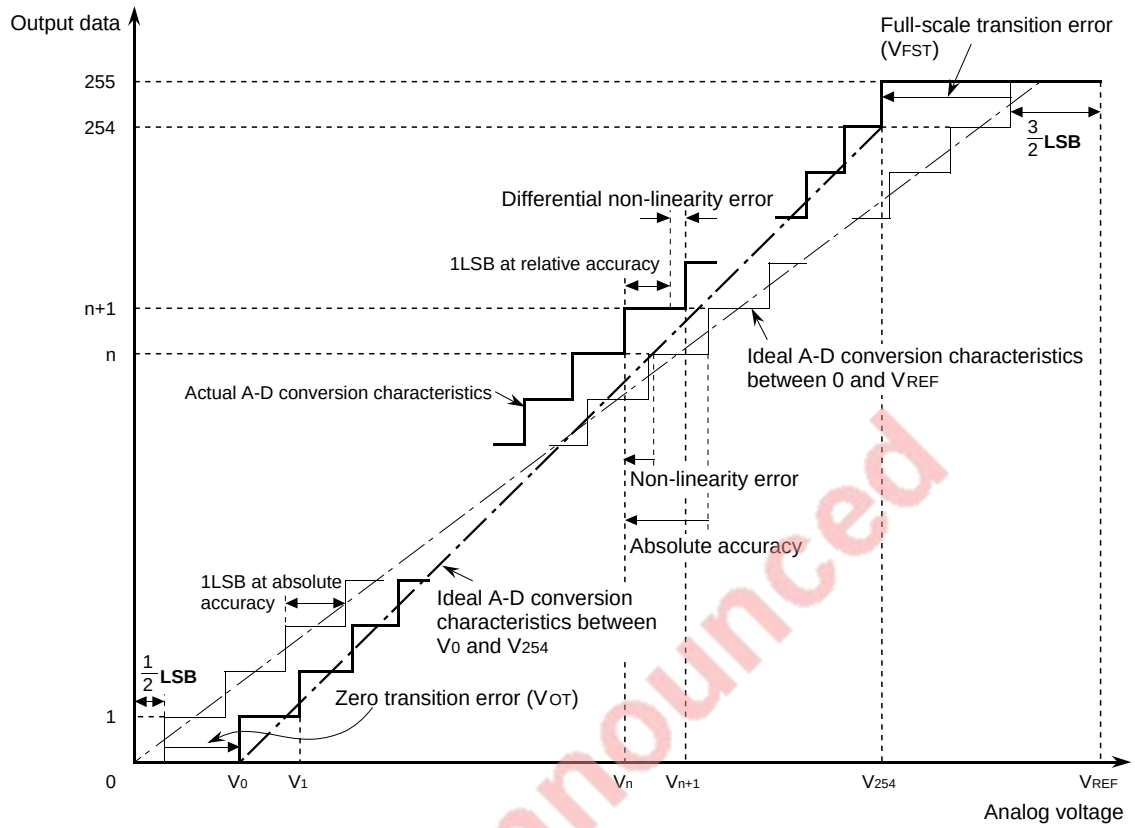


Fig. 1.14.3 Definition of A-D conversion precision

Vn: Analog input voltage when output data changes from "n" to "n + 1" (n = 0 – 254).

- 1LSB = $\frac{V_{254} - V_O}{254}$ (V) → 1LSB at relative precision
- 1LSB = $\frac{V_{REF}}{256}$ (V) → 1LSB at absolute precision

1.14.5 Related registers

(1) A-D conversion register (AD: Address 00DA16)

The A-D conversion register stores A-D conversion results. This register is a read-only type.

Figure 1.14.4 shows a structure of the A-D conversion register.

A-D conversion register

b7 b6 b5 b4 b3 b2 b1 b0

A-D conversion register (AD) [Address 00DA16]

B	Function	At reset	R	W
0 to 7	This is a read-only register to store A-D conversion results.	?	○	×

Fig. 1.14.4 Structure of A-D conversion register

(2) A-D control register (ADCON: Address 00D916)

The A-D control register consists of bits that exerts various types of control over the A-D converter.

Figure 1.14.5 shows a structure of the A-D control register.

1.14 A-D converter

A-D control register

b7	b6	b5	b4	b3	b2	b1	b0
0							

A-D control register (ADCON) [Address 00D9$_{16}$]

B	Name	Function	At reset	R	W
0	A-D input selection bits	b2 b1 b0 0 0 0 : P2$_0$/IN$_0$ 0 0 1 : P2$_1$/IN$_1$ 0 1 0 : P2$_2$/IN$_2$ 0 1 1 : P2$_3$/IN$_3$ 1 0 0 : P2$_4$/IN$_4$ 1 0 1 : P2$_5$/IN$_5$ 1 1 0 : P2$_6$/IN$_6$ 1 1 1 : P2$_7$/IN$_7$ (Note 1)	0	○	○
1			0	○	○
2			0	○	○
3	A-D conversion end bit (Note 2)	0 : Under conversion 1 : End conversion	1	○	○
4	V$_{REF}$ connection selection bit ✻The 7477/7478 group is not provided with this bit. This bit is undefined at reset.	0 : The VREF pin is separated from the comparison voltage generator. 1 : The VREF pin is connected to comparison voltage generator.	0	○	○
5, 6	Nothing is allocated for these bits. These are write disabled bits and are undefined at reading.		?	?	×
7	Fix this bit to “0.”		0	0	0

Notes 1: Since the 7470/7477 group is not provided with pins P2$_4$–P2$_7$, do not set.

2: •A-D conversion is started by setting bit 3 to “0.”
•Writing “0” into bit 3 is valid. Even if “1” is written into bit 3, this bit is not set to “1.” Accordingly, when writing a value into the A-D control register without affecting bit 3, set bit 3 to “1.”

Fig. 1.14.5 Structure of A-D control register

Each bit of the A-D control register is described below.

■ **Analog input pin selection bit (Bit 2 to 0)**
These bits select an analog input pin.
Pins that are not used as analog input pins of P2 function as programmable I/O ports (input ports in the 7477/7478 group).

■ **A-D conversion end bit (Bit 3)**
This bit indicates the operation state of the A-D converter.
This bit is cleared to “0” during A-D conversion and set to “1” upon termination of A-D conversion. A-D conversion is started by clearing this bit to “0.” (At the time when the bit is cleared to “0” during A-D conversion, A-D conversion is restarted.)

■ **V$_{REF}$ connect selection bit (Bit 4)** (The 7477/7478 group is not provided with this bit.)
This bit connects the V$_{REF}$ pin to a ladder resistor.
When using the A-D converter, be sure to set this bit to “1.”
When the A-D converter is not used, power consumption can be reduced by clearing this bit to “0.”

1.15 Reset

The microcomputer is reset by applying the “L” level to the $\overline{\text{RESET}}$ pin for 2 μs or more when the power source voltage is within the standard value range. After that, when the “H” level is applied to the $\overline{\text{RESET}}$ pin, the reset state of the microcomputer is released, so that the program is run starting with the reset vector address.

1.15.1 Operation description

Figure 1.15.1 shows an internal processing sequence after reset release.

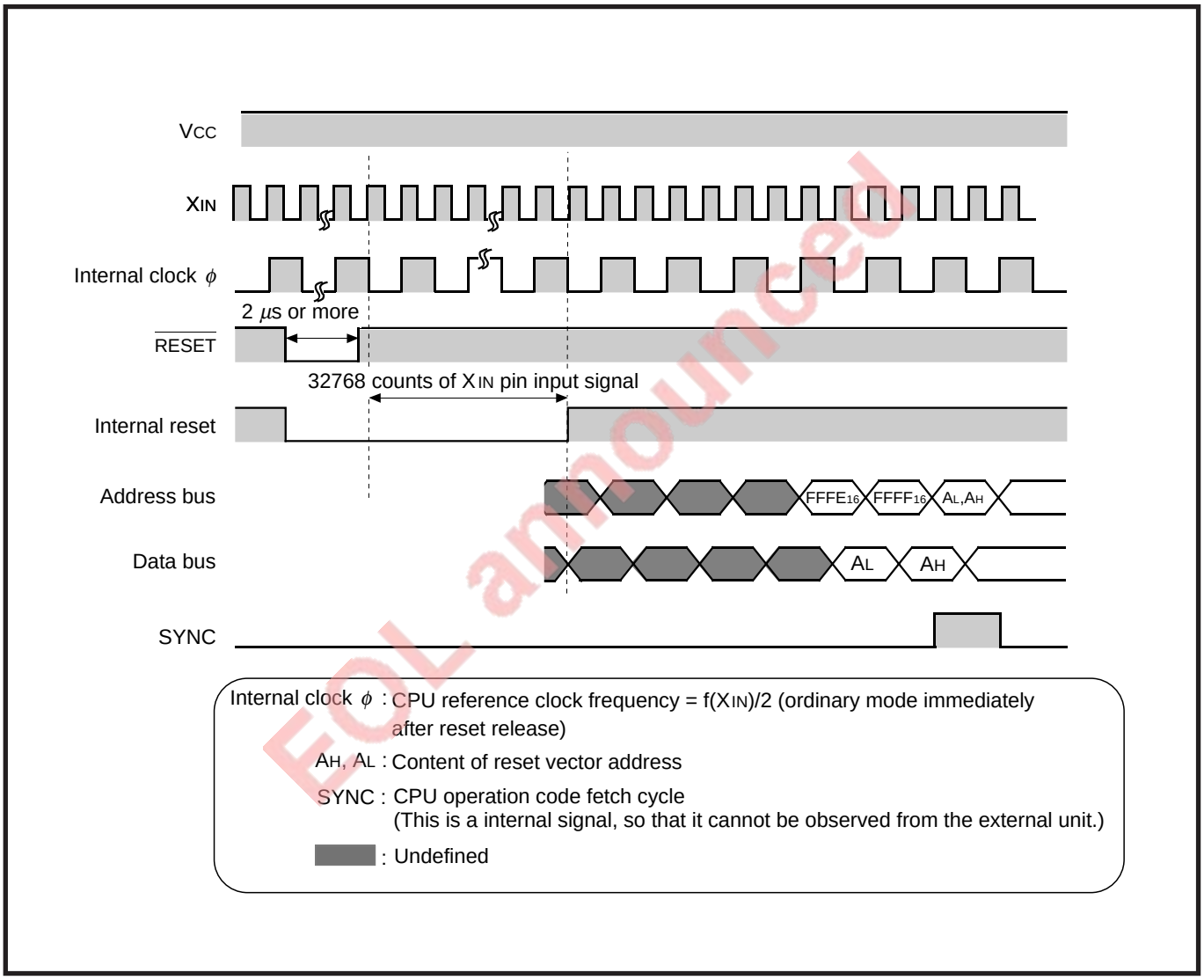


Fig. 1.15.1 Internal processing sequence after reset release

1.15 Reset

When the “H” level is applied to the $\overline{\text{RESET}}$ pin at the reset state, the contents of timer 3 and timer 4 and the count source are automatically set as shown in Table 1.15.1, so that the internal reset state is released by an overflow of timer 4.

Table 1.15.1 Timer 3 and 4 at reset

Item	Timer 3	Timer 4
Setting value	FF_{16}	07_{16}
count source	f(XIN)/16	overflow of timer 3

After the “H” level is applied to the $\overline{\text{RESET}}$ pin, only the main clock oscillates regardless of the oscillation state precedent to the reset state, so that the microcomputer starts to operate in the ordinary mode. The XCIN pin and the XCOUT pin become $P5_0$ and $P5_1$, respectively.
After the reset state is released, the microcomputer runs the program starting with the high-order address corresponding to the contents of address $FFFF_{16}$ and the low-order address corresponding to the contents of address $FFFE_{16}$.

Note: The 7470/7477 group is not provided with the XCIN and XCOUT pins.

1.15.2 Internal status immediately after reset release

Figure 1.15.2 shows an internal register status immediately after reset release.

		Address	
(1)	Port P0 direction register (P0D)	($C1_{16}$)•••	00_{16}
(2)	Port P1 direction register (P1D)	($C3_{16}$)•••	00_{16}
(3)	Port P2 direction register (P2D) (The 7477/7478 group is not provided.)	($C5_{16}$)•••	00_{16}
(4)	Port P4 direction register (P4D)	($C9_{16}$)•••	□ □ □ □ 0 0 0 0
(5)	Port P0 pull-up control register	($D0_{16}$)•••	00_{16}
(6)	Port P1–P5 pull-up control register (In the 7470/7477 group, port P1–P4 pull-up control register)	($D1_{16}$)•••	□ 0 □ 0 0 0 0 0
(7)	Edge polarity selection register (EG)	($D4_{16}$)•••	□ □ 0 0 0 0 0 0
(8)	A-D control register (ADCON)	($D9_{16}$)•••	0 □ □ 0 1 0 0 0
(9)	Serial I/O mode register (SM) (The 7477/7478 group is not provided.)	(DC_{16})•••	00_{16}
(10)	Serial I/O status register (SIOSTS) (The 7470/7471 group is not provided.)	($E1_{16}$)•••	1 0 0 0 0 0 0 0
(11)	Serial I/O control register (SIOCON) (The 7470/7471 group is not provided.)	($E2_{16}$)•••	00_{16}
(12)	UART control register (UARTCON) (The 7470/7471 group is not provided.)	($E3_{16}$)•••	1 1 1 1 0 0 0 0
(13)	Timer 3 (T3)	($F2_{16}$)•••	FF_{16}
(14)	Timer 4 (T4)	($F3_{16}$)•••	07_{16}
(13)	Timer FF register (TF)	($F7_{16}$)•••	□ □ □ □ □ □ 0 0
(14)	Timer 12 mode register (T12M)	($F8_{16}$)•••	00_{16}
(15)	Timer 34 mode register (T34M)	($F9_{16}$)•••	00_{16}
(16)	Timer mode register 2 (TM2)	(FA_{16})•••	0 0 □ □ □ □ 0 0
(17)	CPU mode register (CPUM)	(FB_{16})•••	0 0 0 0 □ 0 0 0
(18)	Interrupt request register 1 (IR1)	(FC_{16})•••	0 0 0 □ 0 0 0 0
(19)	Interrupt request register 2 (IR2)	(FD_{16})•••	□ □ □ □ □ 0 0 0
(20)	Interrupt control register 1 (IE1)	(FE_{16})•••	0 0 0 □ 0 0 0 0
(21)	Interrupt control register 2 (IE2)	(FF_{16})•••	□ □ □ □ □ 0 0 0
(22)	Program counter (PC_H)		Contents of address $FFFF_{16}$
	(PC_L)		Contents of address $FFFE_{16}$
(23)	Processor status register (PS)		□ □ □ □ □ 1 □ □

□ : The contents are undefined at reset release.

Note : Since the contents of the registers and RAM not mentioned above are undefined at reset, initialize them by software.
The bits are different depending on the product. Refer to the structure of each register.

Fig. 1.15.2 Internal status immediately after reset release

1.15 Reset

1.15.3 Notes on use

1. The timer continues to perform a count operation after reset release.

2. After the reset state is released, the microcomputer runs the program starting with the high-order address corresponding to the contents of address FFFF16 and the low-order address corresponding to the contents of address FFFE16.

3. After the "H" level is applied to the $\overline{\text{RESET}}$ pin, only the main clock oscillates regardless of the oscillation state precedent to the reset state, so that the microcomputer starts to operate in the ordinary mode. The XCIN pin and the XCOUT pin become P50 and P51, respectively.
(The 7470/7477 group is not provided with the XCIN and XCOUT pins.)

4. When the STP instruction is executed in the ordinary mode, I/O ports are held in the state just precedent to a stop of system clock oscillation. After that, the I/O ports are put into the input mode after a reset operation of the microcomputer is performed, so that they go to the high-impedance state.

1.16 Oscillation circuit

The 7470/7471/7477/7478 group has the following circuits to obtain clocks required for operations.

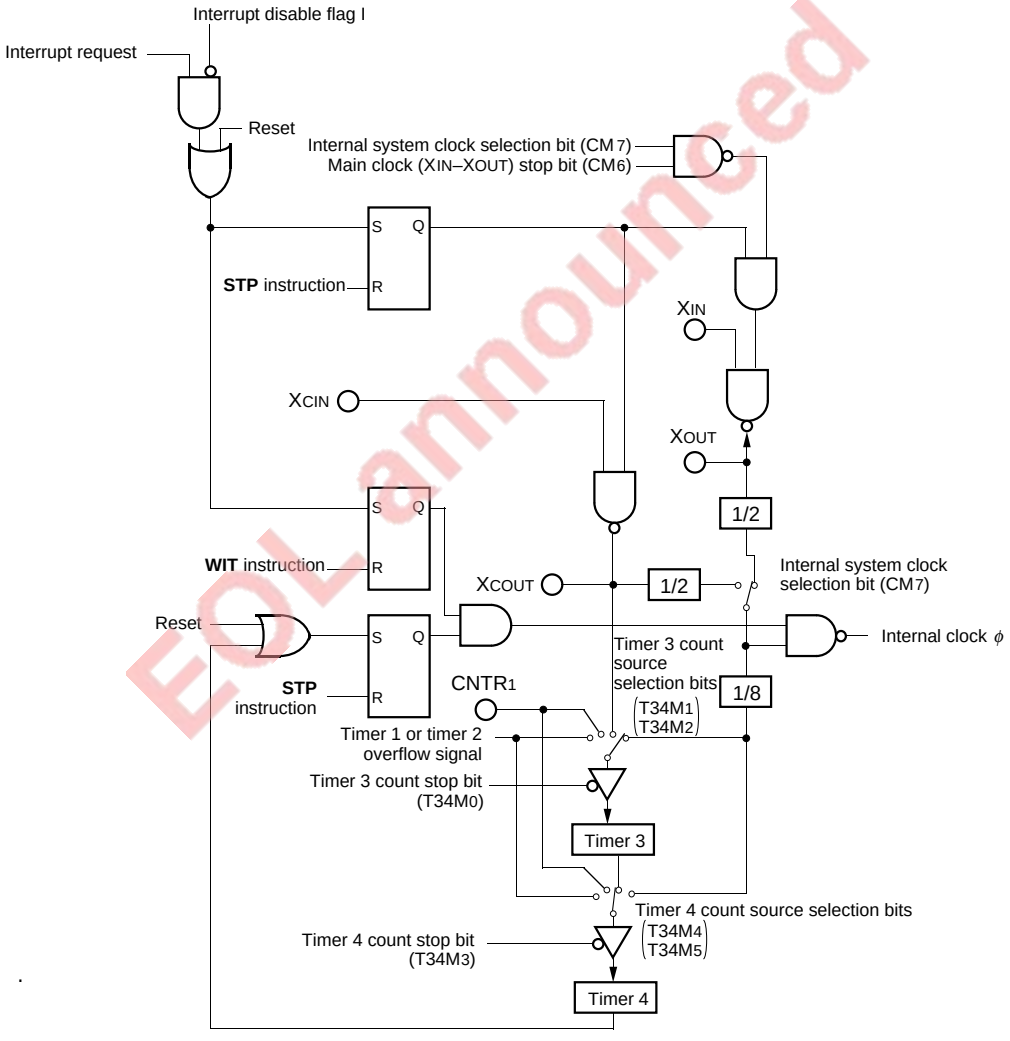
- 7470/7477 group: Main clock oscillation circuit
- 7471/7478 group: Main clock oscillation circuit and sub-clock oscillation circuit

1.16.1 Oscillation circuit

(1) Clock generating circuit

The clock generating circuit controls the oscillation of the oscillation circuit and a generated clock (internal clock ϕ) is supplied to the CPU and peripheral units.

Figure 1.16.1 shows a clock generating circuit block diagram.



Note : The 7470/7477 group is not provided with pins XCIN and XCOUT.

Fig. 1.16.1 Clock generating circuit block diagram

1.16 Oscillation circuit

This oscillation circuit can stop and start oscillation.

- XIN-XOUT oscillation circuit ..... Main clock f(XIN)
- XCIN-XCOUT oscillation circuit ..... Sub-clock f(XCIN)

There are two oscillation circuits (the main clock only in the 7470/7477 group) as shown above. The clock obtained by dividing a signal input to the clock input pin XIN or XCIN becomes an internal clock ϕ^* which is used as a reference for operations.

$*$: The internal clock ϕ varies with the operation modes of the microcomputer.

- Ordinary mode ..... Signal input to the XIN pin divided by 2
- Low-speed mode..... Signal input to the XCIN pin divided by 2

(2) Oscillation circuit using a ceramic resonator or a crystal oscillator

An oscillation circuit can be formed by connecting a ceramic resonator or a crystal oscillator between the XIN pin and the XOUT pin and between the XCIN pin and the XCOUT pin.

For a circuit example, refer to **"Chapter 2 Application, 2.7 Oscillation Circuit."**

Please ask the oscillator maker for information on circuit constants and then set the value recommended by the maker.

(3) External clock input circuit

It is also possible to supply a clock to the oscillation circuit from the outside.

As an external clock to be input to the XIN and XCIN pins, use a pulse signal with a duty ratio of 50 %. At this time, make the XOUT and XCOUT pins open.

For a circuit example, refer to **"Chapter 2 Application, 2.7 Oscillation Circuit."**

Note: Because the 7470/7477 group is not provided with the XCIN and XCOUT pins, the sub-clock f(XCIN) is not available.

1.16.2 Sub-clock oscillation circuit

In the 7471/7478 group, the sub-clock f(XCIN) is available when the P50/XCIN pin and the P51/XCOUT pin are used as the XCIN pin and the XCOUT pin.

The power supplied to the sub-clock oscillation circuit is given through a voltage reduction regulator to reduce power dissipation in the sub-clock mode. That is, power is reduced by reducing the voltage applied to the VCC pin by the voltage reduction regulator. The supply voltage to this oscillation circuit can be set to one of the 2 stages of high power mode and low power mode in bit 5 of the CPU mode register.

Notes 1: When using the sub-clock, set f(XCIN) ≦ 50 kHz < f(XIN)/3.

2: When using the sub-clock f(XCIN) in the 7471/7478 group, set the P50-P53 pull-up control bit (bit 6) of the P1-P5 pull-up control register to "0" and disconnect the pull-up transistor of the P50/XCIN pin and P51/XCOUT pin.

3: When using the sub-clock as the internal clock ϕ, use it in one of the following states.

- Fix the XCOUT drive capacity to the high power mode (set the XCOUT drive capacity selection bit of the CPU mode register to "1").
- When fixing the XCOUT drive capacity to the Low power mode (set the XCOUT drive capacity selection bit of the CPU mode register to "0"), lower the value of the resistor Rd* in the sub-clock oscillation circuit to a level at which the oscillation of f(XCIN) does not stop.

* "Resistor Rd": Refer to the circuit example in **"Chapter 2 Application, 2.7 Oscillation circuit."**

1.16 Oscillation circuit

1.16.3 Oscillation operation

(1) Oscillation operation

The microcomputer is put into the ordinary mode at reset release. At this time, only the main clock oscillates and the P5$_0$/XCIN pin and the P5$_1$/XCOUT pin function as input ports P5$_0$ and P5$_1$.

Notes 1: The 7470/7477 group is not provided with XCIN and XCOUT pins.

2: When using the sub-clock f(XCIN) in the 7471/7478 group, set the P5$_0$-P5$_3$ pull-up control bit (bit 6) of the P1-P5 pull-up control register to "0" and disconnect the pull-up transistor of the P5$_0$/XCIN pin and P5$_1$/XCOUT pin.

■ **Ordinary mode**

The clock resulting from dividing a signal input to the XIN pin by 2 becomes internal clock ϕ.

Changing the mode to the low-speed mode (7471/7478 group)

Execute the following procedure.

① Set the P5$_0$, P5$_1$/XCIN, XCOUT selection bit (bit 4) of the CPU mode register to "1" (XCIN, XCOUT).
② Set the XCOUT drive capacity selection bit (bit 5) of the CPU mode register to "1" (High power).
③ Generate oscillation stabilizing wait time of f(XCIN) by software.
④ Set the system clock selection bit (bit 7) of the CPU mode register to "1" f(XCIN). At that time, set the XCOUT drive capacity selection bit to "0" (low power) as required.

■ **Low-speed mode (7471/7478 group)**

The clock resulting from dividing a signal input to the XCIN pin by 2 becomes internal clock ϕ. In the low-speed mode, a low power dissipation operation can be attained by setting the main clock (XIN-XOUT) stop bit (bit 6) of the CPU mode register to "1."

Changing the mode to the ordinary mode

Execute the following procedure.

① Clear the main clock (XIN-XOUT) stop bit (bit 6) of the CPU mode register to "0" (oscillate).
② Generate oscillation stabilizing wait time of f(XIN) by software.
③ Clear the system clock selection bit (bit 7) of the CPU mode register to "0" f(XIN).

Notes 1: Switch between the ordinary mode and the low-speed mode after the oscillation of the main clock and the sub-clock becomes stable. For the oscillation stablizing time, ask the oscillator maker for information.

2: Use the low-speed mode in one of the following states.

- Fix the XCOUT drive capacity to the high power mode (set the XCOUT drive capacity selection bit of the CPU mode register to "1").
- When fixing the XCOUT drive capacity to the Low power mode (clear the XCOUT drive capacity selection bit of the CPU mode register to "0"), lower the value of the resistor Rd* in the sub-clock oscillation circuit to a level at which the oscillation of f(XCIN) does not stop.

* Resister Rd: Refer to a example of circuit in **"chapter 2 application, 2.7 Oscillation circuit."**

3: When using the sub-clock especially, it takes a long time until the oscillation becomes stable. When the ordinary mode is changed to the stop mode while the sub-clock is in the oscillation state and then the ordinary mode is restored from the stop mode, the oscillation of the sub-clock is not stabilized even if the main clock becomes stable and the CPU is restored.

(2) Oscillation operation in the stop mode

After the stop mode is provided by execution the STP instruction, all oscillation stops. After that, when the previous mode is restored from the stop mode by inputting the reset signal or generating a restoration interrupt request, the oscillation starts.

For the details of the stop mode, refer to **"1.17.1 Stop mode."**

(3) Oscillation operation in the wait mode

When the wait mode is provided by execution the WIT instruction, the internal clock ϕ supplied to the CPU stops.

When the previous mode is restored from the wait mode by inputting the reset signal or generating an interrupt request, the supply of internal clock ϕ to the CPU starts.

For the details of the wait mode, refer to **"1.17.2 Wait mode."**

(4) State transitions of internal clock

Refer to **"1.18 State transitions."**

1.16 Oscillation circuit

1.16.4 Oscillation stabilizing time

In the oscillation circuit using a ceramic resonator or a crystal oscillator, the oscillation becomes unstable for a certain period at a start of oscillation of the oscillator. The time required for stabilization of oscillation is called oscillation stabilizing time.
A proper oscillation stabilizing wait time fit for the used oscillation circuit is required. For the oscillation stabilizing time, ask the oscillator maker for information.

(1) Oscillation stabilizing wait time at power on

In the 7470/7471/7477/7478 group, the oscillation stabilizing wait time for 32768 counts of the XIN pin input signal is automatically generated in the period from power on to reset release.

Figure 1.16.2 shows a oscillation stabilizing wait time after power on.

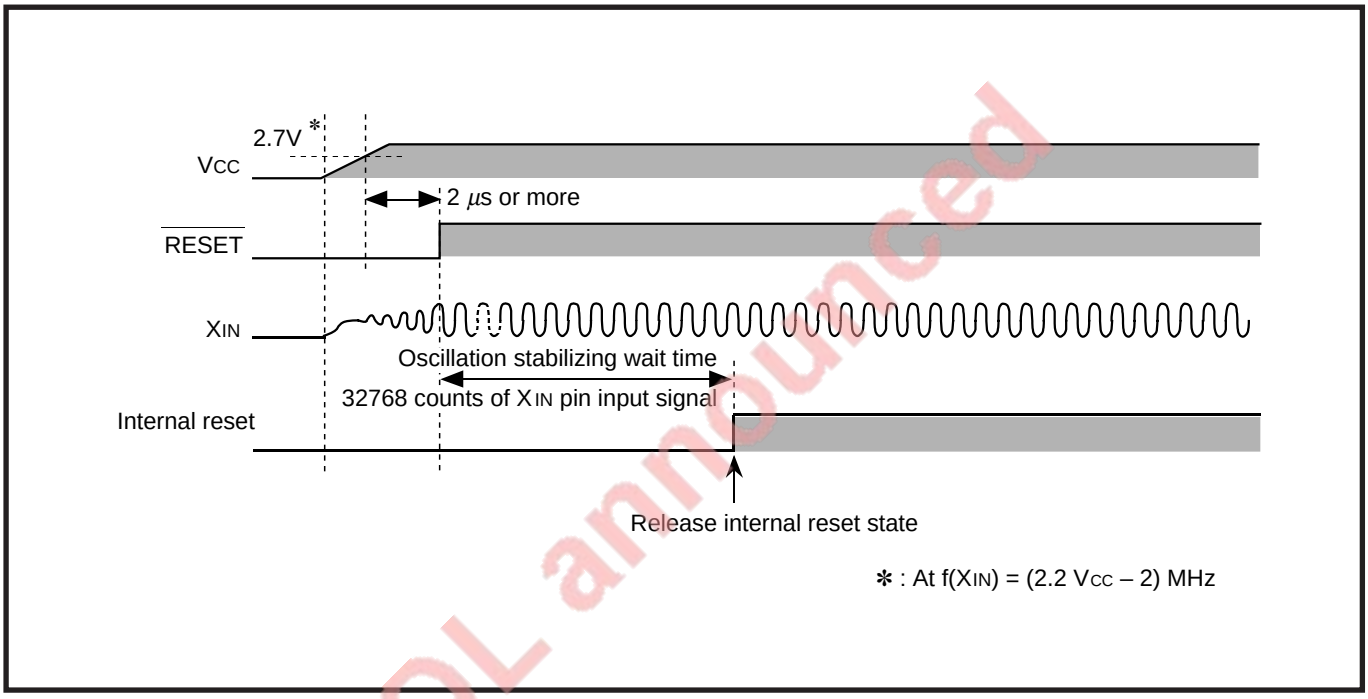


Fig. 1.16.2 Oscillation stabilizing wait time after power on

(2) Oscillation stabilizing wait time at recovery from stop mode

In the stop mode, oscillation stops. When the previous mode is restored from the stop mode by inputting a reset signal or generating an interrupt, the oscillation stabilizing wait time for 32768 counts of the XIN pin input signal or the XCIN pin input signal is automatically generated in the same way as the power on time.

- At recovery by reset, f(XIN) becomes a count source that generates oscillation stabilizing wait time.
- At recovery by interrupt, the count source of timer 3 set immediately before execution of the STP instruction becomes a count source that generates oscillation stabilizing wait time. Note that when f(XIN) is the system clock, the oscillation of the f(XCIN) side may not be stabilized after the lapse of this oscillation stabilizing wait time.

For the details of the stop mode, refer to **"1.17.1 Stop mode."**

Note: In the 7470/7477 group, f(XCIN) is not available.

1.16.5 Notes on use

1. When inputting the external clock to the XIN pin or the XCIN pin, use a pulse signal with a duty ratio of 50 % as an input signal. At this time, make the XOUT pin and the XCOUT pin open.
   Refer to a example of circuit in **"Chapter 2 application, 2.7 Oscillation circuit."**

2. When using the sub-clock f(XCIN) in the 7471/7478 group, set f(XCIN) ≦ 50 kHz < f(XIN)/3.

3. In the 7471/7478 group, switch between the ordinary mode and the low-speed mode after the oscillation of the main clock and the sub-clock becomes stable. For the oscillation stabilizing time, ask the oscillator maker for information.

4. Use the low-speed mode in one of the following states.
   - Fix the XCOUT drive capacity to the high power mode (Set the XCOUT drive capacity selection bit of the CPU mode register to "1").
   - When fixing the XCOUT drive capacity to the Low-power mode (clear the XCOUT drive capacity selection bit of the CPU mode register to "0"), lower the value of the resistor Rd* in the sub-clock oscillation circuit to a level at which the oscillation of f(XCIN) does not stop.

   * Resister Rd: Refer to a example of circuit in **"Chapter 2 application, 2.7 Oscillation circuit."**

5. When using the sub-clock f(XCIN) in the 7471/7478 group, it takes a long time until the oscillation becomes stable.
   When the ordinary mode is changed to the stop mode while the sub-clock is in the oscillation state and then the ordinary mode is recovered from the stop mode, the oscillation of the sub-clock is not stabilized even if the main clock becomes stable and the CPU is restored.

Note: In the 7470/7477 group, the sub-clock f(XCIN) is not available because neither XCIN pin nor XCOUT pin is provided.

1.17 Low-power dissipation function

The 7470/7471/7477/7478 group is provided with a function to put the CPU into a wait state with low-power dissipation by stopping the CPU operation by softoware.
The low-power dissipation function has the following 2 modes.

- Stop mode by a STP instruction
- Wait mode by a WIT instruction

Figure 1.17.1 shows the operation states of the microcomputer at low-power dissipation and Figure 1.17.2 shows a state transition.

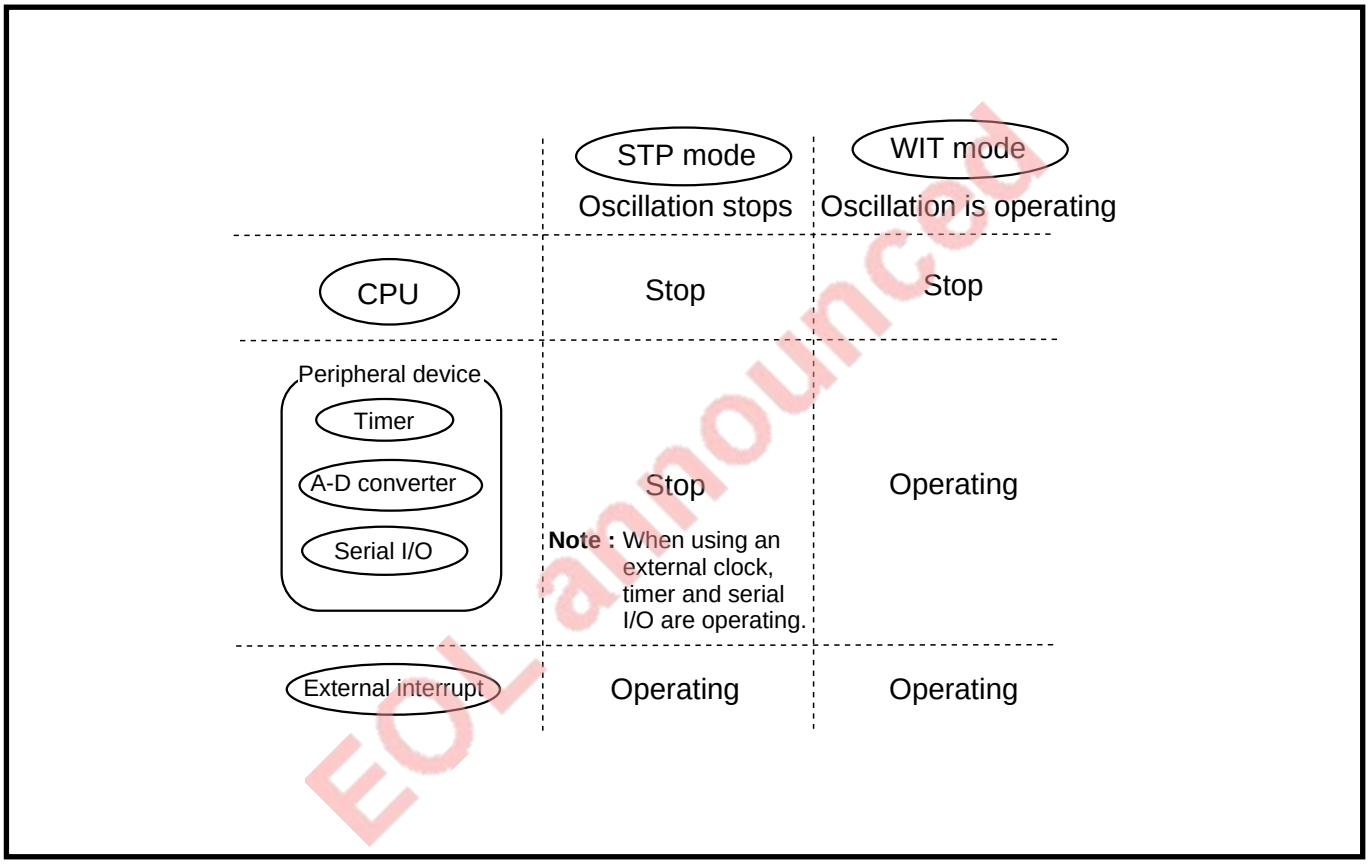


Fig. 1.17.1 Operation states of the microcomputer at low-power dissipation

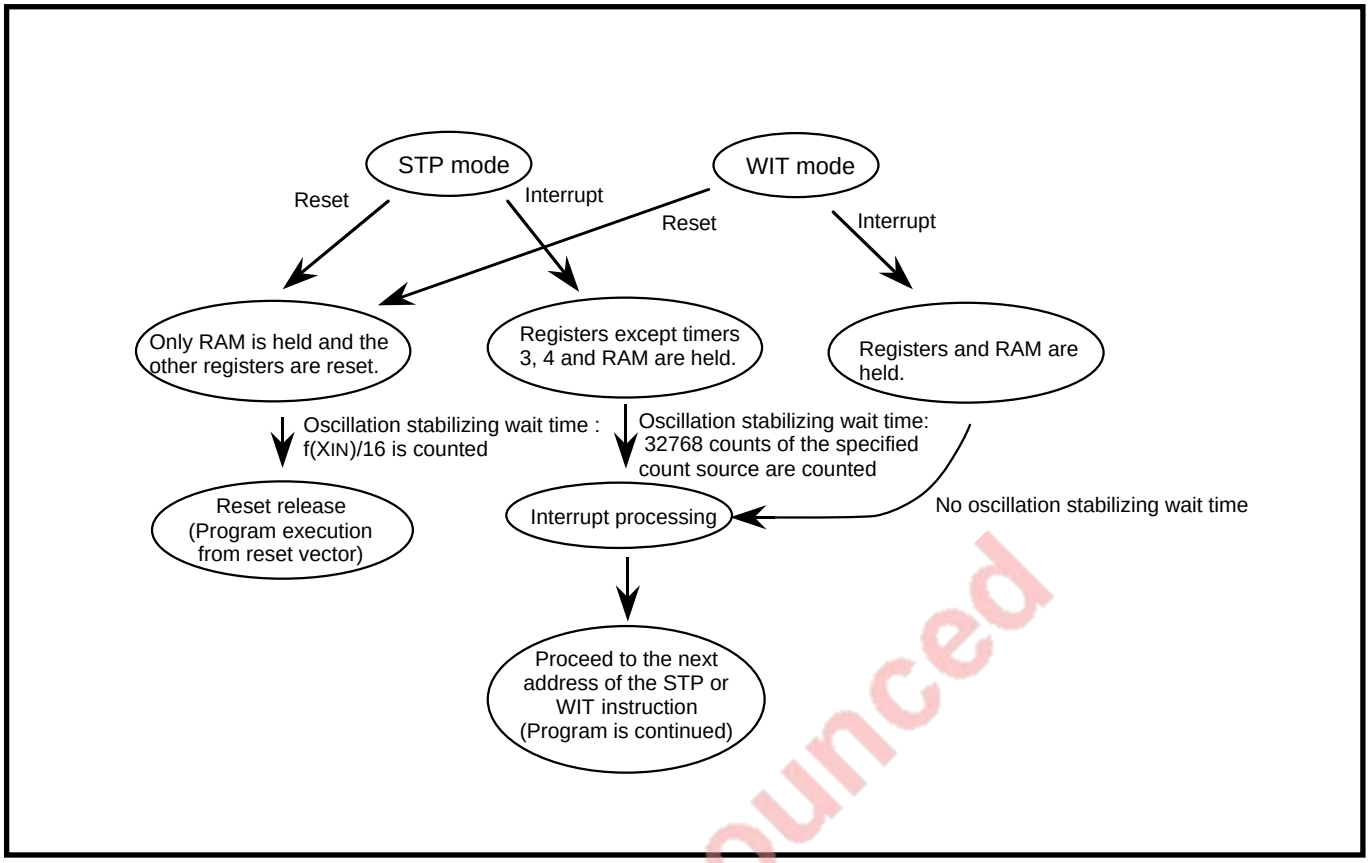


Fig. 1.17.2 State transition at low-power dissipation

1.17 Low-power dissipation function

1.17.1 Stop mode

To switch to the stop mode, execute the STP instruction. In the stop mode, the oscillation of both f(XIN) and f(XCIN) stops and the internal clock ϕ stops.
Accordingly, the CPU stops and the peripheral units also stop. This leads to a reduction of power dissipation.

Note: In the 7470/7477 group, the f(XCIN) is not available.

(1) State of stop mode

Table 1.17.1 shows a state of stop mode.

Note: When the STP instruction is executed, "FF_{16}" and "07_{16}" are automatically set in timer 3 and timer 4, respectively.

Table 1.17.1 State of stop mode

Item	State of stop mode
Oscillation	Stop
CPU	Stop
I/O port P0 to P5	State at execution STP instruction is held.
Timer	When internal count source selected : Stop When external count source selected : Operate
Serial I/O	Internal clock mode: Stop External clock mode: Operate
RAM	Held
SFR	Held (except timer3, timer4)
CPU register *	Held

* CPU register :
The following 6 registers are incorporated in the CPU.

- Accumulator
- Index register X
- Index register Y
- Stack pointer
- Program counter
- Processor status register

(2) Releasion of stop mode

The stop mode is released by inputting a reset signal or generating an interrupt request. There is a difference in restore processing from the stop mode between the use of reset input and the use of interrupt.

■ **Recovery by reset input**

The microcomputer is reset by applying the "L" level to the $\overline{\text{RESET}}$ pin for 2 µs or more in the stop mode, thereby releasing the stop mode.

After the stop mode is released, oscillator starts. (At this time, the inside is in the reset state.)

The reset state is released after 32768 counts of the XIN pin input after the “H” level is applied to the $\overline{\text{RESET}}$ pin.

At a start of oscillation of the oscillator, the oscillation is unstable. It takes time before stabilization of oscillation (oscillation stabilizing time). The oscillation stabilizing wait time is secured by the time for holding this internal reset state.

For the details of the reset, Refer to **“1.15 Reset.”**

Note: When the stop mode is released, the contents of the RAM before reset are held. However, the contents of the CPU register and the SFR cannot be held but are reset.

Figure 1.17.3 shows the oscillation stabilizing wait time at recovery from stop mode by reset input.

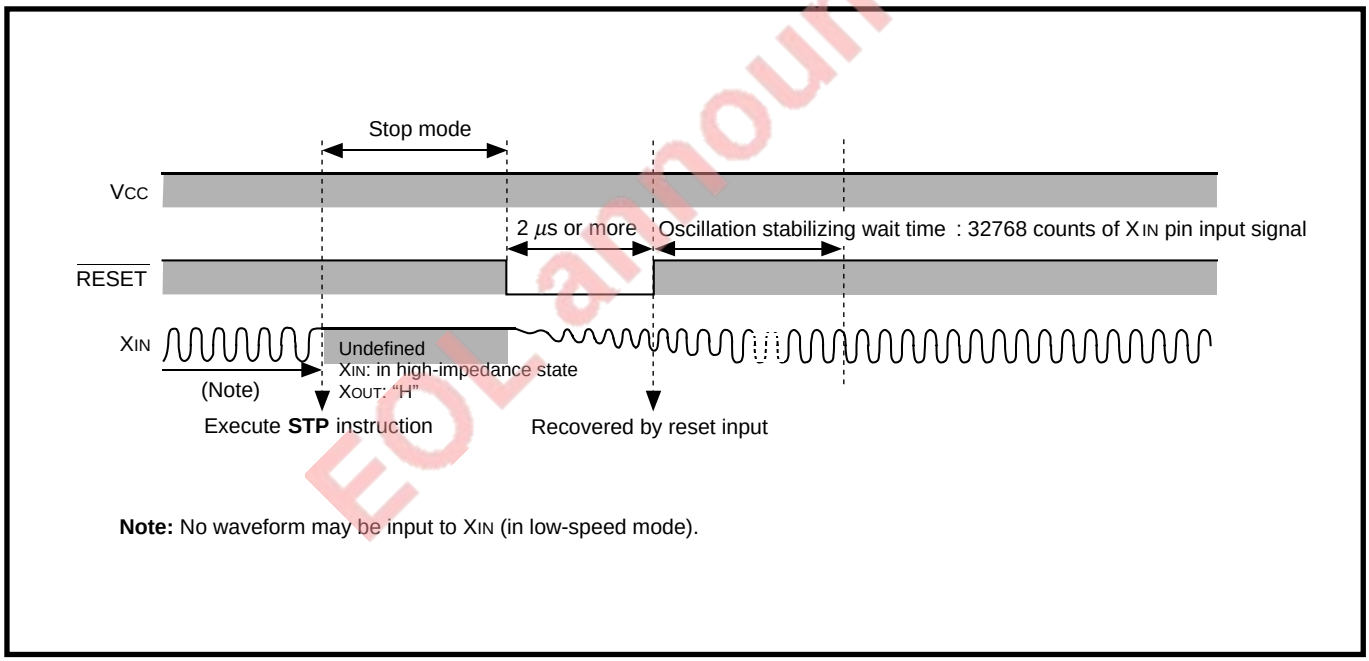


Fig. 1.17.3 oscillation stabilizing wait time at recovery from stop mode by reset input

1.17 Low-power dissipation function

■ Recovery by interrupt

When an interrupt request is generated in the stop mode, this stop mode is released and oscillation is started. The interrupt sources that are available for recovery are shown below.

- INT_0, INT_1
- $CNTR_0$, $CNTR_1$
- Serial I/O at using external clock
- Timer (timer 1, timer 2) at using external clock
- Key input (key on wake up)

However, when the above interrupt sources are used for recovery from the stop mode, perform the following setting and then execute the STP instruction to permit an interrupt to be used.

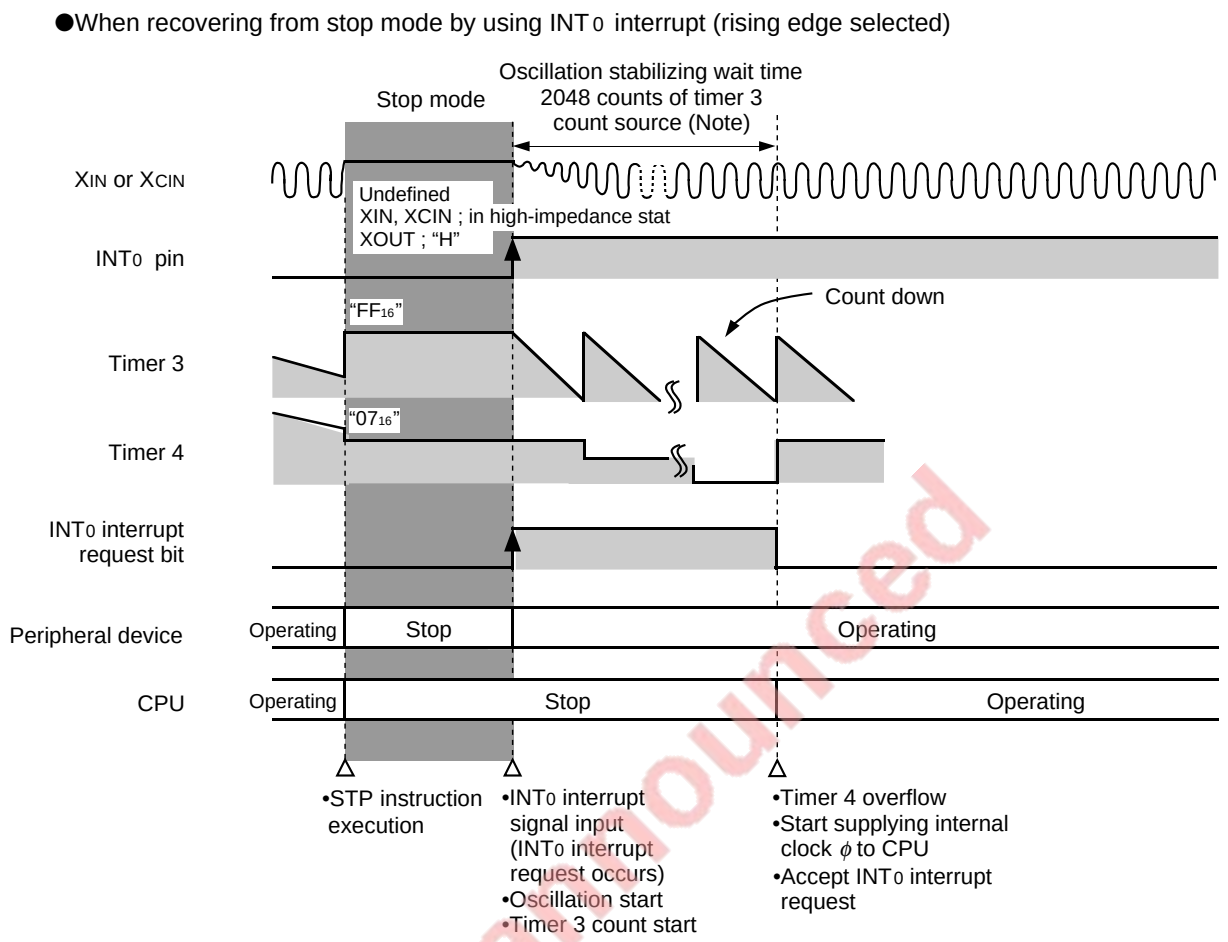
[Register setting]

① Clear the interrupt enable bit of timer 3 and timer 4 to "0." (Disabled)
② Set the count stop bit of timer 3 and timer 4 to "1." (Stop)
③ Select a count source of timer 3 in consideration of the oscillation stabilizing time of the oscillator.
Note: Re-set the previous count source at recovery.
④ Clear the interrupt request bit of the interrupt source to be used for recovery to "0."
⑤ Set the interrupt enable bit of the interrupt source to be used for recovery to "1."(Enabled)
⑥ Clear the count stop bit of timer 3 and timer 4 to "0." (Count starts)
⑦ When using the sub-clock, set the X_{COUT} drive capacity to high power. (Refer to **"1.16.2 Sub-clock oscillation circuit."**)
⑧ Clear the interrupt disable flag I to "0." (Enabled)

Note: In the stop mode, A-D conversion operation stops. Accordingly, execute the STP instruction after termination of the A-D conversion.

For the details of the Interrupt, refer to **"1.11 Interrupts."**
At a start of oscillation of the oscillator, the oscillation is unstable. It takes time before stabilization of oscillation (oscillation stabilizing time). At recovery by interrupt, the waiting time for the supply of internal clock ϕ to the CPU by timer 3 and timer 4^{*1} is automatically generated^{*2}. The oscillation stabilizing time of the system clock side is secured by this waiting time.
Figure 1.17.4 shows an example of restoration sequence from the stop mode by the INT_0 interrupt.

∗1: When the STP instruction is executed, "FF_{16}" and "07_{16}" are automatically set in the counter and latch of timer 3 and the counter and latch of timer 4, respectively.
∗2: The count source is supplied to timer 3 immediately after a start of oscillation, thereby starting a count operation. The supply of internal clock ϕ to the CPU is started when timer 4 overflows.



Note: The count source is a count source of timer 3 before execution of the STP instruction.

Fig. 1.17.4 Example of recovery sequence from stop mode by INT0 interrupt

1.17 Low-power dissipation function

1.17.2 Wait mode

To switch to the wait mode, execute the WIT instruction. In the wait mode, oscillation is continued but the internal clock ϕ stops. Accordingly, the CPU stops but the peripheral units operate since oscillation is continued.

(1) State of wait mode

Table 1.17.2 shows a state of wait mode.

Table 1.17.2 State of wait mode

Item	State of wait mode
Oscillation	Operate
CPU	Stop
I/O port P0 to P5	State at execution WIT instruction is held.
Timer	Operate
Serial I/O	Operate
RAM	Held
SFR	Held
CPU register*	Held

* CPU register :
The following 6 registers are incorporated in the CPU.
- Accumulator
- Index register X
- Index register Y
- Stack pointer
- Program counter
- Processor status register

(2) Releasion of wait mode

The wait mode is released by inputting a reset signal or generating an interrupt request. There is a difference in restore processing from the wait mode between the use of reset input and the use of interrupt.

■ **Recovery by reset input**

The microcomputer is reset by applying the "L" level to the $\overline{\text{RESET}}$ pin for 2 µs or more in the wait mode, thereby releasing the wait mode.

After the wait mode is released by inputting the reset signal, the supply of internal clock ϕ to the CPU is started.

The reset state is released after 32768 counts of the XIN pin input signal after the "H" level is applied to the $\overline{\text{RESET}}$ pin.

For the details of the reset, refer to **"1.15 Reset."**

Note: When the wait mode is released, the contents of the RAM before reset are held. However, the contents of the CPU register and the SFR cannot be held but are reset.

Figure 1.17.5 shows the reset input time.

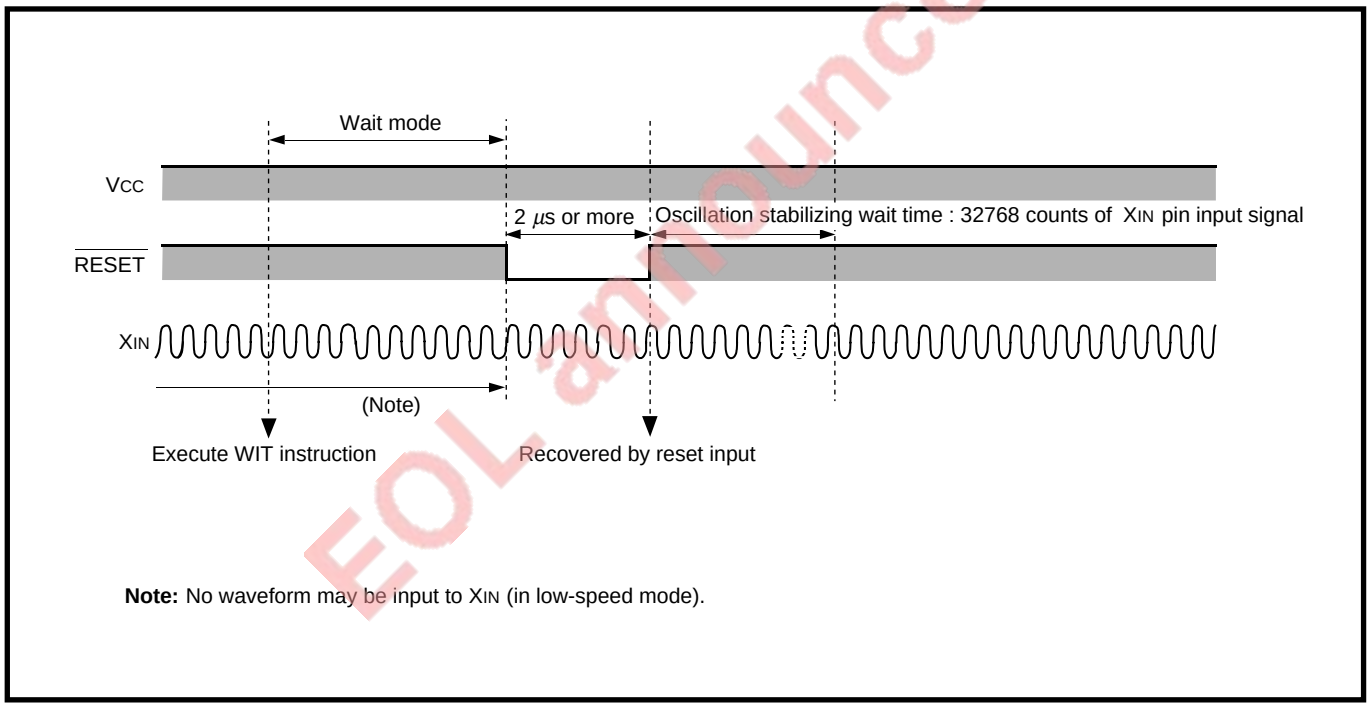


Fig. 1.17.5 Reset input time

1.17 Low-power dissipation function

■ Recovery by interrupt

In the wait mode, oscillation is continued. Accordingly, as soon as the wait mode is released, an instruction is executed.

When an interrupt request is generated in the wait mode, this wait mode is released and the supply of internal clock ϕ to the CPU is started. At the same time, the interrupt request used for recovery is accepted, so that the interrupt processing routine is executed.

The interrupt sources that are available for recovery are shown below.

- INT_0, INT_1
- $CNTR_0$, $CNTR_1$
- Serial I/O
- A-D conversion
- Timer 1 to timer 4
- Key input (key on wake up)

However, when the above interrupt sources are used for recovery from the wait mode, perform the following setting and then execute the WIT instruction to permit an interrupt to be used.

[Register setting]

① Clear the interrupt request bit of the interrupt source to be used for recovery to “0.”(No request)
② Set the interrupt enable bit of the interrupt source to be used for recovery to “1.”(Enabled)
③ Clear the interrupt disable flag I to “0.” (Enabled)

For the details of the Interrupt, refer to **“1.11 Interrupts.”**

1.17.3 Notes on use

[Notes on use of the stop mode]

■ **Clock after recovery**

After recovery from the stop mode by interrupt, the contents of the CPU mode register before execution of the STP instruction are held. Accordingly, if both f(XIN) and f(XCIN) were oscillating before execution of the STP instruction, the oscillation of both f(XIN) and f(XCIN) is restarted after recovery by interrupt.

In the above case, if f(XIN) is set as the system clock, the oscillation stabilizing wait time for 32768 counts of the XIN pin input signal is secured at recovery from the stop mode.

Note that the f(XCIN) clock may not be stabilized even after the lapse of the f(XIN) oscillation stabilizing wait time.

Note: In the 7470/7477 group, the f(XCIN) is not available.

■ **Interrupt processing after recovery**

After recovery from the stop mode, the interrupt request bit of timer 3 and timer 4 is “1.” Clear it to “0” if necessary. The interrupt request bit of timer 1 and timer 2 may also be set to “1.” Clear it to “0” as required.

1.17 Low-power dissipation function

1.17.4 Related register

(1) CPU mode register (Address 00FB16)

The CPU mode register consists of a stack page selection bit*1 and system clock control bits*2.

*1: In series having a RAM capacity of 192 bytes or less, this bit is not used because no RAM is located on page 1. (Be sure to set this bit to "0.")

*2: In the 7470/7477 group, which is not provided with a sub-clock generating circuit, this bit is not used. (Be sure to set this bit to "0.")

Figure 1.17.6 shows a structure of CPU mode register.

CPU mode register

b7 b6 b5 b4 b3 b2 b1 b0

0 0

CPU mode register (CPUM) [Address 00FB16]

B	Name	Function	At reset	R	W
0, 1	Fix these bits to "0."		0	○	○
2	Stack page selection bit	0: In page 0 area 1: In page 1 area (Note 1)	0	○	○
3	Nothing is allocated for this bit. This is write enabled bit and is undefined at reading.		?	?	×
4	P50, P51/XCIN,XCOUT selection bit	0: P50, P51 1: XCIN, XCOUT (Note 2)	0	○	○
5	XCOUT drive capacity selection bit	0: Low 1: High (Note 2)	0	○	○
6	Main clock (XIN–XOUT) stop bit	0: Oscillates 1: Stops (Note 2)	0	○	○
7	Internal system clock selection bit	0: XIN–XOUT selected (Ordinary mode) 1: XCIN–XCOUT selected (Low speed mode) (Note 2)	0	○	○

Notes 1: In the products having a RAM capacity of 192 bytes or less, set this bit to "0."

2: Since the 7470/7477 group is not provided with the sub-clock generating circuit, f(XCIN) cannot be used. Fix these bits to "0."

Fig. 1.17.6 Structure of CPU mode register

1.18 State transitions

The operation modes of the 7470/7471/7477/7478 group are classified as follows.

- Reset
- Oridinary mode
- Low-speed mode (7471/7478 group)
- Sub-clock mode (7471/7478 group)
- Stop mode
- Wait mode

Figure 1.18.1 shows a state transitions.

