MITSUBISHI 8-BIT SINGLE-CHIP MICROCOMPUTER
740 FAMILY / 38000 SERIES

38B5 Group

User's Manual

MITSUBISHI
ELECTRIC

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 38B5 Group.
After reading this manual, the user should have a through knowledge of the functions and features of the 38B5 Group, and should be able to fully utilize the product. The manual starts with specifications and ends with application examples.
For details of software, refer to the "740 Family Software Manual."
For details of development support tools, refer to the "DEVELOPMENT SUPPORT TOOLS FOR MICROCOMPUTERS" data book.

BEFORE USING THIS USER'S MANUAL

This user's manual consists of the following three chapters. Refer to the chapter appropriate to your conditions, such as hardware design or software development.

1. Organization

● **CHAPTER 1 HARDWARE**

This chapter describes features of the microcomputer and operation of each peripheral function.

● **CHAPTER 2 APPLICATION**

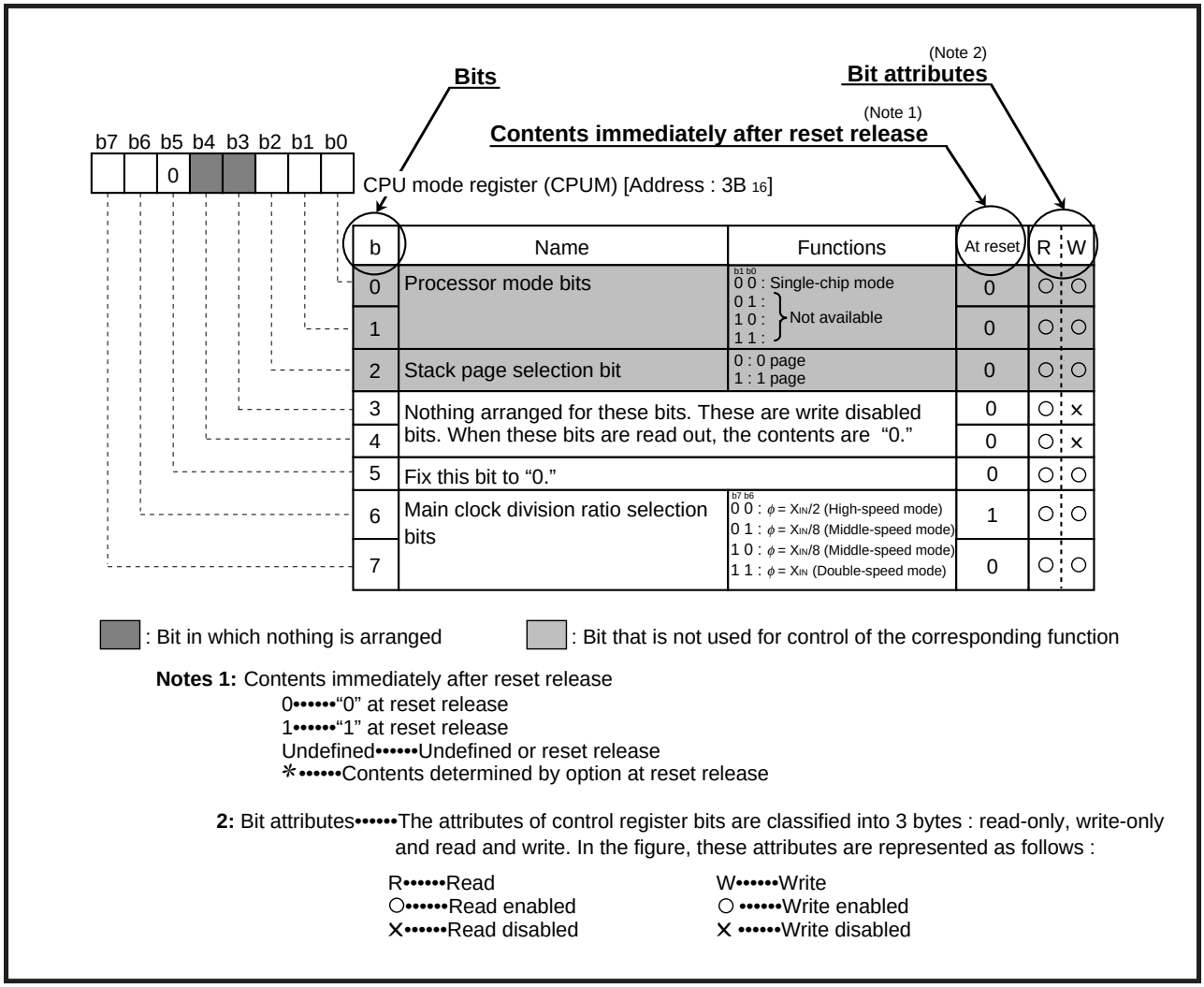
This chapter describes usage and application examples of peripheral functions, based mainly on setting examples of relevant registers.

● **CHAPTER 3 APPENDIX**

This chapter includes a list of registers, and necessary information for systems development using the microcomputer, the mask ROM confirmation (for mask ROM version), ROM programming confirmation, and the 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:



Notes 1: Contents immediately after reset release

0••••••"0" at reset release
1••••••"1" at reset release
Undefined••••••Undefined or reset release
✻••••••Contents determined by option at reset release

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

R••••••Read
○••••••Read enabled
×••••••Read disabled
W••••••Write
○ ••••••Write enabled
× ••••••Write disabled

Table of contents

CHAPTER 1 HARDWARE

CHAPTER 2 APPLICATION

CHAPTER 3 APPENDIX

List of figures

CHAPTER 1 HARDWARE

List of figures

CHAPTER 2 APPLICATION

List of figures

CHAPTER 3 APPENDIX

List of figures

List of tables

CHAPTER 1 HARDWARE

CHAPTER 2 APPLICATION

CHAPTER 3 APPENDIX

CHAPTER 1
HARDWARE

HARDWARE

DESCRIPTION/FEATURES/APPLICATION/PIN CONFIGURATION

DESCRIPTION

The 38B5 group is the 8-bit microcomputer based on the 740 family core technology.

The 38B5 group has six 8-bit timers, a 16-bit timer, a fluorescent display automatic display circuit, 12-channel 10-bit A-D converter, a serial I/O with automatic transfer function, which are available for controlling musical instruments and household appliances.

The 38B5 group has variations of internal memory size and packaging. For details, refer to the section on part numbering.

For details on availability of microcomputers in the 38B5 group, refer to the section on group expansion.

Built-in pull-down resistors connected to high-breakdown voltage ports are available by specifying with the mask option in some products. For the details, refer to the section on the mask option of pull-down resistor.

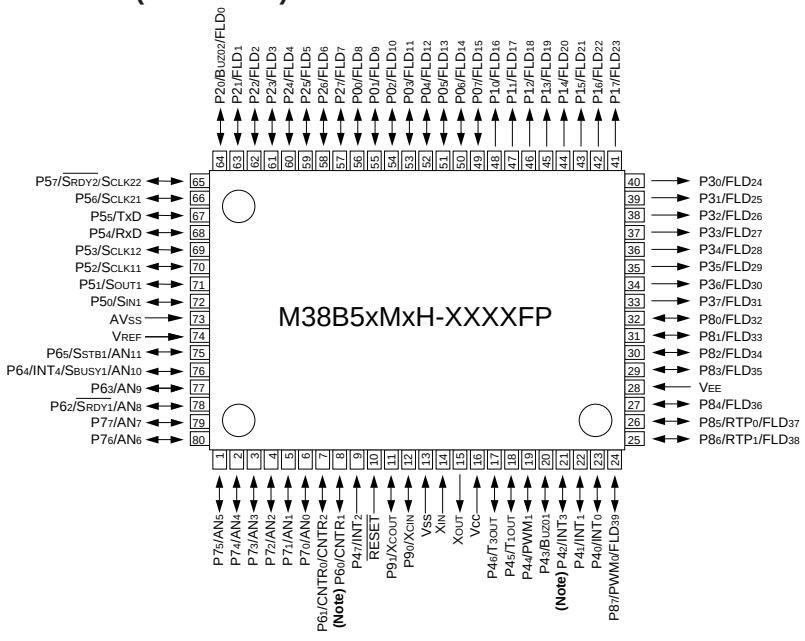
FEATURES

- Basic machine-language instructions 71
- The minimum instruction execution time ..... 0.48 μs
  (at 4.19 MHz oscillation frequency)
- Memory size
  ROM ..... 24K to 60K bytes
  RAM ..... 1024 to 2048 bytes
- Programmable input/output ports 55
- High-breakdown-voltage output ports 36
- Software pull-up resistors ..... (Ports P5, P6$_1$ to P6$_5$, P7, P8$_4$ to P8$_7$, P9)
- Interrupts ..... 21 sources, 16 vectors
- Timers ..... 8-bit × 6, 16-bit × 1
- Serial I/O1 (Clock-synchronized) ..... 8-bit × 1
  ..... (max. 256-byte automatic transfer function)
- Serial I/O2 (UART or Clock-synchronized) ..... 8-bit × 1
- PWM ..... 14-bit × 1
  8-bit × 1 (also functions as timer 6)
- A-D converter ..... 10-bit × 12 channels
- Fluorescent display function ..... Total 40 control pins
- Interrupt interval determination function 1
- Watchdog timer ..... 20-bit × 1
- Buzzer output 1
- 2 Clock generating circuit
  Main clock (X_{IN}–X_{OUT}) ..... Internal feedback resistor
  Sub-clock (X_{CIN}–X_{COUT}) ..... Without internal feedback resistor
  (connect to external ceramic resonator or quartz-crystal oscillator)
- Power source voltage
  In high-speed mode ..... 4.0 to 5.5 V
  (at 4.19 MHz oscillation frequency and high-speed selected)
  In middle-speed mode ..... 2.7 to 5.5 V
  (at 4.19 MHz oscillation frequency and middle-speed selected)
  In low-speed mode ..... 2.7 to 5.5 V
  (at 32 kHz oscillation frequency)
- Power dissipation
  In high-speed mode ..... 35 mW
  (at 4.19 MHz oscillation frequency)
  In low-speed mode ..... 60 μW
  (at 32 kHz oscillation frequency, at 3 V power source voltage)
- Operating temperature range ..... –20 to 85 °C

APPLICATION

Musical instruments, VCR, household appliances, etc.

PIN CONFIGURATION (TOP VIEW)



Note: In the mask option type P, INT$_3$ and CNTR$_1$ cannot be used.

Package type : 80P6N-A
80-pin plastic-molded QFP

Fig. 1 Pin configuration of M38B5xMxH-XXXXFP

FUNCTIONAL BLOCK

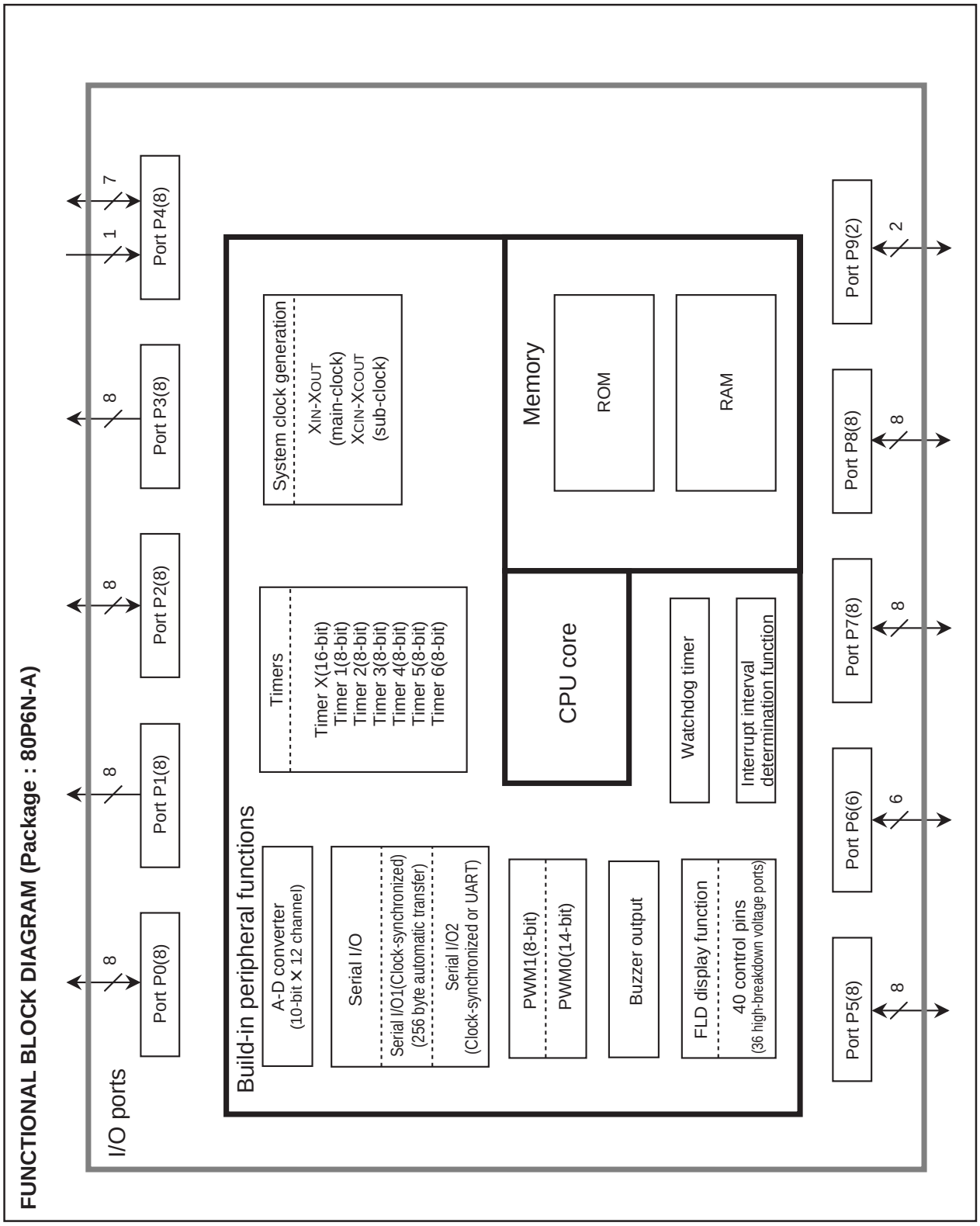


Fig. 2 Functional block diagram

PIN DESCRIPTION

Table 1 Pin description (1)

Pin	Name	Function	Function except a port function
V_{CC}, V_{SS}	Power source	• Apply voltage of 4.0–5.5 V to V_{CC}, and 0 V to V_{SS}.	
V_{EE}	Pull-down power source	• Apply voltage supplied to pull-down resistors of ports P0, P1, and P3.	
V_{REF}	Reference voltage	• Reference voltage input pin for A-D converter.	
AVSS	Analog power source	• Analog power source input pin for A-D converter. • Connect to V_{SS}.	
RESET	Reset input	• Reset input pin for active "L."	
X_{IN}	Clock input	• Input and output pins for the main clock generating circuit. • Feedback resistor is built in between X_{IN} pin and X_{OUT} pin. • Connect a ceramic resonator or quartz-crystal oscillator between the X_{IN} and X_{OUT} pins to set the oscillation frequency. • When an external clock is used, connect the clock source to the X_{IN} pin and leave the X_{OUT} pin open. • The clock is used as the oscillating source of system clock.	
X_{OUT}	Clock output		
$P0_0$/FLD_8– $P0_7$/FLD_{15}	I/O port P0	• 8-bit I/O port. • I/O direction register allows each pin to be individually programmed as either input or output. • At reset, this port is set to input mode. • A pull-down resistor is built in between port P0 and the V_{EE} pin. • CMOS compatible input level. • High-breakdown-voltage P-channel open-drain output structure. • At reset, this port is set to V_{EE} level.	• FLD automatic display pins
$P1_0$/FLD_{16}– $P1_7$/FLD_{23}	Output port P1	• 8-bit output port. • A pull-down resistor is built in between port P1 and the V_{EE} pin. • High-breakdown-voltage P-channel open-drain output structure. • At reset, this port is set to V_{EE} level.	• FLD automatic display pins
$P2_0$/BUZ_{02}/ FLD_0– $P2_7$/FLD_7	I/O port P2	• 8-bit I/O port with the same function as port P0. • Low-voltage input level. • High-breakdown-voltage P-channel open-drain output structure.	• FLD automatic display pins • Buzzer output pin ($P2_0$)
$P3_0$/FLD_{24}– $P3_7$/FLD_{31}	Output port P3	• 8-bit output port. • A pull-down resistor is built in between port P3 and the V_{EE} pin. • High-breakdown-voltage P-channel open-drain output structure. • At reset, this port is set to V_{EE} level.	• FLD automatic display pins
$P4_0$/INT_0, $P4_1$/INT_1, $P4_2$/INT_3	I/O port P4	• 7-bit I/O port with the same function as port P0. • CMOS compatible input level • N-channel open-drain output structure.	• Interrupt input pins In the mask option type P, INT_3 cannot be used.
$P4_3$/BUZ_{01}			• Buzzer output pin
$P4_4$/PWM_1			• PWM output pin (Timer output pin)
$P4_5$/T_{1OUT}, $P4_6$/T_{3OUT}			• Timer output pin
$P4_7$/INT_2	Input port P4	• 1-bit input port. • CMOS compatible input level.	• Interrupt input pin

Table 2 Pin description (2)

Pin	Name	Function	Function except a port function
P5$_0$/SIN$_1$, P5$_1$/SOUT$_1$, P5$_2$/SCLK$_{11}$, P5$_3$/SCLK$_{12}$	I/O port P5	• 8-bit CMOS I/O port with the same function as port P0. • CMOS compatible input level. • CMOS 3-state output structure.	• Serial I/O1 function pins
P5$_4$/RxD, P5$_5$/TxD, P5$_6$/SCLK$_{21}$, P5$_7$/$\overline{\text{SRDY}_2}$/SCLK$_{22}$			• Serial I/O2 function pins
P6$_0$/CNTR$_1$	I/O port P6	• 1-bit I/O port with the same function as port P0. • CMOS compatible input level. • N-channel open-drain output structure.	• Timer input pin In the mask option type P, CNTR$_1$ cannot be used.
P6$_1$/CNTR$_0$/CNTR$_2$		• 5-bit CMOS I/O port with the same function as port P0. • CMOS compatible input level. • CMOS 3-state output structure.	• Timer I/O pin
P6$_2$/$\overline{\text{SRDY}_1}$/AN$_8$			• Serial I/O1 function pin • A-D conversion input pin
P6$_3$/AN$_9$			• A-D conversion input pin • Dimmer signal output pin
P6$_4$/INT$_4$/SBUSY$_1$/AN$_{10}$, P6$_5$/SSTB$_1$/AN$_{11}$			• Serial I/O1 function pin • A-D conversion input pin • Interrupt input pin (P6$_4$)
P7$_0$/AN$_0$–P7$_7$/AN$_7$	I/O port P7	• 8-bit CMOS I/O port with the same function as port P0. • CMOS compatible input level. • CMOS 3-state output structure.	• A-D conversion input pin
P8$_0$/FLD$_{32}$–P8$_3$/FLD$_{35}$	I/O port P8	• 4-bit I/O port with the same function as port P0. • Low-voltage input level. • High-breakdown-voltage P-channel open-drain output structure.	• FLD automatic display pins
P8$_4$/FLD$_{36}$		• 4-bit CMOS I/O port with the same function as port P0. • Low-voltage input level. • CMOS 3-state output structure	
P8$_5$/RTP$_0$/FLD$_{37}$, P8$_6$/RTP$_1$/FLD$_{38}$			• FLD automatic display pins • Real time port output
P8$_7$/PWM$_0$/FLD$_{39}$			• FLD automatic display pins • 14-bit PWM output
P9$_0$/XCIN, P9$_1$/XCOUT	I/O port P9	• 2-bit CMOS I/O port with the same function as port P0. • CMOS compatible input level. • CMOS 3-state output structure.	• I/O pins for sub-clock generating circuit (connect a ceramic resonator or a quarts-crystal oscillator)

PART NUMBERING

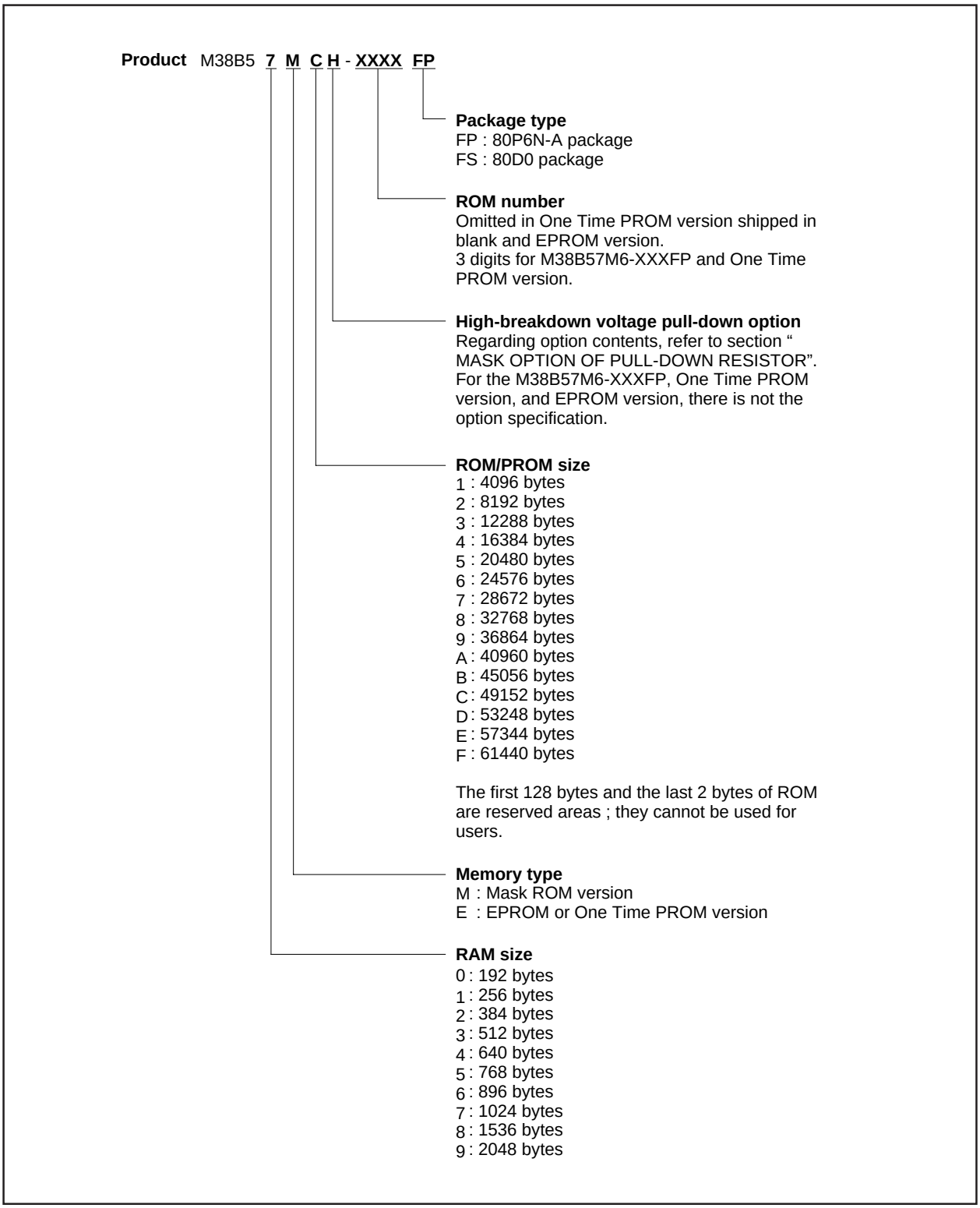


Fig. 3 Part numbering

GROUP EXPANSION

Mitsubishi plans to expand the 38B5 group as follows:

Memory Type

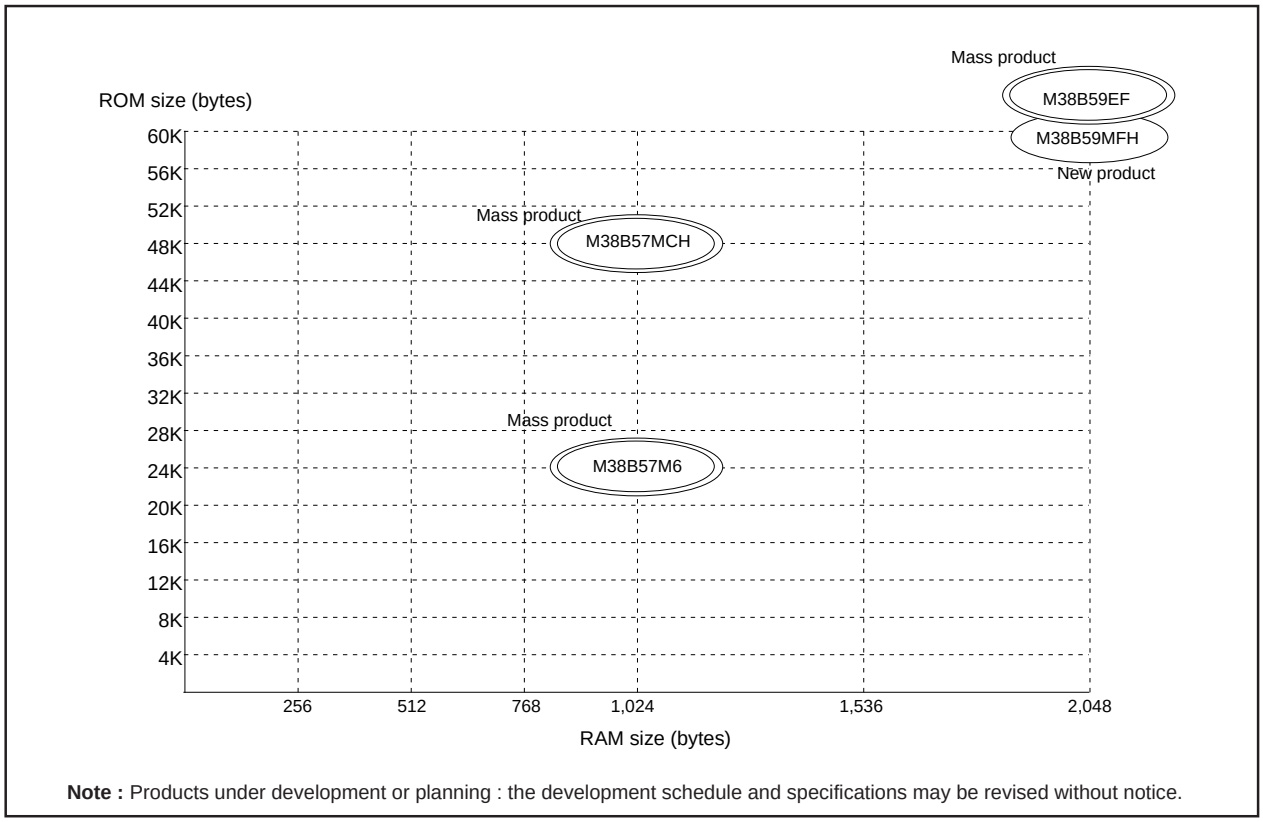
Support for Mask ROM, One Time PROM and EPROM versions.

Memory Size

ROM/PROM size ..... 24K to 60K bytes
RAM size ..... 1024 to 2048 bytes

Package

80P6N-A ..... 0.8 mm-pitch plastic molded QFP
80D0 ..... 0.8 mm-pitch ceramic LCC (EPROM version)



Note : Products under development or planning : the development schedule and specifications may be revised without notice.

Fig. 4 Memory expansion plan

Currently supported products are listed below.

Table 3 List of supported products **As of Nov. 1998**

Product	(P) ROM size (bytes) ROM size for User ( )	RAM size (bytes)	Package	Remarks
M38B57M6-XXXFP	24576 (24446)	1024	80P6N-A	Mask ROM version Corresponded to mask option
M38B57MCH-XXXXFP	49152 (49022)	1024	80P6N-A	Mask ROM version
M38B59MFH-XXXXFP	61440 (61310)	2048	80P6N-A	Mask ROM version Corresponded to mask option
M38B59EF-XXXFP	61440 (61310)	2048	80P6N-A	One Time PROM version
M38B59EFFP	61440 (61310)	2048	80P6N-A	One Time PROM version (blank)
M38B59EFFS	61440 (61310)	2048	80D0	EPROM version

FUNCTIONAL DESCRIPTION
Central Processing Unit (CPU)

The 38B5 group uses the standard 740 Family instruction set. Refer to the table of 740 Series addressing modes and machine instructions or the 740 Series Software Manual for details on the instruction set.

Machine-resident 740 Series instructions are as follows:

The FST and SLW instructions cannot be used.

The STP, WIT, MUL, and DIV instructions can be used.

[Accumulator (A)]

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

[Index Register X (X)]

The index register X is an 8-bit register. In the index addressing modes, the value of the OPERAND is added to the contents of register X and specifies the real address.

[Index Register Y (Y)]

The index register Y is an 8-bit register. In partial instruction, the value of the OPERAND is added to the contents of register Y and specifies the real address.

[Stack Pointer (S)]

The stack pointer is an 8-bit register used during subroutine calls and interrupts. This register indicates start address of stored area (stack) for storing registers during subroutine calls and interrupts.

The low-order 8 bits of the stack address are determined by the contents of the stack pointer. The high-order 8 bits of the stack address are determined by the stack page selection bit. If the stack page selection bit is “0” , the high-order 8 bits becomes “00_{16}”. If the stack page selection bit is “1”, the high-order 8 bits becomes “01_{16}”.

The operations of pushing register contents onto the stack and popping them from the stack are shown in Figure 6.

Store registers other than those described in Figure 6 with program when the user needs them during interrupts or subroutine calls.

[Program Counter (PC)]

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

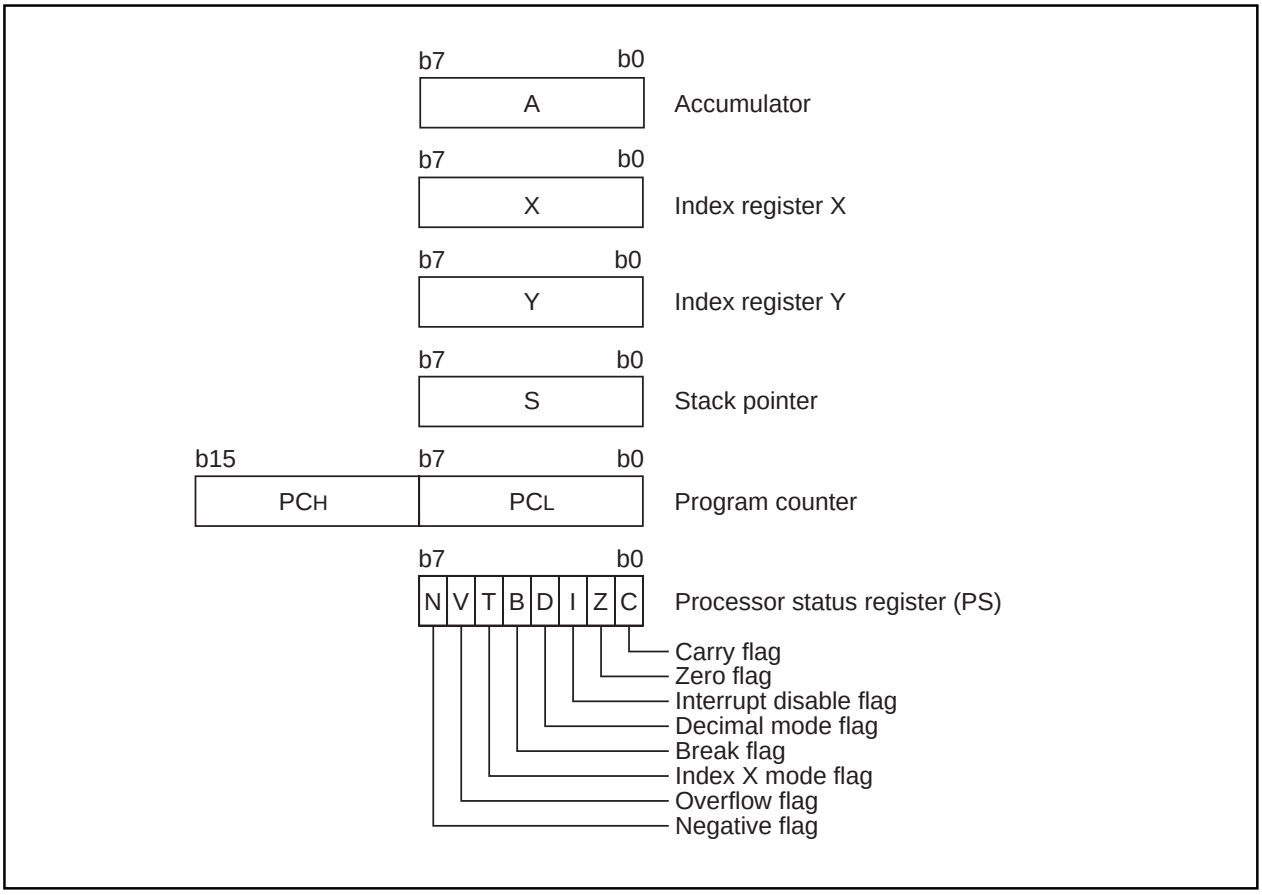


Fig. 5 740 Family CPU register structure

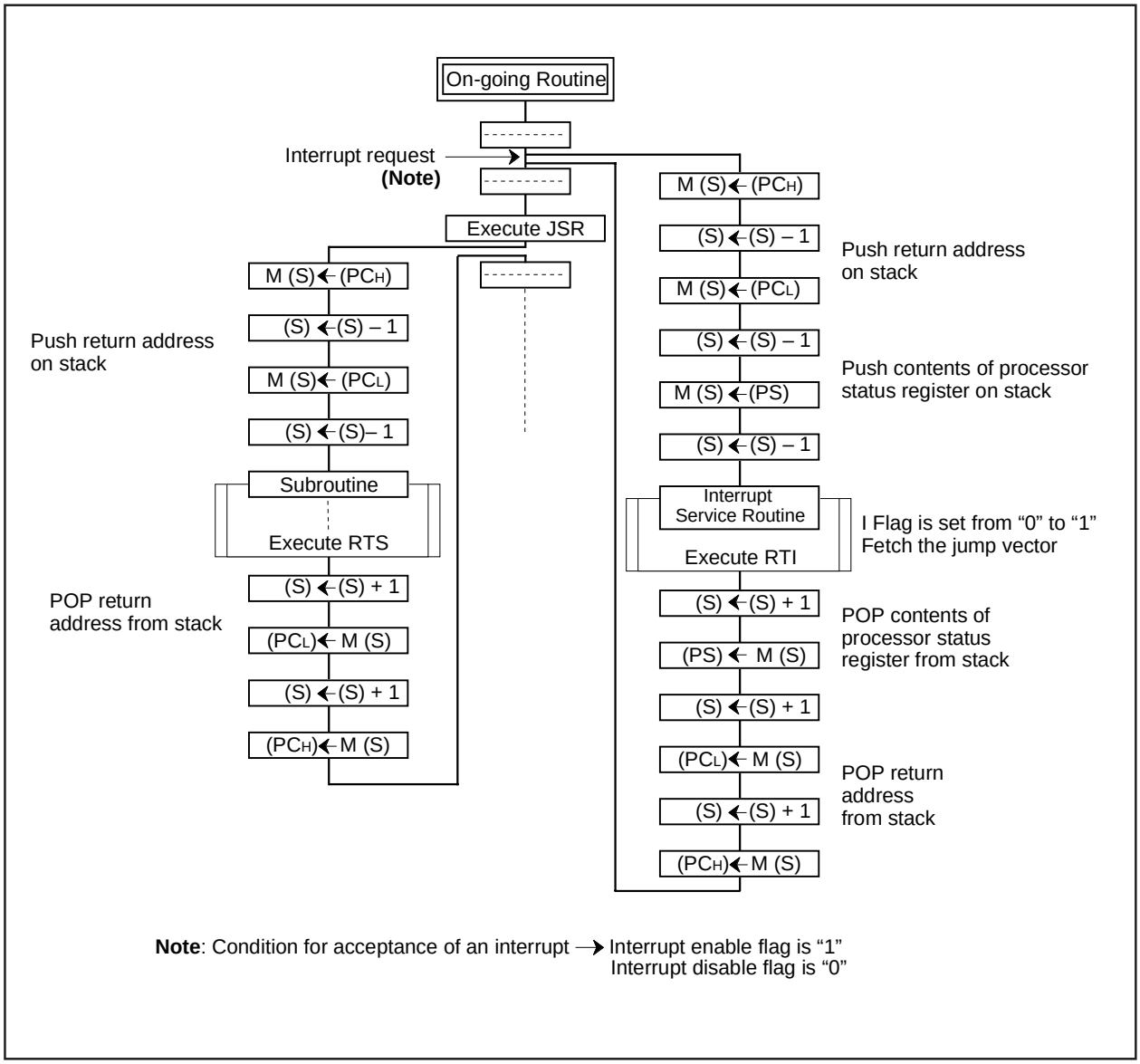


Fig. 6 Register push and pop at interrupt generation and subroutine call

Table 4 Push and pop instructions of accumulator or processor status register

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

[Processor status register (PS)]

The processor status register is an 8-bit register consisting of 5 flags which indicate the status of the processor after an arithmetic operation and 3 flags which decide MCU operation. Branch operations can be performed by testing the Carry (C) flag , Zero (Z) flag, Overflow (V) flag, or the Negative (N) flag. In decimal mode, the Z, V, N flags are not valid.

•Bit 0: Carry flag (C)

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

•Bit 1: Zero flag (Z)

The Z flag is set if the result of an immediate arithmetic operation or a data transfer is “0”, and cleared if the result is anything other than “0”.

•Bit 2: Interrupt disable flag (I)

The I flag disables all interrupts except for the interrupt generated by the BRK instruction.

Interrupts are disabled when the I flag is “1”.

•Bit 3: Decimal mode flag (D)

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

•Bit 4: Break flag (B)

The B flag is used to indicate that the current interrupt was generated by the BRK instruction. The BRK flag in the processor status register is always “0”. When the BRK instruction is used to generate an interrupt, the processor status register is pushed onto the stack with the break flag set to “1”.

•Bit 5: Index X mode flag (T)

When the T flag is “0”, arithmetic operations are performed between accumulator and memory. When the T flag is “1”, direct arithmetic operations and direct data transfers are enabled between memory locations.

•Bit 6: Overflow flag (V)

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

•Bit 7: Negative flag (N)

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

Table 5 Set and clear instructions of each bit of processor status register

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

[CPU Mode Register (CPUM)] 003B16

The CPU mode register contains the stack page selection bit and the internal system clock selection bit etc.
The CPU mode register is allocated at address 003B16.

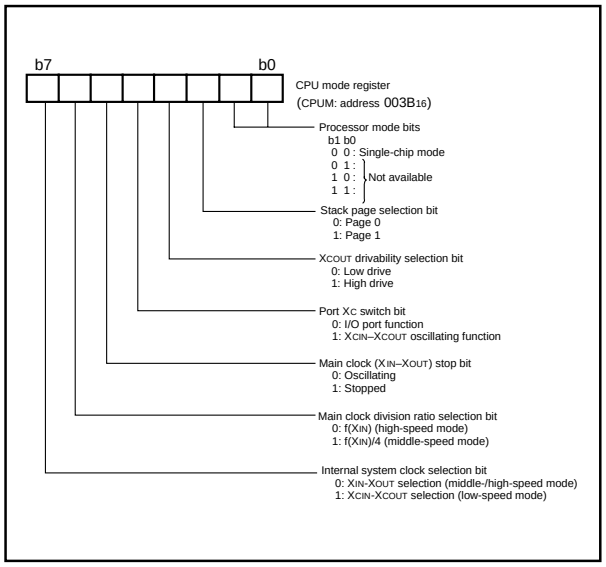


Fig. 7 Structure of CPU mode register

Memory

Special function register (SFR) area

The special function register (SFR) area in the zero page contains control registers such as I/O ports and timers.

RAM

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

ROM

The first 128 bytes and the last 2 bytes of ROM are reserved for device testing, and the other areas are user areas for storing programs.

Interrupt vector area

The interrupt vector area contains reset and interrupt vectors.

Zero page

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

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

Special page

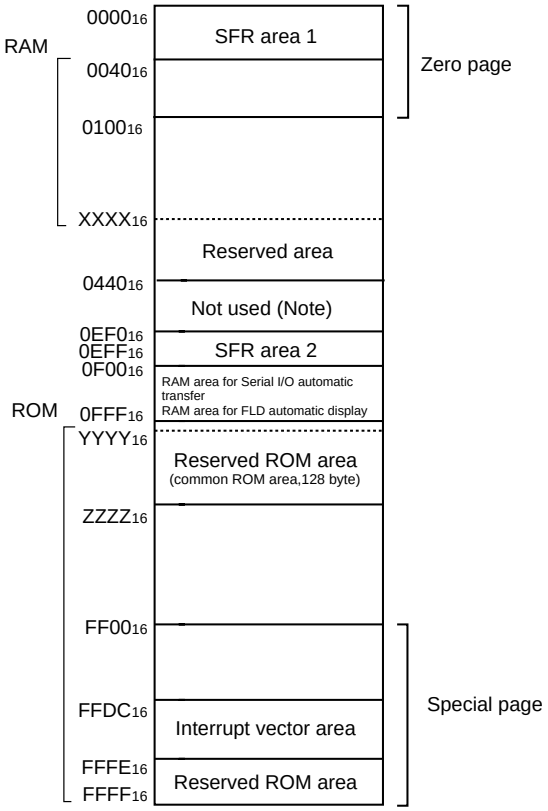
The 256 bytes from addresses $FF00_{16}$ to $FFFF_{16}$ are called the special page area. The special page addressing mode can be used to specify memory addresses in the special page area. Access to this area with only 2 bytes is possible in the special page addressing mode.

RAM area

RAM size (byte)	Address XXXX16
192	$00FF_{16}$
256	$013F_{16}$
384	$01BF_{16}$
512	$023F_{16}$
640	$02BF_{16}$
768	$033F_{16}$
896	$03BF_{16}$
1024	$043F_{16}$
1536	$063F_{16}$
2048	$083F_{16}$

ROM area

ROM size (byte)	Address YYYY16	Address ZZZZ16
4096	$F000_{16}$	$F080_{16}$
8192	$E000_{16}$	$E080_{16}$
12288	$D000_{16}$	$D080_{16}$
16384	$C000_{16}$	$C080_{16}$
20480	$B000_{16}$	$B080_{16}$
24576	$A000_{16}$	$A080_{16}$
28672	9000_{16}	9080_{16}
32768	8000_{16}	8080_{16}
36864	7000_{16}	7080_{16}
40960	6000_{16}	6080_{16}
45056	5000_{16}	5080_{16}
49152	4000_{16}	4080_{16}
53248	3000_{16}	3080_{16}
57344	2000_{16}	2080_{16}
61440	1000_{16}	1080_{16}



Note: When 1024 bytes or more are used as RAM area, this area can be used.

Fig. 8 Memory map diagram

0000_{16}	Port P0 (P0)	0020_{16}	Timer 1 (T1)
0001_{16}	Port P0 direction register (P0D)	0021_{16}	Timer 2 (T2)
0002_{16}	Port P1 (P1)	0022_{16}	Timer 3 (T3)
0003_{16}		0023_{16}	Timer 4 (T4)
0004_{16}	Port P2 (P2)	0024_{16}	Timer 5 (T5)
0005_{16}	Port P2 direction register (P2D)	0025_{16}	Timer 6 (T6)
0006_{16}	Port P3 (P3)	0026_{16}	PWM control register (PWMCON)
0007_{16}		0027_{16}	Timer 6 PWM register (T6PWM)
0008_{16}	Port P4 (P4)	0028_{16}	Timer 12 mode register (T12M)
0009_{16}	Port P4 direction register (P4D)	0029_{16}	Timer 34 mode register (T34M)
$000A_{16}$	Port P5 (P5)	$002A_{16}$	Timer 56 mode register (T56M)
$000B_{16}$	Port P5 direction register (P5D)	$002B_{16}$	Watchdog timer control register (WDTCON)
$000C_{16}$	Port P6 (P6)	$002C_{16}$	Timer X (low-order) (TXL)
$000D_{16}$	Port P6 direction register (P6D)	$002D_{16}$	Timer X (high-order) (TXH)
$000E_{16}$	Port P7 (P7)	$002E_{16}$	Timer X mode register 1 (TXM1)
$000F_{16}$	Port P7 direction register (P7D)	$002F_{16}$	Timer X mode register 2 (TXM2)
0010_{16}	Port P8 (P8)	0030_{16}	Interrupt interval determination register (IID)
0011_{16}	Port P8 direction register (P8D)	0031_{16}	Interrupt interval determination control register (IIDCON)
0012_{16}	Port P9 (P9)	0032_{16}	A-D control register (ADCON)
0013_{16}	Port P9 direction register (P9D)	0033_{16}	A-D conversion register (low-order) (ADL)
0014_{16}	PWM register (high-order) (PWMH)	0034_{16}	A-D conversion register (high-order) (ADH)
0015_{16}	PWM register (low-order) (PWM L)	0035_{16}	
0016_{16}	Baud rate generator (BRG)	0036_{16}	
0017_{16}	UART control register (UARTCON)	0037_{16}	
0018_{16}	Serial I/O1 automatic transfer data pointer (SIO1DP)	0038_{16}	
0019_{16}	Serial I/O1 control register 1 (SIO1CON1)	0039_{16}	Interrupt source switch register (IFR)
$001A_{16}$	Serial I/O1 control register 2 (SIO1CON2)	$003A_{16}$	Interrupt edge selection register (INTEDGE)
$001B_{16}$	Serial I/O1 register/Transfer counter (SIO1)	$003B_{16}$	CPU mode register (CPUM)
$001C_{16}$	Serial I/O1 control register 3 (SIO1CON3)	$003C_{16}$	Interrupt request register 1(IREQ1)
$001D_{16}$	Serial I/O2 control register (SIO2CON)	$003D_{16}$	Interrupt request register 2(IREQ2)
$001E_{16}$	Serial I/O2 status register (SIO2STS)	$003E_{16}$	Interrupt control register 1(ICON1)
$001F_{16}$	Serial I/O2 transmit/receive buffer register (TB/RB)	$003F_{16}$	Interrupt control register 2(ICON2)

$0EF0_{16}$	Pull-up control register 1 (PULL1)	$0EF8_{16}$	FLD data pointer (FLDDP)
$0EF1_{16}$	Pull-up control register 2 (PULL2)	$0EF9_{16}$	Port P0FLD/port switch register (P0FPR)
$0EF2_{16}$	P1FLDRAM write disable register (P1FLDRAM)	$0EFA_{16}$	Port P2FLD/port switch register (P2FPR)
$0EF3_{16}$	P3FLDRAM write disable register (P3FLDRAM)	$0EFB_{16}$	Port P8FLD/port switch register (P8FPR)
$0EF4_{16}$	FLDC mode register (FLDM)	$0EFC_{16}$	Port P8FLD output control register (P8FLDCON)
$0EF5_{16}$	Tdisp time set register (TDISP)	$0EFD_{16}$	Buzzer output control register (BUZCON)
$0EF6_{16}$	Toff1 time set register (TOFF1)	$0EFE_{16}$	
$0EF7_{16}$	Toff2 time set register (TOFF2)	$0EFF_{16}$	

Fig. 9 Memory map of special function register (SFR)

I/O Ports
[Direction Registers] PiD

The 38B5 group has 55 programmable I/O pins arranged in eight individual I/O ports (P0, P2, $P4_0$–$P4_6$, and P5–P9). The I/O ports have direction registers which determine the input/output direction of each individual pin. Each bit in a direction register corresponds to one pin, and each pin can be set to be input port or output port. When "0" is written to the bit corresponding to a pin, that pin becomes an input pin. When "1" is written to that pin, that pin becomes an output pin. If data is read from a pin set to output, the value of the port output latch is read, not the value of the pin itself. Pins set to input (the bit corresponding to that pin must be set to "0") are floating and the value of that pin can be read. If a pin set to input is written to, only the port output latch is written to and the pin remains floating.

[High-Breakdown-Voltage Output Ports]

The 38B5 group has 5 ports with high-breakdown-voltage pins (ports P0–P3 and $P8_0$–$P8_3$). The high-breakdown-voltage ports have P-channel open-drain output with Vcc- 45 V of breakdown voltage. Each pin in ports P0, P1, and P3 has an internal pull-down resistor connected to V_{EE}. At reset, the P-channel output transistor of each port latch is turned off, so that it goes to V_{EE} level ("L") by the pull-down resistor.

Writing "1" (weak drivability) to bit 7 of the FLDC mode register (address $0EF4_{16}$) shows the rising transition of the output transistors for reducing transient noise. At reset, bit 7 of the FLDC mode register is set to "0" (strong drivability).

[Pull-up Control Register] PULL

Ports P5, $P6_1$–$P6_5$, P7, $P8_4$–$P8_7$ and P9 have built-in programmable pull-up resistors. The pull-up resistors are valid only in the case that the each control bit is set to "1" and the corresponding port direction registers are set to input mode.

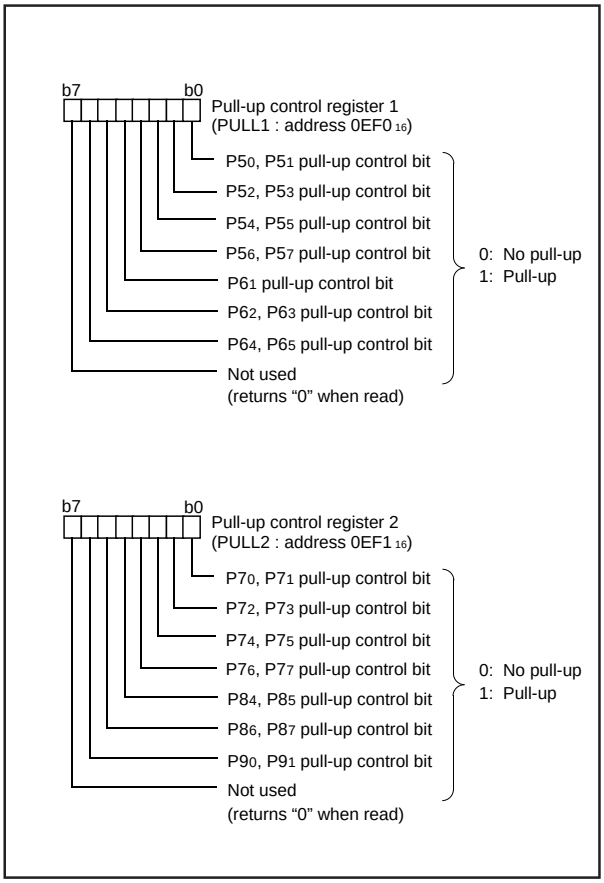


Fig. 10 Structure of pull-up control registers (PULL1 and PULL2)

Table 6 List of I/O port functions (1)

Pin	Name	Input/Output	I/O Format	Non-Port Function	Related SFRs	Ref.No.
$P0_0$/FLD_8–$P0_7$/FLD_{15}	Port P0	Input/output, individual bits	CMOS compatible input level High-breakdown voltage P-channel open-drain output with pull-down resistor	FLD automatic display function	FLDC mode register Port P0FLD/port switch register	(1)
$P1_0$/FLD_{16}–$P1_7$/FLD_{23}	Port P1	Output	High-breakdown voltage P-channel open-drain output with pull-down resistor		FLDC mode register	(2)
$P2_0$/BUZ02/FLD_0	Port P2	Input/output, individual bits	Low-voltage input level High-breakdown voltage P-channel open-drain output	Buzzer output ($P2_0$) FLD automatic display function	FLDC mode register Port P2FLD/port switch register	(3)
$P2_1$/FLD_1–$P2_7$/FLD_7				FLD automatic display function	Buzzer output control register	(1)
$P3_0$/FLD_{24}–$P3_7$/FLD_{31}	Port P3	Output	High-breakdown voltage P-channel open-drain output with pull-down resistor		FLDC mode register	(2)
$P4_0$/INT_0, $P4_1$/INT_1, $P4_2$/INT_3	Port P4	Input/output, individual bits	CMOS compatible input level N-channel open-drain output	External interrupt input In the mask option type P, INT_3 cannot be used.	Interrupt edge selection register	(5-1) (5-2)
$P4_3$/BUZ01				Buzzer output	Buzzer output control register	(4)
$P4_4$/PWM_1				PWM output	Timer 56 mode register	(6)
$P4_5$/T_{1OUT}				Timer output	Timer 12 mode register	(7)
$P4_6$/T_{3OUT}				Timer output	Timer 34 mode register	(7)
$P4_7$/INT_2		Input	CMOS compatible input level	External interrput input	Interrupt edge selection register Interrupt interval determination control register	(8)
$P5_0$/S_{IN1}	Port P5	Input/output, individual bits	CMOS compatible input level CMOS 3-state output	Serial I/O1 function I/O	Serial I/O1 control register 1, 2	(9)
$P5_1$/S_{OUT1}, $P5_2$/S_{CLK11}, $P5_3$/S_{CLK12}						(10)
$P5_4$/RxD,				Serial I/O2 function I/O	Serial I/O2 control register UART control register	(9)
$P5_5$/TxD, $P5_6$/S_{CLK21}						(10)
$P5_7$/$\overline{S_{RDY2}}$/S_{CLK22}						(11)
$P6_0$/$CNTR_1$	Port P6		CMOS compatible input level N-channel open-drain output	External count input In the mask option type P, $CNTR_1$ cannot be used.	Interrupt edge selection register	(5-1) (5-2)
$P6_1$/$CNTR_0$/$CNTR_2$			CMOS compatible input level CMOS 3-state output			(12)
$P6_2$/$\overline{S_{RDY1}}$/AN_8				Serial I/O1 function I/O A-D conversion input	Serial I/O1 control register 1, 2 A-D control register	(13)
$P6_3$/AN_9				A-D conversion input Dimmer signal output	A-D control register P8FLD output control bit	(14)
$P6_4$/INT_4/S_{BUSY1}/AN_{10}				Serial I/O1 function I/O A-D conversion input External interrupt input	Serial I/O1 control register 1, 2 A-D control register Interrupt edge selection register	(15)
$P6_5$/S_{STB1}/AN_{11}				Serial I/O1 function I/O A-D conversion input	Serial I/O1 control register 1, 2 A-D control register	(16)
$P7_0$/AN_0–$P7_7$/AN_7	Port P7			A-D conversion input	A-D control register	(14)

Table 7 List of I/O port functions (2)

Pin	Name	Input/Output	I/O Format	Non-Port Function	Related SFRs	Ref.No.
P80/FLD32–P83/FLD35	Port P8	Input/output, individual bits	Low-voltage input level High-breakdown voltage P-channel open-drain output	FLD automatic display function	FLDC mode register Port P8FLD/port switch register	(1)
P84/FLD36			Low-voltage input level			(17)
P85/RTP0/FLD37, P86/RTP1/FLD38			CMOS 3-state output	FLD automatic display function Real time port output	FLDC mode register Port P8FLD/port switch register Timer X mode register 2	(18)
P87/PWM0/FLD39				FLD automatic display function PWM output	FLDC mode register Port P8FLD/port switch register PWM control register	(19)
P90/XCIN	Port P9		CMOS compatible input level	Sub-clock generating circuit I/O	CPU mode register	(20)
P91/XCOUT			CMOS 3-state output			(21)

Notes 1 : How to use double-function ports as function I/O ports, refer to the applicable sections.

2 : Make sure that the input level at each pin is either 0 V or Vcc during execution of the STP instruction.
When an input level is at an intermediate potential, a current will flow from Vcc to Vss through the input-stage gate.

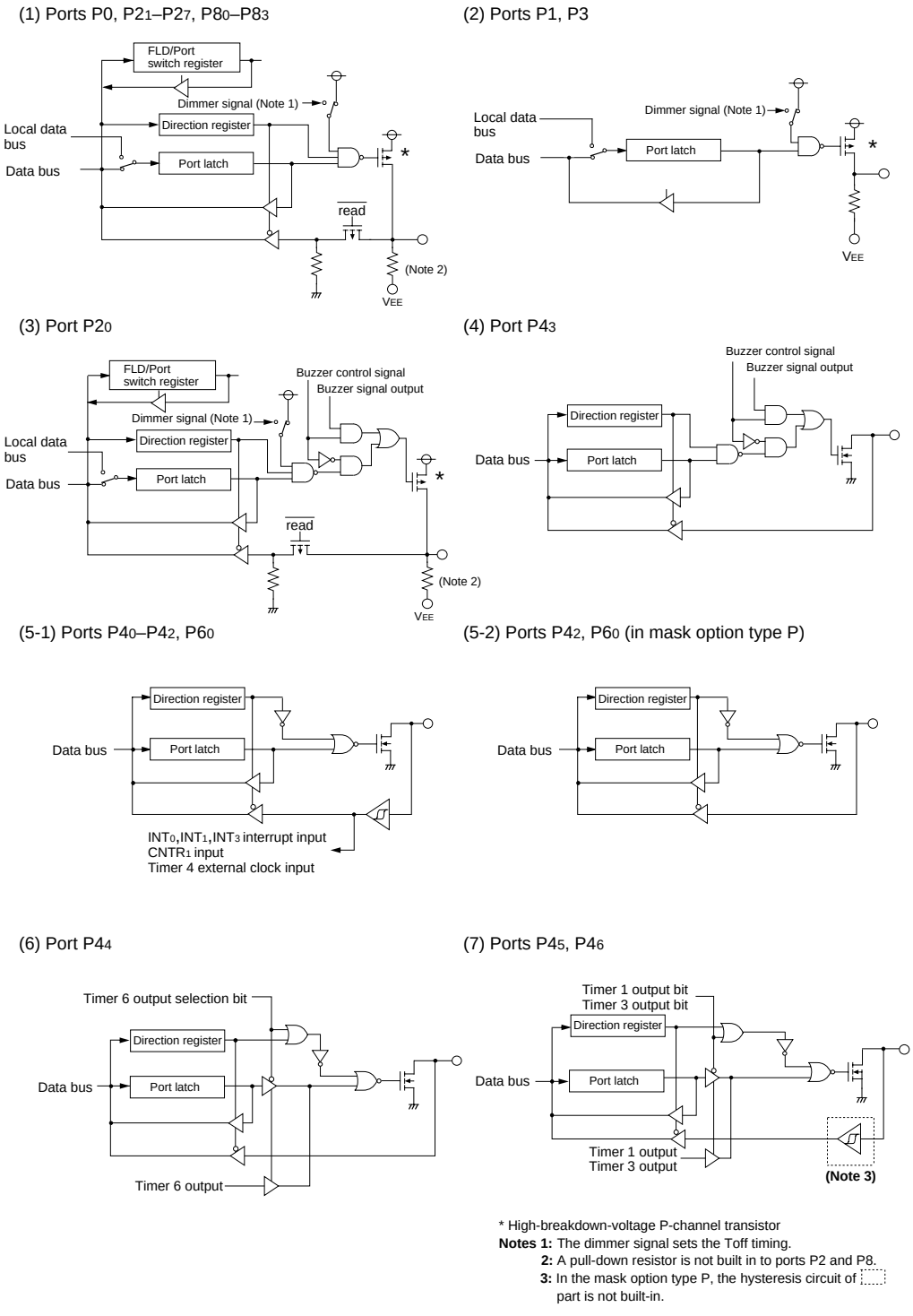
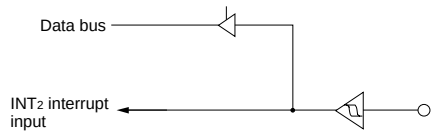
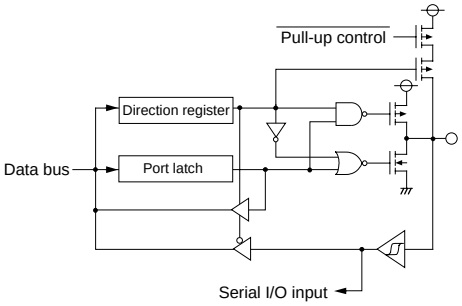


Fig. 11 Port block diagram (1)

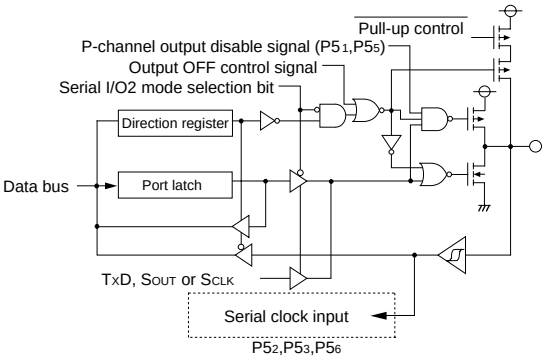
(8) Port P4$_7$



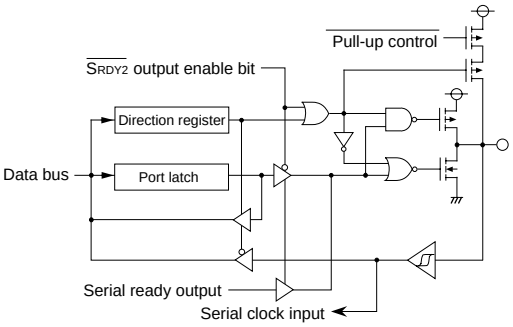
(9) Ports P5$_0$, P5$_4$



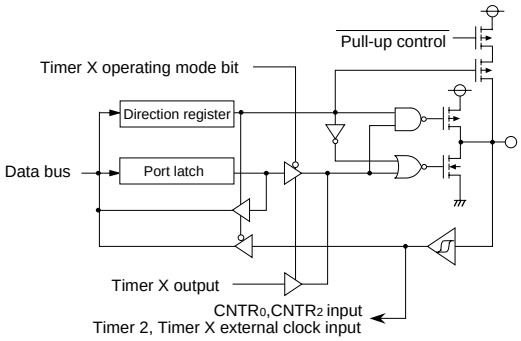
(10) Ports P5$_1$–P5$_3$, P5$_5$, P5$_6$



(11) Port P5$_7$



(12) Port P6$_1$



(13) Port P6$_2$

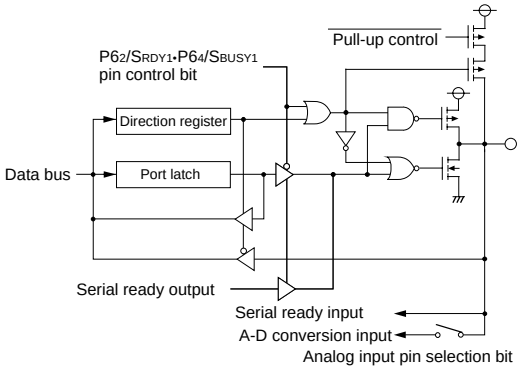


Fig. 12 Port block diagram (2)

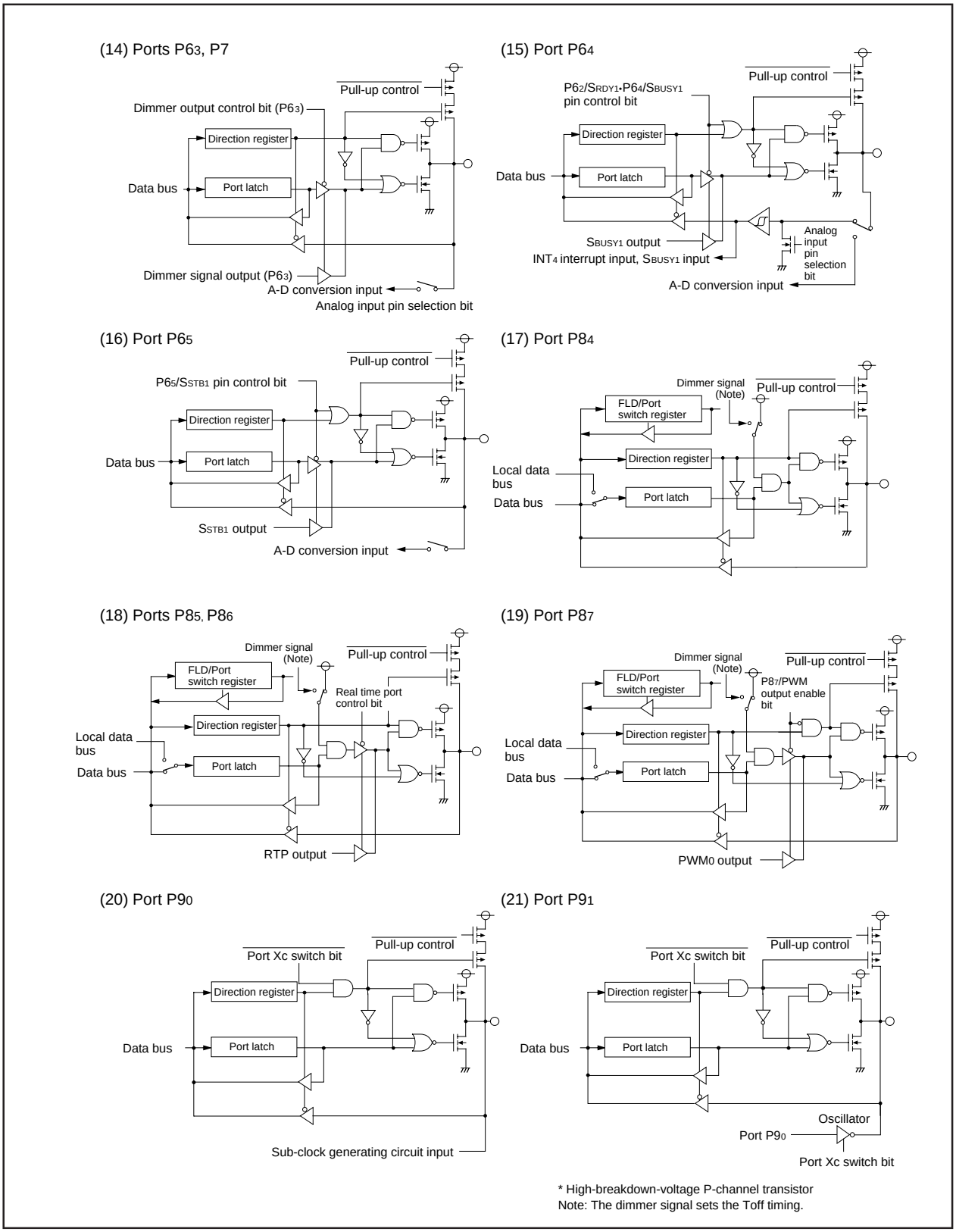


Fig. 13 Port block diagram (3)

Interrupts

Interrupts occur by twenty one sources: five external, fifteen internal, and one software.

(1) Interrupt Control

Each interrupt except the BRK instruction interrupt have both an interrupt request bit and an interrupt enable bit, and is controlled by the interrupt disable flag. An interrupt occurs if the corresponding interrupt request and enable bits are “1” and the interrupt disable flag is “0.” Interrupt enable bits can be set or cleared by software. Interrupt request bits can be cleared by software, but cannot be set by software. The BRK instruction interrupt and reset cannot be disabled with any flag or bit. The I flag disables all interrupts except the BRK instruction interrupt and reset. If several interrupts requests occurs at the same time the interrupt with highest priority is accepted first.

(2) Interrupt Operation

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

1. The contents of the program counter and processor status register are automatically pushed onto the stack.
2. The interrupt disable flag is set and the corresponding interrupt request bit is cleared.
3. The interrupt jump destination address is read from the vector table into the program counter.

■Notes on Use

When the active edge of an external interrupt (INT_0–INT_4) is set or when switching interrupt sources in the same vector address, the corresponding interrupt request bit may also be set. Therefore, please take following sequence:

(1) Disable the external interrupt which is selected.
(2) Change the active edge in interrupt edge selection register
(3) Clear the set interrupt request bit to “0.”
(4) Enable the external interrupt which is selected.

Table 8 Interrupt vector addresses and priority

Interrupt Source	Priority	Vector Addresses **(Note 1)**		Interrupt Request Generating Conditions	Remarks
		High	Low		
Reset **(Note 2)**	1	FFFD16	FFFC16	At reset	Non-maskable
INT0	2	FFFB16	FFFA16	At detection of either rising or falling edge of INT0 input	External interrupt (active edge selectable)
INT1	3	FFF916	FFF816	At detection of either rising or falling edge of INT1 input	External interrupt (active edge selectable)
INT2	4	FFF716	FFF616	At detection of either rising or falling edge of INT2 input	External interrupt (active edge selectable)
Remote control/ counter overflow				At 8-bit counter overflow	Valid when interrupt interval determination is operating
Serial I/O1	5	FFF516	FFF416	At completion of data transfer	Valid when serial I/O ordinary mode is selected
Serial I/O auto-matic transfer				At completion of the last data transfer	Valid when serial I/O automatic transfer mode is selected
Timer X	6	FFF316	FFF216	At timer X underflow	
Timer 1	7	FFF116	FFF016	At timer 1 underflow	
Timer 2	8	FFEF16	FFEE16	At timer 2 underflow	STP release timer underflow
Timer 3	9	FFED16	FFEC16	At timer 3 underflow	
Timer 4	10	FFEB16	FFEA16	At timer 4 underflow	**(Note 3)**
Timer 5	11	FFE916	FFE816	At timer 5 underflow	
Timer 6	12	FFE716	FFE616	At timer 6 underflow	
Serial I/O2 receive	13	FFE516	FFE416	At completion of serial I/O2 data receive	
INT3	14	FFE316	FFE216	At detection of either rising or falling edge of INT3 input	External interrupt **(Note 4)** (active edge selectable)
Serial I/O2 transmit				At completion of serial I/O2 data transmit	
INT4	15	FFE116	FFE016	At detection of either rising or falling edge of INT4 input	External interrupt (active edge selectable)
A-D conversion				At completion of A-D conversion	Valid when INT4 interrupt is selected Valid when A-D conversion is selected
FLD blanking	16	FFDF16	FFDE16	At falling edge of the last timing immediately before blanking period starts	Valid when FLD blanking interrupt is selected
FLD digit				At rising edge of digit (each timing)	Valid when FLD digit interrupt is selected
BRK instruction	17	FFDD16	FFDC16	At BRK instruction execution	Non-maskable software interrupt

Notes 1 : Vector addresses contain interrupt jump destination addresses.

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

3 : In the mask option type P, timer 4 interrupt whose count source is CNTR1 input cannot be used.

4 : In the mask option type P, INT3 interrupt cannot be used.

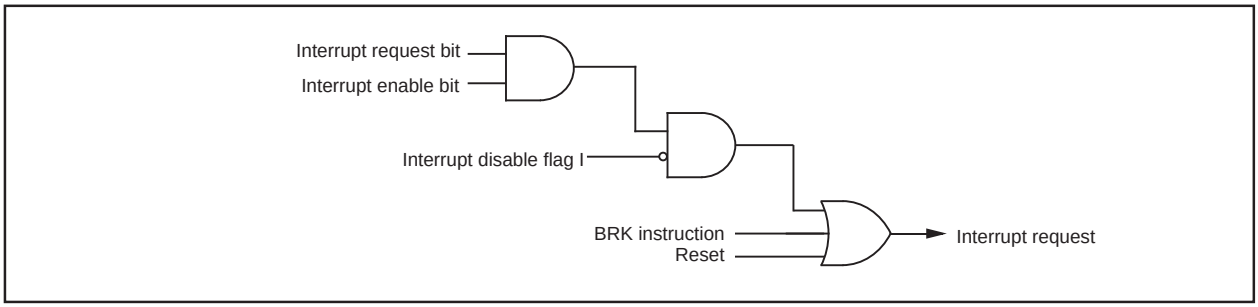
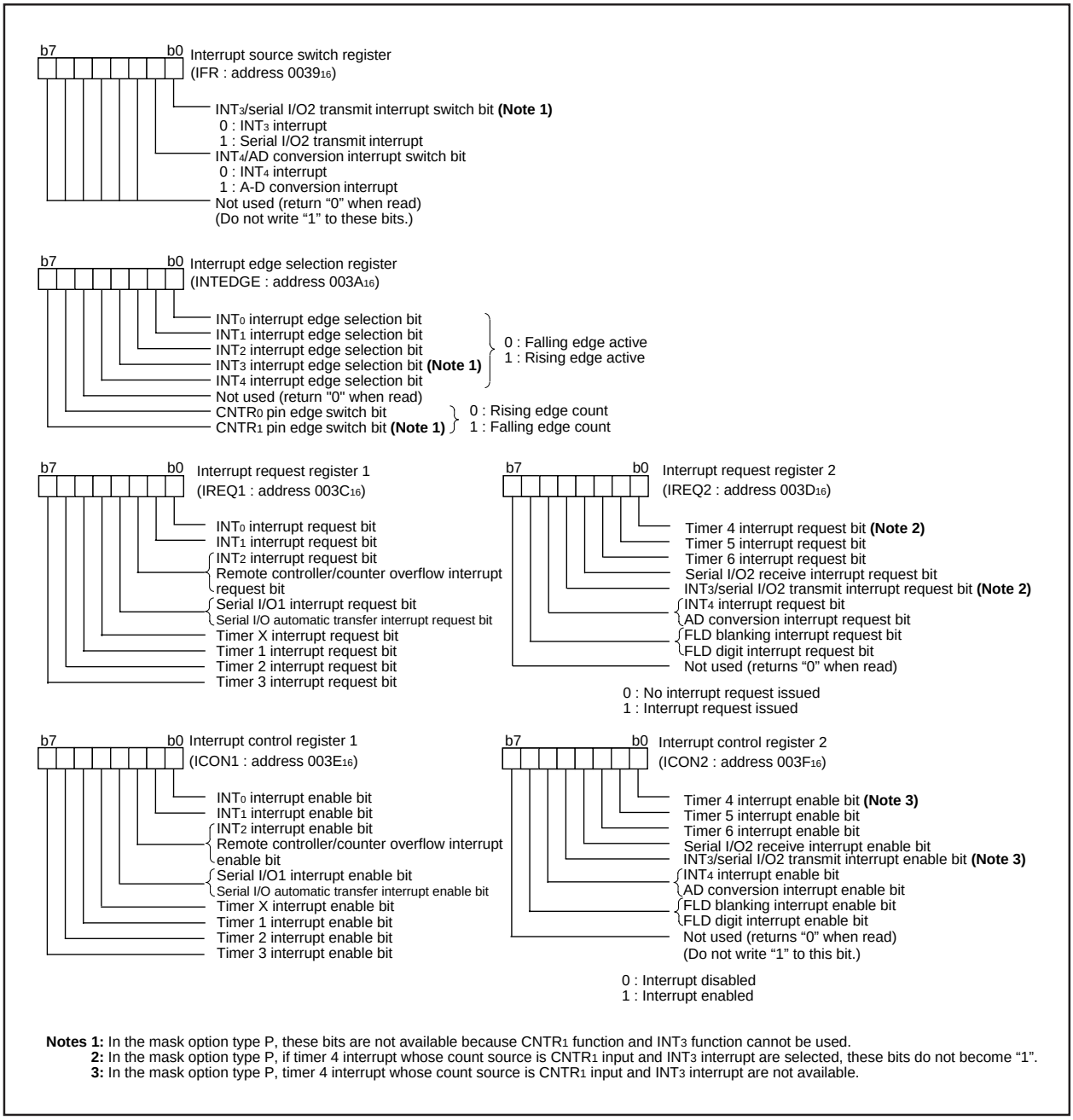


Fig. 14 Interrupt control



Notes 1: In the mask option type P, these bits are not available because CNTR₁ function and INT₃ function cannot be used.
2: In the mask option type P, if timer 4 interrupt whose count source is CNTR₁ input and INT₃ interrupt are selected, these bits do not become "1".
3: In the mask option type P, timer 4 interrupt whose count source is CNTR₁ input and INT₃ interrupt are not available.

Fig. 15 Structure of interrupt related registers

Timers
8-Bit Timer

The 38B5 group has six built-in timers : Timer 1, Timer 2, Timer 3, Timer 4, Timer 5, and Timer 6.

Each timer has the 8-bit timer latch. All timers are down-counters. When the timer reaches "00_{16}," an underflow occurs with the next count pulse. Then the contents of the timer latch is reloaded into the timer and the timer continues down-counting. When a timer underflows, the interrupt request bit corresponding to that timer is set to "1."

The count can be stopped by setting the stop bit of each timer to "1."

The internal system clock can be set to either the high-speed mode or low-speed mode with the CPU mode register. At the same time, timer internal count source is switched to either f(X_{IN}) or f(X_{CIN}).

●Timer 1, Timer 2

The count sources of timer 1 and timer 2 can be selected by setting the timer 12 mode register. A rectangular waveform of timer 1 underflow signal divided by 2 can be output from the $P4_5/T_{1OUT}$ pin. The active edge of the external clock $CNTR_0$ can be switched with the bit 6 of the interrupt edge selection register.

At reset or when executing the STP instruction, all bits of the timer 12 mode register are cleared to "0," timer 1 is set to "FF_{16}," and timer 2 is set to "01_{16}."

●Timer 3, Timer 4

The count sources of timer 3 and timer 4 can be selected by setting the timer 34 mode register. A rectangular waveform of timer 3 underflow signal divided by 2 can be output from the $P4_6/T_{3OUT}$ pin. The active edge of the external clock $CNTR_1$ **(Note)** can be switched with the bit 7 of the interrupt edge selection register.

Note: In the mask option type P, $CNTR_1$ function cannot be used.

●Timer 5, Timer 6

The count sources of timer 5 and timer 6 can be selected by setting the timer 56 mode register. A rectangular waveform of timer 6 underflow signal divided by 2 can be output from the $P4_4/PWM_1$ pin.

●Timer 6 PWM_1 Mode

Timer 6 can output a PWM rectangular waveform with "H" duty cycle n/(n+m) from the $P4_4/PWM_1$ pin by setting the timer 56 mode register (refer to Figure 18). The n is the value set in timer 6 latch (address 0025_{16}) and m is the value in the timer 6 PWM register (address 0027_{16}). If n is "0," the PWM output is "L," if m is "0," the PWM output is "H" (n = 0 is prior than m = 0). In the PWM mode, interrupts occur at the rising edge of the PWM output.

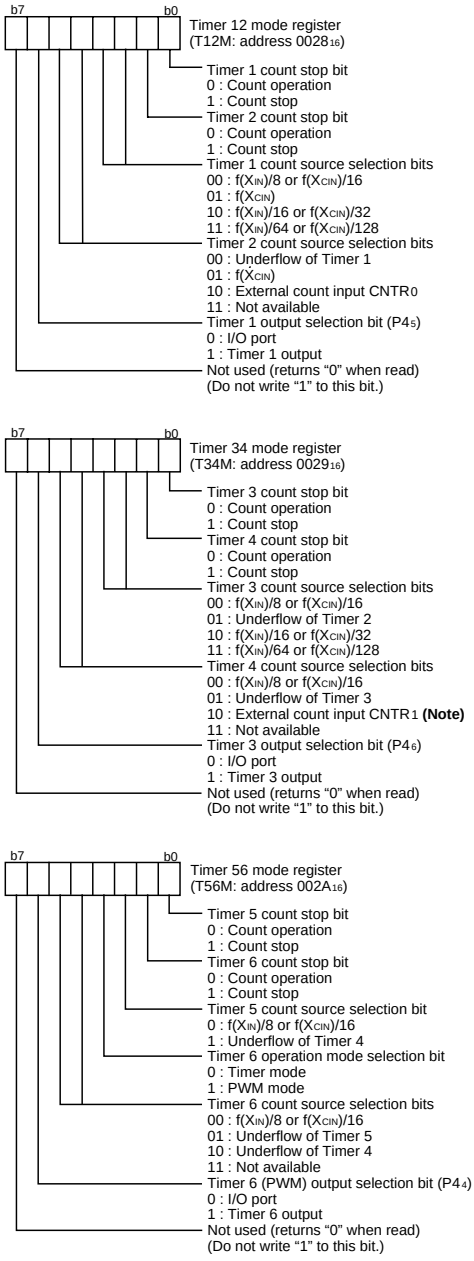


Fig. 16 Structure of timer related register

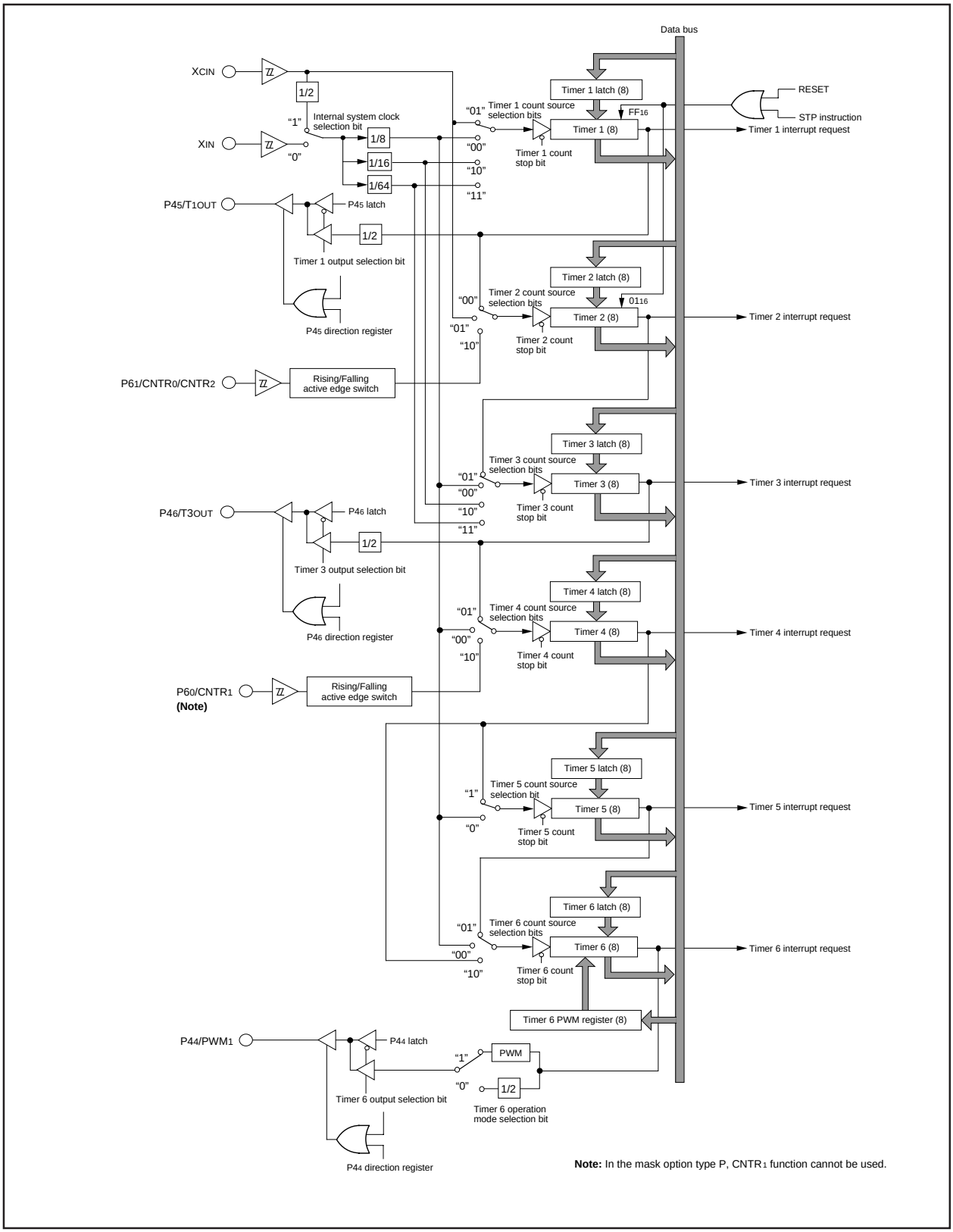
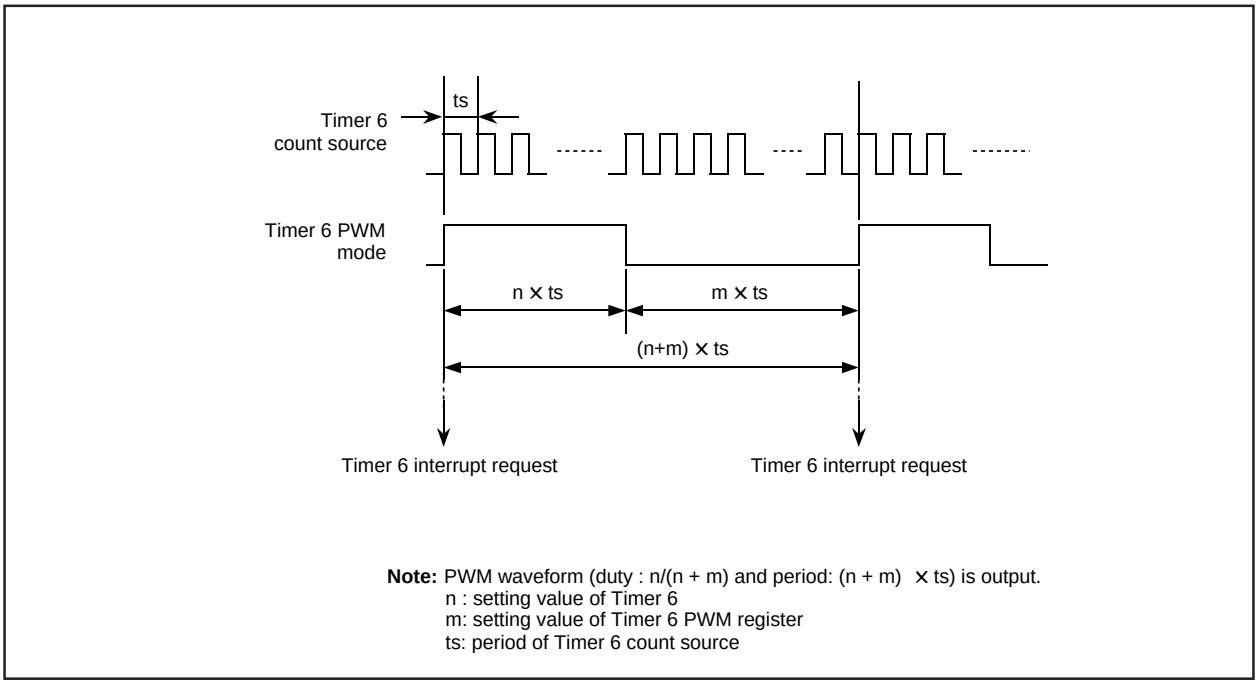


Fig. 17 Block diagram of timer



Note: PWM waveform (duty : n/(n + m) and period: (n + m) × ts) is output.
n : setting value of Timer 6
m: setting value of Timer 6 PWM register
ts: period of Timer 6 count source

Fig. 18 Timing chart of timer 6 PWM1 mode

16-Bit Timer

Timer X is a 16-bit timer that can be selected in one of four modes by the Timer X mode registers 1, 2 and can be controlled the timer X write and the real time port by setting the timer X mode registers.
Read and write operation on 16-bit timer must be performed for both high- and low-order bytes. When reading a 16-bit timer, read from the high-order byte first. When writing to 16-bit timer, write to the low-order byte first. The 16-bit timer cannot perform the correct operation when reading during write operation, or when writing during read operation.

●Timer X

Timer X is a down-counter. When the timer reaches "0000_{16}," an underflow occurs with the next count pulse. Then the contents of the timer latch is reloaded into the timer and the timer continues down-counting. When a timer underflows, the interrupt request bit corresponding to that timer is set to "1."

(1) Timer mode

A count source can be selected by setting the Timer X count source selection bits (bits 1 and 2) of the Timer X mode register 1.

(2) Pulse output mode

Each time the timer underflows, a signal output from the $CNTR_2$ pin is inverted. Except for this, the operation in pulse output mode is the same as in timer mode. When using a timer in this mode, set the port shared with the $CNTR_2$ pin to output.

(3) Event counter mode

The timer counts signals input through the $CNTR_2$ pin. Except for this, the operation in event counter mode is the same as in timer mode. When using a timer in this mode, set the port shared with the $CNTR_2$ pin to input.

(4) Pulse width measurement mode

A count source can be selected by setting the Timer X count source selection bits (bits 1 and 2) of the Timer X mode register 1. When $CNTR_2$ active edge switch bit is "0," the timer counts while the input signal of the $CNTR_2$ pin is at "H." When it is "1," the timer counts while the input signal of the $CNTR_2$ pin is at "L." When using a timer in this mode, set the port shared with the $CNTR_2$ pin to input.

■ Note

•Timer X Write Control

If the timer X write control bit is "0," when the value is written in the address of timer X, the value is loaded in the timer X and the latch at the same time.
If the timer X write control bit is "1," when the value is written in the address of timer X, the value is loaded only in the latch. The value in the latch is loaded in timer X after timer X underflows.
When the value is written in latch only, unexpected value may be set in the high-order counter if the writing in high-order latch and the underflow of timer X are performed at the same timing.

•Real Time Port Control

While the real time port function is valid, data for the real time port are output from ports $P8_5$ and $P8_6$ each time the timer X underflows. (However, if the real time port control bit is changed from "0" to "1," data are output without the timer X.) When the data for the real time port is changed while the real time port function is valid, the changed data are output at the next underflow of timer X.
Before using this function, set the corresponding port direction registers to output mode.

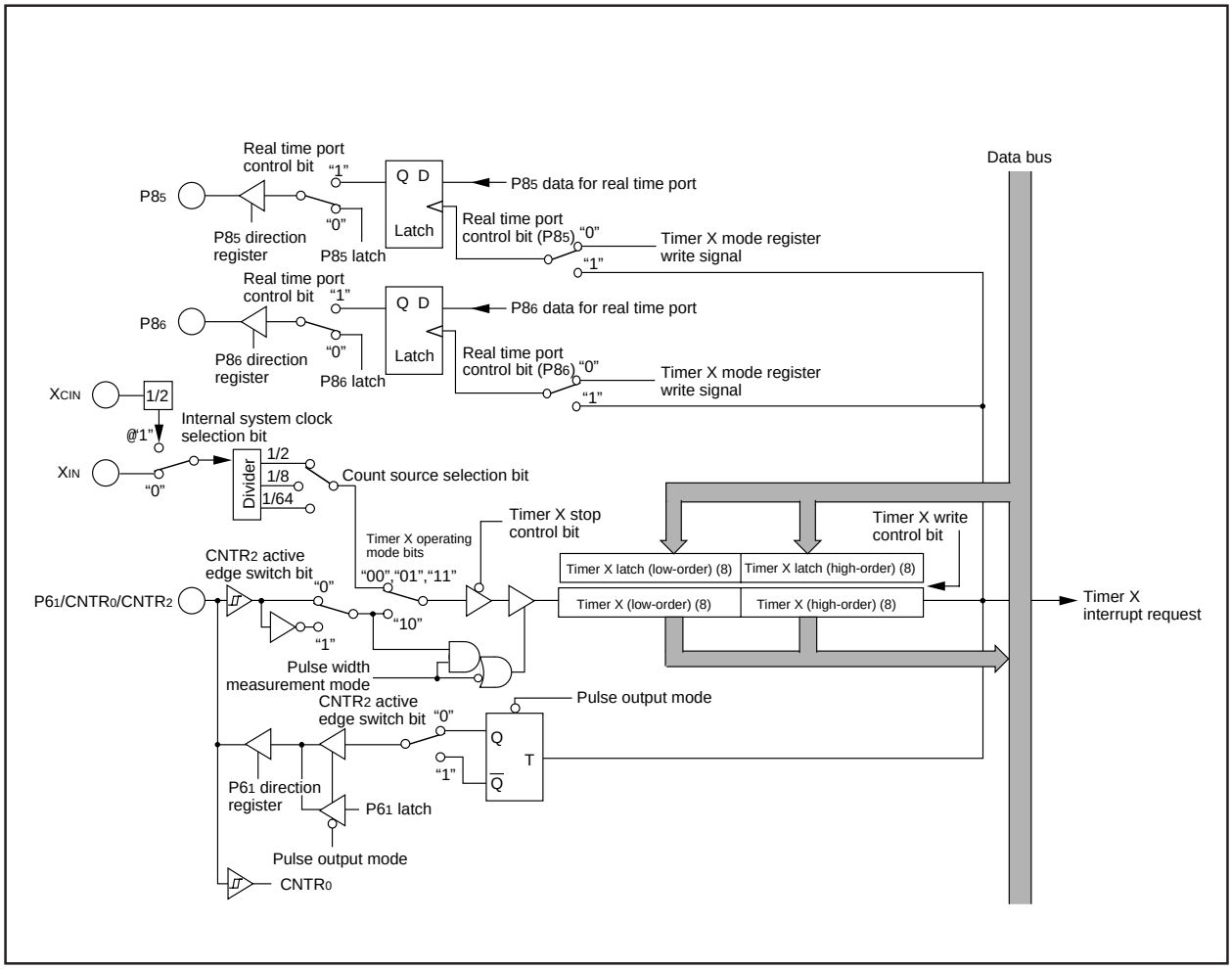


Fig. 19 Block diagram of timer X

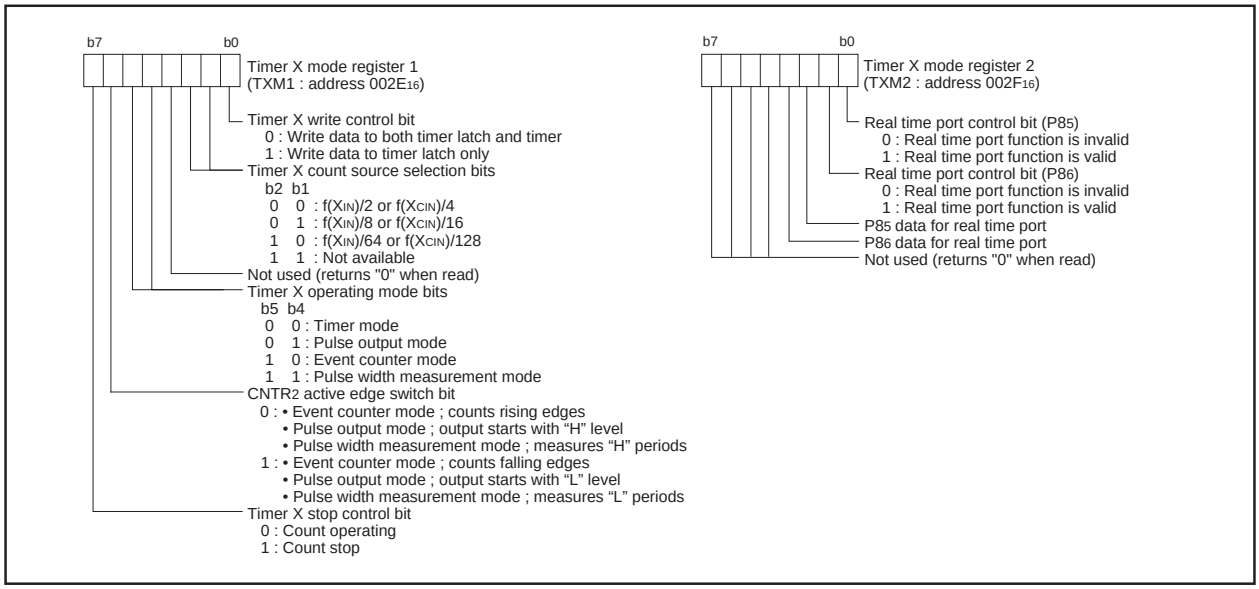


Fig. 20 Structure of timer X related registers

Serial I/O

●Serial I/O1

Serial I/O1 is used as the clock synchronous serial I/O and has an ordinary mode and an automatic transfer mode. In the automatic transfer mode, serial transfer is performed through the serial I/O automatic transfer RAM which has up to 256 bytes (addresses $0F00_{16}$ to $0FFF_{16}$: addresses $0F60_{16}$ to $0FFF_{16}$ are also used as FLD automatic display RAM).

The $P6_2/S_{RDY1}/AN_8$, $P6_4/INT_4/S_{BUSY1}/AN_{10}$, and $P6_5/S_{STB1}/AN_{11}$ pins each have a handshake I/O signal function and can select either "H" active or "L" active for active logic.

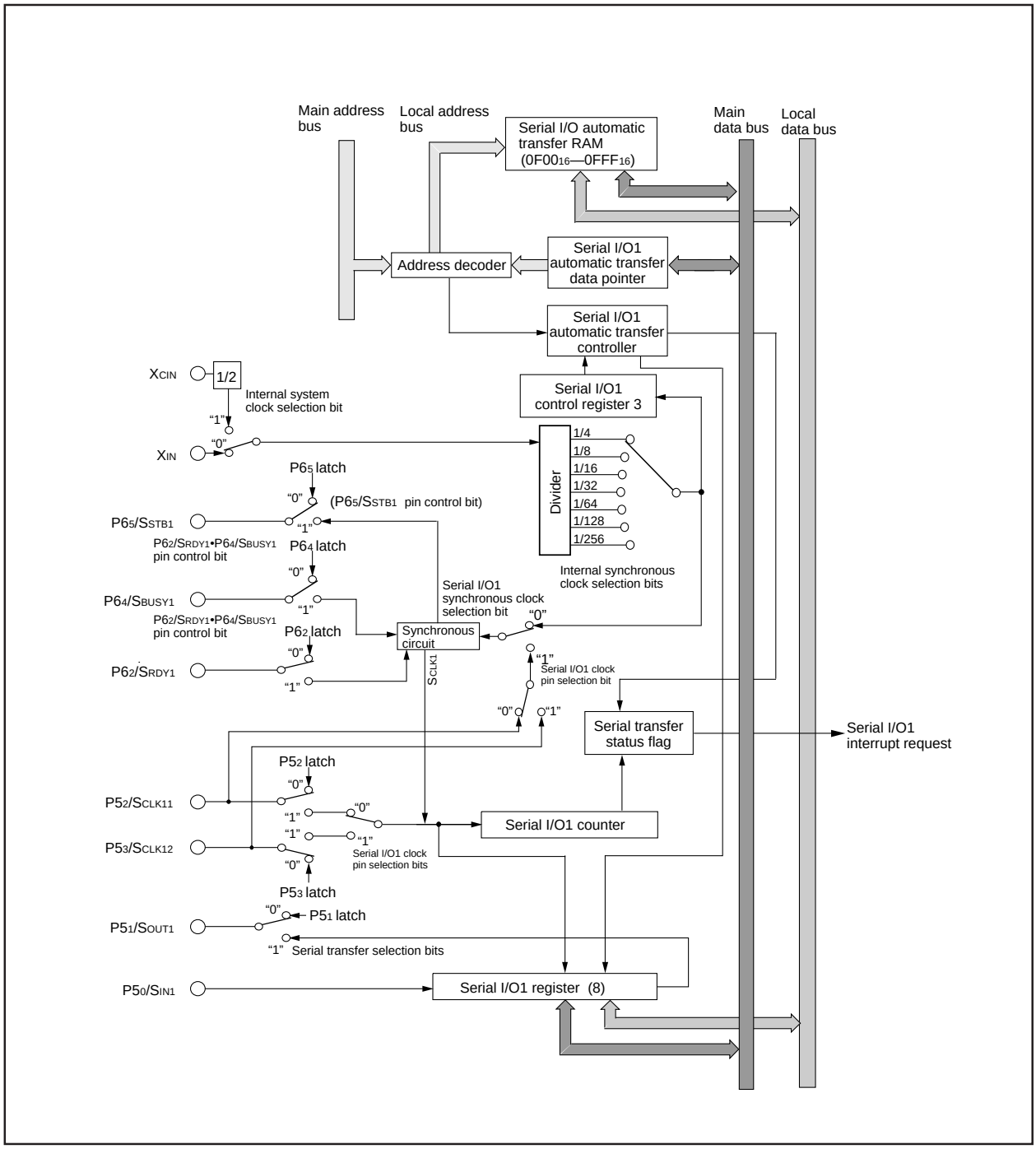
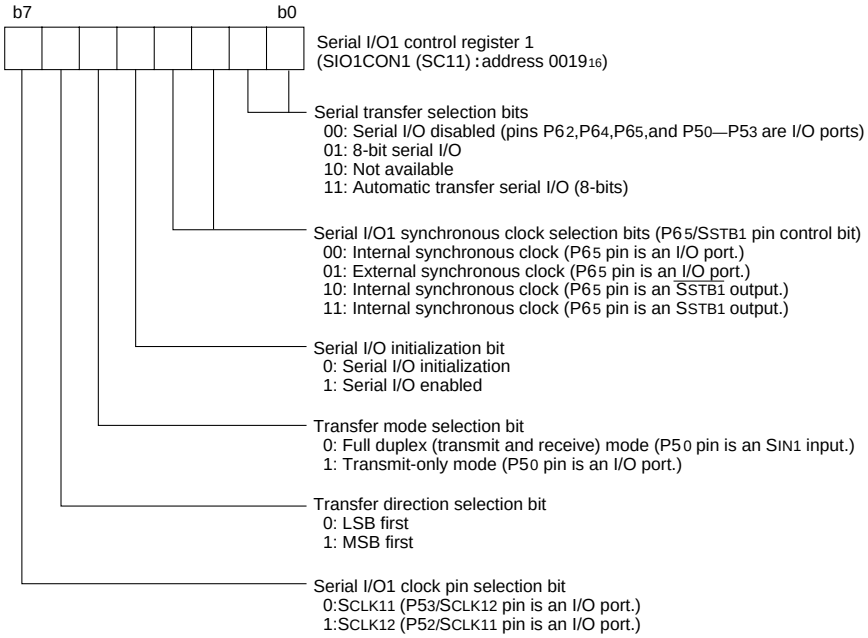


Fig. 21 Block diagram of serial I/O1



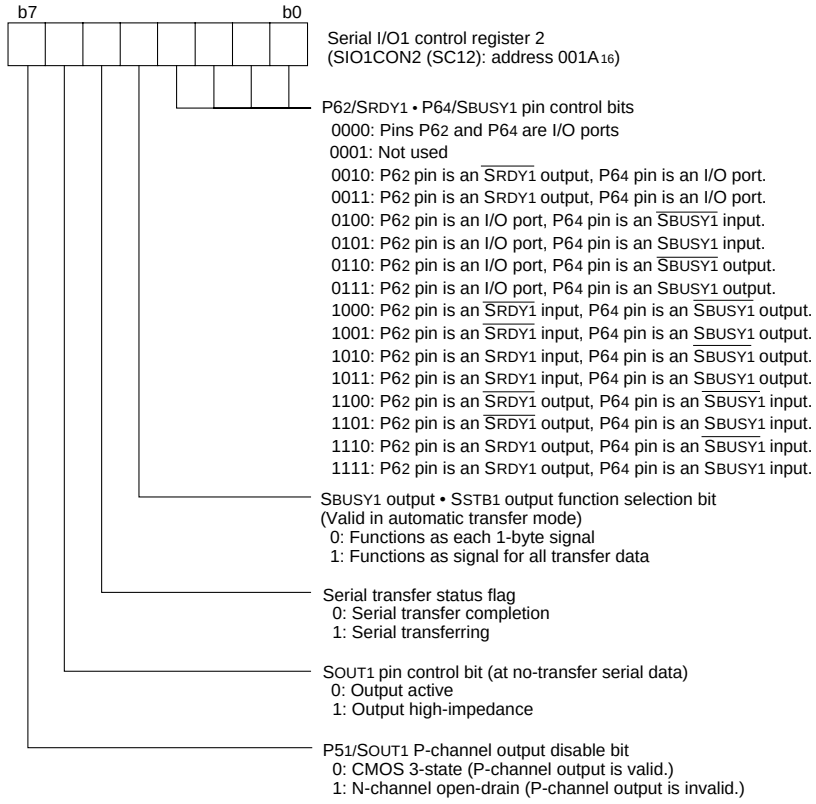


Fig. 22 Structure of serial I/O1 control registers 1, 2

(1) Serial I/O1 Operation

Either the internal synchronous clock or external synchronous clock can be selected by the serial I/O1 synchronous clock selection bits (b2 and b3 of address 0019_{16}) of serial I/O1 control register 1 as synchronous clock for serial transfer.

The internal synchronous clock has a built-in dedicated divider where 7 different clocks are selected by the internal synchronous clock selection bits (b5, b6 and b7 of address $001C_{16}$) of serial I/O1 control register 3.

The $P6_2$/S_{RDY1}/AN_8, $P6_4$/INT_4/S_{BUSY1}/AN_{10}, and $P6_5$/S_{STB1}/AN_{11} pins each select either I/O port or handshake I/O signal by the serial I/O1 synchronous clock selection bits (b2 and b3 of address 0019_{16}) of serial I/O1 control register 1 as well as the $P6_2$/S_{RDY1} • $P6_4$/S_{BUSY1} pin control bits (b0 to b3 of address $001A_{16}$) of serial I/O1 control register 2.

For the S_{OUT1} being used as an output pin, either CMOS output or N-channel open-drain output is selected by the $P5_1$/S_{OUT1} P-channel output disable bit (b7 of address $001A_{16}$) of serial I/O1 control register 2.

Either output active or high-impedance can be selected as a S_{OUT1} pin state at serial non-transfer by the S_{OUT1} pin control bit (b6 of address $001A_{16}$) of serial I/O1 control register 2. However, when the external synchronous clock is selected, perform the following setup to put the S_{OUT1} pin into a high-impedance state.

When the S_{CLK1} input is "H" after completion of transfer, set the S_{OUT1} pin control bit to "1."

When the S_{CLK1} input goes to "L" after the start of the next serial transfer, the S_{OUT1} pin control bit is automatically reset to "0" and put into an output active state.

Regardless of whether the internal synchronous clock or external synchronous clock is selected, the full duplex mode and the transmit-only mode are available for serial transfer, one of which is selected by the transfer mode selection bit (b5 of address 0019_{16}) of serial I/O1 control register 1.

Either LSB first or MSB first is selected for the I/O sequence of the serial transfer bit strings by the transfer direction selection bit (b6 of address 0019_{16}) of serial I/O1 control register 1.

When using serial I/O1, first select either 8-bit serial I/O or automatic transfer serial I/O by the serial transfer selection bits (b0 and b1 of address 0019_{16}) of serial I/O1 control register 1, after completion of the above bit setup. Next, set the serial I/O initialization bit (b4 of address 0019_{16}) of serial I/O1 control register 1 to "1" (Serial I/O enable) .

When stopping serial transfer while data is being transferred, regardless of whether the internal or external synchronous clock is selected, reset the serial I/O initialization bit (b4) to "0."

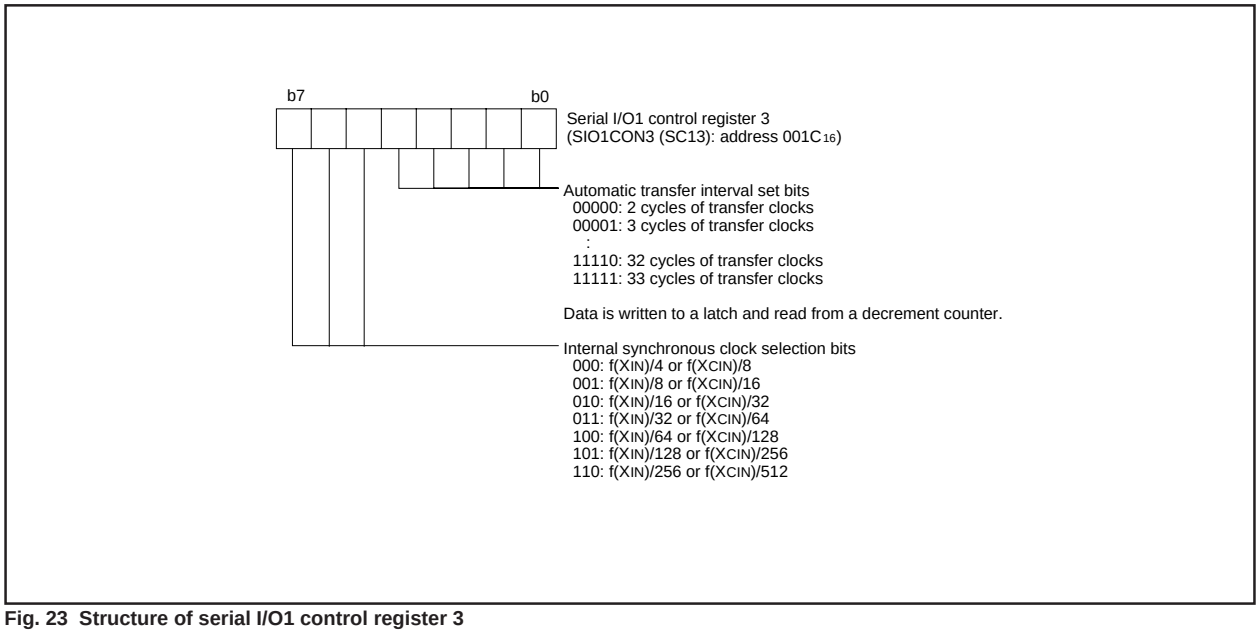


Fig. 23 Structure of serial I/O1 control register 3

(2) 8-bit Serial I/O Mode

Address 001B16 is assigned to the serial I/O1 register.

When the internal synchronous clock is selected, a serial transfer of the 8-bit serial I/O is started by a write signal to the serial I/O1 register (address 001B16).

The serial transfer status flag (b5 of address 001A16) of serial I/O1 control register 2 indicates the shift register status of serial I/O1, and is set to "1" by writing into the serial I/O1 register, which becomes a transfer start trigger and reset to "0" after completion of 8-bit transfer. At the same time, a serial I/O1 interrupt request occurs.

When the external synchronous clock is selected, the contents of the serial I/O1 register are continuously shifted while transfer clocks are input to SCLK1. Therefore, the clock needs to be controlled externally.

(3) Automatic Transfer Serial I/O Mode

The serial I/O1 automatic transfer controller controls the write and read operations of the serial I/O1 register, so the function of address 001B16 is used as a transfer counter (1-byte units).

When performing serial transfer through the serial I/O automatic transfer RAM (addresses 0F0016 to 0FFF16), it is necessary to set the serial I/O1 automatic transfer data pointer (address 001816) beforehand.

Input the low-order 8 bits of the first data store address to be serially transferred to the automatic transfer data pointer set bits.

When the internal synchronous clock is selected, the transfer interval for each 1-byte data can be set by the automatic transfer interval set bits (b0 to b4 of address 001C16) of serial I/O1 control register 3 in the following cases:

1. When using no handshake signal
2. When using the SRDY1 output, SBUSY1 output, and SSTB1 output of the handshake signal independently
3. When using a combination of SRDY1 output and SSTB1 output or a combination of SBUSY1 output and SSTB1 output of the handshake signal

It is possible to select one of 32 different values, namely 2 to 33 cycles of the transfer clock, as a setting value.

When using the SBUSY1 output and selecting the SBUSY1 output • SSTB1 output function selection bit (b4 of address 001A16) of serial I/O1 control register 2 as the signal for all transfer data, provided that the automatic transfer interval setting is valid, a transfer interval is placed before the start of transmission/reception of the first data and after the end of transmission/reception of the last data.

For SSTB1 output, regardless of the contents of the SBUSY1 output • SSTB1 output function selection bit (b4), the transfer interval for each 1-byte data is longer than the set value by 2 cycles.

Furthermore, when using a combination of SBUSY1 output and SSTB1 output as a signal for all transfer data, the transfer interval after the end of transmission/reception of the last data is longer than the set value by 2 cycles.

When the external synchronous clock is selected, automatic transfer interval setting is disabled.

After completion of the above bit setup, if the internal synchronous clock is selected, automatic serial transfer is started by writing the value of "number of transfer bytes - 1" into the transfer counter (address 001B16).

When the external synchronous clock is selected, write the value of "number of transfer bytes - 1" into the transfer counter and input an internal system clock interval of 5 cycles or more. After that, input transfer clock to SCLK1.

As a transfer interval for each 1-byte data transfer, input an internal system clock interval of 5 cycles or more from the clock rise time of the last bit.

Regardless of whether the internal or external synchronous clock is selected, the automatic transfer data pointer and the transfer counter are decremented after each 1-byte data is received and then written into the automatic transfer RAM. The serial transfer status flag (b5 of address 001A16) is set to "1" by writing data into the transfer counter. Writing data becomes a transfer start trigger, and the serial transfer status flag is reset to "0" after the last data is written into the automatic transfer RAM. At the same time, a serial I/O1 interrupt request occurs.

The values written in the automatic transfer data pointer set bits (b0 to b7 of address 001816) and the automatic transfer interval set bits (b0 to b4 of address 001C16) are held in the latch.

When data is written into the transfer counter, the values latched in the automatic transfer data pointer set bits (b0 to b7) and the automatic transfer interval set bits (b0 to b4) are transferred to the decrement counter.

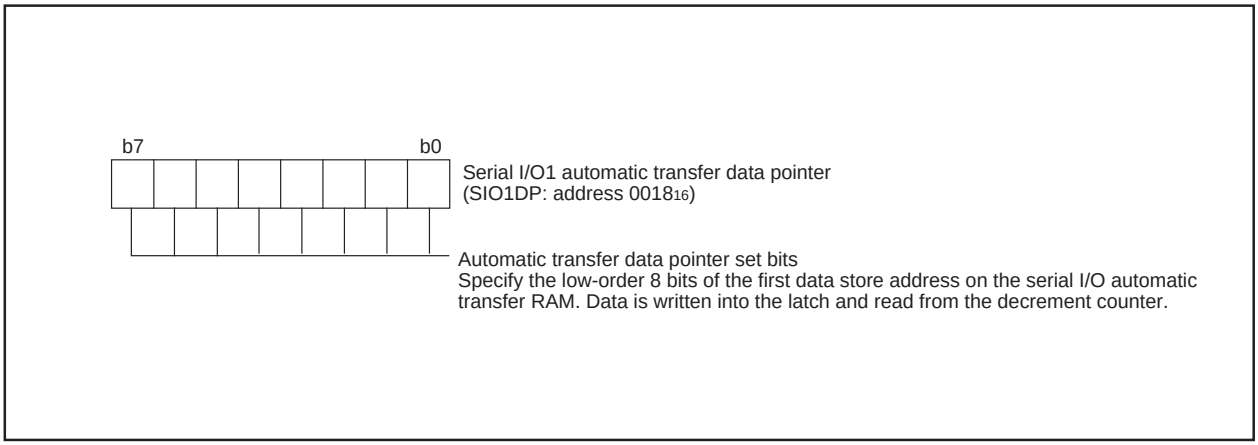


Fig. 24 Structure of serial I/O1 automatic transfer data pointer

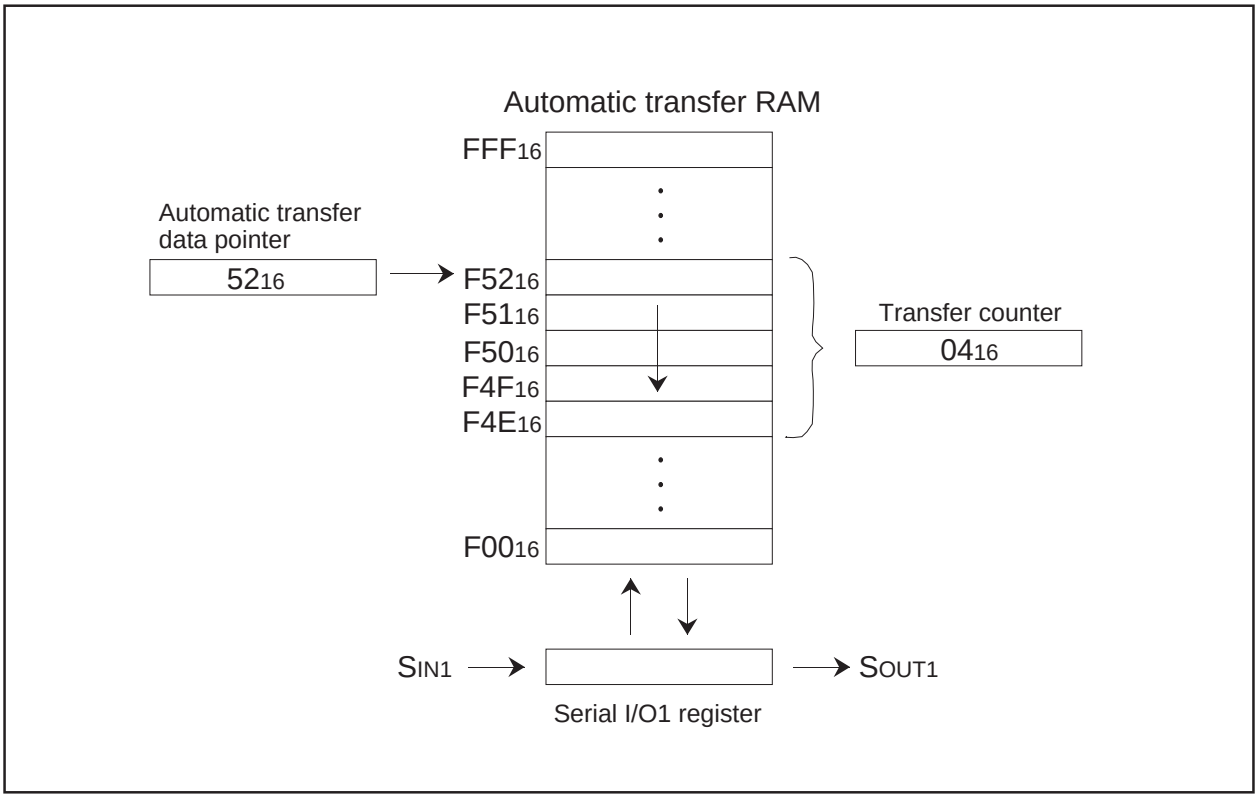


Fig. 25 Automatic transfer serial I/O operation

(4) Handshake Signal

1. SSTB1 output signal

The SSTB1 output is a signal to inform an end of transmission/reception to the serial transfer destination . The SSTB1 output signal can be used only when the internal synchronous clock is selected.

In the initial status, namely, in the status in which the serial I/O initialization bit (b4) is reset to "0," the SSTB1 output goes to "L," or the $\overline{\text{SSTB1}}$ output goes to "H."

At the end of transmit/receive operation, when the data of the serial I/O1 register is all output from SOUT1, pulses are output in the period of 1 cycle of the transfer clock so as to cause the SSTB1 output to go "H" or the $\overline{\text{SSTB1}}$ output to go "L." After that, each pulse is returned to the initial status in which SSTB1 output goes to "L" or the $\overline{\text{SSTB1}}$ output goes to "H."

Furthermore, after 1 cycle, the serial transfer status flag (b5) is reset to "0."

In the automatic transfer serial I/O mode, whether the SSTB1 output is to be active at an end of each 1-byte data or after completion of transfer of all data can be selected by the SBUSY1 output • SSTB1 output function selection bit (b4 of address $001A_{16}$) of serial I/O1 control register 2.

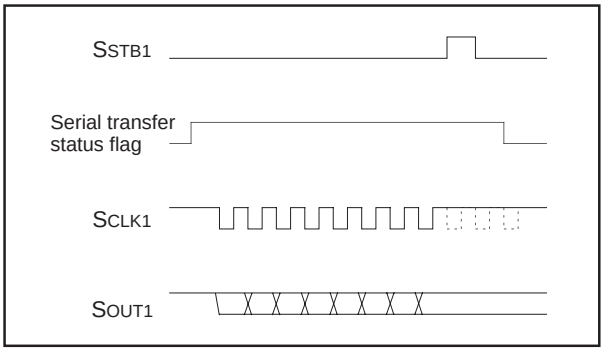


Fig. 26 SSTB1 output operation

2. SBUSY1 input signal

The SBUSY1 input is a signal which receives a request for a stop of transmission/reception from the serial transfer destination.

When the internal synchronous clock is selected, input an "H" level signal into the SBUSY1 input and an "L" level signal into the $\overline{\text{SBUSY1}}$ input in the initial status in which transfer is stopped.

When starting a transmit/receive operation, input an "L" level signal into the SBUSY1 input and an "H" level signal into the $\overline{\text{SBUSY1}}$ input in the period of 1.5 cycles or more of the transfer clock. Then, transfer clocks are output from the SCLK1 output.

When an "H" level signal is input into the SBUSY1 input and an "L" level signal into the $\overline{\text{SBUSY1}}$ input after a transmit/receive operation is started, this transmit/receive operation are not stopped immediately and the transfer clocks from the SCLK1 output is not stopped until the specified number of bits are transmitted and received.

The handshake unit of the 8-bit serial I/O is 8 bits and that of the automatic transfer serial I/O is 8 bits.

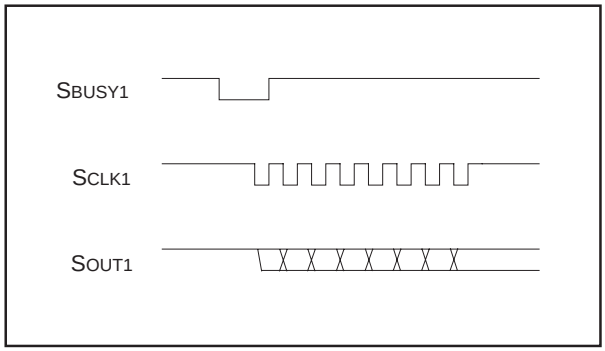


Fig. 27 SBUSY1 input operation (internal synchronous clock)

When the external synchronous clock is selected, input an "H" level signal into the SBUSY1 input and an "L" level signal into the $\overline{\text{SBUSY1}}$ input in the initial status in which transfer is stopped. At this time, the transfer clocks to be input in SCLK1 become invalid.

During serial transfer, the transfer clocks to be input in SCLK1 become valid, enabling a transmit/receive operation, while an "L" level signal is input into the SBUSY1 input and an "H" level signal is input into the $\overline{\text{SBUSY1}}$ input.

When changing the input values in the SBUSY1 input and the $\overline{\text{SBUSY1}}$ input at these operations, change them when the SCLK1 input is in a high state.

When the high impedance of the SOUT1 output is selected by the SOUT1 pin control bit (b6), the SOUT1 output becomes active, enabling serial transfer by inputting a transfer clock to SCLK1, while an "L" level signal is input into the SBUSY1 input and an "H" level signal is input into the $\overline{\text{SBUSY1}}$ input.

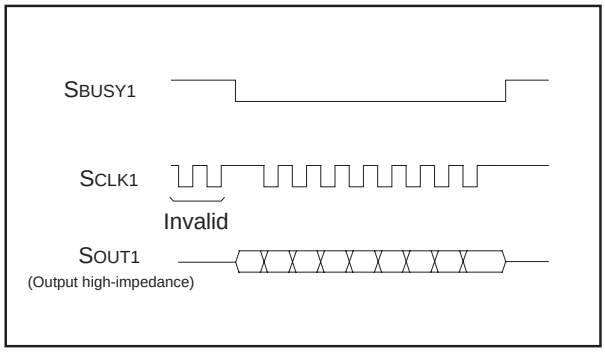


Fig. 28 SBUSY1 input operation (external synchronous clock)

3. SBUSY1 output signal

The SBUSY1 output is a signal which requests a stop of transmission/reception to the serial transfer destination. In the automatic transfer serial I/O mode, regardless of the internal or external synchronous clock, whether the SBUSY1 output is to be active at transfer of each 1-byte data or during transfer of all data can be selected by the SBUSY1 output • SSTB1 output function selection bit (b4).

In the initial status, the status in which the serial I/O initialization bit (b4) is reset to "0," the SBUSY1 output goes to "H" and the $\overline{\text{SBUSY1}}$ output goes to "L."

When the internal synchronous clock is selected, in the 8-bit serial I/O mode and the automatic transfer serial I/O mode (SBUSY1 output function outputs in 1-byte units), the SBUSY1 output goes to "L" and the $\overline{\text{SBUSY1}}$ output goes to "H" before 0.5 cycle (transfer clock) of the timing at which the transfer clock from the SCLK1 output goes to "L" at a start of transmit/receive operation.

In the automatic transfer serial I/O mode (the SBUSY1 output function outputs all transfer data), the SBUSY1 output goes to "L" and the $\overline{\text{SBUSY1}}$ output goes to "H" when the first transmit data is written into the serial I/O1 register (address $001B_{16}$).

When the external synchronous clock is selected, the SBUSY1 output goes to "L" and the $\overline{\text{SBUSY1}}$ output goes to "H" when transmit data is written into the serial I/O1 register to start a transmit operation, regardless of the serial I/O transfer mode.

At termination of transmit/receive operation, the SBUSY1 output returns to "H" and the $\overline{\text{SBUSY1}}$ output returns to "L", the initial status, when the serial transfer status flag is set to "0", regardless of whether the internal or external synchronous clock is selected.

Furthermore, in the automatic transfer serial I/O mode (SBUSY1 output function outputs in 1-byte units), the SBUSY1 output goes to "H" and the $\overline{\text{SBUSY1}}$ output goes to "L" each time 1-byte of receive data is written into the automatic transfer RAM.

