To all our customers

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

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

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

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

RENESAS
Renesas Technology Corp.

MITSUBISHI 8-BIT SINGLE-CHIP MICROCOMPUTER

740 FAMILY / 38000 SERIES

38B7

Group

User's Manual

http://www.infomicom.maec.co.jp/indexe.htm

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

RENESAS
Renesas Technology Corp.

Rev. 1.3
Revision date: Jan. 29, 2003

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 nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

- These materials are intended as a reference to assist our customers in the selection of the 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, programs, algorithms, or circuit application examples contained in these materials.
- All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by 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.
  The information described here may contain technical inaccuracies or typographical errors. Mitsubishi Electric Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
  Please also pay attention to information published by Mitsubishi Electric Corporation by various means, including the Mitsubishi Semiconductor home page (http://www.mitsubishichips.com).
- When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Mitsubishi Electric Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained 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.

REVISION HISTORY	38B7 GROUP USER'S MANUAL

Rev.	Date	Description	
		Page	Summary
1.0	07/07/00		First Edition
1.1	03/10/00	74	Mask options B to G are shaded to show that they cannoto be specified. Note 4 added.
		100	Absolute maximum ratings VEE VCC–45 to VCC +0.3 VI VCC–45 to VCC +0.3 VO VCC–45 to VCC +0.3
1.2	11/01/01	1-2	Explanations of "DESCRIPTION" are partly eliminated.
		1-2	Oscillation frequency value of "FEATURES" are partly revised.
		1-7	Figure 3 is partly revised.
		1-8	Figure 4 is partly revised.
		1-75	"MASK OPTION OF PULL-DOWN RESISTOR" is eliminated.
1.3	01/29/03	1-21	Explanations of "■Note" are revised.
		1-41	"■Notes" is added.
		1-100	"Electric Characteristic Differences Between Mask ROM and Flash Memory Version MCUs" is added.
		2-91	Sub clause name and explanations of "(7) Setting procedure when serial I/O2 transmit interrupt is used" are revised.
		2-164	Clause name and explanations of "2.11.3 Each port state during "L" state of $\overline{\text{RESET}}$ pin" are revised.
		2-164	Table name of Table 2.11.1 is revised.
		2-164	Note of Table 2.11.1 is eliminated.
		2-177	Table 2.13.1 is partly revised.
		2-177	Explanations of "2.13.5 Serial I/O mode" are partly revised.
		2-177	Table 2.13.2 is partly revised.
		3-10	Figure 3.2.2 is revised.
		3-15	Sub clause name and explanations of "(1) Change of relevant register settings" are revised.
		3-21	Sub clause name and explanations of "(7) Setting procedure when serial I/O2 transmit interrupt is used" are revised.
		3-23	Clause name and explanations of "3.3.11 Each port state during "L" state of $\overline{\text{RESET}}$ pin" are revised.
		3-23	Table name of Table 3.3.3 is revised.

Preface

This user’s manual describes Mitsubishi’s CMOS 8-bit microcomputers 38B7 Group.
After reading this manual, the user should have a through knowledge of the functions and features of the 38B7 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 “Mitsubishi Microcomputer Development Support Tools” Homepage (http://www.tool-spt.maec.co.jp/index_e.htm).

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

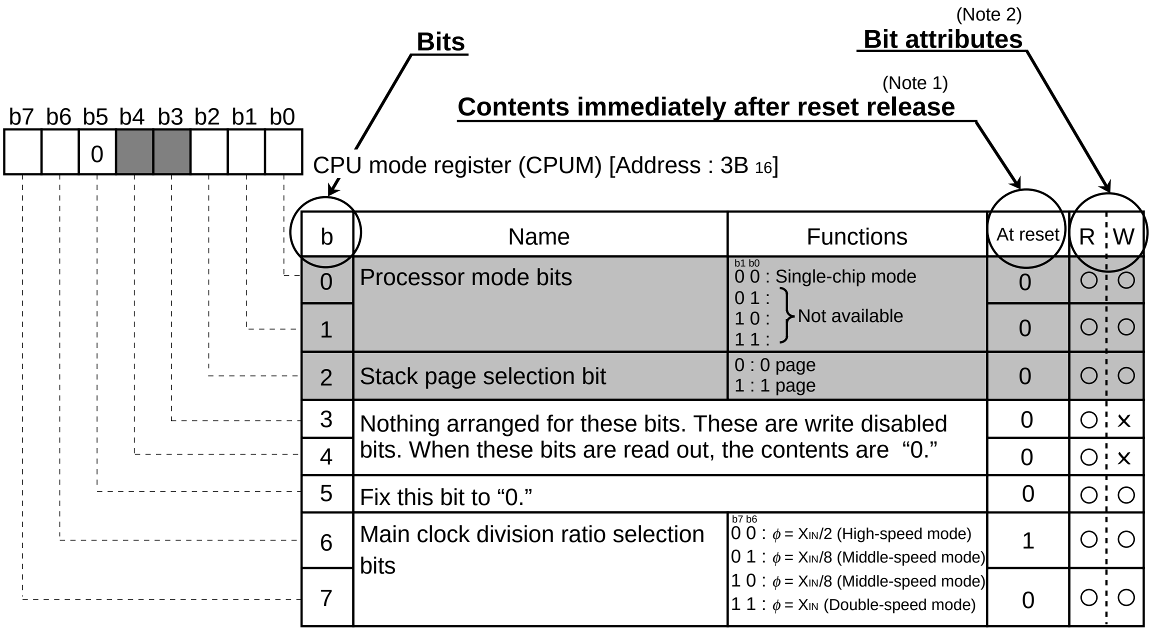
- **CHAPTER 1 HARDWARE**
  This chapter describes features of the microcomputer and operation of each peripheral function.

- **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.

2. Structure of Register

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



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

b	Name	Functions	At reset	R	W
0	Processor mode bits	b1 b0 0 0 : Single-chip mode 0 1 : 1 0 : Not available 1 1 :	0	○	○
1			0	○	○
2	Stack page selection bit	0 : 0 page 1 : 1 page	0	○	○
3	Nothing arranged for these bits. These are write disabled bits. When these bits are read out, the contents are "0."		0	○	×
4			0	○	×
5	Fix this bit to "0."		0	○	○
6	Main clock division ratio selection bits	b7 b6 0 0 : ϕ = X_{IN}/2 (High-speed mode) 0 1 : ϕ = X_{IN}/8 (Middle-speed mode) 1 0 : ϕ = X_{IN}/8 (Middle-speed mode) 1 1 : ϕ = X_{IN} (Double-speed mode)	1	○	○
7			0	○	○

(dark grey) : Bit in which nothing is arranged   (light grey) : Bit that is not used for control of the corresponding function

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

Table of contents

CHAPTER 1 HARDWARE

CHAPTER 2 APPLICATION

CHAPTER 3 APPENDIX

List of figures

CHAPTER 1 HARDWARE

CHAPTER 2 APPLICATION

CHAPTER 3 APPENDIX

List of tables

CHAPTER 1 HARDWARE

CHAPTER 2 APPLICATION

CHAPTER 3 APPENDIX

MEMORANDUM

CHAPTER 1
HARDWARE

DESCRIPTION
FEATURES
APPLICATION
PIN CONFIGURATION
FUNCTIONAL BLOCK
PIN DESCRIPTION
PART NUMBERING
GROUP EXPANSION
FUNCTIONAL DESCRIPTION
NOTES ON PROGRAMMING
NOTES ON USAGE
DATA REQUIRED FOR MASK ORDERS

DESCRIPTION/FEATURES

DESCRIPTION

The 38B7 group is the 8-bit microcomputer based on the 740 family core technology.

The 38B7 group has six 8-bit timers, one 16-bit timer, a fluorescent display automatic display circuit, 16-channel 10-bit A-D converter, a serial I/O with automatic transfer function, which are available for controlling musical instruments and household appliances.

FEATURES

<Microcomputer mode>

- Basic machine-language instructions 71
- The minimum instruction execution time .... 0.48 µs (at 4.2 MHz oscillation frequency)
- Memory size
  ROM .... 60K bytes
  RAM .... 2048 bytes
- Programmable input/output ports 75
- High-breakdown-voltage output ports 52
- Software pull-up resistors . (Ports $P6_4$ to $P6_7$, P7, $P8_0$ to $P8_3$, P9, PA, PB)
- Interrupts .... 22 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
- Serial I/O3 (Clock-synchronized) .... 8-bit ✕ 1
- PWM .... 14-bit ✕ 1
  8-bit ✕ 1 (also functions as timer 6)
- A-D converter .... 10-bit ✕ 16 channels
- D-A converter .... 1 channel
- Fluorescent display function .... Total 56 control pins
- Interrupt interval determination function 1
  (Serviceable even in low-speed mode)
- Watchdog timer .... 16-bit ✕ 1
- Buzzer output 1
- Two clock generating circuits
  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 4.2 MHz oscillation frequency and high-speed selected)
  In middle-speed mode .... 2.7 to 5.5 V (*)
  (at 4.2 MHz oscillation frequency and middle-speed selected)
  In low-speed mode .... 2.7 to 5.5 V (*)
  (at 32 kHz oscillation frequency)
  (*: 4.0 to 5.5 V for Flash memory version)
- Power dissipation
  In high-speed mode .... 35 mW
  (at 4.2 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

<Flash memory mode>

- Supply voltage .... V_{CC} = 5 V ± 10 %
- Program/Erase voltage .... V_{PP} = 11.7 to 12.6 V
- Programming method .... Programming in unit of byte
- Erasing method
  Batch erasing .... Parallel/Serial I/O mode
  Block erasing .... CPU reprogramming mode
- Program/Erase control by software command
- Number of times for programming/erasing 100
- Operating temperature range (at programming/erasing)
  .... Normal temperature

■Notes

1. The flash memory version cannot be used for application embedded in the MCU card.
2. Power source voltage Vcc of the flash memory version is 4.0 to 5.5 V.

APPLICATION

Musical instruments, VCR, household appliances, etc.

PIN CONFIGURATION (TOP VIEW)

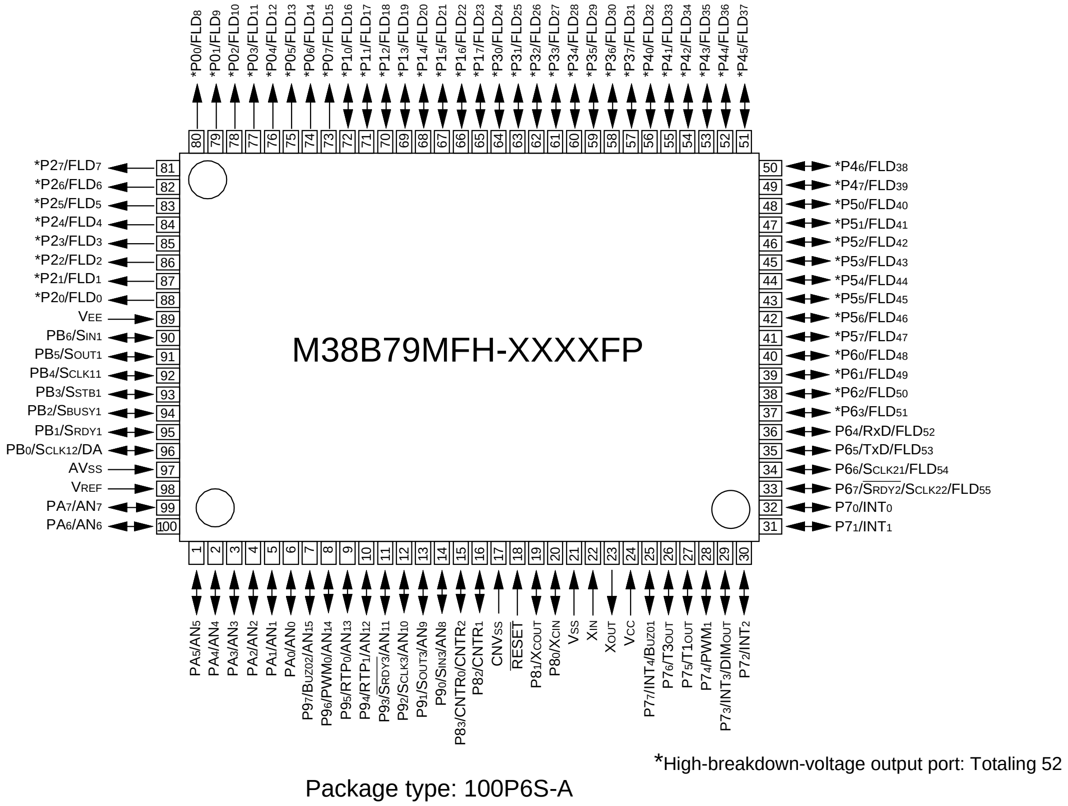


Fig. 1 Pin configuration of M38B79MFH-XXXXFP

FUNCTIONAL BLOCK

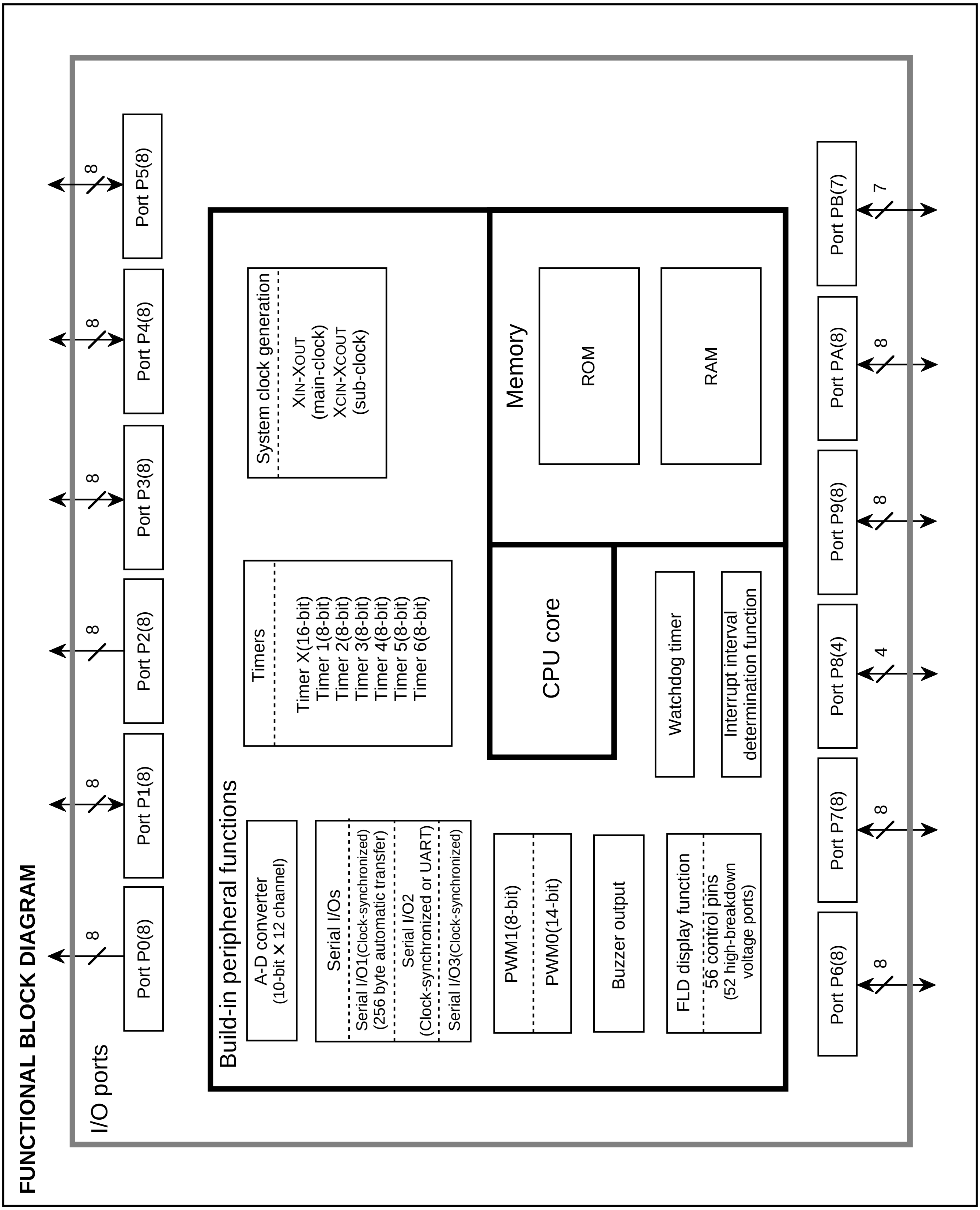


Fig. 2 Functional block diagram

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}.	
CNV_{SS}	CNV_{SS}	• Connect to V_{SS}. • VPP power input pin in flash memory mode.	
V_{EE}	Pull-down power source	• Apply voltage supplied to pull-down resistors of ports P0, P1, P2 and P3.	
V_{REF}	Reference voltage	• Reference voltage input pin for A-D converter.	
AV_{SS}	Analog power source	• Analog power source input pin for A-D converter. • Connect to V_{SS}.	
$\overline{RESET}$	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		
$P0_0/FLD_8$–$P0_7/FLD_{15}$	Output port P0	• 8-bit output port. • High-breakdown-voltage P-channel open-drain output structure. • A pull-down resistor is built in between port P0 and the V_{EE} pin. • At reset, this port is set to V_{EE} level.	• FLD automatic display pins
$P1_0/FLD_{16}$–$P1_7/FLD_{23}$	I/O port P1	• 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. • Low-voltage input level. • High-breakdown-voltage P-channel open-drain output structure. • A pull-down resistor is built in between port P1 and the V_{EE} pin. • At reset, this port is set to V_{EE} level.	• FLD automatic display pins
$P2_0/FLD_0$–$P2_7/FLD_7$	Output port P2	• 8-bit output port with the same function as port P0. • High-breakdown-voltage P-channel open-drain output structure. • A pull-down resistor is built in between port P2 and the V_{EE} pin. • At reset, this port is set to V_{EE} level.	• FLD automatic display pins
$P3_0/FLD_{24}$–$P3_7/FLD_{31}$	I/O port P3	• 8-bit I/O port with the same function as port P1. • Low-voltage input level. • High-breakdown-voltage P-channel open-drain output structure. • A pull-down resistor is built in between port P3 and the V_{EE} pin. • At reset, this port is set to V_{EE} level.	• FLD automatic display pins
$P4_0/FLD_{32}$–$P4_7/FLD_{39}$	I/O port P4	• 8-bit I/O port with the same function as port P1. • Low-voltage input level. • High-breakdown-voltage P-channel open-drain output structure. • A pull-down resistor is not built in between port P4 and the V_{EE} pin.	• FLD automatic display pins
$P5_0/FLD_{40}$–$P5_7/FLD_{47}$	I/O port P5	• 8-bit I/O port with the same function as port P1. • Low-voltage input level. • High-breakdown-voltage P-channel open-drain output structure. • A pull-down resistor is not built in between port P5 and the V_{EE} pin.	• FLD automatic display pins
$P6_0/FLD_{48}$–$P6_3/FLD_{51}$	I/O port P6	• 4-bit I/O port with the same function as port P1. • Low-voltage input level. • High-breakdown-voltage P-channel open-drain output structure. • A pull-down resistor is not built in between port P6 and the V_{EE} pin.	• FLD automatic display pins

PIN DESCRIPTION

Table 2 Pin description (2)

Pin	Name	Function	Function except a port function
P6$_4$/R$_X$D/FLD$_{52}$, P6$_5$/T$_X$D/FLD$_{53}$, P6$_6$/S$_{CLK21}$/FLD$_{54}$, P6$_7$/$\overline{S_{RDY2}}$/S$_{CLK22}$/ FLD$_{55}$,	I/O port P6	• 4-bit I/O port . • Low-voltage input level for input ports. • CMOS compatible input level for RxD, S$_{CLK21}$, S$_{CLK22}$. • CMOS 3-state output structure.	• FLD automatic display pins • Serial I/O2 function pins
P7$_0$/INT$_0$, P7$_1$/INT$_1$, P7$_2$/INT$_2$,	I/O port P7	• 8-bit I/O port. • CMOS compatible input level. • CMOS 3-state output structure.	• Interrupt input pins
P7$_3$/INT$_3$/DIM$_{OUT}$,			• Interrupt input pin • Dimmer signal output pin
P7$_4$/PWM$_1$			• PWM output pin
P7$_5$/T1$_{OUT}$, P7$_6$/T3$_{OUT}$,			• Timer output pins
P7$_7$/INT$_4$/B$_{UZ01}$			• Interrupt input pin • Buzzer output pin
P8$_0$/X$_{CIN}$, P8$_1$/X$_{COUT}$	I/O port P8	• 4-bit I/O port with the same function as port P7. • 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)
P8$_2$/CNTR$_1$, P8$_3$/CNTR$_0$/CNTR$_2$			• Timer input pin • Timer I/O pin
P9$_0$/S$_{IN3}$/AN$_8$, P9$_1$/S$_{OUT3}$/AN$_9$, P9$_2$/S$_{CLK3}$/AN$_{10}$, P9$_3$/$\overline{S_{RDY3}}$/AN$_{11}$,	I/O port P9	• 8-bit I/O port with the same function as port P7. • CMOS compatible input level. • CMOS 3-state output structure.	• Serial I/O3 function pins • A-D converter input pins
P9$_4$/RTP$_1$/AN$_{12}$, P9$_5$/RTP$_0$/AN$_{13}$			• Real time port output pins • A-D converter input pins
P9$_6$/PWM$_0$/AN$_{14}$			• 14-bit PWM output pin • A-D converter input pin
P9$_7$/B$_{UZ02}$/AN$_{15}$			• Buzzer output pin • A-D converter input pin
PA$_0$/AN$_0$–PA$_7$/AN$_7$	I/O port PA	• 8-bit I/O port with the same function as port P7. • CMOS compatible input level. • CMOS 3-state output structure.	• A-D converter input pin
PB$_0$/S$_{CLK12}$/DA	I/O port PB	• 7-bit I/O port with the same function as port P7. • CMOS compatible input level. • CMOS 3-state output structure.	• Serial I/O1 function pin • D-A converter output pin
PB$_1$/S$_{RDY1}$, PB$_2$/S$_{BUSY1}$, PB$_3$/S$_{STB1}$, PB$_4$/S$_{CLK11}$, PB$_5$/S$_{OUT1}$, PB$_6$/S$_{IN1}$			• Serial I/O1 function pins

Product M38B7 9 M F H - AXXX FP

Package type
FP : 100P6S-A package

ROM number
Omitted in Flash memory version

ROM/Flash memory size
1 : 4096 bytes
2 : 8192 bytes
3 : 12288 bytes
4 : 16384 bytes
5 : 20480 bytes
6 : 24576 bytes
7 : 28672 bytes
8 : 32768 bytes
9 : 36864 bytes
A : 40960 bytes
B : 45056 bytes
C : 49152 bytes
D : 53248 bytes
E : 57344 bytes
F : 61440 bytes

The first 128 bytes and the last 2 bytes of ROM are reserved areas ; they cannot be used for users.

Memory type
M : Mask ROM version
F : Flash memory version

RAM size
0 : 192 bytes
1 : 256 bytes
2 : 384 bytes
3 : 512 bytes
4 : 640 bytes
5 : 768 bytes
6 : 896 bytes
7 : 1024 bytes
8 : 1536 bytes
9 : 2048 bytes

Fig. 3 Part numbering

GROUP EXPANSION

Mitsubishi plans to expand the 38B7 group as follows.

Memory Type

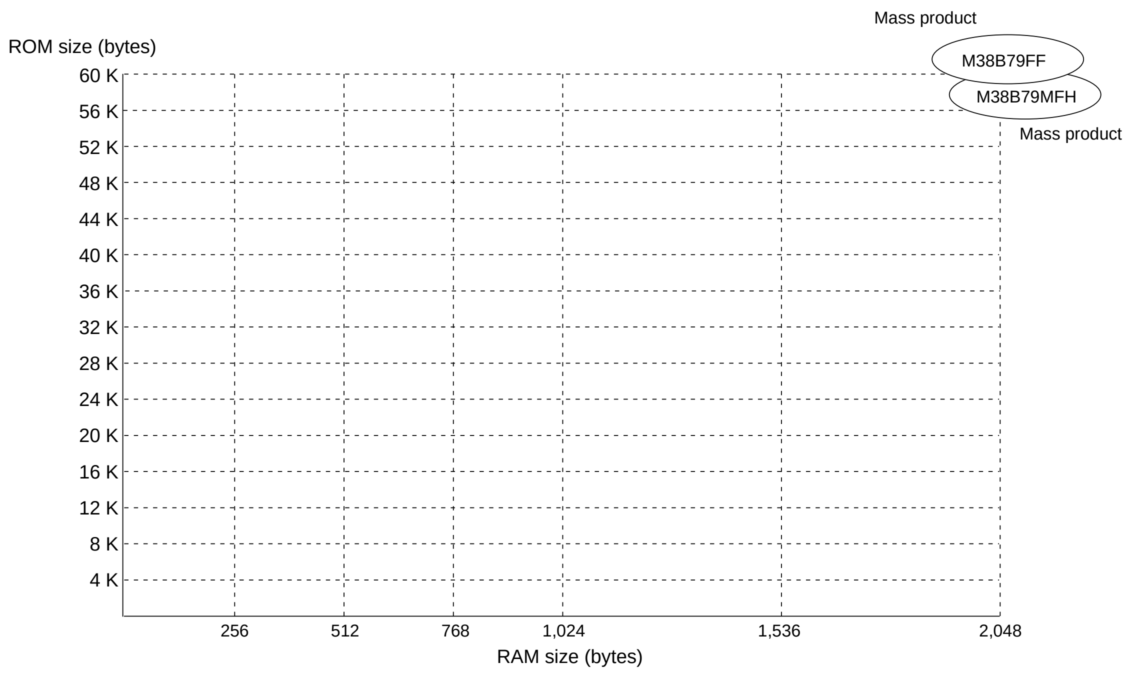
Support for Mask ROM and Flash memory versions.

Memory Size

Flash memory size .... 60K bytes
Mask ROM size .... 60K bytes
RAM size ....2048 bytes

Package

100P6S-A .... 0.65 mm-pitch plastic molded QFP



Note : Products under development: 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. 2001**

Product	ROM size (bytes) ROM size for User ( )	RAM size (bytes)	Package	Remarks
M38B79MFH-XXXXFP	61440 (61310)	2048	100P6S-A	Mask ROM version
M38B79FFFP				Flash memory version

FUNCTIONAL DESCRIPTION
Central Processing Unit (CPU)

The 38B7 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.

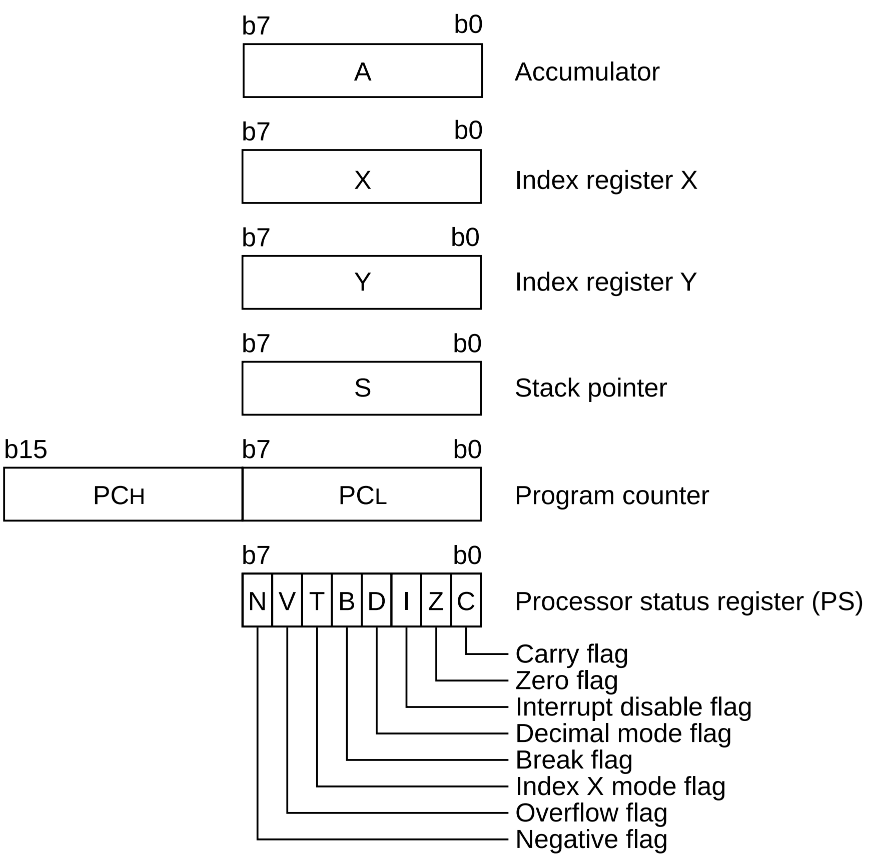


Fig. 5 740 Family CPU register structure

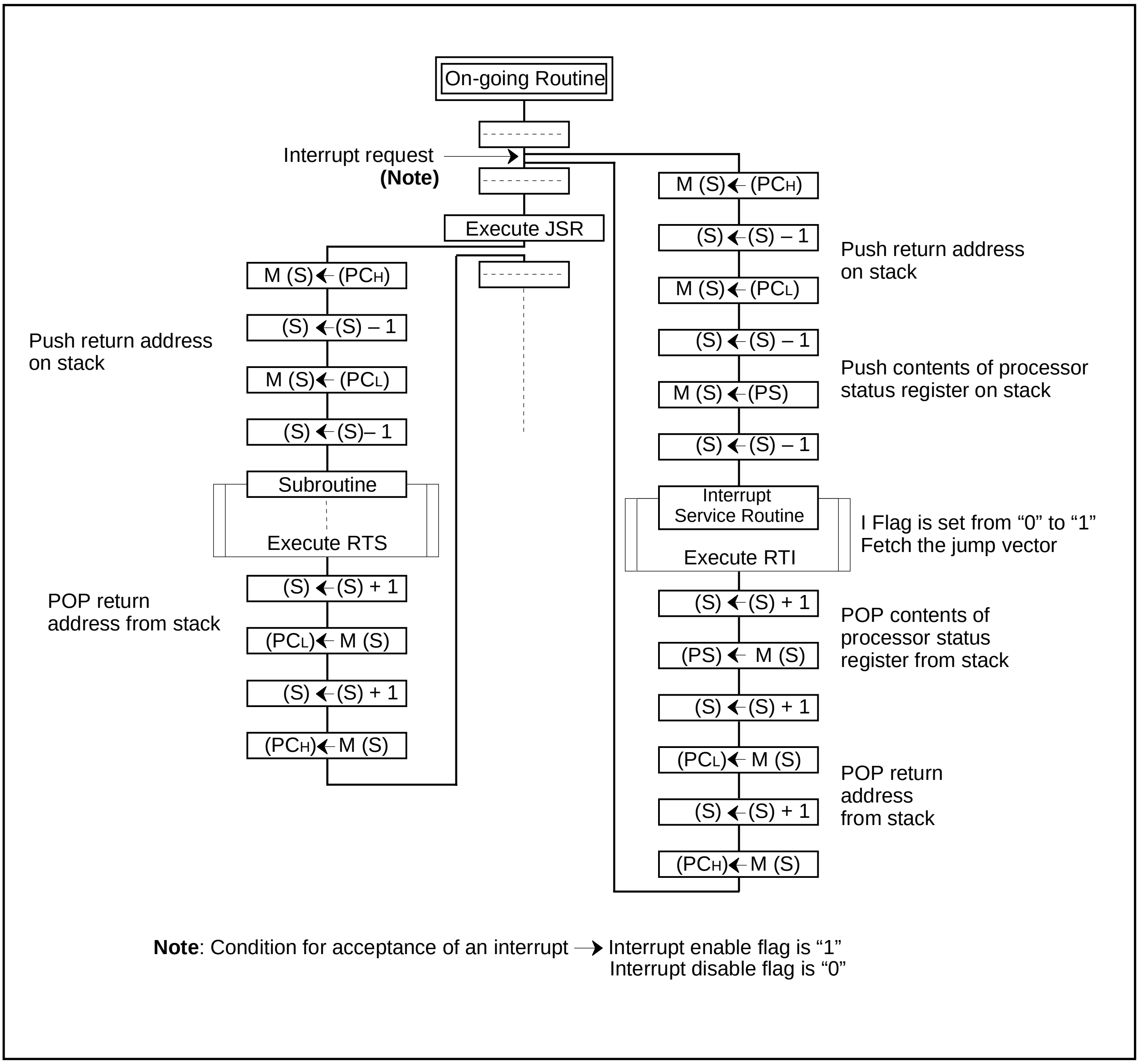


Fig. 6 Register push and pop at interrupt generation and subroutine call

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)] $003B_{16}$

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 $003B_{16}$.

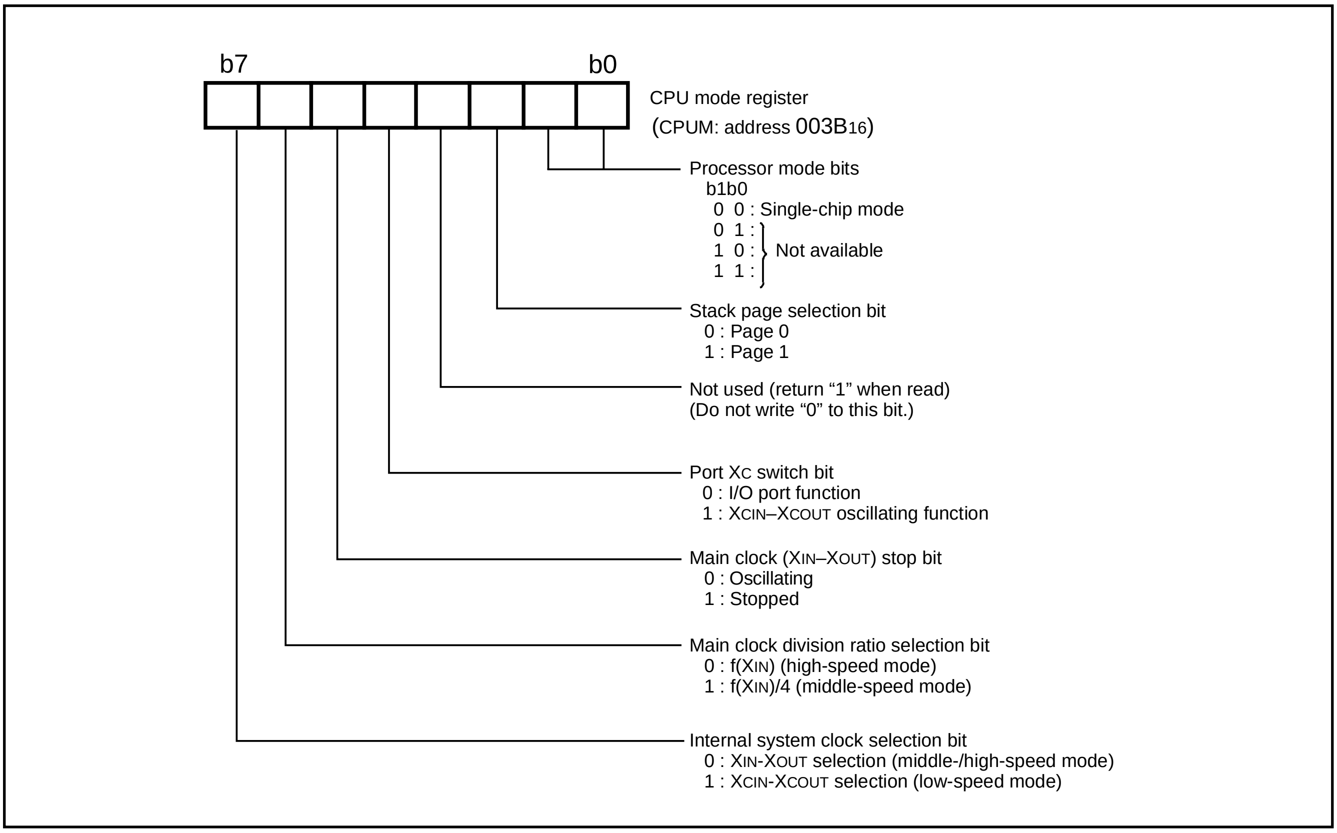


Fig. 7 Structure of CPU mode register

MEMORY
Special Function Register (SFR) Area
The special function register (SFR) area contains control registers for I/O ports, timers and other functions.

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 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.

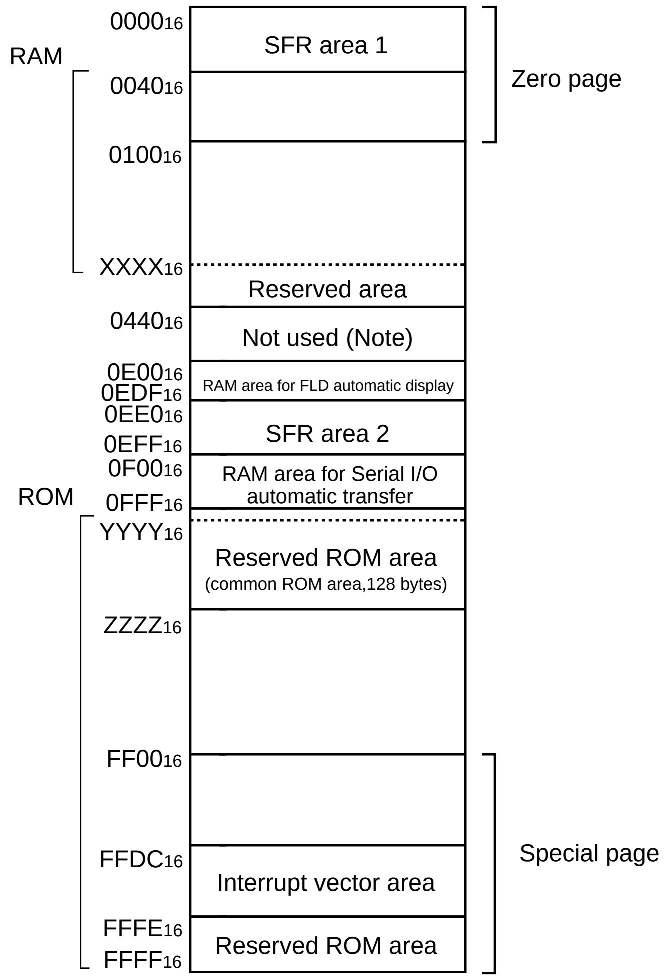
Special Page
The special page addressing mode can be used to specify memory addresses in the special page area. Access to this area with only 2 bytes is possible in the special page addressing mode.

RAM area

RAM size (byte)	Address $XXXX_{16}$
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 $YYYY_{16}$	Address $ZZZZ_{16}$
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

Address	Register	Address	Register
0000_{16}	Port P0 (P0)	0020_{16}	Timer 1 (T1)
0001_{16}		0021_{16}	Timer 2 (T2)
0002_{16}	Port P1 (P1)	0022_{16}	Timer 3 (T3)
0003_{16}	Port P1 direction register (P1D)	0023_{16}	Timer 4 (T4)
0004_{16}	Port P2 (P2)	0024_{16}	Timer 5 (T5)
0005_{16}		0025_{16}	Timer 6 (T6)
0006_{16}	Port P3 (P3)	0026_{16}	PWM control register (PWMCON)
0007_{16}	Port P3 direction register (P3D)	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)	$002B_{16}$	D-A conversion register (DA)
$000C_{16}$	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}	AD/DA control register (ADCON)
0013_{16}	Port P9 direction register (P9D)	0033_{16}	A-D conversion register (low-order) (ADL)
0014_{16}	Port PA (PA)	0034_{16}	A-D conversion register (high-order) (ADH)
0015_{16}	Port PA direction register (PAD)	0035_{16}	PWM register (high-order) (PWMH)
0016_{16}	Port PB (PB)	0036_{16}	PWM register (low-order) (PWML)
0017_{16}	Port PB direction register (PBD)	0037_{16}	Baud rate generator (BRG)
0018_{16}	Serial I/O1 automatic transfer data pointer (SIO1DP)	0038_{16}	UART control register (UARTCON)
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)

Address	Register	Address	Register	
$0EEC_{16}$	Serial I/O3 control register (SIO3CON)	$0EF6_{16}$	Toff1 time set register (TOFF1)	
$0EED_{16}$	Serial I/O3 register (SIO3)	$0EF7_{16}$	Toff2 time set register (TOFF2)	
$0EEE_{16}$	Watchdog timer control register (WDTCON)	$0EF8_{16}$	FLD data pointer (FLDDP)	
$0EEF_{16}$	Pull-up control register 3 (PULL3)	$0EF9_{16}$	Port P4 FLD/Port switch register (P4FPR)	
$0EF0_{16}$	Pull-up control register 1 (PULL1)	$0EFA_{16}$	Port P5 FLD/Port switch register (P5FPR)	
$0EF1_{16}$	Pull-up control register 2 (PULL2)	$0EFB_{16}$	Port P6 FLD/Port switch register (P6FPR)	
$0EF2_{16}$	Port P0 digit output set switch register (P0DOR)	$0EFC_{16}$	FLD output control register (FLDCON)	
$0EF3_{16}$	Port P2 digit output set switch register (P2DOR)	$0EFD_{16}$	Buzzer output control register (BUZCON)	
$0EF4_{16}$	FLDC mode register (FLDM)	$0EFE_{16}$	Flash memory control register (FCON)	**(Note)**
$0EF5_{16}$	Tdisp time set register (TDISP)	$0EFF_{16}$	Flash command register (FCMD)	**(Note)**

Note: Flash memory version only.

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

I/O PORTS
[Direction Registers] PiD

The 38B7 group has 75 programmable I/O pins arranged in ten individual I/O ports (P1, P3, P4, P5, P6, P7, P8, P9, PA and PB). 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 38B7 group has seven ports with high-breakdown-voltage pins (ports P0 to P5 and P6$_0$–P6$_3$). The high-breakdown-voltage ports have P-channel open-drain output with Vcc – 45 V of breakdown voltage. Each pin in ports P0 to 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 P6$_4$–P6$_7$, P7, P8$_0$–P8$_3$, P9, PA and PB 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.

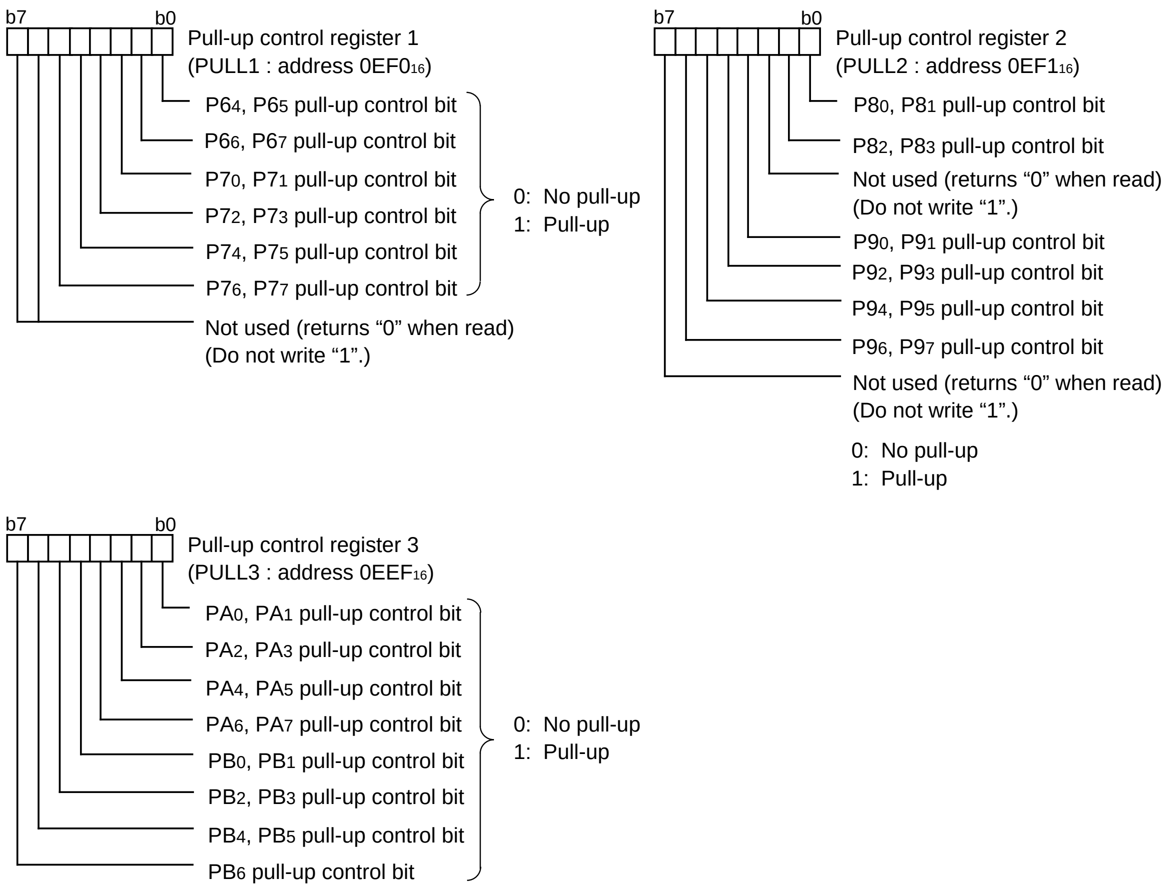


Fig. 10 Structure of pull-up control registers (PULL1, PULL2 and PULL3)

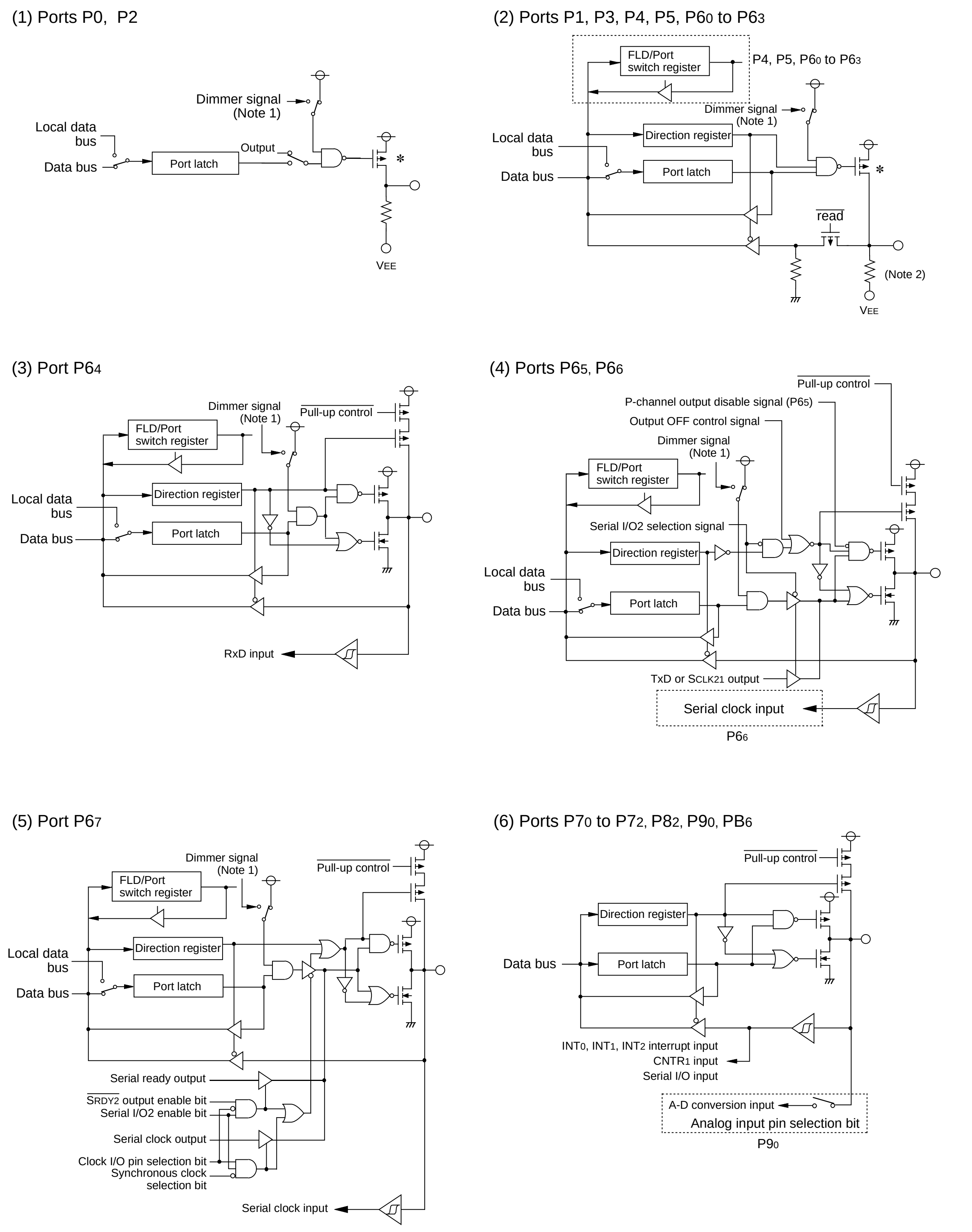
Table 6 List of I/O port functions (1)

Pin	Nama	Input/Output	I/O Format	Non-Port Function	Related SFRs	Ref.No.
P0$_0$/FLD$_8$–P0$_7$/FLD$_{15}$	Port P0	Output	High-breakdown voltage P-channel open-drain output with pull-down resistor	FLD automatic display function	FLDC mode register P0 digit output set switch register	(1)
P1$_0$/FLD$_{16}$–P1$_7$/FLD$_{23}$	Port P1	Input/output, individual bits	Low-voltage input level High-breakdown voltage P-channel open-drain output with pull-down resistor		FLDC mode register	(2)
P2$_0$/FLD$_0$–P2$_7$/FLD$_7$	Port P2	Output	High-breakdown voltage P-channel open-drain output with pull-down resistor		FLDC mode register P2 digit output set switch register	(1)
P3$_0$/FLD$_{24}$–P3$_7$/FLD$_{31}$	Port P3	Input/output, individual bits	Low-voltage input level High-breakdown voltage P-channel open-drain output with pull-down resistor		FLDC mode register	(2)
P4$_0$/FLD$_{32}$–P4$_7$/FLD$_{39}$	Port P4	Input/output, individual bits	Low-voltage input level High-breakdown voltage P-channel open-drain output		FLDC mode register Port P4 FLD/Port switch register	(2)
P5$_0$/FLD$_{40}$–P5$_7$/FLD$_{47}$	Port P5	Input/output, individual bits	Low-voltage input level High-breakdown voltage P-channel open-drain output		FLDC mode register Port P5 FLD/Port switch register	(2)
P6$_0$/FLD$_{48}$–P6$_3$/FLD$_{51}$	Port P6	Input/output, individual bits	Low-voltage input level High-breakdown voltage P-channel open-drain output		FLDC mode register Port P6 FLD/Port switch register	(2)
P6$_4$/RxD/FLD$_{52}$			Low-voltage input level (port input) CMOS compatible input level (RxD, S$_{CLK21}$, S$_{CLK22}$) CMOS 3-state output	FLD automatic display function Serial I/O2 function I/O	FLDC mode register Serial I/O2 control register UART control register	(3)
P6$_5$/TxD/FLD$_{53}$, P6$_6$/S$_{CLK21}$/FLD$_{54}$						(4)
P6$_7$/$\overline{S_{RDY2}}$/S$_{CLK22}$/FLD$_{55}$						(5)
P7$_0$/INT$_0$, P7$_1$/INT$_1$	Port P7	Input/output, individual bits	CMOS compatible input level CMOS 3-state output	External interrput input	Interrupt edge selection register	(6)
P7$_2$/INT$_2$					Interrupt edge selection register Interrupt interval determination control register	
P7$_3$/INT$_3$/DIM$_{OUT}$				External interrput input Dimmer signal output	Interrupt edge selection register FLD output control register	(7)
P7$_4$/PWM$_1$				PWM output	Timer 56 mode register	(8)
P7$_5$/T1$_{OUT}$				Timer output	Timer 12 mode register	
P7$_6$/T3$_{OUT}$				Timer output	Timer 34 mode register	
P7$_7$/INT$_4$/B$_{UZ01}$				Buzzer output External interrput input	Buzzer output control register Interrupt edge selection register	(9)
P8$_0$/X$_{CIN}$	Port P8	Input/output, individual bits	CMOS compatible input level CMOS 3-state output	Sub-clock generating circuit I/O	CPU mode register	(10)
P8$_1$/X$_{COUT}$						(11)
P8$_2$/CNTR$_1$				External count input	Interrupt edge selection register	(6)
P8$_3$/CNTR$_0$/CNTR$_2$						(12)

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

Pin	Nama	Input/Output	I/O Format	Non-Port Function	Related SFRs	Ref.No.
P9$_0$/S$_{IN3}$/AN$_8$	Port P9	Input/output, individual bits	CMOS compatible input level CMOS 3-state output	Serial I/O3 function I/O A-D conversion input	Serial I/O3 control register AD/DA control register	(6)
P9$_1$/S$_{OUT3}$/AN$_9$, P9$_2$/S$_{CLK3}$/AN$_{10}$						(13)
P9$_3$/$\overline{S_{RDY3}}$/AN$_{11}$						(14)
P9$_4$/RTP$_1$/AN$_{12}$, P9$_5$/RTP$_0$/AN$_{13}$				Real time port output A-D conversion input	Timer X mode register 2 AD/DA control register	(15)
P9$_6$/PWM$_0$/AN$_{14}$				PWM output A-D conversion input	PWM control register AD/DA control register	(16)
P9$_7$/B$_{UZ02}$/AN$_{15}$				Buzzer output A-D conversion input	Buzzer output control register AD/DA control register	(16)
PA$_0$/AN$_0$–PA$_7$/AN$_7$	Port PA	Input/output, individual bits	CMOS compatible input level CMOS 3-state output	A-D conversion input	AD/DA control register	(17)
PB$_0$/S$_{CLK12}$/DA	Port PB	Input/output, individual bits	CMOS compatible input level CMOS 3-state output	Serial I/O1 function I/O D-A conversion output	Serial I/O1 control registers 1, 2 AD/DA control register	(18)
PB$_1$/S$_{RDY1}$				Serial I/O1 function I/O	Serial I/O1 control registers 1, 2	(19)
PB$_2$/S$_{BUSY1}$						(18)
PB$_3$/S$_{STB1}$						(20)
PB$_4$/S$_{CLK11}$ PB$_5$/S$_{OUT1}$						(21)
PB$_6$/S$_{IN1}$						(6)

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.



∗ High-breakdown-voltage P-channel transistor

Notes 1: The dimmer signal sets the Toff timing.

2: A pull-down resistor is not built in to ports P4, P5 and $P6_0$ to $P6_3$.

Fig. 11 Port block diagram (1)

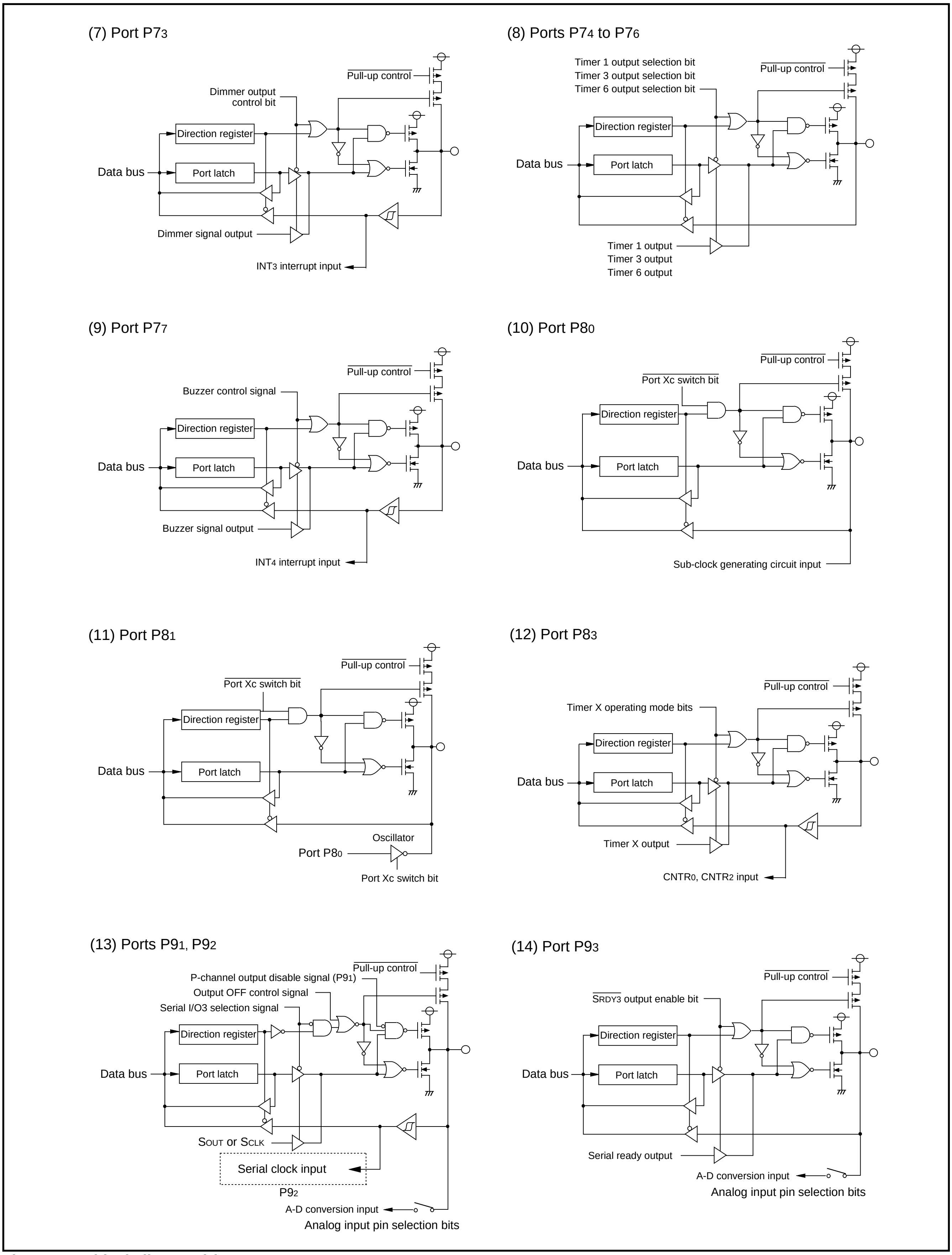


Fig. 12 Port block diagram (2)

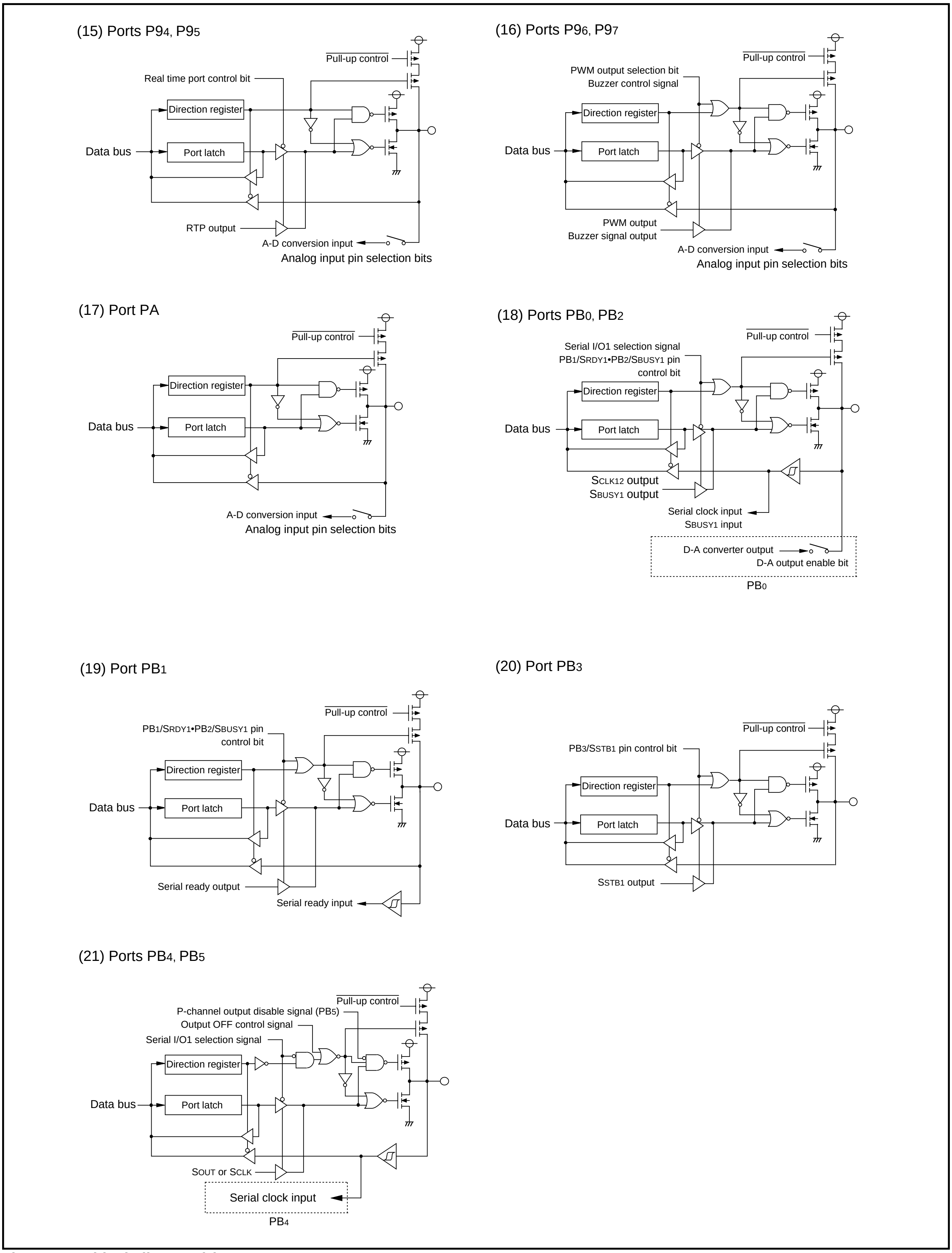


Fig. 13 Port block diagram (3)

INTERRUPTS

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

Interrupt Control

Each interrupt except the BRK instruction interrupt has 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 occur at the same time, the interrupt with highest priority is accepted first.

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.

Interrupt Source Selection

Any of the following interrupt sources can be selected by the interrupt source switch register (address 0039_{16}).

1. INT_1 or Serial I/O3
2. INT_3 or Serial I/O2 transmit
3. INT_4 or A-D conversion

■Note

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

•When switching external interrupt active edge
  Related register: Interrupt edge selection register (address $3A_{16}$)
•When switching interrupt sources of an interrupt vector address where two or more interrupt sources are allocated
  Related register: Interrupt source switch register (address 39_{16})

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

① Set the corresponding interrupt enable bit to “0” (disabled).
② Set the interrupt edge select bit (active edge switch bit) or the interrupt (source) select/switch bit.
③ Set the corresponding interrupt request bit to “0” after 1 or more instructions have been executed.
④ Set the corresponding interrupt enable bit to “1” (enabled).

Table 8 Interrupt vector addresses and priority

Interrupt Source	Priority	Vector Addresses **(Note 1)** High	Vector Addresses **(Note 1)** Low	Interrupt Request Generating Conditions	Remarks
Reset **(Note 2)**	1	$FFFD_{16}$	$FFFC_{16}$	At reset	Non-maskable
INT_0	2	$FFFB_{16}$	$FFFA_{16}$	At detection of either rising or falling edge of INT_0 input	External interrupt (active edge selectable)
INT_1	3	$FFF9_{16}$	$FFF8_{16}$	At detection of either rising or falling edge of INT_1 input	External interrupt (active edge selectable) Valid when INT_1 interrupt is selected
Serial I/O3				At completion of data transfer	Valid when serial I/O3 is selected
INT_2	4	$FFF7_{16}$	$FFF6_{16}$	At detection of either rising or falling edge of INT_2 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	$FFF5_{16}$	$FFF4_{16}$	At completion of data transfer	Valid when serial I/O ordinary mode is selected
Serial I/O automatic transfer				At completion of the last data transfer	Valid when serial I/O automatic transfer mode is selected
Timer X	6	$FFF3_{16}$	$FFF2_{16}$	At timer X underflow	
Timer 1	7	$FFF1_{16}$	$FFF0_{16}$	At timer 1 underflow	
Timer 2	8	$FFEF_{16}$	$FFEE_{16}$	At timer 2 underflow	STP release timer underflow
Timer 3	9	$FFED_{16}$	$FFEC_{16}$	At timer 3 underflow	
Timer 4	10	$FFEB_{16}$	$FFEA_{16}$	At timer 4 underflow	
Timer 5	11	$FFE9_{16}$	$FFE8_{16}$	At timer 5 underflow	
Timer 6	12	$FFE7_{16}$	$FFE6_{16}$	At timer 6 underflow	
Serial I/O2 receive	13	$FFE5_{16}$	$FFE4_{16}$	At completion of serial I/O2 data receive	
INT_3	14	$FFE3_{16}$	$FFE2_{16}$	At detection of either rising or falling edge of INT_3 input	External interrupt (active edge selectable) Valid when INT_3 interrupt is selected
Serial I/O2 transmit				At completion of serial I/O2 data transmit	
INT_4	15	$FFE1_{16}$	$FFE0_{16}$	At detection of either rising or falling edge of INT_4 input	External interrupt (active edge selectable) Valid when INT_4 interrupt is selected
A-D conversion				At completion of A-D conversion	Valid when A-D conversion is selected
FLD blanking	16	$FFDF_{16}$	$FFDE_{16}$	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	$FFDD_{16}$	$FFDC_{16}$	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.

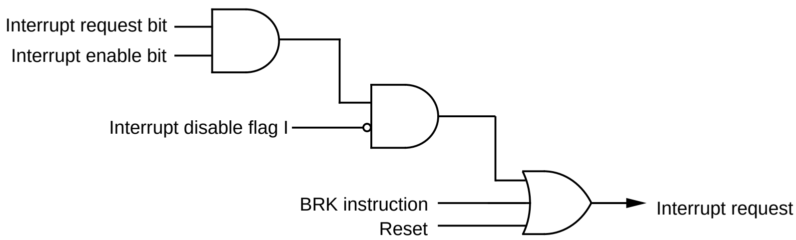


Fig. 14 Interrupt control

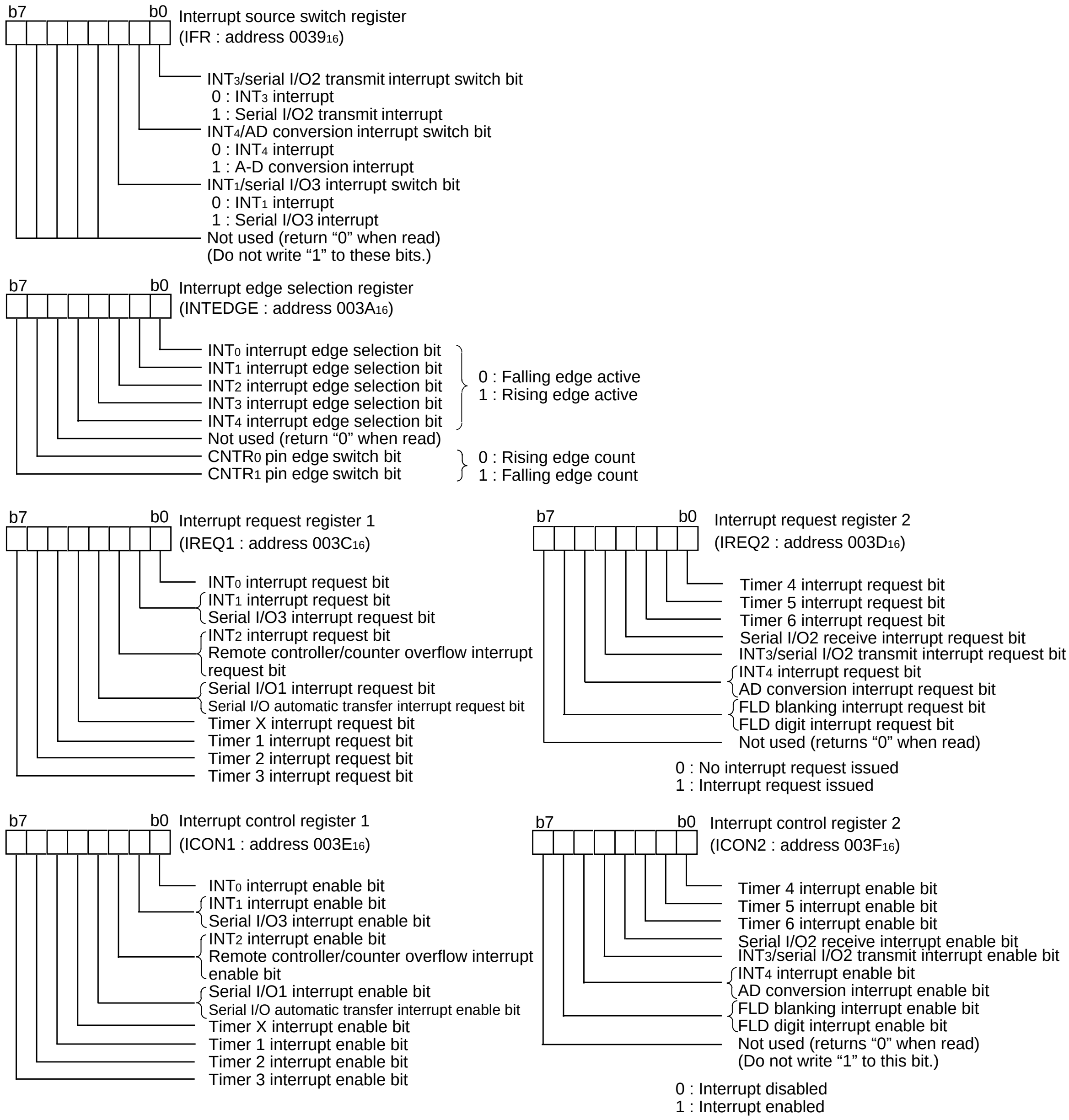


Fig. 15 Structure of interrupt related registers

TIMERS
8-Bit Timer

The 38B7 group has six built-in 8-bit 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, the 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 $P7_5/T1_{OUT}$ 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 $P7_6/T3_{OUT}$ pin. The active edge of the external clock $CNTR_1$ can be switched with the bit 7 of the interrupt edge selection register.

●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 $P7_4/PWM_1$ pin.

●Timer 6 PWM_1 Mode

Timer 6 can output a PWM rectangular waveform with "H" duty cycle n/(n+m) from the $P7_4/PWM_1$ 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.

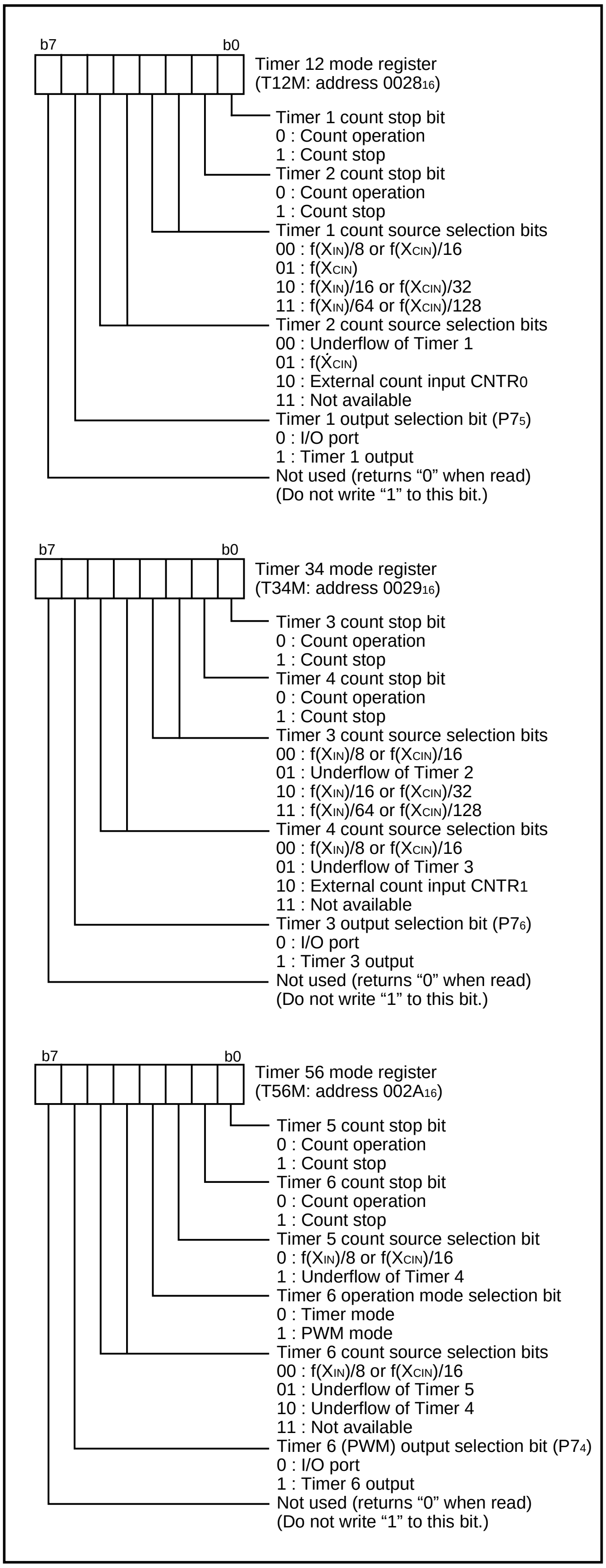


Fig. 16 Structure of timer related registers

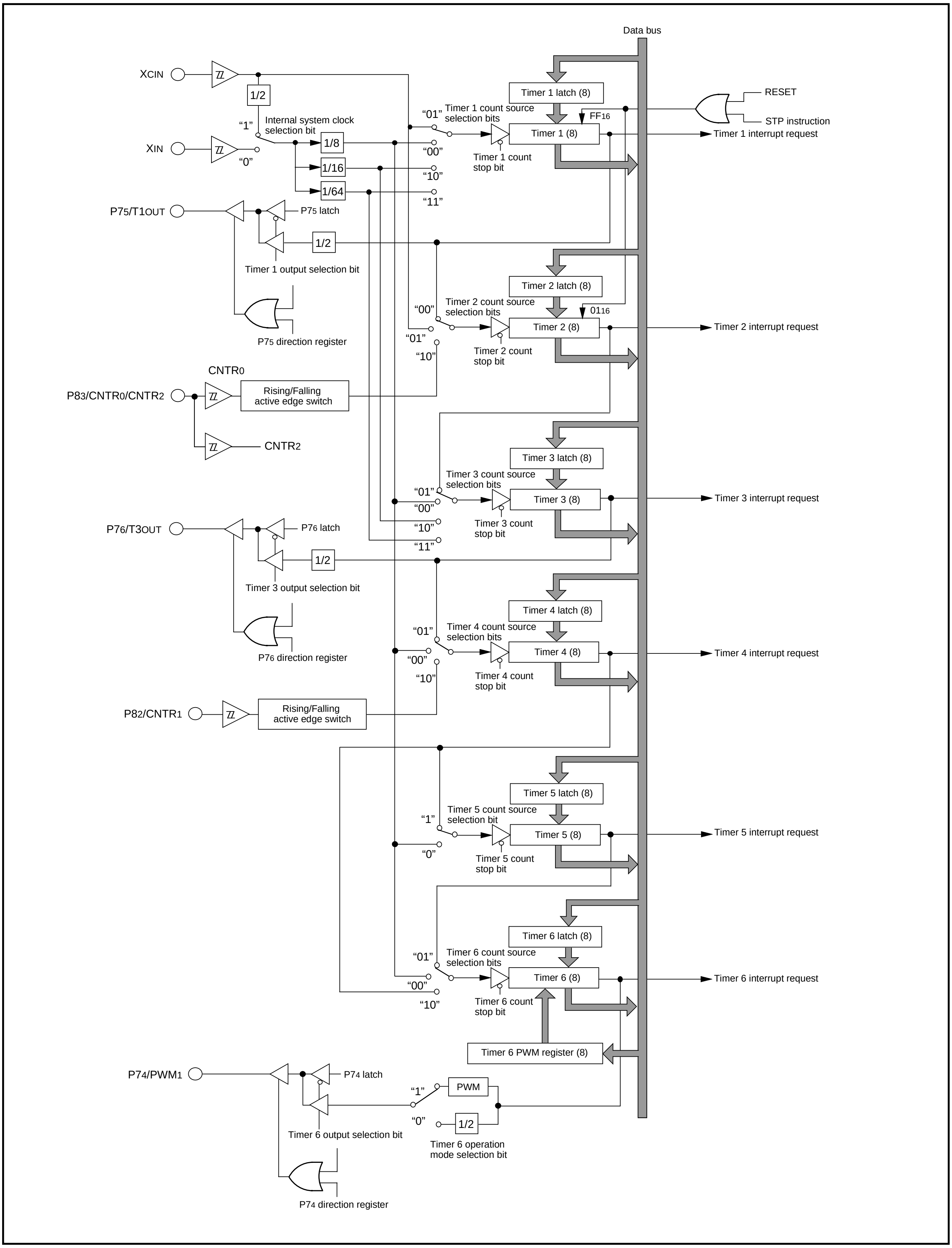
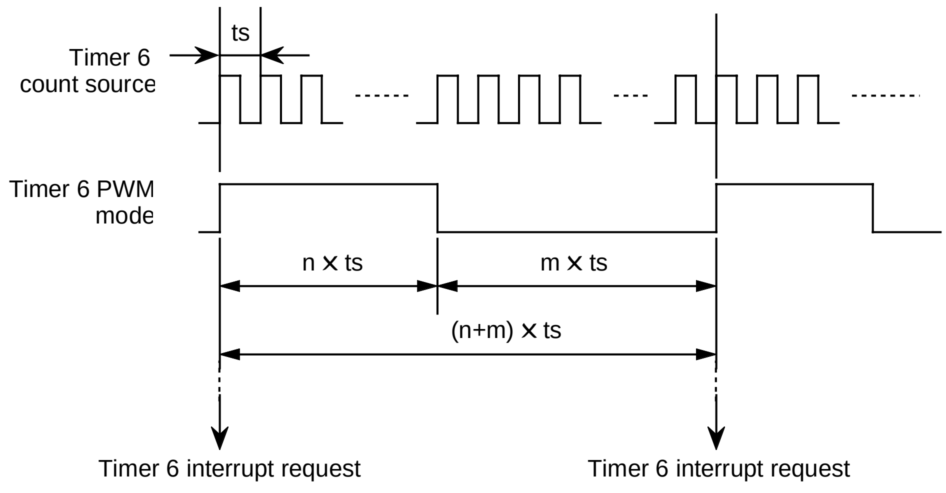


Fig. 17 Block diagram of timer



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 PWM_1 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 for 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 $P9_4$ and $P9_5$ each time the timer X underflows. (However, if the real time port control bit is changed from “0” to “1”, data are output independent of 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.

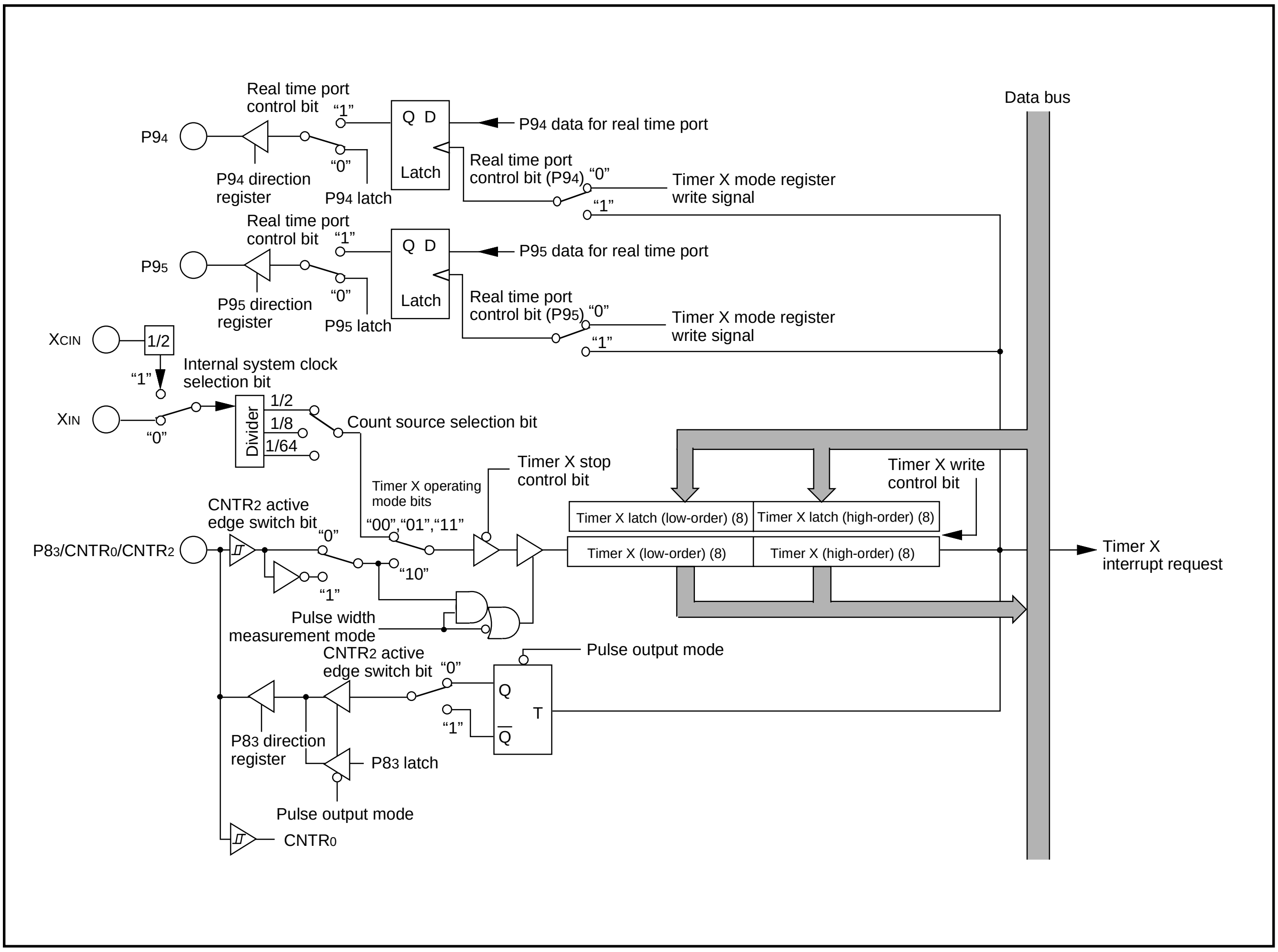


Fig. 19 Block diagram of timer X

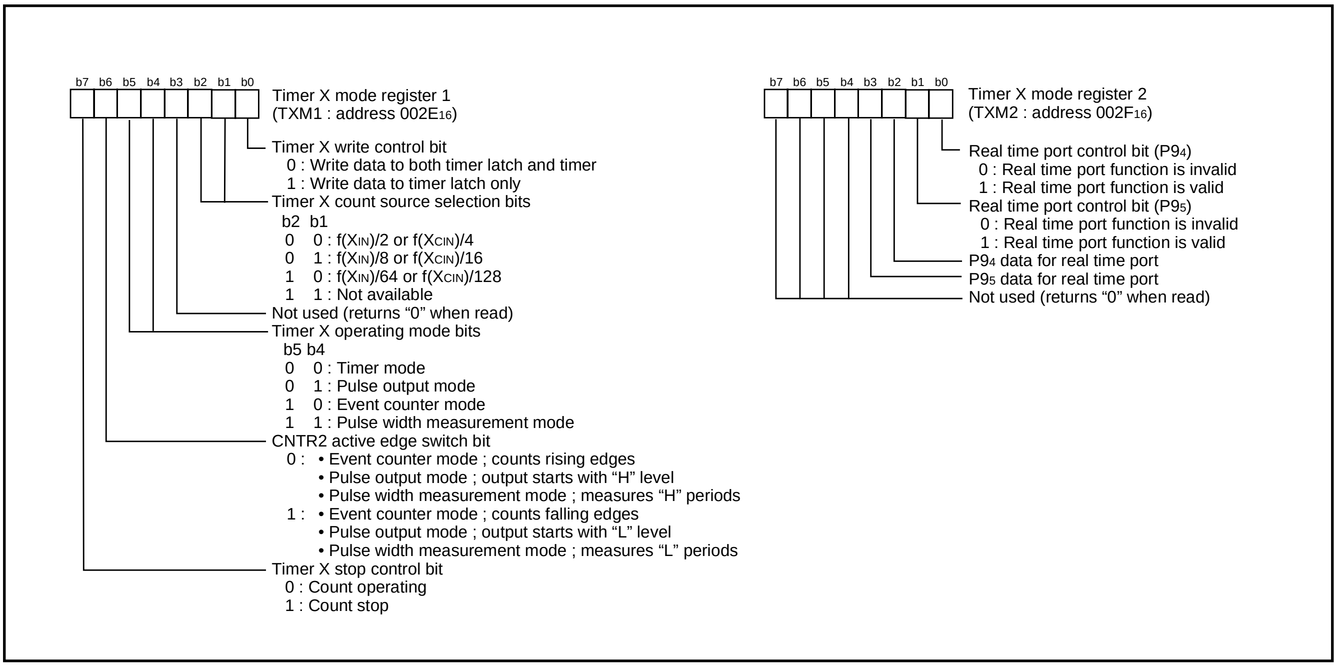


Fig. 20 Structure of timer X related registers

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 $0FFF_{16}$).

The PB1/SRDY1, PB2/SBUSY1, and PB3/SSTB1 pins each have a handshake I/O signal function and can select either "H" active or "L" active for active logic.

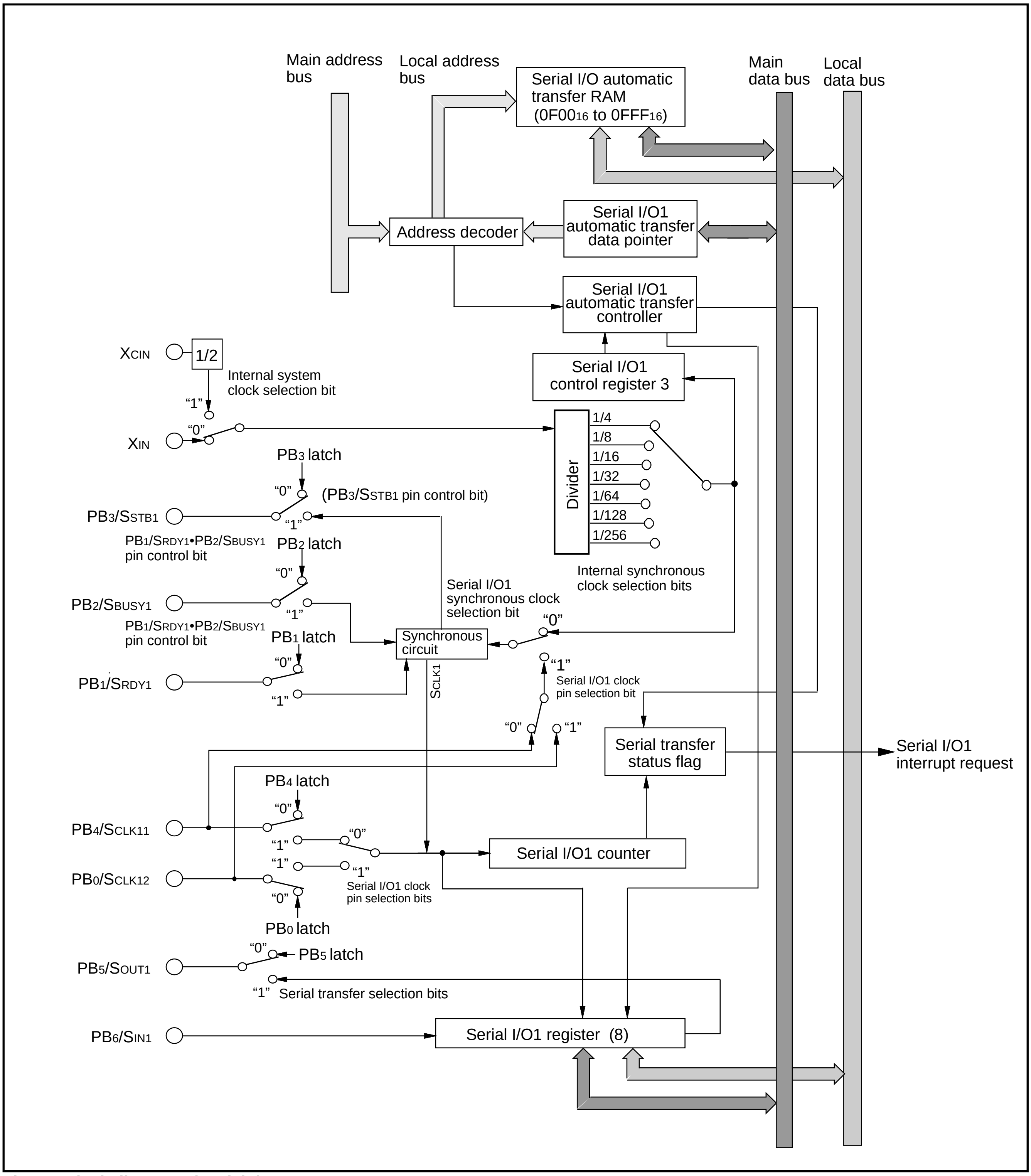


Fig. 21 Block diagram of serial I/O1

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O1 control register 1
(SIO1CON1 (SC11) : address 0019_{16})

- b1, b0: Serial transfer selection bits
  b1 b0
  0 0 : Serial I/O disabled (pins PB_0 to PB_6 are I/O ports)
  0 1 : 8-bit serial I/O
  1 0 : Not available
  1 1 : Automatic transfer serial I/O (8-bits)
- b3, b2: Serial I/O1 synchronous clock selection bits (PB_3/S_{STB1} pin control bit)
  b3 b2
  0 0 : Internal synchronous clock (PB_3 pin is an I/O port.)
  0 1 : External synchronous clock (PB_3 pin is an I/O port.)
  1 0 : Internal synchronous clock (PB_3 pin is an $\overline{S_{STB1}}$ output.)
  1 1 : Internal synchronous clock (PB_3 pin is an S_{STB1} output.)
- b4: Serial I/O initialization bit
  0: Serial I/O initialization
  1: Serial I/O enabled
- b5: Transfer mode selection bit
  0: Full duplex (transmit and receive) mode (PB_6 pin is an S_{IN1} input.)
  1: Transmit-only mode (PB_6 pin is an I/O port.)
- b6: Transfer direction selection bit
  0: LSB first
  1: MSB first
- b7: Serial I/O1 clock pin selection bit
  0: S_{CLK11} (PB_0/S_{CLK12} pin is an I/O port.)
  1: S_{CLK12} (PB_4/S_{CLK11} pin is an I/O port.)

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O1 control register 2
(SIO1CON2 (SC12): address $001A_{16}$)

- b3–b0: PB_1/S_{RDY1} • PB_2/S_{BUSY1} pin control bits
  b3b2b1b0
  0 0 0 0: Pins PB_1 and PB_2 are I/O ports
  0 0 0 1: Not used
  0 0 1 0: PB_1 pin is an $\overline{S_{RDY1}}$ output, PB_2 pin is an I/O port.
  0 0 1 1: PB_1 pin is an S_{RDY1} output, PB_2 pin is an I/O port.
  0 1 0 0: PB_1 pin is an I/O port, PB_2 pin is an $\overline{S_{BUSY1}}$ input.
  0 1 0 1: PB_1 pin is an I/O port, PB_2 pin is an S_{BUSY1} input.
  0 1 1 0: PB_1 pin is an I/O port, PB_2 pin is an $\overline{S_{BUSY1}}$ output.
  0 1 1 1: PB_1 pin is an I/O port, PB_2 pin is an S_{BUSY1} output.
  1 0 0 0: PB_1 pin is an $\overline{S_{RDY1}}$ input, PB_2 pin is an $\overline{S_{BUSY1}}$ output.
  1 0 0 1: PB_1 pin is an $\overline{S_{RDY1}}$ input, PB_2 pin is an S_{BUSY1} output.
  1 0 1 0: PB_1 pin is an S_{RDY1} input, PB_2 pin is an $\overline{S_{BUSY1}}$ output.
  1 0 1 1: PB_1 pin is an S_{RDY1} input, PB_2 pin is an S_{BUSY1} output.
  1 1 0 0: PB_1 pin is an $\overline{S_{RDY1}}$ output, PB_2 pin is an $\overline{S_{BUSY1}}$ input.
  1 1 0 1: PB_1 pin is an $\overline{S_{RDY1}}$ output, PB_2 pin is an S_{BUSY1} input.
  1 1 1 0: PB_1 pin is an S_{RDY1} output, PB_2 pin is an $\overline{S_{BUSY1}}$ input.
  1 1 1 1: PB_1 pin is an S_{RDY1} output, PB_2 pin is an S_{BUSY1} input.
- b4: S_{BUSY1} output • S_{STB1} output function selection bit
  (Valid in automatic transfer mode)
  0: Functions as each 1-byte signal
  1: Functions as signal for all transfer data
- b5: Serial transfer status flag
  0: Serial transfer completion
  1: Serial transferring
- b6: S_{OUT1} pin control bit (at no-transfer serial data)
  0: Output active
  1: Output high-impedance
- b7: PB_5/S_{OUT1} P-channel output disable bit
  0: CMOS 3-state (P-channel output is valid.)
  1: N-channel open-drain (P-channel output is invalid.)

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 PB1/SRDY1, PB2/SBUSY1, and PB3/SSTB1 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 PB1/SRDY1 • PB2/SBUSY1 pin control bits (b0 to b3 of address $001A_{16}$) of serial I/O1 control register 2.
For the SOUT1 being used as an output pin, either CMOS output or N-channel open-drain output is selected by the PB5/SOUT1 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 SOUT1 pin state at serial non-transfer by the SOUT1 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 SOUT1 pin into a high-impedance state:
When the SCLK1 input is "H" after completion of transfer, set the SOUT1 pin control bit to "1".
When the SCLK1 input goes to "L" after the start of the next serial transfer, the SOUT1 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".

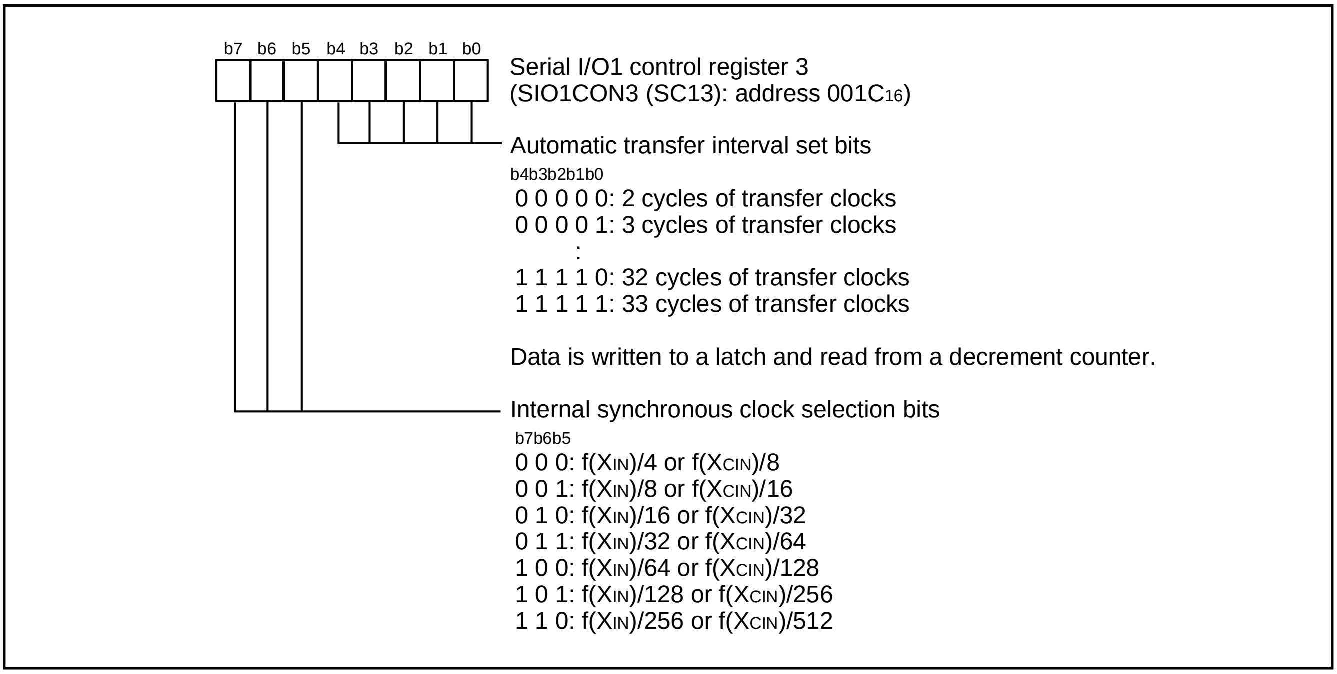


Fig. 23 Structure of serial I/O1 control register 3

(2) 8-bit serial I/O mode

Address $001B_{16}$ 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 $001B_{16}$).

The serial transfer status flag (b5 of address $001A_{16}$) 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 that the function of address $001B_{16}$ is used as a transfer counter (1-byte unit).

When performing serial transfer through the serial I/O automatic transfer RAM (addresses $0F00_{16}$ to $0FFF_{16}$), it is necessary to set the serial I/O1 automatic transfer data pointer (address 0018_{16}) 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 $001C_{16}$) 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 $001A_{16}$) 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 $001B_{16}$).

When the external synchronous clock is selected, write the value of “number of transfer bytes – 1” into the transfer counter and keep 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, keep 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 $001A_{16}$) 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 0018_{16}) and the automatic transfer interval set bits (b0 to b4 of address $001C_{16}$) 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.

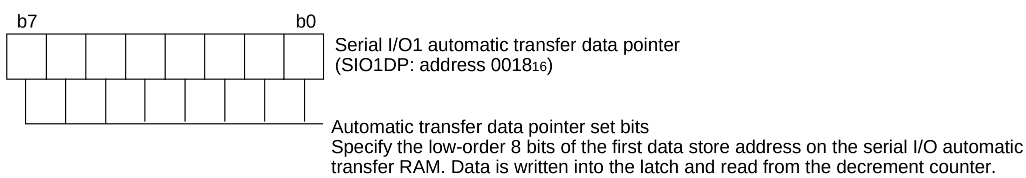


Fig. 24 Structure of serial I/O1 automatic transfer data pointer

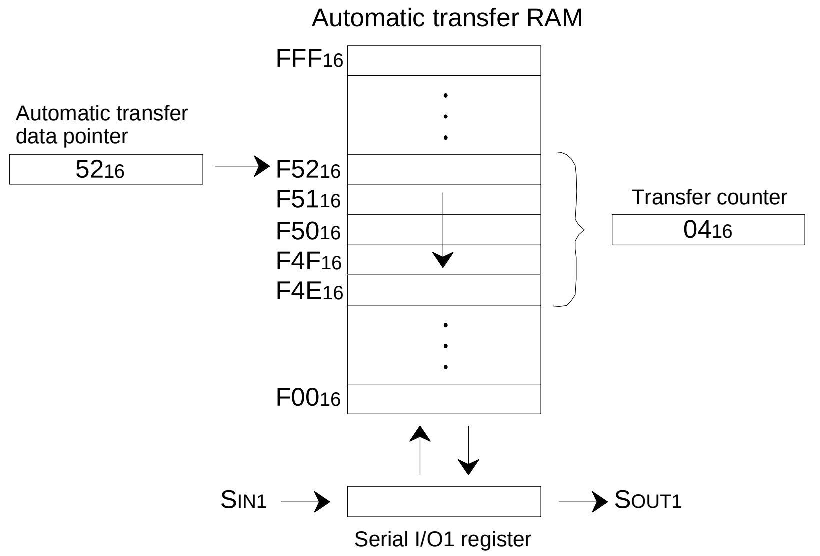


Fig. 25 Automatic transfer serial I/O operation

(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 001A16) of serial I/O1 control register 2.

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.
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.

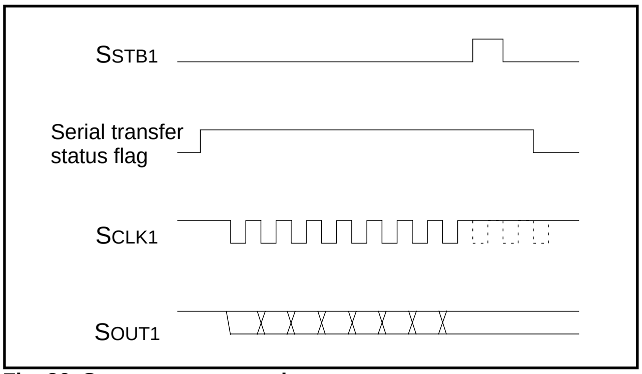


Fig. 26 SSTB1 output operation

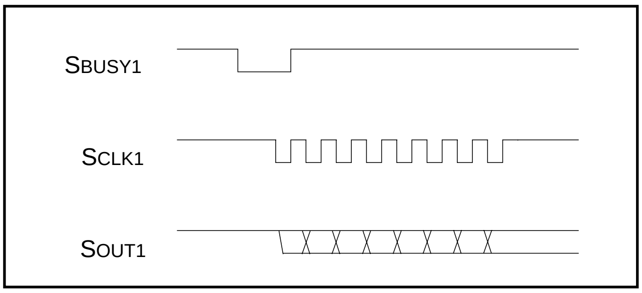


Fig. 27 SBUSY1 input operation (internal synchronous clock)

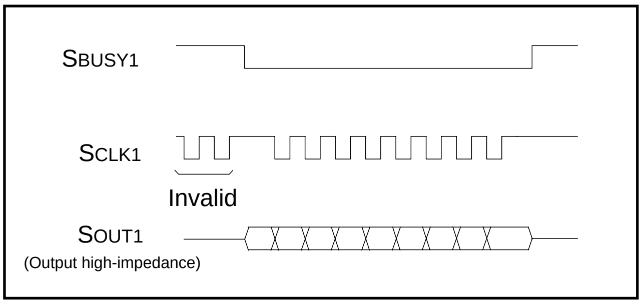


Fig. 28 SBUSY1 input operation (external synchronous clock)

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.

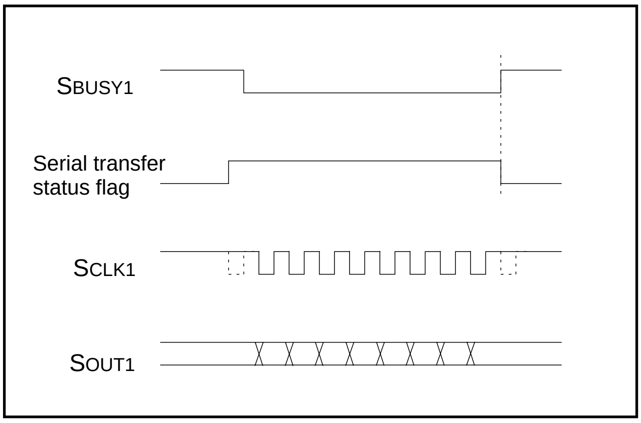


Fig. 29 SBUSY1 output operation (internal synchronous clock, 8-bit serial I/O)

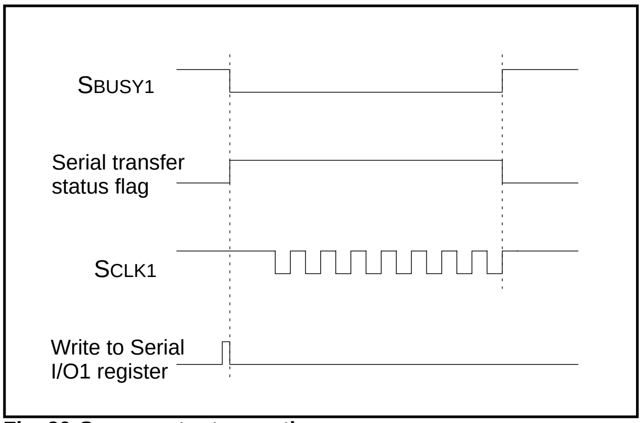


Fig. 30 SBUSY1 output operation (external synchronous clock, 8-bit serial I/O)

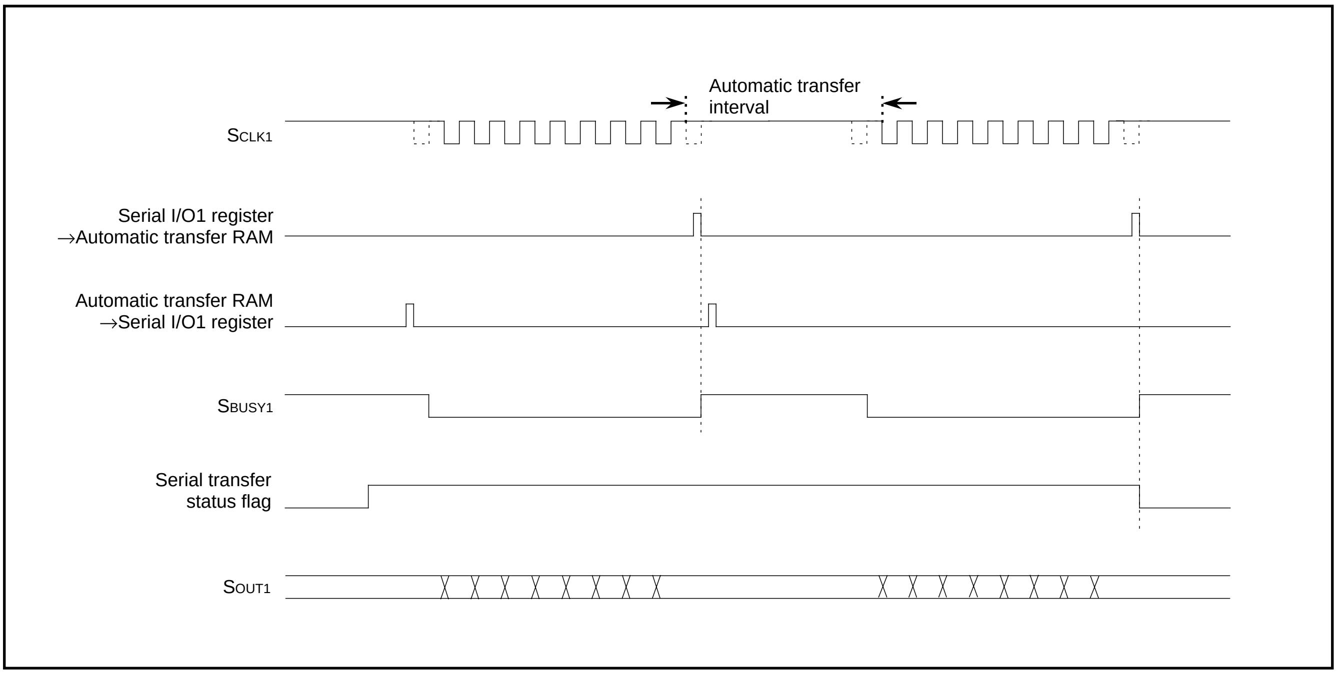


Fig. 31 SBUSY1 output operation in automatic transfer serial I/O mode (internal synchronous clock, SBUSY1 output function outputs each 1-byte)

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 001B16) 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.

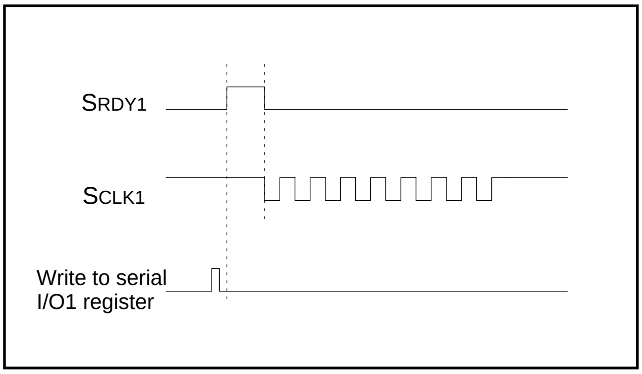


Fig. 32 SRDY1 output operation

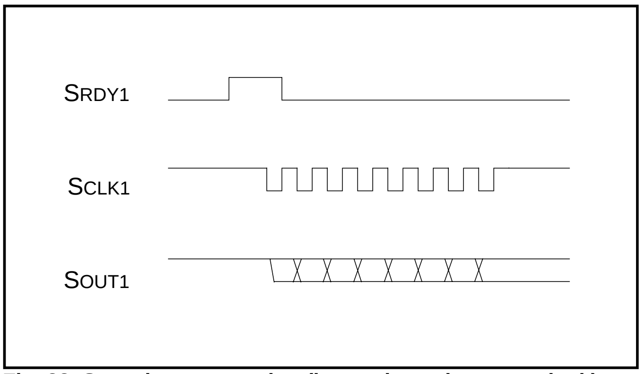


Fig. 33 SRDY1 input operation (internal synchronous clock)

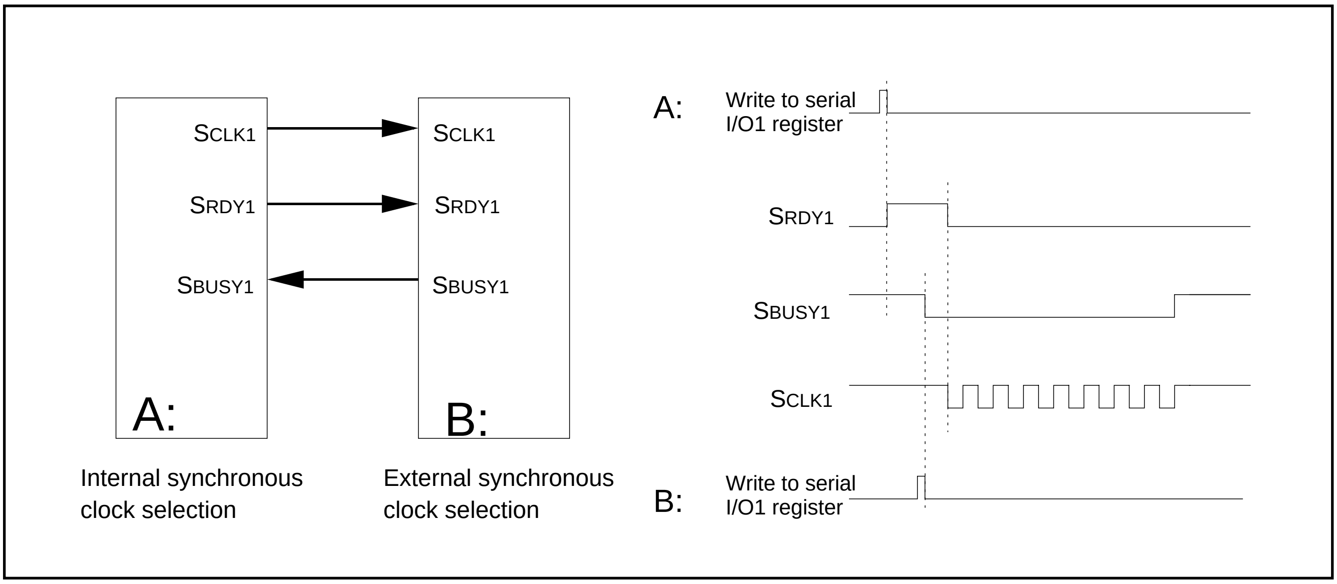


Fig. 34 Handshake operation at serial I/O1 mutual connecting (1)

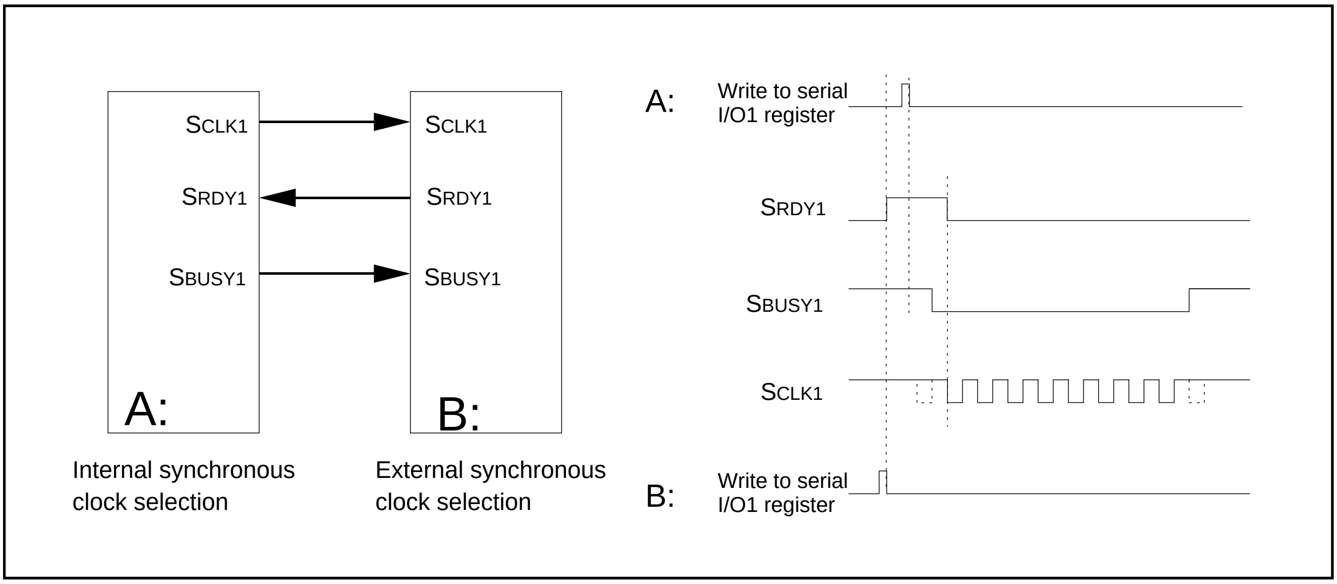


Fig. 35 Handshake operation at serial I/O1 mutual connecting (2)

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 $001D_{16}$) 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 $001F_{16}$).

When $P6_7$ (S_{CLK22}) is selected as a clock I/O pin, $\overline{S_{RDY2}}$ output function is invalid, and $P6_6$ (S_{CLK21}) is used as an I/O port.

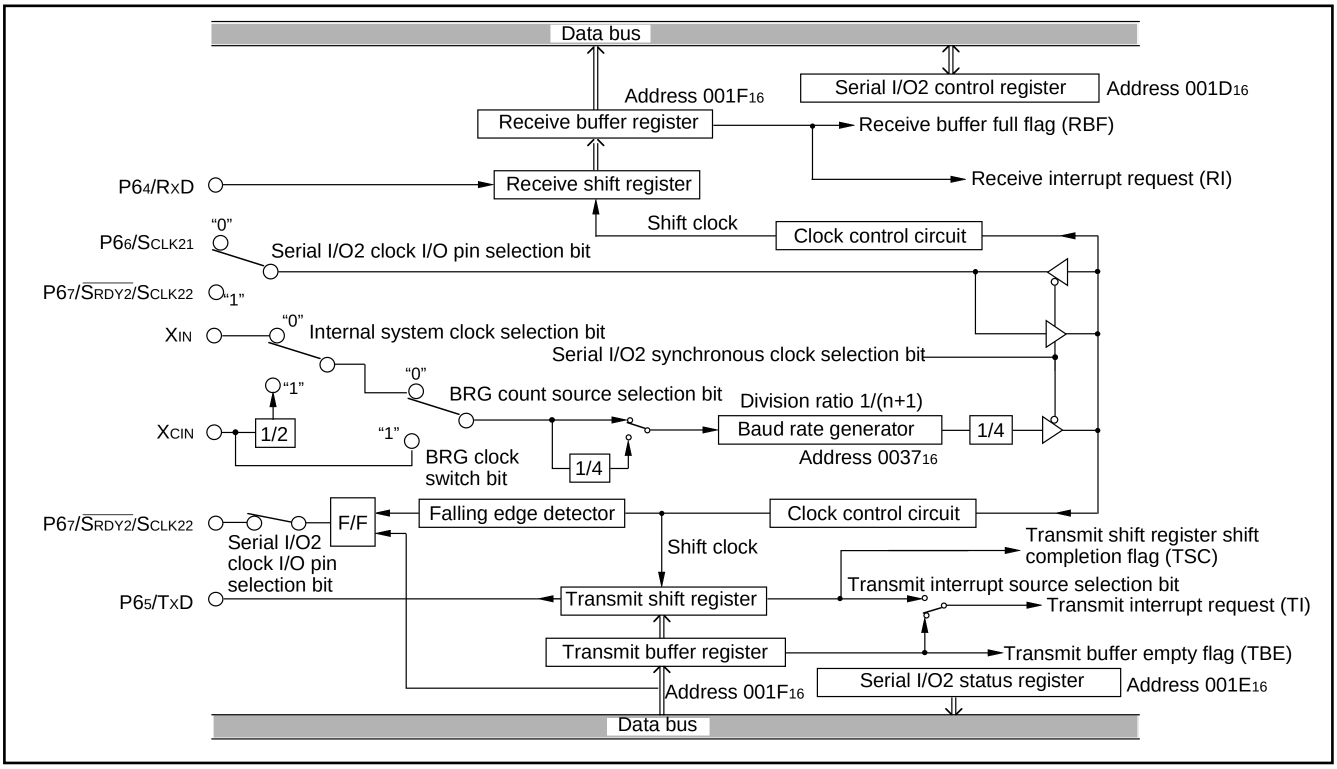
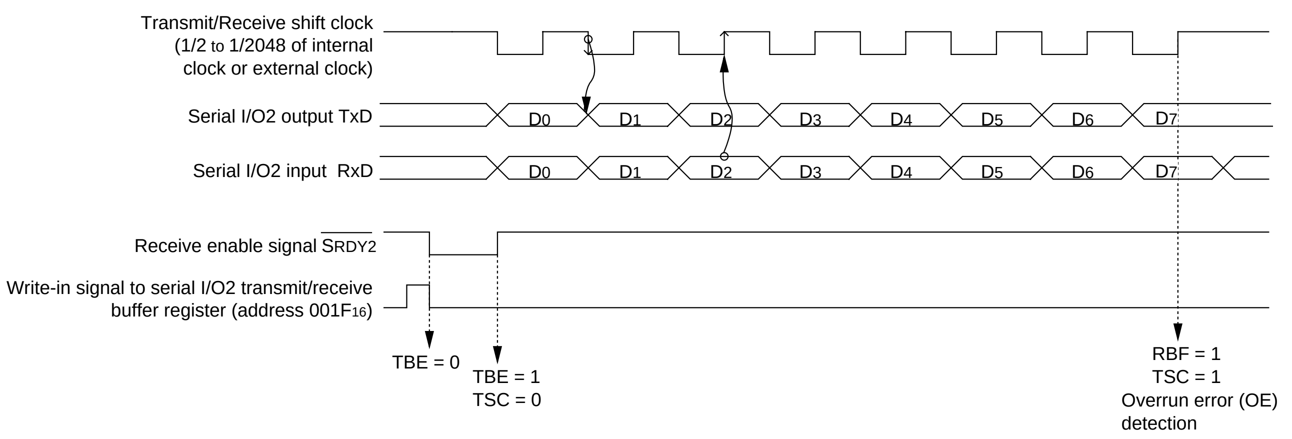


Fig. 36 Block diagram of clock synchronous serial I/O2



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 $001D_{16}$) 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.

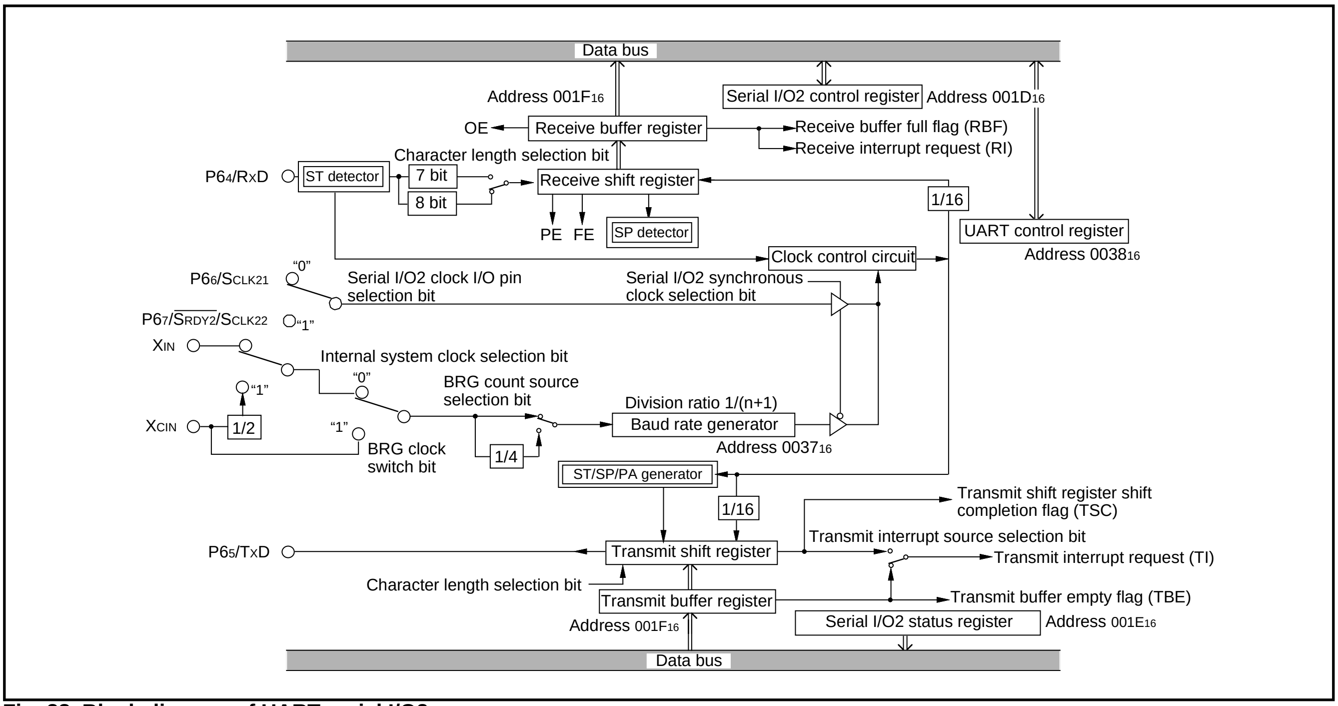


Fig. 38 Block diagram of UART serial I/O2

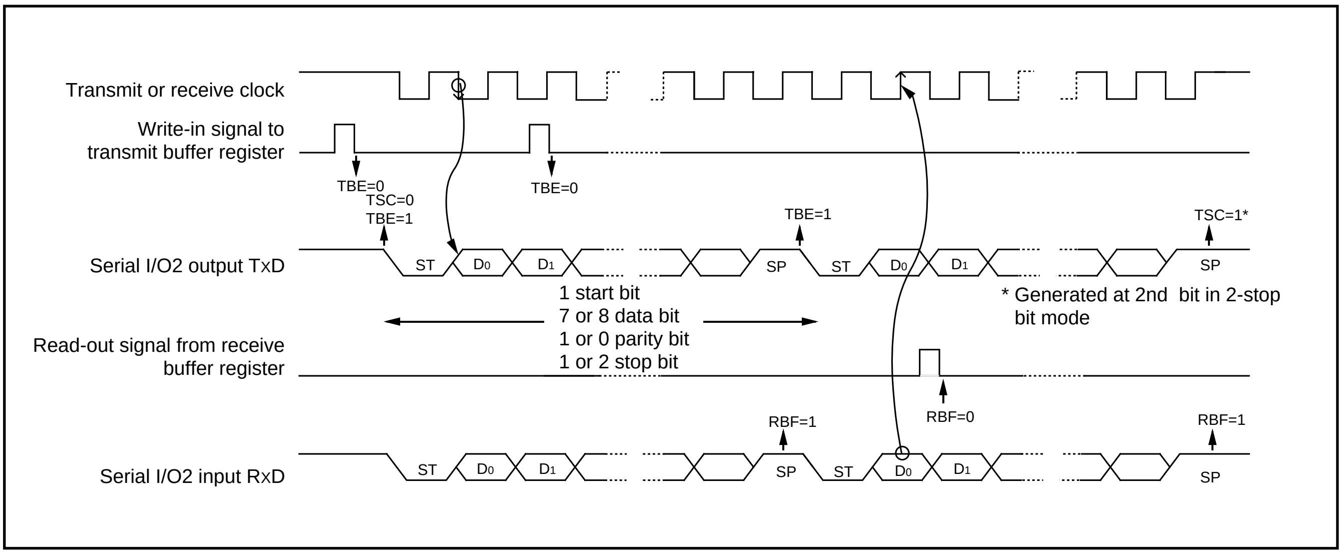


Fig. 39 Operation of UART serial I/O2 function

[Serial I/O2 Control Register] SIO2CON ($001D_{16}$)

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

[UART Control Register] UARTCON (0038_{16})

This is a 7 bit register containing four control bits, of which four bits are valid when UART is selected, and of which three bits are always valid.
Data format of serial data receive/transfer and the output structure of the $P6_5$/TxD pin and others are set by this register.