1.18 State transitions

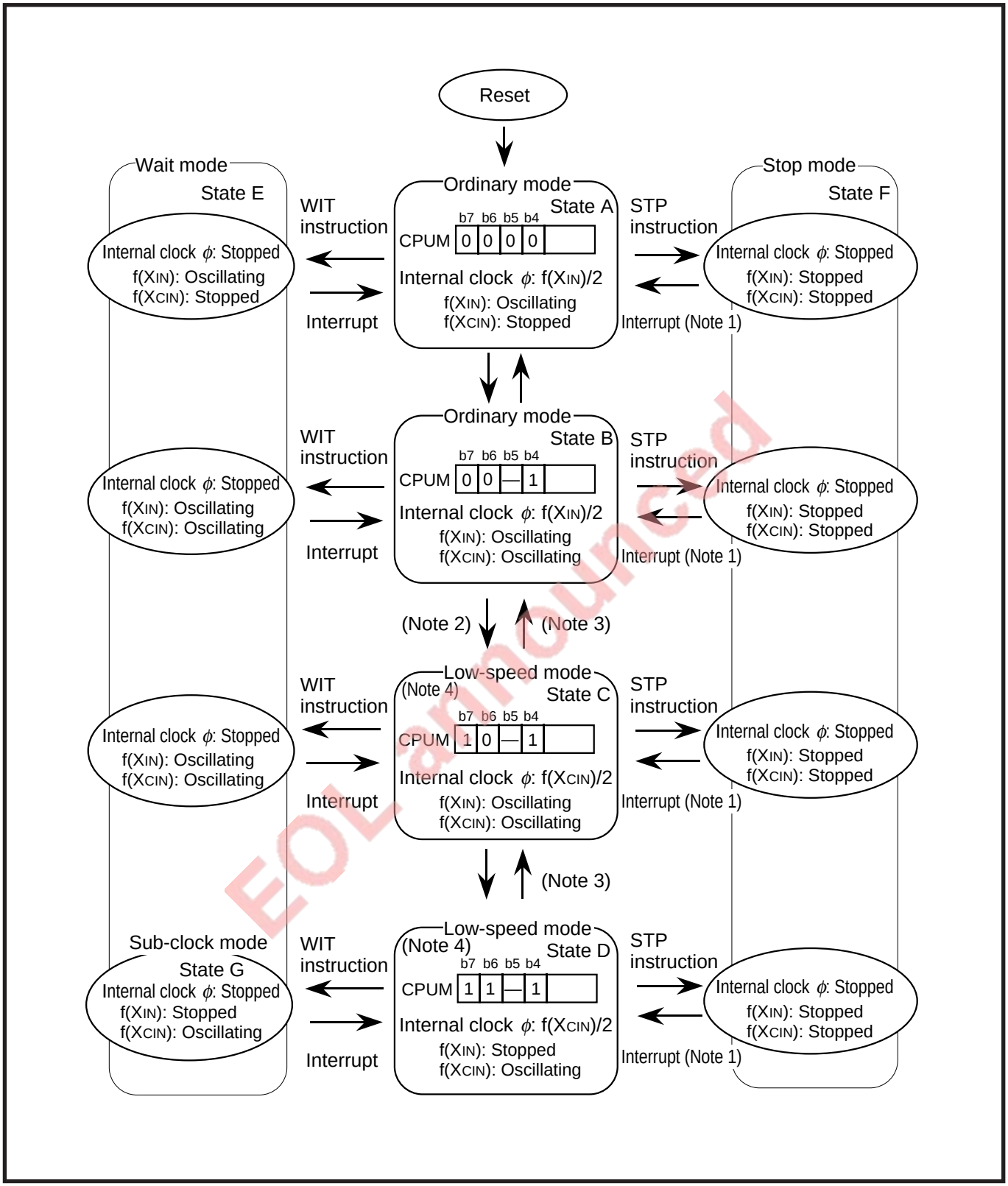


Fig. 1.18.1 State transitions

Notes 1: When changing from the stop mode to another mode, oscillation stabilizing wait time is generated automatically by connecting timer 3 and timer 4.

2: In the 7471/7478 group, where oscillating the stopped clock and switching the system clock, it is necessary to wait at software until oscillation is stabilized. At this time, set the bit 5 of the CPU mode register to "1" and set the XCOUT drive capacity to the high power mode. After the oscillation of sub-clock f(XCIN) becomes stable, clear the bit to "0" (low power mode) as required.

3: In the 7471/7478 group, when returning from the low-speed mode to the ordinary mode, use the main clock f(XIN) as a count source of the internal clock ϕ (state B). After that, clear bit 4 of the CPU mode register to "0" to stop the oscillation of f(XCIN) if necessary.

4: When using the low-speed mode in the 7471/7478 group, use it in one of the following states.

- Fix the XCOUT drive capacity to the high power mode (set the XCOUT drive capacity selection bit of the CPU register to "1.")
- When fixing the XCOUT drive capacity to the low power mode (clear the XCOUT drive capacity selection bit of the CPU mode register to "0"), lower the value of the resistor Rd* in the sub-clock oscillation circuit to a level at which the oscillation of f(XCIN) does not stop.
  * "Resistor Rd": Refer to the circuit example in **"Chapter 2 Application, 2.7 Oscillation circuit."**

1.18 State transitions

■ **Reset → Ordinary mode (State A)**

Immediately after reset, the main clock divided by 2 (f(XIN)/2) is selected as an internal clock ϕ and the I/O pins XCIN and XCOUT of the sub-clock f(XCIN) become ordinary ports. "FF16" and "0716" are set in timer 3 and timer 4 respectively, and also the main clock divided by 16 (f(XIN)/16) is selected as a count source of timer 3 and the overflow signal of timer 3 is selected as a count source of timer 4. Then a down-count is started.

When timer 4 overflows, the internal reset is released and the program starts from the address specified by reset vector.

■ **Low-speed mode (Stae C and state D)**

To the low-speed mode (state C and state D) using the sub-clock divided by 2 (f(XCIN)/2) as an internal clock ϕ, a transition is made by way of the ordinary mode (state A) → state B (→ state C).

In the 7470/7477 group, which is not provided with a sub-clock oscillation circuit, this mode is not provided.

■ **Wait mode (State E)**

In this mode, all the states of registers, I/O ports and internal RAMs are held. The internal clock ϕ stops at "H" but the oscillator does not stop.

From any of state A, state B, state C and state D, a return is made to the wait mode by executing the WIT instruction. When a return is made from the state D to the wait mode, the sub-clock mode in which only the timer function operates is provided. (In the 7470/7477 group, which is not provided with a sub-clock oscillation circuit, the sub-clock mode is not provided.)

Refer to **"1.17.2 Wait mode."**

■ **Stop mode (State F)**

In this mode, all the states of registers, I/O ports and internal RAMs except timer 3 and timer 4 are held and the oscillation of both main sub-clock is stopped. From any of state A, B, C and D, a return is made to the stop mode by executing the STP instruction.

Refer to **"1.17.1 Stop mode."**

■ **Sub-clock mode (State G)**

Only the clock-function is made to operate by sub-clock mode at low-power dissipation.

The sub-clock mode (state G) is provided by executing the WIT instruction in the low-speed mode (state D), and restoration from this state to the low-speed mode is attached by each interrupt.

In the 7470/7477 group, which is not provided with a sub-clock oscillation circuit, this mode is not provided.

1.19 Built-in PROM version

In contrast with the mask ROM version, a microcomputer incorporating a programmable ROM is called built-in PROM version.
There are two types of built-in PROM version as shown below.

- One Time PROM version
  Writing to the built-in PROM can be performed only once. Neither erase nor rewrite operations are enabled.
- Built-in EPROM version
  The built-in EPROM version is a programmable microcomputer with window and can perform write, erase, and rewrite operations.

The built-in PROM version has the EPROM mode for writing to the built-in PROM in addition to the same functions as those of the mask ROM version.
For an outline of performance, a pin configuration and a functional block diagram of the built-in PROM version, refer to **"1.3 Performance overview"**, **"1.4 Pin configuration"**, and **"1.6 Functional block diagram"**, respectively.
The 7470/7471/7477/7478 group supports the built-in PROM versions shown in Table 1.19.1.

Table 1.19.1 7470/7471/7477/7478 group built-in PROM version supporting products

Product name	(P)ROM size (bytes)	RAM size (bytes)	I/O Ports	Package	Remarks
M37470E4-XXXSP	8192	192	I/O ports: 22 (Including 4 analog input pins.) Input ports: 4	32P4B	One Time PROM version
M37470E8-XXXSP	16384	384			
M37471E4-XXXSP	8192	192	I/O ports: 28 (Including 8 analog input pins.) Input ports: 8	42P4B	One Time PROM version
M37471E4-XXXFP				56P6N-A	
M37471E8-XXXSP	16384	384		42P4B	
M37471E8-XXXFP				56P6N-A	
M37471E8SS				42S1B-A	EPROM version
M37477E8-XXXSP	16384	384	I/O ports: 18 Input ports: 8 (Including 4 analog input pins.)	32P4B	One Time PROM version
M37477E8-XXXFP				32P2W-A	
M37478E8-XXXSP	16384	384	I/O ports: 20 Input ports: 16 (Including 8 analog input pins.)	42P4B	One Time PROM version
M37478E8-XXXFP				56P6N-A	
M37478E8SS				42S1B-A	EPROM version

(As of Dec. 1997)

1.19 Built-in PROM version

1.19.1 EPROM mode

The built-in PROM version has the EPROM mode in addition to the same operation modes as those of the mask ROM version. The EPROM mode permits writing to the built-in PROM and reading from the built-in PROM. To the built-in PROM, writing, reading and erasing can be performed by the same operations as those of the M5M27C256K.

Table 1.19.2 shows a pin function in EPROM mode and Figure 1.19.1 to 1.19.6 show a connections in EPROM mode.

Table 1.19.2 Pin functions in EPROM mode

	Built-in PROM version	M5M27C256K
Pin name	V_{CC}	V_{CC}
	$P3_3$	V_{PP}
	V_{SS}	V_{SS}
	$P1_1$ to $P1_7$, $P2_0$ to $P2_3$, $P3_0$, $P3_1$, $P4_0$, $P4_1$	A_0 to A_{14}
	$P0_0$ to $P0_7$	D_0 to D_7
	V_{REF}	$\overline{CE}$
	$P3_2$	$\overline{OE}$

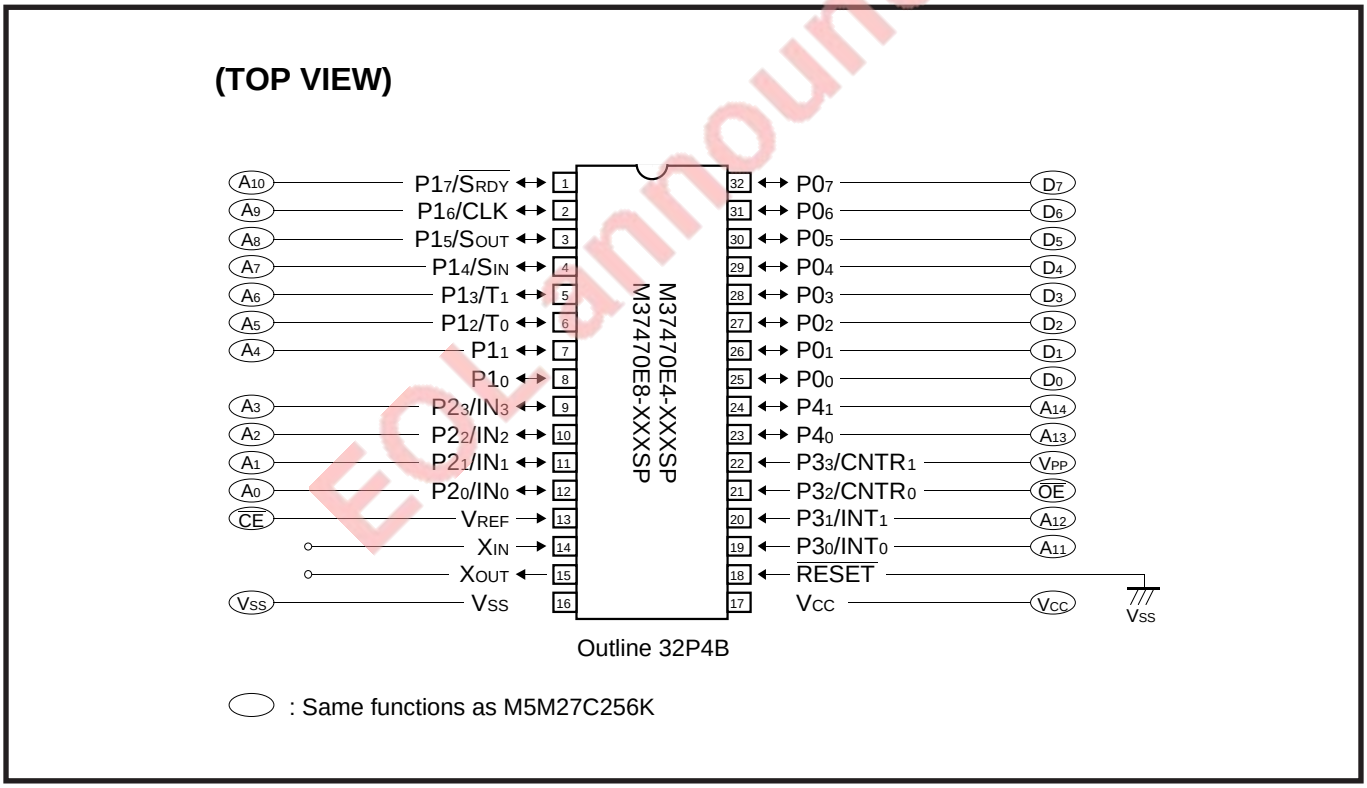
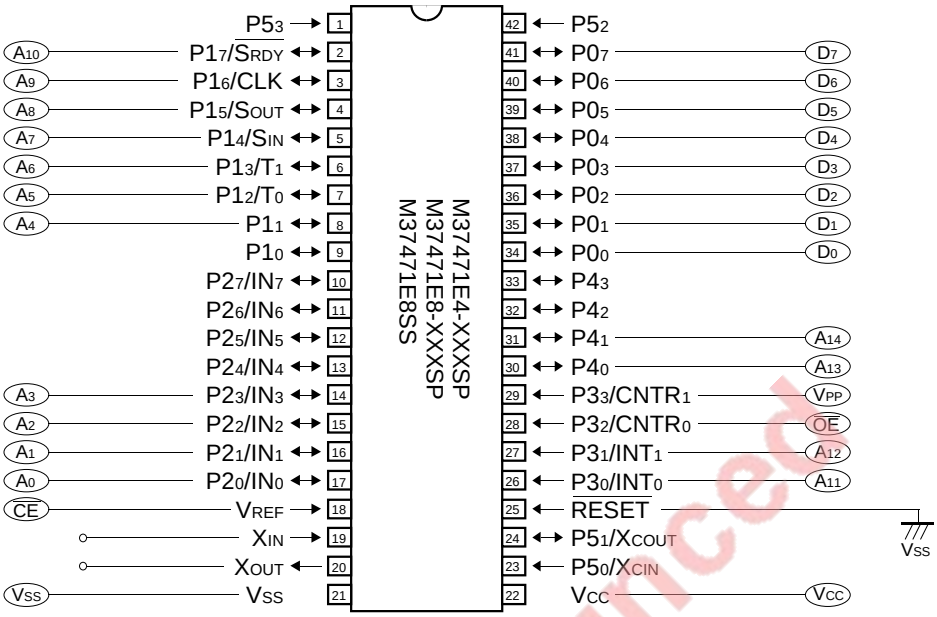


Fig. 1.19.1 Pin connection in EPROM mode of 7470 group

(TOP VIEW)



Outline 42P4B (Note)
42S1B-A (M37471E8SS)

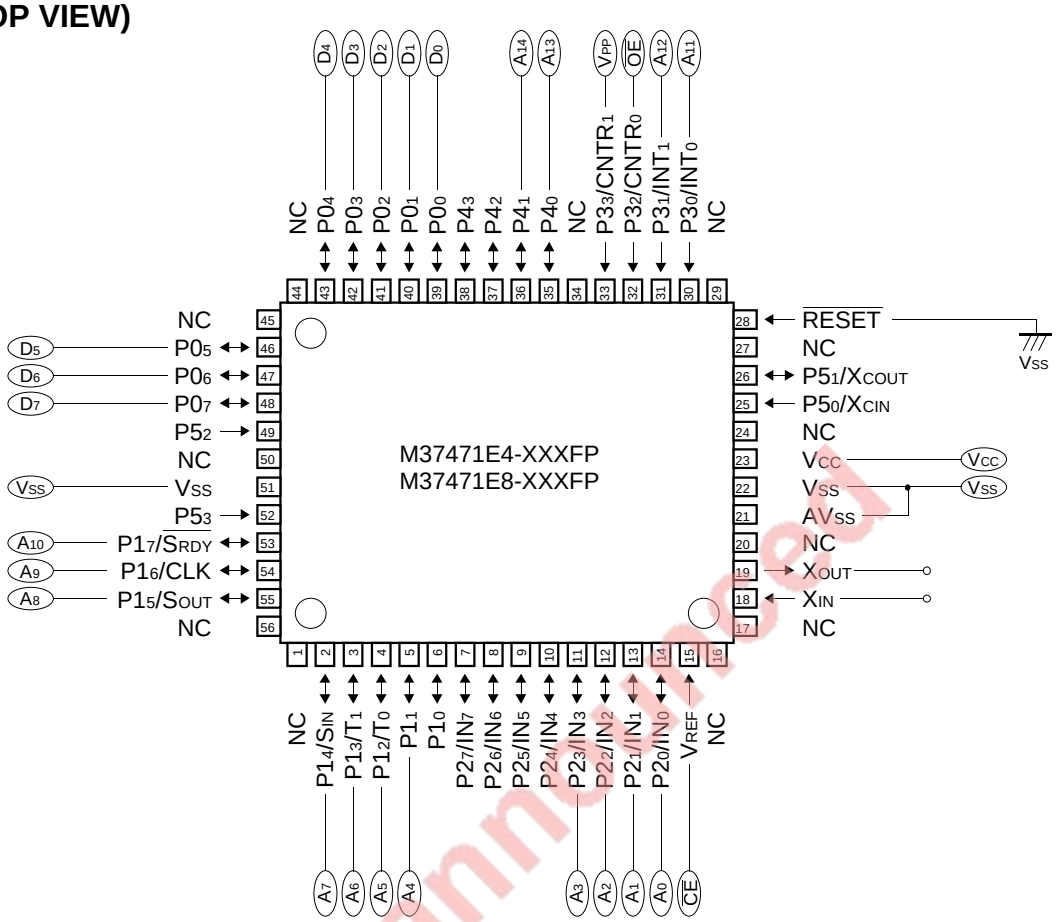
: Same functions as M5M27C256K

Note: The only difference between the 42P4B package product and the 56P6N-A package product are package shape, absolute maximum ratings and the fact that the 56P6N-A package product has an AVSS pin.

Fig. 1.19.2 Pin connection in EPROM mode of 7471 group (1)

1.19 Built-in PROM version

(TOP VIEW)



Outline 56P6N-A (Note)

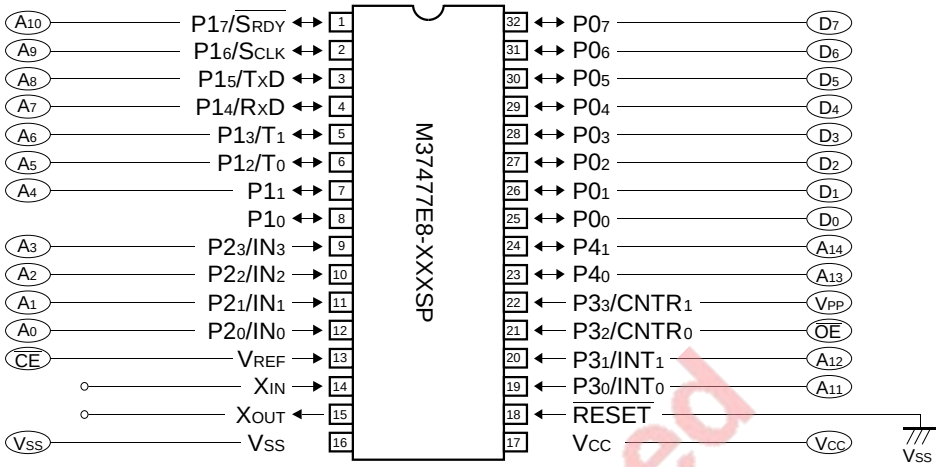
NC: No connection

⬭ : Same functions as M5M27C256K

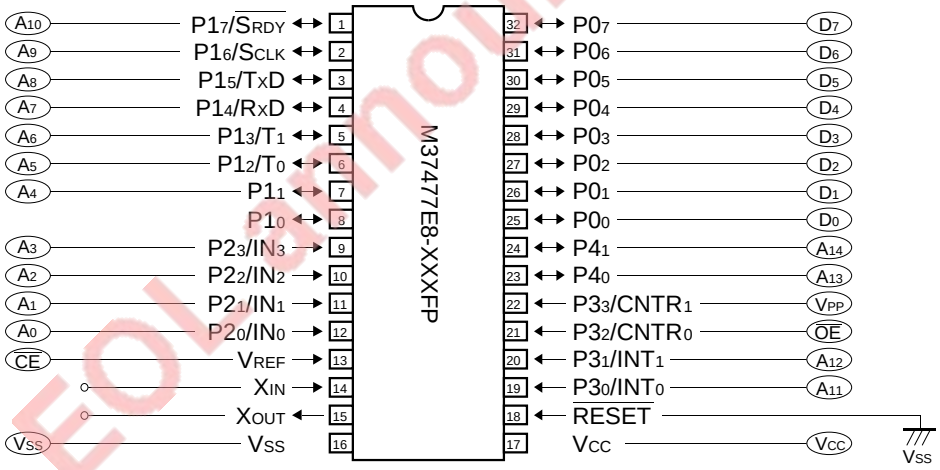
Note: The only difference between the 42P4B package product and the 56P6N-A package product are package shape, absolute maximum ratings and the fact that the 56P6N-A package product has an AVSS pin.

Fig. 1.19.3 Pin connection in EPROM mode of 7471 group (2)

(TOP VIEW)



Outline 32P4B (Note)



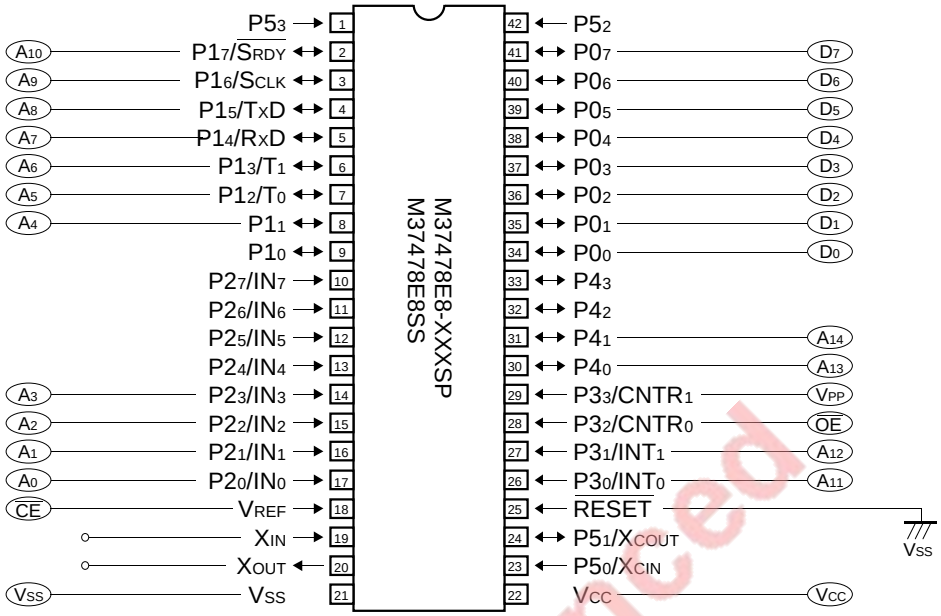
Outline 32P2W-A (Note)

◯ : Same functions as M5M27C256K

Note: The only difference between the 32P2W-A package product and the 32P4B package product are package shape, absolute maximum ratings.

Fig. 1.19.4 Pin connection in EPROM mode of 7477 group

(TOP VIEW)



Outline 42P4B (Note)
42S1B-A (M37478E8SS)

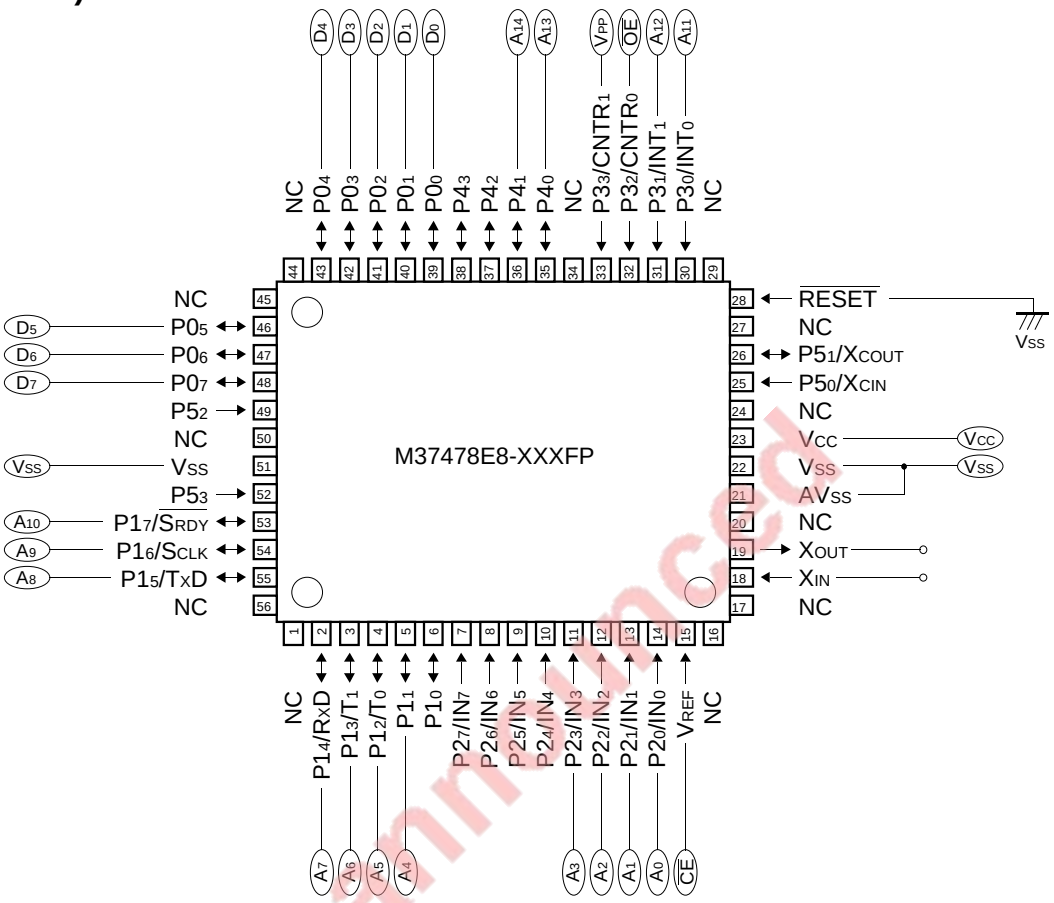
NC: No connection

⬭ : Same functions as M5M27C256K

Note: The only difference between the 42P4B package product and the 56P6N-A package product are package shape, absolute maximum ratings and the fact that the 56P6N-A package product has an AVSS pin.

Fig. 1.19.5 Pin connection in EPROM mode of 7478 group (1)

(TOP VIEW)



Outline 56P6N-A (Note)

NC: No connection

: Same functions as M5M27C256K

Note: The only difference between the 42P4B package product and the 56P6N-A package product are package shape, absolute maximum ratings and the fact that the 56P6N-A package product has an AVSS pin.

Fig. 1.19.6 Pin connection in EPROM mode of 7478 group (2)

1.19 Built-in PROM version

1.19.2 Pin description

Table 1.19.3 to Table 1.19.5 show the description of pin functions in the ordinary mode and the EPROM mode.

Table 1.19.3 Pin description (1)

Pin	Mode	Name	Input/ Output	Functions
V_{CC}, V_{SS}	Ordinary /EPROM	Power source		• Apply the following voltage to the V_{CC} pin: 2.7 V to 4.5 V (at f(X_{IN}) = (2.2 V_{CC}–2) MHz) or 4.5 V to 5.5 V (at f(X_{IN}) = 8 MHz). • Apply 0 V to the V_{SS} pin.
AV_{SS}	Ordinary /EPROM	Analog power source		• Ground level input pin for the A-D converter. • Apply the same voltage as the V_{SS} pin to the AV_{SS} pin. **Note:** This pin is dedicated to 56P6N-A package products among the 7471/7478 group.
V_{REF}	Ordinary	Reference voltage input	Input	• Reference voltage input pin for the A-D converter. • When using the A-D converter, apply 0.5 V_{CC} (≧2) to V_{CC} [V]. • When not using the A-D converter, connect to V_{CC}.
	EPROM	Mode input	Input	• V_{REF} works as $\overline{CE}$ input.
$\overline{RESET}$	Ordinary	Reset input	Input	• Reset input pin • The microcomputer is put into a reset state by keeping the $\overline{RESET}$ pin at “L” for 2 μs or more, and the reset state is released by returning the $\overline{RESET}$ pin to “H.”
	EPROM	Reset input	Input	• Connect to the V_{SS} pin.
X_{IN}	Ordinary /EPROM	Clock input	Input	• An input pin and an output pin for the main clock generating circuit. • Connect a ceramic resonator or a quartz-crystal oscillator between pins X_{IN} and X_{OUT}. • A feedback resistor is incorporated between the X_{IN} and the X_{OUT} pins. • To use an external clock input, connect the clock oscillation source to the X_{IN} pin and leave the X_{OUT} pin open.
X_{OUT}	Ordinary /EPROM	Clock output	Output	
$P0_0$–$P0_7$	Ordinary	I/O port P0	Input/ output	• Port P0 is an 8-bit I/O port. • The output structure is CMOS output. • In input mode, a pull-up transistor is connectable in units of one bit. • In input mode, a key-on wake up function is provided.
	EPROM	Data I/O D_0–D_7	Input/ output	• Port P0 works as data I/O (D_0 – D_7)

Table 1.19.4 Pin description (2)

Pin	Mode	Name	Input/ Output	Functions
$P1_0$–$P1_7$	Ordinary	I/O port P1	Input/ output	• Port P1 is an 8-bit I/O port. • The output structure is CMOS output. • In input mode, pull-up transistor can be connected in units of 4-bit. • Pins $P1_2$ and $P1_3$ are in common with timer output pins T_0, T_1 respectively. • In the case of the 7470/7471 group, $P1_4$ – $P1_7$ are in common with serial I/O pins SIN, SOUT, CLK, $\overline{SRDY}$ respectively. • In the case of the 7470/7471 group, the outputs of pins SOUT and the $\overline{SRDY}$ can be N-channel open drain outputs. • In the case of the 7477/7478 group, pins $P1_4$ – $P1_7$ are in common with serial I/O pins RXD, TXD, SCLK, $\overline{SRDY}$, respectively.
	EPROM	Address input A_4–A_{10}	Input	• The $P1_1$–$P1_7$ pins are address (A_4 – A_{10}) input pins. Put $P1_0$ into the open state.
$P2_0$–$P2_7$	Ordinary	I/O port P2 (7470/7471 group)	Input/ output	• Port P2 is an 8-bit I/O port. • The output structure is CMOS output. • In input mode, pull-up transistor can be connected in units of 4-bit. • Pins $P2_0$–$P2_7$ are in common with analog input pins IN_0–IN_7 respectively. **Note:** The 7470 group has only the 4 pins $P2_0$–$P2_3$ (IN_0–IN_3).
		Input port P2 (7477/7478 group)	Input	• Port P2 is an 8-bit input port. • It is impossible to connect a pull-up transistor. • Pins $P2_0$–$P2_7$ are in common with analog input pins IN_0–IN_7 respectively. **Note:** The 7477 group has only the 4 pins $P2_0$–$P2_3$ (IN_0–IN_3) are available.
	EPROM	Address input A_0–A_3	Input	• The $P2_0$ to $P2_3$ pins are address (A_0–A_3) input pins. • In the case of the 7471/7478 group, put the $P2_4$–$P2_7$ pins into the open state.

1.19 Built-in PROM version

Table 1.19.5 Pin description (3)

Pin	Mode	Name	Input/ Output	Functions
$P3_0$–$P3_3$	Ordinary	Input port P3	Input	• Port P3 is a 4-bit input port. • Pins $P3_0$, $P3_1$ are in common with external interrupt input pins INT_0, INT_1 respectively. • Pins $P3_2$, $P3_3$ are in common with timer input pins $CNTR_0$, $CNTR_1$ respectively.
	EPROM	Address input A_{11}, A_{12} Mode input V_{PP} input	Input	• The $P3_0$ and $P3_1$ pins are the address (A_{11}, A_{12}) input pins. • The $P3_2$ pin becomes an $\overline{OE}$ input pin. • The $P3_3$ pin is a V_{PP} input pin and V_{PP} is applied to it when the V_{PP} is input and the program is verified.
$P4_0$–$P4_3$	Ordinary	I/O port P4	Input/ output	• Port P4 is a 4-bit I/O port. • The output structure is CMOS output. • In input mode, pull-up transistor can be connected in units of 4-bit. **Note:** The 7470/7477 group has only 2 pins $P4_0$ and $P4_1$.
	EPROM	Address input A_{13}, A_{14}	Input	• The $P4_0$ to $P4_1$ pins are the address (A_{13}, A_{14}) input pins. • In the case of the 7471/7478 group, put the $P4_2$ and $P4_3$ pins into the open state.
$P5_0$–$P5_3$	Ordinary	Input port P5	Input	• Port P5 is a 4-bit input port. • Pull-up transistor can be connected in units of 4-bit. • Pins $P5_0$, $P5_1$ are in common with input/output pins for sub-clock generating circuit X_{CIN}, X_{COUT} respectively. • When using pins $P5_0$ and $P5_1$ as pins X_{CIN} and X_{COUT}, connect a quartz-crystal oscillator between pins X_{CIN} and X_{COUT}. • When using pins $P5_0$ and $P5_1$ as pins X_{CIN} and X_{COUT}, a feedback resistor is connected between pins X_{CIN} and X_{COUT}. • To use an external clock input, connect the clock oscillation source to the X_{CIN} pin and leave the X_{COUT} pin open. **Note:** Only the 7471/7478 group has pins $P5_0$ to $P5_3$.
	EPROM	Input port P5	Input	• Put this port into the open state.

1.19.3 Writing, reading, and erasing to built-in PROM

The built-in PROM version is put into the EPROM mode by applying the "L" level to the $\overline{\text{RESET}}$ pin. Write, read and erase operations to the built-in PROM in the EPROM mode are described below.
Table 1.19.6 shows input signals in each mode.

(1) Reading

- Apply 0 V to the $\overline{\text{RESET}}$ pin and 5 V to the V_{CC} pin.
- When an address signal (A_0-A_{14}) is input and the $\overline{CE}$ pin and the $\overline{OE}$ pin are caused to go "L", the contents of the PROM appear to data I/O pins (D_0-D_7).
- The data I/O pins (D_0-D_7) are put into a floating when either the $\overline{CE}$ pin or the $\overline{OE}$ pin is in the "H" state.

(2) Writing

- Apply 0 V to the $\overline{\text{RESET}}$ pin and 5 V to the V_{CC} pin.
- When the $\overline{OE}$ pin is caused to go to "H" and V_{PP} is applied to the V_{PP} pin, the program mode is provided.
- Set an address to the address input pins (A_0-A_{14}) and give write data to the data I/O pins (D_0-D_7) in parallel.
- In the above condition, a write operation is performed by causing the $\overline{CE}$ pin to go to "L".

When using a PROM programmer, specify an address into the following area.

- Address 6000_{16} to address $7FFF_{16}$ (for the M3747xE4)
- Address 4000_{16} to address $7FFF_{16}$ (for the M3747xE8)

(3) Erasing

- An erase operation is enabled only in the built-in EPROM version with window (M37471E8SS/M37478E8SS).
- Data can be erased by irradiating ultraviolet rays having a wave length of 2537Å.
- The minimum amount of irradiation required for an erase operation is 15 W•s/cm^2.

Table 1.19.6 Input/Output signal on each mode

Mode \ Pin	$\overline{CE}$	$\overline{OE}$	V_{PP}	V_{CC}	$\overline{\text{RESET}}$	D_0 to D_7
Reading	V_{IL}	V_{IL}	V_{CC}	V_{CC}	0 V	Output
Output disable	V_{IL}	V_{IH}	V_{CC}			Floating
Writing	V_{IL}	V_{IH}	V_{PP}			Input
Writing verify	V_{IH}	V_{IL}	V_{PP}			Output
Writing disable	V_{IH}	V_{IH}	V_{PP}			Floating

Note: V_{IL} denotes an "L" input voltage and V_{IH} denotes an "H" input voltage.

1.19 Built-in PROM version

1.19.4 Notes on use

The notes on using the built-in PROM version are shown below.

(1) All built-in PROM version products

■ Precautions at write operation

- Be careful not to apply an overvoltage to pins because a high voltage is used for a write operation. Exercise special care when turning on the power supply.
- For writing the contents of the PROM, use a dedicated programming adapter. This permits using a general-purpose PROM programmer for writing data. For details of dedicated programming adapters, refer to the “DEVELOPMENT SUPPORT TOOLS FOR MICROCOMPUTERS” data book.

■ Precautions at read operation

- When reading the contents of the PROM, use a dedicated programming adapter, so that reading can be performed by a general-purpose PROM programmer. For details of dedicated programming adapters, refer to the “DEVELOPMENT SUPPORT TOOLS FOR MICROCOMPUTERS” data book.

(2) One Time PROM version

■ Precautions before use

● The PROM of the 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 1.19.7 is recommended to verify programming.

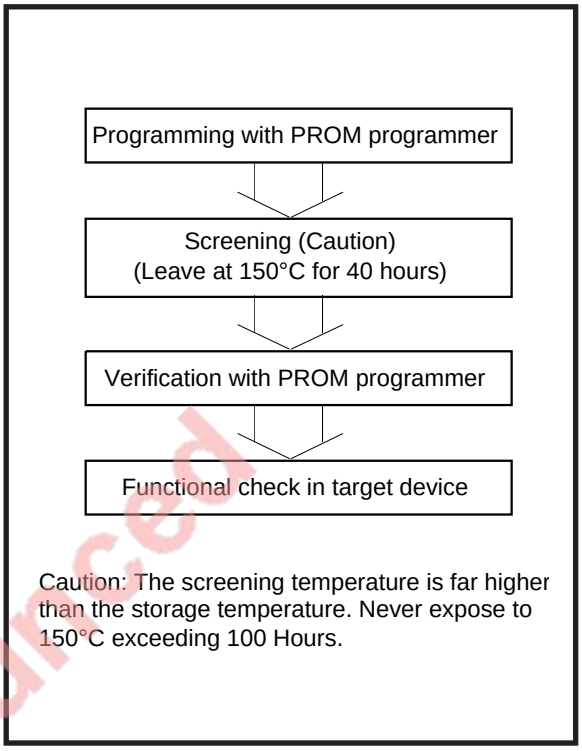


Fig. 1.19.7 Programming and testing of One Time PROM version

(3) Built-in EPROM version

■ Precautions on erasing

● Sunlight and fluorescent light include light that may erase the information written in the built-in PROM. When using the built-in EPROM version in the read mode, be sure to cover the transparent glass portion with a seal.

● This seal to cover the transparent glass portion is prepared on our side. Be careful not to bring the seal into contact with the microcomputer lead wires when covering the portion with the seal because this seal is made of metal (aluminum).

● Before erasing data, clean the transparent glass. If any finger stain or seal adhesive is stuck to the transparent glass, this prevents ultraviolet rays from passing, thereby affecting the erase characteristic adversely.

1.20 Electrical characteristics

1.20.1 Electrical characteristics

(1) 7470 group electrical characteristics

Table 1.20.1 shows the absolute maximum ratings of the 7470 group. Table 1.20.2 shows the recommended operating conditions. Table 1.20.3 shows the electrical characteristics. Table 1.20.4 shows the A-D converter characteristics.

Table 1.20.1 Absolute maximum ratings (7470 group)

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 +7	V
V_I	Input voltage		–0.3 to V_{CC}+0.3	V
V_O	Output voltage		–0.3 to V_{CC}+0.3	V
Pd	Power dissipation	Ta = 25 °C	1000	mW
Topr	Operating temperature		–20 to +85	°C
Tstg	Storage temperature		–40 to +150	°C

Table 1.20.2 Recommended operating conditions (7470 group)

(V_{CC} = 2.7 V to 5.5 V, Vss = 0 V, Ta = –20 °C to +85 °C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
V_{CC}	Power source voltage	f(X_{IN}) = 8.0 MHz	4.5	5.0	5.5	V
		f(X_{IN}) = (2.2V_{CC}–2.0)MHz	2.7		4.5	V
V_{SS}	Power source voltage			0		V
V_{IH}	"H" input voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P3_0$ to $P3_3$		0.8V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage $P2_0$ to $P2_3$, $P4_0$, $P4_1$		0.7V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage X_{IN}, $\overline{RESET}$		0.8V_{CC}		V_{CC}	V
V_{IL}	"L" input voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P3_0$ to $P3_3$		0		0.2V_{CC}	V
V_{IL}	"L" input voltage $P2_0$ to $P2_3$, $P4_0$, $P4_1$		0		0.25V_{CC}	V
V_{IL}	"L" input voltage X_{IN}		0		0.16V_{CC}	V
V_{IL}	"L" input voltage $\overline{RESET}$		0		0.12V_{CC}	V
I_{OH}(sum)	"H" sum output current of $P0_0$ to $P0_7$ and $P4_0$ and $P4_1$				–30	mA
I_{OH}(sum)	"H" sum output current of $P1_0$ to $P1_7$ and $P2_0$ to $P2_3$				–30	mA
I_{OL}(sum)	"L" sum output current of $P0_0$ to $P0_7$, $P4_0$ and $P4_1$				60	mA
I_{OL}(sum)	"L" sum output current of $P1_0$ to $P1_7$ and $P2_0$ to $P2_3$				60	mA
I_{OH}(peak)	"H" peak output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_3$, $P4_0$, $P4_1$				–10	mA
I_{OL}(peak)	"L" peak output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_3$, $P4_0$, $P4_1$				20	mA
I_{OH}(avg)	"H" average output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_3$, $P4_0$, $P4_1$ (Note 1)				–5	mA
I_{OL}(avg)	"L" average output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_3$, $P4_0$, $P4_1$ (Note 1)				10	mA
f(CNTR)	Timer input frequency $CNTR_0$ ($P3_2$), $CNTR_1$ ($P3_3$) (Note 2)	f(X_{IN}) = 8 MHz			2	MHz
		f(X_{IN}) = 4 MHz			1	MHz
f(CLK)	Serial I/O clock input frequency CLK ($P1_6$) (Note 2)	f(X_{IN}) = 8 MHz			2	MHz
		f(X_{IN}) = 4 MHz			1	MHz
f(X_{IN})	Clock input oscillation frequency (Note 2)	V_{CC} = 4.5 V to 5.5 V			8	MHz
		V_{CC} = 2.7 V to 4.5 V			2.2V_{CC}–2.0	MHz

Notes 1: The average output current I_{OH} (avg) or I_{OL} (avg) are the average value during a 100 ms.

2: The oscillation frequency is at 50 % duty cycle.

Table 1.20.3 Electrical characteristics (7470 group)

(V_{CC} = 2.7 V to 5.5 V, V_{SS} = 0 V, Ta = –20 °C to +85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions		Limits			Unit
				Min.	Typ.	Max.	
V_{OH}	"H" output voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$ $P2_0$ to $P2_3$, $P4_0$, $P4_1$	V_{CC} = 5 V, I_{OH} = –5 mA		3.0			V
		V_{CC} = 3 V, I_{OH} = –1.5 mA		2.0			V
V_{OL}	"L" output voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$ $P2_0$ to $P2_3$, $P4_0$, $P4_1$	V_{CC} = 5 V, I_{OL} = 10 mA				2.0	V
		V_{CC} = 3 V, I_{OL} = 3 mA				1.0	V
$V_{T+} - V_{T-}$	Hysteresis $P0_0$ to $P0_7$ $P3_0$ to $P3_3$	V_{CC} = 5 V			0.5		V
		V_{CC} = 3 V			0.3		V
$V_{T+} - V_{T-}$	Hysteresis $\overline{RESET}$	V_{CC} = 5 V			0.5		V
		V_{CC} = 3 V			0.3		V
$V_{T+} - V_{T-}$	Hysteresis $P1_4$/S_{IN} $P1_6$/CLK	Use as S_{IN} or CLK	V_{CC} = 5 V		0.5		V
			V_{CC} = 3 V		0.3		V
I_{IL}	"L" input current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$, $P4_1$	V_I = 0 V Not use pull-up transistor	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
		V_I = 0 V Use pull-up transistor	V_{CC} = 5 V	–0.25	–0.5	–1.0	mA
			V_{CC} = 3 V	–0.08	–0.18	–0.35	mA
I_{IL}	"L" input current $P3_0$ to $P3_3$	V_I = 0 V	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
I_{IL}	"L" input current $P2_0$ to $P2_3$	V_I = 0 V, not use pull-up transistor, not use as analog input	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
		V_I = 0 V, use pull-up transistor, not use as analog input	V_{CC} = 5 V	–0.25	–0.5	–1.0	mA
			V_{CC} = 3 V	–0.08	–0.18	–0.35	mA
I_{IL}	"L" input current XIN, $\overline{RESET}$	V_I = 0 V XIN is at stop mode	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
I_{IH}	"H" input current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$, $P4_1$	V_I = V_{CC} Not use pull-up transistor	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current $P3_0$ to $P3_3$	V_I = V_{CC}	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current $P2_0$ to $P2_3$	V_I = V_{CC}, not use pull-up transistor, not use as analog input	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current XIN, $\overline{RESET}$	V_I = V_{CC} XIN is at stop mode	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{CC}	Power source current	At system operation, A-D conversion is not executed	f(XIN) = 8 MHz, V_{CC} = 5 V		7	14	mA
			f(XIN) = 4 MHz, V_{CC} = 5 V		3.5	7	mA
			f(XIN) = 4 MHz, V_{CC} = 3 V		1.8	3.6	mA
		At system operation, A-D conversion is executed	f(XIN) = 8 MHz, V_{CC} = 5 V		7.5	15	mA
			f(XIN) = 4 MHz, V_{CC} = 5 V		4	8	mA
			f(XIN) = 4 MHz, V_{CC} = 3 V		2	4	mA
		At wait mode	f(XIN) = 8 MHz, V_{CC} = 5 V		2	4	mA
			f(XIN) = 4 MHz, V_{CC} = 5 V		1	2	mA
			f(XIN) = 4 MHz, V_{CC} = 3 V		0.5	1	mA
		At stop mode V_{CC} = 5 V	Ta = 25 °C		0.1	1	µA
			Ta = 85 °C		1	10	µA
V_{RAM}	RAM retention voltage	Stop all oscillation		2.0		5.5	V

1.20 Electrical characteristics

Table 1.20.4 A-D converter characteristics (7470 group)

(V_{CC} = 2.7 V to 5.5 V, V_{SS} = 0 V, Ta = –20 °C to +85 °C, f(X_{IN}) = 4 MHz, unless otherwise noted)

Symbol	Parameter	Test conditions		Limits			Unit
				Min.	Typ.	Max.	
—	Resolution					8	bits
—	Non-linearity error					±2	LSB
—	Differential non-linearity error					±0.9	LSB
V_{OT}	Zero transition error	V_{CC} = V_{REF} = 5.12 V, I_{OL}(sum) = 0 mA				2	LSB
		V_{CC} = V_{REF} = 3.072 V, I_{OL}(sum) = 0 mA				3	LSB
V_{FST}	Full scale transition error	V_{CC} = V_{REF} = 5.12 V				4	LSB
		V_{CC} = V_{REF} = 3.072 V				7	LSB
T_{CONV}	Conversion time		f(X_{IN}) = 8 MHz			12.5	µs
			f(X_{IN}) = 4 MHz			25	µs
V_{REF}	Reference input voltage			0.5V_{CC} (Note)		V_{CC}	V
R_{LADDER}	Ladder resistance value			2	5	10	kΩ
V_{IA}	Analog supply voltage			0		V_{REF}	V

Note: Set the V_{REF} voltage to 0.5 V_{CC} or more and 2 V or more. When using no A-D converter, connect it to V_{CC}.

(2) 7471 group electrical characteristics

Table 1.20.5 shows the absolute maximum ratings of the 7471 group. Table 1.20.6 shows the recommended operating conditions. Table 1.20.7 shows the electrical characteristics. Table 1.20.8 shows the A-D converter characteristics.

Table 1.20.5 Absolute maximum ratings (7471 group)

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 +7	V
V_I	Input voltage		–0.3 to V_{CC}+0.3	V
V_O	Output voltage		–0.3 to V_{CC}+0.3	V
Pd	Power dissipation	Ta = 25 °C	1000 (Note)	mW
Topr	Operating temperature		–20 to +85	°C
Tstg	Storage temperature		–40 to +150	°C

Note: The rating is 500 mW for the 56P6N-A package product.

Table 1.20.6 Recommended operating conditions (7471 group)
(Vcc = 2.7 V to 5.5 V, Vss = AVss = 0 V, Ta = –20 °C to +85 °C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
V_{CC}	Power source voltage	$f(X_{IN})$ = 8.0 MHz	4.5	5.0	5.5	V
		$f(X_{IN})$ = (2.2V_{CC}–2.0) MHz	2.7		4.5	V
V_{SS}	Power source voltage			0		V
AVss	Analog supply voltage			0		V
V_{IH}	"H" input voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P3_0$ to $P3_3$		0.8V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage $P2_0$ to $P2_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_3$ (Note 1)		0.7V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage X_{IN}, $\overline{RESET}$		0.8V_{CC}		V_{CC}	V
V_{IL}	"L" input voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P3_0$ to $P3_3$		0		0.2V_{CC}	V
V_{IL}	"L" input voltage $P2_0$ to $P2_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_3$ (Note 1)		0		0.25V_{CC}	V
V_{IL}	"L" input voltage X_{IN}		0		0.16V_{CC}	V
V_{IL}	"L" input voltage $\overline{RESET}$		0		0.12V_{CC}	V
I_{OH}(sum)	"H" sum output current of $P0_0$ to $P0_7$ and $P4_0$ to $P4_3$				–30	mA
I_{OH}(sum)	"H" sum output current of $P1_0$ to $P1_7$ and $P2_0$ to $P2_7$				–30	mA
I_{OL}(sum)	"L" sum output current of $P0_0$ to $P0_7$ and $P4_0$ to $P4_3$				60	mA
I_{OL}(sum)	"L" sum output current of $P1_0$ to $P1_7$ and $P2_0$ to $P2_7$				60	mA
I_{OH}(peak)	"H" peak output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P4_0$ to $P4_3$				–10	mA
I_{OL}(peak)	"L" peak output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P4_0$ to $P4_3$				20	mA
I_{OH}(avg)	"H" average output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P4_0$ to $P4_3$ (Note 2)				–5	mA
I_{OL}(avg)	"L" average output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P4_0$ to $P4_3$ (Note 2)				10	mA
f(CNTR)	Timer input frequency $CNTR_0$ ($P3_2$) $CNTR_1$ ($P3_3$) (Note 3)	$f(X_{IN})$ = 8 MHz			2	MHz
		$f(X_{IN})$ = 4 MHz			1	MHz
f(CLK)	Serial I/O clock input frequency CLK ($P1_6$) (Note 3)	$f(X_{IN})$ = 8 MHz			2	MHz
		$f(X_{IN})$ = 4 MHz			1	MHz
$f(X_{IN})$	Clock input oscillation frequency (Note 3)	V_{CC} = 4.5 V to 5.5 V			8	MHz
		V_{CC} = 2.7 V to 4.5 V			2.2V_{CC}–2.0	MHz
$f(X_{CIN})$	Sub-clock input oscillation frequency (Note 3, 4)			32	50	kHz

Notes 1: Except when $P5_0$ is used as X_{CIN}.
2: The average output current I_{OH}(avg) or I_{OL}(avg) are the average value during a 100 ms.
3: The oscillation frequency is at 50 % duty cycle.
4: Set $f(X_{CIN}) < f(X_{IN})/3$ when the sub-clock is used.

1.20 Electrical characteristics

Table 1.20.7 Electrical characteristics (7471 group)

(V_{CC} = 2.7 V to 5.5 V, V_{SS} = AV_{SS} = 0 V, Ta = –20 °C to +85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions		Limits			Unit
				Min.	Typ.	Max.	
V_{OH}	"H" output voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$ $P2_0$ to $P2_7$, $P4_0$ to $P4_3$	V_{CC} = 5 V, I_{OH} = –5 mA		3.0			V
		V_{CC} = 3 V, I_{OH} = –1.5 mA		2.0			V
V_{OL}	"L" output voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$ $P2_0$ to $P2_7$, $P4_0$ to $P4_3$	V_{CC} = 5 V, I_{OL} = 10 mA				2.0	V
		V_{CC} = 3 V, I_{OL} = 3 mA				1.0	V
$V_{T+} - V_{T-}$	Hysteresis $P0_0$ to $P0_7$ $P3_0$ to $P3_3$	V_{CC} = 5 V			0.5		V
		V_{CC} = 3 V			0.3		V
$V_{T+} - V_{T-}$	Hysteresis $\overline{RESET}$	V_{CC} = 5 V			0.5		V
		V_{CC} = 3 V			0.3		V
$V_{T+} - V_{T-}$	Hysteresis $P1_4$/S_{IN} $P1_6$/CLK	When used as S_{IN} or CLK	V_{CC} = 5 V		0.5		V
			V_{CC} = 3 V		0.3		V
I_{IL}	"L" input current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_3$	V_I = 0 V Not use pull-up transistor	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
		V_I = 0 V Use pull-up transistor	V_{CC} = 5 V	–0.25	–0.5	–1.0	mA
			V_{CC} = 3 V	–0.08	–0.18	–0.35	mA
I_{IL}	"L" input current $P3_0$ to $P3_3$	V_I = 0V	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
I_{IL}	"L" input current $P2_0$ to $P2_7$	V_I = 0 V, not use pull-up transistor, not use as analog input	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
		V_I = 0 V, use pull-up transistor, not use as analog input	V_{CC} = 5 V	–0.25	–0.5	–1.0	mA
			V_{CC} = 3 V	–0.08	–0.18	–0.35	mA
I_{IL}	"L" input current X_{IN}, $\overline{RESET}$	V_I = 0 V X_{IN} is at stop mode	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
I_{IH}	"H" input current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_3$	V_I = V_{CC} Not use pull-up transistor	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current $P3_0$ to $P3_3$	V_I = V_{CC}	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current $P2_0$ to $P2_7$	V_I = V_{CC}, not use pull-up transistor, not use as analog input	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current X_{IN}, $\overline{RESET}$	V_I = V_{CC} X_{IN} is at stop mode	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{CC}	Power source current	At system operation, A-D conversion is not executed, f(X_{IN}) = 8 MHz	V_{CC} = 5 V		7	14	mA
		At system operation, A-D conversion is not executed, f(X_{IN}) = 4 MHz			3.5	7	mA
			V_{CC} = 3 V		1.8	3.6	mA
		At system operation, A-D conversion is executed, f(X_{IN}) = 8 MHz	V_{CC} = 5 V		7.5	15	mA
		At system operation, A-D conversion is executed, f(X_{IN}) = 4 MHz			4	8	mA
			V_{CC} = 3 V		2	4	mA
		In low-speed mode, Ta = 25°C, low-power mode f(X_{CIN}) = 32 kHz At A-D conversion is not executed	V_{CC} = 5 V		30	80	µA
			V_{CC} = 3 V		15	40	µA
		At wait mode, f(X_{IN}) = 8 MHz	V_{CC} = 5 V		2	4	mA
		At wait mode, f(X_{IN}) = 4 MHz			1	2	mA
			V_{CC} = 3 V		0.5	1	mA
		At wait mode, Ta = 25°C, low-power mode, f(X_{CIN}) = 32 kHz	V_{CC} = 5 V		3	12	µA
			V_{CC} = 3 V		2	8	µA
		At stop mode, V_{CC} = 5 V	Ta = 25 °C		0.1	1	µA
			Ta = 85 °C		1	10	µA
V_{RAM}	RAM retention voltage	Stop all oscillation		2.0		5.5	V

Table 1.20.8 A-D converter characteristics (7471 group)

(V_{CC} = 2.7 V to 5.5 V, V_{SS} = AV_{SS} = 0 V, Ta = –20 °C to +85 °C, f(X_{IN}) = 4 MHz, unless otherwise noted)

Symbol	Parameter	Test conditions		Limits			Unit
				Min.	Typ.	Max.	
—	Resolution					8	bits
—	Non-linearity error					±2	LSB
—	Differential non-linearity error					±0.9	LSB
V_{OT}	Zero transition error	V_{CC} = V_{REF} = 5.12 V, I_{OL}(sum) = 0 mA				2	LSB
		V_{CC} = V_{REF} = 3.072 V, I_{OL}(sum) = 0 mA				3	LSB
V_{FST}	Full scale transition error	V = V_{REF} = 5.12 V				4	LSB
		V_{CC} = V_{REF} = 3.072 V				7	LSB
T_{CONV}	Conversion time		f(X_{IN}) = 8 MHz			12.5	µs
			f(X_{IN}) = 4 MHz			25	µs
V_{REF}	Reference input voltage			0.5V_{CC} (Note)		V_{CC}	V
R_{LADDER}	Ladder resistance value			2	5	10	kΩ
V_{IA}	Analog input voltage			0		V_{REF}	V

Note: Set the V_{REF} voltage to 0.5 V_{CC} or more and 2 V or more. When using no A-D converter, connect it to V_{CC}.

1.20 Electrical characteristics

(3) 7477 group electrical characteristics

Table 1.20.9 shows the absolute maximum ratings of the 7477 group. Table 1.20.10 shows the recommended operating conditions. Table 1.20.11 shows the electrical characteristics. Table 1.20.12 shows the A-D converter characteristics.

Table 1.20.9 Absolute maximum ratings (7477group)

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 +7	V
V_I	Input voltage		–0.3 to V_{CC}+0.3	V
V_O	Output voltage		–0.3 to V_{CC}+0.3	V
Pd	Power dissipation	Ta = 25 °C	1000 (Note)	mW
Topr	Operating temperature		–20 to +85	°C
Tstg	Storage temperature		–40 to +150	°C

Note : The rating is 500 mW for the 32P2W-A package product.

Table 1.20.10 Recommended operating conditions (7477 group)

(V_{CC} = 2.7 V to 5.5 V, V_{SS} = 0 V, Ta = –20 °C to +85 °C, unless otherwise noted)

Symbol	Parameter			Limits			Unit
				Min.	Typ.	Max.	
V_{CC}	Power source voltage	$f(X_{IN})$ = 8.0 MHz		4.5	5.0	5.5	V
		$f(X_{IN})$ = (2.2V_{CC}–2.0) MHz		2.7		4.5	V
V_{SS}	Power source voltage				0		V
V_{IH}	"H" input voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P3_0$ to $P3_3$			0.8V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage $P2_0$ to $P2_3$, $P4_0$, $P4_1$			0.7V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage X_{IN}, $\overline{RESET}$			0.8V_{CC}		V_{CC}	V
V_{IL}	"L" input voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P3_0$ to $P3_3$			0		0.2V_{CC}	V
V_{IL}	"L" input voltage $P2_0$ to $P2_3$, $P4_0$, $P4_1$			0		0.25V_{CC}	V
V_{IL}	"L" input voltage X_{IN}			0		0.16V_{CC}	V
V_{IL}	"L" input voltage $\overline{RESET}$			0		0.12V_{CC}	V
I_{OH}(sum)	"H" sum output current $P0_0$ to $P0_7$, $P4_0$ and $P4_1$					–30	mA
I_{OH}(sum)	"H" sum output current $P1_0$ to $P1_7$					–30	mA
I_{OL}(sum)	"L" sum output current $P0_0$ to $P0_7$, $P4_0$ and $P4_1$					60	mA
I_{OL}(sum)	"L" sum output current $P1_0$ to $P1_7$					60	mA
I_{OH}(peak)	"H" peak output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$, $P4_1$					–10	mA
I_{OL}(peak)	"L" peak output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$, $P4_1$					20	mA
I_{OH}(avg)	"H" average output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$, $P4_1$ (Note 1)					–5	mA
I_{OL}(avg)	"L" average output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$, $P4_1$ (Note 1)					10	mA
f(CNTR)	Timer input frequency $CNTR_0$ ($P3_2$), $CNTR_1$ ($P3_3$) (Note 2)	$f(X_{IN})$ = 8 MHz				2	MHz
		$f(X_{IN})$ = 4 MHz				1	MHz
f(S_{CLK})	Serial I/O clock input frequency S_{CLK} ($P1_6$) (Note 2)	Use as clock synchronous serial I/O mode	$f(X_{IN})$ = 8 MHz			500	kHz
			$f(X_{IN})$ = 4 MHz			250	kHz
		Use as UART mode	$f(X_{IN})$ = 8 MHz			2	MHz
			$f(X_{IN})$ = 4 MHz			1	MHz
f(X_{IN})	Clock input oscillation frequency (Note 2)	V_{CC} = 4.5 V to 5.5 V				8	MHz
		V_{CC} = 2.7 V to 4.5 V				2.2V_{CC}–2.0	MHz

Notes 1: The average output current I_{OH} (avg) or I_{OL} (avg) are the average value during a 100 ms.

2: The oscillation frequency is at 50 % duty cycle.