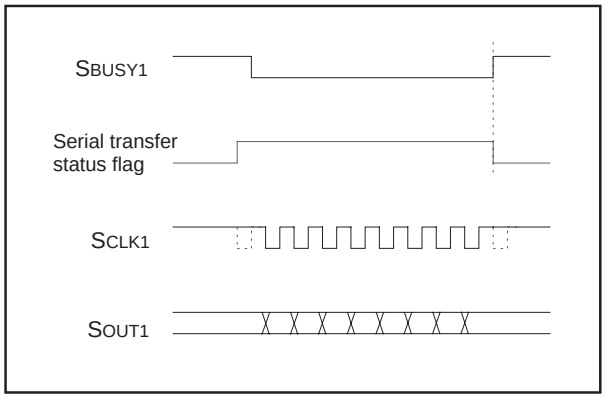


Fig. 29 SBUSY1 output operation (internal synchronous clock, 8-bits serial I/O)

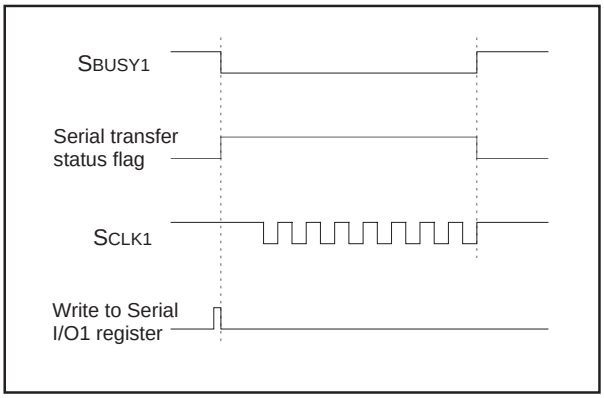


Fig. 30 SBUSY1 output operation (external synchronous clock, 8-bits serial I/O)

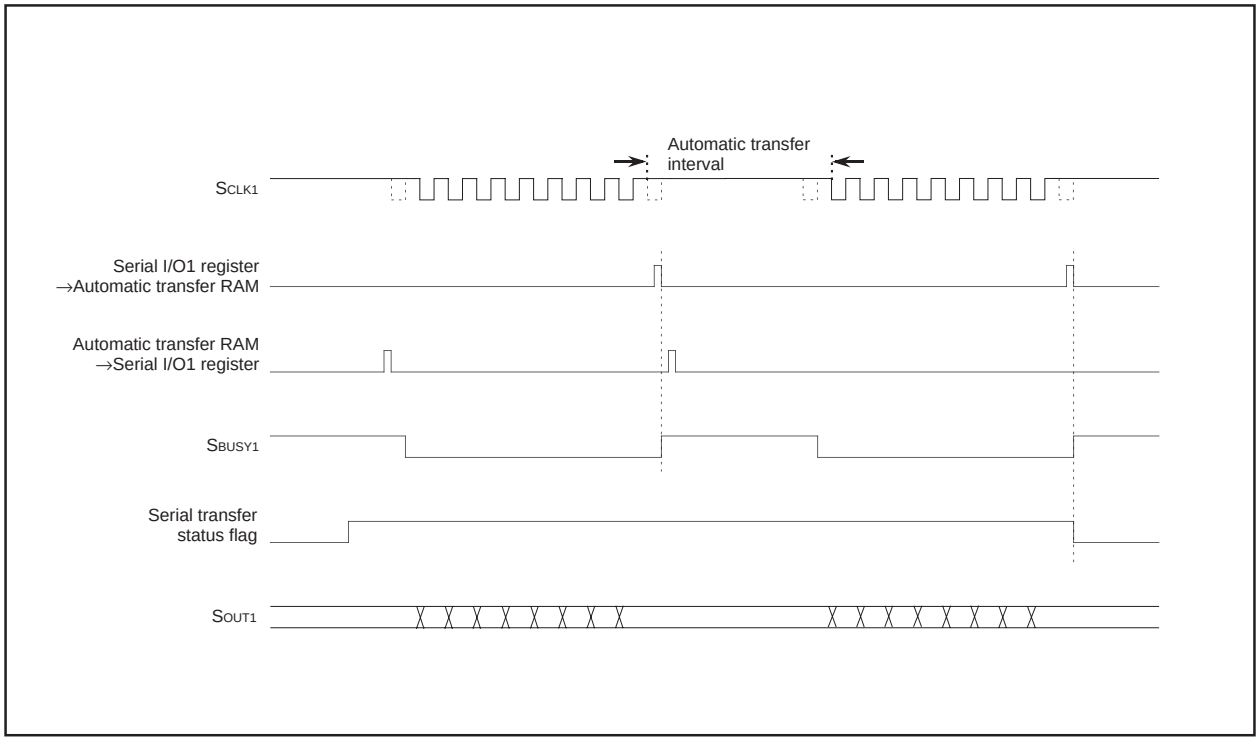


Fig. 31 SBUSY1 output operation in automatic transfer serial I/O mode (internal synchronous clock, SBUSY1 output function outputs each 1-byte)

4. SRDY1 output signal

The SRDY1 output is a transmit/receive enable signal which informs the serial transfer destination that transmit/receive is ready. In the initial status, when the serial I/O initialization bit (b4) is reset to "0," the SRDY1 output goes to "L" and the $\overline{\text{SRDY1}}$ output goes to "H". After transmitted data is stored in the serial I/O1 register (address $001B_{16}$) and a transmit/receive operation becomes ready, the SRDY1 output goes to "H" and the $\overline{\text{SRDY1}}$ output goes to "L". When a transmit/receive operation is started and the transfer clock goes to "L", the SRDY1 output goes to "L" and the $\overline{\text{SRDY1}}$ output goes to "H".

5. SRDY1 input signal

The SRDY1 input signal becomes valid only when the SRDY1 input and the SBUSY1 output are used. The SRDY1 input is a signal for receiving a transmit/receive ready completion signal from the serial transfer destination.

When the internal synchronous clock is selected, input a low level signal into the SRDY1 input and a high level signal into the $\overline{\text{SRDY1}}$ input in the initial status in which the transfer is stopped.

When an "H" level signal is input into the SRDY1 input and an "L" level signal is input into the $\overline{\text{SRDY1}}$ input for a period of 1.5 cycles or more of transfer clock, transfer clocks are output from the SCLK1 output and a transmit/receive operation is started.

After the transmit/receive operation is started and an "L" level signal is input into the SRDY1 input and an "H" level signal into the $\overline{\text{SRDY1}}$ input, this operation cannot be immediately stopped.

After the specified number of bits are transmitted and received, the transfer clocks from the SCLK1 output is stopped. The handshake unit of the 8-bit serial I/O and that of the automatic transfer serial I/O are of 8 bits.

When the external synchronous clock is selected, the SRDY1 input becomes one of the triggers to output the SBUSY1 signal.

To start a transmit/receive operation (SBUSY1 output: "L," $\overline{\text{SBUSY1}}$ output: "H"), input an "H" level signal into the SRDY1 input and an "L" level signal into the $\overline{\text{SRDY1}}$ input, and also write transmit data into the serial I/O1 register.

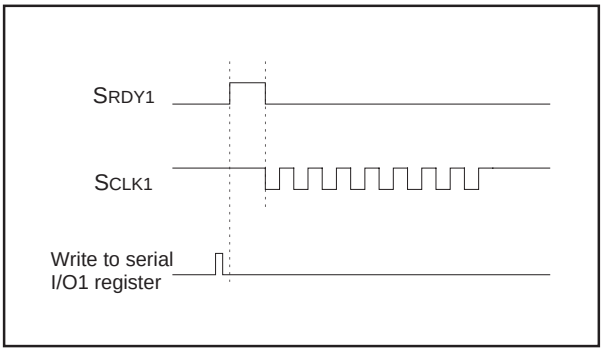


Fig. 32 SRDY1 output operation

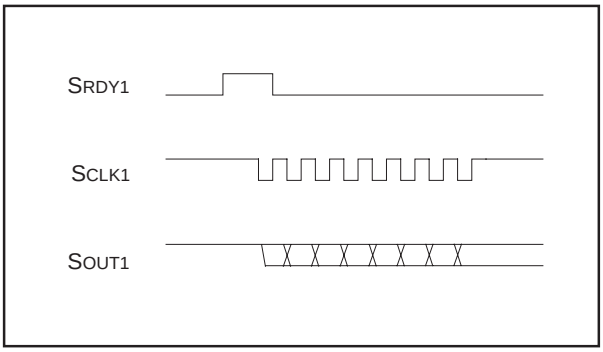


Fig. 33 SRDY1 input operation (internal synchronous clock)

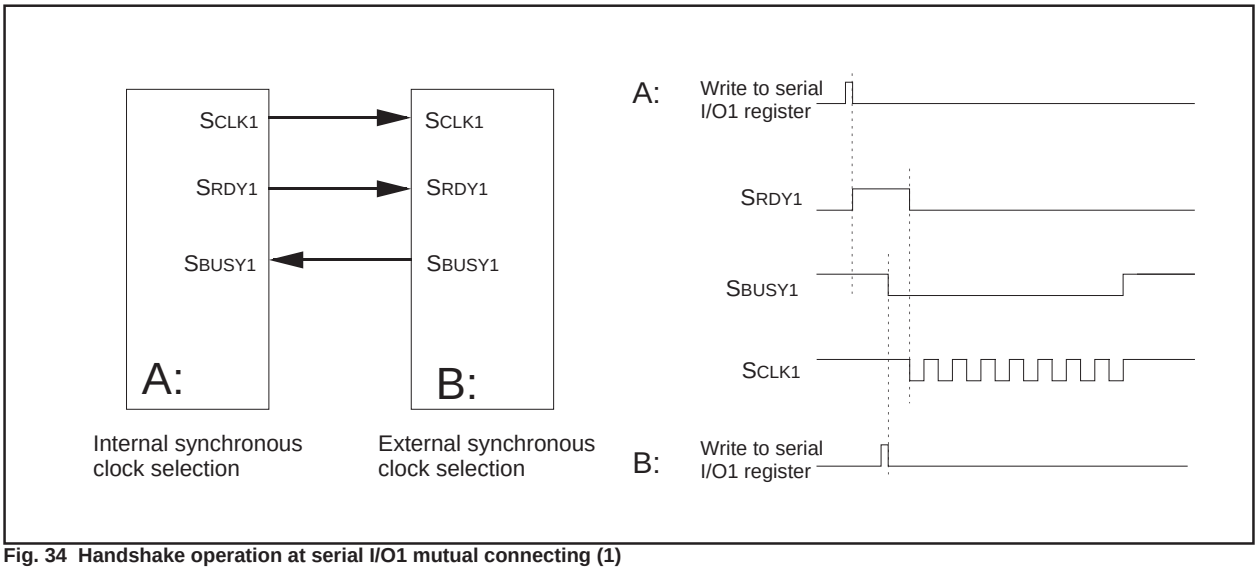


Fig. 34 Handshake operation at serial I/O1 mutual connecting (1)

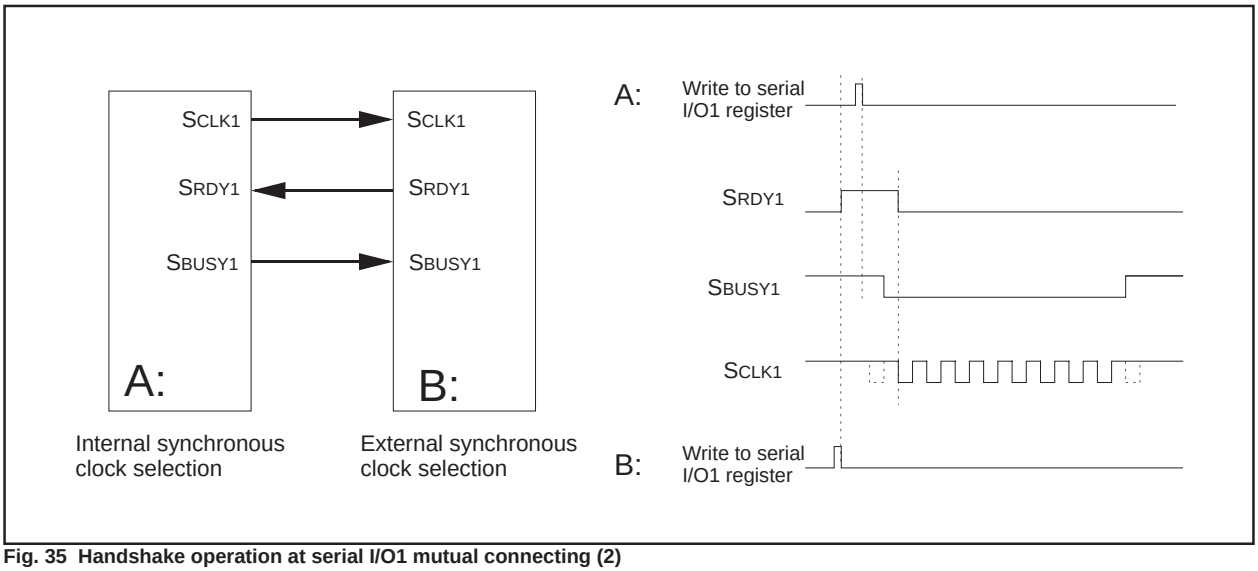


Fig. 35 Handshake operation at serial I/O1 mutual connecting (2)

●Serial I/O2

Serial I/O2 can be used as either clock synchronous or asynchronous (UART) serial I/O. A dedicated timer (baud rate generator) is also provided for baud rate generation during serial I/O2 operation.

(1) Clock Synchronous Serial I/O Mode

The clock synchronous serial I/O mode can be selected by setting the serial I/O2 mode selection bit (b6) of the serial I/O2 control register (address 001D16) to "1." For clock synchronous serial I/O, the transmitter and the receiver must use the same clock for serial I/O2 operation. If an internal clock is used, transmit/receive is started by a write signal to the serial I/O2 transmit/receive buffer register (TB/RB) (address 001F16).

When P57 (SCLK22) is selected as a clock I/O pin, $\overline{\text{SRDY2}}$ output function is invalid, and P56 (SCLK21) is used as an I/O port.

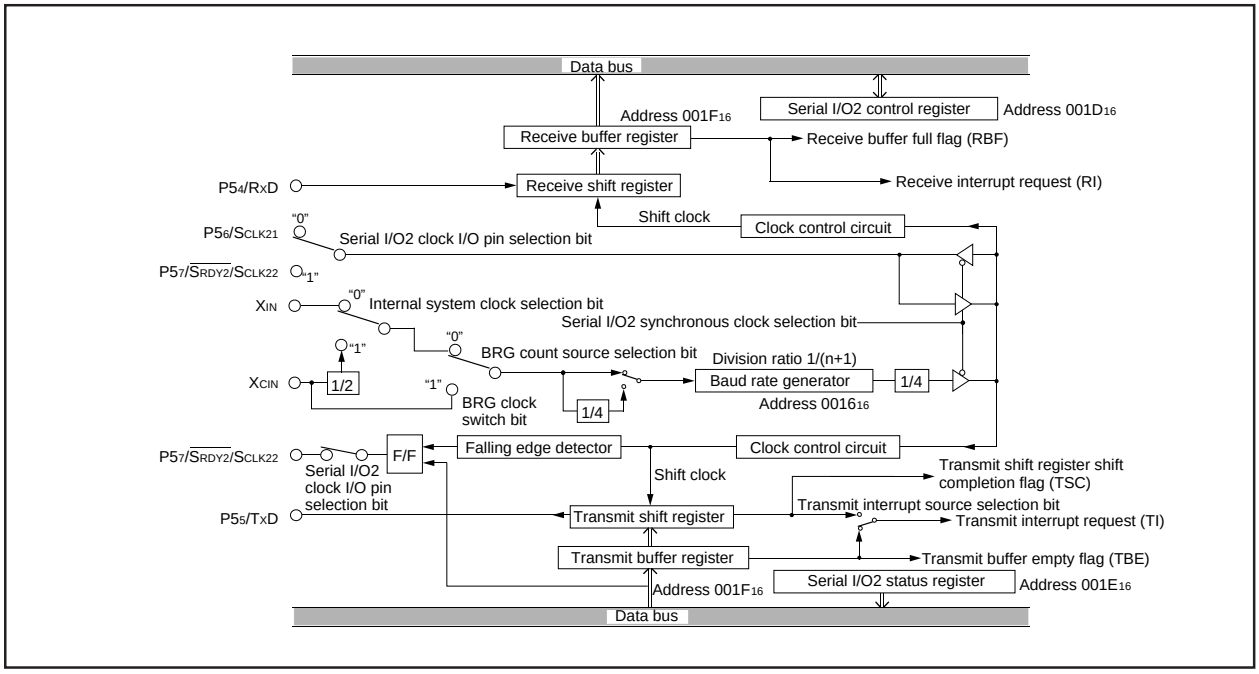
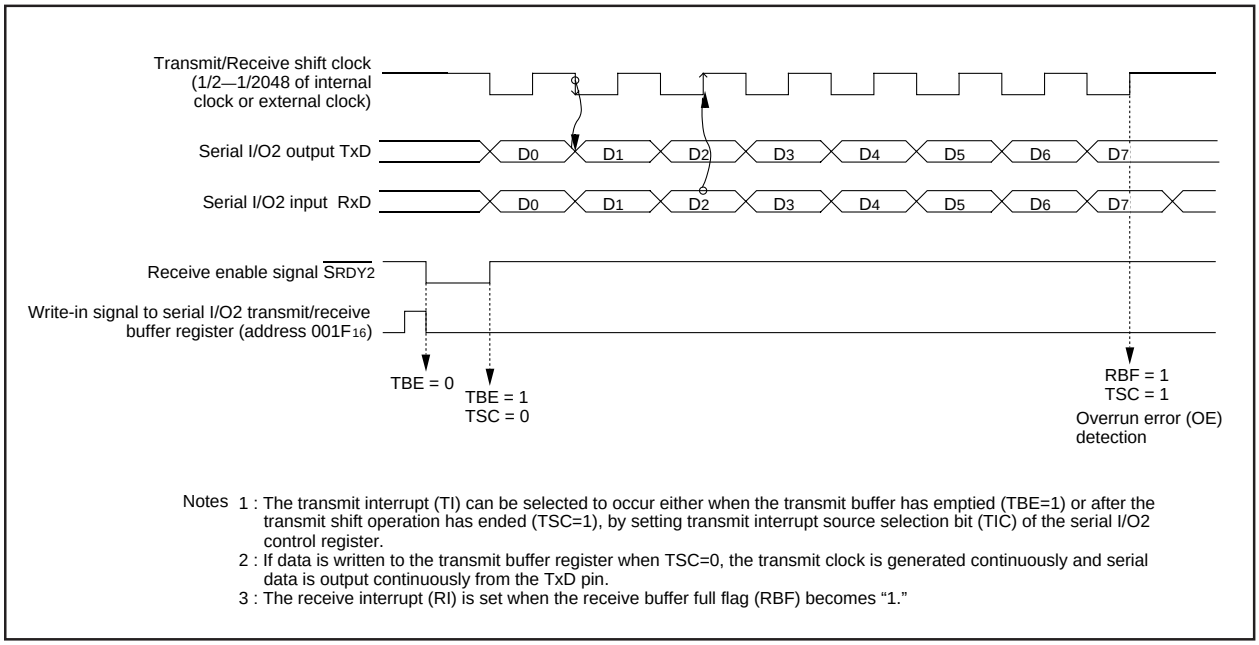


Fig. 36 Block diagram of clock synchronous serial I/O2



Notes 1 : The transmit interrupt (TI) can be selected to occur either when the transmit buffer has emptied (TBE=1) or after the transmit shift operation has ended (TSC=1), by setting transmit interrupt source selection bit (TIC) of the serial I/O2 control register.
2 : If data is written to the transmit buffer register when TSC=0, the transmit clock is generated continuously and serial data is output continuously from the TxD pin.
3 : The receive interrupt (RI) is set when the receive buffer full flag (RBF) becomes "1."

Fig. 37 Operation of clock synchronous serial I/O2 function

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

The asynchronous serial I/O (UART) mode can be selected by clearing the serial I/O2 mode selection bit (b6) of the serial I/O2 control register (address 001D16) to "0." Eight serial data transfer formats can be selected and the transfer formats used by the transmitter and receiver must be identical.

The transmit and receive shift registers each have a buffer (the two buffers have the same address in memory). Since the shift register cannot be written to or read from directly, transmit data is written to the transmit buffer, and receive data is read from the receive buffer. The transmit buffer can also hold the next data to be transmitted, and the receive buffer can receive 2-byte data continuously.

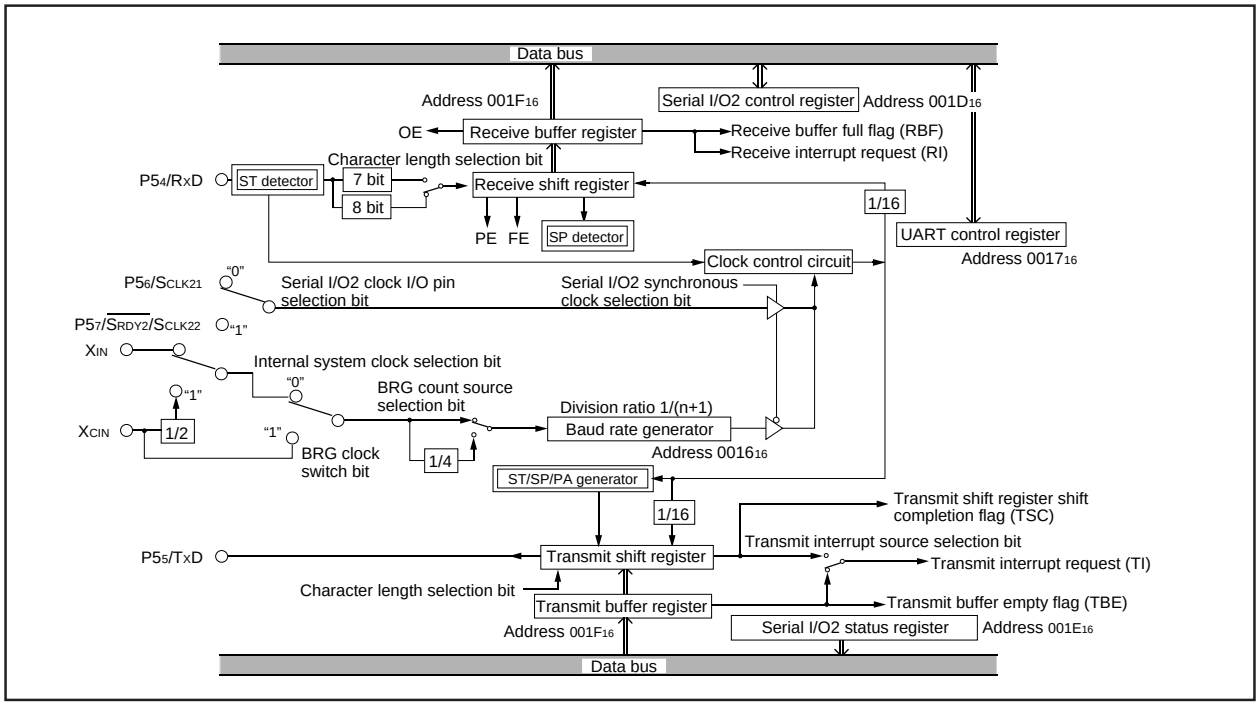


Fig. 38 Block diagram of UART serial I/O2

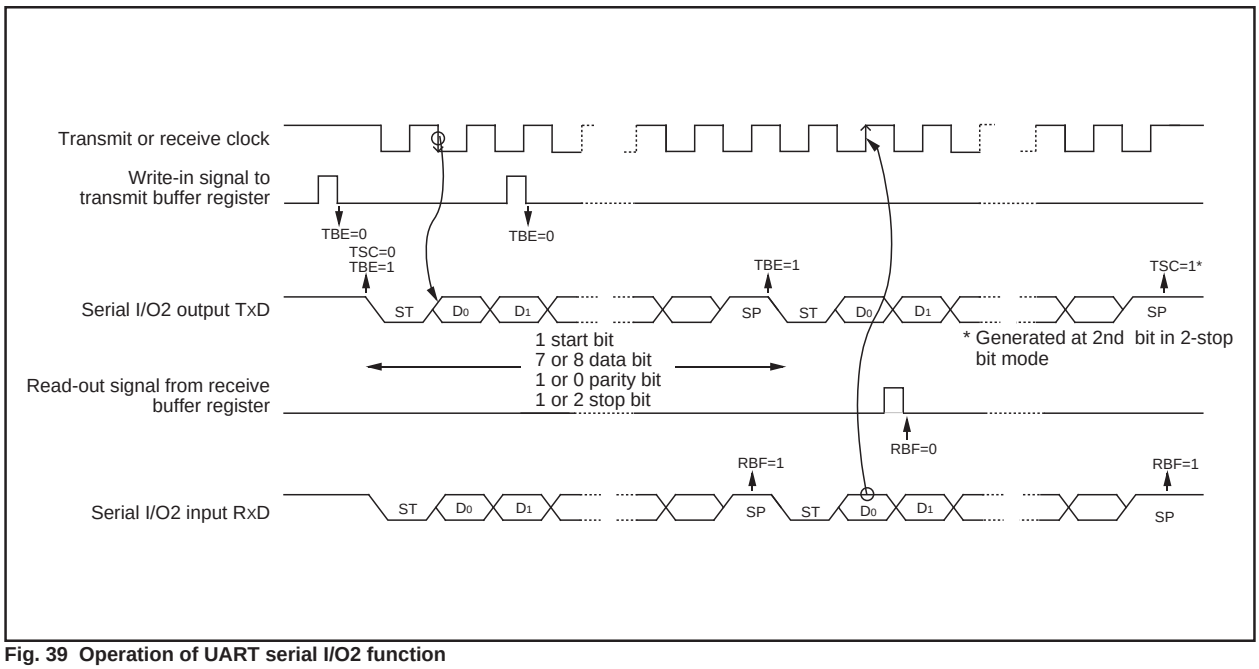


Fig. 39 Operation of UART serial I/O2 function

[Serial I/O2 Control Register] SIO2CON (001D16)

The serial I/O2 control register contains eight control bits for serial I/O2 functions.

[UART Control Register] UARTCON (001716)

This is a 7 bit register containing four control bits, which are valid when UART is selected, two control bits, which are valid when using serial I/O2, and one control bit, which is always valid.
Data format of serial data receive/transfer and the output structure of the P55/TxD pin, etc. are set by this register.

[Serial I/O2 Status Register] SIO2STS (001E16)

The read-only serial I/O2 status register consists of seven flags (b0 to b6) which indicate the operating status of the serial I/O2 function and various errors. Three of the flags (b4 to b6) are only valid in the UART mode. The receive buffer full flag (b1) is cleared to “0” when the receive buffer is read.
The error detection is performed at the same time data is transferred from the receive shift register to the receive buffer register, and the receive buffer full flag is set. A writing to the serial I/O2 status register clears error flags OE, PE, FE, and SE (b3 to b6, respectively). Writing “0” to the serial I/O2 enable bit (SIOE : b7 of the serial I/O2 control register) also clears all the status flags, including the error flags.
All bits of the serial I/O2 status register are initialized to “0” at reset, but if the transmit enable bit (b4) of the serial I/O2 control register has been set to “1,” the transmit shift register shift completion flag (b2) and the transmit buffer empty flag (b0) become “1.”

[Serial I/O2 Transmit Buffer Register/Receive Buffer Register] TB/RB (001F16)

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

[Baud Rate Generator] BRG (001616)

The baud rate generator determines the baud rate for serial transfer. With the 8-bit counter having a reload register, the baud rate generator divides the frequency of the count source by 1/(n+1), where n is the value written to the baud rate generator.

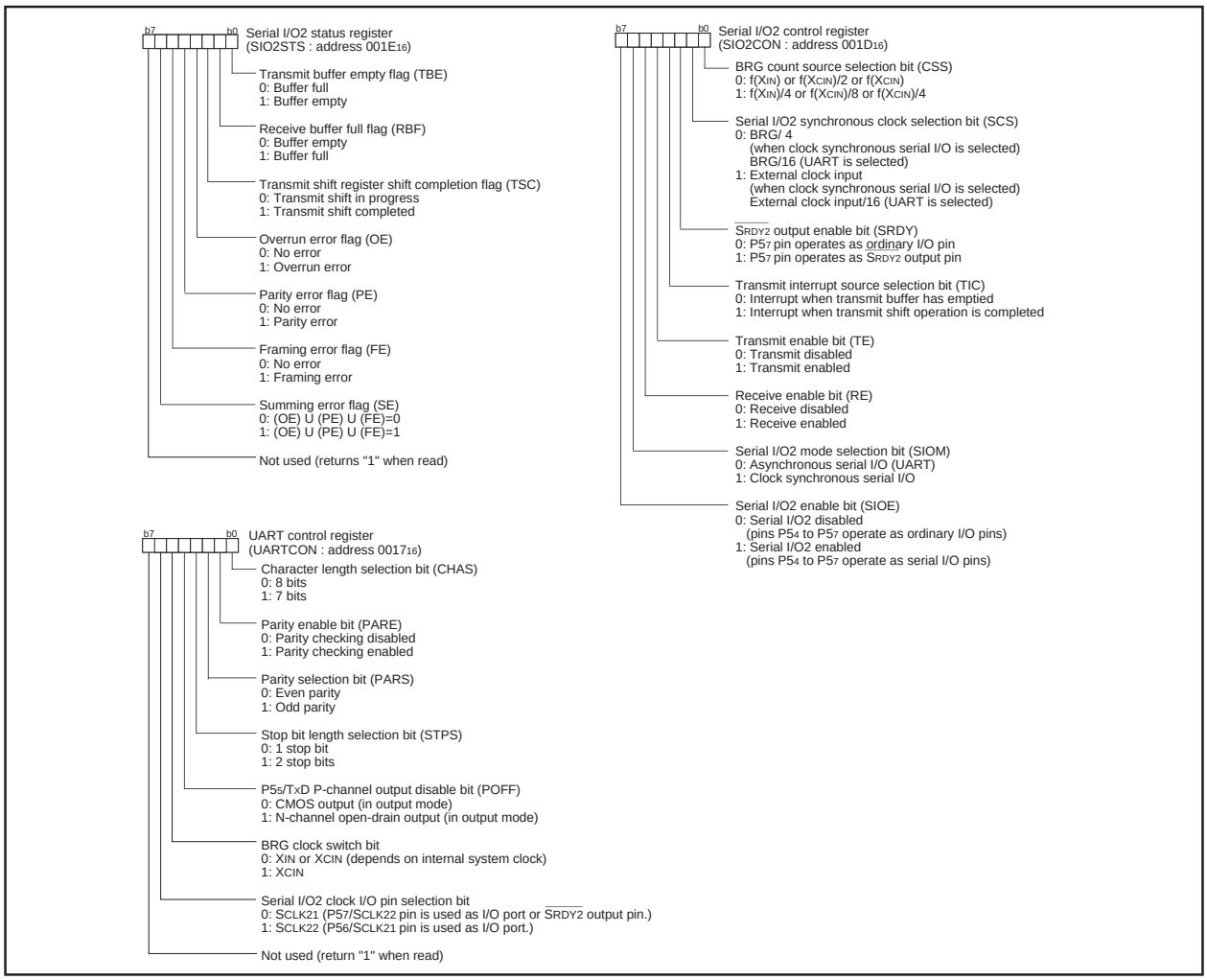


Fig. 40 Structure of serial I/O2 related register

FLD Controller

The 38B5 group has fluorescent display (FLD) drive and control circuits.

The FLD controller consists of the following components:

- 40 pins for FLD control pins
- FLDC mode register
- FLD data pointer
- FLD data pointer reload register
- Tdisp time set register
- Toff1 time set register
- Toff2 time set register
- Port P0FLD/port switch register
- Port P2FLD/port switch register
- Port P8FLD/port switch register
- Port P8 FLD output control register
- FLD automatic display RAM (max. 160 bytes)

A gradation display mode can be used for bright/dark display as a display function.

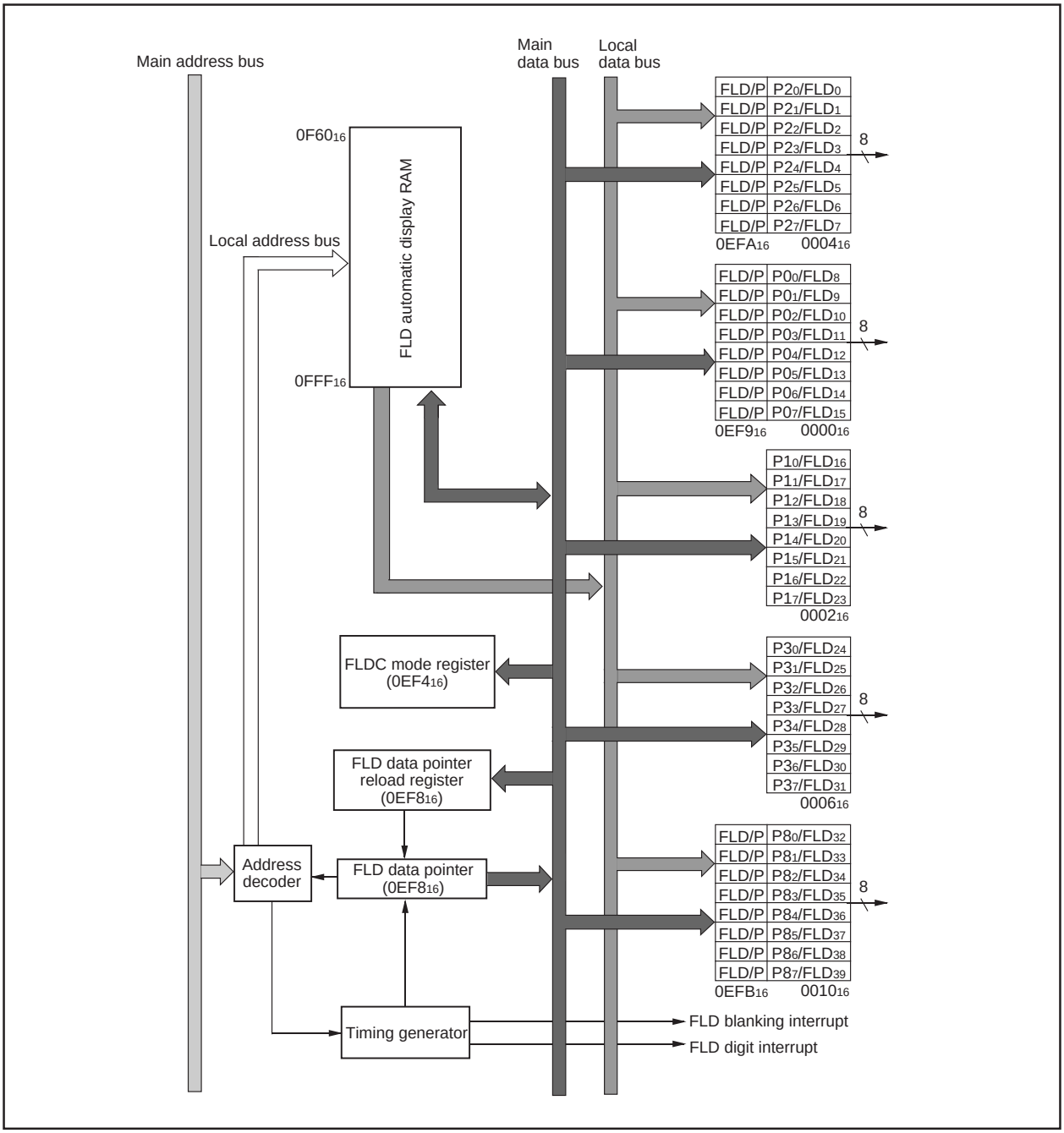
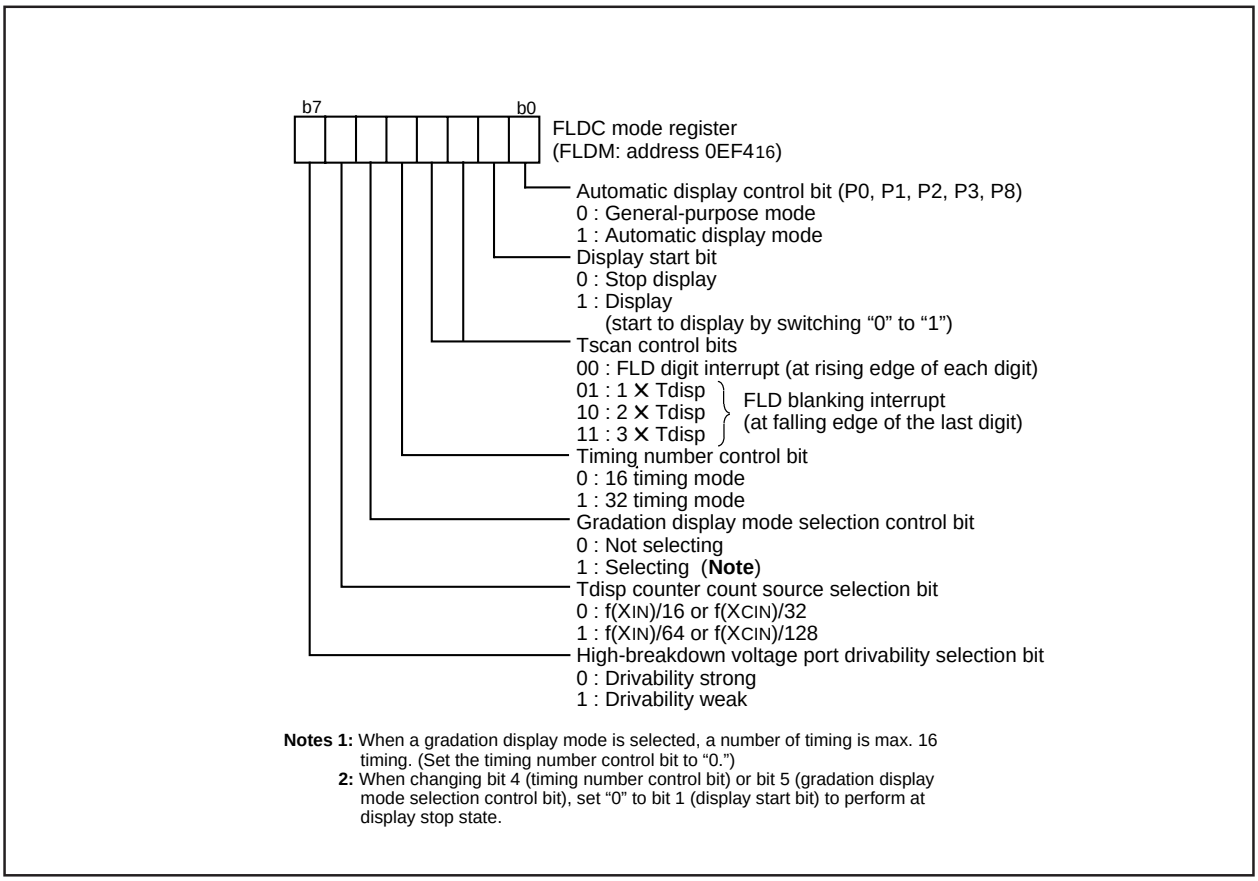


Fig. 41 Block diagram for FLD control circuit

[FLDC Mode Register] FLDM

The FLDC mode register is a 8-bit register respectively which is used to control the FLD automatic display and to set the blanking time Tscan for key-scan.



Notes 1: When a gradation display mode is selected, a number of timing is max. 16 timing. (Set the timing number control bit to “0.”)
2: When changing bit 4 (timing number control bit) or bit 5 (gradation display mode selection control bit), set “0” to bit 1 (display start bit) to perform at display stop state.

Fig. 42 Structure of FLDC mode register

FLD automatic display pins

When the automatic display control bits of the FLDC mode register (address 0EF4$_{16}$) are set to "1," the ports of P0, P1, P2, P3 and P8 are used as FLD automatic display pins.

When using the FLD automatic display mode, set each port to the FLD pin or the general-purpose port using the respective switch register in accordance with the number of segments and the number of digits.

This setting is performed by writing a value into the FLD/port switch register (addresses 0EF9$_{16}$ to 0EFB$_{16}$) of each port.

This setting can be performed in units of bit. When "0" is set, the port is set to the general-purpose port. When "1" is set, the port is set to the FLD pin. There is no restriction on whether the FLD pin is to be used as a segment pin or a digit pin.

Table 9 Pins in FLD automatic display mode

Port Name	Automatic Display Pins	Setting Method
P0, P2, P8$_0$–P8$_3$	FLD$_0$–FLD$_{15}$ FLD$_{32}$–FLD$_{35}$	The individual bits of the FLD/port switch register (addresses 0EF9$_{16}$–0EFB$_{16}$) can be set each pin either FLD port ("1") or general-purpose port ("0").
P1, P3	FLD$_{16}$–FLD$_{31}$	None (FLD only)
P8$_4$–P8$_7$	FLD$_{36}$–FLD$_{39}$	The individual bits of the FLD/port switch register (address 0EFB$_{16}$) can be set each pin to either FLD port ("1") or general-purpose port ("0"). The output can be reversed by the port P8 FLD output control register (address 0EFC$_{16}$). The port output format is the CMOS output format. When using the port as a display pin, a driver must be installed externally.

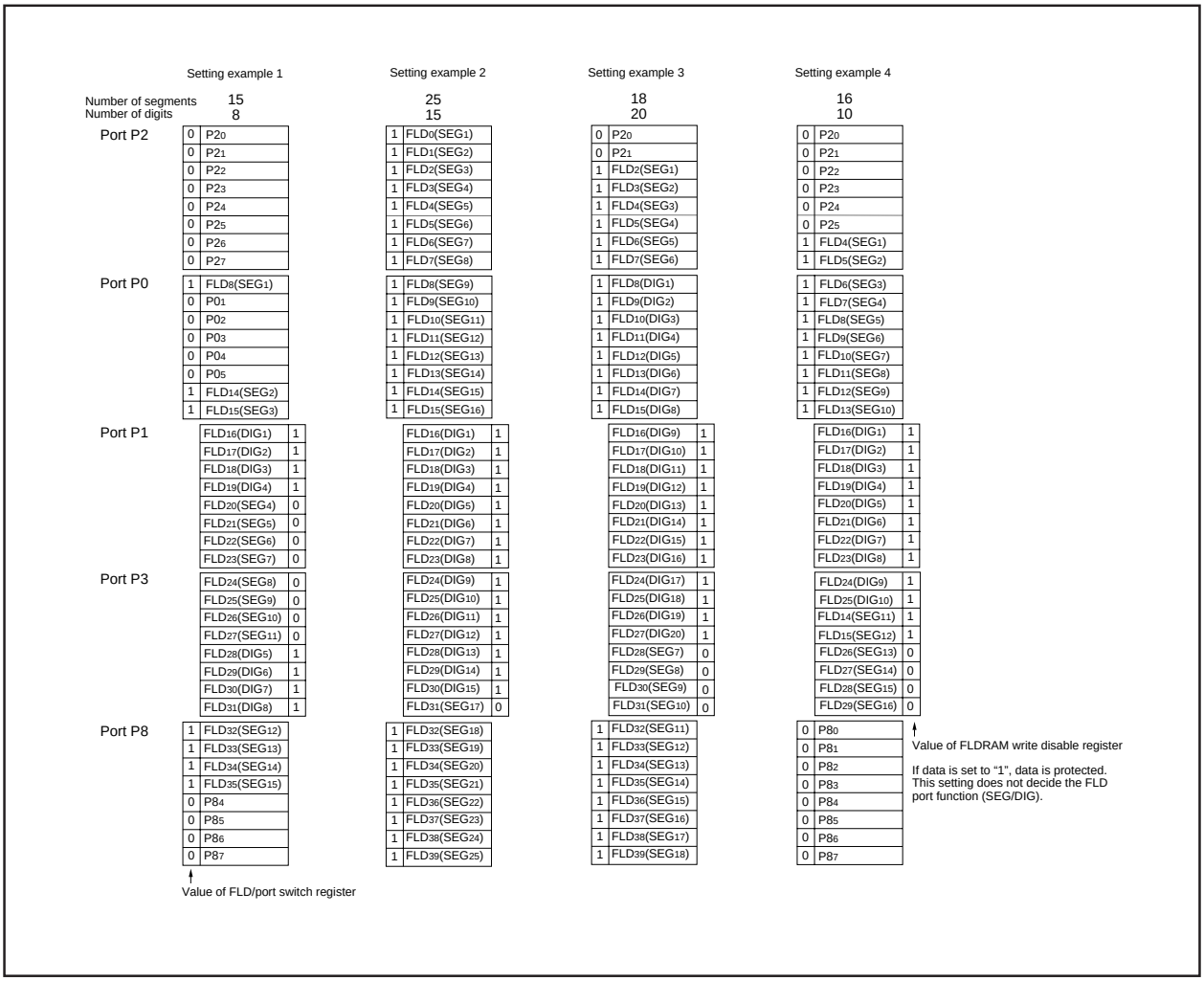


Fig. 43 Segment/Digit setting example

FLD automatic display RAM

The FLD automatic display RAM uses the 160 bytes of addresses $0F60_{16}$ to $0FFF_{16}$. For FLD, the 3 modes of 16-timing ordinary mode, 16-timing•gradation display mode and 32-timing mode are available depending on the number of timings and the presence/absence of gradation display.

The automatic display RAM in each mode is as follows:

(1) 16-timing•Ordinary Mode

The 80 bytes of addresses $0FB0_{16}$ to $0FFF_{16}$ are used as a FLD display data store area. Because addresses $0F60_{16}$ to $0FAF_{16}$ are not used as the automatic display RAM, they can be the ordinary RAM or serial I/O automatic transfer RAM.

(2) 16-timing•Gradation Display Mode

The 160 bytes of addresses $0F60_{16}$ to $0FFF_{16}$ are used. The 80 bytes of addresses $0FB0_{16}$ to $0FFF_{16}$ are used as an FLD display data store area, while the 80 bytes of addresses $0F60_{16}$ to $0FAF_{16}$ are used as a gradation display control data store area.

(3) 32-timing Mode

The 160 bytes of addresses $0F60_{16}$ to $0FFF_{16}$ are used as an FLD display data store area.

[FLD Data Pointer and FLD Data Pointer Reload Register] FLDDP ($0EF8_{16}$)

Both the FLD data pointer and FLD data pointer reload register are 8-bit registers assigned at address $0EF8_{16}$. When writing data to this address, the data is written to the FLD data pointer reload register; when reading data from this address, the value in the FLD data pointer is read.

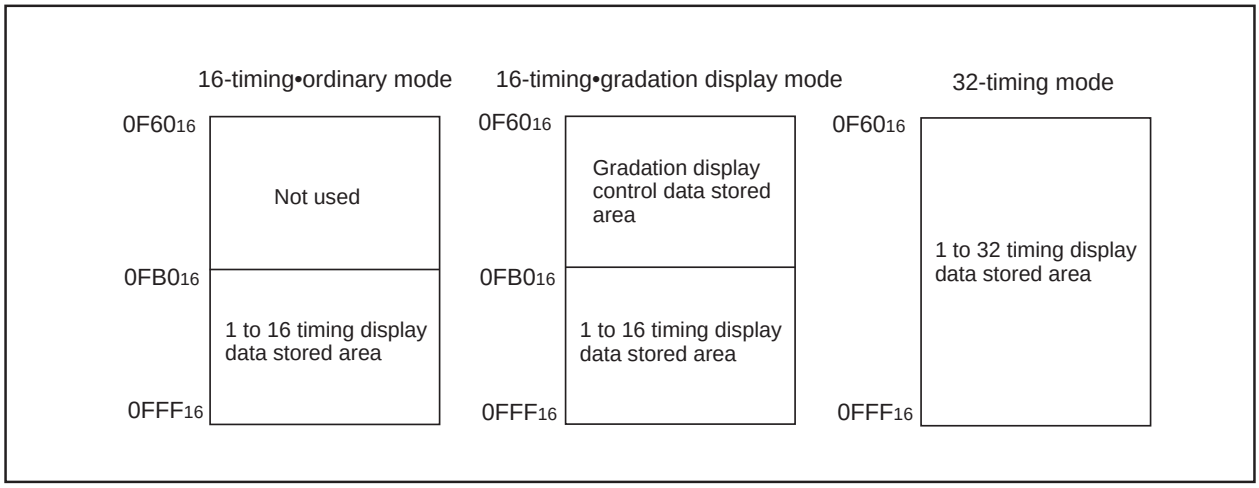


Fig. 44 FLD automatic display RAM assignment

Data setup

(1) 16-timing•Ordinary Mode

The area of addresses 0FB0$_{16}$ to 0FFF$_{16}$ are used as a FLD automatic display RAM.

When data is stored in the FLD automatic display RAM, the last data of FLD port P2 is stored at address 0FB0$_{16}$, the last data of FLD port P0 is stored at address 0FC0$_{16}$, the last data of FLD port P1 is stored at address 0FD0$_{16}$, the last data of FLD port P3 is stored at address 0FE0$_{16}$, and the last data of FLD port P8 is stored at address 0FF0$_{16}$, to assign in sequence from the last data respectively.

The first data of the FLD port P2, P0, P1, P3, and P8 is stored at an address which adds the value of (the timing number – 1) to the corresponding address 0FB0$_{16}$, 0FC0$_{16}$, 0FD0$_{16}$, 0FE0$_{16}$, and 0FF0$_{16}$.

Set the FLD data pointer reload register to the value given by the timing number – 1. “1” is always written to bits 7, 6, and 5. Note that “0” is always read from bits 7, 6, and 5 when reading. “1” is always set to bit 4, but this bit become written value when reading.

(2) 16-timing•Gradation Display Mode

Display data setting is performed in the same way as that of the 16-timing•ordinary mode. Gradation display control data is arranged at an address resulting from subtracting 0050$_{16}$ from the display data store address of each timing and pin. Bright display is performed by setting “0,” and dark display is performed by setting “1.”

Set the FLD data pointer reload register to the value given by the timing number – 1. “1” is always written to bits 7, 6, and 5. Note that “0” is always read from bits 7, 6, and 5 when reading. “1” is always set to bit 4, but this bit become written value when reading.

(3) 32-timing Mode

The area of addresses 0F60$_{16}$ to 0FFF$_{16}$ are used as a FLD automatic display RAM. When data is stored in the FLD automatic display RAM, the last data of FLD port P2 is stored at address 0F60$_{16}$, the last data of FLD port P0 is stored at address 0F80$_{16}$, the last data of FLD port P1 is stored at address 0FA0$_{16}$, the last data of FLD port P3 is stored at address 0FC0$_{16}$, and the last data of FLD port P8 is stored at address 0FE0$_{16}$, to assign in sequence from the last data respectively.

The first data of the FLD port P2, P0, P1, P3, and P8 is stored at an address which adds the value of (the timing number – 1) to the corresponding address 0F60$_{16}$, 0F80$_{16}$, 0FA0$_{16}$, 0FC0$_{16}$, and 0FE0$_{16}$.

Set the FLD data pointer reload register to the value given by the timing number –1. “1” is always written to bits 7, 6, and 5. Note that “0” is always read from bits 7, 6, and 5 when reading.

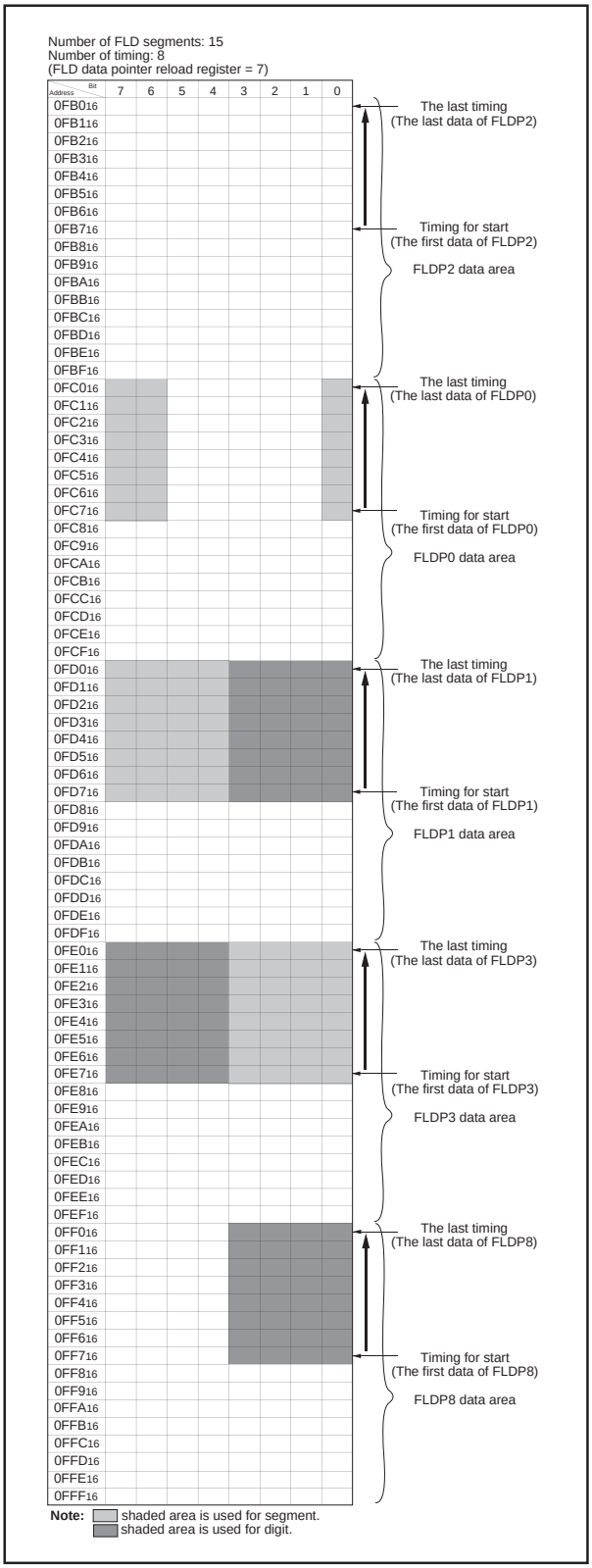


Fig. 45 Example of using FLD automatic display RAM in 16-timing•ordinary mode

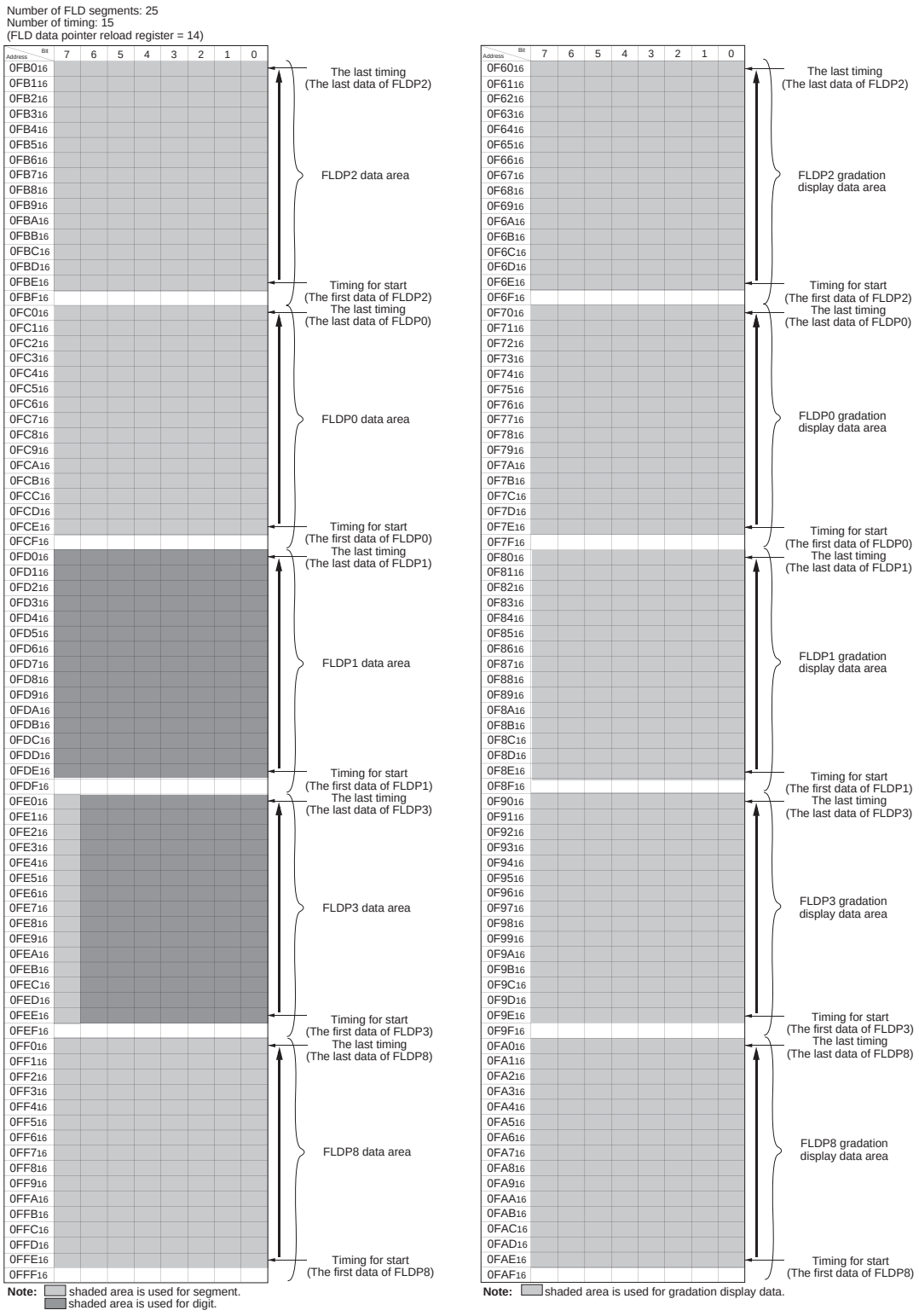


Fig. 46 Example of using FLD automatic display RAM in 16-timing•gradation display mode

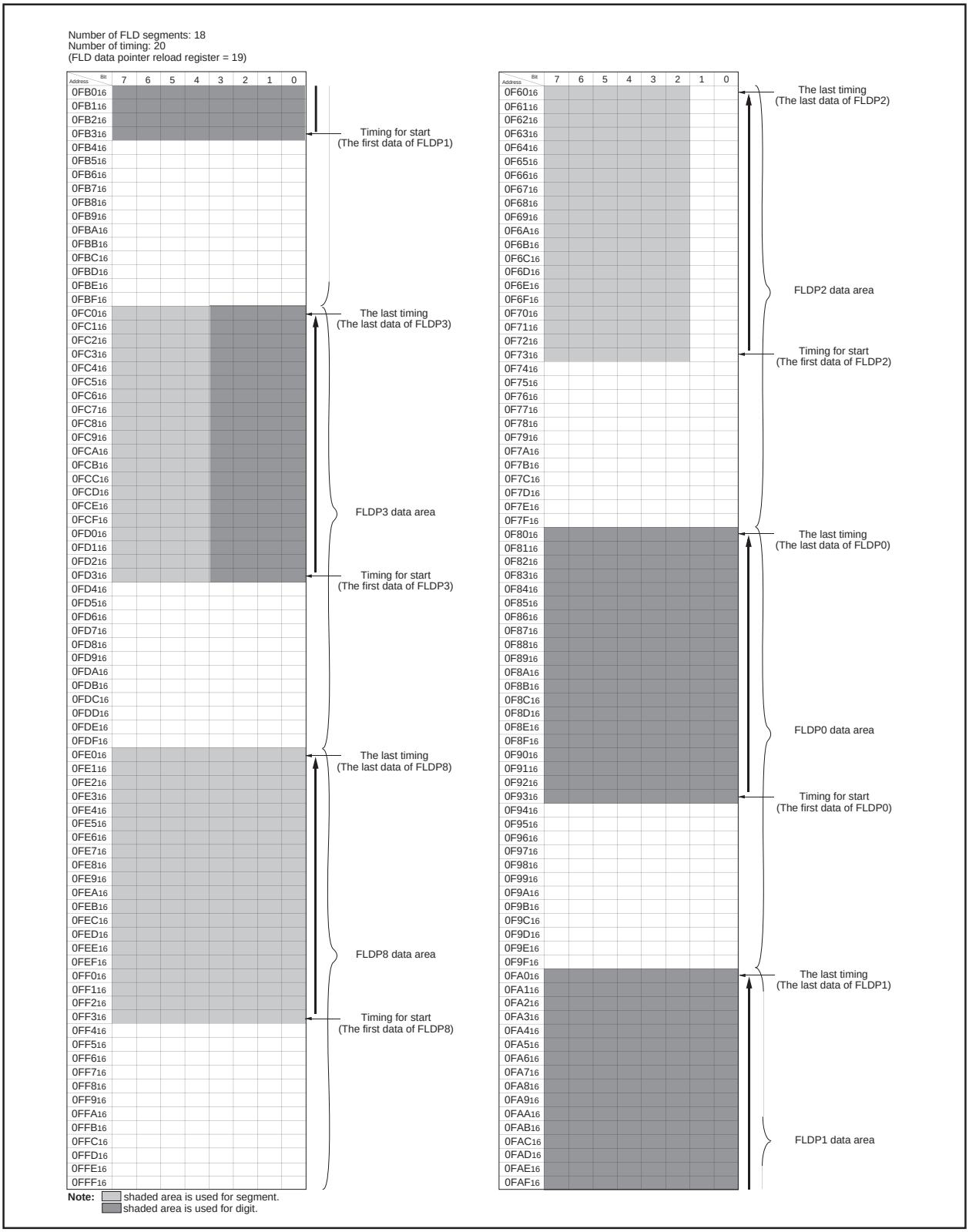


Fig. 47 Example of using FLD automatic display RAM in 32-timing mode

Digit data protect function

The FLD automatic display RAM is provided with a data protect function that disables the RAM area data to be rewritten as digit data.

This function can disable data from being written in optional bits in the RAM area corresponding to P1 to P3. A programming load can be reduced by protecting an area that requires no change after data such as digit data is written.

Write digit data beforehand; then set “1” in the corresponding bits. With this, the setting is completed.

The data protect area becomes the maximum RAM area of P1 and P3. For example, when bit 0 of P1 is protected in the 16-timing•ordinary mode, bits 0 of RAM addresses $0FD0_{16}$ to $0FDF_{16}$ can be protected. Likewise, in the 16-timing•gradation display mode, bits 0 of addresses $0FD0_{16}$ to $0FDF_{16}$ and $0F80_{16}$ to $0F8F_{16}$ can be protected. In the 32-timing mode, bits 0 of addresses $0FA0_{16}$ to $0FBF_{16}$ can be protected.

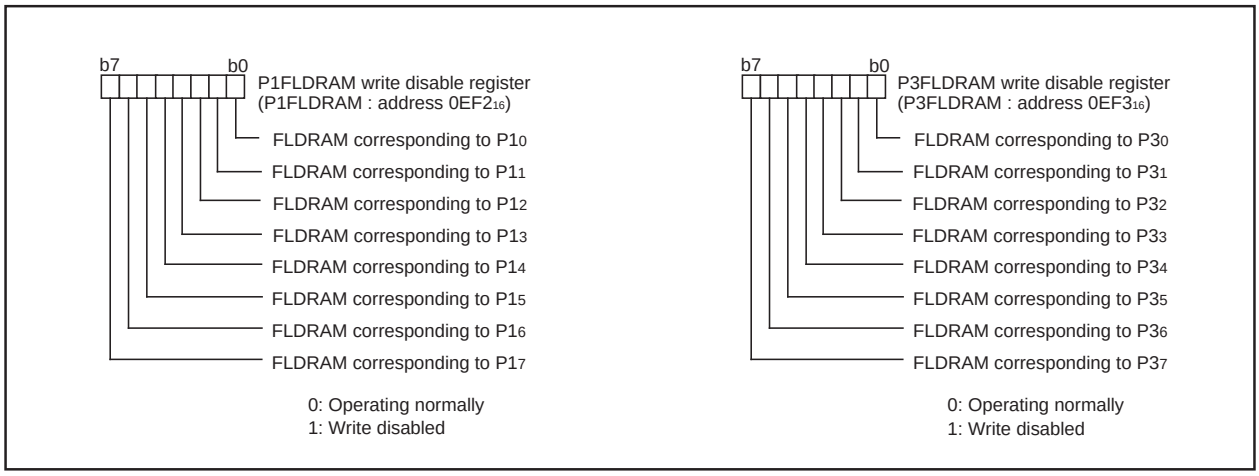


Fig. 48 Structure of FLDRAM write disable register

Setting method when using the grid scan type FLD

When using the grid scan type FLD, set “1” in the RAM area corresponding to the digit ports that output “1” at each timing. Set “0” in the RAM area corresponding to the other digit ports.

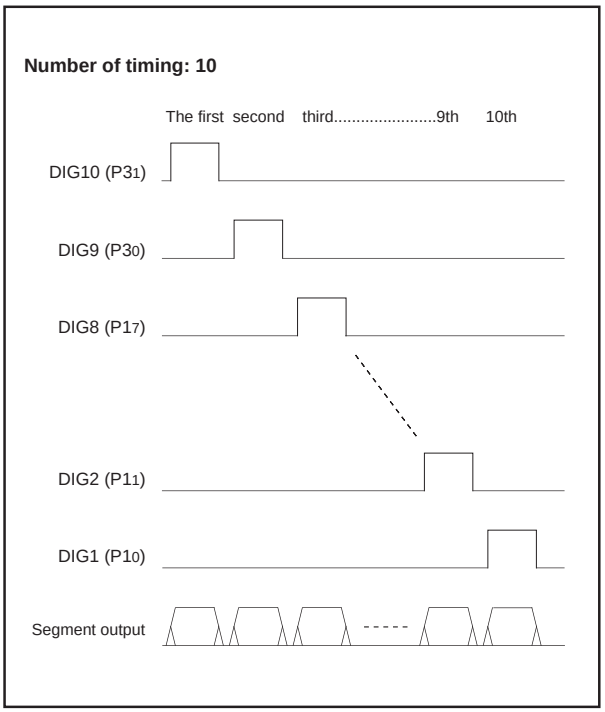


Fig. 49 Example of digit timing using grid scan type

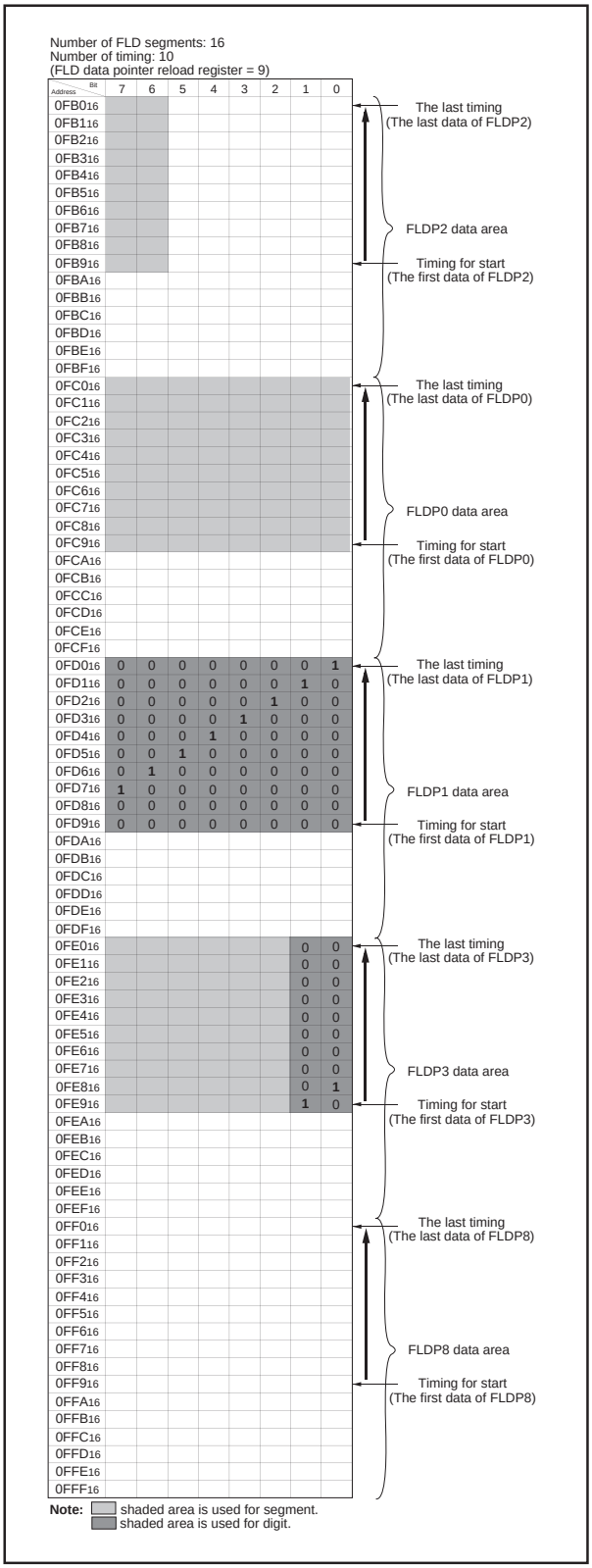


Fig. 50 Example of using FLD automatic display RAM using grid scan type

Timing setting

Each timing is set by the FLDC mode register, Tdisp time set register, Toff1 time set register, and Toff2 time set register.

•Tdisp time setting

Set the Tdisp time by the Tdisp counter count source selection bit of the FLDC mode register and the Tdisp time set register. Supposing that the value of the Tdisp time set register is n, the Tdisp time is represented as Tdisp = (n+1) × t (t: count source synchronization).

When the Tdisp counter count source selection bit of the FLDC mode register is "0" and the value of the Tdisp time set register is 200 ($C8_{16}$), the Tdisp time is: Tdisp = (200+1) × 4 (at X_{IN}= 4 MHz) = 804 µs. When reading the Tdisp time set register, the value in the counter is read out.

•Toff1 time setting

Set the Toff1 time by the Toff1 time set register.

Supposing that the value of the Toff1 time set register is n1, the Toff1 time is represented as Toff1 = n1 × t.

When the Tdisp counter count source selection bit of the FLDC mode register is "0" and the value of the Toff1 time set register is 30 ($1E_{16}$), Toff1 = 30 × 4 (at X_{IN} = 4 MHz) = 120 µs.

Set a value of 03_{16} or more to the Toff1 time set register (address $0EF6_{16}$).

•Toff2 time setting

Set the Toff2 time by the Toff2 time set register.

Supposing that the value of the Toff2 time set register is n2, the Toff2 time is represented as Toff2 = n2 × t.

When the Tdisp counter count source selection bit of the FLDC mode register is "0" and the value of the Toff2 time set register is 180 ($B4_{16}$), Toff2 = 180 × 4 (at X_{IN} = 4 MHz) = 720 µs.

This Toff2 time setting is valid only for FLD ports which are in the gradation display mode and whose gradation display control RAM value is "1."

When setting "1" to bit 7 of the P8FLD output control register (address $0EFC_{16}$), set a value of 03_{16} or more to the Toff2 time set register (address $0EF7_{16}$).

FLD automatic display start

To perform FLD automatic display, set the following registers.

•Port P0FLD/port switch register
•Port P2FLD/port switch register
•Port P8FLD/port switch register
•FLDC mode register
•Tdisp time set register
•Toff1 time set register
•Toff2 time set register
•FLD data pointer

FLD automatic display mode is selected by writing "1" to the bit 0 of the FLDC mode register (address $0EF4_{16}$), and the automatic display is started by writing "1" to bit 1. During FLD automatic display, bit 1 of the FLDC mode register (address $0EF4_{16}$) always keeps "1," and FLD automatic display can be interrupted by writing "0" to bit 1.

Key-scan

When a key-scan is performed with the segment during key-scan blanking period Tscan, take the following sequence:

1. Write "0" to bit 0 of the FLDC mode register (address $0EF4_{16}$).
2. Set the port corresponding to the segment for key-scan to the output port.
3. Perform the key-scan.
4. After the key-scan is performed, write "1" to bit 0 of FLDC mode register (address $0EF4_{16}$).

■ Note

When performing a key-scan according to the above step 1 to 4, take the following points into consideration.

1. Do not set "0" in bit 1 of the FLDC mode register (address $0EF4_{16}$).
2. Do not set "1" in the ports corresponding to digits.

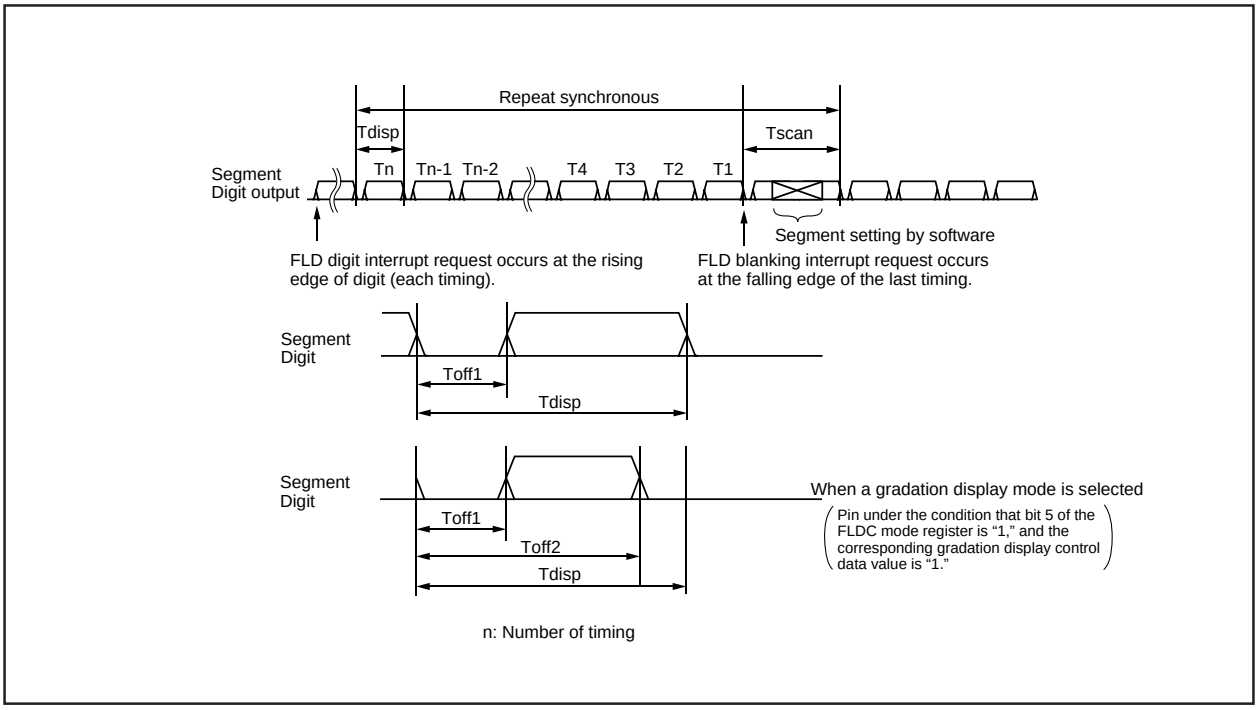


Fig. 51 FLDC timing

P84 to P87 FLD output reverse function

P84 to P87 are provided with a function to reverse the polarity of the FLD output. This function is useful in adjusting the polarity when using an externally installed driver.
The output polarity can be reversed by setting "1" to bit 0 of the port P8 FLD output control register.

P84 to P87 FLDRAM write disable function

This function can disable writing data in the RAM area corresponding to P84 to P87. This function can be set by setting "1" to bit 1 of the port P8FLD output control register (address 0EFC16).

P84 to P87 Toff invalid function

P84 to P87 can output waveform in which Toff is invalid, when P84 to P87 is selected FLD ports (See Figure 52).
The function is useful when using a 4 bits →16 bits decoder. The Toff can be invalid by setting "1" to bit 2 of the port P8FLD output control register (address 0EFC16).

P84 to P87 output delay function

P84 to P87 can output waveform in which is delayed for 16 µs, when selecting FLD port and selecting Toff invalid function (See Figure 52). When using a 4 bits →16 bits decoder, the function can be useful for prevention of leak radiation caused by phase discrepancy between segment output waveform and digit output waveform. This function can be set by setting "1" to bit 3 of the port P8FLD output control register (address 0EFC16).

Dimmer signal output function

P63 can output the dimmer signal. When using a 4 bits →16 bits decoder, the dimmer signal can be used as a control signal for a 4 bits →16 bits decoder. When using M35501FP, the dimmer signal can be used as the CLK signal. The dimmer signal can be output by setting "1" to bit 4 of the port P8FLD output control register (address 0EFC16).

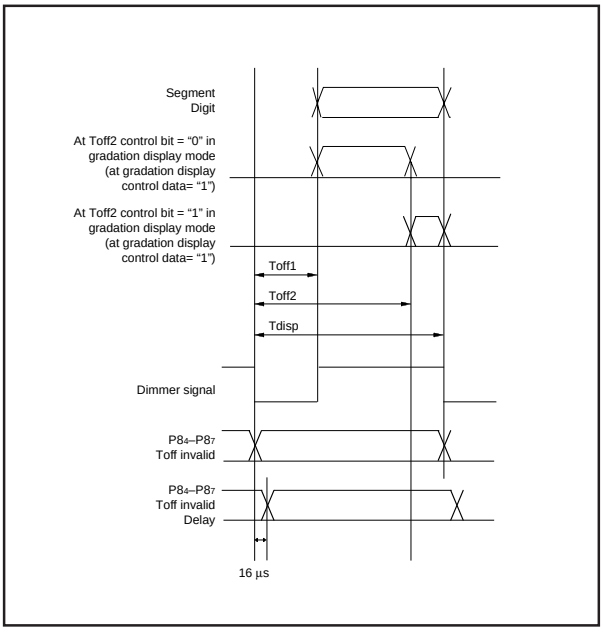


Fig. 52 P84 to P87 FLD output waveform

Toff2 SET/RESET change function

The value of the Toff2 time set register is valid when gradation display mode is selected. The FLD ports output (set) the data of display RAM at the end of the Toff1 time and output "0" (reset) at the end of the Toff2 time, when bit 7 of the port P8FLD output control register is "0".
The FLD ports output (set) the data of display RAM at the end of the Toff2 time and output "0" (reset) at the end of Tdisp time, when bit 7 of the port P8FLD output control register is "1".

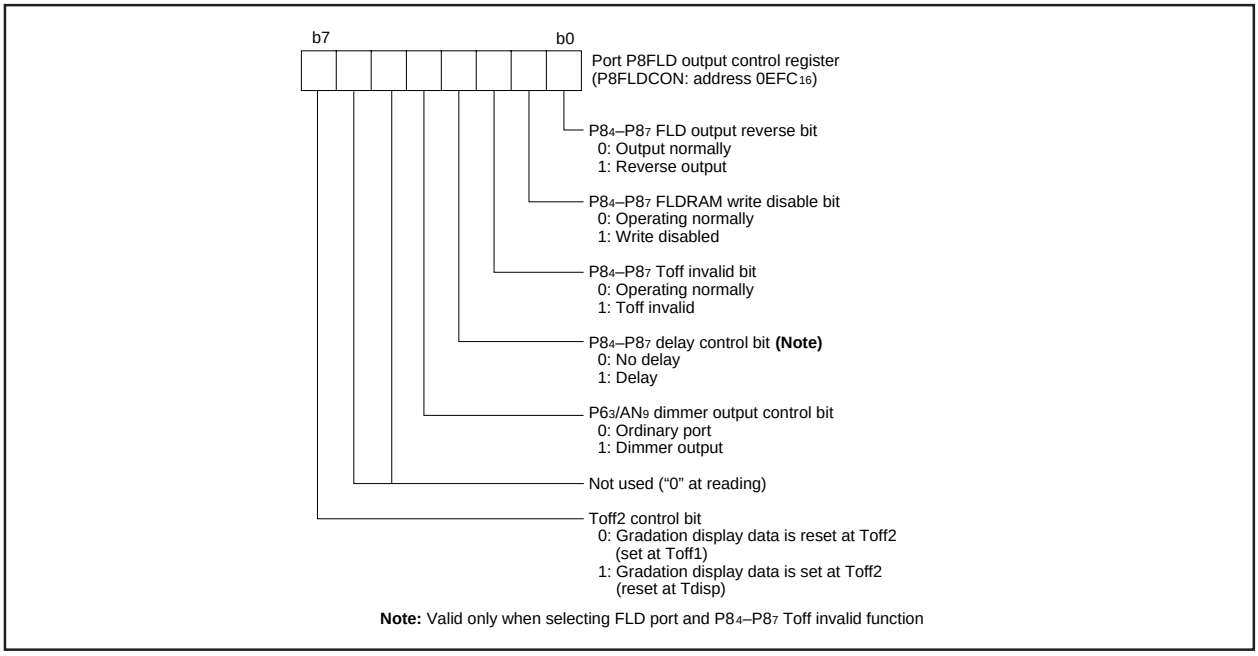


Fig. 53 Structure of port P8 FLD output control register

A-D Converter

The 38B5 group has a 10-bit A-D converter. The A-D converter performs successive approximation conversion.

[A-D Conversion Register] AD

One of these registers is a high-order register, and the other is a low-order register. The high-order 8 bits of a conversion result is stored in the A-D conversion register (high-order) (address 0034_{16}), and the low-order 2 bits of the same result are stored in bit 7 and bit 6 of the A-D conversion register (low-order) (address 0033_{16}). During A-D conversion, do not read these registers.

[A-D Control Register] ADCON

This register controls A-D converter. Bits 3 to 0 are analog input pin selection bits. Bit 4 is an AD conversion completion bit and "0" during A-D conversion. This bit is set to "1" upon completion of A-D conversion.

A-D conversion is started by setting "0" in this bit.

[Comparison Voltage Generator]

The comparison voltage generator divides the voltage between AV_{SS} and V_{REF}, and outputs the divided voltages.

[Channel Selector]

The channel selector selects one of the input ports $P7_7/AN_7$–$P7_0/AN_0$, and $P6_5/S_{STB1}/AN_{11}$–$P6_2/S_{RDY1}/AN_8$ and inputs it to the comparator.

When port $P6_4$ is selected as an analog input pin, an external interrupt function (INT_4) is invalid.

[Comparator and Control Circuit]

The comparator and control circuit compares an analog input voltage with the comparison voltage and stores the result in the A-D conversion register. When an A-D conversion is completed, the control circuit sets the AD conversion completion bit and the AD conversion interrupt request bit to "1."

Note that the comparator is constructed linked to a capacitor, so set $f(X_{IN})$ to at least 250 kHz during A-D conversion. Use a CPU system clock dividing the main clock X_{IN} as the internal system clock.

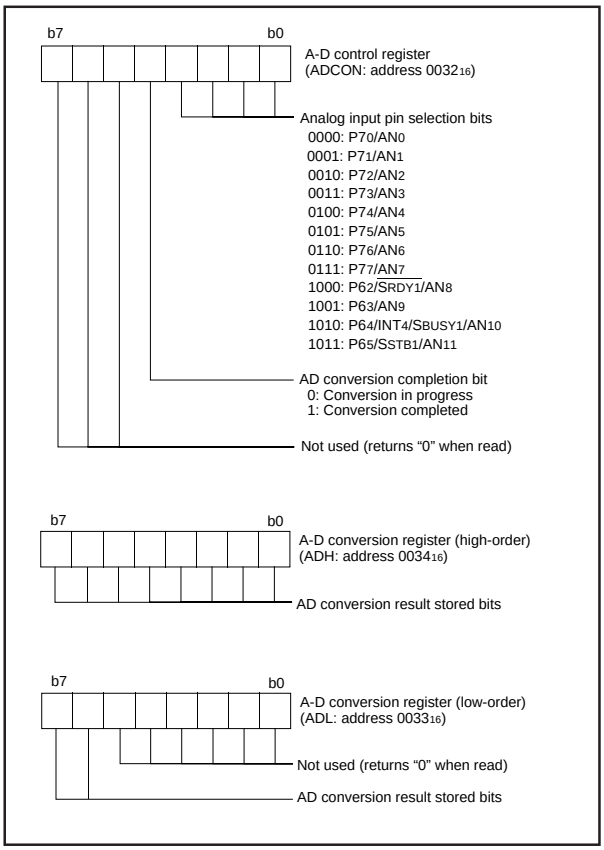


Fig. 54 Structure of A-D control register

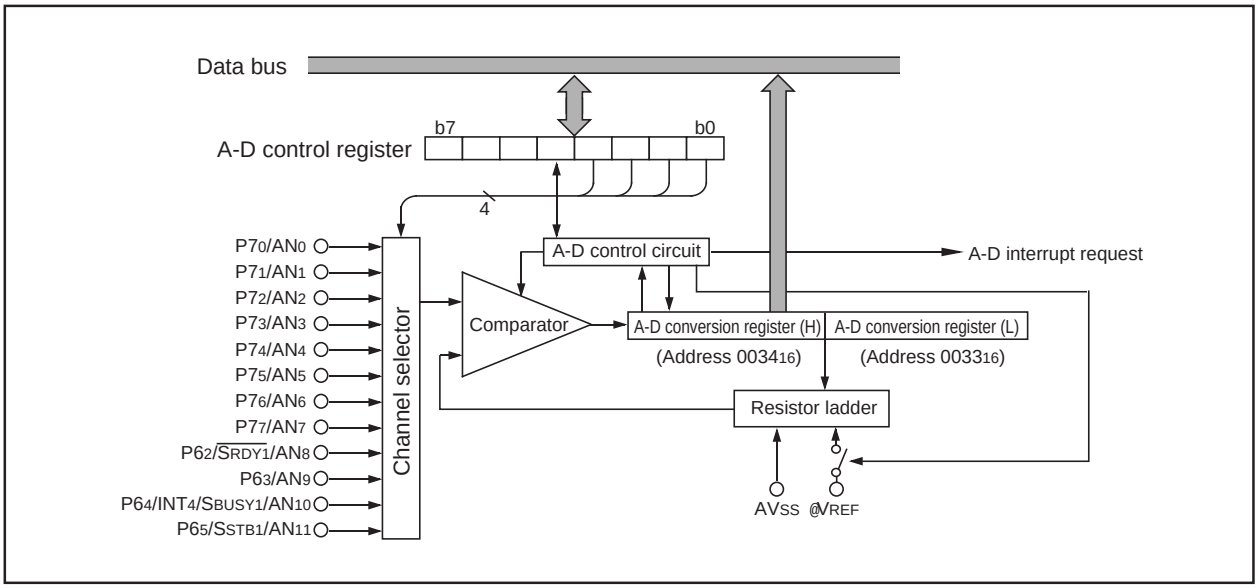


Fig. 55 Block diagram of A-D converter

Pulse Width Modulation (PWM)

The 38B5 group has a PWM function with a 14-bit resolution. When the oscillation frequency X_{IN} is 4 MHz, the minimum resolution bit width is 250 ns and the cycle period is 4096 µs. The PWM timing generator supplies a PWM control signal based on a signal that is the frequency of the X_{IN} clock.

The explanation in the rest assumes X_{IN} = 4 MHz.

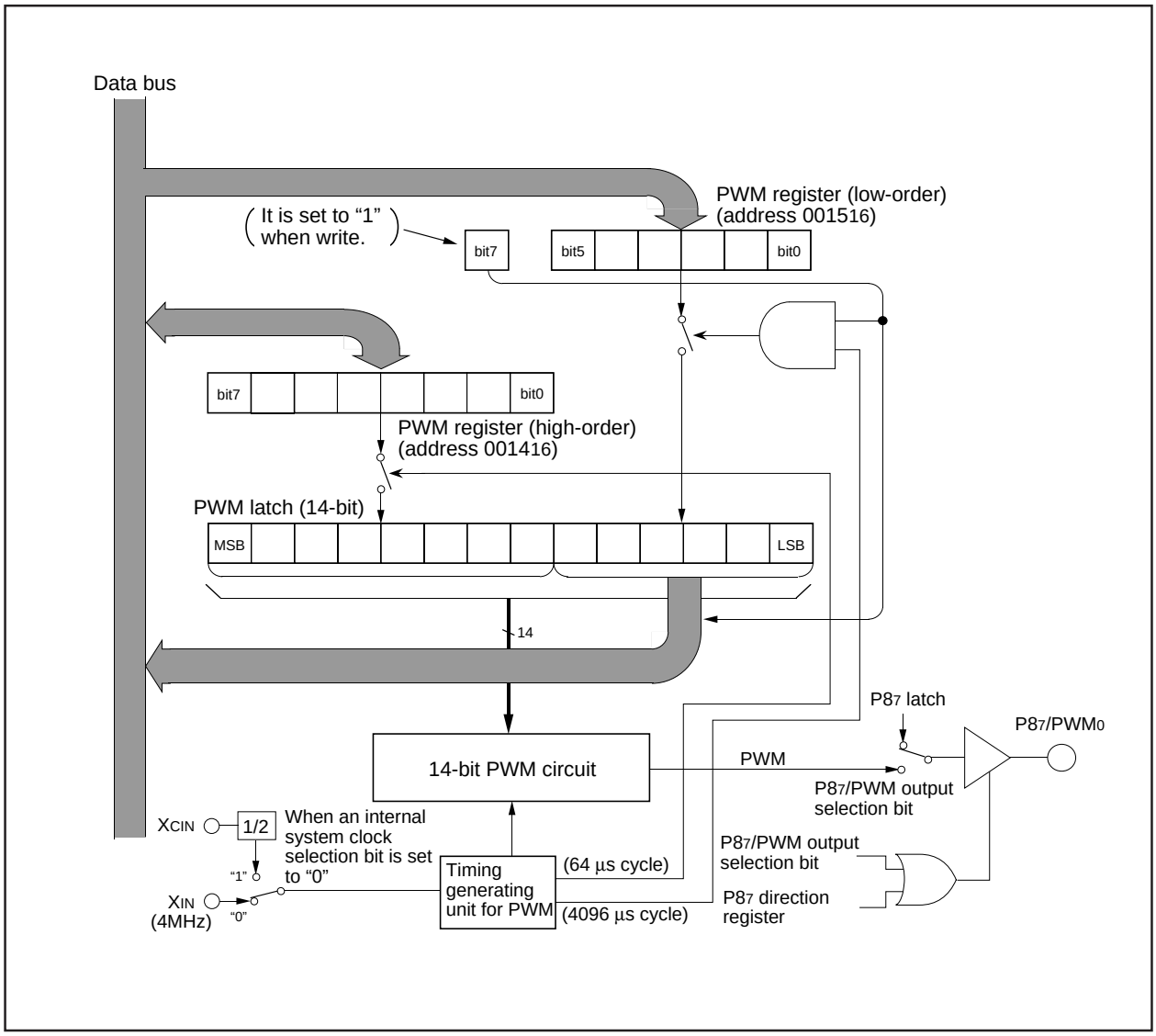


Fig. 56 PWM block diagram

1. Data setup

The PWM output pin also function as port $P8_7$. Set port $P8_7$ to be the PWM output pin by setting bit 0 of the PWM control register (address 0026_{16}) to "1." The high-order 8 bits of output data are set in the high-order PWM register PWMH (address 0014_{16}) and the low-order 6 bits are set in the low-order PWM register PWML (address 0015_{16}).

2. PWM operation

The timing of the 14-bit PWM function is shown in Figure 57.

The 14-bit PWM data is divided into the low-order 6 bits and the high-order 8 bits in the PWM latch.

The high-order 8 bits of data determine how long an "H" level signal is output during each sub-period. There are 64 sub-periods in each period, and each sub-period t is 256 × τ (= 64 μs) long. The signal's "H" has a length equal to N times τ, and its minimum resolution = 250 ns.

The last bit of the sub-period becomes the ADD bit which is specified either "H" or "L," by the contents of PWML. As shown in Table 10, the ADD bit is decided either "H" or "L."

That is, only in the sub-period tm shown in Table 10 in the PWM cycle period T = 64t, the "H" duration is lengthened during the minimum resolution width τ period in comparison with the other period.

For example, if the high-order eight bits of the 14-bit data are "03_{16}" and the low-order six bits are "05_{16}," the length of the "H" level output in sub-periods t_8, t_{24}, t_{32}, t_{40} and t_{56} is 4 τ, and its length 3 τ in all other sub-periods.

Time at the "H" level of each sub-period almost becomes equal because the time becomes length set in the high-order 8 bits or becomes the value plus τ, and this sub-period t (= 64 μs, approximate 15.6 kHz) becomes cycle period approximately.

3. Transfer from register to latch

Data written to the PWML register is transferred to the PWM latch once in each PWM period (every 4096 μs), and data written to the PWMH register is transferred to the PWM latch once in each sub-period (every 64 μs). When the PWML register is read, the contents of the latch are read. However, bit 7 of the PWML register indicates whether the transfer to the PWM latch is completed; the transfer is completed when bit 7 is "0."

Table 10 Relationship between low-order 6-bit data and setting period of ADD bit

Low-order 6-bit data (LSB)	Sub-periods tm lengthened (m = 0 to 63)
0 0 0 0 0 0	None
0 0 0 0 0 1	m = 32
0 0 0 0 1 0	m = 16, 48
0 0 0 1 0 0	m = 8, 24, 40, 56
0 0 1 0 0 0	m = 4, 12, 20, 28, 36, 44, 52, 60
0 1 0 0 0 0	m = 2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62
1 0 0 0 0 0	m = 1, 3, 5, 7, ..., 57, 59, 61, 63

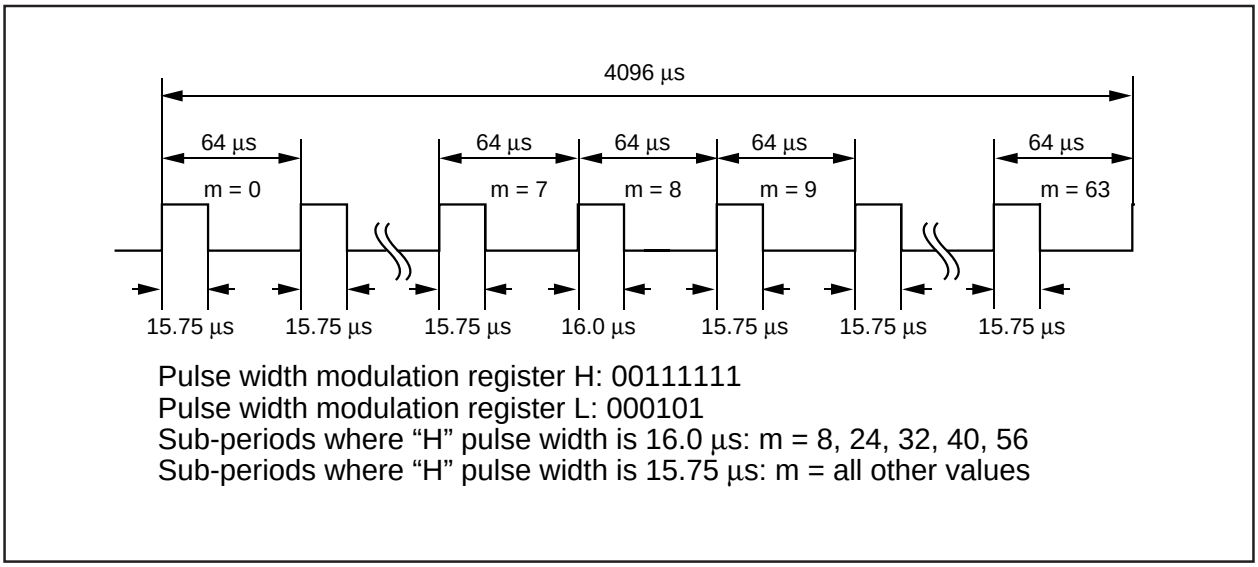


Fig. 57 PWM timing

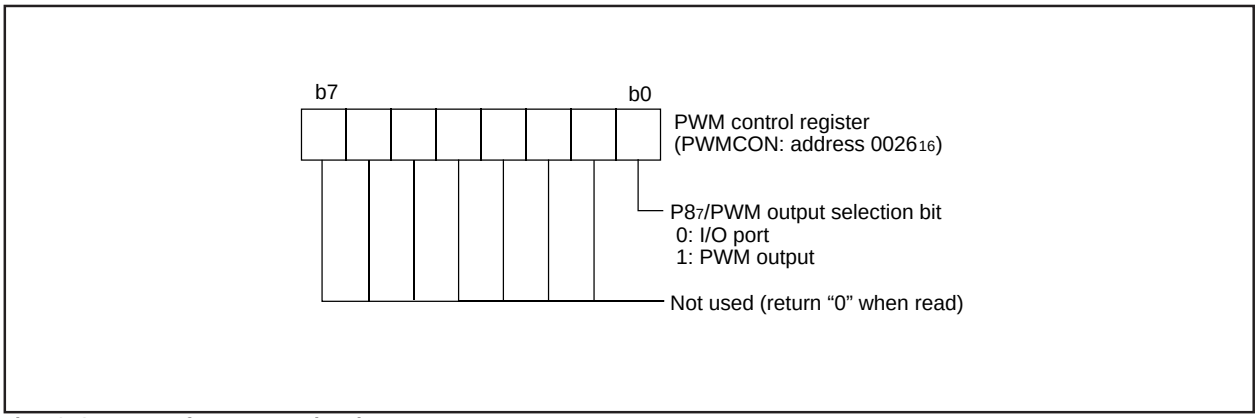


Fig. 58 Structure of PWM control register

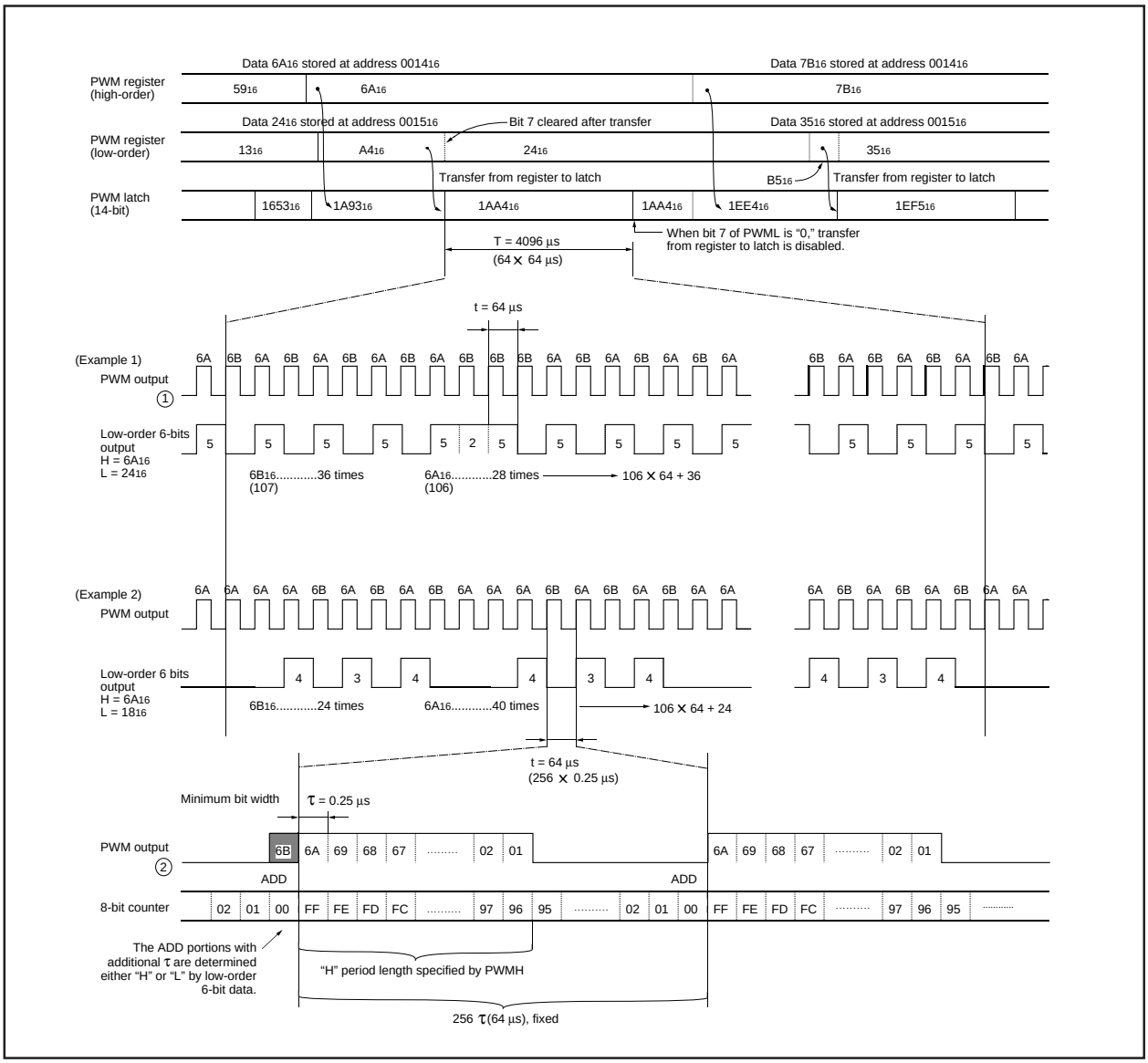


Fig. 59 14-bit PWM timing

Interrupt Interval Determination Function

The 38B5 group has an interrupt interval determination circuit.

This interrupt interval determination circuit has an 8-bit binary up counter. Using this counter, it determines a duration of time from the rising edge (falling edge) of an input signal pulse on the P47/INT2 pin to the rising edge (falling edge) of the signal pulse that is input next.

How to determine the interrupt interval is described below.

1. Enable the INT2 interrupt by setting bit 2 of the interrupt control register 1 (address 003E16). Select the rising interval or falling interval by setting bit 2 of the interrupt edge selection register (address 003A16).
2. Set bit 0 of the interrupt interval determination control register (address 003116) to "1" (interrupt interval determination operating).
3. Select the sampling clock of 8-bit binary up counter by setting bit 1 of the interrupt interval determination control register. When writing "0," f(XIN)/128 is selected (the sampling interval: 32 µs at f(XIN) = 4.19 MHz); when "1," f(XIN)/256 is selected (the sampling interval: 64 µs at f(XIN) = 4.19 MHz).
4. When the signal of polarity which is set on the INT2 pin (rising or falling edge) is input, the 8-bit binary up counter starts counting up of the selected counter sampling clock.
5. When the signal of polarity above 4 is input again, the value of the 8-bit binary up counter is transferred to the interrupt interval determination register (address 003016), and the remote control interrupt request occurs. Immediately after that, the 8-bit binary up counter continues to count up again from "0016."
6. When count value reaches "FF16," the 8-bit binary up counter stops counting up. Then, simultaneously when the next counter sampling clock is input, the counter sets value "FF16" to the interrupt interval determination register to generate the counter overflow interrupt request.

Noise filter

The P47/INT2 pin builds in the noise filter.

The noise filter operation is described below.

1. Select the sampling clock of the input signal with bits 2 and 3 of the interrupt interval determination control register. When not using the noise filter, set "00."
2. The P47/INT2 input signal is sampled in synchronization with the selected clock. When sampling the same level signal in a series of three sampling, the signal is recognized as the interrupt signal, and the interrupt request occurs.
   When setting bit 4 of interrupt interval determination control register to "1," the interrupt request can occur at both rising and falling edges.
   When using the noise filter, set the minimum pulse width of the INT2 input signal to 3 cycles or more of the sample clock.

Note: In the low-speed mode (CM7 = 1), the interrupt interval determination function cannot operate.

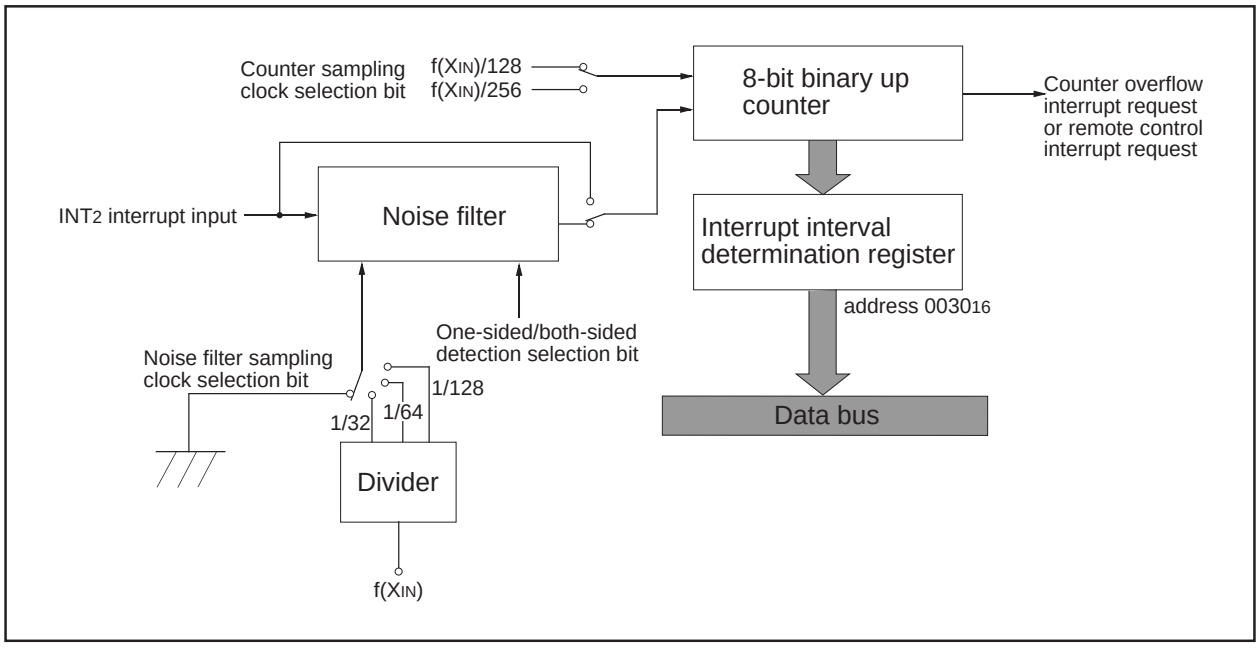


Fig. 60 Interrupt interval determination circuit block diagram

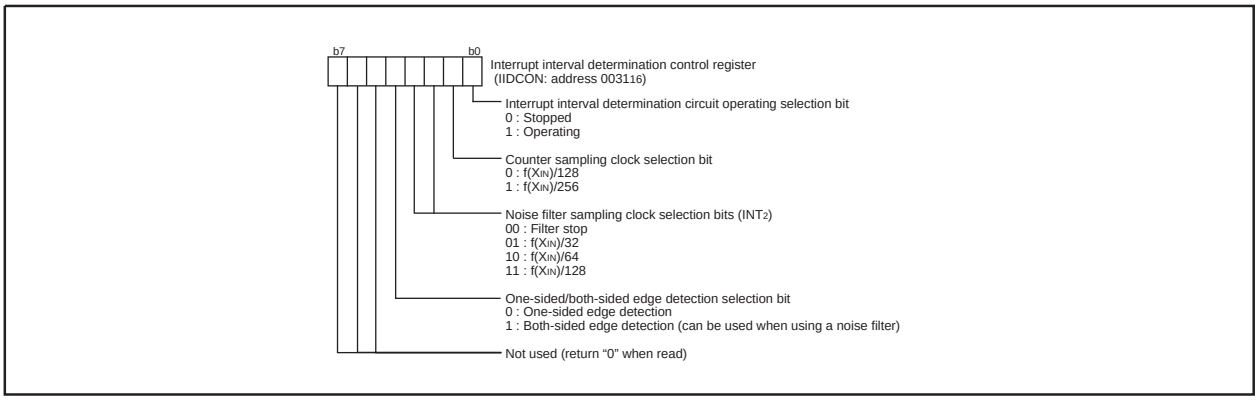


Fig. 61 Structure of interrupt interval determination control register

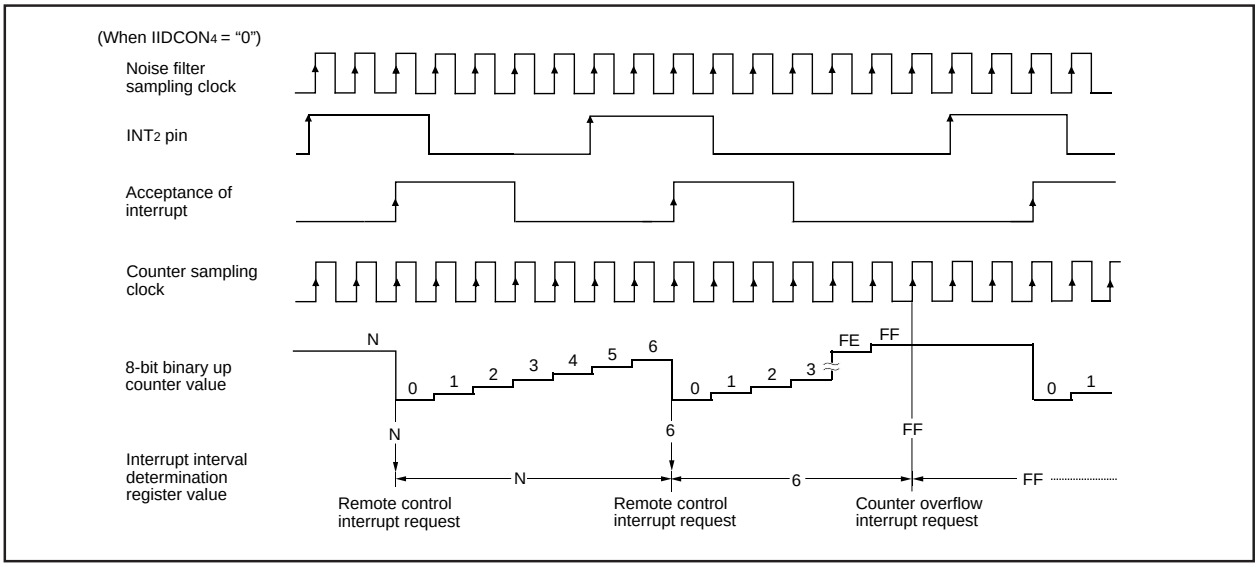


Fig. 62 Interrupt interval determination operation example (at rising edge active)

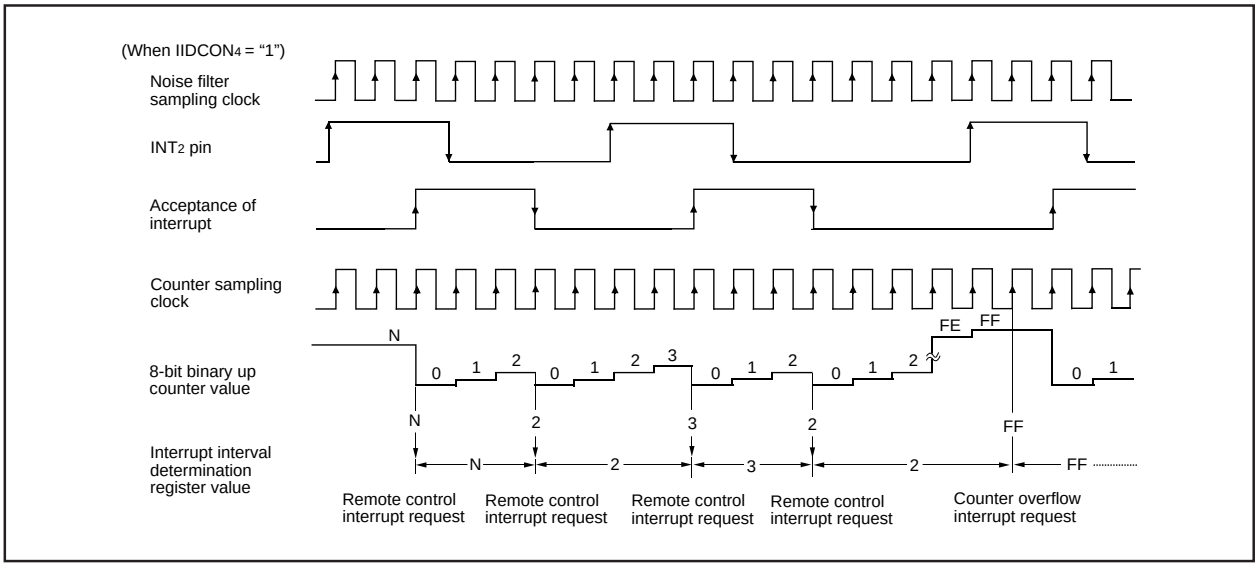


Fig. 63 Interrupt interval determination operation example (at both-sided edge active)

Watchdog Timer

The watchdog timer gives a mean of returning to the reset status when a program cannot run on a normal loop (for example, because of a software runaway). The watchdog timer consists of an 8-bit watchdog timer L and a 12-bit watchdog timer H.

●Standard operation of watchdog timer

When any data is not written into the watchdog timer control register (address 002B16) after resetting, the watchdog timer is in the stop state. The watchdog timer starts to count down by writing an optional value into the watchdog timer control register (address 002B16) and an internal reset occurs at an underflow of the watchdog timer H. Accordingly, programming is usually performed so that writing to the watchdog timer control register (address 002B16) may be started before an underflow. When the watchdog timer control register (address 002B16) is read, the values of the high-order 6 bits of the watchdog timer H, STP instruction disable bit, and watchdog timer H count source selection bit are read.

(1) Initial value of watchdog timer

At reset or writing to the watchdog timer control register (address 002B16), a watchdog timer H is set to "FFF16" and a watchdog timer L to "FF16."

(2) Watchdog timer H count source selection bit operation

Bit 7 of the watchdog timer control register (address 002B16) permits selecting a watchdog timer H count source. When this bit is set to "0," the underflow signal of watchdog timer L becomes the count source. The detection time is set then to f(XIN) = 2.1 s at 4 MHz frequency and f(XCIN) = 512 s at 32 kHz frequency.

When this bit is set to "1," the count source becomes the signal divided by 8 for f(XIN) (or divided by 16 for f(XCIN)). The detection time in this case is set to f(XIN) = 8.2 ms at 4 MHz frequency and f(XCIN) = 2 s at 32 KHz frequency. This bit is cleared to "0" after resetting.

(3) Operation of STP instruction disable bit

Bit 6 of the watchdog timer control register (address 002B16) permits disabling the STP instruction when the watchdog timer is in operation.

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

When this bit is "1," the STP instruction is disabled.

Once the STP instruction is executed, an internal resetting occurs. When this bit is set to "1," it cannot be rewritten to "0" by program. This bit is cleared to "0" after resetting.

■ Note

When releasing the stop mode, the watchdog timer performs its count operation even in the stop release waiting time. Be careful not to cause the watchdog timer H to underflow in the stop release waiting time, for example, by writing data in the watchdog timer control register (address 002B16) before executing the STP instruction.

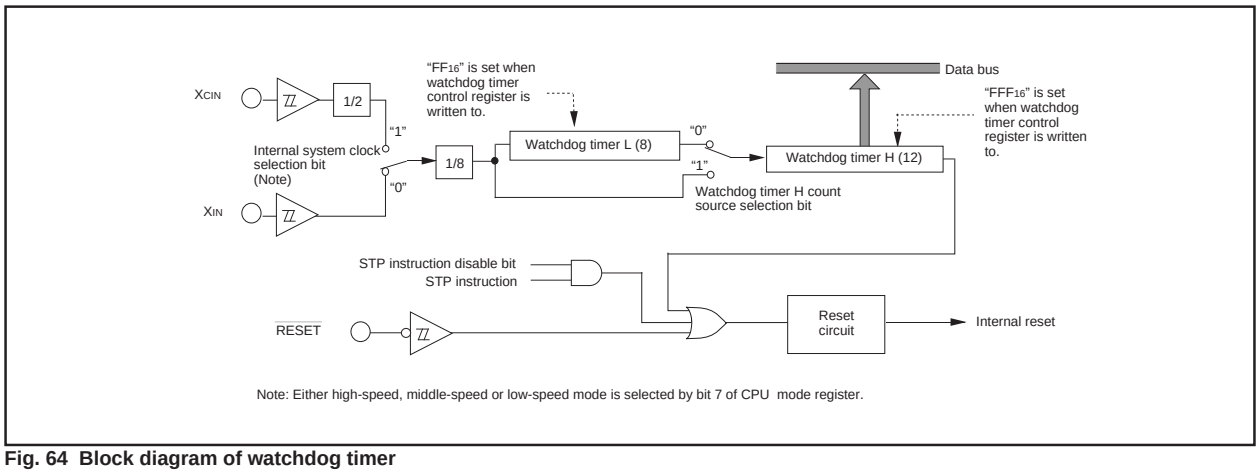


Fig. 64 Block diagram of watchdog timer

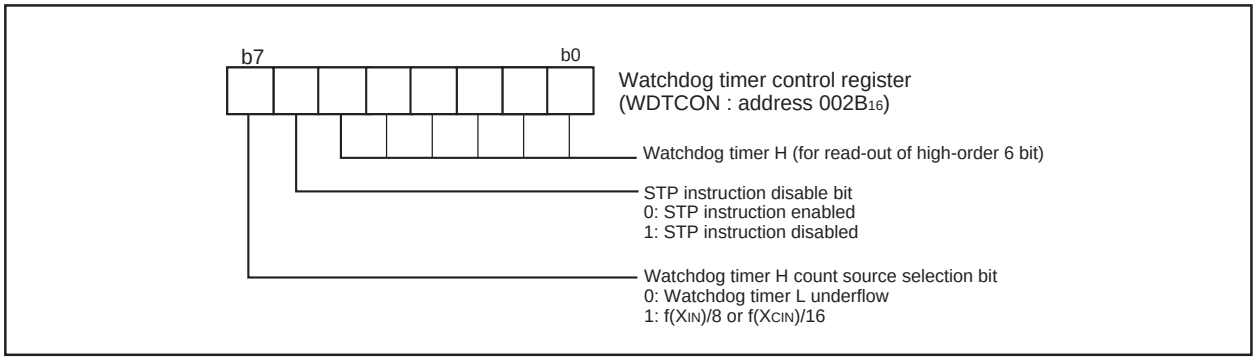


Fig. 65 Structure of watchdog timer control register

Buzzer Output Circuit

The 38B5 group has a buzzer output circuit. One of 1 kHz, 2 kHz and 4 kHz (at X_{IN} = 4.19 MHz) frequencies can be selected by the buzzer output control register (address 0EFD16). Either $P4_3/BUZ0_1$ or $P2_0/BUZ0_2/FLD_0$ can be selected as a buzzer output port by the output port selection bits (b2 and b3 of address 0EFD16).
The buzzer output is controlled by the buzzer output ON/OFF bit (b4).

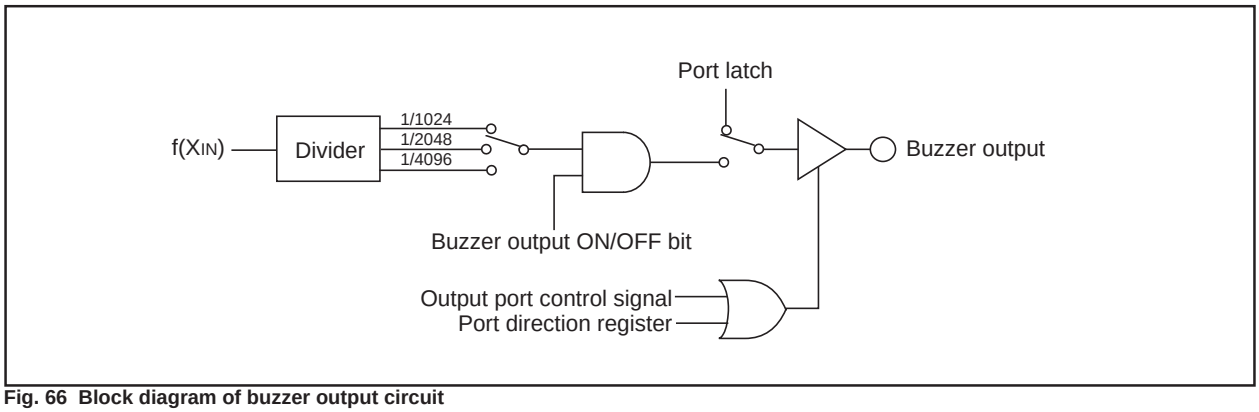


Fig. 66 Block diagram of buzzer output circuit

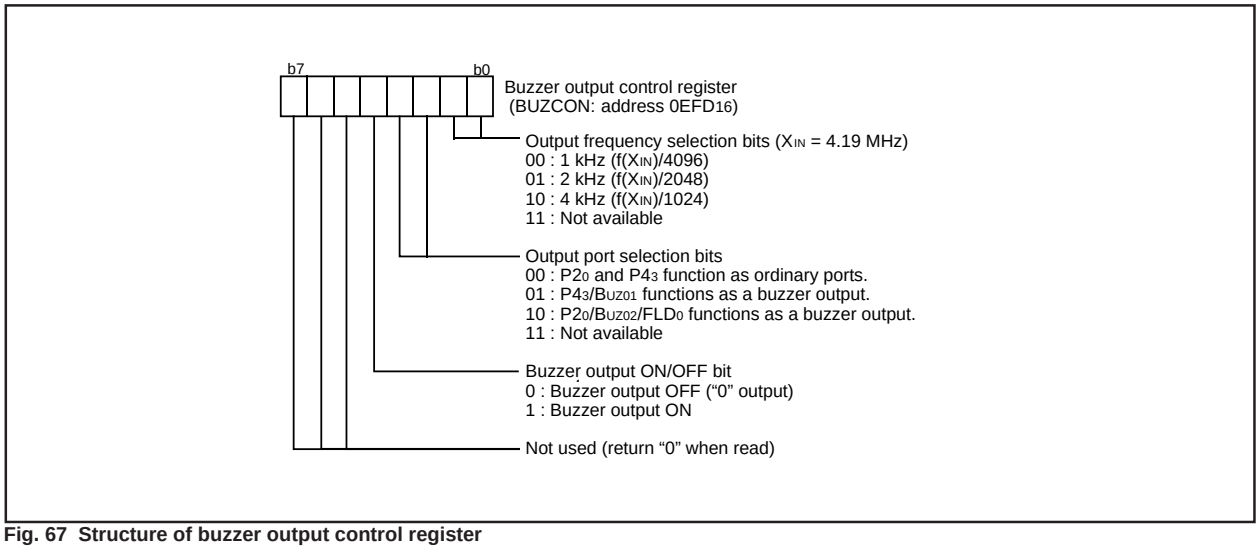


Fig. 67 Structure of buzzer output control register

Reset Circuit

To reset the microcomputer, $\overline{\text{RESET}}$ pin should be held at an "L" level for 2 µs or more. Then the $\overline{\text{RESET}}$ pin is returned to an "H" level (the power source voltage should be between 2.7 V and 5.5 V, and the oscillation should be stable), reset is released. After the reset is completed, the program starts from the address contained in address $FFFD_{16}$ (high-order byte) and address $FFFC_{16}$ (low-order byte). Make sure that the reset input voltage is less than 0.5 V for V_{CC} of 2.7 V (switching to the high-speed mode, a power source voltage must be between 4.0 V and 5.5 V).

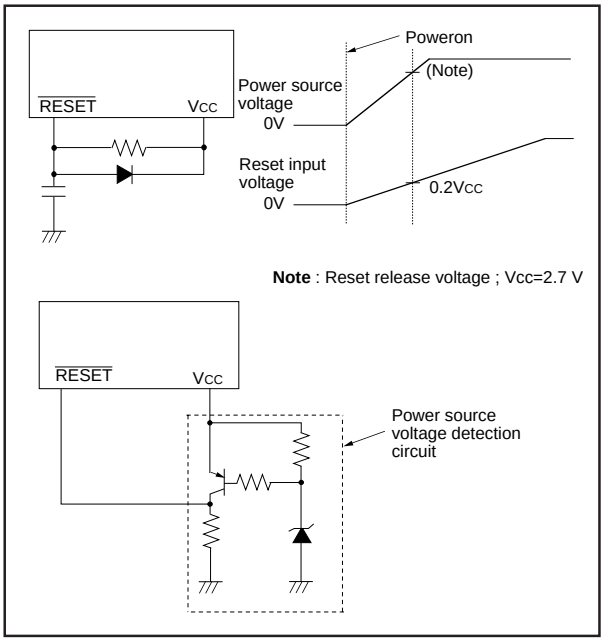


Fig. 68 Reset circuit example

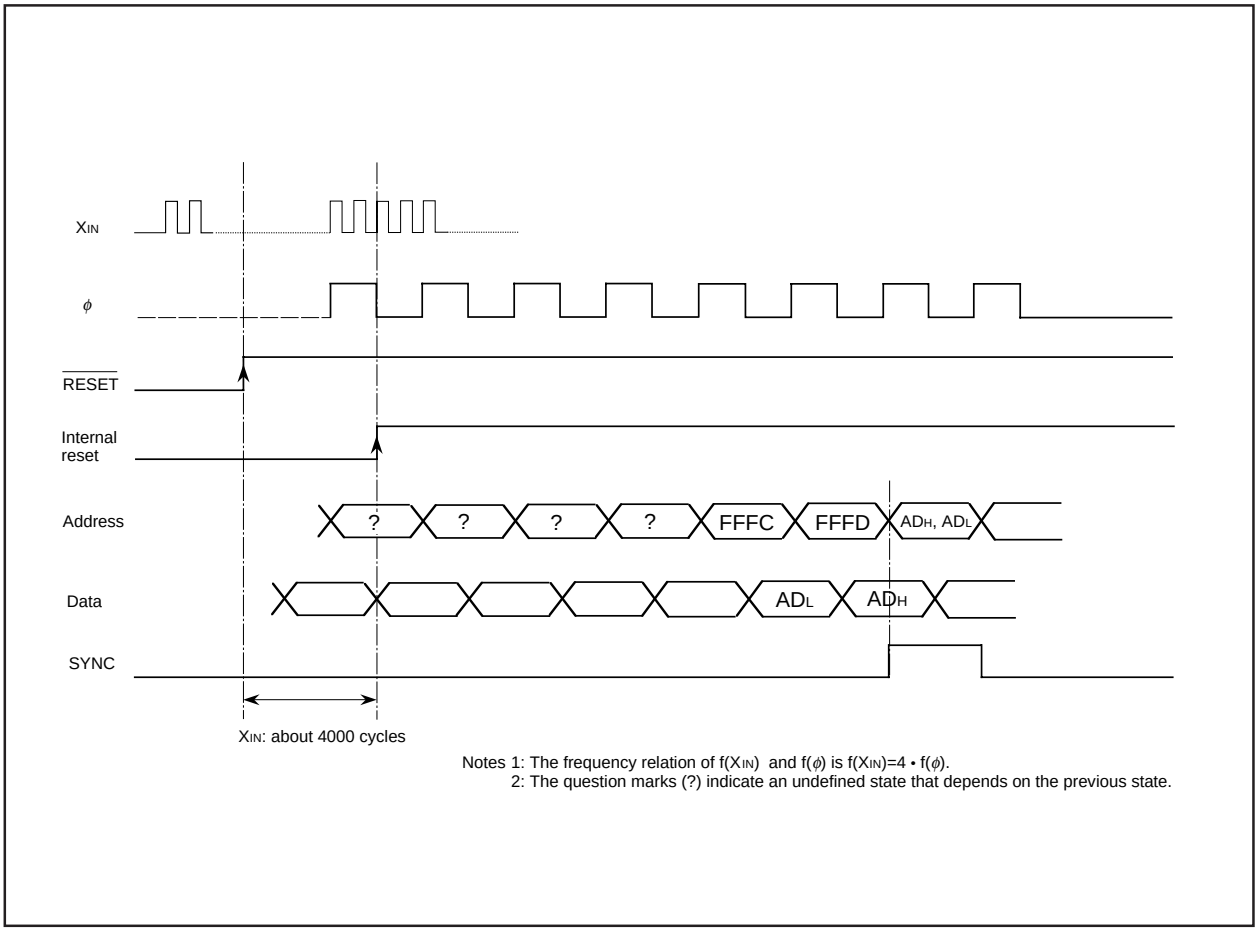


Fig. 69 Reset sequence

	Address	Register contents
(1) Port P0	0000_{16}	00_{16}
(2) Port P0 direction register	0001_{16}	00_{16}
(3) Port P1	0002_{16}	00_{16}
(4) Port P2	0004_{16}	00_{16}
(5) Port P2 direction register	0005_{16}	00_{16}
(6) Port P3	0006_{16}	00_{16}
(7) Port P4	0008_{16}	00_{16}
(8) Port P4 direction register	0009_{16}	00_{16}
(9) Port P5	$000A_{16}$	00_{16}
(10) Port P5 direction register	$000B_{16}$	00_{16}
(11) Port P6	$000C_{16}$	00_{16}
(12) Port P6 direction register	$000D_{16}$	00_{16}
(13) Port P7	$000E_{16}$	00_{16}
(14) Port P7 direction register	$000F_{16}$	00_{16}
(15) Port P8	0010_{16}	00_{16}
(16) Port P8 direction register	0011_{16}	00_{16}
(17) Port P9	0012_{16}	00_{16}
(18) Port P9 direction register	0013_{16}	00_{16}
(19) UART control register	0017_{16}	80_{16}
(20) Serial I/O1 control register 1	0019_{16}	00_{16}
(21) Serial I/O1 control register 2	$001A_{16}$	00_{16}
(22) Serial I/O1 control register 3	$001C_{16}$	00_{16}
(23) Serial I/O2 control register	$001D_{16}$	00_{16}
(24) Serial I/O2 status register	$001E_{16}$	80_{16}
(25) Timer 1	0020_{16}	FF_{16}
(26) Timer 2	0021_{16}	01_{16}
(27) Timer 3	0022_{16}	FF_{16}
(28) Timer 4	0023_{16}	FF_{16}
(29) Timer 5	0024_{16}	FF_{16}
(30) Timer 6	0025_{16}	FF_{16}
(31) PWM control register	0026_{16}	00_{16}
(32) Timer 12 mode register	0028_{16}	00_{16}

	Address	Register contents
(33) Timer 34 mode register	0029_{16}	00_{16}
(34) Timer 56 mode register	$002A_{16}$	00_{16}
(35) Watchdog timer control register	$002B_{16}$	$3F_{16}$
(36) Timer X (low-order)	$002C_{16}$	FF_{16}
(37) Timer X (high-order)	$002D_{16}$	FF_{16}
(38) Timer X mode register 1	$002E_{16}$	00_{16}
(39) Timer X mode register 2	$002F_{16}$	00_{16}
(40) Interrupt interval determination control register	0031_{16}	00_{16}
(41) A-D control register	0032_{16}	10_{16}
(42) Interrupt source switch register	0039_{16}	00_{16}
(43) Interrupt edge selection register	$003A_{16}$	00_{16}
(44) CPU mode register	$003B_{16}$	0 1 0 0 1 0 0 0
(45) Interrupt request register 1	$003C_{16}$	00_{16}
(46) Interrupt request register 2	$003D_{16}$	00_{16}
(47) Interrupt control register 1	$003E_{16}$	00_{16}
(48) Interrupt control register 2	$003F_{16}$	00_{16}
(49) Pull-up control register 1	$0EF0_{16}$	00_{16}
(50) Pull-up control register 2	$0EF1_{16}$	00_{16}
(51) P1FLDRAM write disable register	$0EF2_{16}$	00_{16}
(52) P3FLDRAM write disable register	$0EF3_{16}$	00_{16}
(53) FLDC mode register	$0EF4_{16}$	00_{16}
(54) Tdisp time set register	$0EF5_{16}$	00_{16}
(55) Toff1 time set register	$0EF6_{16}$	FF_{16}
(56) Toff2 time set register	$0EF7_{16}$	FF_{16}
(57) Port P0FLD/port switch register	$0EF9_{16}$	00_{16}
(58) Port P2FLD/port switch register	$0EFA_{16}$	00_{16}
(59) Port P8FLD/port switch register	$0EFB_{16}$	00_{16}
(60) Port P8FLD output control register	$0EFC_{16}$	00_{16}
(61) Buzzer output control register	$0EFD_{16}$	00_{16}
(62) Processor status register	(PS)	X X X X X 1 X X
(63) Program counter	(PC_H)	$FFFD_{16}$ contents
	(PC_L)	$FFFC_{16}$ contents

×: Not fixed
Since the initial values for other than above mentioned registers and RAM contents are indefinite at reset, they must be set.