[Serial I/O2 Status Register] SIO2STS ($001E_{16}$)

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 ($001F_{16}$)

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 (0037_{16})

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.

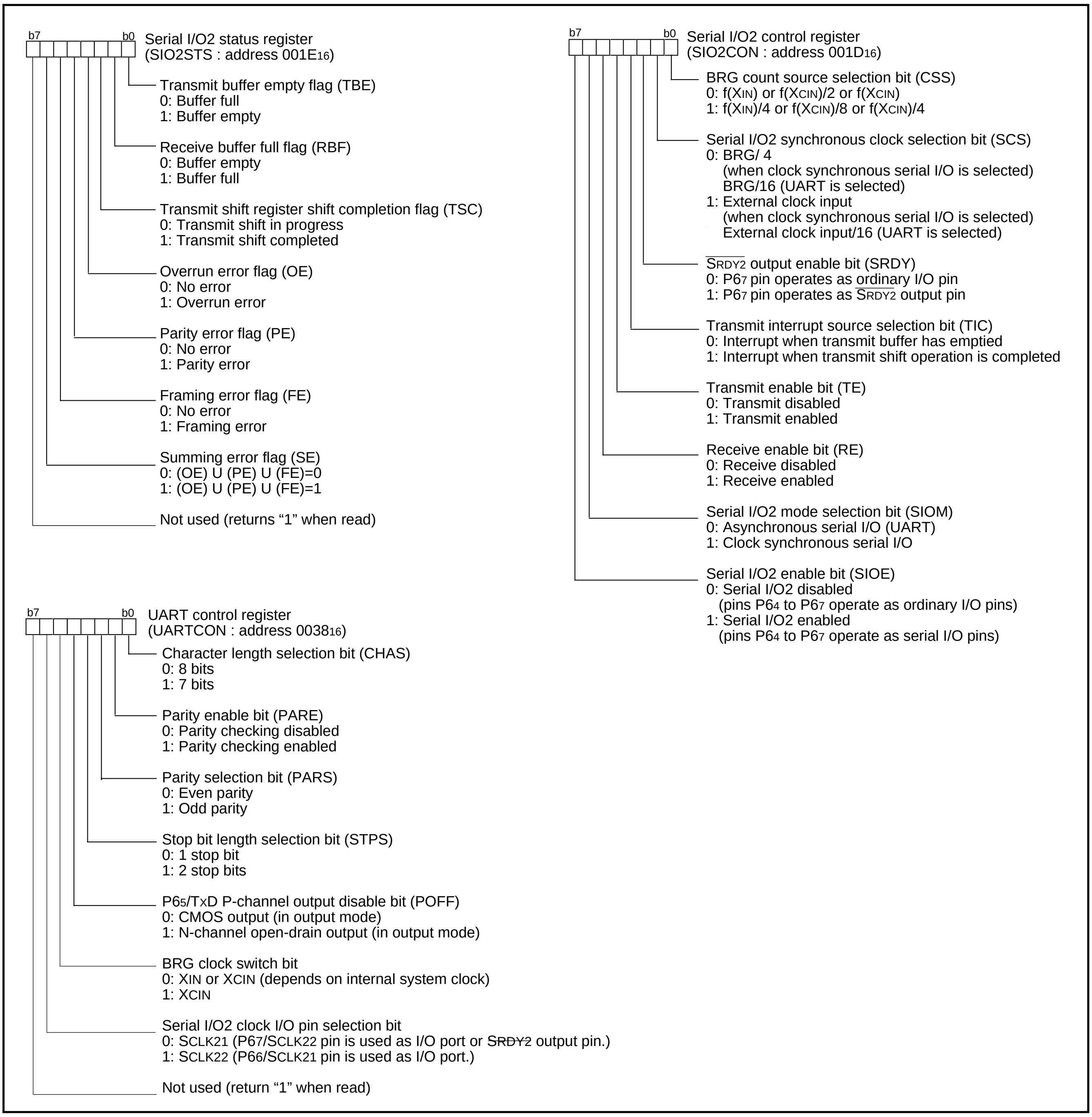


Fig. 40 Structure of serial I/O2 related register

■**Notes**

When setting the transmit enable bit to “1”, the serial I/O2 transmit interrupt request bit is automatically set to “1”. When not requiring the interrupt occurrence synchronized with the transmission enabled, take the following sequence.

① Set the serial I/O1 transmit interrupt enable bit to “0” (disabled).
② Set the transmit enable bit to “1”.
③ Set the serial I/O1 transmit interrupt request bit to “0” after 1 or more instructions have been executed.
④ Set the serial I/O1 tranmit interrupt enable bit to “1” (enabled).

Serial I/O3

The serial I/O3 function can be used only for 8-bit clock synchronous serial I/O.

All serial I/O pins are shared with port P9, which can be set with the serial I/O3 control register (address $0EEC_{16}$).

[Serial I/O3 Control Register (SIO3CON)] $0EEC_{16}$

The serial I/O3 control register contains eight bits which control various serial I/O functions.

● **Serial I/O3 Operation**

Either the internal clock or external clock can be selected as synchronous clock for serial I/O3 transfer.

The internal clock can use a built-in dedicated divider where 6 different clocks are selected. In the case of the internal clock used, transfer is started by a write signal to the serial I/O3 register (address $0EED_{16}$). When 8-bit data has been transferred, the S_{OUT3} pin goes to high impedance state.

In the case of the external clock used, the clock must be externally controlled. It is because the contents of serial I/O3 register is kept shifted while the clock is being input. Additionally, the function to put the S_{OUT3} pin high impedance state at completion of data transfer is not available.

The serial I/O3 interrupt request bit is set at completion of 8-bit data transfer, regardless of use of the internal clock or external clock.

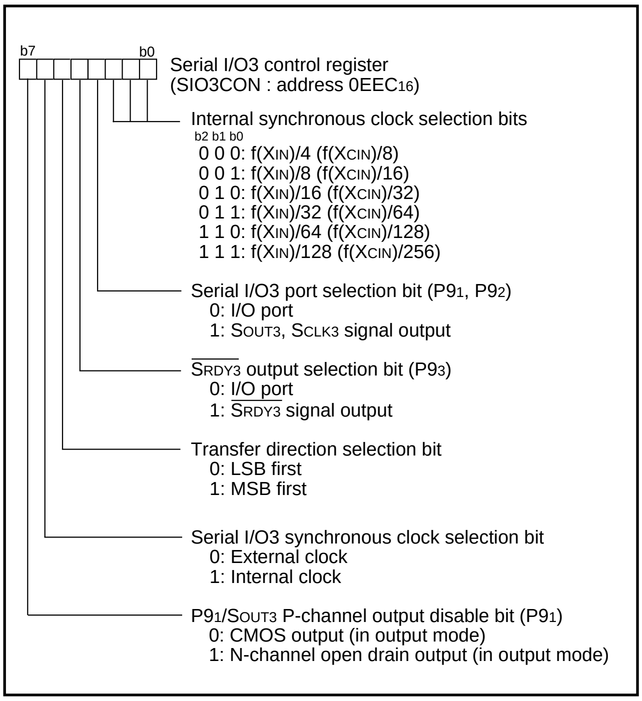


Fig. 42 Structure of serial I/O3 control register

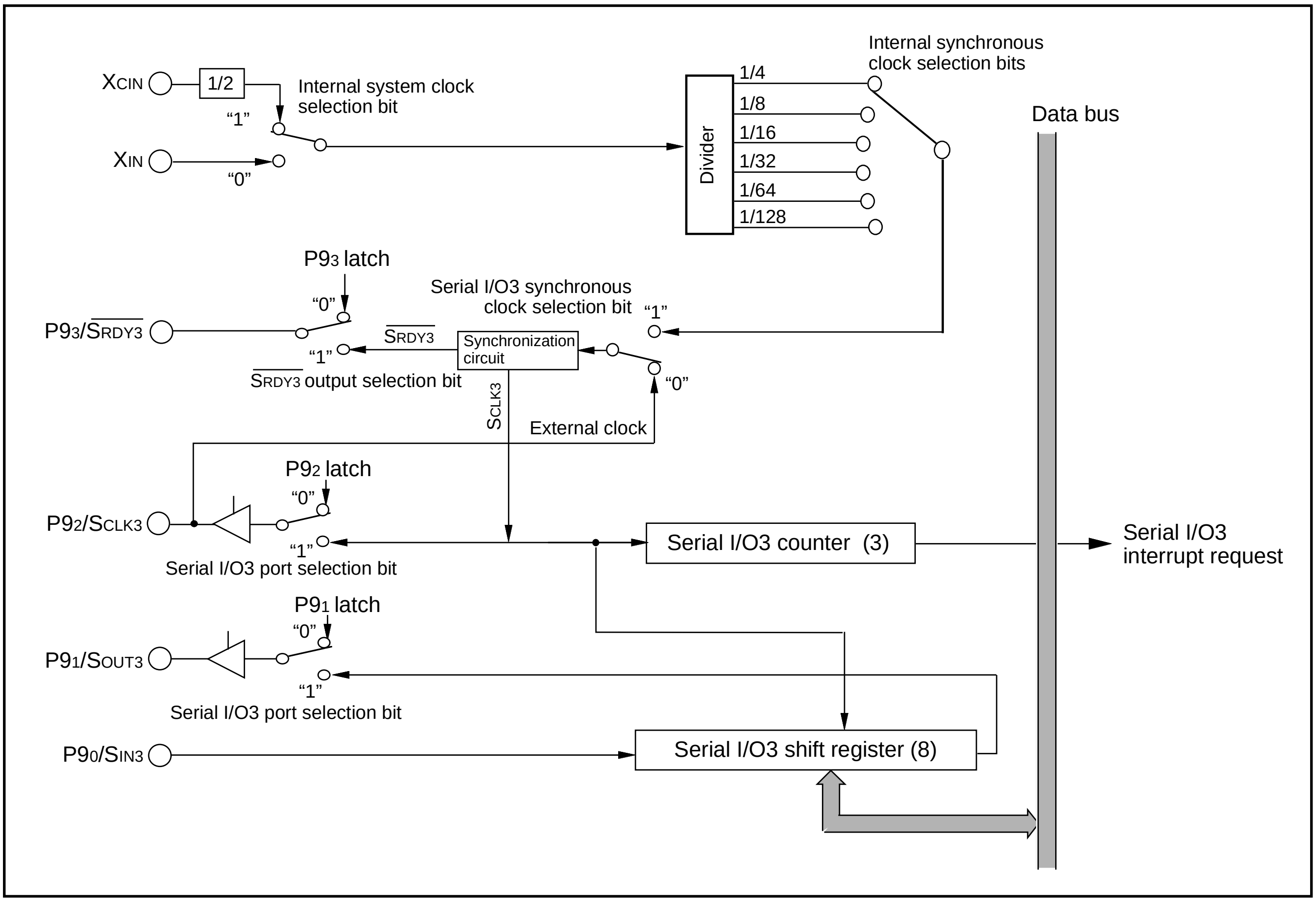


Fig. 41 Block diagram of serial I/O3

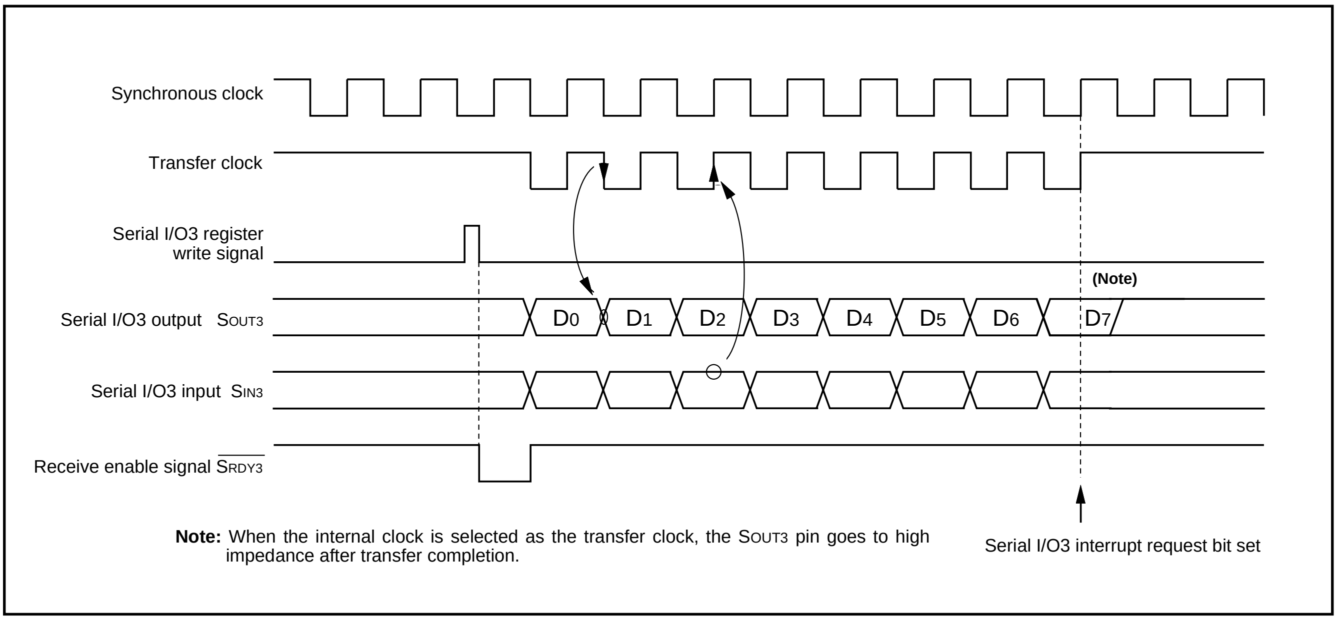


Fig. 43 Timing of serial I/O3 (LSB first)

FLD CONTROLLER

The M38B7 group has fluorescent display (FLD) drive and control circuits.

Table 9 shows the FLD controller specifications.

Table 9 FLD controller specifications

Item		Specifications
FLD controller port	High-breakdown-voltage output port	• 52 pins (20 pins can be switched to general-purpose ports)
	CMOS port	• 4 pins (all 4 pins can be switched to general-purpose ports) (A driver IC must be installed externally)
Display pixel number		• Used FLD output 28 segment × 28 digit (segment number + digit number ≤ 56) • Used digit output 40 segment × 16 digit (segment number ≤ 40, digit number ≤ 16) • Connected to M35501 56 segment × (connected number of M35501) digit (segment number ≤ 56, digit number ≤ number of M35501 × 16) • Used $P6_4$ to $P6_7$ expansion 52 segment × 16 digit (segment number ≤ 52, digit number ≤ 16)
Period		• 4.0 µs to 1024 µs (count source X_{IN}/16, 4 MHz) • 16.0 µs to 4096 µs (count source X_{IN}/64, 4 MHz)
Dimmer time		• 4.0 µs to 1024 µs (count source X_{IN}/16, 4 MHz) • 16.0 µs to 4096 µs (count source X_{IN}/64, 4 MHz)
Interrupt		• Digit interrupt • FLD blanking interrupt
Key-scan		• Key-scan using digit • Key-scan using segment
Expanded function		• Digit pulse output function This function automatically outputs digit pulses. • M35501 connection function The number of digits can be increased easily by using the output of DIM_{OUT}($P7_3$) as CLK for the M35501. • Toff section generating/nothing function This function does not generate Toff1 section when the connected outputs are the same. • Gradation display function This function allows each segment to be set for dark or bright display. • $P6_4$ to $P6_7$ expansion function This function provides 16 lines of digit outputs from four ports by attaching the decoder converting 4-bit data to 16-bit data.

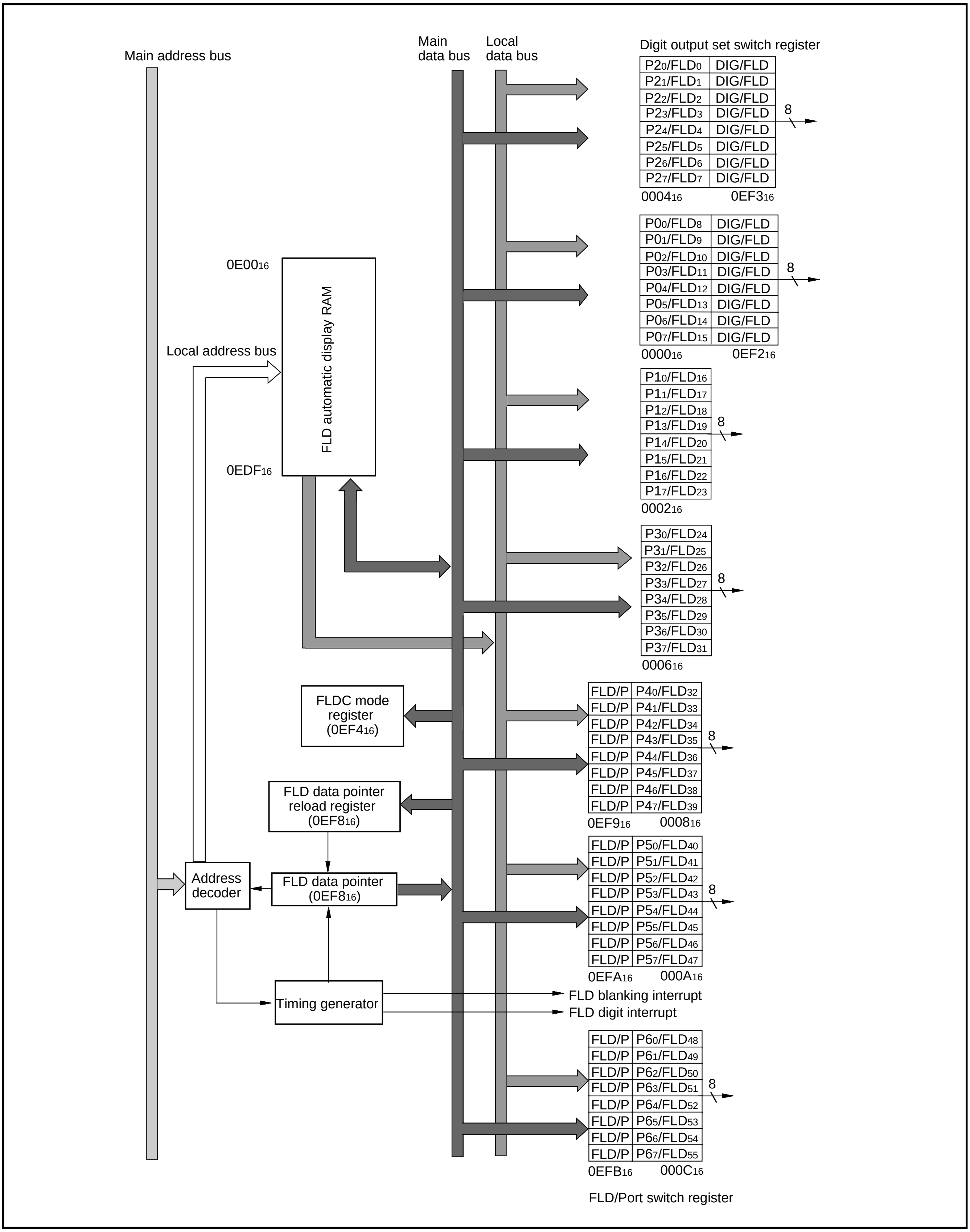


Fig. 44 Block diagram of FLD control circuit

b7 b6 b5 b4 b3 b2 b1 b0

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

- b0: Automatic display control bit
  - 0 : General-purpose mode
  - 1 : Automatic display mode
- b1: Display start bit
  - 0 : Stop display
  - 1 : Display(start to display by switching “0” to “1”)
- b3, b2: Tscan control bits
  - b3b2
  - 0 0 : FLD digit interrupt (at rising edge of each digit)
  - 0 1 : 1 × Tdisp
  - 1 0 : 2 × Tdisp
  - 1 1 : 3 × Tdisp
  - (0 1 to 1 1:) FLD blanking interrupt (at falling edge of the last digit)
- b4: Timing number control bit
  - 0 : 16 timing mode
  - 1 : 32 timing mode
- b5: Gradation display mode selection control bit
  - 0 : Not selecting
  - 1 : Selecting (**Note**)
- b6: Tdisp counter count source selection bit
  - 0 : $f(X_{IN})/16$
  - 1 : $f(X_{IN})/64$
- b7: High-breakdown voltage port drivability selection bit
  - 0 : Drivability strong
  - 1 : Drivability weak

Note: When the gradation display mode is selected, the max. number of timing is 16 timing. (Be sure to set the timing number control bit to “0”.)

b7 b6 b5 b4 b3 b2 b1 b0

FLD output control register
(FLDCON: address $0EFC_{16}$)

- b0: $P6_4$ to $P6_7$ output reverse bit
  - 0 : Output normally
  - 1 : Reverse output
- b1: Not used (return “0” when read); (Do not write “1”.)
- b2: $P6_4$ to $P6_7$ Toff invalid bit
  - 0 : Operation normally
  - 1 : Toff invalid
- b3: Not used (return “0” when read); (Do not write “1”.)
- b4: $P7_3$ dimmer output control bit
  - 0 : Normal port
  - 1 : Dimmer output
- b5: Generating/Not of CMOS port Toff section selection bit
  - 0 : Toff section not generated
  - 1 : Toff section generated
- b6: Generating/Not of high-breakdown voltage port Toff section selection bit
  - 0 : Toff section not generated
  - 1 : Toff section generated
- b7: Toff2 SET/RESET switch bit
  - 0 : Toff2 RESET; Toff1 SET
  - 1 : Toff2 SET; Tdisp RESET

Fig. 45 Structure of FLDC related registers (1)

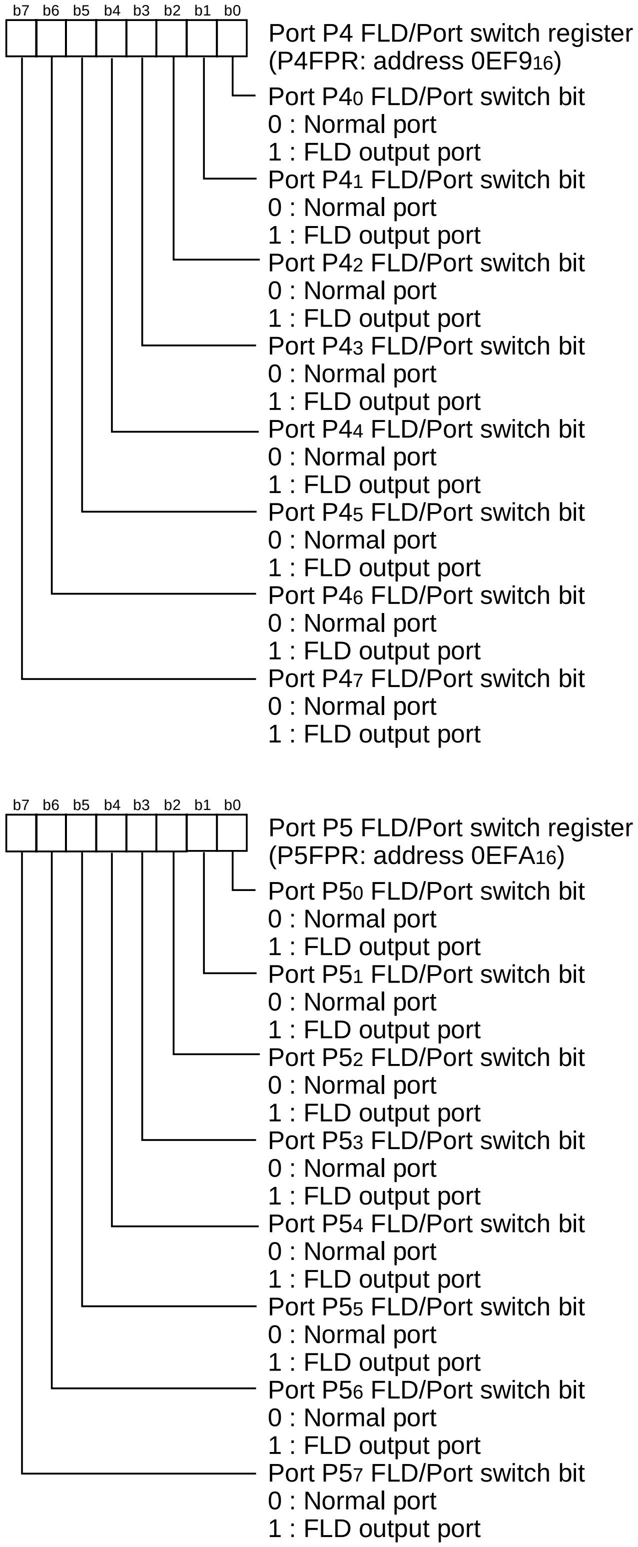


Fig. 46 Structure of FLDC related registers (2)

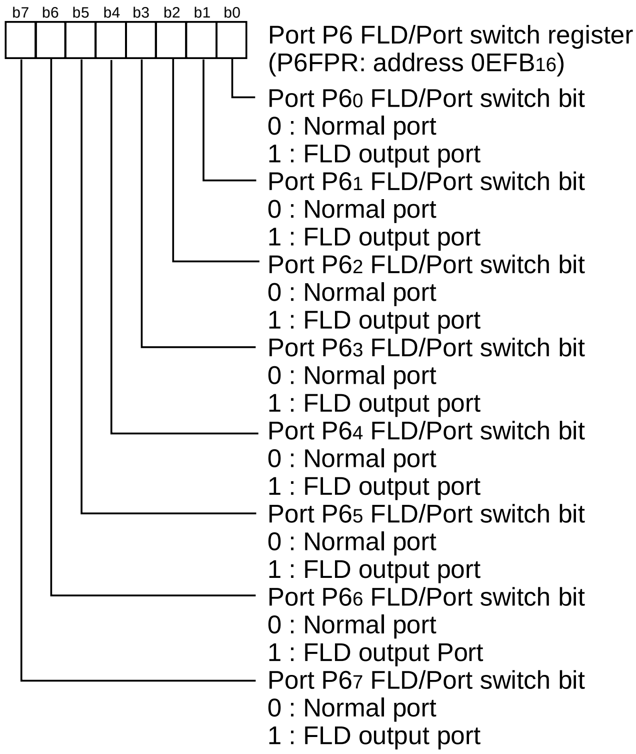


Fig. 47 Structure of FLDC related registers (3)

b7 b6 b5 b4 b3 b2 b1 b0

Port P0 digit output set switch register
($P0DOR$: address $0EF2_{16}$)

- b0: Port $P0_0$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b1: Port $P0_1$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b2: Port $P0_2$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b3: Port $P0_3$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b4: Port $P0_4$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b5: Port $P0_5$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b6: Port $P0_6$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b7: Port $P0_7$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output

b7 b6 b5 b4 b3 b2 b1 b0

Port P2 digit output set switch register
($P2DOR$: address $0EF3_{16}$)

- b0: Port $P2_0$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b1: Port $P2_1$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b2: Port $P2_2$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b3: Port $P2_3$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b4: Port $P2_4$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b5: Port $P2_5$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b6: Port $P2_6$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output
- b7: Port $P2_7$ FLD/Digit switch bit
  0 : FLD output
  1 : Digit output

Fig. 48 Structure of FLDC related registers (4)

FLD Automatic Display Pins

P0 to P6 are the pins capable of automatic display output for the FLD. The FLD starts operating by setting the automatic display control bit (bit 0 at address $0EF4_{16}$) to “1”. There is the FLD output function that outputs the RAM contents from the port every timing or the digit output function that drives the port high with a digit timing. The FLD can be displayed using the FLD output for the segments and the digit or FLD output for the digits. When using the FLD output for the digits, be sure to write digit display patterns to the RAM in advance. The remaining segment and digit lines can be used as general-purpose ports. Settings of each port are shown below.

Table 10 Pins in FLD automatic display mode

Port	Automatic display pin	Setting method
P0, P2	FLD_0 to FLD_{15}	The individual bits of the digit output set switch registers (addresses $0EF2_{16}$, $0EF3_{16}$) can set each pin to either an FLD port (“0”) or a digit port (“1”). When the pins are set for the digit port, the digit pulse output function is enabled, so that the digit pulses can always be output regardless the value of FLD automatic display RAM.
P1, P3	FLD_{16} to FLD_{31}	Setting the automatic display control bit (bit 0 of address $0EF4_{16}$) to “1” can set these ports to the FLD exclusive use port.
P4, P5, $P6_0$ to $P6_3$	FLD_{32} to FLD_{51}	The individual bits of the FLD/Port switch register (addresses $0EF9_{16}$ to $0EFB_{16}$) can set each pin to either an FLD port (“1”) or a general-purpose port (“0”).
$P6_4$ to $P6_7$	FLD_{52} to FLD_{55}	The individual bits of the port P6 FLD/Port switch register (address $0EFB_{16}$) can set each pin to either FLD port (“1”) or general-purpose port (“0”). A variety of output pulses can be available by setting of the FLD output control register (address $0EFC_{16}$). The port output structure is the CMOS output. When using the port as a display pin, a driver IC must be installed externally.

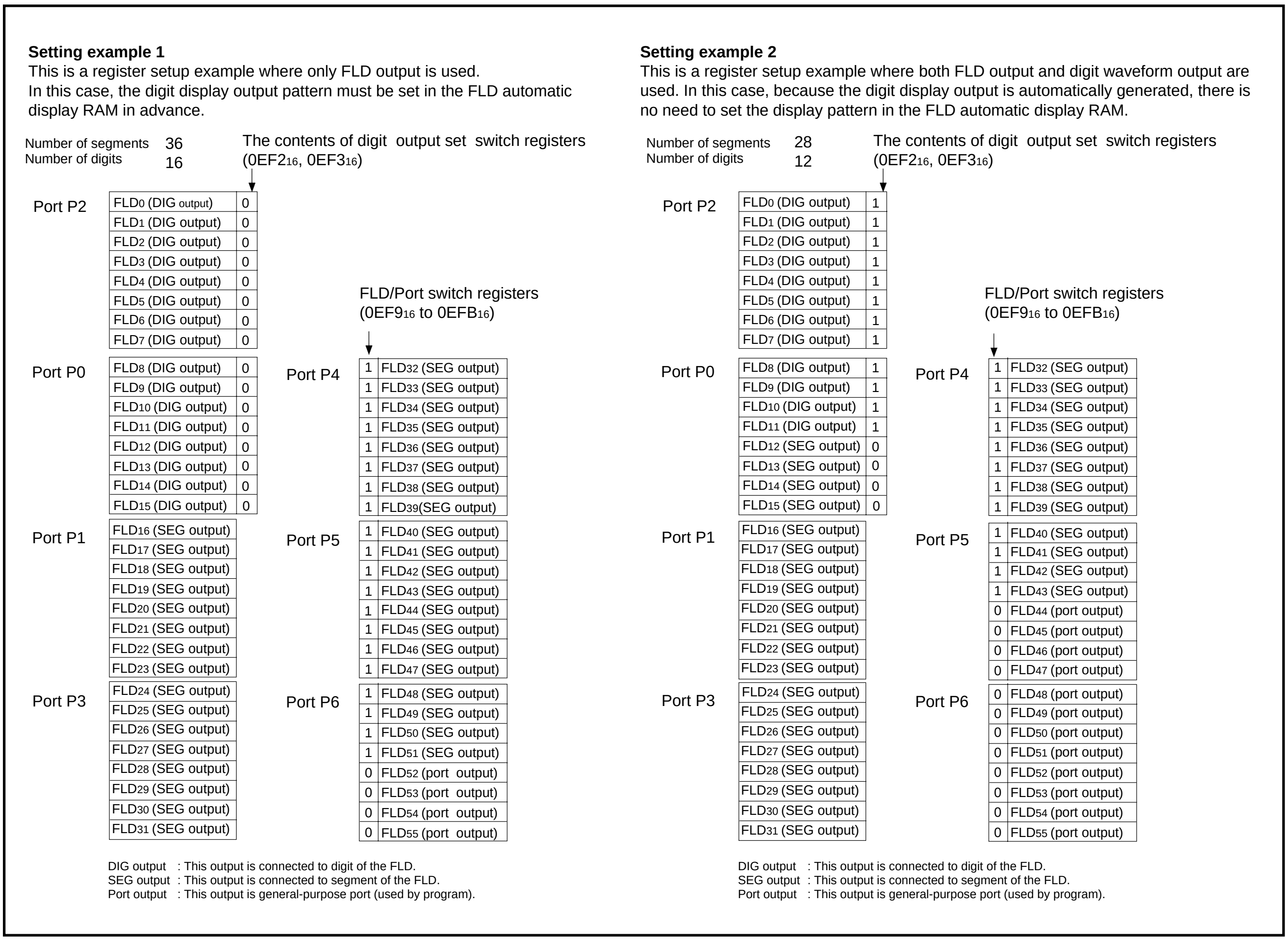


Fig. 49 Segment/Digit setting example

FLD Automatic Display RAM

The FLD automatic display RAM uses the 224 bytes of addresses $0E00_{16}$ to $0EDF_{16}$. 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 use/not use of gradation display.
The automatic display RAM in each mode is as follows:

(1) 16-timing•ordinary mode

This mode is used when the display timing is 16 or less. The 112 bytes of addresses $0E70_{16}$ to $0EDF_{16}$ are used as a FLD display data store area. Because addresses $0E00_{16}$ to $0E6F_{16}$ are not used as the automatic display RAM, they can be the ordinary RAM.

(2) 16-timing•gradation display mode

This mode is used when the display timing is 16 or less, in which mode each segment can be set for dark or bright display. The 224 bytes of addresses $0E00_{16}$ to $0EDF_{16}$ are used. The 112 bytes of addresses $0E70_{16}$ to $0EDF_{16}$ are used as an FLD display data store area, while the 112 bytes of addresses $0E00_{16}$ to $0E6F_{16}$ are used as a gradation display control data store area.

(3) 32-timing mode

This mode is used when the display timing is 16 or greater. This mode can be used for up to 32-timing.
The 224 bytes of addresses $0E00_{16}$ to $0EDF_{16}$ are used as an FLD display data store area.

The FLD data pointer (address $0EF8_{16}$) is a register to count display timings. This pointer has a reload register. When the pointer underflow occurs, it starts counting over again after being reloaded with the initial value in the reload register. Make sure that (the timing counts – 1) is set to the FLD data pointer. 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.

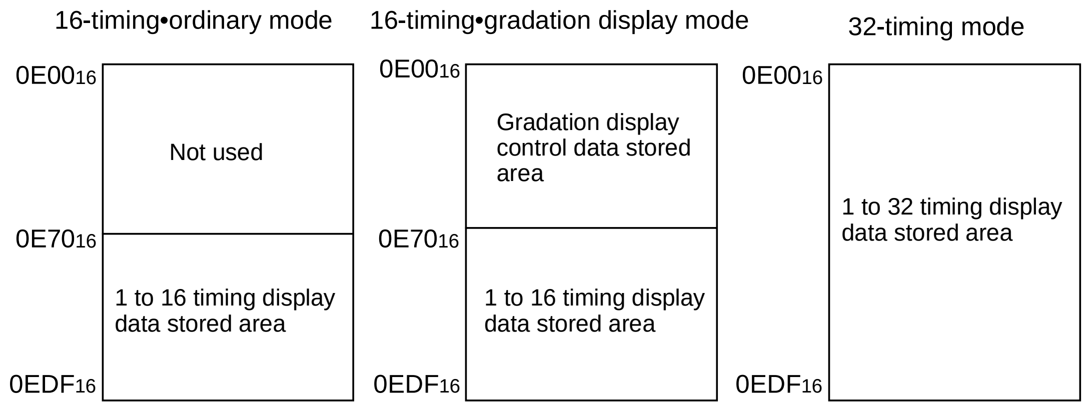


Fig. 50 FLD automatic display RAM assignment

Data Setup

(1) 16-timing•ordinary mode

The area of addresses 0E70$_{16}$ to 0EDF$_{16}$ are used as a FLD automatic display RAM.

When data is stored in the FLD automatic display RAM, the last data of FLD port P6 is stored at address 0E70$_{16}$, the last data of FLD port P5 is stored at address 0E80$_{16}$, the last data of FLD port P4 is stored at address 0E90$_{16}$, the last data of FLD port P3 is stored at address 0EA0$_{16}$, the last data of FLD port P1 is stored at address 0EB0$_{16}$, the last data of FLD port P0 is stored at address 0EC0$_{16}$, and the last data of FLD port P2 is stored at address 0ED0$_{16}$, to assign in sequence from the last data respectively.

The first data of the FLD port P6, P5, P4, P3, P1, P0, and P2 is stored at an address which adds the value of (the timing number – 1) to the corresponding addresses 0E70$_{16}$, 0E80$_{16}$, 0E90$_{16}$, 0EA0$_{16}$, 0EB0$_{16}$, 0EC0$_{16}$ and 0ED0$_{16}$.

Set the FLD data pointer reload register to the value given by (the timing number – 1).

(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 0070$_{16}$ 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” .

(3) 32-timing Mode

The area of addresses 0E00$_{16}$ to 0EDF$_{16}$ is used as a FLD automatic display RAM.

When data is stored in the FLD automatic display RAM, the last data of FLD port P6 is stored at address 0E00$_{16}$, the last data of FLD port P5 is stored at address 0E20$_{16}$, the last data of FLD port P4 is stored at address 0E40$_{16}$, the last data of FLD port P3 is stored at address 0E60$_{16}$, the last data of FLD port P1 is stored at address 0E80$_{16}$, the last data of FLD port P0 is stored at address 0EA0$_{16}$, and the last data of FLD port P2 is stored at address 0EC0$_{16}$, to assign in sequence from the last data respectively.

The first data of the FLD port P6, P5, P4, P3, P1, P0, and P2 is stored at an address which adds the value of (the timing number – 1) to the corresponding addresses 0E00$_{16}$, 0E20$_{16}$, 0E40$_{16}$, 0E60$_{16}$, 0E80$_{16}$, 0EA0$_{16}$ and 0EC0$_{16}$.

Set the FLD data pointer reload register to the value given by (the timing number – 1).

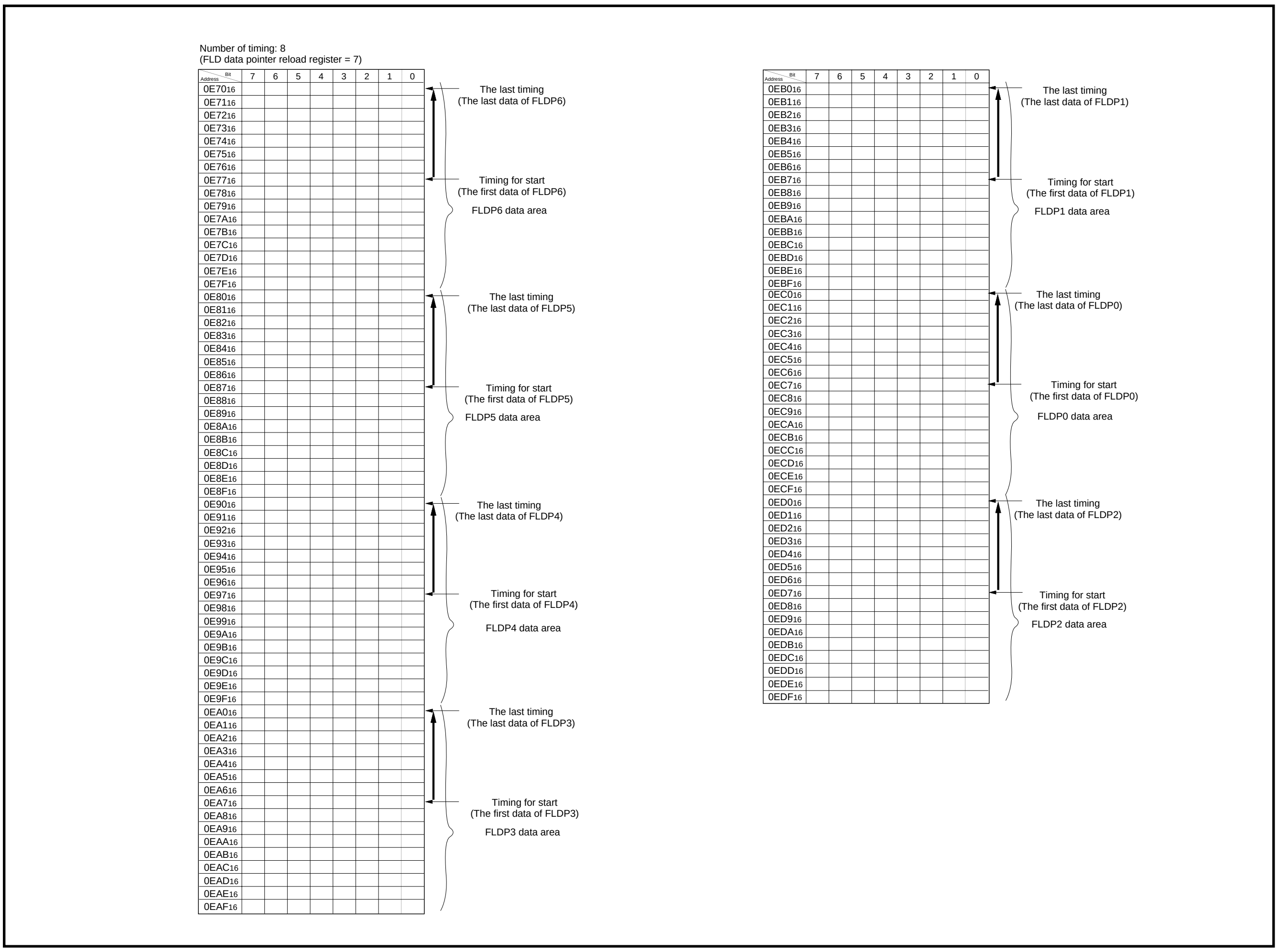


Fig. 51 Example of using FLD automatic display RAM in 16-timing•ordinary mode

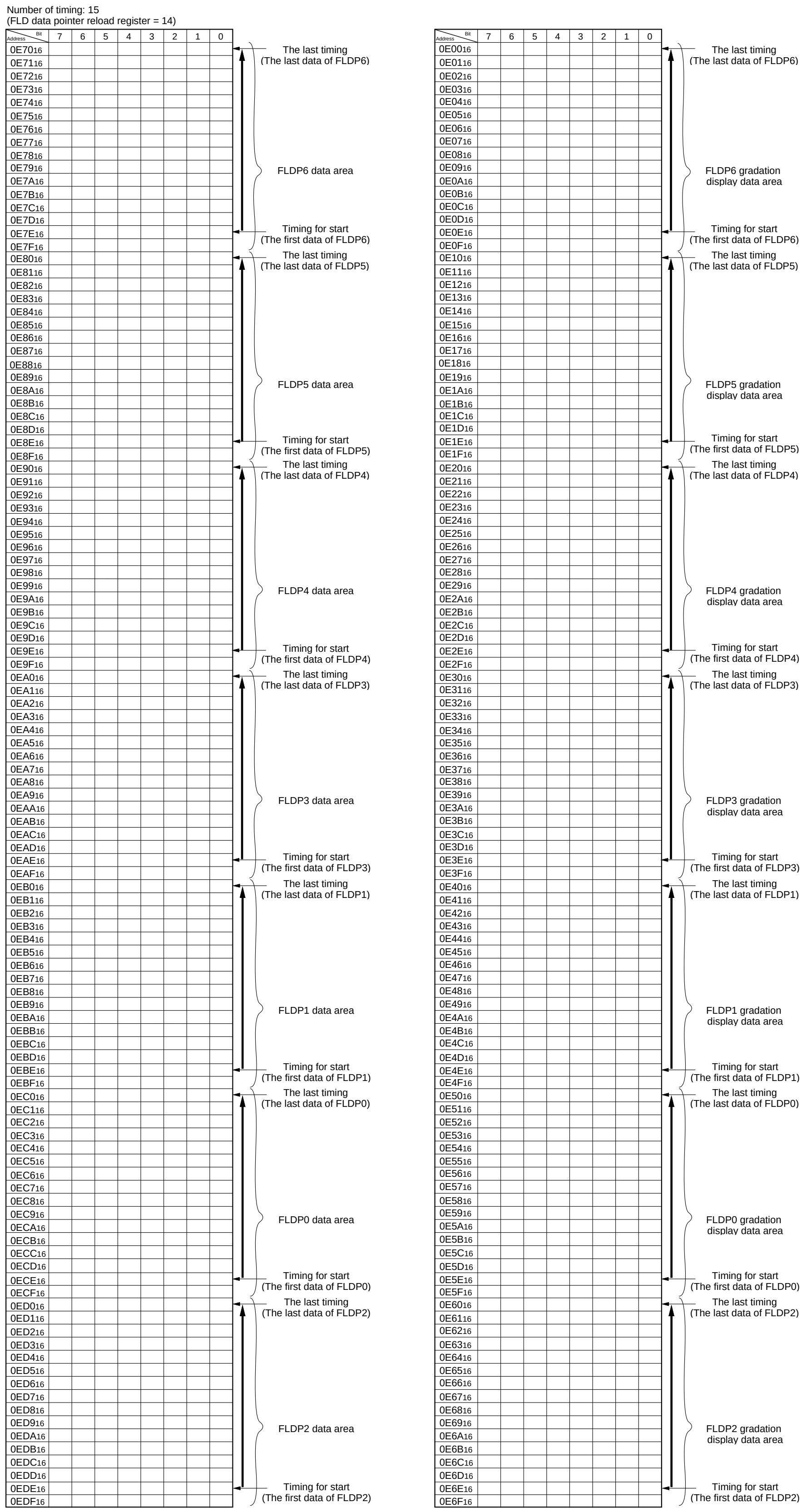


Fig. 52 Example of using FLD automatic display RAM in 16-timing•gradation display mode

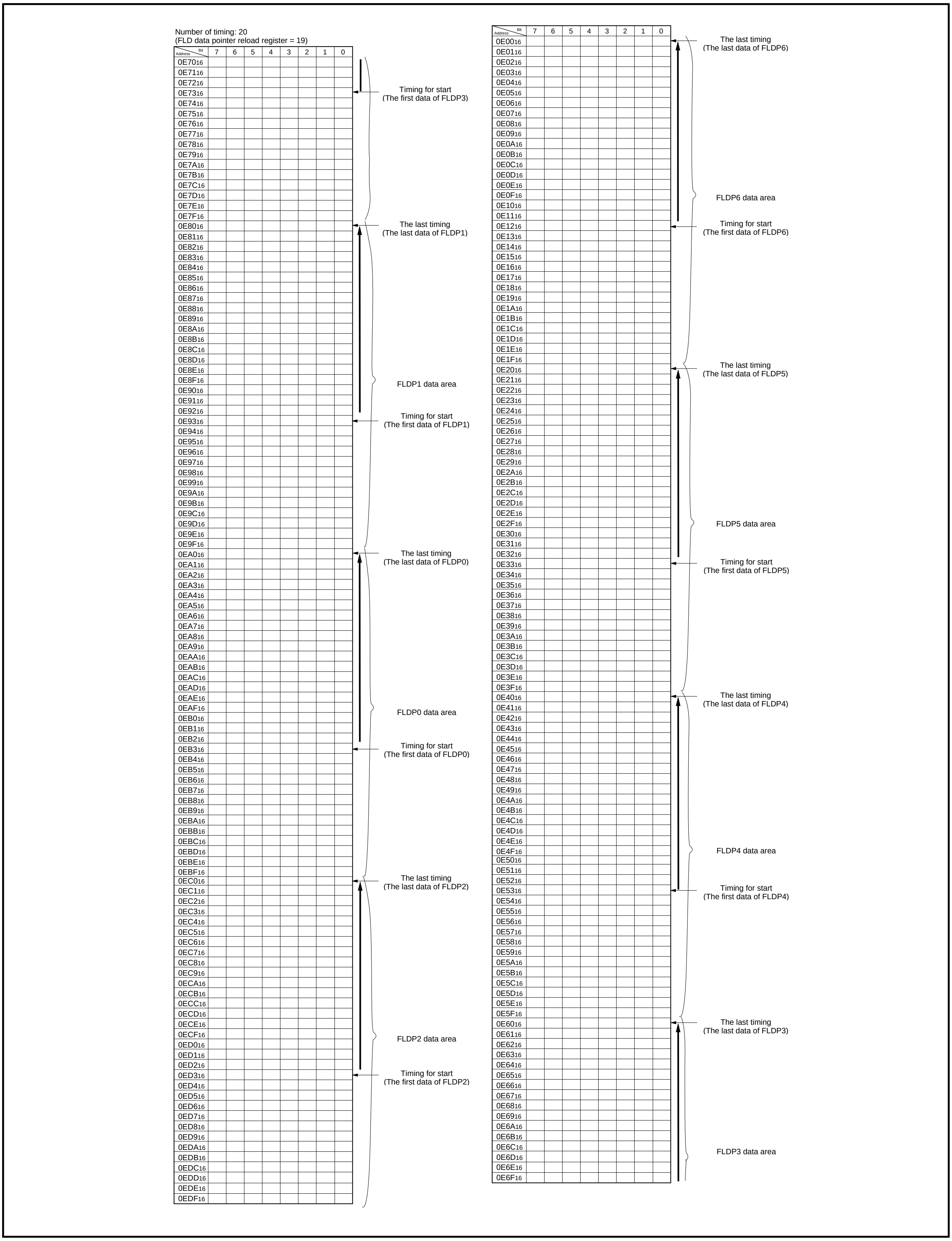


Fig. 53 Example of using FLD automatic display RAM in 32-timing mode

Timing Setting

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

(1) Tdisp time setting

The Tdisp time means the length of display timing. In non-gradation display mode, it consists of the FLD display output term and the Toff1 time. In gradation display mode, it consists of the display output term and the Toff1 time plus a low signal output term for dark display. 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). 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) × 4.0 µs (at X_{IN} = 4 MHz) = 804 µs. When reading the Tdisp time set register, the counting value is read out.

(2) Toff1 time setting

The Toff1 time means a non-output (low signal output) time to prevent blurring of FLD and for dimmer display. Use the Toff1 time set register to set this Toff1 time. Make sure the value set to Toff1 is smaller than Tdisp and Toff2. 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 × 4.0 µs (at X_{IN} = 4 MHz) = 120 µs.
Be sure to set the value of 03_{16} or more to the Toff1 time set register (address $0EF6_{16}$).

(3) Toff2 time setting

The Toff2 time is time for dark display. For bright display, the FLD display output remains effective until the counter that is counting Tdisp underflows. For dark display, however, "L" (or "off") signal is output when the counter that is counting Toff2 underflows. 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" .
Set the Toff2 time by the Toff2 time set register. Make sure the value set to Toff2 is smaller than Tdisp but larger than Toff1. 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 × 4.0 µs (at X_{IN} = 4 MHz) = 720 µs.
When bit 7 of the FLD output control register (address $0EFC_{16}$) is set to "1", be sure to set the value of 03_{16} or more to the Toff2 time set register (address $0EF7_{16}$).

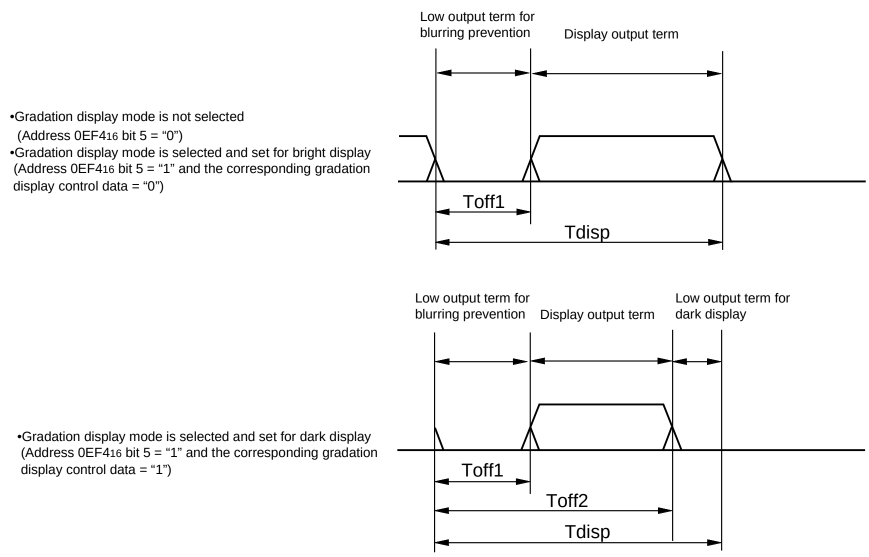


Fig. 54 FLD and digit output timing

FLD Automatic Display Start

Automatic display starts by setting both the automatic display control bit (bit 0 of address $0EF4_{16}$) and the display start bit (bit 1 of address $0EF4_{16}$) to "1". The RAM contents at a location apart from the start address of the automatic display RAM for each port by (FLD data pointer (address $0EF8_{16}$) – 1) are output to each port. The FLD data pointer (address $0EF8_{16}$) counts down in the Tdisp interval. When the count results in "FF_{16}", the pointer is reloaded and starts counting over again. Before setting the display start bit (bit 1 of address $0EF4_{16}$) to "1", be sure to set the FLD/port switch registers, digit output set switch registers, FLDC mode register, Tdisp time set register, Toff1 time set register, Toff2 time set register, and FLD data pointer.

During FLD automatic display, the display start bit always keeps "1", and FLD automatic display can be interrupted by writing "0" to this bit.

Key-scan and Interrupt

Either the FLD digit interrupt or FLD blanking interrupt can be selected using the Tscan control bits (bits 2, 3 of address $0EF4_{16}$).

The FLD digit interrupt is generated when the Toff1 time in each timing expires (at rising edge of digit output). Key scanning that makes use of FLD digits can be achieved using each FLD digit interrupt. To use FLD digit interrupts for key scanning, follow the procedure described below:

(1) Read the port value each time the interrupt occurs.

(2) The key is fixed on the last digit interrupt.

The output digit positions can be determined by reading the FLD data pointer (address $0EF8_{16}$).

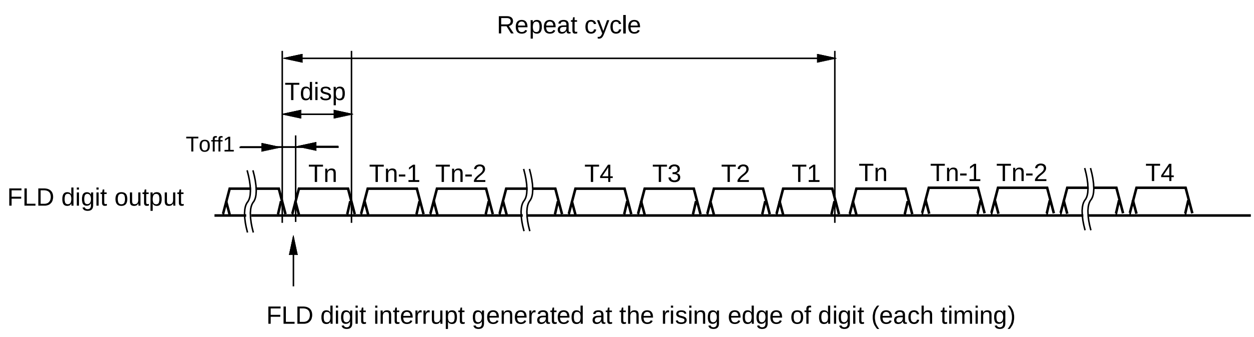


Fig. 55 Timing using digit interrupt

The FLD blanking interrupt is generated when the FLD data pointer (address $0EF8_{16}$) reaches "FF_{16}". The FLD automatic display output is turned off for a duration of 1 × Tdisp, 2 × Tdisp, or 3 × Tdisp depending on post-interrupt settings. During this time, key scanning that makes use of FLD segments can be achieved.
When the key scanning is performed with the segment during key-scan blanking time Tscan, follow the procedure described below:
(1) Write "0" to the automatic display control bit (bit 0 of address $0EF4_{16}$).
(2) Set the port corresponding to the segment for key scanning to the output port.
(3) Perform key scanning.
(4) Write "1" to the automatic display control bit.

■ Note

When performing a key-scan according to the above steps 1 to 4, take the following points into consideration.
1. Do not set the display start bit (bit 1 of address $0EF4_{16}$) to "0".
2. Do not set "1" in the ports corresponding to digits.

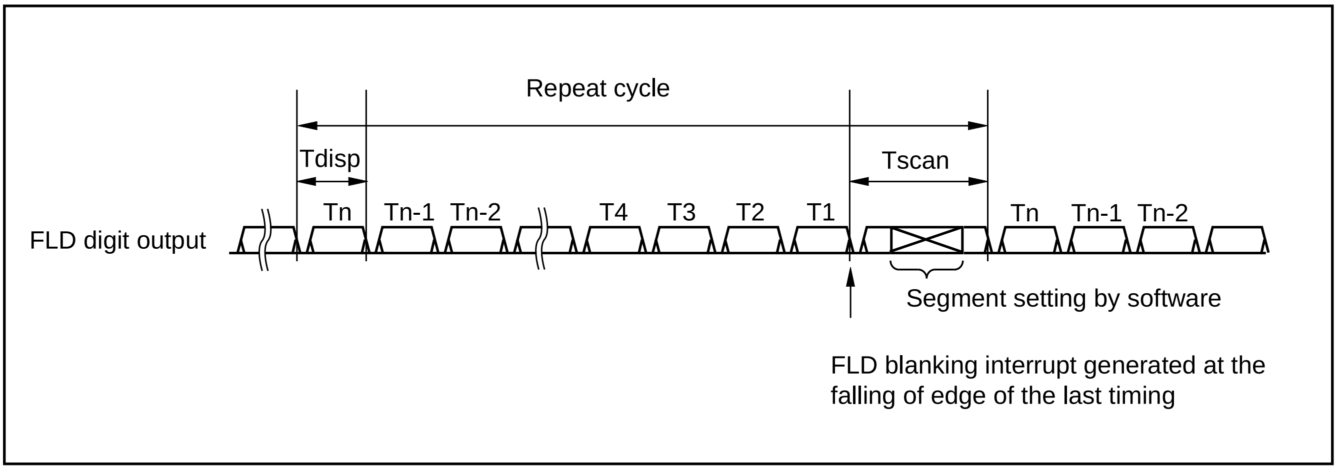


Fig. 56 Timing using FLD blanking interrupt

P64 to P67 Expansion Function

Ports P64 to P67 are CMOS output structure. FLD digit outputs can be increased as many as 16 lines by connecting a decoder converting 4-bit to 16-bit data to these ports. P64 to P67 have the function to allow for connection to a decoder converting 4-bit to 16-bit data.

(1) P64 to P67 Toff invalid function

This function disables the Toff1 time and Toff2 time and outputs display data for the duration of Tdisp. (See Figure 57.) This can be achieved by setting the P64 to P67 Toff invalid bit (bit 2 of address 0EFC16) to “1”.

(2) Dimmer signal output function

This function allows a dimmer signal creation signal to be output from DIMOUT (P73). The dimmer function can be achieved by controlling the decoder with this signal. (See Figure 57.) This function can be set by setting P73 dimmer output control bit (bit 4 of address 0EFC16) to “1”.

Unlike the Toff section generating/nothing function, this function disables all display data.

(3) P64 to P67 FLD output reverse function

P64 to P67 have the 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 the P64 to P67 output reverse bit of the FLD output control register (bit 0 of address 0EFC16) to “1”.

■ Note

In the case of gradation display mode and dark display, P64 to P67 Toff invalid function is disabled.

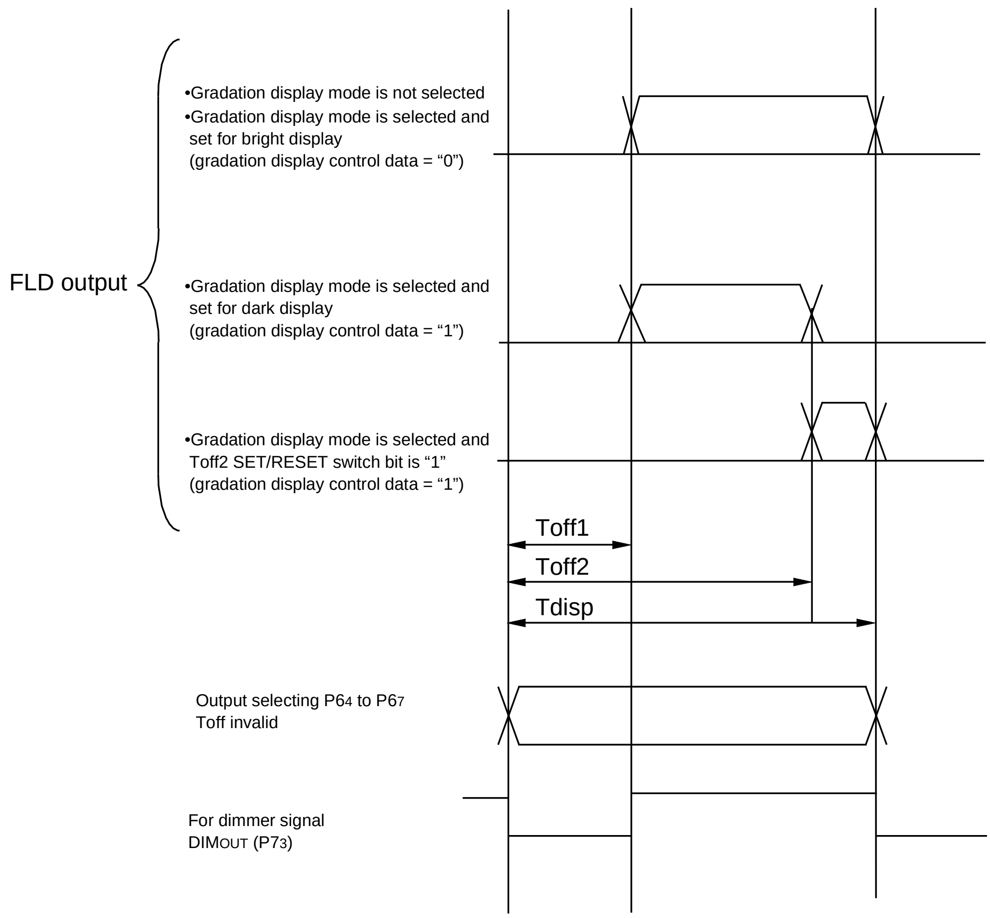


Fig. 57 P64 to P67 FLD output pulses

Toff Section Generate/Nothing Function

The function is for reduction of useless noises which generated as every switching of ports, because of the combined capacity of among FLD ports. When the continuous data is output to each FLD port, the Toff1 section of the continuous parts is not generated. (See Figure 58)

If it needs Toff1 section on FLD pulses, set the generating /not of CMOS port Toff section selection bit (bit 5 of address $0EFC_{16}$) to "1" and set the generating /not of high-breakdown-voltage port Toff section selection bit to "1".

High-breakdown-voltage ports (P2, P0, P1, P3, P4, P5, $P6_3$ to $P6_0$, total 52 pins) generate Toff1 section by setting the generating /not of high-breakdown-voltage port Toff section selection bit to "1".

The CMOS ports ($P6_4$ to $P6_7$, total 4 pins) generate Toff1 section by setting the generating /not of CMOS port Toff section selection bit to "1".

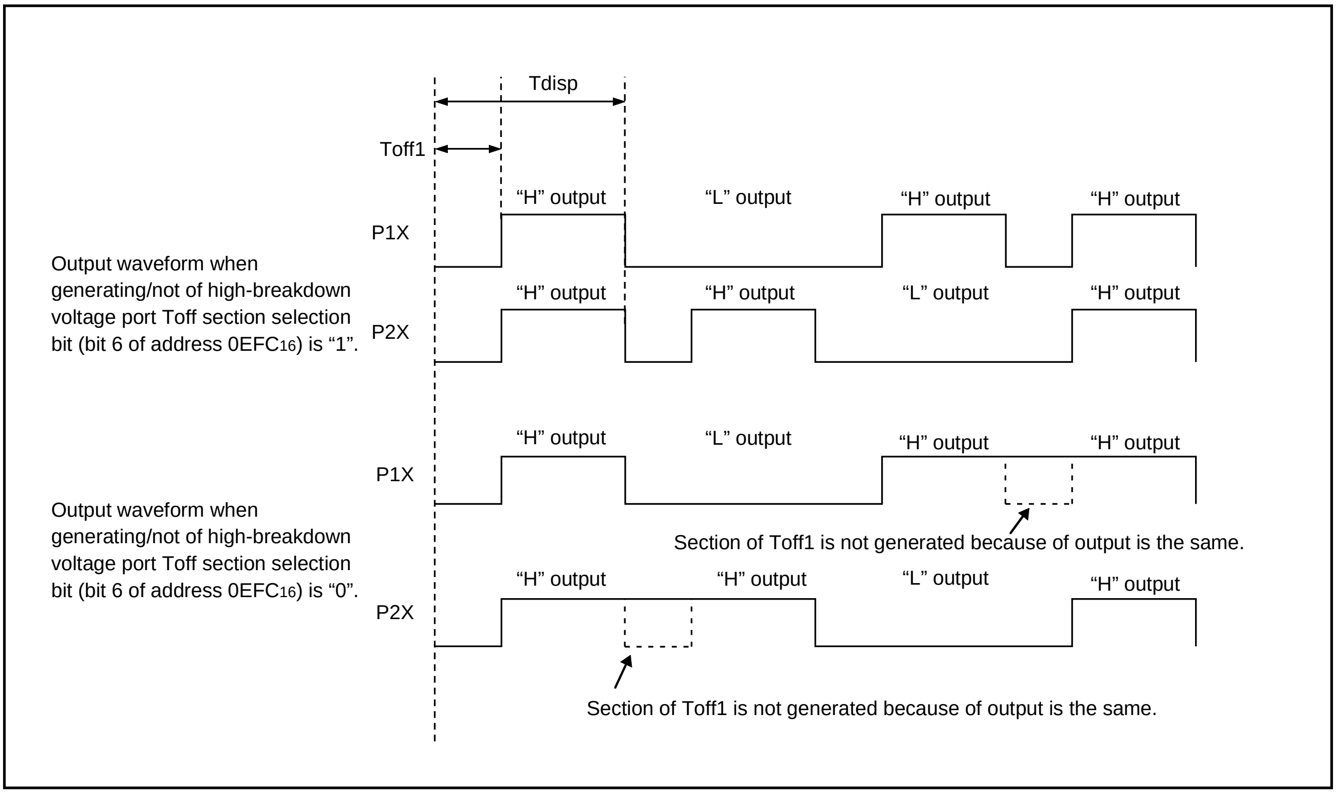


Fig. 58 Toff section generating/nothing function

Toff2 SET/RESET Switch Function

In gradation display mode, the values set by the Toff2 time set register (TOFF2) are effective. When the Toff2 SET/RESET switch bit of FLD output control register (bit 7 of address $0EFC_{16}$) is "0", RAM data is output to the FLD output ports (SET) at the time that is set by TOFF1 and it is turned to "0" (RESET) at the time that is set by TOFF2.

When Toff2 SET/RESET switch bit is "1", RAM data is output (SET) at the time that is set by TOFF2 and it is turned to "0" (RESET) when the Tdisp time expires.

■ Note

In the case of gradation display mode and dark display, the Toff section generate/nothing function is disabled.

Digit Pulses Output Function

$P0_0$ to $P0_7$ and $P2_0$ to $P2_7$ can output digit pulses by using the digit output set switch registers. Set the digit output set switch registers by setting as many consecutive 1s as the timing count from $P2_0$. The contents of FLD automatic display RAM for the ports that have been selected for digit output are disabled, and the pulse shown in Figure 59 is output automatically.

The output timing consists of Tdisp time and Toff1 time, and Toff2 time does not exist.

Because the contents of FLD automatic display RAM are disabled, the segment data can be changed easily even when segment data and digit data coexist at the same address in the FLD automatic display RAM.

This function is effective in 16-timing•ordinary mode and 16-timing gradation display mode. If a value is set exceeding the timing count (FLD data pointer reload register's set value + 1) for any port, the output of such port is "L".

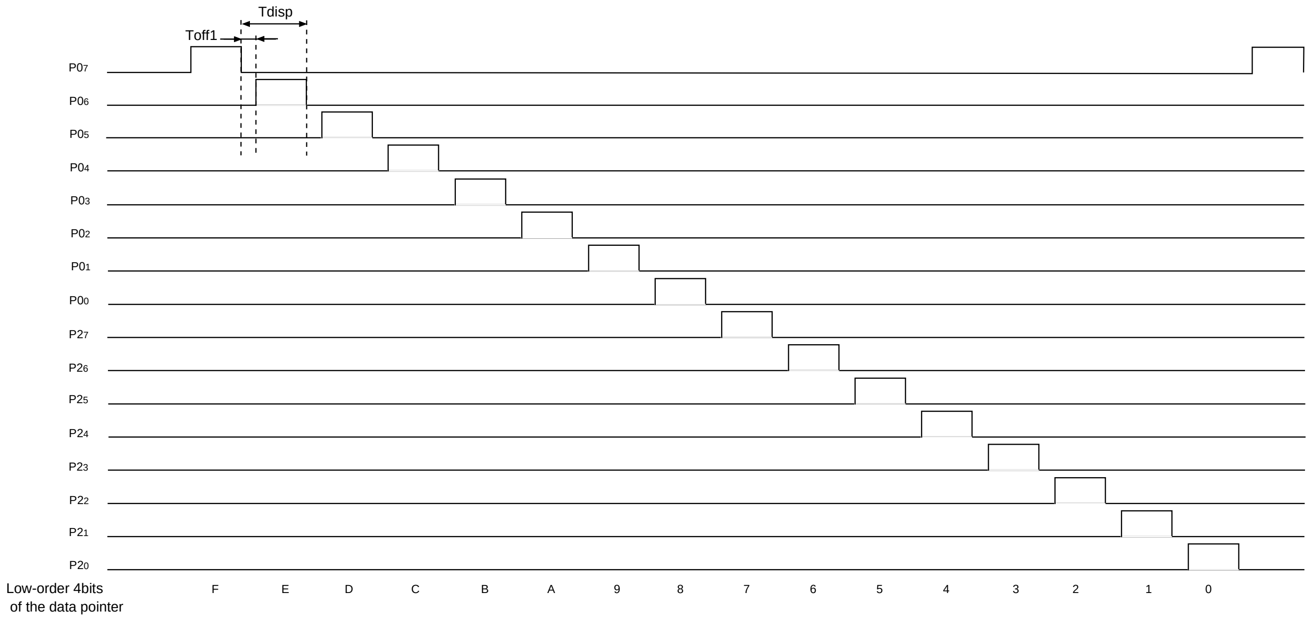


Fig. 59 Digit pulses output function

A-D CONVERTER

The 38B7 group has a 10-bit A-D converter. The A-D converter performs successive approximation conversion.

[A-D Conversion Register] ADH, ADL

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.

[AD/DA 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 writing “0” in this bit.

[Comparison Voltage Generator]

The comparison voltage generator divides the voltage between AVss and V_{REF} by 1024, and outputs the divided voltages.

[Channel Selector]

The channel selector selects one of the input ports PA_7/AN_7–PA_0/AN_0, and $P9_7/B_{UZ02}/AN_{15}$ to $P9_0/S_{IN3}/AN_8$ and inputs it to the comparator.

[Comparator and Control Circuit]

The comparator and control circuit compares an analog inputvoltage 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 that set $f(X_{IN})$ to at least 250 kHz during A-D conversion. Additionally, bit 7 of the CPU mode register (address $003B_{16}$) must be set to “0”.

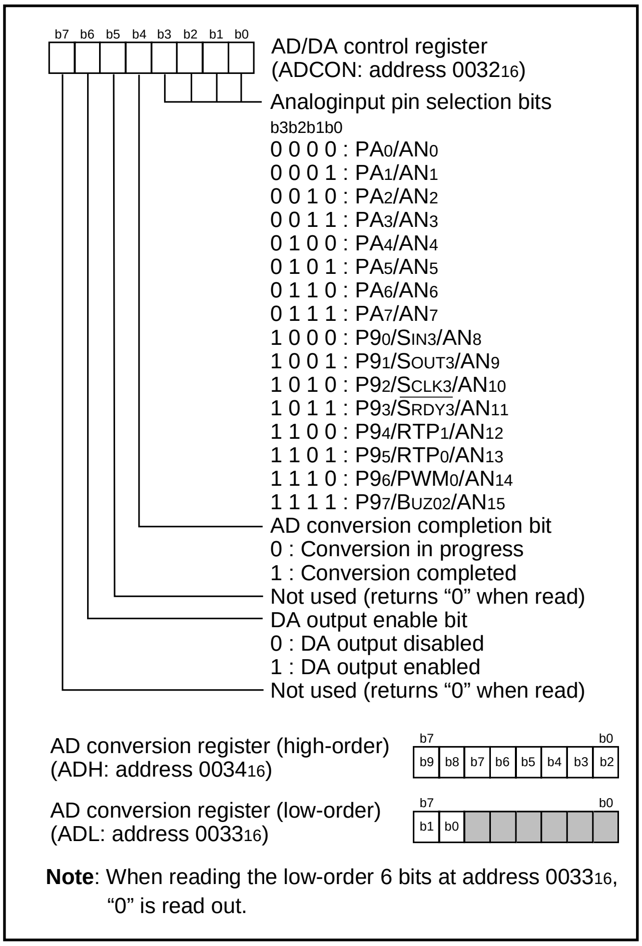


Fig. 60 Structure of AD/DA control register

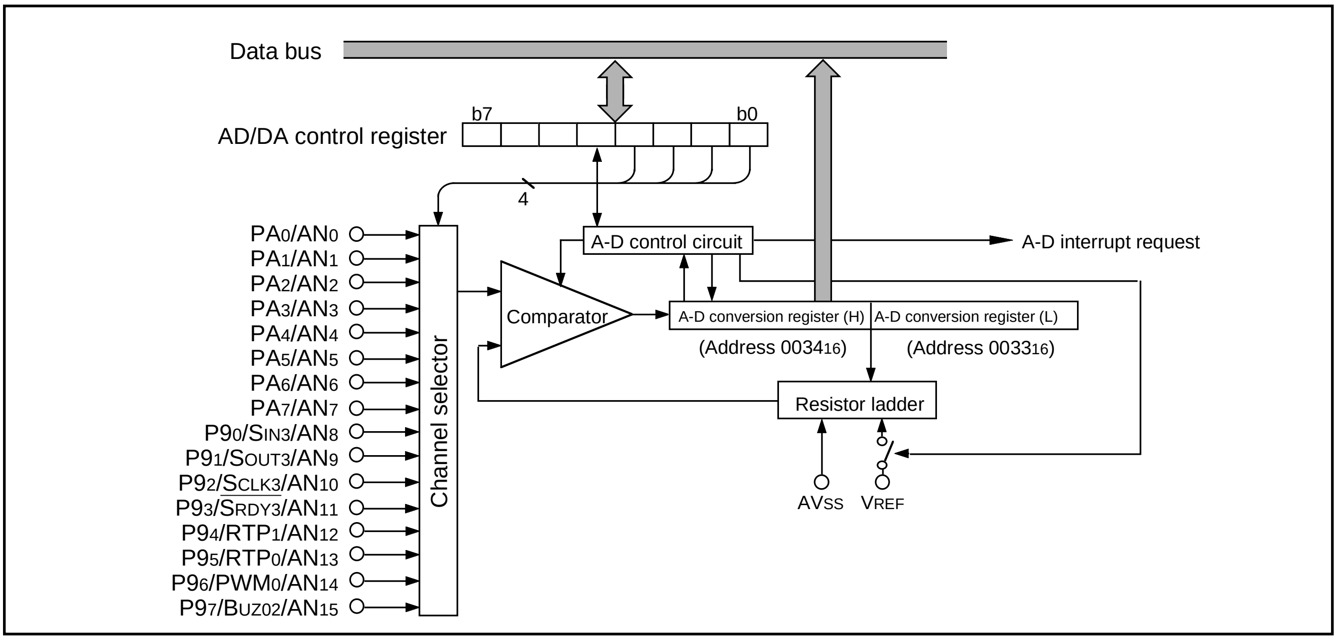


Fig. 61 Block diagram of A-D converter

D-A CONVERTER

The 38B7 group has one internal D-A converter with 8-bit resolution.

The D-A conversion is performed by setting the value in the D-A conversion register. The result of D-A conversion is output from the DA pin by setting the DA output enable bit to “1”.

When using the D-A converter, the PB_0/DA port direction register bit must be set to “0” (input status).

The output analog voltage V is determined by the value n (decimal notation) in the D-A conversion register as follows:

$V = V_{REF} \times n/256$ (n = 0 to 255)

Where V_{REF} is the reference voltage.

At reset, the D-A conversion register is cleared to “00_{16}”, and the DA output enable bit is cleared to “0”, and PB_0/DA pin becomes high impedance.

The DA output does not have buffers. Accordingly, connect an external buffer when driving a low-impedance load.

Set V_{CC} to 3.0 V or more when using the D-A converter.

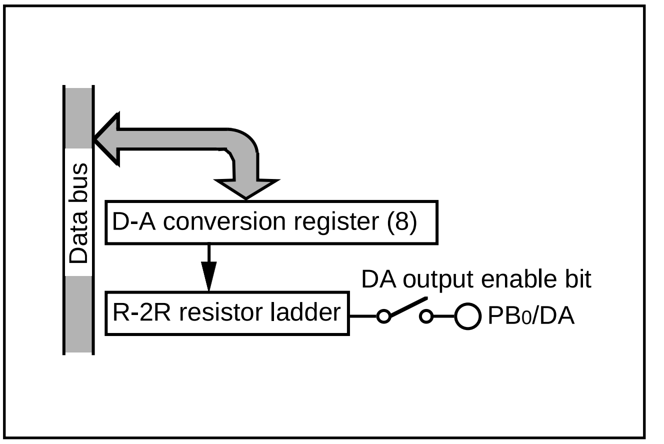


Fig. 62 Block diagram of D-A converter

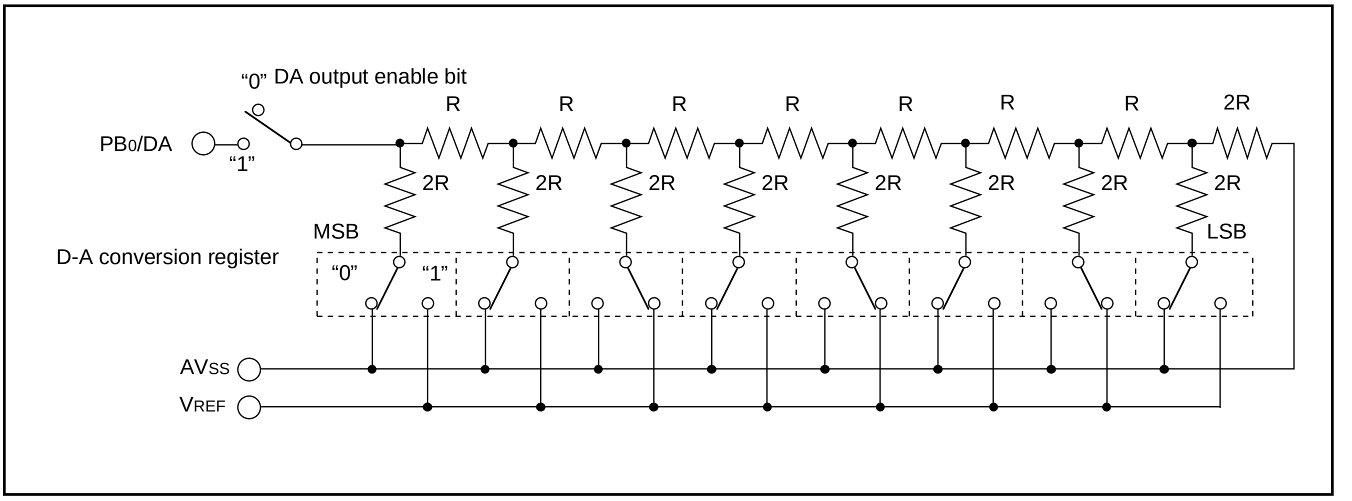


Fig. 63 Equivalent connection circuit of D-A converter

PWM (Pulse Width Modulation)

The 38B7 group has a PWM function with a 14-bit resolution. When the oscillation frequency XIN 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 XIN clock.
The explanation in the rest assumes XIN = 4 MHz.

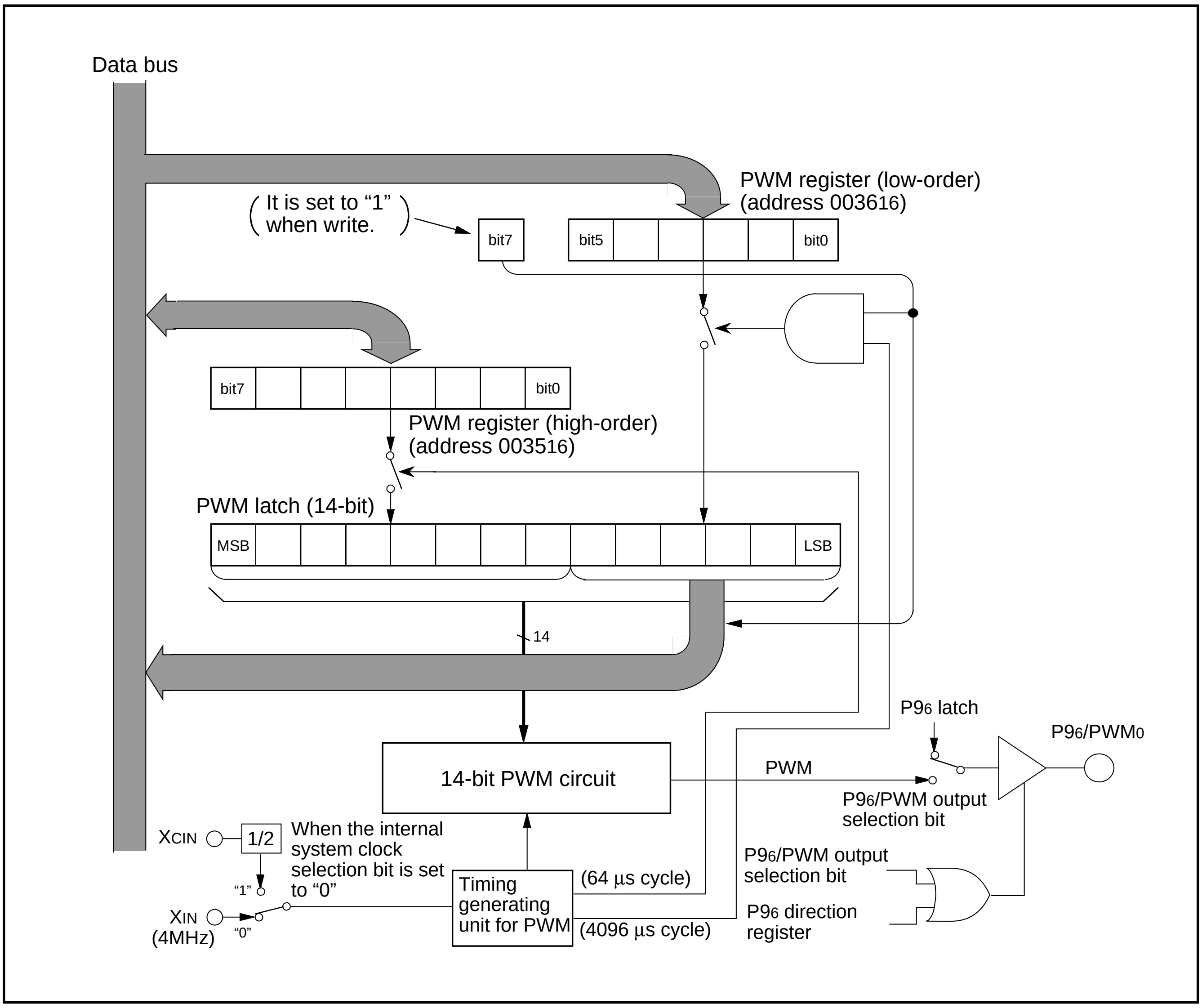


Fig. 64 PWM block diagram

Data Setup

The PWM output pin also function as port $P9_6$. Set port $P9_6$ 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 0035_{16}) and the low-order 6 bits are set in the low-order PWM register PWML (address 0036_{16}).

PWM Operation

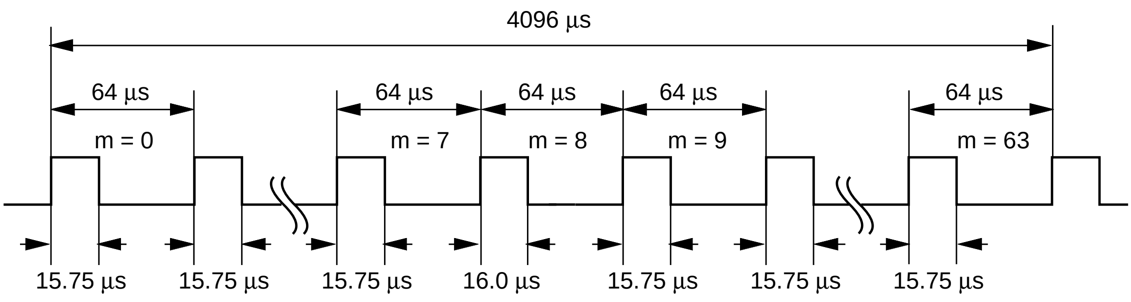
The timing of the 14-bit PWM function is shown in Figure 65.
The 14-bit PWM data is divided into the low-order 6 bits and the high-order 8 bits in the PWM latch.
The high-order 8 bits of data determine how long an “H” level signal is output during each sub-period. There are 64 sub-periods in each period, 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 11, the ADD bit is decided either “H” or “L.”
That is, only in the sub-period tm shown in Table 11 in the PWM cycle period T = 64 t, 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 t8, t24, t32, t40 and t56 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 t, and this sub-period t (= 64 µs, approximate 15.6 kHz) becomes cycle period approximately.

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). Pulses output from the PWM output pin correspond to this latch contents.
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”, it is not done when bit 7 is “1”.

Table 11 Relationship between low-order 6-bit data and setting period of ADD bit

Low-order 6-bit data	Sub-periods tm lengthened (m = 0 to 63)
0 0 0 0 0 0 (LSB)	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



Pulse width modulation register H: 00111111
Pulse width modulation register L: 000101
Sub-periods where “H” pulse width is 16.0 µs: m = 8, 24, 32, 40, 56
Sub-periods where “H” pulse width is 15.75 µs: m = all other values

Fig. 65 PWM timing

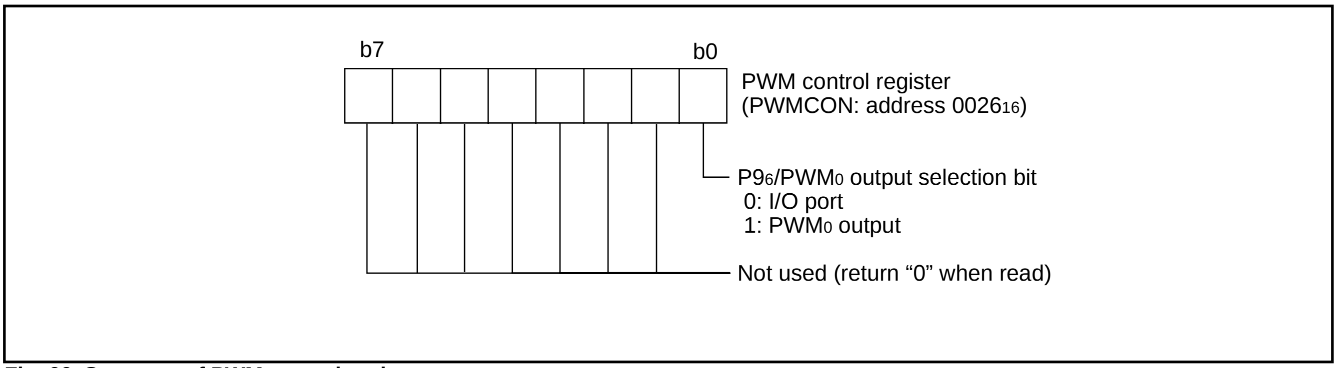


Fig. 66 Structure of PWM control register

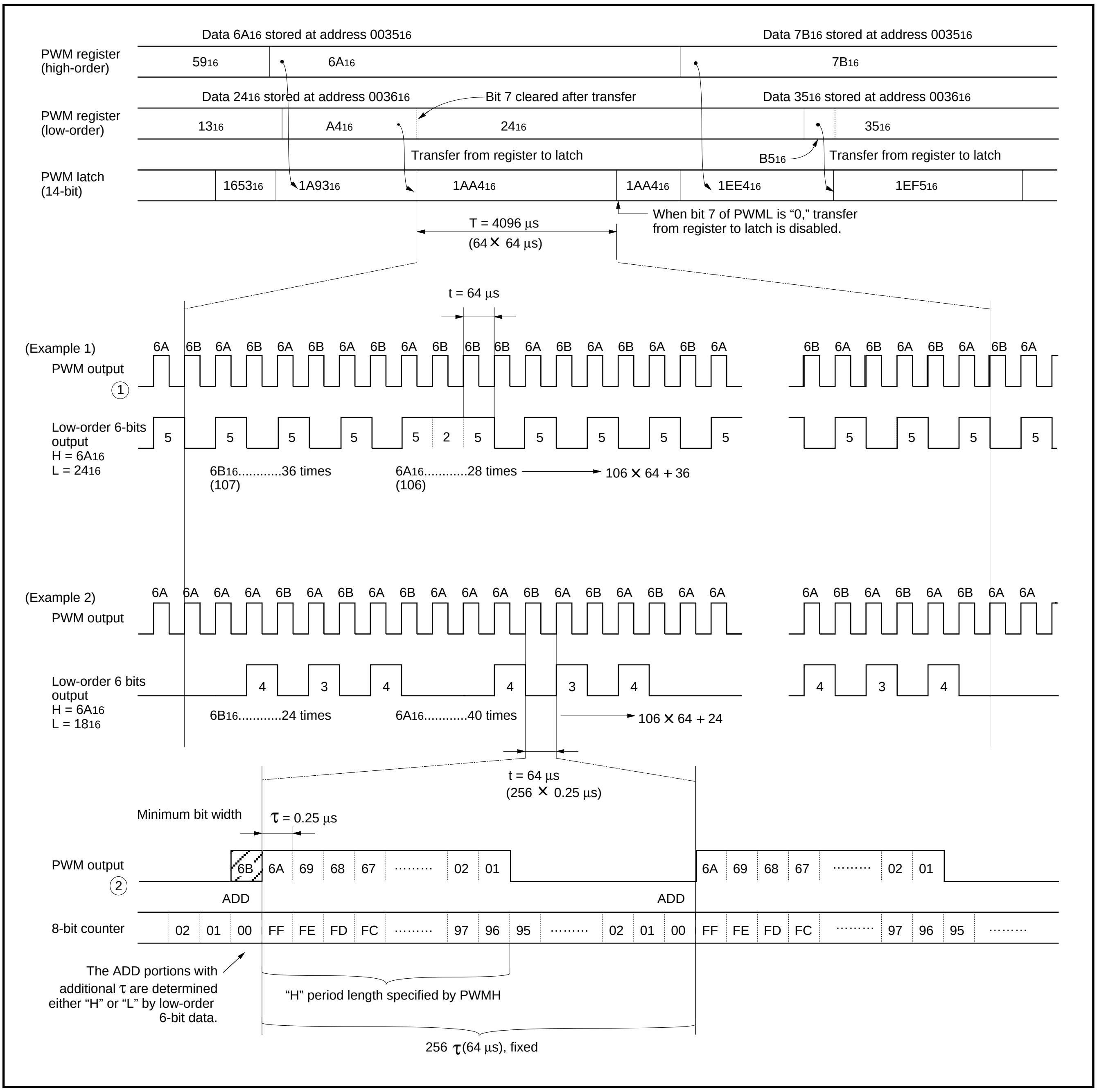


Fig. 67 14-bit PWM timing

INTERRUPT INTERVAL DETERMINATION FUNCTION

The 38B7 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 P7₂/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 INT_2 interrupt by setting bit 2 of the interrupt control register 1 (address $003E_{16}$). Select the rising interval or falling interval by setting bit 2 of the interrupt edge selection register (address $003A_{16}$).
2. Set bit 0 of the interrupt interval determination control register (address 0031_{16}) 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.
4. When the signal of polarity which is set on the INT_2 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 selected above is input again, the value of the 8-bit binary up counter is transferred to the interrupt interval determination register (address 0030_{16}), and the remote control interrupt request occurs. Immediately after that, the 8-bit binary up counter continues to count up again from "00_{16}".
6. When count value reaches "FF_{16}", the 8-bit binary up counter stops counting up. Then, simultaneously when the next counter sampling clock is input, the counter sets value "FF_{16}" to the interrupt interval determination register to generate the counter overflow interrupt request.

Noise Filter

The $P7_2/INT_2$ 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_{16}".
2. The $P7_2/INT_2$ 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 INT_2 input signal to 3 cycles or more of the sample clock.

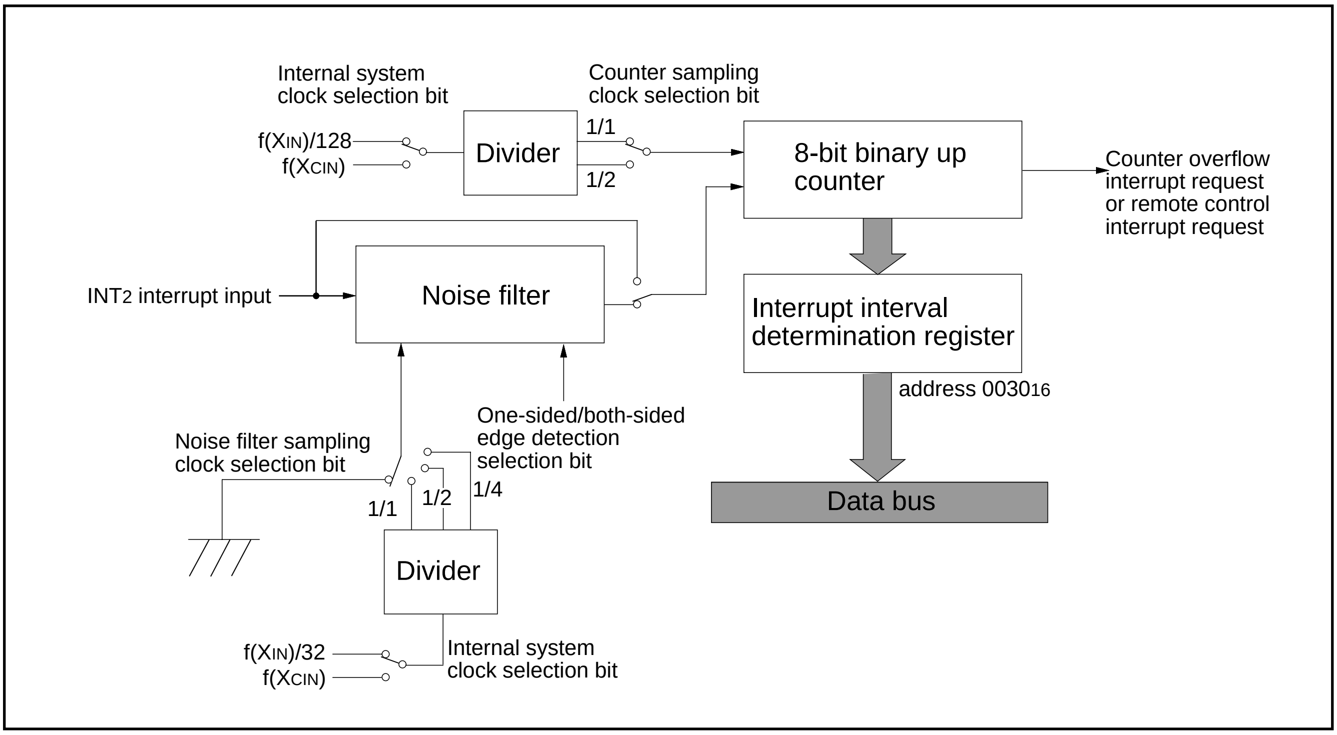


Fig. 68 Interrupt interval determination circuit block diagram

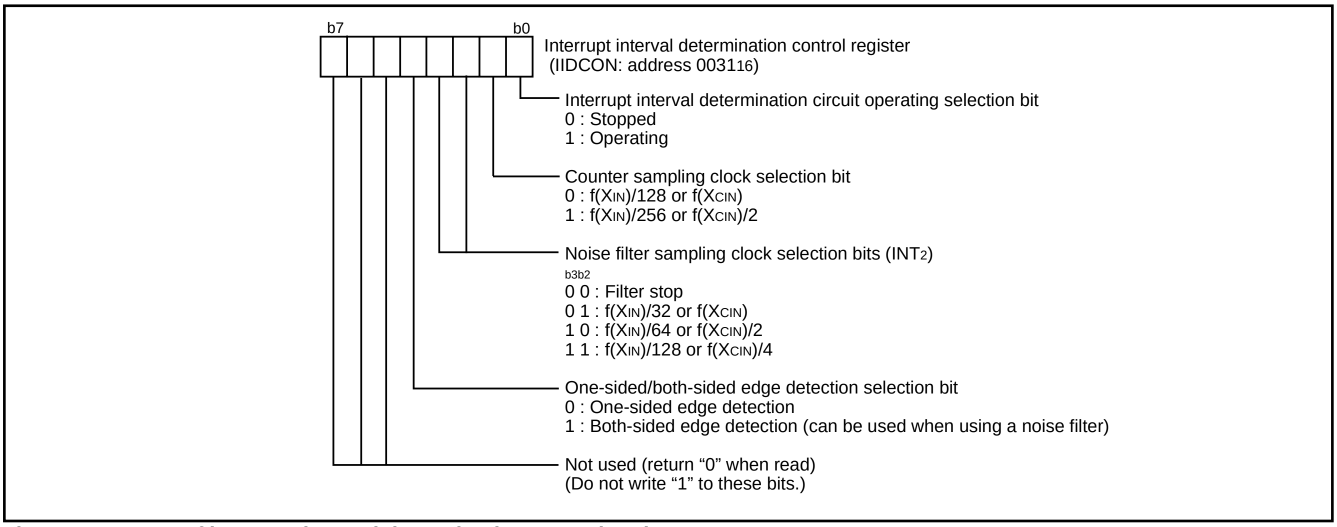


Fig. 69 Structure of interrupt interval determination control register

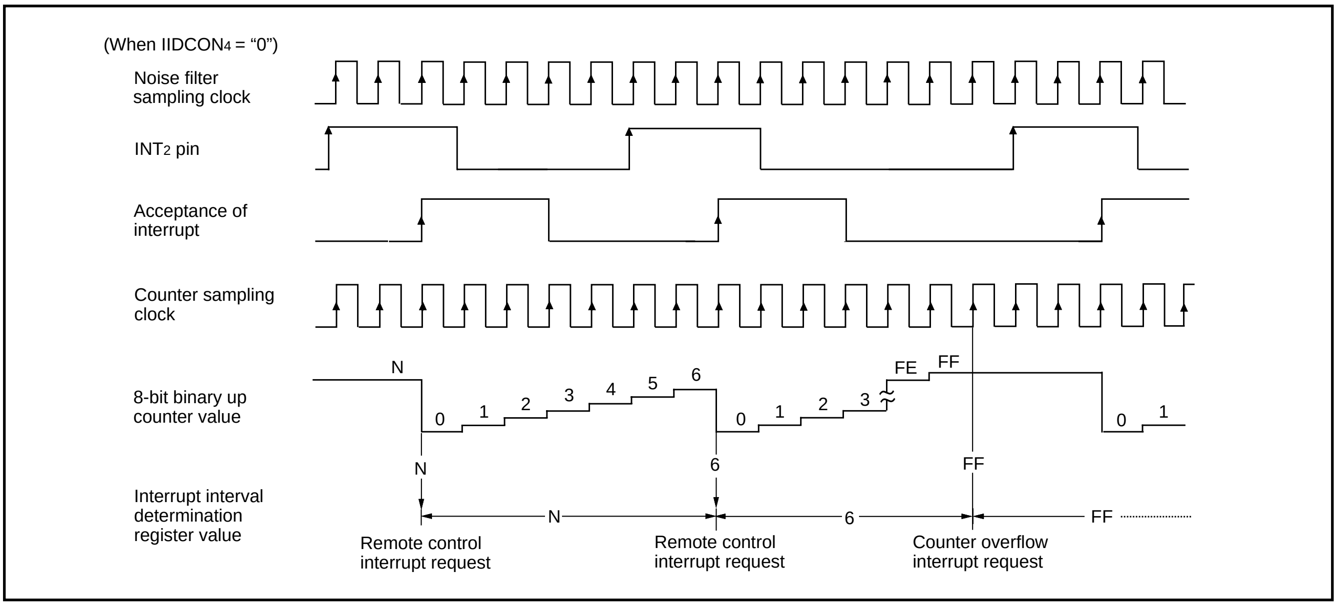


Fig. 70 Interrupt interval determination operation example (at rising edge active)

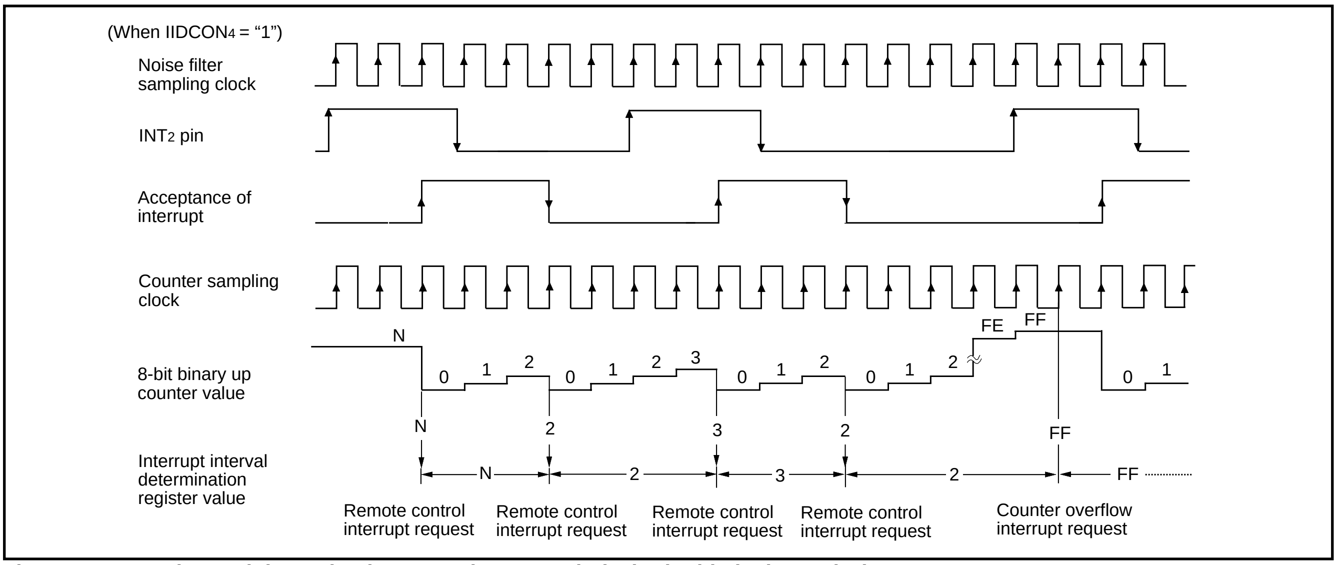


Fig. 71 Interrupt interval determination operation example (at both-sided edge active)

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 8-bit watchdog timer H.

Standard Operation Of Watchdog Timer

When any data is not written into the watchdog timer control register (address $0EEE_{16}$) after reset, 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 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 may be started before an underflow. When the watchdog timer control register 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 $0EEE_{16}$), a watchdog timer H is set to "FF_{16}" and a watchdog timer L to "FF_{16}".

(2) Watchdog timer H count source selection bit operation

Bit 7 of the watchdog timer control register (address $0EEE_{16}$) 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 to 131.072 ms at f(X_{IN}) = 4 MHz frequency, and 32.768 s at f(X_{CIN}) = 32 kHz frequency.

When this bit is set to "1", the count source becomes the signal divided by 8 for f(X_{IN}) or divided by 16 for f(X_{CIN}). The detection time in this case is set to 512 µs at f(X_{IN}) = 4 MHz frequency, and 128 ms at f(X_{CIN}) = 32 kHz frequency.

This bit is cleared to "0" after reset.

(3) Operation of STP instruction disable bit

Bit 6 of the watchdog timer control register (address $0EEE_{16}$) 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.

If 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 reset.

■ 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 any data in the watchdog timer control register (address $0EEE_{16}$) before executing the STP instruction.

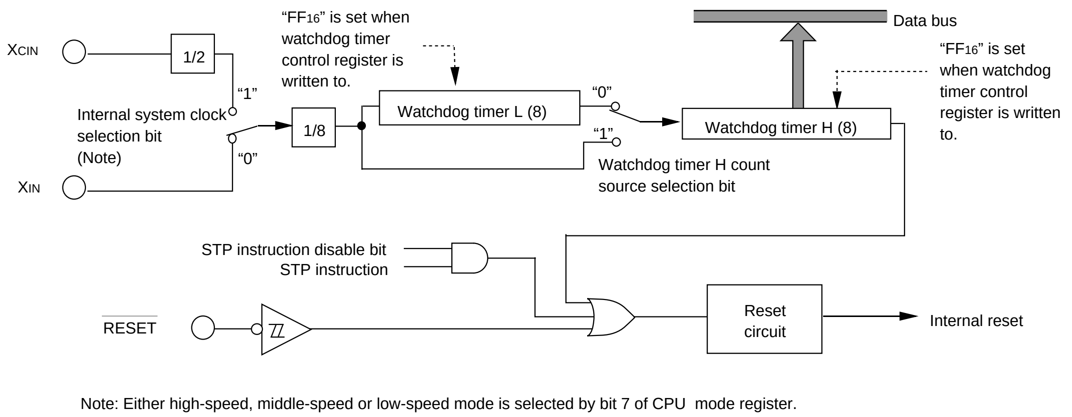


Fig. 72 Block diagram of watchdog timer

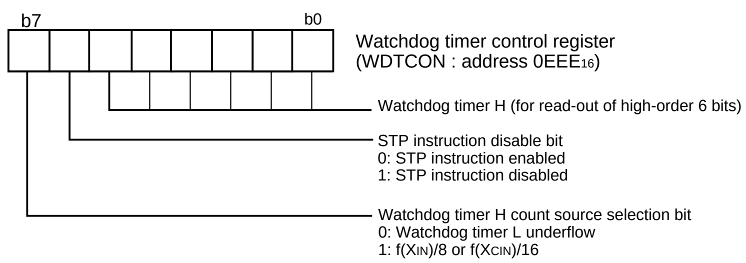


Fig. 73 Structure of watchdog timer control register

BUZZER OUTPUT CIRCUIT

The 38B7 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 $0EFD_{16}$). Either $P7_7/B_{UZ01}$ or $P9_7/B_{UZ02}/AN_{15}$ can be selected as a buzzer output port by the output port selection bits (b2 and b3 of address $0EFD_{16}$).
The buzzer output is controlled by the buzzer output ON/OFF bit (b4).

Note: In the low-speed mode, a buzzer output is made OFF.

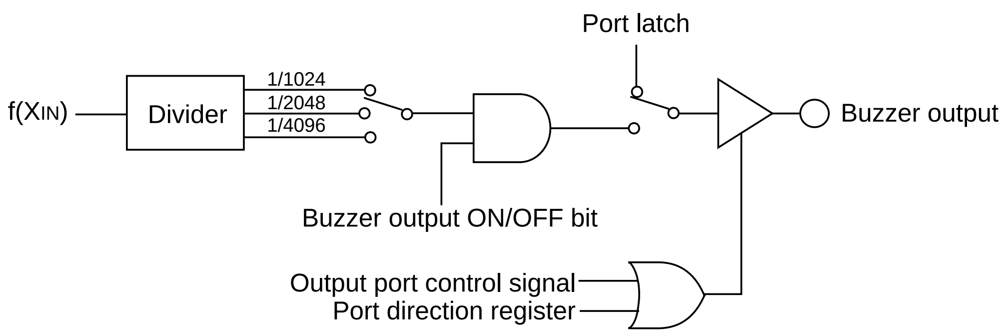


Fig. 74 Block diagram of buzzer output circuit

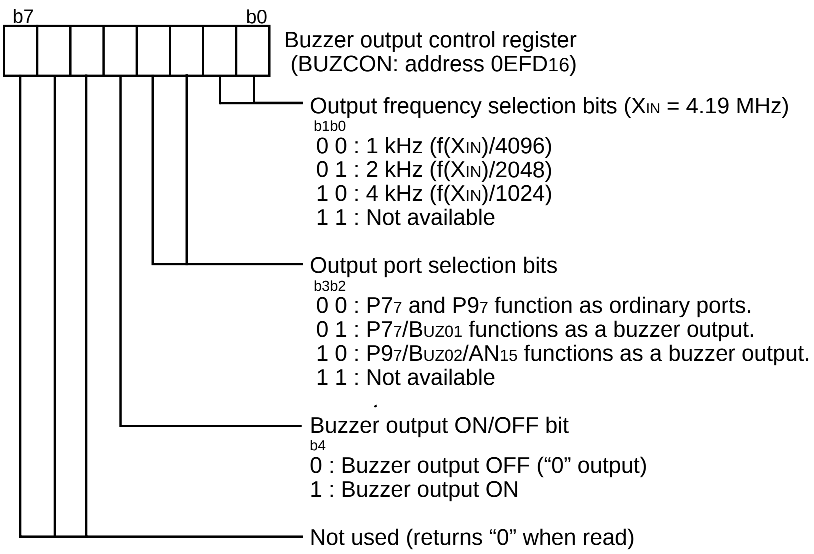


Fig. 75 Structure of buzzer output control register

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 0.54 V for Vcc of 2.7 V (switching to the high-speed mode, a power source voltage must be between 4.0 V and 5.5 V).

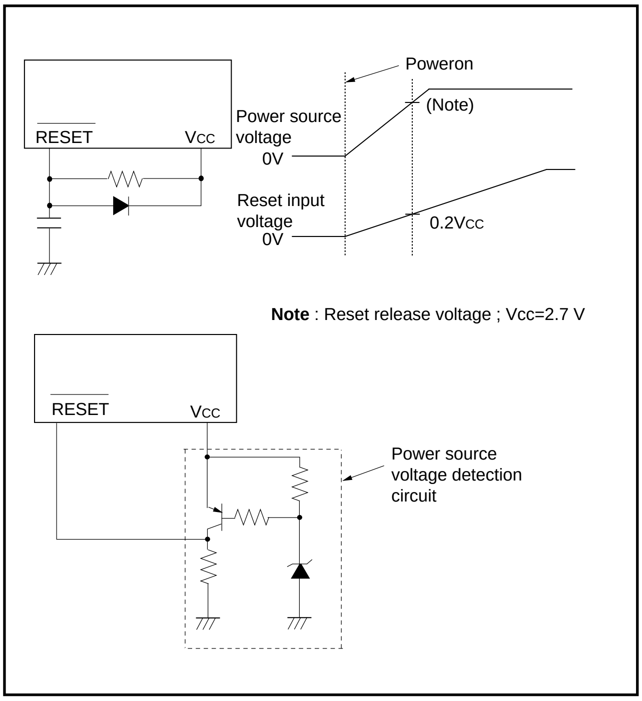


Fig. 76 Reset circuit example

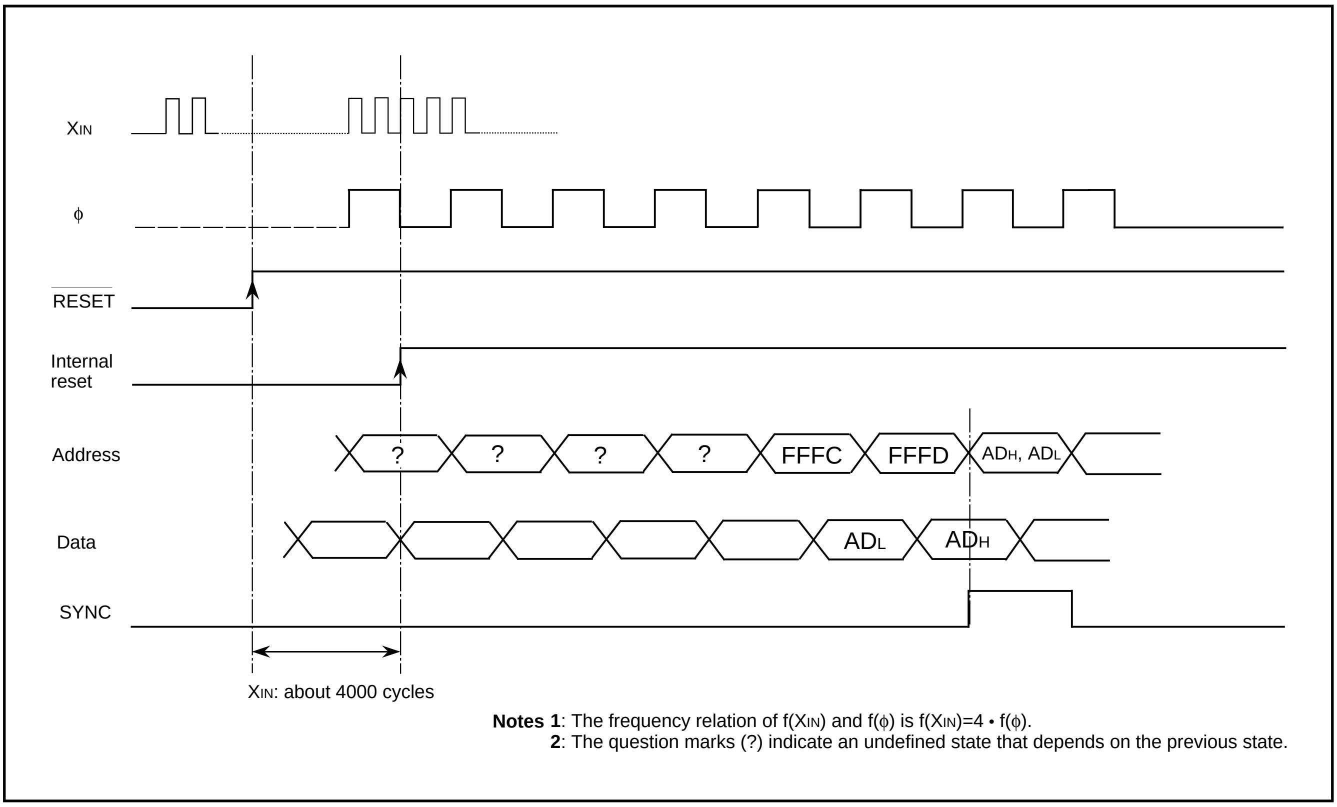


Fig. 77 Reset sequence

	Register	Address	Register contents		Register	Address	Register contents
(1)	Port P0	0000_{16}	00_{16}	(38)	D-A conversion register	$002B_{16}$	00_{16}
(2)	Port P1	0002_{16}	00_{16}	(39)	Timer X (low-order)	$002C_{16}$	FF_{16}
(3)	Port P1 direction register	0003_{16}	00_{16}	(40)	Timer X (high-order)	$002D_{16}$	FF_{16}
(4)	Port P2	0004_{16}	00_{16}	(41)	Timer X mode register 1	$002E_{16}$	00_{16}
(5)	Port P3	0006_{16}	00_{16}	(42)	Timer X mode register 2	$002F_{16}$	00_{16}
(6)	Port P3 direction register	0007_{16}	00_{16}	(43)	Interrupt interval determination register	0030_{16}	00_{16}
(7)	Port P4	0008_{16}	00_{16}	(44)	Interrupt interval determination control register	0031_{16}	00_{16}
(8)	Port P4 direction register	0009_{16}	00_{16}	(45)	AD/DA control register	0032_{16}	10_{16}
(9)	Port P5	$000A_{16}$	00_{16}	(46)	UART control register	0038_{16}	80_{16}
(10)	Port P5 direction register	$000B_{16}$	00_{16}	(47)	Interrupt source switch register	0039_{16}	00_{16}
(11)	Port P6	$000C_{16}$	00_{16}	(48)	Interrupt edge selection register	$003A_{16}$	00_{16}
(12)	Port P6 direction register	$000D_{16}$	00_{16}	(49)	CPU mode register	$003B_{16}$	0 1 0 0 1 0 0 0
(13)	Port P7	$000E_{16}$	00_{16}	(50)	Interrupt request register 1	$003C_{16}$	00_{16}
(14)	Port P7 direction register	$000F_{16}$	00_{16}	(51)	Interrupt request register 2	$003D_{16}$	00_{16}
(15)	Port P8	0010_{16}	00_{16}	(52)	Interrupt control register 1	$003E_{16}$	00_{16}
(16)	Port P8 direction register	0011_{16}	00_{16}	(53)	Interrupt control register 2	$003F_{16}$	00_{16}
(17)	Port P9	0012_{16}	00_{16}	(54)	Serial I/O3 control register	$0EEC_{16}$	00_{16}
(18)	Port P9 direction register	0013_{16}	00_{16}	(55)	Watchdog timer control register	$0EEE_{16}$	$3F_{16}$
(19)	Port PA	0014_{16}	00_{16}	(56)	Pull-up control register 3	$0EEF_{16}$	00_{16}
(20)	Port PA direction register	0015_{16}	00_{16}	(57)	Pull-up control register 1	$0EF0_{16}$	00_{16}
(21)	Port PB	0016_{16}	00_{16}	(58)	Pull-up control register 2	$0EF1_{16}$	00_{16}
(22)	Port PB direction register	0017_{16}	00_{16}	(59)	Port P0 digit output set switch register	$0EF2_{16}$	00_{16}
(23)	Serial I/O1 control register 1	0019_{16}	00_{16}	(60)	Port P2 digit output set switch register	$0EF3_{16}$	00_{16}
(24)	Serial I/O1 control register 2	$001A_{16}$	00_{16}	(61)	FLDC mode register	$0EF4_{16}$	00_{16}
(25)	Serial I/O1 control register 3	$001C_{16}$	00_{16}	(62)	Tdisp time set register	$0EF5_{16}$	00_{16}
(26)	Serial I/O2 control register	$001D_{16}$	00_{16}	(63)	Toff1 time set register	$0EF6_{16}$	FF_{16}
(27)	Serial I/O2 status register	$001E_{16}$	80_{16}	(64)	Toff2 time set register	$0EF7_{16}$	FF_{16}
(28)	Timer 1	0020_{16}	FF_{16}	(65)	Port P4 FLD/Port switch register	$0EF9_{16}$	00_{16}
(29)	Timer 2	0021_{16}	01_{16}	(66)	Port P5 FLD/Port switch register	$0EFA_{16}$	00_{16}
(30)	Timer 3	0022_{16}	FF_{16}	(67)	Port P6 FLD/Port switch register	$0EFB_{16}$	00_{16}
(31)	Timer 4	0023_{16}	FF_{16}	(68)	FLD output control register	$0EFC_{16}$	00_{16}
(32)	Timer 5	0024_{16}	FF_{16}	(69)	Buzzer output control register	$0EFD_{16}$	00_{16}
(33)	Timer 6	0025_{16}	FF_{16}	(70)	Flash memory control register	$0EFE_{16}$	00_{16}
(34)	PWM control register	0026_{16}	00_{16}	(71)	Flash command register	$0EFF_{16}$	00_{16}
(35)	Timer 12 mode register	0028_{16}	00_{16}	(72)	Processor status register	(PS)	× × × × × 1 × ×
(36)	Timer 34 mode register	0029_{16}	00_{16}	(73)	Program counter	(PC_H)	$FFFD_{16}$ contents
(37)	Timer 56 mode register	$002A_{16}$	00_{16}			(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. 78 Internal status at reset

CLOCK GENERATING CIRCUIT

The 38B7 group has two built-in oscillation circuits. An oscillation circuit can be formed by connecting a resonator between XIN and XOUT or 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) > 3 • f(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 the main clock stop bit (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.

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 2 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.

