To all our customers

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

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

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

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

MITSUBISHI 8-BIT SINGLE-CHIP MICROCOMPUTER
740 FAMILY / 7200 SERIES

7220 Group

User's Manual

RENESAS
Renesas Technology Corp.

keep safety first in your circuit designs !

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

Notes regarding these materials

- These materials are intended as a reference to assist our customers in the selection of the Mitsubishi semiconductor product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Mitsubishi Electric Corporation or a third party.

- Mitsubishi Electric Corporation assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts or circuit application examples contained in these materials.

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

- Mitsubishi Electric Corporation semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.

- The prior written approval of Mitsubishi Electric Corporation is necessary to reprint or reproduce in whole or in part these materials.

- If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
  Any diversion or reexport contrary to the export control laws and regulations of JAPAN and/or the country of destination is prohibited.

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

Preface

This manual describes the hardware of the Mitsubishi CMOS 8-bit microcomputers 7220 group.
After reading this manual, the user should have a through knowledge of the functions and features of 7220 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 **"SERIES 740 <SOFTWARE> USER'S MANUAL."**
For details of development support tools, refer to the **"DEVELOPMENT SUPPORT TOOLS FOR MICROCOMPUTERS"** data book.

BEFORE USING THIS MANUAL

This user's manual consists of the following chapters. Refer to the chapter appropriate to your conditions, such as hardware design or software development. The M37221M6-XXXSP/FP is used as a general example in describing the functions of the 7220 group, unless other wise noted.

1. Organization

● **CHAPTER 1 HARDWARE**

This chapter describes features of the microcomputer, pin configuration, pin description, functional block diagram.

● **CHAPTER 2 FUNCTIONAL DESCRIPTION**

This chapter describes operation of each peripheral function.

● **CHAPTER 3 ELECTRIC CHARACTERISTICS**

This chapter describes electric characteristics and standard characteristics.

● **CHAPTER 4 M37220M3-XXXSP/FP**

This chapter describes differences between the M37220M3-XXXSP/FP and M37221M6-XXXSP/FP.

● **CHAPTER 5 APPLICATION**

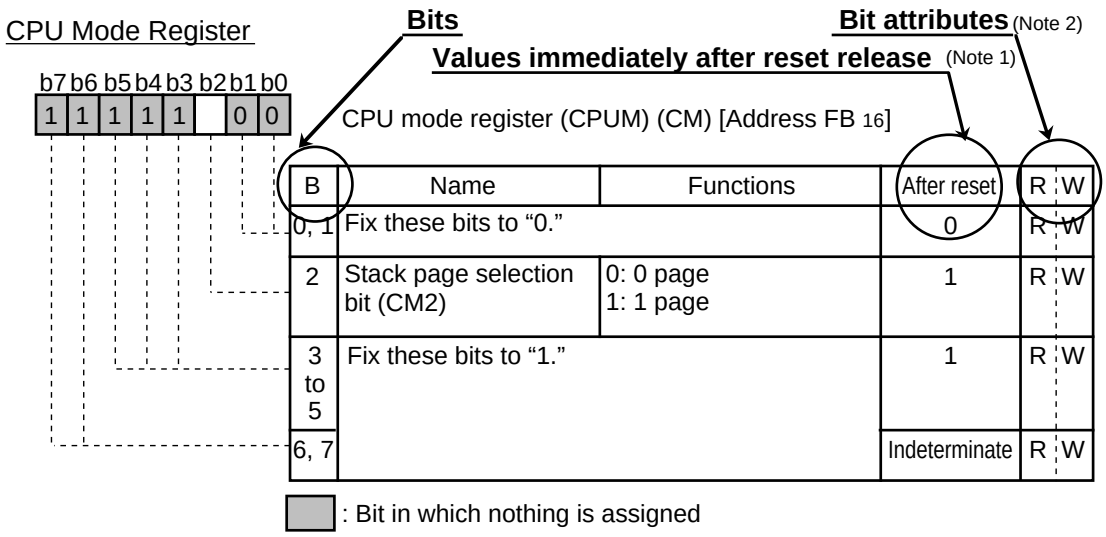
This chapter describes usage and application examples of peripheral functions, based mainly on setting examples of related registers.

● **CHAPTER 6 APPENDIX**

This chapter includes precautions for systems development using the microcomputer, a list of control registers, the mask ROM confirmation forms (mask ROM version) and mark specification forms which are to be submitted when ordering.

2. Register diagram

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



Notes 1: Values immediately after reset release

0••••••“0” after reset release
1••••••“1” after reset release
?••••••Indeterminate after reset release

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

R••••••Read
R ••••••Read enabled
— ••••••Read disabled

W••••••Write
W ••••••Write enabled
— ••••••Write disabled
✻ ••••••“0” can be set by software, but “1” cannot be set.

Table of contents

CHAPTER 1. OVERVIEW

CHAPTER 2. FUNCTIONAL DESCRIPTION

CHAPTER 4. M37220M3-XXXSP/FP

CHAPTER 5. APPLICATION

CHAPTER 6. APPENDIX

List of figures

CHAPTER 1. OVERVIEW

CHAPTER 2. FUNCTIONAL DESCRIPTION

List of figures

CHAPTER 3. ELECTRICAL CHARACTERISTICS

CHAPTER 4. M37220M3-XXXSP/FP

CHAPTER 5. APPLICATION

CHAPTER 6. APPENDIX

List of figures

List of tables

CHAPTER 1. OVERVIEW

CHAPTER 2. FUNCTIONAL DESCRIPTION

CHAPTER 4. M37220M3-XXXSP/FP

CHAPTER 5. APPLICATION

CHAPTER 6. APPENDIX

CHAPTER 1
OVERVIEW

1.1 Performance overview

The 8-bit microcomputers:
-M37221M4-XXXSP
-M37221M6-XXXSP/FP
-M37221M8-XXXSP
-M37221MA-XXXSP
-M37220M3-XXXSP/FP
have their simple instruction set; the ROM, RAM, and I/O addresses are placed on the same memory map to enable easy programming.
Furthermore, they have many additional functions for tuning system for TV:

- PWM output (14-bit and 8-bit)
- CRT display
- A-D comparator (resistance string method)
- Software runaway detection
- Multi-master I^2C-BUS interface function
- ROM correction function

And also, they can allow low power dissipation by the use of CMOS processing.
The M37221M6-XXXSP/FP is used as a general example in describing the functions of the above microcomputers, unless otherwise noted.

The performance overview is shown in Table 1.1.1.

M37220M3-XXXSP/FP

Refer to **"CHAPTER 4. M37220M3-XXXSP/FP."**

Table 1.1.1 Performance overview (1)

Parameter			Performance	
Number of basic instructions			71	
Instruction execution time			0.5 μs (the minimum instruction execution time, at 8 MHz oscillation frequency)	
Clock frequency			8 MHz (maximum)	
Memory size	M37221M4-XXXSP		ROM	16 K bytes
			RAM	320 bytes
	M37221M6-XXXSP/FP		ROM	24 K bytes
			RAM	384 bytes
	M37221M8-XXXSP		ROM	32 K bytes
			RAM	512 bytes
	M37221MA-XXXSP		ROM	40 K bytes
			RAM	640 bytes
	CRT ROM		8 K bytes	
	CRT RAM		96 bytes	
Input/Output ports	$P0_0$–$P0_7$	I/O	8-bit × 1 (N-channel open-drain output structure, can be used as PWM output pins, INT input pins, A-D input pin)	
	$P1_0$, $P1_5$–$P1_7$	I/O	4-bit × 1 (CMOS input/output structure, can be used as CRT output pin, A-D input pins, INT input pin)	
	$P1_1$–$P1_4$	I/O	4-bit × 1 (CMOS input/output structure, can be used as multi-master I^2C-BUS interface)	
	$P2_0$, $P2_1$	I/O	2-bit × 1 (CMOS input/output or N-channel open-drain output structure, can be used as serial I/O pins)	
	$P2_2$–$P2_7$	I/O	6-bit × 1 (CMOS input/output structure, can be used as serial input pin, external clock input pins)	
	$P3_0$, $P3_1$	I/O	2-bit × 1 (CMOS input/output or N-channel open-drain output structure, can be used as A-D input pins)	
	$P3_2$	I/O	1-bit × 1 (N-channel open-drain output structure)	
	$P3_3$, $P3_4$	Input	2-bit × 1 (can be used as CRT display clock I/O pins)	
	$P5_2$–$P5_5$	Output	4-bit × 1 (CMOS output structure, can be used as CRT output pins)	
Serial I/O			8-bit × 1	
Multi-master I^2C-BUS interface			1 (2 systems)	
A-D comparator			6 channels (6-bit resolution)	
PWM output circuit			14-bit × 1, 8-bit × 6	
Timers			8-bit timer × 4	
ROM correction function (See note)			32 bytes × 2	

1.1 Performance overview

Table 1.1.2 Performance overview (2)

Parameter		Performance
Subroutine nesting	M37221M4-XXXSP M37221M6-XXXSP/FP	96 levels (maximum)
	M37221M8-XXXSP M37221MA-XXXSP	128 levels (maximum)
Interrupt		External interrupt × 3, Internal timer interrupt × 4, Serial I/O interrupt × 1, CRT interrupt × 1, Multi-master I^2C-BUS interface interrupt × 1, $f(X_{IN})/4096$ interrupt × 1, V_{SYNC} interrupt × 1, BRK interrupt × 1
Clock generating circuit		2 built-in circuits (externally connected to a ceramic resonator or a quartz-crystal oscillator)
Power source voltage		5 V ± 10 %
Power dissipation	CRT ON	165 mW typ. (at oscillation frequency $f(X_{IN})$ = 8 MHz, f_{CRT} = 8 MHz)
	CRT OFF	110 mW typ. (at oscillation frequency $f(X_{IN})$ = 8 MHz)
	In stop mode	1.65 mW (maximum)
12V withstand ports		6
LED drive ports		4
Operating temperature range		–10 °C to 70 °C
Device structure		CMOS silicon gate process
Package	M37221M4-XXXSP M37221M6-XXXSP M37221M8-XXXSP M37221MA-XXXSP	42-pin shrink plastic molded DIP
	M37221M6-XXXFP	42-pin shrink plastic molded SOP
CRT display function	Number of display characters	24 characters × 2 lines (maximum 16 lines by software)
	Dot structure	12 × 16 dots
	Kinds of characters	256 kinds
	Kinds of character sizes	3 kinds
	Kinds of character colors	Maximum 7 kinds (R, G, B); can be specified by the character
	Display position (horizontal, vertical)	64 levels (horizontal) × 128 levels (vertical)

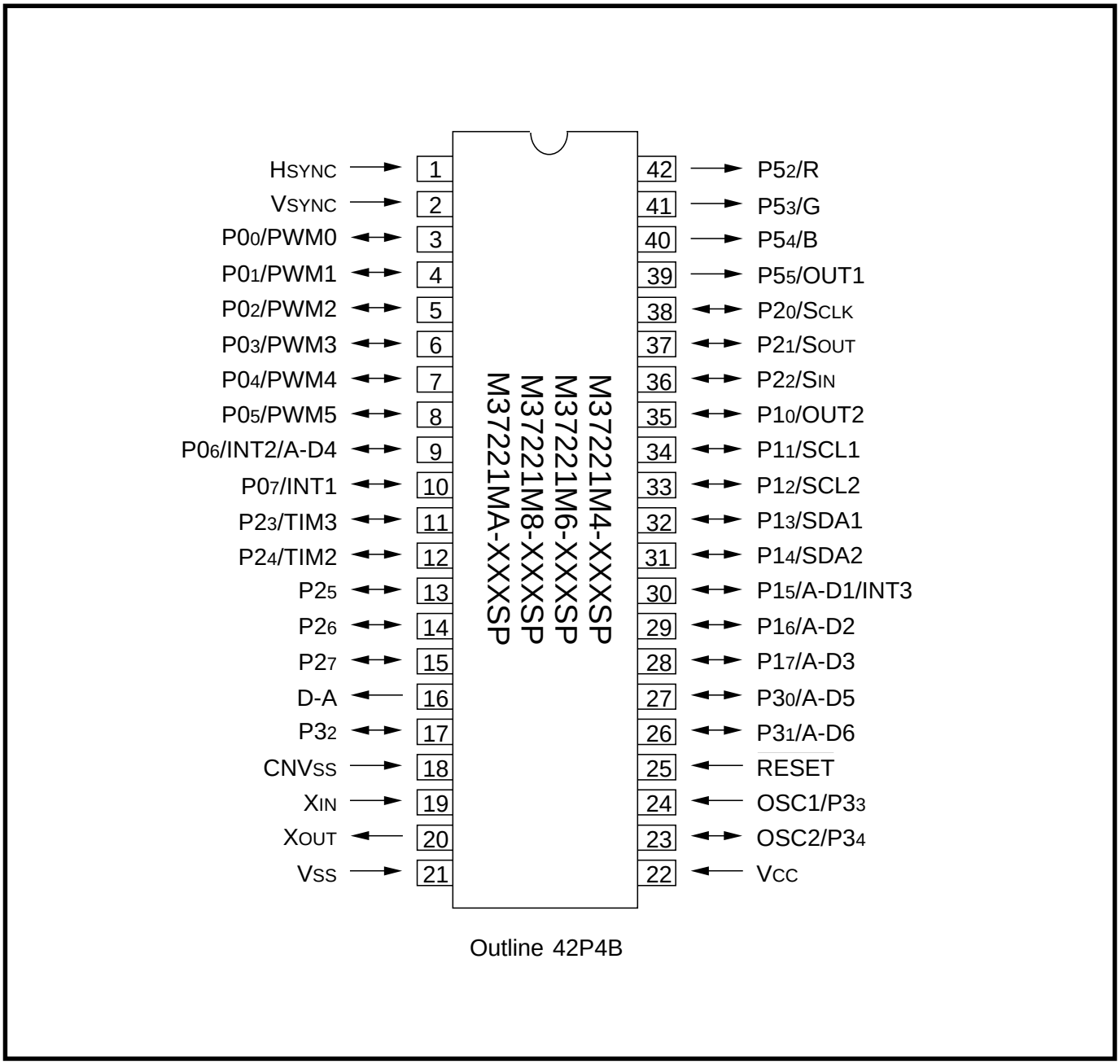
Note: Only M37221M8-XXXSP and M37221MA-XXXSP have the function.

1.2 Pin configuration

The pin configurations are shown in Figures 1.2.1 and 1.2.2.

M37220M3-XXXSP/FP

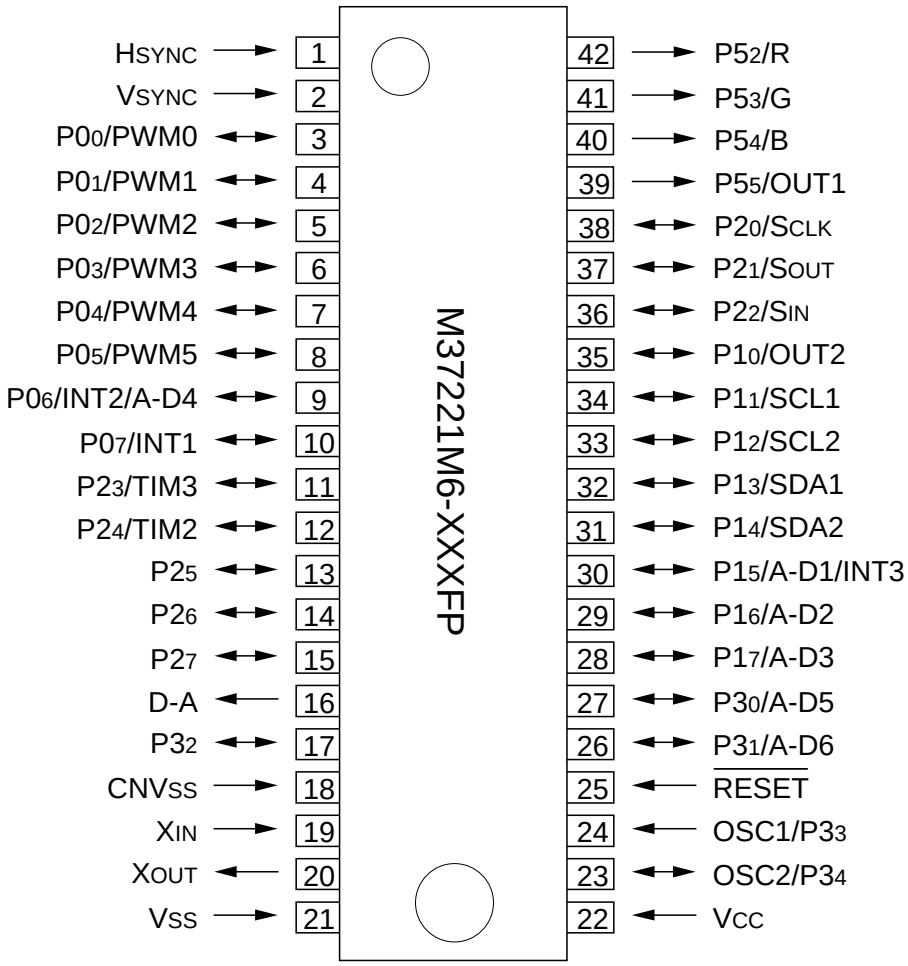
Refer to **"CHAPTER 4. M37220M3-XXXSP/FP."**



Outline 42P4B

Fig. 1.2.1 Pin configuration (top view) (1)

1.2 Pin configuration



Outline 42P2R-A

Fig. 1.2.2 Pin configuration (top view) (2)

1.3 Pin description

The pin description is shown in Table 1.3.1.

M37220M3-XXXSP/FP

Refer to **"CHAPTER 4. M37220M3-XXXSP/FP."**

Table 1.3.1 Pin description (1)

Pin	Name	Input/ Output	Functions
V_{CC}, V_{SS}	Power source		Apply voltage of 5 V ± 10 % (typical) to V_{CC}, and 0 V to V_{SS}.
CNV_{SS}	CNV_{SS}		Connected to V_{SS}.
$\overline{RESET}$	Reset input	Input	To enter the reset state, the reset input pin must be kept at a "L" for 2 μs or more (under normal V_{CC} conditions). If more time is needed for the quartz-crystal oscillator to stabilize, this "L" condition should be maintained for the required time.
X_{IN}	Clock input	Input	This chip has an internal clock generating circuit. To control generating frequency, an external ceramic resonator or a quartz-crystal oscillator is connected between pins X_{IN} and X_{OUT}. If an external clock is used, the clock source should be connected to the X_{IN} pin and the X_{OUT} pin should be left open.
X_{OUT}	Clock output	Output	
$P0_0$ PWM0–$P0_5$/ PWM5, $P0_6$/INT2/ A-D4, $P0_7$/INT1	I/O port P0	I/O	Port P0 is an 8-bit I/O port with direction register allowing each I/O bit to be individually programmed as input or output. At reset, this port is set to input mode. The output structure is N-channel open-drain output.
	PWM output	Output	Pins $P0_0$–$P0_5$ are also used as PWM output pins PWM0–PWM5 respectively. The output structure is N-channel open-drain output.
	External interrupt input	Input	Pins $P0_6$, $P0_7$ are also used as external interrupt input pins INT2, INT1 respectively.
	Analog input	Input	$P0_6$ pin is also used as analog input pin A-D4.
$P1_0$/OUT2, $P1_1$/SCL1, $P1_2$/SCL2, $P1_3$/SDA1, $P1_4$/SDA2, $P1_5$/A-D1/ INT3, $P1_6$/A-D2, $P1_7$/A-D3	I/O port P1	I/O	Port P1 is an 8-bit I/O port and has basically the same functions as port P0. The output structure is CMOS output.
	CRT output	Output	Pins $P1_0$ is also used as CRT output pin OUT2. The output structure is CMOS output.
	Multi-master I^2C-BUS interface	I/O	Pins $P1_1$–$P1_4$ are used as SCL1, SCL2, SDA1 and SDA2 respectively, when multi-master I^2C-BUS interface is used. The output structure is N-channel open-drain output.
	Analog input	Input	Pins $P1_5$–$P1_7$ are also used as analog input pins A-D1 to A-D3 respectively.
	External interrupt input	Input	$P1_5$ pin is also used as external interrupt input pin INT3.

1.3 Pin description

Table 1.3.2 Pin description (2)

Pin	Name	Input/ Output	Functions
$P2_0/S_{CLK}$, $P2_1/S_{OUT}$, $P2_2/S_{IN}$, $P2_3$/TIM3, $P2_4$/TIM2, $P2_5$–$P2_7$	I/O port P2	I/O	Port P2 is an 8-bit I/O port and has basically the same functions as port P0. The output structure is CMOS output.
	External clock input	Input	Pins $P2_3$, $P2_4$ are also used as external clock input pins TIM3, TIM2 respectively.
	Serial I/O synchronous clock input/ output	I/O	$P2_0$ pin is also used as serial I/O synchronous clock input/output pin S_{CLK}. The output structure is N-channel open-drain output.
	Serial I/O data input/output	I/O	Pins $P2_1$, $P2_2$ are also used as serial I/O data input/output pins S_{OUT}, S_{IN} respectively. The output structure is N-channel open-drain output.
$P3_0$/A-D5/ DA1, $P3_1$/A-D6/ DA2, $P3_2$	I/O port P3	I/O	Ports $P3_0$–$P3_2$ are 3-bit I/O ports and have basically the same functions as port P0. Either CMOS output or N-channel open-drain output structure can be selected as the port $P3_0$ and $P3_1$. The output structure of port $P3_2$ is N-channel open-drain output.
	Analog input	Input	Pins $P3_0$, $P3_1$ are also used as analog input pins A-D5, A-D6 respectively.
$P3_3$/OSC1, $P3_4$/OSC2	Input port P3	Input	Ports $P3_3$, $P3_4$ are 2-bit input ports.
	Clock input for CRT display	Input	$P3_3$ pin is also used as CRT display clock input pin OSC1.
	Clock output for CRT display	Output	$P3_4$ pin is also used as CRT display clock output pin OSC2. The output structure is CMOS output.
$P5_2$/R, $P5_3$/G, $P5_4$/B, $P5_5$/OUT1	Output port P5	Output	Ports $P5_2$–$P5_5$ are 4-bit output ports. The output structure is CMOS output.
	CRT output	Output	Pins $P5_2$–$P5_5$ are also used as CRT output pins R, G, B, OUT1 respectively. The output structure is CMOS output.
H_{SYNC}	H_{SYNC} input	Input	This is a horizontal synchronous signal input for CRT.
V_{SYNC}	V_{SYNC} input	Input	This is a vertical synchronous signal input for CRT.
D-A	DA output	Output	This is a 14-bit PWM output pin. The output structure is CMOS output.

1.4 Functional block diagram

The functional block diagram is shown in Figure 1.4.1.

M37220M3-XXXSP/FP

Refer to **"CHAPTER 4. M37220M3-XXXSP/FP."**

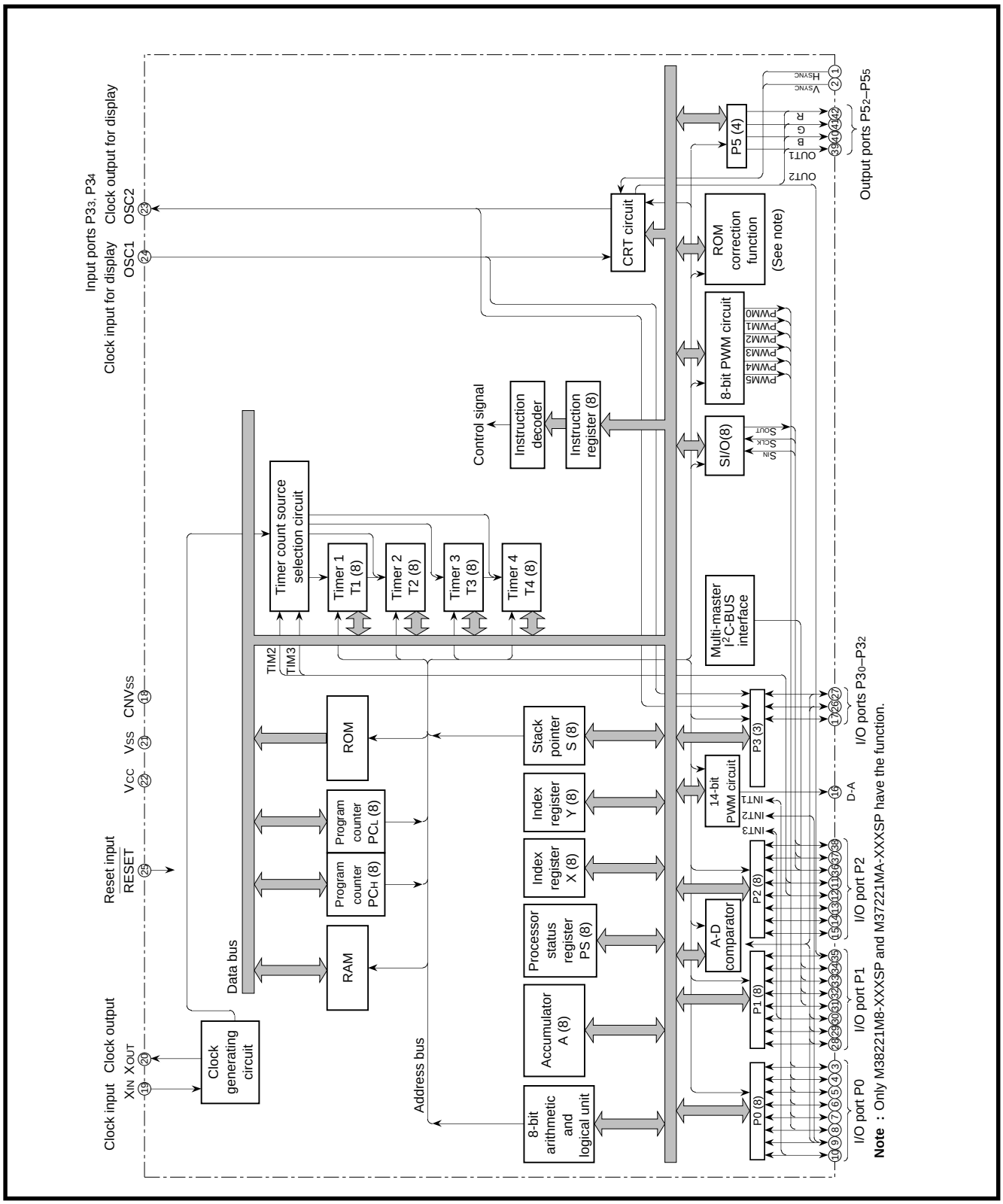


Fig. 1.4.1 Functional block diagram

CHAPTER 2

FUNCTIONAL DESCRIPTION

2.1 Central processing unit

The CPU of the M37221M6-XXXSP/FP has six main registers.
The program counter (PC) is a 16-bit register consists of PC_H and PC_L, both of which are 8-bit registers. The other five registers: the accumulator (A), index register X (X), index register Y (Y), stack pointer (S) and processor status register (PS), all have an 8-bit configuration.

Note: The contents of registers above except the following are indeterminate after a hardware reset. Therefore, initialize these registers by software.
- The Interrupt disable flag I of the processor status register = "1"
- The program counter = the contents of addresses $FFFE_{16}$ and $FFFF_{16}$

Figure 2.1.1 shows the registers configuration diagram of M37221M6-XXXSP/FP.

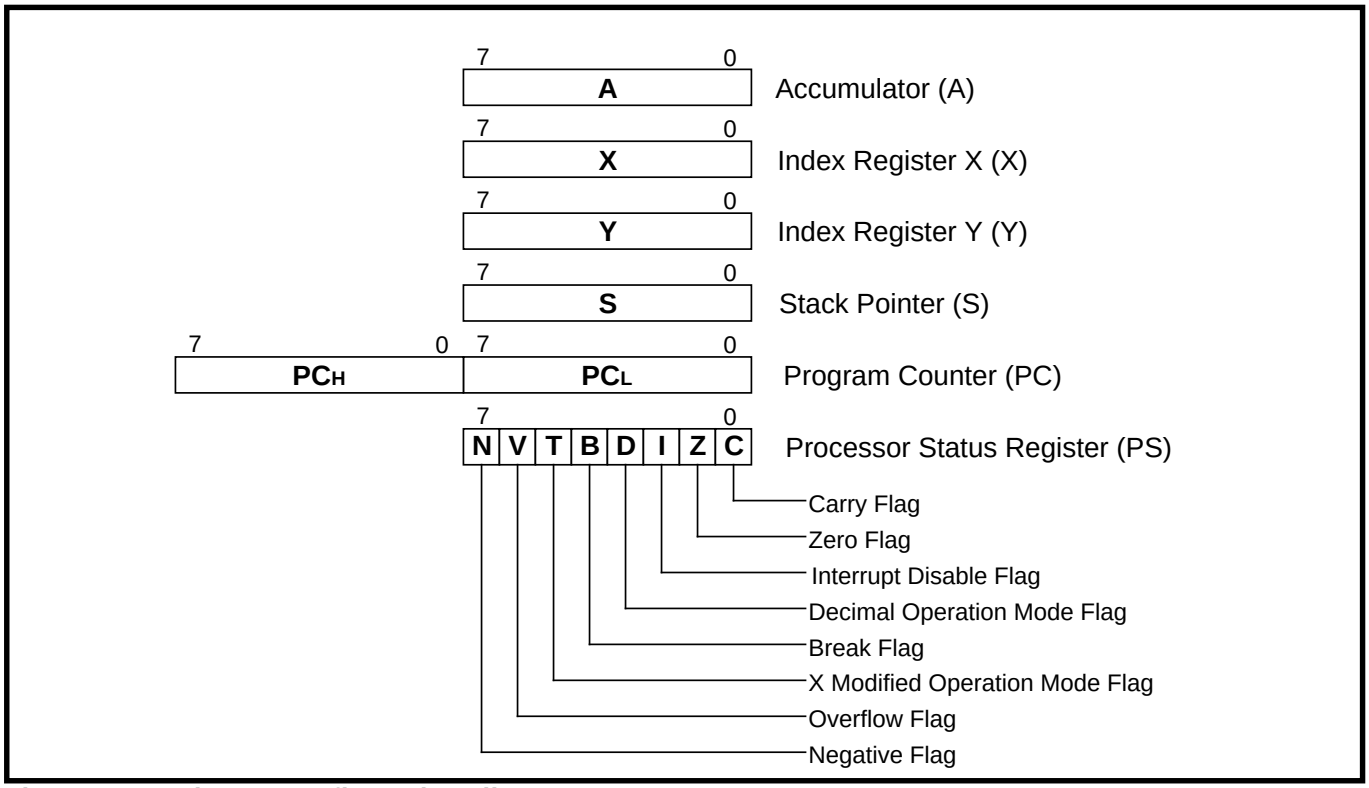


Fig. 2.1.1 Registers configuration diagram

2.1.1 Accumulator (A)

The accumulator is the central register of the microcomputer and 8-bit register.
This general-purpose register is used with considerable for arithmetic operations, data transfer, temporary clearing, condition judgments, etc.

2.1.2 Index register X (X), index register Y (Y)

The M37221M6-XXXSP/FP has the index register X and the index register Y, both of which are 8-bit registers.
In the addressing modes which use these index registers, the register contents are added to the specified address and this becomes the actual address. These modes are used for referencing subroutine tables and memory tables.
The index registers, which have increment, decrement, comparison and data transfer functions, are also used as simple accumulators.

2.1.3 Stack pointer (S)

The stack pointer is an 8-bit register used for interrupts and subroutine calls.
The stack area can be assigned into the internal RAM.
The internal RAM of M37221M6-XXXSP/FP is assigned in the zero page and the page 1. The both area can use for the stack area. The stack area is specified with the CPU mode register (address $00FB_{16}$). At reset, the stack area is specified to the page 1 automatically.

Note: Storing data in the stack area fills the RAM area with stored data in order, therefore make sure the depth of interrupt levels and the subroutine nesting.

The stack area and stack pointer (S) should be specified in the initialization of software. When the stack area is specified to "1," even if the value of stack pointer is over "00_{16}" (stack address is 0100_{16}), the stack area value never change to "0" automatically. Therefore in this case, change the stack area value by software.

CPU Mode Register

b7	b6	b5	b4	b3	b2	b1	b0
1	1	1	1	1		0	0

CPU mode register (CPUM) (CM) [Address $00FB_{16}$]

B	Name	Functions	After reset	R	W
0, 1	Fix these bits to "0."		0	R	W
2	Stack page selection bit (CM2)	0: 0 page (Note) 1: 1 page	1	R	W
3 to 5	Fix these bits to "1."		1	R	W
6, 7			Indeterminate	R	W

Note: This bit is set to "1" after reset release.

Fig. 2.1.2 CPU mode register

With the stack pointer during a interrupt or subroutine call, the processing is performed automatically in the following sequence (refer to **"Figure 2.1.3"**).

① The contents of high-order 8 bits of the program counter (PC_H) are stored at an address indicated as below:
- The high-order 8 bits are the stack area value ("00_{16}" or "01_{16}").
- The low-order 8 bits are the stack pointer contents.

② The stack pointer contents are decremented by 1.

③ The contents of low-order 8 bits of the program counter (PC_L) are stored at an address indicated as below:
- The high-order 8 bits are the stack area value ("00_{16}" or "01_{16}").
- The low-order 8 bits are the stack pointer contents.

④ The stack pointer contents are decremented by 1.

⑤ The contents of the processor status register (PS) are stored at an address indicated as below:
- The high-order 8 bits are the stack area value ("00_{16}" or "01_{16}").
- The low-order 8 bits are the stack pointer contents.

⑥ The stack pointer contents are decremented by 1.

2.1 Central processing unit

Storing of the processor status register in items ⑤ and ⑥ above is not performed during a subroutine call. Execute the **PHP** instruction in a program to push the processor status register onto a stack.
To prevent data from losing during interrupts and subroutine calls, push the other registers onto a stack by software as described above.
For example, execute the **PHA** instruction to push the accumulator contents onto a stack. Executing the **PHA** instruction stores the accumulator contents at an address indicated as below:

- The high-order 8 bits are the stack area value ("00_{16}" or "01_{16}").
- The low-order 8 bits are the stack pointer contents.

The stack pointer contents are then decremented by 1.

Execute the **RTI** instruction to return from an interrupt routine.
When the **RTI** instruction is executed, the processing is performed automatically in the following sequence (refer to **"Figure 2.1.3"**).

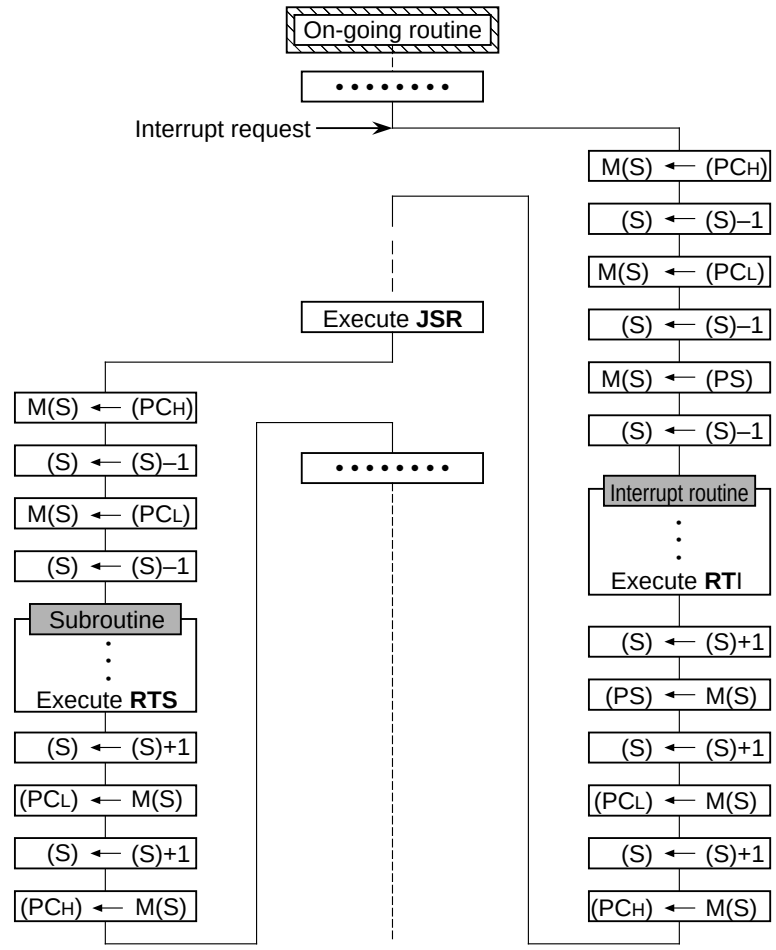
① The stack pointer contents are incremented by 1.
② The contents at the address indicated as below are restored to the processor status register.
- The high-order 8 bits are the stack area value ("00_{16}" or "01_{16}").
- The low-order 8 bits are the stack pointer contents.

③ The stack pointer contents are incremented by 1.
④ The contents at the address indicated as below are restored to low-order 8 bits of the program counter (PC_L).
- The high-order 8 bits are the stack area value ("00_{16}" or "01_{16}").
- The low-order 8 bits are the stack pointer contents.

⑤ The stack pointer contents are incremented by 1.
⑥ The contents at the address indicated as below are restored to high-order 8 bits of the program counter (PC_H).
- The high-order 8 bits are the stack area value ("00_{16}" or "01_{16}").
- The low-order 8 bits are the stack pointer contents.

Restoring of the processor status register in items ① and ② above is not performed in this case. Execute the **RTS** instruction to return from a subroutine.
Execute the **PLP** instruction and **PLA** instruction to restore the processor status register and the accumulator, respectively.
Executing the **PLP** (**PLA**) instruction increments the stack pointer by 1 and restores the contents at the address indicated as below to the processor status register.

- The high-order 8 bits are the stack area value ("00_{16}" or "01_{16}").
- The low-order 8 bits are the stack pointer.



(PCH) : Contents of high-order 8 bits of program counter
(PCL) : Contents of low-order 8 bits of program counter
(PS) : Contents of processor status register
(S) : Contents of stack pointer
M : Memory
RTI : Instruction for returning from interrupt routine to main routine
RTS : Instruction for returning from subroutine to main routine

Fig. 2.1.3 Sequence of push onto/pop from a stack during interrupts and subroutine calls

2.1.4 Program counter (PC)

The program counter is a 16-bit counter consists of PC_H and PC_L, both of which are 8-bit registers.
The program counter indicates the address of the program to be executed next.
The M37221M6-XXXSP/FP uses the stored program system. To start a new operation, transfer the instruction and the data, from the memory to the CPU. Ordinary, the program counter is controlled to indicate the memory address to be sent next. After each instruction is executed, the instruction required next is called out and this cycle is repeated until finished.

Note: The program counter of the M37221M6-XXXSP/FP is controlled automatically; however, make sure to avoid differences between program flow and the program counter contents when operating the stack pointer or directly changing the program counter contents.

2.1.5 Processor status register (PS)

The processor status register is an 8-bit register. It consists of 5 flags, which indicate the state after arithmetic operations related to the internal CPU, and 3 flags which determine operation.
The following explains each of these flags. Refer to **"6.9 Machine instruction table"** of this USER'S MANUAL or **"SERIES 740 <SOFTWARE> USER'S MANUAL"** concerning the change of these flags.

(1) Carry flag (C) ..... Bit 0
This flag stores any carry or borrow from the ALU after an arithmetic operation and is also changed by the Shift instruction or Rotate instruction.
This flag is set to "1" by using the **SEC** instruction and is cleared to "0" by using the **CLC** instruction.

(2) Zero flag (Z) ..... Bit 1
This flag is set to "1" when the result of an arithmetic operation or a data transfer is "0" and is cleared to "0" by any other result.
This flag has no meaning in the decimal mode.

(3) Interrupt disable flag (I) ..... Bit 2
This flag disables interrupts. When this flag is "1," all interrupts except the BRK interrupt and reset are disabled. This flag immediately becomes "1" when an interrupt is received. This flag is set to "1" by using the **SEI** instruction and is cleared to "0" by using the **CLI** instruction.

(4) Decimal operation mode flag (D) ..... Bit 3
This flag determines whether addition and substruction are performed in binary or decimal notation. Binary arithmetic is performed when this flag is "0" and decimal arithmetic is performed with treating each word as a 2-digit decimal when this flag is "1." Decimal adjust is performed automatically at this time. This flag is set to "1" by using the **SED** instruction and is cleared to "0" by using the **CLD** instruction. Only the **ADC** and **SBC** instructions are used for decimal arithmetic.
Since this flag directly affects calculations, always initialize it after a reset.

(5) Break flag (B) ..... Bit 4
This flag determines whether or not an interrupt occurred by using the **BRK** instruction. When a BRK instruction interrupt occurs, the flag B is set to "1"; for all other interrupts the flag is set to "0" and pushed to the stack.
For the M37221M6-XXXSP/FP, interrupt vectors by using the **BRK** instruction are independent of other interrupts, and it is possible to determine the cause of interrupt by jumping to the vector address inherent to each interrupt. Therefore, it is not specifically necessary to refer to this flag.

Note: The **BRK** instruction will be used for debugging.

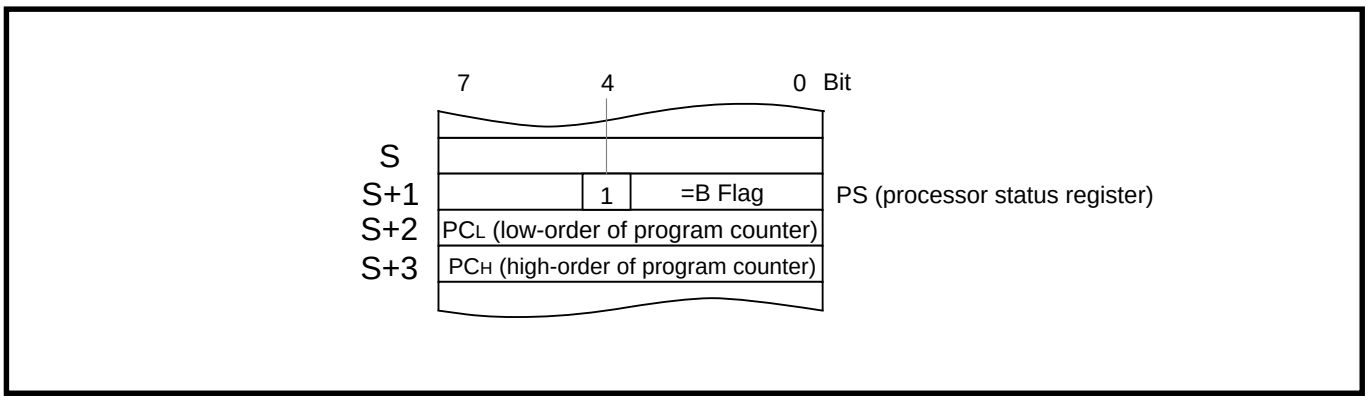


Fig. 2.1.4 Contents of stack after execution of BRK instruction

(6) X modified operation mode flag (T) ..... Bit 5

This flag determines whether arithmetic operations are performed via the accumulator or directly between memories. When the flag is set to "0", arithmetic operations are performed between the accumulator and memory. When "1," arithmetic operations are performed directly between memories. This flag is set to "1" with the **SET** instruction and is cleared to "0" with the **CLT** instruction.
Since this flag directly affects calculations, always initialize it after a reset.

■ **When the T flag = "0"**

A←A ✻ M

✻ : indicates an arithmetic operation
A : accumulator contents
M: contents of the memory specified by the addressing of the arithmetic operation

■ **When the T flag = "1"**

M1←M1 ✻ M2

✻ : indicates arithmetic operation
M1: contents of memory specified directly with index register X
M2: contents of the memory specified by the addressing of the arithmetic operation

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

This flag is set to "1" when an overflow occurs in the result of an arithmetic operation involving signs. An overflow occurs when the result of an addition or subtraction exceeds +127 ($7F_{16}$) or –128 (80_{16}). The **CLV** instruction clears the overflow flag to "0." There is no instruction for setting this flag to "1." When the **BIT** instruction is executed except the above, bit 6 of the memory executed by the **BIT** instruction is set to the overflow flag.
This flag has no meaning in decimal mode.

Note: Overflows do not occur when the result of an addition or subtraction is smaller than the above numerical values or an addition is performed between different signs.

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

This flag is set to "1" when the result of a data transfer or arithmetic operation is negative (bit 7 is "1"). When the **BIT** instruction is executed, bit 7 of the memory executed by the **BIT** instruction is set to the negative flag. This flag can be used to determine whether the results of arithmetic operations are positive or negative, and also to perform a simple bit test. There are no instructions for directly setting or clearing this flag.
This flag has no meaning in decimal mode.

2.2 Access area

The ROM, RAM and various I/O control registers are assigned within the same memory area. Therefore, the same instructions are used for data transfers and arithmetic operations without making any distinction between memory and I/O.

Since the program counter is a 16-bit register, 64 K-byte memory area can be accessed: from addresses as 0000_{16} to $FFFF_{16}$.

The first 256 bytes of the 64 K-byte memory area are called the “zero page” and the last 256 bytes are called the “special page.” These areas can be accessed with only 2 bytes by using each special addressing mode.

M37220M3-XXXSP/FP

Refer to **“CHAPTER 4. M37220M3-XXXSP/FP.”**

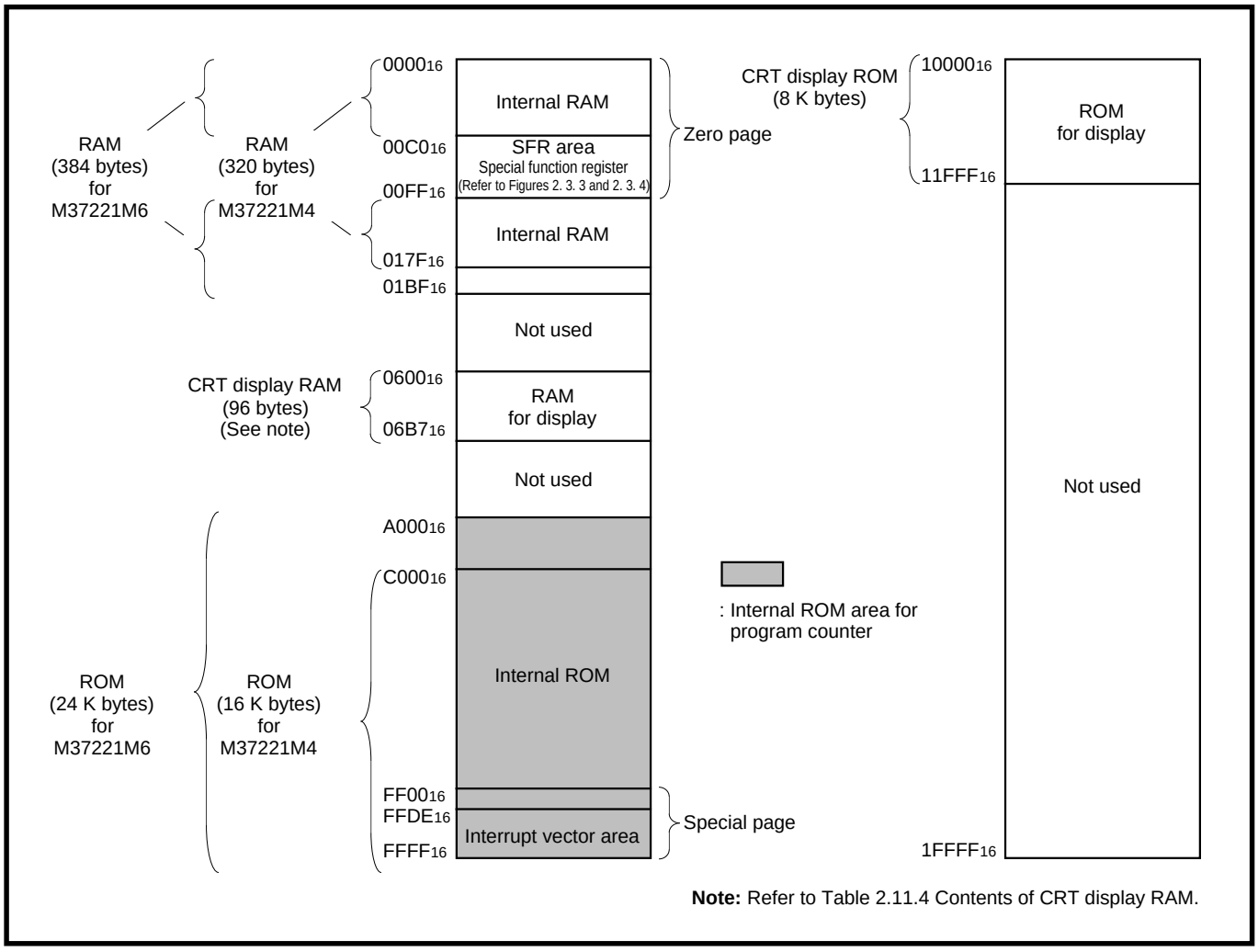
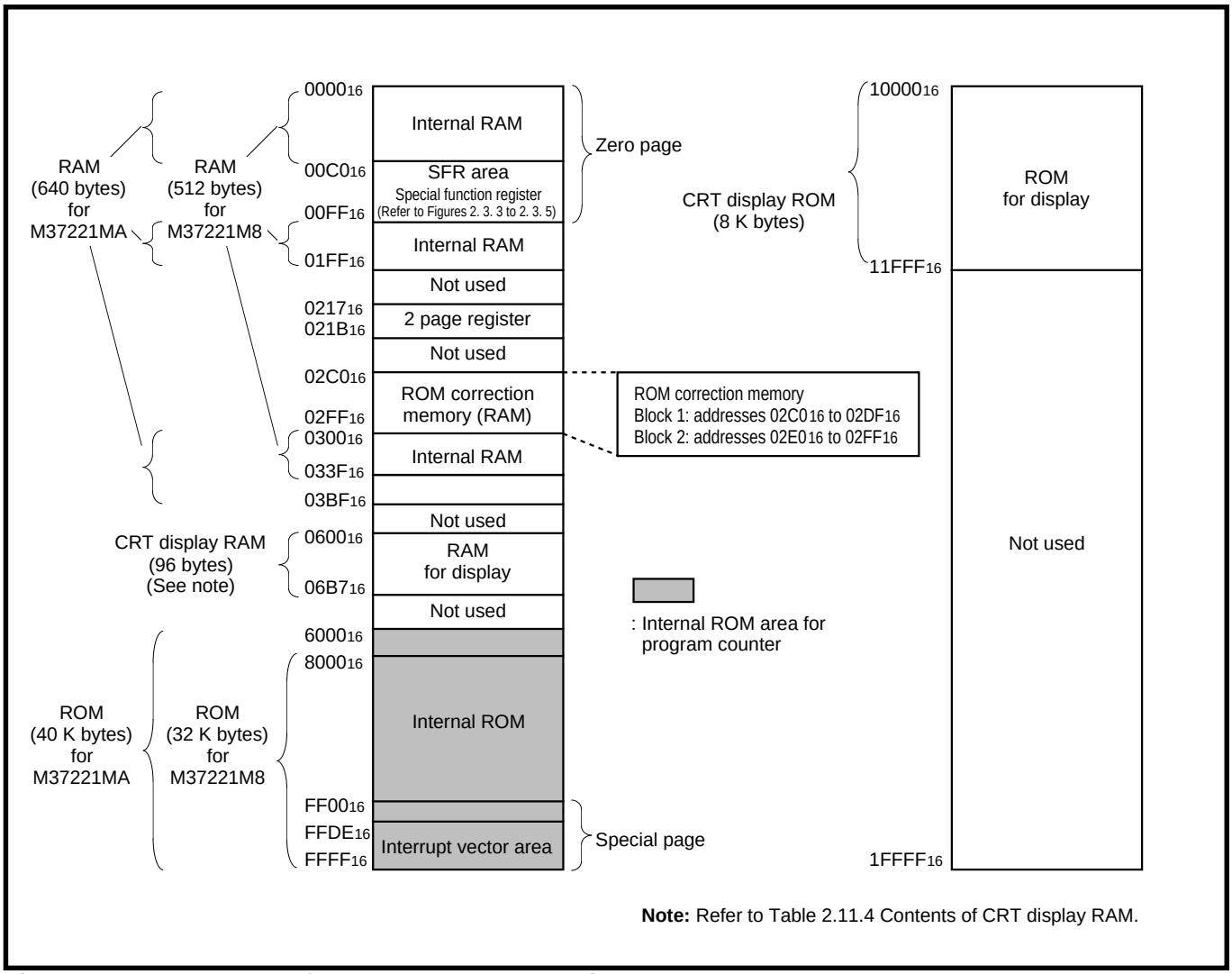


Fig. 2.2.1 Access area of M37221M4-XXXSP and M37221M6-XXXSP/FP



Note: Refer to Table 2.11.4 Contents of CRT display RAM.

Fig. 2.2.2 Access area of M37221M8-XXXSP and M37221MA-XXXSP

2.2 Access area

2.2.1 Zero page (addresses 0000_{16} to $00FF_{16}$)

The 256 bytes from address 0000_{16} to address $00FF_{16}$ are called “zero page”.

The internal RAM, I/O ports, timer, serial I/O, A-D comparison, PWM output, CRT display and interrupt related registers all present within this area.

These registers were called “special function registers” in distinction from the accumulator, index registers and so on in the CPU.

The addressing modes as shown in Table 2.2.1 are used to specify memory (RAM) and special function registers in the zero page area.

Those modes dedicated to the zero page area are marked with a symbol (✻).

This area can be accessed with shorter instructions by using these modes.

Table 2.2.1 Zero page addressing

Addressing mode	Bytes required
✻ Zero page	2
✻ Zero page Indirect	2
✻ Zero page X	2
✻ Zero page Y	2
✻ Zero page Bit	2
✻ Zero page Bit Relative	3
Absolute	3
Absolute X	3
Absolute Y	3
Relative	2
Indirect	3
Indirect X	2
Indirect Y	2

2.2.2 Special page (addresses $FF00_{16}$ to $FFFF_{16}$)

The 256 bytes from address $FF00_{16}$ to address $FFFF_{16}$ within the internal ROM are called “special page area”.

The addressing modes as shown in Table 2.2.2 are used to specify memory in the special page area.

Those modes dedicated to the special page area are marked with a symbol (✻).

This area can be accessed with shorter instructions by using these modes.

Subroutines used with considerable frequency are ordinary assigned in this area.

Table 2.2.2 Special page addressing

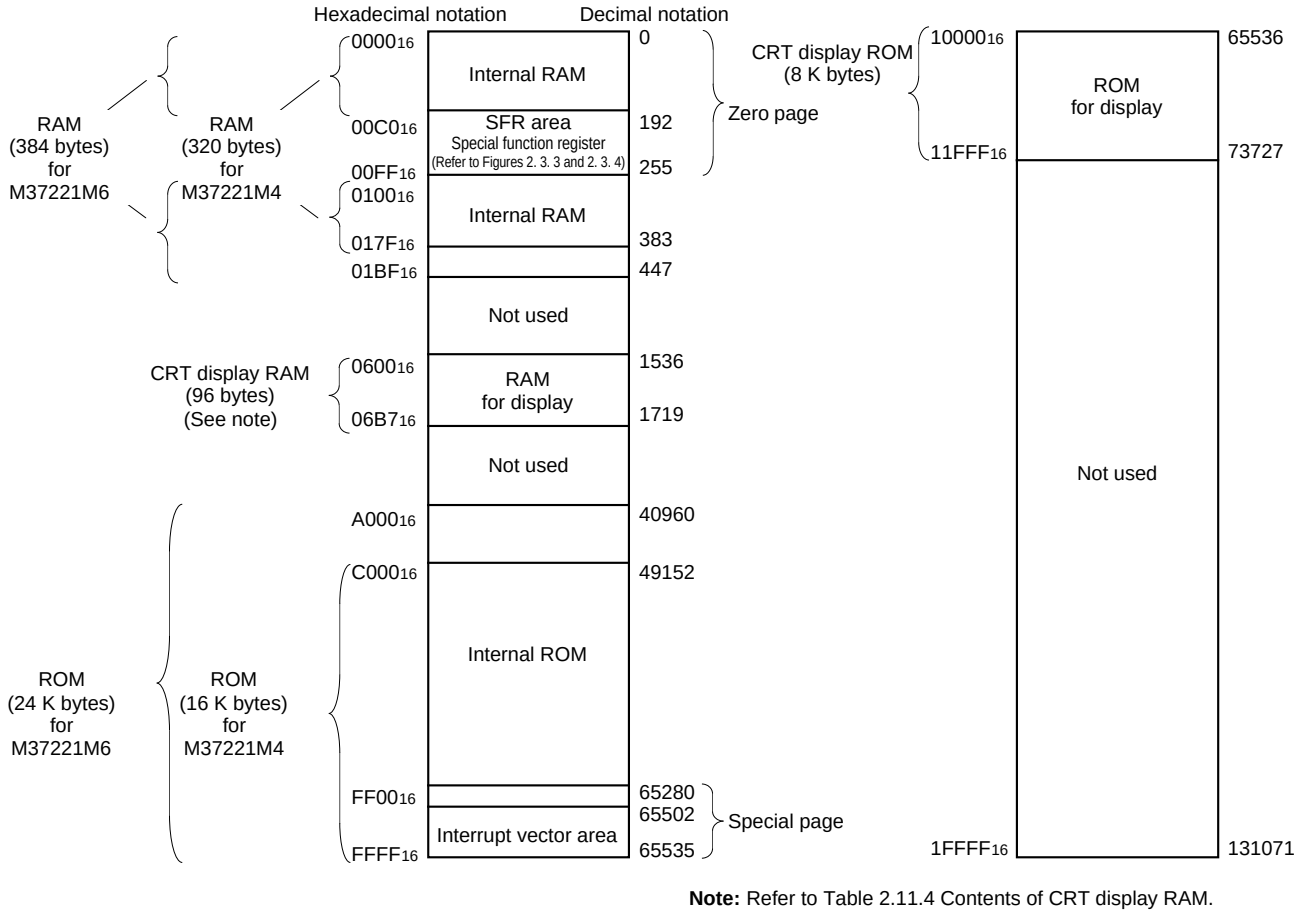
Addressing mode	Bytes required
✻ Special page	2
Absolute	3
Absolute X	3
Absolute Y	3
Relative	2
Indirect	3
Indirect X	2
Indirect Y	2

2.3 Memory assignment

Figures 2.3.1 and 2.3.2 show the memory assignment. The ROM, RAM and I/O assigned in this memory area are described below.

M37220M3-XXXSP/FP

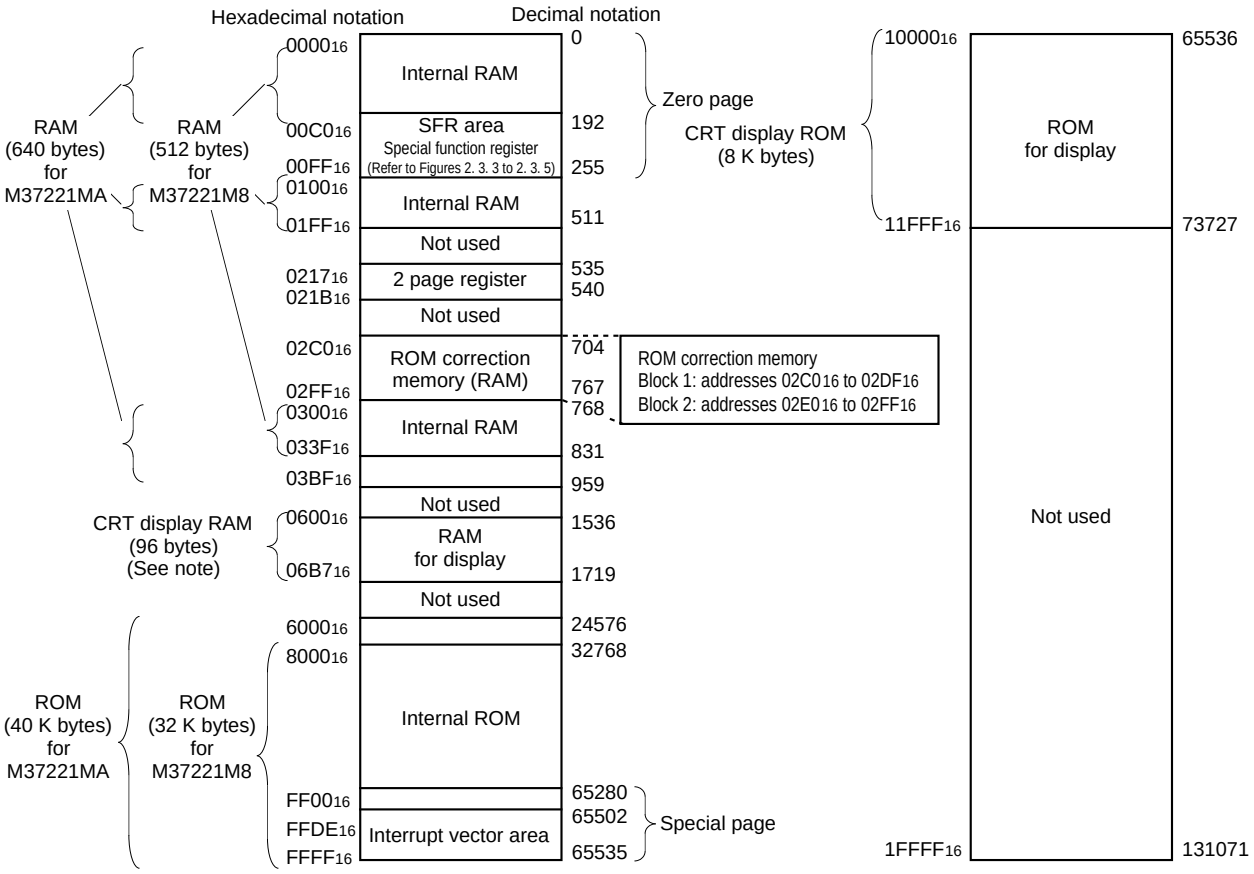
Refer to **"CHAPTER 4. M37220M3-XXXSP/FP."**



Note: Refer to Table 2.11.4 Contents of CRT display RAM.

Fig. 2.3.1 Memory assignment of M37221M4-XXXSP and M37221M6-XXXSP/FP

2.3 Memory assignment



Note: Refer to Table 2.11.4 Contents of CRT display RAM.

Fig. 2.3.2 Memory assignment of M37221M8-XXXSP and M37221MA-XXXSP

■SFR Area (addresses $C0_{16}$ to DF_{16})

<Bit allocation>

: } Function bit

Name

: No function bit

0 : Fix this bit to "0" (do not write "1")

1 : Fix this bit to "1" (do not write "0")

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately after reset

Address	Register	Bit allocation b7							b0	State immediately after reset b7							b0
$C0_{16}$	Port P0 (P0)									?							
$C1_{16}$	Port P0 direction register (D0)									00_{16}							
$C2_{16}$	Port P1 (P1)									?							
$C3_{16}$	Port P1 direction register (D1)									00_{16}							
$C4_{16}$	Port P2 (P2)									?							
$C5_{16}$	Port P2 direction register (D2)									00_{16}							
$C6_{16}$	Port P3 (P3)									0	0	0	?	?	?	?	?
$C7_{16}$	Port P3 direction register (D3)									00_{16}							
$C8_{16}$										?							
$C9_{16}$										?							
CA_{16}	Port P5 (P5)									0	0	?	?	?	?	?	?
CB_{16}	Port P5 direction register (D5)									00_{16}							
CC_{16}										?							
CD_{16}	Port P3 output mode control register (P3S)					0	0	P31S	P30S	00_{16}							
CE_{16}	DA-H register (DA-H)									?							
CF_{16}	DA-L register (DA-L)									0	0	?	?	?	?	?	?
$D0_{16}$	PWM0 register (PWM0)									?							
$D1_{16}$	PWM1 register (PWM1)									?							
$D2_{16}$	PWM2 register (PWM2)									?							
$D3_{16}$	PWM3 register (PWM3)									?							
$D4_{16}$	PWM4 register (PWM4)									?							
$D5_{16}$	PWM output control register 1 (PW)	PW7	PW6	PW5	PW4	PW3	PW2	PW1	PW0	00_{16}							
$D6_{16}$	PWM output control register 2 (PN)				PN4	PN3	PN2			00_{16}							
$D7_{16}$	I²C data shift register (S0)	D7	D6	D5	D4	D3	D2	D1	D0	?							
$D8_{16}$	I²C address register (S0D)	SAD6	SAD5	SAD4	SAD3	SAD2	SAD1	SAD0	RBW	00_{16}							
$D9_{16}$	I²C status register (S1)	MST	TRX	BB	PIN	AL	AAS	AD0	LRB	0	0	0	1	0	0	0	?
DA_{16}	I²C control register (S1D)	BSEL1	BSEL0	10 BIT SAD	ALS	ES0	BC2	BC1	BC0	00_{16}							
DB_{16}	I²C clock control register (S2)	ACK	ACK BIT	FAST MODE	CCR4	CCR3	CCR2	CCR1	CCR0	00_{16}							
DC_{16}	Serial I/O mode register (SM)		SM6	SM5	0	SM3	SM2	SM1	SM0	00_{16}							
DD_{16}	Serial I/O register (SIO)									?							
DE_{16}		00_{16}								00_{16}							
DF_{16}		00_{16}								00_{16}							

Fig. 2.3.3 Memory map of SFR (special function register) (1)

■SFR Area (addresses $E0_{16}$ to FF_{16})

<Bit allocation>

: } Function bit
Name : }

: No function bit

0 : Fix this bit to "0" (do not write "1")

1 : Fix this bit to "1" (do not write "0")

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately after reset

Address	Register	Bit allocation b7							b0	State immediately after reset b7							b0
$E0_{16}$	Horizontal position register (HR)			HR5	HR4	HR3	HR2	HR1	HR0	00_{16}							
$E1_{16}$	Vertical position register 1 (CV1)		CV16	CV15	CV14	CV13	CV12	CV11	CV10	0	?	?	?	?	?	?	?
$E2_{16}$	Vertical position register 2 (CV2)		CV26	CV25	CV24	CV23	CV22	CV21	CV20	0	?	?	?	?	?	?	?
$E3_{16}$										?							
$E4_{16}$	Character size register (CS)					CS21	CS20	CS11	CS10	0	0	0	0	?	?	?	?
$E5_{16}$	Border selection register (MD)						MD20		MD10	0	0	0	0	0	?	0	?
$E6_{16}$	Color register 0 (CO0)	CO07	CO06	CO05	CO04	CO03	CO02	CO01		00_{16}							
$E7_{16}$	Color register 1 (CO1)	CO17	CO16	CO15	CO14	CO13	CO12	CO11		00_{16}							
$E8_{16}$	Color register 2 (CO2)	CO27	CO26	CO25	CO24	CO23	CO22	CO21		00_{16}							
$E9_{16}$	Color register 3 (CO3)	CO37	CO36	CO35	CO34	CO33	CO32	CO31		00_{16}							
EA_{16}	CRT control register (CC)	CC7					CC2	CC1	CC0	00_{16}							
EB_{16}										?							
EC_{16}	CRT port control register (CRTP)	OP7	OP6	OP5	OUT1	OUT2	R/G/B	VSYC	HSYC	00_{16}							
ED_{16}	CRT clock selection register (CK)	0	0	0	0	0	0	CK1	CK0	00_{16}							
EE_{16}	A-D control register 1 (AD1)				ADM4		ADM2	ADM1	ADM0	0	0	0	?	0	0	0	0
EF_{16}	A-D control register 2 (AD2)			ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	00_{16}							
$F0_{16}$	Timer 1 (TM1)									FF_{16}							
$F1_{16}$	Timer 2 (TM2)									07_{16}							
$F2_{16}$	Timer 3 (TM3)									FF_{16}							
$F3_{16}$	Timer 4 (TM4)									07_{16}							
$F4_{16}$	Timer 12 mode register (T12M)			0	T12M4	T12M3	T12M2	T12M1	T12M0	00_{16}							
$F5_{16}$	Timer 34 mode register (T34M)			T34M5	T34M4	T34M3	T34M2	T34M1	T34M0	00_{16}							
$F6_{16}$	PWM5 register (PWM5)									?							
$F7_{16}$										?							
$F8_{16}$										?							
$F9_{16}$	Interrupt input polarity register (RE)	0		RE5	RE4	RE3	0	0		0	0	0	0	0	0	0	?
FA_{16}		00_{16}								00_{16}							
FB_{16}	CPU mode register (CPUM)	1	1	1	1	1	CM2	0	0	?	?	1	1	1	1	0	0
FC_{16}	Interrupt request register 1 (IREQ1)	IT3R	IICR	VSCR	CRTR	TM4R	TM3R	TM2R	TM1R	00_{16}							
FD_{16}	Interrupt request register 2 (IREQ2)	0			MSR		S1R	1T2R	1T1R	00_{16}							
FE_{16}	Interrupt control register 1 (ICON1)	IT3E	IICE	VSCE	CRTE	TM4E	TM3E	TM2E	TM1E	00_{16}							
FF_{16}	Interrupt control register 2 (ICON2)	0	0	0	MSE	0	S1E	1T2E	1T1E	00_{16}							

Fig. 2.3.4 Memory map of SFR (special function register) (2)

■2 Page Register Area (addresses 217_{16} to $21B_{16}$)

<Bit allocation>

□ : } Function bit
Name :

: No function bit

0 : Fix this bit to "0" (do not write "1")

1 : Fix this bit to "1" (do not write "0")

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately after reset

Address	Register	Bit allocation b7 b6 b5 b4 b3 b2 b1 b0	State immediately after reset b7 b0
217_{16}	ROM correction address 1 (high-order)	ADH17 ADH16 ADH15 ADH14 ADH13 ADH12 ADH11 ADH10	?
218_{16}	ROM correction address 1 (low-order)	ADL17 ADL16 ADL15 ADL14 ADL13 ADL12 ADL11 ADL10	?
219_{16}	ROM correction address 2 (high-order)	ADH27 ADH26 ADH25 ADH24 ADH23 ADH22 ADH21 ADH20	?
$21A_{16}$	ROM correction address 2 (low-order)	ADL27 ADL26 ADL25 ADL24 ADL23 ADL22 ADL21 ADL20	?
$21B_{16}$	ROM correction enable register (RCR)	— — — — 0 0 RCR1 RCR0	00_{16}

Note: Only M37221M8-XXXSP and M37221MA-XXXSP have this area.

Fig. 2.3.5 Memory map of 2 page register (only M37221M8-XXXSP and M37221MA-XXXSP)

2.3 Memory assignment

2.3.1 Internal RAM

The static RAM is assigned.

The internal RAM is used as a stack area for subroutine calls and interrupts as well as for storing data. Both zero page and page 1 are used as a stack area. At reset, the page 1 is specified automatically.

Ordinary, the stack pointer is set to the highest address in the internal RAM of the page 1 during initialization immediately after power on.

This stack pointer moves to lower addresses as the nesting depth increases; therefore, make sure the subroutine nesting and interrupt levels to prevent the stored data destroying necessary data in the RAM. When the stack page is specified "1," if the value of stack pointer exceeds address 0100_{16}, the value of stack page never change to "0" automatically. In this case, set the stack page value to "0" and set the stack pointer value to the highest address by software.

2.3.2 I/O ports (addresses $00C0_{16}$ to $00CD_{16}$)

Addresses $00C0_{16}$ to $00CD_{16}$ are assigned to the ports, port direction registers and the port P3 output mode control register. There are 5 ports: P0, P1, P2, P3 and P5. Ports P0, P1 and P2 are the 8-bit programmable I/O ports. Port P3 consists of 5 bits. The low-order 3 bits ($P3_0$–$P3_2$) are the programmable I/O ports, and the high-order 2 bits ($P3_3$ and $P3_4$) are the input ports.

For I/O ports P0, P1, P2 and $P3_0$–$P3_2$, input or output can be specified in bit units by setting the relevant values to each port direction register.

To specify port bits as output pins, write "1" to the corresponding bit of the port direction register.

Conversely, write "0" to the corresponding bit to specify as an input pin.

For example, to use the even numbered bits of port P2 as output ports and the odd numbered bits as input ports, write "55_{16} (01010101_2)" to address $00C5_{16}$ (the port P2 direction register) at initialization.

Although Port P5 is an output port, it can be specified as the CRT output pins (R, G, B, OUT1) or as general-purpose port ($P5_2$–$P5_5$) by setting each bit in the port P5 direction register.

When setting "0," it is used for the CRT output pins (R, G, B, OUT1), and when setting "1," it is used as general-purpose output ports ($P5_2$–$P5_5$).

Note: Each port direction register default is "input" (port P5 is "CRT output") immediately after reset release.

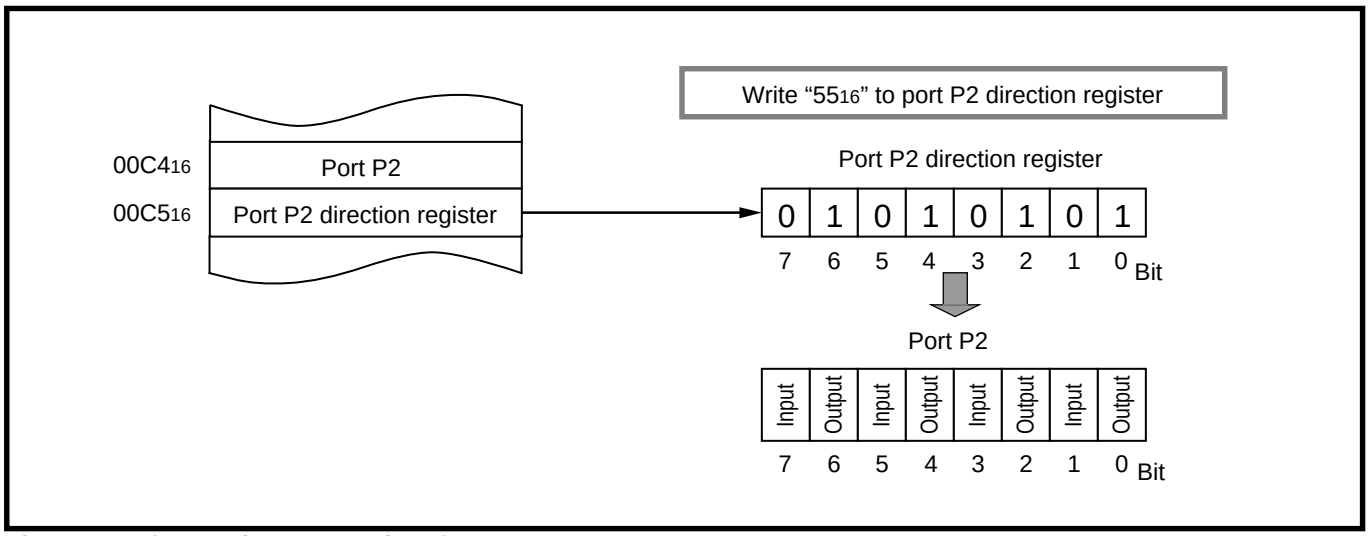


Fig. 2.3.6 I/O setting example of port

2.3.3 DA registers (addresses $00CE_{16}$ and $00CF_{16}$)

The DA-H register is assigned to address $00CE_{16}$, and the DA-L register is assigned to address $00CF_{16}$. Both registers consist of 8 bits.

The DA-H register is used to set the high-order 8 bits of 14-bit PWM output data. The DA-L register is used to set the low-order 6 bits of 14-bit PWM output data (set to bits 0 to 5). Bits 7 is not used.

2.3.4 PWM registers (addresses $00D0_{16}$ to $00D4_{16}$ and $00F6_{16}$)

The PWM0 to PWM4 registers are assigned to addresses $00D0_{16}$ to $00D4_{16}$ and PWM5 register is address $00F6_{16}$. All registers consist of 8 bits.

These registers are used to set the output data corresponding to six 8-bit PWM (PWM0–PWM5).

2.3.5 PWM output control registers (addresses $00D5_{16}$ and $00D6_{16}$)

The PWM output control register 1 is assigned to address $00D5_{16}$ and the PWM output control register 2 is assigned to address $00D6_{16}$.

Both registers consist of 8 bits, and used to select the PWM count source etc. The high-order 3 bits and the low-order 2 bits of the PWM output control register 2 are not used.

2.3.6 Multi-master I^2C-BUS related registers (addresses $00D7_{16}$ to $00DB_{16}$)

The I^2C data shift register, the I^2C address register, the I^2C status register, I^2C control register and the I^2C clock control register are assigned to addresses $00D7_{16}$, $00D8_{16}$, $00D9_{16}$, $00DA_{16}$ and $00DB_{16}$ respectively. All registers consist of 8 bits.

The I^2C data shift register is a 8-bit shift register to store receive data and write transmit data.

The I^2C address register consists of a 7-bit slave address and a $\overline{\text{read}}$/write bit.

The I^2C status register controls the I^2C-BUS interface status. The low-order 4 bits are read-only bits and the high-order 4 bits can be read out and written to.

The I^2C control register controls data communication format.

The I^2C clock control register is used to set ACK control, SCL mode and SCL frequency.

2.3.7 Serial I/O related registers (addresses $00DC_{16}$ and $00DD_{16}$)

The serial I/O mode register is assigned to address $00DC_{16}$ and the serial I/O register is assigned to address $00DD_{16}$. Both registers consist of 8 bits.

The serial I/O mode register is used to select the synchronous clock and the serial I/O port function by its low-order 4 bits. Bit 5 selects the transfer direction, and bit 6 selects the serial data input pin. Bit 4 is set to “0.” Bit 7 is not used.

The serial I/O register is used to write transfer data.

2.3.8 CRT display related registers (addresses $00E0_{16}$ to $00EC_{16}$)

(1) Horizontal position register (address $00E0_{16}$)

The horizontal position register is assigned to address $00E0_{16}$. This register consists of 8 bits, and is used to specify the horizontal position of CRT display. Bits 7 and 6 are not used.

(2) Vertical display position registers (addresses $00E1_{16}$ and $00E2_{16}$)

The vertical display position register 1 is assigned to address $00E1_{16}$ and the vertical display position register 2 is assigned to address $00E2_{16}$. These registers are corresponded to blocks 1 and 2, and used to set the vertical position to start display. Bit 7 of each register is not used.

(3) Character size register (address $00E4_{16}$)

The character size register is assigned to address $00E4_{16}$. This register consists of 8 bits, and is used to specify one of the three sizes of display characters. Bits 4 to 7 are not used.

(4) Border selection register (address $00E5_{16}$)

The border selection register is assigned to address $00E5_{16}$. This register consists of 8 bits, and is used to set the border for blocks 1 and 2 by using one bit each. Bits 1 and 3 to 7 are not used.

(5) Color registers (addresses $00E6_{16}$ to $00E9_{16}$)

Color registers 0 to 3 are assigned to addresses $00E6_{16}$ to $00E9_{16}$. All color registers consist of 8 bits, and are used to set character output, blank output and character background color by CRT output (R, G, B, OUT1). Bit 0 is not used.

(6) CRT control register (address $00EA_{16}$)

The CRT control register is assigned to address $00EA_{16}$. This register consists of 8 bits, and is used to set display on/off for each block. Bits 3 to 6 are not used.

(7) CRT port control register (address $00EC_{16}$)

The CRT port control register is assigned to address $00EC_{16}$. This register consists of 8 bits, and is used to set the input polarity (H_{SYNC} and V_{SYNC}) and the output polarity (R, G, B, OUT1 and OUT2).

2.3.9 A-D control registers (addresses $00EE_{16}$ and $00EF_{16}$)

The A-D control register 1 is assigned to address $00EE_{16}$, the A-D control register 2 is assigned to address $00EF_{16}$. Both registers consist of 8 bits

The A-D control register 1 is used to select analog input pins and hold the results of comparator operation. Bits 3 and 5 to 7 are not used.

The A-D control register 2 is used to set the internal analog voltage. Bits 6 and 7 are not used.

2.3.10 Timer registers (addresses $00F0_{16}$ to $00F3_{16}$)

The timer registers are assigned to addresses $00F0_{16}$ to $00F3_{16}$. Both the timer and timer latch are written in this area when writing, but only the timer is read when reading.

To write data to address $00F1_{16}$, for example, the data are stored to the timer 2 latch and timer 2. After that, the timer 2 contents are decremented by synchronizing with the clock pulse but the timer 2 latch contents are not changed. Accordingly, when reading data at address $00F1_{16}$, the contents of timer 2 is read out at the time.

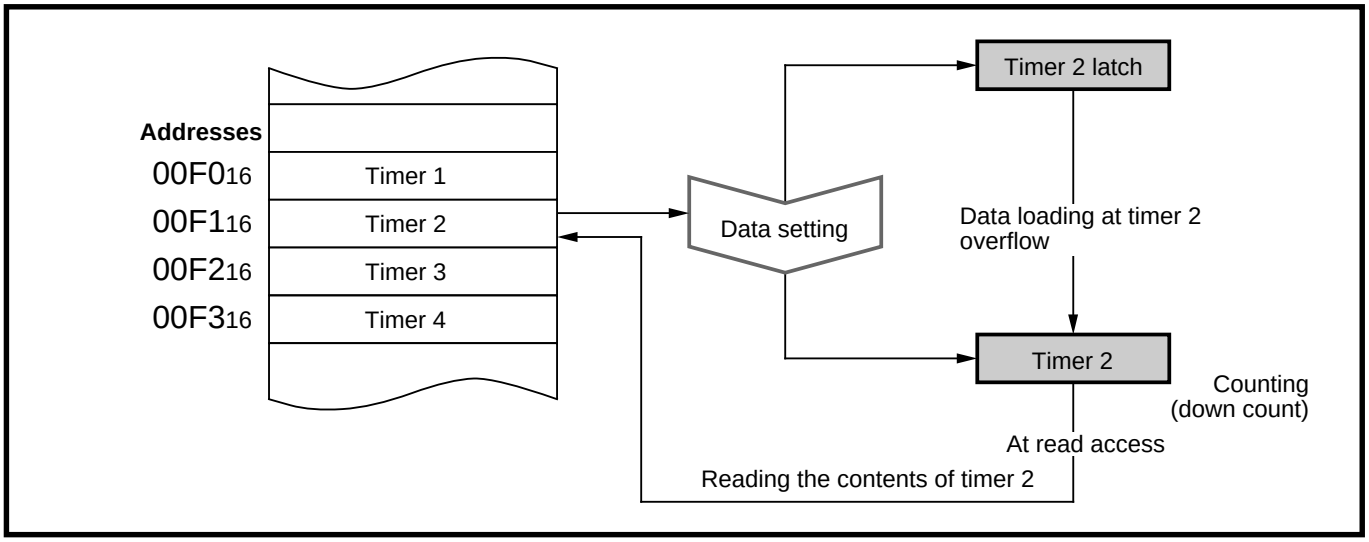


Fig. 2.3.7 Access to timer registers

2.3.11 Timer mode registers (address $00F4_{16}$ and $00F5_{16}$)

The timer 12 mode register is assigned to address $00F4_{16}$ and the timer 34 mode register is assigned to address $00F5_{16}$. Both registers consist of 8 bits. They select the count source of timer and control the count stop bit. Bits 5 to 7 of the timer 12 mode register and bits 6, 7 of the timer 34 mode register are not used.

2.3.12 CPU mode register (address $00FB_{16}$)

The CPU mode register is assigned to address $00FB_{16}$. This register consists of 8 bits, and specifies the stack page. Set bits 0 and 1 are set to "0," and set bits 3 to 7 to "0."

2.3.13 Interrupt request registers (addresses $00FC_{16}$ and $00FD_{16}$)

The interrupt request register 1 is assigned to address 00FC16 and the interrupt request register 2 is assigned to address $00FD_{16}$. Both registers consist of 8 bits, and hold content of each interrupt request bit. Bits 3 and 5 to 7 of the interrupt request register 2 are not used.

2.3.14 Interrupt control registers (addresses $00FE_{16}$ and $00FF_{16}$)

The interrupt control register 1 is assigned to address $00FE_{16}$ and the interrupt control register 2 is assigned to address $00FF_{16}$. Both registers consist of 8 bits, and sets enable/disable of interrupts.
Bits 7 to 5 and 3 of the interrupt control register 2 are not used.

2.3.15 2 page register (addresses 0217_{16} to $021B_{16}$) (only M37221M8-XXXSP and M37221MA-XXXSP)

(1) ROM correction addresses (address 0217_{16} to $021A_{16}$)
Addresses 0217_{16} to $021A_{16}$ are assigned to ROM correction address. The ROM data addresses to be corrected are set to the ROM correction addresses.

(2) ROM correction enable register (address $021B_{16}$)
The ROM correction enable register is assigned to address $021B_{16}$. This register consist of 8 bits, and controls the ROM correction function. Bits 2 to 7 are not used.

2.3.16 CRT display RAM (addresses 0600_{16} to $06B7_{16}$)

The display RAM is used to specify the character to be displayed on the CRT and its color. Two addresses are used for one character: one address (8 bits) to specify each character code and the other (8 bits) to specify the color of the character.

2.3.17 ROM (addresses $A000_{16}$ to $FFFF_{16}$)

The mask ROM is assigned.
In this internal ROM, addresses $FFDE_{16}$, $FFDF_{16}$, $FFE4_{16}$, $FFF5_{16}$, and $FFF8_{16}$ to $FFFF_{16}$ are assigned to vector area for reset and for interrupts. A vector jump destination storage address (16 bits) are stored in 2 addresses by the 1 interrupt source.

2.3.18 CRT display ROM (addresses 10000_{16} to $11FFF_{16}$)

The display ROM stores (masks) character patterns of each character to be displayed on the CRT. Although one character consists of 16 (vertical) $\times$ 12 (horizontal) dots, it is divided into a 16 $\times$ 8 dot and a 16 $\times$ 4 dot pattern, with each pattern stored in one address. In other words, two addresses (16 bits) are used for one character. The ROM can store up to 256 kinds of characters.

2.4 Input/Output pins

The M37221M6-XXXSP/FP has 33 programable ports (I/O ports, input ports, output ports). The double-function ports function as ports and as pins for internal peripheral devices.

M37220M3-XXXSP/FP

Refer to **"CHAPTER 4. M37220M3-XXXSP/FP."**

●Double-function ports ..... <I/O ports>
$P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_4$, $P3_0$, $P3_1$
<Input ports>
$P3_3$, $P3_4$
<Output ports>
$P5_2$–$P5_5$

●I/O port-only ports ..... <I/O ports>
$P2_5$–$P2_7$, $P3_2$

And also, the M37221M6-XXXSP/FP has 9 pins with only the dedicated function.

●Dediated pins ..... V_{CC}, V_{SS}, $\overline{RESET}$, X_{IN}, CNV_{SS}, X_{OUT}, D-A, H_{SYNC}, V_{SYNC}

2.4.1 Programmable ports

(1) Port P0

Port P0 is an 8-bit input/output port. This is an N-channel open drain output. Port P0 is assigned to memory at address $00C0_{16}$ on zero page.

Port P0 has the direction register (at address $00C1_{16}$ on zero page), so that it is possible to program each bit whether the port is used for input or output. The pins of which the direction register is programmed to "0" are set for input; when programmed to "1", the pins are set for output.

When pins are programmed as output pins, the output data are written into the port latch and then output. When reading data from the output pins, the output pin level is not read but the port latch data is read. This allows a previously-output value to be read correctly even if the output LOW voltage has risen, for example, because a light emitting diode was directly driven.

The input pins float, so the values of the pins can be read. When writing data into the input pin, it is written only into the port latch, while the pin remains floating.

Ports $P0_0$–$P0_5$ are also used as PWM output pins PWM0–PWM5 respectively. Port $P0_6$ is also used as external interrupt pin INT2 and analog input pin A-D4. The $P0_7$ pin is also used as external interrupt input pin INT1. When external interrupts INT1 and INT2 are enabled, an interrupt is processed according to transition in the level on these pins.

Ports $P0_6$ and $P0_7$ have the schmit characteristics when they are used as INT input pins. In this case, set these pins for input by the port P0 direction register.

(2) Port P1

Port P1 is an 8-bit I/O port. The output structure is CMOS output, however, only when ports $P1_1$–$P1_4$ are used as multi-master I^2C-BUS interface, the output structure is N-channel open-drain output. Port P1 has basically the same function as port P0.

Port $P1_0$ is also used as CRT output pin OUT2. Pin OUT2 is a CRT output pin. When setting "1" to bit 7 of the CRT control register, the pin functions as CRT output pin, when setting "0," the pin functions as a general-purpose I/O port.

Ports $P1_1$–$P1_4$ are used as SCL1, SCL2, SDA1 and SDA2 respectively.

Port $P1_5$ is also used as external interrupt input pin INT3 and analog input pin A-D1.

Ports $P1_6$ and $P1_7$ are also used as analog input pins A-D2 and A-D3 respectively.

(3) Port P2

Port P2 is an 8-bit I/O port. The output structure is CMOS output, however, only when ports $P2_0$ and $P2_1$ are used as serial I/O pins, the output structure is N-channel open-drain output. Port P2 has basically the same function as port P0.

Port $P2_0$ is also used as serial I/O synchronous clock input/output pin S_{CLK}. Port $P2_1$ is also used as serial I/O data output pin S_{OUT}. Port $P2_2$ is also used as serial I/O data input pin S_{IN}.

Port $P2_3$ is also used as external clock input pin TIM3. When the timer 3 count source is supplied form an external device (as set by the timer 34 mode register), the input signal to this pin is the timer 3 count source.

The port $P2_4$ is also used as external clock input pin TIM2. When the count source for timer 2 is supplied form an external device (as set by the timer 12 mode register), the input signal to this pin is the timer 2 count source.

Ports $P2_5$–$P2_7$ has only I/O port function.

(4) Port P3

Ports $P3_0$–$P3_2$ are 3-bit I/O ports, ports $P3_3$ and $P3_4$ are a 2-bit input port. For the output structure of ports $P3_0$ and $P3_1$, either CMOS output or N-channel open-drain output structure can be selected by bit 0 or 1 of the port P3 output mode control register (address $00CD_{16}$). When "1," N-channel open-drain output structure is selected; when "0," CMOS output structure is selected.

Port $P3_2$ has only I/O port function. The output structure is N-channel open-drain output. Port $P3_2$ has basically the same function as port P0.

Ports $P3_0$ and $P3_1$ are also used as analog input pins A-D5 and A-D6 respectively.

Ports $P3_3$ and $P3_4$ are also used as CRT display clock input pins OSC1 and OSC2 respectively. Pin OSC1 is a clock input for CRT display, pin OSC2 is a clock output for CRT display. The output structure of pin OSC2 is CMOS output.

(5) Port P5

Ports $P5_2$–$P5_5$ are 4-bit output ports. The output structure is CMOS output. Ports $P5_2$–$P5_5$ are also used as CRT output pins R, G, B, OUT1 respectively.

Pins R, G, B, and OUT1 are CRT output pins. When setting each bit of the port P5 direction register to "0," the pins function as CRT output pins; when setting to "1," the pins function as general-purpose output ports. The output structure of CRT output pin is CMOS output structure.

2.4 Input/Output pins

Table 2.4.1 List of programmable port functions

Ports	Functions except port	Name
$P0_0$–$P0_5$	PWM0–PWM5	PWM output pin
$P0_6$	INT2/A-D4	External interrupt input pin/Analog input pin
$P0_7$	INT1	External interrupt input pin
$P1_0$	OUT2	CRT output pin
$P1_1$	SCL1	Multi-master I^2C-BUS interface pin
$P1_2$	SCL2	Multi-master I^2C-BUS interface pin
$P1_3$	SDA1	Multi-master I^2C-BUS interface pin
$P1_4$	SDA2	Multi-master I^2C-BUS interface pin
$P1_5$	A-D1/INT3	Analog input pin/External interrupt pin
$P1_6$	A-D2	Analog input pin
$P1_7$	A-D3	Analog input pin
$P2_0$	S_{CLK}	Serial I/O synchronous clock input/output pin
$P2_1$	S_{OUT}	Serial I/O data input /output pin
$P2_2$	S_{IN}	Serial I/O data input pin
$P2_3$	TIM3	External clock input pin
$P2_4$	TIM2	External clock input pin
$P2_5$–$P2_7$	—	Function as only programmable I/O ports
$P3_0$	A-D5	Analog input pin
$P3_1$	A-D6	Analog input pin
$P3_2$	—	Functions as only programmable I/O port.
$P3_3$	OSC1	CRT display clock input pin
$P3_4$	OSC2	CRT display clock output pin
$P5_2$	R	CRT output pin
$P5_3$	G	CRT output pin
$P5_4$	B	CRT output pin
$P5_5$	OUT1	CRT output pin

2.4.2 Dedicated pins

(1) 14-bit PWM output (D-A) pin

This is a 14-bit PWM signal output pin. This pin also can be used for 1-bit general-purpose output port. The output structure is CMOS output.

(2) Vertical and horizontal synchronous signal input pins (V_{SYNC}, H_{SYNC})

These pins input the vertical and horizontal synchronous signals for CRT display.

(3) Test input pin (CNV_{SS})

Connect this pin to V_{SS}.

(4) Reset input pin ($\overline{RESET}$)

This pin inputs reset signal. To reset the microcomputer, hold the $\overline{RESET}$ pin at a LOW level for 2 μs or more. Reset is released when HIGH level is applied to the $\overline{RESET}$ pin. For details, refer to **"2.15 Reset."**

(5) Clock I/O pins (X_{IN}, X_{OUT})

These pins are I/O pins of main clock f(X_{IN}). Since a microcomputer has on-chip clock oscillation circuit, set the oscillation frequency by connecting an external ceramic resonator or a quartz-crystal oscillator between pins X_{IN} and X_{OUT}.

When inputting an external clock, connect the external clock to the X_{IN} pin and leave the X_{OUT} pin open.

The output structure of X_{OUT} pin is CMOS output.

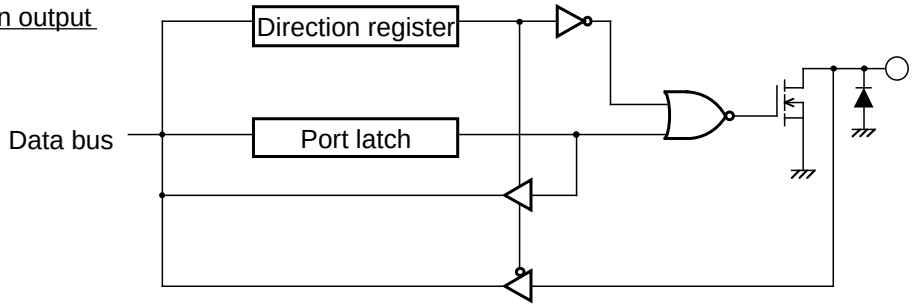
(6) Power source input pin (V_{CC}, V_{SS})

These pins supply the power source to a microcomputer. Apply voltage of 5 V ± 10 % to pin V_{CC} and 0 V to pin V_{SS}.

2.4 Input/Output pins

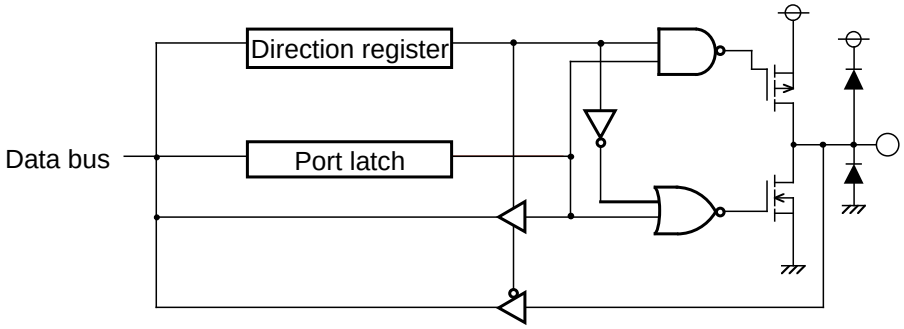
P0$_0$/PWM0–P0$_5$/PWM5, P3$_2$

N-channel open-drain output



P1$_0$/OUT2, P1$_1$/SCL1, P1$_2$/SCL2, P1$_3$/SDA1, P1$_4$/SDA2, P1$_5$/A-D1/INT3, P1$_6$/A-D2, P1$_7$/A-D3, P2$_0$/SCLK, P2$_1$/SOUT, P2$_2$/SIN, P2$_3$/TIM3, P2$_4$/TIM2, P2$_5$–P2$_7$, P3$_0$/A-D5, P3$_1$/A-D6

CMOS output

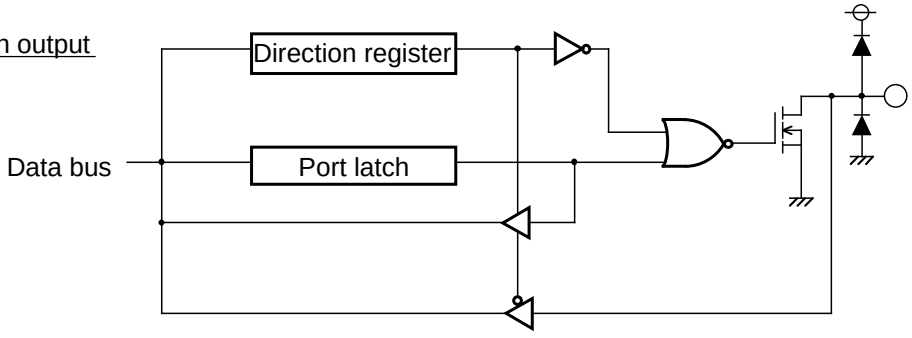


Notes 1 : When ports P1$_1$–P1$_4$ are used as multi-master I^{2}C-BUS interface pin and when ports P2$_0$, P2$_1$ are used as serial I/O output pins, their output structure is N-channel open-drain output.