Fig. 70 Internal status at reset

Clock Generating Circuit

The 38B5 group has two built-in oscillation circuits. An oscillation circuit can be formed by connecting a resonator between XIN and XOUT (XCIN and XCOUT). Use the circuit constants in accordance with the resonator manufacturer's recommended values. No external resistor is needed between XIN and XOUT since a feedback resistor exists on-chip. However, an external feedback resistor is needed between XCIN and XCOUT.

Immediately after power on, only the XIN oscillation circuit starts oscillating, and XCIN and XCOUT pins function as I/O ports.

●Frequency control

(1) Middle-speed mode

The internal system clock is the frequency of XIN divided by 4. After reset, this mode is selected.

(2) High-speed mode

The internal system clock is the frequency of XIN.

(3) Low-speed mode

The internal system clock is the frequency of XCIN divided by 2.

■Note

If you switch the mode between middle/high-speed and low-speed, stabilize both XIN and XCIN oscillations. The sufficient time is required for the sub clock to stabilize, especially immediately after power on and at returning from stop mode. When switching the mode between middle/high-speed and low-speed, set the frequency on condition that f(XIN) > 3f(XCIN).

(4) Low power consumption mode

The low power consumption operation can be realized by stopping the main clock XIN in low-speed mode. To stop the main clock, set bit 5 of the CPU mode register to "1." When the main clock XIN is restarted (by setting the main clock stop bit to "0"), set enough time for oscillation to stabilize.

By clearing furthermore the XCOUT drivability selection bit (b3) of CPU mode register to "0," low power consumption operation of less than 200 µA (f(XCIN) = 32 kHz) can be realized by reducing the drivability between XCIN and XCOUT. At reset or during STP instruction execution this bit is set to "1" and a strong drivability that has an easy oscillation start is set.

●Oscillation control

(1) Stop mode

If the STP instruction is executed, the internal system clock stops at an "H" level, and XIN and XCIN oscillators stop. Timer 1 is set to "FF16" and timer 2 is set to "0116."

Either XIN divided by 8 or XCIN divided by 16 is input to timer 1 as count source, and the output of timer 1 is connected to timer 2. The bits of the timer 12 mode register are cleared to "0." Set the interrupt enable bits of the timer 1 and timer 2 to disabled ("0") before executing the STP instruction. Oscillator restarts when an external interrupt is received, but the internal system clock is not supplied to the CPU until timer 1 underflows. This allows time for the clock circuit oscillation to stabilize.

(2) Wait mode

If the WIT instruction is executed, the internal system clock stops at an "H" level. The states of XIN and XCIN are the same as the state before executing the WIT instruction. The internal system clock restarts at reset or when an interrupt is received. Since the oscillator does not stop, normal operation can be started immediately after the clock is restarted.

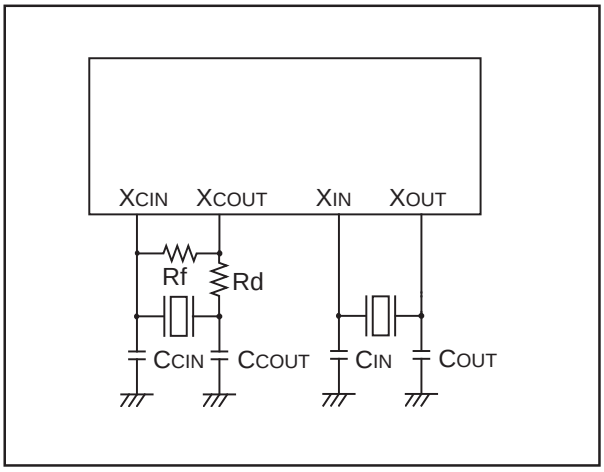


Fig. 71 Ceramic resonator circuit

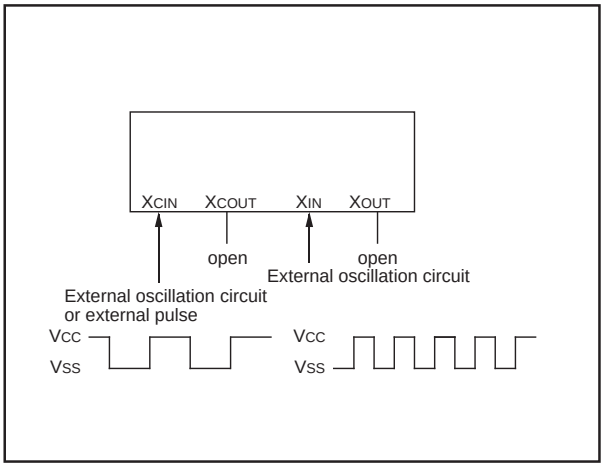
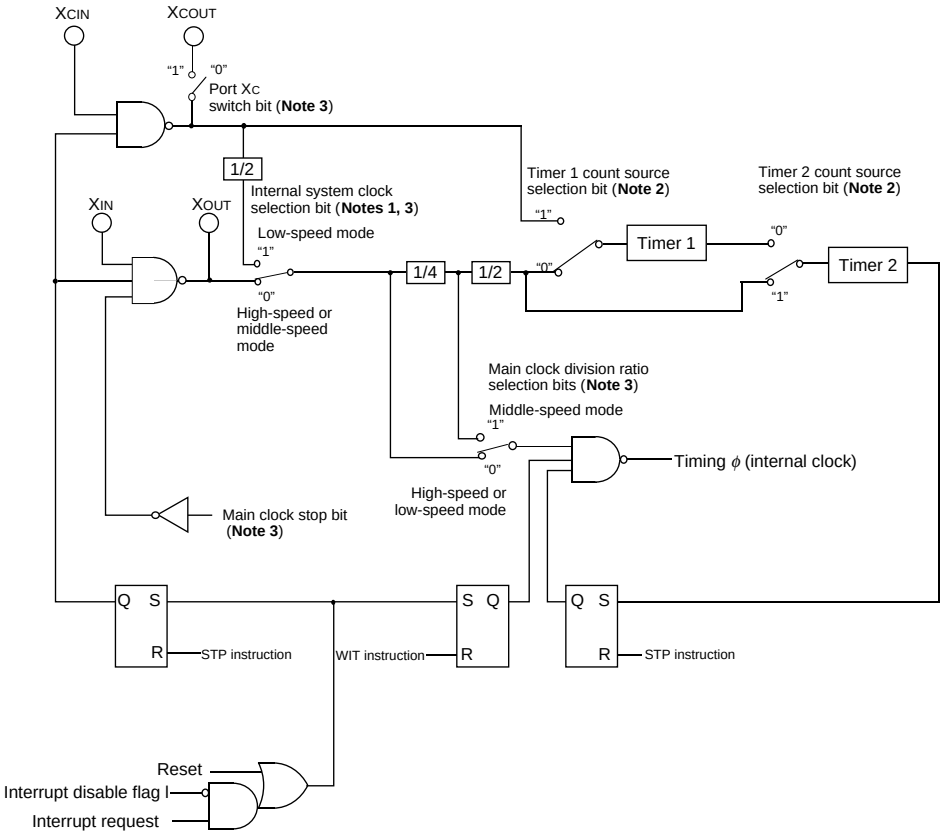
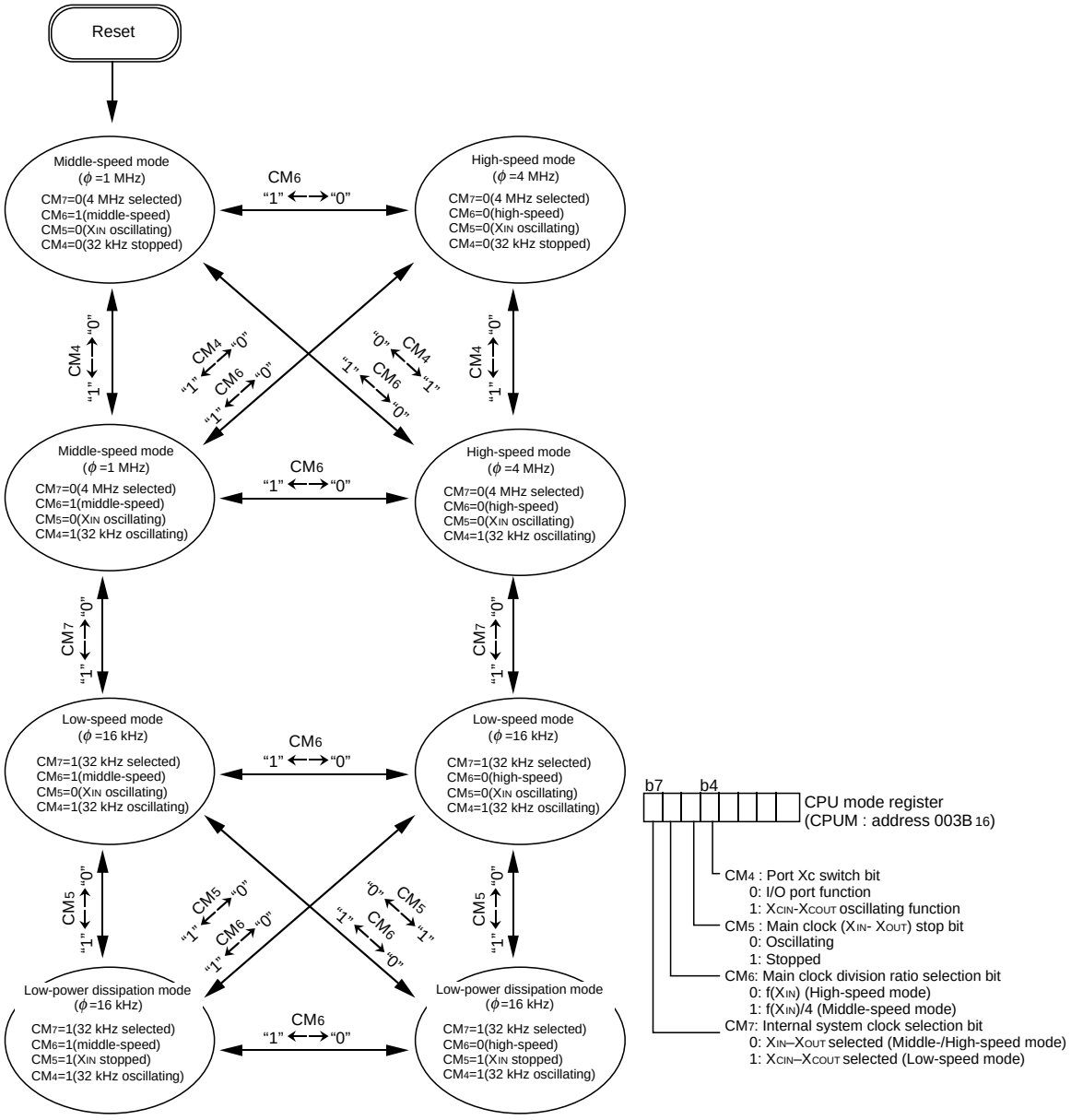


Fig. 72 External clock input circuit



Notes 1: When low-speed mode is selected, set the port Xc switch bit (b4) to "1."
2: Refer to the structure of the timer 12 mode register.
3: Refer to the structure of the CPU mode register.

Fig. 73 Clock generating circuit block diagram



Notes 1: Switch the mode by the allows shown between the mode blocks. (Do not switch between the mode directly without an allow.)
2: The all modes can be switched to the stop mode or the wait mode and return to the source mode when the stop mode or the wait mode is ended.
3: Timer operates in the wait mode.
4: When the stop mode is ended, a delay of approximately 1 ms occurs by Timer 1 in middle-/high-speed mode.
5: When the stop mode is ended, a delay of approximately 0.25 s occurs by Timer 1 in low-speed mode.
6: The example assumes that 4 MHz is being applied to the X IN pin and 32 kHz to the X CIN pin. ϕ indicates the internal system clock.

Fig. 74 State transitions of system clock

NOTES ON PROGRAMMING

Processor Status Register

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

Interrupts

The contents of the interrupt request bits do not change immediately after they have been written. After writing to an interrupt request register, execute at least one instruction before performing a BBC or BBS instruction.

Decimal Calculations

- To calculate in decimal notation, set the decimal mode flag (D) to "1," then execute an ADC or SBC instruction. Only the ADC and SBC instructions yield proper decimal results. After executing an ADC or SBC instruction, execute at least one instruction before executing a SEC, CLC, or CLD instruction.
- In decimal mode, the values of the negative (N), overflow (V), and zero (Z) flags are invalid.

Timers

If a value n (between 0 and 255) is written to a timer latch, the frequency division ratio is 1/(n+1).

Multiplication and Division Instructions

- The index X mode (T) and the decimal mode (D) flags do not affect the MUL and DIV instruction.
- The execution of these instructions does not change the contents of the processor status register.

Ports

The contents of the port direction registers cannot be read. The following cannot be used:

- The data transfer instruction (LDA, etc.)
- The operation instruction when the index X mode flag (T) is "1"
- The addressing mode which uses the value of a direction register as an index
- The bit-test instruction (BBC or BBS, etc.) to a direction register
- The read-modify-write instructions (ROR, CLB, or SEB, etc.) to a direction register.

Use instructions such as LDM and STA, etc., to set the port direction registers.

Serial I/O

- Using an external clock

  When using an external clock, input "H" to the external clock input pin and clear the serial I/O interrupt request bit before executing serial I/O transfer and serial I/O automatic transfer.
- Using an internal clock

  When using an internal clock, set the synchronous clock to the internal clock, then clear the serial I/O interrupt request bit before executing a serial I/O transfer and serial I/O automatic transfer.

A-D Converter

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

Therefore, make sure that f(X_{IN}) is at least on 250 kHz during an A-D conversion.

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

Instruction Execution Time

The instruction execution time is obtained by multiplying the frequency of the internal system clock by the number of cycles needed to execute an instruction.

The number of cycles required to execute an instruction is shown in the list of machine instructions.

The frequency of the internal system clock is the same of the X_{IN} frequency in high-speed mode.

At STP Instruction Release

At the STP instruction release, all bits of the timer 12 mode register are cleared.

The X_{COUT} drivability selection bit (the CPU mode register) is set to "1" (high drive) in order to start oscillating.

NOTES ON USE

Notes on Built-in EPROM Version

The $P4_7$ pin of the One Time PROM version or the EPROM version functions as the power source input pin of the internal EPROM.

Therefore, this pin is set at low input impedance, thereby being affected easily by noise.

To prevent a malfunction due to noise, insert a resistor (approx. 5 kΩ) in series with the $P4_7$ pin.

DATA REQUIRED FOR MASK ORDERS

The following are necessary when ordering a mask ROM production:

(1) Mask ROM Order Confirmation Form
(2) Mark Specification Form
(3) Data to be written to ROM, in EPROM form (three identical copies)

DATA REQUIRED FOR ROM WRITING ORDERS

The following are necessary when ordering a ROM writing:

(1) ROM Writing Confirmation Form
(2) Mark Specification Form
(3) Data to be written to ROM, in EPROM form (three identical copies)

ROM PROGRAMMING METHOD

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

Table 11 Special programming adapter

Package	Name of Programming Adapter
80P6N-A	PCA7438F-80A
80D0	PCA7438L-80A

The PROM of the blank One Time PROM version is not tested or screened in the assembly process and following processes. To ensure proper operation after programming, the procedure shown in Figure 75 is recommended to verify programming.

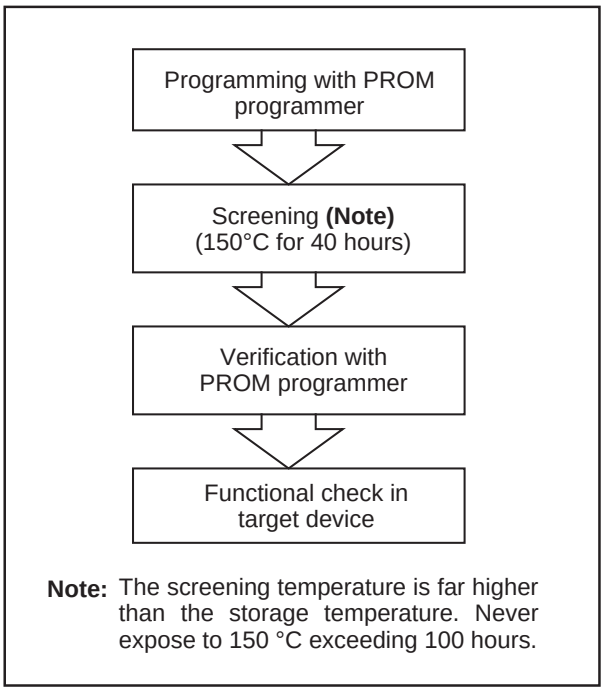


Fig. 75 Programming and testing of One Time PROM version

MASK OPTION OF PULL-DOWN RESISTOR
(object product: M38B5XMXH-XXXFP)

Whether built-in pull-down resistors are connected or not to high-breakdown voltage ports P20 to P27 and P80 to P83 can be specified in ordering mask ROM. The option type can be specified from among 8 types; A to G, P as shown Table 12.

Table 12 Mask option type of pull-down resistor

Option type	Connective port of pull-down resistor (connected at "1" writing)												Restriction
	P20	P21	P22	P23	P24	P25	P26	P27	P80	P81	P82	P83	
A ($41)													
B ($42)							1	1					
C ($43)					1	1	1	1					
D ($44)			1	1	1	1	1	1					
E ($45)	1	1	1	1	1	1	1	1					
F ($46)	1	1	1	1	1	1	1	1	1	1			
G ($47)	1	1	1	1	1	1	1	1	1	1	1	1	
P ($50)	1	1	1	1	1	1	1	1					**(Note 4)**

Notes 1: The electrical characteristics of high-breakdown voltage ports P20 to P27 and P80 to P83's built-in pull-down resistors are the same as that of high-breakdown voltage ports P00 to P07.
2: The absolute maximum ratings of power dissipation may be exceed owing to the number of built-in pull-down resistor. After calculating the power dissipation, specify the option type.
3: One time PROM version and EPROM version cannot be specified whether built-in pull-down resistors are connected or not likewise option type A.
4: INT3 function and CNTR1 function cannot be used in the option type P.

Power Dissipation Calculating Method

● **Fixed number depending on microcomputer's standard**
- VOH output fall voltage of high-breakdown port
  2 V (max.); | Current value | = at 18 mA
- Resistor value 43 V / 900 μA = 48 kΩ (min.)
- Power dissipation of internal circuit (CPU, ROM, RAM etc.) = 5 V × 15 mA = 75 mW

● **Fixed number depending on use condition**
- Apply voltage to VEE pin: Vcc – 45 V
- Timing number a; digit number b; segment number c
- Ratio of Toff time corresponding Tdisp time: 1/16
- Turn ON segment number during repeat cycle: d
- All segment number during repeat cycle: c (= a × c)
- Total number of built-in resistor: for digit; f, for segment; g
- Digit pin current value h (mA)
- Segment pin current value i (mA)

(1) Digit pin power dissipation
{h × b × (1–Toff/Tdisp) × voltage} / a
(2) Segment pin power dissipation
{i × d × (1–Toff/Tdisp) × voltage} / a
(3) Pull-down resistor power dissipation (digit)
{power dissipation per 1 digit × (b × f / b) × (1–Toff/Tdisp) } / a
(4) Pull-down resistor power dissipation (segment)
{power dissipation per 1 segment × (d × g / c) × (1–Toff/Tdisp) } / a
(5) Internal circuit power dissipation (CPU, ROM, RAM etc.) = 75 mW

(1) + (2)+ (3) + (4) + (5) = X mW

Power Dissipation Calculating example 1

● **Fixed number depending on microcomputer's standard**
- VOH output fall voltage of high-breakdown port
  2 V (max.); | Current value | = at 18 mA
- Resistor value 43 V / 900 μA = 48 kΩ (min.)
- Power dissipation of internal circuit (CPU, ROM, RAM etc.) = 5 V × 15 mA = 75 mW

● **Fixed number depending on use condition**
- Apply voltage to VEE pin: Vcc – 45 V
- Timing number 17; digit number 16; segment number 20
- Ratio of Toff time corresponding Tdisp time: 1/16
- Turn ON segment number during repeat cycle: 31
- All segment number during repeat cycle: 340 (= 17 × 20)
- Total number of built-in resistor: for digit; 16, for segment; 20
- Digit pin current value: 18 (mA)
- Segment pin current value: 3 (mA)

(1) Digit pin power dissipation
{18 × 16 × (1–1/16) × 2} / 17 = 31.77 mW
(2) Segment pin power dissipation
{3 × 31 × (1–1/16) × 2} / 17 = 10.26 mW
(3) Pull-down resistor power dissipation (digit)
$(45 – 2)^2$ /48 × (16 × 16/16) × (1 – 1/16) / 17 = 33.99 mW
(4) Pull-down resistor power dissipation (segment)
$(45 – 2)^2$ /48 × (31 × 20/20) × (1 – 1/16) / 17 = 65.86 mW
(5) Internal circuit power dissipation (CPU, ROM, RAM etc.) = 75 mW

(1) + (2)+ (3) + (4) + (5) = 217 mW

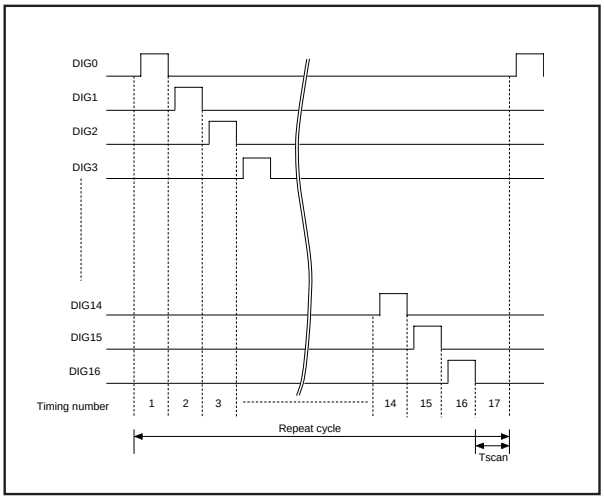


Fig. 76 Digit timing waveform (1)

Power Dissipation Calculating example 2 (when 2 or more digit is turned ON at same time)

● **Fixed number depending on microcomputer's standard**

• V_{OH} output fall voltage of high-breakdown port
2 V (max.); | Current value | = at 18 mA
• Resistor value 43 V / 900 μA = 48 kΩ (min.)
• Power dissipation of internal circuit (CPU, ROM, RAM etc.) = 5 V × 15 mA = 75 mW

● **Fixed number depending on use condition**

• Apply voltage to V_{EE} pin: Vcc – 45 V
• Timing number 11; digit number 12; segment number 24
• Ratio of Toff time corresponding Tdisp time: 1/16
• Turn ON segment number during repeat cycle: 114
• All segment number during repeat cycle: 264 (= 11 × 24)
• Total number of built-in resistor: for digit; 10, for segment; 22
• Digit pin current value: 18 (mA)
• Segment pin current value: 3 (mA)

(1) Digit pin power dissipation
{18 × 12 × (1–1/16) × 2} / 11 = 36.82 mW
(2) Segment pin power dissipation
{3 × 114 × (1–1/16) × 2} / 11 = 58.30 mW
(3) Pull-down resistor power dissipation (digit)
$(45 - 2)^2$ /48 × (12 × 10/12) × (1 – 1/16) / 11 = 32.84 mW
(4) Pull-down resistor power dissipation (segment)
$(45 - 2)^2$ /48 × (114 × 22/24) × (1 – 1/16) / 11 = 343.08 mW
(5) Internal circuit power dissipation (CPU, ROM, RAM etc.) = 75 mW

(1) + (2)+ (3) + (4) + (5) = 547 mW

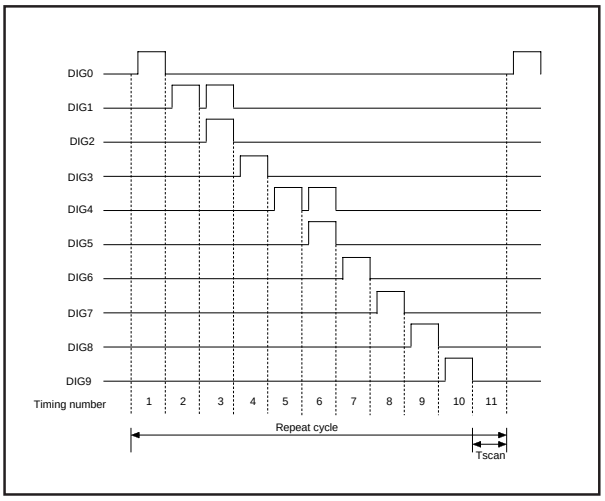


Fig. 77 Digit timing waveform (2)

FUNCTIONAL DESCRIPTION SUPPLEMENT

Interrupt

38B5 group permits interrupts on the basis of 21 sources.
It is vector interrupts with a fixed priority system. Accordingly, when two or more interrupt requests occur during the same sampling, the higher-priority interrupt is accepted first. This priority is determined by hardware, but various priority processing can be performed by software, using an interrupt enable bit and an interrupt disable flag.
For interrupt sources, vector addresses and interrupt priority, refer to Table 13.

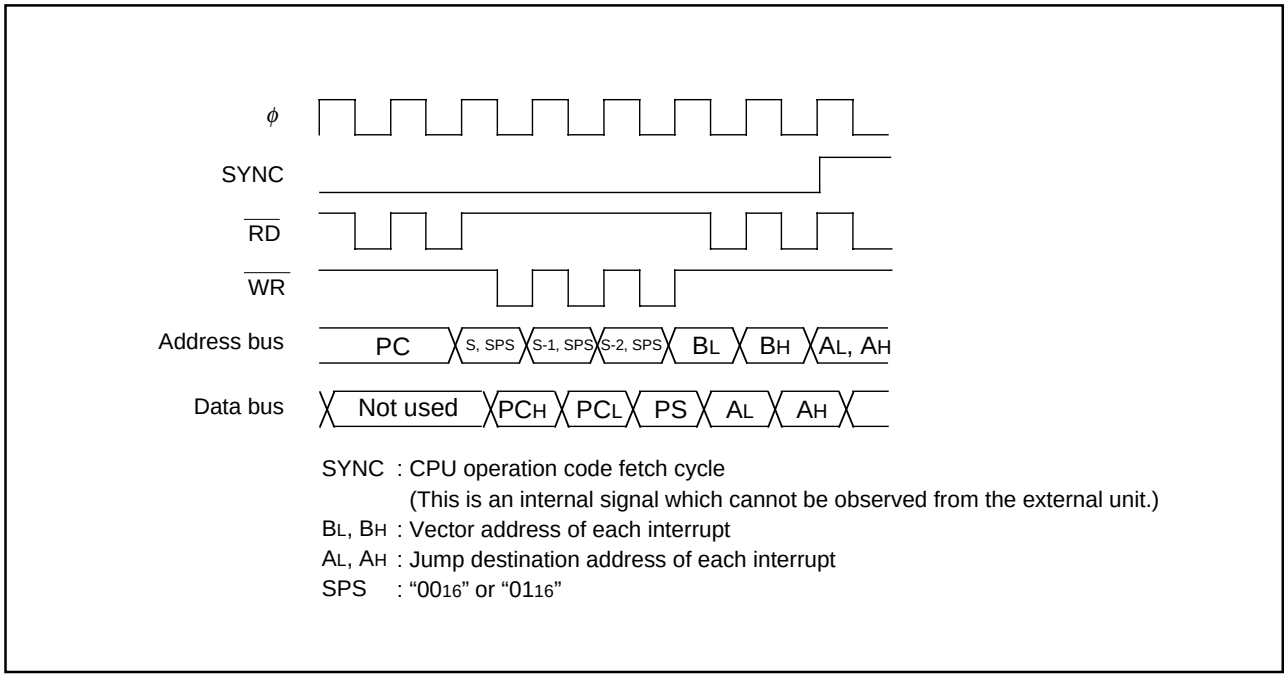
Table 13 Interrupt sources, vector addresses and interrupt priority

Interrupt source	Priority	Vector Addresses **(Note 1)**		Remarks
		High	Low	
Reset **(Note 2)**	1	$FFFD_{16}$	$FFFC_{16}$	Non-maskable
INT_0	2	$FFFB_{16}$	$FFFA_{16}$	External interrupt (active edge selectable)
INT_1	3	$FFF9_{16}$	$FFF8_{16}$	External interrupt (active edge selectable)
INT_2	4	$FFF7_{16}$	$FFF6_{16}$	External interrupt (active edge selectable)
Remote control/counter overflow				Valid when interrupt interval determination is operating
Serial I/O1	5	$FFF5_{16}$	$FFF4_{16}$	Valid when serial I/O1 ordinary mode is selected
Serial I/O1 automatic transfer				Valid when serial I/O1 automatic transfer mode is selected
Timer X	6	$FFF3_{16}$	$FFF2_{16}$	
Timer 1	7	$FFF1_{16}$	$FFF0_{16}$	
Timer 2	8	$FFEF_{16}$	$FFEE_{16}$	STP release timer underflow
Timer 3	9	$FFED_{16}$	$FFEC_{16}$	
Timer 4	10	$FFEB_{16}$	$FFEA_{16}$	**(Note 3)**
Timer 5	11	$FFE9_{16}$	$FFE8_{16}$	
Timer 6	12	$FFE7_{16}$	$FFE6_{16}$	
Serial I/O2 receive	13	$FFE5_{16}$	$FFE4_{16}$	
INT_3	14	$FFE3_{16}$	$FFE2_{16}$	External interrupt (active edge selectable) **(Note 4)**
Serial I/O2 transmit				
INT_4	15	$FFE1_{16}$	$FFE0_{16}$	External interrupt (active edge selectable)
A-D conversion				Valid when INT_4 interrupt is selected Valid when A-D conversion is selected
FLD blanking	16	$FFDF_{16}$	$FFDE_{16}$	Valid when FLD blanking interrupt is selected
FLD digit				Valid when FLD digit interrupt is selected
BRK instruction	17	$FFDD_{16}$	$FFDC_{16}$	Non-maskable software interrupt

Notes 1 : Vector addresses contain interrupt jump destination addresses.
2 : Reset function in the same way as an interrupt with the highest priority.
3 : In the mask option type P, timer 4 interrupt whose count source is $CNTR_1$ input cannot be used.
4 : In the mask option type P, INT_3 interrupt cannot be used.

Timing After Interrupt

The interrupt processing routine begins with the machine cycle following the completion of the instruction that is currently in execution. Figure 78 shows a timing chart after an interrupt occurs, and Figure 79 shows the time up to execution of the interrupt processing routine.



SYNC : CPU operation code fetch cycle
(This is an internal signal which cannot be observed from the external unit.)
BL, BH : Vector address of each interrupt
AL, AH : Jump destination address of each interrupt
SPS : "00_{16}" or "01_{16}"

Fig. 78 Timing chart after interrupt occurs

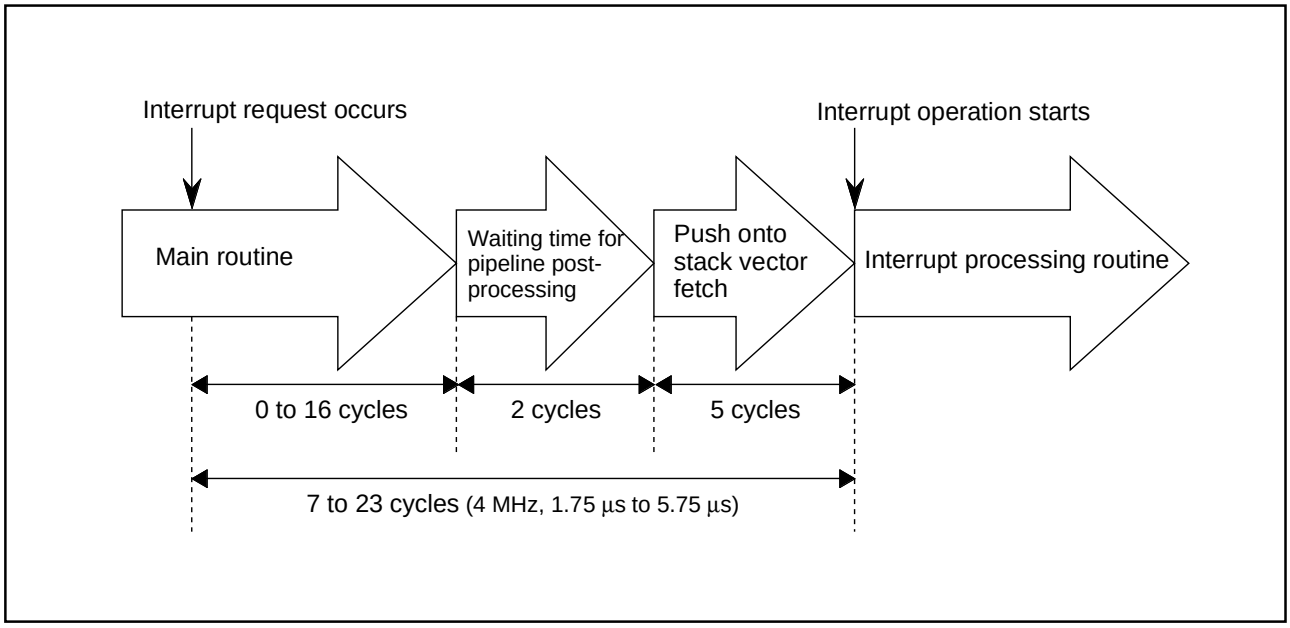


Fig. 79 Time up to execution of interrupt processing routine

A-D Converter

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

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

By repeating the above operations up to the lowest-order bit of the A-D conversion register, an analog value converts into a digital value. A-D conversion completes at 61 clock cycles (15.25 µs at f(X_{IN}) = 8 MHz) after it is started, and the result of the conversion is stored into the A-D conversion register.

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

Table 14 Relative formula for a reference voltage V_{REF} of A-D converter and V_{ref}

When n = 0	$V_{ref} = 0$
When n = 1 to 1023	$V_{ref} = \frac{V_{REF}}{1024} \times n$

n: Value of A-D converter (decimal numeral)

Table 15 Change of A-D conversion register during A-D conversion

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

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

Figures 80 shows the A-D conversion equivalent circuit, and Figure 81 shows the A-D conversion timing chart.

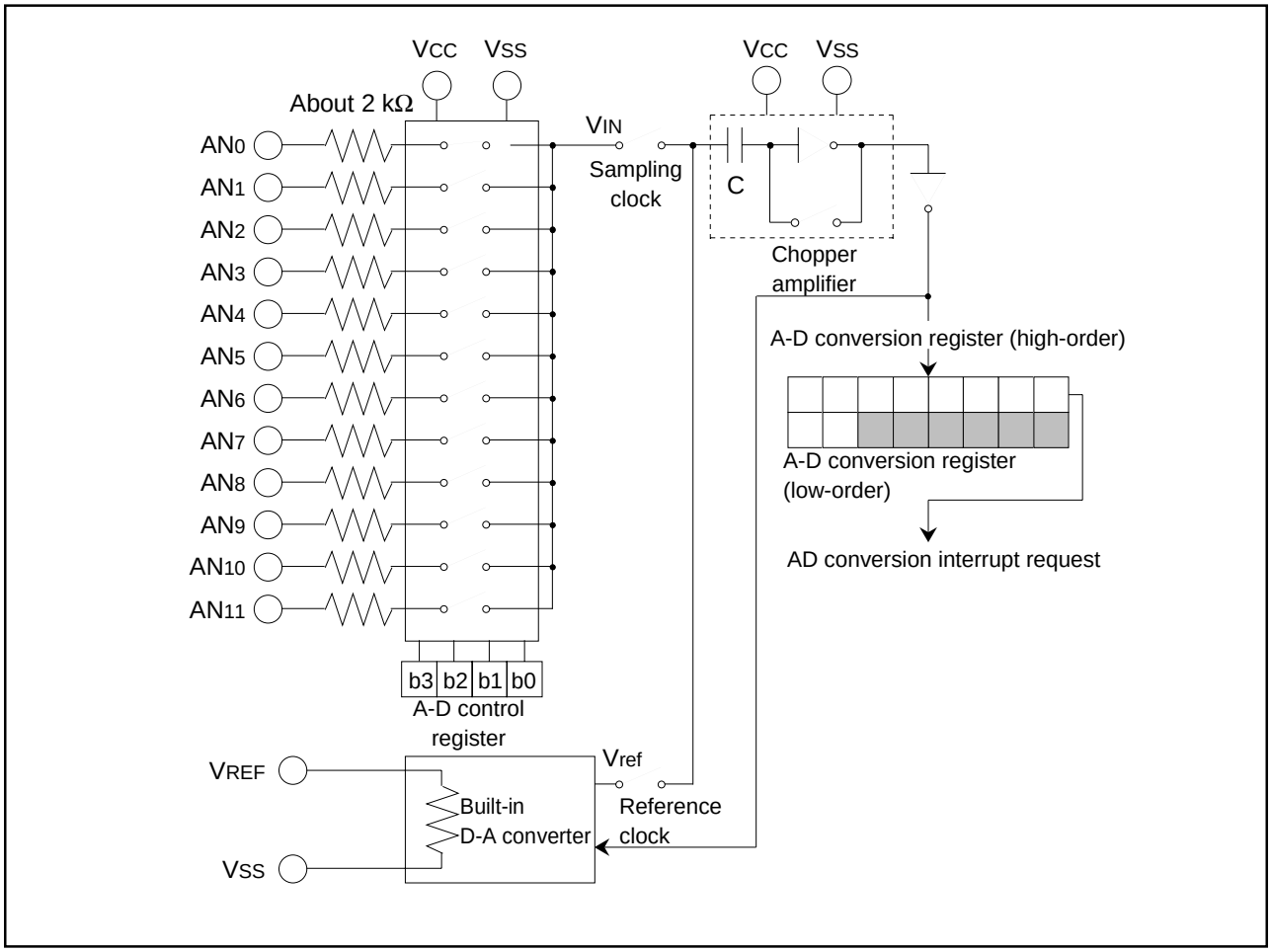


Fig. 80 A-D conversion equivalent circuit

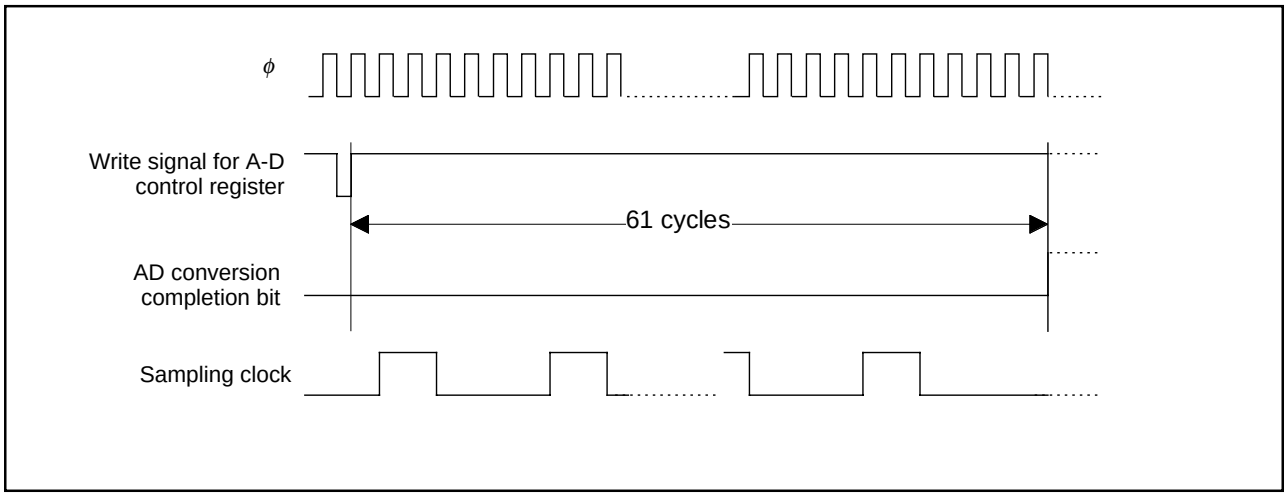


Fig. 81 A-D conversion timing chart

CHAPTER 2
APPLICATION

2.1 I/O port

This paragraph describes the setting method of I/O port relevant registers, notes etc.

2.1.1 Memory assignment

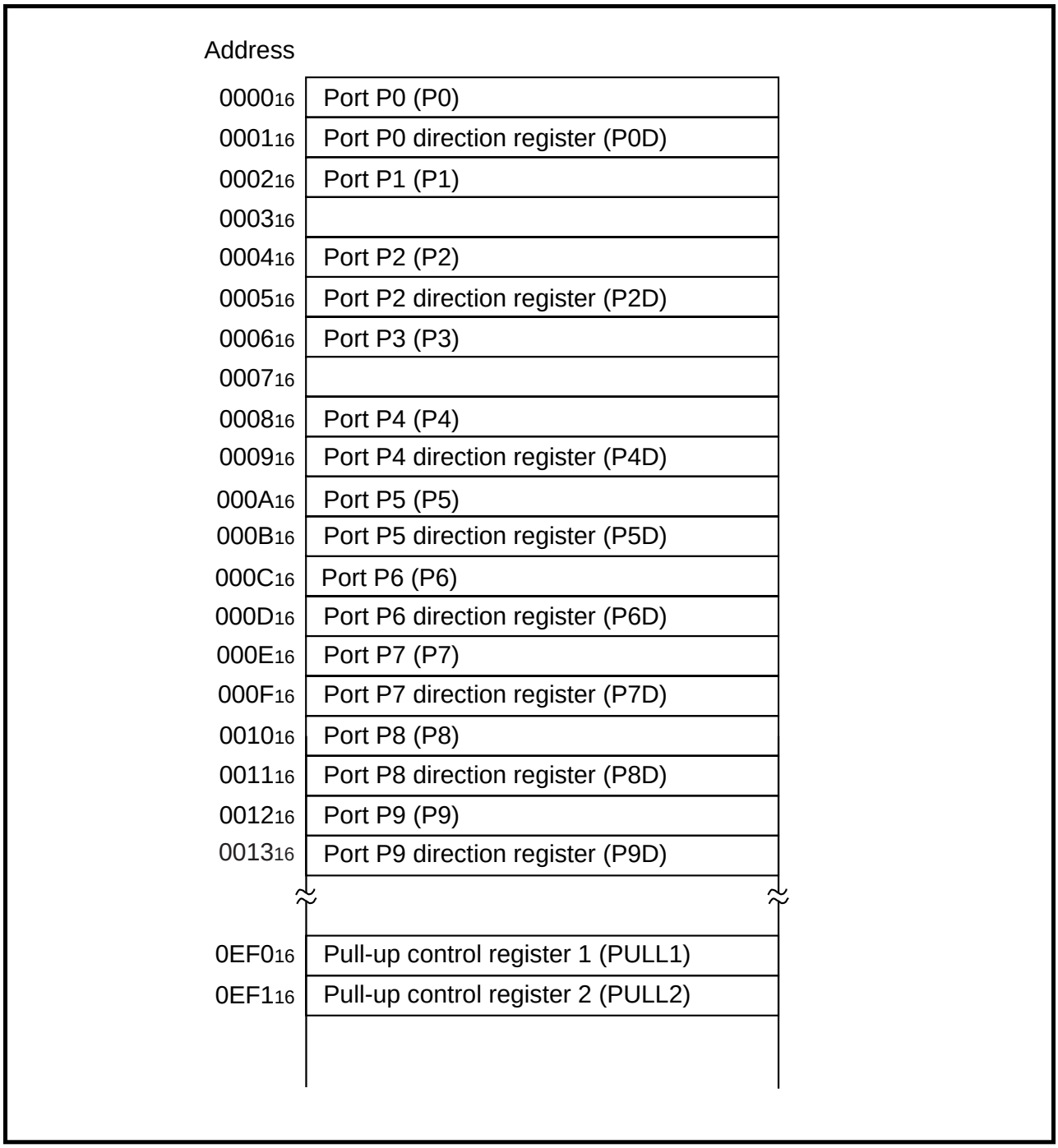


Fig. 2.1.1 Memory assignment of I/O port relevant registers

2.1.2 Relevant registers

Port Pi

b7 b6 b5 b4 b3 b2 b1 b0

Port Pi (i = 0, 1, 2, 3, 4, 5, 7, 8)
(Pi: addresses 00_{16}, 02_{16}, 04_{16}, 06_{16}, 08_{16}, $0A_{16}$, $0E_{16}$, 10_{16})

b	Name	Functions	At reset	R	W
0	Port Pi_0	●In output mode Write ••••••••• Port latch Read ••••••••• Port latch ●In input mode Write ••••••••• Port latch Read ••••••••• Value of pin	0	○	○
1	Port Pi_1		0	○	○
2	Port Pi_2		0	○	○
3	Port Pi_3		0	○	○
4	Port Pi_4		0	○	○
5	Port Pi_5		0	○	○
6	Port Pi_6		0	○	○
7	Port Pi_7		0	○	○

Fig. 2.1.2 Structure of port Pi (i = 0, 1, 2, 3, 4, 5, 7, 8)

Port P6

b7 b6 b5 b4 b3 b2 b1 b0

Port P6
(P6: address $0C_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P6_0$	●In output mode Write ••••••••• Port latch Read ••••••••• Port latch ●In input mode Write ••••••••• Port latch Read ••••••••• Value of pin	0	○	○
1	Port $P6_1$		0	○	○
2	Port $P6_2$		0	○	○
3	Port $P6_3$		0	○	○
4	Port $P6_4$		0	○	○
5	Port $P6_5$		0	○	○
6	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
7			0	×	×

Fig. 2.1.3 Structure of port P6

Port P9

b7 b6 b5 b4 b3 b2 b1 b0

Port P9
(P9: address 12_{16})

b	Name	Functions	At reset	R	W
0	Port $P9_0$	●In output mode Write ••••••••• Port latch Read ••••••••• Port latch	0	○	○
1	Port $P9_1$	●In input mode Write ••••••••• Port latch Read ••••••••• Value of pin	0	○	○
2	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
3			0	×	×
4			0	×	×
5			0	×	×
6			0	×	×
7			0	×	×

Fig. 2.1.4 Structure of port P9

Port Pi direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port Pi direction register (i = 0, 2, 4, 5, 7, 8)
(PiD: addresses 01_{16}, 05_{16}, 09_{16}, $0B_{16}$, $0F_{16}$, 11_{16})

b	Name	Functions	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 **(Note)**	0	×	○

Note: Bit 7 of the port P4 direction register (address 09_{16}) does not have direction register function because $P4_7$ is input port. When writing to bit 7 of the port P4 direction register, write "0" to the bit.

Fig. 2.1.5 Structure of port Pi (i = 0, 2, 4, 5, 7, 8) direction register

Port P6 direction register

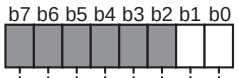
b7 b6 b5 b4 b3 b2 b1 b0

Port P6 direction register
(P6D: address $0D_{16}$)

b	Name	Functions	At reset	R	W
0	Port P6 direction register	0 : Port $P6_0$ input mode 1 : Port $P6_0$ output mode	0	×	○
1		0 : Port $P6_1$ input mode 1 : Port $P6_1$ output mode	0	×	○
2		0 : Port $P6_2$ input mode 1 : Port $P6_2$ output mode	0	×	○
3		0 : Port $P6_3$ input mode 1 : Port $P6_3$ output mode	0	×	○
4		0 : Port $P6_4$ input mode 1 : Port $P6_4$ output mode	0	×	○
5		0 : Port $P6_5$ input mode 1 : Port $P6_5$ output mode	0	×	○
6	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
7			0	×	×

Fig. 2.1.6 Structure of port P6 direction register

Port P9 direction register

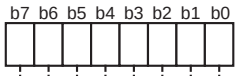


Port P9 direction register
(P9D: address 13_{16})

b	Name	Functions	At reset	R	W
0	Port P9 direction register	0 : Port $P9_0$ input mode 1 : Port $P9_0$ output mode	0	×	○
1		0 : Port $P9_1$ input mode 1 : Port $P9_1$ output mode	0	×	○
2	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
3			0	×	×
4			0	×	×
5			0	×	×
6			0	×	×
7			0	×	×

Fig. 2.1.7 Structure of port P9 direction register

Pull-up control register 1



Pull-up control register 1
(PULL1: address $0EF0_{16}$)

b	Name	Functions	At reset	R	W
0	Ports $P5_0$, $P5_1$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
1	Ports $P5_2$, $P5_3$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
2	Ports $P5_4$, $P5_5$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
3	Ports $P5_6$, $P5_7$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
4	Port $P6_1$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
5	Ports $P6_2$, $P6_3$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
6	Ports $P6_4$, $P6_5$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×

Note: The pin set to output port is cut off from pull-up control.

Fig. 2.1.8 Structure of pull-up control register 1

2.1 I/O port

Pull-up control register 2

b7 b6 b5 b4 b3 b2 b1 b0

Pull-up control register 2
(PULL2: address 0EF1$_{16}$)

b	Name	Functions	At reset	R	W
0	Ports P7$_0$, P7$_1$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
1	Ports P7$_2$, P7$_3$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
2	Ports P7$_4$, P7$_5$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
3	Ports P7$_6$, P7$_7$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
4	Ports P8$_4$, P8$_5$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
5	Ports P8$_6$, P8$_7$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
6	Ports P9$_0$, P9$_1$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×

Note: The pin set to output port is cut off from pull-up control.

Fig. 2.1.9 Structure of pull-up control register 2

2.1.3 Terminate unused pins

Table 2.1.1 Termination of unused pins

Pins	Termination
P1, P3	Open at “H” output state.
P5, P6$_1$–P6$_5$, P7, P8$_4$–P8$_7$, P9	• Set to the input mode and connect each to V_{CC} or V_{SS} through a resistor of 1 kΩ to 10 kΩ. • Set to the output mode and open at “L” or “H” output state.
P4$_0$–P4$_6$, P6$_0$	• Set to the input mode and connect each to V_{CC} or V_{SS} through a resistor of 1 kΩ to 10 kΩ. • Set to the output mode and open at “L” output state.
P0, P2, P8$_0$–P8$_3$	• Set to the input mode and connect each to V_{CC} or V_{SS} through a resistor of 1 kΩ to 10 kΩ. • Set to the output mode and open at “H” output state.
P4$_7$	Disable INT_2 interrupt and connect to V_{CC} or V_{SS} through a resistor of 1 KΩ to 10 kΩ.
V_{REF}	Open
X_{OUT}	Open (only when using external clock)
AV_{SS}, V_{EE}	Connect to V_{SS} (GND).

2.1.4 Notes on use

(1) Notes in standby state

In standby state*1 for low-power dissipation, do not make input levels of an input port and an I/O port "undefined", especially for I/O ports of the P-channel open-drain and 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, note the following points:
- External circuit
- Variation of output levels during the ordinary operation

When using built-in pull-up resistor, note on varied current values:
- When setting as an input port : Fix its input level
- When setting as an output port : Prevent current from flowing out to external

● **Reason**

Even when setting as an output port with its direction register, in the following state :
- P-channel.....when the content of the port latch is "0"
- 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. Note that the level becomes "undefined" depending on external circuits.
Accordingly, the potential which is input to the input buffer in a microcomputer is unstable in the state that input levels of a input port and an I/O port are "undefined". This may cause power source current.

✻1 standby state: stop mode by executing **STP** instruction
wait mode by executing **WIT** instruction

(2) N-channel open-drain port

$P4_0$–$P4_2$, $P4_5$, $P4_6$, $P6_0$ of N-channel open-drain output ports have the built-in hysteresis circuit for input. In standby state for low-power dissipation, do not make these pins floating state.

● **Reason**

When power sources for pull-up of these pins are cut off in standby state, these ports become floating. Accordingly, a current may flow from Vcc to Vss through the built-in hysteresis circuit.

(3) Modifying port latch of I/O port with bit managing instruction

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

● **Reason**

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

•As for bit which is set for input port:
The pin state is read in the CPU, and is written to this bit after bit managing.

•As for bit which is set for output port:
The bit value is read in the CPU, and is written to this bit after bit managing.

Note the following:

•Even when a port which is set as an output port is changed for an input port, its port latch holds the output data.

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

✻2 Bit managing instructions: **SEB** and **CLB** instructions

(4) Pull-up control

When each port which has built-in pull-up resistor (P5, $P6_1$–$P6_5$, P7, $P8_4$–$P8_7$, P9) is set to output port, pull-up control of corresponding port become invalid. (Pull-up cannot be set.)

● **Reason**

Pull-up control is valid only when each direction register is set to the input mode.

2.1.5 Termination of unused pins

(1) Terminate unused pins

① Output ports : Open

② Input ports :
Connect each pin to V_{CC} or V_{SS} through each resistor of 1 kΩ to 10 kΩ.
As for pins whose potential affects to operation modes such as pin INT or others, select the V_{CC} pin or the V_{SS} pin according to their operation mode.

③ I/O ports :

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

(2) Termination remarks

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

● **Reason**

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

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

● **Reason**

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

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

● **Reason**

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

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

2.2 Timer

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

2.2.1 Memory map

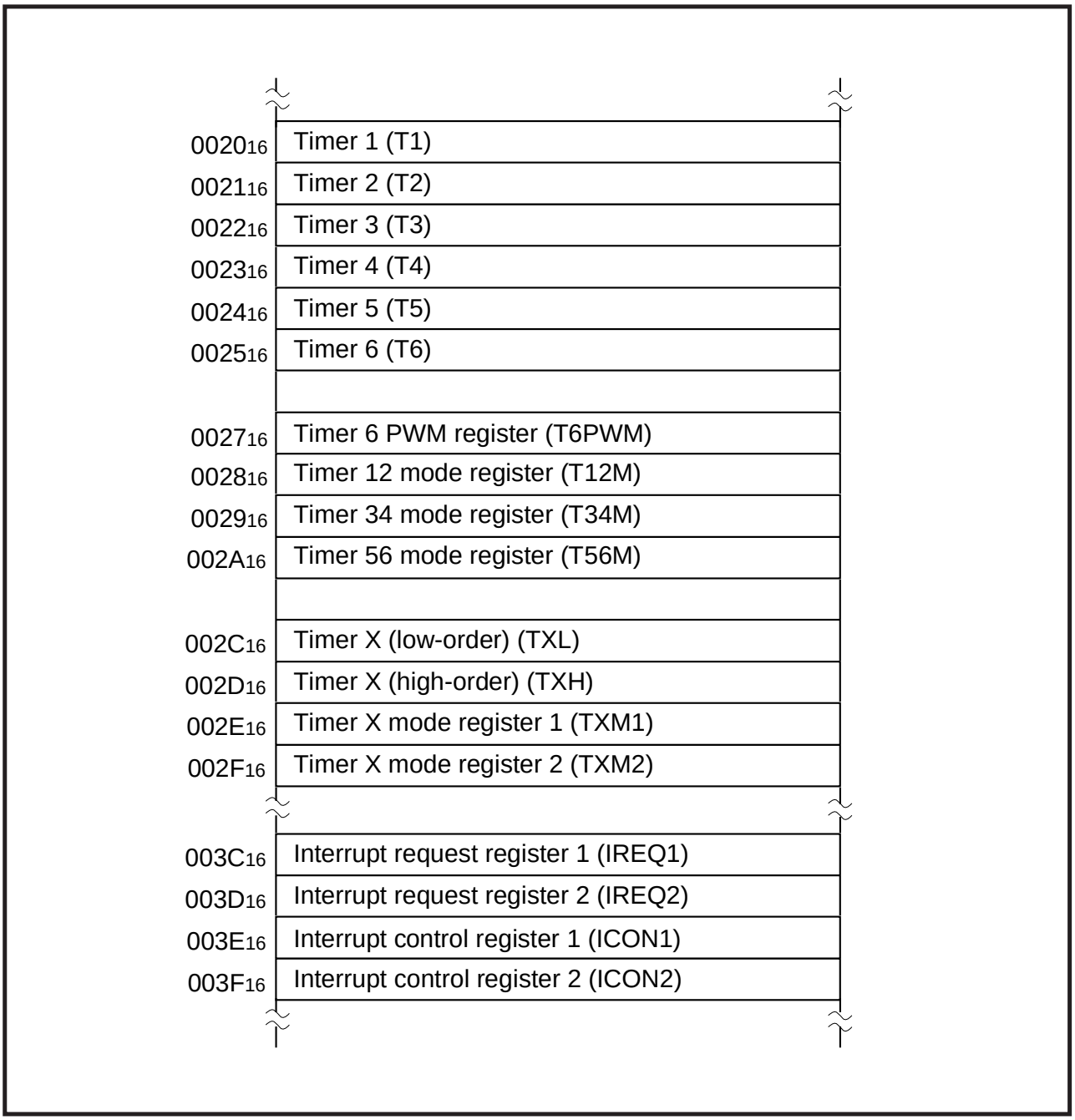


Fig. 2.2.1 Memory map of registers relevant to timers

2.2.2 Relevant registers

(1) 8-bit timer

Timer i

b7 b6 b5 b4 b3 b2 b1 b0

Timer i (i = 1, 3, 4, 5, 6)
(Ti: addresses 20_{16}, 22_{16}, 23_{16}, 24_{16}, 25_{16})

b	Functions	At reset	R	W
0	• Set timer i count value. • The value set in this register is written to both the timer i and the timer i latch at one time. • When the timer i is read out, the count value of the timer i is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Fig. 2.2.2 Structure of Timer i (i=1, 3, 4, 5, 6)

Timer 2

b7 b6 b5 b4 b3 b2 b1 b0

Timer 2
(T2: address 21_{16})

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

Fig. 2.2.3 Structure of Timer 2

Timer 6 PWM register

b7 b6 b5 b4 b3 b2 b1 b0

Timer 6 PWM register
(T6PWM: address 27_{16})

b	Functions	At reset	R	W
0	• In timer 6 PWM_1 mode "L" level width of PWM rectangular waveform is set. • Duty of PWM rectangular waveform: n/(n + m) Period: (n + m) × ts n = timer 6 set value m = timer 6 PWM register set value ts = timer 6 count source period At n = 0, all PWM output "L". At m = 0, all PWM output "H". (However, n = 0 has priority.) • Selection of timer 6 PWM_1 mode Set "1" to the timer 6 operation mode selection bit.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 2.2.4 Structure of Timer 6 PWM register

Timer 12 mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer 12 mode register
(T12M: address 28_{16})

b	Name	Functions	At reset	R	W
0	Timer 1 count stop bit	0: Count operation 1: Count stop	0	○	○
1	Timer 2 count stop bit	0: Count operation 1: Count stop	0	○	○
2 3	Timer 1 count source selection bits	b3 b2 0 0: $f(X_{IN})/8$ or $f(X_{CIN})/16$ 0 1: $f(X_{CIN})$ 1 0: $f(X_{IN})/16$ or $f(X_{CIN})/32$ 1 1: $f(X_{IN})/64$ or $f(X_{CIN})/128$	0	○	○
4	Timer 2 count source selection bits	b5 b4 0 0: Timer 1 underflow 0 1: $f(X_{CIN})$ 1 0: External count input $CNTR_0$ 1 1: Not available	0	○	○
5			0	○	○
6	Timer 1 output selection bit ($P4_5$)	0: I/O port 1: Timer 1 output	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×

Fig. 2.2.5 Structure of Timer 12 mode register

Timer 34 mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer 34 mode register
(T34M: address 29_{16})

b	Name	Functions	At reset	R	W
0	Timer 3 count stop bit	0: Count operation 1: Count stop	0	○	○
1	Timer 4 count stop bit	0: Count operation 1: Count stop	0	○	○
2 3	Timer 3 count source selection bits	b3 b2 0 0: $f(X_{IN})/8$ or $f(X_{CIN})/16$ 0 1: Timer 2 underflow 1 0: $f(X_{IN})/16$ or $f(X_{CIN})/32$ 1 1: $f(X_{IN})/64$ or $f(X_{CIN})/128$	0	○	○
4	Timer 4 count source selection bits	b5 b4 0 0: $f(X_{IN})/8$ or $f(X_{CIN})/16$ 0 1: Timer 3 underflow 1 0: External count input $CNTR_1$ **(Note)** 1 1: Not available	0	○	○
5			0	○	○
6	Timer 3 output selection bit ($P4_6$)	0: I/O port 1: Timer 3 output	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×

Note: In the mask option type P, $CNTR_1$ function cannot be used.

Fig. 2.2.6 Structure of Timer 34 mode register

Timer 56 mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer 56 mode register
(T56M: address 2A16)

b	Name	Functions	At reset	R	W
0	Timer 5 count stop bit	0: Count operation 1: Count stop	0	○	○
1	Timer 6 count stop bit	0: Count operation 1: Count stop	0	○	○
2	Timer 5 count source selection bit	0: f(XIN)/8 or f(XCIN)/16 1: Timer 4 underflow	0	○	○
3	Timer 6 operation mode selection bit	0: Timer mode 1: PWM mode	0	○	○
4	Timer 6 count source selection bits	b5 b4 0 0: f(XIN)/8 or f(XCIN)/16 0 1: Timer 5 underflow 1 0: Timer 4 underflow 1 1: Not available	0	○	○
5			0	○	○
6	Timer 6 (PWM) output selection bit (P44)	0: I/O port 1: Timer 6 output	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are "0".		0	○	×

Fig. 2.2.7 Structure of Timer 56 mode register

(2) 16-bit timer

Timer X (low-order, high-order)

b7 b6 b5 b4 b3 b2 b1 b0

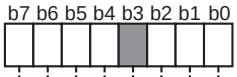
Timer X (low-order, high-order)
(TXL, TXH: addresses 2C16, 2D16)

b	Functions	At reset	R	W
0	• Set timer X count value. • When the timer X write control bit of the timer X mode register 1 is "0", the value is written to timer X and the latch at one time. When the timer X write control bit of the timer X mode register 1 is "1", the value is written only to the latch. • The timer X count value is read out by reading this register.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Notes 1: When reading and writing, perform them to both the high-order and low-order bytes.
2: Read both registers in order of TXH and TXL following.
3: Write both registers in order of TXL and TXH following.
4: Do not read both registers during a write, and do not write to both registers during a read.

Fig. 2.2.8 Structure of Timer X (low-order, high-order)

Timer X mode register 1

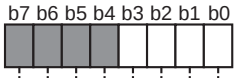


Timer X mode register 1
(TXM1: address 2E16)

b	Name	Functions	At reset	R	W
0	Timer X write control bit	0 : Write value in latch and counter 1 : Write value in latch only	0	○	○
1	Timer X count source selection bits	b2 b1 0 0: f(XIN)/2 or f(XCIN)/4 0 1: f(XIN)/8 or f(XCIN)/16 1 0: f(XIN)/64 or f(XCIN)/128 1 1: Not available	0	○	○
2			0	○	○
3	Nothing is arranged for this bit. This is write disabled bit. When this bit is read out, the contents are "0".		0	○	×
4	Timer X operating mode bits	b5 b4 0 0 : Timer mode 0 1 : Pulse output mode 1 0 : Event counter mode 1 1 : Pulse width measurement mode	0	○	○
5			0	○	○
6	CNTR2 active edge switch bit	0 : •Count at rising edge in event counter mode •Start from "H" output in pulse output mode •Measure "H" pulse width in pulse width measurement mode 1 : •Count at falling edge in event counter mode •Start from "L" output in pulse output mode •Measure "L" pulse width in pulse width measurement mode	0	○	○
7	Timer X stop control bit	0 : Count operating 1 : Count stop	0	○	○

Fig. 2.2.9 Structure of Timer X mode register 1

Timer X mode register 2



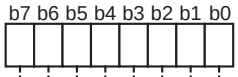
Timer X mode register 2
(TXM2: address $2F_{16}$)

b	Name	Functions	At reset	R	W
0	Real time port control bit ($P8_5$)	0: Real time port function is invalid 1: Real time port function is valid	0	○	○
1	Real time port control bit ($P8_6$)	0: Real time port function is invalid 1: Real time port function is valid	0	○	○
2	$P8_5$ data for real time port	0: "L" output 1: "H" output	0	○	○
3	$P8_6$ data for real time port	0: "L" output 1: "H" output	0	○	○
4	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are "0".		0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Fig. 2.2.10 Structure of Timer X mode register 2

(3) 8-bit timer, 16-bit timer

Interrupt request register 1



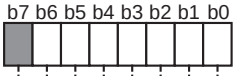
Interrupt request register 1
(IREQ1 : address $3C_{16}$)

b	Name	Functions	At reset	R	W
0	INT_0 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	INT_1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	INT_2 interrupt request bit Remote controller /counter overflow interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O1 interrupt request bit Serial I/O automatic transfer interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	Timer 2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Timer 3 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻

✻: "0" can be set by software, but "1" cannot be set.

Fig. 2.2.11 Structure of Interrupt request register 1

Interrupt request register 2



Interrupt request register 2
(IREQ2 : address 3D16)

b	Name	Functions	At reset	R	W
0	Timer 4 interrupt request bit **(Note)**	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Timer 5 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	Timer 6 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O2 receive interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	INT3/Serial I/O2 transmit interrupt request bit **(Note)**	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	INT4 interrupt request bit A-D converter interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	FLD blanking interrupt request bit FLD digit interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×

✻: “0” can be set by software, but “1” cannot be set.

Note: In the mask option type P, if timer 4 interrupt whose count source is CNTR1 input and INT3 interrupt are selected, these bits do not become “1”.

Fig. 2.2.12 Structure of Interrupt request register 2

Interrupt control register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 1
(ICON1 : address $3E_{16}$)

b	Name	Functions	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	INT_2 interrupt enable bit Remote controller /counter overflow interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O1 interrupt enable bit Serial I/O automatic transfer interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Timer 2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	Timer 3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Fig. 2.2.13 Structure of Interrupt control register 1

Interrupt control register 2

b7 b6 b5 b4 b3 b2 b1 b0
0

Interrupt control register 2
(ICON2 : address $3F_{16}$)

b	Name	Functions	At reset	R	W
0	Timer 4 interrupt enable bit **(Note)**	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer 5 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer 6 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O2 receive interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
4	INT_3/Serial I/O2 transmit interrupt enable bit **(Note)**	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
5	INT_4 interrupt enable bit A-D converter interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
6	FLD blanking interrupt enable bit FLD digit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
7	Fix "0" to this bit.		0	○	○

Note: In the mask option type P, timer 4 interrupt whose count source is $CNTR_1$ input and INT_3 interrupt are not available.

Fig. 2.2.14 Structure of Interrupt control register 2

2.2.3 Timer application examples

(1) Basic functions and uses

[Function 1] Control of event interval (Timer 1 to Timer 6, Timer X: timer mode)

When a certain time, by setting a count value to each timer, has passed, the timer interrupt request occurs.

<Use>

•Generating of an output signal timing

•Generating of a wait time

[Function 2] Control of cyclic operation (Timer 1 to Timer 6, Timer X: timer mode)

The value of the timer latch is automatically written to the corresponding timer each time the timer underflows, and each timer interrupt request occurs in cycles.

<Use>

•Generating of cyclic interrupts

•Clock function (measurement of 1 s); see "(2) Timer application example 1"

•Control of a main routine cycle

[Function 3] Output of rectangular waveform (Timer 1, Timer 3, Timer 6, Timer X: pulse output mode)

The output level of the T_{1OUT} pin, T_{3OUT} pin, PWM_1 pin or $CNTR_2$ pin is inverted each time the timer underflows.

<Use>

•Piezoelectric buzzer output; see "(3) Timer application example 2"

•Generating of the remote control carrier waveforms

[Function 4] Count of external pulses (Timer 2, Timer 4, Timer X: event counter mode)

External pulses input to the $CNTR_0$ pin, $CNTR_1$ pin, $CNTR_2$ pin are counted as the timer count source (in the event counter mode).

<Use>

•Frequency measurement; see "(4) Timer application example 3"

•Division of external pulses

•Generating of interrupts due to a cycle using external pulses as the count source; count of a reel pulse

[Function 5] Output of PWM signal (Timer 6)

"H" interval and "L" interval are specified, respectively, and the output of pulses from $P4_4/PWM_1$ pin is repeated.

<Use>

•Control of electric volume

[Function 6] Measurement of external pulse width (Timer X: pulse width measurement mode)

The "H" or "L" level width of external pulses input to $CNTR_2$ pin is measured.

<Use>

•Measurement of external pulse frequency (measurement of pulse width of FG pulse* for a motor); see "(5) Timer application example 4"

•Measurement of external pulse duty (when the frequency is fixed)

FG pulse*: Pulse used for detecting the motor speed to control the motor speed.

[Function 7] Control of real time port (Timer X: real time port function)

The data for real time is output from the $P8_5$ pin or $P8_6$ pin each time the timer underflows.

<Use>

•Stepping motor control; see "(6) Timer application example 5"

(2) Timer application example 1: Clock function (measurement of 1 s)

Outline: The input clock is divided by the timer so that the clock can count up at 1 s intervals.

Specifications: •The clock f(X_{IN}) = 4.19 MHz (2^{22} Hz) is divided by the timer.
•The timer 3 interrupt request bit is checked in main routine, and if the interrupt request is issued, the clock is counted up.
• The timer 1 interrupt occurs every 244 µs to execute processing of other interrupts.

Figure 2.2.15 shows the timers connection and setting of division ratios; Figure 2.2.16 shows the relevant registers setting; Figure 2.2.17 shows the control procedure.

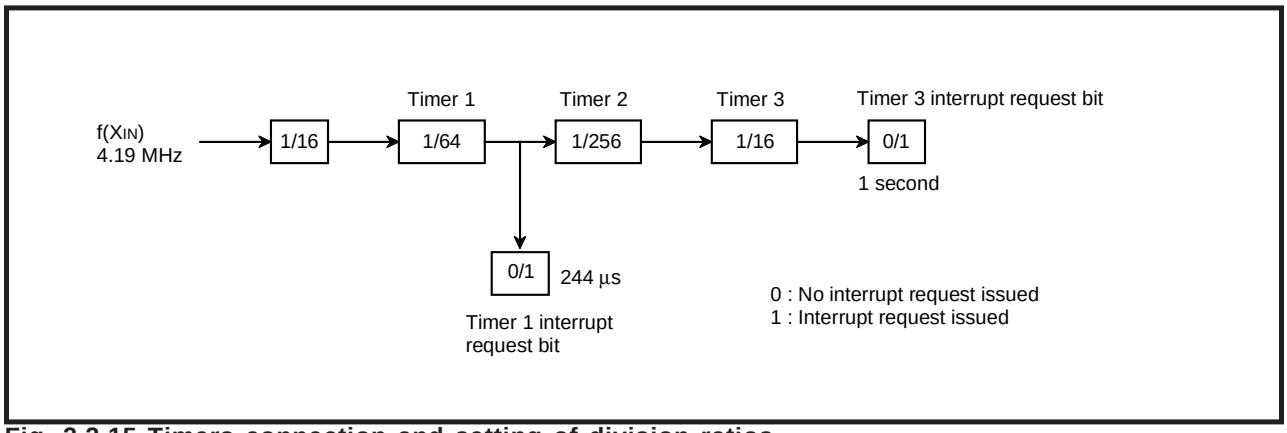


Fig. 2.2.15 Timers connection and setting of division ratios

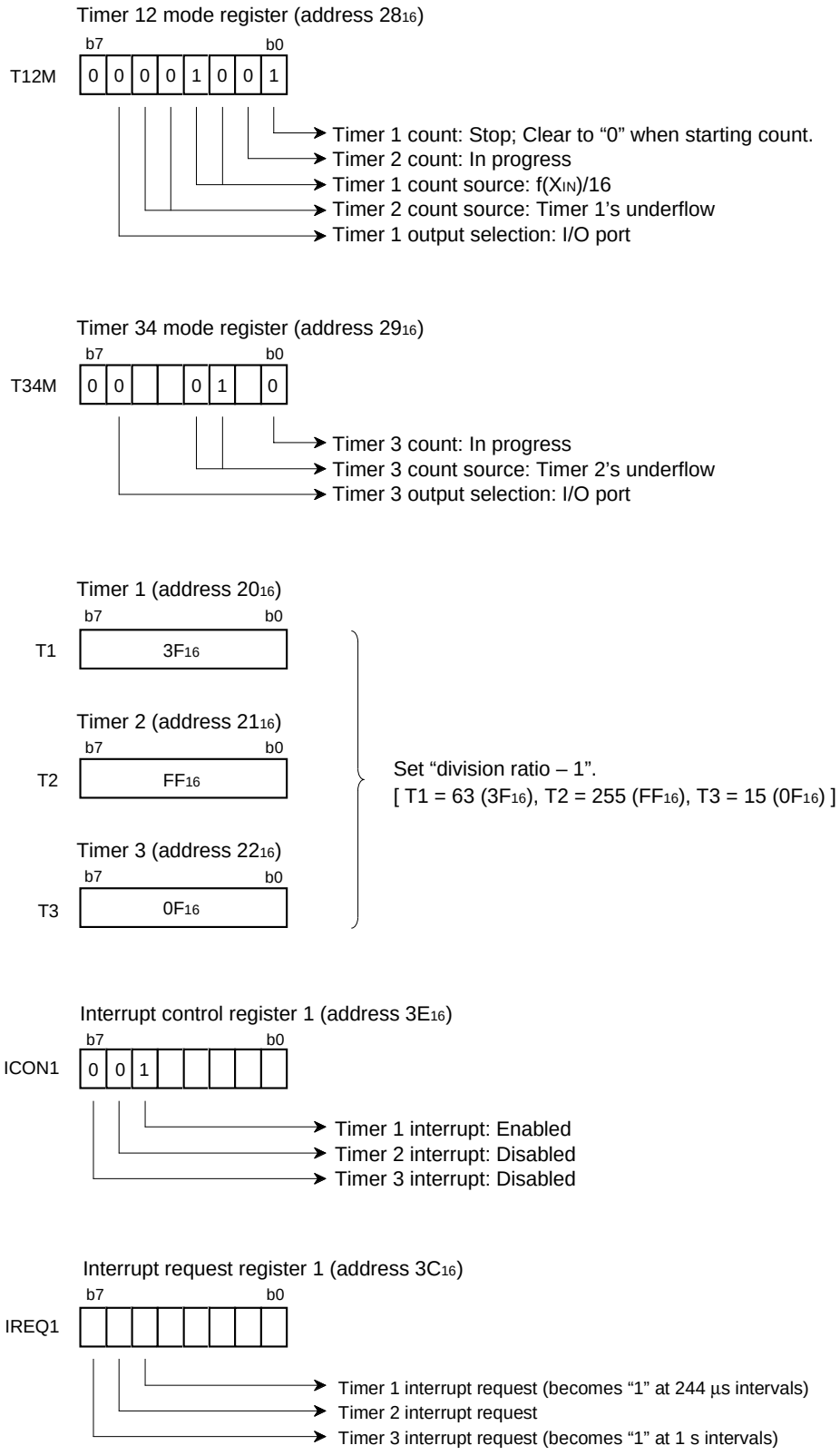
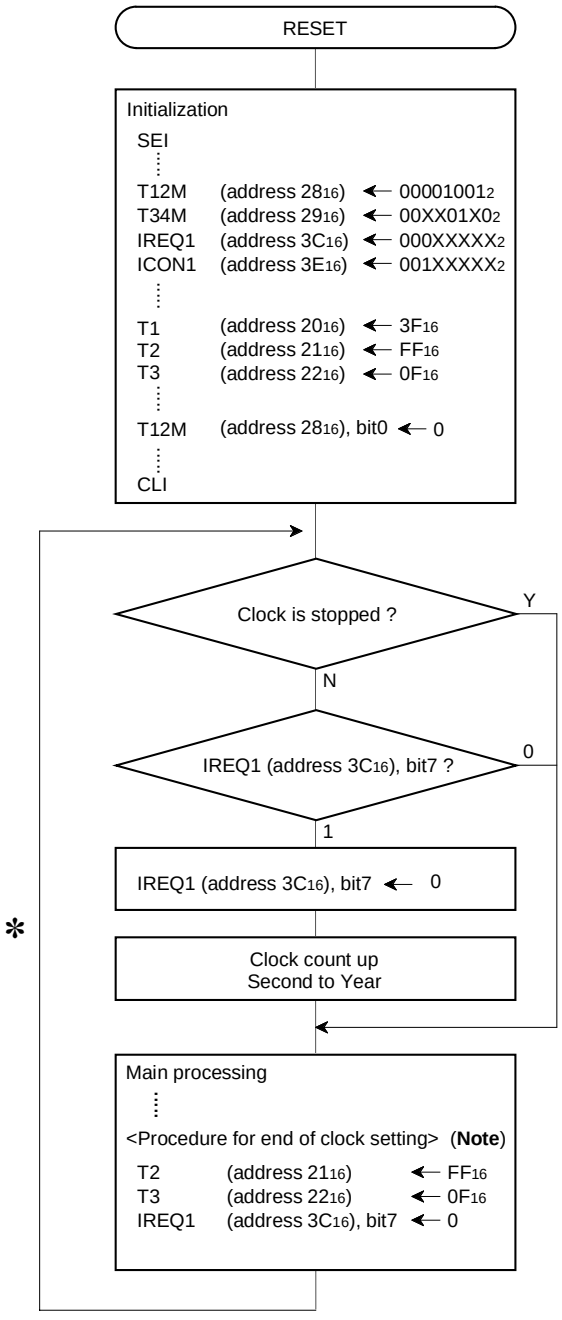


Fig. 2.2.16 Relevant registers setting



● X: This bit is not used here. Set it to “0” or “1” arbitrarily.

•All interrupts disabled

•Connection of Timers 1 to 3

•Setting of Interrupt request bits of Timers 1 to 3 to “0”
•Timer 1 interrupt enabled, Timers 2 and 3 interrupts disabled

•Setting “Division ratio – 1” to Timers 1 to 3

•Timer count start

•Interrupts enabled

•Judgment whether time is not set or time is being set

•Confirmation that 1 sec. has passed
(Check of Timer 3 interrupt request bit)

•Interrupt request bit cleared
(Clear it by software when not using the interrupt.)

•Clock count up

•Adjust the main processing so that all processing in the loop ✻ will be processed within 1 second interval.

•Set Timers again when starting clock from 0 second after end of clcok setting.
The procedure is Timer 2 setting followed by Timer 3 setting.
•Do not set Timer 1 again because Timer 1 is used to generate the interrupt at 244 μs intervals.

Note : Perform procedure for end of clock setting only when end of clock setting.

Fig. 2.2.17 Control procedure

(3) Timer application example 2: Piezoelectric buzzer output

Outline: The rectangular waveform output function of the timer is applied for a piezoelectric buzzer output.

Specifications: •The rectangular waveform, dividing the clock f(X_{IN}) = 4.19 MHz (2^{22} Hz) into about 2 kHz (2048 Hz), is output from the $P4_6/T_{3OUT}$ pin.

•The level of the $P4_6/T_{3OUT}$ pin is fixed to "H" while a piezoelectric buzzer output stops.

Figure 2.2.18 shows a peripheral circuit example, and Figure 2.2.19 shows the timers connection and setting of division ratios. Figures 2.2.20 shows the relevant registers setting, and Figure 2.2.21 shows the control procedure.

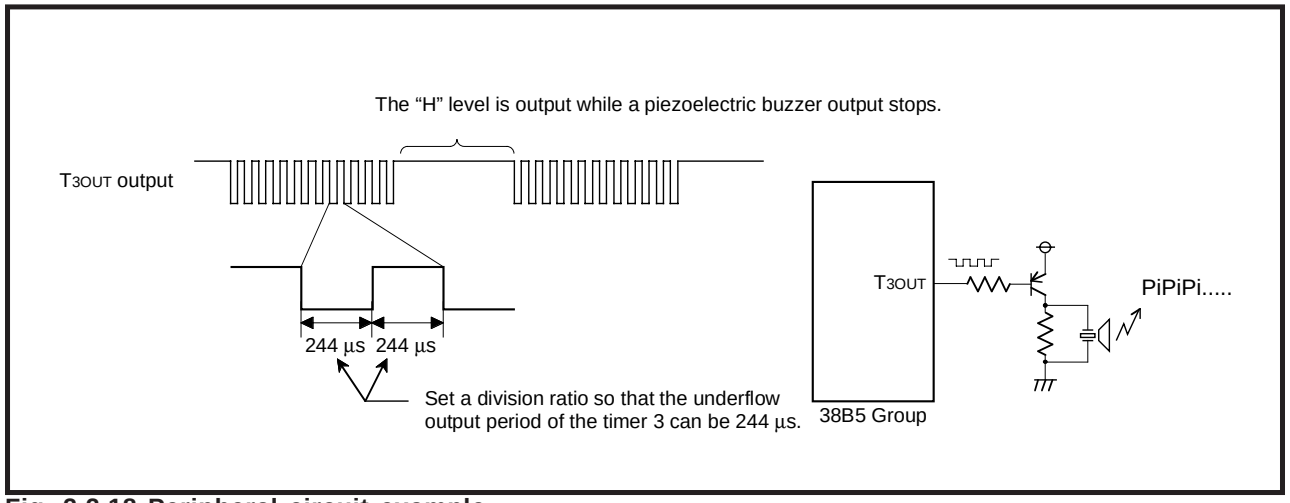


Fig. 2.2.18 Peripheral circuit example

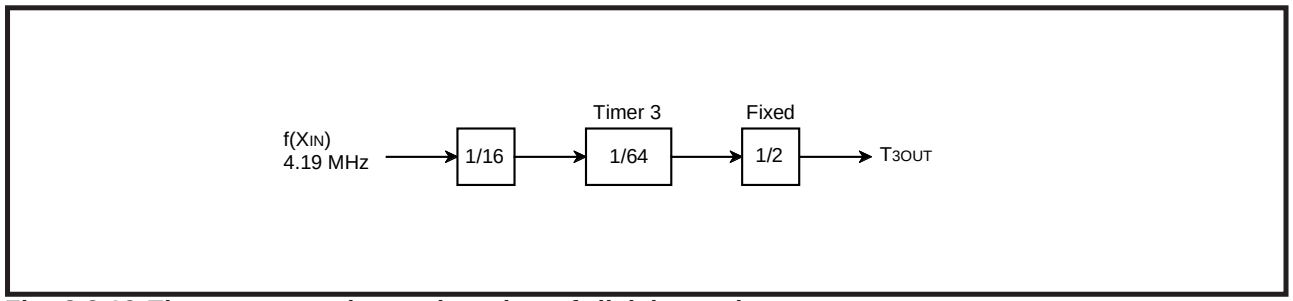


Fig. 2.2.19 Timers connection and setting of division ratios

2.2 Timer

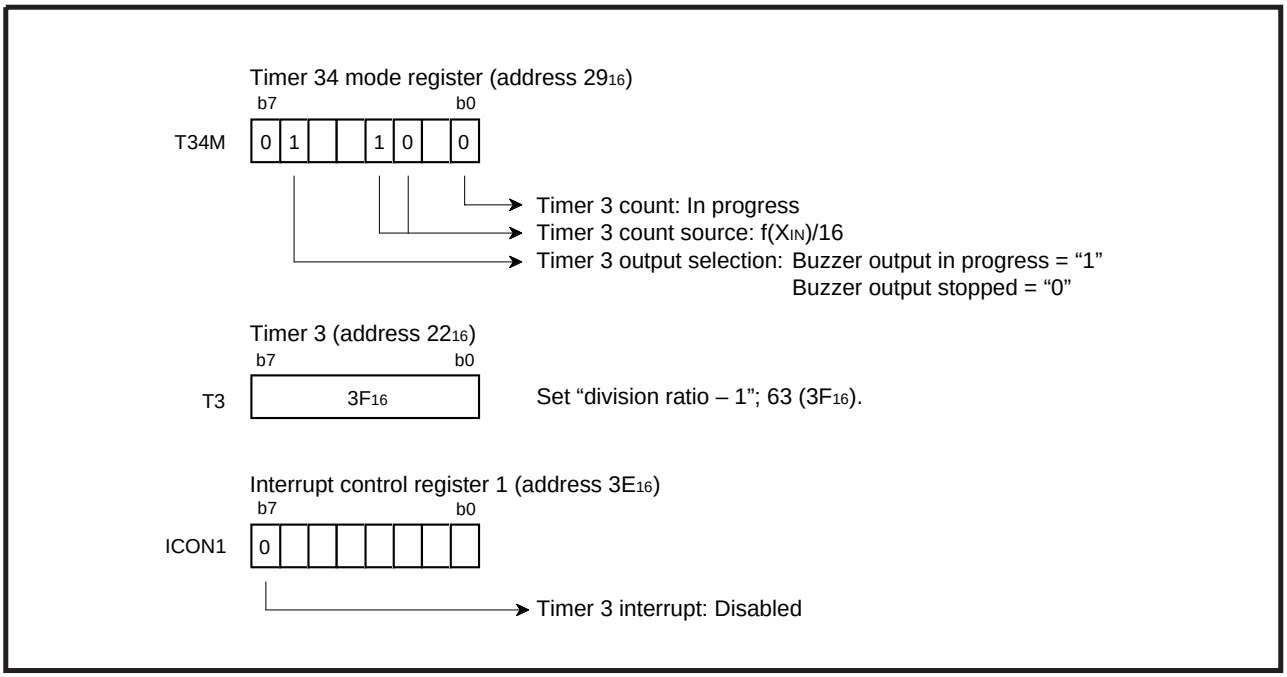


Fig. 2.2.20 Relevant registers setting

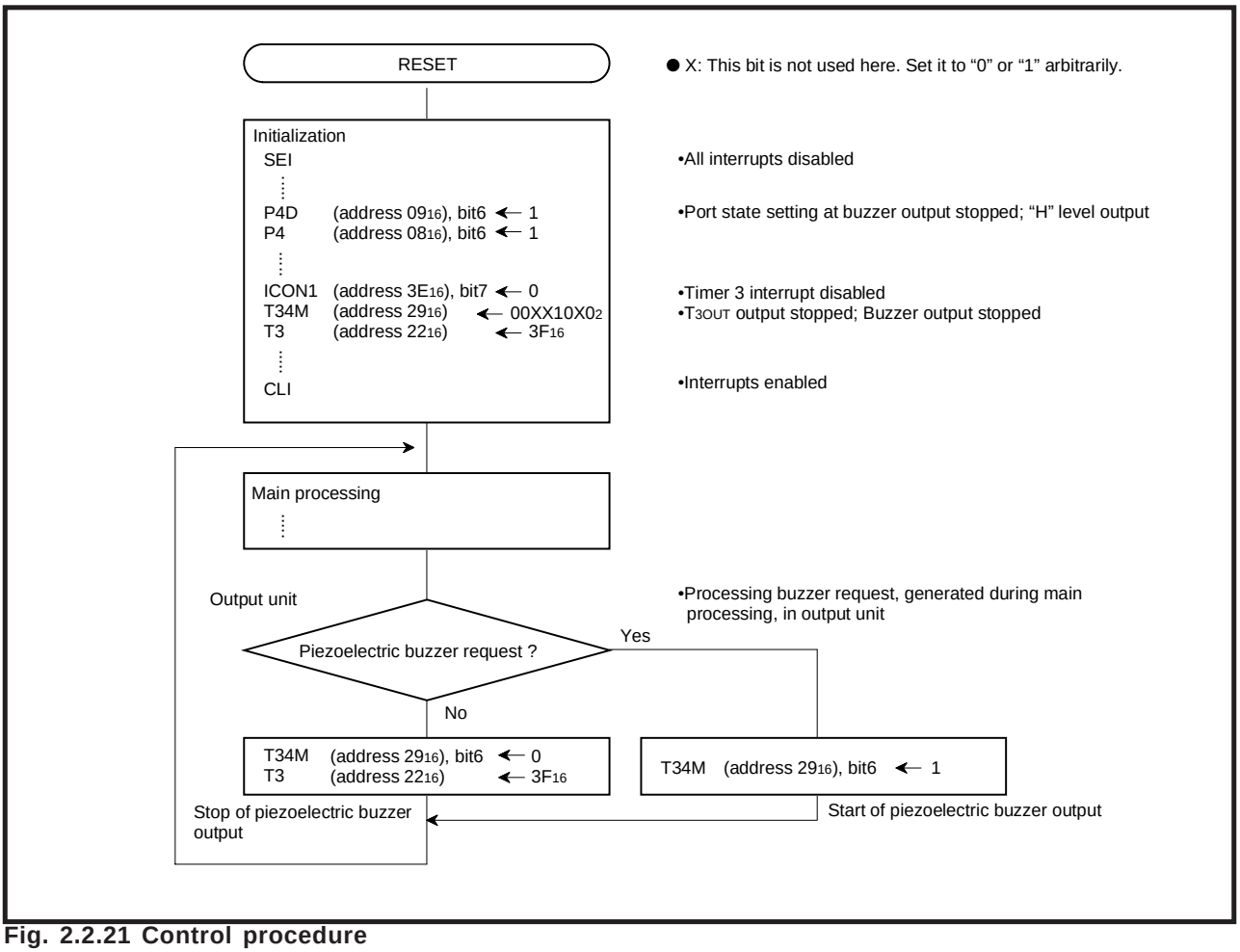


Fig. 2.2.21 Control procedure

(4) Timer application example 3: Frequency measurement

Outline: The following two values are compared to judge whether the frequency is within a valid range.

- A value by counting pulses input to $P6_0/CNTR_1$ pin with the timer.
- A reference value

Specifications: •The pulse is input to the $P6_0/CNTR_1$ pin and counted by the timer 4. (**Note 1**)

•A count value of timer 4 is read out at about 2 ms intervals, the timer 1 interrupt interval. When the count value is 28 to 40, it is judged that the input pulse is valid.

•Because the timer is a down-counter, the count value is compared with 227 to 215 (**Note 2**).

Notes 1: In the mask option type P, use the $CNTR_0$ pin and timer 2.

2: 227 to 215 = {255 (initial value of counter) – 28} to {255 – 40}; 28 to 40 means the number of valid value.

Figure 2.2.22 shows the judgment method of valid/invalid of input pulses; Figure 2.2.23 shows the relevant registers setting; Figure 2.2.24 shows the control procedure.

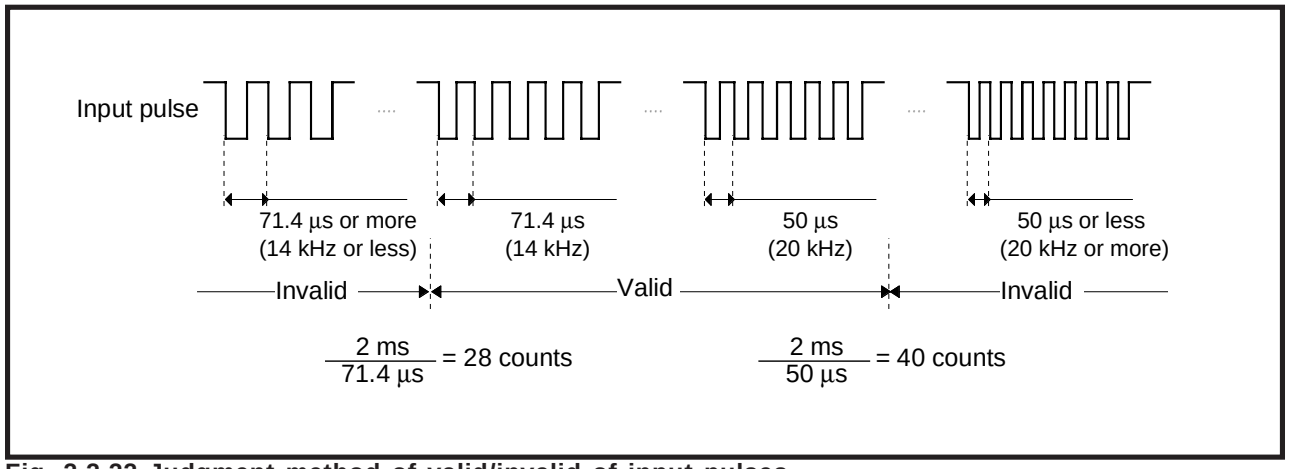


Fig. 2.2.22 Judgment method of valid/invalid of input pulses

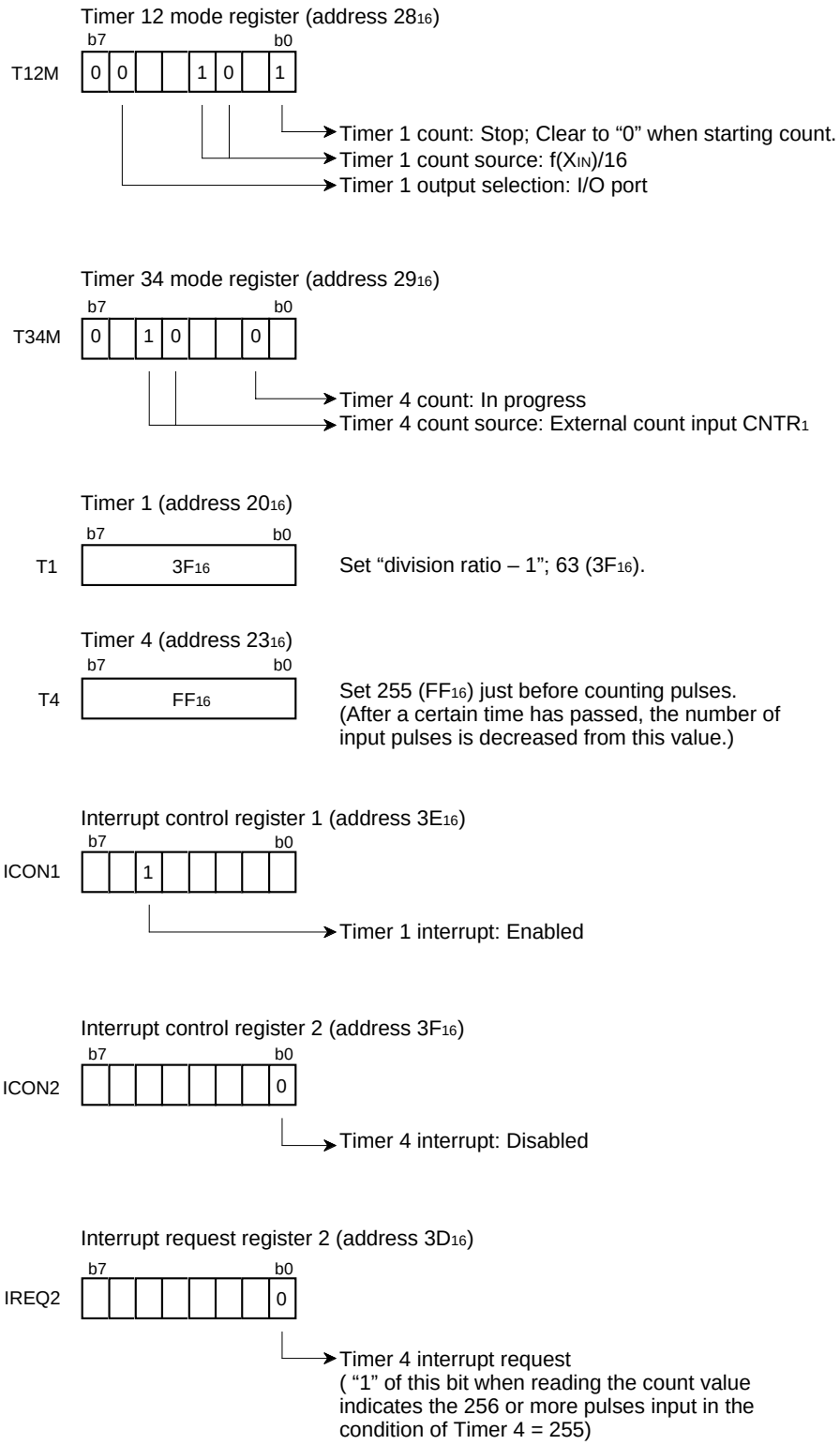


Fig. 2.2.23 Relevant registers setting

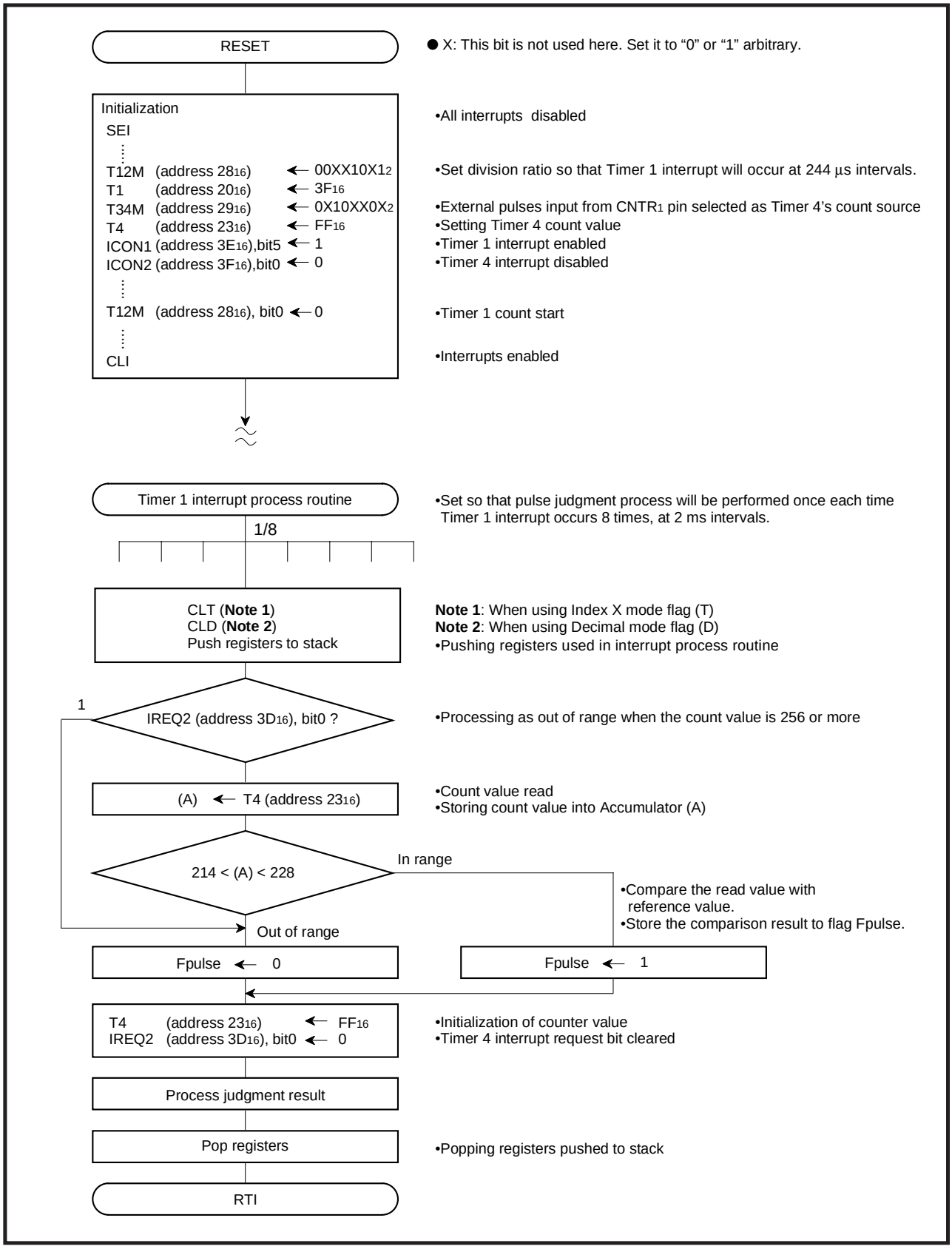


Fig. 2.2.24 Control procedure

(5) Timer application example 4: Measurement of FG pulse width for motor

Outline: The timer X counts the "H" level width of the pulses input to the $P6_1/CNTR_0/CNTR_2$ pin. An underflow is detected by the timer X interrupt and an end of the input pulse "H" level is detected by the timer 2 interrupt of which count source is the input to $P6_1/CNTR_0/CNTR_2$ pin.

Specifications: •The timer X counts the "H" level width of the FG pulse input to the $P6_1/CNTR_0/CNTR_2$ pin.

<Example>

When $f(X_{IN})$ = 4.19 MHz, the count source is 15.2 µs, which is obtained by dividing the clock frequency by 64. Measurement can be made up to 1 s in the range of $FFFF_{16}$ to 0000_{16}.

Figure 2.2.25 shows the timers connection and setting of division ratio; Figure 2.2.26 shows the relevant registers setting; Figure 2.2.27 shows the control procedure.

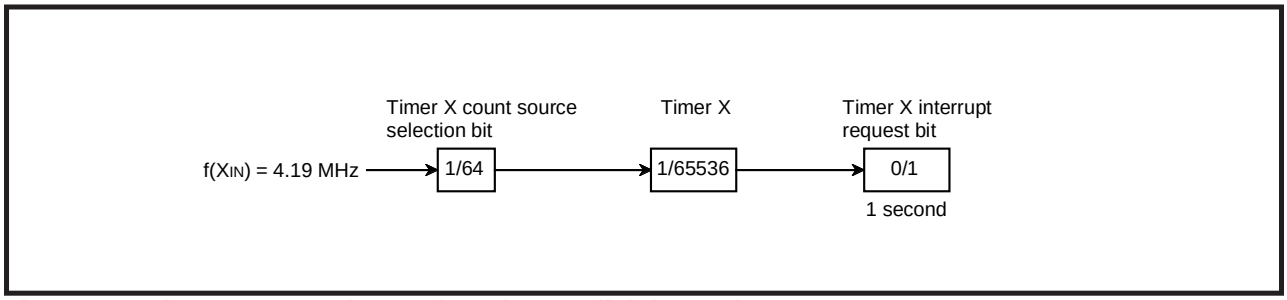


Fig. 2.2.25 Timers connection and setting of division ratios

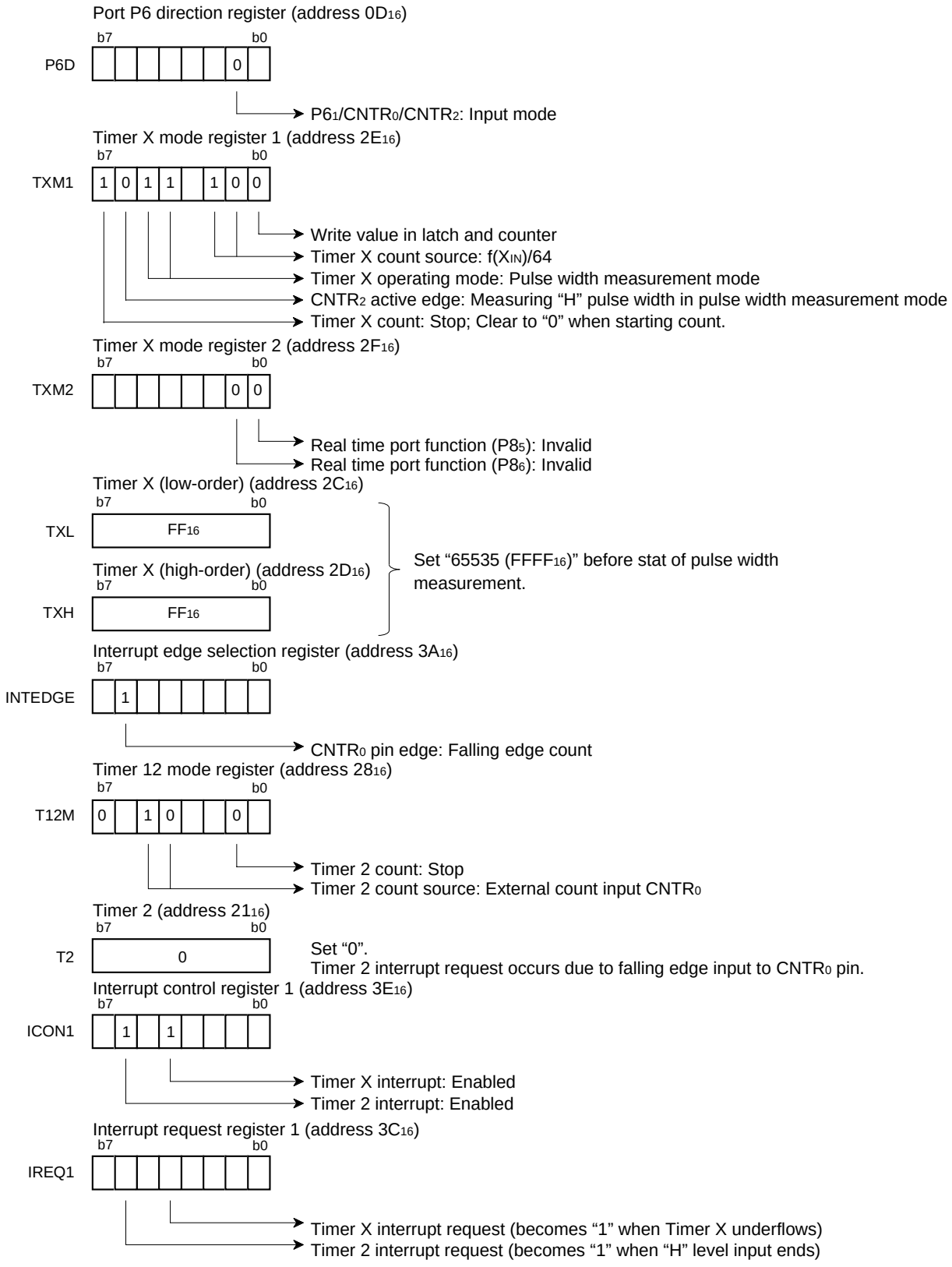
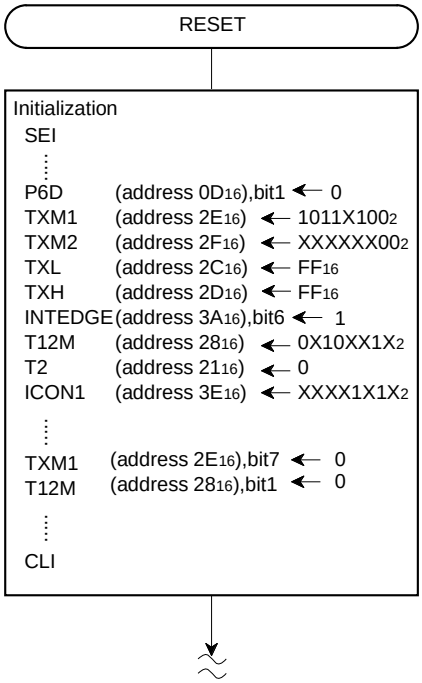


Fig. 2.2.26 Relevant registers setting



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

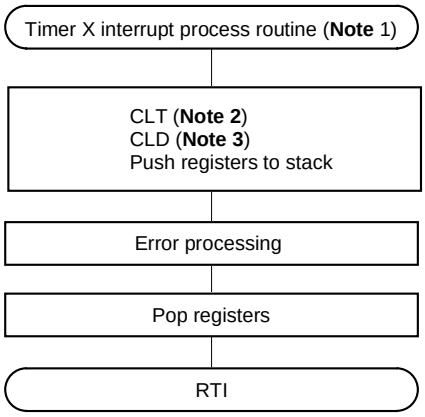
•All interrupts disabled

•Setting $P6_1$/$CNTR_0$/$CNTR_2$ pin to input mode
•Timer X: Pulse width measurement mode
(Measuring "H" pulse width of input pulses from $CNTR_2$ pin)
•Setting Timer X count value

•$CNTR_0$ pin edge: Falling edge count
•External pulses input from $CNTR_0$ pin selected as Timer 2's count source
•Setting "0" to Timer 2
•Timers X and 2 interrupts: Enabled

•Timers X and 2 count start

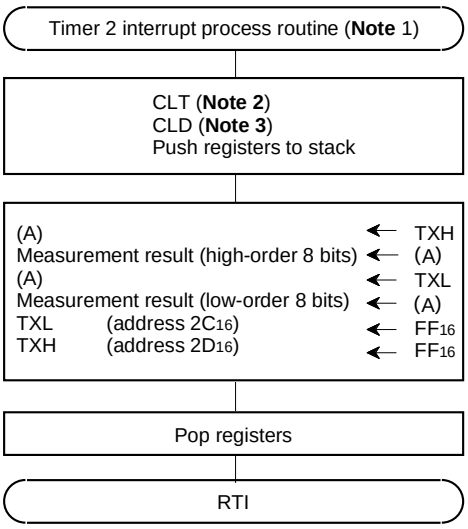
•Interrupts enabled



Notes 1: Timer X interrupt also occurs owing to factors other than measurement level.($CNTR_2$ input = "L" in this application) Process it by software as error proccesing is performed for measurement level as necessary . $CNTR_2$ input level can be checked by reading the contents of sharing port $P6_1$ register.
2: When using Index X mode flag (T)
3: When using Decimal mode flag (D)
•Pushing registers used in interrupt process routine

•Popping registers pushed to stack

Fig. 2.2.27 Control procedure



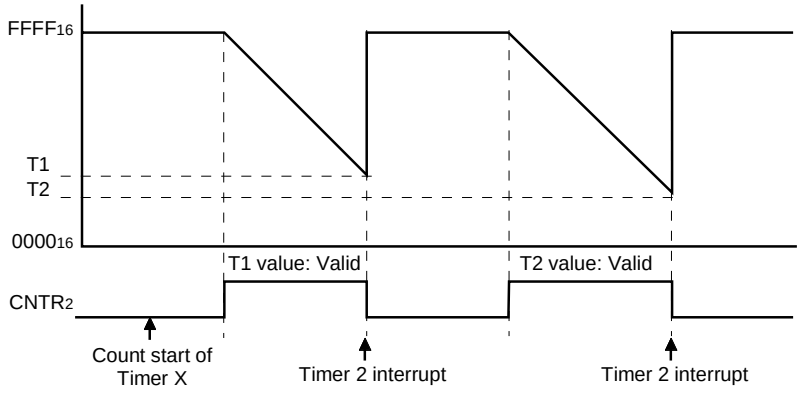
Notes **2**: When using Index X mode flag (T)
3: When using Decimal mode flag (D)
•Pushing registers used in interrupt process routine

•Count value read and storing it to RAM

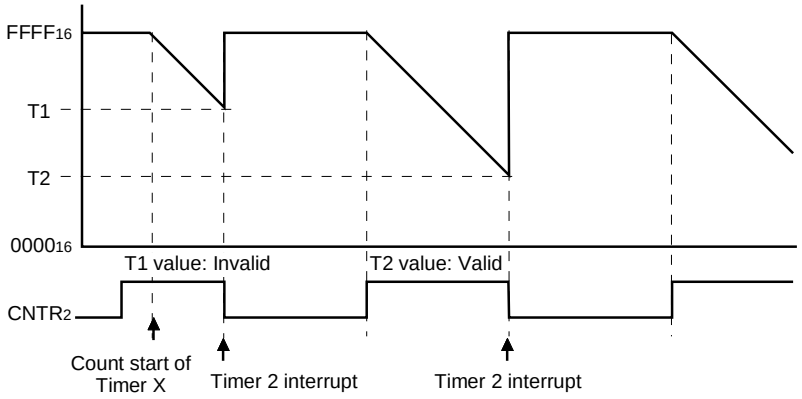
•Popping registers pushed to stack

Note 1: The first value becomes invalid depending on start timing of Time X count shown by the following figure.
Process it by software as necessary.

[Example 1] • Start Timer X count when $CNTR_2$ input level is "L".
($CNTR_2$ input level can be checked by reading the contents of sharing port $P6_1$ register.



[Example 2] • Start Timer X count when $CNTR_2$ input level is "H".
Invalidate the first Timer 2 interrupt after start of Timer X count.

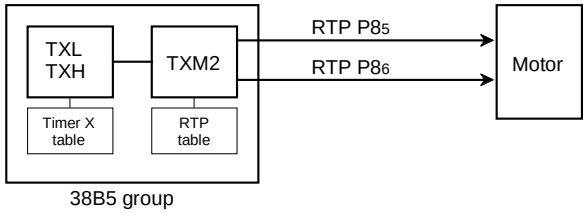


(6) Timer application example 5: Control of stepping motor

Outline: The rotating of stepping motor is controlled by using real time output ports.

Specifications: •The motor is controlled by using 2 real time output ports.
•The count source is f(X_{IN}) = 4.19 MHz divided by 8.
•Values of Timer X and real time output are updated in the timer X interrupt routine

Figure 2.2.28 shows the timers connection and the table example of timer X/RTP setting values; Figure 29 shows the RTP output example; Figure 2.2.30 shows the relevant registers setting; Figure 2.2.31 shows the control procedure.