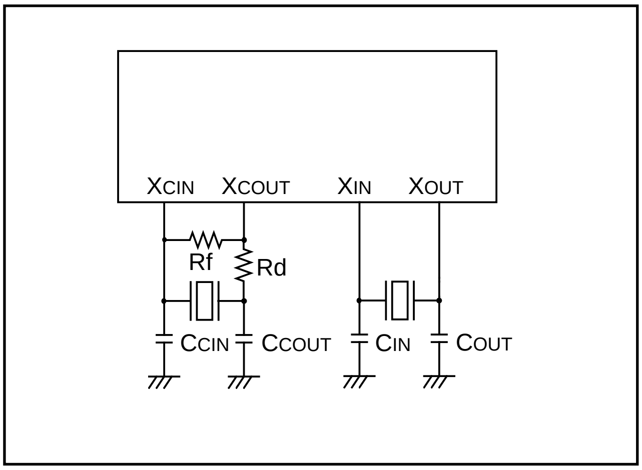


Fig. 79 Ceramic resonator circuit

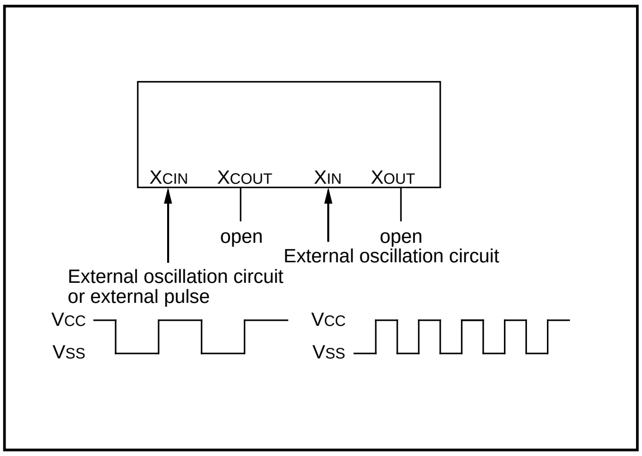
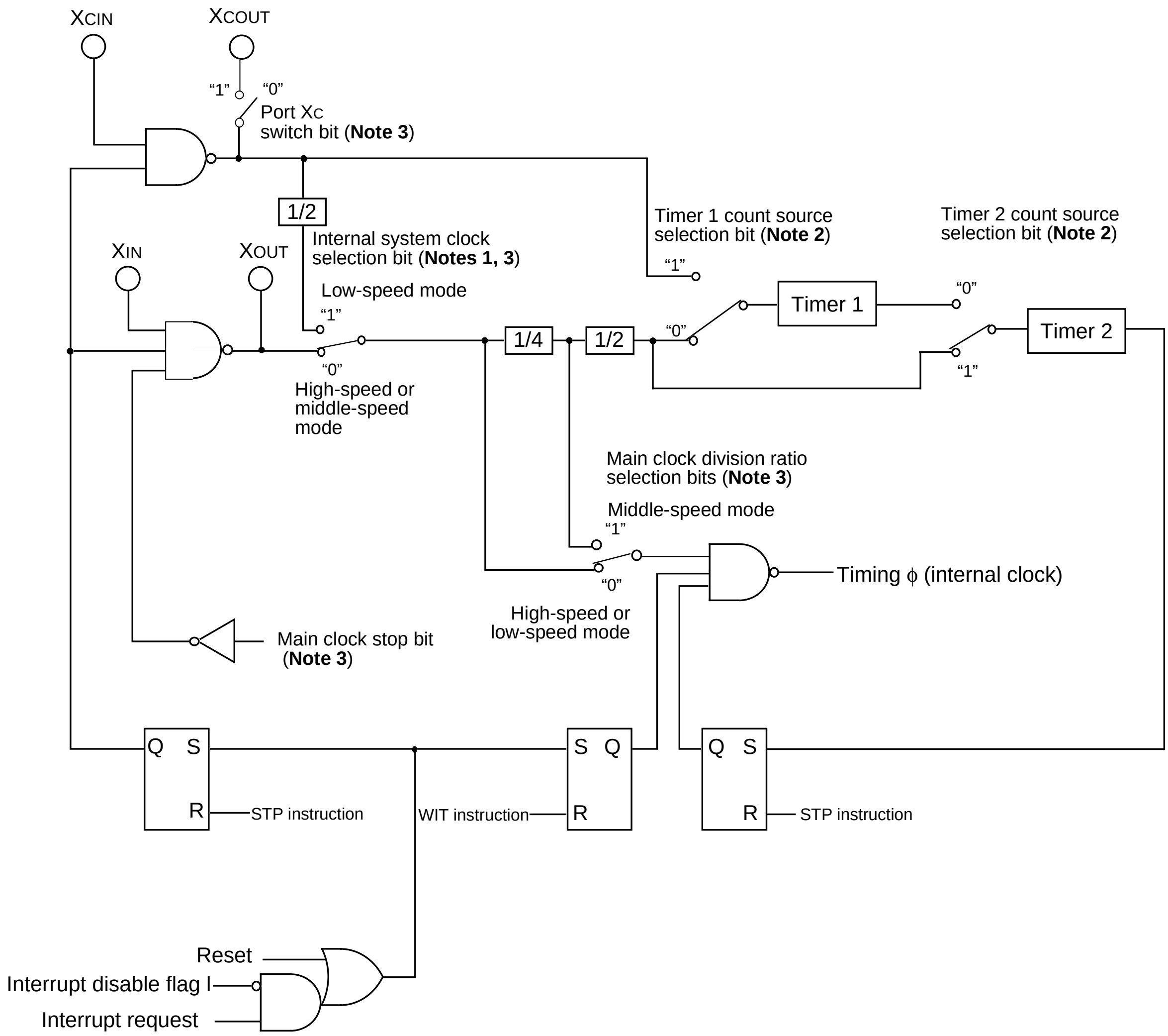
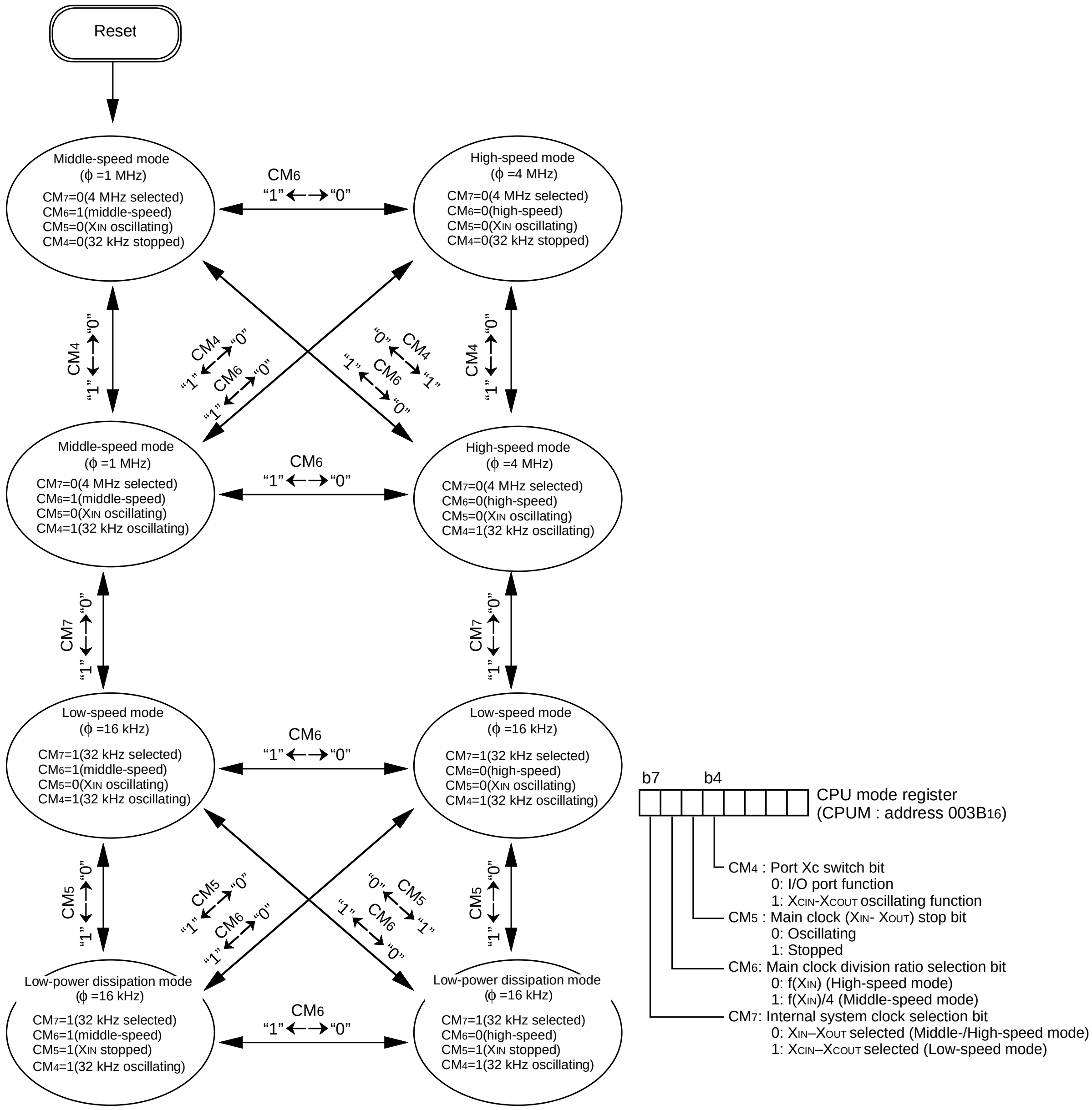


Fig. 80 External clock input circuit



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. 81 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. 82 State transitions of system clock

Power Dissipation Calculating Method

(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 = 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 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: e (= 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.)
= 190 mW

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

Power Dissipation Calculating Example 1

(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 µs = 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 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 = 33.94 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) = <u>217 mW</u>

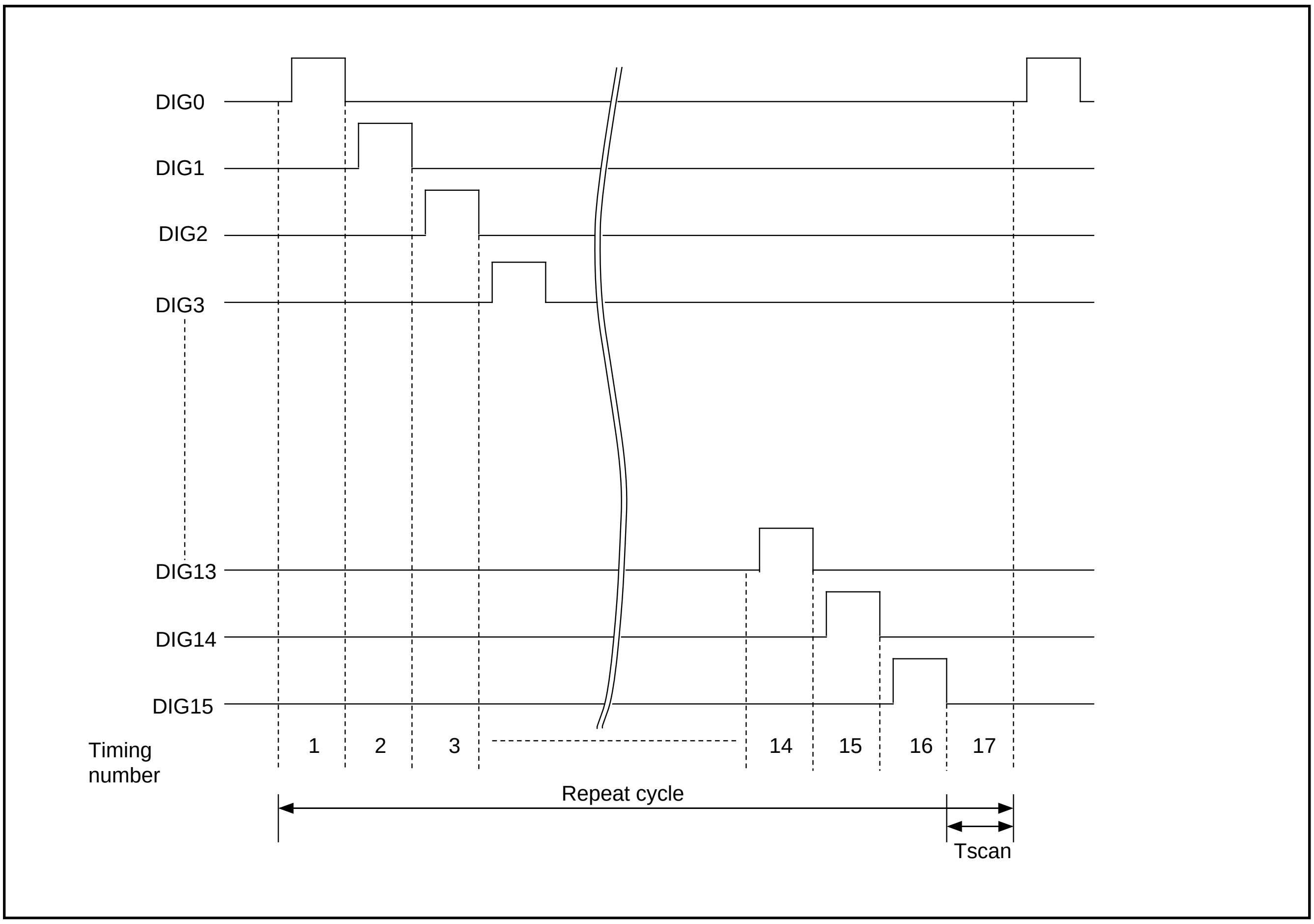


Fig. 83 Digit timing waveform (1)

Power Dissipation Calculating Example 2 (2 or more digits turned ON at the 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 µs = 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
$\{18 \times 12 \times (1 - 1/16) \times 2\} / 11 = 36.82$ mW

(2) Segment pin power dissipation
$\{3 \times 114 \times (1 - 1/16) \times 2\} / 11 = 58.30$ mW

(3) Pull-down resistor power dissipation (digit)
$[\{45 - 2\}^2 / 48 \times (12 \times 10 / 12) \times (1 - 1/16)] / 11 = 32.84$ mW

(4) Pull-down resistor power dissipation (segment)
$[\{45 - 2\}^2 / 48 \times (114 \times 22 / 24) \times (1 - 1/16)] / 11 = 343.08$ mW

(5) Internal circuit power dissipation (CPU, ROM, RAM etc.)
= 75 mW

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

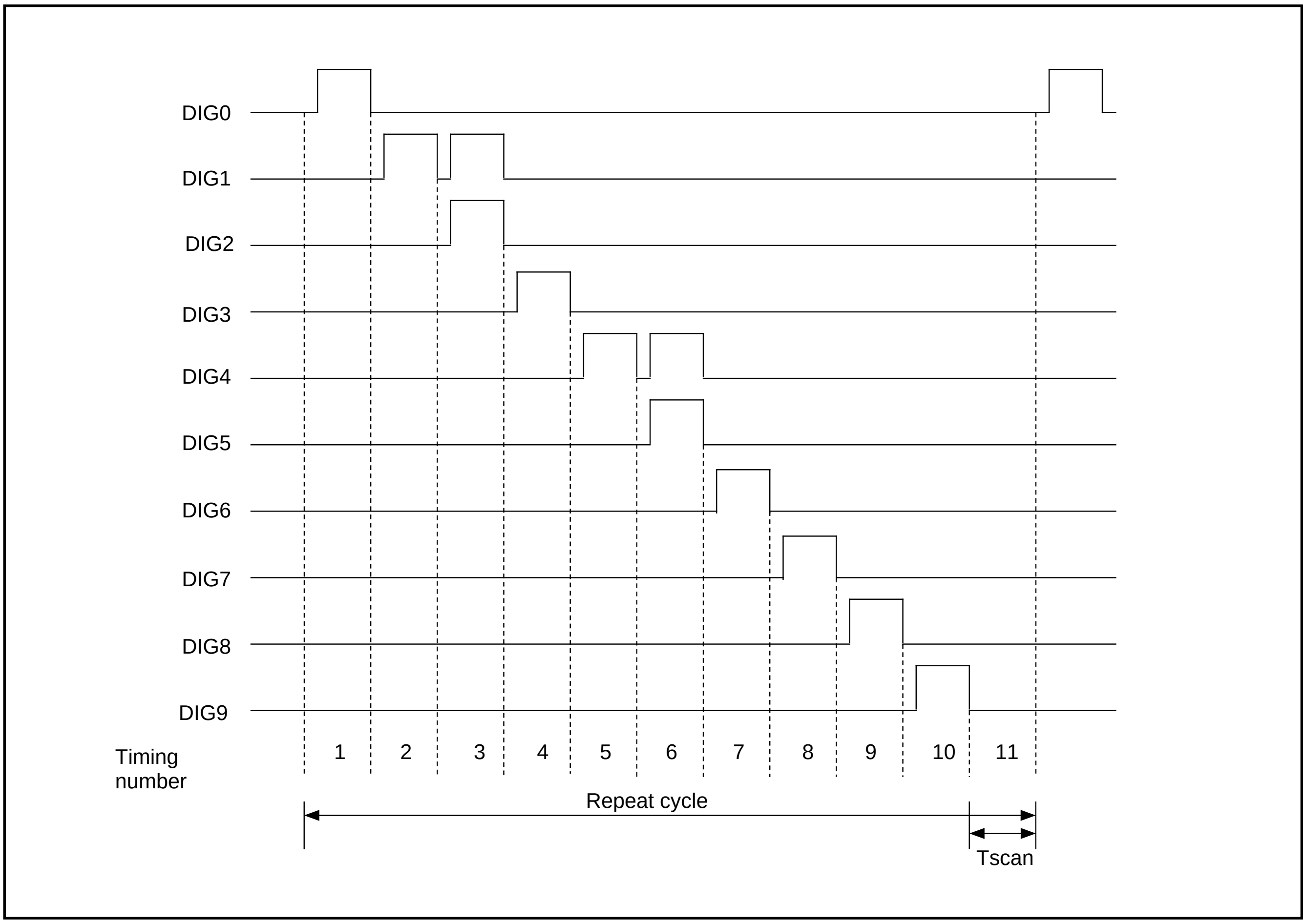


Fig. 84 Digit timing waveform (2)

FLASH MEMORY MODE

The M38B79FF has the flash memory mode in addition to the normal operation mode (microcomputer mode). The user can use this mode to perform read, program, and erase operations for the internal flash memory.

The M38B79FF has three modes the user can choose: the parallel input/output and serial input/output mode, where the flash memory is handled by using the external programmer, and the CPU reprogramming mode, where the flash memory is handled by the central processing unit (CPU). The following explains these modes.

(1) Flash memory mode 1 (parallel I/O mode)

The parallel I/O mode can be selected by connecting wires as shown in Figures 85 and supplying power to the V_{CC} and V_{PP} pins. In this mode, the M38B79FF operates as an equivalent of MITSUBISHI's CMOS flash memory M5M28F101. However, because the M38B79FF's internal memory has a capacity of 60 Kbytes, programming is available for addresses 01000_{16} to $0FFFF_{16}$, and make sure that the data in addresses 00000_{16} to $00FFF_{16}$ and addresses 10000_{16} to $1FFFF_{16}$ are FF_{16}. Note also that the M38B79FF does not contain a facility to read out a device identification code by applying a high voltage to address input (A_9). Be careful not to erratically set program conditions when using a general-purpose PROM programmer.

Table 12 shows the pin assignments when operating in the parallel input/output mode.

Table 12 Pin assignments of M38B79FF when operating in the parallel input/output mode

	M38B79FF	M5M28F101
V_{CC}	V_{CC}	V_{CC}
V_{PP}	CNV_{SS}	V_{PP}
V_{SS}	V_{SS}	V_{SS}
Address input	Ports P0, P1, $P3_1$	A_0–A_{16}
Data I/O	Port P2	D_0–D_7
$\overline{CE}$	$P3_6$	$\overline{CE}$
$\overline{OE}$	$P3_7$	$\overline{OE}$
$\overline{WE}$	$P3_3$	$\overline{WE}$

Functional Outline (parallel input/output mode)

In the parallel input/output mode, the M38B79FF allow the user to choose an operation mode between the read-only mode and the read/write mode (software command control mode) depending on the voltage applied to the V_{PP} pin. When $V_{PP} = V_{PPL}$, the read-only mode is selected, and the user can choose one of three states (e.g., read, output disable, or standby) depending on inputs to the $\overline{CE}$, $\overline{OE}$, and $\overline{WE}$ pins. When $V_{PP} = V_{PPH}$, the read/write mode is selected, and the user can choose one of four states (e.g., read, output disable, standby, or write) depending on inputs to the $\overline{CE}$, $\overline{OE}$, and $\overline{WE}$ pins. Table 13 shows assignment states of control input and each state.

● Read

The microcomputer enters the read state by driving the $\overline{CE}$, and $\overline{OE}$ pins low and the $\overline{WE}$ pin high; and the contents of memory corresponding to the address to be input to address input pins (A_0–A_{16}) are output to the data input/output pins (D_0–D_7).

● Output disable

The microcomputer enters the output disable state by driving the $\overline{CE}$ pin low and the $\overline{WE}$ and $\overline{OE}$ pins high; and the data input/output pins enter the floating state.

● Standby

The microcomputer enters the standby state by driving the $\overline{CE}$ pin high. The M38B79FF is placed in a power-down state consuming only a minimal supply current. At this time, the data input/output pins enter the floating state.

● Write

The microcomputer enters the write state by driving the V_{PP} pin high ($V_{PP} = V_{PPH}$) and then the $\overline{WE}$ pin low when the $\overline{CE}$ pin is low and the $\overline{OE}$ pin is high. In this state, software commands can be input from the data input/output pins, and the user can choose program or erase operation depending on the contents of this software command.

Table 13 Assignment states of control input and each state

Mode	State \ Pin	$\overline{CE}$	$\overline{OE}$	$\overline{WE}$	V_{PP}	Data I/O
Read-only	Read	V_{IL}	V_{IL}	V_{IH}	V_{PPL}	Output
	Output disable	V_{IL}	V_{IH}	V_{IH}	V_{PPL}	Floating
	Standby	V_{IH}	×	×	V_{PPL}	Floating
Read/Write	Read	V_{IL}	V_{IL}	V_{IH}	V_{PPH}	Output
	Output disable	V_{IL}	V_{IH}	V_{IH}	V_{PPH}	Floating
	Standby	V_{IH}	×	×	V_{PPH}	Floating
	Write	V_{IL}	V_{IH}	V_{IL}	V_{PPH}	Input

Note: × can be V_{IL} or V_{IH}.

Table 14 Pin description (flash memory parallel I/O mode)

Pin	Name	Input /Output	Functions
V_{CC}, V_{SS}	Power supply	—	Supply 5 V ± 10 % to V_{CC} and 0 V to V_{SS}.
CNV_{SS}	V_{PP} input	Input	Connect to 5 V ± 10 % in read-only mode, connect to 11.7 V to 12.6 V in read/write mode.
$\overline{RESET}$	Reset input	Input	Connect to V_{SS}.
X_{IN}	Clock input	Input	Connect a ceramic resonator between X_{IN} and X_{OUT}.
X_{OUT}	Clock output	Output	
AV_{SS}	Analog supply input	—	Connect to V_{SS}.
V_{REF}	Reference voltage input	Input	Connect to V_{SS}.
$P0_0$–$P0_7$	Address input (A_0–A_7)	Input	Port P0 functions as 8-bit address input (A_0–A_7).
$P1_0$–$P1_7$	Address input (A_8–A_{15})	Input	Port P1 functions as 8-bit address input (A_8–A_{15}).
$P2_0$–$P2_7$	Data I/O (D_0–D_7)	I/O	Function as 8-bit data's I/O pins (D_0–D_7). Connect them to Vss through each resistor of 6.8 kΩ.
$P3_0$–$P3_7$	Control signal input	Input	$P3_7$, $P3_6$ and $P3_3$ function as the $\overline{OE}$, $\overline{CE}$ and $\overline{WE}$ input pins respectively. $P3_1$ functions as the A_{16} input pin. Connect $P3_0$ and $P3_2$ to V_{SS}. Input “H” or “L” to $P3_4$, $P3_5$, or keep them open.
$P4_0$–$P4_7$	Input port P4	Input	Input “H” or “L”, or keep them open.
$P5_0$–$P5_7$	Input port P5	Input	Input “H” or “L”, or keep them open.
$P6_0$–$P6_7$	Input port P6	Input	Connect $P6_4$ and $P6_6$ to V_{SS}. Input “H” or “L” to $P6_0$–$P6_3$, $P6_5$, $P6_7$, or keep them open.
$P7_0$–$P7_7$	Input port P7	Input	Input “H” or “L”, or keep them open.
$P8_0$–$P8_3$	Input port P8	Input	Input “H” or “L”, or keep them open.
$P9_0$–$P9_7$	Input port P9	Input	Input “H” or “L”, or keep them open.
PA_0–PA_7	Input port PA	Input	Input “H” or “L”, or keep them open.
PB_0–PB_6	Input port PB	Input	Input “H” or “L”, or keep them open.
V_{EE}	Pull-down power supply		Keep this open.

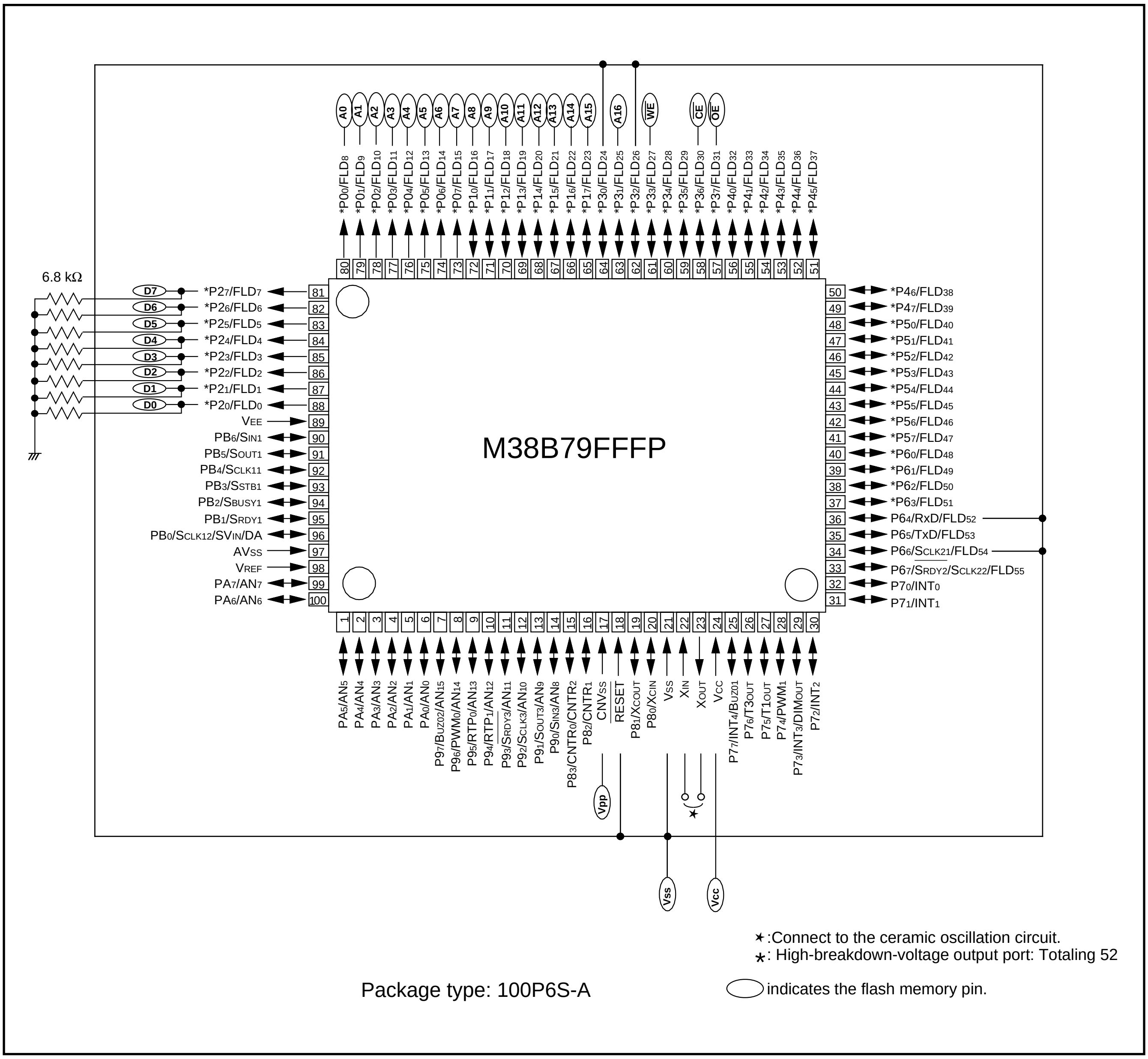


Fig. 85 Pin connection of M38B79FF when operating in parallel input/output mode

Read-only Mode

The microcomputer enters the read-only mode by applying VPPL to the VPP pin. In this mode, the user can input the address of a memory location to be read and the control signals at the timing shown in Figure 86, and the M38B79FF will output the contents of the user's specified address from data I/O pin to the external. In this mode, the user cannot perform any operation other than read.

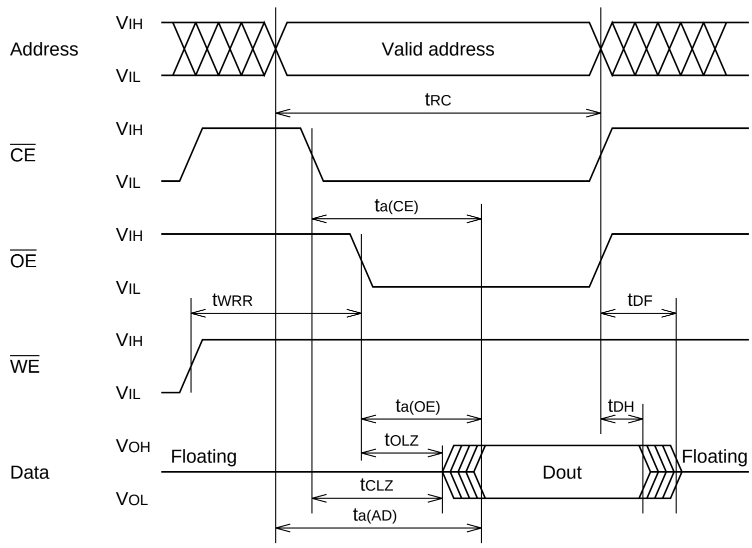


Fig. 86 Read timing

Read/Write Mode

The microcomputer enters the read/write mode by applying VPPH to the VPP pin. In this mode, the user must first input a software command to choose the operation (e. g., read, program, or erase) to be performed on the flash memory (this is called the first cycle), and then input the information necessary for execution of the command (e.g, address and data) and control signals (this is called the second cycle). When this is done, the M38B79FF executes the specified operation.

Table 15 shows the software commands and the input/output information in the first and the second cycles. The input address is latched internally at the falling edge of the $\overline{WE}$ input; software commands and other input data are latched internally at the rising edge of the $\overline{WE}$ input.

The following explains each software command. Refer to Figures 87 to 89 for details about the signal input/output timings.

Table 15 Software command (parallel input/output mode)

Symbol	First cycle		Second cycle	
	Address input	Data input	Address input	Data I/O
Read	×	00_{16}	Read address	Read data (Output)
Program	×	40_{16}	Program address	Program data (Input)
Program verify	×	$C0_{16}$	×	Verify data (Output)
Erase	×	20_{16}	×	20_{16} (Input)
Erase verify	Verify address	$A0_{16}$	×	Verify data (Output)
Reset	×	FF_{16}	×	FF_{16} (Input)
Device identification	×	90_{16}	ADI	DDI (Output)

Note: ADI = Device identification address : manufacturer's code 0000_{16}, device code 0001_{16}
DDI = Device identification data : manufacturer's code $1C_{16}$, device code $D0_{16}$
× can be VIL or VIH.

● **Read command**

The microcomputer enters the read mode by inputting command code "00_{16}" in the first cycle. The command code is latched into the internal command latch at the rising edge of the $\overline{WE}$ input. When the address of a memory location to be read is input in the second cycle, with control signals input at the timing shown in Figure 87, the M38B79FF outputs the contents of the specified address from the data I/O pins to the external.

The read mode is retained until any other command is latched into the command latch. Consequently, once the M38B79FF enters the read mode, the user can read out the successive memory contents simply by changing the input address and executing the second cycle only. Any command other than the read command must be input beginning from its command code over again each time the user execute it. The contents of the command latch immediately after power-on is 00_{16}.

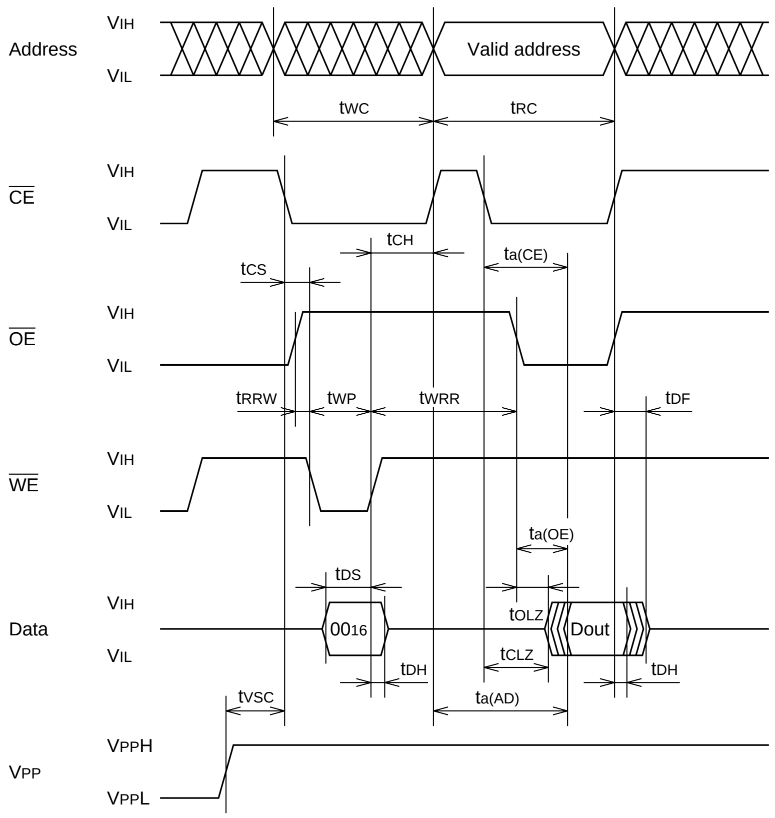


Fig. 87 Timings during reading

● **Program command**

The microcomputer enters the program mode by inputting command code "40₁₆" in the first cycle. The command code is latched into the internal command latch at the rising edge of the $\overline{WE}$ input. When the address which indicates a program location and data is input in the second cycle, the M38B79FF internally latches the address at the falling edge of the $\overline{WE}$ input and the data at the rising edge of the $\overline{WE}$ input. The M38B79FF starts programming at the rising edge of the $\overline{WE}$ input in the second cycle and finishes programming within 10 µs as measured by its internal timer. Programming is performed in units of bytes.

Note: A programming operation is not completed by executing the program command once. Always be sure to execute a program verify command after executing the program command. When the failure is found in this verification, the user must repeatedly execute the program command until the pass. Refer to Figure 90 for the programming flowchart.

● **Program verify command**

The microcomputer enters the program verify mode by inputting command code "C0₁₆" in the first cycle. This command is used to verify the programmed data after executing the program command. The command code is latched into the internal command latch at the rising edge of the $\overline{WE}$ input. When control signals are input in the second cycle at the timing shown in Figure 88, the M38B79FF outputs the programmed address's contents to the external. Since the address is internally latched when the program command is executed, there is no need to input it in the second cycle.

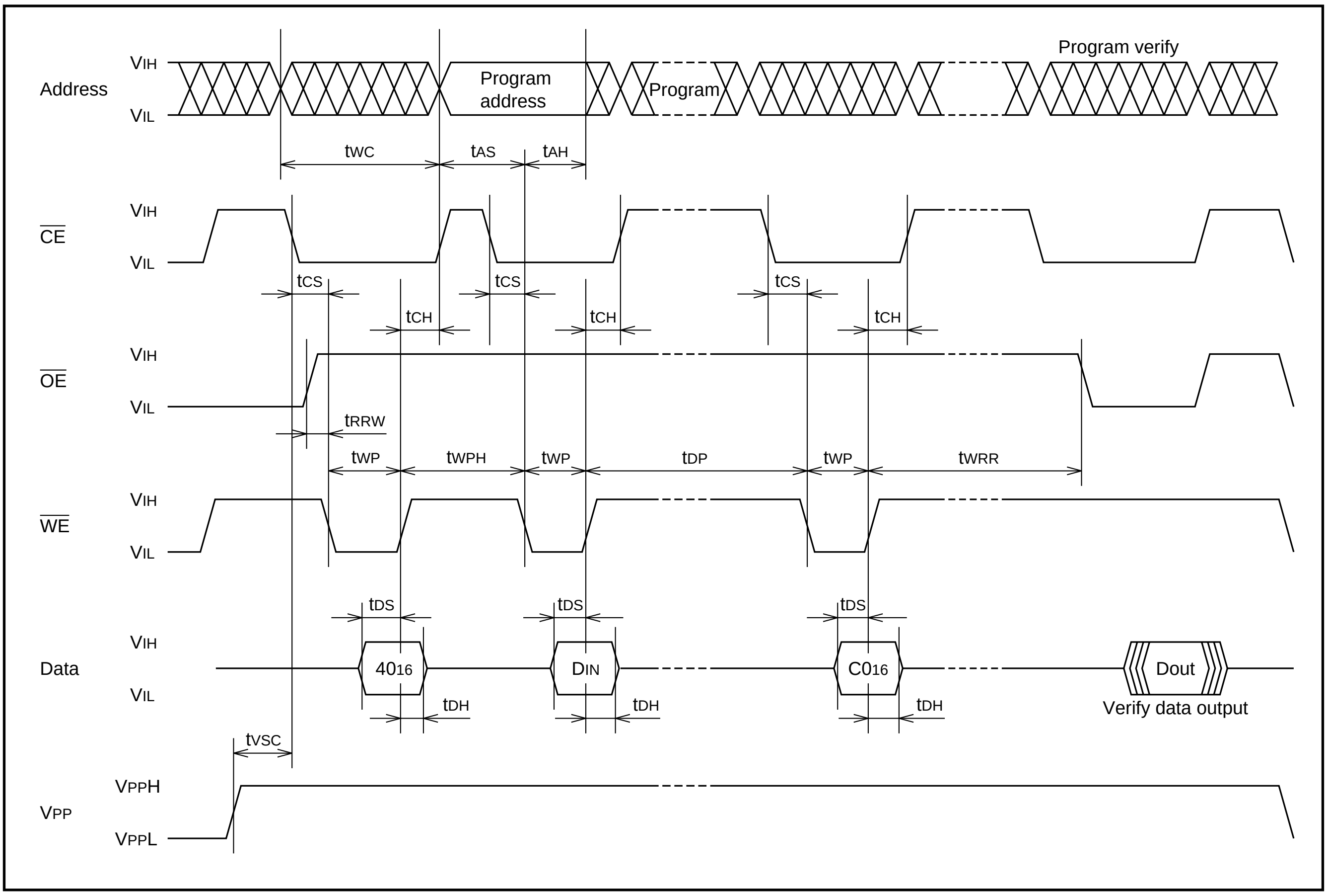


Fig. 88 Input/output timings during programming (Verify data is output at the same timing as for read.)

● Erase command

The erase command is executed by inputting command code 20_{16} in the first cycle and command code 20_{16} again in the second cycle. The command code is latched into the internal command latch at the rising edges of the $\overline{WE}$ input in the first cycle and in the second cycle, respectively. The erase operation is initiated at the rising edge of the $\overline{WE}$ input in the second cycle, and the memory contents are collectively erased within 9.5 ms as measured by the internal timer. Note that data 00_{16} must be written to all memory locations before executing the erase command.

Note: An erase operation is not completed by executing the erase command once. Always be sure to execute an erase verify command after executing the erase command. When the failure is found in this verification, the user must repeatedly execute the erase command until the pass. Refer to Figure 90 for the erase flowchart.

● Erase verify command

The user must verify the contents of all addresses after completing the erase command. The microcomputer enters the erase verify mode by inputting the verify address and command code $A0_{16}$ in the first cycle. The address is internally latched at the falling edge of the $\overline{WE}$ input, and the command code is internally latched at the rising edge of the $\overline{WE}$ input. When control signals are input in the second cycle at the timing shown in Figure 89, the M38B79FF outputs the contents of the specified address to the external.

Note: If any memory location where the contents have not been erased is found in the erase verify operation, execute the operation of “erase → erase verify” over again. In this case, however, the user does not need to write data 00_{16} to memory locations before erasing.

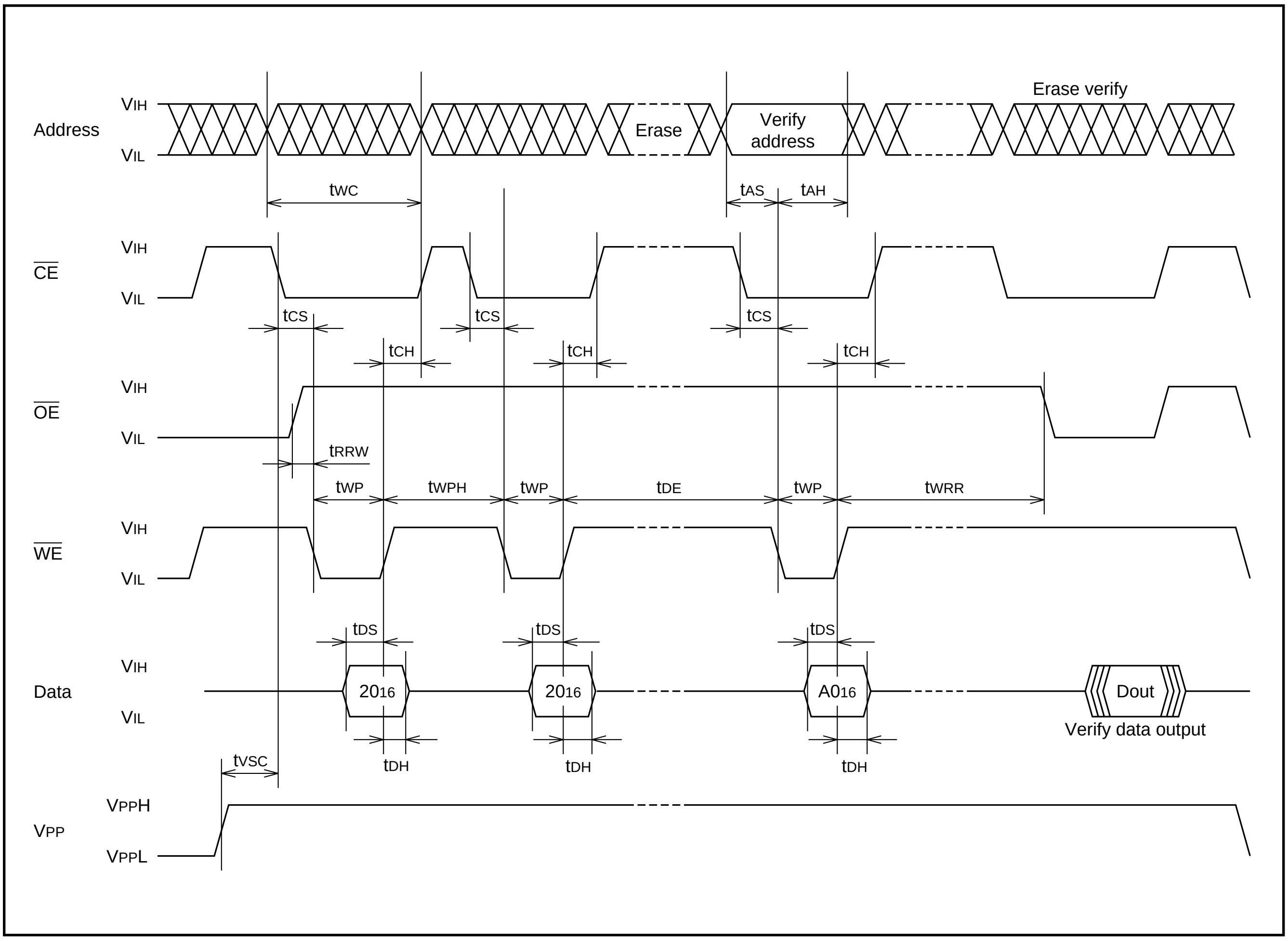


Fig. 89 Input/output timings during erasing (verify data is output at the same timing as for read.)

● **Reset command**

The reset command provides a means of stopping execution of the erase or program command safely. If the user inputs command code FF_{16} in the second cycle after inputting the erase or program command in the first cycle and again input command code FF_{16} in the third cycle, the erase or program command is disabled (i.e., reset), and the M38B79FF is placed in the read mode. If the reset command is executed, the contents of the memory does not change.

● **Device identification code command**

By inputting command code 90_{16} in the first cycle, the user can read out the device identification code. The command code is latched into the internal command latch at the rising edge of the $\overline{WE}$ input. At this time, the user can read out manufacture's code $1C_{16}$ (i.e., MITSUBISHI) by inputting 0000_{16} to the address input pins in the second cycle; the user can read out device code $D0_{16}$ (i. e., 1M-bit flash memory) by inputting 0001_{16}.

These command and data codes are input/output at the same timing as for read.

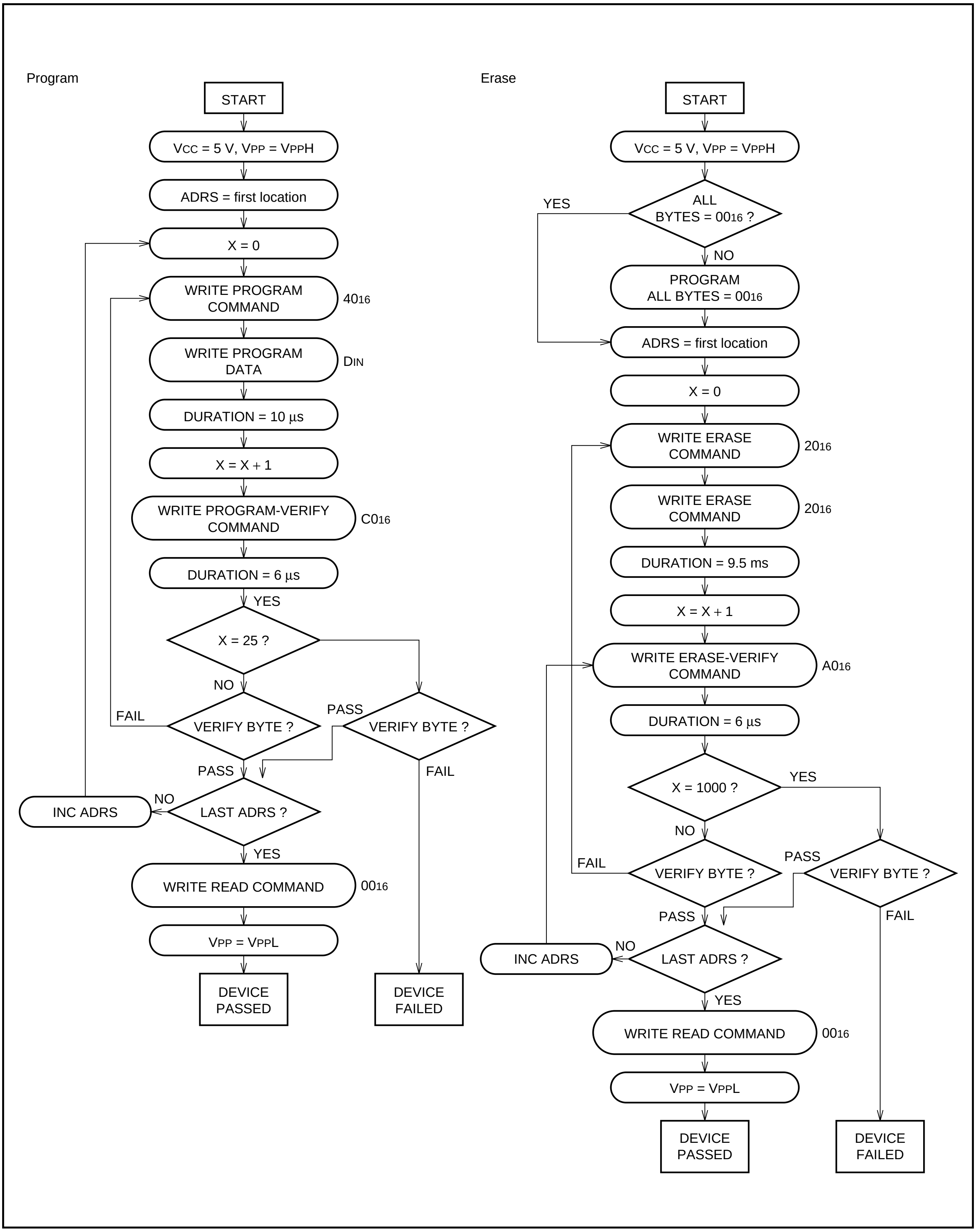


Fig. 90 Programming/Erasing algorithm flow chart

Table 16 DC ELECTRICAL CHARACTERISTICS (T_a = 25 °C, V_{CC} = 5 V ± 10 %, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
I_{SB1}	V_{CC} supply current (at standby)	V_{CC} = 5.5 V, $\overline{CE}$ = V_{IH}			1	mA
I_{SB2}		V_{CC} = 5.5 V, $\overline{CE}$ = V_{CC} ± 0.2 V			100	µA
I_{CC1}	V_{CC} supply current (at read)	V_{CC} = 5.5 V, $\overline{CE}$ = V_{IL}, t_{RC} = 150 ns, I_{OUT} = 0 mA			15	mA
I_{CC2}	V_{CC} supply current (at program)	V_{PP} = V_{PPH}			15	mA
I_{CC3}	V_{CC} supply current (at erase)	V_{PP} = V_{PPH}			15	mA
I_{PP1}	V_{PP} supply current (at read)	$0 \leq V_{PP} \leq V_{CC}$			10	µA
		$V_{CC} < V_{PP} \leq V_{CC}$ + 1.0 V			100	µA
		V_{PP} = V_{PPH}			100	µA
I_{PP2}	V_{PP} supply current (at program)	V_{PP} = V_{PPH}			30	mA
I_{PP3}	V_{PP} supply current (at erase)	V_{PP} = V_{PPH}			30	mA
V_{IL}	"L" input voltage		0		0.2Vcc	V
V_{IH}	"H" input voltage		0.52Vcc		V_{CC}	V
V_{OH1}	"H" output voltage	I_{OH} = –400 µA	2.4			V
V_{OH2}		I_{OH} = –100 µA	V_{CC} –0.4			V
V_{PPL}	V_{PP} supply voltage (read only)		V_{CC}		V_{CC} + 1.0	V
V_{PPH}	V_{PP} supply voltage (read/write)		11.7	12.0	12.6	V

AC ELECTRICAL CHARACTERISTICS (T_a = 25 °C, V_{CC} = 5 V ± 10 %, unless otherwise noted)

Table 17 Read-only mode

Symbol	Parameter	Limits		Unit
		Min.	Max.	
t_{RC}	Read cycle time	500		ns
$t_{a(AD)}$	Address access time		500	ns
$t_{a(CE)}$	$\overline{CE}$ access time		500	ns
$t_{a(OE)}$	$\overline{OE}$ access time		200	ns
t_{CLZ}	Output enable time (after $\overline{CE}$)	0		ns
t_{OLZ}	Output enable time (after $\overline{OE}$)	0		ns
t_{DF}	Output floating time (after $\overline{OE}$)		70	ns
t_{DH}	Output valid time (after $\overline{CE}$, $\overline{OE}$, address)	0		ns
t_{WRR}	Write recovery time (before read)	6		µs

Table 18 Read/Write mode

Symbol	Parameter	Limits		Unit
		Min.	Max.	
t_{WC}	Write cycle time	300		ns
t_{AS}	Address set up time	0		ns
t_{AH}	Address hold time	120		ns
t_{DS}	Data setup time	100		ns
t_{DH}	Data hold time	20		ns
t_{WRR}	Write recovery time (before read)	6		µs
t_{RRW}	Read recovery time (before write)	0		µs
t_{CS}	$\overline{CE}$ setup time	40		ns
t_{CH}	$\overline{CE}$ hold time	0		ns
t_{WP}	Write pulse width	120		ns
t_{WPH}	Write pulse waiting time	40		ns
t_{DP}	Program time	10		µs
t_{DE}	Erase time	9.5		ms
t_{VSC}	V_{PP} setup time	1		µs

Note: Read timing of Read/Write mode is same as Read-only mode.

(2) Flash memory mode 2 (serial I/O mode)

The M38B79FF has a function to serially input/output the software commands, addresses, and data required for operation on the internal flash memory (e. g., read, program, and erase) using only a few pins. This is called the serial I/O (input/output) mode. This mode can be selected by driving the SDA (serial data input/output), SCLK (serial clock input), and OE pins high after connecting wires as shown in Figures 91 and powering on the VCC pin and then applying VPPH to the VPP pin.

In the serial I/O mode, the user can use six types of software commands: read, program, program verify, erase, erase verify and error check.

Serial input/output is accomplished synchronously with the clock, beginning from the LSB (LSB first).

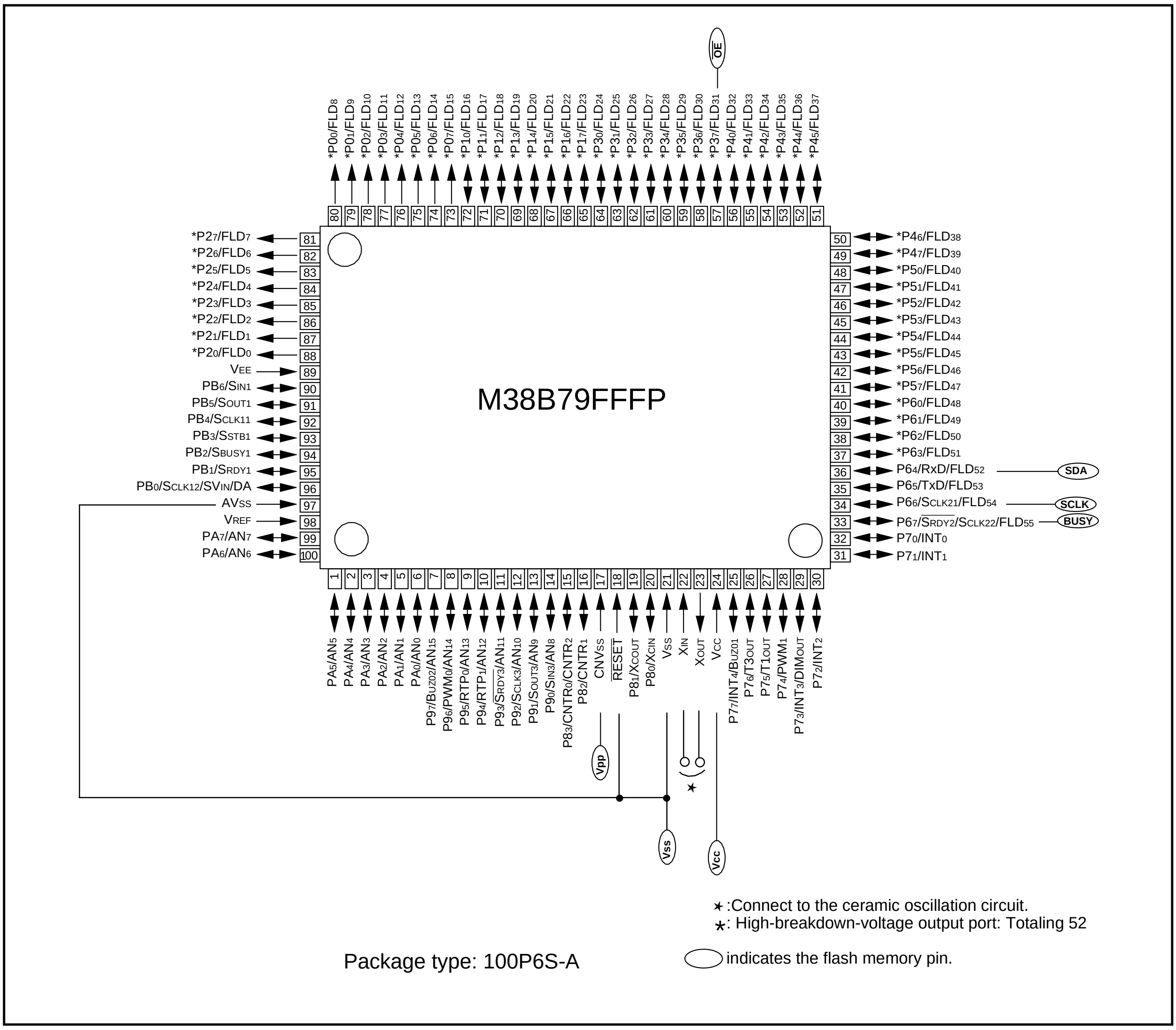


Fig. 91 Pin connection of M38B79FF when operating in serial I/O mode

Table 19 Pin description (flash memory serial I/O mode)

Pin	Name	Input /Output	Functions
V_{CC}, V_{SS}	Power supply	—	Supply 5 V ± 10 % to V_{CC} and 0 V to V_{SS}.
CNV_{SS}	V_{PP} input	Input	Connect to 11.7 V to 12.6 V.
$\overline{RESET}$	Reset input	Input	Connect to V_{SS}.
X_{IN}	Clock input	Input	Connect a ceramic resonator between X_{IN} and X_{OUT}.
X_{OUT}	Clock output	Output	
AV_{SS}	Analog supply input	—	Connect to V_{SS}.
V_{REF}	Reference voltage input	Input	Input an arbitrary level between the range of V_{SS} and V_{CC}.
$P0_0$–$P0_7$	Input port P0	Input	Input “H” or “L”, or keep them open.
$P1_0$–$P1_7$	Input port P1	Input	Input “H” or “L”, or keep them open.
$P2_0$–$P2_7$	Input port P2	Input	Input “H” or “L”, or keep them open.
$P3_0$–$P3_6$	Input port P3	Input	Input “H” or “L”, or keep them open.
$P3_7$	Control signal input	Input	$\overline{OE}$ input pin
$P4_0$–$P4_7$	Input port P4	Input	Input “H” or “L” , or keep them open.
$P5_0$–$P5_7$	Input port P5	Input	Input “H” or “L”, or keep them open.
$P6_0$–$P6_3$, $P6_5$	Input port P6	Input	Input “H” or “L” to $P6_0$–$P6_3$, $P6_5$, or keep them open.
$P6_4$	SDA I/O	I/O	This pin is for serial data I/O.
$P6_6$	SCLK input	Input	This pin is for serial clock input.
$P6_7$	BUSY output	Output	This pin is for BUSY signal output.
$P7_0$–$P7_7$	Input port P7	Input	Input “H” or “L”, or keep them open.
$P8_0$–$P8_3$	Input port P8	Input	Input “H” or “L”, or keep them open.
$P9_0$–$P9_7$	Input port P9	Input	Input “H” or “L”, or keep them open.
PA_0–PA_7	Input port PA	Input	Input “H” or “L”, or keep them open.
PB_0–PB_6	Input port PB	Input	Input “H” or “L”, or keep them open.
V_{EE}	Pull-down power supply		Keep this open.

Functional Outline (serial I/O mode)

In the serial I/O mode, data is transferred synchronously with the clock using serial input/output. The input data is read from the SDA pin into the internal circuit synchronously with the rising edge of the serial clock pulse; the output data is output from the SDA pin synchronously with the falling edge of the serial clock pulse.

Data is transferred in units of eight bits.

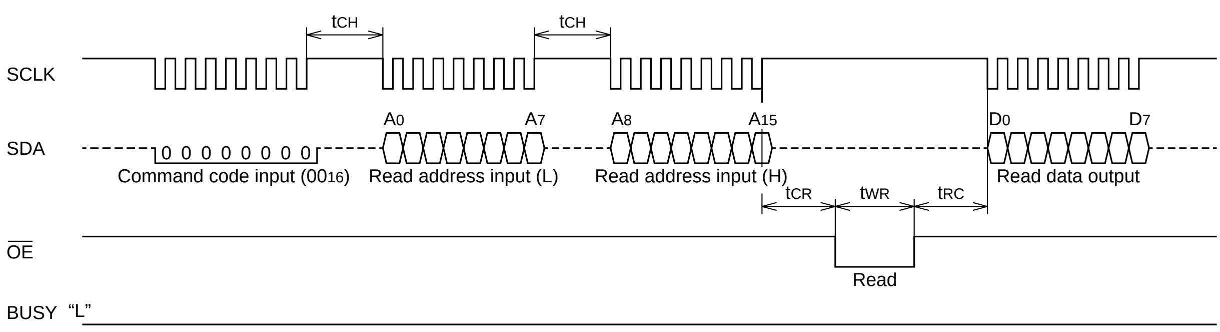
In the first transfer, the user inputs the command code. This is followed by address input and data input/output according to the contents of the command. Table 20 shows the software commands used in the serial I/O mode. The following explains each software command.

Table 20 Software command (serial I/O mode)

Command \ Number of transfers	First command code input	Second	Third	Fourth
Read	00_{16}	Read address L (Input)	Read address H (Input)	Read data (Output)
Program	40_{16}	Program address L (Input)	Program address H (Input)	Program data (Input)
Program verify	$C0_{16}$	Verify data (Output)	————	————
Erase	20_{16}	20_{16} (Input)	————	————
Erase verify	$A0_{16}$	Verify address L (Input)	Verify address H (Input)	Verify data (Output)
Error check	80_{16}	Error code (Output)	————	————

● **Read command**

Input command code 00_{16} in the first transfer. Proceed and input the low-order 8 bits and the high-order 8 bits of the address and pull the $\overline{OE}$ pin low. When this is done, the M38B79FF reads out the contents of the specified address, and then latchs it into the internal data latch. When the $\overline{OE}$ pin is released back high and serial clock is input to the SCLK pin, the read data that has been latched into the data latch is serially output from the SDA pin.



Note : When outputting the read data, the SDA pin is switched for output at the first falling edge of SCLK. The SDA pin is placed in the floating state during the period of $t_{h(C-E)}$ after the last rising edge of SCLK (at the 8th bit).

Fig. 92 Timings during reading

● **Program command**

Input command code 40_{16} in the first transfer. Proceed and input the low-order 8 bits and the high-order 8 bits of the address and then program data. Programming is initiated at the last rising edge of the serial clock during program data transfer. The BUSY pin is driven high during program operation. Programming is completed within 10 µs as measured by the internal timer, and the BUSY pin is pulled low.

Note : A programming operation is not completed by executing the program command once. Always be sure to execute a program verify command after executing the program command. When the failure is found in the verification, the user must repeatedly execute the program command until the pass in the verification. Refer to Figure 90 for the programming flowchart.

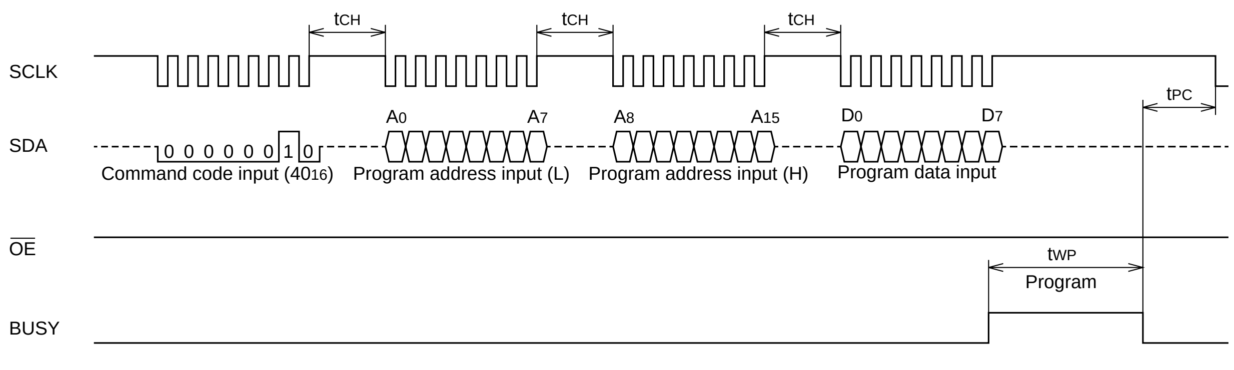
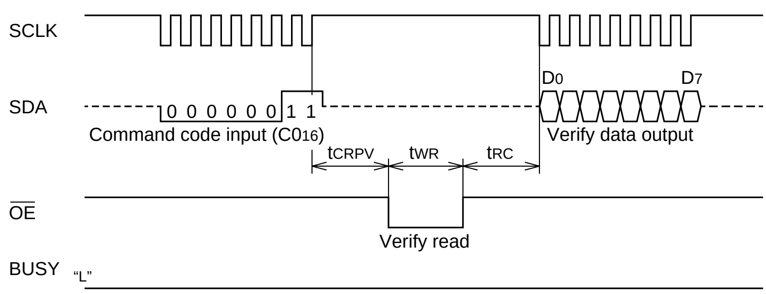


Fig. 93 Timings during programming

● **Program verify command**

Input command code $C0_{16}$ in the first transfer. Proceed and drive the $\overline{OE}$ pin low. When this is done, The M38B79FF verify-reads the programmed address's contents, and then latchs it into the internal data latch. When the $\overline{OE}$ pin is released back high and serial clock is input to the SCLK pin, the verify data that has been latched into the data latch is serially output from the SDA pin.



Note: When outputting the verify data, the SDA pin is switched for output at the first falling edge of SCLK. The SDA pin is placed in the floating state during the period of $t_{h(C-E)}$ after the last rising edge of SCLK (at the 8th bit).

Fig. 94 Timings during program verify

● **Erase command**

Input command code 20_{16} in the first transfer and command code 20_{16} again in the second transfer. When this is done, the M38B79FF executes an erase command. Erase is initiated at the last rising edge of the serial clock. The BUSY pin is driven high during the erase operation. Erase is completed within 9.5 ms as measured by the internal timer, and the BUSY pin is pulled low. Note that data 00_{16} must be written to all memory locations before executing the erase command.

Note: A erase operation is not completed by executing the erase command once. Always be sure to execute a erase verify command after executing the erase command. When the failure is found in the verification, the user must repeatedly execute the erase command until the pass in the verification. Refer to Figure 90 for the erase flowchart.

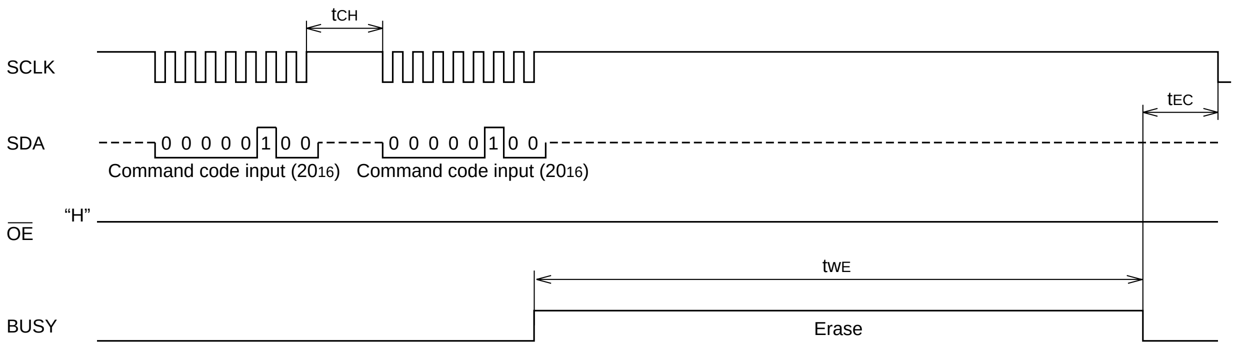
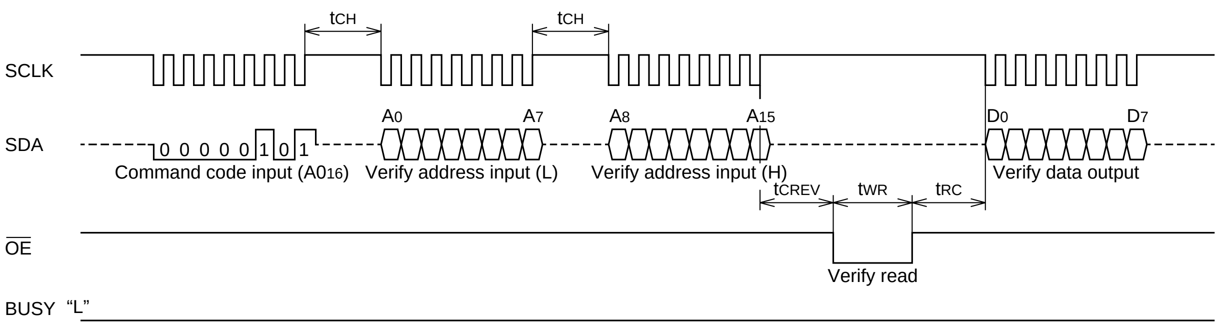


Fig. 95 Timings at erasing

● **Erase verify command**

The user must verify the contents of all addresses after completing the erase command. Input command code $A0_{16}$ in the first transfer. Proceed and input the low-order 8 bits and the high-order 8 bits of the address and pull the $\overline{OE}$ pin low. When this is done, the M38B79FF reads out the contents of the specified address, and then latchs it into the internal data latch. When the $\overline{OE}$ pin is released back high and serial clock is input to the SCLK pin, the verify data that has been latched into the data latch is serially output from the SDA pin.

Note: If any memory location where the contents have not been erased is found in the erase verify operation, execute the operation of "erase → erase verify" over again. In this case, however, the user does not need to write data 00_{16} to memory locations before erasing.



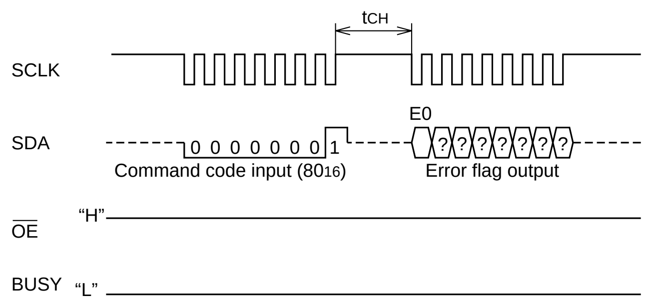
Note : When outputting the verify data, the SDA pin is switched for output at the first falling edge of SCLK. The SDA pin is placed in the floating state during the period of th(C-E) after the last rising edge of SCLK (at the 8th bit).

Fig. 96 Timings during erase verify

● **Error check command**

Input command code 80_{16} in the first transfer, and the M38B79FF outputs error information from the SDA pin, beginning at the next falling edge of the serial clock. If the LSB bit of the 8-bit error information is 1, it indicates that a command error has occurred. A command error means that some invalid commands other than commands shown in Table 20 has been input.

When a command error occurs, the serial communication circuit sets the corresponding flag and stops functioning to avoid an erroneous programming or erase. When being placed in this state, the serial communication circuit does not accept the subsequent serial clock and data (even including an error check command). Therefore, if the user wants to execute an error check command, temporarily drop the V_{PP} pin input to the V_{PPL} level to terminate the serial input/output mode. Then, place the M38B79FF into the serial I/O mode back again. The serial communication circuit is reset by this operation and is ready to accept commands. The error flag alone is not cleared by this operation, so the user can examine the serial communication circuit's error conditions before reset. This examination is done by the first execution of an error check command after the reset. The error flag is cleared when the user has executed the error check command. Because the error flag is undefined immediately after power-on, always be sure to execute the error check command.



Note: When outputting the error flag, the SDA pin is switched for output at the first falling edge of the serial clock. The SDA pin is placed in the floating state during the period of $t_{h(C-E)}$ after the last rising edge of the serial clock (at the 8th bit).

Fig. 97 Timings at error checking

DC ELECTRICAL CHARACTERISTICS (Ta = 25 °C, VCC = 5 V ± 10 %, VPP = 11.7 to 12.6 V, unless otherwise noted)

I_{CC}, I_{PP}-relevant standards during read, program, and erase are the same as in the parallel input/output mode. V_{IH}, V_{IL}, V_{OH}, V_{OL}, I_{IH}, and I_{IL} for the SCLK, SDA, BUSY, $\overline{OE}$ pins conform to the microcomputer modes.

Table 21 AC Electrical characteristics
(Ta = 25 °C, VCC = 5 V ± 10 %, VPP = 11.7 to 12.6 V, f(XIN) = 4 MHz, unless otherwise noted)

Symbol	Parameter	Limits		Unit
		Min.	Max.	
t_{CH}	Serial transmission interval	625 (Note 1)		ns
t_{CR}	Read waiting time after transmission	625 (Note 1)		ns
t_{WR}	Read pulse width	500 (Note 2)		ns
t_{RC}	Transfer waiting time after read	625 (Note 1)		ns
t_{CRPV}	Waiting time before program verify	6		µs
t_{WP}	Programming time		10	µs
t_{PC}	Transfer waiting time after programming	625 (Note 1)		ns
t_{CREV}	Waiting time before erase verify	6		µs
t_{WE}	Erase time		9.5	ms
t_{EC}	Transfer waiting time after erase	625 (Note 1)		ns
$t_{c(CK)}$	SCLK input cycle time	250		ns
$t_{w(CKH)}$	SCLK high-level pulse width	100		ns
$t_{w(CKL)}$	SCLK low-level pulse width	100		ns
$t_{r(CK)}$	SCLK rise time	20		ns
$t_{f(CK)}$	SCLK fall time	20		ns
$t_{d(C-Q)}$	SDA output delay time	0	90	ns
$t_{h(C-Q)}$	SDA output hold time	0		ns
$t_{h(C-E)}$	SDA output hold time (only the 8th bit)	187.5 (Note 3)	312.5 (Note 4)	ns
$t_{su(D-C)}$	SDA input set up time	30		ns
$t_{h(C-D)}$	SDA input hold time	90		ns

Notes 1: When f(XIN) = 4 MHz or less, calculate the minimum value according to formula 1.

Formula 1 : $\frac{2500}{f(X_{IN})} \times 10^6$

2: When f(XIN) = 4 MHz or less, calculate the minimum value according to formula 2.

Formula 2 : $\frac{2000}{f(X_{IN})} \times 10^6$

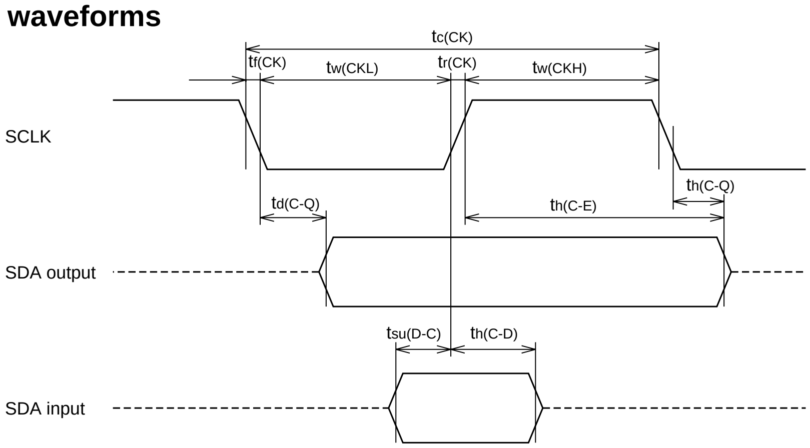
3: When f(XIN) = 4 MHz or less, calculate the minimum value according to formula 3.

Formula 3 : $\frac{750}{f(X_{IN})} \times 10^6$

4: When f(XIN) = 4 MHz or less, calculate the minimum value according to formula 4

Formula 4 : $\frac{1250}{f(X_{IN})} \times 10^6$

AC waveforms



Test conditions for AC characteristics

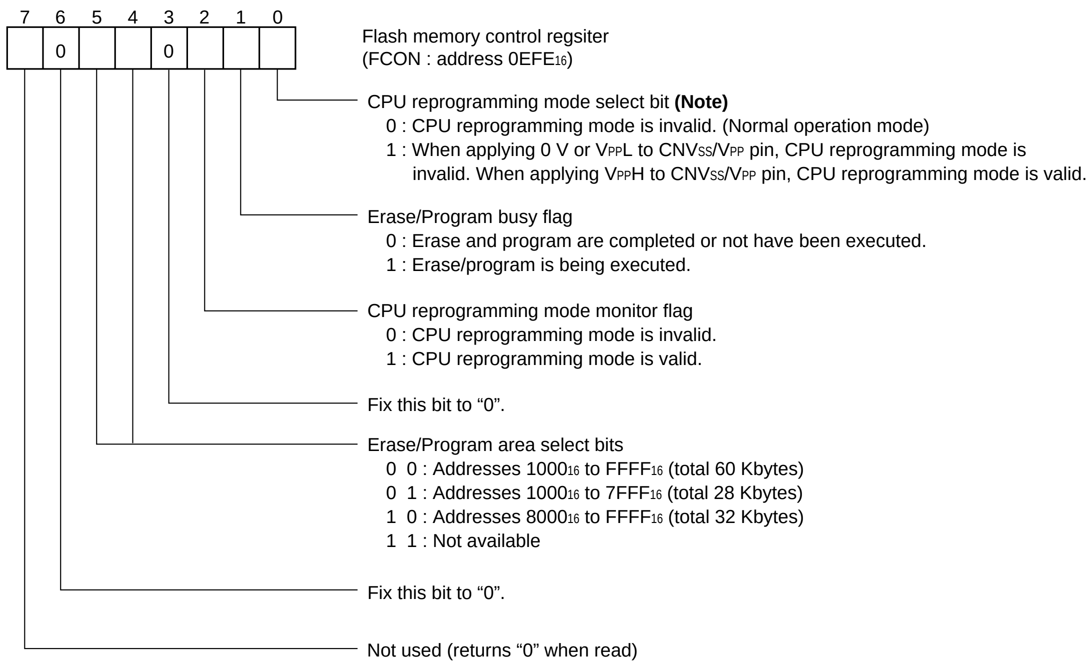
- Output timing voltage : V_{OL} = 0.8 V, V_{OH} = 2.0 V
- Input timing voltage : V_{IL} = 0.2 V_{CC}, V_{IH} = 0.8 V_{CC}

(3) Flash memory mode 3 (CPU reprogramming mode)

The M38B79FF has the CPU reprogramming mode where a built-in flash memory is handled by the central processing unit (CPU).
In CPU reprogramming mode, the flash memory is handled by writing and reading to/from the flash memory control register (see Figure 98) and the flash command register (see Figure 99).
The CNVSS pin is used as the VPP power supply pin in CPU reprogramming mode. It is necessary to apply the power-supply voltage of VPPH from the external to this pin.

Functional Outline (CPU reprogramming mode)

Figure 98 shows the flash memory control register bit configuration. Figure 99 shows the flash command register bit configuration.
Bit 0 of the flash memory control register is the CPU reprogramming mode select bit. When this bit is set to “1” and VPPH is applied to the CNVss/VPP pin, the CPU reprogramming mode is selected. Whether the CPU reprogramming mode is realized or not is judged by reading the CPU reprogramming mode monitor flag (bit 2 of the flash memory control register).
Bit 1 is a busy flag which becomes “1” during erase and program execution.
Whether these operations have been completed or not is judged by checking this flag after each command of erase and the program is executed.
Bits 4, 5 of the flash memory control register are the erase/program area select bits. These bits specify an area where erase and program is operated. When the erase command is executed after an area is specified by these bits, only the specified area is erased. Only for the specified area, programming is enabled; for the other areas, programming is disabled.
Figure 100 shows the CPU mode register bit configuration in the CPU reprogramming mode.



Note: Bit 0 can be reprogrammed only when 0 V is applied to the CNVSS/VPP pin.

Fig. 98 Flash memory control register bit configuration

● **CPU reprogramming mode operation procedure**

The operation procedure in CPU reprogramming mode is described below.

< Beginning procedure >

① Apply 0 V to the CNVss/VPP pin for reset release.
② Set the CPU mode register. (see Figure 100)
③ After CPU reprogramming mode control program is transferred to internal RAM, jump to this control program on RAM. (The following operations are controlled by this control program).
④ Set "1" to the CPU reprogramming mode select bit.
⑤ Apply VPPH to the CNVSS/VPP pin.
⑥ Wait till CNVSS/VPP pin becomes 12 V.
⑦ Read the CPU reprogramming mode monitor flag to confirm whether the CPU reprogramming mode is valid.
⑧ The operation of the flash memory is executed by software-command-writing to the flash command register .

Note: The following are necessary other than this:
- •Control for data which is input from the external (serial I/O etc.) and to be programmed to the flash memory
- •Initial setting for ports etc.
- •Writing to the watchdog timer

< Release procedure >

① Apply 0V to the CNVSS/VPP pin.
② Wait till CNVSS/VPP pin becomes 0V.
③ Set the CPU reprogramming mode select bit to "0".

Each software command is explained as follows.

● **Read command**

When "00_{16}" is written to the flash command register, the M38B79FF enters the read mode. The contents of the corresponding address can be read by reading the flash memory (For instance, with the LDA instruction etc.) under this condition.
The read mode is maintained until another command code is written to the flash command register. Accordingly, after setting the read mode once, the contents of the flash memory can continuously be read.
After reset and after the reset command is executed, the read mode is set.

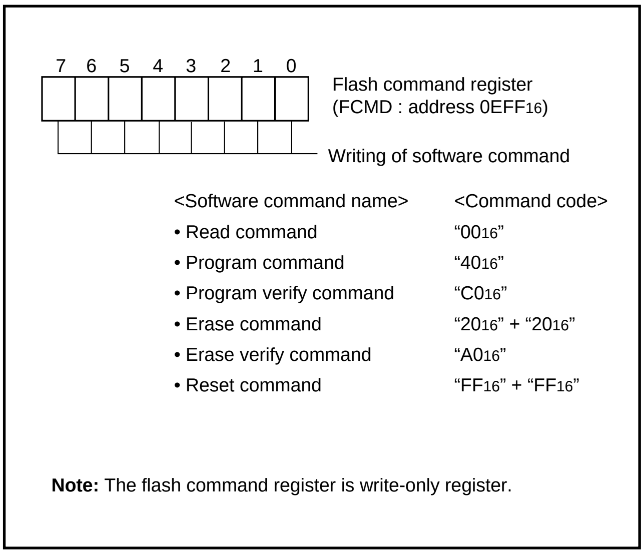


Fig. 99 Flash command register bit configuration

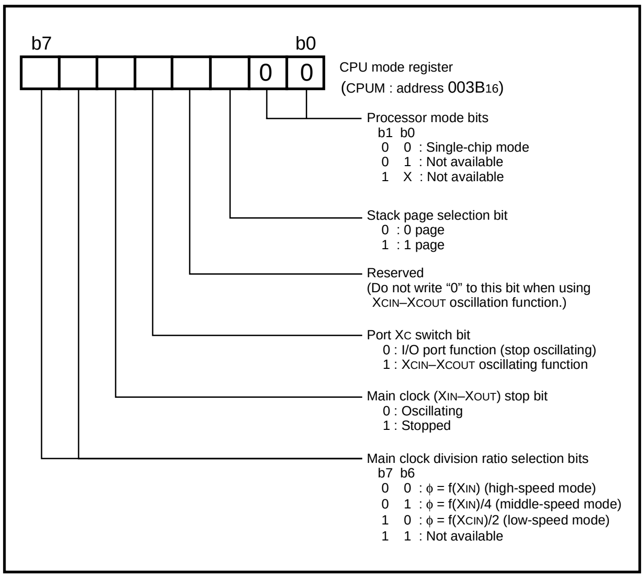


Fig. 100 CPU mode register bit configuration in CPU rewriting mode

● **Program command**

When "40$_{16}$" is written to the flash command register, the M38B79FF enters the program mode.

Subsequently to this, if the instruction (for instance, STA instruction) for writing byte data in the address to be programmed is executed, the control circuit of the flash memory executes the program. The erase/program busy flag of the flash memory control register is set to "1" when the program starts, and becomes "0" when the program is completed. Accordingly, after the write instruction is executed, CPU can recognize the completion of the program by polling this bit.

The programmed area must be specified beforehand by the erase/program area select bits.

During programming, watchdog timer stops with "FFFF$_{16}$" set.

Note: A programming operation is not completed by executing the program command once. Always be sure to execute a program verify command after executing the program command. When the failure is found in this verification, the user must repeatedly execute the program command until the pass. Refer to Figure 101 for the flow chart of the programming.

● **Program verify command**

When "C0$_{16}$" is written to the flash command register, the M38B79FF enters the program verify mode. Subsequently to this, if the instruction (for instance, LDA instruction) for reading byte data from the address to be verified (i.e., previously programmed address), the contents which has been written to the address actually is read.

CPU compares this read data with data which has been written by the previous program command. In consequence of the comparison, if not agreeing, the operation of "program → program verify" must be executed again.

● **Erase command**

When writing "20$_{16}$" twice continuously to the flash command register, the flash memory control circuit performs erase to the area specified beforehand by the erase/program area select bits.

Erase/program busy flag of the flash memory control register becomes "1" when erase begins, and it becomes "0" when erase completes. Accordingly, CPU can recognize the completion of erase by polling this bit.

Data "00$_{16}$" must be written to all areas to be erased by the program and the program verify commands before the erase command is executed.

During erasing, watchdog timer stops with "FFFF$_{16}$" set.

Note: The erasing operation is not completed by executing the erase command once. Always be sure to execute an erase verify command after executing the erase command. When the failure is found in this verification, the user must repeatedly execute the erase command until the pass. Refer to Figure 101 for the erasing flowchart.

● **Erase verify command**

When "A0$_{16}$" is written to the flash command register, the M38B79FF enters the erase verify mode. Subsequently to this, if the instruction (for instance, LDA instruction) for reading byte data from the address to be verified, the contents of the address is read.

CPU must erase and verify to all erased areas in a unit of address.

If the address of which data is not "FF$_{16}$" (i.e., data is not erased) is found, it is necessary to discontinue erasure verification there, and execute the operation of "erase → erase verify" again.

Note: By executing the operation of "erase → erase verify" again when the memory not erased is found. It is unnecessary to write data "00$_{16}$" before erasing in this case.

● **Reset command**

The reset command is a command to discontinue the program or erase command on the way. When "FF$_{16}$" is written to the command register two times continuously after "40$_{16}$" or "20$_{16}$" is written to the flash command register, the program, or erase command becomes invalid (reset), and the M38B79FF enters the reset mode.

The contents of the memory does not change even if the reset command is executed.

DC Electric Characteristics

Note: The characteristic concerning the flash memory part are the same as the characteristic of the parallel I/O mode.

AC Electric Characteristics

Note: The characteristics are the same as the characteristic of the microcomputer mode.

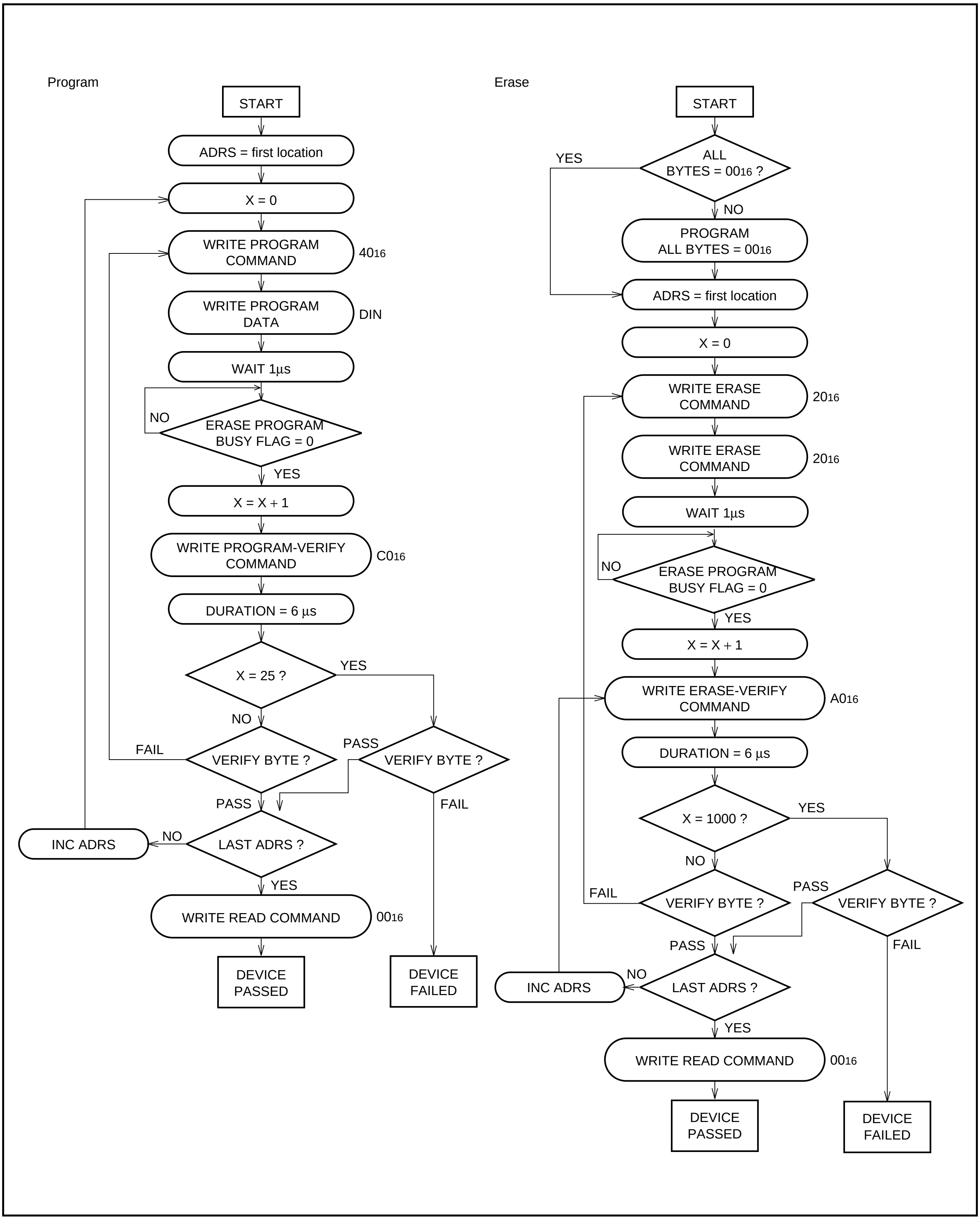


Fig. 101 Flowchart of program/erase operation at CPU reprogramming mode

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. 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 instruction with 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 serial I/O transfer and serial I/O automatic transfer.

Automatic Transfer Serial I/O

When using the automatic transfer serial I/O mode of the serial I/O1, set an automatic transfer interval as the following.
Otherwise the serial data might be incorrectly transmitted/received.

•Set an automatic transfer interval for each 1-byte data transfer as the following:

(1) Not using FLD controller
Keep the interval for **5 cycles or more of internal system clock** from clock rising of the last bit of 1-byte data.

(2) Using FLD controller
(a) Not using gradation display
Keep the interval for **17 cycles or more of internal system clock** from clock rising of the last bit of 1-byte data.

(b) Using gradation display
Keep the interval for **27 cycles or more of internal system clock** from clock rising of the last bit of 1-byte data.

A-D Converter

The comparator uses capacitive coupling amplifier 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.

D-A Converter

The accuracy of the D-A converter becomes rapidly poor under the V_{CC} = 4.0 V or less condition; a supply voltage of $V_{CC} \geq 4.0$ V is recommended. When a D-A converter is not used, set the value of D-A conversion register to "00_{16}".

Instruction Execution Time

The instruction execution time is obtained by multiplying the period of the internal clock ϕ 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 period of the internal clock ϕ is half of the X_{IN} period in high-speed mode.

NOTES ON USAGE

Handling of Power Source Pins

In order to avoid a latch-up occurrence, connect a capacitor suitable for high frequencies as bypass capacitor between power source pin (V_{CC} pin) and GND pin (V_{SS} pin), between power source pin (V_{CC} pin) and analog power source input pin (AV_{SS} pin), and between program power source pin (CNV_{SS}/V_{PP}) and GND pin for flash memory version when on-board reprogramming is executed. Besides, connect the capacitor to as close as possible. For bypass capacitor which should not be located too far from the pins to be connected, a ceramic capacitor of 0.01 µF–0.1 µF is recommended.

Flash Memory Version

The CNV_{SS} pin is connected to the internal memory circuit block by a low-ohmic resistance, since it has the multiplexed function to be a programmable power source pin (V_{PP} pin) as well.
To improve the noise reduction, connect a track between CNV_{SS} pin and V_{SS} pin or V_{CC} pin with 1 to 10 kΩ resistance.
The mask ROM version track of CNV_{SS} pin has no operational interference even if it is connected to Vss pin or Vcc pin via a resistor.

Electric Characteristic Differences Between Mask ROM and Flash Memory Version MCUs

There are differences in electric characteristics, operation margin, noise immunity, and noise radiation between Mask ROM and Flash Memory version MCUs due to the difference in the manufacturing processes.
When manufacturing an application system with Flash Memory version and then switching to use of Mask ROM version, please perform sufficient evaluations for the commercial samples of Mask ROM version.

DATA REQUIRED FOR MASK ORDERS

The following are necessary when ordering a mask ROM production:

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

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

Address	
0000_{16}	Port P0 (P0)
0001_{16}	
0002_{16}	Port P1 (P1)
0003_{16}	Port P1 direction register (P0D)
0004_{16}	Port P2 (P2)
0005_{16}	
0006_{16}	Port P3 (P3)
0007_{16}	Port P3 direction register (P2D)
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}	Port PA (PA)
0015_{16}	Port PA direction register (PAD)
0016_{16}	Port PB (PB)
0017_{16}	Port PB direction register (PBD)
≈	≈
$0EEF_{16}$	Pull-up control register 3 (PULL3)
$0EF0_{16}$	Pull-up control register 1 (PULL1)
$0EF1_{16}$	Pull-up control register 2 (PULL2)
≈	≈

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 to 7, 9, A)
(Pi: addresses 00_{16}, 02_{16}, 04_{16}, 06_{16}, 08_{16}, $0A_{16}$, $0C_{16}$, $0E_{16}$, 12_{16}, 14_{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	O	O
1	Port Pi_1		0	O	O
2	Port Pi_2		0	O	O
3	Port Pi_3		0	O	O
4	Port Pi_4		0	O	O
5	Port Pi_5		0	O	O
6	Port Pi_6		0	O	O
7	Port Pi_7		0	O	O

Fig. 2.1.2 Structure of port Pi (i = 0 to 7, 9, A)

Port P8

b7 b6 b5 b4 b3 b2 b1 b0

Port P8
(P8: address 10_{16})

b	Name	Functions	At reset	R	W
0	Port $P8_0$	●In output mode Write ········ Port latch Read ········ Port latch ●In input mode Write ········ Port latch Read ········ Value of pin	0	O	O
1	Port $P8_1$		0	O	O
2	Port $P8_2$		0	O	O
3	Port $P8_3$		0	O	O
4	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
5			0	×	×
6			0	×	×
7			0	×	×

Fig. 2.1.3 Structure of port P8

Port PB

b7 b6 b5 b4 b3 b2 b1 b0

Port PB
(PB: address 16_{16})

b	Name	Functions	At reset	R	W
0	Port PB_0	●In output mode Write ········ Port latch Read ········ Port latch ●In input mode Write ········ Port latch Read ········ Value of pin	0	○	○
1	Port PB_1		0	○	○
2	Port PB_2		0	○	○
3	Port PB_3		0	○	○
4	Port PB_4		0	○	○
5	Port PB_5		0	○	○
6	Port PB_6		0	○	○
7	Nothing is arranged for this bit. When this bit is read out, the contents are undefined.		0	×	×

Fig. 2.1.4 Structure of port PB

Port Pi direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port Pi direction register (PiD) (i = 1, 3 to 7, 9, A)
[Addresses 03_{16}, 07_{16}, 09_{16}, $0B_{16}$, $0D_{16}$, $0F_{16}$, 13_{16}, 15_{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	0	×	○

Fig. 2.1.5 Structure of port Pi (i = 1, 3 to 7, 9, A) direction register

Port P8 direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port P8 direction register
(P8D: address 11_{16})

b	Name	Functions	At reset	R	W
0	Port P8 direction register	0 : Port $P8_0$ input mode 1 : Port $P8_0$ output mode	0	×	O
1		0 : Port $P8_1$ input mode 1 : Port $P8_1$ output mode	0	×	O
2		0 : Port $P8_2$ input mode 1 : Port $P8_2$ output mode	0	×	O
3		0 : Port $P8_3$ input mode 1 : Port $P8_3$ output mode	0	×	O
4	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
5			0	×	×
6			0	×	×
7			0	×	×

Fig. 2.1.6 Structure of port P8 direction register

Port PB direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port PB direction register
(PBD: address 17_{16})

b	Name	Functions	At reset	R	W
0	Port PB direction register	0 : Port PB_0 input mode 1 : Port PB_0 output mode	0	×	O
1		0 : Port PB_1 input mode 1 : Port PB_1 output mode	0	×	O
2		0 : Port PB_2 input mode 1 : Port PB_2 output mode	0	×	O
3		0 : Port PB_3 input mode 1 : Port PB_3 output mode	0	×	O
4		0 : Port PB_4 input mode 1 : Port PB_4 output mode	0	×	O
5		0 : Port PB_5 input mode 1 : Port PB_5 output mode	0	×	O
6		0 : Port PB_6 input mode 1 : Port PB_6 output mode	0	×	O
7	Nothing is arranged for this bit. When this bit is read out, the contents are undefined.		0	×	×

Fig. 2.1.7 Structure of port PB direction register

Pull-up control register 1

b7 b6 b5 b4 b3 b2 b1 b0

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

b	Name	Functions	At reset	R	W
0	Ports $P6_4$, $P6_5$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
1	Ports $P6_6$, $P6_7$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
2	Ports $P7_0$, $P7_1$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
3	Ports $P7_2$, $P7_3$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
4	Ports $P7_4$, $P7_5$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
5	Ports $P7_6$, $P7_7$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
6	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	O	×
7			0	O	×

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

Pull-up control register 2

b7 b6 b5 b4 b3 b2 b1 b0

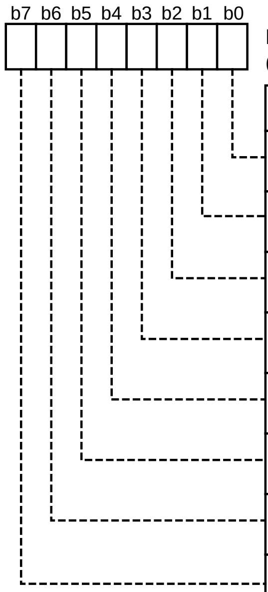
Pull-up control register 2
(PULL2: address $0EF1_{16}$)

b	Name	Functions	At reset	R	W
0	Ports $P8_0$, $P8_1$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
1	Ports $P8_2$, $P8_3$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
2	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	O	×
3	Ports $P9_0$, $P9_1$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
4	Ports $P9_2$, $P9_3$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
5	Ports $P9_4$, $P9_5$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
6	Ports $P9_6$, $P9_7$ pull-up control	0: No pull-up 1: Pull-up	0	O	O
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	O	×

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

Pull-up control register 3



Pull-up control register 3
(PULL3: address 0EEF16)

b	Name	Functions	At reset	R	W
0	Ports PA_0, PA_1 pull-up control	0: No pull-up 1: Pull-up	0	O	O
1	Ports PA_2, PA_3 pull-up control	0: No pull-up 1: Pull-up	0	O	O
2	Ports PA_4, PA_5 pull-up control	0: No pull-up 1: Pull-up	0	O	O
3	Ports PA_6, PA_7 pull-up control	0: No pull-up 1: Pull-up	0	O	O
4	Ports PB_0, PB_1 pull-up control	0: No pull-up 1: Pull-up	0	O	O
5	Ports PB_2, PB_3 pull-up control	0: No pull-up 1: Pull-up	0	O	O
6	Ports PB_4, PB_5 pull-up control	0: No pull-up 1: Pull-up	0	O	O
7	Ports PB_6 pull-up control	0: No pull-up 1: Pull-up	0	O	O

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

Fig. 2.1.10 Structure of pull-up control register 3

2.1.3 Terminate unused pins

Table 2.1.1 Termination of unused pins

Pins	Termination
P0, P2	Open at “H” output state.
P1, P3–P5, $P6_0$–$P6_3$	• 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.
$P6_4$–$P6_7$, P7, $P8_0$–$P8_3$, P9, PA, PB_0–PB_6	• 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.
V_{REF}	Open.
X_{OUT}	Open (only when using external clock).
AV_{SS}	Connect to V_{SS} (GND).
V_{EE}	Connect to V_{SS} (GND).
CNV_{SS}	Connect to V_{SS} through a resistor of 1 kΩ to 10 kΩ.

2.1.4 Notes on I/O port

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

● **Reason**

The 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) 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

(3) Pull-up/Pull-down control

When each port which has built-in pull-up/pull-down resistor is set to output port, pull-up/pull-down control of corresponding port becomes invalid. (Pull-up/pull-down cannot be set.)

● **Reason**

Pull-up/pull-down 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 Vcc or Vss 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 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 Vcc or 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.

2.2 Timer

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

2.2.1 Memory map

Address	Register
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)
$002B_{16}$	(D-A conversion register (DA))
$002C_{16}$	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)
$003C_{16}$	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)

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 to 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	O	O
1		1	O	O
2		1	O	O
3		1	O	O
4		1	O	O
5		1	O	O
6		1	O	O
7		1	O	O

Fig. 2.2.2 Structure of Timer i (i=1, 3 to 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	O	O
1		0	O	O
2		0	O	O
3		0	O	O
4		0	O	O
5		0	O	O
6		0	O	O
7		0	O	O

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: $(n + m) \times ts$ 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	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. 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	O	O
1	Timer 2 count stop bit	0: Count operation 1: Count stop	0	O	O
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	O	O
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	O	O
5			0	O	O
6	Timer 1 output selection bit ($P7_5$)	0: I/O port 1: Timer 1 output	0	O	O
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	O	×

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	O	O
1	Timer 4 count stop bit	0: Count operation 1: Count stop	0	O	O
2	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	O	O
3					
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 $CNTR_1$ 1 1: Not available	0	O	O
5			0	O	O
6	Timer 3 output selection bit ($P7_6$)	0: I/O port 1: Timer 3 output	0	O	O
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	O	×

Fig. 2.2.6 Structure of Timer 34 mode register

Timer 56 mode register

b7 b6 b5 b4 b3 b2 b1 b0

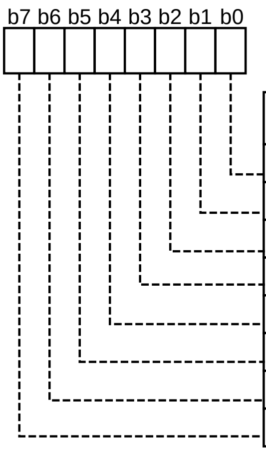
Timer 56 mode register
(T56M: address $2A_{16}$)

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: $f(X_{IN})/8$ or $f(X_{CIN})/16$ 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: $f(X_{IN})/8$ or $f(X_{CIN})/16$ 0 1: Timer 5 underflow 1 0: Timer 4 underflow 1 1: Not available	0	○	○
5			0	○	○
6	Timer 6 (PWM) output selection bit ($P7_4$)	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)



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	O	O
1		1	O	O
2		1	O	O
3		1	O	O
4		1	O	O
5		1	O	O
6		1	O	O
7		1	O	O

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

b7 b6 b5 b4 b3 b2 b1 b0

Timer X mode register 1
(TXM1: address $2E_{16}$)

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: $f(X_{IN})/2$ or $f(X_{CIN})/4$ 0 1: $f(X_{IN})/8$ or $f(X_{CIN})/16$ 1 0: $f(X_{IN})/64$ or $f(X_{CIN})/128$ 1 1: Not available	0	○	○
2			0	○	○
3	Nothing is arranged for this bit. This is a 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	$CNTR_2$ active edge switch bit	0 : •Start from "H" output in pulse output mode •Count at rising edge in event counter mode •Measure "H" pulse width in pulse width measurement mode 1 : •Start from "L" output in pulse output mode •Count at falling edge in event counter 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

b7 b6 b5 b4 b3 b2 b1 b0

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

b	Name	Functions	At reset	R	W
0	Real time port control bit ($P9_4$)	0: Real time port function is invalid 1: Real time port function is valid	0	O	O
1	Real time port control bit ($P9_5$)	0: Real time port function is invalid 1: Real time port function is valid	0	O	O
2	$P9_4$ data for real time port	0: "L" output 1: "H" output	0	O	O
3	$P9_5$ data for real time port	0: "L" output 1: "H" output	0	O	O
4	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are "0".		0	O	X
5			0	O	X
6			0	O	X
7			0	O	X

Fig. 2.2.10 Structure of Timer X mode register 2

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

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	O	✻
1	INT_1/serial I/O3 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻
2	INT_2 interrupt request bit Remote controller /counter overflow interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻
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	O	✻
4	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻
6	Timer 2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻
7	Timer 3 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻

✻: “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

b7 b6 b5 b4 b3 b2 b1 b0

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

b	Name	Functions	At reset	R	W
0	Timer 4 interrupt request bit	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	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.

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	O	O
1	INT_1/serial I/O3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
2	INT_2 interrupt enable bit Remote controller /counter overflow interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
3	Serial I/O1 interrupt enable bit Serial I/O automatic transfer interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
4	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
6	Timer 2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
7	Timer 3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O

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	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
1	Timer 5 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
2	Timer 6 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
3	Serial I/O2 receive interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
4	INT_3/Serial I/O2 transmit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
5	INT_4 interrupt enable bit A-D converter interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
6	FLD blanking interrupt enable bit FLD digit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
7	Fix “0” to this bit.		0	O	O

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 $P7_4$/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 $P9_4$ pin or $P9_5$ 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.

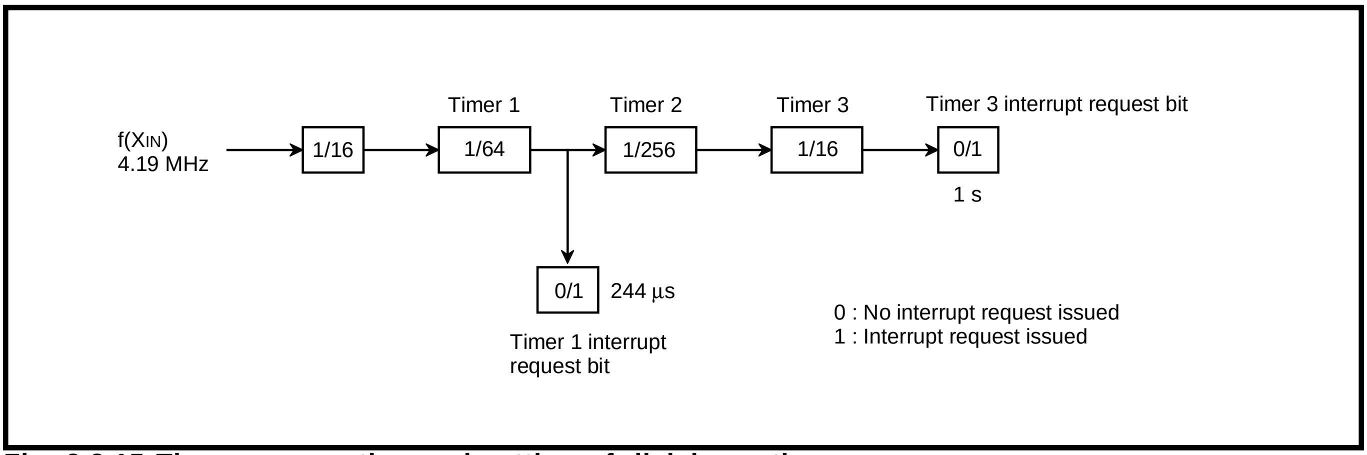


Fig. 2.2.15 Timers connection and setting of division ratios

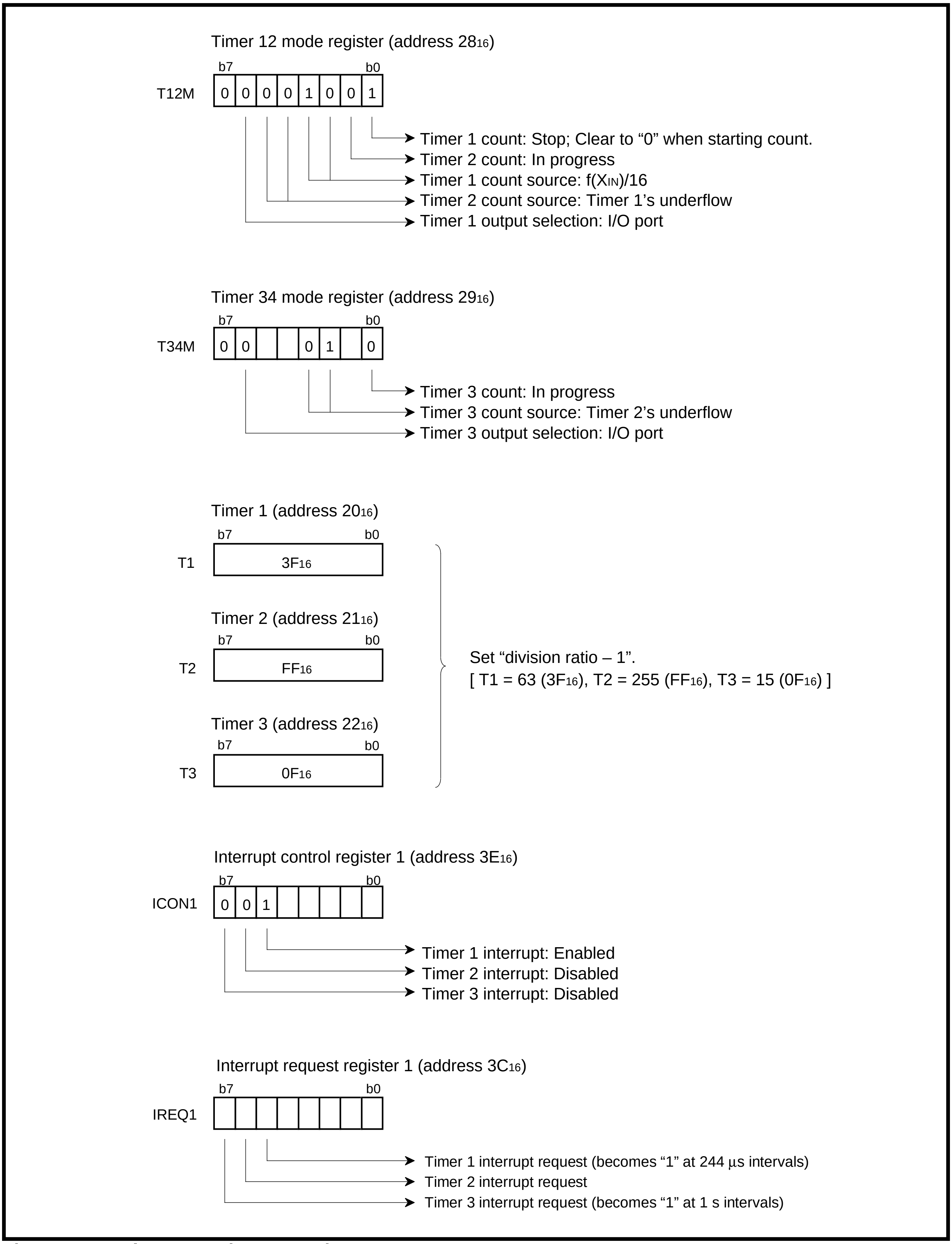
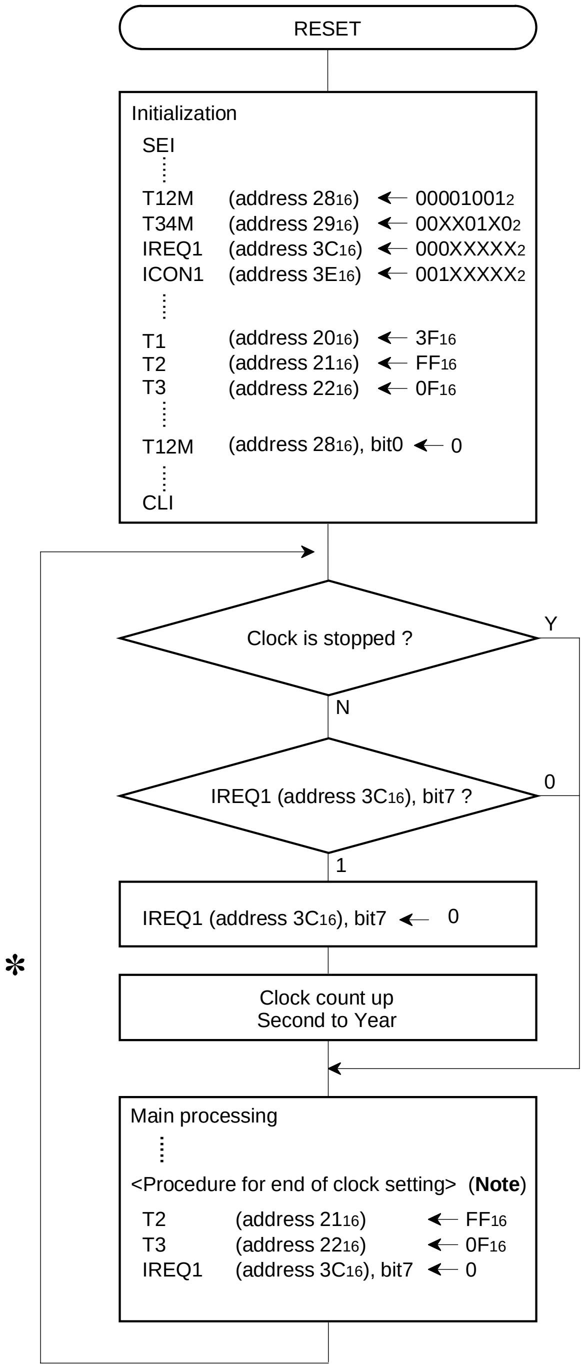


Fig. 2.2.16 Relevant registers setting

● 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 s 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 s interval.

•Set Timers again when starting clock from 0 s 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 $P7_6/T_{3OUT}$ pin.
•The level of the $P7_6/T_{3OUT}$ 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. Figure 2.2.20 shows the relevant registers setting, and Figure 2.2.21 shows the control procedure.

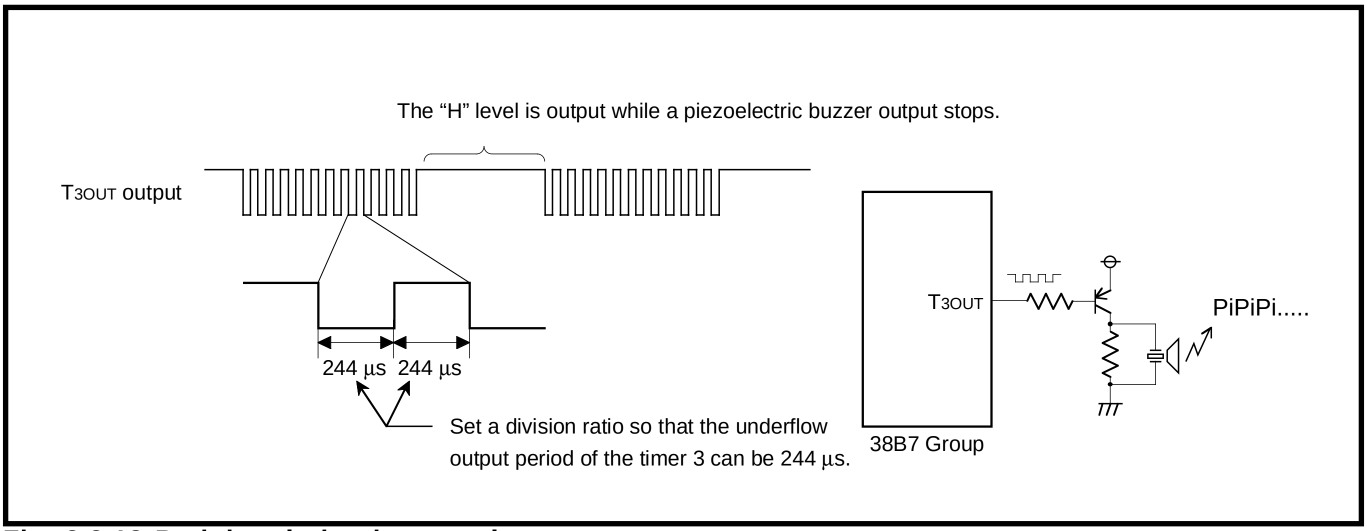


Fig. 2.2.18 Peripheral circuit example

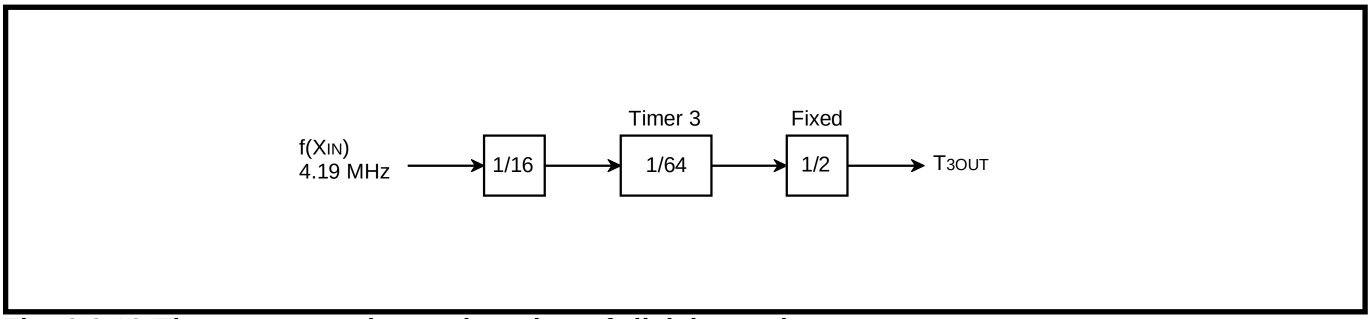


Fig. 2.2.19 Timers connection and setting of division ratios

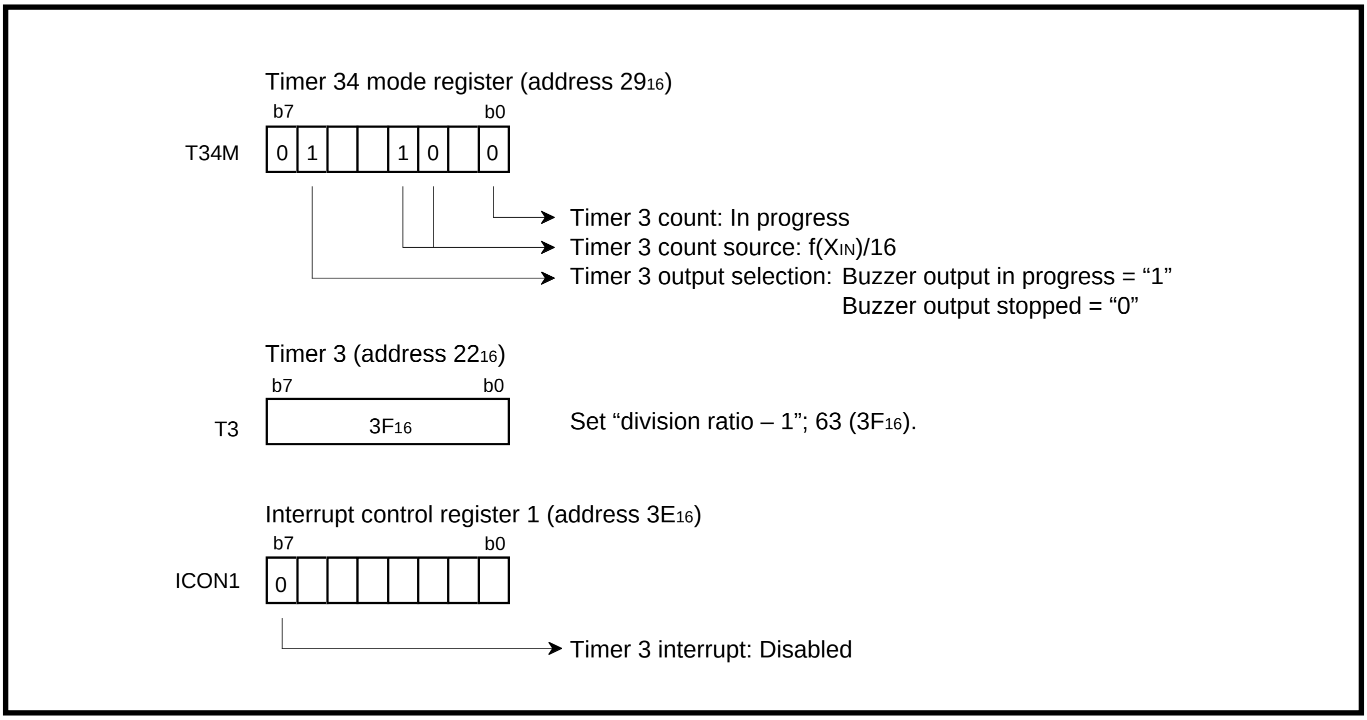


Fig. 2.2.20 Relevant registers setting

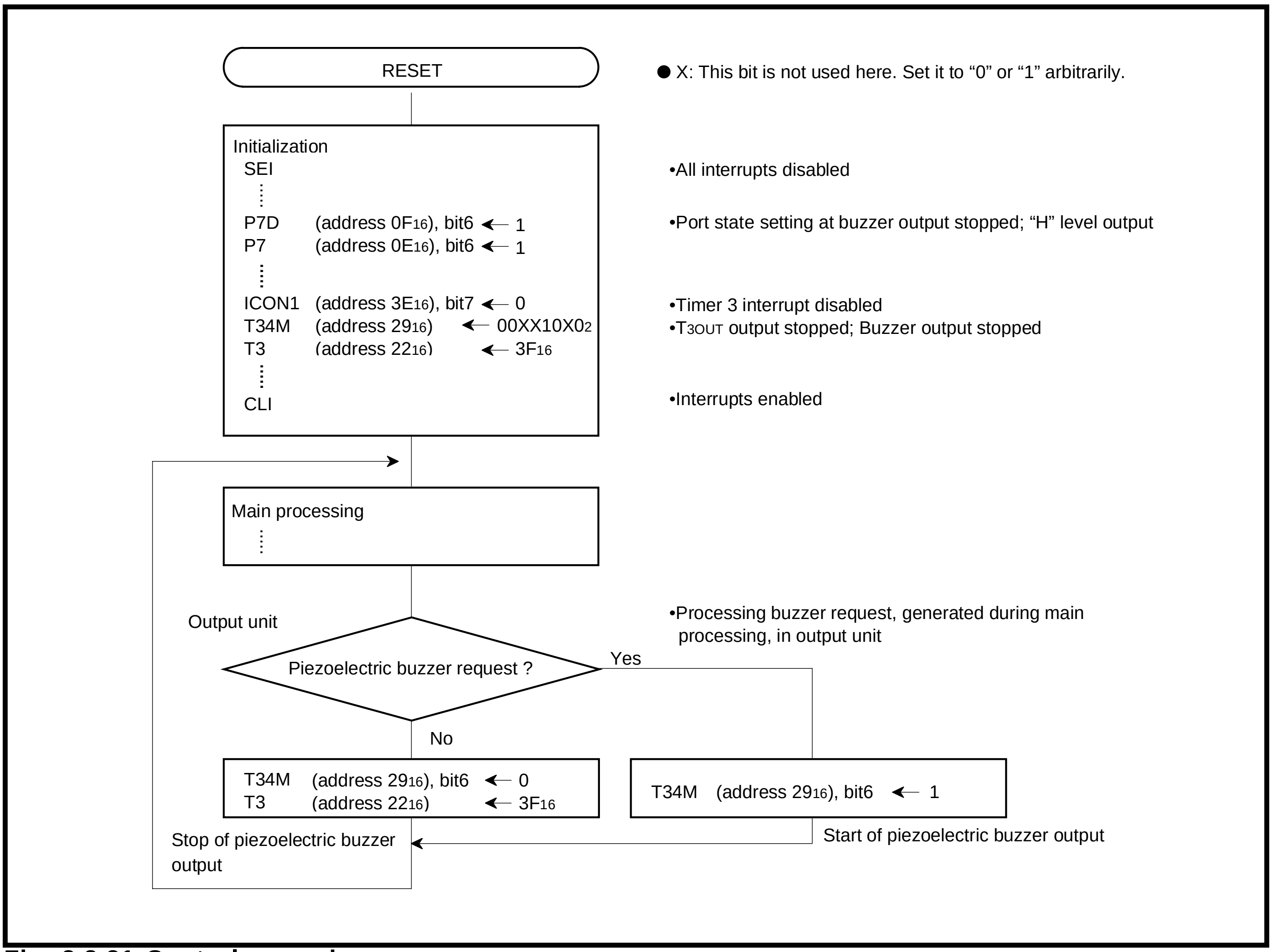


Fig. 2.2.21 Control procedure

(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 $P8_2/CNTR_1$ pin with the timer.
•A reference value

Specifications: •The pulse is input to the $P8_2/CNTR_1$ 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.