2 : For the output structure of ports P3$_0$, P3$_1$, either CMOS output or N-channel open-drain output is selected (In the case of N-channel open-drain output, the block diagram is the same as below).

P0$_6$/INT2/A-D4, P0$_7$/INT1

N-channel open-drain output

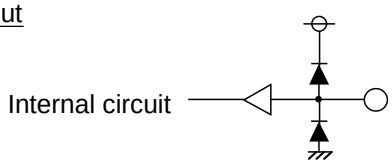


○ indicates a pin.

Fig. 2.4.1 I/O pin block diagram (1)

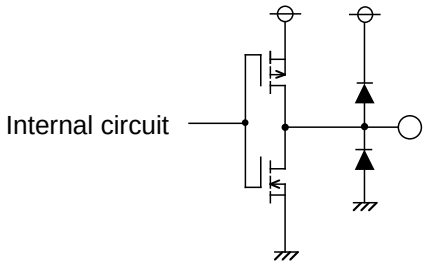
P3$_3$/OSC1, P3$_4$

Input



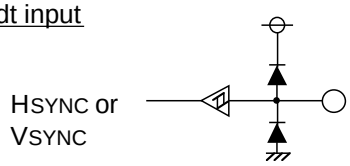
D-A, P5$_2$/R, P5$_3$/G, P5$_4$/B, P5$_5$/OUT1

CMOS output



HSYNC, VSYNC

Schmidt input



○ indicates a pin.

Fig. 2.4.2 I/O pin block diagram (2)

2.5 Interrupts

Interrupts are used in the following cases.

- When there is a request to execute a higher priority routine than current processing routine.
- When it is necessary to process according to a certain timing.

The M37221M6-XXXSP/FP has 14 interrupt sources (including reset).
These are vector interrupts with a fixed priority sequence. Table 2.5.1 shows the interrupt sources, vector addresses and the interrupt priority sequence.

M37220M3-XXXSP/FP
Refer to **"CHAPTER 4. M37220M3-XXXSP/FP."**

Table 2.5.1 Interrupt sources, vector addresses and priority

Priority	Interrupt sources	Vector addresses		Remarks
		High-order byte	Low-order byte	
1	Reset (Note)	$FFFF_{16}$	$FFFE_{16}$	Non-maskable
2	CRT interrupt	$FFFD_{16}$	$FFFC_{16}$	
3	INT2 interrupt	$FFFB_{16}$	$FFFA_{16}$	Active edge selectable
4	INT1 interrupt	$FFF9_{16}$	$FFF8_{16}$	Active edge selectable
5	Timer 4 interrupt	$FFF5_{16}$	$FFF4_{16}$	
6	$f(X_{IN})/4096$ interrupt	$FFF3_{16}$	$FFF2_{16}$	
7	V_{SYNC} interrupt	$FFF1_{16}$	$FFF0_{16}$	Active edge selectable
8	Timer 3 interrupt	$FFEF_{16}$	$FFEE_{16}$	
9	Timer 2 interrupt	$FFED_{16}$	$FFEC_{16}$	
10	Timer 1 interrupt	$FFEB_{16}$	$FFEA_{16}$	
11	Serial I/O interrupt	$FFE9_{16}$	$FFE8_{16}$	
12	Multi-master I^2C-BUS interface interrupt	$FFE7_{16}$	$FFE6_{16}$	
13	INT3 interrupt	$FFE5_{16}$	$FFE4_{16}$	Active edge selectable
14	BRK instruction interrupt	$FFDF_{16}$	$FFDE_{16}$	Non-maskable (software interrupt)

Note: Reset are included in the table because it operates in the same way as interrupts.

These 14-source, 14-vector interrupts have the priority sequence as shown in Table 2.5.1 (reset has a higher priority than interrupts).
When two or more interrupt requests occur at the same sampling point, the interrupt with the higher priority (in order of 1 to 14) is received. This priority sequence is determined by hardware, but priority processing is possible to be varied by software, by using the interrupt enable bit and the interrupt disable flag.

2.5.1 Interrupt sources

The following explains interrupt sources, in order of priority (except reset).

(1) CRT interrupt

When displaying a character block with the CRT display function, the CRT interrupt request occurs at the completion of the display.

(2) INT2 interrupt

An INT2 interrupt request is generated by detecting a level transition on pin INT2 (external interrupt input).

Detecting either positive polarity (LOW to HIGH transition) or negative polarity (HIGH to LOW transition) is set with RE4 (the interrupt input polarity register: bit 4 at address $00F9_{16}$). When RE4 is set to "0," a positive polarity is detected; when RE4 is set to "1," a negative polarity is detected.

The INT2 pin is also used for port $P0_6$ and pin A-D4. An INT2 interrupt by a level transition on the pin may cause software runaway. Therefore, when this pin is used as port $P0_6$, disable an INT2 interrupt by using an interrupt enable bit and the interrupt disable flag (I).

(3) INT1 interrupt

An INT1 interrupt request is generated by detecting a level transition on pin INT1 (external interrupt input).

Detecting either positive polarity (LOW to HIGH transition) or negative polarity (HIGH to LOW transition) to be detected is set with RE3 (the interrupt input polarity register: bit 3 at address $00F9_{16}$). When RE3 is set to "0," a positive polarity is detected; when RE3 is set to "1," a negative polarity is detected.

Pin INT1 is also used for port $P0_7$. An INT1 interrupt by a level transition on the pin may cause software runaway. Therefore, when this pin is used as port $P0_7$, disable the INT1 interrupt by using an interrupt enable bit and interrupt disable flag (I).

(4) Timer 4 interrupt

Timer 4 value is counted down. Timer 4 interrupt request occurs when the count source next to "00_{16}" is input.

(5) $f(X_{IN})/4096$ interrupt

A $f(X_{IN})/4096$ interrupt request occurs for a $f(X_{IN})/4096$ period.

This interrupt is valid when the PWM count source is supplied (when bit 0 of PWM output control register 1 is "0").

(6) V_{SYNC} interrupt

A V_{SYNC} interrupt request occurs synchronized with the vertical synchronous signal which is input to pin V_{SYNC}.

When the V_{SYNC} input polarity is positive (the CRT port control register: bit 1 at address $00EC_{16}$ is "0"), an interrupt request is generated by a rising edge (LOW to HIGH transition) of the V_{SYNC} input; conversely, when the polarity is negative, an interrupt request is generated by a falling edge.

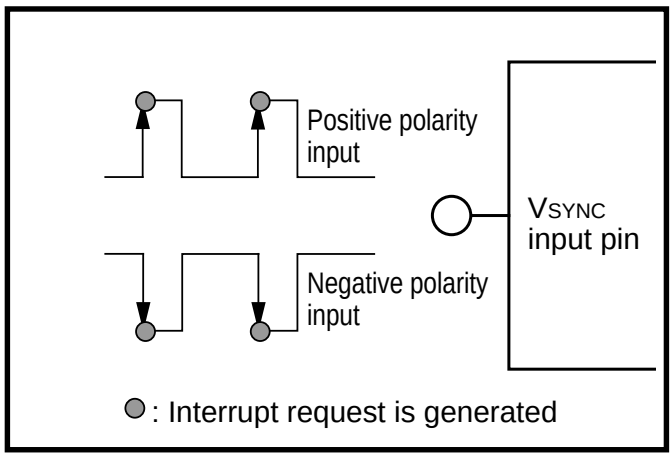


Fig. 2.5.1 V_{SYNC} interrupt generation timing

2.5 Interrupts

(7) Timer 3 interrupt

Timer 3 value is counted down. Timer 3 interrupt request occurs when the count source next to "00_{16}" is input.

(8) Timer 2 interrupt

Timer 2 value is counted down. Timer 2 interrupt request occurs when a count source next to "00_{16}" is input

(9) Timer 1 interrupt

Timer 1 value is counted down. Timer 1 interrupt request occurs when a count source next to "00_{16}" is input.

(10) Serial I/O interrupt

The serial I/O interrupt request is generated by detecting a rising edge of the eighth serial transfer clock after writing to the serial I/O register.

(11) Multi-master I^2C-BUS interface interrupt

A multi-master interrupt request occurs synchronized with a falling edge serial clock (SCL) every completion of 1-byte data communication.

(12) INT3 interrupt

An INT3 interrupt request is generated by detecting a transition in the level on pin INT3 (external interrupt input).

Detecting either positive polarity (LOW to HIGH transition) or negative polarity (HIGH to LOW transition) to be detected is set with RE5 (the interrupt input polarity register: bit 5 at address $00F9_{16}$). When RE5 is set to "0," a positive polarity is detected, when RE5 is set to "1," a negative polarity is detected.

Pin INT3 is also used for port $P1_5$ and pin A-D1. An INT3 interrupt by a level transition on the pin may cause software runaway. Therefore, when this pin is used as port $P1_5$, disable an INT3 interrupt by using an interrupt enable bit and interrupt disable flag (1).

(13) BRK instruction interrupt

This software interrupt has the least significant priority and generates an interrupt request is generated by executing when the **BRK** instruction. There is no corresponding interrupt enable bit and no influence by the interrupt disable flag (I).

2.5.2 Interrupt control

Each interrupt can be controlled with the interrupt request bit, the interrupt control bit, and the interrupt disable flag.

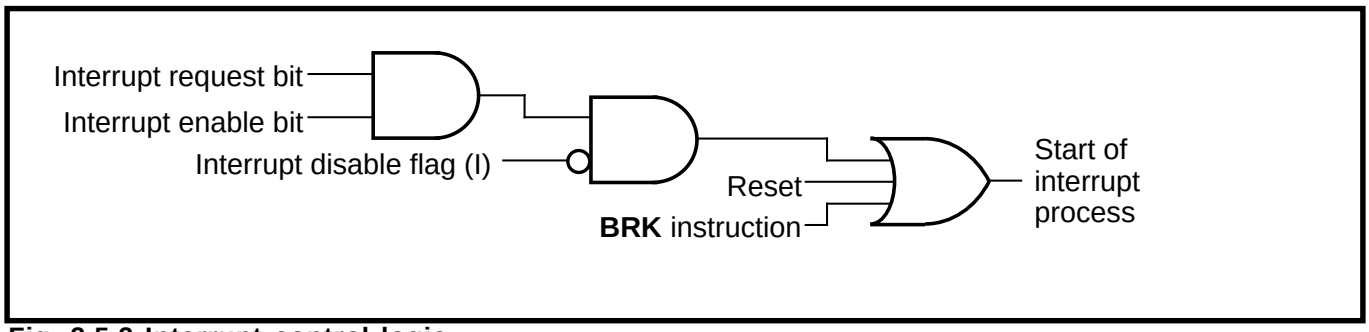


Fig. 2.5.2 Interrupt control logic

(1) Interrupt request bit

When an interrupt request occurs, the corresponding bit of the interrupt request register is set to "1." The interrupt request is held active until an interrupt is accepted or "0" is written to the relevant bit by software. The bit is automatically cleared to "0" simultaneously when the interrupt is accepted. Interrupt request bits are cleared to "0" (to clear the interrupt request) by software but are not set to "1" (to generate the interrupt request) by software.

Each interrupt request bit is assigned to interrupt request registers 1 and 2 (addresses $00FC_{16}$ and $00FD_{16}$).

(2) Interrupt enable bit

Interrupt enable bits control the acceptance of each interrupt.

When the interrupt enable bit is cleared to "0" (to disable an interrupt), the interrupt cannot be accepted. Conversely, when the interrupt enable bit is set to "1" (to enable an interrupt), the interrupt is accepted. However, if the interrupt disable flag is set to "1," the interrupt cannot be accepted even when the interrupt enable bit is set to "1."

Each interrupt enable bit is assigned to interrupt control registers 1 and 2 (addresses $00FE_{16}$ and $00FF_{16}$).

(3) Interrupt disable flag (I)

The interrupt disable flag (I) is assigned to bit 2 of the processor status register. When the interrupt disable flag is set to "1," all interrupts except the **BRK** instruction interrupt are disabled; when the flag is cleared to "0," interrupts are enabled. However, if the interrupt disable flag is cleared to "0," the interrupt cannot be accepted even when the interrupt enable bit is "0."

2.5 Interrupts

Interrupt Request Register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt request register 1 (IREQ1) [Address 00FC16]

B	Name	Functions	After reset	R	W
0	Timer 1 interrupt request bit (TM1R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
1	Timer 2 interrupt request bit (TM2R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
2	Timer 3 interrupt request bit (TM3R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
3	Timer 4 interrupt request bit (TM4R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
4	CRT interrupt request bit (CRTR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
5	VSYNC interrupt request bit (VSCR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
6	Multi-master I2C-BUS interface interrupt request bit (IICR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
7	INT3 interrupt request bit (IT3R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻

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

Fig. 2.5.3 Interrupt request register 1 (address 00FC16)

Interrupt Request Register 2

b7 b6 b5 b4 b3 b2 b1 b0

0

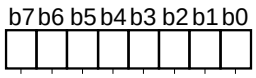
Interrupt request register 2 (IREQ2) [Address 00FD16]

B	Name	Functions	After reset	R	W
0	INT1 interrupt request bit (ITIR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
1	INT2 interrupt request bit (IT2R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
2	Serial I/O interrupt request bit (S1R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
3	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
4	f(XIN)/4096 interrupt request bit (MSR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
5, 6	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—
7	Fix this bit to "0."		0	R	W

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

Fig. 2.5.4 Interrupt request register 2 (address 00FD16)

Interrupt Control Register 1

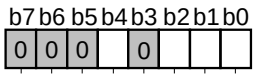


Interrupt control register 1 (ICON1) [Address $00FE_{16}$]

B	Name	Functions	After reset	R	W
0	Timer 1 interrupt enable bit (TM1E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
1	Timer 2 interrupt enable bit (TM2E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
2	Timer 3 interrupt enable bit (TM3E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
3	Timer 4 interrupt enable bit (TM4E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
4	CRT interrupt enable bit (CRTE)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
5	V_{SYNC} interrupt enable bit (VSCE)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
6	Multi-master I^2C-BUS interface interrupt enable bit (IICE)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
7	INT3 interrupt enable bit (IT3E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W

Fig. 2.5.5 Interrupt control register 1 (address $00FE_{16}$)

Interrupt Control Register 2



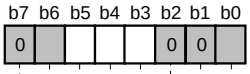
Interrupt control register 2 (ICON2) [Address $00FF_{16}$]

B	Name	Functions	After reset	R	W
0	INT1 interrupt enable bit (IT1E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
1	INT2 interrupt enable bit (IT2E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
2	Serial I/O interrupt enable bit (S1E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
3	Fix this bit to "0."		0	R	W
4	f(X_{IN})/4096 interrupt enable bit (MSE)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
5 to 7	Fix these bits to "0."		0	R	W

Fig. 2.5.6 Interrupt control register 2 (address $00FF_{16}$)

2.5 Interrupts

Interrupt Input Polarity Register

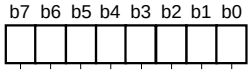


Interrupt input polarity register(RE) [Address 00F9$_{16}$]

b	Name	Functions	After reset	R	W
0	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		Indeterminate	R	—
1, 2	Fix these bits to "0."		0	R	W
3	INT1 polarity switch bit (RE3)	0 : Positive polarity 1 : Negative polarity	0	R	W
4	INT2 polarity switch bit (RE4)	0 : Positive polarity 1 : Negative polarity	0	R	W
5	INT3 polarity switch bit (RE5)	0 : Positive polarity 1 : Negative polarity	0	R	W
6	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
7	Fix this bit to "0."		0	R	W

Fig. 2.5.7 Interrupt input polatiry register (address 00F9$_{16}$)

CRT port control register

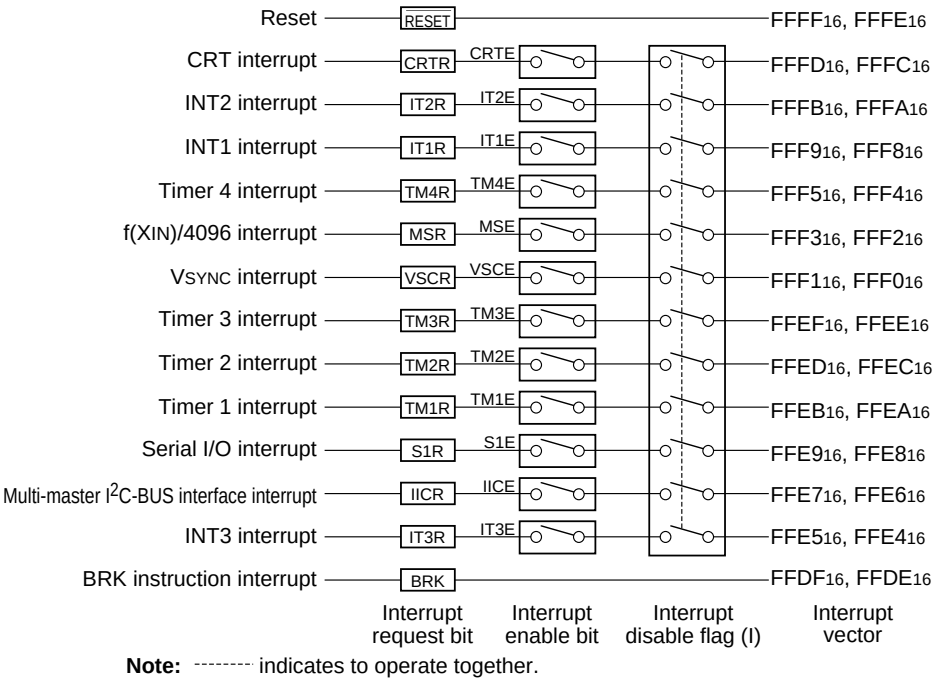


CRT port control register(CRTP) [Address 00EC$_{16}$]

b	Name	Functions	After reset	R	W
0	HSYNC input polarity switch bit (HSYC)	0 : Positive polarity 1 : Negative polarity	0	R	W
1	VSYNC input polarity switch bit (VSYC)	0 : Positive polarity 1 : Negative polarity	0	R	W
2	R, G, B output polarity switch bit (R, G, B)	0 : Positive polarity 1 : Negative polarity	0	R	W
3	OUT2 output polarity switch bit (OUT2)	0 : Positive polarity 1 : Negative polarity	0	R	W
4	OUT1 output polarity switch bit (OUT1)	0 : Positive polarity 1 : Negative polarity	0	R	W
5	R signal output switch bit (OP5)	0 : R signal output 1 : MUTE signal output	0	R	W
6	G signal output switch bit (OP6)	0 : G signal output 1 : MUTE signal output	0	R	W
7	B signal output switch bit (OP7)	0 : B signal output 1 : MUTE signal output	0	R	W

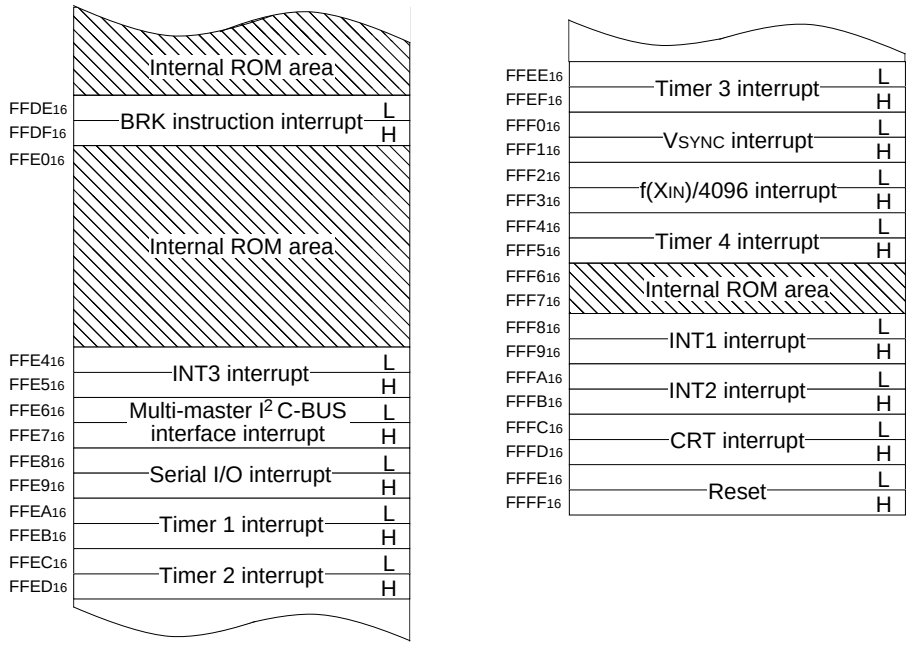
Fig. 2.5.8 CRT port control register (address 00EC$_{16}$)

Interrupt request bits are set to “1” by occurrence of an interrupt request, even if the interrupt is disabled. Therefore, to disable interrupt processing, clear the interrupt request bit to “0” immediately before the interrupt disable state is cancelled (interrupt enable state, i.e., the interrupt enable bit = “1” and the interrupt disable flag = “0”).



Note: --------- indicates to operate together.

Fig. 2.5.9 Interrupt control system



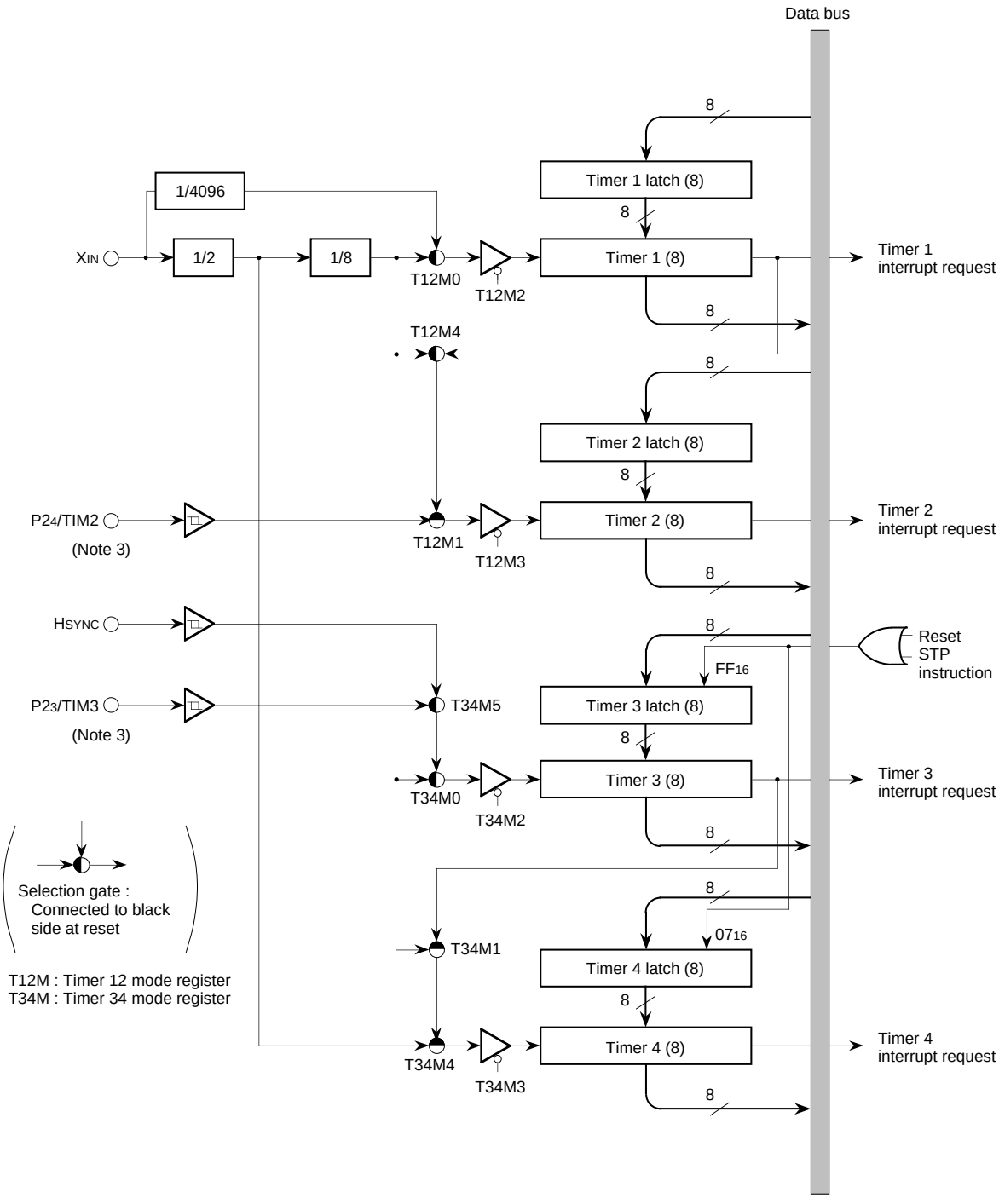
Interrupt Vector Table

The low-order 8 bits and the high-order 8 bits of jump destination address when an interrupt occurs, are stored to addresses “L” and “H” respectively.

Fig. 2.5.10 Interrupt vector table

2.6 Timers

M37221M6-XXXSP/FP has four 8-bit timers with reload latch. Figure 2.6.1 shows the timer block diagram.



Notes 1: HIGH pulse width of external clock inputs TIM2 and TIM3 needs 4 machine cycles or more.
2: When the external clock source is selected, timers 2 and 3 are counted at a rising edge of input signal.
3: In the stop mode or the wait mode, external clock inputs TIM2 and TIM3 cannot be used.

Fig. 2.6.1 Timer 1, timer 2, timer 3, and timer 4 block diagram

2.6.1 Timer functions

There are four timers; Timer 1, Timer 2, Timer 3, Timer 4 and each timer has an 8-bit reload latch. All timers are the count-down type, and when the timer latch value is “n”, the divide ratio is 1/(n+1)(“n” = 0 to 255). When the valve “n” is written to reload latch, is also set “n” to its timer, simultaneously.

Timer value is counted down each rising edge of count source. The timer overflows at the count next pulse, after the count value reaches “00_{16},” and the interrupt request occurs. At the same time of timer overflow, the reload latch value “n” is set (reload) to timer, and timer continues to count down. The divide ratio is 1/(n+1). Make sure that set “n” in the range “00_{16}” to “FF_{16}.”

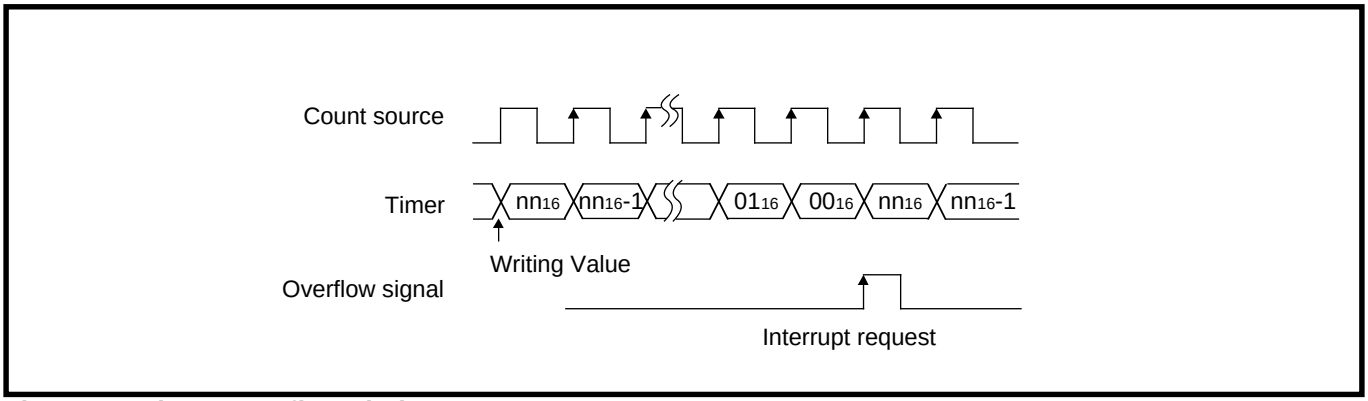


Fig. 2.6.2 Timer overflow timing

(1) Timer 1

Timer 1 can select one of the following count sources:

- f(X_{IN})/16
- f(X_{IN})/4096

(This is a clock by f(X_{IN})/4096 interrupt and is valid only when PWM count source is supplied.)

The count source of timer 1 is selected by setting bit 0 of the timer 12 mode register (address $00F4_{16}$).

Timer 1 interrupt request occurs at timer 1 overflow.

(2) Timer 2

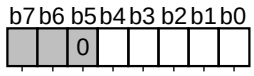
Timer 2 can select one of the following count sources:

- f(X_{IN})/16
- Timer 1 overflow signal
- External clock from pin $P2_4$/TIM2

The count source of timer 2 is selected by setting bits 4 and 1 of the timer 12 mode register (address $00F4_{16}$). When timer 1 overflow signal is a count source for timer 2, timer 1 functions as an 8-bit prescaler.

Timer 2 interrupt request occurs at timer 2 overflow.

Timer 12 Mode Register



Timer 12 mode register (T12M) [Address $00F4_{16}$]

B	Name	Functions	After reset	R	W
0	Timer 1 count source selection bit (T12M0)	0: $f(X_{IN})/16$ 1: $f(X_{IN})/4096$	0	R	W
1	Timer 2 count source selection bit (T12M1)	0: Internal clock 1: External clock from $P2_4$/TIM2 pin	0	R	W
2	Timer 1 count stop bit (T12M2)	0: Count start 1: Count stop	0	R	W
3	Timer 2 count stop bit (T12M3)	0: Count start 1: Count stop	0	R	W
4	Timer 2 internal count source selection bit (T12M4)	0: $f(X_{IN})/16$ 1: Timer 1 overflow	0	R	W
5	Fix this bit to "0."		0	R	W
6,7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 2.6.3 Timer 12 mode register (address $00F4_{16}$)

(3) Timer 3

Timer 3 can select one of the following count sources:

- $f(X_{IN})/16$
- External clock from pin H_{SYNC}
- External clock from pin $P2_3$/TIM3

The count source of timer 3 is selected by setting bits 5 and 0 of the timer 34 mode register (address $00F5_{16}$)

Timer 3 interrupt request occurs at timer 3 overflow.

(4) Timer 4

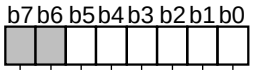
Timer 4 can select one of the following count sources:

- $f(X_{IN})/16$
- $f(X_{IN})/2$
- Timer 3 overflow signal

The count source of timer 3 is selected by setting bits 4 and 1 of the timer 34 mode register (address $00F5_{16}$). When timer 3 overflow signal is a count source for timer 4, timer 3 functions as an 8-bit prescaler.

Timer 4 interrupt request occurs at timer 4 overflow.

Timer 34 Mode Register



Timer 34 mode register (T34M) [Address $00F5_{16}$]

B	Name	Functions	After reset	R	W
0	Timer 3 count source selection bit (T34M0)	0: f(XIN)/16 1: External clock	0	R	W
1	Timer 4 internal count source selection bit (T34M1)	0: Timer 3 overflow 1: f(XIN)/16	0	R	W
2	Timer 3 count stop bit (T34M2)	0: Count start 1: Count stop	0	R	W
3	Timer 4 count stop bit (T34M3)	0: Count start 1: Count stop	0	R	W
4	Timer 4 count source selection bit (T34M4)	0: Internal clock 1: f(XIN)/2	0	R	W
5	Timer 3 external count source selection bit (T34M5)	0: External clock from P23/TIM3 pin 1: External clock from HSYNC pin	0	R	W
6,7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 2.6.4 Timer 34 mode register (address $00F5_{16}$)

Table 2.6.1 Memory map of timer-related registers

Addresses	Contents
$00F0_{16}$	Timer 1 (TM1)
$00F1_{16}$	Timer 2 (TM2)
$00F2_{16}$	Timer 3 (TM3)
$00F3_{16}$	Timer 4 (TM4)
$00F4_{16}$	Timer 12 mode register (T12M)
$00F5_{16}$	Timer 34 mode register (T34M)

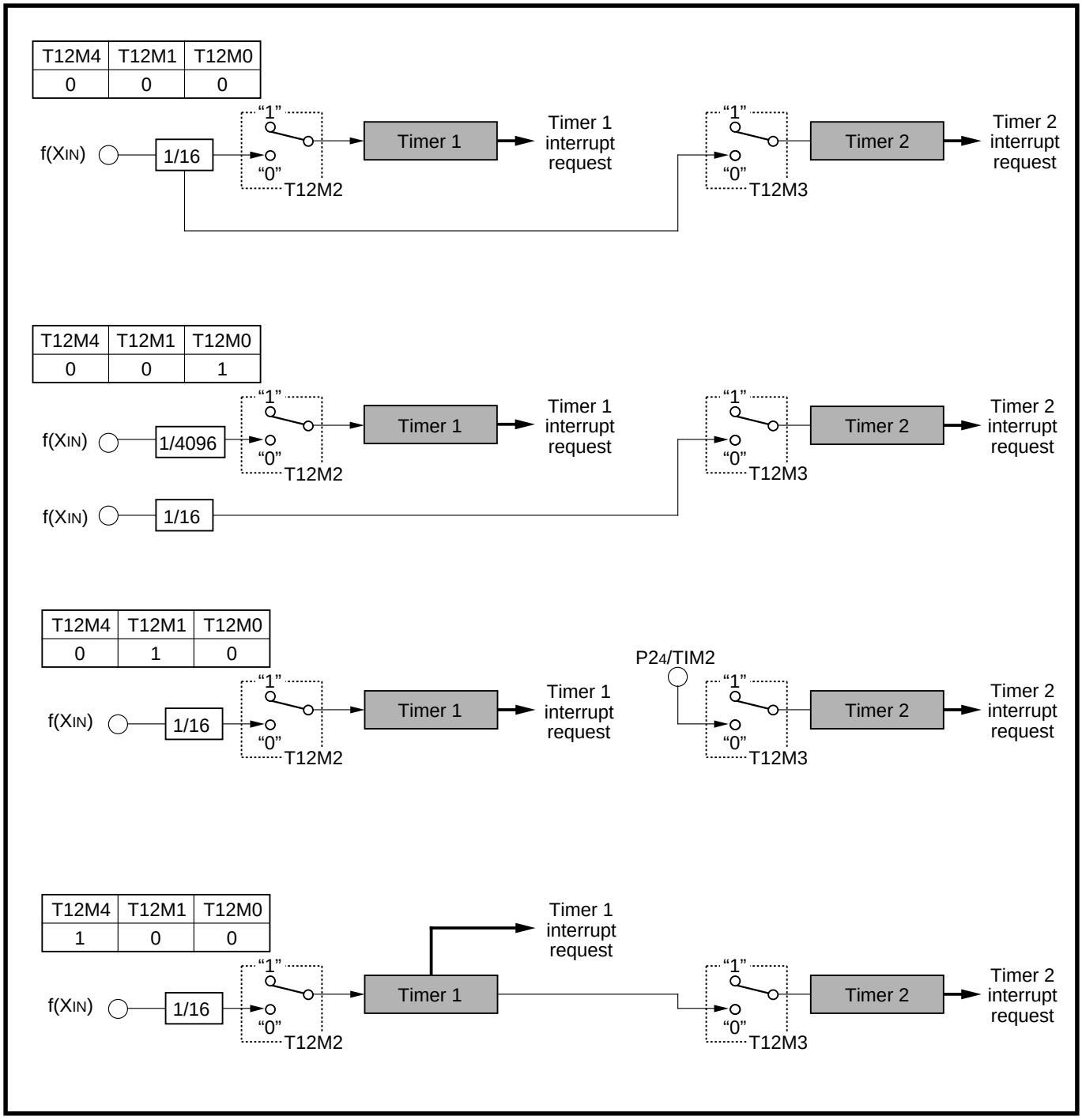


Fig. 2.6.5 Example of timer system

2.6.2 Timer 3 and timer 4 when reset and when executing the STP instruction

Timers 3 and 4 start counting down immediately after reset status is released or stop mode is released, and CPU starts operating by supplying the internal clock ϕ at overflow of these timers. Therefore, the program can start under a stable clock.

(1) When reset

When reset, Timers 3 and 4 are automatically set by hardware as shown in Table 2.6.2, and immediately start counting down. The counting is continued, then, Timer 4 overflows and the internal clock ϕ is supplied (the internal reset is released). The program can start again.

(2) When executing the STP instruction

Immediately after the **STP** instruction is executed, Timers 3 and 4 are automatically set as shown in Table 2.6.2 as in the case of reset and placed in the stop mode. When the stop mode is entered, the processor stops supplying the internal clock ϕ, and contents of Timers 3 and 4 are retained.

When the stop mode is released by reset input or external interrupt input, the processor simultaneously supplies f(X_{IN}), and Timers 3 and 4 start counting down.

The counting is continued, then, when timer 4 overflows and the internal clock ϕ is supplied. The program can start again.

Table 2.6.2 Contents of timers 3 and 4 when reset or when executing STP instruction

Contents	Timer 3	Timer 4
Value	FF_{16}	07_{16}
Count source	f(X_{IN})/16 (except when executing the **STP** instructions)	Timer 3 overflow signal

Note: When executing the **STP** instruction, f(X_{IN})/16 is not automatically selected as the timer 3 count source. Accordingly, set bit 0 of the timer 34 mode register (address $00F5_{16}$) to "0" before executing the **STP** instruction select (f(X_{IN})/16 is selected as the timer 3 count source).

2.7 Serial I/O

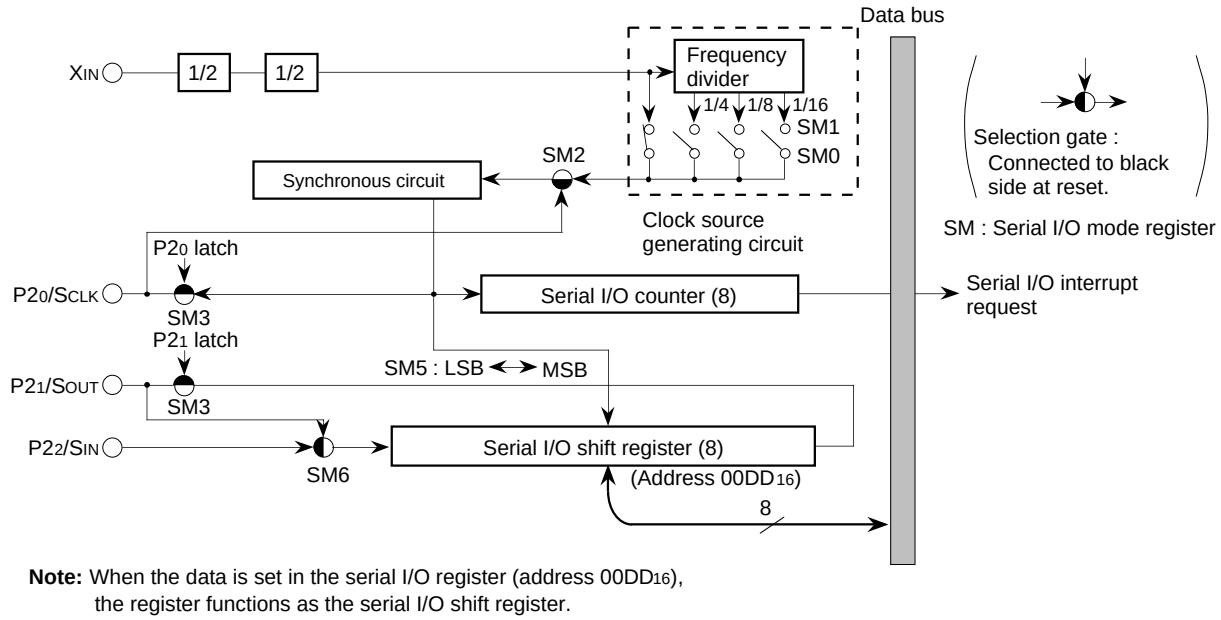
The M37221M6-XXXSP/FP has on-chip clock synchronous serial I/O which can receive and transmit 8-bit data serially.
Because pin S_{OUT} also can be used as the serial I/O data input pin, it can transmit and receive with only one signal line.

2.7.1 Structure of serial I/O

Serial I/O consists of
- Serial I/O register
- Serial I/O mode register
- Serial I/O counter
- Clock source generating counter

The serial I/O register is the register which 8-bit transfer data is written into. Each function of serial I/O can be controlled by setting appropriate values to the serial I/O mode register.
Serial I/O transfers data to and from the internal CPU via the data bus, and it transfers data to and from external devices via ports $P2_2$–$P2_0$. When using the serial I/O, ports $P2_2$–$P2_0$ have the following functions:
- $P2_0$: Serial I/O synchronous clock input/output pin (S_{CLK})
- $P2_1$: Serial I/O data input/output pin (S_{OUT})
- $P2_2$: Serial I/O data input pin (S_{IN})

The functions of these ports can be selected by the serial I/O mode register.
The transfer clock that determines the serial data transfer rate can selected 4 kinds of clock sources with the serial I/O mode register.
Figure 2.7.1 shows the serial I/O block diagram, Figure 2.7.2 shows the serial I/O mode register.



Note: When the data is set in the serial I/O register (address 00DD16), the register functions as the serial I/O shift register.

Fig. 2.7.1 Serial I/O block diagram

Serial I/O Mode Register

b7 b6 b5 b4 b3 b2 b1 b0

0

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

B	Name	Functions	After reset	R	W
0, 1	Internal synchronous clock selection bits (SM0, SM1)	b1 b0 0 0: f(XIN)/4 0 1: f(XIN)/16 1 0: f(XIN)/32 1 1: f(XIN)/64	0	R	W
2	Synchronous clock selection bit (SM2)	0: External clock 1: Internal clock	0	R	W
3	Serial I/O port selection bit (SM3)	0: P20, P21 functions as port 1: SCLK, SOUT	0	R	W
4	Fix this bit to "0."		0	R	W
5	Transfer direction selection bit (SM5)	0: LSB first 1: MSB first	0	R	W
6	Serial input pin selection bit (SM6)	0: Input signal from SIN pin 1: Input signal from SOUT pin	0	R	W
7	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—

Fig. 2.7.2 Serial I/O mode register (address 00DC16)

2.7 Serial I/O

2.7.2 Serial I/O register (address $00DD_{16}$)

The serial I/O register is serial-parallel conversion register used for data transfer. This register consists of 8-bit and can be used as both transmit and receive register.

Serial I/O register is assigned to address $00DD_{16}$.

Although data transfer is performed bit by bit, it is possible to specify whether the data is transferred beginning with most-significant-bit (MSB) or least-significant-bit (LSB) by using bit 5 of the serial I/O mode register.

(1) When bit 5 of the serial I/O mode register is "0"

- Receive: Data is received bit by bit beginning with the MSB (bit 7) of the serial I/O register.
- Transmit: Data is transmitted bit by bit beginning with the LSB (bit 0) of the serial I/O register.

(2) When bit 5 of the serial I/O mode register is "1"

- Receive: Data is received bit by bit beginning with the LSB (bit 0) of the serial I/O register.
- Transmit: Data is transmitted bit by bit beginning with the MSB (bit 7) of the serial I/O register.

2.7.3 Clock source generating circuit

The clock source generating circuit can select oscillation frequency divided by 4, 16, 32, and 64 as the internal clock. Also, it can select an external clock (the external clock is selected immediately after reset). Bit 2 of the serial I/O mode register specifies internal clock or external clock. When bit 2 of the serial I/O mode register is set to "0," an external clock is selected, when "1," an internal clock is selected.

When selecting an internal clock, set the division of oscillation frequency by bits 0 and 1 of the serial I/O mode register.

- Oscillation frequency divided by 4: set both bits 1 and 0 to "0."
- Oscillation frequency divided by 16: set bit 1 = "0" and bit 0 = "1."
- Oscillation frequency divided by 32: set bit 1 = "1" and bit 0 = "0."
- Oscillation frequency divided by 64: set both bit 1 and 0 to "1."

Table 2.7.1 Clock source selection

Serial I/O clock		Serial I/O mode register		
		Bit 2	Bit 1	Bit 0
External clock		0	0 or 1 (Invalid)	
Internal clock	$f(X_{IN})/4$	1	0	0
	$f(X_{IN})/16$		0	1
	$f(X_{IN})/32$		1	0
	$f(X_{IN})/64$		1	1

The contents of bits 0 and 1 of the serial I/O mode register are invalid, when selecting an external clock.

2.7.4 Serial input/output common transmission/reception mode

Pin $P2_1/S_{OUT}$ can also be used as the serial I/O data input pin when the serial input pin selection bit (bit 6 of the serial I/O mode register; address $00DC_{16}$) is set to "1." (It is not necessary to set the corresponding bit of port P2 direction register to input mode.)

With this function, pin $P2_2/S_{IN}$ can also be used as general-purpose input port $P2_2$.

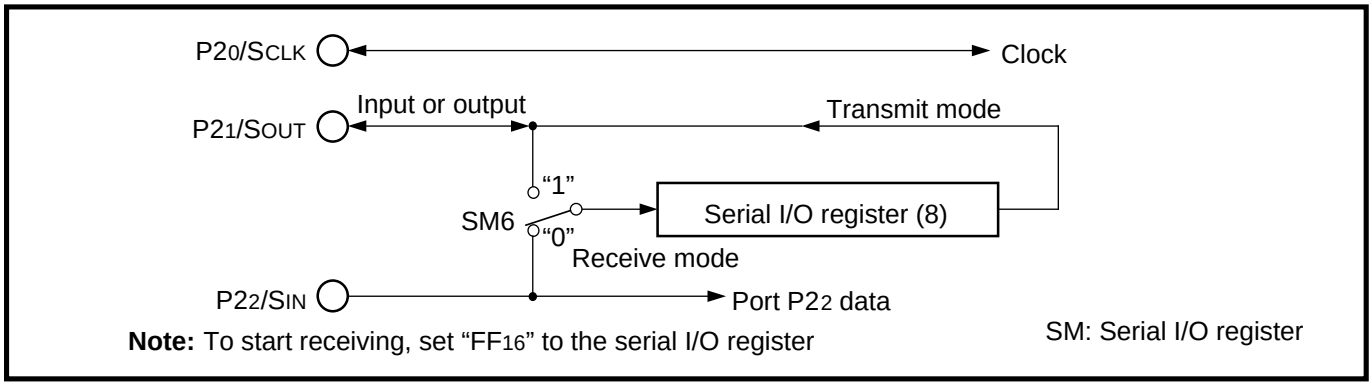


Fig. 2.7.3 Serial input/output common transfer mode block diagram

2.7.5 Serial I/O data receive method (when an internal clock is selected)

(1) Initialization

First, set the serial I/O mode register (address $00DC_{16}$) as follows.

① Select the synchronous clock (SM2 = "1," SM1, SM0).

② Set $P2_0$ as pin S_{CLK} (SM3 = "1"). Pin $P2_1/S_{OUT}$ is not used when receiving serial data. However, since the serial I/O port selection bit (SM3) is also used for setting pin S_{OUT}, port $P2_1$ is automatically set as pin S_{OUT} and loses its general-purpose I/O port function.

③ Select the serial input pin by the serial input pin selection bit (SM6). When SM6 = "0," signal is input from pin $P2_2/S_{IN}$; when SM6 = "1," signal is input from pin $P2_1/S_{OUT}$. When pin $P2_2/S_{IN}$ is a input pin, set the port P2 direction register to input mode ("0"). For pins $P2_0/S_{CLK}$ and $P2_1/S_{OUT}$, the corresponding bits of the port P2 direction register are automatically set by setting the serial I/O mode register.

(2) Receive enable state

After the above setting have been made, write "FF_{16}" to the serial I/O register (address $00DD_{16}$). The serial I/O counter is then set to "07_{16}" during the write cycle and receive is enabled.

(3) Receive operation

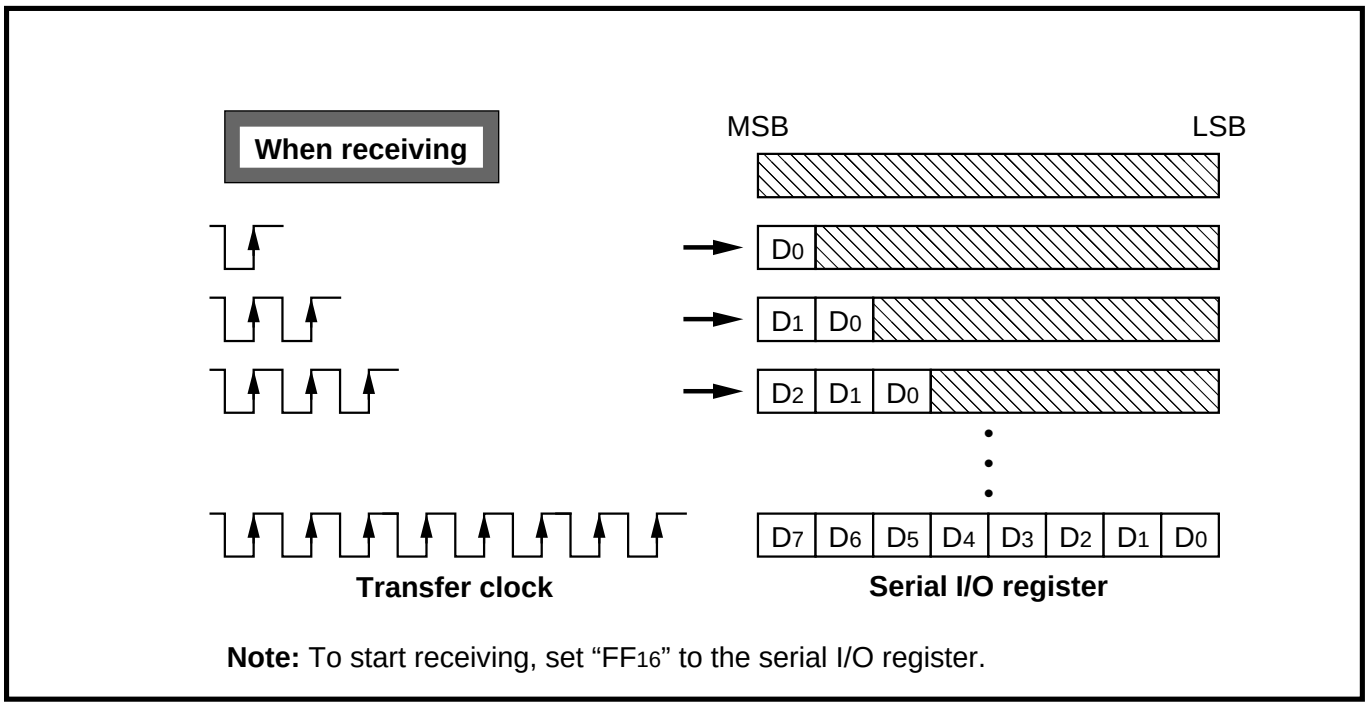
The data from the serial I/O data input pins (S_{OUT} or S_{IN}) is received one bit at a time into the serial I/O register in synchronization with rising edges of the transfer clock.

Receive operation is performed according to bit 5 (SM5) of the serial I/O mode register:

① When SM5 is set to "0," data is received from MSB (bit 7) of the register and shifted to the right (to low-order bit) every time new data is received.

② When SM5 is set to "1," data is received from LSB (bit 0) of the register and shifted to the left (to high-order bit) every time new data is received.

When all 8-bit data have been received, the serial I/O interrupt request bit (bit 2) of the interrupt request register 2 (address $00FD_{16}$) is set to "1."



Note: To start receiving, set "FF_{16}" to the serial I/O register.

Fig. 2.7.4 Serial I/O register when receiving (when SM5 = "0")

2.7 Serial I/O

2.7.6 Serial I/O data transmit method (when an external clock is selected)

(1) Initialization

First, set the serial I/O mode register (address $00DC_{16}$) as follows.

① Select the synchronous clock (SM2 = "0").

② Set $P2_0$ as pin S_{CLK} (SM3 = "1"). Since the serial I/O port selection bit (SM3) is also used for the setting pin S_{OUT}, port $P2_1$ is automatically becomes the S_{OUT} pin.

Note: It is not necessary to set pin $P2_2/S_{IN}$ as pin S_{IN} when transmitting. It can be used as general-purpose input pin.

(2) Transmit enable state

When transmit data are written to the serial I/O register, the serial I/O counter is set to "07_{16}" and transmit is enabled.

(3) Transmit operation

When transmit is enabled (the serial I/O counter value = "07_{16}"), simultaneously, the data of the serial I/O register is transmitted from pin $P2_1/S_{OUT}$ in synchronization with a falling edge of the transfer clock.

Transmission is performed according to bit 5 (SM5) of the serial I/O mode register:

① When SM5 is set to "0," data is transmitted from LSB (bit 0) of the register and shifted to the right (to low-order bit) every time new data is transmitted.

② When SM5 is set to "1," data is transmitted from MSB (bit 7) of the register and shifted to the left (to high-order bit) every time new data is transmitted.

When all 8-bit data have been transmitted, the serial I/O interrupt request bit (bit 2) of the interrupt request register 2 (address $00FD_{16}$) is set to "1."

Pin $P2_1/S_{OUT}$ will be in after transmit operation has been completed.

Note: On programming, note that the serial I/O counter is set even by writing to the serial I/O register with bit management instructions, such as **SEB** and **CLB**.

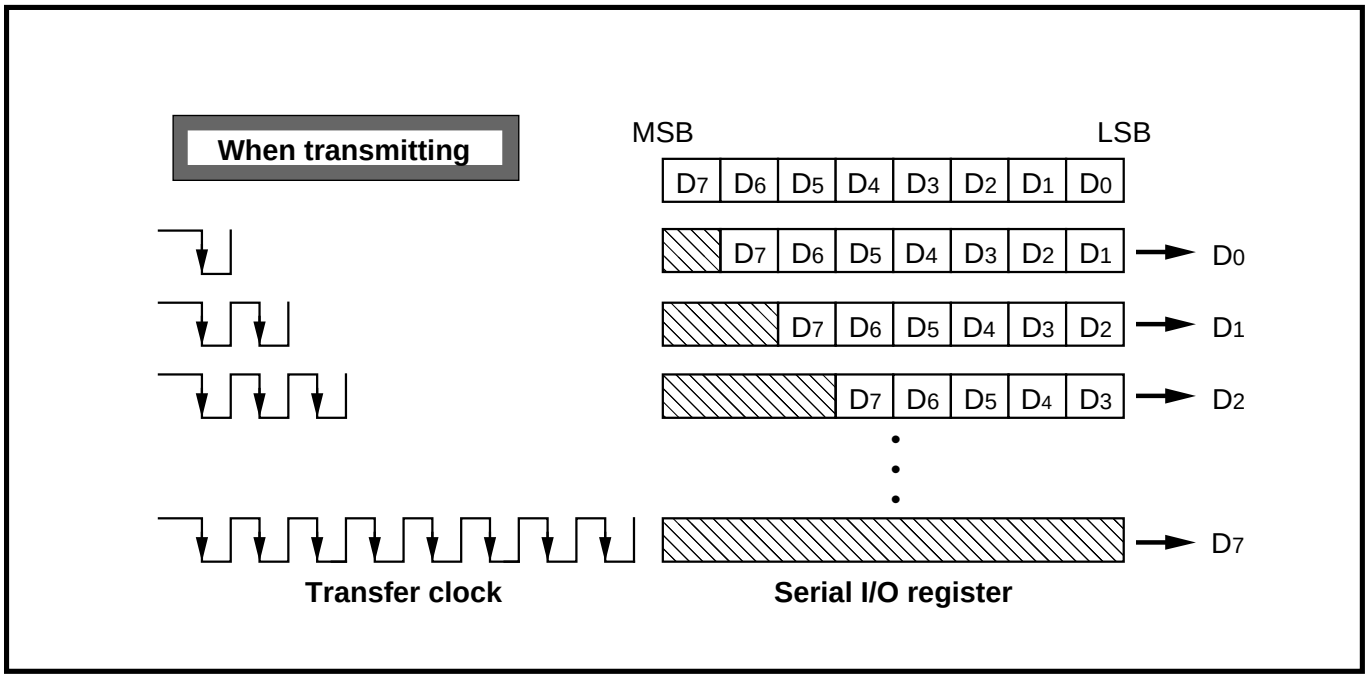
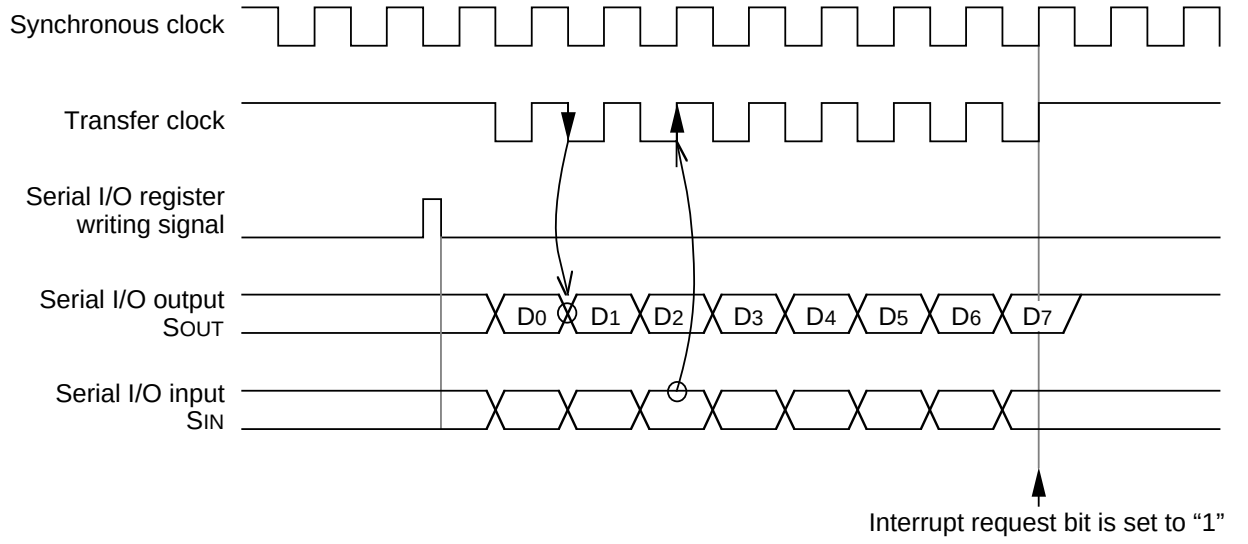


Fig. 2.7.5 Serial I/O register when transmitting (when SM5 = "0")

2.7.7 Note when selecting a synchronous clock

Regardless of either an internal or external clock is selected as the serial I/O synchronous clock source, the interrupt request bit is set to “1” after 8 transfer clocks.
However, the serial I/O register contents will continue to be shifted as long as the transfer clock is being input to the serial I/O circuit, so it is necessary to stop after 8 transfer clocks.
When an internal clock is selected, the transfer clock stops automatically after 8 clocks.
When an external clock is selected, control the transfer clock externally. Moreover, use an external clock of 1 MHz or less with a duty cycle of 50 %.
When selecting an external clock as the synchronizing clock, write transmit data to the serial I/O register transfer clock input level is HIGH
Figure 2.7.6 shows the serial I/O timing.



Note: When an internal clock is selected, pin SOUT is at high-impedance after transfer is completed.

Fig. 2.7.6 Timing diagram of serial I/O

2.7 Serial I/O

The transmit side in Figure 2.7.7, $P2_1$ is set as the serial I/O data output pin and $P2_0$ is set as the serial I/O synchronous clock output pin by the initialization program.
The receive side, $P2_1$ is set as the serial I/O data input pin and $P2_0$ is used for the serial I/O synchronous clock (external clock) input pin by the initialization program.
Figure 2.7.8 shows the serial data transmit/receive processing sequence using the above structure.

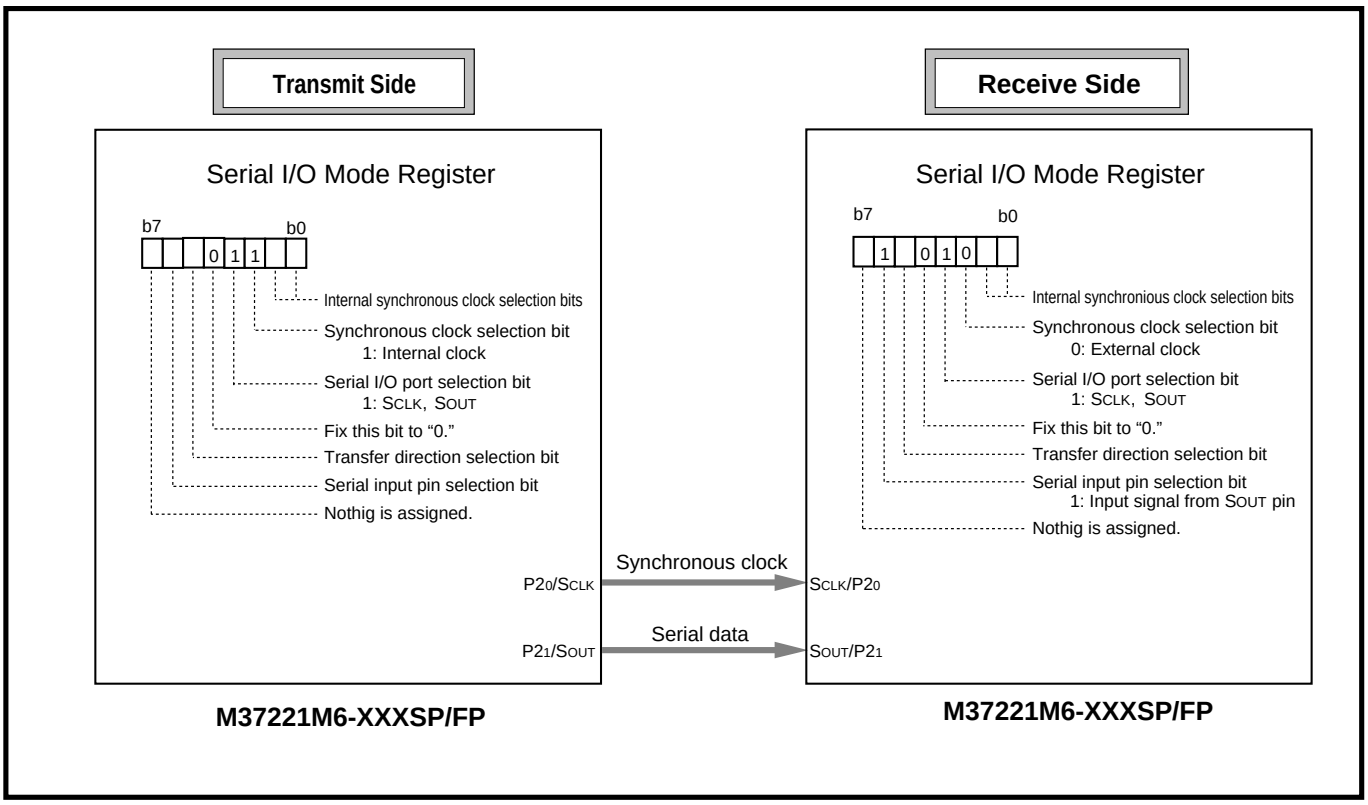


Fig. 2.7.7 Connection example for serial I/O transmit/receive

Transmit side			Receive side		
LDM	#$0C, $DC	; Set serial I/O mode register.	LDM	#$48, $DC	; Set serial I/O mode register.
CLB	2, $FD	; Reset serial I/O interrupt request bit.	CLB	2, $FD	; Reset serial I/O interrupt request bit.
SEB	2, $FF	; Set the serial I/O interrupt enable bit to "1."	SEB	2, $FF	; Set serial I/O interrupt enable bit to "1."
LDM	#DATA, $DD	; Write transfer data to serial I/O register.	▼LDM	#$FF, $DD	; Write dummy data to serial I/O register.

As a result of the above processing 1-byte data is transferred from the transmit side to the receive side. When the transmit operation is completed, interrupts occur on both sides, so that completion of the data transfer can be reported.
After that, repeating the processing after the symbol (▼) can transmit/receive more data.

Fig. 2.7.8 Serial data transmit/receive processing sequence

2.8 Multi-master I²C-BUS interface

The multi-master I²C-BUS interface is a serial communications circuit, conforming to the Philips I²C-BUS data transfer format. This interface, offering both an arbitration lost detection and a synchronous functions, is useful for the multi-master serial communications.

Figure 2.8.1 shows a block diagram of the multi-master I²C-BUS interface and Table 2.8.1 shows multi-master I²C-BUS interface functions.

The M37220M3-XXSP/FP does not have this function.

Table 2.8.1 Multi-master I²C-BUS interface functions

Item	Function
Format	In conformity with Philips I²C-BUS standard: 10-bit addressing format 7-bit addressing format High-speed clock mode Standard clock mode
Communication mode	In conformity with Philips I²C-BUS standard: Master transmission Master reception Slave transmission Slave reception
SCL clock frequency	16.1 kHz to 400 kHz (at ϕ = 4 MHz)

ϕ: System clock = f(X_{IN})/2

Note: We are not responsible for any third party's infringement of patent rights or other rights attributable to the use of the control function (bits 6 and 7 of the I²C control register at address $00DA_{16}$) for connections between the I²C-BUS interface and ports (SCL1, SCL2, SDA1, SDA2).

2.8 Multi-master I²C-BUS interface

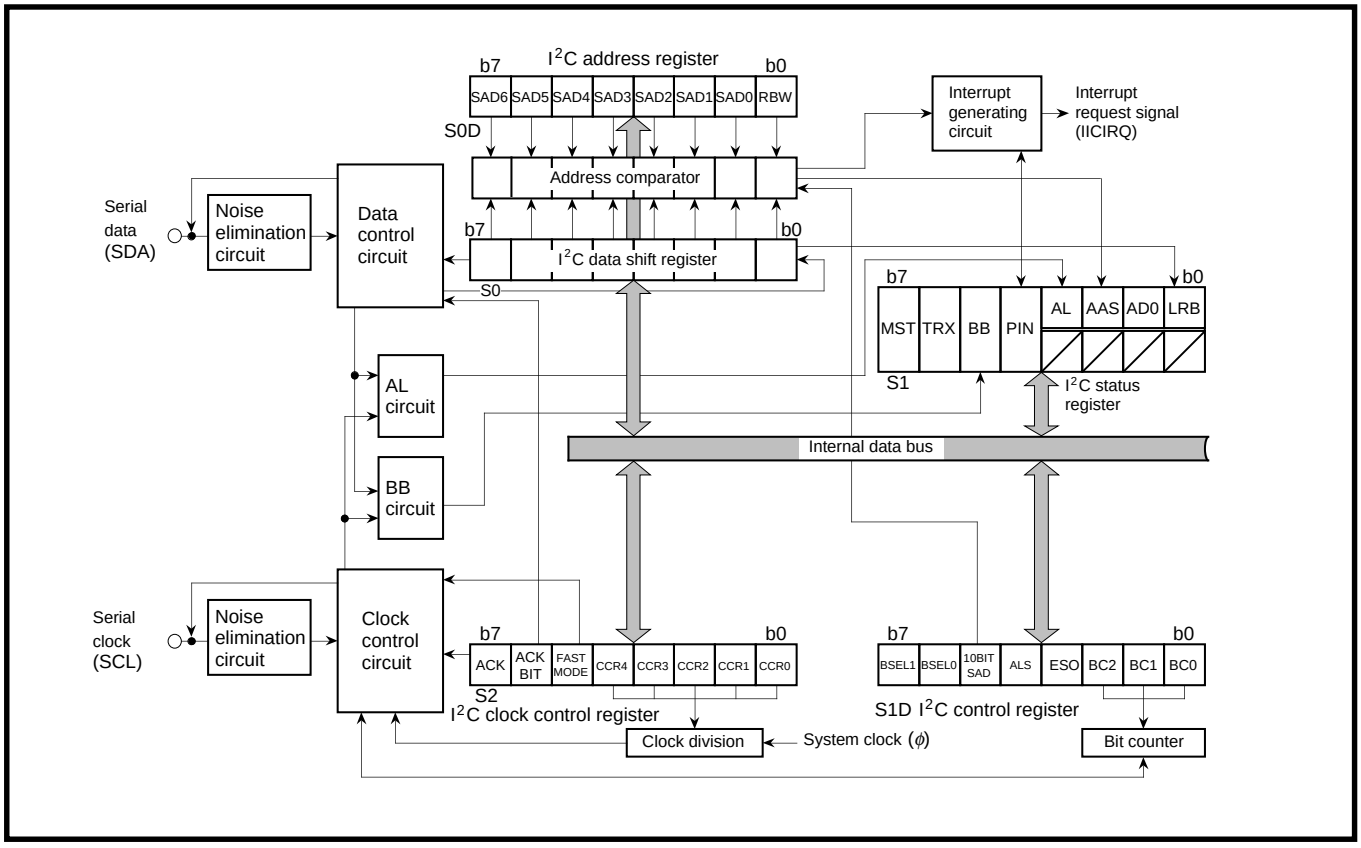


Fig. 2.8.1 Block diagram of multi-masteer I²C-BUS interface

2.8.1 Construction of multi-master I²C-BUS interface

The multi-master I²C-BUS interface consists of the following :

- I²C address register
- I²C data shift register
- I²C clock control register
- I²C control register
- I²C status register
- Other control circuits

The data transfer with the internal CPU is performed via data bus, the data transfer with an external device is performed via ports $P1_1$–$P1_4$. When using multi-master I²C-BUS interface, these ports $P1_1$–$P1_4$ are assigned to the following functions.

- $P1_1$: Multi-master I²C-BUS interface Synchronous clock input/output pin 1 (SCL1)
- $P1_2$: Multi-master I²C-BUS interface Synchronous clock input/output pin 2 (SCL2)
- $P1_3$: Multi-master I²C-BUS interface data input/output pin 1 (SDA1)
- $P1_4$: Multi-master I²C-BUS interface data input/output pin 2 (SDA2)

The shift clock to determine the transfer speed of serial data is selected by the I²C clock control register (refer to "**Figure 2.8.4**").
A serial data and a serial clock is referred as "SDA," "SCL" respectively, hereafter.

2.8.2 Multi-master I²C-BUS interface-related registers

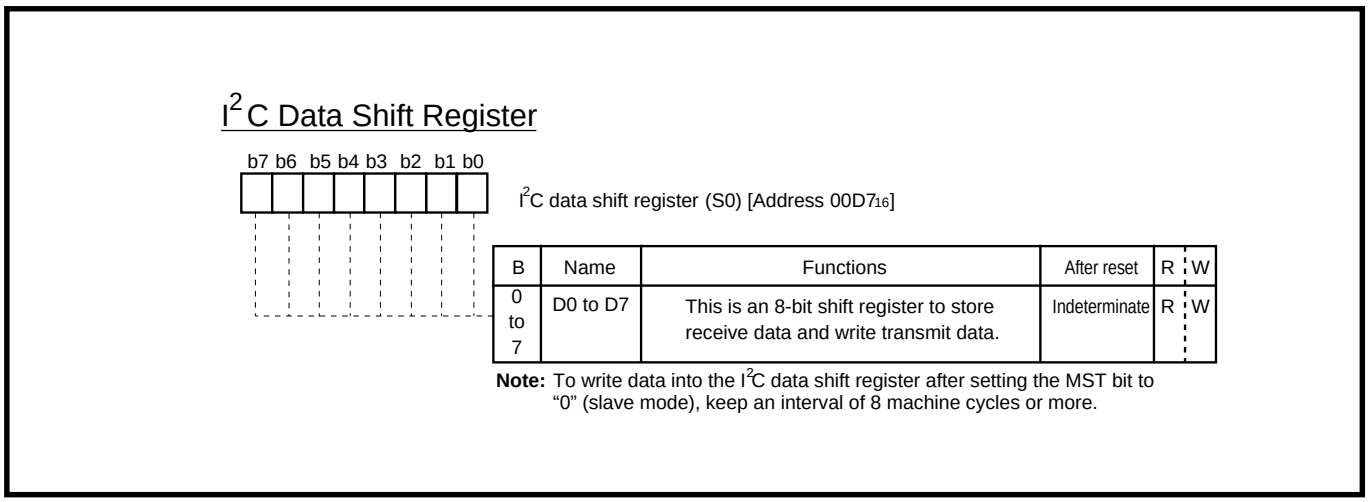
(1) I²C data shift register (S0: address $00D7_{16}$)

The I²C data shift register (S0 : address $00D7_{16}$) is an 8-bit shift register to store receive data and write transmit data.

When transmit data is written into this register, it is transferred to the outside from bit 7 in synchronization with the SCL clock, and each time one-bit data is output, the data of this register are shifted one bit to the left. When data is received, it is input to this register from bit 0 in synchronization with the SCL clock, and each time one-bit data is input, the data of this register are shifted one bit to the left.

The I²C data shift register is in a write enable status only when the ESO bit of the I²C control register (address $00DA_{16}$) is "1." The bit counter is reset by a write instruction to the I²C data shift register. When both the ESO bit and the MST bit of the I²C status register (address $00F9_{16}$) are "1," the SCL is output by a write instruction to the I²C data shift register. Reading data from the I²C data shift register is always enabled regardless of the ESO bit value.

Figure 2.8.2 shows the I²C data shift register.



B	Name	Functions	After reset	R	W
0 to 7	D0 to D7	This is an 8-bit shift register to store receive data and write transmit data.	Indeterminate	R	W

Note: To write data into the I²C data shift register after setting the MST bit to "0" (slave mode), keep an interval of 8 machine cycles or more.

Fig. 2.8.2 I²C data shift register

2.8 Multi-master I²C-BUS interface

(2) I²C address register (S0D: address 00D8₁₆)

The I²C address register (address 00D8₁₆) consists of a 7-bit slave address and a $\overline{\text{read}}$/write bit. In the addressing mode, the slave address written in this register is compared with the address data to be received immediately after the START condition are detected.

■ **Bit 0: $\overline{\text{Read}}$/write bit (RBW)**

Not used when comparing addresses, in the 7-bit addressing mode. In the 10-bit addressing mode, the first address data to be received is compared with the contents (SAD6 to SAD0 + RBW) of the I²C address register.

The RBW bit is cleared to "0" automatically when the stop condition is detected.

■ **Bits 1 to 7: Slave address (SAD0–SAD6)**

These bits store slave addresses. Regardless of the 7-bit addressing mode and the 10-bit addressing mode, the address data transmitted from the master is compared with the contents of these bits.

Figure 2.8.3 shows the I²C address register.

I²C Address Register

b7 b6 b5 b4 b3 b2 b1 b0

I²C address register (S0D) [Address 00D8₁₆]

B	Name	Functions	After reset	R	W
0	$\overline{\text{Read}}$/write bit (RBW)	0: Read 1: Write	0	R	W
1 to 7	Slave address (SAD0 to SAD6)	The address data transmitted from the master is compared with the contents of these bits.	0	R	W

Fig. 2.8.3 I²C address register

(3) I²C clock control register (S2: address 00DB$_{16}$)

The I²C clock control register (address 00DB$_{16}$) is used to set ACK control, SCL mode and SCL frequency.

■ **Bits 0 to 4: SCL frequency control bits (CCR0–CCR4)**

These bits control the SCL frequency. Refer to **"Table 2.8.4."**

■ **Bit 5: SCL mode specification bit (FAST MODE)**

This bit specifies the SCL mode. When this bit is set to "0," the standard clock mode is set. When the bit is set to "1," the high-speed clock mode is set.

■ **Bit 6: ACK bit (ACK BIT)**

This bit sets the SDA status when an ACK clock* is generated. When this bit is set to "0," the ACK return mode is set and SDA goes to LOW at the occurrence of an ACK clock. When the bit is set to "1," the ACK non-return mode is set. The SDA is held in the HIGH status at the occurrence of an ACK clock.

However, when the slave address matches the address data in the reception of address data at ACK BIT = "0," the SDA is automatically made LOW (ACK is returned). If there is a mismatch between the slave address and the address data, the SDA is automatically made HIGH (ACK is not returned).

✻ACK clock: Clock for acknowledgment

■ **Bit 7: ACK clock bit (ACK)**

This bit specifies a mode of acknowledgment which is an acknowledgment response of data transmission. When this bit is set to "0," the no ACK clock mode is set. In this case, no ACK clock occurs after data transmission. When the bit is set to "1," the ACK clock mode is set and the master generates an ACK clock upon completion of each 1-byte data transmission. The device for transmitting address data and control data releases the SDA at the occurrence of an ACK clock (make SDA HIGH) and receives the ACK bit generated by the data receiving device.

Figure 2.8.4 shows the I²C clock control register.

Note: Do not write data into the I²C clock control register during transmission. If data is written during transmission, the I²C clock generator is reset, so that data cannot be transmitted normally.

I²C Clock Control Register

b7 b6 b5 b4 b3 b2 b1 b0

I²C clock control register (S2 : address 00DB$_{16}$)

B	Name	Functions			After reset	R	W
0 to 4	SCL frequency control bits (CCR0 to CCR4)	Setup value of CCR4–CCR0	Standard clock mode	High speed clock mode	0	R	W
		00 to 02	Setup disabled	Setup disabled			
		03	Setup disabled	333			
		04	Setup disabled	250			
		05	100	400 (See note)			
		06	83.3	166			
		⋮	500/CCR value	1000/CCR value			
		1D	17.2	34.5			
		1E	16.6	33.3			
		1F	16.1	32.3			
		(at ϕ = 4 MHz, unit : kHz)					
5	SCL mode specification bit (FAST MODE)	0 : Standard clock mode 1 : High-speed clock mode			0	R	W
6	ACK bit (ACK BIT)	0 : ACK is returned. 1 : ACK is not returned.			0	R	W
7	ACK clock bit (ACK)	0 : No ACK clock 1 : ACK clock			0	R	W

Note: At 4000kHz in the high-speed clock mode, the duty is as below .
"0" period : "1" period = 3 : 2
In the other cases, the duty is as below.
"0" period : "1" period = 1 : 1

Fig. 2.8.4 I²C clock control register

(4) I²C Control Register (S1D: address 00DA$_{16}$)

The I²C control register (address 00DA$_{16}$) controls the data communication format.

■ **Bits 0 to 2: Bit counter (BC0–BC2)**

These bits decide the number of bits for the next 1-byte data to be transmitted. An interrupt request signal occurs immediately after the number of bits specified with these bits are transmitted. When a START condition is received, these bits become "000$_2$" and the address data is always transmitted and received in 8 bits.

■ **Bit 3: I²C-BUS interface use enable bit (ESO)**

This bit enables usage of the multi-master I²C-BUS interface. When this bit is set to "0," the use disable status is provided, so the SDA and the SCL become high-impedance. When the bit is set to "1," use of the interface is enabled.

When ESO = "0," the following is performed.

- PIN = "1," BB = "0" and AL = "0" are set (they are bits of the I²C status register at address 00F8$_{16}$).
- Writing data to the I²C data shift register (address 00D7$_{16}$) is disabled.

■ **Bit 4: Data format selection bit (ALS)**

This bit decides whether or not to recognize slave addresses. When this bit is set to "0," the addressing format is selected, so that address data is recognized. When a match is found between a slave address and address data as a result of comparison or when a general call (refer to "(5) I²C Status Register," bit 1) is received, transmission processing can be performed. When this bit is set to "1," the free data format is selected, so that slave addresses are not recognized.

■ **Bit 5: Addressing format selection bit (10BIT SAD)**

This bit selects a slave address specification format. When this bit is set to "0," the 7-bit addressing format is selected. In this case, only the high-order 7 bits (slave address) of the I²C address register (address 00D8$_{16}$) are compared with address data. When this bit is set to "1," the 10-bit addressing format is selected, all the bits of the I²C address register are compared with address data.

■ **Bit 6 and 7: Connection control bits between I²C-BUS interface and ports (BSEL0, BSEL1)**

This bits controls the connection between SCL and ports or SDA and ports. When using the ports as multi-master I²C-BUS interface, set the corresponding bits of port P1 direction register to "1" (output mode).

Figure 2.8.5 shows the connection port control by BSEL0 and BSEL1, Figure 2.8.6 shows the I²C control register.

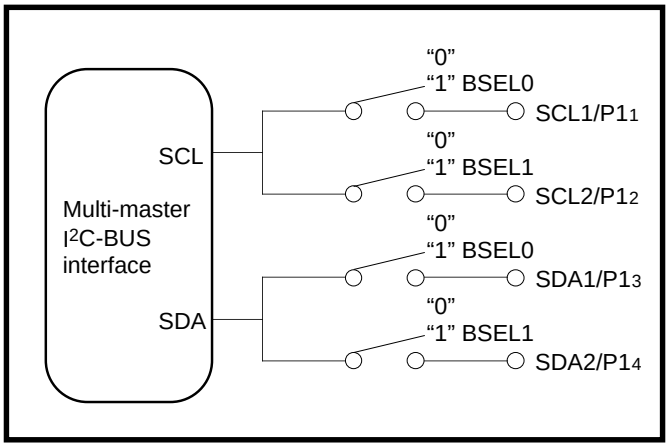
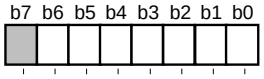


Fig. 2.8.5 Connection port control by BSEL0 and BSEL1

2.8 Multi-master I²C-BUS interface

I²C Control Register



I²C control register (S1D : address 00DA16)

B	Name	Functions	After reset	R	W
0 to 2	Bit counter (Number of transmit/recieve bits) (BC0 to BC2)	b2 b1 b0 0 0 0 : 8 0 0 1 : 7 0 1 0 : 6 0 1 1 : 5 1 0 0 : 4 1 0 1 : 3 1 1 0 : 2 1 1 1 : 1	0	R	W
3	I²C-BUS interface use enable bit (ESO)	0 : Disabled 1 : Enabled	0	R	W
4	Data format selection bit (ALS)	0 : Addressing mode 1 : Free data format	0	R	W
5	Addressing format selection bit (10BIT SAD)	0 : 7-bit addressing format 1 : 10-bit addressing format	0	R	W
6, 7	Connection control bits between I²C-BUS interface and ports (BSEL0, BSEL1)	b7 b6 Connection port 0 0 : None 0 1 : SCL1, SDA1 1 0 : SCL2, SDA2 1 1 : SCL1, SDA1 SCL2, SDA2	0	R	W

Note: When using ports P11-P14 as I²C-BUS interface, the output structure changes automatically from CMOS output to N-channel open-drain output. However, set the port direction register to "1" (output mode).

Fig. 2.8.6 I²C control register

(5) I²C status register (S1: address 00D9₁₆)

The I²C status register (address 00D9₁₆) controls the I²C-BUS interface status. The low-order 4 bits are read-only bits and the high-order 4 bits can be read out and written to.

■ **Bit 0: Last receive bit (LRB)**

This bit stores the last bit value of received data and can also be used for ACK receive confirmation. If ACK is returned when an ACK clock occurs, the LRB bit is set to "0." If ACK is not returned, this bit is set to "1." Except in the ACK mode, the last bit value of received data is input. The state of this bit is changed from "1" to "0" by executing a write instruction to the I²C data shift register (address 00D7₁₆).

■ **Bit 1: General call detecting flag (AD0)**

This bit is set to "1" when a general call* whose address data is all "0" is received in the slave mode. By a general call of the master device, every slave device receives control data after the general call. The AD0 bit is set to "0" by detecting the STOP condition or START condition.

■ **Bit 2: Slave address comparison flag (AAS)**

This flag indicates a comparison result of address data.

①In the slave receive mode, when the 7-bit addressing format is selected, this bit is set to "1" in one of the following conditions.

- The address data immediately after occurrence of a START condition matches the slave address stored in the high-order 7 bits of the I²C address register (address 00D8₁₆).
- A general call is received.

②In the slave reception mode, when the 10-bit addressing format is selected, this bit is set to "1" with the following condition.

- When the address data is compared with the I²C address register (8 bits consists of slave address and RBW), the first bytes match.

③The state of this bit is changed from "1" to "0" by executing a write instruction to the I²C data shift register (address 00D7₁₆).

■ **Bit 3: Arbitration lost* detecting flag (AL)**

In the master transmission mode, when a device other than the microcomputer sets the SDA to LOW by any other device, arbitration is judged to have been lost, so that this bit is set to "1." At the same time, the TRX bit is set to "0," so that immediately after transmission of the byte, whose arbitration was lost is completed, the MST bit is set to "0." When arbitration is lost during slave address transmission, the TRX bit is set to "0" and the reception mode is set. Consequently, it becomes possible to receive and recognize its own slave address transmitted by another master device.

✻Arbitration lost: The status in which communication as a master is disabled.

■ **Bit 4: I²C-BUS interface interrupt request bit (PIN)**

This bit generates an interrupt request signal. Each time 1-byte data is transmitted, the state of the PIN bit changes from "1" to "0." At the same time, an interrupt request signal is sent to the CPU. The PIN bit is set to "0" in synchronization with a falling edge of the last clock (including the ACK clock) of an internal clock and an interrupt request signal occurs in synchronization with a falling edge of the PIN bit. When the PIN bit is "0," the SCL is kept in the "0" state and clock generation is disabled. Figure 2.8.7 shows an interrupt request signal generating timing chart.

The PIN bit is set to "1" in any one of the following conditions.

- Executing a write instruction to the I²C data shift register (address $00D7_{16}$).
- When the ESO bit is "0"
- At reset

The conditions in which the PIN bit is set to "0" are shown below:

- Immediately after completion of 1-byte data transmission (including when arbitration lost is detected)
- Immediately after completion of 1-byte data reception
- In the slave reception mode, with ALS = "0" and immediately after completion of slave address or general call address reception
- In the slave reception mode, with ALS = "1" and immediately after completion of address data reception

■ **Bit 5: Bus busy flag (BB)**

This bit indicates the status of use of the bus system. When this bit is set to "0," this bus system is not busy and a START condition can be generated. When this bit is set to "1," this bus system is busy and the occurrence of a START condition is disabled by the START condition duplication prevention function (Note).

This flag can be written by software only in the master transmission mode. In the other modes, this bit is set to "1" by detecting a START condition and set to "0" by detecting a STOP condition. When the ESO bit of the I²C control register (address $00DA_{16}$) is "0" and at reset, the BB flag is kept in the "0" state.

■ **Bit 6: Communication mode specification bit (transfer direction specification bit: TRX)**

This bit decides the direction of transfer for data communication. When this bit is "0," the reception mode is selected and the data of a transmitting device is received. When the bit is "1," the transmission mode is selected and address data and control data are output into the SDA in synchronization with the clock generated on the SCL.

When the ALS bit of the I²C control register (address $00F9_{16}$) is "0" in the slave reception mode is selected, the TRX bit is set to "1" (transmit) if the least significant bit (R/$\overline{W}$ bit) of the address data transmitted by the master is "1." When the ALS bit is "0" and the R/$\overline{W}$ bit is "0," the TRX bit is cleared to "0" (receive).

The TRX bit is cleared to "0" in one of the following conditions.

- When arbitration lost is detected.
- When a STOP condition is detected.
- When occurrence of a START condition is disabled by the START condition duplication prevention function (Note).
- With MST = "0" and when a START condition is detected.
- With MST = "0" and when ACK non-return is detected.
- At reset

■ **Bit 7: Communication mode specification bit (master/slave specification bit: MST)**

This bit is used for master/slave specification for data communication. When this bit is "0," the slave is specified, so that a START condition and a STOP condition generated by the master are received, and data communication is performed in synchronization with the clock generated by the master. When this bit is "1," the master is specified and a START condition and a STOP condition are generated, and also the clocks required for data communication are generated on the SCL. The MST bit is cleared to "0" in one of the following conditions.

- Immediately after completion of 1-byte data transmission when arbitration lost is detected
- When a STOP condition is detected.
- When occurrence of a START condition is disabled by the START condition duplication preventing function (Note).
- At reset

Figure 2.8.7 shows the interrupt request signal generating timing, Figure 2.8.8 shows the I²C status register.

Note: The START condition duplication prevention function disables the START condition generation, reset of bit counter reset, and SCL output when the following condition is satisfied: a START condition is set by another master device.

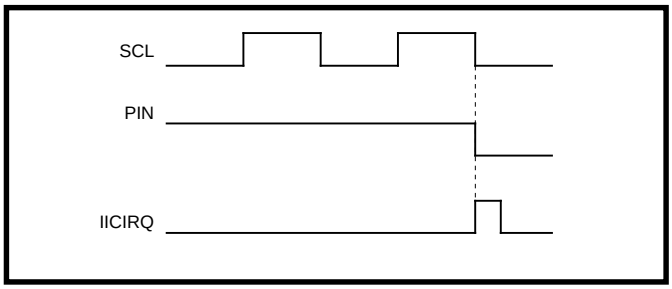


Fig. 2.8.7 Interrupt request signal generating timing

I²C Status Register

b7 b6 b5 b4 b3 b2 b1 b0

I²C status register (S1) [Address 00D9₁₆]

B	Name	Functions	After reset	R	W
0	Last receive bit (LRB) (See note)	0 : Last bit = "0 " 1 : Last bit = "1 "	Indeterminate	R	—
1	General call detecting flag (AD0) (See note)	0 : No general call detected 1 : General call detected	0	R	—
2	Slave address comparison flag (AAS) (See note)	0 : Address mismatch 1 : Address match	0	R	—
3	Arbitration lost detecting flag (AL) (See note)	0 : Not detected 1 : Detected	0	R	—
4	I²C-BUS interface interrupt request bit (PIN)	0 : Interrupt request issued 1 : No interrupt request issued	1	R	W
5	Bus busy flag (BB)	0 : Bus free 1 : Bus busy	0	R	W
6, 7	Communication mode specification bits (TRX, MST)	b7 b6 0 0 : Slave recieve mode 0 1 : Slave transmit mode 1 0 : Master recieve mode 1 1 : Master transmit mode	0	R	W

Note : These bits and flags can be read out, but cannnot be written.

Fig. 2.8.8 I²C status register

2.8 Multi-master I²C-BUS interface

2.8.3 START condition, STOP condition generation method

(1) START condition generation method

When the ESO bit of the I²C control register (address $00DA_{16}$) is "1," execute a write instruction to the I²C status register (address $00D9_{16}$) to set the MST, TRX and BB bits to "1." A START condition will then be generated. After that, the bit counter becomes "000_2" and an SCL for 1 byte is output. The START condition generating timing and BB bit set timing are different in the standard clock mode and the high-speed clock mode. Refer to **"Figure 2.8.9"** for the START condition generation timing diagram, and **"Table 2.8.2"** for the START condition/STOP condition generation timing table.

(2) STOP condition generation method

When the ESO bit of the I²C control register (address $00DA_{16}$) is "1," execute a write instruction to the I²C status register (address $00D9_{16}$) for setting the MST bit and the TRX bit to "1" and the BB bit to "0". A STOP condition will then be generated. The STOP condition generation timing and the BB flag reset timing are different in the standard clock mode and the high-speed clock mode. Refer to **"Figure 2.8.10"** for the STOP condition generating timing diagram, and **"Table 2.8.2"** for the START condition/STOP condition generation timing table.

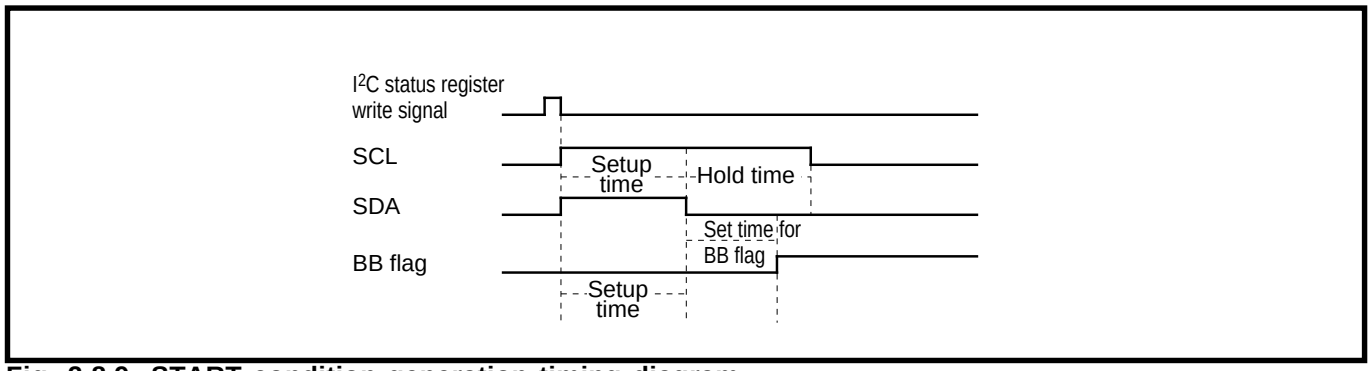


Fig. 2.8.9 START condition generation timing diagram

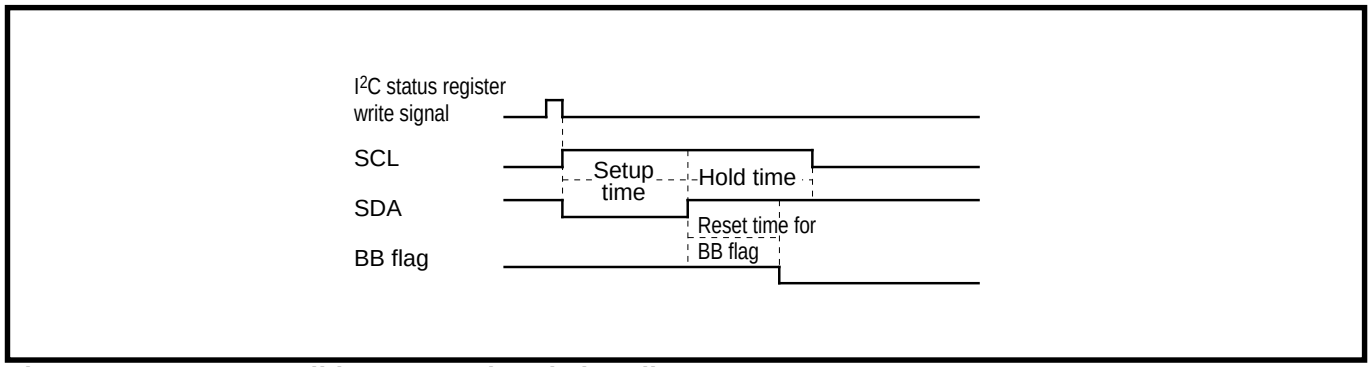


Fig. 2.8.10 STOP condition generation timing diagram

Table 2.8.2 START condition/STOP condition generation timing table

Item	Standard clock mode	High-speed clock mode
Setup time	5.0 μs (20 cycles)	2.5 μs (10 cycles)
Hold time	5.0 μs (20 cycles)	2.5 μs (10 cycles)
Set/reset time for BB flag	3.0 μs (12 cycles)	1.5 μs (6 cycles)

Note: Absolute time at ϕ = 4 MHz. The value in parentheses denotes the number of ϕ cycles.

(3) START/STOP condition detect conditions

The START/STOP condition detect conditions are shown in Figure 2.8.11 and Table 2.8.3. Only when the 3 conditions of Table 10 are satisfied, a START/STOP condition can be detected.

Note: When a STOP condition is detected in the slave mode (MST = 0), an interrupt request signal "IICIRQ" is generated to the CPU.

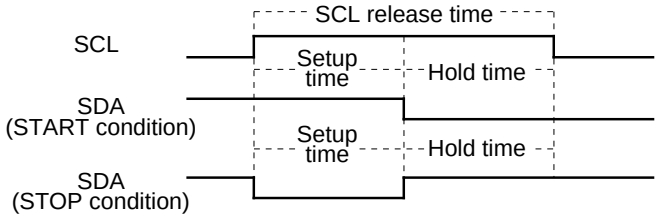


Fig. 2.8.11 START condition/STOP condition detect timing diagram

Table 2.8.3 START condition/STOP condition detect conditions

Standard clock mode	High-speed clock mode
6.5 μs (26 cycles) < SCL release time	1.0 μs (4 cycles) < SCL release time
3.25 μs (13 cycles) < Setup time	0.5 μs (2 cycles) < Setup time
3.25 μs (13 cycles) < Hold time	0.5 μs (2 cycles) < Hold time

Note: Absolute time at ϕ = 4 MHz. The value in parentheses denotes the number of ϕ cycles.