Timer X setting table example

RTP output time	Timer X value
T_1	$2FD0_{16}$
T_2	$2B71_{16}$
T_3	2081_{16}
T_4	1869_{16}
T_5	$13C9_{16}$
T_6	$13A9_{16}$
T_7	1221_{16}
T_8	$11C1_{16}$

RTP setting table example

RTP output pattern	RTP value	
	TXM2,b2	TXM2,b3
(1)	0	0
(2)	0	1
(3)	1	0
(4)	1	1

RTP: Real Time Port

Fig. 2.2.28 Timers connection and table example of timer X/RTP setting values

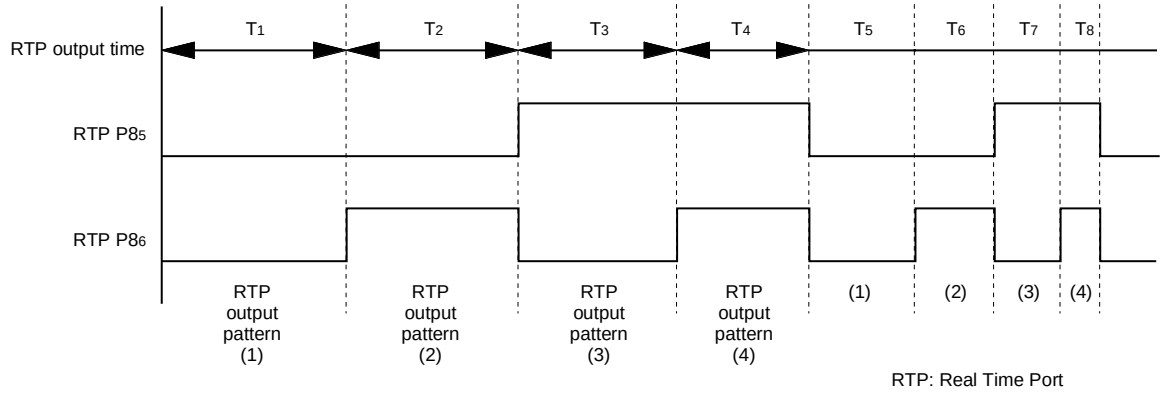


Fig. 2.2.29 RTP output example

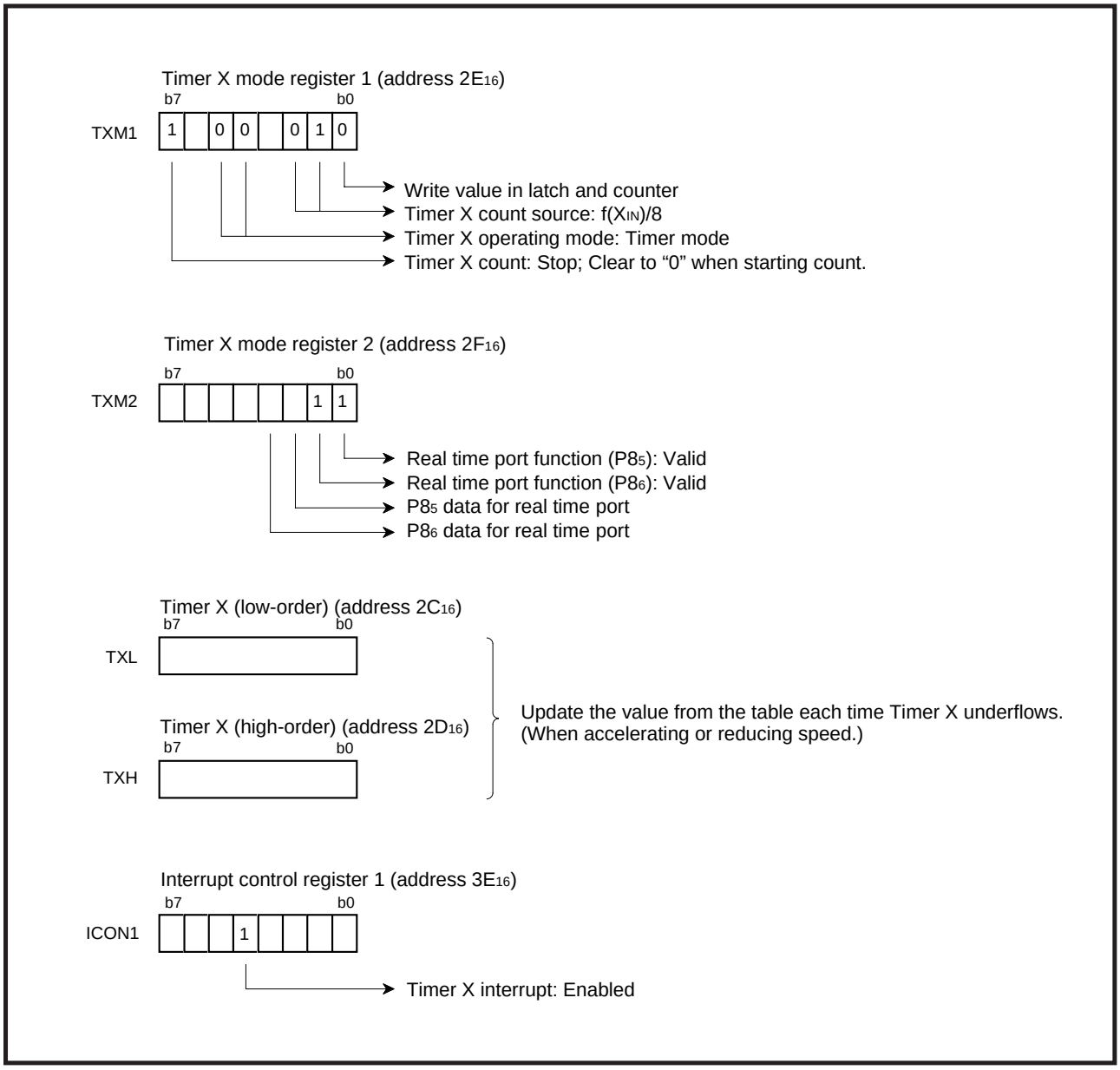


Fig. 2.2.30 Relevant registers setting

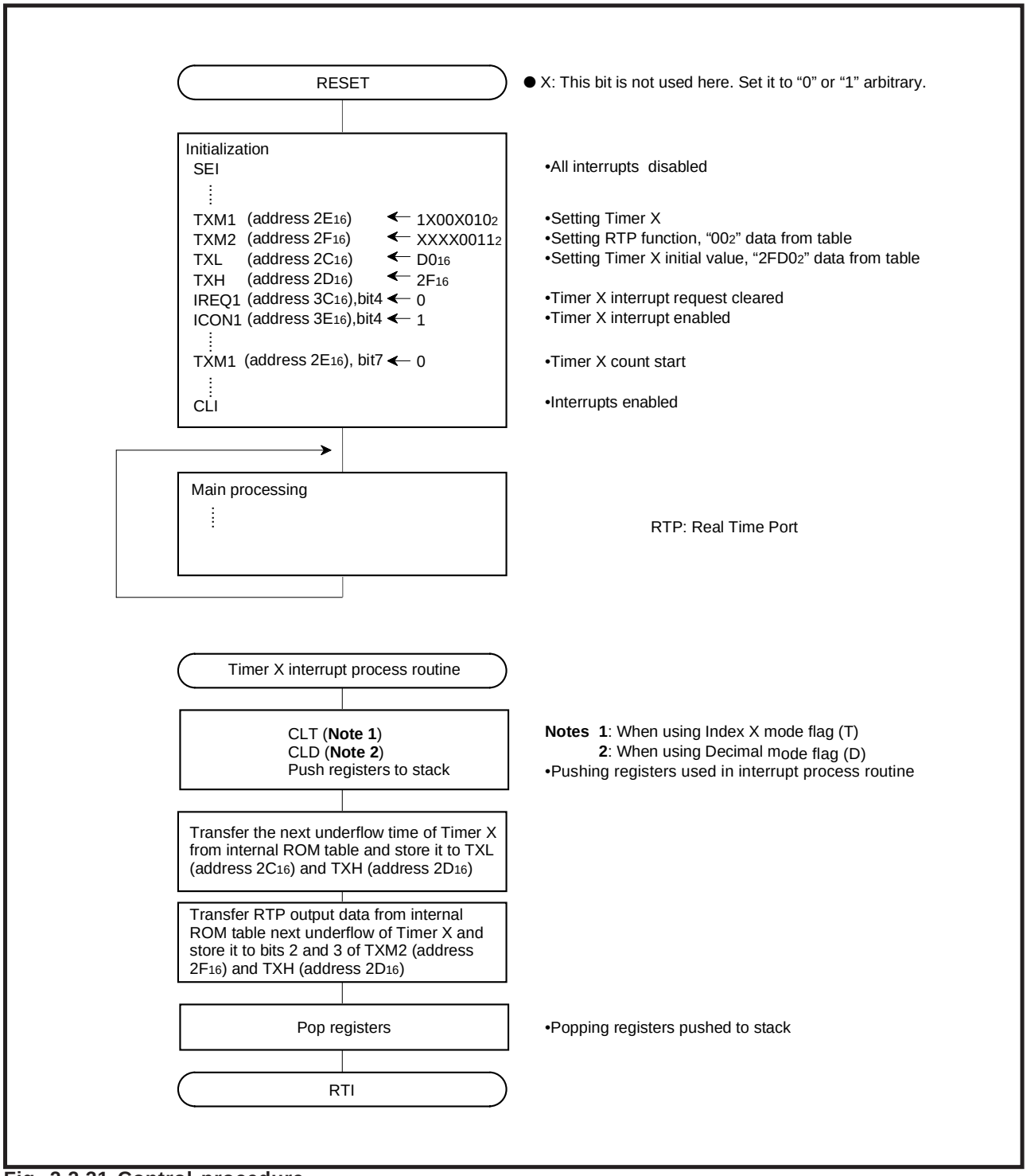


Fig. 2.2.31 Control procedure

2.3 Serial I/O

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

2.3.1 Memory map

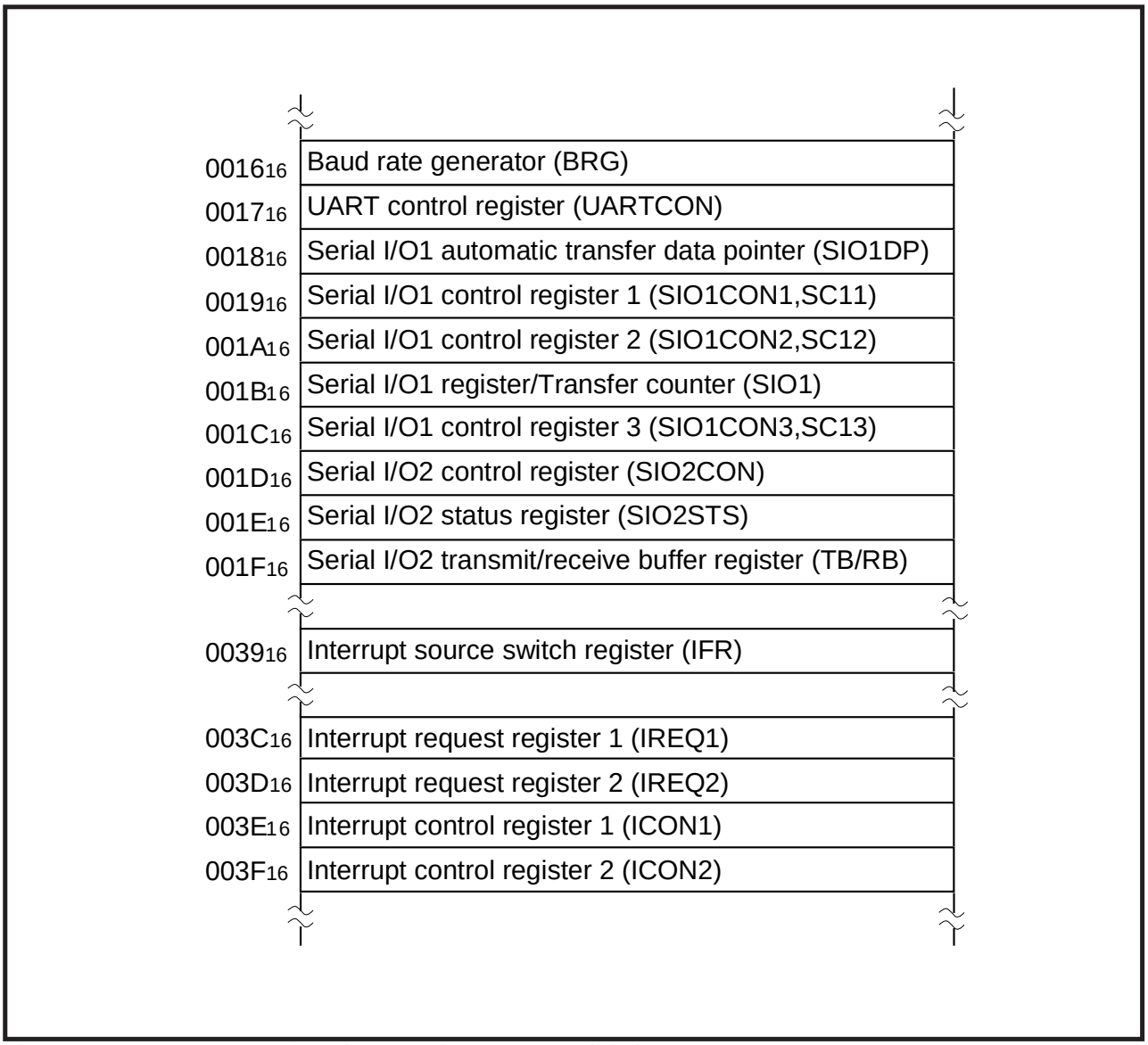
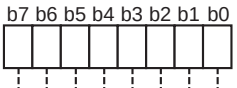


Fig. 2.3.1 Memory map of registers relevant to Serial I/O

2.3.2 Relevant registers

(1) Serial I/O1

Serial I/O1 automatic transfer data pointer

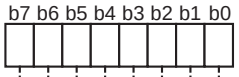


Serial I/O1 automatic transfer data pointer
(SIO1DP: address 18_{16})

b	Functions	At reset	R	W
0	• Indicates the low-order 8 bits of the address storing the start data on the serial I/O. automatic transfer RAM. • Data is written into the latch and read from the decrement counter.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 2.3.2 Structure of Serial I/O1 automatic transfer data pointer

Serial I/O1 control register 1



Serial I/O1 control register 1
(SIO1CON1•SC11: address 19_{16})

b	Name	Functions	At reset	R	W
0	Serial transfer selection bits	b1b0 0 0: Serial I/O disabled (Pins $P6_2$, $P6_4$, $P6_5$, $P5_0$–$P5_3$ pins are I/O ports.) 0 1: 8-bit serial I/O 1 0: Not available 1 1: Automatic transfer serial I/O (8 bits)	0	○	○
1			0	○	○
2	Serial I/O1 synchronous clock selection bits ($P6_5$/S_{STB1} pin control bits)	b3b2 0 0: Internal synchronous clock ($P6_5$ pin is I/O port.) 0 1: External synchronous clock ($P6_5$ pin is I/O port.) 1 0: Internal synchronous clock ($P6_5$ pin is S_{STB1} output.) 1 1: Internal synchronous clock ($P6_5$ pin is S_{STB1} output.)	0	○	○
3			0	○	○
4	Serial I/O initialization bit	0: Serial I/O initialization 1: Serial I/O enabled	0	○	○
5	Transfer mode selection bit	0: Full-duplex (transmit/receive) mode ($P5_0$ pin is S_{IN1} input.) 1: Transmit-only mode ($P5_0$ pin is I/O port.)	0	○	○
6	Transfer direction selection bit	0: LSB first 1: MSB first	0	○	○
7	Serial I/O1 clock pin selection bit	0: S_{CLK11} ($P5_3$/S_{CLK12} pin is I/O port.) 1: S_{CLK12} ($P5_2$/S_{CLK11} pin is I/O port.)	0	○	○

Fig. 2.3.3 Structure of Serial I/O1 control register 1

Serial I/O1 control register 2

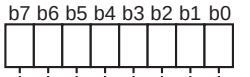
b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O1 control register 2
(SIO1CON2 • SC12: address 1A16)

b	Name	Functions	At reset	R	W
0	P62/SRDY1 • P64/SBUSY1 pin control bits	b3b2b1b0 0 0 0 0: P62, P64 pins are I/O ports. 0 0 0 1: Not used 0 0 1 0: P62 pin is SRDY1 output; P64 pin is I/O port. 0 0 1 1: P62 pin is SRDY1 output; P64 pin is I/O port. 0 1 0 0: P62 pin is I/O port; P64 pin is SBUSY1 input. 0 1 0 1: P62 pin is I/O port; P64 pin is SBUSY1 input. 0 1 1 0: P62 pin is I/O port; P64 pin is SBUSY1 output. 0 1 1 1: P62 pin is I/O port; P64 pin is SBUSY1 output. 1 0 0 0: P62 pin is SRDY1 input; P64 pin is SBUSY1 output. 1 0 0 1: P62 pin is SRDY1 input; P64 pin is SBUSY1 output. 1 0 1 0: P62 pin is SRDY1 input; P64 pin is SBUSY1 output. 1 0 1 1: P62 pin is SRDY1 input; P64 pin is SBUSY1 output. 1 1 0 0: P62 pin is SRDY1 output; P64 pin is SBUSY1 input. 1 1 0 1: P62 pin is SRDY1 output; P64 pin is SBUSY1 input. 1 1 1 0: P62 pin is SRDY1 output; P64 pin is SBUSY1 input. 1 1 1 1: P62 pin is SRDY1 output; P64 pin is SBUSY1 input.	0	○	○
1			0	○	○
2			0	○	○
3			0	○	○
4	SBUSY1 output • SSTB1 output function selection bit (Valid in serial I/O1 automatic transfer mode)	0: Functions as signal for each 1-byte 1: Functions as signal for each transfer data set	0	○	○
5	Serial transfer status flag	0: Serial transfer completed 1: Serial transfer in-progress	0	○	×
6	SOUT1 pin control bit (when serial data is not transferred)	0: Output active 1: Output high-impedance	0	○	○
7	P51/SOUT1 P-channel output disable bit	0: CMOS 3 state (P-channel output is valid.) 1: N-channel open-drain output (P-channel output is invalid.)	0	○	○

Fig. 2.3.4 Structure of Serial I/O1 control register 2

Serial I/O1 register/Transfer counter



Serial I/O1 register/Transfer counter
(SIO1: address $1B_{16}$)

b	Name	Functions	At reset	R	W
0	•In 8-bit serial I/O mode: Serial I/O1 register •In automatic transfer serial I/O mode: Transfer counter	•At function as serial I/O1 register: This register becomes the shift register to perform serial transmit/reception. Set transmit data to this register. The serial transfer is started by writing the transmit data. •At function as transfer counter: Set (transfer byte number – 1) to this register. When selecting an internal clock, the automatic transfer is started by writing the transmit data. (When selecting an external clock, after writing a value to this register, wait for 5 or more cycles of the internal system clock before inputting the transfer clock to the S_{CLK1} pin.)	Undefined	○	○
1			Undefined	○	○
2			Undefined	○	○
3			Undefined	○	○
4			Undefined	○	○
5			Undefined	○	○
6			Undefined	○	○
7			Undefined	○	○

Fig. 2.3.5 Structure of Serial I/O1 register/Transfer counter

Serial I/O1 control register 3

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O1 control register 3
(SIO1CON3 • SC13: address 1C16)

b	Name	Functions	At reset	R	W
0	Automatic transfer interval set bits (valid only when selecting internal synchronous clock)	b4b3b2b1b0 0 0 0 0 0: 2 cycles of transfer clock 0 0 0 0 1: 3 cycles of transfer clock to 1 1 1 1 0: 32 cycles of transfer clock 1 1 1 1 1: 33 cycles of transfer clock Data is written into the latch and read from the decrement counter.	0	○	○
1			0	○	○
2			0	○	○
3			0	○	○
4			0	○	○
5	Internal synchronous clock selection bits	b7b6b5 0 0 0 : f(XIN)/4 or f(XCIN)/8 0 0 1 : f(XIN)/8 or f(XCIN)/16 0 1 0 : f(XIN)/16 or f(XCIN)/32 0 1 1 : f(XIN)/32 or f(XCIN)/64 1 0 0 : f(XIN)/64 or f(XCIN)/128 1 0 1 : f(XIN)/128 or f(XCIN)/256 1 1 0 : f(XIN)/256 or f(XCIN)/512 1 1 1 : Not used	0	○	○
6			0	○	○
7			0	○	○

Fig. 2.3.6 Structure of Serial I/O1 control register 3

(2) Serial I/O2

Baud rate generator

b7 b6 b5 b4 b3 b2 b1 b0

Baud rate generator
(BRG: address 16_{16})

b	Functions	At reset	R	W
0	• Bit rate of the serial transfer is determined. • This is the 8-bit counter and has the reload register. The count source is divided by n+1 owing to specifying a value n.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 2.3.7 Structure of Baud rate generator

UART control register

b7 b6 b5 b4 b3 b2 b1 b0

UART control register
(UARTCON: address 17_{16})

b	Name	Functions	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	$P5_5$/TxD P-channel output disable bit (POFF)	0: CMOS output (in output mode) 1: N-channel open-drain output (in output mode)	0	○	○
5	BRG clock switch bit	0: X_{IN} or X_{CIN}/2 (depending on internal system clock) 1: X_{CIN}	0	○	○
6	Serial I/O2 clock I/O pin selection bit	0: S_{CLK21} ($P5_7$/S_{CLK22} pin is used as I/O port or $\overline{S_{RDY2}}$ output pin.) 1: S_{CLK22} ($P5_6$/S_{CLK21} pin is used as I/O port.)	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are "1".		1	○	×

Fig. 2.3.8 Structure of UART control register

Serial I/O2 control register

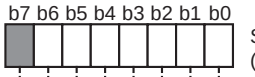
b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O2 control register (SIO2CON: address $1D_{16}$)

b	Name	Functions	At reset	R	W
0	BRG count source selection bit (CSS)	0: f(XIN) or f(XCIN)/2 or f(XCIN) 1: f(XIN)/4 or f(XCIN)/8 or f(XCIN)/4	0	○	○
1	Serial I/O2 synchronous clock selection bit (SCS)	•In clock synchronous mode 0: BRG output/4 1: External clock input •In UART mode 0: BRG output/16 1: External clock input/16	0	○	○
2	$\overline{SRDY2}$ output enable bit (SRDY)	0: P57 pin operates as normal I/O pin 1: $\overline{P57}$ pin operates as $\overline{SRDY2}$ 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/O2 mode selection bit (SIOM)	0: Clock asynchronous serial I/O (UART) mode 1: Clock synchronous serial I/O mode	0	○	○
7	Serial I/O2 enable bit (SIOE)	0: Serial I/O2 disabled (pins P54–P57 operate as normal I/O pins) 1: Serial I/O2 enabled (pins P54–P57 operate as serial I/O pins)	0	○	○

Fig. 2.3.9 Structure of Serial I/O2 control register

Serial I/O2 status register

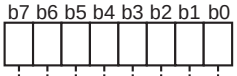


Serial I/O2 status register
(SIO2STS: address 1E16)

b	Name	Functions	At reset	R	W
0	Transmit buffer empty flag (TBE)	0: Buffer full 1: Buffer empty	0	○	○
1	Receive buffer full flag (RBF)	0: Buffer empty 1: Buffer full	0	○	○
2	Transmit shift register shift completion flag (TSC)	0: Transmit shift in progress 1: Transmit shift completed	0	○	○
3	Overrun error flag (OE)	0: No error 1: Overrun error	0	○	○
4	Parity error flag (PE)	0: No error 1: Parity error	0	○	○
5	Framing error flag (FE)	0: No error 1: Framing error	0	○	○
6	Summing error flag (SE)	0: (OE) U (PE) U (FE) = 0 1: (OE) U (PE) U (FE) = 1	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “1”.		1	○	×

Fig. 2.3.10 Structure of Serial I/O2 status register

Serial I/O2 transmit/receive buffer register



Serial I/O2 transmit/receive buffer register
(TB/RB: address 1F16)

b	Functions	At reset	R	W
0	This is the buffer register which is used to write transmit data or to read receive data. • At write : The value is written to the transmit buffer register. The value cannot be written to the receive buffer register. • At read : The contents of the receive buffer register is read out. When a character bit length is 7 bits, the MSB of data stored in the receive buffer is “0”. The contents of the transmit buffer register cannot be read out.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 2.3.11 Structure of Serial I/O2 transmit/receive buffer register

(3) Serial I/O1 and Serial I/O2

Interrupt source switch register

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt source switch register
(IFR: address 39_{16})

b	Name	Functions	At reset	R	W
0	INT_3/serial I/O2 transmit interrupt switch bit **(Note)**	0: INT_3 intrrupt 1: Serial I/O2 transmit interrupt	0	○	○
1	INT_4/A-D conversion interrupt switch bit	0: INT_4 interrupt 1: A-D conversion intrerrupt	0	○	○
2	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are "0".		0	○	×
3			0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Note: In the mask option type P, this bit is not available because INT_3 funciton is not used.

Fig. 2.3.12 Structure of Interrupt source switch register

Interrupt request register 1

b7 b6 b5 b4 b3 b2 b1 b0

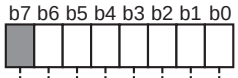
Interrupt request register 1
(IREQ1 : address $3C_{16}$)

b	Name	Functions	At reset	R	W
0	INT_0 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	INT_1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	INT_2 interrupt request bit Remote controller /counter overflow interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O1 interrupt request bit Serial I/O automatic transfer interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	Timer 2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Timer 3 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻

✻: "0" can be set by software, but "1" cannot be set.

Fig. 2.3.13 Structure of Interrupt request register 1

Interrupt request register 2



Interrupt request register 2
(IREQ2 : address $3D_{16}$)

b	Name	Functions	At reset	R	W
0	Timer 4 interrupt request bit **(Note)**	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Timer 5 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	Timer 6 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O2 receive interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	INT_3/Serial I/O2 transmit interrupt request bit **(Note)**	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	INT_4 interrupt request bit A-D converter interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	FLD blanking interrupt request bit FLD digit interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are "0".		0	○	×

✻: "0" can be set by software, but "1" cannot be set.

Note: In the mask option type P, if timer 4 interrupt whose count source is $CNTR_1$ input and INT_3 interrupt are selected, these bits do not become "1".

Fig. 2.3.14 Structure of Interrupt request register 2

Interrupt control register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 1
(ICON1 : address 3E16)

b	Name	Functions	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	INT2 interrupt enable bit Remote controller /counter overflow interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O1 interrupt enable bit Serial I/O automatic transfer interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Timer 2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	Timer 3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Fig. 2.3.15 Structure of Interrupt control register 1

Interrupt control register 2

b7 b6 b5 b4 b3 b2 b1 b0
0

Interrupt control register 2
(ICON2 : address 3F16)

b	Name	Functions	At reset	R	W
0	Timer 4 interrupt enable bit **(Note)**	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer 5 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer 6 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O2 receive interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
4	INT3/Serial I/O2 transmit interrupt enable bit **(Note)**	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
5	INT4 interrupt enable bit A-D converter interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
6	FLD blanking interrupt enable bit FLD digit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
7	Fix “0” to this bit.		0	○	○

Note: In the mask option type P, timer 4 interrupt whose count source is CNTR1 input and INT3 interrupt are not available.

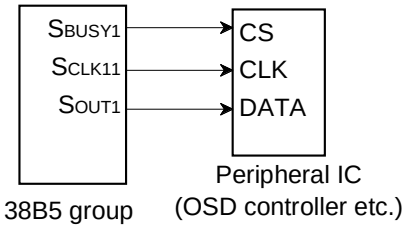
Fig. 2.3.16 Structure of Interrupt control register 2

2.3.3 Serial I/O1 connection examples

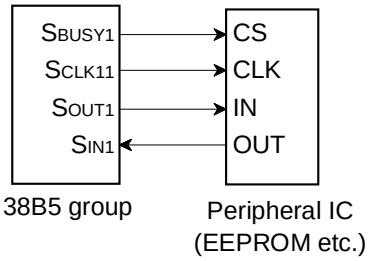
(1) Control of peripheral IC equipped with CS pin

Figure 2.3.17 shows connection examples with peripheral ICs equipped with the CS pin. All examples can use the automatic transfer function.

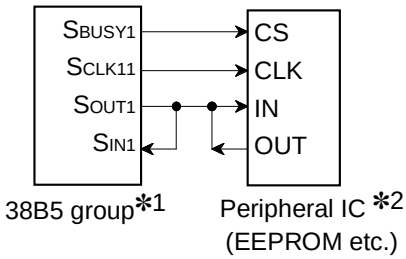
(1) Only transmission
(Using SIN1 pin as I/O port)



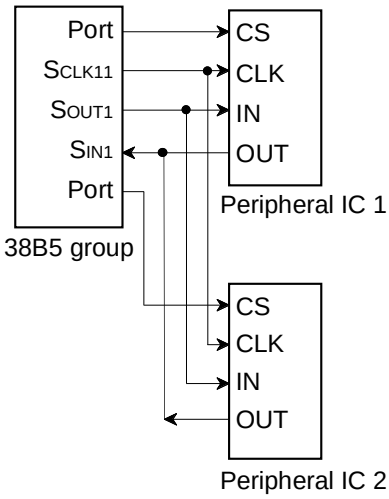
(2) Transmission and reception



(3) Transmission and reception
(When connecting SIN1 with SOUT1)
(When connecting IN with OUT in peripheral IC)



(4) Connection of plural IC



*1: Select an N-channel open-drain output for SOUT1 pin output control.

*2: Use the OUT pin of peripheral IC which is an N-channel open-drain output and becomes high impedance during receiving data.

Note: "Port" means an output port controlled by software.

Fig. 2.3.17 Serial I/O1 connection examples (1)

(2) Connection with microcomputer

Figure 2.3.18 shows connection examples with another microcomputer.

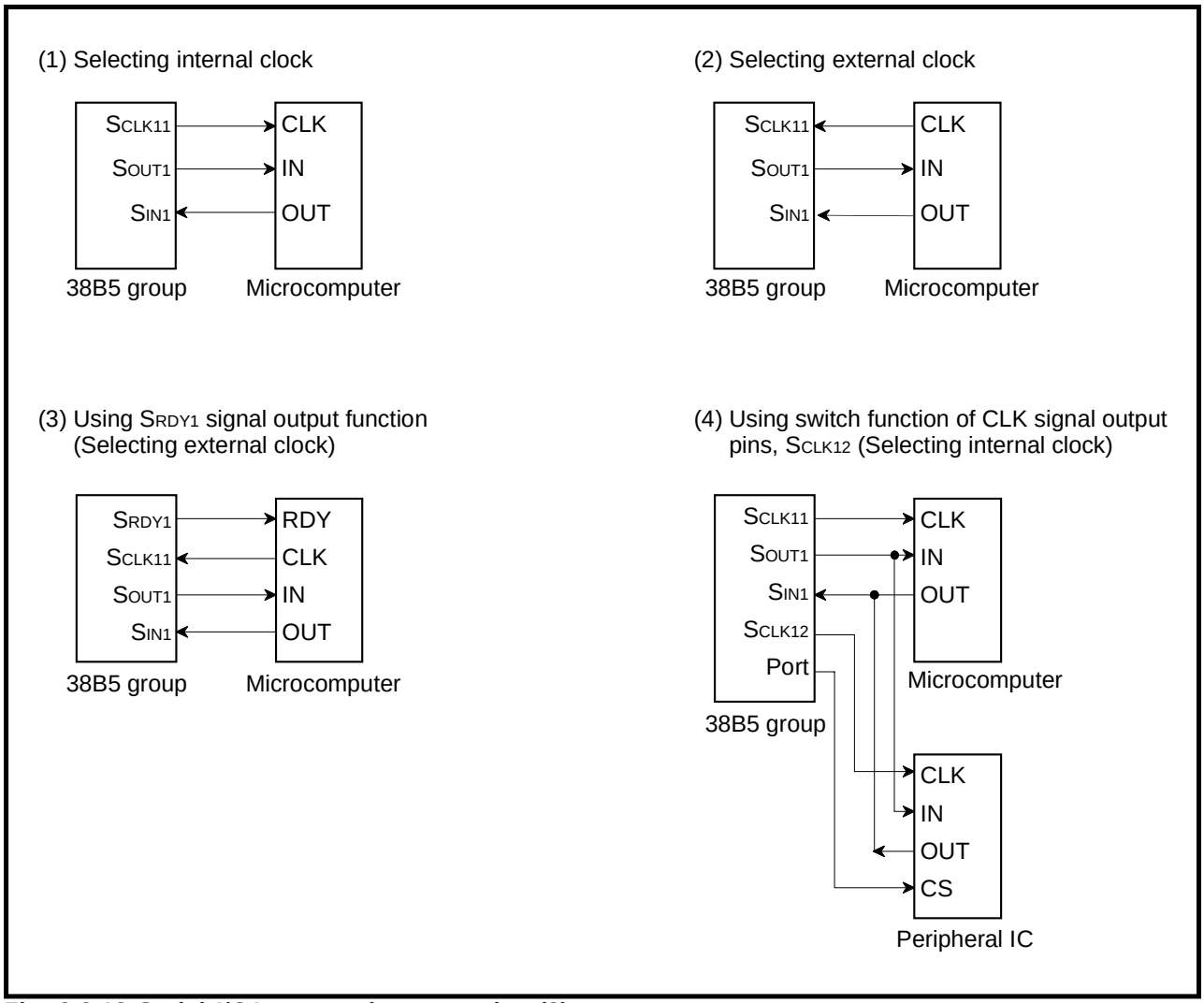


Fig. 2.3.18 Serial I/O1 connection examples (2)

2.3.4 Serial I/O1's modes

Figure 2.3.19 shows the serial I/O1's modes.

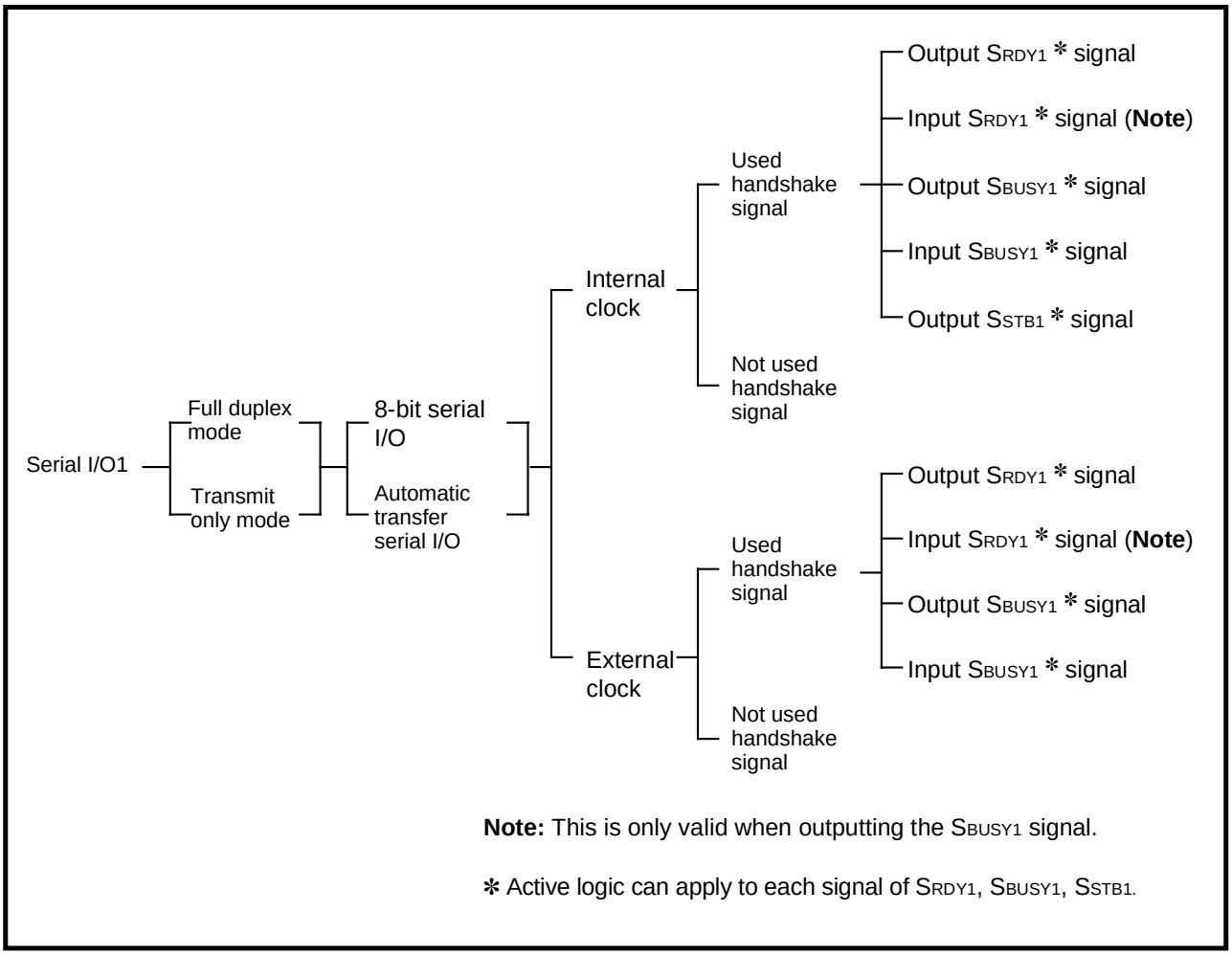


Fig. 2.3.19 Serial I/O1's modes

2.3.5 Serial I/O1 application examples

(1) Output of serial data (control of peripheral IC)

Outline : Serial communication is performed, connecting ports with the $\overline{CS}$ pin of a peripheral IC.

Figure 2.3.20 shows a connection diagram, and Figure 2.3.21 shows a timing chart.

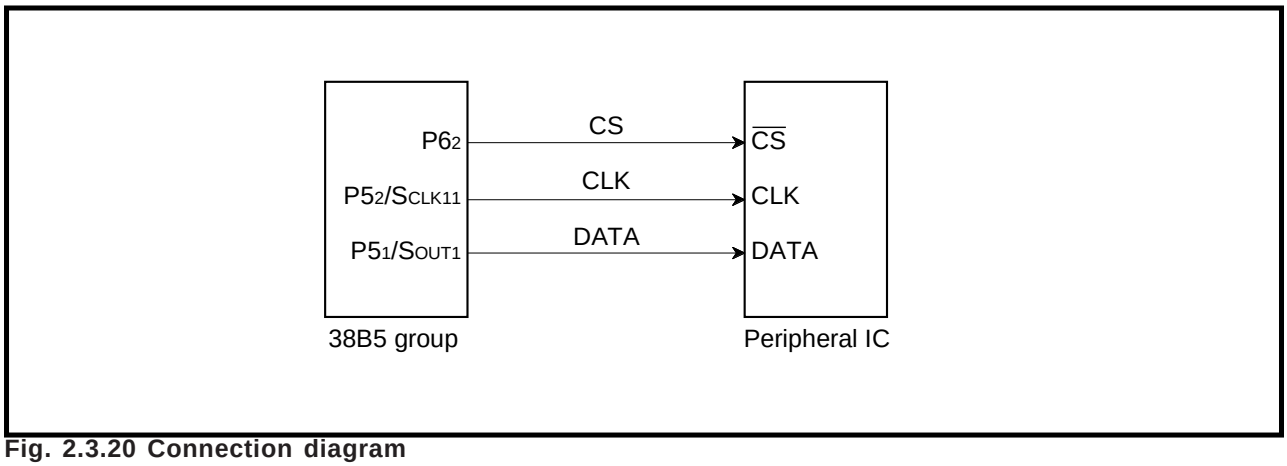


Fig. 2.3.20 Connection diagram

Specifications :
- Use of serial I/O1 (Not using automatic transfer function)
- Synchronous clock frequency : 131 kHz ($f(X_{IN})$ = 4.19 MHz is divided by 32)
- Transfer direction : LSB first
- Not use of serial I/O1 interrupt
- Port $P6_2$ is connected to the $\overline{CS}$ pin ("L" active) of the peripheral IC for transmission control; the output level of port $P6_2$ is controlled by software.

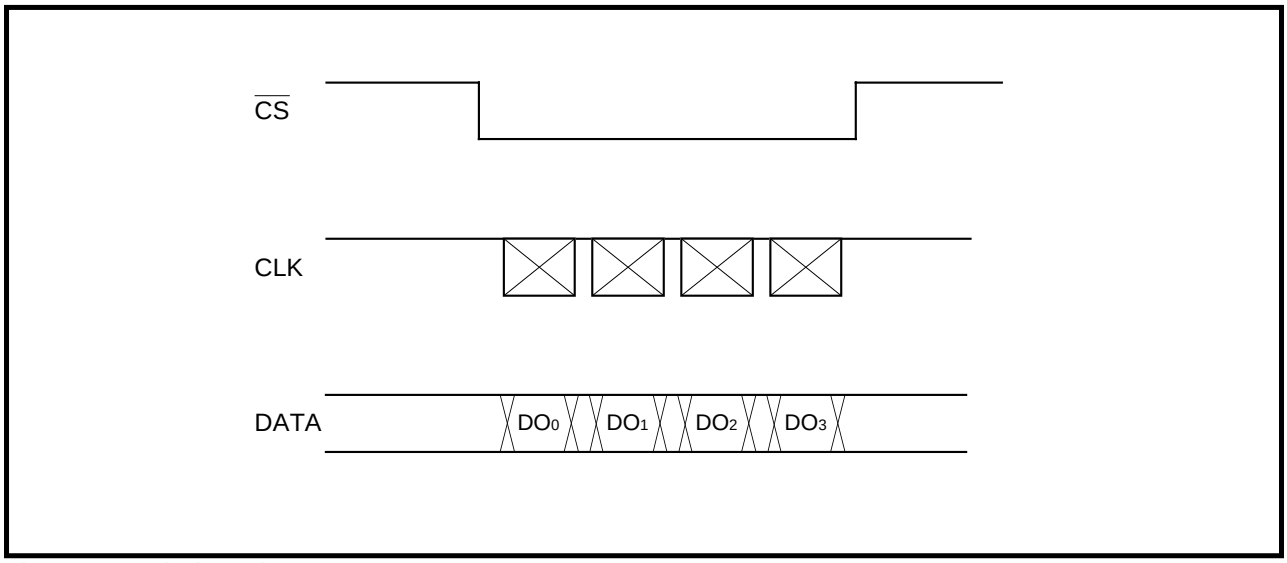


Fig. 2.3.21 Timing chart

Figure 2.3.22 shows the registers setting relevant to the transmission side, and Figure 2.3.23 shows the setting of transmission data.

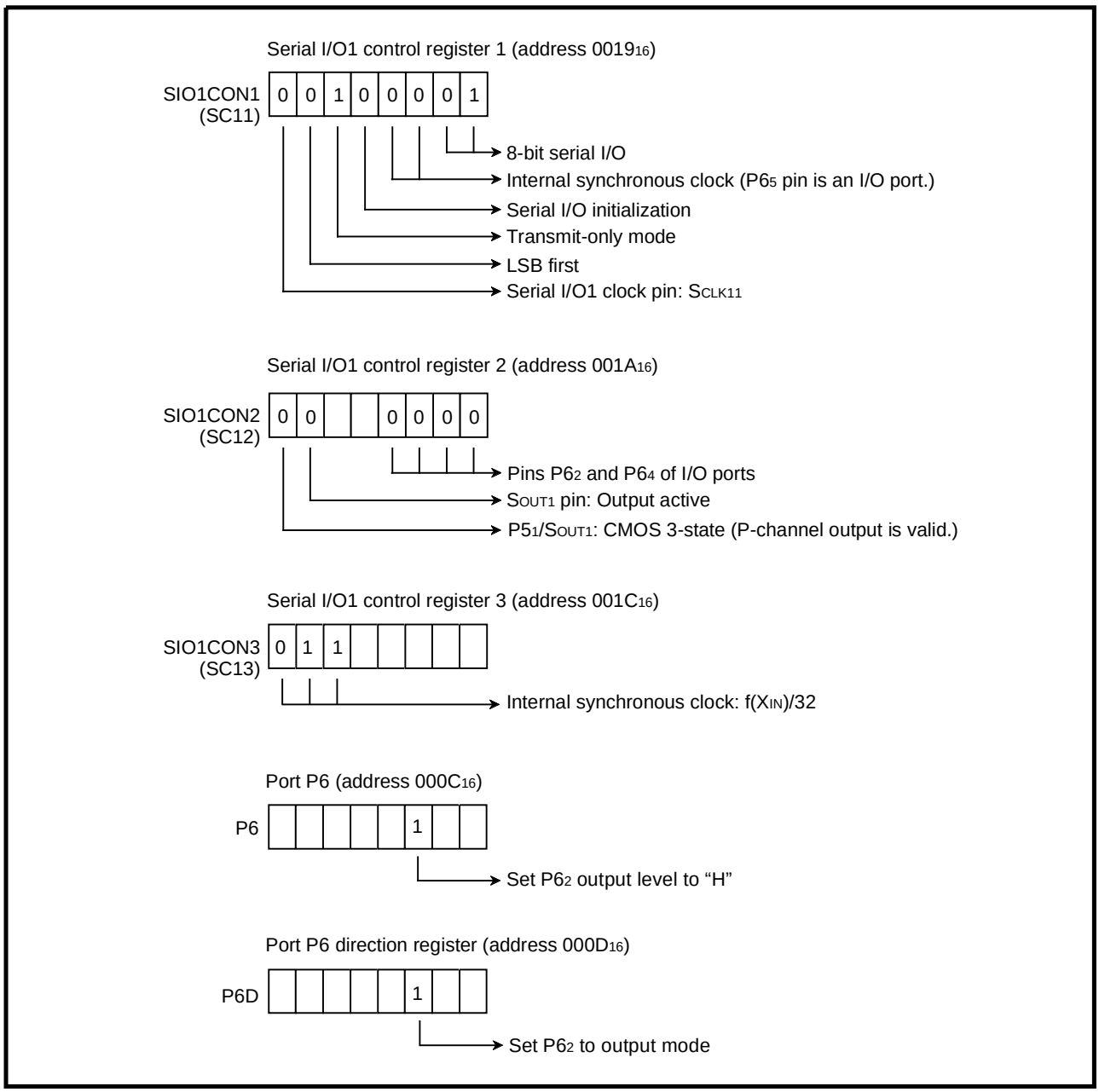


Fig. 2.3.22 Registers setting relevant to transmission side

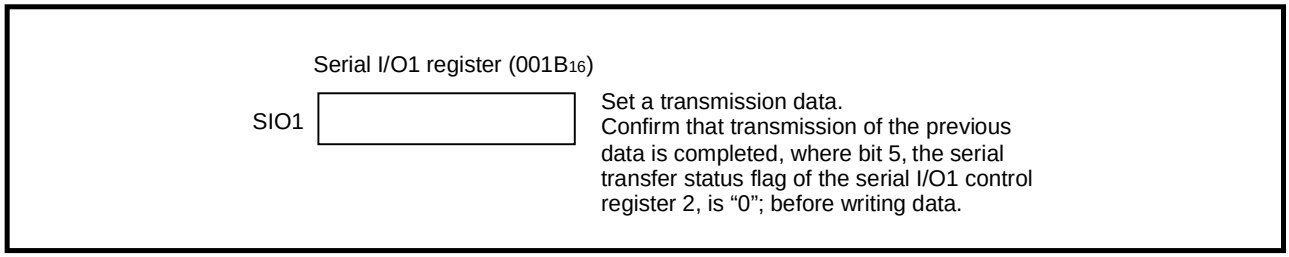


Fig. 2.3.23 Setting of transmission data

Control procedure: When the registers are set as shown in Figure 2.3.22, the serial I/O1 can transmit 1-byte data by writing data to the serial I/O1 register.
Thus, after setting the $\overline{CS}$ signal to "L", write the transmission data to the serial I/O1 register by each 1 byte; and return the $\overline{CS}$ signal to "H" when the target number of bytes has been transmitted.
Figure 2.3.24 shows a control procedure.

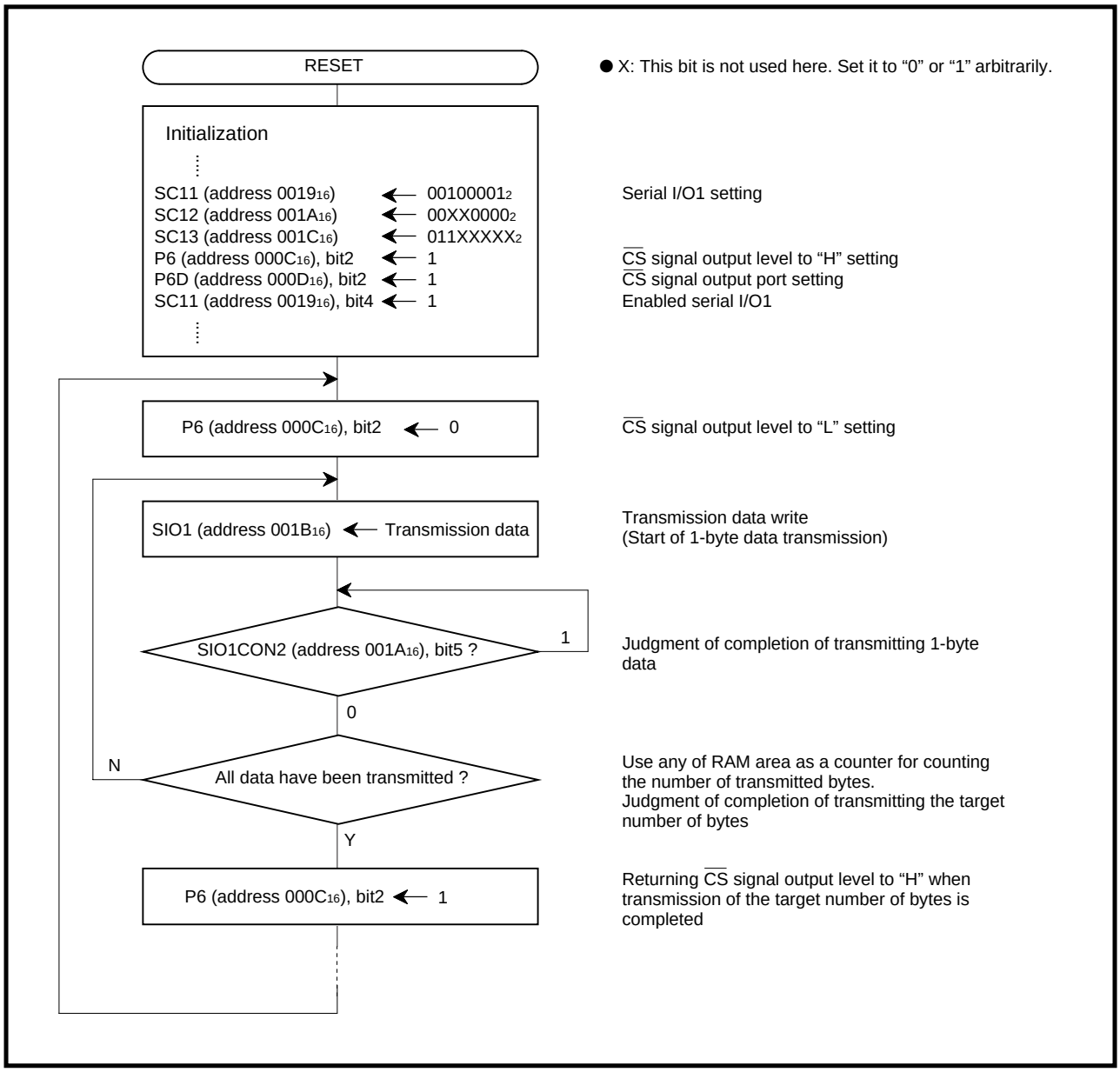


Fig. 2.3.24 Control procedure

(2) Transmission/Reception using automatic transfer

Outline: Serial transmission/reception control is performed, using the serial automatic transfer function.

Figure 2.3.25 shows a connection diagram, and Figure 2.3.26 shows a timing chart of serial data transmission/reception.

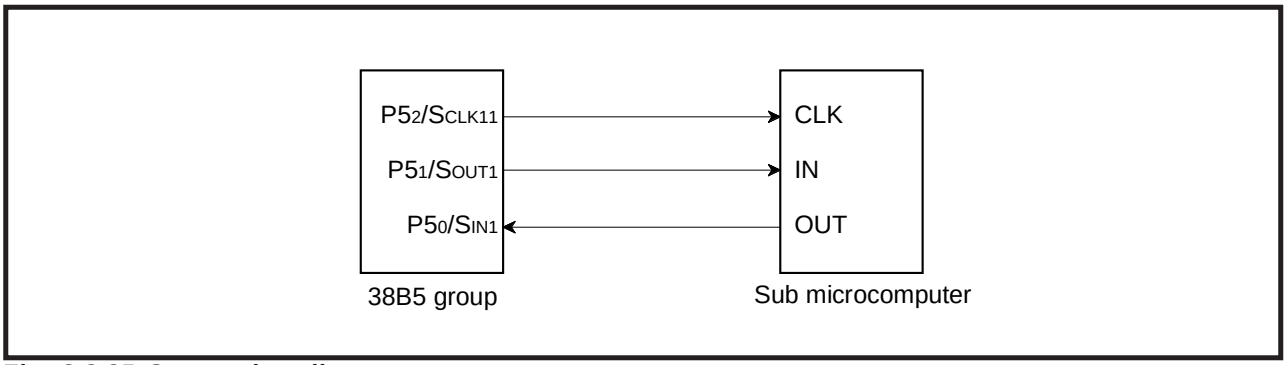


Fig. 2.3.25 Connection diagram

Specifications:
- Use of serial I/O1 using automatic transfer function
- Synchronous clock frequency: 131 kHz ($f(X_{IN})$ = 4.19 MHz is divided by 32.)
- Transfer direction: LSB first
- Transmission/reception byte number: 8 bytes/block each
- Transfer interval for 1-byte: 244 μs (32 cycles of transfer clock)
- Not use of serial I/O1 automatic transfer interrupt

Figure 2.3.27 shows the relevant registers setting, and Figure 2.3.28 shows the control procedure.

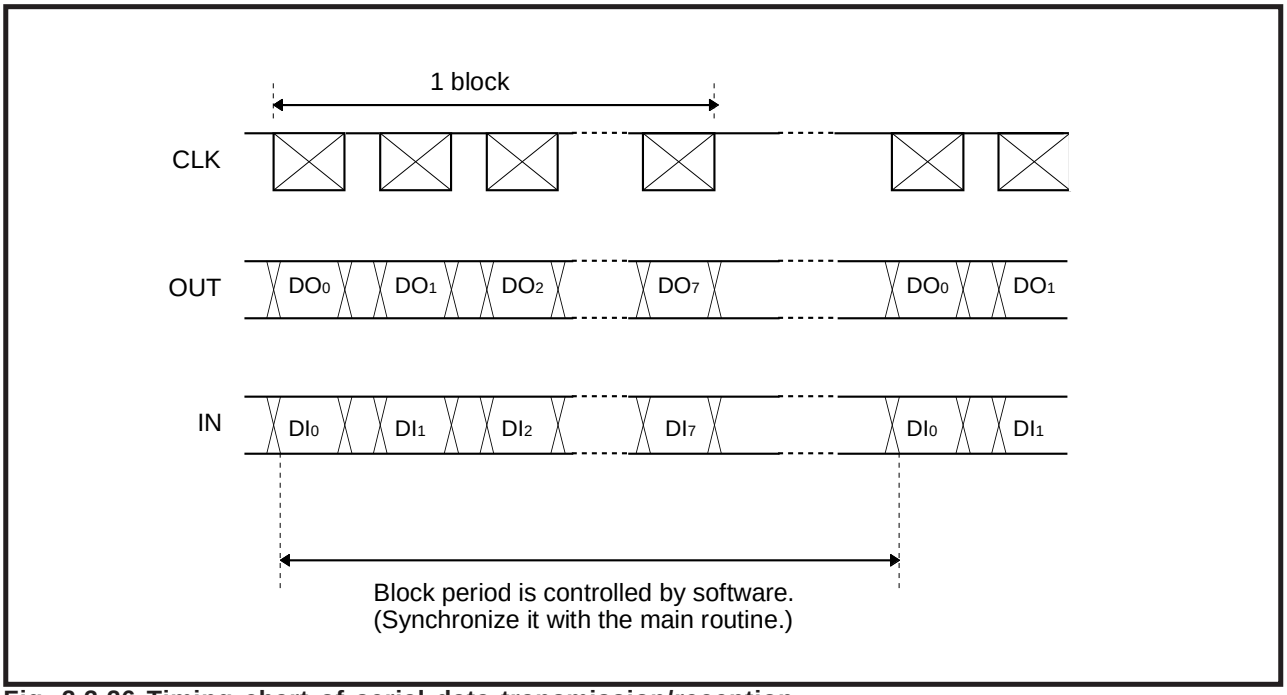


Fig. 2.3.26 Timing chart of serial data transmission/reception

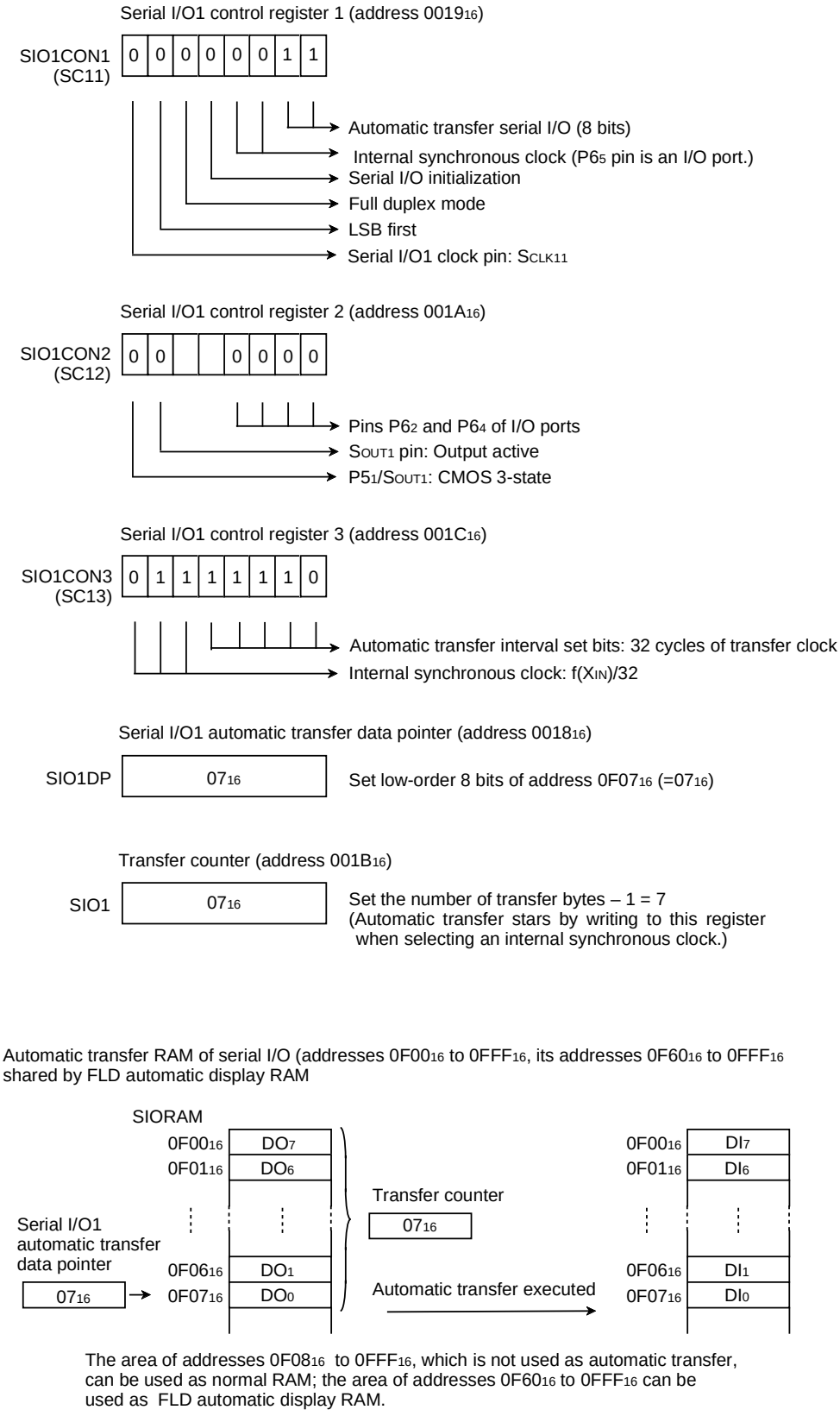


Fig. 2.3.27 Relevant registers setting

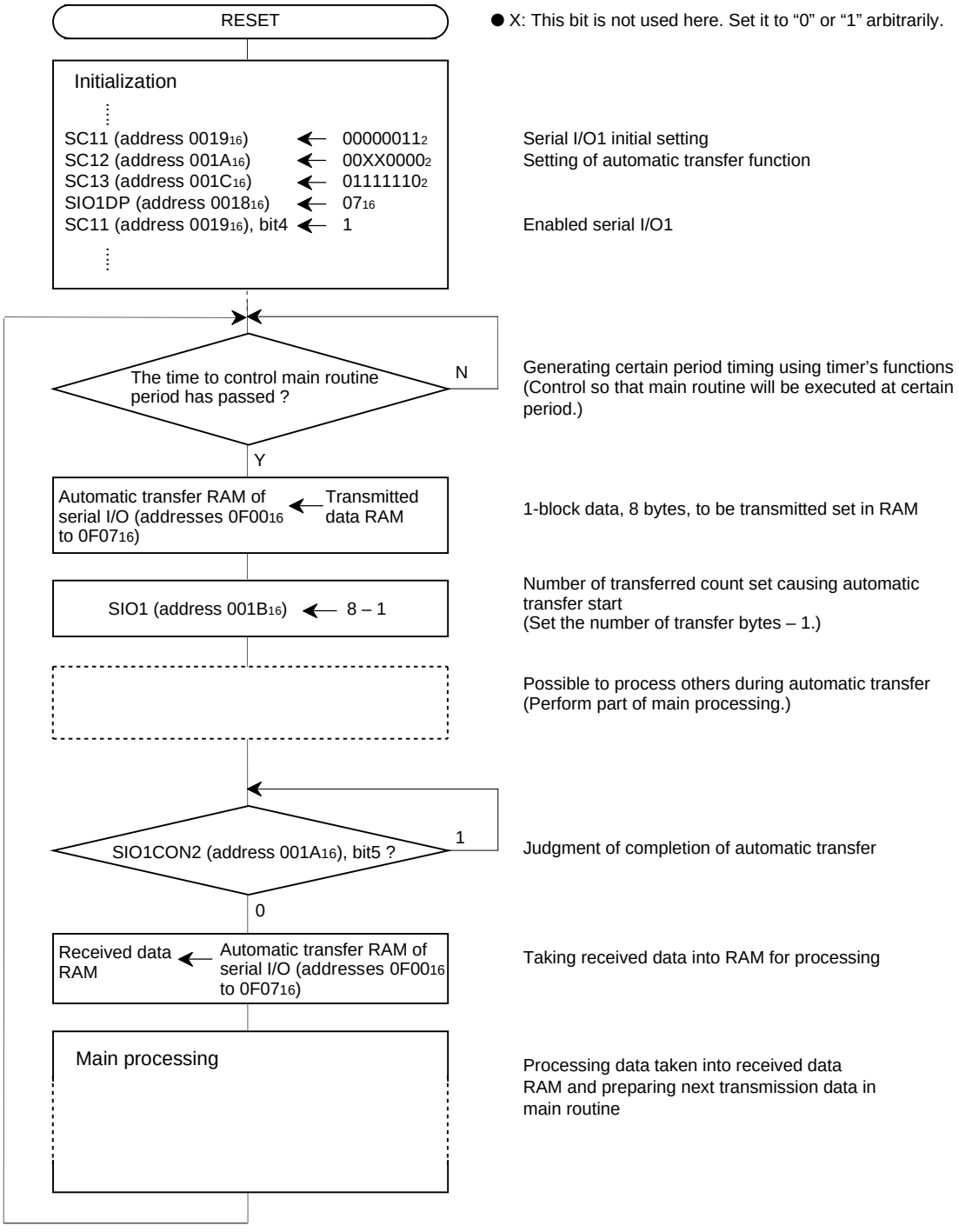


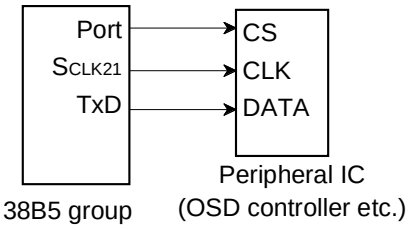
Fig. 2.3.28 Control procedure

2.3.6 Serial I/O2 connection examples

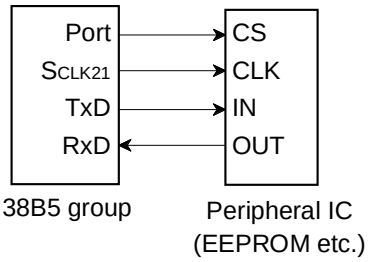
(1) Control of peripheral IC equipped with CS pin

Figure 2.3.29 shows connection examples with peripheral ICs equipped with the CS pin.

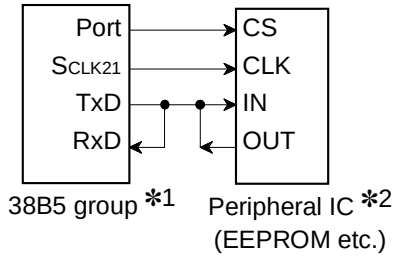
(1) Only transmission
(Using RxD pin as I/O port)



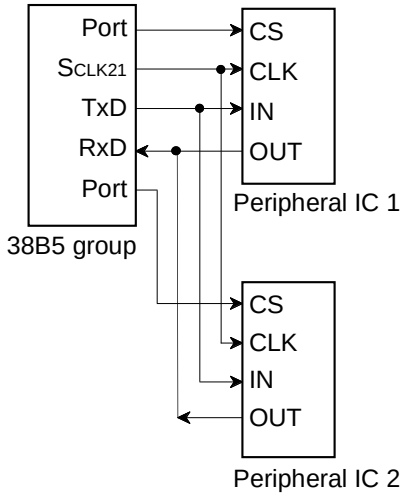
(2) Transmission and reception



(3) Transmission and reception
(When connecting RxD with TxD)
(When connecting IN with OUT in peripheral IC)



(4) Connection of plural IC



✻1: Select an N-channel open-drain output for TxD pin output control.

✻2: Use the OUT pin of peripheral IC which is an N-channel open-drain output and becomes high impedance during receiving data.

Note: "Port" means an output port controlled by software.

Fig. 2.3.29 Serial I/O2 connection examples (1)

(2) Connection with microcomputer

Figure 2.3.30 shows connection examples with another microcomputer.

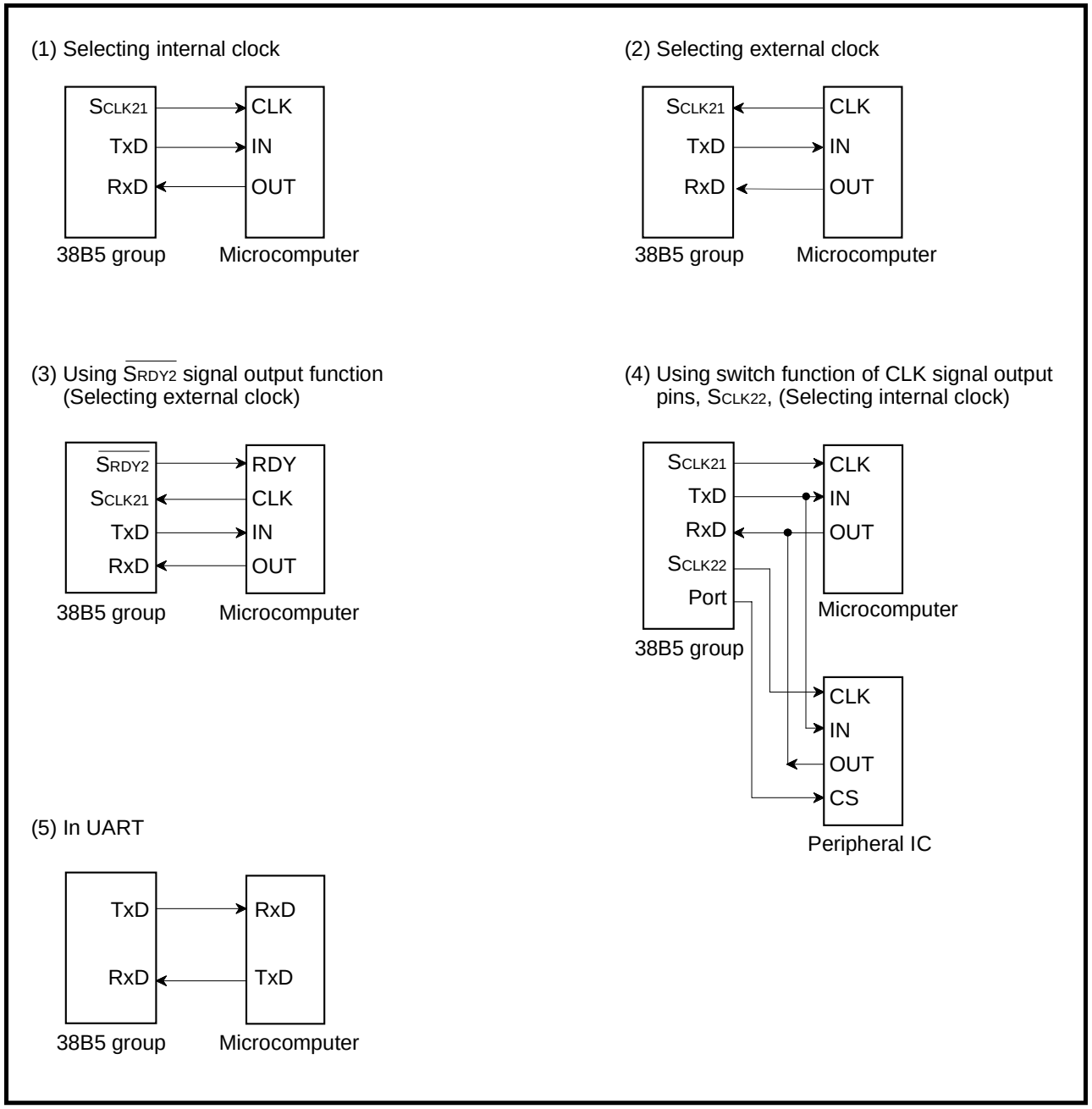


Fig. 2.3.30 Serial I/O2 connection examples (2)

2.3 Serial I/O

2.3.7 Serial I/O2's modes

A clock synchronous or clock asynchronous (UART) can be selected for the serial I/O2.
Figure 2.3.31 shows the serial I/O2's modes, and Figure 2.3.32 shows the serial I/O2 transfer data format.

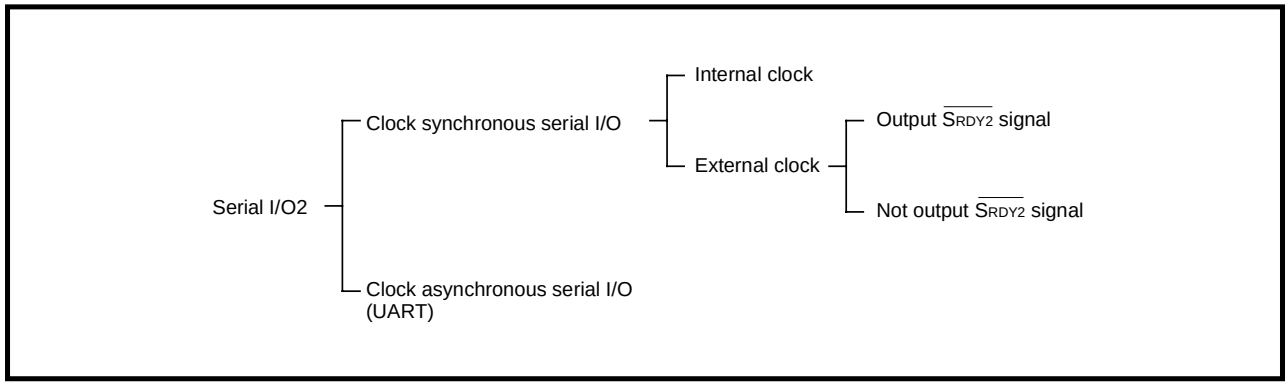


Fig. 2.3.31 Serial I/O2's modes

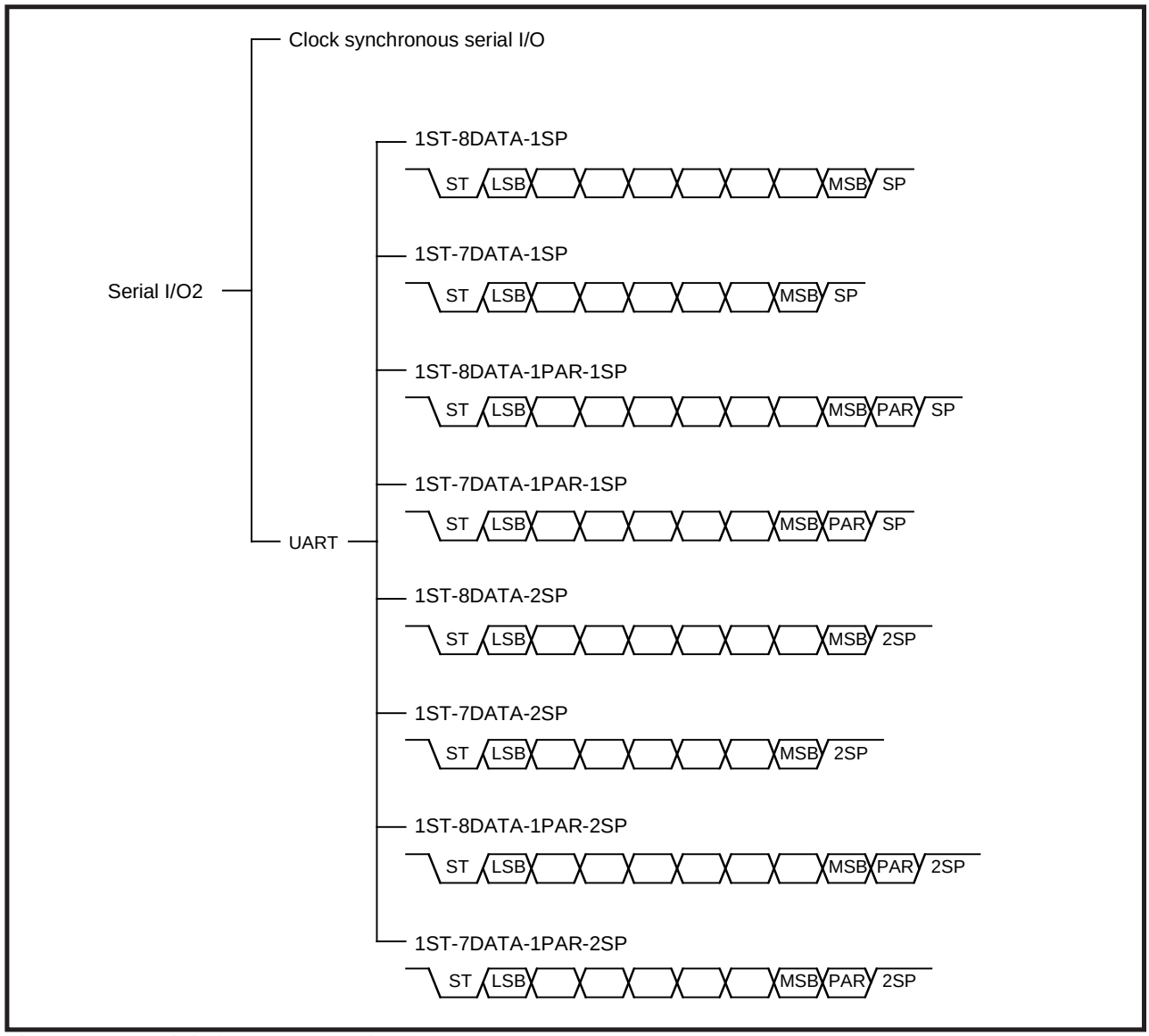


Fig. 2.3.32 Serial I/O2 transfer data format

2.3.8 Serial I/O2 application examples

(1) Communication (transmission/reception) using clock synchronous serial I/O

Outline : 2-byte data is transmitted and received, using the clock synchronous serial I/O. The $\overline{S_{RDY2}}$ signal is used for communication control.

Figure 2.3.33 shows a connection diagram, and Figure 2.3.34 shows a timing chart.

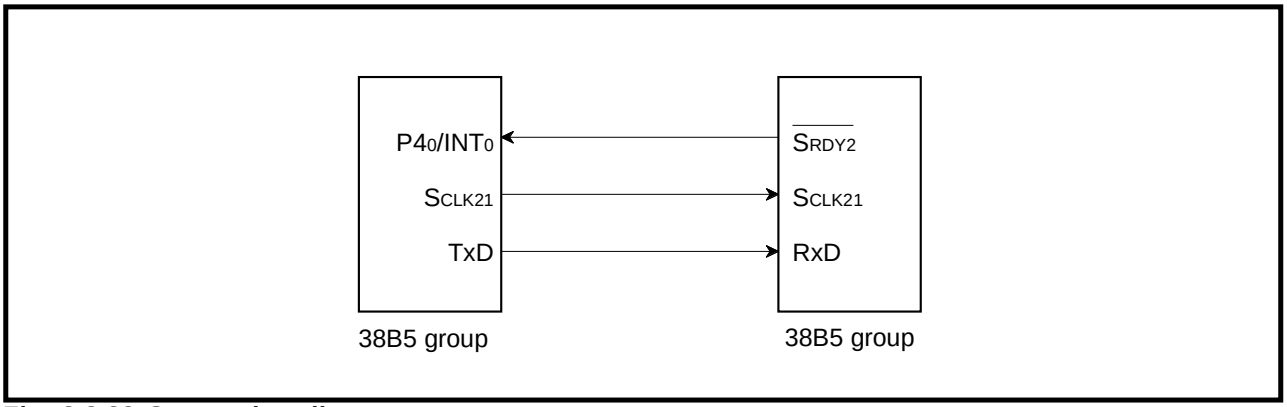


Fig. 2.3.33 Connection diagram

Specifications :
- Use of serial I/O2 in clock synchronous serial I/O
- Synchronous clock frequency : 125 kHz ($f(X_{IN})$ = 4 MHz is divided by 32)
- Use of $\overline{S_{RDY2}}$ (receivable signal)
- The reception side outputs the $\overline{S_{RDY2}}$ signal at intervals of 2 ms (generated by the timer), and 2-byte data is transferred from the transmission side to the reception side.

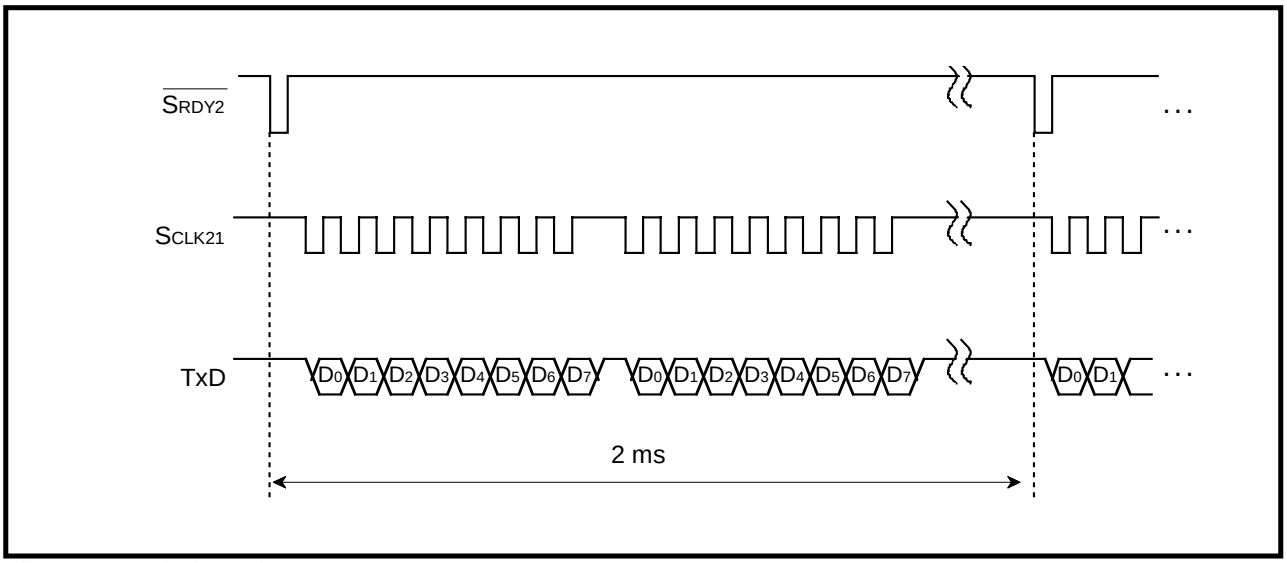


Fig. 2.3.34 Timing chart

2.3 Serial I/O

Figure 2.3.35 shows the registers setting relevant to the transmission side, and Figure 2.3.36 shows the registers setting relevant to the reception side.

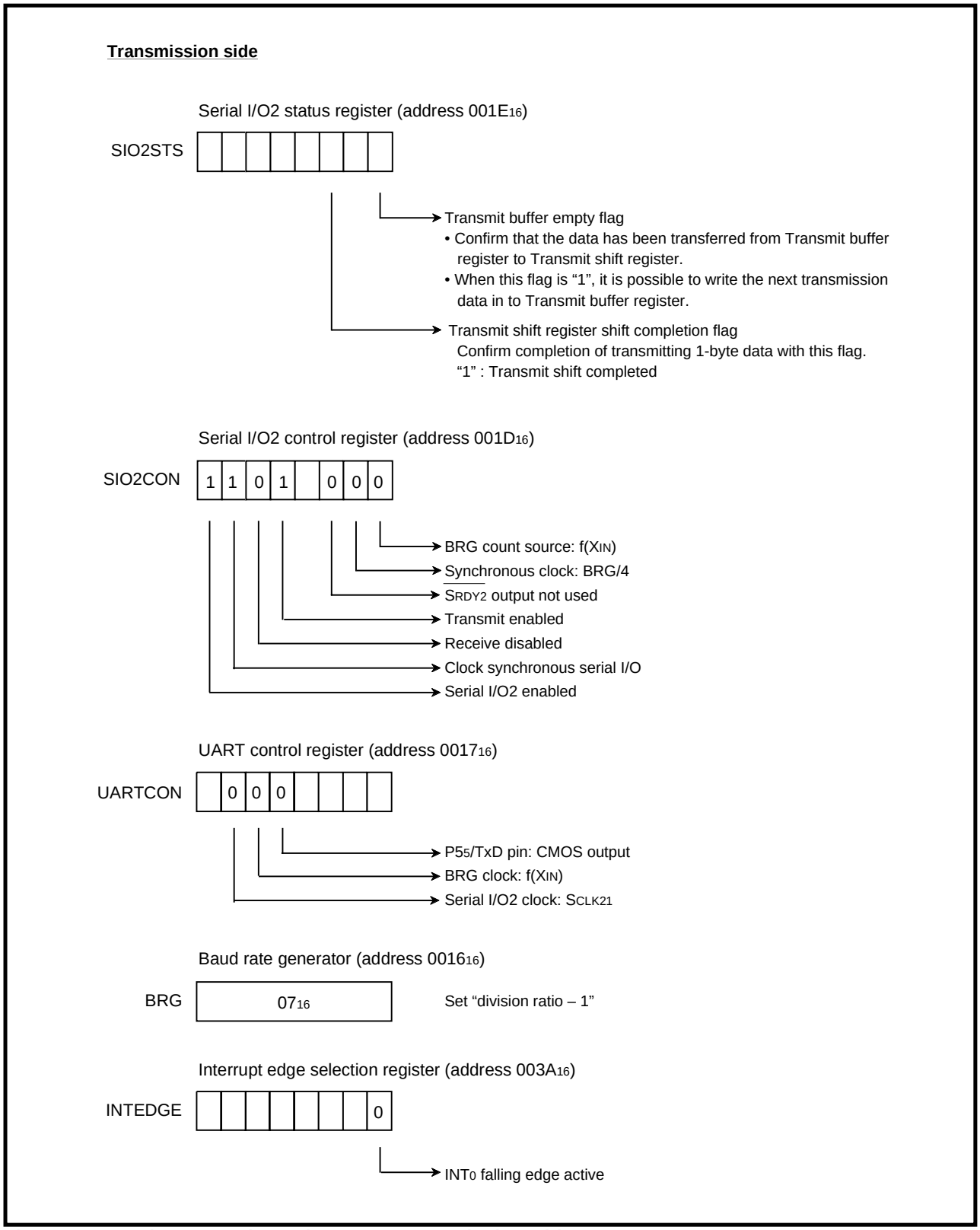


Fig. 2.3.35 Registers setting relevant to transmission side

Reception side

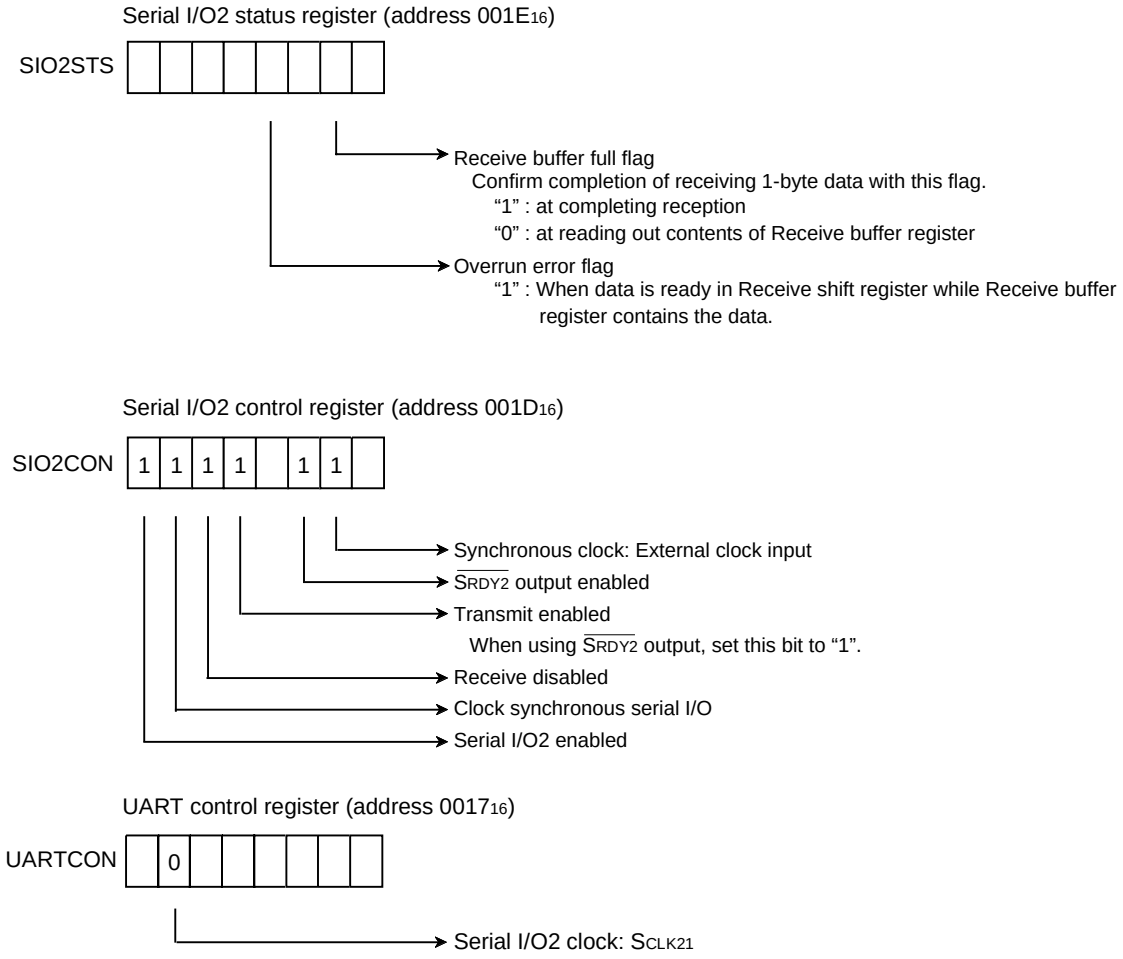


Fig. 2.3.36 Registers setting relevant to reception side

Figure 2.3.37 shows a control procedure of the transmission side, and Figure 2.3.38 shows a control procedure of the reception side.

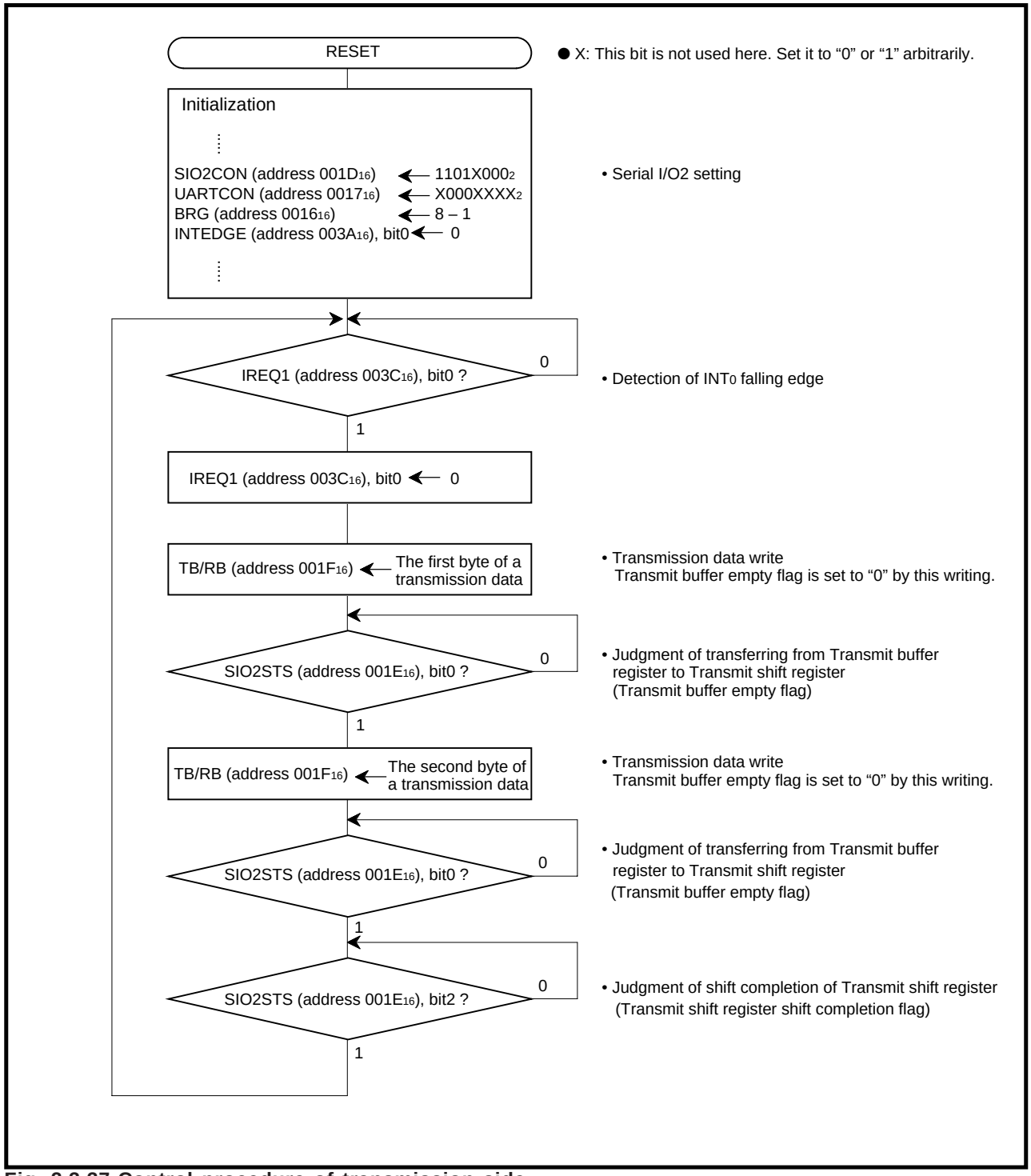
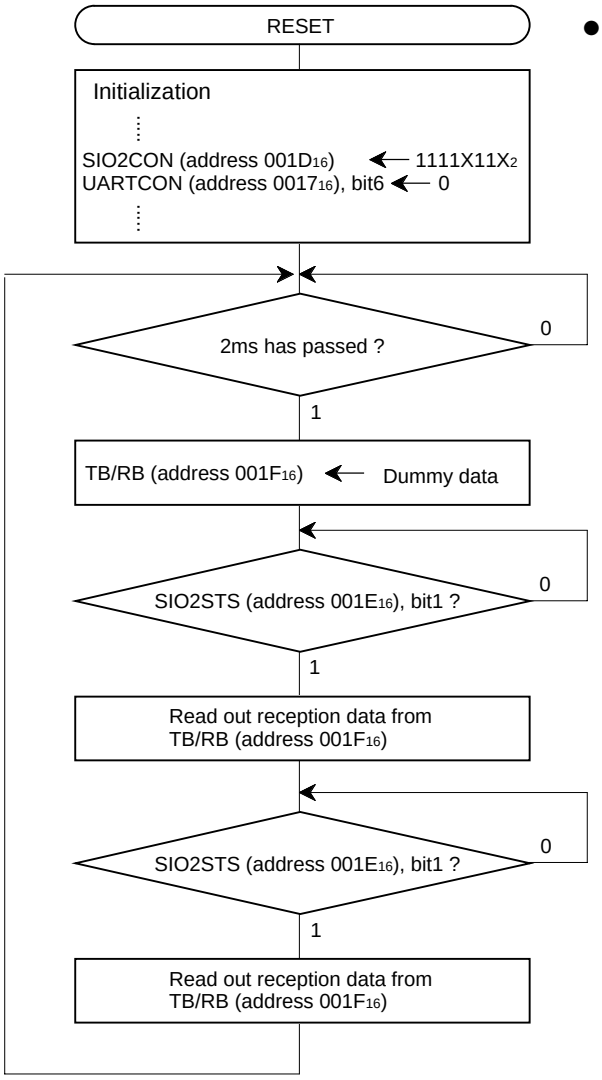


Fig. 2.3.37 Control procedure of transmission side



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

• Serial I/O2 setting

• An interval of 2 ms generated by Timer.

• $\overline{SRDY2}$ output
$\overline{SRDY2}$ signal is output by writing data to the TB/RB.
When using $\overline{SRDY2}$, set Transmit enable bit (bit4) of SIO2CON to "1."

• Judgment of completion of receiving (Receive buffer full flag)

• Reception of the first byte data.
Receive buffer full flag is set to "0" by reading data.

• Judgment of completion of receiving (Receive buffer full flag)

• Reception of the second byte data.
Receive buffer full flag is set to "0" by reading data.

Fig. 2.3.38 Control procedure of reception side

(2) Output of serial data (control of peripheral IC)

Outline : Serial communication is performed, connecting port $P5_7$ with the $\overline{CS}$ pin of a peripheral IC.

Figure 2.3.39 shows a connection diagram, and Figure 2.3.40 shows a timing chart.

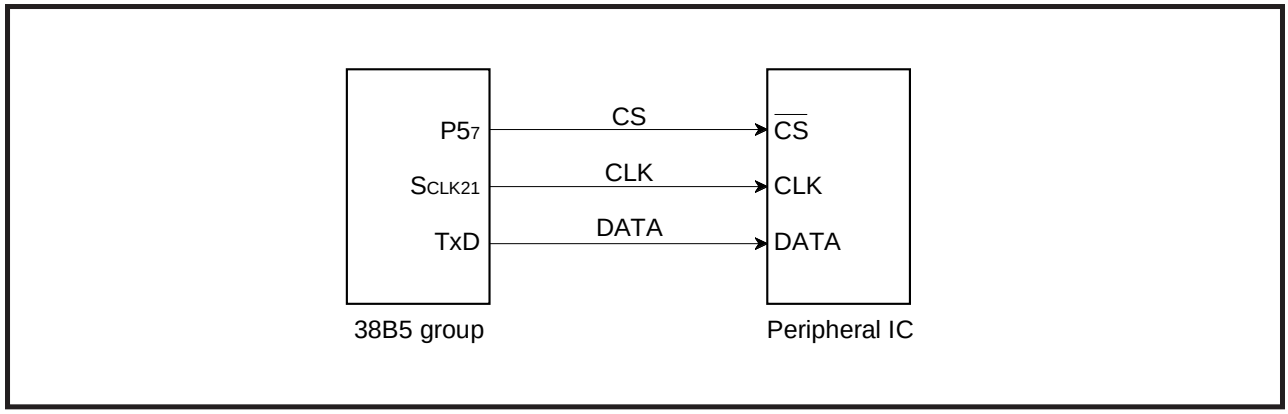


Fig. 2.3.39 Connection diagram

Specifications :
- Use of serial I/O2 in clock synchronous serial I/O
- Synchronous clock frequency : 125 kHz ($f(X_{IN})$ = 4 MHz is divided by 32)
- Transfer direction : LSB first
- Not use of receive/transmit interrupts of serial I/O2
- Port $P5_7$ is connected with the $\overline{CS}$ pin ("L" active) of the peripheral IC for transmission control; the output level of port $P5_7$ is controlled by software.

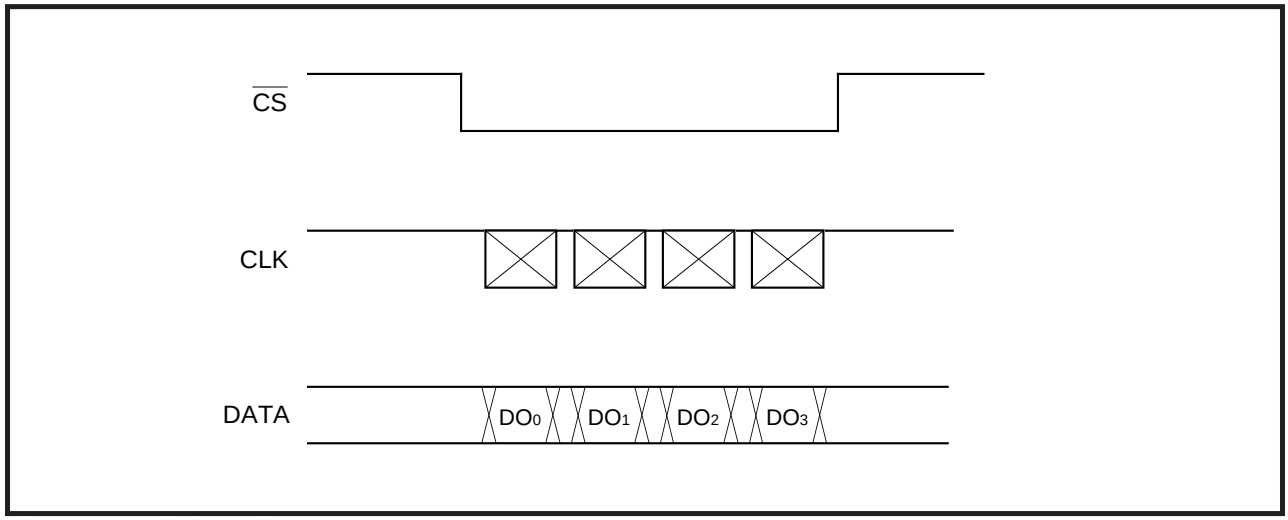


Fig. 2.3.40 Timing chart

Figure 2.3.41 shows the relevant registers setting and Figure 2.3.42 shows the setting of transmission data.

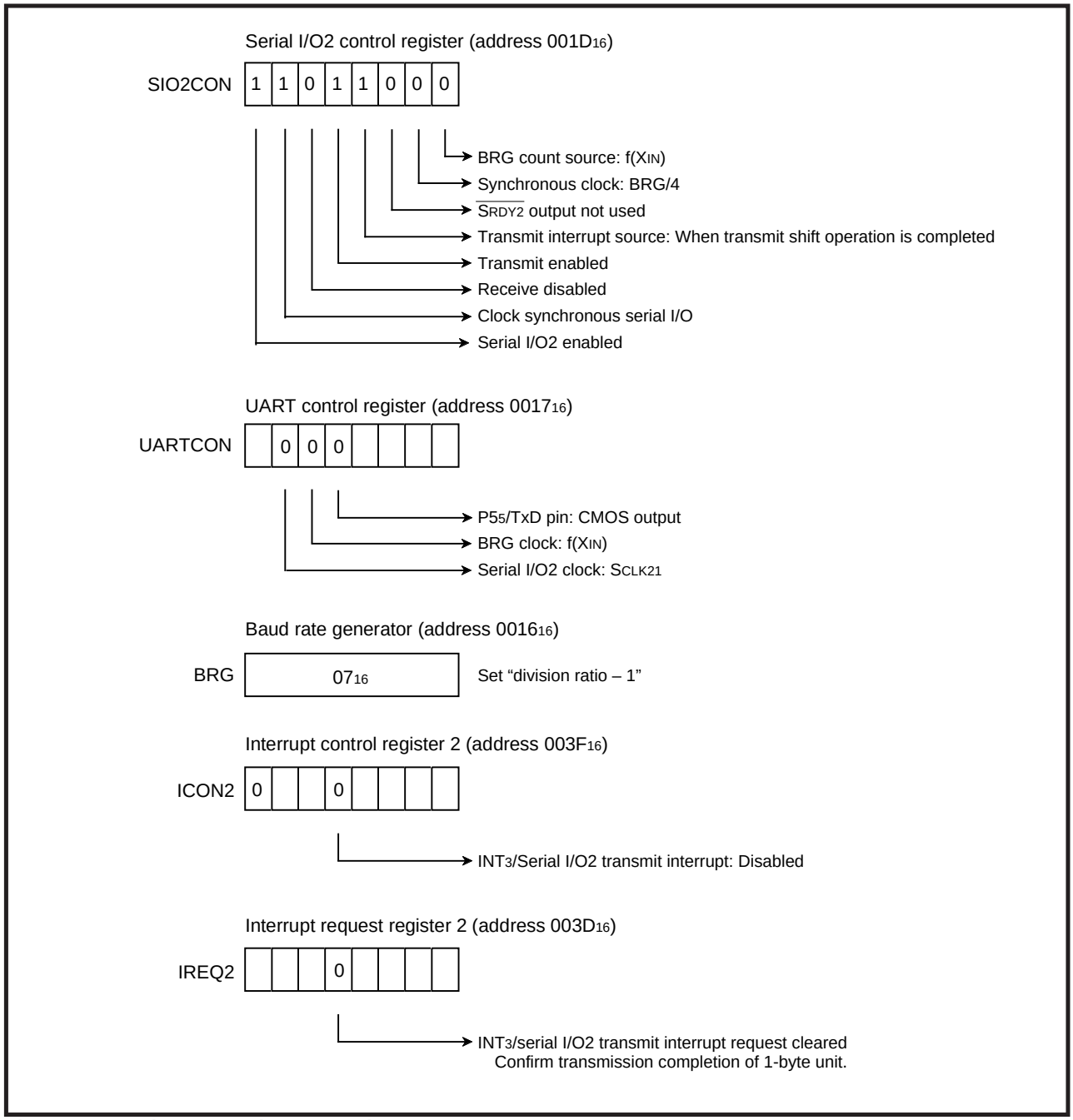


Fig. 2.3.41 Relevant registers setting

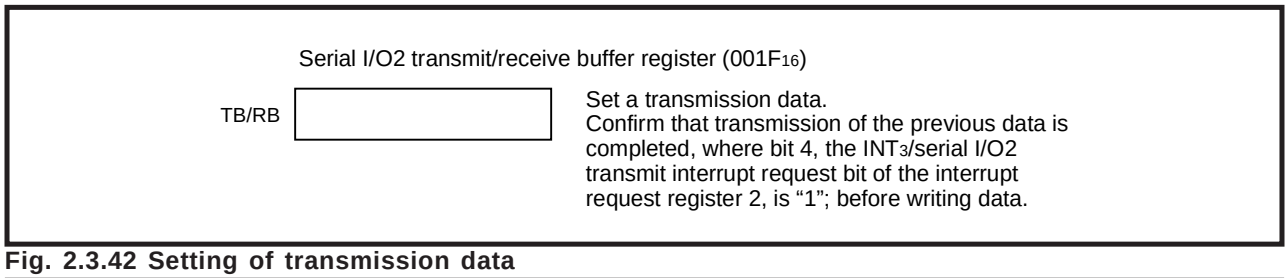


Fig. 2.3.42 Setting of transmission data

Figure 2.3.43 shows a control procedure.

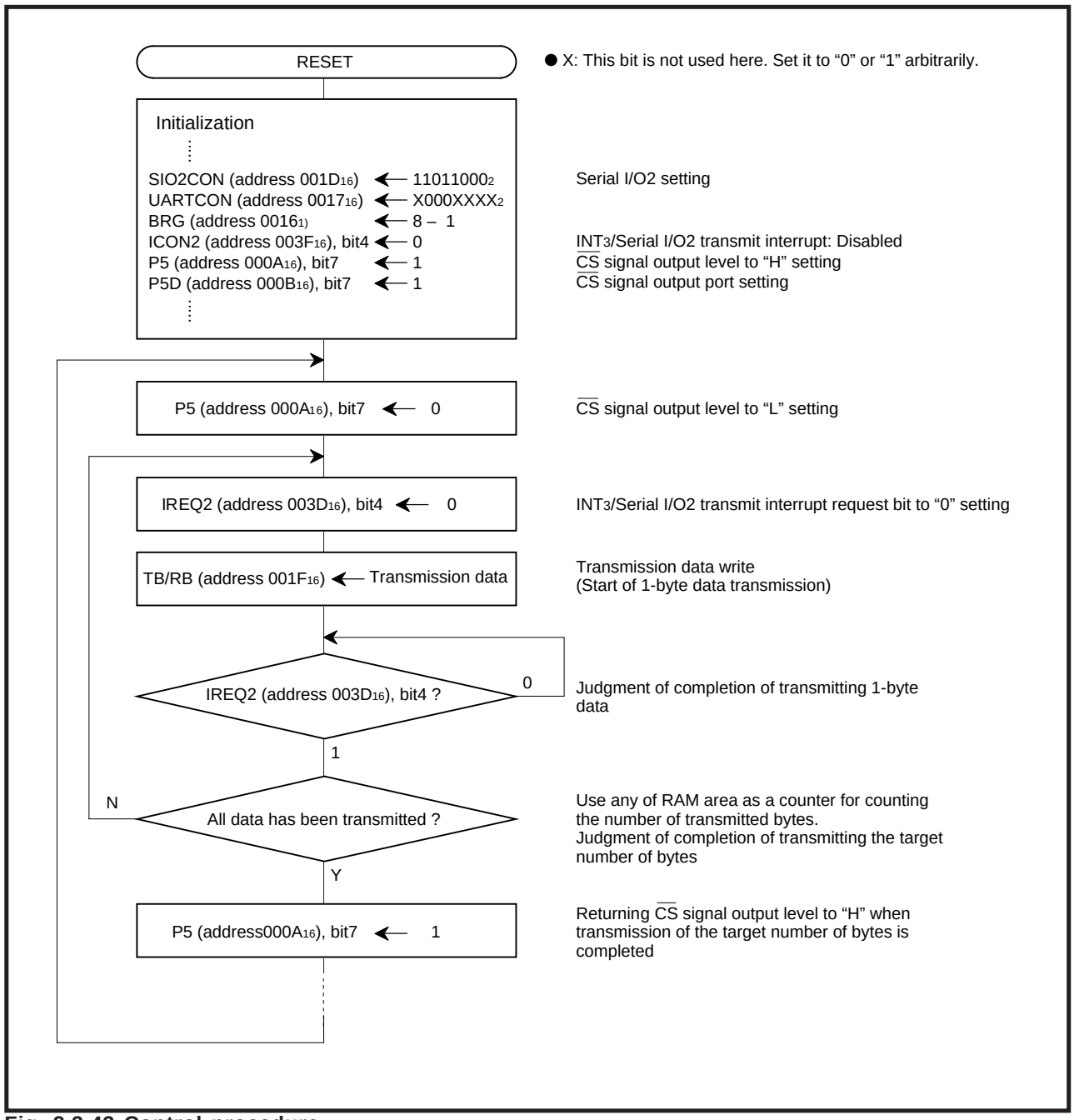


Fig. 2.3.43 Control procedure

(3) Cyclic transmission or reception of block data (data of specified number of bytes) between two microcomputers

Outline : When the clock synchronous serial I/O is used for communication, synchronization of the clock and the data between the transmitting and receiving sides may be lost because of noise included in the synchronous clock. It is necessary to correct that constantly, using "heading adjustment".
This "heading adjustment" is carried out by using the interval between blocks in this example.

Figure 2.3.44 shows a connection diagram.

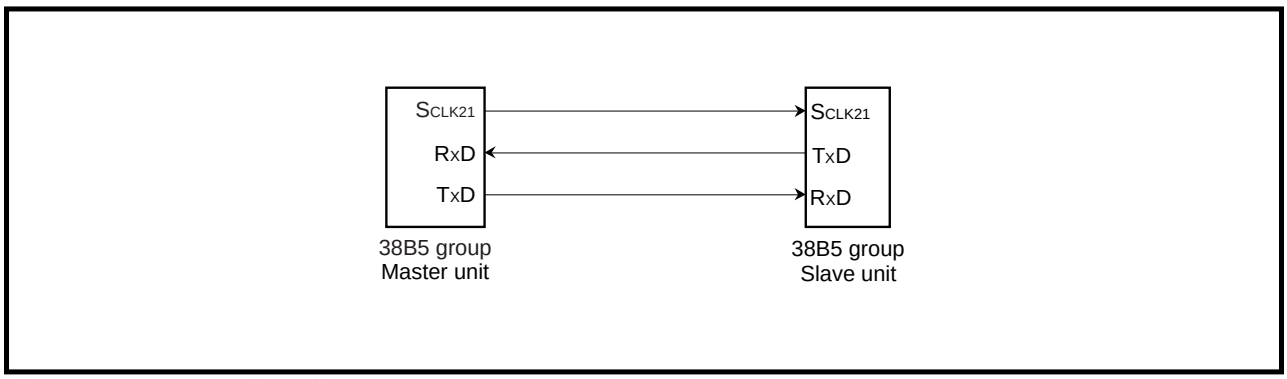


Fig. 2.3.44 Connection diagram

Specifications:
- Use of serial I/O2 in clock synchronous serial I/O
- Synchronous clock frequency : 131 kHz ($f(X_{IN})$ = 4.19 MHz is divided by 32.)
- Byte cycle: 488 µs
- Number of bytes for transmission or reception : 8 bytes/block each
- Block transfer cycle : 16 ms
- Block transfer term : 3.5 ms
- Interval between blocks : 12.5 ms
- Heading adjustment time : 8 ms
- Transfer direction : LSB first

Limitations of the specifications:
- Reading of the reception data and setting of the next transmission data must be completed within the time obtained from "byte cycle – time for transferring 1-byte data" (in this example, the time taken from generating of the serial I/O2 receive interrupt request to input of the next synchronous clock is 431 µs).
- "Heading adjustment time < interval between blocks" must be satisfied.

2.3 Serial I/O

The communication is performed according to the timing shown in Figure 2.3.45. In the slave unit, when a synchronous clock is not input within a certain time (heading adjusment time), the next clock input is processed as the beginning (heading) of a block.
When a clock is input again after one block (8 bytes) is received, the clock is ignored.
Figure 2.3.46 shows the relevant registers setting in the master unit and Figure 2.3.47 shows the relevant registers setting in the slave unit.

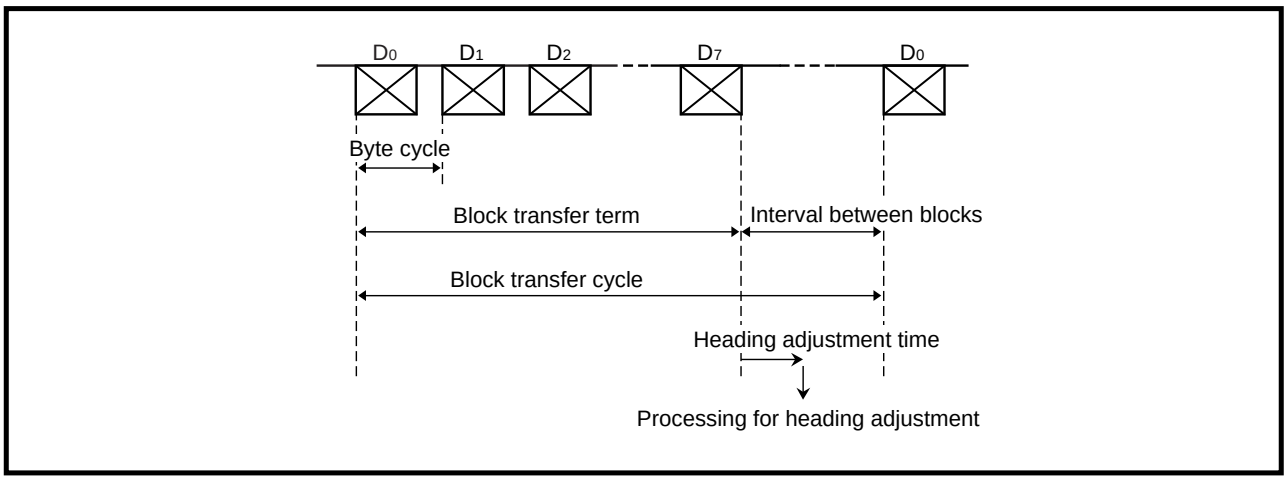


Fig. 2.3.45 Timing chart

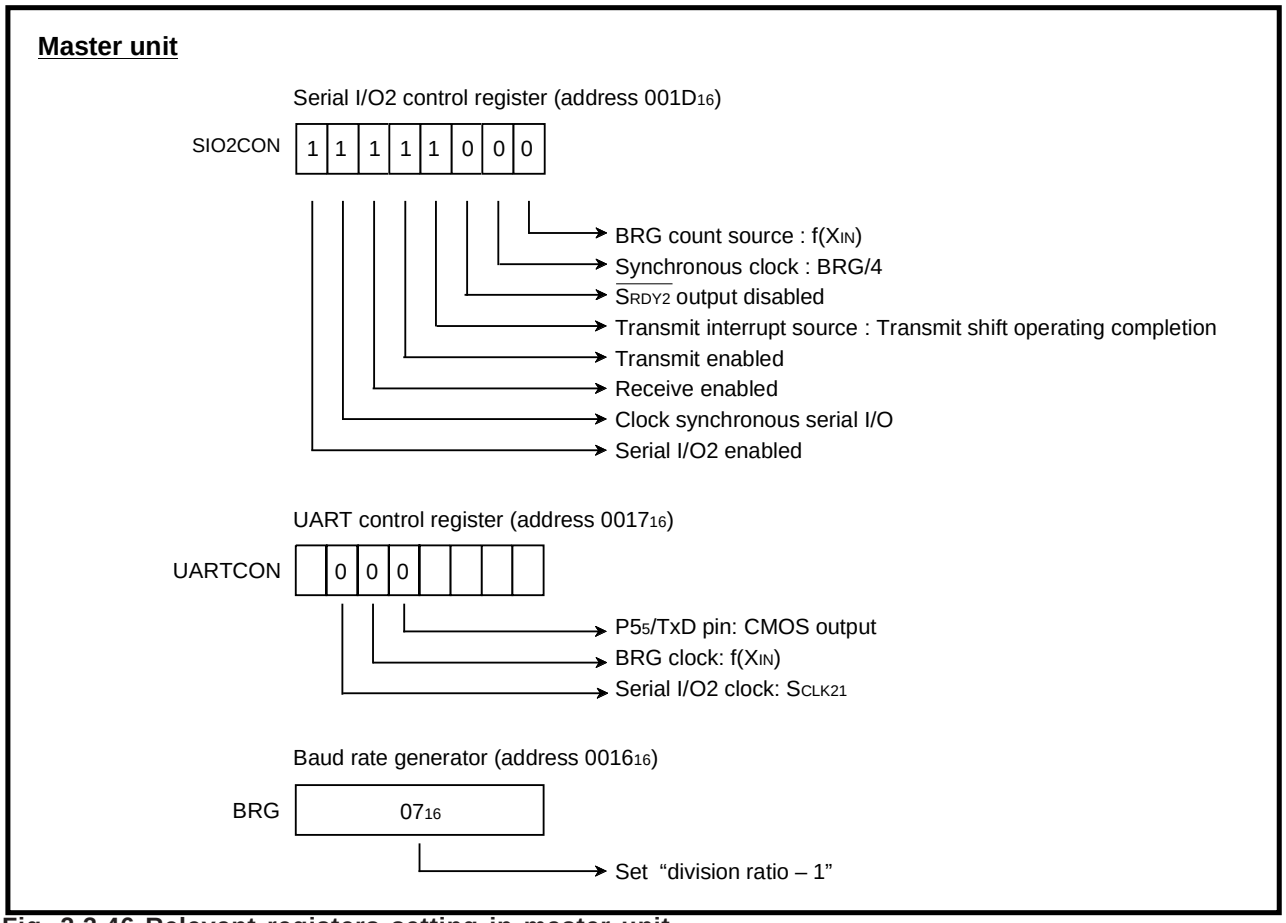


Fig. 2.3.46 Relevant registers setting in master unit

Slave unit

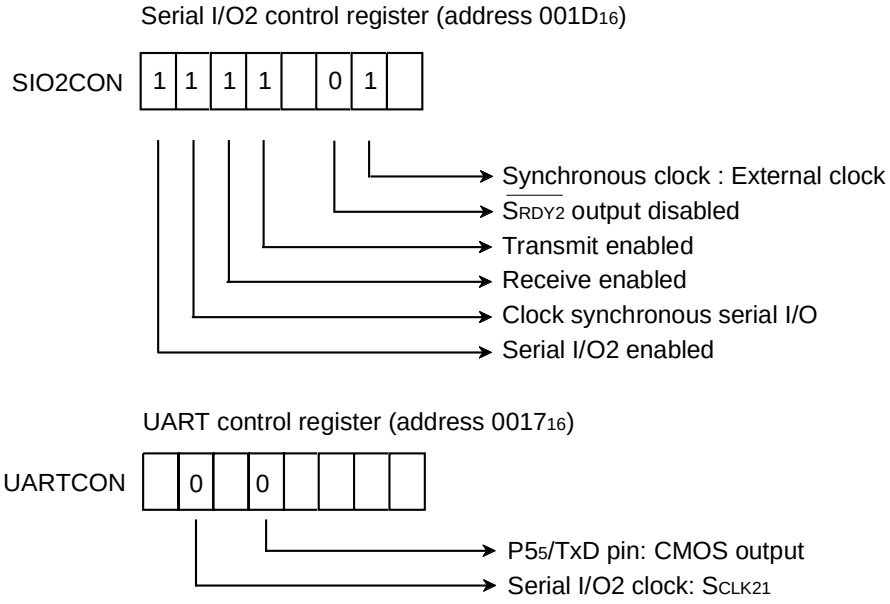


Fig. 2.3.47 Relevant registers setting in slave unit

Control procedure by software:

● Control in the master unit

After setting the relevant registers shown in Figure 2.3.46, the master unit starts transmission or reception of 1-byte data by writing transmission data to the serial I/O2 transmit buffer register. To perform the communication in the timing shown in Figure 2.3.45, take the timing into account and write transmission data. Additionally, read out the reception data when the serial I/O2 transmit interrupt request bit is set to "1," or before the next transmission data is written to the serial I/O2 transmit buffer register.

Figure 2.3.48 shows a control procedure of the master unit using timer interrupts.

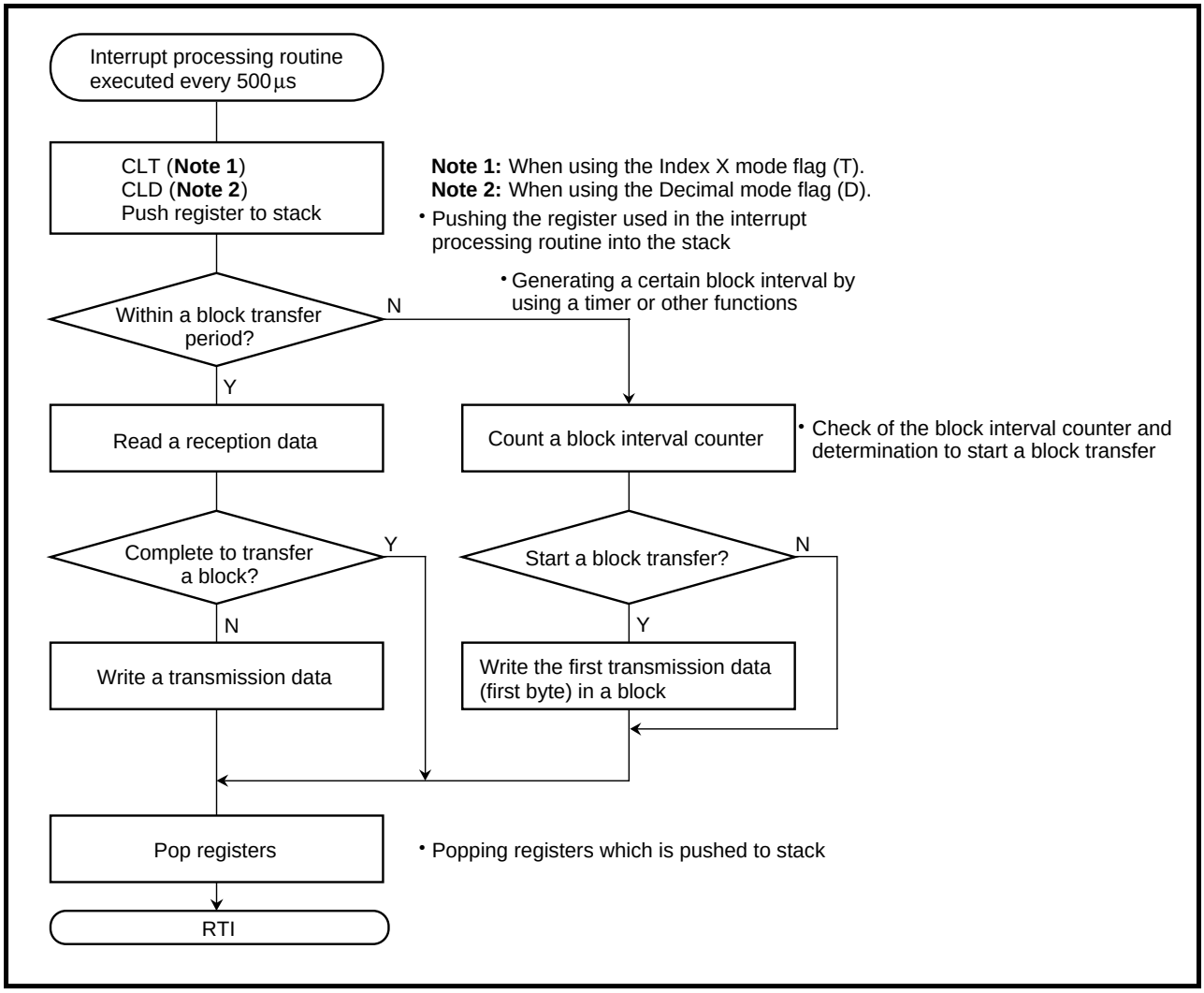


Fig. 2.3.48 Control procedure of master unit

● Control in the slave unit

After setting the relevant registers as shown in Figure 2.3.47, the slave unit becomes the state where a synchronous clock can be received at any time, and the serial I/O2 receive interrupt request bit is set to "1" each time an 8-bit synchronous clock is received.

In the serial I/O2 receive interrupt processing routine, the data to be transmitted next is written to the transmit buffer register after the received data is read out.

However, if no serial I/O2 receive interrupt occurs for a certain time (heading adjustment time or more), the following processing will be performed.

1. The first 1-byte data of the transmission data in the block is written into the transmit buffer register.
2. The data to be received next is processed as the first 1 byte of the received data in the block.

Figure 2.3.49 shows a control procedure of the slave unit using the serial I/O2 receive interrupt and any timer interrupt (for heading adjustment).

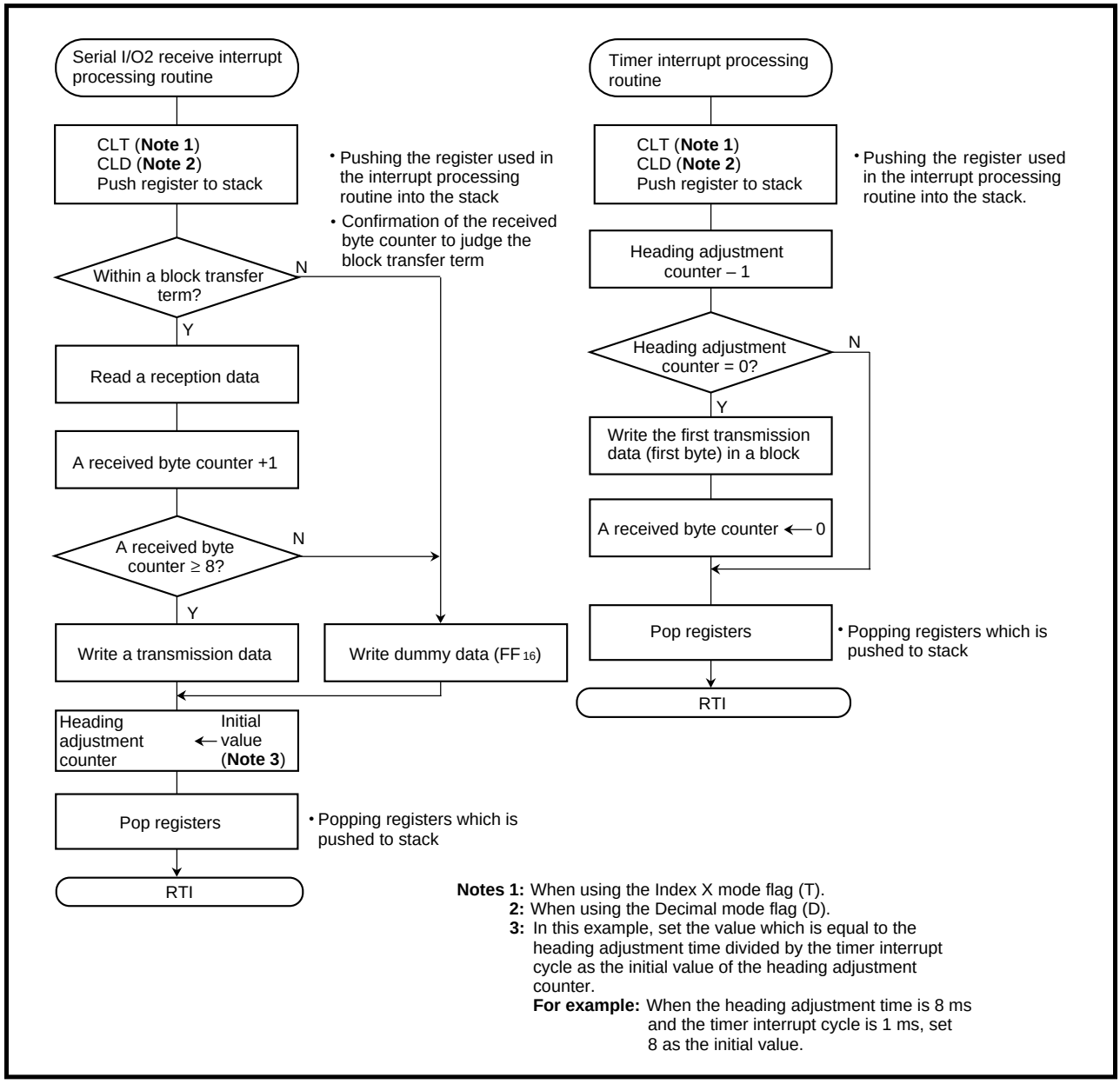


Fig. 2.3.49 Control procedure of slave unit

(4) Communication (transmission/reception) using asynchronous serial I/O (UART)

Outline : 2-byte data is transmitted and received, using the asynchronous serial I/O. Port $P5_6$ is used for communication control.

Figure 2.3.50 shows a connection diagram, and Figure 2.3.51 shows a timing chart.

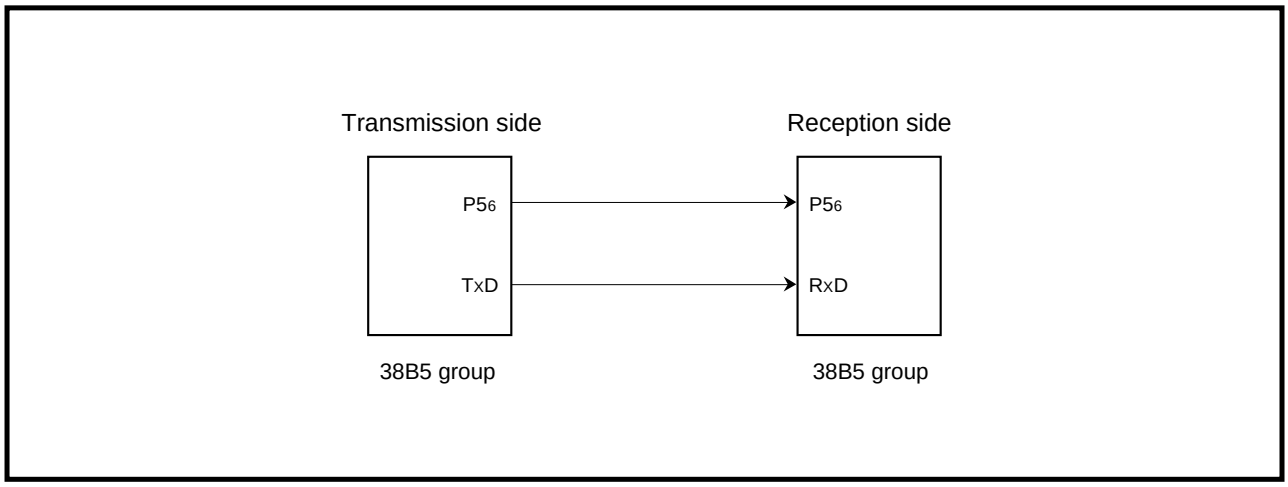


Fig. 2.3.50 Connection diagram

Specifications :
- Use of serial I/O2 in UART
- Transfer bit rate : 9600 bps ($f(X_{IN})$ = 3.6864 MHz is divided by 384)
- Data format : 1ST-8DADA-2ST
- Communication control using port $P5_6$ (The output level of port $P5_6$ is controlled by softoware.)
- 2-byte data is transferred from the transmission side to the receiption side at intervals of 10 ms generated by the timer.

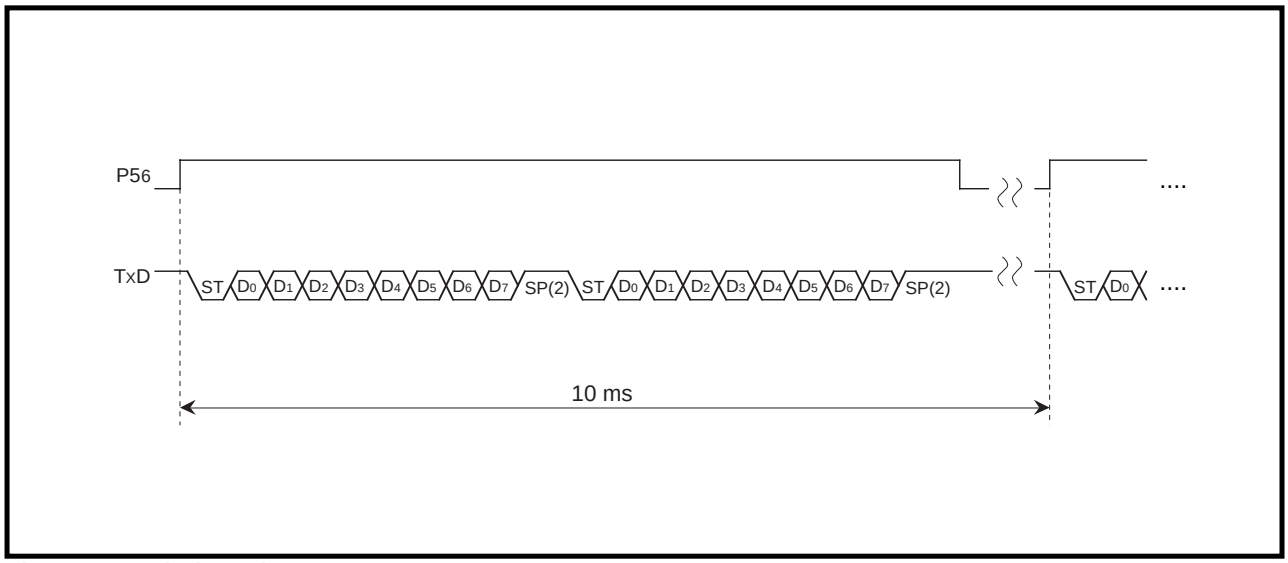


Fig. 2.3.51 Timing chart

Table 2.3.1 shows setting examples of the baud rate generator (BRG) values and transfer bit rate values.

Table 2.3.1 Setting examples of baud rate generator values and transfer bit rate values

Transfer bit rate (**Note 1**)	$f(X_{IN})$ = 3.6864 MHz			$f(X_{IN})$ = 4 MHz		
	BRG count source (**Note 2**)	BRG setting value	Actual rate	BRG count source (**Note 2**)	BRG setting value	Actual rate
600	$f(X_{IN})/4$	$95(5F_{16})$	600.00	$f(X_{IN})/4$	$103(67_{16})$	600.96
1200	$f(X_{IN})/4$	$47(2F_{16})$	1200.00	$f(X_{IN})/4$	$51(33_{16})$	1201.92
2400	$f(X_{IN})/4$	$23(17_{16})$	2400.00	$f(X_{IN})/4$	$25(19_{16})$	2403.85
4800	$f(X_{IN})/4$	$11(0B_{16})$	4800.00	$f(X_{IN})/4$	$12(0C_{16})$	4807.69
9600	$f(X_{IN})/4$	$5(05_{16})$	9600.00	$f(X_{IN})$	$25(19_{16})$	9615.38
19200	$f(X_{IN})/4$	$2(02_{16})$	19200.00	$f(X_{IN})$	$12(0C_{16})$	19230.77
38400	$f(X_{IN})$	$5(05_{16})$	38400.00	$f(X_{IN})$	$5(05_{16})$	41666.67
76800	$f(X_{IN})$	$2(02_{16})$	76800.00	$f(X_{IN})$	$2(02_{16})$	83333.33
31250	—	—	—	$f(X_{IN})$	$7(07_{16})$	31250.00
62500	—	—	—	$f(X_{IN})$	$3(03_{16})$	62500.00

Notes 1: Equation of transfer bit rate:

$$\text{Transfer bit rate (bps)} = \frac{f(X_{IN})}{(\text{BRG setting value} + 1) \times 16 \times m^*}$$

✻m: When bit 0 of the serial I/O2 control register (address $001D_{16}$) is set to "0", a value of m is 1.
When bit 0 of the serial I/O2 control register is set to "1", a value of m is 4.

2: Select the BRG count source with bit 0 of the serial I/O2 control register (address $001D_{16}$).

Figure 2.3.52 shows the registers setting relevant to the transmission side; Figure 2.3.53 shows the registers setting relevant to the reception side.