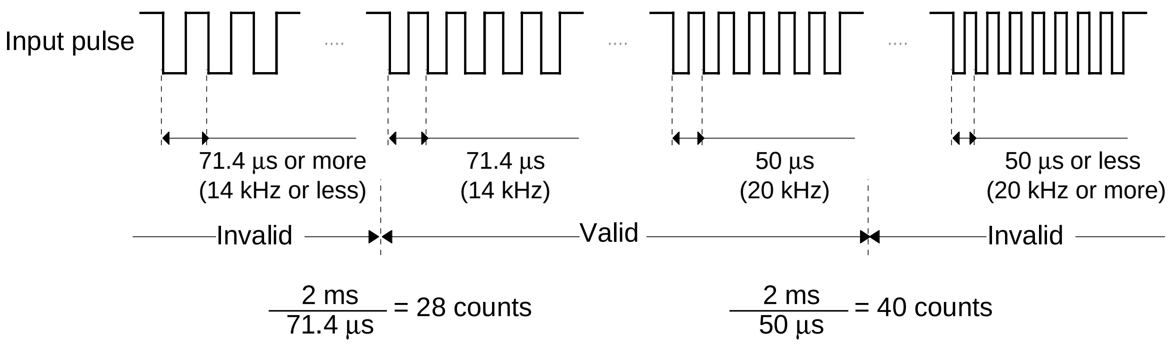


Fig. 2.2.22 Judgment method of valid/invalid of input pulses

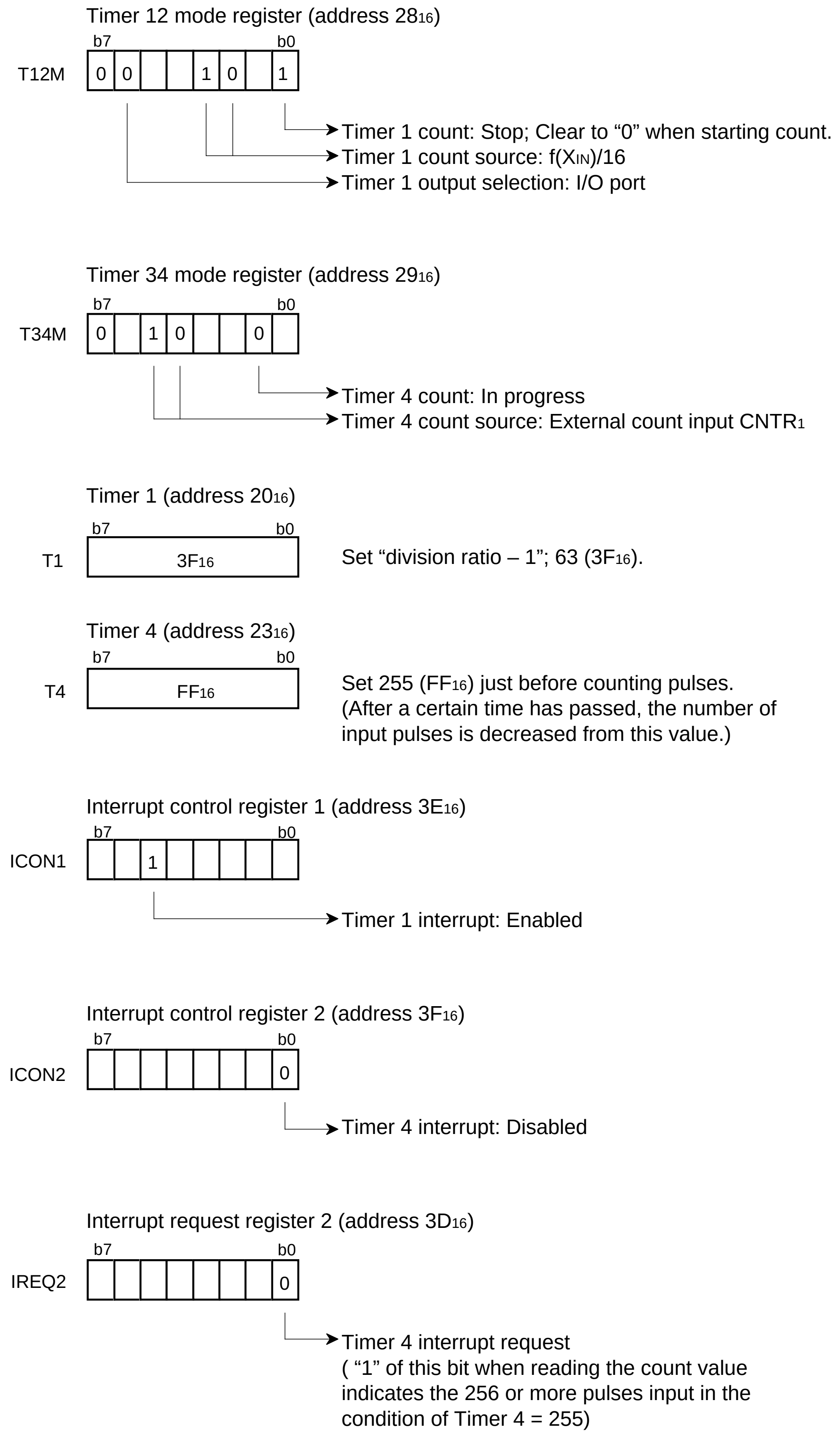


Fig. 2.2.23 Relevant registers setting

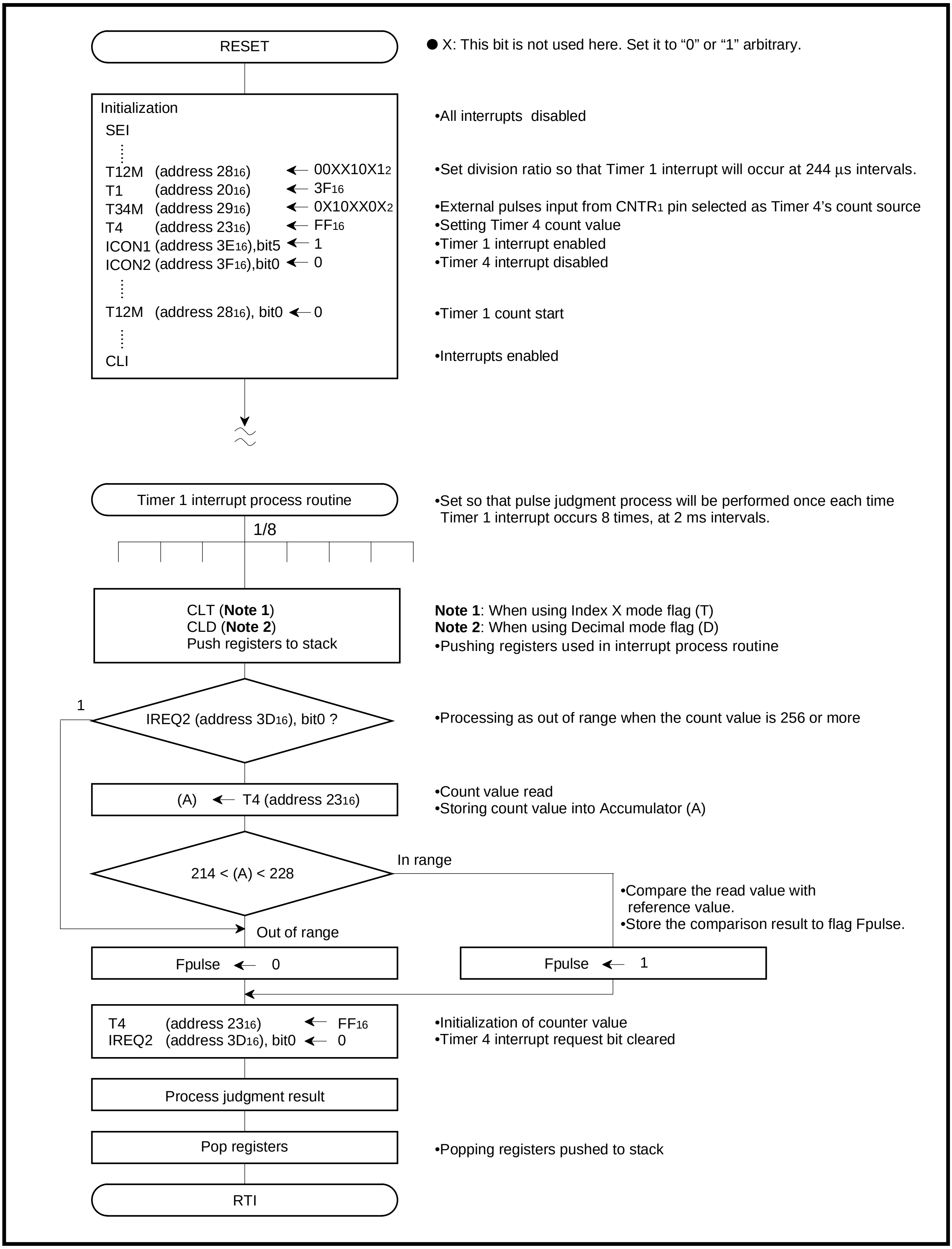


Fig. 2.2.24 Control procedure

(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 $P8_3/CNTR_0/CNTR_2$ 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 $P8_3/CNTR_0/CNTR_2$ pin.

Specifications: •The timer X counts the “H” level width of the FG pulse input to the $P8_3/CNTR_0/CNTR_2$ 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.

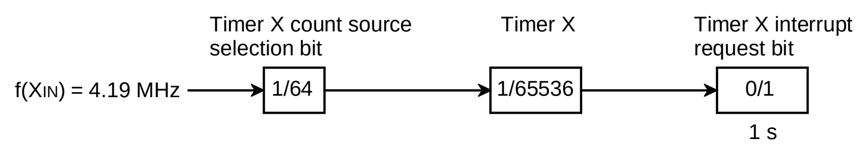


Fig. 2.2.25 Timers connection and setting of division ratios

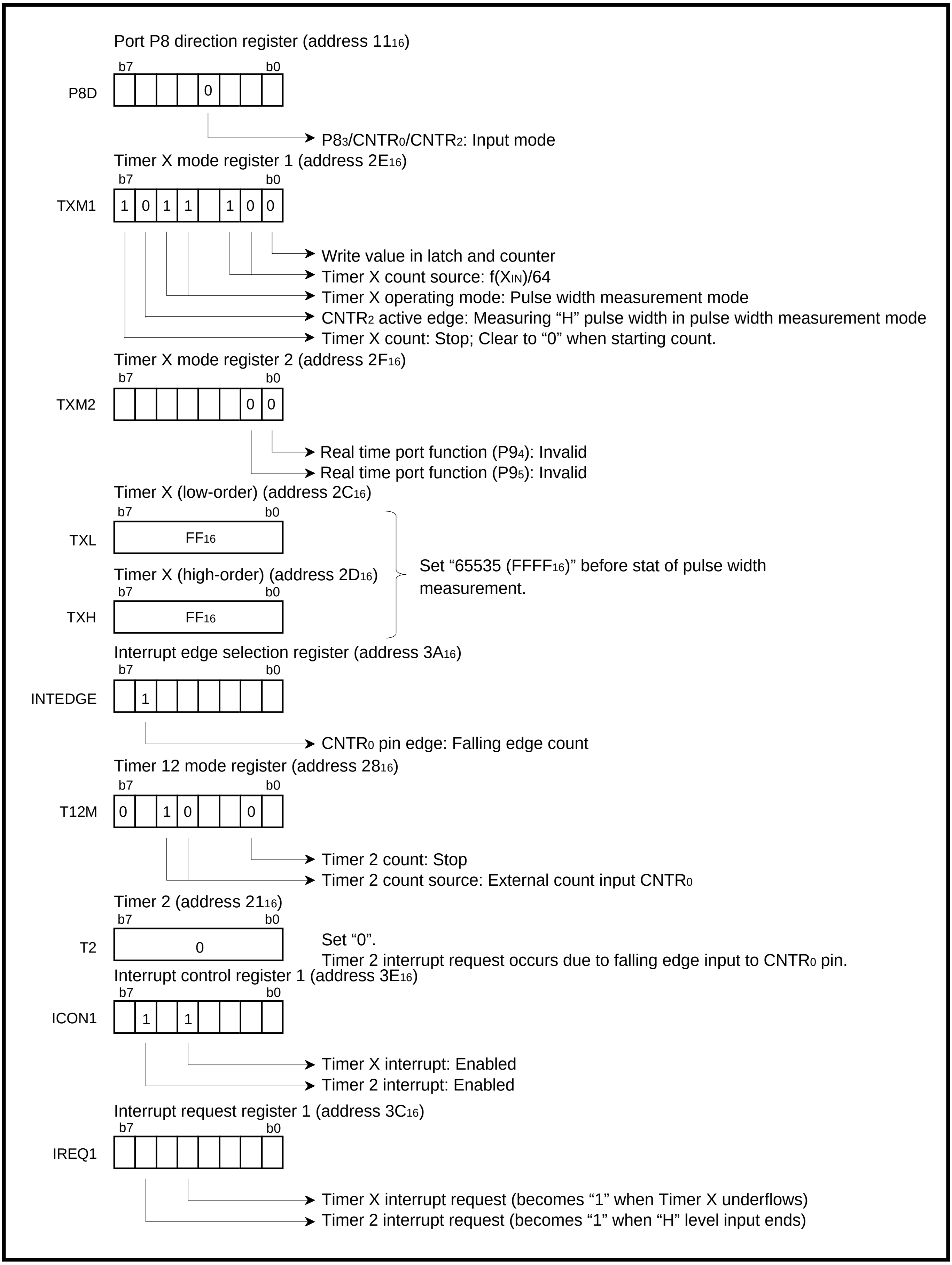


Fig. 2.2.26 Relevant registers setting

RESET

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

Initialization

Instruction	Setting	Comment
SEI		•All interrupts disabled
⋮		
P8D	(address 11_{16}),bit3 ← 0	•Setting $P8_3$/$CNTR_0$/$CNTR_2$ pin to input mode
TXM1	(address $2E_{16}$) ← $1011X100_2$	•Timer X: Pulse width measurement mode
TXM2	(address $2F_{16}$) ← $XXXXXX00_2$	(Measuring "H" pulse width of input pulses from $CNTR_2$ pin)
TXL	(address $2C_{16}$) ← FF_{16}	•Setting Timer X count value
TXH	(address $2D_{16}$) ← FF_{16}	
INTEDGE	(address $3A_{16}$),bit6 ← 1	•$CNTR_0$ pin edge: Falling edge count
T12M	(address 28_{16}) ← $0X10XX1X_2$	•External pulses input from $CNTR_0$ pin selected as Timer 2's count source
T2	(address 21_{16}) ← 0	•Setting "0" to Timer 2
ICON1	(address $3E_{16}$) ← $XXXX1X1X_2$	•Timers X and 2 interrupts: Enabled
⋮		
TXM1	(address $2E_{16}$),bit7 ← 0	•Timers X and 2 count start
T12M	(address 28_{16}),bit1 ← 0	
⋮		
CLI		•Interrupts enabled

Timer X interrupt process routine (**Note** 1)

CLT (**Note 2**)
CLD (**Note 3**)
Push registers to stack

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 $P8_3$ register.
2: When using Index X mode flag (T)
3: When using Decimal mode flag (D)

•Pushing registers used in interrupt process routine

Error processing

Pop registers

•Popping registers pushed to stack

RTI

Fig. 2.2.27 Control procedure

Timer 2 interrupt process routine (**Note** 1)

CLT (**Note 2**)
CLD (**Note 3**)
Push registers to stack

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

(A) ← TXH
Measurement result (high-order 8 bits) ← (A)
(A) ← TXL
Measurement result (low-order 8 bits) ← (A)
TXL (address $2C_{16}$) ← FF_{16}
TXH (address $2D_{16}$) ← FF_{16}

•Count value read and storing it to RAM

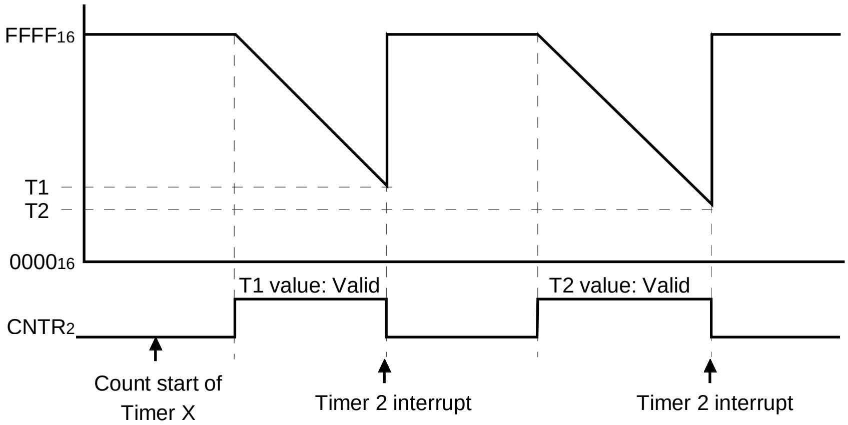
Pop registers

•Popping registers pushed to stack

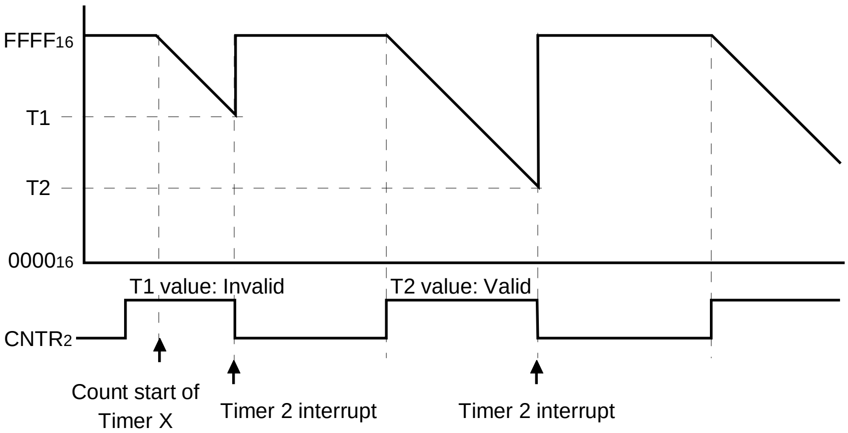
RTI

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 $P8_3$ 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.

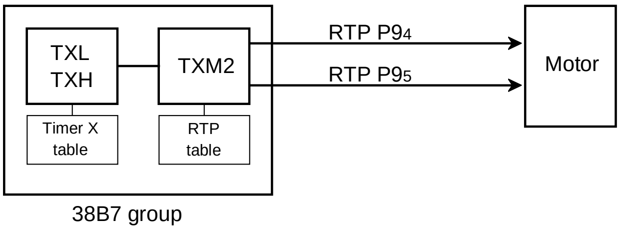


(6) Timer application example 5: Control of stepping motor

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 2.2.29 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

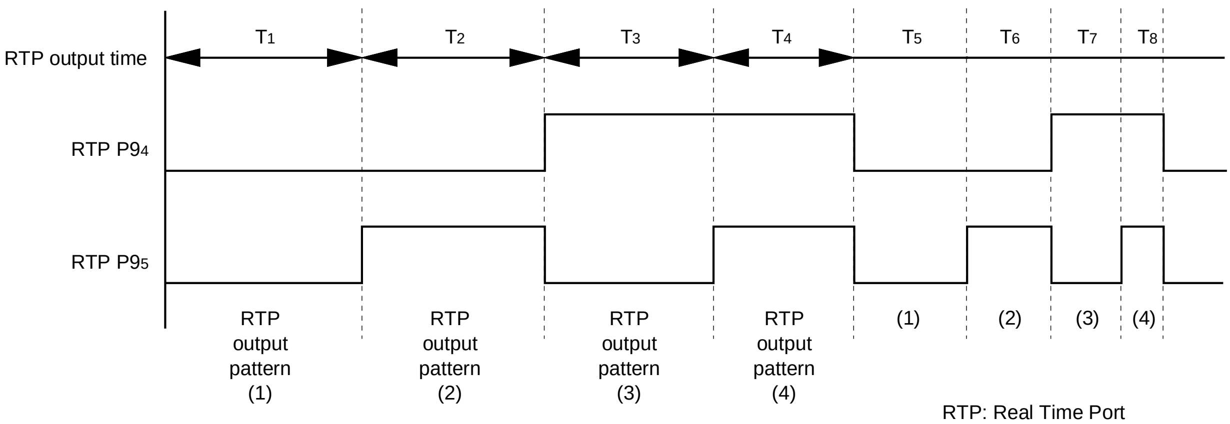


Fig. 2.2.29 RTP output example

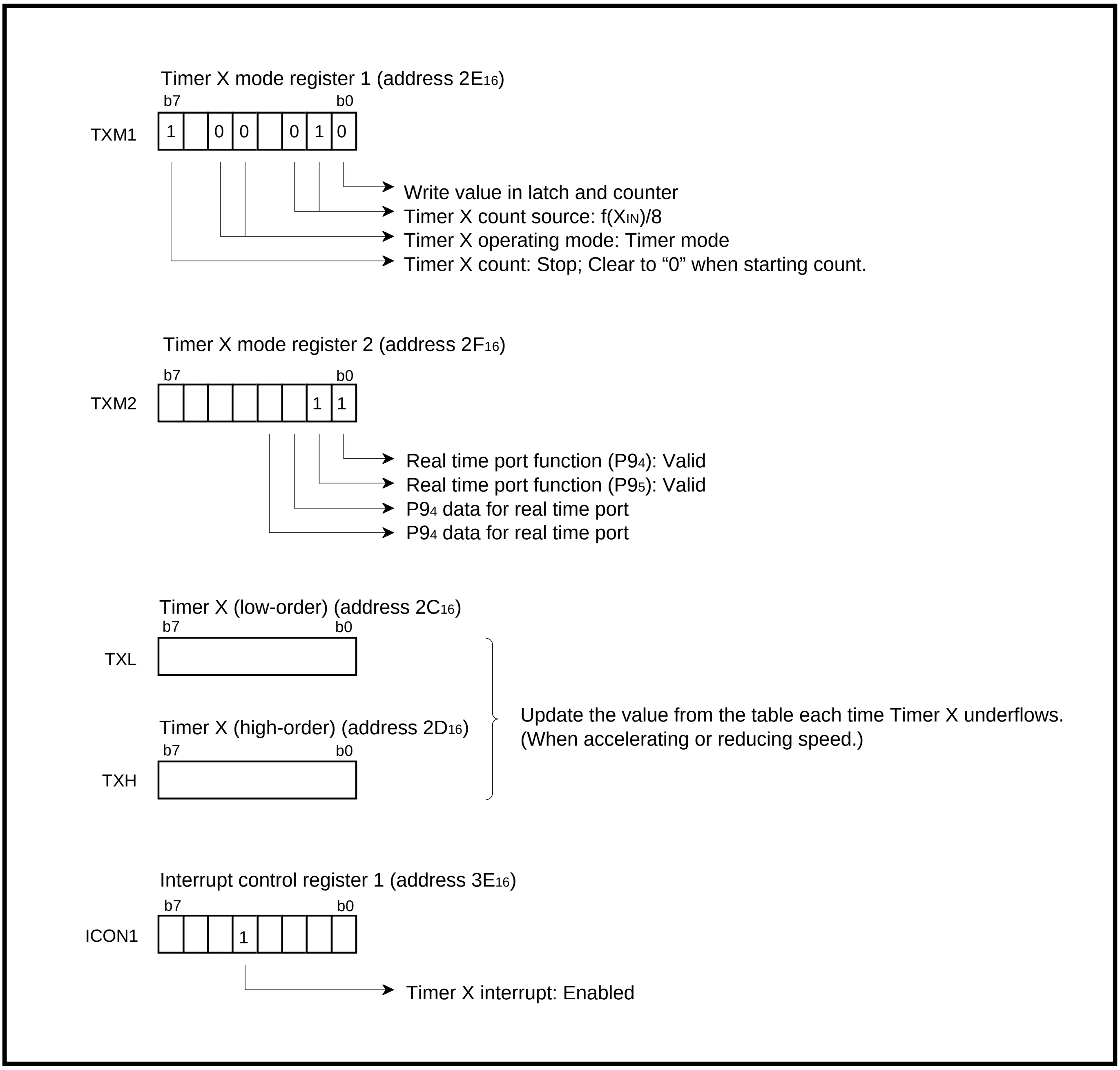


Fig. 2.2.30 Relevant registers setting

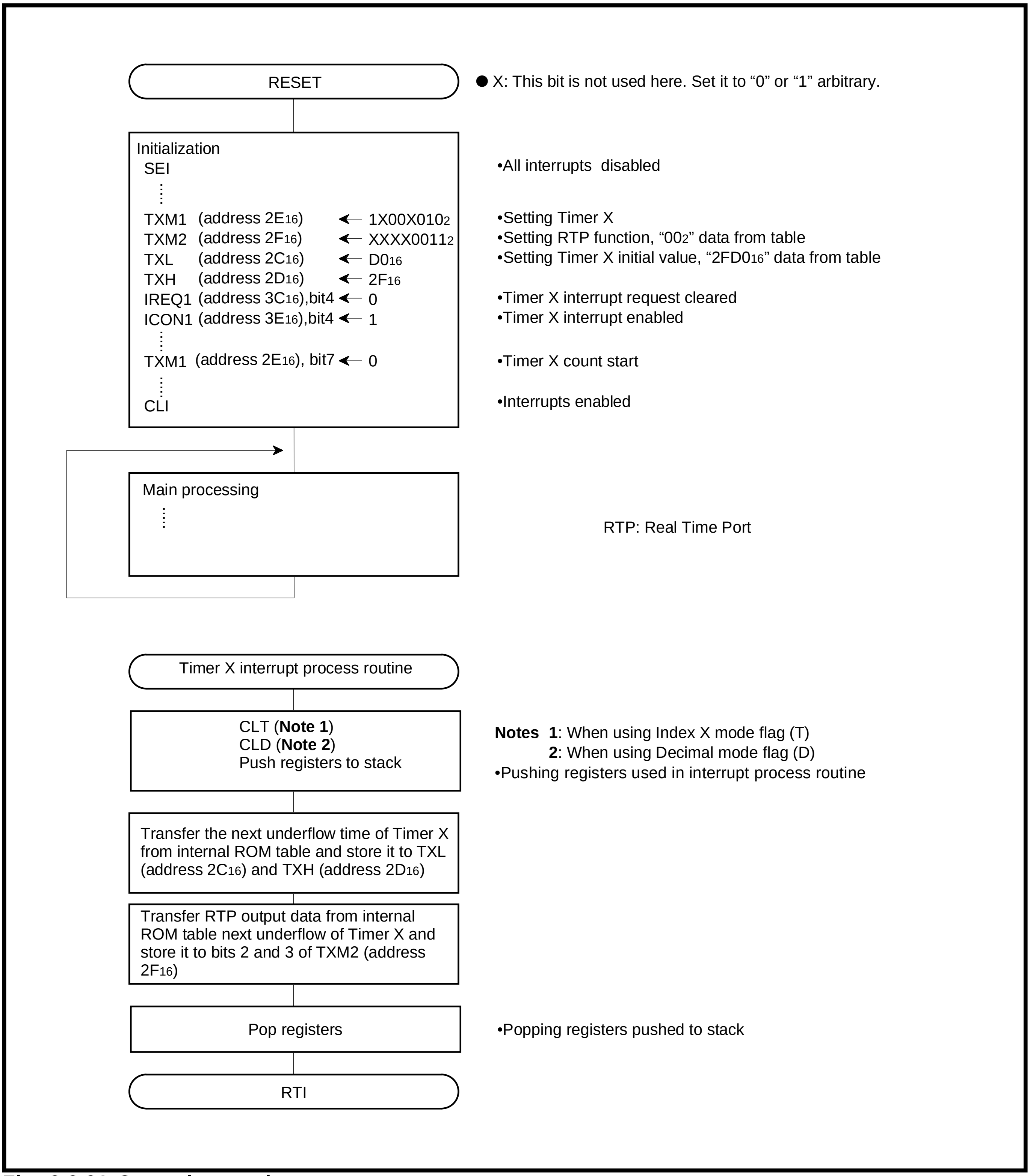


Fig. 2.2.31 Control procedure

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

Address	Register
0018_{16}	Serial I/O1 automatic transfer data pointer (SIO1DP)
0019_{16}	Serial I/O1 control register 1 (SIO1CON1,SC11)
$001A_{16}$	Serial I/O1 control register 2 (SIO1CON2,SC12)
$001B_{16}$	Serial I/O1 register/Transfer counter (SIO1)
$001C_{16}$	Serial I/O1 control register 3 (SIO1CON3,SC13)
$001D_{16}$	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)
0037_{16}	Baud rate generator (BRG)
0038_{16}	UART control register (UARTCON)
0039_{16}	Interrupt source switch register (IFR)
$003C_{16}$	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)
$0EEC_{16}$	Serial I/O3 control register (SIO3CON)
$0EED_{16}$	Serial I/O3 register (SIO3)

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

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 (00_{16} to FF_{16}) 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	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. 2.3.2 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 PB_0–PB_6 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	O	O
1			0	O	O
2	Serial I/O1 synchronous clock selection bits (PB_3/SSTB1 pin control bits)	b3b2 0 0: Internal synchronous clock (PB_3 pin is I/O port.) 0 1: External synchronous clock (PB_3 pin is I/O port.) 1 0: Internal synchronous clock (PB_3 pin is SSTB1 output.) 1 1: Internal synchronous clock (PB_3 pin is SSTB1 output.)	0	O	O
3			0	O	O
4	Serial I/O initialization bit	0: Serial I/O initialization 1: Serial I/O enabled	0	O	O
5	Transfer mode selection bit	0: Full-duplex (transmit/receive) mode (PB_6 pin is SIN1 input.) 1: Transmit-only mode (PB_6 pin is I/O port.)	0	O	O
6	Transfer direction selection bit	0: LSB first 1: MSB first	0	O	O
7	Serial I/O1 clock pin selection bit	0: SCLK11 (PB_0/SCLK12 pin is I/O port.) 1: SCLK12 (PB_4/SCLK11 pin is I/O port.)	0	O	O

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

Serial I/O1 control register 2

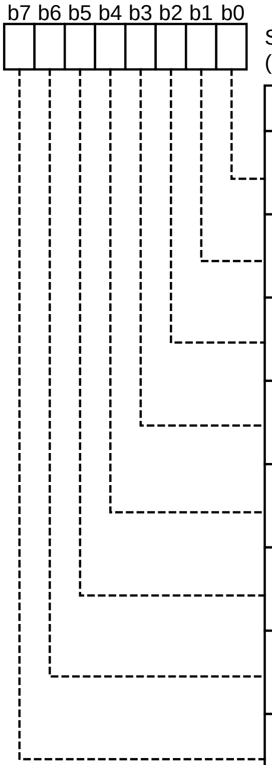
b7 b6 b5 b4 b3 b2 b1 b0

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

b	Name	Functions	At reset	R	W
0	PB1/SRDY1 • PB2/SBUSY1 pin control bits	b3b2b1b0 0 0 0 0: PB1, PB2 pins are I/O ports. 0 0 0 1: Not used 0 0 1 0: PB1 pin is SRDY1 output; PB2 pin is I/O port. 0 0 1 1: PB1 pin is SRDY1 output; PB2 pin is I/O port. 0 1 0 0: PB1 pin is I/O port; PB2 pin is SBUSY1 input. 0 1 0 1: PB1 pin is I/O port; PB2 pin is SBUSY1 input. 0 1 1 0: PB1 pin is I/O port; PB2 pin is SBUSY1 output. 0 1 1 1: PB1 pin is I/O port; PB2 pin is SBUSY1 output. 1 0 0 0: PB1 pin is SRDY1 input; PB2 pin is SBUSY1 output. 1 0 0 1: PB1 pin is SRDY1 input; PB2 pin is SBUSY1 output. 1 0 1 0: PB1 pin is SRDY1 input; PB2 pin is SBUSY1 output. 1 0 1 1: PB1 pin is SRDY1 input; PB2 pin is SBUSY1 output. 1 1 0 0: PB1 pin is SRDY1 output; PB2 pin is SBUSY1 input. 1 1 0 1: PB1 pin is SRDY1 output; PB2 pin is SBUSY1 input. 1 1 1 0: PB1 pin is SRDY1 output; PB2 pin is SBUSY1 input. 1 1 1 1: PB1 pin is SRDY1 output; PB2 pin is SBUSY1 input.	0	O	O
1			0	O	O
2			0	O	O
3			0	O	O
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	O	O
5	Serial transfer status flag	0: Serial transfer completed 1: Serial transfer in-progress	0	O	×
6	SOUT1 pin control bit (when serial data is not transferred)	0: Output active 1: Output high-impedance	0	O	O
7	PB5/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	O	O

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 SCLK1 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. 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 $1C_{16}$)

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 : $f(X_{IN})/4$ or $f(X_{CIN})/8$ 0 0 1 : $f(X_{IN})/8$ or $f(X_{CIN})/16$ 0 1 0 : $f(X_{IN})/16$ or $f(X_{CIN})/32$ 0 1 1 : $f(X_{IN})/32$ or $f(X_{CIN})/64$ 1 0 0 : $f(X_{IN})/64$ or $f(X_{CIN})/128$ 1 0 1 : $f(X_{IN})/128$ or $f(X_{CIN})/256$ 1 1 0 : $f(X_{IN})/256$ or $f(X_{CIN})/512$ 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 37_{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	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. 2.3.7 Structure of Baud rate generator

UART control register

b7 b6 b5 b4 b3 b2 b1 b0

UART control register (UARTCON: address 38_{16})

b	Name	Functions	At reset	R	W
0	Character length selection bit (CHAS)	0: 8 bits 1: 7 bits	0	O	O
1	Parity enable bit (PARE)	0: Parity checking disabled 1: Parity checking enabled	0	O	O
2	Parity selection bit (PARS)	0: Even parity 1: Odd parity	0	O	O
3	Stop bit length selection bit (STPS)	0: 1 stop bit 1: 2 stop bits	0	O	O
4	P6₅/TxD P-channel output disable bit (POFF)	0: CMOS output (in output mode) 1: N-channel open-drain output (in output mode)	0	O	O
5	BRG clock switch bit	0: X_{IN} or X_{CIN}/2 (depending on internal system clock) 1: X_{CIN}	0	O	O
6	Serial I/O2 clock I/O pin selection bit	0: S_{CLK21} ($P6_7$/S_{CLK22} pin is used as I/O port or $\overline{S_{RDY2}}$ output pin.) 1: S_{CLK22} ($P6_6$/S_{CLK21} pin is used as I/O port.)	0	O	O
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “1”.		1	O	×

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: $f(X_{IN})$ 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	$\overline{\text{SRDY2}}$ output enable bit (SRDY)	0: $P6_7$ pin operates as normal I/O pin 1: $P6_7$ pin operates as $\overline{\text{SRDY2}}$ 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 $P6_4$–$P6_7$ operate as normal I/O pins) 1: Serial I/O2 enabled (pins $P6_4$–$P6_7$ operate as serial I/O pins)	0	○	○

Fig. 2.3.9 Structure of Serial I/O2 control register

Serial I/O2 status register

b7 b6 b5 b4 b3 b2 b1 b0

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

Fig. 2.3.10 Structure of Serial I/O2 status register

Serial I/O2 transmit/receive buffer register

b7 b6 b5 b4 b3 b2 b1 b0

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

(3) Serial I/O3

Serial I/O3 control register

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O3 control register (SIO3CON: address 0EEC16)

b	Name	Functions	At reset	R	W
0	Internal synchronous clock selection bits	b2b1b0 000: f(XIN)/4 or f(XCIN)/8 001: f(XIN)/8 or f(XCIN)/16 010: f(XIN)/16 or f(XCIN)/32 011: f(XIN)/32 or f(XCIN)/64 110: f(XIN)/64 or f(XCIN)/128 111: f(XIN)/128 or f(XCIN)/256	0	O	O
1			0	O	O
2			0	O	O
3	Serial I/O3 port selection bit (P91, P92)	0: I/O port 1: SOUT3, SCLK3 signal output	0	O	O
4	SRDY3 output selection bit (P93)	0: I/O port 1: SRDY3 signal output	0	O	O
5	Transfer direction selection bit	0: LSB first 1: MSB first	0	O	O
6	Synchronous clock selection bit	0: External clock 1: Internal clock	0	O	O
7	P91/SOUT3 P-channel output disable bit (P91)	0: CMOS output (in output mode) 1: N-channel open drain output (in output mode)	0	O	O

Fig. 2.3.12 Structure of Serial I/O3 control register

Serial I/O3 register

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O3 register (SIO3: address 0EED16)

b	Functions	At reset	R	W
0	This is the buffer register which is used to write transmit data or to read receive data. When selecting an internal clock, the serial transfer is started by writing this register.	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. 2.3.13 Structure of Serial I/O3 register

(4) 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	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	INT_1/serial I/O3 interrupt switch bit	0: INT_1 intrrupt 1: Serial I/O3 interrupt	0	○	○
3	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

Fig. 2.3.14 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/serial I/O3 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. 2.3.15 Structure of Interrupt request register 1

Interrupt request register 2

b7 b6 b5 b4 b3 b2 b1 b0

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

b	Name	Functions	At reset	R	W
0	Timer 4 interrupt request bit	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	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.

Fig. 2.3.16 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	O	O
1	INT_1/serial I/O3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
2	INT_2 interrupt enable bit Remote controller /counter overflow interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
3	Serial I/O1 interrupt enable bit Serial I/O automatic transfer interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
4	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
6	Timer 2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
7	Timer 3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O

Fig. 2.3.17 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	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
1	Timer 5 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
2	Timer 6 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
3	Serial I/O2 receive interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
4	INT_3/Serial I/O2 transmit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
5	INT_4 interrupt enable bit A-D converter interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
6	FLD blanking interrupt enable bit FLD digit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
7	Fix “0” to this bit.		0	O	O

Fig. 2.3.18 Structure of Interrupt control register 2

2.3.3 Serial I/O1 connection examples

(1) Control of peripheral IC equipped with CS pin

Figure 2.3.19 shows connection examples with peripheral ICs equipped with the CS pin. All examples can use the automatic transfer function.

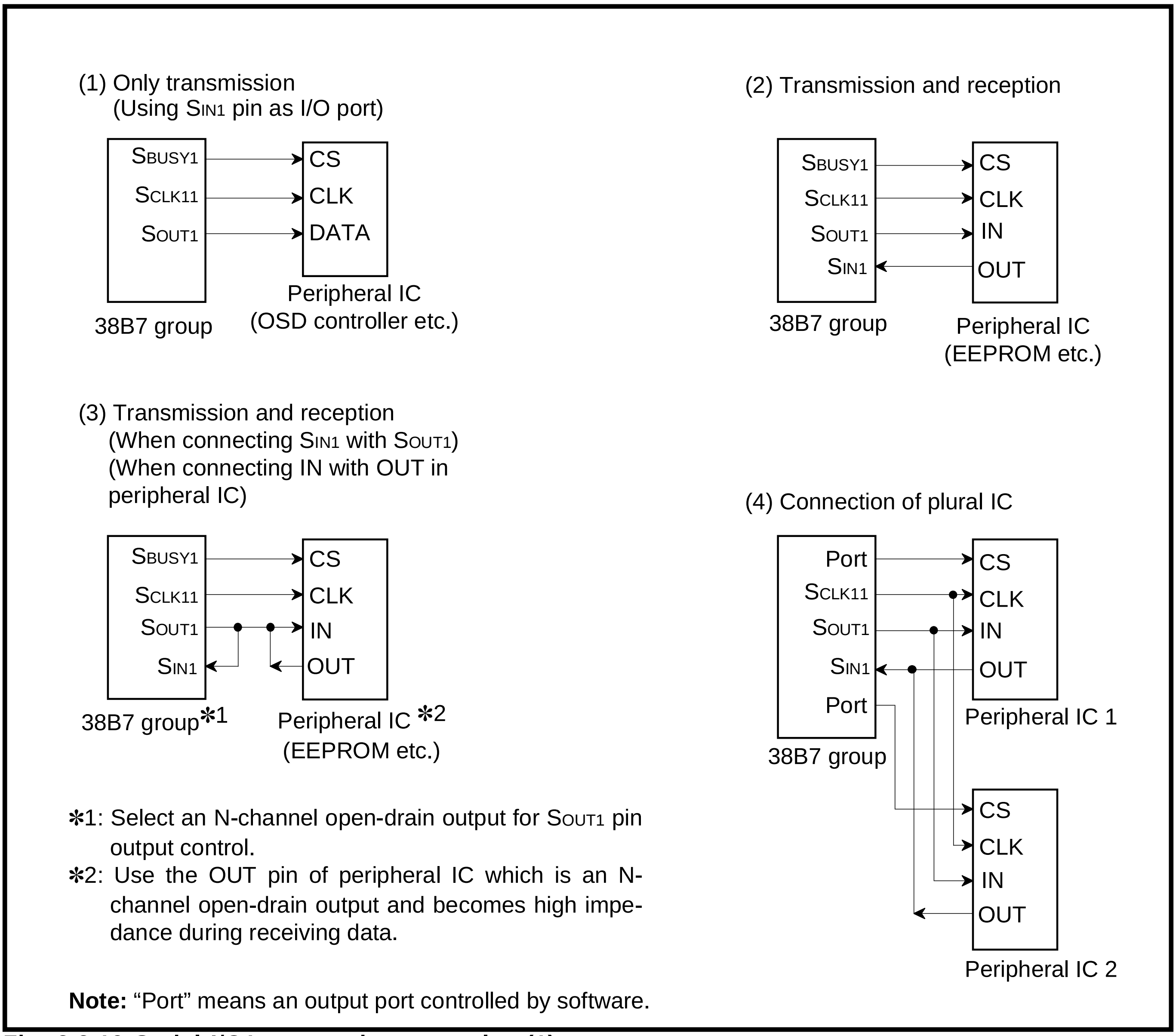


Fig. 2.3.19 Serial I/O1 connection examples (1)

(2) Connection with microcomputer

Figure 2.3.20 shows connection examples with another microcomputer.

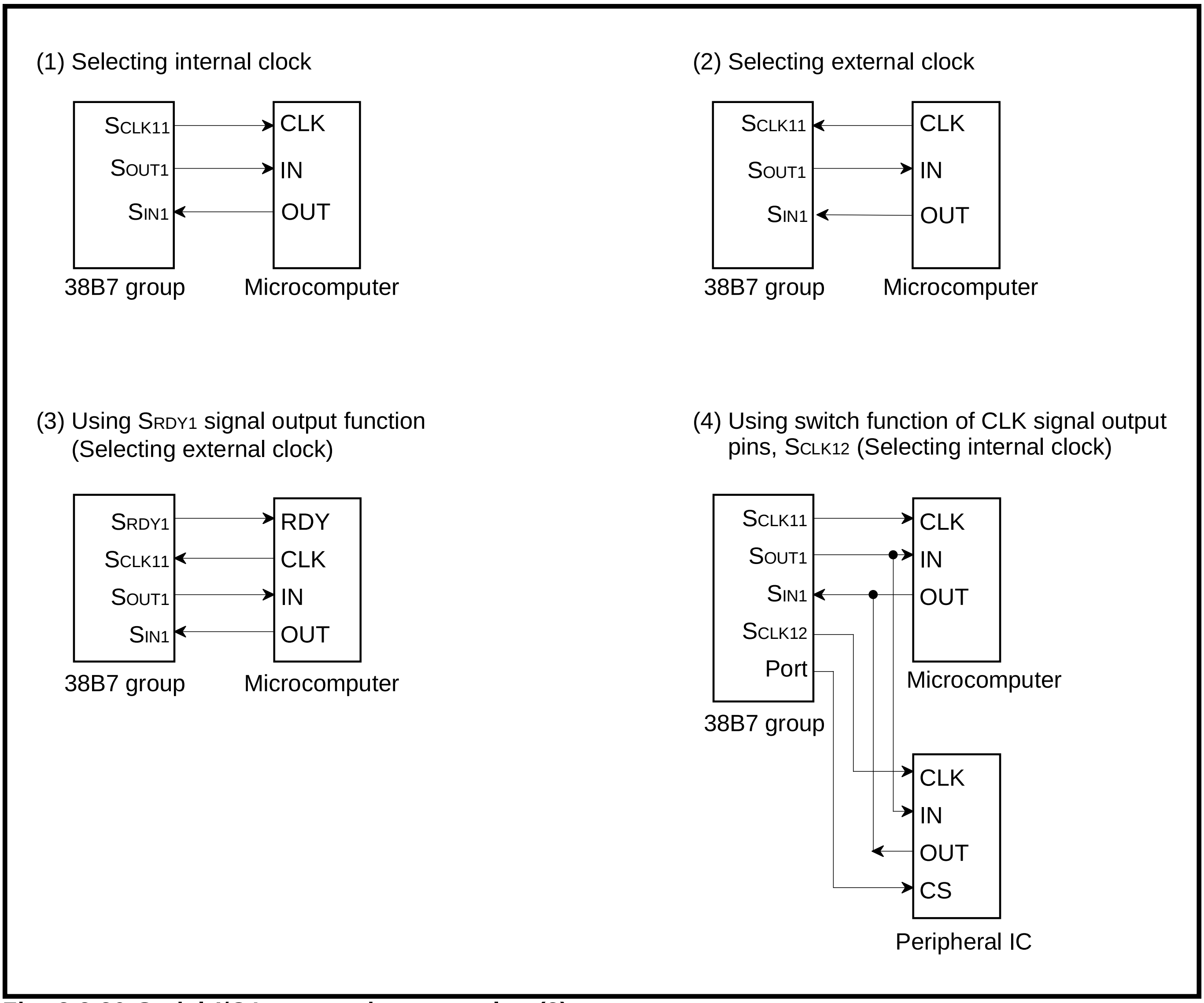


Fig. 2.3.20 Serial I/O1 connection examples (2)

2.3.4 Serial I/O1's modes

Figure 2.3.21 shows the serial I/O1's modes.

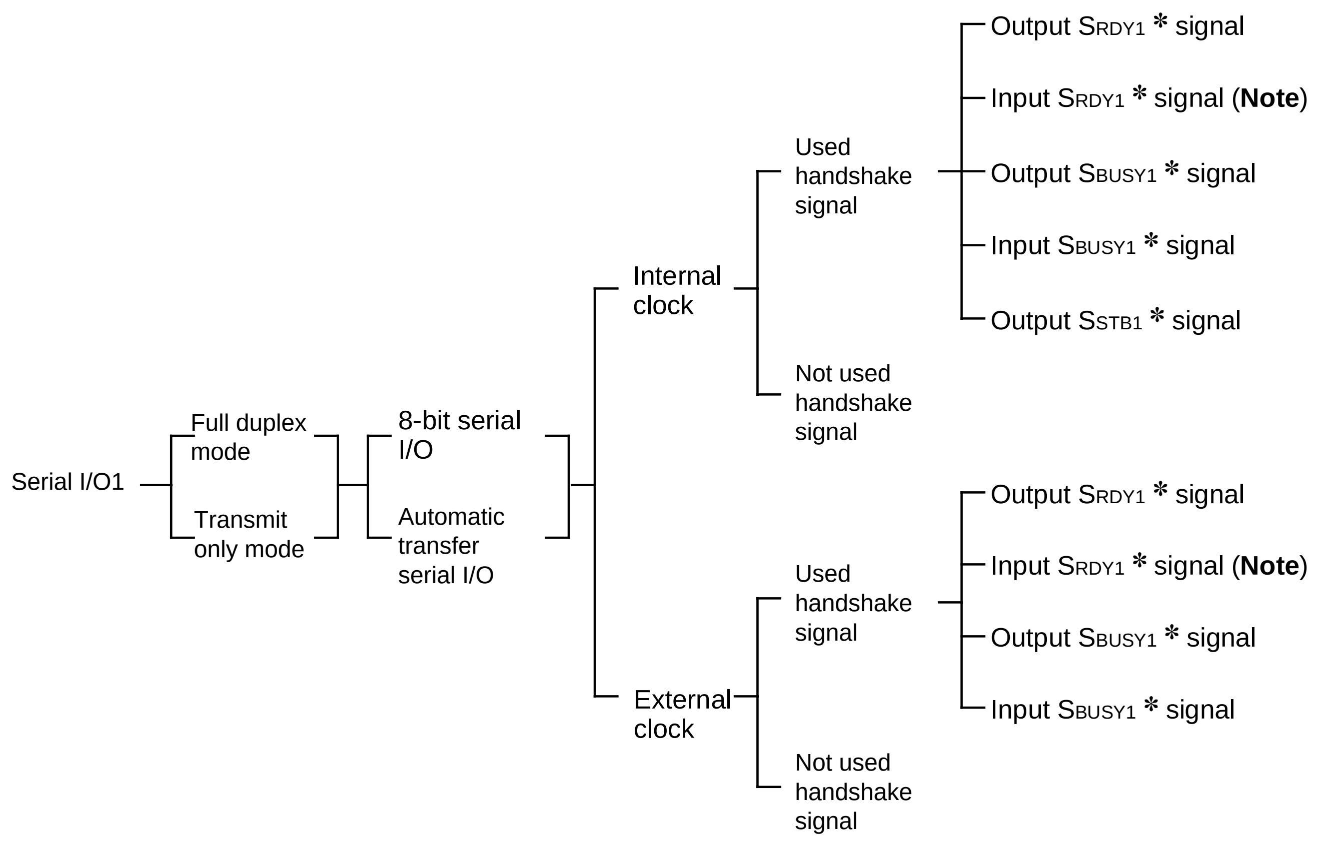


Fig. 2.3.21 Serial I/O1's modes

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 2.3.22 shows a connection diagram, and Figure 2.3.23 shows a timing chart.

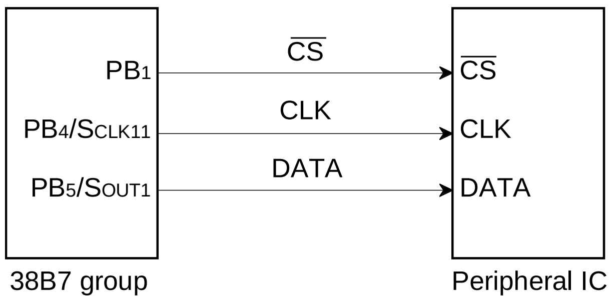


Fig. 2.3.22 Connection diagram

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 PB_1 is connected to the $\overline{CS}$ pin (“L” active) of the peripheral IC for transmission control; the output level of port PB_1 is controlled by software.

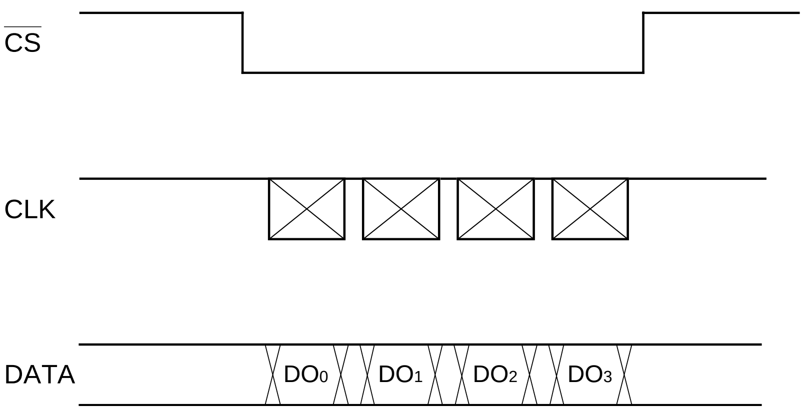


Fig. 2.3.23 Timing chart

Figure 2.3.24 shows the registers setting relevant to the transmission side, and Figure 2.3.25 shows the setting of transmission data.

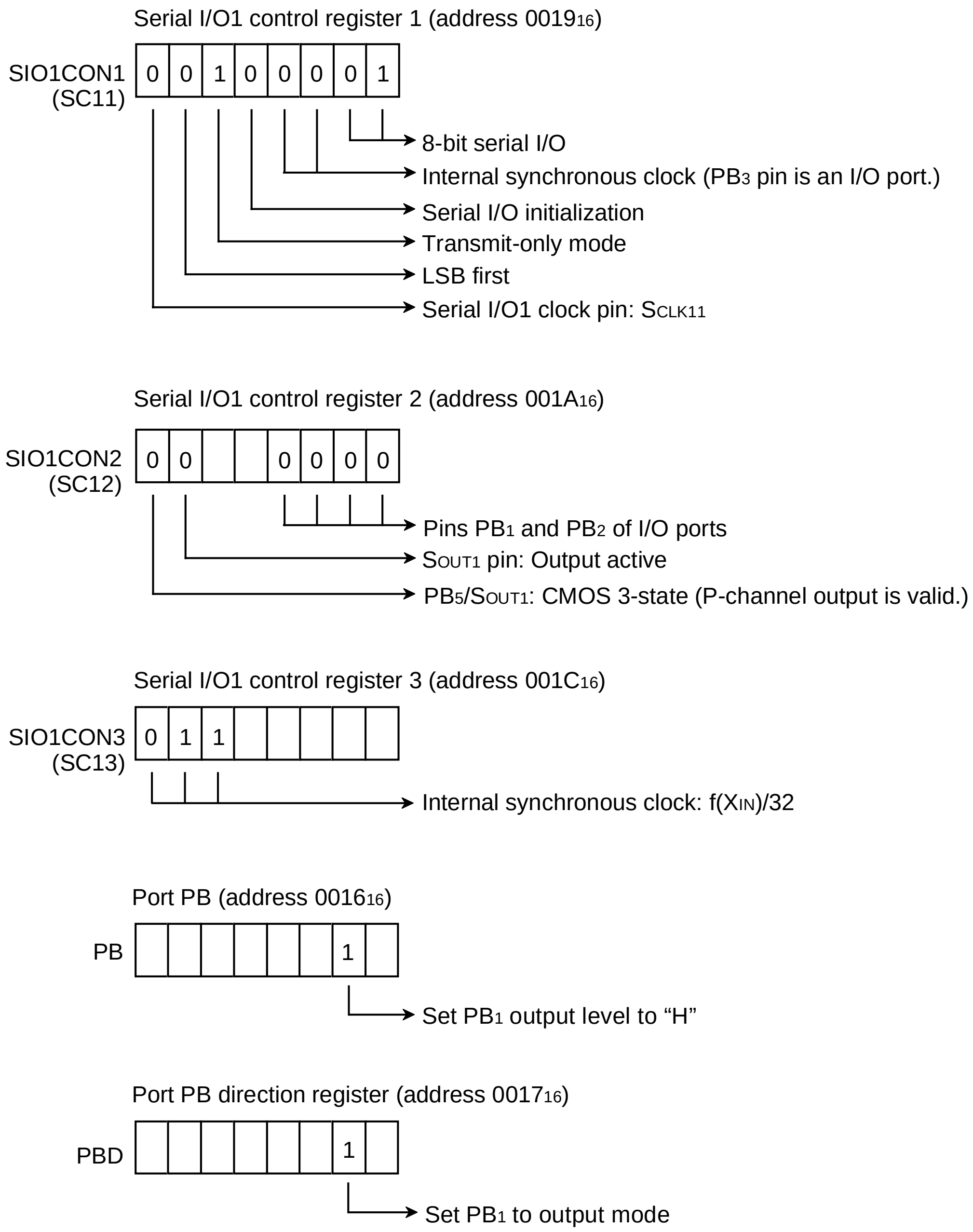


Fig. 2.3.24 Registers setting relevant to transmission side

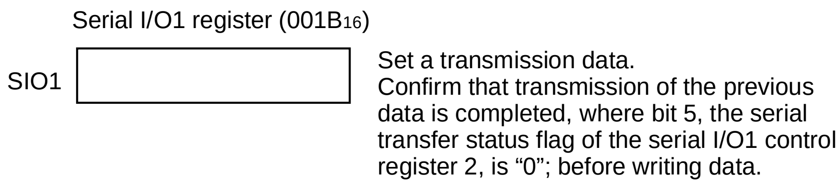


Fig. 2.3.25 Setting of transmission data

Control procedure: When the registers are set as shown in Figure 2.3.24, 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 2.3.26 shows a control procedure.

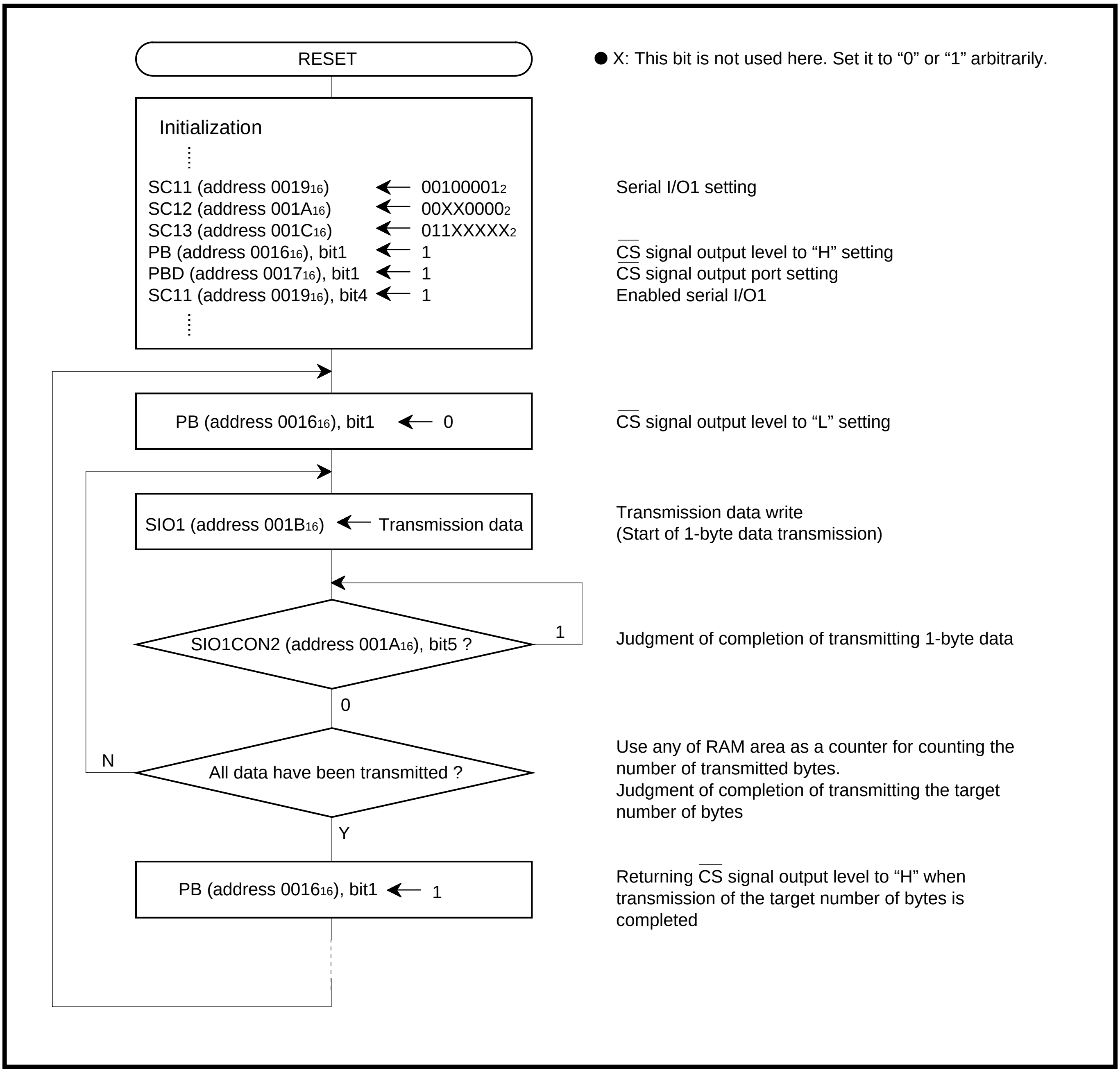


Fig. 2.3.26 Control procedure

(2) Transmission/Reception using automatic transfer

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

Figure 2.3.27 shows a connection diagram, and Figure 2.3.28 shows a timing chart of serial data transmission/reception.

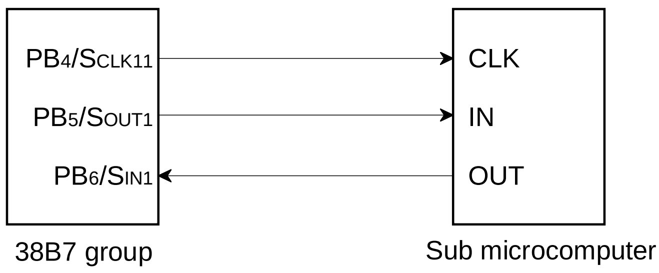


Fig. 2.3.27 Connection diagram

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 2.3.29 shows the relevant registers setting, and Figure 2.3.30 shows the control procedure.

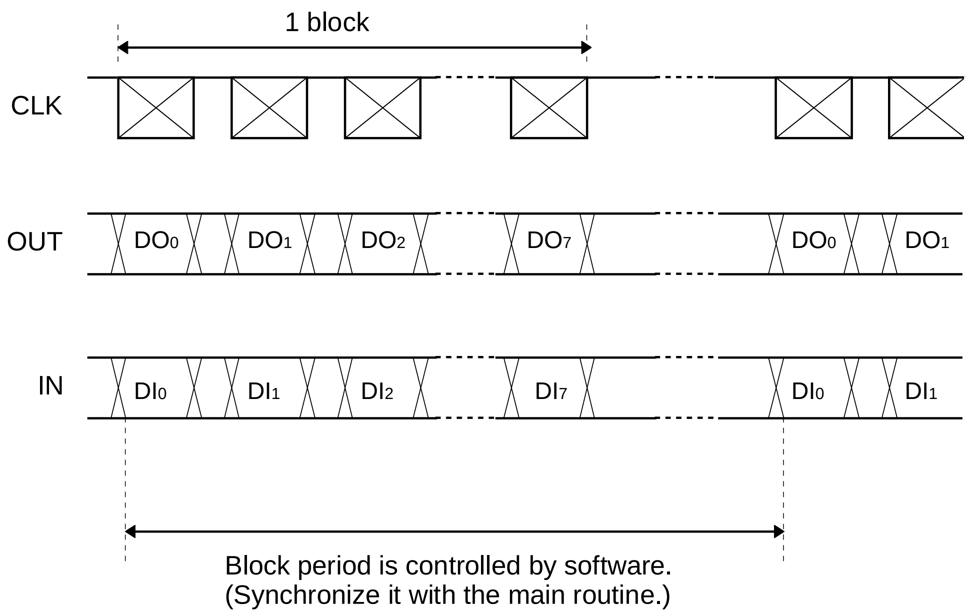


Fig. 2.3.28 Timing chart of serial data transmission/reception

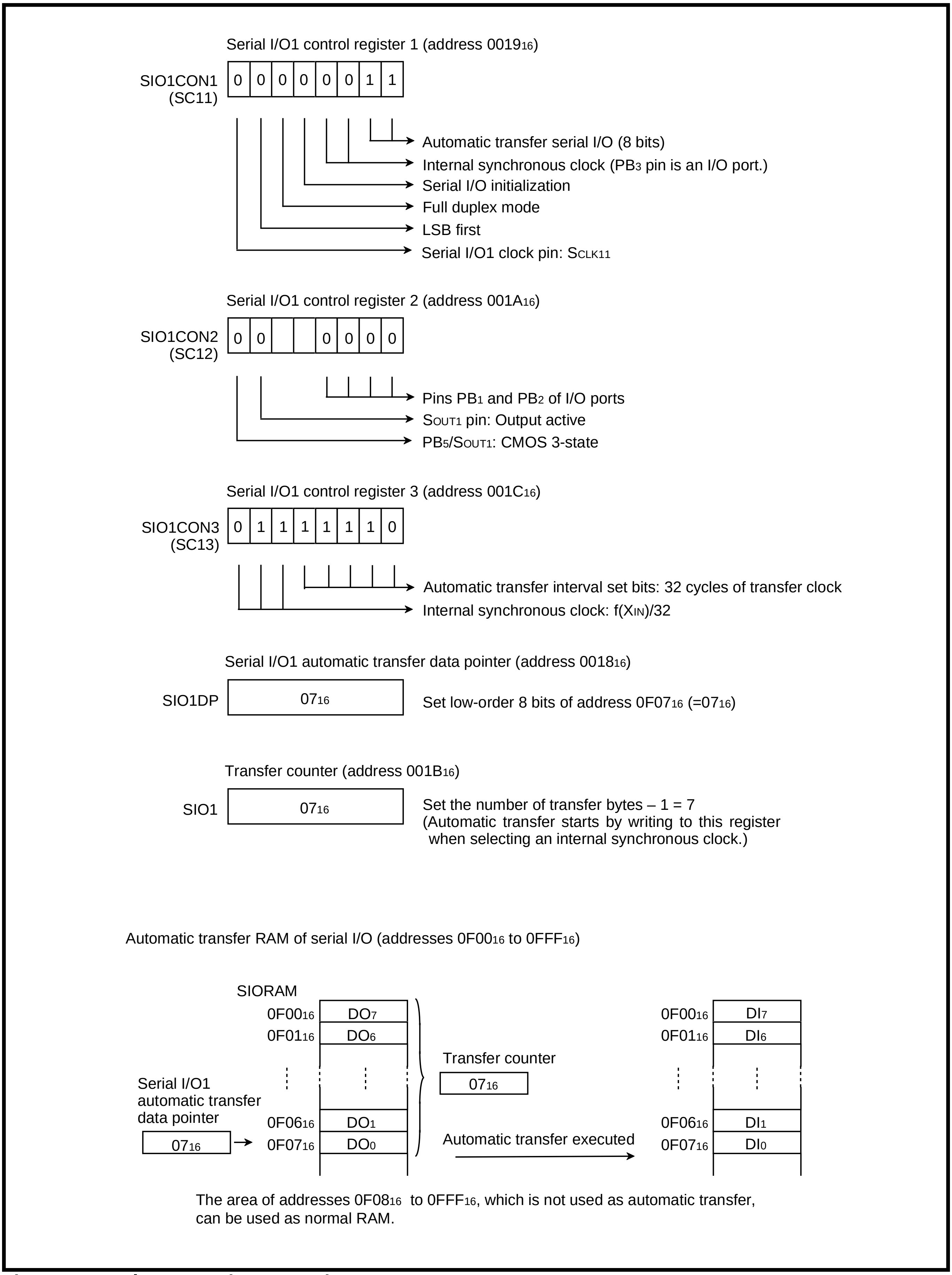


Fig. 2.3.29 Relevant registers setting

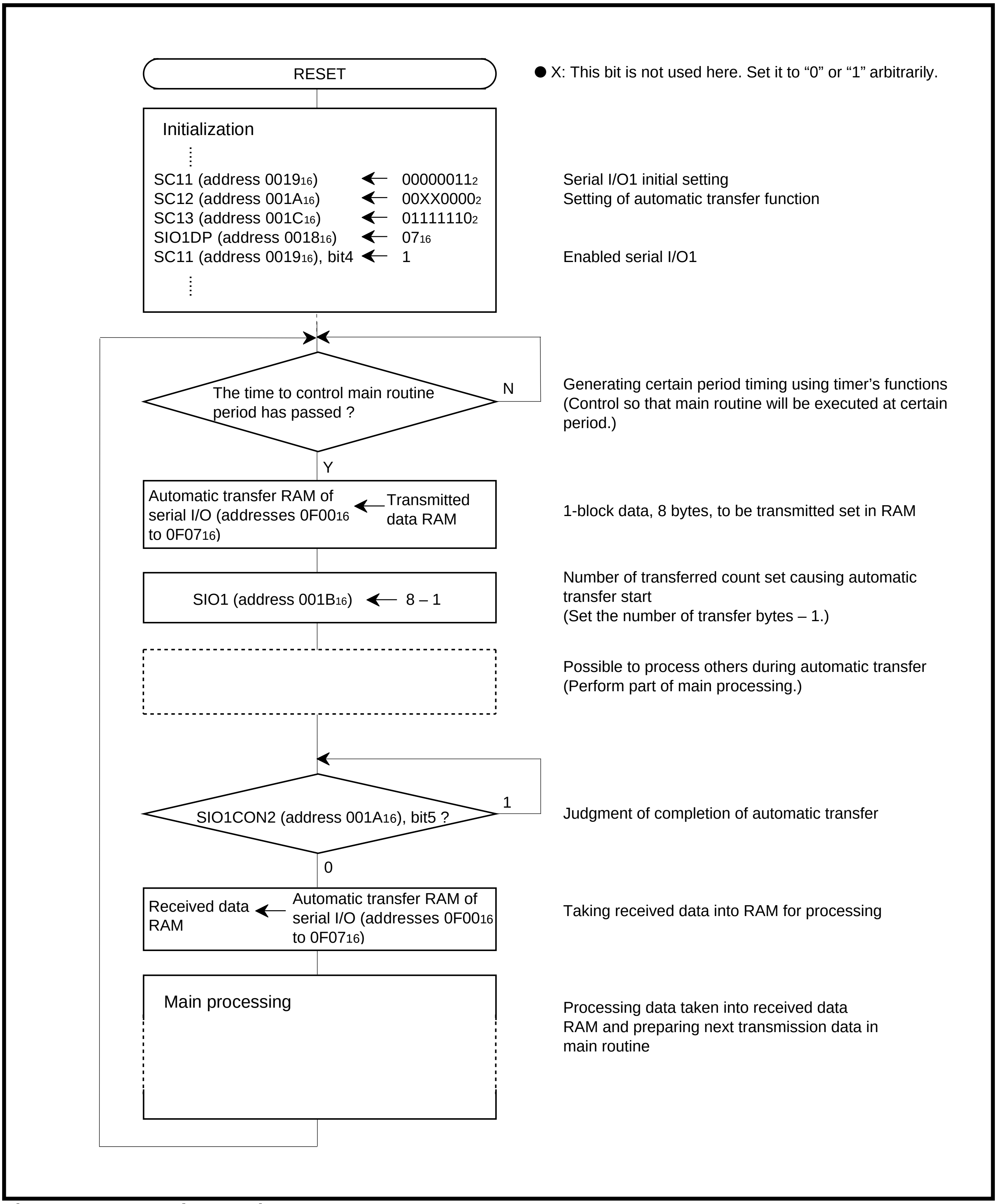
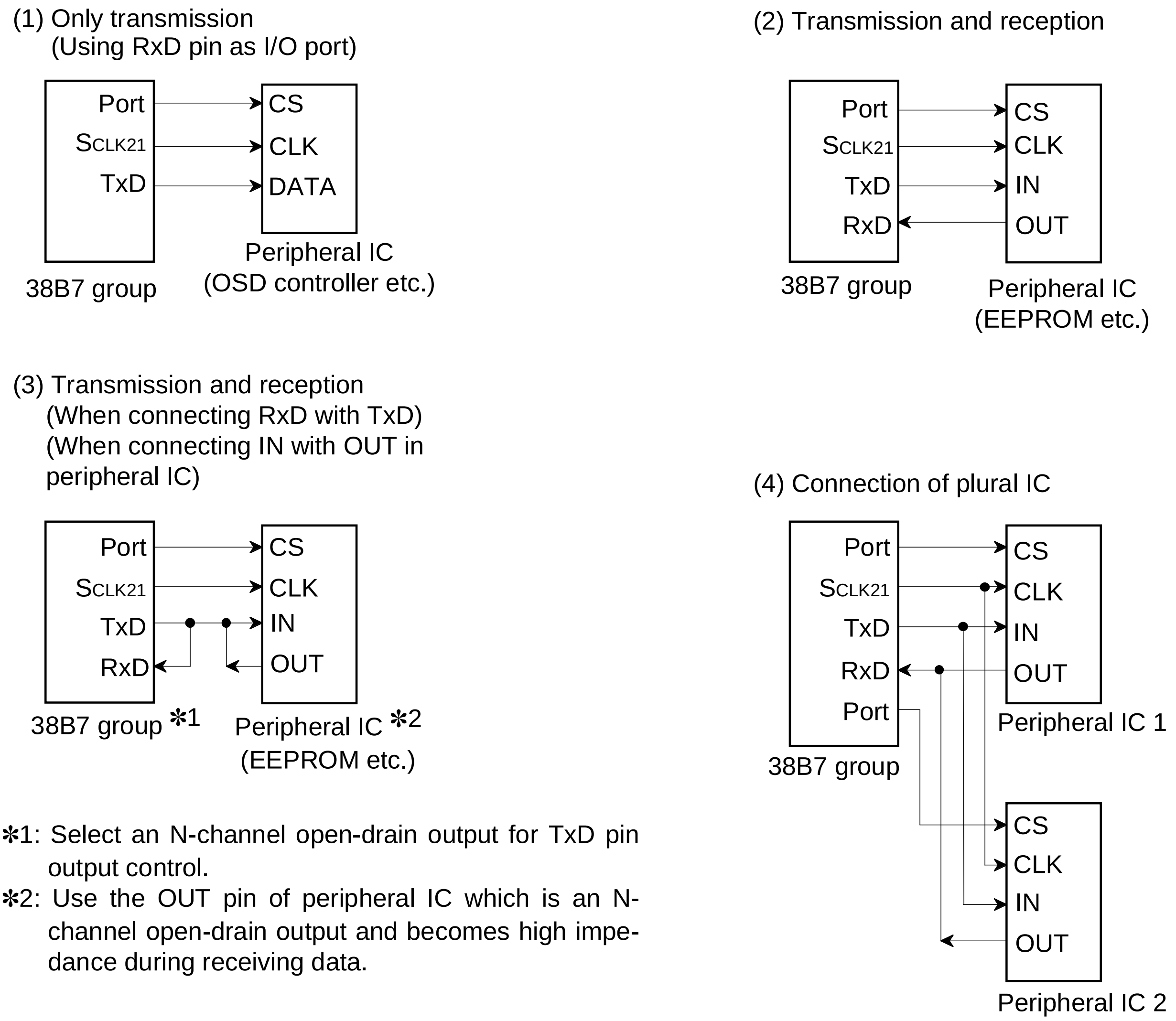


Fig. 2.3.30 Control procedure

2.3.6 Serial I/O2 connection examples

(1) Control of peripheral IC equipped with CS pin

Figure 2.3.31 shows connection examples with peripheral ICs equipped with the CS pin.



*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. 2.3.31 Serial I/O2 connection examples (1)

(2) Connection with microcomputer

Figure 2.3.32 shows connection examples with another microcomputer.

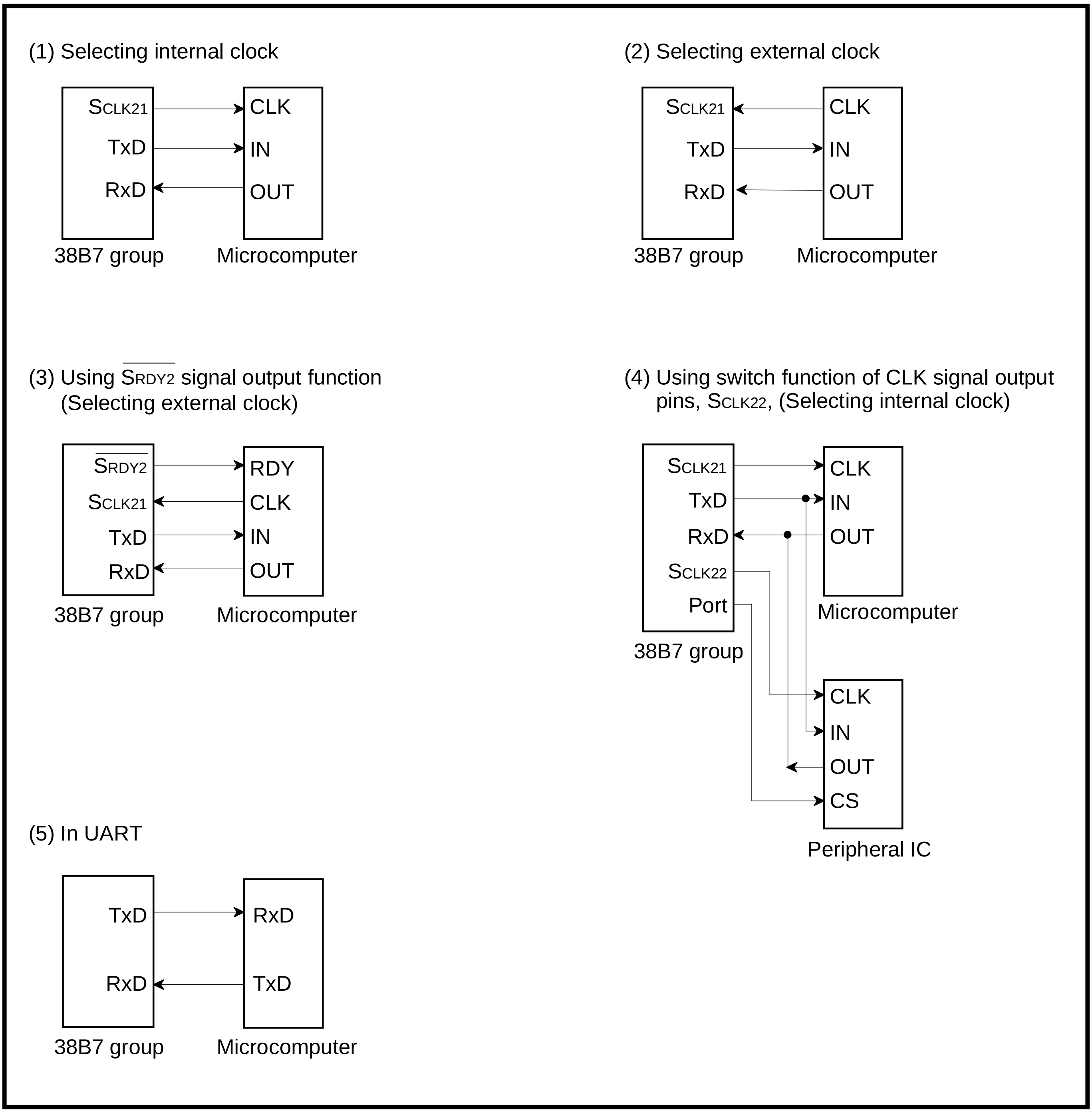


Fig. 2.3.32 Serial I/O2 connection examples (2)

2.3.7 Serial I/O2's modes

A clock synchronous or clock asynchronous (UART) can be selected for the serial I/O2.
Figure 2.3.33 shows the serial I/O2's modes, and Figure 2.3.34 shows the serial I/O2 transfer data format.

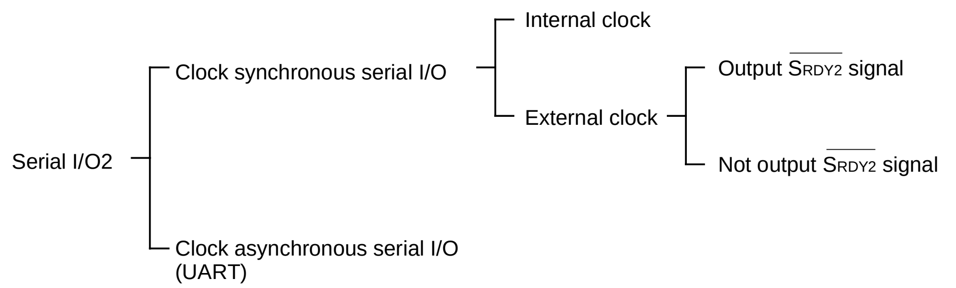


Fig. 2.3.33 Serial I/O2's modes

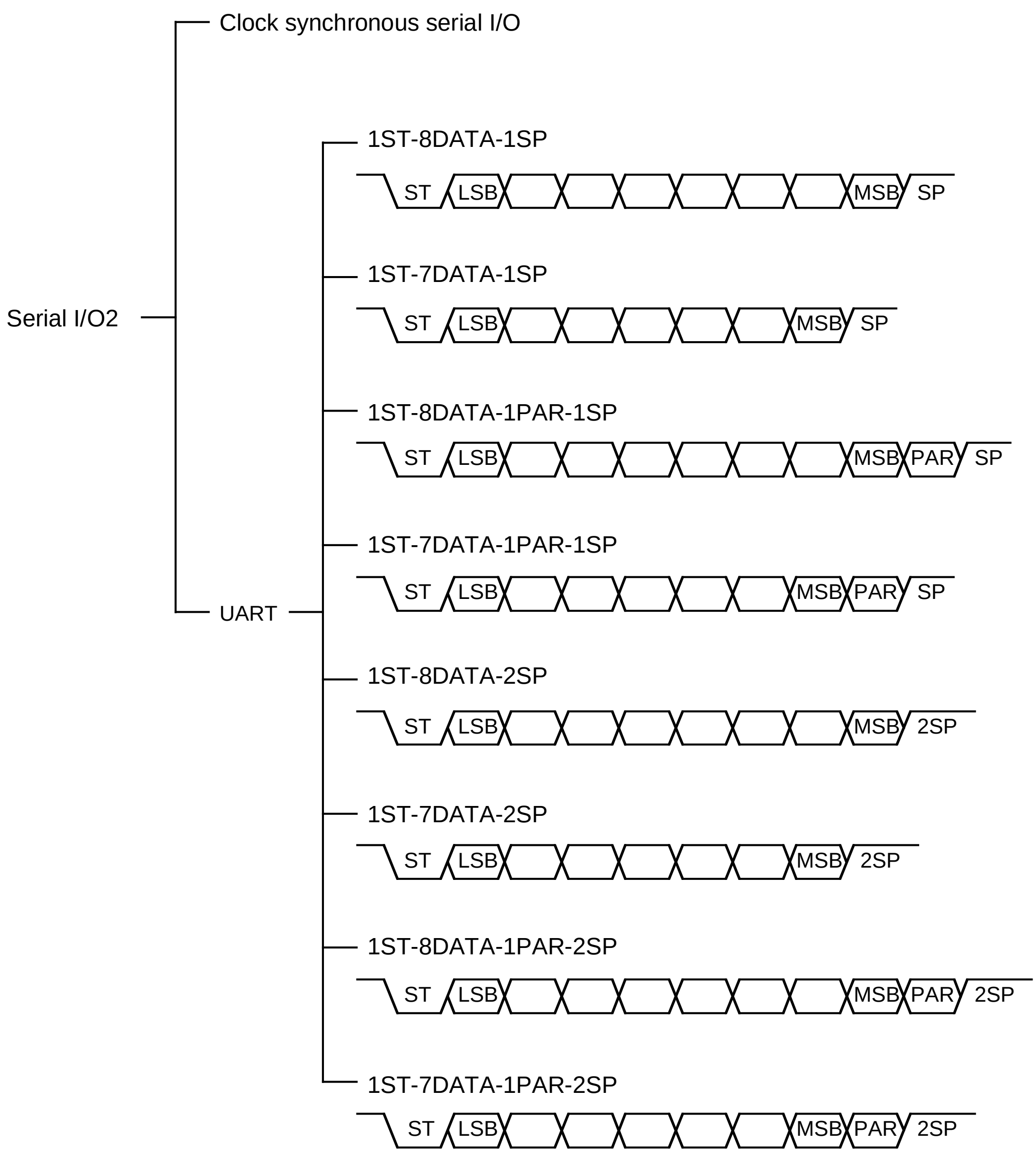


Fig. 2.3.34 Serial I/O2 transfer data format

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 2.3.35 shows a connection diagram, and Figure 2.3.36 shows a timing chart.

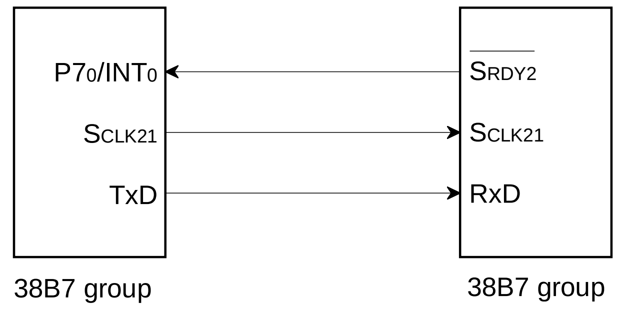


Fig. 2.3.35 Connection diagram

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.

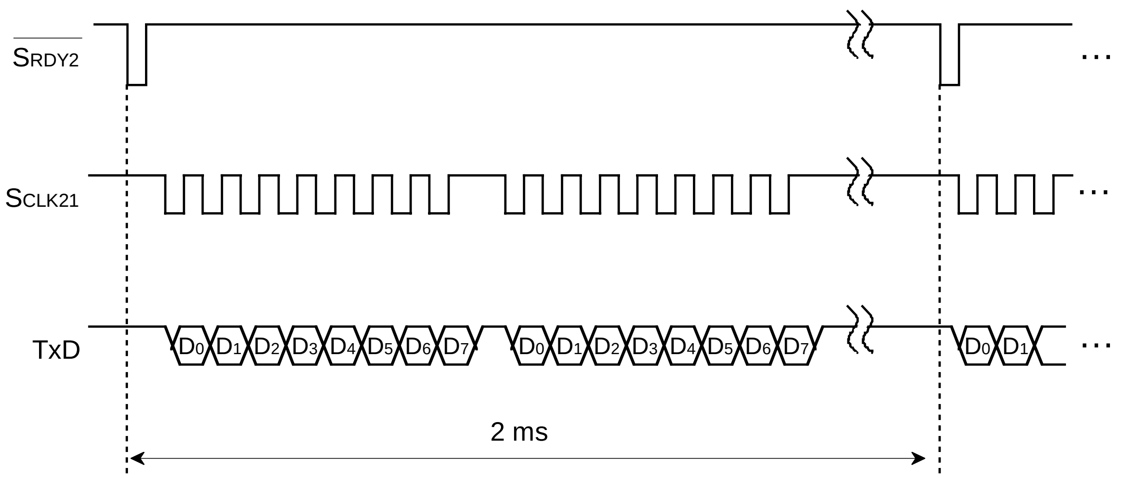


Fig. 2.3.36 Timing chart

Figure 2.3.37 shows the registers setting relevant to the transmission side, and Figure 2.3.38 shows the registers setting relevant to the reception side.

Transmission side

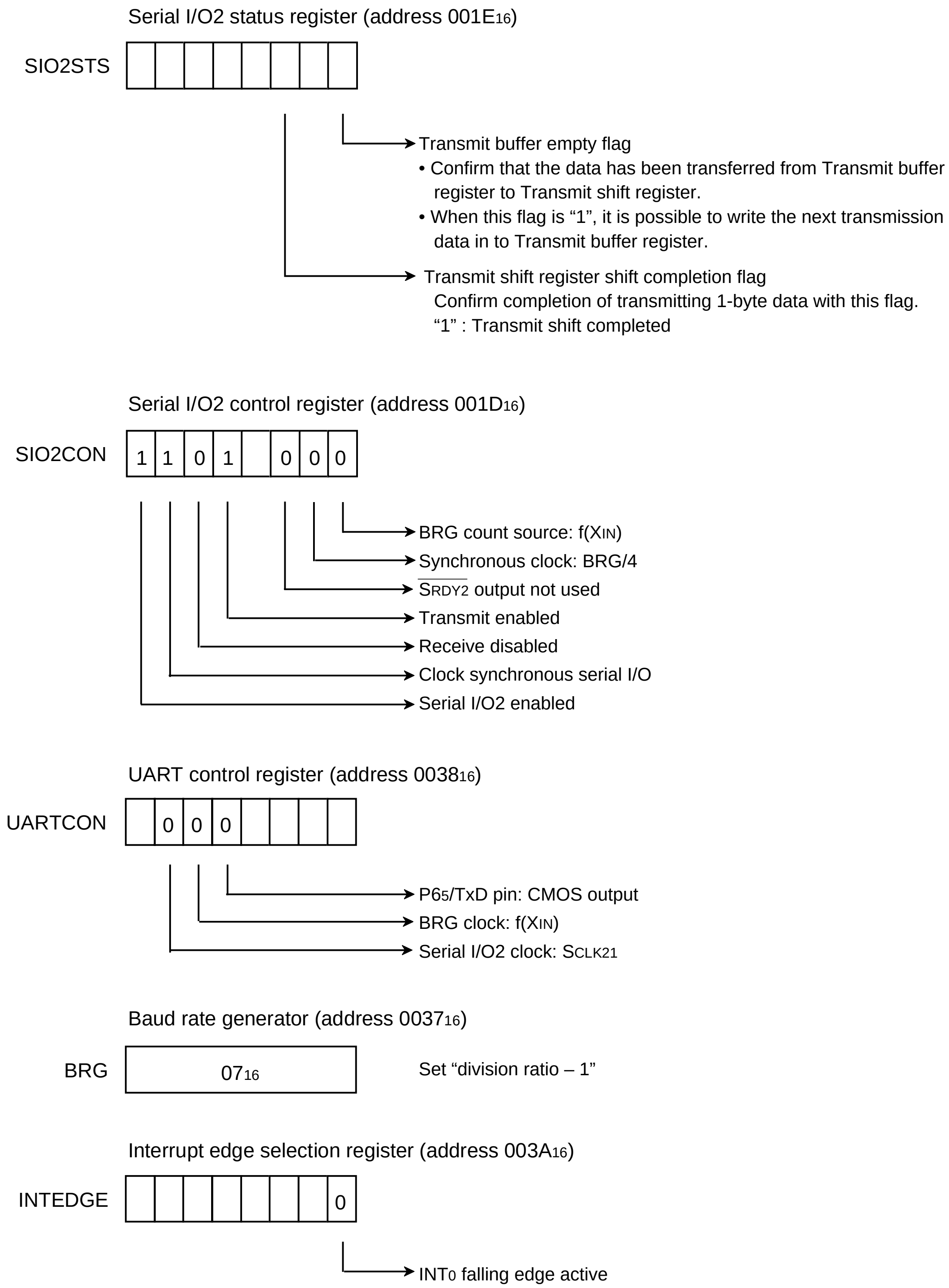


Fig. 2.3.37 Registers setting relevant to transmission side

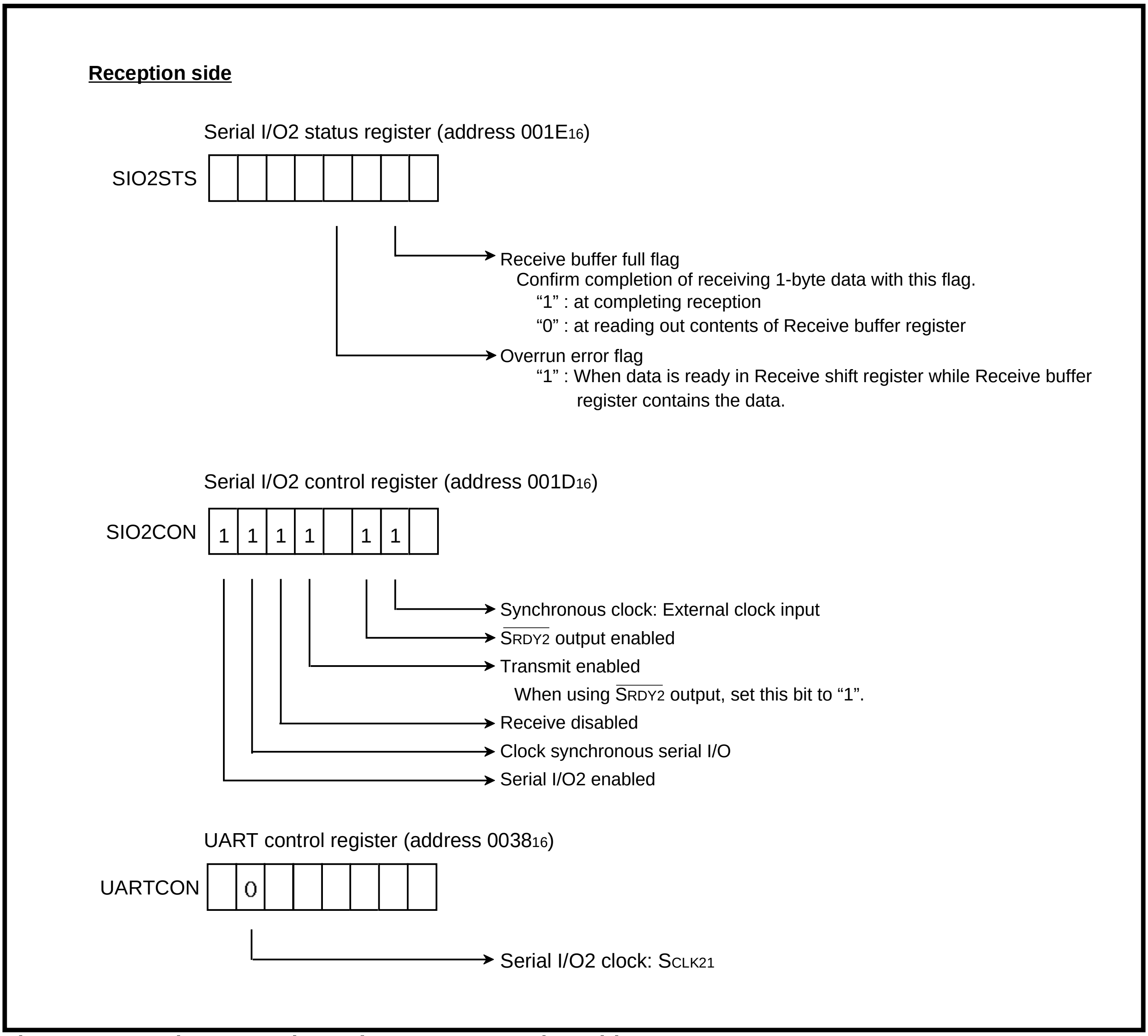


Fig. 2.3.38 Registers setting relevant to reception side

Figure 2.3.39 shows a control procedure of the transmission side, and Figure 2.3.40 shows a control procedure of the reception side.

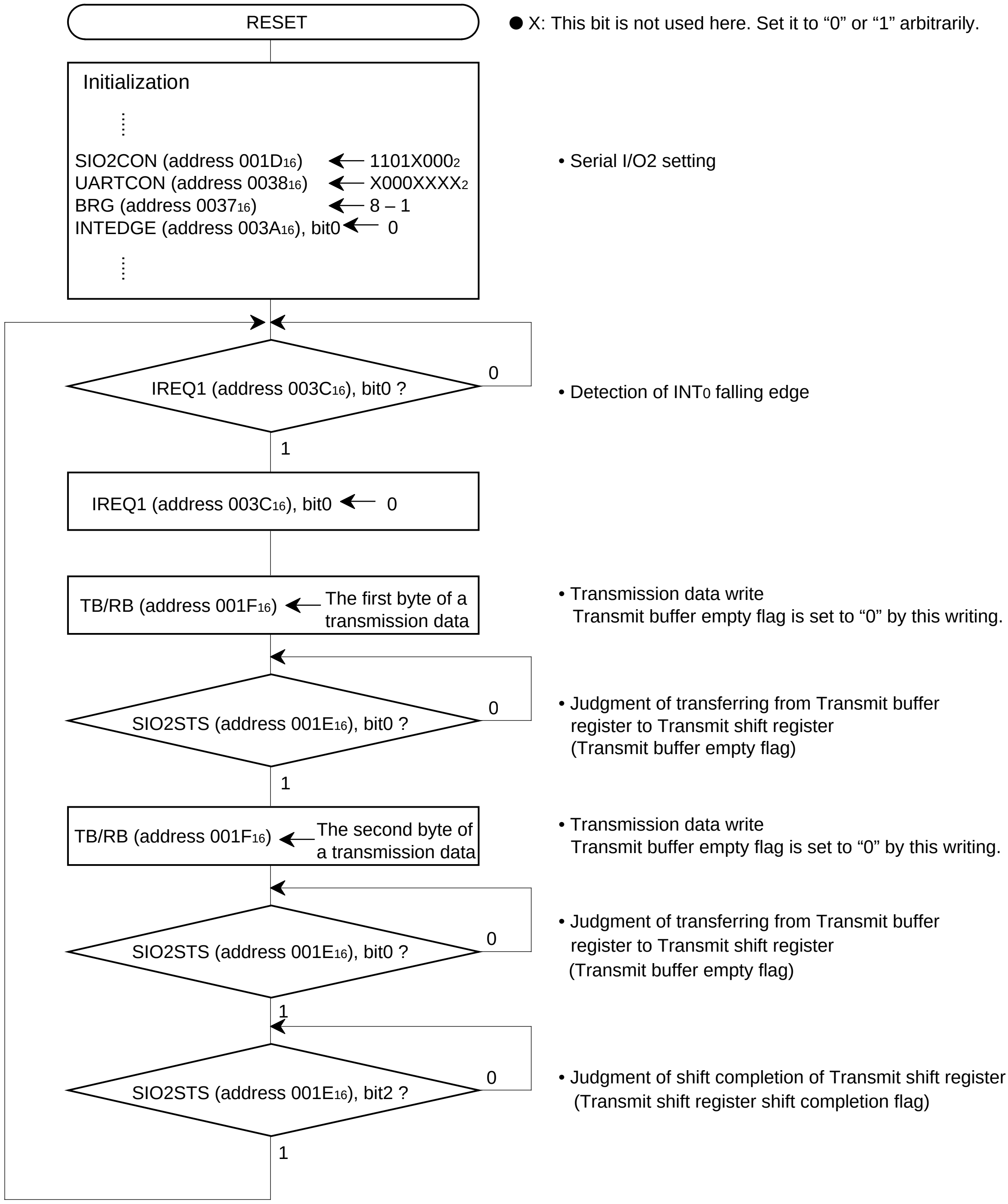


Fig. 2.3.39 Control procedure of transmission side

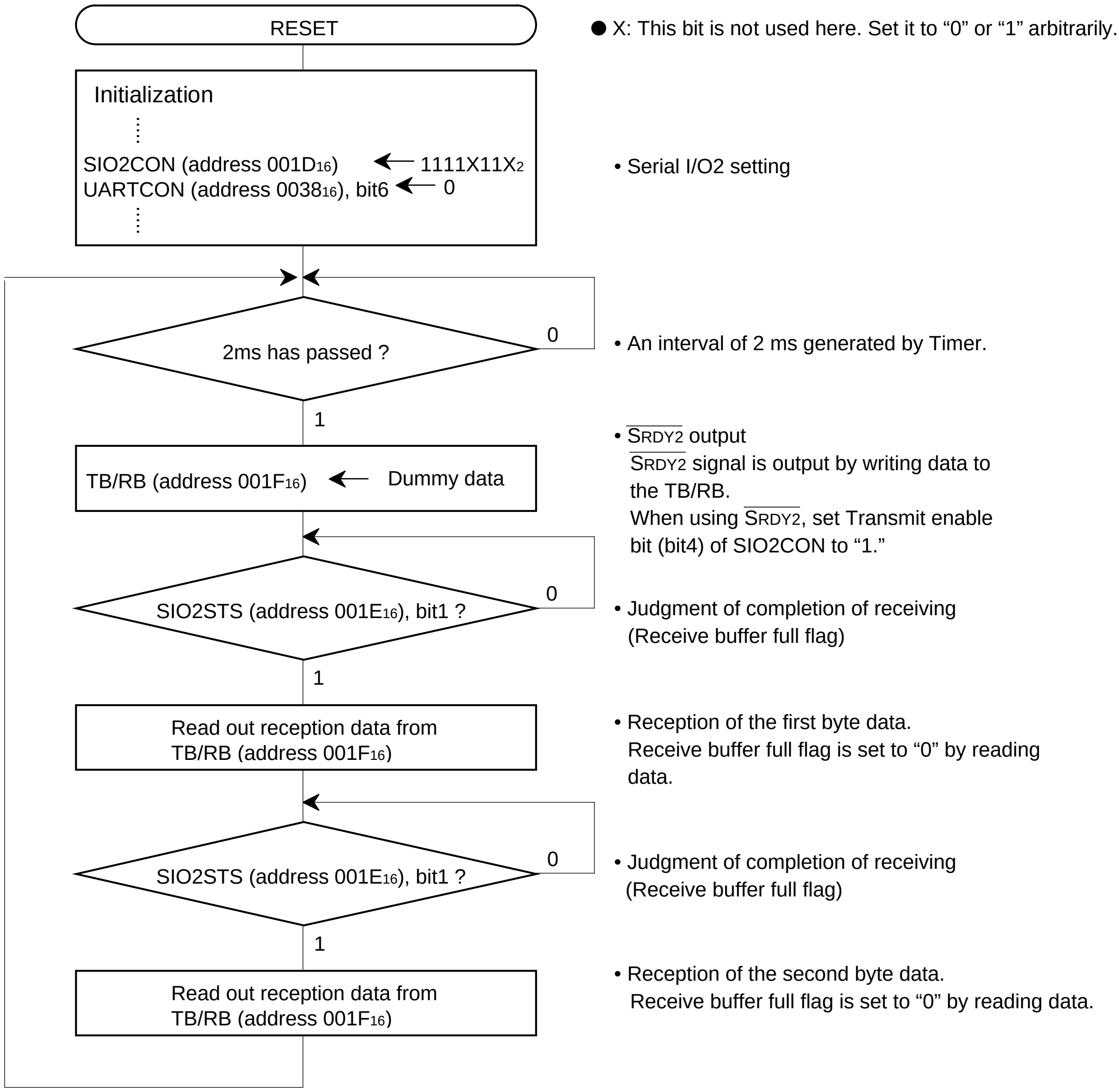


Fig. 2.3.40 Control procedure of reception side

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

Outline : Serial communication is performed, connecting port $P7_7$ with the $\overline{CS}$ pin of a peripheral IC.

Figure 2.3.41 shows a connection diagram, and Figure 2.3.42 shows a timing chart.

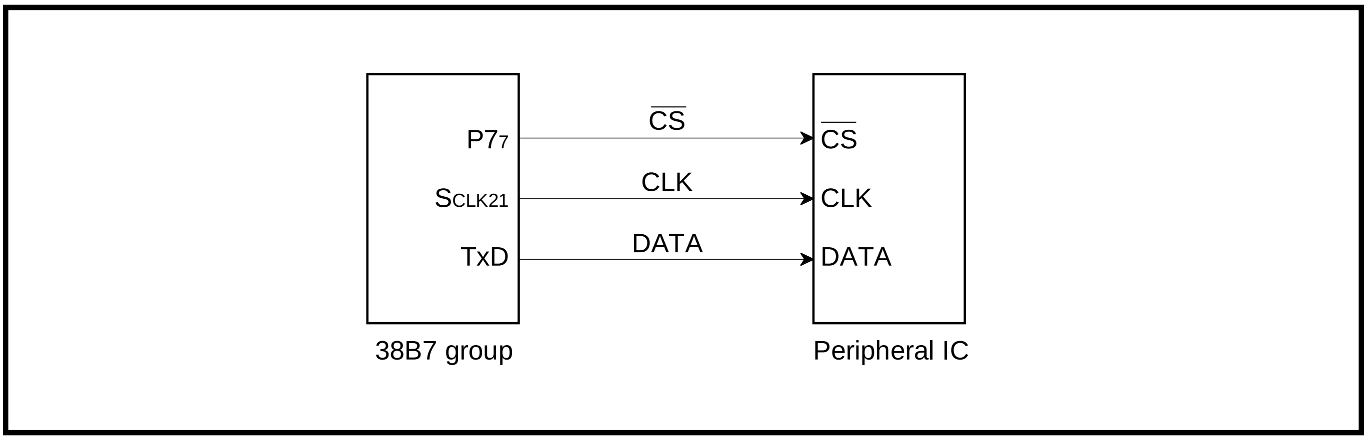


Fig. 2.3.41 Connection diagram

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 $P7_7$ is connected with the $\overline{CS}$ pin ("L" active) of the peripheral IC for transmission control; the output level of port $P7_7$ is controlled by software.

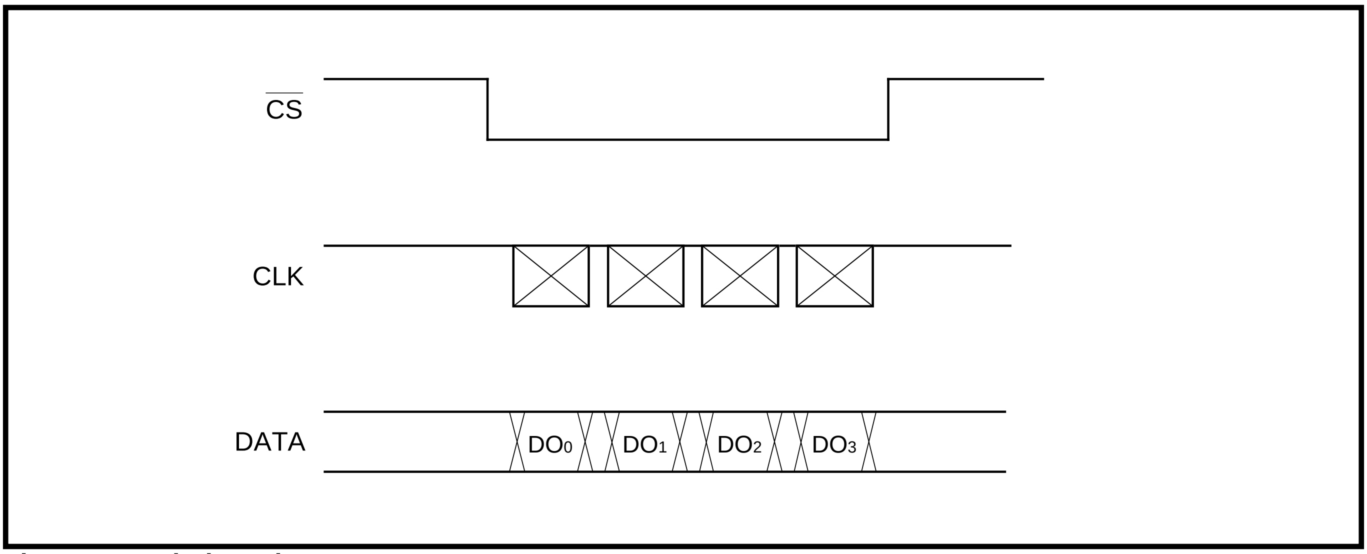


Fig. 2.3.42 Timing chart

Figure 2.3.43 shows the relevant registers setting and Figure 2.3.44 shows the setting of transmission data.

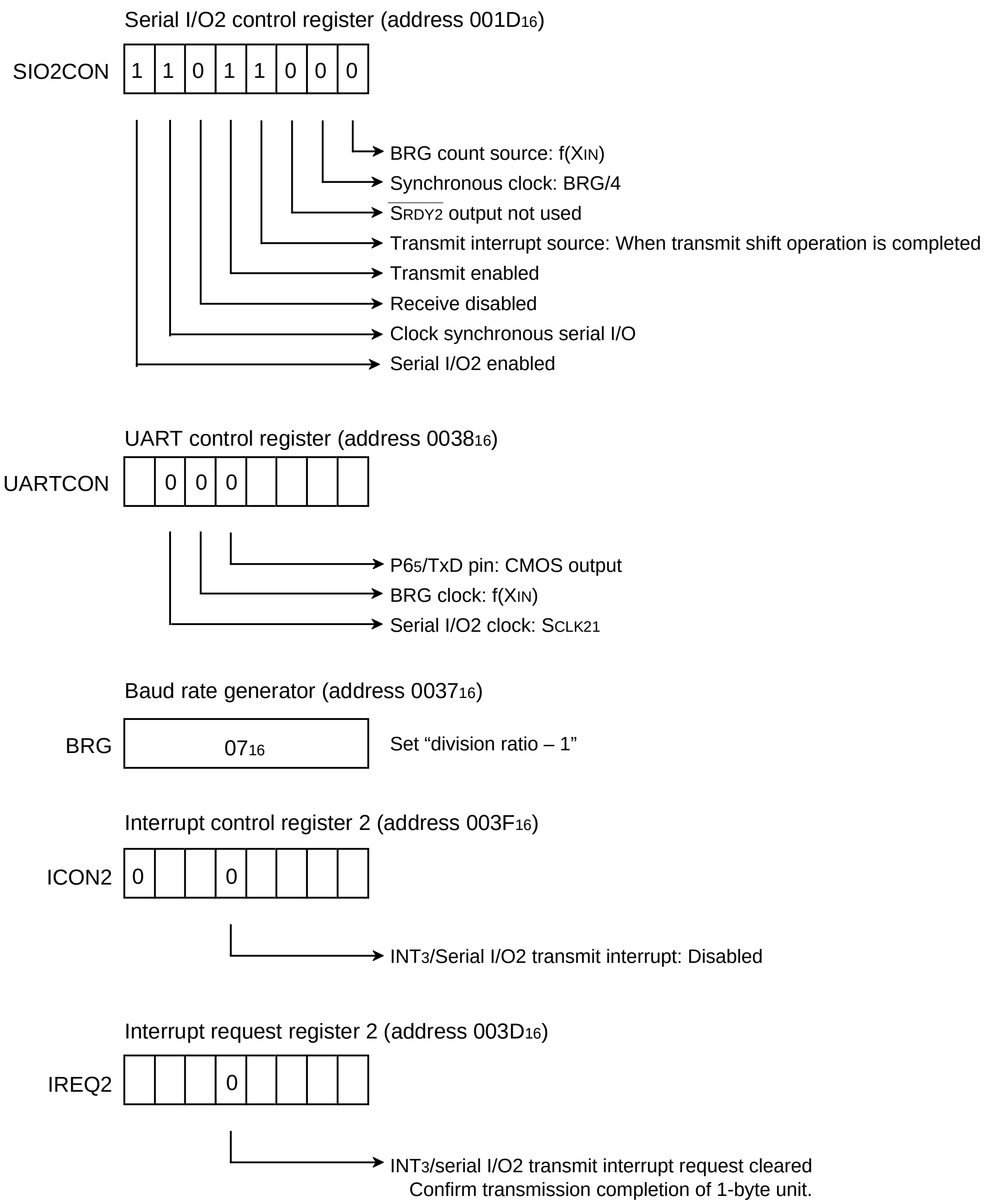


Fig. 2.3.43 Relevant registers setting

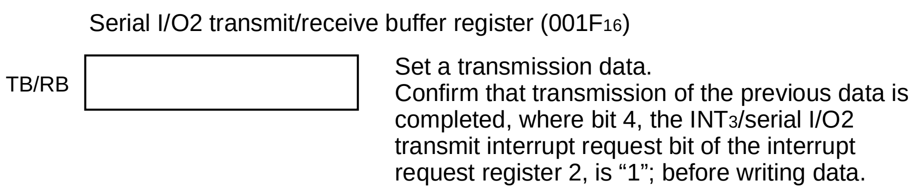


Fig. 2.3.44 Setting of transmission data

Figure 2.3.45 shows a control procedure.

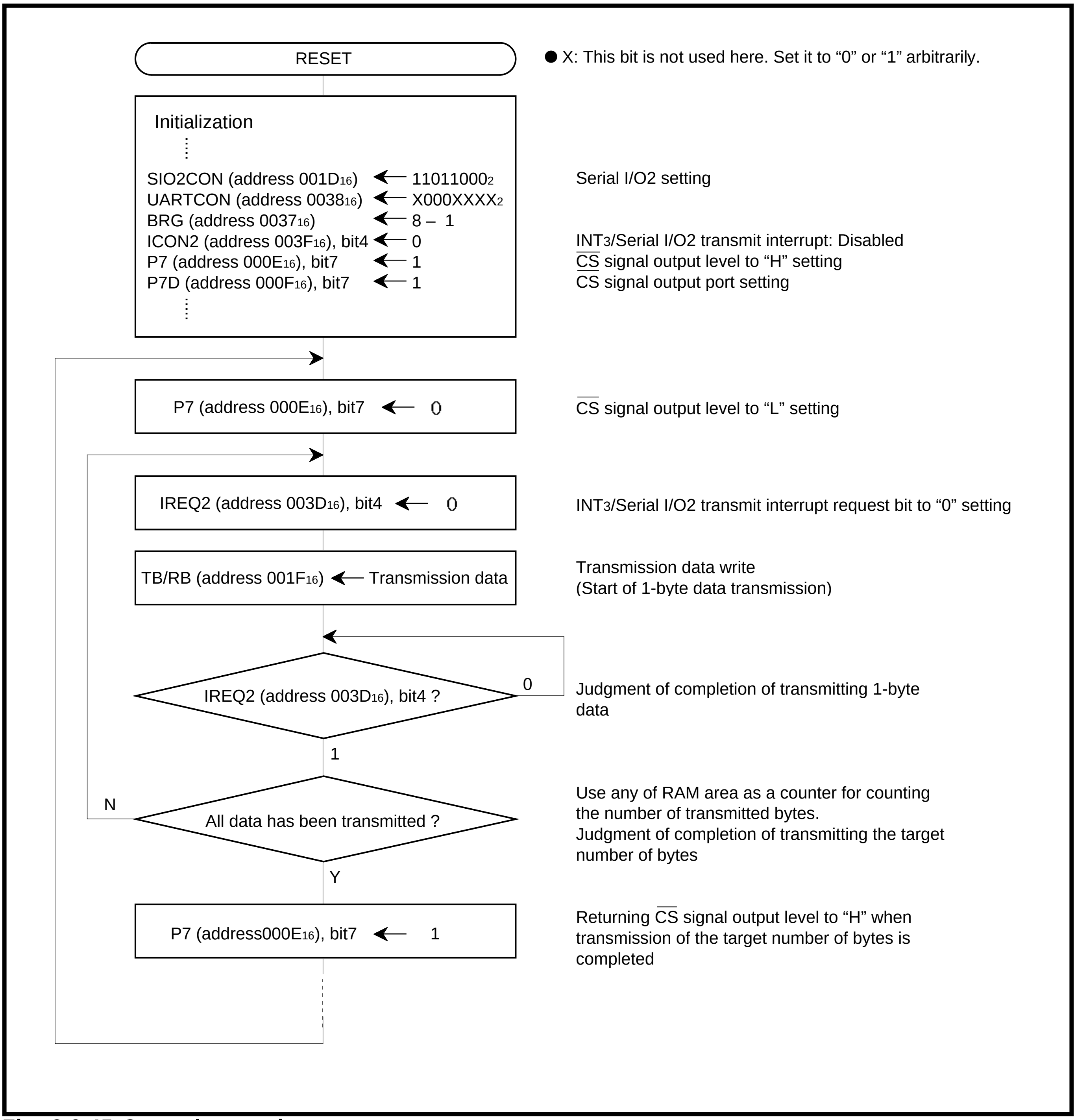


Fig. 2.3.45 Control procedure

(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 2.3.46 shows a connection diagram.

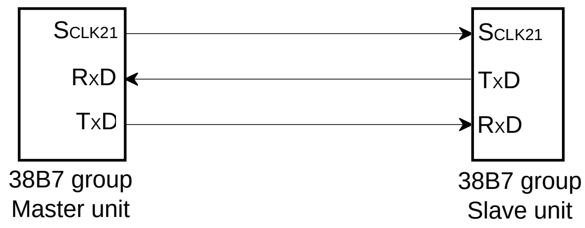


Fig. 2.3.46 Connection diagram

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.

The communication is performed according to the timing shown in Figure 2.3.47. 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 2.3.48 shows the relevant registers setting in the master unit and Figure 2.3.49 shows the relevant registers setting in the slave unit.

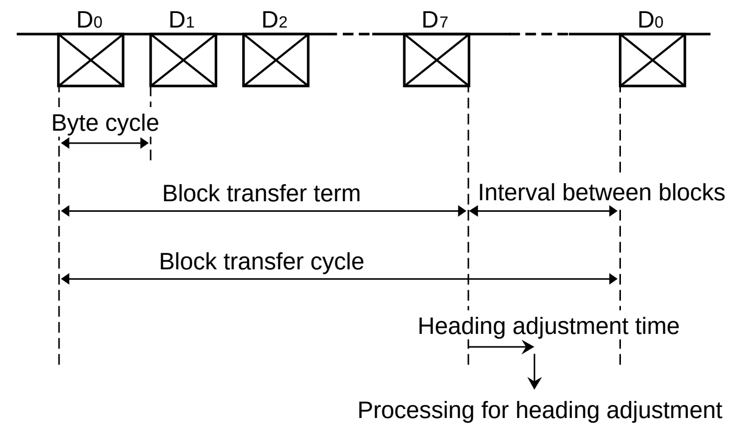


Fig. 2.3.47 Timing chart

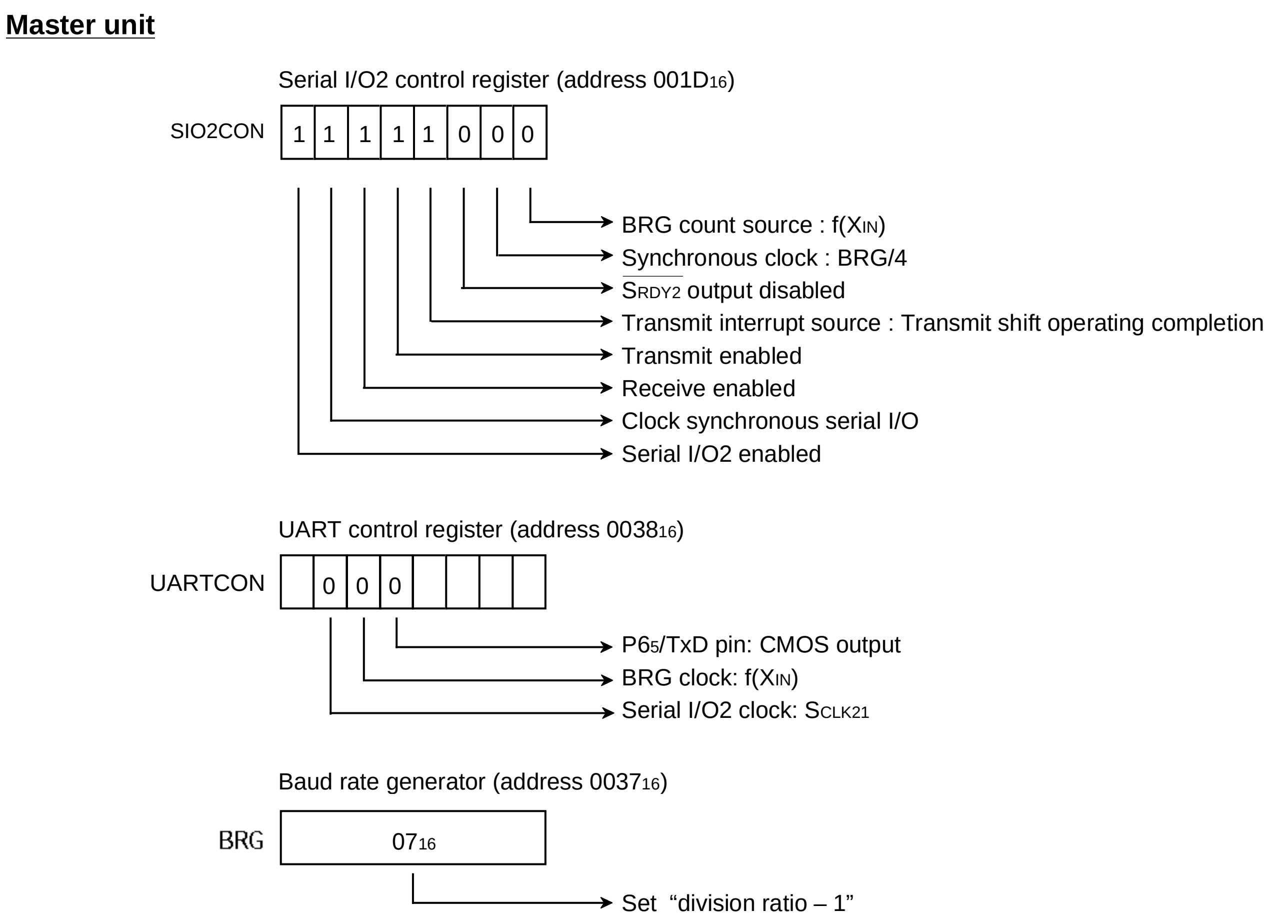


Fig. 2.3.48 Relevant registers setting in master unit

Slave unit

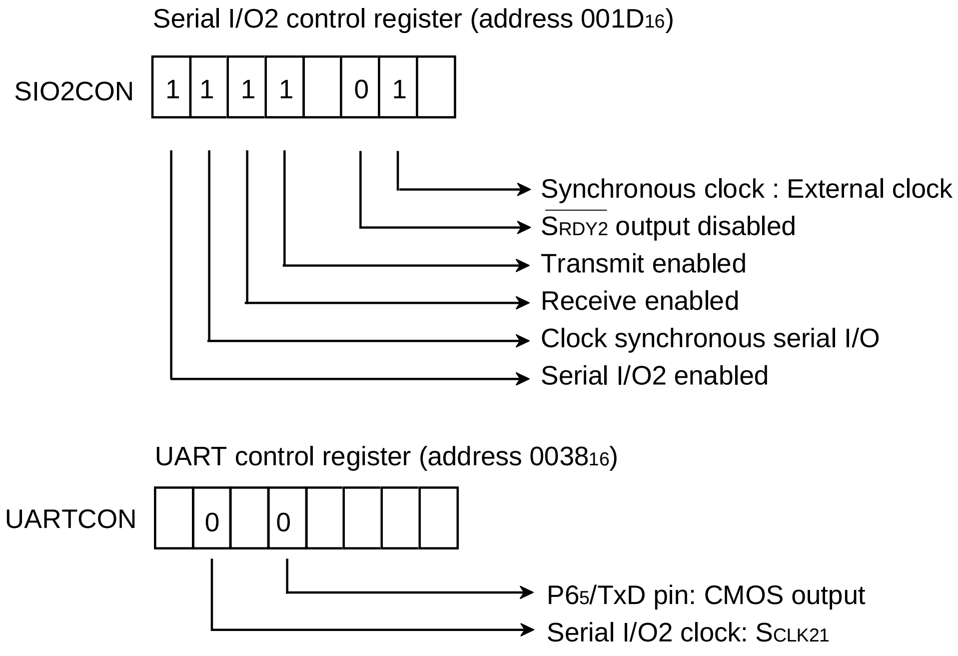


Fig. 2.3.49 Relevant registers setting in slave unit

Control procedure by software:

● Control in the master unit

After setting the relevant registers shown in Figure 2.3.48, 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 2.3.47, 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 2.3.50 shows a control procedure of the master unit using timer interrupts.

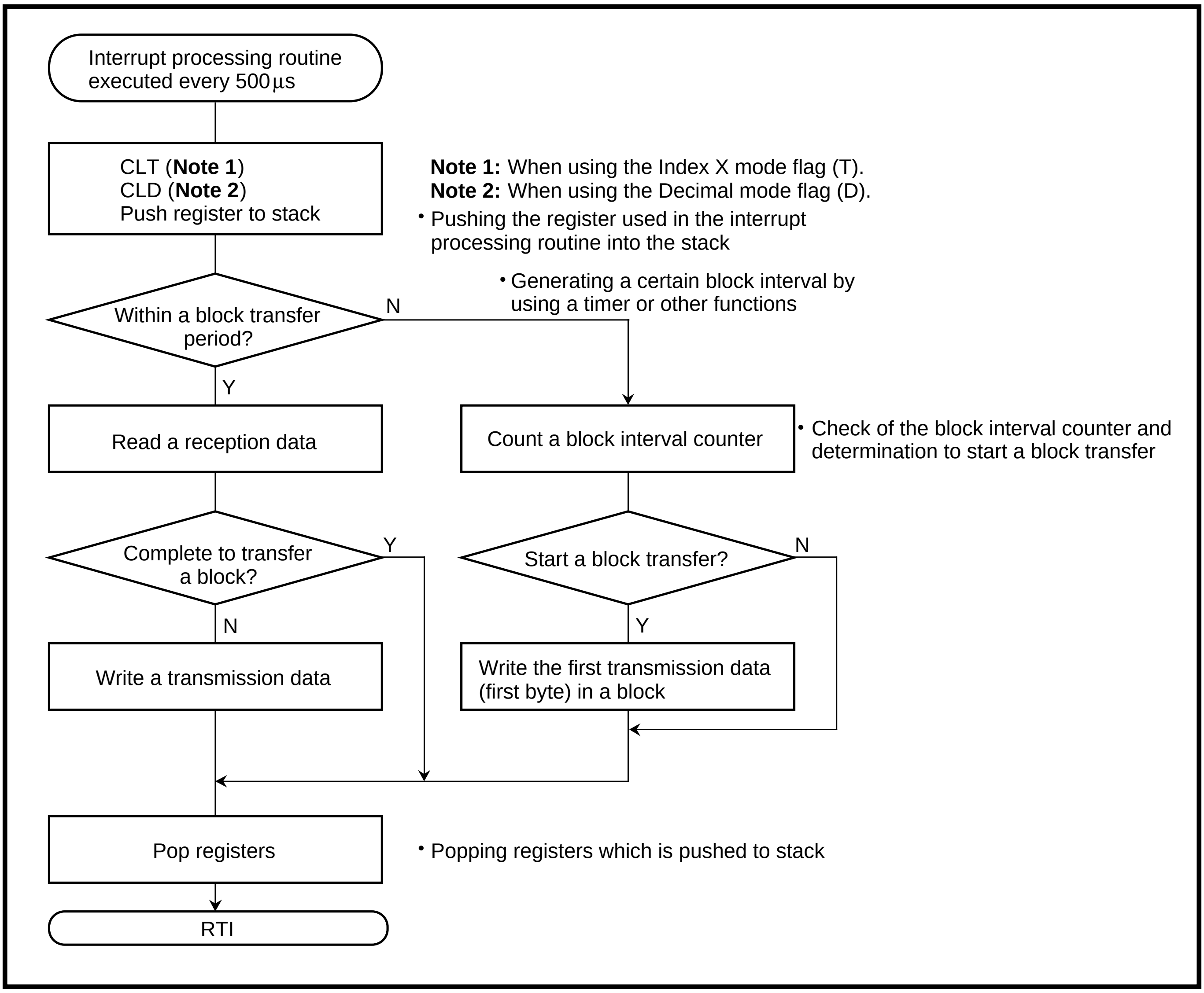


Fig. 2.3.50 Control procedure of master unit

● Control in the slave unit

After setting the relevant registers as shown in Figure 2.3.49, 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 2.3.51 shows a control procedure of the slave unit using the serial I/O2 receive interrupt and any timer interrupt (for heading adjustment).