Table 1.20.11 Electrical characteristics (7477 group)

(Vcc = 2.7 V to 5.5 V, Vss = 0 V, Ta = –20 °C to +85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions		Limits			Unit
				Min.	Typ.	Max.	
V_{OH}	"H" output voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$ $P4_0$, $P4_1$	V_{CC} = 5 V, I_{OH} = –5 mA		3.0			V
		V_{CC} = 3 V, I_{OH} = –1.5 mA		2.0			V
V_{OL}	"L" output voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$ $P4_0$, $P4_1$	V_{CC} = 5 V, I_{OL} = 10 mA				2.0	V
		V_{CC} = 3 V, I_{OL} = 3 mA				1.0	V
$V_{T+} - V_{T-}$	Hysteresis $P0_0$ to $P0_7$ $P3_0$ to $P3_3$	V_{CC} = 5 V			0.5		V
		V_{CC} = 3 V			0.3		V
$V_{T+} - V_{T-}$	Hysteresis $\overline{RESET}$	V_{CC} = 5 V			0.5		V
		V_{CC} = 3 V			0.3		V
$V_{T+} - V_{T-}$	Hysteresis $P1_4$/RxD $P1_6$/S_{CLK}	When used as RxD or S_{CLK}	V_{CC} = 5 V		0.5		V
			V_{CC} = 3 V		0.3		V
I_{IL}	"L" input current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$, $P4_1$	V_I = 0 V Not use pull-up transistor	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
		V_I = 0 V Use pull-up transistor	V_{CC} = 5 V	–0.25	–0.5	–1.0	mA
			V_{CC} = 3 V	–0.08	–0.18	–0.35	mA
I_{IL}	"L" input current $P3_0$ to $P3_3$	V_I = 0V	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
I_{IL}	"L" input current $P2_0$ to $P2_3$	V_I = 0 V, Not use as analog input	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
I_{IL}	"L" input current X_{IN}, $\overline{RESET}$	V_I = 0 V X_{IN} is at stop mode	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
I_{IH}	"H" input current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$, $P4_1$	V_I = V_{CC} Not use pull-up transistor	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current $P3_0$ to $P3_3$	V_I = V_{CC}	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current $P2_0$ to $P2_3$	V_I = V_{CC} Not use as analog input	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current X_{IN}, $\overline{RESET}$	V_I = V_{CC} X_{IN} is at stop mode	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{CC}	Power source current	At system operation, A-D conversion is not executed	f(X_{IN}) = 8 MHz, V_{CC} = 5 V		7	14	mA
			f(X_{IN}) = 4 MHz, V_{CC} = 5 V		3.5	7	mA
			f(X_{IN}) = 4 MHz, V_{CC} = 3 V		1.8	3.6	mA
		At system operation, A-D conversion is executed	f(X_{IN}) = 8 MHz, V_{CC} = 5 V		7.5	15	mA
			f(X_{IN}) = 4 MHz, V_{CC} = 5 V		4	8	mA
			f(X_{IN}) = 4 MHz, V_{CC} = 3 V		2	4	mA
		At wait mode	f(X_{IN}) = 8 MHz, V_{CC} = 5 V		2	4	mA
			f(X_{IN}) = 4 MHz, V_{CC} = 5 V		1	2	mA
			f(X_{IN}) = 4 MHz, V_{CC} = 3 V		0.5	1	mA
		At stop mode, V_{CC} = 5 V	Ta = 25 °C		0.1	1	µA
			Ta = 85 °C		1	10	µA
V_{RAM}	RAM retention voltage	Stop all oscillation		2.0		5.5	V

1.20 Electrical characteristics

Table 1.20.12 A-D converter characteristics (7477 group)

(V_{CC} = 2.7 V to 5.5 V, Vss = 0 V, Ta = –20 °C to +85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
—	Resolution				8	bits
—	Absolute accuracy				±3	LSB
T_{CONV}	Conversion time	V_{CC} = 4.5 V to 5.5 V, f(X_{IN}) = 8 MHz			12.5	µs
		V_{CC} = 2.7 V to 5.5 V, f(X_{IN}) = 4 MHz			25	µs
V_{REF}	Reference input voltage		0.5V_{CC} (Note)		V_{CC}	V
R_{LADDER}	Ladder resistance value		2	5	10	kΩ
V_{IA}	Analog input voltage		0		V_{REF}	V

Note : Set the V_{REF} voltage to 0.5 V_{CC} or more and 2 V or more. When using no A-D converter, connect it to V_{CC}.

(4) 7478 group electrical characteristics

Table 1.20.13 shows the absolute maximum ratings of the 7478 group. Table 1.20.14 shows the recommended operating conditions. Table 1.20.15 shows the electrical characteristics. Table 1.20.16 shows the A-D converter characteristics.

Table 1.20.13 Absolute maximum ratings (7478 group)

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 +7	V
V_I	Input voltage		–0.3 to V_{CC}+0.3	V
V_O	Output voltage		–0.3 to V_{CC}+0.3	V
Pd	Power dissipation	Ta = 25 °C	1000 (Note)	mW
Topr	Operating temperature		–20 to +85	°C
Tstg	Storage temperature		–40 to +150	°C

Note : The rating is 500 mW for the 56P6N-A package product.

Table 1.20.14 Recommended operating conditions (7478 group)

(V_{CC} = 2.7 V to 5.5 V, V_{SS} = AVss = 0 V, Ta = –20 °C to +85 °C, unless otherwise noted)

Symbol	Parameter			Limits			Unit
				Min.	Typ.	Max.	
V_{CC}	Power surce voltage	$f(X_{IN})$ = 8.0 MHz		4.5	5.0	5.5	V
		$f(X_{IN})$ =(2.2V_{CC}–2.0) MHz		2.7		4.5	V
V_{SS}	Power surce voltage				0		V
AVss	Analog supply voltage				0		V
V_{IH}	"H" input voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P3_0$ to $P3_3$			0.8V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage $P2_0$ to $P2_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_3$ (Note 1)			0.7V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage X_{IN}, $\overline{RESET}$			0.8V_{CC}		V_{CC}	V
V_{IL}	"L" input voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P3_0$ to $P3_3$			0		0.2V_{CC}	V
V_{IL}	"L" input voltage $P2_0$ to $P2_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_3$ (Note 1)			0		0.25V_{CC}	V
V_{IL}	"L" input voltage X_{IN}			0		0.16V_{CC}	V
V_{IL}	"L" input voltage $\overline{RESET}$			0		0.12V_{CC}	V
I_{OH}(sum)	"H" sum output current of $P0_0$ to $P0_7$ and $P4_0$ to $P4_3$					–30	mA
I_{OH}(sum)	"H" sum output current of $P1_0$ to $P1_7$					–30	mA
I_{OL}(sum)	"L" sum output current of $P0_0$ to $P0_7$ and $P4_0$ to$P4_3$					60	mA
I_{OL}(sum)	"L" sum output current of $P1_0$ to $P1_7$					60	mA
I_{OH}(peak)	"H" peak output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$ to $P4_3$					–10	mA
I_{OL}(peak)	"L" peak output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$ to $P4_3$					20	mA
I_{OH}(avg)	"H" average output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$ to $P4_3$ (Note 2)					–5	mA
I_{OL}(avg)	"L" average output current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$ to $P4_3$ (Note 2)					10	mA
f(CNTR)	Timer input frequency $CNTR_0$ ($P3_2$), $CNTR_1$ ($P3_3$) (Note 3)	$f(X_{IN})$ = 8 MHz				2	MHz
		$f(X_{IN})$ = 4 MHz				1	MHz
f(S_{CLK})	Serial I/O clock input frequency S_{CLK} ($P1_6$) (Note 3)	Use as clock synchronous serial I/O mode	$f(X_{IN})$ = 8 MHz			500	kHz
			$f(X_{IN})$ = 4 MHz			250	kHz
		Use as UART mode	$f(X_{IN})$ = 8 MHz			2	MHz
			$f(X_{IN})$ = 4 MHz			1	MHz
f(X_{IN})	Clock input oscillation frequency (Note 3)	V_{CC} = 4.5 V to 5.5 V				8	MHz
		V_{CC} = 2.7 V to 4.5 V				2.2V_{CC}–2.0	MHz
f(X_{CIN})	Sub-clock input oscillation frequency (Notes 3, 4)				32	50	kHz

Notes 1: Except when $P5_0$ is used as X_{CIN}.

2: The average output current I_{OH} (avg) and I_{OL} (avg) are the average value during a 100 ms.

3: The oscillation frequency is at 50 % duty cycle.

4: Set f(X_{CIN}) < f(X_{IN})/3 when the sub-clock is used.

1.20 Electrical characteristics

Table 1.20.15 Electrical characteristics (7478 group)

(V_{CC} = 2.7 V to 5.5 V, Vss = AVss = 0 V, Ta = –20 °C to +85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions		Limits			Unit
				Min.	Typ.	Max.	
V_{OH}	"H" output voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$ $P4_0$ to $P4_3$	V_{CC} = 5 V, I_{OH} = –5 mA		3.0			V
		V_{CC} = 3 V, I_{OH} = –1.5 mA		2.0			V
V_{OL}	"L" output voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$ $P4_0$ to $P4_3$	V_{CC} = 5 V, I_{OL} = 10 mA				2.0	V
		V_{CC} = 3 V, I_{OL} = 3 mA				1.0	V
$V_{T+} - V_{T-}$	Hysteresis $P0_0$ to $P0_7$ $P3_0$ to $P3_3$	V_{CC} = 5 V			0.5		V
		V_{CC} = 3 V			0.3		V
$V_{T+} - V_{T-}$	Hysteresis $\overline{RESET}$	V_{CC} = 5 V			0.5		V
		V_{CC} = 3 V			0.3		V
$V_{T+} - V_{T-}$	Hysteresis $P1_4$/RxD $P1_6$/S_{CLK}	When used as RxD or S_{CLK}	V_{CC} = 5 V		0.5		V
			V_{CC} = 3 V		0.3		V
I_{IL}	"L" input current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_3$	V_I = 0 V Not use pull-up transistor	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
		V_I = 0 V Use pull-up transistor	V_{CC} = 5 V	–0.25	–0.5	–1.0	mA
			V_{CC} = 3 V	–0.08	–0.18	–0.35	mA
I_{IL}	"L" input current $P3_0$ to $P3_3$	V_I = 0V	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
I_{IL}	"L" input current $P2_0$ to $P2_7$	V_I = 0 V Not use as analog input	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
I_{IL}	"L" input current X_{IN}, $\overline{RESET}$	V_I = 0 V X_{IN} is at stop mode	V_{CC} = 5 V			–5	µA
			V_{CC} = 3 V			–3	µA
I_{IH}	"H" input current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_3$	V_I = V_{CC}, Not use pull-up transistor	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current $P3_0$ to $P3_3$	V_I = V_{CC}	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current $P2_0$ to $P2_7$	V_I = V_{CC} Not use as analog input	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{IH}	"H" input current X_{IN}, $\overline{RESET}$	V_I = V_{CC} X_{IN} is at stop mode	V_{CC} = 5 V			5	µA
			V_{CC} = 3 V			3	µA
I_{CC}	Power source current	At system operation, A-D conversion is not executed	f(X_{IN}) = 8 MHz	V_{CC} = 5 V		7	14	mA
			f(X_{IN}) = 4 MHz			3.5	7	mA
				V_{CC} = 3 V		1.8	3.6	mA
		At system operation, A-D conversion is executed	f(X_{IN}) = 8 MHz	V_{CC} = 5 V		7.5	15	mA
			f(X_{IN}) = 4 MHz			4	8	mA
				V_{CC} = 3 V		2	4	mA
		At low-speed mode, Ta = 25°C, low-power mode, f (X_{CIN}) = 32 kHz, A-D conversion is not executed		V_{CC} = 5 V		30	80	µA
				V_{CC} = 3 V		15	40	µA
		At wait mode	f(X_{IN}) = 8 MHz	V_{CC} = 5 V		2	4	mA
			f(X_{IN}) = 4 MHz			1	2	mA
				V_{CC} = 3 V		0.5	1	mA
		At wait mode, Ta = 25°C, low-power mode, f (X_{CIN}) = 32 kHz		V_{CC} = 5 V		3	12	µA
				V_{CC} = 3 V		2	8	µA
		At stop mode, V_{CC} = 5 V		Ta = 25 °C		0.1	1	µA
				Ta = 85 °C		1	10	µA
V_{RAM}	RAM retention voltage	Stop all oscillation		2.0		5.5	V

Table 1.20.16 A-D converter characteristics (7478 group)

(V_{CC} = 2.7 V to 5.5 V, Vss = AVss = 0 V, Ta = –20 °C to +85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
—	Resolution				8	bits
—	Absolute accuracy				±3	LSB
T_{CONV}	Conversion time	V_{CC} = 4.5 V to 5.5 V, f(X_{IN}) = 8 MHz			12.5	µs
		V_{CC} = 2.7 V to 5.5 V, f(X_{IN}) = 4 MHz			25	µs
V_{REF}	Reference input voltage		0.5V_{CC} (Note)		V_{CC}	V
R_{LADDER}	Ladder resistance value		2	5	10	kΩ
V_{IA}	Analog input voltage		0		V_{REF}	V

Note : Set the V_{REF} voltage to 0.5 V or more and 2 V or more. When using no A-D converter, connect it to V_{CC}.

1.20 Electrical characteristics

1.20.2 Timing requirements, switching characteristics

(1) 7470/7471 group timing requirements, switching characteristics

Table 1.20.17 shows the timing requirements and switching characteristics of the 7470/7471 group. Figure 1.20.1 shows the timing chart.

Table 1.20.17 Timing requirements and switching characteristics (7470/7471 group)

(V_{CC} = 4.0 V to 5.5 V, V_{SS} = 0V, Ta = –20 °C to +85°C, f(X_{IN}) = 4 MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
t_c(CLK)	Serial I/O clock input cycle time	1000			ns
t_{WH}(CLK)	Serial I/O clock input “H” pulse width	400			ns
t_{WL}(CLK)	Serial I/O clock input “L” pulse width	400			ns
t_{su}(SIN–CLK)	Serial I/O input set up time	200			ns
t_h(CLK-SIN)	Serial I/O input hold time	200			ns
t_d(CLK-SOUT)	Serial I/O output delay time			150	ns

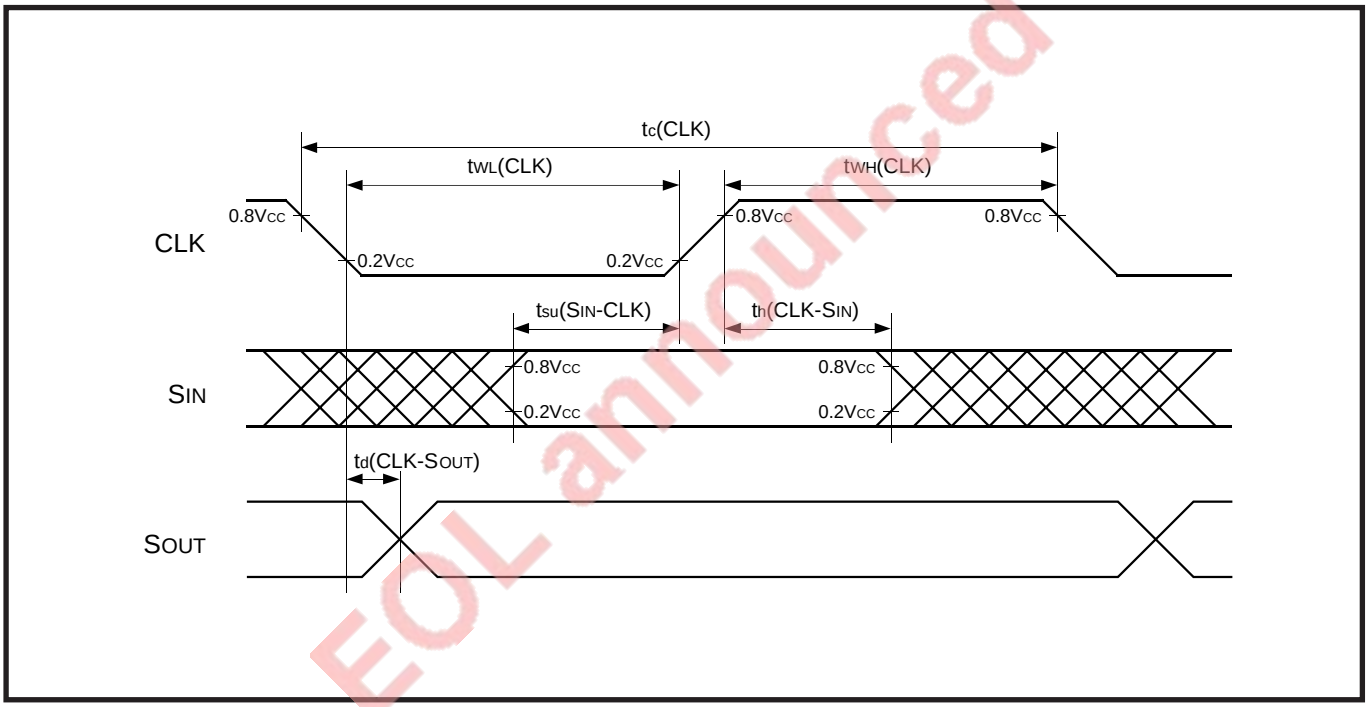


Fig. 1.20.1 Timing chart (7470/7471 group)

(2) 7477/7478 group timing requirements, switching characteristics

Table 1.20.18 shows the timing requirements and switching characteristics of the 7477/7478 group. Figure 1.20.2 shows the timing chart.

Table 1.20.18 Timing requirements and switching characteristics (7477/7478 group)

(V_{CC} = 4.5 V to 5.5 V, V_{SS} = 0V, Ta = –20 °C to +85°C, f(X_{IN}) = 8 MHz)

	Symbol	Parameter	Limits			Unit
			Min.	Typ.	Max.	
Clock synchronous	t_c(SCLK)	Serial I/O clock input cycle time	2000			ns
	t_{WH}(SCLK)	Serial I/O clock input “H” pulse width	880			ns
	t_{WL}(SCLK)	Serial I/O clock input “L” pulse width	880			ns
	t_{su}(RxD–SCLK)	Serial I/O input set up time	160			ns
	t_h(SCLK-RxD)	Serial I/O input hold time	80			ns
	t_d(SCLK-TxD)	Serial I/O output delay time			100	ns
Clock asynchronous	t_c(SCLK)	Serial I/O clock input cycle time	500			ns
	t_{WH}(SCLK)	Serial I/O clock input “H” pulse width	220			ns
	t_{WL}(SCLK)	Serial I/O clock input “L” pulse width	220			ns

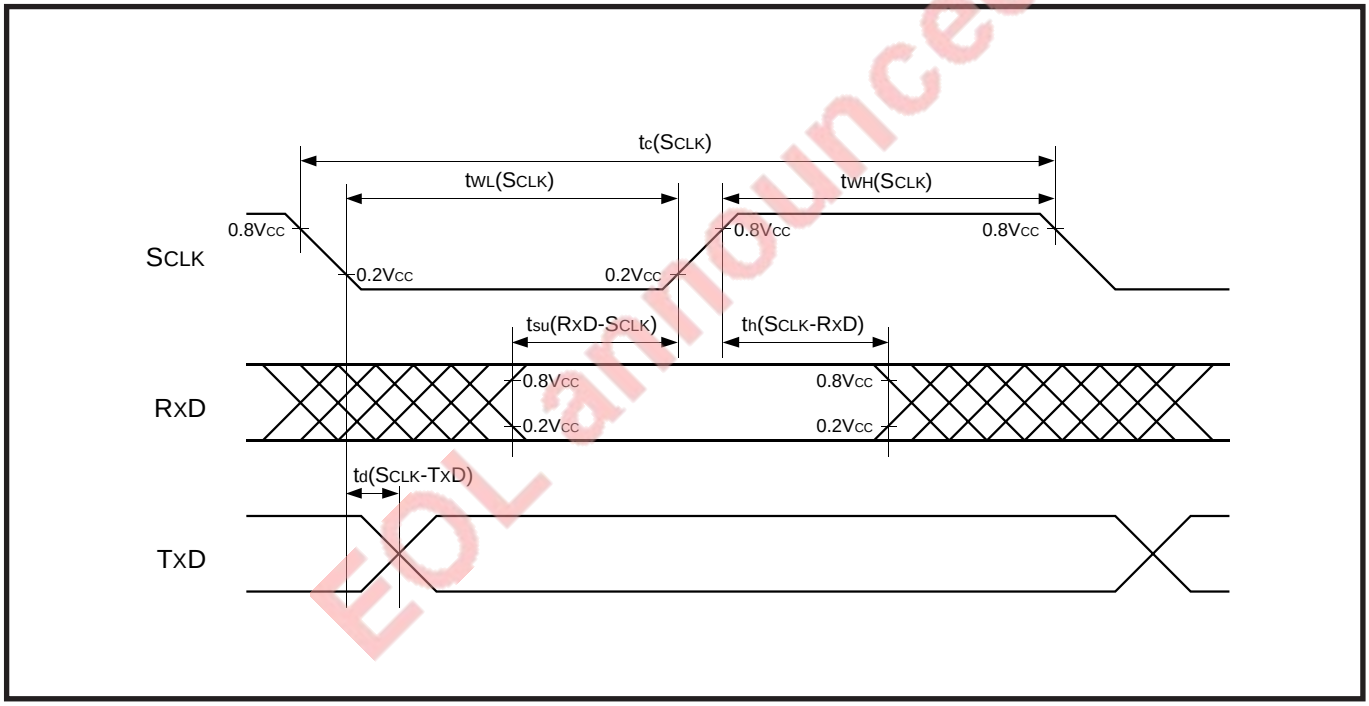


Fig. 1.20.2 Timing chart (7477/7478 group)

1.20 Electrical characteristics

1.20.3 Power source current standard characteristics

The power source current standard characteristics described in this section are mentioned as an characteristic example of the 7470/7471/7477/7478 group but not guaranteed by us. For standard values, refer to **"1.20.1 Electrical characteristics."**

Figure 1.20.3 shows the power source current standard characteristics measuring circuit.

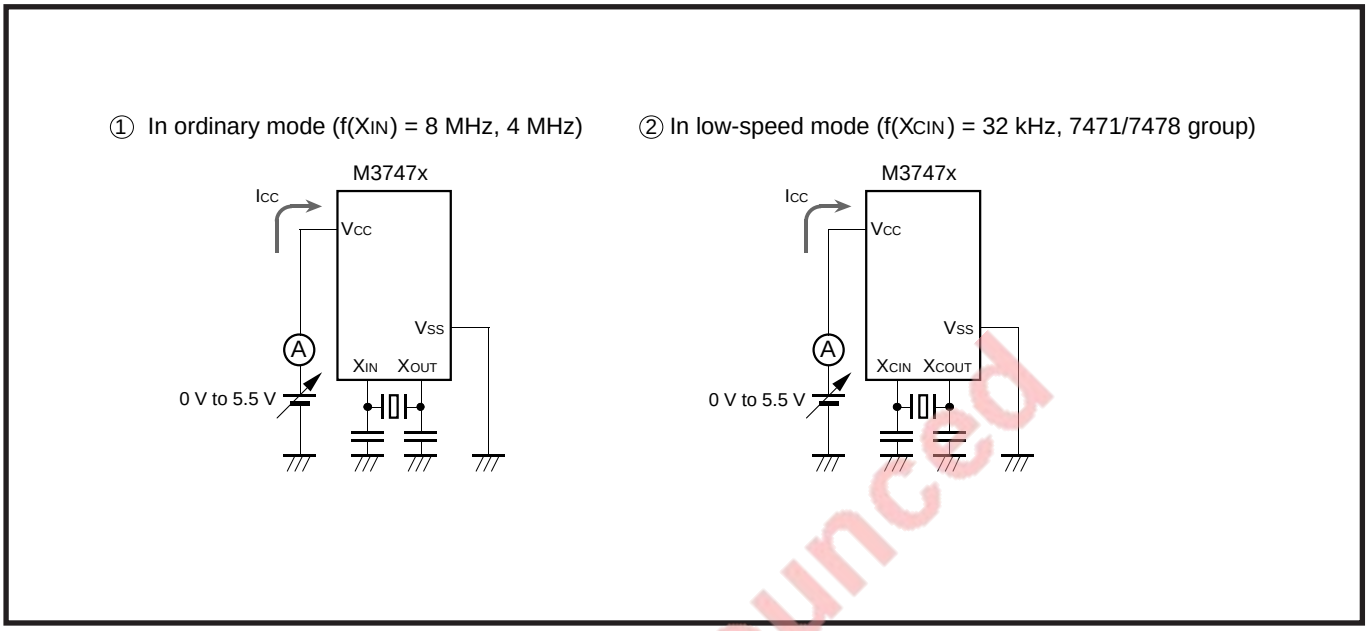


Fig. 1.20.3 Power source current standard characteristics measuring circuit

(1) 7470/7471 group power source current standard characteristics

Figure 1.20.4 to Figure 1.20.6 show the Icc - Vcc characteristics of the 7470/7471 group.

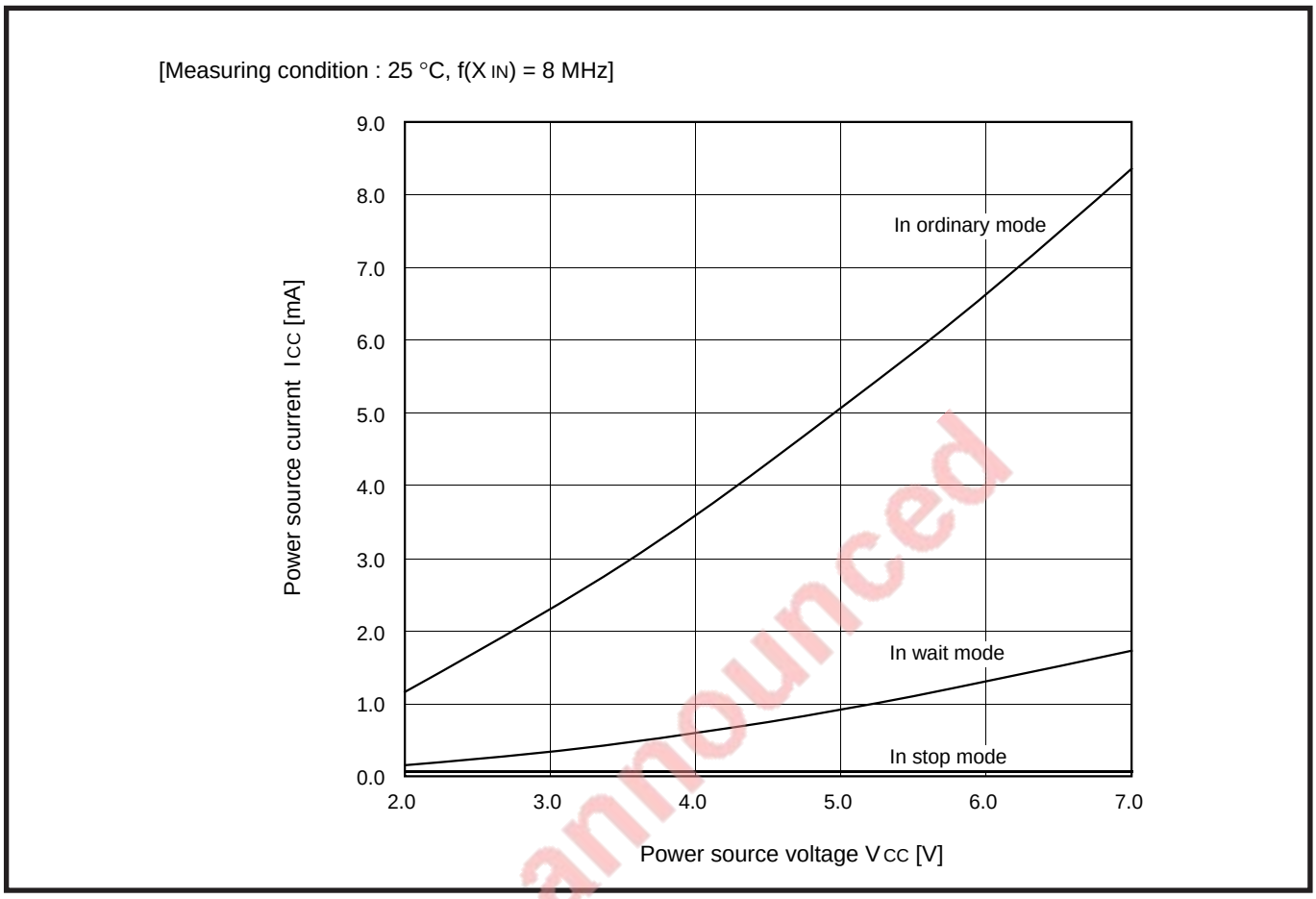


Fig. 1.20.4 Icc – Vcc characteristics (f(XIN) = 8 MHz, 7470/7471 group)

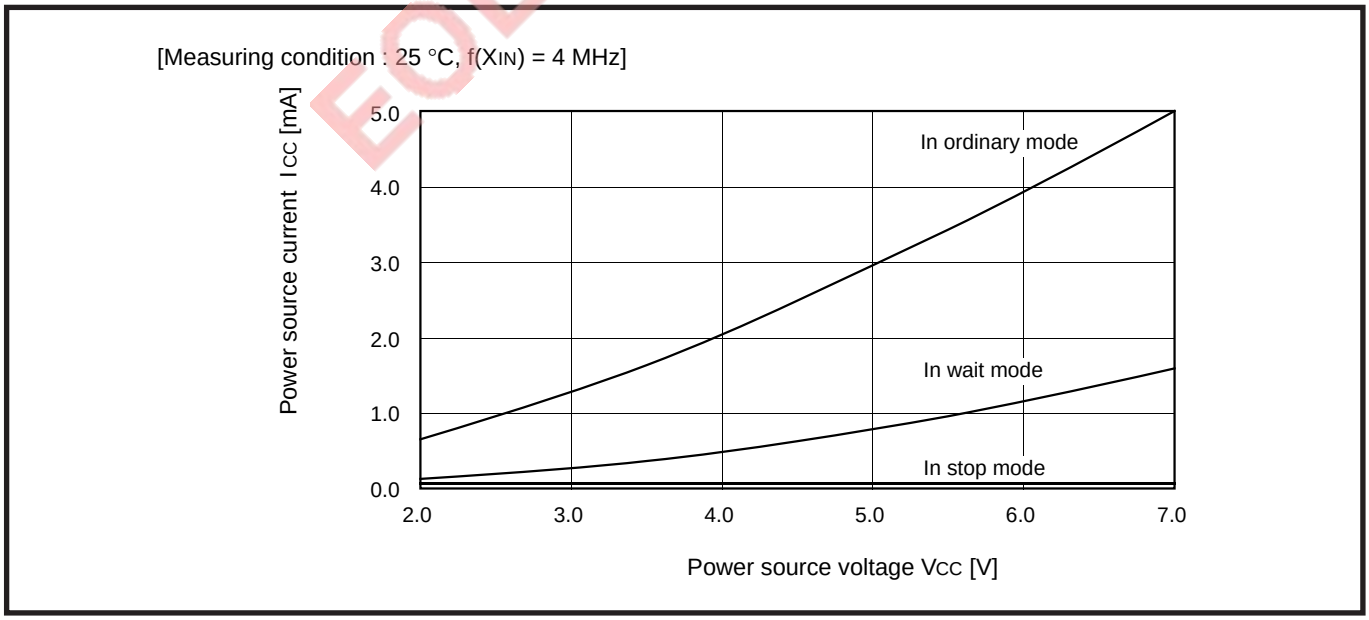


Fig. 1.20.5 Icc – Vcc characteristics (f(XIN) = 4 MHz, 7470/7471 group)

1.20 Electrical characteristics

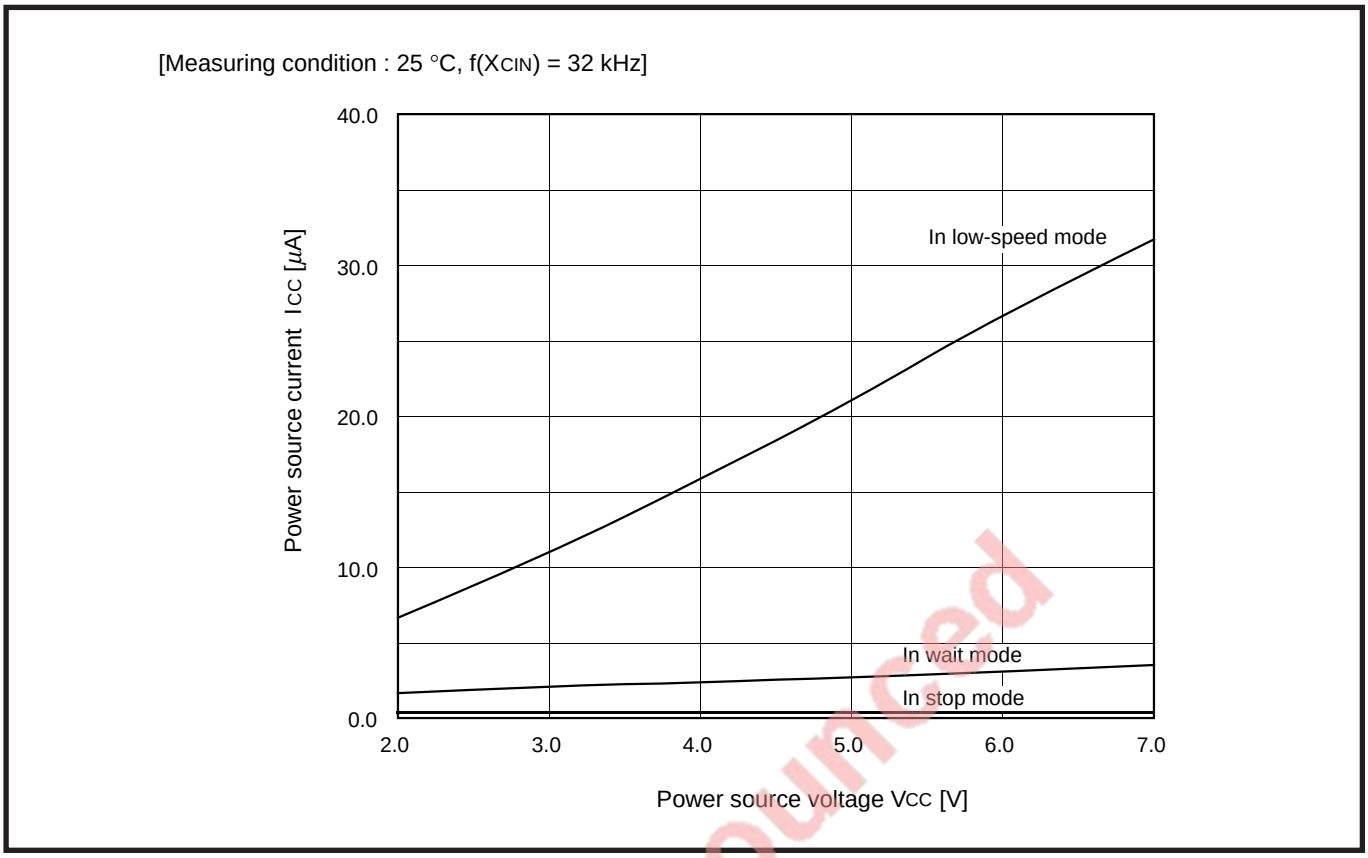


Fig. 1.20.6 ICC – VCC characteristics (f(XCIN) = 32 kHz, 7471 group)

(2) 7477/7478 group power source current standard characteristics

Figure 1.20.7 to Figure 1.20.9 show the Icc - Vcc characteristics of the 7477/7478 group.

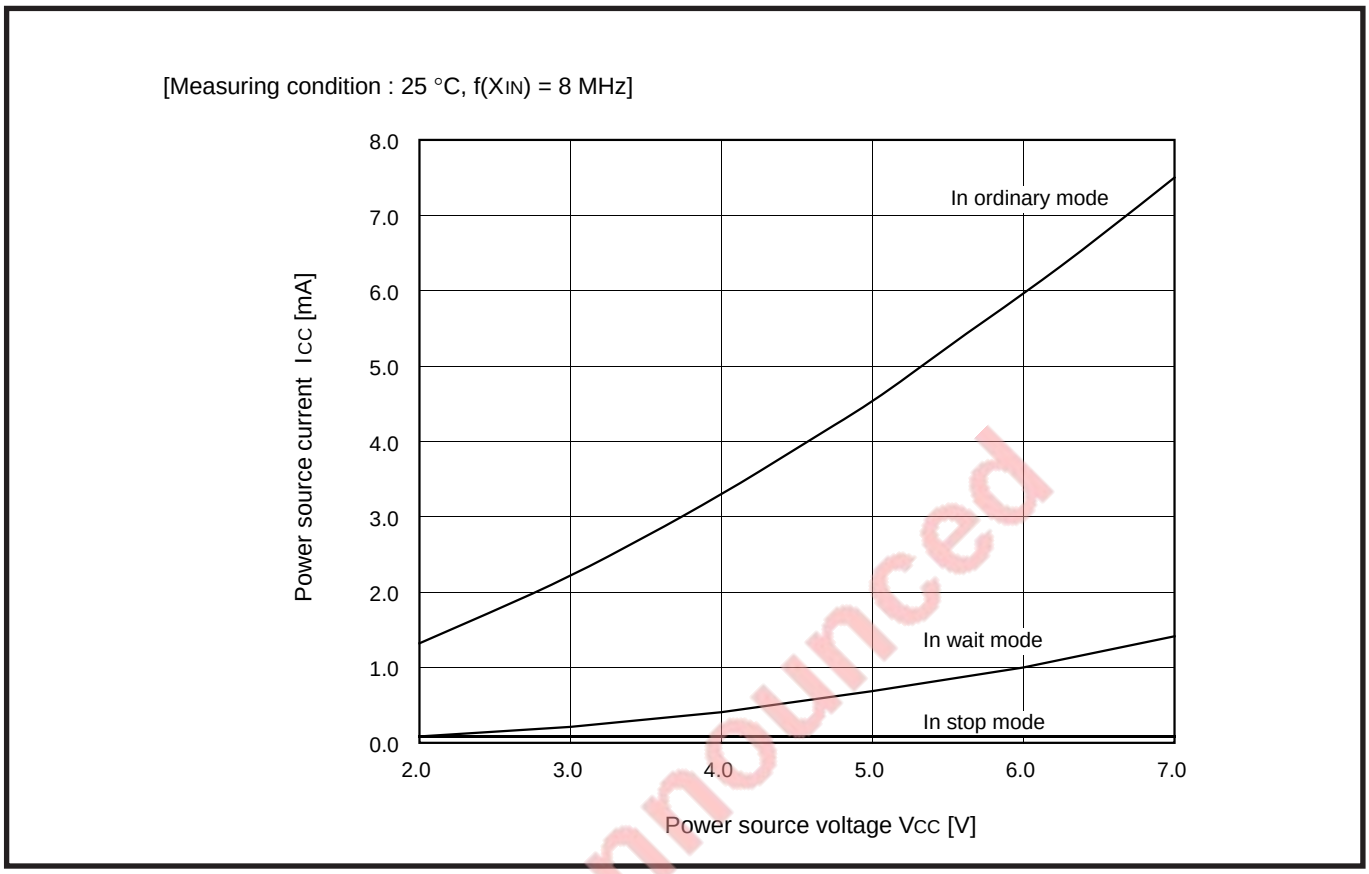


Fig. 1.20.7 Icc – Vcc characteristics (f(XIN) = 8 MHz, 7477/7478 group)

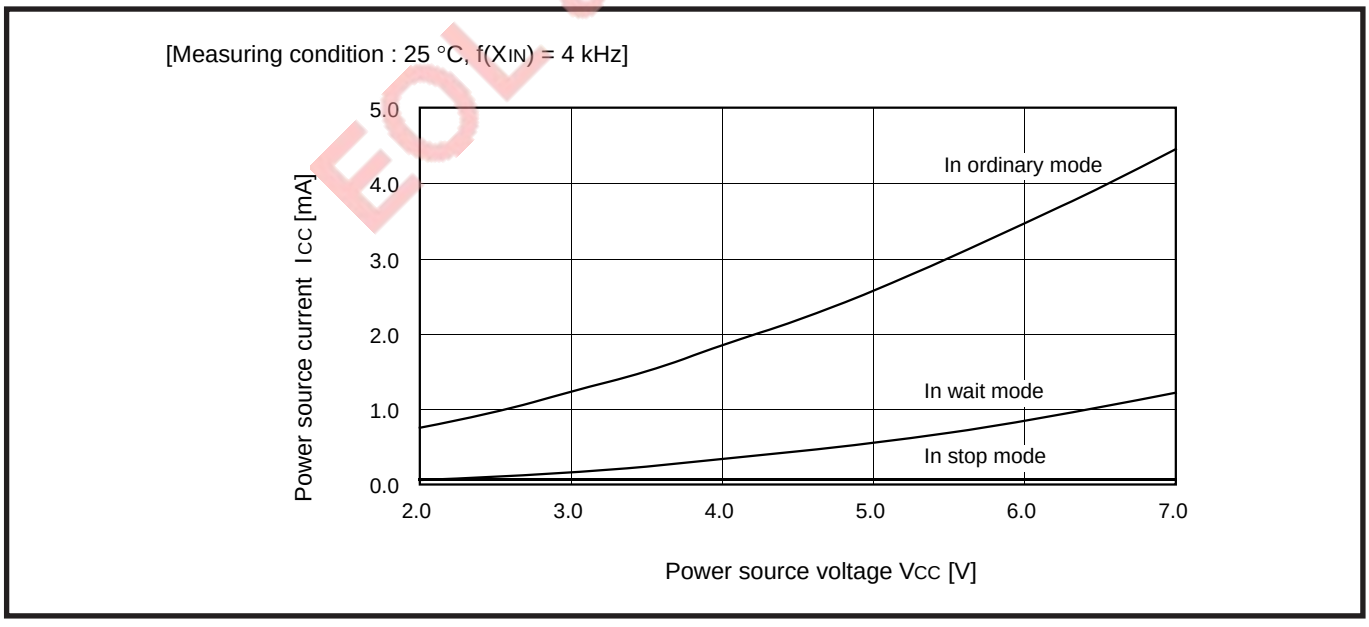


Fig. 1.20.8 Icc – Vcc characteristics (f(XIN) = 4 MHz, 7477/7478 group)

1.20 Electrical characteristics

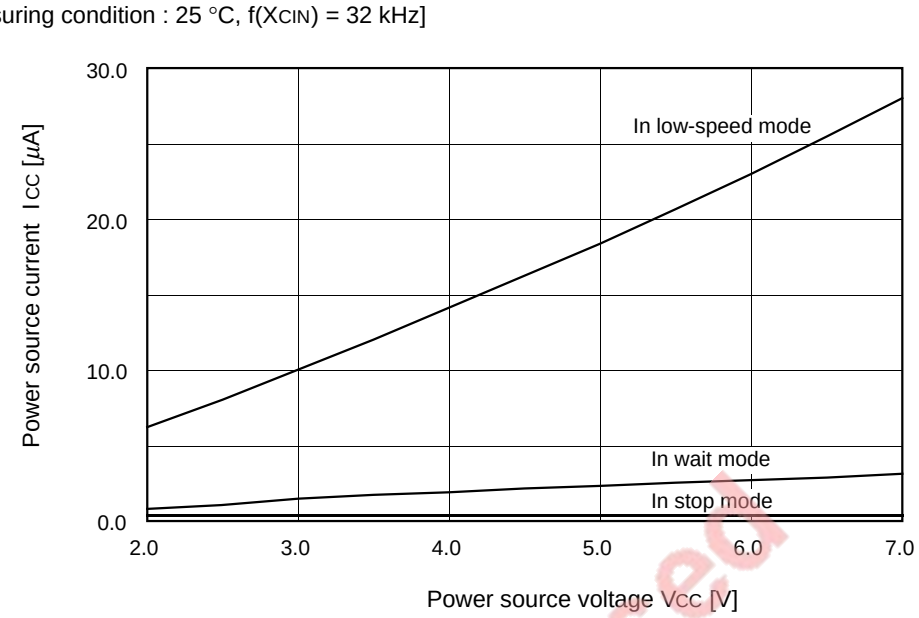


Fig. 1.20.9 ICC – VCC characteristics (f(XCIN) = 32 kHz, 7478 group)

1.20.4 Port standard characteristic

The port standard characteristics described in this section are mentioned as a characteristic example of the 7470/7471/7477/7478 group but not guaranteed by us. For standard values, refer to **"1.20.1 Electrical characteristics."**

Figure 1.20.10 shows the port standard characteristic measuring circuits.

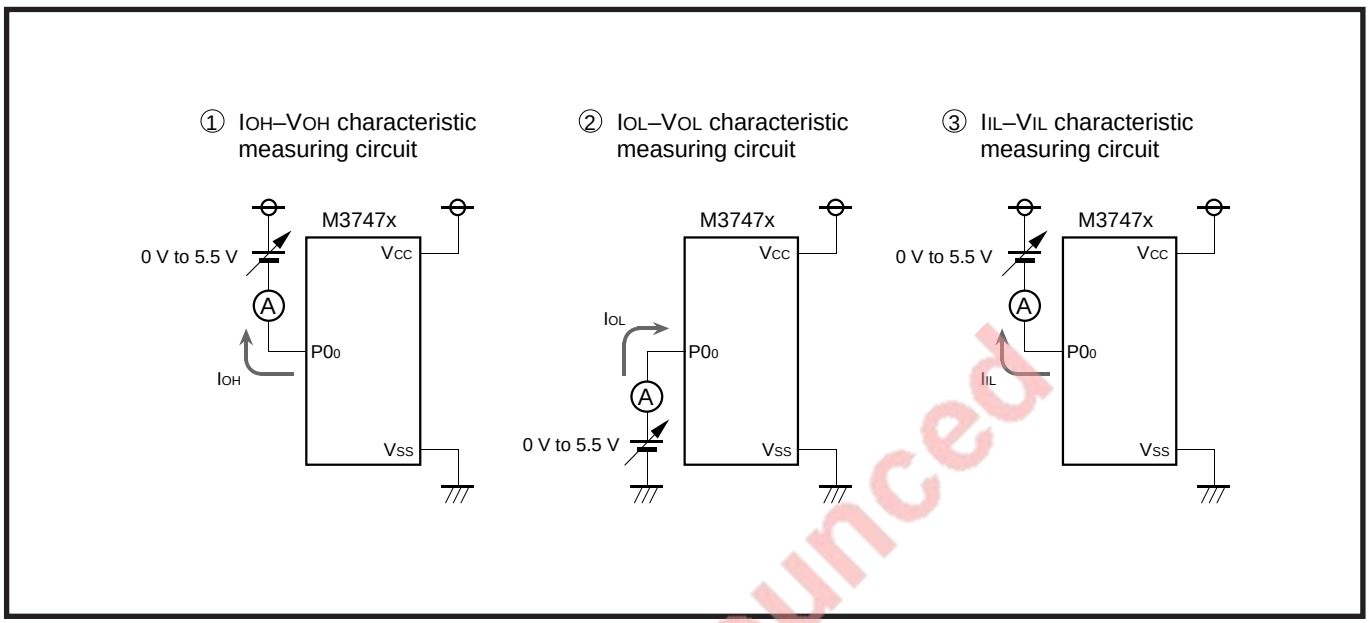


Fig. 1.20.10 Port standard characteristic measuring circuits

1.20 Electrical characteristics

(1) 7470/7471 group port standard characteristic

Figure 1.20.11 to Figure 1.20.13 show the port standard characteristics of the 7470/7471 group.

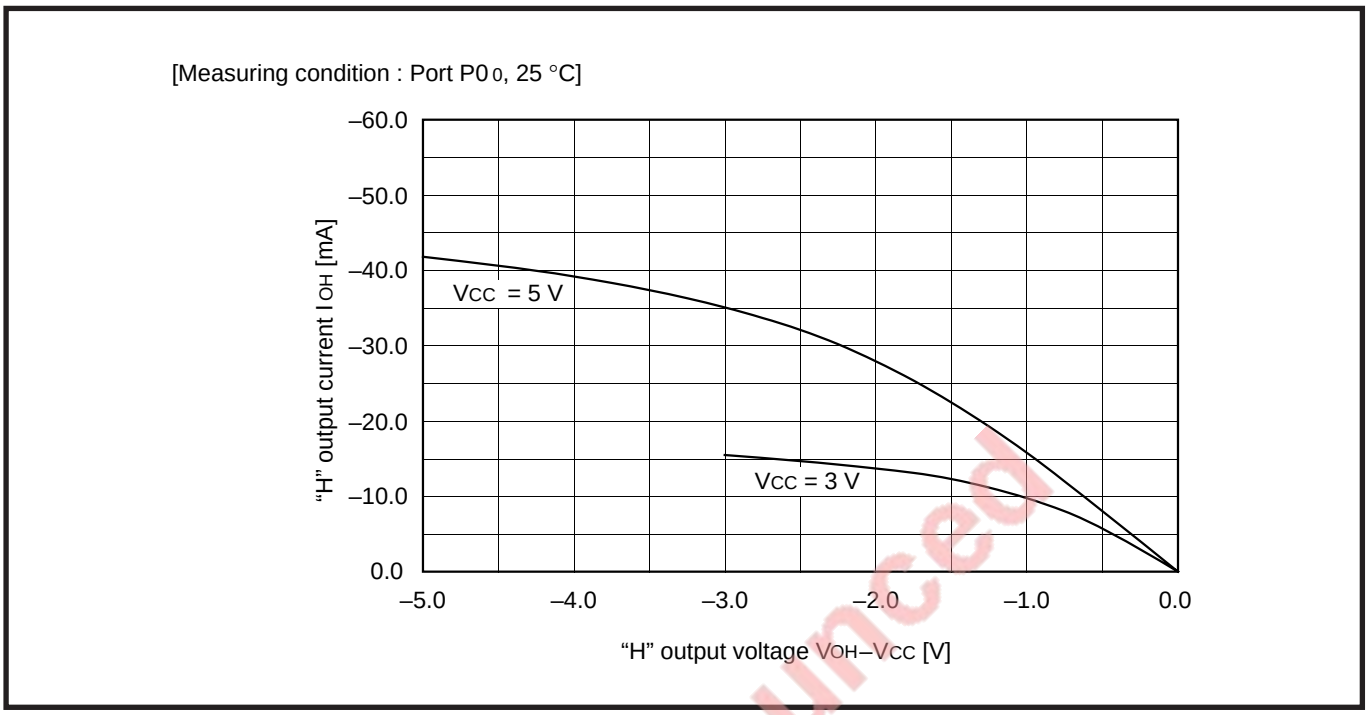


Fig. 1.20.11 I_{OH} – V_{OH} characteristics of programmable I/O port (CMOS output) P-channel side (7470/7471 group)

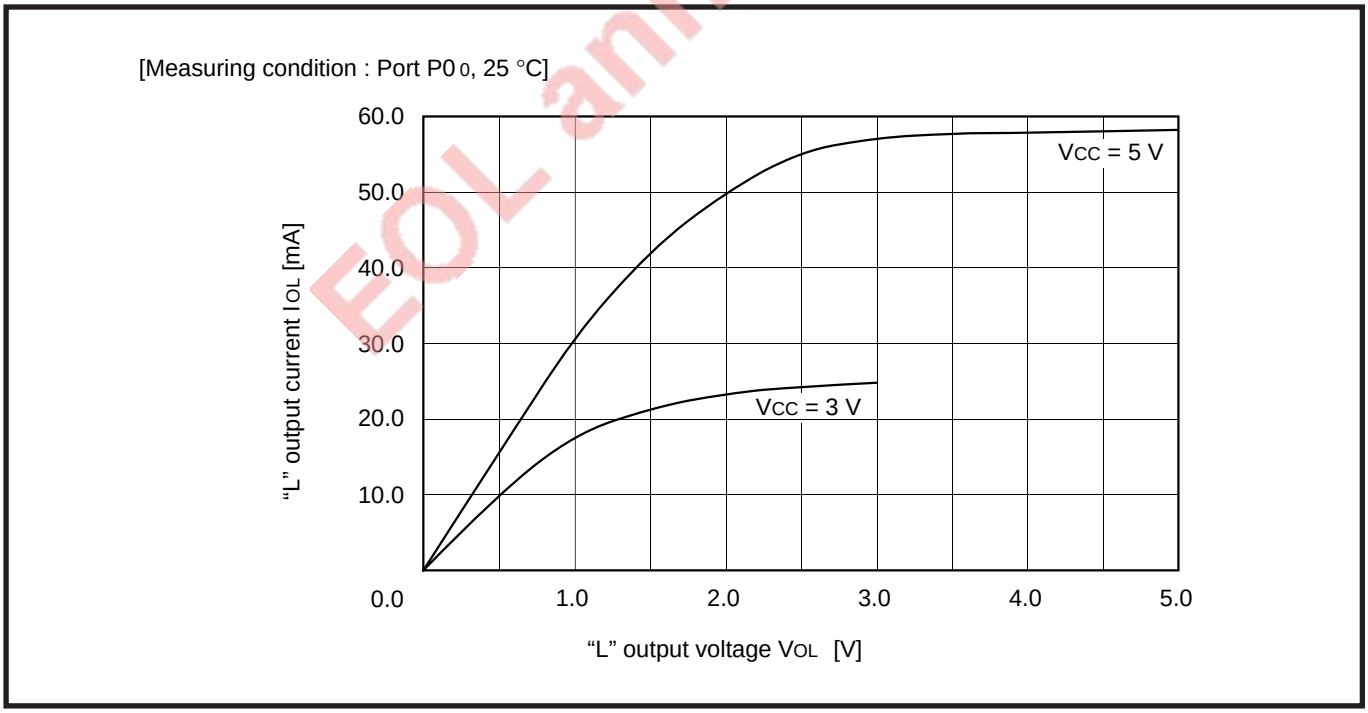


Fig. 1.20.12 I_{OL} – V_{OL} characteristics of programmable I/O port (CMOS output) N-channel side (7470/7471 group)

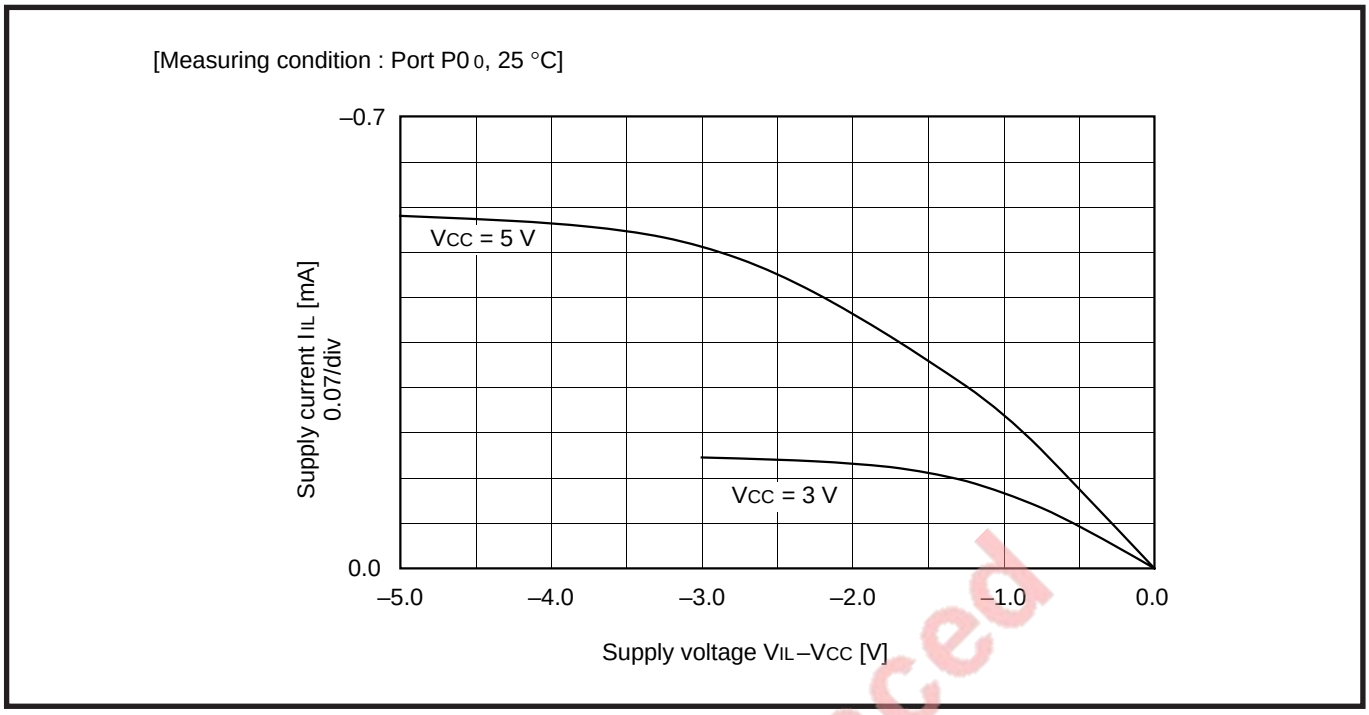


Fig. 1.20.13 I_{IL} – V_{IL} characteristics of programmable I/O port (CMOS output) pull-up transistor (7470/7471 group)

1.20 Electrical characteristics

(2) 7477/7478 group port standard characteristic

Figure 1.20.14 to Figure 1.20.16 show the port standard characteristics of the 7477/7478 group.

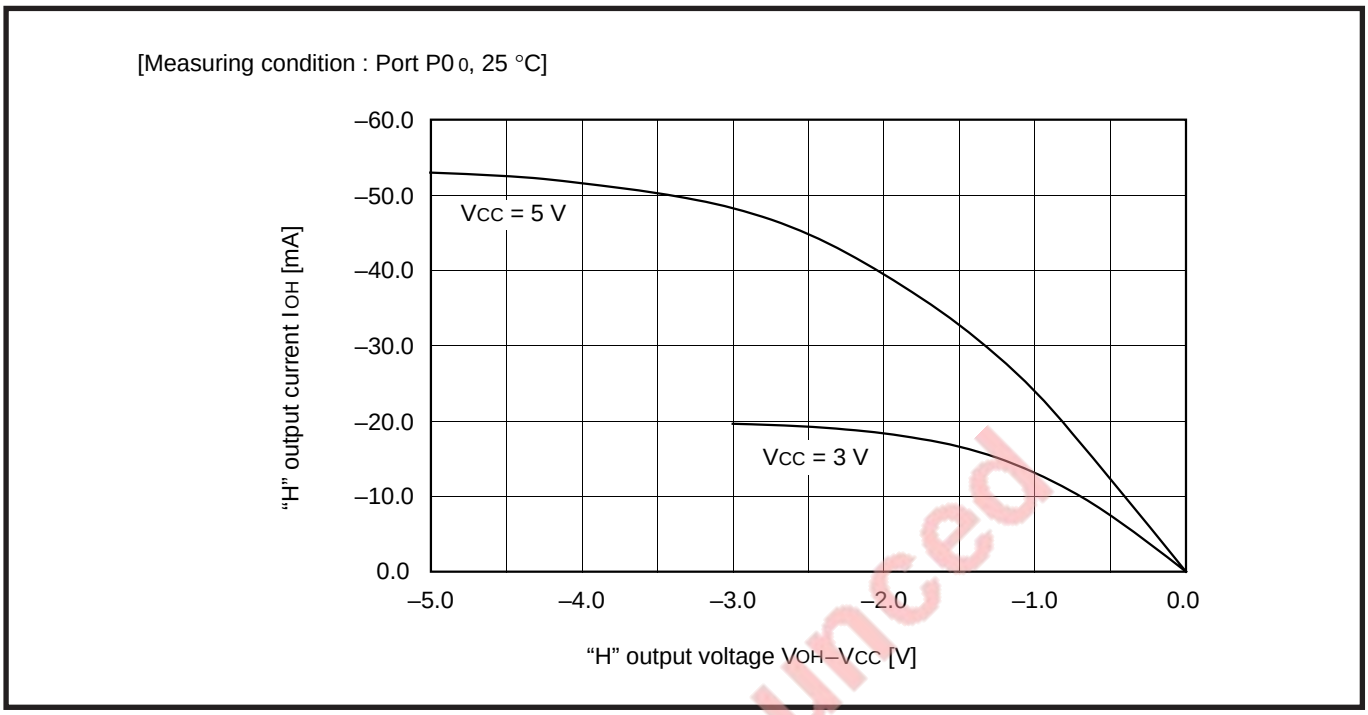


Fig. 1.20.14 I_{OH} – V_{OH} characteristics of programmable I/O port (CMOS output) P-channel side (7477/7478 group)

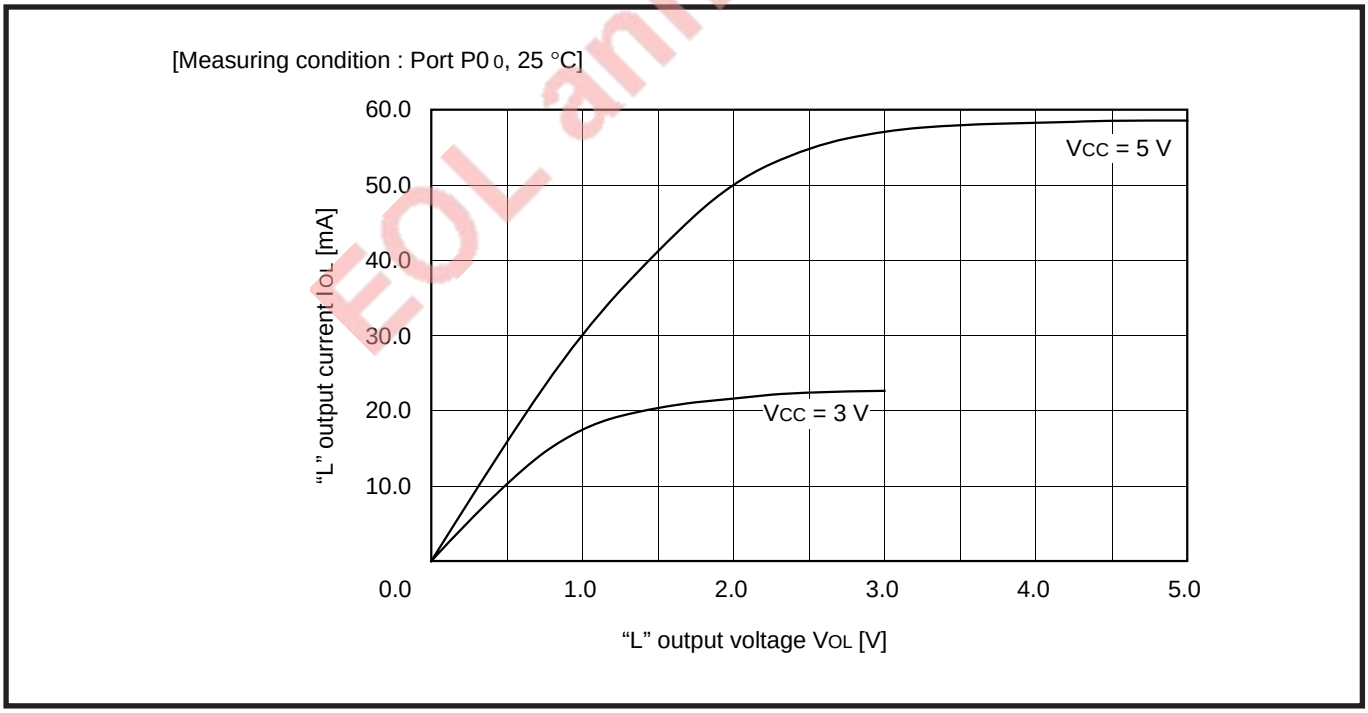


Fig. 1.20.15 I_{OL} – V_{OL} characteristics of programmable I/O port (CMOS) N-channel side (7477/7478 group)

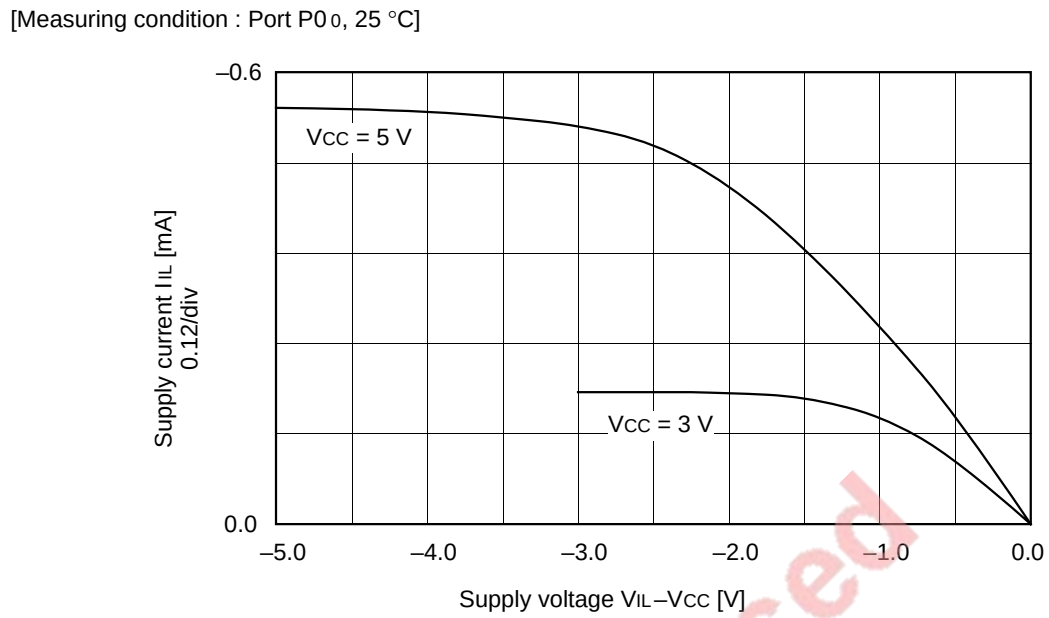


Fig. 1.20.16 I_{IL} – V_{IL} characteristics of programmable I/O port (CMOS output) pull-up transistor (7477/7478 group)

1.20 Electrical characteristics

1.20.5 A-D conversion standard characteristic

(1) Relative precision (7470/7471 group)

Figure 1.20.17 to Figure 1.20.18 show the A-D conversion standard characteristics on the relative precision of the 7470/7471 group.