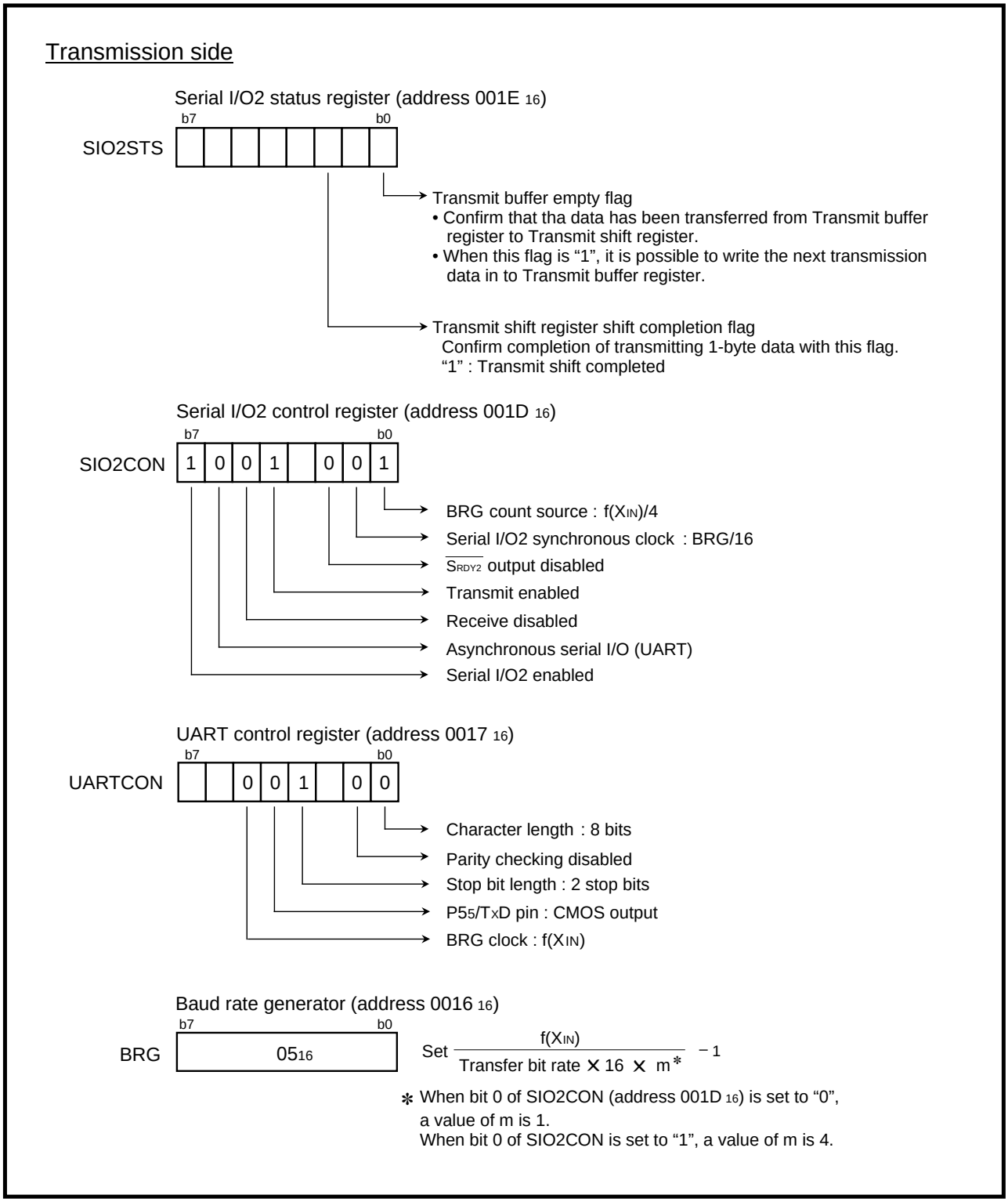


Fig. 2.3.52 Registers setting relevant to transmission side

Reception side

Serial I/O2 status register (address 001E 16)

SIO2STS (b7 … b0)

- Receive buffer full flag
  Confirm completion of receiving 1-byte data with this flag.
  "1" : at completing reception
  "0" : at reading out contents of Receive buffer register
- Overrun error flag
  "1" : When data is ready in Receive shift register while Receive buffer register contains the data.
- Parity error flag
  "1" : When a parity error occurs in enabled parity.
- Framing error flag
  "1" : When stop bits cannot be detected at the specified timing
- Summing error flag
  "1" : when any one of the following errors occurs.
  - Overrun error
  - Parity error
  - Framing error

Serial I/O2 control register (address 001D 16)

SIO2CON (b7 … b0) = 1 0 1 0 _ 0 0 1

- BRG count source : f(XIN)/4
- Serial I/O synchronous clock : BRG/16
- $\overline{SRDY}$ output disabled
- Transmit disabled
- Receive enabled
- Asynchronous serial I/O (UART)
- Serial I/O2 enabled

UART control register (address 0017 16)

UARTCON (b7 … b0) = _ _ 0 _ 1 _ 0 0

- Character length : 8 bits
- Parity checking disabled
- Stop bit length : 2 stop bits
- BRG clock: f(XIN)

Baud rate generator (address 0016 16)

BRG (b7 … b0) = 05_{16}

Set $\frac{f(X_{IN})}{\text{Transfer bit rate} \times 16 \times m^{*}} - 1$

* When bit 0 of SIO2CON (address 001D 16) is set to "0", a value of m is 1.
When bit 0 of SIO2CON is set to "1", a value of m is 4.

Fig. 2.3.53 Registers setting relevant to reception side

2.3 Serial I/O

Figure 2.3.54 shows a control procedure of the transmission side, and Figure 2.3.55 shows a control procedure of the reception side.

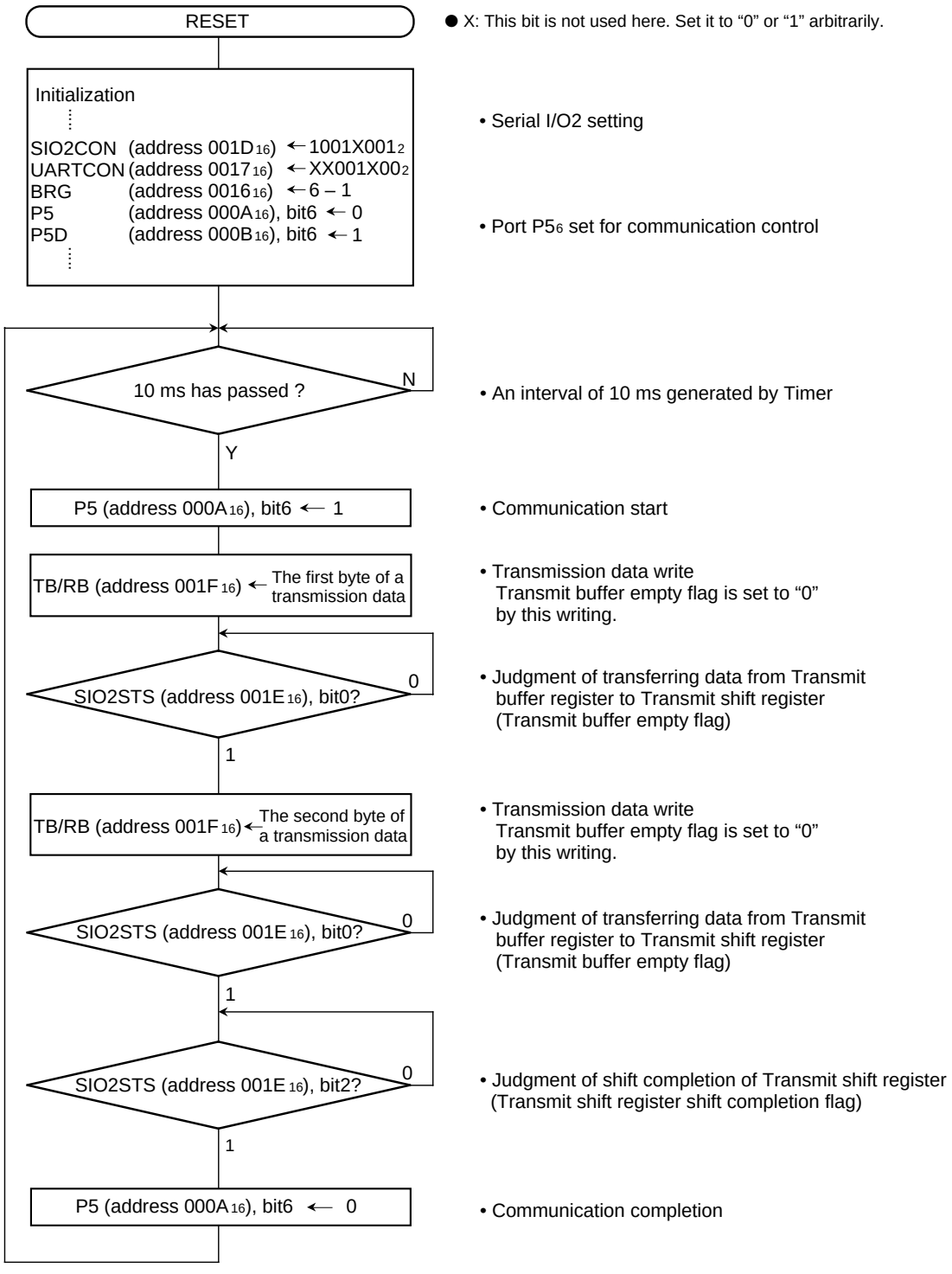


Fig. 2.3.54 Control procedure of transmission side

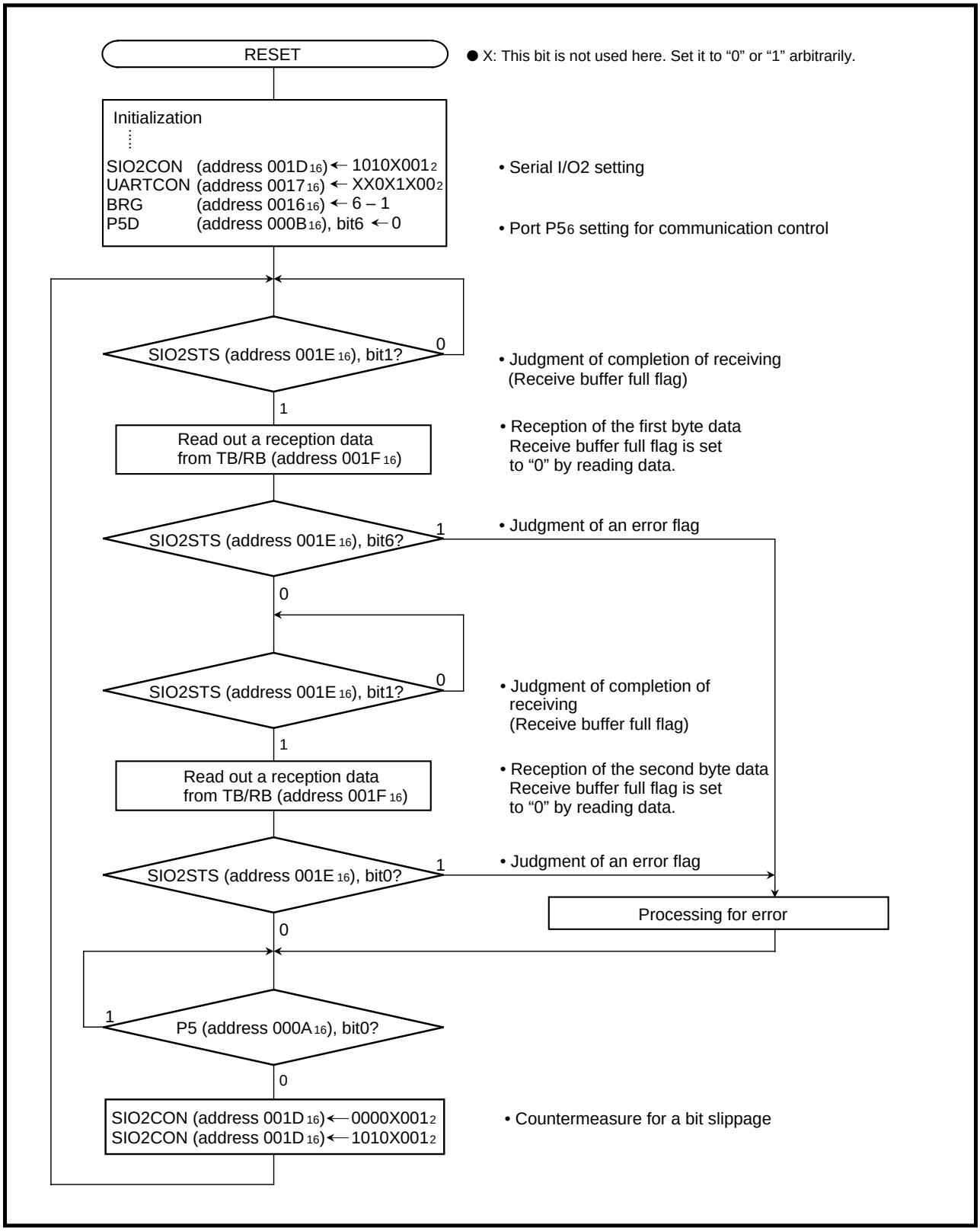


Fig. 2.3.55 Control procedure of reception side

2.3.9 Notes on serial I/O1

(1) Clock

■ **Using internal clock**

After setting the synchronous clock to an internal clock, clear the serial I/O interrupt request bit before perform the normal serial I/O transfer or the serial I/O automatic transfer.

■ **Using external clock**

After inputting "H" level to the external clock input pin, clear the serial I/O interrupt request bit before performing the normal serial I/O transfer or the serial I/O automatic transfer.

(2) Using serial I/O1 interrupt

Clear bit 3 of the interrupt request register 1 to "0" by software.

(3) State of S_{OUT1} pin

The S_{OUT1} pin control bit of the serial I/O1 control register 2 can be used to select the state of the S_{OUT1} pin when serial data is not transferred; either output active or high-impedance. However, when selecting an external synchronous clock; the S_{OUT1} pin can become the high-impedance state by setting the S_{OUT1} pin control bit to "1" when the serial I/O1 clock input is at "H" after transfer completion.

(4) Serial I/O initialization bit

- Set "0" to the serial I/O initialization bit of the serial I/O1 control register 1 when terminating a serial transfer during transferring.
- When writing "1" to the serial I/O initialization bit, the serial I/O1 is enabled, but each register is not initialized. Set the value of each register by program.

(5) Handshake signal

■ **S_{BUSY1} input signal**

Input an "H" level to the S_{BUSY1} input and an "L" level signal to the $\overline{S_{BUSY1}}$ input in the initial state. When the external synchronous clock is selected, switch the input level to the S_{BUSY1} input and the $\overline{S_{BUSY1}}$ input while the serial I/O1 clock input is in "H" state.

■ **S_{RDY1} input•output signal**

When selecting the internal synchronous clock, input an "L" level to the S_{RDY1} input and an "H" level signal to the $\overline{S_{RDY1}}$ input in the initial state.

(6) 8-bit serial I/O mode

■ **When selecting external synchronous clock**

When an external synchronous clock is selected, the contents of the serial I/O1 register are being shifted continually while the transfer clock is input to the serial I/O1 clock pin. In this case, control the clock externally.

(7) In automatic transfer serial I/O mode

■ **Set of automatic transfer interval**

- When the S_{BUSY1} output is used, and the S_{BUSY1} output and the S_{STB1} output function as signals for each transfer data set by the S_{BUSY1} output•S_{STB1} output function selection bit of serial I/O1 control register 2; the transfer interval is inserted before the first data is transmitted/received, and after the last data is transmitted/received. Accordingly, regardless of the contents of the S_{BUSY1} output•S_{STB1} output function selection bit, this transfer interval for each 1-byte data becomes 2 cycles longer than the value set by the automatic transfer interval set bits of serial I/O1 control register 3.

- When using the S_{STB1} output, regardless of the contents of the S_{BUSY1} output•S_{STB1} output function selection bit, this transfer interval for each 1-byte data becomes 2 cycles longer than the value set by the automatic transfer interval set bits of serial I/O1 control register 3.
- When using the combined output of S_{BUSY1} and S_{STB1} as the signal for each of all transfer data set, the transfer interval after completion of transmission/reception of the last data becomes 2 cycles longer than the value set by the automatic transfer interval set bits.
- Set the transfer interval of each 1-byte data transfer to 5 or more cycles of the internal clock ϕ after the rising edge of the last bit of a 1-byte data.
- When selecting an external clock, the set of automatic transfer interval becomes invalid.

■ Set of serial I/O1 transfer counter

- Write the value decreased by 1 from the number of transfer data bytes to the serial I/O1 transfer counter.
- When selecting an external clock, after writing a value to the serial I/O1 register/transfer counter, wait for 5 or more cycles of internal clock ϕ before inputting the transfer clock to the serial I/O1 clock pin.

■ Serial I/O initialization bit

A serial I/O1 automatic transfer interrupt request occurs when “0” is written to the serial I/O initialization bit during an operation. Disable it with the interrupt enable bit as necessary by program.

2.3.10 Notes on serial I/O2

(1) Notes when selecting clock synchronous serial I/O

① **Stop of transmission operation**

As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the transmit enable bit to “0” (transmit disabled).

● **Reason**

Since transmission is not stopped and the transmission circuit is not initialized even if only the serial I/O2 enable bit is cleared to “0” (serial I/O2 disabled), the internal transmission is running (in this case, since pins TxD, RxD, S_{CLK21}, S_{CLK22} and $\overline{S_{RDY2}}$ function as I/O ports, the transmission data is not output). When data is written to the transmit buffer register in this state, data starts to be shifted to the transmit shift register. When the serial I/O2 enable bit is set to “1” at this time, the data during internally shifting is output to the TxD pin and an operation failure occurs.

② **Stop of receive operation**

As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the receive enable bit to “0” (receive disabled), or clear the serial I/O2 enable bit to “0” (serial I/O2 disabled).

③ **Stop of transmit/receive operation**

As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, simultaneously clear both the transmit enable bit and receive enable bit to “0” (transmit and receive disabled).

(when data is transmitted and received in the clock synchronous serial I/O mode, any one of data transmission and reception cannot be stopped.)

● **Reason**

In the clock synchronous serial I/O mode, the same clock is used for transmission and reception. If any one of transmission and reception is disabled, a bit error occurs because transmission and reception cannot be synchronized.

In this mode, the clock circuit of the transmission circuit also operates for data reception. Accordingly, the transmission circuit does not stop by clearing only the transmit enable bit to “0” (transmit disabled). Also, the transmission circuit is not initialized by clearing the serial I/O2 enable bit to “0” (serial I/O2 disabled) (refer to (1), ①).

(2) Notes when selecting clock asynchronous serial I/O

① **Stop of transmission operation**
As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the transmit enable bit to "0" (transmit disabled).

● **Reason**
Since transmission is not stopped and the transmission circuit is not initialized even if only the serial I/O2 enable bit is cleared to "0" (serial I/O2 disabled), the internal transmission is running (in this case, since pins TxD, RxD, S_{CLK21}, S_{CLK22} and $\overline{S_{RDY2}}$ function as I/O ports, the transmission data is not output). When data is written to the transmit buffer register in this state, data starts to be shifted to the transmit shift register. When the serial I/O2 enable bit is set to "1" at this time, the data during internally shifting is output to the TxD pin and an operation failure occurs.

② **Stop of receive operation**
As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the receive enable bit to "0" (receive disabled).

③ **Stop of transmit/receive operation**
Only transmission operation is stopped.
As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the transmit enable bit to "0" (transmit disabled).

● **Reason**
Since transmission is not stopped and the transmission circuit is not initialized even if only the serial I/O2 enable bit is cleared to "0" (serial I/O2 disabled), the internal transmission is running (in this case, since pins TxD, RxD, S_{CLK21}, S_{CLK22} and $\overline{S_{RDY2}}$ function as I/O ports, the transmission data is not output). When data is written to the transmit buffer register in this state, data starts to be shifted to the transmit shift register. When the serial I/O2 enable bit is set to "1" at this time, the data during internally shifting is output to the TxD pin and an operation failure occurs.

Only receive operation is stopped.
As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the receive enable bit to "0" (receive disabled).

(3) $\overline{S_{RDY2}}$ output of reception side
When signals are output from the $\overline{S_{RDY2}}$ pin on the reception side by using an external clock in the clock synchronous serial I/O mode, set all of the receive enable bit, the $\overline{S_{RDY2}}$ output enable bit, and the transmit enable bit to "1" (transmit enabled).

(4) Setting serial I/O2 control register again
Set the serial I/O2 control register again after the transmission and the reception circuits are reset by clearing both the transmit enable bit and the receive enable bit to "0."

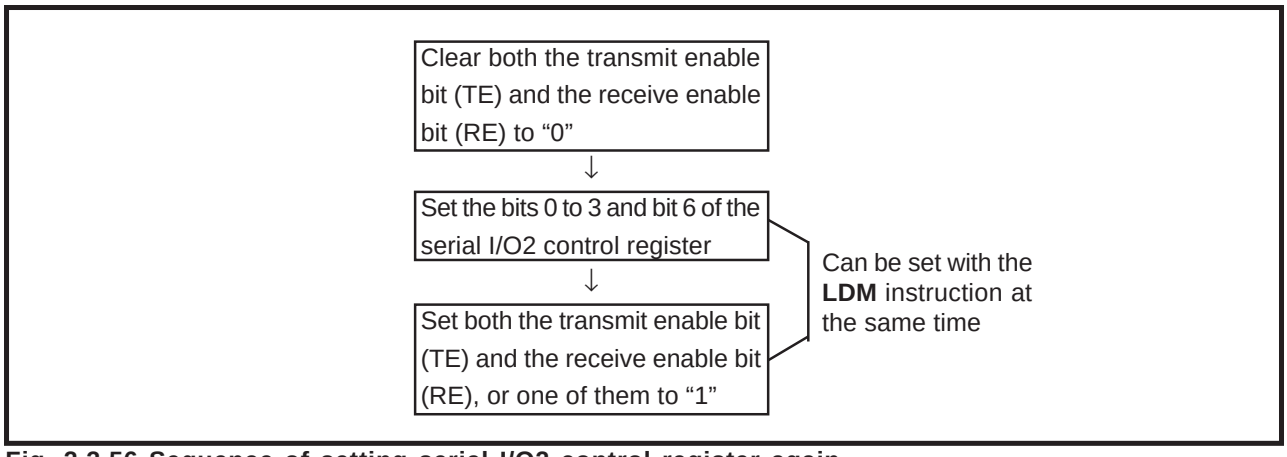


Fig. 2.3.56 Sequence of setting serial I/O2 control register again

(5) Data transmission control with referring to transmit shift register completion flag

The transmit shift register completion flag changes from "1" to "0" with a delay of 0.5 to 1.5 shift clocks. When data transmission is controlled with referring to the flag after writing the data to the transmit buffer register, note the delay.

(6) Transmission control when external clock is selected

When an external clock is used as the synchronous clock for data transmission, set the transmit enable bit to "1" at "H" of the serial I/O2 clock input level. Also, write the transmit data to the transmit buffer register (serial I/O shift register) at "H" of the serial I/O2 clock input level.

(7) Transmit interrupt request when transmit enable bit is set

The transmission interrupt request bit is set and the interruption request is generated even when selecting timing that either of the following flags is set to "1" as timing where the transmission interruption is generated.

- Transmit buffer empty flag is set to "1"
- Transmit shift register completion flag is set to "1"

Therefore, when the transmit interrupt is used, set the transmit interrupt enable bit to transmit enabled as the following sequence.

① Transmit enable bit is set to "1"
② Transmit interrupt request bit is set to "0"

● **Reason**

When the transmission enable bit is set to "1", the transmit buffer empty flag and transmit shift register completion flag are set to "1".

(8) Using TxD pin

The $P5_5$/TxD P-channel output disable bit of UART control register is valid in both cases: using as a normal I/O port and as the TxD pin. Do not supply Vcc + 0.3 V or more even when using the $P5_5$/TxD pin as an N-channel open-drain output.

Additionally, in the serial I/O2, the TxD pin latches the last bit and continues to output it after completing transmission.

2.4 FLD controller

This paragraph describes the setting method of FLD controller relevant registers, notes etc.

2.4.1 Memory assignment

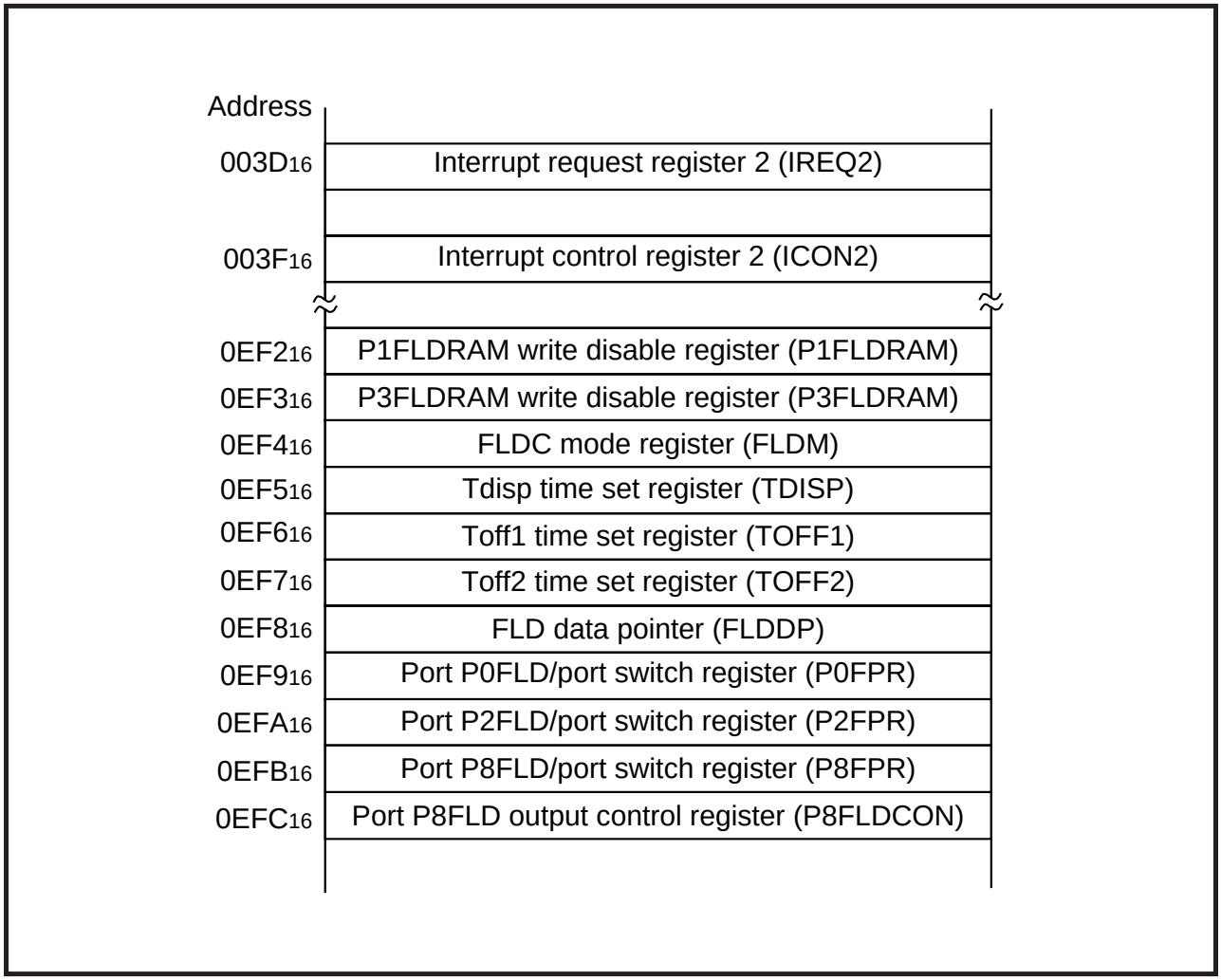
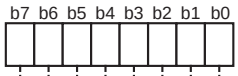


Fig. 2.4.1 Memory assignment of FLD controller relevant registers

2.4.2 Relevant registers

P1FLDRAM write disable register

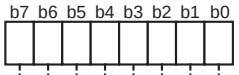


P1FLDRAM write disable register
(P1FLDRAM: address $0EF2_{16}$)

b	Name	Functions	At reset	R	W
0	FLDRAM corresponding to port $P1_0$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
1	FLDRAM corresponding to port $P1_1$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
2	FLDRAM corresponding to port $P1_2$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
3	FLDRAM corresponding to port $P1_3$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
4	FLDRAM corresponding to port $P1_4$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
5	FLDRAM corresponding to port $P1_5$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
6	FLDRAM corresponding to port $P1_6$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
7	FLDRAM corresponding to port $P1_7$ write disable bit	0: Operating normally 1: Write disabled	0	○	○

Fig. 2.4.2 Structure of P1FLDRAM write disable register

P3FLDRAM write disable register



P3FLDRAM write disable register
(P3FLDRAM: address $0EF3_{16}$)

b	Name	Functions	At reset	R	W
0	FLDRAM corresponding to port $P3_0$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
1	FLDRAM corresponding to port $P3_1$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
2	FLDRAM corresponding to port $P3_2$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
3	FLDRAM corresponding to port $P3_3$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
4	FLDRAM corresponding to port $P3_4$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
5	FLDRAM corresponding to port $P3_5$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
6	FLDRAM corresponding to port $P3_6$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
7	FLDRAM corresponding to port $P3_7$ write disable bit	0: Operating normally 1: Write disabled	0	○	○

Fig. 2.4.3 Structure of P3FLDRAM write disable register

FLDC mode register

b7 b6 b5 b4 b3 b2 b1 b0

FLDC mode register
(FLDM: address 0EF4$_{16}$)

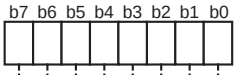
b	Name	Functions	At reset	R	W
0	Automatic display control bit (P0, P1, P2, P3, P8)	0 : General-purpose mode 1 : Automatic display mode	0	○	○
1	Display start bit	0 : Display stopped 1 : Display in progress (display starts by writing "1")	0	○	○
2	Tscan control bits	b3 b2 0 0 : 0 FLD digit interrupt (at rising edge of each digit) 0 1 : 1 × Tdisp 1 0 : 2 × Tdisp 1 1 : 3 × Tdisp FLD blanking interrupt (at falling edge of last digit)	0	○	○
3			0	○	○
4	Timing number control bit	0 : 16 timing mode 1 : 32 timing mode **(Note 2)**	0	○	○
5	Gradation display mode selection control bit	0 : Not selected 1 : Selected **(Notes 1, 2)**	0	○	○
6	Tdisp counter count source selection bit	0 : f(XIN)/16 or f(XCIN)/32 1 : f(XIN)/64 or f(XCIN)/128	0	○	○
7	High-breakdown voltage port drivability selection bit	0 : Drivability strong 1 : Drivability weak	0	○	○

Notes 1: When the gradation display mode is selected, the number of timing is max. 16 timing. (Set "0" to the timing number control bit (b4).)

2: When switching the timing number control bit (b4) or the gradation display mode selection control bit (b5), set "0" to the display start bit (b1) (display stop state) before that.

Fig. 2.4.4 Structure of FLD mode register

Tdisp time set register



Tdisp time set register
(TDISP: address 0EF5$_{16}$)

b	Functions	At reset	R	W
0	•Set the Tdisp time. •When a value n is written to this register, Tdisp time is expressed as Tdisp = (n + 1) × count source. •When reading this register, the value in the counter is read out. (Example) When the following condition is satisfied, Tdisp becomes 804 µs {(200 + 1) × 4 µs}; •f(X$_{IN}$) = 4 MHz •bit 6 of FLDC mode register = 0 (f(X$_{IN}$)/16 is selected as Tdisp counter count source.) •Tdisp time set register = 200 (C8$_{16}$).	0	○	○
1		0	○	○
2		0	○	○
3		0	○	○
4		0	○	○
5		0	○	○
6		0	○	○
7		0	○	○

Fig. 2.4.5 Structure of Tdisp time set register

Toff1 time set register

b7 b6 b5 b4 b3 b2 b1 b0

Toff1 time set register
(TOFF1: address $0EF6_{16}$)

b	Functions	At reset	R	W
0	•Set the Toff1 time. •When a value n1 is written to this register, Toff1 time is expressed as Toff1 = n1 × count source. (Example) When the following condition is satisfied, Toff1 becomes 120 µs (= 30 × 4 µs); •f(X_{IN}) = 4 MHz •bit 6 of FLDC mode register = 0 (f(X_{IN})/16 is selected as Tdisp counter count source.) •Toff1 time set register = 30 ($1E_{16}$).	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Note: Set value of 03_{16} or more.

Fig. 2.4.6 Structure of Toff1 time set register

Toff2 time set register

b7 b6 b5 b4 b3 b2 b1 b0

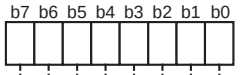
Toff2 time set register
(TOFF2: address $0EF7_{16}$)

b	Functions	At reset	R	W
0	•Set the Toff2 time. •When a value n2 is written to this register, Toff2 time is expressed as Toff2 = n2 × count source. However, setting of Toff2 time is valid only for the FLD port which is satisfied the following; •gradation display mode •value of FLD automatic display RAM (in gradation display mode) = "1" (dark display). (Example) When the following condition is satisfied, Toff2 becomes 720 µs (= 180 × 4 µs); •f(X_{IN}) = 4 MHz •bit 6 of FLDC mode register = 0 (f(X_{IN})/16 is selected as Tdisp counter count source.) •Toff2 time set register = 180 ($B4_{16}$).	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Note: When the Toff2 control bit (b7) of the port P8FLD output control register (address $0EFC_{16}$) is set to "1", set value of 03_{16} or more to the Toff2 control register.

Fig. 2.4.7 Structure of Toff2 time set register

FLD data pointer/FLD data pointer reload register

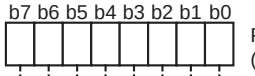


FLD data pointer/FLD data pointer reload register
(FLDDP: address 0EF8$_{16}$)

b	Functions	At reset	R	W
0	The start address of each data of FLD ports P0, P1, P2, P3, and P8, which is transferred from FLD automatic display RAM, is set to this register. The start address becomes the address adding the value set to this register into the last data address of each FLD port. Set a value of (timing number – 1) to this register. The value which is set to this address is written to the FLD data pointer reload register. When reading data from this address, the value in the FLD data pointer is read. When bits 5 to 7 of this register is read, “0” is always read.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 2.4.8 Structure of FLD data pointer/FLD data pointer reload register

Port P0FLD/port switch register



Port P0FLD/port switch register
(P0FPR: address 0EF9$_{16}$)

b	Name	Functions	At reset	R	W
0	Port P0$_0$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
1	Port P0$_1$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
2	Port P0$_2$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
3	Port P0$_3$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
4	Port P0$_4$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
5	Port P0$_5$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
6	Port P0$_6$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
7	Port P0$_7$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○

Fig. 2.4.9 Structure of port P0FLD/port switch register

Port P2FLD/port switch register

b7 b6 b5 b4 b3 b2 b1 b0

Port P2FLD/port switch register
(P2FPR: address 0EFA16)

b	Name	Functions	At reset	R	W
0	Port P20FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
1	Port P21FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
2	Port P22FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
3	Port P23FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
4	Port P24FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
5	Port P25FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
6	Port P26FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
7	Port P27FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○

Fig. 2.4.10 Structure of port P2FLD/port switch register

Port P8FLD/port switch register

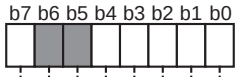
b7 b6 b5 b4 b3 b2 b1 b0

Port P8FLD/port switch register
(P8FPR: address 0EFB16)

b	Name	Functions	At reset	R	W
0	Port P80FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
1	Port P81FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
2	Port P82FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
3	Port P83FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
4	Port P84FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
5	Port P85FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
6	Port P86FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
7	Port P87FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○

Fig. 2.4.11 Structure of port P8FLD/port switch register

Port P8FLD output control register



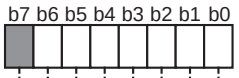
Port P8FLD output control register
(P8FLDCON : address 0EFC16)

b	Name	Functions	At reset	R	W
0	P84–P87 FLD output reverse bit	0 : Output normally 1 : Reverse output	0	○	○
1	P84–P87/FLDRAM write disable bit	0 : Operating normally 1 : Write disabled	0	○	○
2	P84–P87 Toff invalid bit	0 : Operating normally 1 : Toff invalid	0	○	○
3	P84–P87 delay control bit **(Note)**	0 : No delay 1 : Delay	0	○	○
4	P63/AN9 dimmer output control bit	0 : Ordinary port 1 : Dimmer output	0	○	○
5	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	○	×
6			0	○	×
7	Toff2 control bit	0 : Operating normally (falling operation) 1 : Rising operation	0	○	○

Note: Valid only when selecting FLD port and P84–P87 Toff invalid function

Fig. 2.4.12 Structure of port P8FLD output control register

Interrupt request register 2



Interrupt request register 2
(IREQ2 : address 3D16)

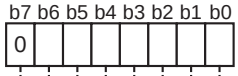
b	Name	Functions	At reset	R	W
0	Timer 4 interrupt request bit **(Note)**	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Timer 5 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	Timer 6 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O2 receive interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	INT3/serial I/O2 transmit interrupt request bit **(Note)**	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	INT4 interrupt request bit A-D converter interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	FLD blanking interrupt request bit FLD digit interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×

✻: “0” can be set by software, but “1” cannot be set.

Note: In the mask option type P, if timer 4 interrupt whose count source is CNTR1 input and INT3 interrupt are selected, these bits do not become “1”.

Fig. 2.4.13 Structure of interrupt request register 2

Interrupt control register 2



Interrupt control register 2
(ICON2 : address $3F_{16}$)

b	Name	Functions	At reset	R	W
0	Timer 4 interrupt enable bit **(Note)**	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer 5 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer 6 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O2 receive interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
4	INT_3/serial I/O2 transmit interrupt enable bit **(Note)**	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
5	INT_4 interrupt enable bit A-D converter interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
6	FLD blanking interrupt enable bit FLD digit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
7	Fix “0” to this bit.		0	○	○

Note: In the mask option type P, timer 4 interrupt whose count source is $CNTR_1$ input and INT_3 interrupt are not available.

Fig. 2.4.14 Structure of interrupt control register 2

2.4.3 FLD controller application examples

(1) Key-scan using FLD automatic display and segments

Outline: Key read-in with segment pins is performed by software using the FLD automatic display mode.

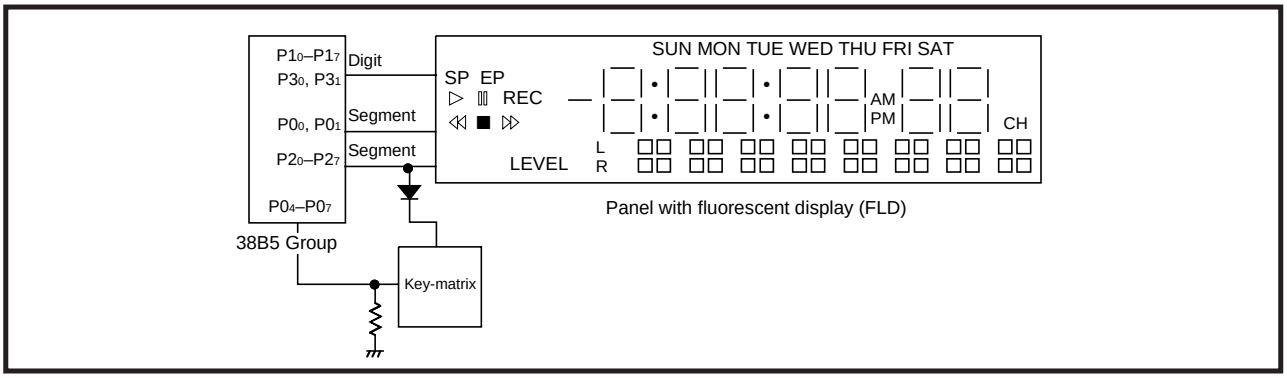


Fig. 2.4.15 Connection diagram

Specifications: •Use of total 20 FLD ports (10 digits; 10 segments (8 key-scan included))
•Use of FLD automatic display mode
•Display in gradation display mode and 16 timing mode
•Toff1 = 40 µs, Toff2 = 64 µs, Tdisp = 204 µs, Tscan = 3 × Tdisp = 720 µs, f(X_{IN}) = 4 MHz
•Use of FLD blanking interrupt

Figure 2.4.16 shows the timing chart of key-scan, and Figure 2.4.17 shows the enlarged view of Tscan. After switching the segment pin to an output port, generate the waveform shown Figure 2.4.17 by software and perform key-scan.

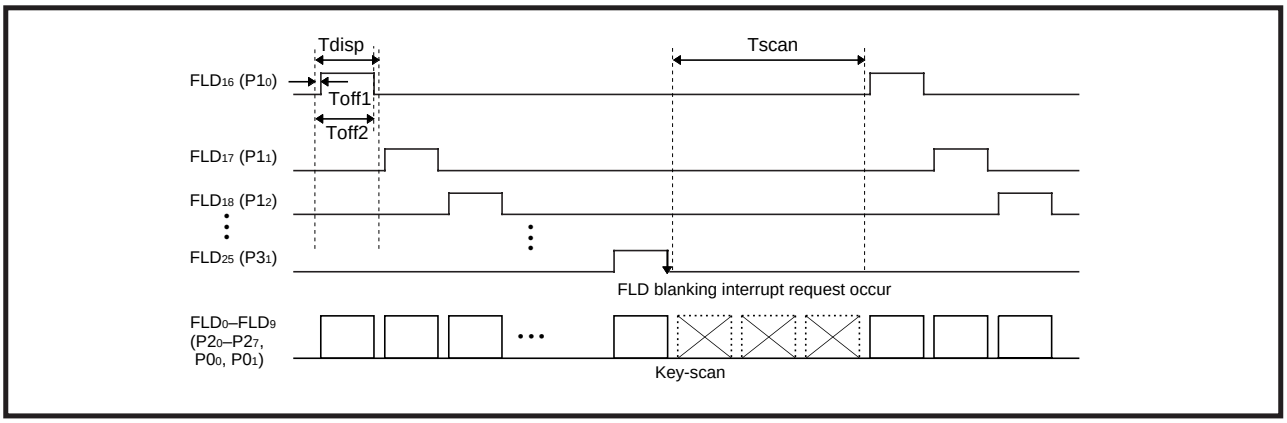


Fig. 2.4.16 Timing chart of key-scan using FLD automatic display mode and segments

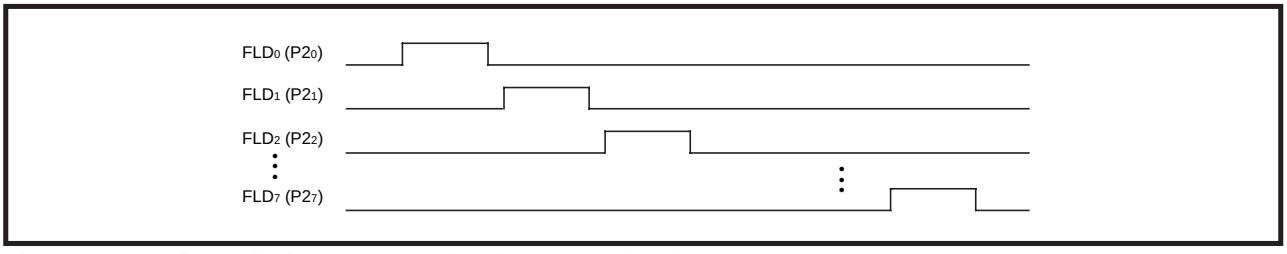


Fig. 2.4.17 Enlarged view of FLD₀ (P2₀) to FLD₇ (P2₇) Tscan

2.4 FLD controller

Figure 2.4.18 shows the setting of relevant registers.

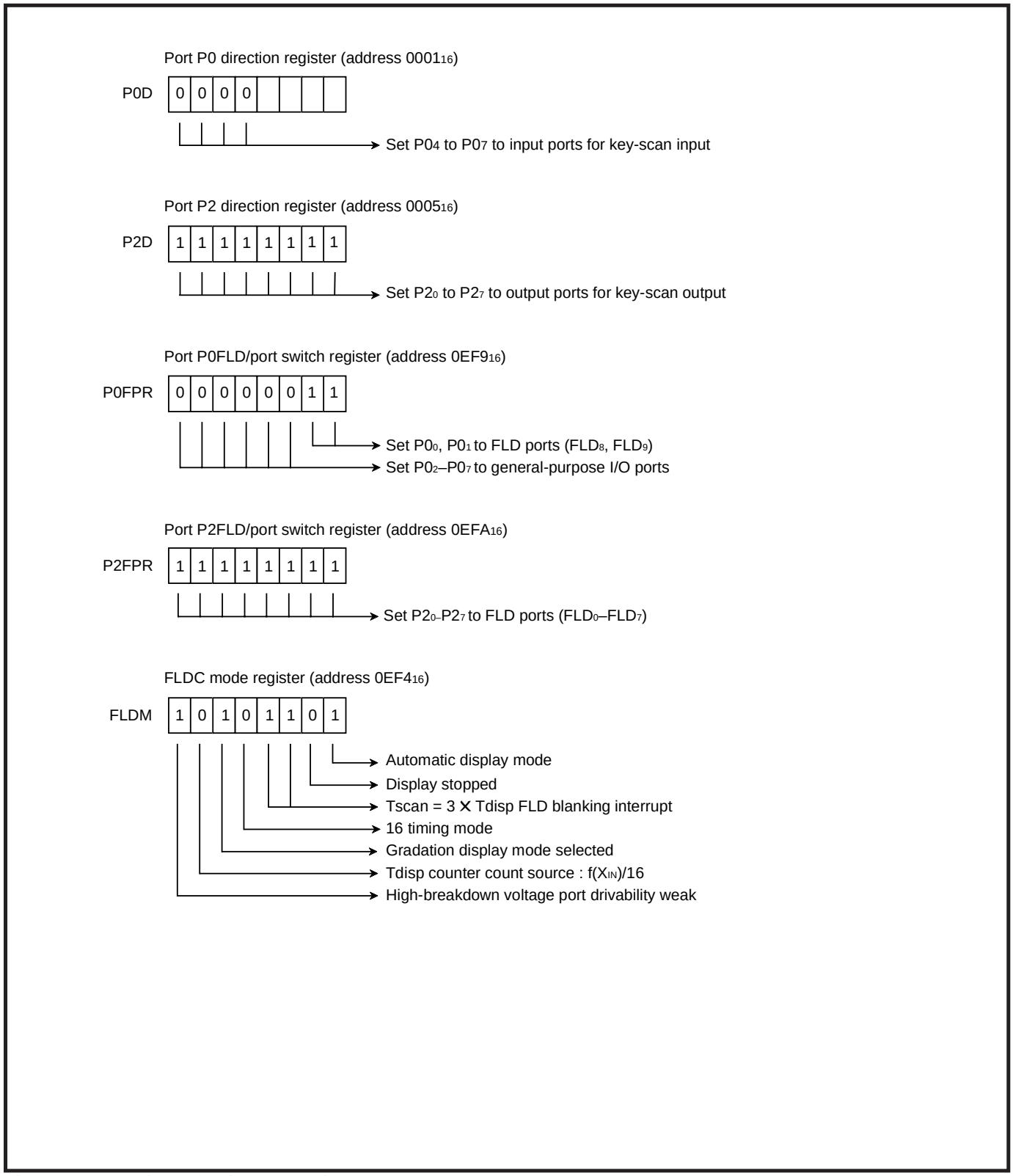


Fig. 2.4.18 Setting of relevant registers

Tdisp time set register (address $0EF5_{16}$)

TDISP | 32_{16}

50 (32_{16}) set; (50 + 1) × count source = 204 µs
Count source = $f(X_{IN})/16$ = 4 µs, at $f(X_{IN})$ = 4 MHz

Toff1 time set register (address $0EF6_{16}$)

TOFF1 | A_{16}

10 (A_{16}) set; 10 × count source = 40 µs
Count source = $f(X_{IN})/16$ = 4 µs, at $f(X_{IN})$ = 4 MHz

Toff2 time set register (address $0EF7_{16}$)

TOFF2 | 10_{16}

16 (10_{16}) set; 16 × count source = 64 µs
Count source = $f(X_{IN})/16$ = 4 µs, at $f(X_{IN})$ = 4 MHz

Note: Perform this setting when the gradation display mode is selected.

FLD data pointer (address $0EF8_{16}$)

FLDDP | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |

→ Set {(digit number) – 1} = 9

P1FLDRAM write disable register (address $0EF2_{16}$)

P1FLDRAM | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

→ Disable writing to FLDRAM corresponding to $P1_0$ to $P1_7$

P3FLDRAM write disable register (address $0EF3_{16}$)

P3FLDRAM | | | | | | | 1 | 1 |

→ Disable writing to FLDRAM corresponding to $P3_0$, $P3_1$

Interrupt request register 2 (address $003D_{16}$)

IREQ2 | | 0 | | | | | | |

→ Clear FLD blanking interrupt request bit

Interrupt control register 2 (address $003F_{16}$)

ICON2 | 0 | 1 | | | | | | |

→ FLD blanking interrupt: Enabled

FLDC mode register (address $0EF4_{16}$)

FLDM | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |

→ Display start

Setting of FLD automatic display RAM:

Table 2.4.1 FLD automatic display RAM map

1 to 16 timing display data stored area

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0FB0$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB1$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB2$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB3$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB4$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB5$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB6$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB7$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB8$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB9$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FBA$_{16}$								
0FBB$_{16}$								
0FBC$_{16}$								
0FBD$_{16}$								
0FBE$_{16}$								
0FBF$_{16}$								
0FC0$_{16}$							FLD$_9$	FLD$_8$
0FC1$_{16}$							FLD$_9$	FLD$_8$
0FC2$_{16}$							FLD$_9$	FLD$_8$
0FC3$_{16}$							FLD$_9$	FLD$_8$
0FC4$_{16}$							FLD$_9$	FLD$_8$
0FC5$_{16}$							FLD$_9$	FLD$_8$
0FC6$_{16}$							FLD$_9$	FLD$_8$
0FC7$_{16}$							FLD$_9$	FLD$_8$
0FC8$_{16}$							FLD$_9$	FLD$_8$
0FC9$_{16}$							FLD$_9$	FLD$_8$
0FCA$_{16}$								
0FCB$_{16}$								
0FCC$_{16}$								
0FCD$_{16}$								
0FCE$_{16}$								
0FCF$_{16}$								
0FD0$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD1$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD2$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD3$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD4$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD5$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD6$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD7$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD8$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD9$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FDA$_{16}$								
0FDB$_{16}$								
0FDC$_{16}$								
0FDD$_{16}$								
0FDE$_{16}$								
0FDF$_{16}$								
0FE0$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE1$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE2$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE3$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE4$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE5$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE6$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE7$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE8$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE9$_{16}$							FLD$_{25}$	FLD$_{24}$

Gradation display control data stored area

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Corresponding digit pin
0F60$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$	→ FLD$_{25}$ (P3$_1$)
0F61$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$	→ FLD$_{24}$ (P3$_0$)
0F62$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$	→ FLD$_{23}$ (P1$_7$)
0F63$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$	→ FLD$_{22}$ (P1$_6$)
0F64$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$	→ FLD$_{21}$ (P1$_5$)
0F65$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$	→ FLD$_{20}$ (P1$_4$)
0F66$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$	→ FLD$_{19}$ (P1$_3$)
0F67$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$	→ FLD$_{18}$ (P1$_2$)
0F68$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$	→ FLD$_{17}$ (P1$_1$)
0F69$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$	→ FLD$_{16}$ (P1$_0$)
0F6A$_{16}$									
0F6B$_{16}$									
0F6C$_{16}$									
0F6D$_{16}$									
0F6E$_{16}$									
0F6F$_{16}$									
0F70$_{16}$							FLD$_9$	FLD$_8$	→ FLD$_{25}$ (P3$_1$)
0F71$_{16}$							FLD$_9$	FLD$_8$	→ FLD$_{24}$ (P3$_0$)
0F72$_{16}$							FLD$_9$	FLD$_8$	→ FLD$_{23}$ (P1$_7$)
0F73$_{16}$							FLD$_9$	FLD$_8$	→ FLD$_{22}$ (P1$_6$)
0F74$_{16}$							FLD$_9$	FLD$_8$	→ FLD$_{21}$ (P1$_5$)
0F75$_{16}$							FLD$_9$	FLD$_8$	→ FLD$_{20}$ (P1$_4$)
0F76$_{16}$							FLD$_9$	FLD$_8$	→ FLD$_{19}$ (P1$_3$)
0F77$_{16}$							FLD$_9$	FLD$_8$	→ FLD$_{18}$ (P1$_2$)
0F78$_{16}$							FLD$_9$	FLD$_8$	→ FLD$_{17}$ (P1$_1$)
0F79$_{16}$							FLD$_9$	FLD$_8$	→ FLD$_{16}$ (P1$_0$)
0F7A$_{16}$									
0F7B$_{16}$									
0F7C$_{16}$									
0F7D$_{16}$									
0F7E$_{16}$									
0F7F$_{16}$									
0F80$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0F81$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0F82$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0F83$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0F84$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0F85$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0F86$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0F87$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0F88$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0F89$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0F8A$_{16}$									
0F8B$_{16}$									
0F8C$_{16}$									
0F8D$_{16}$									
0F8E$_{16}$									
0F8F$_{16}$									
0F90$_{16}$							FLD$_{25}$	FLD$_{24}$	
0F91$_{16}$							FLD$_{25}$	FLD$_{24}$	
0F92$_{16}$							FLD$_{25}$	FLD$_{24}$	
0F93$_{16}$							FLD$_{25}$	FLD$_{24}$	
0F94$_{16}$							FLD$_{25}$	FLD$_{24}$	
0F95$_{16}$							FLD$_{25}$	FLD$_{24}$	
0F96$_{16}$							FLD$_{25}$	FLD$_{24}$	
0F97$_{16}$							FLD$_{25}$	FLD$_{24}$	
0F98$_{16}$							FLD$_{25}$	FLD$_{24}$	
0F99$_{16}$							FLD$_{25}$	FLD$_{24}$	

: Area which is used to set segment data

: Area which is used to set digit data

: Area which is available as ordinary RAM

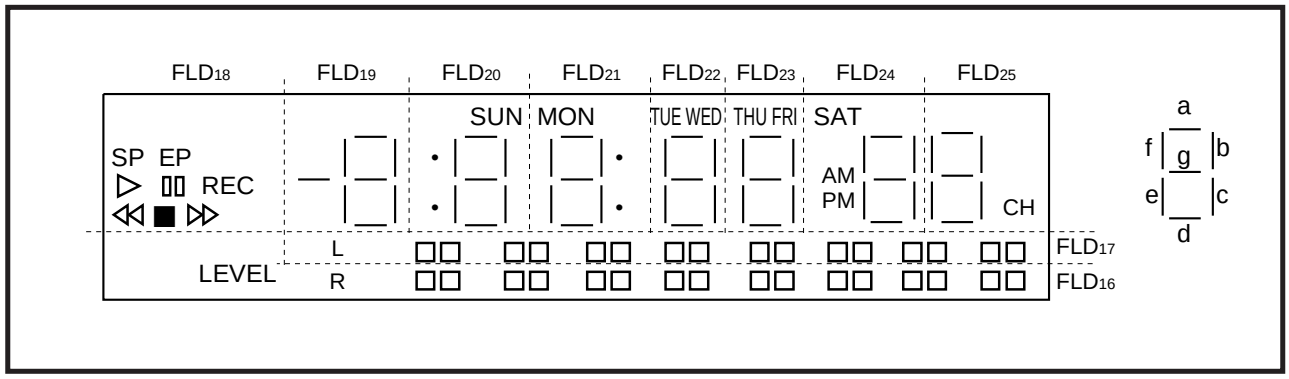


Fig. 2.4.19 FLD digit allocation example

Table 2.4.2 FLD automatic display RAM map example

1 to 16 timing display data stored area

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0FB0$_{16}$	CH	g	f	e	d	c	b	a
0FB1$_{16}$	SAT	g	f	e	d	c	b	a
0FB2$_{16}$	FRI	g	f	e	d	c	b	a
0FB3$_{16}$	WED	g	f	e	d	c	b	a
0FB4$_{16}$	MON	g	f	e	d	c	b	a
0FB5$_{16}$	SUN	g	f	e	d	c	b	a
0FB6$_{16}$	–	g	f	e	d	c	b	a
0FB7$_{16}$	■	◁◁	▷▷	▯▯	▷	REC	SP	EP
0FB8$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□
0FB9$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□
0FBA$_{16}$								
0FBB$_{16}$								
0FBC$_{16}$								
0FBD$_{16}$								
0FBE$_{16}$								
0FBF$_{16}$								
0FC0$_{16}$								
0FC1$_{16}$							PM	AM
0FC2$_{16}$								THU
0FC3$_{16}$								TUE
0FC4$_{16}$								:
0FC5$_{16}$								:
0FC6$_{16}$								
0FC7$_{16}$								
0FC8$_{16}$							L	
0FC9$_{16}$							R	LEVEL

Gradation display control data stored area

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Corresponding digit pin
0F60$_{16}$	CH	g	f	e	d	c	b	a	→ FLD$_{25}$ (P3$_1$)
0F61$_{16}$	SAT	g	f	e	d	c	b	a	→ FLD$_{24}$ (P3$_0$)
0F62$_{16}$	FRI	g	f	e	d	c	b	a	→ FLD$_{23}$ (P1$_7$)
0F63$_{16}$	WED	g	f	e	d	c	b	a	→ FLD$_{22}$ (P1$_6$)
0F64$_{16}$	MON	g	f	e	d	c	b	a	→ FLD$_{21}$ (P1$_5$)
0F65$_{16}$	SUN	g	f	e	d	c	b	a	→ FLD$_{20}$ (P1$_4$)
0F66$_{16}$	–	g	f	e	d	c	b	a	→ FLD$_{19}$ (P1$_3$)
0F67$_{16}$	■	◁◁	▷▷	▯▯	▷	REC	SP	EP	→ FLD$_{18}$ (P1$_2$)
0F68$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□	→ FLD$_{17}$ (P1$_1$)
0F69$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□	→ FLD$_{16}$ (P1$_0$)
0F6A$_{16}$									
0F6B$_{16}$									
0F6C$_{16}$									
0F6D$_{16}$									
0F6E$_{16}$									
0F6F$_{16}$									
0F70$_{16}$									→ FLD$_{25}$ (P3$_1$)
0F71$_{16}$							PM	AM	→ FLD$_{24}$ (P3$_0$)
0F72$_{16}$								THU	→ FLD$_{23}$ (P1$_7$)
0F73$_{16}$								TUE	→ FLD$_{22}$ (P1$_6$)
0F74$_{16}$								:	→ FLD$_{21}$ (P1$_5$)
0F75$_{16}$								:	→ FLD$_{20}$ (P1$_4$)
0F76$_{16}$									→ FLD$_{19}$ (P1$_3$)
0F77$_{16}$									→ FLD$_{18}$ (P1$_2$)
0F78$_{16}$							L		→ FLD$_{17}$ (P1$_1$)
0F79$_{16}$							R	LEVEL	→ FLD$_{16}$ (P1$_0$)

: Unused

Control procedure:

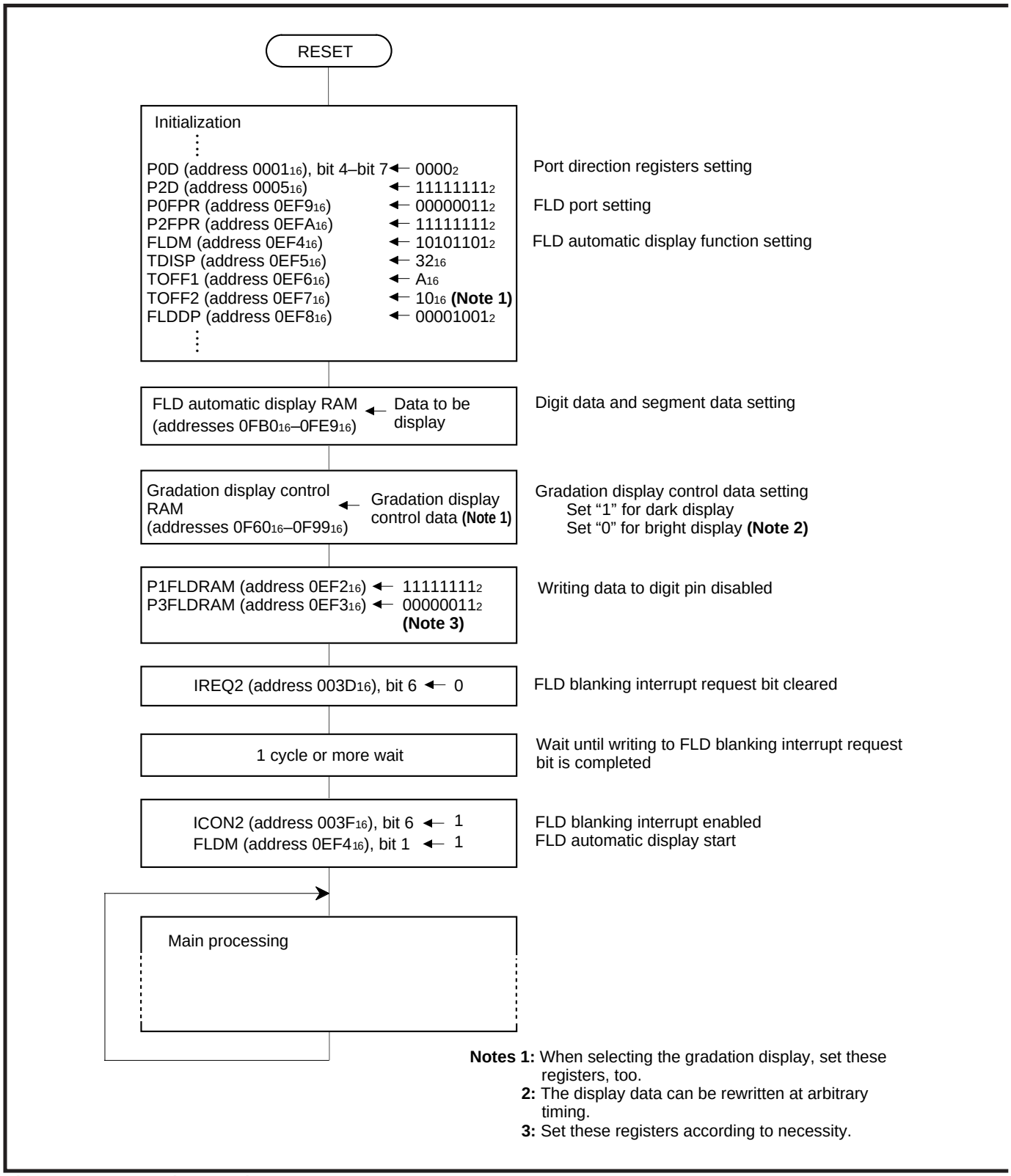
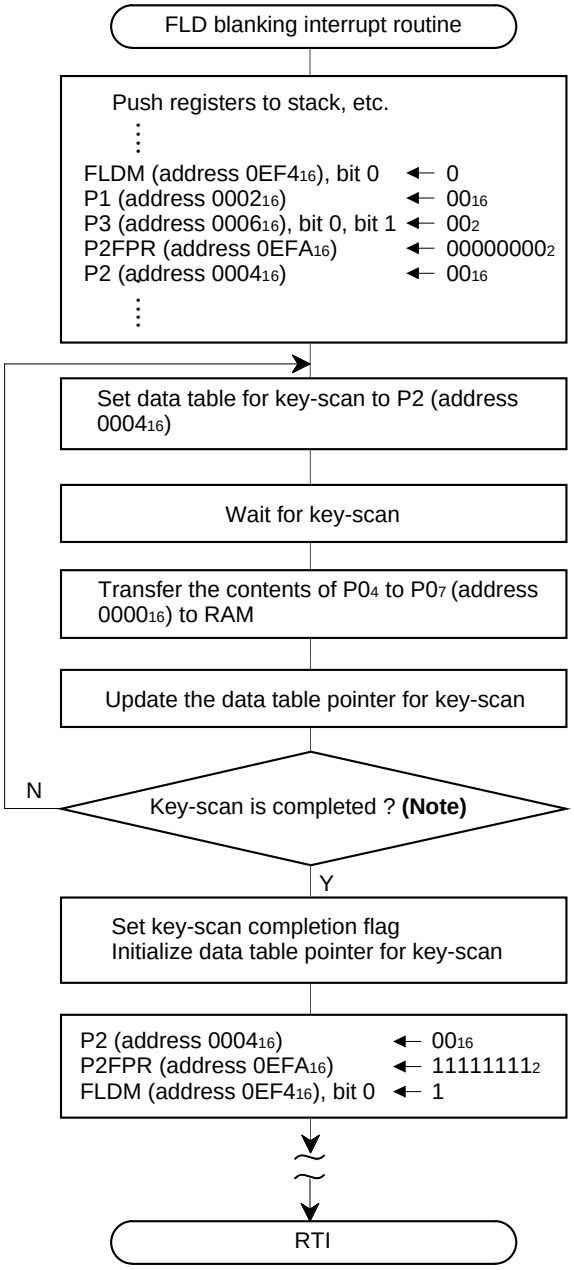


Fig. 2.4.20 Control procedure



Segment key-scan

Switching from automatic display mode to general-purpose mode
Setting of "L" level to port corresponding to digit

Setting of port to be used for key-scan to general-purpose port
Output of "L" level from all ports for key-scan

Wait until "H" level output of P2 is stabilized

Keys read-in
(Set the port P0 direction register (P0D) (address 0001₁₆) to the input mode in the initialization, etc.)

Data table reference pointer for the next key-scan updated

Setting of flag which judge whether key-scan is completed or not

Output of "L" level from all key-scan ports
Setting of general-purpose ports to FLD ports
Switching from general-purpose mode to the automatic display mode

Note: If key-scan is not completed within Tscan set time, perform key-scan separately.

(2) Key-scan using FLD automatic display and digits

Outline: Key read-in with digit output waveforms is performed by software using the FLD automatic display mode.

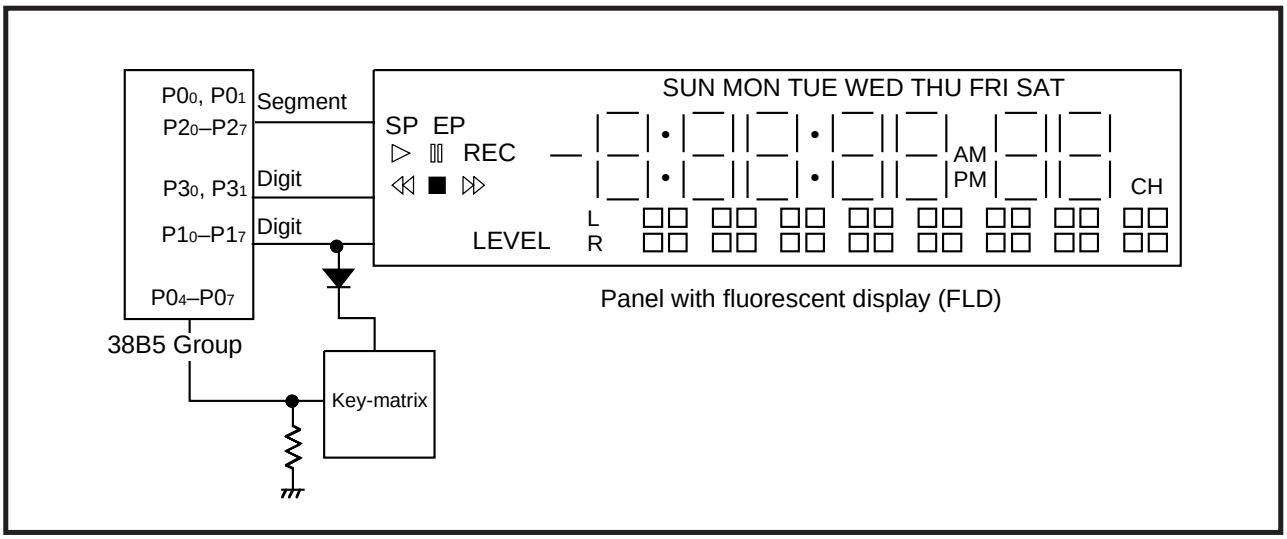


Fig. 2.4.21 Connection diagram

Specifications:
- Use of total 20 FLD ports (10 digits, 8 key-scan included; 10 segments)
- Use of FLD automatic display mode
- Display in gradation display mode and 16 timing mode
- Toff1 = 40 ms, Toff2 = 64 ms, Tdisp = 204 ms, Tscan = 0 ms, f(X_{IN}) = 4 MHz
- Use of FLD digit interrupt

Figure 2.4.22 shows the timing chart of key-scan.

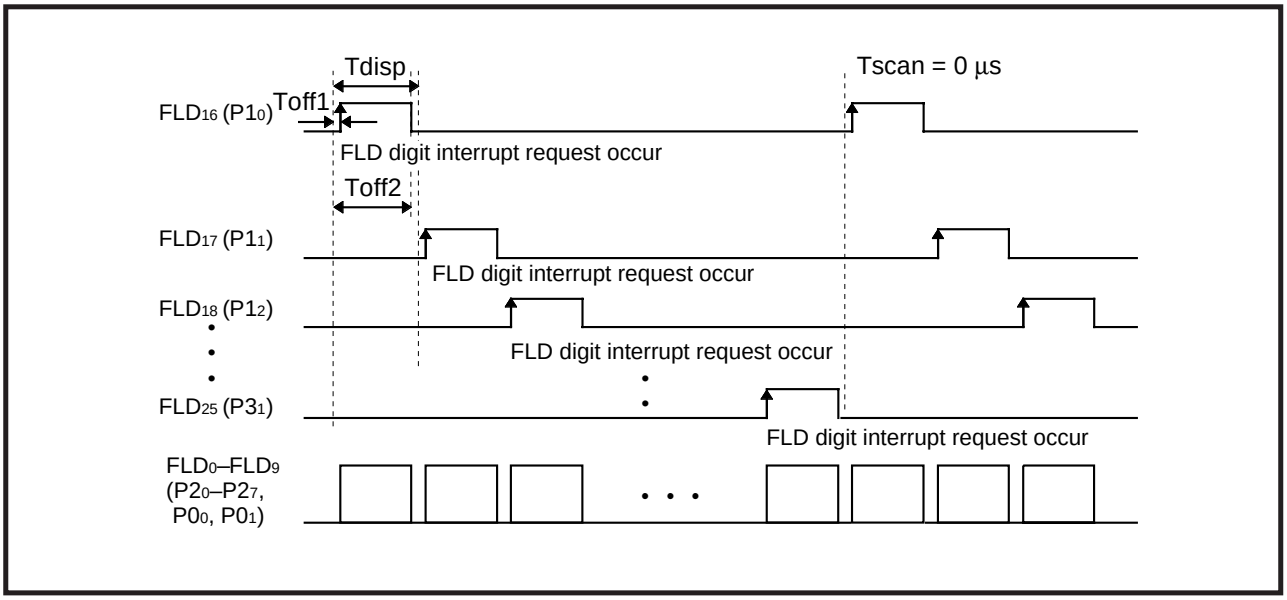


Fig. 2.4.22 Timing chart of key-scan using FLD automatic display mode and digits

2.4 FLD controller

Figure 2.4.23 shows the setting of relevant registers.

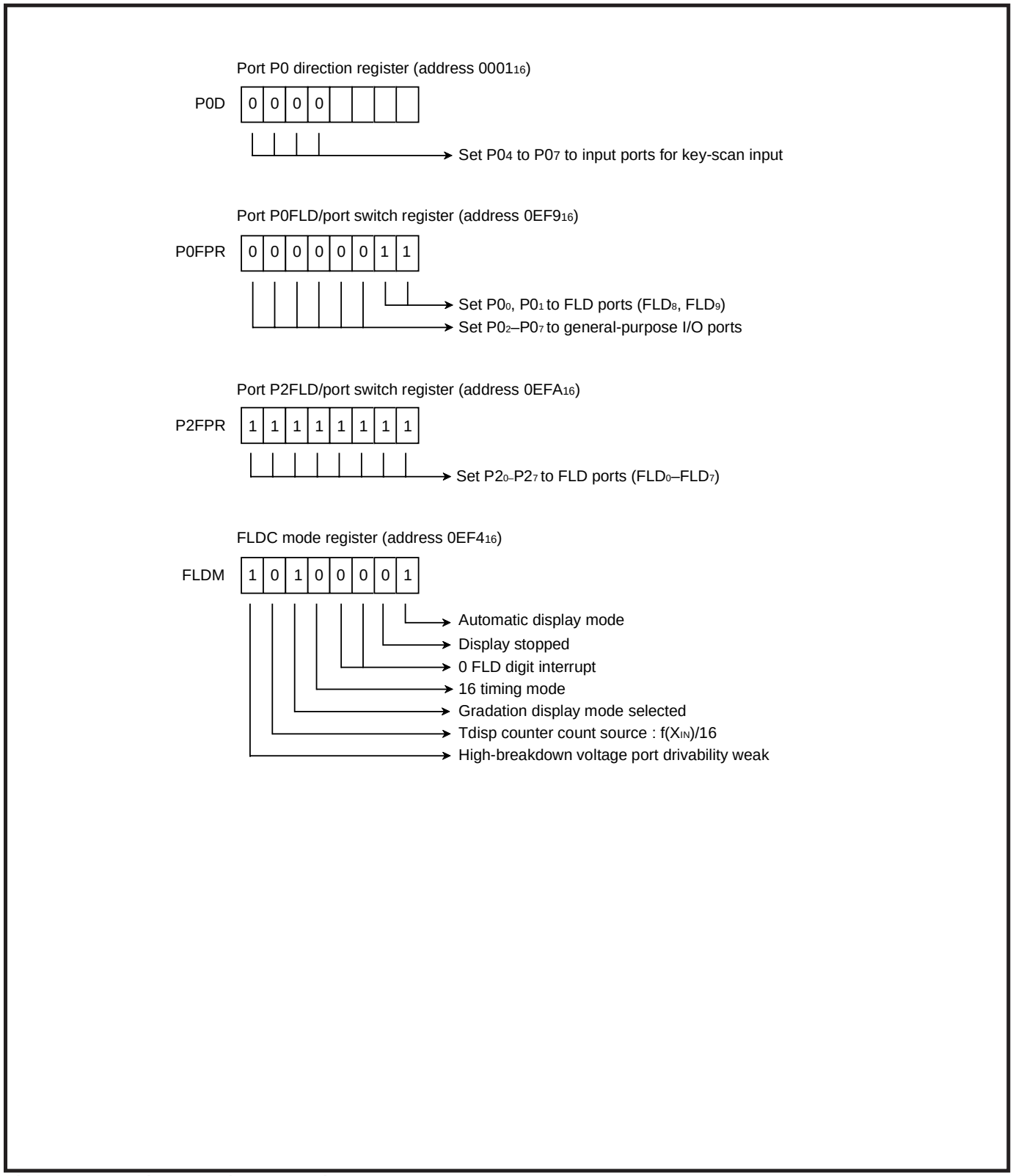


Fig. 2.4.23 Setting of relevant registers

Tdisp time set register (address $0EF5_{16}$)

TDISP | 32_{16}

50 (32_{16}) set; (50 + 1) × count source = 204 µs
Count source = $f(X_{IN})/16$ = 4 µs, at $f(X_{IN})$ = 4 MHz

Toff1 time set register (address $0EF6_{16}$)

TOFF1 | A_{16}

10 (A_{16}) set; 10 × count source = 40 µs
Count source = $f(X_{IN})/16$ = 4 µs, at $f(X_{IN})$ = 4 MHz

Toff2 time set register (address $0EF7_{16}$)

TOFF2 | 10_{16}

16 (10_{16}) set; 16 × count source = 64 µs
Count source = $f(X_{IN})/16$ = 4 µs, at $f(X_{IN})$ = 4 MHz

Note: Perform this setting when the gradation display mode is selected.

FLD data pointer (address $0EF8_{16}$)

FLDDP | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |

→ Set {(digit number) – 1} = 9

P1FLDRAM write disable register (address $0EF2_{16}$)

P1FLDRAM | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

→ Disable writing to FLDRAM corresponding to $P1_0$ to $P1_7$.

P3FLDRAM write disable register (address $0EF3_{16}$)

P3FLDRAM | | | | | | | 1 | 1 |

→ Disable writing to FLDRAM corresponding to $P3_0$, $P3_1$.

Interrupt request register 2 (address $003D_{16}$)

IREQ2 | | 0 | | | | | | |

→ Clear FLD digit interrupt request bit

Interrupt control register 2 (address $003F_{16}$)

ICON2 | 0 | 1 | | | | | | |

→ FLD digit interrupt: Enabled

FLDC mode register (address $0EF4_{16}$)

FLDM | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |

→ Display start

Setting of FLD automatic display RAM:

Table 2.4.3 FLD automatic display RAM map

1 to 16 timing display data stored area

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0FB0$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB1$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB2$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB3$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB4$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB5$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB6$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB7$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB8$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB9$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FBA$_{16}$								
0FBB$_{16}$								
0FBC$_{16}$								
0FBD$_{16}$								
0FBE$_{16}$								
0FBF$_{16}$								
0FC0$_{16}$							FLD$_9$	FLD$_8$
0FC1$_{16}$							FLD$_9$	FLD$_8$
0FC2$_{16}$							FLD$_9$	FLD$_8$
0FC3$_{16}$							FLD$_9$	FLD$_8$
0FC4$_{16}$							FLD$_9$	FLD$_8$
0FC5$_{16}$							FLD$_9$	FLD$_8$
0FC6$_{16}$							FLD$_9$	FLD$_8$
0FC7$_{16}$							FLD$_9$	FLD$_8$
0FC8$_{16}$							FLD$_9$	FLD$_8$
0FC9$_{16}$							FLD$_9$	FLD$_8$
0FCA$_{16}$								
0FCB$_{16}$								
0FCC$_{16}$								
0FCD$_{16}$								
0FCE$_{16}$								
0FCF$_{16}$								
0FD0$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD1$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD2$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD3$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD4$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD5$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD6$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD7$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD8$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD9$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FDA$_{16}$								
0FDB$_{16}$								
0FDC$_{16}$								
0FDD$_{16}$								
0FDE$_{16}$								
0FDF$_{16}$								
0FE0$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE1$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE2$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE3$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE4$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE5$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE6$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE7$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE8$_{16}$							FLD$_{25}$	FLD$_{24}$
0FE9$_{16}$							FLD$_{25}$	FLD$_{24}$

Gradation display control data stored area

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0F60$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0F61$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0F62$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0F63$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0F64$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0F65$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0F66$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0F67$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0F68$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0F69$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0F6A$_{16}$								
0F6B$_{16}$								
0F6C$_{16}$								
0F6D$_{16}$								
0F6E$_{16}$								
0F6F$_{16}$								
0F70$_{16}$							FLD$_9$	FLD$_8$
0F71$_{16}$							FLD$_9$	FLD$_8$
0F72$_{16}$							FLD$_9$	FLD$_8$
0F73$_{16}$							FLD$_9$	FLD$_8$
0F74$_{16}$							FLD$_9$	FLD$_8$
0F75$_{16}$							FLD$_9$	FLD$_8$
0F76$_{16}$							FLD$_9$	FLD$_8$
0F77$_{16}$							FLD$_9$	FLD$_8$
0F78$_{16}$							FLD$_9$	FLD$_8$
0F79$_{16}$							FLD$_9$	FLD$_8$
0F7A$_{16}$								
0F7B$_{16}$								
0F7C$_{16}$								
0F7D$_{16}$								
0F7E$_{16}$								
0F7F$_{16}$								
0F80$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0F81$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0F82$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0F83$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0F84$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0F85$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0F86$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0F87$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0F88$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0F89$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0F8A$_{16}$								
0F8B$_{16}$								
0F8C$_{16}$								
0F8D$_{16}$								
0F8E$_{16}$								
0F8F$_{16}$								
0F90$_{16}$							FLD$_{25}$	FLD$_{24}$
0F91$_{16}$							FLD$_{25}$	FLD$_{24}$
0F92$_{16}$							FLD$_{25}$	FLD$_{24}$
0F93$_{16}$							FLD$_{25}$	FLD$_{24}$
0F94$_{16}$							FLD$_{25}$	FLD$_{24}$
0F95$_{16}$							FLD$_{25}$	FLD$_{24}$
0F96$_{16}$							FLD$_{25}$	FLD$_{24}$
0F97$_{16}$							FLD$_{25}$	FLD$_{24}$
0F98$_{16}$							FLD$_{25}$	FLD$_{24}$
0F99$_{16}$							FLD$_{25}$	FLD$_{24}$

Corresponding digit pin

0FB0$_{16}$	→ FLD$_{25}$ (P3$_1$)
0FB1$_{16}$	→ FLD$_{24}$ (P3$_0$)
0FB2$_{16}$	→ FLD$_{23}$ (P1$_7$)
0FB3$_{16}$	→ FLD$_{22}$ (P1$_6$)
0FB4$_{16}$	→ FLD$_{21}$ (P1$_5$)
0FB5$_{16}$	→ FLD$_{20}$ (P1$_4$)
0FB6$_{16}$	→ FLD$_{19}$ (P1$_3$)
0FB7$_{16}$	→ FLD$_{18}$ (P1$_2$)
0FB8$_{16}$	→ FLD$_{17}$ (P1$_1$)
0FB9$_{16}$	→ FLD$_{16}$ (P1$_0$)
0FC0$_{16}$	→ FLD$_{25}$ (P3$_1$)
0FC1$_{16}$	→ FLD$_{24}$ (P3$_0$)
0FC2$_{16}$	→ FLD$_{23}$ (P1$_7$)
0FC3$_{16}$	→ FLD$_{22}$ (P1$_6$)
0FC4$_{16}$	→ FLD$_{21}$ (P1$_5$)
0FC5$_{16}$	→ FLD$_{20}$ (P1$_4$)
0FC6$_{16}$	→ FLD$_{19}$ (P1$_3$)
0FC7$_{16}$	→ FLD$_{18}$ (P1$_2$)
0FC8$_{16}$	→ FLD$_{17}$ (P1$_1$)
0FC9$_{16}$	→ FLD$_{16}$ (P1$_0$)

: Area which is used to set segment data

: Area which is used to set digit data

: Area which is available as ordinary RAM

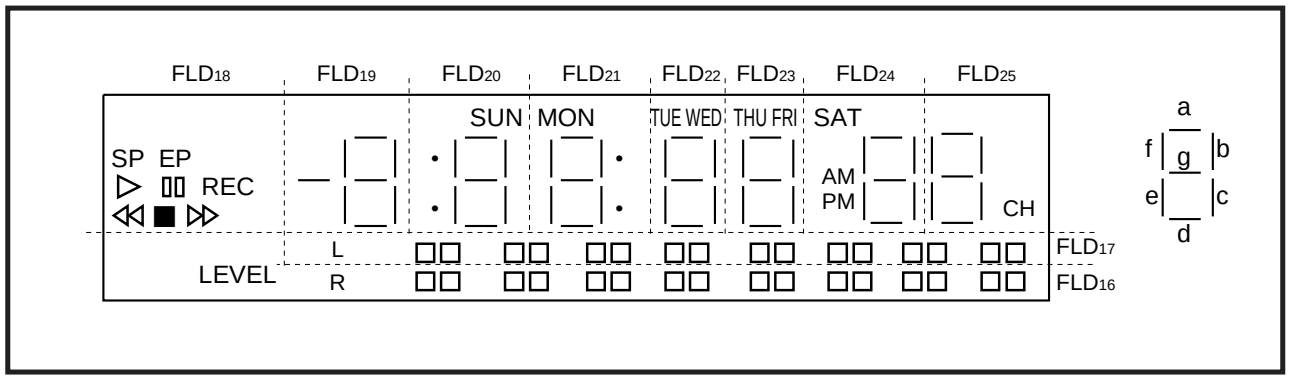


Fig. 2.4.24 FLD digit allocation example

Table 2.4.4 FLD automatic display RAM map example

1 to 16 timing display data stored area

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0FB0$_{16}$	CH	g	f	e	d	c	b	a
0FB1$_{16}$	SAT	g	f	e	d	c	b	a
0FB2$_{16}$	FRI	g	f	e	d	c	b	a
0FB3$_{16}$	WED	g	f	e	d	c	b	a
0FB4$_{16}$	MON	g	f	e	d	c	b	a
0FB5$_{16}$	SUN	g	f	e	d	c	b	a
0FB6$_{16}$	–	g	f	e	d	c	b	a
0FB7$_{16}$	■	◁◁	▷▷	▯▯	▷	REC	SP	EP
0FB8$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□
0FB9$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□
0FBA$_{16}$								
0FBB$_{16}$								
0FBC$_{16}$								
0FBD$_{16}$								
0FBE$_{16}$								
0FBF$_{16}$								
0FC0$_{16}$								
0FC1$_{16}$							PM	AM
0FC2$_{16}$								THU
0FC3$_{16}$								TUE
0FC4$_{16}$								:
0FC5$_{16}$								:
0FC6$_{16}$								
0FC7$_{16}$								
0FC8$_{16}$							L	
0FC9$_{16}$							R	LEVEL

Gradation display control data stored area

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0F60$_{16}$	CH	g	f	e	d	c	b	a
0F61$_{16}$	SAT	g	f	e	d	c	b	a
0F62$_{16}$	FRI	g	f	e	d	c	b	a
0F63$_{16}$	WED	g	f	e	d	c	b	a
0F64$_{16}$	MON	g	f	e	d	c	b	a
0F65$_{16}$	SUN	g	f	e	d	c	b	a
0F66$_{16}$	–	g	f	e	d	c	b	a
0F67$_{16}$	■	◁◁	▷▷	▯▯	▷	REC	SP	EP
0F68$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□
0F69$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□
0F6A$_{16}$								
0F6B$_{16}$								
0F6C$_{16}$								
0F6D$_{16}$								
0F6E$_{16}$								
0F6F$_{16}$								
0F70$_{16}$								
0F71$_{16}$							PM	AM
0F72$_{16}$								THU
0F73$_{16}$								TUE
0F74$_{16}$								:
0F75$_{16}$								:
0F76$_{16}$								
0F77$_{16}$								
0F78$_{16}$							L	
0F79$_{16}$							R	LEVEL

Corresponding digit pin

0FB0$_{16}$	→ FLD$_{25}$ (P3$_1$)
0FB1$_{16}$	→ FLD$_{24}$ (P3$_0$)
0FB2$_{16}$	→ FLD$_{23}$ (P1$_7$)
0FB3$_{16}$	→ FLD$_{22}$ (P1$_6$)
0FB4$_{16}$	→ FLD$_{21}$ (P1$_5$)
0FB5$_{16}$	→ FLD$_{20}$ (P1$_4$)
0FB6$_{16}$	→ FLD$_{19}$ (P1$_3$)
0FB7$_{16}$	→ FLD$_{18}$ (P1$_2$)
0FB8$_{16}$	→ FLD$_{17}$ (P1$_1$)
0FB9$_{16}$	→ FLD$_{16}$ (P1$_0$)
0FC0$_{16}$	→ FLD$_{25}$ (P3$_1$)
0FC1$_{16}$	→ FLD$_{24}$ (P3$_0$)
0FC2$_{16}$	→ FLD$_{23}$ (P1$_7$)
0FC3$_{16}$	→ FLD$_{22}$ (P1$_6$)
0FC4$_{16}$	→ FLD$_{21}$ (P1$_5$)
0FC5$_{16}$	→ FLD$_{20}$ (P1$_4$)
0FC6$_{16}$	→ FLD$_{19}$ (P1$_3$)
0FC7$_{16}$	→ FLD$_{18}$ (P1$_2$)
0FC8$_{16}$	→ FLD$_{17}$ (P1$_1$)
0FC9$_{16}$	→ FLD$_{16}$ (P1$_0$)

: Unused

Control procedure:

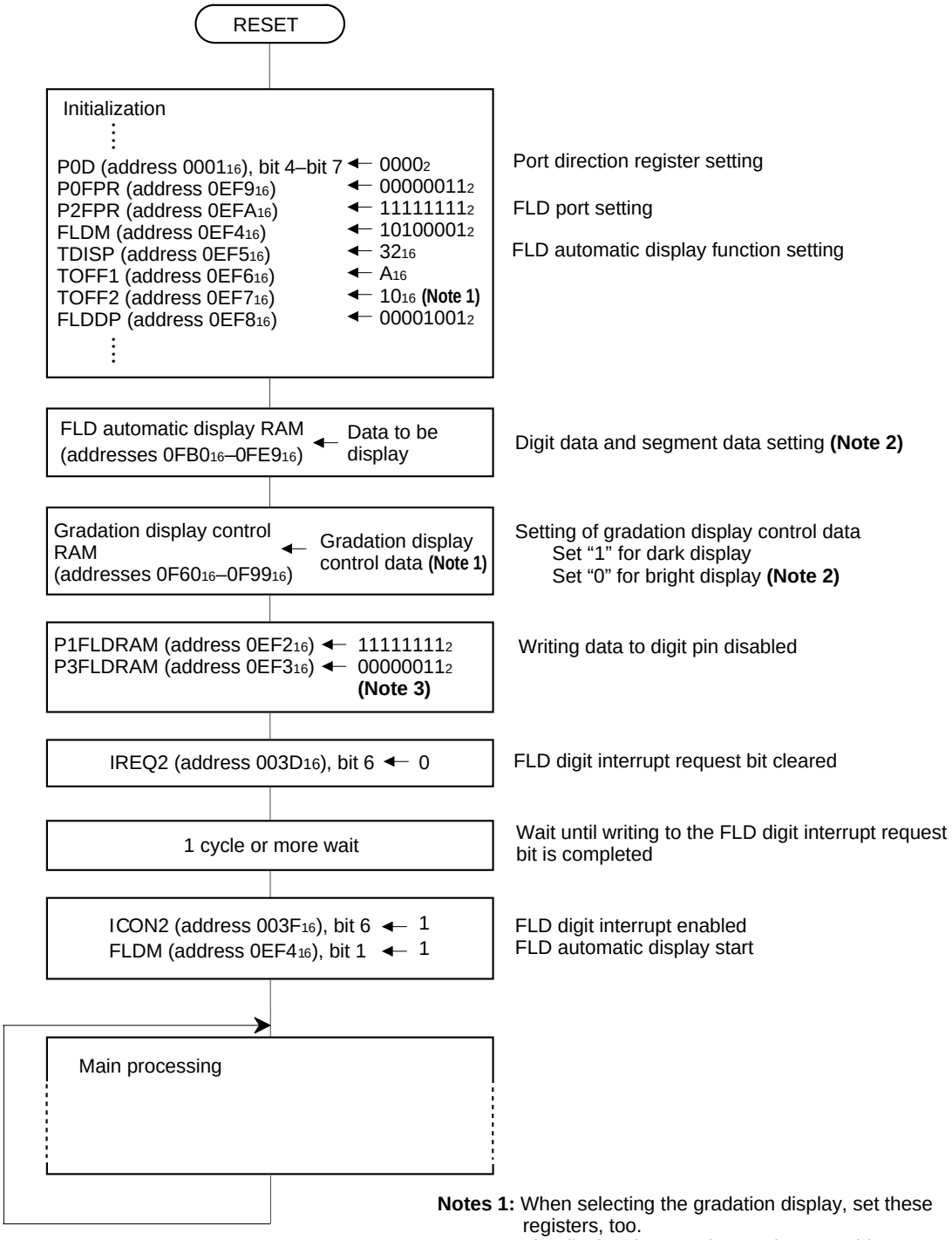
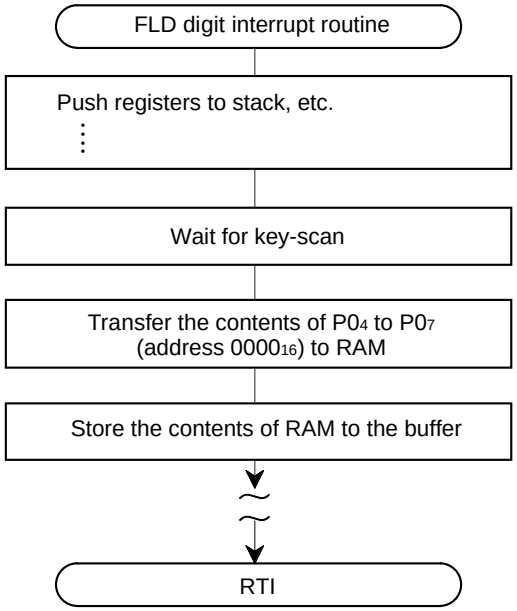


Fig. 2.4.25 Control procedure



Digit key-scan

Wait until the digit output is stabilized since the digit output waveform may become dull depending on the PCB pattern wiring length etc.

Keys read-in
(Set the port P0 direction register (P0D) (address 0001_{16}) to the input mode in the initialization, etc.)

(3) FLD display by software (example of not used FLD controller)

Outline: FLD display and key read-in is performed, using a timer interrupt.

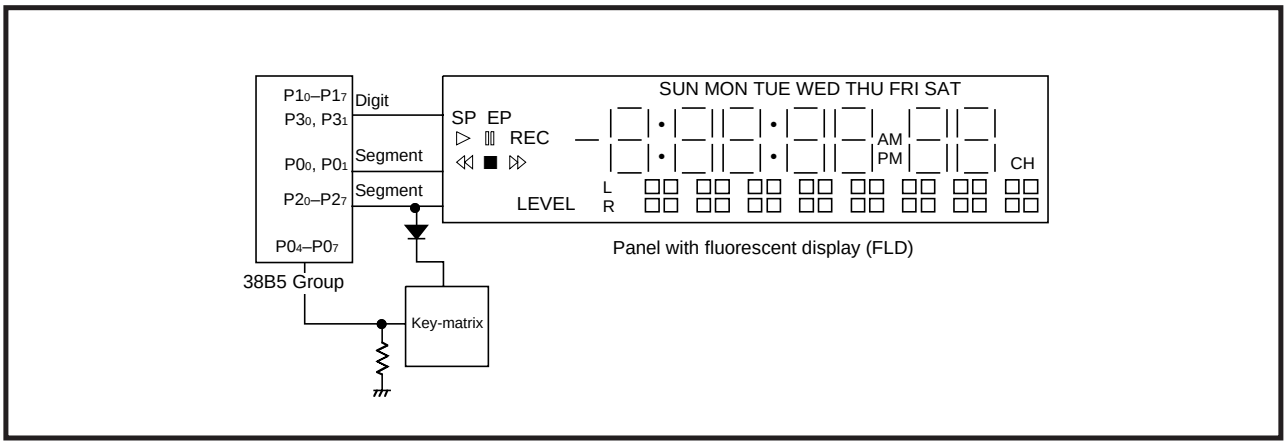


Fig. 2.4.26 Connection diagram

Specifications: •Use of 10 digits and 10 segments (8 key-scan included)
•Display controlled by software
•Use of timer 1 interrupt

Figure 2.4.27 shows the timing chart of FLD display by software, and Figure 2.4.28 shows the enlarged view of $P2_0$ to $P2_7$ key-scan. Generate the waveform shown Figure 2.4.28 by software and perform key-scan.

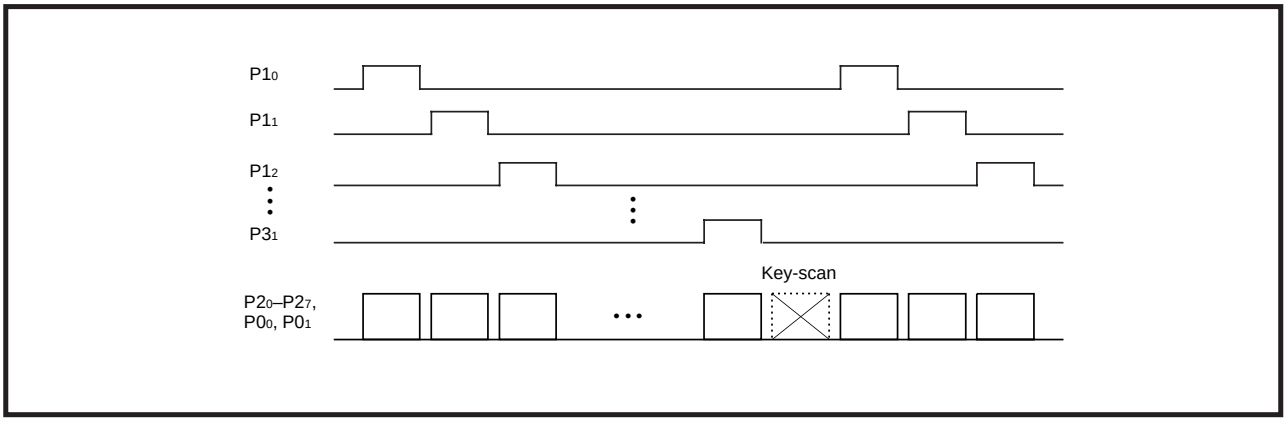


Fig. 2.4.27 Timing chart of FLD display by software

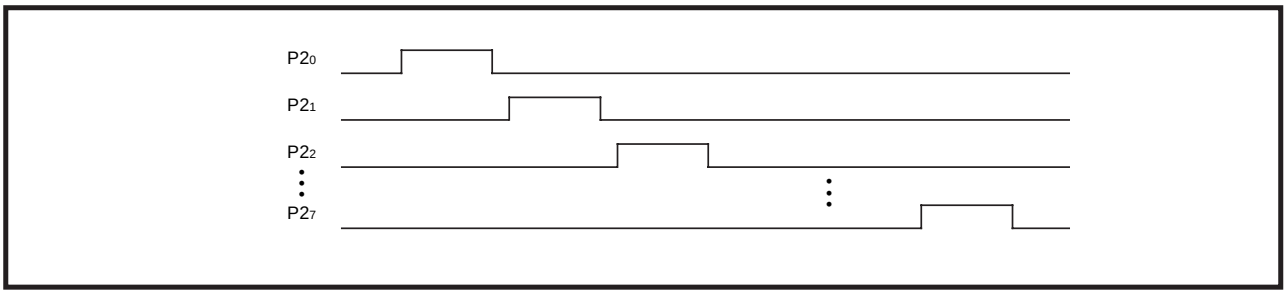


Fig. 2.4.28 Enlarged view of $P2_0$ to $P2_7$ key-scan

Figure 2.4.29 shows the setting of relevant registers.

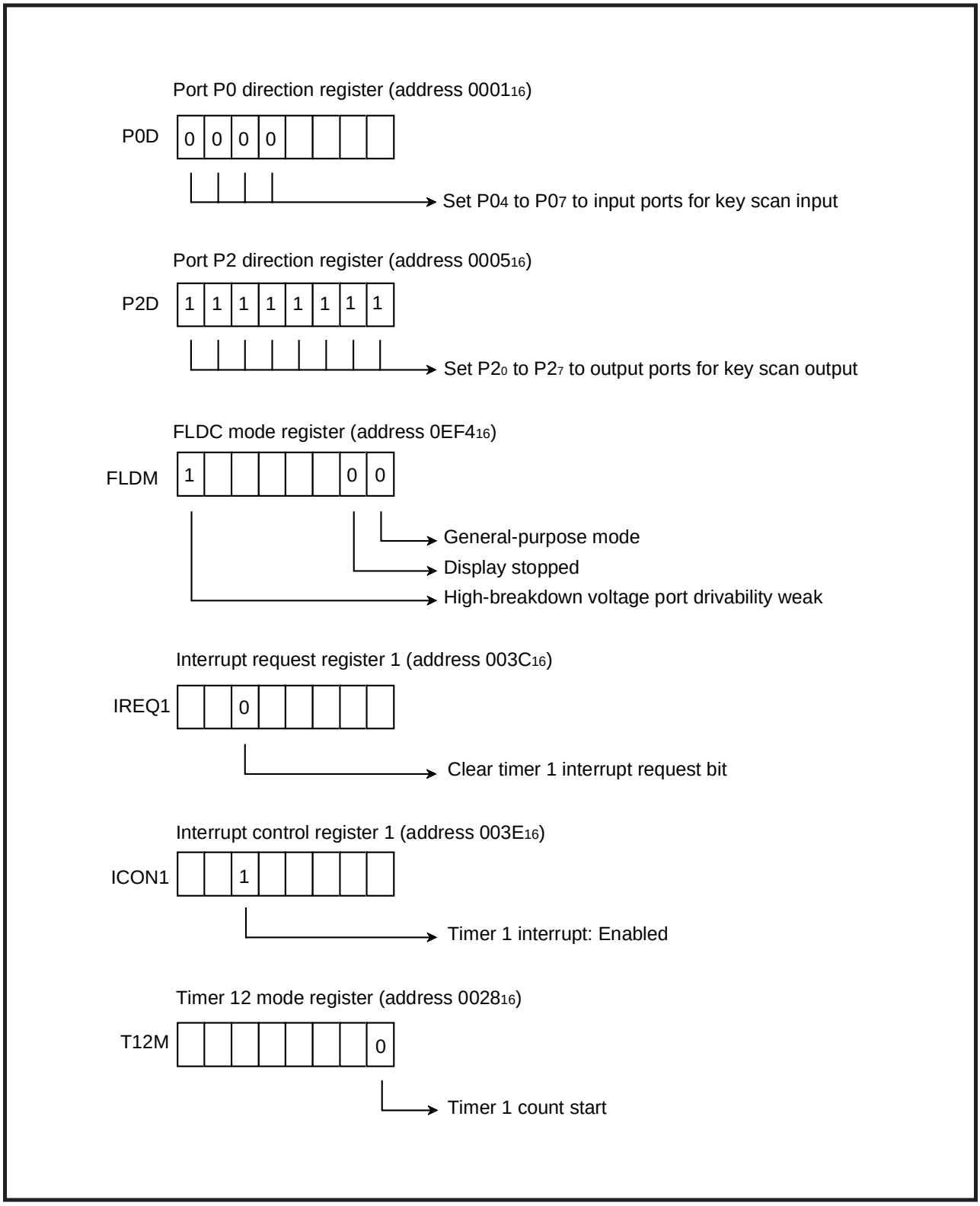


Fig. 2.4.29 Setting of relevant registers

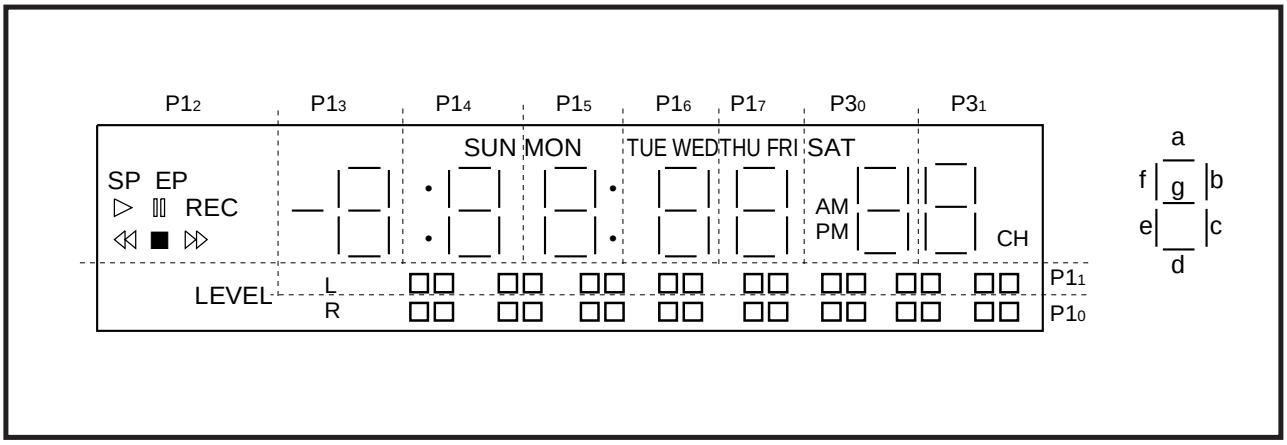


Fig. 2.4.30 FLD digit allocation example

Table 2.4.5 FLD automatic display RAM map example

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Corresponding digit pin
0FB0$_{16}$	CH	g	f	e	d	c	b	a	→ P3$_1$
0FB1$_{16}$	SAT	g	f	e	d	c	b	a	→ P3$_0$
0FB2$_{16}$	FRI	g	f	e	d	c	b	a	→ P1$_7$
0FB3$_{16}$	WED	g	f	e	d	c	b	a	→ P1$_6$
0FB4$_{16}$	MON	g	f	e	d	c	b	a	→ P1$_5$
0FB5$_{16}$	SUN	g	f	e	d	c	b	a	→ P1$_4$
0FB6$_{16}$	–	g	f	e	d	c	b	a	→ P1$_3$
0FB7$_{16}$	■	◁◁	▷▷	‖	▷	REC	SP	EP	→ P1$_2$
0FB8$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□	→ P1$_1$
0FB9$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□	→ P1$_0$
0FBA$_{16}$									
0FBB$_{16}$									
0FBC$_{16}$									
0FBD$_{16}$									
0FBE$_{16}$									
0FBF$_{16}$									
0FC0$_{16}$									→ P3$_1$
0FC1$_{16}$							PM	AM	→ P3$_0$
0FC2$_{16}$								THU	→ P1$_7$
0FC3$_{16}$								TUE	→ P1$_6$
0FC4$_{16}$								:	→ P1$_5$
0FC5$_{16}$								:	→ P1$_4$
0FC6$_{16}$									→ P1$_3$
0FC7$_{16}$									→ P1$_2$
0FC8$_{16}$							L		→ P1$_1$
0FC9$_{16}$							R	LEVEL	→ P1$_0$

: Unused

(The automatic display is not performed because FLD controller is not used.)

Control procedure:

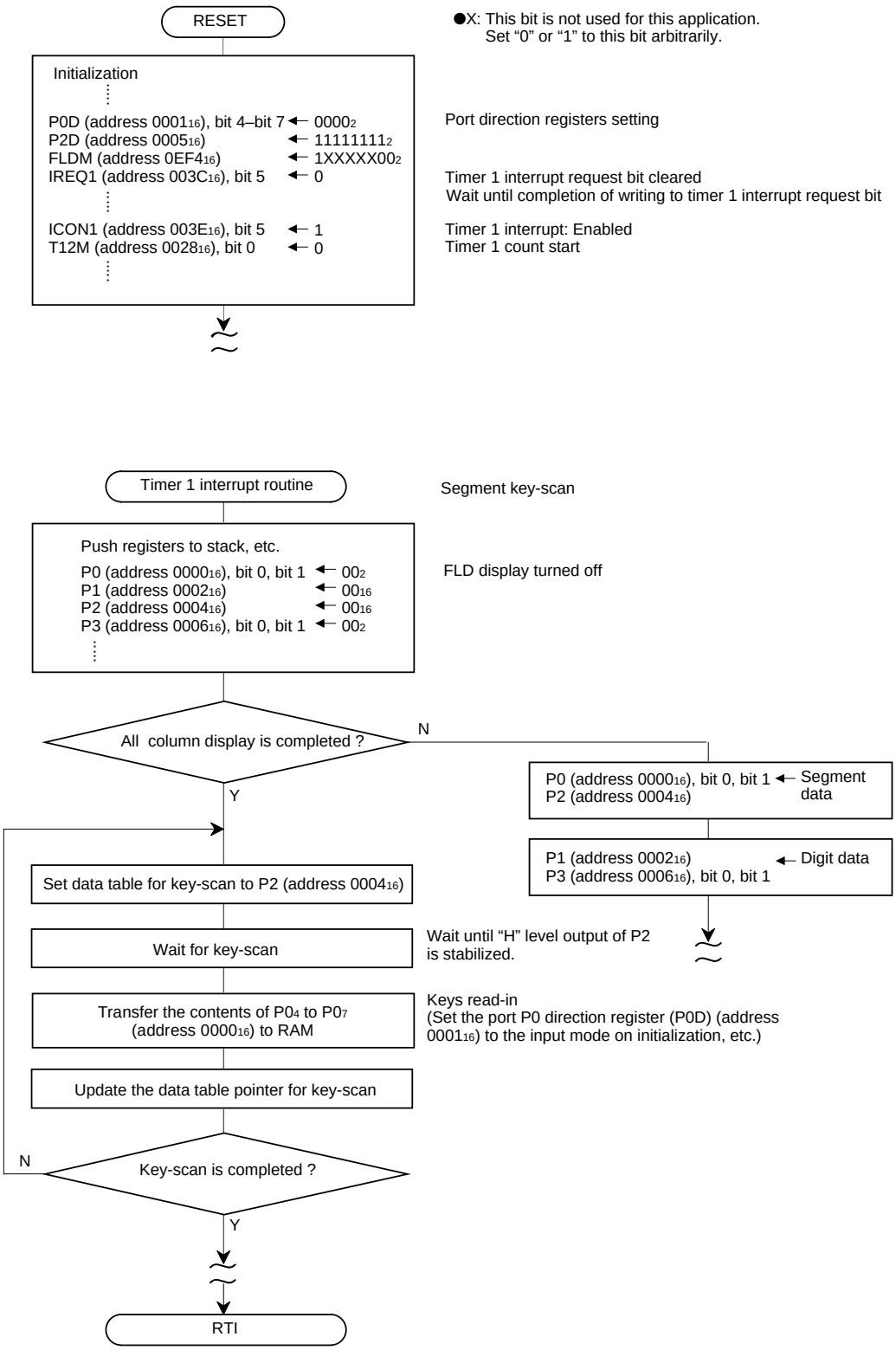


Fig. 2.4.31 Control procedure

(4) Display by combination with digit expander (M35501FP*) (basic combination example)
* For M35501FP, refer to section **"3.12 M35501FP"**.

Outline: The fluorescent display which has many display numbers (36 segments × 16 digits) is displayed by using the digit expander (M35501FP).

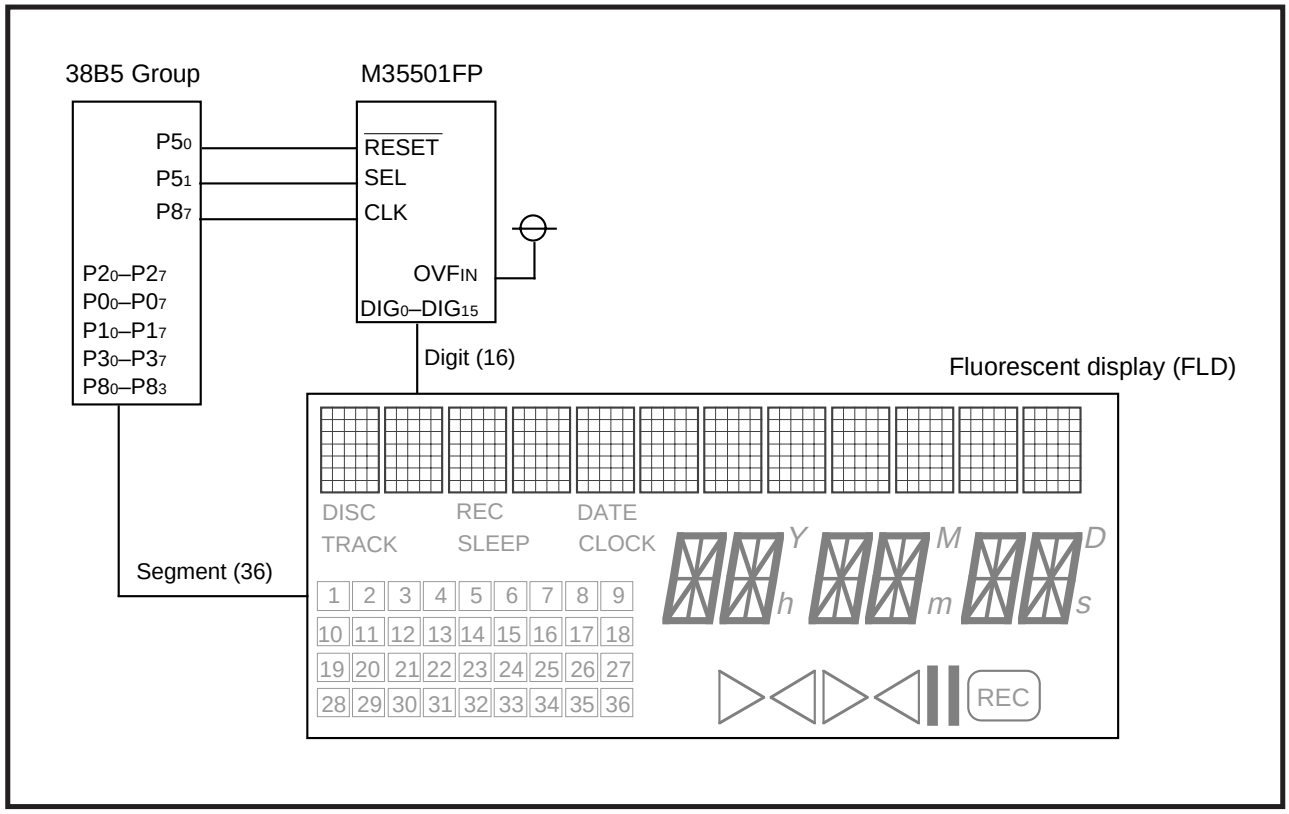


Fig. 2.4.32 Connection diagram

Specifications: •Use of M35501FP (M35501FP: 16 digits, 38B5 Group: 36 segments)
Ports $P5_0$ and $P5_1$ of 38B5 Group supply signals to the $\overline{RESET}$ and SEL pins of M35501FP respectively.
The $P8_7$ pin (FLD port vacant pin) supply signals to the CLK pin of M35501FP.
•Use of FLD automatic display mode of 38B5 Group
•Display in gradation display mode and 16 timing mode
•Toff1 = 40 µs, Toff2 = 64 µs, Tdisp = 204 µs, $f(X_{IN})$ = 4 MHz

Figure 2.4.33 shows the timing chart of 38B5 Group and M35501FP, and Figure 2.4.34 shows the timing chart (enlarged view) of digit and segment output.

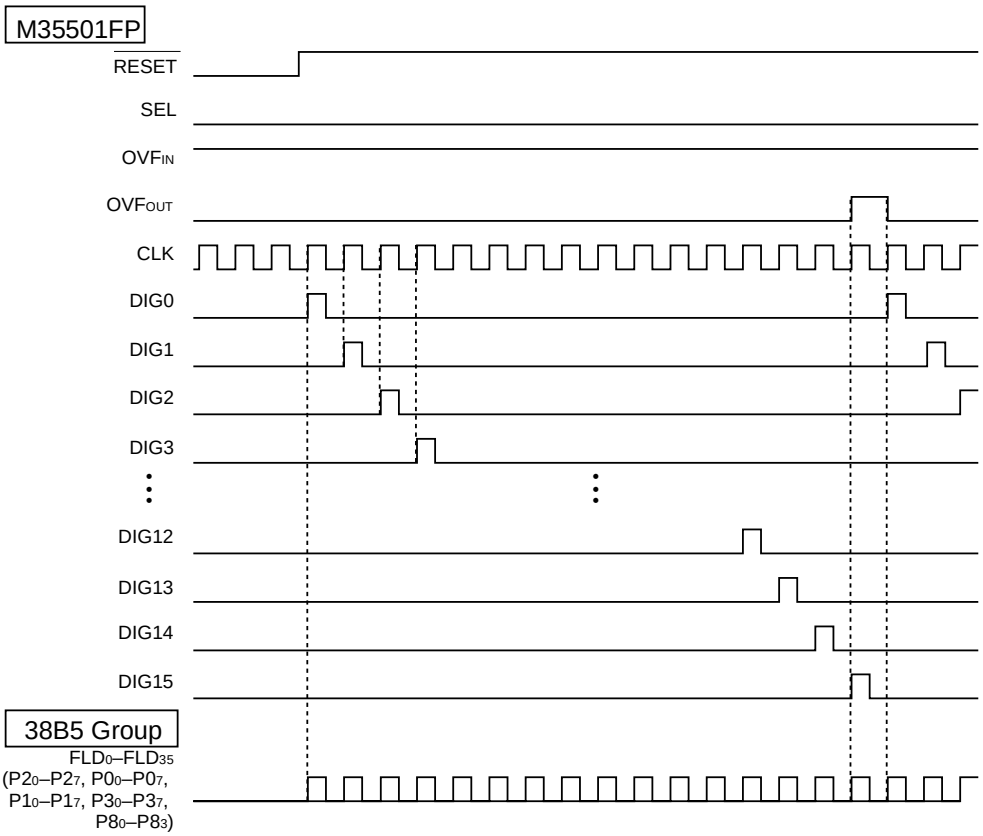


Fig. 2.4.33 Timing chart of 38B5 Group and M35501FP

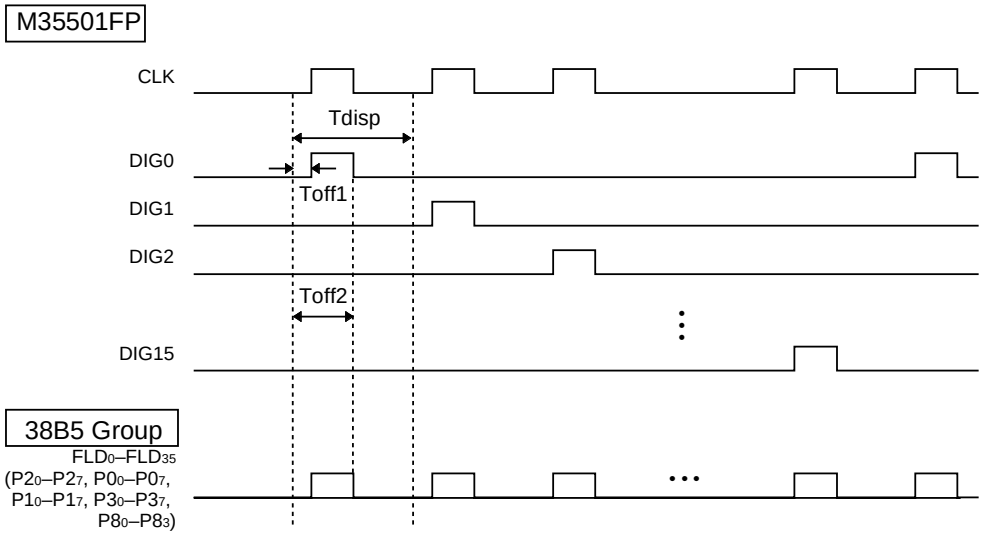
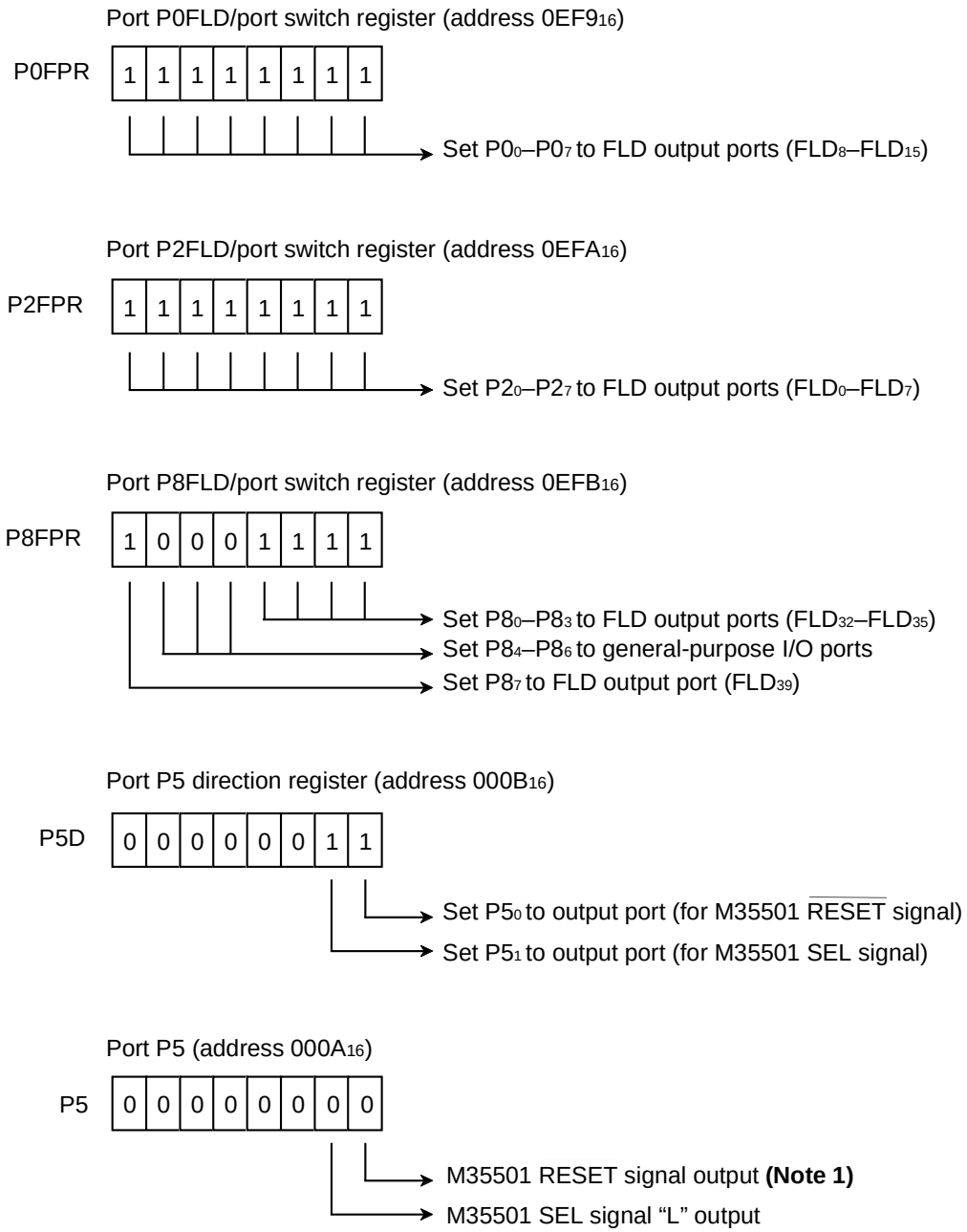


Fig. 2.4.34 Timing chart (enlarged view) of digit and segment output

2.4 FLD controller

Figure 2.4.35 shows the setting of relevant registers.



Note 1: After retain $\overline{\text{RESET}}$ signal output “L” for 2 µs or more, release reset by outputting “H” level from $\overline{\text{RESET}}$ signal output at CLK signal = “L” .

Fig. 2.4.35 Setting of relevant registers

FLDC mode register (address $0EF4_{16}$)

FLDM

1	0	1	0	0	0	0	1

- Automatic display mode
- Display stopped
- Tscan = 0 FLD digit interrupt
- 16 timing mode
- Gradation display mode selected
- Tdisp counter count source : $f(X_{IN})/16$
- High-breakdown voltage port drivability weak

Tdisp time set register (address $0EF5_{16}$)

TDISP | 32_{16} |

50 (32_{16}) set; (50 + 1) × count source = 204 µs
Count source = $f(X_{IN})/16$ = 4 µs, at $f(X_{IN})$ = 4 MHz

Toff1 time set register (address $0EF6_{16}$)

TOFF1 | A_{16} |

10 (A_{16}) set; 10 × count source = 40 µs
Count source = $f(X_{IN})/16$ = 4 µs, at $f(X_{IN})$ = 4 MHz

Toff2 time set register (address $0EF7_{16}$) **(Note 2)**

TOFF2 | 10_{16} |

16 (10_{16}) set; 16 × count source = 64 µs
Count source = $f(X_{IN})/16$ = 4 µs, at $f(X_{IN})$ = 4 MHz

Note 2: Perform this setting when the gradation display mode is selected.