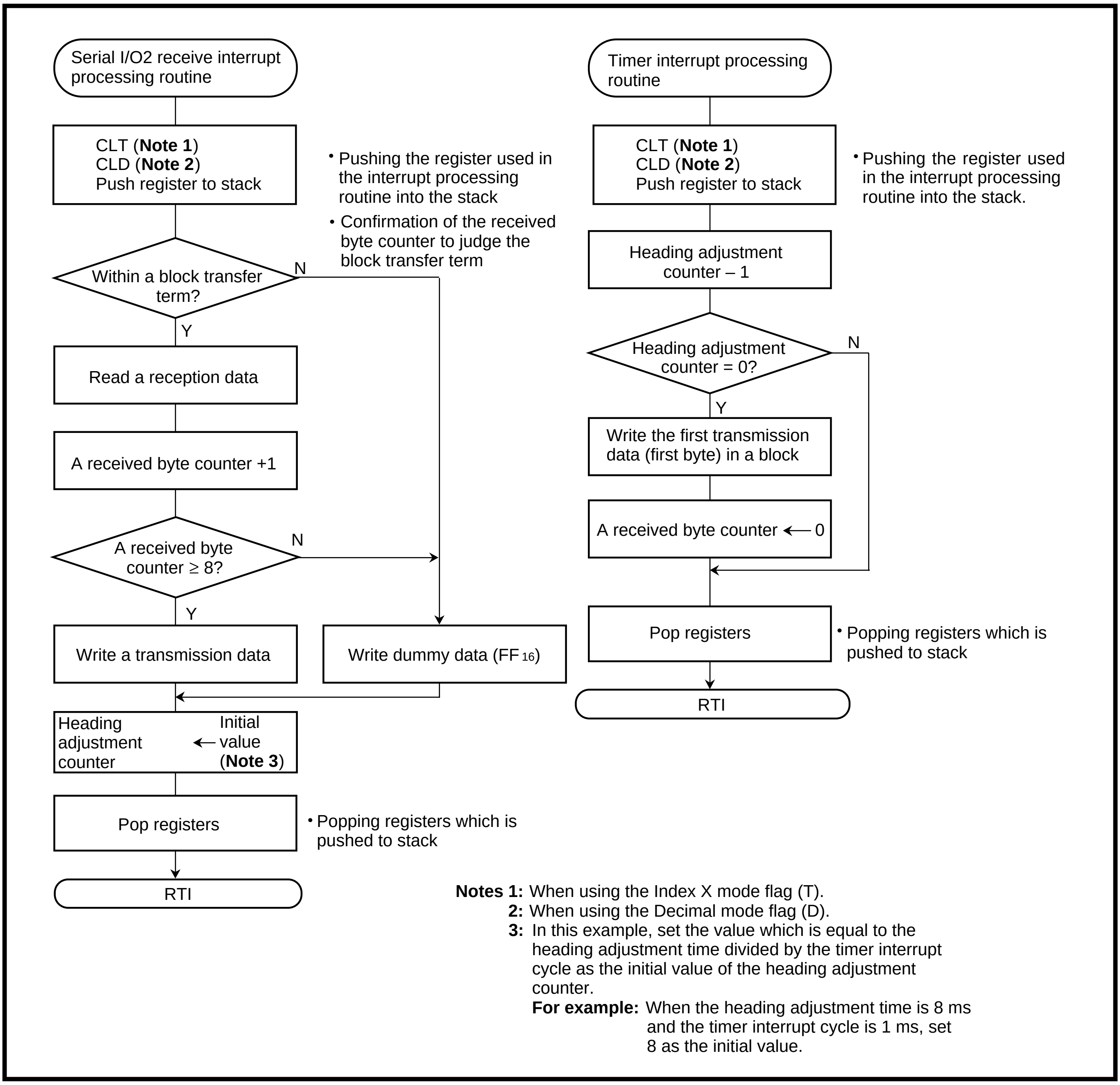


Fig. 2.3.51 Control procedure of slave unit

(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 $P7_6$ is used for communication control.

Figure 2.3.52 shows a connection diagram, and Figure 2.3.53 shows a timing chart.

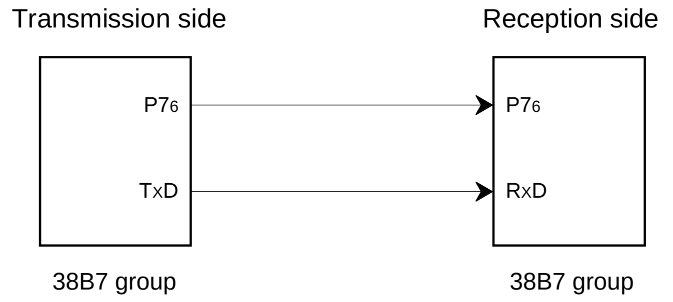


Fig. 2.3.52 Connection diagram

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 $P7_6$
  (The output level of port $P7_6$ 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.

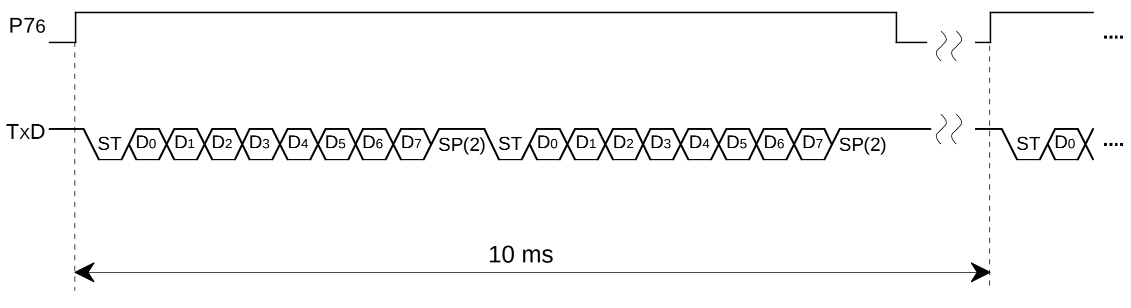


Fig. 2.3.53 Timing chart

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(XIN) = 3.6864 MHz			f(XIN) = 4 MHz		
	BRG count source (**Note 2**)	BRG setting value	Actual rate	BRG count source (**Note 2**)	BRG setting value	Actual rate
600	f(XIN)/4	95(5F16)	600.00	f(XIN)/4	103(6716)	600.96
1200	f(XIN)/4	47(2F16)	1200.00	f(XIN)/4	51(3316)	1201.92
2400	f(XIN)/4	23(1716)	2400.00	f(XIN)/4	25(1916)	2403.85
4800	f(XIN)/4	11(0B16)	4800.00	f(XIN)/4	12(0C16)	4807.69
9600	f(XIN)/4	5(0516)	9600.00	f(XIN)	25(1916)	9615.38
19200	f(XIN)/4	2(0216)	19200.00	f(XIN)	12(0C16)	19230.77
38400	f(XIN)	5(0516)	38400.00	f(XIN)	5(0516)	41666.67
76800	f(XIN)	2(0216)	76800.00	f(XIN)	2(0216)	83333.33
31250	—	—	—	f(XIN)	7(0716)	31250.00
62500	—	—	—	f(XIN)	3(0316)	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 001D16) 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 001D16).

Figure 2.3.54 shows the registers setting relevant to the transmission side; Figure 2.3.55 shows the registers setting relevant to the reception side.

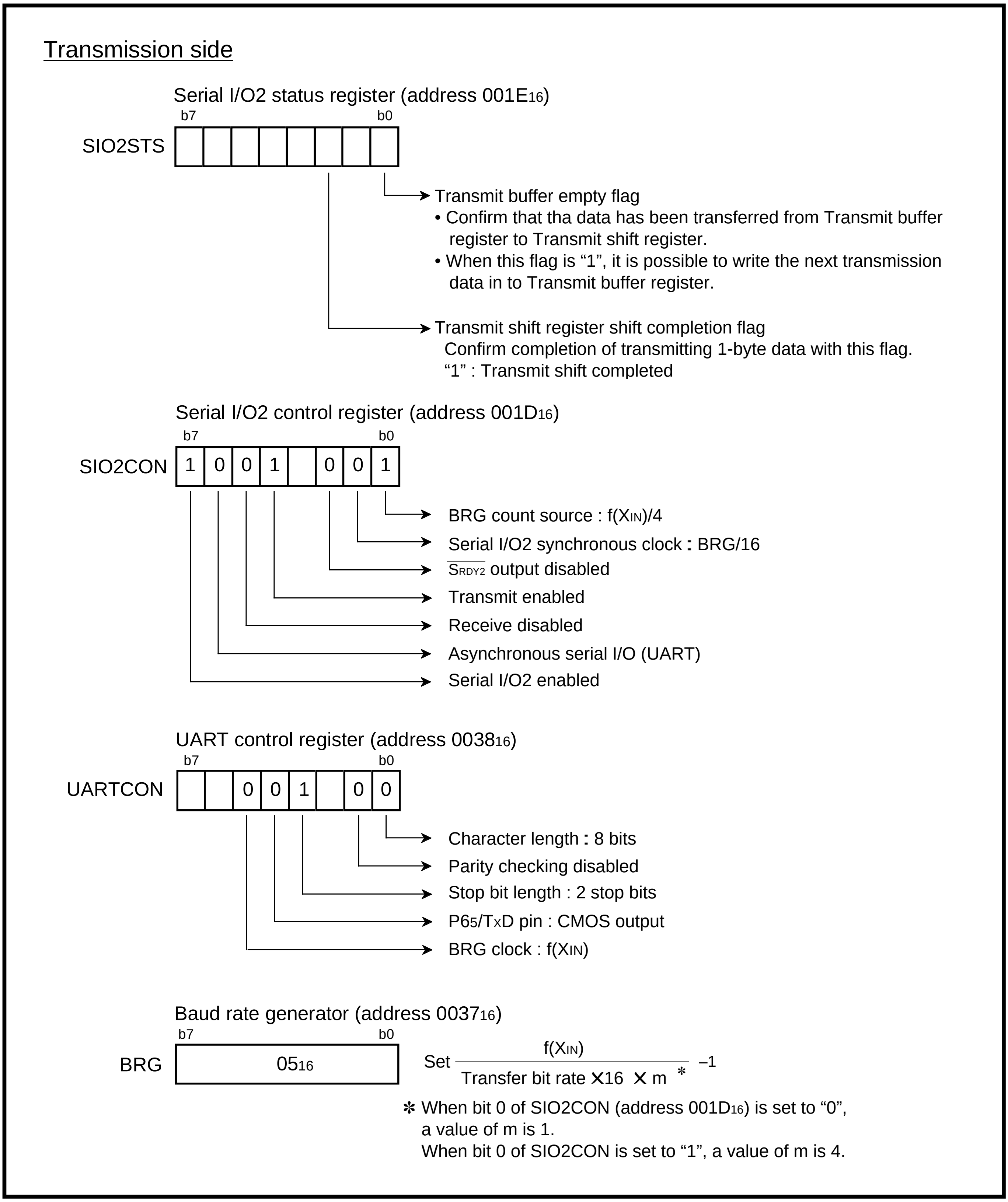


Fig. 2.3.54 Registers setting relevant to transmission side

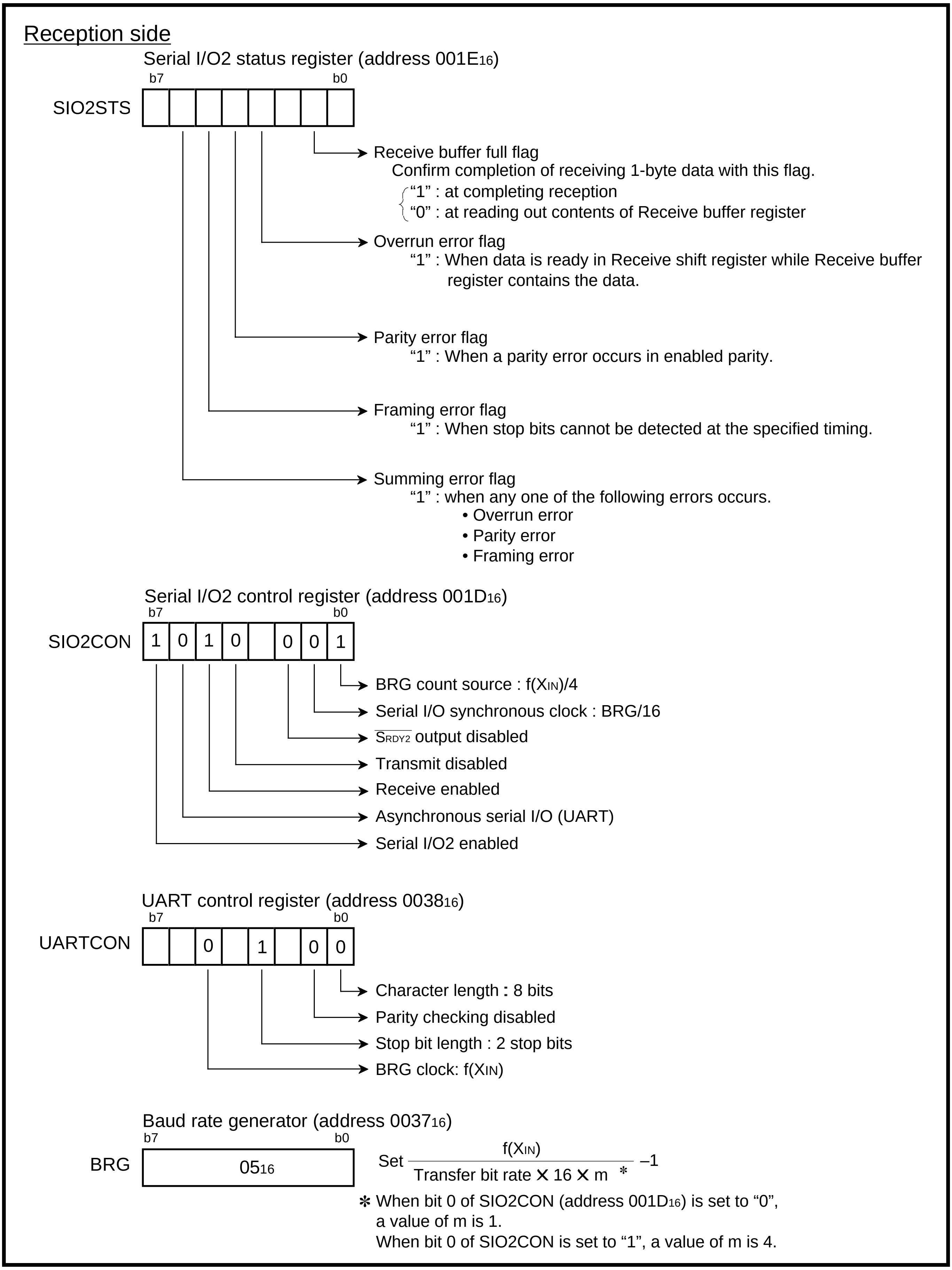


Fig. 2.3.55 Registers setting relevant to reception side

Figure 2.3.56 shows a control procedure of the transmission side, and Figure 2.3.57 shows a control procedure of the reception side.

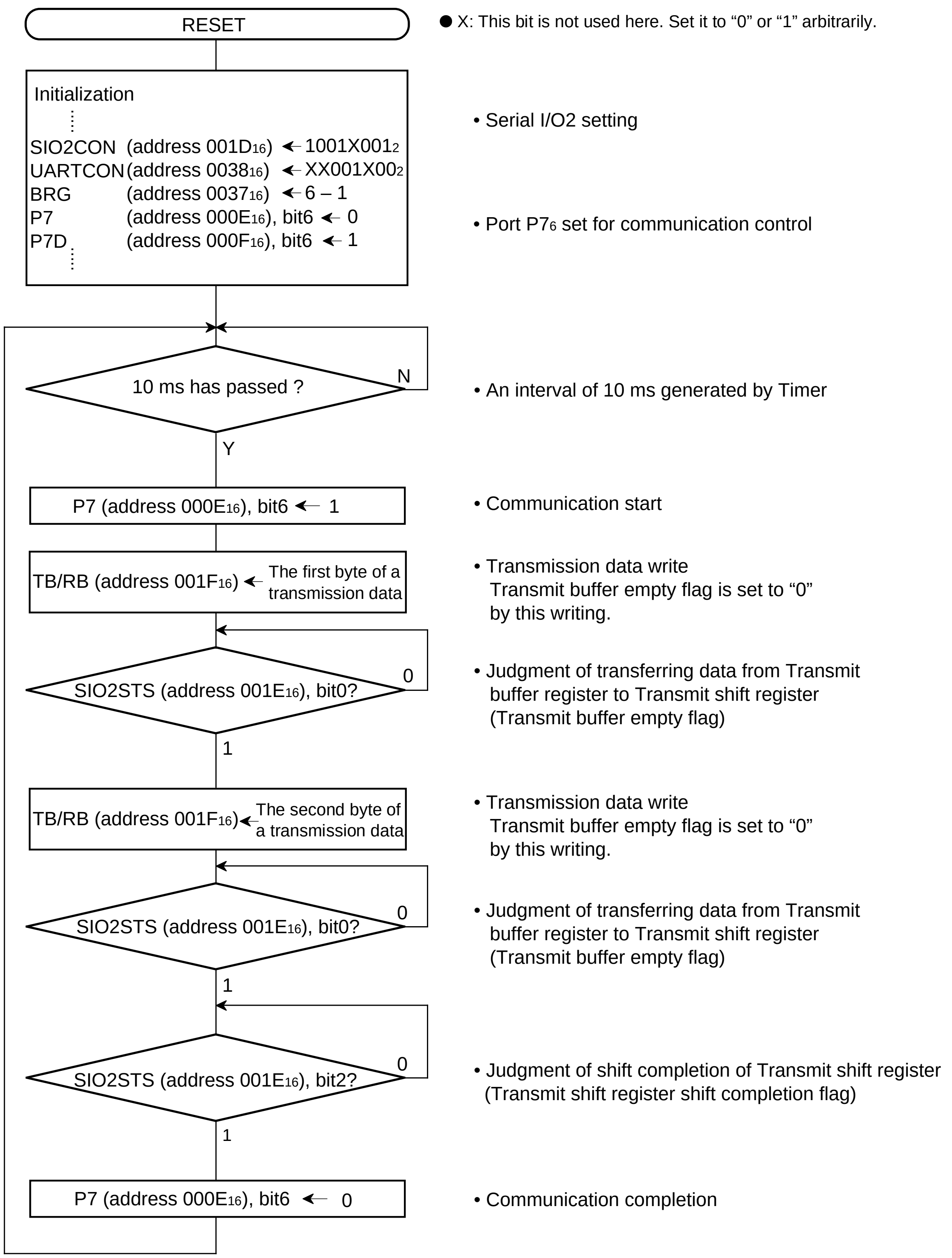


Fig. 2.3.56 Control procedure of transmission side

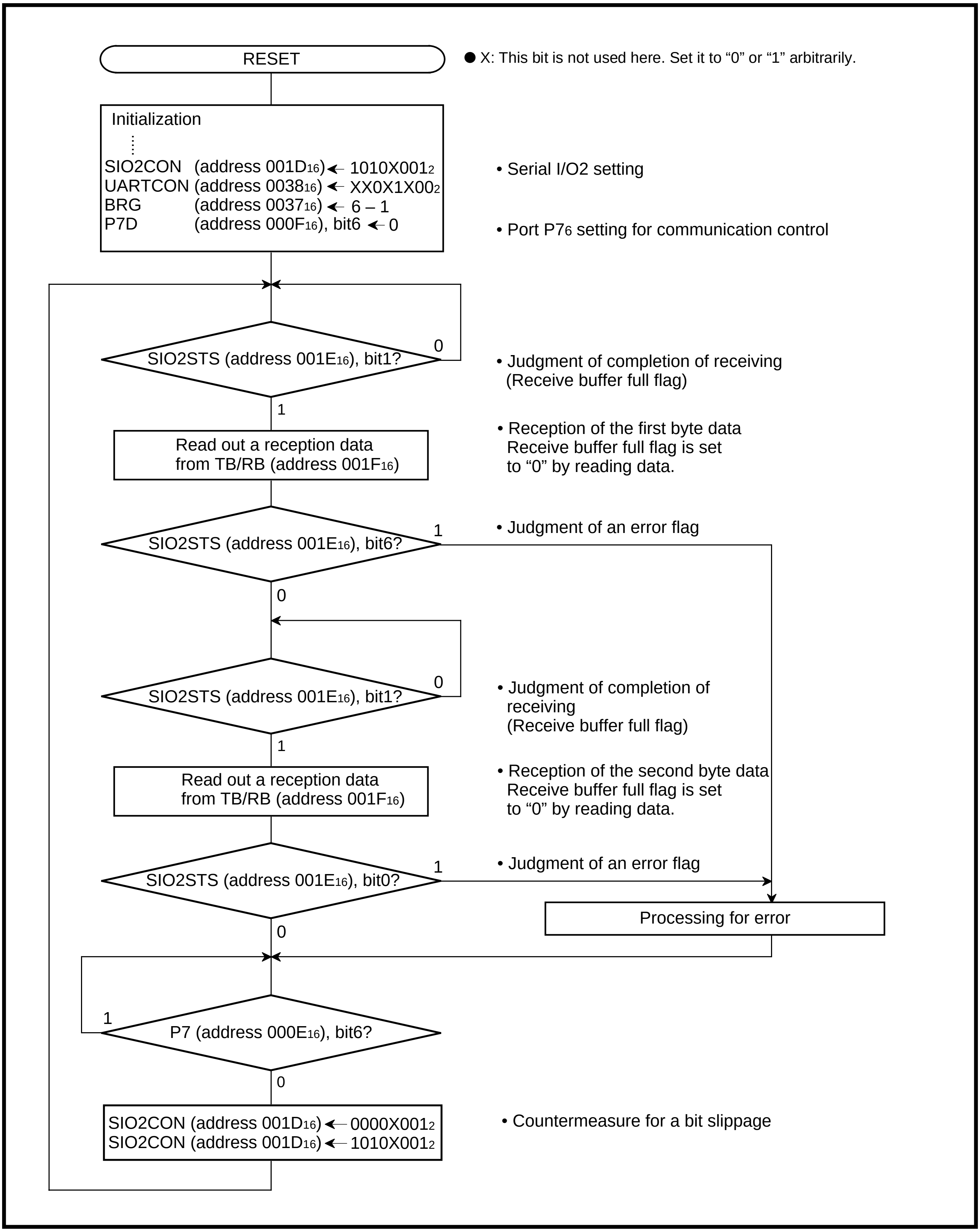


Fig. 2.3.57 Control procedure of reception side

2.3.9 Serial I/O3 connection examples

(1) Control of peripheral IC equipped with CS pin

Figure 2.3.58 shows connection examples with peripheral ICs equipped with the CS pin.

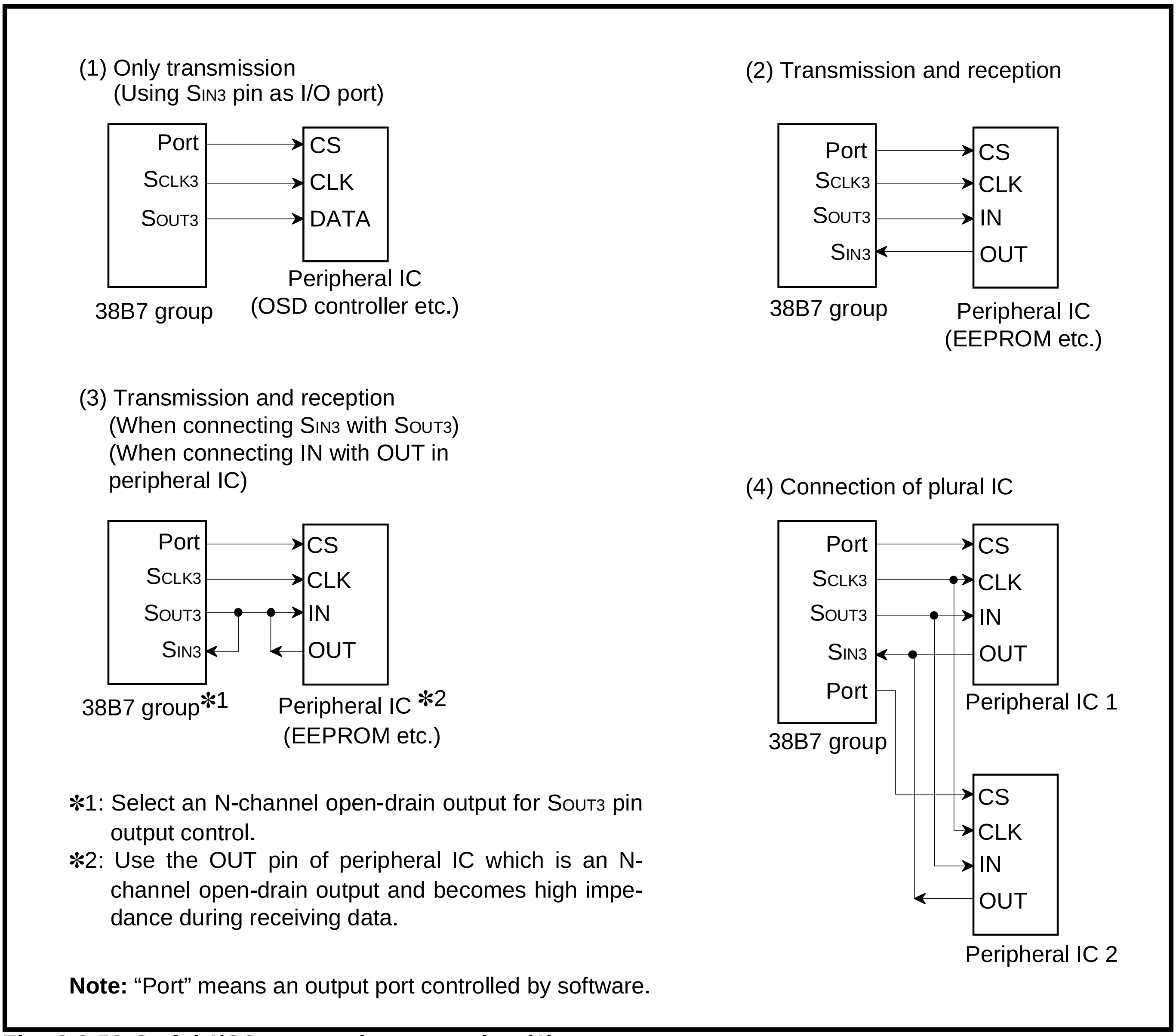


Fig. 2.3.58 Serial I/O3 connection examples (1)

(2) Connection with microcomputer

Figure 2.3.59 shows connection examples with another microcomputer.

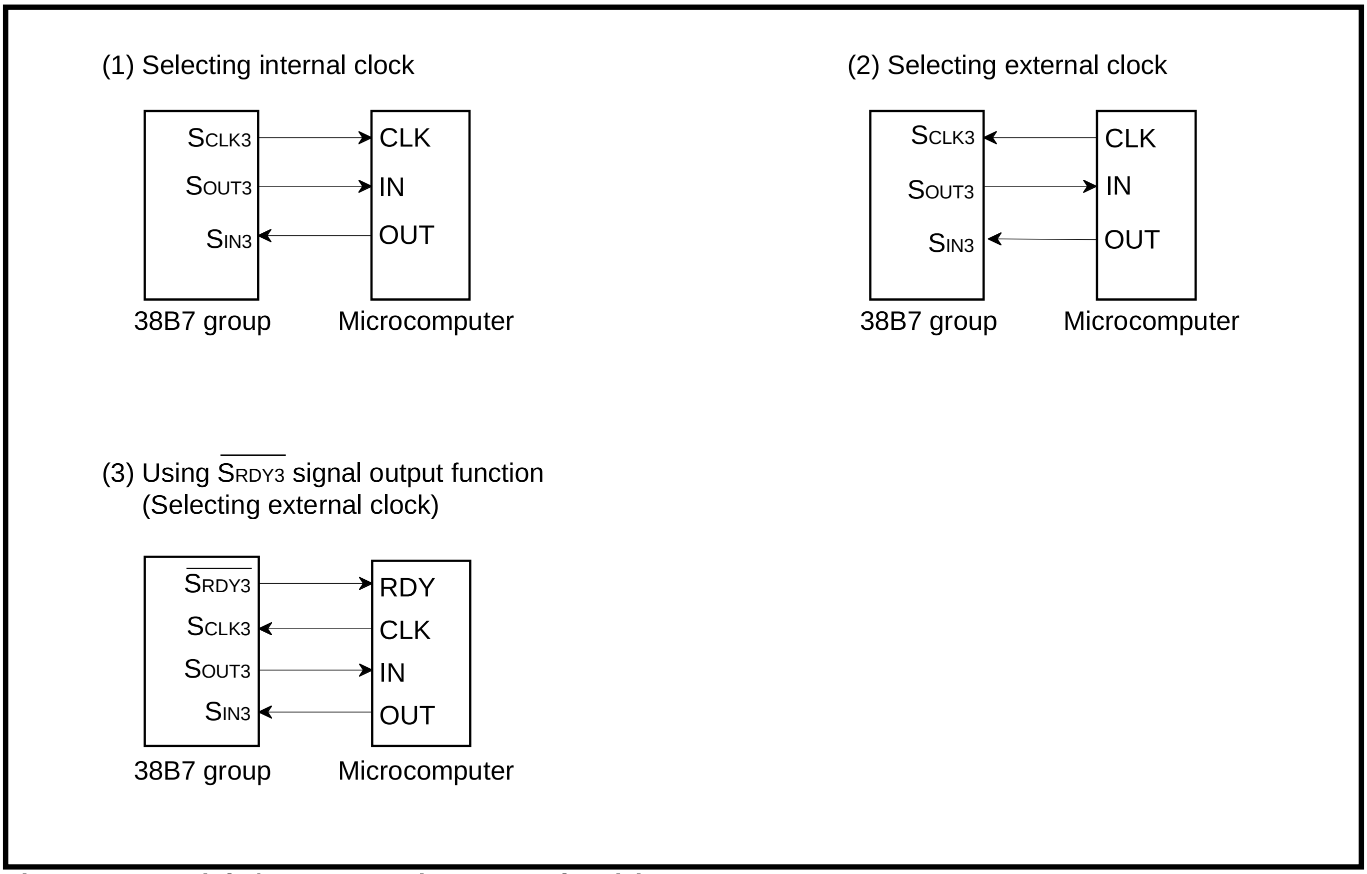


Fig. 2.3.59 Serial I/O3 connection examples (2)

2.3.10 Serial I/O3's modes

Figure 2.3.60 shows the serial I/O3's modes.

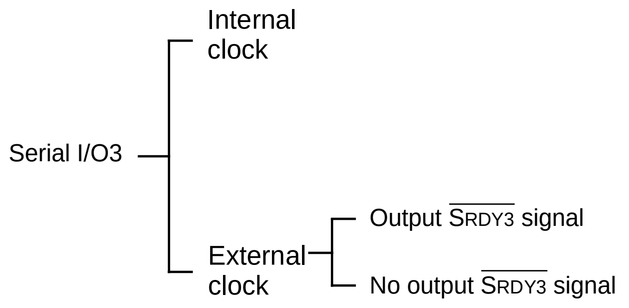


Fig. 2.3.60 Serial I/O3's modes

2.3.11 Serial I/O3 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 2.3.61 shows a connection diagram, and Figure 2.3.62 shows a timing chart.

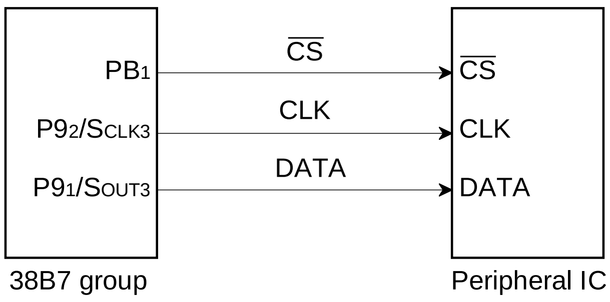


Fig. 2.3.61 Connection diagram

Specifications :
- Use of serial I/O3
- Synchronous clock frequency : 131 kHz (f(X_{IN}) = 4.19 MHz is divided by 32)
- Transfer direction : LSB first
- Not use of serial I/O3 interrupt
- Port PB_1 is connected to the $\overline{CS}$ pin (“L” active) of the peripheral IC for transmission control; the output level of port PB_1 is controlled by software.

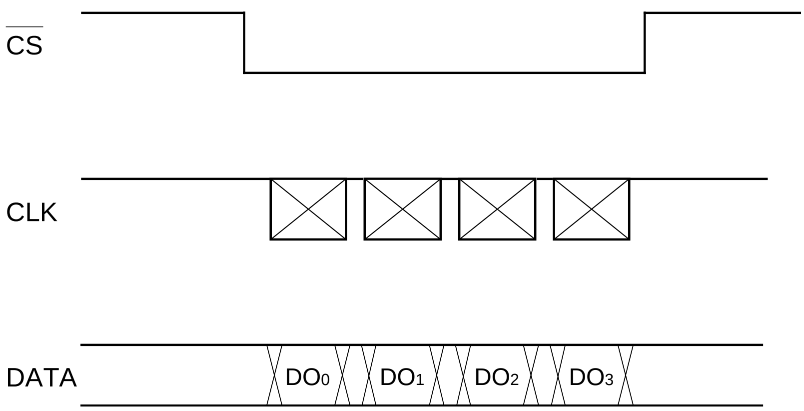


Fig. 2.3.62 Timing chart

Figure 2.3.63 shows the registers setting relevant to the transmission side, and Figure 2.3.64 shows the setting of transmission data.

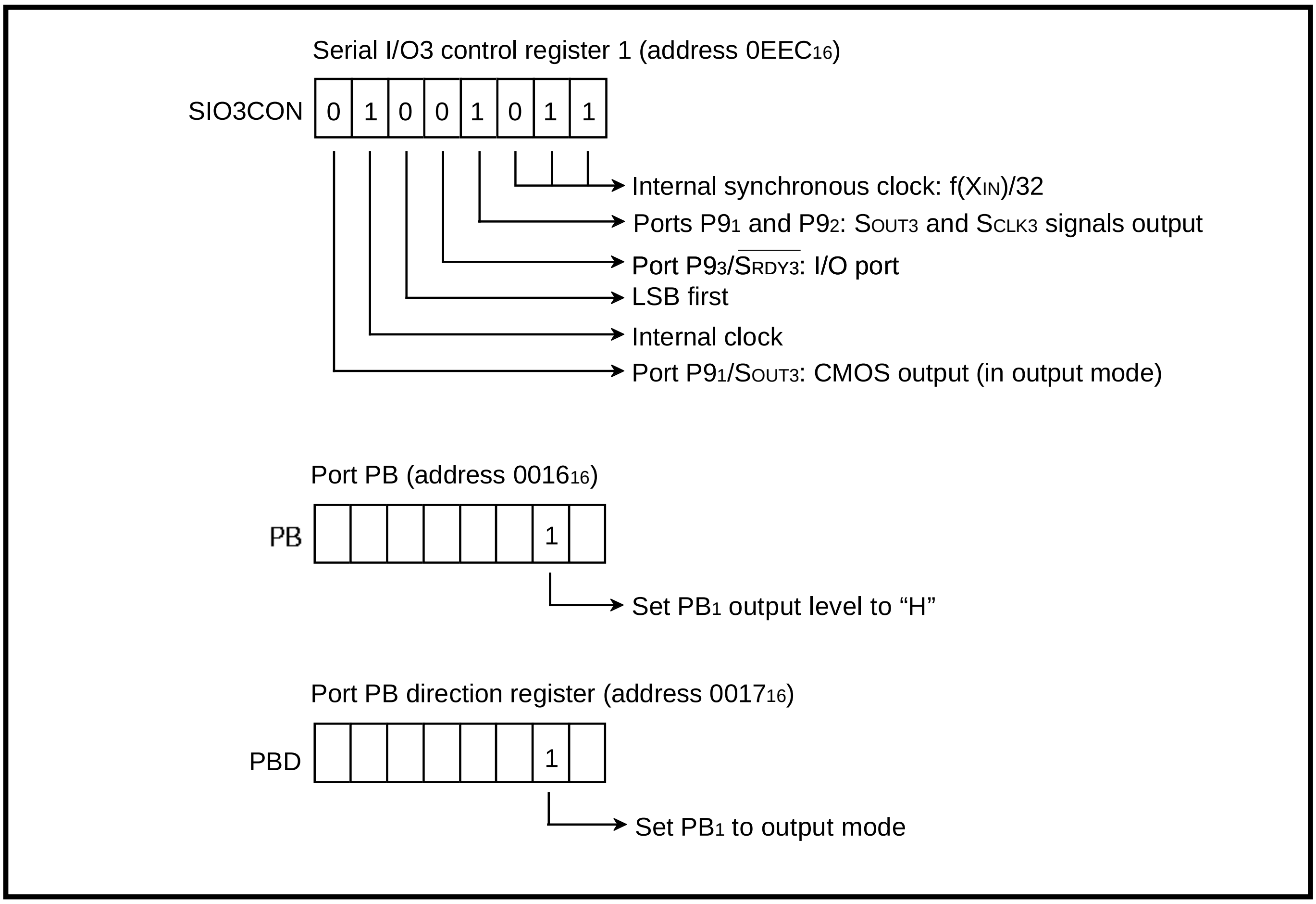


Fig. 2.3.63 Registers setting relevant to transmission side

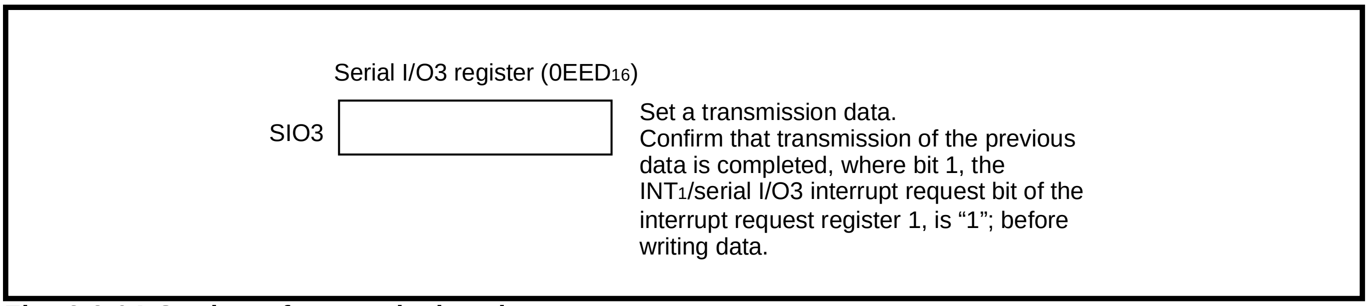


Fig. 2.3.64 Setting of transmission data

Control procedure: When the registers are set as shown in Figure 2.3.65, the serial I/O3 can transmit 1-byte data by writing data to the serial I/O3 register.
Thus, after setting the $\overline{CS}$ signal to "L", write the transmission data to the serial I/O3 register by each 1 byte; and return the $\overline{CS}$ signal to "H" when the target number of bytes has been transmitted.
Figure 2.3.65 shows a control procedure.

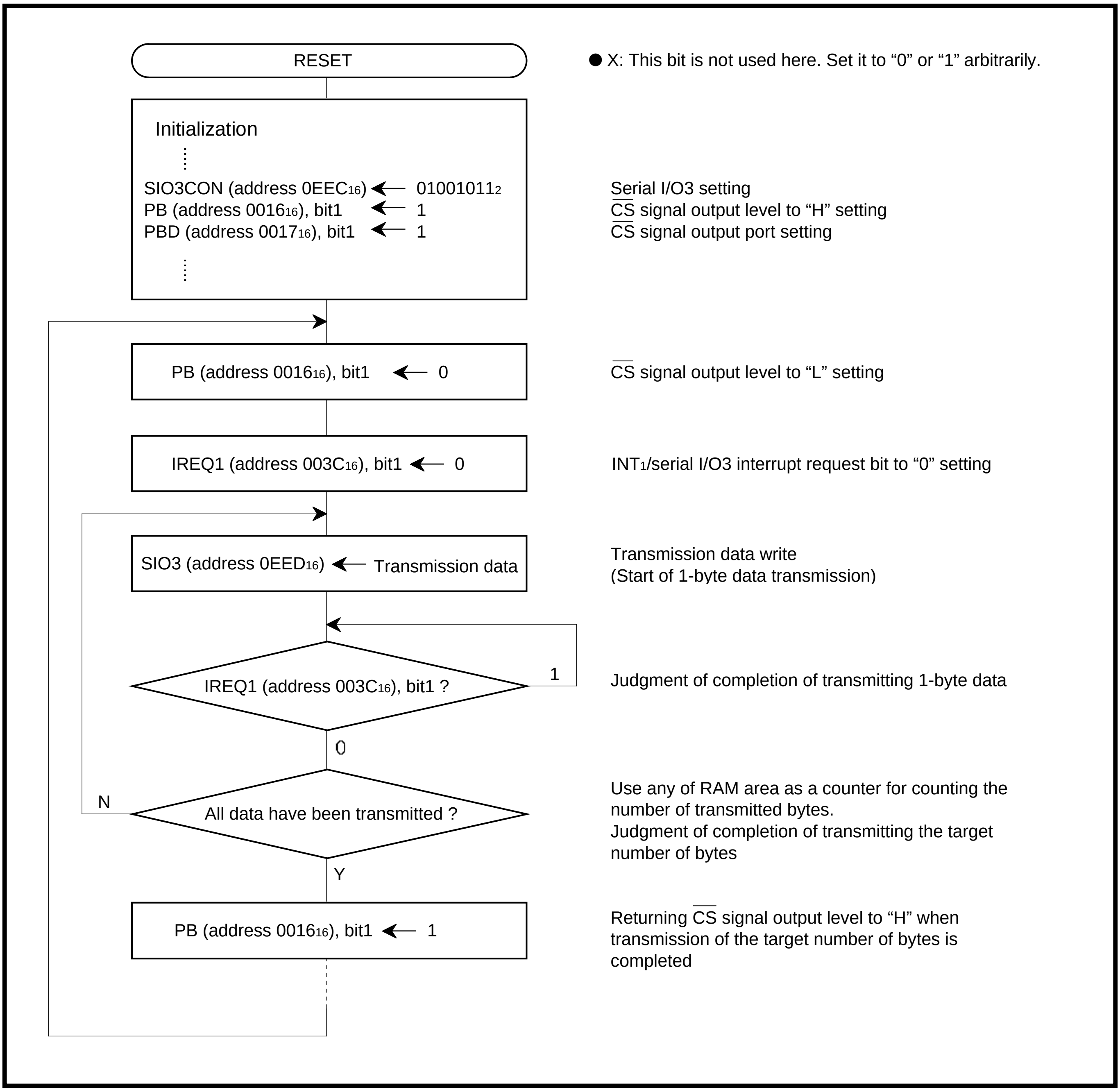


Fig. 2.3.65 Control procedure

2.3.12 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 before enabling interrupts.

(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.
- When selecting an external clock, the set of automatic transfer interval becomes invalid.
- Set the transfer interval of each 1-byte data transfer as the following:

(1) Not using FLD controller

Keep the interval for 5 cycles or more of internal system clock from clock rising of the last bit of 1-byte data.

(2) Using FLD controller

(a) Not using gradation display

Keep the interval for 17 cycles or more of internal system clock from clock rising of the last bit of 1-byte data.

(b) Using gradation display

Keep the interval for 27 cycles or more of internal system clock from clock rising of the last bit of 1-byte data.

<Serial I/O1 control register 3, SIO1CON3 (address $001C_{16}$) setting example>

Table 2.3.2 SIO1CON3 (address $001C_{16}$) setting example selecting internal synchronous clock

Serial I/O1 control register 3, SIO1CON3 (address $001C_{16}$)		Not using FLDC	Not using gradation display mode	Using gradation display mode
Internal synchronous clock selection bits (b7 to b5)	Automatic transfer interval set bits (b4 to b0)			
0 0 0 : f(XIN) / 4	0 0 0 0 0 : 2 cycles of transfer clocks	Usable	Prohibited	Prohibited
	0 0 0 0 1 : 3 cycles of transfer clocks	Usable	Prohibited	Prohibited
	0 0 0 1 0 : 4 cycles of transfer clocks	Usable	Prohibited	Prohibited
	0 0 0 1 1 : 5 cycles of transfer clocks	Usable	Usable	Usable
0 0 1 : f(XIN) / 8	0 0 0 0 0 : 2 cycles of transfer clocks	Usable	Prohibited	Prohibited
	0 0 0 0 1 : 3 cycles of transfer clocks	Usable	Usable	Usable
0 1 0 : f(XIN) / 16	0 0 0 0 0 : 2 cycles of transfer clocks	Usable	Usable	Usable

Table 2.3.3 SIO1CON3 (address $001C_{16}$) setting example selecting external synchronous clock

Serial I/O1 control register 3, SIO1CON3 (address $001C_{16}$), Automatic transfer interval set bits	"n" cycles of transfer clocks
Not using FLDC	Transfer clock × n cycles ≥ 5 cycles of internal system clock
Not using gradation display mode	Transfer clock × n cycles ≥ 17 cycles of internal system clock
Using gradation display mode	Transfer clock × n cycles ≥ 27 cycles of internal system clock

■ 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 system clock 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.

2.3.13 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.”

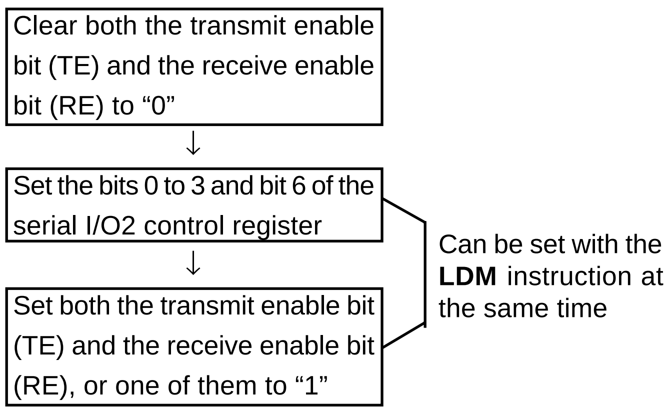


Fig. 2.3.66 Sequence of setting serial I/O2 control register again

(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) Setting procedure when serial I/O2 transmit interrupt is used

When setting the transmit enable bit to "1", the serial I/O2 transmit interrupt request bit is automatically set to "1". When not requiring the interrupt occurrence synchronized with the transmission enabled, take the following sequence.

①Set the serial I/O1 transmit interrupt enable bit to "0" (disabled).
②Set the transmit enable bit to "1".
③Set the serial I/O1 transmit interrupt request bit to "0" after 1 or more instructions have been executed.
④Set the serial I/O1 tranmit interrupt enable bit to "1" (enabled).

(8) Using TxD pin

The $P6_5$/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 $P6_5$/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

Address	
$003D_{16}$	Interrupt request register 2 (IREQ2)
$003E_{16}$	(Interrupt control register 1 (ICON1))
$003F_{16}$	Interrupt control register 2 (ICON2)
≈	≈
$0EF2_{16}$	Port P0 digit output set switch register (P0DOR)
$0EF3_{16}$	Port P2 digit output set switch register (P2DOR)
$0EF4_{16}$	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 P4FLD/port switch register (P4FPR)
$0EFA_{16}$	Port P5FLD/port switch register (P5FPR)
$0EFB_{16}$	Port P6FLD/port switch register (P6FPR)
$0EFC_{16}$	FLD output control register (FLDCON)

Fig. 2.4.1 Memory assignment of FLD controller relevant registers

2.4.2 Relevant registers

Port P0 digit output set switch register

b7 b6 b5 b4 b3 b2 b1 b0

Port P0 digit output set switch register
(P0DOR: address $0EF2_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P0_0$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
1	Port $P0_1$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
2	Port $P0_2$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
3	Port $P0_3$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
4	Port $P0_4$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
5	Port $P0_5$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
6	Port $P0_6$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
7	Port $P0_7$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○

Fig. 2.4.2 Structure of Port P0 digit output set switch register

Port P2 digit output set switch register

b7 b6 b5 b4 b3 b2 b1 b0

Port P2 digit output set switch register
(P2DOR: address $0EF3_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P2_0$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
1	Port $P2_1$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
2	Port $P2_2$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
3	Port $P2_3$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
4	Port $P2_4$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
5	Port $P2_5$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
6	Port $P2_6$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○
7	Port $P2_7$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	○	○

Fig. 2.4.3 Structure of Port P2 digit output set switch register

FLDC mode register

b7 b6 b5 b4 b3 b2 b1 b0

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

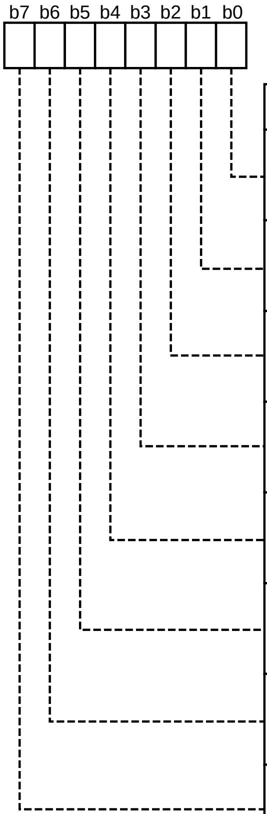
b	Name	Functions	At reset	R	W
0	Automatic display control bit	0 : General-purpose mode 1 : Automatic display mode	0	O	O
1	Display start bit	0 : Display stopped 1 : Display in progress (display starts by writing “1”)	0	O	O
2	Tscan control bits	b3 b2 0 0 : FLD digit interrupt (at rising edge of each digit)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

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 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	O	O
1		0	O	O
2		0	O	O
3		0	O	O
4		0	O	O
5		0	O	O
6		0	O	O
7		0	O	O

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(XIN) = 4 MHz •bit 6 of FLDC mode register = 0 (f(XIN)/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(XIN) = 4 MHz •bit 6 of FLDC mode register = 0 (f(XIN)/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 SET/RESET switch bit (b7) of the FLD output control register (address $0EFC_{16}$) is set to “1”, set value of 03_{16} or more to the Toff2 time set register.

Fig. 2.4.7 Structure of Toff2 time set register

FLD data pointer/FLD data pointer reload register

b7 b6 b5 b4 b3 b2 b1 b0

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 P6, P5, P4, P3, P1, P0, and P2, 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	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. 2.4.8 Structure of FLD data pointer/FLD data pointer reload register

Port P4FLD/port switch register

b7 b6 b5 b4 b3 b2 b1 b0

Port P4FLD/port switch register
(P4FPR: address $0EF9_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P4_0$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
1	Port $P4_1$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
2	Port $P4_2$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
3	Port $P4_3$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
4	Port $P4_4$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
5	Port $P4_5$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
6	Port $P4_6$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
7	Port $P4_7$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O

Fig. 2.4.9 Structure of port P4FLD/port switch register

Port P5FLD/port switch register

b7 b6 b5 b4 b3 b2 b1 b0

Port P5FLD/port switch register
(P5FPR: address $0EFA_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P5_0$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
1	Port $P5_1$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
2	Port $P5_2$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
3	Port $P5_3$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
4	Port $P5_4$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
5	Port $P5_5$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
6	Port $P5_6$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
7	Port $P5_7$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O

Fig. 2.4.10 Structure of port P5FLD/port switch register

Port P6FLD/port switch register

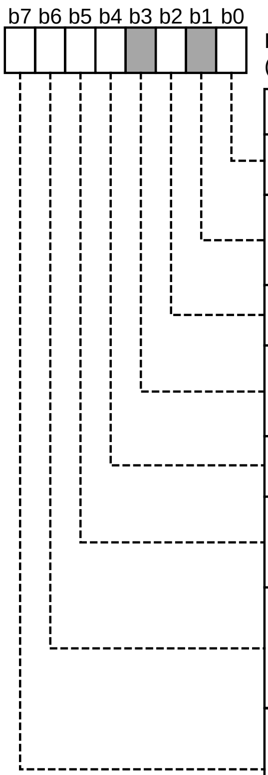
b7 b6 b5 b4 b3 b2 b1 b0

Port P6FLD/port switch register
(P6FPR: address $0EFB_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P6_0$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
1	Port $P6_1$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
2	Port $P6_2$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
3	Port $P6_3$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
4	Port $P6_4$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
5	Port $P6_5$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
6	Port $P6_6$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
7	Port $P6_7$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O

Fig. 2.4.11 Structure of port P6FLD/port switch register

FLD output control register



FLD output control register
(FLDCON : address $0EFC_{16}$)

b	Name	Functions	At reset	R	W
0	$P6_4$–$P6_7$ FLD output reverse bit	0 : Output normally 1 : Reverse output	0	○	○
1	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×
2	$P6_4$–$P6_7$ Toff invalid bit	0 : Operating normally 1 : Toff invalid	0	○	○
3	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×
4	$P7_3$ dimmer output control bit	0 : Ordinary port 1 : Dimmer output	0	○	○
5	Generating/Not of CMOS port Toff section selection bit	0 : Toff section not generated 1 : Toff section generated	0	○	○
6	Generating/Not of high-breakdown voltage port Toff section selection bit	0 : Toff section not generated 1 : Toff section generated	0	○	○
7	Toff2 SET/RESET switch bit	0 : Toff2 RESET; Toff1 SET 1 : Toff2 SET; Tdisp RESET	0	○	○

Fig. 2.4.12 Structure of FLD output control register

Interrupt request register 2

b7 b6 b5 b4 b3 b2 b1 b0

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

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

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

Fig. 2.4.13 Structure of Interrupt request register 2

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	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
1	Timer 5 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
2	Timer 6 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
3	Serial I/O2 receive interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
4	INT_3/Serial I/O2 transmit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
5	INT_4 interrupt enable bit A-D converter interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
6	FLD blanking interrupt enable bit FLD digit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
7	Fix "0" to this bit.		0	O	O

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.

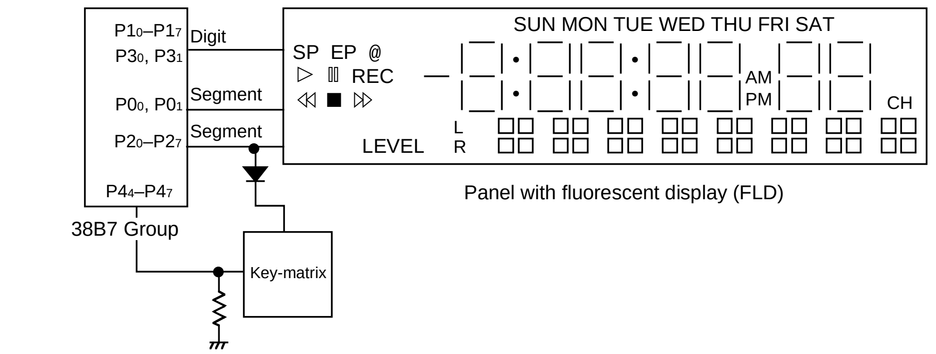


Fig. 2.4.15 Connection diagram

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 = 612 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.

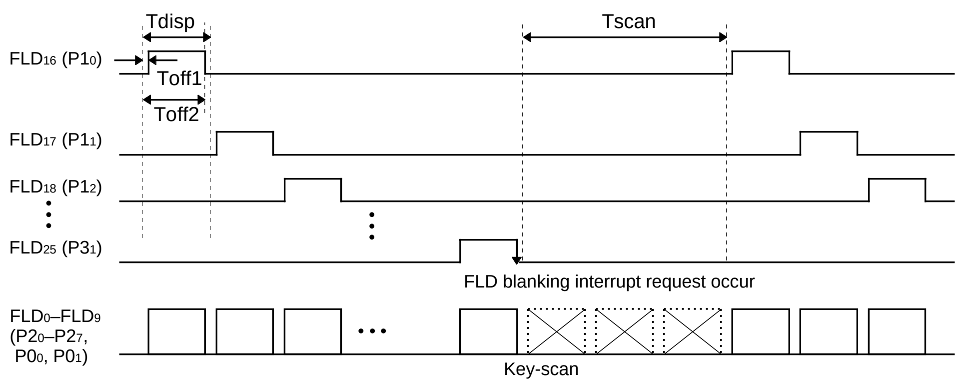


Fig. 2.4.16 Timing chart of key-scan using FLD automatic display mode and segments

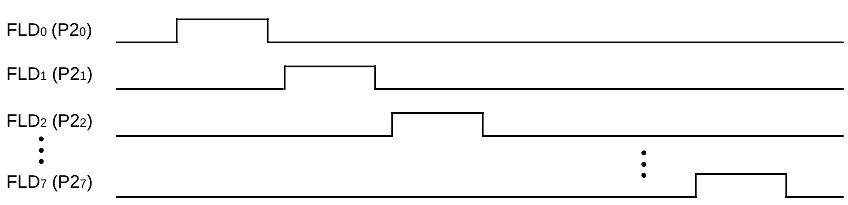


Fig. 2.4.17 Enlarged view of FLD_0 ($P2_0$) to FLD_7 ($P2_7$) Tscan

Figure 2.4.18 shows the setting of relevant registers.

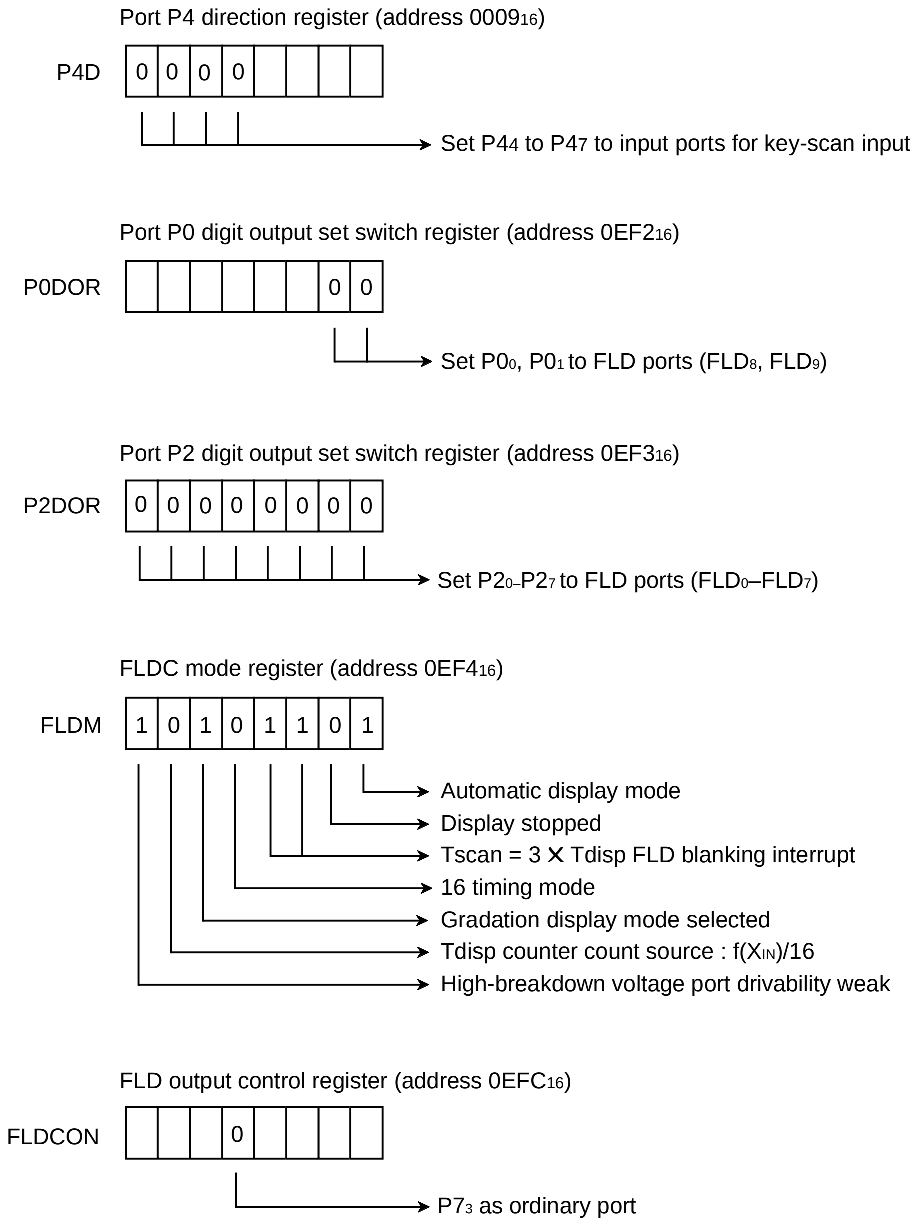


Fig. 2.4.18 Setting of relevant registers

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: $0A_{16}$ — 10 ($0A_{16}$) 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

All bits → Set {(digit number) – 1} = 9

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

IREQ2		0						

Bit 6 → Clear FLD blanking interrupt request bit

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

ICON2	0	1						

Bit 6 → FLD blanking interrupt: Enabled

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

FLDM	1	0	1	0	1	1	1	1

Bit 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
0EA0$_{16}$							FLD$_{25}$	FLD$_{24}$
0EA1$_{16}$							FLD$_{25}$	FLD$_{24}$
0EA2$_{16}$							FLD$_{25}$	FLD$_{24}$
0EA3$_{16}$							FLD$_{25}$	FLD$_{24}$
0EA4$_{16}$							FLD$_{25}$	FLD$_{24}$
0EA5$_{16}$							FLD$_{25}$	FLD$_{24}$
0EA6$_{16}$							FLD$_{25}$	FLD$_{24}$
0EA7$_{16}$							FLD$_{25}$	FLD$_{24}$
0EA8$_{16}$							FLD$_{25}$	FLD$_{24}$
0EA9$_{16}$							FLD$_{25}$	FLD$_{24}$
0EAA$_{16}$								
0EAB$_{16}$								
0EAC$_{16}$								
0EAD$_{16}$								
0EAE$_{16}$								
0EAF$_{16}$								
0EB0$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0EB1$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0EB2$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0EB3$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0EB4$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0EB5$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0EB6$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0EB7$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0EB8$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0EB9$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$
0EBA$_{16}$								
0EBB$_{16}$								
0EBC$_{16}$								
0EBD$_{16}$								
0EBE$_{16}$								
0EBF$_{16}$								
0EC0$_{16}$							FLD$_{9}$	FLD$_{8}$
0EC1$_{16}$							FLD$_{9}$	FLD$_{8}$
0EC2$_{16}$							FLD$_{9}$	FLD$_{8}$
0EC3$_{16}$							FLD$_{9}$	FLD$_{8}$
0EC4$_{16}$							FLD$_{9}$	FLD$_{8}$
0EC5$_{16}$							FLD$_{9}$	FLD$_{8}$
0EC6$_{16}$							FLD$_{9}$	FLD$_{8}$
0EC7$_{16}$							FLD$_{9}$	FLD$_{8}$
0EC8$_{16}$							FLD$_{9}$	FLD$_{8}$
0EC9$_{16}$							FLD$_{9}$	FLD$_{8}$
0ECA$_{16}$								
0ECB$_{16}$								
0ECC$_{16}$								
0ECD$_{16}$								
0ECE$_{16}$								
0ECF$_{16}$								
0ED0$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$
0ED1$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$
0ED2$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$
0ED3$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$
0ED4$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$
0ED5$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$
0ED6$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$
0ED7$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$
0ED8$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$
0ED9$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$

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
0E30$_{16}$							FLD$_{25}$	FLD$_{24}$	
0E31$_{16}$							FLD$_{25}$	FLD$_{24}$	
0E32$_{16}$							FLD$_{25}$	FLD$_{24}$	
0E33$_{16}$							FLD$_{25}$	FLD$_{24}$	
0E34$_{16}$							FLD$_{25}$	FLD$_{24}$	
0E35$_{16}$							FLD$_{25}$	FLD$_{24}$	
0E36$_{16}$							FLD$_{25}$	FLD$_{24}$	
0E37$_{16}$							FLD$_{25}$	FLD$_{24}$	
0E38$_{16}$							FLD$_{25}$	FLD$_{24}$	
0E39$_{16}$							FLD$_{25}$	FLD$_{24}$	
0E3A$_{16}$									
0E3B$_{16}$									
0E3C$_{16}$									
0E3D$_{16}$									
0E3E$_{16}$									
0E3F$_{16}$									
0E40$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0E41$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0E42$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0E43$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0E44$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0E45$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0E46$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0E47$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0E48$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0E49$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0E4A$_{16}$									
0E4B$_{16}$									
0E4C$_{16}$									
0E4D$_{16}$									
0E4E$_{16}$									
0E4F$_{16}$									
0E50$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{25}$(P3$_{1}$)
0E51$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{24}$(P3$_{0}$)
0E52$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{23}$(P1$_{7}$)
0E53$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{22}$(P1$_{6}$)
0E54$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{21}$(P1$_{5}$)
0E55$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{20}$(P1$_{4}$)
0E56$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{19}$(P1$_{3}$)
0E57$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{18}$(P1$_{2}$)
0E58$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{17}$(P1$_{1}$)
0E59$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{16}$(P1$_{0}$)
0E5A$_{16}$									
0E5B$_{16}$									
0E5C$_{16}$									
0E5D$_{16}$									
0E5E$_{16}$									
0E5F$_{16}$									
0E60$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{25}$(P3$_{1}$)
0E61$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{24}$(P3$_{0}$)
0E62$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{23}$(P1$_{7}$)
0E63$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{22}$(P1$_{6}$)
0E64$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{21}$(P1$_{5}$)
0E65$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{20}$(P1$_{4}$)
0E66$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{19}$(P1$_{3}$)
0E67$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{18}$(P1$_{2}$)
0E68$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{17}$(P1$_{1}$)
0E69$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{16}$(P1$_{0}$)

(gray) : Area which is used to sed segment data

(hatched) : Area which is used to sed digit data

(blank) : Area which is available as ordinary RAM

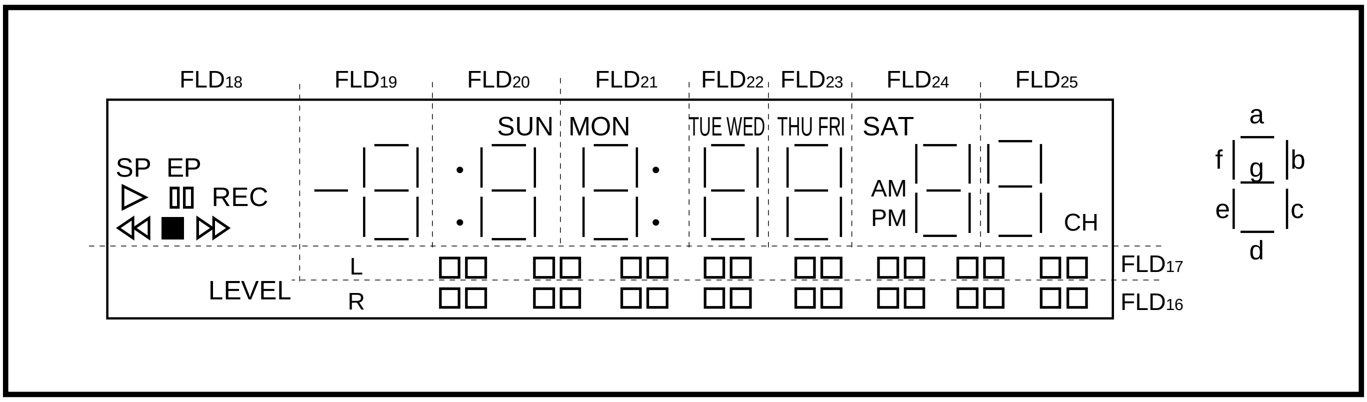


Fig. 2.4.19 FLD digit allocation example

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
0EC0$_{16}$								
0EC1$_{16}$							PM	AM
0EC2$_{16}$								THU
0EC3$_{16}$								TUE
0EC4$_{16}$								:
0EC5$_{16}$								:
0EC6$_{16}$								
0EC7$_{16}$								
0EC8$_{16}$							L	
0EC9$_{16}$							R	LEVEL
0ECA$_{16}$								
0ECB$_{16}$								
0ECC$_{16}$								
0ECD$_{16}$								
0ECE$_{16}$								
0ECF$_{16}$								
0ED0$_{16}$	CH	g	f	e	d	c	b	a
0ED1$_{16}$	SAT	g	f	e	d	c	b	a
0ED2$_{16}$	FRI	g	f	e	d	c	b	a
0ED3$_{16}$	WED	g	f	e	d	c	b	a
0ED4$_{16}$	MON	g	f	e	d	c	b	a
0ED5$_{16}$	SUN	g	f	e	d	c	b	a
0ED6$_{16}$	–	g	f	e	d	c	b	a
0ED7$_{16}$	■	◁◁	▷▷	▯▯	▷	REC	SP	EP
0ED8$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□
0ED9$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□

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
0E50$_{16}$									→ FLD$_{25}$(P3$_1$)
0E51$_{16}$							PM	AM	→ FLD$_{24}$(P3$_0$)
0E52$_{16}$								THU	→ FLD$_{23}$(P1$_7$)
0E53$_{16}$								TUE	→ FLD$_{22}$(P1$_6$)
0E54$_{16}$								:	→ FLD$_{21}$(P1$_5$)
0E55$_{16}$								:	→ FLD$_{20}$(P1$_4$)
0E56$_{16}$									→ FLD$_{19}$(P1$_3$)
0E57$_{16}$									→ FLD$_{18}$(P1$_2$)
0E58$_{16}$							L		→ FLD$_{17}$(P1$_1$)
0E59$_{16}$							R	LEVEL	→ FLD$_{16}$(P1$_0$)
0E5A$_{16}$									
0E5B$_{16}$									
0E5C$_{16}$									
0E5D$_{16}$									
0E5E$_{16}$									
0E5F$_{16}$									
0E60$_{16}$	CH	g	f	e	d	c	b	a	→ FLD$_{25}$(P3$_1$)
0E61$_{16}$	SAT	g	f	e	d	c	b	a	→ FLD$_{24}$(P3$_0$)
0E62$_{16}$	FRI	g	f	e	d	c	b	a	→ FLD$_{23}$(P1$_7$)
0E63$_{16}$	WED	g	f	e	d	c	b	a	→ FLD$_{22}$(P1$_6$)
0E64$_{16}$	MON	g	f	e	d	c	b	a	→ FLD$_{21}$(P1$_5$)
0E65$_{16}$	SUN	g	f	e	d	c	b	a	→ FLD$_{20}$(P1$_4$)
0E66$_{16}$	–	g	f	e	d	c	b	a	→ FLD$_{19}$(P1$_3$)
0E67$_{16}$	■	◁◁	▷▷	▯▯	▷	REC	SP	EP	→ FLD$_{18}$(P1$_2$)
0E68$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□	→ FLD$_{17}$(P1$_1$)
0E69$_{16}$	□□	□□	□□	□□	□□	□□	□□	□□	→ FLD$_{16}$(P1$_0$)

(hatched) : Unused

Control procedure:

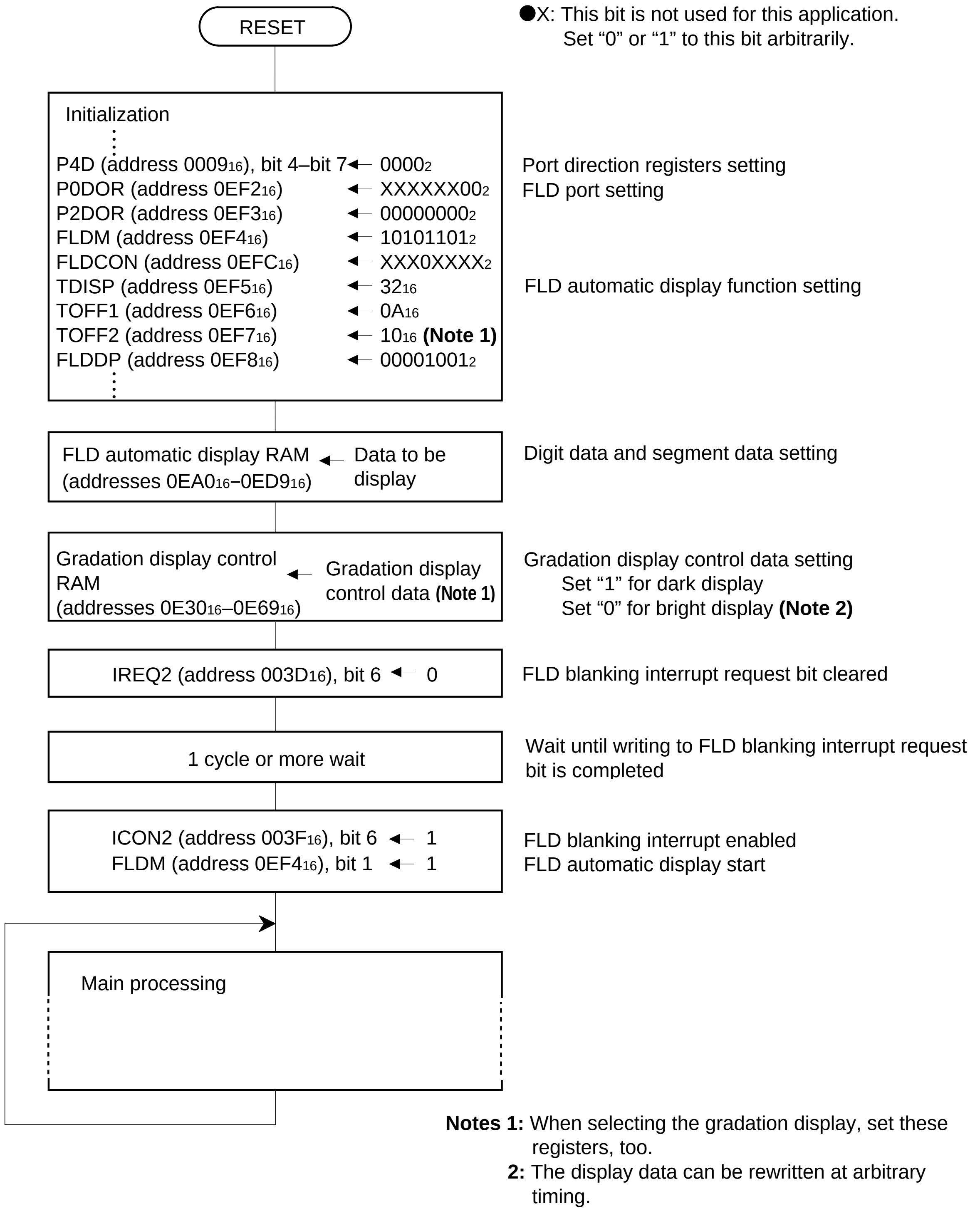
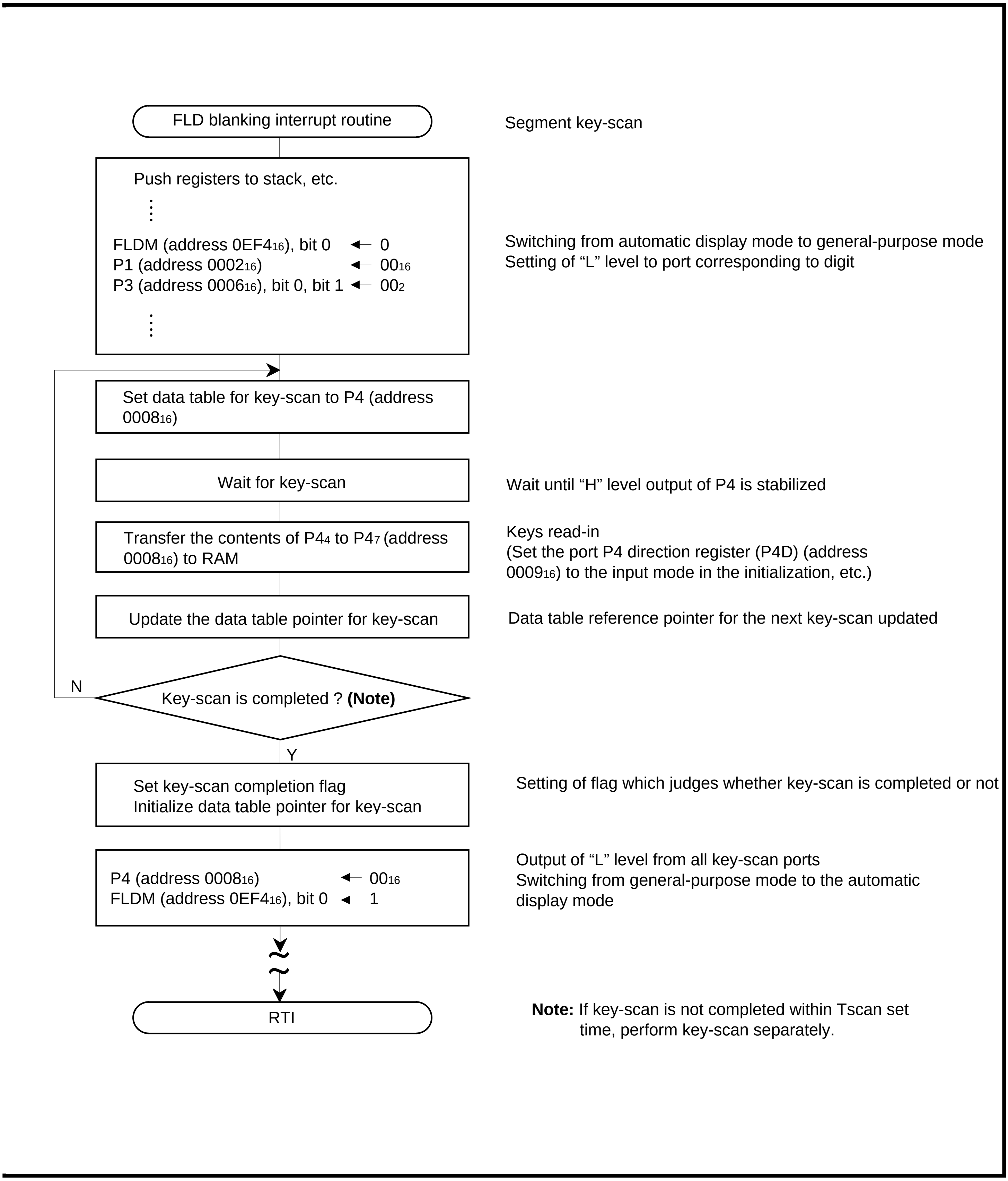


Fig. 2.4.20 Control procedure


FLD blanking interrupt routine
Segment key-scan
Push registers to stack, etc.
FLDM (address 0EF416), bit 0 ← 0
P1 (address 000216) ← 0016
P3 (address 000616), bit 0, bit 1 ← 002
Switching from automatic display mode to general-purpose mode
Setting of “L” level to port corresponding to digit
Set data table for key-scan to P4 (address 000816)
Wait for key-scan
Wait until “H” level output of P4 is stabilized
Transfer the contents of P44 to P47 (address 000816) to RAM
Keys read-in
(Set the port P4 direction register (P4D) (address 000916) to the input mode in the initialization, etc.)
Update the data table pointer for key-scan
Data table reference pointer for the next key-scan updated
N
Key-scan is completed ? (Note)
Y
Set key-scan completion flag
Initialize data table pointer for key-scan
Setting of flag which judges whether key-scan is completed or not
P4 (address 000816) ← 0016
FLDM (address 0EF416), bit 0 ← 1
Output of “L” level from all key-scan ports
Switching from general-purpose mode to the automatic display mode
RTI
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.

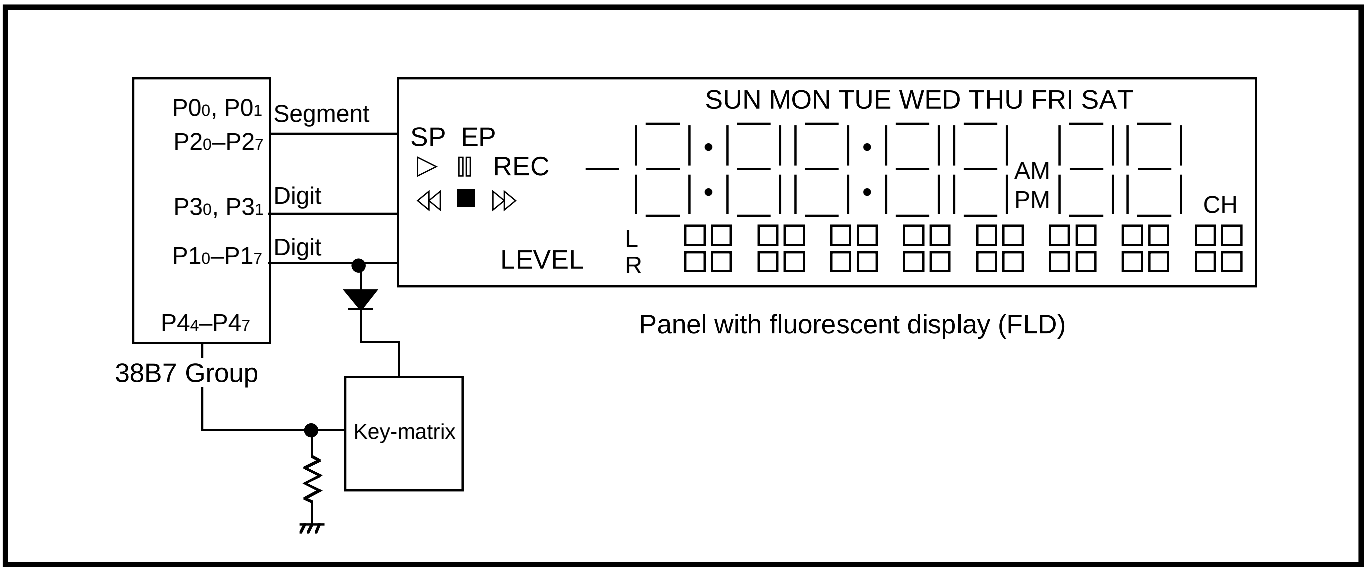


Fig. 2.4.21 Connection diagram

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 µs, Toff2 = 64 µs, Tdisp = 204 µs, Tscan = 0 µs, $f(X_{IN})$ = 4 MHz
- •Use of FLD digit interrupt

Figure 2.4.22 shows the timing chart of key-scan.

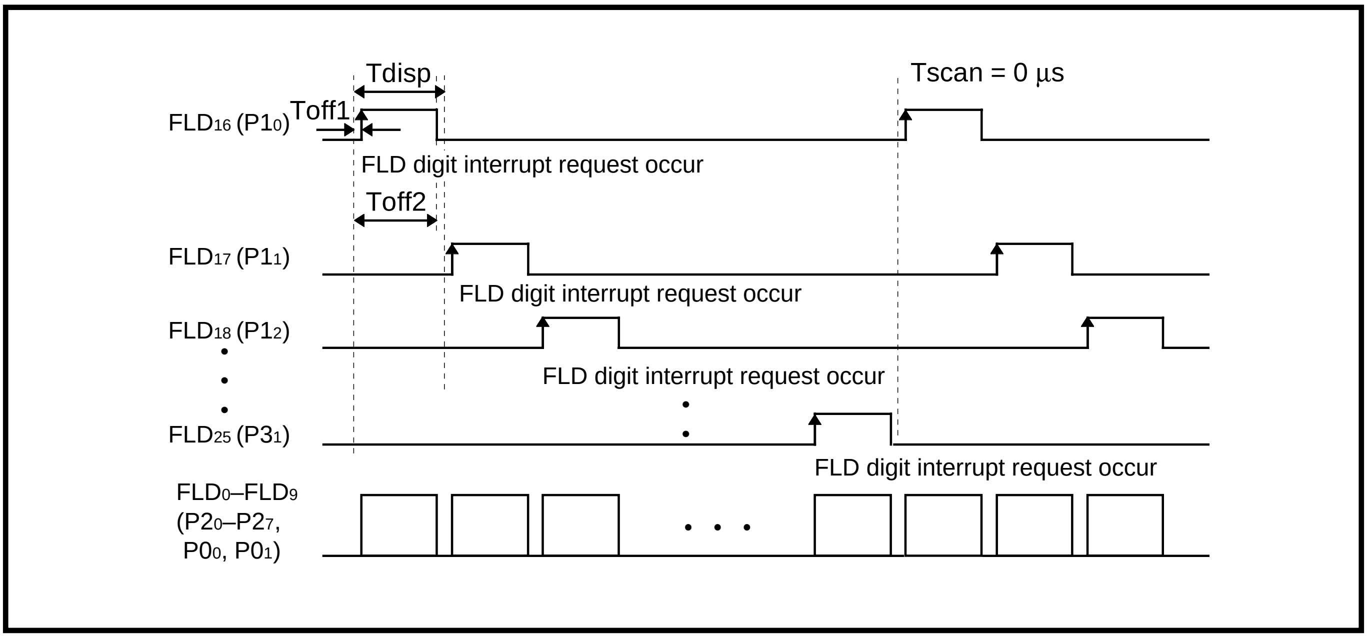


Fig. 2.4.22 Timing chart of key-scan using FLD automatic display mode and digits

Figure 2.4.23 shows the setting of relevant registers.

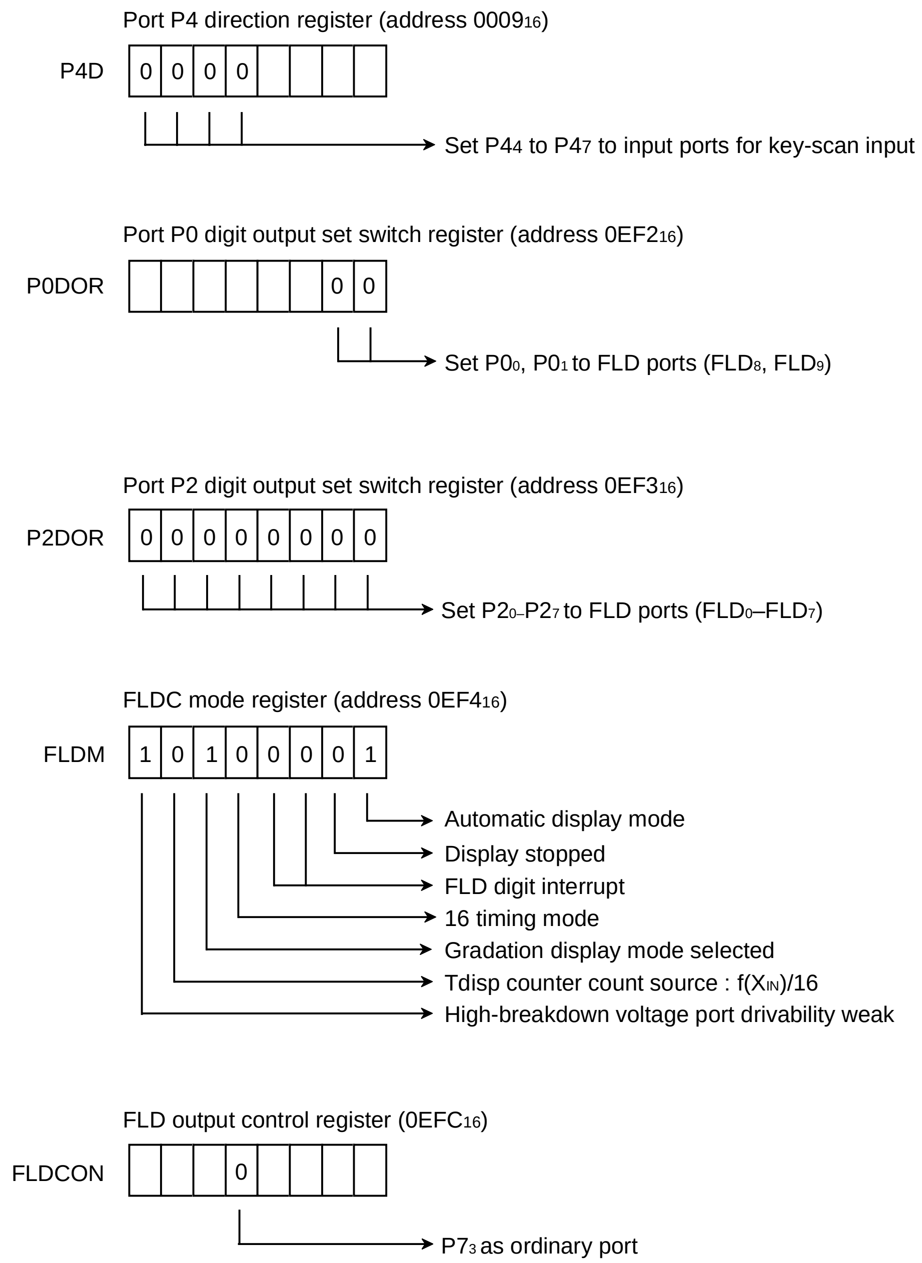
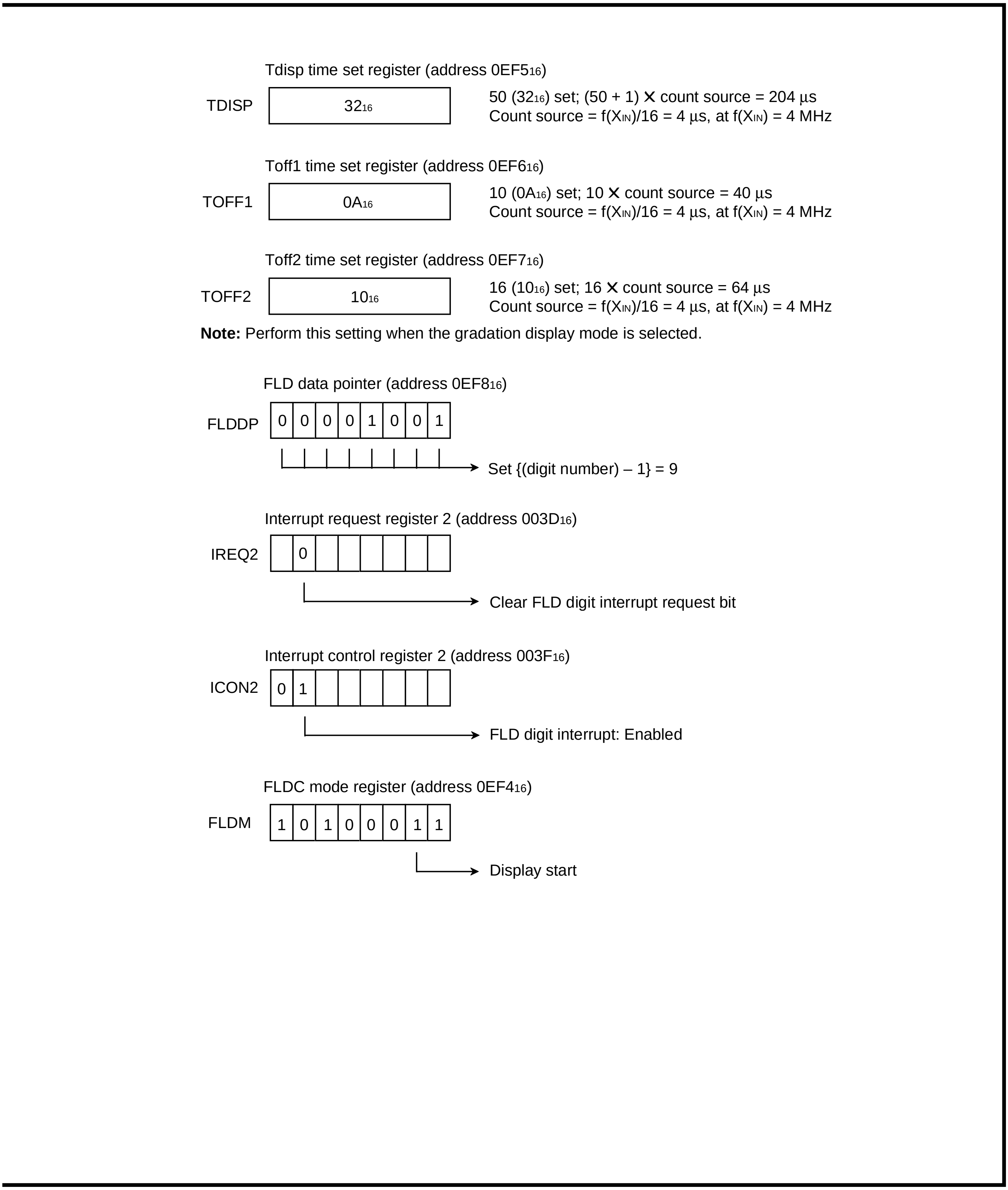


Fig. 2.4.23 Setting of relevant registers


Tdisp time set register (address 0EF5₁₆)
TDISP
32₁₆
50 (32₁₆) set; (50 + 1) × count source = 204 µs
Count source = f(XIN)/16 = 4 µs, at f(XIN) = 4 MHz
Toff1 time set register (address 0EF6₁₆)
TOFF1
0A₁₆
10 (0A₁₆) set; 10 × count source = 40 µs
Count source = f(XIN)/16 = 4 µs, at f(XIN) = 4 MHz
Toff2 time set register (address 0EF7₁₆)
TOFF2
10₁₆
16 (10₁₆) set; 16 × count source = 64 µs
Count source = f(XIN)/16 = 4 µs, at f(XIN) = 4 MHz
Note: Perform this setting when the gradation display mode is selected.
FLD data pointer (address 0EF8₁₆)
FLDDP
0 0 0 0 1 0 0 1
Set {(digit number) – 1} = 9
Interrupt request register 2 (address 003D₁₆)
IREQ2
0
Clear FLD digit interrupt request bit
Interrupt control register 2 (address 003F₁₆)
ICON2
0 1
FLD digit interrupt: Enabled
FLDC mode register (address 0EF4₁₆)
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 | Gradation display control data stored area | Corresponding digit pin

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Corresponding digit pin
0EA0$_{16}$							FLD$_{25}$	FLD$_{24}$	0E30$_{16}$							FLD$_{25}$	FLD$_{24}$	
0EA1$_{16}$							FLD$_{25}$	FLD$_{24}$	0E31$_{16}$							FLD$_{25}$	FLD$_{24}$	
0EA2$_{16}$							FLD$_{25}$	FLD$_{24}$	0E32$_{16}$							FLD$_{25}$	FLD$_{24}$	
0EA3$_{16}$							FLD$_{25}$	FLD$_{24}$	0E33$_{16}$							FLD$_{25}$	FLD$_{24}$	
0EA4$_{16}$							FLD$_{25}$	FLD$_{24}$	0E34$_{16}$							FLD$_{25}$	FLD$_{24}$	
0EA5$_{16}$							FLD$_{25}$	FLD$_{24}$	0E35$_{16}$							FLD$_{25}$	FLD$_{24}$	
0EA6$_{16}$							FLD$_{25}$	FLD$_{24}$	0E36$_{16}$							FLD$_{25}$	FLD$_{24}$	
0EA7$_{16}$							FLD$_{25}$	FLD$_{24}$	0E37$_{16}$							FLD$_{25}$	FLD$_{24}$	
0EA8$_{16}$							FLD$_{25}$	FLD$_{24}$	0E38$_{16}$							FLD$_{25}$	FLD$_{24}$	
0EA9$_{16}$							FLD$_{25}$	FLD$_{24}$	0E39$_{16}$							FLD$_{25}$	FLD$_{24}$	
0EAA$_{16}$									0E3A$_{16}$									
0EAB$_{16}$									0E3B$_{16}$									
0EAC$_{16}$									0E3C$_{16}$									
0EAD$_{16}$									0E3D$_{16}$									
0EAE$_{16}$									0E3E$_{16}$									
0EAF$_{16}$									0E3F$_{16}$									
0EB0$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	0E40$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0EB1$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	0E41$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0EB2$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	0E42$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0EB3$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	0E43$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0EB4$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	0E44$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0EB5$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	0E45$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0EB6$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	0E46$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0EB7$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	0E47$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0EB8$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	0E48$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0EB9$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	0E49$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	
0EBA$_{16}$									0E4A$_{16}$									
0EBB$_{16}$									0E4B$_{16}$									
0EBC$_{16}$									0E4C$_{16}$									
0EBD$_{16}$									0E4D$_{16}$									
0EBE$_{16}$									0E4E$_{16}$									
0EBF$_{16}$									0E4F$_{16}$									
0EC0$_{16}$							FLD$_{9}$	FLD$_{8}$	0E50$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{25}$(P3$_{1}$)
0EC1$_{16}$							FLD$_{9}$	FLD$_{8}$	0E51$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{24}$(P3$_{0}$)
0EC2$_{16}$							FLD$_{9}$	FLD$_{8}$	0E52$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{23}$(P1$_{7}$)
0EC3$_{16}$							FLD$_{9}$	FLD$_{8}$	0E53$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{22}$(P1$_{6}$)
0EC4$_{16}$							FLD$_{9}$	FLD$_{8}$	0E54$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{21}$(P1$_{5}$)
0EC5$_{16}$							FLD$_{9}$	FLD$_{8}$	0E55$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{20}$(P1$_{4}$)
0EC6$_{16}$							FLD$_{9}$	FLD$_{8}$	0E56$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{19}$(P1$_{3}$)
0EC7$_{16}$							FLD$_{9}$	FLD$_{8}$	0E57$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{18}$(P1$_{2}$)
0EC8$_{16}$							FLD$_{9}$	FLD$_{8}$	0E58$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{17}$(P1$_{1}$)
0EC9$_{16}$							FLD$_{9}$	FLD$_{8}$	0E59$_{16}$							FLD$_{9}$	FLD$_{8}$	→ FLD$_{16}$(P1$_{0}$)
0ECA$_{16}$									0E5A$_{16}$									
0ECB$_{16}$									0E5B$_{16}$									
0ECC$_{16}$									0E5C$_{16}$									
0ECD$_{16}$									0E5D$_{16}$									
0ECE$_{16}$									0E5E$_{16}$									
0ECF$_{16}$									0E5F$_{16}$									
0ED0$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	0E60$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{25}$(P3$_{1}$)
0ED1$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	0E61$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{24}$(P3$_{0}$)
0ED2$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	0E62$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{23}$(P1$_{7}$)
0ED3$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	0E63$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{22}$(P1$_{6}$)
0ED4$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	0E64$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{21}$(P1$_{5}$)
0ED5$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	0E65$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{20}$(P1$_{4}$)
0ED6$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	0E66$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{19}$(P1$_{3}$)
0ED7$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	0E67$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{18}$(P1$_{2}$)
0ED8$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	0E68$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{17}$(P1$_{1}$)
0ED9$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	0E69$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ FLD$_{16}$(P1$_{0}$)

(grey) : Area which is used to set segment data

(hatched) : Area which is used to set digit data

(blank) : Area which is available as ordinary RAM

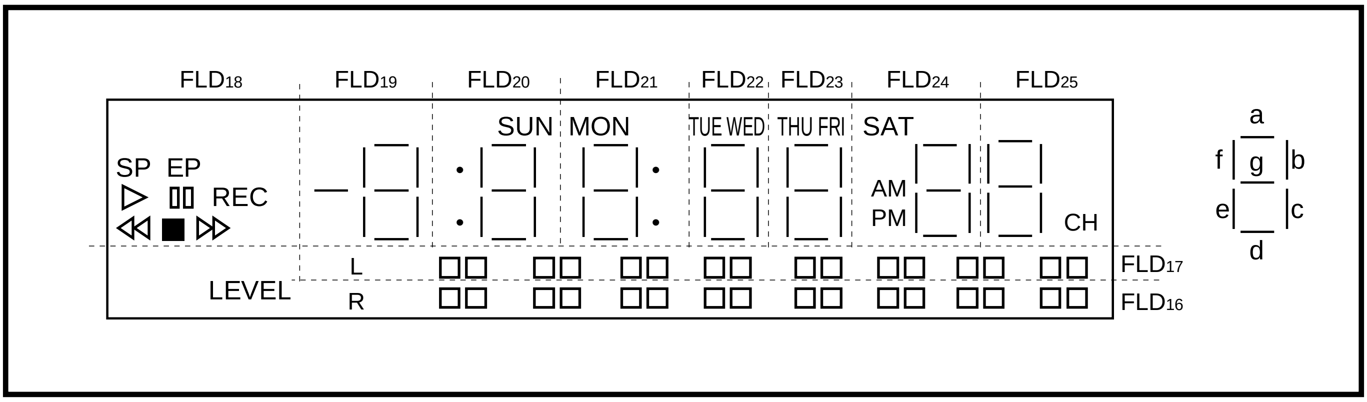


Fig. 2.4.24 FLD digit allocation example

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
$0EC0_{16}$								
$0EC1_{16}$							PM	AM
$0EC2_{16}$								THU
$0EC3_{16}$								TUE
$0EC4_{16}$								:
$0EC5_{16}$								:
$0EC6_{16}$								
$0EC7_{16}$								
$0EC8_{16}$							L	
$0EC9_{16}$							R	LEVEL
$0ECA_{16}$								
$0ECB_{16}$								
$0ECC_{16}$								
$0ECD_{16}$								
$0ECE_{16}$								
$0ECF_{16}$								
$0ED0_{16}$	CH	g	f	e	d	c	b	a
$0ED1_{16}$	SAT	g	f	e	d	c	b	a
$0ED2_{16}$	FRI	g	f	e	d	c	b	a
$0ED3_{16}$	WED	g	f	e	d	c	b	a
$0ED4_{16}$	MON	g	f	e	d	c	b	a
$0ED5_{16}$	SUN	g	f	e	d	c	b	a
$0ED6_{16}$	–	g	f	e	d	c	b	a
$0ED7_{16}$	■	◁◁	▷▷	▯▯	▷	REC	SP	EP
$0ED8_{16}$	□□	□□	□□	□□	□□	□□	□□	□□
$0ED9_{16}$	□□	□□	□□	□□	□□	□□	□□	□□

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
$0E50_{16}$									→ $FLD_{25}(P3_1)$
$0E51_{16}$							PM	AM	→ $FLD_{24}(P3_0)$
$0E52_{16}$								THU	→ $FLD_{23}(P1_7)$
$0E53_{16}$								TUE	→ $FLD_{22}(P1_6)$
$0E54_{16}$								:	→ $FLD_{21}(P1_5)$
$0E55_{16}$								:	→ $FLD_{20}(P1_4)$
$0E56_{16}$									→ $FLD_{19}(P1_3)$
$0E57_{16}$									→ $FLD_{18}(P1_2)$
$0E58_{16}$							L		→ $FLD_{17}(P1_1)$
$0E59_{16}$							R	LEVEL	→ $FLD_{16}(P1_0)$
$0E5A_{16}$									
$0E5B_{16}$									
$0E5C_{16}$									
$0E5D_{16}$									
$0E5E_{16}$									
$0E5F_{16}$									
$0E60_{16}$	CH	g	f	e	d	c	b	a	→ $FLD_{25}(P3_1)$
$0E61_{16}$	SAT	g	f	e	d	c	b	a	→ $FLD_{24}(P3_0)$
$0E62_{16}$	FRI	g	f	e	d	c	b	a	→ $FLD_{23}(P1_7)$
$0E63_{16}$	WED	g	f	e	d	c	b	a	→ $FLD_{22}(P1_6)$
$0E64_{16}$	MON	g	f	e	d	c	b	a	→ $FLD_{21}(P1_5)$
$0E65_{16}$	SUN	g	f	e	d	c	b	a	→ $FLD_{20}(P1_4)$
$0E66_{16}$	–	g	f	e	d	c	b	a	→ $FLD_{19}(P1_3)$
$0E67_{16}$	■	◁◁	▷▷	▯▯	▷	REC	SP	EP	→ $FLD_{18}(P1_2)$
$0E68_{16}$	□□	□□	□□	□□	□□	□□	□□	□□	→ $FLD_{17}(P1_1)$
$0E69_{16}$	□□	□□	□□	□□	□□	□□	□□	□□	→ $FLD_{16}(P1_0)$

(hatched) : Unused

Control procedure:

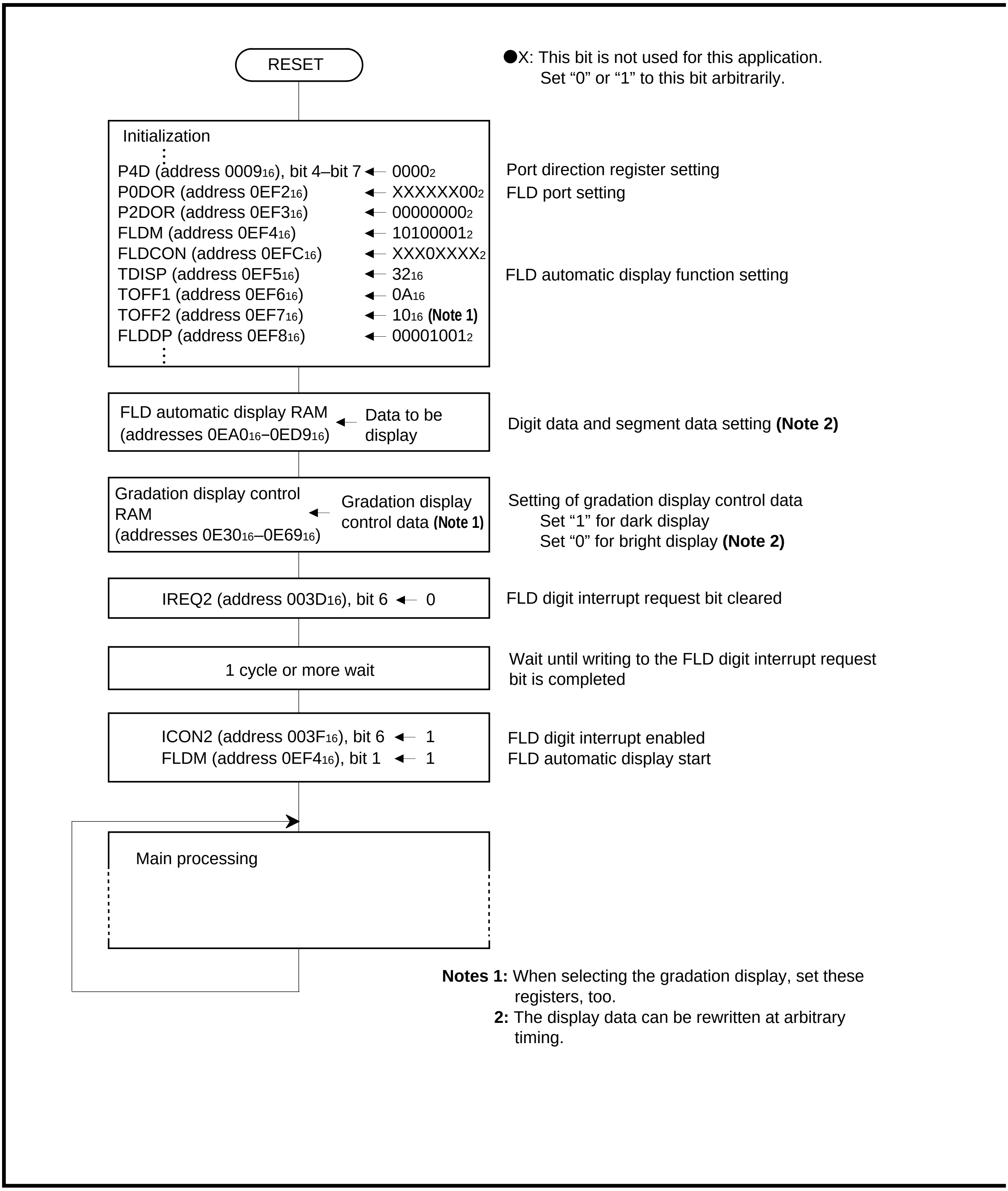
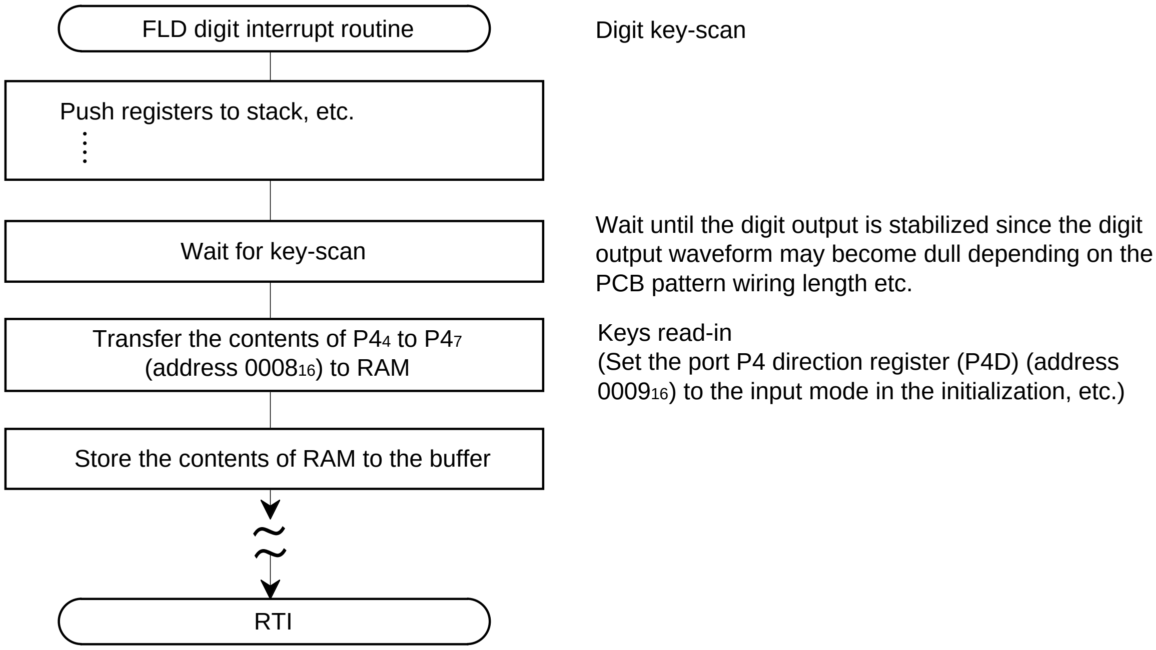


Fig. 2.4.25 Control procedure


FLD digit interrupt routine
Digit key-scan
Push registers to stack, etc.
Wait for 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.
Transfer the contents of P4$_4$ to P4$_7$ (address 0008$_{16}$) to RAM
Keys read-in
(Set the port P4 direction register (P4D) (address 0009$_{16}$) to the input mode in the initialization, etc.)
Store the contents of RAM to the buffer
RTI

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

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

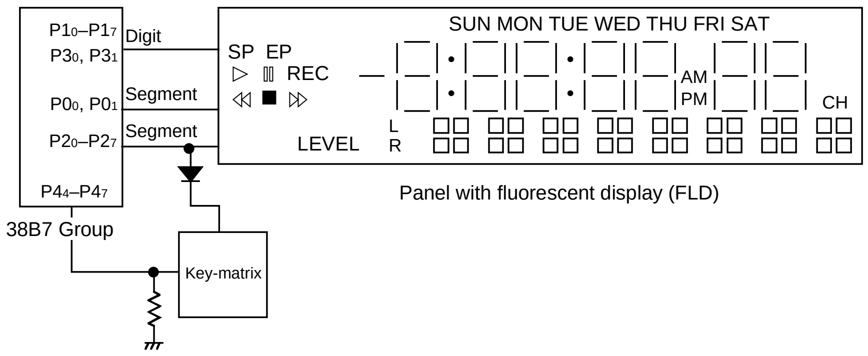


Fig. 2.4.26 Connection diagram

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 in Figure 2.4.28 by software and perform key-scan.

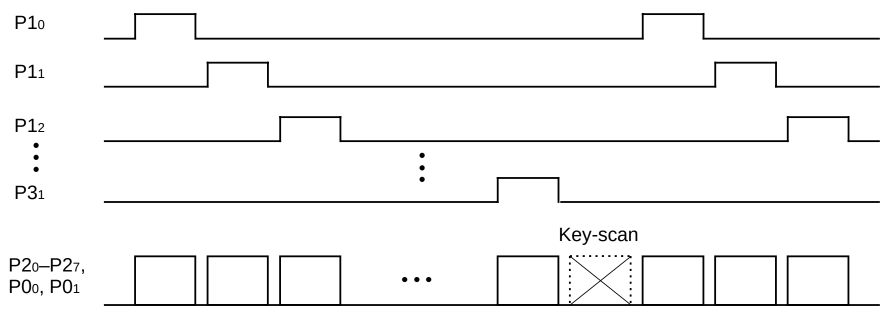


Fig. 2.4.27 Timing chart of FLD display by software

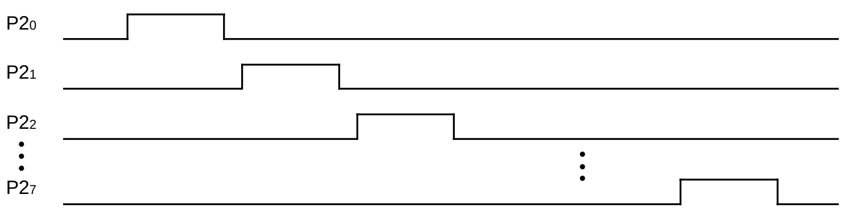


Fig. 2.4.28 Enlarged view of $P2_0$ to $P2_7$ key-scan

Figure 2.4.29 shows the setting of relevant registers.

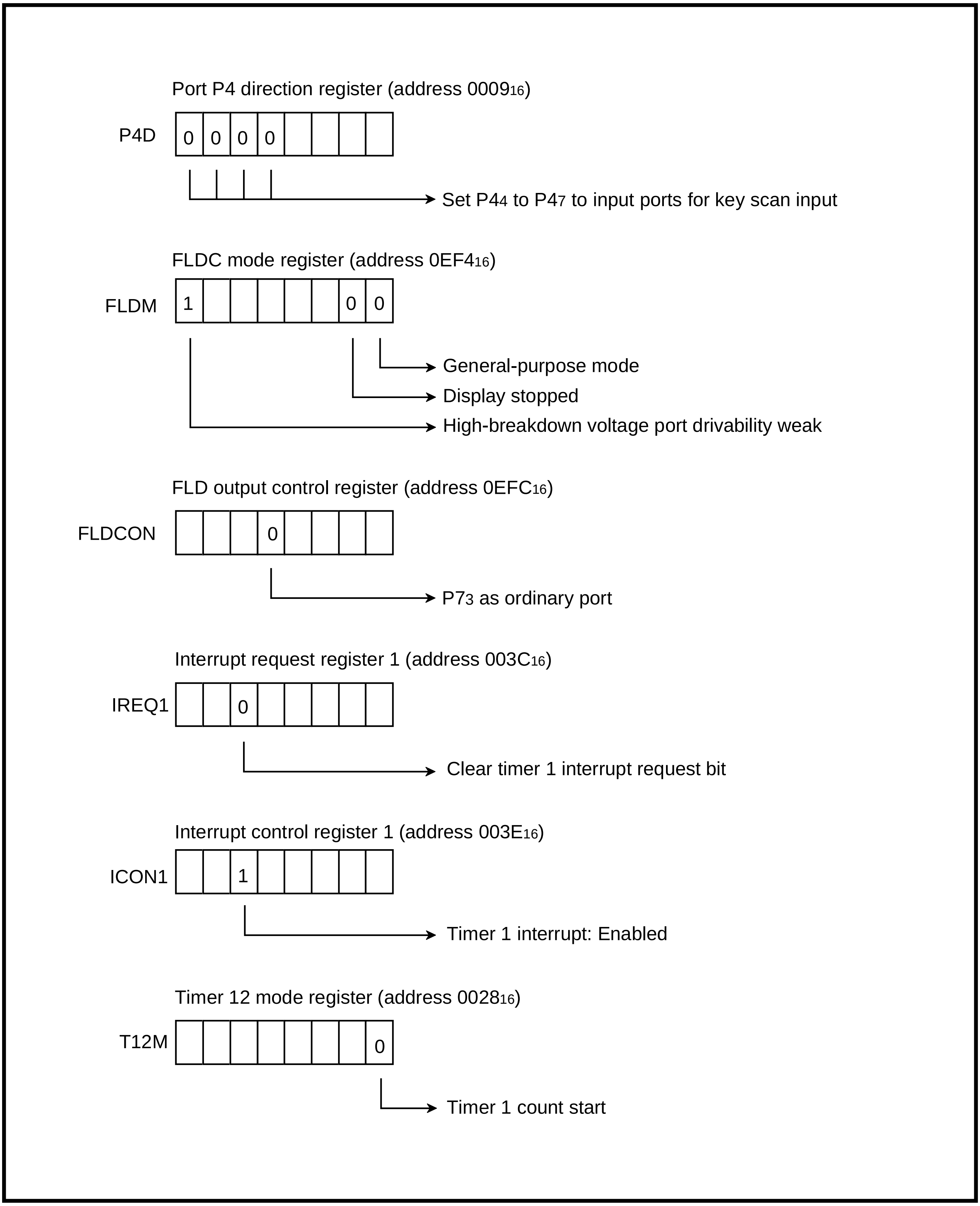


Fig. 2.4.29 Setting of relevant registers

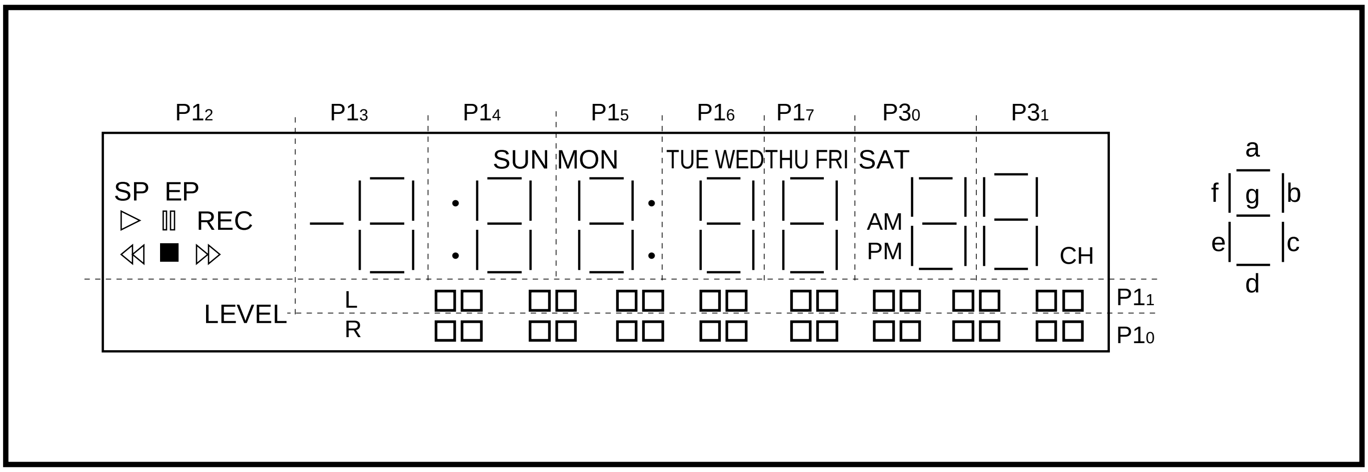


Fig. 2.4.30 FLD digit allocation example

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
$0EC0_{16}$									→ $P3_1$
$0EC1_{16}$							PM	AM	→ $P3_0$
$0EC2_{16}$								THU	→ $P1_7$
$0EC3_{16}$								TUE	→ $P1_6$
$0EC4_{16}$								:	→ $P1_5$
$0EC5_{16}$								:	→ $P1_4$
$0EC6_{16}$									→ $P1_3$
$0EC7_{16}$									→ $P1_2$
$0EC8_{16}$							L		→ $P1_1$
$0EC9_{16}$							R	LEVEL	→ $P1_0$
$0ECA_{16}$									
$0ECB_{16}$									
$0ECC_{16}$									
$0ECD_{16}$									
$0ECE_{16}$									
$0ECF_{16}$									
$0ED0_{16}$	CH	g	f	e	d	c	b	a	→ $P3_1$
$0ED1_{16}$	SAT	g	f	e	d	c	b	a	→ $P3_0$
$0ED2_{16}$	FRI	g	f	e	d	c	b	a	→ $P1_7$
$0ED3_{16}$	WED	g	f	e	d	c	b	a	→ $P1_6$
$0ED4_{16}$	MON	g	f	e	d	c	b	a	→ $P1_5$
$0ED5_{16}$	SUN	g	f	e	d	c	b	a	→ $P1_4$
$0ED6_{16}$	–	g	f	e	d	c	b	a	→ $P1_3$
$0ED7_{16}$	■	◁◁	▷▷	▯▯	▷	REC	SP	EP	→ $P1_2$
$0ED8_{16}$	□□	□□	□□	□□	□□	□□	□□	□□	→ $P1_1$
$0ED9_{16}$	□□	□□	□□	□□	□□	□□	□□	□□	→ $P1_0$

(shaded) : Unused

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

Control procedure:

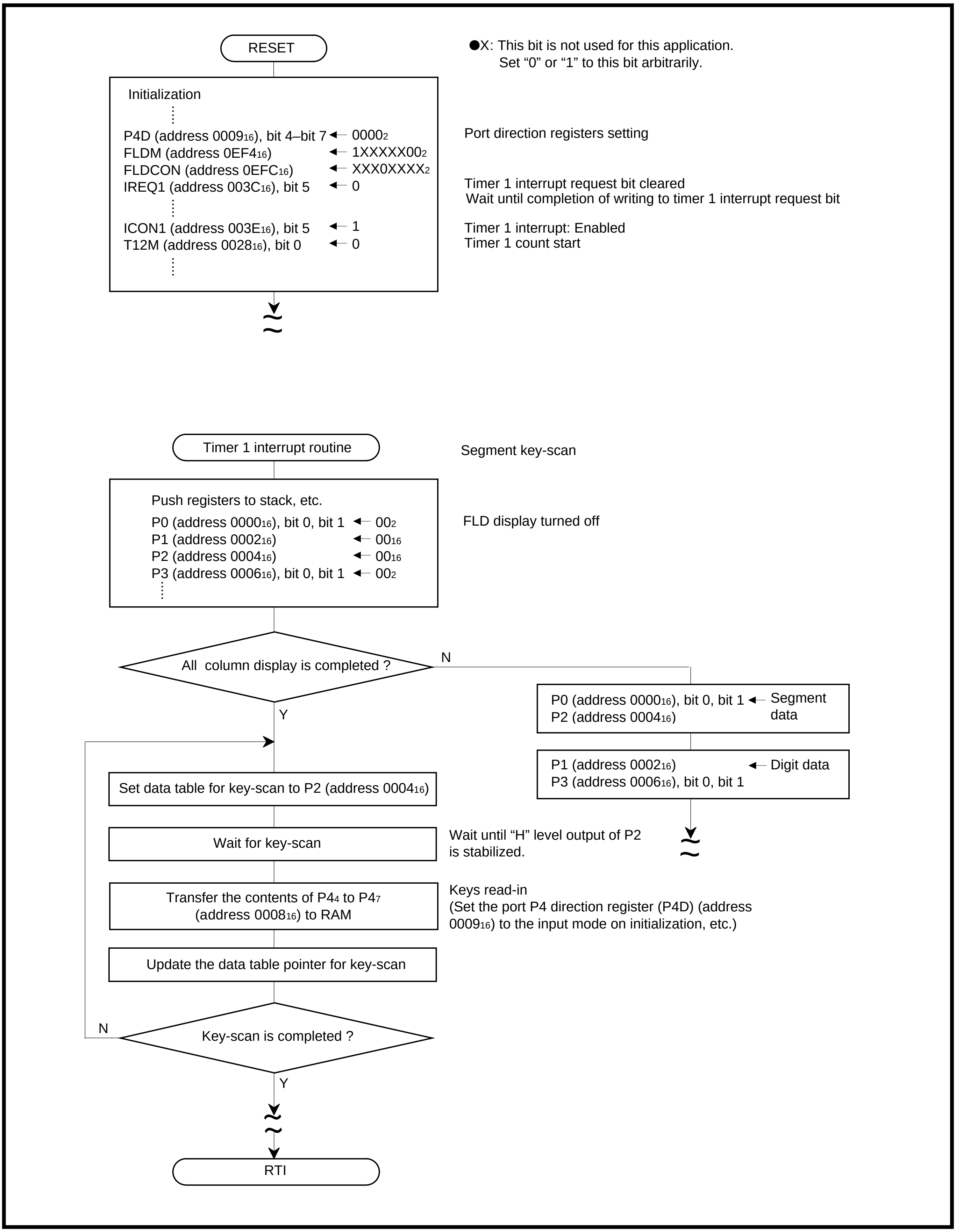


Fig. 2.4.31 Control procedure

(4) Display by combination with digit expander (M35501FP*) (basic combination example)
* For M35501FP, refer to section **“3.9 M35501FP”**.