In the graph, the lower line indicates a deviation from the ideal value at the point where the output code changes, namely, relative precision error (ERROR). For example, in Figure 1.20.17, the change of "$3F_{16}$ to 40_{16}" of the output code occurs ideally at the point of IN_0 = 757.32 mV. However, since the relative precision error is -3.567 mV, "757.32 - 3.567 = 753.753 mV" represents a measuring change point.

In the graph, the upper line indicates an input voltage width (1 LSB WIDTH) in which the output code is the same. For example, in Figure 1.20.17, since the measured value of input voltage width, when the output code is "$3F_{16}$", is 10.701 mV, the differential nonlinear error on the relative precision represents "10.701 -11.89 = -1.189 mV (-0.1 LSB)".

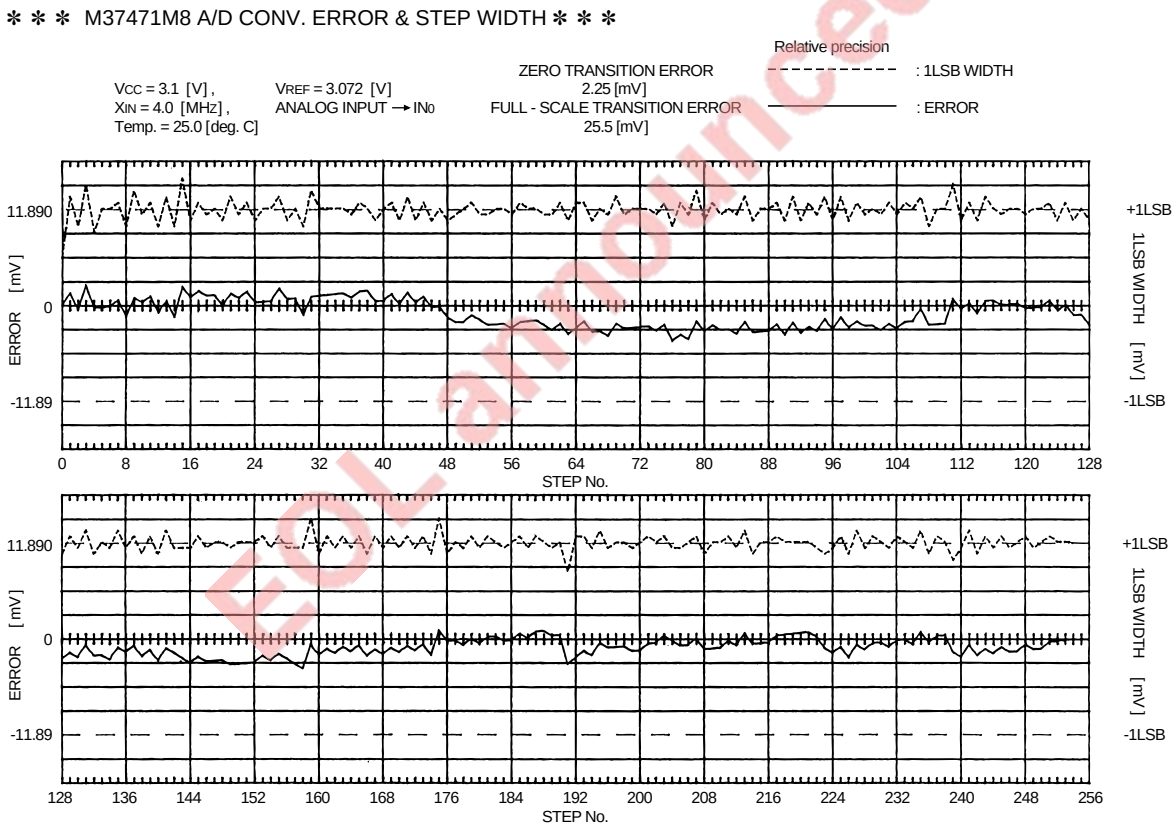


Fig. 1.20.17 A-D conversion standard characteristics, relative precision error (1)

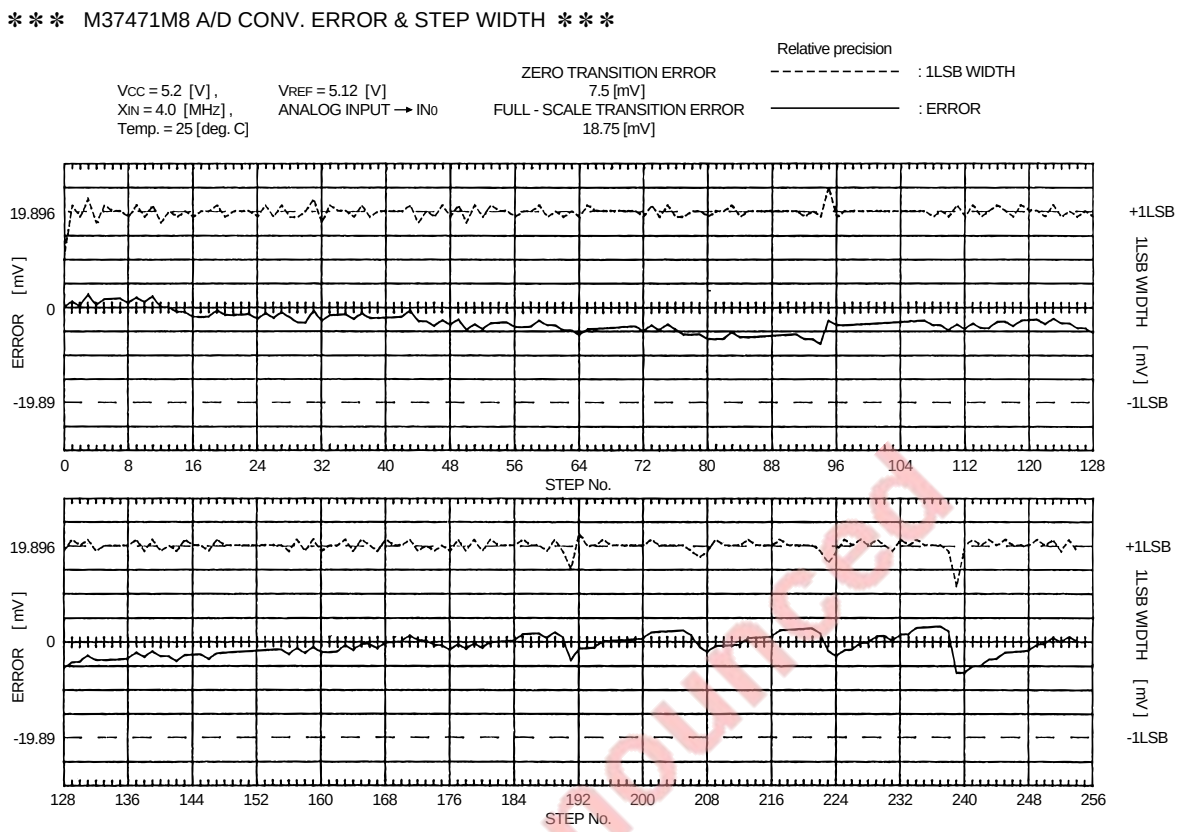


Fig. 1.20.18 A-D conversion standard characteristics, relative precision error (2)

1.20 Electrical characteristics

(2) Absolute precision

Figure 1.20.19 to Figure 1.20.23 show the A-D conversion standard characteristics on the absolute precision of the 7470/7471/7477/7478 group.

In the graph, the lower line indicates a deviation from the ideal value at the point where the output code changes, namely, absolute precision error (ERROR). For example, in Figure 1.20.17, the change of "$3F_{16}$ to 40_{16}" of the output code occurs ideally at the point of IN_0 = 762 mV. However, since the absolute precision error is -8.4 mV, "762 - 8.4 = 753.6 mV represents the measuring change point.

In the graph, the upper line indicates an input voltage with (1 LSB WIDTH) in which the output code is the same. For example, since the measured value of input voltage width, when the output code is "$3F_{16}$", is 10.8 mV, the differential nonlinear error represents "10.8 - 12 = -1.2 mV (-0.1 LSB).

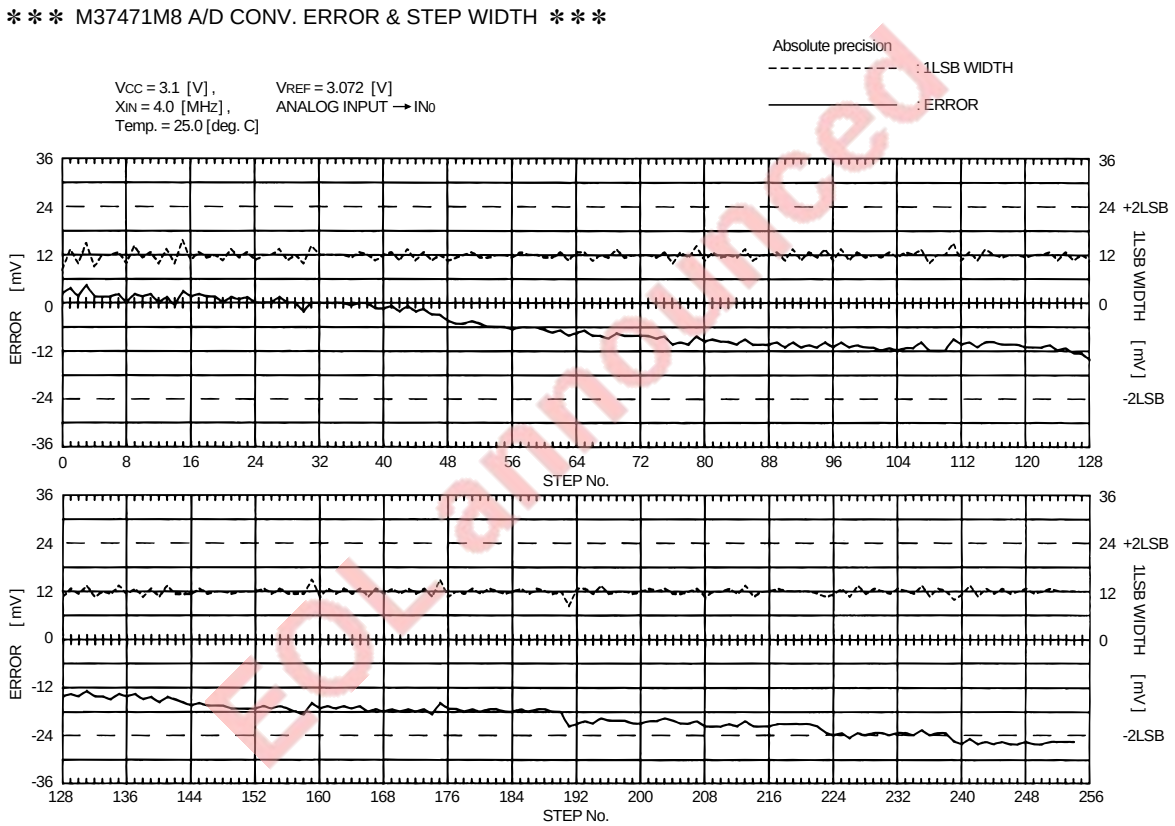


Fig. 1.20.19 A-D conversion standard characteristics, absolute precision error (1)

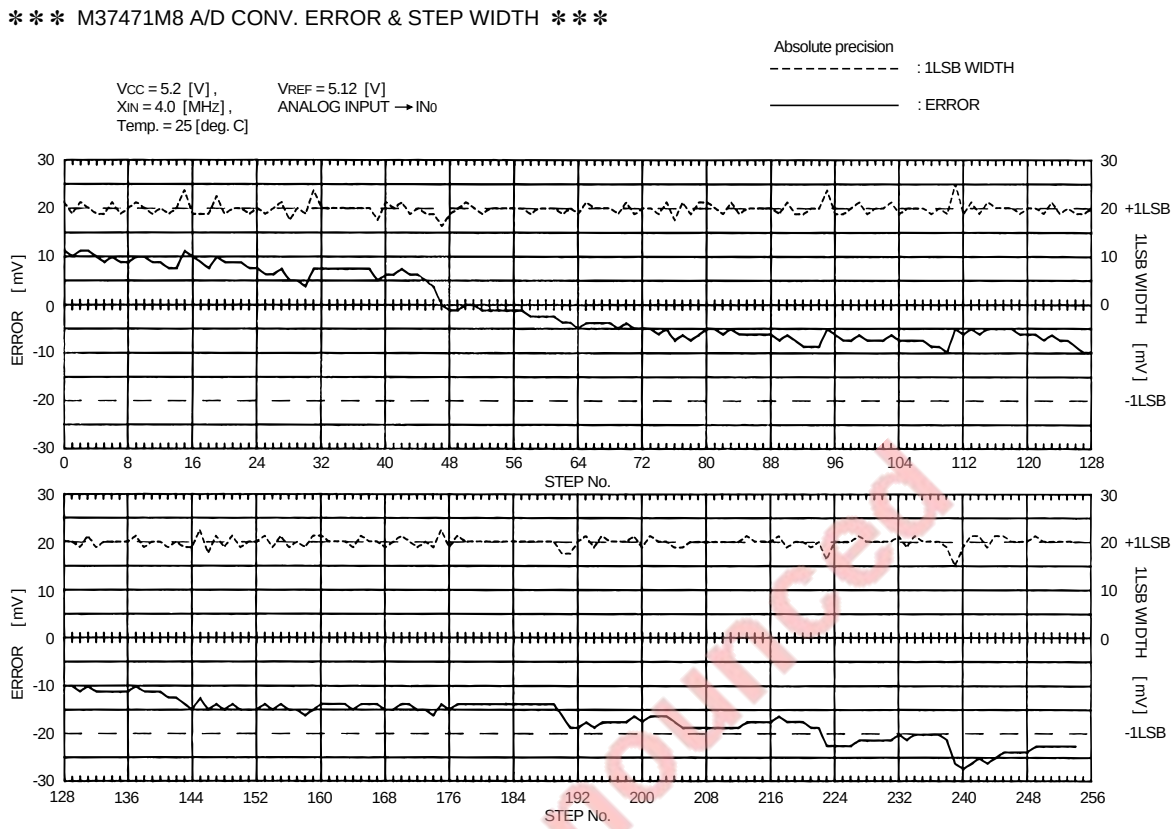


Fig. 1.20.20 A-D conversion standared characteristics, absolute precision error (2)

1.20 Electrical characteristics

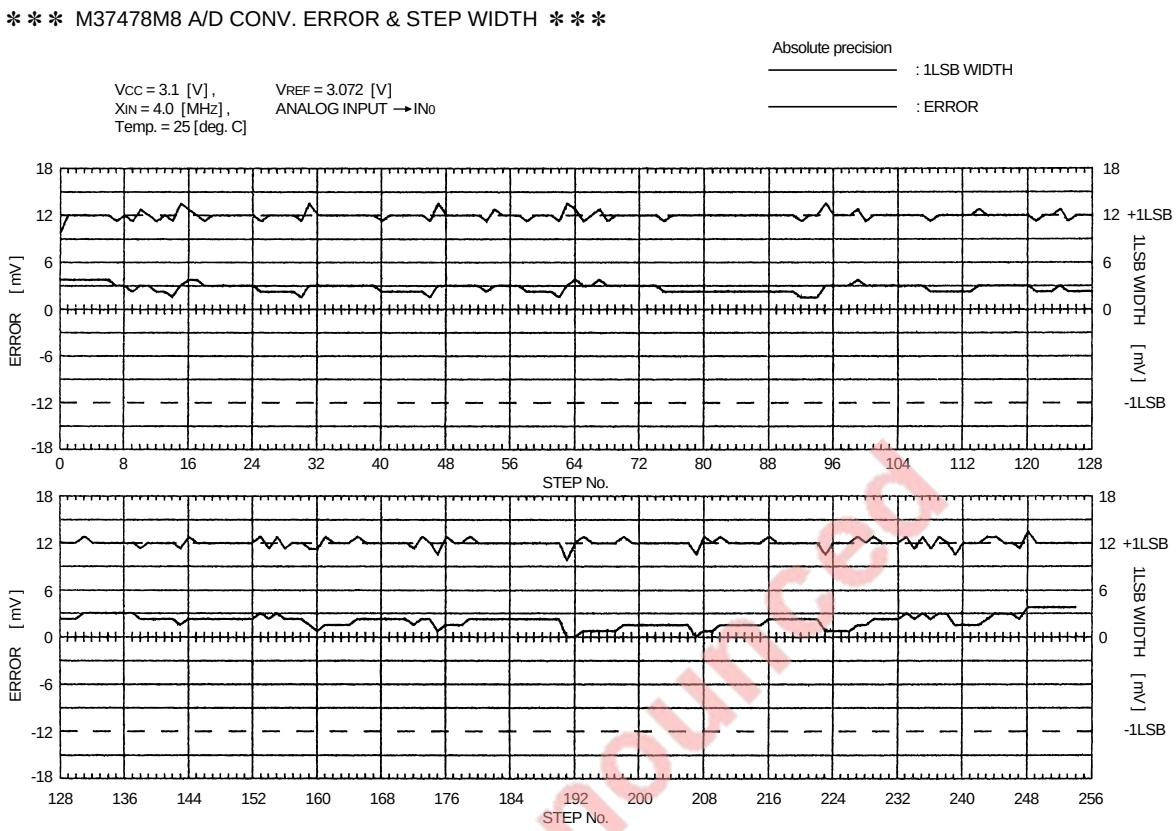


Fig. 1.20.21 A-D conversion standard characteristics, absolute precision error (3)

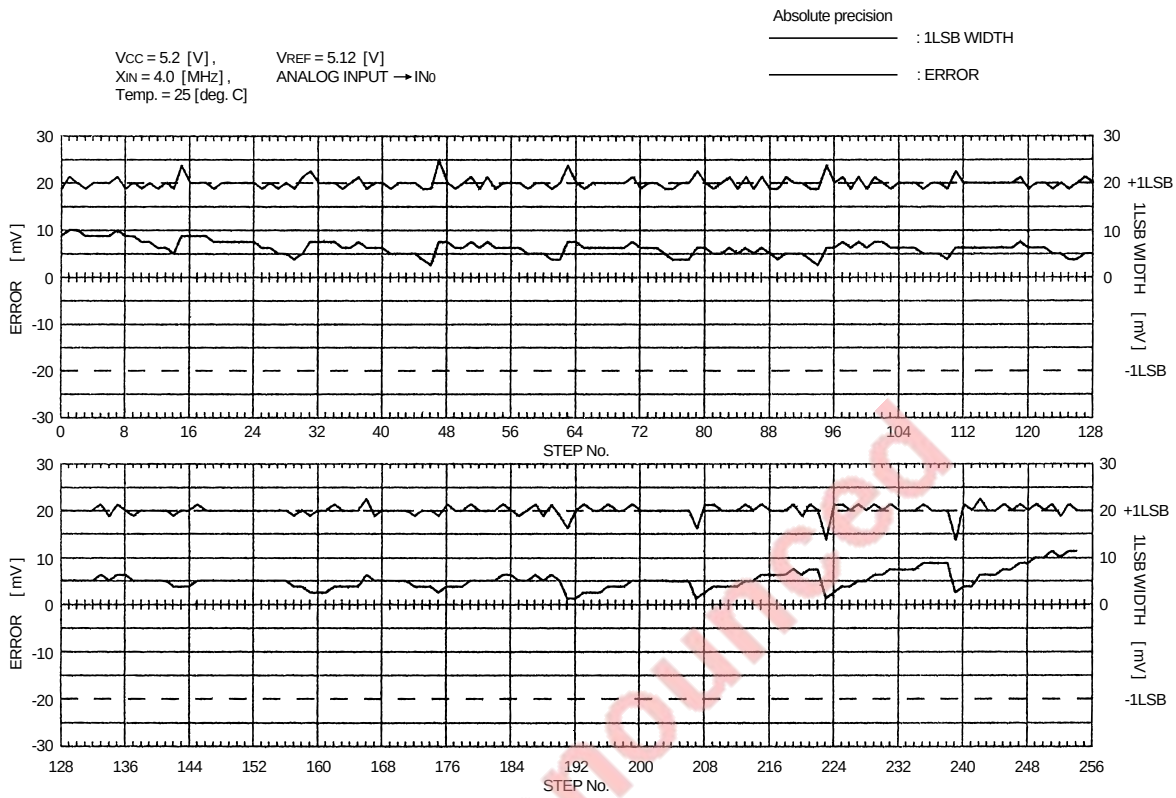


Fig. 1.20.22 A-D conversion standard characteristics, absolute precision error (4)

1.20 Electrical characteristics

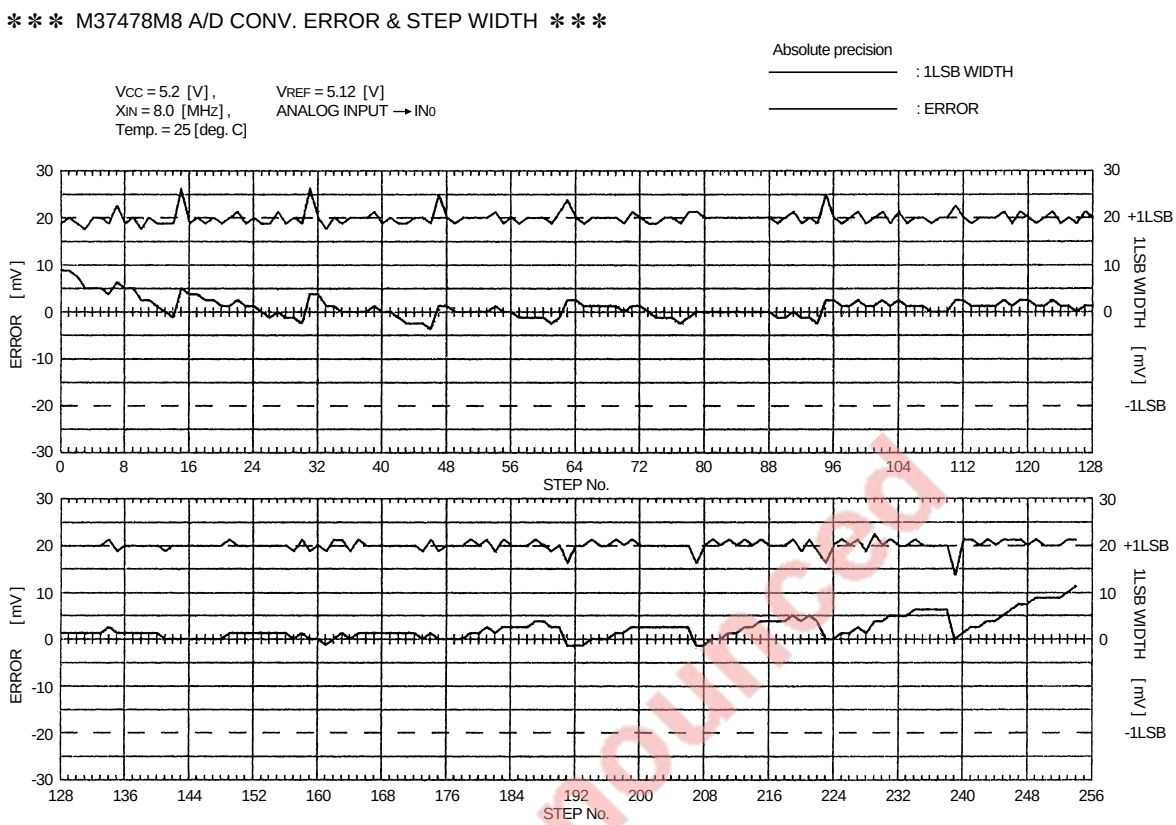


Fig. 1.20.23 A-D conversion standard characteristics, absolute precision error (5)

CHAPTER 2

APPLICATION

2.1 I/O pins

7470	7471	7477	7478
○	○	○	○

2.1.1 I/O port

(1) Port register

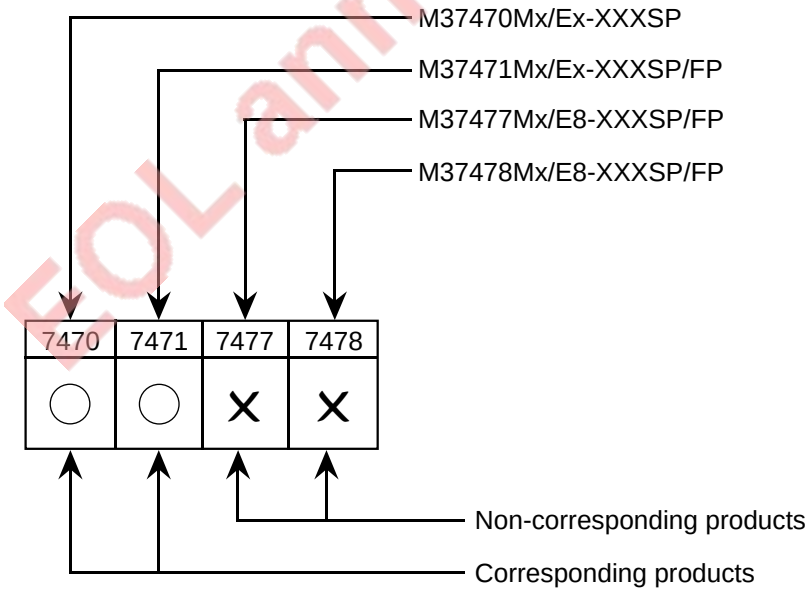
Table 2.1.1 shows a memory allocation of port register corresponding to each port.

Table 2.1.1 Port register memory allocation

Port	Address of port register			
	7470 group	7471 group	7477 group	7478 group
P0	$00C0_{16}$	$00C0_{16}$	$00C0_{16}$	$00C0_{16}$
P1	$00C2_{16}$	$00C2_{16}$	$00C2_{16}$	$00C2_{16}$
P2	$00C4_{16}$	$00C4_{16}$	$00C4_{16}$	$00C4_{16}$
P3	$00C6_{16}$	$00C6_{16}$	$00C6_{16}$	$00C6_{16}$
P4	$00C8_{16}$	$00C8_{16}$	$00C8_{16}$	$00C8_{16}$
P5	–	$00CA_{16}$	–	$00CA_{16}$

Note: In the 7470/7477 group, P2 is 4 bits of b0 - b3 and P4 is 2 bits of b0 and b1.
In the 7471/7478 group, P5 is 4 bits of b0 - b3.

In Chapter 2, each page describes the corresponding products by using the following table.



(2) Port Pi direction register (i = 0 to 5)

7470	7471	7477	7478
○	○	○	○

Switching between input and output for programmable I/O ports is performed by the port direction register corresponding to each port. Table 2.1.2 shows a memory allocation of the port direction register corresponding to each port and Figure 2.1.1 shows an example of port direction register setting.

Table 2.1.2 Port direction register memory allocation

Port	Address of port direction register			
	7470 group	7471 group	7477 group	7478 group
P0	$00C1_{16}$	$00C1_{16}$	$00C1_{16}$	$00C1_{16}$
P1	$00C3_{16}$	$00C3_{16}$	$00C3_{16}$	$00C3_{16}$
P2	$00C5_{16}$	$00C5_{16}$	–	–
P3	–	–	–	–
P4	$00C9_{16}$	$00C9_{16}$	$00C9_{16}$	$00C9_{16}$
P5	–	–	–	–

Note: In the 7470 group, P2 is 4 bits of b0 - b3 and P4 is 2 bits of b0 and b1.
In the 7477 group, P4 is 2 bits of b0 and b1.

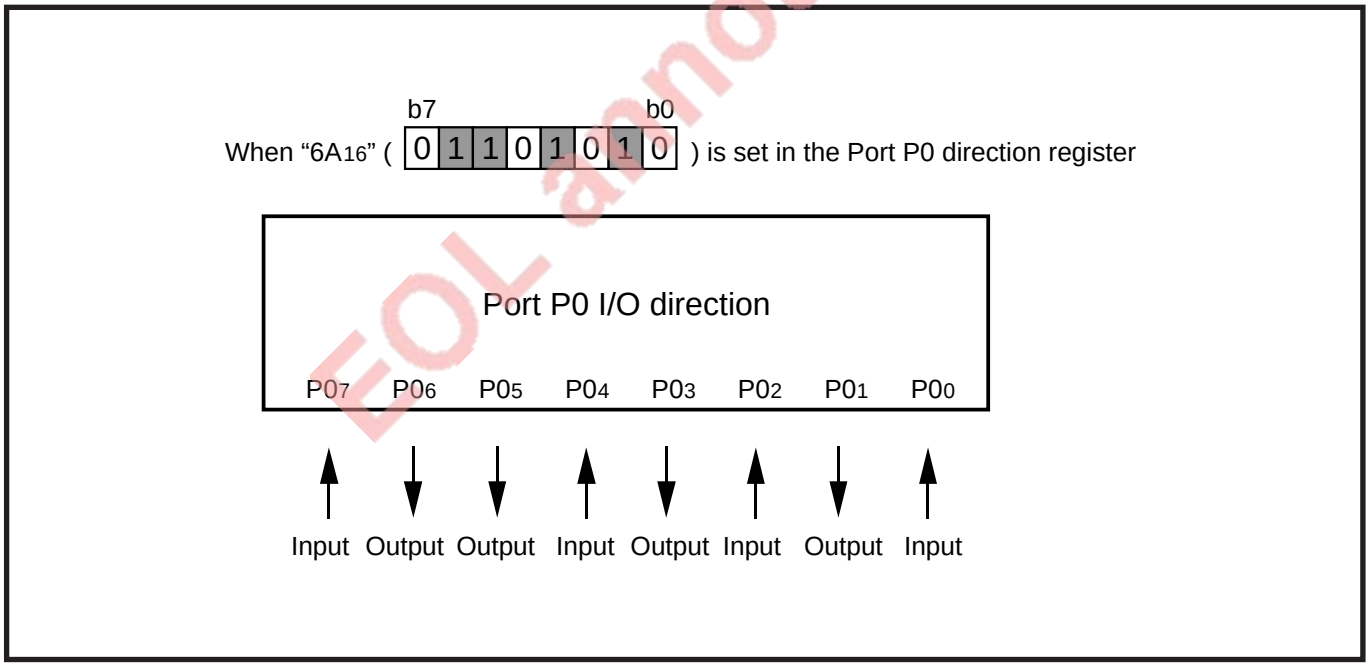


Fig. 2.1.1 Example of port direction register setting

2.1 I/O pins

(3) Pull-up control register

7470	7471	7477	7478
○	○	○	○

The ports shown in Table 2.1.3 can be pulled up by software. A pull-up operation can be performed by the P0 pull-up control register (address $00D0_{16}$) and P1-P5 pull-up control register (address $00D1_{16}$).

Table 2.1.3 I/O ports that permit pull-up by software

Register / Device	Port P0 pull-up control register	Port P1-P5 pull-up control register*
	Control by 1-bit unit	Control by 4-bit unit
7470 group	P0	P1, P2, P4
7471 group	P0	P1, P2, P4, P5
7477 group	P0	P1, P4
7478 group	P0	P1, P4, P5

*: In the 7470/7477 group, the P1-P4 pull-up control register is arranged.

Note: In the 7470 group, P2 is 4 bits of b0 - b3 and P4 is 2 bits of b0 and b4.
In the 7477 group, P4 is 2 bits of b0 and b1.
In the 7471/7478 group, P5 is 4 bits of b0 - b3.

2.1.2 Notes on use

7470	7471	7477	7478
○	○	○	○

When using I/O pins, take the following points into consideration.

(1) Double function ports

Table 2.1.4 shows double function ports. For setting, refer to a structure of each register.

Table 2.1.4 Double function port and control register

Pin	Double function port		Control register	
	7470/7471 group	7477/7478 group	7470/7471 group	7477/7478 group
P12	T0		Timer 12 mode register (T12M: Address 00F816)	
P13	T1		Timer 34 mode register (T34M: Address 00F916)	
P14	SIN	RXD	Serial I/O mode register (SM: Address 00DC16)	Serial I/O control register (SIOSTS: Address 00E216)
P15	SOUT	TXD	Serial I/O mode register	Serial I/O control register
P16	CLK	SCLK	Serial I/O mode register	Serial I/O control register
P17	$\overline{\text{SRDY}}$		Serial I/O mode register	Serial I/O control register
P20	IN0		A-D control register (ADCON: Address 00D916)	
P21	IN1		A-D control register	
P22	IN2		A-D control register	
P23	IN3		A-D control register	
P24	IN4		A-D control register	
P25	IN5		A-D control register	
P26	IN6		A-D control register	
P27	IN7		A-D control register	
P30	INT0		Edge polarity selection register (EG: Address 00D416)	
P31	INT1		Edge polarity selection register	
P32	CNTR0		Edge polarity selection register, Timer 12 mode register	
P33	CNTR1		Edge polarity selection register, Timer 34 mode register	
P50	XCIN		CPU mode register (CPUM: Address 00FB16)	
P51	XCOUT		CPU mode register	

Note: In the 7470/7477 group, P2 is 4 bits of b0 to b3. The 7470/7477 group is not provided with P5.

APPLICATION

2.1 I/O pins

(2)Description of Pull-up control

7470	7471	7477	7478
○	○	○	○

When pulling up a port by software, take the following points into consideration.

- When P1 is used in the serial I/O mode, the pull-up settings corresponding to $P1_4$ to $P1_7$ are invalidated. (Pull-up is impossible.)
- Pull-up control is exerted in the following bit units.
  P0 : 1-bit unit
  P1 to P5 : 4-bit unit

When using an external pull-up resistor and software pull-up control for the same port in combined form, use P0, which can be controlled in bit units.

(3)Notes on external circuit design for I/O ports

① When designing an external circuit for I/O ports, be sure to set the following items within the standard value range.

- Sum output current
- Peak output current
- Average output current

Figure 2.1.2 shows the example of external circuit design for I/O port.

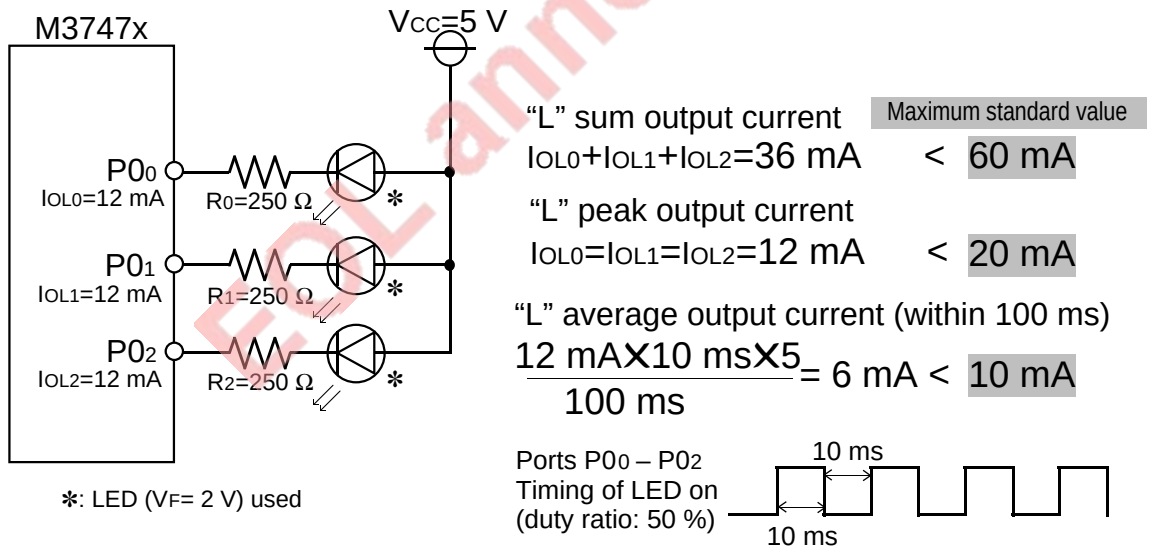


Fig. 2.1.2 Example of external circuit design for I/O port

② When performing multiple key-in operations by forming a key matrix, design in consideration of the port input current for multiple key-in operations.

■ For other notes, refer to **"1.10 I/O Pins."**

2.2 Interrupts

7470	7471	7477	7478
○	○	○	○

2.2.1 Memory allocation

Figure 2.2.1 shows a memory map of interrupt related registers.

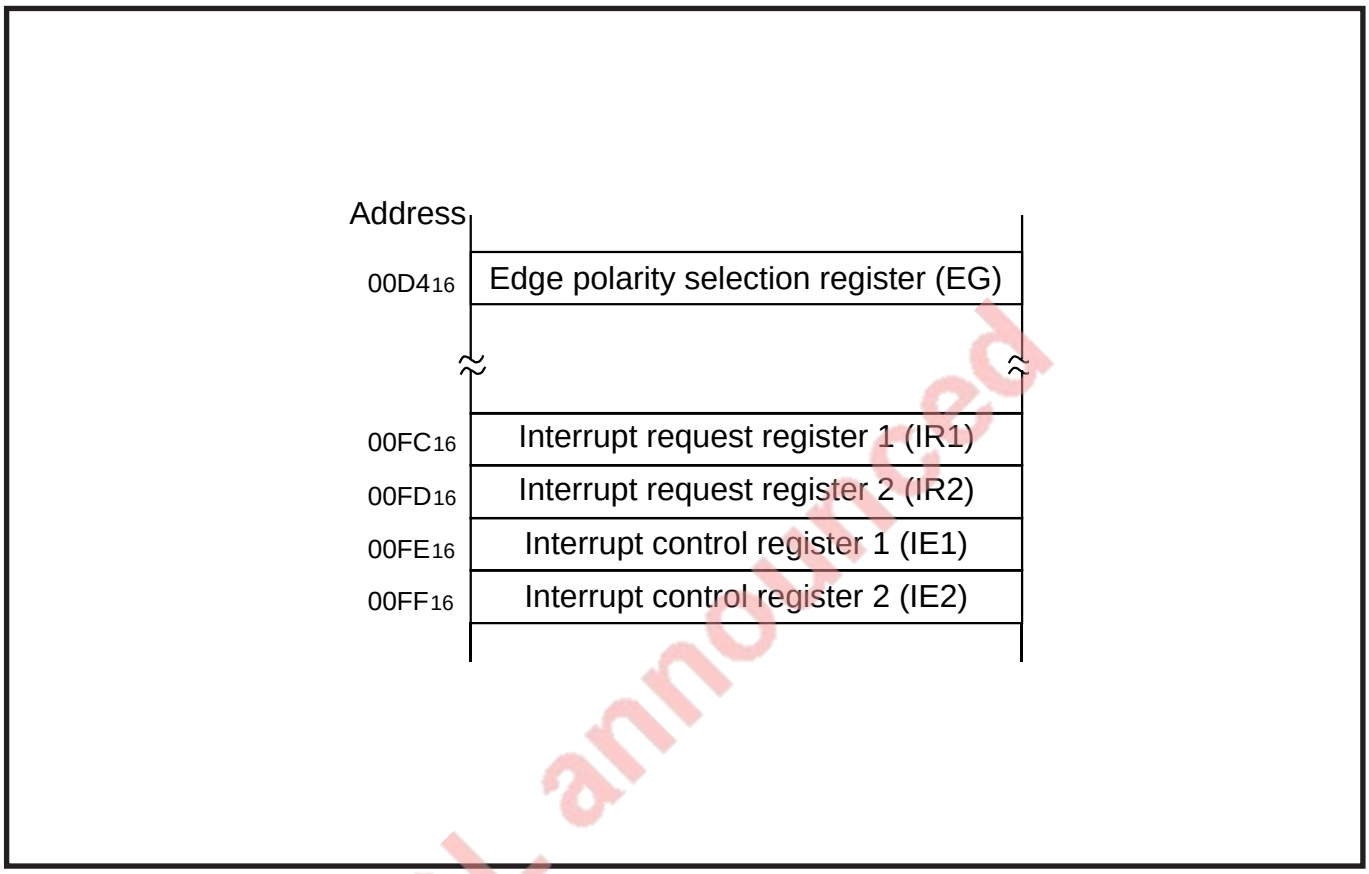


Fig. 2.2.1 Memory map of interrupt related registers

2.2 Interrupts

2.2.2. Processor status register (PS)

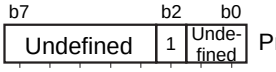
7470	7471	7477	7478
○	○	○	○

The Processor status register consists of 8 bits.
Figure 2.2.2 shows the structure of the Processor status register. Bit 2 related to interrupts is described below.

■ Interrupt disable flag: b2

The interrupt disable flag controls the acceptance of interrupt requests other than the BRK instruction. When this flag is “1,” the acceptance of interrupt requests is disabled. When the flag is “0,” the acceptance of interrupt requests is enabled. The instruction to set this flag to “1” is the SEI instruction and the instruction to set this flag to “0” is the CLI instruction.
At a branch to an interrupt processing routine, this flag is automatically set to “1,” thereby multiple interrupts are disabled. To use multiple interrupt, set this flag to “0” by using the CLI instruction in the interrupt processing routine.

Processor status register



b	Flag name
0	C : Carry flag
1	Z : Zero flag
2	I : Interrupt disable flag
3	D : Decimal mode flag
4	B : Break flag
5	T : Index X mode flag
6	V : Overflow flag
7	N : Negative flag

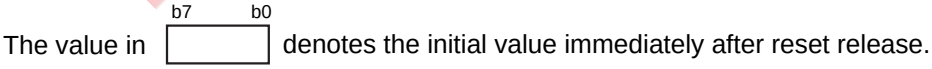


Fig. 2.2.2 Structure of Processor status register

2.2.3 Application example

7470	7471	7477	7478
○	○	○	○

(1)External event detection by CNTR

To detect a rising edge or a falling edge of the level of an input pin by using a pin other than the INT_0 pin and the INT_1 pin, it is possible to use the CNTR pin. Examples of use are shown below.

● When the $CNTR_0$ pin is used

<Interrupt source> CNTR interrupt*1
<Setting>
① Set the edge polarity selection register.
• Select a $CNTR_0$ edge polarity.
• Select $CNTR_0$ as an interrupt source.
② Clear the CNTR interrupt request bit to "0."
③ Execute the NOP instruction.
④ Set the CNTR interrupt enable bit to "1."

● When the $CNTR_1$ pin is used

<Interrupt source> Timer 3 interrupt*1,*2
<Setting>
① Stop the count operation of timer 3.
② Select $CNTR_1$ as a count source of timer 3.
③ Select a $CNTR_1$ edge polarity by the Edge polarity selection register.
④ Set timer 3 to "0."
⑤ Clear the timer 3 interrupt request bit to "0."
⑥ Set the timer 3 interrupt enable bit to "1."
⑦ Start the count operation of timer 3.

*1: It is possible to use the $CNTR_0$ pin for a timer interrupt and the $CNTR_1$ pin for a CNTR interrupt.
*2: It is possible to use timer 4 as an interrupt source.

2.2.4 Notes on use

■ For notes on use, refer to **"1.11 Interrupts."**

2.3 Timers

7470	7471	7477	7478
○	○	○	○

2.3.1 Memory allocation

Figure 2.3.1 shows a memory map of timer related registers.

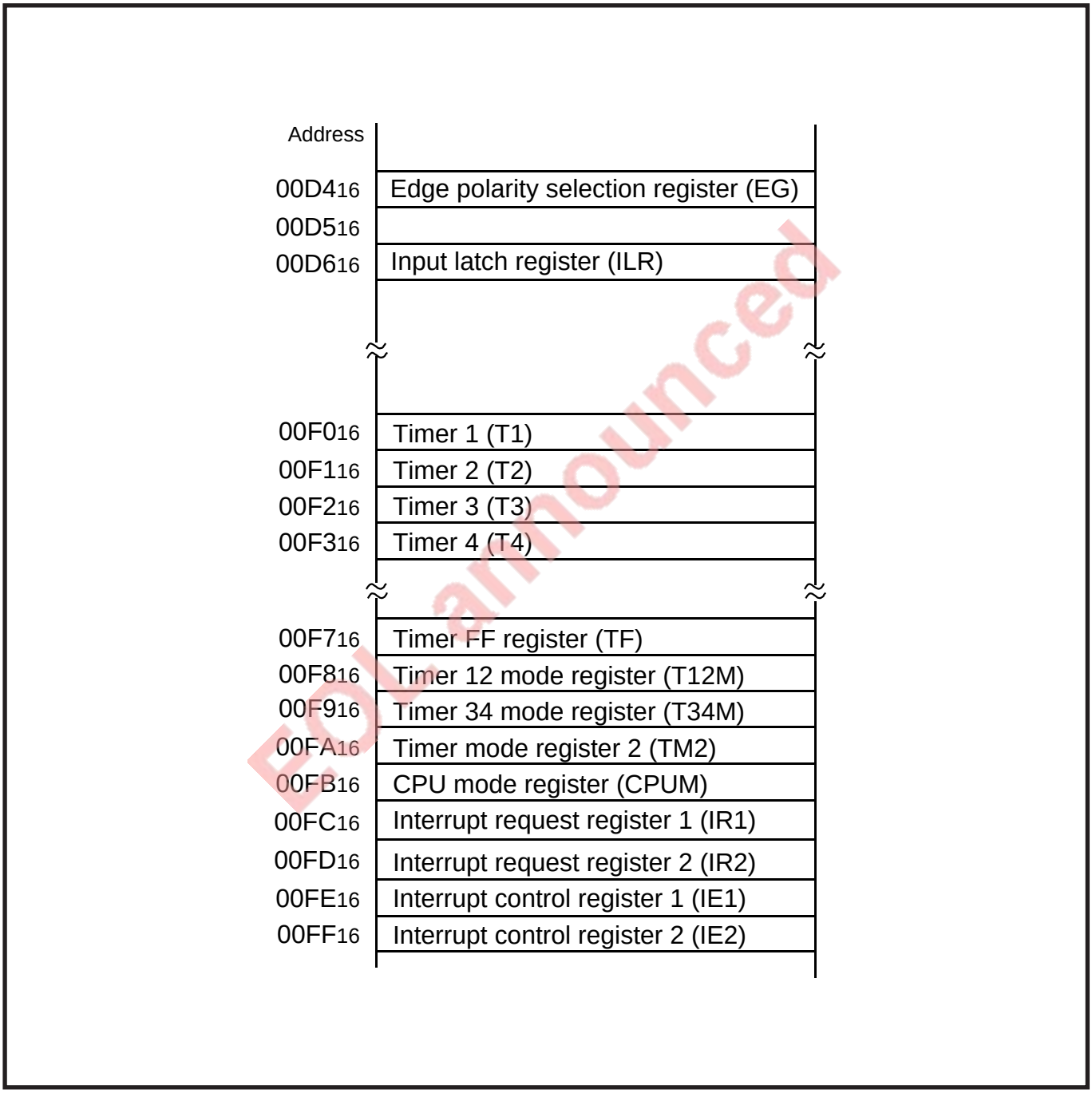


Fig. 2.3.1 Memory map of timer related registers

2.3.2 Application example

7470	7471	7477	7478
○	○	○	○

(1) Each mode of timer

For timers 1, 2, 3 and 4, the following 5 operation modes are available. For each timer mode and the details of it, refer to **"1.12 Timers."**

① Timer mode
② Event counter mode
③ Pulse output mode
④ External pulse width measurement mode
⑤ PWM mode

Each timer mode has relation to the T_0, T_1, $CNTR_0$ and $CNTR_1$ pins as shown in Table 2.3.1. There are some modes that cannot be used for some combinations of timer and pin. Consider this when designing timers.

Table 2.3.1 Relation between timer-used pins and modes

Pin / Timer	T_0	T_1	$CNTR_0$	$CNTR_1$
Timer 1	Pulse output mode	×	Event counter mode	×
Timer 2	×	×	×	×
Timer 3	×	PWM mode	×	Event counter mode
Timer 4	×	Pulse output mode / PWM mode	External pulse width measurement mode	Event counter mode / External pulse width measurement mode

(2) Example of use of each mode

7470	7471	7477	7478
×	○	×	○

An example of use of each mode is shown below.

① Timer mode: One-second measurement (timer function)

Outline: Divide the clock by the timer. Count one second by a timer 1 interrupt that is generated at an internal of 0.4 ms. Cause the timer to count up at each second.

Specifications: Divide f(XCIN) = 32 kHz by timer 1 to generate an interrupt. Check the value of the counter that counts with an timer 1 interrupt by the main routine. If one second has elapsed, execute timer count-up processing.

Figure 2.3.2 shows an example of control procedure.

7470	7471	7477	7478
×	○	×	○

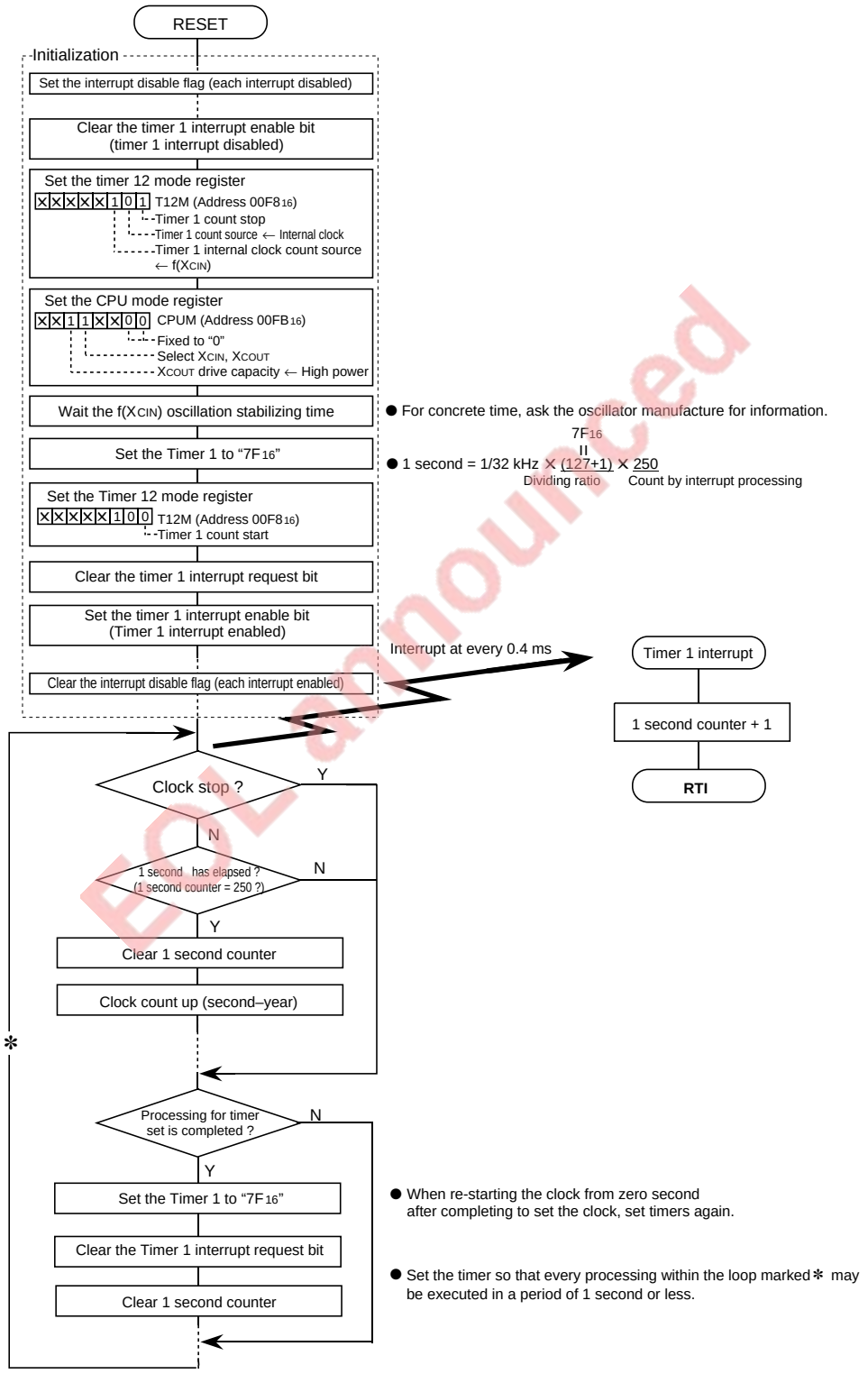


Fig. 2.3.2 Example of control procedure [Clock function]

2.3 Timers

② Event counter mode: Frequency measurement

7470	7471	7477	7478
○	○	○	○

Outline: The frequency of the pulse input to the CNTR0 pin ("H" active) is measured by the number of events in a certain period.

Specifications: A count operation is started specifying the count source of timer 1 as CNTR0. Timer 2 (count source: f(XIN)/64) detects 1 ms and the frequency of the pulse input to CNTR0 is calculated from the number of events counted within 1 ms.

Note: The number of events of an input pulse is specified as 255 or less within 1 ms.

Figure 2.3.3 shows an example of measurement method of frequency and Figure 2.3.4 shows an example of control procedure.

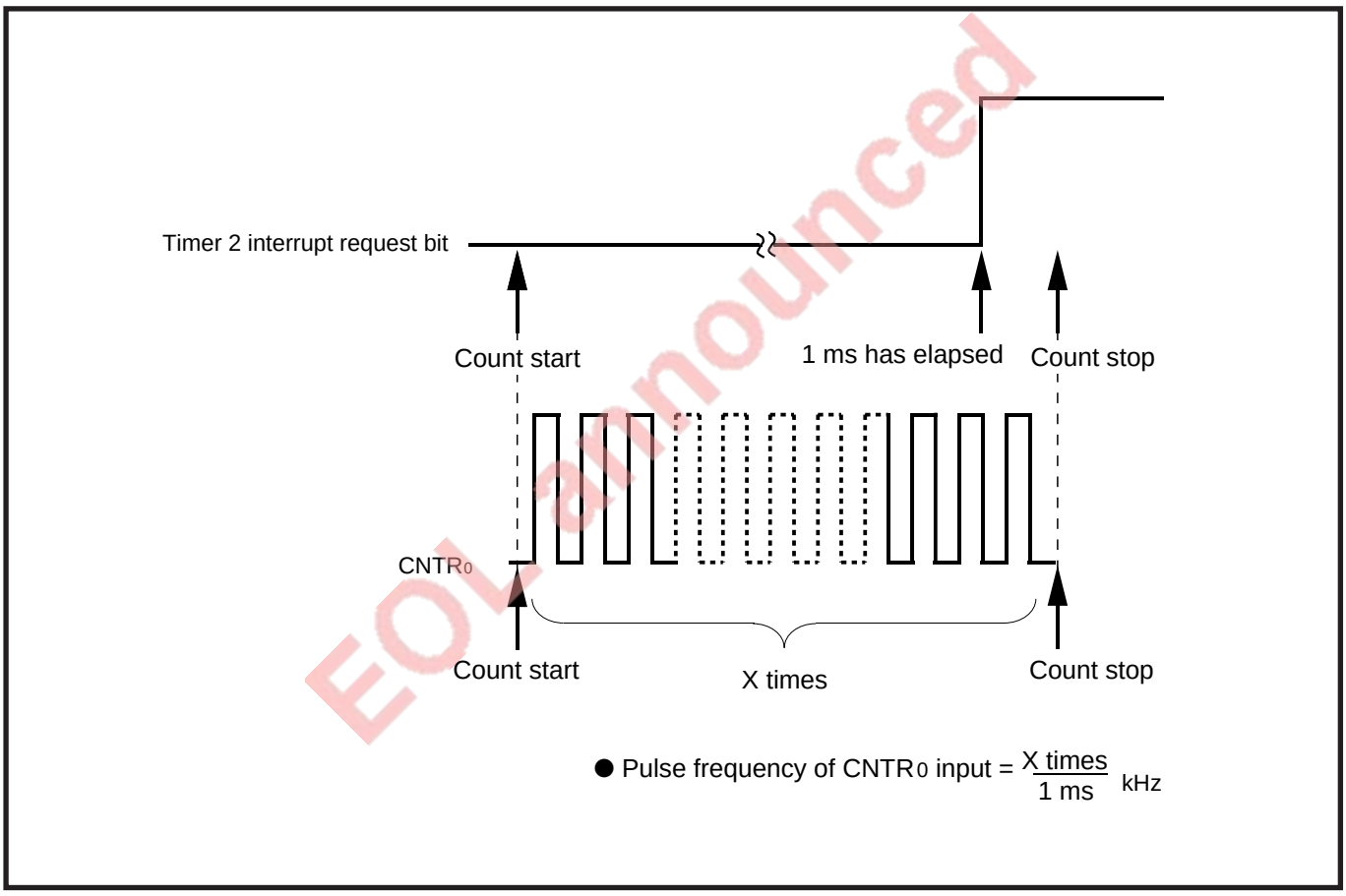


Fig. 2.3.3 Example of measurement method of frequency

7470	7471	7477	7478
○	○	○	○

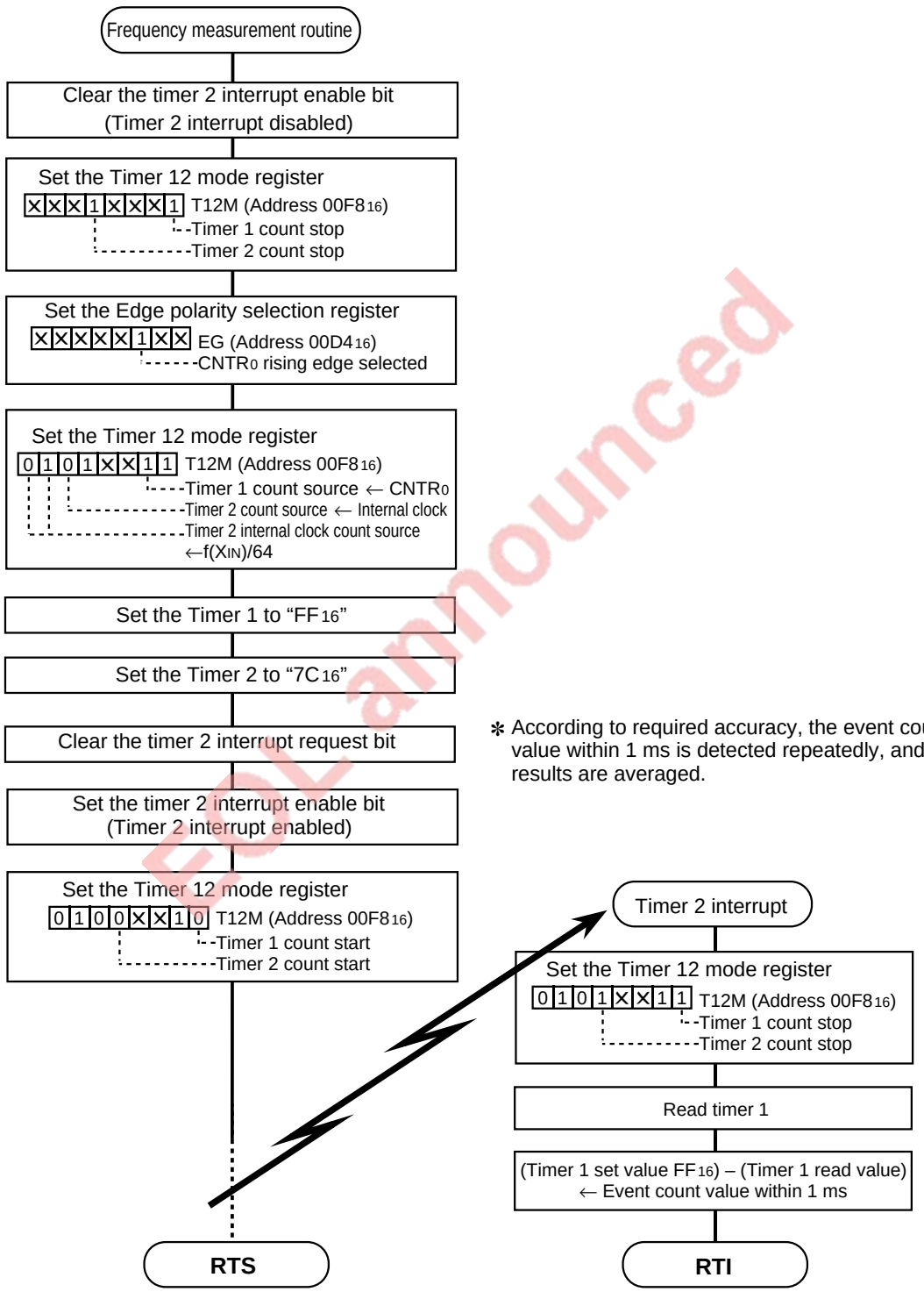


Fig. 2.3.4 Example of control procedure [Frequency measurement]

2.3 Timers

③ Pulse output mode: Piezoelectric buzzer output

7470	7471	7477	7478
○	○	○	○

Outline: The pulse output function of the timer is applied for a piezoelectric buzzer output.
Specifications: A square wave obtained by dividing the clock f(XIN) = 8 MHz into about 2 kHz is output from the T0 pin. While the buzzer output stops, the level of the T0 pin is fixed at "H."