(4) Address Data Communication

There are two address data communication formats, namely, 7-bit addressing format and 10-bit addressing format. The respective address communication formats is described below.

① 7-bit addressing format

To meet the 7-bit addressing format, set the 10BIT SAD bit of the I²C control register (address $00DA_{16}$) to "0." The first 7-bit address data transmitted from the master is compared with the high-order 7-bit slave address stored in the I²C address register (address $00D8_{16}$). At the time of this comparison, address comparison of the RBW bit of the I²C address register (address $00D8_{16}$) is not made. For the data transmission format when the 7-bit addressing format is selected, refer to **"Figure 2.8.12, (1) and (2)."**

② 10-bit addressing format

To meet the 10-bit addressing format, set the 10BIT SAD bit of the I²C control register (address $00DA_{16}$) to "1." An address comparison is made between the first-byte address data transmitted from the master and the 7-bit slave address stored in the I²C address register (address $00D8_{16}$). At the time of this comparison, an address comparison between the RBW bit of the I²C address register (address $00D8_{16}$) and the R/$\overline{W}$ bit which is the last bit of the address data transmitted from the master is made. In the 10-bit addressing mode, the R/$\overline{W}$ bit which is the last bit of the address data not only specifies the direction of communication for control data but also is processed as an address data bit.

When the first-byte address data matches the slave address, the AAS bit of the I²C status register (address $00D9_{16}$) is set to "1." After the second-byte address data is stored into the I²C data shift register (address $00D7_{16}$), make an address comparison between the second-byte data and the slave address by software. When the address data of the 2nd byte matches the slave address, set the RBW bit of the I²C address register (address $00D8_{16}$) to "1" by software. This processing can match the 7-bit slave address and R/$\overline{W}$ data, which are received after a RESTART condition is detected, with the value of the I²C address register (address $00D8_{16}$). For the data transmission format when the 10-bit addressing format is selected, refer to **"Figure 2.8.12, (3) and (4)."**

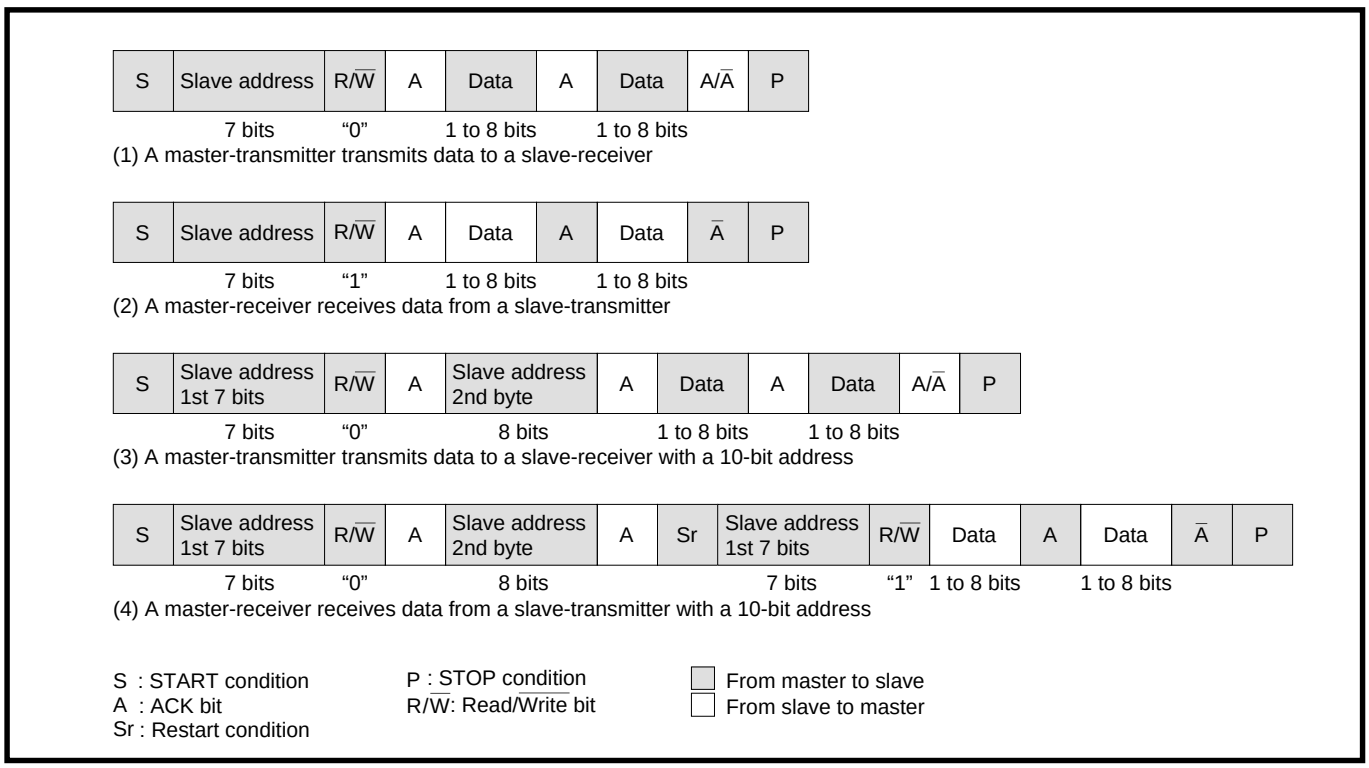


Fig. 2.8.12 Address data communication format

2.9 A-D comparator

The M37221M6-XXXSP/FP has A-D comparator consists of the 6-bit D-A converter by resistance string method and a comparator. Figure 2.9.1 shows the A-D comparator block diagram.

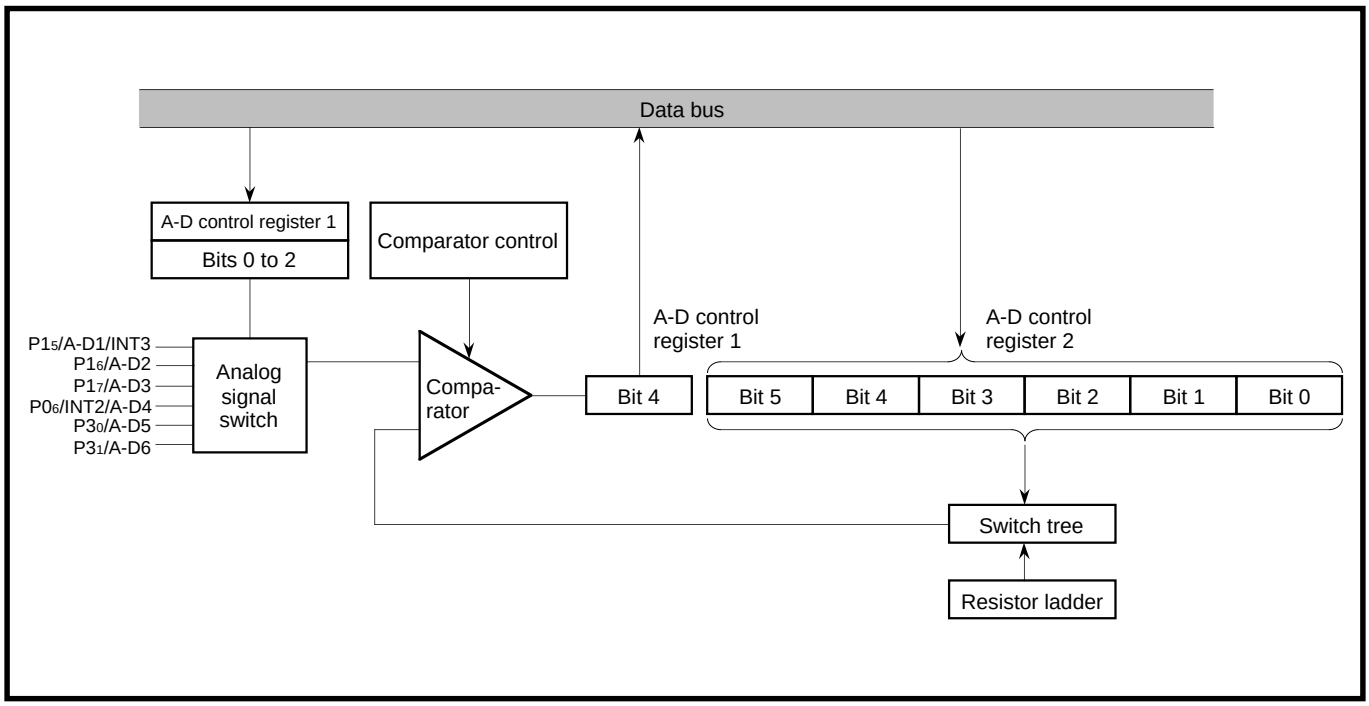


Fig. 2.9.1 A-D comparator block diagram

The following explains A-D comparison method.

① Set "0" to corresponding bits of the direction register to use ports as analog input pins.
② Select the analog input pin with bits 0 to 2 of A-D control register 1 (address $00EE_{16}$).
③ Set the comparison voltage "V_{ref}" for D-A conversion by bits 0 to 5 of A-D control register 2 (address $00EF_{16}$). Table 2.9.1 shows the V_{ref} values corresponding to the set values above.
A-D comparison starts by writing to A-D control register 2.
④ This voltage comparison needs for 16 machine cycles (**NOP** instruction × 8).
⑤ The comparison result is stored in bit 4 of the A-D control register 1 (address $00EE_{16}$).

When the input voltage value is lower than the comparison voltage value, bit 4 is cleared to "0"; when the input voltage value is higher than the comparison voltage value, bit 4 is set to "1" (refer to **"Figure 2.9.2"**).

2.9 A-D comparator

Table 2.9.1 Relationship between contents of A-D control register 2 and reference voltage "V_{ref}"

A-D control register 2						Internal analog voltage (comparison voltage V_{ref})
bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	
0	0	0	0	0	0	1/128V_{CC}
0	0	0	0	0	1	3/128V_{CC}
0	0	0	0	1	0	5/128V_{CC}
:	:	:	:	:	:	:
1	1	1	1	0	1	123/128V_{CC}
1	1	1	1	1	0	125/128V_{CC}
1	1	1	1	1	1	127/128V_{CC}

A-D Control Register 1

b7 b6 b5 b4 b3 b2 b1 b0

A-D control register 1 (AD1) [Address $00EE_{16}$]

B	Name	Functions	After reset	R	W
0 to 2	Analog input pin selection bits (ADM0, ADM1, ADM2)	b2 b1 b0 0 0 0 : A-D1 0 0 1 : A-D2 0 1 0 : A-D3 0 1 1 : A-D4 1 0 0 : A-D5 1 0 1 : A-D6 1 1 0 : } Do not set. 1 1 1 : }	0	R	W
3	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
4	Storage bit of comparison result (ADM4)	0: Input voltage < reference voltage 1: Input voltage > reference voltage	Indeterminate	R	W
5 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 2.9.2 A-D control register 1 (address $00EE_{16}$)

A-D Control Register 2

b7 b6 b5 b4 b3 b2 b1 b0

A-D control register 2(AD2) [Address $00EF_{16}$]

B	Name	Functions	After reset	R	W
0 to 5	D-A converter set bits (ADC0, ADC1, ADC2, ADC3, ADC4, ADC5)	b5 b4 b3 b2 b1 b0 0 0 0 0 0 0 : 1/128Vcc 0 0 0 0 0 1 : 3/128Vcc 0 0 0 0 1 0 : 5/128Vcc 1 1 1 1 0 1 : 123/128Vcc 1 1 1 1 1 0 : 125/128Vcc 1 1 1 1 1 1 : 127/128Vcc	0	R	W
6, 7	Nothing is assigned. These bits are write disable bits. When these bits are reed out, the values are " 0."		0	R	—

Fig. 2.9.3 A-D control register 2 (address $00EF_{16}$)

2.10 PWM

The M37221M6-XXXSP/FP has one 14-bit PWM (pulse width modulator) [DA], and six 8-bit PWM [PWM0–PWM5].
Table 2.10.1 shows the PWM function performance.

Table 2.10.1 PWM function performance (at oscillation frequency = 8 MHz)

Performance	14-bit PWM [DA]	8-bit PWM
Resolution (bits)	14	8
Minimum resolution bit width (μs)	0.25	4
Repeat cycle (μs)	4096	1024

Figure 2.10.1 shows the 14-bit PWM block diagram and Figure 2.10.2 shows the 8-bit PWM block diagram.

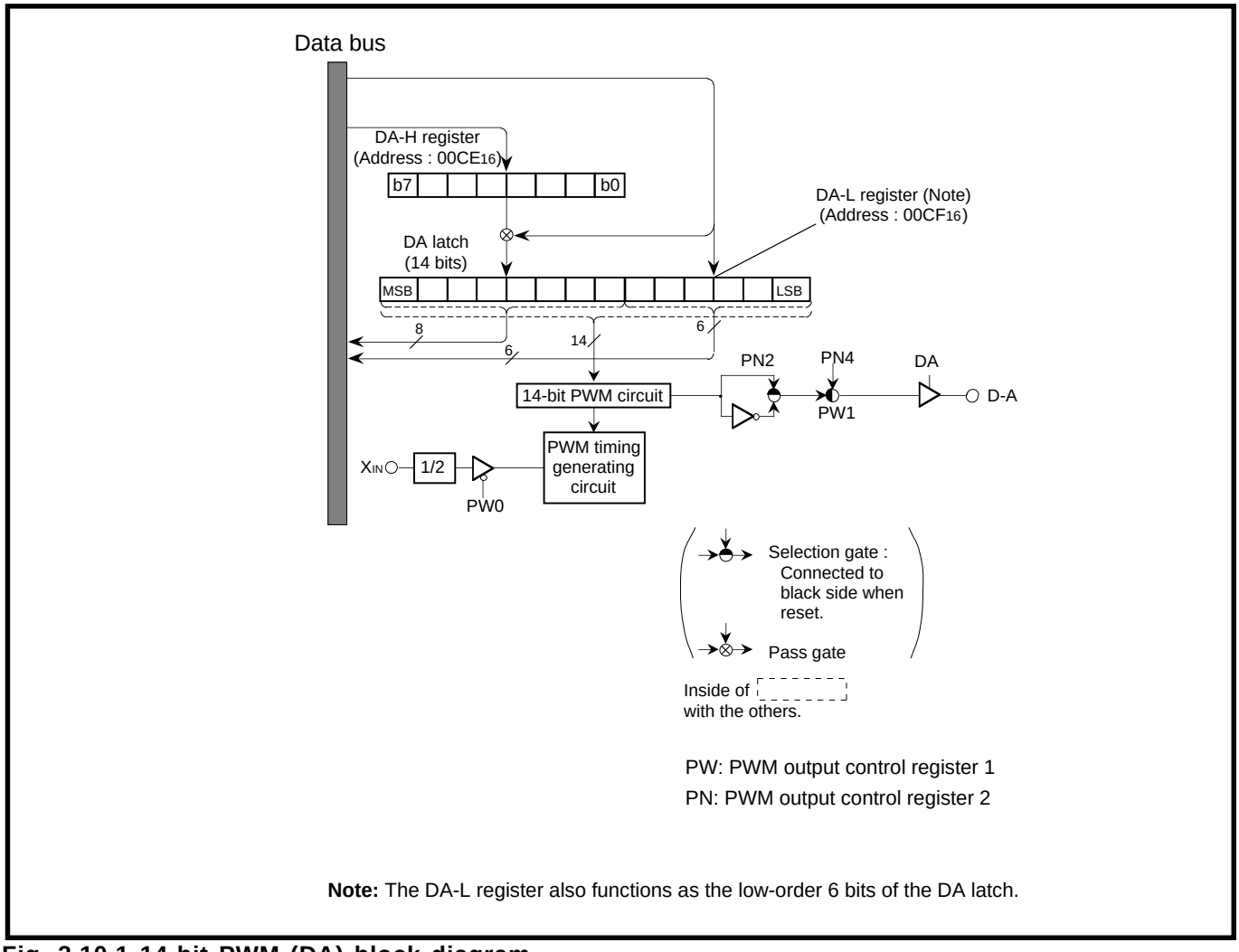


Fig. 2.10.1 14-bit PWM (DA) block diagram

2.10 PWM

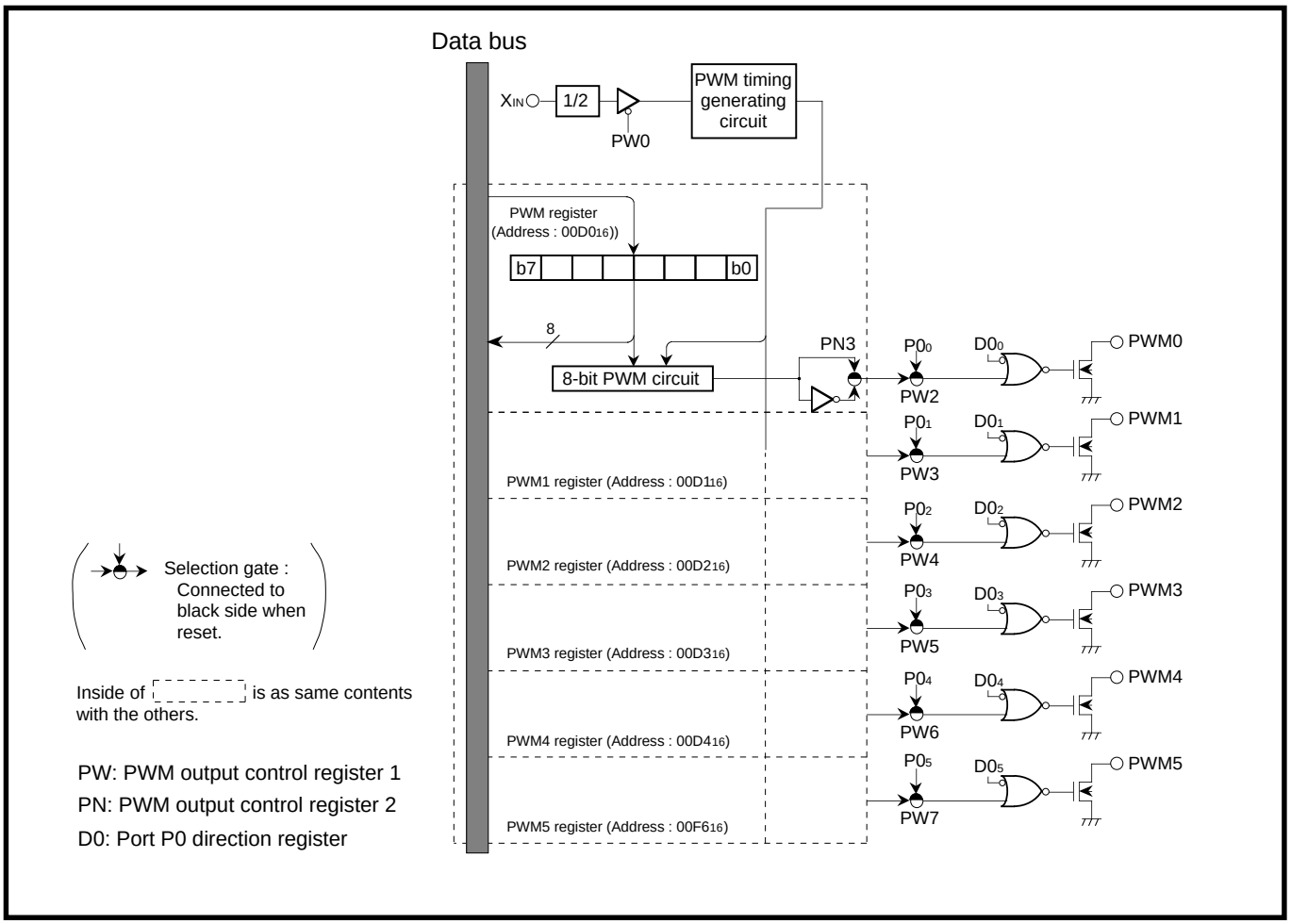


Fig. 2.10.2 8-bit PWM block diagram

2.10.1 8-bit PWM registers (addresses $00D0_{16}$ to $00D4_{16}$ and $00F6_{16}$) /DA registers (addresses $00CE_{16}$ and $00CF_{16}$)

Data transfer from the 8-bit PWM registers (addresses $00D0_{16}$ to $00D4_{16}$ and $00F6_{16}$) to the 8-bit PWM circuit is executed when writing data to the registers. The output signal from the 8-bit PWM output pin corresponds to the contents of this register.

Also, data transfer from the DA registers (addresses $00CE_{16}$ and $00CF_{16}$) to the 14-bit PWM circuit is executed when writing data to the DA-L register (address $00CF_{16}$). The output signal from the D–A output pin corresponds to the contents of the DA latch.

Reading from the DA register (address $00CE_{16}$) means the DA latch contents. Therefore, it is possible to confirm the data being output from the D-A output pin by reading the DA register.

The contents of the 8-bit PWM register and DA register are indeterminate after reset.

2.10.2 14-bit PWM (DA output)

The 14-bit PWM automatically outputs a PWM rectangular waveform from the D-A pin by writing high-order 8 bits of the output data to the DA-H register and the low-order 6 bits to the DA-L register. Data of the DA-H register are transferred to the 14-bit PWM circuit when writing to the DA-L register.

The following explains the output operation of 14-bit PWM rectangular waveform (when $f(X_{IN})$ = 8 MHz).

①The repeat cycle "T" (4,096 μs) of output waveform is divided into 2^6 = 64 smaller interval "t" (t = 64 μs). The "t" is further divided into the minimum resolution bit "τ" of 2^8 = 256 (τ = 0.25 μs).

②The HIGH duration of the fundamental waveform is determined by the high-order 8 bits "D_H" of the DA latch.

HIGH duration (time) = $\tau \times D_H$ (when $f(X_{IN})$ = 8 MHz, $0.25D_H$ μs)

Because the D_H values are "0" to "255," the HIGH duration can be selected a total of 256.

③The smaller interval "t_m" with a longer HIGH level area by "τ" is specified by the low-order 6 bits "D_L" of the DA latch. The t_m is specified from among 64 smaller intervals (t_0 to t_{63}).

Therefore, a rectangular waveform consisted of 2-kind waveforms with different HIGH duration are output from pin D–A (a length of entirely HIGH output cannot be output).

Figure 2.10.3 shows the 14-bit PWM output example, Table 2.10.2 shows the relation between D_L and t_m (m = "0" to "63").

Table 2.10.2 The relation between D_L and t_m (m = "0" to "63")

Low-order 6-bit data of DA register (D_L)	Smaller intervals that HIGH duration is longer by τ t_m (m = "0" to "63")	Number
MSB 0 0 0 0 0 0 LSB	Nothing	0
0 0 0 0 0 1	m = 32	1
0 0 0 0 1 0	m = 16, 48	2
0 0 0 0 1 1	m = 16, 32, 48	3
0 0 0 1 0 0	m = 8, 24, 40, 56	4
0 0 0 1 0 1	m = 8, 24, 32, 40, 56	5
0 0 0 1 1 0	m = 8, 16, 24, 40, 48, 56	6
:	:	:
0 0 1 0 0 0	m = 4, 12, 20, 28, 36, 44, 52, 60	8
:	:	:
0 1 0 0 0 0	m = 2, 6, 10, 14, 18, ... 46, 50, 54, 58, 62	16
:	:	:
1 0 0 0 0 0	m = 1, 3, 5, 7, 9 ... 55, 57, 59, 61, 63	32
:	:	:
1 0 1 0 0 0	m = 1, 3, 5, 7, 9 ... 52, 55, 57, 59, 60, 61, 63	40
:	:	:
1 1 1 1 1 1	m = 1 to 63 ("0" is not included)	63

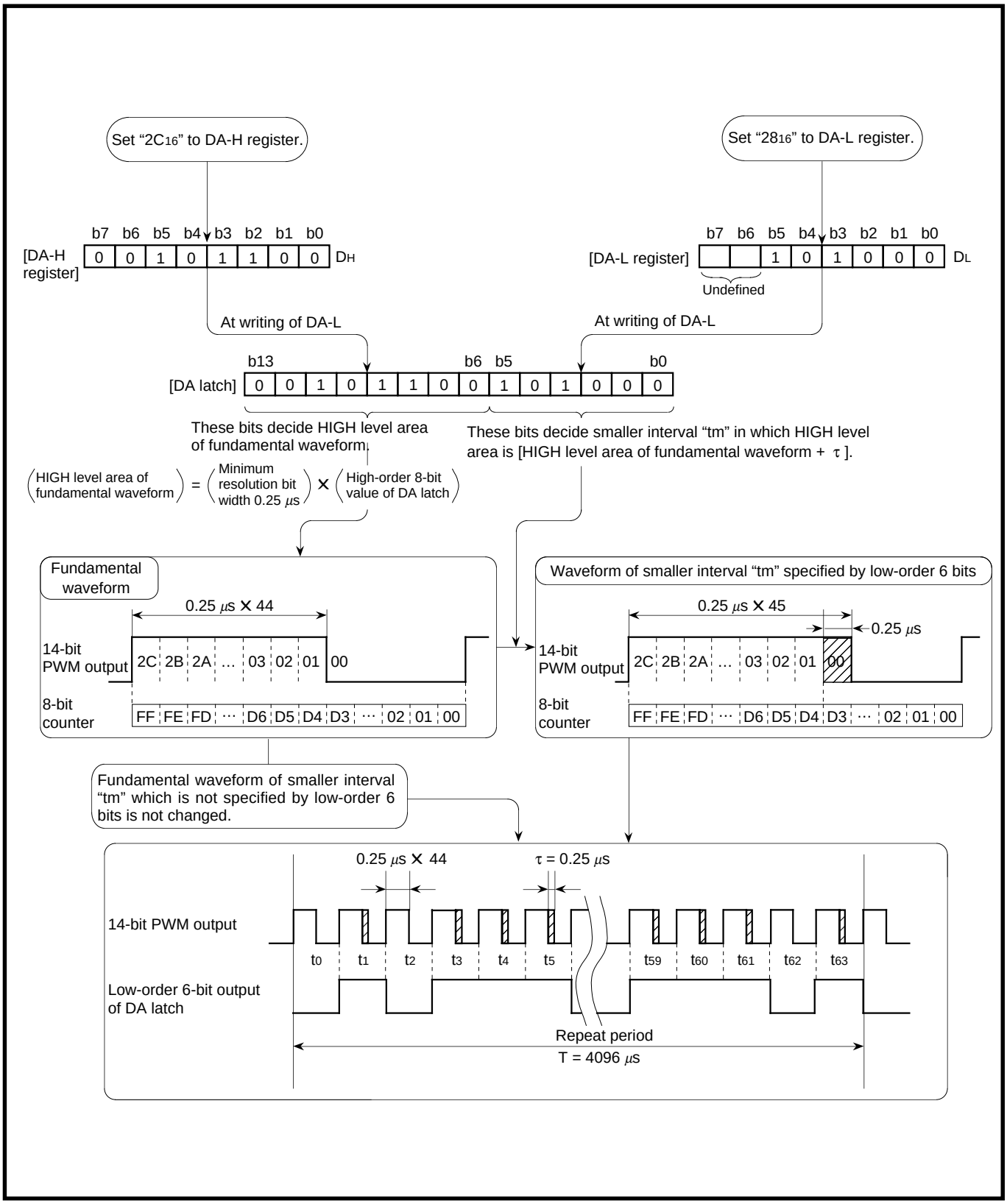


Fig. 2.10.3 14-bit PWM output example ($f(X_{IN})$ = 8 MHz)

2.10.3 8-bit PWM (PWM0 to PWM5: address $00D0_{16}$ to $00D4_{16}$ and $00F6_{16}$)

The 8-bit PWM outputs waveform which is the logical sum (OR) of pulses corresponding to bits 0 to 7 of the 8-bit PWM register.

That is to say, 8 kinds of pulses corresponding to the weight of each bit of the 8-bit PWM register are output inside the circuit during 1 cycle. Among these pulses, OR of pulses that correspond to bits, which is set to “1,” in the 8-bit PWM register to external devices as PWM output.

Figure 2.10.4 shows the pulse waveforms corresponding to the weight of each bit of the 8-bit PWM register. Figure 2.10.5 shows the example of 8-bit PWM output.

As shown in the Figures, 256 kinds of output (HIGH duration: 0/256 to 255/256) are selected by changing the contents of the PWM register (a length of entirely HIGH cannot be output).

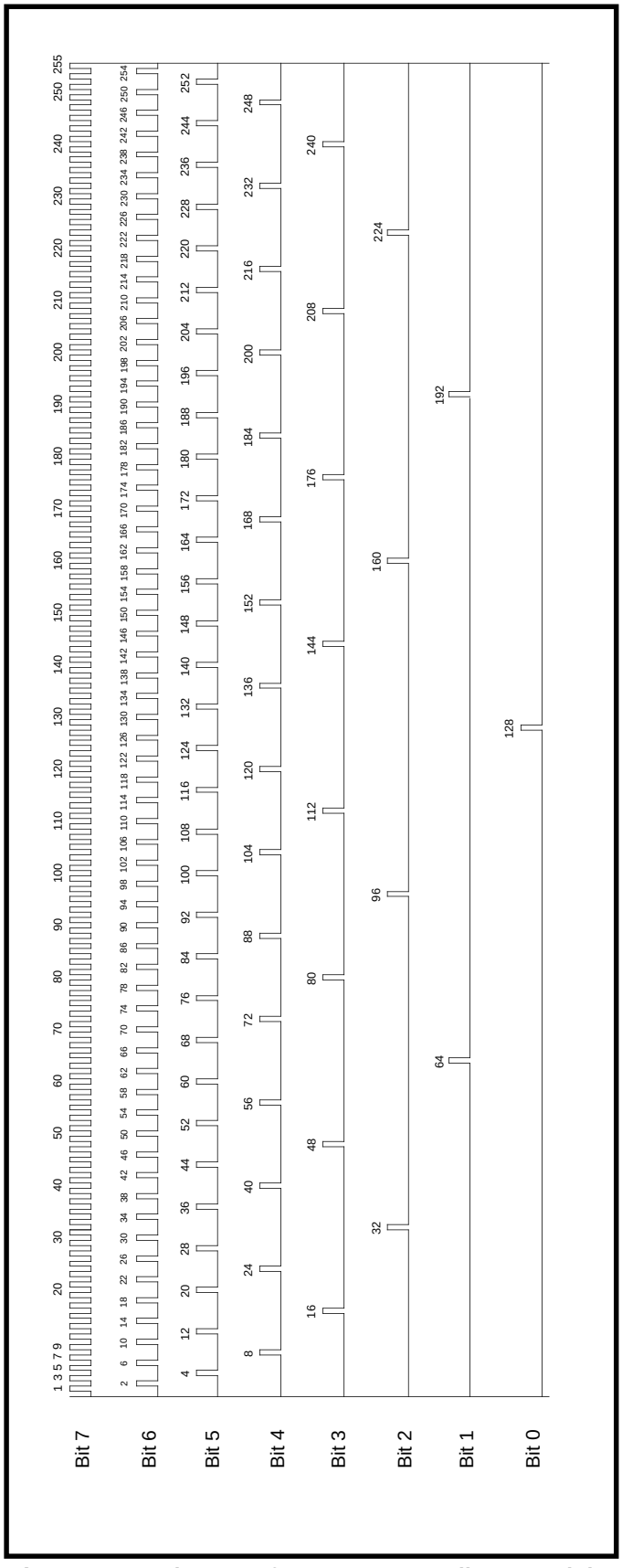


Fig. 2.10.4 Pulse waveforms corresponding to weight of each bit of 8-bit PWM register

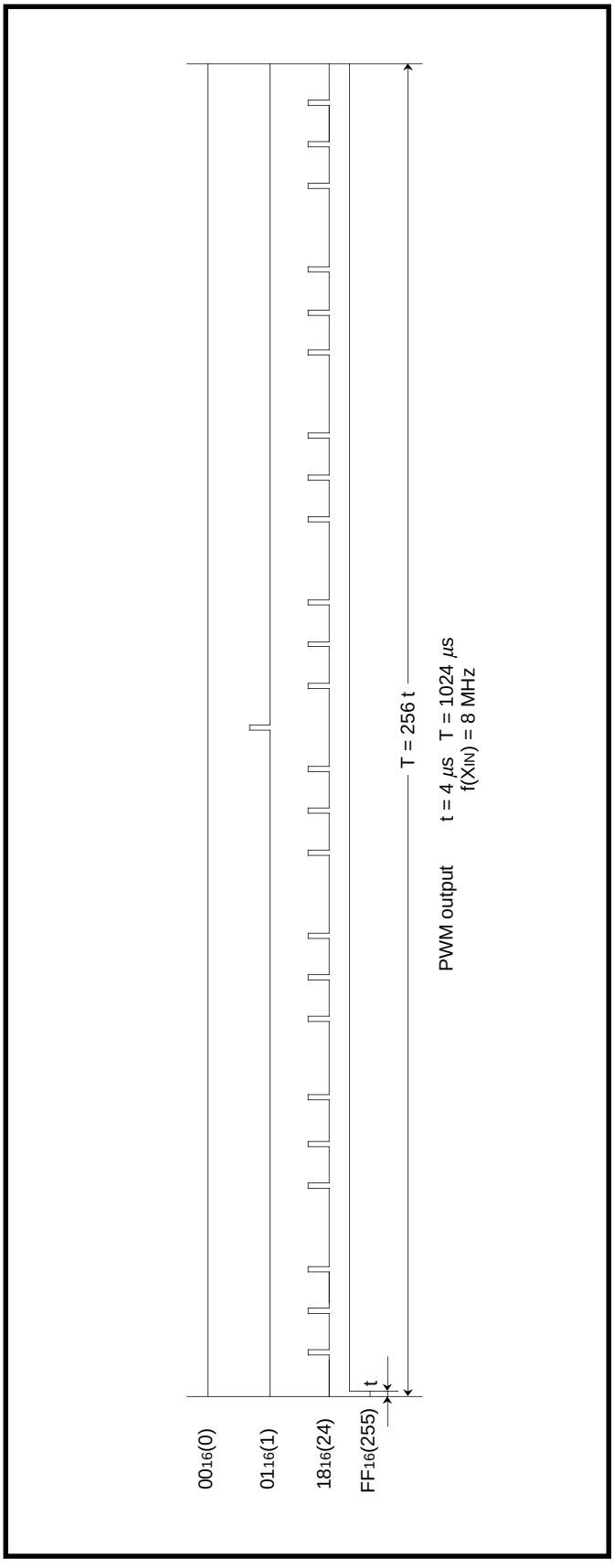


Fig. 2.10.5 Example of 8-bit PWM output

2.10.4 14-bit PWM output control

How to control the 14-bit PWM output is described below.

① Set "0" to bit 0 of PWM output control register 1 (address $00D5_{16}$) to supply the PWM count source (this bit is cleared to "0" when reset).

② Set the high-order 8 bits of the output data to the DA-H register.

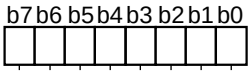
③ Set the low-order 6 bits of the output data to the DA-L register.

④ Data is written to the 14-bit PWM circuit by writing data to the DA-L register.
For this reason, even when changing only the high-order 8 bits of the output data, be sure to write the low-order 6 bits data to the DA-L register again.
Conversely, when changing low-order 6 bits only, it needs to only write data to the DA-L register, and needs not write the high-order 8-bit data again.

⑤ Select the output polarity by bit 2 of PWM output control register 2 (address $00D6_{16}$). When setting to "0," a positive polarity is selected; when "1," a negative polarity is selected.

⑥ 14-bit PWM is output from the D-A pin by clearing bit 1 of PWM output control register 1 to "0." When setting to "1," pin D-A functions as a 1-bit general-purpose output port. In this case, it is possible to specify either HIGH output (= "1") or LOW output(= "0") output by bit 4 of PWM output control register 2.

PWM Output Control Register 1



PWM output control register 1 (PW) [Address $00D5_{16}$]

B	Name	Functions	After reset	R	W
0	DA, PWM count source selection bit (PW0)	0 : Count source supply 1 : Count source stop	0	R	W
1	DA/PN4 output selection bit (PW1)	0 : DA output 1 : PN4 output	0	R	W
2	$P0_0$/PWM0 output selection bit (PW2)	0: $P0_0$ output 1: PWM0 output	0	R	W
3	$P0_1$/PWM1 output selection bit (PW3)	0: $P0_1$ output 1: PWM1 output	0	R	W
4	$P0_2$/PWM2 output selection bit (PW4)	0: $P0_2$ output 1: PWM2 output	0	R	W
5	$P0_3$/PWM3 output selection bit (PW5)	0: $P0_3$ output 1: PWM3 output	0	R	W
6	$P0_4$/PWM4 output selection bit (PW6)	0: $P0_4$ output 1: PWM4 output	0	R	W
7	$P0_5$/PWM5 output selection bit (PW7)	0: $P0_5$ output 1: PWM5 output	0	R	W

Fig. 2.10.6 PWM output control register 1 (address $00D5_{16}$)

2.10 PWM

2.10.5 8-bit PWM output control

How to control the 8-bit PWM output is described below.

The PWM0–PWM7 output pins are also used for port $P0_0$–$P0_3$ and $P6_0$–$P6_3$.

① Set "0" to bit 0 of the PWM output control register 1 (address $00D5_{16}$) to supply the PWM count source (this bit is cleared to "0" after reset).

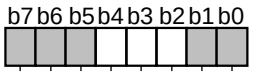
② Write output data to the corresponding 8-bit PWM registers (addresses $00D0_{16}$ to $00D4_{16}$ and $00F6_{16}$).

③ Set the corresponding bit of the port P0 direction register to "1" to specify the output mode.

④ Select the output polarity by bit 3 of the PWM output control register 2 (address $00D6_{16}$). When this bit is cleared to "0," a positive polarity is selected; when set to "1," a negative polarity is selected.

⑤ By setting "1" to the corresponding bits among bits 2 to 7 of the PWM output control register 1, the pins are given the PWM output function to output the PWM. When clearing to "0," the pins become general-purpose ports (ports $P0_0$–$P0_5$).

PWM Output Control Register 2



PWM output control register 2 (PN) [Address $00D6_{16}$]

B	Name	Functions	After reset	R	W
0, 1	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—
2	DA output polarity selection bit (PN3)	0 : Positive polarity 1 : Negative polarity	0	R	W
3	PWM output polarity selection bit (PN4)	0 : Positive polarity 1 : Negative polarity	0	R	W
4	DA general-purpose output bit (PN5)	0 : Output LOW 1 : Output HIGH	0	R	W
5 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 2.10.7 PWM output control register 2 (address $00D6_{16}$)

2.11 CRT display function

Table 2.11.1 shows the outline the CRT display function of the M37221M6-XXXSP/FP.

M37220M3-XXXSP/FP

Refer to **"CHAPTER 4. M37220M3-XXXSP/FP."**

The M37221M6-XXXSP/FP has the 24 characters × 2 lines CRT display circuit. CRT display is controlled by the CRT control register.

Up to 256 kinds of characters can be displayed, and colors can be specified for each character. Up to 4 kinds of colors can be displayed on 1 screen. A combination of up to 7 colors can be obtained by using each output signal (R, G, and B). Characters are displayed in a 12 × 16 dot structure to display smooth character patterns (refer to **"Figure 2.11.1"**).

How to display characters on the CRT screen is described below.

① Write the display character code in the display RAM.
② Specify the display color by the color register.
③ Write the color register in which the display color is set in the display RAM.
④ Specify the vertical position by the vertical position register.
⑤ Specify the character size by the character size register.
⑥ Specify the horizontal position by the horizontal position register.
⑦ Write the display control bit to the designated block display flag of the CRT control register. When this is done, the CRT starts according to the input of the V_{SYNC} signal.

The CRT display circuit has an extended display mode. This mode allows multi-line (more than 3 lines) to be displayed on the screen by interrupting each time 1 line is displayed and rewriting data in the block which display is terminated by software.

Figure 2.11.2 shows the CRT display circuit block diagram. Figure 2.11.3 shows the CRT control register.

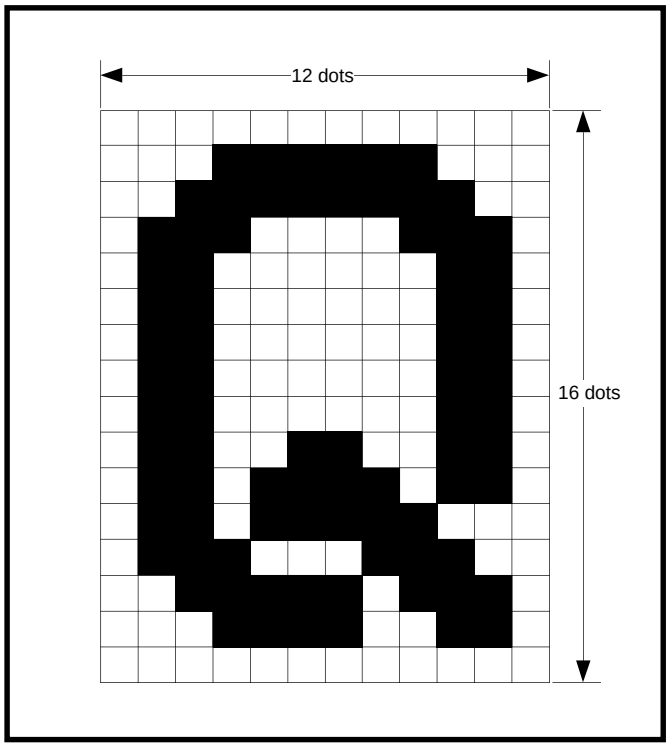


Fig. 2.11.1 Structure of CRT display character

Table 2.11.1 Outline of CRT display function

Parameter		Performance
Number of display character		24 characters × 2 lines
Dot structure		12 dots × 16 dots (Refer to **"Figure 2.11.1"**)
Kinds of character		256 kinds
Kinds of character sizes		3 kinds
Color	Kind of colors	1 screen; 4 kinds, maximum 7 kinds
	Coloring unit	A character
Display extension		Possible (multi-line display)
Raster coloring		Possible (maximum 7 kinds)
Character background coloring		Possible (a character unit, 1 screen; 4 kinds, maximum 7 kinds)

2.11 CRT display function

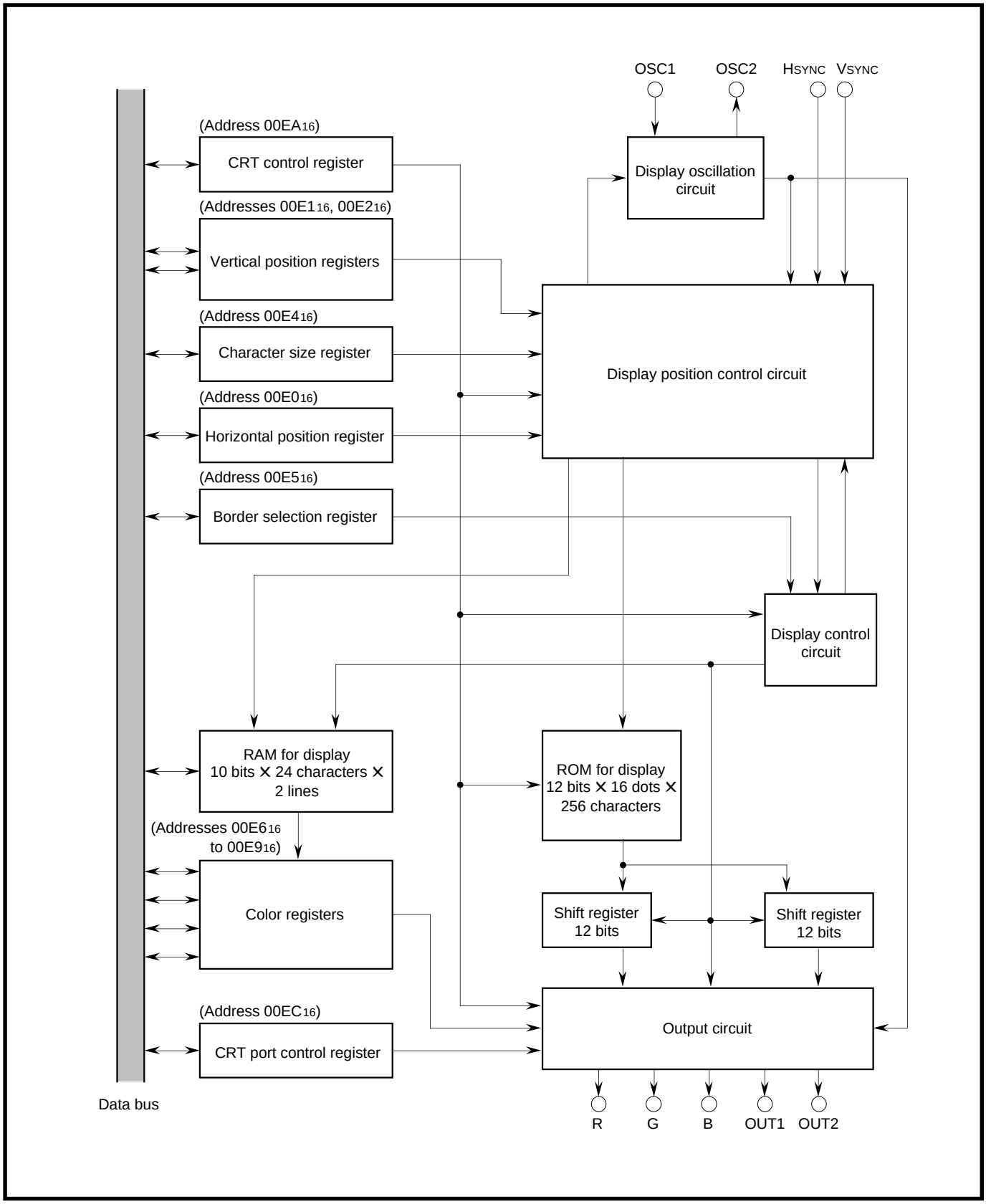
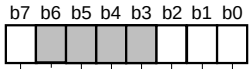


Fig. 2.11.2 CRT display circuit block diagram

CRT Control Register



CRT control register (CC) [Address $00EA_{16}$]

B	Name	Functions	After reset	R	W
0	All-blocks display control bit (Note) (CC0)	0 : All-blocks display off 1 : All-blocks display on	0	R	W
1	Block 1 display control bit (CC1)	0 : Block 1 display off 1 : Block 1 display on	0	R	W
2	Block 2 display control bit (CC2)	0 : Block 2 display off 1 : Block 2 display on	0	R	W
3 to 6	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—
7	$P1_0$/OUT2 pin switch bit (CC7)	0 : $P1_0$ 1 : OUT2	0	R	W

Note: Display is controlled by logical product (AND) between the all-blocks display control bit and each block control bit.

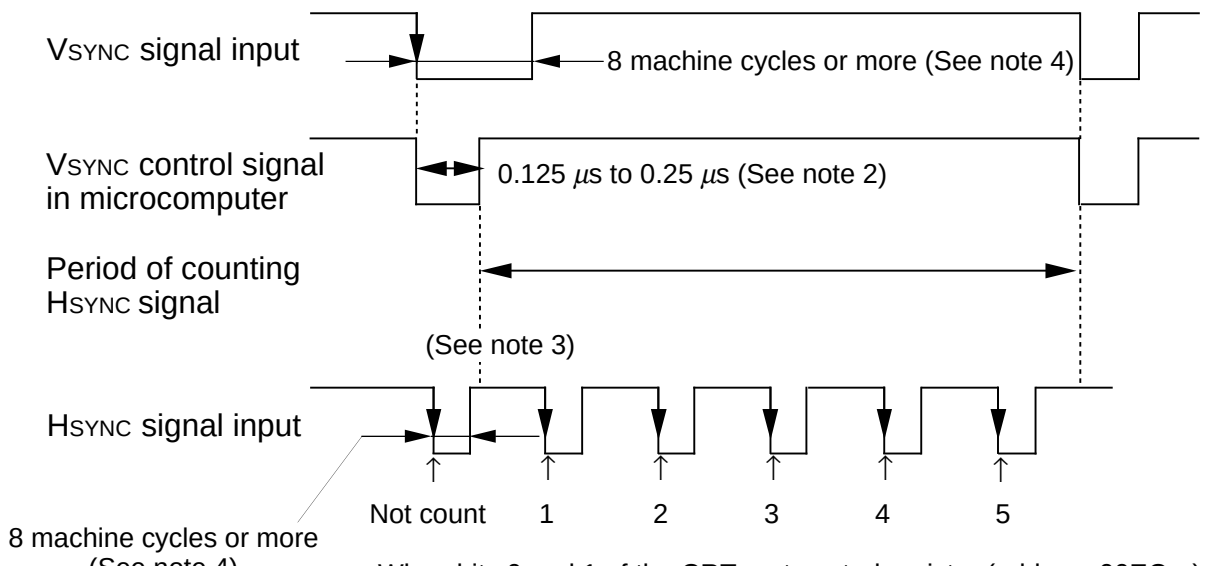
Fig. 2.11.3 CRT control register (address $00EA_{16}$)

2.11 CRT display function

2.11.1 Display position

The display positions of characters are specified in units called a “block”. There are 2 blocks, block 1 and block 2. Up to 24 characters can be displayed in 1 block (refer to **“2.11.3 Memory for display”**).
The display position of each block in both horizontal and vertical directions can be set by software.
The horizontal direction is common to all blocks, and is selected from 64-step display positions in units of 4 T_C (T_C = oscillation cycle for display).
The display position in the vertical direction is selected from 128-step display positions for each block in units of 4 scanning lines. The display position in the vertical direction is determined by counting the horizontal sync signal (H_{SYNC}).
At this time, it starts to count the rising edge (falling edge*) of H_{SYNC} signal from after about 1 machine cycle of rising edge (falling edge*) of V_{SYNC} signal. So interval from rising edge (falling edge*) of H_{SYNC} signal needs enough time (2 machine cycles or more) for avoiding jitter.

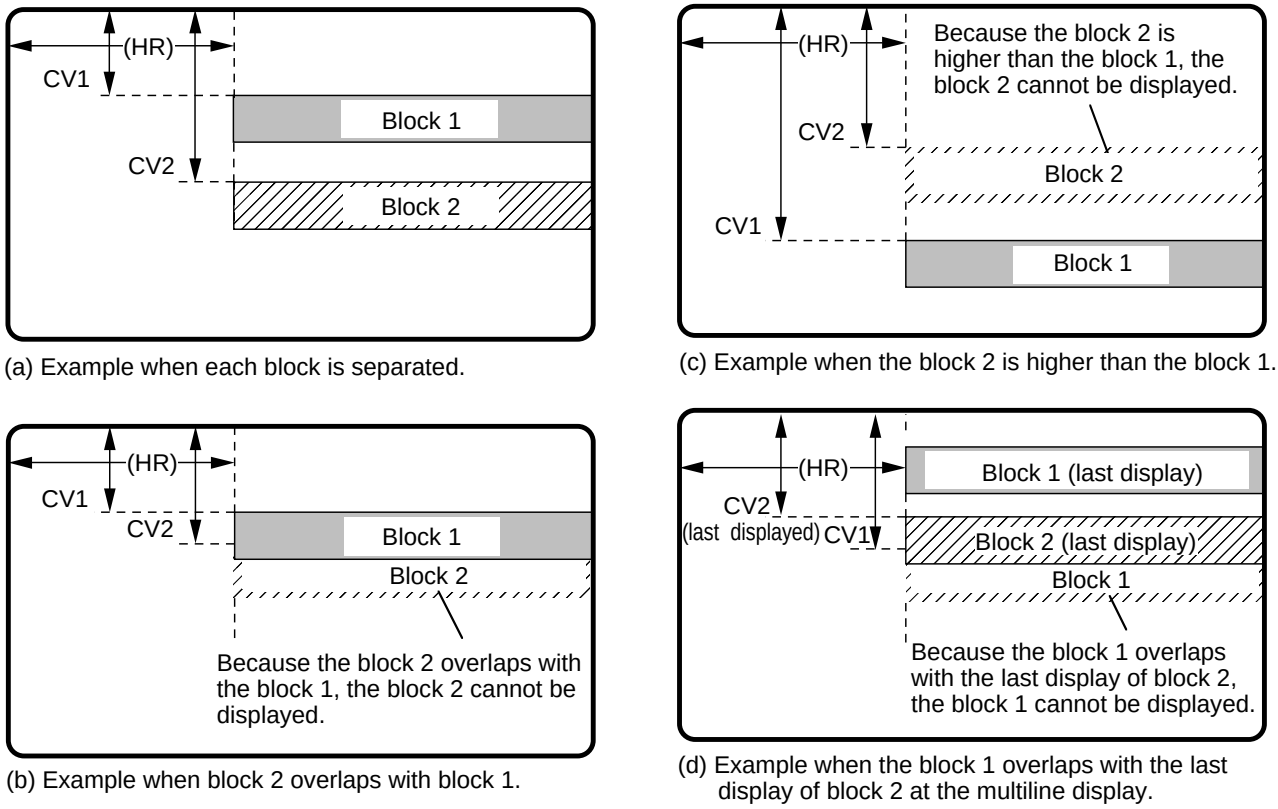
✻: The polarity of H_{SYNC} and V_{SYNC} signals can select by the CRT port control register (address $00EC_{16}$). When clearing corresponding bits to “0,” positive polarity is selected, when setting to “1,” negative polarity is selected. Refer to **“2.11.7 CRT output pin control”** for detail.



Note 1: The vertical position is determined by counting falling edge of H_{SYNC} signal after rising edge of V_{SYNC} control signal in the microcomputer.
2: At $f(X_{IN})$ = 8 MHz
3: Do not generate falling edge of H_{SYNC} signal near rising edge of V_{SYNC} control signal in microcomputer to avoid jitter.
4: Pulse width of V_{SYNC} and of H_{SYNC} signals needs 8 machine cycles or more.

Fig. 2.11.4 Count method of synchronous signal

The block 2 is displayed after the display of block 1 is completed (refer to **"Figure 2.11.5 (a)"**). Therefore, set vertical display start position of block 2 to be lower than the display end position of block 1. The block 2 cannot display when the display position of block 2 is overlapped with the display position of block 1 (refer to **"Figure 2.11.5 (b)"**) or is higher than the display position of block 1 (refer to **"Figure 2.11.5 (c)"**). Same as above, at the multiline display, the next block 1 cannot be displayed until the display of block 2 is completed. Therefore, set the display start position of the second and later block 1 to be lower than the display position of the last block 2 (refer to **"Figure 2.11.5 (d)"**).



(a) Example when each block is separated.

(b) Example when block 2 overlaps with block 1.

(c) Example when the block 2 is higher than the block 1.

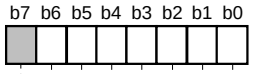
(d) Example when the block 1 overlaps with the last display of block 2 at the multiline display.

Fig. 2.11.5 Display position

2.11 CRT display function

The vertical position can specify 128-step positions (4 scanning lines per step) for each block by setting values "00_{16}" to "$7F_{16}$" to bits 0 to 6 of the vertical position registers (the blocks 1 and 2 are assigned to addresses to $00E1_{16}$, $00E2_{16}$ respectively). Figure 2.11.6 shows the vertical position registers.

Vertical Position Register n



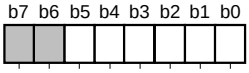
Vertical position register n (CV1,CV2) (n = 1 and 2) [Addresses $00E1_{16}$, $00E2_{16}$]

B	Name	Functions	After reset	R	W
0 to 6	Vertical display start positions (CV1 : CV10 to CV16) (CV2 : CV20 to CV26)	128 steps (00_{16} to $7F_{16}$)	Indeterminate	R	W
7	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—

Fig. 2.11.6 Vertical position register n (addresses $00E1_{16}$ and $00E2_{16}$)

The horizontal direction is common to both blocks, and can specify 64-step display positions (4 T_C per step, T_C: oscillation cycle for display) by setting values "00_{16}" to "$3F_{16}$" to bits 0 to 5 of the horizontal position register (address $00E0_{16}$).
Figure 2.11.7 shows the horizontal position register.

Horizontal Position Register



Horizontal position register (HR) [Address $00E0_{16}$]

B	Name	Functions	After reset	R	W
0 to 5	Horizontal display start positions (HR0 to HR5)	64 steps (00_{16} to $3F_{16}$)	0	R	W
6, 7	Nothing is assigned. These bits are write disable bits. When thses bits are read out, the values are "0."		0	R	—

Fig. 2.11.7 Horizontal position register (address $00E0_{16}$)

2.11.2 Character size

The size of characters to be displayed can select from 3 sizes for each block. Set a character size by the character size register (address $00E4_{16}$).

The character size in block 1 can be specified by bits 0 and 1 of the character size register; the character size in block 2 can be specified by bits 2 and 3. Figure 2.11.8 shows the character size register.

Character Size Register

b7 b6 b5 b4 b3 b2 b1 b0

Character size register (CS) [Address $00E4_{16}$]

B	Name	Functions	After reset	R	W
0, 1	Character size of block 1 selection bits (CS10, CS11)	00 : Minimum size 01 : Medium size 10 : Large size 11 : Do not set.	Indeterminate	R	W
2,3	Character size of block 2 selection bits (CS20,CS21)	00 : Minimum size 01 : Medium size 10 : Large size 11 : Do not set.	Indeterminate	R	W
4 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 2.11.8 Character size register (address $00E4_{16}$)

The character size can select three sizes: minimum size, medium size, and large size. Each character size is determined with the number of scanning lines in the height (vertical) direction and the oscillation cycle for display (= T_C) in the width (horizontal) direction.

The minimum size consists of [1 scanning line] × [1 T_C]; the medium size consists of [2 scanning lines] × [2 T_C]; and the large size consists of [3 scanning lines] × [3 T_C]. Table 2.11.2 shows the relationship between the set values in the character size register and the character sizes.

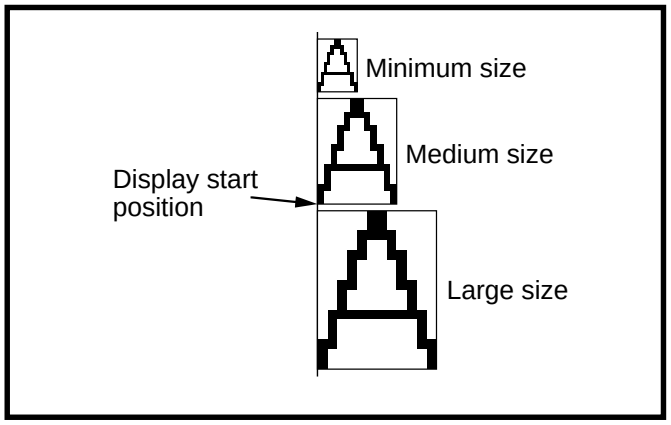


Fig. 2.11.9 Display start position (horizontal direction) for each character size

Table 2.11.2 Relationship between set value in character size register and character sizes

Set values in character size register		Character size	Width (horizontal) direction	Height (vertical) direction
CSn1	CSn0		T_C: oscillation cycle for display	scanning lines
0	0	Minimum	1 T_C	1 line
0	1	Medium	2 T_C	2 lines
1	0	Large	3 T_C	3 lines
1	1	This is not available.		

Note: The display start position in the horizontal direction is not affected by the character size. In other words, the horizontal display start position is common to all blocks even when the character size varies with each block (refer to **"Figure 2.11.9"**).

2.11.3 Memory for display

There are 2 types of display memory: CRT display ROM (addresses 10000_{16} to $11FFF_{16}$) used to store (masked) character dot data and CRT display RAM (addresses 0600_{16} to $06B7_{16}$) used to specify the colors and characters to be displayed. Each type of display memory is described below.

(1) CRT display ROM (addresses 10000_{16} to $11FFF_{16}$)

CRT display ROM stores dot pattern data for characters to be displayed. When actually displaying characters stored in this ROM, it is necessary to specify them by writing the character code inherent to each character (code determined based on the addresses in CRT display ROM) into CRT display RAM.

CRT display ROM has a capacity of 8 K bytes. Since 32 bytes are required for 1 character data, the ROM can stores up to 256 kinds of characters.

CRT display ROM is broadly divided into 2 areas. The [vertical 16 dots] × [horizontal (left side) 8 dots] data of display characters are stored in addresses 10000_{16} to $107FF_{16}$ and 11000_{16} to $117FF_{16}$; the [vertical 16 dots] × [horizontal (right side) 4 dots] data of display characters are stored in addresses 10800_{16} to $10FFF_{16}$, 11800_{16} to $11FFF_{16}$ (refer to **"Figure 2.11.10"**). Note however that the high-order 4 bits of the data to be written to addresses 10800_{16} to $10FFF_{16}$ and 11800_{16} to $11FFF_{16}$ must be set to "1" (by writing data FX_{16}).

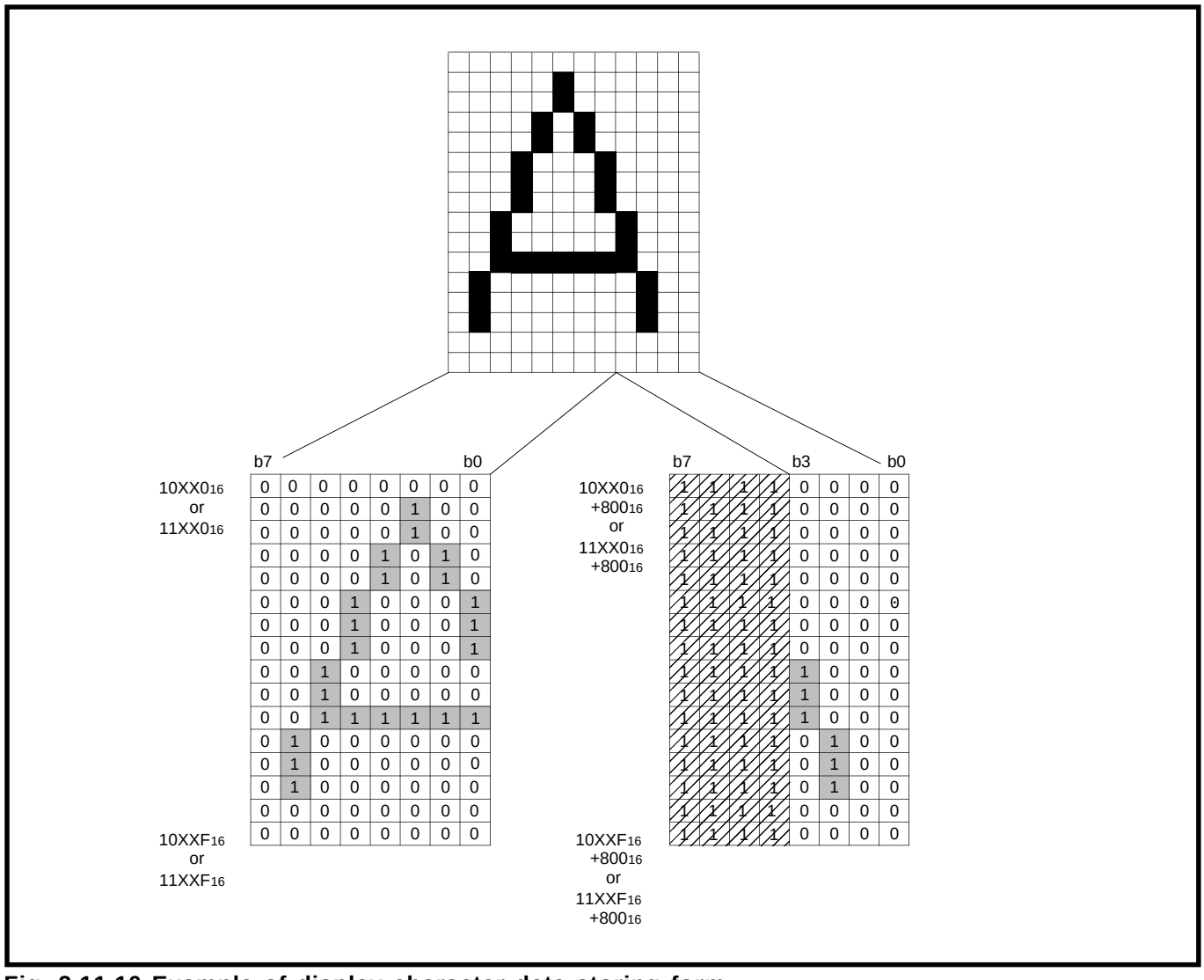


Fig. 2.11.10 Example of display character data storing form

The character code used to specify a display character is determined based on the address in the CRT display ROM in which that character data is stored.
Assume that 1 character data is stored in addresses $10XX0_{16}$ to $10XXF_{16}$ (XX denotes "00_{16}" to "$7F_{16}$") and $10YY0_{16}$ to $10YYF_{16}$ (YY denotes "XX + 800_{16}"), then the character code is "XX_{16}."
In other words, a character code is constructed with the low-order second and third digits (hexadecimal notation) of the 5-digit address (10000_{16} to $107FF_{16}$) where that character data is stored.
A character code is "YY_{16}" in addresses 11000_{16} to $11FFF_{16}$.
Table 2.11.3 shows the character code table.

Table 2.11.3 Character code table (be omitted partly)

Character code	Character data stored address	
	Left side 8 dots	Right 4 side 8 dots
00_{16}	10000_{16} to $1000F_{16}$	10800_{16} to $1080F_{16}$
01_{16}	10010_{16} to $1001F_{16}$	10810_{16} to $1081F_{16}$
02_{16}	10020_{16} to $1002F_{16}$	10820_{16} to $1082F_{16}$
03_{16}	10030_{16} to $1003F_{16}$	10830_{16} to $1083F_{16}$
:	:	:
$7E_{16}$	$107E0_{16}$ to $107EF_{16}$	$10FE0_{16}$ to $10FEF_{16}$
$7F_{16}$	$107F0_{16}$ to $107EF_{16}$	$10FF0_{16}$ to $10FFF_{16}$
80_{16}	11000_{16} to $1100F_{16}$	11800_{16} to $1180F_{16}$
81_{16}	11010_{16} to $1101F_{16}$	11810_{16} to $1181F_{16}$
:	:	:
FD_{16}	$117D0_{16}$ to $117DF_{16}$	$11FD0_{16}$ to $11FDF_{16}$
FE_{16}	$117E0_{16}$ to $117EF_{16}$	$11FE0_{16}$ to $11FEF_{16}$
FF_{16}	$117F0_{16}$ to $117EF_{16}$	$11FF0_{16}$ to $11FFF_{16}$

2.11 CRT display function

(2) CRT display RAM (addresses 0600_{16} to $06B7_{16}$)

CRT display RAM is assigned to addresses 0600_{16} to $06B7_{16}$, and is divided into a display character code specification part and display color specification part for each block. Table 2.11.4 shows the contents of CRT display RAM.

For example, to display a character at the first character position (leftmost) in block 1, it is necessary to write the character code in address 0600_{16} and the color register No. to the low-order 2 bits (bits 0 and 1) at address 0680_{16}. The color register No. to be written here is one of the 4 color registers in which display color is set in advance. For details on color registers, refer to **"2.11.4 Color registers."**

Table 2.11.4 Contents of CRT display RAM

Block number	Display position (from left side)	Character code specifying	Color specifying
Block 1	1st character	0600_{16}	0680_{16}
	2nd character	0601_{16}	0681_{16}
	3rd character	0602_{16}	0682_{16}
	:	:	:
	22nd character	0615_{16}	0695_{16}
	23rd character	0616_{16}	0696_{16}
	24th character	0617_{16}	0697_{16}
Not used		0618_{16} to $061F_{16}$	0698_{16} : $069F_{16}$
Block 2	1st character	0620_{16}	$06A0_{16}$
	2nd character	0621_{16}	$06A1_{16}$
	3rd character	0622_{16}	$06A2_{16}$
	:	:	:
	22nd character	0635_{16}	$06B5_{16}$
	23rd character	0636_{16}	$06B6_{16}$
	24th character	0637_{16}	$06B7_{16}$

Figure 2.11.11 shows the structure of CRT display RAM.

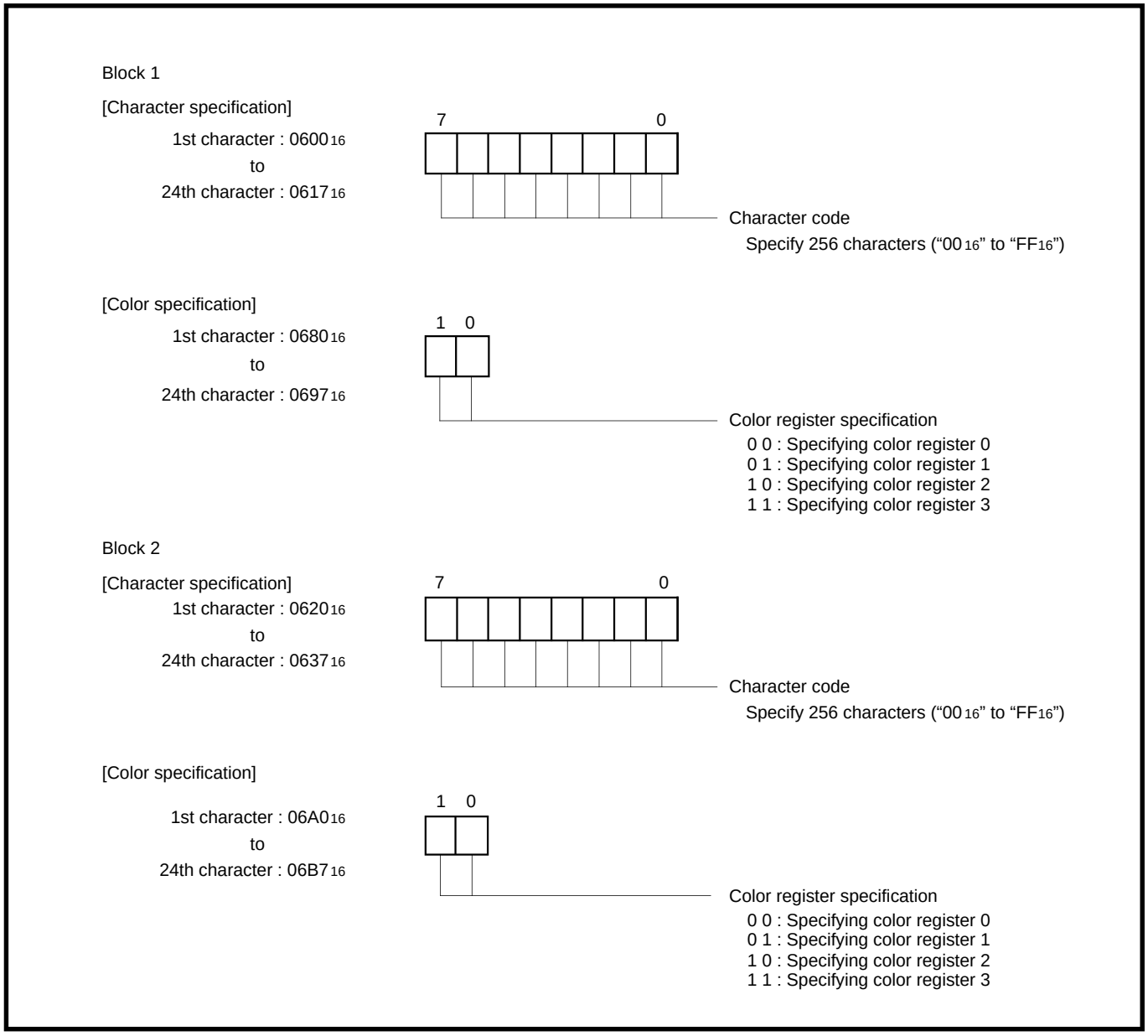


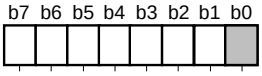
Fig. 2.11.11 Structure of CRT display RAM

2.11 CRT display function

2.11.4 Color registers

A display character color can be specified by setting a color to one of 4 color registers (CO0 to CO3: addresses $00E6_{16}$ to $00E9_{16}$) and then by specifying the color register with the CRT display RAM.
There are 3 color outputs: R, G, and B. By a combination of these outputs, it is possible to set $2^3 - 1$ (no output) = 7 colors. However, since color registers are only 4, up to 4 colors can be displayed at one time. R, G, and B outputs are set by bits 1 to 3 of the color register. Bit 5 is used to specify either a character output or blank output. Figure 2.11.12 shows the color register.
Either character output or blank output is selected as the OUT1 pin output. Whether blank output or not is selected as the OUT2 pin output.

Color Register n



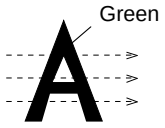
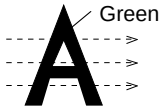
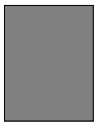
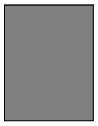
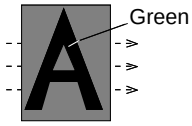
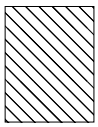
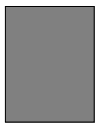
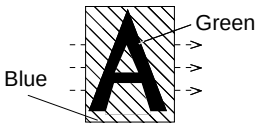
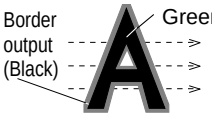
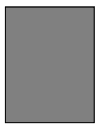
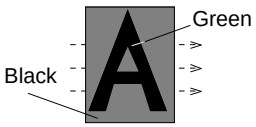
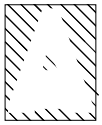
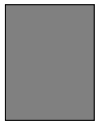
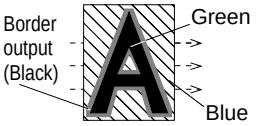
Color register n (CO0 to CO3) (n = 0 to 3) [Addresses $00E6_{16}$ to $00E9_{16}$]

B	Name	Functions	After reset	R	W
0	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
1	B signal output selection bit (COn1)	0 : No character is output 1 : Character is output	0	R	W
2	G signal output selection bit (COn2)	0 : No character is output 1 : Character is output	0	R	W
3	R signal output selection bit (COn3)	0 : No character is output 1 : Character is output	0	R	W
4	B signal output (background) selection bit (COn4)	0 : No background color is output 1 : Background color is output (See note 1)	0	R	W
5	OUT1 signal output control bit (COn5)	0 : Character is output 1 : Blank is output (See notes 1, 2)	0	R	W
6	G signal output (background) selection bit (COn6)	0 : No background color is output 1 : Background color is output	0	R	W
7	R signal output (background) selection bit (COn7)	0 : No background color is output 1 : Background color is output (See note 2)	0	R	W

Notes 1: When bit 5 = "0" and bit 4 = "1," there is output same as a character or border output from the OUT1 pin.
Do not set bit 5 = "0" and bit 4 = "0."
2: When only bit 7 = "1" and bit 5 = "0," there is output from the OUT2 pin.

Fig. 2.11.12 Color register n (addresses $00E6_{16}$ to $00E9_{16}$)

Table 2.11.5 Display example of character background coloring (when green is set for a character and blue is set for background color)

Border selection register / Color register								G output	B output	OUT1 output	Character output	OUT2 output
MD_0	COn_7	COn_6	COn_5	COn_4	COn_3	COn_2	COn_1					
0	0	×	0	1 (Note 1)	0	1	0		No output	 Same output as character A	 Video signal and character color (green) are not mixed.	No output (Note 2)
0	1	×	0	1	0	1	0		No output	 Same output as character A	 Video signal and character color (green) are not mixed.	 Blank output
0	0	0	1	0	0	1	0		No output	 Blank output	 TV image of character background is not displayed.	No output (Note 2)
0	0	0	1	1	0	1	0		 Background color	 Blank output	 TV image of character background is not displayed.	No output (Note 2)
1	×	×	0	1	0	1	0		No output	 Border output (Black)	 Video signal and character color (green) are not mixed.	No output (Note 2)
1	0	0	1	0	0	1	0		No output	 Blank output	 TV image of character background is not displayed.	No output (Note 2)
1	0	0	1	1	0	1	0		 Background color – border	 Blank output	 TV image of character background is not displayed.	No output (Note 2)

Notes 1 : When COn_5 = "0" and COn_4 = "1," there is output same as a character or border output from the OUT1 pin. Do not set COn_5 = "0" and COn_4 = "0."

2 : When only COn_7 = "1" and COn_5 = "0," there is output from the OUT2 pin.

3 : The portion "A" in which character dots are displayed is not mixed with any TV video signal.

4 : The wavy-lined arrows in the Table denote video signals.

5 : n : 0 to 3, × : 0 or 1

2.11 CRT display function

2.11.5 Multi-line display

The M37221M6-XXXSP/FP can ordinarily display 2 lines on the CRT screen by displaying 2 blocks at different vertical positions. In addition, it can display up to 16 lines by using a CRT interrupt.

A CRT interrupt request occurs at which display of each block has been completed. In other words, character display of a certain block starts when a scanning line reaches the display position (specified by vertical position registers) for that block, and an interrupt occurs when the scanning line exceeds the block. For multi-line display, it is necessary to enable the CRT interrupt (by clearing the interrupt disable flag to "0" and setting the CRT interrupt enable bit = bit 4 at address $00FE_{16}$ to "1").

In a CRT interrupt processing routine, the character data and vertical position of the block of which display has been completed (the display as CRT interrupt cause is completed) is then replaced with the character data (contents of CRT display RAM) and display position (contents of vertical position register) for next display.

Notes 1: Set the second and later block 1 display start positions of block 1 to be lower than display position of the last block 2.

2: The CRT interrupt request does not occur at the end of display when the block is not displayed. In other words, if a block is set to off display with the display control bit of the CRT control register (at address $00EA_{16}$), a CRT interrupt request does not occurs (refer to **"Figure 2.11.14"**).

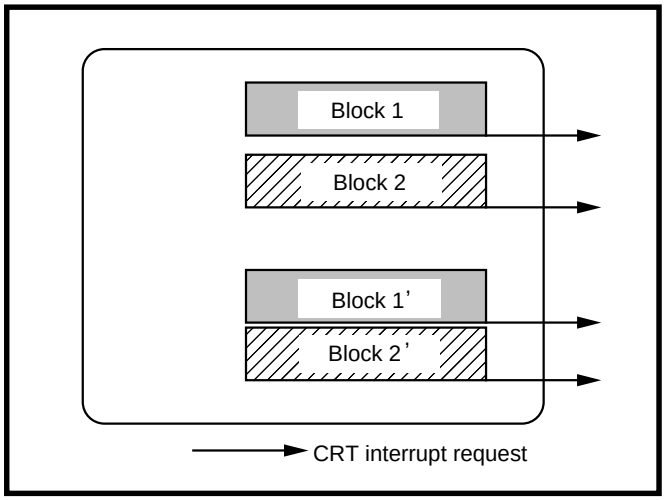


Fig. 2.11.13 Generation timing of CRT interrupt request

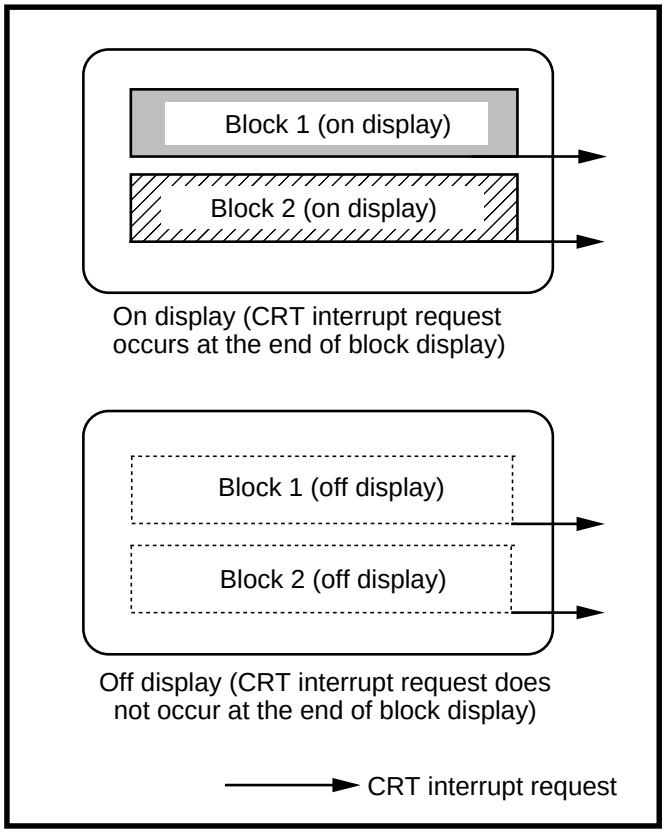


Fig. 2.11.14 Display state of blocks and occurrence of CRT interrupt request

2.11.6 Character border function

An border of 1 clock (1 dot) equivalent size can be added to a display character in both horizontal and vertical directions. The border is output from pin OUT 1. In this case, set bit 5 of a color register to "0" (character is output).

Border can be specified each block by the border selection register (address $00E5_{16}$). Table 2.11.6 shows the relationship between the set values of the border selection register and the character border function. Figure 2.11.16 shows the border selection register.

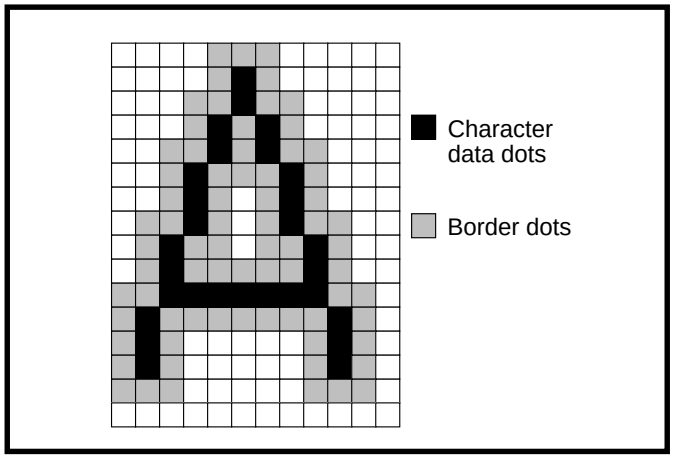
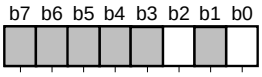


Fig. 2.11.15 Border example

Table 2.11.6 Relationship between set value of border selection register and character border function

Border selection register MDn0	Functions	Example of output
0	Ordinary	R, G, B output OUT1 output
1	Border including character	R, G, B output OUT1 output

Border Selection Register



Border selection register (MD) [Address $00E5_{16}$]

B	Name	Functions	After reset	R	W
0	Block 1 OUT1 output border selection bit (MD10)	0 : Same output as character output 1 : Border output	Indeterminate	R	W
1	Nothing is assigned. This bit is a write disable bits. When this bit is read out, the value is "0."		0	R	—
2	Block 1 OUT1 output border selection bit (MD20)	0 : Same output as character output 1 : Border output	Indeterminate	R	W
3 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 2.11.16 Border selection register (address $00E5_{16}$)

2.11 CRT display function

2.11.7 CRT output pin control

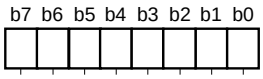
CRT display output pins R, G, B, and OUT1 are also used for ports $P5_2$–$P5_5$ respectively. When clearing the corresponding bits of the port P5 direction register (address $00CB_{16}$) to "0," the pins are set for CRT output pins, when setting to "1," the pins are set for general-purpose port P5. Pin PUT2 is also used for port $P1_0$. When clearing bit 7 of the CRT control register (address $00EA_{16}$) to "0," the pin is set for port $P1_0$, when setting to "1," the pin is set for pin OUT2.

Immediately after reset release, because the port P5 direction register is reset, they become CRT output pins R, G, B, and OUT.

Bits 0 to 4 of the CRT port control register (address $00EC_{16}$) can determine H_{SYNC} and V_{SYNC} input polarity and R, G, B, OUT1, and OUT2 output polarity. When clearing corresponding bits to "0," positive polarity is selected, when setting to "1," negative polarity is selected.

Figure 2.11.17 shows the CRT port control.

CRT Port Control Register



CRT port control register (CRTP) [Address $00EC_{16}$]

B	Name	Functions	After reset	R	W
0	H_{SYNC} input polarity switch bit (HSYC)	0 : Positive polarity 1 : Negative polarity	0	R	W
1	V_{SYNC} input polarity switch bit (VSYC)	0 : Positive polarity 1 : Negative polarity	0	R	W
2	R, G, B output polarity switch bit (R/G/B)	0 : Positive polarity 1 : Negative polarity	0	R	W
3	OUT2 output polarity switch bit (OUT2)	0 : Positive polarity 1 : Negative polarity	0	R	W
4	OUT1 output polarity switch bit (OUT1)	0 : Positive polarity 1 : Negative polarity	0	R	W
5	R signal output switch bit (OP5)	0 : R signal output 1 : MUTE signal output	0	R	W
6	G signal output switch bit (OP6)	0 : G signal output 1 : MUTE signal output	0	R	W
7	B signal output switch bit (OP7)	0 : B signal output 1 : MUTE signal output	0	R	W

Fig. 2.11.17 CRT port control register (address $00EC_{16}$)

2.11.8 Raster coloring function

R, G, B, and OUT1 output can be switched to MUTE output. MUTE output can color all displaying area (raster) of screen.

For example, the case that pin B is specified for MUTE signal output is shown in Figure 2.11.18.

When the MUTE signal is output from pin B, the background of the entire screen is colored “BLUE.” Then, a character data is output from pin R, for example. When B and R signal outputs are set to “character is output” by the color register at the character “I” output, the output character is colored “YELLOW” (“RED” mixed “BLUE”) regardless of the OUT1 signal output.

When outputting the character “O,” the output character is colored only “RED” that is not mixed “BLUE” by setting only R signal output to “character is output.” However, in this case, set pin OUT1 to “blank is output.”

The TV image can be also erase by setting the all R, G, and B pins to MUTE output. The MUTE signal is output from pin OUT1 output, regardless of setting CRT display RAM for pin OUT1.

Whether ordinary video signal outputs or MUTE signal outputs from pins R, G, and B is controlled by bits 5 to 7 of the CRT port control register (refer to “**Figure 2.11.17**”).

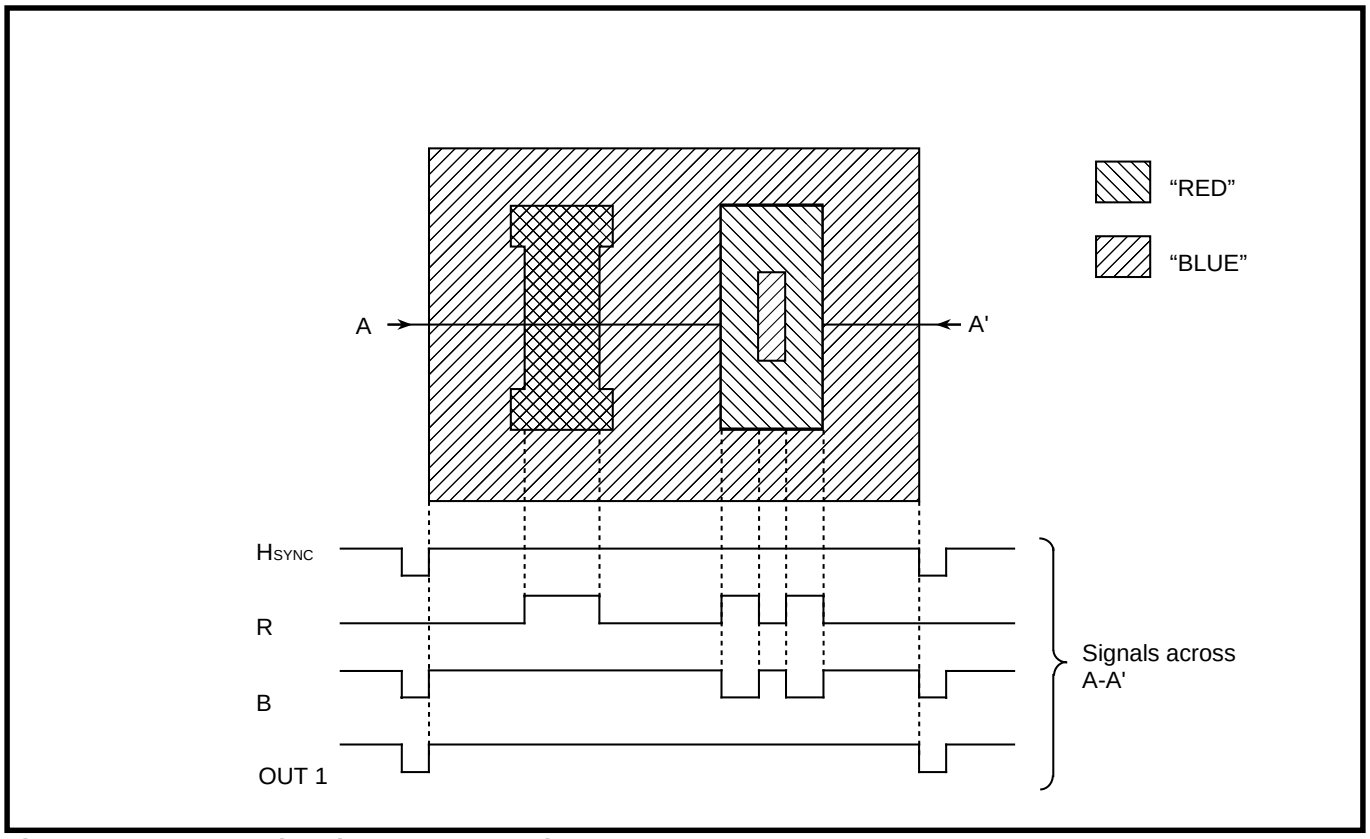


Fig. 2.11.18 MUTE signal output example

2.11 CRT display function

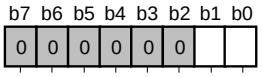
2.11.9 Clock for display

As a clock for display to be used for CRT display, it is possible to select one of the following 4 types.

- Main clock supplied from the X_{IN} pin
- Main clock supplied from the X_{IN} pin divided by 1.5
- Clock from the LC or RC supplied from the pins OSC1 and OSC2.
- Clock from the ceramic resonator or quartz-crystal oscillator supplied from the pins OSC1 and OSC2.

This clock for display can be selected by the CRT clock selection register (address $00ED_{16}$). When selecting the main clock, set the oscillation frequency to 8 MHz.

CRT Clock Selection Register



CRT clock selection register (CK) [Address $00ED_{16}$]

B	Name	Functions			After reset	R	W
0, 1	CRT clock selection bits (CK0,CK1)	b1	b0	Functions	0	R	W
		1	0	The clock for display is supplied by connecting RC or LC across the pins OSC1 and OSC2.			
		0	1	Since the main clock is used as the clock for display, the oscillation frequency is limited. Because of this, the character size in width (horizontal) direction is also limited. In this case, pins OSC1 and OSC2 are also used as input ports $P3_3$ and $P3_4$ respectively. CRT oscillation frequency = $f(X_{IN})$			
		1	0	CRT oscillation frequency = $f(X_{IN})/1.5$			
		1	1	The clock for display is supplied by connecting the following across the pins OSC1 and OSC2. • a ceramic resonator only for CRT display and a feedback resistor • a quartz-crystal oscillator only for CRT display and a feedback resistor (Note)			
2 to 7	Fix these bits to "0."				0	R	W

Note: It is necessary to connect other ceramic resonator or quartz-crystal oscillator across the pins X_{IN} and X_{OUT}.

Fig. 2.11.19 CRT clock selection register

2.12 ROM correction function

Only the M37221M8-XXXSP and the M37221MA-XXXSP have this function.

This can correct ROM program data in ROM. Up to 2 addresses (2 blocks) can be corrected, a program for correction is stored in the ROM correction memory in RAM. The ROM memory for correction is 32 bytes × 2 blocks.

Block 1 : addresses $02C0_{16}$ to $02DF_{16}$

Block 2 : addresses $02E0_{16}$ to $02FF_{16}$

Set an address of the ROM data to be corrected into the ROM correction address register. When the value of the counter matches the ROM data address in the ROM correction address, the main program branches to the correction program stored in the ROM correction memory. To return from the correction program to the main program, the op code and operand of the **JMP** instruction (total of 3 bytes) are necessary at the end of the correction program. When the blocks 1 and 2 are used in series, the above instruction is not needed at the end of the block 1.

The ROM correction function is controlled by the ROM correction enable register.

Notes 1 : Specify the first address (op code address) of each instruction as the ROM correction address.

2 : Use the **JMP** instruction (total of 3 bytes) to return from the correction program to the main program.

3 : Do not set the same address to ROM correction addresses 1 and 2 (addresses to 0217_{16} to $021A_{16}$).

ROM correction address 1 (high-order)	0217_{16}
ROM correction address 1 (low-order)	0218_{16}
ROM correction address 2 (high-order)	0219_{16}
ROM correction address 2 (low-order)	$021A_{16}$

Fig. 2.12.1 ROM correction address registers

ROM Correction Enable Register

b7	b6	b5	b4	b3	b2	b1	b0
				0	0		

ROM correction enable register (RCR) [Address 0212_{16}]

B	Name	Functions	After reset	R	W
0	Block 1 enable bit (RCR0)	0: Disabled 1: Enabled	0	R	W
1	Block 2 enable bit (RCR1)	0: Disabled 1: Enabled	0	R	W
2, 3	Fix these bits to "0."		0	R	—
4 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 2.12.2 ROM correction enable register

2.13 Software runaway detect function

The M37221M6-XXXSP/FP has a function to decode undefined instructions to detect a software runaway. When an undefined op-code is input to the CPU as an instruction code during operation of the M37221M6-XXXSP/FP, the following processing is done.

① The CPU generates an undefined instruction decoding signal.
② The device is internally reset because of occurrence of the undefined instruction decoding signal.
③ As a result of internal reset, the same reset processing as in the case of ordinary reset operation is done, and the program restarts from the reset vector.

Note, however, that the software runaway detecting function cannot be invalid.

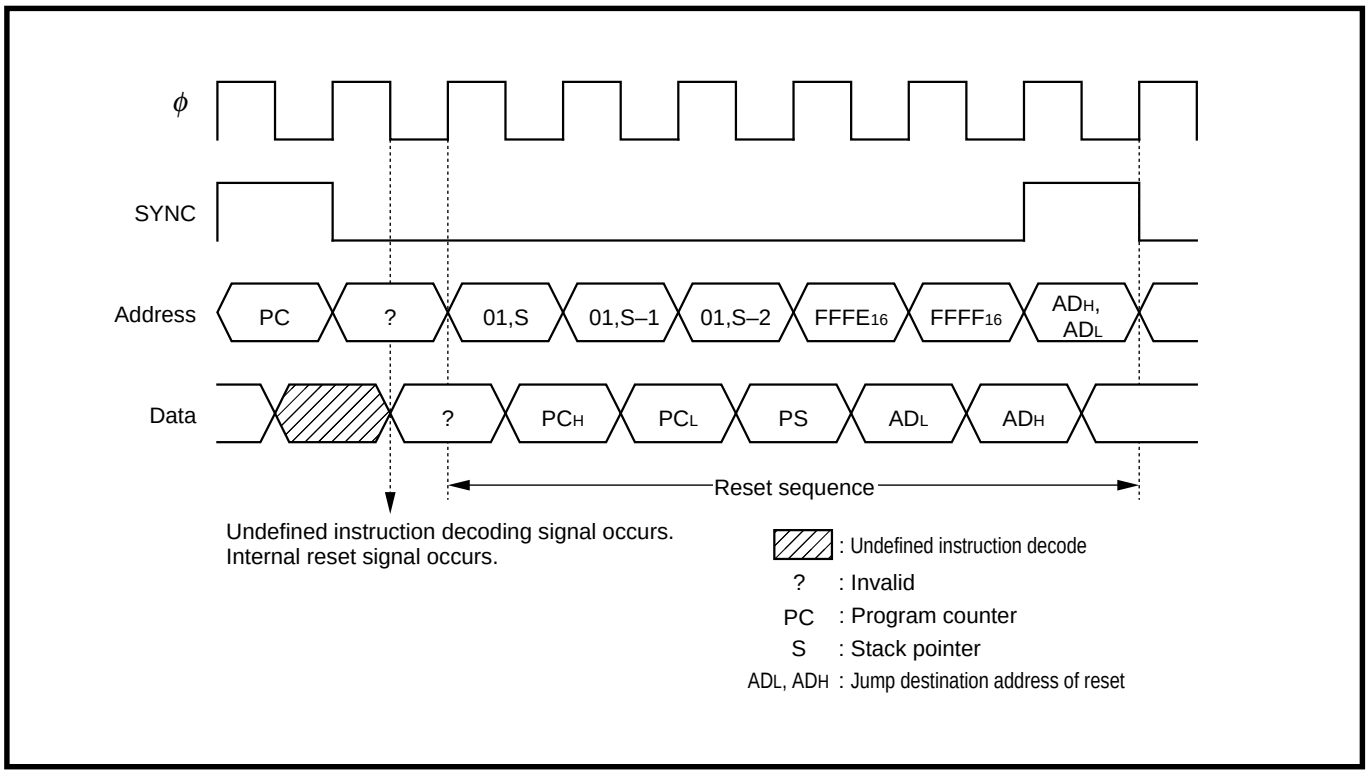


Fig. 2.13.1 Sequence at detecting software runaway detection

2.14 Low-power dissipation mode

The M37221M6-XXXSP/FP has 2 low-power dissipation modes: the stop mode and the wait mode.