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

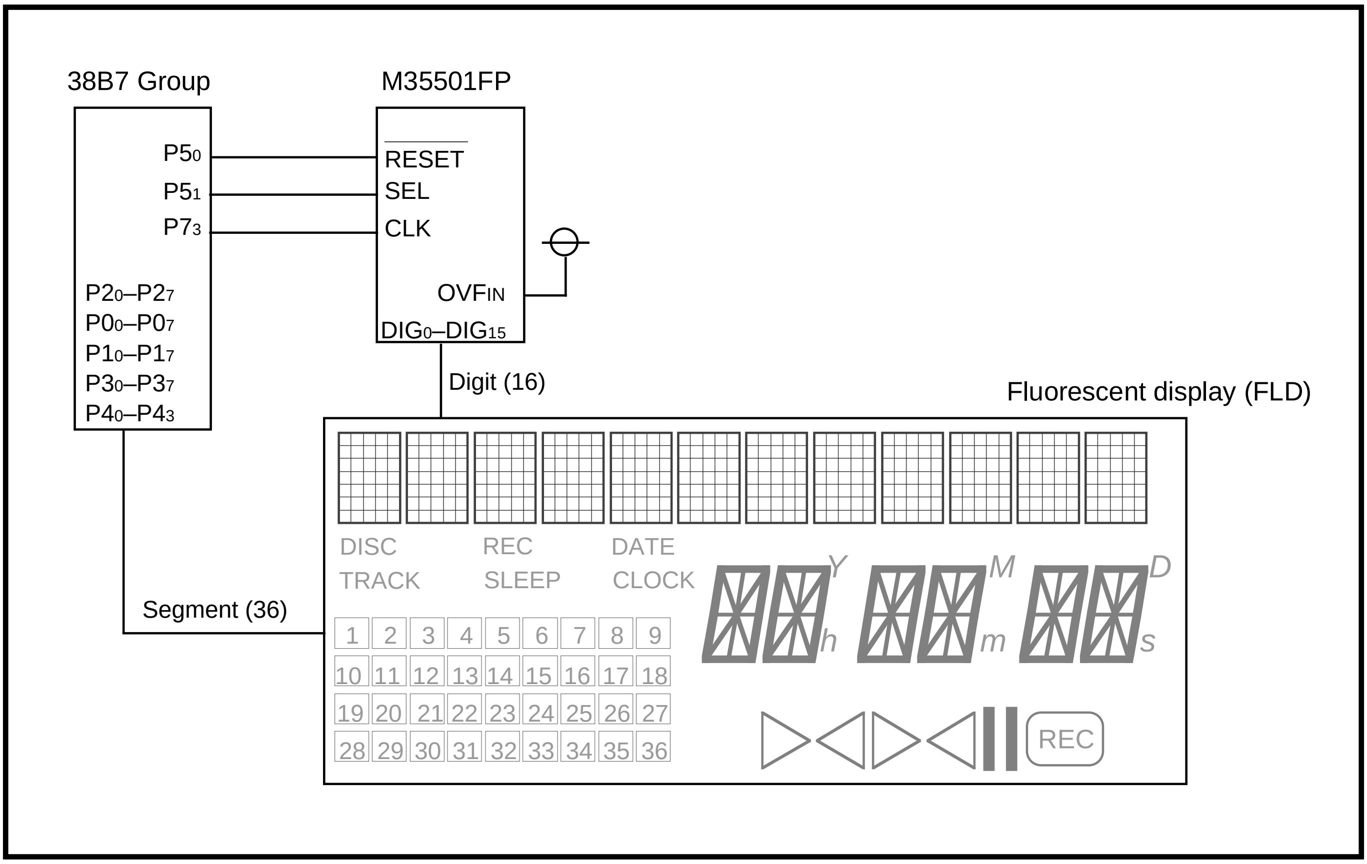


Fig. 2.4.32 Connection diagram

Specifications: •Use of M35501FP (M35501FP: 16 digits, 38B7 Group: 36 segments)
Ports $P5_0$ and $P5_1$ of 38B7 Group supply signals to the $\overline{RESET}$ and SEL pins of M35501FP respectively.
The $P7_3$ pin (dimmer output pin) supply signals to the CLK pin of M35501FP.
•Use of FLD automatic display mode of 38B7 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 38B7 Group and M35501FP, and Figure 2.4.34 shows the timing chart (enlarged view) of digit and segment output.

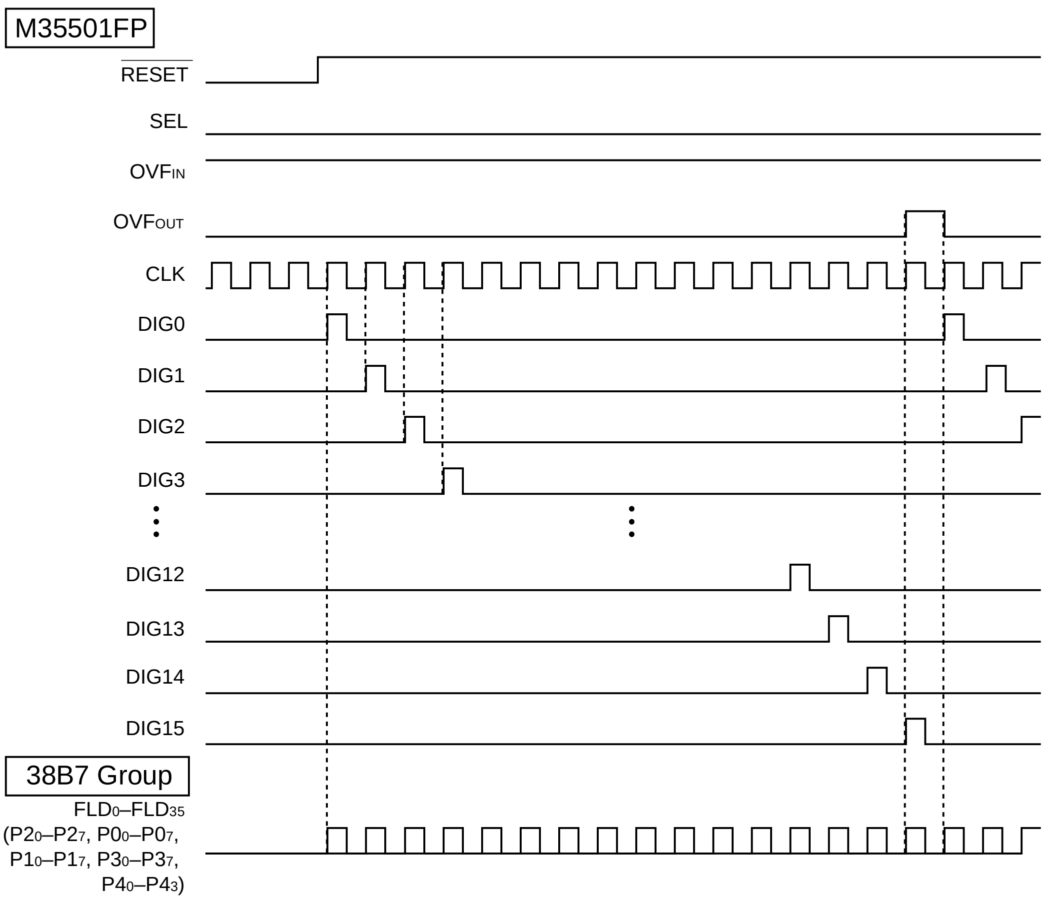


Fig. 2.4.33 Timing chart of 38B7 Group and M35501FP

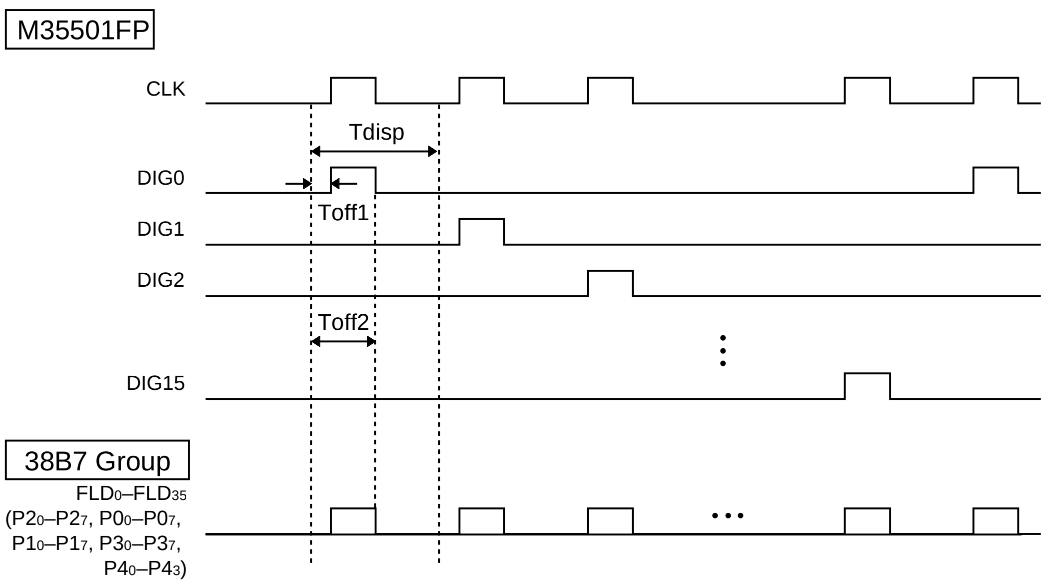
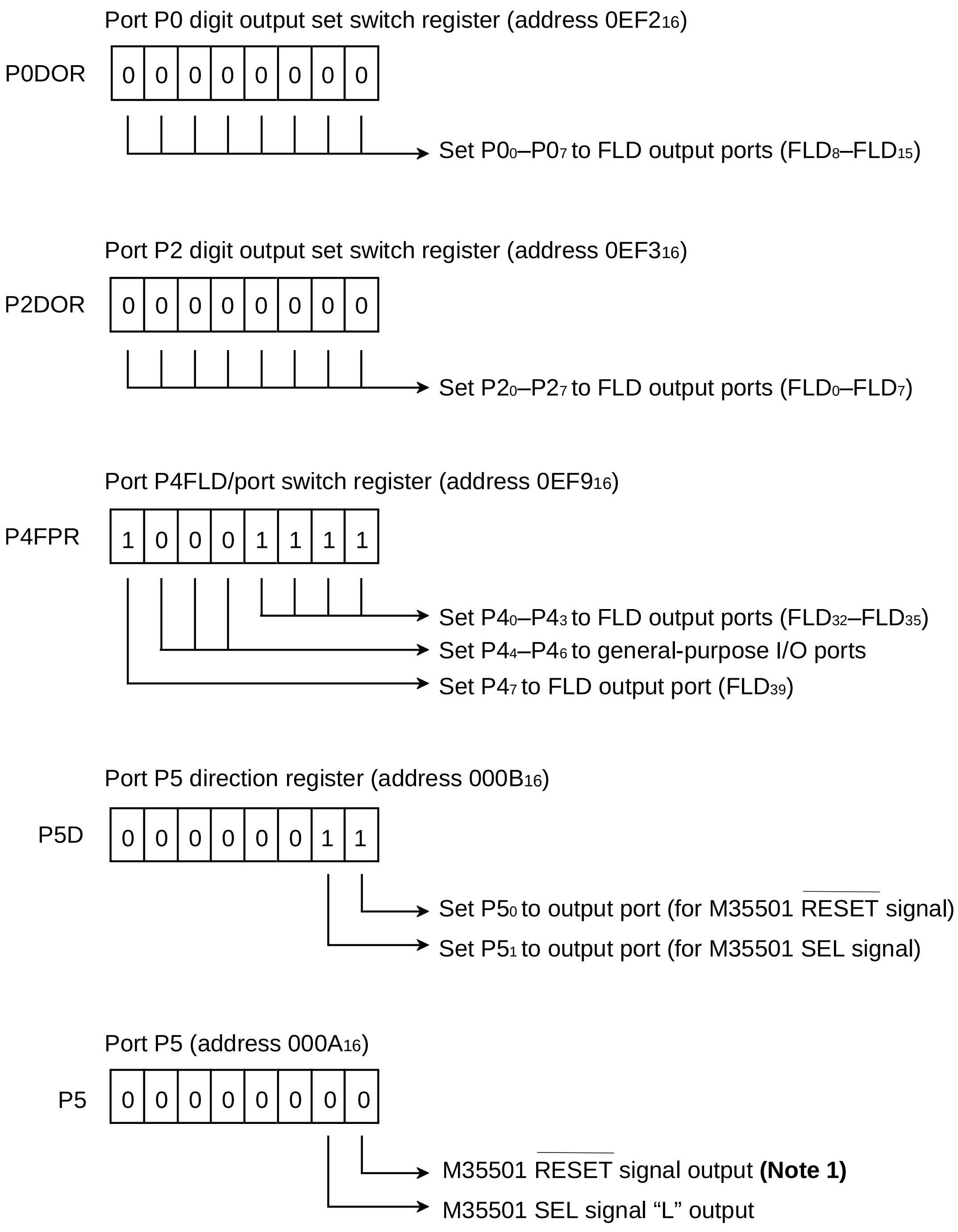


Fig. 2.4.34 Timing chart (enlarged view) of digit and segment output

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

- b0 → Automatic display mode
- b1 → Display stopped
- b2, b3 → FLD digit interrupt
- b4 → 16 timing mode
- b5 → Gradation display mode selected
- b6 → Tdisp counter count source : $f(X_{IN})/16$
- b7 → High-breakdown voltage port drivability weak

FLD output control register (address $0EFC_{16}$)

FLDCON: _ _ _ 1 _ _ _ _

- b4 → $P7_3$ as dimmer output

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: $0A_{16}$

10 ($0A_{16}$) 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

- b7–b0 → Set {(digit number) – 1} = 15

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

FLDM: 1 0 1 0 0 0 1 1

- b1 → 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 | Gradation display control data stored area | Corresponding digit pin of M35501FP

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Corresponding digit pin of M35501FP
0E90$_{16}$	1				FLD$_{35}$	FLD$_{34}$	FLD$_{33}$	FLD$_{32}$	0E20$_{16}$	1				FLD$_{35}$	FLD$_{34}$	FLD$_{33}$	FLD$_{32}$	→ DIG$_{15}$
0E91$_{16}$	1								0E21$_{16}$	1								→ DIG$_{14}$
0E92$_{16}$	1								0E22$_{16}$	1								→ DIG$_{13}$
0E93$_{16}$	1								0E23$_{16}$	1								→ DIG$_{12}$
0E94$_{16}$	1								0E24$_{16}$	1								→ DIG$_{11}$
0E95$_{16}$	1								0E25$_{16}$	1								→ DIG$_{10}$
0E96$_{16}$	1								0E26$_{16}$	1								→ DIG$_{9}$
0E97$_{16}$	1								0E27$_{16}$	1								→ DIG$_{8}$
0E98$_{16}$	1								0E28$_{16}$	1								→ DIG$_{7}$
0E99$_{16}$	1								0E29$_{16}$	1								→ DIG$_{6}$
0E9A$_{16}$	1								0E2A$_{16}$	1								→ DIG$_{5}$
0E9B$_{16}$	1								0E2B$_{16}$	1								→ DIG$_{4}$
0E9C$_{16}$	1								0E2C$_{16}$	1								→ DIG$_{3}$
0E9D$_{16}$	1								0E2D$_{16}$	1								→ DIG$_{2}$
0E9E$_{16}$	1				▼	▼	▼	▼	0E2E$_{16}$	1				▼	▼	▼	▼	→ DIG$_{1}$
0E9F$_{16}$	1								0E2F$_{16}$	1								→ DIG$_{0}$
0EA0$_{16}$	FLD$_{31}$	FLD$_{30}$	FLD$_{29}$	FLD$_{28}$	FLD$_{27}$	FLD$_{26}$	FLD$_{25}$	FLD$_{24}$	0E30$_{16}$	FLD$_{31}$	FLD$_{30}$	FLD$_{29}$	FLD$_{28}$	FLD$_{27}$	FLD$_{26}$	FLD$_{25}$	FLD$_{24}$	→ DIG$_{15}$
0EA1$_{16}$									0E31$_{16}$									→ DIG$_{14}$
0EA2$_{16}$									0E32$_{16}$									→ DIG$_{13}$
0EA3$_{16}$									0E33$_{16}$									→ DIG$_{12}$
0EA4$_{16}$									0E34$_{16}$									→ DIG$_{11}$
0EA5$_{16}$									0E35$_{16}$									→ DIG$_{10}$
0EA6$_{16}$									0E36$_{16}$									→ DIG$_{9}$
0EA7$_{16}$									0E37$_{16}$									→ DIG$_{8}$
0EA8$_{16}$									0E38$_{16}$									→ DIG$_{7}$
0EA9$_{16}$									0E39$_{16}$									→ DIG$_{6}$
0EAA$_{16}$									0E3A$_{16}$									→ DIG$_{5}$
0EAB$_{16}$									0E3B$_{16}$									→ DIG$_{4}$
0EAC$_{16}$									0E3C$_{16}$									→ DIG$_{3}$
0EAD$_{16}$									0E3D$_{16}$									→ DIG$_{2}$
0EAE$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼	0E3E$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼	→ DIG$_{1}$
0EAF$_{16}$									0E3F$_{16}$									→ DIG$_{0}$
0EB0$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	0E40$_{16}$	FLD$_{23}$	FLD$_{22}$	FLD$_{21}$	FLD$_{20}$	FLD$_{19}$	FLD$_{18}$	FLD$_{17}$	FLD$_{16}$	→ DIG$_{15}$
0EB1$_{16}$									0E41$_{16}$									→ DIG$_{14}$
0EB2$_{16}$									0E42$_{16}$									→ DIG$_{13}$
0EB3$_{16}$									0E43$_{16}$									→ DIG$_{12}$
0EB4$_{16}$									0E44$_{16}$									→ DIG$_{11}$
0EB5$_{16}$									0E45$_{16}$									→ DIG$_{10}$
0EB6$_{16}$									0E46$_{16}$									→ DIG$_{9}$
0EB7$_{16}$									0E47$_{16}$									→ DIG$_{8}$
0EB8$_{16}$									0E48$_{16}$									→ DIG$_{7}$
0EB9$_{16}$									0E49$_{16}$									→ DIG$_{6}$
0EBA$_{16}$									0E4A$_{16}$									→ DIG$_{5}$
0EBB$_{16}$									0E4B$_{16}$									→ DIG$_{4}$
0EBC$_{16}$									0E4C$_{16}$									→ DIG$_{3}$
0EBD$_{16}$									0E4D$_{16}$									→ DIG$_{2}$
0EBE$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼	0E4E$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼	→ DIG$_{1}$
0EBF$_{16}$									0E4F$_{16}$									→ DIG$_{0}$
0EC0$_{16}$	FLD$_{15}$	FLD$_{14}$	FLD$_{13}$	FLD$_{12}$	FLD$_{11}$	FLD$_{10}$	FLD$_{9}$	FLD$_{8}$	0E50$_{16}$	FLD$_{15}$	FLD$_{14}$	FLD$_{13}$	FLD$_{12}$	FLD$_{11}$	FLD$_{10}$	FLD$_{9}$	FLD$_{8}$	→ DIG$_{15}$
0EC1$_{16}$									0E51$_{16}$									→ DIG$_{14}$
0EC2$_{16}$									0E52$_{16}$									→ DIG$_{13}$
0EC3$_{16}$									0E53$_{16}$									→ DIG$_{12}$
0EC4$_{16}$									0E54$_{16}$									→ DIG$_{11}$
0EC5$_{16}$									0E55$_{16}$									→ DIG$_{10}$
0EC6$_{16}$									0E56$_{16}$									→ DIG$_{9}$
0EC7$_{16}$									0E57$_{16}$									→ DIG$_{8}$
0EC8$_{16}$									0E58$_{16}$									→ DIG$_{7}$
0EC9$_{16}$									0E59$_{16}$									→ DIG$_{6}$
0ECA$_{16}$									0E5A$_{16}$									→ DIG$_{5}$
0ECB$_{16}$									0E5B$_{16}$									→ DIG$_{4}$
0ECC$_{16}$									0E5C$_{16}$									→ DIG$_{3}$
0ECD$_{16}$									0E5D$_{16}$									→ DIG$_{2}$
0ECE$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼	0E5E$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼	→ DIG$_{1}$
0ECF$_{16}$									0E5F$_{16}$									→ DIG$_{0}$
0ED0$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	0E60$_{16}$	FLD$_{7}$	FLD$_{6}$	FLD$_{5}$	FLD$_{4}$	FLD$_{3}$	FLD$_{2}$	FLD$_{1}$	FLD$_{0}$	→ DIG$_{15}$
0ED1$_{16}$									0E61$_{16}$									→ DIG$_{14}$
0ED2$_{16}$									0E62$_{16}$									→ DIG$_{13}$
0ED3$_{16}$									0E63$_{16}$									→ DIG$_{12}$
0ED4$_{16}$									0E64$_{16}$									→ DIG$_{11}$
0ED5$_{16}$									0E65$_{16}$									→ DIG$_{10}$
0ED6$_{16}$									0E66$_{16}$									→ DIG$_{9}$
0ED7$_{16}$									0E67$_{16}$									→ DIG$_{8}$
0ED8$_{16}$									0E68$_{16}$									→ DIG$_{7}$
0ED9$_{16}$									0E69$_{16}$									→ DIG$_{6}$
0EDA$_{16}$									0E6A$_{16}$									→ DIG$_{5}$
0EDB$_{16}$									0E6B$_{16}$									→ DIG$_{4}$
0EDC$_{16}$									0E6C$_{16}$									→ DIG$_{3}$
0EDD$_{16}$									0E6D$_{16}$									→ DIG$_{2}$
0EDE$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼	0E6E$_{16}$	▼	▼	▼	▼	▼	▼	▼	▼	→ DIG$_{1}$
0EDF$_{16}$									0E6F$_{16}$									→ DIG$_{0}$

(Bit 7 of 0E90$_{16}$–0E9F$_{16}$ and 0E20$_{16}$–0E2F$_{16}$, "1"): CLK signal set area to M35501FP

(Bits 6–4 of 0E90$_{16}$–0E9F$_{16}$ and 0E20$_{16}$–0E2F$_{16}$): Unused

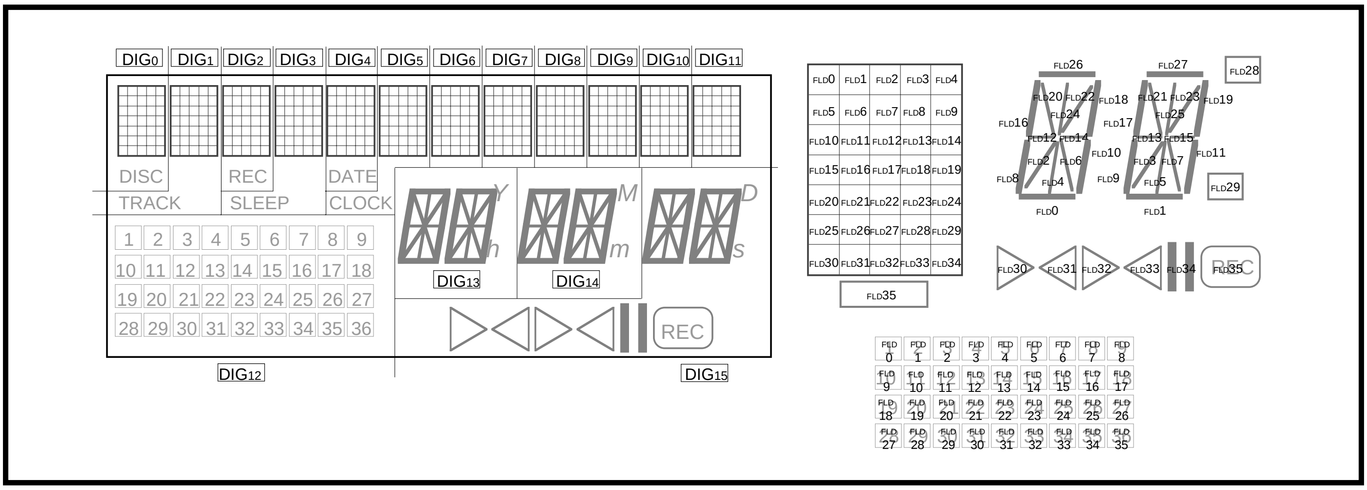


Fig. 2.4.36 FLD digit allocation example

Control procedure:

Figure 2.4.37 shows the control procedure.

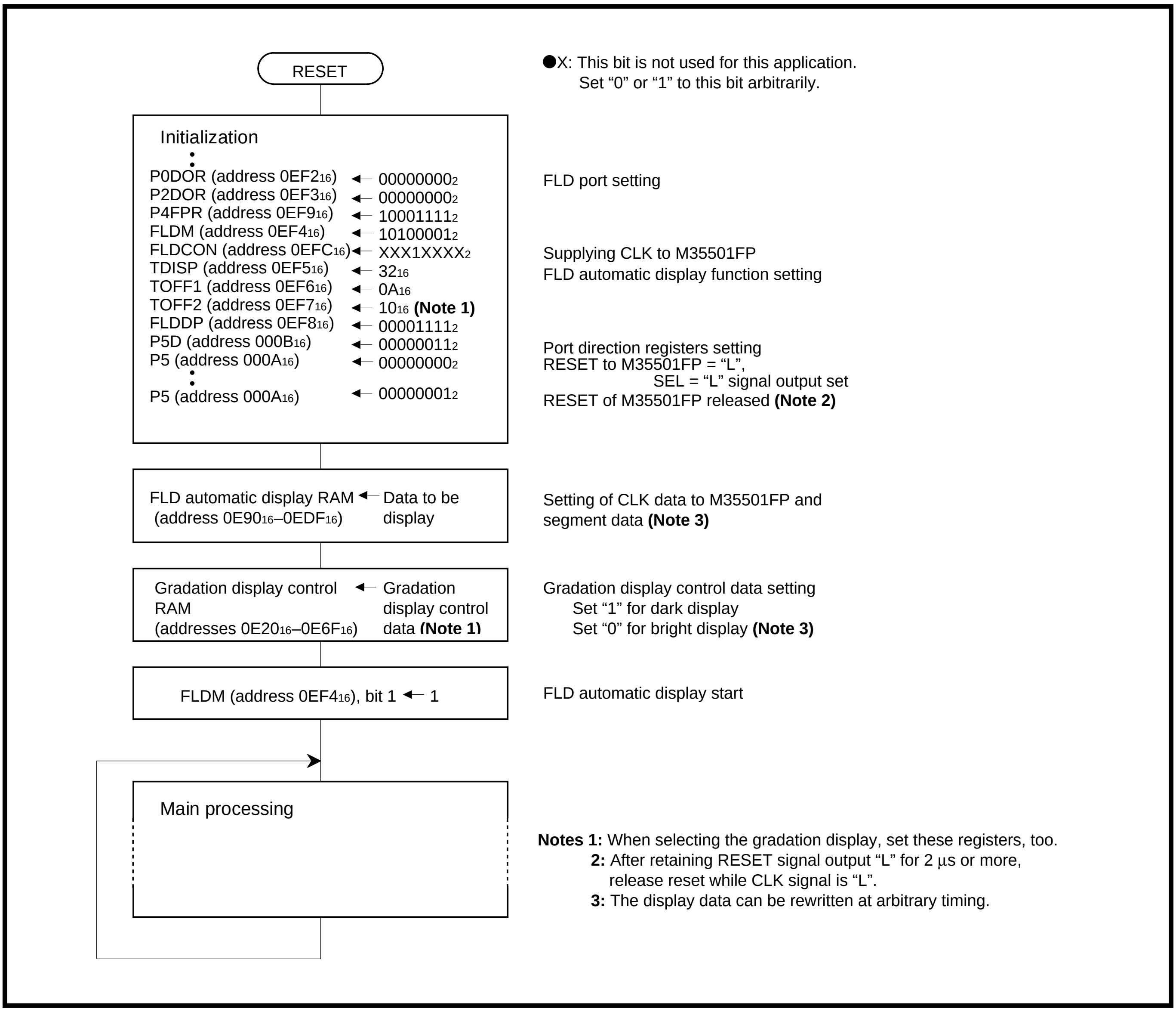


Fig. 2.4.37 Control procedure

(5) Display by combination with digit expander (M35501FP*) (example considering column discrepancy prevention)

* For M35501FP, refer to section **“3.9 M35501FP”**.

Outline: In the case of (4), which is displayed by using the digit expander (M35501FP), if a noise enters signals between 38B7 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.

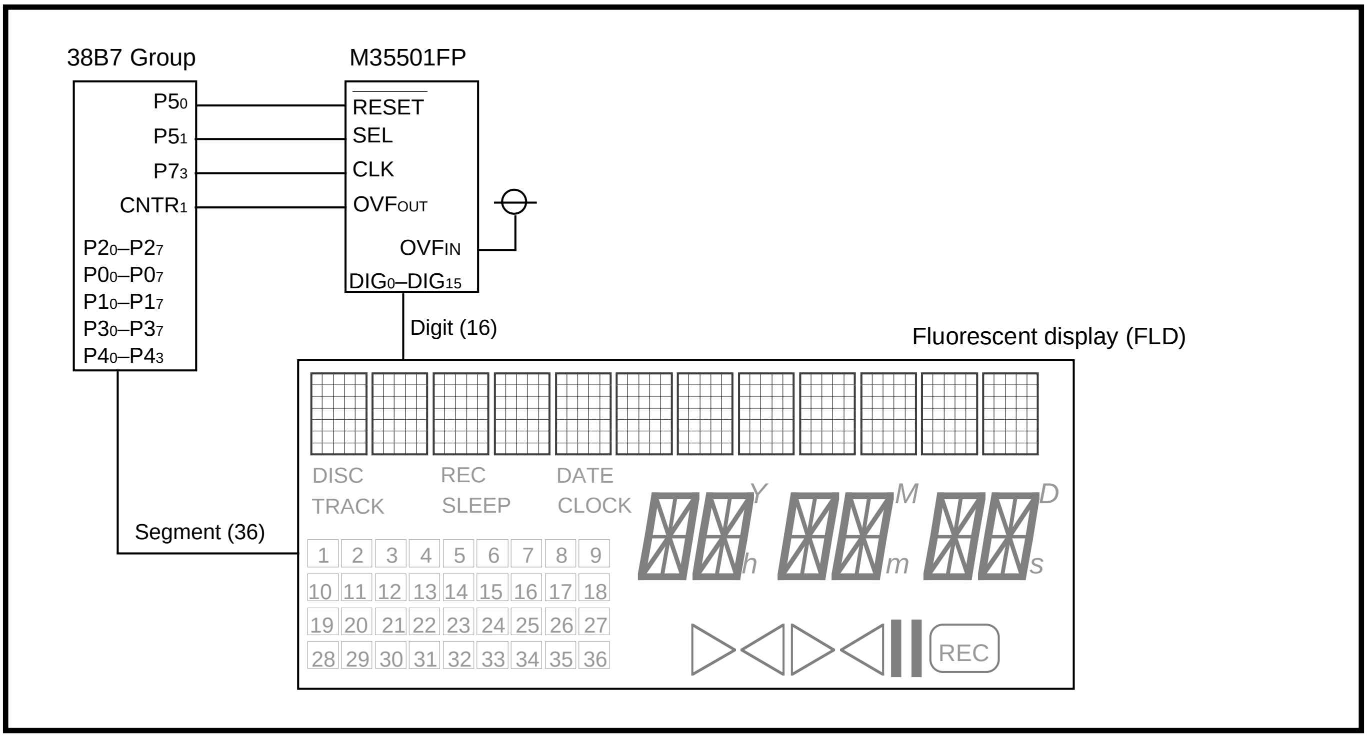


Fig. 2.4.38 Connection diagram

Specifications: •Use of M35501FP (M35501: 16 digits, 38B7 Group: 36 segments)
Ports $P5_0$ and $P5_1$ of 38B7 Group supply signal to the $\overline{\text{RESET}}$ and SEL pins of M35501FP respectively.
The $P7_3$ pin (dimmer output pin) supply signals to the CLK pin of M35501FP.
•Use of FLD automatic display mode of 38B7 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 38B7 Group
Input signal to $CNTR_1$ pin is counted as a count source by timer 4 of 38B7 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 38B7 Group and M35501FP, and Figure 2.4.40 shows the timing chart (at incorrect state) of 38B7 Group and M35501FP.

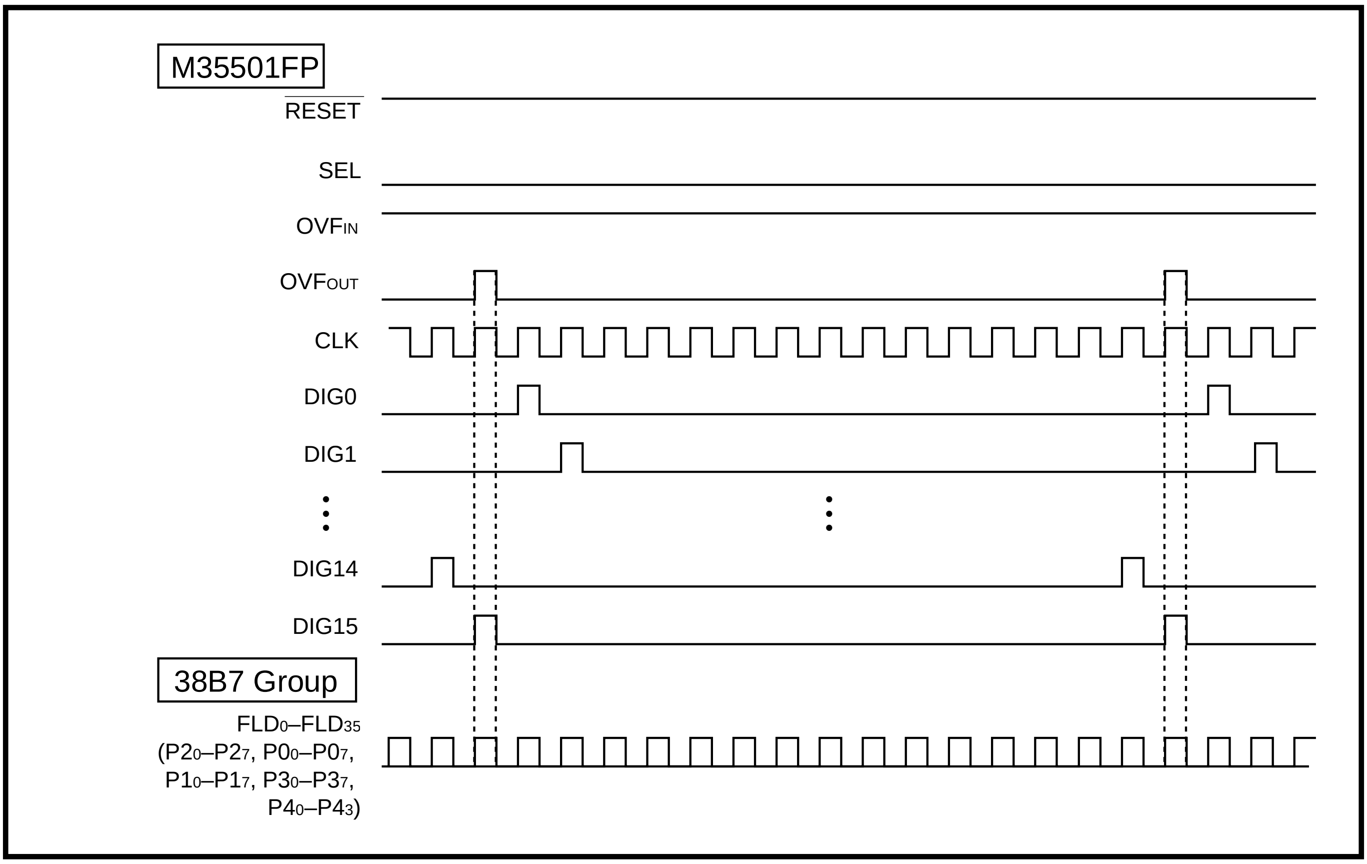


Fig. 2.4.39 Timing chart (at correct state) of 38B7 Group and M35501FP

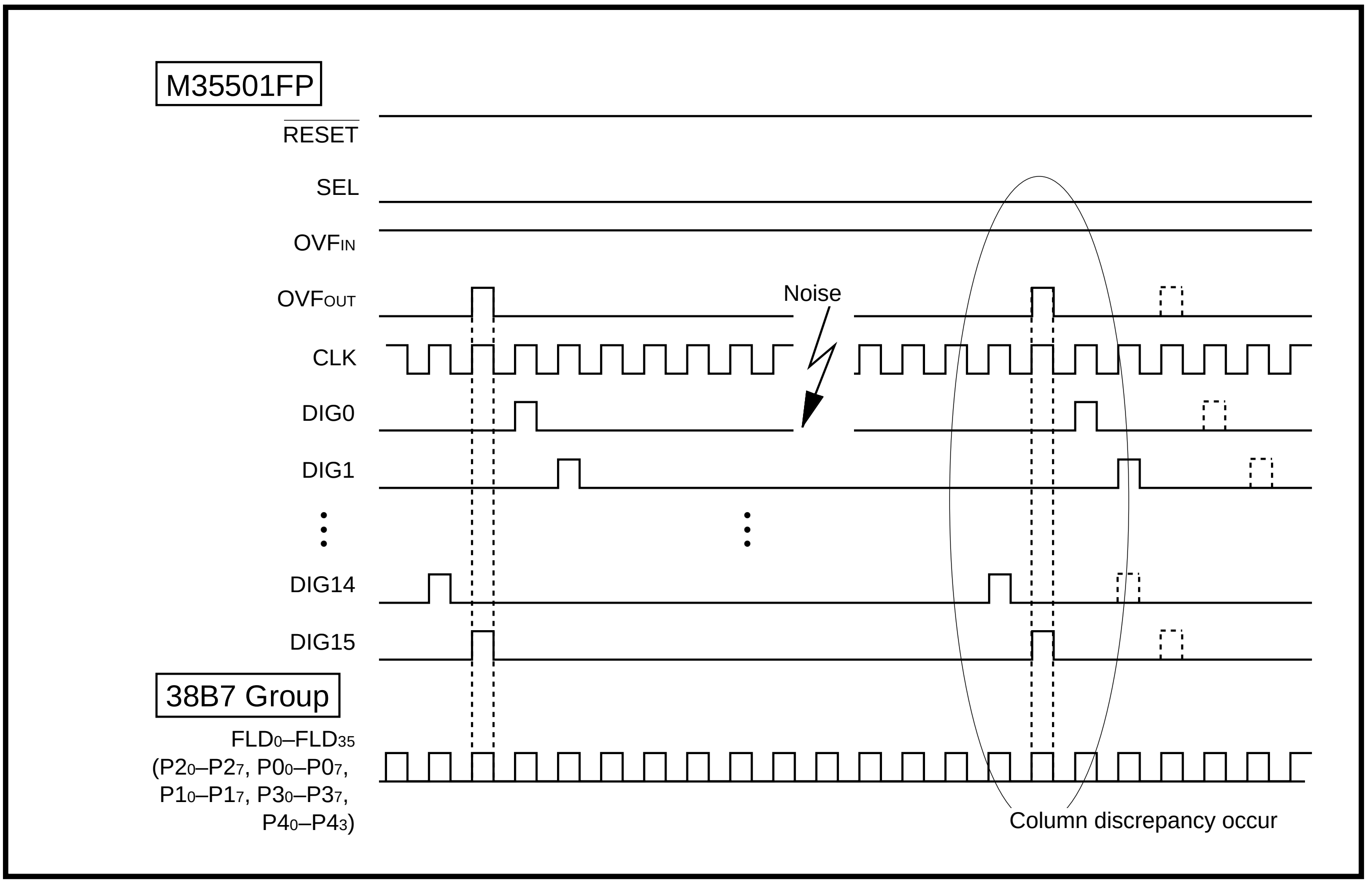


Fig. 2.4.40 Timing chart (at incorrect state) of 38B7 Group and M35501FP

Figure 2.4.41 shows the setting of relevant registers.

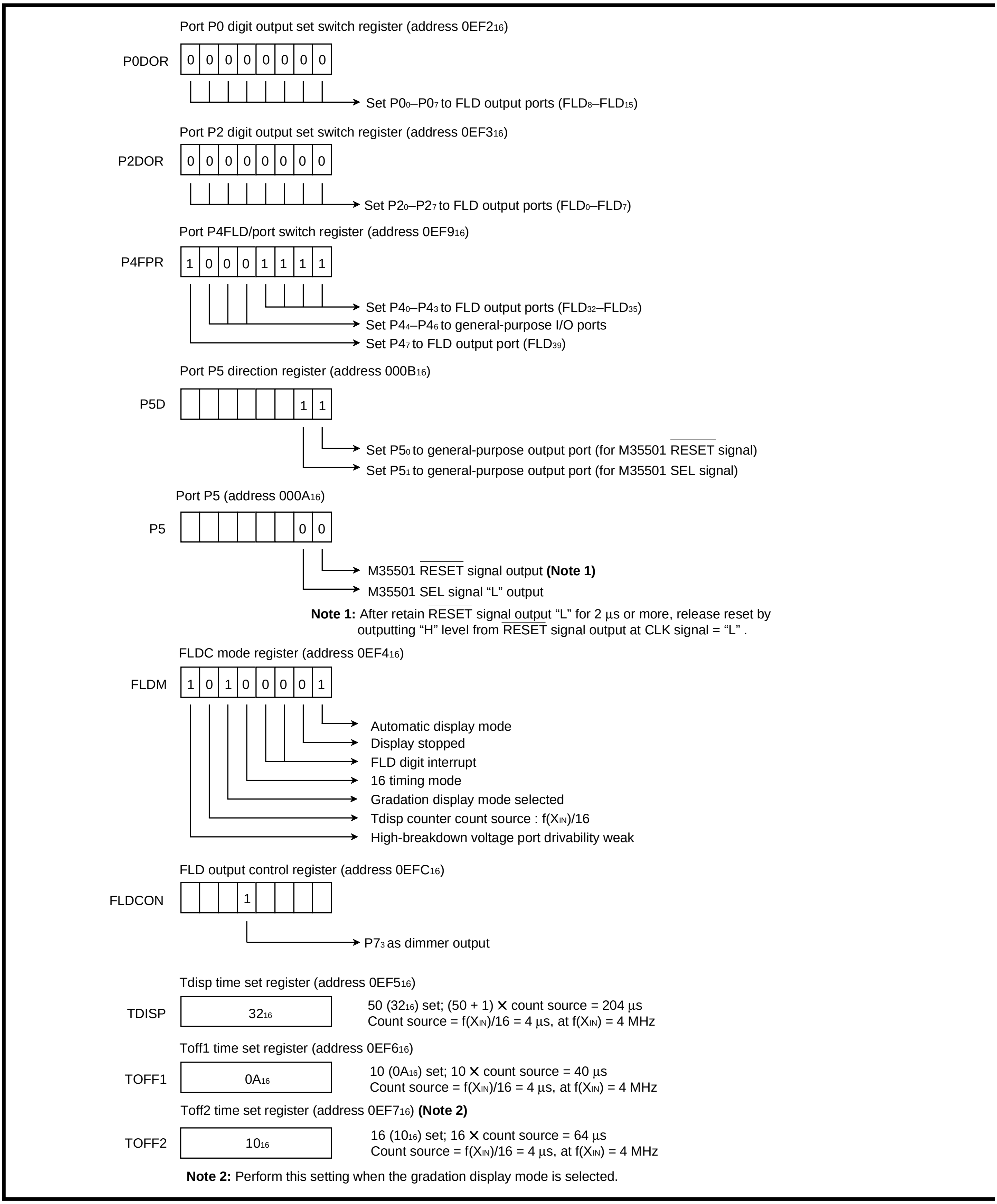
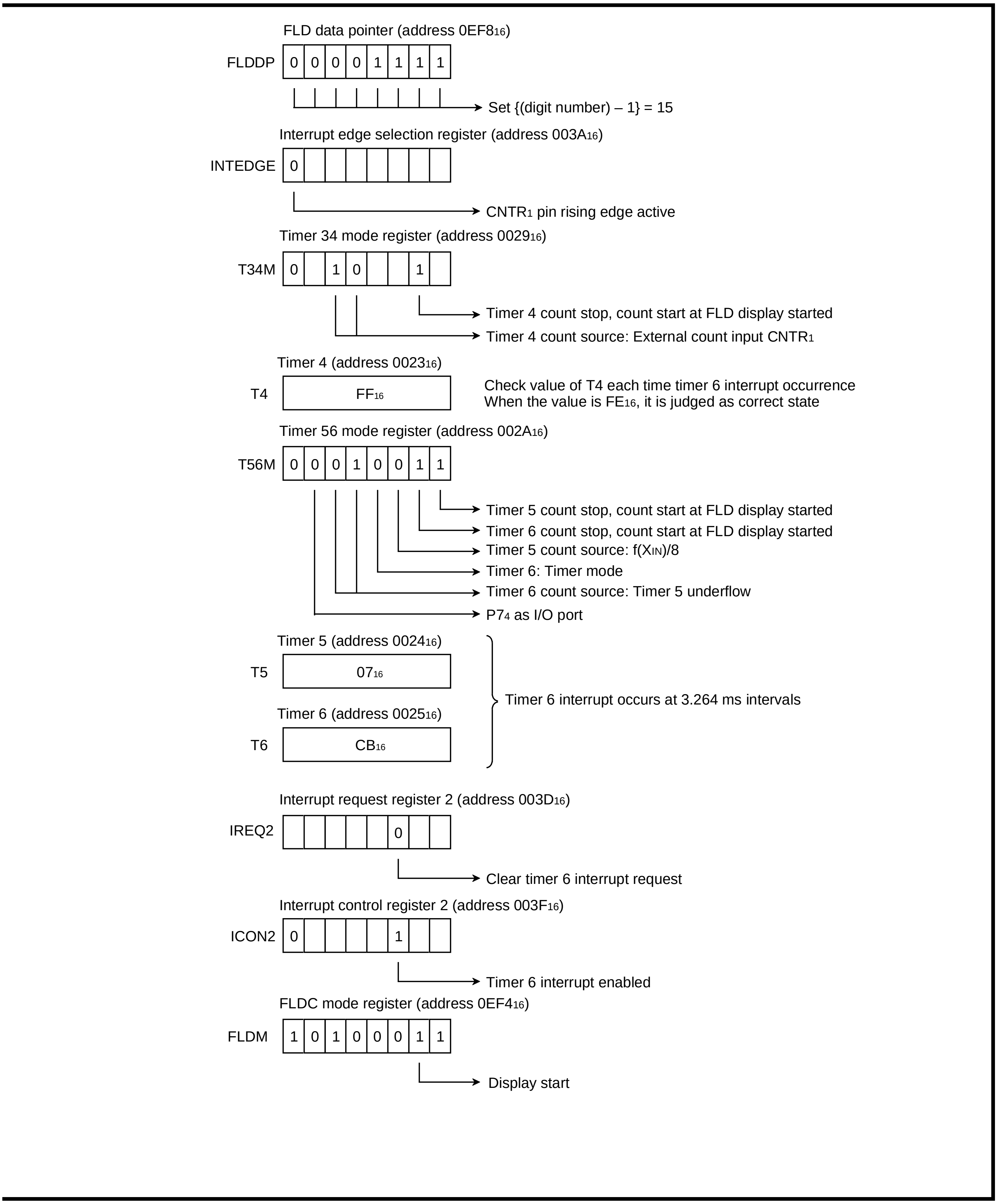


Fig. 2.4.41 Setting of relevant registers


FLD data pointer (address 0EF816)
FLDDP
0 0 0 0 1 1 1 1
Set {(digit number) – 1} = 15
Interrupt edge selection register (address 003A16)
INTEDGE
0
CNTR1 pin rising edge active
Timer 34 mode register (address 002916)
T34M
0 1 0 1
Timer 4 count stop, count start at FLD display started
Timer 4 count source: External count input CNTR1
Timer 4 (address 002316)
T4
FF16
Check value of T4 each time timer 6 interrupt occurrence
When the value is FE16, it is judged as correct state
Timer 56 mode register (address 002A16)
T56M
0 0 0 1 0 0 1 1
Timer 5 count stop, count start at FLD display started
Timer 6 count stop, count start at FLD display started
Timer 5 count source: f(XIN)/8
Timer 6: Timer mode
Timer 6 count source: Timer 5 underflow
P74 as I/O port
Timer 5 (address 002416)
T5
0716
Timer 6 (address 002516)
T6
CB16
Timer 6 interrupt occurs at 3.264 ms intervals
Interrupt request register 2 (address 003D16)
IREQ2
0
Clear timer 6 interrupt request
Interrupt control register 2 (address 003F16)
ICON2
0 1
Timer 6 interrupt enabled
FLDC mode register (address 0EF416)
FLDM
1 0 1 0 0 0 1 1
Display start

Control procedure:

Figure 2.4.42 shows the control procedure.

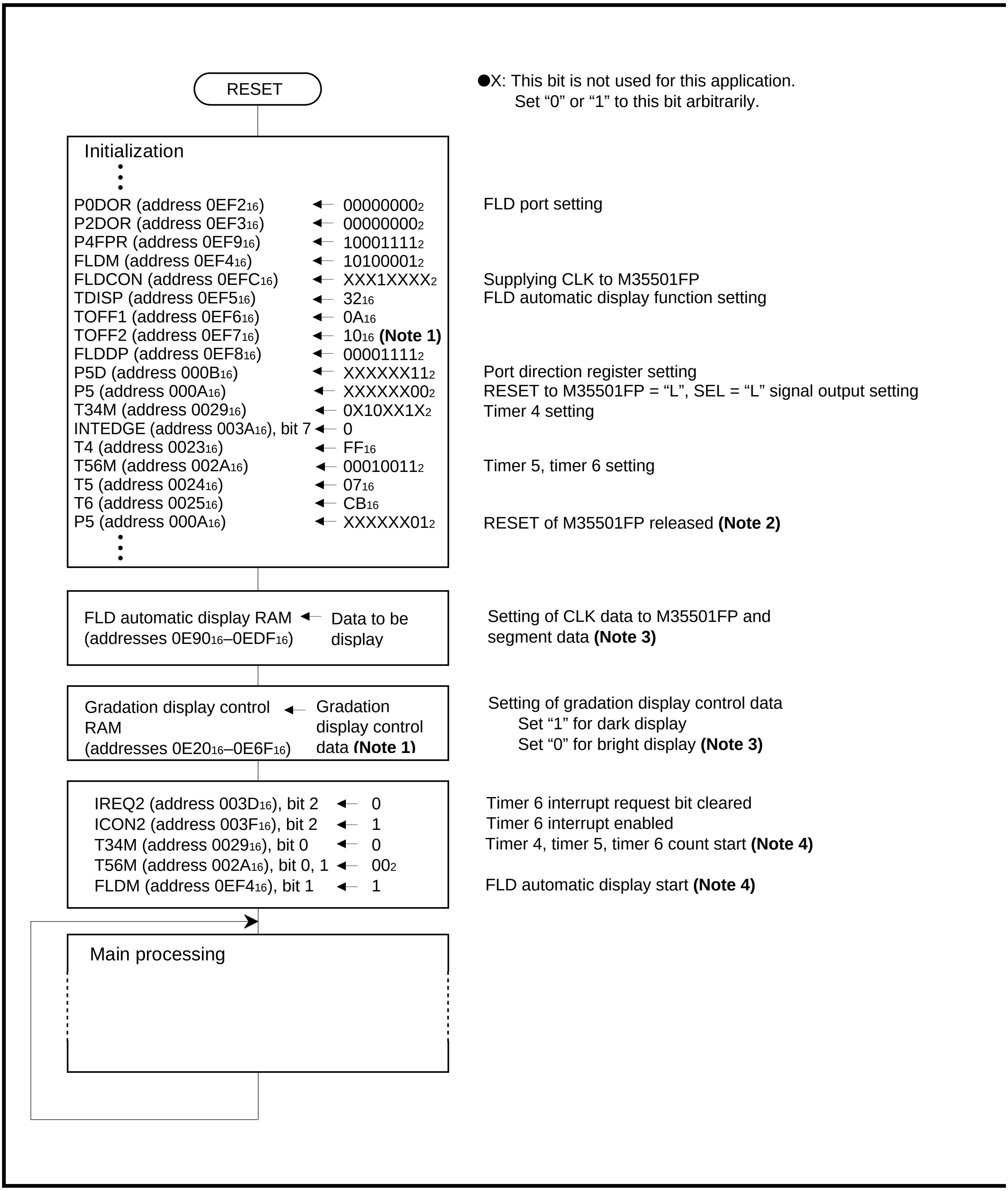
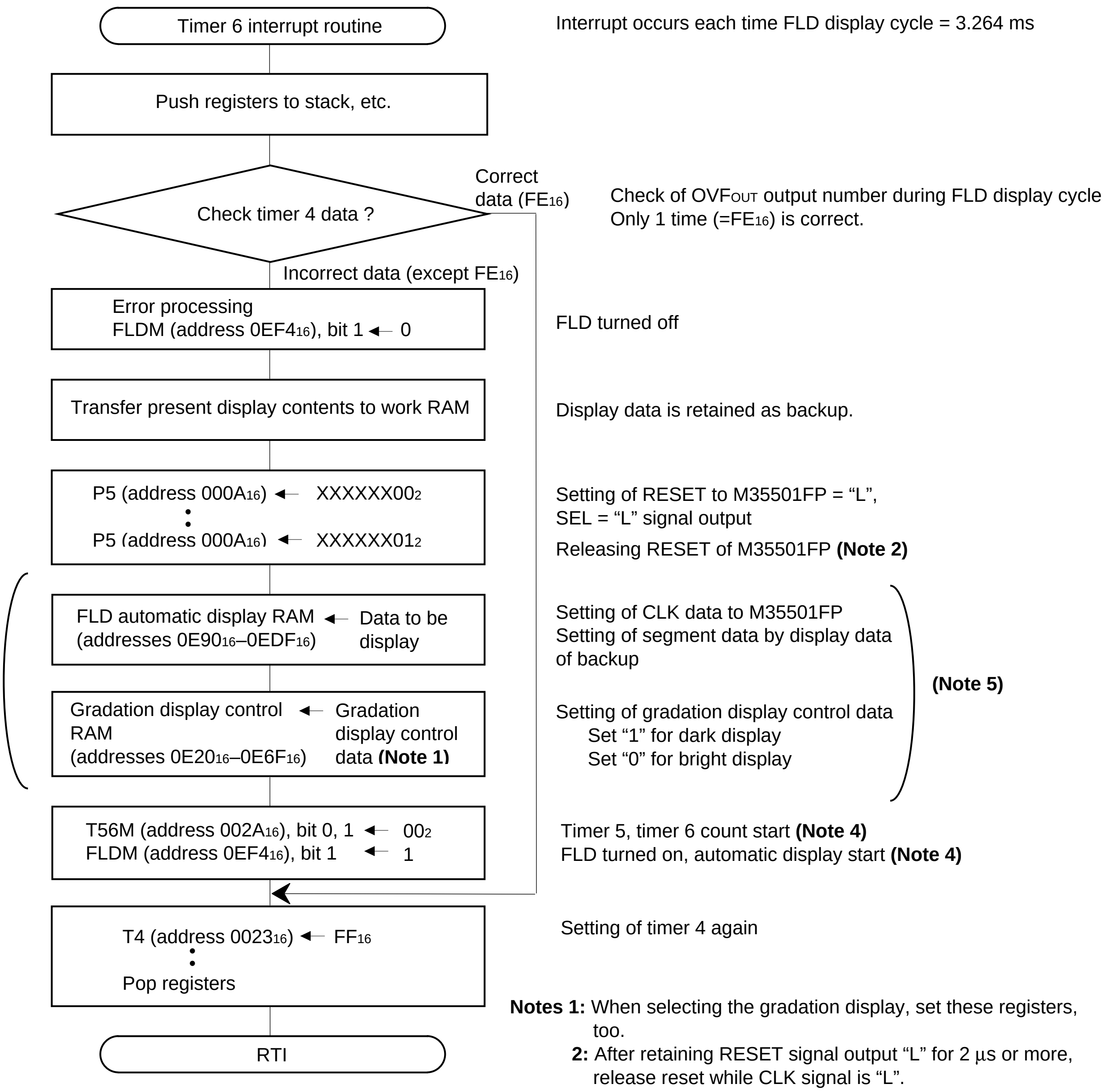


Fig. 2.4.42 Control procedure



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 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”).

2.5 A-D converter

This paragraph describes the setting method of A-D converter relevant registers, notes etc.

2.5.1 Memory assignment

Address	
0032_{16}	AD/DA control register (ADCON)
0033_{16}	A-D conversion register (low-order) (ADL)
0034_{16}	A-D conversion register (high-order) (ADH)
0039_{16}	Interrupt source switch register (IFR)
$003D_{16}$	Interrupt request register 2 (IREQ2)
$003E_{16}$	(Interrupt control register 1 (ICON1))
$003F_{16}$	Interrupt control register 2 (ICON2)

Fig. 2.5.1 Memory assignment of A-D converter relevant registers

2.5.2 Relevant registers

AD/DA control register

b7 b6 b5 b4 b3 b2 b1 b0

AD/DA 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: PA_0/AN_0 0 0 0 1: PA_1/AN_1 0 0 1 0: PA_2/AN_2 0 0 1 1: PA_3/AN_3 0 1 0 0: PA_4/AN_4 0 1 0 1: PA_5/AN_5 0 1 1 0: PA_6/AN_6 0 1 1 1: PA_7/AN_7 1 0 0 0: $P9_0/S_{IN3}/AN_8$ 1 0 0 1: $P9_1/S_{OUT3}/AN_9$ 1 0 1 0: $P9_2/\overline{S_{CLK3}}/AN_{10}$ 1 0 1 1: $P9_3/\overline{S_{RDY3}}/AN_{11}$ 1 1 0 0: $P9_4/RTP_1/AN_{12}$ 1 1 0 1: $P9_5/RTP_0/AN_{13}$ 1 1 1 0: $P9_6/PWM_0/AN_{14}$ 1 1 1 1: $P9_7/B_{UZ02}/AN_{15}$	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 this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×
6	DA output enable bit	0: DA output disabled 1: DA output enabled	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. 2.5.2 Structure of AD/DA control register

A-D conversion register (low-order)

b7 b6 b5 b4 b3 b2 b1 b0

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)

b7 b6 b5 b4 b3 b2 b1 b0

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 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	INT3/serial I/O2 transmit interrupt switch bit	0: INT3 intrrupt 1: Serial I/O2 transmit interrupt	0	O	O
1	INT4/A-D conversion interrupt switch bit	0: INT4 interrupt 1: A-D conversion intrerrupt	0	O	O
2	INT1/serial I/O3 interrupt switch bit	0: INT1 intrrupt 1: Serial I/O3 interrupt	0	O	O
3	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	O	×
4			0	O	×
5			0	O	×
6			0	O	×
7			0	O	×

Fig. 2.5.5 Structure of Interrupt source switch register

Interrupt request register 2

b7 b6 b5 b4 b3 b2 b1 b0

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

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

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

Fig. 2.5.6 Structure of Interrupt request register 2

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	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
1	Timer 5 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
2	Timer 6 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
3	Serial I/O2 receive interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
4	INT_3/Serial I/O2 transmit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
5	INT_4 interrupt enable bit A-D converter interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
6	FLD blanking interrupt enable bit FLD digit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
7	Fix “0” to this bit.		0	O	O

Fig. 2.5.7 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 2.5.8 shows a connection diagram, and Figure 2.5.9 shows the setting of relevant registers.

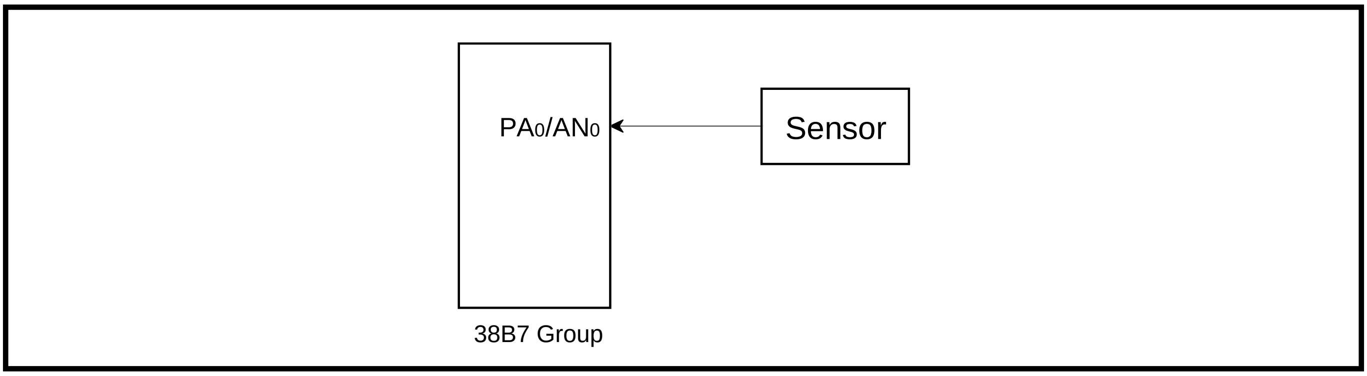


Fig. 2.5.8 Connection diagram

Specifications: •Conversion of analog input voltage input from sensor to digital values
•Use of PA_0/AN_0 pin as analog input pin

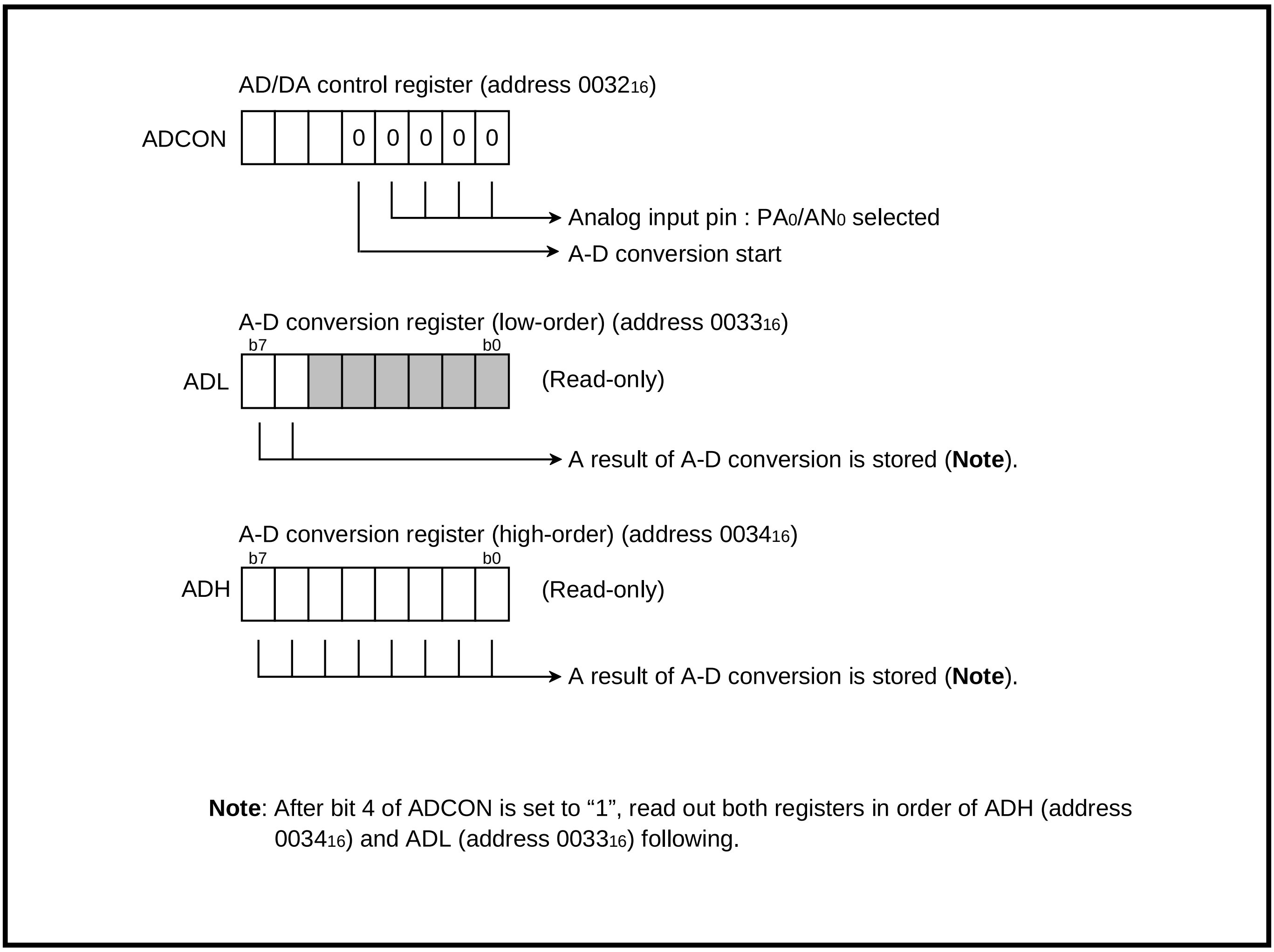


Fig. 2.5.9 Setting of relevant registers

Control procedure: A-D converter is started by performing register setting shown Figure 2.5.9. Figure 2.5.10 shows the control procedure.

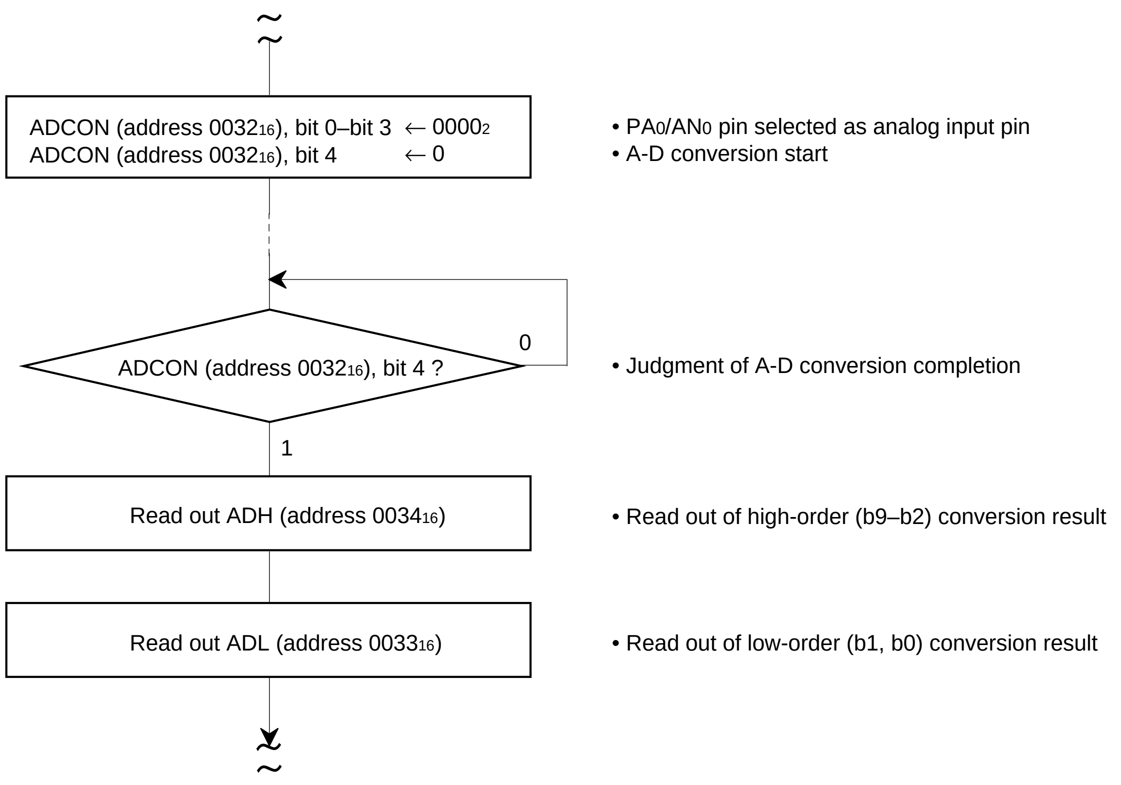


Fig. 2.5.10 Control procedure

2.5.4 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.

(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

• Do not execute the **STP** instruction and **WIT** instruction

2.6 D-A converter

This paragraph describes the setting method of D-A converter relevant registers, notes etc.

2.6.1 Memory assignment

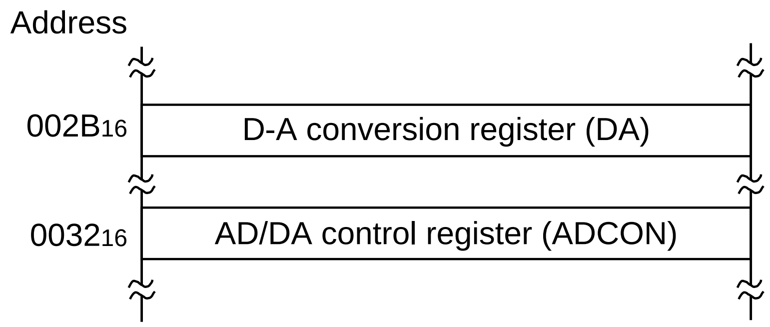
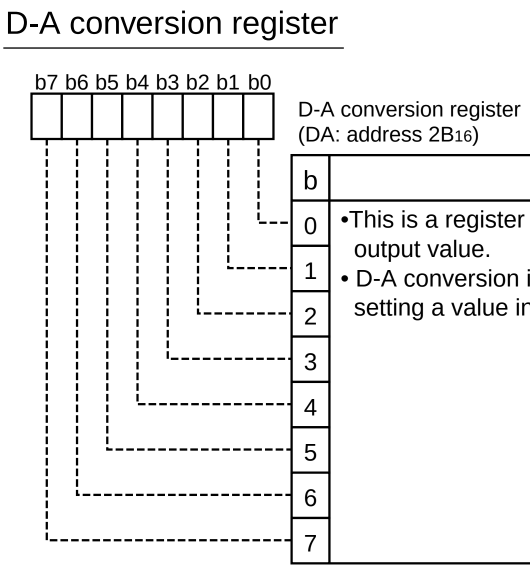


Fig. 2.6.1 Memory assignment of D-A converter relevant registers

2.6.2 Relevant registers



b	Functions	At reset	R	W
0	•This is a register for set of D-A conversion output value. • D-A conversion is performed automatically by setting a value in this register.	0	O	O
1		0	O	O
2		0	O	O
3		0	O	O
4		0	O	O
5		0	O	O
6		0	O	O
7		0	O	O

Fig. 2.6.2 Structure of D-A conversion register

AD/DA control register

b7 b6 b5 b4 b3 b2 b1 b0

AD/DA 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: PA$_0$/AN$_0$ 0 0 0 1: PA$_1$/AN$_1$ 0 0 1 0: PA$_2$/AN$_2$ 0 0 1 1: PA$_3$/AN$_3$ 0 1 0 0: PA$_4$/AN$_4$ 0 1 0 1: PA$_5$/AN$_5$ 0 1 1 0: PA$_6$/AN$_6$ 0 1 1 1: PA$_7$/AN$_7$ 1 0 0 0: P9$_0$/S$_{IN3}$/AN$_8$ 1 0 0 1: P9$_1$/S$_{OUT3}$/AN$_9$ 1 0 1 0: P9$_2$/$\overline{S_{CLK3}}$/AN$_{10}$ 1 0 1 1: P9$_3$/$\overline{S_{RDY3}}$/AN$_{11}$ 1 1 0 0: P9$_4$/RTP$_1$/AN$_{12}$ 1 1 0 1: P9$_5$/RTP$_0$/AN$_{13}$ 1 1 1 0: P9$_6$/PWM$_0$/AN$_{14}$ 1 1 1 1: P9$_7$/B$_{UZ02}$/AN$_{15}$	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 this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×
6	DA output enable bit	0: DA output disabled 1: DA output enabled	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. 2.6.3 Structure of AD/DA control register

2.6.3 D-A converter application examples

Outline: Digital value is converted to the analog output voltage.

Figure 2.6.4 shows a connection diagram, and Figure 2.6.5 shows the setting of relevant registers.

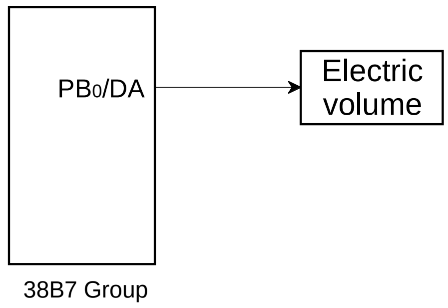


Fig. 2.6.4 Connection diagram

Specifications: •Conversion of digital value to analog output voltage.

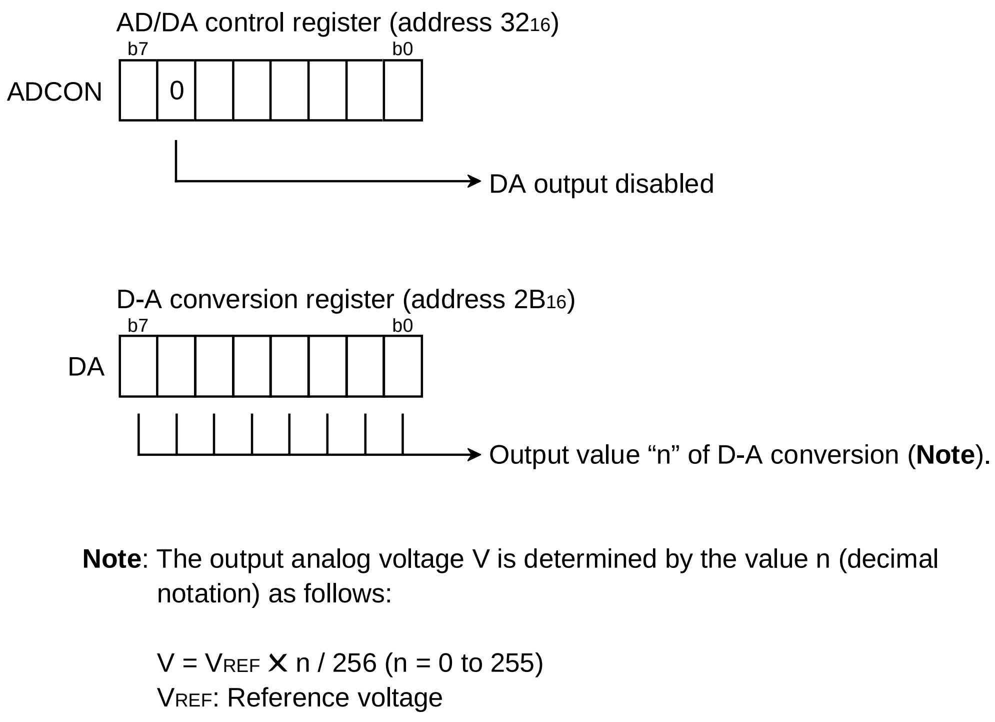


Fig. 2.6.5 Setting of relevant registers

Control procedure: D-A converter is started by performing register setting shown Figure 2.6.5. Figure 2.6.6 shows the control procedure.

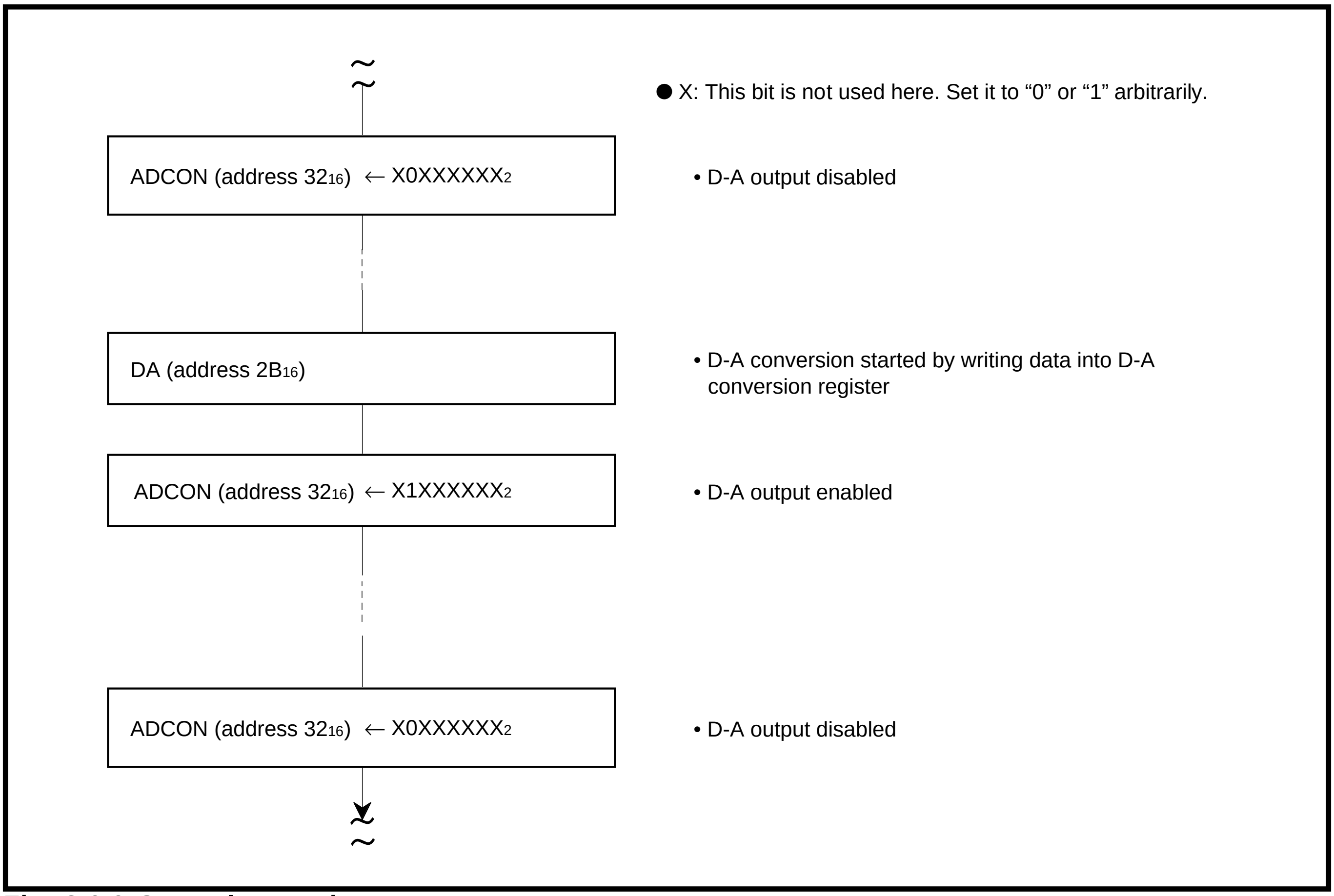


Fig. 2.6.6 Control procedure

2.6.4 Notes on D-A converter

(1) PB_0/DA pin state at reset

The PB_0/DA pin becomes a high-impedance state at reset.

(2) Connection with low-impedance load

If connecting a D-A output with a load having a low impedance, use an external buffer. It is because the D-A converter circuit does not include a buffer.

(3) Usable voltage

Vcc must be 3.0 V or more when using the D-A converter.

2.7 PWM

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

2.7.1 Memory assignment

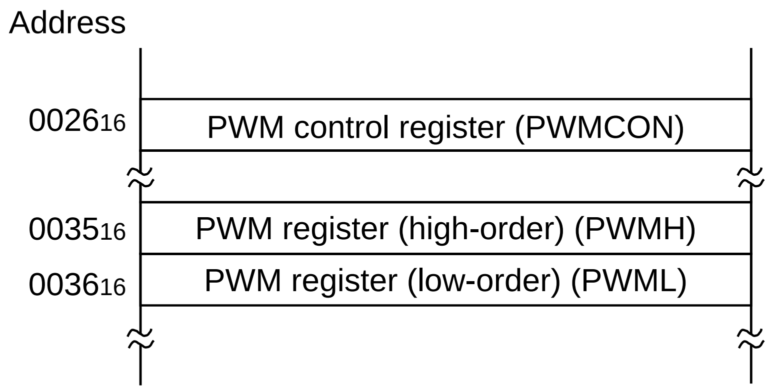
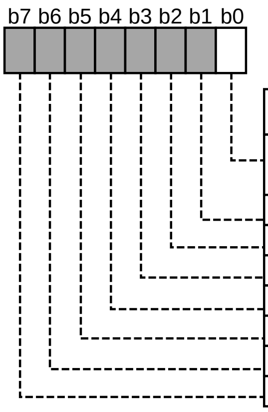


Fig. 2.7.1 Memory assignment of PWM relevant registers

2.7.2 Relevant registers

PWM control register



PWM control register
(PWMCON: address 26_{16})

b	Name	Functions	At reset	R	W
0	$P9_6$/PWM_0 output selection bit	0: I/O port 1: PWM_0 output	0	O	O
1	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	O	×
2			0	O	×
3			0	O	×
4			0	O	×
5			0	O	×
6			0	O	×
7			0	O	×

Fig. 2.7.2 Structure of PWM control register

PWM register (high-order)

b7 b6 b5 b4 b3 b2 b1 b0

PWM register (high-order)
(PWMH: address 35_{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	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. 2.7.3 Structure of PWM register (high-order)

PWM register (low-order)

b7 b6 b5 b4 b3 b2 b1 b0

PWM register (low-order)
(PWML: address 36_{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	O	O
1		Undefined	O	O
2		Undefined	O	O
3		Undefined	O	O
4		Undefined	O	O
5		Undefined	O	O
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	O	X
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	O	X

Fig. 2.7.4 Structure of PWM register (low-order)

2.7.3 PWM application example

(1) Control of VS tuner

Figure 2.7.5 shows a connection diagram, and Figure 2.7.6 shows the setting of relevant registers.

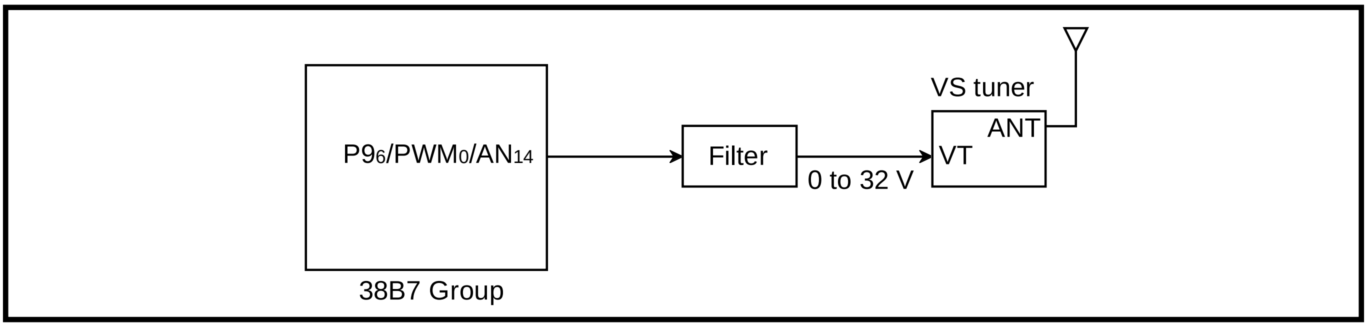


Fig. 2.7.5 Connection diagram

Outline: • Control of VS tuner by using the 14-bit resolution PWM_0 output function
• $f(X_{IN})$ = 4 MHz

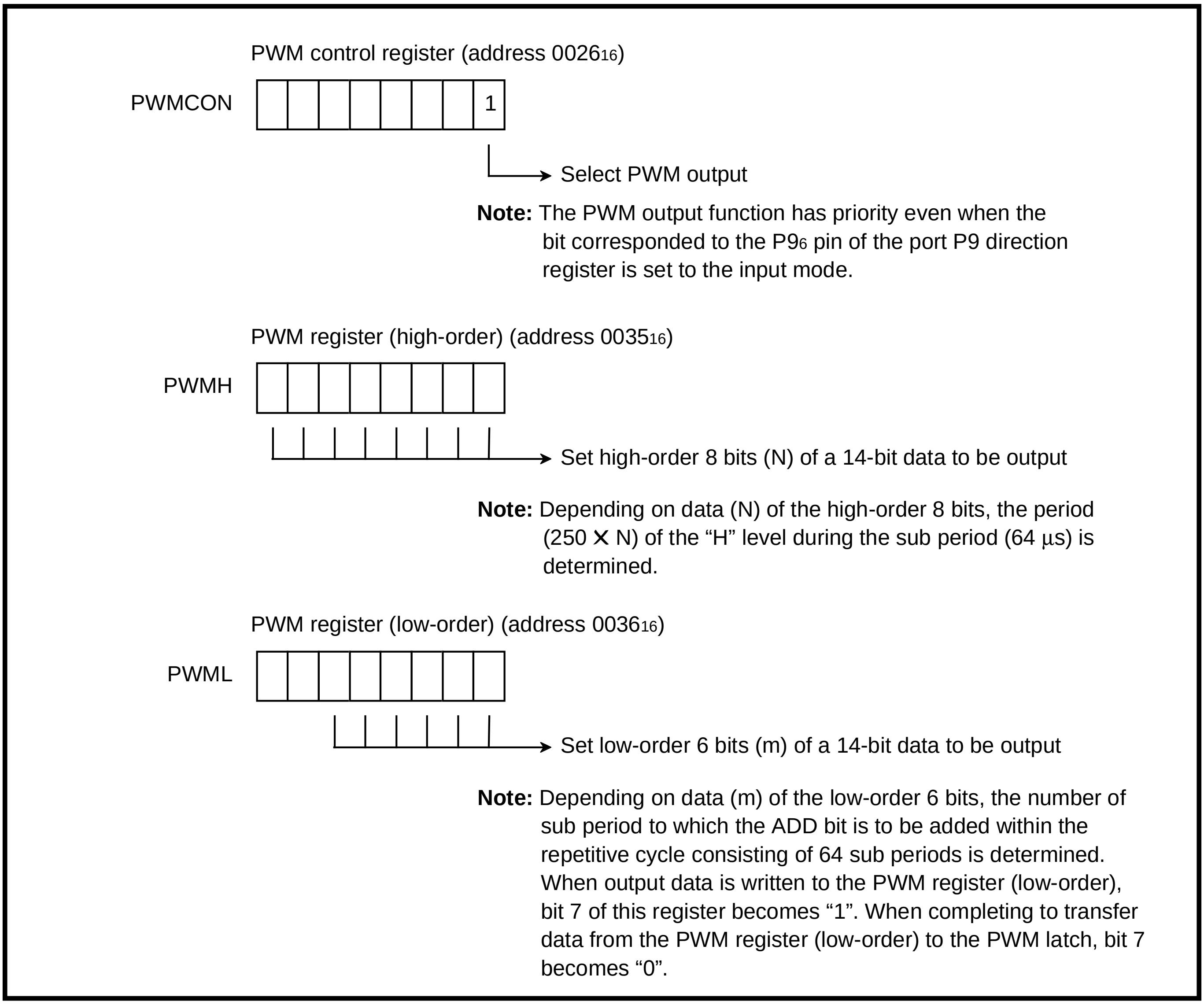


Fig. 2.7.6 Setting of relevant registers

Control procedure: PWM waveform is output to the external by setting relevant registers shown in Figure 2.7.6. This PWM_0 output is integrated through the low pass filter and converted into DC signals for control of the VS tuner.
Figure 2.7.7 shows the control procedure.

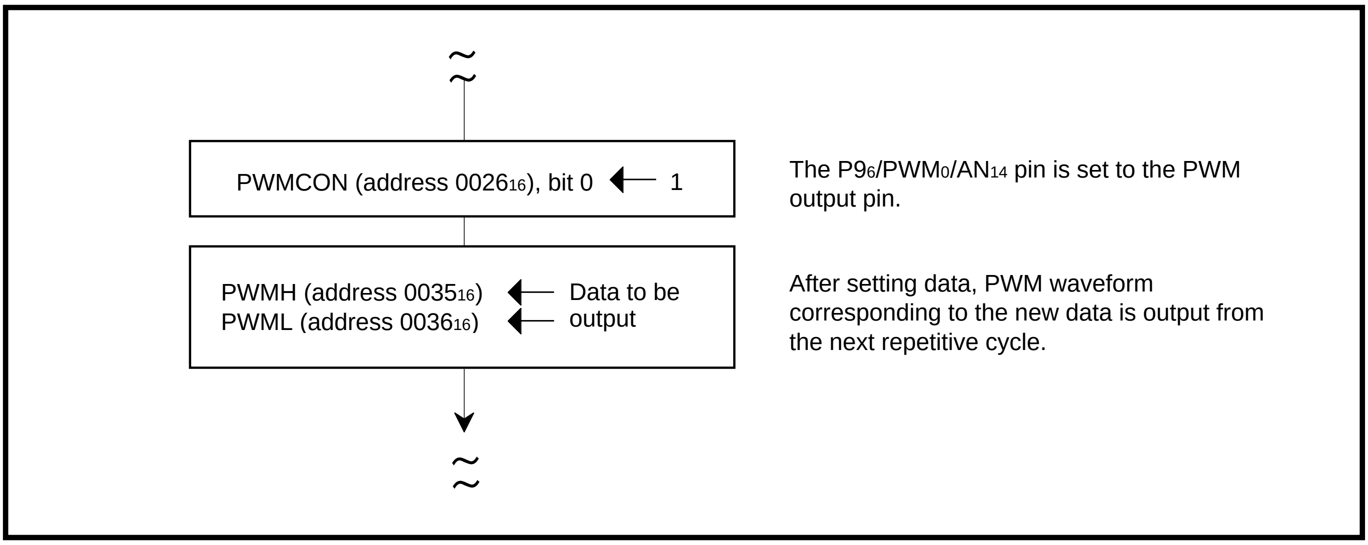


Fig. 2.7.7 Control procedure

2.7.4 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.

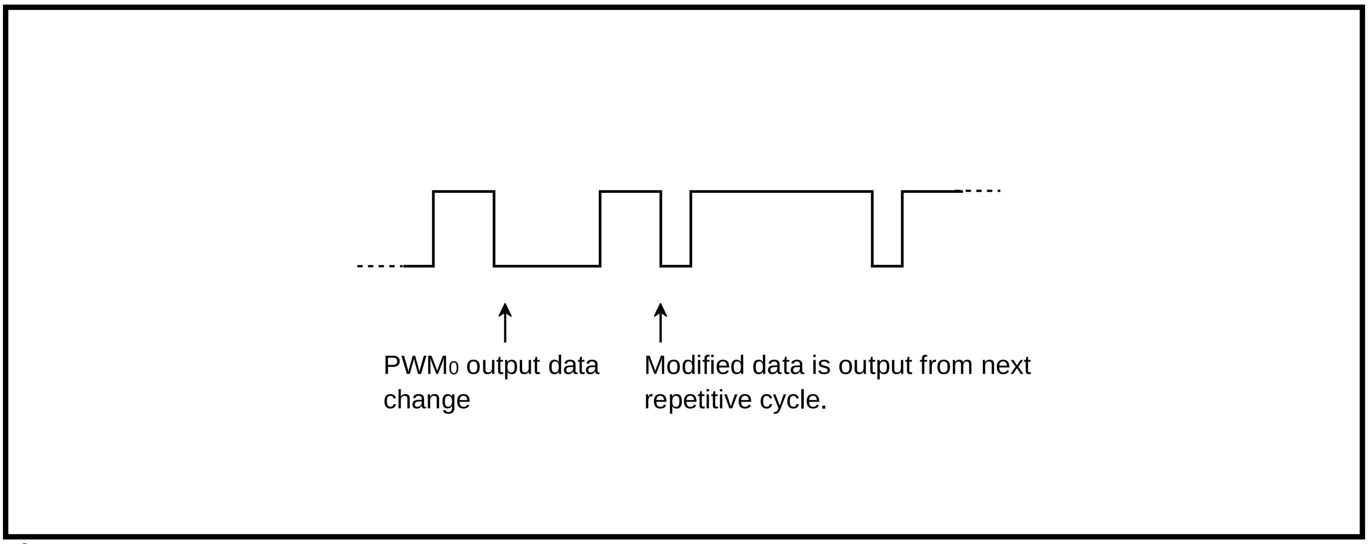


Fig. 2.7.8 PWM_0 output

2.8 Interrupt interval determination function

This paragraph describes the setting method of interrupt interval determination function relevant registers, notes etc.

2.8.1 Memory assignment

Address	
0030_{16}	Interrupt interval determination register (IID)
0031_{16}	Interrupt interval determination control register (IIDCON)
$003A_{16}$	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)

Fig. 2.8.1 Memory assignment of interrupt interval determination function relevant registers

2.8.2 Relevant registers

Interrupt interval determination register

b7 b6 b5 b4 b3 b2 b1 b0

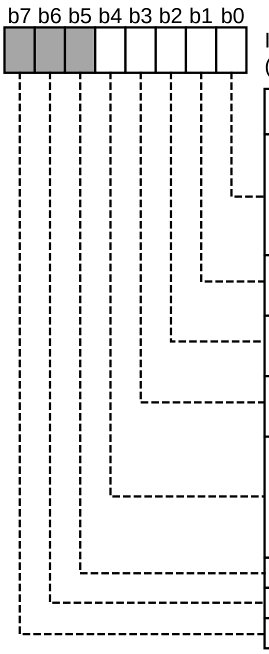
Interrupt interval determination register (IID: address 30_{16})

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	0	○	×
1		0	○	×
2		0	○	×
3		0	○	×
4		0	○	×
5		0	○	×
6		0	○	×
7		0	○	×

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. 2.8.2 Structure of Interrupt interval determination register

Interrupt interval determination control register



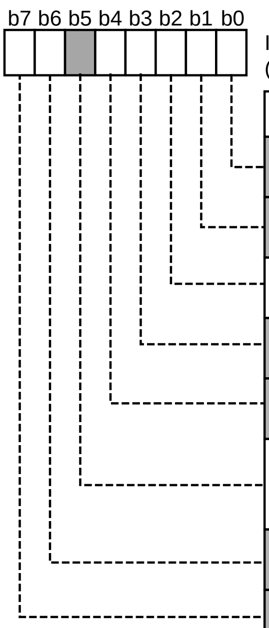
Interrupt interval determination control register
(IIDCON: address 31_{16})

b	Name	Functions	At reset	R	W
0	Interrupt interval determination circuit operating selection bit	0: Stopped 1: Operating	0	O	O
1	Counter sampling clock selection bit	0: f(X_{IN})/128 or f(X_{CIN}) 1: f(X_{IN})/256 or f(X_{CIN})/2	0	O	O
2	Noise filter sampling clock selection bits (INT_2)	b3 b2 0 0: Filter is not used. 0 1: f(X_{IN})/32 or f(X_{CIN}) 1 0: f(X_{IN})/64 or f(X_{CIN})/2 1 1: f(X_{IN})/128 or f(X_{CIN})/4	0	O	O
3			0	O	O
4	One-sided/both-sided edge detection selection bit	0: One-sided edge detection 1: Both-sided edge detection **(Note)**	0	O	O
5	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	O	×
6			0	O	×
7			0	O	×

Note: When the noise filter sampling clock selection bits (bits 2, 3) is “00”, the both-sided edge detection function cannot be used.

Fig. 2.8.3 Structure of Interrupt interval determination control register

Interrupt edge selection register

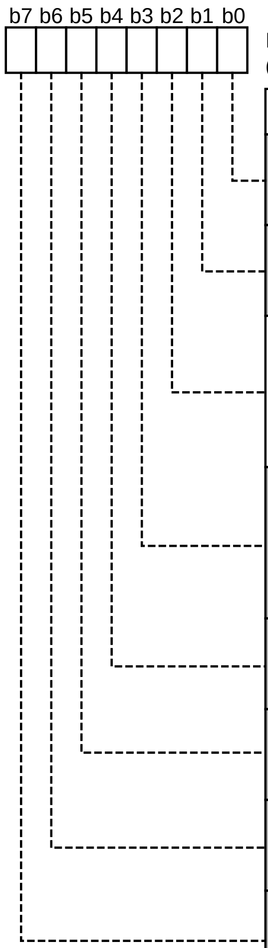


Interrupt edge selection register
(INTEDGE : address $3A_{16}$)

b	Name	Functions	At reset	R	W
0	INT_0 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	O	O
1	INT_1 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	O	O
2	INT_2 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	O	O
3	INT_3 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	O	O
4	INT_4 interrupt edge selection bit	0 : Falling edge active 1 : Rising edge active	0	O	O
5	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	O	×
6	$CNTR_0$ pin edge switch bit	0 : Rising edge count 1 : Falling edge count	0	O	O
7	$CNTR_1$ pin edge switch bit	0 : Rising edge count 1 : Falling edge count	0	O	O

Fig. 2.8.4 Structure of Interrupt edge selection 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/serial I/O3 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. 2.8.5 Structure of Interrupt request register 1

Interrupt control register 1

b7 b6 b5 b4 b3 b2 b1 b0

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

b	Name	Functions	At reset	R	W
0	INT_0 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
1	INT_1/serial I/O3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
2	INT_2 interrupt enable bit Remote controller /counter overflow interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
3	Serial I/O1 interrupt enable bit Serial I/O automatic transfer interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
4	Timer X interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
5	Timer 1 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
6	Timer 2 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O
7	Timer 3 interrupt enable bit	0 : Interrupt disabled 1 : Interrupt enabled	0	O	O

Fig. 2.8.6 Structure of Interrupt control register 1

2.8.3 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.

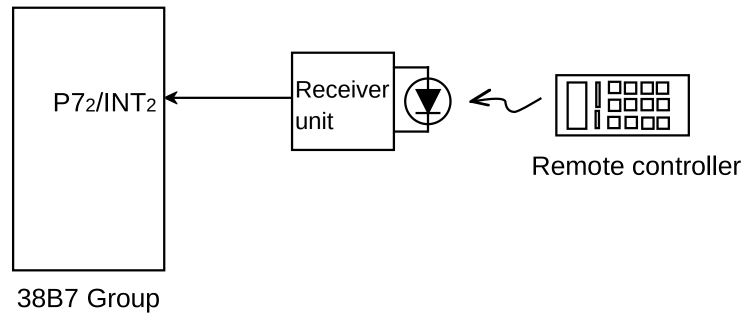


Fig. 2.8.7 Connection diagram

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 2.8.8 shows the function block diagram, and Figure 2.8.9 shows a timing chart of data determination.

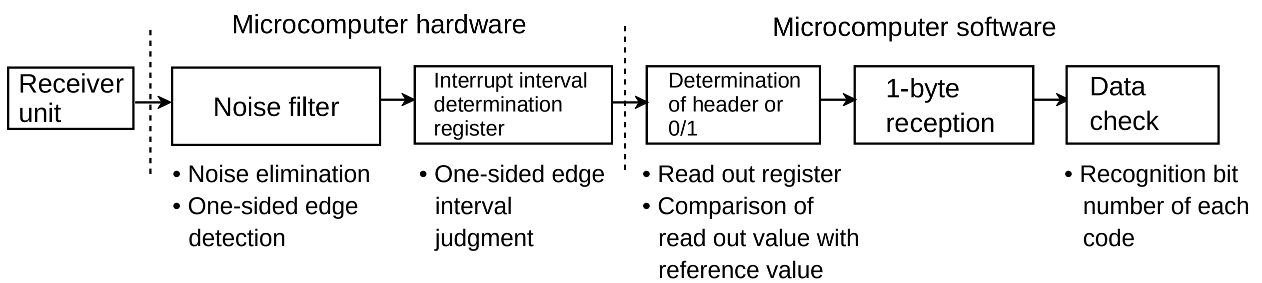


Fig. 2.8.8 Function block diagram

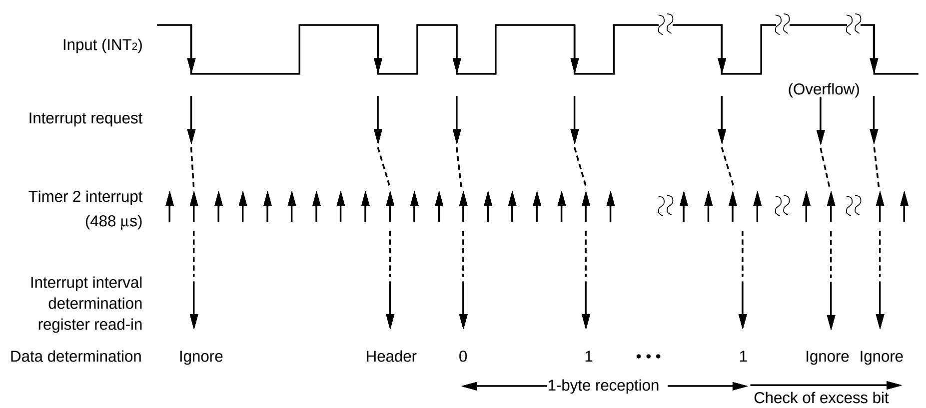


Fig. 2.8.9 Timing chart of data determination

Figure 2.8.10 shows the setting of relevant registers.

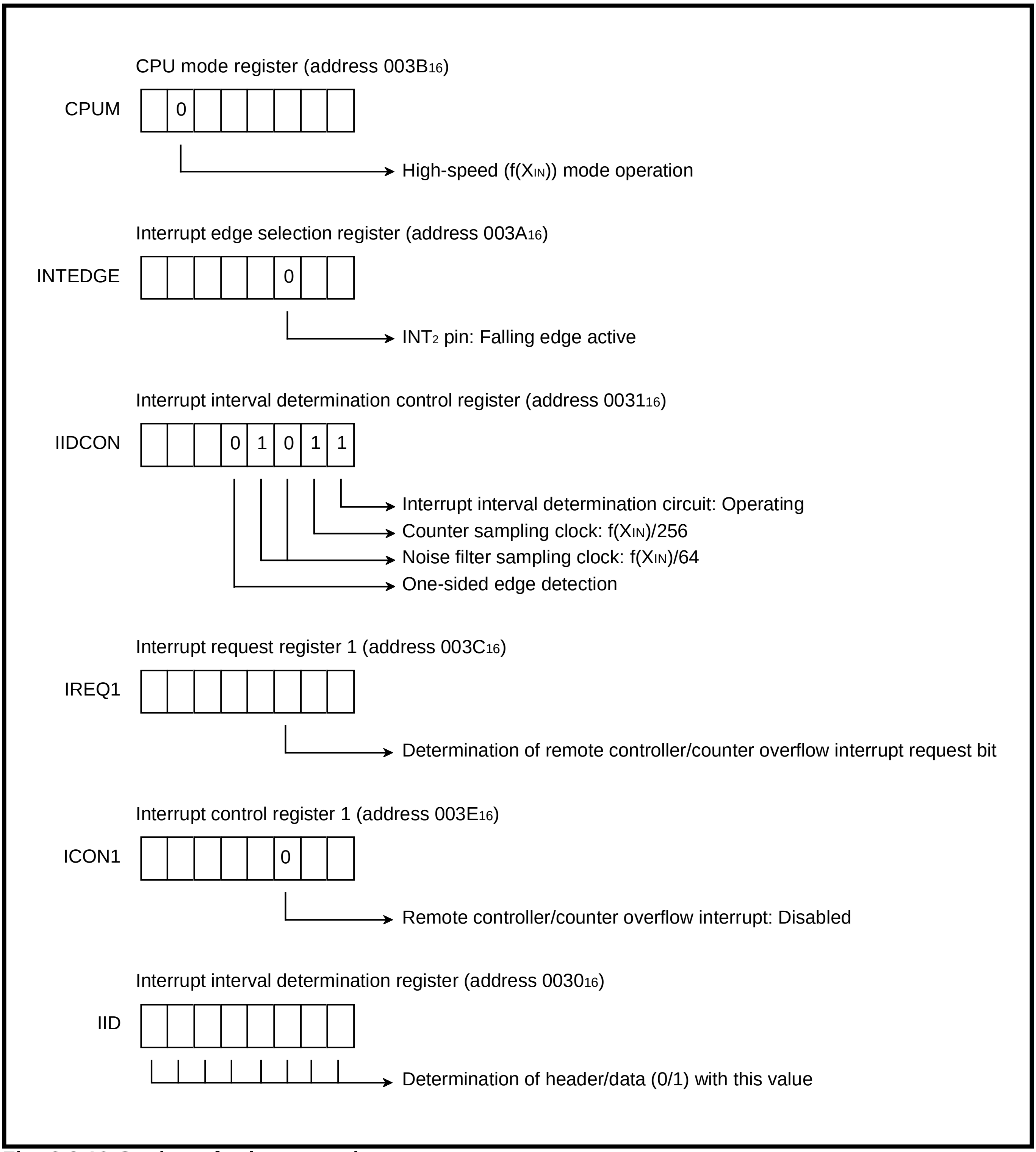


Fig. 2.8.10 Setting of relevant registers

Control procedure: When the registers are set as shown in Figure 2.8.10, remote-control signals are receivable. Figure 2.8.11 shows the control procedure, and Figure 2.8.12 shows the reception of remote-control data (timer 2 interrupt).

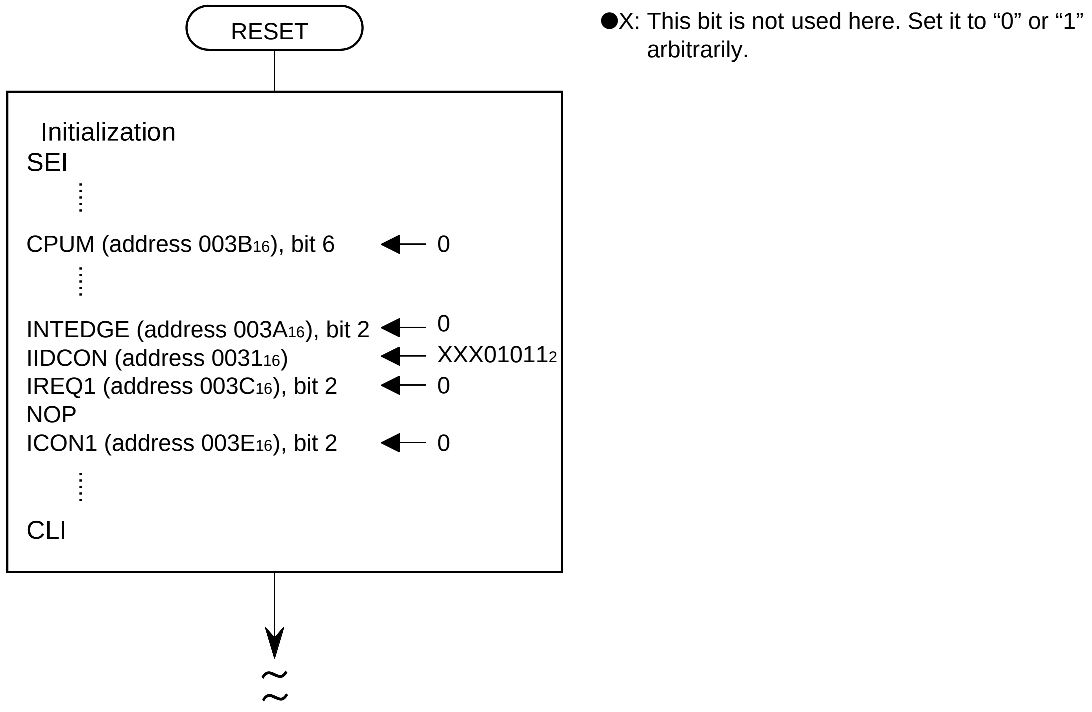


Fig. 2.8.11 Control procedure

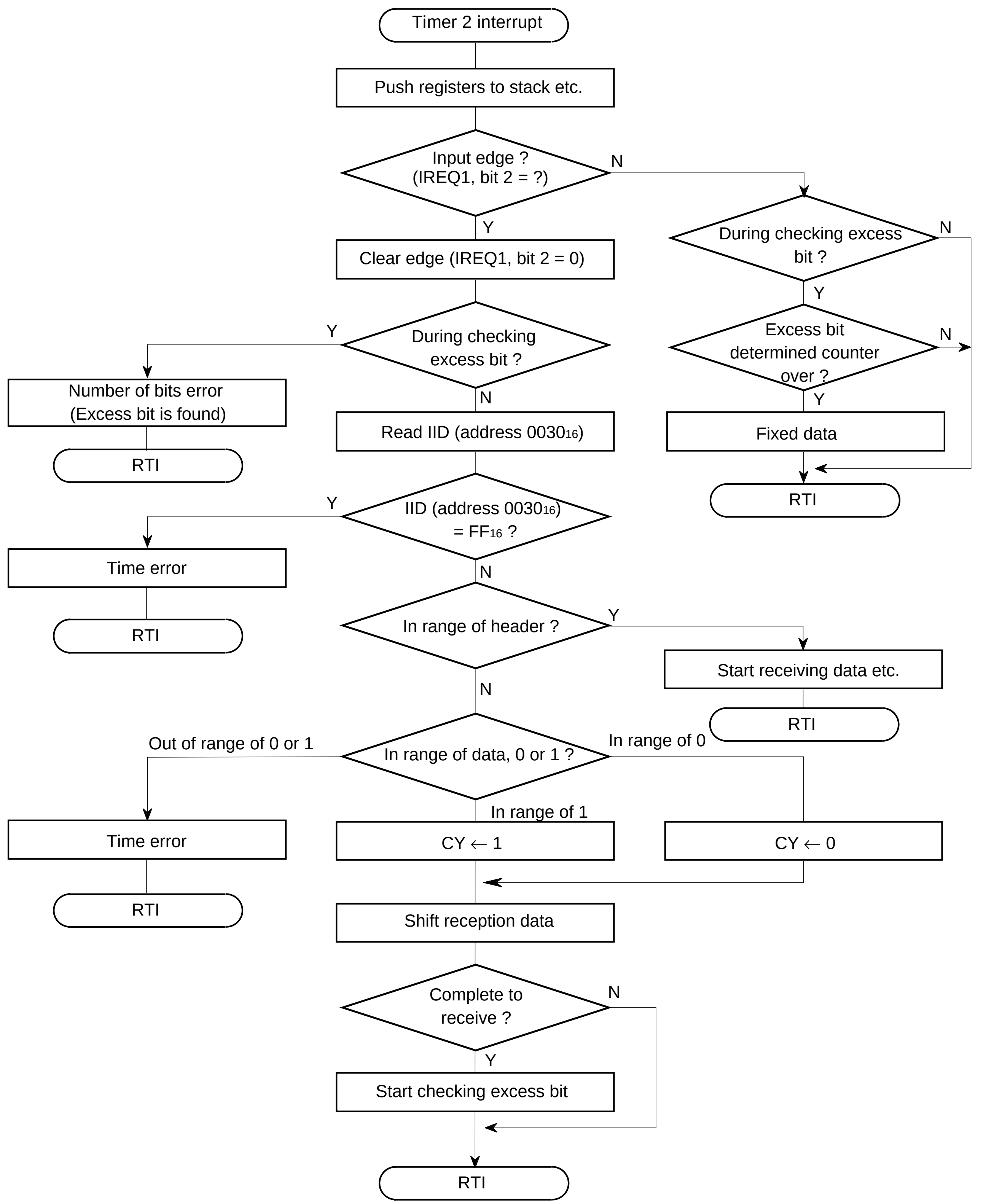


Fig. 2.8.12 Reception of remote-control data (timer 2 interrupt)

2.9 Watchdog timer

This paragraph describes the setting method of watchdog timer relevant register, notes etc.

2.9.1 Memory assignment

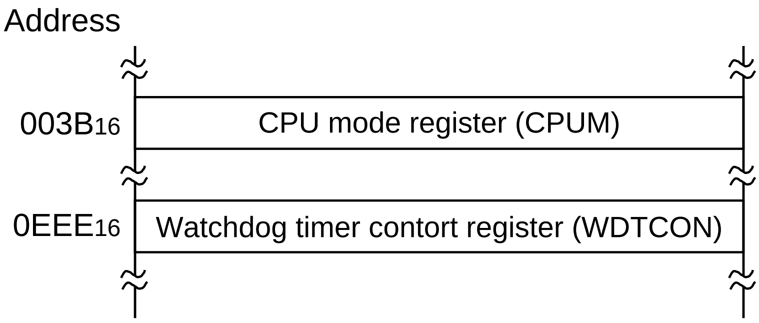


Fig. 2.9.1 Memory assignment of watchdog timer relevant register

2.9.2 Relevant register

Watchdog timer control register

b7 b6 b5 b4 b3 b2 b1 b0

Watchdog timer control register
(WDTCON: address $0EEE_{16}$)

b	Name	Functions	At reset	R	W
0	Watchdog timer H (high-order 6 bits of reading exclusive)		1	O	×
1			1	O	×
2			1	O	×
3			1	O	×
4			1	O	×
5			1	O	×
6	STP instruction disable bit	0: STP instruction enabled 1: STP instruction disabled	0	O	O
7	Watchdog timer H count source selection bit	0: Watchdog timer L underflow 1: $f(X_{IN})/8$ or $f(X_{CIN})/16$	0	O	O

Fig. 2.9.2 Structure of Watchdog timer control register

CPU mode register

b7 b6 b5 b4 b3 b2 b1 b0: [ ][ ][ ][ ][1][ ][0][0]

CPU mode register
(CPUM: address $3B_{16}$)

b	Name	Functions	At reset	R	W
0	Processor mode bits	b1 b0 00 : Single-chip mode 01 : } Not available 10 : } 11 : }	0	O	O
1			0	O	O
2	Stack page selection bit	0 : Page 0 1 : Page 1	0	O	O
3	Fix this bit to “1”.		1	O	O
4	Port Xc switch bit	0: I/O port function 1: X_{CIN}-X_{COUT} oscillation function	0	O	O
5	Main clock (X_{IN}-X_{OUT}) stop bit	0: Oscillating 1: Stopped	0	O	O
6	Main clock division ratio selection bit	0: f(X_{IN}) (high-speed mode) 1: f(X_{IN})/4 (middle-speed mode)	1	O	O
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	O	O

Fig. 2.9.3 Structure of CPU mode register

2.9.3 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 each cycle of 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 2.9.4 shows the connection of watchdog timer and the setting of the division ratio.
Figure 2.9.5 shows the setting of relevant registers and Figure 2.9.6 shows the control procedure.

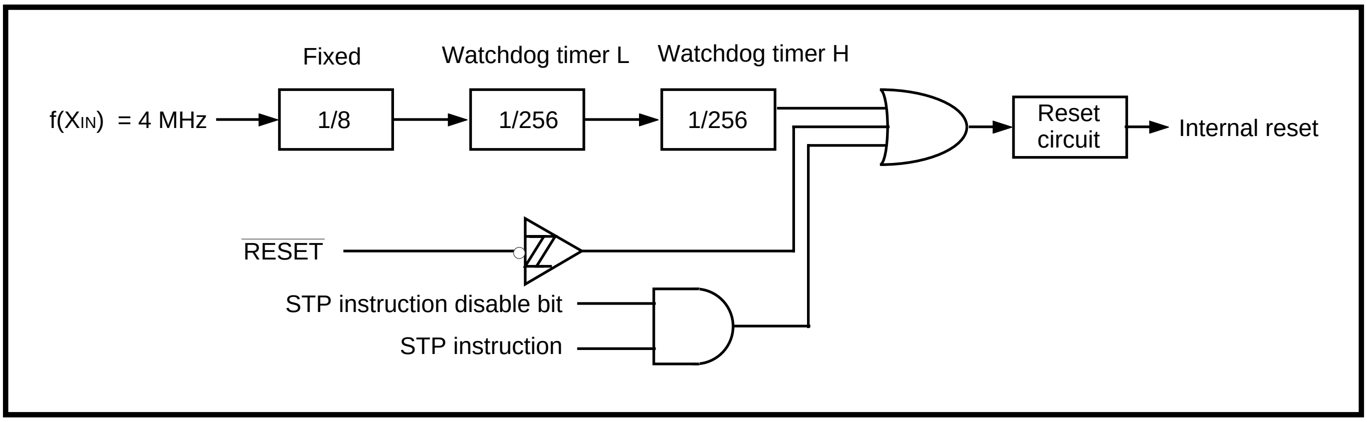


Fig. 2.9.4 Connection of watchdog timer and setting of division ratio

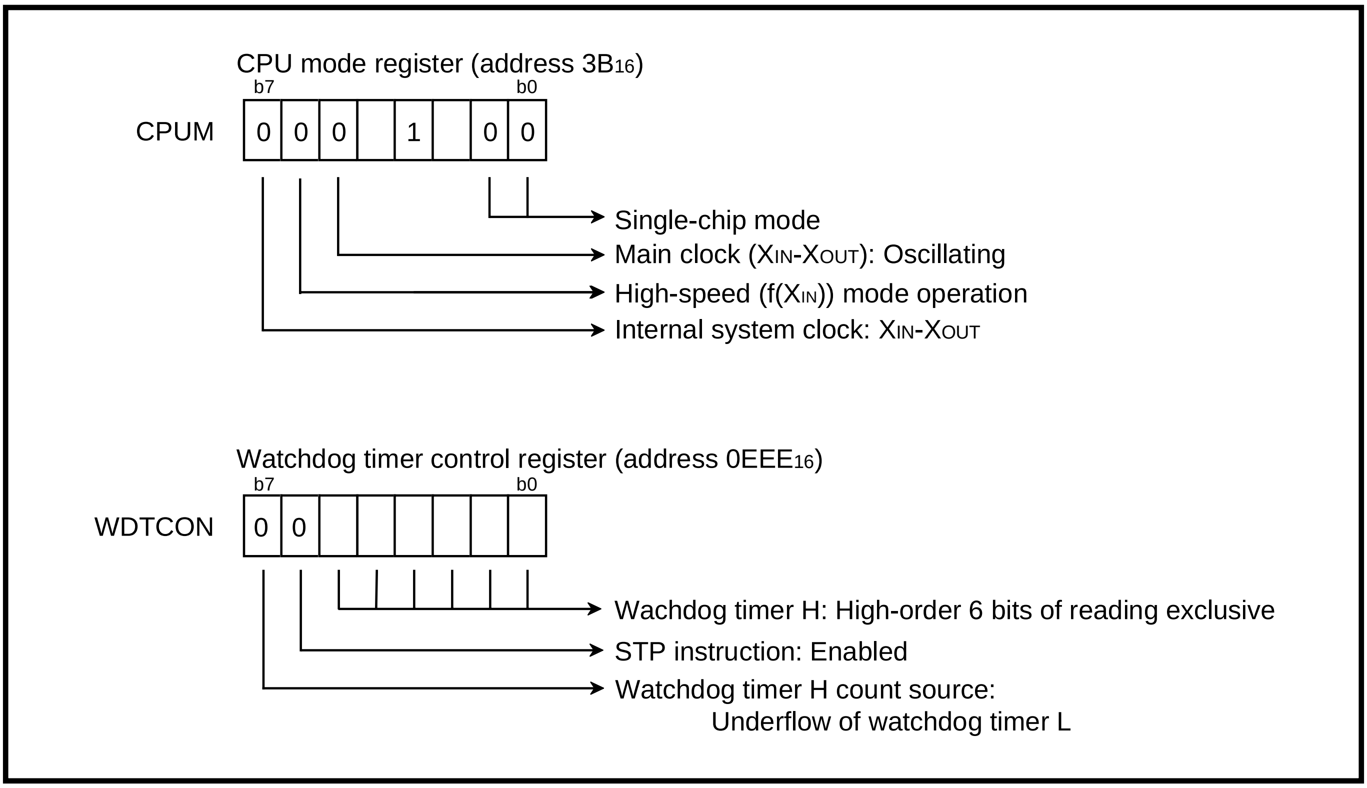


Fig. 2.9.5 Setting of relevant registers

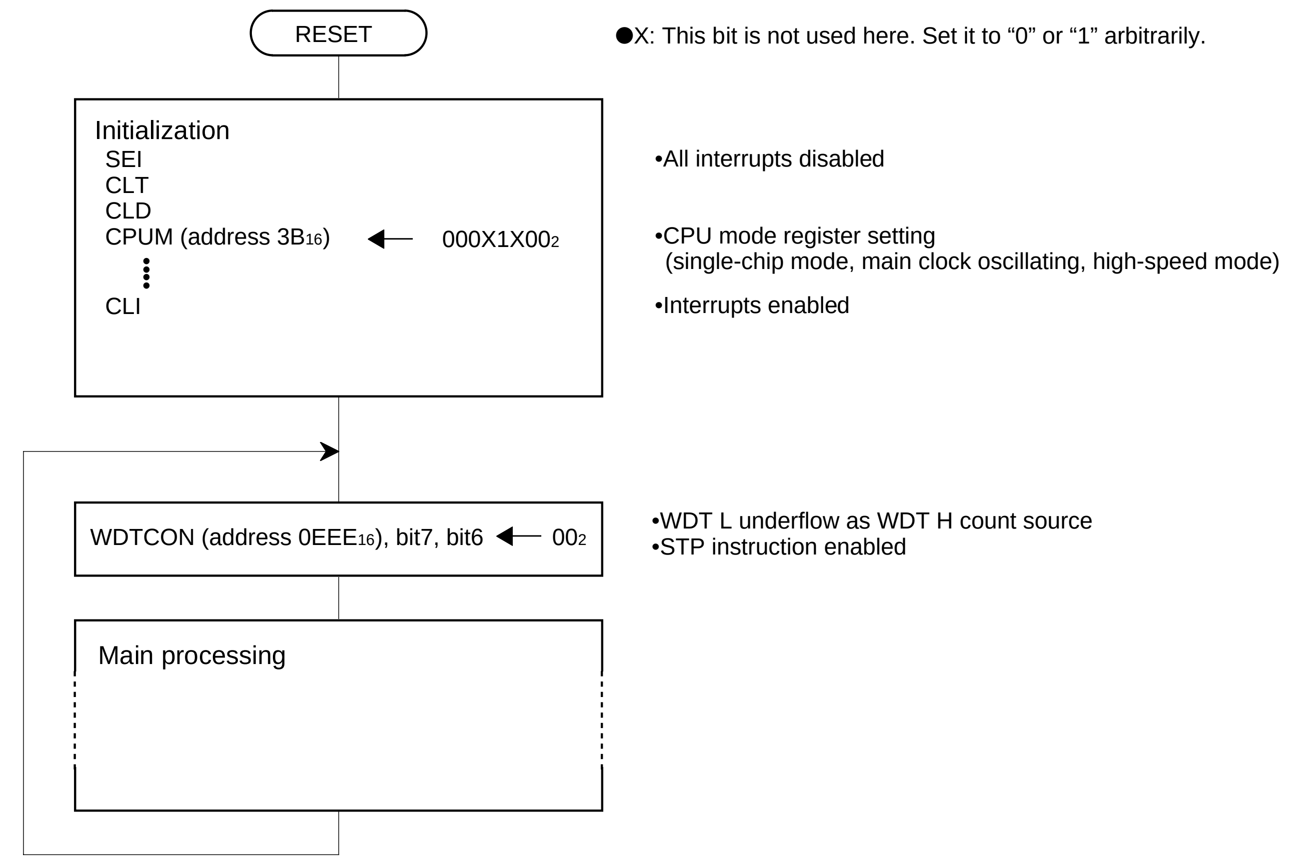


Fig. 2.9.6 Control procedure

2.9.4 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 $0EEE_{16}$) 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 $0EEE_{16}$), it cannot be programmed to “0” again. This bit becomes “0” after reset.

2.10 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.

2.10.1 Memory assignment

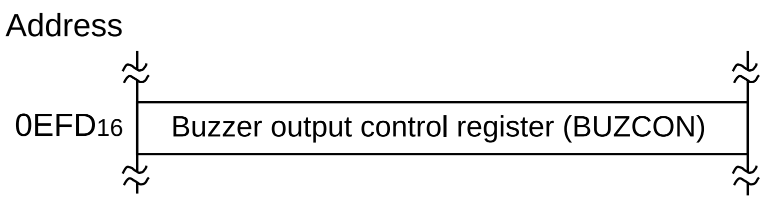


Fig. 2.10.1 Memory assignment of buzzer output circuit relevant register

2.10.2 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 2.10.2 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 $0EFD_{16}$)

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: $P7_7$ and $P9_7$ function as ordinary ports. 0 1: $P7_7$/B_{UZ01} functions as a buzzer output. 1 0: $P9_7$/B_{UZ02}/AN_{15} 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. 2.10.2 Structure of buzzer output control register

2.10.3 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 2.10.3 shows the connection of buzzer output circuit and the setting of the division ratio. Figure 2.10.4 shows the setting of relevant register. Figure 2.10.5 shows the control procedure.