FLD data pointer (address $0EF8_{16}$)

FLDDP

0	0	0	0	1	1	1	1

- Set {(digit number) – 1} = 15

FLDC mode register (address $0EF4_{16}$)

FLDM

1	0	1	0	0	0	1	1

- Display start

Setting of FLD automatic display RAM:

Table 2.4.6 FLD automatic display RAM map

1 to 16 timing display data stored area

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0FB0$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0FB1$_{16}$								
0FB2$_{16}$								
0FB3$_{16}$								
0FB4$_{16}$								
0FB5$_{16}$								
0FB6$_{16}$								
0FB7$_{16}$								
0FB8$_{16}$								
0FB9$_{16}$								
0FBA$_{16}$								
0FBB$_{16}$								
0FBC$_{16}$								
0FBD$_{16}$								
0FBE$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼
0FBF$_{16}$								
0FC0$_{16}$	FLD$_{15}$	FLD$_{14}$	FLD$_{13}$	FLD$_{12}$	FLD$_{11}$	FLD$_{10}$	FLD$_9$	FLD$_8$
0FC1$_{16}$								
0FC2$_{16}$								
0FC3$_{16}$								
0FC4$_{16}$								
0FC5$_{16}$								
0FC6$_{16}$								
0FC7$_{16}$								
0FC8$_{16}$								
0FC9$_{16}$								
0FCA$_{16}$								
0FCB$_{16}$								
0FCC$_{16}$								
0FCD$_{16}$								
0FCE$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼
0FCF$_{16}$								
0FD0$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0FD1$_{16}$								
0FD2$_{16}$								
0FD3$_{16}$								
0FD4$_{16}$								
0FD5$_{16}$								
0FD6$_{16}$								
0FD7$_{16}$								
0FD8$_{16}$								
0FD9$_{16}$								
0FDA$_{16}$								
0FDB$_{16}$								
0FDC$_{16}$								
0FDD$_{16}$								
0FDE$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼
0FDF$_{16}$								
0FE0$_{16}$	FLD$_{31}$	FLD$_{30}$	FLD$_{29}$	FLD$_{28}$	FLD$_{27}$	FLD$_{26}$	FLD$_{25}$	FLD$_{24}$
0FE1$_{16}$								
0FE2$_{16}$								
0FE3$_{16}$								
0FE4$_{16}$								
0FE5$_{16}$								
0FE6$_{16}$								
0FE7$_{16}$								
0FE8$_{16}$								
0FE9$_{16}$								
0FEA$_{16}$								
0FEB$_{16}$								
0FEC$_{16}$								
0FED$_{16}$								
0FEE$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼
0FEF$_{16}$								
0FF0$_{16}$	1				FLD$_{35}$	FLD$_{34}$	FLD$_{33}$	FLD$_{32}$
0FF1$_{16}$	1							
0FF2$_{16}$	1							
0FF3$_{16}$	1							
0FF4$_{16}$	1							
0FF5$_{16}$	1							
0FF6$_{16}$	1							
0FF7$_{16}$	1							
0FF8$_{16}$	1							
0FF9$_{16}$	1							
0FFA$_{16}$	1							
0FFB$_{16}$	1							
0FFC$_{16}$	1							
0FFD$_{16}$	1							
0FFE$_{16}$	1				▼	▼	▼	▼
0FFF$_{16}$	1							

Gradation display control data stored area

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0F60$_{16}$	FLD$_7$	FLD$_6$	FLD$_5$	FLD$_4$	FLD$_3$	FLD$_2$	FLD$_1$	FLD$_0$
0F61$_{16}$								
0F62$_{16}$								
0F63$_{16}$								
0F64$_{16}$								
0F65$_{16}$								
0F66$_{16}$								
0F67$_{16}$								
0F68$_{16}$								
0F69$_{16}$								
0F6A$_{16}$								
0F6B$_{16}$								
0F6C$_{16}$								
0F6D$_{16}$								
0F6E$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼
0F6F$_{16}$								
0F70$_{16}$	FLD$_{15}$	FLD$_{14}$	FLD$_{13}$	FLD$_{12}$	FLD$_{11}$	FLD$_{10}$	FLD$_9$	FLD$_8$
0F71$_{16}$								
0F72$_{16}$								
0F73$_{16}$								
0F74$_{16}$								
0F75$_{16}$								
0F76$_{16}$								
0F77$_{16}$								
0F78$_{16}$								
0F79$_{16}$								
0F7A$_{16}$								
0F7B$_{16}$								
0F7C$_{16}$								
0F7D$_{16}$								
0F7E$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼
0F7F$_{16}$								
0F80$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0F81$_{16}$								
0F82$_{16}$								
0F83$_{16}$								
0F84$_{16}$								
0F85$_{16}$								
0F86$_{16}$								
0F87$_{16}$								
0F88$_{16}$								
0F89$_{16}$								
0F8A$_{16}$								
0F8B$_{16}$								
0F8C$_{16}$								
0F8D$_{16}$								
0F8E$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼
0F8F$_{16}$								
0F90$_{16}$	FLD$_{31}$	FLD$_{30}$	FLD$_{29}$	FLD$_{28}$	FLD$_{27}$	FLD$_{26}$	FLD$_{25}$	FLD$_{24}$
0F91$_{16}$								
0F92$_{16}$								
0F93$_{16}$								
0F94$_{16}$								
0F95$_{16}$								
0F96$_{16}$								
0F97$_{16}$								
0F98$_{16}$								
0F99$_{16}$								
0F9A$_{16}$								
0F9B$_{16}$								
0F9C$_{16}$								
0F9D$_{16}$								
0F9E$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼
0F9F$_{16}$								
0FA0$_{16}$	1				FLD$_{35}$	FLD$_{34}$	FLD$_{33}$	FLD$_{32}$
0FA1$_{16}$	1							
0FA2$_{16}$	1							
0FA3$_{16}$	1							
0FA4$_{16}$	1							
0FA5$_{16}$	1							
0FA6$_{16}$	1							
0FA7$_{16}$	1							
0FA8$_{16}$	1							
0FA9$_{16}$	1							
0FAA$_{16}$	1							
0FAB$_{16}$	1							
0FAC$_{16}$	1							
0FAD$_{16}$	1							
0FAE$_{16}$	1				▼	▼	▼	▼
0FAF$_{16}$	1							

Corresponding digit pin of M35501FP

→ DIG$_{15}$	→ DIG$_{15}$
→ DIG$_{14}$	→ DIG$_{14}$
→ DIG$_{13}$	→ DIG$_{13}$
→ DIG$_{12}$	→ DIG$_{12}$
→ DIG$_{11}$	→ DIG$_{11}$
→ DIG$_{10}$	→ DIG$_{10}$
→ DIG$_9$	→ DIG$_9$
→ DIG$_8$	→ DIG$_8$
→ DIG$_7$	→ DIG$_7$
→ DIG$_6$	→ DIG$_6$
→ DIG$_5$	→ DIG$_5$
→ DIG$_4$	→ DIG$_4$
→ DIG$_3$	→ DIG$_3$
→ DIG$_2$	→ DIG$_2$
→ DIG$_1$	→ DIG$_1$
→ DIG$_0$	→ DIG$_0$

: CLK signal set area to M35501FP

: Unused

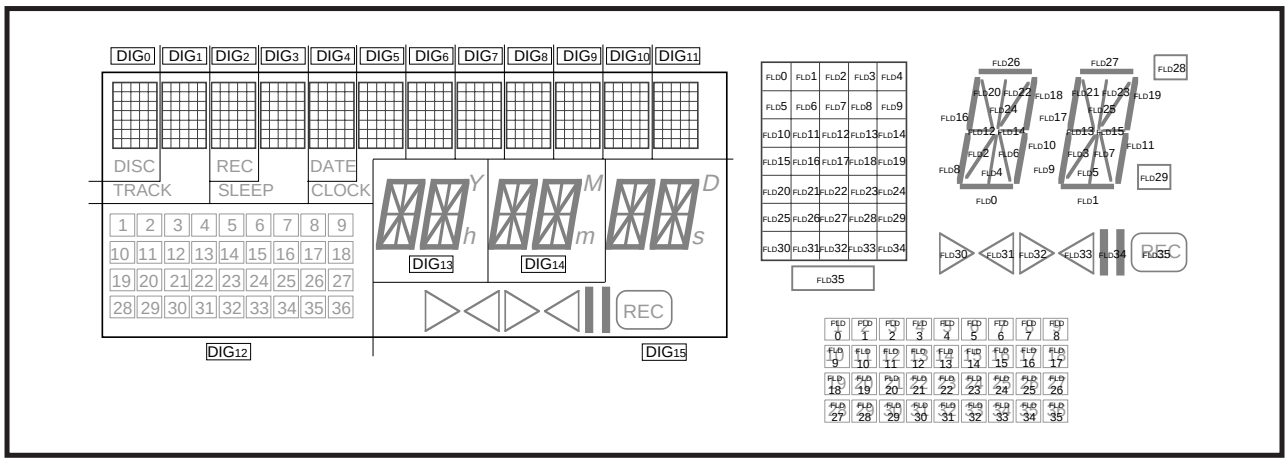


Fig. 2.4.36 FLD digit allocation example

Control procedure:

Figure 2.4.37 shows the control procedure.

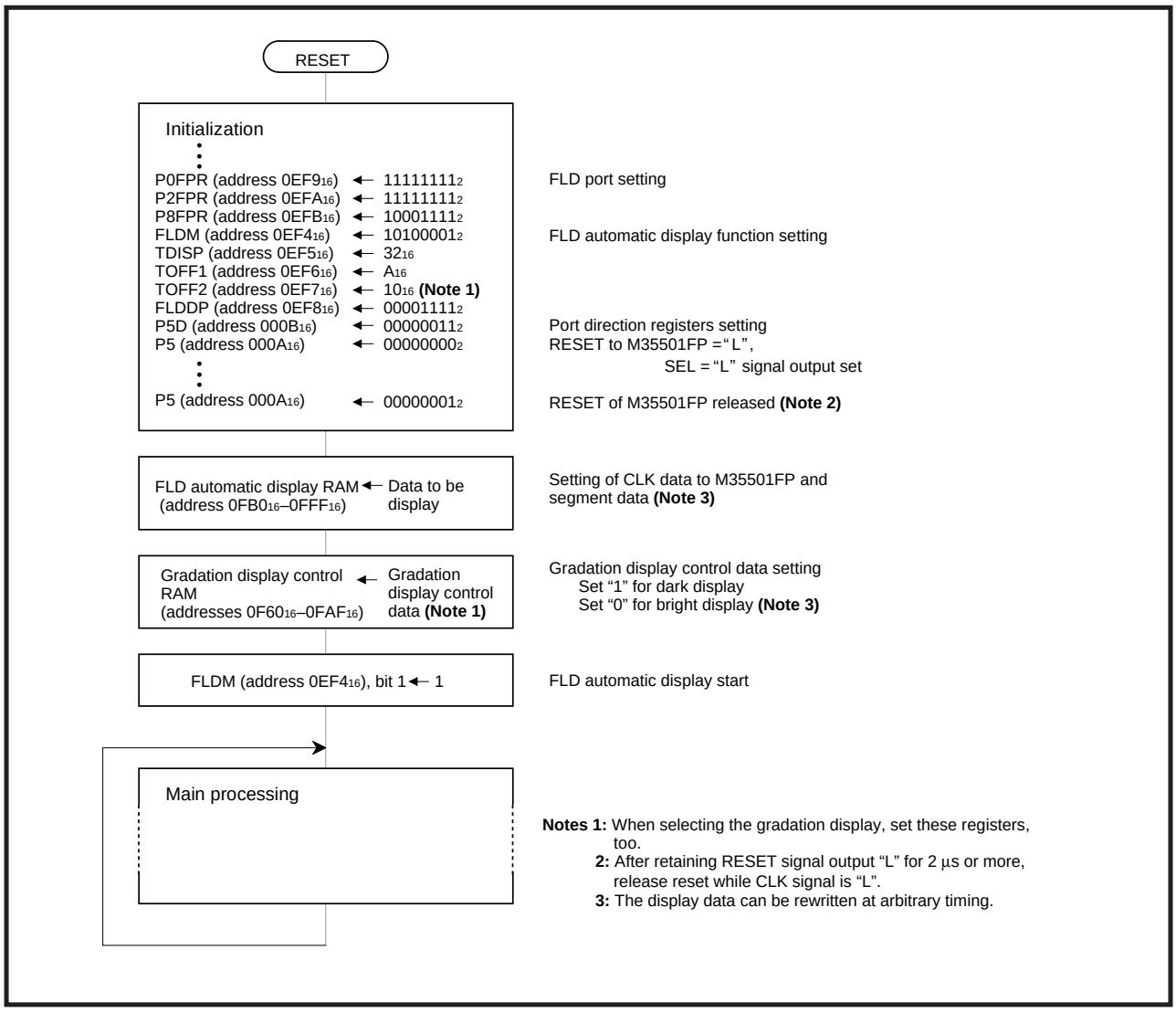


Fig. 2.4.37 Control procedure

(5) Display by combination with digit expander (M35501FP*) (example considering column discrepancy prevention)

* For M35501FP, refer to section **"3.12 M35501FP"**.

Outline: In the case of (4), which is displayed by using the digit expander (M35501FP), if a noise enters signals between 38B5 Group and M35501FP, a column discrepancy of display may occur. Prevent the column discrepancy by using the OVF_{OUT} output of M35501FP.

> The OVF_{OUT} pin of M35501FP outputs an overflow signal. The overflow signal is the signal which outputs "H" synchronizing to the last digit output signal of M35501FP, and the signal is output at definite intervals in the correct state. Incorrect state is detected by measuring the output period of this signal, and a column discrepancy is prevented.

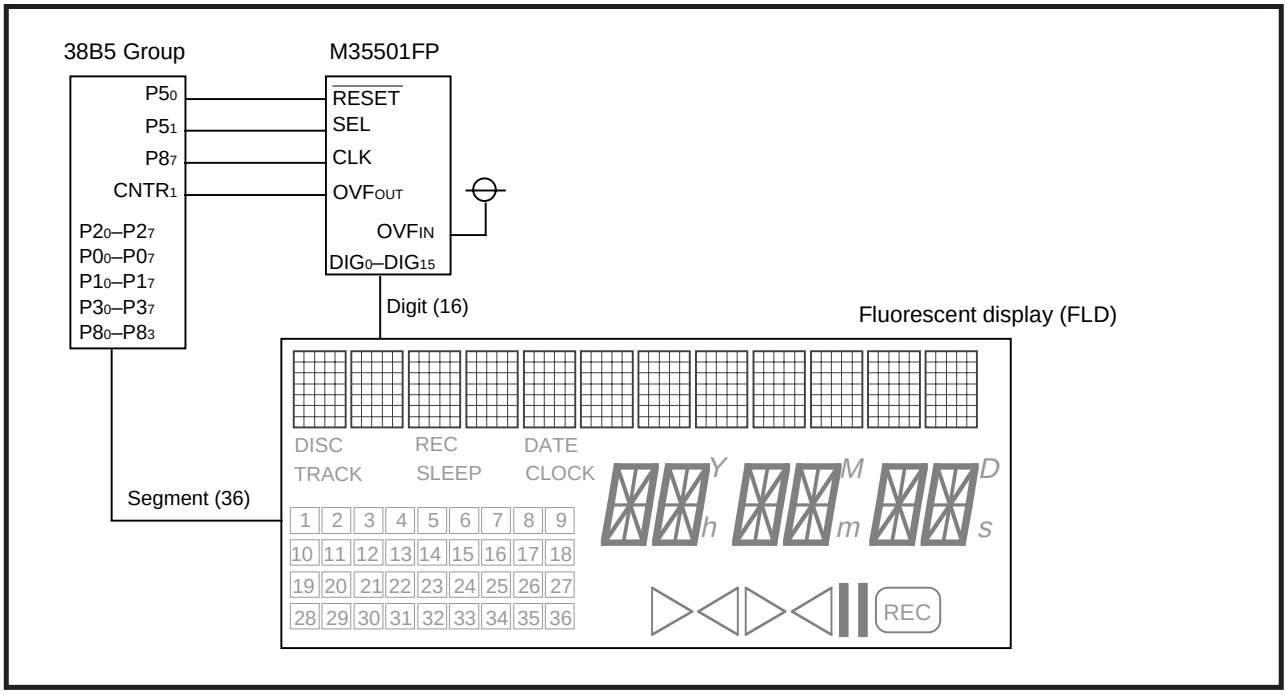


Fig. 2.4.38 Connection diagram

Specifications:
- Use of M35501FP (M35501: 16 digits, 38B5 Group: 36 segments)
  Ports $P5_0$ and $P5_1$ of 38B5 Group supply signal to the $\overline{RESET}$ and SEL pins of M35501FP respectively.
  The $P8_7$ pin (FLD port vacant pin) supply signals to the CLK pin of M35501FP.
- Use of FLD automatic display mode of 38B5 Group
- Display in gradation display mode and 16 timing mode
- Toff1 = 40 µs, Toff2 = 64 µs, Tdisp = 204 µs, $f(X_{IN})$ = 4 MHz

Countermeasures against column discrepancycolumn discrepancy → •OVF_{OUT} output of M35501FP input to $CNTR_1$ pin of 38B5 Group
Input signal to $CNTR_1$ pin is counted as a count source by timer 4 of 38B5 Group
The timer 6 interrupt is generated each time FLD display period (Tdisp (204 µs) × 16 column = 3.264 ms), and a value of timer 4 is confirmed. M35501FP is reset at incorrect state.

Figure 2.4.39 shows the timing chart (at correct state) of 38B5 Group and M35501FP, and Figure 2.4.40 shows the timing chart (at incorrect state) of 38B5 Group and M35501FP.

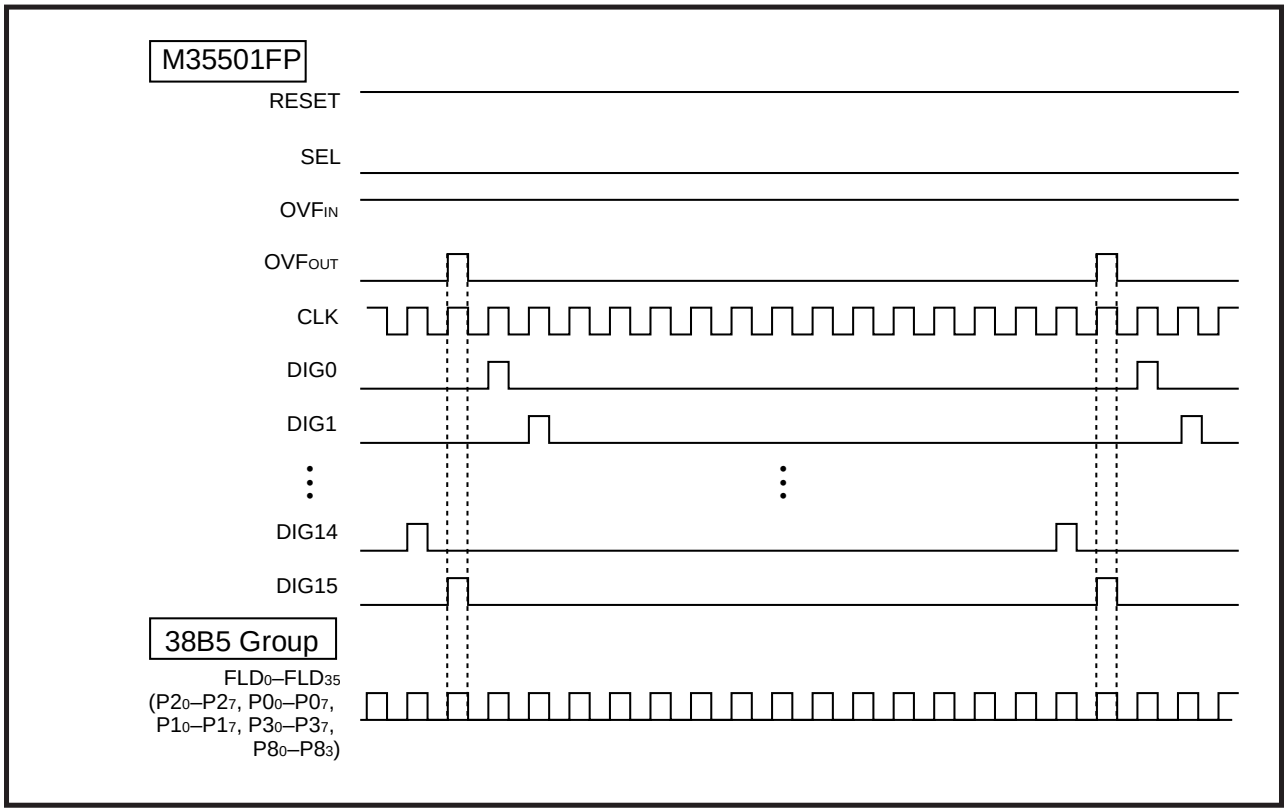


Fig. 2.4.39 Timing chart (at correct state) of 38B5 Group and M35501FP

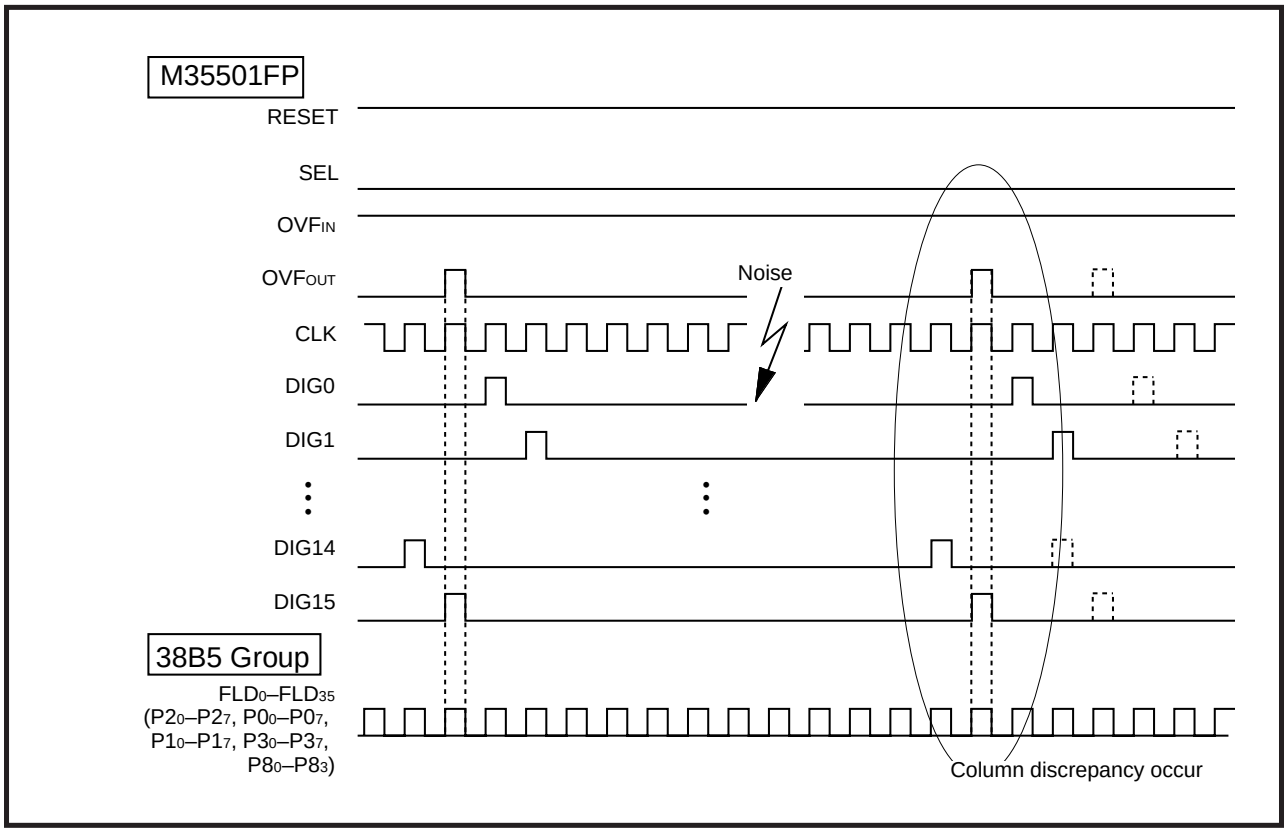


Fig. 2.4.40 Timing chart (at incorrect state) of 38B5 Group and M35501FP

Figure 2.4.41 shows the setting of relevant registers.

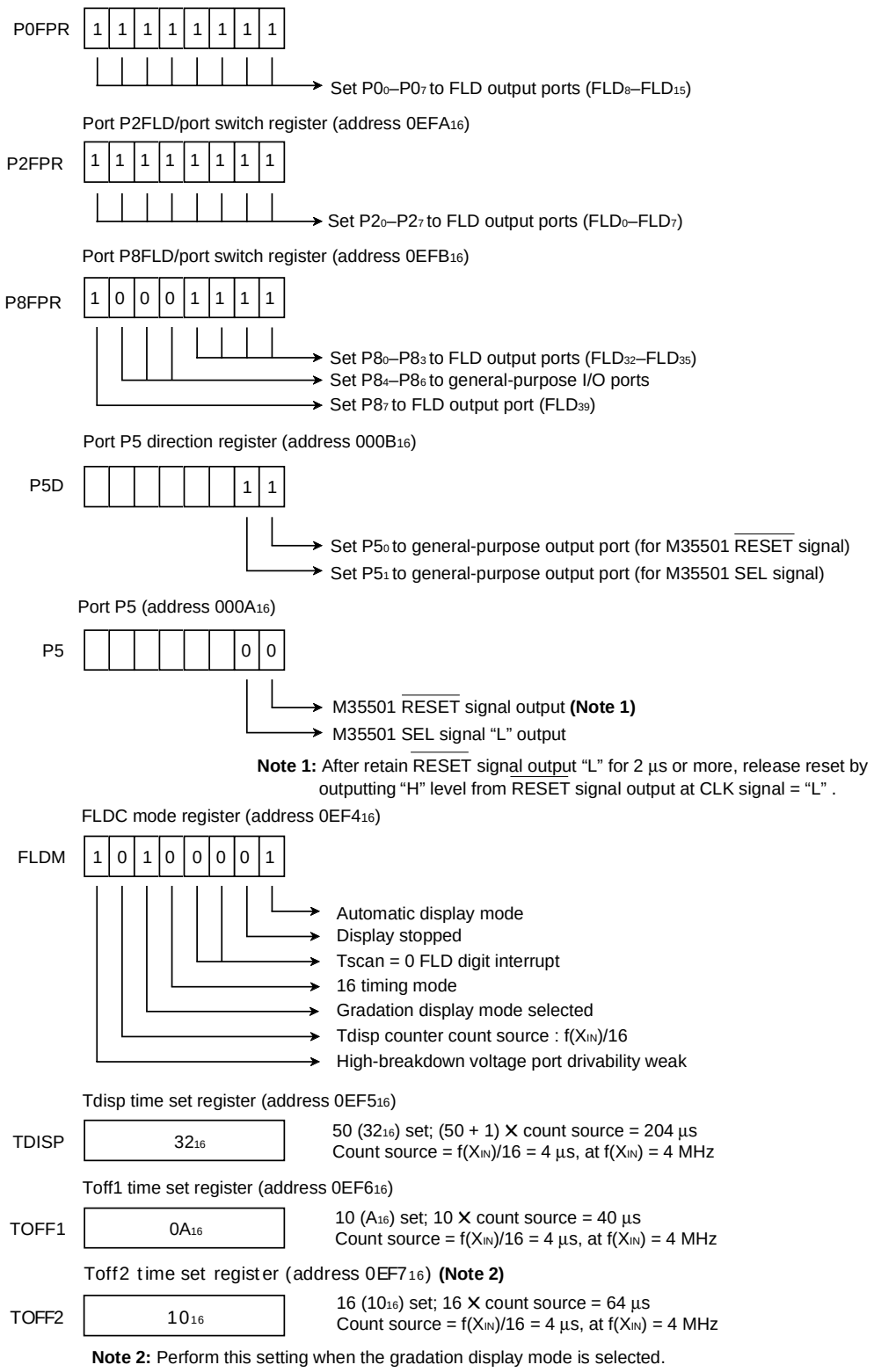
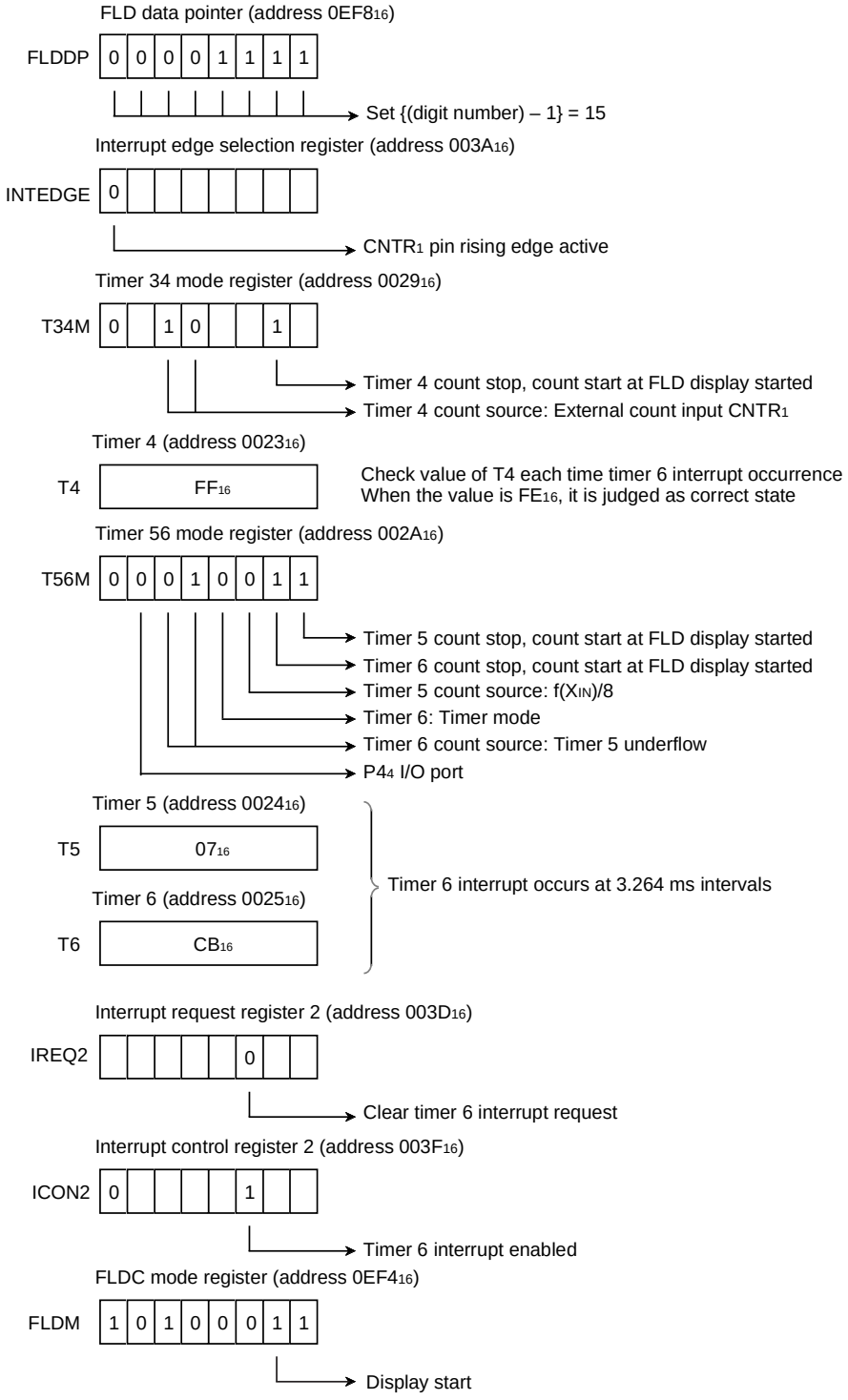


Fig. 2.4.41 Setting of relevant registers



2.4 FLD controller

Control procedure:

Figure 2.4.42 shows the control procedure.

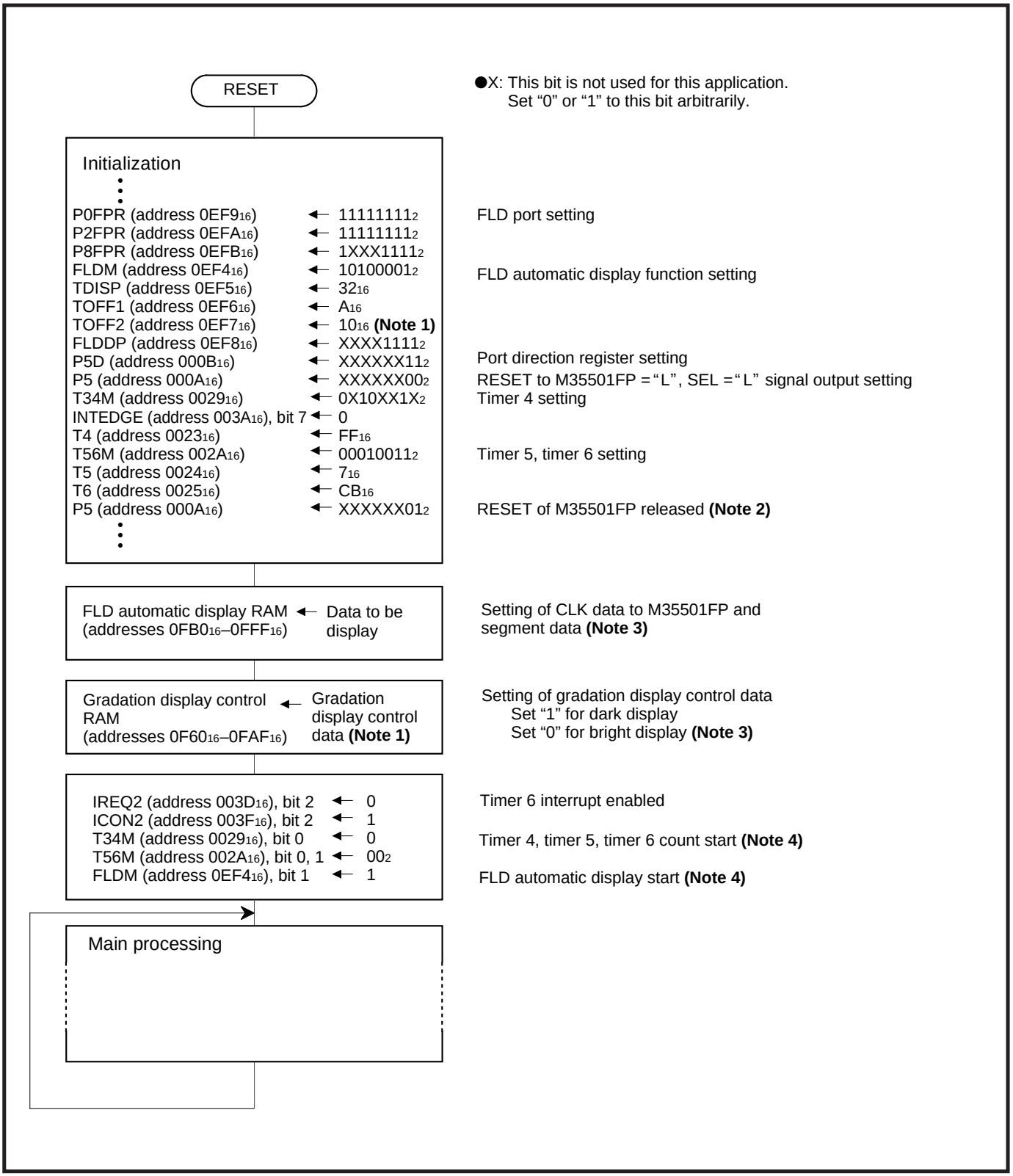
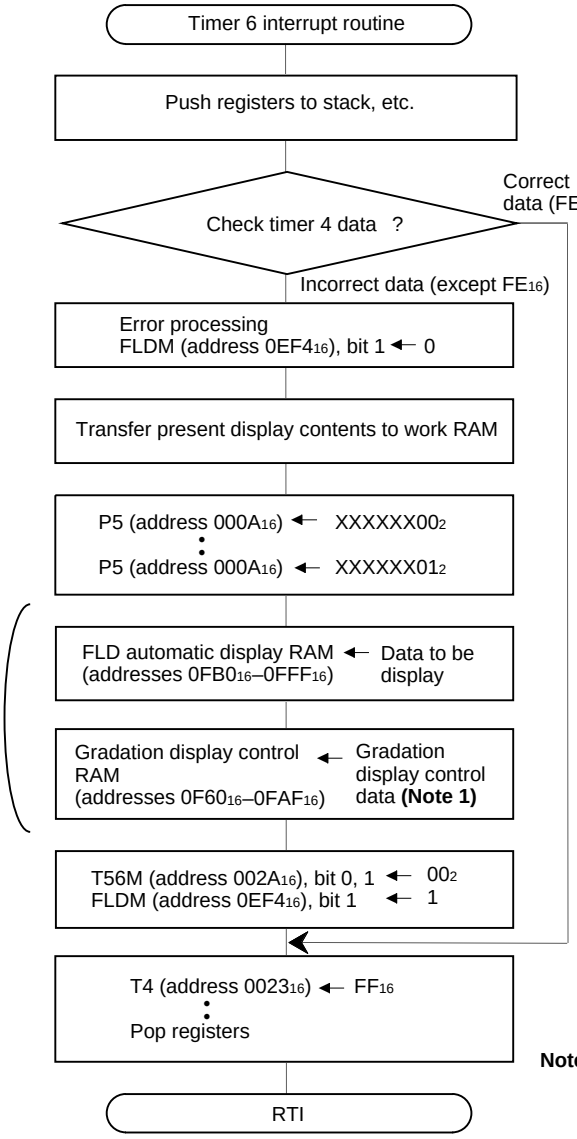


Fig. 2.4.42 Control procedure



Interrupt occurs each time FLD display cycle = 3.264 ms

Check of OVF_{OUT} output number during FLD display cycle
Only 1 time (=FE_{16}) is correct.

FLD turned off

Display data is retained as backup.

Setting of RESET to M35501FP = "L",
SEL = "L" signal output

Releasing RESET of M35501FP **(Note 2)**

Setting of CLK data to M35501FP
Setting of segment data by display data of backup

Setting of gradation display control data
Set "1" for dark display
Set "0" for bright display

(Note 5)

Timer 5, timer 6 count start **(Note 4)**
FLD turned on, automatic display start **(Note 4)**

Setting of timer 4 again

Notes 1: When selecting the gradation display, set these registers, too.
2: After retaining RESET signal output "L" for 2 µs or more, release reset while CLK signal is "L".
3: The display data can be rewritten at arbitrary timing.
4: Synchronize count start timing of timer 5 and timer 6 with FLD automatic display start timing as possible.
5: Set segment data of M35501FP at reset and others according to necessity.

2.4.4 Notes on use

- Set a value of 03_{16} or more to the Toff1 time set register.
- When displaying in the gradation display mode, select the 16 timing mode by the timing number control bit (bit 4 of FLDC mode register (address $0EF4_{16}$) = “0”).

2.5 A-D converter

This paragraph describes the setting method of A-D converter relevant registers, notes etc.

2.5.1 Memory assignment

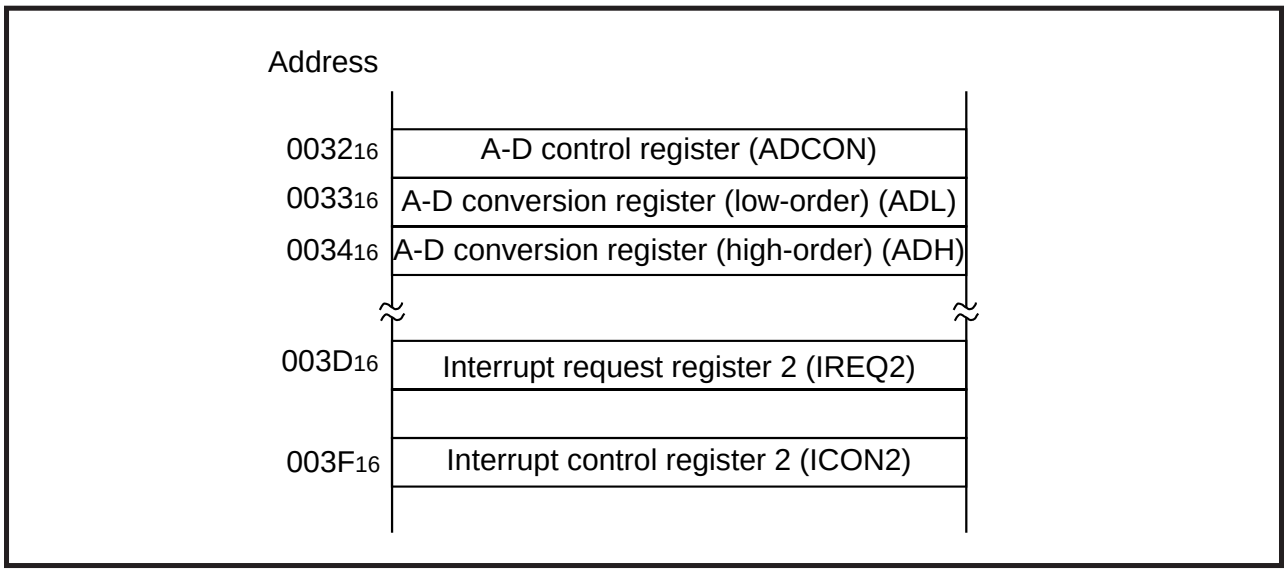


Fig. 2.5.1 Memory assignment of A-D converter relevant registers

2.5.2 Relevant registers

A-D control register

b7 b6 b5 b4 b3 b2 b1 b0

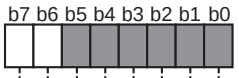
A-D control register
(ADCON: address 32₁₆)

b	Name	Functions	At reset	R	W
0	Analog input pin selection bits	b3 b2 b1 b0 0 0 0 0: P7₀/AN₀ 0 0 0 1: P7₁/AN₁ 0 0 1 0: P7₂/AN₂ 0 0 1 1: P7₃/AN₃ 0 1 0 0: P7₄/AN₄ 0 1 0 1: P7₅/AN₅ 0 1 1 0: P7₆/AN₆ 0 1 1 1: P7₇/AN₇ 1 0 0 0: P6₂/SRDY1/AN₈ 1 0 0 1: P6₃/AN₉ 1 0 1 0: P6₄/INT₄/SBUSY1/AN₁₀ 1 0 1 1: P6₅/SSTB1/AN₁₁	0	○	○
1			0	○	○
2			0	○	○
3			0	○	○
4	AD conversion completion bit	0: Conversion in progress 1: Conversion completed	1	○	○
5	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	○	×
6			0	○	×
7			0	○	×

Fig. 2.5.2 Structure of A-D control register

2.5 A-D converter

A-D conversion register (low-order)



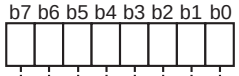
A-D conversion register (low-order)
(ADL: address 33_{16})

b	Functions	At reset	R	W
0	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are "0".	Undefined	○	×
1		Undefined	○	×
2		Undefined	○	×
3		Undefined	○	×
4		Undefined	○	×
5		Undefined	○	×
6	These are A-D conversion result (low-order 2 bits) stored bits. This is read exclusive register.	Undefined	○	×
7		Undefined	○	×

Note: Do not read this register during A-D conversion.

Fig. 2.5.3 Structure of A-D conversion register (low-order)

A-D conversion register (high-order)



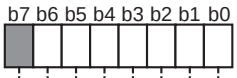
A-D conversion register (high-order)
(ADH: address 34_{16})

b	Functions	At reset	R	W
0	This is A-D conversion result (high-order 8 bits) stored bits. This is read exclusive register.	Undefined	○	×
1		Undefined	○	×
2		Undefined	○	×
3		Undefined	○	×
4		Undefined	○	×
5		Undefined	○	×
6		Undefined	○	×
7		Undefined	○	×

Note: Do not read this register during A-D conversion.

Fig. 2.5.4 Structure of A-D conversion register (high-order)

Interrupt request register 2



Interrupt request register 2
(IREQ2 : address $3D_{16}$)

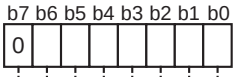
b	Name	Functions	At reset	R	W
0	Timer 4 interrupt request bit (Note)	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Timer 5 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	Timer 6 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O2 receive interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	INT_3/Serial I/O2 transmit interrupt request bit **(Note)**	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	INT_4 interrupt request bit A-D converter interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	FLD blanking interrupt request bit FLD digit interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are "0".		0	○	×

✻: "0" can be set by software, but "1" cannot be set.

Note: In the mask option type P, if timer 4 interrupt whose count source is $CNTR_1$ input and INT_3 interrupt are selected, these bits do not become "1".

Fig. 2.5.5 Structure of interrupt request register 2

Interrupt control register 2



Interrupt control register 2
(ICON2 : address $3F_{16}$)

b	Name	Functions	At reset	R	W
0	Timer 4 interrupt enable bit **(Note)**	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer 5 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer 6 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O2 receive interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
4	INT_3/Serial I/O2 transmit interrupt enable bit **(Note)**	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
5	INT_4 interrupt enable bit A-D converter interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
6	FLD blanking interrupt enable bit FLD digit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
7	Fix “0” to this bit.		0	○	○

Note: In the mask option type P, timer 4 interrupt whose count source is $CNTR_1$ input and INT_3 interrupt are not available.

Fig. 2.5.6 Structure of interrupt control register 2

2.5.3 A-D converter application examples

(1) Read-in of analog signal

Outline: The analog input voltage input from a sensor is converted to digital values.

Figure 2.5.7 shows a connection diagram, and Figure 2.5.8 shows the setting of relevant registers.

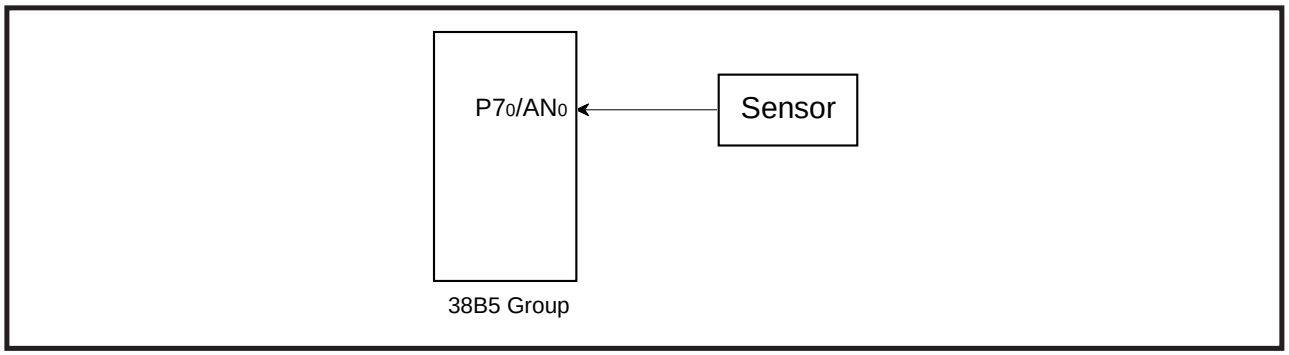
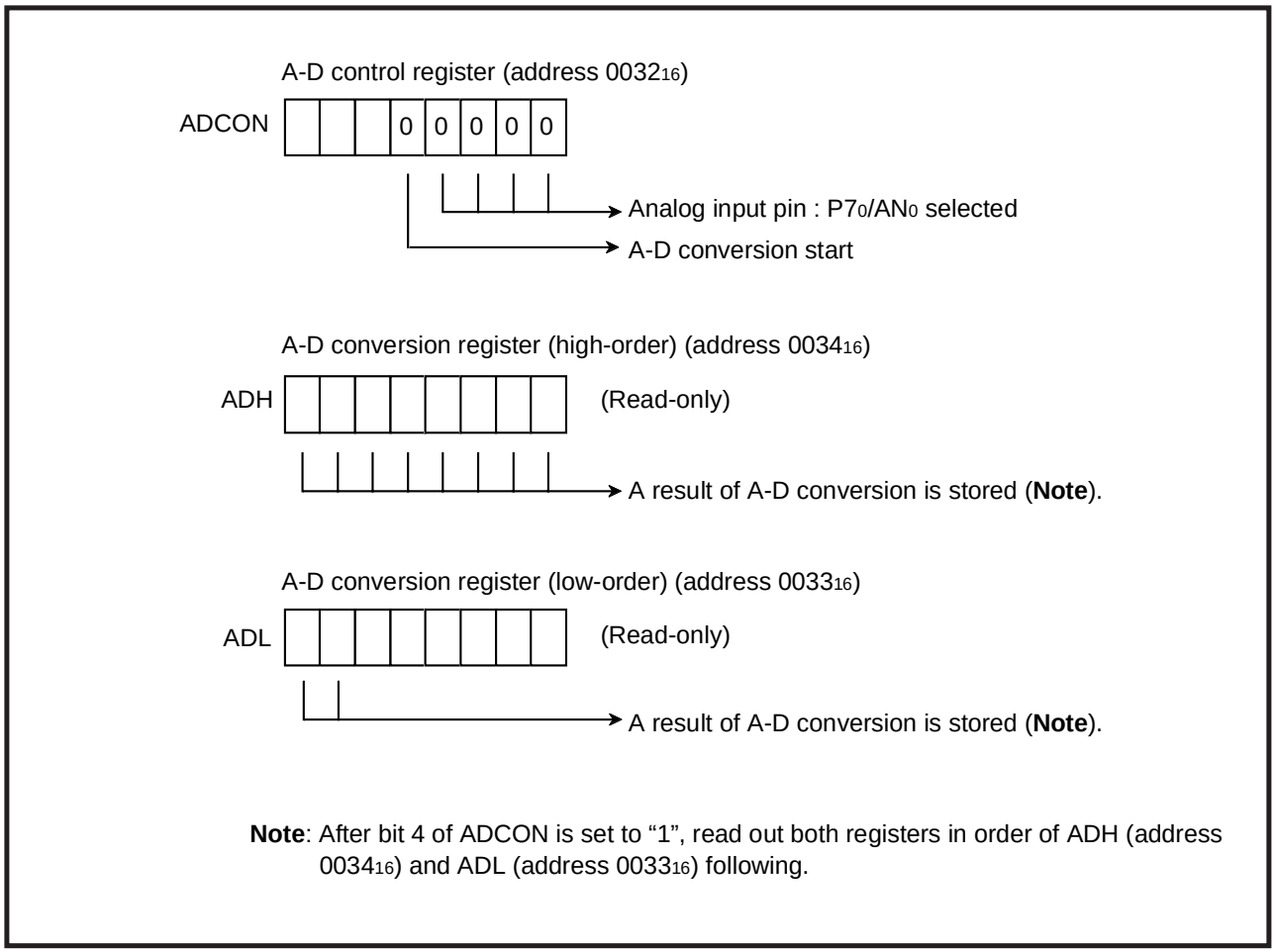


Fig. 2.5.7 Connection diagram

Specifications: •Conversion of analog input voltage input from sensor to digital values
•Use of $P7_0/AN_0$ pin as analog input pin



Note: After bit 4 of ADCON is set to "1", read out both registers in order of ADH (address 0034_{16}) and ADL (address 0033_{16}) following.

Fig. 2.5.8 Setting of relevant registers

2.5 A-D converter

Control procedure: A-D converter is started by performing register setting shown Figure 2.5.8. Figure 2.5.9 shows the control procedure.

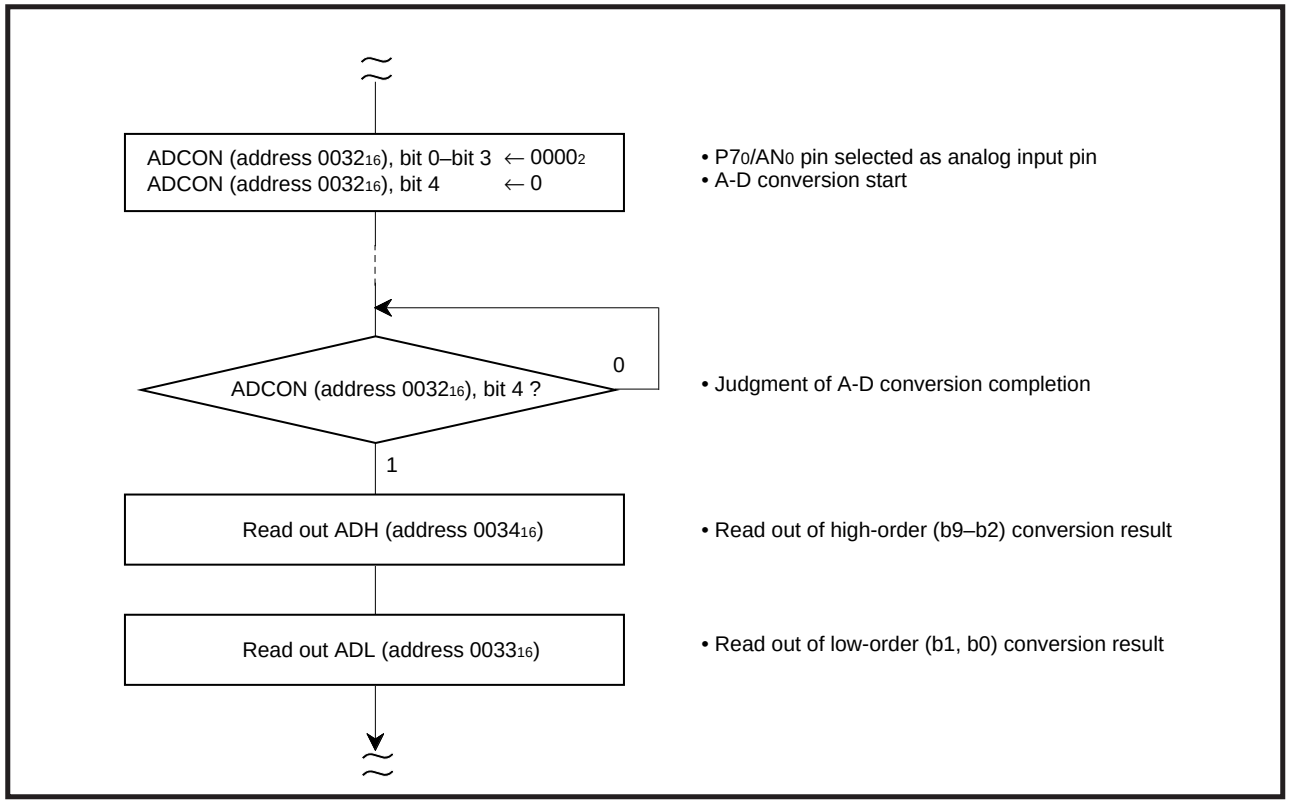


Fig. 2.5.9 Control procedure

2.5.4 Notes on use

(1) Analog input pin

■ Make the signal source impedance for analog input low, or equip an analog input pin with an external capacitor of 0.01 μF to 1 μF. Further, be sure to verify the operation of application products on the user side.

● **Reason**

An analog input pin includes the capacitor for analog voltage comparison. Accordingly, when signals from signal source with high impedance are input to an analog input pin, charge and discharge noise generates. This may cause the A-D conversion precision to be worse.

■ When the $P6_4/INT_4/S_{BUSY1}/AN_{10}$ pin is selected as analog input pin, external interrupt function (INT_4) becomes invalid.

(2) A-D converter power source pin

The AV_{SS} pin is A-D converter power source pin. Regardless of using the A-D conversion function or not, connect it as following :

• AV_{SS} : Connect to the V_{SS} line

● **Reason**

If the AV_{SS} pin is opened, the microcomputer may have a failure because of noise or others.

(3) Clock frequency during A-D conversion

The comparator consists of a capacity coupling, and a charge of the capacity will be lost if the clock frequency is too low. Thus, make sure the following during an A-D conversion.

• $f(X_{IN})$ is 250 kHz or more
• Use clock divided by main clock ($f(X_{IN})$) as internal system clock.
• Do not execute the **STP** instruction and **WIT** instruction

2.6 PWM

This paragraph describes the setting method of PWM relevant registers, notes etc.

2.6.1 Memory assignment

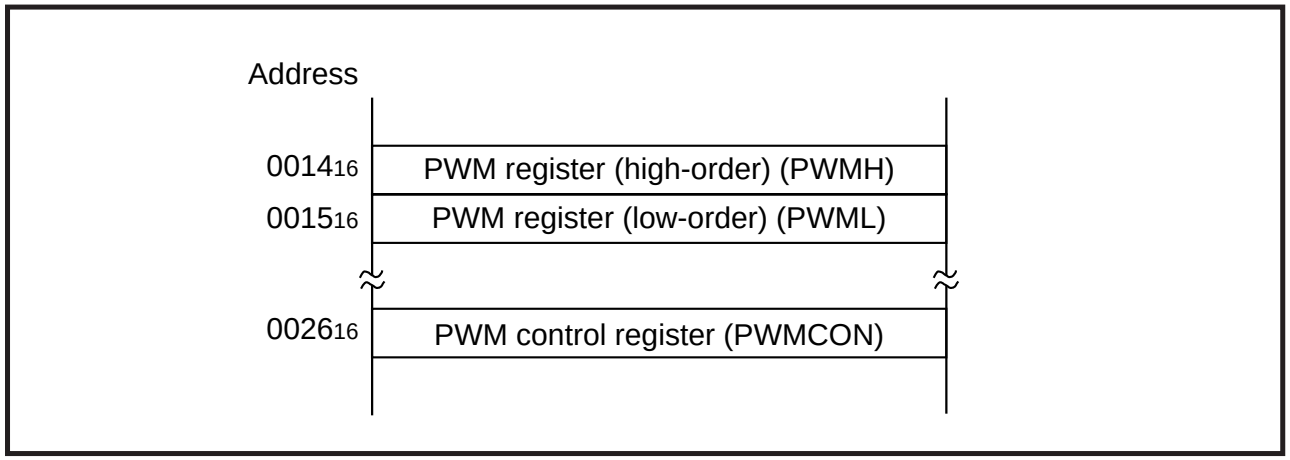


Fig. 2.6.1 Memory assignment of PWM relevant registers

2.6.2 Relevant registers

PWM register (high-order)

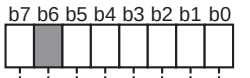
b7 b6 b5 b4 b3 b2 b1 b0

PWM register (high-order)
(PWMH: address 14_{16})

b	Functions	At reset	R	W
0	• High-order 8 bits of PWM_0 output data is set. • The values set in this register is transferred to the PWM latch each sub-period cycle (64 μs). (At $f(X_{IN})$ = 4 MHz) • When this register is read out, the value of the PWM register (high-order) is read out.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 2.6.2 Structure of PWM register (high-order)

PWM register (low-order)



PWM register (low-order)
(PWML: address 15_{16})

b	Functions	At reset	R	W
0	• Low-order 6 bits of PWM_0 output data is set. • The values set in this register is transferred to the PWM latch at each PWM cycle period (4096 µs). (At f(X_{IN}) = 4 MHz) • When this register is read out, the value of the PWM latch (low-order 6 bits) is read out.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6	Nothing is arranged for this bit. This bit is a write disabled bit. When this bit is read out, the contents are “0”.	Undefined	○	×
7	• This bit indicates whether the transfer to the PWM latch is completed. 0: Transfer is completed 1: Transfer is not completed • This bit is set to “1” at writing.	Undefined	○	×

Fig. 2.6.3 Structure of PWM register (low-order)

PWM control register

PWM control register
(PWMCON: address 26_{16})

b	Name	Functions	At reset	R	W
0	$P8_7$/PWM output selection bit	0: I/O port 1: PWM output	0	○	○
1	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	○	×
2			0	○	×
3			0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Fig. 2.6.4 Structure of PWM control register

2.6.3 PWM application example

(1) Control of VS tuner

Figure 2.6.5 shows a connection diagram, and Figure 2.6.6 shows the setting of relevant registers.

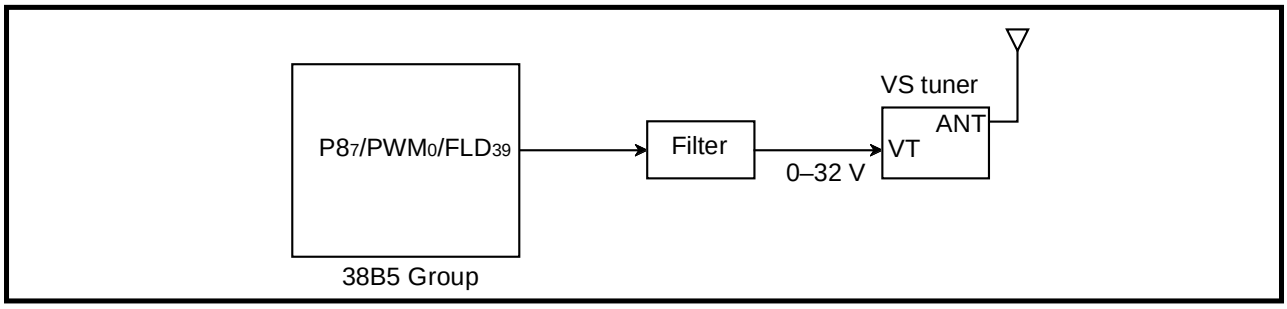


Fig. 2.6.5 Connection diagram

Outline: • Control of VS tuner by using the 14-bit resolution PWM_0 output function
• $f(X_{IN})$ = 4 MHz

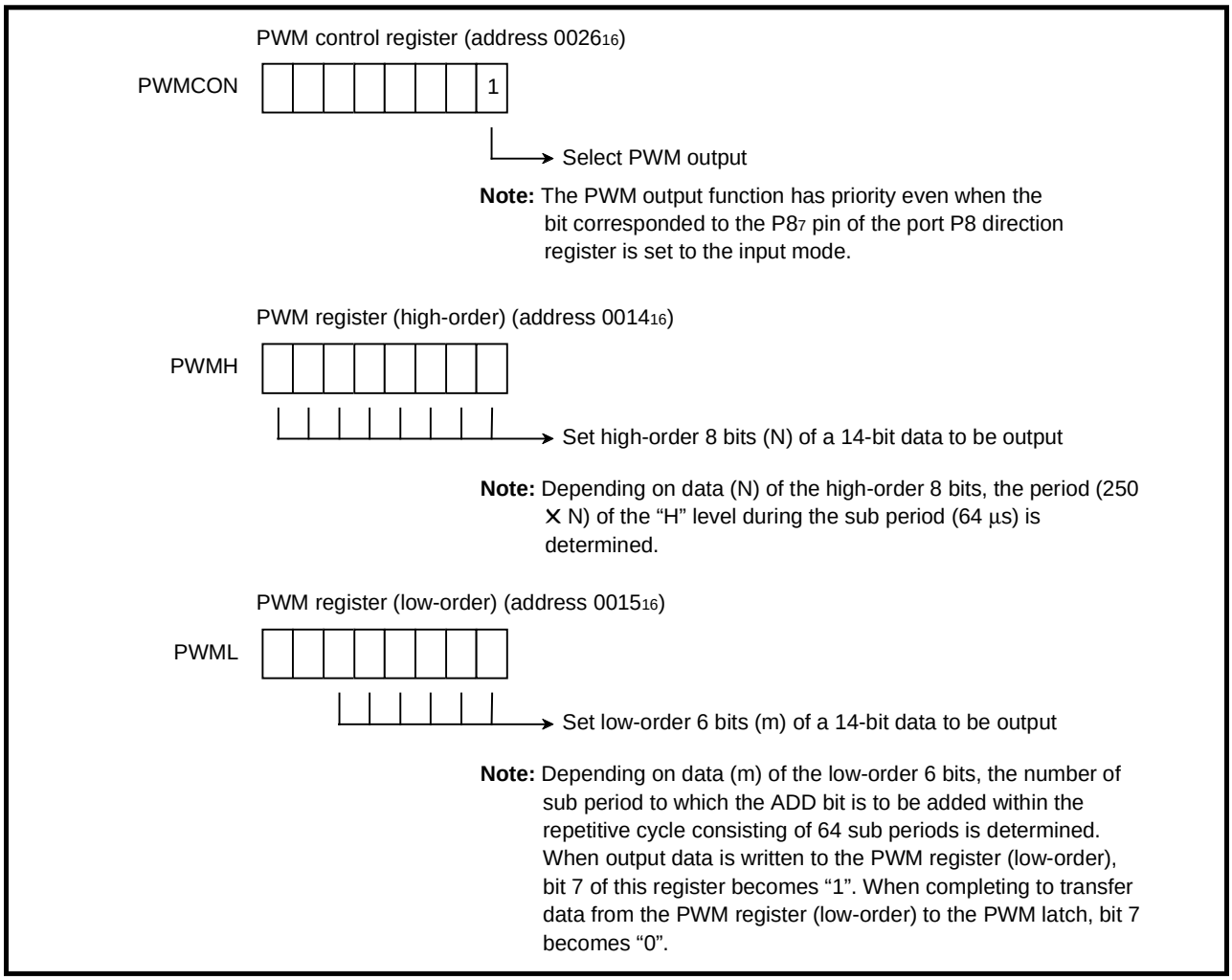


Fig. 2.6.6 Setting of relevant registers

Control procedure: PWM waveform is output to the external by setting relevant registers shown Figure 2.6.6. This PWM_0 output is integrated through the low pass filter and converted into DC signals for control of the VS tuner.
Figure 2.6.7 shows the control procedure.

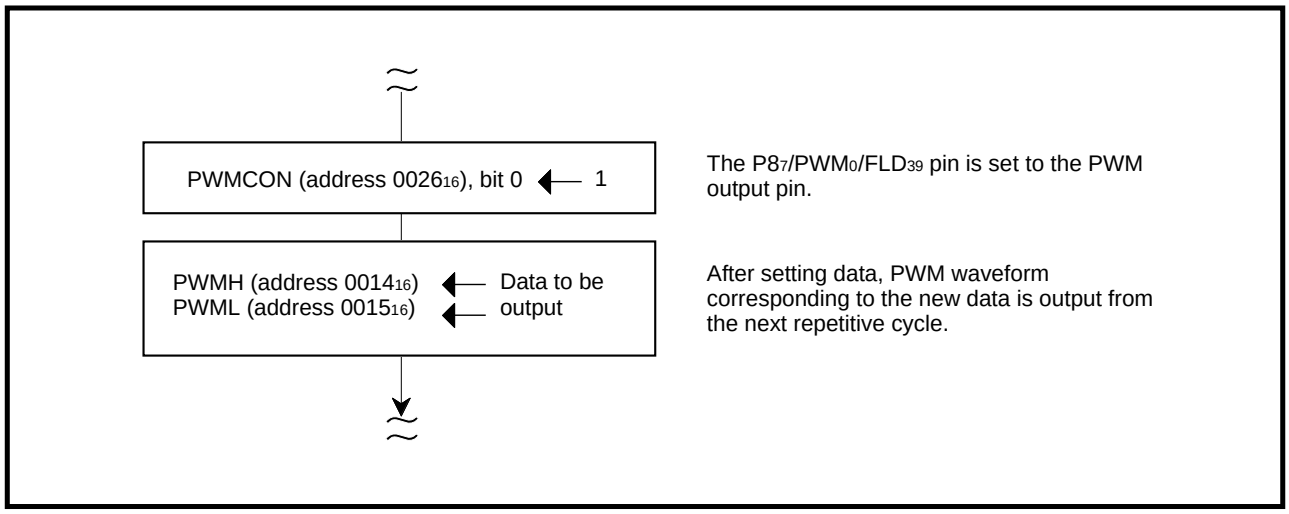


Fig. 2.6.7 Control procedure

2.6.4 Notes on use

- For PWM_0 output, "L" level is output first.
- After data is set to the PWM register (low-order) and the PWM register (high-order), PWM waveform corresponding to new data is output from next repetitive cycle.

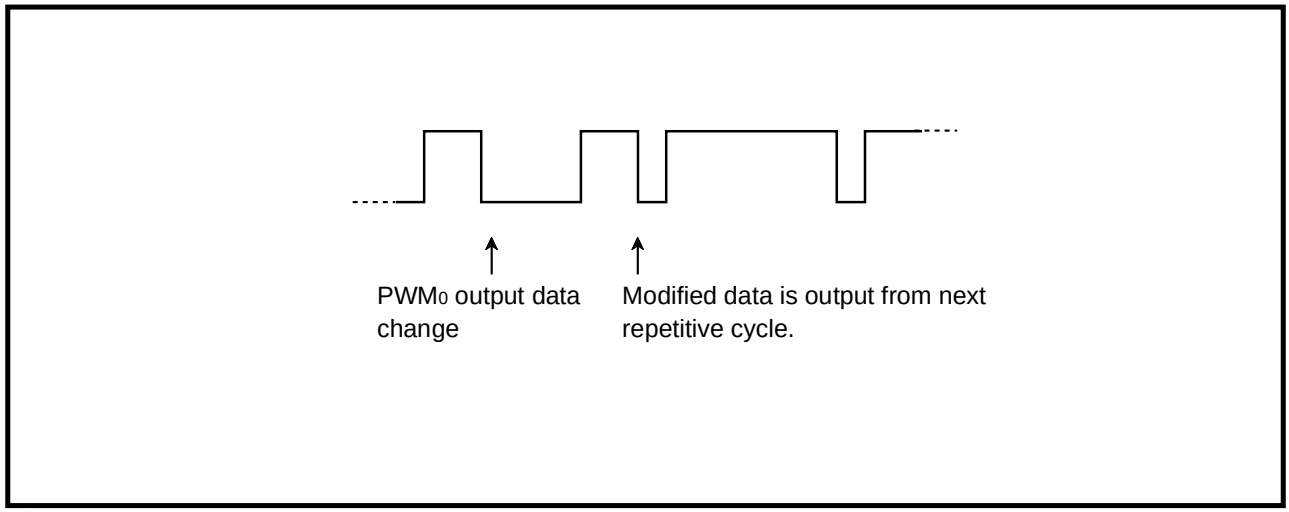


Fig. 2.6.8 PWM_0 output

2.7 Interrupt interval determination function

This paragraph describes the setting method of interrupt interval determination function relevant registers, notes etc.

2.7.1 Memory assignment

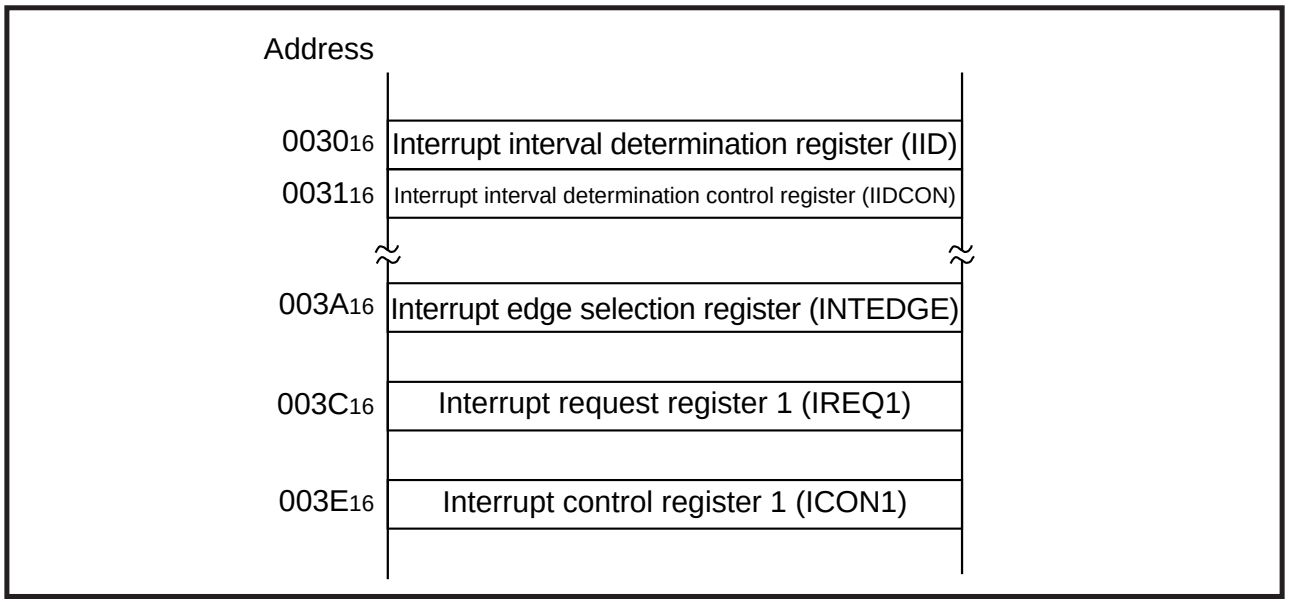


Fig. 2.7.1 Memory assignment of interrupt interval determination function relevant registers

2.7.2 Relevant registers

Interrupt interval determination register

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt interval determination register
(IID: address 30_{16})

b	Functions	At reset	R	W
0	• This register stores a value which is obtained by counting a following interval with the counter sampling clock. Rising interval Falling interval Both edges interval **(Note)** (Selected by interrupt edge selection register) • Read exclusive register	Undefined	○	×
1		Undefined	○	×
2		Undefined	○	×
3		Undefined	○	×
4		Undefined	○	×
5		Undefined	○	×
6		Undefined	○	×
7		Undefined	○	×

Note: When the noise filter sampling clock selection bits (bits 2, 3) of the interrupt interval determination control register is "00", the both-sided edge detection function cannot be used.

Fig. 2.7.2 Structure of interrupt interval determination register

Interrupt interval determination control register

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt interval determination control register
(IIDCON: address 31_{16})

b	Name	Functions	At reset	R	W
0	Interrupt interval determination circuit operating selection bit	0: Stopped 1: Operating	0	○	○
1	Counter sampling clock selection bit	0: $f(X_{IN})/128$ 1: $f(X_{IN})/256$	0	○	○
2	Noise filter sampling clock selection bits (INT_2)	b3 b2 0 0: Filter is not used. 0 1: $f(X_{IN})/32$ 1 0: $f(X_{IN})/64$ 1 1: $f(X_{IN})/128$	0	○	○
3			0	○	○
4	One-sided/both-sided edge detection selection bit	0: One-sided edge detection 1: Both-sided edge detection **(Note)**	0	○	○
5	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are "0".		0	○	×
6			0	○	×
7			0	○	×

Note: When the noise filter sampling clock selection bits (bits 2, 3) is "00", the both-sided edge detection function cannot be used.

Fig. 2.7.3 Structure of interrupt interval determination control register

Interrupt edge selection register

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt edge selection register
(INTEDGE : address $3A_{16}$)

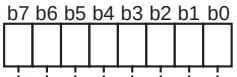
b	Name	Functions	At reset	R	W
0	INT_0 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
1	INT_1 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
2	INT_2 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
3	INT_3 interrupt edge selection bit **(Note)**	0 : Falling edge active 1 : Rising edge active	0	○	○
4	INT_4 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
5	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are "0".		0	○	×
6	$CNTR_0$ pin edge switch bit	0 : Rising edge count 1 : Falling edge count	0	○	○
7	$CNTR_1$ pin edge switch bit **(Note)**	0 : Rising edge count 1 : Falling edge count	0	○	○

Note: In the mask option type P, these bits are not available because $CNTR_1$ function and INT_3 function cannot be used.

Fig. 2.7.4 Structure of interrupt edge selection register

2.7 Interrupt interval determination function

Interrupt request register 1



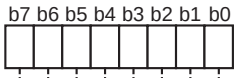
Interrupt request register 1
(IREQ1 : address $3C_{16}$)

b	Name	Functions	At reset	R	W
0	INT_0 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	INT_1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	INT_2 interrupt request bit Remote controller /counter overflow interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O1 interrupt request bit Serial I/O automatic transfer interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	Timer 2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Timer 3 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻

✻: "0" can be set by software, but "1" cannot be set.

Fig. 2.7.5 Structure of interrupt request register 1

Interrupt control register 1



Interrupt control register 1
(ICON1 : address $3E_{16}$)

b	Name	Functions	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	INT_2 interrupt enable bit Remote controller /counter overflow interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O1 interrupt enable bit Serial I/O automatic transfer interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Timer 2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	Timer 3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Fig. 2.7.6 Structure of interrupt control register 1

2.7 Interrupt interval determination function

2.7.3 Interrupt interval determination function application examples

(1) Reception of remote-control signal

Outline: Remote-control signal is read in by both of the interrupt interval determination function using a noise filter and a timer interrupt.

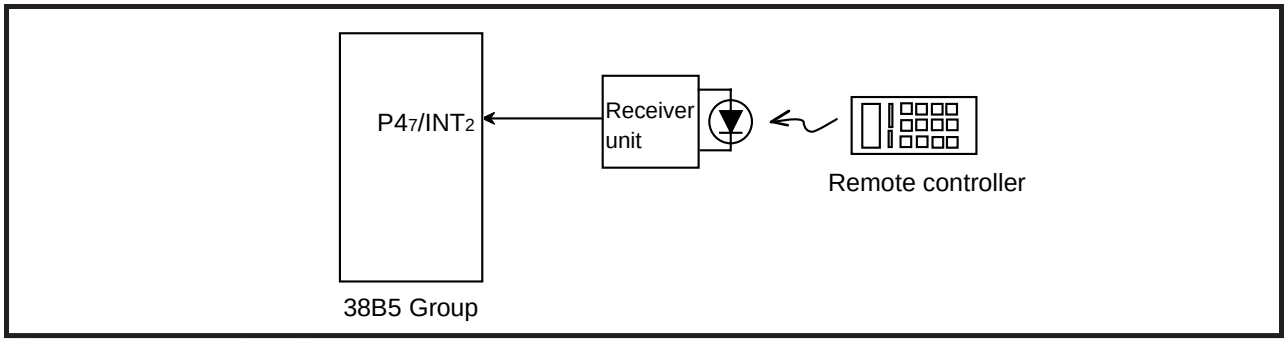


Fig. 2.7.7 Connection diagram

Specifications:
- Measurement of one-sided edge interval
- Use of noise filter
- Check of remote control interrupt request within the timer 2 interrupt (488 µs period) processing routine
- Operation at f(X_{IN}) = 4 MHz in high-speed mode

Figure 2.7.8 shows the function block diagram, and Figure 2.7.9 shows a timing chart of data determination.

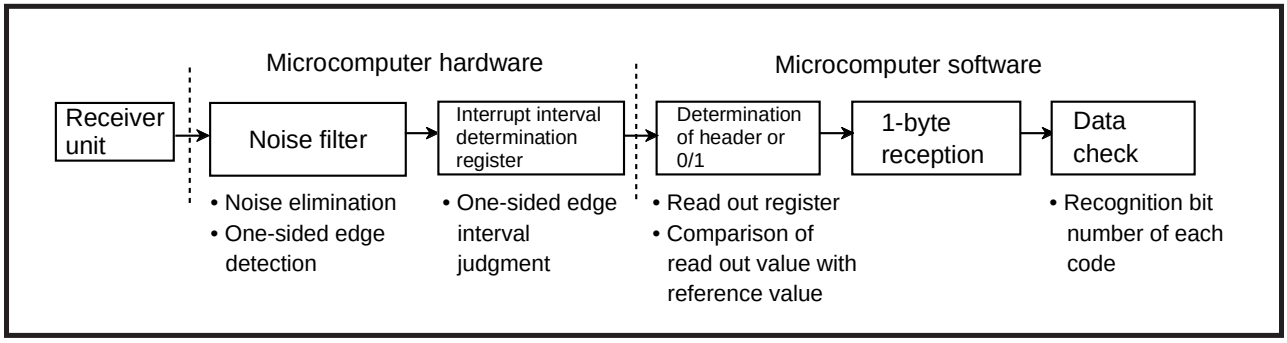


Fig. 2.7.8 Function block diagram

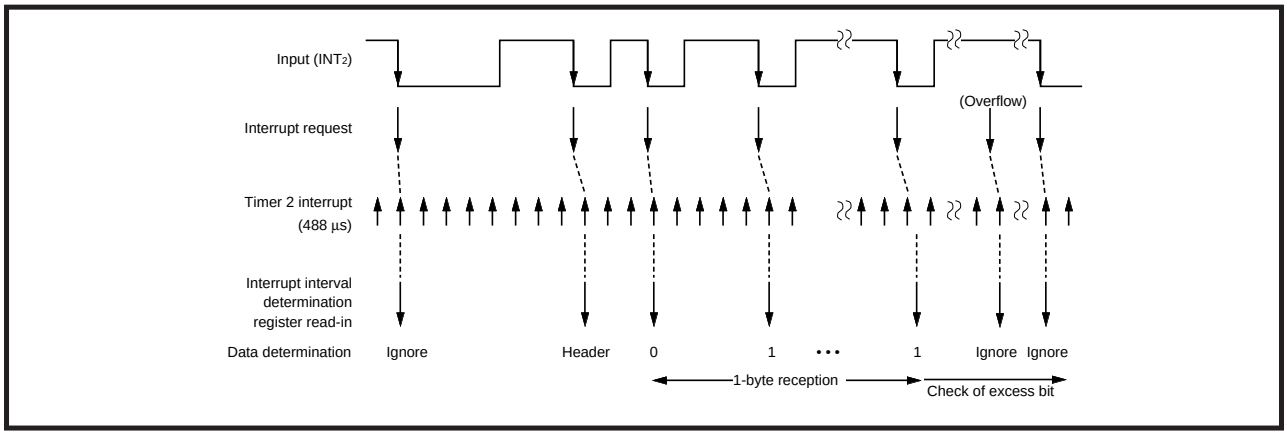


Fig. 2.7.9 Timing chart of data determination

Figure 2.7.10 shows the setting of relevant registers.

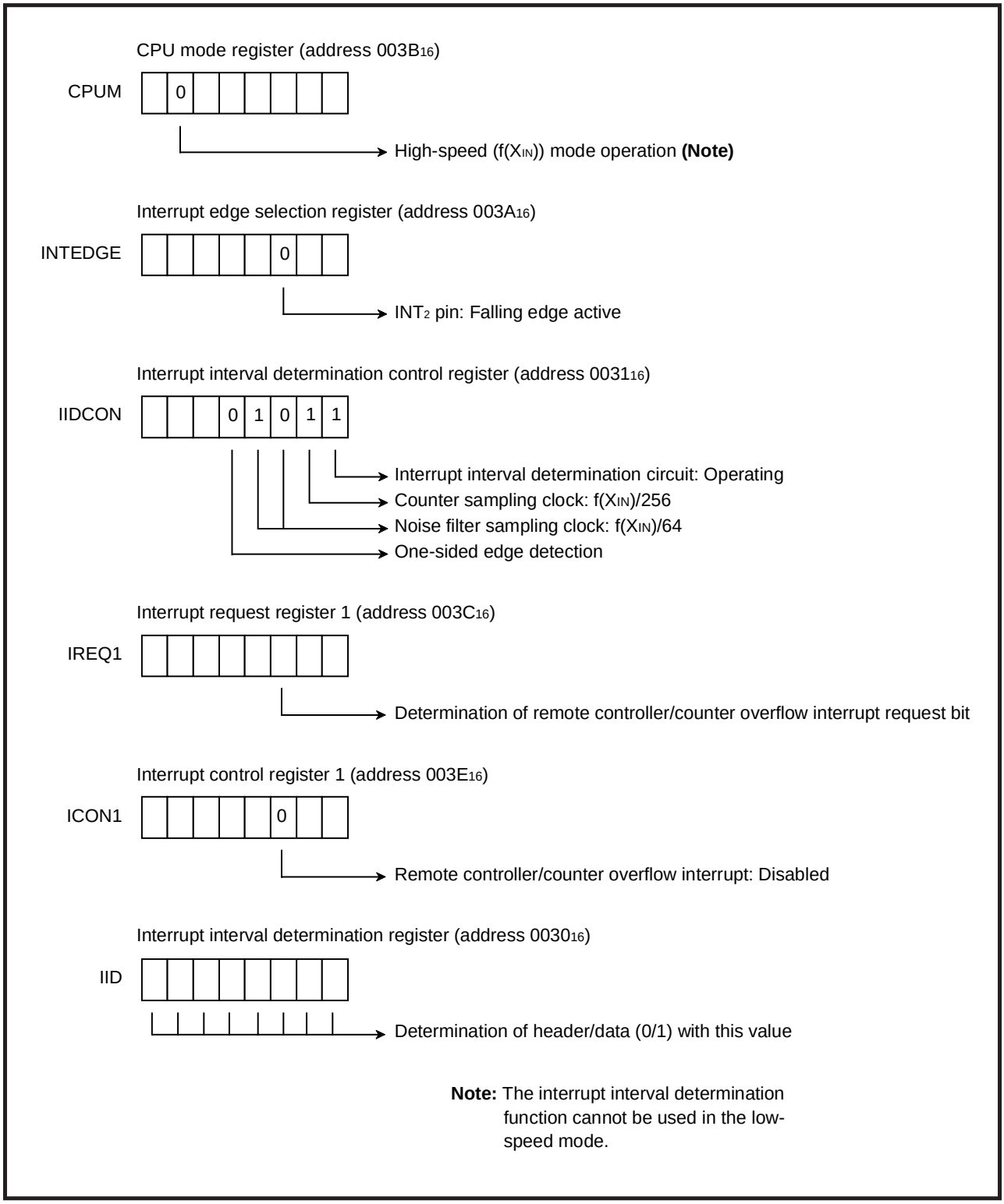


Fig. 2.7.10 Setting of relevant registers

2.7 Interrupt interval determination function

Control procedure: When the registers are set as shown Figure 2.7.10, remote-control signals are receivable. Figure 2.7.11 shows the control procedure, and Figure 2.7.12 shows the reception of remote-control data (timer 2 interrupt).

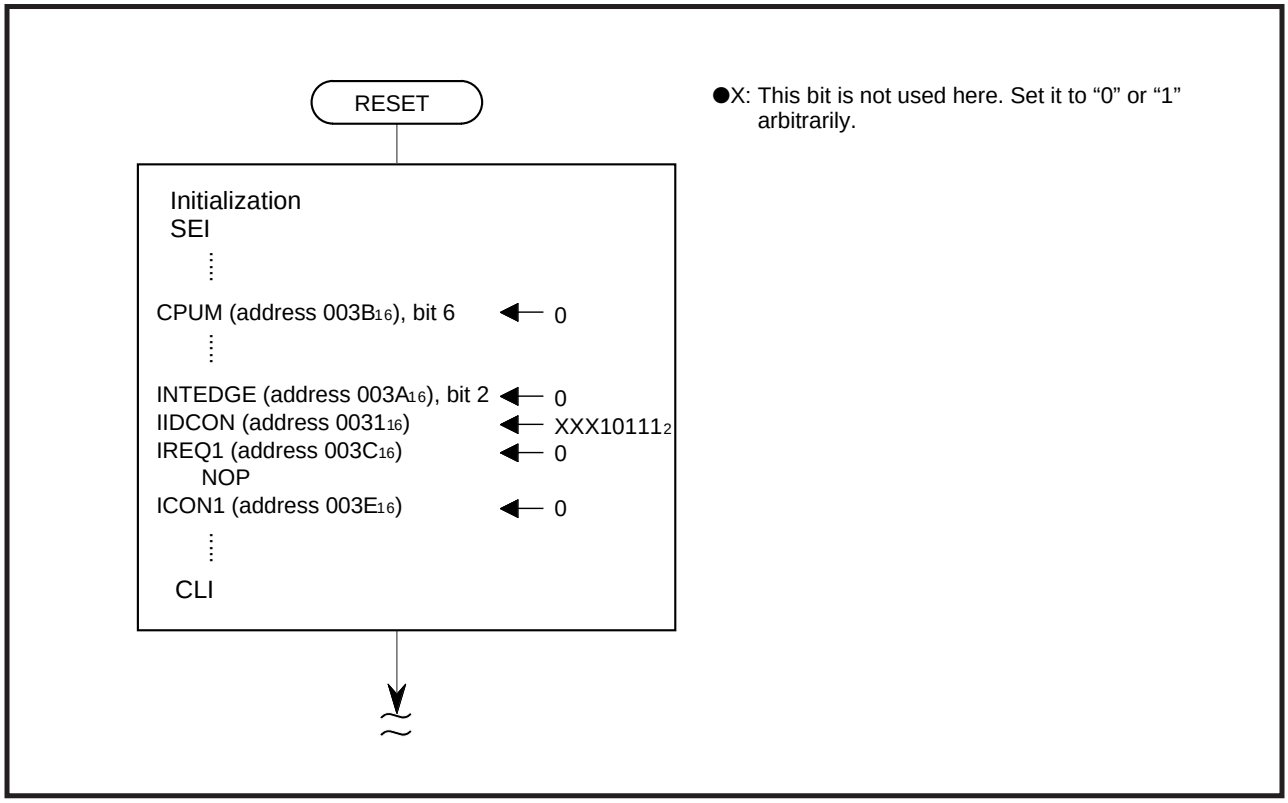


Fig. 2.7.11 Control procedure

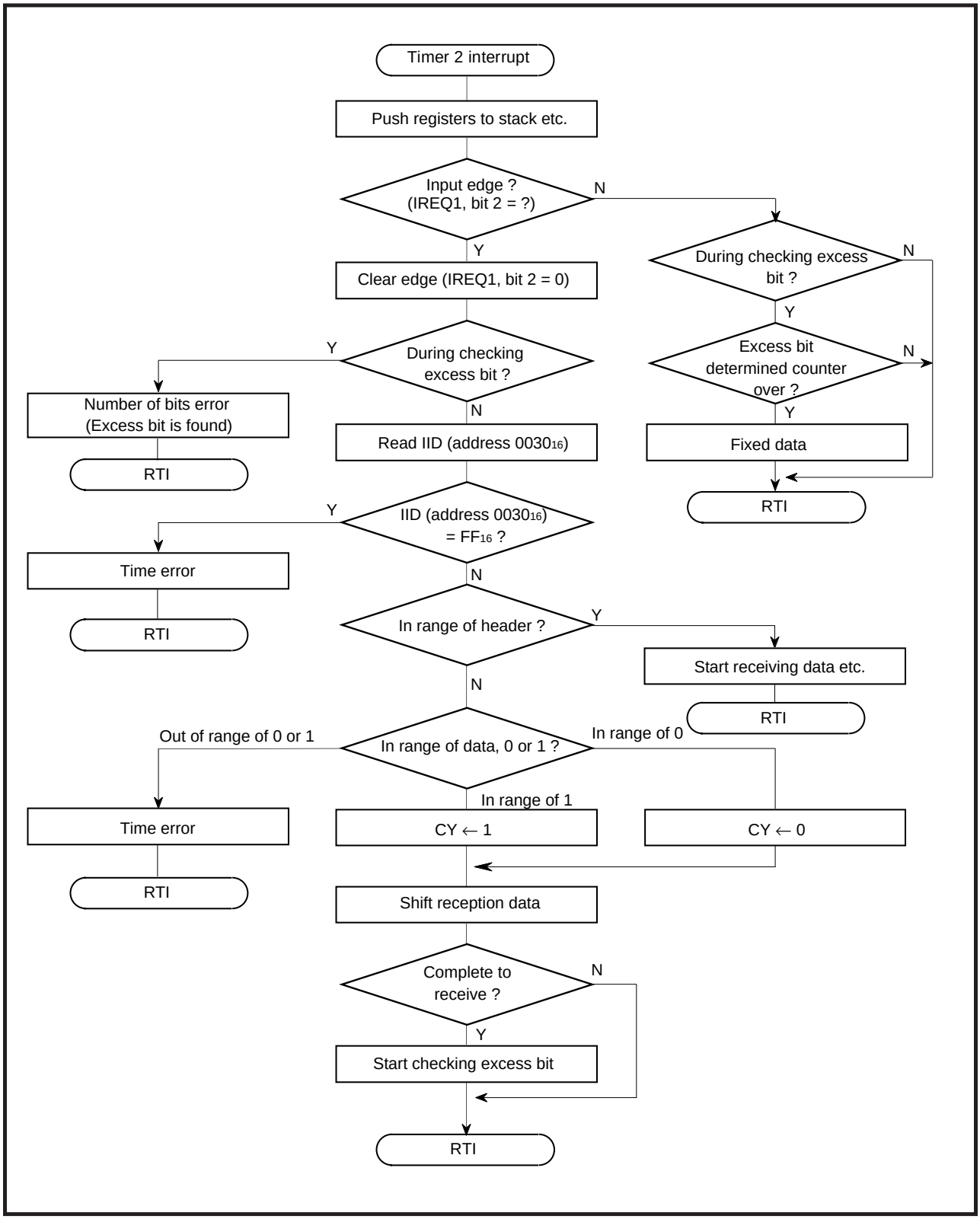


Fig. 2.7.12 Reception of remote-control data (timer 2 interrupt)

2.8 Watchdog timer

The watchdog timer is a 20-bit down-count counter consisting of a low-order 8 bits and a high-order 12 bits. "1" is subtracted from the watchdog timer each time a count source inputs.
This paragraph describes the setting method of watchdog timer relevant register, notes etc.

2.8.1 Memory assignment

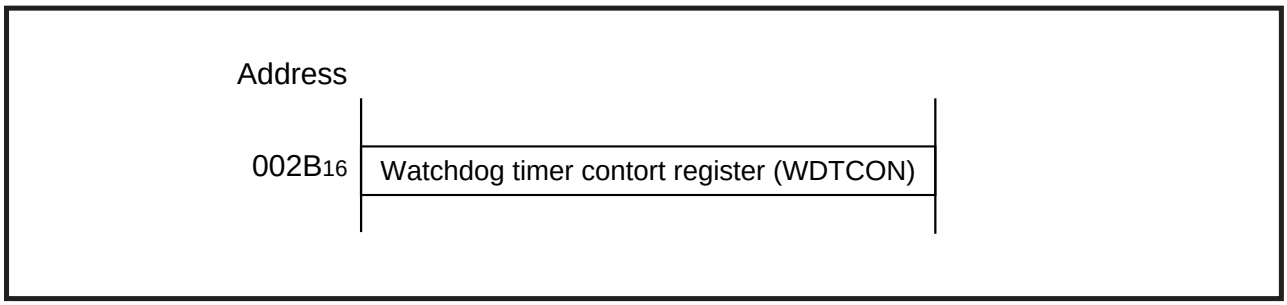


Fig. 2.8.1 Memory assignment of watchdog timer relevant register

2.8.2 Relevant register

The watchdog timer starts counting by writing an arbitrary value to the watchdog timer control register. Figure 2.8.2 shows the structure of the watchdog timer control register.

Watchdog timer control register

b7 b6 b5 b4 b3 b2 b1 b0

Watchdog timer control register
(WDTCON: address 2B16)

b	Name	Functions	At reset	R	W
0	Watchdog timer H (high-order 6 bits of reading exclusive)		1	○	×
1			1	○	×
2			1	○	×
3			1	○	×
4			1	○	×
5			1	○	×
6	STP instruction disable bit	0: STP instruction enabled 1: STP instruction disabled	0	○	○
7	Watchdog timer H count source selection bit	0: Watchdog timer L underflow 1: f(XIN)/16 or f(XCIN)/16	0	○	○

Fig. 2.8.2 Structure of watchdog timer control register

2.8.3 Watchdog timer application examples

Outline: When a program runs away, the watchdog timer makes the microcomputer return to the reset state.

Specifications: •When the watchdog timer H underflows, it is judged as incorrect program, and the microcomputer is returned to the reset state.
•Bit 7 of the watchdog timer control register is set to "0" at 1-cycle intervals in the main routine before underflow of the watchdog timer H. (Initialization of watchdog timer value)
•Use of watchdog timer L underflow as count source of watchdog timer H
•Setting of main clock division ratio to f(X_{IN}) (high-speed mode)

Figure 2.8.3 shows the connection of watchdog timer and the setting of the division ratio.
Figure 2.8.4 shows the setting of relevant registers.

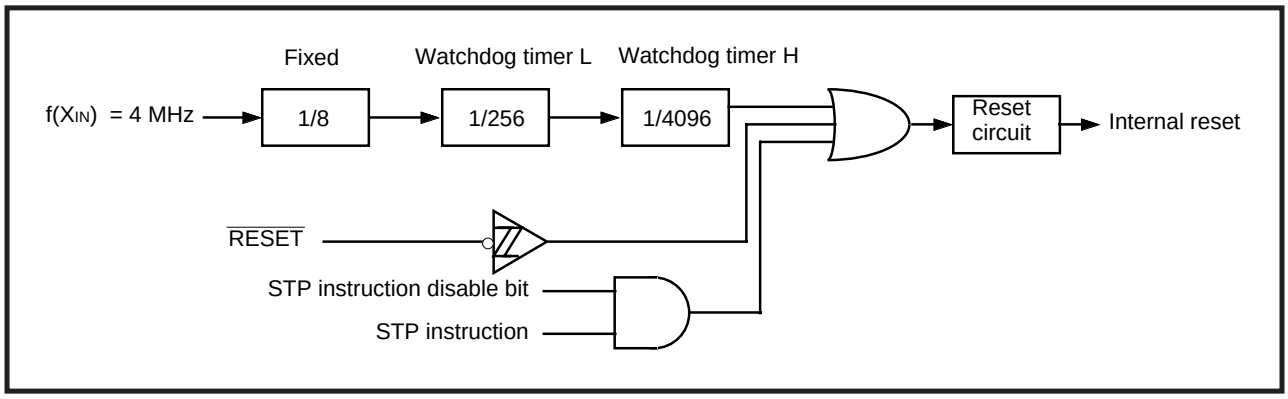


Fig. 2.8.3 Connection of watchdog timer and setting of division ratio

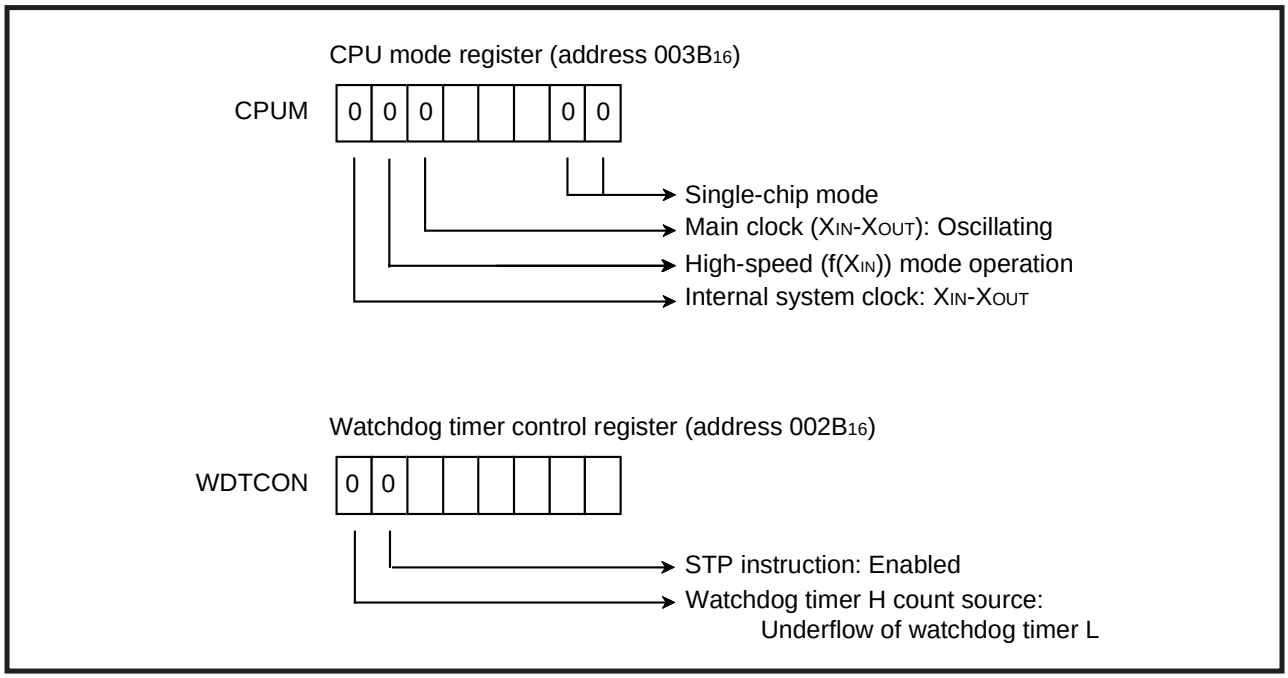


Fig. 2.8.4 Setting of relevant registers

Figure 2.8.5 shows the control procedure.

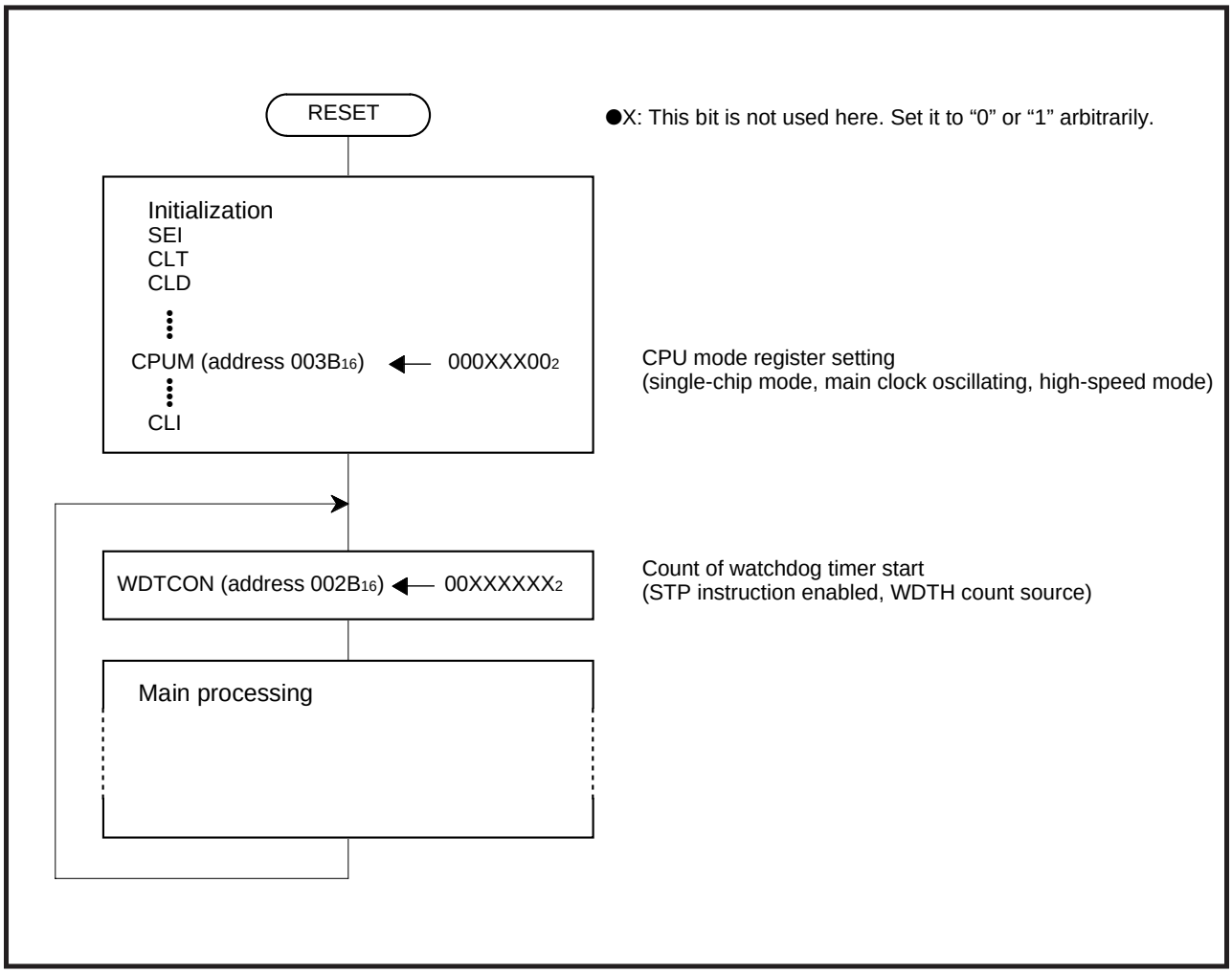


Fig. 2.8.5 Control procedure

2.8.4 Notes on use

- The watchdog timer continues to count even while waiting for stop release. Accordingly, make sure that watchdog timer does not underflow during this term by writing to the watchdog timer control register (address $002B_{16}$) once before executing the STP instruction, etc.
- Once a “1” is written to the STP instruction disable bit (bit 6) of the watchdog timer control register (address $002B_{16}$), it cannot be programmed to “0” again. This bit becomes “0” after reset.

2.9 Buzzer output circuit

The output frequency can be selected from 1 kHz, 2 kHz, or 4 kHz (at f(X_{IN}) = 4.19 MHz), and the output port can be selected between either the B_{UZ01} pin or the B_{UZ02} pin.
This paragraph describes the setting method of buzzer output circuit relevant register, notes etc.

2.9.1 Memory assignment

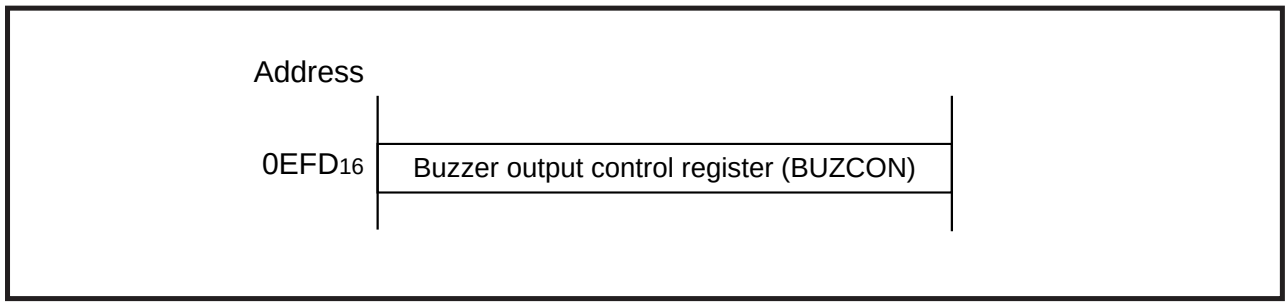


Fig. 2.9.1 Memory assignment of buzzer output circuit relevant register

2.9.2 Relevant register

The buzzer output circuit starts outputting a buzzer by setting the buzzer output ON/OFF bit (bit 4) of the buzzer output control register.
Figure 2.9.2 shows the structure of the buzzer output control register.

Buzzer output control register

b7 b6 b5 b4 b3 b2 b1 b0

Buzzer output control register
(BUZCON: address 0EFD16)

b	Name	Functions	At reset	R	W
0	Output frequency selection bits	b1b0 0 0: 1 kHz (f(X_{IN})/4096) 0 1: 2 kHz (f(X_{IN})/2048) 1 0: 4 kHz (f(X_{IN})/1024) 1 1: Not available	0	○	○
1			0	○	○
2	Output port selection bits	b3b2 0 0: $P2_0$ and $P4_3$ function as ordinary ports. 0 1: $P4_3/B_{UZ01}$ functions as a buzzer output. 1 0: $P2_0/B_{UZ02}/FLD_0$ functions as a buzzer output. 1 1: Not available	0	○	○
3			0	○	○
4	Buzzer output ON/OFF bit	0: Buzzer output OFF (“0” output) 1: Buzzer output ON	0	○	○
5	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	○	×
6			0	○	×
7			0	○	×

Fig. 2.9.2 Structure of buzzer output control register

2.9 Buzzer output circuit

2.9.3 Buzzer output circuit application examples

Outline: A buzzer output is performed by using the buzzer output circuit.

Specifications: •f(X_{IN}) = 4.19 MHz, buzzer output frequency = 4 kHz
•Buzzer output from B_{UZ01} pin

Figure 2.9.3 shows the connection of buzzer output circuit and the setting of the division ratio. Figure 2.9.4 shows the setting of relevant register. Figure 2.9.5 shows the control procedure.

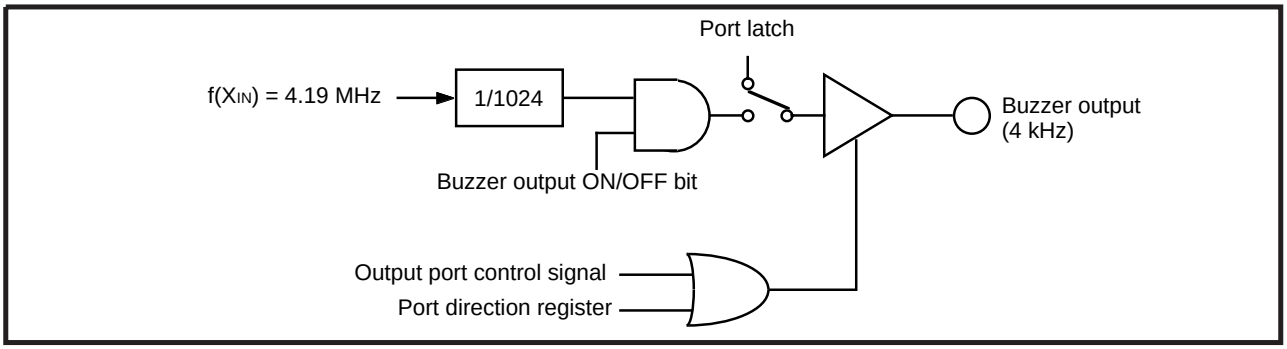


Fig. 2.9.3 Connection of buzzer output circuit and setting of division ratio

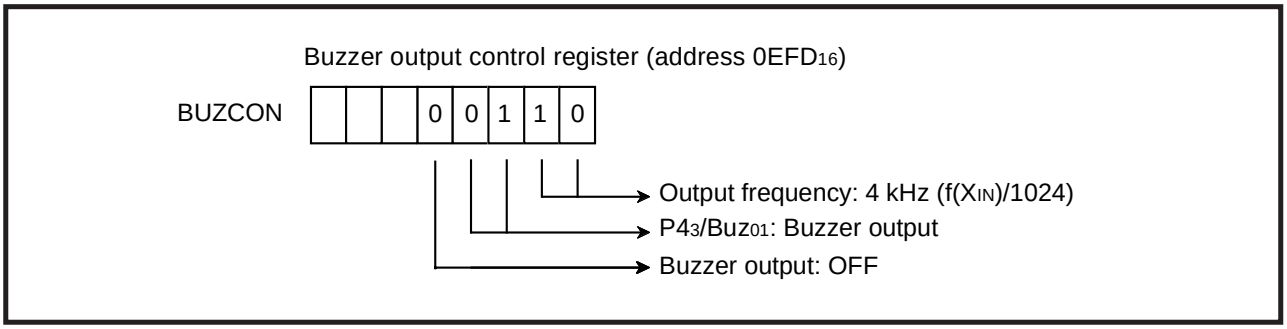


Fig. 2.9.4 Setting of relevant register

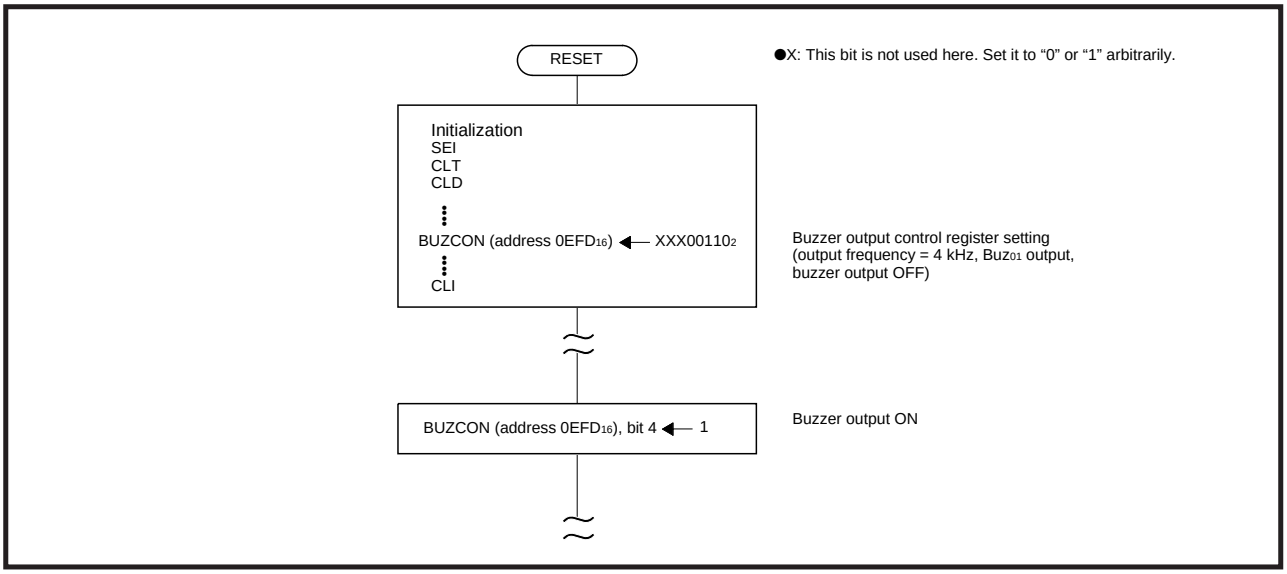


Fig. 2.9.5 Control procedure

2.10 Reset circuit

The reset state is caused by applying an “L” level to the $\overline{\text{RESET}}$ pin. After that, the reset state is released by applying an “H” level to the $\overline{\text{RESET}}$ pin, so that the program is executed in the middle-speed mode from the contents of the reset vector address.

2.10.1 Connection example of reset IC

Figure 2.10.1 shows the example of power-on reset circuit. Figure 2.10.2 shows the system example which switches to the RAM backup mode by detecting a drop of the system power source voltage with the INT interrupt.

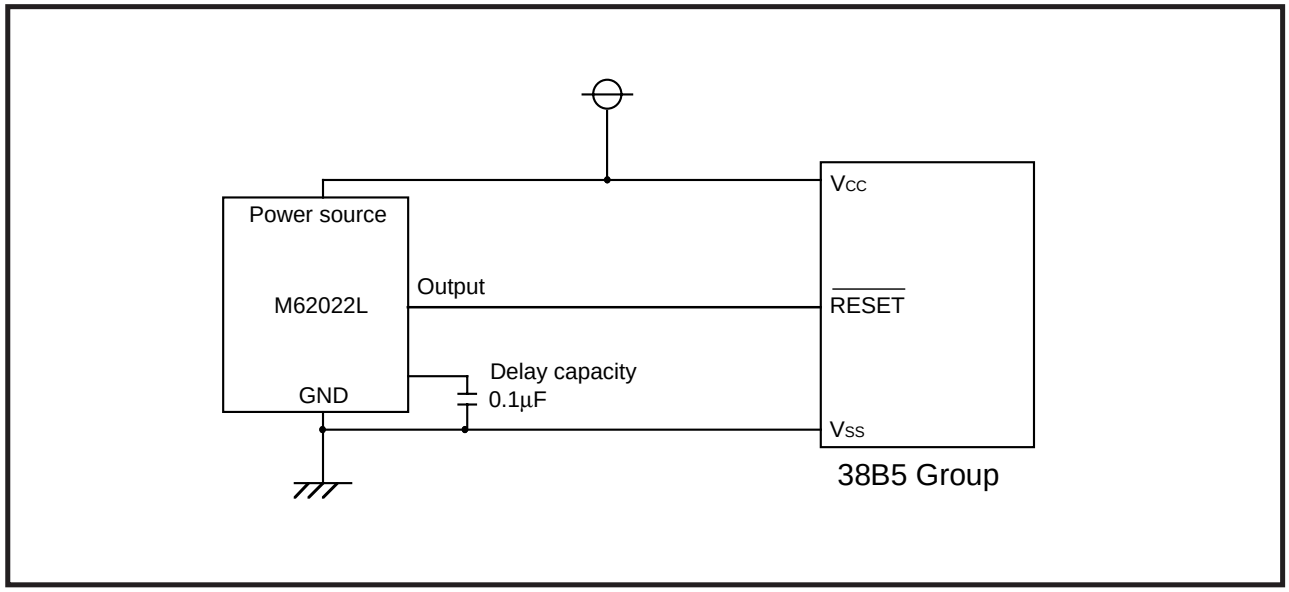


Fig. 2.10.1 Example of power-on reset circuit

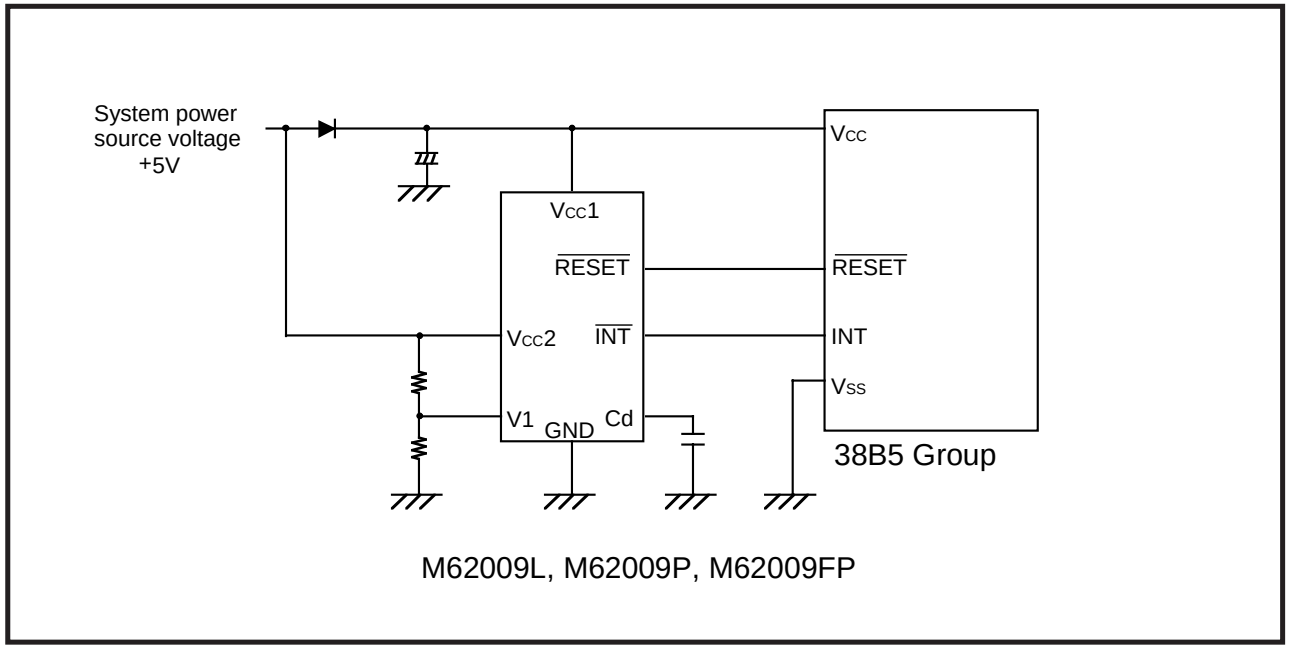


Fig. 2.10.2 RAM backup system example

2.10.2 Notes on use

(1) Reset input voltage control

Make sure that the reset input voltage is 0.5 V or less for Vcc of 2.7 V.
Perform switch to the high-speed mode when power source voltage is within 4.0 to 5.5 V.

(2) Countermeasure when $\overline{\text{RESET}}$ signal rise time is long

In case where the $\overline{\text{RESET}}$ signal rise time is long, connect a ceramic capacitor or others across the $\overline{\text{RESET}}$ pin and the Vss pin. And use a 1000 pF or more capacitor for high frequency use. When connecting the capacitor, note the following :

- Make the length of the wiring which is connected to a capacitor as short as possible.
- Be sure to verify the operation of application products on the user side.

● **Reason**

If the several nanosecond or several ten nanosecond impulse noise enters the $\overline{\text{RESET}}$ pin, it may cause a microcomputer failure.

2.11 Clock generating circuit

2.11.1 Relevant register

Figure 2.11.1 shows the structure of the CPU mode register.

CPU mode register

b7 b6 b5 b4 b3 b2 b1 b0

CPU mode register
(CPUM, CM: address $3B_{16}$)

b	Name	Functions	At reset	R	W
0	Processor mode bits	b1 b0 00 : Single-chip mode 01 : 10 : Not available 11 :	0	○	○
1			0	○	○
2	Stack page selection bit	0 : Page 0 1 : Page 1	0	○	○
3	XCOUT drivability selection bit	0: Low drive 1: High drive	1	○	○
4	Port Xc switch bit	0: I/O port function 1: XCIN-XCOUT oscillation function	0	○	○
5	Main clock (XIN-XOUT) stop bit	0: Oscillating 1: Stopped	0	○	○
6	Main clock division ratio selection bit	0: f(XIN) (high-speed mode) 1: f(XIN)/4 (middle-speed mode)	1	○	○
7	Internal system clock selection bit	0: XIN–XOUT selection (middle-/high-speed mode) 1: XCIN–XCOUT selection (low-speed mode)	0	○	○

Fig. 2.11.1 Structure of CPU mode register

2.11 Clock generating circuit

2.11.2 Clock generating circuit application examples

(1) Status transition during power failure

Outline: The clock is counted up every one second by using the timer interrupt during a power failure.

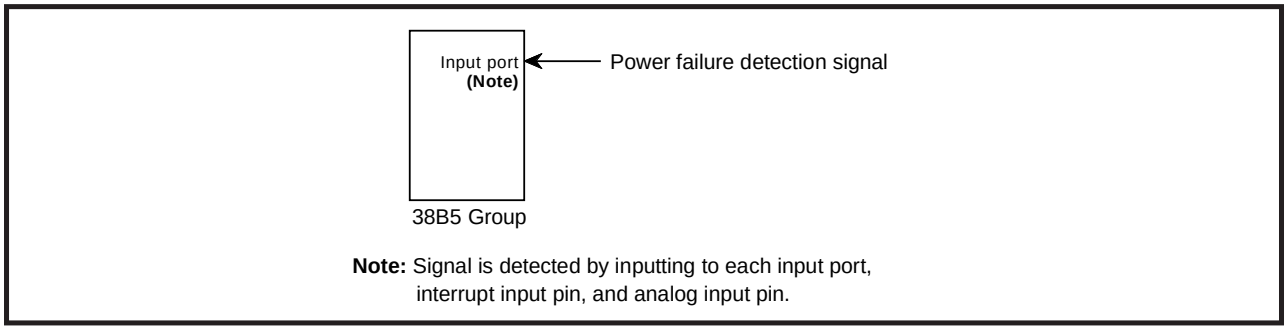


Fig. 2.11.2 Connection diagram

Specifications: •Reducing power dissipation as low as possible while maintaining clock function
•Clock: f(X_{IN}) = 4.19 MHz, f(X_{CIN}) = 32.768 kHz
•Port processing
Input port: Fixed to "H" or "L" level on the external
Output port: Fixed to output level that does not cause current flow to the external
(Example) When a circuit turns on LED at "L" output level, fix the output level to "H".
I/O port: Input port → Fixed to "H" or "L" level on the external
Output port → Output of data that does not consume current
V_{REF}: Stop to supply to reference voltage input pin by external circuit

Figure 2.11.3 shows the status transition diagram during power failure and Figure 2.11.4 shows the setting of relevant registers.

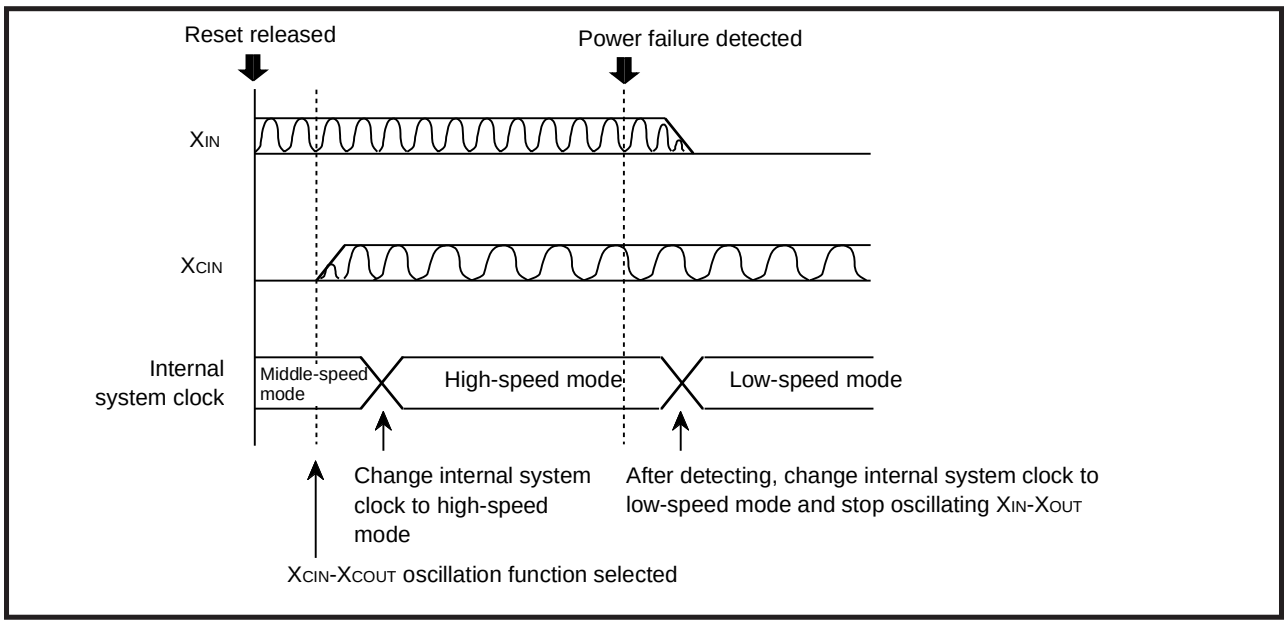
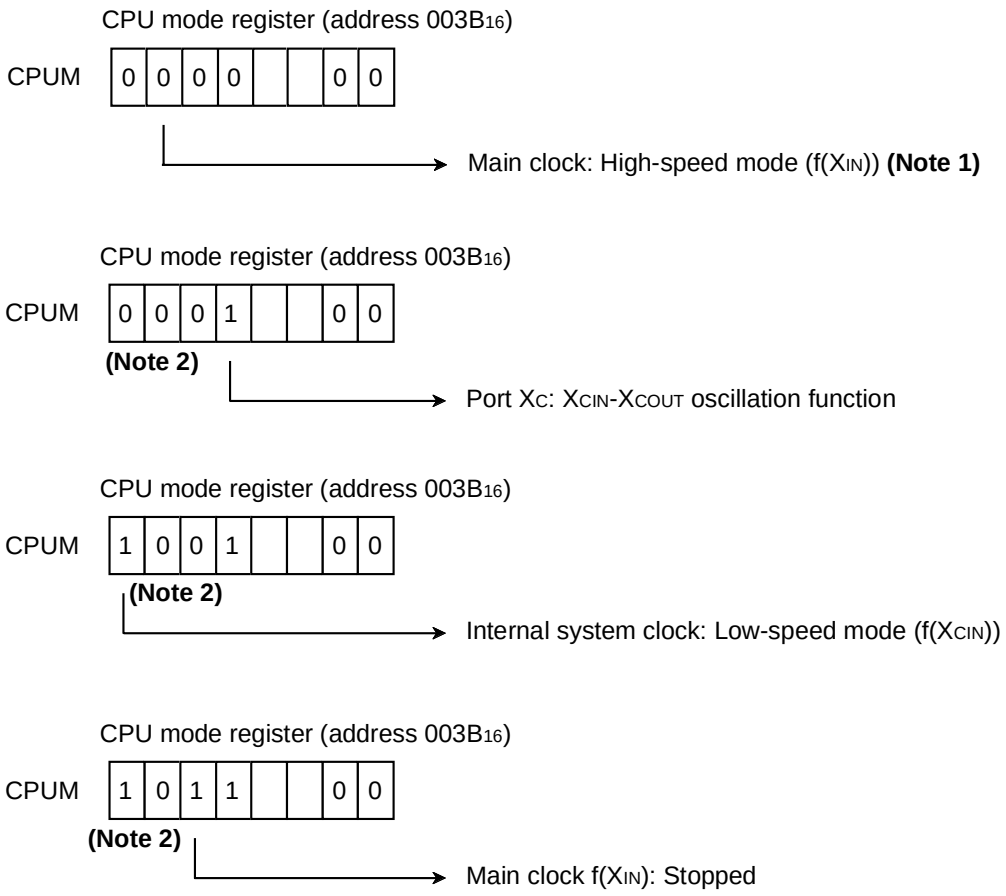


Fig. 2.11.3 Status transition diagram during power failure



Notes 1: This setting is necessary only when selecting the high-speed mode.
2: When selecting the middle-speed mode, bit 6 is “1”.

Fig. 2.11.4 Setting of relevant registers

2.11 Clock generating circuit

Control procedure: Set the relevant registers in the order shown below to prepare for a power failure.

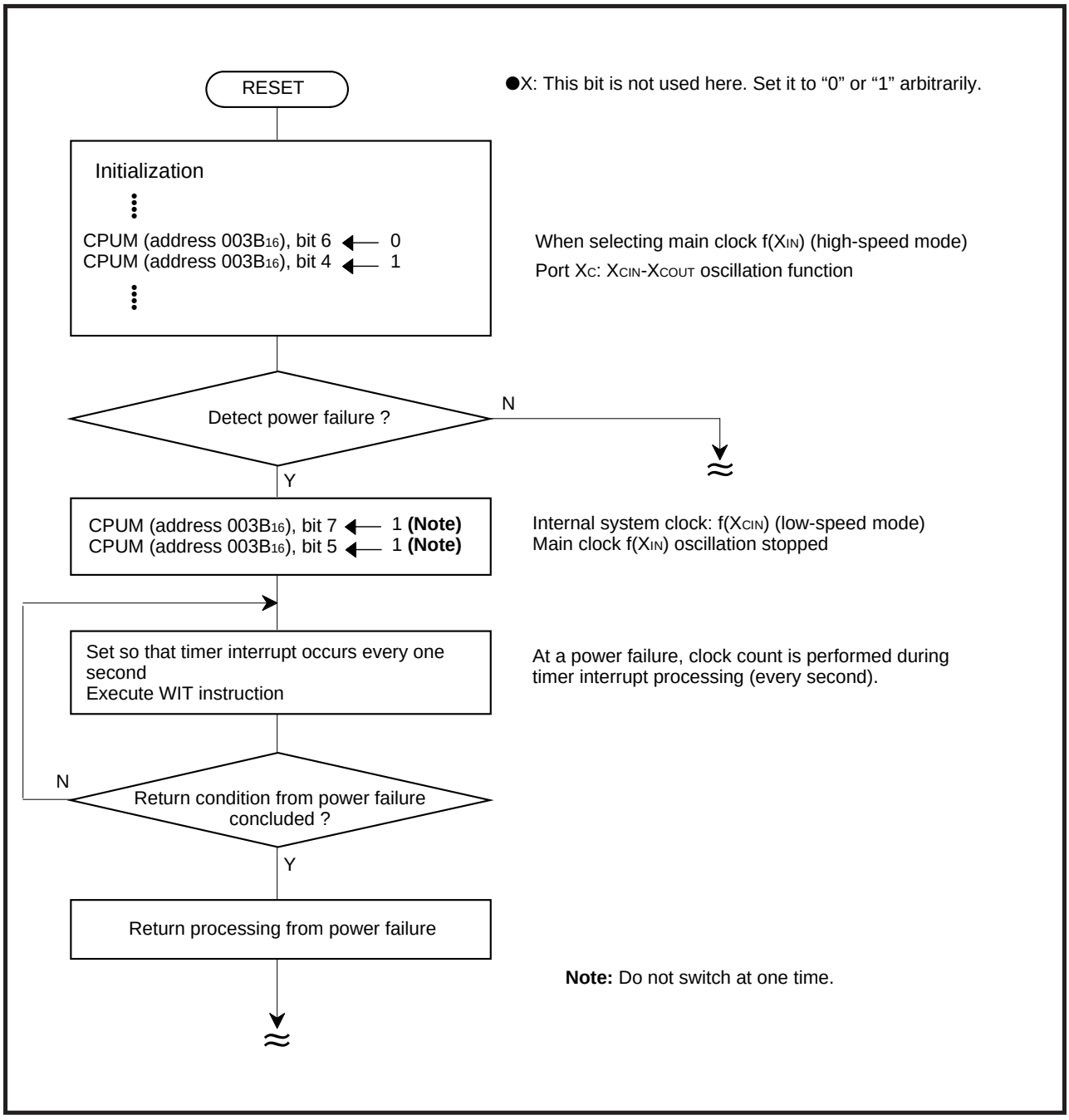


Fig. 2.11.5 Control procedure

(2) Counting without clock error during power failure

Outline: It keeps counting without clock error during a power failure.