2.14.1 Stop mode

The M37221M6-XXXSP/FP allows the oscillation of X_{IN} to be stopped with keeping all states of registers except timers 3 and 4, input/output ports, and internal RAM. Therefore, the M37221M6-XXXSP/FP can be restarted with the same state where oscillation was stopped, and as a result, the power dissipation can be greatly reduced.

To stop oscillating in such a way, execute the **STP** instruction. The stop mode is set by executing the **STP** instruction. In this mode, the address to fetch the instruction next to the **STP** instruction is output to the address bus, and the oscillation stops with HIGH state of the internal clock ϕ. At this time, the timer 3 overflow signal is further connected to timer 4. Value "FF_{16}" is automatically set to timer 3; value "07_{16}" is automatically set to timer 4.

Immediately before executing the **STP** instruction, process the following sequence:

① Store registers (accumulator, index registers, etc.) in the CPU to internal RAM.
② Disable timers 3 and 4 interrupts (TM3E = TM4E = "0").
③ Clear timers 3 and 4 count stop bits to "0" (T34M2 = T34M3 = "0").
④ When an interrupt is used for return from the stop mode, enable that interrupt (by clearing the interrupt disable flag to "0" and setting the interrupt enable bit to "1").
⑤ Set bit 0 of the timer 34 mode register (address $00F5_{16}$) to "0" (TM34M0="0") to select $f(X_{IN})/16$ as the timer 3 count source.

Oscillation is restarted (return from the stop mode) by accepting reset input or interrupt request of INT1, INT2 or INT3. When the interrupt request is accepted, the interrupt processing routine is executed. Note, however, that the internal clock ϕ is not supplied to the CPU until timer 4 overflows after the interrupt request is accepted. This is because a finite time is required for stabilizing of oscillation when an external quartz-crystal oscillator, etc. is used.

When the internal clock ϕ is supplied to the CPU, the CPU executes the interrupt routine. At this time, the address for the first byte of the instruction next to the **STP** instruction is pushed to the stack as a return address. Also note that the timers 3 and 4 interrupt request bits are remained setting to "1." Therefore, clear each bit to "0" in the interrupt routine. Enable one of the INT1, INT2 and INT3 interrupts to use interrupts for restarting oscillation before the executing **STP** instruction (described in ④ above).

Table 2.14.1 State in stop mode

Item	State in stop mode
Oscillation	Stops
CPU	Stops
Internal clock ϕ	Stops at HIGH level
I/O ports	State where **STP** instruction is executed is held.
Timer, CRT display functions	Stops

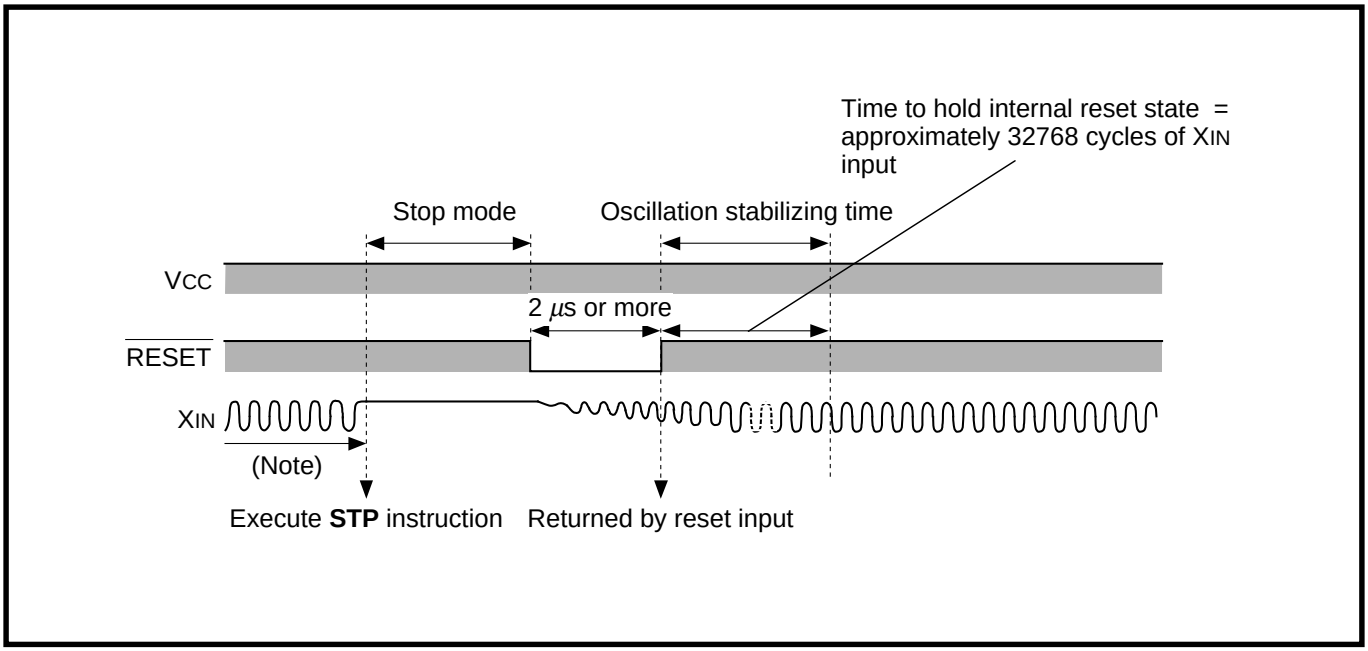


Fig. 2.14.1 Oscillation stabilizing time at return by reset input

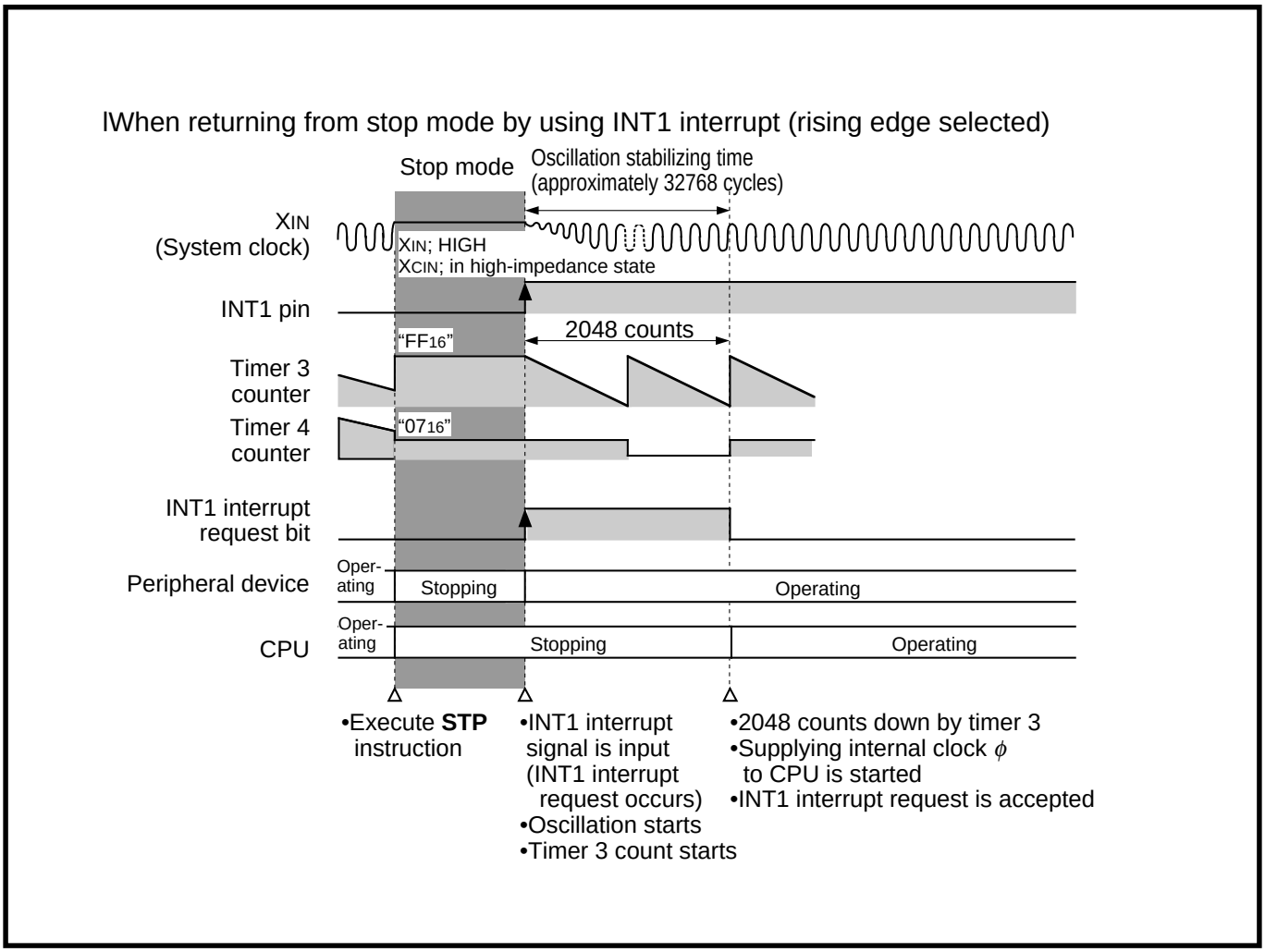


Fig. 2.14.2 Execution sequence example at return by occurrence of INT0 interrupt request

2.14.2 Wait mode

The wait mode is set by executing the **WIT** instruction.

In the wait mode, only the internal clock ϕ stops with supplying f(X_{IN}) continuously.

In this case, there is no need to create a wait time by timers as in the case of return from the stop mode, and operation is restarted immediately after return from the wait state.

When reset input or interrupt is accepted, supply of the internal clock ϕ is immediately started, and the device is returned from the wait state. Because the clock f(X_{IN}) is continuously supplied in the wait state, return by an internal interrupt as a timer, etc. can also be used.

Table 2.14.2 State in wait mode

Item	State in wait mode
Oscillation	Operating
CPU	Stop
Internal clock ϕ	Stop at HIGH level
I/O ports	State where **WIT** instruction is executed is held.
Timer, CRT display functions	Operating

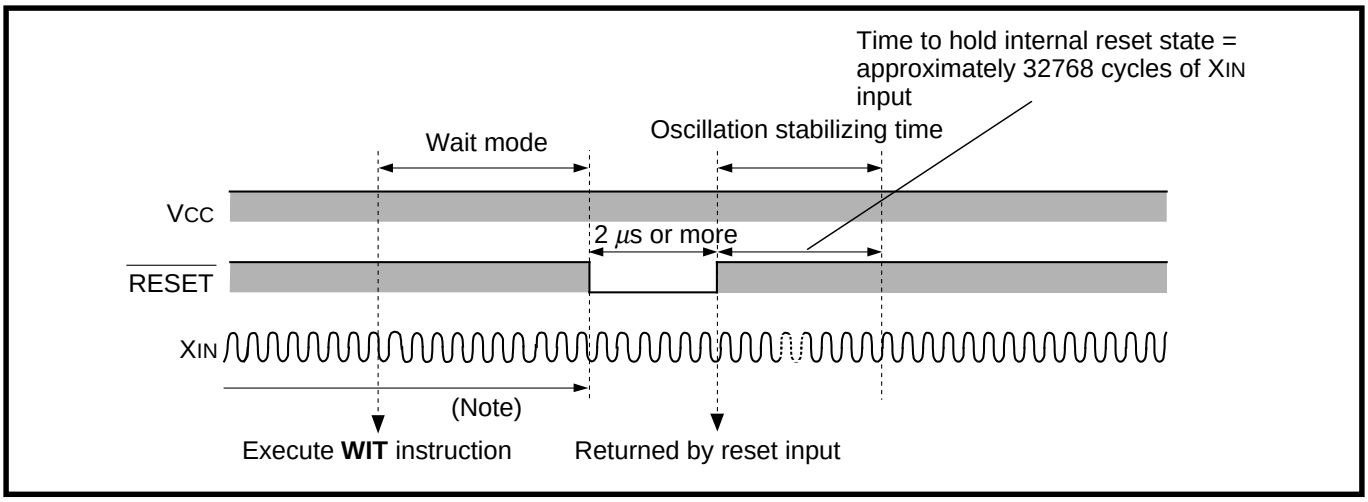


Fig. 2.14.3 Reset input time

2.14.3 Interrupts in low-power dissipation mode

The following 4 kinds of interrupts are invalid in the wait mode. Therefore, 4 interrupts below cannot be used to return from the wait mode to the ordinary mode.

Table 2.14.3 Invalid interrupts in the wait mode

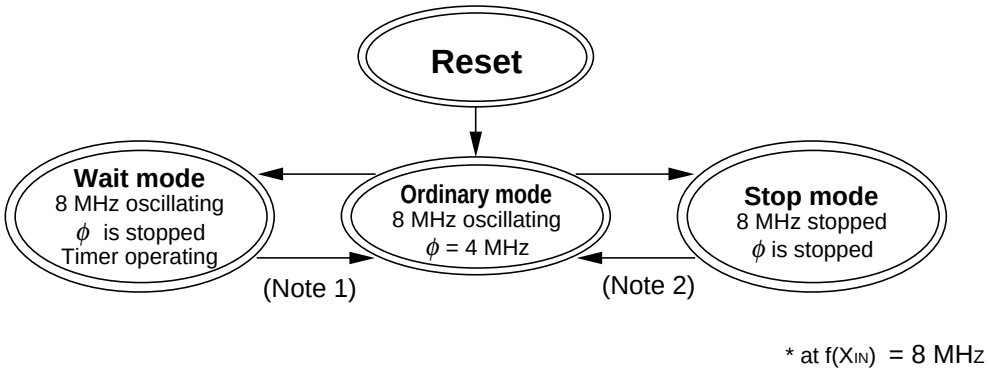
Interrupt source	Condition	Reason
V_{SYNC} interrupt	—	The interrupt request bit cannot be set.
CRT interrupt		
Timer 2 interrupt	Count source is input from pin $P2_4$/TIM2.	The count source cannot be supplied.
Timer 3 interrupt	Count source is input from pin $P2_3$/TIM3.	The count source cannot be supplied.

The following 2 kinds of interrupts can be used to return from the stop mode to the ordinary mode.

① INT1 interrupt
② INT2 interrupt
③ INT3 interrupt

Figure 2.14.4 shows a transitions of low-power dissipation mode.

2.14 Low-power dissipation mode



Notes 1: The following interrupts are invalid in the wait mode.
(1) VSYNC interrupt
(2) CRT interrupt
(3) Timer 2 interrupt that count source is supplied from pin P24/TIM2.
(4) Timer 3 interrupt that count source is supplied from pin P23/TIM3.

Fig. 2.14.4 State transitions of low-power dissipation mode

2.15 Reset

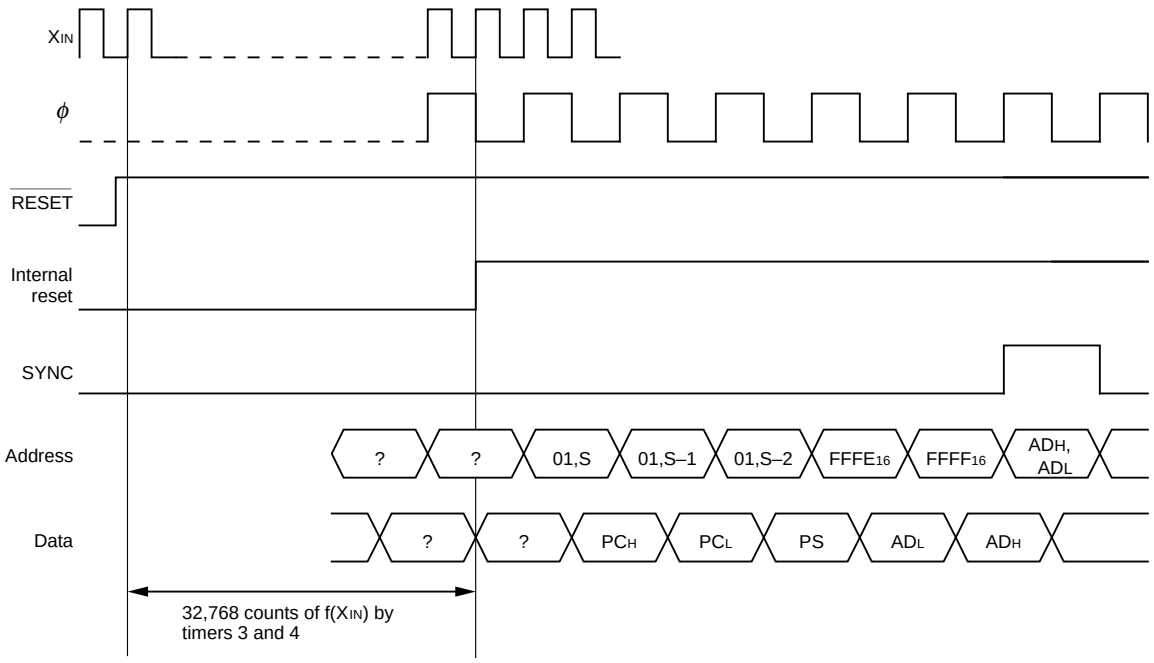
To reset the microcomputer, applied LOW level to pin $\overline{\text{RESET}}$ for 2 μs or more. Reset is released when HIGH level is applied to pin $\overline{\text{RESET}}$, and the program starts from the address indicated with the reset vector table.

2.15.1 Reset operation

If pin $\overline{\text{RESET}}$ is returned to an HIGH level after being held LOW for 2 μs or more when the power source voltage is within the recommended range (4.5 V to 5.5 V), timers 3 and 4 are connected by hardware with internally reset state (internal timing signal ϕ is not supplied).

At this time, "FF_{16}" is set to timer 3, and "07_{16}" is set to timer 4. Timer 3 counts down $f(X_{IN})/16$ as its count source; timer 4 counts down the timer 3 overflow signal (even when the device is in internally reset state, $f(X_{IN})$ is continuously supplied to timer 3).

The internal reset is released by timer 4 overflow, and the program is started from an address determined with the contents of address $FFFF_{16}$ (as high-order address) and contents of address $FFFE_{16}$ (as low-order address). Figure 2.15.1 shows this sequence.



Notes 1: $f(X_{IN})$ and $f(\phi)$ are in the relation : $f(X_{IN}) = 2 \cdot f(\phi)$.
2: A question mark (?) indicates an undefined state that depends on the previous state.
3: Immediately after a reset, timer 3 and timer 4 are connected by hardware. At this time, "FF_{16}" is set in timer 3 and "07_{16}" is set to timer 4. Timer 3 counts down with $f(X_{IN})/16$, and reset state is released by the timer 4 overflow signal.

Fig. 2.15.1 Timing diagram at reset

2.15 Reset

2.15.2 Internal state immediately after reset

Figures 2.15.2 to 2.15.4 show the internal state immediately after reset.

■SFR Area (addresses $C0_{16}$ to DF_{16})

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately after reset

Address	Register	State immediately after reset (b7 b6 b5 b4 b3 b2 b1 b0)
$C0_{16}$	Port P0 (P0)	?
$C1_{16}$	Port P0 direction register (D0)	00_{16}
$C2_{16}$	Port P1 (P1)	?
$C3_{16}$	Port P1 direction register (D1)	00_{16}
$C4_{16}$	Port P2 (P2)	?
$C5_{16}$	Port P2 direction register (D2)	00_{16}
$C6_{16}$	Port P3 (P3)	0 0 0 ? ? ? ? ?
$C7_{16}$	Port P3 direction register (D3)	00_{16}
$C8_{16}$		?
$C9_{16}$		?
CA_{16}	Port P5 (P5)	0 0 ? ? ? ? ? ?
CB_{16}	Port P5 direction register (D5)	00_{16}
CC_{16}		?
CD_{16}	Port P3 output mode control register (P3S)	00_{16}
CE_{16}	DA-H register (DA-H)	?
CF_{16}	DA-L register (DA-L)	0 0 ? ? ? ? ? ?
$D0_{16}$	PWM0 register (PWM0)	?
$D1_{16}$	PWM1 register (PWM1)	?
$D2_{16}$	PWM2 register (PWM2)	?
$D3_{16}$	PWM3 register (PWM3)	?
$D4_{16}$	PWM4 register (PWM4)	?
$D5_{16}$	PWM output control register 1 (PW)	00_{16}
$D6_{16}$	PWM output control register 2 (PN)	00_{16}
$D7_{16}$	I^2C data shift register (S0)	?
$D8_{16}$	I^2C address register (S0D)	00_{16}
$D9_{16}$	I^2C status register (S1)	0 0 0 1 0 0 0 ?
DA_{16}	I^2C control register (S1D)	00_{16}
DB_{16}	I^2C clock control register (S2)	00_{16}
DC_{16}	Serial I/O mode register (SM)	00_{16}
DD_{16}	Serial I/O register (SIO)	?
DE_{16}		00_{16}
DF_{16}		00_{16}

Fig. 2.15.2 Internal state immediately after reset (1)

■SFR Area (addresses $E0_{16}$ to FF_{16})

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately after reset

Address	Register	State immediately after reset (b7 b6 b5 b4 b3 b2 b1 b0)
$E0_{16}$	Horizontal position register (HR)	00_{16}
$E1_{16}$	Vertical position register 1 (CV1)	0 ? ? ? ? ? ? ?
$E2_{16}$	Vertical position register 2 (CV2)	0 ? ? ? ? ? ? ?
$E3_{16}$		?
$E4_{16}$	Character size register (CS)	0 0 0 0 ? ? ? ?
$E5_{16}$	Border selection register (MD)	0 0 0 0 0 ? 0 ?
$E6_{16}$	Color register 0 (CO0)	00_{16}
$E7_{16}$	Color register 1 (CO1)	00_{16}
$E8_{16}$	Color register 2 (CO2)	00_{16}
$E9_{16}$	Color register 3 (CO3)	00_{16}
EA_{16}	CRT control register (CC)	00_{16}
EB_{16}		?
EC_{16}	CRT port control register (CRTP)	00_{16}
ED_{16}	CRT clock selection register (CK)	00_{16}
EE_{16}	A-D control register 1 (AD1)	0 0 0 ? 0 0 0 0
EF_{16}	A-D control register 2 (AD2)	00_{16}
$F0_{16}$	Timer 1 (TM1)	FF_{16}
$F1_{16}$	Timer 2 (TM2)	07_{16}
$F2_{16}$	Timer 3 (TM3)	FF_{16}
$F3_{16}$	Timer 4 (TM4)	07_{16}
$F4_{16}$	Timer 12 mode register (T12M)	00_{16}
$F5_{16}$	Timer 34 mode register (T34M)	00_{16}
$F6_{16}$	PWM5 register (PWM5)	?
$F7_{16}$		?
$F8_{16}$		?
$F9_{16}$	Interrupt input polarity register (RE)	0 0 0 0 0 0 0 ?
FA_{16}		00_{16}
FB_{16}	CPU mode register (CPUM)	? ? 1 1 1 1 0 0
FC_{16}	Interrupt request register 1 (IREQ1)	00_{16}
FD_{16}	Interrupt request register 2 (IREQ2)	00_{16}
FE_{16}	Interrupt control register 1 (ICON1)	00_{16}
FF_{16}	Interrupt control register 2 (ICON2)	00_{16}

Fig. 2.15.3 Internal state immediately after reset (2)

■2 Page Register Area (addresses 217_{16} to $21B_{16}$)

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately after reset

Address	Register	State immediately after reset (b7 b0)
217_{16}	ROM correction address 1 (high-order)	?
218_{16}	ROM correction address 1 (low-order)	?
219_{16}	ROM correction address 2 (high-order)	?
$21A_{16}$	ROM correction address 2 (low-order)	?
$21B_{16}$	ROM correction enable register (RCR)	00_{16}

Fig. 2.15.4 Internal state immediately after reset (3) (only M37221M8-XXXSP and M37221MA-XXXSP)

2.15.3 Notes for poweron reset

When poweron reset, set the external reset circuit so that the reset input voltage must be kept 0.6 V or less until the power source voltage reaches 4.5 V after the power is turned on.

Set the external reset circuit so that the reset input voltage must be kept 0.6 V or less when the power source voltage falls 4.5 V after the power is turned off.

Figures 2.15.5 to 2.15.7 show examples of external reset circuit.

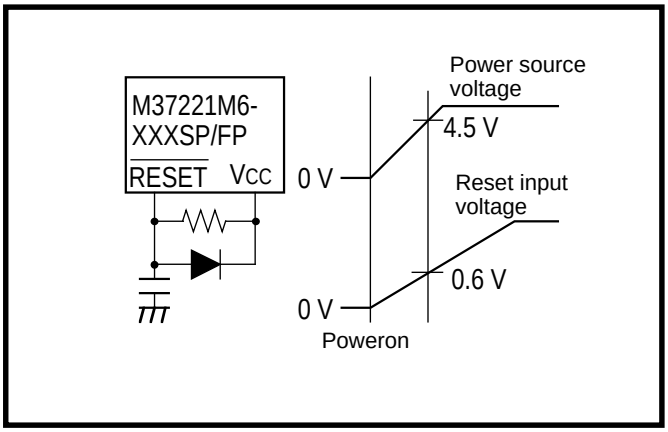


Fig. 2.15.5 Voltage at poweron reset

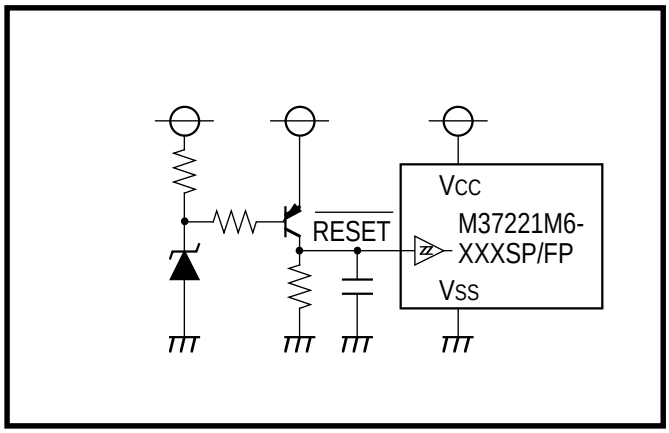


Fig. 2.15.6 Example of reset circuit (1)

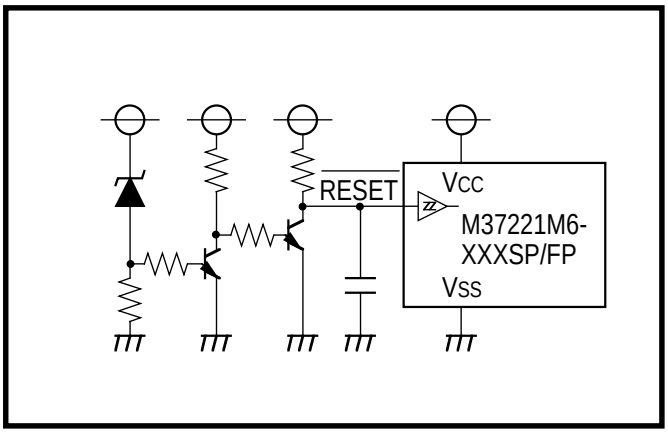


Fig. 2.15.7 Example of reset circuit (2)

2.16 Clock generating circuit

Oscillation circuit consists of an “oscillation gate” which operates as an amplifier to provide the gain required for oscillation and an “oscillating control flip-flop” to control this. Because of that, it is possible to start and stop oscillating as required. For details concerning start and stop of oscillation, refer to **“2.14 Low-power dissipation mode.”** Figure 2.16.1 shows the clock generating circuit block diagram.

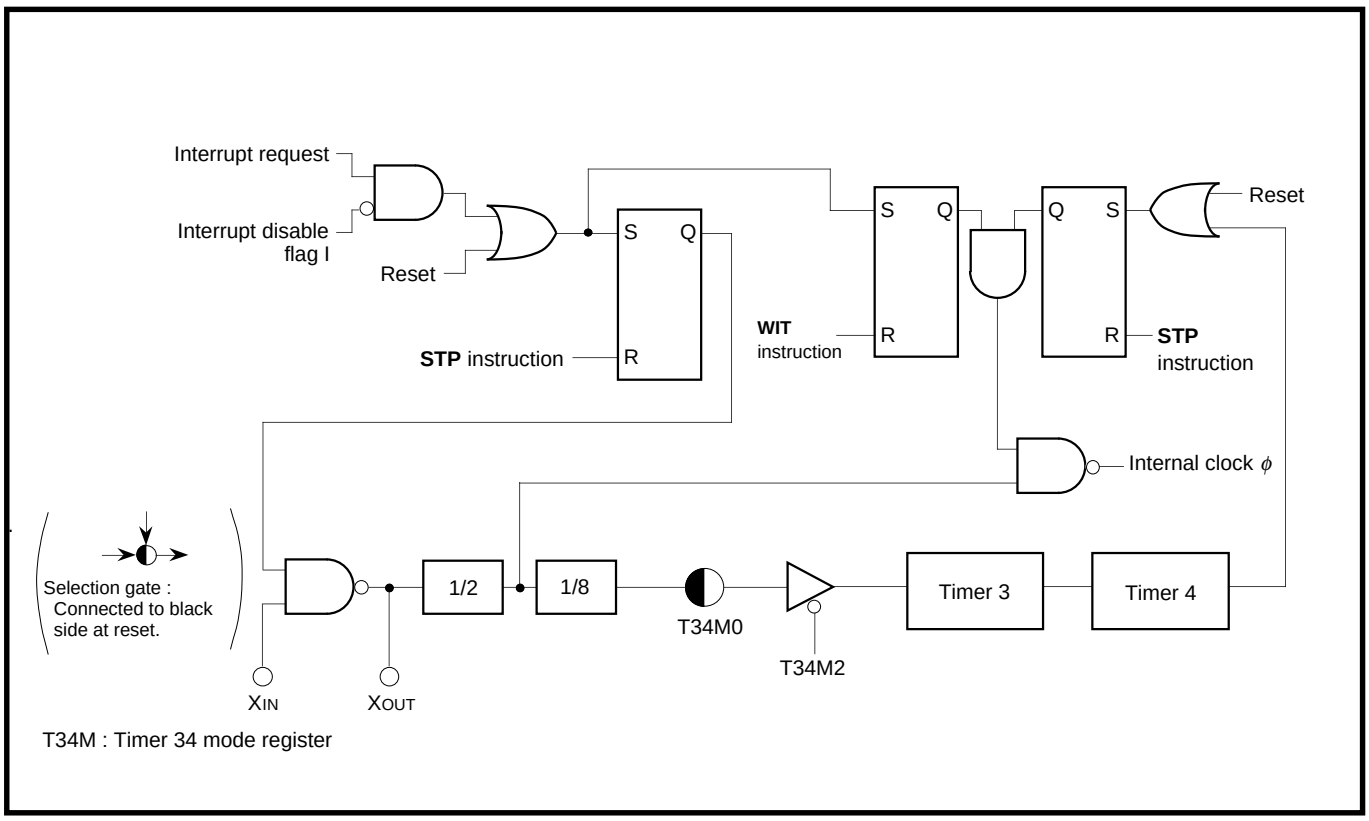


Fig. 2.16.1 Clock generating circuit block diagram

2.17 Oscillation circuit

The M37221M6-XXXSP/FP has a internal oscillation circuits used to obtain the clocks required for operation. Ordinarily, the frequency on clock input pin X_{IN} divided by 2 is the internal clock (internal timing output) ϕ. A quartz-crystal oscillator or ceramic resonator can be connected externally to these circuits.

(1) Oscillation circuit using a quartz-crystal oscillator or ceramic resonator

Figure 2.17.1 shows the circuit example using a quartz-crystal oscillator or a ceramic resonator. As shown in the diagram, oscillation circuit can be constructed by connecting a ceramic resonator (a quartz-crystal oscillator) between pins X_{IN} and X_{OUT}. In this case, set the circuit constants for C_{IN} and C_{OUT} to the values recommended by the resonator manufacturer.

(2) External clock oscillation circuit

Supplying an external clock is possible, Figure 2.17.2 shows the circuit example.

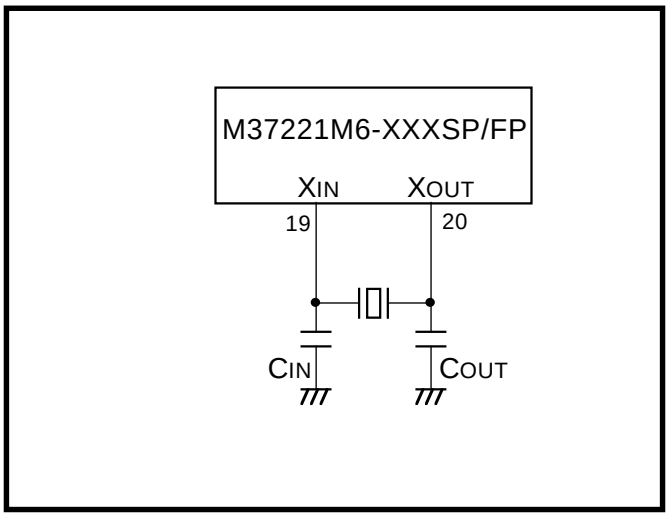


Fig. 2.17.1 Clock oscillation circuit using a ceramic resonator

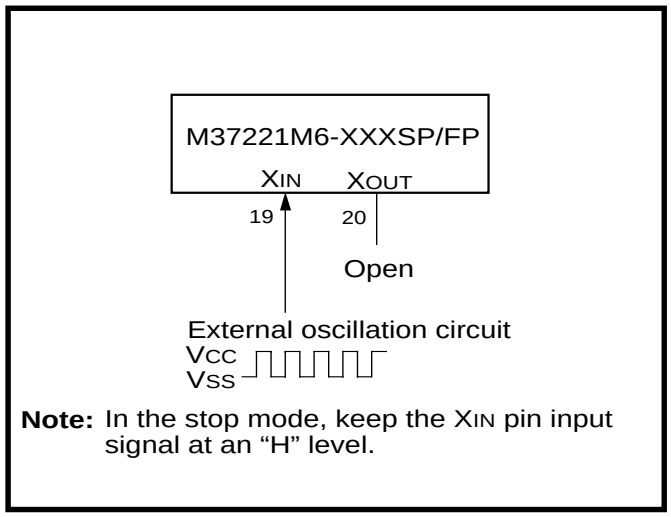


Fig. 2.17.2 External clock input circuit example

The M37221M6-XXXSP/FP has a CRT display clock oscillation circuit, so that display clock can be obtained simply by connecting a inductor and capacitor between pins OSC1 and OSC2. Figure 2.17.3 shows the circuit example.

Refer to "**2.11.9 Clock for display.**"

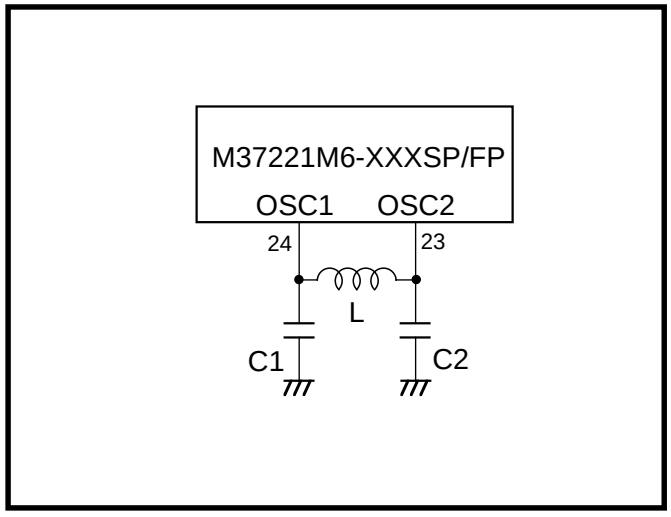


Fig. 2.17.3 Clock oscillation circuit for CRT display

CHAPTER 3

ELECTRICAL CHARACTERISTICS

3.1 Electrical characteristics

Absolute maximum ratings

Symbol	Parameter		Conditions	Ratings	Unit
V_{CC}	Power source voltage V_{CC}		All voltages are based on V_{SS}. Output transistors are cut off.	–0.3 to 6	V
V_I	Input voltage	CNV_{SS}		–0.3 to 6	V
V_I	Input voltage	$P0_0$–$P0_7$,$P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_4$, OSC1, X_{IN}, H_{SYNC}, V_{SYNC}, $\overline{RESET}$		–0.3 to V_{CC} + 0.3	V
V_O	Output voltage	$P0_6$, $P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_2$, R, G, B, OUT1, D-A, X_{OUT}, OSC2		–0.3 to V_{CC} + 0.3	V
V_O	Output voltage	$P0_0$–$P0_5$		–0.3 to 13	V
I_{OH}	Circuit current	R, G, B, OUT1, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$, $P3_1$, D-A		0 to 1 (Note 1)	mA
I_{OL1}	Circuit current	R, G, B, OUT1, $P0_6$, $P0_7$, $P1_0$, $P1_5$–$P1_7$, $P2_0$–$P2_3$, $P3_0$–$P3_2$, D-A		0 to 2 (Note 2)	mA
I_{OL2}	Circuit current	$P1_1$–$P1_4$		0 to 6 (Note 2)	mA
I_{OL3}	Circuit current	$P0_0$–$P0_5$		0 to 1 (Note 2)	mA
I_{OL4}	Circuit current	$P2_4$–$P2_7$		0 to 10 (Note 3)	mA
P_d	Power dissipation		T_a = 25 °C	550	mW
T_{opr}	Operating temperature			–10 to 70	°C
T_{stg}	Storage temperature			–40 to 125	°C

Notes 1: The total current that flows out of the IC must be 20 mA (max.).
2: The total input current to IC (I_{OL1} + I_{OL2} + I_{OL3}) must be 30 mA or less.
3: The total average input current for ports $P2_4$–$P2_7$ to IC must be 20 mA or less.

Recommended operating conditions (T_a = –10 °C to 70 °C, V_{CC} = 5 V ± 10 %, unless otherwise noted)

Symbol	Parameter		Min.	Typ.	Max.	Unit
V_{CC}	Power source voltage (Note 4), During CPU, CRT operation		4.5	5.0	5.5	V
V_{SS}	Power source voltage		0	0	0	V
V_{IH1}	HIGH input voltage	$P0_0$–$P0_7$,$P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_4$, S_{IN}, S_{CLK}, H_{SYNC}, V_{SYNC}, $\overline{RESET}$, X_{IN}, OSC1, TIM2, TIM3, INT1, INT2, INT3	0.8V_{CC}		V_{CC}	V
V_{IH2}	HIGH input voltage	SCL1, SCL2, SDA1, SDA2 (When using I^2C-BUS)	0.7V_{CC}		V_{CC}	V
V_{IL1}	LOW input voltage	$P0_0$–$P0_7$,$P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_4$	0		0.4V_{CC}	V
V_{IL2}	LOW input voltage	SCL1, SCL2, SDA1, SDA2 (When using I^2C-BUS)	0		0.3V_{CC}	V
V_{IL3}	LOW input voltage	H_{SYNC}, V_{SYNC}, $\overline{RESET}$, TIM2, TIM3, INT1, INT2, INT3, X_{IN}, OSC1, S_{IN}, S_{CLK}	0		0.2V_{CC}	V
I_{OH}	HIGH average output current (Note 1)	R, G, B, OUT1, D-A, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$, $P3_1$			1	mA
I_{OL1}	LOW average output current (Note 2)	R, G, B, OUT1, D-A, $P0_6$, $P0_7$, $P1_0$, $P1_5$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_2$			2	mA
I_{OL2}	LOW average output current (Note 2)	$P1_1$–$P1_4$			6	mA
I_{OL3}	LOW average output current (Note 2)	$P0_0$–$P0_5$			1	mA
I_{OL4}	LOW average output current (Note 3)	$P2_4$–$P2_7$			10	mA
$f(X_{IN})$	Oscillation frequency (for CPU operation) (Note 5)	X_{IN}	7.9	8.0	8.1	MHz
f_{CRT}	Oscillation frequency (for CRT display) (Note 5)	OSC1	5.0		8.0	MHz
f_{hs1}	Input frequency	TIM2, TIM3			100	kHz
f_{hs2}	Input frequency	S_{CLK}			1	MHz
f_{hs3}	Input frequency	SCL1, SCL2			400	kHz

Notes 1: The total current that flows out of the IC must be 20 mA (max.).

2: The total input current to IC (I_{OL1} + I_{OL2} + I_{OL3}) must be 30 mA or less.

3: The total average input current for ports $P2_4$–$P2_7$ to IC must be 20 mA or less.

4: Connect 0.1 μF or more capacitor externally across the power source pins V_{CC}–V_{SS} so as to reduce power source noise. Also connect 0.1 μF or more capacitor externally across the pins V_{CC}–CNV$_{SS}$.

5: Use a quartz-crystal oscillator or a ceramic resonator for the CPU oscillation circuit.

ELECTRICAL CHARACTERISTICS

3.1 Electrical characteristics

Electric characteristics (V_{CC} = 5 V ± 10 %, V_{SS} = 0 V, f(X_{IN}) = 8 MHz, T_a = –10 °C to 70 °C, unless otherwise noted)

Symbol	Parameter		Test conditions		Limits			Unit
					Min.	Typ.	Max.	
I_{CC}	Power source current	System operation	V_{CC} = 5.5 V, f(X_{IN}) = 8 MHz	CRT OFF		20	40	mA
				CRT ON		30	60	
		Stop mode	V_{CC} = 5.5 V, f(X_{IN}) = 0				300	μA
V_{OH}	HIGH output voltage	R, G, B, OUT1, D-A, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$, $P3_1$	V_{CC} = 4.5 V, I_{OH} = –0.5 mA		2.4			V
V_{OL}	LOW output voltage	R, G, B, OUT1, D-A, $P0_0$–$P0_7$, $P1_0$, $P1_5$–$P1_7$, $P2_0$–$P2_3$, $P3_0$–$P3_2$	V_{CC} = 4.5 V, I_{OL} = 0.5 mA				0.4	V
	LOW output voltage	$P1_1$–$P1_4$	V_{CC} = 4.5 V	I_{OL} = 3 mA			0.4	
				I_{OL} = 6 mA			0.6	
	LOW output voltage	$P2_4$–$P2_7$	V_{CC} = 4.5 V, I_{OL} = 10.0 mA				3.0	
V_{T+}–V_{T-}	Hysteresis	$\overline{RESET}$	V_{CC} = 5.0 V			0.5	0.7	V
	Hysteresis (Note)	H_{SYNC}, V_{SYNC}, TIM2, TIM3, INT1, INT2, INT3, SCL1, SCL2, SDA1, SDA2, S_{IN}, S_{CLK}	V_{CC} = 5.0 V			0.5	1.3	
I_{IZH}	HIGH input leak current	$\overline{RESET}$, $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_4$, H_{SYNC}, V_{SYNC}	V_{CC} = 5.5 V, V_I = 5.5 V				5	μA
I_{IZL}	LOW input leak current	$\overline{RESET}$, $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_4$, H_{SYNC}, V_{SYNC}	V_{CC} = 5.5 V, V_I = 0 V				5	μA
I_{OZH}	HIGH output leak current	$P0_0$–$P0_5$	V_{CC} = 5.5 V, V_O = 12 V				10	μA
R_{BS}	I^2C-BUS·BUS switch connection resistor (between SCL1 and SCL2, SDA1 and SDA2)		V_{CC} = 4.5 V				130	W

Note: $P0_6$, $P0_7$, $P1_5$, $P2_3$ and $P2_4$ have the hysteresis when these pins are used as interrupt input pins or timer input pins. $P2_0$–$P2_2$ have the hysteresis when these pins are used as serial I/O pins. $P1_1$–$P1_4$ have the hysteresis when these pins are used as multi-master I^2C-BUS interface pins.

A-D Comparator characteristics

(V_{CC} = 5 V ± 10 %, V_{SS} = 0 V, f(X_{IN}) = 8 MHz, T_a = –10 °C to 70 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
—	Resolution				6	bits
—	Absolute accuracy		0	±1	±2	LSB

Note: When V_{CC} = 5 V, 1 LSB = 5/64 V.

Multi-master I²C-BUS bus line characteristics

Symbol	Parameter	Standard clod mode		High-speed clock mode		Unit
		Min.	Max.	Typ.	Max.	
t_{BUF}	Bus free time	4.7		1.3		μs
$t_{HD:STA}$	Hold time for START condition	4.0		0.6		μs
t_{LOW}	LOW period of SCL clock	4.7		1.3		μs
t_R	Rising time of both SCL and SDA signals		1000	20+0.1C_b	300	ns
$t_{HD:DAT}$	Data hold time	0		0	0.9	μs
t_{HIGH}	HIGH period of SCL clock	4.0		0.6		μs
t_F	Falling time of both SCL and SDA signals		300	20+0.1C_b	300	ns
$t_{SU:DAT}$	Data set-up time	250		100		ns
$t_{SU:STA}$	Set-up time for repeated START condition	4.7		0.6		μs
$t_{SU:STO}$	Set-up time for STOP condition	4.0		0.6		μs

Note: C_b = total capacitance of 1 bus line

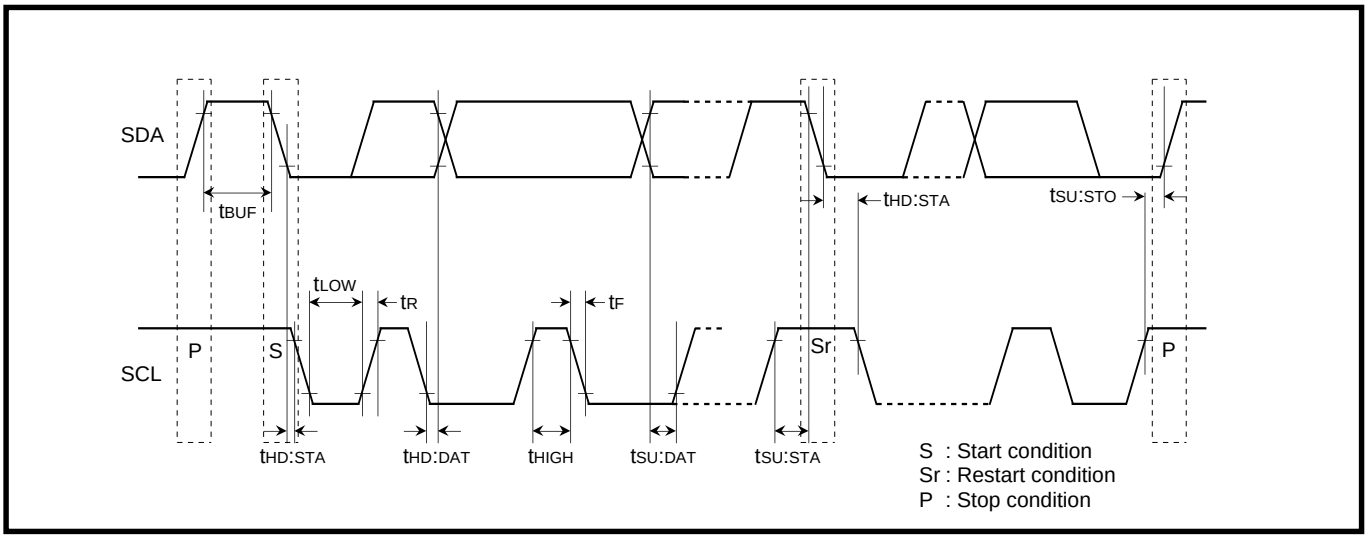


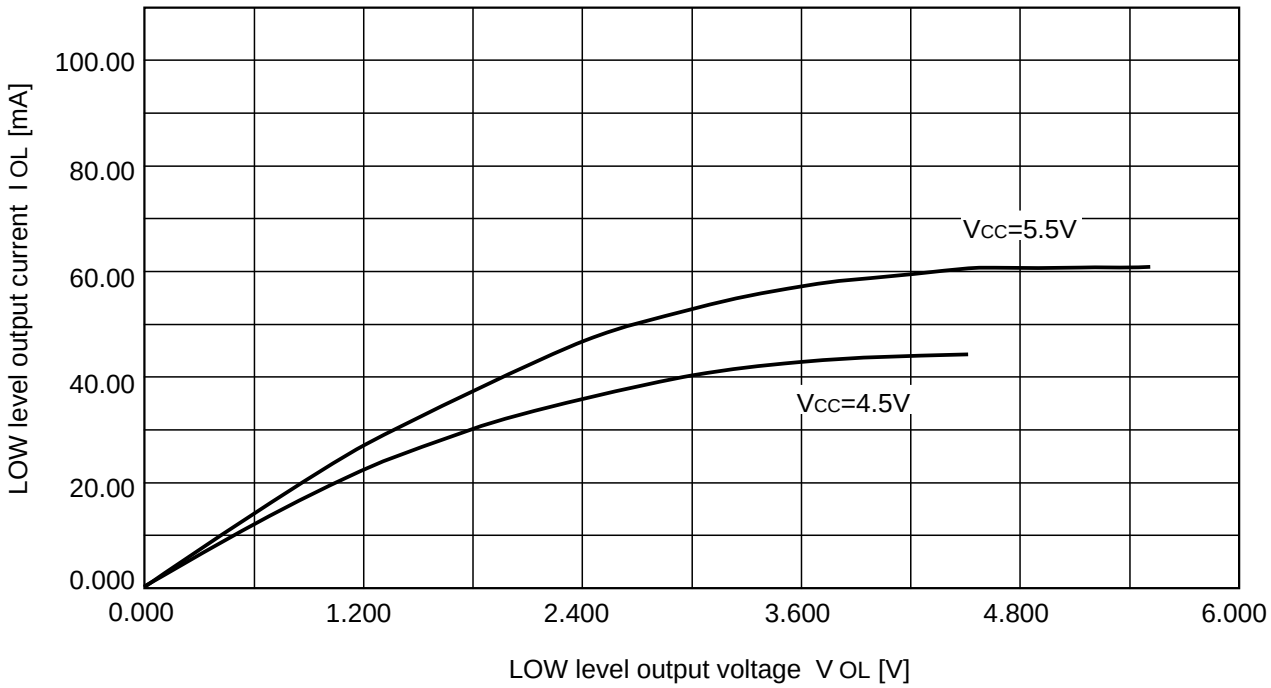
Fig. 3.1.1 Definition diagram of timing on multi-master I²C-BUS

3.2 Standard characteristics

The data described in this section are characteristic examples. Refer to **"3.1 Electrical characteristics"** for rated values.

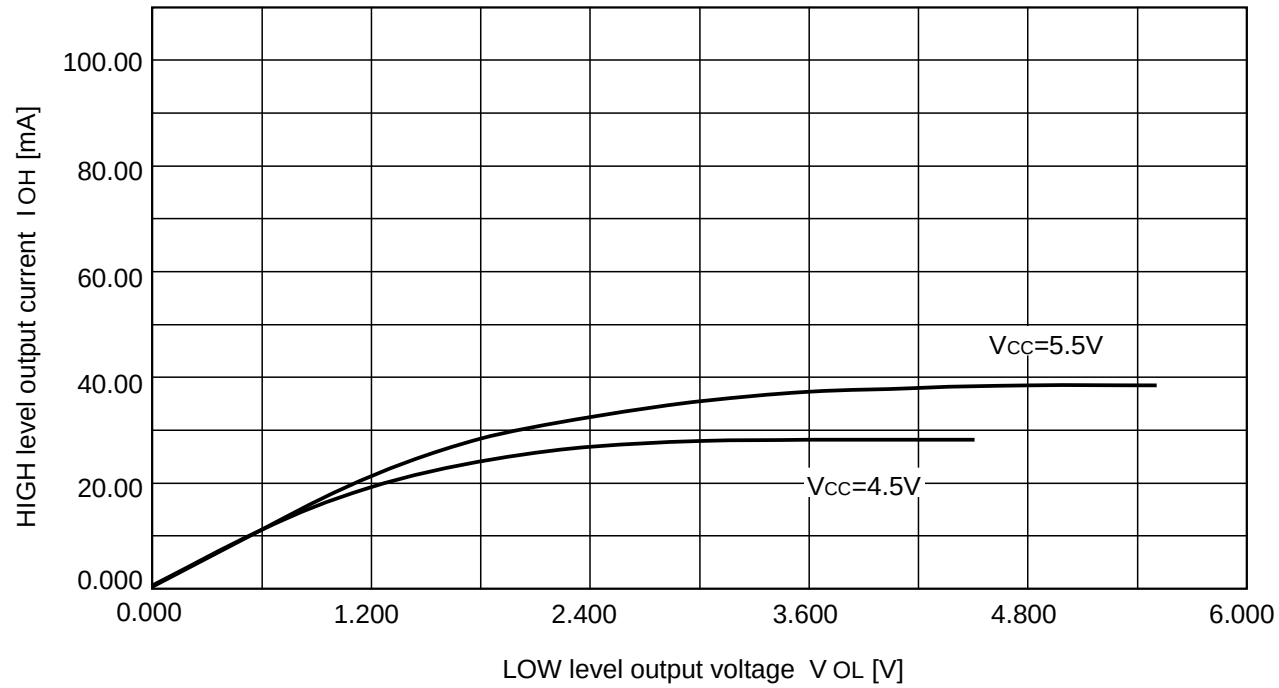
1. Ports $P0_0$–$P0_5$ and $P3_2$

(a) I_{OL}–V_{OL} characteristics



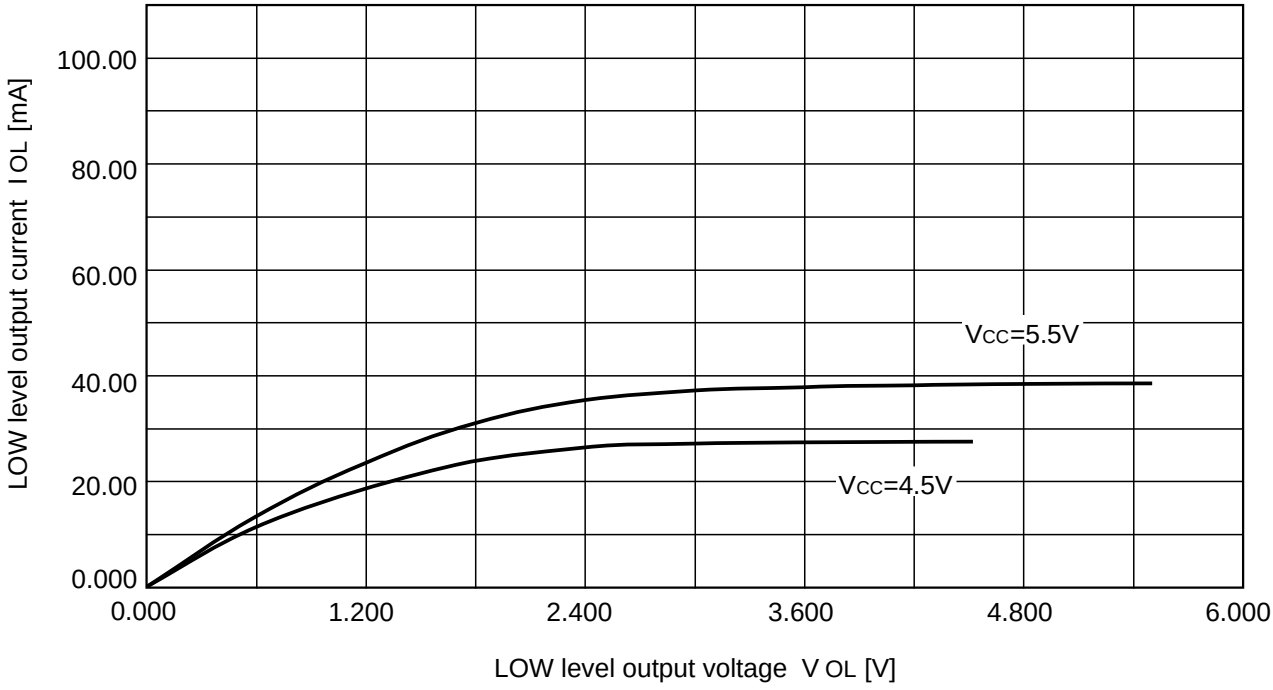
2. Ports $P0_6$ and $P0_7$

(a) I_{OH}–V_{OL} characteristics

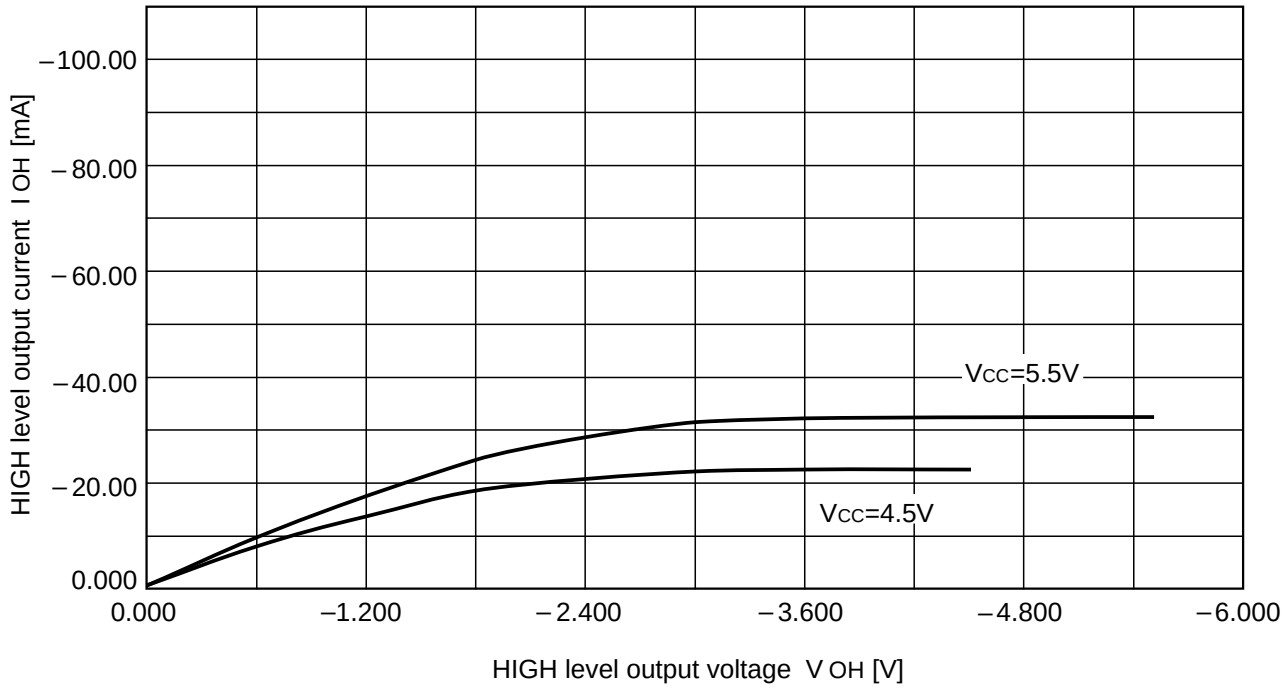


3. Ports $P1_0$, $P1_5$–$P1_7$, $P2_0$–$P2_3$, $P3_0$, $P3_1$ and D-A

(a) I_{OL}–V_{OL} characteristics



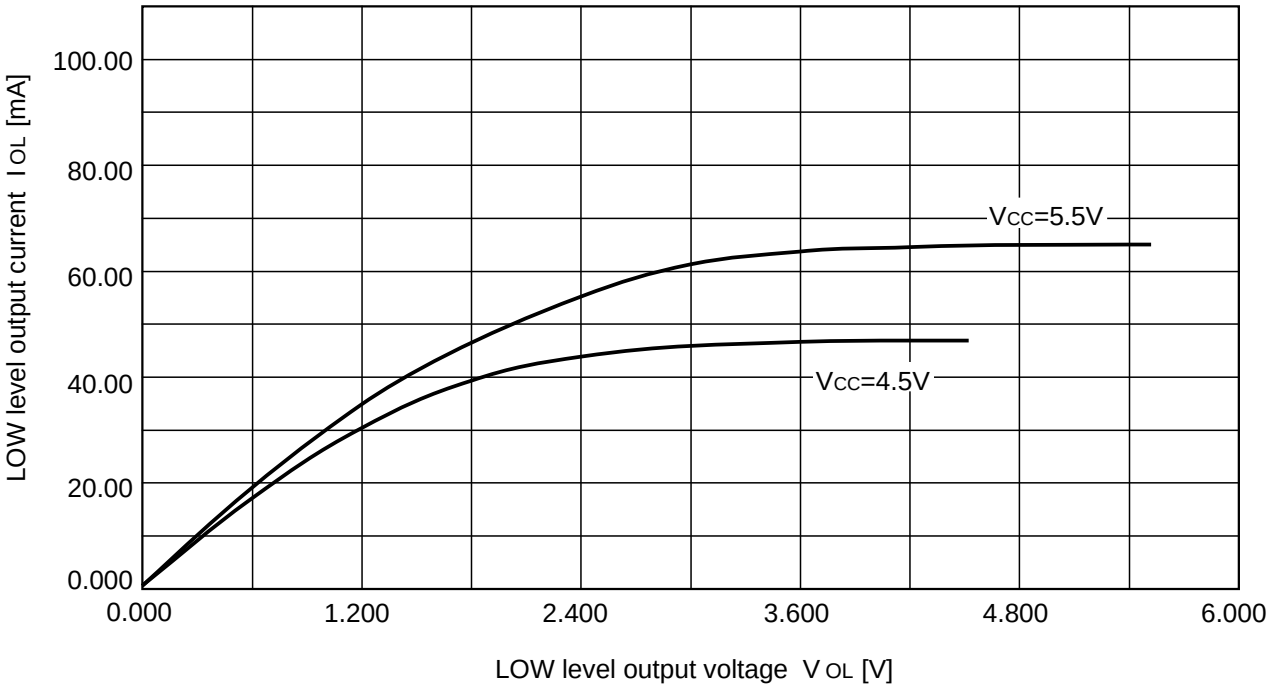
(b) I_{OH}–V_{OH} characteristics



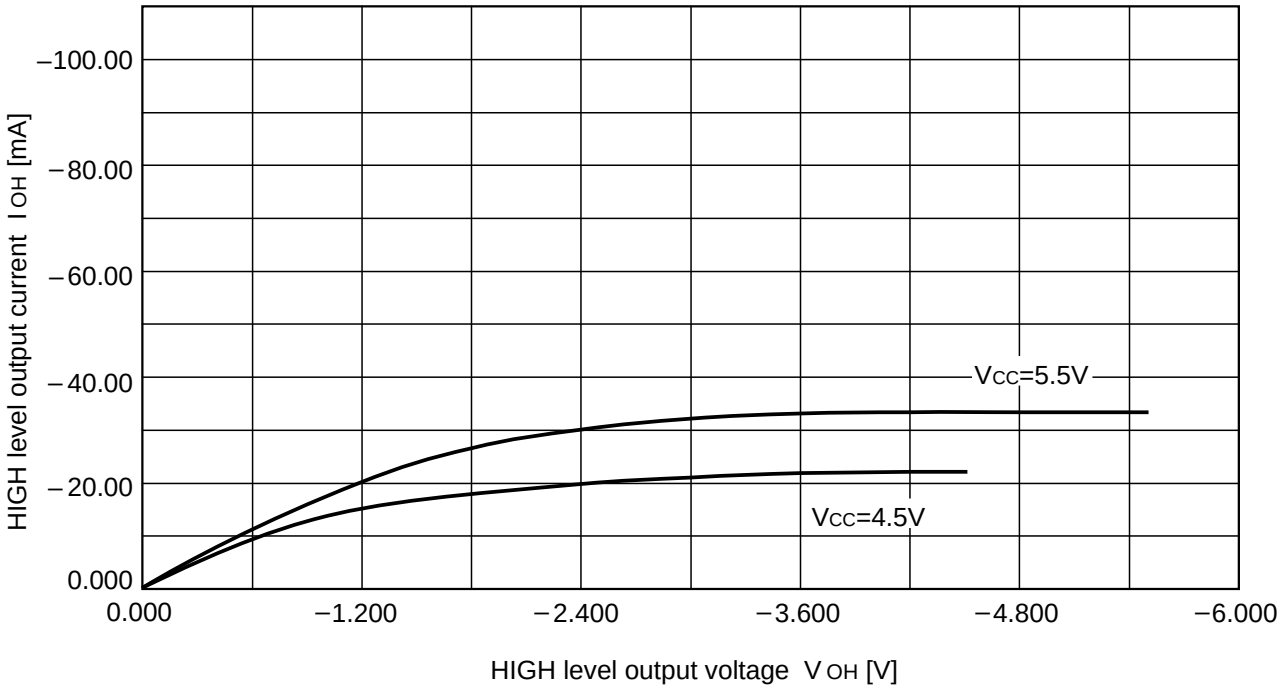
3.2 Standard characteristics

4. Ports P11–P14

(a) I_{OL}–V_{OL} characteristics

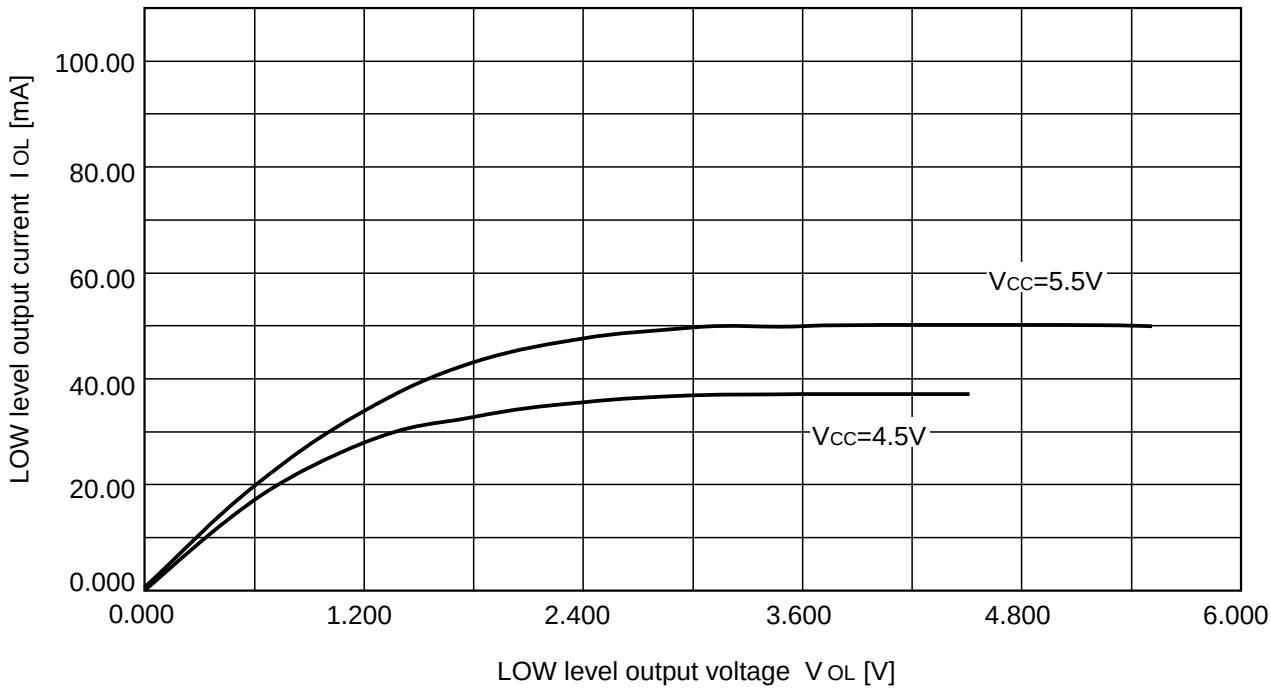


(b) I_{OH}–V_{OH} characteristics

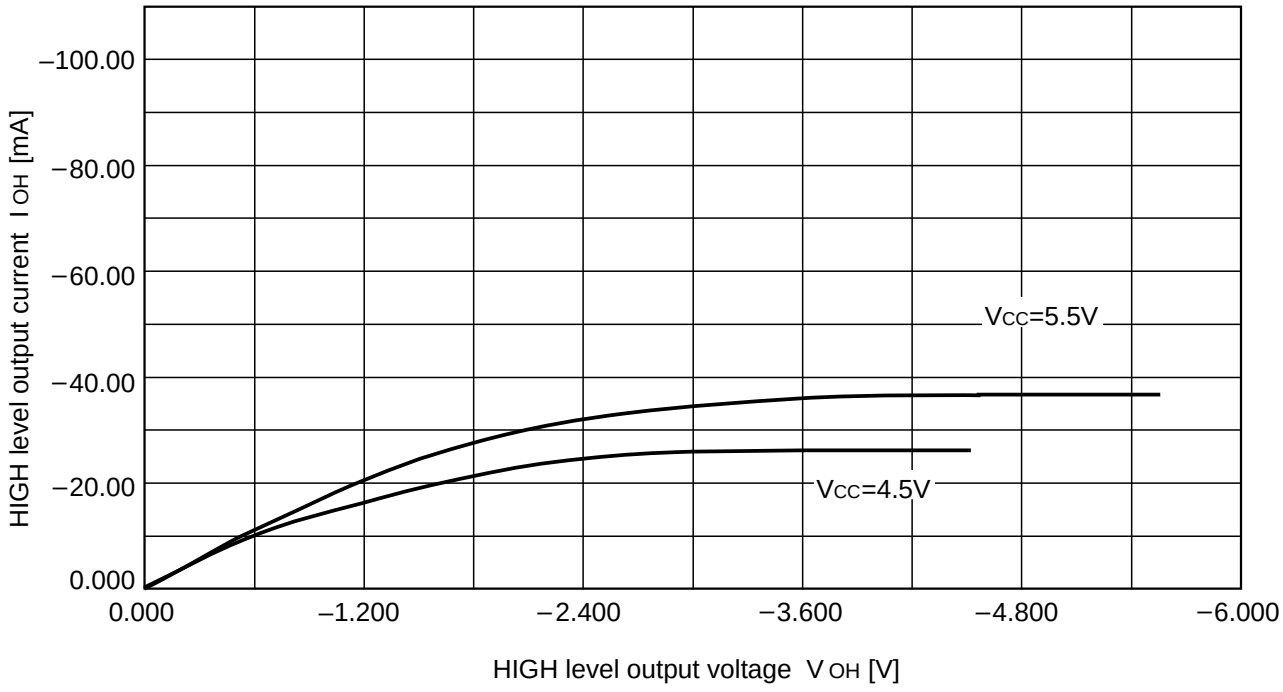


5. Ports $P2_4$–$P2_7$

(a) I_{OL}–V_{OL} characteristics

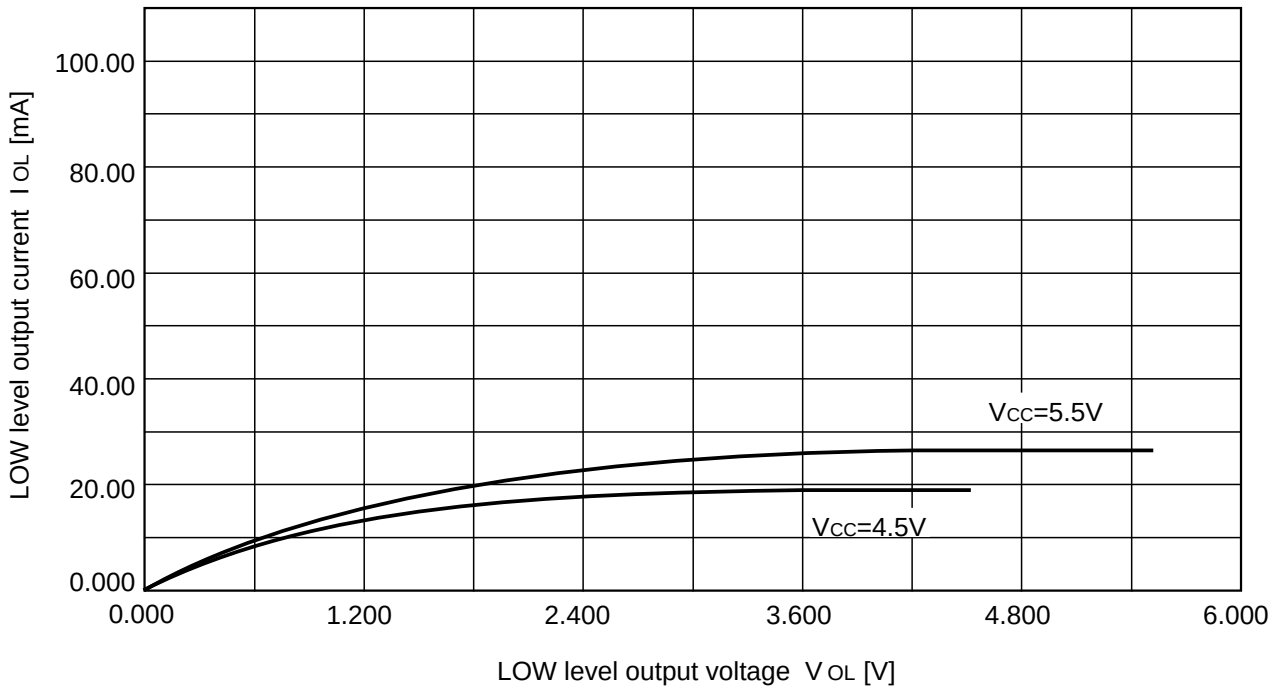


(b) I_{OH}–V_{OH} characteristics

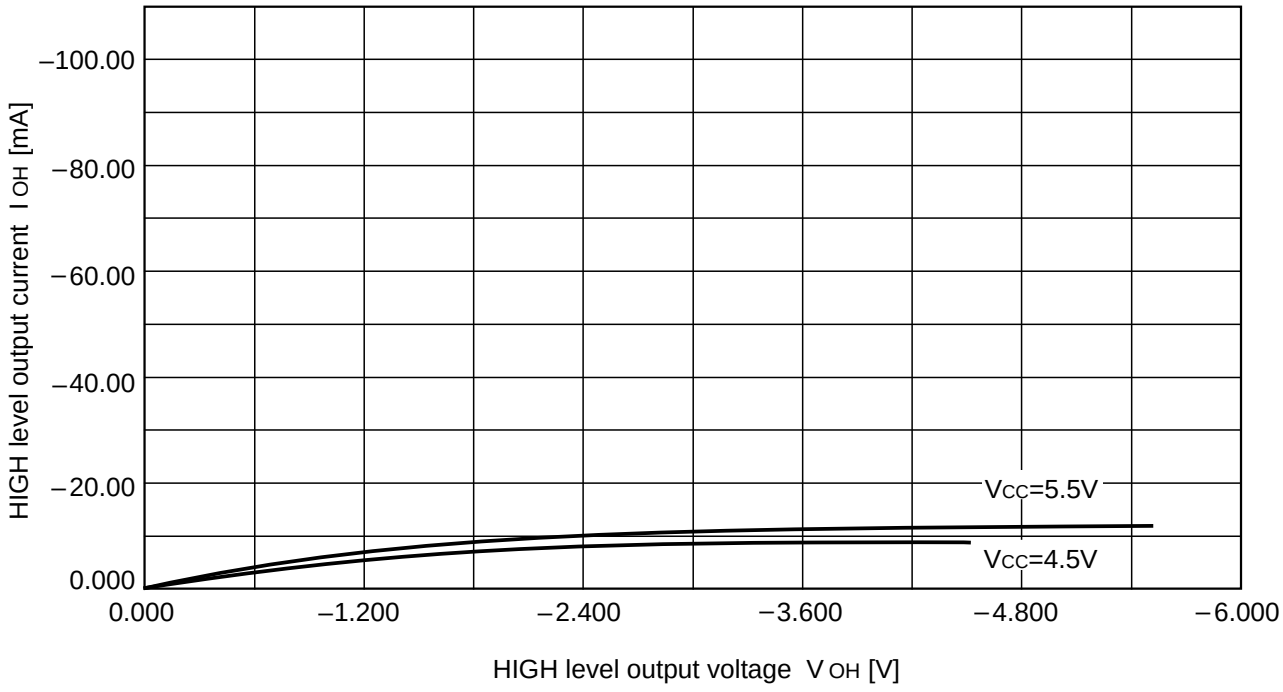


6. Ports P52–P55

(a) I_{OL}–V_{OL} characteristics



(b) I_{OH}–V_{OH} characteristics



CHAPTER 4

M37220M3-XXXSP/FP

4.1 Performance overview

This chapter is described about M37220M3-XXXSP/FP.
M37220M3-XXXSP/FP has the common functions with M37221M6-XXXSP/FP except for part of functions. This chapter explains the differences between M37220M3-XXXSP/FP and M37221M6-XXXSP/FP. Therefore, refer to the corresponding descriptions of M37221M6-XXXSP/FP about the common functions.
The 8-bit microcomputer M37220M3-XXXSP/FP has many additional functions for tuning system for TV:

Table 4.1.1 Performance overview (1)

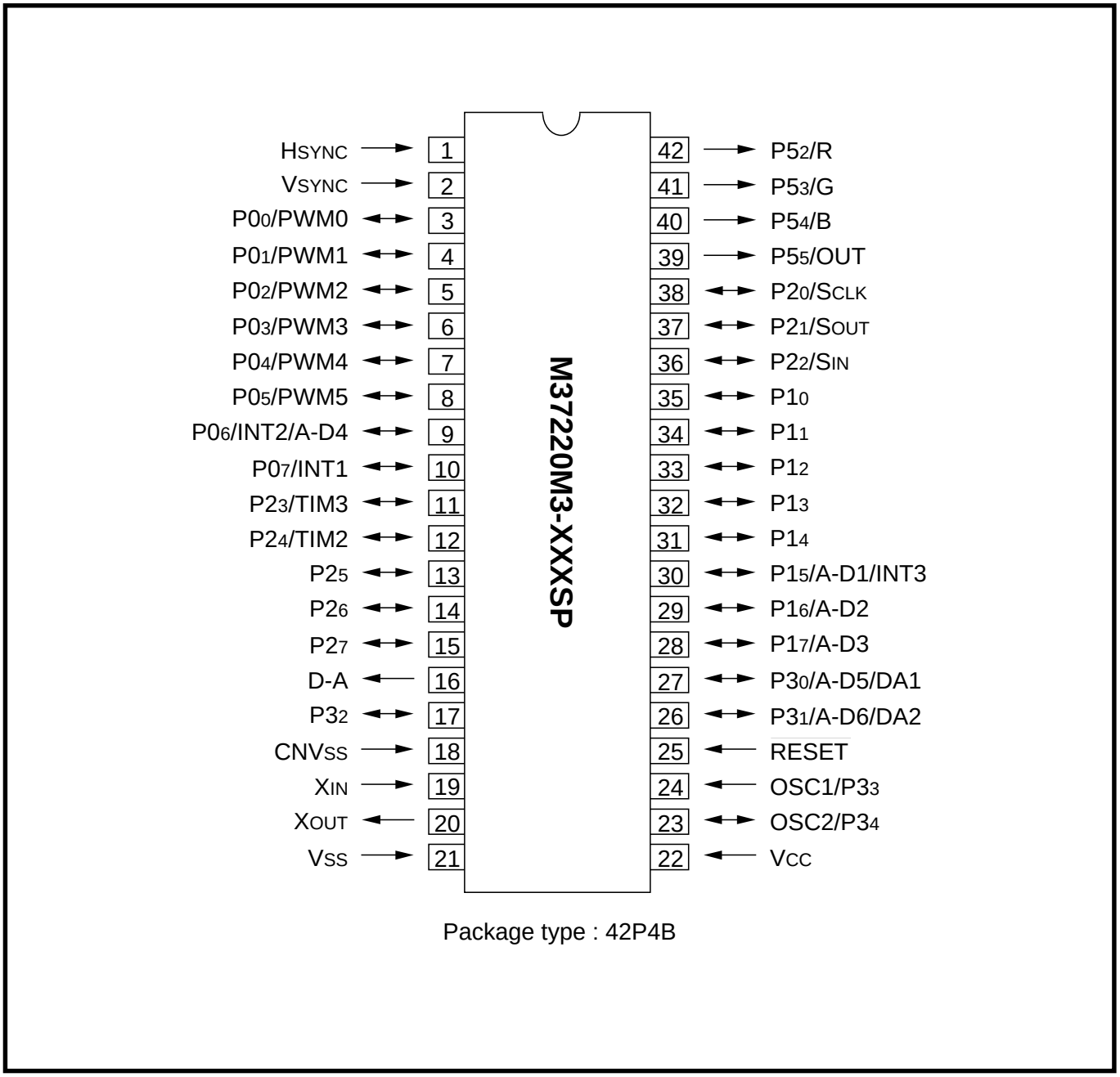
Parameter			Performance
Number of basic instructions			71
Instruction execution time			0.5 μs (the minimum instruction execution time, at 8 MHz oscillation frequency)
Clock frequency			8 MHz (maximum)
Memory size	ROM		12 K bytes
	RAM		256 bytes
	CRT ROM		4 K bytes
	CRT RAM		80 bytes
Input/Output ports	$P0_0$–$P0_7$	I/O	8-bit × 1 (N-channel open-drain output structure, can be used as PWM output pins, INT input pins, A-D input pin)
	$P1_0$–$P1_7$	I/O	8-bit × 1 (CMOS input/output structure, can be used as A-D input pins, INT input pin)
	$P2_0$, $P2_1$	I/O	2-bit × 1 (CMOS input/output or N-channel open-drain output structure, can be used as serial I/O pins)
	$P2_2$–$P2_7$	I/O	6-bit × 1 (CMOS input/output structure, can be used as serial input pin, external clock input pins)
	$P3_0$, $P3_1$	I/O	2-bit × 1 (CMOS input/output or N-channel open-drain output structure, can be used as D-A conversion output pins, A-D input pins)
	$P3_2$	I/O	1-bit × 1 (N-channel open-drain output structure)
	$P3_3$, $P3_4$	Input	2-bit × 1 (can be used as CRT display clock I/O pins)
	$P5_2$–$P5_5$	Output	4-bit × 1 (CMOS output structure, can be used as CRT output pins)
Serial I/O			8-bit × 1
A-D comparator			6 channels (6-bit resolution)
D-A converter			2 (6-bit resolution)
PWM output circuit			14-bit × 1, 8-bit × 6
Timers			8-bit timer × 4
Subroutine nesting			96 levels (maximum)
Interrupt			External interrupt × 3, Internal timer interrupt × 4, Serial I/O interrupt × 1, CRT interrupt × 1, $f(X_{IN})/4096$ interrupt × 1, V_{SYNC} interrupt × 1, BRK interrupt × 1
Clock generating circuit			2 built-in circuits (externally connected a ceramic resonator or a quartz-crystal oscillator)
Power source voltage			5 V ± 10 %

Table 4.1.2 Performance overview (2)

Parameter		Performance
Power dissipation	CRT ON	165 mW typ. (at oscillation frequency f(X_{IN}) = 8 MHz, f_{CRT} = 8 MHz)
	CRT OFF	110 mW typ. (at oscillation frequency f(X_{IN}) = 8 MHz)
	In stop mode	1.65 mW (maximum)
12V withstand ports		6
LED drive ports		4
Operating temperature range		–10 °C to 70 °C
Device structure		CMOS silicon gate process
Package	M37220M3-XXXSP	42-pin shrink plastic molded DIP
	M37220M3-XXXFP	42-pin shrink plastic molded SOP
CRT display function	Number of display characters	20 characters × 2 lines (maximum 16 lines by software)
	Dot structure	12 × 16 dots
	Kinds of characters	128 kinds
	Kinds of character sizes	3 kinds
	Kinds of character colors	Maximum 7 kinds (R, G, B); can be specified by the character
	Display position (horizontal, vertical)	64 levels (horizontal) × 128 levels (vertical)

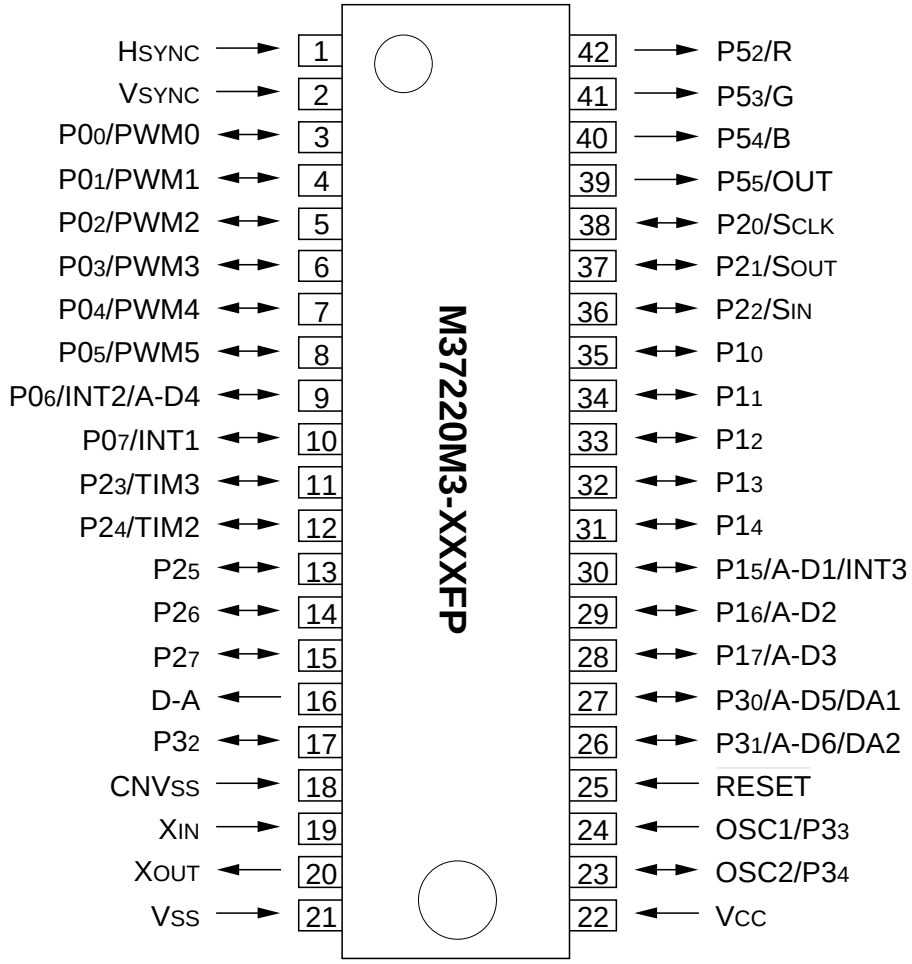
4.2 Pin configuration

The pin configurations are shown in Figures 4.2.1 and 4.2.2.



Package type : 42P4B

Fig. 4.2.1 Pin configuration (top view) (1)



Package type : 42P2R-A

Fig. 4.2.2 Pin configuration (top view) (2)

4.3 Pin description

The pin description of M37220M3-XXXSP/FP is shown in Table 4.3.1.

Table 4.3.1 Pin description (1)

Pin	Name	Input/ Output	Functions
V_{CC}, V_{SS}	Power source		Apply voltage of 5 V ± 10 % (typical) to V_{CC}, and 0 V to V_{SS}.
CNV_{SS}	CNV_{SS}		This is connected to V_{SS}.
$\overline{RESET}$	Reset input	Input	To enter the reset state, the reset input pin must be kept at a "L" for 2 μs or more (under normal V_{CC} conditions). If more time is needed for the quartz-crystal oscillator to stabilize, this "L" condition should be maintained for the required time.
X_{IN}	Clock input	Input	This chip has an internal clock generating circuit. To control generating frequency, an external ceramic resonator or a quartz-crystal oscillator is connected between pins X_{IN} and X_{OUT}. If an external clock is used, the clock source should be connected to the X_{IN} pin and the X_{OUT} pin should be left open.
X_{OUT}	Clock output	Output	
$P0_0$ PWM0–$P0_5$/ PWM5,	I/O port P0	I/O	Port P0 is an 8-bit I/O port with direction register allowing each I/O bit to be individually programmed as input or output. At reset, this port is set to input mode. The output structure is N-channel open-drain output. The note out of this Table gives a full of port P0 function.
$P0_6$/INT2/ A-D4,	PWM output	Output	Pins $P0_0$–$P0_5$ are also used as PWM output pins PWM0–PWM5 respectively. The output structure is N-channel open-drain output.
$P0_7$/INT1	External interrupt input	Input	Pins $P0_6$, $P0_7$ are also used as external interrupt input pins INT2, INT1 respectively.
	Analog input	Input	$P0_6$ pin is also used as analog input pin A-D4.
$P1_1$–$P1_4$, $P1_5$/A-D1/ INT3, $P1_6$/A-D2, $P1_7$/A-D3	I/O port P1	I/O	Port P1 is an 8-bit I/O port and has basically the same functions as port P0. The output structure is CMOS output.
	Analog input	Input	Pins $P1_5$–$P1_7$ are also used as analog input pins A-D1 to A-D3 respectively.
	External interrupt input	Input	$P1_5$ pin is also used as external interrupt input pin INT3.

Table 4.3.2 Pin description (2)

Pin	Name	Input/ Output	Functions
$P2_0/S_{CLK}$, $P2_1/S_{OUT}$, $P2_2/S_{IN}$, $P2_3$/TIM3, $P2_4$/TIM2, $P2_5$–$P2_7$	I/O port P2	I/O	Port P2 is an 8-bit I/O port and has basically the same functions as port P0. The output structure is CMOS output.
	External clock input	Input	Pins $P2_3$, $P2_4$ are also used as external clock input pins TIM3, TIM2 respectively.
	Serial I/O synchronous clock input/ output	I/O	$P2_0$ pin is also used as serial I/O synchronous clock input/output pin S_{CLK}. The output structure is N-channel open-drain output.
	Serial I/O data input/output	I/O	Pins $P2_1$, $P2_2$ are also used as serial I/O data input/output pins S_{OUT}, S_{IN} respectively. The output structure is N-channel open-drain output.
$P3_0$/A-D5/ DA1, $P3_1$/A-D6/ DA2, $P3_2$	I/O port P3	I/O	Ports $P3_0$–$P3_2$ are 3-bit I/O ports and have basically the same functions as port P0. Either CMOS output or N-channel open-drain output structure can be selected as the port $P3_0$ and $P3_1$. The output structure of port $P3_2$ is N-channel open-drain output.
	Analog input	Input	Pins $P3_0$, $P3_1$ are also used as analog input pins A-D5, A-D6 respectively.
	D-A conversion output	Output	Pins $P3_0$, $P3_1$ are also used as D-A conversion output pins DA1, DA2 respectively.
$P3_3$/OSC1, $P3_4$/OSC2	Input port P3	Input	Ports $P3_3$, $P3_4$ are 2-bit input ports.
	Clock input for CRT display	Input	$P3_3$ pin is also used as CRT display clock input pin OSC1.
	Clock output for CRT display	Output	$P3_4$ pin is also used as CRT display clock output pin OSC2. The output structure is CMOS output.
$P5_2$/R, $P5_3$/G, $P5_4$/B, $P5_5$/OUT	Output port P5	Output	Ports $P5_2$–$P5_5$ are 4-bit output ports. The output structure is CMOS output.
	CRT output	Output	Pins $P5_2$–$P5_5$ are also used as CRT output pins R, G, B, OUT respectively. The output structure is CMOS output.
H_{SYNC}	H_{SYNC} input	Input	This is a horizontal synchronous signal input for CRT.
V_{SYNC}	V_{SYNC} input	Input	This is a vertical synchronous signal input for CRT.
D-A	DA output	Output	This is a 14-bit PWM output pin.

4.4 Functional block diagram

The functional block diagram is shown in Figure 4.4.1.

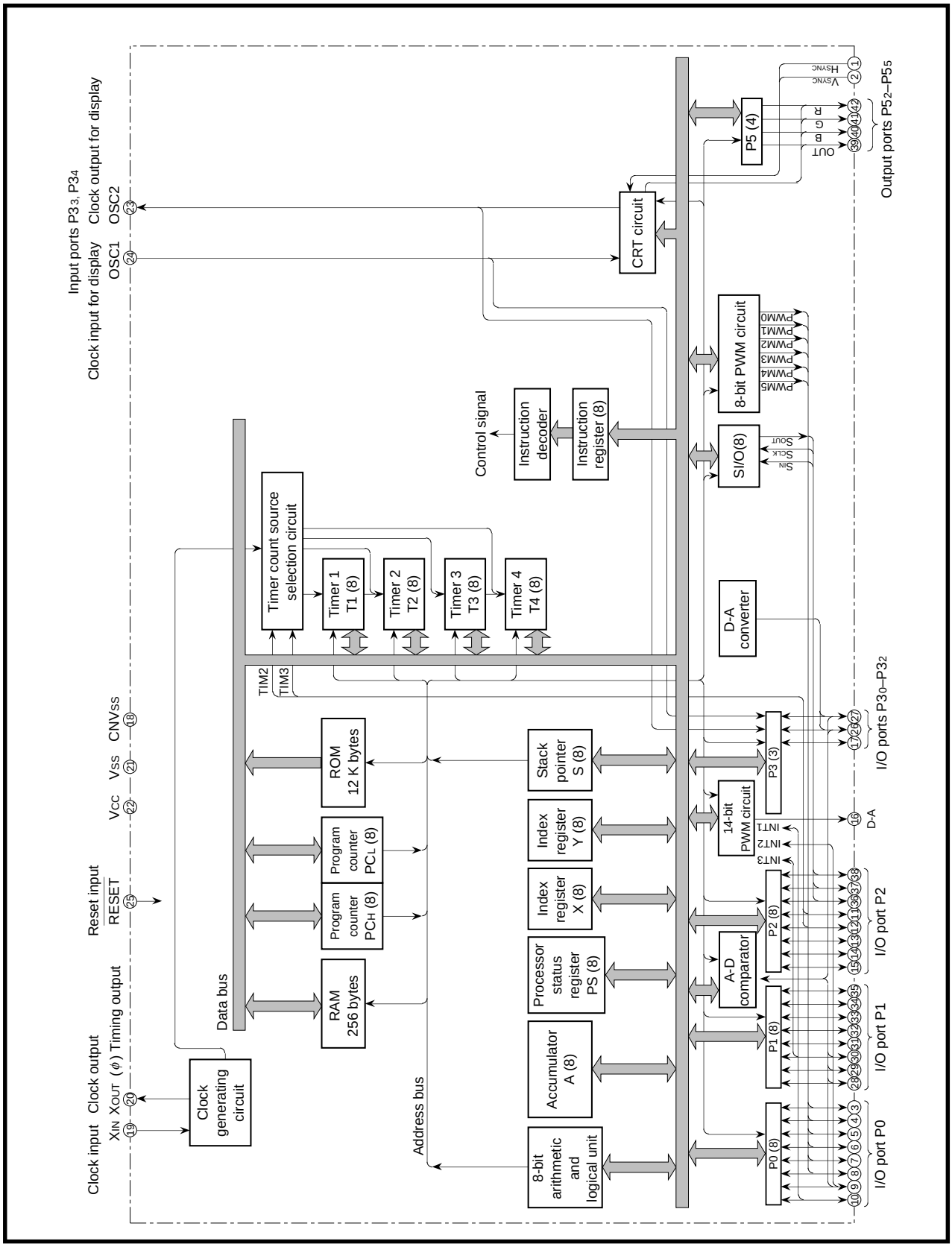


Fig. 4.4.1 Functional block diagram

4.5 Functional description

Functions of M37220M3-XXXSP/FP are partially different from those of M37221M6-XXXSP/FP. Table 4.5.1 shows the difference between M37220M3-XXXSP/FP and M37221M6-XXXSP/FP.