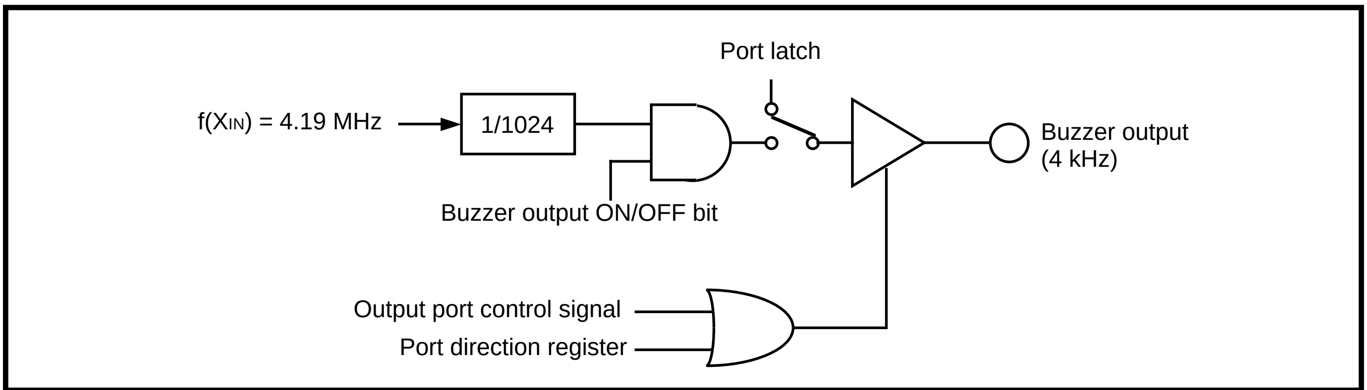


Fig. 2.10.3 Connection of buzzer output circuit and setting of division ratio

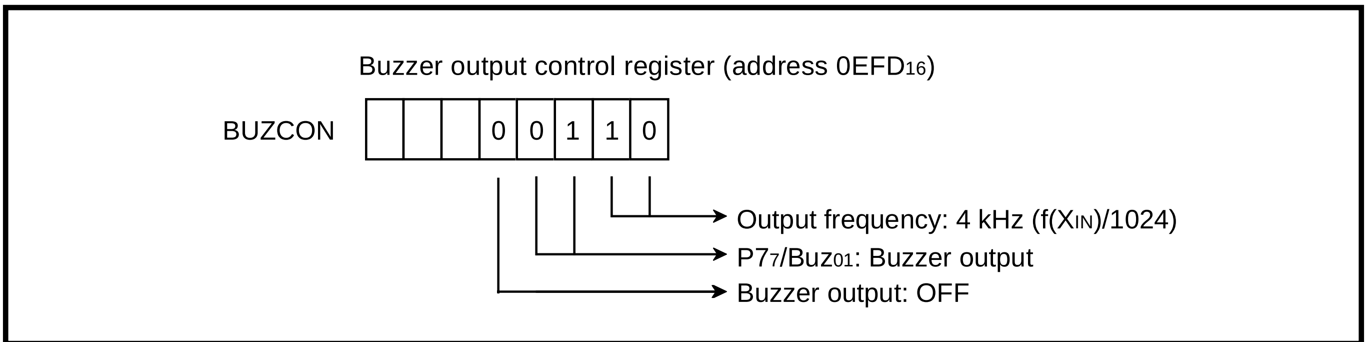


Fig. 2.10.4 Setting of relevant register

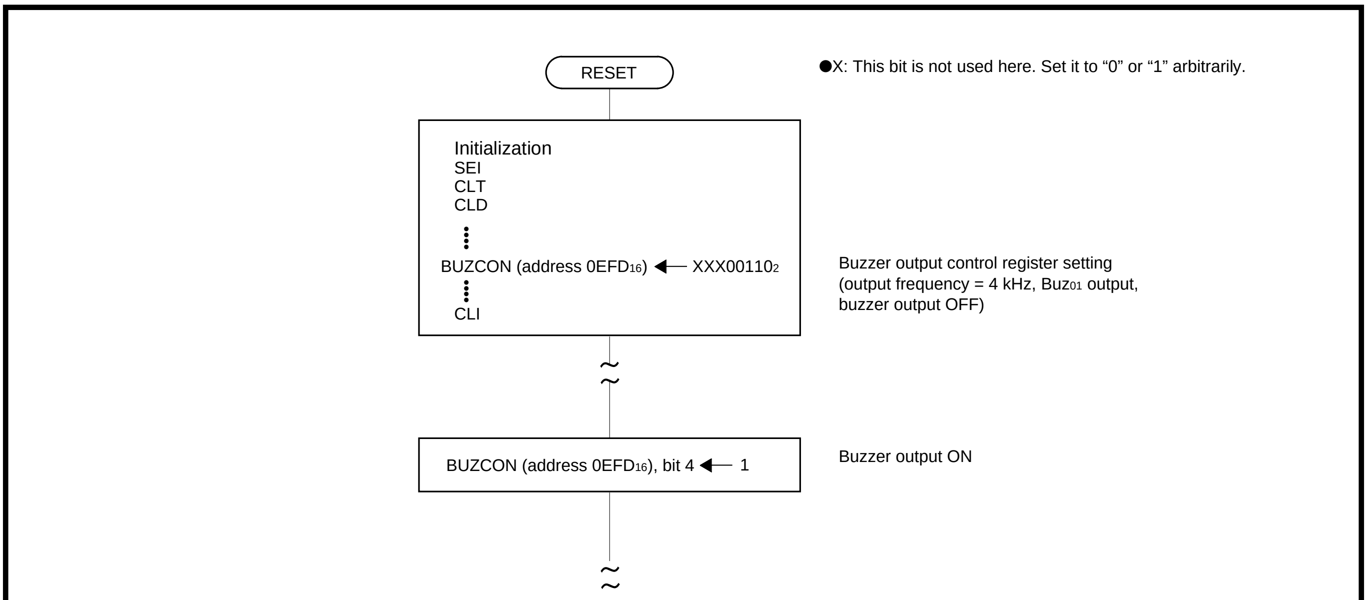


Fig. 2.10.5 Control procedure

2.11 Reset circuit

The reset state is caused by applying an "L" level to the $\overline{RESET}$ pin. After that, the reset state is released by applying an "H" level to the $\overline{RESET}$ pin, so that the program is executed in the middle-speed mode from the contents of the reset vector address.

2.11.1 Connection example of reset IC

Figure 2.11.1 shows the example of power-on reset circuit. Figure 2.11.2 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.

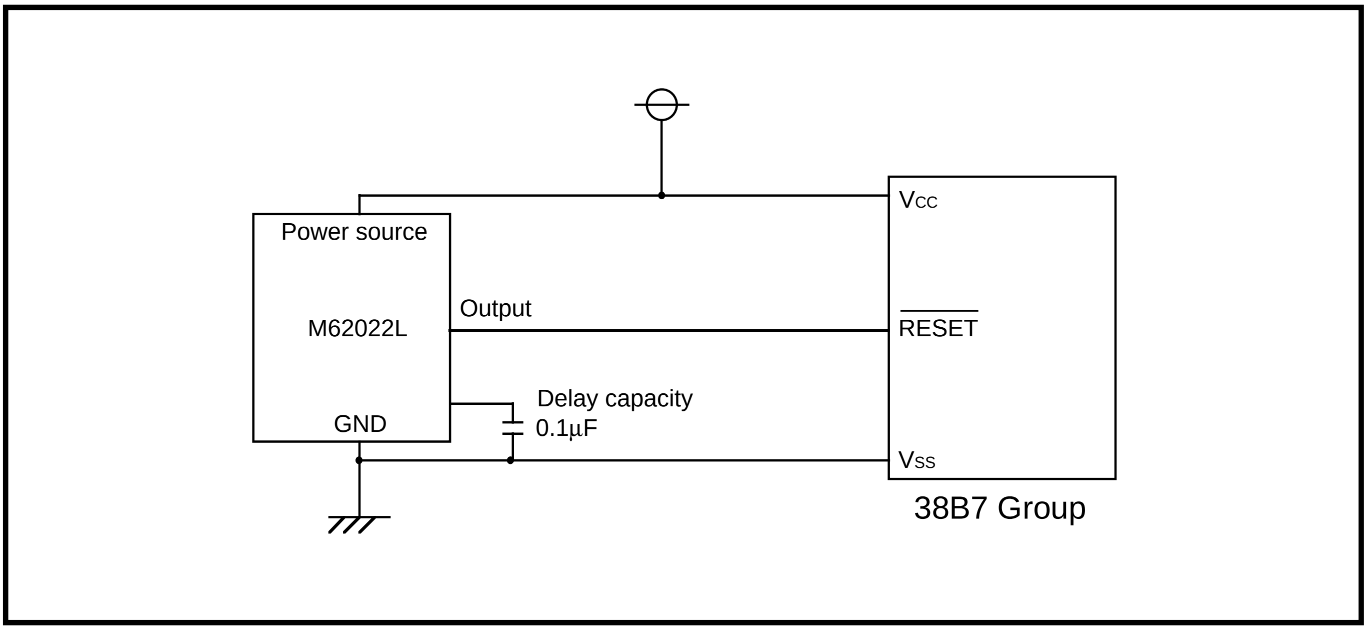


Fig. 2.11.1 Example of power-on reset circuit

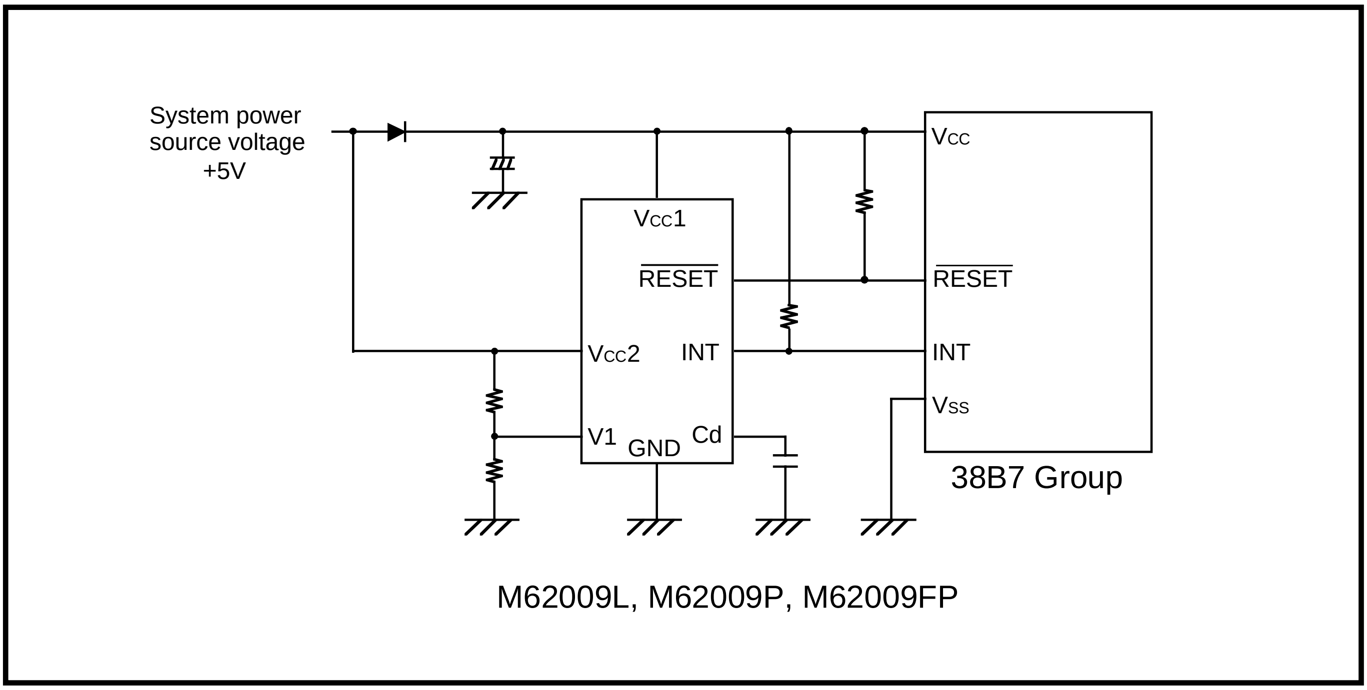


Fig. 2.11.2 RAM backup system example

2.11 Reset circuit

2.11.2 Notes on reset

(1) Reset input voltage control

Make sure that the reset input voltage is 0.54 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.

2.11.3 Each port state during “L” state of $\overline{\text{RESET}}$ pin

Table 2.11.1 shows a pin state during “L” state of $\overline{\text{RESET}}$ pin.

Table 2.11.1 Pin state during “L” state of $\overline{\text{RESET}}$ pin

Pin name	Pin state
P0, P2	Output port (with pull-down resistor)
P1, P3	Input port (with pull-down resistor)
P4, P5, $P6_0$ to $P6_3$	Input port (without pull-down resistor)
$P6_4$ to $P6_7$, P7, $P8_0$ to $P8_3$, P9, PA, PB_0 to PB_6	Input port (floating)

2.12 Clock generating circuit

This paragraph explains the setting method of clock generating circuit relevant register, etc.

2.12.1 Relevant register

Figure 2.12.1 shows the structure of the CPU mode register.

CPU mode register

b7	b6	b5	b4	b3	b2	b1	b0
				1		0	0

CPU mode register
(CPUM: address $3B_{16}$)

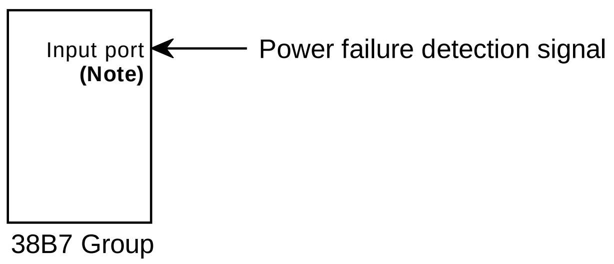
b	Name	Functions	At reset	R	W
0	Processor mode bits	b1 b0 00 : Single-chip mode 01 : 10 : } Not available 11 :	0	O	O
1			0	O	O
2	Stack page selection bit	0 : Page 0 1 : Page 1	0	O	O
3	Fix this bit to “1”.		1	O	O
4	Port Xc switch bit	0: I/O port function 1: XCIN-XCOUT oscillation function	0	O	O
5	Main clock (XIN-XOUT) stop bit	0: Oscillating 1: Stopped	0	O	O
6	Main clock division ratio selection bit	0: f(XIN) (high-speed mode) 1: f(XIN)/4 (middle-speed mode)	1	O	O
7	Internal system clock selection bit	0: XIN–XOUT selection (middle-/high-speed mode) 1: XCIN–XCOUT selection (low-speed mode)	0	O	O

Fig. 2.12.1 Structure of CPU mode register

2.12.2 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.



Note: Signal is detected by inputting to each input port, interrupt input pin, and analog input pin.

Fig. 2.12.2 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 2.12.3 shows the status transition diagram during power failure and Figure 2.12.4 shows the setting of relevant registers.

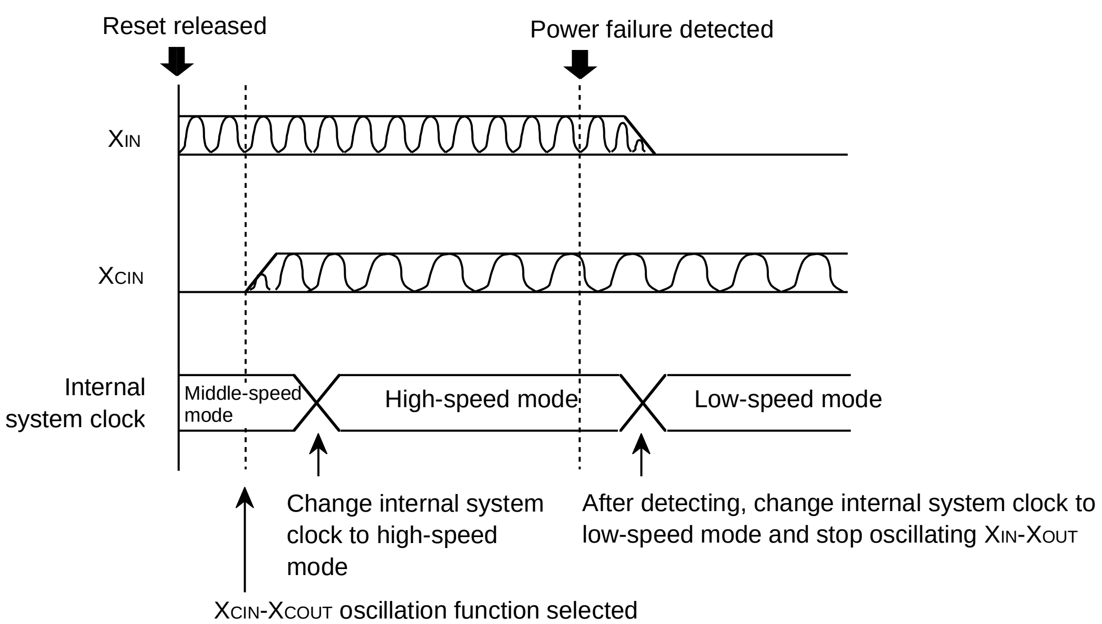
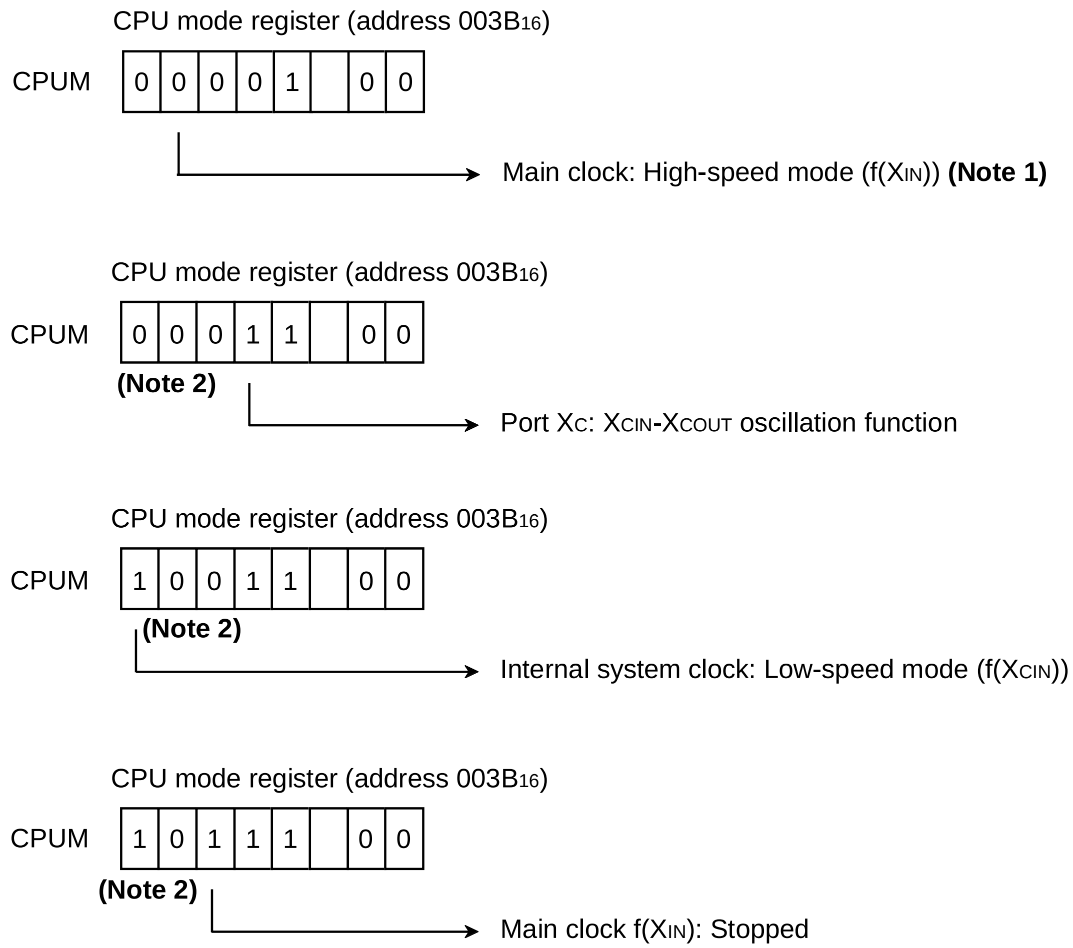


Fig. 2.12.3 Status transition diagram during power failure



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. 2.12.4 Setting of relevant registers

Control procedure: Set the relevant registers in the order shown below to prepare for a power failure.

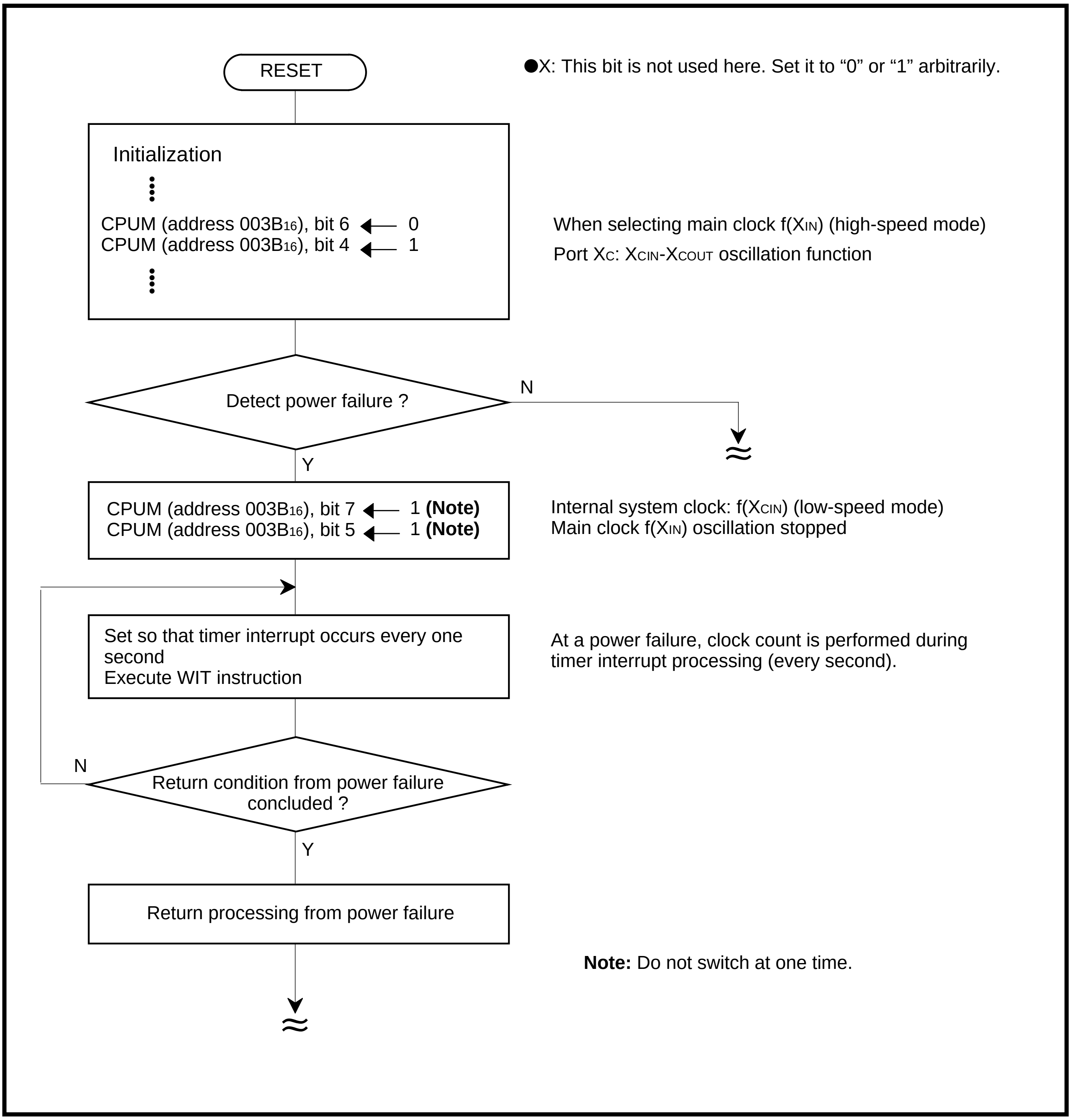


Fig. 2.12.5 Control procedure

(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 2.12.2 and 2.12.3.

Figure 2.12.6 shows the structure of clock counter, Figures 2.12.7 and 2.12.8 show the setting of relevant registers.

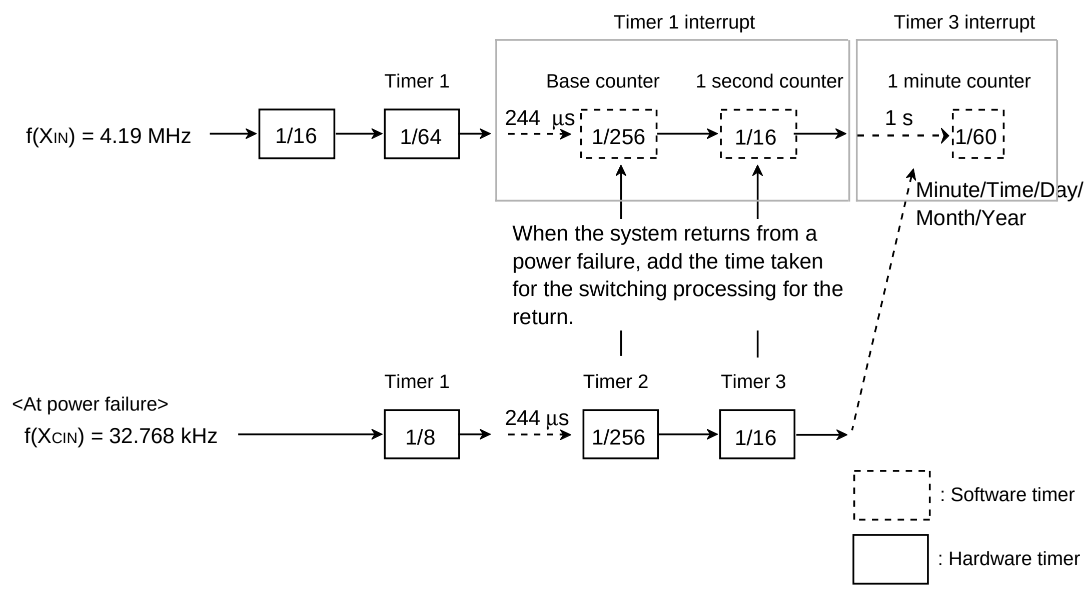


Fig. 2.12.6 Structure of clock counter

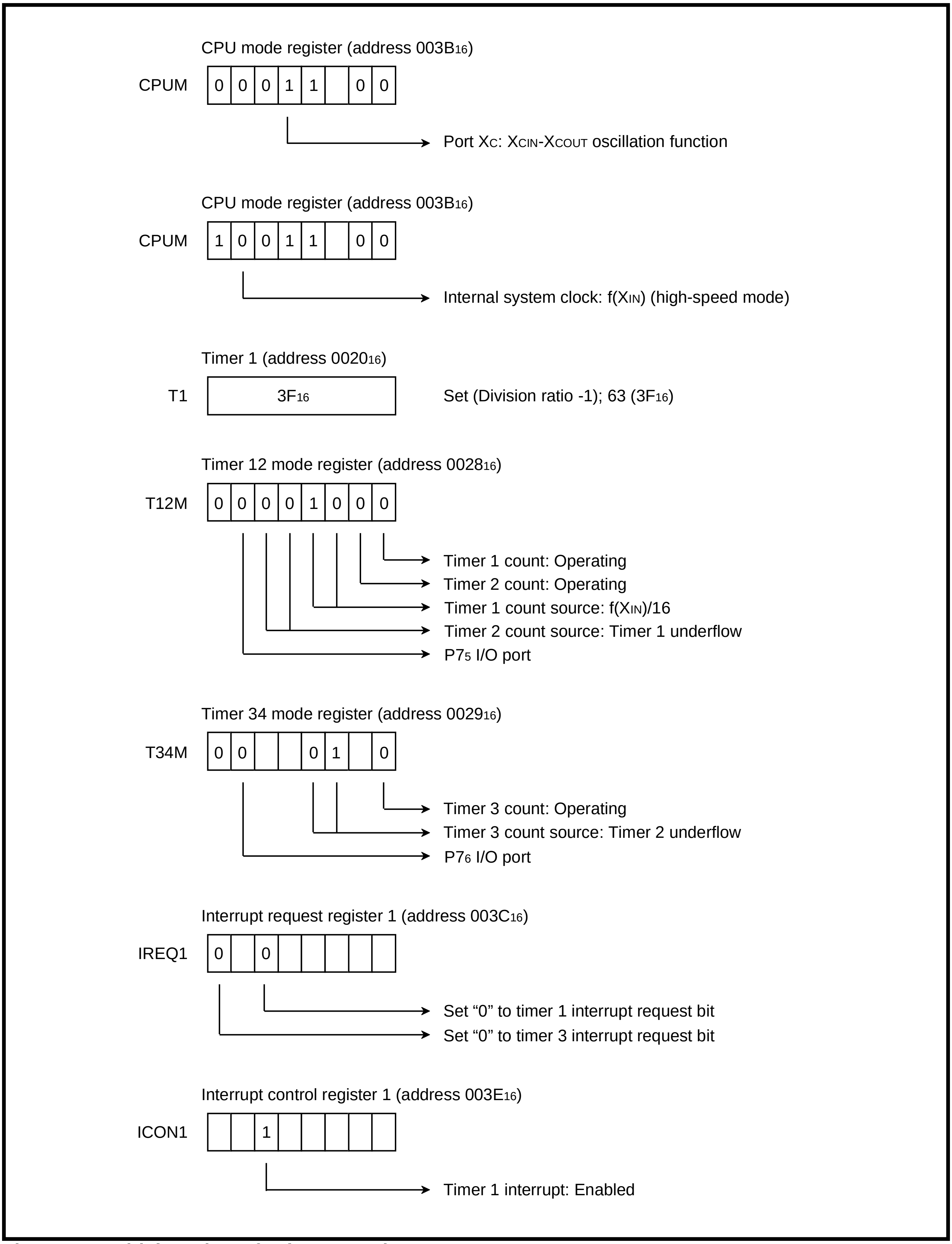


Fig. 2.12.7 Initial setting of relevant registers

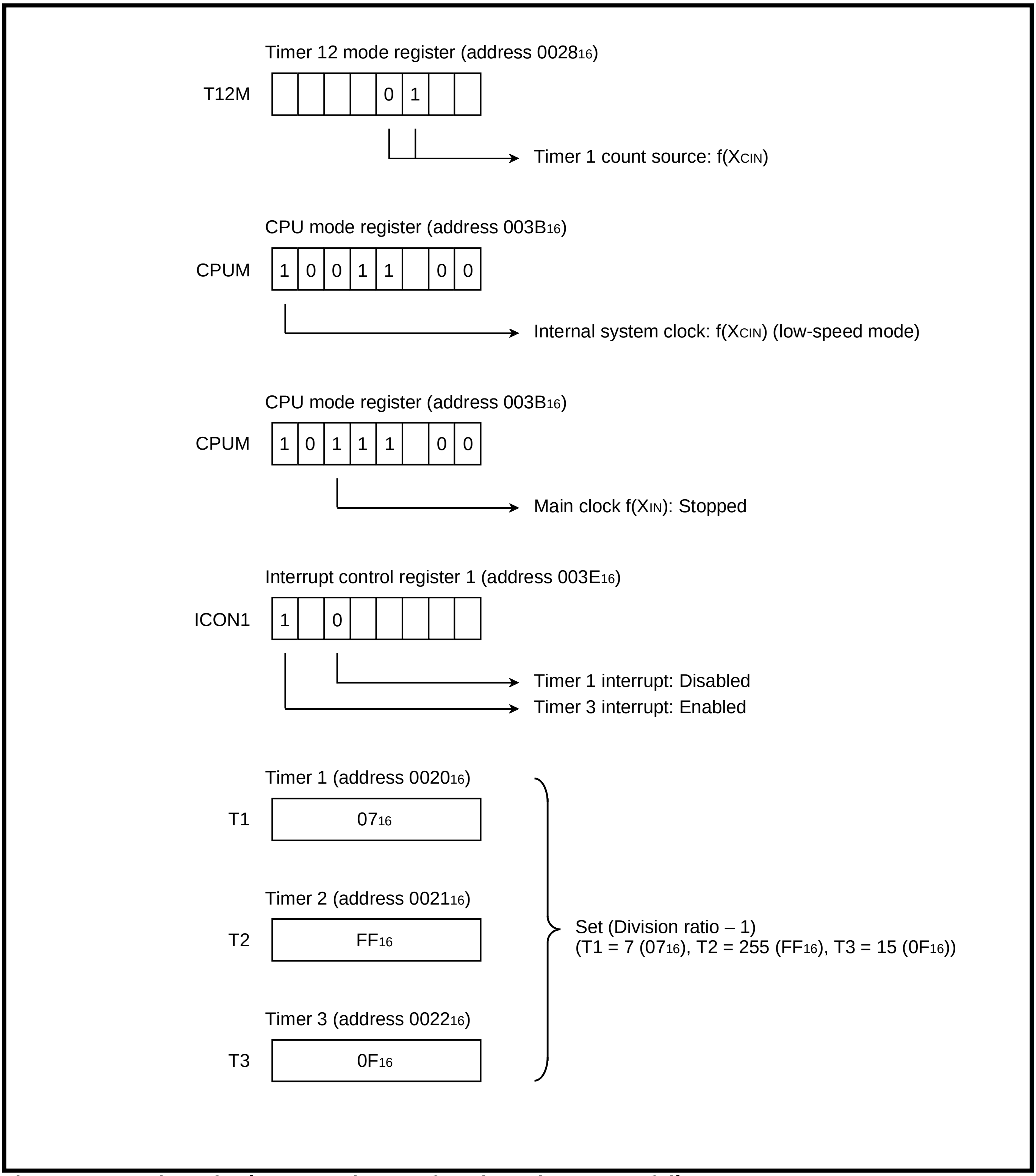


Fig. 2.12.8 Setting of relevant registers after detecting power failure

Control procedure: Set the relevant registers in the order shown below to prepare for a power failure.

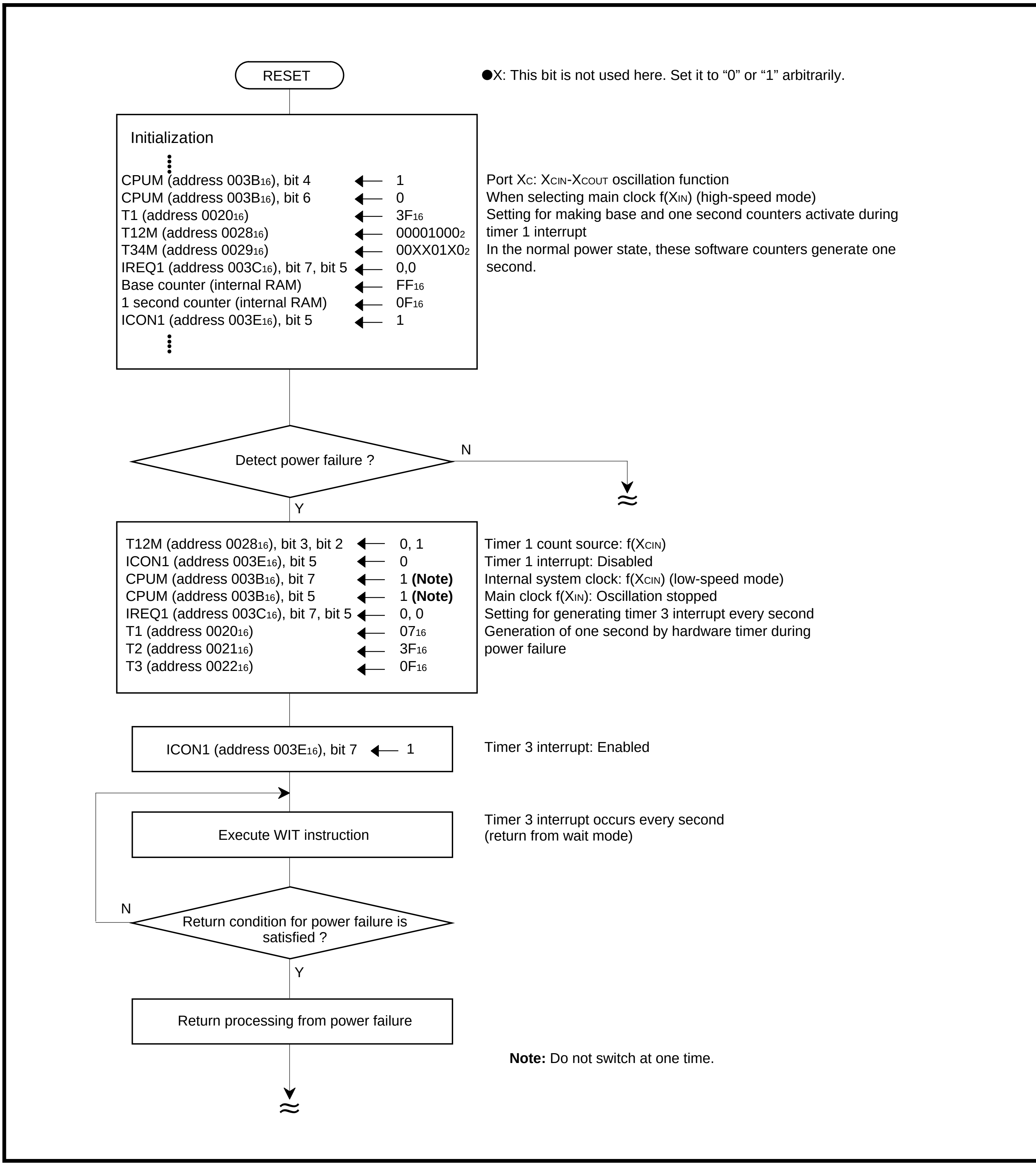
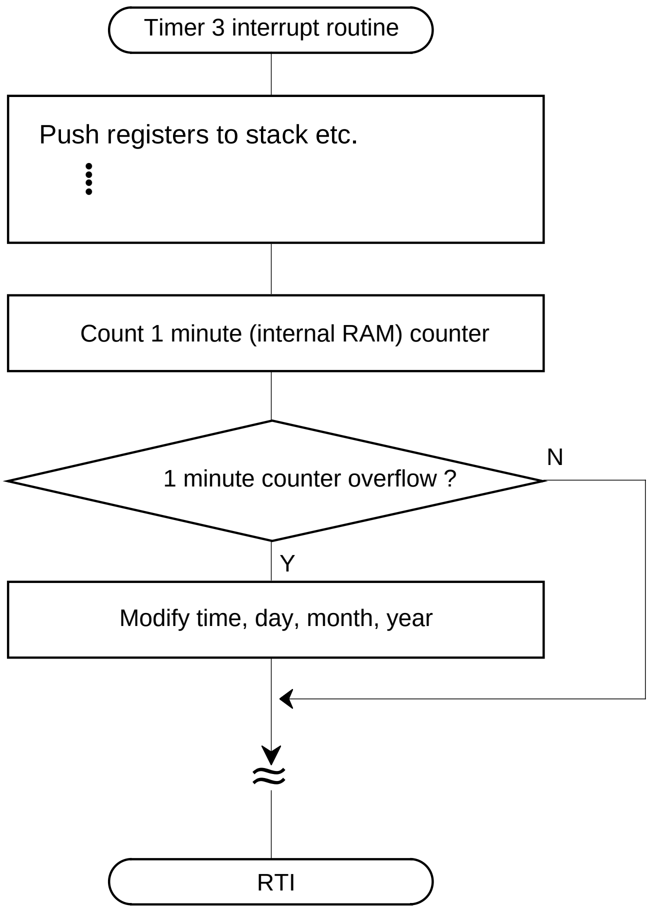


Fig. 2.12.9 Control procedure



Timer 3 interrupt routine
Push registers to stack etc.
Count 1 minute (internal RAM) counter
1 minute counter overflow ?
N
Y
Modify time, day, month, year
RTI

2.13 Flash memory

This paragraph explains the registers setting method and the notes relevant to the flash memory version.

2.13.1 Overview

The flash memory version has functions similar to those of the mask ROM version except that the flash memory is built-in. However, some of SFR area of flash memory version is different from those of the mask ROM version (refer to “**2.13.2 Memory map**”).

In the flash memory version, the built-in flash memory can be operated by using the following three modes.

• CPU reprogramming mode
• Parallel input/output mode
• Serial input/output mode

2.13.2 Memory map

M38B79FFFP has the built-in flash memory of 60 Kbytes.

Figure 2.13.1 shows the memory map of the flash memory version.

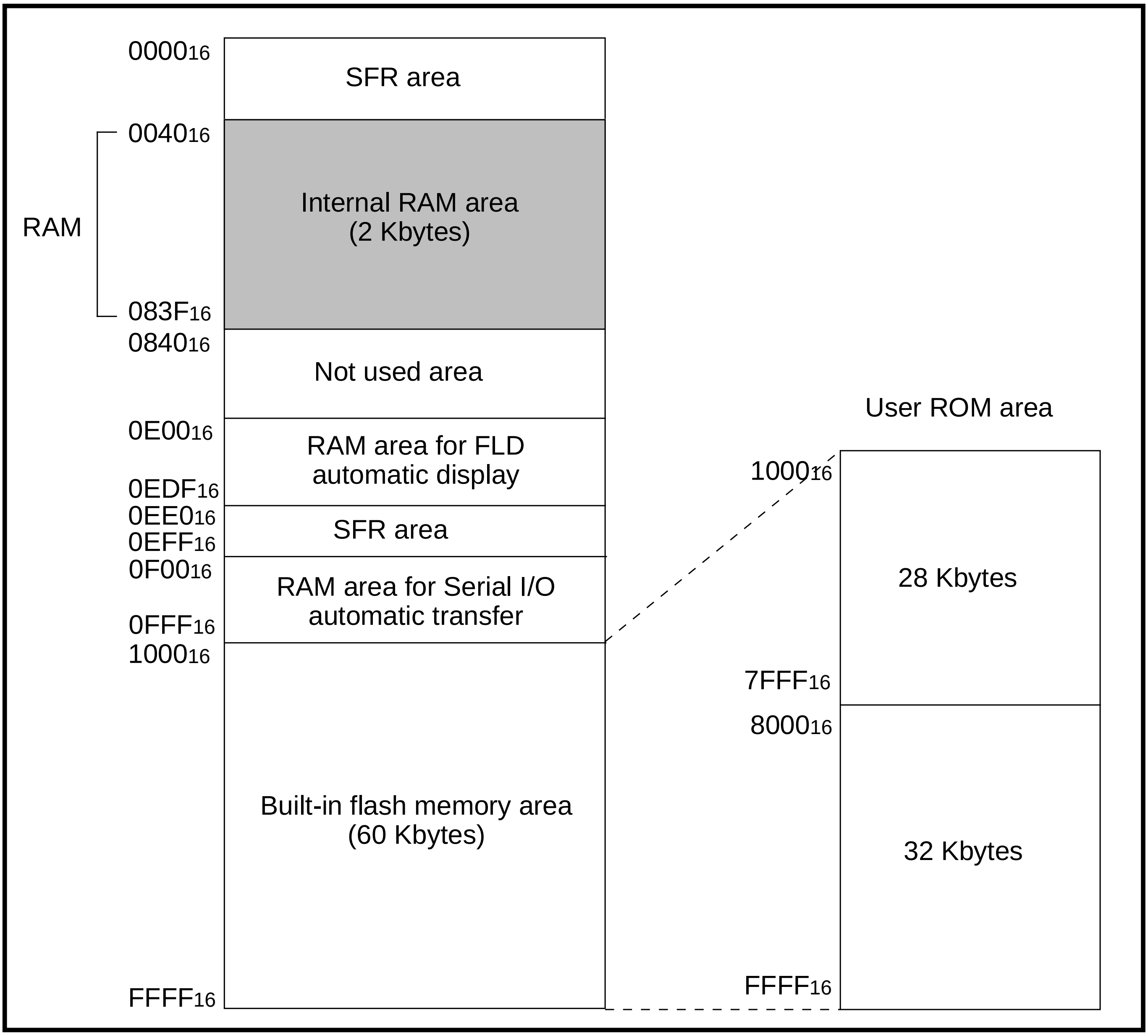


Fig. 2.13.1 Memory map of flash memory version for 38B7 Group

2.13.3 Relevant registers

Address	Register
$003B_{16}$	CPU mode register (CPUM)
$0EFE_{16}$	Flash memory control register (FCON)
$0EFF_{16}$	Flash command register (FCMD)

Fig. 2.13.2 Memory map of registers relevant to flash memory

Flash memory control register

b7 b6 b5 b4 b3 b2 b1 b0: ■ 0 □ □ 0 □ □ □

Flash memory control register (FCON: address $0EFE_{16}$)

b	Name	Functions	At reset	R	W
0	CPU reprogramming mode select bit **(Note)**	0: CPU reprogramming mod is invalid. (Normal operation mode) 1: When applying 0 V or V_{PPL} to CNV_{SS}/V_{PP} pin, CPU reprogramming mode is invalid. When applying V_{PPH} to CNV_{SS}/V_{PP} pin, CPU reprogramming mode is valid.	0	○	○
1	Erase/Program busy flag	0: Erase and program are completed or not have been executed. 1: Erase/program is being executed.	0	○	○
2	CPU reprogramming mode monitor flag	0: CPU reprogramming mode is invalid. 1: CPU reprogramming mode is valid.	0	○	○
3	Fix this bit to “0”.		0	○	○
4	Erase/Program area select bits	b5 b4 0 0: Addresses 1000_{16} to $FFFF_{16}$ (total 60 Kbytes) 0 1: Addresses 1000_{16} to $7FFF_{16}$ (total 28 Kbytes) 1 0: Addresses 8000_{16} to $FFFF_{16}$ (total 32 Kbytes) 1 1: Not available	0	○	○
5			0	○	○
6	Fix this bit to “0”.		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: Bit 0 can be reprogrammed only when 0 V is applied to the CNV_{SS}/V_{PP} pin.

Fig. 2.13.3 Structure of Flash memory control register

Flash command register

b7 b6 b5 b4 b3 b2 b1 b0

Flash command register
(FCMD: address $0EFF_{16}$)

b	Functions	At reset	R	W
0	Writing of software command <Software command name> <Command code> •Read command "00_{16}" •Program command "40_{16}" •Program verify command "$C0_{16}$" •Erase command "20_{16}" + "20_{16}" •Erase verify command "$A0_{16}$" •Reset command "FF_{16}" + "FF_{16}"	0	×	○
1		0	×	○
2		0	×	○
3		0	×	○
4		0	×	○
5		0	×	○
6		0	×	○
7		0	×	○

Note: The flash command register is write exclusive register.

Fig. 2.13.4 Structure of Flash command register

CPU mode register

b7 b6 b5 b4 b3 b2 b1 b0
(b3 = 1, b1 = 0, b0 = 0)

CPU mode register
(CPUM: 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	Fix this bit to "1".		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	○	○

Note: B6, b7 function in the CPU reprogramming mode is described below.

b	Name	Functions	At reset	R	W
6	Main clock division ratio selection bits	b7 b6 0 0: φ = f(XIN) (high-speed mode) 0 1: φ = f(XIN)/4 (middle-speed mode) 1 0: φ = f(XCIN)/2 (low-speed mode) 1 1: Not available	1	○	○
7			0	○	○

Fig. 2.13.5 Structure of CPU mode register

2.13.4 Parallel I/O mode

In the parallel I/O mode, program/erase to the built-in flash memory area can be performed by a general ERPOM programmer.
Set the programming mode of EPROM programmer to M5M28F101 and the memory area of program/erase to 01000_{16} to $0FFFF_{16}$. Be careful especially when erasing because if the setting of the memory area is mistaken when erasing, the products are damaged eternally.
Table 2.13.1 shows the setting of EPROM programmer when programming in the parallel I/O mode.

Recommended programmer: R4945A provided by ADVANTEST CORPORATION (http://www.advantest.co.jp/index-e.html)

Table 2.13.1 Setting of EPROM programmer when parallel programming

Products	Programming adapter	Programming mode	Memory area
M38B79FFFP	PCA4738F-100	M5M28F101	01000_{16} to $0FFFF_{16}$

2.13.5 Serial I/O mode

Table 2.13.2 shows the pin connection example using EFP-I* between the programmer and the microcomputer when programming in the serial I/O mode.
*EFP-I provided by Suisei Electronics System Co., Ltd. (http://www.suisei.co.jp/index_e.htm)
(Asia and Oceania limited-product)

Table 2.13.2 Connection example to programmer when serial programming

EFP-I		38B7 Group flash memory version	
Signal name	Target connector Line number	Pin name	Pin number
BUSY	1	$P6_7/\overline{S_{RDY2}}/S_{CLK22}/FLD_{55}$	33
VPP (**Note 1**)	2	CNV_{SS} (**Note 1**)	17
VDD (**Note 3**)	3	V_{CC} (**Note 3**)	24
SCL	4	$P6_6/S_{CLK21}/FLD_{54}$	34
SDA	5	$P6_4$/RxD/FLD_{52}	36
PGM/OE	6	$P3_7/FLD_{31}$	57
RESET	7	$\overline{RESET}$	18
GND (**Note 2**)	8	V_{SS}, AV_{SS} (**Note 2**)	21, 97

Notes 1: Connect an approximate 0.01 µF capacitor between CNV_{SS}/V_{PP} and GND for noise elimination.
2: When a serial programmer is connected, at first, connect both GNDs to be the same GND level.
3: When the V_{CC} power has been already supplied to the target board, do not connect the VDD supply pin of the serial programmer to V_{CC} of the target board.

2.13.6 CPU reprogramming mode

In the CPU reprogramming mode, by executing the software command with Central Processing Unit (CPU), the built-in flash memory area can be reprogrammed. Accordingly, the contents of the built-in flash memory area can be reprogrammed with the microcomputer mounted on board, without using the ROM programmer. Program the reprogramming program in advance to the built-in flash memory area. However, in the CPU reprogramming mode, the read from the built-in flash memory cannot be performed. Accordingly, after transferring the reprogramming control program on the internal RAM, not the built-in flash memory, execute it on the RAM.

In the CPU reprogramming mode, read command, program command, program verify command, erase command, erase verify command, and reset command can be used. As for details of each command, refer to "**CHAPTER 1 Flash memory mode 3 (CPU reprogramming mode)**".

(1) CPU reprogramming mode beginning/release procedure

Operation procedure in the reprogramming mode for the built-in flash memory is described.
As for the control example, refer to "**2.13.7 (2) Control example in the CPU reprogramming mode**."

[Beginning procedure]

① Apply 0 V to the CNV_{SS}/V_{PP} pin for reset release.
② Set the CPU mode register.
③ After CPU reprogramming mode control program is transferred to internal RAM, jump to this control program on RAM. (The following operations are controlled by this control program).
④ Set "1" to the CPU reprogramming mode select bit (bit 0 of address $0EFE_{16}$).
⑤ Apply $V_{PP}H$ to the CNV_{SS}/V_{PP} pin.
⑥ Wait till CNV_{SS}/V_{PP} pin becomes 12 V.
⑦ Read the CPU reprogramming mode monitor flag (bit 2 of address $0EFE_{16}$) to confirm that the CPU reprogramming mode is valid.
⑧ The operation of the flash memory is executed by software-command-writing to the flash command register (address $0EFF_{16}$).

Note: The following are necessary other than this:
- Control for data which is input from the external (serial I/O etc.) and to be programmed to the flash memory.
- Initial setting for ports, etc.
- Writing to the watchdog timer

[Release procedure]

① Apply 0 V to the CNV_{SS}/V_{PP} pin.
② Wait till CNV_{SS}/V_{PP} pin becomes 0 V.
③ Set the CPU reprogramming mode select bit (bit 0 of address $0EFE_{16}$) to "0".

Also, execute the following processing before the CPU reprogramming mode is selected so that interrupts will not occur during the CPU reprogramming mode.
• Set the interrupt disable flag (I) to “1”
In the CPU reprogramming mode, write to the watchdog timer control register (address $0EEE_{16}$) periodically in order not to generate the reset by the underflow of the watchdog timer H.
During the program execution (programming time: max. 10 µs), watchdog timer H is set to “FF_{16}”, watchdog timer L is set to “FF_{16}” and the count is stopped. The count is started again after the program is executed or the execution of erase is completed. Accordingly, the setting of write period of the watchdog timer control register is no problem except for the program time and erase time.
When the interrupt request or reset occurs in the CPU reprogramming mode, the microcomputer enters the following state;
• Interrupt occurs
This may cause a program runaway because the read from the flash memory which has the interrupt vector area cannot be performed.
• Underflow of watchdog timer H, reset
This may cause a microcomputer reset; the built-in flash memory control circuit and the flash memory control register are reset.

Also, when the above interrupt and reset occur during program/erase, error data may still exist after reset release because the reprogramming of the flash memory is not completed, so that be careful. In this case, reprogramming of the flash memory in the parallel I/O mode or serial I/O mode is required.

2.13.7 Flash memory mode application examples

The control pin processing example on the system board in the serial I/O mode and the control example in the CPU reprogramming mode are described below.

(1) Control pin processing example on the system board in serial I/O mode

As shown in Figure 2.13.6, in the serial I/O mode, the contents of the built-in flash memory can be reprogrammed with the microcomputer mounted on board. In the serial I/O mode, the processing example of control pins ($P3_7$, $P6_4$, $P6_6$, $P6_7$, CNV_{SS} and $\overline{RESET}$ pin) is described below.

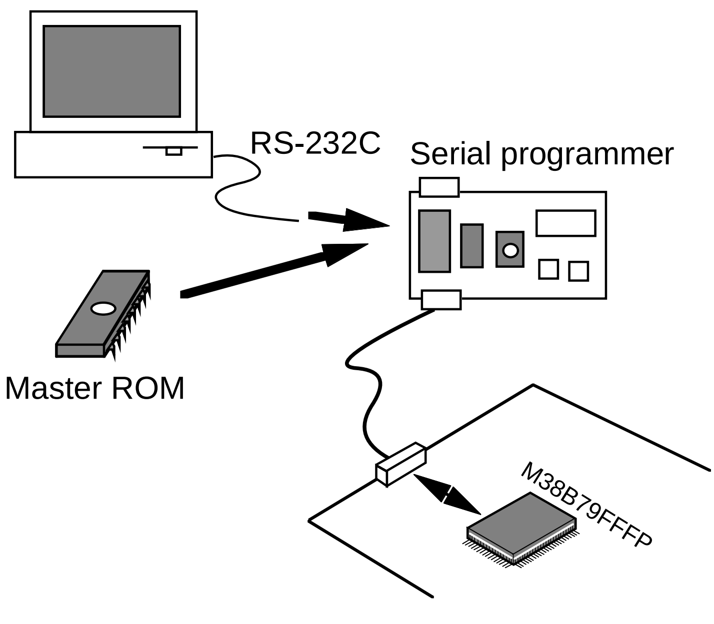


Fig. 2.13.6 Reprogramming example of built-in flash memory by serial I/O mode

① **When control signals are not affected to user system circuit**

When the control signals in the serial I/O mode are not used or not affected to the user system circuit, they can be connected as shown in Figure 2.13.7.

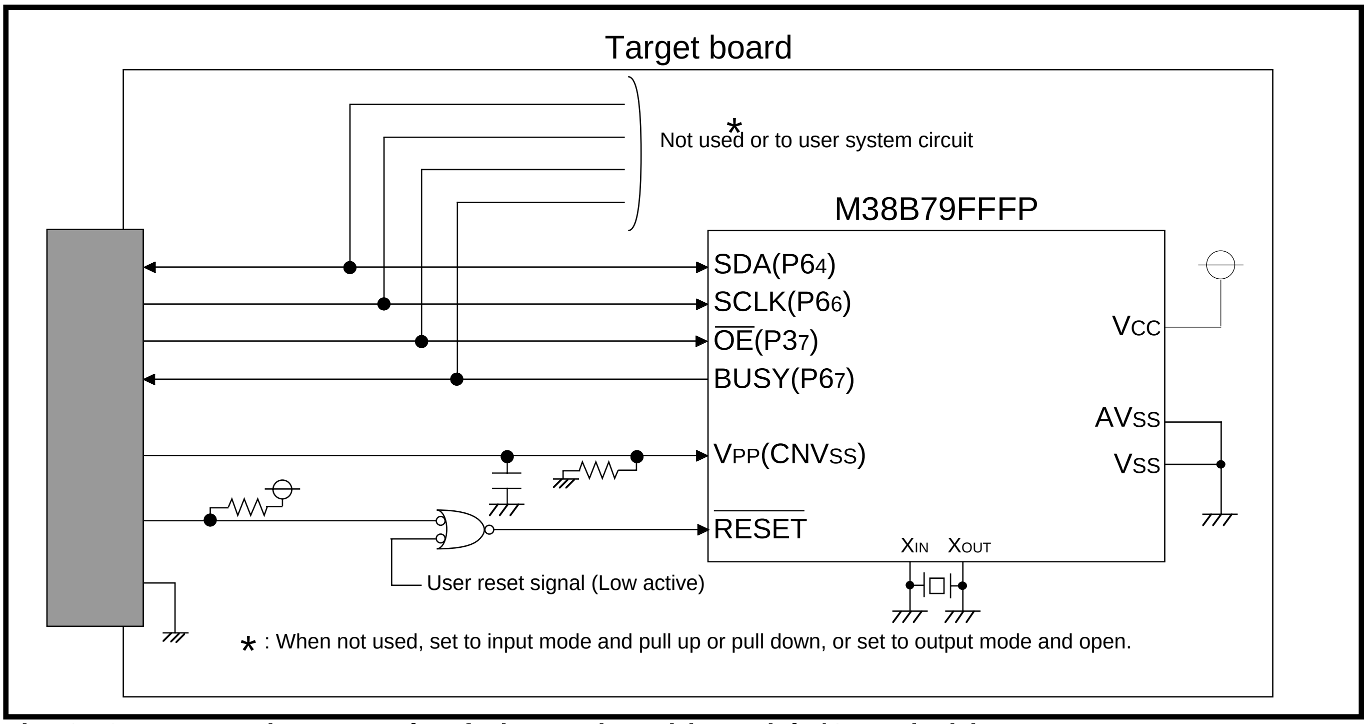


Fig. 2.13.7 Processing example of pins on board in serial I/O mode (1)

② **When control signals are affected to user system circuit-1**

Figure 2.13.8 shows the example that the control signals supplied to the user system circuit are cut-off by a jumper switch in the serial I/O mode.

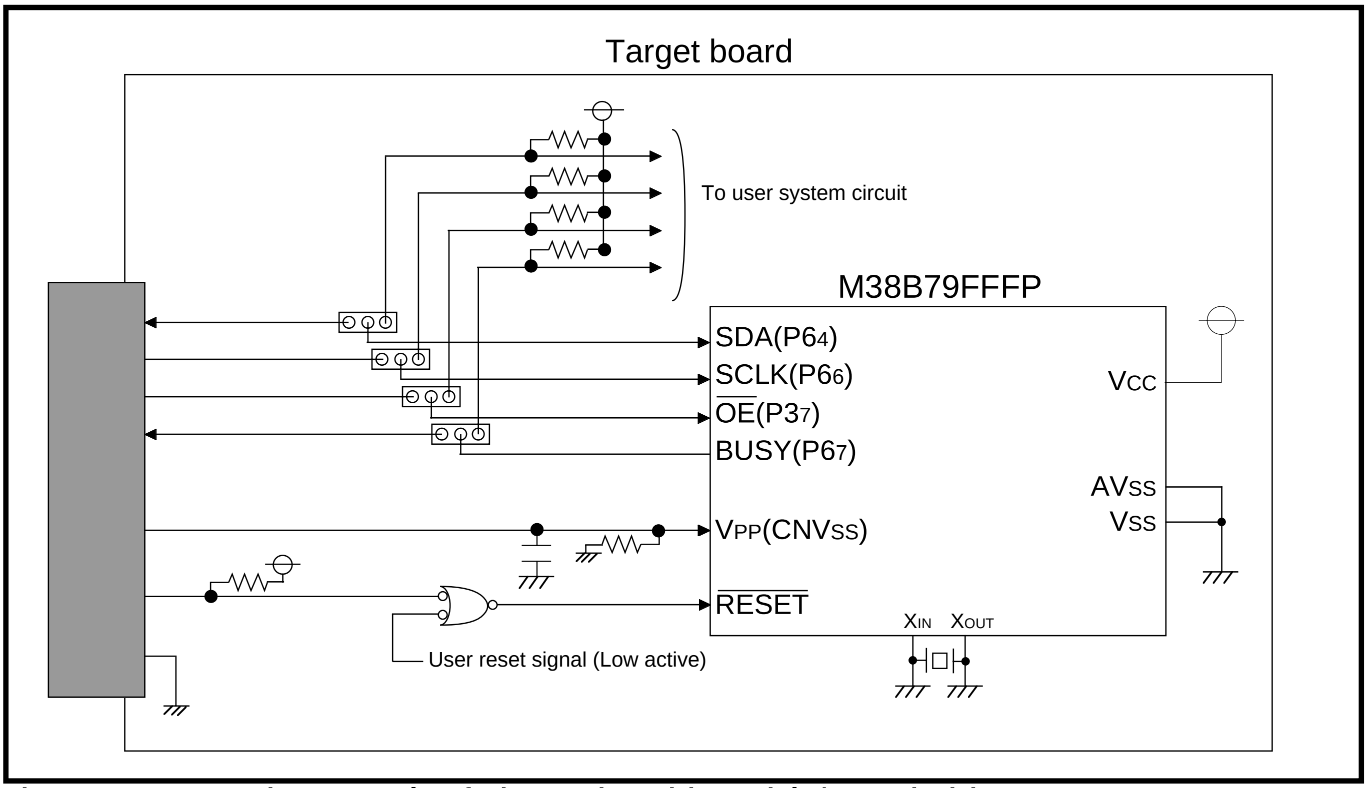


Fig. 2.13.8 Processing example of pins on board in serial I/O mode (2)

③ When control signals are affected to user system circuit-2

Figure 2.13.9 shows the example that the control signals supplied to the user system circuit are cut-off by an analog switch (74HC4066) in the serial I/O mode.

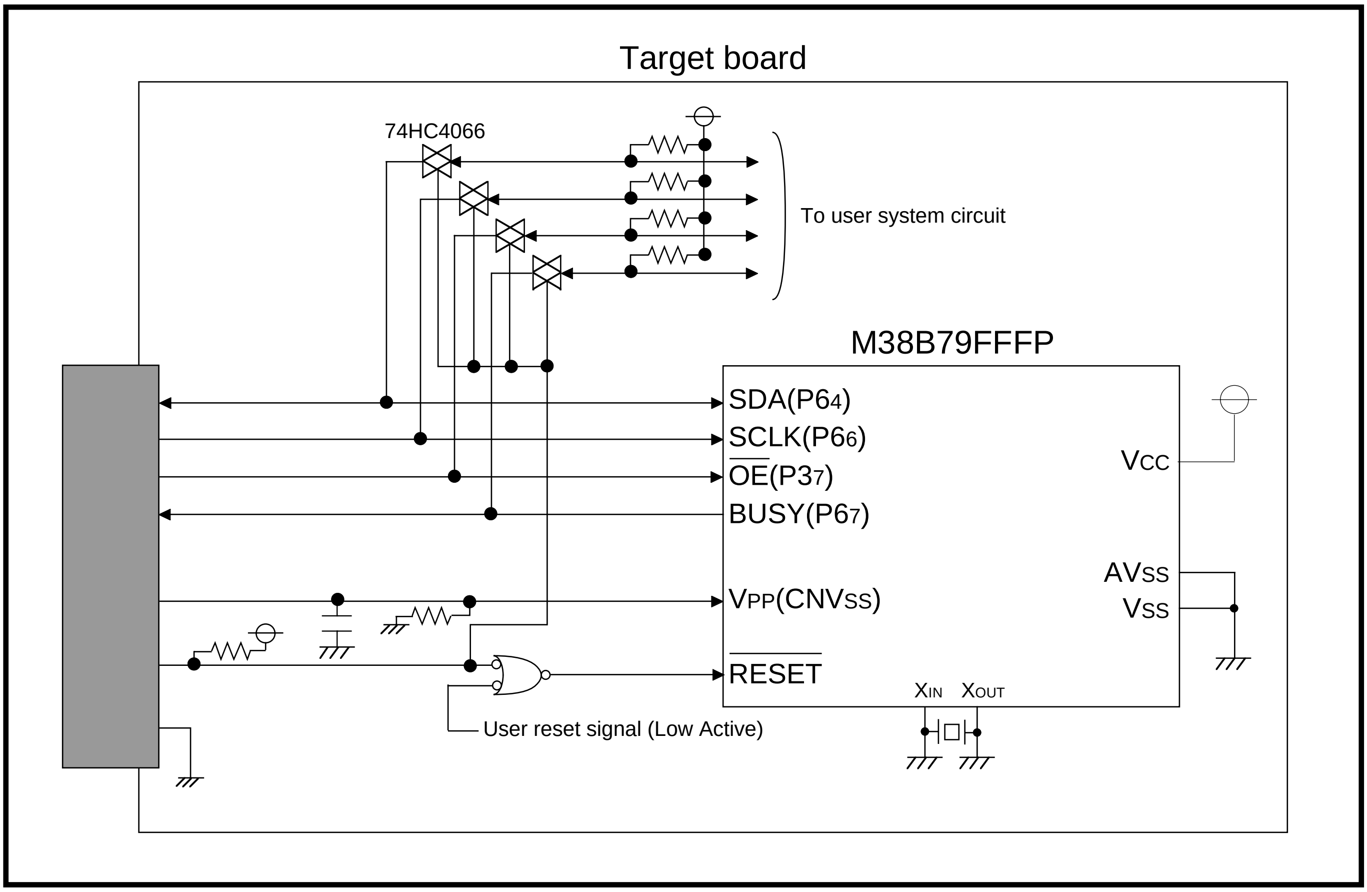


Fig. 2.13.9 Processing example of pins on board in serial I/O mode (3)

(2) Control example in CPU reprogramming mode

In this example, the built-in flash memory is reprogrammed in the CPU reprogramming mode by serial I/O2, receiving the reprogramming data (updated data).

Figure 2.13.10 shows the example for the reprogramming system of the built-in flash memory by the CPU reprogramming mode.

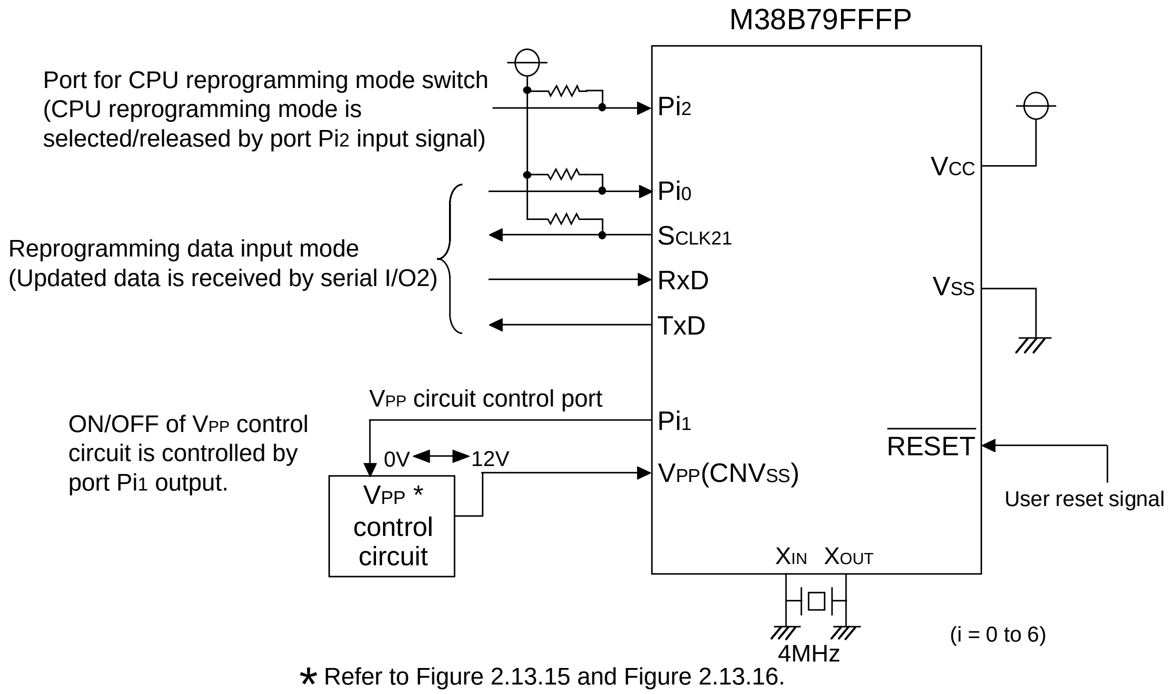
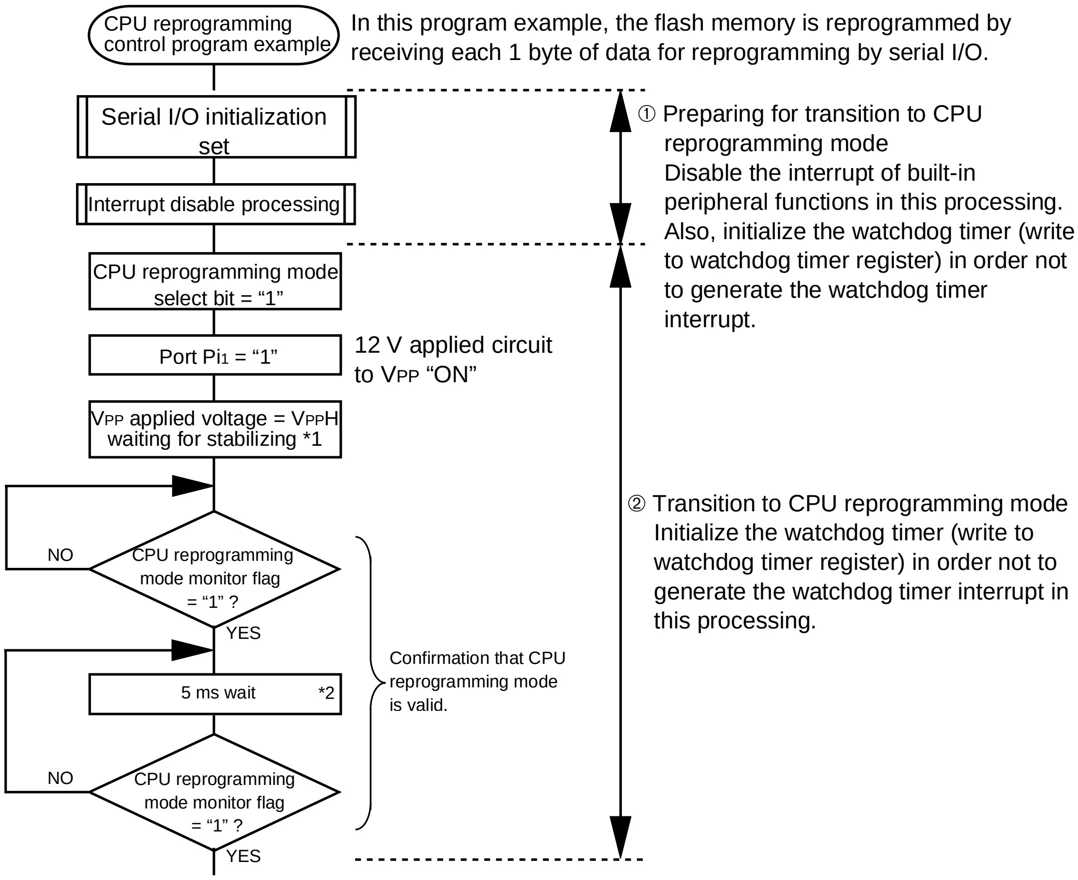


Fig. 2.13.10 Example for reprogramming system of built-in flash memory by CPU reprogramming mode

● **Specifications**

① CPU reprogramming mode is selected/released by the input signal to Pi_2.
② Updated data is received by serial I/O2.
③ The transfer enable state of serial transmit side is judged by "L" level input to Pi_0.
④ V_{PP} control circuit is turned ON/OFF by the output from Pi_1 (refer to Figure 2.13.15 and Figure 2.13.16).

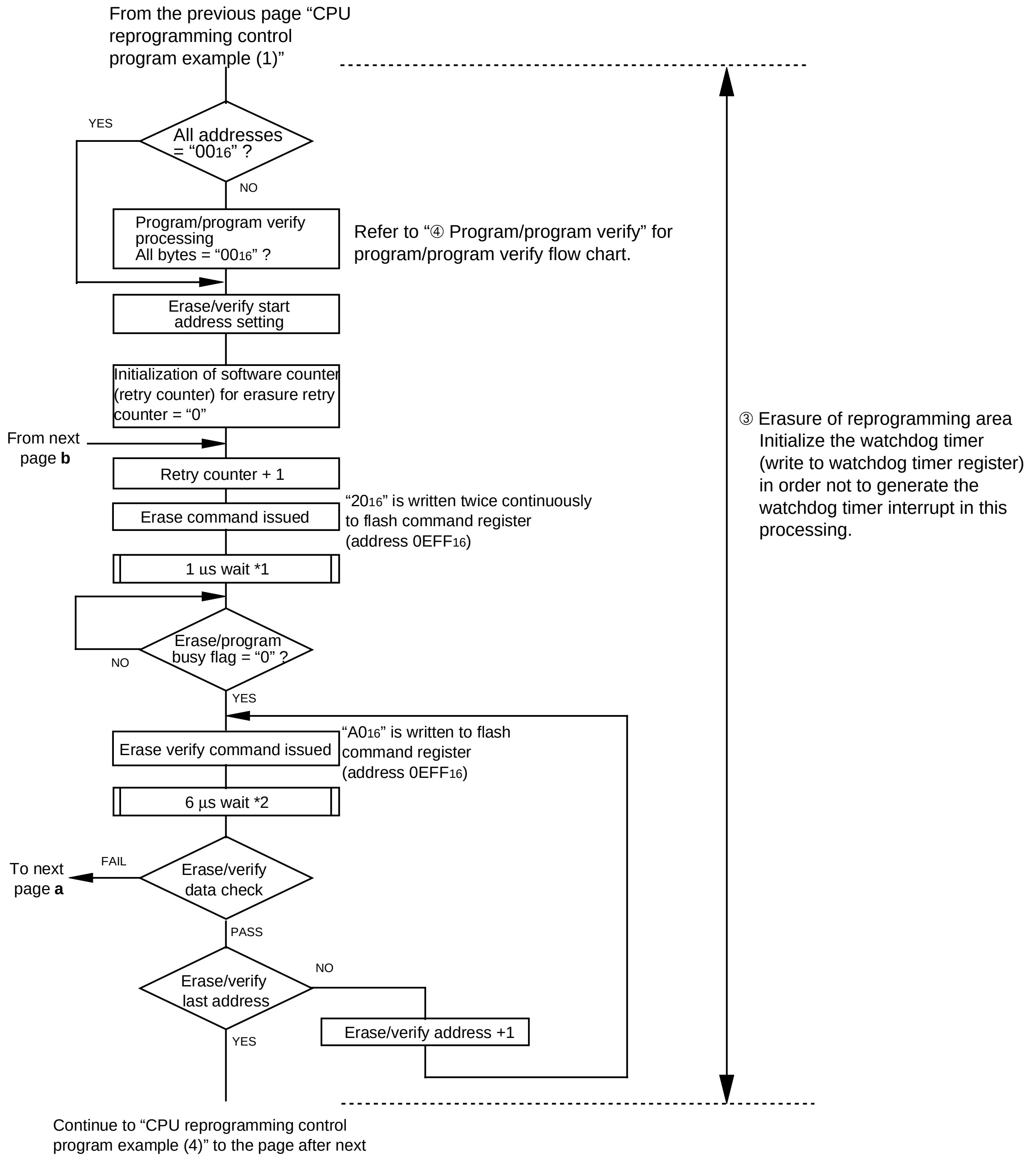
Note: In this example, the following program is transferred to the internal RAM and executed on the internal RAM.



Continue to “CPU reprogramming control program example (2)” to the next page.

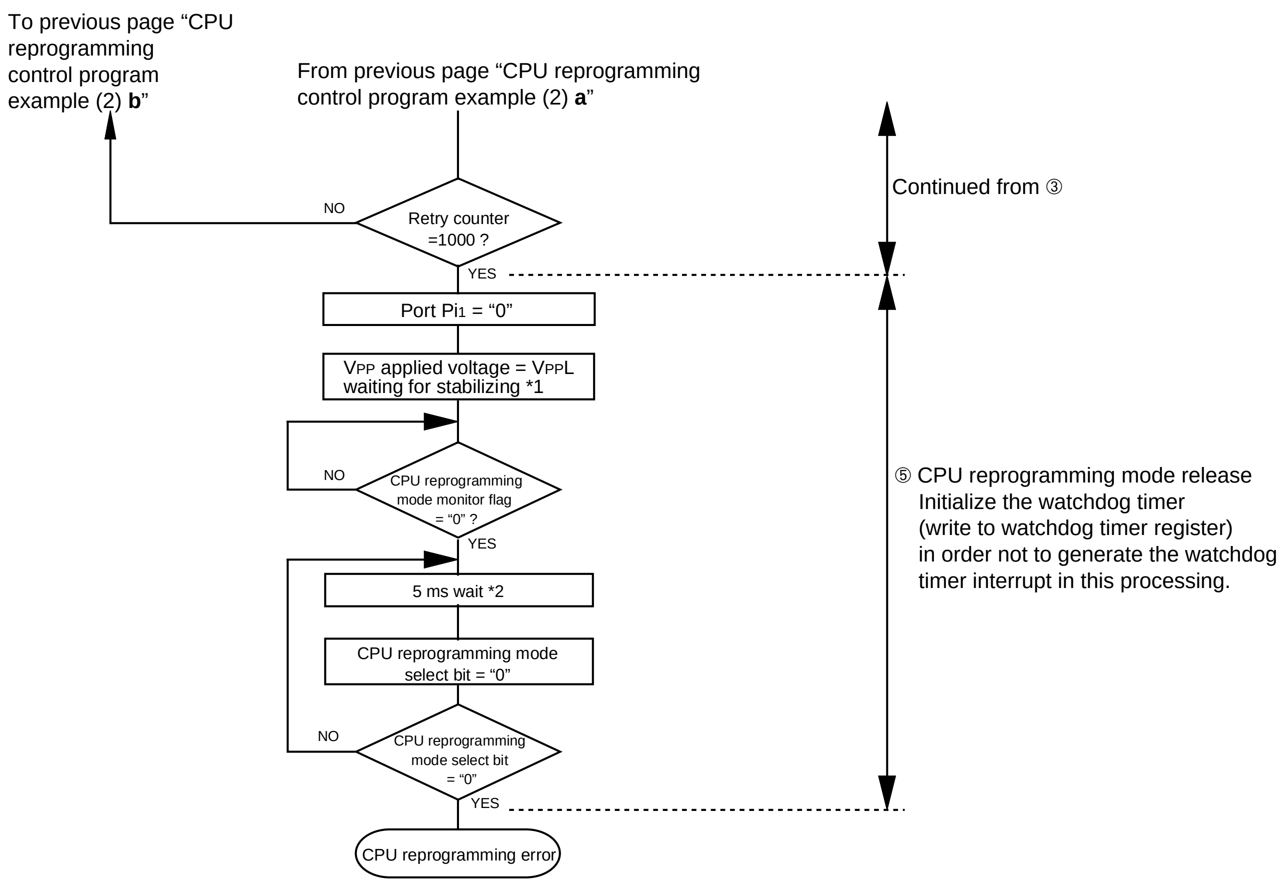
*1: Waiting by software until V_{PP} input voltage is stabilized at V_{PPH} is recommended. (Refer to Figure 2.13.15 and Figure 2.13.16 V_{PP} voltage control timing Ⓐ.)
*2: The wait time depends on V_{PP} control circuit (Refer to Figure 2.13.15 and Figure 2.13.16 V_{PP} voltage control timing Ⓒ.)

Fig. 2.13.11 CPU reprogramming control program example (1)



*1: The wait processing time shown in the flow chart is required regardless of the external clock input frequency.
*2: The wait time depends on the VPP control circuit (refer to Figure 2.13.15 and Figure 2.13.16 VPP voltage control timing Ⓒ).

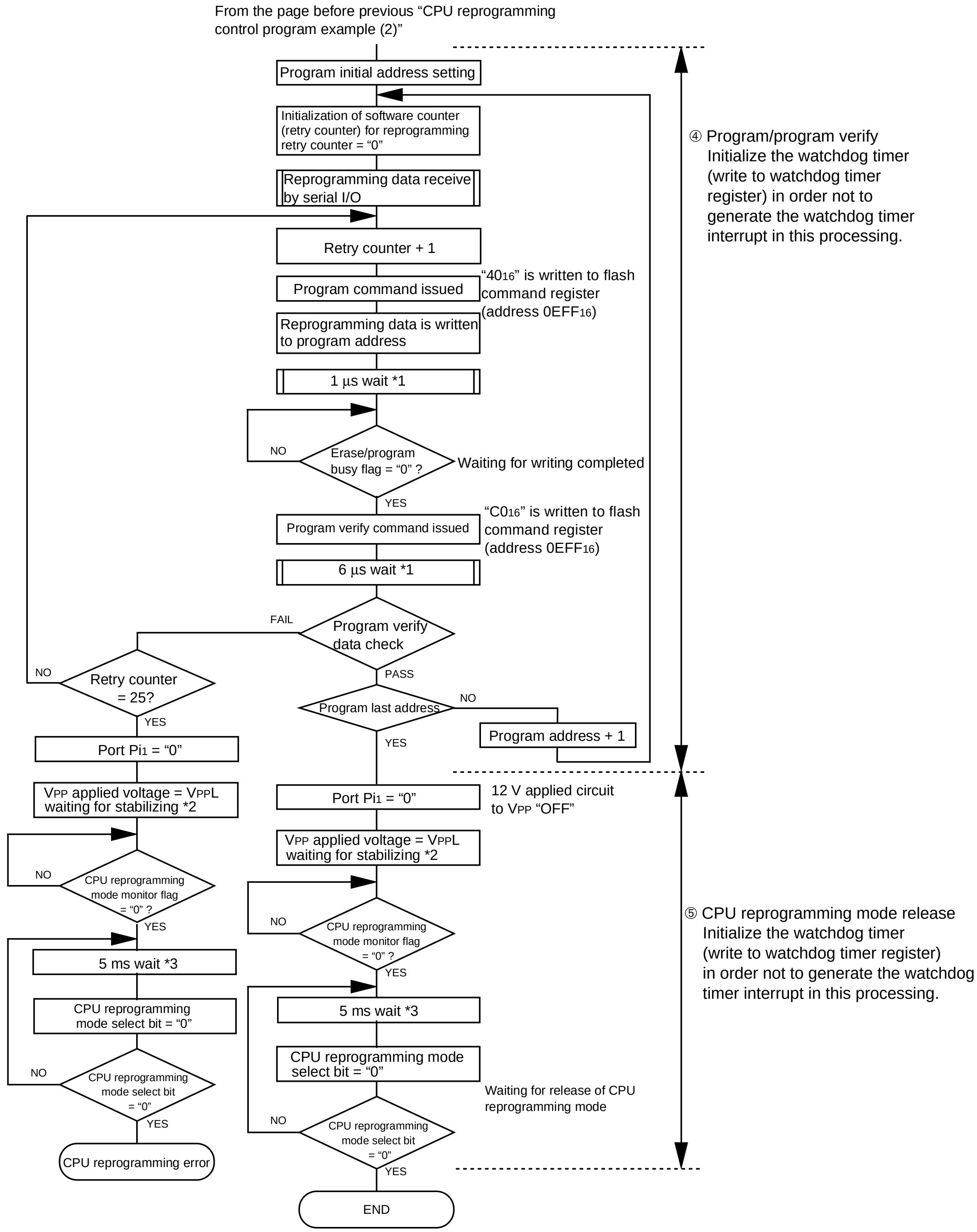
Fig. 2.13.12 CPU reprogramming control program example (2)



*1: Waiting by software until V_{PP} input voltage is stabilized at V_{PPL} is recommended. (Refer to Figure 2.13.15 and Figure 2.13.16 V_{PP} voltage control timing Ⓑ)

*2: The wait time depends on V_{PP} control circuit (Refer to Figure 2.13.15 and Figure 2.13.16 V_{PP} voltage control timing Ⓒ)

Fig. 2.13.13 CPU reprogramming control program example (3)



*1: The wait processing time shown in the flow chart is required regardless of the external clock input frequency.
*2: Waiting by software until V_{PP} input voltage is stabilized at V_{PPL} is recommended. (Refer to Figure 2.13.15 and Figure 2.13.16 V_{PP} voltage control timing Ⓑ.)
*3: The wait time depends on V_{PP} control circuit (Refer to Figure 2.13.15 and Figure 2.13.16 V_{PP} voltage control timing Ⓒ.)

Fig. 2.13.14 CPU reprogramming control program example (4)

● When 12 V voltage is supplied to target system

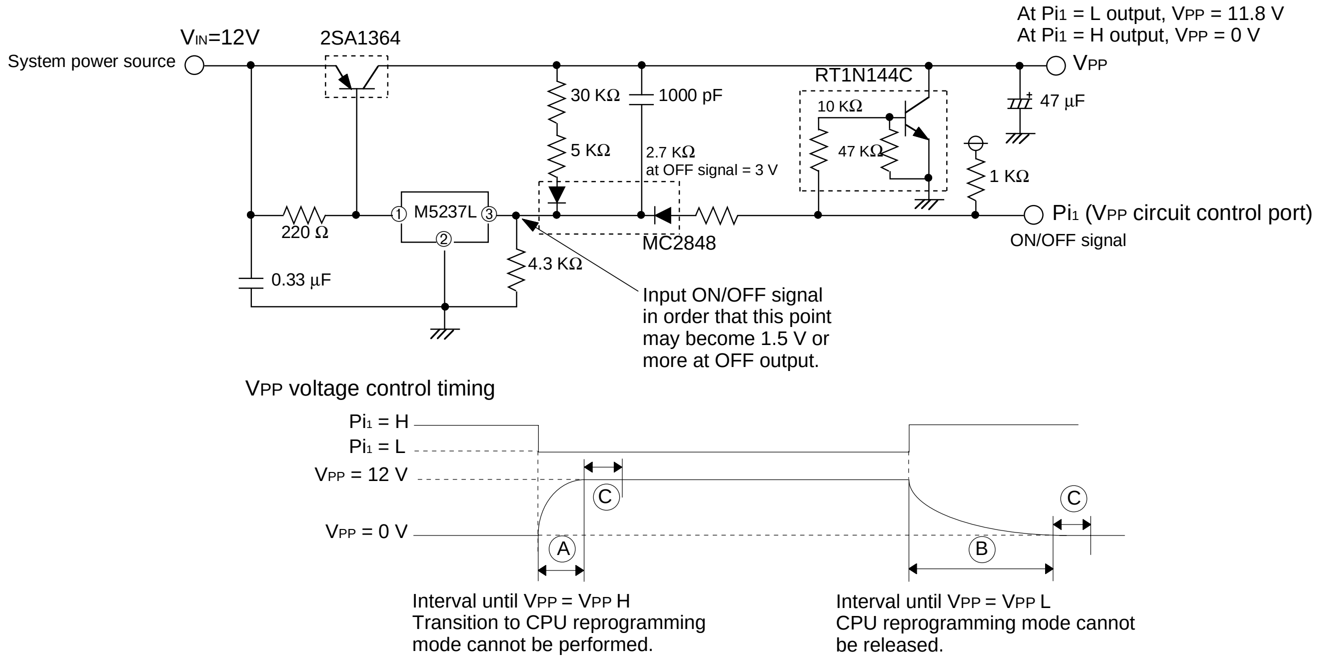


Fig. 2.13.15 V_{PP} control circuit example (1)

● When only 5 V voltage is supplied to target system

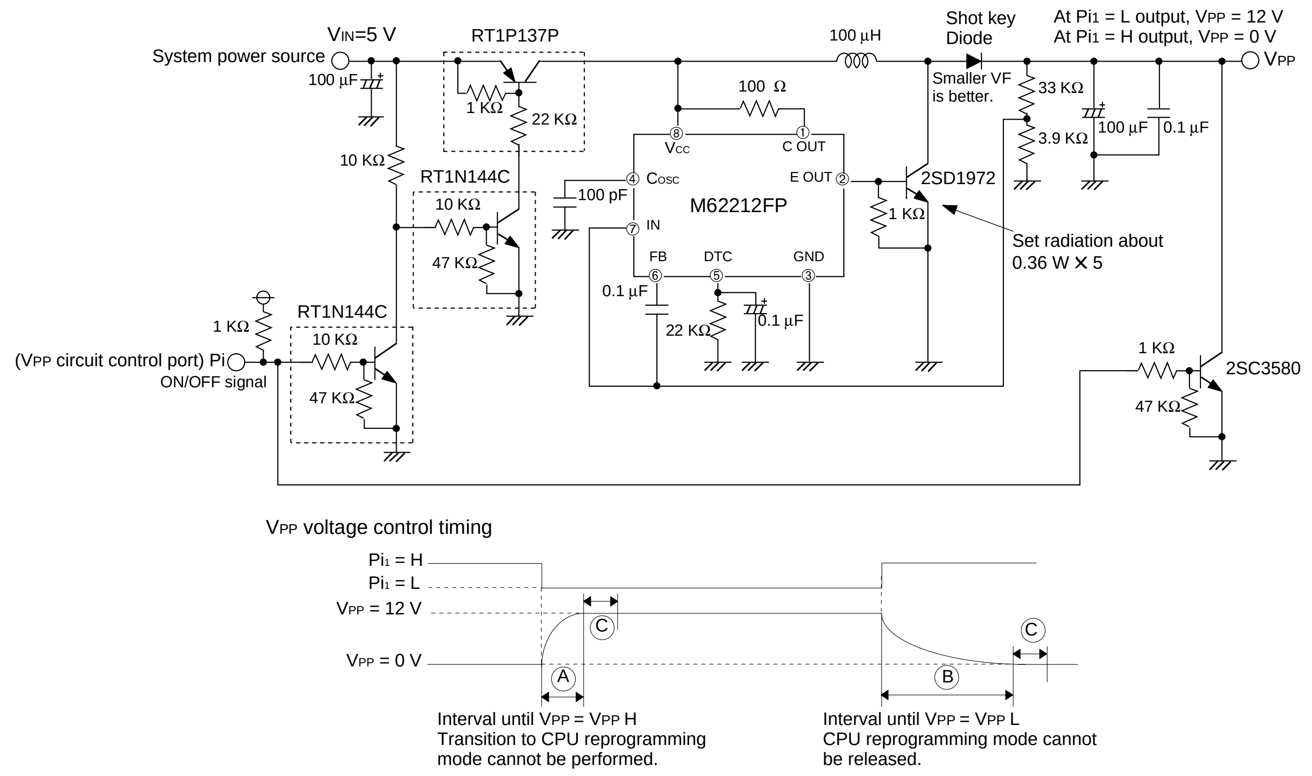


Fig. 2.13.16 V_{PP} control circuit example (2)

2.13.8 Notes on CPU reprogramming mode

(1) Transfer the CPU reprogramming mode control program to the internal RAM before selecting the CPU reprogramming mode, and then, execute it on the internal RAM. Additionally, when the subroutine or stack operation instruction is used in the control program, make sure in order not to destroy the control program transferred to the internal RAM through the stack area.

(2) Be careful of the instruction description (specifying address, and so on) because the CPU reprogramming mode control program is transferred to the internal RAM and executed on the internal RAM.

(3) Write to the watchdog timer control register periodically in order not to generate the watchdog timer interrupt by the CPU reprogramming mode control program (refer to "**2.9 Watchdog timer**").

2.13.9 Notes on flash memory version

The CNV_{SS} pin is connected to the internal memory circuit block by a low-ohmic resistance, since it has the multiplexed function to be a programmable power source pin (V_{PP} pin) as well.
To improve the noise margin, connect the CNV_{SS} pin to V_{SS} through 1 to 10 kΩ resistance.
Even when the wiring of the CNV_{SS} pin of the mask ROM version is connected to Vss through this resistor, that will not affect operation.

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 voltages	All voltages are based on V_{SS}. Output transistors are cut off.	–0.3 to 6.5	V
V_{EE}	Pull-down power source voltages		V_{CC} –45 to V_{CC} +0.3	V
V_I	Input voltage P64–P67, P70–P77, P80–P83, P90–P97, PA0–PA7, PB0–PB6		–0.3 to V_{CC} +0.3	V
V_I	Input voltage P10–P17, P30–P37, P40–P47, P50–P57, P60–P63		V_{CC} –45 to V_{CC} +0.3	V
V_I	Input voltage $\overline{\text{RESET}}$, XIN, CNVSS		–0.3 to V_{CC} +0.3	V
V_I	Input voltage XCIN		–0.3 to V_{CC} +0.3	V
V_O	Output voltage P00–P07, P10–P17, P20–P27, P30–P37, P40–P47, P50–P57, P60–P63		V_{CC} –45 to V_{CC} +0.3	V
V_O	Output voltage P64–P67, P80–P83, P70–P77, P90–P97, PA0–PA7, PB0–PB6, XOUT, XCOUT		–0.3 to V_{CC} +0.3	V
P_d	Power dissipation	T_a = –20 to 65 °C	800	mW
		T_a = 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
(V_{CC} = 4.0 to 5.5 V, T_a = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
V_{CC}	Power source voltage (mask ROM version)	High-speed mode	4.0	5.0	5.5	V
		Middle/Low-speed mode	2.7	5.0	5.5	V
V_{CC}	Power source voltage (flash memory version)		4.0	5.0	5.5	V
V_{SS}	Power source voltage			0		V
V_{EE}	Pull-down power source voltage		Vcc –43		V_{CC}	V
V_{REF}	Analog reference voltage	when A-D converter is used	2.0		V_{CC}	V
		when D-A converter is used	3.0		V_{CC}	V
AV_{SS}	Analog power source voltage			0		V
V_{IA}	Analog input voltage AN0–AN15		0		V_{CC}	V
V_{IH}	“H” input voltage P70–P77, P80–P83, P90–P97, PA0–PA7, PB0–PB6		0.75V_{CC}		V_{CC}	V
V_{IH}	“H” input voltage P64–P67		0.4V_{CC}		V_{CC}	V
V_{IH}	“H” input voltage P10–P17, P30–P37, P40–P47, P50–P57, P60–P63		0.52V_{CC}		V_{CC}	V
V_{IH}	“H” input voltage RxD, SCLK21, SCLK22		0.8V_{CC}		V_{CC}	V
V_{IH}	“H” input voltage XIN, XCIN, $\overline{\text{RESET}}$, CNVss		0.8V_{CC}		V_{CC}	V
V_{IL}	“L” input voltage P70–P77, P80–P83, P90–P97, PA0–PA7, PB0–PB6		0		0.25V_{CC}	V
V_{IL}	“L” input voltage P64–P67		0		0.16V_{CC}	V
V_{IL}	“L” input voltage P10–P17, P30–P37, P40–P47, P50–P57, P60–P63		0		0.2V_{CC}	V
V_{IL}	“L” input voltage RxD, SCLK21, SCLK22		0		0.2V_{CC}	V
V_{IL}	“L” input voltage XIN, XCIN, $\overline{\text{RESET}}$, CNVss		0		0.2V_{CC}	V

Table 3.1.3 Recommended operating conditions
(V_{CC} = 4.0 to 5.5 V, T_a = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$\Sigma I_{OH(peak)}$	"H" total peak output current **(Note 1)** $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P4_0$–$P4_7$, $P5_0$–$P5_7$, $P6_0$–$P6_7$, $P7_0$–$P7_7$			–240	mA
$\Sigma I_{OH(peak)}$	"H" total peak output current **(Note 1)** $P8_0$–$P8_3$, $P9_0$–$P9_7$, PA_0–PA_7, PB_0–PB_6			–60	mA
$\Sigma I_{OL(peak)}$	"L" total peak output current **(Note 1)** $P6_4$–$P6_7$, $P7_0$–$P7_7$			100	mA
$\Sigma I_{OL(peak)}$	"L" total peak output current **(Note 1)** $P8_0$–$P8_3$, $P9_0$–$P9_7$, PA_0–PA_7, PB_0–PB_6			60	mA
$\Sigma I_{OH(avg)}$	"H" total average output current **(Note 1)** $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P4_0$–$P4_7$, $P5_0$–$P5_7$, $P6_0$–$P6_3$			–120	mA
$\Sigma I_{OH(avg)}$	"H" total average output current **(Note 1)** $P6_4$–$P6_7$, $P7_0$–$P7_7$, $P8_0$–$P8_3$, $P9_0$–$P9_7$, PA_0–PA_7, PB_0–PB_6			–30	mA
$\Sigma I_{OL(avg)}$	"L" total average output current **(Note 1)** $P6_4$–$P6_7$, $P7_0$–$P7_7$, $P8_0$–$P8_3$, $P9_0$–$P9_7$, PA_0–PA_7, PB_0–PB_6			50	mA
$I_{OH(peak)}$	"H" peak output current **(Note 2)** $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P4_0$–$P4_7$, $P5_0$–$P5_7$, $P6_0$–$P6_3$			–40	mA
$I_{OH(peak)}$	"H" peak output current **(Note 2)** $P6_4$–$P6_7$, $P7_0$–$P7_7$, $P8_0$–$P8_3$, $P9_0$–$P9_7$, PA_0–PA_7, PB_0–PB_6			–10	mA
$I_{OL(peak)}$	"L" peak output current **(Note 2)** $P6_4$–$P6_7$, $P7_0$–$P7_7$, $P8_0$–$P8_3$, $P9_0$–$P9_7$, PA_0–PA_7, PB_0–PB_6			10	mA
$I_{OH(avg)}$	"H" average output current **(Note 3)** $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P4_0$–$P4_7$, $P5_0$–$P5_7$, $P6_0$–$P6_3$			–18	mA
$I_{OH(avg)}$	"H" average output current **(Note 3)** $P6_4$–$P6_7$, $P7_0$–$P7_7$, $P8_0$–$P8_3$, $P9_0$–$P9_7$, PA_0–PA_7, PB_0–PB_6			–5	mA
$I_{OL(avg)}$	"L" average output current **(Note 3)** $P6_4$–$P6_7$, $P7_0$–$P7_7$, $P8_0$–$P8_3$, $P9_0$–$P9_7$, PA_0–PA_7, PB_0–PB_6			5	mA
f(CNTR)	Clock input frequency for timers 2, 4, and X (duty cycle 50 %)			250	kHz
$f(X_{IN})$	Main clock input oscillation frequency **(Note 4)**			4.2	MHz
$f(X_{CIN})$	Sub-clock input oscillation frequency **(Notes 4, 5)**		32.768	50	kHz

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) are average value measured over 100 ms.
4: When the oscillation frequency has a duty cycle of 50%.
5: When using the microcomputer in low-speed mode, set the sub-clock input oscillation frequency on condition that $f(X_{CIN}) < f(X_{IN})/3$.

APPENDIX

3.1 Electrical characteristics

3.1.3 Electrical characteristics

Table 3.1.4 Electrical characteristics
(V_{CC} = 4.0 to 5.5 V, T_a = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min.	Limits Typ.	Limits Max.	Unit
V_{OH}	“H” output voltage P0$_0$–P0$_7$, P1$_0$–P1$_7$, P2$_0$–P2$_7$, P3$_0$–P3$_7$, P4$_0$–P4$_7$, P5$_0$-P5$_7$, P6$_0$–P6$_3$	I_{OH} = –18 mA	V_{CC}–2.0			V
V_{OH}	“H” output voltage P6$_4$–P6$_7$, P7$_0$–P7$_7$, P8$_0$–P8$_3$, P9$_0$–P9$_7$, PA$_0$–PA$_7$, PB$_0$–PB$_6$	I_{OH} = –10 mA	V_{CC}–2.0			V
V_{OL}	“L” output voltage P6$_4$–P6$_7$, P7$_0$–P7$_7$, P8$_0$–P8$_3$, P9$_0$–P9$_7$, PA$_0$–PA$_7$, PB$_0$–PB$_6$	I_{OL} = 10 mA			2.0	V
V_{T+}–V_{T-}	Hysteresis RxD, S_{CLK21}, S_{CLK22}, S_{RDY1}, P7$_0$–P7$_3$, P7$_7$, P8$_2$–P8$_3$, P9$_0$–P9$_2$, PB$_0$, PB$_2$, PB$_4$–PB$_6$			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 P6$_4$–P6$_7$, P7$_0$–P7$_7$, P8$_0$–P8$_3$, P9$_0$–P9$_7$, PA$_0$–PA$_7$, PB$_0$–PB$_6$	V_I = V_{CC}			5.0	µA
I_{IH}	“H” input current P1$_0$–P1$_7$, P3$_0$–P3$_7$, P4$_0$–P4$_7$, P5$_0$-P5$_7$, P6$_0$–P6$_3$ **(Note)**	V_I = V_{CC}			5.0	µA
I_{IH}	“H” input current $\overline{RESET}$, CNVss, 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 P6$_4$–P6$_7$, P7$_0$–P7$_7$, P8$_0$–P8$_3$, P9$_0$–P9$_7$, PA$_0$–PA$_7$, PB$_0$–PB$_6$	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 P1$_0$–P1$_7$, P3$_0$–P3$_7$, P4$_0$–P4$_7$, P5$_0$–P5$_7$, P6$_0$–P6$_3$ **(Note)**	V_I = V_{SS}			–5.0	µA
I_{IL}	“L” input current $\overline{RESET}$, CNVss, X_{CIN}	V_I = V_{SS}			–5.0	µA
I_{IL}	“L” input current X_{IN}	V_I = V_{SS}		–4.0		µA

Note: Except when reading ports P1, P3, P4, P5 or P6.

Table 3.1.5 Electrical characteristics

(V_{CC} = 4.0 to 5.5 V, V_{SS} = 0 V, T_a = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
I_{LOAD}	Output load current $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, ($P4_0$–$P4_7$, $P5_0$–$P5_7$, $P6_0$–$P6_3$ at option)	V_{EE} = V_{CC}–43 V, V_{OL} =V_{CC} Output transistors “off”	400	600	900	µA
I_{LEAK}	Output leak current $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P4_0$–$P4_7$, $P5_0$–$P5_7$, $P6_0$–$P6_3$	V_{EE} = V_{CC}–43 V, V_{OL} =V_{CC}–43 V Output transistors “off”			–10	µA
I_{READH}	“H” read current $P1_0$–$P1_7$, $P3_0$–$P3_7$, $P4_0$–$P4_7$, $P5_0$–$P5_7$, $P6_0$–$P6_3$	V_I = 5 V		1		µA
V_{RAM}	RAM hold voltage	When clock is stopped	2		5.5	V
I_{CC}	Power source current	High-speed mode, Vcc = 5 V, f(X_{IN}) = 4.2 MHz f(X_{CIN}) = 32.768 kHz Output transistors “off”		7.0	15	mA
		High-speed mode, Vcc = 5 V, f(X_{IN}) = 4.2 MHz (in WIT state) f(X_{CIN}) = 32.768 kHz Output transistors “off”		1		mA
		Middle-speed mode, Vcc = 5 V, f(X_{IN}) = 4.2 MHz f(X_{CIN}) = stopped Output transistors “off”		3		mA
		Middle-speed mode, Vcc = 5 V, f(X_{IN}) = 4.2 MHz (in WIT state) f(X_{CIN}) = stopped Output transistors “off”		1		mA
		Low-speed mode, Vcc = 3 V, f(X_{IN}) = stopped f(X_{CIN}) = 32.768 kHz Output transistors “off”		20	55	µA
		Low-speed mode, Vcc = 3 V, f(X_{IN}) = stopped f(X_{CIN}) = 32.768 kHz (in WIT state) Output transistors “off”		8	20	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
		All oscillation stopped (in STP state) Output transistors “off”; Ta = 85 °C			10	µA

3.1 Electrical characteristics

3.1.4 A-D converter charactristics

Table 3.1.6 A-D converter characteristics

(V_{CC} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = 0 V, T_a = –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)
I_{VREF}	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

3.1.5 D-A converter charactristics

Table 3.1.7 D-A converter characteristics

(V_{CC} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = 0 V, V_{REF} = 3.0 to Vcc, T_a = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
–	Resolution				8	Bits
–	Absolute accuracy (excluding quantization error)	V_{CC} = 4.0–5.5 V			1.0	%
		V_{CC} = 3.0–5.5 V			2.5	%
t_{su}	Setting time				3	µs
R_O	Output resistor		1	2.5	4	kΩ
I_{VREF}	Reference power source input current **(Note)**				3.2	mA

Note: Except ladder resistor for A-D converter

3.1.6 Timing requirements and switching characteristics

Table 3.1.8 Timing requirements (1)

(V_{CC} = 4.0 to 5.5 V, V_{SS} = 0 V, T_a = –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
$t_C(X_{IN})$	Main clock input cycle time (X_{IN} input)	238			ns
$t_{WH}(X_{IN})$	Main clock input "H" pulse width	60			ns
$t_{WL}(X_{IN})$	Main clock input "L" pulse width	60			ns
$t_C(X_{CIN})$	Sub-clock input cycle time (X_{CIN} input)	20			µs
$t_{WH}(X_{CIN})$	Sub-clock input "H" pulse width	5.0			µs
$t_{WL}(X_{CIN})$	Sub-clock input "L" pulse width	5.0			µs
$t_C(CNTR)$	$CNTR_0$–$CNTR_2$ input cycle time	4.0			µs
$t_{WH}(CNTR)$	$CNTR_0$–$CNTR_2$ input "H" pulse width	1.6			µs
$t_{WL}(CNTR)$	$CNTR_0$–$CNTR_2$ input "L" pulse width	1.6			µs
$t_{WH}(INT)$	INT_0–INT_4 input "H" pulse width (INT_2 when noise filter is not used) **(Note 1)**	80			ns
$t_{WL}(INT)$	INT_0–INT_4 input "L" pulse width (INT_2 when noise filter is not used) **(Note 1)**	80			ns
$t_{WH}(INT_2)$	INT_2 input "H" pulse width (when noise filter is used) **(Notes 1, 2)**	3			CLKs
$t_{WL}(INT_2)$	INT_2 input "L" pulse width (when noise filter is used) **(Notes 1, 2)**	3			CLKs
$t_C(S_{CLK1})$	Serial I/O1 clock input cycle time	950			ns
$t_{WH}(S_{CLK1})$	Serial I/O1 clock input "H" pulse width	400			ns
$t_{WL}(S_{CLK1})$	Serial I/O1 clock input "L" pulse width	400			ns
$t_{su}(S_{IN1}-S_{CLK1})$	Serial I/O1 input setup time	200			ns
$t_h(S_{CLK1}-S_{IN1})$	Serial I/O1 input hold time	200			ns
$t_C(S_{CLK2})$	Serial I/O2 clock input cycle time	800			ns
$t_{WH}(S_{CLK2})$	Serial I/O2 clock input "H" pulse width	370			ns
$t_{WL}(S_{CLK2})$	Serial I/O2 clock input "L" pulse width	370			ns
$t_{su}(RxD-S_{CLK2})$	Serial I/O2 input setup time	220			ns
$t_h(S_{CLK2}-RxD)$	Serial I/O2 input hold time	100			ns
$t_C(S_{CLK3})$	Serial I/O3 clock input cycle time	1000			ns
$t_{WH}(S_{CLK3})$	Serial I/O3 clock input "H" pulse width	400			ns
$t_{WL}(S_{CLK3})$	Serial I/O3 clock input "L" pulse width	400			ns
$t_{su}(S_{IN3}-S_{CLK3})$	Serial I/O3 input setup time	200			ns
$t_h(S_{CLK3}-S_{IN3})$	Serial I/O3 input hold time	200			ns

Notes 1: IIDCON2, IIDCON3 = "00" when noise filter is not used
IIDCON2, IIDCON3 = "01" or "10" when noise filter is used
2: Unit indicates sample clock number of noise filter.

Table 3.1.9 Switching characteristics

(V_{CC} = 4.0 to 5.5 V, V_{SS} = 0 V, T_a = –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 (SCLK1-SOUT1)	Serial I/O1 output delay time **(Note 1)**				200	ns
t_V (SCLK1-SOUT1)	Serial I/O1 output valid time **(Note 1)**		0			ns
t_d (SCLK2-TxD)	Serial I/O2 output delay time **(Note 2)**				140	ns
t_V (SCLK2-TxD)	Serial I/O2 output valid time **(Note 2)**		–30			ns
t_d (SCLK3-SOUT3)	Serial I/O3 output delay time **(Note 3)**				200	ns
t_V (SCLK3-SOUT3)	Serial I/O3 output valid time **(Note 3)**		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-breakdodwn-voltage output rising time **(Note 4)**	C_L = 100 pF V_{EE} = Vcc –43 V		55		ns
t_r (Pch–weak)	P-channel high-breakdodwn-voltage output rising time **(Note 5)**	C_L = 100 pF V_{EE} = Vcc –43 V		1.8		µs

Notes 1: When the PB5/SOUT1 P-channel output disable bit of the serial I/O1 control register (bit 7 of address $001A_{16}$) is "0".
2: When the P65/TxD P-channel output disable bit of the UART control register (bit 4 of address 0038_{16}) is "0".
3: When the P91/SOUT3 P-channel output disable bit of the serial I/O3 control register (bit 7 of address $0EEC_{16}$) is "0".
4: When the high-breakdown voltage port drivability selection bit of the FLDC mode register (bit 7 of address $0EF4_{16}$) is "0".
5: When the high-breakdown voltage port drivability selection bit of the FLDC mode register (bit 7 of address $0EF4_{16}$) is "1".

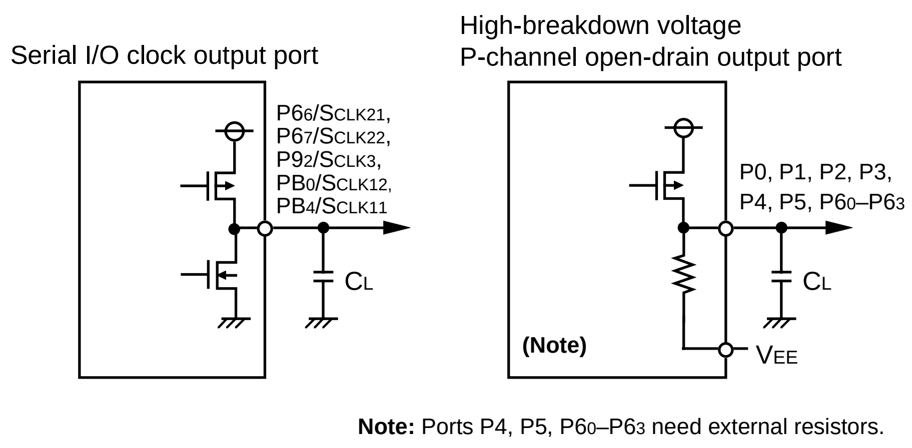


Fig. 3.1.1 Circuit for measuring output switching characteristics

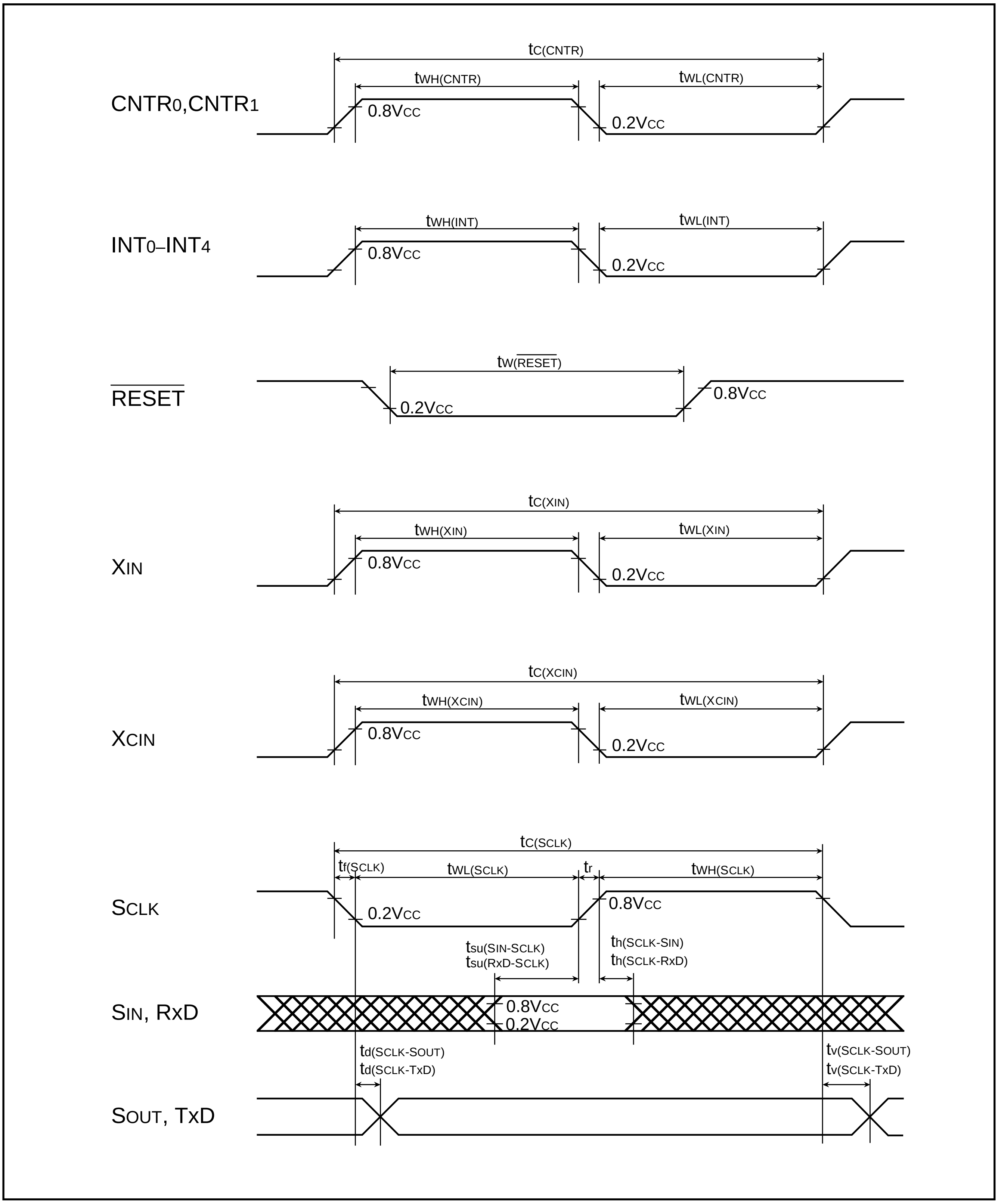


Fig. 3.1.2 Timing diagram

3.2 Standard characteristics

Standard characteristics described below are just examples. These are NOT guaranteed. For rated values, refer to "**3.1 Electrical characteristics**".

3.2.1 Power source current standard characteristics

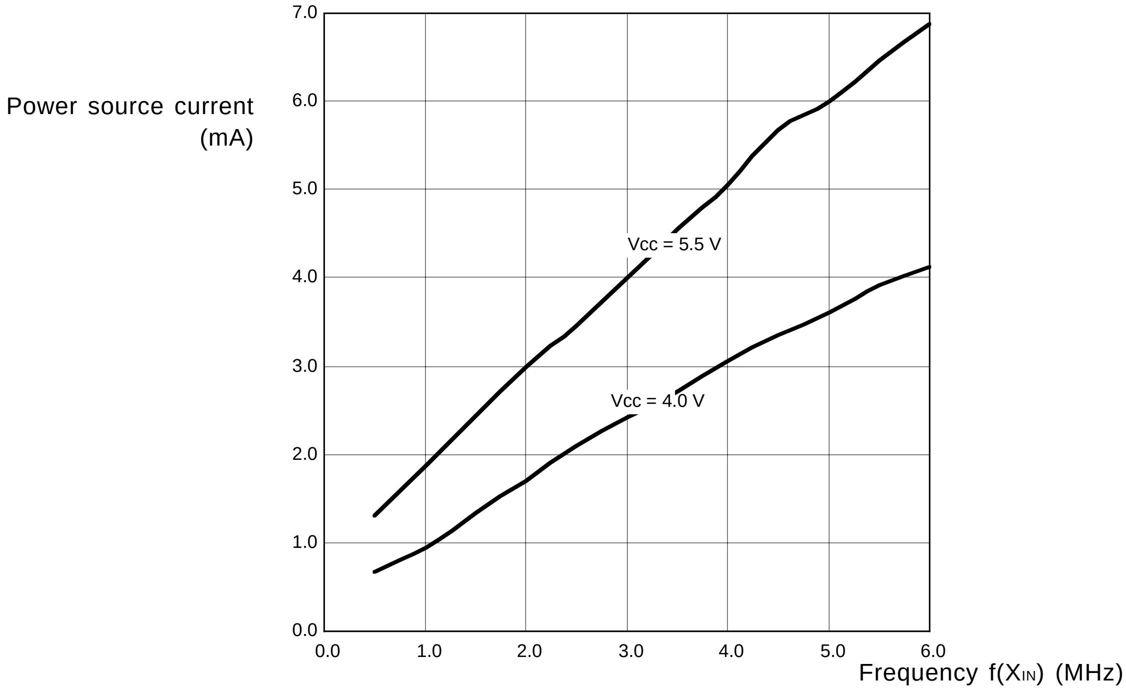


Fig. 3.2.1 Power source current standard characteristics

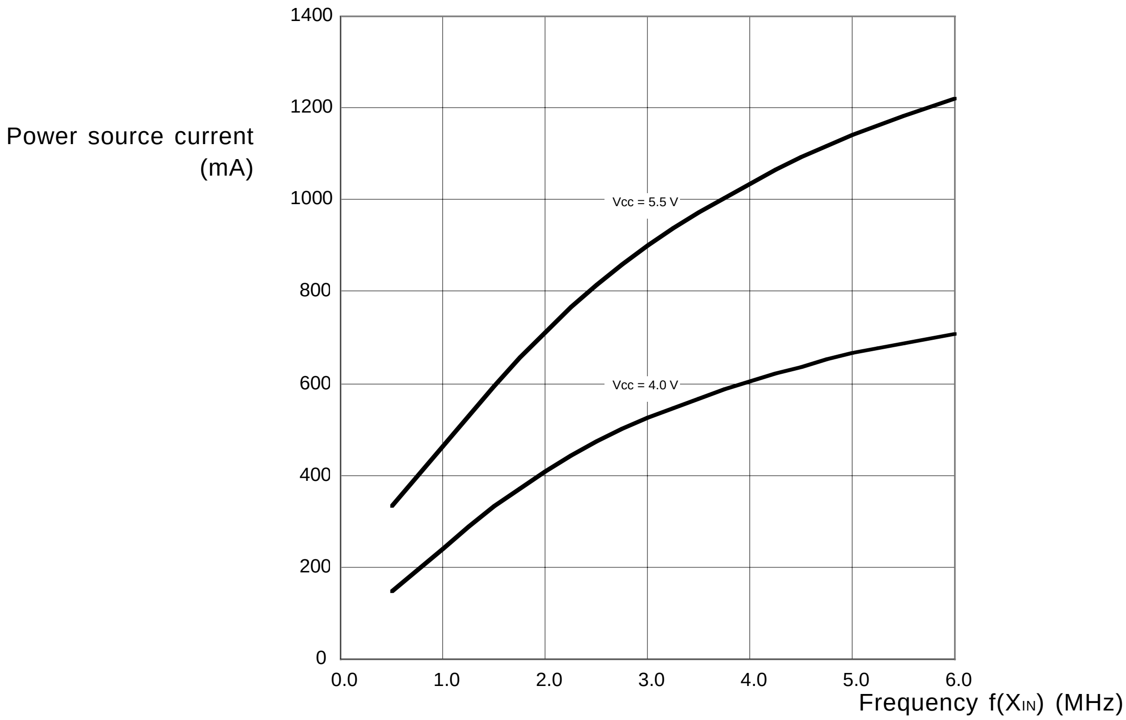


Fig. 3.2.2 Power source current standard characteristics (in wait mode)

3.2.2 Port standard characteristics

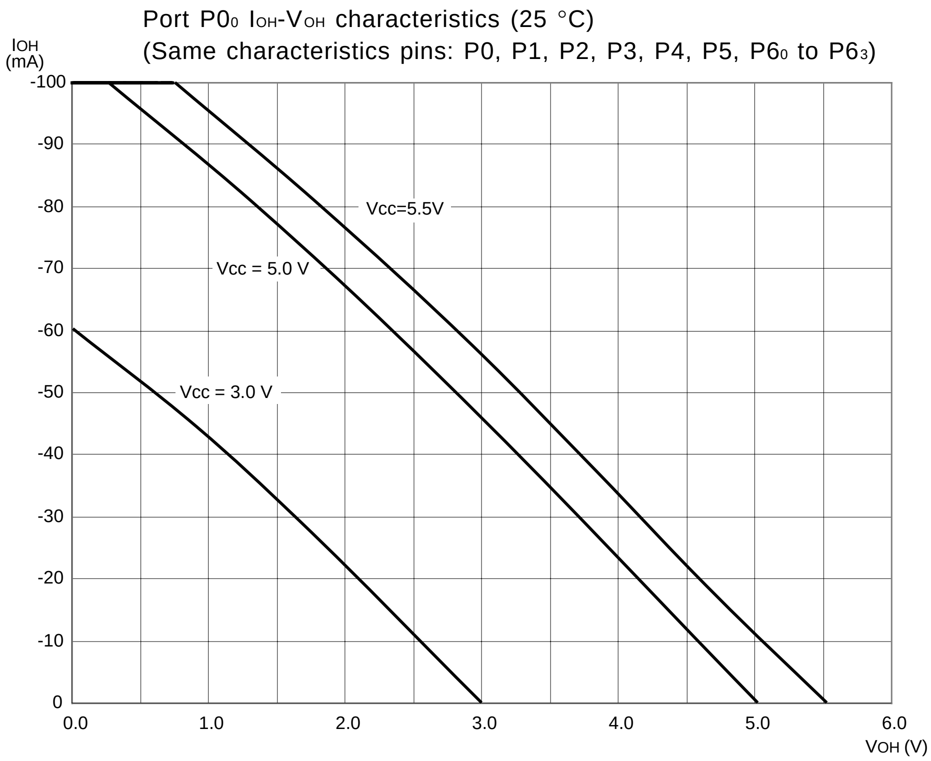


Fig. 3.2.3 High-breakdown P-channel open-drain output port characteristics (25 °C)

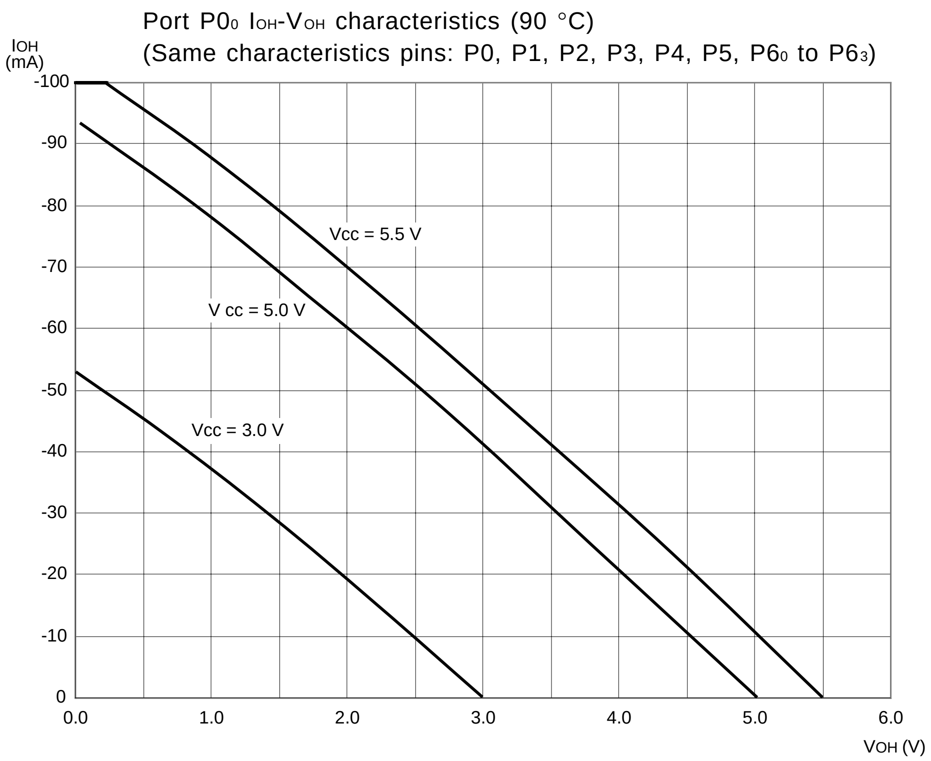


Fig. 3.2.4 High-breakdown P-channel open-drain output port characteristics (90 °C)

3.2 Standard characteristics

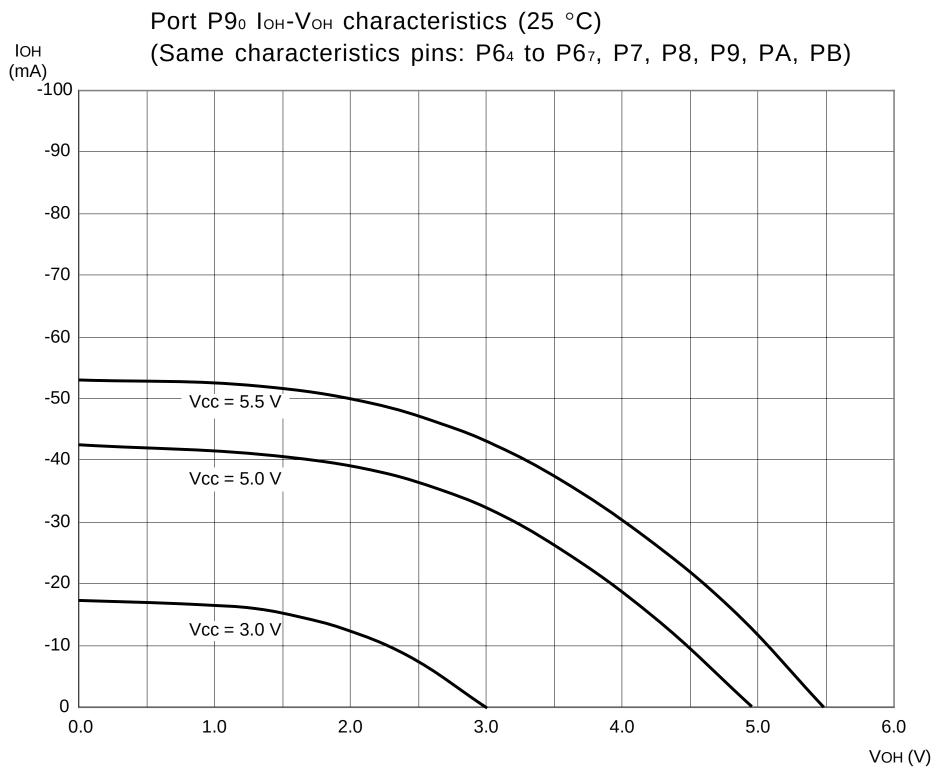


Fig. 3.2.5 CMOS output port P-channel side characteristics (25 °C)

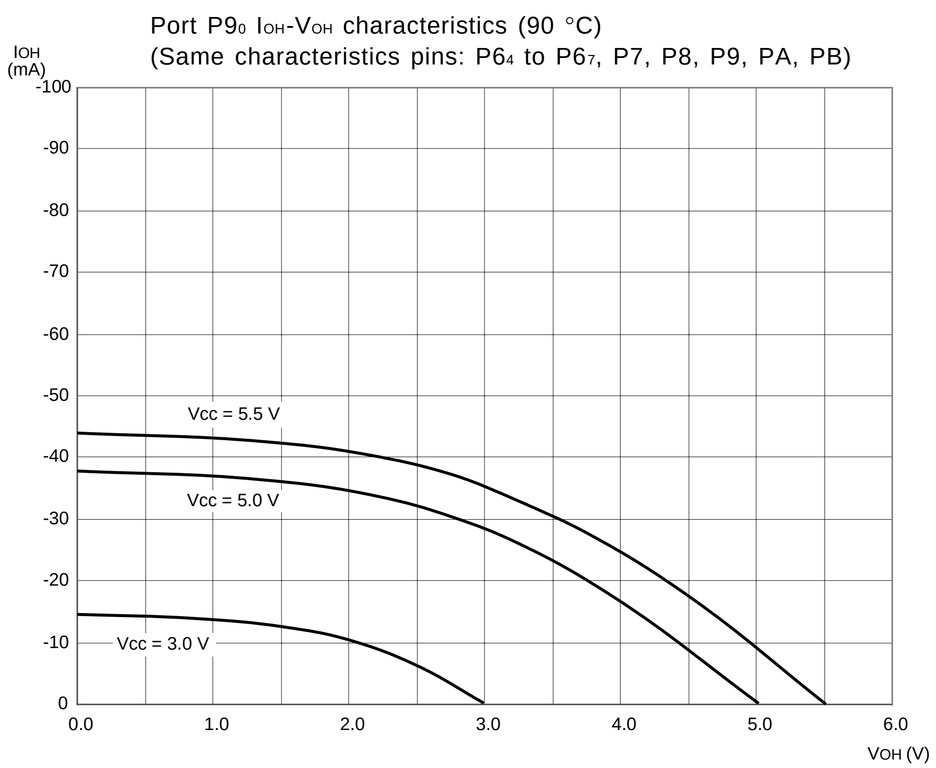


Fig. 3.2.6 CMOS output port P-channel side characteristics (90 °C)

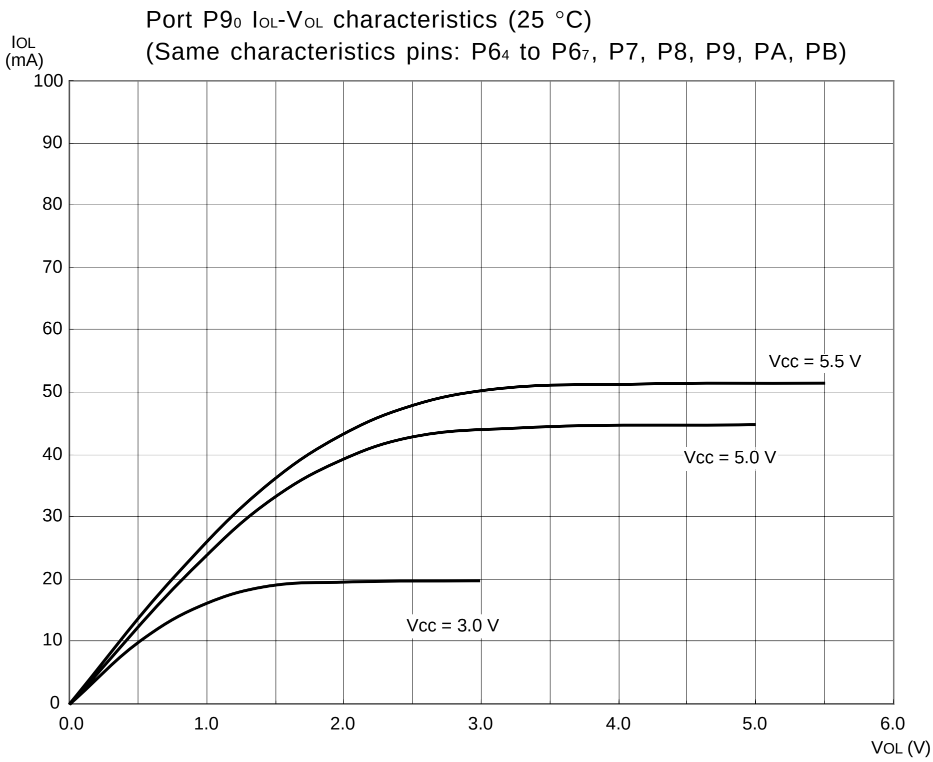


Fig. 3.2.7 CMOS output port N-channel side characteristics (25 °C)

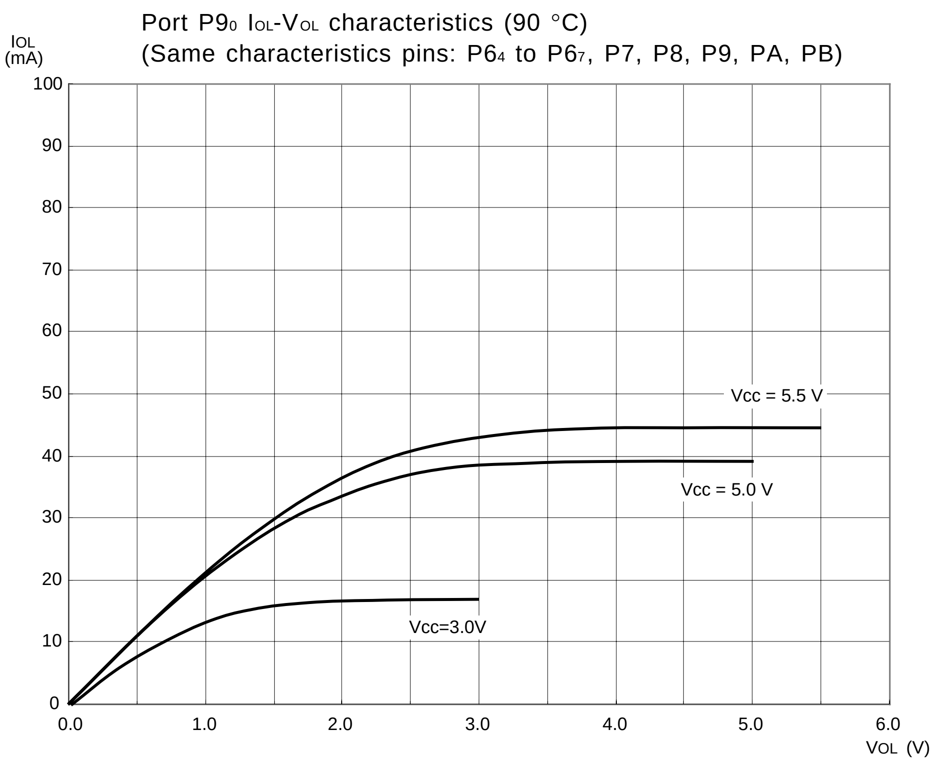


Fig. 3.2.8 CMOS output port N-channel side characteristics (90 °C)

3.2.3 A-D conversion standard characteristics

Figure 3.2.9 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)”.