Figure 2.3.5 shows an example of a peripheral circuit. Figure 2.3.6 shows a connection of the timer and setting of the division ratio. Figure 2.3.7 shows an example of control procedure.

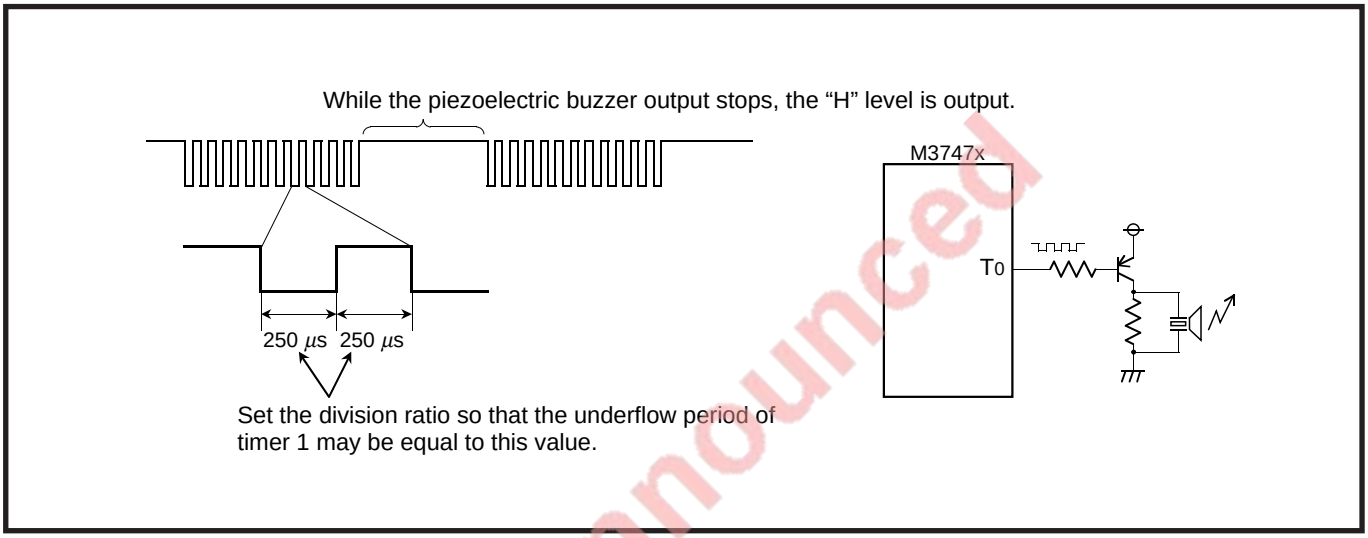


Fig. 2.3.5 Example of a peripheral circuit [Pulse output mode]

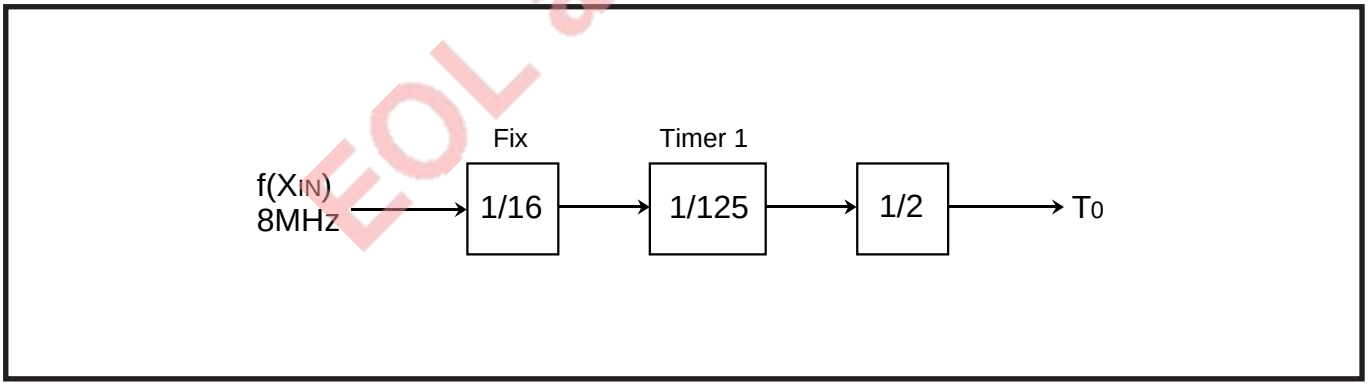


Fig. 2.3.6 Connection of timer and setting of division ratio

7470	7471	7477	7478
○	○	○	○

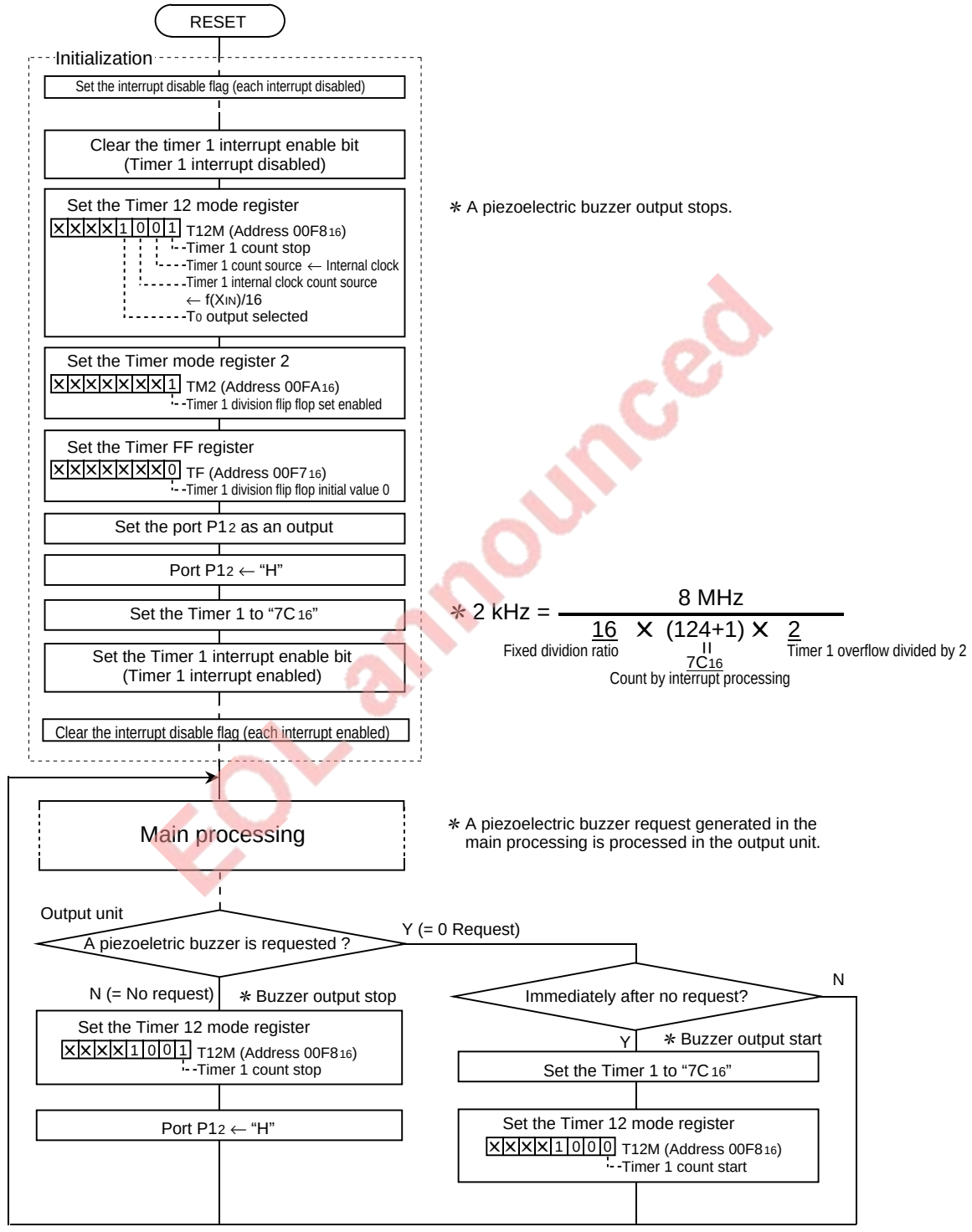


Fig. 2.3.7 Example of control procedure [Piezoelectric buzzer output]

2.3 Timers

④ Pulse width measurement mode: Feedback control of phase control signal

7470	7471	7477	7478
○	○	○	○

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

Specifications: The M3747x controls a load by phase control. At this time, the width of the pulse output from the load as a feedback signal is measured. With this result, the control over the load is compensated.

Figure 2.3.8 shows an example of peripheral circuit and Figure 2.3.9 shows an example of control procedure.

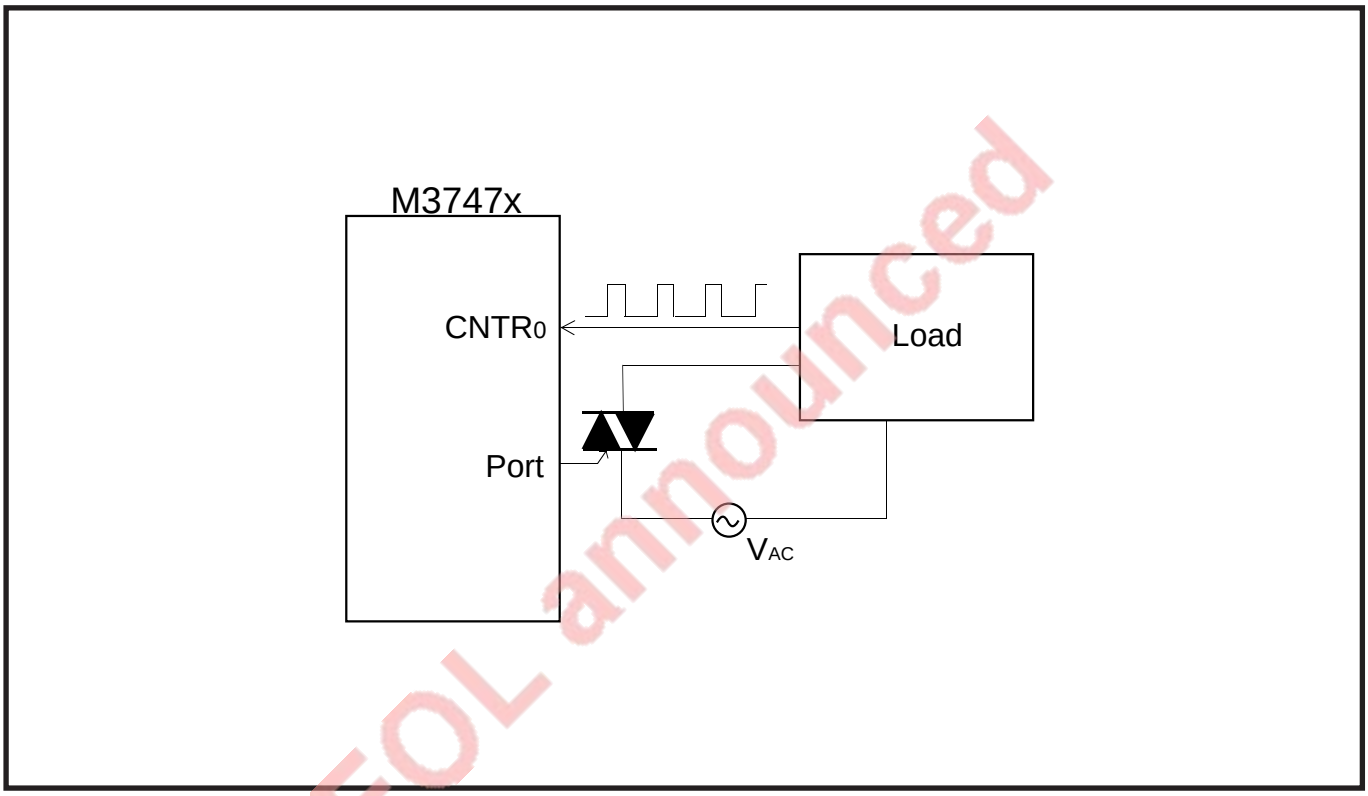


Fig. 2.3.8 Example of peripheral circuit [Pulse width measurement mode]

7470	7471	7477	7478
○	○	○	○

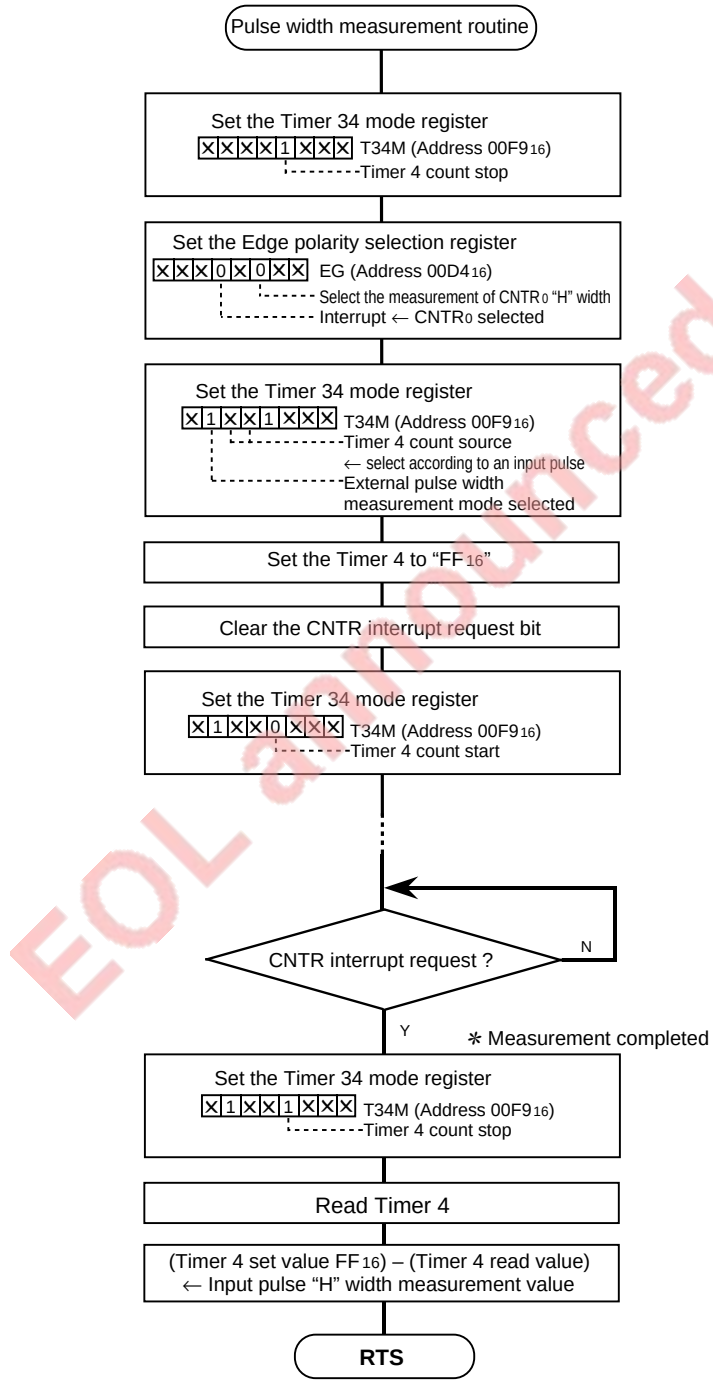


Fig. 2.3.9 Example of control procedure [Pulse width measurement mode]

2.3 Timers

⑤ PWM mode: Analog output

7470	7471	7477	7478
○	○	○	○

Outline: An analog output is performed by using the PWM function of the timer.
Specifications: A count source of Timer 3 and Timer 4 is selected and a PWM waveform is output from the T_1 pin. The PWM waveform is converted into an analog voltage by the external circuit of the T_1 pin, and then this voltage is output.

Note: The analog voltage to be output varies depending on the duty of the PWM waveform.

Figure 2.3.10 shows an example of peripheral circuit and Figure 2.3.11 shows an example of control procedure.

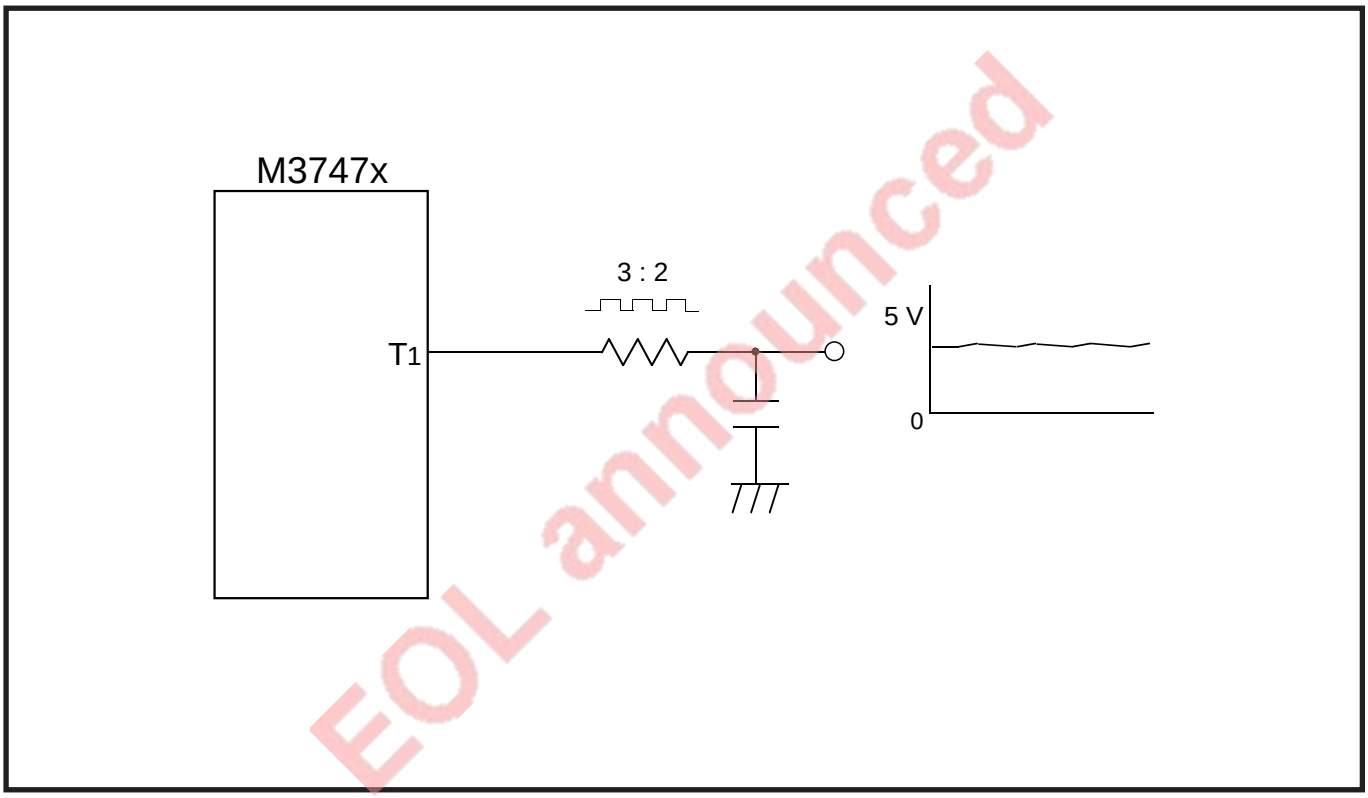


Fig. 2.3.10 Example of peripheral circuit [PWM mode]

7470	7471	7477	7478
○	○	○	○

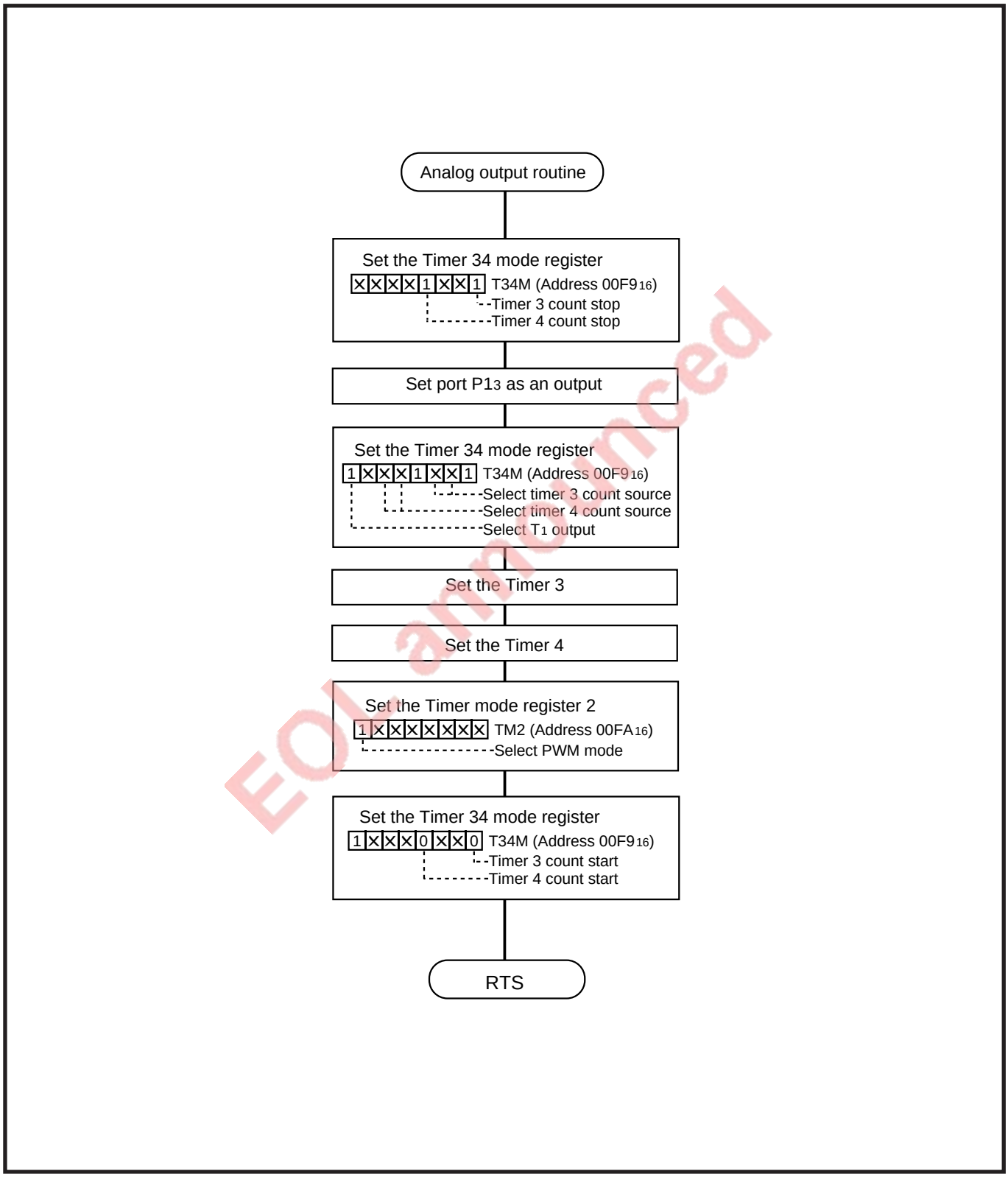


Fig. 2.3.11 Example of control procedure [PWM mode]

2.3.3 Notes on use

7470	7471	7477	7478
○	○	○	○

■ When using a 16-bit counter by using two timers, take the following points into consideration according to **"1.12.5 Notes on use (2)."**

●The timing at which the timer value and the read value change, when two 8-bit timers are connected in series, is shown in Figure 2.3.12, taking the case where timer 1 and timer 2 are connected as an example (When Timer 1 and Timer 2 are connected and the set value of the Timer 1 is 2_{16} and the set value of the Timer 2 is 1_{16}).
The count source of timer 2 is the overflow signal of timer 1. In this case, the read value of timer 2 changes at the fall of the count source. When Timer 1 and Timer 2 are read continuously as a 16-bit counter, the count source of timer 2 changes at the falling edge of the count source of timer 1, so the A section can not be distinguished from the B section. Likewise, the C section cannot be distinguished from the D section.

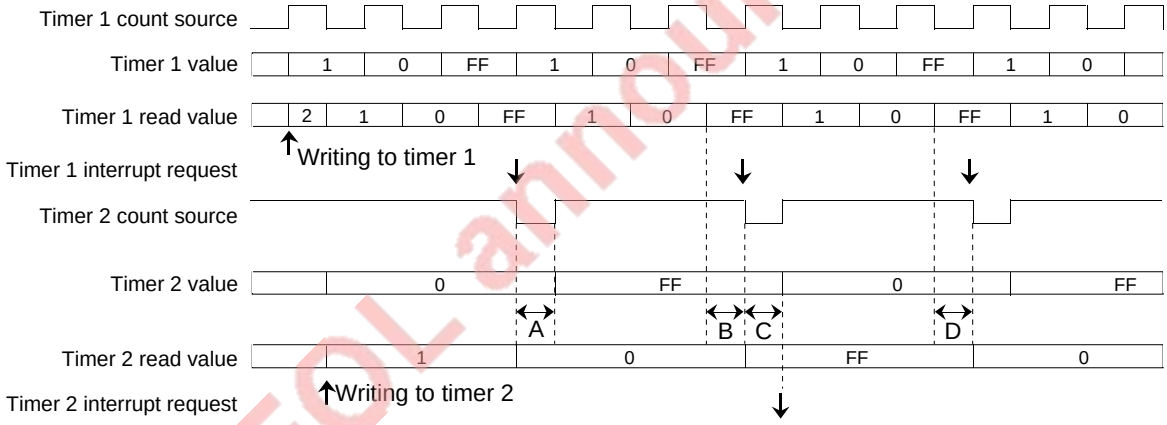


Fig. 2.3.12 Timing at which timer value and read value change in the case where two timers are connected in series

■ For other notes on use, refer to **"1.12 timers."**

2.4 Serial I/O

7470	7471	7477	7478
○	○	×	×

2.4.1 7470/7471 group memory allocation

Figure 2.4.1 shows a memory map of serial I/O related registers in the 7470/7471 group.

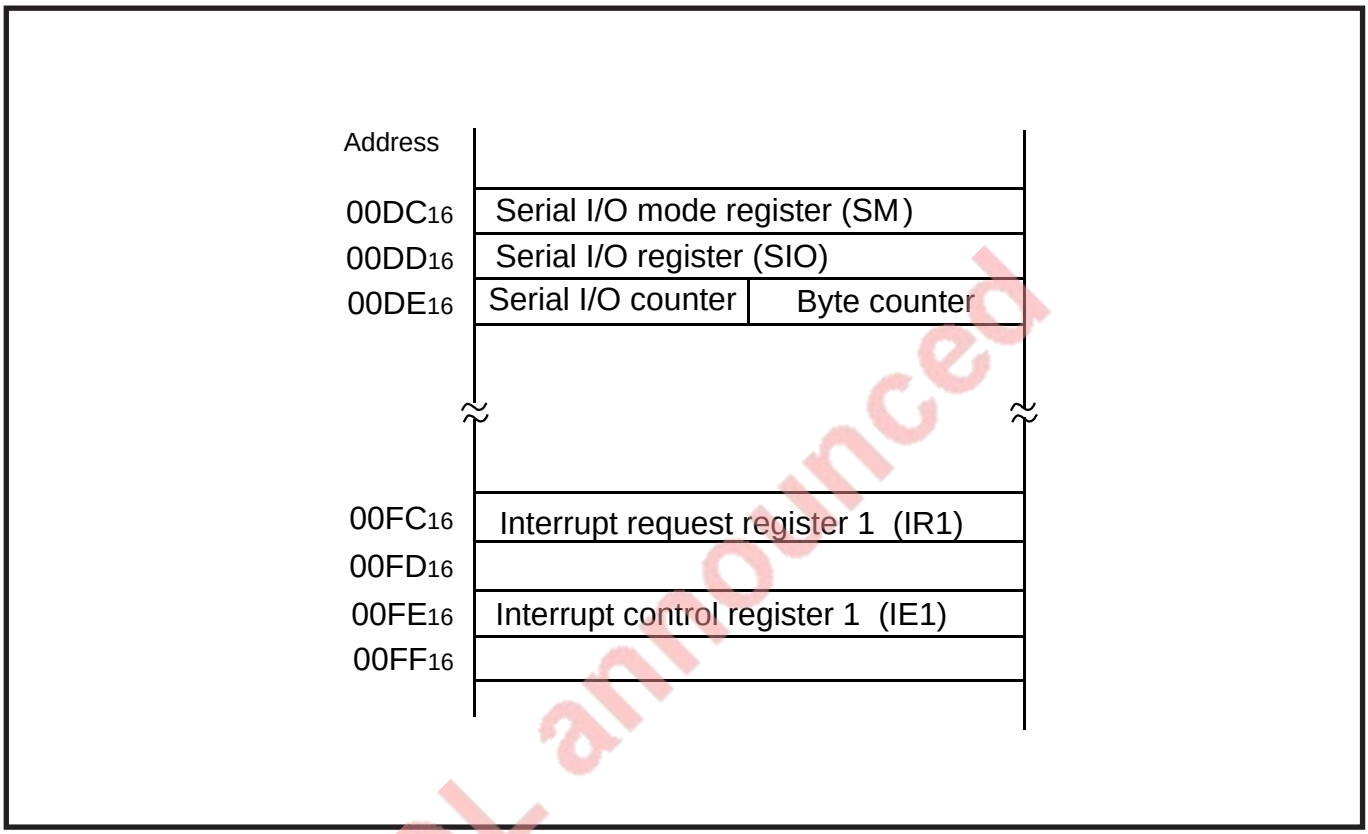


Fig. 2.4.1 Memory map of serial I/O related registers in 7470/7471 group

2.4.2 Application example

7470	7471	7477	7478
○	○	×	×

(1) Clock synchronous serial I/O mode

Outline: Clock synchronous communication is performed among the 7470/7471 group.
Specifications: M3747x①……Transmit side in half-duplex communication
- Synchronous clock: f(XIN)/16
- Port P17 is used as an $\overline{\text{SRDY}}$ signal input pin.

M3747x②……Receive side in half-duplex communication
- Synchronous clock: External clock
- $\overline{\text{SRDY}}$ signal output

Figure 2.4.2 shows an example of connections and Figure 2.4.3 shows an example of control procedure.

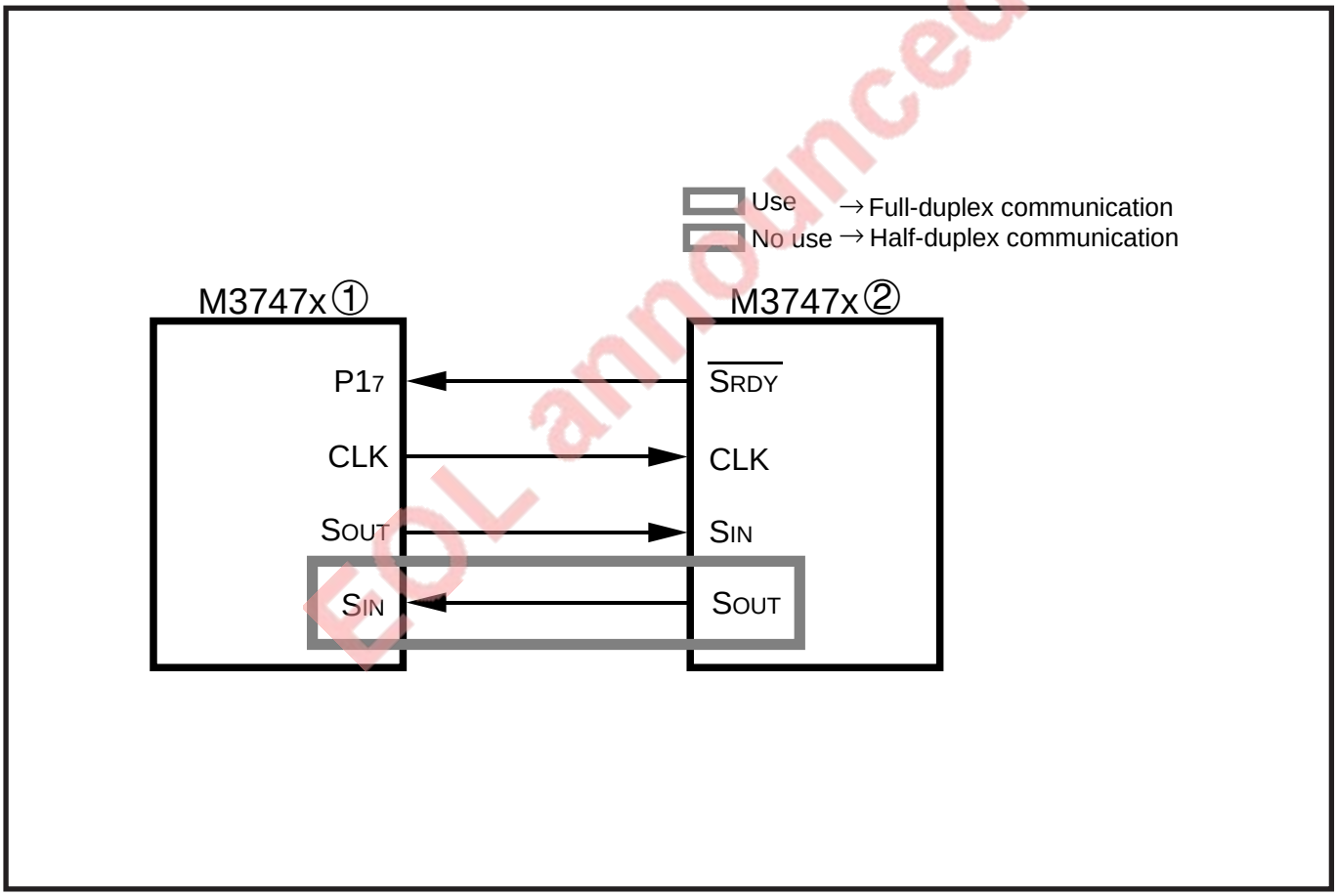


Fig. 2.4.2 Example of connections [Clock synchronous serial I/O mode, 7470/7471 group]

7470	7471	7477	7478
○	○	×	×

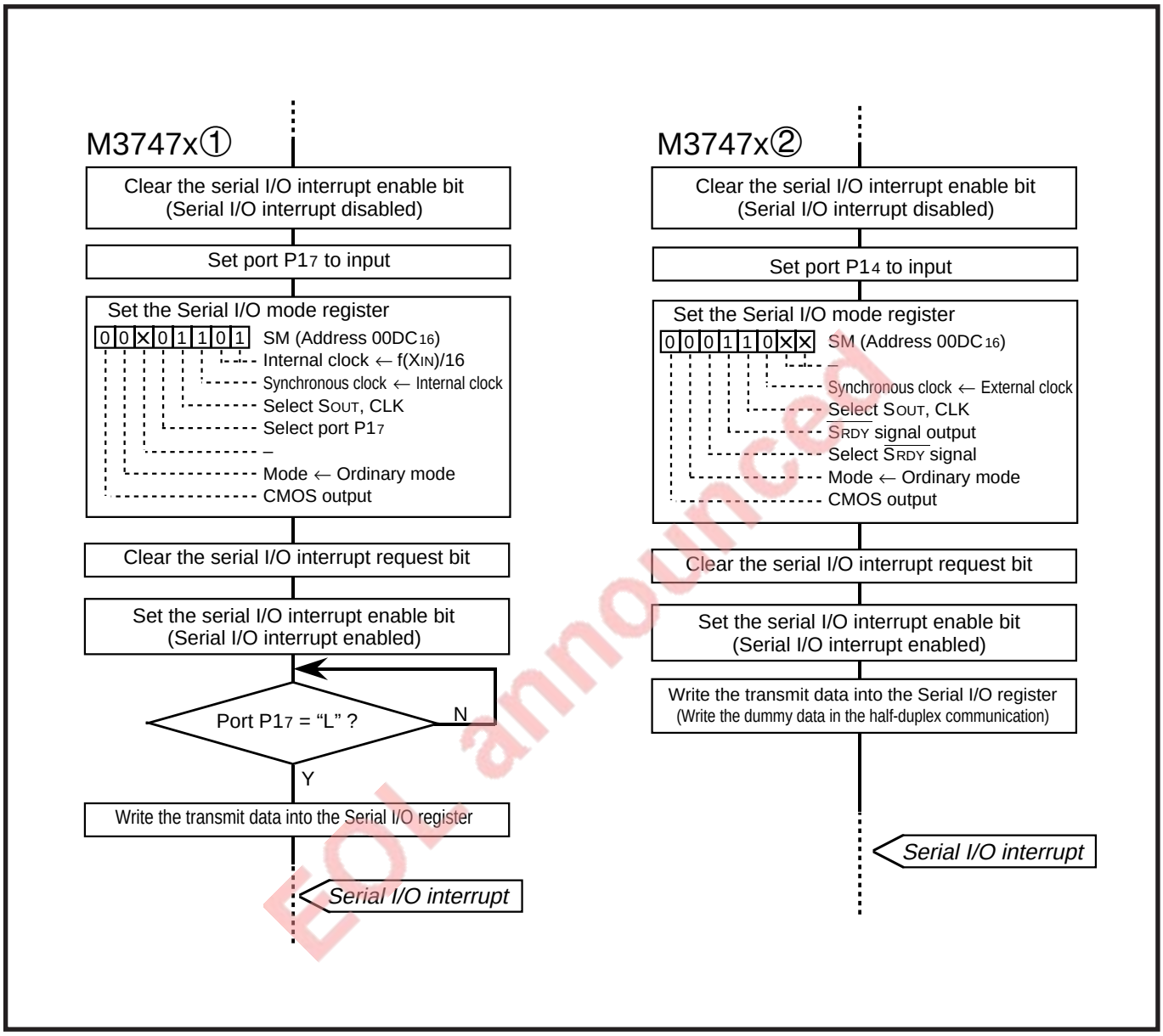


Fig. 2.4.3 Example of control procedure [Clock synchronous serial I/O mode, 7470/7471 group]

2.4 Serial I/O

(2)Byte specification mode

7470	7471	7477	7478
○	○	×	×

Outline: Among the 7470/7471 group, transfer is performed for two or more microcomputers by using the clock synchronous byte specification mode.

Specifications: Transmit side M3747x①
- Synchronous clock: f(XIN)/32
- Port P17 is used as an $\overline{\text{SRDY}}$ signal input pin.
- Port P10 is used as an transmit preparation command signal output pin.

Receive side M3747x② , M3747x③ , M3747x④
- Synchronous clock: External clock
- SARDY signal output
- Port P10 is used as an transmit preparation command signal output pin.

Figure 2.4.4 shows an example of connections and Figure 2.4.5 and 2.4.6 show an example of control procedure.

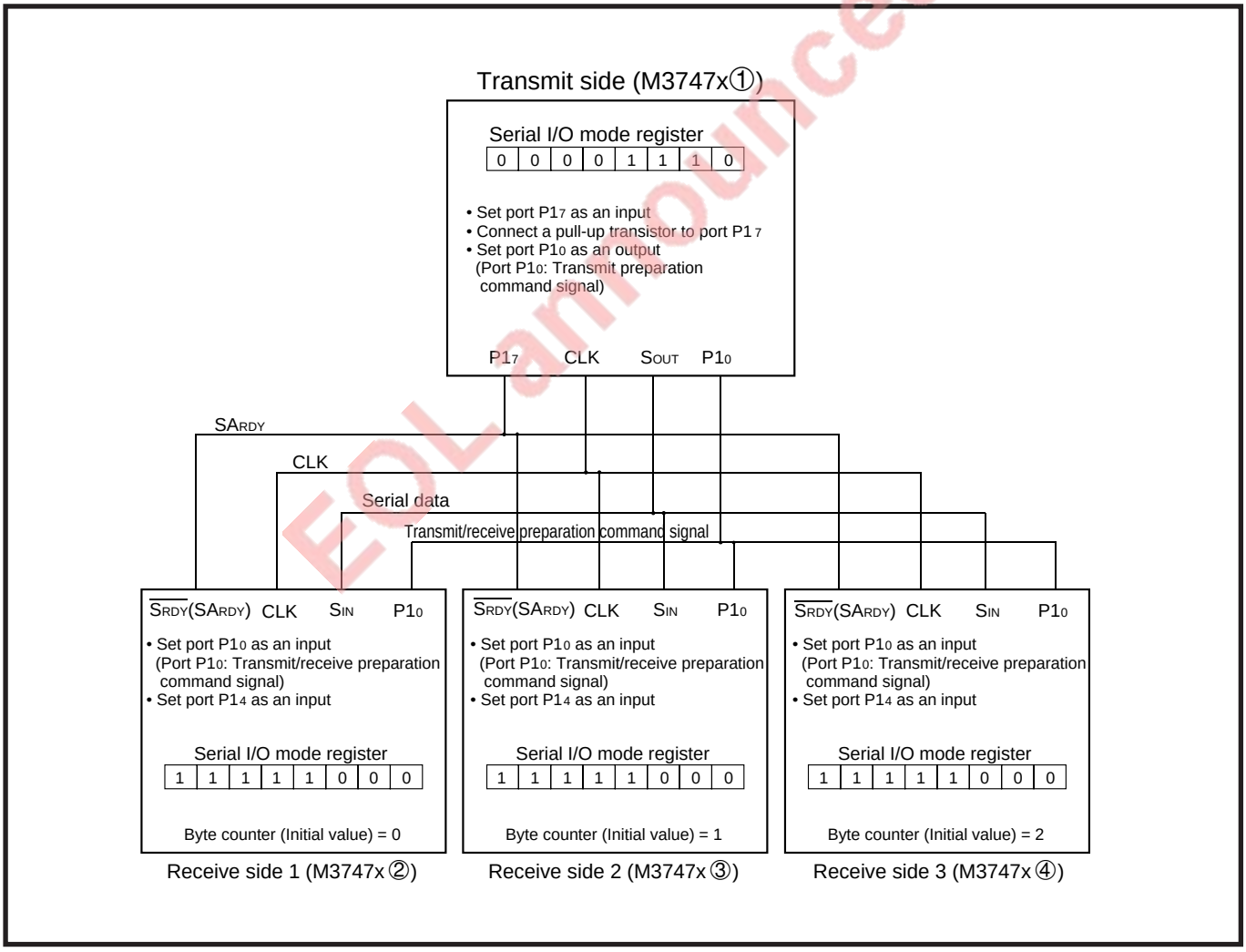


Fig. 2.4.4 Example of connections [Byte specification mode, 7470/7471 group]

7470	7471	7477	7478
○	○	×	×

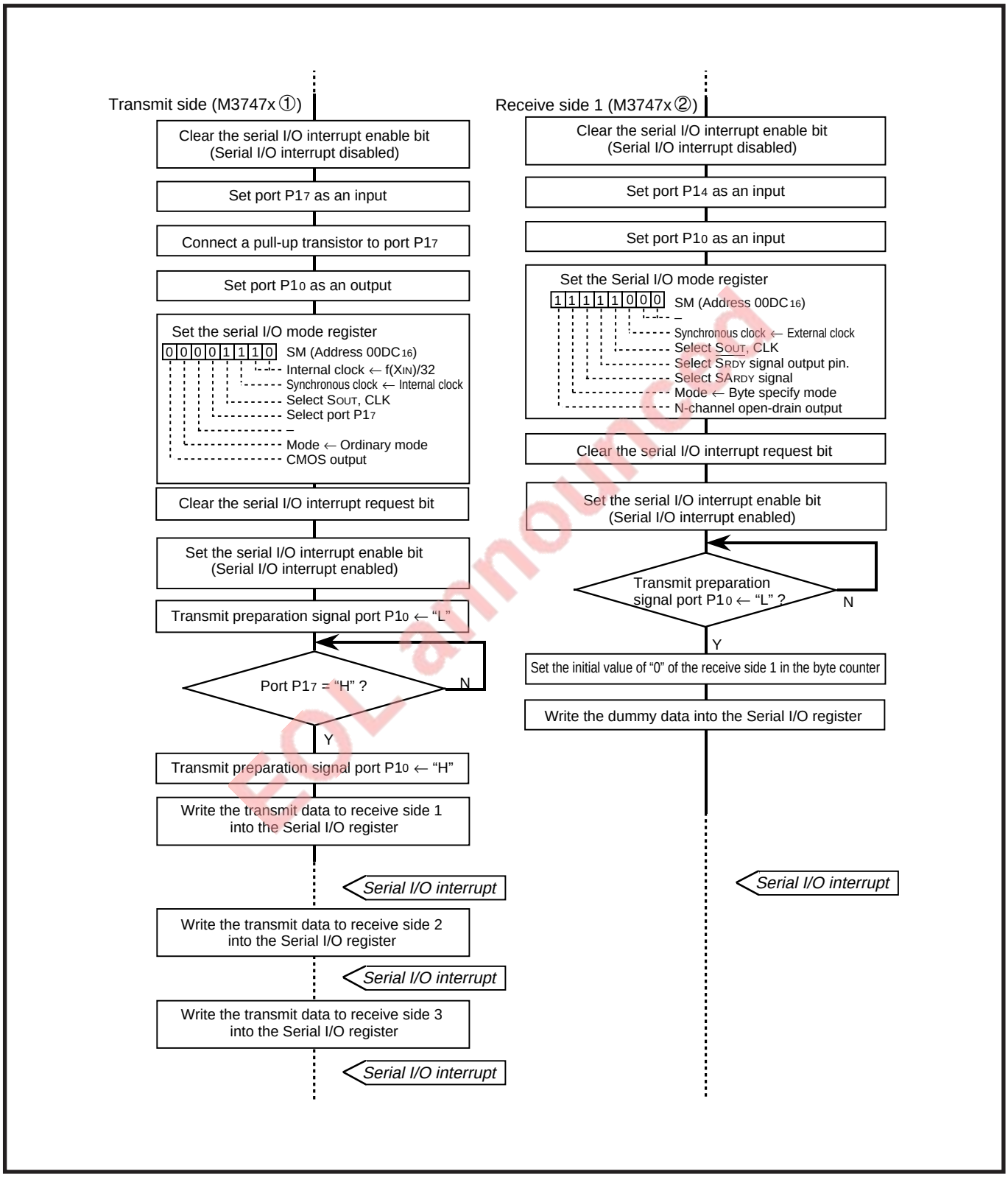


Fig. 2.4.5 Example of control procedure (1) [Byte specification mode, 7470/7471 group]

2.4 Serial I/O

7470	7471	7477	7478
○	○	×	×

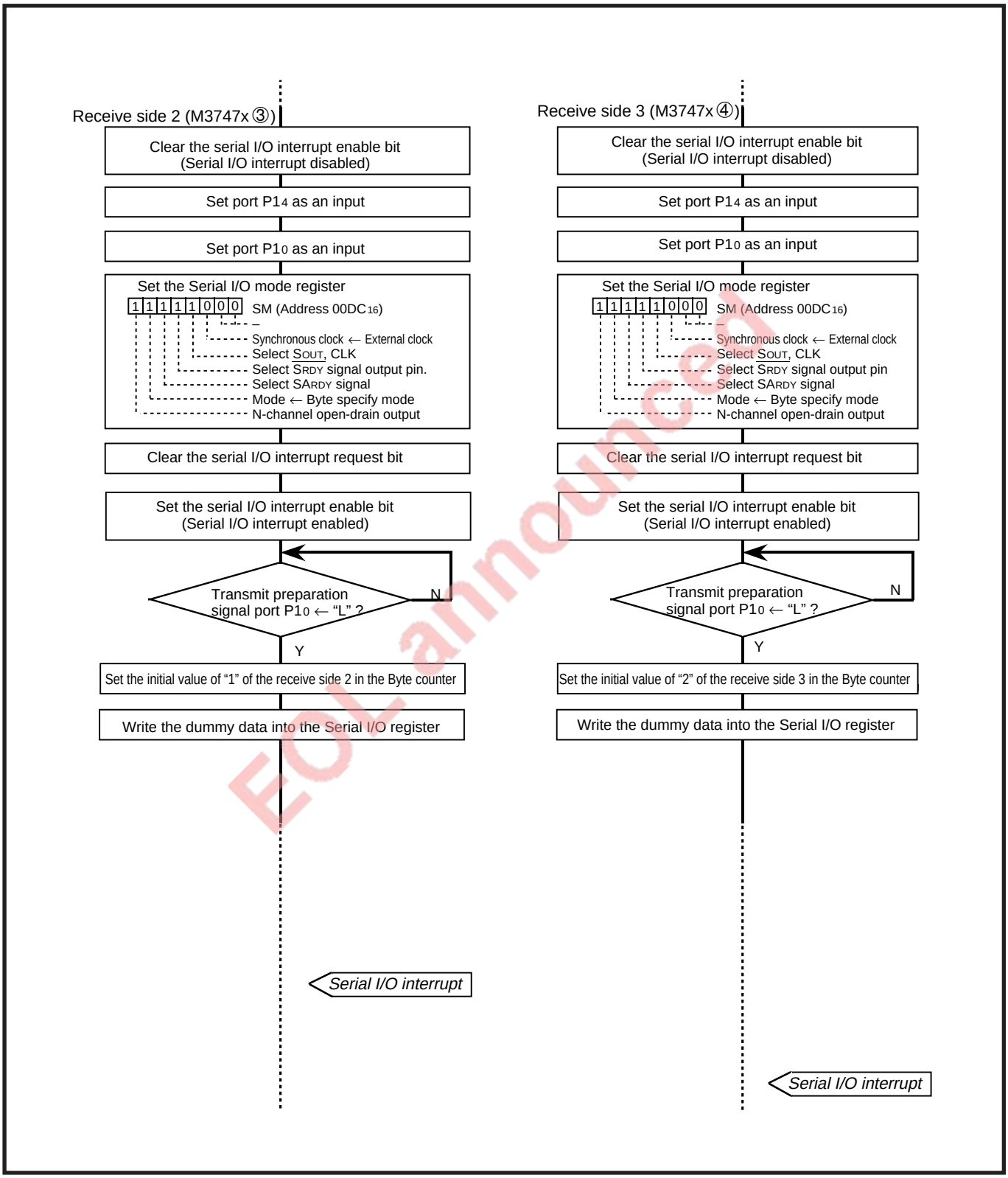


Fig. 2.4.6 Example of control procedure (2) [Byte specification mode, 7470/7471 group]

2.4.3 7477/7478 group memory allocation

7470	7471	7477	7478
×	×	○	○

Figure 2.4.7 shows a memory map of serial I/O related registers in the 7477/7478 group.

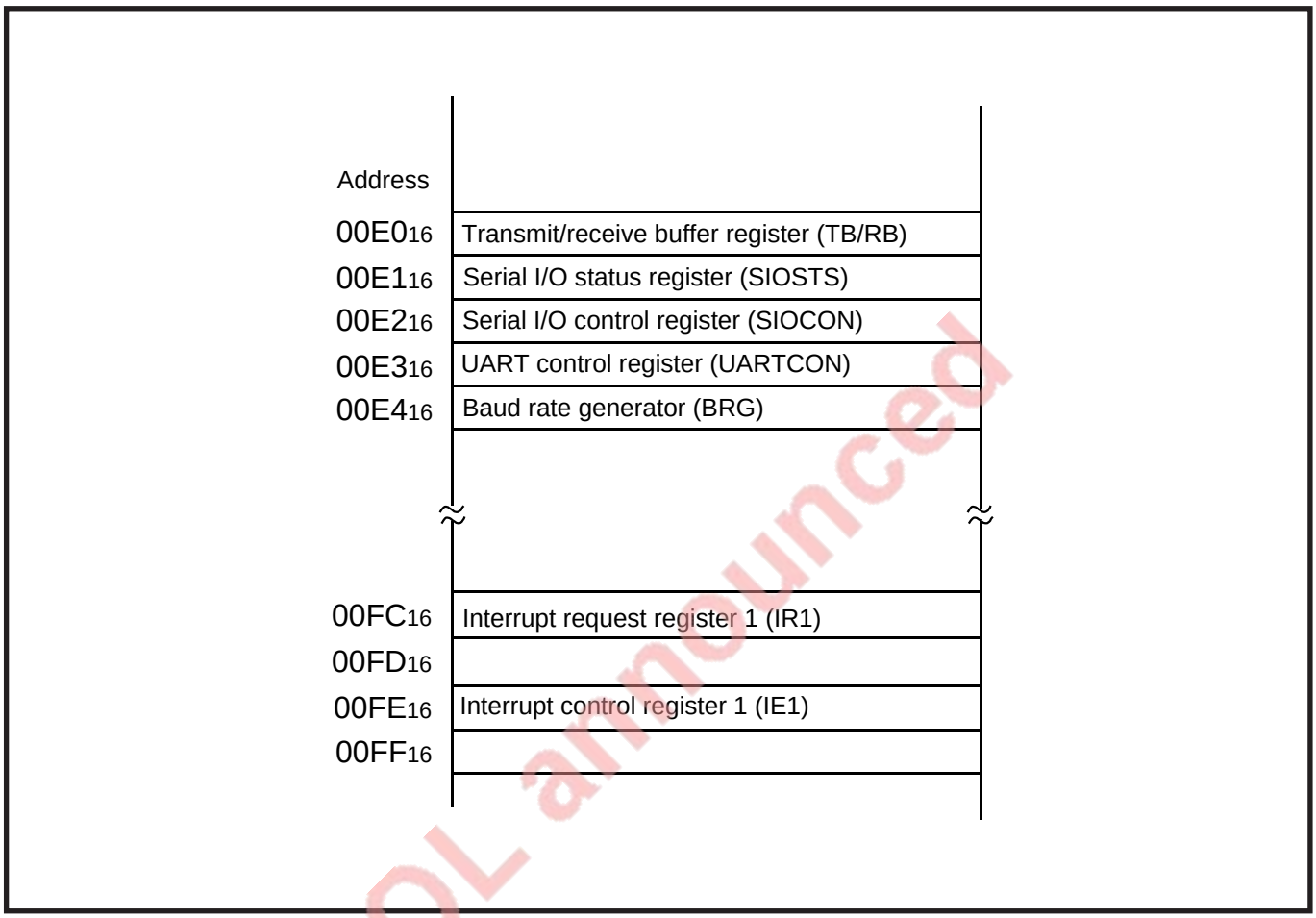


Fig. 2.4.7 Memory map of serial I/O related registers in 7477/7478 group

2.4.4 Application examples

7470	7471	7477	7478
×	×	○	○

(1) Clock synchronous serial I/O mode

Outline: Clock synchronous communication is performed among the 7477/7478 group.
Specifications: M3747x①……Transmit side in half-duplex communication.
- Synchronous clock: BRG output (f(XIN)/4 or f(XIN)/16)/4.
- Port P17 is used as an $\overline{\text{SRDY}}$ signal input pin.

M3747x②……Receive side in half-duplex communication
- Synchronous clock: External clock
- $\overline{\text{SRDY}}$ signal output

Figure 2.4.8 shows an example of connections and Figure 2.4.9 shows an example of control procedure.

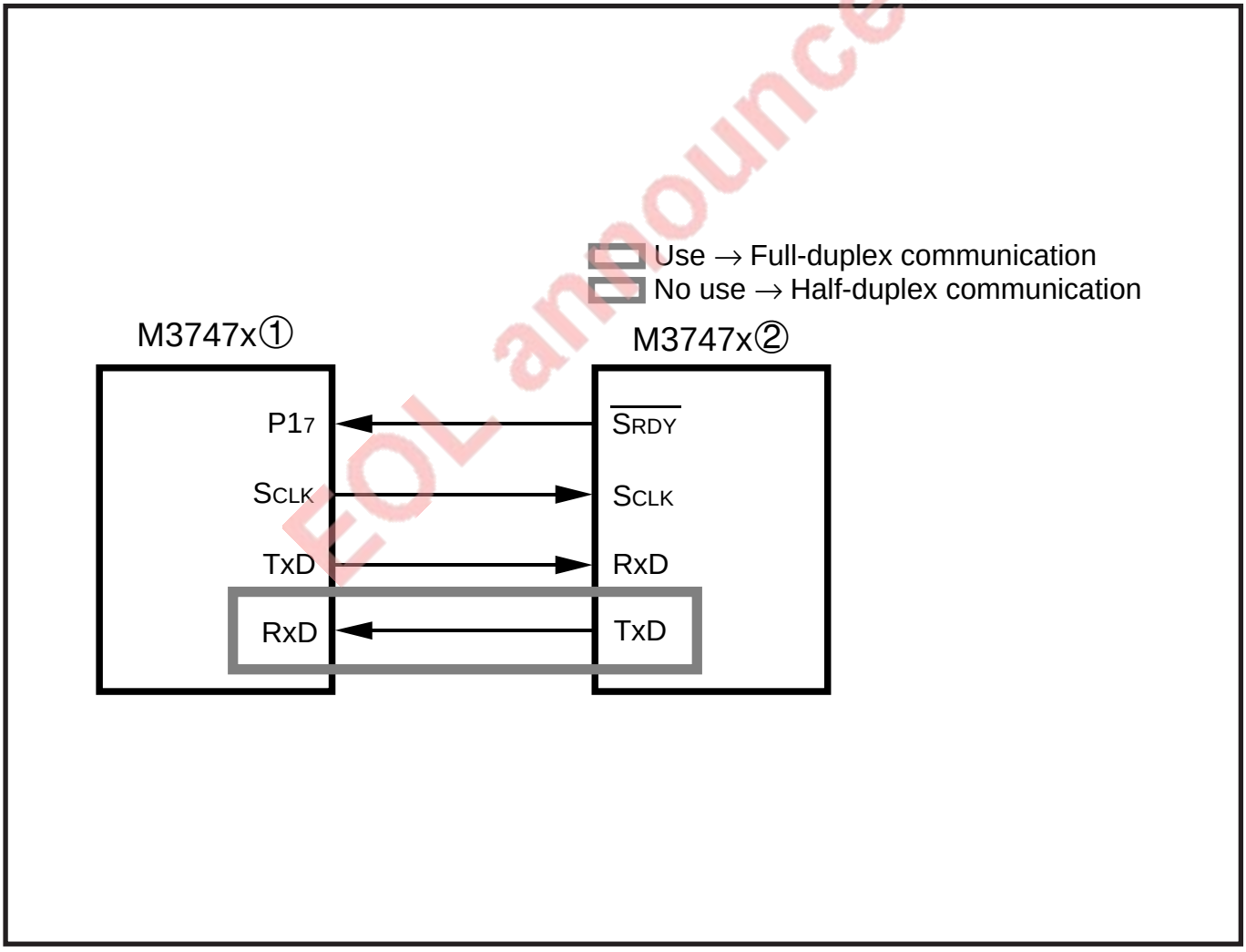


Fig. 2.4.8 Example of connections [Clock synchronous serial I/O mode, 7477/7478 group]

7470	7471	7477	7478
×	×	○	○

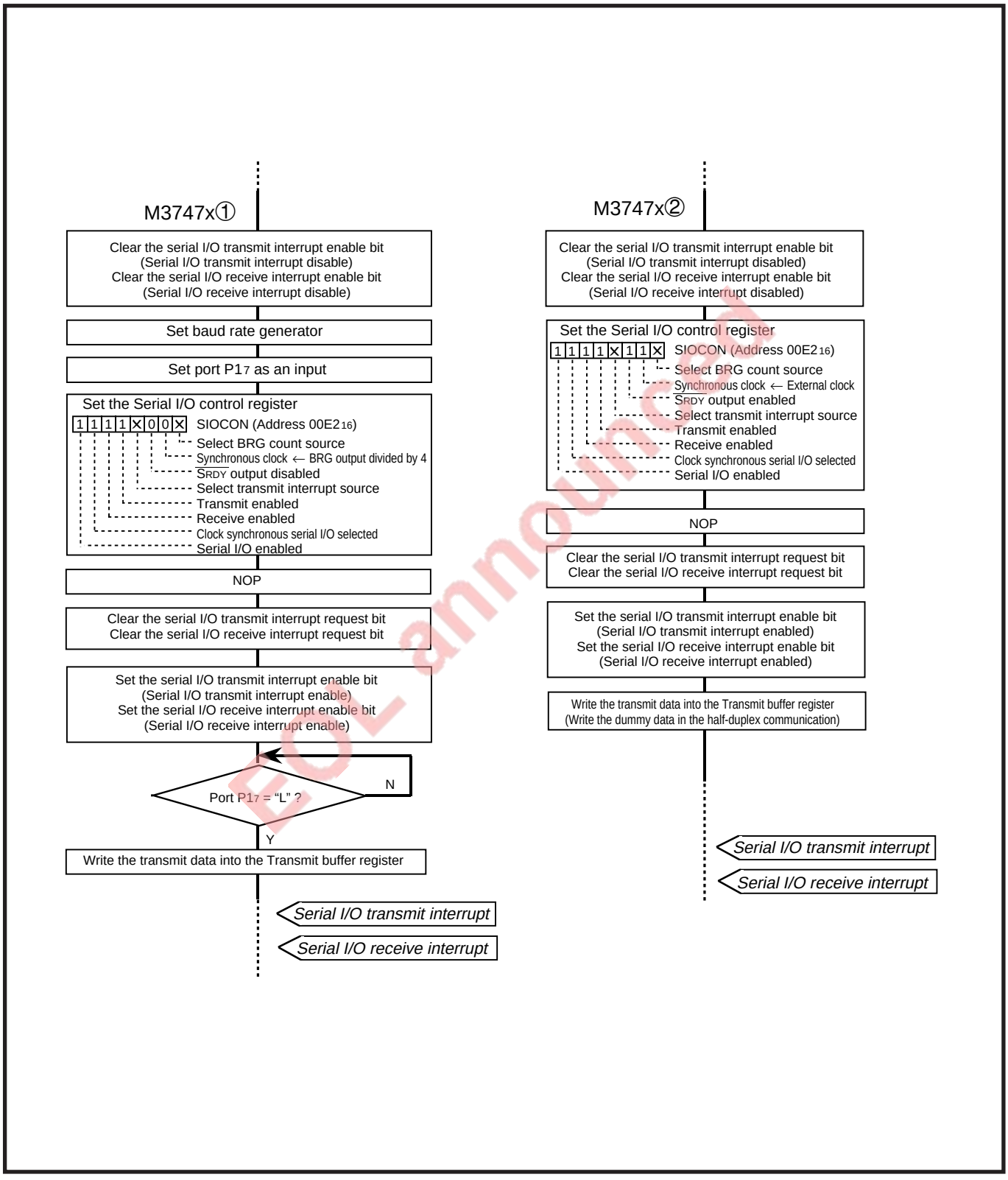


Fig. 2.4.9 Example of control procedure [Clock synchronous serial I/O mode, 7477/7478 group]

(2)Clock asynchronous serial I/O mode

7470	7471	7477	7478
×	×	○	○

Outline: Clock asynchronous communication is performed among the 7477/77478 groups.
Specifications: M3747x①……Transmit side in half-duplex communication

• Baud rate: $\frac{f(X_{IN})}{4 \times (XX_{16} + 1) \times 16}$ bps

M3747x②……Receive side in half-duplex communication

• Baud rate: $\frac{f(X_{IN})}{4 \times (XX_{16} + 1) \times 16}$ bps

"XX_{16}" is a set value of the baud rate generator.