Table 4.5.1 Difference between M37220M3-XXXSP/FP and M37221M6-XXXSP/FP

Paramater		M37220M3-XXXSP/FP	M37221M6-XXXSP/FP
Programmable I/O ports		33	33
	Port P0	8 bits	8 bits
	Port P1	8 bits (Functions except port are partially different.)	8 bits
	Port P2	8 bits	8 bits
	Port P3	8 bits (Functions except port are partically different.)	8 bits
	Port P5	4 bits	4 bits
Interrupts		No multi-master I^2C-BUS interface interrupt (Priority level is the same as M37221M6-XXXSP/FP.)	There is multi-master I^2C-BUS interface interrupt
D-A converter		Included 2 (6-bit resolution)	———
Multi-master I^2C-BUS interface		———	Included 1 (2 systems)
CRT display function			
	Number of display characters	20 characters × 2 lines	24characters × 2 lines
	Kinds of characters	128 kinds	256 kinds
	Kinds of character back ground colors	Not available	Possible (It can be specified by the character.) Maximum 7 kinds

4.5 Functional description

4.5.1 Access area

Figure 4.5.1 shows the M37220M3-XXXSP/FP access area.

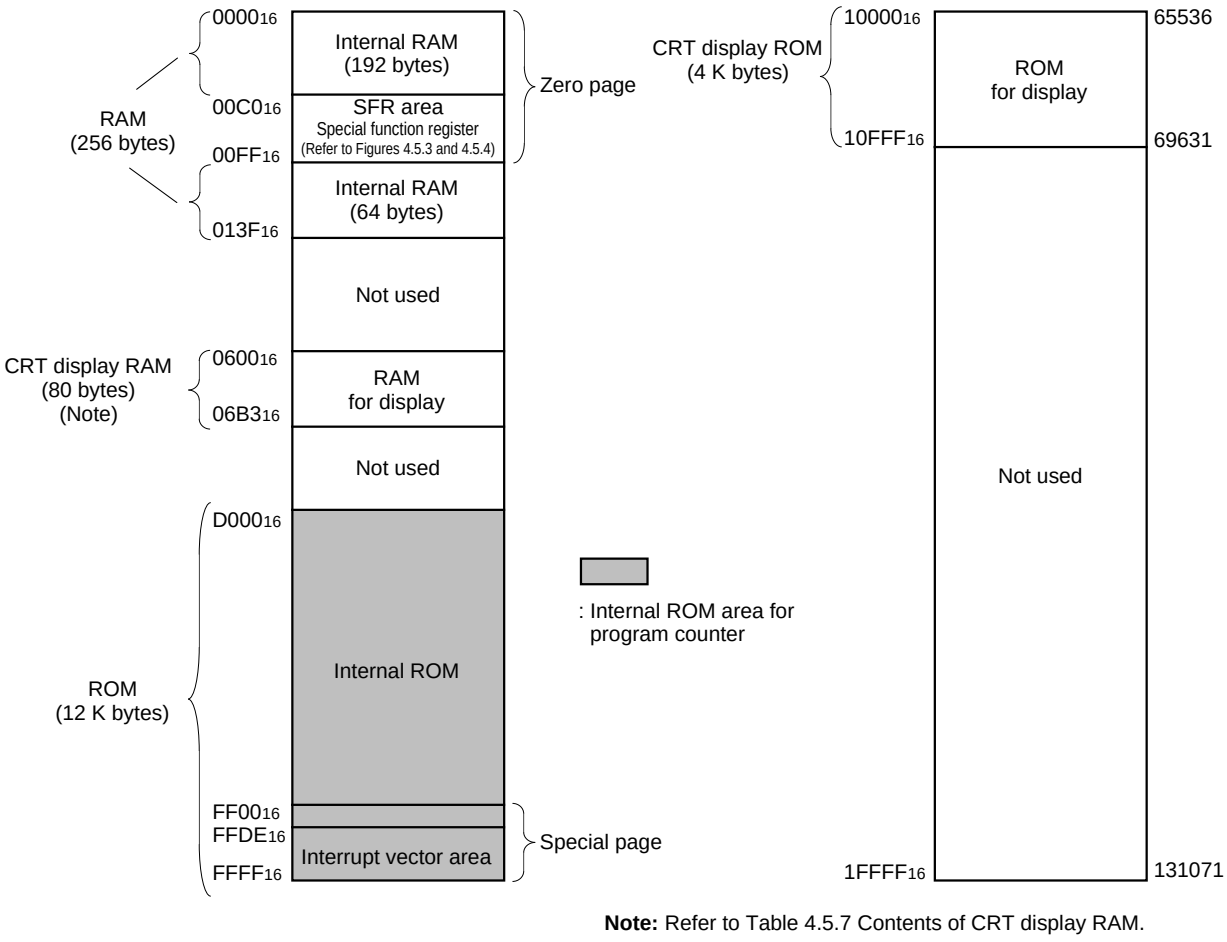


Fig. 4.5.1 Access area

4.5.2 Memory assignment

Figure 4.5.2 shows the memory assignment M37220M3-XXXSP/FP.

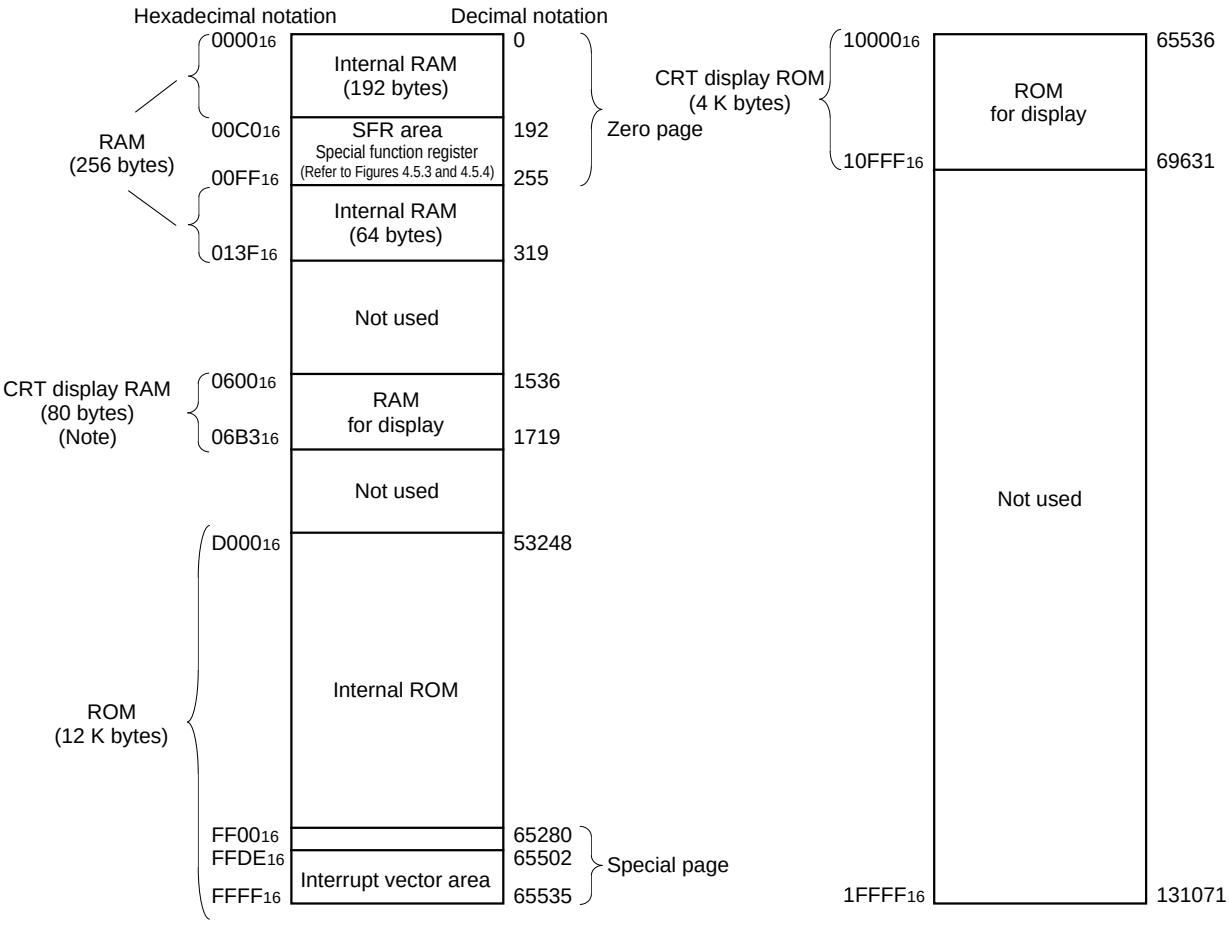


Fig. 4.5.2 Memory assignment

4.5 Functional description

■SFR Area (addresses $C0_{16}$ to DF_{16})

<Bit allocation>

Function bit

: No function bit

0 : Fix this bit to "0" (do not write "1")

1 : Fix this bit to "1" (do not write "0")

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Undefined immediately after reset

Address	Register	Bit allocation b7 b6 b5 b4 b3 b2 b1 b0	State immediately after reset b7 b6 b5 b4 b3 b2 b1 b0
$C0_{16}$	Port P0 (P0)		?
$C1_{16}$	Port P0 direction register (D0)		00_{16}
$C2_{16}$	Port P1 (P1)		?
$C3_{16}$	Port P1 direction register (D1)		00_{16}
$C4_{16}$	Port P2 (P2)		?
$C5_{16}$	Port P2 direction register (D2)		00_{16}
$C6_{16}$	Port P3 (P3)		0 0 0 ? ? ? ? ?
$C7_{16}$	Port P3 direction register (D3)		00_{16}
$C8_{16}$			?
$C9_{16}$			?
CA_{16}	Port P5 (P5)		0 0 ? ? ? ? ? ?
CB_{16}	Port P5 direction register (D5)		00_{16}
CC_{16}			?
CD_{16}	Port P3 output mode control register (P3S)	DA2S DA1S P31S P30S	00_{16}
CE_{16}	DA-H register (DA-H)		?
CF_{16}	DA-L register (DA-L)		0 0 ? ? ? ? ? ?
$D0_{16}$	PWM0 register (PWM0)		?
$D1_{16}$	PWM1 register (PWM1)		?
$D2_{16}$	PWM2 register (PWM2)		?
$D3_{16}$	PWM3 register (PWM3)		?
$D4_{16}$	PWM4 register (PWM4)		?
$D5_{16}$	PWM output control register 1 (PW)	PW7 PW6 PW5 PW4 PW3 PW2 PW1 PW0	00_{16}
$D6_{16}$	PWM output control register 2 (PN)	PN4 PN3 PN2	00_{16}
$D7_{16}$			?
$D8_{16}$			?
$D9_{16}$			?
DA_{16}			?
DB_{16}			?
DC_{16}	Serial I/O mode register (SM)	SM6 SM5 0 SM3 SM2 SM1 SM0	00_{16}
DD_{16}	Serial I/O regsiter (SIO)		?
DE_{16}	DA1 conversion register (DA1)	0 DA15 DA14 DA13 DA12 DA11 DA10	0 0 ? ? ? ? ? ?
DF_{16}	DA2 conversion register (DA2)	0 DA25 DA24 DA23 DA22 DA21 DA20	0 0 ? ? ? ? ? ?

Fig. 4.5.3 Memory map of SFR (special function register) (1)

■SFR Area (addresses $E0_{16}$ to FF_{16})

<Bit allocation>

: } Function bit

Name : } Function bit

: No function bit

0 : Fix this bit to "0" (do not write "1")

1 : Fix this bit to "1" (do not write "0")

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Undefined immediately after reset

Address	Register	Bit allocation b7							b0	State immediately after reset b7							b0
$E0_{16}$	Horizontal position register (HR)			HR5	HR4	HR3	HR2	HR1	HR0	00_{16}							
$E1_{16}$	Vertical position register 1 (CV1)		CV16	CV15	CV14	CV13	CV12	CV11	CV10	0	?	?	?	?	?	?	?
$E2_{16}$	Vertical position register 2 (CV2)		CV26	CV25	CV24	CV23	CV22	CV21	CV20	0	?	?	?	?	?	?	?
$E3_{16}$										?							
$E4_{16}$	Character size register (CS)					CS21	CS20	CS11	CS10	0	0	0	0	?	?	?	?
$E5_{16}$	Border selection register (MD)						MD20		MD10	0	0	0	0	0	?	0	?
$E6_{16}$	Color register 0 (CO0)			CO05		CO03	CO02	CO01		00_{16}							
$E7_{16}$	Color register 1 (CO1)			CO15		CO13	CO12	CO11		00_{16}							
$E8_{16}$	Color register 2 (CO2)			CO25		CO23	CO22	CO21		00_{16}							
$E9_{16}$	Color register 3 (CO3)			CO35		CO33	CO32	CO31		00_{16}							
EA_{16}	CRT control register (CC)						CC2	CC1	CC0	00_{16}							
EB_{16}										?							
EC_{16}	CRT port control register (CRTP)	OP7	OP6	OP5	OUT		R/G/B	VSYC	HSYC	00_{16}							
ED_{16}	CRT clock selection register (CK)	0	0	0	0	0	0	CK1	CK0	00_{16}							
EE_{16}	A-D control register 1 (AD1)				ADM4		ADM2	ADM1	ADM0	0	0	0	?	0	0	0	0
EF_{16}	A-D control register 2 (AD2)			ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	00_{16}							
$F0_{16}$	Timer 1 (TM1)									FF_{16}							
$F1_{16}$	Timer 2 (TM2)									07_{16}							
$F2_{16}$	Timer 3 (TM3)									FF_{16}							
$F3_{16}$	Timer 4 (TM4)									07_{16}							
$F4_{16}$	Timer 12 mode register (T12M)			0	T12M4	T12M3	T12M2	T12M1	T12M0	00_{16}							
$F5_{16}$	Timer 34 mode register (T34M)			T34M5	T34M4	T34M3	T34M2	T34M1	T34M0	00_{16}							
$F6_{16}$	PWM5 register (PWM5)									?							
$F7_{16}$										?							
$F8_{16}$										?							
$F9_{16}$	Interrupt input polarity register (RE)	0		RE5	RE4	RE3	0	0		0	0	0	0	0	0	0	?
FA_{16}	Test register (TEST)	00_{16}								00_{16}							
FB_{16}	CPU mode register (CPUM)	1	1	1	1	1	CM2	0	0	1	1	1	1	1	1	0	0
FC_{16}	Interrupt request register 1 (IREQ1)	IT3R		VSCR	CRTR	TM4R	TM3R	TM2R	TM1R	00_{16}							
FD_{16}	Interrupt request register 2 (IREQ2)	0			MSR		S1R	1T2R	1T1R	00_{16}							
FE_{16}	Interrupt control register 1 (ICON1)	IT3E		VSCE	CRTE	TM4E	TM3E	TM2E	TM1E	00_{16}							
FF_{16}	Interrupt control register 2 (ICON2)	0	0	0	MSE	0	S1E	1T2E	1T1E	00_{16}							

Fig. 4.5.4 Memory map of SFR (special function register) (2)

4.5 Functional description

4.5.3 Input/Output pins

Table 4.5.2 shows the difference of programmable ports between M37221M6-XXXSP/FP and M37220M3-XXXSP/FP.

Table 4.5.2 Difference of programmable ports between M37221M6-XXXSP/FP and M37220M3-XXXSP/FP

Port	Functions except port	
	M37220M3-XXXSP/FP	M37221M6-XXXSP/FP
$P0_0$–$P0_5$	✻	PWM0–PWM5
$P0_6$	✻	INT2/A-D4
$P0_7$	✻	INT1
$P1_0$	No function	OUT2
$P1_1$	No function	SCL1
$P1_2$	No function	SCL2
$P1_3$	No function	SDA1
$P1_4$	No function	SDA2
$P1_5$	✻	A-D1/INT3
$P1_6$	✻	A-D2
$P1_7$	✻	A-D3
$P2_0$	✻	S_{CLK}
$P2_1$	✻	S_{OUT}
$P2_2$	✻	S_{IN}
$P2_3$	✻	TIM3
$P2_4$	✻	TIM2
$P2_5$–$P2_7$	✻	—
$P3_0$	A-D5/DA1	A-D5
$P3_1$	A-D6/DA2	A-D6
$P3_2$	✻	—
$P3_3$	✻	OSC1
$P3_4$	✻	OSC2
$P5_2$	✻	R
$P5_3$	✻	G
$P5_4$	✻	B
$P5_5$	OUT	OUT1

✻: It is the same as M37221M6-XXXSP/FP.

4.5.4 Interrupts

The M37220M3-XXXSP/FP has 13 sources (reset is included) of interrupts.

Table 4.5.3 Interrupt sources, vector addresses and priority

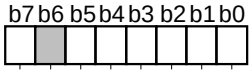
Priority	Interrupt sources	Vector addresses		Remarks
		High-order byte	Low-order byte	
1	Reset (Note)	$FFFF_{16}$	$FFFE_{16}$	Non-maskable
2	CRT interrupt	$FFFD_{16}$	$FFFC_{16}$	
3	INT2 interrupt	$FFFB_{16}$	$FFFA_{16}$	Active edge selectable
4	INT1 interrupt	$FFF9_{16}$	$FFF8_{16}$	Active edge selectable
5	Timer 4 interrupt	$FFF5_{16}$	$FFF4_{16}$	
6	$f(X_{IN})/4096$ interrupt	$FFF3_{16}$	$FFF2_{16}$	
7	VSYNC interrupt	$FFF1_{16}$	$FFF0_{16}$	Active edge selectable
8	Timer 3 interrupt	$FFEF_{16}$	$FFEE_{16}$	
9	Timer 2 interrupt	$FFED_{16}$	$FFEC_{16}$	
10	Timer 1 interrupt	$FFEB_{16}$	$FFEA_{16}$	
11	Serial I/O interrupt	$FFE9_{16}$	$FFE8_{16}$	
12	INT3 interrupt	$FFE5_{16}$	$FFE4_{16}$	Active edge selectable
13	BRK instruction interrupt	$FFDF_{16}$	$FFDE_{16}$	Non-maskable (software interrupt)

Note: Reset are included in the table because it operates in the same way as interrupts.

The different interrupt-related registers from those of M37221M6-XXXSP/FP are shown in the following pages.

4.5 Functional description

Interrupt Request Register 1



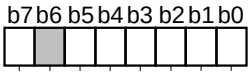
Interrupt request register 1 (IREQ1) [Address 00FC16]

B	Name	Functions	After reset	R	W
0	Timer 1 interrupt request bit (TM1R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
1	Timer 2 interrupt request bit (TM2R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
2	Timer 3 interrupt request bit (TM3R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
3	Timer 4 interrupt request bit (TM4R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
4	CRT interrupt request bit (CRTR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
5	VSYNC interrupt request bit (VSCR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
6	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is “0.”		0	R	—
7	INT3 interrupt request bit (IT3R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻

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

Fig. 4.5.5 Interrupt request register 1 (address 00FC16)

Interrupt Control Register 1



Interrupt control register 1 (ICON1) [Address 00FE16]

B	Name	Functions	After reset	R	W
0	Timer 1 interrupt enable bit (TM1E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
1	Timer 2 interrupt enable bit (TM2E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
2	Timer 3 interrupt enable bit (TM3E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
3	Timer 4 interrupt enable bit (TM4E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
4	CRT interrupt enable bit (CRTE)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
5	VSYNC interrupt enable bit (VSCE)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
6	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is “0.”		0	R	—
7	INT3 interrupt enable bit (IT3E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W

Fig. 4.5.6 Interrupt control register 1 (address 00FE16)

4.5.5 D-A converter

M37220M3-XXXSP/FP has 2 D-A converter with 6-bit resolution. Figure 4.5.7 shows the D-A converter block diagram.

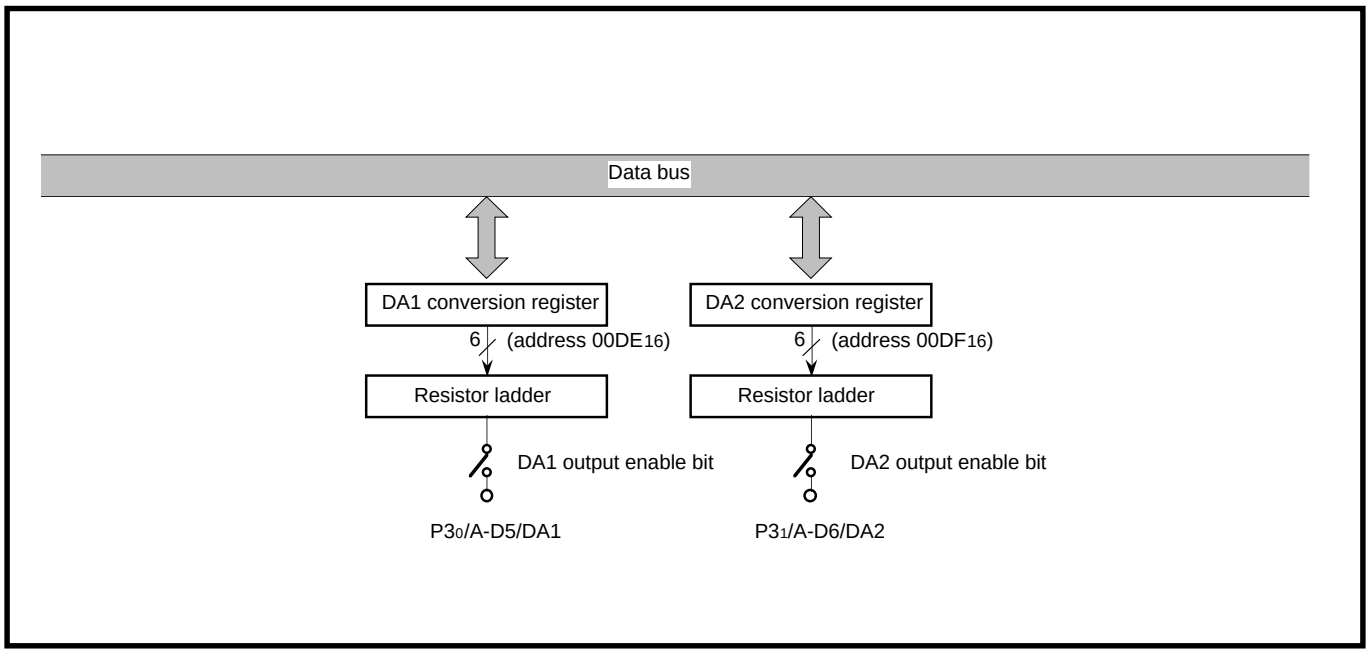


Fig. 4.5.7 D-A converter block diagram

D-A conversion is performed by setting the value in the DA conversion register. The result of D-A conversion is output from the DA pin by setting "1" to the DA output enable bit of the port P3 output mode control register (bits 2 and 3 at address $00CD_{16}$). The output analog voltage V is determined with the value n (n: decimal number) in the DA conversion register.

$$V = V_{CC} \times \frac{n}{64} \quad (n = 0 \text{ to } 63)$$

The DA output does not build in a buffer, so connect an external buffer when driving a low-impedance load.

Table 4.5.4 Relationship between contents of D-A conversion register and output voltage "V"

A-D control register						Output
bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	voltage "V"
0	0	0	0	0	0	0/64 V_{CC}
0	0	0	0	0	1	1/64 V_{CC}
0	0	0	0	1	0	2/64 V_{CC}
:	:	:	:	:	:	:
1	1	1	1	0	1	61/64 V_{CC}
1	1	1	1	1	0	62/64 V_{CC}
1	1	1	1	1	1	63/64 V_{CC}

4.5 Functional description

DA n Conversion Register

b7 b6 b5 b4 b3 b2 b1 b0

0

DA n conversion register (DAn) (n = 1 and 2) [Address 00DE16, 00DF16]

B	Name	Functions	After reset	R	W
0 to 5	DA conversion set bits (DAn0–DAn5)	b5 b4 b3 b2 b1 b0 0 0 0 0 0 0 : 0/64Vcc 0 0 0 0 0 1 : 1/64Vcc 0 0 0 0 1 0 : 2/64Vcc ⋮ 1 1 1 1 0 1 : 61/64Vcc 1 1 1 1 1 0 : 62/64Vcc 1 1 1 1 1 1 : 63/64Vcc	Indeterminate	R	W
6	Fix this bit to “0.”		0	R	—
7	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is “ 0.”		0	R	—

Fig. 4.5.8 DA n conversion register (addresses 00DE16 and 00DF16)

Port P3 Output Mode Control Register

b7 b6 b5 b4 b3 b2 b1 b0

Port P3 output mode control register (P3S) [address 00CD16]

B	Name	Functions	After reset	R	W
0	$P3_0$ output structure selection bit (P30S)	0 : CMOS output 1 : N-channel open-drain output	0	R	W
1	$P3_1$ output structure selection bit (P31S)	0 : CMOS output 1 : N-channel open-drain output	0	R	W
2	DA1 output enable bit (DA1S)	0 : $P3_0$ input/output 1 : DA1 output	0	R	W
3	DA2 output enable bit (DA2S)	0 : $P3_1$ input/output 1 : DA2 output	0	R	W
4 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are “0.”		0	R	—

Fig. 4.5.9 Port P3 output mode control register (address 00CD16)

4.5.6 CRT Display function

Table 4.5.5 shows the outline the CRT display function of the M37220M3-XXXSP/FP.

Table 4.5.5 Outline of CRT display function

Parameter		Performance
Number of display character		20 characters × 2 lines
Dot structure		12 dots × 16 dots
Kinds of character		128 kinds
Kinds of character sizes		3 kinds
Color	Kind of colors	1 screen; 4 kinds, maximum 7 kinds
	Coloring unit	A character
Display extension		Possible (multiline display)
Raster coloring		Possible (maximum 7 kinds)

4.5 Functional description

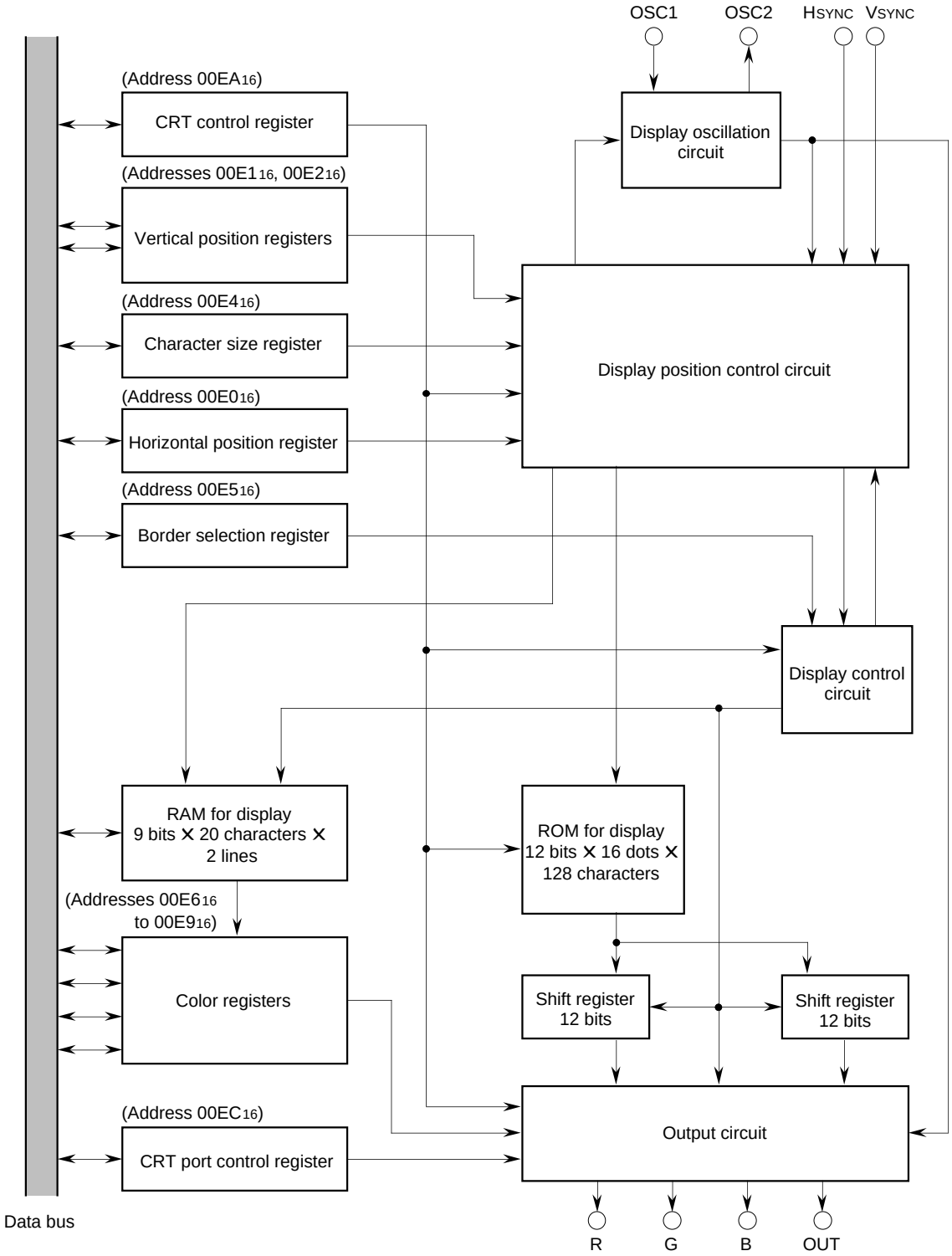


Fig. 4.5.10 CRT display circuit block diagram

(1) Memory for display

There are 2 types of display memory: CRT display ROM (addresses 10000_{16} to $10FFF_{16}$) and CRT display RAM (addresses 0600_{16} to $06B3_{16}$). Each type of display memory is described below.

■ **CRT display ROM (addresses 10000_{16} to $10FFF_{16}$)**

CRT display ROM has a capacity of 4 K bytes. Since 32 bytes are required for 1 character data, the ROM can stores up to 128 kinds of characters.

CRT display ROM is broadly divided into 2 areas. The [vertical 16 dots] × [horizontal (left side) 8 dots] data of display characters are stored in addresses 10000_{16} to $107FF_{16}$; the [vertical 16 dots] × [horizontal (right side) 4 dots] data of display characters are stored in addresses 10800_{16} to $10FFF_{16}$ (refer to **"Figure 4.5.11"**). Note however that the high-order 4 bits of the data to be written to addresses 10800_{16} to $10FFF_{16}$ must be set to "1" (by writing data FX_{16}).

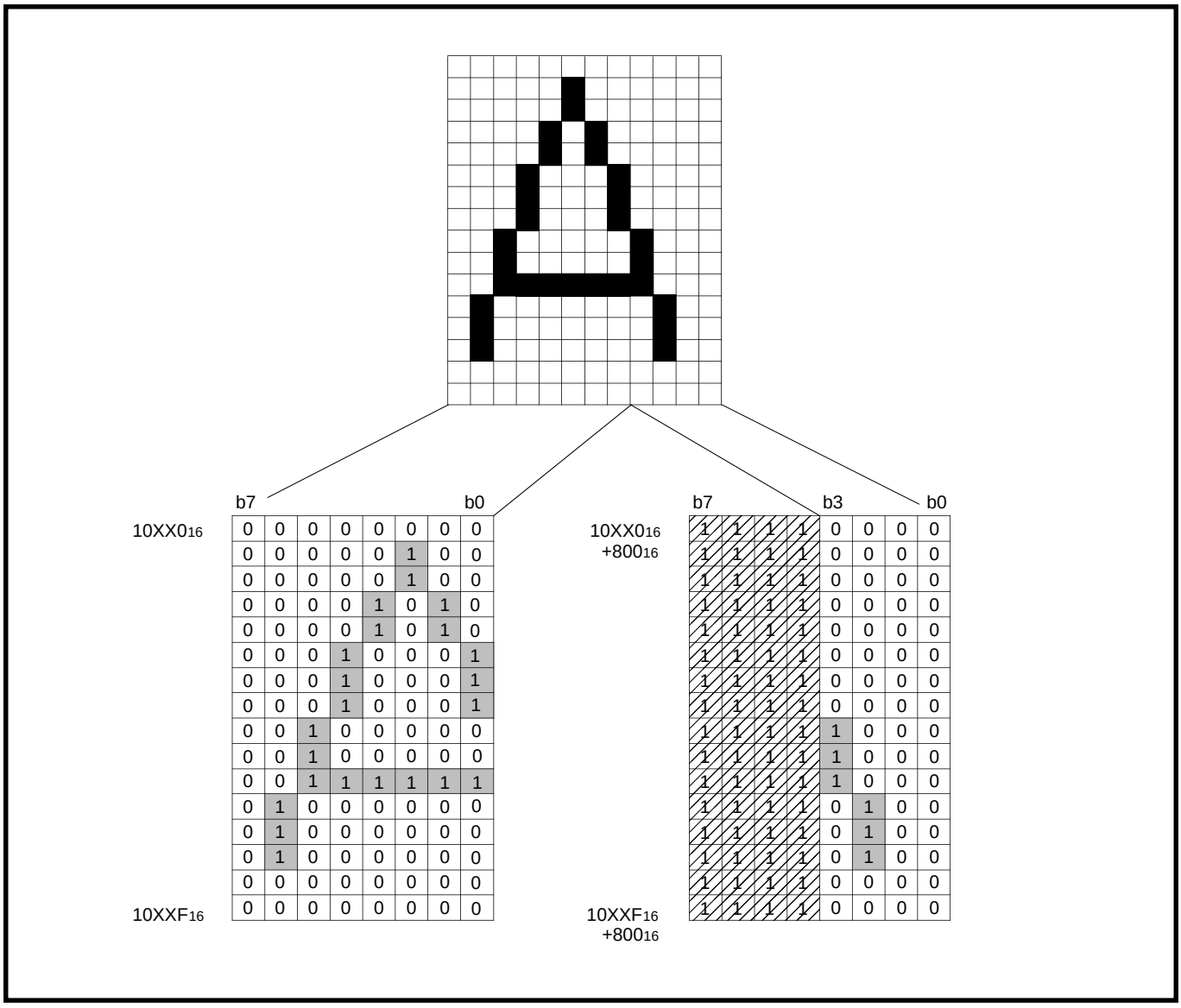


Fig. 4.5.11 Example of display character data storing form

4.5 Functional description

The character code used to specify a display character is determined based on the address in the CRT display ROM in which that character data is stored.
Assume that 1 character data is stored in addresses $10XX0_{16}$ to $10XXF_{16}$ (XX denotes 00_{16} to $7F_{16}$) and $10YY0_{16}$ to $10YYF_{16}$ (YY denotes "XX+800_{16}"), then the character code is "XX_{16}."
In other words, a character code is constructed with the low-order second and third digits (hexadecimal notation) of the 5-digit address (10000_{16} to $107FF_{16}$) where that character data is stored.
A character code is "YY_{16}" in addresses 11000_{16} to $11FFF_{16}$.
Table 4.5.6 shows the character code table.

Table 4.5.6 Character code table (be omitted partly)

Character code	Character data stored address	
	Left side 8 dots	Right 4 side 8 dots
00_{16}	10000_{16} to $1000F_{16}$	10800_{16} to $1080F_{16}$
01_{16}	10010_{16} to $1001F_{16}$	10810_{16} to $1081F_{16}$
02_{16}	10020_{16} to $1002F_{16}$	10820_{16} to $1082F_{16}$
03_{16}	10030_{16} to $1003F_{16}$	10830_{16} to $1083F_{16}$
:	:	:
$7E_{16}$	$107E0_{16}$ to $107EF_{16}$	$10FE0_{16}$ to $10FEF_{16}$
$7F_{16}$	$107F0_{16}$ to $107EF_{16}$	$10FF0_{16}$ to $10FFF_{16}$

■ **CRT display RAM (addresses 0600_{16} to $06B3_{16}$)**
CRT display RAM is assigned to addresses 0600_{16} to $06B3_{16}$. Table 4.5.7 shows the contents of CRT display RAM.

Table 4.5.7 Contents of CRT display RAM

Block number	Display position (from left side)	Character code specifying	Color specifying
Block 1	1st character	0600_{16}	0680_{16}
	2nd character	0601_{16}	0681_{16}
	3rd character	0602_{16}	0682_{16}
	:	:	:
	18th character	0611_{16}	0691_{16}
	19th character	0612_{16}	0692_{16}
	20th character	0613_{16}	0693_{16}
Not used		0614_{16} to $061F_{16}$	0694_{16} : $069F_{16}$
Block 2	1st character	0620_{16}	$06A0_{16}$
	2nd character	0621_{16}	$06A1_{16}$
	3rd character	0622_{16}	$06A2_{16}$
	:	:	:
	18th character	0631_{16}	061_{16}
	19th character	0632_{16}	062_{16}
	20th character	0633_{16}	063_{16}

Figure 4.5.12 shows the structure of CRT display RAM.

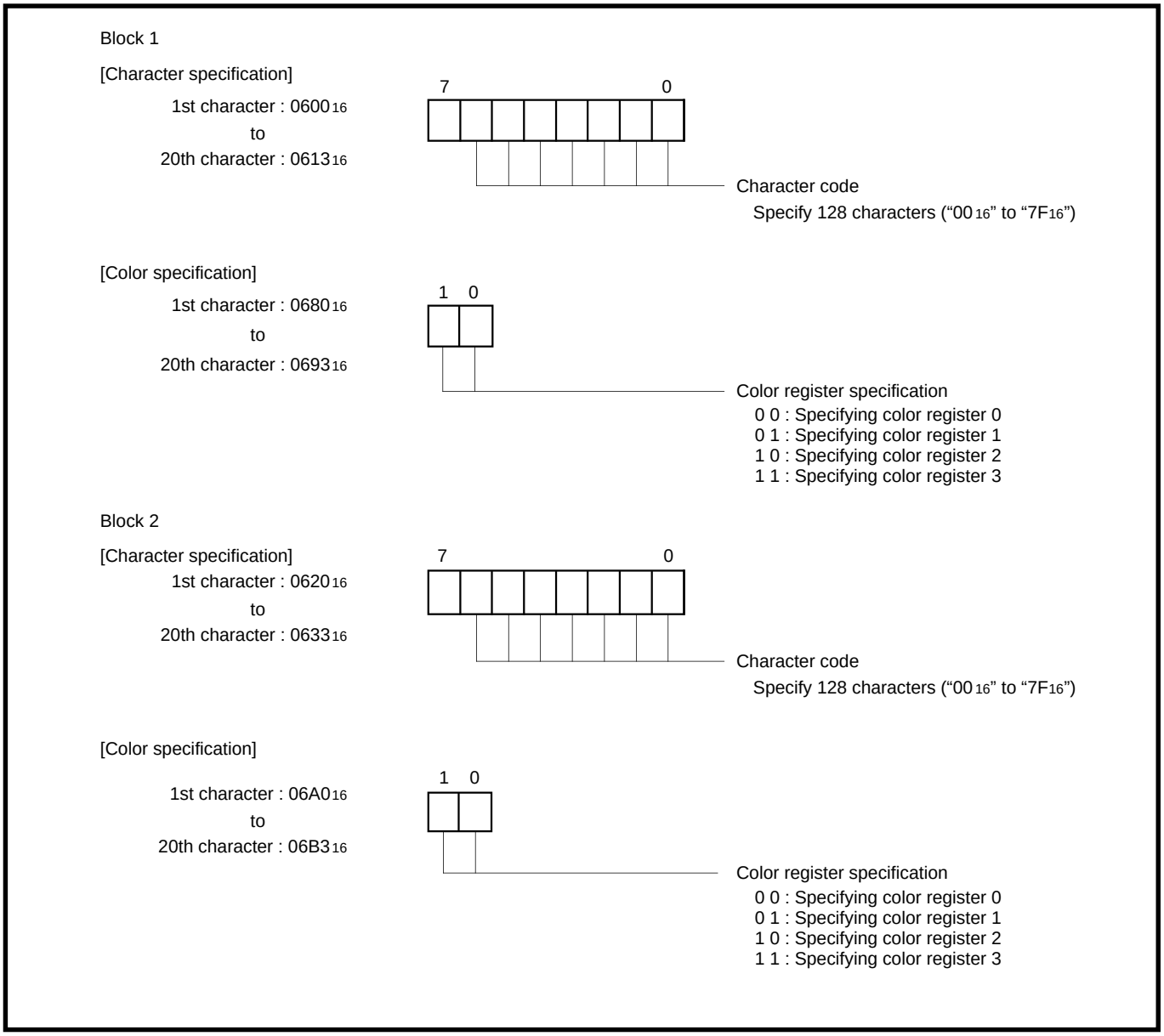


Fig. 4.5.12 Structure of CRT display RAM

4.5 Functional description

The different CRT display function–related registers from those of M37221M6-XXXSP/FP are shown in the following pages.

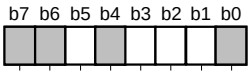
Border Selection Register

Border selection register (MD) [Address $00E5_{16}$]

B	Name	Functions	After reset	R	W
0	Block 1 OUT output border selection bit (MD10)	0 : Same output as character output 1 : Border output	Indeterminate	R	W
1	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
2	Block 1 OUT output border selection bit (MD20)	0 : Same output as character output 1 : Border output	Indeterminate	R	W
3 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 4.5.13 Border selection register (addresses $00E5_{16}$)

Color Register n

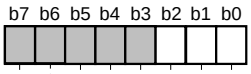


Color register n (COn) (n = 0 to 3) [Addresses $00E6_{16}$ to $00E9_{16}$]

B	Name	Functions	After reset	R	W
0	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
1	B signal output selection bit (COn1)	0 : No character is output 1 : Character is output	0	R	W
2	G signal output selection bit (COn2)	0 : No character is output 1 : Character is output	0	R	W
3	R signal output selection bit (COn3)	0 : No character is output 1 : Character is output	0	R	W
4	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
5	OUT signal output control bit (COn5)	0 : Character is output 1 : Blank is output	0	R	W
6, 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 4.5.14 Color register n (addresses $00E6_{16}$ to $00E9_{16}$)

CRT Control Register



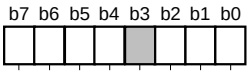
CRT control register (CC) [Address $00EA_{16}$]

B	Name	Functions	After reset	R	W
0	All-blocks display control bit (Note) (CC0)	0 : All-blocks display off 1 : All-blocks display on	0	R	W
1	Block 1 display control bit (CC1)	0 : Block 1 display off 1 : Block 1 display on	0	R	W
2	Block 2 display control bit (CC2)	0 : Block 2 display off 1 : Block 2 display on	0	R	W
3 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Note: Display is controlled by logical product (AND) between the all-blocks display control bit and each block control bit.

Fig. 4.5.15 CRT control register (address $00EA_{16}$)

CRT Port Control Register



CRT port control register (CRTP) [Address $00EC_{16}$]

B	Name	Functions	After reset	R	W
0	H_{SYNC} input polarity switch bit (HSYC)	0 : Positive polarity 1 : Negative polarity	0	R	W
1	V_{SYNC} input polarity switch bit (VSYC)	0 : Positive polarity 1 : Negative polarity	0	R	W
2	R, G, B output polarity switch bit (R/G/B)	0 : Positive polarity 1 : Negative polarity	0	R	W
3	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
4	OUT output polarity switch bit (OUT)	0 : Positive polarity 1 : Negative polarity	0	R	W
5	R signal output switch bit (OP5)	0 : R signal output 1 : MUTE signal output	0	R	W
6	G signal output switch bit (OP6)	0 : G signal output 1 : MUTE signal output	0	R	W
7	B signal output switch bit (OP7)	0 : B signal output 1 : MUTE signal output	0	R	W

Fig. 4.5.16 CRT port control register (address $00EC_{16}$)

4.5.7 Internal state immediately after reset

Figures 4.5.17 and 4.5.18 show the internal state immediately after reset.

■SFR Area (addresses $C0_{16}$ to DF_{16})

<State immediately after reset>

[0] : "0" immediately after reset

[1] : "1" immediately after reset

[?] : Undefined immediately after reset

Address	Register	State immediately after reset (b7 b6 b5 b4 b3 b2 b1 b0)
$C0_{16}$	Port P0 (P0)	?
$C1_{16}$	Port P0 direction register (D0)	00_{16}
$C2_{16}$	Port P1 (P1)	?
$C3_{16}$	Port P1 direction register (D1)	00_{16}
$C4_{16}$	Port P2 (P2)	?
$C5_{16}$	Port P2 direction register (D2)	00_{16}
$C6_{16}$	Port P3 (P3)	0 0 0 ? ? ? ? ?
$C7_{16}$	Port P3 direction register (D3)	00_{16}
$C8_{16}$		?
$C9_{16}$		?
CA_{16}	Port P5 (P5)	0 0 ? ? ? ? ? ?
CB_{16}	Port P5 direction register (D5)	00_{16}
CC_{16}		?
CD_{16}	Port P3 output mode control register (P3S)	00_{16}
CE_{16}	DA-H register (DA-H)	?
CF_{16}	DA-L register (DA-L)	0 0 ? ? ? ? ? ?
$D0_{16}$	PWM0 register (PWM0)	?
$D1_{16}$	PWM1 register (PWM1)	?
$D2_{16}$	PWM2 register (PWM2)	?
$D3_{16}$	PWM3 register (PWM3)	?
$D4_{16}$	PWM4 register (PWM4)	?
$D5_{16}$	PWM output control register 1 (PW)	00_{16}
$D6_{16}$	PWM output control register 2 (PN)	00_{16}
$D7_{16}$		?
$D8_{16}$		?
$D9_{16}$		?
DA_{16}		?
DB_{16}		?
DC_{16}	Serial I/O mode register (SM)	00_{16}
DD_{16}	Serial I/O regsiter (SIO)	?
DE_{16}	DA1 conversion register (DA1)	0 0 ? ? ? ? ? ?
DF_{16}	DA2 conversion register (DA2)	0 0 ? ? ? ? ? ?

Fig. 4.5.17 Internal state immediately after reset (1)

■SFR Area (addresses $E0_{16}$ to FF_{16})

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Undefined immediately after reset

Address	Register	State immediately after reset (b7 b6 b5 b4 b3 b2 b1 b0)
$E0_{16}$	Horizontal position register (HR)	00_{16}
$E1_{16}$	Vertical position register 1 (CV1)	0 ? ? ? ? ? ? ?
$E2_{16}$	Vertical position register 2 (CV2)	0 ? ? ? ? ? ? ?
$E3_{16}$		?
$E4_{16}$	Character size register (CS)	0 0 0 0 ? ? ? ?
$E5_{16}$	Border selection register (MD)	0 0 0 0 0 ? 0 ?
$E6_{16}$	Color register 0 (CO0)	00_{16}
$E7_{16}$	Color register 1 (CO1)	00_{16}
$E8_{16}$	Color register 2 (CO2)	00_{16}
$E9_{16}$	Color register 3 (CO3)	00_{16}
EA_{16}	CRT control register (CC)	00_{16}
EB_{16}		?
EC_{16}	CRT port control register (CRTP)	00_{16}
ED_{16}	CRT clock selection register (CK)	00_{16}
EE_{16}	A-D control register 1 (AD1)	0 0 0 ? 0 0 0 0
EF_{16}	A-D control register 2 (AD2)	00_{16}
$F0_{16}$	Timer 1 (TM1)	FF_{16}
$F1_{16}$	Timer 2 (TM2)	07_{16}
$F2_{16}$	Timer 3 (TM3)	FF_{16}
$F3_{16}$	Timer 4 (TM4)	07_{16}
$F4_{16}$	Timer 12 mode register (T12M)	00_{16}
$F5_{16}$	Timer 34 mode register (T34M)	00_{16}
$F6_{16}$	PWM5 register (PWM5)	?
$F7_{16}$		?
$F8_{16}$		?
$F9_{16}$	Interrupt input polarity register (RE)	0 0 0 0 0 0 0 ?
FA_{16}	Test register (TEST)	00_{16}
FB_{16}	CPU mode register (CPUM)	1 1 1 1 1 1 0 0
FC_{16}	Interrupt request register 1 (IREQ1)	00_{16}
FD_{16}	Interrupt request register 2 (IREQ2)	00_{16}
FE_{16}	Interrupt control register 1 (ICON1)	00_{16}
FF_{16}	Interrupt control register 2 (ICON2)	00_{16}

Fig. 4.5.18 Internal state immediately after reset (2)

4.6 Electrical characteristics

Absolute maximum ratings

Symbol	Parameter		Conditions	Ratings	Unit
V_{CC}	Power source voltage	V_{CC}	All voltages are based on V_{SS}. Output transistors are cut off.	–0.3 to 6	V
V_I	Input voltage	CNV_{SS}		–0.3 to 6	V
V_I	Input voltage	$P0_0$–$P0_7$,$P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_4$, OSC1, X_{IN}, H_{SYNC}, V_{SYNC}, $\overline{RESET}$		–0.3 to V_{CC} + 0.3	V
V_O	Output voltage	$P0_6$, $P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_2$, R, G, B, OUT, D-A, X_{OUT}, OSC2		–0.3 to V_{CC} + 0.3	V
V_O	Output voltage	$P0_0$–$P0_5$		–0.3 to 13	V
I_{OH}	Circuit current	R, G, B, OUT, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$, $P3_1$, D-A		0 to 1 (Note 1)	mA
I_{OL1}	Circuit current	R, G, B, OUT1, $P0_6$, $P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_3$, $P3_0$–$P3_2$, D-A		0 to 2 (Note 2)	mA
I_{OL2}	Circuit current	$P0_0$–$P0_5$		0 to 1 (Note 2)	mA
I_{OL3}	Circuit current	$P2_4$–$P2_7$		0 to 10 (Note 3)	mA
P_d	Power dissipation			550	mW
T_{opr}	Operating temperature		T_a = 25 °C	–10 to 70	°C
T_{stg}	Storage temperature			–40 to 125	°C

Notes 1: The total current that flows out of the IC must be 20 mA (max.).
2: The total input current to IC (I_{OL1} + I_{OL2}) must be 30 mA or less.
3: The total average input current for ports $P2_4$–$P2_7$ to IC must be 20 mA or less.

Recommended operating conditions (T_a = –10 °C to 70 °C, V_{CC} = 5 V ± 10 %, unless otherwise noted)

Symbol	Parameter		Min.	Typ.	Max.	Unit
V_{CC}	Power source voltage (Note 4), During CPU, CRT operation		4.5	5.0	5.5	V
V_{SS}	Power source voltage		0	0	0	V
V_{IH}	HIGH input voltage	$P0_0$–$P0_7$,$P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_4$, S_{IN}, S_{CLK}, H_{SYNC}, V_{SYNC}, $\overline{RESET}$, X_{IN}, OSC1, TIM2, TIM3, INT1, INT2, INT3	0.8V_{CC}		V_{CC}	V V
V_{IL1}	LOW input voltage	$P0_0$–$P0_7$,$P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_4$	0		0.4V_{CC}	V
V_{IL2}	LOW input voltage	H_{SYNC}, V_{SYNC}, $\overline{RESET}$, TIM2, TIM3, INT1, INT2, INT3, X_{IN}, OSC1, S_{IN}, S_{CLK}	0		0.2V_{CC}	V
I_{OH}	HIGH average output current (Note 1)	R, G, B, OUT, D-A, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$, $P3_1$			1	mA
I_{OL1}	LOW average output current (Note 2)	R, G, B, OUT, D-A, $P0_6$, $P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_2$			2	mA
I_{OL2}	LOW average output current (Note 2)	$P0_0$–$P0_5$			1	mA
I_{OL3}	LOW average output current (Note 3)	$P2_4$–$P2_7$			10	mA
f(X_{IN})	Oscillation frequency (for CPU operation) (Note 5)	X_{IN}	7.9	8.0	8.1	MHz
f_{CRT}	Oscillation frequency (for CRT display) (Note 5)	OSC1	5.0		8.0	MHz
f_{hs1}	Input frequency	TIM2, TIM3			100	kHz
f_{hs2}	Input frequency	S_{CLK}			1	MHz

Notes 1: The total current that flows out of the IC must be 20 mA (max.).

2: The total input current to IC (I_{OL1} + I_{OL2}) must be 30 mA or less.

3: The total average input current for ports $P2_4$–$P2_7$ to IC must be 20 mA or less.

4: Connect 0.1 μF or more capacitor externally across the power source pins V_{CC}–V_{SS} so as to reduce power source noise. Also connect 0.1 μF or more capacitor externally across the pins V_{CC}–CNV_{SS}.

5: Use a quartz-crystal oscillator or a ceramic resonator for the CPU oscillation circuit.

4.6 Electrical characteristics

Electric characteristics (V_{CC} = 5 V ± 10 %, V_{SS} = 0 V, f(X_{IN}) = 8 MHz, T_a = –10 °C to 70 °C, unless otherwise noted)

Symbol	Parameter		Test conditions		Limits			Unit
					Min.	Typ.	Max.	
I_{CC}	Power source current	System operation	V_{CC} = 5.5 V, f(X_{IN}) = 8 MHz	CRT OFF		20	40	mA
				CRT ON		30	60	
		Stop mode	V_{CC} = 5.5 V, f(X_{IN}) = 0				300	μA
V_{OH}	HIGH output voltage	R, G, B, OUT, D-A, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$, $P3_1$	V_{CC} = 4.5 V, I_{OH} = –0.5 mA		2.4			V
V_{OL}	LOW output voltage	R, G, B, OUT, D-A, $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_3$, $P3_0$–$P3_2$	V_{CC} = 4.5 V, I_{OL} = 0.5 mA				0.4	V
	LOW output voltage	$P2_4$–$P2_7$	V_{CC} = 4.5 V, I_{OL} = 10.0 mA				3.0	
V_{T+}–V_{T-}	Hysteresis $\overline{\text{RESET}}$		V_{CC} = 5.0 V			0.5	0.7	V
	Hysteresis (Note)	H_{SYNC}, V_{SYNC}, TIM2, TIM3, INT1, INT2, INT3, S_{IN}, S_{CLK}	V_{CC} = 5.0 V			0.5	1.3	
I_{IZH}	HIGH input leak current	$\overline{\text{RESET}}$, $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_4$, H_{SYNC}, V_{SYNC}	V_{CC} = 5.5 V, V_I = 5.5 V				5	μA
I_{IZL}	LOW input leak current	$\overline{\text{RESET}}$, $P0_0$–$P0_7$, $P1_0$–$P1_7$, $P2_0$–$P2_7$, $P3_0$–$P3_4$, H_{SYNC}, V_{SYNC}	V_{CC} = 5.5 V, V_I = 0 V				5	μA
I_{OZH}	HIGH output leak current	$P0_0$–$P0_5$	V_{CC} = 5.5 V, V_O = 12 V				10	μA

Note: $P0_6$, $P0_7$, $P1_5$, $P2_3$ and $P2_4$ have the hysteresis when these pins are used as interrupt input pins or timer input pins. $P2_0$–$P2_2$ have the hysteresis when these pins are used as serial I/O pins.

A-D Comparator characteristics

(V_{CC} = 5 V ± 10 %, V_{SS} = 0 V, f(X_{IN}) = 8 MHz, T_a = –10 °C to 70 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
—	Resolution				6	bits
—	Absolute accuracy		0	±1	±2	LSB

Note: When V_{CC} = 5 V, 1 LSB = 5/64 V.

D-A Converter characteristics

(V_{CC} = 5 V ± 10 %, V_{SS} = 0 V, f(X_{IN}) = 8 MHz, T_a = –10 °C to 70 °C, unless otherwise noted)

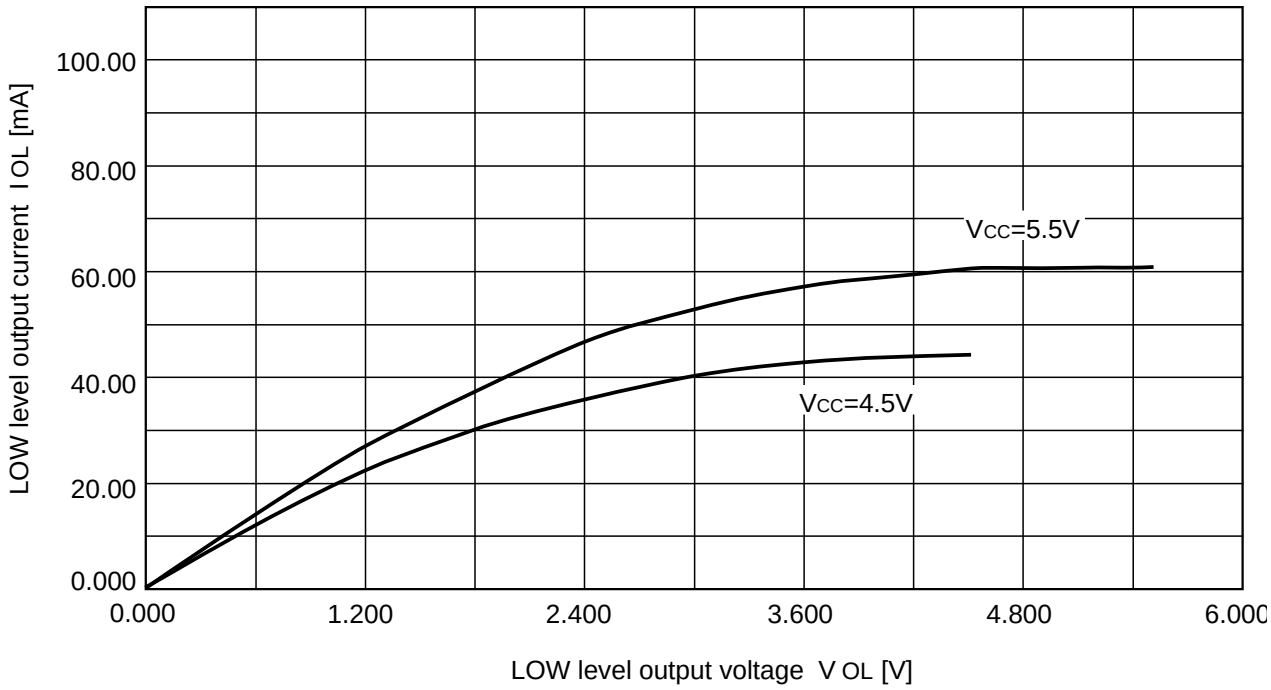
Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
—	Resolution				6	bits
—	Absolute accuracy				2	%
t_{su}	Setting time				3	μs
R_O	Output resistor		1	2.5	4	kΩ

4.7 Standard characteristics

The data described in this section are characteristic examples. Refer to **"4.6 Electrical characteristics"** for rated values.

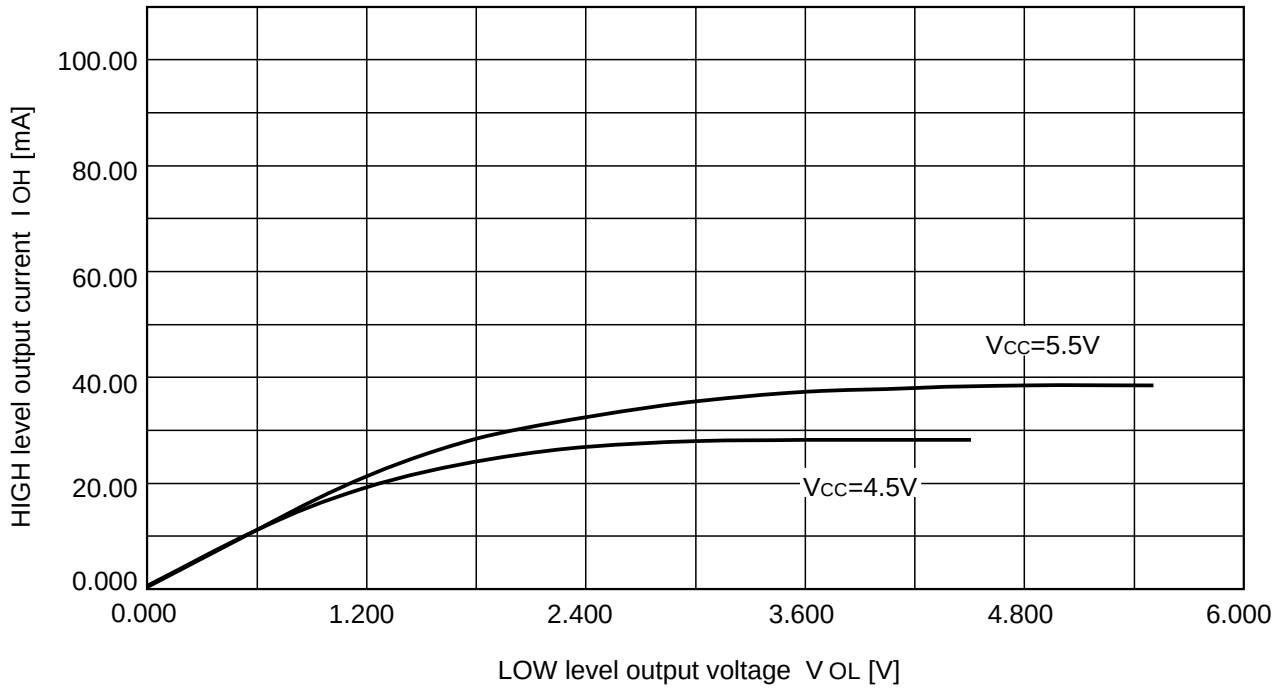
1. Ports $P0_0$–$P0_5$ and $P3_2$

(a) I_{OL}–V_{OL} characteristics



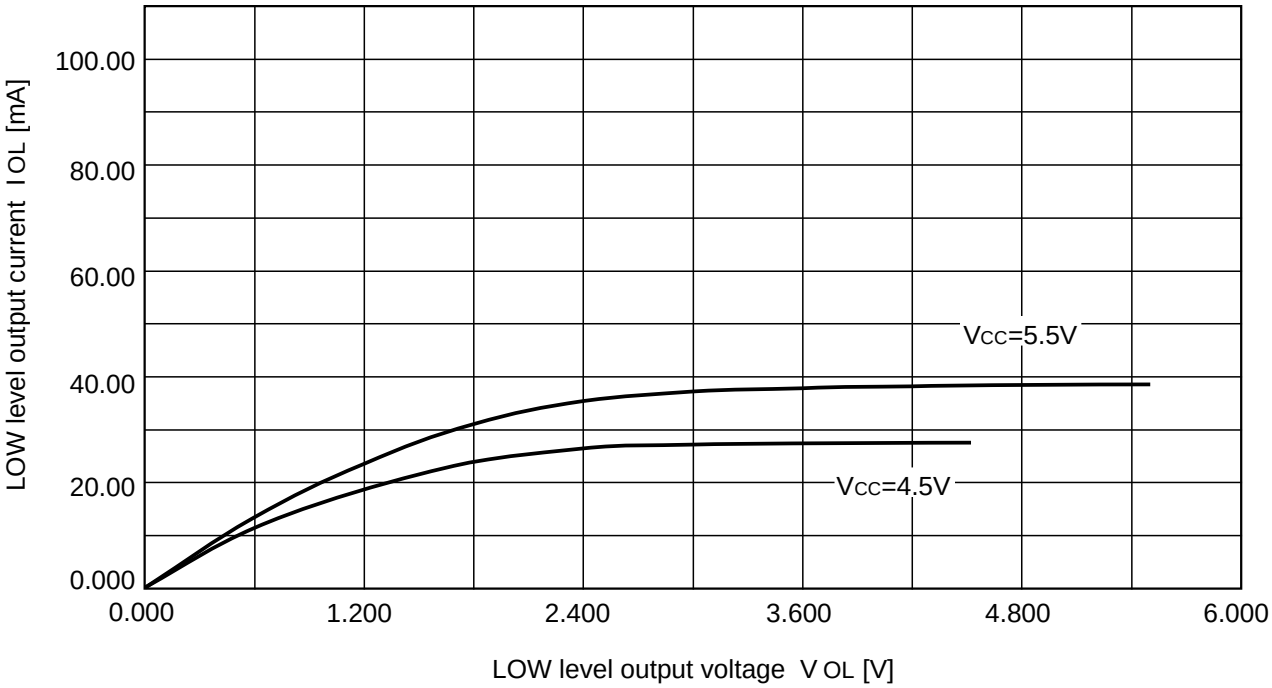
2. Ports $P0_6$ and $P0_7$

(a) I_{OH}–V_{OL} characteristics

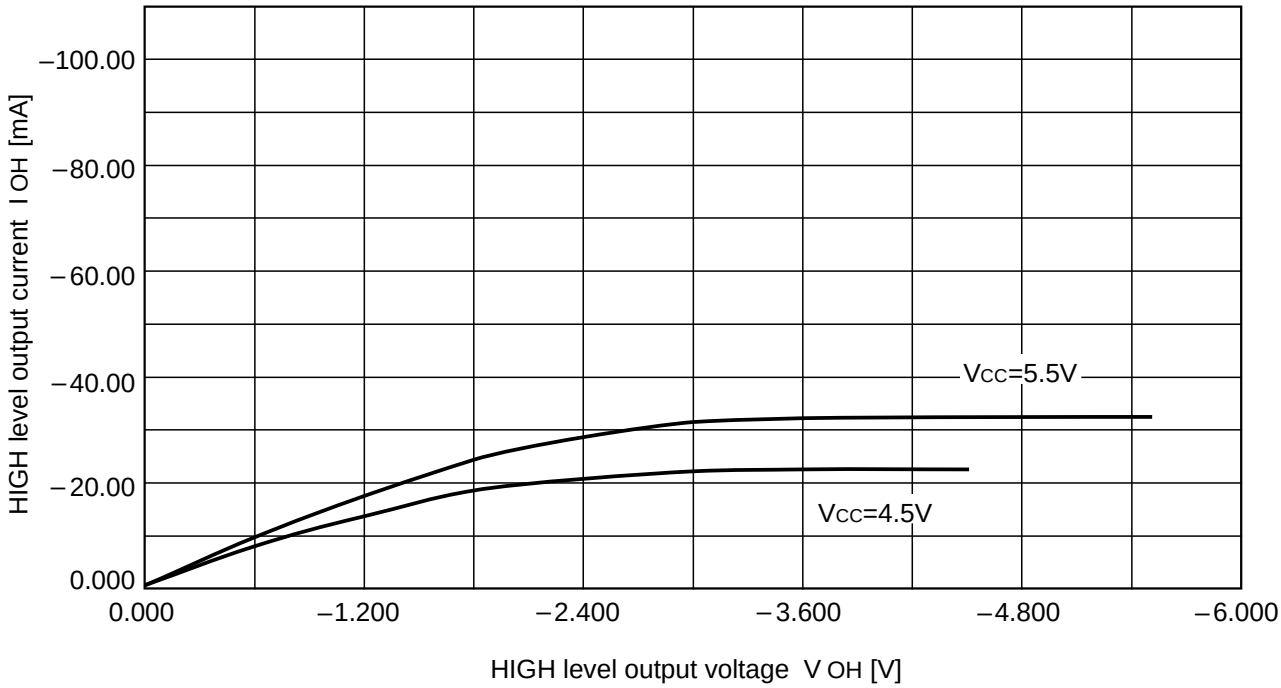


3. Ports $P1_0$–$P1_7$, $P2_0$–$P2_3$, $P3_0$, $P3_1$ and D-A

(a) I_{OL}–V_{OL} characteristics



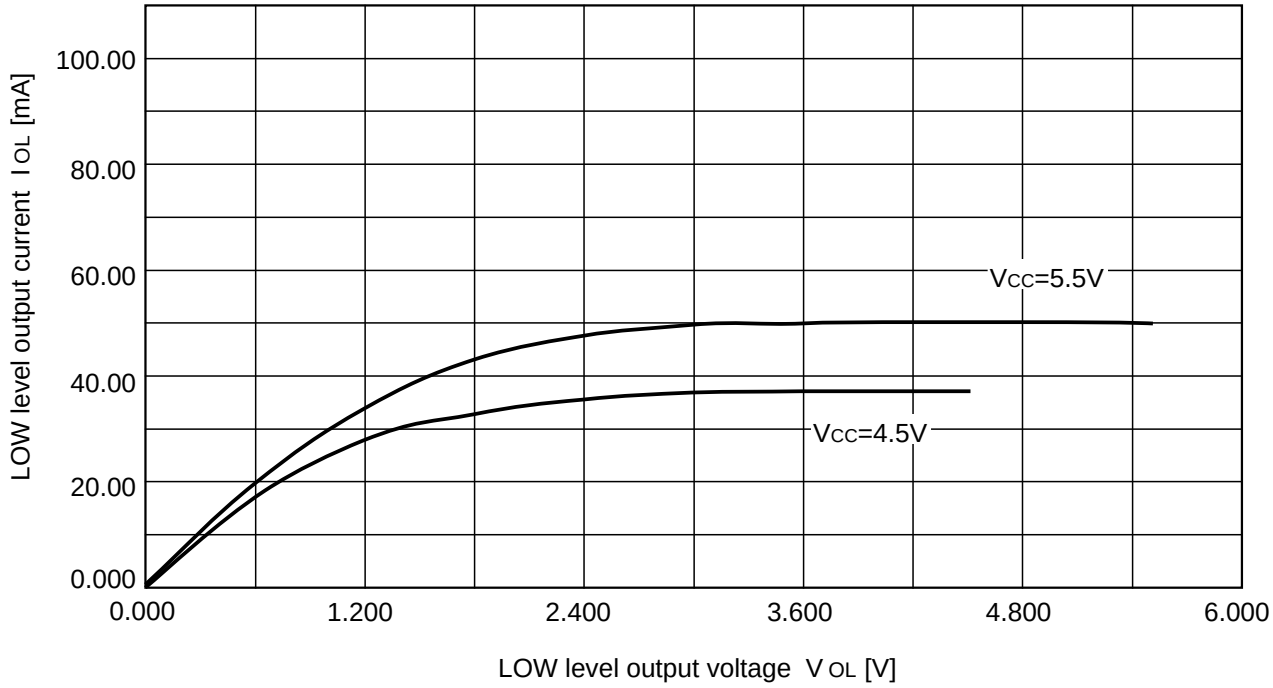
(b) I_{OH}–V_{OH} characteristics



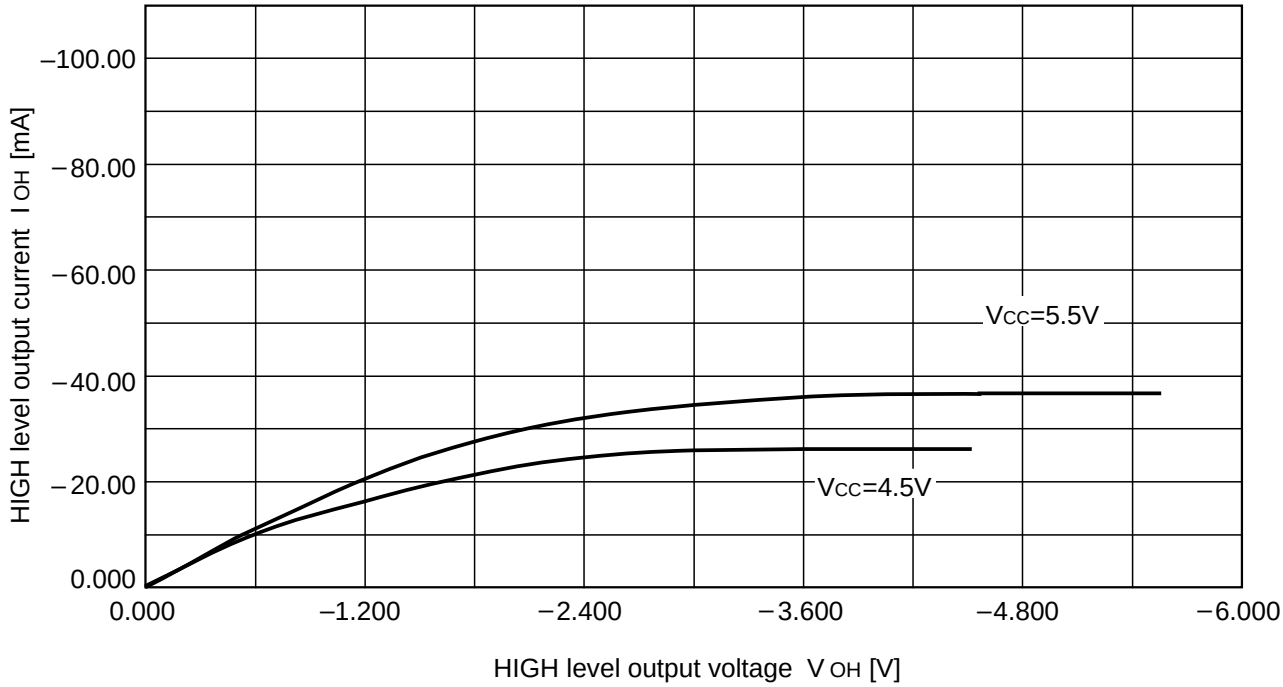
4.7 Standard characteristics

4. Ports $P2_4$–$P2_7$

(a) I_{OL}–V_{OL} characteristics

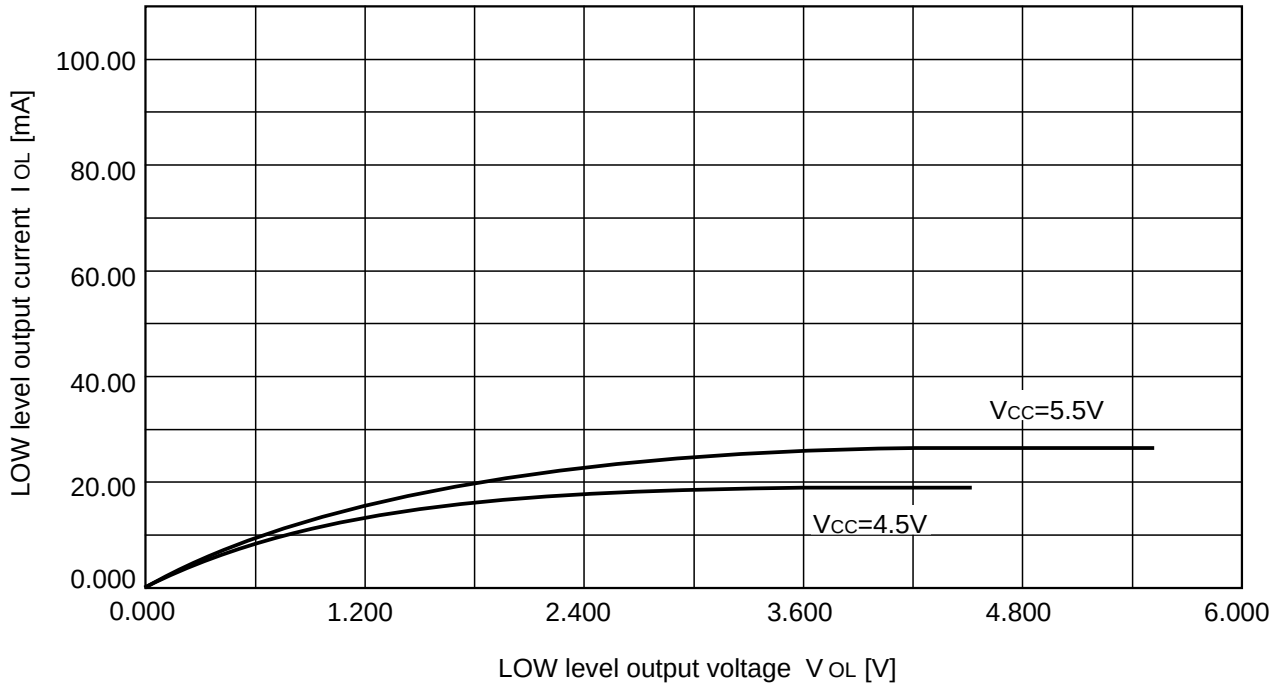


(b) I_{OH}–V_{OH} characteristics

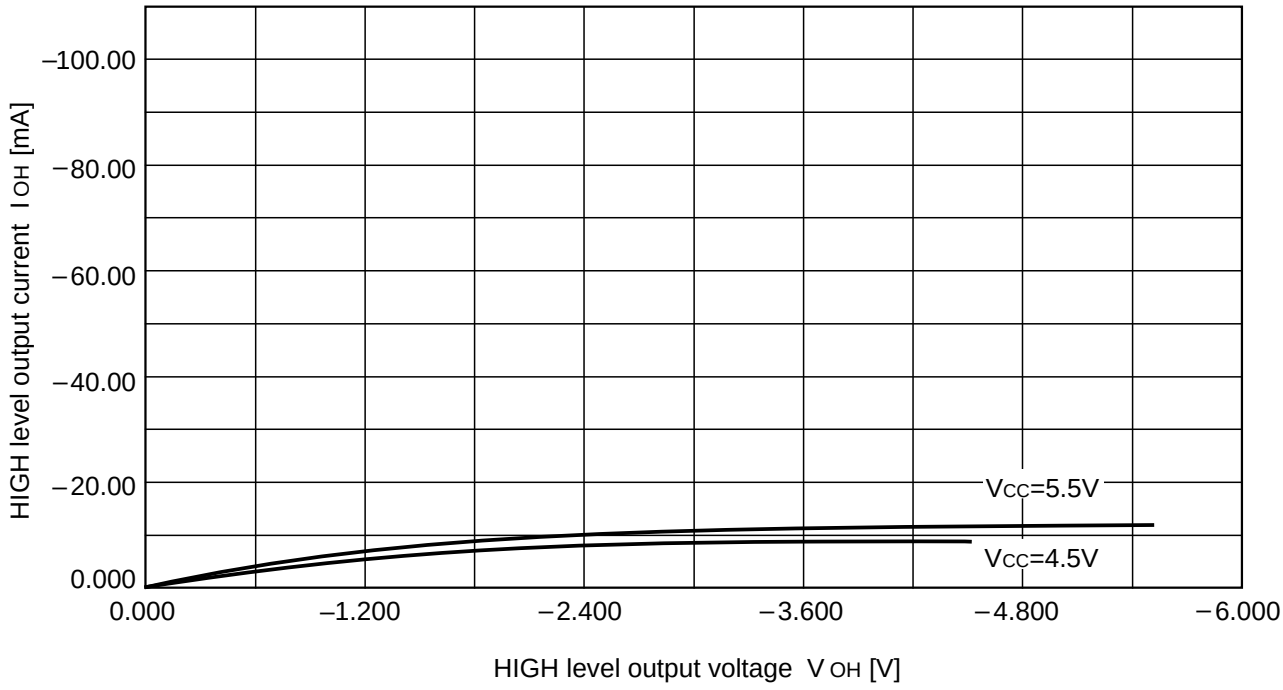


5. Ports P52–P55

(a) I_{OL}–V_{OL} characteristics



(b) I_{OH}–V_{OH} characteristics



CHAPTER 5
APPLICATION

5.1 Example of multi-line display

The M37221Mx-XXXSP/FP is used as a general example in describing this application for the 7220 group. The M377221Mx-XXXSP/FP ordinarily displays 2 lines on a CRT screen by displaying 2 blocks at different vertical positions. In addition to this, it can display 3 lines or more (multi-line display) by rewriting both character data and display positions during interrupt processing, using CRT interrupts. An example of the software processing for implemention this multi-line display is described below. This example is 12-line multi-line display using blocks 1 and 2.
For CRT display details, refer to **"2.11 CRT display function."**

5.1.1 Specifications

- Pins required: R, G, B, OUT1, H_{SYNC}, and V_{SYNC}
- H_{SYNC}/V_{SYNC} input polarity: positive polarity input
- R/G/B/OUT1 output polarity: positive polarity output
- Character colors: red, blue, white, and cyan
- No character background color
- Bordering (OUT) is available
- Character size: minimum size
- 12-line display

5.1.2 Connection example

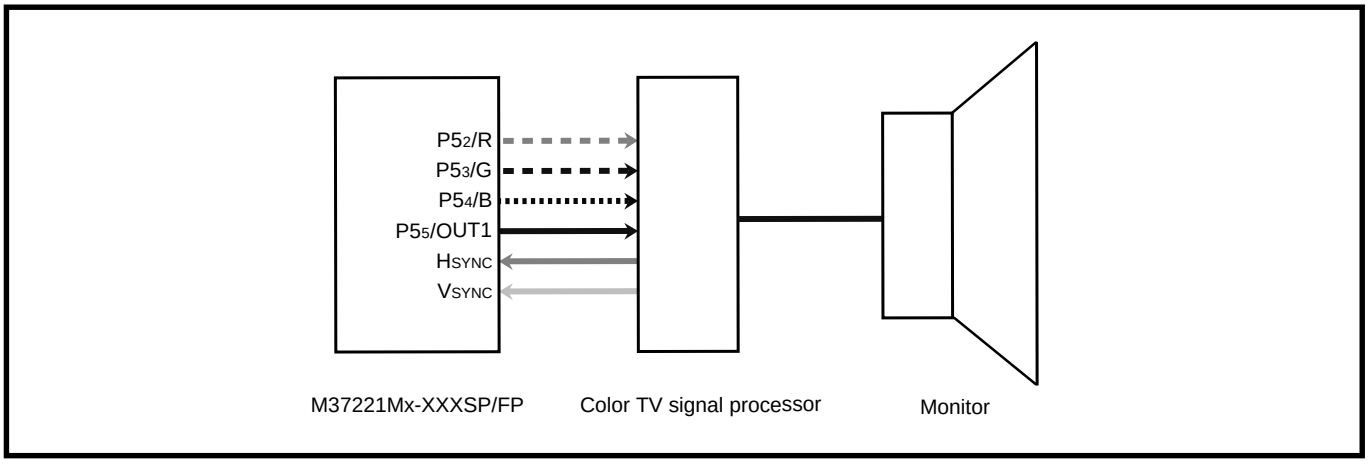


Fig. 5.1.1 Connection example

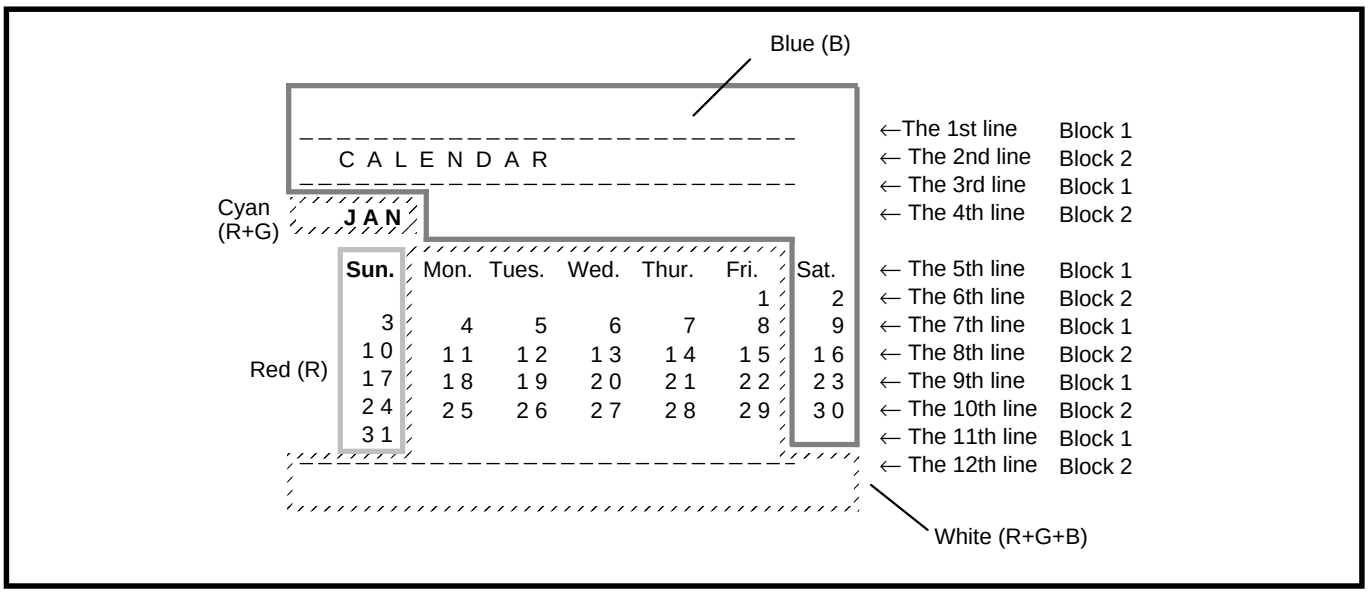


Fig. 5.1.2 Display example

5.1.3 General flowchart

The multi-line display processing routine consists of initialization processing routine, V_{SYNC} interrupt processing routine, and CRT interrupt processing routine.

(1) Initialization processing routine

This routine is used to initialize to cause a CRT interrupt.

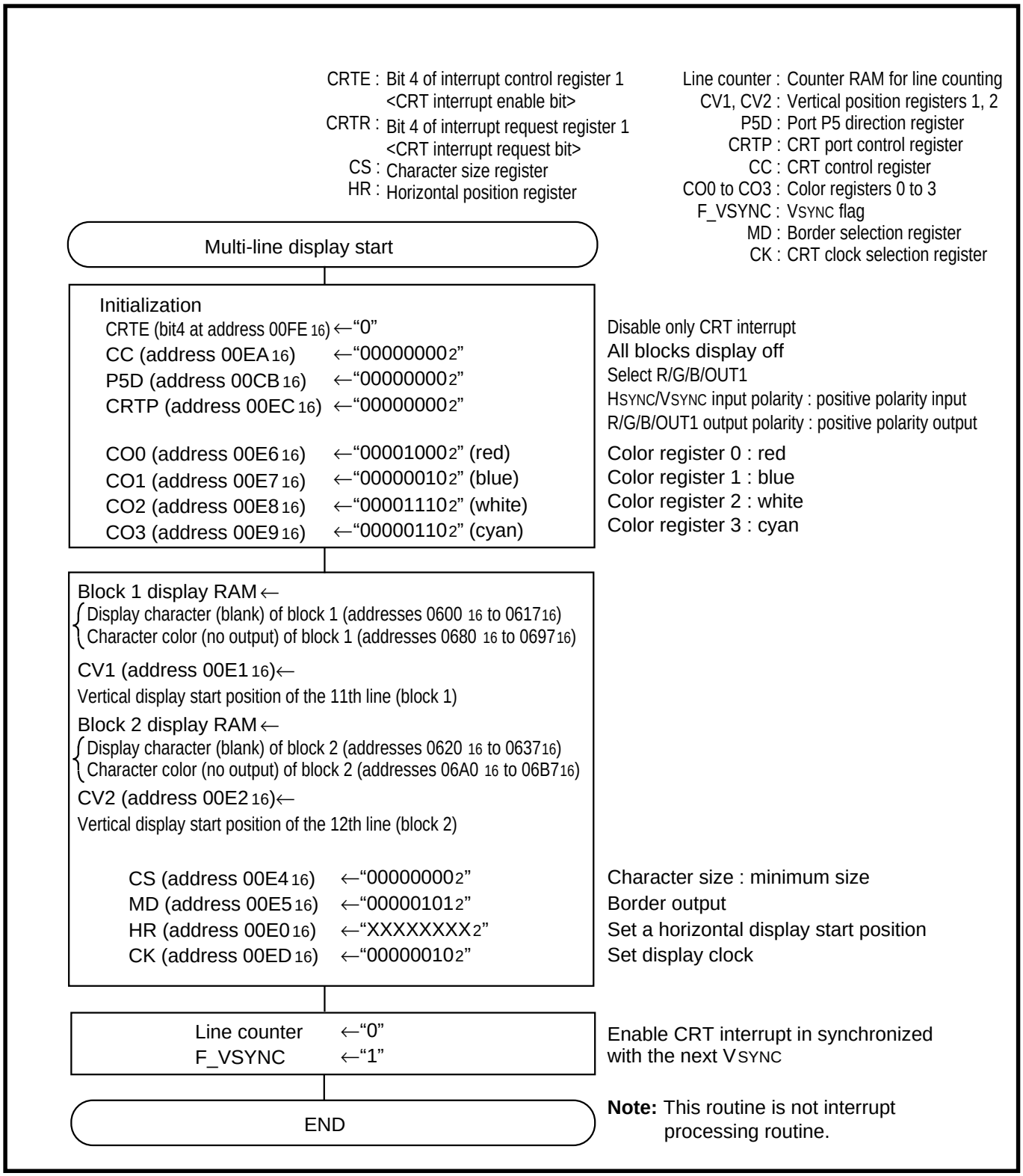


Fig. 5.1.3 Flowchart of initialization processing routine

(2) V_{SYNC} interrupt processing routine

The V_{SYNC} interrupt processing routine consists of; multi-line display start processing and multi-line display correction processing. The correction processing corrects erroneous multi-line display due to various influences.

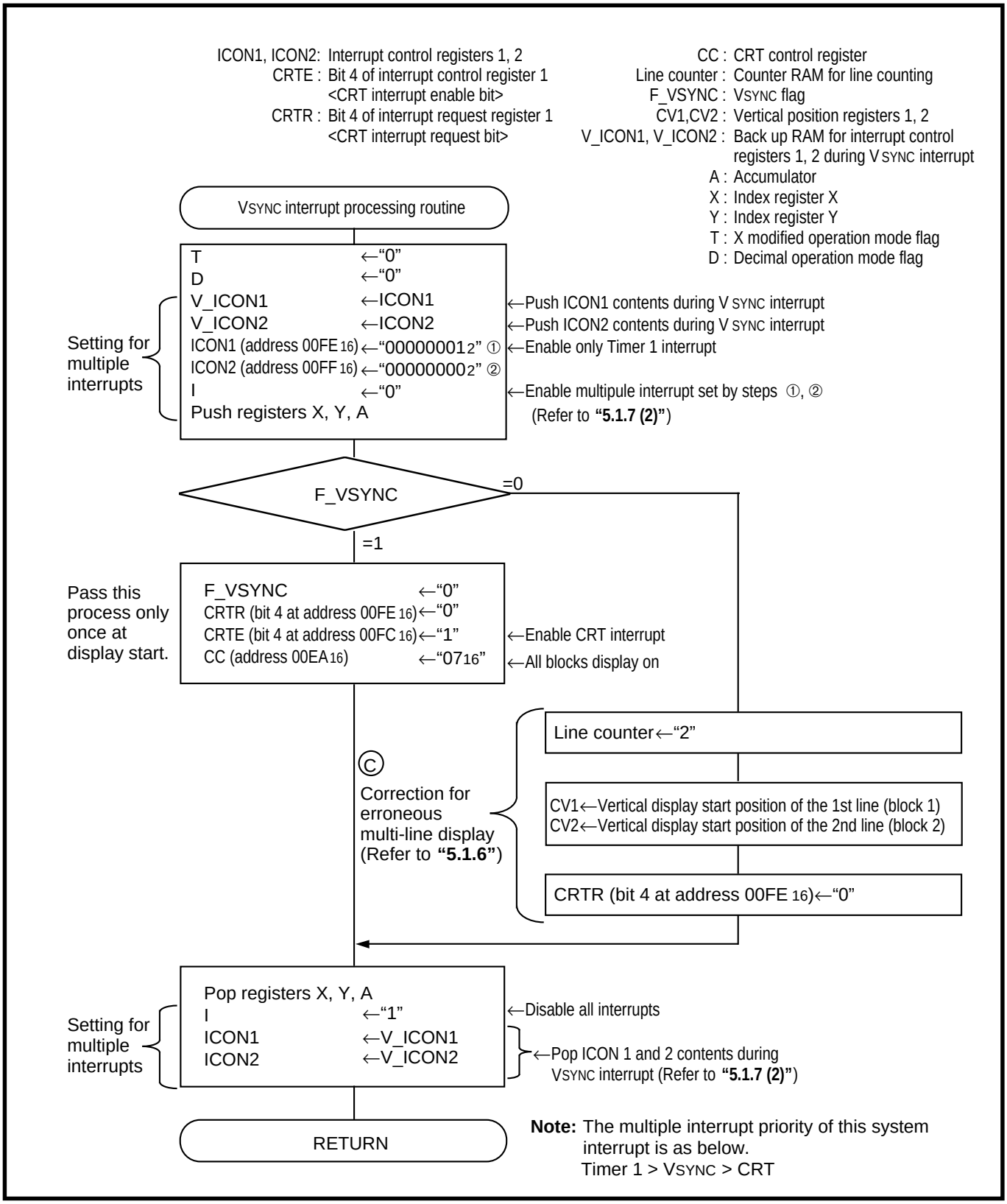


Fig. 5.1.4 Flowchart of V_{SYNC} interrupt processing routine

(3) CRT interrupt processing routine

The CRT interrupt processing routine executes the display character data setup routine for each line, in order to perform multi-line display. The line to be displayed is determined by the line counter value.

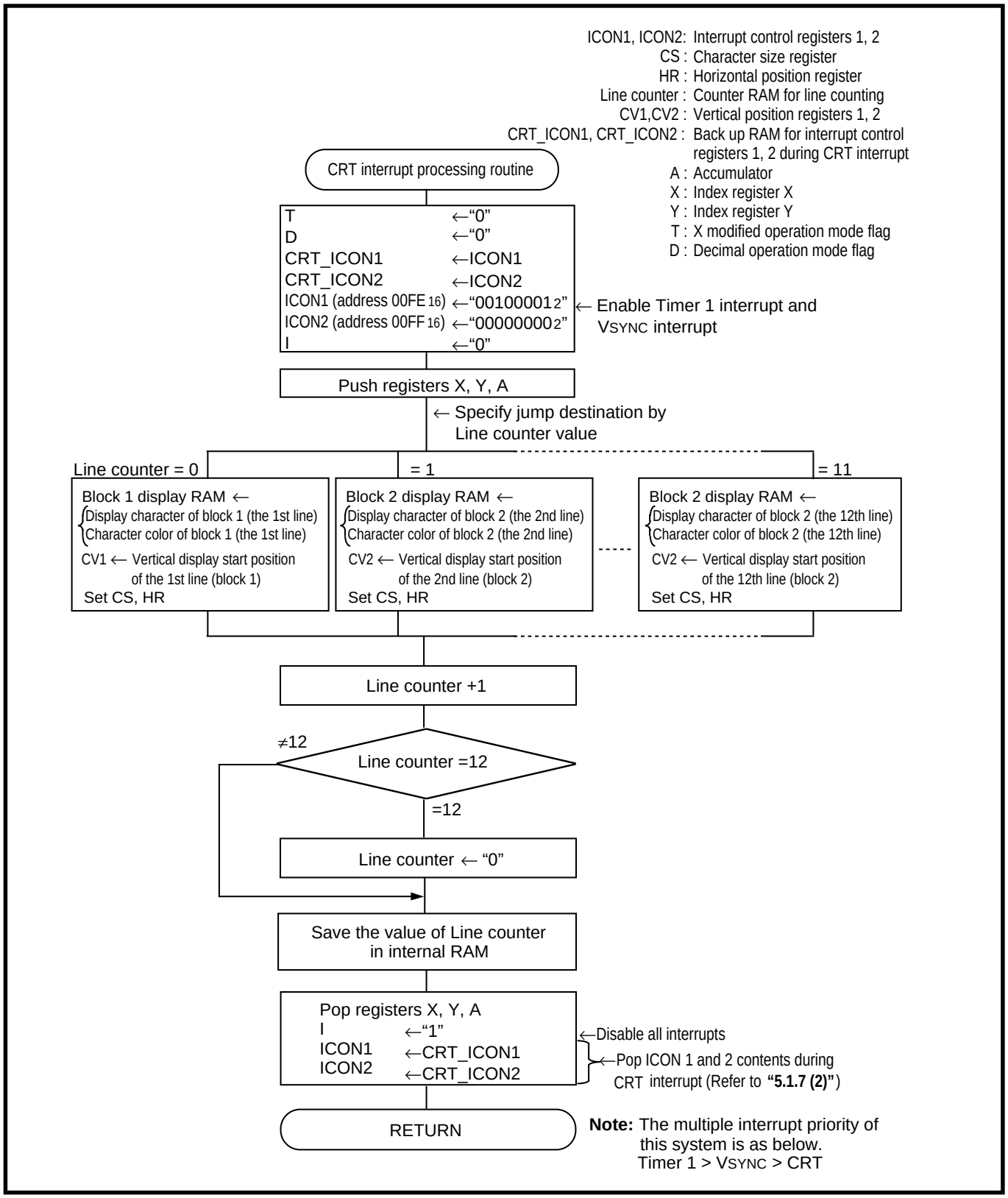


Fig. 5.1.5 Flowchart of CRT interrupt processing routine

5.1.4 Set of display character data

To display the character data, set the character codes ("00_{16}" to "FF_{16}") in the character addresses (block 1: addresses 0600_{16} to 0617_{16}, block 2: addresses 0620_{16} to 0637_{16}). Also, set the color register specifying ("00_2" to "11_2") in the color addresses (block 1: addresses 0680_{16} to 0697_{16}, block 2: addresses $06A0_{16}$ to $06B7_{16}$).