Specifications: •Reducing power consumption as low as possible while maintaining clock function
•Clock: f(X_{IN}) = 4.19 MHz
•Sub clock: f(X_{CIN}) = 32.768 kHz
•Use of Timer 3 interrupt

For the peripheral circuit and the status transition during a power failure, refer to **"Figures 2.11.2 and 2.11.3"**.
Figure 2.11.6 shows the structure of clock counter, Figures 2.11.7 and 2.11.8 show the setting of relevant registers.

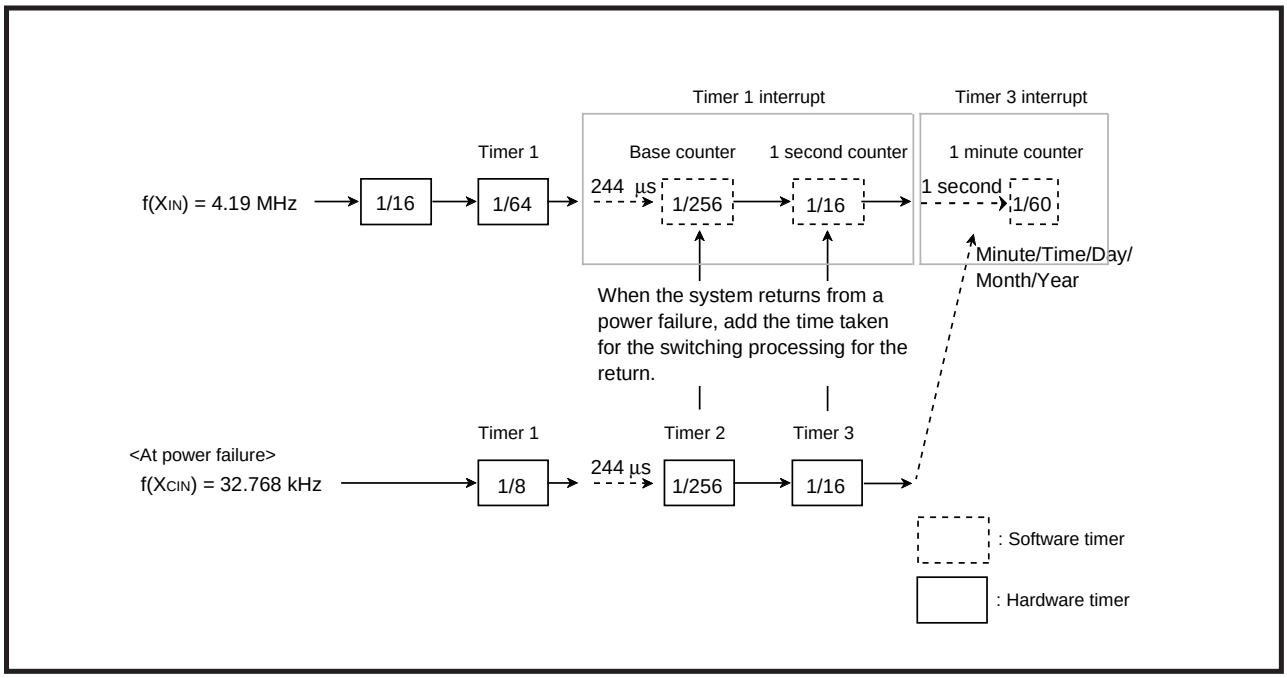


Fig. 2.11.6 Structure of clock counter

2.11 Clock generating circuit

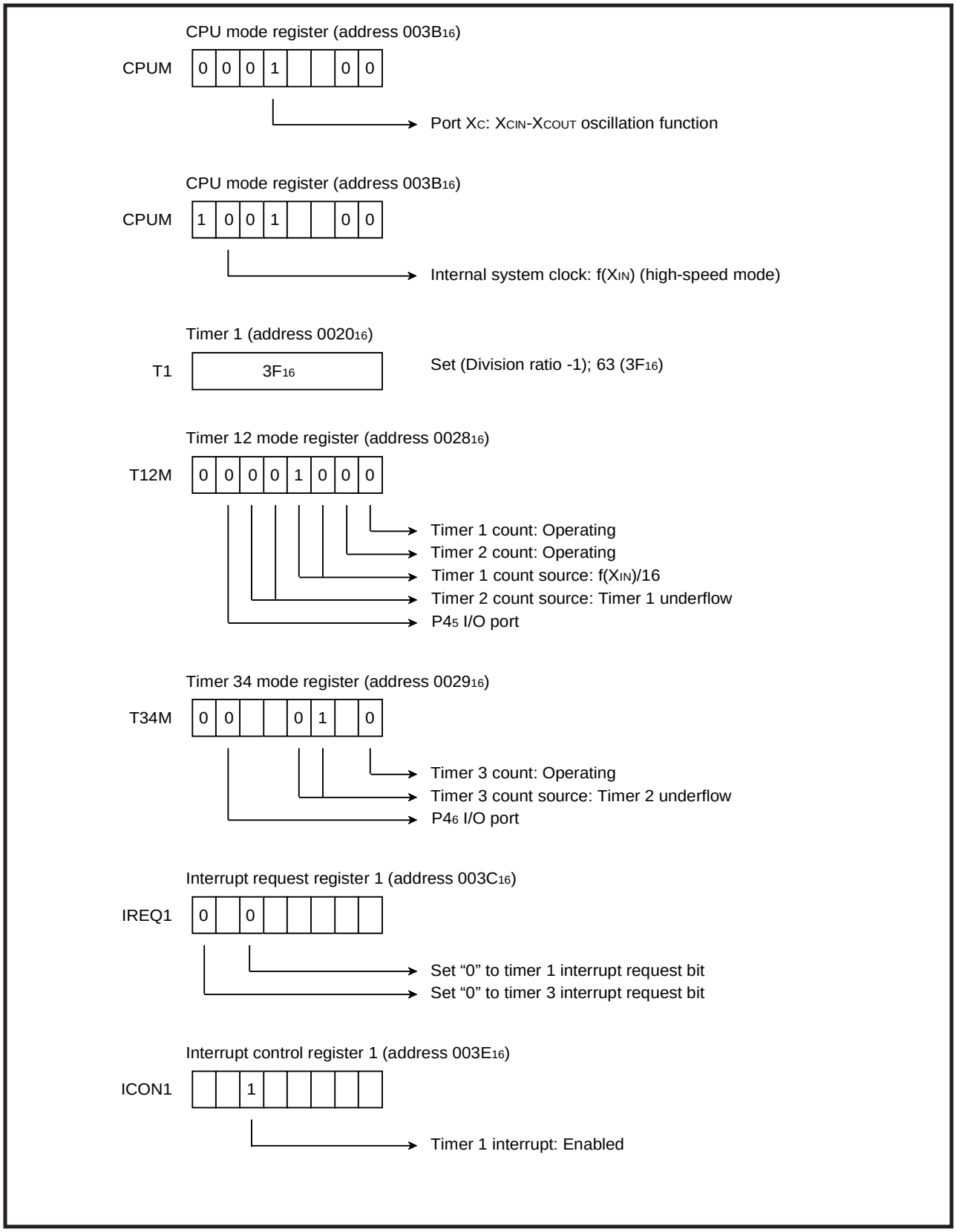


Fig. 2.11.7 Initial setting of relevant registers

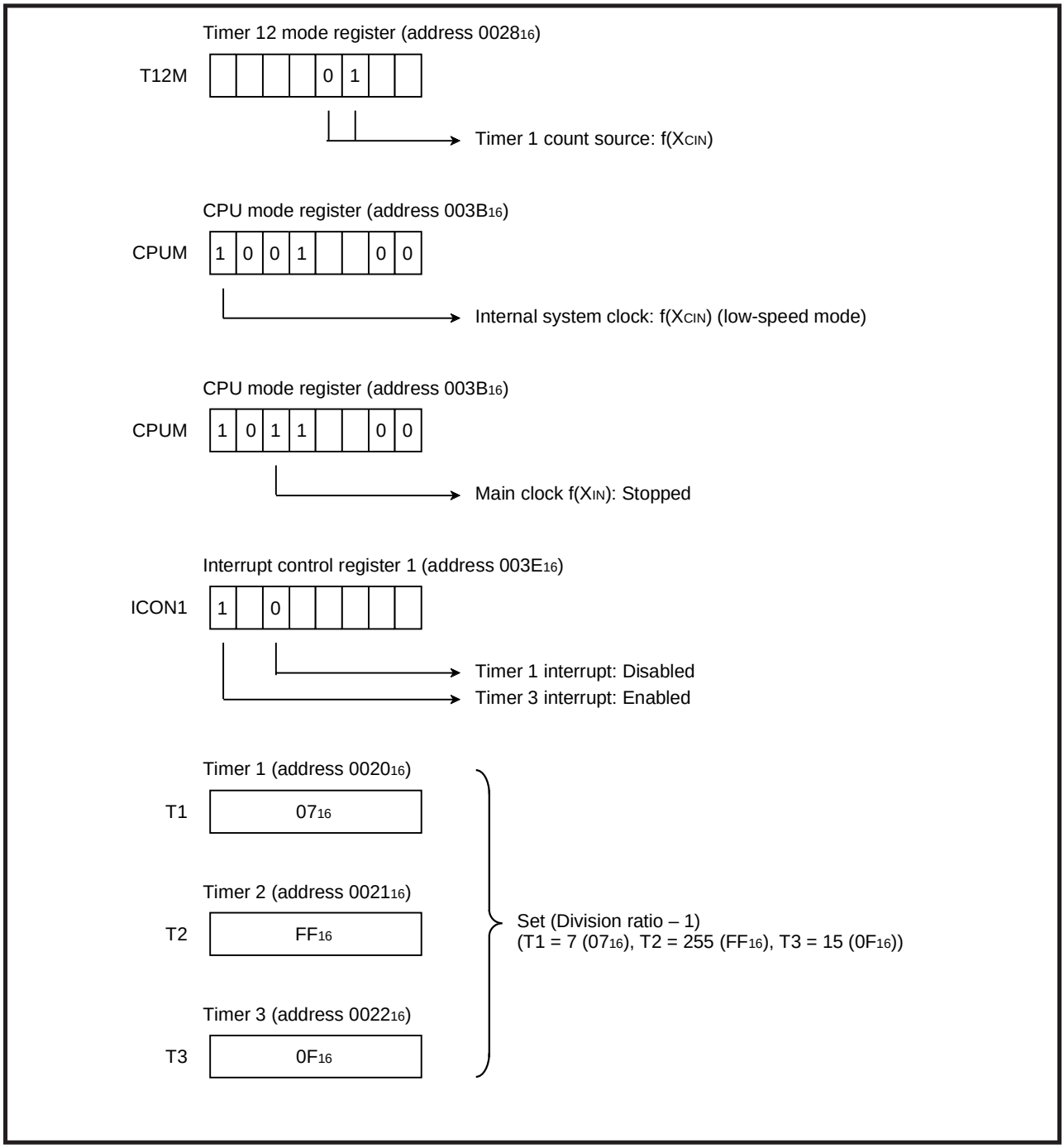


Fig. 2.11.8 Setting of relevant registers after detecting power failure

Control procedure: Set the relevant registers in the order shown below to prepare for a power failure.

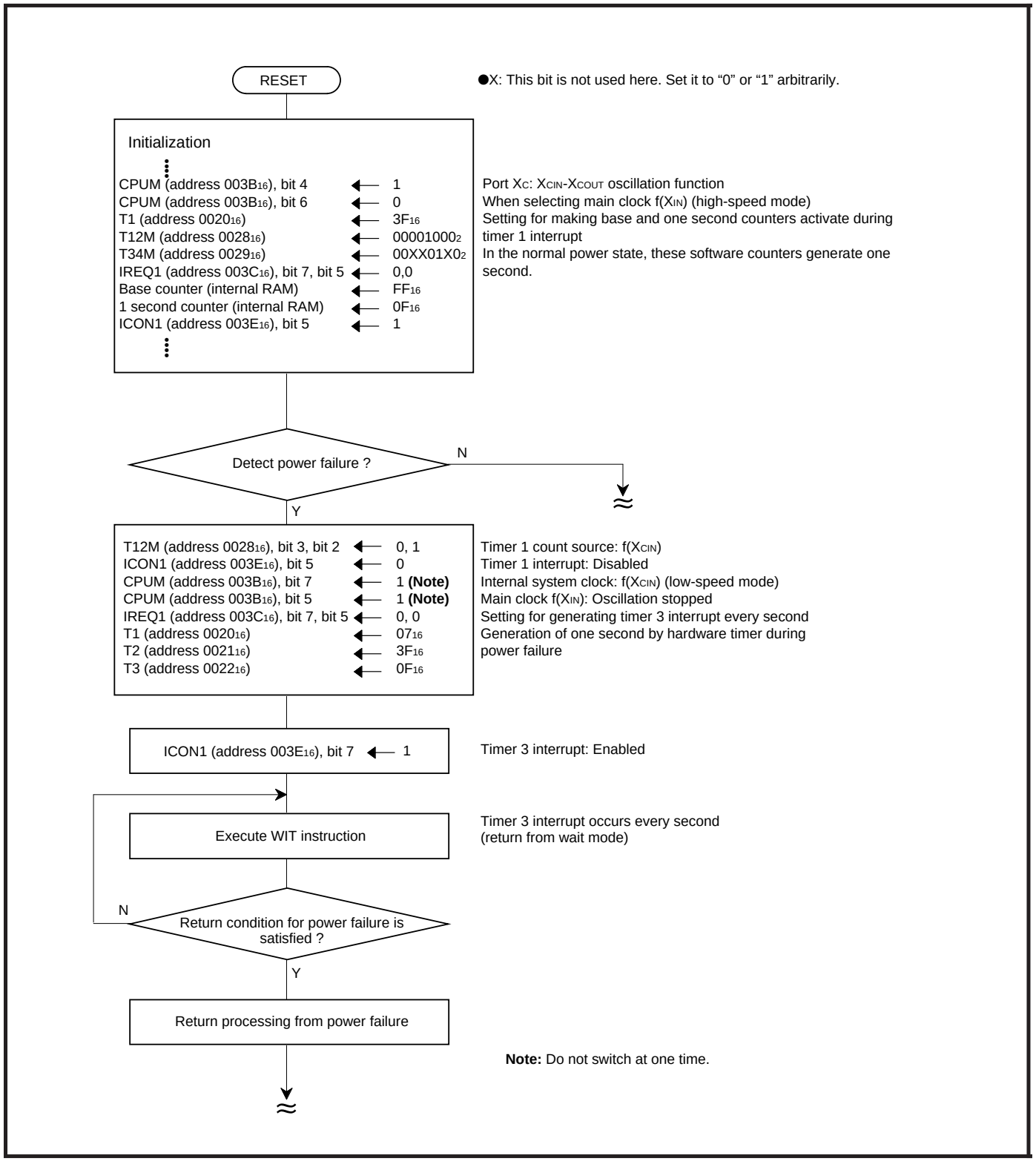
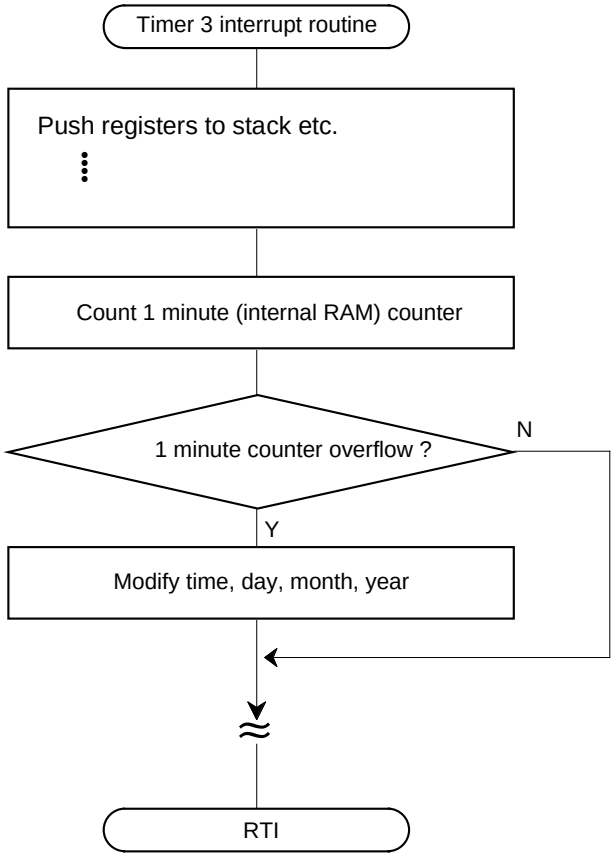


Fig. 2.11.9 Control procedure



MEMORANDUM

CHAPTER 3
APPENDIX

3.1 Electrical characteristics

3.1.1 Absolute maximum ratings

Table 3.1.1 Absolute maximum ratings

Symbol	Parameter		Conditions	Ratings	Unit
V_{CC}	Power source voltage		All voltages are based on V_{SS}. Output transistors are cut off.	–0.3 to 7.0	V
V_{EE}	Pull-down power source voltage			V_{CC} – 45 to V_{CC} +0.3	V
V_I	Input voltage	$P4_7$, $P5_0$–$P5_7$, $P6_1$–$P6_5$, $P7_0$–$P7_7$, $P8_4$–$P8_7$, $P9_0$, $P9_1$		–0.3 to V_{CC} +0.3	V
V_I	Input voltage	$P4_0$–$P4_6$, $P6_0$		–0.3 to 13	V
V_I	Input voltage	$P0_0$–$P0_7$, $P2_0$–$P2_7$, $P8_0$–$P8_3$		V_{CC} – 45 to V_{CC} +0.3	V
V_I	Input voltage	$\overline{RESET}$, X_{IN}		–0.3 to V_{CC} +0.3	V
V_I	Input voltage	X_{CIN}		–0.3 to V_{CC} +0.3	V
V_O	Output voltage	$P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P8_0$–$P8_3$		V_{CC} – 45 to V_{CC} +0.3	V
V_O	Output voltage	$P5_0$–$P5_7$, $P6_1$–$P6_5$, $P7_0$–$P7_7$, $P8_4$–$P8_7$, $P9_0$, $P9_1$, X_{OUT}, X_{COUT}		–0.3 to V_{CC} +0.3	V
V_O	Output voltage	$P4_0$–$P4_6$, $P6_0$		–0.3 to 13	V
P_d	Power dissipation		Ta = –20 to 65 °C	800	mW
			Ta = 65 to 85 °C	800 – 12.5 × (Ta – 65)	mW
T_{opr}	Operating temperature			–20 to 85	°C
T_{stg}	Storage temperature			–40 to 125	°C

3.1.2 Recommended operating conditions

Table 3.1.2 Recommended operating conditions (1)

(Vcc = 4.0 to 5.5 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
V_{CC}	Power source voltage	In high-speed mode	4.0	5.0	5.5	V
		In middle-/low-speed mode	2.7	5.0	5.5	V
V_{SS}	Power source voltage			0		V
V_{EE}	Pull-down power source voltage		V_{CC}–43		V_{CC}	V
V_{REF}	Analog reference voltage (when A-D converter is used)		2.0		V_{CC}	V
AVSS	Analog power source voltage			0		V
V_{IA}	Analog input voltage	AN_0–AN_{11}	0		V_{CC}	V
V_{IH}	"H" input voltage	$P4_0$–$P4_7$, $P5_0$–$P5_7$, $P6_0$–$P6_5$, $P7_0$–$P7_7$, $P9_0$, $P9_1$	0.75V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage	$P8_4$–$P8_7$	0.4V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage	$P0_0$–$P0_7$	0.8V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage	$P2_0$–$P2_7$, $P8_0$–$P8_3$	0.52V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage	$\overline{RESET}$	0.8V_{CC}		V_{CC}	V
V_{IH}	"H" input voltage	X_{IN}, X_{CIN}	0.8V_{CC}		V_{CC}	V
V_{IL}	"L" input voltage	$P4_0$–$P4_7$, $P5_0$–$P5_7$, $P6_0$–$P6_5$, $P7_0$–$P7_7$, $P9_0$, $P9_1$	0		0.25V_{CC}	V
V_{IL}	"L" input voltage	$P8_4$–$P8_7$	0		0.16V_{CC}	V
V_{IL}	"L" input voltage	$P0_0$–$P0_7$, $P2_0$–$P2_7$, $P8_0$–$P8_3$	0		0.2V_{CC}	V
V_{IL}	"L" input voltage	$\overline{RESET}$	0		0.2V_{CC}	V
V_{IL}	"L" input voltage	X_{IN}, X_{CIN}	0		0.2V_{CC}	V
$\Sigma I_{OH(peak)}$	H" total peak output current **(Note 1)** $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P8_0$–$P8_3$				–240	mA
$\Sigma I_{OH(peak)}$	"H" total peak output current **(Note 1)** $P5_0$–$P5_7$, $P6_1$–$P6_5$, $P7_0$–$P7_7$, $P9_0$, $P9_1$				–60	mA
$\Sigma I_{OL(peak)}$	"L" total peak output current **(Note 1)** $P5_0$–$P5_7$, $P6_0$–$P6_5$, $P7_0$–$P7_7$, $P9_0$, $P9_1$				100	mA
$\Sigma I_{OL(peak)}$	"L" total peak output current **(Note 1)** $P4_0$–$P4_6$, $P8_4$–$P8_7$				60	mA
$\Sigma I_{OH(avg)}$	"H" total average output current **(Note 1)** $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P8_0$–$P8_7$				–120	mA
$\Sigma I_{OH(avg)}$	"H" total average output current **(Note 1)** $P5_0$–$P5_7$, $P6_1$–$P6_5$, $P7_0$–$P7_7$, $P9_0$, $P9_1$				–30	mA
$\Sigma I_{OL(avg)}$	"L" total average output current **(Note 1)** $P5_0$–$P5_7$, $P6_0$–$P6_5$, $P7_0$–$P7_7$, $P9_0$, $P9_1$				50	mA
$\Sigma I_{OL(avg)}$	"L" total average output current **(Note 1)** $P4_0$–$P4_6$, $P8_4$–$P8_7$				30	mA
$I_{OH(peak)}$	"H" peak output current **(Note 2)** $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P8_0$–$P8_3$				–40	mA
$I_{OH(peak)}$	"H" peak output current **(Note 2)** $P5_0$–$P5_7$, $P6_1$–$P6_5$, $P7_0$–$P7_7$, $P8_4$–$P8_7$, $P9_0$, $P9_1$				–10	mA
$I_{OL(peak)}$	"L" peak output current **(Note 2)** $P5_0$–$P5_7$, $P6_1$–$P6_5$, $P7_0$–$P7_7$, $P8_4$–$P8_7$, $P9_0$, $P9_1$				10	mA
$I_{OL(peak)}$	"L" peak output current **(Note 2)** $P4_0$–$P4_6$, $P6_0$				30	mA
$I_{OH(avg)}$	"H" average output current **(Note 3)** $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P8_0$–$P8_3$				–18	mA
$I_{OH(avg)}$	"H" average output current **(Note 3)** $P5_0$–$P5_7$, $P6_0$–$P6_5$, $P7_0$–$P7_7$, $P8_4$–$P8_7$, $P9_0$, $P9_1$				–5	mA
$I_{OL(avg)}$	"L" average output current **(Note 3)** $P5_0$–$P5_7$, $P6_1$–$P6_5$, $P7_0$–$P7_7$, $P8_4$–$P8_7$, $P9_0$, $P9_1$				5	mA
$I_{OL(avg)}$	"L" average output current **(Note 3)** $P4_0$–$P4_6$, $P6_0$				15	mA

Notes 1: The total output current is the sum of all the currents flowing through all the applicable ports. The total average current is an average value measured over 100 ms. The total peak current is the peak value of all the currents.
2: The peak output current is the peak current flowing in each port.
3: The average output current I_{OL} (avg), I_{OH}(avg) in an average value measured over 100 ms.

APPENDIX

3.1 Electrical characteristics

Table 3.1.3 Recommended operating conditions (2)

(V_{CC} = 4.0 to 5.5 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
f(CNTR0) f(CNTR1)	Clock input frequency for timers 2, 4, and X (duty cycle 50 %)			250	kHz
f(XIN)	Main clock input oscillation frequency **(Note 1)**			4.2	MHz
f(XCIN)	Sub-clock input oscillation frequency **(Notes 1, 2)**		32.768	50	kHz

Notes 1: When the oscillation frequency has a duty cycle of 50%.
2: When using the microcomputer in low-speed mode, set the sub-clock input oscillation frequency on condition that f(X_{CIN}) < f(X_{IN})/3.

3.1.3 Electrical characteristics

Table 3.1.4 Electrical characteristics (1)

(V_{CC} = 4.0 to 5.5 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min.	Typ.	Max.	
V_{OH}	"H" output voltage	$P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P8_0$–$P8_3$	I_{OH} = –18 mA	V_{CC}–2.0			V
V_{OH}	"H" output voltage	$P5_0$–$P5_7$, $P6_0$–$P6_5$, $P7_0$–$P7_7$, $P8_4$–$P8_7$, $P9_0$, $P9_1$	I_{OH} = –10 mA	V_{CC}–2.0			V
V_{OL}	"L" output voltage	$P5_0$–$P5_7$, $P6_1$–$P6_5$, $P8_4$–$P8_7$, $P9_0$, $P9_1$	I_{OL} = 10 mA			2.0	V
V_{OL}	"L" output voltage	$P4_0$–$P4_6$, $P6_0$	I_{OL} = 15 mA		0.6	2.0	V
V_{T+}–V_{T-}	Hysteresis	$P4_0$–$P4_2$, $P4_5$–$P4_7$, P5, $P6_0$, $P6_1$, $P6_4$ **(Note 1)**			0.4		V
V_{T+}–V_{T-}	Hysteresis	$\overline{RESET}$, X_{IN}			0.5		V
V_{T+}–V_{T-}	Hysteresis	X_{CIN}			0.5		V
I_{IH}	"H" input current	$P4_7$, $P5_0$–$P5_7$, $P6_1$–$P6_5$, $P7_0$–$P7_7$, $P8_4$–$P8_7$	V_I = V_{CC}			5.0	µA
I_{IH}	"H" input current	$P4_0$–$P4_6$, $P6_0$	V_I = 12 V			10.0	µA
I_{IH}	"H" input current	$P0_0$–$P0_7$, $P2_0$–$P2_7$, $P8_0$–$P8_3$ **(Note 2)**	V_I = V_{CC}			5.0	µA
I_{IH}	"H" input current	$\overline{RESET}$, X_{CIN}	V_I = V_{CC}			5.0	µA
I_{IH}	"H" input current	X_{IN}	V_I = V_{CC}		4.0		µA
I_{IL}	"L" input current	$P4_0$–$P4_7$, $P6_0$	V_I = V_{SS}			–5.0	µA
I_{IL}	"L" input current	$P5_0$–$P5_7$, $P6_1$–$P6_5$, $P7_0$–$P7_7$, $P8_4$–$P8_7$, $P9_0$, $P9_1$	V_I = V_{SS} Pull-up "off"			–5.0	µA
			V_{CC} = 5 V, V_I = V_{SS} Pull-up "on"	–30	–70	–140	µA
			V_{CC} = 3 V, V_I = V_{SS} Pull-up "on"	–6.0	–25	–45	µA
I_{IL}	"L" input current	$P0_0$–$P0_7$, $P2_0$–$P2_7$, $P8_0$–$P8_3$ **(Note 2)**	V_I = V_{SS}			–5.0	µA
I_{IL}	"L" input current	$\overline{RESET}$, X_{CIN}	V_I = V_{SS}			–5.0	µA
I_{IL}	"L" input current	X_{IN}	V_I = V_{SS}		–4.0		µA
I_{LOAD}	Output load current	$P0_0$–$P0_7$, $P1_0$–$P1_7$, $P3_0$–$P3_7$	V_{EE} = V_{CC}–43 V, V_{OL} = V_{CC} Output transistors "off"	300	600	900	µA
I_{LEAK}	Output leak current	$P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P8_0$–$P8_3$	V_{EE} = V_{CC}–43 V, V_{OL} = V_{CC}–43 V Output transistors "off"			–10	µA
I_{READH}	"H" read current	$P0_0$–$P0_7$, $P2_0$–$P2_7$, $P8_0$–$P8_3$	V_I = 5 V		1		µA
V_{RAM}			When clock is stopped	2		5.5	V

Notes 1: $P4_2$, $P4_5$, $P4_6$, and $P6_0$ of the mask option type P do not have hysteresis characteristics.
2: Except when reading ports P0, P2, or P8.

Table 3.1.5 Electrical characteristics (2)

(V_{CC} = 4.0 to 5.5 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
I_{CC}	Power source current	High-speed mode $f(X_{IN})$ = 4.2 MHz $f(X_{CIN})$ = 32 kHz Output transistors “off”		7.5	15	mA
		High-speed mode $f(X_{IN})$ = 4.2 MHz (in WIT state) $f(X_{CIN})$ = 32 kHz Output transistors “off”		1		mA
		Middle-speed mode $f(X_{IN})$ = 4.2 MHz $f(X_{CIN})$ = stopped Output transistors “off”		3		mA
		Middle-speed mode $f(X_{IN})$ = 4.2 MHz (in WIT state) $f(X_{CIN})$ = stopped Output transistors “off”		1		mA
		Low-speed mode $f(X_{IN})$ = stopped $f(X_{CIN})$ = 32 kHz Low-power dissipation mode (CM_3 = 0) Output transistors “off”		60	200	µA
		Low-speed mode $f(X_{IN})$ = stopped $f(X_{CIN})$ = 32 kHz (in WIT state) Low-power dissipation mode (CM_3 = 0) Output transistors “off”		20	40	µA
		Increment when A-D conversion is executed		0.6		mA
		All oscillation stopped (in STP state) Output transistors “off” Ta = 25 °C		0.1	1	µA
		Ta = 85 °C			10	µA

3.1.4 A-D converter characteristics

Table 3.1.6 A-D converter characteristics

(V_{CC} = 4.0 to 5.5V, V_{SS} = 0 V, Ta = –20 to 85 °C, f(X_{IN}) = 250 kHz to 4.2 MHz in high-speed mode, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
—	Resolution				10	Bits
—	Absolute accuracy (excluding quantization error)	V_{CC} = V_{REF} = 5.12 V		±1	±2.5	LSB
T_{CONV}	Conversion time		61		62	tc(ϕ)
IV_{REF}	Reference input current	V_{REF} = 5.0 V	50	150	200	µA
I_{IA}	Analog port input current			0.5	5.0	µA
R_{LADDER}	Ladder resistor			35		kΩ

APPENDIX

3.1 Electrical characteristics

3.1.5 Timing requirements and switching characteristics

Table 3.1.7 Timing requirements

(V_{CC} = 4.0 to 5.5 V, V_{SS} = 0 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{W(\overline{RESET})}$	Reset input "L" pulse width	2.0			µs
$t_{C(XIN)}$	Main clock input cycle time (X_{IN} input)	238			ns
$t_{WH(XIN)}$	Main clock input "H" pulse width	60			ns
$t_{WL(XIN)}$	Main clock input "L" pulse width	60			ns
$t_{C(XCIN)}$	Sub-clock input cycle time (X_{CIN} input)	20			µs
$t_{WH(XCIN)}$	Sub-clock input "H" pulse width	5.0			µs
$t_{WL(XCIN)}$	Sub-clock input "L" pulse width	5.0			µs
$t_{C(CNTR)}$	$CNTR_0$, $CNTR_1$ input cycle time	4.0			µs
$t_{WH(CNTR)}$	$CNTR_0$, $CNTR_1$ input "H" pulse width	1.6			µs
$t_{WL(CNTR)}$	$CNTR_0$, $CNTR_1$ input "L" pulse width	1.6			µs
$t_{WH(INT)}$	INT_0 to INT_4 input "H" pulse width	80			ns
$t_{WL(INT)}$	INT_0 to INT_4 input "L" pulse width	80			ns
$t_{C(SCLK)}$	Serial I/O clock input cycle time	0.95			µs
$t_{WH(SCLK)}$	Serial I/O clock input "H" pulse width	400			ns
$t_{WL(SCLK)}$	Serial I/O clock input "L" pulse width	400			ns
$t_{su(SCLK–SIN)}$	Serial I/O input set up time	200			ns
$t_{h(SCLK–SIN)}$	Serial I/O input hold time	200			ns

Table 3.1.8 Switching characteristics

(V_{CC} = 4.0 to 5.5 V, V_{SS} = 0 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{WH(SCLK)}$	Serial I/O clock output "H" pulse width	C_L = 100 pF	$t_{C(SCLK)}$/2–160			ns
$t_{WL(SCLK)}$	Serial I/O clock output "L" pulse width	C_L = 100 pF	$t_{C(SCLK)}$/2–160			ns
$t_{d(SCLK–SOUT)}$	Serial I/O output delay time				0.2 tc	ns
$t_{v(SCLK–SOUT)}$	Serial I/O output valid time		0			ns
$t_{r(SCLK)}$	Serial I/O clock output rising time	C_L = 100 pF			40	ns
$t_{f(SCLK)}$	Serial I/O clock output falling time	C_L = 100 pF			40	ns
$t_{r(Pch–strg)}$	P-channel high-breakdown voltage output rising time **(Note 1)**	C_L = 100 pF V_{EE} = V_{CC}–43 V		55		ns
$t_{r(Pch–weak)}$	P-channel high-breakdown voltage output rising time **(Note 2)**	C_L = 100 pF V_{EE} = V_{CC}–43 V		1.8		µs

Notes 1: When bit 7 of the FLDC mode register (address $0EF4_{16}$) is at "0".
2: When bit 7 of the FLDC mode register (address $0EF4_{16}$) is at "1".

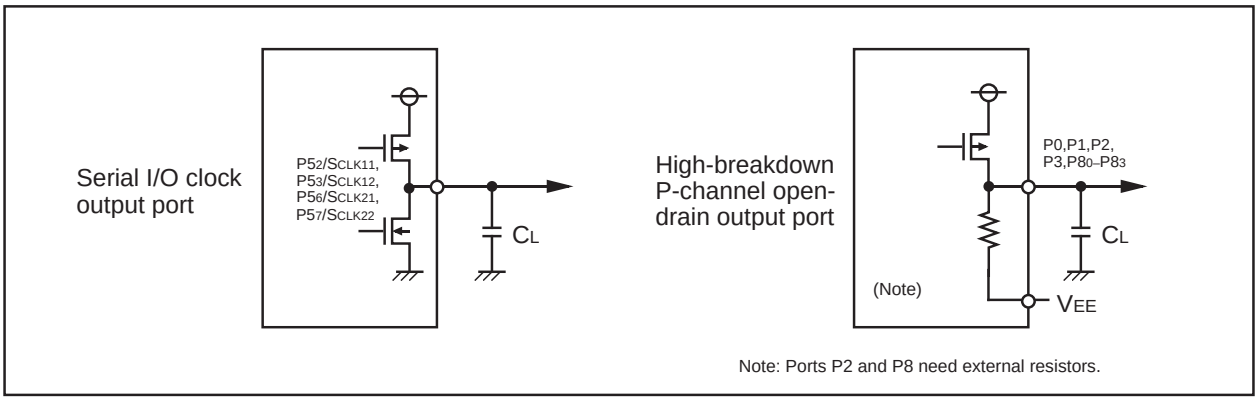


Fig. 3.1.1 Circuit for measuring output switching characteristics

Timing Diagram

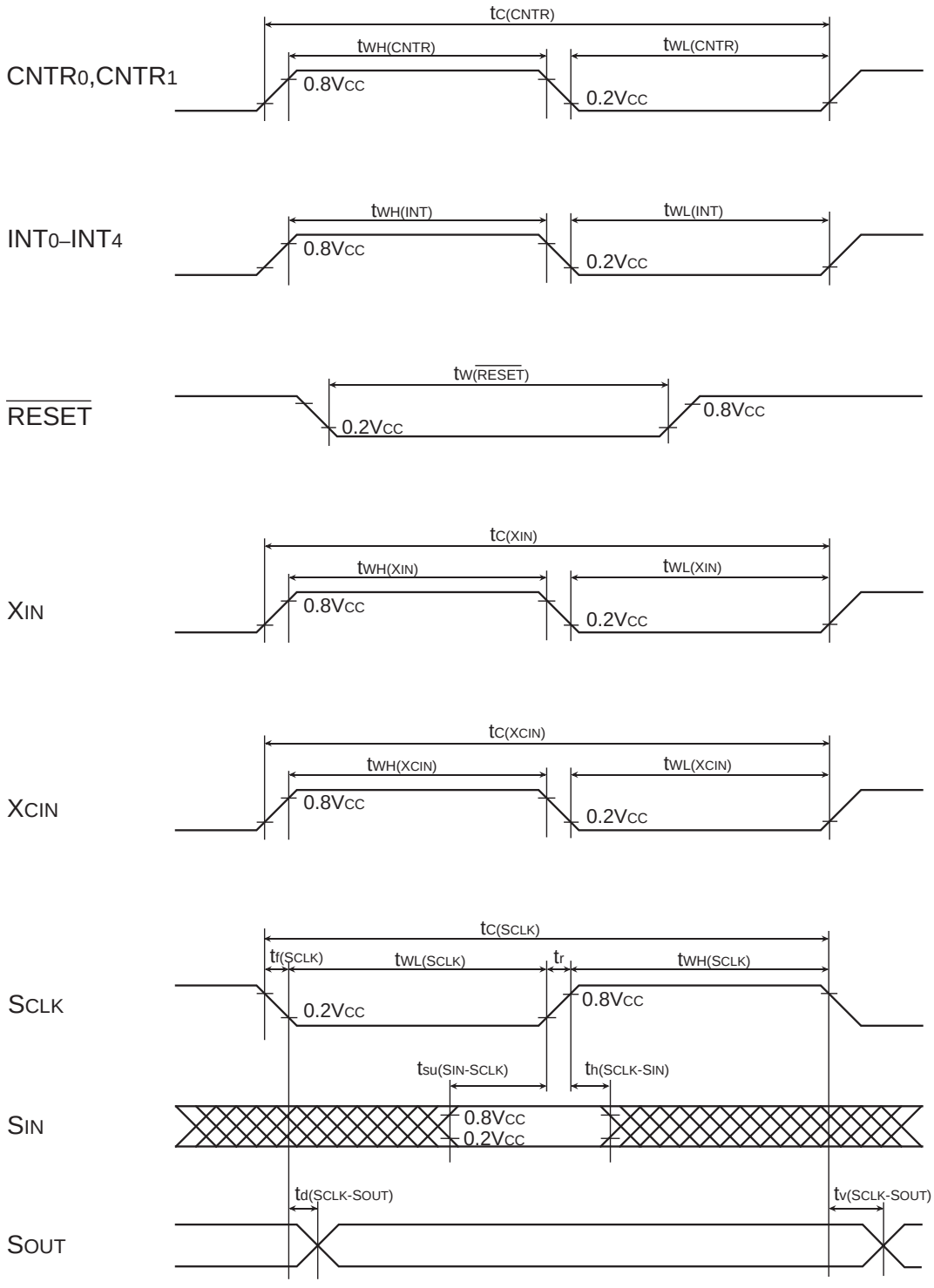


Fig. 3.1.2 Timing diagram

3.2 Standard characteristics

3.2.1 Power source current standard characteristics

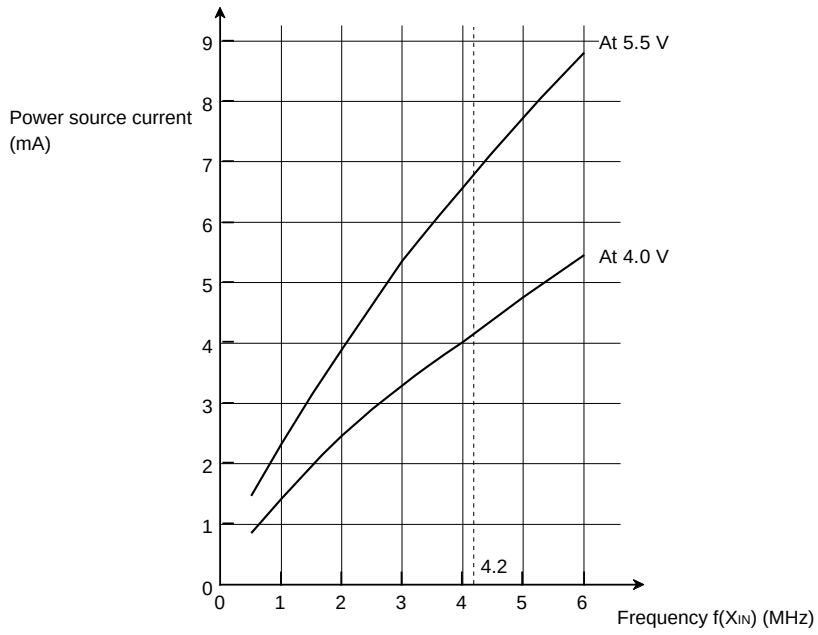


Fig. 3.2.1 Power source current standard characteristics

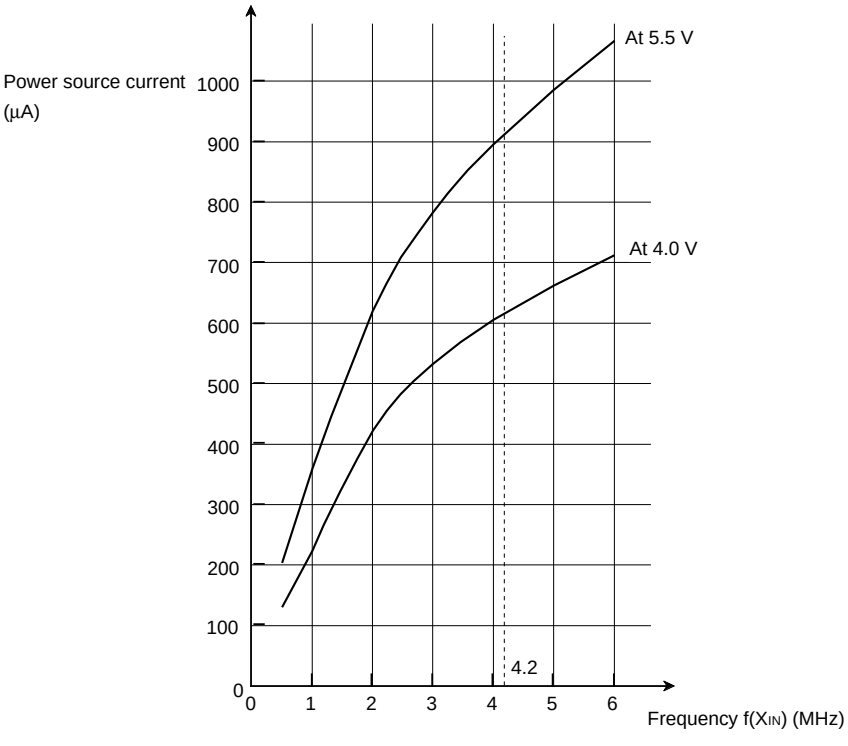


Fig. 3.2.2 Power source current standard characteristics (in wait mode)

3.2.2 Port standard characteristics

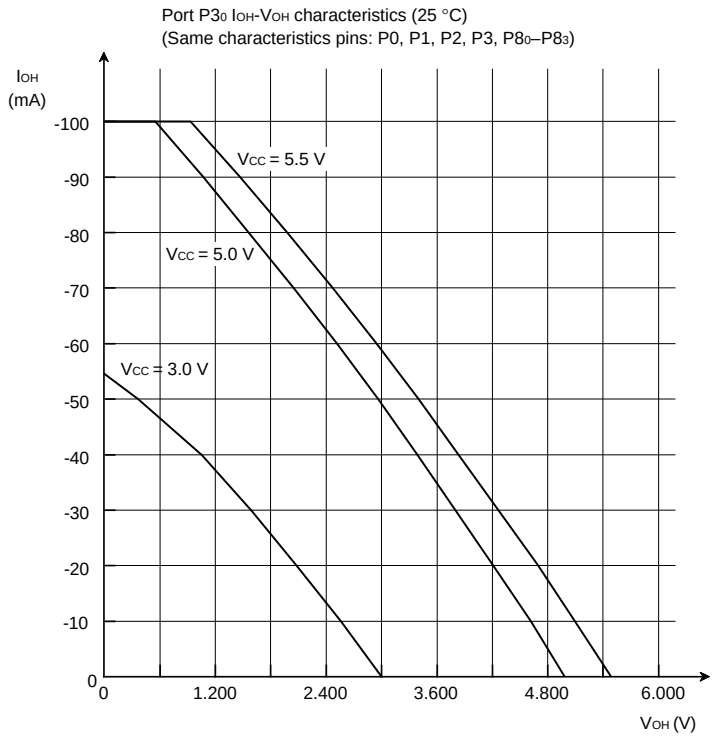


Fig. 3.2.3 High-breakdown P-channel open-drain output port characteristics (25 °C)

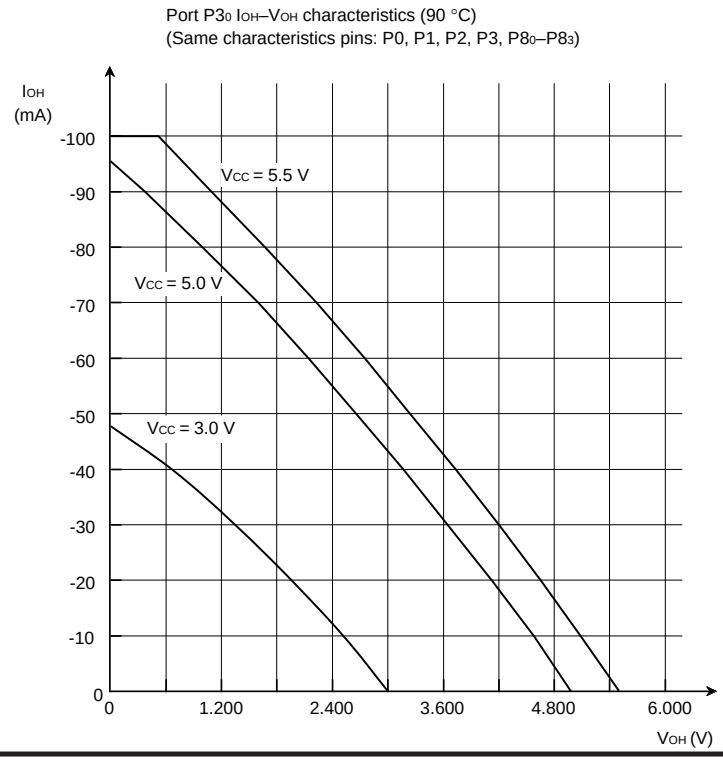


Fig. 3.2.4 High-breakdown P-channel open-drain output port characteristics (90 °C)

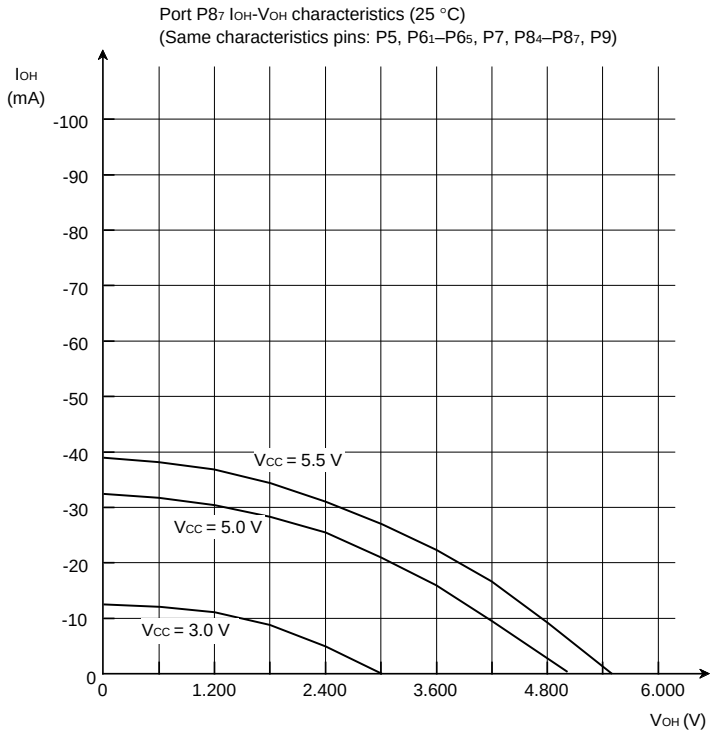


Fig. 3.2.5 CMOS output port P-channel side characteristics (25 °C)

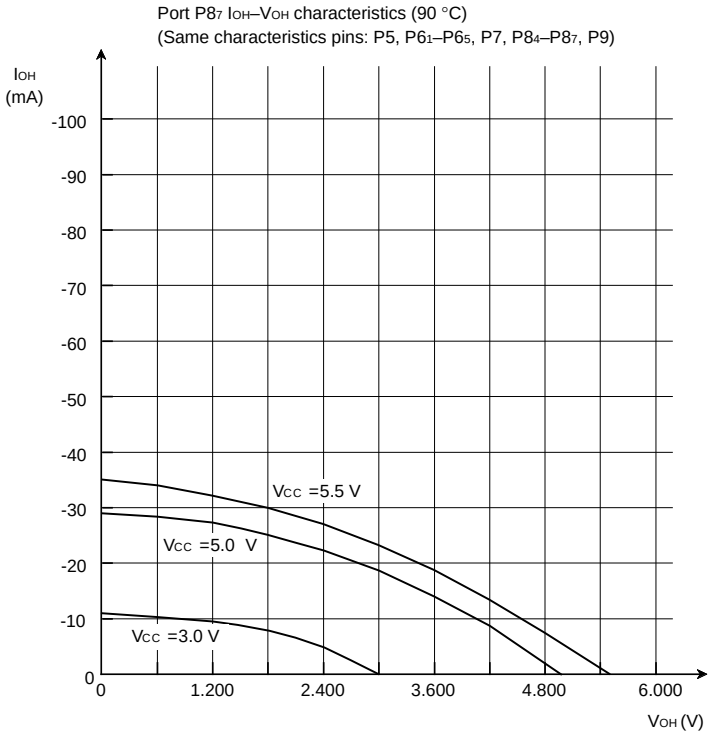


Fig. 3.2.6 CMOS output port P-channel side characteristics (90 °C)

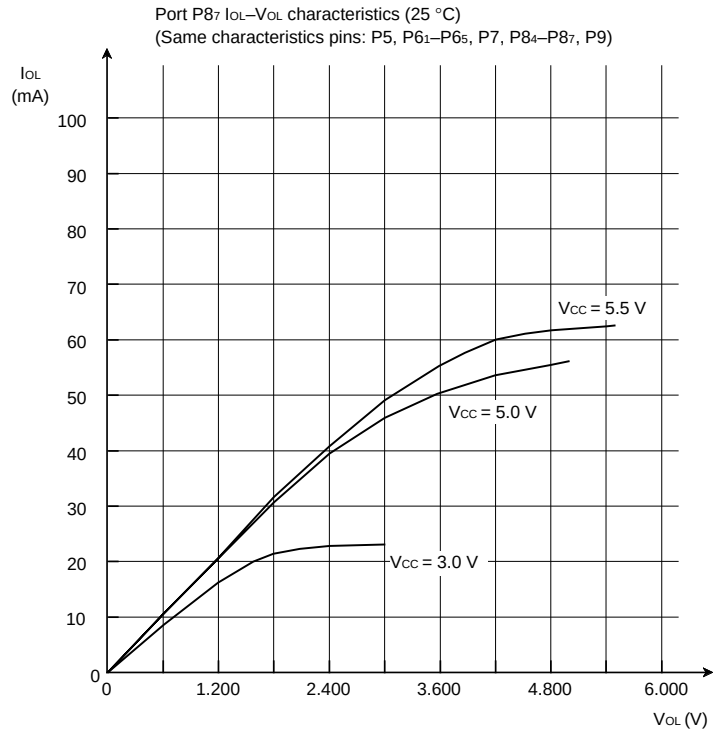


Fig. 3.2.7 CMOS output port N-channel side characteristics (25 °C)

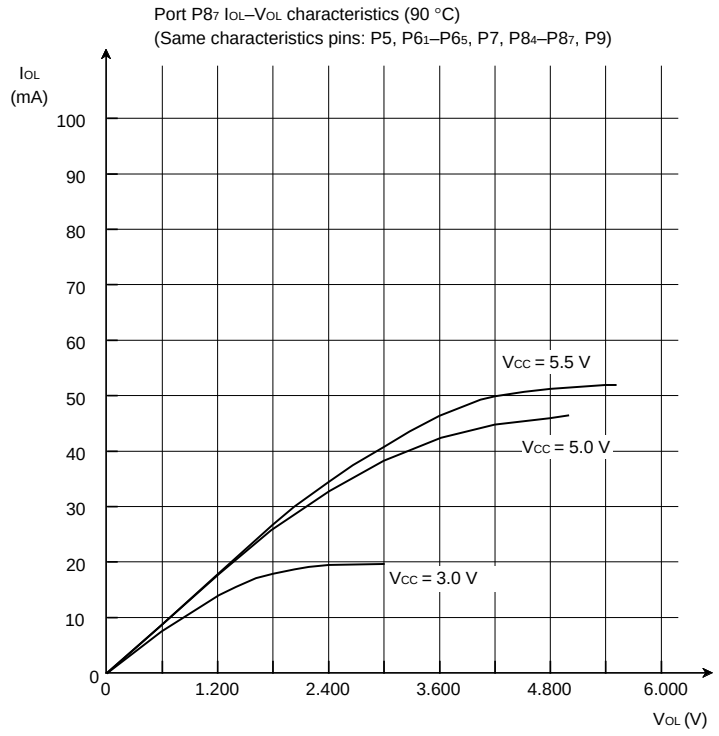


Fig. 3.2.8 CMOS output port N-channel side characteristics (90 °C)

3.2 Standard characteristics

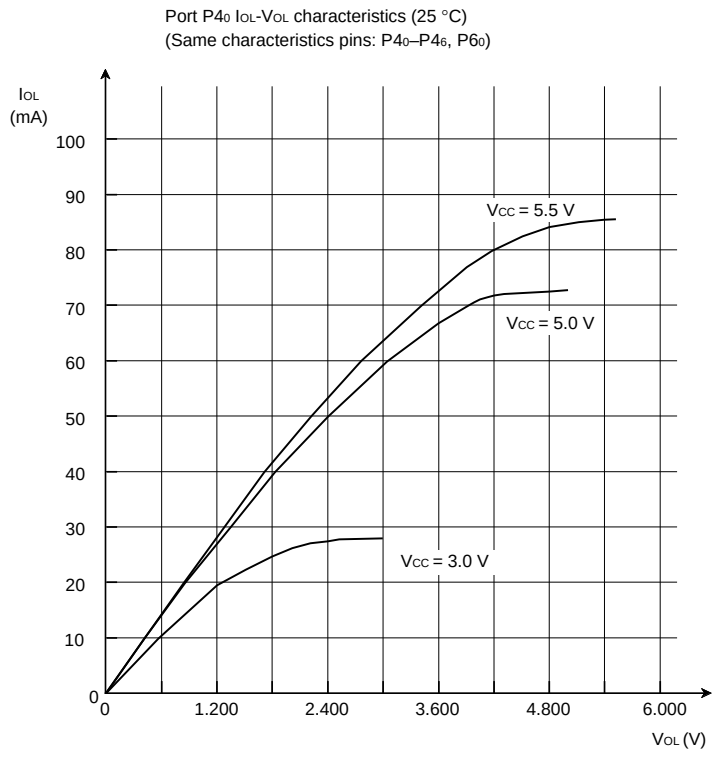


Fig. 3.2.9 N-channel open-drain output port characteristics (25 °C)

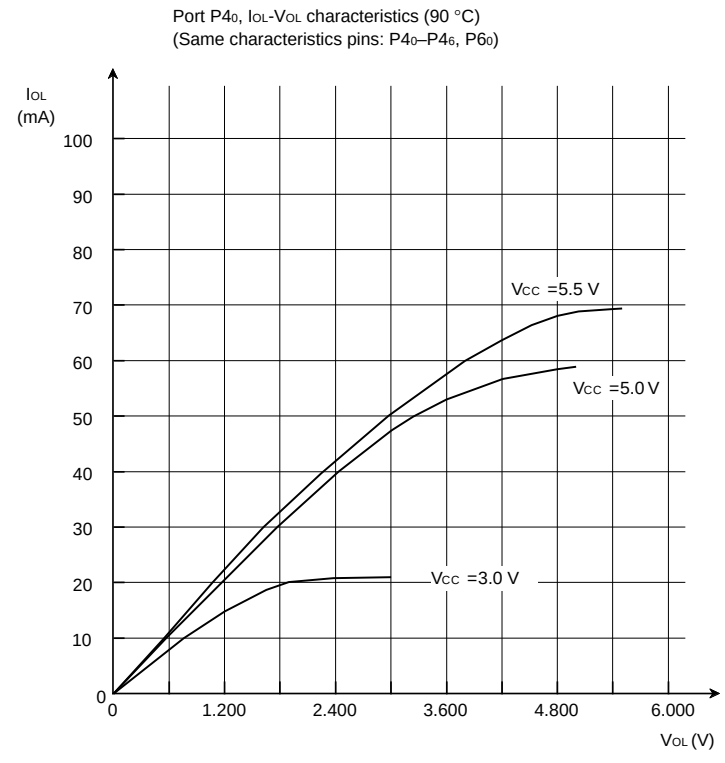


Fig. 3.2.10 N-channel open-drain output port characteristics (90 °C)

3.2.3 A-D conversion standard characteristics

Figure 3.2.11 shows the A-D conversion standard characteristics.

The lower line on the graph indicates the absolute precision error. It expresses the deviation from the ideal value. For example, the conversion of output code from 00_{16} to 01_{16} occurs ideally at the point of AN_0 = 2.5 mV, but the measured value is –2 mV. Accordingly, the measured point of conversion is defined as "2.5 – 2 = 0.5 mV".

The upper line on the graph indicates the width of input voltages equivalent to output codes. For example, the measured width of the input voltage for output code 60_{16} is 6 mV, so that the differential nonlinear error is defined as "6 – 5 = 1 mV (0.2 LSB)".

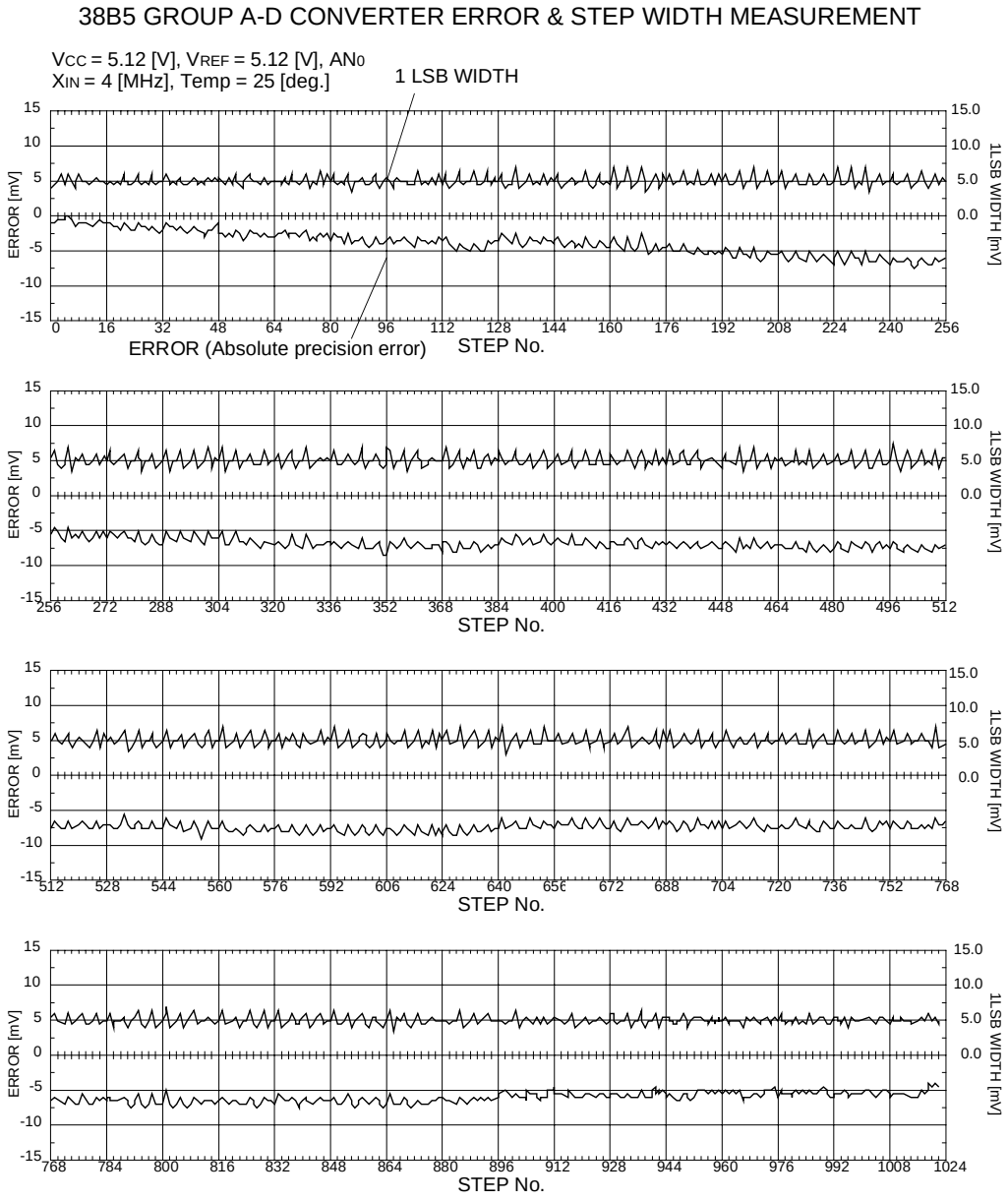


Fig. 3.2.11 A-D conversion standard characteristics

3.3 Notes on use

3.3.1 Notes on interrupts

(1) Switching external interrupt detection edge

For the products able to switch the external interrupt detection edge, switch it as the following sequence.

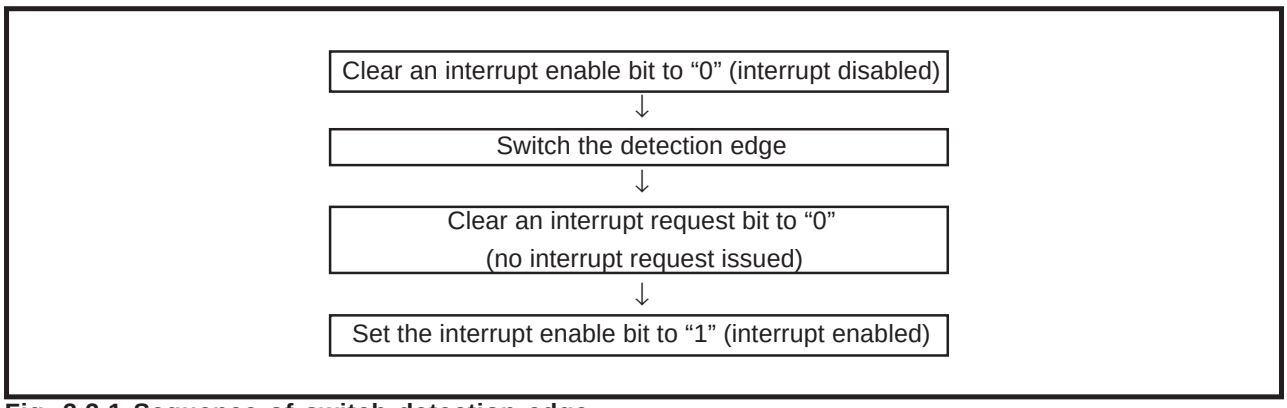


Fig. 3.3.1 Sequence of switch detection edge

■ **Reason**

The interrupt circuit recognizes the switching of the detection edge as the change of external input signals. This may cause an unnecessary interrupt.

(2) Check of interrupt request bit

● When executing the **BBC** or **BBS** instruction to an interrupt request bit of an interrupt request register immediately after this bit is set to “0” by using a data transfer instruction, execute one or more instructions before executing the **BBC** or **BBS** instruction.

■ **Reason**

If the **BBC** or **BBS** instruction is executed immediately after an interrupt request bit of an interrupt request register is cleared to “0”, the value of the interrupt request bit before being cleared to “0” is read.

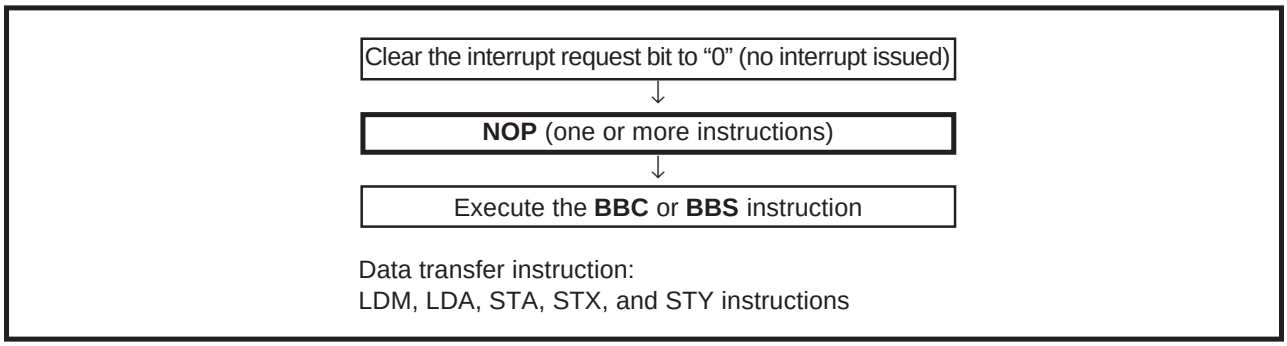


Fig. 3.3.2 Sequence of check of interrupt request bit

(3) Structure of interrupt control register 2

Fix the bit 7 of the interrupt control register 2 to "0". Figure 3.3.3 shows the structure of the interrupt control register 2.

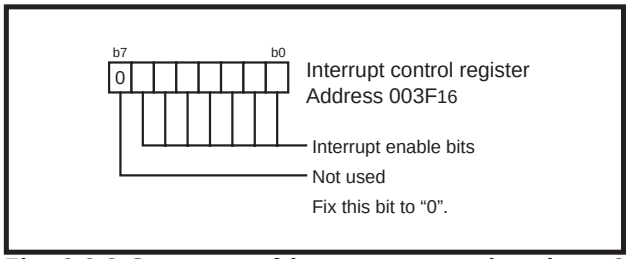


Fig. 3.3.3 Structure of interrupt control register 2

3.3.2 Notes on serial I/O1

(1) Clock

■ **Using internal clock**

After setting the synchronous clock to an internal clock, clear the serial I/O interrupt request bit before perform the normal serial I/O transfer or the serial I/O automatic transfer.

■ **Using external clock**

After inputting "H" level to the external clock input pin, clear the serial I/O interrupt request bit before performing the normal serial I/O transfer or the serial I/O automatic transfer.

(2) Using serial I/O1 interrupt

Clear bit 3 of the interrupt request register 1 to "0" by software.

(3) State of S_{OUT1} pin

The S_{OUT1} pin control bit of the serial I/O1 control register 2 can be used to select the state of the S_{OUT1} pin when serial data is not transferred; either output active or high-impedance. However, when selecting an external synchronous clock; the S_{OUT1} pin can become the high-impedance state by setting the S_{OUT1} pin control bit to "1" when the serial I/O1 clock input is at "H" after transfer completion.

(4) Serial I/O initialization bit

- Set "0" to the serial I/O initialization bit of the serial I/O1 control register 1 when terminating a serial transfer during transferring.
- When writing "1" to the serial I/O initialization bit, the serial I/O1 is enabled, but each register is not initialized. Set the value of each register by program.

(5) Handshake signal

■ **S_{BUSY1} input signal**

Input an "H" level to the S_{BUSY1} input and an "L" level signal to the $\overline{S_{BUSY1}}$ input in the initial state. When the external synchronous clock is selected, switch the input level to the S_{BUSY1} input and the $\overline{S_{BUSY1}}$ input while the serial I/O1 clock input is in "H" state.

■ **S_{RDY1} input•output signal**

When selecting the internal synchronous clock, input an "L" level to the S_{RDY1} input and an "H" level signal to the $\overline{S_{RDY1}}$ input in the initial state.

(6) 8-bit serial I/O mode

■ **When selecting external synchronous clock**

When an external synchronous clock is selected, the contents of the serial I/O1 register are being shifted continually while the transfer clock is input to the serial I/O1 clock pin. In this case, control the clock externally.

(7) In automatic transfer serial I/O mode

■ **Set of automatic transfer interval**

- When the S_{BUSY1} output is used, and the S_{BUSY1} output and the S_{STB1} output function as signals for each transfer data set by the S_{BUSY1} output•S_{STB1} output function selection bit of serial I/O1 control register 2; the transfer interval is inserted before the first data is transmitted/received, and after the last data is transmitted/received. Accordingly, regardless of the contents of the S_{BUSY1} output•S_{STB1} output function selection bit, this transfer interval for each 1-byte data becomes 2 cycles longer than the value set by the automatic transfer interval set bits of serial I/O1 control register 3.
- When using the S_{STB1} output, regardless of the contents of the S_{BUSY1} output•S_{STB1} output function selection bit, this transfer interval for each 1-byte data becomes 2 cycles longer than the value set by the automatic transfer interval set bits of serial I/O1 control register 3.
- When using the combined output of S_{BUSY1} and S_{STB1} as the signal for each of all transfer data set, the transfer interval after completion of transmission/reception of the last data becomes 2 cycles longer than the value set by the automatic transfer interval set bits.
- Set the transfer interval of each 1-byte data transfer to 5 or more cycles of the internal clock ϕ after the rising edge of the last bit of a 1-byte data.
- When selecting an external clock, the set of automatic transfer interval becomes invalid.

■ **Set of serial I/O1 transfer counter**

- Write the value decreased by 1 from the number of transfer data bytes to the serial I/O1 transfer counter.
- When selecting an external clock, after writing a value to the serial I/O1 register/transfer counter, wait for 5 or more cycles of internal clock ϕ before inputting the transfer clock to the serial I/O1 clock pin.

■ **Serial I/O initialization bit**

A serial I/O1 automatic transfer interrupt request occurs when “0” is written to the serial I/O initialization bit during an operation. Disable it with the interrupt enable bit as necessary by program.

3.3.3 Notes on serial I/O2

(1) Notes when selecting clock synchronous serial I/O

① **Stop of transmission operation**

As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the transmit enable bit to “0” (transmit disabled).

● **Reason**

Since transmission is not stopped and the transmission circuit is not initialized even if only the serial I/O2 enable bit is cleared to “0” (serial I/O2 disabled), the internal transmission is running (in this case, since pins TxD, RxD, S_{CLK21}, S_{CLK22} and $\overline{S_{RDY2}}$ function as I/O ports, the transmission data is not output). When data is written to the transmit buffer register in this state, data starts to be shifted to the transmit shift register. When the serial I/O2 enable bit is set to “1” at this time, the data during internally shifting is output to the TxD pin and an operation failure occurs.

② **Stop of receive operation**

As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the receive enable bit to “0” (receive disabled), or clear the serial I/O2 enable bit to “0” (serial I/O2 disabled).

③ **Stop of transmit/receive operation**

As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, simultaneously clear both the transmit enable bit and receive enable bit to "0" (transmit and receive disabled).

(when data is transmitted and received in the clock synchronous serial I/O mode, any one of data transmission and reception cannot be stopped.)

● **Reason**

In the clock synchronous serial I/O mode, the same clock is used for transmission and reception. If any one of transmission and reception is disabled, a bit error occurs because transmission and reception cannot be synchronized.

In this mode, the clock circuit of the transmission circuit also operates for data reception. Accordingly, the transmission circuit does not stop by clearing only the transmit enable bit to "0" (transmit disabled). Also, the transmission circuit is not initialized by clearing the serial I/O2 enable bit to "0" (serial I/O2 disabled) (refer to (1), ①).

(2) Notes when selecting clock asynchronous serial I/O

① **Stop of transmission operation**

As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the transmit enable bit to "0" (transmit disabled).

● **Reason**

Since transmission is not stopped and the transmission circuit is not initialized even if only the serial I/O2 enable bit is cleared to "0" (serial I/O2 disabled), the internal transmission is running (in this case, since pins TxD, RxD, S_{CLK21}, S_{CLK22} and $\overline{S_{RDY2}}$ function as I/O ports, the transmission data is not output). When data is written to the transmit buffer register in this state, data starts to be shifted to the transmit shift register. When the serial I/O2 enable bit is set to "1" at this time, the data during internally shifting is output to the TxD pin and an operation failure occurs.

② **Stop of receive operation**

As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the receive enable bit to "0" (receive disabled).

③ **Stop of transmit/receive operation**

Only transmission operation is stopped.

As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the transmit enable bit to "0" (transmit disabled).

● **Reason**

Since transmission is not stopped and the transmission circuit is not initialized even if only the serial I/O2 enable bit is cleared to "0" (serial I/O2 disabled), the internal transmission is running (in this case, since pins TxD, RxD, S_{CLK21}, S_{CLK22} and $\overline{S_{RDY2}}$ function as I/O ports, the transmission data is not output). When data is written to the transmit buffer register in this state, data starts to be shifted to the transmit shift register. When the serial I/O2 enable bit is set to "1" at this time, the data during internally shifting is output to the TxD pin and an operation failure occurs.

Only receive operation is stopped.

As for the serial I/O2 that can be used as either a clock synchronous or an asynchronous (UART) serial I/O, clear the receive enable bit to "0" (receive disabled).

(3) $\overline{S_{RDY2}}$ output of reception side

When signals are output from the $\overline{S_{RDY2}}$ pin on the reception side by using an external clock in the clock synchronous serial I/O mode, set all of the receive enable bit, the $\overline{S_{RDY2}}$ output enable bit, and the transmit enable bit to "1" (transmit enabled).

(4) Setting serial I/O2 control register again

Set the serial I/O2 control register again after the transmission and the reception circuits are reset by clearing both the transmit enable bit and the receive enable bit to "0."

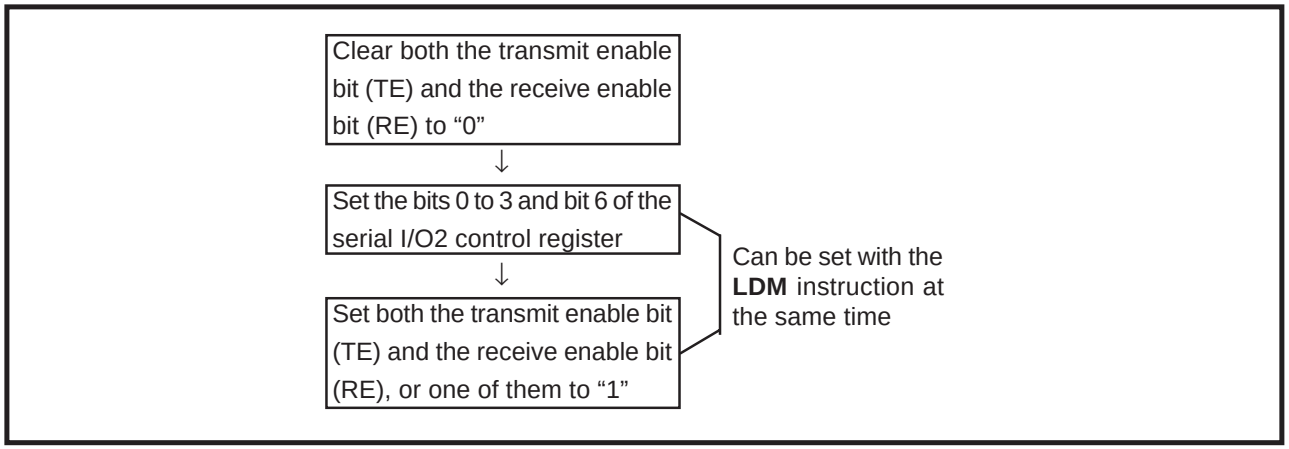


Fig. 3.3.4 Sequence of setting serial I/O2 control register again

(5) Data transmission control with referring to transmit shift register completion flag

The transmit shift register completion flag changes from "1" to "0" with a delay of 0.5 to 1.5 shift clocks. When data transmission is controlled with referring to the flag after writing the data to the transmit buffer register, note the delay.

(6) Transmission control when external clock is selected

When an external clock is used as the synchronous clock for data transmission, set the transmit enable bit to "1" at "H" of the serial I/O2 clock input level. Also, write the transmit data to the transmit buffer register (serial I/O shift register) at "H" of the serial I/O2 clock input level.

(7) Transmit interrupt request when transmit enable bit is set

The transmission interrupt request bit is set and the interruption request is generated even when selecting timing that either of the following flags is set to "1" as timing where the transmission interruption is generated.

- Transmit buffer empty flag is set to "1"
- Transmit shift register completion flag is set to "1"

Therefore, when the transmit interrupt is used, set the transmit interrupt enable bit to transmit enabled as the following sequence.

① Transmit enable bit is set to "1"

② Transmit interrupt request bit is set to "0"

● **Reason**

When the transmission enable bit is set to "1", the transmit buffer empty flag and transmit shift register completion flag are set to "1".

(8) Using TxD pin

The $P5_5$/TxD P-channel output disable bit of UART control register is valid in both cases: using as a normal I/O port and as the TxD pin. Do not supply Vcc + 0.3 V or more even when using the $P5_5$/TxD pin as an N-channel open-drain output.

Additionally, in the serial I/O2, the TxD pin latches the last bit and continues to output it after completing transmission.

3.3.4 Notes on FLD controller

- Set a value of 03_{16} or more to the Toff1 time set register.
- When displaying in the gradation display mode, select the 16 timing mode by the timing number control bit (bit 4 of FLDC mode register (address $0EF4_{16}$) = "0").

3.3.5 Notes on A-D converter

(1) Analog input pin

■ Make the signal source impedance for analog input low, or equip an analog input pin with an external capacitor of 0.01 μF to 1 μF. Further, be sure to verify the operation of application products on the user side.

● **Reason**

An analog input pin includes the capacitor for analog voltage comparison. Accordingly, when signals from signal source with high impedance are input to an analog input pin, charge and discharge noise generates. This may cause the A-D conversion precision to be worse.

■ When the $P6_4/INT_4/S_{BUSY1}/AN_{10}$ pin is selected as analog input pin, external interrupt function (INT_4) becomes invalid.

(2) A-D converter power source pin

The AV_{SS} pin is A-D converter power source pin. Regardless of using the A-D conversion function or not, connect it as following :

• AV_{SS} : Connect to the V_{SS} line

● **Reason**

If the AV_{SS} pin is opened, the microcomputer may have a failure because of noise or others.

(3) Clock frequency during A-D conversion

The comparator consists of a capacity coupling, and a charge of the capacity will be lost if the clock frequency is too low. Thus, make sure the following during an A-D conversion.

• $f(X_{IN})$ is 250 kHz or more
• Use clock divided by main clock ($f(X_{IN})$) as internal system clock.
• Do not execute the **STP** instruction and **WIT** instruction

3.3.6 Notes on PWM

- For PWM_0 output, "L" level is output first.
- After data is set to the PWM register (low-order) and the PWM register (high-order), PWM waveform corresponding to new data is output from next repetitive cycle.

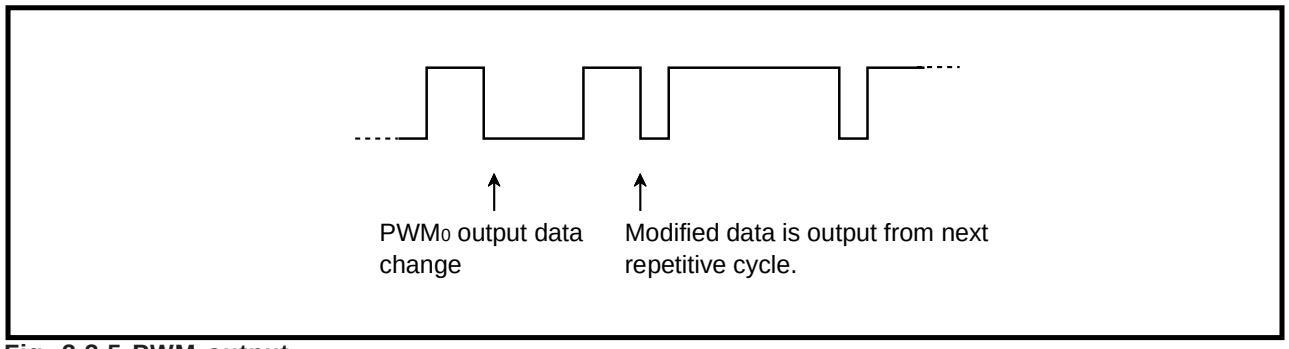


Fig. 3.3.5 PWM output

3.3 Notes on use

3.3.7 Notes on watchdog timer

- The watchdog timer continues to count even while waiting for stop release. Accordingly, make sure that watchdog timer does not underflow during this term by writing to the watchdog timer control register (address $002B_{16}$) once before executing the STP instruction, etc.
- Once a "1" is written to the STP instruction disable bit (bit 6) of the watchdog timer control register (address $002B_{16}$), it cannot be programmed to "0" again. This bit becomes "0" after reset.

3.3.8 Notes on reset circuit

(1) Reset input voltage control

Make sure that the reset input voltage is 0.5 V or less for Vcc of 2.7 V.
Perform switch to the high-speed mode when power source voltage is within 4.0 to 5.5 V.

(2) Countermeasure when $\overline{\text{RESET}}$ signal rise time is long

In case where the $\overline{\text{RESET}}$ signal rise time is long, connect a ceramic capacitor or others across the $\overline{\text{RESET}}$ pin and the Vss pin. And use a 1000 pF or more capacitor for high frequency use. When connecting the capacitor, note the following :

- Make the length of the wiring which is connected to a capacitor as short as possible.
- Be sure to verify the operation of application products on the user side.

● **Reason**

If the several nanosecond or several ten nanosecond impulse noise enters the $\overline{\text{RESET}}$ pin, it may cause a microcomputer failure.

3.3.9 Notes on input and output pins

(1) Notes in stand-by state

In stand-by state*1 for low-power dissipation, do not make input levels of an input port and an I/O port "undefined", especially for I/O ports of the N-channel open-drain.
Pull-up (connect the port to Vcc) or pull-down (connect the port to Vss) these ports through a resistor.
When determining a resistance value, note the following points:

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

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

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

● **Reason**

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

- P-channel.....when the content of the port latch is "0"
- 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. Note that the level becomes "undefined" depending on external circuits.
Accordingly, the potential which is input to the input buffer in a microcomputer is unstable in the state that input levels of a input port and an I/O port are "undefined". This may cause power source current.

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

(2) N-channel open-drain port

$P4_0$–$P4_2$, $P4_5$, $P4_6$, $P6_0$ of N-channel open-drain output ports have built-in hysteresis circuit for input. In standby state for low-power dissipation, do not make these pins floating state.

● **Reason**

When power sources for pull-up of these pins are cut off in standby state, these ports become floating. Accordingly, a current may flow from Vcc to Vss through built-in hysteresis circuit.

(3) Modifying output data with bit managing instruction

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

● **Reason**

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

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

Note the following :

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

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

3.3 Notes on use

3.3.10 Notes on programming

(1) Processor status register

① **Initializing of processor status register**

Flags which affect program execution must be initialized after a reset.
In particular, it is essential to initialize the T and D flags because they have an important effect on calculations.

● **Reason**

After a reset, the contents of the processor status register (PS) are undefined except for the I flag which is "1".

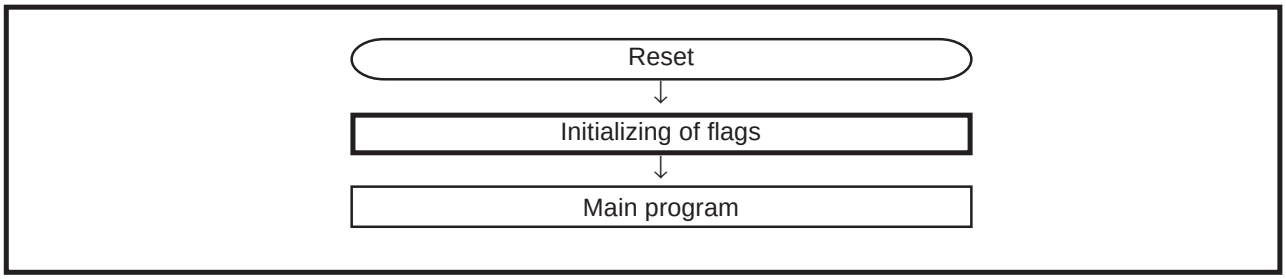


Fig. 3.3.6 Initialization of processor status register

② **How to reference the processor status register**

To reference the contents of the processor status register (PS), execute the **PHP** instruction once then read the contents of (S+1). If necessary, execute the **PLP** instruction to return the PS to its original status.
A **NOP** instruction should be executed after every **PLP** instruction.

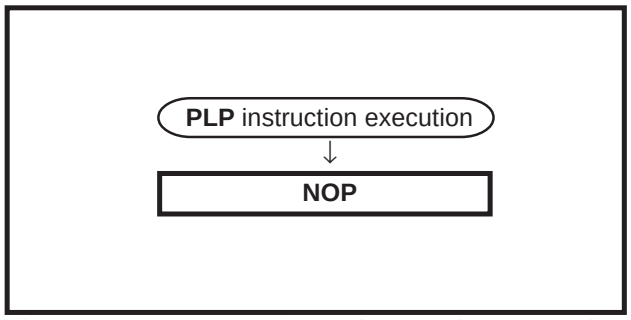


Fig. 3.3.7 Sequence of PLP instruction execution

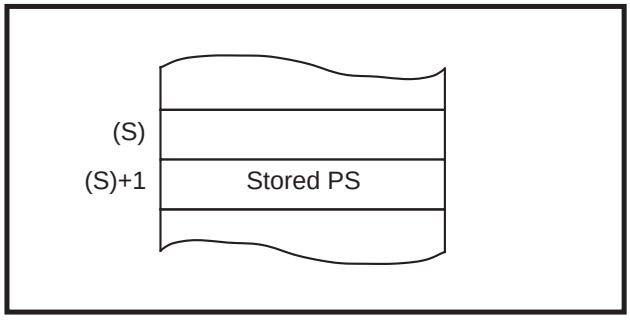


Fig. 3.3.8 Stack memory contents after PHP instruction execution

(2) Decimal calculations

① **Execution of decimal calculations**
The **ADC** and **SBC** are the only instructions which will yield proper decimal notation, set the decimal mode flag (D) to "1" with the **SED** instruction. After executing the **ADC** or **SBC** instruction, execute another instruction before executing the **SEC**, **CLC**, or **CLD** instruction.

② **Notes on status flag in decimal mode**
When decimal mode is selected, the values of three of the flags in the status register (the N, V, and Z flags) are invalid after a **ADC** or **SBC** instruction is executed.
The carry flag (C) is set to "1" if a carry is generated as a result of the calculation, or is cleared to "0" if a borrow is generated. To determine whether a calculation has generated a carry, the C flag must be initialized to "0" before each calculation. To check for a borrow, the C flag must be initialized to "1" before each calculation.

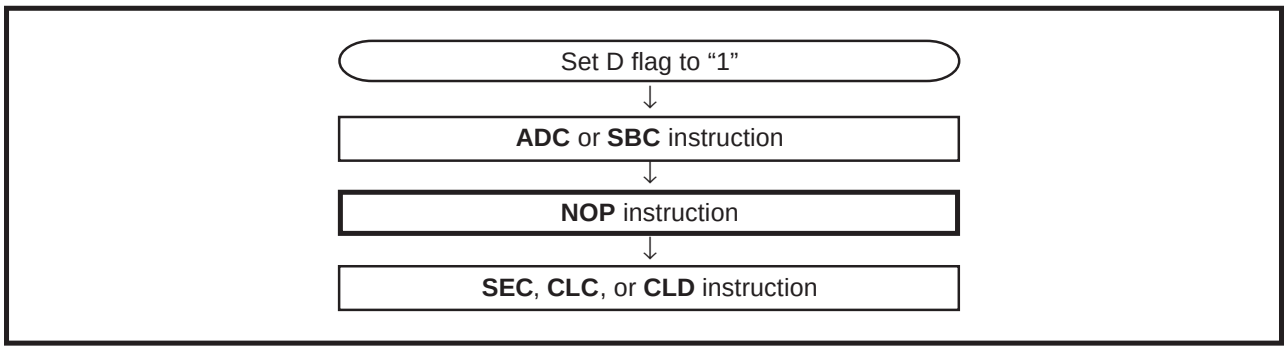


Fig. 3.3.9 Status flag at decimal calculations

(3) JMP instruction
When using the **JMP** instruction in indirect addressing mode, do not specify the last address on a page as an indirect address.

3.3.11 Programming and test of built-in PROM version

As for in the One Time PROM version (shipped in blank) and the built-in EPROM version, their built-in PROM can be read or programmed with a general-purpose PROM programmer using a special programming adapter.
The built-in EPROM version is available only for program development and on-chip program evaluation. The programming test and screening for PROM of the One Time PROM version (shipped in blank) are not performed in the assembly process and the following processes. To ensure reliability after programming, performing programming and test according to the Figure 3.3.10 before actual use are recommended.

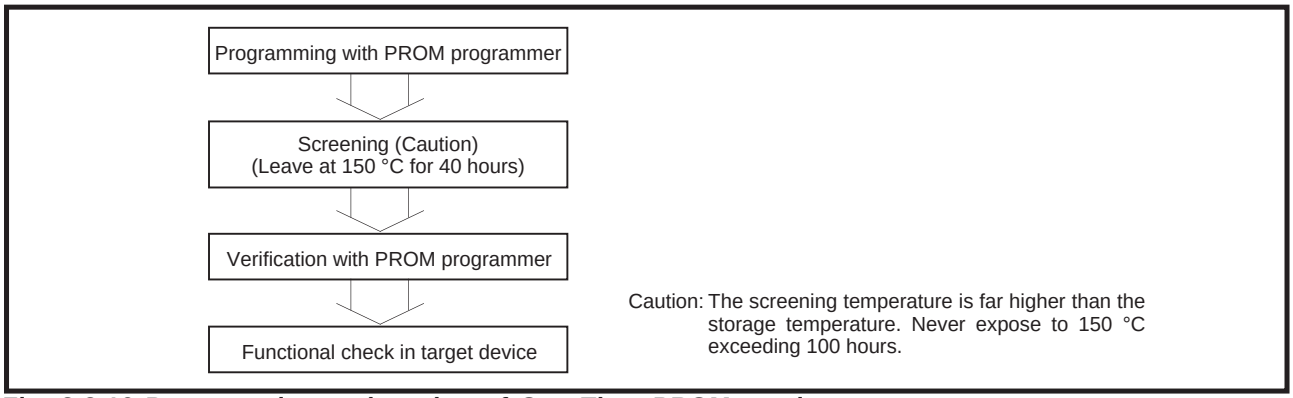


Fig. 3.3.10 Programming and testing of One Time PROM version

3.3 Notes on use

3.3.12 Notes on built-in PROM version

(1) Programming adapter

Use a special programming adapter shown in Table 3.3.1 and a general-purpose PROM programmer when reading from or programming to the built-in PROM in the built-in PROM version.

Table 3.3.1 Programming adapter

Microcomputer	Programming adapter
M38B59EFFP (One Time PROM version shipped in blank)	PCA4738F-80A
M38B59EFFS	PCA4738L-80A

(2) Programming/reading

In PROM mode, operation is the same as that of the M5M27C101K, but programming conditions of PROM programmer are not set automatically because there are no internal device ID codes.
Accurately set the following conditions for data programming/reading. Take care not to apply 21 V to the VPP pin (is also used as port $P4_7$), or the product may be permanently damaged.

① Programming voltage: 12.5 V

② Setting of PROM programmer address: Refer to **"Table 3.3.2"**

Table 3.3.2 PROM programmer address setting

Microcomputer	PROM programmer start address	PROM programmer end address
M38B59EFFP	Address 1080_{16}	Address $FFFD_{16}$
M38B59EFFS		

(3) Erasing

Contents of the windowed EPROM are erased through an ultraviolet light source with the wavelength 2537 Å ngstrom. At least 15 W•sec/cm^2 are required to erase EPROM contents.

3.3.13 Termination of unused pins

(1) Terminate unused pins

① Output ports : Open

② Input ports :

Connect each pin to Vss through each resistor of 1 kΩ to 10 kΩ.

As for pins whose potential affects to operation modes such as pins INT or others, select the Vcc pin or the Vss pin according to their operation mode.

③ I/O ports :

- Set the I/O ports for the input mode and connect them to Vss through each resistor of 1 kΩ to 10 kΩ.

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

(2) Termination remarks

① Input ports and I/O ports :

Do not open in the input mode.

● **Reason**

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

② I/O ports :

When setting for the input mode, do not connect to Vcc or Vss directly.

● **Reason**

If the direction register setup changes for the output mode because of a program runaway or noise, a short circuit may occur between a port and Vcc (or Vss).

③ I/O ports :

When setting for the input mode, do not connect multiple ports in a lump to Vcc or Vss through a resistor.

● **Reason**

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

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

3.4 Countermeasures against noise

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.

3.4.1 Shortest wiring length

The wiring on a printed circuit board can function 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) Package

Select the smallest possible package to make the total wiring length short.

● **Reason**

The wiring length depends on a microcomputer package. Use of a small package, for example QFP and not DIP, makes the total wiring length short to reduce influence of noise.

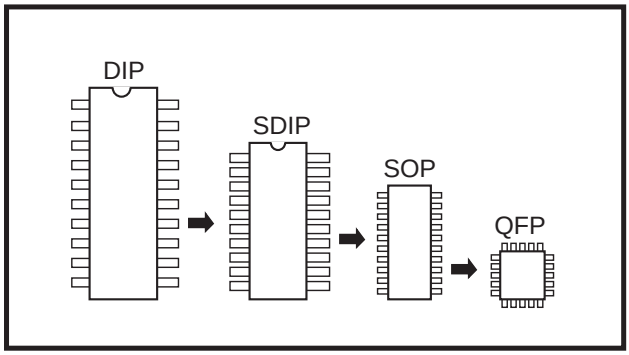


Fig. 3.4.1 Selection of packages

(2) Wiring for $\overline{\text{RESET}}$ pin

Make the length of wiring which is connected to the $\overline{\text{RESET}}$ pin as short as possible. Especially, connect a capacitor across the $\overline{\text{RESET}}$ pin and the V_{SS} pin with the shortest possible wiring (within 20mm).

● **Reason**

The width of a pulse input into the $\overline{\text{RESET}}$ pin is determined by the timing necessary conditions. If noise having a shorter pulse width than the standard is input to the $\overline{\text{RESET}}$ pin, the reset is released before the internal state of the microcomputer is completely initialized. This may cause a program runaway.

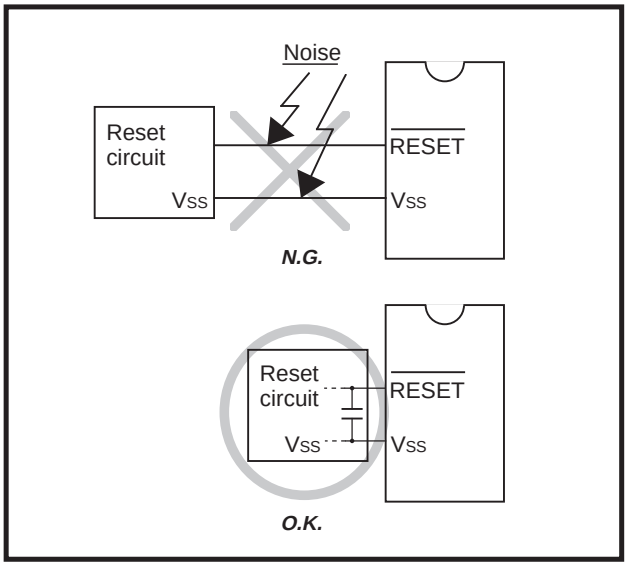


Fig. 3.4.2 Wiring for the $\overline{\text{RESET}}$ pin

(3) Wiring for clock input/output pins

- Make the length of wiring which is connected to clock I/O pins as short as possible.
- Make the length of wiring (within 20 mm) across the grounding lead of a capacitor which is connected to an oscillator and the Vss pin of a microcomputer as short as possible.
- Separate the Vss pattern only for oscillation from other Vss patterns.

● **Reason**

If noise enters clock I/O pins, clock waveforms may be deformed. This may cause a program failure or program runaway. Also, if a potential difference is caused by the noise between the Vss level of a microcomputer and the Vss level of an oscillator, the correct clock will not be input in the microcomputer.

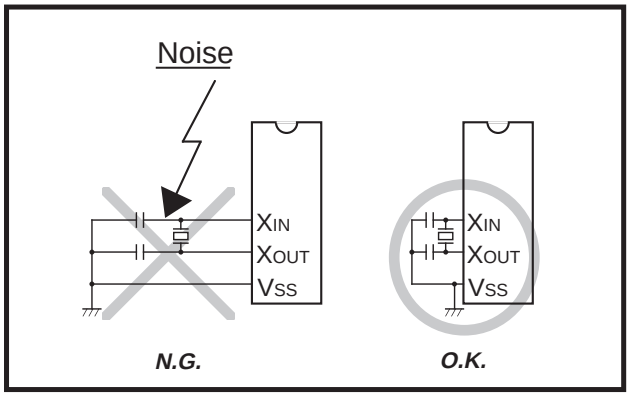


Fig. 3.4.3 Wiring for clock I/O pins

(4) Wiring to V_{PP} pin of One Time PROM version and EPROM version

Connect an approximately 5 kΩ resistor to the V_{PP} pin the shortest possible in series. When not connecting the resistor, make the length of wiring between the V_{PP} pin and the V_{SS} pin the shortest possible.

Note: Even when a circuit which included an approximately 5 kΩ resistor is used in the Mask ROM version, the microcomputer operates correctly.

● **Reason**

The V_{PP} pin of the One Time PROM 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 writing flow into the PROM. Because of this, noise can enter easily. If noise enters the V_{PP} pin, abnormal instruction codes or data are read from the built-in PROM, which may cause a program runaway.

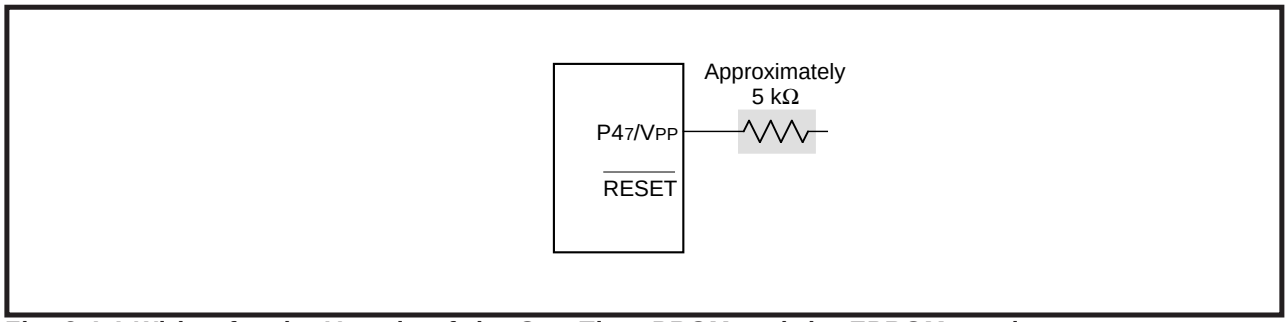


Fig. 3.4.4 Wiring for the V_{PP} pin of the One Time PROM and the EPROM version

3.4.2 Connection of bypass capacitor across Vss line and Vcc line

Connect an approximately 0.1 μF bypass capacitor across the V_{SS} line and the V_{CC} line as follows:

- Connect a bypass capacitor across the V_{SS} pin and the V_{CC} pin at equal length.
- Connect a bypass capacitor across the V_{SS} pin and the V_{CC} pin with the shortest possible wiring.
- Use lines with a larger diameter than other signal lines for V_{SS} line and V_{CC} line.
- Connect the power source wiring via a bypass capacitor to the V_{SS} pin and the V_{CC} pin.

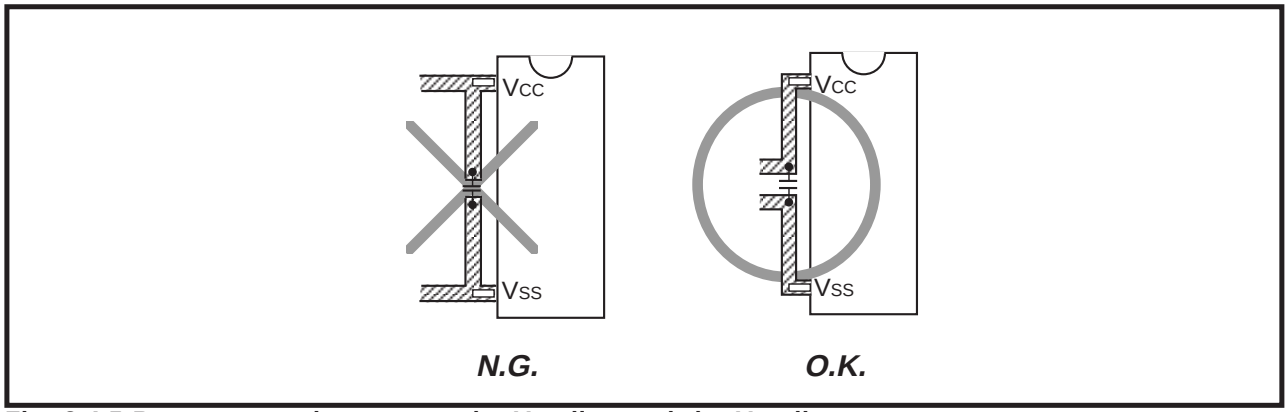


Fig. 3.4.5 Bypass capacitor across the V_{SS} line and the V_{CC} line

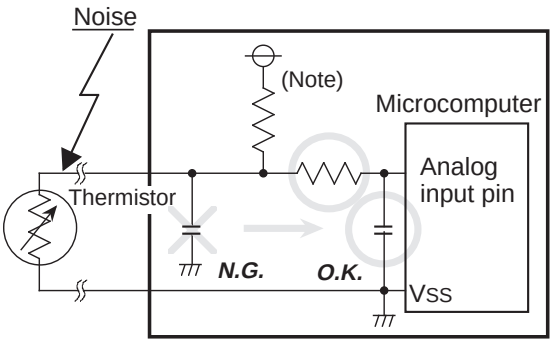
3.4.3 Wiring to analog input pins

- Connect an approximately 100 Ω to 1 kΩ resistor to an analog signal line which is connected to an analog input pin in series. Besides, connect the resistor to the microcomputer as close as possible.
- Connect an approximately 1000 pF capacitor across the VSS pin and the analog input pin. Besides, connect the capacitor to the VSS pin as close as possible. Also, connect the capacitor across the analog input pin and the VSS pin at equal length.

● **Reason**

Signals which is input in an analog input pin (such as an A-D converter/comparator input pin) are usually output signals from sensor. The sensor which detects a change of event is installed far from the printed circuit board with a microcomputer, the wiring to an analog input pin is longer necessarily. This long wiring functions as an antenna which feeds noise into the microcomputer, which causes noise to an analog input pin.

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.



Note : The resistor is used for dividing resistance with a thermistor.

Fig. 3.4.6 Analog signal line and a resistor and a capacitor

3.4.4 Oscillator concerns

Take care to prevent an oscillator that generates clocks for a microcomputer operation from being affected by other signals.

(1) Keeping oscillator away from large current signal lines

Install a microcomputer (and especially an oscillator) as far as possible from signal lines where a current larger than the tolerance of current value flows.

● **Reason**

In the system using a microcomputer, there are signal lines for controlling motors, LEDs, and thermal heads or others. When a large current flows through those signal lines, strong noise occurs because of mutual inductance.

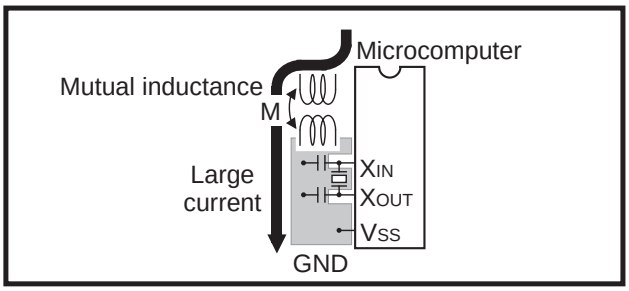


Fig. 3.4.7 Wiring for a large current signal line

3.4 Countermeasures against noise

(2) Installing oscillator away from signal lines where potential levels change frequently

Install an oscillator and a connecting pattern of an oscillator away from signal lines where potential levels change frequently. Also, do not cross such signal lines over the clock lines or the signal lines which are sensitive to noise.

● **Reason**

Signal lines where potential levels change frequently (such as the CNTR pin signal line) may affect other lines at signal rising edge or falling edge. If such lines cross over a clock line, clock waveforms may be deformed, which causes a microcomputer failure or a program runaway.

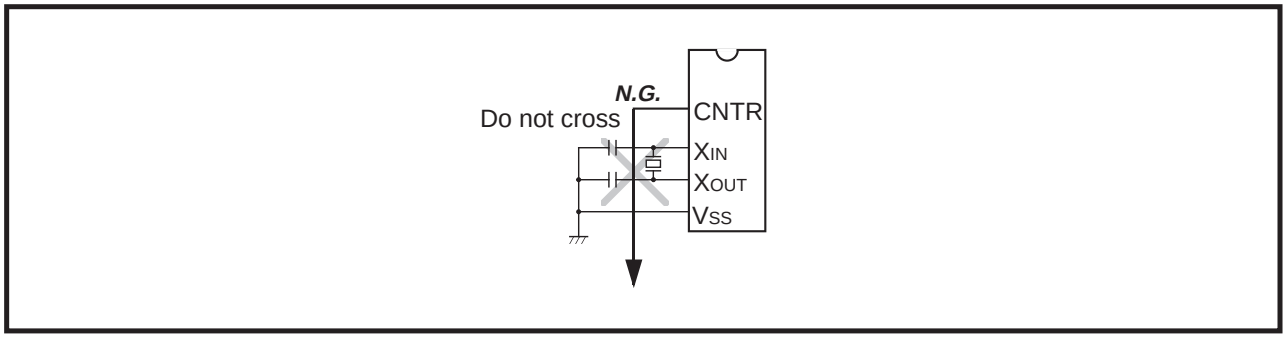


Fig. 3.4.8 Wiring of signal lines where potential levels change frequently

(3) Oscillator protection using Vss pattern

As for a two-sided printed circuit board, print a Vss pattern on the underside (soldering side) of the position (on the component side) where an oscillator is mounted.

Connect the Vss pattern to the microcomputer Vss pin with the shortest possible wiring. Besides, separate this Vss pattern from other Vss patterns.

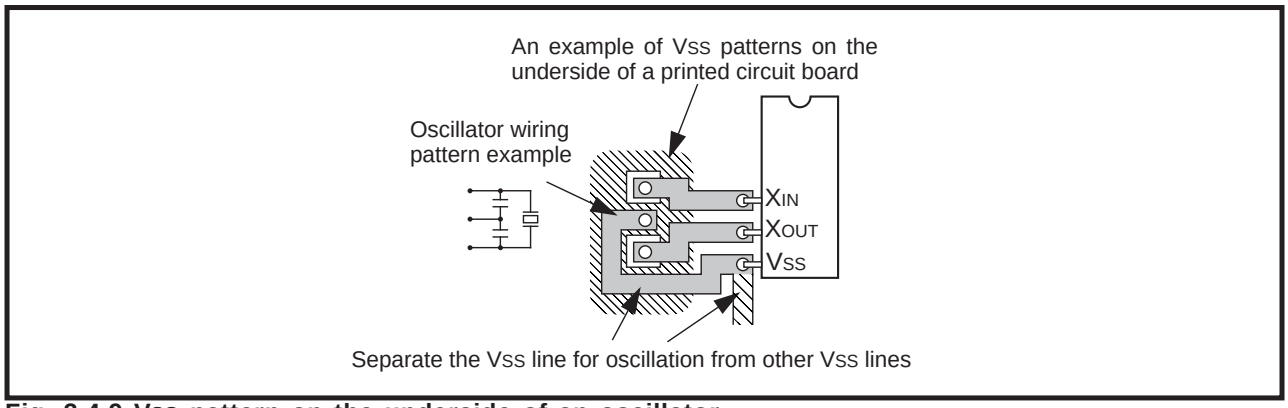


Fig. 3.4.9 Vss pattern on the underside of an oscillator

3.4.5 Setup for I/O ports

Setup I/O ports using hardware and software as follows:

<Hardware>

- Connect a resistor of 100 Ω or more to an I/O port in series.

<Software>

- As for an input port, read data several times by a program for checking whether input levels are equal or not.
- As for an output port, since the output data may reverse because of noise, rewrite data to its port latch at fixed periods.
- Rewrite data to direction registers and pull-up control registers at fixed periods.

Note: When a direction register is set for input port again at fixed periods, a several-nanosecond short pulse may be output from this port. If this is undesirable, connect a capacitor to this port to remove the noise pulse.

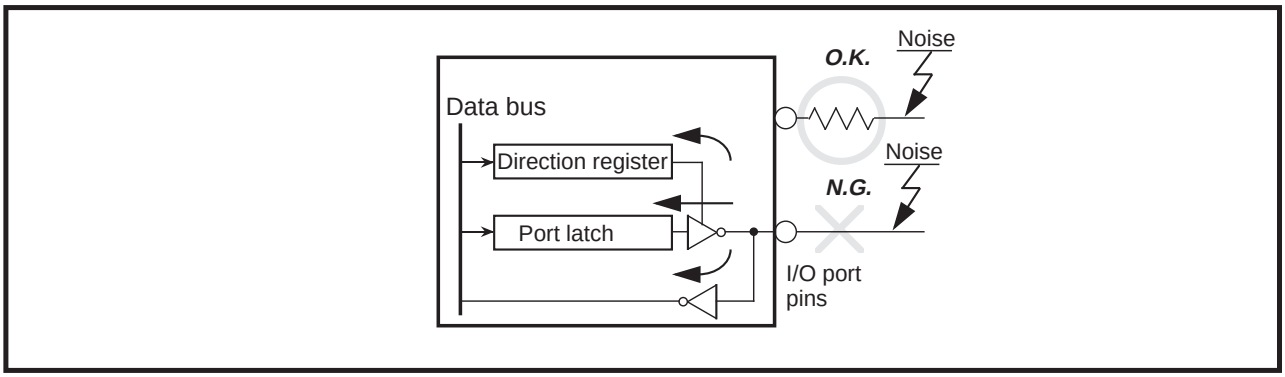


Fig. 3.4.10 Setup for I/O ports

3.4 Countermeasures against noise

3.4.6 Providing of watchdog timer function by software

If a microcomputer runs away because of noise or others, it can be detected by a software watchdog timer and the microcomputer can be reset to normal operation. This is equal to or more effective than program runaway detection by a hardware watchdog timer. The following shows an example of a watchdog timer provided by software.

In the following example, to reset a microcomputer to normal operation, the main routine detects errors of the interrupt processing routine and the interrupt processing routine detects errors of the main routine. This example assumes that interrupt processing is repeated multiple times in a single main routine processing.

<The main routine>

- Assigns a single byte of RAM to a software watchdog timer (SWDT) and writes the initial value N in the SWDT once at each execution of the main routine. The initial value N should satisfy the following condition:
  N+1 ≥ (Counts of interrupt processing executed in each main routine)
  As the main routine execution cycle may change because of an interrupt processing or others, the initial value N should have a margin.
- Watches the operation of the interrupt processing routine by comparing the SWDT contents with counts of interrupt processing after the initial value N has been set.
- Detects that the interrupt processing routine has failed and determines to branch to the program initialization routine for recovery processing in the following case:
  If the SWDT contents do not change after interrupt processing.

<The interrupt processing routine>

- Decrements the SWDT contents by 1 at each interrupt processing.
- Determines that the main routine operates normally when the SWDT contents are reset to the initial value N at almost fixed cycles (at the fixed interrupt processing count).
- Detects that the main routine has failed and determines to branch to the program initialization routine for recovery processing in the following case:
  If the SWDT contents are not initialized to the initial value N but continued to decrement and if they reach 0 or less.

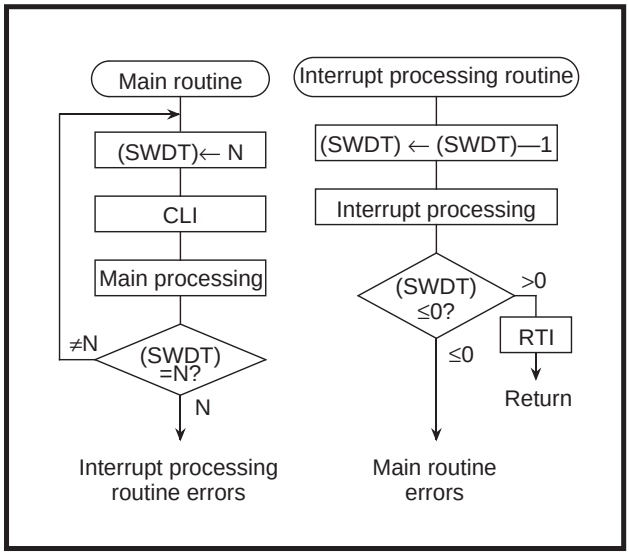


Fig. 3.4.11 Watchdog timer by software

3.5 Control registers

Port Pi

b7 b6 b5 b4 b3 b2 b1 b0

Port Pi (i = 0, 1, 2, 3, 4, 5, 7, 8)
(Pi: addresses 00_{16}, 02_{16}, 04_{16}, 06_{16}, 08_{16}, $0A_{16}$, $0E_{16}$, 10_{16})

b	Name	Functions	At reset	R	W
0	Port Pi_0	●In output mode Write •••••••• Port latch Read •••••••• Port latch ●In input mode Write •••••••• Port latch Read •••••••• Value of pin	0	○	○
1	Port Pi_1		0	○	○
2	Port Pi_2		0	○	○
3	Port Pi_3		0	○	○
4	Port Pi_4		0	○	○
5	Port Pi_5		0	○	○
6	Port Pi_6		0	○	○
7	Port Pi_7		0	○	○

Fig. 3.5.1 Structure of port Pi

Port Pi direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port Pi direction register (i = 0, 2, 4, 5, 7, 8)
(PiD: addresses 01_{16}, 05_{16}, 09_{16}, $0B_{16}$, $0F_{16}$, 11_{16})

b	Name	Functions	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 **(Note)**	0	×	○

Note: Bit 7 of the port P4 direction register (address 09_{16}) does not have direction register function because $P4_7$ is input port. When writing to bit 7 of the port P4 direction register, write "0" to the bit.

Fig. 3.5.2 Structure of port Pi direction register

Port P6

b7 b6 b5 b4 b3 b2 b1 b0

Port P6
(P6: address $0C_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P6_0$	●In output mode Write •••••••• Port latch Read •••••••• Port latch ●In input mode Write •••••••• Port latch Read •••••••• Value of pin	0	○	○
1	Port $P6_1$		0	○	○
2	Port $P6_2$		0	○	○
3	Port $P6_3$		0	○	○
4	Port $P6_4$		0	○	○
5	Port $P6_5$		0	○	○
6	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
7			0	×	×

Fig. 3.5.3 Structure of port P6

Port P6 direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port P6 direction register
(P6D: address $0D_{16}$)

b	Name	Functions	At reset	R	W
0	Port P6 direction register	0 : Port $P6_0$ input mode 1 : Port $P6_0$ output mode	0	×	○
1		0 : Port $P6_1$ input mode 1 : Port $P6_1$ output mode	0	×	○
2		0 : Port $P6_2$ input mode 1 : Port $P6_2$ output mode	0	×	○
3		0 : Port $P6_3$ input mode 1 : Port $P6_3$ output mode	0	×	○
4		0 : Port $P6_4$ input mode 1 : Port $P6_4$ output mode	0	×	○
5		0 : Port $P6_5$ input mode 1 : Port $P6_5$ output mode	0	×	○
6	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
7			0	×	×

Fig. 3.5.4 Structure of port P6 direction register

Port P9

b7 b6 b5 b4 b3 b2 b1 b0

Port P9
(P9: address 12_{16})

b	Name	Functions	At reset	R	W
0	Port $P9_0$	●In output mode Write •••••••• Port latch Read •••••••• Port latch ●In input mode Write •••••••• Port latch Read •••••••• Value of pin	0	○	○
1	Port $P9_1$		0	○	○
2	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
3			0	×	×
4			0	×	×
5			0	×	×
6			0	×	×
7			0	×	×

Fig. 3.5.5 Structure of port P9

Port P9 direction register

b7 b6 b5 b4 b3 b2 b1 b0

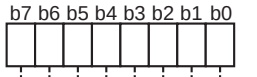
Port P9 direction register
(P9D: address 13_{16})

b	Name	Functions	At reset	R	W
0	Port P9 direction register	0 : Port $P9_0$ input mode 1 : Port $P9_0$ output mode	0	×	○
1		0 : Port $P9_1$ input mode 1 : Port $P9_1$ output mode	0	×	○
2	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
3			0	×	×
4			0	×	×
5			0	×	×
6			0	×	×
7			0	×	×

Fig. 3.5.6 Structure of port P9 direction register

3.5 Control registers

PWM register (high-order)

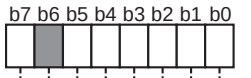


PWM register (high-order)
(PWMH: address 14_{16})

b	Functions	At reset	R	W
0	• High-order 8 bits of PWM_0 output data is set. • The values set in this register is transferred to the PWM latch each sub-period cycle (64 µs). (At $f(X_{IN})$ = 4 MHz) • When this register is read out, the value of the PWM register (high-order) is read out.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 3.5.7 Structure of PWM register (high-order)

PWM register (low-order)

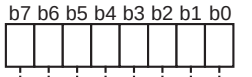


PWM register (low-order)
(PWML: address 15_{16})

b	Functions	At reset	R	W
0	• Low-order 6 bits of PWM_0 output data is set. • The values set in this register is transferred to the PWM latch at each PWM cycle period (4096 µs). (At $f(X_{IN})$ = 4 MHz) • When this register is read out, the value of the PWM latch (low-order 6 bits) is read out.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6	Nothing is arranged for this bit. This bit is a write disabled bit. When this bit is read out, the contents are “0”.	Undefined	○	×
7	• This bit indicates whether the transfer to the PWM latch is completed. 0: Transfer is completed 1: Transfer is not completed • This bit is set to “1” at writing.	Undefined	○	×

Fig. 3.5.8 Structure of PWM register (low-order)

Baud rate generator

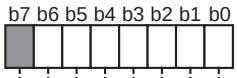


Baud rate generator
(BRG: address 16_{16})

b	Functions	At reset	R	W
0	• Bit rate of the serial transfer is determined. • This is the 8-bit counter and has the reload register. The count source is divided by n+1 owing to specifying a value n.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 3.5.9 Structure of baud rate generator

UART control register



UART control register
(UARTCON: address 17_{16})

b	Name	Functions	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	$P5_5$/TxD P-channel output disable bit (POFF)	0: CMOS output (in output mode) 1: N-channel open-drain output (in output mode)	0	○	○
5	BRG clock switch bit	0: X_{IN} or X_{CIN}/2 (depending on internal system clock) 1: X_{CIN}	0	○	○
6	Serial I/O2 clock I/O pin selection bit	0: S_{CLK21} ($P5_7$/S_{CLK22} pin is used as I/O port or $\overline{S_{RDY2}}$ output pin.) 1: S_{CLK22} ($P5_6$/S_{CLK21} pin is used as I/O port.)	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “1”.		1	○	×

Fig. 3.5.10 Structure of UART control register

Serial I/O1 automatic transfer data pointer

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O1 automatic transfer data pointer
(SIO1DP: address 18_{16})

b	Functions	At reset	R	W
0	• Indicates the low-order 8 bits of the address storing the start data on the serial I/O. automatic transfer RAM. • Data is written into the latch and read from the decrement counter.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 3.5.11 Structure of serial I/O1 automatic transfer data pointer

Serial I/O1 control register 1

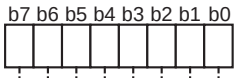
b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O1 control register 1
(SIO1CON1•SC11: address 19_{16})

b	Name	Functions	At reset	R	W
0	Serial transfer selection bits	b1b0 0 0: Serial I/O disabled (Pins $P6_2$, $P6_4$, $P6_5$, $P5_0$–$P5_3$ pins are I/O ports.)	0	○	○
1		0 1: 8-bit serial I/O 1 0: Not available 1 1: Automatic transfer serial I/O (8 bits)	0	○	○
2	Serial I/O1 synchronous clock selection bits ($P6_5$/S_{STB1} pin control bits)	b3b2 0 0: Internal synchronous clock ($P6_5$ pin is I/O port.) 0 1: External synchronous clock ($P6_5$ pin is I/O port.)	0	○	○
3		1 0: Internal synchronous clock ($P6_5$ pin is S_{STB1} output.) 1 1: Internal synchronous clock ($P6_5$ pin is S_{STB1} output.)	0	○	○
4	Serial I/O initialization bit	0: Serial I/O initialization 1: Serial I/O enabled	0	○	○
5	Transfer mode selection bit	0: Full-duplex (transmit/receive) mode ($P5_0$ pin is S_{IN1} input.) 1: Transmit-only mode ($P5_0$ pin is I/O port.)	0	○	○
6	Transfer direction selection bit	0: LSB first 1: MSB first	0	○	○
7	Serial I/O1 clock pin selection bit	0: S_{CLK11} ($P5_3$/S_{CLK12} pin is I/O port.) 1: S_{CLK12} ($P5_2$/S_{CLK11} pin is I/O port.)	0	○	○

Fig. 3.5.12 Structure of serial I/O1 control register 1

Serial I/O1 control register 2

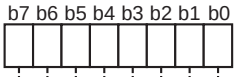


Serial I/O1 control register 2
(SIO1CON2 • SC12: address 1A16)

b	Name	Functions	At reset	R	W
0	P62/SRDY1 • P64/SBUSY1 pin control bits	b3b2b1b0 0 0 0 0: P62, P64 pins are I/O ports. 0 0 0 1: Not used 0 0 1 0: P62 pin is SRDY1 output; P64 pin is I/O port. 0 0 1 1: P62 pin is SRDY1 output; P64 pin is I/O port. 0 1 0 0: P62 pin is I/O port; P64 pin is SBUSY1 input. 0 1 0 1: P62 pin is I/O port; P64 pin is SBUSY1 input. 0 1 1 0: P62 pin is I/O port; P64 pin is SBUSY1 output. 0 1 1 1: P62 pin is I/O port; P64 pin is SBUSY1 output. 1 0 0 0: P62 pin is SRDY1 input; P64 pin is SBUSY1 output. 1 0 0 1: P62 pin is SRDY1 input; P64 pin is SBUSY1 output. 1 0 1 0: P62 pin is SRDY1 input; P64 pin is SBUSY1 output. 1 0 1 1: P62 pin is SRDY1 input; P64 pin is SBUSY1 output. 1 1 0 0: P62 pin is SRDY1 output; P64 pin is SBUSY1 input. 1 1 0 1: P62 pin is SRDY1 output; P64 pin is SBUSY1 input. 1 1 1 0: P62 pin is SRDY1 output; P64 pin is SBUSY1 input. 1 1 1 1: P62 pin is SRDY1 output; P64 pin is SBUSY1 input.	0	○	○
1			0	○	○
2			0	○	○
3			0	○	○
4	SBUSY1 output • SSTB1 output function selection bit (Valid in serial I/O1 automatic transfer mode)	0: Functions as signal for each 1-byte 1: Functions as signal for each transfer data set	0	○	○
5	Serial transfer status flag	0: Serial transfer completed 1: Serial transfer in-progress	0	○	×
6	SOUT1 pin control bit (when serial data is not transferred)	0: Output active 1: Output high-impedance	0	○	○
7	P51/SOUT1 P-channel output disable bit	0: CMOS 3 state (P-channel output is valid.) 1: N-channel open-drain output (P-channel output is invalid.)	0	○	○

Fig. 3.5.13 Structure of serial I/O1 control register 2

Serial I/O1 register/Transfer counter



Serial I/O1 register/Transfer counter
(SIO1: address $1B_{16}$)

b	Name	Functions	At reset	R	W
0	•In 8-bit serial I/O mode: Serial I/O1 register •In automatic transfer serial I/O mode: Transfer counter	•At function as serial I/O1 register: This register becomes the shift register to perform serial transmit/reception. Set transmit data to this register. The serial transfer is started by writing the transmit data. •At function as transfer counter: Set (transfer byte number – 1) to this register. When selecting an internal clock, the automatic transfer is started by writing the transmit data. (When selecting an external clock, after writing a value to this register, wait for 5 or more cycles of the internal system clock before inputting the transfer clock to the S_{CLK1} pin.)	Undefined	O	O
1			Undefined	O	O
2			Undefined	O	O
3			Undefined	O	O
4			Undefined	O	O
5			Undefined	O	O
6			Undefined	O	O
7			Undefined	O	O

Fig. 3.5.14 Structure of serial I/O1 register/Transfer counter

Serial I/O1 control register 3

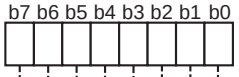
b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O1 control register 3
(SIO1CON3 • SC13: address 1C16)

b	Name	Functions	At reset	R	W
0	Automatic transfer interval set bits (valid only when selecting internal synchronous clock)	b4b3b2b1b0 0 0 0 0 0: 2 cycles of transfer clock 0 0 0 0 1: 3 cycles of transfer clock to 1 1 1 1 0: 32 cycles of transfer clock 1 1 1 1 1: 33 cycles of transfer clock Data is written into the latch and read from the decrement counter.	0	○	○
1			0	○	○
2			0	○	○
3			0	○	○
4			0	○	○
5	Internal synchronous clock selection bits	b7b6b5 0 0 0 : f(XIN)/4 or f(XCIN)/8 0 0 1 : f(XIN)/8 or f(XCIN)/16 0 1 0 : f(XIN)/16 or f(XCIN)/32 0 1 1 : f(XIN)/32 or f(XCIN)/64 1 0 0 : f(XIN)/64 or f(XCIN)/128 1 0 1 : f(XIN)/128 or f(XCIN)/256 1 1 0 : f(XIN)/256 or f(XCIN)/512 1 1 1 : Not used	0	○	○
6			0	○	○
7			0	○	○

Fig. 3.5.15 Structure of serial I/O1 control register 3

Serial I/O2 control register

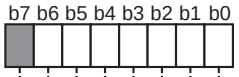


Serial I/O2 control register (SIO2CON: address $1D_{16}$)

b	Name	Functions	At reset	R	W
0	BRG count source selection bit (CSS)	0: f(X_{IN}) or f(X_{CIN})/2 or f(X_{CIN}) 1: f(X_{IN})/4 or f(X_{CIN})/8 or f(X_{CIN})/4	0	○	○
1	Serial I/O2 synchronous clock selection bit (SCS)	•In clock synchronous mode 0: BRG output/4 1: External clock input •In UART mode 0: BRG output/16 1: External clock input/16	0	○	○
2	$\overline{SRDY2}$ output enable bit (SRDY)	0: $P5_7$ pin operates as normal I/O pin 1: $\overline{P5_7}$ pin operates as $\overline{SRDY2}$ 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/O2 mode selection bit (SIOM)	0: Clock asynchronous serial I/O (UART) mode 1: Clock synchronous serial I/O mode	0	○	○
7	Serial I/O2 enable bit (SIOE)	0: Serial I/O2 disabled (pins $P5_4$–$P5_7$ operate as normal I/O pins) 1: Serial I/O2 enabled (pins $P5_4$–$P5_7$ operate as serial I/O pins)	0	○	○

Fig. 3.5.16 Structure of serial I/O2 control register

Serial I/O2 status register

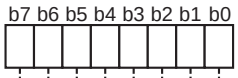


Serial I/O2 status register
(SIO2STS: address $1E_{16}$)

b	Name	Functions	At reset	R	W
0	Transmit buffer empty flag (TBE)	0: Buffer full 1: Buffer empty	0	○	○
1	Receive buffer full flag (RBF)	0: Buffer empty 1: Buffer full	0	○	○
2	Transmit shift register shift completion flag (TSC)	0: Transmit shift in progress 1: Transmit shift completed	0	○	○
3	Overrun error flag (OE)	0: No error 1: Overrun error	0	○	○
4	Parity error flag (PE)	0: No error 1: Parity error	0	○	○
5	Framing error flag (FE)	0: No error 1: Framing error	0	○	○
6	Summing error flag (SE)	0: (OE) U (PE) U (FE) = 0 1: (OE) U (PE) U (FE) = 1	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “1”.		1	○	×

Fig. 3.5.17 Structure of serial I/O2 status register

Serial I/O2 transmit/receive buffer register

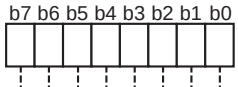


Serial I/O2 transmit/receive buffer register
(TB/RB: address $1F_{16}$)

b	Functions	At reset	R	W
0	This is the buffer register which is used to write transmit data or to read receive data. • At write : The value is written to the transmit buffer register. The value cannot be written to the receive buffer register. • At read : The contents of the receive buffer register is read out. When a character bit length is 7 bits, the MSB of data stored in the receive buffer is “0”. The contents of the transmit buffer register cannot be read out.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 3.5.18 Structure of serial I/O2 transmit/receive buffer register

Timer i

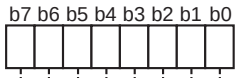


Timer i (i = 1, 3, 4, 5, 6)
(Ti: addresses 20_{16}, 22_{16}, 23_{16}, 24_{16}, 25_{16})

b	Functions	At reset	R	W
0	• Set timer i count value. • The value set in this register is written to both the timer i and the timer i latch at one time. • When the timer i is read out, the count value of the timer i is read out.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Fig. 3.5.19 Structure of timer i

Timer 2

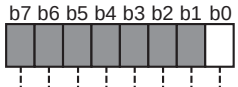


Timer 2
(T2: address 21_{16})

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

Fig. 3.5.20 Structure of timer 2

PWM control register



PWM control register
(PWMCON: address 26_{16})

b	Name	Functions	At reset	R	W
0	P87/PWM output selection bit	0: I/O port 1: PWM output	0	○	○
1	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	○	×
2			0	○	×
3			0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Fig. 3.5.21 Structure of PWM control register

Timer 6 PWM register

b7 b6 b5 b4 b3 b2 b1 b0

Timer 6 PWM register
(T6PWM: address 27_{16})

b	Functions	At reset	R	W
0	• In timer 6 PWM_1 mode "L" level width of PWM rectangular waveform is set. • Duty of PWM rectangular waveform: n/(n + m) Period: (n + m) × ts n = timer 6 set value m = timer 6 PWM register set value ts = timer 6 count source period At n = 0, all PWM output "L". At m = 0, all PWM output "H". (However, n = 0 has priority.) • Selection of timer 6 PWM_1 mode Set "1" to the timer 6 operation mode selection bit.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 3.5.22 Structure of timer 6 PWM register

Timer 12 mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer 12 mode register
(T12M: address 28_{16})

b	Name	Functions	At reset	R	W
0	Timer 1 count stop bit	0: Count operation 1: Count stop	0	○	○
1	Timer 2 count stop bit	0: Count operation 1: Count stop	0	○	○
2	Timer 1 count source selection bits	b3 b2 0 0: f(X_{IN})/8 or f(X_{CIN})/16 0 1: f(X_{CIN}) 1 0: f(X_{IN})/16 or f(X_{CIN})/32 1 1: f(X_{IN})/64 or f(X_{CIN})/128	0	○	○
3					
4	Timer 2 count source selection bits	b5 b4 0 0: Timer 1 underflow 0 1: f(X_{CIN}) 1 0: External count input $CNTR_0$ 1 1: Not available	0	○	○
5			0	○	○
6	Timer 1 output selection bit ($P4_5$)	0: I/O port 1: Timer 1 output	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are "0".		0	○	×

Fig. 3.5.23 Structure of timer 12 mode register

Timer 34 mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer 34 mode register
(T34M: address 29_{16})

b	Name	Functions	At reset	R	W
0	Timer 3 count stop bit	0: Count operation 1: Count stop	0	○	○
1	Timer 4 count stop bit	0: Count operation 1: Count stop	0	○	○
2 3	Timer 3 count source selection bits	b3 b2 0 0: f(XIN)/8 or f(XCIN)/16 0 1: Timer 2 underflow 1 0: f(XIN)/16 or f(XCIN)/32 1 1: f(XIN)/64 or f(XCIN)/128	0	○	○
4 5	Timer 4 count source selection bits	b5 b4 0 0: f(XIN)/8 or f(XCIN)/16 0 1: Timer 3 underflow 1 0: External count input CNTR1 **(Note)** 1 1: Not available	0 0	○ ○	○ ○
6	Timer 3 output selection bit (P46)	0: I/O port 1: Timer 3 output	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×

Note: In the mask option type P, CNTR1 function cannot be used.