Figure 2.4.10 shows an example of connections and Figure 2.4.11 shows an example of control procedure.

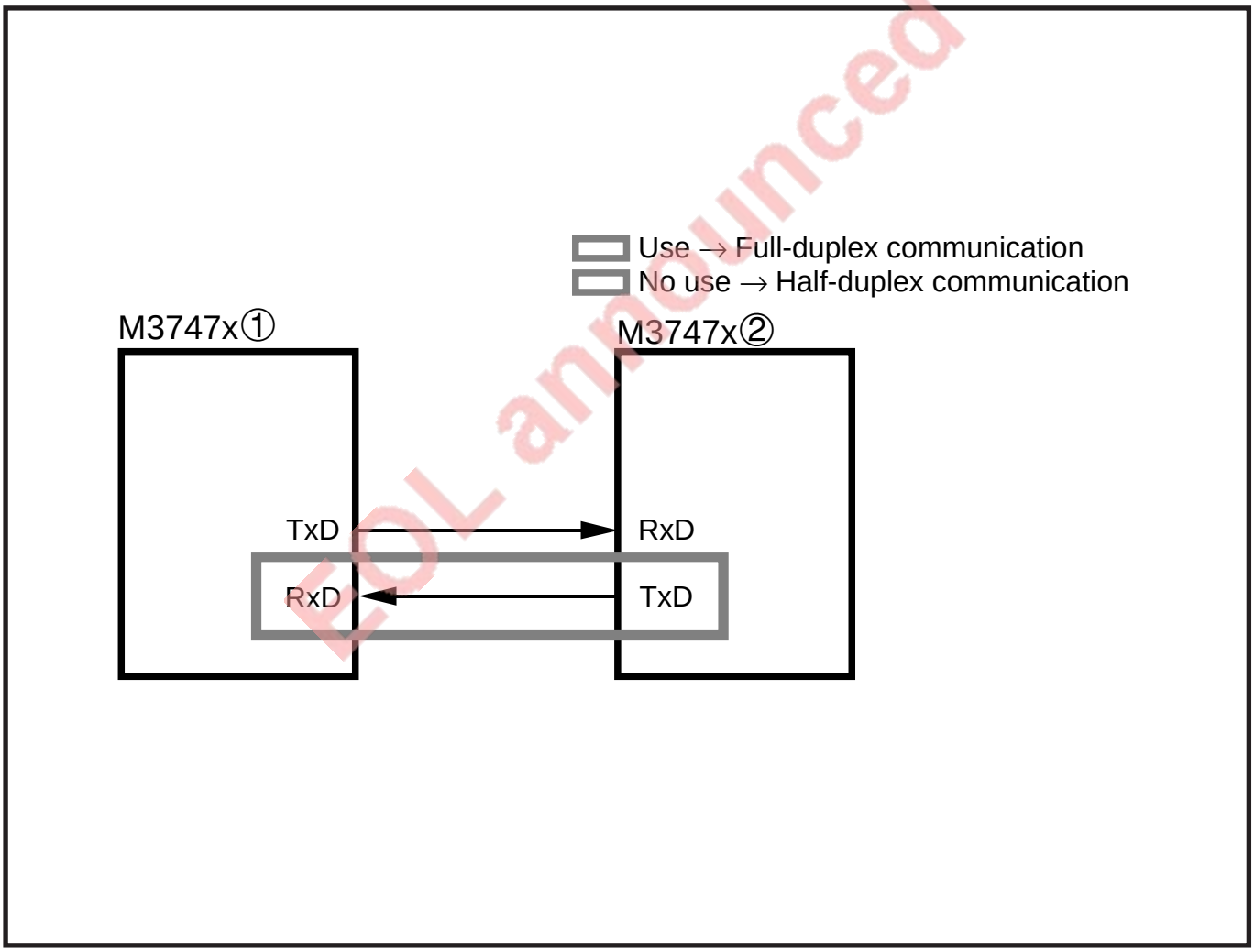


Fig. 2.4.10 Example of connections [Clock asynchronous serial I/O mode, 7477/7478 group]

7470	7471	7477	7478
×	×	○	○

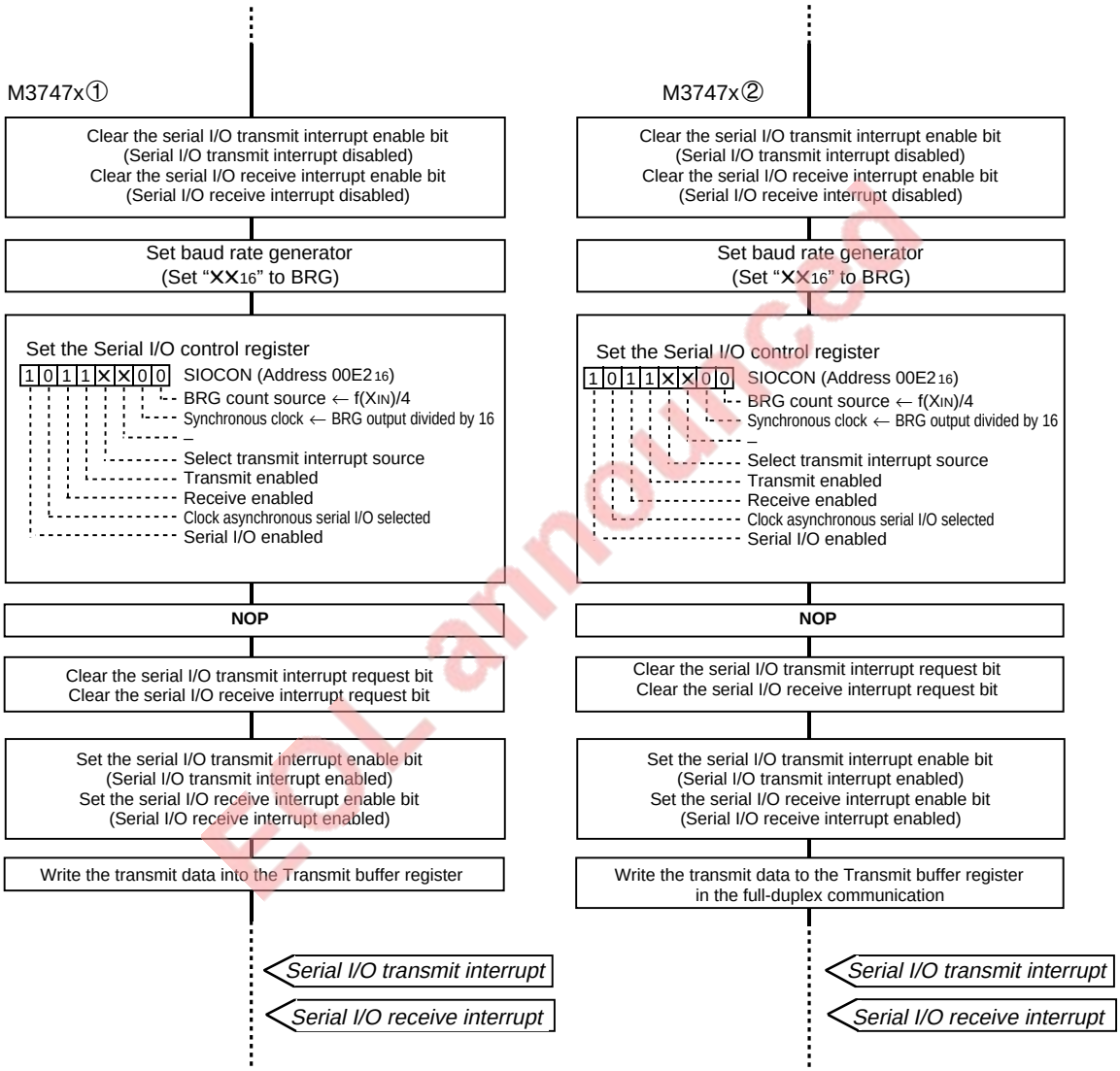


Fig. 2.4.11 Example of control procedure [Clock asynchronous serial I/O mode, 7477/7478 group]

2.4.5 Notes on use

7470	7471	7477	7478
○	○	○	○

■ For notes on use, refer to **"1.13 Serial I/O."**

2.5 A-D converter

7470	7471	7477	7478
○	○	○	○

2.5.1 Memory allocation

Figure 2.5.1 shows a memory map of A-D conversion related registers.

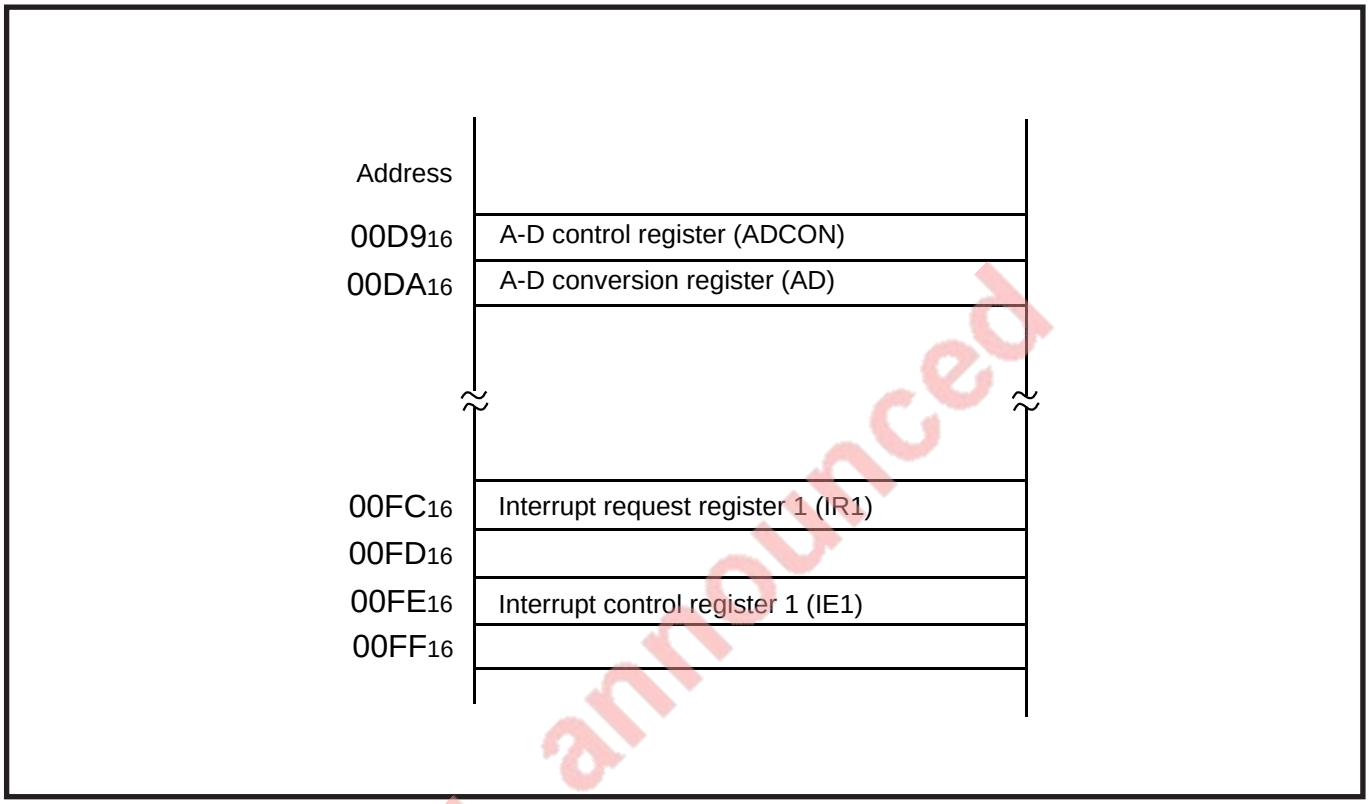


Fig. 2.5.1 Memory map of A-D conversion related registers

2.5.2 Application examples

7470	7471	7477	7478
○	○	○	○

(1) A-D conversion value determination methods

For improvement of the accuracy of A-D conversion results, we recommend you perform sampling several times to determine a value.
The following A-D conversion value sampling methods are available (m, n: Arbitrary values based on the specification).

Example: ①Sampling 2^n times
②Moving sampling 2^n times
③Sampling (2^n + 2) times

For value determination, the following methods are available.

Example: [1] The sum of sampling result is divided by sampling times.
[2] After execution of sampling (2^n + 2) times, the minimum value and the maximum value are excluded and then the remaining values are added and then divided by 2^n times.
[3] When updating the average value calculated by [1]or [2], this average value is not updated if the difference from the previous value is ± m or more.

In **"2.5.2 Application examples, (2) Example of A-D conversion setting,"** an example of using the sampling methods ② + ③ and the determination method [3] is shown.

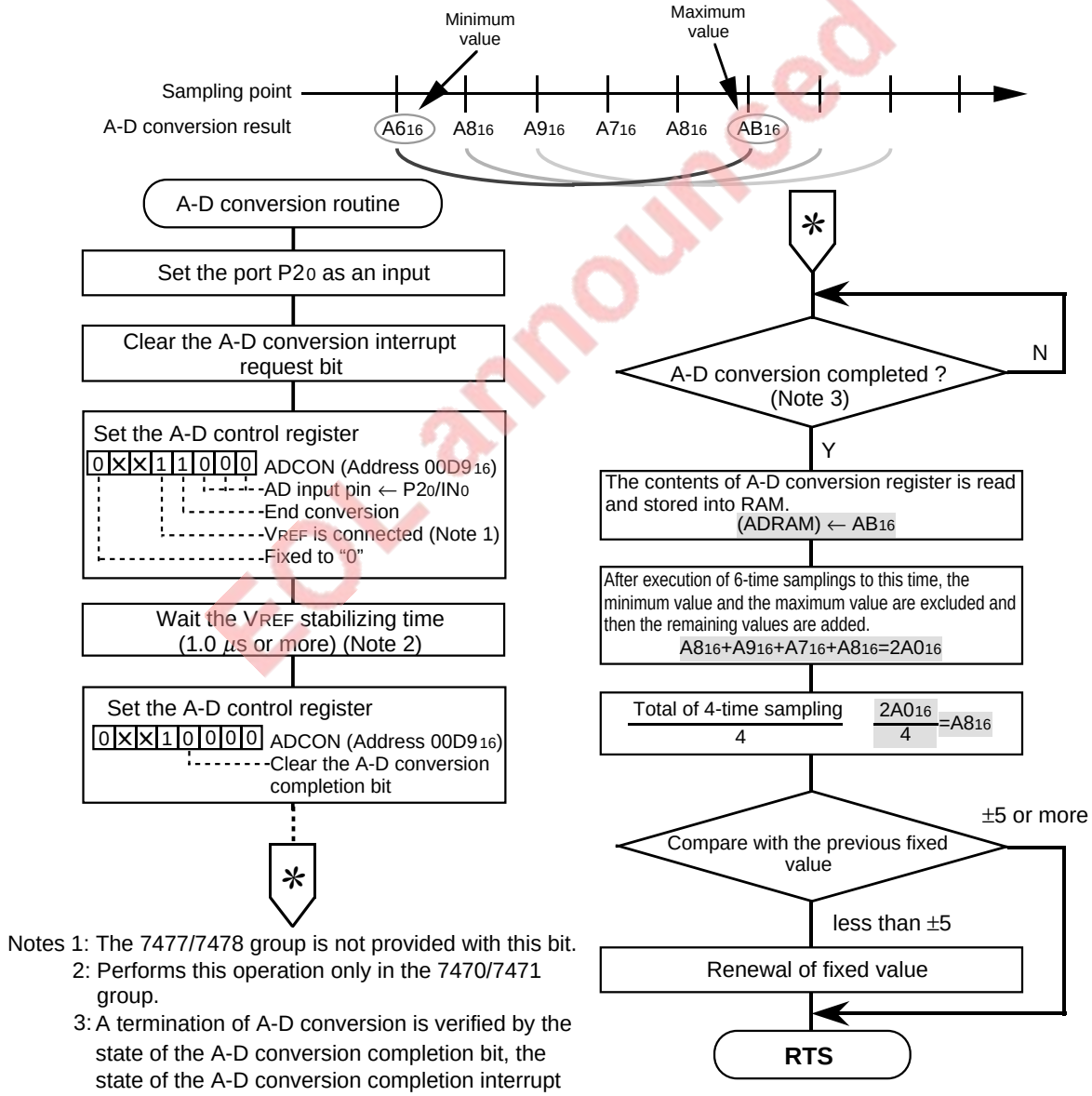
(2) Example of A-D conversion setting

7470	7471	7477	7478
○	○	○	○

An example of A-D conversion setting using the sampling methods ② + ③ and the determination method [3] described on the previous page is shown below.

Specifications: After execution of 6-time moving sampling the maximum value and the minimum value are excluded and then the remaining values are added. This result is divided by 4 (times). If the difference from the previous value is less than ±5, the value is updated. If the same difference is ±5 or more, it is not updated.

Figure 2.5.2 shows an example of control procedure.



Notes 1: The 7477/7478 group is not provided with this bit.
2: Performs this operation only in the 7470/7471 group.
3: A termination of A-D conversion is verified by the state of the A-D conversion completion bit, the state of the A-D conversion completion interrupt request bit and a branch to the A-D conversion completion interrupt processing routine.

Fig. 2.5.2 Example of A-D conversion control procedure

2.5 A-D converter

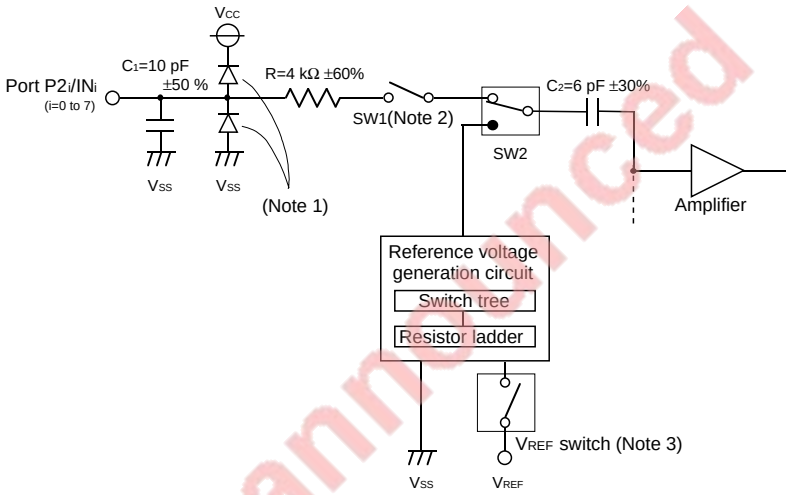
2.5.3 Notes on use

7470	7471	7477	7478
○	○	○	○

■ The analog input internal equivalent circuit is shown in Figure 2.5.3. For correct A-D conversion, it is necessary that the internal capacitor should be completely charged within the specified time. The maximum value of output impedance of the analog input source required to terminate capacitor charging within this time is shown below.

- At f(XIN) = 4 MHz, Approximately 10 kΩ
- At f(XIN) = 8 MHz, Approximately 2 kΩ

If the maximum value of output impedance exceeds the above value, take a proper measure, for example, insert a capacitor (0.1 μF to 1 μF) between analog input pin and VSS.



Notes 1: This is a parasitic diode of the output transistor.
2: SW1 is turned on only when the analog input pin is selected.
3: The VREF switch is not provided in the 7477/7478 group.

Fig. 2.5.3 Analog input internal equivalent circuit

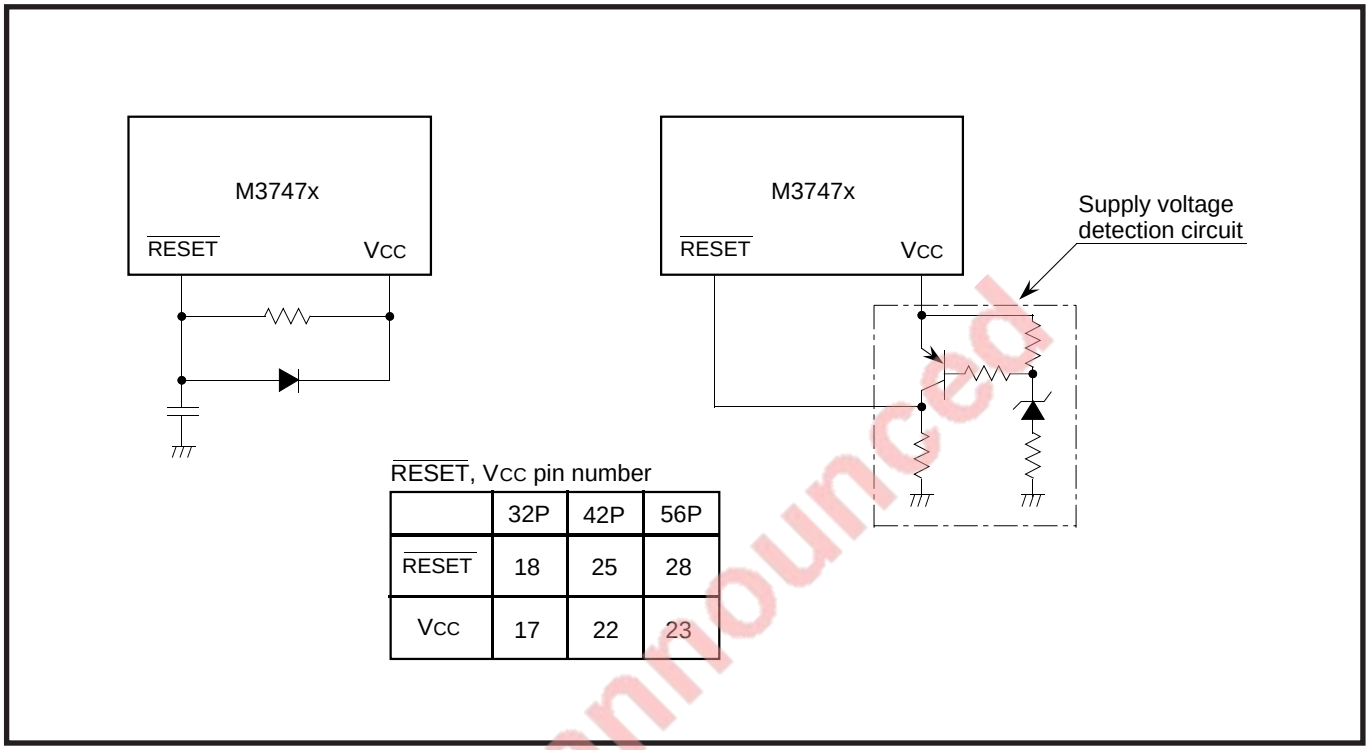
■ For other notes on use, refer to **"1.14 A-D converter."**

2.6 Reset

7470	7471	7477	7478
○	○	○	○

2.6.1 Reset circuit

Figure 2.6.1 shows an example of reset circuit.



RESET, Vcc pin number

	32P	42P	56P
RESET	18	25	28
Vcc	17	22	23

Fig. 2.6.1 Example of reset circuit

2.6.2 Notes on use

■ For notes on use, refer to **"1.15 Reset."**

2.7 Oscillation circuit

7470	7471	7477	7478
○	○	○	○

(1) Oscillation circuit using a ceramic resonator

An oscillation circuit can be formed by connecting a ceramic resonator or a crystal oscillator between the XIN pin and the XOUT pin and between the XCIN pin and the XCOUT pin.

Figure 2.7.1 shows an example of oscillation circuit using a ceramic resonator.

Regarding such circuit constants as Rd, CIN and COUT, ask the oscillator maker for information and then set the recommended value.

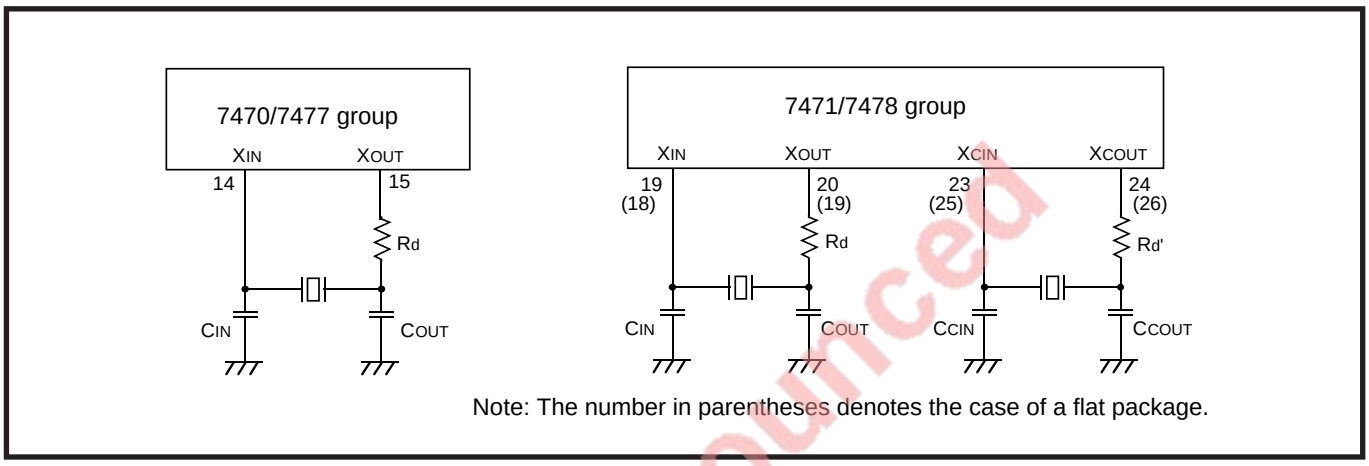


Fig. 2.7.1 Example of Oscillation circuit using ceramic resonator

(2) External clock input

To the main clock and timer clock oscillation circuits, clocks can also be supplied from the outside. Figure 2.7.2 shows an example of circuits in this case. At this time, make the XOUT (XCOUT) pin open. As an external clock to be input to the XIN (XCIN) pin, use a pulse signal with a duty ratio of 50 %.

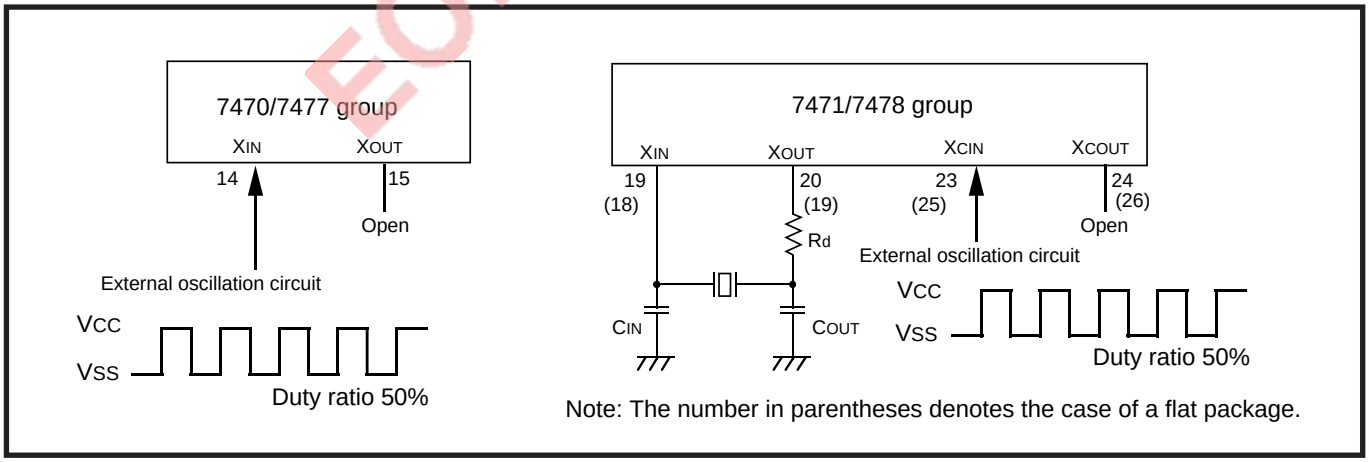


Fig. 2.7.2 Example of external clock input circuit

Note: The XCIN and the XCOUT pin are not provided in the 7470/7477 group.

2.8 Low-power dissipation function

7470	7471	7477	7478
○	○	○	○

2.8.1 CPU mode register

The CPU mode register consists of a stack page selection bit*1 and internal system clock control bits*2.

*1: In the products having a RAM capacity of 192 bytes or less, a RAM is not arranged on page 1, so this bit is not available. (Be sure to set this bit to "0.")
*2: In the 7470/7477 group, which is not provided with a sub-clock (f(XCIN)) generating circuit, f(XCIN) is not used. (Be sure to set this bit to "0.")

Figure 2.8.1 shows a structure of CPU mode register.

CPU mode register

b7 b6 b5 b4 b3 b2 b1 b0 — 0 0

CPU mode register (CPUM) [Address 00FB16]

B	Name	Function	At reset	R	W
0, 1	Fix these bits to "0."		0	○	○
2	Stack page selection bit	0: In page 0 area 1: In page 1 area (Note 1)	0	○	○
3	Nothing is allocated for this bit. This is write enabled bit and is undefined at reading.		?	?	X
4	P50, P51/XCIN,XCOUT selection bit	0: P50, P51 1: XCIN, XCOUT (Note 2)	0	○	○
5	XCOUT drive capacity selection bit	0: Low 1: High (Note 2)	0	○	○
6	Main clock (XIN–XOUT) stop bit	0: Oscillates 1: Stops (Note 2)	0	○	○
7	Internal system clock selection bit	0: XIN–XOUT selected (Ordinary mode) 1: XCIN–XCOUT selected (Low speed mode) (Note 2)	0	○	○

Notes 1: In the products having a RAM capacity of 192 bytes or less, set this bit to "0."
2: Since the 7470/7477 group is not provided with the sub-clock generating circuit, f(XCIN) cannot be used. Fix these bits to "0."

Fig. 2.8.1 Structure of CPU mode register

2.8 Low-power dissipation function

2.8.2 Application examples

7470	7471	7477	7478
○	○	○	○

As examples of application, examples of setting between modes are shown below.

(1) Ordinary mode → Stop mode → Ordinary mode
(2) Ordinary mode → Wait mode → Ordinary mode
(3) Ordinary mode → Low speed mode
(4) Low speed mode → Ordinary mode

Note: In the 7470/7477 group, which is not provided with a sub-clock generating circuit, the low-speed mode is not available.

(1) Ordinary mode → Stop mode → Ordinary mode

7470	7471	7477	7478
○	○	○	○

Specifications: The stop mode is executed by the STP instruction.
Restoration to the ordinary mode is attained by INT0 interrupt.

Figure 2.8.2 shows an example of control procedure.

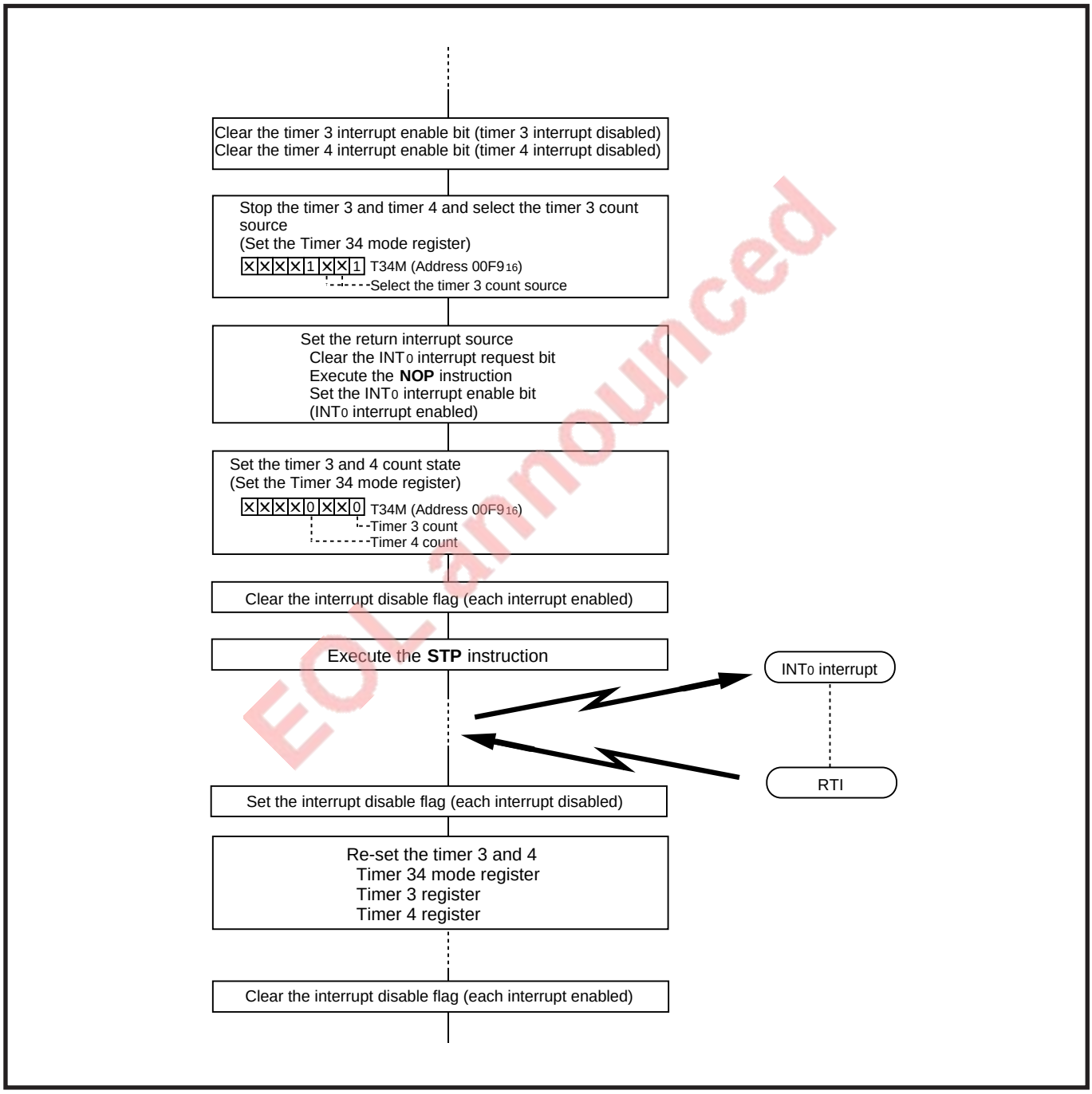


Fig. 2.8.2 Example of control procedure [Ordinary mode → Stop mode → Ordinary mode]

2.8 Low-power dissipation function

(2) Ordinary mode → Wait mode → Ordinary mode

7470	7471	7477	7478
○	○	○	○

Specifications: The wait mode is executed by the WIT instruction.
Restoration to the ordinary mode is attained by INT0 interrupt.

Figure 2.8.3 shows an example of control procedure.

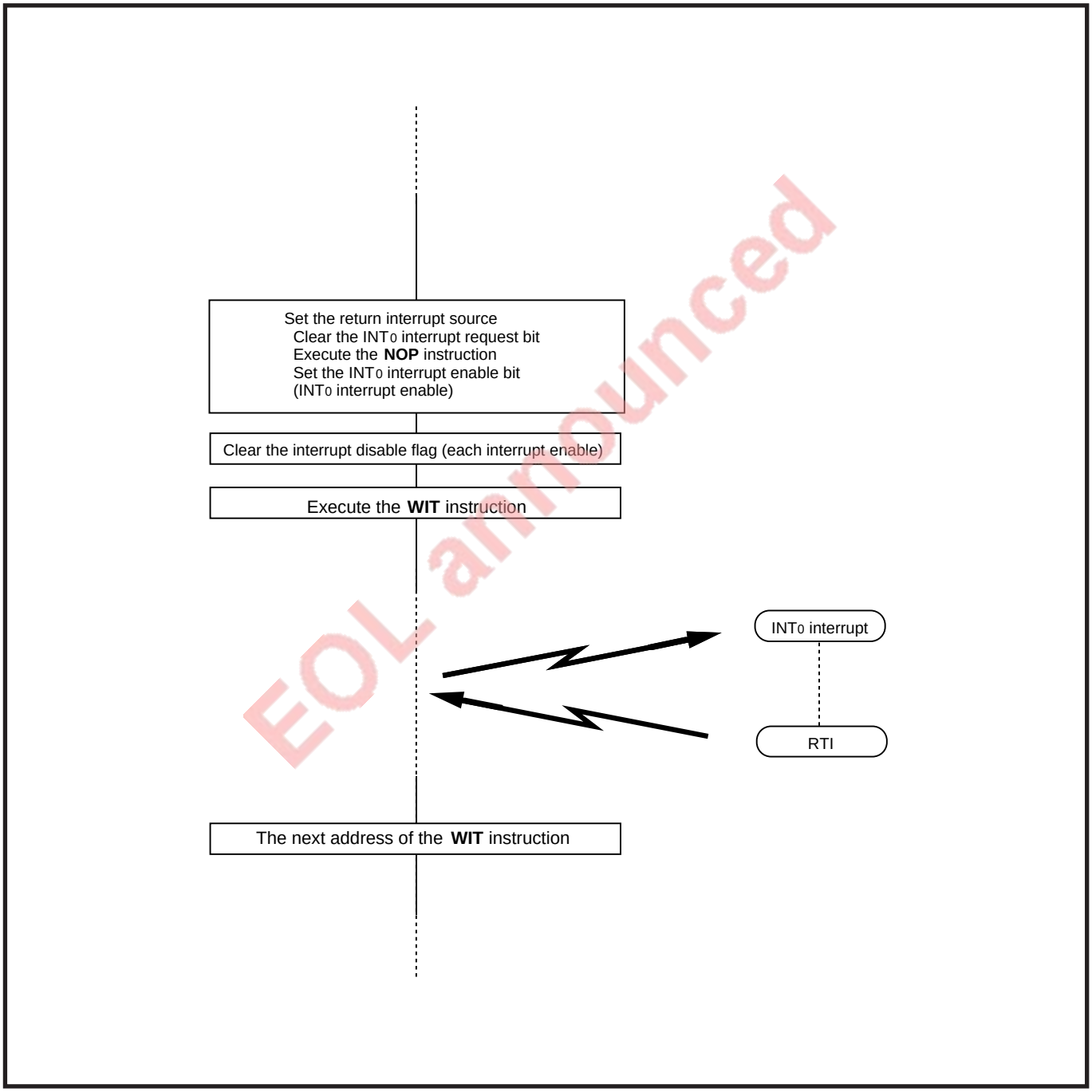


Fig. 2.8.3 Example of control procedure [Ordinary mode → Wait mode → Ordinary mode]

(3) Ordinary mode → Low speed mode

7470	7471	7477	7478
×	○	×	○

Specifications: The system clock is switched from the main clock (f(XIN) = 8 MHz) to the sub-clock (f(XCIN) = 32 kHz). The main clock is stopped.

Figure 2.8.4 shows an example of control procedure.

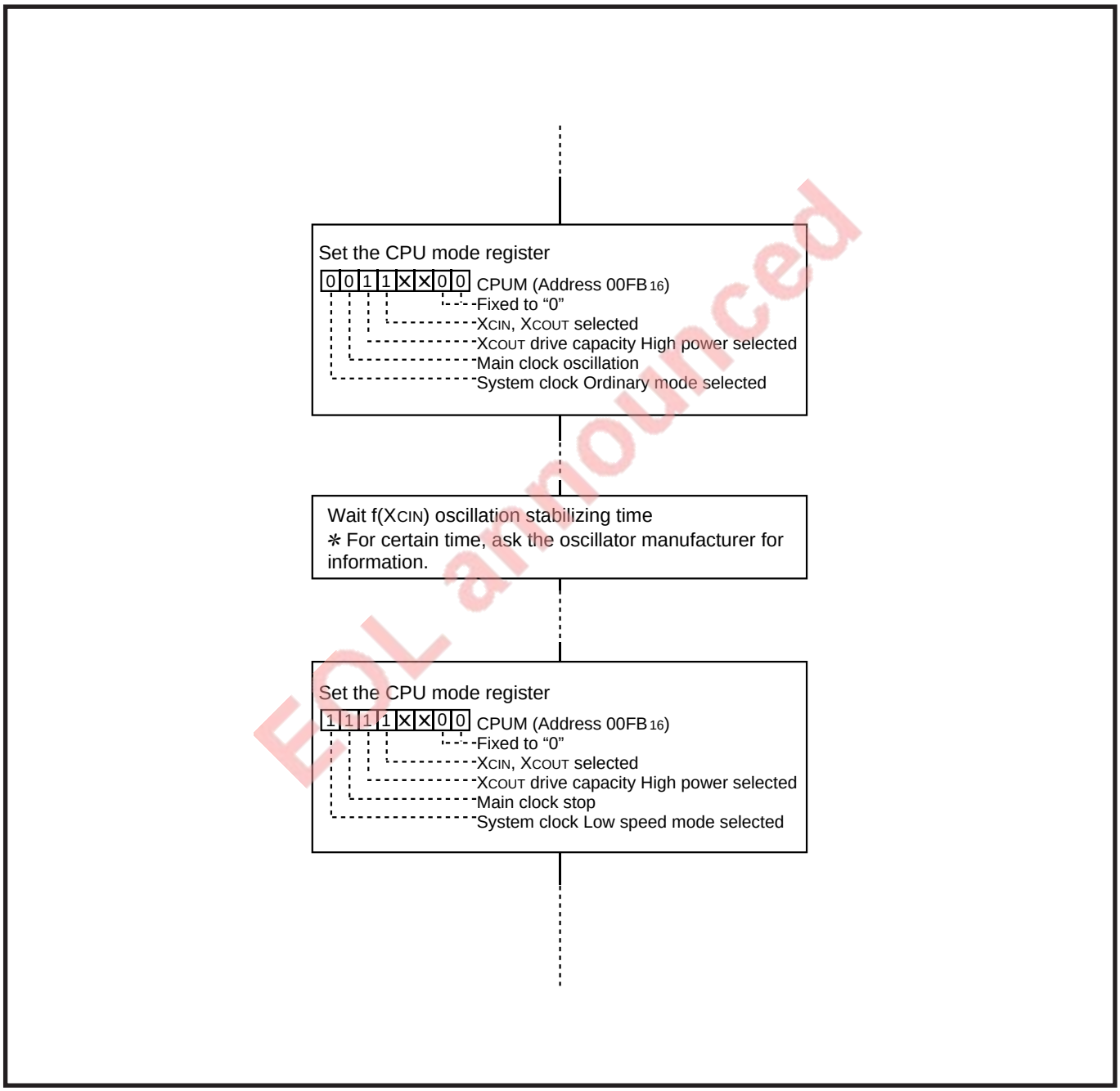


Fig. 2.8.4 Example of control procedure [Ordinary mode → Low speed mode]

2.8 Low-power dissipation function

(4) Low speed mode → Ordinary mode

7470	7471	7477	7478
×	○	×	○

Specifications: The system clock is switched from the sub-clock (f(XCIN) = 32 kHz) to the main clock (f(XIN) = 8 MHz). The sub-clock is stopped.

Figure 2.8.5 shows an example of control procedure.

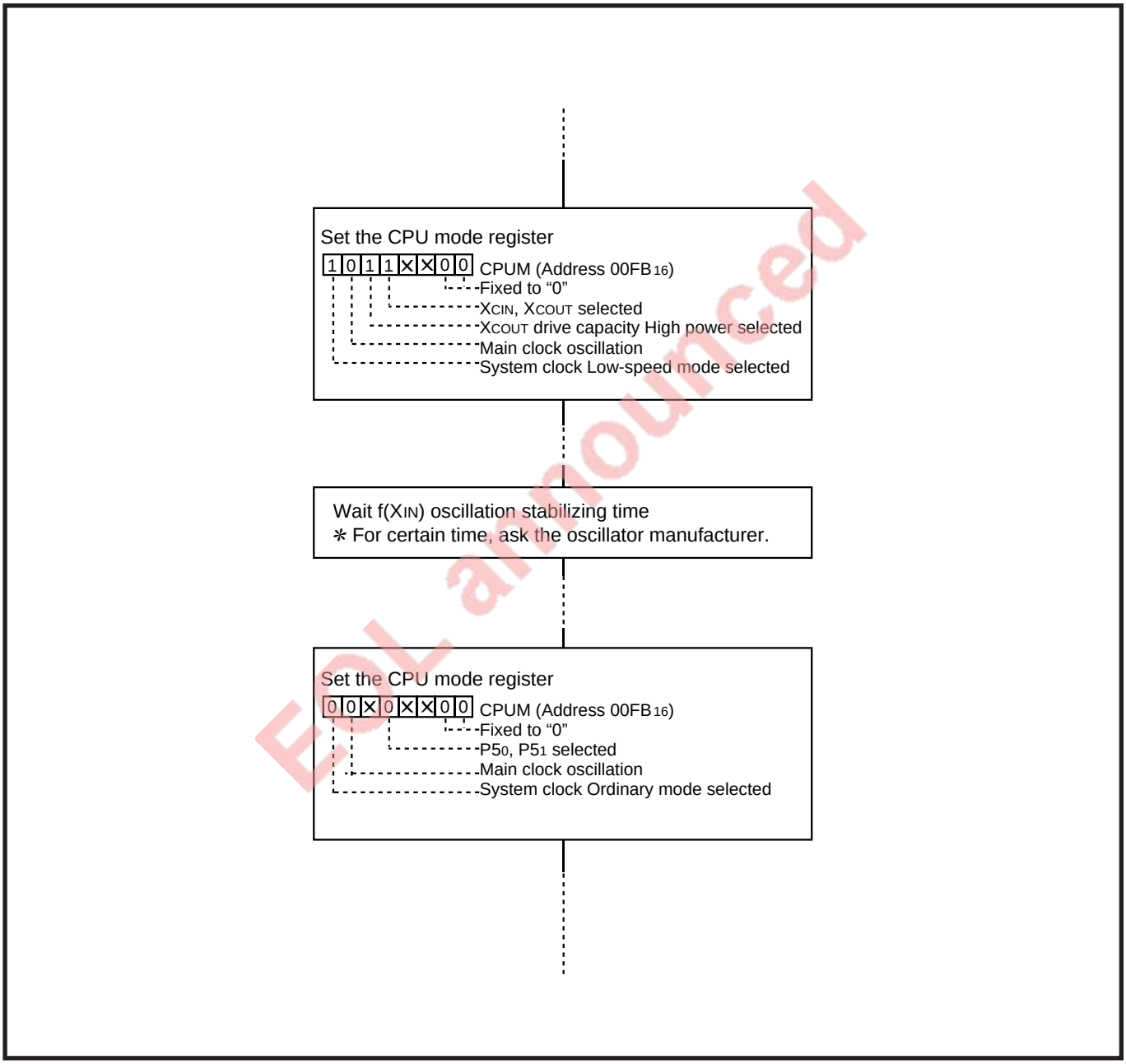


Fig. 2.8.5 Example of control procedure [Low speed mode → Ordinary mode]

2.8.3 Notes on use

7470	7471	7477	7478
○	○	○	○

■ For notes on use, refer to **“1.17 Low-power dissipation function.”**

2.9 Countermeasures against noise

7470	7471	7477	7478
○	○	○	○

Countermeasures against noise are described below. The following countermeasures are effective against noise in theory, however, it is necessary not only to take measures as follows but to evaluate before actual use.

2.9.1 Shortest wiring length

The wiring on a printed circuit board can be as an antenna which feeds noise into the microcomputer. The shorter the total wiring length (by mm unit), the less the possibility of noise insertion into a microcomputer.

(1) Wiring for the $\overline{\text{RESET}}$ 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 Vss pin with the shortest possible wiring (within 20mm).

Reason

The reset works to initialize a microcomputer.
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.

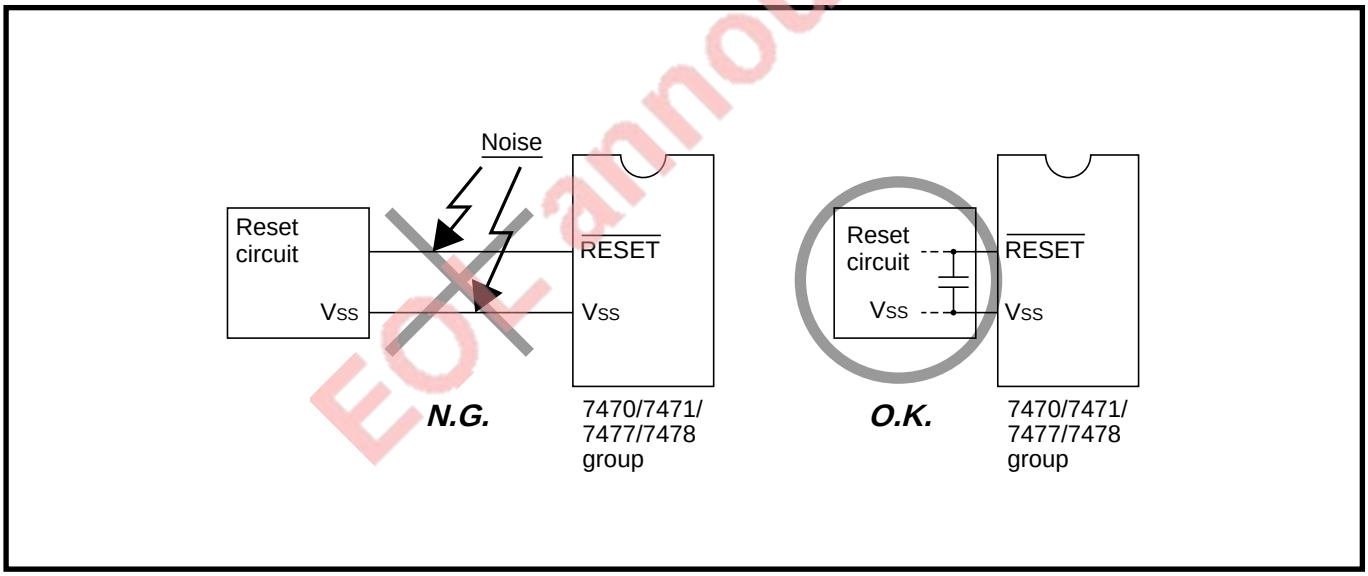


Fig. 2.9.1 Wiring for the $\overline{\text{RESET}}$ pin

(2) Wiring for clock input/output pins

7470	7471	7477	7478
○	○	○	○

- 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

A microcomputer's operation synchronizes with a clock generated by the oscillator (circuit). If noise enters clock I/O pins, clock waveforms may be deformed. This may cause a malfunction 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.

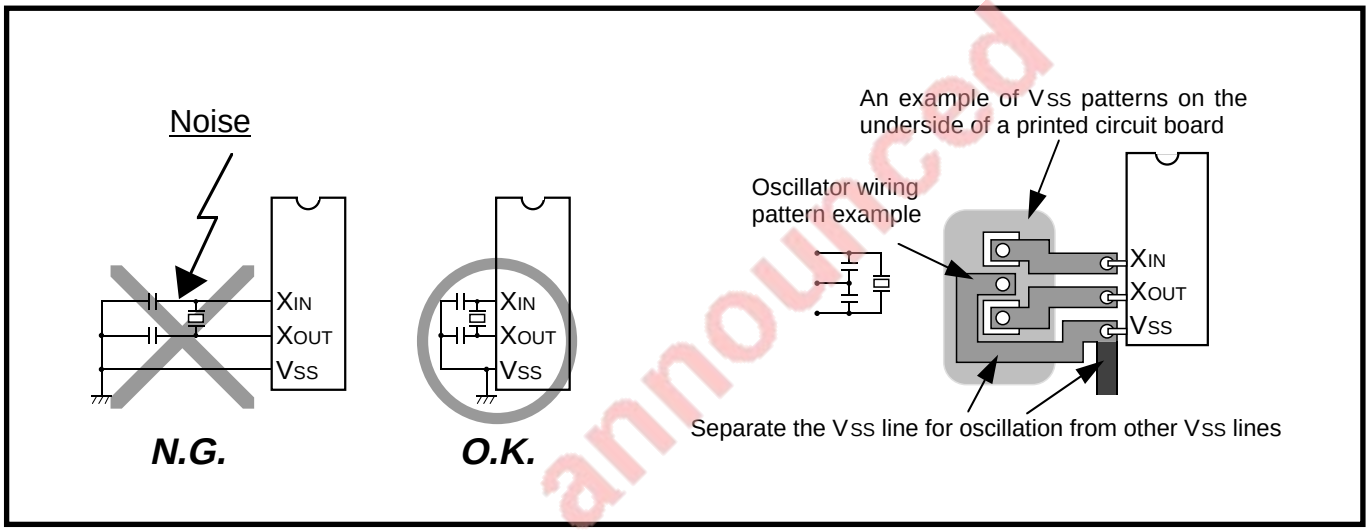


Fig. 2.9.2 Wiring for clock I/O pins

2.9 Countermeasures against noise

(3) Wiring for the V_{PP} pin of the One Time PROM version and the EPROM version

7470	7471	7477	7478
○	○	○	○

- Make the length of wiring which is connected to the V_{PP} pin as short as possible.
- Connect an approximately 5 kΩ resistor to the V_{PP} pin in serial.
- The $P3_3$ pin is also used as the V_{PP} pin.

Reason

The V_{PP} pin of the One Time PROM and the EPROM version 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 wiring 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.

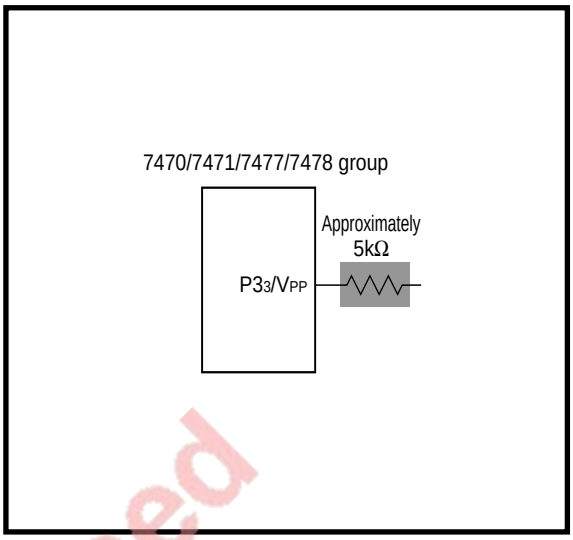


Fig. 2.9.3 Wiring for the V_{PP} pin of the One Time PROM and the EPROM version

2.9.2 Connection of a bypass capacitor across the Vss line and the Vcc line

7470	7471	7477	7478
○	○	○	○

Connect an approximately 0.1 μF bypass capacitor across the Vss line and the Vcc line as follows:

- Connect a bypass capacitor across the Vss pin and the Vcc pin at equal length.
- Connect a bypass capacitor across the Vss pin and the Vcc pin with the shortest possible wiring.
- Use lines with a larger diameter than other signal lines for Vss line and Vcc line.

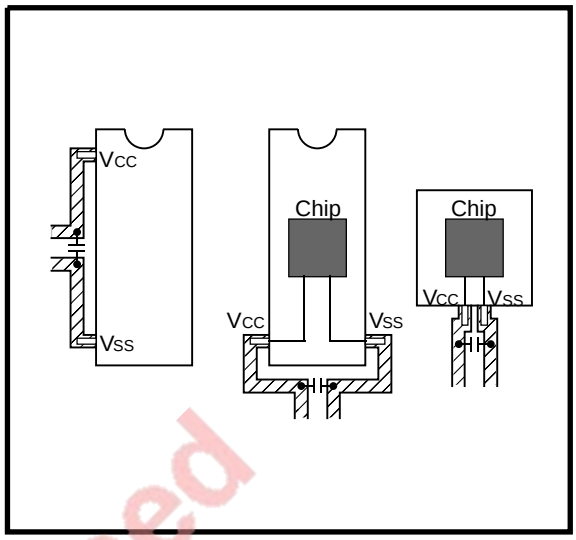


Fig. 2.9.4 Bypass capacitor across the Vss line and the Vcc line

2.9.3 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 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.

If a capacitor between an analog input pin and the Vss pin is grounded at a position far away from the Vss pin, noise on the GND line may enter a microcomputer through the capacitor.

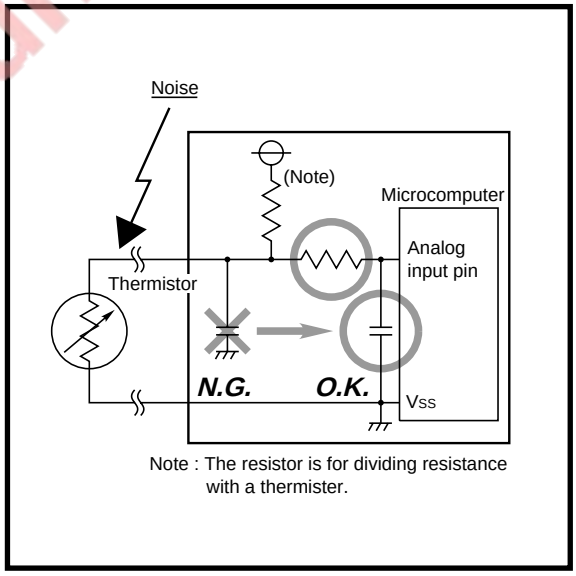


Fig. 2.9.5 Analog signal line and a resistor and a capacitor

2.9 Countermeasures against noise

2.9.4 Consideration for oscillator

7470	7471	7477	7478
○	○	○	○

Take care to prevent an oscillator that generates clocks for a microcomputer operation from being affected by other signals.

(1) Keeping an 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.

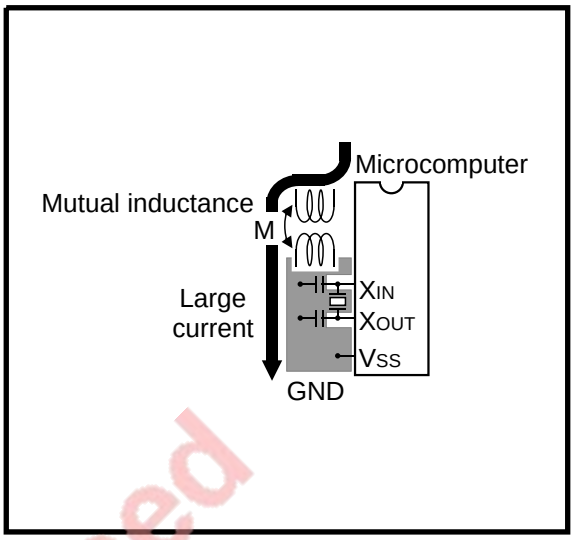


Fig. 2.9.6 Wiring for a large current signal line

(2) Keeping an oscillator away from signal lines where potential levels change frequently

Install an oscillator and a connecting pattern of an osillator 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 line) may affect other lines at signal rising 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.

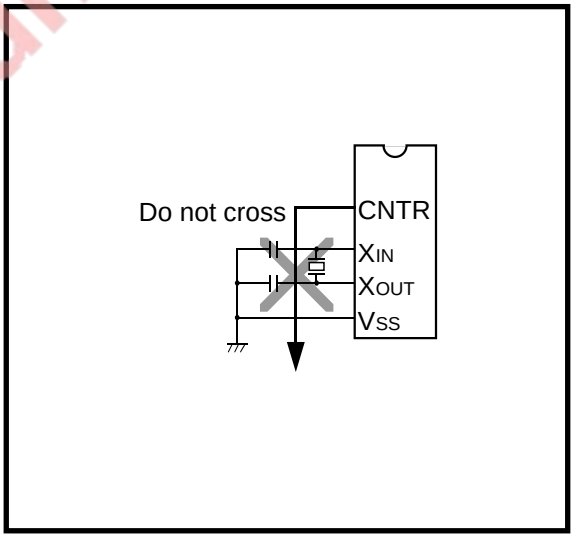


Fig. 2.9.7 Wiring to a signal line where potential levels change frequently

2.9.5 Setup for I/O ports

7470	7471	7477	7478
○	○	○	○

Setup I/O ports using hardware and software as follows:

<Hardware>

- Connect a resistor of 100 Ω or more to an I/O port inseries.