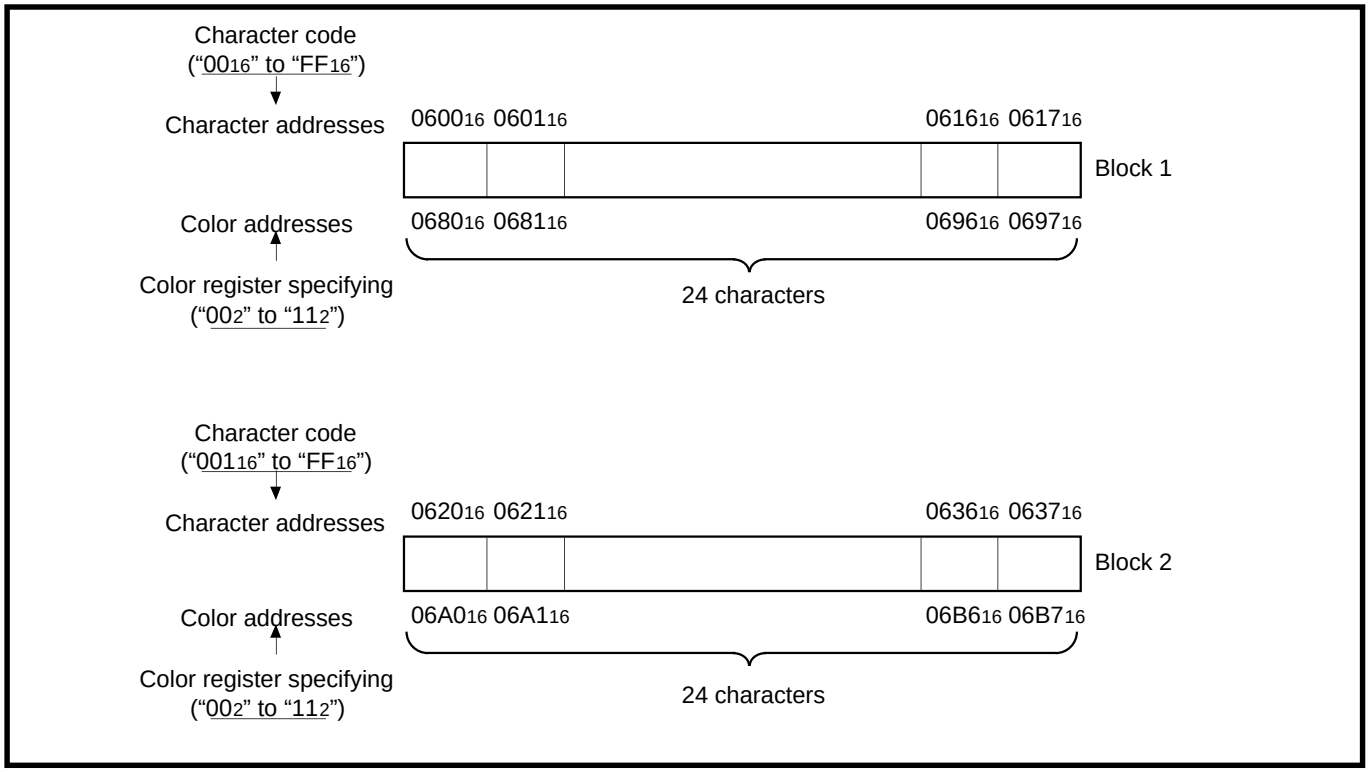


Fig. 5.1.6 Set of display character data

5.1.5 Line counter

The line counter determines which line of display data is to be set. For example, if a CRT interrupt occurs at the end of the first line display, the line counter value will be “2.” Therefore, the 3rd line display data must be set from the end of the 1st line display to the start of the 3rd line display. Then, the line counter value is incremented and becomes “3.”

Figure 5.1.7 shows the example of the setup timing for the line counter and the display character data.

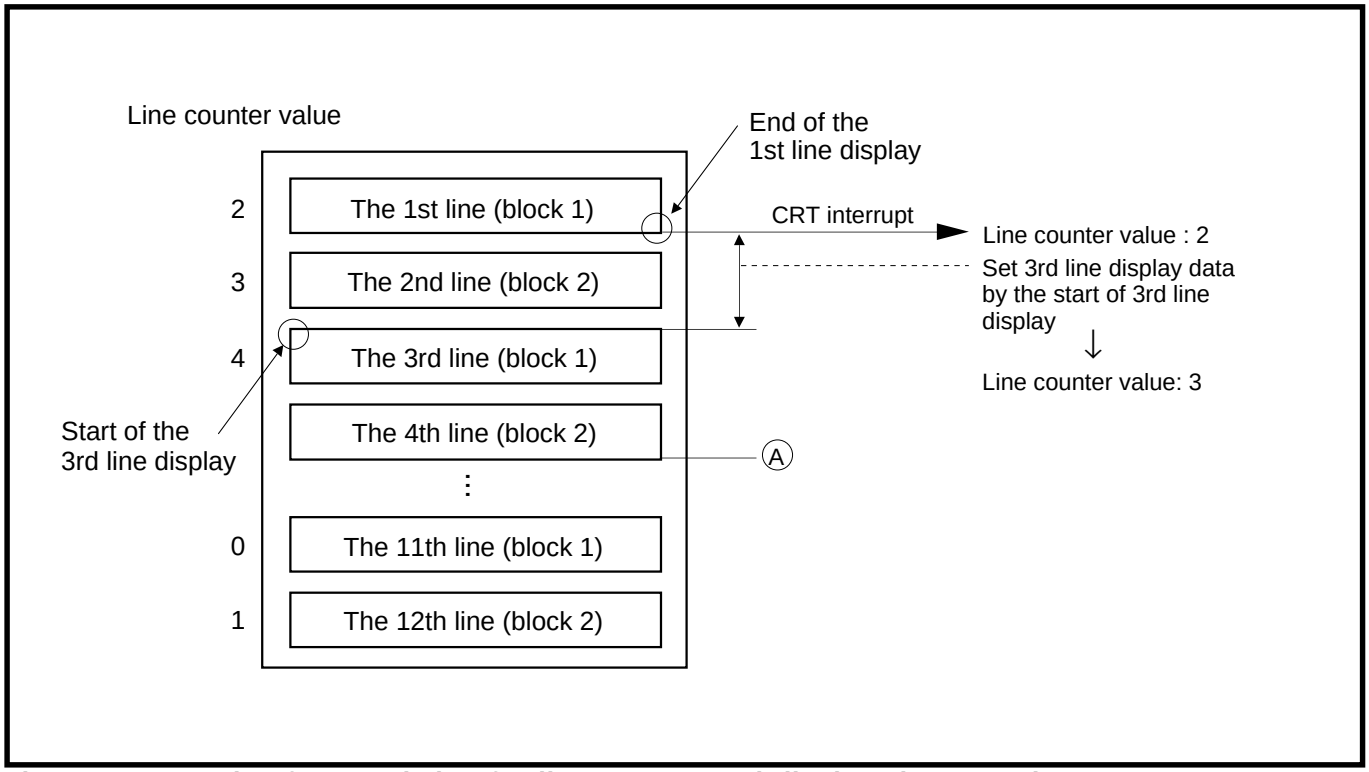


Fig. 5.1.7 Example of setup timing for line counter and display character data

5.1.6 Processing time

When setting display data by a CRT interrupt, the processing time is limited.

As shown in Figure 5.1.7, a CRT interrupt occurs at the end of the first line (block 1) display and setting for the 3rd line display is started. This setting must be completed before a scanning line reaches to the 3rd line display position. If the setting is not completed, display characters flicker or rewriting is looked. And also, for multi-line display of 12 lines, be sure that CRT interrupts occur 12 times from a V_{SYNC} to the next V_{SYNC}. If CRT interrupts do not occur as many times as the number of display lines, the following causes can be assumed.

- Display position overlaps.
- CRT interrupt processing time is too long, resulting in no display of that line (2 lines after the line being displayed).

For example, a CRT interrupt occurs at the end of the second line display in Figure 5.1.7. Within this interrupt processing, setting for the 4th line display is completed. However, if a scanning line is over the display position of the 4th line (that is, Ⓐ in Figure 5.1.7) during this setting, one CRT interrupt request is deleted (or does not occur). Therefore, the line counter value is disordered and multi-line display is not displayed correctly. In such cases, due to whatever causes, correct the value with processing Ⓒ of V_{SYNC} interrupt processing (refer to **"Figure 5.1.4"**). When the CRT interrupt software processing overtime causes this state, change the display positions or shorten the CRT interrupt software processing time.

5.1.7 Set of multiple interrupts

(1) When not setting multiple interrupts

When two or more interrupt requests occur at the same sampling point, the interrupt with the higher priority (refer to **"2.5 Interrupts, Table 2.5.1"**) is received. This priority level is determined by hardware but various priority processing by software can be executed using the interrupt enable bit and each interrupt disable flag (I).

Assume, for example, that all interrupts (CRT, V_{SYNC}, Timer 1) are enabled. When the multiple interrupt is not set, these interrupt request bits are set to "1" and the interrupts are determined by hardware as follows:

①CRT interrupt
②V_{SYNC} interrupt
③Timer 1 interrupt

Figure 5.1.8 shows the timing of interrupt processing when not setting multiple interrupts.

The I flag is set to "1" (all interrupts are disabled) automatically by hardware as soon as the interrupt processing starts. Unless the I flag is cleared to "0," other interrupt will not occur during the interrupt processing.

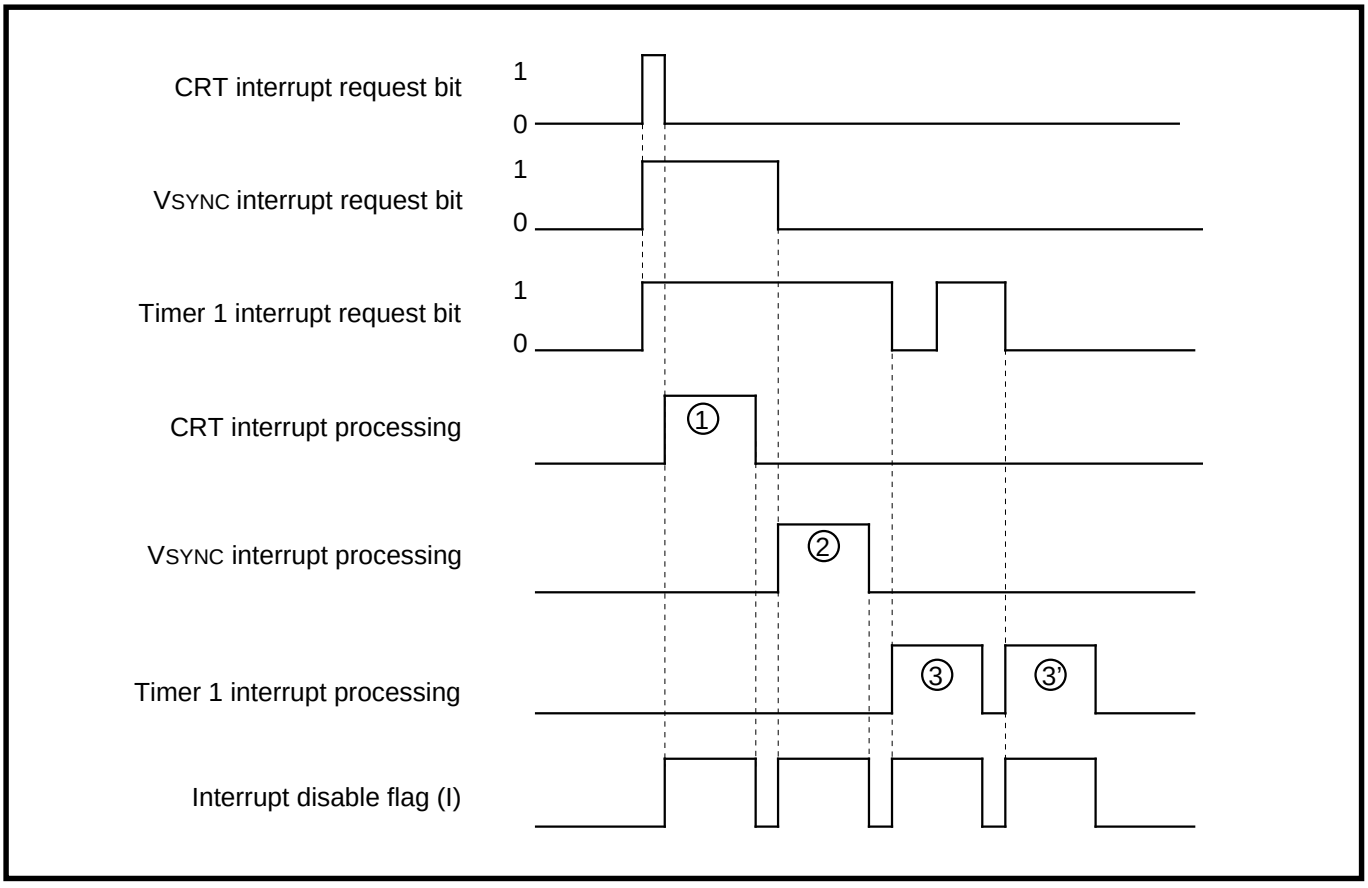


Fig. 5.1.8 Timing of interrupt processing when not setting multiple interrupts

(2) When setting multiple interrupts

Various priority processings are executed by enabling multiple interrupts and by setting priorities by software. For example, to set the priority listed below;

①Timer 1 interrupt
②V_{SYNC} interrupt
③CRT interrupt

execute the following process:

Set only interrupt enable bits (ICON1, ICON2) whose priorities are higher than the current interrupt, and enable the interrupt disable flag (I) in only the current interrupt processing routine.

Figure 5.1.9 shows the timing when all interrupt request bits (CRT, V_{SYNC}, Timer 1) are "1" at the same sampling point.

Note: When setting multiple interrupts, be sure to determine priority levels to prevent occurrence of plural interrupts with the same priority level.

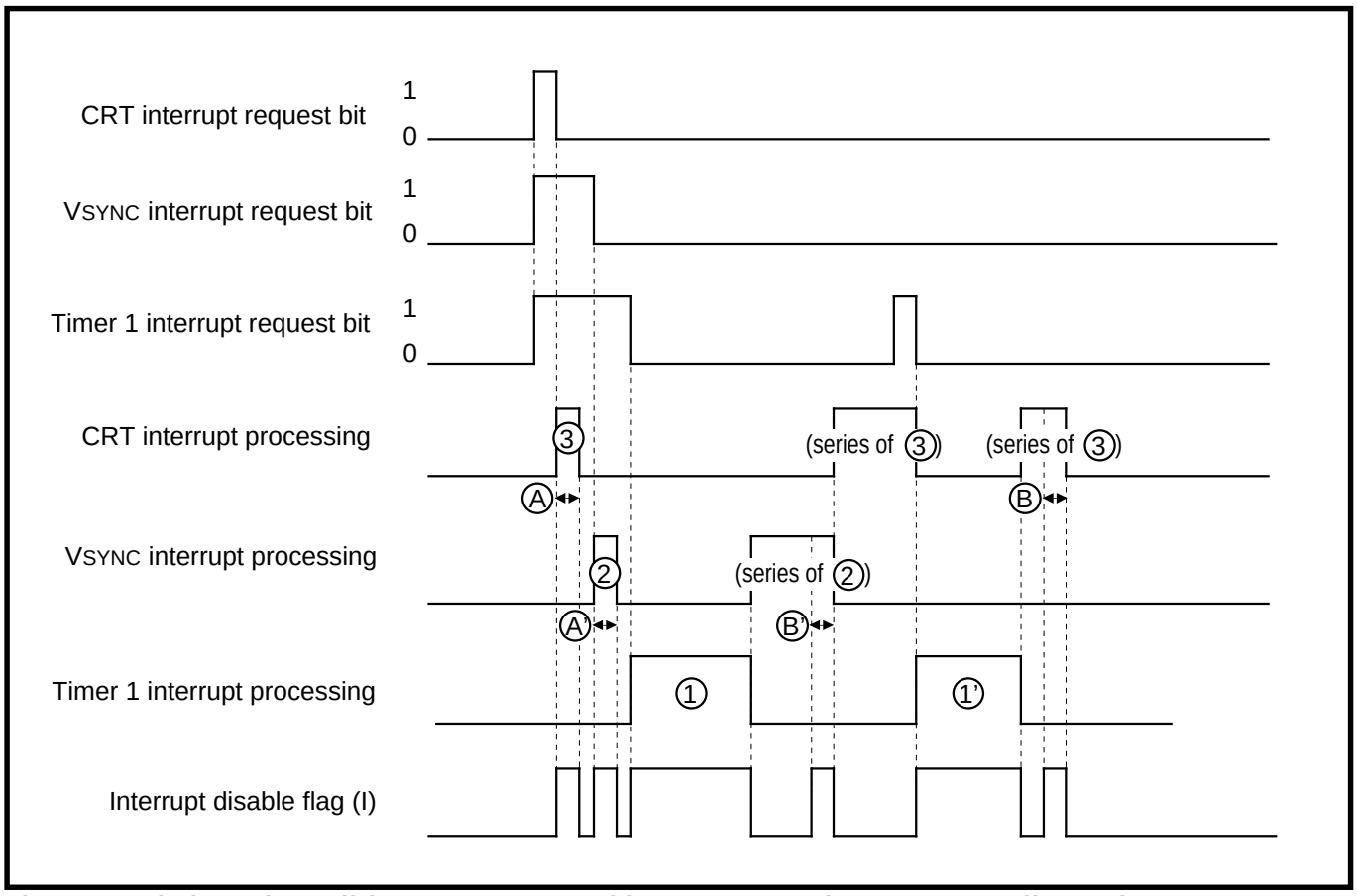


Fig. 5.1.9 Timing when all interrupt request bits are "1" at the same sampling point

(3) CRT interrupt processing routine when setting multiple interrupts

Figure 5.1.10 shows the flowchart of CRT interrupt processing routine when setting multiple interrupts. Ⓐ and Ⓑ are the setting routines for multiple interrupts.

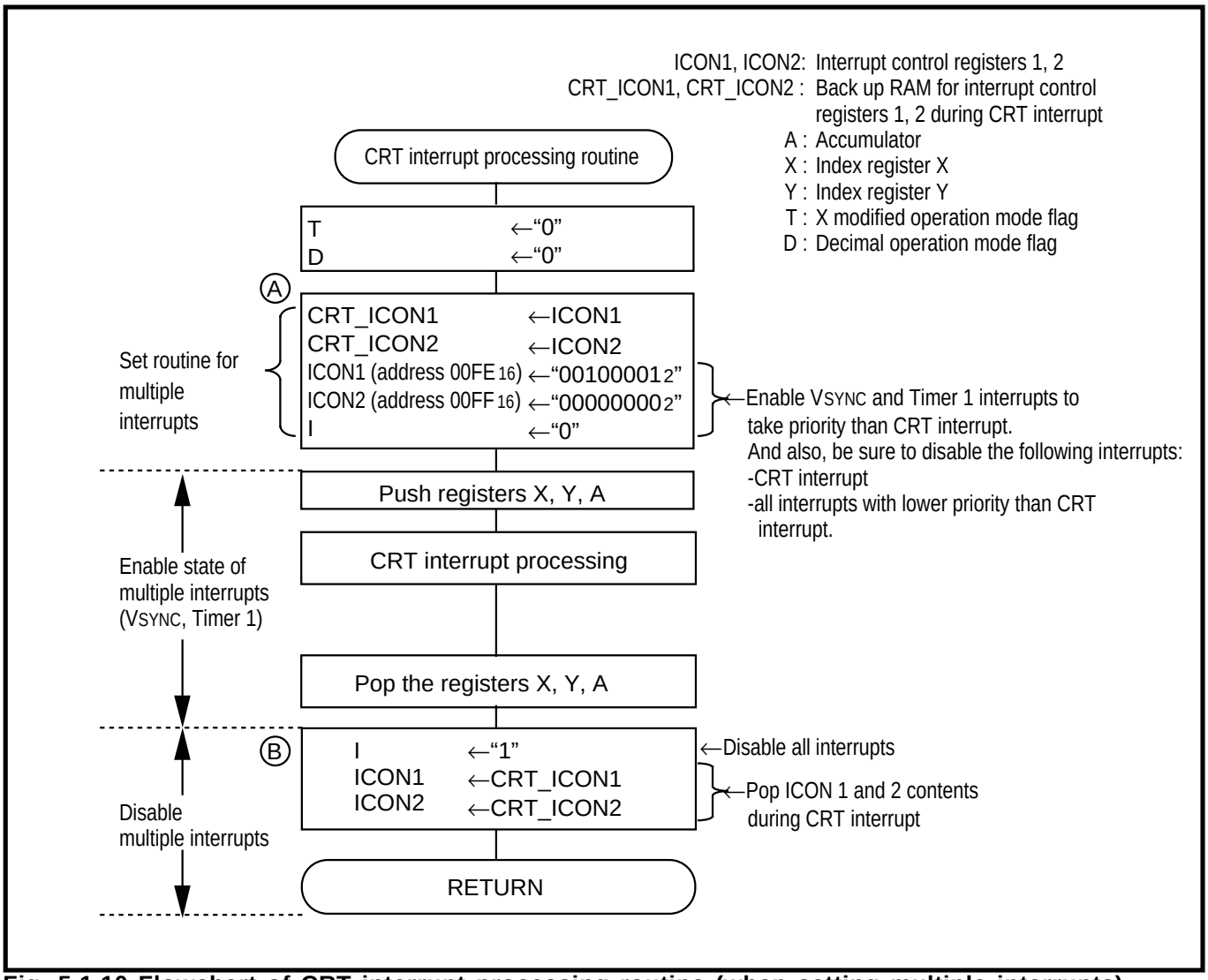


Fig. 5.1.10 Flowchart of CRT interrupt processing routine (when setting multiple interrupts)

(4) V_{SYNC} interrupt processing routine when setting multiple interrupts

Figure 5.1.11 shows the flowchart of V_{SYNC} interrupt processing routine when setting multiple interrupts. Ⓐ' and Ⓑ' are the setting routines for multiple interrupts.

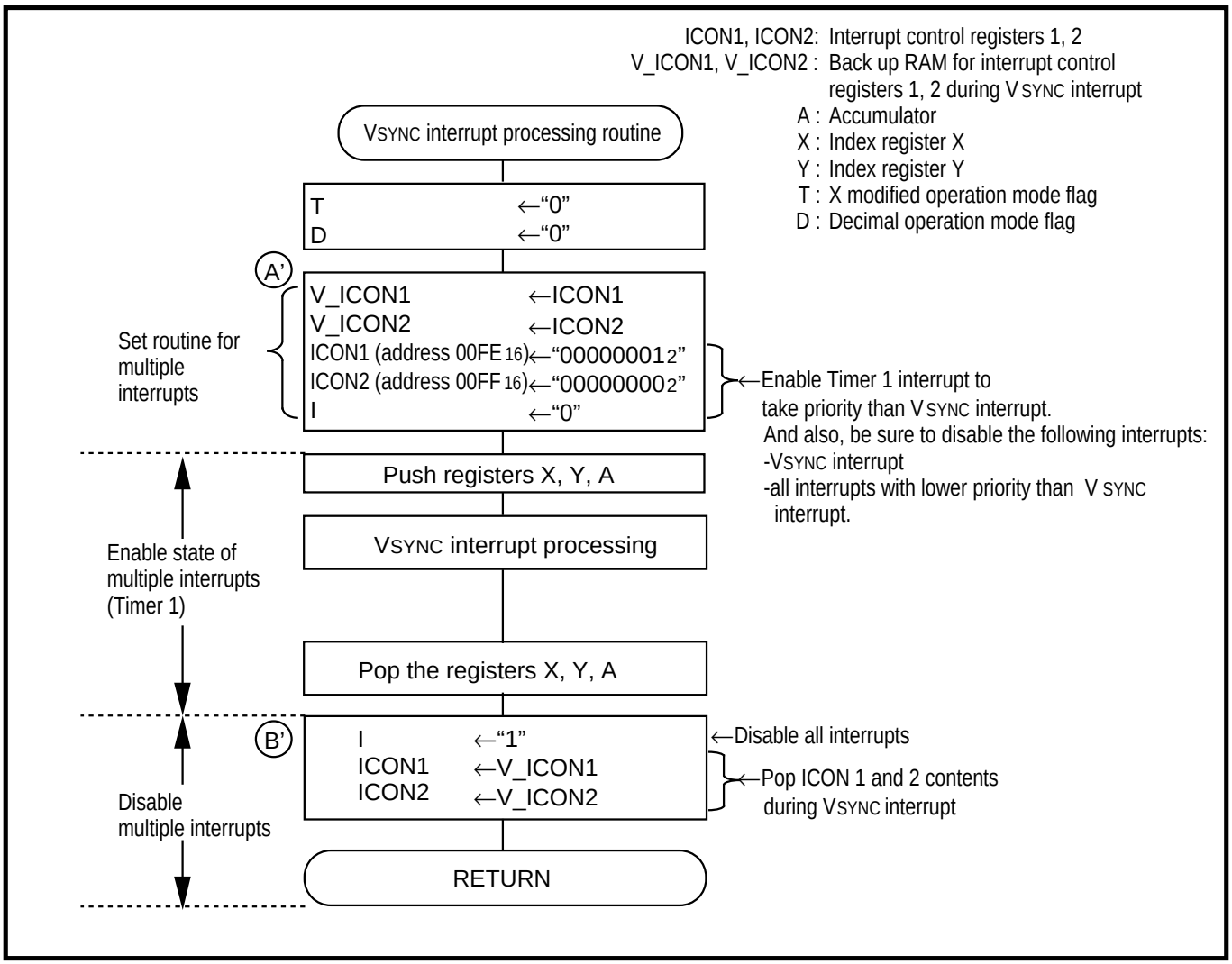


Fig. 5.1.11 Flowchart of V_{SYNC} interrupt processing routine (when setting multiple interrupts)

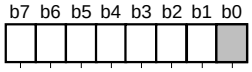
5.2 Notes on programming for OSD (M37220M3-XXXSP/FP)

The emulator MCU M37221ERSS is used for programming development with the M37220M3-XXXSP/FP. However, the functions of the M37221ERSS are compatible with those of the M37221Mx-XXXSP/FP, and therefore has some functions which the M37220M3-XXXSP/FP does not have. Note the following differences when programming using the M37220M3-XXXSP/FP.

5.2.1 Setting of color registers

The color registers of M37220M3-XXXSP/FP are different from those of M37221ERSS (refer to **"Figures 5.2.1 and 5.2.2"**). Character background colors can be output when programming with the M37221ERSS, but not with the M37220M3-XXXSP/FP mask version. This character background color program does not operate on the M37220M3-XXXSP/FP and therefore, it cannot output character background colors (except character background colors by OUT signal; bit 5).

Color Register n



Color register n (CO0 to CO3) (n = 0 to 3) [Addresses 00E6 16 to 00E916]

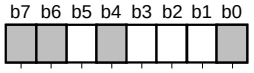
B	Name	Functions	After reset	R	W
0	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
1	B signal output selection bit (COn1)	0 : No character is output 1 : Character is output	0	R	W
2	G signal output selection bit (COn2)	0 : No character is output 1 : Character is output	0	R	W
3	R signal output selection bit (COn3)	0 : No character is output 1 : Character is output	0	R	W
4	B signal output (background) selection bit (COn4)	0 : No background color is output 1 : Background color is output (See note 1)	0	R	W
5	OUT1 signal output control bit (COn5)	0 : Character is output 1 : Blank is output (See notes 1, 2)	0	R	W
6	G signal output (background) selection bit (COn6)	0 : No background color is output 1 : Background color is output	0	R	W
7	R signal output (background) selection bit (COn7)	0 : No background color is output 1 : Background color is output (See note 2)	0	R	W

Notes 1: When bit 5 = "0" and bit 4 = "1," there is output same as a character or border output from the OUT1 pin.
Do not set bit 5 = "0" and bit 4 = "0."
2: When only bit 7 = "1" and bit 5 = "0," there is output from the OUT2 pin.

Fig. 5.2.1 Color register n (M37221ERSS)

5.2 Notes on programming for OSD (M37220M3-XXXSP/FP)

Color Register n



Color register n (COn) (n = 0 to 3) [Addresses 00E6 16 to 00E9 16]

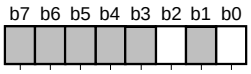
B	Name	Functions	After reset	R	W
0	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
1	B signal output selection bit (COn1)	0 : No character is output 1 : Character is output	0	R	W
2	G signal output selection bit (COn2)	0 : No character is output 1 : Character is output	0	R	W
3	R signal output selection bit (COn3)	0 : No character is output 1 : Character is output	0	R	W
4	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
5	OUT signal output control bit (COn5)	0 : Character is output 1 : Blank is output	0	R	W
6, 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 5.2.2 Color register n (M37220M3-XXXSP/FP)

5.2.2 Setting border selection register

The M37220M3-XXXSP/FP can output neither character background (OUT) nor border at the same time. When setting the border selection bits (bit 0 or 2) to "1," the border output takes over OUT (setting of bit 5 of the color registers). Therefore, select either the character background output or the border output.

Border Selection Register



Border selection register (MD) [Address 00E5 16]

B	Name	Functions	After reset	R	W
0	Block 1 OUT output border selection bit (MD10)	0 : Same output as character output 1 : Border output	Indeterminate	R	W
1	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
2	Block 2 OUT output border selection bit (MD20)	0 : Same output as character output 1 : Border output	Indeterminate	R	W
3 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 5.2.3 Border selection register (M37220M3-XXXSP/FP)

5.2.3 Number of display characters

The M37221ERSS can display up to 24 characters in each block. However, the M37220M3-XXXSP/FP can display only up to 20 characters in each block. Note this when programming using the M37220M3-XXXSP/FP.

5.3 Usage example of ROM correction function (M37221M8/MA-XXXSP)

Application example using the ROM correction function is described below. In this example, it is assumed that the program must be changed by specifications modification after completion of ROM mask. Also, E^2PROM is connected to this system.

5.3.1 Connection example

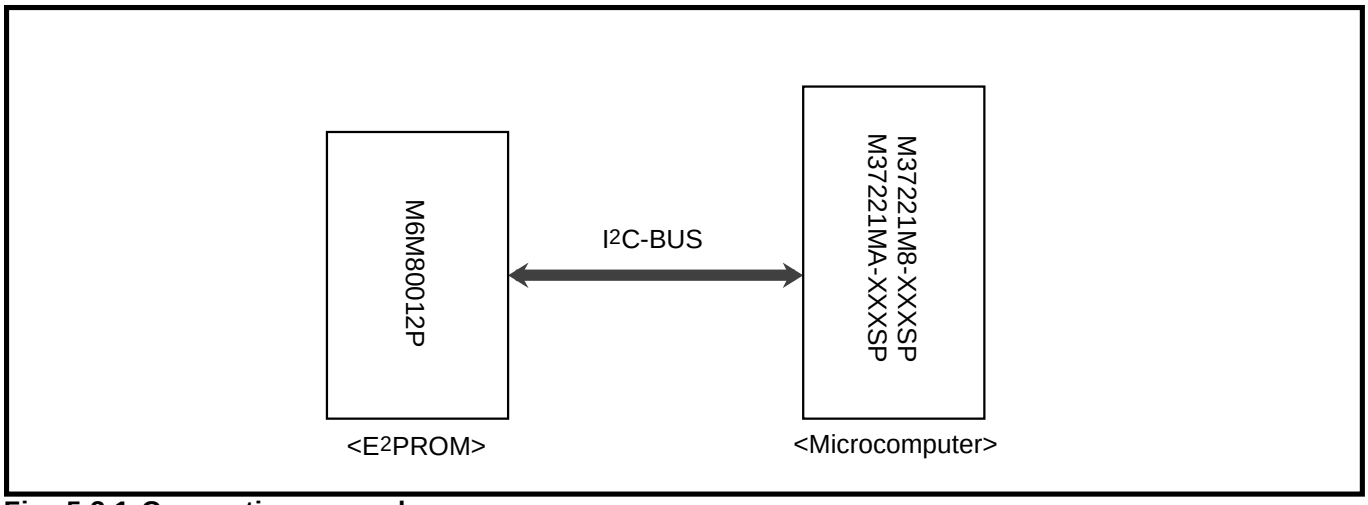


Fig. 5.3.1 Connection example

5.3.2 Correction example

The following is an example when 2 addresses (2 blocks) of ROM are corrected.

(1) Correction example 1

Program before correction		
Address	Machine instructions	Description style
E120	A900	LDA #00H
E122	3A	INC A
E123	8D1101	STA 0111H
E126	60	RTS

Correction program		
Address (block 1)	Machine instructions	Description style
02C0	A980	LDA #80H
02C2	1A	DEC A
02C3	8D1201	STA 0112H
02C6	1A	DEC A
02C7	8D1301	DEC 0113H
02CA	4C26E1	JMP E126H ① (See note)

Note: In ①, E126H is specified as the return destination address of **JMP**.
In this example, since the instruction at the return destination address is **RTS**, even if **RTS** is used instead of **JMP**, the operation is the same as that of **JMP**. As a result, the number of bytes is reduced.

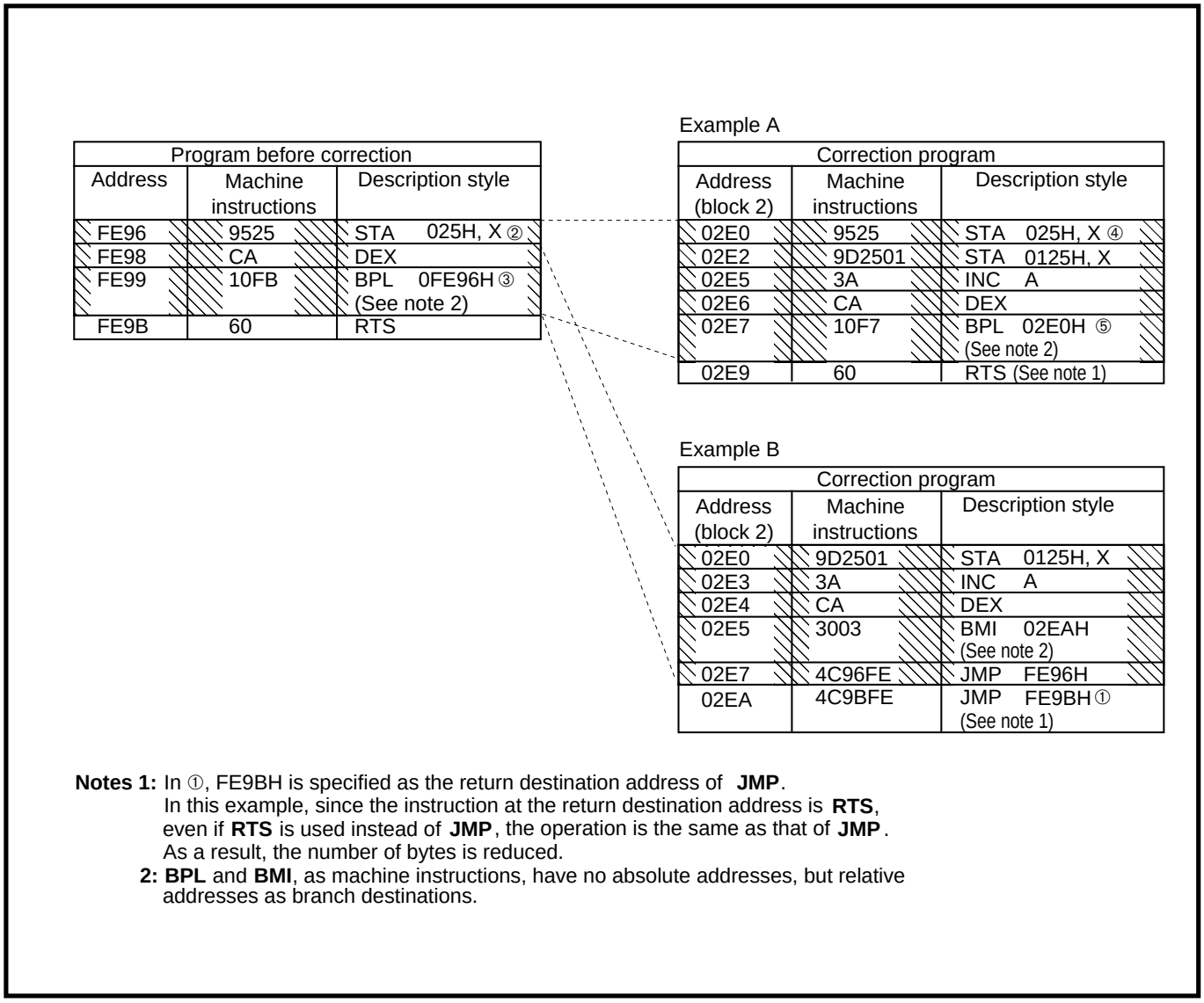
Fig. 5.3.2 Correction example (1)

5.3 Usage example of ROM correction function (M37221M8/MA-XXXSP)

(2) Correction example 2

The loop processing is performed between ② and ③ in Figure 5.3.3. Two examples of this part are shown in detail. Example A corrects in loop units and example B corrects only error instructions. Examples A and B are the same operation, differing in processing time and correction bytes only. Depending on the contents of loop processing, it may be preferable to include correct codes with the codes to be corrected, simplifying the correction program and making it easier to read.

When omitting FE96H (④) and correcting from FE98H, the program cannot move to FE96H by the **BPL** instruction (the jump destination addresses of the **BPL** instruction are limited to bytes between –128 and +127). Therefore, the example B is provided.



Notes 1: In ①, FE9BH is specified as the return destination address of **JMP**.
In this example, since the instruction at the return destination address is **RTS**, even if **RTS** is used instead of **JMP**, the operation is the same as that of **JMP**. As a result, the number of bytes is reduced.

2: **BPL** and **BMI**, as machine instructions, have no absolute addresses, but relative addresses as branch destinations.

Fig. 5.3.3 Correction example (2)

5.3.3 E²PROM map

Figures 5.3.4 and 5.3.5 show the E²PROM map when using the ROM correction function. To store correction codes by using both ROM correction functions 1 and 2, the capacity of E²PROM needs to be approximately 70 bytes.

E²PROM address	Contents	Stored data (Machine instruction)	Instructions in correction program
000₁₆	ROM correction function 1 Valid/invalid (55H: valid, others: invalid)	55H	
001₁₆	ROM correction function 1 Execution address (high-order)	E1H	
002₁₆	ROM correction function 1 Execution address (low-order)	20H	
003₁₆	ROM correction function 1 Correction code 1	A9H	LDA #80H
004₁₆	ROM correction function 1 Correction code 2	80H	
005₁₆	ROM correction function 1 Correction code 3	1AH	DEC A
006₁₆	ROM correction function 1 Correction code 4	8DH	STA 0112H
007₁₆	ROM correction function 1 Correction code 5	12H	
008₁₆	ROM correction function 1 Correction code 6	01H	
009₁₆	ROM correction function 1 Correction code 7	1AH	DEC A
00A₁₆	ROM correction function 1 Correction code 8	8DH	STA 0113H
00B₁₆	ROM correction function 1 Correction code 9	13H	
00C₁₆	ROM correction function 1 Correction code 10	01H	
00D₁₆	ROM correction function 1 Correction code 11	4CH	JMP E126H
00E₁₆	ROM correction function 1 Correction code 12	26H	
00F₁₆	ROM correction function 1 Correction code 13	E1H	
010₁₆	ROM correction function 1 Correction code 14	EAH	NOP (see note)
011₁₆	ROM correction function 1 Correction code 15	EAH	NOP
012₁₆	ROM correction function 1 Correction code 16	EAH	NOP
013₁₆	ROM correction function 1 Correction code 17	EAH	NOP
014₁₆	ROM correction function 1 Correction code 18	EAH	NOP
015₁₆	ROM correction function 1 Correction code 19	EAH	NOP
016₁₆	ROM correction function 1 Correction code 20	EAH	NOP
017₁₆	ROM correction function 1 Correction code 21	EAH	NOP
018₁₆	ROM correction function 1 Correction code 22	EAH	NOP
019₁₆	ROM correction function 1 Correction code 23	EAH	NOP
01A₁₆	ROM correction function 1 Correction code 24	EAH	NOP
01B₁₆	ROM correction function 1 Correction code 25	EAH	NOP
01C₁₆	ROM correction function 1 Correction code 26	EAH	NOP
01D₁₆	ROM correction function 1 Correction code 27	EAH	NOP
01E₁₆	ROM correction function 1 Correction code 28	EAH	NOP
01F₁₆	ROM correction function 1 Correction code 29	EAH	NOP
020₁₆	ROM correction function 1 Correction code 30	4CH	JMP YYXXH
021₁₆	ROM correction function 1 Correction code 31	XXH	Set reset vector address to YYXXH (see note).
022₁₆	ROM correction function 1 Correction code 32	YYH	

Refer to **"Figure 5.3.2"**

Note: When operating normally, this instruction is not executed. This is a redundant processing to reset during program runaway.

Fig. 5.3.4 E²PROM map when using ROM correction function (1)

5.3 Usage example of ROM correction function (M37221M8/MA-XXXSP)

E²PROM address	Contents	Stored data (Machine instruction)	Instructions in correction program
02316	ROM correction function 2 Valid/invalid (55H: valid, others: invalid)	55H	
02416	ROM correction function 2 Execution address (high-order)	FEH	
02516	ROM correction function 2 Execution address (low-order)	96H	
02616	ROM correction function 2 Correction code 1	95H	LDA 025H, X
02716	ROM correction function 2 Correction code 2	25H	
02816	ROM correction function 2 Correction code 3	9DH	LDA 0125H, X
02916	ROM correction function 2 Correction code 4	25H	
02A16	ROM correction function 2 Correction code 5	01H	
02B16	ROM correction function 2 Correction code 6	3AH	INC A
02C16	ROM correction function 2 Correction code 7	CAH	DEX
02D16	ROM correction function 2 Correction code 8	10H	BPL 02E0H
02E16	ROM correction function 2 Correction code 9	F7H	
02F16	ROM correction function 2 Correction code 10	60H	RTS
03016	ROM correction function 2 Correction code 11	EAH	NOP (see note)
03116	ROM correction function 2 Correction code 12	EAH	NOP
03216	ROM correction function 2 Correction code 13	EAH	NOP
03316	ROM correction function 2 Correction code 14	EAH	NOP
03416	ROM correction function 2 Correction code 15	EAH	NOP
03516	ROM correction function 2 Correction code 16	EAH	NOP
03616	ROM correction function 2 Correction code 17	EAH	NOP
03716	ROM correction function 2 Correction code 18	EAH	NOP
03816	ROM correction function 2 Correction code 19	EAH	NOP
03916	ROM correction function 2 Correction code 20	EAH	NOP
03A16	ROM correction function 2 Correction code 21	EAH	NOP
03B16	ROM correction function 2 Correction code 22	EAH	NOP
03C16	ROM correction function 2 Correction code 23	EAH	NOP
03D16	ROM correction function 2 Correction code 24	EAH	NOP
03E16	ROM correction function 2 Correction code 25	EAH	NOP
03F16	ROM correction function 2 Correction code 26	EAH	NOP
04016	ROM correction function 2 Correction code 27	EAH	NOP
04116	ROM correction function 2 Correction code 28	EAH	NOP
04216	ROM correction function 2 Correction code 29	EAH	NOP
04316	ROM correction function 2 Correction code 30	4CH	JMP YYXXH Set reset vector address to YYXXH (see note).
04416	ROM correction function 2 Correction code 31	XXH	
04516	ROM correction function 2 Correction code 32	YYH	

Refer to "Figure 5.3.3 example A"

Note: When operating normally, this instruction is not executed. This is a redundant processing to reset during program runaway.

Fig. 5.3.5 E²PROM map when using ROM correction function (2)

5.3.4 General flowchart

Figure 5.3.6 shows the general flowchart when using ROM correction function. E²PROM addresses in the flowchart corresponds to E²PROM map (refer to **"Figures 5.3.4 and 5.3.5"**).

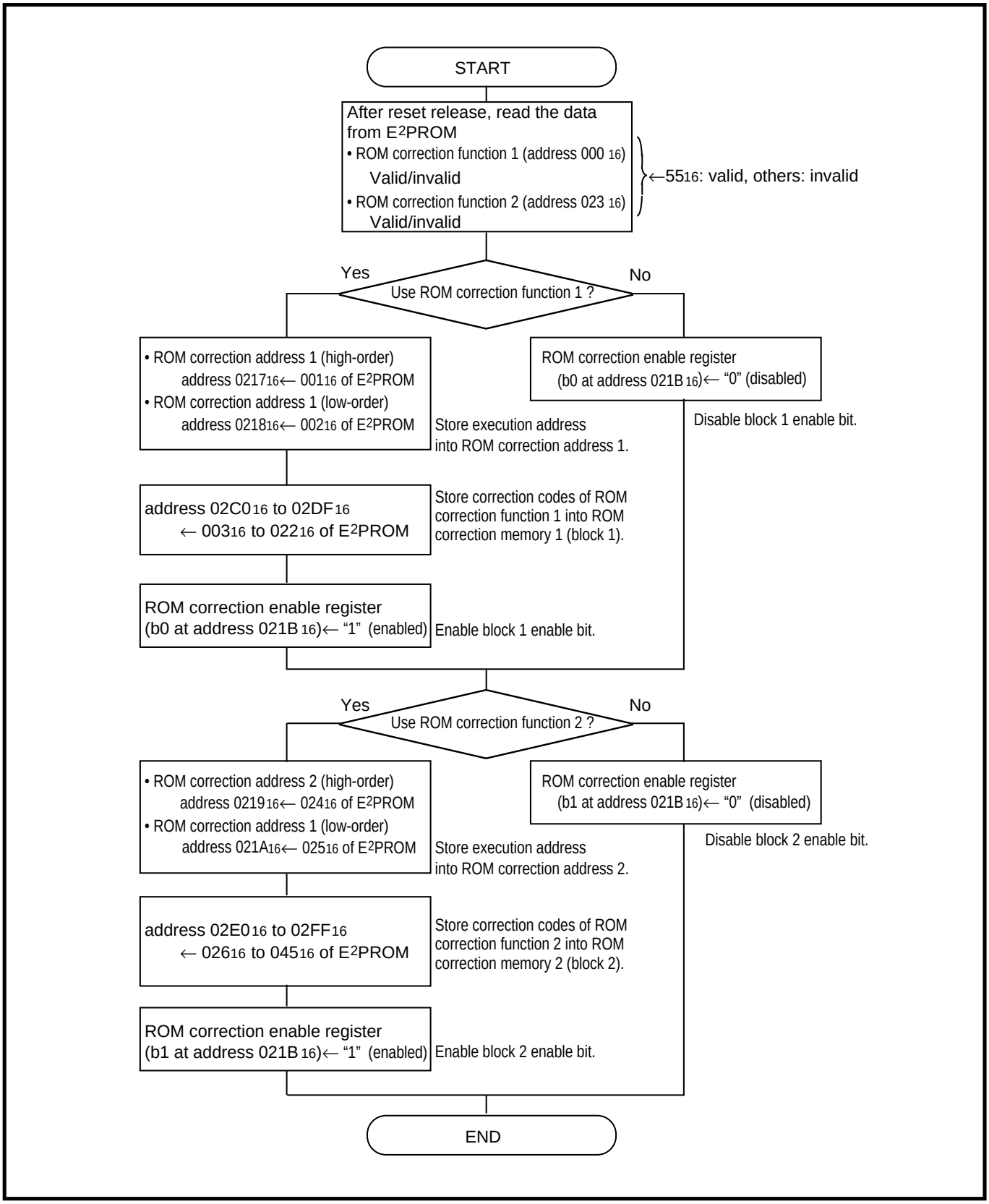


Fig. 5.3.6 General flowchart when using ROM correction function

5.3 Usage example of ROM correction function (M37221M8/MA-XXXSP)

5.3.5 Notes on use

When using the ROM correction function, note the following.

- Specify the first address (op code address) of each instruction as the ROM correction address.
- Use the **RTS**, **RTI** or **JMP** instruction (total of 3 bytes) to return from the correction program to the main program.
- Do not set the same address to ROM correction addresses 1 and 2 (addresses 0217_{16} to $021A_{16}$).

5.4 Example of I²C-BUS interface control (M37221Mx-XXXSP/FP)

The M37221Mx-XXXSP/FP has multi-master I²C-BUS interface. This interface, offering both arbitration lost detection and synchronous functions, is useful for the multi-master serial communications.
This paragraph explains transmit/receive control example of E²PROM (M6M80012P) adaptable to the I²C-BUS interface.
For details on the I²C-BUS interface, refer to **"2.8 Multi-master I²C-BUS interface."**

5.4.1 Specifications

- E²PROM required: M6M80012P
- Synchronous clock: internal clock
- Standard clock mode: 100 kHz
- Number of transfer bits: 8 bits
- Data format: addressing format
- Pins required: SCL1, SDA1
- Direction of data transfer: MSB first

5.4.2 Connection example

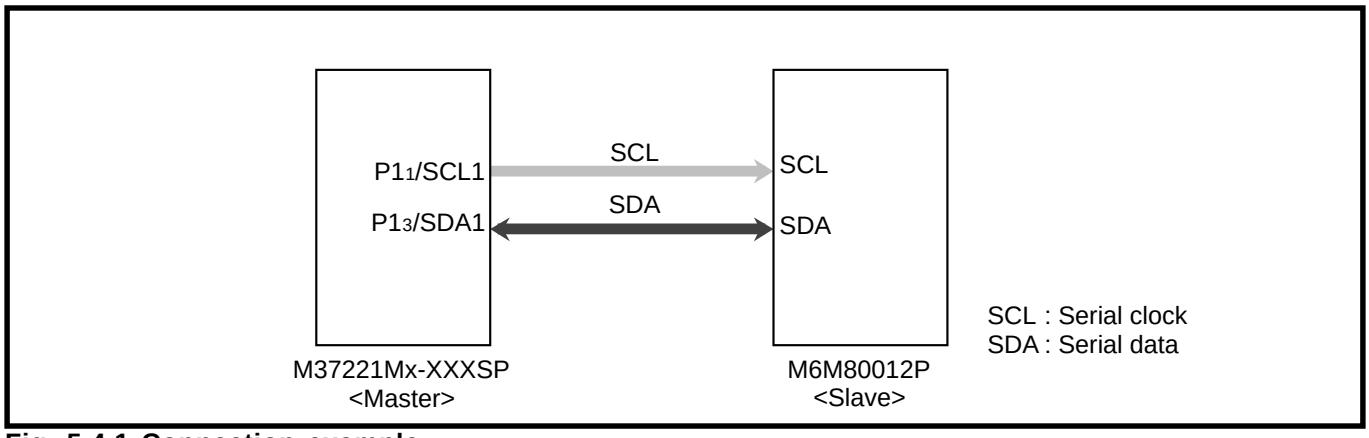


Fig. 5.4.1 Connection example

5.4.3 E²PROM functions

(1) Byte write

Bytes are written by sending the START condition, slave address "$A0_{16}$," sub-address (1 byte), data (1 byte), and the STOP condition from the master. Writing to the E²PROM will be started after the master sends the STOP condition, that is, in synchronization with a rising edge of the SDA signal. This writing will be automatically terminated by the on-chip sequential controller. In this period, no acknowledge bits are generated.

Figure 5.4.2 shows the byte write timing.

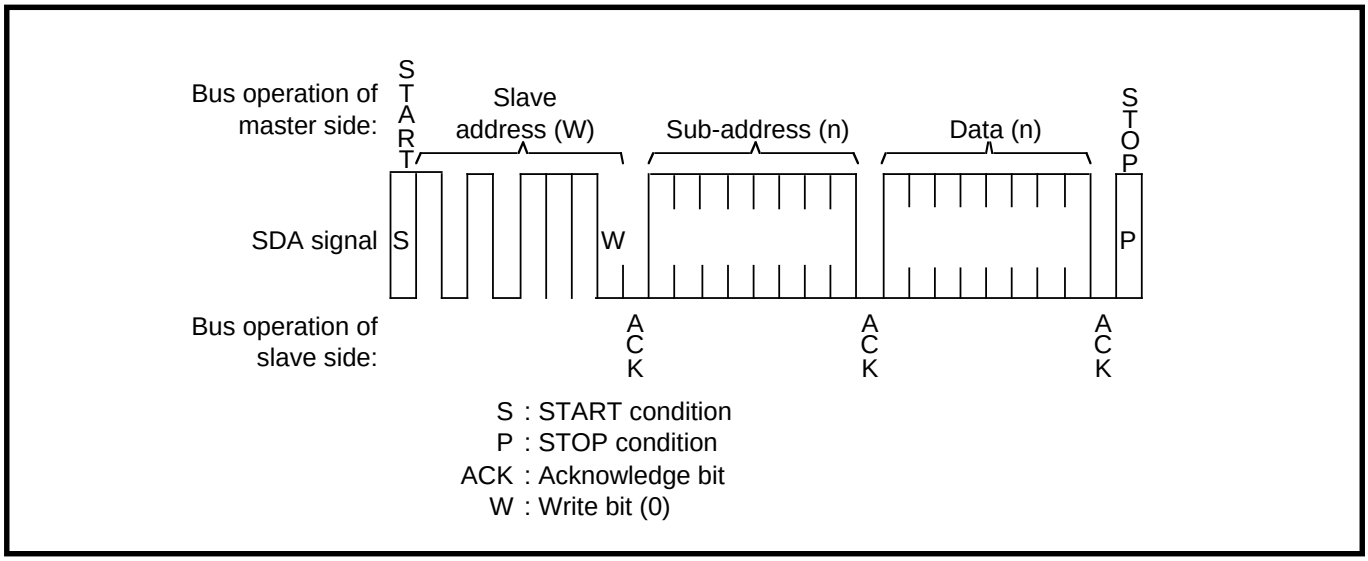


Fig. 5.4.2 Byte write timing

(2) Random address read

In this mode, the data of an arbitrary address is read. To set the first-read address, the master sends the START condition, slave address "$A0_{16}$," and sub-address (1 byte). Upon receiving the acknowledge bit (ACK) from the E²PROM, the master sends the RESTART condition signal and slave address "$A1_{16}$" again. After ACK is generated from the E²PROM, the data of the corresponding sub-address is read out.

After the data is output, no acknowledge bits are generated, but the STOP condition is sent by the master, completing this read operation.

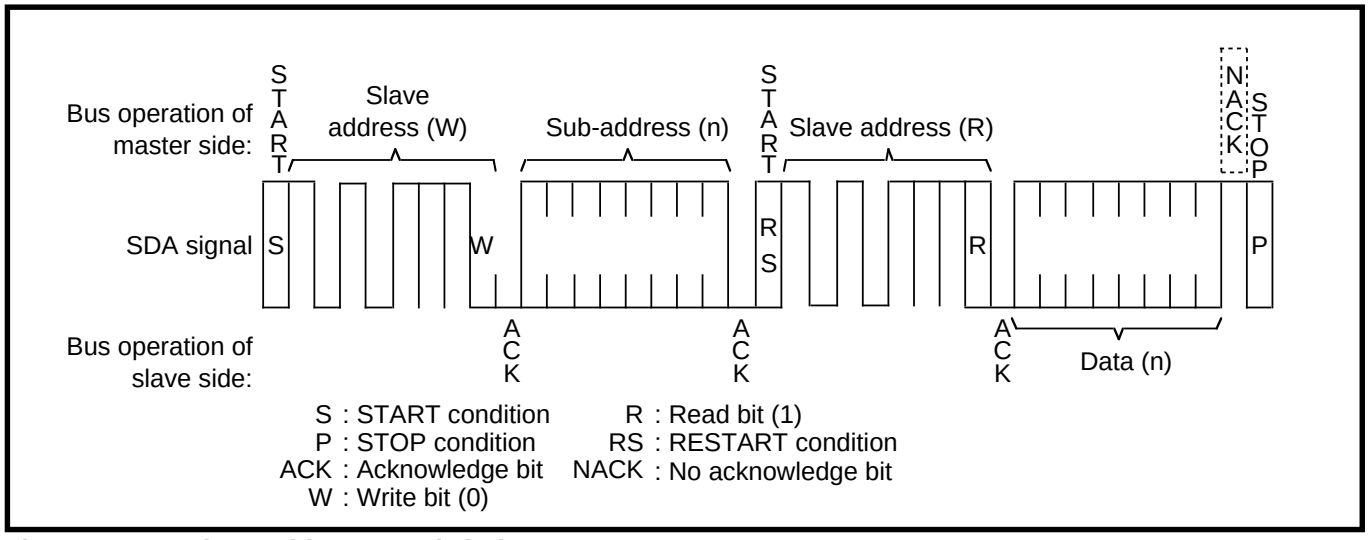


Fig. 5.4.3 Random address read timing

5.4.4 General flowchart

The processing routines which controls I²C-BUS devices branch to the write processing routine and the read processing routine. The data output processing routine is used as the common processing routine.

(1) Write processing routine

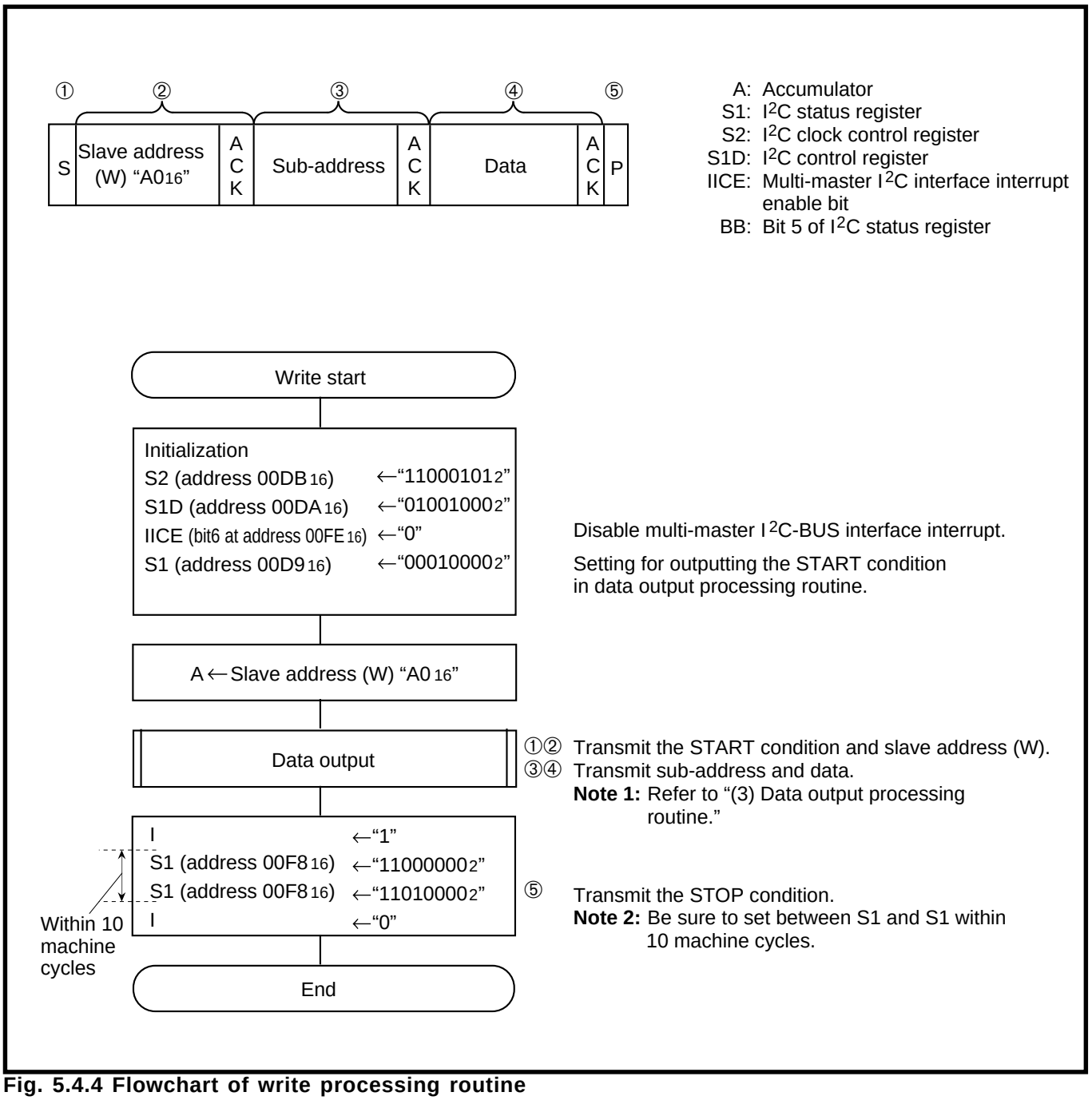
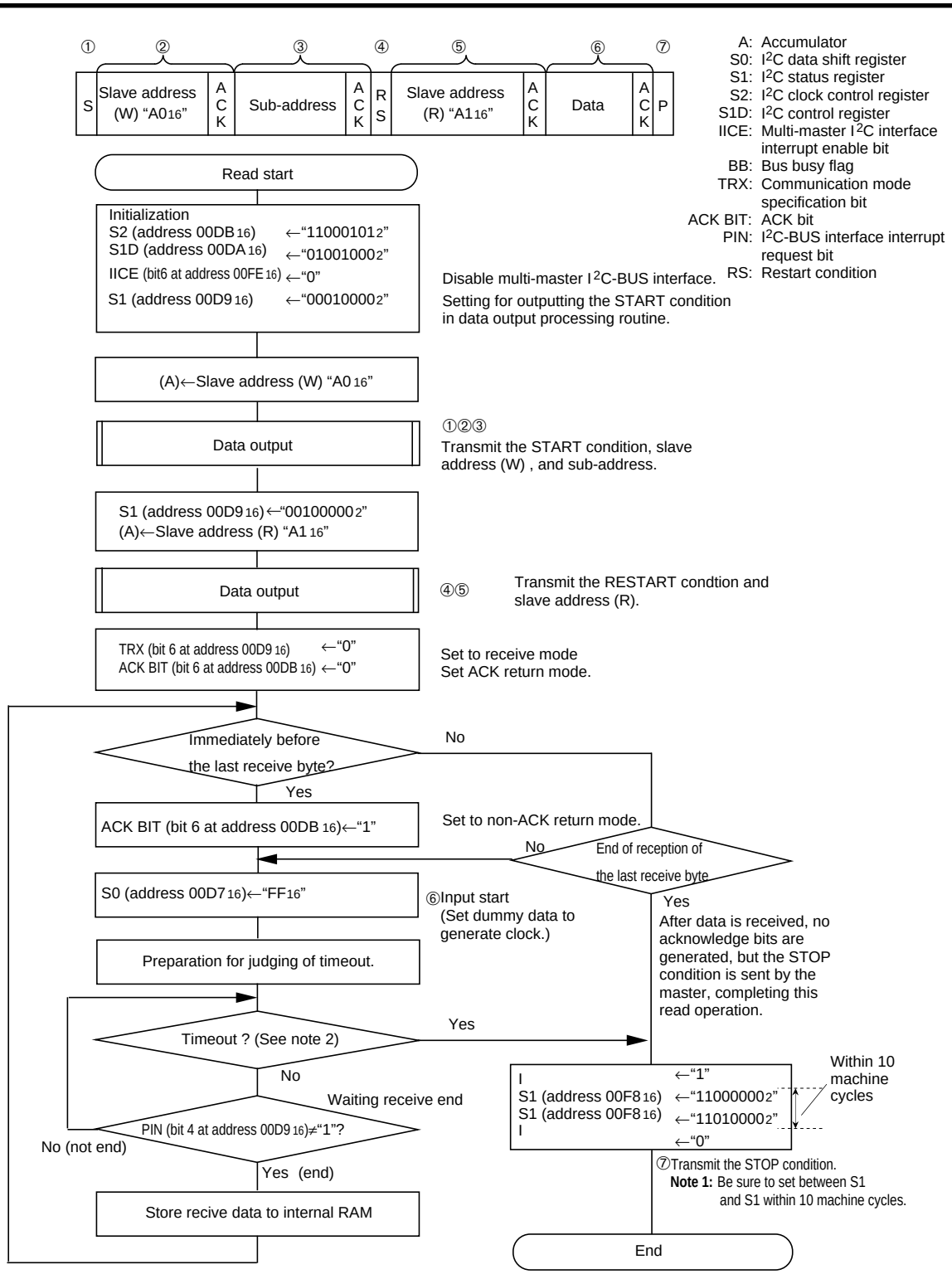


Fig. 5.4.4 Flowchart of write processing routine

(2) Read processing routine



Note 2: The timeout count is performed by software with interrupts, such as timers. Accordingly, if receive operation is not completed due to various influences, the loop continues. Therefore, if receive operation does not complete within a certain time, I²C-BUS access is stopped by outputting STOP condition. If I²C-BUS access is stopped by timeout, the obtained data is incorrect data.

Fig. 5.4.5 Flowchart of read processing routine

(3) Data output processing routine

The data output processing routine is the common routine within the transmit/receive processing routine.

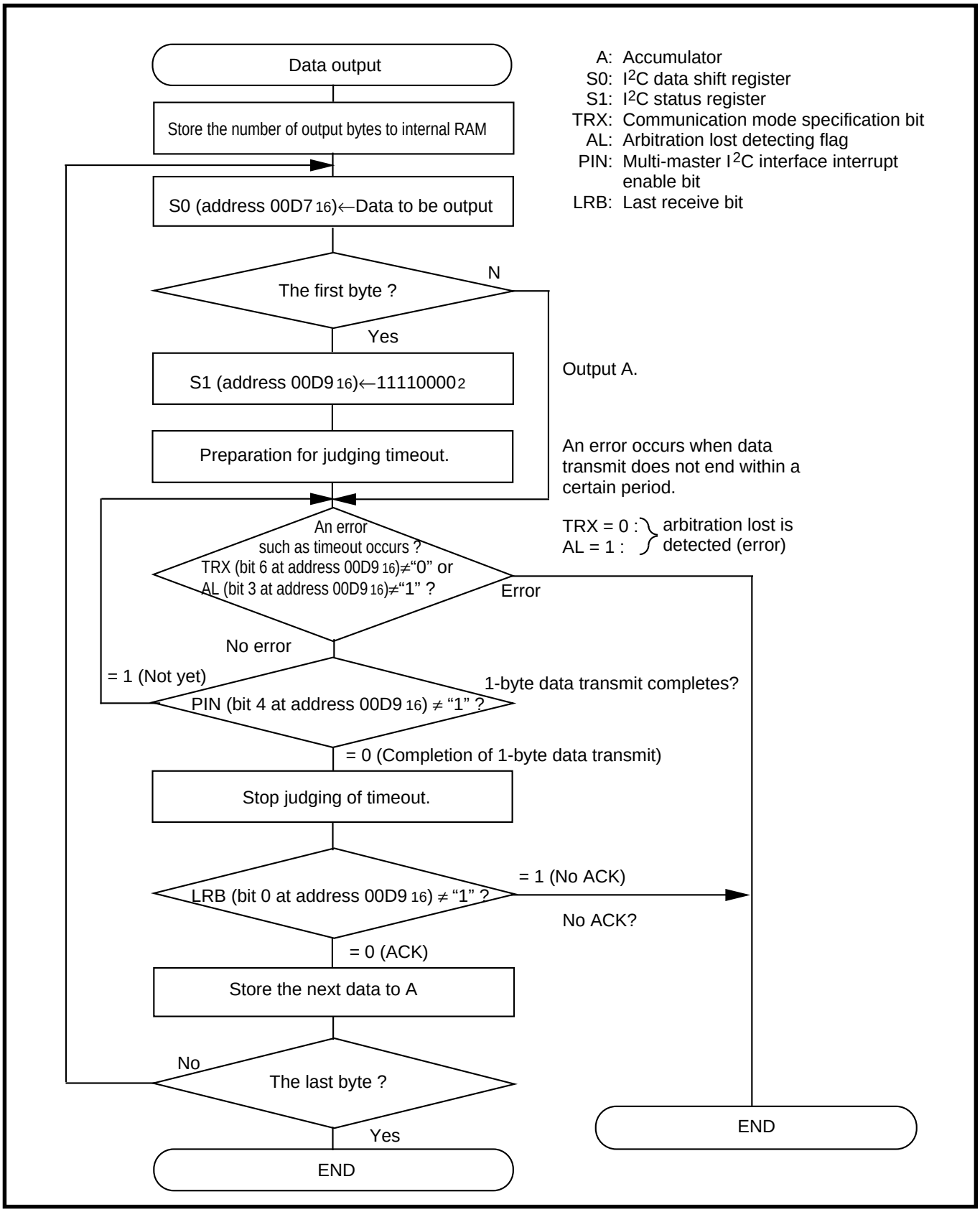


Fig. 5.4.6 Flowchart of data output processing routine

5.5 Example of I²C-BUS control by software (M37220M3-XXXSP/FP)

Althogh, the M37220M3-XXXSP/FP has no multi-master I²C-BUS interface, it can control single-master I²C-BUS by software. Most TV systems can be controlled in this way.
This paragraph explains transmit/receive control example of a single-chip color TV signal processor (M52340SP) adaptable to the I²C-BUS interface.

5.5.1 Specifications

- Single-chip color TV signal processor required: M52340SP
- Number of transfer bits: 8 bits
- Data format: addressing format
- Pins required: $P2_1$, $P2_0$
- Direction of data transfer: MSB first

5.5.2 Connection example

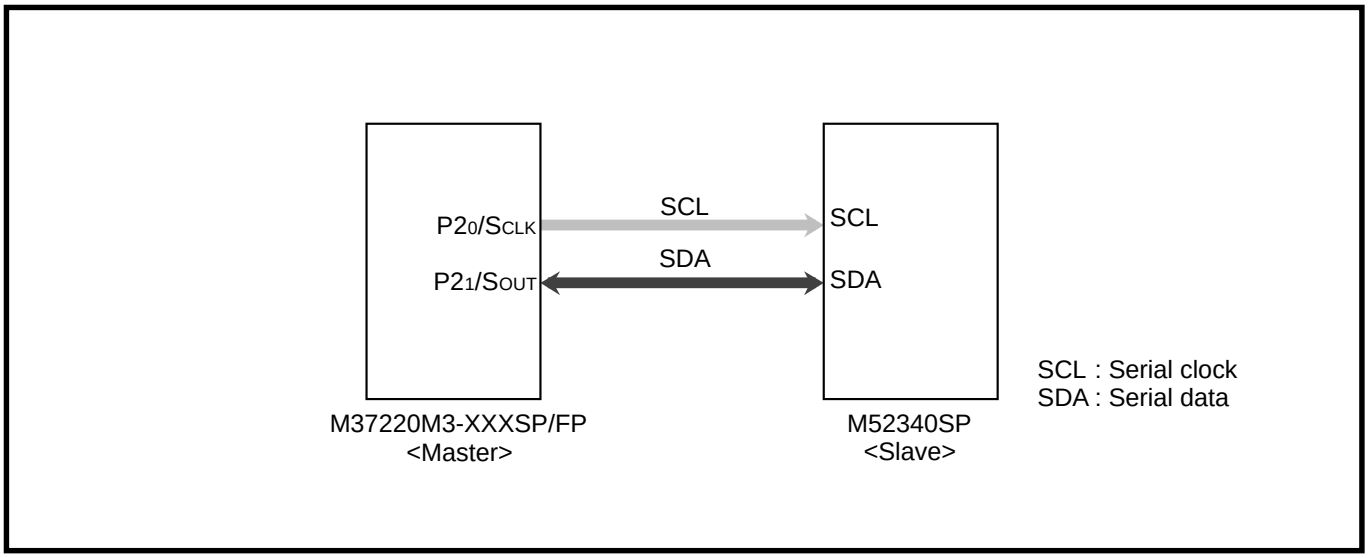


Fig. 5.5.1 Connection example

5.5.3 Single-chip color TV signal processor function

(1) Status read

Status is read by sending the START condition, slave address "BB_{16}." After ACK is generated from the M52340SP, the status data is read out. After the status data is output, any acknowledge bit is not generated, but the STOP condition is sent by the master. Then this read operation is completed.

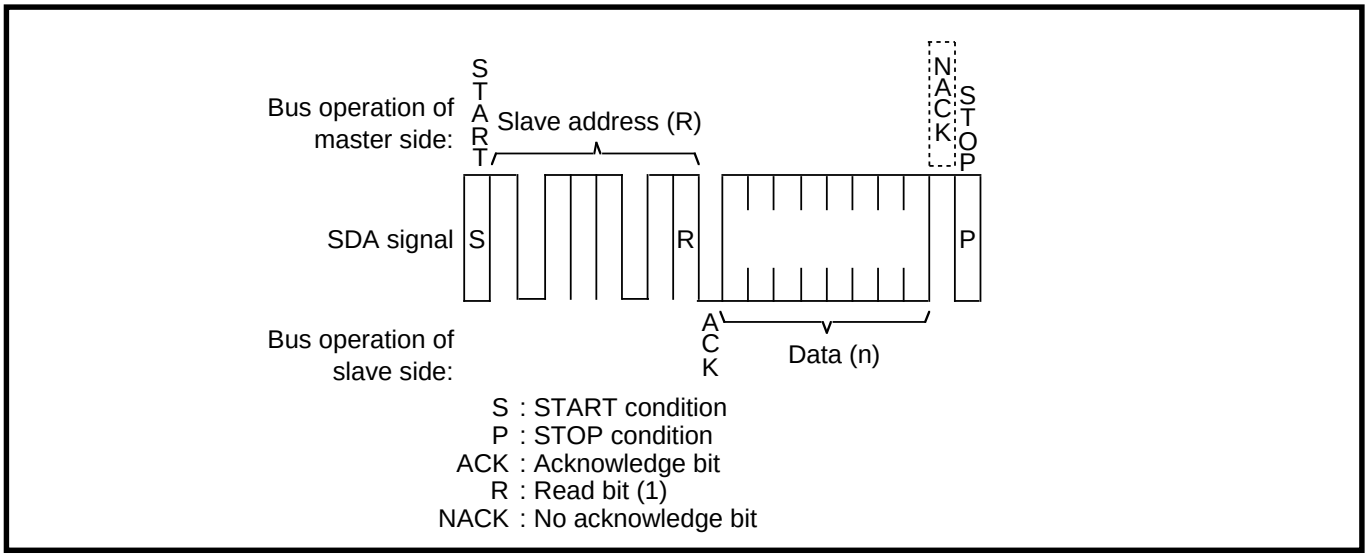


Fig. 5.5.2 Staus read timing

(2) Byte write

Bytes are written by sending the START condition, slave address "BA_{16}," sub-address (1 byte), data (1 byte), and the STOP condition from the master.

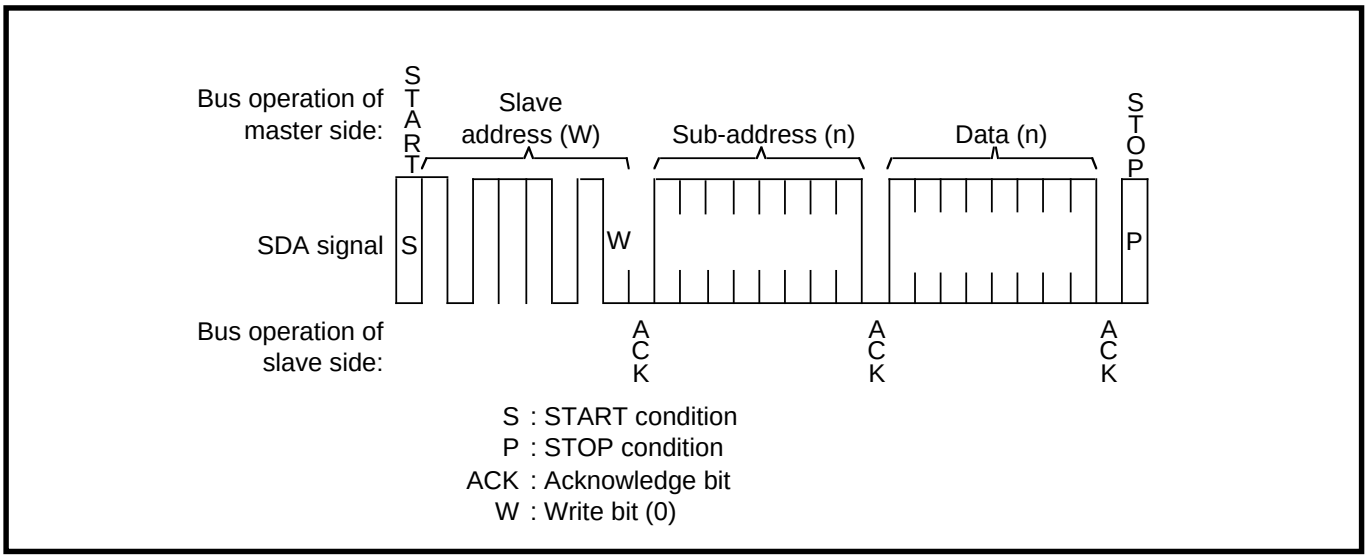


Fig. 5.5.3 Byte write timing

5.5 Example of I²C-BUS control by software (M37220M3-XXXSP/FP)

5.5.4 General flowchart

(1) Write processing routine

The processing routine which controls I²C-BUS devices branch to the write processing routine and the read processing routine. The START condition, the STOP condition and the data output processing routine are used as the common processing routine.

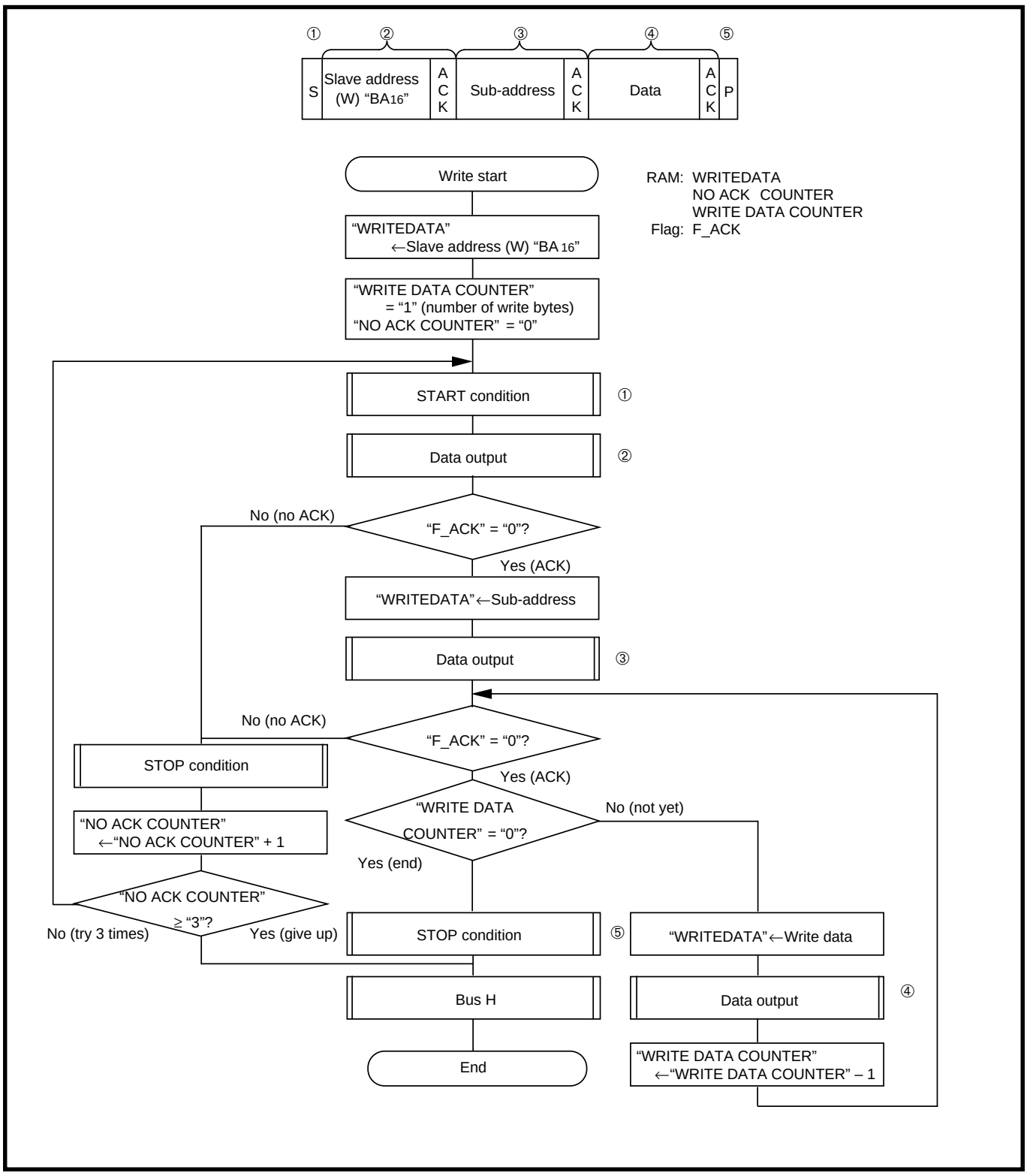


Fig. 5.5.4 Flowchart of write processing routine

(2) Read processing routine

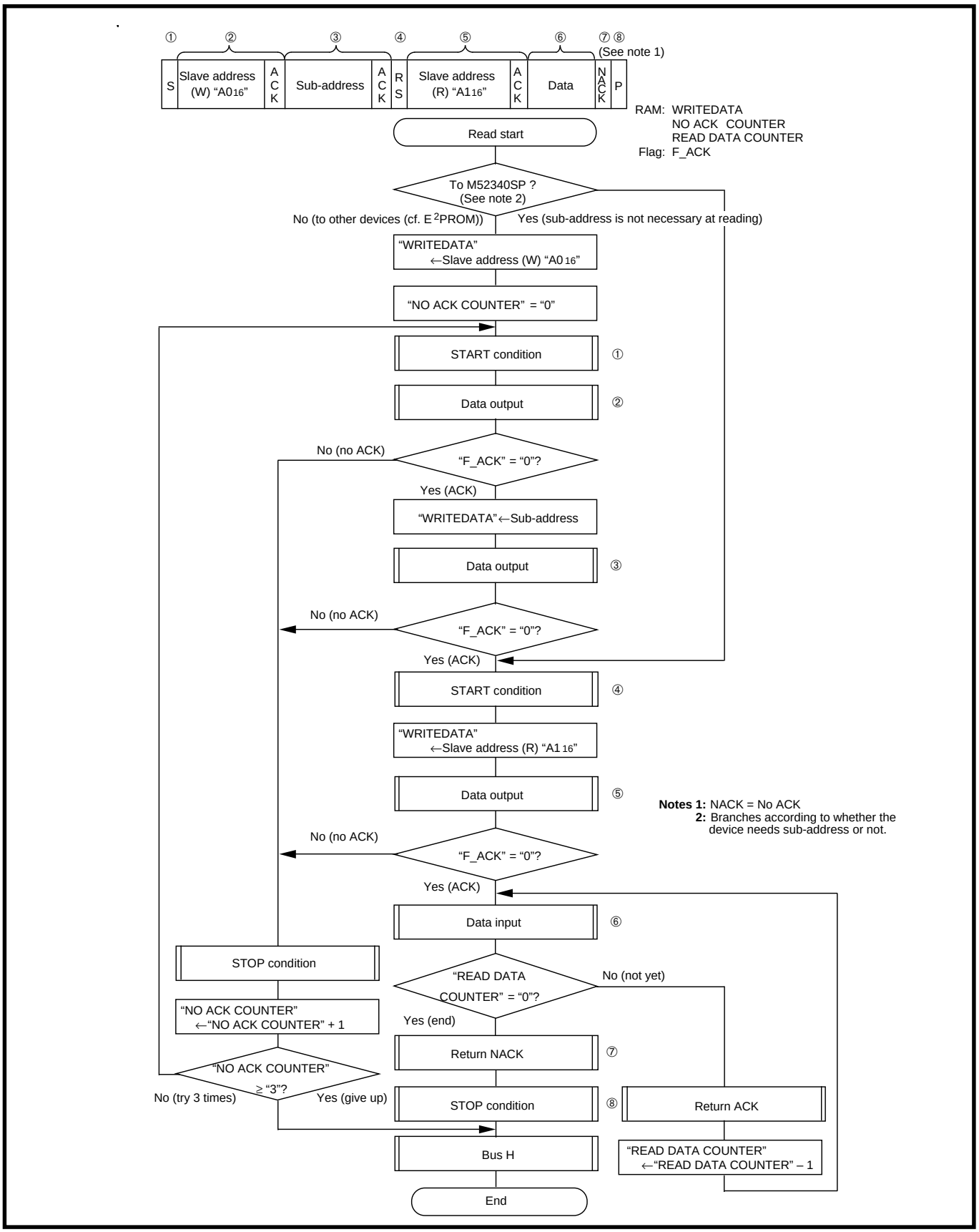


Fig. 5.5.5 Flowchart of read processing routine

5.5 Example of I²C-BUS control by software (M37220M3-XXXSP/FP)

(3) Data output processing routine

The data output, the START condition, the STOP condition, and the bus H processing routines are the common routines within the transmit/receive processing routine.

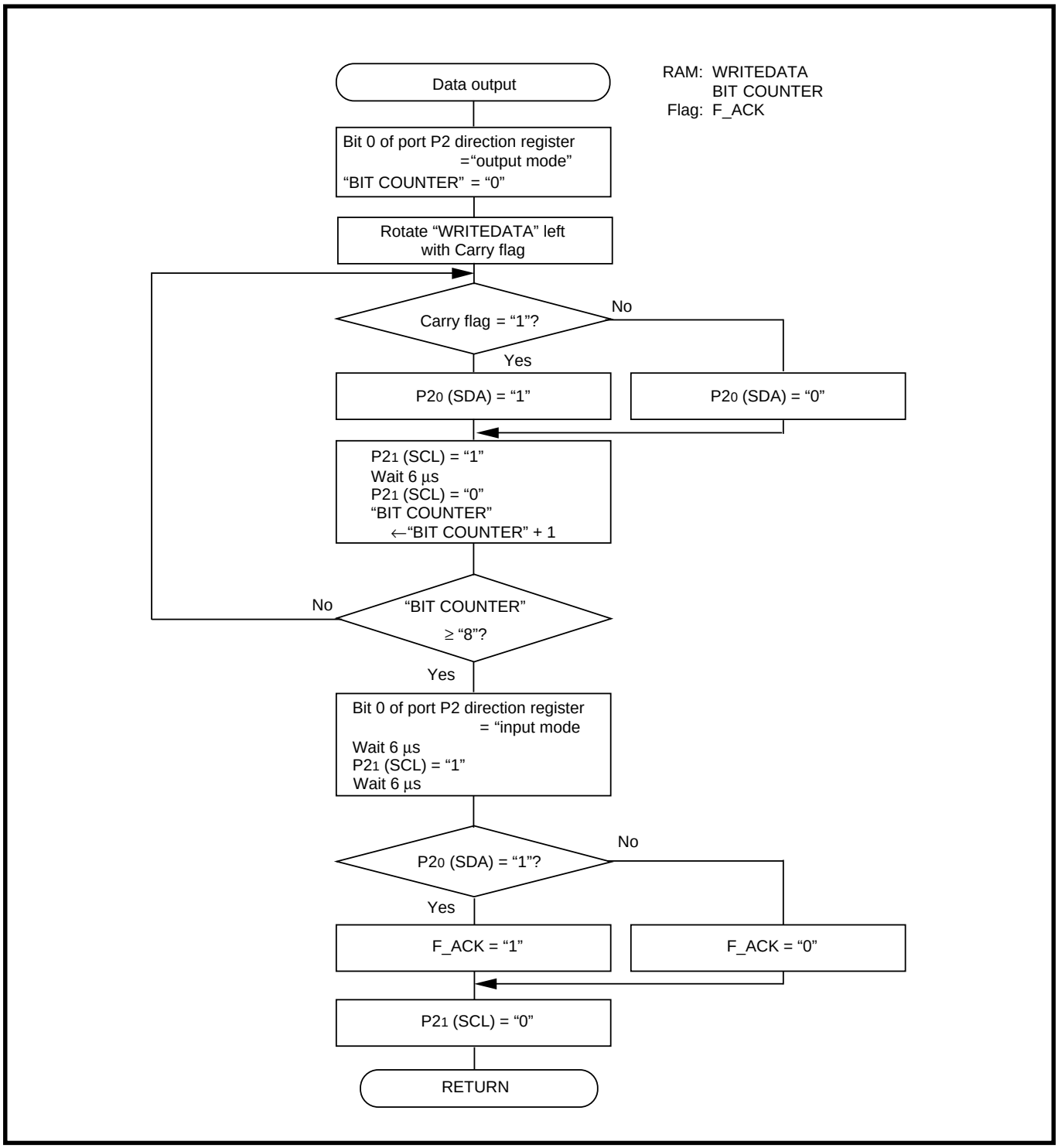


Fig. 5.5.6 Flowchart of data output processing routine

(4) START condition processing routine

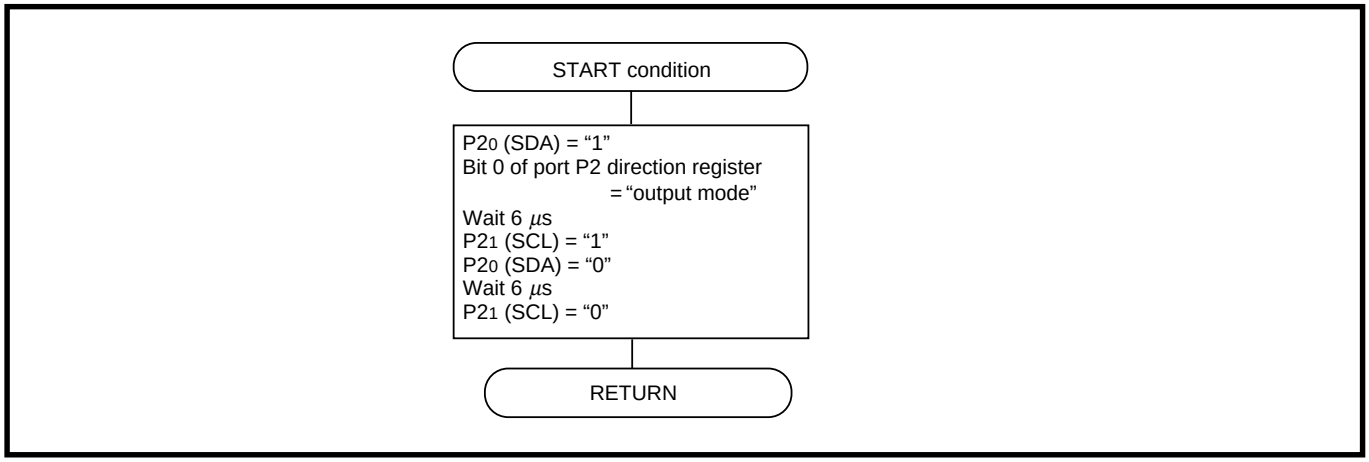


Fig. 5.5.7 Flowchart of START condition processing routine

(5) STOP condition processing routine

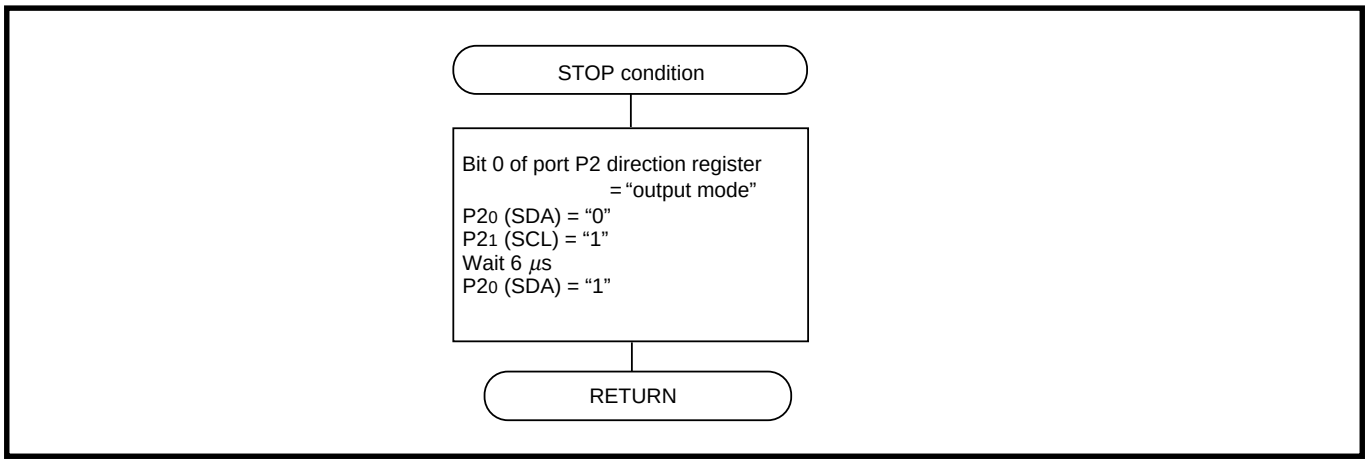


Fig. 5.5.8 Flowchart of STOP condition processing routine

(6) Bus H processing routine

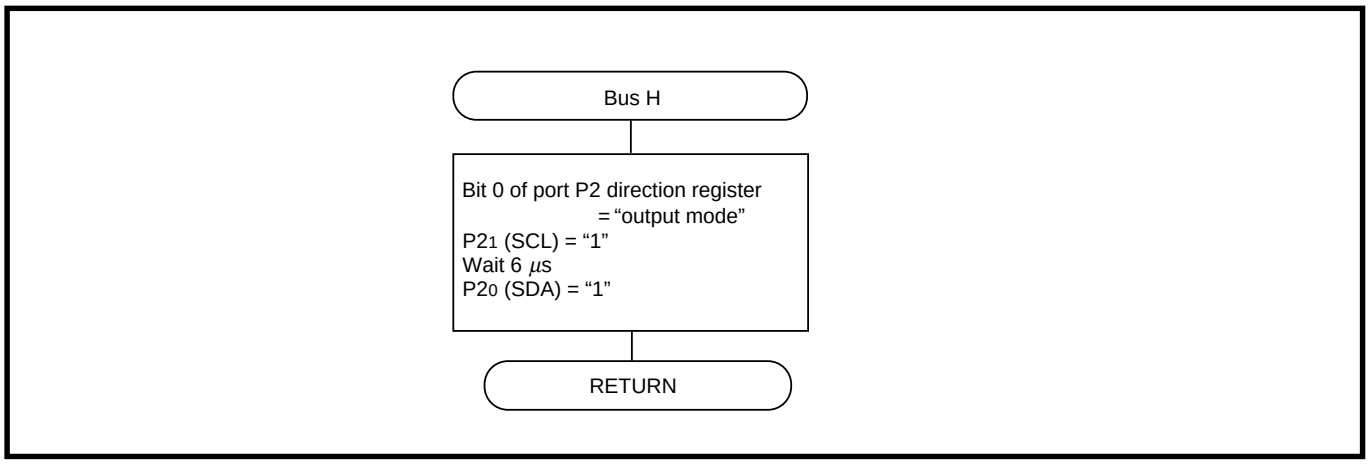


Fig. 5.5.9 Flowchart of bus H processing routine

5.5 Example of I²C-BUS control by software (M37220M3-XXXSP/FP)

(7) Data input processing routine

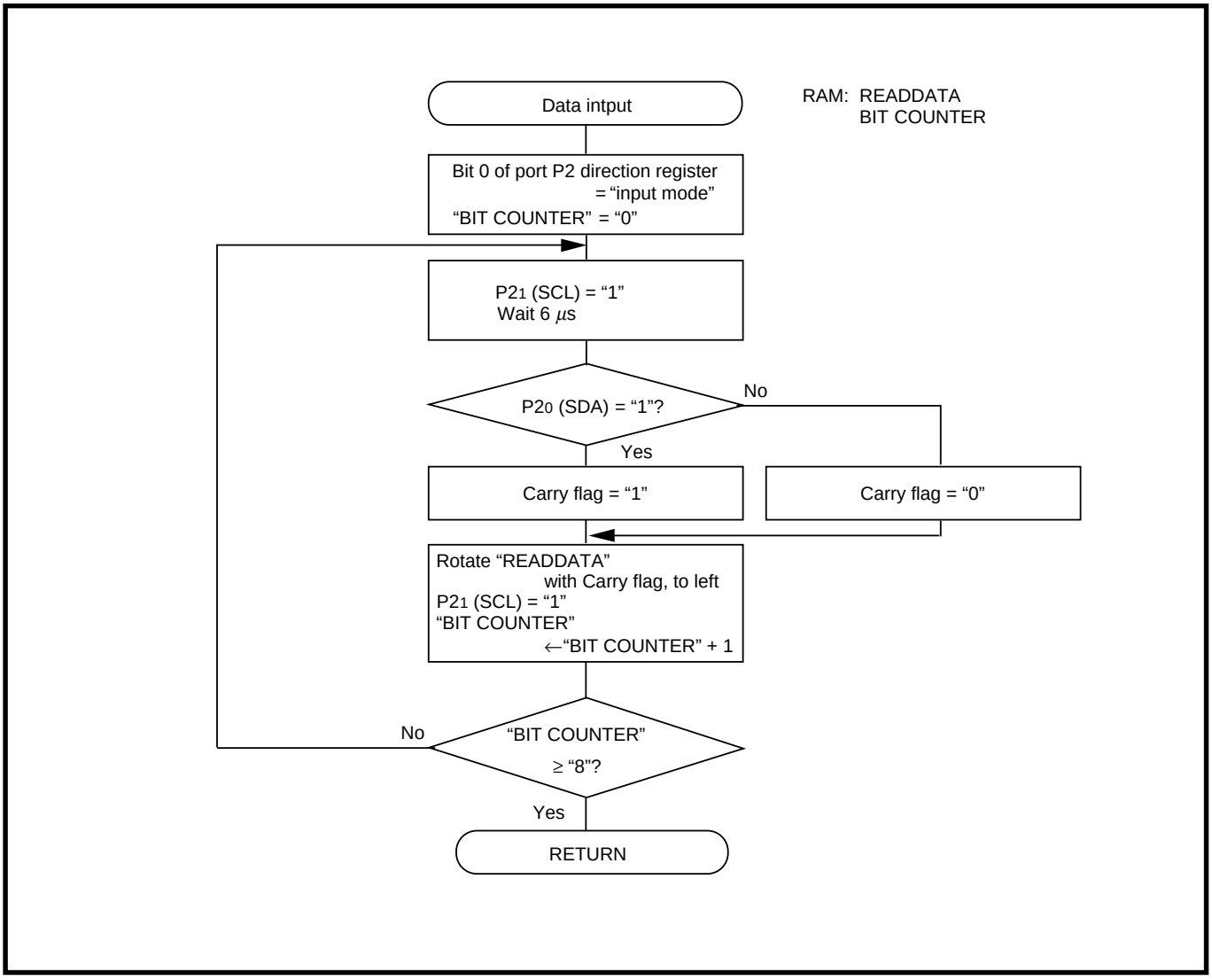


Fig. 5.5.10 Flowchart of data input processing routine

(8) Return ACK processing routine

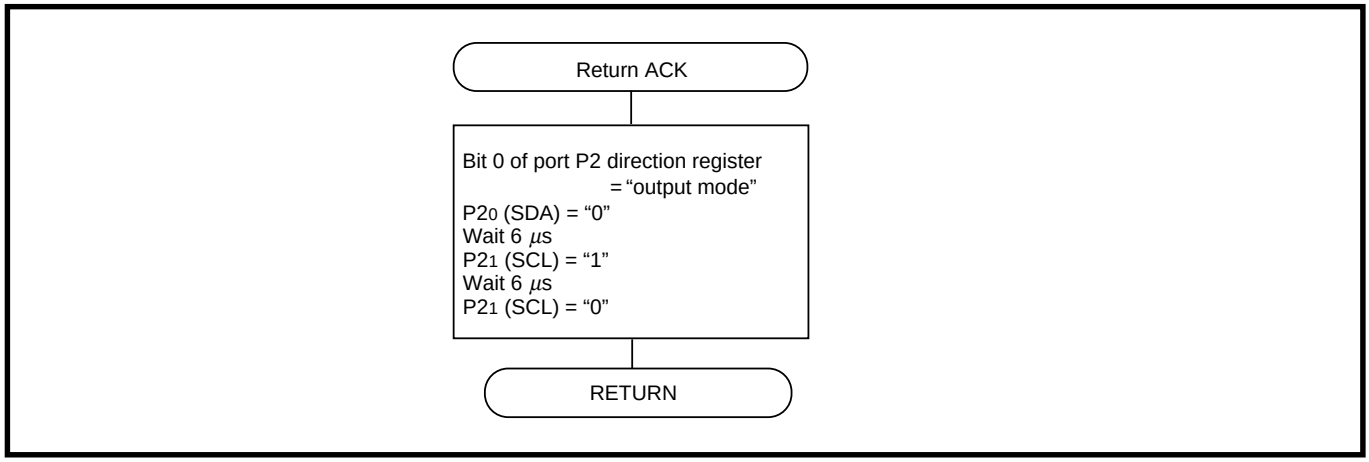


Fig. 5.5.11 Flowchart of return ACK processing routine

(9) Return NACK processing routine

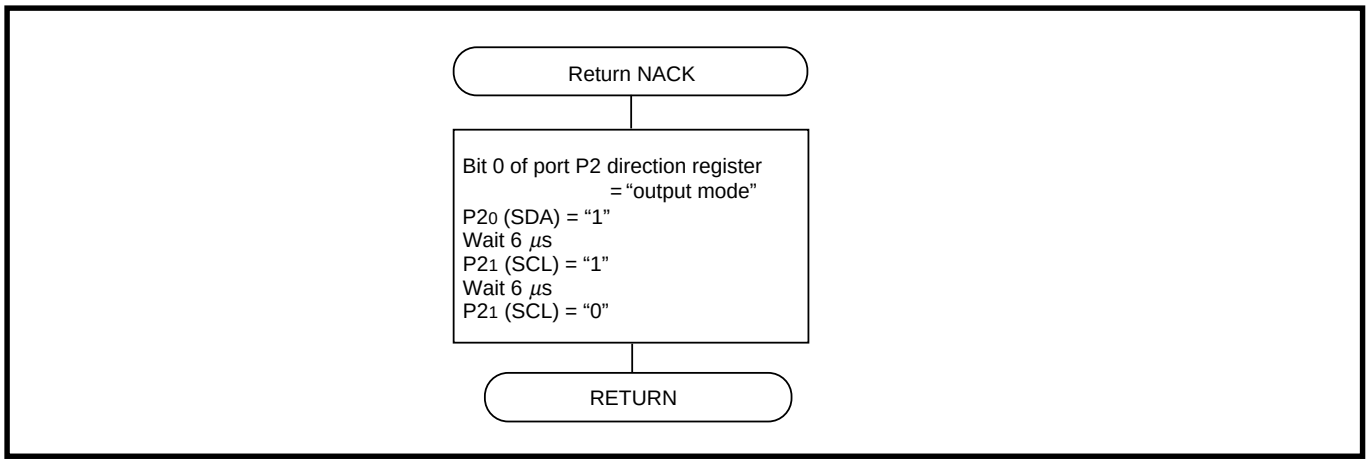


Fig. 5.5.12 Flowchart of return NACK processing routine

5.5 Example of I²C-BUS control by software (M37220M3-XXXSP/FP)

5.5.5 Data setting according to key processing

Examples of the M52340SP settings, corresponding to each actual TV set key input, are described below.