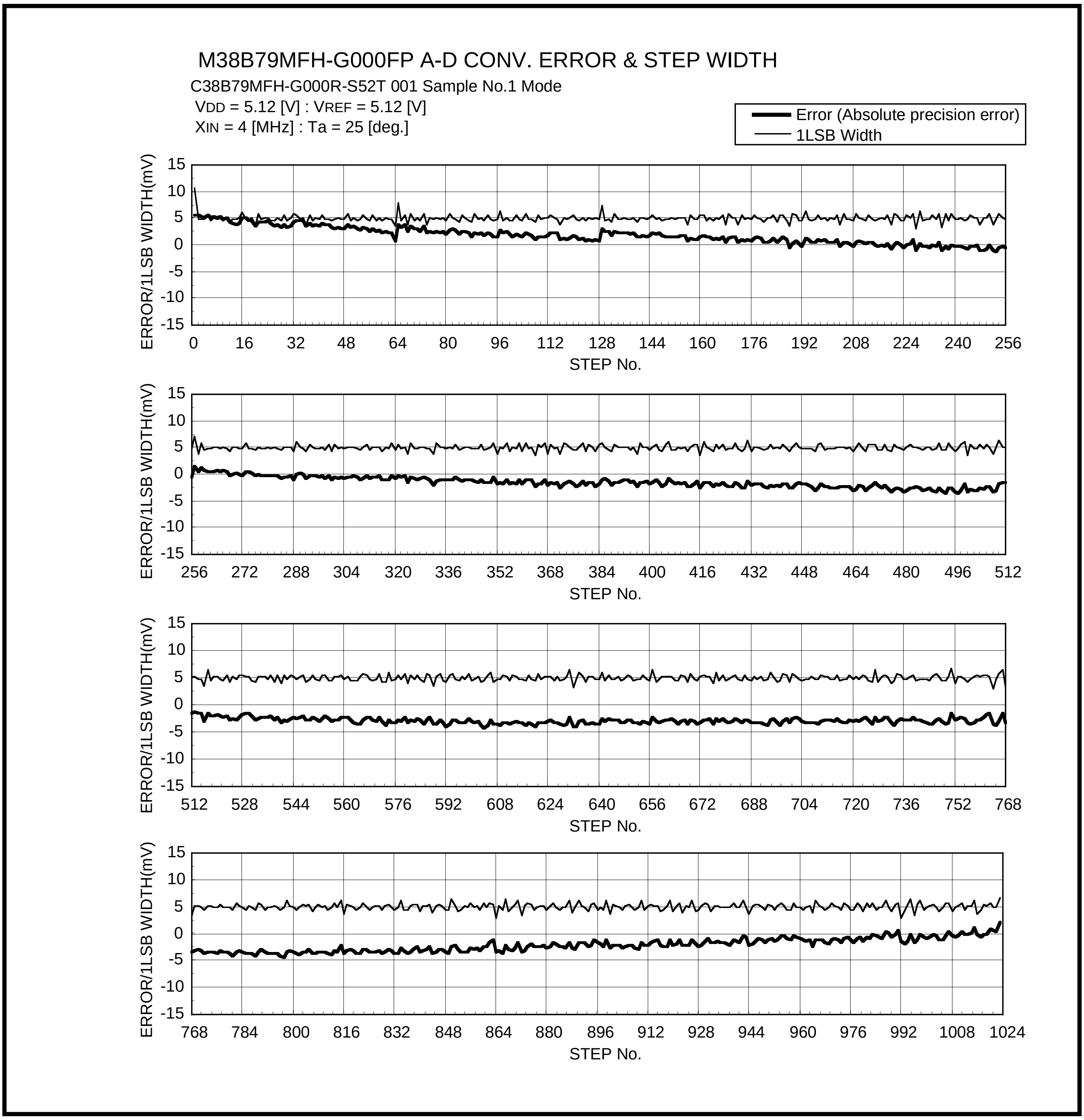


Fig. 3.2.9 A-D conversion standard characteristics

3.3 Notes on use

3.3.1 Notes on interrupts

(1) Change of relevant register settings

When switching an active edge of an external interrupt or switching an interrupt sources of an interrupt vector address where two or more interrupt sources are allocated, the interrupt request bit may be set to “1”. When not requiring the interrupt occurrence synchronized with these setting, take the following sequence.

Set the corresponding interrupt enable bit to “0” (disabled) .

↓

Set the interrupt edge select bit, the active edge switch bit, or the interrupt source select bit to “1”.

↓

NOP (One or more instructions)

↓

Set the corresponding interrupt request bit to “0” (no interrupt request issued).

↓

Set the corresponding interrupt enable bit to “1” (enabled).

Fig. 3.3.1 Setting procedure of relevant registers

■ **Reason**

When setting the followings, the interrupt request bit may be set to “1”.
•When switching external interrupt active edge
Related register: Interrupt edge selection register (address $3A_{16}$)
•When switching interrupt sources of an interrupt vector address where two or more interrupt sources are allocated
Related register: Interrupt source switch register (address 39_{16})

(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.

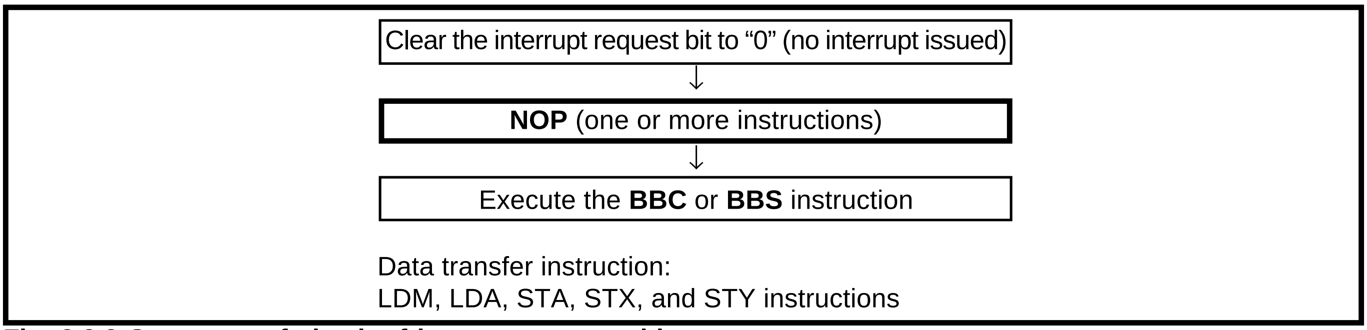


Fig. 3.3.2 Sequence of check of interrupt request bit

(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.

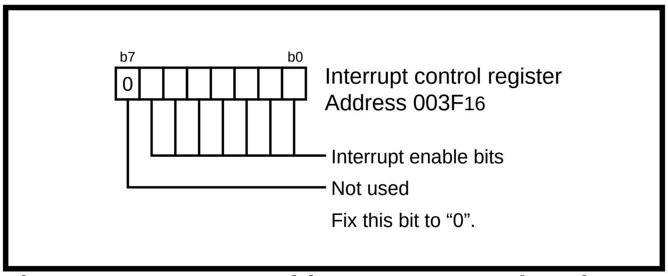


Fig. 3.3.3 Structure of interrupt control register 2

3.3.2 Notes on I/O port

(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”.

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 an optional built-in pull-up resistor, note on varied current values:

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

● **Reason**

The 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) 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

(3) Pull-up/Pull-down control
When each port which has built-in pull-up/pull-down resistor is set to output port, pull-up/pull-down control of corresponding port becomes invalid. (Pull-up/pull-down cannot be set.)
● **Reason**
Pull-up/pull-down control is valid only when each direction register is set to the input mode.

3.3.3 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 before enabling interrupts.

(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.

● When selecting an external clock, the set of automatic transfer interval becomes invalid.

● Set the transfer interval of each 1-byte data transfer as the following:

(1) Not using FLD controller

Keep the interval for 5 cycles or more of internal system clock from clock rising of the last bit of 1-byte data.

(2) Using FLD controller

(a) Not using gradation display

Keep the interval for 17 cycles or more of internal system clock from clock rising of the last bit of 1-byte data.

(b) Using gradation display

Keep the interval for 27 cycles or more of internal system clock from clock rising of the last bit of 1-byte data.

■ **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 system clock 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.

Table 3.3.1 SIO1CON3 (address $001C_{16}$) setting example selecting internal synchronous clock

Serial I/O1 control register 3, SIO1CON3 (address $001C_{16}$)		Not using FLDC	Not using gradation display mode	Using gradation display mode
Internal synchronous clock selection bits (b7 to b5)	Automatic transfer interval set bits (b4 to b0)			
0 0 0 : f(XIN) / 4	0 0 0 0 0 : 2 cycles of transfer clocks	Usable	Prohibited	Prohibited
	0 0 0 0 1 : 3 cycles of transfer clocks	Usable	Prohibited	Prohibited
	0 0 0 1 0 : 4 cycles of transfer clocks	Usable	Prohibited	Prohibited
	0 0 0 1 1 : 5 cycles of transfer clocks	Usable	Usable	Usable
0 0 1 : f(XIN) / 8	0 0 0 0 0 : 2 cycles of transfer clocks	Usable	Prohibited	Prohibited
	0 0 0 0 1 : 3 cycles of transfer clocks	Usable	Usable	Usable
0 1 0 : f(XIN) / 16	0 0 0 0 0 : 2 cycles of transfer clocks	Usable	Usable	Usable

Table 3.3.2 SIO1CON3 (address $001C_{16}$) setting example selecting external synchronous clock

Serial I/O1 control register 3, SIO1CON3 (address $001C_{16}$), Automatic transfer interval set bits	“n” cycles of transfer clocks
Not using FLDC	Transfer clock × n cycles ≥ 5 cycles of internal system clock
Not using gradation display mode	Transfer clock × n cycles ≥ 17 cycles of internal system clock
Using gradation display mode	Transfer clock × n cycles ≥ 27 cycles of internal system clock

3.3.4 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.”

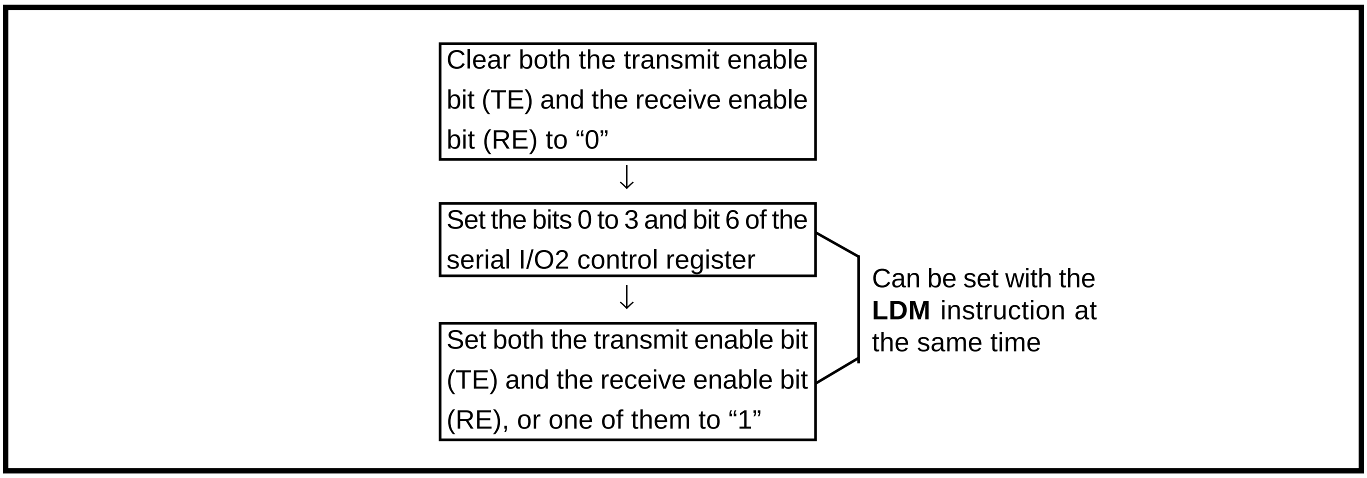


Fig. 3.3.4 Sequence of setting serial I/O2 control register again

(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) Setting procedure when serial I/O2 transmit interrupt is used
When setting the transmit enable bit to "1", the serial I/O2 transmit interrupt request bit is automatically set to "1". When not requiring the interrupt occurrence synchronized with the transmission enabled, take the following sequence.
①Set the serial I/O1 transmit interrupt enable bit to "0" (disabled).
②Set the transmit enable bit to "1".
③Set the serial I/O1 transmit interrupt request bit to "0" after 1 or more instructions have been executed.
④Set the serial I/O1 tranmit interrupt enable bit to "1" (enabled).

(8) Using TxD pin
The $P6_5$/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 $P6_5$/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.

3.3.5 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").

3.3.6 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.

(2) A-D converter power source pin

The AVSS pin is A-D converter power source pin. Regardless of using the A-D conversion function or not, connect it as following :

• AVSS : Connect to the VSS line

● **Reason**

If the AVSS 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

• Do not execute the **STP** instruction and **WIT** instruction

3.3.7 Notes on D-A converter

(1) PB_0/DA state at reset

The PB_0/DA pin becomes a high-impedance state at reset.

(2) Connection with low-impedance load

If connecting a D-A output with a load having a low impedance, use an external buffer. It is because the D-A converter circuit does not include a buffer.

(3) Usable voltage

Vcc must be 3.0 V or more when using the D-A converter.

3.3.8 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.

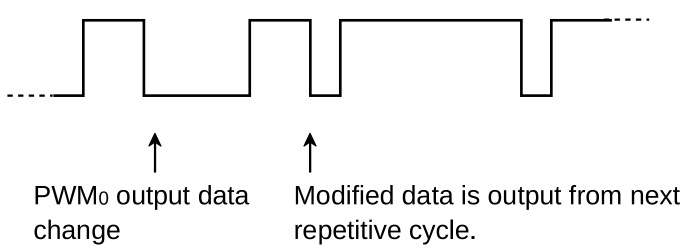


Fig. 3.3.5 PWM_0 output

3.3.9 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 $0EEE_{16}$) 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 $0EEE_{16}$), it cannot be programmed to “0” again. This bit becomes “0” after reset.

3.3.10 Notes on reset

(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.

3.3.11 Each port state during “L” state of $\overline{\text{RESET}}$ pin

Table 3.3.3 shows a pin state during “L” state of $\overline{\text{RESET}}$ pin.

Table 3.3.3 Pin state during “L” state of $\overline{\text{RESET}}$ pin

Pin name	Pin state
P0, P2	Output port (with pull-down resistor)
P1, P3	Input port (with pull-down resistor)
P4, P5, $P6_0$ to $P6_3$	Input port (without pull-down resistor) (**Note**)
$P6_4$ to $P6_7$, P7, $P8_0$ to $P8_3$, P9, PA, PB_0 to PB_6	Input port (floating)

Note: Whether built-in pull-down resistors are connected or not can be specified in ordering mask ROM.

3.3.12 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”.

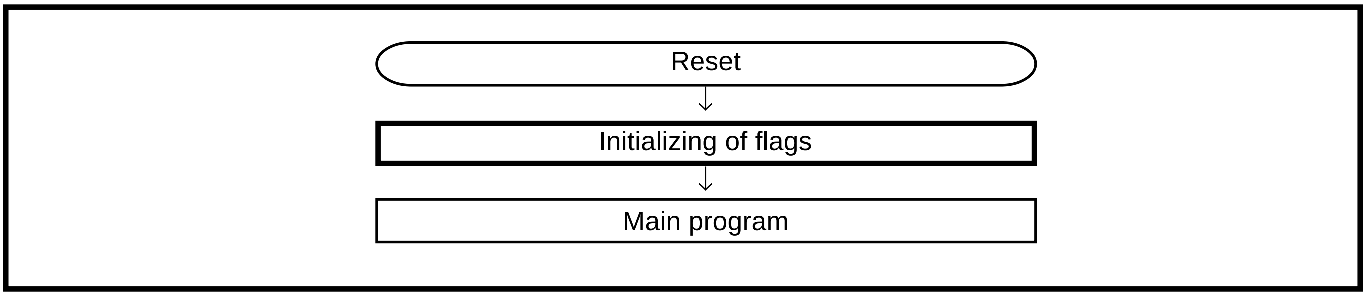


Fig. 3.3.6 Initialization of processor status register

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.

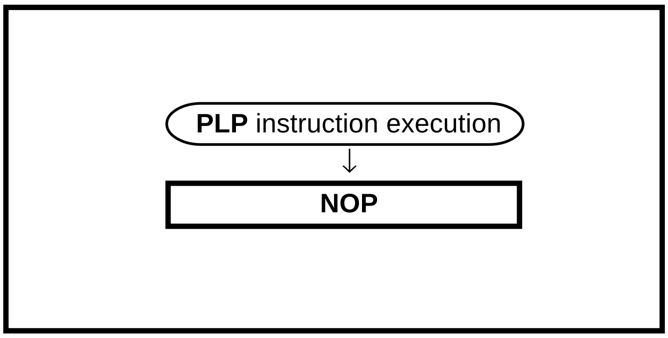


Fig. 3.3.7 Sequence of PLP instruction execution

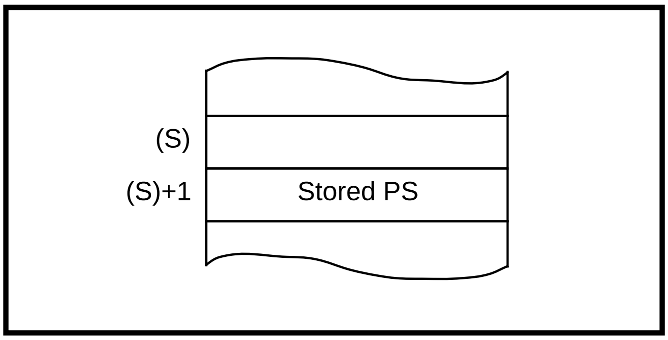


Fig. 3.3.8 Stack memory contents after PHP instruction execution

(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.

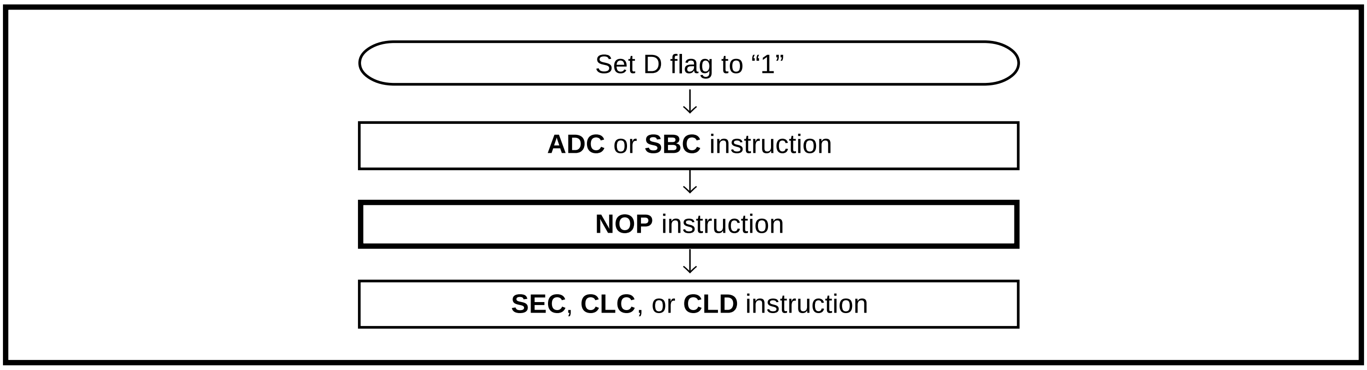


Fig. 3.3.9 Status flag at decimal calculations

(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.

3.3.13 Notes on CPU reprogramming mode

(1) Transfer the CPU reprogramming mode control program to the internal RAM before selecting the CPU reprogramming mode, and then, execute it on the internal RAM. Additionally, when the subroutine or stack operation instruction is used in the control program, make sure in order not to destroy the control program transferred to the internal RAM through the stack area.

(2) Be careful of the instruction description (specifying address, and so on) because the CPU reprogramming mode control program is transferred to the internal RAM and executed on the internal RAM.

(3) Write to the watchdog timer control register periodically in order not to generate the watchdog timer interrupt by the CPU reprogramming mode control program (refer to "**2.9 Watchdog timer**").

3.3.14 Notes on flash memory version

The CNV_{SS} pin is connected to the internal memory circuit block by a low-ohmic resistance, since it has the multiplexed function to be a programmable power source pin (V_{PP} pin) as well.
To improve the noise margin, connect the CNV_{SS} pin to V_{SS} through 1 to 10 kΩ resistance.
Even when the wiring of the CNV_{SS} pin of the mask ROM version is connected to Vss through this resistor, that will not affect operation.

3.3.15 Termination of unused pins

(1) Terminate unused pins

① Output ports : Open

② Input ports :
Connect each pin to VCC or VSS 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 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 VCC or 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) 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 Vss pin with the shortest possible wiring (within 20 mm).

● **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.

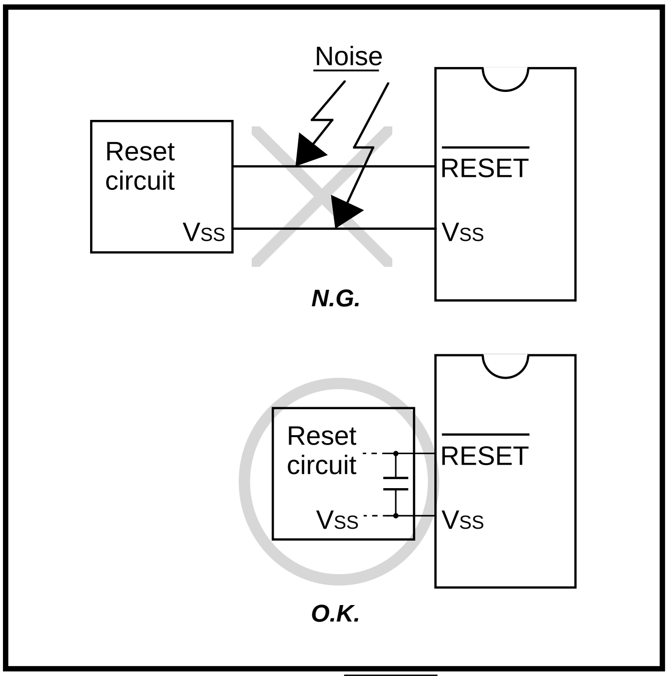


Fig. 3.4.1 Wiring for the $\overline{\text{RESET}}$ pin

(2) 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.

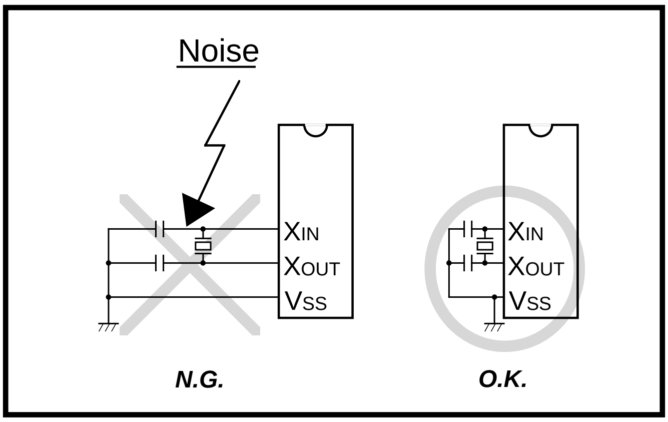


Fig. 3.4.2 Wiring for clock I/O pins

(3) Wiring to CNVss pin

Connect the CNVss pin to the Vss pin with the shortest possible wiring.

● **Reason**

The processor mode of a microcomputer is influenced by a potential at the CNVss pin. If a potential difference is caused by the noise between pins CNVss and Vss, the processor mode may become unstable. This may cause a microcomputer malfunction or a program runaway.

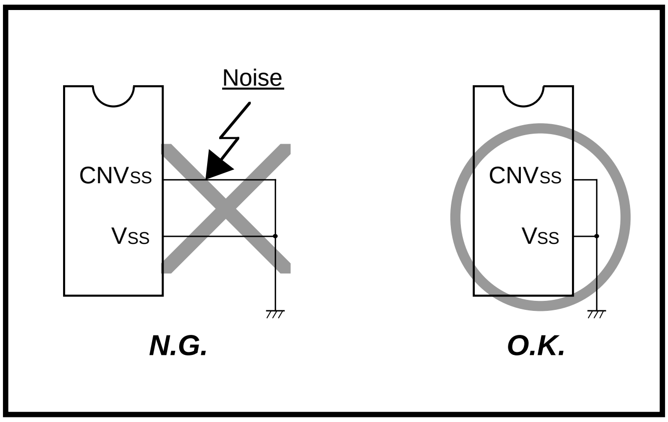


Fig. 3.4.3 Wiring for CNVss pin

(4) Wiring to VPP pin of flash memory version

Connect an approximately 5 kΩ resistor to the VPP pin the shortest possible in series and also to the Vss pin. When not connecting the resistor, make the length of wiring between the VPP pin and the VSS 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 VPP pin of the flash memory version is the power source input pin for the built-in flash memory. When programming/erasing in the built-in flash memory, the impedance of the VPP pin is low to allow the electric current for writing/erasing flow into the flash memory. Because of this, noise can enter easily. If noise enters the VPP pin, abnormal instruction codes or data are read from the built-in flash memory, which may cause a program runaway.

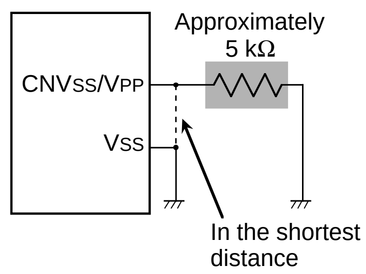


Fig. 3.4.4 Wiring for the VPP pin of the flash memory version

3.4.2 Connection of bypass capacitor across Vss line and Vcc line

Connect an approximately 0.1 μF bypass capacitor across the VSS line and the VCC line as follows:

- Connect a bypass capacitor across the VSS pin and the VCC pin at equal length.
- Connect a bypass capacitor across the VSS pin and the VCC pin with the shortest possible wiring.
- Use lines with a larger diameter than other signal lines for VSS line and VCC line.
- Connect the power source wiring via a bypass capacitor to the VSS pin and the VCC pin.

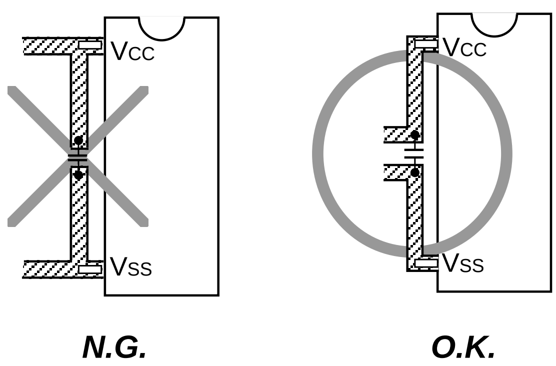


Fig. 3.4.5 Bypass capacitor across the VSS line and the VCC line

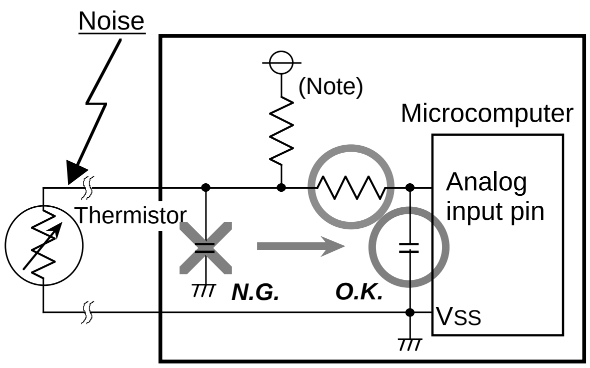
3.4.3 Wiring to analog input pins

- Connect an approximately 100 Ω to 1 kΩ resistor to an analog signal line which is connected to an analog input pin in series. Besides, connect the resistor to the microcomputer as close as possible.
- Connect an approximately 1000 pF capacitor across the VSS pin and the analog input pin. Besides, connect the capacitor to the VSS pin as close as possible. Also, connect the capacitor across the analog input pin and the VSS pin at equal length.

● Reason

Signals which is input in an analog input pin (such as an A-D converter/comparator input pin) are usually output signals from sensor. The sensor which detects a change of event is installed far from the printed circuit board with a microcomputer, the wiring to an analog input pin is longer necessarily. This long wiring functions as an antenna which feeds noise into the microcomputer, which causes noise to an analog input pin.

If a capacitor between an analog input pin and the V_{SS} pin is grounded at a position far away from the V_{SS} 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.

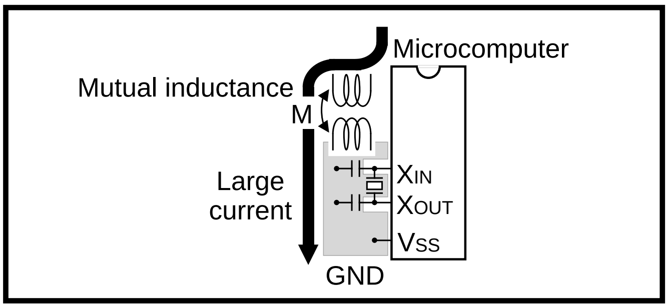


Fig. 3.4.7 Wiring for a large current signal line

(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.

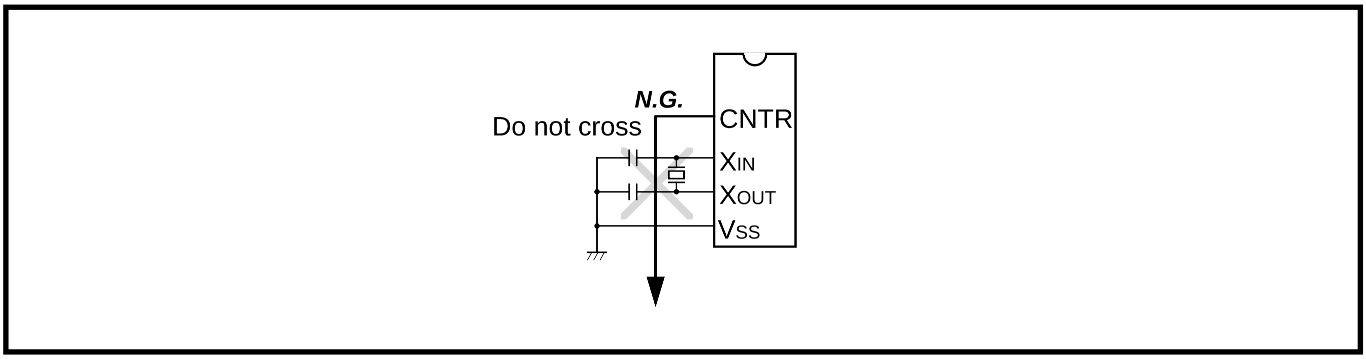


Fig. 3.4.8 Wiring of signal lines where potential levels change frequently

(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.

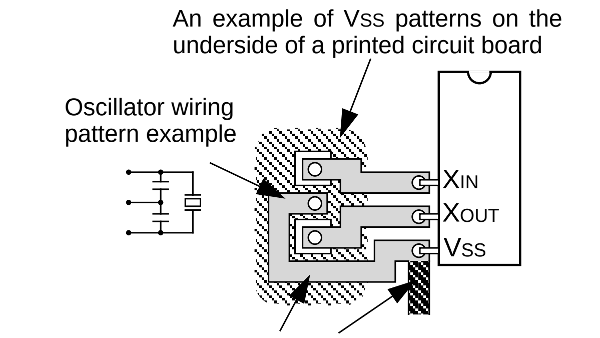


Fig. 3.4.9 Vss pattern on the underside of an oscillator

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.

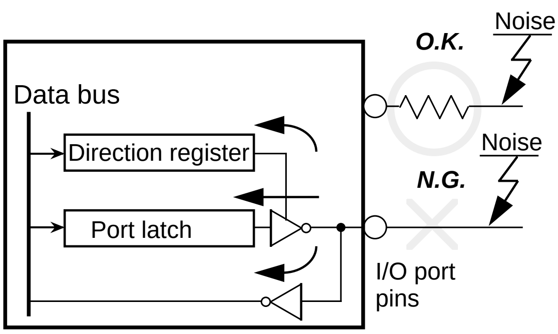


Fig. 3.4.10 Setup for I/O ports

3.4.6 Providing of watchdog timer function by software

If a microcomputer runs away because of noise or others, it can be detected by a software watchdog timer and the microcomputer can be reset to normal operation. This is equal to or more effective than program runaway detection by a hardware watchdog timer. The following shows an example of a watchdog timer provided by software.
In the following example, to reset a microcomputer to normal operation, the main routine detects errors of the interrupt processing routine and the interrupt processing routine detects errors of the main routine.
This example assumes that interrupt processing is repeated multiple times in a single main routine processing.

<The main routine>

- Assigns a single byte of RAM to a software watchdog timer (SWDT) and writes the initial value N in the SWDT once at each execution of the main routine. The initial value N should satisfy the following condition:
  N+1 ≥ (Counts of interrupt processing executed in each main routine)
  As the main routine execution cycle may change because of an interrupt processing or others, the initial value N should have a margin.
- Watches the operation of the interrupt processing routine by comparing the SWDT contents with counts of interrupt processing after the initial value N has been set.
- Detects that the interrupt processing routine has failed and determines to branch to the program initialization routine for recovery processing in the following case:
  If the SWDT contents do not change after interrupt processing.

<The interrupt processing routine>

- Decrements the SWDT contents by 1 at each interrupt processing.
- Determines that the main routine operates normally when the SWDT contents are reset to the initial value N at almost fixed cycles (at the fixed interrupt processing count).
- Detects that the main routine has failed and determines to branch to the program initialization routine for recovery processing in the following case:
  If the SWDT contents are not initialized to the initial value N but continued to decrement and if they reach 0 or less.

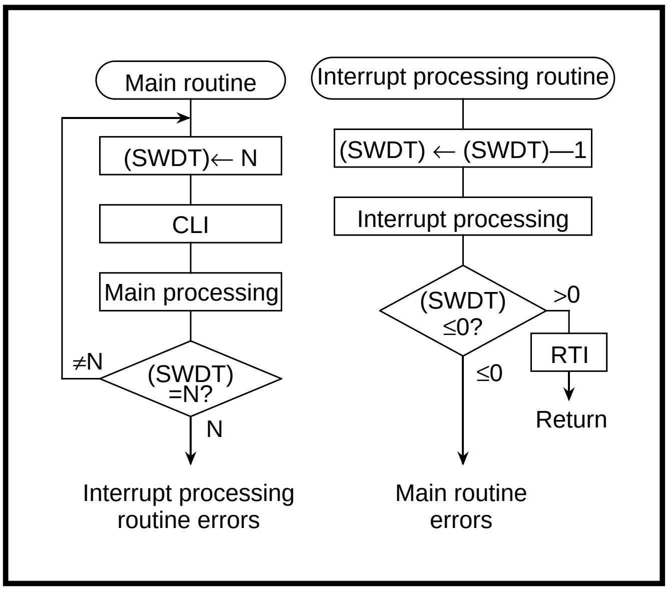


Fig. 3.4.11 Watchdog timer by software

3.5 Control registers

Port Pi

b7 b6 b5 b4 b3 b2 b1 b0

Port Pi (i = 0 to 7, 9, A)
(Pi: addresses 00_{16}, 02_{16}, 04_{16}, 06_{16}, 08_{16}, $0A_{16}$, $0C_{16}$, $0E_{16}$, 12_{16}, 14_{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 (i =0–7, 9, A)

Port P8

b7 b6 b5 b4 b3 b2 b1 b0

Port P8
(P8: address 10_{16})

b	Name	Functions	At reset	R	W
0	Port $P8_0$	●In output mode Write ········ Port latch Read ········ Port latch ●In input mode Write ········ Port latch Read ········ Value of pin	0	○	○
1	Port $P8_1$		0	○	○
2	Port $P8_2$		0	○	○
3	Port $P8_3$		0	○	○
4	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
5			0	×	×
6			0	×	×
7			0	×	×

Fig. 3.5.2 Structure of Port P8

Port PB

b7 b6 b5 b4 b3 b2 b1 b0

Port PB
(PB: address 16_{16})

b	Name	Functions	At reset	R	W
0	Port PB_0	●In output mode Write ········ Port latch Read ········ Port latch ●In input mode Write ········ Port latch Read ········ Value of pin	0	O	O
1	Port PB_1		0	O	O
2	Port PB_2		0	O	O
3	Port PB_3		0	O	O
4	Port PB_4		0	O	O
5	Port PB_5		0	O	O
6	Port PB_6		0	O	O
7	Nothing is arranged for this bit. When this bit is read out, the contents are undefined.		0	×	×

Fig. 3.5.3 Structure of Port PB

Port Pi direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port Pi direction register (PiD) (i = 1, 3 to 7, 9, A)
[Addresses 03_{16}, 07_{16}, 09_{16}, $0B_{16}$, $0D_{16}$, $0F_{16}$, 13_{16}, 15_{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	×	O
1		0 : Port Pi_1 input mode 1 : Port Pi_1 output mode	0	×	O
2		0 : Port Pi_2 input mode 1 : Port Pi_2 output mode	0	×	O
3		0 : Port Pi_3 input mode 1 : Port Pi_3 output mode	0	×	O
4		0 : Port Pi_4 input mode 1 : Port Pi_4 output mode	0	×	O
5		0 : Port Pi_5 input mode 1 : Port Pi_5 output mode	0	×	O
6		0 : Port Pi_6 input mode 1 : Port Pi_6 output mode	0	×	O
7		0 : Port Pi_7 input mode 1 : Port Pi_7 output mode	0	×	O

Fig. 3.5.4 Structure of Port Pi direction register (i = 1, 3–7, 9, A)

Port P8 direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port P8 direction register
(P8D: address 11_{16})

b	Name	Functions	At reset	R	W
0	Port P8 direction register	0 : Port P8$_0$ input mode 1 : Port P8$_0$ output mode	0	×	○
1		0 : Port P8$_1$ input mode 1 : Port P8$_1$ output mode	0	×	○
2		0 : Port P8$_2$ input mode 1 : Port P8$_2$ output mode	0	×	○
3		0 : Port P8$_3$ input mode 1 : Port P8$_3$ output mode	0	×	○
4	Nothing is arranged for these bits. When these bits are read out, the contents are undefined.		0	×	×
5			0	×	×
6			0	×	×
7			0	×	×

Fig. 3.5.5 Structure of Port P8 direction register

Port PB direction register

b7 b6 b5 b4 b3 b2 b1 b0

Port PB direction register
(PBD: address 17_{16})

b	Name	Functions	At reset	R	W
0	Port PB direction register	0 : Port PB$_0$ input mode 1 : Port PB$_0$ output mode	0	×	○
1		0 : Port PB$_1$ input mode 1 : Port PB$_1$ output mode	0	×	○
2		0 : Port PB$_2$ input mode 1 : Port PB$_2$ output mode	0	×	○
3		0 : Port PB$_3$ input mode 1 : Port PB$_3$ output mode	0	×	○
4		0 : Port PB$_4$ input mode 1 : Port PB$_4$ output mode	0	×	○
5		0 : Port PB$_5$ input mode 1 : Port PB$_5$ output mode	0	×	○
6		0 : Port PB$_6$ input mode 1 : Port PB$_6$ output mode	0	×	○
7	Nothing is arranged for this bit. When this bit is read out, the contents are undefined.		0	×	×

Fig. 3.5.6 Structure of Port PB direction 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 (00_{16} to FF_{16}) 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	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. 3.5.7 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 PB_0–PB_6 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	O	O
1			0	O	O
2	Serial I/O1 synchronous clock selection bits (PB_3/S_{STB1} pin control bits)	b3b2 0 0: Internal synchronous clock (PB_3 pin is I/O port.) 0 1: External synchronous clock (PB_3 pin is I/O port.) 1 0: Internal synchronous clock (PB_3 pin is S_{STB1} output.) 1 1: Internal synchronous clock (PB_3 pin is S_{STB1} output.)	0	O	O
3			0	O	O
4	Serial I/O initialization bit	0: Serial I/O initialization 1: Serial I/O enabled	0	O	O
5	Transfer mode selection bit	0: Full-duplex (transmit/receive) mode (PB_6 pin is S_{IN1} input.) 1: Transmit-only mode (PB_6 pin is I/O port.)	0	O	O
6	Transfer direction selection bit	0: LSB first 1: MSB first	0	O	O
7	Serial I/O1 clock pin selection bit	0: S_{CLK11} (PB_0/S_{CLK12} pin is I/O port.) 1: S_{CLK12} (PB_4/S_{CLK11} pin is I/O port.)	0	O	O

Fig. 3.5.8 Structure of Serial I/O1 control register 1

Serial I/O1 control register 2

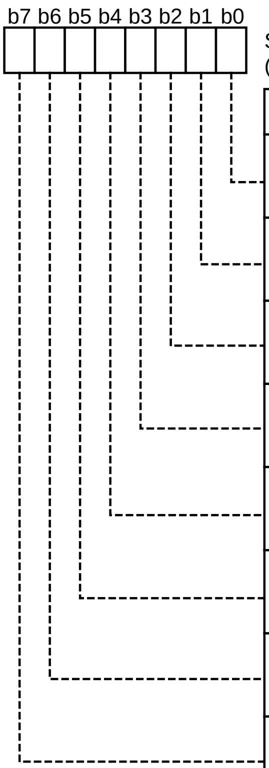
b7 b6 b5 b4 b3 b2 b1 b0

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

b	Name	Functions	At reset	R	W
0	PB1/SRDY1 • PB2/SBUSY1 pin control bits	b3b2b1b0 0 0 0 0: PB1, PB2 pins are I/O ports. 0 0 0 1: Not used 0 0 1 0: PB1 pin is SRDY1 output; PB2 pin is I/O port. 0 0 1 1: PB1 pin is SRDY1 output; PB2 pin is I/O port. 0 1 0 0: PB1 pin is I/O port; PB2 pin is SBUSY1 input. 0 1 0 1: PB1 pin is I/O port; PB2 pin is SBUSY1 input. 0 1 1 0: PB1 pin is I/O port; PB2 pin is SBUSY1 output. 0 1 1 1: PB1 pin is I/O port; PB2 pin is SBUSY1 output. 1 0 0 0: PB1 pin is SRDY1 input; PB2 pin is SBUSY1 output. 1 0 0 1: PB1 pin is SRDY1 input; PB2 pin is SBUSY1 output. 1 0 1 0: PB1 pin is SRDY1 input; PB2 pin is SBUSY1 output. 1 0 1 1: PB1 pin is SRDY1 input; PB2 pin is SBUSY1 output. 1 1 0 0: PB1 pin is SRDY1 output; PB2 pin is SBUSY1 input. 1 1 0 1: PB1 pin is SRDY1 output; PB2 pin is SBUSY1 input. 1 1 1 0: PB1 pin is SRDY1 output; PB2 pin is SBUSY1 input. 1 1 1 1: PB1 pin is SRDY1 output; PB2 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	PB5/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. 3.5.9 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 SCLK1 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. 3.5.10 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 $1C_{16}$)

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	O	O
1			0	O	O
2			0	O	O
3			0	O	O
4			0	O	O
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	O	O
6			0	O	O
7			0	O	O

Fig. 3.5.11 Structure of Serial I/O1 control register 3

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: $f(X_{IN})$ 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	O	O
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	O	O
2	$\overline{SRDY2}$ output enable bit (SRDY)	0: $P6_7$ pin operates as normal I/O pin 1: $P6_7$ pin operates as $\overline{SRDY2}$ output pin	0	O	O
3	Transmit interrupt source selection bit (TIC)	0: When transmit buffer has emptied 1: When transmit shift operation is completed	0	O	O
4	Transmit enable bit (TE)	0: Transmit disabled 1: Transmit enabled	0	O	O
5	Receive enable bit (RE)	0: Receive disabled 1: Receive enabled	0	O	O
6	Serial I/O2 mode selection bit (SIOM)	0: Clock asynchronous serial I/O (UART) mode 1: Clock synchronous serial I/O mode	0	O	O
7	Serial I/O2 enable bit (SIOE)	0: Serial I/O2 disabled (pins $P6_4$–$P6_7$ operate as normal I/O pins) 1: Serial I/O2 enabled (pins $P6_4$–$P6_7$ operate as serial I/O pins)	0	O	O

Fig. 3.5.12 Structure of Serial I/O2 control register

Serial I/O2 status register

b7 b6 b5 b4 b3 b2 b1 b0

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. 3.5.13 Structure of Serial I/O2 status register

Serial I/O2 transmit/receive buffer register

b7 b6 b5 b4 b3 b2 b1 b0

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. 3.5.14 Structure of Serial I/O2 transmit/receive buffer register

Timer i

b7 b6 b5 b4 b3 b2 b1 b0

Timer i (i = 1, 3 to 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.	1	○	○
1	• The value set in this register is written to both	1	○	○
2	the timer i and the timer i latch at one time.	1	○	○
3	• When the timer i is read out, the count value	1	○	○
4	of the timer i is read out.	1	○	○
5		1	○	○
6		1	○	○
7		1	○	○

Fig. 3.5.15 Structure of Timer i

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

Fig. 3.5.16 Structure of Timer 2

PWM control register

b7 b6 b5 b4 b3 b2 b1 b0

PWM control register
(PWMCON: address 26_{16})

b	Name	Functions	At reset	R	W
0	$P9_6/PWM_0$ output selection bit	0: I/O port 1: PWM_0 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. 3.5.17 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	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. 3.5.18 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	O	O
1	Timer 2 count stop bit	0: Count operation 1: Count stop	0	O	O
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	O	O
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	O	O
5			0	O	O
6	Timer 1 output selection bit ($P7_5$)	0: I/O port 1: Timer 1 output	0	O	O
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are "0".		0	O	×

Fig. 3.5.19 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	O	O
1	Timer 4 count stop bit	0: Count operation 1: Count stop	0	O	O
2	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	O	O
3					
4	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 1 1: Not available	0	O	O
5			0	O	O
6	Timer 3 output selection bit (P76)	0: I/O port 1: Timer 3 output	0	O	O
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	O	×

Fig. 3.5.20 Structure of Timer 34 mode register

Timer 56 mode register

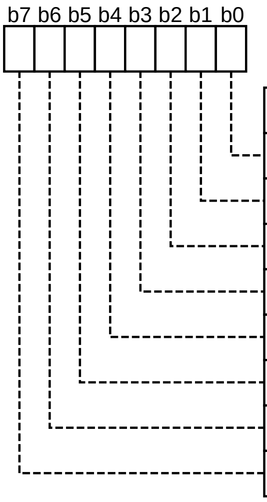
b7 b6 b5 b4 b3 b2 b1 b0

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	O	O
1	Timer 6 count stop bit	0: Count operation 1: Count stop	0	O	O
2	Timer 5 count source selection bit	0: f(XIN)/8 or f(XCIN)/16 1: Timer 4 underflow	0	O	O
3	Timer 6 operation mode selection bit	0: Timer mode 1: PWM mode	0	O	O
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	O	O
5			0	O	O
6	Timer 6 (PWM) output selection bit (P74)	0: I/O port 1: Timer 6 output	0	O	O
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	O	×

Fig. 3.5.21 Structure of Timer 56 mode register

D-A conversion register

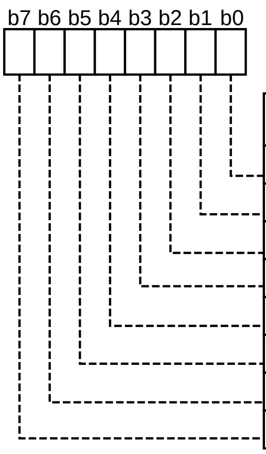


D-A conversion register
(DA: address $2B_{16}$)

b	Functions	At reset	R	W
0	•This is a register for set of D-A conversion output value. • D-A conversion is performed automatically by setting a value in this register.	0	O	O
1		0	O	O
2		0	O	O
3		0	O	O
4		0	O	O
5		0	O	O
6		0	O	O
7		0	O	O

Fig. 3.5.22 Structure of D-A conversion register

Timer X (low-order, high-order)



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	O	O
1		1	O	O
2		1	O	O
3		1	O	O
4		1	O	O
5		1	O	O
6		1	O	O
7		1	O	O

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

Timer X mode register 1

b7 b6 b5 b4 b3 b2 b1 b0

Timer X mode register 1
(TXM1: address $2E_{16}$)

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	O	O
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	O	O
2			0	O	O
3	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	O	×
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	O	O
5			0	O	O
6	CNTR2 active edge switch bit	0 : •Start from “H” output in pulse output mode •Count at rising edge in event counter mode •Measure “H” pulse width in pulse width measurement mode 1 : •Start from “L” output in pulse output mode •Count at falling edge in event counter mode •Measure “L” pulse width in pulse width measurement mode	0	O	O
7	Timer X stop control bit	0 : Count operating 1 : Count stop	0	O	O

Fig. 3.5.24 Structure of Timer X mode register 1

Timer X mode register 2

b7 b6 b5 b4 b3 b2 b1 b0

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

b	Name	Functions	At reset	R	W
0	Real time port control bit ($P9_4$)	0: Real time port function is invalid 1: Real time port function is valid	0	O	O
1	Real time port control bit ($P9_5$)	0: Real time port function is invalid 1: Real time port function is valid	0	O	O
2	$P9_4$ data for real time port	0: "L" output 1: "H" output	0	O	O
3	$P9_5$ data for real time port	0: "L" output 1: "H" output	0	O	O
4	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are "0".		0	O	×
5			0	O	×
6			0	O	×
7			0	O	×

Fig. 3.5.25 Structure of Timer X mode register 2

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	0	O	×
1		0	O	×
2		0	O	×
3		0	O	×
4		0	O	×
5		0	O	×
6		0	O	×
7		0	O	×

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. 3.5.26 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	O	O
1	Counter sampling clock selection bit	0: $f(X_{IN})/128$ or $f(X_{CIN})$ 1: $f(X_{IN})/256$ or $f(X_{CIN})/2$	0	O	O
2	Noise filter sampling clock selection bits (INT_2)	b3 b2 0 0: Filter is not used. 0 1: $f(X_{IN})/32$ or $f(X_{CIN})$ 1 0: $f(X_{IN})/64$ or $f(X_{CIN})/2$ 1 1: $f(X_{IN})/128$ or $f(X_{CIN})/4$	0	O	O
3			0	O	O
4	One-sided/both-sided edge detection selection bit	0: One-sided edge detection 1: Both-sided edge detection **(Note)**	0	O	O
5	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	O	×
6			0	O	×
7			0	O	×

Note: When the noise filter sampling clock selection bits (bits 2, 3) is “00”, the both-sided edge detection function cannot be used.

Fig. 3.5.27 Structure of Interrupt interval determination control register

AD/DA control register

b7 b6 b5 b4 b3 b2 b1 b0

AD/DA 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: PA_0/AN_0 0 0 0 1: PA_1/AN_1 0 0 1 0: PA_2/AN_2 0 0 1 1: PA_3/AN_3 0 1 0 0: PA_4/AN_4 0 1 0 1: PA_5/AN_5 0 1 1 0: PA_6/AN_6 0 1 1 1: PA_7/AN_7 1 0 0 0: $P9_0/S_{IN3}/AN_8$ 1 0 0 1: $P9_1/S_{OUT3}/AN_9$ 1 0 1 0: $P9_2/\overline{S_{CLK3}}/AN_{10}$ 1 0 1 1: $P9_3/\overline{S_{RDY3}}/AN_{11}$ 1 1 0 0: $P9_4/RTP_1/AN_{12}$ 1 1 0 1: $P9_5/RTP_0/AN_{13}$ 1 1 1 0: $P9_6/PWM_0/AN_{14}$ 1 1 1 1: $P9_7/B_{UZ02}/AN_{15}$	0	O	O
1			0	O	O
2			0	O	O
3			0	O	O
4	AD conversion completion bit	0: Conversion in progress 1: Conversion completed	1	O	O
5	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are "0".		0	O	×
6	DA output enable bit	0: DA output disabled 1: DA output enabled	0	O	O
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are "0".		0	O	×

Fig. 3.5.28 Structure of AD/DA control register

A-D conversion register (low-order)

b7 b6 b5 b4 b3 b2 b1 b0

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	O	×
1		Undefined	O	×
2		Undefined	O	×
3		Undefined	O	×
4		Undefined	O	×
5		Undefined	O	×
6	These are A-D conversion result (low-order 2 bits) stored bits. This is read exclusive register.	Undefined	O	×
7		Undefined	O	×

Note: Do not read this register during A-D conversion.

Fig. 3.5.29 Structure of A-D conversion register (low-order)

A-D conversion register (high-order)

b7 b6 b5 b4 b3 b2 b1 b0

A-D conversion register (high-order)
(ADH: address 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	O	×
1		Undefined	O	×
2		Undefined	O	×
3		Undefined	O	×
4		Undefined	O	×
5		Undefined	O	×
6		Undefined	O	×
7		Undefined	O	×

Note: Do not read this register during A-D conversion.

Fig. 3.5.30 Structure of A-D conversion register (high-order)

PWM register (high-order)

b7 b6 b5 b4 b3 b2 b1 b0

PWM register (high-order)
(PWMH: address 35_{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	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. 3.5.31 Structure of PWM register (high-order)

PWM register (low-order)

b7 b6 b5 b4 b3 b2 b1 b0

PWM register (low-order)
(PWML: address 36_{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	O	O
1		Undefined	O	O
2		Undefined	O	O
3		Undefined	O	O
4		Undefined	O	O
5		Undefined	O	O
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	O	X
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	O	X

Fig. 3.5.32 Structure of PWM register (low-order)

Baud rate generator

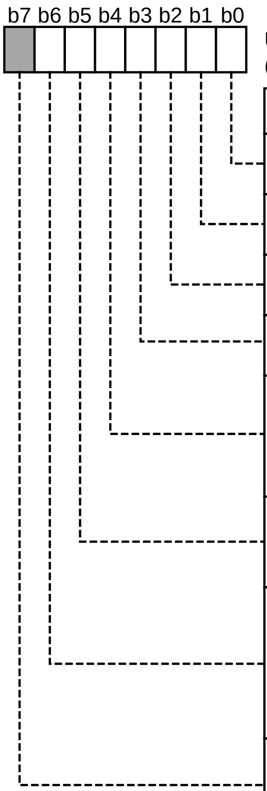
b7 b6 b5 b4 b3 b2 b1 b0

Baud rate generator
(BRG: address 37_{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	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. 3.5.33 Structure of Baud rate generator

UART control register



UART control register
(UARTCON: address 38_{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	$P6_5$/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} ($P6_7$/S_{CLK22} pin is used as I/O port or $\overline{S_{RDY2}}$ output pin.) 1: S_{CLK22} ($P6_6$/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. 3.5.34 Structure of UART control register

Interrupt source switch register

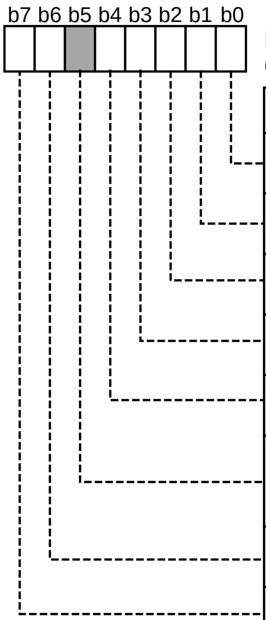
b7 b6 b5 b4 b3 b2 b1 b0

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

b	Name	Functions	At reset	R	W
0	INT3/serial I/O2 transmit interrupt switch bit	0: INT3 intrrupt 1: Serial I/O2 transmit interrupt	0	○	○
1	INT4/A-D conversion interrupt switch bit	0: INT4 interrupt 1: A-D conversion intrerrupt	0	○	○
2	INT1/serial I/O3 interrupt switch bit	0: INT1 intrrupt 1: Serial I/O3 interrupt	0	○	○
3	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	○	×
4			0	○	×
5			0	○	×
6			0	○	×
7			0	○	×

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	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	0 : Rising edge count 1 : Falling edge count	0	○	○

Fig. 3.5.36 Structure of Interrupt edge selection register

CPU mode register

b7	b6	b5	b4	b3	b2	b1	b0
				1		0	0

CPU mode register
(CPUM: 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	O	O
1			0	O	O
2	Stack page selection bit	0 : Page 0 1 : Page 1	0	O	O
3	Fix this bit to “1”.		1	O	O
4	Port Xc switch bit	0: I/O port function 1: X_{CIN}-X_{COUT} oscillation function	0	O	O
5	Main clock (X_{IN}-X_{OUT}) stop bit	0: Oscillating 1: Stopped	0	O	O
6	Main clock division ratio selection bit	0: f(X_{IN}) (high-speed mode) 1: f(X_{IN})/4 (middle-speed mode)	1	O	O
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	O	O

Note: B6, b7 function in the CPU reprogramming mode is described below.

b	Name	Functions	At reset	R	W
6	Main clock division ratio selection bits	b7 b6 0 0: ϕ = f(X_{IN}) (high-speed mode) 0 1: ϕ = f(X_{IN})/4 (middle-speed mode) 1 0: ϕ = f(X_{CIN})/2 (low-speed mode) 1 1: Not available	1	O	O
7			0	O	O

Fig. 3.5.37 Structure of CPU mode register

Interrupt request register 1

b7 b6 b5 b4 b3 b2 b1 b0

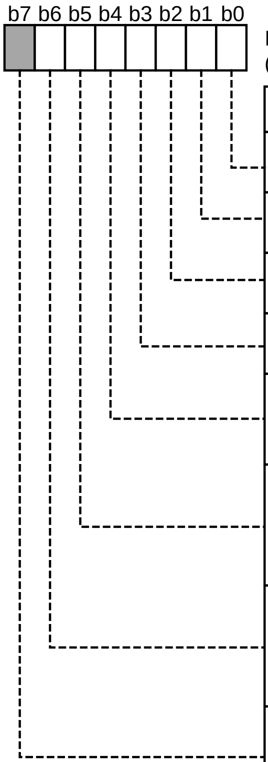
Interrupt request register 1
(IREQ1 : address $3C_{16}$)

b	Name	Functions	At reset	R	W
0	INT_0 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻
1	INT_1/serial I/O3 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻
2	INT_2 interrupt request bit Remote controller /counter overflow interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻
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	O	✻
4	Timer X interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻
5	Timer 1 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻
6	Timer 2 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻
7	Timer 3 interrupt request bit	0 : No interrupt request issued 1 : Interrupt request issued	0	O	✻

✻: “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



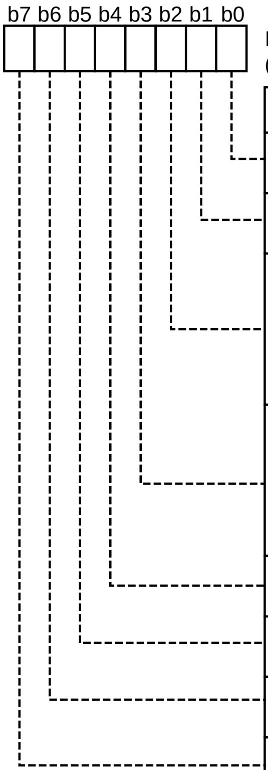
Interrupt request register 2
(IREQ2 : address $3D_{16}$)

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

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

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/serial I/O3 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

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	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
1	Timer 5 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
2	Timer 6 interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
3	Serial I/O2 receive interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
4	INT_3/Serial I/O2 transmit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
5	INT_4 interrupt enable bit A-D converter interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
6	FLD blanking interrupt enable bit FLD digit interrupt enable bit	0 : interrupt disabled 1 : Interrupt enabled	0	O	O
7	Fix “0” to this bit.		0	O	O

Fig. 3.5.41 Structure of Interrupt control register 2

Serial I/O3 control register

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O3 control register
(SIO3CON: address $0EEC_{16}$)

b	Name	Functions	At reset	R	W
0	Internal synchronous clock selection bits	b2b1b0 000: f(XIN)/4 or f(XCIN)/8 001: f(XIN)/8 or f(XCIN)/16 010: f(XIN)/16 or f(XCIN)/32 011: f(XIN)/32 or f(XCIN)/64 110: f(XIN)/64 or f(XCIN)/128 111: f(XIN)/128 or f(XCIN)/256	0	O	O
1			0	O	O
2			0	O	O
3	Serial I/O3 port selection bit ($P9_1$, $P9_2$)	0: I/O port 1: SOUT3, SCLK3 signal output	0	O	O
4	$\overline{\text{SRDY3}}$ output selection bit ($P9_3$)	0: I/O port 1: $\overline{\text{SRDY3}}$ signal output	0	O	O
5	Transfer direction selection bit	0: LSB first 1: MSB first	0	O	O
6	Synchronous clock selection bit	0: External clock 1: Internal clock	0	O	O
7	$P9_1$/SOUT3 P-channel output disable bit ($P9_1$)	0: CMOS output (in output mode) 1: N-channel open drain output (in output mode)	0	O	O

Fig. 3.5.42 Structure of Serial I/O3 control register

Serial I/O3 register

b7 b6 b5 b4 b3 b2 b1 b0

Serial I/O3 register
(SIO3: address $0EED_{16}$)

b	Functions	At reset	R	W
0	This is the buffer register which is used to write transmit data or to read receive data. When selecting an internal clock, the serial transfer is started by writing this register.	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. 3.5.43 Structure of Serial I/O3 register

Watchdog timer control register

Watchdog timer control register
(WDTCON: address $0EEE_{16}$)

b	Name	Functions	At reset	R	W
0	Watchdog timer H (high-order 6 bits of reading exclusive)		1	O	×
1			1	O	×
2			1	O	×
3			1	O	×
4			1	O	×
5			1	O	×
6	STP instruction disable bit	0: STP instruction enabled 1: STP instruction disabled	0	O	O
7	Watchdog timer H count source selection bit	0: Watchdog timer L underflow 1: $f(X_{IN})/8$ or $f(X_{CIN})/16$	0	O	O

Fig. 3.5.44 Structure of Watchdog timer control register

Pull-up control register 3

Pull-up control register 3
(PULL3: address $0EEF_{16}$)

b	Name	Functions	At reset	R	W
0	Ports PA_0, PA_1 pull-up control	0: No pull-up 1: Pull-up	0	O	O
1	Ports PA_2, PA_3 pull-up control	0: No pull-up 1: Pull-up	0	O	O
2	Ports PA_4, PA_5 pull-up control	0: No pull-up 1: Pull-up	0	O	O
3	Ports PA_6, PA_7 pull-up control	0: No pull-up 1: Pull-up	0	O	O
4	Ports PB_0, PB_1 pull-up control	0: No pull-up 1: Pull-up	0	O	O
5	Ports PB_2, PB_3 pull-up control	0: No pull-up 1: Pull-up	0	O	O
6	Ports PB_4, PB_5 pull-up control	0: No pull-up 1: Pull-up	0	O	O
7	Ports PB_6 pull-up control	0: No pull-up 1: Pull-up	0	O	O

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

Fig. 3.5.45 Structure of Pull-up control register 3

Pull-up control register 1

b7 b6 b5 b4 b3 b2 b1 b0

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

b	Name	Functions	At reset	R	W
0	Ports $P6_4$, $P6_5$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
1	Ports $P6_6$, $P6_7$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
2	Ports $P7_0$, $P7_1$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
3	Ports $P7_2$, $P7_3$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
4	Ports $P7_4$, $P7_5$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
5	Ports $P7_6$, $P7_7$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
6	Nothing is arranged for these bits. These are write disabled bits. When these bits are read out, the contents are “0”.		0	○	×
7			0	○	×

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

Fig. 3.5.46 Structure of Pull-up control register 1

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 $P8_0$, $P8_1$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
1	Ports $P8_2$, $P8_3$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
2	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	○	×
3	Ports $P9_0$, $P9_1$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
4	Ports $P9_2$, $P9_3$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
5	Ports $P9_4$, $P9_5$ pull-up control	0: No pull-up 1: Pull-up	0	○	○
6	Ports $P9_6$, $P9_7$ 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. 3.5.47 Structure of Pull-up control register 2

Port P0 digit output set switch register

b7 b6 b5 b4 b3 b2 b1 b0

Port P0 digit output set switch register
(P0DOR: address $0EF2_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P0_0$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
1	Port $P0_1$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
2	Port $P0_2$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
3	Port $P0_3$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
4	Port $P0_4$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
5	Port $P0_5$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
6	Port $P0_6$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
7	Port $P0_7$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O

Fig. 3.5.48 Structure of Port P0 digit output set switch register

Port P2 digit output set switch register

b7 b6 b5 b4 b3 b2 b1 b0

Port P2 digit output set switch register
(P2DOR: address $0EF3_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P2_0$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
1	Port $P2_1$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
2	Port $P2_2$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
3	Port $P2_3$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
4	Port $P2_4$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
5	Port $P2_5$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
6	Port $P2_6$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O
7	Port $P2_7$ FLD/Digit switch bit	0: FLD output 1: Digit output	0	O	O

Fig. 3.5.49 Structure of Port P2 digit output set switch 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	0 : General-purpose mode 1 : Automatic display mode	0	O	O
1	Display start bit	0 : Display stopped 1 : Display in progress (display starts by writing “1”)	0	O	O
2	Tscan control bits	b3 b2 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	O	O
3			0	O	O
4	Timing number control bit	0 : 16 timing mode 1 : 32 timing mode **(Note 2)**	0	O	O
5	Gradation display mode selection control bit	0 : Not selected 1 : Selected **(Notes 1, 2)**	0	O	O
6	Tdisp counter count source selection bit	0 : $f(X_{IN})/16$ 1 : $f(X_{IN})/64$	0	O	O
7	High-breakdown voltage port drivability selection bit	0 : Drivability strong 1 : Drivability weak	0	O	O

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. 3.5.50 Structure of FLDC mode register

Tdisp time set register

b7 b6 b5 b4 b3 b2 b1 b0

Tdisp time set register
(TDISP: address 0EF5₁₆)

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(XIN) = 4 MHz •bit 6 of FLDC mode register = 0 (f(XIN)/16 is selected as Tdisp counter count source.) •Tdisp time set register = 200 (C8₁₆).	0	○	○
1		0	○	○
2		0	○	○
3		0	○	○
4		0	○	○
5		0	○	○
6		0	○	○
7		0	○	○

Fig. 3.5.51 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(XIN) = 4 MHz •bit 6 of FLDC mode register = 0 (f(XIN)/16 is selected as Tdisp counter count source.) •Toff1 time set register = 30 ($1E_{16}$).	1	O	O
1		1	O	O
2		1	O	O
3		1	O	O
4		1	O	O
5		1	O	O
6		1	O	O
7		1	O	O

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

Fig. 3.5.52 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(XIN) = 4 MHz •bit 6 of FLDC mode register = 0 (f(XIN)/16 is selected as Tdisp counter count source.) •Toff2 time set register = 180 ($B4_{16}$).	1	O	O
1		1	O	O
2		1	O	O
3		1	O	O
4		1	O	O
5		1	O	O
6		1	O	O
7		1	O	O

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. 3.5.53 Structure of Toff2 time set register

FLD data pointer/FLD data pointer reload register

b7 b6 b5 b4 b3 b2 b1 b0

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 P6, P5, P4, P3, P1, P0, and P2, 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	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. 3.5.54 Structure of FLD data pointer/FLD data pointer reload register

Port P4FLD/port switch register

b7 b6 b5 b4 b3 b2 b1 b0

Port P4FLD/port switch register
(P4FPR: address $0EF9_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P4_0$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
1	Port $P4_1$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
2	Port $P4_2$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
3	Port $P4_3$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
4	Port $P4_4$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
5	Port $P4_5$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
6	Port $P4_6$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
7	Port $P4_7$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O

Fig. 3.5.55 Structure of Port P4FLD/port switch register

Port P5FLD/port switch register

b7 b6 b5 b4 b3 b2 b1 b0

Port P5FLD/port switch register
(P5FPR: address $0EFA_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P5_0$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
1	Port $P5_1$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
2	Port $P5_2$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
3	Port $P5_3$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
4	Port $P5_4$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
5	Port $P5_5$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
6	Port $P5_6$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
7	Port $P5_7$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O

Fig. 3.5.56 Structure of Port P5FLD/port switch register

Port P6FLD/port switch register

b7 b6 b5 b4 b3 b2 b1 b0

Port P6FLD/port switch register
(P6FPR: address $0EFB_{16}$)

b	Name	Functions	At reset	R	W
0	Port $P6_0$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
1	Port $P6_1$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
2	Port $P6_2$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
3	Port $P6_3$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
4	Port $P6_4$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
5	Port $P6_5$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
6	Port $P6_6$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O
7	Port $P6_7$ FLD/port switch bit	0 : Normal port 1 : FLD port	0	O	O

Fig. 3.5.57 Structure of Port P6FLD/port switch register

FLD output control register

b7 b6 b5 b4 b3 b2 b1 b0

FLD output control register
(FLDCON : address $0EFC_{16}$)

b	Name	Functions	At reset	R	W
0	$P6_4$–$P6_7$ FLD output reverse bit	0 : Output normally 1 : Reverse output	0	O	O
1	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	O	×
2	$P6_4$–$P6_7$ Toff invalid bit	0 : Operating normally 1 : Toff invalid	0	O	O
3	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	O	×
4	$P7_3$ dimmer output control bit	0 : Ordinary port 1 : Dimmer output	0	O	O
5	Generating/Not of CMOS port Toff section selection bit	0 : Toff section not generated 1 : Toff section generated	0	O	O
6	Generating/Not of high-breakdown voltage port Toff section selection bit	0 : Toff section not generated 1 : Toff section generated	0	O	O
7	Toff2 SET/RESET switch bit	0 : Toff2 RESET; Toff1 SET 1 : Toff2 SET; Tdisp RESET	0	O	O

Fig. 3.5.58 Structure of FLD output control register

Buzzer output control register

b7 b6 b5 b4 b3 b2 b1 b0

Buzzer output control register
(BUZCON: address $0EFD_{16}$)

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: $P7_7$ and $P9_7$ function as ordinary ports. 0 1: $P7_7$/BUZ_{01} functions as a buzzer output. 1 0: $P9_7$/BUZ_{02}/AN_{15} 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. 3.5.59 Structure of Buzzer output control register

Flash memory control register

b7 b6 b5 b4 b3 b2 b1 b0

b7	b6	b5	b4	b3	b2	b1	b0
	0			0			

Flash memory control register
(FCON: address $0EFE_{16}$)

b	Name	Functions	At reset	R	W
0	CPU reprogramming mode select bit **(Note)**	0: CPU reprogramming mod is invalid. (Normal operation mode) 1: When applying 0 V or V_{PPL} to CNV_{SS}/V_{PP} pin, CPU reprogramming mode is invalid. When applying V_{PPH} to CNV_{SS}/V_{PP} pin, CPU reprogramming mode is valid.	0	O	O
1	Erase/Program busy flag	0: Erase and program are completed or not have been executed. 1: Erase/program is being executed.	0	O	O
2	CPU reprogramming mode monitor flag	0: CPU reprogramming mode is invalid. 1: CPU reprogramming mode is valid.	0	O	O
3	Fix this bit to “0”.		0	O	O
4	Erase/Program area select bits	b5 b4 0 0: Addresses 1000_{16} to $FFFF_{16}$ (total 60 Kbytes) 0 1: Addresses 1000_{16} to $7FFF_{16}$ (total 28 Kbytes) 1 0: Addresses 8000_{16} to $FFFF_{16}$ (total 32 Kbytes) 1 1: Not available	0	O	O
5			0	O	O
6	Fix this bit to “0”.		0	O	O
7	Nothing is arranged for this bit. This is a write disabled bit. When this bit is read out, the contents are “0”.		0	O	×

Note: Bit 0 can be reprogrammed only when 0 V is applied to the CNV_{SS}/V_{PP} pin.

Fig. 3.5.60 Structure of Flash memory control register

Flash command register

b7 b6 b5 b4 b3 b2 b1 b0

Flash command register
(FCMD: address $0EFF_{16}$)

b	Functions	At reset	R	W
0	Writing of software command	0	×	○
1	<Software command name> <Command code>	0	×	○
2	•Read command "00_{16}"	0	×	○
3	•Program command "40_{16}"	0	×	○
4	•Program verify command "$C0_{16}$"	0	×	○
5	•Erase command "20_{16}" + "20_{16}"	0	×	○
6	•Erase verify command "$A0_{16}$"	0	×	○
7	•Reset command "FF_{16}" + "FF_{16}"	0	×	○

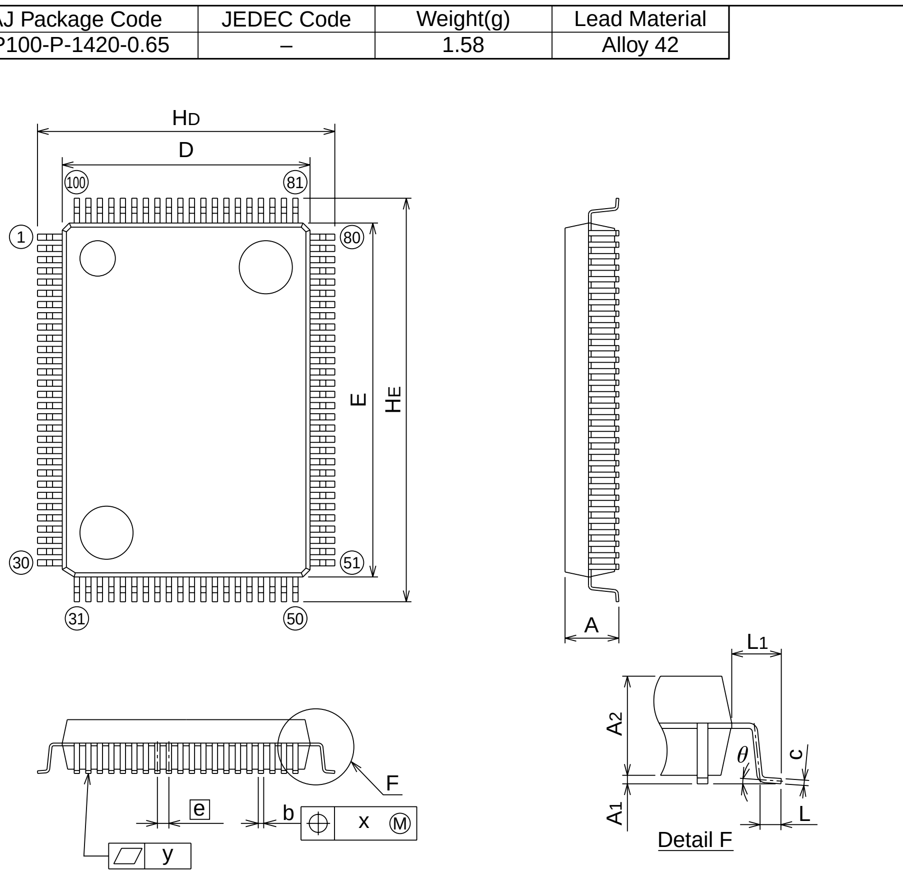
Note: The flash command register is write exclusive register.

Fig. 3.5.61 Structure of Flash command register

3.6 Package outline

100P6S-A (MMP) **Plastic 100pin 14×20mm body QFP**

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
QFP100-P-1420-0.65	–	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	0.25	0.3	0.4
c	0.13	0.15	0.2
D	13.8	14.0	14.2
E	19.8	20.0	20.2
e	–	0.65	–
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	–
x	–	–	0.13
y	–	–	0.1
θ	0°	–	10°
b_2	–	0.35	–
l_2	1.3	–	–
M_D	–	14.6	–
M_E	–	20.6	–

3.7 Machine instructions

Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT, A, R			ZP			BIT, ZP, R		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
ADC (Note 1) (Note 5)	When T = 0 A ← A + M + C When T = 1 M(X) ← M(X) + M + C	When T = 0, this instruction adds the contents M, C, and A; and stores the results in A and C. When T = 1, this instruction adds the contents of M(X), M and C; and stores the results in M(X) and C. When T=1, the contents of A remain unchanged, but the contents of status flags are changed. M(X) represents the contents of memory where is indicated by X.				69	2	2							65	3	2			
AND (Note 1)	When T = 0 A ← A ∧ M When T = 1 M(X) ← M(X) ∧ M	When T = 0, this instruction transfers the contents of A and M to the ALU which performs a bit-wise AND operation and stores the result back in A. When T = 1, this instruction transfers the contents M(X) and M to the ALU which performs a bit-wise AND operation and stores the results back in M(X). When T = 1, the contents of A remain unchanged, but status flags are changed. M(X) represents the contents of memory where is indicated by X.				29	2	2							25	3	2			
ASL	7 0 C ← [ ] ← 0	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																									M_7	M_6	•	•	•	•	Z	•
																											30	2	2				•	•	•	•	•	•	•	•
																											D0	2	2				•	•	•	•	•	•	•	•

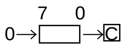
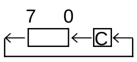
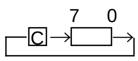
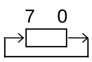
Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT, A			ZP			BIT, ZP		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
BPL (Note 4)	N = 0?	This instruction takes a branch to the appointed address if N is 0. The branch address is specified by a relative address. If N is 1, the 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	•

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	•

3.7 Machine instructions

Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT, A			ZP			BIT, ZP		
			OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
LSR		This instruction shifts either A or M one bit to the right such that bit 7 of the result always is set to 0, and the bit 0 is stored in C.							4A	2	1				46	5	2			
MUL	M(S) • A ← A ✻ M(zz + X) S ← S – 1	This instruction multiply Accumulator with the memory specified by the Zero Page X address mode and stores the high-order byte of the result on the Stack and the low-order byte in A.																		
NOP	PC ← PC + 1	This instruction adds one to the PC but does no otheroperation.	EA	2	1															
ORA (Note 1)	When T = 0 A ← A V M When T = 1 M(X) ← M(X) V M	When T = 0, this instruction transfers the contents of A and M to the ALU which performs a bit-wise "OR", and stores the result in A. When T = 1, this instruction transfers the contents of M(X) and the M to the ALU which performs a bit-wise OR, and stores the result in M(X). The contents of A remain unchanged, but status flags are changed. M(X) represents the contents of memory where is indicated by X.				09	2	2							05	3	2			
PHA	M(S) ← A S ← S – 1	This instruction pushes the contents of A to the memory location designated by S, and decrements the contents of S by one.	48	3	1															
PHP	M(S) ← PS S ← S – 1	This instruction pushes the contents of PS to the memory location designated by S and decrements the contents of S by one.	08	3	1															
PLA	S ← S + 1 A ← M(S)	This instruction increments S by one and stores the contents of the memory designated by S in A.	68	4	1															
PLP	S ← S + 1 PS ← M(S)	This instruction increments S by one and stores the contents of the memory location designated by S in PS.	28	4	1															
ROL		This instruction shifts either A or M one bit left through C. C is stored in bit 0 and bit 7 is stored in C.							2A	2	1				26	5	2			
ROR		This instruction shifts either A or M one bit right through C. C is stored in bit 7 and bit 0 is stored in C.							6A	2	1				66	5	2			
RRF		This instruction rotates 4 bits of the M content to the right.													82	8	2			
RTI	S ← S + 1 PS ← M(S) S ← S + 1 PC_L ← M(S) S ← S + 1 PC_H ← 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 PC_L. S is again incremented by one and stores the contents of memory location designated by S in PC_H.	40	6	1															
RTS	S ← S + 1 PC_L ← M(S) S ← S + 1 PC_H ← M(S) (PC) ← (PC) + 1	This instruction increments S by one and stores the contents of the memory location designated by S in PC_L. S is again incremented by one and the contents of the memory location is stored in PC_H. 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)							
																																	•	•	•	•	•	•	•	•

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	•
																																	•	•	•	•	•	•	•	•

3.7 Machine instructions

Symbol	Contents	Symbol	Contents
IMP	Implied addressing mode	+	Addition
IMM	Immediate addressing mode	–	Subtraction
A	Accumulator or Accumulator addressing mode	✻	Multiplication
BIT, A	Accumulator bit addressing mode	/	Division
BIT, A, R	Accumulator bit relative addressing mode	Λ	Logical OR
ZP	Zero page addressing mode	V	Logical AND
BIT, ZP	Zero page bit addressing mode	∀	Logical exclusive OR
BIT, ZP, R	Zero page bit relative addressing mode	—	Negation
ZP, X	Zero page X addressing mode	←	Shows direction of data flow
ZP, Y	Zero page Y addressing mode	X	Index register X
ABS	Absolute addressing mode	Y	Index register Y
ABS, X	Absolute X addressing mode	S	Stack pointer
ABS, Y	Absolute Y addressing mode	PC	Program counter
IND	Indirect absolute addressing mode	PS	Processor status register
		PC_H	8 high-order bits of program counter
ZP, IND	Zero page indirect absolute addressing mode	PC_L	8 low-order bits of program counter
		AD_H	8 high-order bits of address
IND, X	Indirect X addressing mode	AD_L	8 low-order bits of address
IND, Y	Indirect Y addressing mode	FF	FF in Hexadecimal notation
REL	Relative addressing mode	nn	Immediate value
SP	Special page addressing mode	zz	Zero page address
C	Carry flag	M	Memory specified by address designation of any addressing mode
Z	Zero flag		
I	Interrupt disable flag	M(X)	Memory of address indicated by contents of index register X
D	Decimal mode flag		
B	Break flag	M(S)	Memory of address indicated by contents of stack pointer
T	X-modified arithmetic mode flag		
V	Overflow flag	M(AD_H, AD_L)	Contents of memory at address indicated by AD_H and AD_L, in AD_H is 8 high-order bits and AD_L is 8 low-order bits.
N	Negative flag		
		M(00, AD_L)	Contents of address indicated by zero page AD_L
		Ai	Bit i (i = 0 to 7) of accumulator
		Mi	Bit i (i = 0 to 7) of memory
		OP	Opcode
		n	Number of cycles
		#	Number of bytes

3.8 List of instruction code

D7 – D4 \ D3 – D0		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
	Hexadecimal notation	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

(dark gray) : 3-byte instruction

(light gray) : 2-byte instruction

(white) : 1-byte instruction

3.9 M35501FP

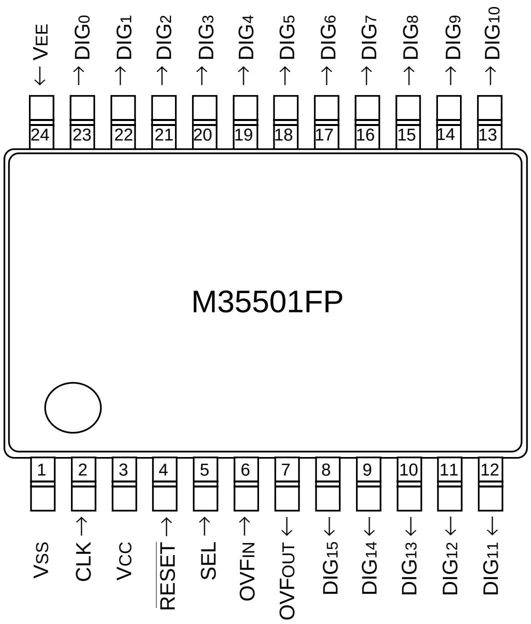
DESCRIPTION

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 38B7 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. 3.9.1 Pin configuration of M35501FP

FUNCTIONAL BLOCK

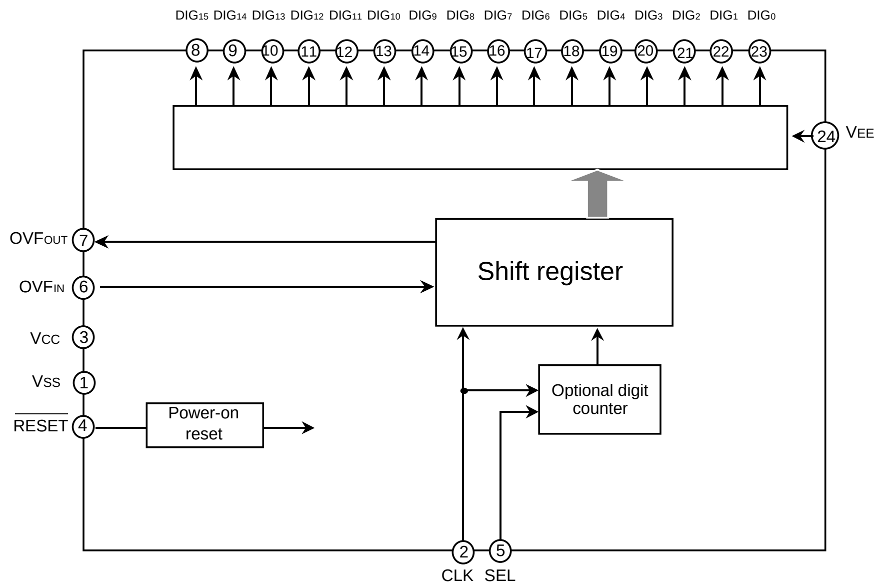


Fig. 3.9.2 Functional block diagram

PIN DESCRIPTION

Table 3.9.1 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.	–	–
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

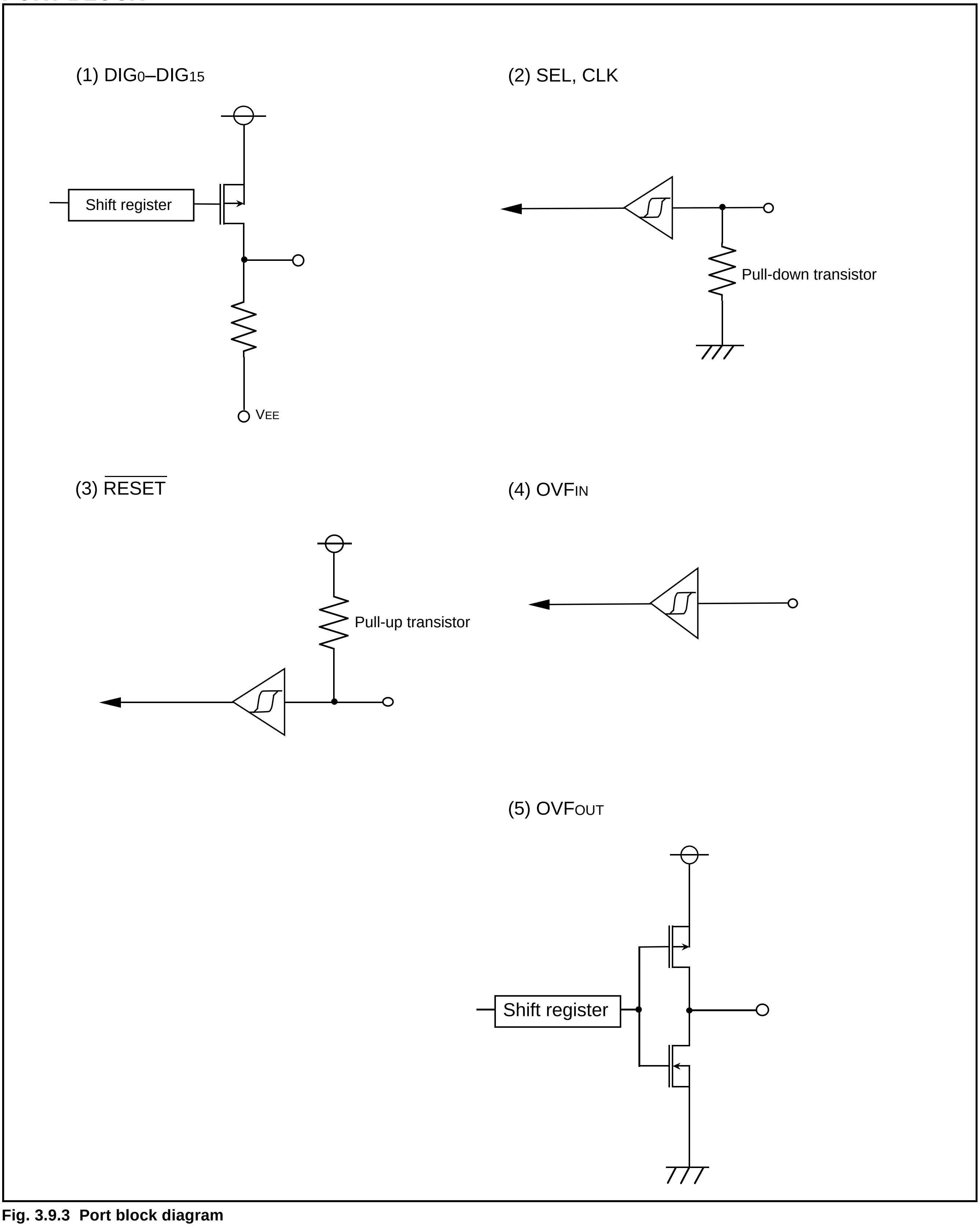


Fig. 3.9.3 Port block diagram

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 3.9.5 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 3.9.6 shows the output waveform.

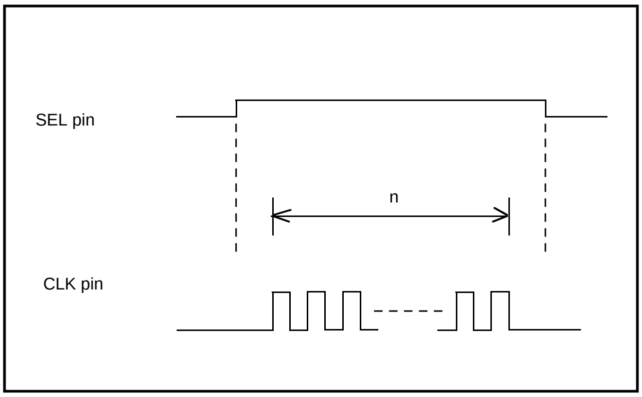


Fig. 3.9.4 Digit setting

(3) Cascade Mode: 17 digits or more selectable

17 digits or more can be used by connecting two M35501FPs or more. Figure 3.9.7 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 3.9.8 shows the output waveform.

DIGIT OUTPUT WAVEFORM

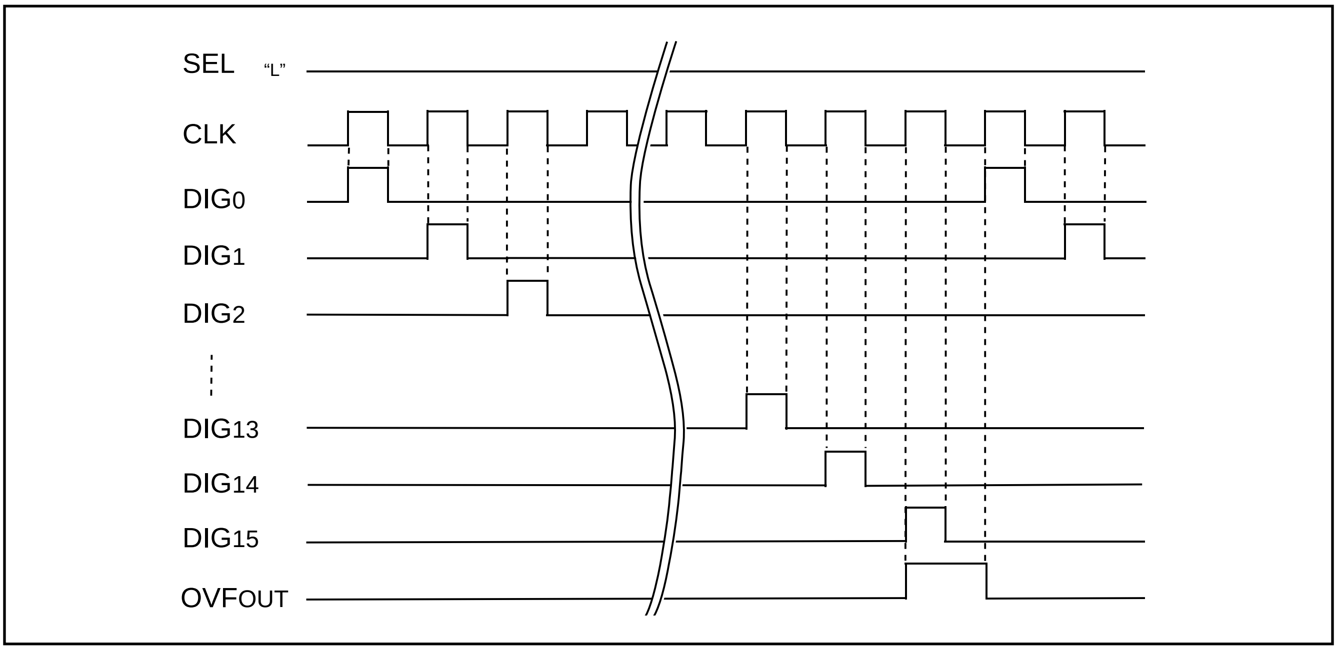


Fig. 3.9.5 16-digit mode output waveform

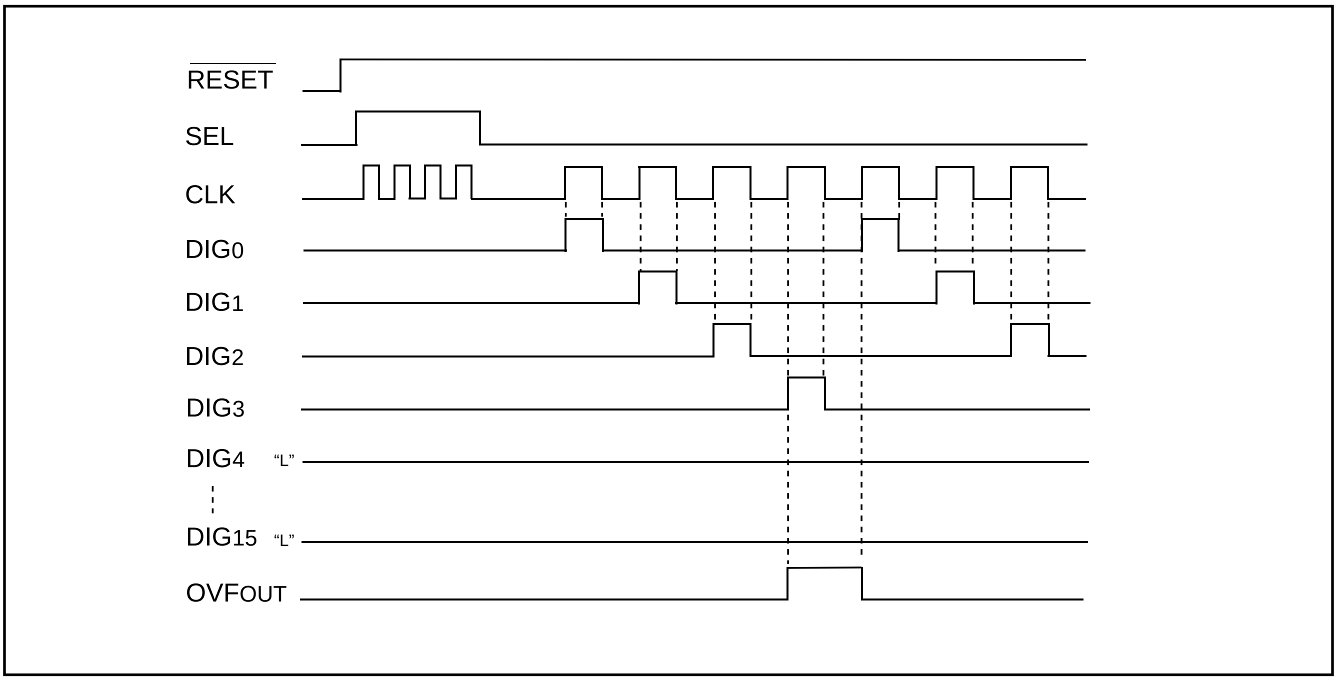


Fig. 3.9.6 Optional digit mode output waveform

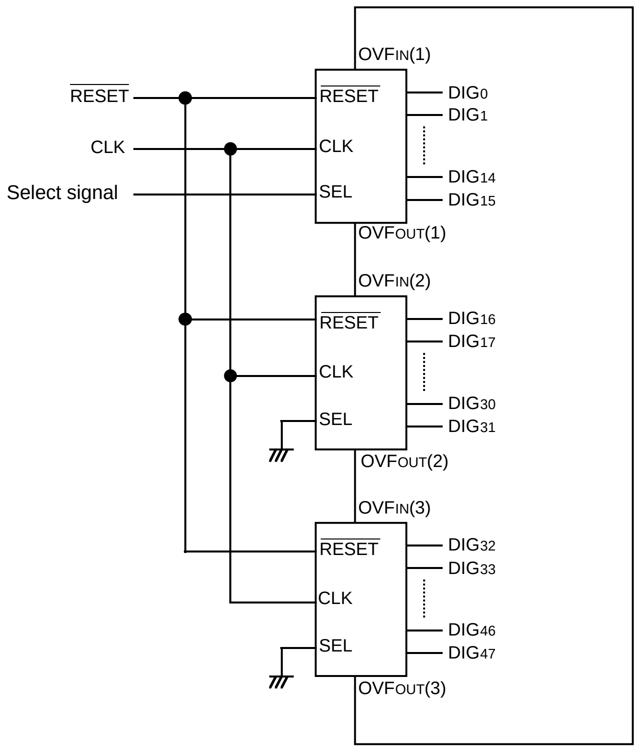


Fig. 3.9.7 Cascade mode connection example: 17 digits or more selected

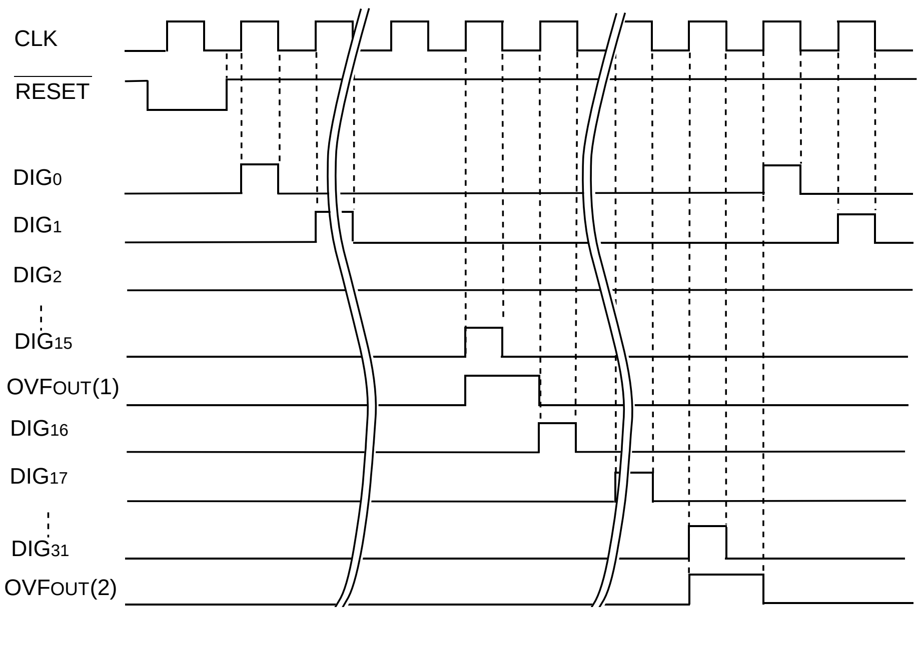


Fig. 3.9.8 Cascade mode output waveform

The number of fluorescent displays can be increased by connecting the M35501FP to the CMOS FLD output pins on a 38B7 Group microcomputer.

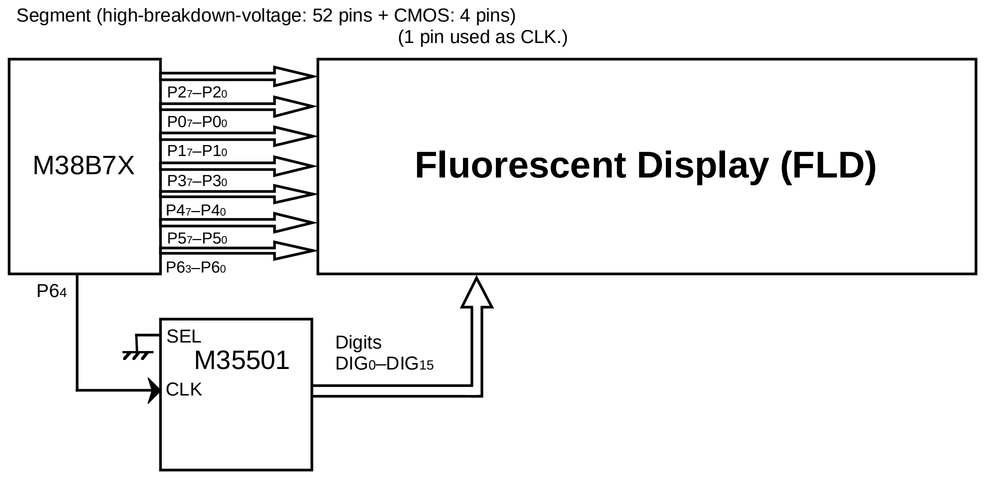


Fig. 3.9.9 Connection example with 38B7 Group microcomputer (1 to 16 digits)

This FLD controller can control up to 32 digits using the 32 timing mode of the 38B7 Group microcomputer.

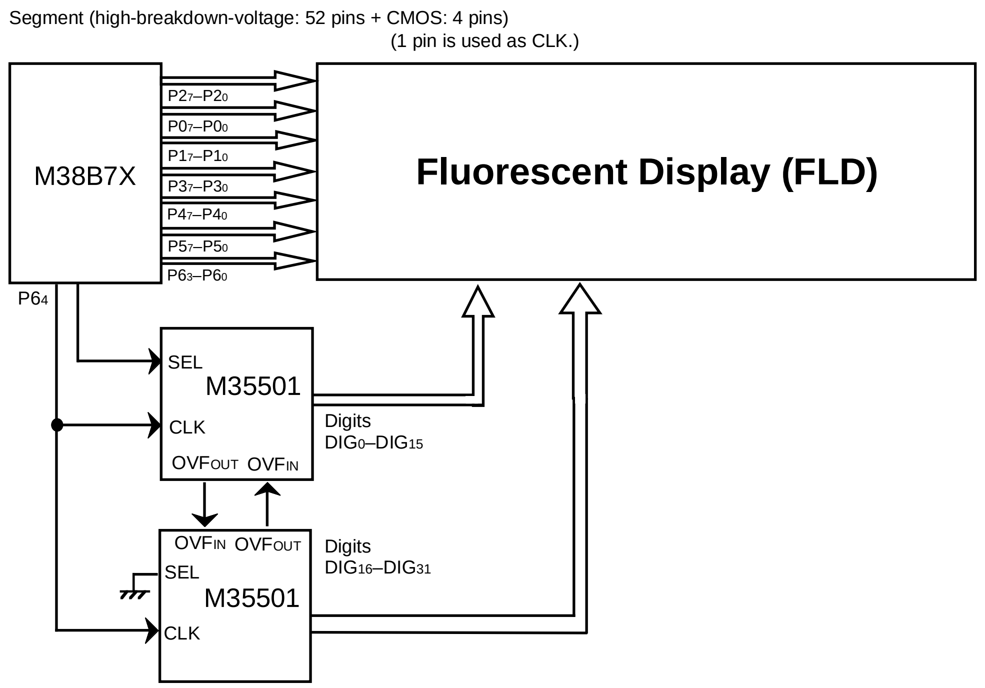


Fig. 3.9.10 Connection example with 38B7 Group microcomputer (17 to 32 digits)

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.

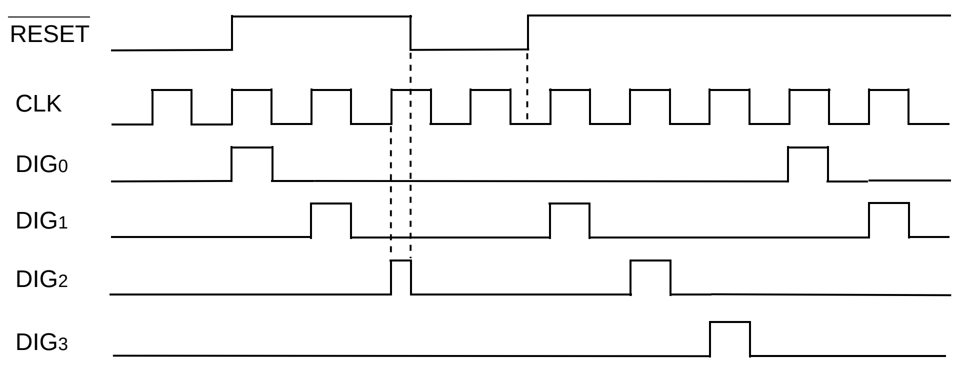


Fig. 3.9.11 Digit output waveform when reset signal is input

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 RESET pin and Vss at the shortest distance. Consequently, the RESET pin should be held at “L” until the minimum operation guaranteed voltage is reached.

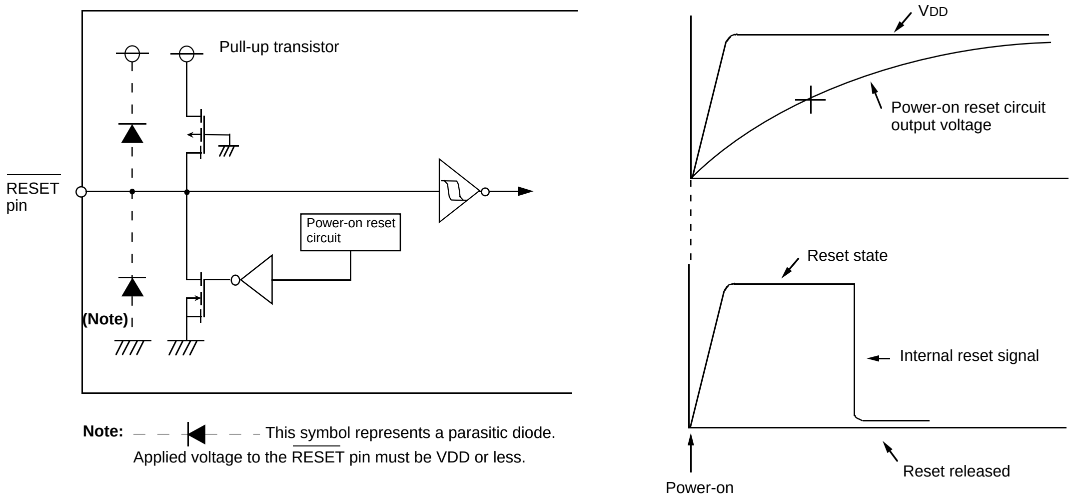


Fig. 3.9.12 Power-on reset circuit

Table 3.9.2 Absolute maximum ratings

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Power source voltage	•All voltages are based on V_{SS}. •Output transistors are 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 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 3.9.3 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 3.9.4 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 3.9.5 Electrical characteristics (V_{CC} = 4.0 to 5.5 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits Min.	Typ.	Max.	Unit
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} $\overline{\text{RESET}}$	V_{CC} = 5.0 V		0.4		V
I_{IH}	“H” input current	OVF_{IN} $\overline{\text{RESET}}$	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	$\overline{\text{RESET}}$	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 3.9.6 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

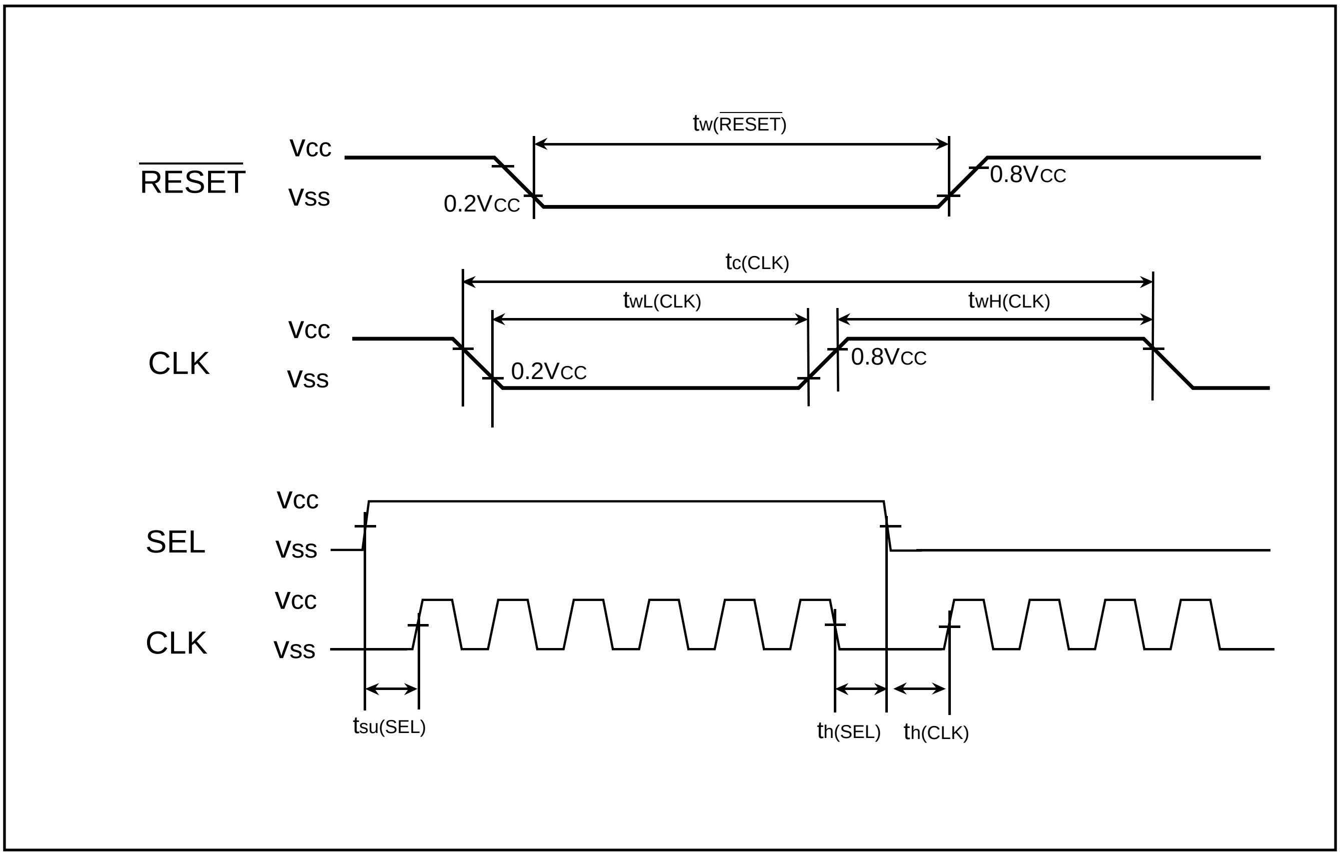


Fig. 3.9.13 Timing diagram

3.10 SFR memory map

Address	Register
0000_{16}	Port P0 (P0)
0001_{16}	
0002_{16}	Port P1 (P1)
0003_{16}	Port P1 direction register (P1D)
0004_{16}	Port P2 (P2)
0005_{16}	
0006_{16}	Port P3 (P3)
0007_{16}	Port P3 direction register (P3D)
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}	Port PA (PA)
0015_{16}	Port PA direction register (PAD)
0016_{16}	Port PB (PB)
0017_{16}	Port PB direction register (PBD)
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)

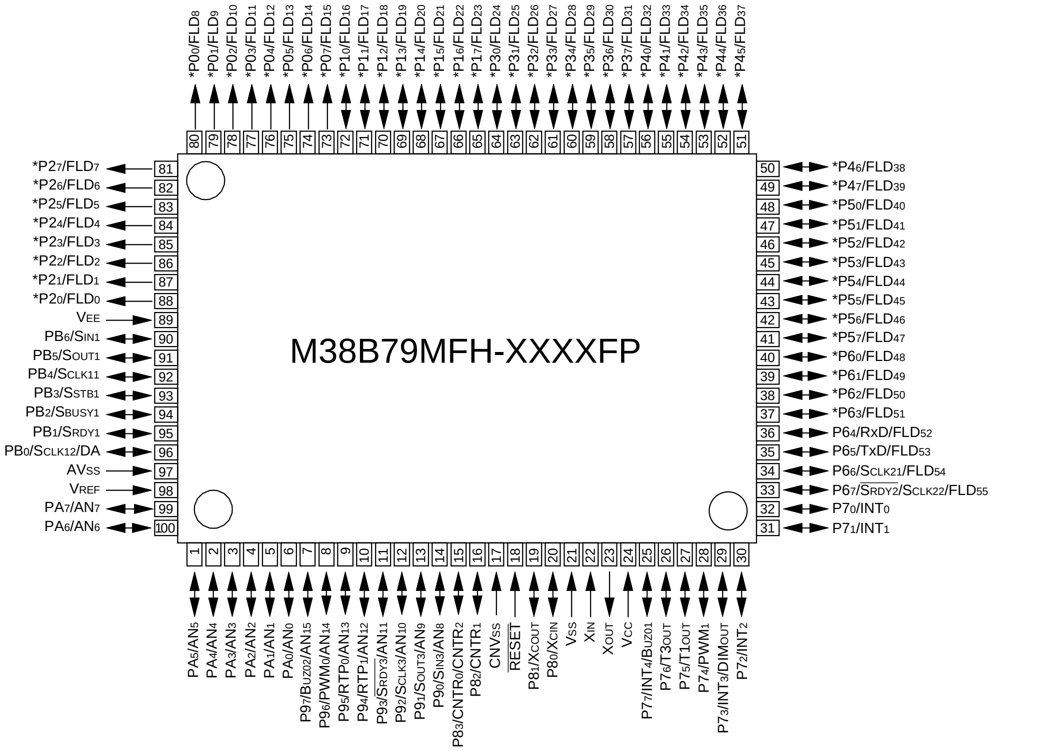
Address	Register
0020_{16}	Timer 1 (T1)
0021_{16}	Timer 2 (T2)
0022_{16}	Timer 3 (T3)
0023_{16}	Timer 4 (T4)
0024_{16}	Timer 5 (T5)
0025_{16}	Timer 6 (T6)
0026_{16}	PWM control register (PWMCON)
0027_{16}	Timer 6 PWM register (T6PWM)
0028_{16}	Timer 12 mode register (T12M)
0029_{16}	Timer 34 mode register (T34M)
$002A_{16}$	Timer 56 mode register (T56M)
$002B_{16}$	D-A conversion register (DA)
$002C_{16}$	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}	AD/DA control register (ADCON)
0033_{16}	A-D conversion register (low-order) (ADL)
0034_{16}	A-D conversion register (high-order) (ADH)
0035_{16}	PWM register (high-order) (PWMH)
0036_{16}	PWM register (low-order) (PWML)
0037_{16}	Baud rate generator (BRG)
0038_{16}	UART control register (UARTCON)
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)

Address	Register
$0EEC_{16}$	Serial I/O3 control register (SIO3CON)
$0EED_{16}$	Serial I/O3 register (SIO3)
$0EEE_{16}$	Watchdog timer control register (WDTCON)
$0EEF_{16}$	Pull-up control register 3 (PULL3)
$0EF0_{16}$	Pull-up control register 1 (PULL1)
$0EF1_{16}$	Pull-up control register 2 (PULL2)
$0EF2_{16}$	Port P0 digit output set switch register (P0DOR)
$0EF3_{16}$	Port P2 digit output set switch register (P2DOR)
$0EF4_{16}$	FLDC mode register (FLDM)
$0EF5_{16}$	Tdisp time set register (TDISP)

Address	Register	
$0EF6_{16}$	Toff1 time set register (TOFF1)	
$0EF7_{16}$	Toff2 time set register (TOFF2)	
$0EF8_{16}$	FLD data pointer (FLDDP)	
$0EF9_{16}$	Port P4 FLD/Port switch register (P4FPR)	
$0EFA_{16}$	Port P5 FLD/Port switch register (P5FPR)	
$0EFB_{16}$	Port P6 FLD/Port switch register (P6FPR)	
$0EFC_{16}$	FLD output control register (FLDCON)	
$0EFD_{16}$	Buzzer output control register (BUZCON)	
$0EFE_{16}$	Flash memory control register (FCON)	**(Note)**
$0EFF_{16}$	Flash command register (FCMD)	**(Note)**

Note: Flash memory version only.

3.11 Pin configuration



Package type: 100P6S-A

MITSUBISHI SEMICONDUCTORS
USER'S MANUAL
38B7 Group

Editioned by
Committee of editing of Mitsubishi Semiconductor User's Manual

Published by
Mitsubishi Electric Corp., Semiconductor Marketing Division

User's Manual
38B7 Group

Renesas Technology Corp.
Nippon Bldg.,6-2,Otemachi 2-chome,Chiyoda-ku,Tokyo,100-0004 Japan

New publication, effective Jan. 2003.
Specifications subject to change without notice.