<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 (only the product having it) at fixed periods.

  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.

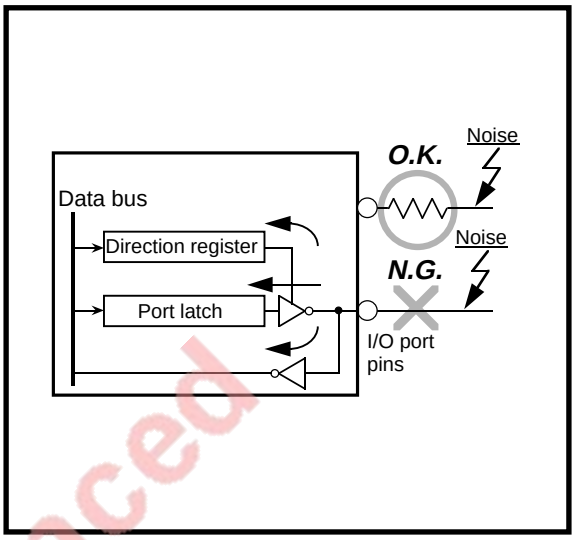


Fig. 2.9.8 Setup for I/O ports

2.9 Countermeasures against noise

2.9.6 Providing of watchdog timer function by software

7470	7471	7477	7478
○	○	○	○

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:

  $$N + 1 \geqq \text{(Counts of interrupt processing executed in each main routine)}$$

  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 count 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 cases:
  If the SWDT contents do not change after interrupt processing.

<The interrupt processing routine>

- Decrements the SWDT contents by 1 at each interrupt processing.
- Determins that the main routine operates normally when the SWDT contents are reset to the initial value N at almost fixed periods (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:
  When the contents of the SWDT reach 0 or less by continuative decrement without initializing to the initial value N.

7470	7471	7477	7478
○	○	○	○

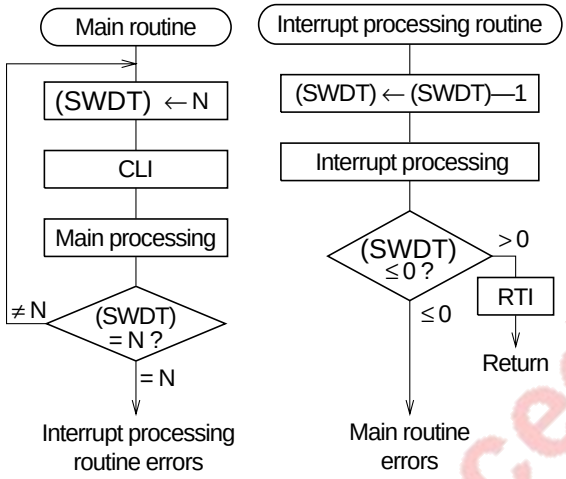


Fig. 2.9.9 Watchdog timer by software

2.10 Notes on programming

7470	7471	7477	7478
○	○	○	○

2.10.1 Processor status register

(1) Initialization of processor status register

After a reset, the contents of the processor status register (PS) are undefined except for the I flag which is "1." Therefore, 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.

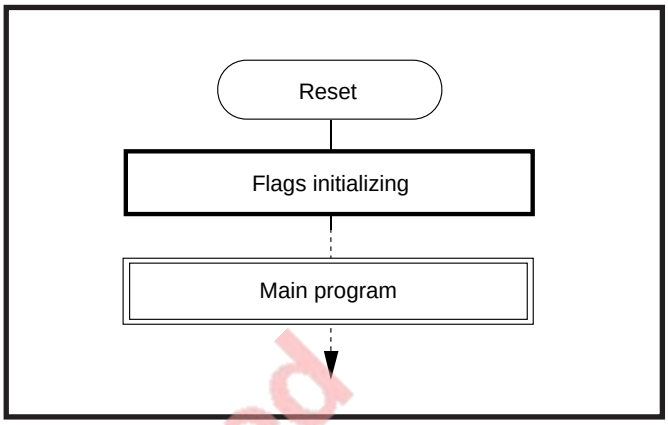


Fig. 2.10.1 Initialization of flags in PS

(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.

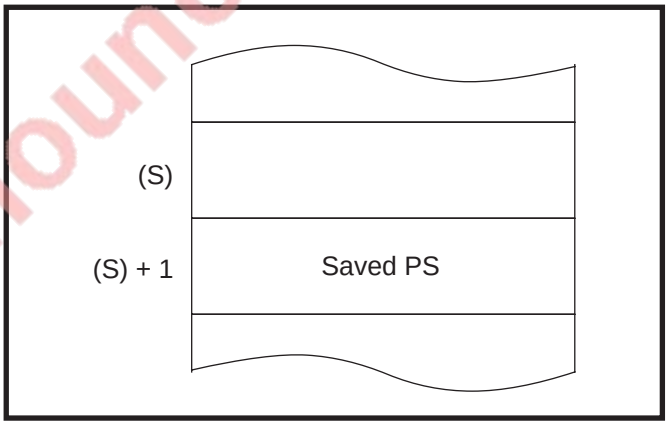


Fig. 2.10.2 Stack memory contents after PHP instruction execution

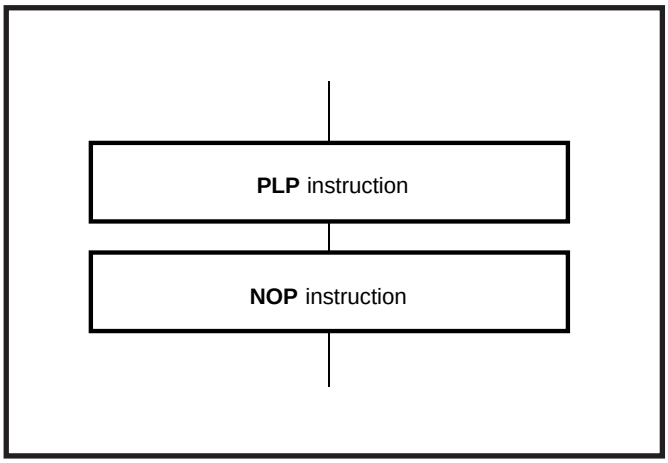


Fig. 2.10.3 Note to execute by PLP instruction

2.10.2 Decimal calculations

7470	7471	7477	7478
○	○	○	○

(1) Execution of decimal calculations

The ADC and SBC are the only instructions which will yield proper decimal results in decimal mode. To calculate in 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) Note on flags 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 before each calculation.

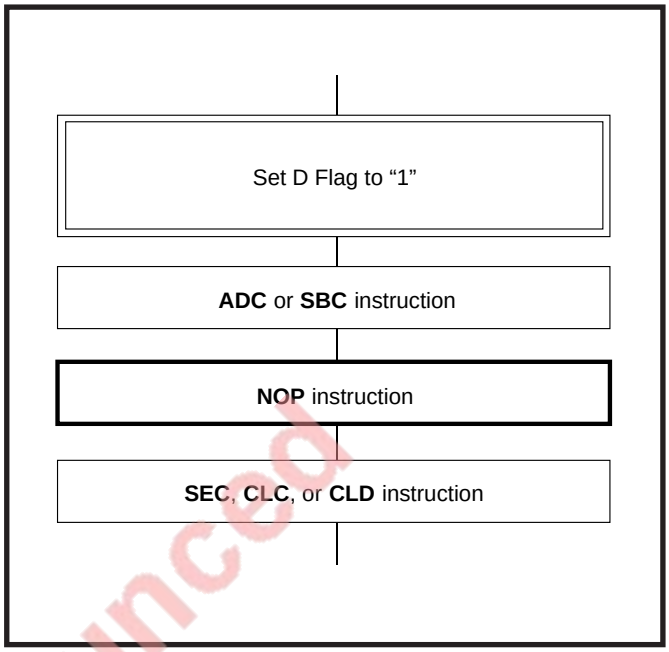


Fig. 2.10.4 Note for decimal operation

■ For other notes, refer to the notes described in each section.

2.11 Differences between 7470/7471 group and 7477/7478 group

7470	7471	7477	7478
○	○	○	○

Table 2.11.1 shows differences between the 7470/7471 group and the 7477/7478 group. Exercise due care at substitution.

Table 2.11.1 Differences between 7470/7471 group and 7477/7478 group

	7470/7471 group	7477/7478 group
RAM size	128/192/384 bytes	192/384 bytes
ROM size	4K/8K/16K byte	8K/16K byte
32-pin SOP	N	Y
Operating temperature range	-20 to 85°C	-20 to 85 °C
Interrupt source types	12	13
Serial I/O	Clock synchronous	Clock synchronous / UART
Byte specification mode	Y	N
Port P2	General I/O port / Analog input	General input port / Analog input
Software pull-up control	P0, P1, P2, P4, P5	P0, P1, P4, P5
A-D conversion VREF OFF function	Y	N

2.12 Example of application circuit

7470	7471	7477	7478
○	×	×	×

Figures 2.12.1 and 2.12.2 show examples of application circuit using the 7470 group, the 7478 group respectively.

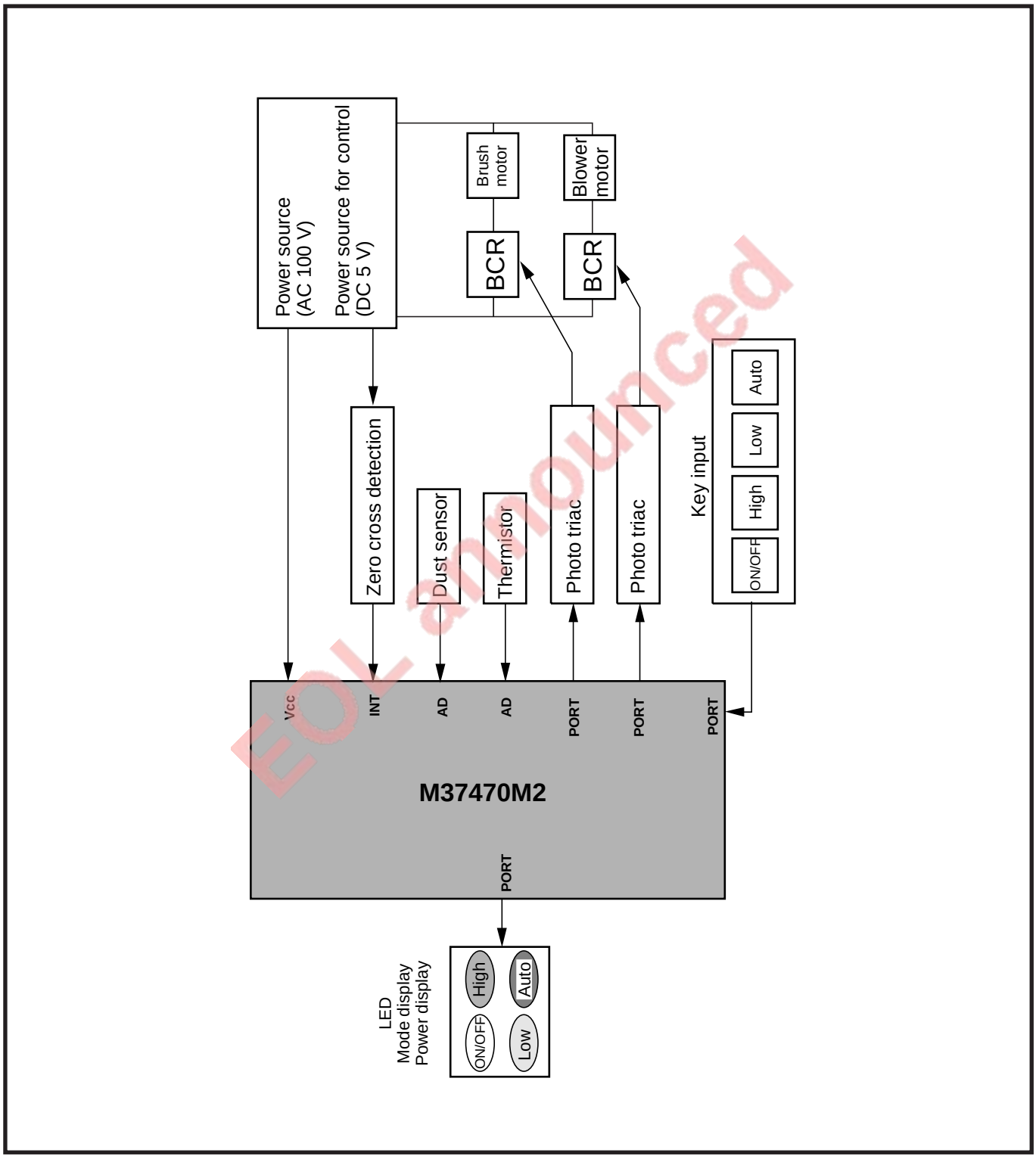
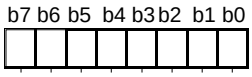


Fig. 2.12.1 Application circuit example (cleaner)

CHAPTER 3
APPENDIX

3.1 Control registers

Port Pi direction register



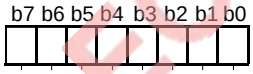
Port Pi direction register (PiD) (i = 0,1,2,4)
[Address 00C1$_{16}$, 00C3$_{16}$, 00C5$_{16}$, 00C9$_{16}$]

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	○	○

Notes 1: The 7477/7478 group is not provided with the port P2 direction register (input only).
2: The Port P4 is provided as below:
•7470/7477 group has 2 bits of P4$_0$ and P4$_1$
•7471/7478 group has 4 bits of P4$_0$ to P4$_3$.

Fig. 3.1.1 Structure of Port Pi direction register (i=0, 1, 2, 4)

Port P0 pull-up control register

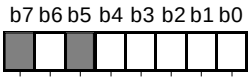


Port P0 pull-up control register [Address 00D0$_{16}$]

B	Name	Function	At reset	R	W
0	Port P0$_0$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
1	Port P0$_1$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
2	Port P0$_2$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
3	Port P0$_3$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
4	Port P0$_4$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
5	Port P0$_5$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
6	Port P0$_6$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
7	Port P0$_7$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○

Fig. 3.1.2 Structure of Port P0 pull-up control register

Ports P1 to P5 pull-up control register



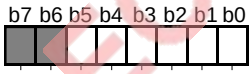
Ports P1 to P5 pull-up control register [Address 00D1$_{16}$]

b	Name	Function	At reset	R	W
0	Ports P1$_0$ to P1$_3$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
1	Ports P1$_4$ to P1$_7$ pull-up control bit	0 : No pull-up 1 : Pull-up	0	○	○
2	Ports P2$_0$ to P2$_3$ pull-up control bit (Note 2)	0 : No pull-up 1 : Pull-up	0	○	○
3	Ports P2$_4$ to P2$_7$ pull-up control bit (Notes 2, 3)	0 : No pull-up 1 : Pull-up	0	○	○
4	Ports P4$_0$ to P4$_3$ pull-up control bit (Note 4)	0 : No pull-up 1 : Pull-up	0	○	○
5	Nothing is allocated for this bit. This is write disabled bit and is undefined at reading.		?	?	×
6	Ports P5$_0$ to P5$_3$ pull-up control bit (Note 3)	0 : No pull-up 1 : Pull-up	0	○	○
7	Nothing is allocated for this bit. This is write disabled bit and is undefined at reading.		?	?	×

Notes 1 : In the 7470/7477 group, the P1 to P4 Pull-up control register is provided.
2 : In the 7477/7478 group, nothing is allocated to these bits. They are undefined at reading.
3 : In the 7470/7477 group, nothing is allocated to these bits. They are undefined at reading.
4 : The 7470/7477 group is provided with only P4$_0$ and P4$_1$.

Fig. 3.1.3 Structure of Ports P1 to P5 pull-up control register

Edge polarity selection register



Edge polarity selection register (EG) [Address 00D4$_{16}$]

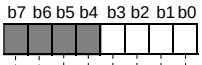
B	Name	Function	At reset	R	W
0	INT$_0$ edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
1	INT$_1$ edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
2	CNTR$_0$ edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
3	CNTR$_1$ edge selection bit	0 : Falling edge 1 : Rising edge	0	○	○
4	CNTR$_0$/CNTR$_1$ interrupt selection bit	0 : CNTR$_0$ 1 : CNTR$_1$	0	○	○
5	INT$_1$ source selection bit (at STP or WIT instruction execution)	0 : P3$_1$/INT$_1$ 1 : P0$_0$ to P0$_7$ "L" level input (for key-on wake-up)	0	○	○
6, 7	Nothing is allocated for these bits. These are write disabled bits and are undefined at reading.		?	?	×

Fig. 3.1.4 Structure of Edge polarity selection register

APPENDIX

3.1 Control registers

Input latch register



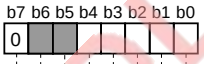
Input latch register (ILR) [Address 00D6$_{16}$]

B	Name	Function	At reset	R	W
0	P3$_0$/INT$_0$ latch bit	When b0 of EG (Note) is "0": reverse level on INT$_0$ pin When b0 of EG (Note) is "1": level on INT$_0$ pin	?	○	×
1	P3$_1$/INT$_1$ latch bit	When b1 of EG (Note) is "0": reverse level on INT$_1$ pin When b1 of EG (Note) is "1": level on INT$_1$ pin	?	○	×
2	P3$_2$/CNTR$_0$ latch bit	When b2 of EG (Note) is "0": reverse level on CNTR$_0$ pin When b2 of EG (Note) is "1": level on CNTR$_0$ pin	?	○	×
3	P3$_3$/CNTR$_1$ latch bit	When b3 of EG (Note) is "0": reverse level on CNTR$_1$ pin When b3 of EG (Note) is "1": level on CNTR$_1$ pin	?	○	×
4 to 7	Nothing is allocated for these bits. These are write disabled bits and are undefined at reading.		?	?	×

Note: EG is the Edge polarity selection register.

Fig. 3.1.5 Structure of Input latch register

A-D control register



A-D control register (ADCON) [Address 00D9$_{16}$]

B	Name	Function	At reset	R	W
0	A-D input selection bits	b2 b1 b0 0 0 0 : P2$_0$/IN$_0$ 0 0 1 : P2$_1$/IN$_1$ 0 1 0 : P2$_2$/IN$_2$ 0 1 1 : P2$_3$/IN$_3$ 1 0 0 : P2$_4$/IN$_4$ 1 0 1 : P2$_5$/IN$_5$ 1 1 0 : P2$_6$/IN$_6$ 1 1 1 : P2$_7$/IN$_7$ (Note 1)	0	○	○
1			0	○	○
2			0	○	○
3	A-D conversion end bit (Note 2)	0 : Under conversion 1 : End conversion	1	○	○
4	V$_{REF}$ connection selection bit *The 7477/7478 group is not provided with this bit. This bit is undefined at reset.	0 : The VREF pin is separated from the comparison voltage generator. 1 : The VREF pin is connected to comparison voltage generator.	0	○	○
5, 6	Nothing is allocated for these bits. These are write disabled bits and are undefined at reading.		?	?	×
7	Fix this bit to "0."		0	0	0

Notes 1: Since the 7470/7477 group is not provided with pins P2$_4$–P2$_7$, do not set.

2: •A-D conversion is started by setting bit 3 to "0."
•Writing "0" into bit 3 is valid. Even if "1" is written into bit 3, this bit is not set to "1." Accordingly, when writing a value into the A-D control register without affecting bit 3, set bit 3 to "1."

Fig. 3.1.6 Structure of A-D control register

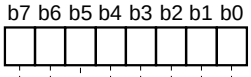
A-D conversion register

A-D conversion register (AD) [Address 00DA16]

B	Function	At reset	R	W
0 to 7	This is a read-only register to store A-D conversion results.	?	○	×

Fig. 3.1.7 Structure of A-D conversion register

Serial I/O mode register (7470/7471 group)



Serial I/O mode register (SM) [Address 00DC16]

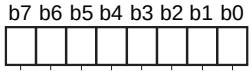
B	Name	Function	At reset	R	W
0, 1	Internal clock selection bits	b1 b0 0 0 : f(XIN)/8 or f(XCIN)/8 0 1 : f(XIN)/16 or f(XCIN)/16 1 0 : f(XIN)/32 or f(XCIN)/32 1 1 : f(XIN)/512 or f(XCIN)/512 (Note)	0	○	○
2	Synchronous clock selection bit	0 : External clock 1 : Internal clock	0	○	○
3	Serial I/O port selection bit	0 : Ordinary I/O port (P15, P16) 1 : Serial I/O port (SOUT, CLK pin)	0	○	○
4	$\overline{\text{SRDY}}$ signal output selection bit	0 : Ordinary I/O port(P17) 1 : $\overline{\text{SRDY}}$ signal output pin	0	○	○
5	$\overline{\text{SRDY}}$ signal selection bit	0 : $\overline{\text{SRDY}}$ signal 1 : SARDY signal	0	○	○
6	Serial I/O byte specify mode selection bit	0 : Ordinary mode 1 : Byte specify mode	0	○	○
7	P15/SOUT, $\overline{\text{SRDY}}$ output structure selection bit	0 : CMOS output 1 : N-channel open-drain output	0	○	○

Note: Since the 7470 group is not provided with the sub-clock generating circuit, do not select f(XCIN).

Fig. 3.1.8 Structure of Serial I/O mode register

3.1 Control registers

Serial I/O register (7470/7471 group)

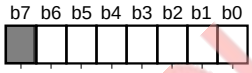


Serial I/O register (SIO) [Address 00DD16]

B	Function	At reset	R	W
0 to 7	At transmit: A value of "0016" to "FF16" can be set as transmit data. At the transmit, data is transmitted one bit at a time starting with the least significant bit. At receive: At the receive, data is received one bit at a time starting with the most significant bit.	?	○	○

Fig. 3.1.9 Structure of Serial I/O register

Serial I/O counter and Byte counter (7470/7471 group)

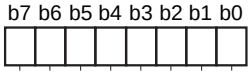


Serial I/O counter and Byte counter [Address 00DE16]

B	Function	At reset	R	W
0 to 3	Byte counter When using the byte specification mode, set a value of "0016" to "0F16." Supposing that the value to be written into the byte counter is "n," a Serial transmit/receive is performed with the clock of the "n + 1"-th byte.	?	○	○
4 to 6	Serial I/O counter When the internal clock is selected as a synchronous clock, this counter generates 8 shift clocks. When transmit data is written into the Serial I/O register, "0716" is set in the Serial I/O counter.	?	○	×
7	Nothing is allocated for this bit. This is write disabled bit and is undefined at reading.	?	?	×

Fig. 3.1.10 Structure of Serial I/O counter and Byte counter

Transmit/receive buffer register (7477/7478 group)

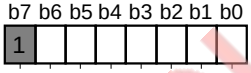


Transmit/receive buffer register (TB/RB) [Address 00E0$_{16}$]

B	Function	At reset	R	W
0 to 7	At transmit: A value of "00$_{16}$" to "FF$_{16}$" can be set as transmit data. The transmit data is transferred automatically by writing the transmit data into the Transmit shift register. At receive: When all receive data has been input into the Receive shift register, the receive data is automatically transferred to the receive buffer register.	?	○	○

Fig. 3.1.11 Structure of Transmit/receive buffer register

Serial I/O status register (7477/7478 group)



Serial I/O status register (SIOSTS) [Address 00E1$_{16}$]

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 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) U (PE) U (FE) = 0 1 : (OE) U (PE) U (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	1	×

Fig. 3.1.12 Structure of Serial I/O status register

3.1 Control registers

Serial I/O control register (7477/7478 group)

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O control register (SIOCON) [Address 00E2$_{16}$]

B	Name	Function	At reset	R	W
0	BRG count source selection bit (CSS)	0 : f(XIN)/4 or f(XCIN)/4 1 : f(XIN)/16 or f(XCIN)/16	0	○	○
1	Serial I/O synchronous clock selection bit (SCS)	•In clock synchronous mode 0 : BRG output divided by 4 1 : External clock input •In UART mode 0 : BRG output divided by 16 1 : External clock input divided by 16	0	○	○
2	$\overline{SRDY}$ output enable bit (SRDY) ✻In the UART mode, this bit is invalid.	0 : P17/$\overline{SRDY}$ pin operates as ordinary I/O pin 1 : P17/$\overline{SRDY}$ pin operates as $\overline{SRDY}$ output pin	0	○	○
3	Transmit interrupt source selection bit (TIC)	0 : When transmit buffer has emptied 1 : 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/O mode selection bit (SIOM)	0 : Clock asynchronous serial I/O (UART) 1 : Clock synchronous	0	○	○
7	Serial I/O enable bit (SIOE)	0 : Serial I/O disabled (pins operates as ordinary I/O pins P14–P17) 1 : Serial I/O enabled (pins operates as serial I/O pins RXD - $\overline{SRDY}$) (Note)	0	○	○

Note: Port P14–P17 are operates as the serial I/O pin only when the serial I/O enable bit is "1" (enable state).
At this time, Port P17 is also used as an ordinary I/O port.
In the UART mode, port P16 is used as an ordinary I/O port when the internal clock is selected.

Fig. 3.1.13 Structure of Serial I/O control register

UART control register (7477/7478 group)

b7 b6 b5 b4 b3 b2 b1 b0

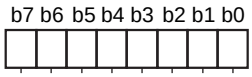
1 1 1 1

UART control register (UARTCON) [Address 00E3$_{16}$]

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 to 7	Nothing is allocated for these bits. These are write disabled bits. When these bits are read out, the values are "1."		1	1	×

Fig. 3.1.14 Structure of UART control register

Timers 1 to 4



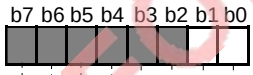
Timer 1, Timer 2, Timer 3, Timer 4 (T1, T2, T3, T4)
[Address $00F0_{16}$, $00F1_{16}$, $00F2_{16}$, $00F3_{16}$]

B	Function	At reset	R	W
0 to 7	•Set "00_{16} to FF_{16}." •The value is decremented by 1 each time a count source is input. •Each Timer values are set to the respective counter. •The count values are read out by reading the respective timer.	(Note)	○	○

Note : Timers 1 and 2 are undefined.
Timer 3 is "FF_{16}."
Timer 4 is "07_{16}."

Fig. 3.1.15 Structure of Timers 1 to 4

Timer FF register



Timer FF register (TF) [Address $00F7_{16}$]

B	Name	Function	At reset	R	W
0	Timer 1 division flip-flop	0 : Initial value is "0" 1 : Initial value is "1"	0	○	○
1	Timer 4 division flip-flop	0 : Initial value is "0" 1 : Initial value is "1"	0	○	○
2 to 7	Nothing is allocated for these bits. These are write disabled bits and are undefined at reading.		?	?	×

Fig. 3.1.16 Structure of Timer FF register

3.1 Control registers

Timer 12 mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer 12 mode register (T12M) [Address 00F8$_{16}$]

B	Name	Function	At reset	R	W
0	Timer 1 count stop bit	0 : Count start 1 : Count stop	0	○	○
1	Timer 1 count source selection bit	0 : Internal clock (Note 1) 1 : P3$_2$/CNTR$_0$ external clock	0	○	○
2	Timer 1 internal clock source selection bit	0 : f(X$_{IN}$)/16 or f(X$_{CIN}$)/16 1 : f(X$_{CIN}$) (Note 2)	0	○	○
3	P12/T$_0$ port output selection bit	0 : P12 port output 1 : T$_0$(Timer 1 overflow divided by 2)	0	○	○
4	Timer 2 count stop bit	0 : Count start 1 : Count stop	0	○	○
5	Timer 2 count source selection bit	0 : Internal clock (Note 1) 1 : Timer 1 overflow signal	0	○	○
6, 7	Timer 2 internal clock source selection bits	b7 b6 0 0 : f(X$_{IN}$)/16 or f(X$_{CIN}$)/16 0 1 : f(X$_{IN}$)/64 or f(X$_{CIN}$)/64 1 0 : f(X$_{IN}$)/128 or f(X$_{CIN}$)/128 1 1 : f(X$_{IN}$)/256 or f(X$_{CIN}$)/256 (Note 3)	0	○	○

Notes 1: In the 7470/7477 group, the internal clock is f(X$_{IN}$)/16.
2: Since the 7470/7477 group is not provided the sub-clock generating circuit, f(X$_{CIN}$) cannot be used. Fix this bit to "0."
3: Since the 7470/7477 group is not provided the sub-clock generating circuit, f(X$_{CIN}$) cannot be used.

Fig. 3.1.17 Structure of Timer 12 mode register

Timer 34 mode register

b7 b6 b5 b4 b3 b2 b1 b0

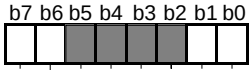
Timer 34 mode register (T34M) [Address 00F9$_{16}$]

B	Name	Function	At reset	R	W
0	Timer 3 count stop bit	0 : Count start 1 : Count stop	0	○	○
1, 2	Timer 3 count source selection bits	b2 b1 0 0 : f(X$_{IN}$)/16 or f(X$_{CIN}$)/16 0 1 : f(X$_{CIN}$) 1 0 : Timer 1 overflow or Timer 2 overflow 1 1 : P3$_3$/CNTR$_1$ external clock (Note 2)	0	○	○
3	Timer 4 count stop bit	0 : Count start 1 : Count stop	0	○	○
4, 5	Timer 4 count source selection bits	b4 b3 0 0 : Timer 3 overflow 0 1 : f(X$_{IN}$)/16 or f(X$_{CIN}$)/16 1 0 : Timer 1 overflow or Timer 2 overflow 1 1 : P3$_3$/CNTR$_1$ external clock (Notes 1, 2)	0	○	○
6	Timer 4 pulse width measurement mode selection bit	0 : Timer mode 1 : External pulse width measurement mode	0	○	○
7	P13/T$_1$ port output selection bit	0 : P13 port 1 : T$_1$(Timer 4 overflow divided by 2 or PWM output)	0	○	○

Notes 1: When Timer 1 overflow is selected as a Timer 2 count source, the Timer 4 count source is the Timer 1 overflow regardless of the value of bit 6 of the Timer mode register 2.
2: Since the 7470/7477 group is not provided the sub-clock generating circuit, f(X$_{CIN}$) cannot be used.

Fig. 3.1.18 Structure of Timer 34 mode register

Timer mode register 2

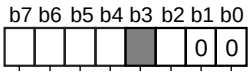


Timer mode register 2 (TM2) [Address 00FA16]

B	Name	Function	At reset	R	W
0	Timer 1 overflow FF set enable bit	0 : Set disable 1 : Set enable	0	○	○
1	Timer 4 overflow FF set enable bit	0 : Set disable 1 : Set enable	0	○	○
2 to 5	Nothing is allocated for these bits. These are write disabled bits and are undefined at reading.		?	?	×
6	Timer 3, timer 4 count overflow signal selection bit	0 : Timer 1 overflow 1 : Timer 2 overflow	0	○	○
7	Timer 3, timer 4 function selection bit	0 : Ordinary mode 1 : PWM mode	0	○	○

Fig. 3.1.19 Structure of Timer mode register 2

CPU mode register



CPU mode register (CPUM) [Address 00FB16]

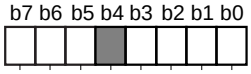
B	Name	Function	At reset	R	W
0, 1	Fix these bits to "0."		0	○	○
2	Stack page selection bit	0: In page 0 area 1: In page 1 area (Note 1)	0	○	○
3	Nothing is allocated for this bit. This is write enabled bit and is undefined at reading.		?	?	×
4	P50, P51/XCIN,XCOUT selection bit	0: P50, P51 1: XCIN, XCOUT (Note 2)	0	○	○
5	XCOUT drive capacity selection bit	0: Low 1: High (Note 2)	0	○	○
6	Main clock (XIN–XOUT) stop bit	0: Oscillates 1: Stops (Note 2)	0	○	○
7	Internal system clock selection bit	0: XIN–XOUT selected (Ordinary mode) 1: XCIN–XCOUT selected (Low speed mode) (Note 2)	0	○	○

Notes 1: In the products having a RAM capacity of 192 bytes or less, set this bit to "0."
2: Since the 7470/7477 group is not provided with the sub-clock generating circuit, f(XCIN) cannot be used. Fix these bits to "0."

Fig. 3.1.20 Structure of CPU mode register

3.1 Control registers

Interrupt request register 1



Interrupt request register 1 (IR1) [Address 00FC16]

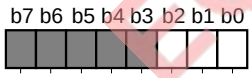
B	Name	Function	At reset	R	W
0	Timer 1 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
1	Timer 2 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
2	Timer 3 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
3	Timer 4 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
4	Nothing is allocated for this bit. This is write disabled bit and is undefined at reading.		?	?	×
5	Serial I/O receive interrupt request bit (7477/7478 group)(Note)	0 : No interrupt request 1 : Interrupt requested	0	○	✻
6	Serial I/O interrupt request bit (7470/7471group) Serial I/O transmit interrupt request bit (7477/7478 group)	0 : No interrupt request 1 : Interrupt requested	0	○	✻
7	A-D conversion completion interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻

Note: In the 7470/7471group, nothing is allocated for bit 5. This is write disabled bit and is undefined at reading.

✻ : "0" is set by software, but not "1."

Fig. 3.1.21 Structure of Interrupt request register 1

Interrupt request register 2



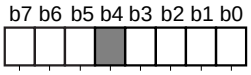
Interrupt request register 2 (IR2) [Address 00FD16]

B	Name	Function	At reset	R	W
0	INT0 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
1	INT1 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
2	CNTR0 or CNTR1 interrupt request bit	0 : No interrupt request 1 : Interrupt requested	0	○	✻
3 to 7	Nothing is allocated for these bits. There are write disabled bits and are undefined at reading.		?	?	×

✻ : "0" is set by software, but not "1."

Fig. 3.1.22 Structure of Interrupt request register 2

Interrupt control register 1



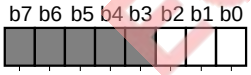
Interrupt control register 1 (IE1) [Address 00FE16]

B	Name	Function	At reset	R	W
0	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer 2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer 3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Timer 4 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Nothing is allocated for this bit. This is write disabled bit and is undefined at reading.		?	?	×
5	Serial I/O receive interrupt enable bit (7477/7478 group) (Note)	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Serial I/O interrupt enable bit (7470/7471 group) Serial I/O transmit interrupt enable bit (7477/7478 group)	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	A-D conversion completion interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Note: In the 7470/7471 group, Nothing is allocated for bit 5.
This is write disabled bit and undefined at reading.

Fig. 3.1.23 Structure of Interrupt control register 1

Interrupt control register 2



Interrupt control register 2 (IE2) [Address 00FF16]

B	Name	Function	At reset	R	W
0	INT_0 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
1	INT_1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
2	$CNTR_0$ or $CNTR_1$ interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3 to 7	Nothing is allocated for these bits. There are write disabled bits and are undefined at reading.		?	?	×

Fig. 3.1.24 Structure of Interrupt control register 2

3.2 Mask ROM ordering method

GZZ-SH02-91B<9YA0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37470M2-XXXSP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

✵ 1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based in this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ 27128	☐ 27256	☐ 27512
EPROM address	EPROM address	EPROM address
0000_{16} Area for ASCII codes of the name of the product 'M37470M2–'	0000_{16} Area for ASCII codes of the name of the product 'M37470M2–'	0000_{16} Area for ASCII codes of the name of the product 'M37470M2–'
$000F_{16}$	$000F_{16}$	$000F_{16}$
0010_{16}	0010_{16}	0010_{16}
$2FFF_{16}$	$6FFF_{16}$	$EFFF_{16}$
3000_{16} ROM (4K)	7000_{16} ROM (4K)	$F000_{16}$ ROM (4K)
$3FFF_{16}$	$7FFF_{16}$	$FFFF_{16}$

(1) Set "FF_{16}" in the shaded area.
(2) Write the ASCII codes that indicates the name of the product 'M37470M2–' to addresses 0000_{16} to $000F_{16}$. ASCII codes 'M37470M2–' are listed on the right. The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = $4D_{16}$	0008_{16}	' – ' = $2D_{16}$
0001_{16}	'3' = 33_{16}	0009_{16}	FF_{16}
0002_{16}	'7' = 37_{16}	$000A_{16}$	FF_{16}
0003_{16}	'4' = 34_{16}	$000B_{16}$	FF_{16}
0004_{16}	'7' = 37_{16}	$000C_{16}$	FF_{16}
0005_{16}	'0' = 30_{16}	$000D_{16}$	FF_{16}
0006_{16}	'M' = $4D_{16}$	$000E_{16}$	FF_{16}
0007_{16}	'2' = 32_{16}	$000F_{16}$	FF_{16}

GZZ-SH02-91B<9YA0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37470M2-XXXSP
MITSUBISHI ELECTRIC

Recommend to writing the following pseudo-command to the start address of the assembler source program.

EPROM type	27128	27256	27512
The pseudo-command	△*=△$C000 △ .BYTE△ 'M37470M2–'	△*=△$8000 △ .BYTE△ 'M37470M2–'	△*=△$0000 △ .BYTE△ 'M37470M2–'

Note : If the name of the product written to the EPROMs does not match the name of the mask confirmation, the ROM processing is disabled. Write the data correctly.

✺ 2. Mark specification

Mark specification must be submitted using the correct form for the type package being ordered fill out the appropriate mark specification form (32P4B for M37470M2-XXXSP) and attach to the mask ROM confirmation form.

✺ 3. Comments

(2/2)

APPENDIX

3.2 Mask ROM ordering method

GZZ-SH02-92B<9YA0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37470M4-XXXSP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

✵ 1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based in this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ 27128	☐ 27256	☐ 27512
EPROM address	EPROM address	EPROM address
0000_{16} Area for ASCII codes of the name of the product 'M37470M4–'	0000_{16} Area for ASCII codes of the name of the product 'M37470M4–'	0000_{16} Area for ASCII codes of the name of the product 'M37470M4–'
$000F_{16}$	$000F_{16}$	$000F_{16}$
0010_{16}	0010_{16}	0010_{16}
$1FFF_{16}$	$5FFF_{16}$	$DFFF_{16}$
2000_{16} ROM (8K)	6000_{16} ROM (8K)	$E000_{16}$ ROM (8K)
$3FFF_{16}$	$7FFF_{16}$	$FFFF_{16}$

(1) Set "FF_{16}" in the shaded area.
(2) Write the ASCII codes that indicates the name of the product 'M37470M4–' to addresses 0000_{16} to $000F_{16}$. ASCII codes 'M37470M4–' are listed on the right. The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = $4D_{16}$	0008_{16}	' – ' = $2D_{16}$
0001_{16}	'3' = 33_{16}	0009_{16}	FF_{16}
0002_{16}	'7' = 37_{16}	$000A_{16}$	FF_{16}
0003_{16}	'4' = 34_{16}	$000B_{16}$	FF_{16}
0004_{16}	'7' = 37_{16}	$000C_{16}$	FF_{16}
0005_{16}	'0' = 30_{16}	$000D_{16}$	FF_{16}
0006_{16}	'M' = $4D_{16}$	$000E_{16}$	FF_{16}
0007_{16}	'4' = 34_{16}	$000F_{16}$	FF_{16}

GZZ-SH02-92B<9YA0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37470M4-XXXSP
MITSUBISHI ELECTRIC

Recommend to writing the following pseudo-command to the start address of the assembler source program.

EPROM type	27128	27256	27512
The pseudo-command	△*=△$C000 △ .BYTE△ 'M37470M4–'	△*=△$8000 △ .BYTE△ 'M37470M4–'	△*=△$0000 △ .BYTE△ 'M37470M4–'

Note : If the name of the product written to the EPROMs does not match the name of the mask confirmation, the ROM processing is disabled. Write the data correctly.

✺ 2. Mark specification
Mark specification must be submitted using the correct form for the type package being ordered fill out the appropriate mark specification form (32P4B for M37470M4-XXXSP) and attach to the mask ROM confirmation form.

✺ 3. Comments

(2/2)

APPENDIX

3.2 Mask ROM ordering method

GZZ-SH02-93B<9YA0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37470M8-XXXSP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based in this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ 27256	☐ 27512
EPROM address	EPROM address
0000_{16} Area for ASCII codes of the name of the product 'M37470M8–'	0000_{16} Area for ASCII codes of the name of the product 'M37470M8–'
$000F_{16}$	$000F_{16}$
0010_{16}	0010_{16}
$3FFF_{16}$	$BFFF_{16}$
4000_{16} ROM (16K)	$C000_{16}$ ROM (16K)
$7FFF_{16}$	$FFFF_{16}$

(1) Set "FF_{16}" in the shaded area.
(2) Write the ASCII codes that indicates the name of the product 'M37470M8–' to addresses 0000_{16} to $000F_{16}$. ASCII codes 'M37470M8–' are listed on the right. The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = $4D_{16}$	0008_{16}	' – ' = $2D_{16}$
0001_{16}	'3' = 33_{16}	0009_{16}	FF_{16}
0002_{16}	'7' = 37_{16}	$000A_{16}$	FF_{16}
0003_{16}	'4' = 34_{16}	$000B_{16}$	FF_{16}
0004_{16}	'7' = 37_{16}	$000C_{16}$	FF_{16}
0005_{16}	'0' = 30_{16}	$000D_{16}$	FF_{16}
0006_{16}	'M' = $4D_{16}$	$000E_{16}$	FF_{16}
0007_{16}	'8' = 38_{16}	$000F_{16}$	FF_{16}

(1/2)

GZZ-SH02-93B<9YA0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37470M8-XXXSP
MITSUBISHI ELECTRIC

Recommend to writing the following pseudo-command to the start address of the assembler source program.

EPROM type	27256	27512
The pseudo-command	△*=△$8000 △ .BYTE△ 'M37470M8–'	△*=△$0000 △ .BYTE△ 'M37470M8–'

Note : If the name of the product written to the EPROMs does not match the name of the mask confirmation, the ROM processing is disabled. Write the data correctly.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type package being ordered fill out the appropriate mark specification form (32P4B for M37470M8-XXXSP) and attach to the mask ROM confirmation form.

✵ 3. Comments

(2/2)

GZZ-SH02-94B<9YB0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37471M2-XXXSP/FP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ❋.

❋ Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
	Date issued	Date:			

❋ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based in this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Microcomputer name : ☐ M37471M2-XXXSP ☐ M37471M2-XXXFP

Checksum code for entire EPROM [| | |] (hexadecimal notation)

EPROM type (indicate the type used)

☐ 27128		☐ 27256		☐ 27512	
EPROM address		EPROM address		EPROM address	
0000_{16} – $000F_{16}$	Area for ASCII codes of the name of the product 'M37471M2–'	0000_{16} – $000F_{16}$	Area for ASCII codes of the name of the product 'M37471M2–'	0000_{16} – $000F_{16}$	Area for ASCII codes of the name of the product 'M37471M2–'
0010_{16} – $2FFF_{16}$	(shaded)	0010_{16} – $6FFF_{16}$	(shaded)	0010_{16} – $EFFF_{16}$	(shaded)
3000_{16} – $3FFF_{16}$	ROM (4K)	7000_{16} – $7FFF_{16}$	ROM (4K)	$F000_{16}$ – $FFFF_{16}$	ROM (4K)

(1) Set "FF_{16}" in the shaded area.
(2) Write the ASCII codes that indicates the name of the product 'M37471M2–' to addresses 0000_{16} to $000F_{16}$. ASCII codes 'M37471M2–' are listed on the right. The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = $4D_{16}$	0008_{16}	' – ' = $2D_{16}$
0001_{16}	'3' = 33_{16}	0009_{16}	FF_{16}
0002_{16}	'7' = 37_{16}	$000A_{16}$	FF_{16}
0003_{16}	'4' = 34_{16}	$000B_{16}$	FF_{16}
0004_{16}	'7' = 37_{16}	$000C_{16}$	FF_{16}
0005_{16}	'1' = 31_{16}	$000D_{16}$	FF_{16}
0006_{16}	'M' = $4D_{16}$	$000E_{16}$	FF_{16}
0007_{16}	'2' = 32_{16}	$000F_{16}$	FF_{16}

(1/2)

GZZ-SH02-94B<9YB0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37471M2-XXXSP/FP
MITSUBISHI ELECTRIC

Recommend to writing the following pseudo-command to the start address of the assembler source program.

EPROM type	27128	27256	27512
The pseudo-command	△*=△$C000 △ .BYTE△ 'M37471M2–'	△*=△$8000 △ .BYTE△ 'M37471M2–'	△*=△$0000 △ .BYTE△ 'M37471M2–'

Note : If the name of the product written to the EPROMs does not match the name of the mask confirmation, the ROM processing is disabled. Write the data correctly.

✺ 2. Mark specification

Mark specification must be submitted using the correct form for the type package being ordered fill out the appropriate mark specification form (42P4B for M37471M2-XXXSP, 56P6N for M37471M2-XXXFP) and attach to the mask ROM confirmation form.

✺ 3. Comments

(2/2)

APPENDIX

3.2 Mask ROM ordering method

GZZ-SH02-95B<9YB0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37471M4-XXXSP/FP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵				Issuance signature	Submitted by	Supervisor
	Customer	Company name	TEL ( )			
		Date issued	Date:			

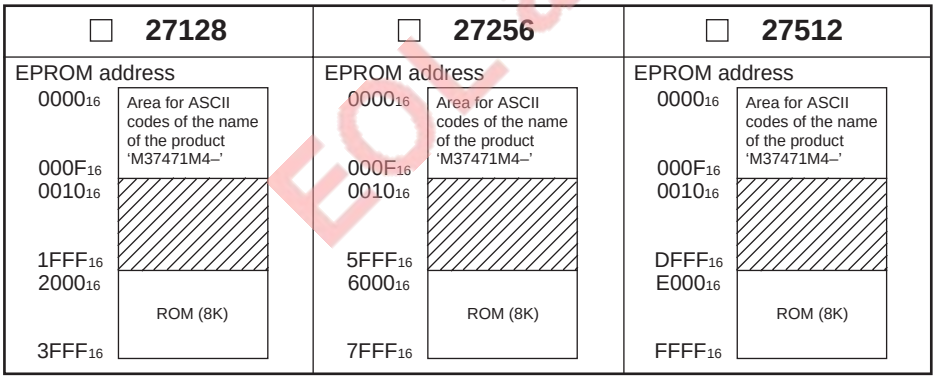
✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based in this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Microcomputer name : ☐ M37471M4-XXXSP ☐ M37471M4-XXXFP

Checksum code for entire EPROM [ ][ ][ ][ ] (hexadecimal notation)

EPROM type (indicate the type used)



(1) Set "FF$_{16}$" in the shaded area.
(2) Write the ASCII codes that indicates the name of the product 'M37471M4–' to addresses 0000$_{16}$ to 000F$_{16}$. ASCII codes 'M37471M4–' are listed on the right. The addresses and data are in hexadecimal notation.

Address		Address	
0000$_{16}$	'M' = 4D$_{16}$	0008$_{16}$	' – ' = 2D$_{16}$
0001$_{16}$	'3' = 33$_{16}$	0009$_{16}$	FF$_{16}$
0002$_{16}$	'7' = 37$_{16}$	000A$_{16}$	FF$_{16}$
0003$_{16}$	'4' = 34$_{16}$	000B$_{16}$	FF$_{16}$
0004$_{16}$	'7' = 37$_{16}$	000C$_{16}$	FF$_{16}$
0005$_{16}$	'1' = 31$_{16}$	000D$_{16}$	FF$_{16}$
0006$_{16}$	'M' = 4D$_{16}$	000E$_{16}$	FF$_{16}$
0007$_{16}$	'4' = 34$_{16}$	000F$_{16}$	FF$_{16}$

GZZ-SH02-95B<9YB0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37471M4-XXXSP/FP
MITSUBISHI ELECTRIC

Recommend to writing the following pseudo-command to the start address of the assembler source program.

EPROM type	27128	27256	27512
The pseudo-command	△*=△$C000 △ .BYTE△ 'M37471M4–'	△*=△$8000 △ .BYTE△ 'M37471M4–'	△*=△$0000 △ .BYTE△ 'M37471M4–'

Note : If the name of the product written to the EPROMs does not match the name of the mask confirmation, the ROM processing is disabled. Write the data correctly.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type package being ordered fill out the appropriate mark specification form (42P4B for M37471M4-XXXSP, 56P6N for M37471M4-XXXFP) and attach to the mask ROM confirmation form.

✵ 3. Comments

GZZ-SH02-96B<9YB0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37471M8-XXXSP/FP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

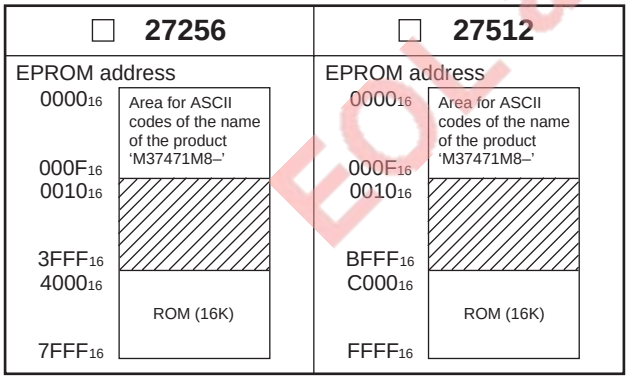
✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based in this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Microcomputer name : ☐ M37471M8-XXXSP ☐ M37471M8-XXXFP

Checksum code for entire EPROM [ ][ ][ ][ ] (hexadecimal notation)

EPROM type (indicate the type used)



(1) Set "FF$_{16}$" in the shaded area.
(2) Write the ASCII codes that indicates the name of the product 'M37471M8–' to addresses 0000$_{16}$ to 000F$_{16}$. ASCII codes 'M37471M8–' are listed on the right. The addresses and data are in hexadecimal notation.

Address		Address	
0000$_{16}$	'M' = 4D$_{16}$	0008$_{16}$	' – ' = 2D$_{16}$
0001$_{16}$	'3' = 33$_{16}$	0009$_{16}$	FF$_{16}$
0002$_{16}$	'7' = 37$_{16}$	000A$_{16}$	FF$_{16}$
0003$_{16}$	'4' = 34$_{16}$	000B$_{16}$	FF$_{16}$
0004$_{16}$	'7' = 37$_{16}$	000C$_{16}$	FF$_{16}$
0005$_{16}$	'1' = 31$_{16}$	000D$_{16}$	FF$_{16}$
0006$_{16}$	'M' = 4D$_{16}$	000E$_{16}$	FF$_{16}$
0007$_{16}$	'8' = 38$_{16}$	000F$_{16}$	FF$_{16}$

GZZ-SH02-96B<9YB0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37471M8-XXXSP/FP
MITSUBISHI ELECTRIC

Recommend to writing the following pseudo-command to the start address of the assembler source program.

EPROM type	27256	27512
The pseudo-command	△*=△$8000 △ .BYTE△ 'M37471M8–'	△*=△$0000 △ .BYTE△ 'M37471M8–'

Note : If the name of the product written to the EPROMs does not match the name of the mask confirmation, the ROM processing is disabled. Write the data correctly.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type package being ordered fill out the appropriate mark specification form (42P4B for M37471M8-XXXSP, 56P6N for M37471M8-XXXFP) and attach to the mask ROM confirmation form.

✵ 3. Comments

GZZ-SH06-67B<2XA1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37477M4-XXXSP/FP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three EPROMs are required for each pattern.
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Microcomputer name : ☐ M37477M4-XXXSP ☐ M37477M4-XXXFP

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ 27128	☐ 27256	☐ 27512
EPROM address	EPROM address	EPROM address
0000_{16} Product name ASCII code : 'M37477M4–'	0000_{16} Product name ASCII code : 'M37477M4–'	0000_{16} Product name ASCII code : 'M37477M4–'
$000F_{16}$	$000F_{16}$	$000F_{16}$
0010_{16}	0010_{16}	0010_{16}
$1FFF_{16}$	$5FFF_{16}$	$DFFF_{16}$
2000_{16} data ROM 8192 bytes	6000_{16} data ROM 8192 bytes	$E000_{16}$ data ROM 8192 bytes
$3FFF_{16}$	$7FFF_{16}$	$FFFF_{16}$

(1) Set the data in the unused area (the shaded area of the diagram) to "FF_{16}".
(2) The ASCII codes of the product name "M37477M4–" must be entered in addresses 0000_{16} to 0008_{16}. And set the data "FF_{16}" in addresses 0009_{16} to $000F_{16}$. The ASCII codes and addresses are listed to the right in hexadecimal notation.

Address		Address	
0000_{16}	'M' = $4D_{16}$	0008_{16}	' – ' = $2D_{16}$
0001_{16}	'3' = 33_{16}	0009_{16}	FF_{16}
0002_{16}	'7' = 37_{16}	$000A_{16}$	FF_{16}
0003_{16}	'4' = 34_{16}	$000B_{16}$	FF_{16}
0004_{16}	'7' = 37_{16}	$000C_{16}$	FF_{16}
0005_{16}	'7' = 37_{16}	$000D_{16}$	FF_{16}
0006_{16}	'M' = $4D_{16}$	$000E_{16}$	FF_{16}
0007_{16}	'4' = 34_{16}	$000F_{16}$	FF_{16}

GZZ-SH06-67B<2XA1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37477M4-XXXSP/FP
MITSUBISHI ELECTRIC

We recommend the use of the following pseudo-command to set the start address of the assembler source program.

EPROM type	27128	27256	27512
The pseudo-command	△*=△$C000 △ .BYTE△ 'M37477M4–'	△*=△$8000 △ .BYTE△ 'M37477M4–'	△*=△$0000 △ .BYTE△ 'M37477M4–'

Note : If the name of the product written to the EPROMs does not match the name of the mask confirmation form, the ROM will not be processed.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the package being ordered. Fill out the appropriate mark specification form (32P4B for M37477M4-XXXSP, 32P2W for M37477M4-XXXFP) and attach it to the mask ROM confirmation form.

✵ 3. Comments

APPENDIX

3.2 Mask ROM ordering method

GZZ-SH06-68B<2XA1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37477M8-XXXSP/FP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

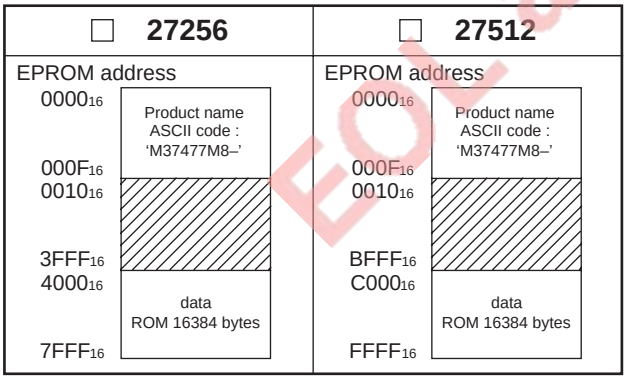
✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three EPROMs are required for each pattern.
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Microcomputer name : ☐ M37477M8-XXXSP ☐ M37477M8-XXXFP

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)



(1) Set the data in the unused area (the shaded area of the diagram) to "FF$_{16}$".
(2) The ASCII codes of the product name "M37477M8–" must be entered in addresses 0000$_{16}$ to 0008$_{16}$. And set the data "FF$_{16}$" in addresses 0009$_{16}$ to 000F$_{16}$. The ASCII codes and addresses are listed to the right in hexadecimal notation.

Address	
0000$_{16}$	'M' = 4D$_{16}$
0001$_{16}$	'3' = 33$_{16}$
0002$_{16}$	'7' = 37$_{16}$
0003$_{16}$	'4' = 34$_{16}$
0004$_{16}$	'7' = 37$_{16}$
0005$_{16}$	'7' = 37$_{16}$
0006$_{16}$	'M' = 4D$_{16}$
0007$_{16}$	'8' = 38$_{16}$

Address	
0008$_{16}$	' – ' = 2D$_{16}$
0009$_{16}$	FF$_{16}$
000A$_{16}$	FF$_{16}$
000B$_{16}$	FF$_{16}$
000C$_{16}$	FF$_{16}$
000D$_{16}$	FF$_{16}$
000E$_{16}$	FF$_{16}$
000F$_{16}$	FF$_{16}$

(1/2)

GZZ-SH06-68B<2XA1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37477M8-XXXSP/FP
MITSUBISHI ELECTRIC

We recommend the use of the following pseudo-command to set the start address of the assembler source program.

EPROM type	27256	27512
The pseudo-command	△*=△$8000 △ .BYTE△ 'M37477M8–'	△*=△$0000 △ .BYTE△ 'M37477M8–'

Note : If the name of the product written to the EPROMs does not match the name of the mask confirmation form, the ROM will not be processed.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the package being ordered. Fill out the appropriate mark specification form (32P4B for M37477M8-XXXSP, 32P2W for M37477M8-XXXFP) and attach it to the mask ROM confirmation form.

✵ 3. Comments

GZZ-SH06-70B<2XA0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37478M4-XXXSP/FP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✺.

✺	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

✺ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three EPROMs are required for each pattern.
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Microcomputer name : ☐ M37478M4-XXXSP ☐ M37478M4-XXXFP

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ 27128	☐ 27256	☐ 27512
EPROM address	EPROM address	EPROM address
0000_{16} Product name ASCII code : 'M37478M4–'	0000_{16} Product name ASCII code : 'M37478M4–'	0000_{16} Product name ASCII code : 'M37478M4–'
$000F_{16}$	$000F_{16}$	$000F_{16}$
0010_{16}	0010_{16}	0010_{16}
$1FFF_{16}$	$5FFF_{16}$	$DFFF_{16}$
2000_{16} data ROM 8192 bytes	6000_{16} data ROM 8192 bytes	$E000_{16}$ data ROM 8192 bytes
$3FFF_{16}$	$7FFF_{16}$	$FFFF_{16}$

(1) Set the data in the unused area (the shaded area of the diagram) to "FF_{16}".
(2) The ASCII codes of the product name "M37478M4–" must be entered in addresses 0000_{16} to 0008_{16}. And set the data "FF_{16}" in addresses 0009_{16} to $000F_{16}$. The ASCII codes and addresses are listed to the right in hexadecimal notation.

Address		Address	
0000_{16}	'M' = $4D_{16}$	0008_{16}	' – ' = $2D_{16}$
0001_{16}	'3' = 33_{16}	0009_{16}	FF_{16}
0002_{16}	'7' = 37_{16}	$000A_{16}$	FF_{16}
0003_{16}	'4' = 34_{16}	$000B_{16}$	FF_{16}
0004_{16}	'7' = 37_{16}	$000C_{16}$	FF_{16}
0005_{16}	'8' = 38_{16}	$000D_{16}$	FF_{16}
0006_{16}	'M' = $4D_{16}$	$000E_{16}$	FF_{16}
0007_{16}	'4' = 34_{16}	$000F_{16}$	FF_{16}

GZZ-SH06-70B<2XA0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37478M4-XXXSP/FP
MITSUBISHI ELECTRIC

We recommend the use of the following pseudo-command to set the start address of the assembler source program.

EPROM type	27128	27256	27512
The pseudo-command	△*=△$C000 △ .BYTE△ 'M37478M4–'	△*=△$8000 △ .BYTE△ 'M37478M4–'	△*=△$0000 △ .BYTE△ 'M37478M4–'

Note : If the name of the product written to the EPROMs does not match the name of the mask confirmation form, the ROM will not be processed.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the package being ordered. Fill out the appropriate mark specification form (42P4B for M37478M4-XXXSP, 56P6N for M37478M4-XXXFP) and attach it to the mask ROM confirmation form.

✵ 3. Comments

GZZ-SH06-71B<2XA0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37478M8-XXXSP/FP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

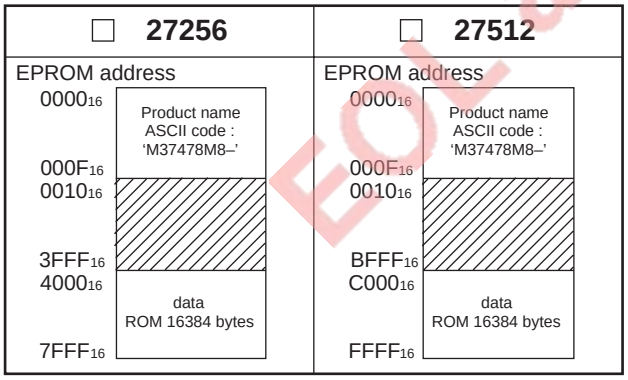
✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three EPROMs are required for each pattern.
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Microcomputer name : ☐ M37478M8-XXXSP ☐ M37478M8-XXXFP

Checksum code for entire EPROM [ ][ ][ ][ ] (hexadecimal notation)

EPROM type (indicate the type used)



(1) Set the data in the unused area (the shaded area of the diagram) to "FF$_{16}$".
(2) The ASCII codes of the product name "M37478M8–" must be entered in addresses 0000$_{16}$ to 0008$_{16}$. And set the data "FF$_{16}$" in addresses 0009$_{16}$ to 000F$_{16}$. The ASCII codes and addresses are listed to the right in hexadecimal notation.

Address		Address	
0000$_{16}$	'M' = 4D$_{16}$	0008$_{16}$	' – ' = 2D$_{16}$
0001$_{16}$	'3' = 33$_{16}$	0009$_{16}$	FF$_{16}$
0002$_{16}$	'7' = 37$_{16}$	000A$_{16}$	FF$_{16}$
0003$_{16}$	'4' = 34$_{16}$	000B$_{16}$	FF$_{16}$
0004$_{16}$	'7' = 37$_{16}$	000C$_{16}$	FF$_{16}$
0005$_{16}$	'8' = 38$_{16}$	000D$_{16}$	FF$_{16}$
0006$_{16}$	'M' = 4D$_{16}$	000E$_{16}$	FF$_{16}$
0007$_{16}$	'8' = 38$_{16}$	000F$_{16}$	FF$_{16}$

GZZ-SH06-71B<2XA0>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37478M8-XXXSP/FP
MITSUBISHI ELECTRIC

We recommend the use of the following pseudo-command to set the start address of the assembler source program.

EPROM type	27256	27512
The pseudo-command	△*=△$8000 △ .BYTE△ 'M37478M8–'	△*=△$0000 △ .BYTE△ 'M37478M8–'

Note : If the name of the product written to the EPROMs does not match the name of the mask confirmation form, the ROM will not be processed.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the package being ordered. Fill out the appropriate mark specification form (42P4B for M37478M8-XXXSP, 56P6N for M37478M8-XXXFP) and attach it to the mask ROM confirmation form.