(1) "Power ON/OFF key" input

When power supply is supplied to the M52340SP by this input, the data is set to all registers (at sub-addresses "00_{16}" to "13_{16}").

(2) "Tuning and search-related keys" (CH UP/DOWN key, CH direct selection key) input

When tuning, the color system data (refer to **"Table 5.5.10"**) is set according to the determination result of the color system by the status data register (refer to **"Table 5.5.1"**). Also, the corresponding data is set when the color system search-related keys are input. However, note that the above-mentioned setting is valid only when setting AUTO (bit 5 at sub-address 06_{16}, write data) to "0." When setting to "1," the data is automatically set inside the M52340SP.

When tuning, the data is set as shown in Table 5.5.1.

Table 5.5.1 Data setting at tuning and searching

Sub-address	Bit	Data
02_{16}	D5	TRAP
	D4	DBF
	D1	DFA
06_{16}	D0, D1	DL TIME

(3) "Volume UP/DOWN key" input

When the volume up/down key is input, the data is set as shown in Table 5.5.2.

Table 5.5.2 Data setting at "volume UP/DOWN key" input

Sub-address	Bit	Data
03_{16}	D0 to D6	AUDIO ATT

(4) "Screen-size-related keys" input

When the screen-related keys are input on TVs with various screen sizes (wide aspect TV, etc.), the screen size data and position data is set as shown in Table 5.5.3. Also, the data of each frequency (50 Hz or 60 Hz) is occasionally held.

Table 5.5.3 Data setting at "screen-size-related keys" input

Sub-address	Bit	Data
09_{16}	D3 to D6	H PHASE

(5) "Picture data control key" and "Picture memory switching key" input

When changing picture data, the data is set to the corresponding write data register as shown in Table 5.5.4.

Table 5.5.4 Data setting at "picture data control key" and "picture memory switching key" input

Sub-address	Bit	Data
04_{16}	D0 to D5	SHARPNESS
05_{16}	D0 to D6	CONTRAST
07_{16}	D0 to D6	TINT
08_{16}	D0 to D6	COLOR
$0A_{16}$	D0 to D6	BRIGHT

(6) Data setting when changing AFT (auto fine tuning) state

To change the state of auto fine tuning at presetting CH and ordinary tuning, the bit is set as shown in Table 5.5.5.

Table 5.5.5 Data setting when changing AFT state

Sub-address	Bit	Data
04_{16}	D6	DEFEAT

(7) Data setting when changing audio mute state

When the audio mute key is input, the bit is set as shown in Table 5.5.6. If it is necessary to delete the sound while tuning with the tuning key input or presetting CH, the bit is set as shown in Table 5.5.6.

Table 5.5.6 Data setting when changing audio mute state

Sub-address	Bit	Data
01_{16}	D6	A MUTE

(8) Data setting when changing video mute state

When muting the video on screen while tuning with the tuning key input, the bit is set as shown in Table 5.5.7.

Table 5.5.7 Data setting when changing video mute state

Sub-address	Bit	Data
$0B_{16}$	D6	MUTE

(9) Data setting when adjusting white balance

When adjusting the TV picture in the factory, set the data shown in Table 5.5.8 to ready the service mode for adjusting the white color.

Table 5.5.8 Data setting when adjusting white color balance

Sub-address	Bit	Data
13_{16}	D3	SERSW

5.5 Example of I²C-BUS control by software (M37220M3-XXXSP/FP)

5.5.6 Flowchart of data setting according to key processing

Figures 5.5.13 to 5.5.15 show the flowcharts of controlling the M52340SP when there are various event inputs to the actual TV system.

(1) Poweron processing by "power key" input

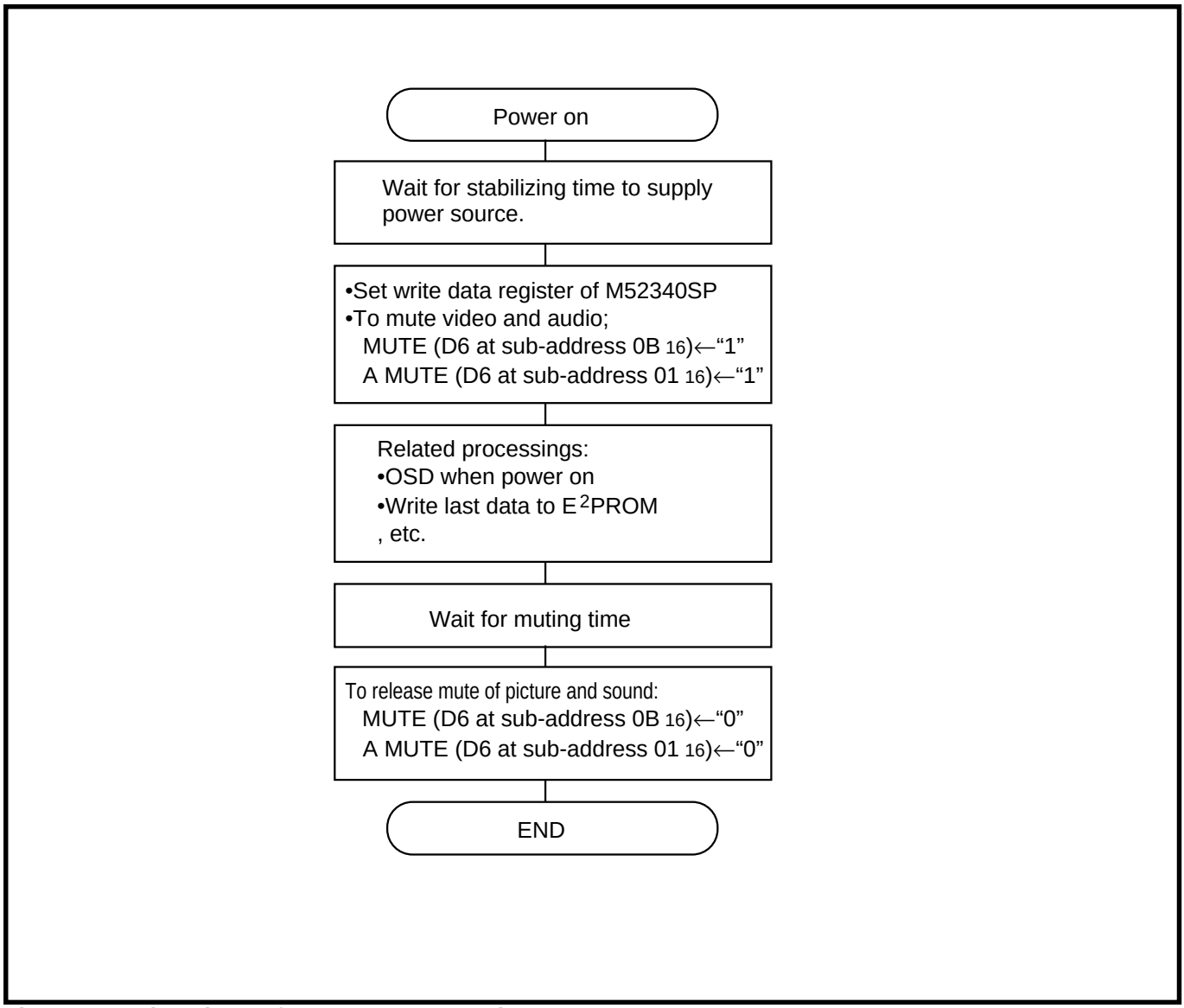


Fig. 5.5.13 Flowchart of poweron processing

(2) "CH UP/DOWN key" input processing

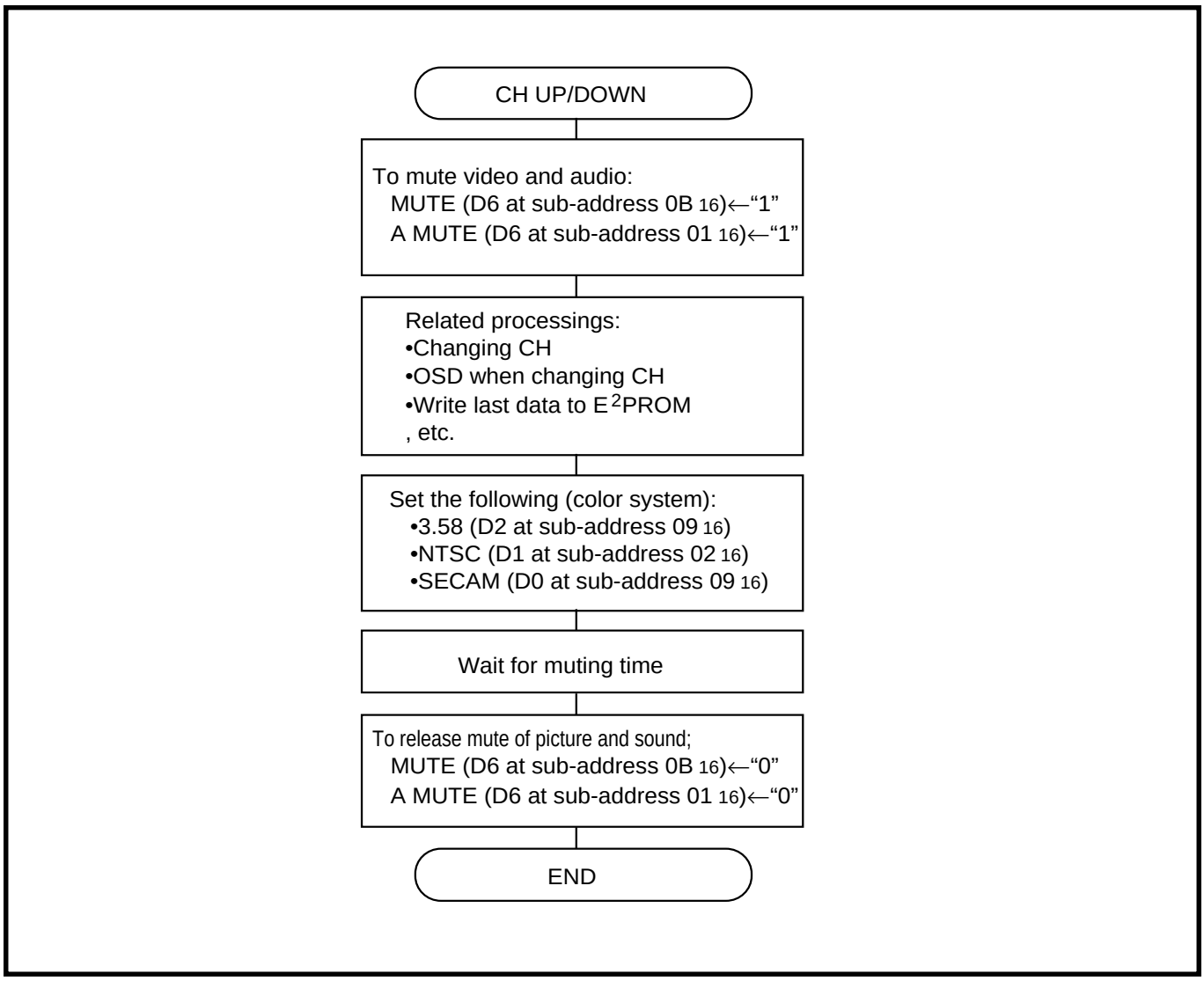


Fig. 5.5.14 Flowchart of "CH UP/DOWN key" input processing

5.5 Example of I²C-BUS control by software (M37220M3-XXXSP/FP)

(3) Processing of “picture memory switching key“ input

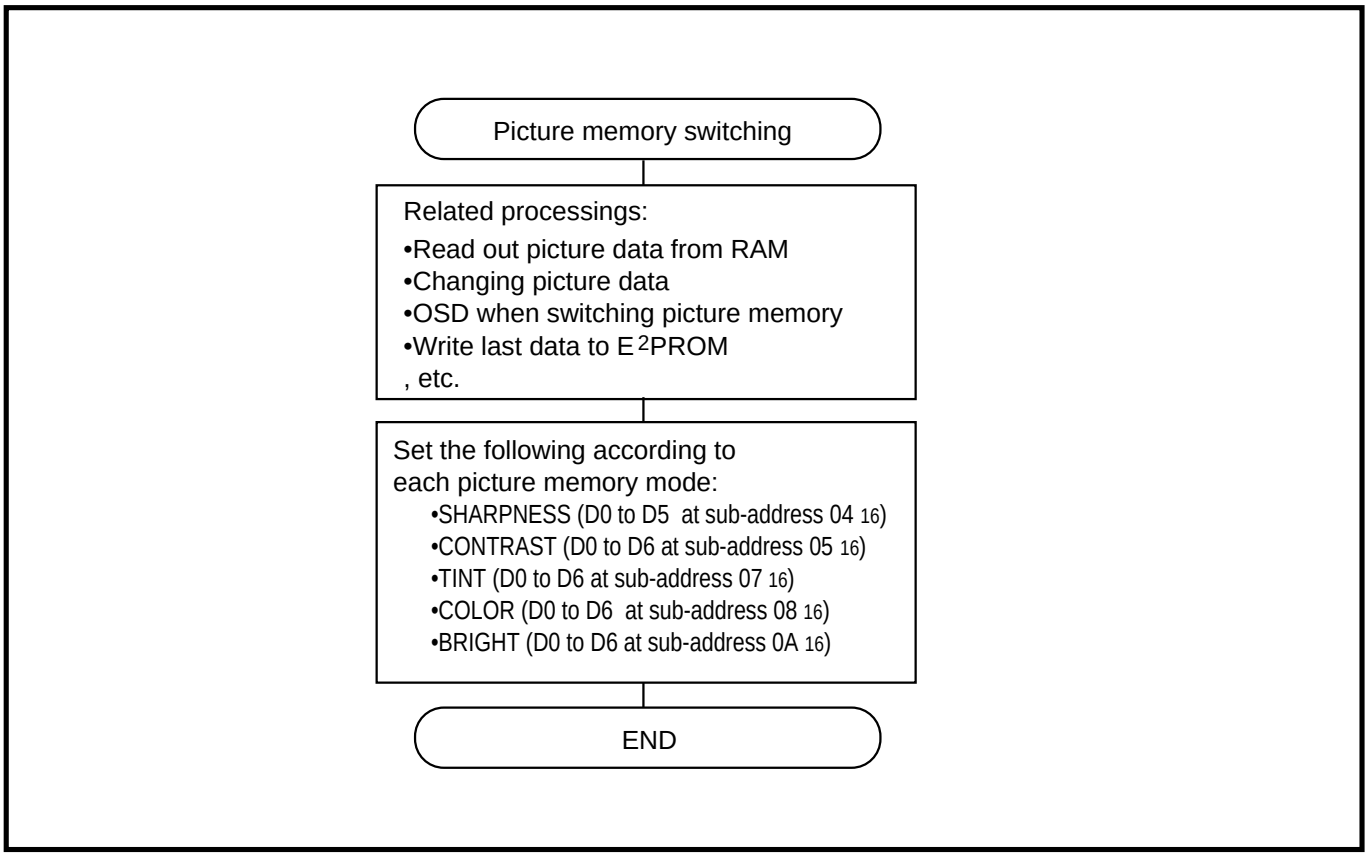


Fig. 5.5.15 Flowchart of “picture memory switching key” input processing

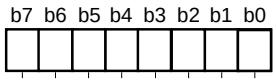
5.5.7 Register map

The M52340SP has 2 kinds of registers; the status data register and the write data registers.

(1) Status data register

The status data register indicates various signal state from the M52340SP side. The state is confirmed by regularly reading each bit.

Status data register



B	Name	Functions	After reset	R / W
0	Color system determination bit 0 (CONDITION)	0 : Under determination 1 : Determination is completed	0	R —
1	Color system determination bit 1 (SECAM)	b2 b1 Image system 0 0 PAL 0 1 SECAM 1 0 NTSC 1 1 SECAM	0	R W
2	Color system determination bit 2 (NTSC)		0	R W
3	3.58/4.43 determination bit (3.58)	0 : 4.43 MHz 1 : 3.58 MHz	0	R W
4	AFT signal detection bit 0 (AFT0)	Pin1 AFT OUT -100kHz fo +100kHz Freq. D5: AFT1 1 1 0 0 D4: AFT0 1 0 0 1	0	R W
5	AFT signal detection bit 1 (AFT1)		0	R W
6	Synchronous presence determination bit (COINCIDENCE)	0 : Without synchronization 1 : It is synchronization	0	R W
7	Frequency determination bit (50/60)	0 : 50 Hz 1 : 60 Hz	0	R W

Fig. 5.5.16 Status data register

5.5 Example of I²C-BUS control by software (M37220M3-XXXSP/FP)

■ **Bit 0: Color system determination bit 0 (CONDITION)**
This bit indicates whether the color system is being determined or not. Figure 5.5.16 shows the state of determination, according to the bit, when AUTO (bit 5 at sub-address 06_{16}, write data) is set to "1." When AUTO (bit 5 at sub-address 06_{16}, write data) is set to "0," bit 0 is invalid as the color system is not determined automatically.

■ **Bit 1: Color system determination bit 1 (SECAM)**
Bit 2: Color system determination bit 2 (NTSC)
These bits determine the color system.

■ **Bit 3: 3.58/4.43 determination bit (3.58)**
This bit determines whether a color signal sub-carrier of the color system is 3.58 MHz or 4.43 MHz.

■ **Bit 4: AFT signal detection bit 0 (AFT0)**
Bit 5: AFT signal detection bit 1 (AFT1)
These bits detect the level of the auto fine tuning signal.

■ **Bit 6: Synchronous presence determination bit (COINCIDENCE)**
This bit determines whether Pin H.OUT output is synchronized with the video signal or not.

■ **Bit 7: Field Frequency determination bit (50/60)**
This bit determines whether the field frequency is 50 Hz or 60 Hz. According to the state of this bit, the display position or a vertical direction size of video can be changed.

(2) Write data register

Write data register

Sub-address	D7	D6	D5	D4	D3	D2	D1	D0
00_{16}		POS/NEG	DELAY ADJ					
01_{16}		A MUTE	VCO ADJ					
02_{16}			TRAP	DBF			DFA	4.5/6.0
03_{16}		AUDIO ATT						
04_{16}		DEFEAT	SHARPNESS					
05_{16}		CONTRAST						
06_{16}			AUTO			TV/EXT	DL TIME	
07_{16}		TINT						
08_{16}		COLOR						
09_{16}		H PHASE				3.58	NTSC	SECAM
$0A_{16}$		BRIGHT						
$0B_{16}$		MUTE	DRIVE R					
$0C_{16}$			DRIVE B					
$0D_{16}$	CUT OFF R							
$0E_{16}$	CUT OFF G							
$0F_{16}$	CUT OFF B							
10_{16}			F TRAP					
11_{16}								
12_{16}								
13_{16}			AFCG	HST	SERSW			

: No function

Fig. 5.5.17 Map of write data register

5.5 Example of I²C-BUS control by software (M37220M3-XXXSP/FP)

■ **DELAY ADJ**

This adjusts the RF AGC delay point. The output level of tuner decreases when the value increase, the output level increases when the value decreases.

■ **POS/NEG**

This switch sets the VIF output signal to either the positive or the negative modulation signal. When "0," the negative modulation signal is selected; when "1," the positive modulation signal is selected.

■ **VCO ADJ**

This register changes the free running frequency of VIF VCO. The frequency increases when the value increases, the frequency decreases when the value decreases.

■ **A MUTE**

This is the audio mute ON/OFF.

■ **4.5/6.0**

This bit must be set to "1" when the sound carrier frequency is 4.5 MHz. Set "0" when the frequency is other values.

■ **DFA, DL TIME**

In order to adjust the color signal and the luminance signal is delayed using the on-chip delay-line. The DL TIME register adjusts the delay approximately, and the DFA register performes the fine adjustments. When DFA is "1," actual delay time is +50 ns; when "0," it is +0 ns. For relationship between DFA and DL TIME, refer to **"Table 5.5.9."**

Table 5.5.9 Relationship between DFA and DL TIME

Data	DL TIME1	DL TIME0	DFA	Actual delay time
0	0	0	0	170 ns
1	0	0	1	120 ns
2	0	1	0	330 ns
3	0	1	1	280 ns
4	1	0	0	410 ns
5	1	0	1	360 ns
6	1	1	0	490 ns
7	1	1	1	440 ns

■ **DBF**

The M52340SP has 2 TRAP; the second TRAP extends the bandwidth of the TRAP, described below. DBF is the ON/OFF switch for the second TRAP. When "1," it is on; when "0," it is off. DBF is used in SECAM and other methods.

■ **TRAP**

This is the TRAP ON/OFF switch for taking out the luminance signal (Y-signal) by Y/C separation (Y = Y-signal, C = color signal) of the composite video signal. When "1," it is on; when "0," it is off.

■ **AUDIO ATT**

Data is set ("0" to "127") to change the volume.

■ **SHARPNESS, CONTRAST, TINT, COLOR, BRIGHT**

Data is set to change the picture data.

Some TVs have a picture mode function (such as the movie mode, standard mode), the fixed data is set according to the mode. Accordingly, it is necessary to change the picture data when changing the picture mode.

■ **DEFEAT**

This switch turns DEFEAT off when AFT is on, and vice versa.

■ **TV/EXT**

This selects either a TV's signal or an external device's signal. This bit should be set to "0" when TV's signal is selected and set to "1" when an external device's signal is selected.

■ **AUTO**

This determines whether the automatic determination of the color system is used or not. When AUTO is "0," manual determination is set; when "1," determination is performed automatically.

■ **3.58/NTSC/SECAM**

When setting AUTO (bit 5 at sub-address 06_{16}, write data) to "1," these bits are automatically set inside the M52340SP. When setting AUTO to "0," it is necessary to set the data shown in Table 5.5.10, according to the color system.

Table 5.5.10 Setting of color system (at sub-address 09_{16}, write data)

Color system	D2	D1	D0
	3.58	NTSC	SECAM
PAL	0	0	0
SECAM	0	0	1
NTSC3.58	1	1	0
NTSC4.43	0	1	0

■ **H PHASE**

The picture's horizontal position is adjusted. Data is given every 50 Hz or 60 Hz and the data is set when frequency changes. For wide TVs etc., data is given for each screen size mode, and the data is set when the screen size mode changes.

■ **DRIVE R, DRIVE B**

Data is used to adjust the output amplitude ratio of R, G and B signals. Since G is the fixed data, its ratio is adjusted by R and B.

■ **MUTE**

This is the video mute ON/OFF switch.

■ **CUT OFF R, CUT OFF G, CUT OFF B**

Data is used to adjust the output DC level of R, G and B signals.

■ **F TRAP**

This register performes the fine adjustments to the trap frequency of TRAP for Y/C separation.

5.5 Example of I²C-BUS control by software (M37220M3-XXXSP/FP)

■ **SERSW**

This switch is for white balance adjustments of the TV picture in the factory. When SERSW is “0,” it is OFF; when “1,” it is ON.

■ **HST**

This switch stops horizontal oscillation. When HST is “0,” the oscillation continues; when “1,” it stops.

■ **AFCG**

This switch increases AFC gain. When AFCG is “0,” AFC gain is normal; when “1,” it is high.

5.6 Application circuit example

5.6.1 Application circuit example 1

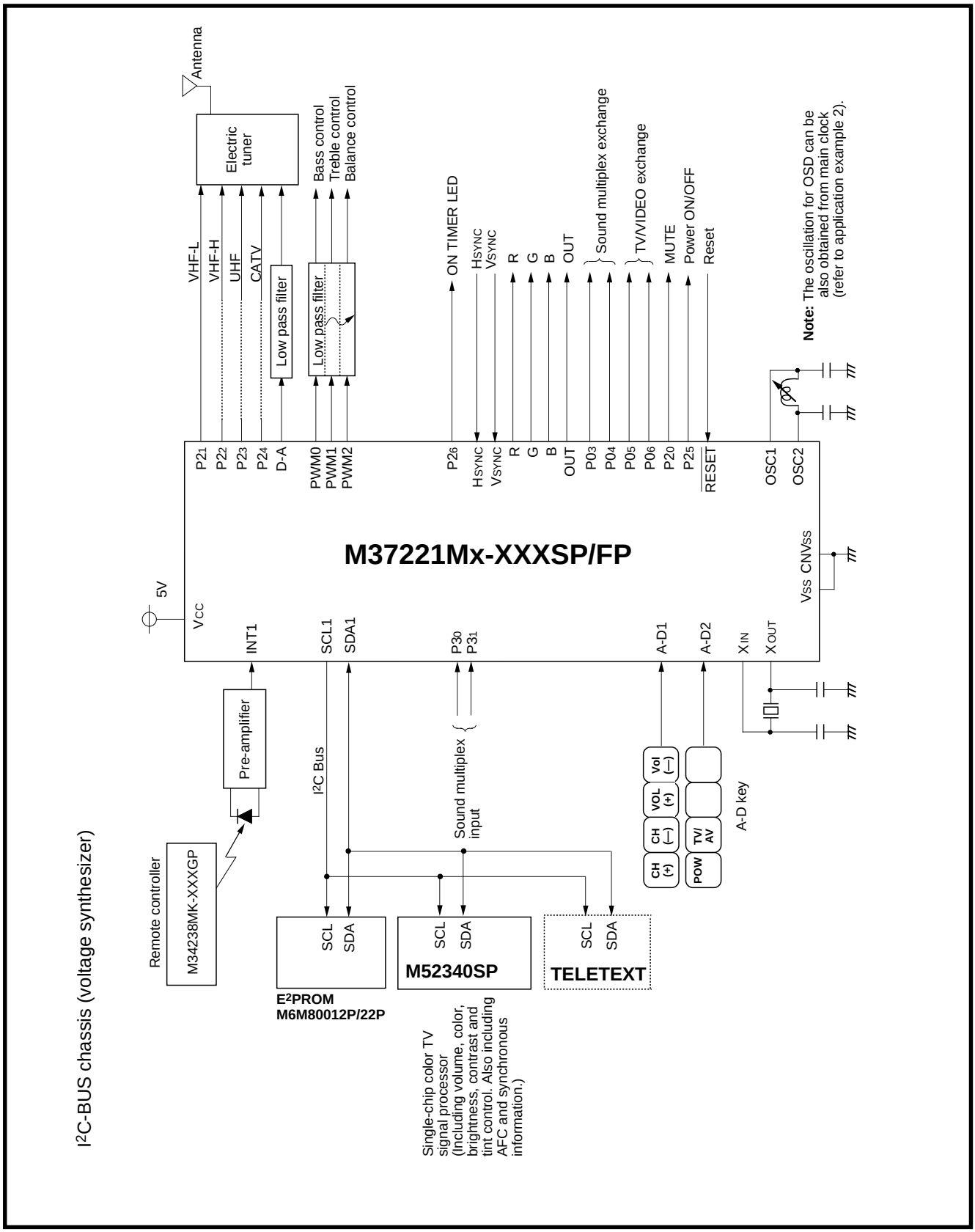


Fig. 5.6.1 Application circuit example 1 (I²C-BUS chassis)

5.6.2 Application circuit example 2

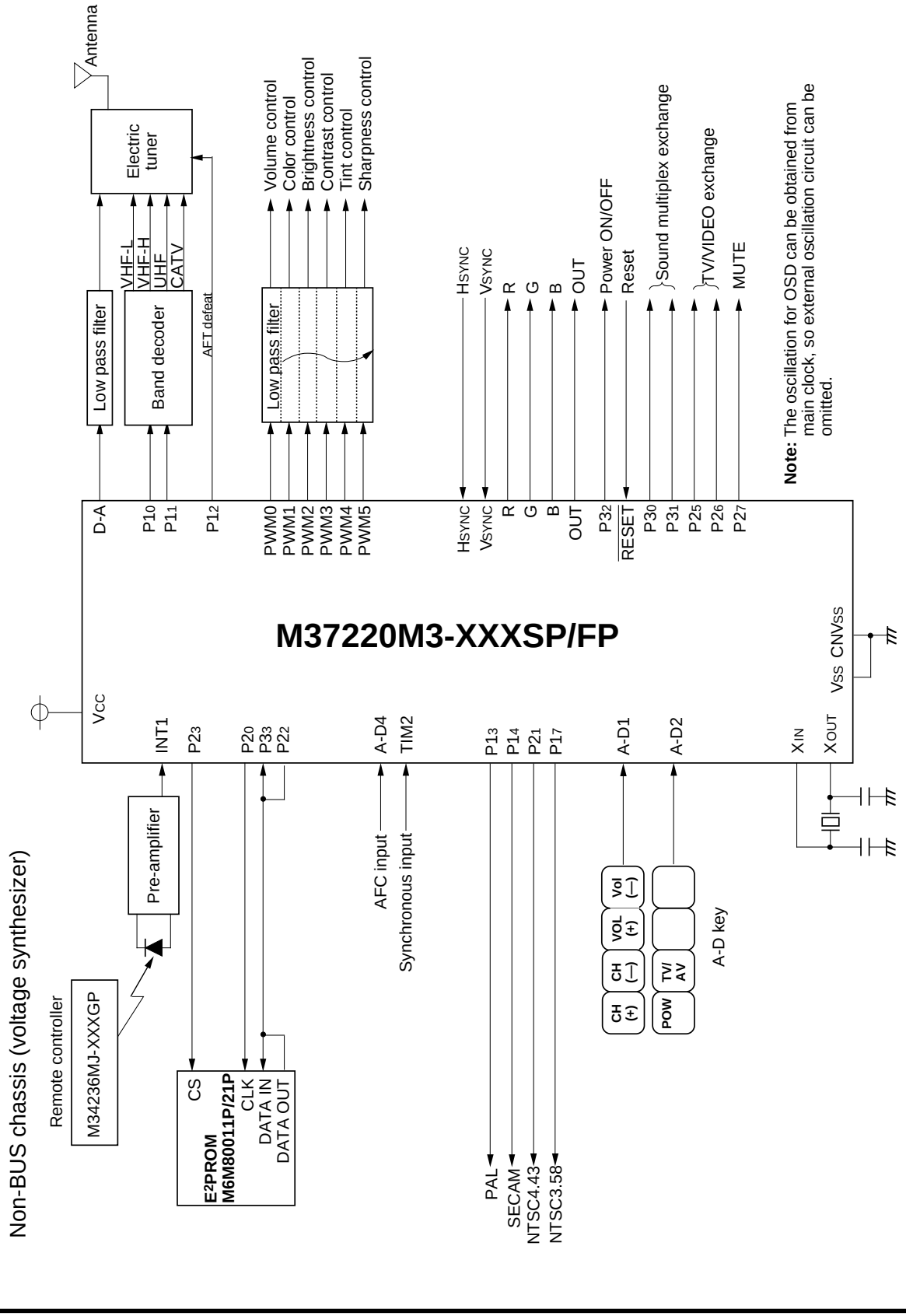


Fig. 5.6.1 Application circuit example 2 (Non-BUS chassis)

CHAPTER 6

APPENDIX

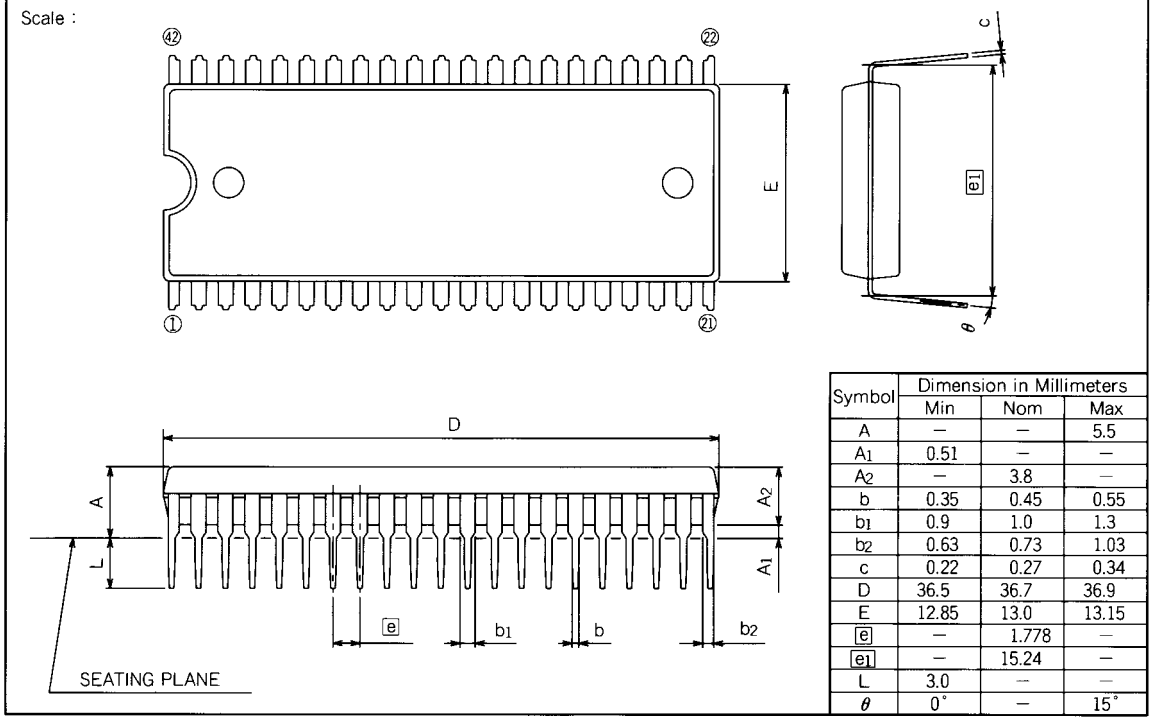
6.1 Package outline

42P4B

Plastic 42pin 600mil SDIP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
SDIP42-P-600-1.78	—	4.1	Alloy 42/Cu Alloy

Scale :



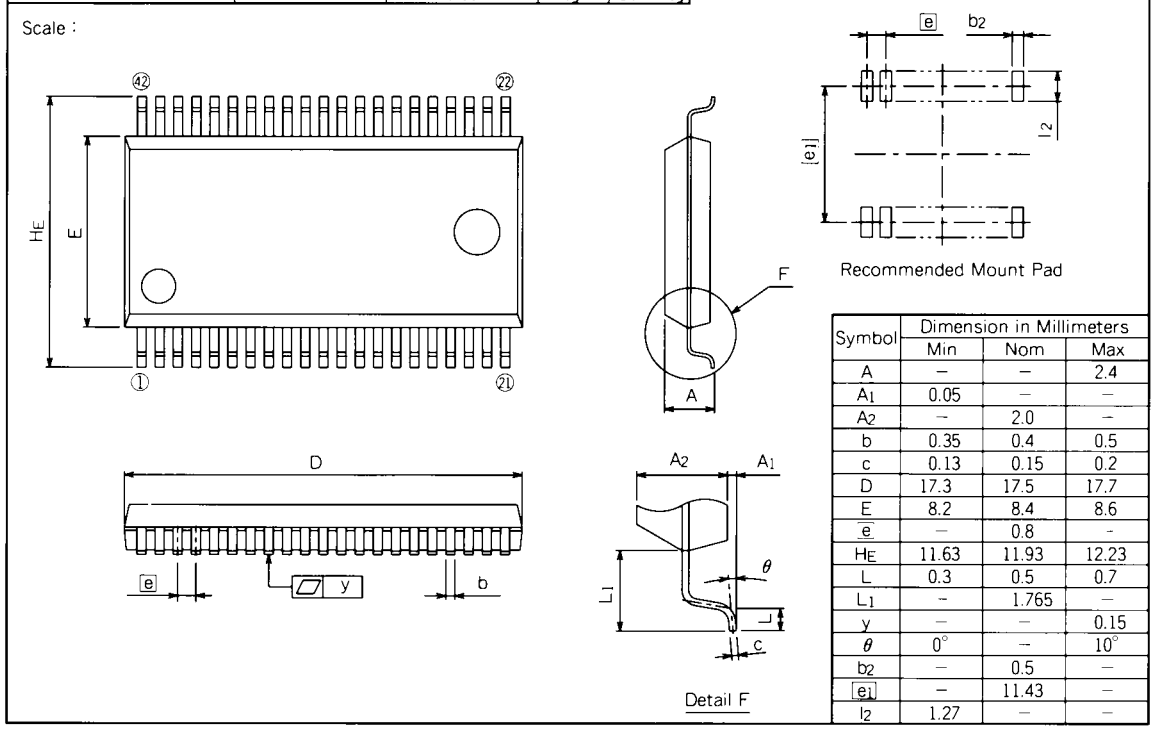
Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	—	—	5.5
A_1	0.51	—	—
A_2	—	3.8	—
b	0.35	0.45	0.55
b_1	0.9	1.0	1.3
b_2	0.63	0.73	1.03
c	0.22	0.27	0.34
D	36.5	36.7	36.9
E	12.85	13.0	13.15
e	—	1.778	—
e_1	—	15.24	—
L	3.0	—	—
θ	0°	—	15°

42P2R−A

Plastic 42pin 450mil SSOP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
SSOP42-P-450-0.80	—	0.63	Alloy 42/Cu Alloy

Scale :



Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	—	—	2.4
A_1	0.05	—	—
A_2	—	2.0	—
b	0.35	0.4	0.5
c	0.13	0.15	0.2
D	17.3	17.5	17.7
E	8.2	8.4	8.6
e	—	0.8	—
H_E	11.63	11.93	12.23
L	0.3	0.5	0.7
L_1	—	1.765	—
y	—	—	0.15
θ	0°	—	10°
b_2	—	0.5	—
e_1	—	11.43	—
l_2	1.27	—	—

6.2 Termination of unused pins

Table 6.2.1 Termination of unused pins

Pin		Input/ Output	Termination
M37221Mx-XXXSP/FP	M37220M3-XXXSP/FP		
$P0_0$/PWM0–$P0_5$/PWM5	✻	I/O	Set the port direction registers for the input mode and pull-down through a resistor.
$P0_6$/INT2/A-D4			
$P0_7$/INT1			
$P1_0$/OUT2	$P1_0$		
$P1_1$/SCL1	$P1_1$		
$P1_2$/SCL2	$P1_2$		
$P1_3$/SDA1	$P1_3$		
$P1_4$/SDA2	$P1_4$		
$P1_5$/A-D1/INT3	✻		
$P1_6$/A-D2			
$P1_7$/A-D3			
$P2_0$/S_{CLK}			
$P2_1$/S_{OUT}			
$P2_2$/S_{IN}			
$P2_3$/TIM3			
$P2_4$/TIM2			
$P2_5$–$P2_7$			
$P3_0$/A-D5	$P3_0$/A-D5/DA1		
$P3_1$/A-D6	$P3_0$/A-D6/DA2		
$P3_2$	✻		
$P3_3$/OSC1	✻	Input	Pull-down through a resistor.
$P3_4$/OSC2			
H_{SYNC}			
V_{SYNC}			
$P5_2$/R	✻	Output	Open
$P5_3$/G			
$P5_4$/B			
$P5_5$/OUT1	$P5_5$/OUT		
X_{OUT}	✻		
D-A			

✻ It is the same as M37221Mx-XXXSP/FP.

6.3 Notes on use

Notes on programming and equipping when using M37221M6-XXXSP/FP are described below.

6.3.1 Notes on processor status register

(1) Initialization of processor status register

The contents of processor status register (PS) are undefined except the I flag (I = "1") immediately after reset. Therefore initialize the flags that affect execution of a program.
Especially be sure to initialize the T and D flags because they have an important effect on calculations.

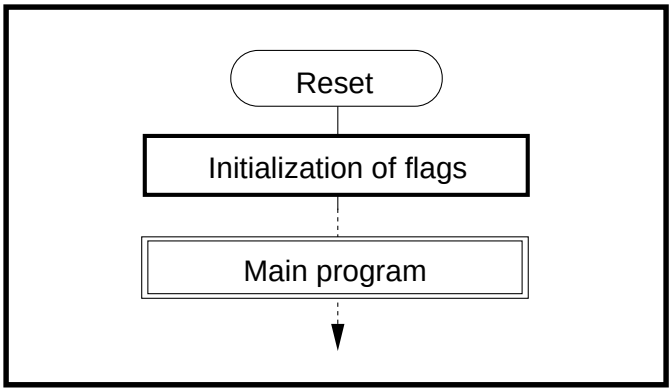


Fig. 6.3.1 Initialization of flags in PS

(2) How to refer to processor status register

When referring to the processor status register (PS) contents, execute the **PHP** instruction to push the processor status register contents into the stack (S) + 1. And then read the contents of stack (S) + 1.
If necessary, execute the **PLP** instruction to pull the pushed PS contents. In that case, be sure to execute the **NOP** instruction immediately after the **PLP** instruction.

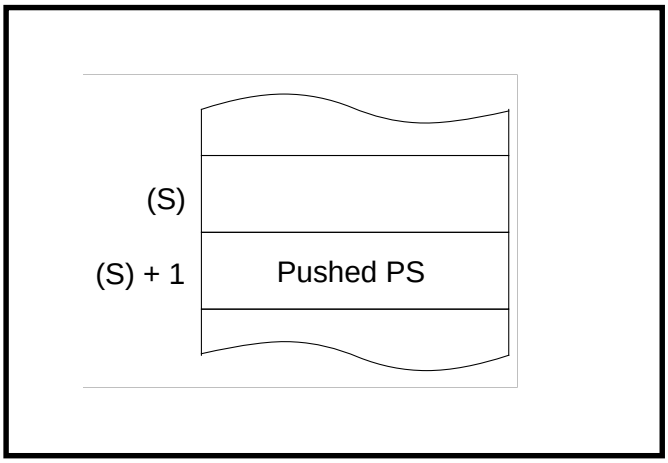


Fig. 6.3.2 Stack contents after PHP instruction execution

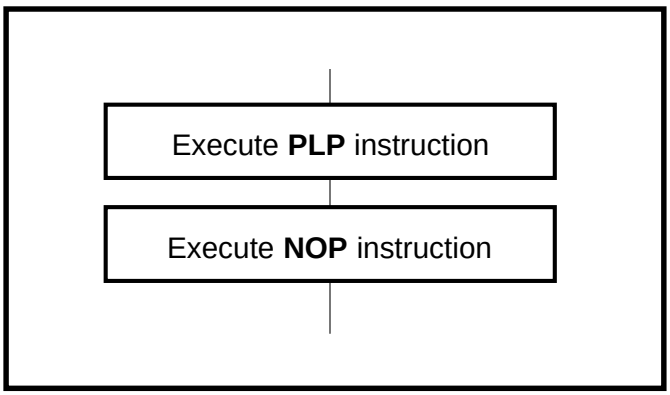


Fig. 6.3.3 Note when executing PLP instruction

6.3.2 Notes on decimal operation

(1) How to execute arithmetic operation instructions in decimal operation mode

To calculate in decimal notation, set the decimal operation mode flag (D) to "1" by using the **SED** instruction, and execute the **ADC** and **SBC** instructions. After that, execute at least one instruction to execute the **SEC**, **CLC,** or **CLD** instruction.

(2) Status flags in decimal operation mode

When the **ADC** or **SBC** instruction are executed in decimal operation mode (D flag = "1"), the N, V, and Z flags are invalid.

The carry flag (C) is set to "1" when a carry occurs as a result of an arithmetic operation, or is cleared to "0" when a borrow occurs. Therefore, the carry flag can be used to determine whether a carry or a borrow has occurred or not. Be sure to initialize the C flag before each arithmetic operation.

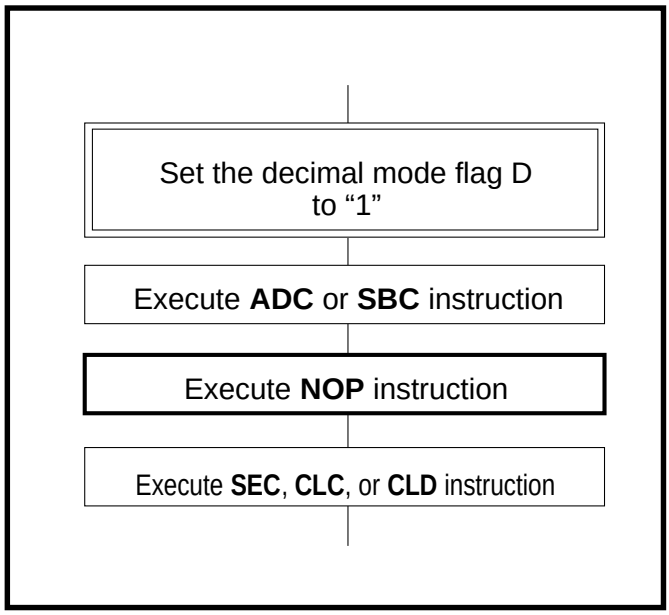


Fig. 6.3.4 Note in decimal arithmetic operation

6.3.3 Notes on Interrupts

(1) Executing BBC or BBS instruction

When executing the **BBC** or **BBS** instruction to an interrupt request bit immediately after this bit is set to "0" by using a data transfer instruction*1, 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.

*1: data transfer instructions: **LDM**, **LDA**, **STA**, **STX**, and **STY** instructions

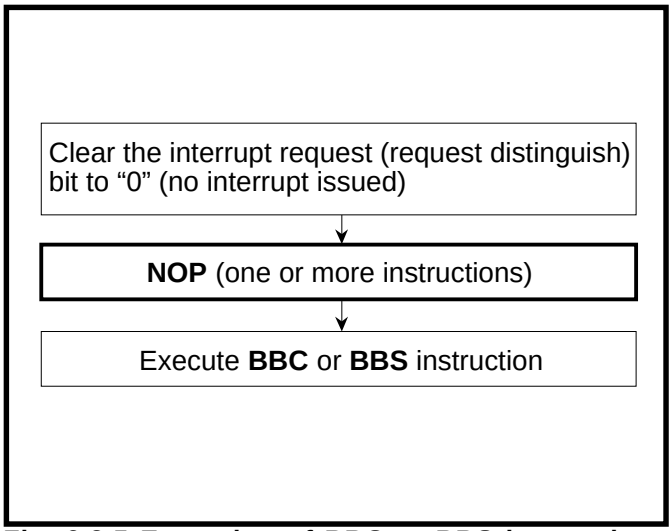


Fig. 6.3.5 Execution of BBC or BBS instruction

(2) How to switch an external interrupt detection edge

For the products able to switch the external interrupt detection edge, switch it as Figure 6.3.6.

Reason

The interrupt circuit recognizes the switching of the detection edge as the change of external input signals. This may cause an unnecessary interrupt.

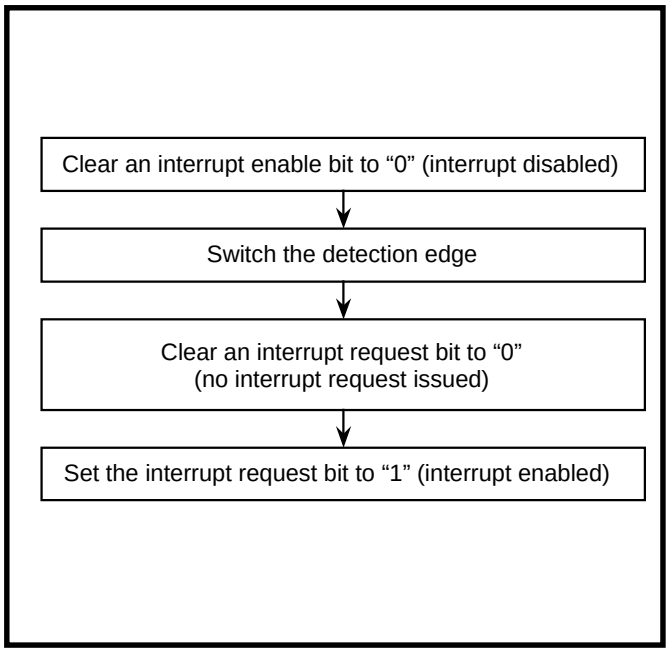


Fig. 6.3.6 Sequence for switching an external interrupt detection edge

6.3.4 Notes on serial I/O

(1) Initialization for the serial I/O

For the serial I/O interrupt, initialize as Figure 6.3.7.

(2) Write transmit data to transmit buffer

When an external clock is used as the synchronous clock for the clock synchronous serial I/O, write the transmit data to the serial I/O shift register at HIGH of the transfer clock input level.

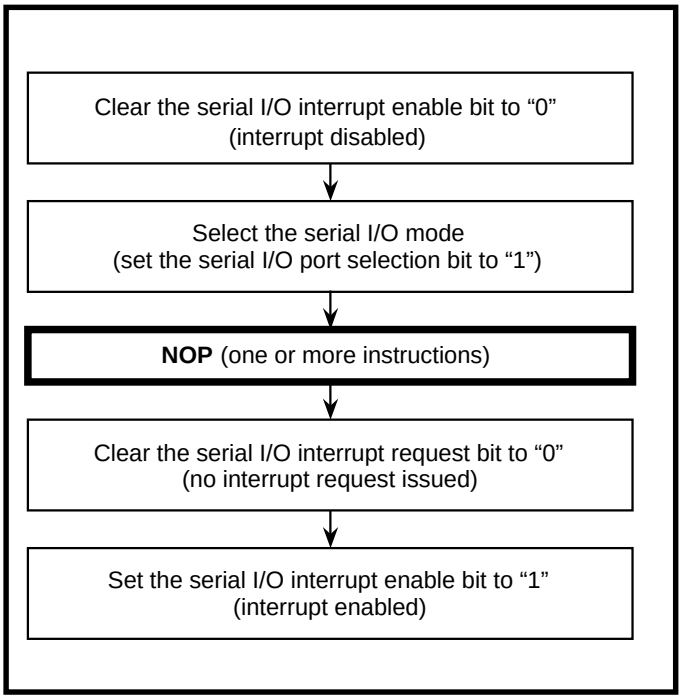


Fig. 6.3.7 Initialization for serial I/O

6.3.5 Notes on timer

When a timer value is read, “the timer value at read timing + 1” may be read.

Reason

Figure 6.3.8 shows the relation between timer values and their values read. Timer values are changed at the rising edge of the count source, but the values read are counted down at the falling edge of the count source. Therefore, “the timer value + 1” may be read in some read timings.

Figure 6.3.9 shows the relation between timer values and their values read when two 8-bit timers are connected in series. In this example, timers 1 and 2 are connected in series and an overflow signal of timer 1 is used as the count source of timer 2. The timer 2 values read are counted down at the falling edge of the count source. When timers 1 and 2 are used as a single 16-bit counter, the timer 2 values read take the same value at timing A and B (or at timing C and D) as shown in Figure 6.3.9. This is because the count source of timer 2 changes at the falling edge of the count source of timer 1.

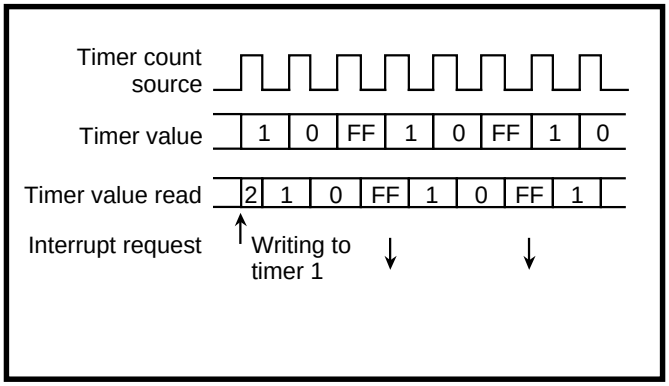


Fig. 6.3.8 Relation between timer values and their values read (timer setting value = 2)

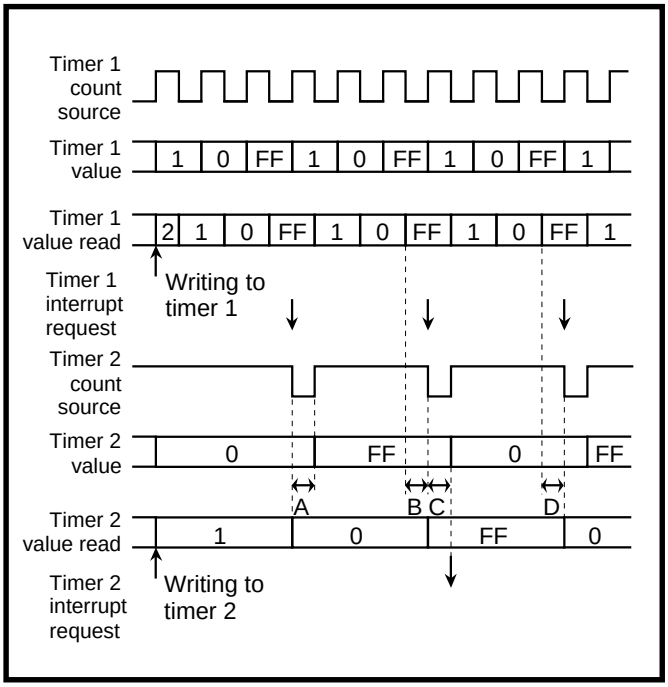


Fig. 6.3.9 Relation between timer values and their values read when two timers are connected in series (timers 1 and 2 are connected, timer 1 setting value = 2, timer 2 setting value = 1)

6.3 Notes on use

6.3.6 Notes on A-D comparator

(1) Signal source impedance for analog input

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 comparison precision to be worse.

(2) Note during an 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 comparison.

- $f(X_{IN})$ is 500 kHz or more
- Do not execute the **STP** instruction and **WIT** instruction

6.3.7 Note on $\overline{\text{RESET}}$ pin

In case where the $\overline{\text{RESET}}$ signal rise time is long, connect a ceramic capacitor or others across the $\overline{\text{RESET}}$ pin and the V_{SS} pin. And use a 1000 pF or more capacitor for high frequency use. When connecting the capacitor, note the following :

- Make the length of the wiring which is connected to a capacitor as short as possible.
- Be sure to check 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.

6.3.8 Notes on input and output pins

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

In stand-by state*2 for low-power dissipation, do not make input levels of an input port and an I/O port "undefined," especially for I/O ports of the P-channel and the N-channel open-drain.
Pull-up (connect the port to V_{CC}) or pull-down (connect the port to V_{SS}) these ports through a resistor. When determining a resistance value, note the following points:

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

When using built-in pull-up or pull-down resistor, note on varied current values.

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

Reason

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

- N-channel.....when the content of the port latch is "1"
  the transistor becomes the OFF state, which causes the ports to be the high-impedance state. Note that the level becomes "undefined" depending on external circuits.
  Accordingly, the potential which is input to the input buffer in a microcomputer is unstable in the state that input levels of a input port and an I/O port are "undefined." This may cause power source current.

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

(2) Modify of the contents of I/O port latch

When the port latch of an I/O port is modified with the bit managing instruction*3, the value of the unspecified bit may be changed.

Reason

The bit managing instructions*3 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 data register of an I/O port, the following is executed to all bits of the data register.

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

✻3 bit managing instructions : **SEB,** and **CLB** instruction

6.3.9 Note on JMP instruction

When using the **JMP** instruction (the indirect addressing mode), do not specify the last address in a page as an indirect address.
Memory (addresses 0000_{16} to $FFFF_{16}$) is separated into pages (by each 256 address).

6.3 Notes on use

6.3.10 Note on multi-master I²C-BUS interface

This function is used at f(X_{IN}) = 8.0 MHz of oscillation frequency.

6.3.11 Termination of unused pins

(1) Proper termination of unused pins

■ **Output ports : Open**

■ **Input ports :**

Connect each pin to V_{CC} or V_{SS} through each resistor of 1 kW to 10 kW.

Ports that permit the selecting of a built-in pull-up or pull-down resistor can also use this resistor. As for pins whose potential affects to operation modes such as pins CNV_{SS}, INT or others, select the V_{CC} pin or the V_{SS} pin according to their operation mode.

■ **I/O ports :**

•Set the I/O ports for the input mode and connect them to V_{CC} or V_{SS} through each resistor of 1 kΩ to 10 kΩ. 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.

(2) Incorrect termination of unused pins

■ **input ports and I/O ports :**

Do not open in the input mode.

Reason

•The power supply current may increase depending on the first-stage circuit.

•An effect due to noise may be easily produced as compared with proper termination (1). shown on the above.

■ **I/O ports :**

Set for input mode and do not connect to V_{CC} or V_{SS} directly.

Reason

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

■ **I/O ports :**

Set for the input mode and do not connect multiple ports in a lump to V_{CC} or V_{SS} through a resistor.

Reason

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

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

6.4 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.

6.4.1 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 reset input pin

Make the length of wiring which is connected to the RESET input pin as short as possible. Especially, connect a capacitor across the RESET input pin and the V_{SS} pin with the shortest possible wiring (within 20mm).

Reason

The width of a pulse input into the RESET pin is determined by the timing necessary conditions. If noise having a shorter pulse width than the standard is input to the RESET input pin, the reset is released before the internal state of the microcomputer is completely initialized. This may cause a program runaway.

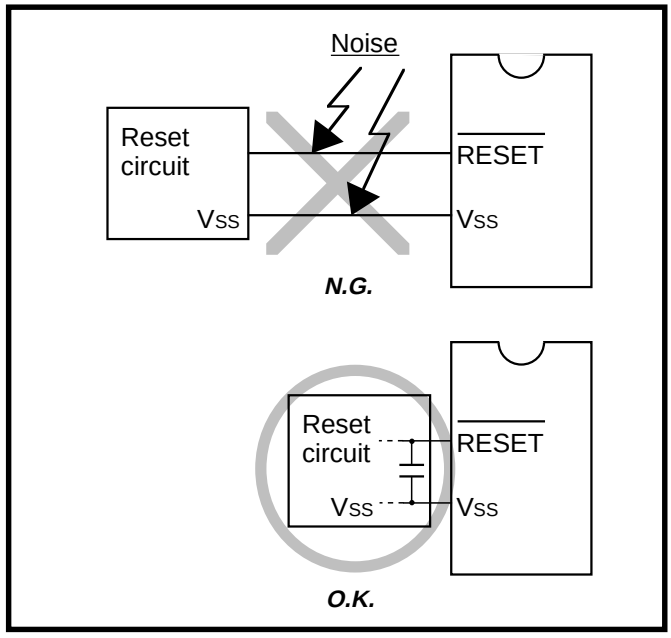


Fig.6.4.1 Wiring for RESET input pin

(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 20mm) across the grounding lead of a capacitor which is connected to an oscillator and the V_{SS} pin of a microcomputer as short as possible.
- Separate the V_{SS} pattern only for oscillation from other V_{SS} 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 V_{SS} level of a microcomputer and the V_{SS} level of an oscillator, the correct clock will not be input in the microcomputer.

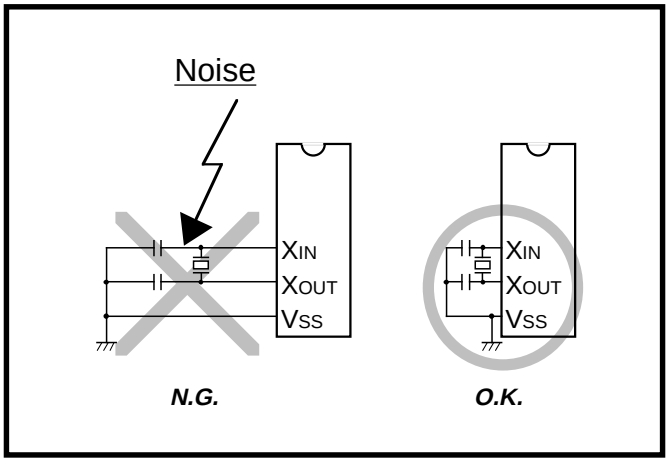


Fig.6.4.2 Wiring for clock I/O pin

6.4 Countermeasures against noise

(3) Wiring to CNVss pin

Connect the CNV_{SS} pin to the V_{SS} pin with the shortest possible wiring.

Reason

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

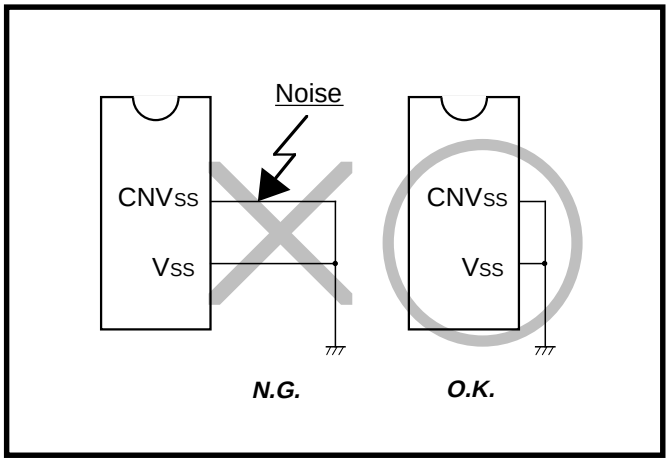


Fig.6.4.3 Wiring for CNV_{SS} pin

(4) Wiring to V_{PP} pin of One Time PROM version and EPROM version

When the V_{PP} pin is also used as the CNV_{SS} pin*1

Connect an approximately 5 kW resistor to the V_{PP} pin the shortest possible in series and also to the V_{SS} pin. When not connecting the resistor, make the length of wiring between the V_{PP} pin and the V_{SS} pin the shortest possible (refer to **"countermeasure example 1 of Figure 6.4.4"**)

✻1 When a microcomputer has the CNV_{SS} pin, the V_{PP} pin is also used as the CNV_{SS} pin.

Note: Even when a circuit which included an approximately 5 kW resistor is used in the Mask ROM version, the microcomputer operates correctly.

Reason

The V_{PP} pin of the One Time PROM and the EPROM version is the power source input pin for the built-in PROM. When programming in the built-in PROM, the impedance of the V_{PP} pin is low to allow the electric current for writing flow into the PROM. Because of this, noise can enter easily. If noise enters the V_{PP} pin, abnormal instruction codes or data are read from the built-in PROM, which may cause a program runaway.

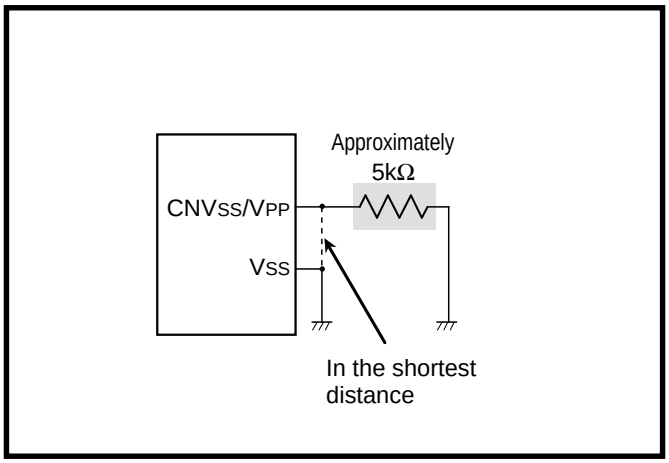


Fig.6.4.4 Wiring for V_{PP} pin of One Time PROM and EPROM version

6.4.2 Connection of a bypass capacitor across V_{SS} line and V_{CC} line

Connect an approximately 0.1 μF bypass capacitor across the V_{SS} line and the V_{CC} line as follows:

- Connect a bypass capacitor across the V_{SS} pin and the V_{CC} pin at equal length.
- Connect a bypass capacitor across the V_{SS} pin and the V_{CC} pin with the shortest possible wiring.
- Use lines with a larger diameter than other signal lines for V_{SS} line and V_{CC} line.

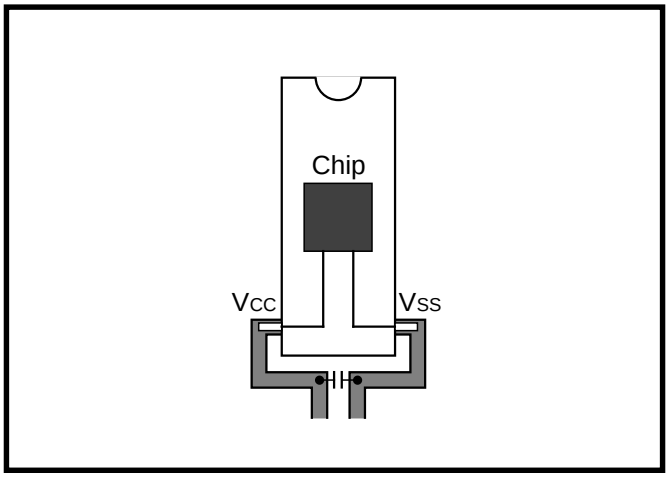


Fig.6.4.5 Bypass capacitor across V_{SS} line and V_{CC} line

6.4.3 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 V_{SS} pin and the analog input pin. Besides, connect the capacitor to the V_{SS} pin as close as possible. Also, connect the capacitor across the analog input pin and the V_{SS} 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.

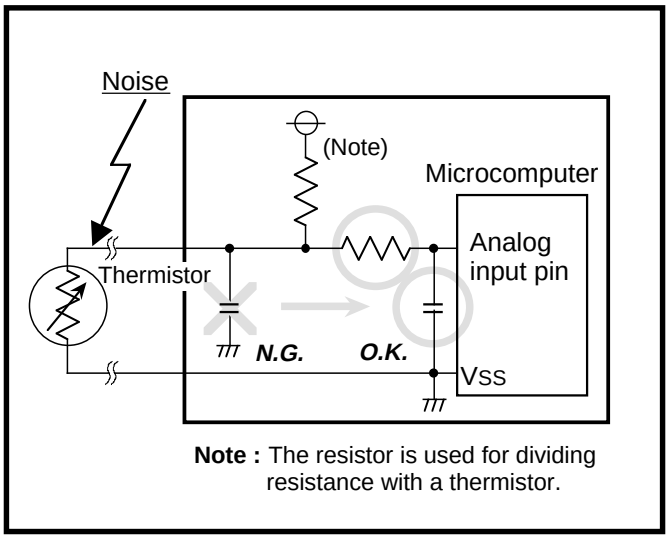


Fig.6.4.6 Analog signal line and resistor and capacitor

6.4 Countermeasures against noise

6.4.4 Oscillator concerns

Take care to prevent an oscillator that generates clocks for a microcomputer operation from being affected by other signals.

(1) Keeping an oscillator away from large current signal lines

Install a microcomputer (and especially an oscillator) as far as possible from signal lines where a current larger than the tolerance of current value flows.

Reason

In the system using a microcomputer, there are signal lines for controlling motors, LEDs, and thermal heads or others. When a large current flows through those signal lines, strong noise occurs because of mutual inductance.

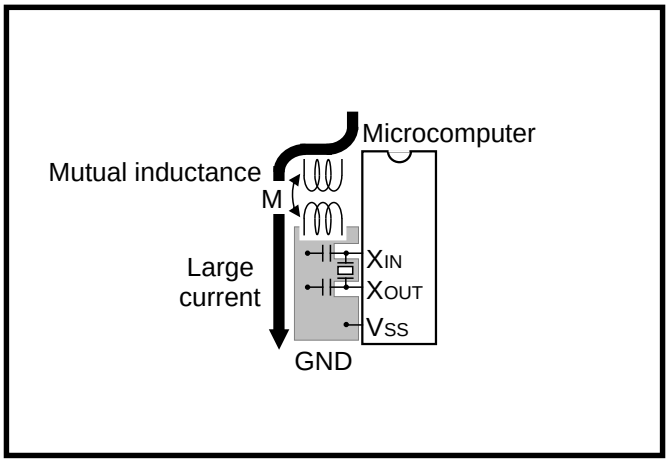


Fig.6.4.7 Wiring for large current signal line

(2) Installing an 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.

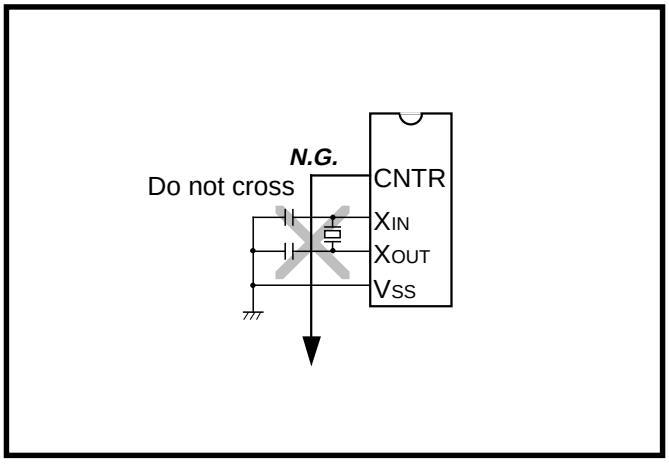


Fig.6.4.8 Wiring for signal line where potential levels charge frequently

(3) Oscillator protection using V_{SS} pattern

As for a two-sided printed circuit board, print a V_{SS} pattern on the underside (soldering side) of the position (on the component side) where an oscillator is mounted.

Connect the V_{SS} pattern to the microcomputer V_{SS} pin with the shortest possible wiring. Besides, separate this V_{SS} pattern from other V_{SS} patterns.

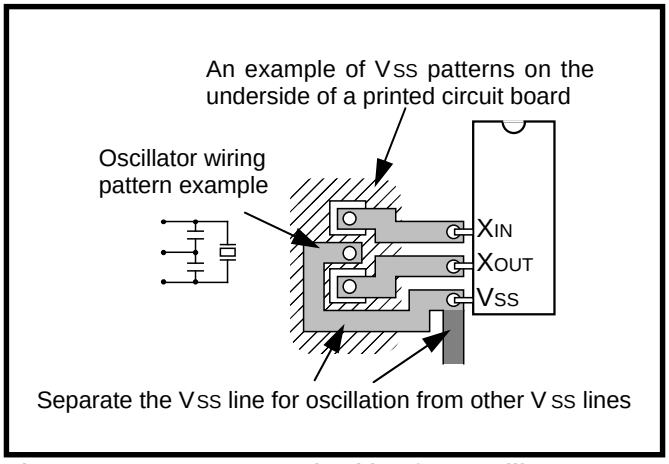


Fig.6.4.9 V_{SS} pattern on underside of an oscillator

6.4.5 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 data register at fixed periods.
- Rewrite data to direction registers and pull-up control registers (only the product having it) at fixed periods.

When a direction register is set for input port again at fixed periods, a several-nanosecond short pulse may be output from this port. If this is undesirable, connect a capacitor to this port to remove the noise pulse.

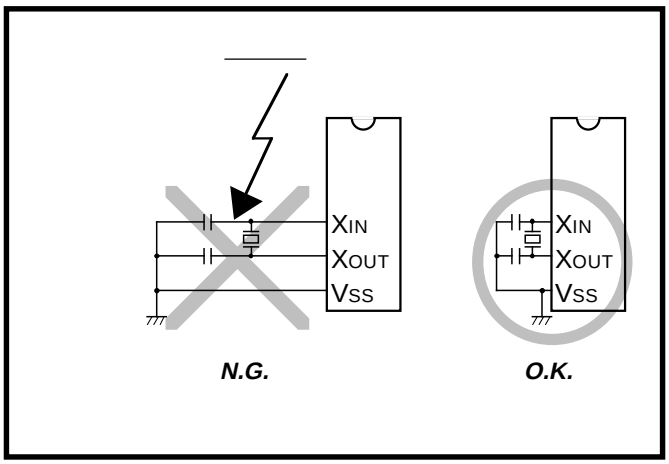


Fig. 6.4.10 Setup for I/O ports

6.4 Countermeasures against noise

6.4.6 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.
- Determins 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.

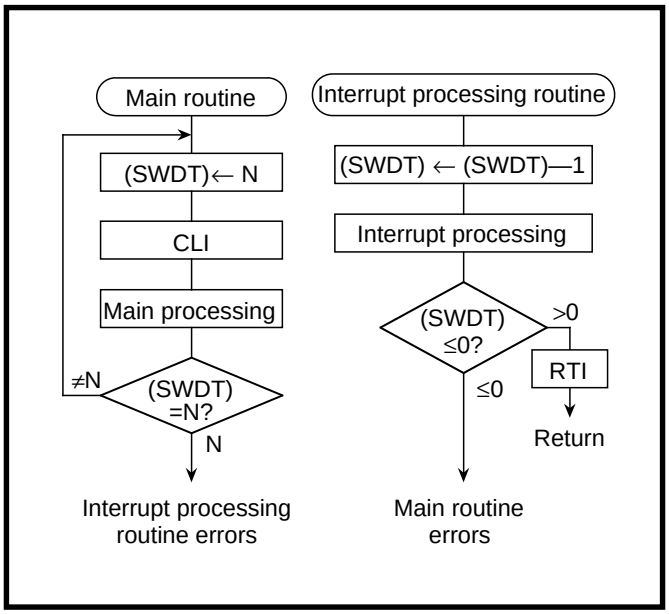
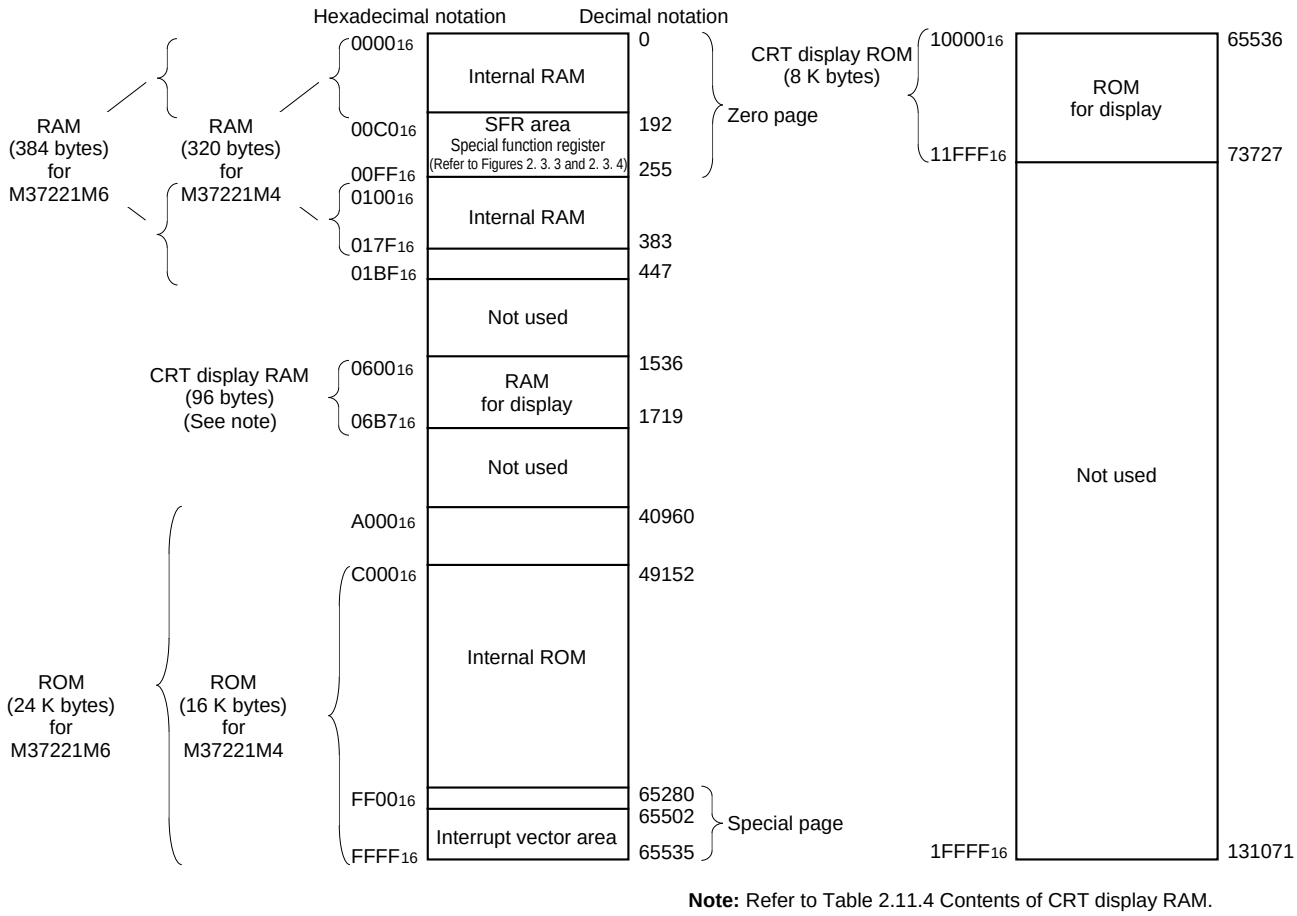


Fig. 6.4.11 Watchidog timer by software

6.5 Memory assignment



Note: Refer to Table 2.11.4 Contents of CRT display RAM.

Fig. 6.5.1 Memory assignment of M37221M4-XXXSP and M37221M6-XXXSP/FP

6.5 Memory assignment

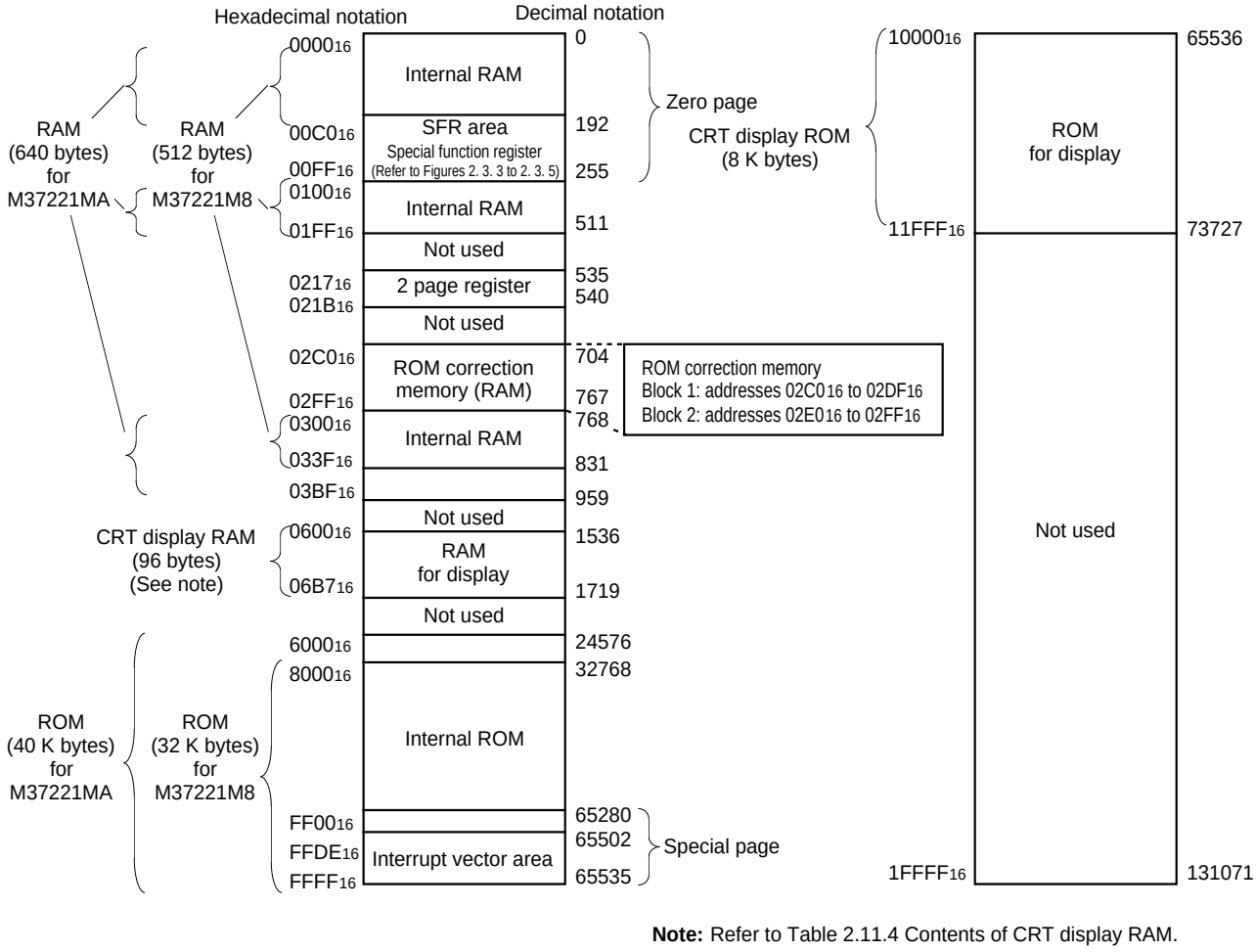


Fig. 6.5.2 Memory assignment of M37221M8-XXXSP and M37221MA-XXXSP

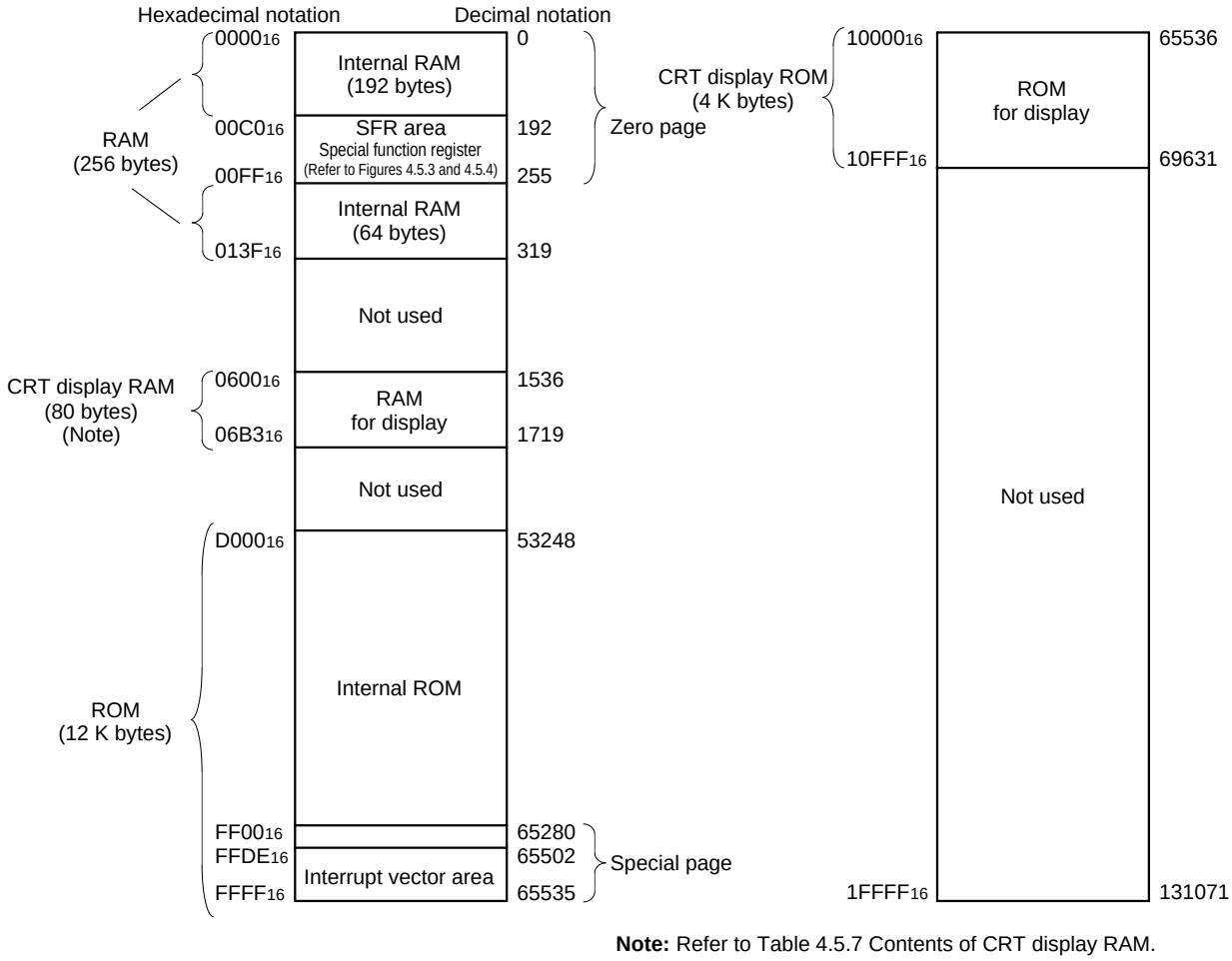


Fig. 6.5.3 Memory assignment of M37220M3-XXXSP/FP

6.6 SFR assignment

■SFR Area (addresses $C0_{16}$ to DF_{16})

<Bit allocation>

: } Function bit
Name :

: No function bit

0 : Fix this bit to "0" (do not write "1")

1 : Fix this bit to "1" (do not write "0")

Address	Register	Bit allocation b7							b0
$C0_{16}$	Port P0 (P0)								
$C1_{16}$	Port P0 direction register (D0)								
$C2_{16}$	Port P1 (P1)								
$C3_{16}$	Port P1 direction register (D1)								
$C4_{16}$	Port P2 (P2)								
$C5_{16}$	Port P2 direction register (D2)								
$C6_{16}$	Port P3 (P3)								
$C7_{16}$	Port P3 direction register (D3)								
$C8_{16}$									
$C9_{16}$									
CA_{16}	Port P5 (P5)								
CB_{16}	Port P5 direction register (D5)								
CC_{16}									
CD_{16}	Port P3 output mode control register (P3S)					0	0	P31S	P30S
CE_{16}	DA-H register (DA-H)								
CF_{16}	DA-L register (DA-L)								
$D0_{16}$	PWM0 register (PWM0)								
$D1_{16}$	PWM1 register (PWM1)								
$D2_{16}$	PWM2 register (PWM2)								
$D3_{16}$	PWM3 register (PWM3)								
$D4_{16}$	PWM4 register (PWM4)								
$D5_{16}$	PWM output control register 1 (PW)	PW7	PW6	PW5	PW4	PW3	PW2	PW1	PW0
$D6_{16}$	PWM output control register 2 (PN)				PN4	PN3	PN2		
$D7_{16}$	I^2C data shift register (S0)	D7	D6	D5	D4	D3	D2	D1	D0
$D8_{16}$	I^2C address register (S0D)	SAD6	SAD5	SAD4	SAD3	SAD2	SAD1	SAD0	RBW
$D9_{16}$	I^2C status register (S1)	MST	TRX	BB	PIN	AL	AAS	AD0	LRB
DA_{16}	I^2C control register (S1D)	BSEL1	BSEL0	10 BIT SAD	ALS	ES0	BC2	BC1	BC0
DB_{16}	I^2C clock control register (S2)	ACK	ACK BIT	FAST MODE	CCR4	CCR3	CCR2	CCR1	CCR0
DC_{16}	Serial I/O mode register (SM)		SM6	SM5	0	SM3	SM2	SM1	SM0
DD_{16}	Serial I/O register (SIO)								
DE_{16}		00_{16}							
DF_{16}		00_{16}							

Fig. 6.6.1 SFR assignment (including internal state immediately after reset and access characteristics) (1) (M37221Mx-XXXSP/FP)

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately after reset

RW : Read enabled, write enabled

RO : Read enabled, write disabled

State immediately after reset

b7							b0
?							
00_{16}							
?							
00_{16}							
?							
00_{16}							
0	0	0	?	?	?	?	?
00_{16}							
?							
?							
0	0	?	?	?	?	?	?
00_{16}							
?							
00_{16}							
?							
0	0	?	?	?	?	?	?
?							
?							
?							
?							
?							
00_{16}							
00_{16}							
?							
00_{16}							
0	0	0	1	0	0	0	?
00_{16}							
00_{16}							
00_{16}							
?							
00_{16}							
00_{16}							

Access characteristics

b7							b0
RW							
RW							
RW							
RW							
RW							
RW							
			RW				
					RW		
		RW					
		RW					
						RW	
RW							
		RW					
RW							
RW							
RW							
RW							
RW							
RW							
			RW				
RW							
RW							
RW		RO					
RW							
RW							
	RW			RW			
RW							

■SFR Area (addresses $E0_{16}$ to FF_{16})

<Bit allocation>

: Function bit

: No function bit

0 : Fix this bit to "0" (do not write "1")

1 : Fix this bit to "1" (do not write "0")

Address	Register	Bit allocation b7							b0
$E0_{16}$	Horizontal position register (HR)			HR5	HR4	HR3	HR2	HR1	HR0
$E1_{16}$	Vertical position register 1 (CV1)		CV16	CV15	CV14	CV13	CV12	CV11	CV10
$E2_{16}$	Vertical position register 2 (CV2)		CV26	CV25	CV24	CV23	CV22	CV21	CV20
$E3_{16}$									
$E4_{16}$	Character size register (CS)					CS21	CS20	CS11	CS10
$E5_{16}$	Border selection register (MD)						MD20		MD10
$E6_{16}$	Color register 0 (CO0)	CO07	CO06	CO05	CO04	CO03	CO02	CO01	
$E7_{16}$	Color register 1 (CO1)	CO17	CO16	CO15	CO14	CO13	CO12	CO11	
$E8_{16}$	Color register 2 (CO2)	CO27	CO26	CO25	CO24	CO23	CO22	CO21	
$E9_{16}$	Color register 3 (CO3)	CO37	CO36	CO35	CO34	CO33	CO32	CO31	
EA_{16}	CRT control register (CC)	CC7					CC2	CC1	CC0
EB_{16}									
EC_{16}	CRT port control register (CRTP)	OP7	OP6	OP5	OUT1	OUT2	R/G/B	VSYC	HSYC
ED_{16}	CRT clock selection register (CK)	0	0	0	0	0	0	CK1	CK0
EE_{16}	A-D control register 1 (AD1)				ADM4		ADM2	ADM1	ADM0
EF_{16}	A-D control register 2 (AD2)			ADC5	ADC4	ADC3	ADC2	ADC1	ADC0
$F0_{16}$	Timer 1 (TM1)								
$F1_{16}$	Timer 2 (TM2)								
$F2_{16}$	Timer 3 (TM3)								
$F3_{16}$	Timer 4 (TM4)								
$F4_{16}$	Timer 12 mode register (T12M)			0	T12M4	T12M3	T12M2	T12M1	T12M0
$F5_{16}$	Timer 34 mode register (T34M)			T34M5	T34M4	T34M3	T34M2	T34M1	T34M0
$F6_{16}$	PWM5 register (PWM5)								
$F7_{16}$									
$F8_{16}$									
$F9_{16}$	Interrupt input polarity register (RE)	0		RE5	RE4	RE3	0	0	
FA_{16}		00_{16}							
FB_{16}	CPU mode register (CPUM)	1	1	1	1	1	CM2	0	0
FC_{16}	Interrupt request register 1 (IREQ1)	IT3R	IICR	VSCR	CRTR	TM4R	TM3R	TM2R	TM1R
FD_{16}	Interrupt request register 2 (IREQ2)	0			MSR		S1R	1T2R	1T1R
FE_{16}	Interrupt control register 1 (ICON1)	IT3E	IICE	VSCE	CRTE	TM4E	TM3E	TM2E	TM1E
FF_{16}	Interrupt control register 2 (ICON2)	0	0	0	MSE	0	S1E	1T2E	1T1E

Fig. 6.6.2 SFR assignment (including internal state immediately after reset and access characteristics) (2) (M37221Mx-XXXSP/FP)

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately after reset

RW : Read enabled, write enabled

RO : Read enabled, write disabled

State immediately after reset

b7							b0
00_{16}							
0	?	?	?	?	?	?	?
0	?	?	?	?	?	?	?
?							
0	0	0	0	?	?	?	?
0	0	0	0	0	?	0	?
00_{16}							
00_{16}							
00_{16}							
00_{16}							
00_{16}							
?							
00_{16}							
00_{16}							
0	0	0	?	0	0	0	0
00_{16}							
FF_{16}							
07_{16}							
FF_{16}							
07_{16}							
00_{16}							
00_{16}							
?							
?							
?							
0	0	0	0	0	0	0	?
00_{16}							
?	?	1	1	1	1	0	0
00_{16}							
00_{16}							
00_{16}							
00_{16}							

Access characteristics

b7							b0
		RW					
		RW					
		RW					
				RW			
					RW		RW
RW							
RW							
RW							
RW							
RW					RW		
RW							
			RO		RW		
		RW					
RW							
RW							
RW							
RW							
			RW				
		RW					
RW							
		RW					
					RW		
RW							
			RW		RW		
RW							
			RW		RW		

■2 Page Register Area (addresses 217_{16} to $21B_{16}$)

<Bit allocation>

: } Function bit

Name : } Function bit

: No function bit

0 : Fix this bit to "0" (do not write "1")

1 : Fix this bit to "1" (do not write "0")

Address	Register	Bit allocation b7							b0
217_{16}	ROM correction address 1 (high-order)	ADH17	ADH16	ADH15	ADH14	ADH13	ADH12	ADH11	ADH10
218_{16}	ROM correction address 1 (low-order)	ADL17	ADL16	ADL15	ADL14	ADL13	ADL12	ADL11	ADL10
219_{16}	ROM correction address 2 (high-order)	ADH27	ADH26	ADH25	ADH24	ADH23	ADH22	ADH21	ADH20
$21A_{16}$	ROM correction address 2 (low-order)	ADL27	ADL26	ADL25	ADL24	ADL23	ADL22	ADL21	ADL20
$21B_{16}$	ROM correction enable register (RCR)					0	0	RCR1	RCR0

Note: Only M37221M8-XXXSP and M37221MA-XXXSP have this area.