Fig. 3.5.24 Structure of timer 34 mode register

Timer 56 mode register

b7 b6 b5 b4 b3 b2 b1 b0

Timer 56 mode register
(T56M: address $2A_{16}$)

b	Name	Functions	At reset	R	W
0	Timer 5 count stop bit	0: Count operation 1: Count stop	0	○	○
1	Timer 6 count stop bit	0: Count operation 1: Count stop	0	○	○
2	Timer 5 count source selection bit	0: f(XIN)/8 or f(XCIN)/16 1: Timer 4 underflow	0	○	○
3	Timer 6 operation mode selection bit	0: Timer mode 1: PWM mode	0	○	○
4 5	Timer 6 count source selection bits	b5 b4 0 0: f(XIN)/8 or f(XCIN)/16 0 1: Timer 5 underflow 1 0: Timer 4 underflow 1 1: Not available	0 0	○ ○	○ ○
6	Timer 6 (PWM) output selection bit (P44)	0: I/O port 1: Timer 6 output	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×

Fig. 3.5.25 Structure of timer 56 mode register

Watchdog timer control register

b7 b6 b5 b4 b3 b2 b1 b0

Watchdog timer control register
(WDTCON: address $2B_{16}$)

b	Name	Functions	At reset	R	W
0	Watchdog timer H (high-order 6 bits of reading exclusive)		1	○	×
1			1	○	×
2			1	○	×
3			1	○	×
4			1	○	×
5			1	○	×
6	STP instruction disable bit	0: STP instruction enabled 1: STP instruction disabled	0	○	○
7	Watchdog timer H count source selection bit	0: Watchdog timer L underflow 1: f(X_{IN})/16 or f(X_{CIN})/16	0	○	○

Fig. 3.5.26 Structure of watchdog timer control register

Timer X (low-order, high-order)

b7 b6 b5 b4 b3 b2 b1 b0

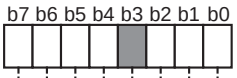
Timer X (low-order, high-order)
(TXL, TXH: addresses $2C_{16}$, $2D_{16}$)

b	Functions	At reset	R	W
0	• Set timer X count value. • When the timer X write control bit of the timer X mode register 1 is "0", the value is written to timer X and the latch at one time. When the timer X write control bit of the timer X mode register 1 is "1", the value is written only to the latch. • The timer X count value is read out by reading this register.	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Notes 1: When reading and writing, perform them to both the high-order and low-order bytes.
2: Read both registers in order of TXH and TXL following.
3: Write both registers in order of TXL and TXH following.
4: Do not read both registers during a write, and do not write to both registers during a read.

Fig. 3.5.27 Structure of timer X (low-order, high-order)

Timer X mode register 1

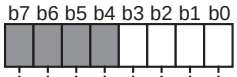


Timer X mode register 1
(TXM1: address 2E16)

b	Name	Functions	At reset	R	W
0	Timer X write control bit	0 : Write value in latch and counter 1 : Write value in latch only	0	○	○
1	Timer X count source selection bits	b2 b1 0 0: f(XIN)/2 or f(XCIN)/4 0 1: f(XIN)/8 or f(XCIN)/16 1 0: f(XIN)/64 or f(XCIN)/128 1 1: Not available	0	○	○
2			0	○	○
3	Nothing is arranged for this bit. This is write disabled bit. When this bit is read out, the contents are "0".		0	○	×
4	Timer X operating mode bits	b5 b4 0 0 : Timer mode 0 1 : Pulse output mode 1 0 : Event counter mode 1 1 : Pulse width measurement mode	0	○	○
5			0	○	○
6	CNTR2 active edge switch bit	0 : •Count at rising edge in event counter mode •Start from "H" output in pulse output mode •Measure "H" pulse width in pulse width measurement mode 1 : •Count at falling edge in event counter mode •Start from "L" output in pulse output mode •Measure "L" pulse width in pulse width measurement mode	0	○	○
7	Timer X stop control bit	0 : Count operating 1 : Count stop	0	○	○

Fig. 3.5.28 Structure of timer X mode register 1

Timer X mode register 2

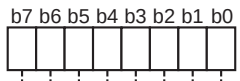


Timer X mode register 2
(TXM2: address $2F_{16}$)

b	Name	Functions	At reset	R	W
0	Real time port control bit ($P8_5$)	0: Real time port function is invalid 1: Real time port function is valid	0	○	○
1	Real time port control bit ($P8_6$)	0: Real time port function is invalid 1: Real time port function is valid	0	○	○
2	$P8_5$ data for real time port	0: "L" output 1: "H" output	0	○	○
3	$P8_6$ data for real time port	0: "L" output 1: "H" output	0	○	○
4	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are "0".		0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Fig. 3.5.29 Structure of timer X mode register 2

Interrupt interval determination register



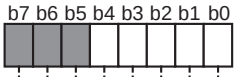
Interrupt interval determination register
(IID: address 30_{16})

b	Functions	At reset	R	W
0	• This register stores a value which is obtained by counting a following interval with the counter sampling clock. Rising interval Falling interval Both edges interval **(Note)** (Selected by interrupt edge selection register) • Read exclusive register	Undefined	○	×
1		Undefined	○	×
2		Undefined	○	×
3		Undefined	○	×
4		Undefined	○	×
5		Undefined	○	×
6		Undefined	○	×
7		Undefined	○	×

Note: When the noise filter sampling clock selection bits (bits 2, 3) of the interrupt interval determination control register is "00", the both-sided edge detection function cannot be used.

Fig. 3.5.30 Structure of interrupt interval determination register

Interrupt interval determination control register



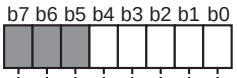
Interrupt interval determination control register
(IIDCON: address 31_{16})

b	Name	Functions	At reset	R	W
0	Interrupt interval determination circuit operating selection bit	0: Stopped 1: Operating	0	○	○
1	Counter sampling clock selection bit	0: f(X_{IN})/128 1: f(X_{IN})/256	0	○	○
2	Noise filter sampling clock selection bits (INT_2)	b3 b2 0 0: Filter is not used. 0 1: f(X_{IN})/32 1 0: f(X_{IN})/64 1 1: f(X_{IN})/128	0	○	○
3			0	○	○
4	One-sided/both-sided edge detection selection bit	0: One-sided edge detection 1: Both-sided edge detection **(Note)**	0	○	○
5	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are "0".		0	○	×
6			0	○	×
7			0	○	×

Note: When the noise filter sampling clock selection bits (bits 2, 3) is "00", the both-sided edge detection function cannot be used.

Fig. 3.5.31 Structure of interrupt interval determination control register

A-D control register



A-D control register
(ADCON: address 32_{16})

b	Name	Functions	At reset	R	W
0	Analog input pin selection bits	b3 b2 b1 b0 0 0 0 0: $P7_0/AN_0$ 0 0 0 1: $P7_1/AN_1$ 0 0 1 0: $P7_2/AN_2$ 0 0 1 1: $P7_3/AN_3$ 0 1 0 0: $P7_4/AN_4$ 0 1 0 1: $P7_5/AN_5$ 0 1 1 0: $P7_6/AN_6$ 0 1 1 1: $P7_7/AN_7$ 1 0 0 0: $P6_2/SRDY_1/AN_8$ 1 0 0 1: $P6_3/AN_9$ 1 0 1 0: $P6_4/INT_4/SBUSY_1/AN_{10}$ 1 0 1 1: $P6_5/SSTB_1/AN_{11}$	0	○	○
1			0	○	○
2			0	○	○
3			0	○	○
4	AD conversion completion bit	0: Conversion in progress 1: Conversion completed	1	○	○
5	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are "0".		0	○	×
6			0	○	×
7			0	○	×

Fig. 3.5.32 Structure of A-D control register

A-D conversion register (low-order)

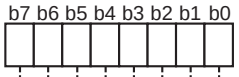
A-D conversion register (low-order)
(ADL: address 33_{16})

b	Functions	At reset	R	W
0	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are "0".	Undefined	○	×
1		Undefined	○	×
2		Undefined	○	×
3		Undefined	○	×
4		Undefined	○	×
5		Undefined	○	×
6	These are A-D conversion result (low-order 2 bits) stored bits. This is read exclusive register.	Undefined	○	×
7		Undefined	○	×

Note: Do not read this register during A-D conversion.

Fig. 3.5.33 Structure of A-D conversion register (low-order)

A-D conversion register (high-order)



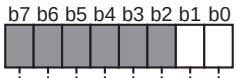
A-D conversion register (high-order)
(ADH: address 34_{16})

b	Functions	At reset	R	W
0	This is A-D conversion result (high-order 8 bits) stored bits. This is read exclusive register.	Undefined	○	×
1		Undefined	○	×
2		Undefined	○	×
3		Undefined	○	×
4		Undefined	○	×
5		Undefined	○	×
6		Undefined	○	×
7		Undefined	○	×

Note: Do not read this register during A-D conversion.

Fig. 3.5.34 Structure of A-D conversion register (high-order)

Interrupt source switch register



Interrupt source switch register
(IFR: address 39_{16})

b	Name	Functions	At reset	R	W
0	INT_3/serial I/O2 transmit interrupt switch bit **(Note)**	0: INT_3 intrrupt 1: Serial I/O2 transmit interrupt	0	○	○
1	INT_4/A-D conversion interrupt switch bit	0: INT_4 interrupt 1: A-D conversion intrerrupt	0	○	○
2	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	○	×
3			0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Note: In the mask option type P, this bit is not available because INT_3 function cannot be used.

Fig. 3.5.35 Structure of interrupt source switch register

Interrupt edge selection register

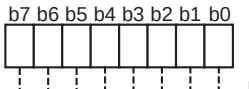
Interrupt edge selection register
(INTEDGE : address $3A_{16}$)

b	Name	Functions	At reset	R	W
0	INT_0 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
1	INT_1 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
2	INT_2 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
3	INT_3 interrupt edge selection bit **(Note)**	0 : Falling edge active 1 : Rising edge active	0	○	○
4	INT_4 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	○	○
5	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×
6	$CNTR_0$ pin edge switch bit	0 : Rising edge count 1 : Falling edge count	0	○	○
7	$CNTR_1$ pin edge switch bit **(Note)**	0 : Rising edge count 1 : Falling edge count	0	○	○

Note: In the mask option type P, these bits are not available because $CNTR_1$ function and INT_3 function cannot be used.

Fig. 3.5.36 Structure of interrupt edge selection register

CPU mode register

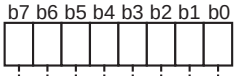


CPU mode register
(CPUM, CM: address $3B_{16}$)

b	Name	Functions	At reset	R	W
0	Processor mode bits	b1 b0 00 : Single-chip mode 01 : 10 : } Not available 11 :	0	○	○
1			0	○	○
2	Stack page selection bit	0 : Page 0 1 : Page 1	0	○	○
3	X_{COUT} drivability selection bit	0: Low drive 1: High drive	1	○	○
4	Port Xc switch bit	0: I/O port function 1: X_{CIN}-X_{COUT} oscillation function	0	○	○
5	Main clock (X_{IN}-X_{OUT}) stop bit	0: Oscillating 1: Stopped	0	○	○
6	Main clock division ratio selection bit	0: f(X_{IN}) (high-speed mode) 1: f(X_{IN})/4 (middle-speed mode)	1	○	○
7	Internal system clock selection bit	0: X_{IN}–X_{OUT} selection (middle-/high-speed mode) 1: X_{CIN}–X_{COUT} selection (low-speed mode)	0	○	○

Fig. 3.5.37 Structure of CPU mode register

Interrupt request register 1



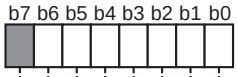
Interrupt request register 1
(IREQ1 : address $3C_{16}$)

b	Name	Functions	At reset	R	W
0	INT_0 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	INT_1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	INT_2 interrupt request bit Remote controller /counter overflow interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O1 interrupt request bit Serial I/O automatic transfer interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	Timer 2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Timer 3 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻

✻: "0" can be set by software, but "1" cannot be set.

Fig. 3.5.38 Structure of interrupt request register 1

Interrupt request register 2



Interrupt request register 2
(IREQ2 : address $3D_{16}$)

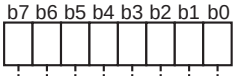
b	Name	Functions	At reset	R	W
0	Timer 4 interrupt request bit **(Note)**	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
1	Timer 5 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
2	Timer 6 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
3	Serial I/O2 receive interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
4	INT_3/Serial I/O2 transmit interrupt request bit **(Note)**	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
5	INT_4 interrupt request bit A-D converter interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
6	FLD blanking interrupt request bit FLD digit interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	○	✻
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×

✻: “0” can be set by software, but “1” cannot be set.

Note: In the mask option type P, if timer 4 interrupt whose count source is $CNTR_1$ input and INT_3 interrupt are selected, these bits do not become “1”.

Fig. 3.5.39 Structure of interrupt request register 2

Interrupt control register 1

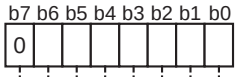


Interrupt control register 1
(ICON1 : address $3E_{16}$)

b	Name	Functions	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	INT_2 interrupt enable bit Remote controller /counter overflow interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O1 interrupt enable bit Serial I/O automatic transfer interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
4	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
6	Timer 2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○
7	Timer 3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	○	○

Fig. 3.5.40 Structure of interrupt control register 1

Interrupt control register 2



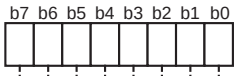
Interrupt control register 2
(ICON2 : address $3F_{16}$)

b	Name	Functions	At reset	R	W
0	Timer 4 interrupt enable bit **(Note)**	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
1	Timer 5 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
2	Timer 6 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
3	Serial I/O2 receive interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
4	INT_3/Serial I/O2 transmit interrupt enable bit **(Note)**	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
5	INT_4 interrupt enable bit A-D converter interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
6	FLD blanking interrupt enable bit FLD digit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	○	○
7	Fix “0” to this bit.		0	○	○

Note: In the mask option type P, timer 4 interrupt whose count source is $CNTR_1$ input and INT_3 interrupt are not available.

Fig. 3.5.41 Structure of interrupt control register 2

Pull-up control register 1



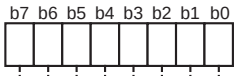
Pull-up control register 1
(PULL1: address 0EF0$_{16}$)

b	Name	Functions	At reset	R	W
0	Ports P5$_0$, P5$_1$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
1	Ports P5$_2$, P5$_3$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
2	Ports P5$_4$, P5$_5$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
3	Ports P5$_6$, P5$_7$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
4	Port P6$_1$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
5	Ports P6$_2$, P6$_3$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
6	Ports P6$_4$, P6$_5$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×

Note: The pin set to output port is cut off from pull-up control.

Fig. 3.5.42 Structure of pull-up control register 1

Pull-up control register 2



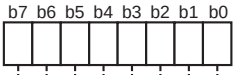
Pull-up control register 2
(PULL2: address 0EF1$_{16}$)

b	Name	Functions	At reset	R	W
0	Ports P7$_0$, P7$_1$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
1	Ports P7$_2$, P7$_3$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
2	Ports P7$_4$, P7$_5$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
3	Ports P7$_6$, P7$_7$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
4	Ports P8$_4$, P8$_5$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
5	Ports P8$_6$, P8$_7$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
6	Ports P9$_0$, P9$_1$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×

Note: The pin set to output port is cut off from pull-up control.

Fig. 3.5.43 Structure of pull-up control register 2

P1FLDRAM write disable register

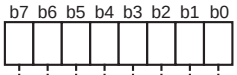


P1FLDRAM write disable register
(P1FLDRAM: address 0EF2$_{16}$)

b	Name	Functions	At reset	R	W
0	FLDRAM corresponding to port P1$_0$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
1	FLDRAM corresponding to port P1$_1$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
2	FLDRAM corresponding to port P1$_2$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
3	FLDRAM corresponding to port P1$_3$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
4	FLDRAM corresponding to port P1$_4$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
5	FLDRAM corresponding to port P1$_5$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
6	FLDRAM corresponding to port P1$_6$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
7	FLDRAM corresponding to port P1$_7$ write disable bit	0: Operating normally 1: Write disabled	0	○	○

Fig. 3.5.44 Structure of P1FLDRAM write disable register

P3FLDRAM write disable register

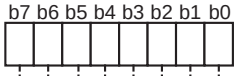


P3FLDRAM write disable register
(P3FLDRAM: address $0EF3_{16}$)

b	Name	Functions	At reset	R	W
0	FLDRAM corresponding to port $P3_0$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
1	FLDRAM corresponding to port $P3_1$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
2	FLDRAM corresponding to port $P3_2$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
3	FLDRAM corresponding to port $P3_3$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
4	FLDRAM corresponding to port $P3_4$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
5	FLDRAM corresponding to port $P3_5$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
6	FLDRAM corresponding to port $P3_6$ write disable bit	0: Operating normally 1: Write disabled	0	○	○
7	FLDRAM corresponding to port $P3_7$ write disable bit	0: Operating normally 1: Write disabled	0	○	○

Fig. 3.5.45 Structure of P3FLDRAM write disable register

FLDC mode register



FLDC mode register
(FLDM: address 0EF4$_{16}$)

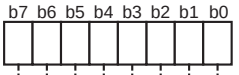
b	Name	Functions	At reset	R	W
0	Automatic display control bit (P0, P1, P2, P3, P8)	0 : General-purpose mode 1 : Automatic display mode	0	○	○
1	Display start bit	0 : Display stopped 1 : Display in progress (display starts by writing "1")	0	○	○
2	Tscan control bits	b3 b2 0 0 : 0 FLD digit interrupt (at rising edge of each digit) 0 1 : 1 × Tdisp 1 0 : 2 × Tdisp 1 1 : 3 × Tdisp	0	○	○
3		FLD blanking interrupt (at falling edge of last digit)	0	○	○
4	Timing number control bit	0 : 16 timing mode 1 : 32 timing mode **(Note 2)**	0	○	○
5	Gradation display mode selection control bit	0 : Not selected 1 : Selected **(Notes 1, 2)**	0	○	○
6	Tdisp counter count source selection bit	0 : f(XIN)/16 or f(XCIN)/32 1 : f(XIN)/64 or f(XCIN)/128	0	○	○
7	High-breakdown voltage port drivability selection bit	0 : Drivability strong 1 : Drivability weak	0	○	○

Notes 1: When the gradation display mode is selected, the number of timing is max. 16 timing. (Set "0" to the timing number control bit (b4).)

2: When switching the timing number control bit (b4) or the gradation display mode selection control bit (b5), set "0" to the display start bit (b1) (display stop state) before that.

Fig. 3.5.46 Structure of FLDC mode register

Tdisp time set register

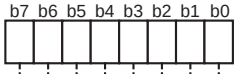


Tdisp time set register
(TDISP: address $0EF5_{16}$)

b	Functions	At reset	R	W
0	•Set the Tdisp time. •When a value n is written to this register, Tdisp time is expressed as Tdisp = (n + 1) × count source. •When reading this register, the value in the counter is read out. (Example) When the following condition is satisfied, Tdisp becomes 804 μs {(200 + 1) × 4 μs}; •f(X_{IN}) = 4 MHz •bit 6 of FLDC mode register = 0 (f(X_{IN})/16 is selected as Tdisp counter count source.) •Tdisp time set register = 200 ($C8_{16}$).	0	○	○
1		0	○	○
2		0	○	○
3		0	○	○
4		0	○	○
5		0	○	○
6		0	○	○
7		0	○	○

Fig. 3.5.47 Structure of Tdisp time set register

Toff1 time set register



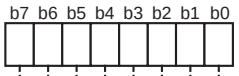
Toff1 time set register
(TOFF1: address 0EF6$_{16}$)

b	Functions	At reset	R	W
0	•Set the Toff1 time. •When a value n1 is written to this register, Toff1 time is expressed as Toff1 = n1 × count source. (Example) When the following condition is satisfied, Toff1 becomes 120 µs (= 30 × 4 µs); •f(X$_{IN}$) = 4 MHz •bit 6 of FLDC mode register = 0 (f(X$_{IN}$)/16 is selected as Tdisp counter count source.) •Toff1 time set register = 30 (1E$_{16}$).	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Note: Set value of 03$_{16}$ or more.

Fig. 3.5.48 Structure of Toff1 time set register

Toff2 time set register



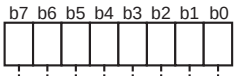
Toff2 time set register
(TOFF2: address 0EF7$_{16}$)

b	Functions	At reset	R	W
0	•Set the Toff2 time. •When a value n2 is written to this register, Toff2 time is expressed as Toff2 = n2 × count source. However, setting of Toff2 time is valid only for the FLD port which is satisfied the following; •gradation display mode •value of FLD automatic display RAM (in gradation display mode) = "1" (dark display). (Example) When the following condition is satisfied, Toff2 becomes 720 µs (= 180 × 4 µs); •f(X$_{IN}$) = 4 MHz •bit 6 of FLDC mode register = 0 (f(X$_{IN}$)/16 is selected as Tdisp counter count source.) •Toff2 time set register = 180 (B4$_{16}$).	1	○	○
1		1	○	○
2		1	○	○
3		1	○	○
4		1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Note: When the Toff2 control bit (b7) of the port P8FLD output control register (address 0EFC$_{16}$) is set to "1", set value of 03$_{16}$ or more to the Toff2 control register.

Fig. 3.5.49 Structure of Toff2 time set register

FLD data pointer/FLD data pointer reload register

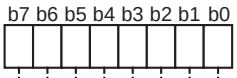


FLD data pointer/FLD data pointer reload register
(FLDDP: address 0EF8$_{16}$)

b	Functions	At reset	R	W
0	The start address of each data of FLD ports P0, P1, P2, P3, and P8, which is transferred from FLD automatic display RAM, is set to this register. The start address becomes the address adding the value set to this register into the last data address of each FLD port. Set a value of (timing number – 1) to this register. The value which is set to this address is written to the FLD data pointer reload register. When reading data from this address, the value in the FLD data pointer is read. When bits 5 to 7 of this register is read, “0” is always read.	Undefined	○	○
1		Undefined	○	○
2		Undefined	○	○
3		Undefined	○	○
4		Undefined	○	○
5		Undefined	○	○
6		Undefined	○	○
7		Undefined	○	○

Fig. 3.5.50 Structure of FLD data pointer/FLD data pointer reload register

Port P0FLD/port switch register

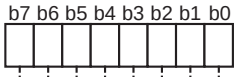


Port P0FLD/port switch register
(P0FPR: address 0EF9$_{16}$)

b	Name	Functions	At reset	R	W
0	Port P0$_0$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
1	Port P0$_1$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
2	Port P0$_2$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
3	Port P0$_3$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
4	Port P0$_4$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
5	Port P0$_5$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
6	Port P0$_6$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
7	Port P0$_7$FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○

Fig. 3.5.51 Structure of port P0FLD/Port switch register

Port P2FLD/port switch register

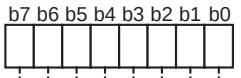


Port P2FLD/port switch register
(P2FPR: address 0EFA16)

b	Name	Functions	At reset	R	W
0	Port P20FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
1	Port P21FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
2	Port P22FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
3	Port P23FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
4	Port P24FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
5	Port P25FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
6	Port P26FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
7	Port P27FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○

Fig. 3.5.52 Structure of port P2FLD/port switch register

Port P8FLD/port switch register

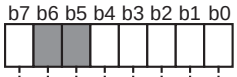


Port P8FLD/port switch register
(P8FPR: address 0EFB16)

b	Name	Functions	At reset	R	W
0	Port P80FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
1	Port P81FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
2	Port P82FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
3	Port P83FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
4	Port P84FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
5	Port P85FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
6	Port P86FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○
7	Port P87FLD/port switch bit	0 : General-purpose port 1 : FLD port	0	○	○

Fig. 3.5.53 Structure of port P8FLD/port switch register

Port P8FLD output control register



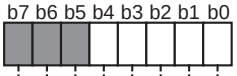
Port P8FLD output control register
(P8FLDCON : address 0EFC16)

b	Name	Functions	At reset	R	W
0	P84–P87 FLD output reverse bit	0 : Output normally 1 : Reverse output	0	○	○
1	P84–P87/FLDRAM write disable bit	0 : Operating normally 1 : Write disabled	0	○	○
2	P84–P87 Toff invalid bit	0 : Operating normally 1 : Toff invalid	0	○	○
3	P84–P87 delay control bit **(Note)**	0 : No delay 1 : Delay	0	○	○
4	P63/AN9 dimmer output control bit	0 : Ordinary port 1 : Dimmer output	0	○	○
5	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	○	×
6			0	○	×
7	Toff2 control bit	0 : Operating normally (falling operation) 1 : Rising operation	0	○	○

Note: Valid only when selecting FLD port and P84–P87 Toff invalid function

Fig. 3.5.54 Structure of port P8FLD output control register

Buzzer output control register



Buzzer output control register
(BUZCON: address 0EFD16)

b	Name	Functions	At reset	R	W
0	Output frequency selection bits	b1b0 0 0: 1 kHz (f(XIN)/4096) 0 1: 2 kHz (f(XIN)/2048) 1 0: 4 kHz (f(XIN)/1024) 1 1: Not available	0	○	○
1			0	○	○
2	Output port selection bits	b3b2 0 0: P20 and P43 function as ordinary ports. 0 1: P43/BUZ01 functions as a buzzer output. 1 0: P20/BUZ02/FLD0 functions as a buzzer output. 1 1: Not available	0	○	○
3			0	○	○
4	Buzzer output ON/OFF bit	0: Buzzer output OFF (“0” output) 1: Buzzer output ON	0	○	○
5	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	○	×
6			0	○	×
7			0	○	×

Fig. 3.5.55 Structure of buzzer output control register

3.6 Mask ROM confirmation form

GZZ-SH54-19B<88A1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M38B57M6-XXXFP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

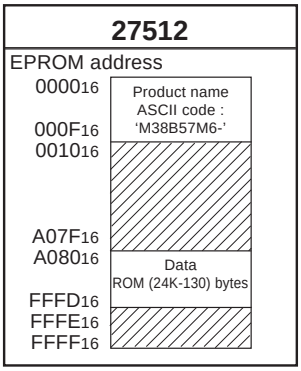
Note : Please fill in all items marked ✵.

✵ 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.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type



In the address space of the microcomputer, the internal ROM area is from address A080₁₆ to FFFD₁₆. The reset vector is stored in addresses FFFC₁₆ and FFFD₁₆.

(1) Set the data in the unused area (the shaded area of the diagram) to "FF₁₆".
(2) The ASCII codes of the product name "M38B57M6–" must be entered in addresses 0000₁₆ to 0008₁₆. And set the data "FF₁₆" in addresses 0009₁₆ to 000F₁₆.
The ASCII codes and addresses are listed to the right in hexadecimal notation.

Address		Address	
0000₁₆	'M' = 4D₁₆	0008₁₆	'–' = 2D₁₆
0001₁₆	'3' = 33₁₆	0009₁₆	FF₁₆
0002₁₆	'8' = 38₁₆	000A₁₆	FF₁₆
0003₁₆	'B' = 42₁₆	000B₁₆	FF₁₆
0004₁₆	'5' = 35₁₆	000C₁₆	FF₁₆
0005₁₆	'7' = 37₁₆	000D₁₆	FF₁₆
0006₁₆	'M' = 4D₁₆	000E₁₆	FF₁₆
0007₁₆	'6' = 36₁₆	000F₁₆	FF₁₆

(1/2)

3.6 Mask ROM confirmation form

GZZ-SH54-19B<88A1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M38B57M6-XXXFP
MITSUBISHI ELECTRIC

We recommend the use of the following pseudo-command to set the start address of the assembler source program because ASCII codes of the product name are written to addresses 0000_{16} to 0008_{16} of EPROM.

EPROM type	27512
The pseudo-command	*= △$0000 .BYTE △'M38B57M6–'

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 (80P6N) and attach it to the mask ROM confirmation form.

✺ 3. Usage conditions

Please answer the following questions about usage for use in our product inspection :

(1) How will you use the XIN-XOUT oscillator?

☐ Ceramic resonator ☐ Quartz crystal

☐ External clock input ☐ Other ( )

At what frequency? f(XIN) = ☐ MHz

(2) How will you use the XCIN-XCOUT oscillator?

☐ Ceramic resonator ☐ Quartz crystal

☐ External clock input ☐ Other ( )

At what frequency? f(XCIN) = ☐ kHz

✺ 4. Comments

(2/2)

GZZ-SH54-20B<88A1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M38B57MCH-☐XXXFP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

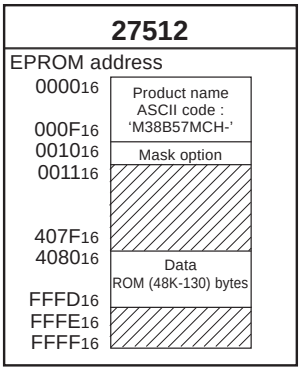
✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date:			

✵ 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.

Checksum code for entire EPROM ☐☐☐☐ (hexadecimal notation)

EPROM type



In the address space of the microcomputer, the internal ROM area is from address 4080₁₆ to FFFD₁₆. The reset vector is stored in addresses FFFC₁₆ and FFFD₁₆.

(1) Set the data in the unused area (the shaded area of the diagram) to "FF₁₆".
(2) The ASCII codes of the product name "M38B57MCH–" must be entered in addresses 0000₁₆ to 0009₁₆. And set the data "FF₁₆" in addresses 000A₁₆ to 000F₁₆.
The ASCII codes and addresses are listed to the right in hexadecimal notation.
The option data must be entered in address 0010₁₆.

Address	
0000₁₆	'M' = 4D₁₆
0001₁₆	'3' = 33₁₆
0002₁₆	'8' = 38₁₆
0003₁₆	'B' = 42₁₆
0004₁₆	'5' = 35₁₆
0005₁₆	'7' = 37₁₆
0006₁₆	'M' = 4D₁₆
0007₁₆	'C' = 43₁₆

Address	
0008₁₆	'H' = 48₁₆
0009₁₆	'–' = 2D₁₆
000A₁₆	FF₁₆
000B₁₆	FF₁₆
000C₁₆	FF₁₆
000D₁₆	FF₁₆
000E₁₆	FF₁₆
000F₁₆	FF₁₆

GZZ-SH54-20B<88A1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M38B57MCH-[*]XXXFP
MITSUBISHI ELECTRIC

We recommend the use of the following pseudo-command to set the start address of the assembler source program because ASCII codes of the product name are written to addresses 000016 to 000916 of EPROM.

EPROM type	27512
The pseudo-command	*= △$0000 .BYTE △'M38B57MCH–'

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. Mask option specification

High-breakdown voltage ports $P2_0$ to $P2_7$ and $P8_0$ to $P8_3$ can be selected whether pull-down resistors are built-in or not from among the following 8 types by the mask option.

Select built-in type of pull-down resistors from among the following A to G, P, and fill out the following certainly.
(Fill out the upper part of page 1/3 also.)
Set the data of the same option type name in EPROM specified address. (Set the ASCII code of A to G, P; 4116 to 4716, 5016.)

Set the following pseudo-command to the assembler source program.

EPROM type	27512
The pseudo-command	*= △$0010 .BYTE△ $XX

Option type	Connective port of pull-down resistor (connected at "1" writing)											
	$P2_0$	$P2_1$	$P2_2$	$P2_3$	$P2_4$	$P2_5$	$P2_6$	$P2_7$	$P8_0$	$P8_1$	$P8_2$	$P8_3$
A ($41)												
B ($42)							1	1				
C ($43)					1	1	1	1				
D ($44)			1	1	1	1	1	1				
E ($45)	1	1	1	1	1	1	1	1				
F ($46)	1	1	1	1	1	1	1	1	1	1		
G ($47)	1	1	1	1	1	1	1	1	1	1	1	1
P ($50)	1	1	1	1	1	1	1	1				

M38B57MCH-[ ]XXXFP

Option type
Fill out with any one of A to G, P.

(2/3)

GZZ-SH54-20B<88A1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M38B57MCH-☆XXXFP
MITSUBISHI ELECTRIC

✵ 3. Mark specification

Mark specification must be submitted using the correct form for the package being ordered. Fill out the appropriate mark specification form (80P6N) and attach it to the mask ROM confirmation form.

✵ 4. Usage conditions

Please answer the following questions about usage for use in our product inspection :

(1) How will you use the XIN-XOUT oscillator?

☐ Ceramic resonator ☐ Quartz crystal

☐ External clock input ☐ Other ( )

At what frequency? f(XIN) = ☐ MHz

(2) How will you use the XCIN-XCOUT oscillator?

☐ Ceramic resonator ☐ Quartz crystal

☐ External clock input ☐ Other ( )

At what frequency? f(XCIN) = ☐ kHz

✵ 5. Comments

APPENDIX

3.6 Mask ROM confirmation form

GZZ-SH54-21B<88A1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M38B59MFH-☐XXXFP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

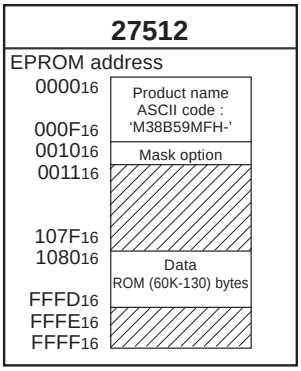
✵ Customer	Company name		TEL ( )	Issuance signature	Submitted by	Supervisor
	Date issued	Date:				

✵ 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.

Checksum code for entire EPROM ☐☐☐☐ (hexadecimal notation)

EPROM type



In the address space of the microcomputer, the internal ROM area is from address 1080₁₆ to FFFD₁₆. The reset vector is stored in addresses FFFC₁₆ and FFFD₁₆.

(1) Set the data in the unused area (the shaded area of the diagram) to "FF₁₆".
(2) The ASCII codes of the product name "M38B59MFH–" must be entered in addresses 0000₁₆ to 0009₁₆. And set the data "FF₁₆" in addresses 000A₁₆ to 000F₁₆.
The ASCII codes and addresses are listed to the right in hexadecimal notation.
The option data must be entered in address 0010₁₆.

Address	
0000₁₆	'M' = 4D₁₆
0001₁₆	'3' = 33₁₆
0002₁₆	'8' = 38₁₆
0003₁₆	'B' = 42₁₆
0004₁₆	'5' = 35₁₆
0005₁₆	'9' = 39₁₆
0006₁₆	'M' = 4D₁₆
0007₁₆	'F' = 46₁₆

Address	
0008₁₆	'H' = 48₁₆
0009₁₆	'–' = 2D₁₆
000A₁₆	FF₁₆
000B₁₆	FF₁₆
000C₁₆	FF₁₆
000D₁₆	FF₁₆
000E₁₆	FF₁₆
000F₁₆	FF₁₆

GZZ-SH54-21B<88A1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M38B59MFH-✻XXXFP
MITSUBISHI ELECTRIC

We recommend the use of the following pseudo-command to set the start address of the assembler source program because ASCII codes of the product name are written to addresses 0000_{16} to 0009_{16} of EPROM.

EPROM type	27512
The pseudo-command	*= △$0000 .BYTE △'M38B59MFH–'

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. Mask option specification

High-breakdown voltage ports $P2_0$ to $P2_7$ and $P8_0$ to $P8_3$ can be selected whether pull-down resistors are built-in or not from among the following 8 types by the mask option.

Select built-in type of pull-down resistors from among the following A to G, P, and fill out the following certainly.
(Fill out the upper part of page 1/3 also.)
Set the data of the same option type name in EPROM specified address. (Set the ASCII code of A to G, P; 41_{16} to 47_{16}, 50_{16}.)

Set the following pseudo-command to the assembler source program.

EPROM type	27512
The pseudo-command	*= △$0010 .BYTE△ $XX

Option type	Connective port of pull-down resistor (connected at "1" writing)											
	$P2_0$	$P2_1$	$P2_2$	$P2_3$	$P2_4$	$P2_5$	$P2_6$	$P2_7$	$P8_0$	$P8_1$	$P8_2$	$P8_3$
A ($41)												
B ($42)							1	1				
C ($43)					1	1	1	1				
D ($44)			1	1	1	1	1	1				
E ($45)	1	1	1	1	1	1	1	1				
F ($46)	1	1	1	1	1	1	1	1	1	1		
G ($47)	1	1	1	1	1	1	1	1	1	1	1	1
P ($50)	1	1	1	1	1	1	1	1				

M38B59MFH- □ XXXFP

Option type
Fill out with any one of A to G, P.

(2/3)

APPENDIX

3.6 Mask ROM confirmation form

GZZ-SH54-21B<88A1>

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M38B59MFH-☆XXXFP
MITSUBISHI ELECTRIC

✵ 3. Mark specification

Mark specification must be submitted using the correct form for the package being ordered. Fill out the appropriate mark specification form (80P6N) and attach it to the mask ROM confirmation form.

✵ 4. Usage conditions

Please answer the following questions about usage for use in our product inspection :

(1) How will you use the XIN-XOUT oscillator?

☐ Ceramic resonator ☐ Quartz crystal

☐ External clock input ☐ Other ( )

At what frequency? f(XIN) = ☐ MHz

(2) How will you use the XCIN-XCOUT oscillator?

☐ Ceramic resonator ☐ Quartz crystal

☐ External clock input ☐ Other ( )

At what frequency? f(XCIN) = ☐ kHz

✵ 5. Comments

3.7 ROM programming confirmation form

GZZ-SH54-22B<88A0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M38B59EF-XXXFP
MITSUBISHI ELECTRIC

Receipt	Date:	
	Section head signature	Supervisor signature

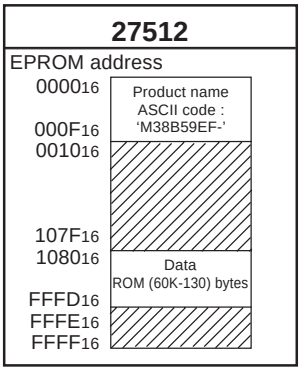
Note : Please fill in all items marked ✵.

✵	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 programming data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type



In the address space of the microcomputer, the internal ROM area is from address 1080_{16} to $FFFD_{16}$. The reset vector is stored in addresses $FFFC_{16}$ and $FFFD_{16}$.

(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 "M38B59EF–" 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}	'8' = 38_{16}
0003_{16}	'B' = 42_{16}
0004_{16}	'5' = 35_{16}
0005_{16}	'9' = 39_{16}
0006_{16}	'E' = 45_{16}
0007_{16}	'F' = 46_{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)

3.7 ROM programming confirmation form

GZZ-SH54-22B<88A0>

ROM number	

740 FAMILY ROM PROGRAMMING CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M38B59EF-XXXFP
MITSUBISHI ELECTRIC

We recommend the use of the following pseudo-command to set the start address of the assembler source program because ASCII codes of the product name are written to addresses 0000_{16} to 0008_{16} of EPROM.

EPROM type	27512
The pseudo-command	*= △$0000 .BYTE △'M38B59EF–'

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 package being ordered. Fill out the appropriate mark specification form (80P6N) and attach it to the ROM programming confirmation form.

✺ 3. Usage conditions

Please answer the following questions about usage for use in our product inspection :

(1) How will you use the XIN-XOUT oscillator?

☐ Ceramic resonator ☐ Quartz crystal

☐ External clock input ☐ Other ( )

At what frequency? f(XIN) = ☐ MHz

(2) How will you use the XCIN-XCOUT oscillator?

☐ Ceramic resonator ☐ Quartz crystal

☐ External clock input ☐ Other ( )

At what frequency? f(XCIN) = ☐ kHz

✺ 4. Comments

(2/2)

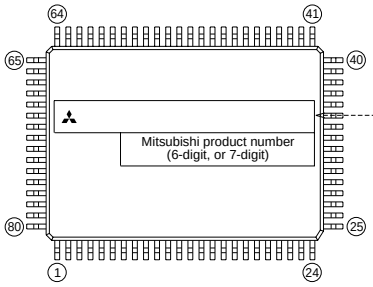
3.8 Mark specification form

80P6N (80-PIN QFP) MARK SPECIFICATION FORM

Mitsubishi IC catalog name

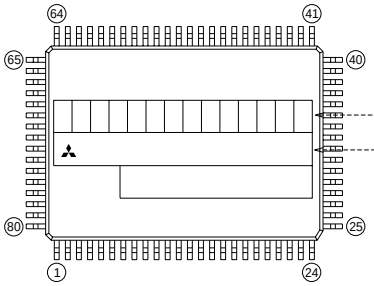
Please choose one of the marking types below (A, B, C), and enter the Mitsubishi IC catalog name and the special mark (if needed).

A. Standard Mitsubishi Mark



Mitsubishi IC catalog name

B. Customer's Parts Number + Mitsubishi IC Catalog Name

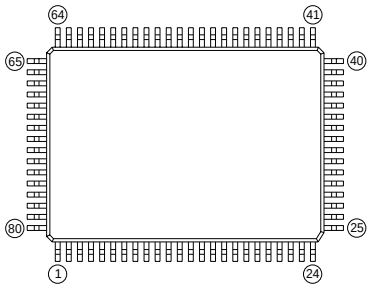


Customer's Parts Number
Note : The fonts and size of characters are standard Mitsubishi type.
Mitsubishi IC catalog name

Notes 1 : 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 14 alphanumeric characters for capital letters, hyphens, commas, periods and so on.
4 : If the Mitsubishi logo is not required, check the box below.

Mitsubishi logo is not required

C. Special Mark Required



Notes1 : If special mark is to be printed, indicate the desired layout of the mark in the left figure. The layout will be duplicated technically as close as possible.
Mitsubishi product number (6-digit, or 7-digit) and Mask ROM number (3-digit) are always marked for sorting the products.
2 : If special character fonts (e,g., customer's trade mark logo) must be used in Special Mark, check the box below.
For the new special character fonts, a clean font original (ideally logo drawing) must be submitted.

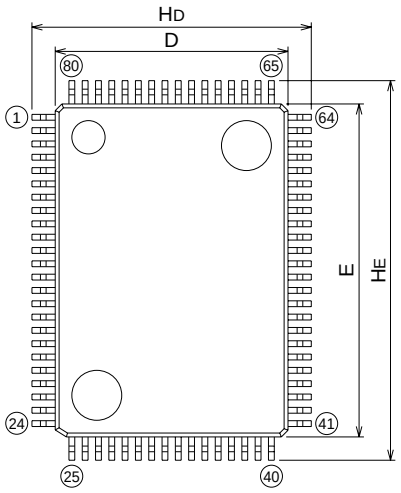
Special character fonts required

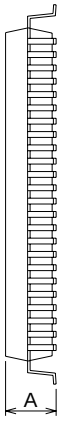
3.9 Package outline

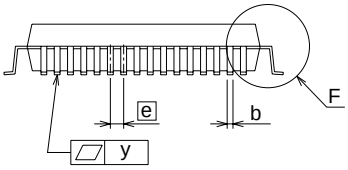
80P6N-A

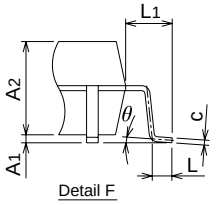
Plastic 80pin 14×20mm body QFP

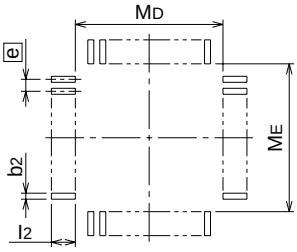
EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
QFP80-P-1420-0.80	–	1.58	Alloy 42











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	13.8	14.0	14.2
E	19.8	20.0	20.2
e	–	0.8	–
H_D	16.5	16.8	17.1
H_E	22.5	22.8	23.1
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	–	–
M_D	–	14.6	–
M_E	–	20.6	–

3.10 List of instruction code

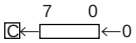
D7 – D4 \ D3 – D0	Hexadecimal notation	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	BRK	ORA IND, X	JSR ZP, IND	BBS 0, A	—	ORA ZP	ASL ZP	BBS 0, ZP	PHP	ORA IMM	ASL A	SEB 0, A	—	ORA ABS	ASL ABS	SEB 0, ZP
0001	1	BPL	ORA IND, Y	CLT	BBC 0, A	—	ORA ZP, X	ASL ZP, X	BBC 0, ZP	CLC	ORA ABS, Y	DEC A	CLB 0, A	—	ORA ABS, X	ASL ABS, X	CLB 0, ZP
0010	2	JSR ABS	AND IND, X	JSR SP	BBS 1, A	BIT ZP	AND ZP	ROL ZP	BBS 1, ZP	PLP	AND IMM	ROL A	SEB 1, A	BIT ABS	AND ABS	ROL ABS	SEB 1, ZP
0011	3	BMI	AND IND, Y	SET	BBC 1, A	—	AND ZP, X	ROL ZP, X	BBC 1, ZP	SEC	AND ABS, Y	INC A	CLB 1, A	LDM ZP	AND ABS, X	ROL ABS, X	CLB 1, ZP
0100	4	RTI	EOR IND, X	STP	BBS 2, A	COM ZP	EOR ZP	LSR ZP	BBS 2, ZP	PHA	EOR IMM	LSR A	SEB 2, A	JMP ABS	EOR ABS	LSR ABS	SEB 2, ZP
0101	5	BVC	EOR IND, Y	—	BBC 2, A	—	EOR ZP, X	LSR ZP, X	BBC 2, ZP	CLI	EOR ABS, Y	—	CLB 2, A	—	EOR ABS, X	LSR ABS, X	CLB 2, ZP
0110	6	RTS	ADC IND, X	MUL ZP, X	BBS 3, A	TST ZP	ADC ZP	ROR ZP	BBS 3, ZP	PLA	ADC IMM	ROR A	SEB 3, A	JMP IND	ADC ABS	ROR ABS	SEB 3, ZP
0111	7	BVS	ADC IND, Y	—	BBC 3, A	—	ADC ZP, X	ROR ZP, X	BBC 3, ZP	SEI	ADC ABS, Y	—	CLB 3, A	—	ADC ABS, X	ROR ABS, X	CLB 3, ZP
1000	8	BRA	STA IND, X	RRF ZP	BBS 4, A	STY ZP	STA ZP	STX ZP	BBS 4, ZP	DEY	—	TXA	SEB 4, A	STY ABS	STA ABS	STX ABS	SEB 4, ZP
1001	9	BCC	STA IND, Y	—	BBC 4, A	STY ZP, X	STA ZP, X	STX ZP, Y	BBC 4, ZP	TYA	STA ABS, Y	TXS	CLB 4, A	—	STA ABS, X	—	CLB 4, ZP
1010	A	LDY IMM	LDA IND, X	LDX IMM	BBS 5, A	LDY ZP	LDA ZP	LDX ZP	BBS 5, ZP	TAY	LDA IMM	TAX	SEB 5, A	LDY ABS	LDA ABS	LDX ABS	SEB 5, ZP
1011	B	BCS	LDA IND, Y	JMP ZP, IND	BBC 5, A	LDY ZP, X	LDA ZP, X	LDX ZP, Y	BBC 5, ZP	CLV	LDA ABS, Y	TSX	CLB 5, A	LDY ABS, X	LDA ABS, X	LDX ABS, Y	CLB 5, ZP
1100	C	CPY IMM	CMP IND, X	WIT	BBS 6, A	CPY ZP	CMP ZP	DEC ZP	BBS 6, ZP	INY	CMP IMM	DEX	SEB 6, A	CPY ABS	CMP ABS	DEC ABS	SEB 6, ZP
1101	D	BNE	CMP IND, Y	—	BBC 6, A	—	CMP ZP, X	DEC ZP, X	BBC 6, ZP	CLD	CMP ABS, Y	—	CLB 6, A	—	CMP ABS, X	DEC ABS, X	CLB 6, ZP
1110	E	CPX IMM	SBC IND, X	DIV ZP, X	BBS 7, A	CPX ZP	SBC ZP	INC ZP	BBS 7, ZP	INX	SBC IMM	NOP	SEB 7, A	CPX ABS	SBC ABS	INC ABS	SEB 7, ZP
1111	F	BEQ	SBC IND, Y	—	BBC 7, A	—	SBC ZP, X	INC ZP, X	BBC 7, ZP	SED	SBC ABS, Y	—	CLB 7, A	—	SBC ABS, X	INC ABS, X	CLB 7, ZP

: 3-byte instruction

: 2-byte instruction

: 1-byte instruction

3.11 Machine instructions

Symbol	Function	Details	Addressing mode																
			IMP			IMM			A			BIT, A, R			ZP			BIT, ZP, R		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
ADC (Note 1) (Note 5)	When T = 0 A ← A + M + C When T = 1 M(X) ← M(X) + M + C	When T = 0, this instruction adds the contents M, C, and A; and stores the results in A and C. When T = 1, this instruction adds the contents of M(X), M and C; and stores the results in M(X) and C. When T=1, the contents of A remain unchanged, but the contents of status flags are changed. M(X) represents the contents of memory where is indicated by X.				69	2	2							65	3	2			
AND (Note 1)	When T = 0 A ← A ∧ M When T = 1 M(X) ← M(X) ∧ M	When T = 0, this instruction transfers the contents of A and M to the ALU which performs a bit-wise AND operation and stores the result back in A. When T = 1, this instruction transfers the contents M(X) and M to the ALU which performs a bit-wise AND operation and stores the results back in M(X). When T = 1, the contents of A remain unchanged, but status flags are changed. M(X) represents the contents of memory where is indicated by X.				29	2	2							25	3	2			
ASL		This instruction shifts the content of A or M by one bit to the left, with bit 0 always being set to 0 and bit 7 of A or M always being contained in C.							0A	2	1				06	5	2			
BBC (Note 4)	Ai or Mi = 0?	This instruction tests the designated bit i of M or A and takes a branch if the bit is 0. The branch address is specified by a relative address. If the bit is 1, next instruction is executed.										13 + 20i	4	2				17 + 20i	5	3
BBS (Note 4)	Ai or Mi = 1?	This instruction tests the designated bit i of the M or A and takes a branch if the bit is 1. The branch address is specified by a relative address. If the bit is 0, next instruction is executed.										03 + 20i	4	2				07 + 20i	5	3
BCC (Note 4)	C = 0?	This instruction takes a branch to the appointed address if C is 0. The branch address is specified by a relative address. If C is 1, the next instruction is executed.																		
BCS (Note 4)	C = 1?	This instruction takes a branch to the appointed address if C is 1. The branch address is specified by a relative address. If C is 0, the next instruction is executed.																		
BEQ (Note 4)	Z = 1?	This instruction takes a branch to the appointed address when Z is 1. The branch address is specified by a relative address. If Z is 0, the next instruction is executed.																		
BIT	A ∧ M	This instruction takes a bit-wise logical AND of A and M contents; however, the contents of A and M are not modified. The contents of N, V, Z are changed, but the contents of A, M remain unchanged.													24	3	2			
BMI (Note 4)	N = 1?	This instruction takes a branch to the appointed address when N is 1. The branch address is specified by a relative address. If N is 0, the next instruction is executed.																		
BNE (Note 4)	Z = 0?	This instruction takes a branch to the appointed address if Z is 0. The branch address is specified by a relative address. If Z is 1, the next instruction is executed.																		

Addressing mode																												Processor status register							
ZP, X			ZP, Y			ABS			ABS, X			ABS, Y			IND			ZP, IND			IND, X			IND, Y			REL			SP			7	6	5	4	3	2	1	0
OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	N	V	T	B	D	I	Z	C
75	4	2				6D	4	3	7D	5	3	79	5	3							61	6	2	71	6	2							N	V	•	•	•	•	Z	C
35	4	2				2D	4	3	3D	5	3	39	5	3							21	6	2	31	6	2							N	•	•	•	•	•	Z	•
16	6	2				0E	6	3	1E	7	3																						N	•	•	•	•	•	Z	C
																																	•	•	•	•	•	•	•	•
																																	•	•	•	•	•	•	•	•
																											90	2	2				•	•	•	•	•	•	•	•
																											B0	2	2				•	•	•	•	•	•	•	•
																											F0	2	2				•	•	•	•	•	•	•	•
						2C	4	3																									M7	M6	•	•	•	•	Z	•
																											30	2	2				•	•	•	•	•	•	•	•
																											D0	2	2				•	•	•	•	•	•	•	•

APPENDIX

3.11 Machine instructions

Symbol	Function	Details	Addressing mode																
			IMP			IMM			A			BIT, A			ZP			BIT, ZP		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
BPL (Note 4)	N = 0?	This instruction takes a branch to the appointed address if N is 0. The branch address is specified by a relative address. If N is 1, the next instruction is executed.																		
BRA	PC ← PC ± offset	This instruction branches to the appointed address. The branch address is specified by a relative address.																		
BRK	B ← 1 (PC) ← (PC) + 2 M(S) ← PC_H S ← S – 1 M(S) ← PC_L S ← S – 1 M(S) ← PS S ← S – 1 I ← 1 PC_L ← AD_L PC_H ← AD_H	When the BRK instruction is executed, the CPU pushes the current PC contents onto the stack. The BADRS designated in the interrupt vector table is stored into the PC.	00	7	1															
BVC (Note 4)	V = 0?	This instruction takes a branch to the appointed address if V is 0. The branch address is specified by a relative address. If V is 1, the next instruction is executed.																		
BVS (Note 4)	V = 1?	This instruction takes a branch to the appointed address when V is 1. The branch address is specified by a relative address. When V is 0, the next instruction is executed.																		
CLB	Ai or Mi ← 0	This instruction clears the designated bit i of A or M.										1B + 20i	2	1				1F + 20i	5	2
CLC	C ← 0	This instruction clears C.	18	2	1															
CLD	D ← 0	This instruction clears D.	D8	2	1															
CLI	I ← 0	This instruction clears I.	58	2	1															
CLT	T ← 0	This instruction clears T.	12	2	1															
CLV	V ← 0	This instruction clears V.	B8	2	1															
CMP (Note 3)	When T = 0 A – M When T = 1 M(X) – M	When T = 0, this instruction subtracts the contents of M from the contents of A. The result is not stored and the contents of A or M are not modified. When T = 1, the CMP subtracts the contents of M from the contents of M(X). The result is not stored and the contents of X, M, and A are not modified. M(X) represents the contents of memory where is indicated by X.				C9	2	2							C5	3	2			
COM	M ← $\overline{M}$	This instruction takes the one's complement of the contents of M and stores the result in M.													44	5	2			
CPX	X – M	This instruction subtracts the contents of M from the contents of X. The result is not stored and the contents of X and M are not modified.				E0	2	2							E4	3	2			
CPY	Y – M	This instruction subtracts the contents of M from the contents of Y. The result is not stored and the contents of Y and M are not modified.				C0	2	2							C4	3	2			
DEC	A ← A – 1 or M ← M – 1	This instruction subtracts 1 from the contents of A or M.							1A	2	1				C6	5	2			

Addressing mode																												Processor status register							
ZP, X			ZP, Y			ABS			ABS, X			ABS, Y			IND			ZP, IND			IND, X			IND, Y			REL			SP			7	6	5	4	3	2	1	0
OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	N	V	T	B	D	I	Z	C
																											10	2	2				•	•	•	•	•	•	•	•
																											80	4	2				•	•	•	•	•	•	•	•
																																		•	•	•	1	•	1	•	•
																											50	2	2				•	•	•	•	•	•	•	•
																											70	2	2				•	•	•	•	•	•	•	•
																																		•	•	•	•	•	•	•	•
																																		•	•	•	•	•	•	•	0
																																		•	•	•	•	0	•	•	•
																																		•	•	•	•	•	0	•	•
																																		•	•	0	•	•	•	•	•
																																		•	0	•	•	•	•	•	•
D5	4	2				CD	4	3	DD	5	3	D9	5	3							C1	6	2	D1	6	2							N	•	•	•	•	•	Z	C
																																		N	•	•	•	•	•	Z	•
						EC	4	3																										N	•	•	•	•	•	Z	C
						CC	4	3																										N	•	•	•	•	•	Z	C
D6	6	2				CE	6	3	DE	7	3																							N	•	•	•	•	•	Z	•

APPENDIX

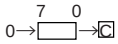
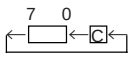
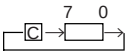
3.11 Machine instructions

Symbol	Function	Details	Addressing mode																
			IMP			IMM			A			BIT, A			ZP			BIT, ZP		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
DEX	X ← X – 1	This instruction subtracts one from the current contents of X.	CA	2	1															
DEY	Y ← Y – 1	This instruction subtracts one from the current contents of Y.	88	2	1															
DIV	A ← (M(zz + X + 1), M(zz + X)) / A M(S) ← one's complement of Remainder S ← S – 1	This instruction divides the 16-bit data in M(zz+(X)) (low-order byte) and M(zz+(X)+1) (high-order byte) by the contents of A. The quotient is stored in A and the one's complement of the remainder is pushed onto the stack.																		
EOR (Note 1)	When T = 0 A ← A ∀ M When T = 1 M(X) ← M(X) ∀ M	When T = 0, this instruction transfers the contents of the M and A to the ALU which performs a bit-wise Exclusive OR, and stores the result in A. When T = 1, the contents of M(X) and M are transferred to the ALU, which performs a bit-wise Exclusive OR and stores the results in M(X). The contents of A remain unchanged, but status flags are changed. M(X) represents the contents of memory where is indicated by X.				49	2	2							45	3	2			
INC	A ← A + 1 or M ← M + 1	This instruction adds one to the contents of A or M.							3A	2	1				E6	5	2			
INX	X ← X + 1	This instruction adds one to the contents of X.	E8	2	1															
INY	Y ← Y + 1	This instruction adds one to the contents of Y.	C8	2	1															
JMP	If addressing mode is ABS PC_L ← AD_L PC_H ← AD_H If addressing mode is IND PC_L ← M (AD_H, AD_L) PC_H ← M (AD_H, AD_L + 1) If addressing mode is ZP, IND PC_L ← M(00, AD_L) PC_H ← M(00, AD_L + 1)	This instruction jumps to the address designated by the following three addressing modes: Absolute Indirect Absolute Zero Page Indirect Absolute																		
JSR	M(S) ← PC_H S ← S – 1 M(S) ← PC_L S ← S – 1 After executing the above, if addressing mode is ABS, PC_L ← AD_L PC_H ← AD_H if addressing mode is SP, PC_L ← AD_L PC_H ← FF If addressing mode is ZP, IND, PC_L ← M(00, AD_L) PC_H ← M(00, AD_L + 1)	This instruction stores the contents of the PC in the stack, then jumps to the address designated by the following addressing modes: Absolute Special Page Zero Page Indirect Absolute																		
LDA (Note 2)	When T = 0 A ← M When T = 1 M(X) ← M	When T = 0, this instruction transfers the contents of M to A. When T = 1, this instruction transfers the contents of M to (M(X)). The contents of A remain unchanged, but status flags are changed. M(X) represents the contents of memory where is indicated by X.				A9	2	2							A5	3	2			
LDM	M ← nn	This instruction loads the immediate value in M.													3C	4	3			
LDX	X ← M	This instruction loads the contents of M in X.				A2	2	2							A6	3	2			
LDY	Y ← M	This instruction loads the contents of M in Y.				A0	2	2							A4	3	2			

Addressing mode																													Processor status register							
ZP, X			ZP, Y			ABS			ABS, X			ABS, Y			IND			ZP, IND			IND, X			IND, Y			REL			SP			7	6	5	4	3	2	1	0
OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	N	V	T	B	D	I	Z	C
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
E2	16	2																															•	•	•	•	•	•	•	•
55	4	2				4D	4	3	5D	5	3	59	5	3							41	6	2	51	6	2							N	•	•	•	•	•	Z	•
F6	6	2				EE	6	3	FE	7	3																						N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
						4C	3	3							6C	5	3	B2	4	2													•	•	•	•	•	•	•	•
						20	6	3										02	7	2										22	5	2	•	•	•	•	•	•	•	•
B5	4	2				AD	4	3	BD	5	3	B9	5	3							A1	6	2	B1	6	2							N	•	•	•	•	•	Z	•
																																	•	•	•	•	•	•	•	•
			B6	4	2	AE	4	3				BE	5	3																			N	•	•	•	•	•	Z	•
B4	4	2				AC	4	3	BC	5	3																						N	•	•	•	•	•	Z	•

APPENDIX

3.11 Machine instructions

Symbol	Function	Details	Addressing mode																
			IMP			IMM			A			BIT, A			ZP			BIT, ZP		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
LSR		This instruction shifts either A or M one bit to the right such that bit 7 of the result always is set to 0, and the bit 0 is stored in C.							4A	2	1				46	5	2			
MUL	M(S) • A ← A ✻ M(zz + X) S ← S – 1	This instruction multiply Accumulator with the memory specified by the Zero Page X address mode and stores the high-order byte of the result on the Stack and the low-order byte in A.																		
NOP	PC ← PC + 1	This instruction adds one to the PC but does no otheroperation.	EA	2	1															
ORA (Note 1)	When T = 0 A ← A V M When T = 1 M(X) ← M(X) V M	When T = 0, this instruction transfers the contents of A and M to the ALU which performs a bit-wise “OR”, and stores the result in A. When T = 1, this instruction transfers the contents of M(X) and the M to the ALU which performs a bit-wise OR, and stores the result in M(X). The contents of A remain unchanged, but status flags are changed. M(X) represents the contents of memory where is indicated by X.				09	2	2							05	3	2			
PHA	M(S) ← A S ← S – 1	This instruction pushes the contents of A to the memory location designated by S, and decrements the contents of S by one.	48	3	1															
PHP	M(S) ← PS S ← S – 1	This instruction pushes the contents of PS to the memory location designated by S and decrements the contents of S by one.	08	3	1															
PLA	S ← S + 1 A ← M(S)	This instruction increments S by one and stores the contents of the memory designated by S in A.	68	4	1															
PLP	S ← S + 1 PS ← M(S)	This instruction increments S by one and stores the contents of the memory location designated by S in PS.	28	4	1															
ROL		This instruction shifts either A or M one bit left through C. C is stored in bit 0 and bit 7 is stored in C.							2A	2	1				26	5	2			
ROR		This instruction shifts either A or M one bit right through C. C is stored in bit 7 and bit 0 is stored in C.							6A	2	1				66	5	2			
RRF		This instruction rotates 4 bits of the M content to the right.													82	8	2			
RTI	S ← S + 1 PS ← M(S) S ← S + 1 PCL ← M(S) S ← S + 1 PCH ← M(S)	This instruction increments S by one, and stores the contents of the memory location designated by S in PS. S is again incremented by one and stores the contents of the memory location designated by S in PCL. S is again incremented by one and stores the contents of memory location designated by S in PCH.	40	6	1															
RTS	S ← S + 1 PCL ← M(S) S ← S + 1 PCH ← M(S) (PC) ← (PC) + 1	This instruction increments S by one and stores the contents of the memory location designated by S in PCL. S is again incremented by one and the contents of the memory location is stored in PCH. PC is incremented by 1.	60	6	1															

Addressing mode																												Processor status register							
ZP, X			ZP, Y			ABS			ABS, X			ABS, Y			IND			ZP, IND			IND, X			IND, Y			REL			SP			7	6	5	4	3	2	1	0
OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	N	V	T	B	D	I	Z	C
56	6	2				4E	6	3	5E	7	3																						0	•	•	•	•	•	Z	C
62	15	2																															•	•	•	•	•	•	•	•
																																	•	•	•	•	•	•	•	•
15	4	2				0D	4	3	1D	5	3	19	5	3							01	6	2	11	6	2							N	•	•	•	•	•	Z	•
																																	•	•	•	•	•	•	•	•
																																	•	•	•	•	•	•	•	•
																																	N	•	•	•	•	•	Z	•
																																	(Value saved in stack)							
36	6	2				2E	6	3	3E	7	3																						N	•	•	•	•	•	Z	C
76	6	2				6E	6	3	7E	7	3																						N	•	•	•	•	•	Z	C
																																	•	•	•	•	•	•	•	•
																																	(Value saved in stack)							
																																	•	•	•	•	•	•	•	•

APPENDIX

3.11 Machine instructions

Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT, A			ZP			BIT, ZP		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
SBC (Note 1) (Note 5)	When T = 0 A ← A – M – $\overline{C}$ When T = 1 M(X) ← M(X) – M – $\overline{C}$	When T = 0, this instruction subtracts the value of M and the complement of C from A, and stores the results in A and C. When T = 1, the instruction subtracts the contents of M and the complement of C from the contents of M(X), and stores the results in M(X) and C. A remain unchanged, but status flag are changed. M(X) represents the contents of memory where is indicated by X.				E9	2	2							E5	3	2			
SEB	Ai or Mi ← 1	This instruction sets the designated bit i of A or M.										0B + 20i	2	1				0F + 20i	5	2
SEC	C ← 1	This instruction sets C.	38	2	1															
SED	D ← 1	This instruction set D.	F8	2	1															
SEI	I ← 1	This instruction set I.	78	2	1															
SET	T ← 1	This instruction set T.	32	2	1															
STA	M ← A	This instruction stores the contents of A in M. The contents of A does not change.													85	4	2			
STP		This instruction resets the oscillation control F/F and the oscillation stops. Reset or interrupt input is needed to wake up from this mode.	42	2	1															
STX	M ← X	This instruction stores the contents of X in M. The contents of X does not change.													86	4	2			
STY	M ← Y	This instruction stores the contents of Y in M. The contents of Y does not change.													84	4	2			
TAX	X ← A	This instruction stores the contents of A in X. The contents of A does not change.	AA	2	1															
TAY	Y ← A	This instruction stores the contents of A in Y. The contents of A does not change.	A8	2	1															
TST	M = 0?	This instruction tests whether the contents of M are “0” or not and modifies the N and Z.													64	3	2			
TSX	X ← S	This instruction transfers the contents of S in X.	BA	2	1															
TXA	A ← X	This instruction stores the contents of X in A.	8A	2	1															
TXS	S ← X	This instruction stores the contents of X in S.	9A	2	1															
TYA	A ← Y	This instruction stores the contents of Y in A.	98	2	1															
WIT		The WIT instruction stops the internal clock but not the oscillation of the oscillation circuit is not stopped. CPU starts its function after the Timer X over flows (comes to the terminal count). All registers or internal memory contents except Timer X will not change during this mode. (Of course needs VDD).	C2	2	1															

Notes 1 : The number of cycles “n” is increased by 3 when T is 1.
2 : The number of cycles “n” is increased by 2 when T is 1.
3 : The number of cycles “n” is increased by 1 when T is 1.
4 : The number of cycles “n” is increased by 2 when branching has occurred.
5 : N, V, and Z flags are invalid in decimal operation mode.

Addressing mode																												Processor status register							
ZP, X			ZP, Y			ABS			ABS, X			ABS, Y			IND			ZP, IND			IND, X			IND, Y			REL			SP			7	6	5	4	3	2	1	0
OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	N	V	T	B	D	I	Z	C
F5	4	2				ED	4	3	FD	5	3	F9	5	3							E1	6	2	F1	6	2							N	V	•	•	•	•	Z	C
																																	•	•	•	•	•	•	•	•
																																	•	•	•	•	•	•	•	1
																																	•	•	•	•	1	•	•	•
																																	•	•	•	•	•	1	•	•
																																	•	•	1	•	•	•	•	•
95	5	2				8D	5	3	9D	6	3	99	6	3							81	7	2	91	7	2							•	•	•	•	•	•	•	•
																																	•	•	•	•	•	•	•	•
			96	5	2	8E	5	3																									•	•	•	•	•	•	•	•
94	5	2				8C	5	3																									•	•	•	•	•	•	•	•
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
																																	N	•	•	•	•	•	Z	•
																																	•	•	•	•	•	•	•	•
																																	N	•	•	•	•	•	Z	•
																																	•	•	•	•	•	•	•	•

APPENDIX

3.11 Machine instructions

Symbol	Contents	Symbol	Contents
IMP	Implied addressing mode	+	Addition
IMM	Immediate addressing mode	–	Subtraction
A	Accumulator or Accumulator addressing mode	✻	Multiplication
BIT, A	Accumulator bit addressing mode	/	Division
BIT, A, R	Accumulator bit relative addressing mode	Λ	Logical OR
ZP	Zero page addressing mode	V	Logical AND
BIT, ZP	Zero page bit addressing mode	∀	Logical exclusive OR
BIT, ZP, R	Zero page bit relative addressing mode	—	Negation
ZP, X	Zero page X addressing mode	←	Shows direction of data flow
ZP, Y	Zero page Y addressing mode	X	Index register X
ABS	Absolute addressing mode	Y	Index register Y
ABS, X	Absolute X addressing mode	S	Stack pointer
ABS, Y	Absolute Y addressing mode	PC	Program counter
IND	Indirect absolute addressing mode	PS	Processor status register
		PC_H	8 high-order bits of program counter
ZP, IND	Zero page indirect absolute addressing mode	PC_L	8 low-order bits of program counter
		AD_H	8 high-order bits of address
IND, X	Indirect X addressing mode	AD_L	8 low-order bits of address
IND, Y	Indirect Y addressing mode	FF	FF in Hexadecimal notation
REL	Relative addressing mode	nn	Immediate value
SP	Special page addressing mode	zz	Zero page address
C	Carry flag	M	Memory specified by address designation of any addressing mode
Z	Zero flag		
I	Interrupt disable flag	M(X)	Memory of address indicated by contents of index register X
D	Decimal mode flag		
B	Break flag	M(S)	Memory of address indicated by contents of stack pointer
T	X-modified arithmetic mode flag		
V	Overflow flag	$M(AD_H, AD_L)$	Contents of memory at address indicated by AD_H and AD_L, in AD_H is 8 high-order bits and AD_L is 8 low-order bits.
N	Negative flag		
		$M(00, AD_L)$	Contents of address indicated by zero page AD_L
		Ai	Bit i (i = 0 to 7) of accumulator
		Mi	Bit i (i = 0 to 7) of memory
		OP	Opcode
		n	Number of cycles
		#	Number of bytes

3.12 M35501FP

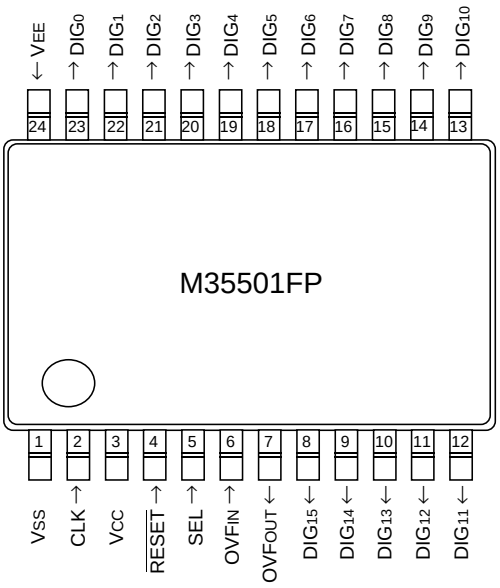
DESCRIPTION

The M35501FP generates digit signals for fluorescent display when connected to the output port of a microcomputer. There are up to 16 digit pins available, and more can be added by connecting additional M35501FPs. The number of fluorescent displays can be increased easily by connecting the M35501FP to the CMOS FLD output pins of an 8-bit microcomputer in MITSUBISHI's 38B5 Group. The M35501FP is suitable for fluorescent display control on household electric appliances, audio products, etc.

FEATURES

- Digit output ..... 16 (maximum)
  - •Up to 16 pins can be selected
  - •More digits available by connecting additional M35501FPs
  - •Output structure: high-breakdown voltage, P-channel open-drain; built-in pull-down resistor between digit output pins and VEE pin
- Power-on reset circuit ..... Built-in
- Power source voltage ..... 4.0 to 5.5 V
- Pull-down power source voltage ..... Vcc – 43 V
- Operating temperature range ..... –20 to 85 °C
- Package 24P2E
- Power dissipation ..... 250 µW (at 100 kHz operation clock)

PIN CONFIGURATION (TOP VIEW)



Outline: 24P2E-A
24-pin plastic-molded SSOP

Fig. 3.12.1 Pin configuration of M35501FP

APPENDIX

3.12 M35501FP

FUNCTIONAL BLOCK

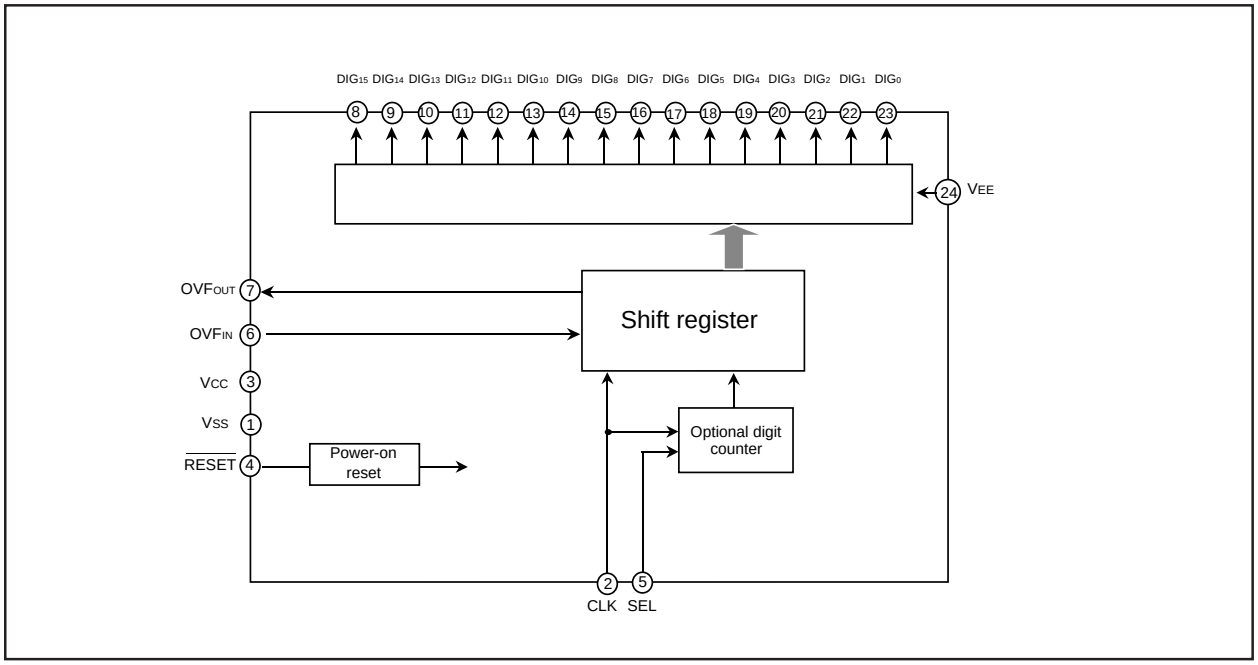


Fig. 3.12.2 Functional block diagram

PIN DESCRIPTION

Table 3.12.1 Pin description

Pin	Name	Function	Output Structure	Fig. No.
VCC, VSS	Power source input	Apply 4.0–5.5 V to Vcc, and 0V to Vss.	–	–
$\overline{\text{RESET}}$	Reset input	Reset internal shift register (built-in power-on reset circuit).	CMOS input level Built-in pull-up resistor	3
CLK	Clock input	Digit output varies according to rising edge of clock input.	CMOS input level Built-in pull-down resistor	2
SEL	Select input	Use when specifying the number of digits.	CMOS input level Built-in pull-down resistor	2
OVFIN	Overflow signal input	Input "H" when using one M35501FP. Connect to OVFOUT pin of additional M35501FPs when using multiple M35501FPs (to use 17 digits or more).	CMOS input level	4
OVFOUT	Overflow signal output	Leave open when using one M35501FP. Connect to OVFIN pin of additional M35501FPs when using multiple M35501FPs (to use 17 digits or more).	CMOS output	5
DIG15–DIG0	Digit output	Output the digit output waveform of fluorescent display. Leave open when not in use (VEE level output).	High-breakdown-voltage P-channel open-drain output Built-in pull-down resistor	1
VEE	Pull-down power source input	Apply voltage to DIG0–DIG15 pull-down resistors.	–	–

PORT BLOCK

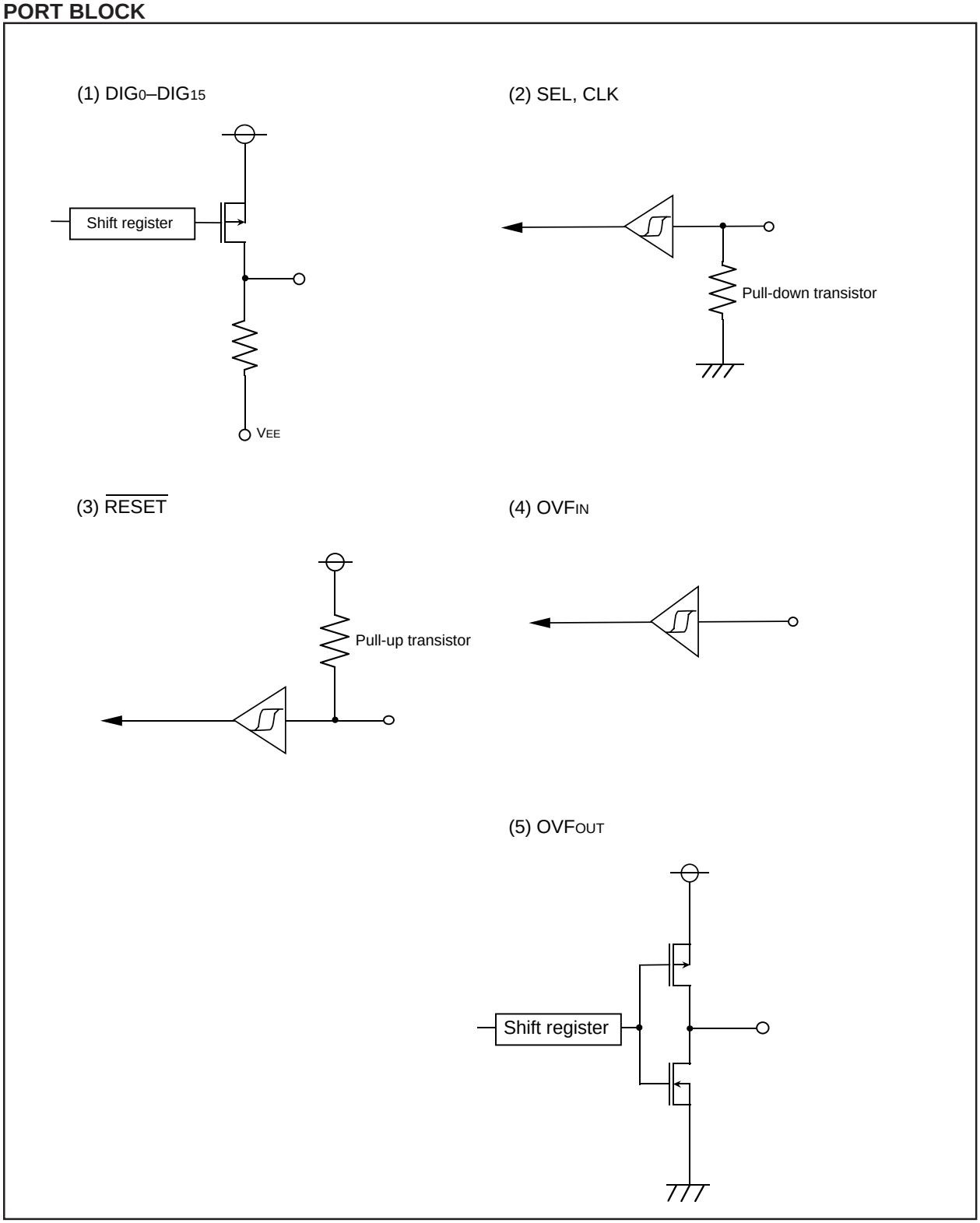


Fig. 3.12.3 Port block diagram

3.12 M35501FP

USAGE

Three usages of the M35501FP are described below.

(1) 16-Digit Mode: 16 digits selected

The number of digits is set to 16 by fixing the OVFIN pin to "H" and the SEL pin to "L." Figure 3.12.5 shows the output waveform.

(2) Optional Digit Mode: 1-16 digits selectable

When the number of CLK pin rising edges during an "H" period of the SEL pin is n and the OVFIN pin is fixed to "H," the number of digits set is n. If n is 16 or more, all 16 digits are set. Figure 3.12.6 shows the output waveform.

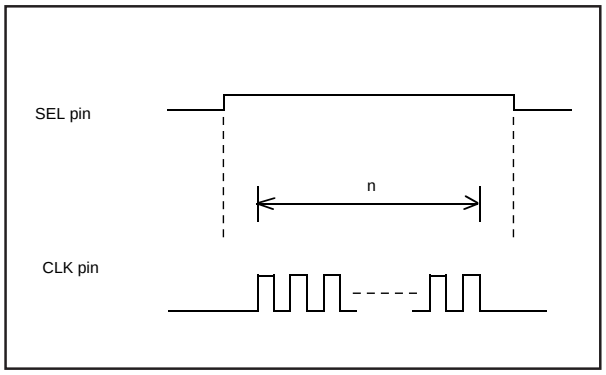


Fig. 3.12.4 Digit setting

(3) Cascade Mode: 17 digits or more selectable

17 digits or more can be used by connecting two M35501FPs or more. Figure 3.12.7 shows an example using three M35501FPs, offering 33 to 48 digit outputs.

Cascade mode will not operate if all M35501FPs are in 16-digit mode (SEL = "L"). Use the most significant M35501FP in the optional digit mode for DIG output. Figure 3.12.8 shows the output waveform.

DIGIT OUTPUT WAVEFORM

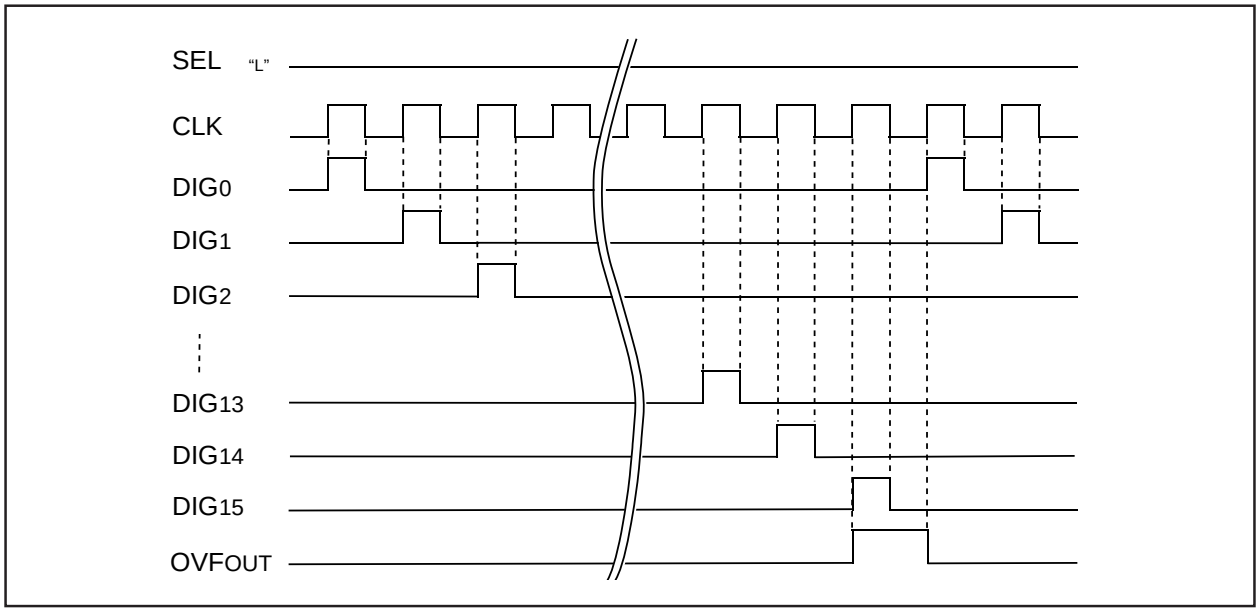


Fig. 3.12.5 16-digit mode output waveform

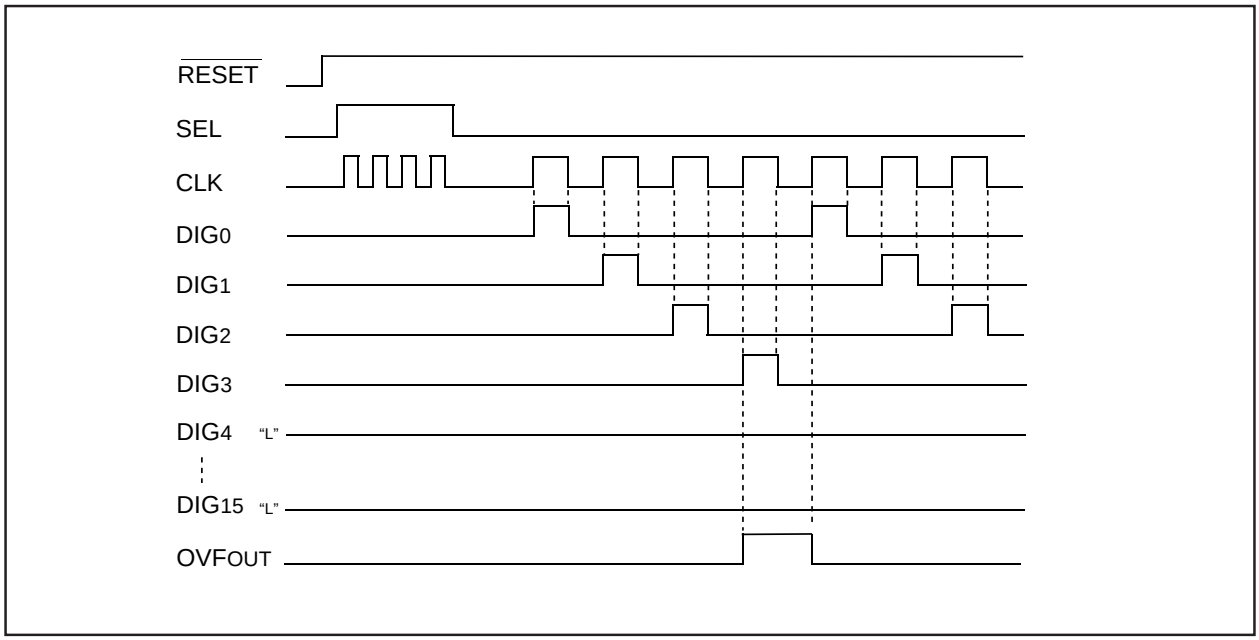


Fig. 3.12.6 Optional digit mode output waveform

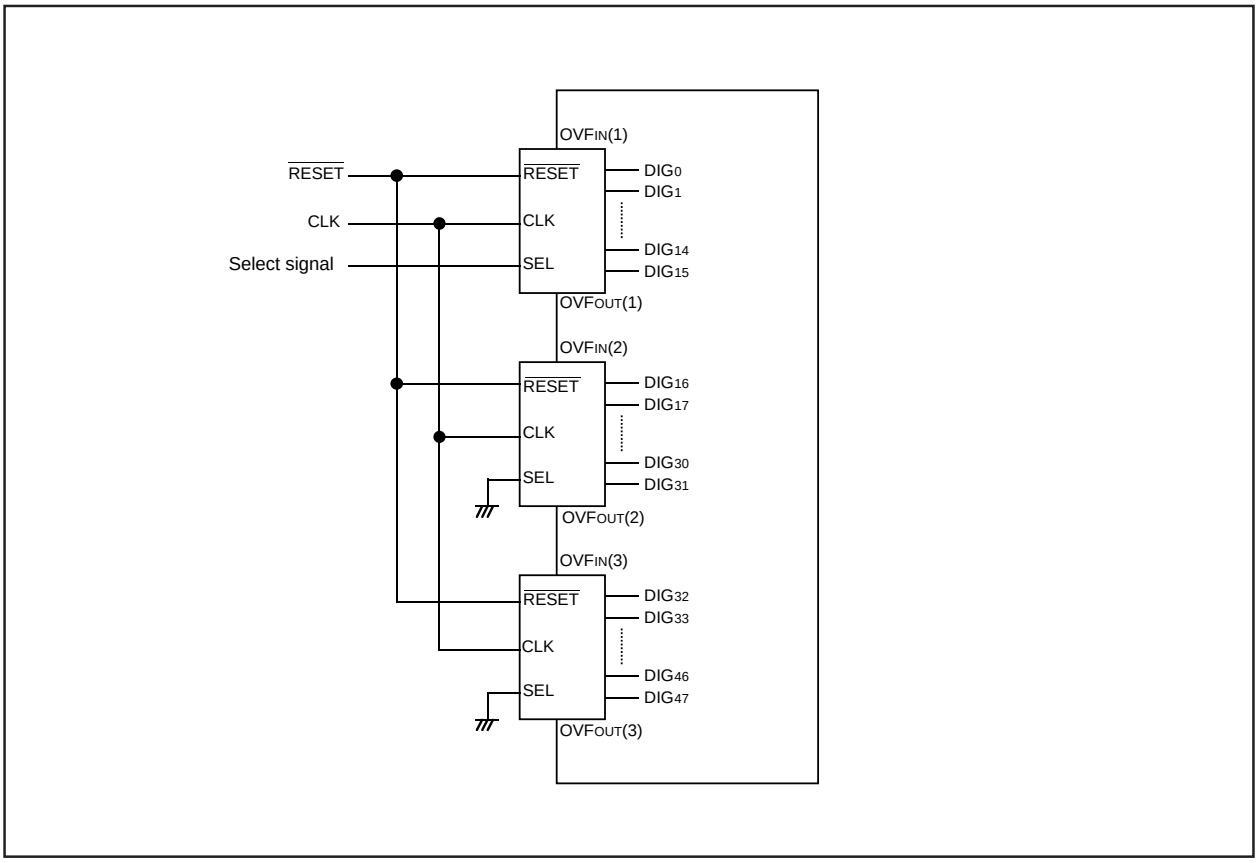


Fig. 3.12.7 Cascade mode connection example: 17 digits or more selected

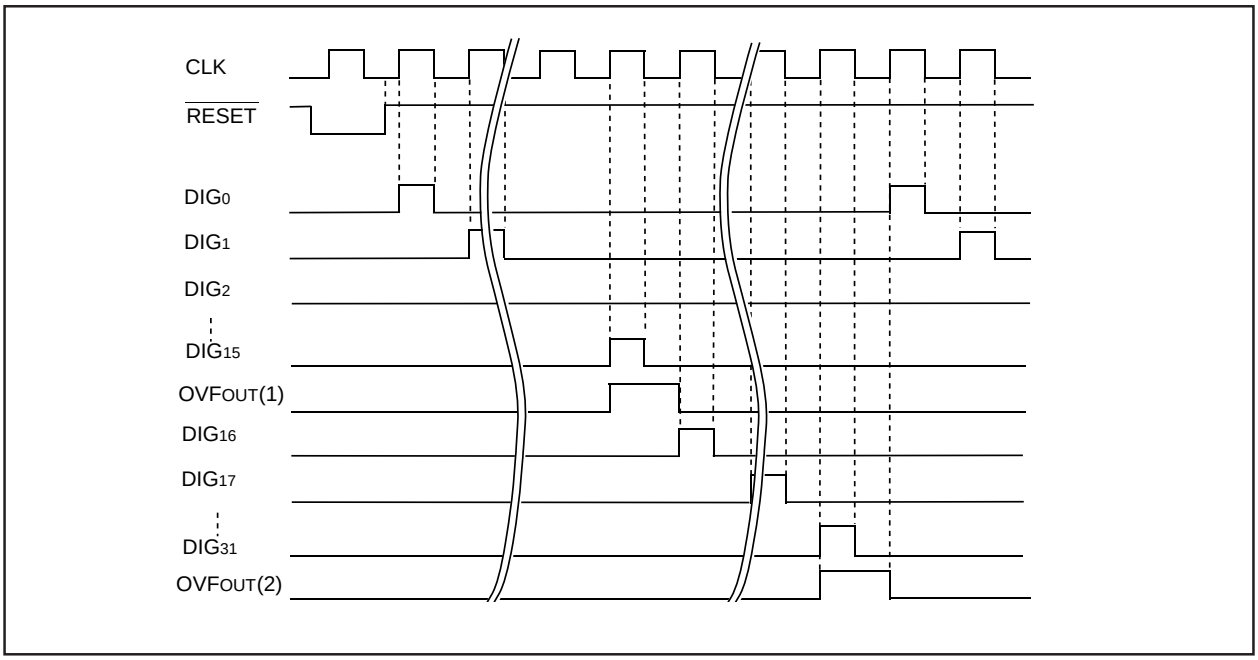


Fig. 3.12.8 Cascade mode output waveform

The number of fluorescent displays can be increased by connecting the M35501FP to the CMOS FLD output pins on a 38B5 Group microcomputer.

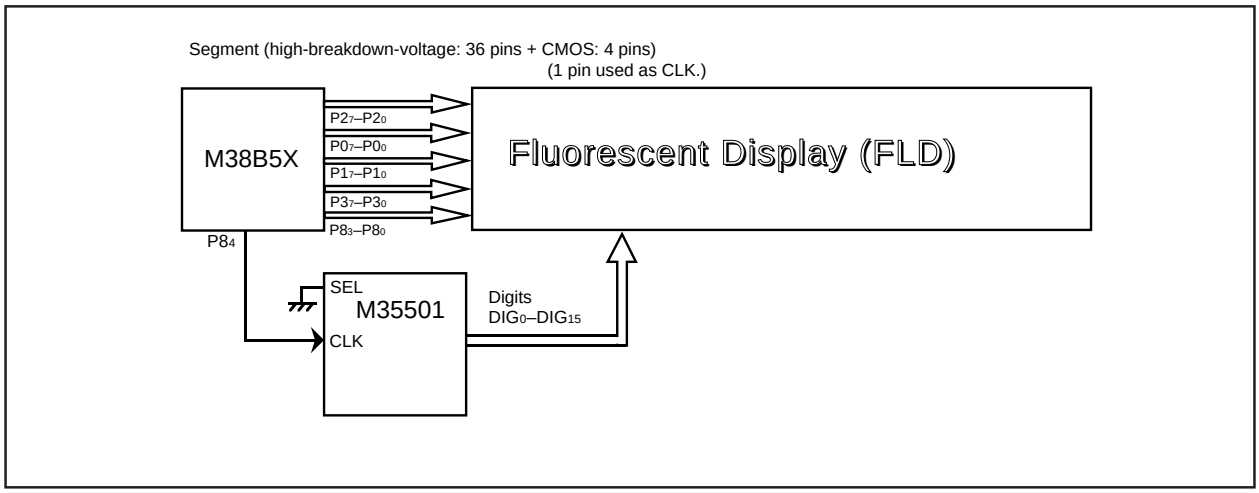


Fig. 3.12.9 Connection example with 38B5 Group microcomputer (1 to 16 digits)

This FLD controller can control up to 32 digits using the 32 timing mode of the 38B5 Group microcomputer.

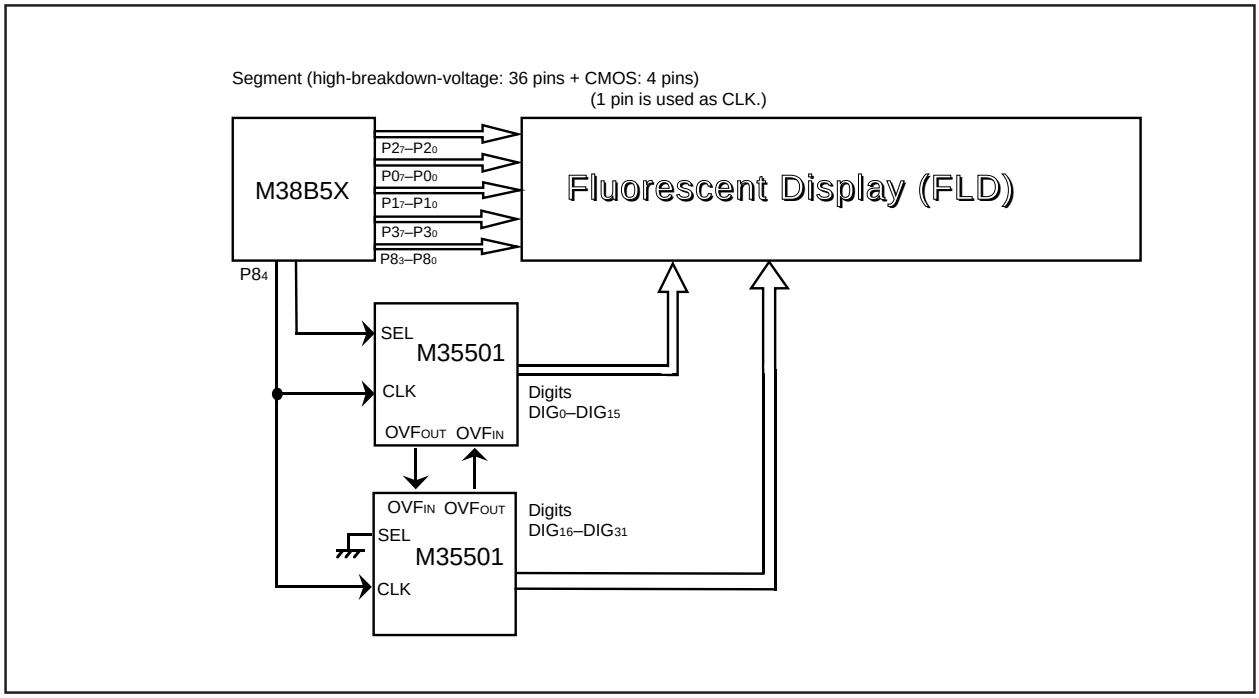


Fig. 3.12.10 Connection example with 38B5 Group microcomputer (17 to 32 digits)

RESET CIRCUIT

To reset the controller, the $\overline{\text{RESET}}$ pin should be held at "L" for 2 µs or more. Reset is released when the $\overline{\text{RESET}}$ pin is returned to "H" and the power source voltage is between 4.0 V and 5.5 V.

Notes1: Perform the reset release when CLK input signal is "L."
2: When setting the number of digits by SEL signal, optional digit counter is set to "0" by reset.

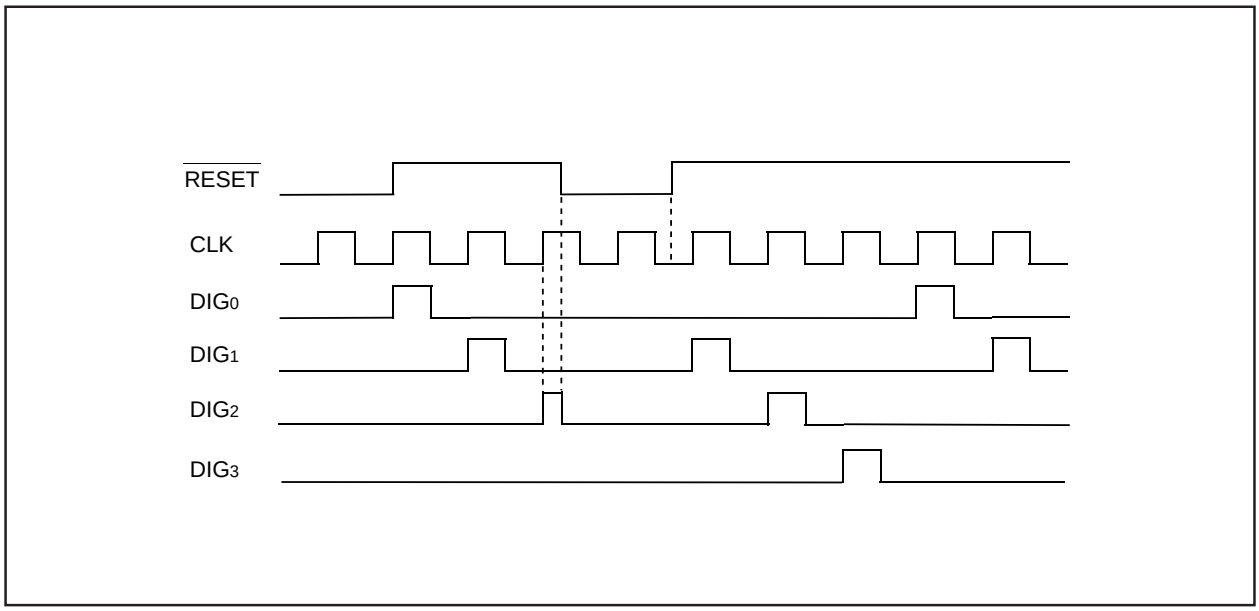


Fig. 3.12.11 Digit output waveform when reset signal is input

POWER-ON RESET

Reset can be performed automatically during power on (power-on reset) by the built-in power-on reset circuit. When using this circuit, set 100 µs or less for the period in which it takes to reach minimum operation guaranteed voltage from reset.

If the rising time exceeds 100 µs, connect the capacitor between the $\overline{\text{RESET}}$ pin and Vss at the shortest distance. Consequently, the $\overline{\text{RESET}}$ pin should be held at "L" until the minimum operation guaranteed voltage is reached.

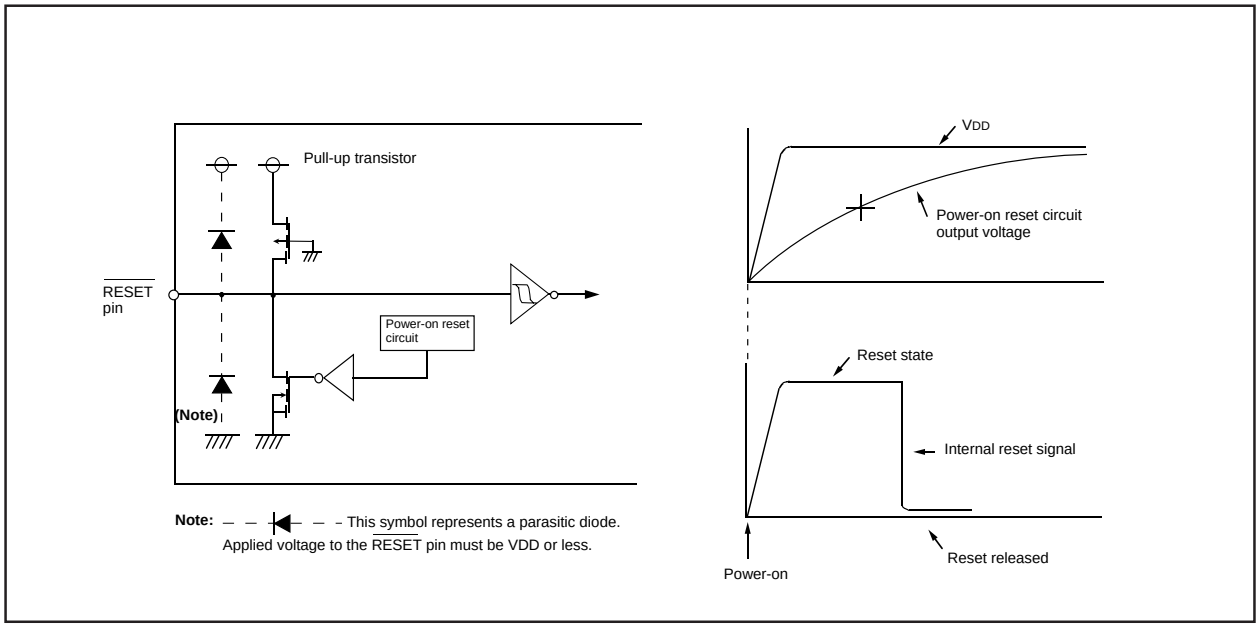


Fig. 3.12.12 Power-on reset circuit

Table 3.12.2 Absolute maximum ratings

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Power source voltage	•All voltages are based on V_{SS}.	–0.3 to 7.0	V
V_{EE}	Pull-down power source voltage	•Output transistors are off.	V_{CC} –45 to V_{CC} +0.3	V
V_I	Input voltage CLK, SEL, OVF_{IN}		–0.3 to V_{CC} +0.3	V
V_I	Input voltage $\overline{RESET}$		–0.3 to V_{CC} +0.3	V
V_O	Output voltage DIG_0–DIG_{15}		V_{CC} –45 to V_{CC} +0.3	V
V_O	Output voltage OVF_{OUT}		–0.3 to V_{CC} +0.3	V
P_d	Power dissipation	Ta = 25 °C	250	mW
T_{opr}	Operating temperature		–20 to 85	°C
T_{stg}	Storage temperature		–40 to 125	°C

Table 3.12.3 Recommended operating conditions (V_{CC} = 4.0 to 5.5 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
V_{CC}	Power source voltage	4.0	5.0	5.5	V
V_{SS}	Power source voltage		0		V
V_{EE}	Pull-down power source voltage	V_{CC} –43		V_{SS}	V
V_{IH}	“H” input voltage CLK, SEL, OVF_{IN}	0.8V_{CC}		V_{CC}	V
V_{IH}	“H” input voltage $\overline{RESET}$	0.8V_{CC}		V_{CC}	V
V_{IL}	“L” input voltage CLK, SEL, OVF_{IN}	0		0.2V_{CC}	V
V_{IL}	“L” input voltage $\overline{RESET}$	0		0.2V_{CC}	V

Table 3.12.4 Recommended operating conditions (V_{CC} = 4.0 to 5.5 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$I_{OH(peak)}$	“H” peak output current DIG_0 – DIG_{15} **(Note 1)**			–36	mA
$I_{OH(peak)}$	“H” peak output current OVF_{OUT} **(Note 1)**			–10	mA
$I_{OL(peak)}$	“L” peak output current OVF_{OUT} **(Note 1)**			10	mA
$I_{OH(avg)}$	“H” average current DIG_0 – DIG_{15} **(Note 2)**			–18	mA
$I_{OH(avg)}$	“H” average current OVF_{OUT} **(Note 2)**			–5.0	mA
$I_{OL(avg)}$	“L” average current OVF_{OUT} **(Note 2)**			5.0	mA
CLK	Clock input frequency			2	MHz

Notes 1: The peak output current is the peak current flowing in each port.
2: The average output current is an average value measured over 100 ms.

Table 3.12.5 Electrical characteristics (V_{CC} = 4.0 to 5.5 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min.	Typ.	Max.	
V_{OH}	"H" output voltage	DIG output DIG_0–DIG_{15}	I_{OH} = –18 mA	V_{CC} –2.0			V
V_{OH}	"H" output voltage	OVF_{OUT}	I_{OH} = –10 mA	V_{CC} –2.0			V
V_{OL}	"L" output voltage	OVF_{OUT}	I_{OL} = 10 mA			2.0	V
V_{T+} — V_{T-}	Hysteresis	CLK, OVF_{IN} $\overline{RESET}$	V_{CC} = 5.0 V		0.4		V
I_{IH}	"H" input current	OVF_{IN} $\overline{RESET}$	V_I = V_{CC}			5.0	µA
I_{IH}	"H" input current	CLK, SEL	V_I = V_{CC} V_{CC} = 5.0 V	30	70	140	µA
I_{IL}	"L" input current	OVF_{IN} CLK, SEL	V_I = V_{SS}			–5.0	µA
I_{IL}	"L" input current	$\overline{RESET}$	V_I = V_{SS} V_{CC} = 5.0 V	–60	–130	–185	µA
I_{LOAD}	Output load current	DIG_0 – DIG_{15}	V_{EE} = V_{CC} –43 V V_{OL} = V_{CC} Output transistors are off.	500	650	800	µA
I_{LEAK}	Output leakage current	DIG_0–DIG_{15}	V_{EE} = V_{CC} –43 V V_{OL} = V_{CC} –43 V Output transistors are off.			–10	µA
I_{CC}	Power source	V_{CC} = 5.0 V, CLK = 100 kHz Output transistors are off.			50		µA

Table 3.12.6 Timing requirements (V_{CC} = 4.0 to 5.5 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{w(\overline{RESET})}$	Reset input “L” pulse width	2			μs
$t_{c(CLK)}$	Clock input cycle time	500			ns
$t_{wH(CLK)}$	Clock input “H” pulse width	200			ns
$t_{wL(CLK)}$	Clock input “L” pulse width	200			ns
$t_{su(SEL)}$	Select input setup time	500			ns
$t_{h(SEL)}$	Select input hold time	500			ns
$t_{h(CLK)}$	Clock input setup time	500			ns

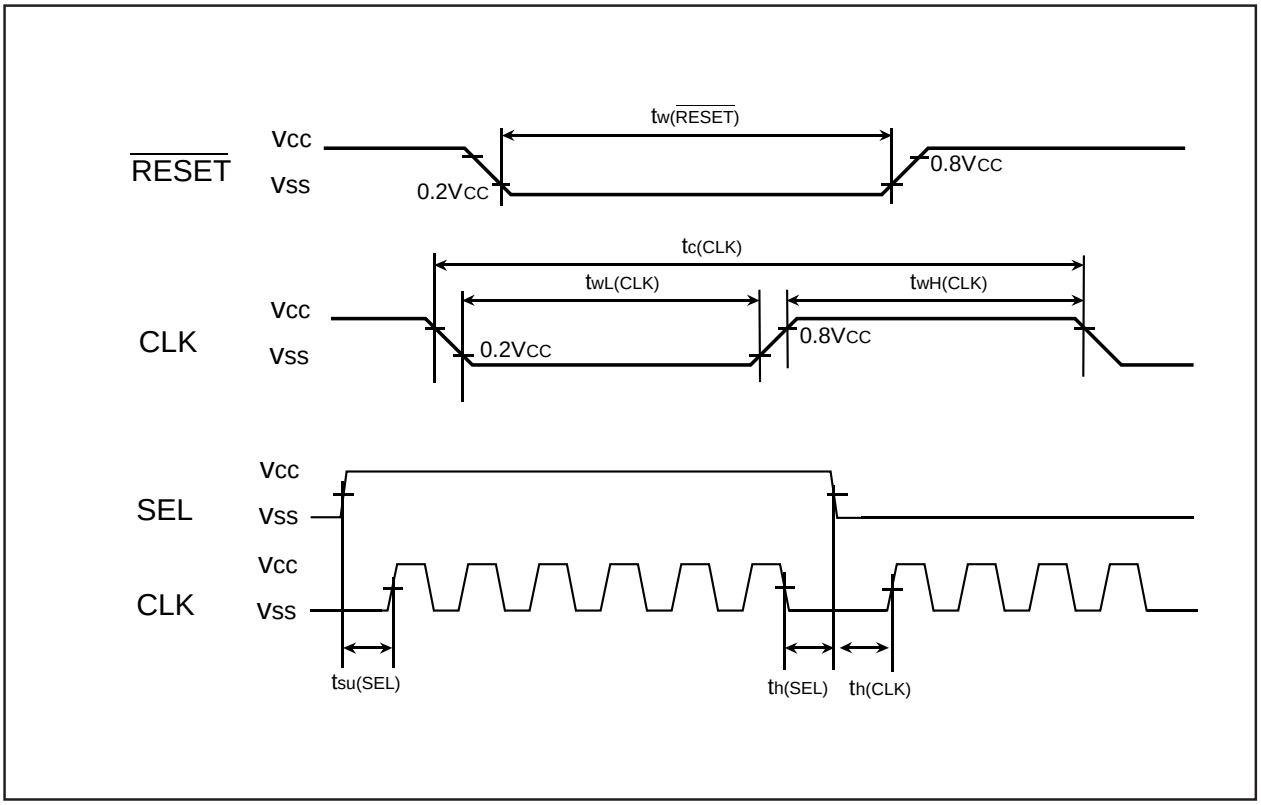


Fig. 3.12.13 Timing diagram

3.13 SFR memory map

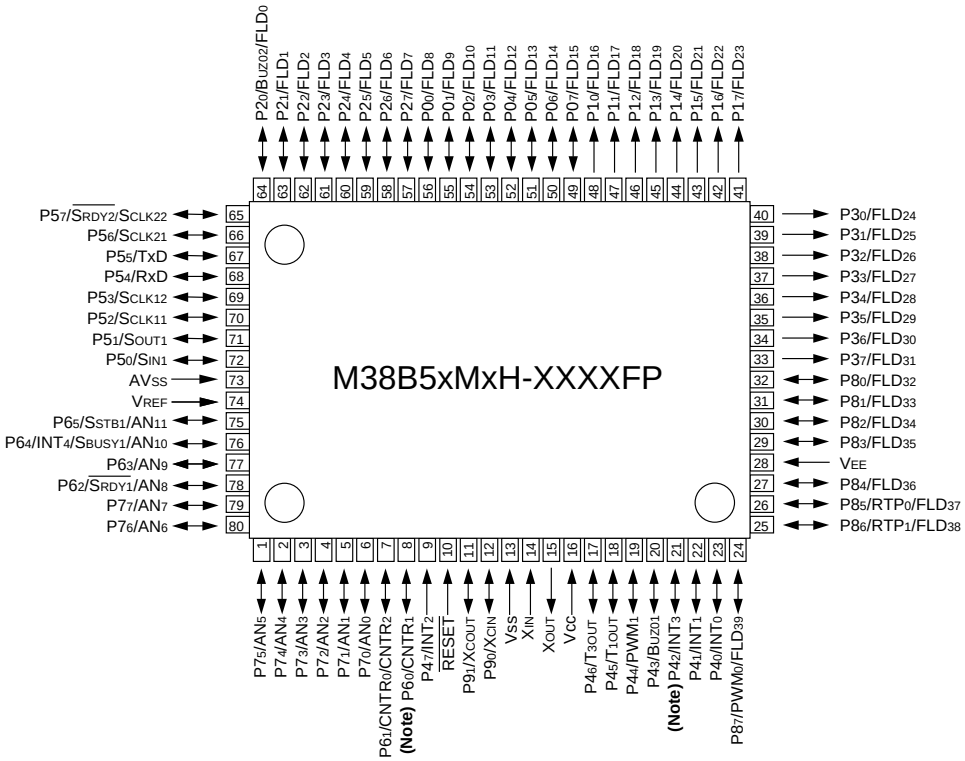
0000_{16}	Port P0 (P0)
0001_{16}	Port P0 direction register (P0D)
0002_{16}	Port P1 (P1)
0003_{16}	
0004_{16}	Port P2 (P2)
0005_{16}	Port P2 direction register (P2D)
0006_{16}	Port P3 (P3)
0007_{16}	
0008_{16}	Port P4 (P4)
0009_{16}	Port P4 direction register (P4D)
$000A_{16}$	Port P5 (P5)
$000B_{16}$	Port P5 direction register (P5D)
$000C_{16}$	Port P6 (P6)
$000D_{16}$	Port P6 direction register (P6D)
$000E_{16}$	Port P7 (P7)
$000F_{16}$	Port P7 direction register (P7D)
0010_{16}	Port P8 (P8)
0011_{16}	Port P8 direction register (P8D)
0012_{16}	Port P9 (P9)
0013_{16}	Port P9 direction register (P9D)
0014_{16}	PWM register (high-order) (PWMH)
0015_{16}	PWM register (low-order) (PWM L)
0016_{16}	Baud rate generator (BRG)
0017_{16}	UART control register (UARTCON)
0018_{16}	Serial I/O1 automatic transfer data pointer (SIO1DP)
0019_{16}	Serial I/O1 control register 1 (SIO1CON1)
$001A_{16}$	Serial I/O1 control register 2 (SIO1CON2)
$001B_{16}$	Serial I/O1 register/Transfer counter (SIO1)
$001C_{16}$	Serial I/O1 control register 3 (SIO1CON3)
$001D_{16}$	Serial I/O2 control register (SIO2CON)
$001E_{16}$	Serial I/O2 status register (SIO2STS)
$001F_{16}$	Serial I/O2 transmit/receive buffer register (TB/RB)

0020_{16}	Timer 1 (T1)
0021_{16}	Timer 2 (T2)
0022_{16}	Timer 3 (T3)
0023_{16}	Timer 4 (T4)
0024_{16}	Timer 5 (T5)
0025_{16}	Timer 6 (T6)
0026_{16}	PWM control register (PWMCON)
0027_{16}	Timer 6 PWM register (T6PWM)
0028_{16}	Timer 12 mode register (T12M)
0029_{16}	Timer 34 mode register (T34M)
$002A_{16}$	Timer 56 mode register (T56M)
$002B_{16}$	Watchdog timer control register (WDTCON)
$002C_{16}$	Timer X (low-order) (TXL)
$002D_{16}$	Timer X (high-order) (TXH)
$002E_{16}$	Timer X mode register 1 (TXM1)
$002F_{16}$	Timer X mode register 2 (TXM2)
0030_{16}	Interrupt interval determination register (IID)
0031_{16}	Interrupt interval determination control register (IIDCON)
0032_{16}	A-D control register (ADCON)
0033_{16}	A-D conversion register (low-order) (ADL)
0034_{16}	A-D conversion register (high-order) (ADH)
0035_{16}	
0036_{16}	
0037_{16}	
0038_{16}	
0039_{16}	Interrupt source switch register (IFR)
$003A_{16}$	Interrupt edge selection register (INTEDGE)
$003B_{16}$	CPU mode register (CPUM)
$003C_{16}$	Interrupt request register 1(IREQ1)
$003D_{16}$	Interrupt request register 2(IREQ2)
$003E_{16}$	Interrupt control register 1(ICON1)
$003F_{16}$	Interrupt control register 2(ICON2)

$0EF0_{16}$	Pull-up control register 1 (PULL1)
$0EF1_{16}$	Pull-up control register 2 (PULL2)
$0EF2_{16}$	P1FLDRAM write disable register (P1FLDRAM)
$0EF3_{16}$	P3FLDRAM write disable register (P3FLDRAM)
$0EF4_{16}$	FLDC mode register (FLDM)
$0EF5_{16}$	Tdisp time set register (TDISP)
$0EF6_{16}$	Toff1 time set register (TOFF1)
$0EF7_{16}$	Toff2 time set register (TOFF2)

$0EF8_{16}$	FLD data pointer (FLDDP)
$0EF9_{16}$	Port P0FLD/port switch register (P0FPR)
$0EFA_{16}$	Port P2FLD/port switch register (P2FPR)
$0EFB_{16}$	Port P8FLD/port switch register (P8FPR)
$0EFC_{16}$	Port P8FLD output control register (P8FLDCON)
$0EFD_{16}$	Buzzer output control register (BUZCON)
$0EFE_{16}$	
$0EFF_{16}$	

3.14 Pin configuration



Note: In the mask option type P, INT3 and CNTR1 cannot be used.

(Top view)

Package type: 80P6N-A
80-pin plastic molded QFP

MITSUBISHI SEMICONDUCTORS
USER'S MANUAL
38B5 Group

Nov. 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
38B5 Group

MITSUBISHI ELECTRIC CORPORATION
HEAD OFFICE: MITSUBISHI DENKI BLDG., MARUNOUCHI, TOKYO 100. TELEX: J24532 CABLE: MELCO TOKYO

New publication, effective Nov. 1998.
Specifications subject to change without notice.

REVISION DESCRIPTION LIST	38B5 Group User's Manual

Rev. No.	Revision Description	Rev. date
1.0	First Edition	981202