✵ 3. Comments

3.3 ROM programming ordering method

GZZ-SH03-60B<06A0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37470E4-XXXSP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

✵ 1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain identical data, we will produce ROM programming based on this data. We shall assume the responsibility for errors only if the ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ 27128	☐ 27256	☐ 27512
EPROM address	EPROM address	EPROM address
0000_{16} Area for ASCII codes of the name of the product 'M37470E4–'	0000_{16} Area for ASCII codes of the name of the product 'M37470E4–'	0000_{16} Area for ASCII codes of the name of the product 'M37470E4–'
$000F_{16}$	$000F_{16}$	$000F_{16}$
0010_{16}	0010_{16}	0010_{16}
$1FFF_{16}$	$5FFF_{16}$	$DFFF_{16}$
2000_{16} ROM (8K)	6000_{16} ROM (8K)	$E000_{16}$ ROM (8K)
$3FFF_{16}$	$7FFF_{16}$	$FFFF_{16}$

(1) Set "FF_{16}" in the shaded area.
(2) Write the ASCII codes that indicates the name of the product 'M37470E4–' to addresses 0000_{16} to $000F_{16}$. ASCII codes 'M37470E4–' are listed on the right. The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = $4D_{16}$	0008_{16}	' – ' = $2D_{16}$
0001_{16}	'3' = 33_{16}	0009_{16}	FF_{16}
0002_{16}	'7' = 37_{16}	$000A_{16}$	FF_{16}
0003_{16}	'4' = 34_{16}	$000B_{16}$	FF_{16}
0004_{16}	'7' = 37_{16}	$000C_{16}$	FF_{16}
0005_{16}	'0' = 30_{16}	$000D_{16}$	FF_{16}
0006_{16}	'E' = 45_{16}	$000E_{16}$	FF_{16}
0007_{16}	'4' = 34_{16}	$000F_{16}$	FF_{16}

GZZ-SH03-60B<06A0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37470E4-XXXSP
MITSUBISHI ELECTRIC

Recommend to writing the following pseudo-command to the start address of the assembler source program.

EPROM type	27128	27256	27512
The pseudo-command	△*=△$C000 △ .BYTE△ 'M37470E4–'	△*=△$8000 △ .BYTE△ 'M37470E4–'	△*=△$0000 △ .BYTE△ 'M37470E4–'

Note : If the name of the product written to the EPROMs does not match the name of the ROM programming confirmation, the ROM processing is disabled. Write the data correctly.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type of package being ordered. Please submit the shrink DIP package Mark Specification Form (only for built-in One Time PROM microcomputer).

✵ 3. Comments

(2/2)

GZZ-SH02-97B<9YA0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37470E8-XXXSP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain identical data, we will produce ROM programming based on this data. We shall assume the responsibility for errors only if the ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ 27256	☐ 27512
EPROM address	EPROM address
0000_{16} Area for ASCII codes of the name of the product 'M37470E8–'	0000_{16} Area for ASCII codes of the name of the product 'M37470E8–'
$000F_{16}$	$000F_{16}$
0010_{16}	0010_{16}
$3FFF_{16}$ ROM (16K)	$BFFF_{16}$ ROM (16K)
4000_{16}	$C000_{16}$
$7FFF_{16}$	$FFFF_{16}$

(1) Set "FF_{16}" in the shaded area.
(2) Write the ASCII codes that indicates the name of the product 'M37470E8–' to addresses 0000_{16} to $000F_{16}$. ASCII codes 'M37470E8–' are listed on the right. The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = $4D_{16}$	0008_{16}	' – ' = $2D_{16}$
0001_{16}	'3' = 33_{16}	0009_{16}	FF_{16}
0002_{16}	'7' = 37_{16}	$000A_{16}$	FF_{16}
0003_{16}	'4' = 34_{16}	$000B_{16}$	FF_{16}
0004_{16}	'7' = 37_{16}	$000C_{16}$	FF_{16}
0005_{16}	'0' = 30_{16}	$000D_{16}$	FF_{16}
0006_{16}	'E' = 45_{16}	$000E_{16}$	FF_{16}
0007_{16}	'8' = 38_{16}	$000F_{16}$	FF_{16}

GZZ-SH02-97B<9YA0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37470E8-XXXSP
MITSUBISHI ELECTRIC

Recommend to writing the following pseudo-command to the start address of the assembler source program.

EPROM type	27256	27512
The pseudo-command	△*=△$8000 △ .BYTE△ 'M37470E8–'	△*=△$0000 △ .BYTE△ 'M37470E8–'

Note : If the name of the product written to the EPROMs does not match the name of the ROM programming confirmation, the ROM processing is disabled. Write the data correctly.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type of package being ordered. Please submit the shrink DIP package Mark Specification Form (only for built-in One Time PROM microcomputer).

✵ 3. Comments

(2/2)

GZZ-SH03-59B<06B0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37471E4-XXXSP/FP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain identical data, we will produce ROM programming based on this data. We shall assume the responsibility for errors only if the ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Microcomputer name : ☐ M37471E4-XXXSP ☐ M37471E4-XXXFP

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ 27128	☐ 27256	☐ 27512
EPROM address	EPROM address	EPROM address
0000_{16} Area for ASCII codes of the name of the product 'M37471E4–'	0000_{16} Area for ASCII codes of the name of the product 'M37471E4–'	0000_{16} Area for ASCII codes of the name of the product 'M37471E4–'
$000F_{16}$	$000F_{16}$	$000F_{16}$
0010_{16}	0010_{16}	0010_{16}
$1FFF_{16}$	$5FFF_{16}$	$DFFF_{16}$
2000_{16} ROM (8K)	6000_{16} ROM (8K)	$E000_{16}$ ROM (8K)
$3FFF_{16}$	$7FFF_{16}$	$FFFF_{16}$

(1) Set "FF_{16}" in the shaded area.
(2) Write the ASCII codes that indicates the name of the product 'M37471E4–' to addresses 0000_{16} to $000F_{16}$. ASCII codes 'M37471E4–' are listed on the right. The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = $4D_{16}$	0008_{16}	' – ' = $2D_{16}$
0001_{16}	'3' = 33_{16}	0009_{16}	FF_{16}
0002_{16}	'7' = 37_{16}	$000A_{16}$	FF_{16}
0003_{16}	'4' = 34_{16}	$000B_{16}$	FF_{16}
0004_{16}	'7' = 37_{16}	$000C_{16}$	FF_{16}
0005_{16}	'1' = 31_{16}	$000D_{16}$	FF_{16}
0006_{16}	'E' = 45_{16}	$000E_{16}$	FF_{16}
0007_{16}	'4' = 34_{16}	$000F_{16}$	FF_{16}

GZZ-SH03-59B<06B0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37471E4-XXXSP/FP
MITSUBISHI ELECTRIC

Recommend to writing the following pseudo-command to the start address of the assembler source program.

EPROM type	27128	27256	27512
The pseudo-command	△*=△$C000 △ .BYTE△ 'M37471E4–'	△*=△$8000 △ .BYTE△ 'M37471E4–'	△*=△$0000 △ .BYTE△ 'M37471E4–'

Note : If the name of the product written to the EPROMs does not match the name of the ROM programming confirmation, the ROM processing is disabled. Write the data correctly.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type of package being ordered. Please submit the shrink DIP package Mark Specification Form (only for built-in One Time PROM microcomputer) for M37471E4-XXXSP or the 56P6N Mark Specification Form the M37471E4-XXXFP.

✵ 3. Comments

(2/2)

GZZ-SH02-98B<9YB0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37471E8-XXXSP/FP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain identical data, we will produce ROM programming based on this data. We shall assume the responsibility for errors only if the ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Microcomputer name : ☐ M37471E8-XXXSP ☐ M37471E8-XXXFP

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ 27256	☐ 27512
EPROM address	EPROM address
0000_{16} Area for ASCII codes of the name of the product 'M37471E8–'	0000_{16} Area for ASCII codes of the name of the product 'M37471E8–'
$000F_{16}$	$000F_{16}$
0010_{16}	0010_{16}
$3FFF_{16}$	$BFFF_{16}$
4000_{16} ROM (16K)	$C000_{16}$ ROM (16K)
$7FFF_{16}$	$FFFF_{16}$

(1) Set "FF_{16}" in the shaded area.
(2) Write the ASCII codes that indicates the name of the product 'M37471E8–' to addresses 0000_{16} to $000F_{16}$. ASCII codes 'M37471E8–' are listed on the right. The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = $4D_{16}$	0008_{16}	' – ' = $2D_{16}$
0001_{16}	'3' = 33_{16}	0009_{16}	FF_{16}
0002_{16}	'7' = 37_{16}	$000A_{16}$	FF_{16}
0003_{16}	'4' = 34_{16}	$000B_{16}$	FF_{16}
0004_{16}	'7' = 37_{16}	$000C_{16}$	FF_{16}
0005_{16}	'1' = 31_{16}	$000D_{16}$	FF_{16}
0006_{16}	'E' = 45_{16}	$000E_{16}$	FF_{16}
0007_{16}	'8' = 38_{16}	$000F_{16}$	FF_{16}

GZZ-SH02-98B<9YB0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37471E8-XXXSP/FP
MITSUBISHI ELECTRIC

Recommend to writing the following pseudo-command to the start address of the assembler source program.

EPROM type	27256	27512
The pseudo-command	△*=△$8000 △ .BYTE△ 'M37471M8–'	△*=△$0000 △ .BYTE△ 'M37471M8–'

Note : If the name of the product written to the EPROMs does not match the name of the ROM programming confirmation, the ROM processing is disabled. Write the data correctly.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type of package being ordered. Please submit the shrink DIP package Mark Specification Form (only for built-in One Time PROM microcomputer) for the M37471E8-XXXSP or the 56P6N Mark Specification Form for the M37471E8-XXXFP.

✵ 3. Comments

(2/2)

APPENDIX

3.3 ROM programming ordering method

GZZ-SH06-79B<2XA1>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37477E8-XXXSP/FP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

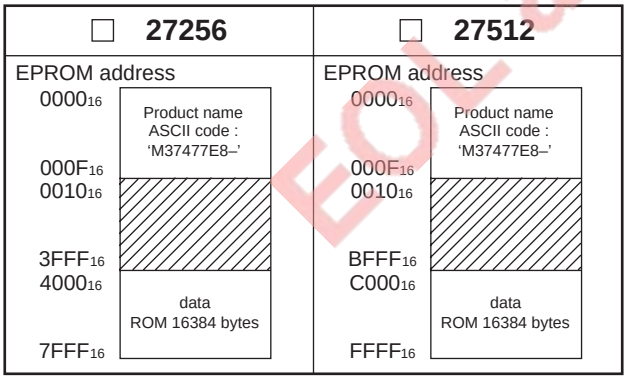
✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three EPROMs are required for each pattern.
If at least two of the three sets of EPROMs submitted contain identical data, we will produce ROM programming based on this data. We shall assume the responsibility for errors only if the ROM data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Microcomputer name : ☐ M37477E8-XXXSP ☐ M37477E8-XXXFP

Checksum code for entire EPROM ☐☐☐☐ (hexadecimal notation)

EPROM type (indicate the type used)



(1) Set the data in the unused area (the shaded area of the diagram) to "FF$_{16}$".
(2) The ASCII codes of the product name "M37477E8–" must be entered in addresses 0000$_{16}$ to 0008$_{16}$. And set the data "FF$_{16}$" in addresses 0009$_{16}$ to 000F$_{16}$. The ASCII codes and addresses are listed to the right in hexadecimal notation.

Address		Address	
0000$_{16}$	'M' = 4D$_{16}$	0008$_{16}$	' – ' = 2D$_{16}$
0001$_{16}$	'3' = 33$_{16}$	0009$_{16}$	FF$_{16}$
0002$_{16}$	'7' = 37$_{16}$	000A$_{16}$	FF$_{16}$
0003$_{16}$	'4' = 34$_{16}$	000B$_{16}$	FF$_{16}$
0004$_{16}$	'7' = 37$_{16}$	000C$_{16}$	FF$_{16}$
0005$_{16}$	'7' = 37$_{16}$	000D$_{16}$	FF$_{16}$
0006$_{16}$	'E' = 45$_{16}$	000E$_{16}$	FF$_{16}$
0007$_{16}$	'8' = 38$_{16}$	000F$_{16}$	FF$_{16}$

(1/2)

GZZ-SH06-79B<2XA1>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37477E8-XXXSP/FP
MITSUBISHI ELECTRIC

We recommend the use of the following pseudo-command to the start address of the assembler source program.

EPROM type	27256	27512
The pseudo-command	△*=△$8000 △ .BYTE△ 'M37477E8–'	△*=△$0000 △ .BYTE△ 'M37477E8–'

Note : If the name of the product written to the EPROMs does not match the name of the ROM programming confirmation, the ROM will not be processed.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type of package being ordered. Please submit the shrink DIP package Mark Specification Form (only for built-in One Time PROM microcomputer) for the M37477E8-XXXSP or the 32P2W Mark Specification Form for the M37477E8-XXXFP.

✵ 3. Comments

(2/2)

GZZ-SH06-81B<2XA0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37478E8-XXXSP/FP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

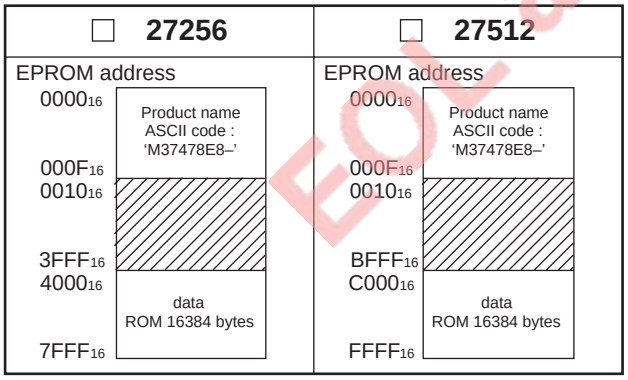
✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three EPROMs are required for each pattern.
If at least two of the three sets of EPROMs submitted contain identical data, we will produce ROM programming based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Microcomputer name : ☐ M37478E8-XXXSP ☐ M37478E8-XXXFP

Checksum code for entire EPROM ☐☐☐☐ (hexadecimal notation)

EPROM type (indicate the type used)



(1) Set the data in the unused area (the shaded area of the diagram) to "FF$_{16}$".
(2) The ASCII codes of the product name "M37478E8–" must be entered in addresses 0000$_{16}$ to 0008$_{16}$. And set the data "FF$_{16}$" in addresses 0009$_{16}$ to 000F$_{16}$. The ASCII codes and addresses are listed to the right in hexadecimal notation.

Address		Address	
0000$_{16}$	'M' = 4D$_{16}$	0008$_{16}$	' – ' = 2D$_{16}$
0001$_{16}$	'3' = 33$_{16}$	0009$_{16}$	FF$_{16}$
0002$_{16}$	'7' = 37$_{16}$	000A$_{16}$	FF$_{16}$
0003$_{16}$	'4' = 34$_{16}$	000B$_{16}$	FF$_{16}$
0004$_{16}$	'7' = 37$_{16}$	000C$_{16}$	FF$_{16}$
0005$_{16}$	'8' = 38$_{16}$	000D$_{16}$	FF$_{16}$
0006$_{16}$	'E' = 45$_{16}$	000E$_{16}$	FF$_{16}$
0007$_{16}$	'8' = 38$_{16}$	000F$_{16}$	FF$_{16}$

GZZ-SH06-81B<2XA0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37478E8-XXXSP/FP
MITSUBISHI ELECTRIC

We recommend the use of the following pseudo-command to the start address of the assembler source program.

EPROM type	27256	27512
The pseudo-command	△*=△$8000 △ .BYTE△ 'M37478E8–'	△*=△$0000 △ .BYTE△ 'M37478E8–'

Note : If the name of the product written to the EPROMs does not match the name of the ROM programming confirmation form, the ROM will not be processed.

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type of package being ordered. Please submit the shrink DIP package Mark Specification Form (only for built-in One Time PROM microcomputer) for the M37478E8-XXXSP or the 56P6N Mark Specification Form for the M37478E8-XXXFP.

✵ 3. Comments

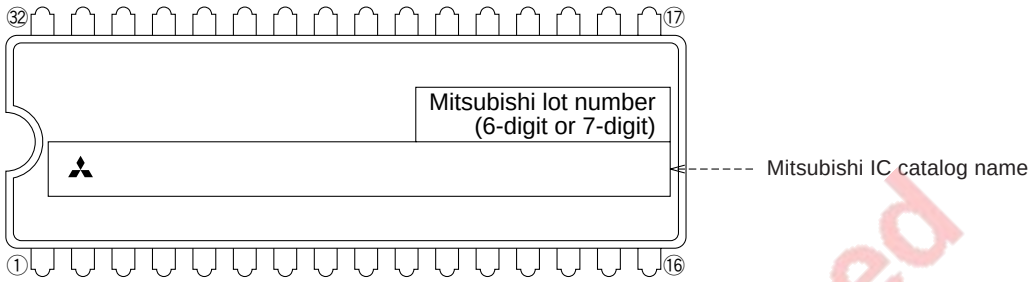
3.4 Mark specification form

32P4B (32-PIN SHRINK DIP) MARK SPECIFICATION FORM

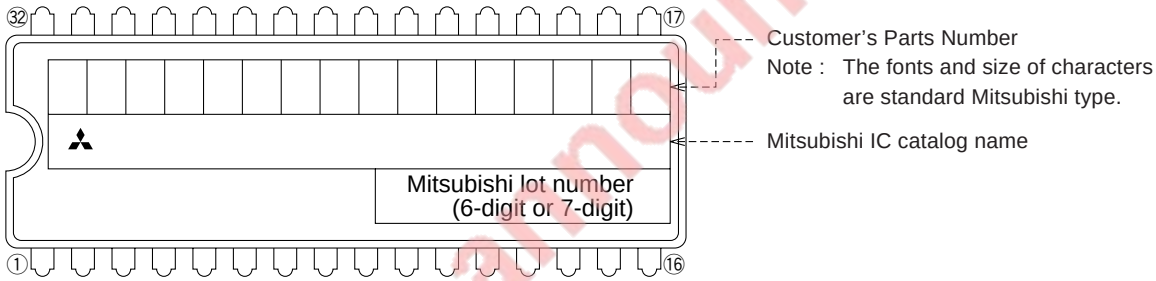
Mitsubishi IC catalog name ☐

Please choose one of the marking types below (A, B, C), and enter the Mitsubishi IC catalog name and the special mark (if needed).

A. Standard Mitsubishi Mark



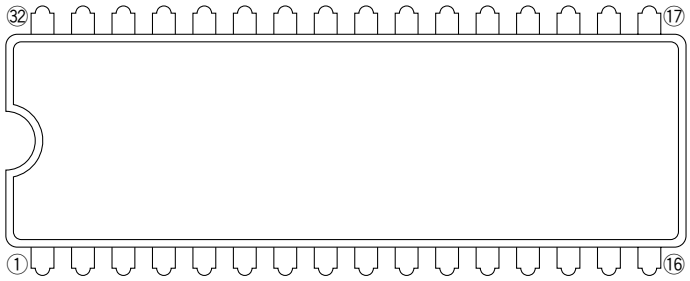
B. Customer's Parts Number + Mitsubishi catalog name



Note1 : The mark field should be written right aligned.
2 : The fonts and size of characters are standard Mitsubishi type.
3 : Customer's Parts Number can be up to 16 characters : Only 0 ~ 9, A ~ Z, +, –, /, (,), &, ©, . (periods), and , (commas) are usable.
4 : If the Mitsubishi logo is not required, check the box on the right.

Mitsubishi logo is not required ☐

C. Special Mark Required



Note1 : If the Special Mark is to be Printed, indicate the desired layout of the mark in the upper figure. The layout will be duplicated as close as possible. Mitsubishi lot number (6-digit or 7-digit) and Mask ROM number (3-digit) are always marked.
2 : If the customer's trade mark logo must be used in the Special Mark, check the box on the right. Please submit a clean original of the logo. For the new special character fonts a clean font original (ideally logo drawing) must be submitted.
3 : The standard Mitsubishi font is used for all characters except for a logo.

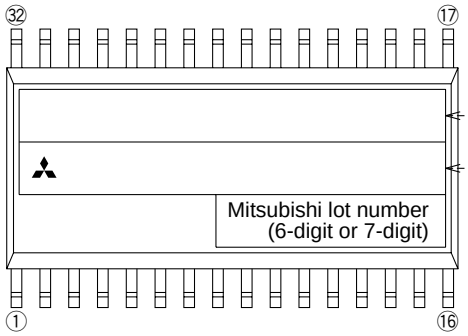
Special logo required ☐

32P2W-A (32-PIN SOP) MARK SPECIFICATION FORM

Mitsubishi IC catalog name

Please choose one of the marking types below (A, B, C), and enter the Mitsubishi catalog name and the special mark (if needed).

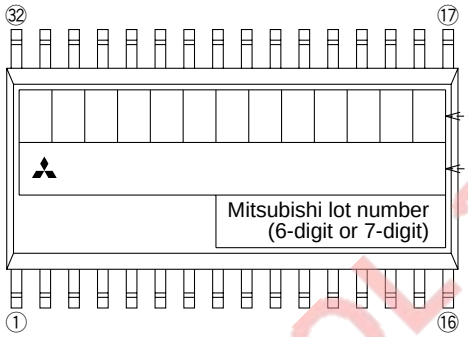
A. Standard Mitsubishi Mark



Mitsubishi IC catalog name

Mitsubishi IC catalog name

B. Customer's Parts Number + Mitsubishi catalog name



Customer's Parts Number

Note : The fonts and size of characters are standard Mitsubishi type.

Mitsubishi IC catalog name

Note1 : The mark field should be written right aligned.

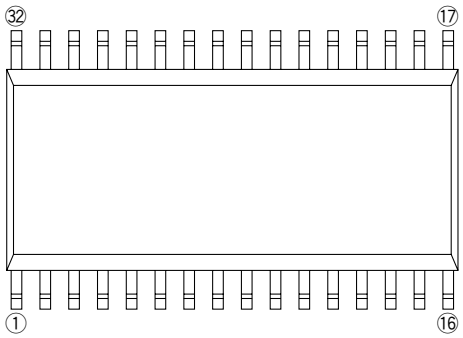
2 : The fonts and size of characters are standard Mitsubishi type.

3 : Customer's Parts Number can be up to 13 characters : Only 0 ~ 9, A ~ Z, +, –, /, (,), &, ©, . (periods), , (commas) are usable.

4 : If the Mitsubishi logo is not required, check the box below.

Mitsubishi logo is not required ☐

C. Special Mark Required



Note1 : If the Special Mark is to be Printed, indicate the desired layout of the mark in the left figure. The layout will be duplicated as close as possible.
Mitsubishi lot number (6-digit or 7-digit) and Mask ROM number (3-digit) are always marked.

2 : If the customer's trade mark logo must be used in the Special Mark, check the box below.
Please submit a clean original of the logo.
For the new special character fonts a clean font original (ideally logo drawing) must be submitted.

Special logo required ☐

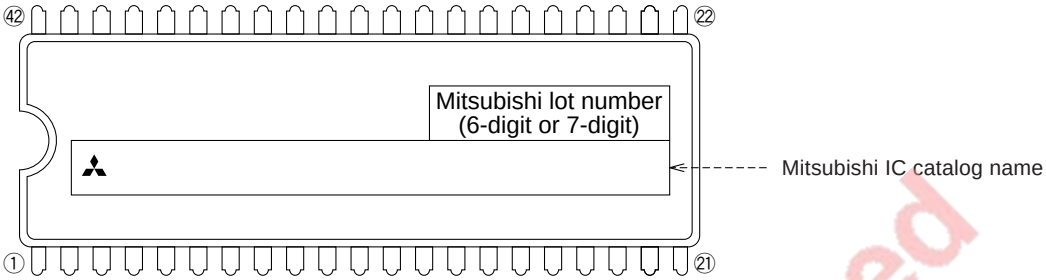
3 : The standard Mitsubishi font is used for all characters except for a logo.

42P4B (42-PIN SHRINK DIP) MARK SPECIFICATION FORM

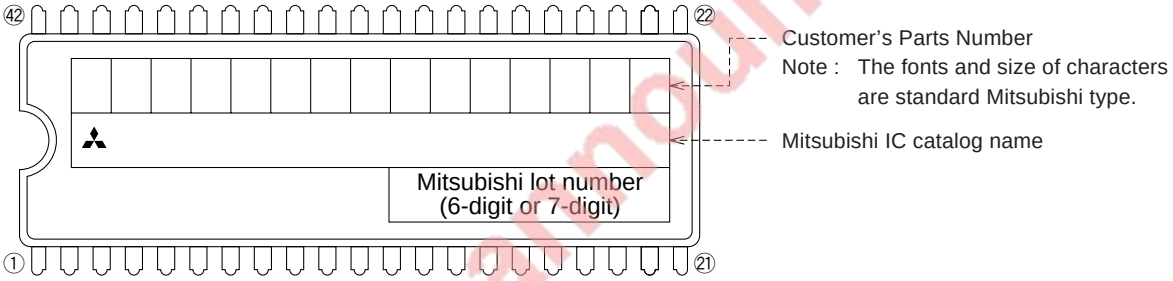
Mitsubishi IC catalog name

Please choose one of the marking types below (A, B, C), and enter the Mitsubishi IC catalog name and the special mark (if needed).

A. Standard Mitsubishi Mark



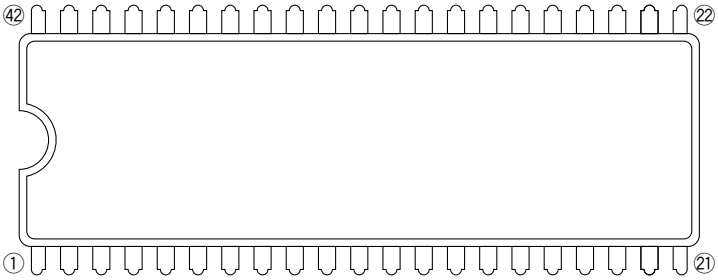
B. Customer's Parts Number + Mitsubishi catalog name



Note1 : The mark field should be written right aligned.
2 : The fonts and size of characters are standard Mitsubishi type.
3 : Customer's Parts Number can be up to 15 characters : Only 0 ~ 9, A ~ Z, +, –, /, (,), &, ©, . (periods), and , (commas) are usable.
4 : If the Mitsubishi logo is not required, check the box on the right.

Mitsubishi logo is not required ☐

C. Special Mark Required



Note1 : If the Special Mark is to be Printed, indicate the desired layout of the mark in the upper figure. The layout will be duplicated as close as possible. Mitsubishi lot number (6-digit or 7-digit) and Mask ROM number (3-digit) are always marked.
2 : If the customer's trade mark logo must be used in the Special Mark, check the box on the right. Please submit a clean original of the logo. For the new special character fonts a clean font original (ideally logo drawing) must be submitted.
3 : The standard Mitsubishi font is used for all characters except for a logo.

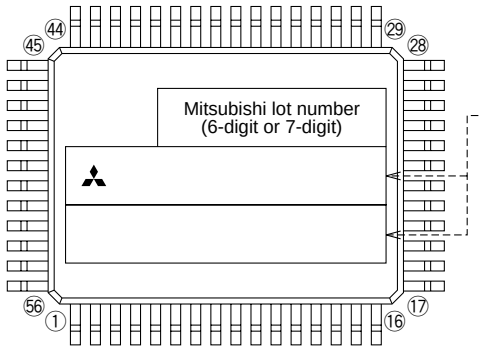
Special logo required ☐

56P6N-A (56-PIN QFP) MARK SPECIFICATION FORM

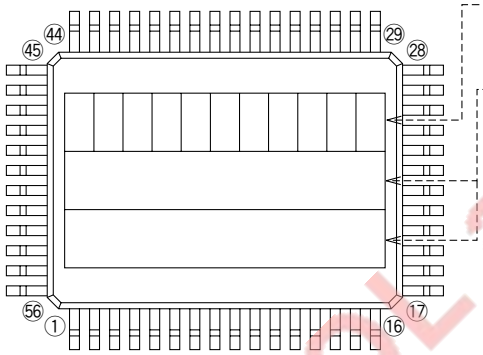
Mitsubishi IC catalog name

Please choose one of the marking types below (A, B, C), and enter the Mitsubishi IC catalog name and the special mark (if needed).

A. Standard Mitsubishi Mark



B. Customer's Parts Number + Mitsubishi IC catalog name



Note : The fonts and size of characters are standard Mitsubishi type.

Note1 : The mark field should be written right aligned.

2 : The fonts and size of characters are standard Mitsubishi type.

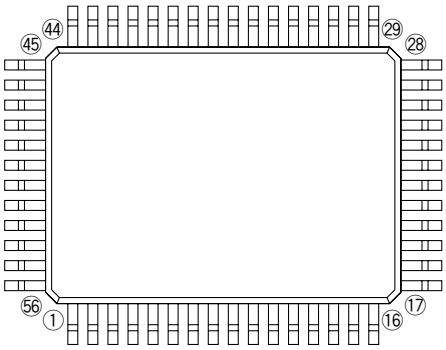
3 : Customer's Parts Number can be up to 11 characters : Only 0 ~ 9, A ~ Z, +, –, /, (,), &, ©, . (period), and , (comma) are usable.

4 : If the Mitsubishi logo is not required, check the box below.

Mitsubishi logo is not required ☐

5 : Arrangement of Mitsubishi IC catalog name and Mitsubishi lot number is dependent on number of Mitsubishi IC catalog name and that Mitsubishi logo is required or not.

C. Special Mark Required



Note1 : If the Special Mark is to be Printed, indicate the desired layout of the mark in the left figure. The layout will be duplicated as close as possible.
Mitsubishi lot number (6-digit or 7-digit) and Mask ROM number (3-digit) are always marked.

2 : If the customer's trade mark logo must be used in the Special Mark, check the box below.
Please submit a clean original of the logo.
For the new special character fonts a clean font original (ideally logo drawing) must be submitted.

Special logo required ☐

3 : The standard Mitsubishi font is used for all characters except for a logo.

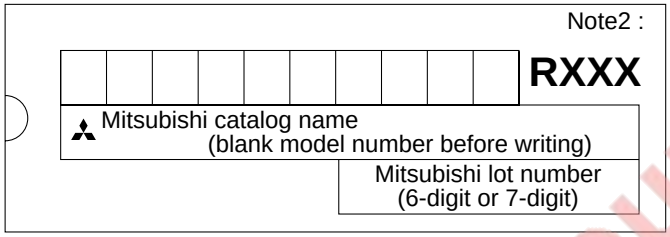
SHRINK DIP MARK SPECIFICATION FORM
for One Time PROM version microcomputers

Enter the catalog number of the microcomputer for which this mark specification is intended. (If you do not know the ROM code number, enter XXX in its place.)

The catalog number of the microcomputer **M**

A. Standard Mitsubishi Mark

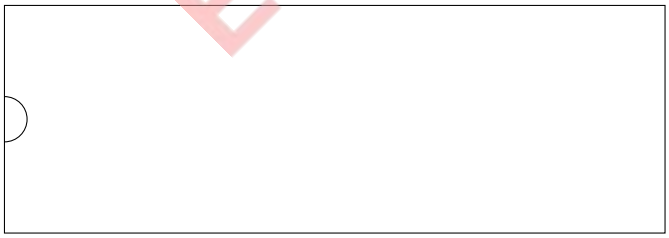
Customer specified part number will be printed together with the ROM code number on the top line.
Enter the desired part number left aligned in the box below. (up to 10 characters)



Note1 : The following characters can be used in the part number :
Uppercase alphabet, numbers, ampersand, hyphen, period, comma, +, /, (,), ©
(© will be printed at 1.5 x character width)
2 : XXX is the ROM code number.

B. Special Mark Required

If you desire anything other than the standard Mitsubishi mark, it will be treated as a special mark.
Special marks will take longer to produce and should be avoided if possible.
If a special mark is to be printed, indicate the desired layout of the mark in the figure below. The layout will be duplicated as closely as possible.



Note1 : If the customer's trademark logo must be used in the Special Mark, please submit a clean original logo.
Note that special marks require extra cost and time to produce.

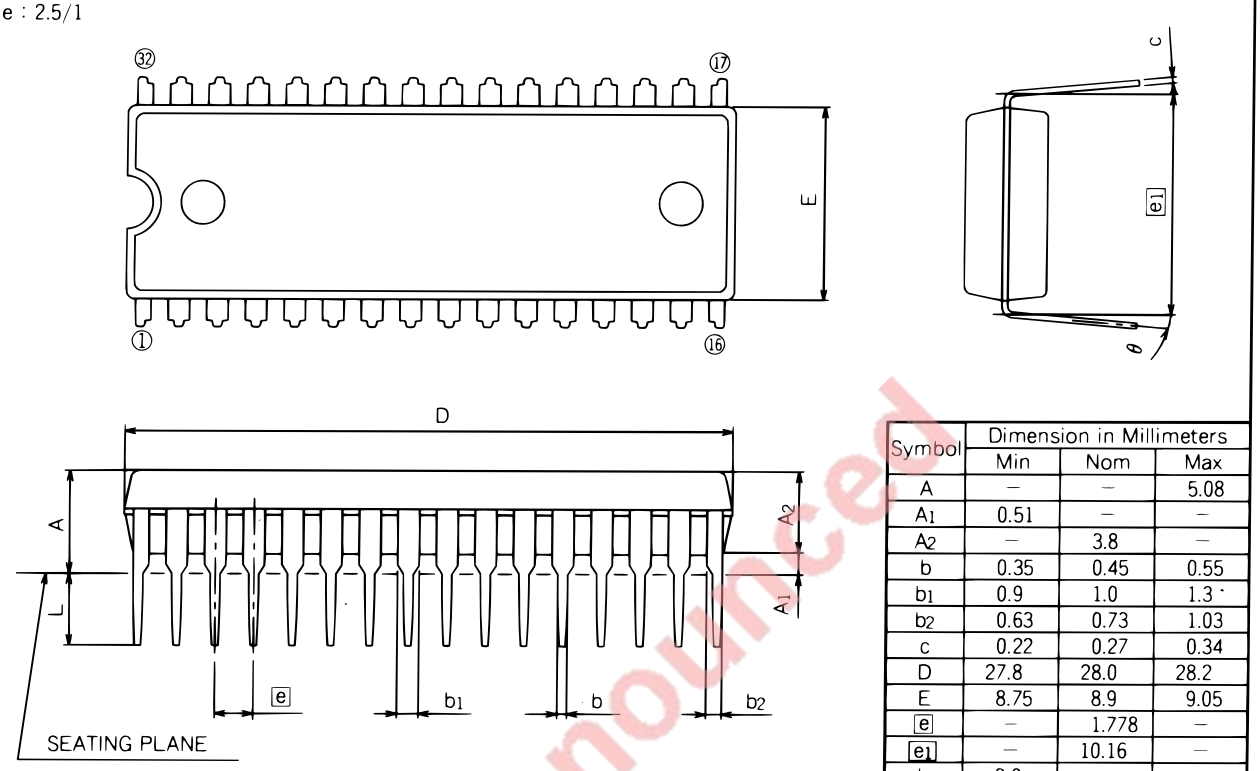
3.5 Package outline

32P4B

Plastic 32pin 400mil SDIP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
SDIP32-P-400-1.78	–	2.2	Alloy 42/Cu Alloy

Scale : 2.5/1



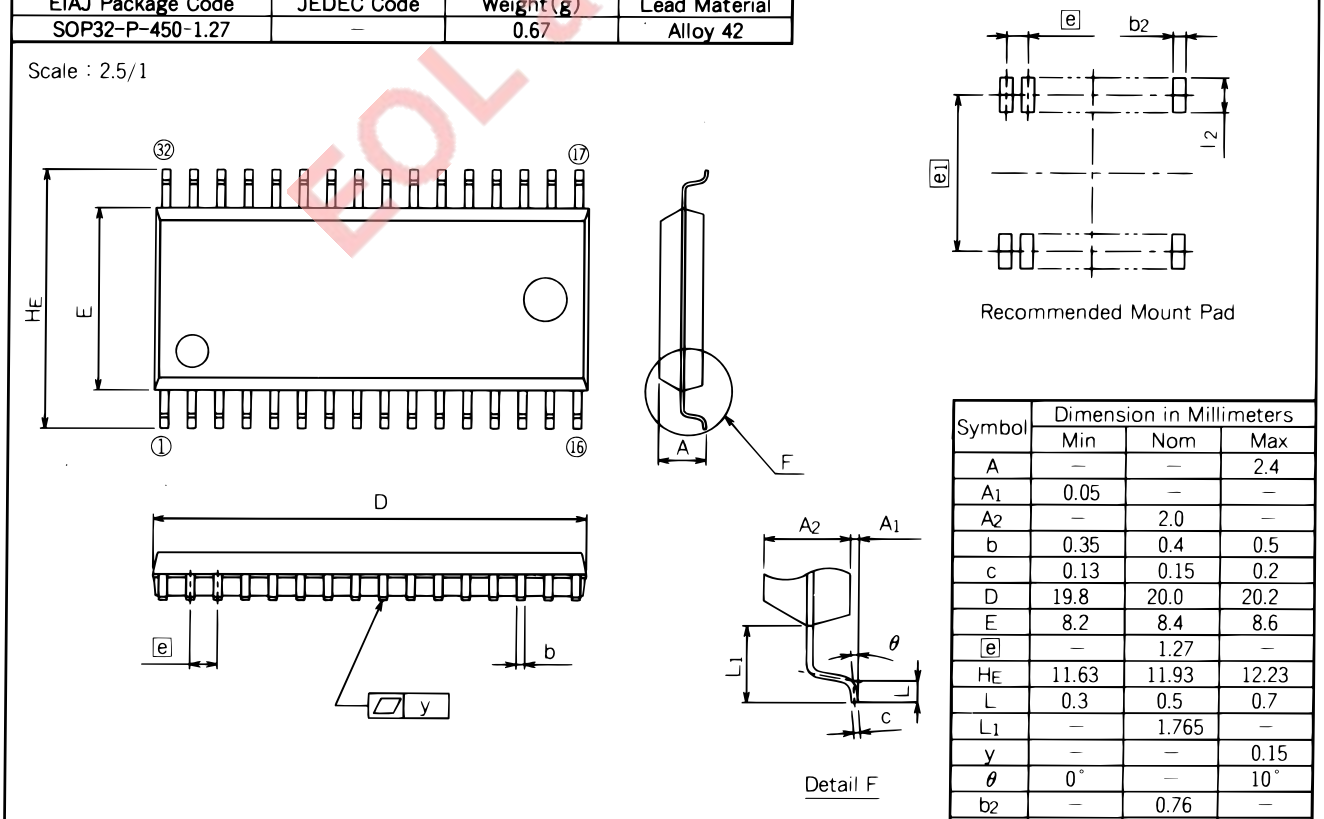
Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	–	–	5.08
A_1	0.51	–	–
A_2	–	3.8	–
b	0.35	0.45	0.55
b_1	0.9	1.0	1.3
b_2	0.63	0.73	1.03
c	0.22	0.27	0.34
D	27.8	28.0	28.2
E	8.75	8.9	9.05
e	–	1.778	–
e_1	–	10.16	–
L	3.0	–	–
θ	0°	–	15°

32P2W-A

Plastic 32pin 450mil SOP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
SOP32-P-450-1.27	–	0.67	Alloy 42

Scale : 2.5/1



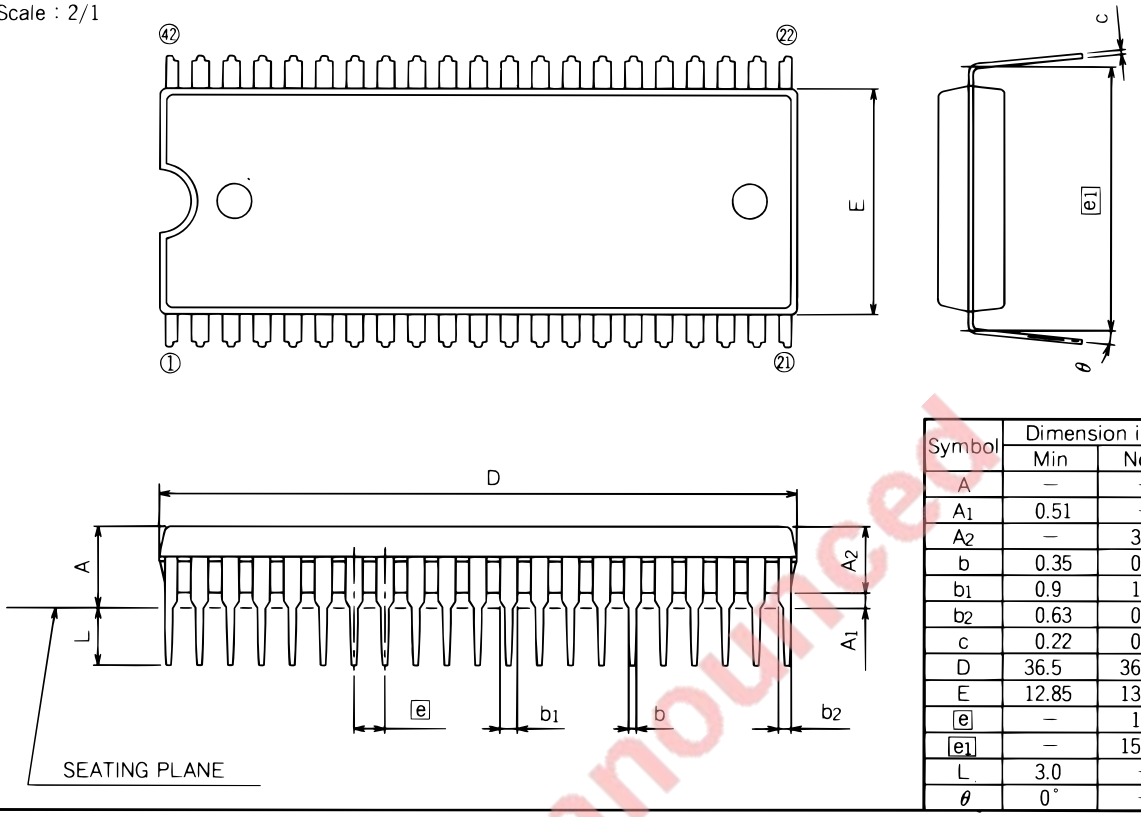
Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	–	–	2.4
A_1	0.05	–	–
A_2	–	2.0	–
b	0.35	0.4	0.5
c	0.13	0.15	0.2
D	19.8	20.0	20.2
E	8.2	8.4	8.6
e	–	1.27	–
H_E	11.63	11.93	12.23
L	0.3	0.5	0.7
L_1	–	1.765	–
y	–	–	0.15
θ	0°	–	10°
b_2	–	0.76	–
e_1	–	11.43	–
l_2	1.27	–	–

42P4B

Plastic 42pin 600mil SDIP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
SDIP42-P-600-1.78	–	4.1	Alloy 42/Cu Alloy

Scale : 2/1



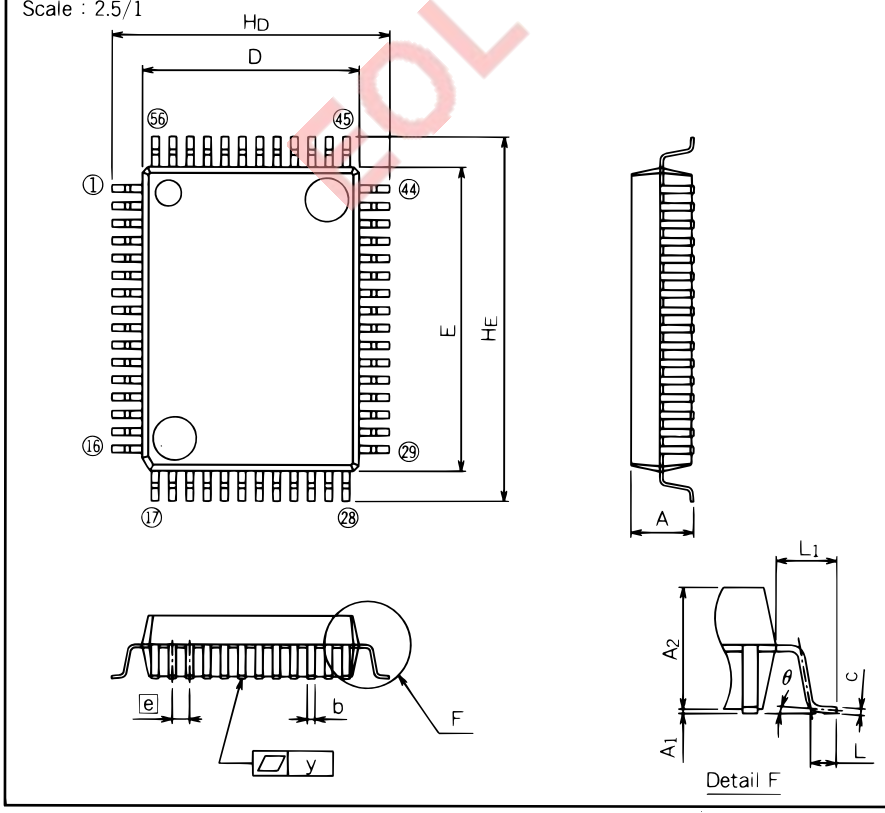
Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	–	–	5.5
A_1	0.51	–	–
A_2	–	3.8	–
b	0.35	0.45	0.55
b_1	0.9	1.0	1.3
b_2	0.63	0.73	1.03
c	0.22	0.27	0.34
D	36.5	36.7	36.9
E	12.85	13.0	13.15
e	–	1.778	–
e1	–	15.24	–
L	3.0	–	–
θ	0°	–	15°

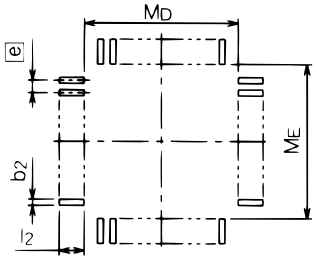
56P6N–A

Plastic 56pin 10×14mm body QFP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
QFP56-P-1014-0.80	–	0.79	Alloy 42

Scale : 2.5/1



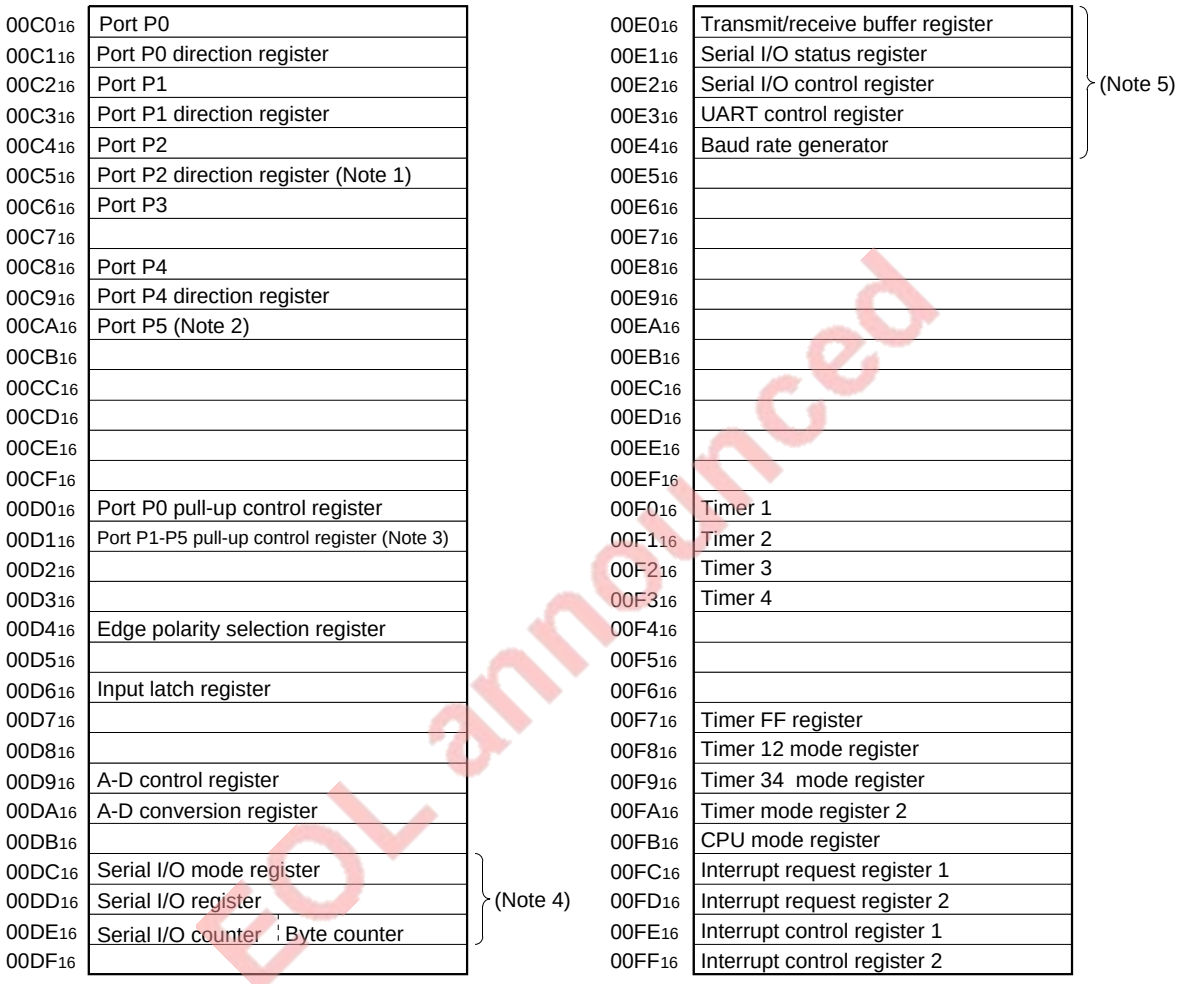


Recommended Mount Pad

Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	–	–	3.05
A_1	0	0.1	0.2
A_2	–	2.8	–
b	0.3	0.35	0.45
c	0.13	0.15	0.2
D	9.8	10.0	10.2
E	13.8	14.0	14.2
e	–	0.8	–
HD	12.5	12.8	13.1
HE	16.5	16.8	17.1
L	0.4	0.6	0.8
L_1	–	1.4	–
y	–	–	0.1
θ	0°	–	10°
b_2	–	0.5	–
l_2	1.3	–	–
MD	–	10.6	–
ME	–	14.6	–

3.6 SFR memory map

Figure 3.6.1 shows the special function register (SFR) memory map.

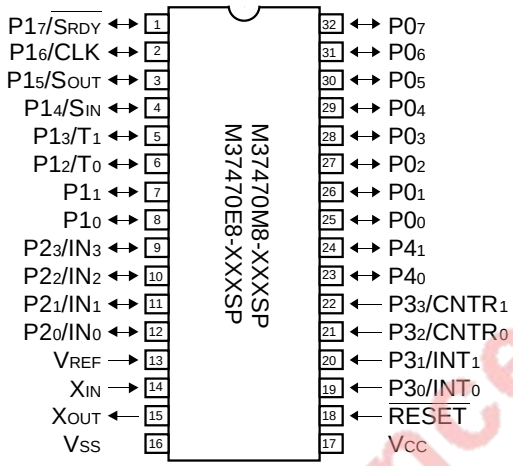


Notes 1: In the 7477/7478 group, this register is not located.
2: In the 7470/7477 group, this register is not located.
3: This address is allocated P1-P4 pull-up control register for the 7470/7477 group.
4: In the 7477/7478 group, this register is not located.
5: In the 7470/7471 group, this register is not located.

Fig. 3.6.1 SFR memory map

3.7 Pin configuration

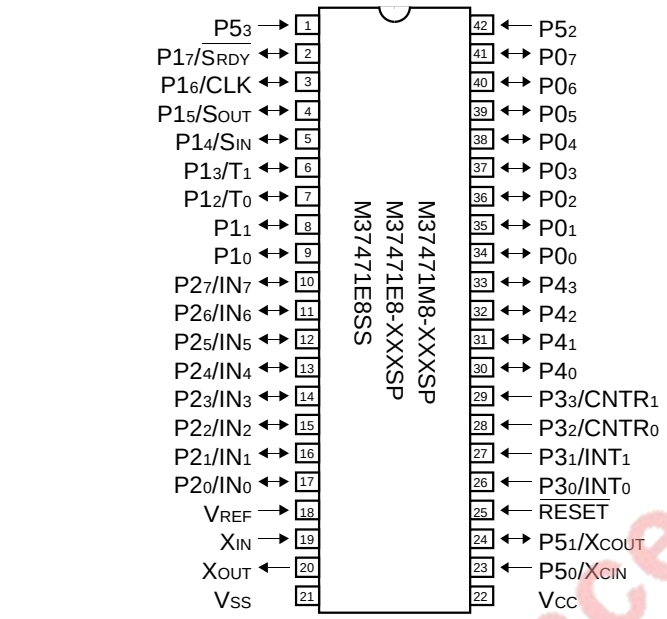
Figures 3.7.1 to 3.7.4 show the pin configuration of 7470/7471/7477/7478 group.



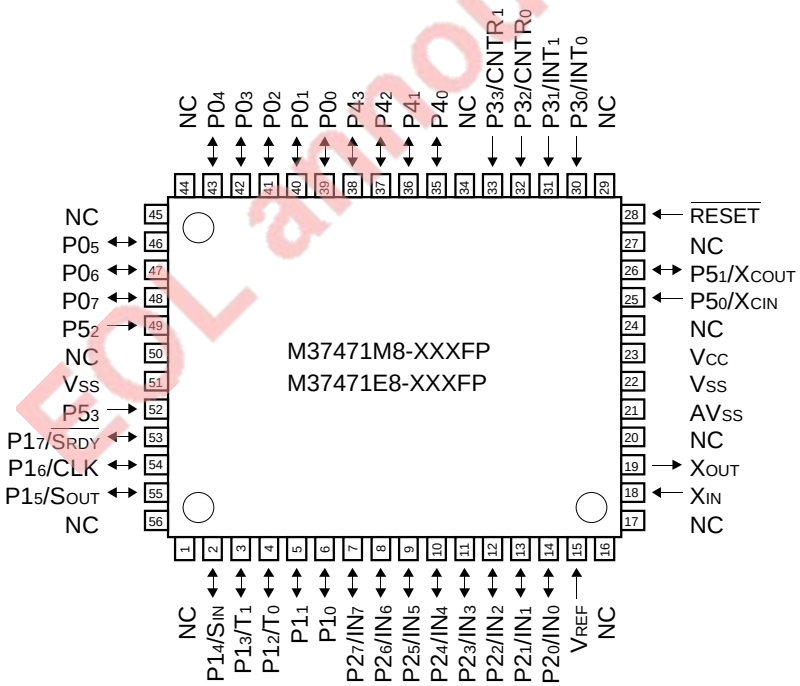
Outline 32P4B (Note)

Note: The M37470M2-XXXSP and M37470M4/E4-XXXSP are included in the 32P4B package. All of these products are pin-compatible.

Fig. 3.7.1 Pin configuration of 7470 group



Outline 42P4B (Note 1)
42S1B-A (M37471E8SS)



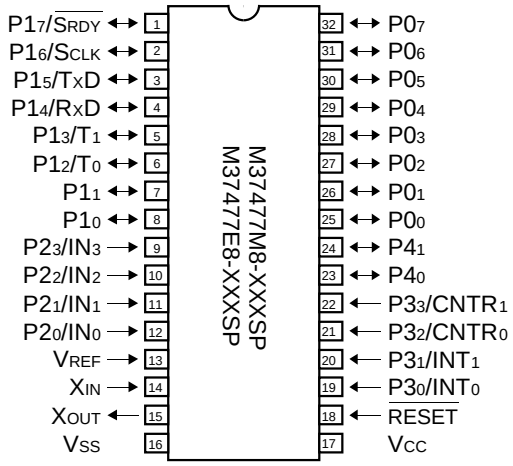
Outline 56P6N-A (Note 2)

NC: No connection

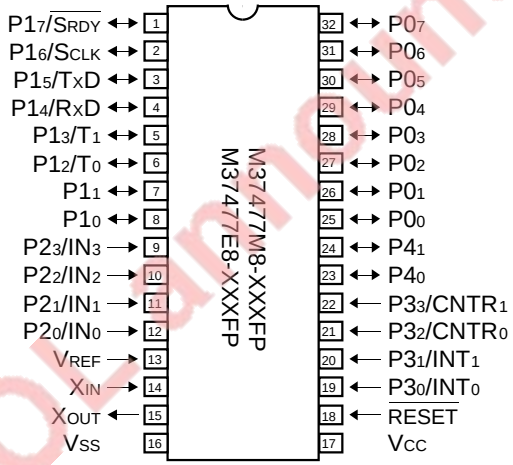
Notes 1 : The M37471M2-XXXSP and M37471M4/E4-XXXSP are included in the 42P4B package. All of these products are pin-compatible.
2 : The M37471M2-XXXFP and M37471M4/E4-XXXFP are included in the 56P6N-A package. All of these products are pin-compatible.
3 : The only differences between the 42P4B package product and the 56P6N-A package product are package shape, absolute maximum ratings and the fact that the 56P6N-A package product has an AVSS pin.

Fig. 3.7.2 Pin configuration of 7471 group

3.7 Pin configuration



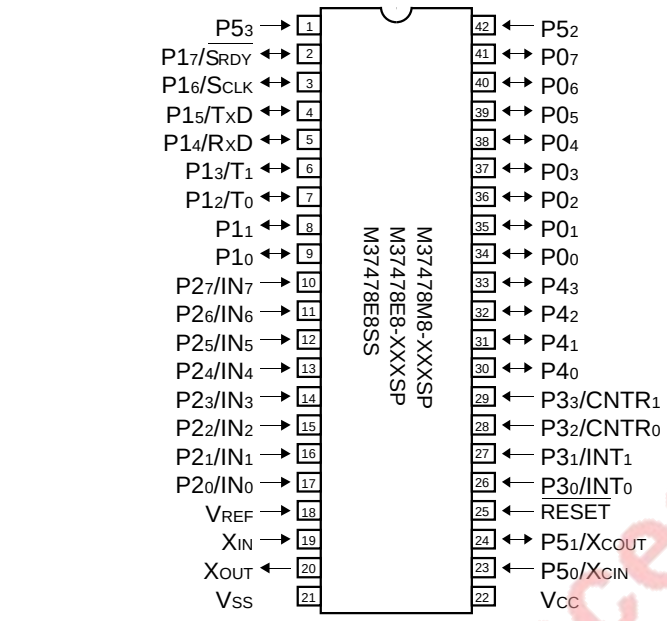
Outline 32P4B (Note 1)



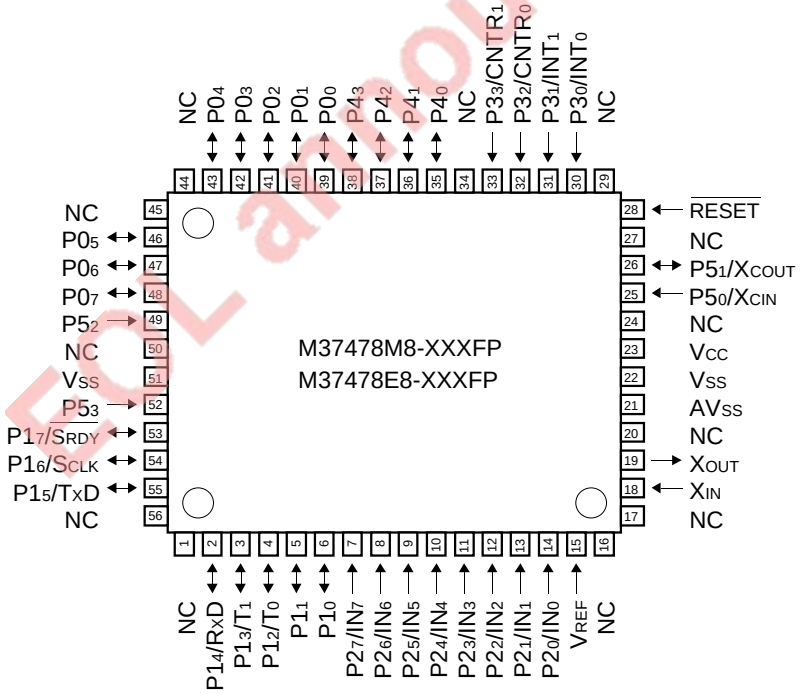
Outline 32P2W-A (Note 2)

Notes 1 : The M37477M4-XXXSP is included in the 32P4B package.
These products are pin-compatible.
2 : The M37477M4-XXXFP is included in the 32P2W-A package.
These products are pin-compatible.
3 : The only differences between the 32P4B package product and the 32P2W-A package product are package shape and absolute maximum ratings.

Fig. 3.7.3 Pin configuration of 7477 group



Outline 42P4B (Note 1)
42S1B-A (M37478E8SS)



Outline 56P6N-A (Note 2)

NC: No connection

Notes 1 : The M37478M4-XXXSP is included in the 42P4B package.
These products are pin-compatible
2 : The M37478M4-XXXFP is included in the 56P6N-A package.
These products are pin-compatible
3 : The only differences between the 42P4B package product and the 56P6N-A package product are package shape, absolute maximum ratings and the fact that the 56P6N-A package product has an AV_{SS} pin.

Fig. 3.7.4 Pin configuration of 7478 group

MITSUBISHI SEMICONDUCTORS
USER'S MANUAL
7470/7471/7477/7478 Group

Jan. First Edition 1998

Editioned by
Committee of editing of Mitsubishi Semiconductor USER'S MANUAL

Published by
Mitsubishi Electric Corp., Semiconductor Marketing Division

User's Manual
7470/7471/7477/7478 Group

Renesas Technology Corp.
Nippon Bldg.,6-2,Otemachi 2-chome,Chiyoda-ku,Tokyo,100-0004 Japan

New publication, effective Jan. 1998.
Specifications subject to change without notice.

REVISION DESCRIPTION LIST	7470/71, 7477/78 USER'S MANUAL

Rev. No.	Revision Description	Rev. date
1.0	First Edition	980131