Fig. 6.6.3 Memory map of 2 page register (including internal state immediately after reset and access characteristics) (3) (only M37221M8-XXXSP and M37221MA-XXXSP)

<State immediately after reset >

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Undefined immediately after reset

RW : Read enabled, write enabled

RO : Read enabled, write disabled

State immediately after reset

b7							b0
?							
?							
?							
?							
00_{16}							

Access characteristics

b7							b0
RW							
RW							
RW							
RW							
						RW	

■SFR Area (addresses $C0_{16}$ to DF_{16})

<Bit allocation>

: Function bit

Name

: No function bit

0 : Fix this bit to "0" (do not write "1")

1 : Fix this bit to "1" (do not write "0")

Address	Register	Bit allocation b7 b0
$C0_{16}$	Port P0 (P0)	
$C1_{16}$	Port P0 direction register (D0)	
$C2_{16}$	Port P1 (P1)	
$C3_{16}$	Port P1 direction register (D1)	
$C4_{16}$	Port P2 (P2)	
$C5_{16}$	Port P2 direction register (D2)	
$C6_{16}$	Port P3 (P3)	
$C7_{16}$	Port P3 direction register (D3)	
$C8_{16}$		
$C9_{16}$		
CA_{16}	Port P5 (P5)	
CB_{16}	Port P5 direction register (D5)	
CC_{16}		
CD_{16}	Port P3 output mode control register (P3S)	DA2S DA1S P31S P30S
CE_{16}	DA-H register (DA-H)	
CF_{16}	DA-L register (DA-L)	
$D0_{16}$	PWM0 register (PWM0)	
$D1_{16}$	PWM1 register (PWM1)	
$D2_{16}$	PWM2 register (PWM2)	
$D3_{16}$	PWM3 register (PWM3)	
$D4_{16}$	PWM4 register (PWM4)	
$D5_{16}$	PWM output control register 1 (PW)	PW7 PW6 PW5 PW4 PW3 PW2 PW1 PW0
$D6_{16}$	PWM output control register 2 (PN)	PN4 PN3 PN2
$D7_{16}$		
$D8_{16}$		
$D9_{16}$		
DA_{16}		
DB_{16}		
DC_{16}	Serial I/O mode register (SM)	SM6 SM5 0 SM3 SM2 SM1 SM0
DD_{16}	Serial I/O regsiter (SIO)	
DE_{16}	DA1 conversion register (DA1)	0 DA15 DA14 DA13 DA12 DA11 DA10
DF_{16}	DA2 conversion register (DA2)	0 DA25 DA24 DA23 DA22 DA21 DA20

Fig. 6.6.4 SFR assignment (including internal state immediately after reset and access characteristics) (4) (M37220M3-XXXSP/FP)

<State immediately after reset>

- 0 : "0" immediately after reset
- 1 : "1" immediately after reset
- ? : Undefined immediately after reset
- RW : Read enabled, write enabled
- RO : Read enabled, write disabled

State immediately after reset (b7 … b0)	Access characteristics (b7 … b0)
?	RW
00_{16}	RW
?	RW
00_{16}	RW
?	RW
00_{16}	RW
0 0 0 ? ? ? ? ?	– – – RW RW RW RW RW
00_{16}	– – – – – RW RW RW
?	– – – – – – – –
?	– – – – – – – –
0 0 ? ? ? ? ? ?	– – RW RW RW RW – –
00_{16}	– – RW RW RW RW – –
?	– – – – – – – –
00_{16}	– – – – RW RW RW RW
?	RW
0 0 ? ? ? ? ? ?	– – RW RW RW RW RW RW
?	RW
?	RW
?	RW
?	RW
?	RW
00_{16}	RW
00_{16}	– – – RW RW RW – –
?	– – – – – – – –
?	– – – – – – – –
?	– – – – – – – –
?	– – – – – – – –
?	– – – – – – – –
00_{16}	– RW RW – RW RW RW RW
?	RW
0 0 ? ? ? ? ? ?	– – RW RW RW RW RW RW
0 0 ? ? ? ? ? ?	– – RW RW RW RW RW RW

■SFR Area (addresses $E0_{16}$ to FF_{16})

<Bit allocation>

Function bit

: No function bit

0 : Fix this bit to "0" (do not write "1")

1 : Fix this bit to "1" (do not write "0")

Address	Register	Bit allocation b7							b0
$E0_{16}$	Horizontal position register (HR)			HR5	HR4	HR3	HR2	HR1	HR0
$E1_{16}$	Vertical position register 1 (CV1)		CV16	CV15	CV14	CV13	CV12	CV11	CV10
$E2_{16}$	Vertical position register 2 (CV2)		CV26	CV25	CV24	CV23	CV22	CV21	CV20
$E3_{16}$									
$E4_{16}$	Character size register (CS)					CS21	CS20	CS11	CS10
$E5_{16}$	Border selection register (MD)						MD20		MD10
$E6_{16}$	Color register 0 (CO0)			CO05		CO03	CO02	CO01	
$E7_{16}$	Color register 1 (CO1)			CO15		CO13	CO12	CO11	
$E8_{16}$	Color register 2 (CO2)			CO25		CO23	CO22	CO21	
$E9_{16}$	Color register 3 (CO3)			CO35		CO33	CO32	CO31	
EA_{16}	CRT control register (CC)						CC2	CC1	CC0
EB_{16}									
EC_{16}	CRT port control register (CRTP)	OP7	OP6	OP5	OUT		R/G/B	VSYC	HSYC
ED_{16}	CRT clock selection register (CK)	0	0	0	0	0	0	CK1	CK0
EE_{16}	A-D control register 1 (AD1)				ADM4		ADM2	ADM1	ADM0
EF_{16}	A-D control register 2 (AD2)			ADC5	ADC4	ADC3	ADC2	ADC1	ADC0
$F0_{16}$	Timer 1 (TM1)								
$F1_{16}$	Timer 2 (TM2)								
$F2_{16}$	Timer 3 (TM3)								
$F3_{16}$	Timer 4 (TM4)								
$F4_{16}$	Timer 12 mode register (T12M)			0	T12M4	T12M3	T12M2	T12M1	T12M0
$F5_{16}$	Timer 34 mode register (T34M)			T34M5	T34M4	T34M3	T34M2	T34M1	T34M0
$F6_{16}$	PWM5 register (PWM5)								
$F7_{16}$									
$F8_{16}$									
$F9_{16}$	Interrupt input polarity register (RE)	0		RE5	RE4	RE3	0	0	
FA_{16}	Test register (TEST)	00_{16}							
FB_{16}	CPU mode register (CPUM)	1	1	1	1	1	CM2	0	0
FC_{16}	Interrupt request register 1 (IREQ1)	IT3R		VSCR	CRTR	TM4R	TM3R	TM2R	TM1R
FD_{16}	Interrupt request register 2 (IREQ2)	0			MSR		S1R	1T2R	1T1R
FE_{16}	Interrupt control register 1 (ICON1)	IT3E		VSCE	CRTE	TM4E	TM3E	TM2E	TM1E
FF_{16}	Interrupt control register 2 (ICON2)	0	0	0	MSE	0	S1E	1T2E	1T1E

Fig. 6.6.5 SFR assignment (including internal state immediately after reset and access characteristics) (5) (M37220M3-XXXSP/FP)

<State immediately after reset>

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Undefined immediately after reset

RW : Read enabled, write enabled

RO : Read enabled, write disabled

State immediately after reset

b7							b0
00_{16}							
0	?	?	?	?	?	?	?
0	?	?	?	?	?	?	?
?							
0	0	0	0	?	?	?	?
0	0	0	0	0	?	0	?
00_{16}							
00_{16}							
00_{16}							
00_{16}							
00_{16}							
?							
00_{16}							
00_{16}							
0	0	0	?	0	0	0	0
00_{16}							
FF_{16}							
07_{16}							
FF_{16}							
07_{16}							
00_{16}							
00_{16}							
?							
?							
?							
0	0	0	0	0	0	0	?
00_{16}							
1	1	1	1	1	1	0	0
00_{16}							
00_{16}							
00_{16}							
00_{16}							

Access characteristics

b7							b0
		RW					
	RW						
	RW						
				RW			
					RW		RW
		RW		RW			
		RW		RW			
		RW		RW			
		RW		RW			
RW					RW		
RW					RW		
						RW	
			RO		RW		
		RW					
RW							
RW							
RW							
RW							
			RW				
		RW					
RW							
		RW					
					RW		
RW		RW					
			RW		RW		
RW		RW					
			RW		RW		

6.7 Control registers

Port Pi Direction Register

b7 b6 b5 b4 b3 b2 b1 b0

Port Pi direction register (Di) (i=0,1,2) [Addresses 00C1$_{16}$, 00C3$_{16}$, 00C5$_{16}$]

B	Name	Functions	After reset	R	W
0	Port Pi direction register	0 : Port Pi$_0$ input mode 1 : Port Pi$_0$ output mode	0	R	W
1		0 : Port Pi$_1$ input mode 1 : Port Pi$_1$ output mode	0	R	W
2		0 : Port Pi$_2$ input mode 1 : Port Pi$_2$ output mode	0	R	W
3		0 : Port Pi$_3$ input mode 1 : Port Pi$_3$ output mode	0	R	W
4		0 : Port Pi$_4$ input mode 1 : Port Pi$_4$ output mode	0	R	W
5		0 : Port Pi$_5$ input mode 1 : Port Pi$_5$ output mode	0	R	W
6		0 : Port Pi$_6$ input mode 1 : Port Pi$_6$ output mode	0	R	W
7		0 : Port Pi$_7$ input mode 1 : Port Pi$_7$ output mode	0	R	W

Fig. 6.7.1 Port Pi direction register

Addresses 00C1$_{16}$, 00C3$_{16}$, 00C5$_{16}$

Port P3 Direction Register

b7 b6 b5 b4 b3 b2 b1 b0

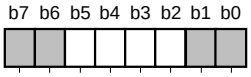
Port P3 direction register (D3) [Address 00C7$_{16}$]

B	Name	Functions	After reset	R	W
0	Port P3 direction register	0 : Port P3$_0$ input mode 1 : Port P3$_0$ output mode	0	R	W
1		0 : Port P3$_1$ input mode 1 : Port P3$_1$ output mode	0	R	W
2		0 : Port P3$_2$ input mode 1 : Port P3$_2$ output mode	0	R	W
3 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are “0.”		0	R	—

Fig. 6.7.2 Port P3 direction register

Address 00C7$_{16}$

Port P5 Direction Register



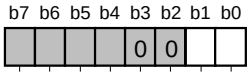
Port P5 direction register (D5) [Address $00CB_{16}$]

B	Name	Functions	After reset	R	W
0, 1	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—
2 to 5	Port P5 direction register	0 : CRT output (R) 1 : Output port $P5_2$	0	R	W
		0 : CRT output (G) 1 : Output port $P5_3$	0	R	W
		0 : CRT output (B) 1 : Output port $P5_4$	0	R	W
		0 : CRT output (OUT1) 1 : Output port $P5_5$	0	R	W
6, 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 6.7.3 Port P5 direction register

Address $00CB_{16}$

Port P3 output mode control register



Port P3 output mode control register (P3S) [address $00CD_{16}$]

B	Name	Functions	After reset	R	W
0	$P3_0$ output structure selection bit (P30S)	0 : CMOS output 1 : N-channel open-drain output	0	R	W
1	$P3_1$ output structure selection bit (P31S)	0 : CMOS output 1 : N-channel open-drain output	0	R	W
2, 3	Fix thes bits to "0."		0	R	W
4 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

(See note)

Note: M37220M3-XXXSP/FP

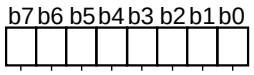
2	DA1 output enable bit	0 : $P3_0$ input/output 1 : DA1 output	0	R	W
3	DA2 output enable bit	0 : $P3_1$ input/output 1 : DA2 output	0	R	W

Fig. 6.7.4 Port P3 output mode control register

Address $00CD_{16}$

6.7 Control registers

PWM Output Control Register 1



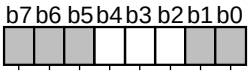
PWM output control register 1 (PW) [Address 00D5$_{16}$]

B	Name	Functions	After reset	R	W
0	DA, PWM count source selection bit (PW0)	0 : Count source supply 1 : Count source stop	0	R	W
1	DA/PN4 output selection bit (PW1)	0 : DA output 1 : PN4 output	0	R	W
2	P0$_0$/PWM0 output selection bit (PW2)	0: P0$_0$ output 1: PWM0 output	0	R	W
3	P0$_1$/PWM1 output selection bit (PW3)	0: P0$_1$ output 1: PWM1 output	0	R	W
4	P0$_2$/PWM2 output selection bit (PW4)	0: P0$_2$ output 1: PWM2 output	0	R	W
5	P0$_3$/PWM3 output selection bit (PW5)	0: P0$_3$ output 1: PWM3 output	0	R	W
6	P0$_4$/PWM4 output selection bit (PW6)	0: P0$_4$ output 1: PWM4 output	0	R	W
7	P0$_5$/PWM5 output selection bit (PW7)	0: P0$_5$ output 1: PWM5 output	0	R	W

Fig. 6.7.5 PWM output control register 1

Address 00D5$_{16}$

PWM Output Control Register 2



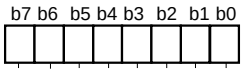
PWM output control register 2 (PN) [Address 00D6$_{16}$]

B	Name	Functions	After reset	R	W
0, 1	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are “0.”		0	R	—
2	DA output polarity selection bit (PN3)	0 : Positive polarity 1 : Negative polarity	0	R	W
3	PWM output polarity selection bit (PN4)	0 : Positive polarity 1 : Negative polarity	0	R	W
4	DA general-purpose output bit (PN5)	0 : Output LOW 1 : Output HIGH	0	R	W
5 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are “0.”		0	R	—

Fig. 6.7.6 PWM output control register 2

Address 00D6$_{16}$

I^2C Data Shift Register



I^2C data shift register (S0) [Address $00D7_{16}$]

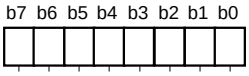
B	Name	Functions	After reset	R	W
0 to 7	D0 to D7	This is an 8-bit shift register to store receive data and write transmit data.	Indeterminate	R	W

Note: To write data into the I^2C data shift register after setting the MST bit to "0" (slave mode), keep an interval of 8 machine cycles or more.

Fig. 6.7.7 I^2C data shift register

Address $00D7_{16}$

I^2C Address Register



I^2C address register (S0D) [Address $00D8_{16}$]

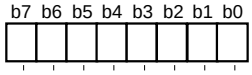
B	Name	Functions	After reset	R	W
0	$\overline{\text{Read}}$/write bit (RBW)	0: Read 1: Write	0	R	W
1 to 7	Slave address (SAD0 to SAD6)	The address data transmitted from the master is compared with the contents of these bits.	0	R	W

Fig. 6.7.8 I^2C address register

Address $00D8_{16}$

6.7 Control registers

I²C Status Register



I²C status register (S1) [Address 00D9₁₆]

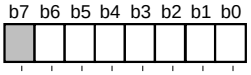
B	Name	Functions	After reset	R	W
0	Last receive bit (LRB) (See note)	0 : Last bit = "0 " 1 : Last bit = "1 "	Indeterminate	R	—
1	General call detecting flag (AD0) (See note)	0 : No general call detected 1 : General call detected	0	R	—
2	Slave address comparison flag (AAS) (See note)	0 : Address mismatch 1 : Address match	0	R	—
3	Arbitration lost detecting flag (AL) (See note)	0 : Not detected 1 : Detected	0	R	—
4	I²C-BUS interface interrupt request bit (PIN)	0 : Interrupt request issued 1 : No interrupt request issued	1	R	W
5	Bus busy flag (BB)	0 : Bus free 1 : Bus busy	0	R	W
6, 7	Communication mode specification bits (TRX, MST)	b7 b6 0 0 : Slave recieve mode 0 1 : Slave transmit mode 1 0 : Master recieve mode 1 1 : Master transmit mode	0	R	W

Note : These bits and flags can be read out, but cannnot be written.

Fig. 6.7.9 I²C status register

Address 00D9₁₆

I²C Control Register



I²C control register (S1D : address 00DA16)

B	Name	Functions	After reset	R	W
0 to 2	Bit counter (Number of transmit/recieve bits) (BC0 to BC2)	b2 b1 b0 0 0 0 : 8 0 0 1 : 7 0 1 0 : 6 0 1 1 : 5 1 0 0 : 4 1 0 1 : 3 1 1 0 : 2 1 1 1 : 1	0	R	W
3	I²C-BUS interface use enable bit (ESO)	0 : Disabled 1 : Enabled	0	R	W
4	Data format selection bit (ALS)	0 : Addressing mode 1 : Free data format	0	R	W
5	Addressing format selection bit (10BIT SAD)	0 : 7-bit addressing format 1 : 10-bit addressing format	0	R	W
6, 7	Connection control bits between I²C-BUS interface and ports (BSEL0, BSEL1)	b7 b6 Connection port 0 0 : None 0 1 : SCL1, SDA1 1 0 : SCL2, SDA2 1 1 : SCL1, SDA1 SCL2, SDA2	0	R	W

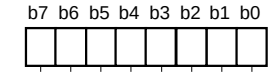
Note: When using ports P11-P14 as I²C-BUS interface, the output structure changes automatically from CMOS output to N-channel open-drain output. However, set the port direction register to "1" (output mode).

Fig. 6.7.10 I²C control register

Address 00DA16

6.7 Control registers

I^2C Clock Control Register



I^2C clock control register (S2 : address $00DB_{16}$)

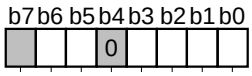
B	Name	Functions			After reset	R	W
0 to 4	SCL frequency control bits (CCR0 to CCR4)	Setup value of CCR4–CCR0	Standard clock mode	High speed clock mode	0	R	W
		00 to 02	Setup disabled	Setup disabled			
		03	Setup disabled	333			
		04	Setup disabled	250			
		05	100	400 (See note)			
		06	83.3	166			
		⋮	500/CCR value	1000/CCR value			
		1D	17.2	34.5			
		1E	16.6	33.3			
		1F	16.1	32.3			
		(at ϕ = 4 MHz, unit : kHz)					
5	SCL mode specification bit (FAST MODE)	0 : Standard clock mode 1 : High-speed clock mode			0	R	W
6	ACK bit (ACK BIT)	0 : ACK is returned. 1 : ACK is not returned.			0	R	W
7	ACK clock bit (ACK)	0 : No ACK clock 1 : ACK clock			0	R	W

Note: At 4000kHz in the high-speed clock mode, the duty is as below .
"0" period : "1" period = 3 : 2
In the other cases, the duty is as below.
"0" period : "1" period = 1 : 1

Fig. 6.7.11 I^2C clock contorol register

Address $00DB_{16}$

Serial I/O Mode Register



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

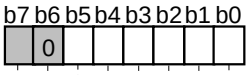
B	Name	Functions	After reset	R	W
0, 1	Internal synchronous clock selection bits (SM0, SM1)	b1 b0 0 0: f(XIN)/4 0 1: f(XIN)/16 1 0: f(XIN)/32 1 1: f(XIN)/64	0	R	W
2	Synchronous clock selection bit (SM2)	0: External clock 1: Internal clock	0	R	W
3	Serial I/O port selection bit (SM3)	0: P20, P21 functions as port 1: SCLK, SOUT	0	R	W
4	Fix this bit to “0.”		0	R	W
5	Transfer direction selection bit (SM5)	0: LSB first 1: MSB first	0	R	W
6	Serial input pin selection bit (SM6)	0: Input signal from SIN pin 1: Input signal from SOUT pin	0	R	W
7	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is “0.”		0	R	—

Fig. 6.7.12 Serial I/O mode register

Address 00DC16

6.7 Control registers

DA n Conversion Register



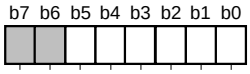
DA n conversion register (DAn) (n = 1 and 2) [Address 00DE16, 00DF16]

B	Name	Functions	After reset	R	W
0 to 5	DA conversion set bits (DAn0–DAn5)	b5 b4 b3 b2 b1 b0 0 0 0 0 0 0 : 0/64Vcc 0 0 0 0 0 1 : 1/64Vcc 0 0 0 0 1 0 : 2/64Vcc ⋮ 1 1 1 1 0 1 : 61/64Vcc 1 1 1 1 1 0 : 62/64Vcc 1 1 1 1 1 1 : 63/64Vcc	Indeterminate	R	W
6	Fix this bit to “0.”		0	R	—
7	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is “ 0.”		0	R	—

Fig. 6.7.13 DA n conversion register (only M37220M3-XXXSP/FP)

Addresses 00DE16, 00DF16

Horizontal Position Register



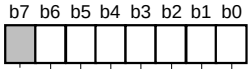
Horizontal position register (HR) [Address 00E016]

B	Name	Functions	After reset	R	W
0 to 5	Horizontal display start positions (HR0 to HR5)	64 steps (0016 to 3F16)	0	R	W
6, 7	Nothing is assigned. These bits are write disable bits. When thses bits are read out, the values are “0.”		0	R	—

Fig. 6.7.14 Horizontal position register

Address 00E016

Vertical Position Register n



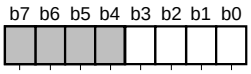
Vertical position register n (CV1,CV2) (n = 1 and 2) [Addresses 00E1$_{16}$, 00E2$_{16}$]

B	Name	Functions	After reset	R	W
0 to 6	Vertical display start positions (CV1 : CV10 to CV16) (CV2 : CV20 to CV26)	128 steps (00$_{16}$ to 7F$_{16}$)	Indeterminate	R	W
7	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—

Fig. 6.7.15 Vertical position register n

Addresses 00E1$_{16}$, 00E2$_{16}$

Character Size Register



Character size register (CS) [Address 00E4$_{16}$]

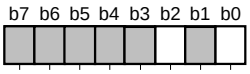
B	Name	Functions	After reset	R	W
0, 1	Character size of block 1 selection bits (CS10, CS11)	00 : Minimum size 01 : Medium size 10 : Large size 11 : Do not set.	Indeterminate	R	W
2,3	Character size of block 2 selection bits (CS20,CS21)	00 : Minimum size 01 : Medium size 10 : Large size 11 : Do not set.	Indeterminate	R	W
4 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 6.7.16 Character size register

Address 00E4$_{16}$

6.7 Control registers

Border Selection Register



Border selection register (MD) [Address $00E5_{16}$]

B	Name	Functions	After reset	R	W	
0	Block 1 OUT1 output border selection bit (MD10)	0 : Same output as character output 1 : Border output	Indeterminate	R	W	(See note)
1	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—	
2	Block 1 OUT1 output border selection bit (MD20)	0 : Same output as character output 1 : Border output	Indeterminate	R	W	(See note)
3 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—	

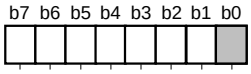
Note : M37220M3-XXXSP/FP

0	Block 1 OUT output border selection bit (MD10)	0 : Same output as character output 1 : Border output	Indeterminate	R	W
2	Block 2 OUT output border selection bit (MD20)	0 : Same output as character output 1 : Border output	Indeterminate	R	W

Fig. 6.7.17 Border selection register

Address $00E5_{16}$

Color Register n



Color register n (CO0 to CO3) (n = 0 to 3) [Addresses 00E6$_{16}$ to 00E9$_{16}$]

B	Name	Functions	After reset	R	W
0	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
1	B signal output selection bit (COn1)	0 : No character is output 1 : Character is output	0	R	W
2	G signal output selection bit (COn2)	0 : No character is output 1 : Character is output	0	R	W
3	R signal output selection bit (COn3)	0 : No character is output 1 : Character is output	0	R	W
4	B signal output (background) selection bit (COn4)	0 : No background color is output 1 : Background color is output (See note 1)	0	R	W
5	OUT1 signal output control bit (COn5)	0 : Character is output 1 : Blank is output (See notes 1, 2)	0	R	W
6	G signal output (background) selection bit (COn6)	0 : No background color is output 1 : Background color is output	0	R	W
7	R signal output (background) selection bit (COn7)	0 : No background color is output 1 : Background color is output (See note 2)	0	R	W

(See note 3)

Notes 1: When bit 5 = "0" and bit 4 = "1," there is output same as a character or border output from the OUT1 pin.
Do not set bit 5 = "0" and bit 4 = "0."
2: When only bit 7 = "1" and bit 5 = "0," there is output from the OUT2 pin.

3: M37220M3-XXXSP/FP

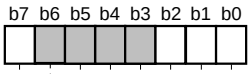
4	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
5	OUT signal output control bit (COn5)	0 : Character is output 1 : Blank is output	0	R	W
6, 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—

Fig. 6.7.18 Color register n

Addresses 00E6$_{16}$ to 00E9$_{16}$

6.7 Control registers

CRT Control Register



CRT control register (CC) [Address 00EA16]

B	Name	Functions	After reset	R	W
0	All-blocks display control bit (Note 1) (CC0)	0 : All-blocks display off 1 : All-blocks display on	0	R	W
1	Block 1 display control bit (CC1)	0 : Block 1 display off 1 : Block 1 display on	0	R	W
2	Block 2 display control bit (CC2)	0 : Block 2 display off 1 : Block 2 display on	0	R	W
3 to 6	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are “0.”		0	R	—
7	P10/OUT2 pin switch bit (CC7)	0 : P10 1 : OUT2	0	R	W

(See note 2)

Notes 1: Display is controlled by logical product (AND) between the all-blocks display control bit and each block control bit.

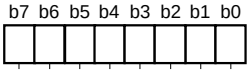
2 : M37220M3-XXXSP/FP

7	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is “0.”	0	R	—

Fig. 6.7.19 CRT control register

Address 00EA16

CRT Port Control Register



CRT port control register (CRTP) [Address $00EC_{16}$]

B	Name	Functions	After reset	R	W	
0	H_{SYNC} input polarity switch bit (HSYC)	0 : Positive polarity 1 : Negative polarity	0	R	W	
1	V_{SYNC} input polarity switch bit (VSYC)	0 : Positive polarity 1 : Negative polarity	0	R	W	
2	R, G, B output polarity switch bit (R/G/B)	0 : Positive polarity 1 : Negative polarity	0	R	W	
3	OUT2 output polarity switch bit (OUT2)	0 : Positive polarity 1 : Negative polarity	0	R	W	(See note)
4	OUT1 output polarity switch bit (OUT1)	0 : Positive polarity 1 : Negative polarity	0	R	W	
5	R signal output switch bit (OP5)	0 : R signal output 1 : MUTE signal output	0	R	W	
6	G signal output switch bit (OP6)	0 : G signal output 1 : MUTE signal output	0	R	W	
7	B signal output switch bit (OP7)	0 : B signal output 1 : MUTE signal output	0	R	W	

Note : M37220M3-XXXSP/FP

3	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is “0.”		0	R	—
4	OUT output polarity switch bit (OUT)	0 : Positive polarity 1 : Negative polarity	0	R	W

Fig. 6.7.20 CRT port control register

Address $00EC_{16}$

CRT Clock Selection Register

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0	0	0		

CRT clock selection register (CK) [Address $00ED_{16}$]

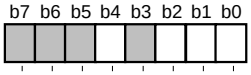
B	Name	Functions			After reset	R	W
0, 1	CRT clock selection bits (CK0,CK1)	b1	b0	Functions	0	R	W
		1	0	The clock for display is supplied by connecting RC or LC across the pins OSC1 and OSC2.			
		0	1	Since the main clock is used as the clock for display, the oscillation frequency is limited. Because of this, the character size in width (horizontal) direction is also limited. In this case, pins OSC1 and OSC2 are also used as input ports $P3_3$ and $P3_4$ respectively. CRT oscillation frequency = $f(X_{IN})$			
		1	0	CRT oscillation frequency = $f(X_{IN})/1.5$			
		1	1	The clock for display is supplied by connecting the following across the pins OSC1 and OSC2. • a ceramic resonator only for CRT display and a feedback resistor • a quartz-crystal oscillator only for CRT display and a feedback resistor (Note)			
2 to 7	Fix these bits to "0."				0	R	W

Note: It is necessary to connect other ceramic resonator or quartz-crystal oscillator across the pins X_{IN} and X_{OUT}.

Fig. 6.7.21 CRT clock selection register

Address $00ED_{16}$

A-D Control Register 1



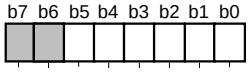
A-D control register 1 (AD1) [Address 00EE 16]

B	Name	Functions	After reset	R	W
0 to 2	Analog input pin selection bits (ADM0, ADM1, ADM2)	b2 b1 b0 0 0 0 : A-D1 0 0 1 : A-D2 0 1 0 : A-D3 0 1 1 : A-D4 1 0 0 : A-D5 1 0 1 : A-D6 1 1 0 : } Do not set. 1 1 1 : }	0	R	W
3	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is “0.”		0	R	—
4	Storage bit of comparison result (ADM4)	0: Input voltage < reference voltage 1: Input voltage > reference voltage	Indeterminate	R	W
5 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are “0.”		0	R	—

Fig. 6.7.22 A-D control register 1

Address 00EE16

A-D Control Register 2



A-D control register 2(AD2) [Address 00EF 16]

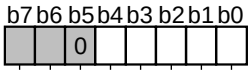
B	Name	Functions	After reset	R	W
0 to 5	D-A converter set bits (ADC0, ADC1, ADC2, ADC3, ADC4, ADC5)	b5 b4 b3 b2 b1 b0 0 0 0 0 0 0 : 1/128Vcc 0 0 0 0 0 1 : 3/128Vcc 0 0 0 0 1 0 : 5/128Vcc 1 1 1 1 0 1 : 123/128Vcc 1 1 1 1 1 0 : 125/128Vcc 1 1 1 1 1 1 : 127/128Vcc	0	R	W
6, 7	Nothing is assigned. These bits are write disable bits. When these bits are reed out, the values are “ 0.”		0	R	—

Fig. 6.7.23 A-D control register 2

Address 00EF16

6.7 Control registers

Timer 12 Mode Register



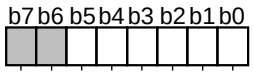
Timer 12 mode register (T12M) [Address 00F4$_{16}$]

B	Name	Functions	After reset	R	W
0	Timer 1 count source selection bit (T12M0)	0: f(XIN)/16 1: f(XIN)/4096	0	R	W
1	Timer 2 count source selection bit (T12M1)	0: Internal clock 1: External clock from P24/TIM2 pin	0	R	W
2	Timer 1 count stop bit (T12M2)	0: Count start 1: Count stop	0	R	W
3	Timer 2 count stop bit (T12M3)	0: Count start 1: Count stop	0	R	W
4	Timer 2 internal count source selection bit (T12M4)	0: f(XIN)/16 1: Timer 1 overflow	0	R	W
5	Fix this bit to “0.”		0	R	W
6,7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are “0.”		0	R	—

Fig. 6.7.24 Timer 12 mode register

Address 00F4$_{16}$

Timer 34 Mode Register



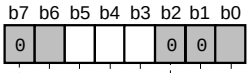
Timer 34 mode register (T34M) [Address 00F5$_{16}$]

B	Name	Functions	After reset	R	W
0	Timer 3 count source selection bit (T34M0)	0: f(XIN)/16 1: External clock	0	R	W
1	Timer 4 internal count source selection bit (T34M1)	0: Timer 3 overflow 1: f(XIN)/16	0	R	W
2	Timer 3 count stop bit (T34M2)	0: Count start 1: Count stop	0	R	W
3	Timer 4 count stop bit (T34M3)	0: Count start 1: Count stop	0	R	W
4	Timer 4 count source selection bit (T34M4)	0: Internal clock 1: f(XIN)/2	0	R	W
5	Timer 3 external count source selection bit (T34M5)	0: External clock from P23/TIM3 pin 1: External clock from HSYNC pin	0	R	W
6,7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are “0.”		0	R	—

Fig. 6.7.25 Timer 34 mode register

Address 00F5$_{16}$

Interrupt Input Polarity Register



Interrupt input polarity register(RE) [Address 00F9$_{16}$]

b	Name	Functions	After reset	R	W
0	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is “0.”		Indeterminate	R	—
1, 2	Fix these bits to “0.”		0	R	W
3	INT1 polarity switch bit (RE3)	0 : Positive polarity 1 : Negative polarity	0	R	W
4	INT2 polarity switch bit (RE4)	0 : Positive polarity 1 : Negative polarity	0	R	W
5	INT3 polarity switch bit (RE5)	0 : Positive polarity 1 : Negative polarity	0	R	W
6	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is “0.”		0	R	—
7	Fix this bit to “0.”		0	R	W

Fig. 6.7.26 Interrupt input polarity register

Address 00F9$_{16}$

6.7 Control registers

CPU Mode Register

b7	b6	b5	b4	b3	b2	b1	b0
1	1	1	1	1		0	0

CPU mode register (CPUM) (CM) [Address 00FB16]

B	Name	Functions	After reset	R	W
0, 1	Fix these bits to "0."		0	R	W
2	Stack page selection bit (CM2)	0: 0 page (Note) 1: 1 page	1	R	W
3 to 5	Fix these bits to "1."		1	R	W
6, 7			Indeterminate	R	W

Note: This bit is set to "1" after reset release.

Fig. 6.7.27 CPU mode register

Address 00FB16

Interrupt Request Register 1

b7	b6	b5	b4	b3	b2	b1	b0

Interrupt request register 1 (IREQ1) [Address 00FC16]

B	Name	Functions	After reset	R	W
0	Timer 1 interrupt request bit (TM1R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
1	Timer 2 interrupt request bit (TM2R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
2	Timer 3 interrupt request bit (TM3R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
3	Timer 4 interrupt request bit (TM4R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
4	CRT interrupt request bit (CRTR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
5	VSYNC interrupt request bit (VSCR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
6	Multi-master I2C-BUS interface interrupt request bit (IICR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻ (See note)
7	INT3 interrupt request bit (IT3R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻

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

Note : M37220M3-XXXSP/FP

6	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."	0	R	—

Fig. 6.7.28 Interrupt request register 1

Address 00FC16

Interrupt Request Register 2

b7 b6 b5 b4 b3 b2 b1 b0
0

Interrupt request register 2 (IREQ2) [Address 00FD16]

B	Name	Functions	After reset	R	W
0	INT1 interrupt request bit (ITIR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
1	INT2 interrupt request bit (IT2R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
2	Serial I/O interrupt request bit (S1R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
3	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."		0	R	—
4	f(XIN)/4096 interrupt request bit (MSR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R	✻
5, 6	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—
7	Fix this bit to "0."		0	R	W

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

Fig. 6.7.29 Interrupt request register 2

Address 00FD16

Interrupt Control Register 1

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt control register 1 (ICON1) [Address 00FE16]

B	Name	Functions	After reset	R	W	
0	Timer 1 interrupt enable bit (TM1E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W	
1	Timer 2 interrupt enable bit (TM2E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W	
2	Timer 3 interrupt enable bit (TM3E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W	
3	Timer 4 interrupt enable bit (TM4E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W	
4	CRT interrupt enable bit (CRTE)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W	
5	VSYNC interrupt enable bit (VSCE)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W	
6	Multi-master I2C-BUS interface interrupt enable bit (IICE)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W	} (See note)
7	INT3 interrupt enable bit (IT3E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W	

Note : M37220M3-XXXSP/FP

6	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is "0."	0	R	—

Fig. 6.7.30 Interrupt control register 1

Address 00FE16

6.7 Control registers

Interrupt Control Register 2

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0		0			

Interrupt control register 2 (ICON2) [Address 00FF16]

B	Name	Functions	After reset	R	W
0	INT1 interrupt enable bit (IT1E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
1	INT2 interrupt enable bit (IT2E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
2	Serial I/O interrupt enable bit (S1E)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
3	Fix this bit to “0.”		0	R	W
4	f(XIN)/4096 interrupt enable bit (MSE)	0 : Interrupt disabled 1 : Interrupt enabled	0	R	W
5 to 7	Fix these bits to “0.”		0	R	W

Fig. 6.7.31 Interrupt control register 2

Address 00FF16

ROM correction enable register

b7	b6	b5	b4	b3	b2	b1	b0
				0	0		

ROM correction enable register (RCR) [Address 021216]

B	Name	Functions	After reset	R	W
0	Block 1 enable bit (RCR0)	0 : Disabled 1 : Enabled	0	○	○
1	Block 2 enable bit (RCR1)	0 : Disabled 1 : Enabled	0	○	○
2, 3	Fix these bits to “0.”		0	○	×
4 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are “0.”		0	○	×

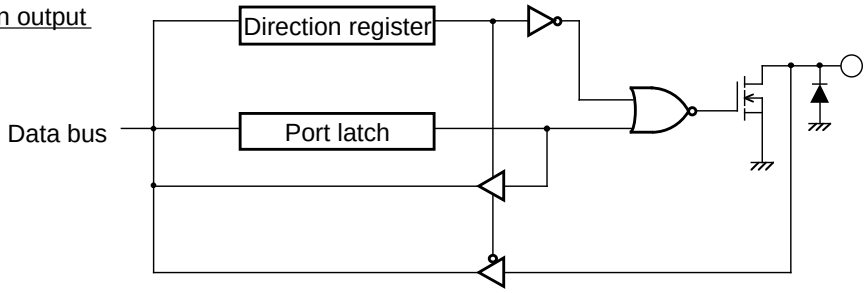
Fig. 6.7.32 ROM correction enable register

Address 021216

6.8 Ports

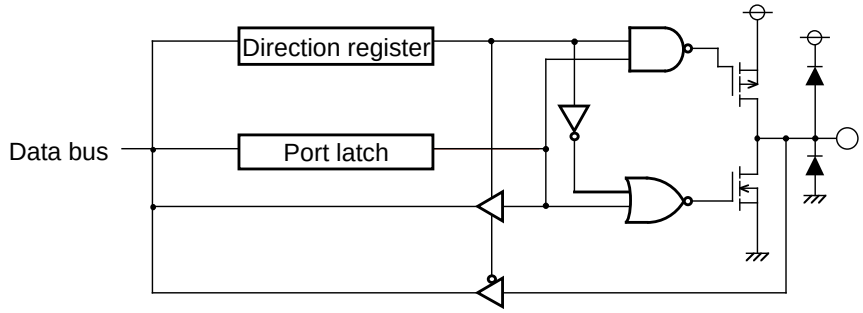
P0$_0$/PWM0–P0$_5$/PWM5, P3$_2$

N-channel open-drain output



●M37221M4-XXXSP, M37221M6-XXXSP/FP, M37221M8-XXXSP, M37221MA-XXXSP
P1$_0$/OUT2, P1$_1$/SCL1, P1$_2$/SCL2, P1$_3$/SDA1, P1$_4$/SDA2, P1$_5$/A-D1/INT3, P1$_6$/A-D2, P1$_7$/A-D3, P2$_0$/SCLK, P2$_1$/SOUT, P2$_2$/SIN, P2$_3$/TIM3, P2$_4$/TIM2, P2$_5$–P2$_7$, P3$_0$/A-D5, P3$_1$/A-D6 (See notes 1, 2)
●M37220M3-XXXSP/FP
P1$_0$–P1$_4$, P1$_5$/A-D/INT3, P1$_6$/A-D2, P1$_7$/A-D3, P2$_0$/SCLK, P2$_1$/SOUT, P2$_2$/SIN, P2$_3$/TIM3, P2$_4$/TIM2, P2$_5$–P2$_7$, P3$_0$/A-D5/DA1, P3$_1$/A-D6/DA2 (See note 2)

CMOS output

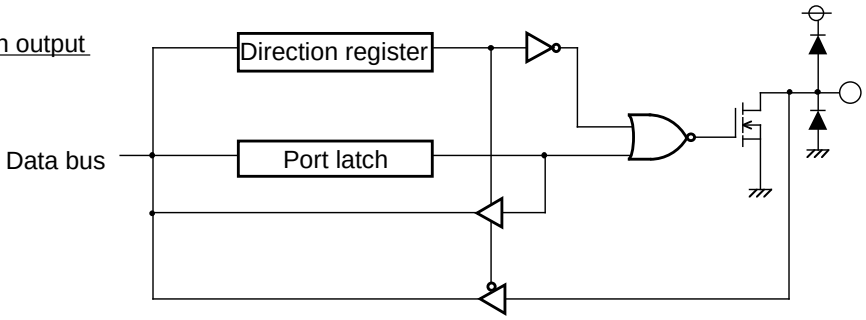


Notes 1 : When ports P1$_1$–P1$_4$ are used as multi-master I^{2}C-BUS interface pin and when ports P2$_0$, P2$_1$ are used as serial I/O output pins, their output structure is N-channel open-drain output.

2 : For the output structure of ports P3$_0$, P3$_1$, either CMOS output or N-channel open-drain output is selected (In the case of N-channel open-drain output, the block diagram is the same as below).

P0$_6$/INT2/A-D4, P0$_7$/INT1

N-channel open-drain output

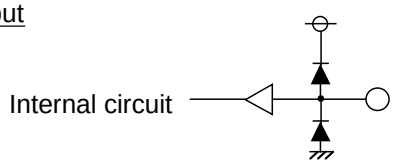


○ indicates a pin.

Fig. 6.8.1 I/O pin block diagram (1)

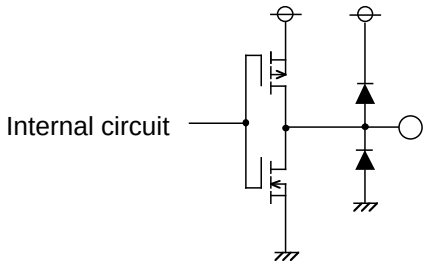
P3$_3$/OSC1, P3$_4$

Input



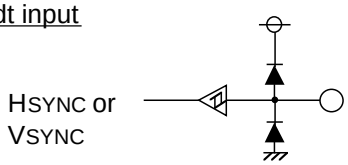
●M37221M4-XXXSP, M37221M6-XXXSP/FP, M37221M8-XXXSP, M37221MA-XXXSP
D-A, P5$_2$/R, P5$_3$/G, P5$_4$/B, P5$_5$/OUT1
●M37220M3-XXXSP/FP
D-A, P5$_2$/R, P5$_3$/G, P5$_4$/B, P5$_5$/OUT

CMOS output



HSYNC, VSYNC

Schmidt input



○ indicates a pin.

Fig. 6.8.2 I/O pin block diagram (2)

6.9 Machine instruction table

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	#
ADC (Note 1) (Note 6)	When T=0 A←A+M+C When T=1 M(X)←M(X)+M+C	Adds the carry, accumulator and memory contents. The results are entered into the accumulator. Adds the contents of the memory in the address indicated by index register X, the contents of the memory specified by the addressing mode and the carry. The results are entered into the memory at the address indicated by index register X.				69	2	2							65	3	2			
AND (Note 1)	When T=0 A←A∧M When T=1 M(X)←M(X)∧M	"AND's" the accumulator and memory contents. The results are entered into the accumulator. "AND's" the contents of the memory of the address indicated by index register X and the contents of the memory specified by the addressing mode. The results are entered into the memory at the address indicated by index register X.				29	2	2							25	3	2			
ASL	7 0 C ← □ ←0	Shifts the contents of accumulator or contents of memory one bit to the left. The low order bit of the accumulator or memory is cleared and the high order bit is shifted into the carry flag.							0A	2	1				06	5	2			
BBC (Note 4)	A_b or M_b=0?	Branches when the contents of the bit specified in the accumulator or memory is "0".										13+20i	4	2				17+20i	5	3
BBS (Note 4)	A_b or M_b=1?	Branches when the contents of the bit specified in the accumulator or memory is "1".										03+20i	4	2				07+20i	5	3
BCC (Note 4)	C=0?	Branches when the contents of carry flag is "0".																		
BCS (Note 4)	C=1?	Branches when the contents of carry flag is "1".																		
BEQ (Note 4)	Z=1?	Branches when the contents of zero flag is "1".																		
BIT	A∧M	"AND's" the contents of accumulator and memory. The results are not entered anywhere.													24	3	2			
BMI (Note 4)	N=1?	Branches when the contents of negative flag is "1".																		
BNE (Note 4)	Z=0?	Branches when the contents of zero flag is "0".																		
BPL (Note 4)	N=0?	Branches when the contents of negative flag is "0".																		
BRA	PC←PC±offset	Jumps to address specified by adding offset to the program counter.																		
BRK	B←1 $M(S)←PC_H$ S←S−1 $M(S)←PC_L$ S←S−1 M(S)←PS S←S−1 $PC_L←AD_L$ $PC_H←AD_H$	Executes a software interrupt.	00	7	1															

6.9 Machine instruction table

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
0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	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				•	•	•	•	•	•	•	•
																											10	2	2				•	•	•	•	•	•	•	•
																											80	4	2				•	•	•	•	•	•	•	•
																																	•	•	•	1	•	1	•	•

Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT,A			ZP			BIT,ZP		
			0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#
BVC (Note 4)	V=0?	Branches when the contents of overflow flag is "0."																		
BVS (Note 4)	V=1?	Branches when the contents of overflow flag is "1."																		
CLB	A_b or M_b←0	Clears the contents of the bit specified in the accumulator or memory to "0."										1B+20i	2	1				1F+20i	5	2
CLC	C←0	Clears the contents of the carry flag to "0."	18	2	1															
CLD	D←0	Clears the contents of decimal mode flag to "0."	D8	2	1															
CLI	I←0	Clears the contents of interrupt disable flag to "0."	58	2	1															
CLT	T←0	Clears the contents of index X mode flag to "0."	12	2	1															
CLV	V←0	Clears the contents overflow flag to "0."	B8	2	1															
CMP (Note 3)	When T=0 A−M When T=1 M(X)−M	Compares the contents of accumulator and memory. Compares the contents of the memory specified by the addressing mode with the contents of the address indicated by index register X.				C9	2	2							C5	3	2			
COM	M←$\overline{M}$	Forms a one's complement of the contents of memory, and stores it into memory.													44	5	2			
CPX	X−M	Compares the contents of index register X and memory.				E0	2	2							E4	3	2			
CPY	Y−M	Compares the contents of index register Y and memory.				C0	2	2							C4	3	2			
DEC	A←A−1 or M←M−1	Decrements the contents of the accumulator or memory by 1.							1A	2	1				C6	5	2			
DEX	X←X−1	Decrements the contents of index register X by 1.	CA	2	1															
DEY	Y←Y−1	Decrements the contents of index register Y by 1.	88	2	1															
DIV (Note 5)	A←(M(zz+X+1), M(zz+X))/A M(S) ← 1's complement of Remainder S←S−1	Divides the 16-bit data that is the contents of M(zz+x+1) for high byte and the contents of M(zz+x) for low byte by the accumulator. Stores the quotient in the accumulator and the 1's complement of the remainder on the stack.																		
EOR (Note 1)	When T=0 A←A∀M When T=1 M(X)←M(X)∀M	"Exclusive-ORs" the contents of accumulator and memory. The results are stored in the accumulator. "Exclusive-ORs" the contents of the memory specified by the addressing mode and the contents of the memory at the address indicated by index register X. The results are stored into the memory at the address indicated by index register X.				49	2	2							45	3	2			
INC	A←A+1 or M←M+1	Increments the contents of accumulator or memory by 1.							3A	2	1				E6	5	2			
INX	X←X+1	Increments the contents of index register X by 1.	E8	2	1															
INY	Y←Y+1	Increments the contents of index register Y by 1.	C8	2	1															

APPENDIX

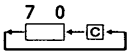
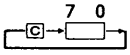
6.9 Machine instruction table

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
0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	N	V	T	B	D	I	Z	C
																											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	•
																																	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	•

Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT,A			ZP			BIT,ZP		
			0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#
JMP	If addressing mode is ABS $PC_L \leftarrow AD_L$ $PC_H \leftarrow AD_H$ If addressing mode is IND $PC_L \leftarrow M(AD_H, AD_L)$ $PC_H \leftarrow M(AD_H, AD_L+1)$ If addressing mode is ZP, IND $PC_L \leftarrow M(00, AD_L)$ $PC_H \leftarrow M(00, AD_L+1)$	Jumps to the specified address.																		
JSR	$M(S) \leftarrow PC_H$ $S \leftarrow S-1$ $M(S) \leftarrow PC_L$ $S \leftarrow S-1$ After executing the above, if addressing mode is ABS, $PC_L \leftarrow AD_L$ $PC_H \leftarrow AD_H$ If addressing mode is SP, $PC_L \leftarrow AD_L$ $PC_H \leftarrow FF$ If addressing mode is ZP, IND, $PC_L \leftarrow M(00, AD_L)$ $PC_H \leftarrow M(00, AD_L+1)$	After storing contents of program counter in stack, and jumps to the specified address.																		
LDA (Note 2)	When T=0 A←M When T=1 M(X)←M	Load accumulator with contents of memory. Load memory indicated by index register X with contents of memory specified by the addressing mode.				A9	2	2							A5	3	2			
LDM	M←nn	Load memory with immediate value.													3C	4	3			
LDX	X←M	Load index register X with contents of memory.				A2	2	2							A6	3	2			
LDY	Y←M	Load index register Y with contents of memory.				A0	2	2							A4	3	2			
LSR	7 0 0 → □ → C	Shift the contents of accumulator or memory to the right by one bit. The low order bit of accumulator or memory is stored in carry, 7th bit is cleared.							4A	2	1				46	5	2			
MUL	M(S)·A←A×M(zz+X) S←S− 1	Multiplies the accumulator with the contents of memory specified by the zero page X addressing mode and stores the high byte of the result on the stack and the low byte in the accumulator.																		
NOP	PC←PC+1	No operation.	EA	2	1															
ORA (Note 1)	When T=0 A←AVM When T=1 M(X)←M(X)VM	"Logical OR's" the contents of memory and accumulator. The result is stored in the accumulator. "Logical OR's" the contents of memory indicated by index register X and contents of memory specified by the addressing mode. The result is stored in the memory specified by index register X.				09	2	2							05	3	2			

6.9 Machine instruction table

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
						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	•
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	•

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	#
PHA	M(S)←A S←S−1	Saves the contents of the accumulator in memory at the address indicated by the stack pointer and decrements the contents of stack pointer by 1.	48	3	1														
PHP	M(S)←PS S←S−1	Saves the contents of the processor status register in memory at the address indicated by the stack pointer and decrements the contents of the stack pointer by 1.	08	3	1														
PLA	S←S+1 A←M(S)	Increments the contents of the stack pointer by 1 and restores the accumulator from the memory at the address indicated by the stack pointer.	68	4	1														
PLP	S←S+1 PS←M(S)	Increments the contents of stack pointer by 1 and restores the processor status register from the memory at the address indicated by the stack pointer.	28	4	1														
ROL		Shifts the contents of the memory or accumulator to the left by one bit. The high order bit is shifted into the carry flag and the carry flag is shifted into the low order bit.							2A	2	1				26	5	2			
ROR		Shifts the contents of the memory or accumulator to the right by one bit. The low order bit is shifted into the carry flag and the carry flag is shifted into the high order bit.							6A	2	1				66	5	2			
RRF		Rotates the contents of memory to the right by 4 bits.													82	8	2			
RTI	S←S+1 PS←M(S) S←S+1 PC_L←M(S) S←S+1 PC_H←M(S)	Returns from an interrupt routine to the main routine.	40	6	1														
RTS	S←S+1 PC_L←M(S) S←S+1 PC_H←M(S)	Returns from a subroutine to the main routine.	60	6	1														
SBC (Note 1) (Note 5)	When T=0 A←A−M−$\overline{C}$ When T=1 M(X)←M(X)−M−$\overline{C}$	Subtracts the contents of memory and complement of carry flag from the contents of accumulator. The results are stored into the accumulator. Subtracts contents of complement of carry flag and contents of the memory indicated by the addressing mode from the memory at the address indicated by index register X. The results are stored into the memory of the address indicated by index register X.				E9	2	2							E5	3	2			
SEB	A_b or M_b←1	Sets the specified bit in the accumulator or memory to "1."										0B+20i	2	1				0F+20i	5	2
SEC	C←1	Sets the contents of the carry flag to "1."	38	2	1														
SED	D←1	Sets the contents of the decimal mode flag to "1."	F8	2	1														
SEI	I←1	Sets the contents of the interrupt disable flag to "1."	78	2	1														
SET	T←1	Sets the contents of the index X mode flag to "1."	32	2	1														

6.9 Machine instruction table

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	•
																															(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)							
																															•	•	•	•	•	•	•	•
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	•	•	•	•	•

Symbol	Function	Details	Addressing mode																	
			IMP			IMM			A			BIT,A			ZP			BIT,ZP		
			0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#	0P	n	#
STA	M←A	Stores the contents of accumulator in memory.													85	4	2			
STP		Stops the oscillator.	42	2	1															
STX	M←X	Stores the contents of index register X in memory.													86	4	2			
STY	M←Y	Stores the contents of index register Y in memory.													84	4	2			
TAX	X←A	Transfers the contents of the accumulator to index register X.	AA	2	1															
TAY	Y←A	Transfers the contents of the accumulator to index register Y.	A8	2	1															
TST	M=0?	Tests whether the contents of memory are "0" or not.													64	3	2			
TSX	X←S	Transfers the contents of the stack pointer to index register X.	BA	2	1															
TXA	A←X	Transfers the contents of index register X to the accumulator.	8A	2	1															
TXS	S←X	Transfers the contents of index register X to the stack pointer.	9A	2	1															
TYA	A←Y	Transfers the contents of index register Y to the accumulator.	98	2	1															
WIT		Stops the internal clock.	C2	2	1															

Note 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.

APPENDIX

6.9 Machine instruction table

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
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	•
																																	•	•	•	•	•	•	•	•

Symbol	Contents
IMP	Implied addressing mode
IMM	Immediate addressing mode
A	Accumulator or Accumulator addressing mode
BIT, A	Accumulator bit relative addressing mode
ZP	Zero page addressing mode
BIT, ZP	Zero page bit relative addressing mode
ZP, X	Zero page X addressing mode
ZP, Y	Zero page Y addressing mode
ABS	Absolute addressing mode
ABS, X	Absolute X addressing mode
ABS, Y	Absolute Y addressing mode
IND	Indirect absolute addressing mode
ZP, IND	Zero page indirect absolute addressing mode
IND, X	Indirect X addressing mode
IND, Y	Indirect Y addressing mode
REL	Relative addressing mode
SP	Special page addressing mode
C	Carry flag
Z	Zero flag
I	Interrupt disable flag
D	Decimal mode flag
B	Break flag
T	X-modified arithmetic mode flag
V	Overflow flag
N	Negative flag

Symbol	Contents
+	Addition
−	Subtraction
∧	Logical OR
∨	Logical AND
∀	Logical exclusive OR
—	Negation
←	Shows direction of data flow
X	Index register X
Y	Index register Y
S	Stack pointer
PC	Program counter
PS	Processor status register
PC_H	8 high-order bits of program counter
PC_L	8 low-order bits of program counter
AD_H	8 high-order bits of address
AD_L	8 low-order bits of address
FF	FF in Hexadecimal notation
nn	Immediate value
M	Memory specified by address designation of any addressing mode
M (X)	Memory of address indicated by contents of index register X
M (S)	Memory of address indicated by contents of stack pointer
$M(AD_H, AD_L)$	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.
$M(00, AD_L)$	Contents of address indicated by zero page AD_L
A_b	1 bit of accumulator
M_b	1 bit of memory
OP	Opcode
n	Number of cycles
#	Number of bytes

6.10 Instruction code table

D3–D0 / D7–D4 (Hexadecimal notation)		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	BRK	ORA IND,X	JSR ZP,IND	BBS 0,A	—	ORA ZP	ASL ZP	BBS 0,ZP	PHP	ORA IMM	ASL A	SEB 0,A	—	ORA ABS	ASL ABS	SEB 0,ZP
0001	1	BPL	ORA IND,Y	CLT	BBC 0,A	—	ORA ZP,X	ASL ZP,X	BBC 0,ZP	CLC	ORA ABS,Y	DEC A	CLB 0,A	—	ORA ABS,X	ASL ABS,X	CLB 0,ZP
0010	2	JSR ABS	AND IND,X	JSR SP	BBS 1,A	BIT ZP	AND ZP	ROL ZP	BBS 1,ZP	PLP	AND IMM	ROL A	SEB 1,A	BIT ABS	AND ABS	ROL ABS	SEB 1,ZP
0011	3	BMI	AND IND,Y	SET	BBC 1,A	—	AND ZP,X	ROL ZP,X	BBC 1,ZP	SEC	AND ABS,Y	INC A	CLB 1,A	LDM ZP	AND ABS,X	ROL ABS,X	CLB 1,ZP
0100	4	RTI	EOR IND,X	STP	BBS 2,A	COM ZP	EOR ZP	LSR ZP	BBS 2,ZP	PHA	EOR IMM	LSR A	SEB 2,A	JMP ABS	EOR ABS	LSR ABS	SEB 2,ZP
0101	5	BVC	EOR IND,Y	—	BBC 2,A	—	EOR ZP,X	LSR ZP,X	BBC 2,ZP	CLI	EOR ABS,Y	—	CLB 2,A	—	EOR ABS,X	LSR ABS,X	CLB 2,ZP
0110	6	RTS	ADC IND,X	—	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,X	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	—	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

6.11 Mask ROM ordering method

When placing an order, please submit the information described below.

① M37221M4-XXXSP Mask ROM Ordering Confirmation Form.....1 set
(Please use the pages P6-65 to P6-67)
M27221M8-XXXSP Mask ROM Ordering Confirmation Form.....1 set
(Please use the pages P6-68 to P6-70)
M37221M6-XXXSP/FP Mask ROM Ordering Confirmation Form.....1 set
(Please use the pages P6-71 to P6-73)
M27221MA-XXXSP Mask ROM Ordering Confirmation From.....1 set
(Please use the pages P6-74 to P6-76)
M27220M3-XXXSP/FP Mask ROM Ordering Confirmation From.....1 set
(Please use the pages P6-77 to P6-79)
②Data to be written to mask ROM.....EPROM (DIP Type 27C101)
(Please provide 3 sets containing the identical data)
③Mark Specification Form.....1 set
(Please use the pages P6-80 and P6-81)

GZZ–SH10–10B < 59B0 >

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221M4-XXXSP
MITSUBISHI ELECTRIC

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.

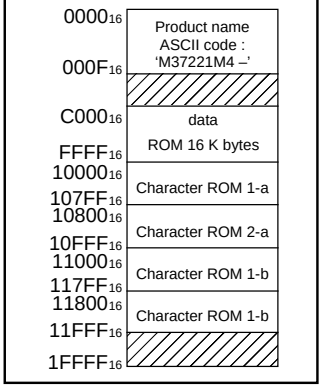
Three EPROMs are required for each pattern.

If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ **27C101**



(1) Set "FF$_{16}$" in the shaded area.

(2) Write the ASCII codes that indicates the product name of "M37221M4–" to addresses 0000$_{16}$ to 000F$_{16}$.

EPROM data check item (Refer the EPROM data and check " ✓" in the appropriate box)

- Do you set "FF$_{16}$" in the shaded area ? → Yes ☐
- Do you write the ASCII codes that indicates the product name of "M37221M4–" to addresses 0000$_{16}$ to 000F$_{16}$? → Yes ☐

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type package being ordered fill out the appropriate mark specification form (42P4B for M37221M4-XXXSP) and attach to the mask ROM confirmation form.

✵ 3. Comments

APPENDIX

6.11 Mask ROM ordering method

GZZ–SH10–10B <59B0 >

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221M4-XXXSP
MITSUBISHI ELECTRIC

Writing the product name and character ROM data onto EPROMs

Addresses 0000_{16} to $000F_{16}$ store the product name, and addresses 10000_{16} to $11FFF_{16}$ store the character pattern.
If the name of the product contained in the EPROMs does not match the name on the mask ROM confirmation form, the ROM processing is disabled. Write the data correctly.

1. Inputting the name of the product with the ASCII code
   ASCII codes 'M37221M4-' are listed on the right.
   The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = 4 D $_{16}$	0008_{16}	'–' = 2 D $_{16}$
0001_{16}	'3' = 3 3 $_{16}$	0009_{16}	F F $_{16}$
0002_{16}	'7' = 3 7 $_{16}$	$000A_{16}$	F F $_{16}$
0003_{16}	'2' = 3 2 $_{16}$	$000B_{16}$	F F $_{16}$
0004_{16}	'2' = 3 2 $_{16}$	$000C_{16}$	F F $_{16}$
0005_{16}	'1' = 3 1 $_{16}$	$000D_{16}$	F F $_{16}$
0006_{16}	'M' = 4 D $_{16}$	$000E_{16}$	F F $_{16}$
0007_{16}	'4' = 3 4 $_{16}$	$000F_{16}$	F F $_{16}$

2. Inputting the character ROM
   Input the character ROM data by dividing it into character ROM1 and character ROM2. For the character ROM data, see the next page and on.

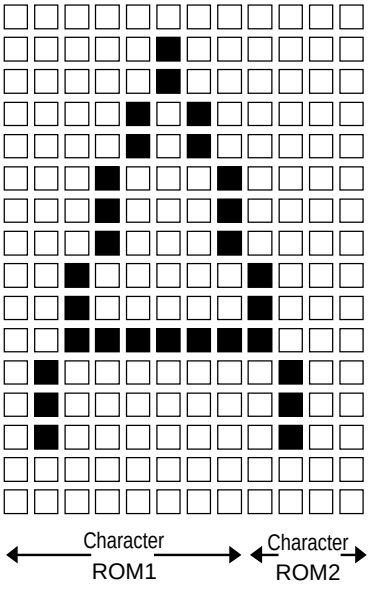
GZZ–SH10–10B< 59B0 >

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221M4-XXXSP
MITSUBISHI ELECTRIC

The structure of character ROM (divided of 12 ×16 dots font)

Example

Character code "$1A_{16}$"



(Note)
Write the character code "00_{16}" to "$7F_{16}$" to addreses 10000_{16} to $10FFF_{16}$.
Write the character code "80_{16}" to "FF_{16}" to addreses 11000_{16} to $11FFF_{16}$.

Example $101A0_{16}$ to $101AF_{16}$

	b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	
0	□ □ □ □ □ □ □ □	00_{16}
1	□ □ □ □ □ ■ □ □	04_{16}
2	□ □ □ □ □ ■ □ □	04_{16}
3	□ □ □ □ ■ □ ■ □	$0A_{16}$
4	□ □ □ □ ■ □ ■ □	$0A_{16}$
5	□ □ □ ■ □ □ □ ■	11_{16}
6	□ □ □ ■ □ □ □ ■	11_{16}
7	□ □ □ ■ □ □ □ ■	11_{16}
8	□ □ ■ □ □ □ □ □	20_{16}
9	□ □ ■ □ □ □ □ □	20_{16}
A	□ □ ■ ■ ■ ■ ■ ■	$3F_{16}$
B	□ ■ □ □ □ □ □ □	40_{16}
C	□ ■ □ □ □ □ □ □	40_{16}
D	□ ■ □ □ □ □ □ □	40_{16}
E	□ □ □ □ □ □ □ □	00_{16}
F	□ □ □ □ □ □ □ □	00_{16}

Example $109A0_{16}$ to $109AF_{16}$

	b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	
0	■ ■ ■ ■ □ □ □ □	$F0_{16}$
1	■ ■ ■ ■ □ □ □ □	$F0_{16}$
2	■ ■ ■ ■ □ □ □ □	$F0_{16}$
3	■ ■ ■ ■ □ □ □ □	$F0_{16}$
4	■ ■ ■ ■ □ □ □ □	$F0_{16}$
5	■ ■ ■ ■ □ □ □ □	$F0_{16}$
6	■ ■ ■ ■ □ □ □ □	$F0_{16}$
7	■ ■ ■ ■ □ □ □ □	$F0_{16}$
8	■ ■ ■ ■ ■ □ □ □	$F8_{16}$
9	■ ■ ■ ■ ■ □ □ □	$F8_{16}$
A	■ ■ ■ ■ ■ □ □ □	$F8_{16}$
B	■ ■ ■ ■ □ ■ □ □	$F4_{16}$
C	■ ■ ■ ■ □ ■ □ □	$F4_{16}$
D	■ ■ ■ ■ □ ■ □ □	$F4_{16}$
E	■ ■ ■ ■ □ □ □ □	$F0_{16}$
F	■ ■ ■ ■ □ □ □ □	$F0_{16}$

F_{16}

GZZ–SH11–58B < 72A0 >

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221M8-XXXSP
MITSUBISHI ELECTRIC

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.

Three EPROMs are required for each pattern.

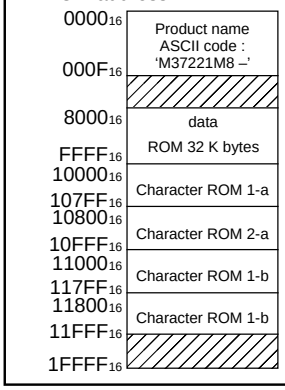
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ **27C101**

EPROM address



(1) Set "FF$_{16}$" in the shaded area.

(2) Write the ASCII codes that indicates the product name of "M37221M8–" to addresses 0000$_{16}$ to 000F$_{16}$.

EPROM data check item (Refer the EPROM data and check " ✓" in the appropriate box)

- Do you set "FF$_{16}$" in the shaded area ? → Yes ☐
- Do you write the ASCII codes that indicates the product name of "M37221M8–" to addresses 0000$_{16}$ to 000F$_{16}$? → Yes ☐

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type package being ordered fill out the appropriate mark specification form (42P4B for M37221M8-XXXSP) and attach to the mask ROM confirmation form.

✵ 3. Comments

GZZ–SH11–58B <72A0 >

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221M8-XXXSP
MITSUBISHI ELECTRIC

Writing the product name and character ROM data onto EPROMs

Addresses 0000_{16} to $000F_{16}$ store the product name, and addresses 10000_{16} to $11FFF_{16}$ store the character pattern.
If the name of the product contained in the EPROMs does not match the name on the mask ROM confirmation form, the ROM processing is disabled. Write the data correctly.

1. Inputting the name of the product with the ASCII code
   ASCII codes 'M37221M8-' are listed on the right.
   The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = 4 D $_{16}$	0008_{16}	'–' = 2 D $_{16}$
0001_{16}	'3' = 3 3 $_{16}$	0009_{16}	F F $_{16}$
0002_{16}	'7' = 3 7 $_{16}$	$000A_{16}$	F F $_{16}$
0003_{16}	'2' = 3 2 $_{16}$	$000B_{16}$	F F $_{16}$
0004_{16}	'2' = 3 2 $_{16}$	$000C_{16}$	F F $_{16}$
0005_{16}	'1' = 3 1 $_{16}$	$000D_{16}$	F F $_{16}$
0006_{16}	'M' = 4 D $_{16}$	$000E_{16}$	F F $_{16}$
0007_{16}	'8' = 3 8 $_{16}$	$000F_{16}$	F F $_{16}$

2. Inputting the character ROM
   Input the character ROM data by dividing it into character ROM1 and character ROM2. For the character ROM data, see the next page and on.

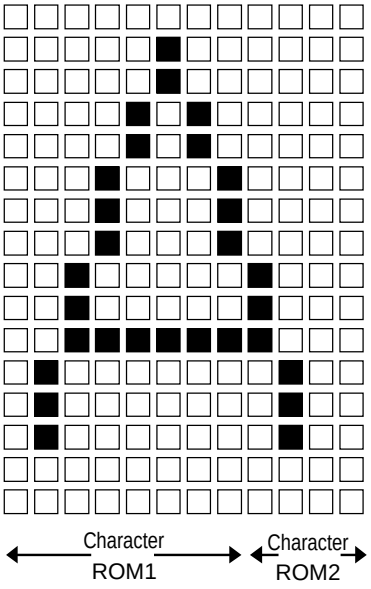
GZZ–SH11–58B< 72A0 >

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221M8-XXXSP
MITSUBISHI ELECTRIC

The structure of character ROM (divided of 12 ×16 dots font)

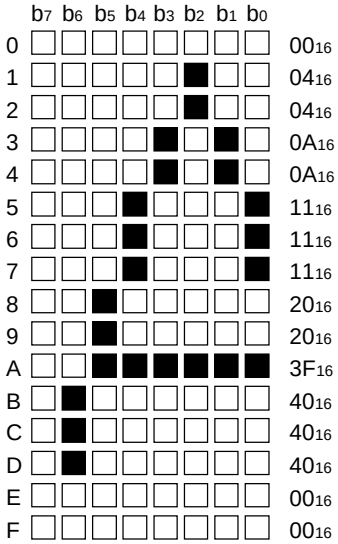
Example

Character code "$1A_{16}$"

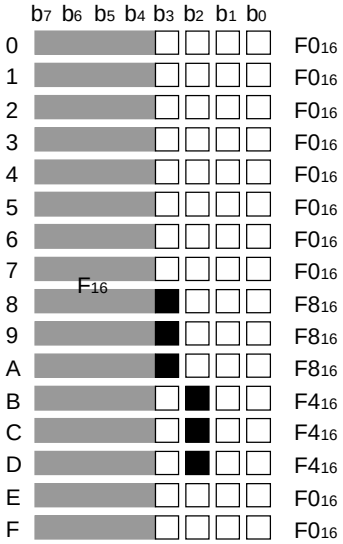


(Note)
Write the character code "00_{16}" to "$7F_{16}$" to addreses 10000_{16} to $10FFF_{16}$.
Write the character code "80_{16}" to "FF_{16}" to addreses 11000_{16} to $11FFF_{16}$.

Example $101A0_{16}$ to $101AF_{16}$



Example $109A0_{16}$ to $109AF_{16}$



(3/3)

GZZ–SH09–46B < 52C0 >

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221M6-XXXSP/FP
MITSUBISHI ELECTRIC

Mask ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.

Three EPROMs are required for each pattern.

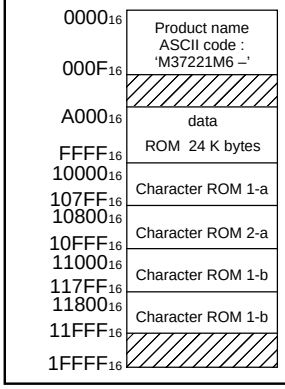
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ **27C101**

EPROM address



(1) Set "FF$_{16}$" in the shaded area.

(2) Write the ASCII codes that indicates the product name of "M37221M6–" to addresses 0000$_{16}$ to 000F$_{16}$.

EPROM data check item (Refer the EPROM data and check " ✓" in the appropriate box)

- Do you set "FF$_{16}$" in the shaded area ? → Yes ☐
- Do you write the ASCII codes that indicates the product name of "M37221M6–" to addresses 0000$_{16}$ to 000F$_{16}$? → Yes ☐

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type package being ordered fill out the appropriate mark specification form (42P4B for M37221M6-XXXSP, 42P2R-A for M37221M6-XXXFP) and attach to the mask ROM confirmation form.

✵ 3. Comments

GZZ–SH09–46B <52C0 >

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221M6-XXXSP/FP
MITSUBISHI ELECTRIC

Writing the product name and character ROM data onto EPROMs

Addresses 0000_{16} to $000F_{16}$ store the product name, and addresses 10000_{16} to $11FFF_{16}$ store the character pattern.
If the name of the product contained in the EPROMs does not match the name on the mask ROM confirmation form, the ROM processing is disabled. Write the data correctly.

1. Inputting the name of the product with the ASCII code
   ASCII codes 'M37221M6-' are listed on the right.
   The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = 4 D $_{16}$	0008_{16}	'–' = 2 D $_{16}$
0001_{16}	'3' = 3 3 $_{16}$	0009_{16}	F F $_{16}$
0002_{16}	'7' = 3 7 $_{16}$	$000A_{16}$	F F $_{16}$
0003_{16}	'2' = 3 2 $_{16}$	$000B_{16}$	F F $_{16}$
0004_{16}	'2' = 3 2 $_{16}$	$000C_{16}$	F F $_{16}$
0005_{16}	'1' = 3 1 $_{16}$	$000D_{16}$	F F $_{16}$
0006_{16}	'M' = 4 D $_{16}$	$000E_{16}$	F F $_{16}$
0007_{16}	'6' = 3 6 $_{16}$	$000F_{16}$	F F $_{16}$

2. Inputting the character ROM
   Input the character ROM data by dividing it into character ROM1 and character ROM2. For the character ROM data, see the next page and on.

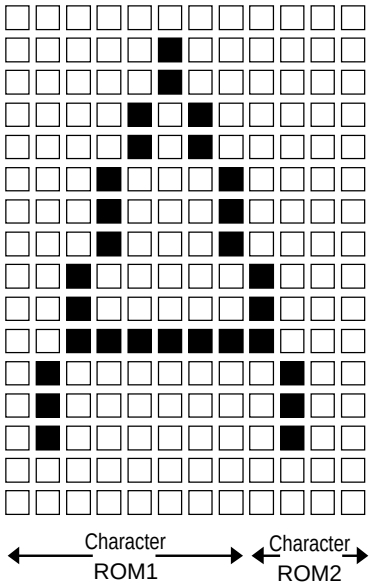
GZZ–SH09–46B< 52C0 >

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221M6-XXXSP/FP
MITSUBISHI ELECTRIC

The structure of character ROM (divided of 12 ×16 dots font)

Example

Character code "1A16"



(Note)
Write the character code "00 16" to "7F16" to addreses 10000 16 to 10FFF16.
Write the character code "80 16" to "FF16" to addreses 11000 16 to 11FFF16.

Example 101A016 to 101AF16

	b7 b6 b5 b4 b3 b2 b1 b0	
0	□□□□□□□□	0016
1	□□□□□■□□	0416
2	□□□□□■□□	0416
3	□□□□■□■□	0A16
4	□□□□■□■□	0A16
5	□□□■□□□■	1116
6	□□□■□□□■	1116
7	□□□■□□□■	1116
8	□□■□□□□□	2016
9	□□■□□□□□	2016
A	□□■■■■■■	3F16
B	□■□□□□□□	4016
C	□■□□□□□□	4016
D	□■□□□□□□	4016
E	□□□□□□□□	0016
F	□□□□□□□□	0016

Example 109A016 to 109AF16

	b7 b6 b5 b4 b3 b2 b1 b0	
0	■■■■□□□□	F016
1	■■■■□□□□	F016
2	■■■■□□□□	F016
3	■■■■□□□□	F016
4	■■■■□□□□	F016
5	■■■■□□□□	F016
6	■■■■□□□□	F016
7	■■■■□□□□	F016
8	■■■■■□□□	F816
9	■■■■■□□□	F816
A	■■■■■□□□	F816
B	■■■■□■□□	F416
C	■■■■□■□□	F416
D	■■■■□■□□	F416
E	■■■■□□□□	F016
F	■■■■□□□□	F016

F16

APPENDIX

6.11 Mask ROM ordering method

GZZ–SH10–46B < 5ZA0 >

SERIES MELPS 740 MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221MA-XXXSP
MITSUBISHI ELECTRIC

Mask ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.

Three EPROMs are required for each pattern.

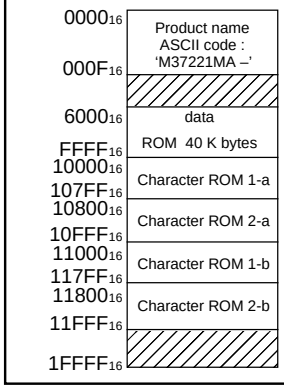
If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ **27C101**

EPROM address



(1) Set "FF$_{16}$" in the shaded area.

(2) Write the ASCII codes that indicates the product name of "M37221MA–" to addresses 0000$_{16}$ to 000F$_{16}$.

EPROM data check item (Refer the EPROM data and check " ✓" in the appropriate box)

- Do you set "FF$_{16}$" in the shaded area ? → Yes ☐
- Do you write the ASCII codes that indicates the product name of "M37221MA–" to addresses 0000$_{16}$ to 000F$_{16}$? → Yes ☐

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type package being ordered fill out the appropriate mark specification form (42P4B for M37221MA-XXXSP) and attach to the mask ROM confirmation form.

✵ 3. Comments

GZZ–SH10–46B <5ZA0 >

SERIES MELPS 740 MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221MA-XXXSP
MITSUBISHI ELECTRIC

Writing the product name and character ROM data onto EPROMs

Addresses 0000_{16} to $000F_{16}$ store the product name, and addresses 10000_{16} to $11FFF_{16}$ store the character pattern. If the name of the product contained in the EPROMs does not match the name on the mask ROM confirmation form, the ROM processing is disabled. Write the data correctly.

1. Inputting the name of the product with the ASCII code
   ASCII codes 'M37221MA-' are listed on the right.
   The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = 4 D $_{16}$	0008_{16}	'–' = 2 D $_{16}$
0001_{16}	'3' = 3 3 $_{16}$	0009_{16}	F F $_{16}$
0002_{16}	'7' = 3 7 $_{16}$	$000A_{16}$	F F $_{16}$
0003_{16}	'2' = 3 2 $_{16}$	$000B_{16}$	F F $_{16}$
0004_{16}	'2' = 3 2 $_{16}$	$000C_{16}$	F F $_{16}$
0005_{16}	'1' = 3 1 $_{16}$	$000D_{16}$	F F $_{16}$
0006_{16}	'M' = 4 D $_{16}$	$000E_{16}$	F F $_{16}$
0007_{16}	'A' = 4 1 $_{16}$	$000F_{16}$	F F $_{16}$

2. Inputting the character ROM
   Input the character ROM data by dividing it into character ROM1 and character ROM2. For the character ROM data, see the next page and on.

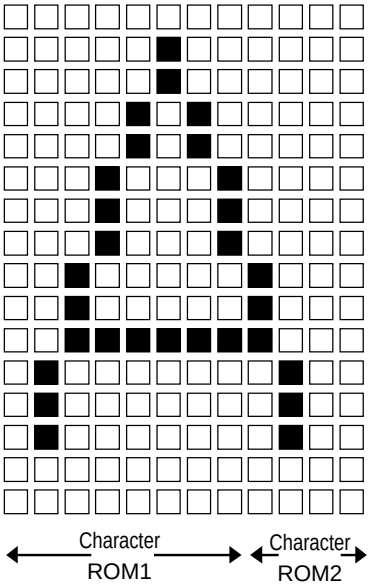
GZZ–SH10–46B< 5ZA0 >

SERIES MELPS 740 MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37221MA-XXXSP
MITSUBISHI ELECTRIC

The structure of character ROM (divided of 12 ×16 dots font)

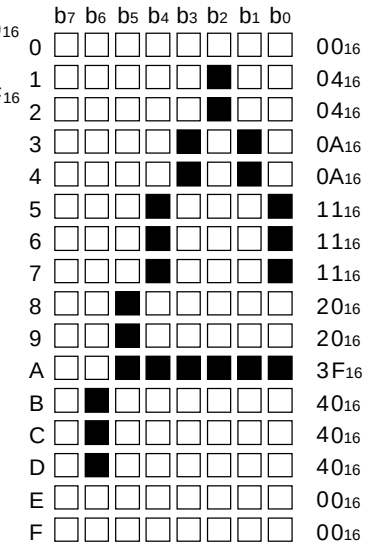
Example

Character code
"1A16"

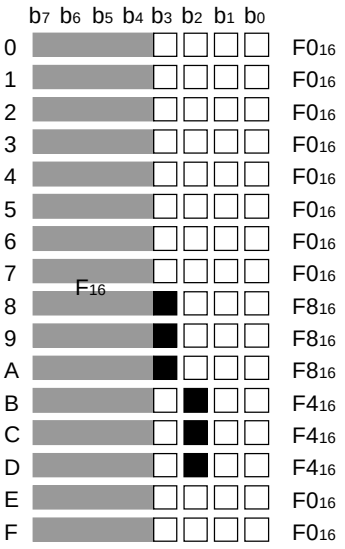


(Note)
Write the character code "00 16" to "7F16"
to addresses 10000 16 to 10FFF16.
Write the character code "80 16" to "FF16"
to addresses 11000 16 to 11FFF16.

Example 101A0$_{16}$ to 101AF$_{16}$



Example 109A0$_{16}$ to 109AF$_{16}$



GZZ–SH09–72B < 56B0 >

Mask ROM number	

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37220M3-XXXSP
MITSUBISHI ELECTRIC

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please fill in all items marked ✵.

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵ 1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.

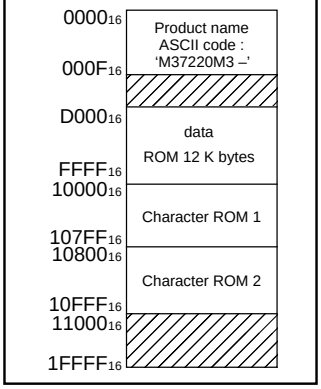
Three EPROMs are required for each pattern.

If at least two of the three sets of EPROMs submitted contain identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differs from this data. Thus, extreme care must be taken to verify the data in the submitted EPROMs.

Checksum code for entire EPROM | | | | | (hexadecimal notation)

EPROM type (indicate the type used)

☐ **27C101**



(1) Set "FF$_{16}$" in the shaded area.

(2) Write the ASCII codes that indicates the product name of "M37220M3–" to addresses 0000$_{16}$ to 000F$_{16}$.

EPROM data check item (Refer the EPROM data and check " ✓" in the appropriate box)

- Do you set "FF$_{16}$" in the shaded area ? → Yes ☐
- Do you write the ASCII codes that indicates the product name of "M37220M3–" to addresses 0000$_{16}$ to 000F$_{16}$? → Yes ☐

✵ 2. Mark specification

Mark specification must be submitted using the correct form for the type package being ordered fill out the appropriate mark specification form (42P4B for M37220M3-XXXSP) and attach to the mask ROM confirmation form.

✵ 3. Comments

GZZ–SH09–72B <56B0 >

740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37220M3-XXXSP
MITSUBISHI ELECTRIC

Writing the product name and character ROM data onto EPROMs

Addresses 0000_{16} to $000F_{16}$ store the product name, and addresses 10000_{16} to $10FFF_{16}$ store the character pattern. If the name of the product contained in the EPROMs does not match the name on the mask ROM confirmation form, the ROM processing is disabled. Write the data correctly.

1. Inputting the name of the product with the ASCII code
   ASCII codes 'M37220M3-' are listed on the right.
   The addresses and data are in hexadecimal notation.

Address		Address	
0000_{16}	'M' = 4 D $_{16}$	0008_{16}	'–' = 2 D $_{16}$
0001_{16}	'3' = 3 3 $_{16}$	0009_{16}	F F $_{16}$
0002_{16}	'7' = 3 7 $_{16}$	$000A_{16}$	F F $_{16}$
0003_{16}	'2' = 3 2 $_{16}$	$000B_{16}$	F F $_{16}$
0004_{16}	'2' = 3 2 $_{16}$	$000C_{16}$	F F $_{16}$
0005_{16}	'0' = 3 0 $_{16}$	$000D_{16}$	F F $_{16}$
0006_{16}	'M' = 4 D $_{16}$	$000E_{16}$	F F $_{16}$
0007_{16}	'3' = 3 3 $_{16}$	$000F_{16}$	F F $_{16}$

2. Inputting the character ROM
   Input the character ROM data by dividing it into character ROM1 and character ROM2. For the character ROM data, see the next page and on.

GZZ–SH09–72B< 56B0 >

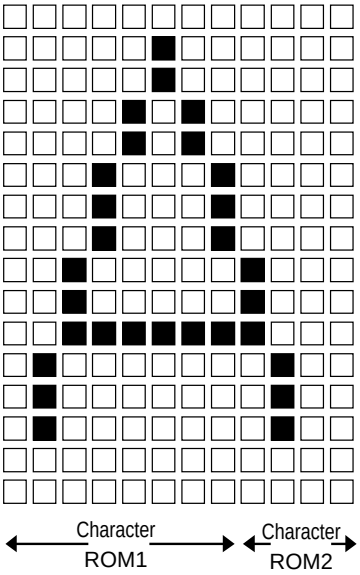
740 FAMILY MASK ROM CONFIRMATION FORM
SINGLE-CHIP MICROCOMPUTER M37220M3-XXXSP
MITSUBISHI ELECTRIC

The structure of character ROM (divided of 12 ×16 dots font)

Example

Character code

"$1A_{16}$"



Example $101A0_{16}$ to $101AF_{16}$

	b7 b6 b5 b4 b3 b2 b1 b0	
0	□□□□□□□□	00_{16}
1	□□□□□■□□	04_{16}
2	□□□□□■□□	04_{16}
3	□□□□■□■□	$0A_{16}$
4	□□□□■□■□	$0A_{16}$
5	□□□■□□□■	11_{16}
6	□□□■□□□■	11_{16}
7	□□□■□□□■	11_{16}
8	□□■□□□□□	20_{16}
9	□□■□□□□□	20_{16}
A	□□■■■■■■	$3F_{16}$
B	□■□□□□□□	40_{16}
C	□■□□□□□□	40_{16}
D	□■□□□□□□	40_{16}
E	□□□□□□□□	00_{16}
F	□□□□□□□□	00_{16}

Example $109A0_{16}$ to $109AF_{16}$

	b7 b6 b5 b4 b3 b2 b1 b0	
0	■■■■□□□□	$F0_{16}$
1	■■■■□□□□	$F0_{16}$
2	■■■■□□□□	$F0_{16}$
3	■■■■□□□□	$F0_{16}$
4	■■■■□□□□	$F0_{16}$
5	■■■■□□□□	$F0_{16}$
6	■■■■□□□□	$F0_{16}$
7	■■■■□□□□	$F0_{16}$
8	■■■■■□□□	$F8_{16}$
9	■■■■■□□□	$F8_{16}$
A	■■■■■□□□	$F8_{16}$
B	■■■■□■□□	$F4_{16}$
C	■■■■□■□□	$F4_{16}$
D	■■■■□■□□	$F4_{16}$
E	■■■■□□□□	$F0_{16}$
F	■■■■□□□□	$F0_{16}$

F_{16}

(3/3)

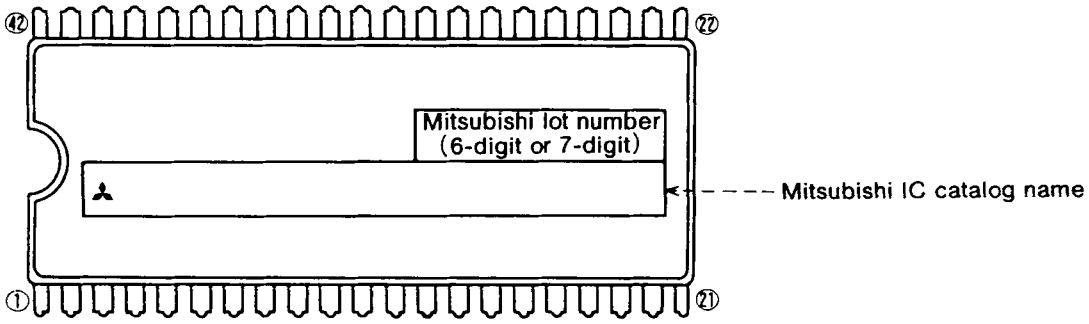
6.12 Mark specification form

42P4B (42-PIN SHRINK DIP) MARK SPECIFICATION FORM

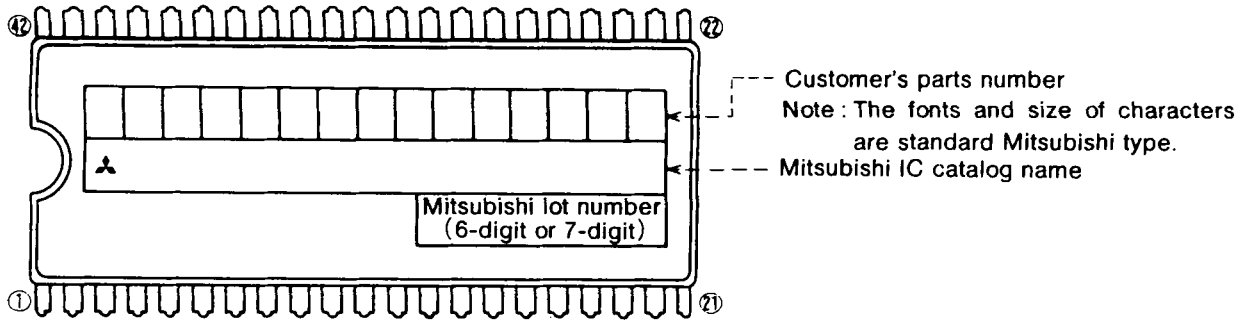
Mitsubishi IC catalog name ☐

Please choose one of the marking types below (A, B, C), and enter the Mitsubishi IC catalog name and the special mark (if needed).

A. Standard Mitsubishi Mark

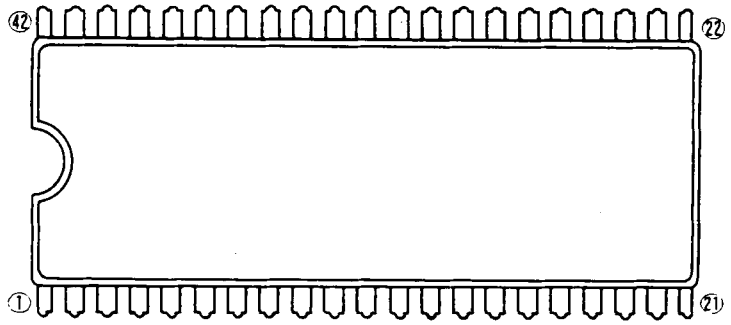


B. Customer's Parts Number + Mitsubishi Catalog Name



Note1 : The mark field should be written right aligned.
2 : The fonts and size of characters are standard Mitsubishi type.
3 : Customer's parts number can be up to 15 characters :
Only 0~9, A~Z, +, −, /, (,), &, ©, . (period), and , (comma) are usable.
4 : If the Mitsubishi logo ⅄ is not required, check the box on the right. ⅄Mitsubishi logo is not required ☐

C. Special Mark Required



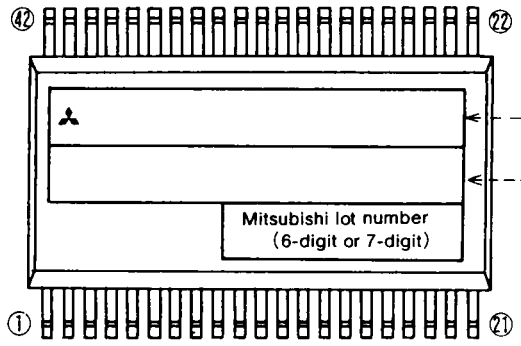
Note1 : If the special mark is to be printed, indicate the desired layout of the mark in the upper figure. The layout will be duplicated as close as possible. Mitsubishi lot number (6-digit or 7-digit) and mask ROM number (3-digit) are always marked.
2 : If the customer's trade mark logo must be used in the special mark, check the box below. Please submit a clean original of the logo.
For the new special character fonts a clean font original (ideally logo drawing) must be submitted.
Special logo required
The standard Mitsubishi font is used for all characters except for a logo. ☐

42P2R-A (42-PIN SHRINK SOP) MARK SPECIFICATION FORM

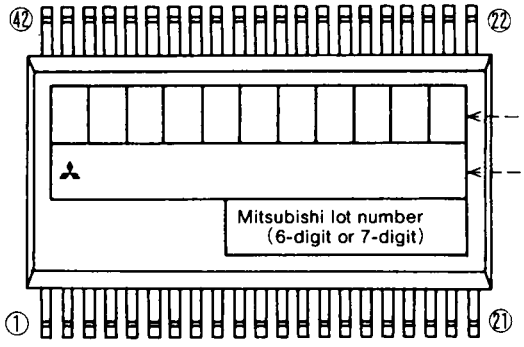
Mitsubishi IC catalog name

Please choose one of the marking types below (A, B, C), and enter the Mitsubishi catalog name and the special mark (if needed).

A. Standard Mitsubishi Mark



B. Customer's Parts Number + Mitsubishi catalog name



Note : The fonts and size of characters are standard Mitsubishi type.

Note1 : The mark field should be written right aligned.

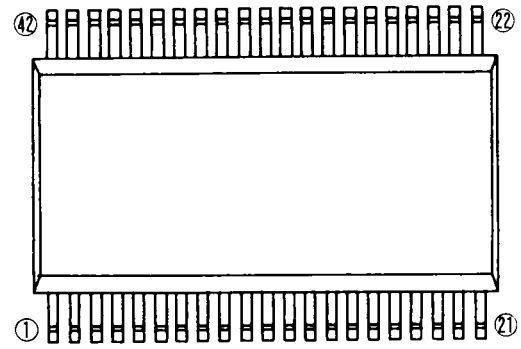
2 : The fonts and size of characters are standard Mitsubishi type.

3 : Customer's Parts Number can be up to 11 characters : Only 0~9, A~Z, +, −, /, (,), &, ©, · (periods), , (commas) are usable.

4 : If the Mitsubishi logo is not required, check the box below.

Mitsubishi logo is not required ☐

C. Special Mark Required



Note1 : If the Special Mark is to be printed, indicate the desired layout of the mark in the left figure. The layout will be duplicated as close as possible. Mitsubishi lot number (6-digit or 7-digit) and Mask ROM number (3-digit) are always marked.

2 : If the customer's trade mark logo must be used in the Special Mark, check the box below. Please submit a clean original of the logo.
For the new special character fonts a clean font original (ideally logo drawing) must be submitted.

Special logo required ☐

3 : The standard Mitsubishi font is used for all characters except for a logo.

MITSUBISHI SEMICONDUCTORS
USER'S MANUAL
7220 Group

Jul. First Edition 1997

Editioned by
Committee of editing of Mitsubishi Semiconductor USER'S MANUAL

Published by
Mitsubishi Electric Corp., Semiconductor Marketing Division

User's Manual
7220 Group

Renesas Technology Corp.
Nippon Bldg.,6-2,Otemachi 2-chome,Chiyoda-ku,Tokyo,100-0004 Japan

New publication, effective Jul. 1997.
Specifications subject to change without notice.

REVISION DESCRIPTION LIST	7220 GROUP USER'S MANUAL

Rev. No.	Revision Description	Rev. date
1.0	First Edition	9708
2.0	Information about copywright note, revision number, release date added (last page).	971130