

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

# MITSUBISHI MICROCOMPUTERS M37274MA-XXXSP

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

## DESCRIPTION

The M37274MA-XXXSP is a single-chip microcomputer designed with CMOS silicon gate technology. It is housed in a 52-pin shrink plastic molded DIP.

In addition to their simple instruction sets, the ROM, RAM and I/O addresses are placed on the same memory map to enable easy programming.

The M37274MA-XXXSP has a OSD function and a data slicer function, so it is useful for a channel selection system for TV with a closed caption decoder.

## FEATURES

- Number of basic instructions ..... 71
- Memory size
  - ROM ..... 40 K bytes
  - RAM ..... 768 bytes
  - ROM correction memory ..... 64 bytes
  - ROM for OSD ..... 11072 bytes
  - RAM for OSD ..... 1296 bytes
- Minimum instruction execution time
  - ..... 0.5  $\mu$ s (at 8 MHz oscillation frequency)
- Power source voltage ..... 5 V  $\pm$  10 %
- Subroutine nesting ..... 128 levels (Max.)
- Interrupts ..... 18 types, 16 vectors
- 8-bit timers ..... 6
- Programmable I/O ports (Ports P0, P1, P2, P30, P31) ..... 26
- Input ports (Ports P40-P46, P63, P64, P70-P72) ..... 12
- Output ports (Ports P52-P55) ..... 4
- 12 V withstand ports ..... 7
- LED drive ports ..... 2
- Serial I/O ..... 8-bit X 1 channel
- Multi-master I<sup>2</sup>C-BUS interface ..... 1 (2 systems)
- A-D converter (8-bit resolution) ..... 4 channels
- PWM output circuit ..... 14-bit X 1, 8-bit X 7
- Power dissipation
  - In high-speed mode ..... 165mW  
 (at V<sub>CC</sub> = 5.5V, 8MHz oscillation frequency, CRT on, and Data slicer on)
  - In low-speed mode ..... 0.33mW  
 (at V<sub>CC</sub> = 5.5V, 32kHz oscillation frequency)
- Data slicer
- ROM correction function

## ● OSD function

Display characters ..... 36 characters X 12 lines  
 Kinds of characters ..... 256 kinds  
 (In EXOSD mode, they can be combined with 16 kinds of extra fonts)

Character display area ..... CC mode : 16 X 26 dots  
 OSD mode : 16 X 20 dots  
 EXOSD mode : 16 X 26 dots

Kinds of character sizes ..... CC mode : 2 types  
 OSD mode : 14 types  
 EXOSD mode : 6 types

It can be specified by a character unit (maximum 7 kinds).

Character font coloring, character background coloring

It can be specified by a screen unit (maximum 7 kinds).

Extra font coloring, raster coloring, border coloring

Kinds of character colors ..... CC mode : 7 kinds (R, G, B)  
 OSD mode : 7 kinds (R, G, B)  
 EXOSD mode : 5 kinds (R, G, B)

## Display position

Horizontal ..... 256 levels

Vertical ..... 1024 levels

Attribute ..... CC mode : smooth italic, underline, flash  
 OSD mode : border  
 EXOSD mode : border,  
 extra font (16 kinds)

Automatic solid space function

Window function

Dual layer OSD function

## APPLICATION

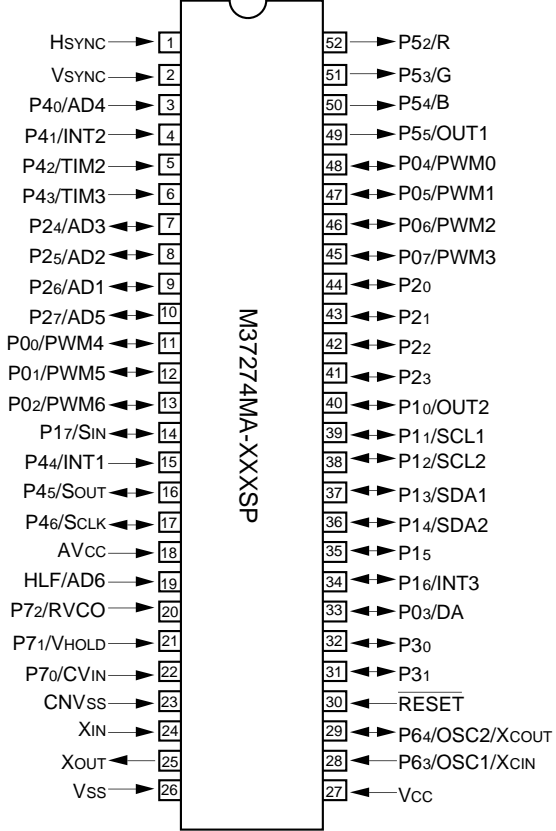
TV with a closed caption decoder

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**PIN CONFIGURATION (TOP VIEW)**

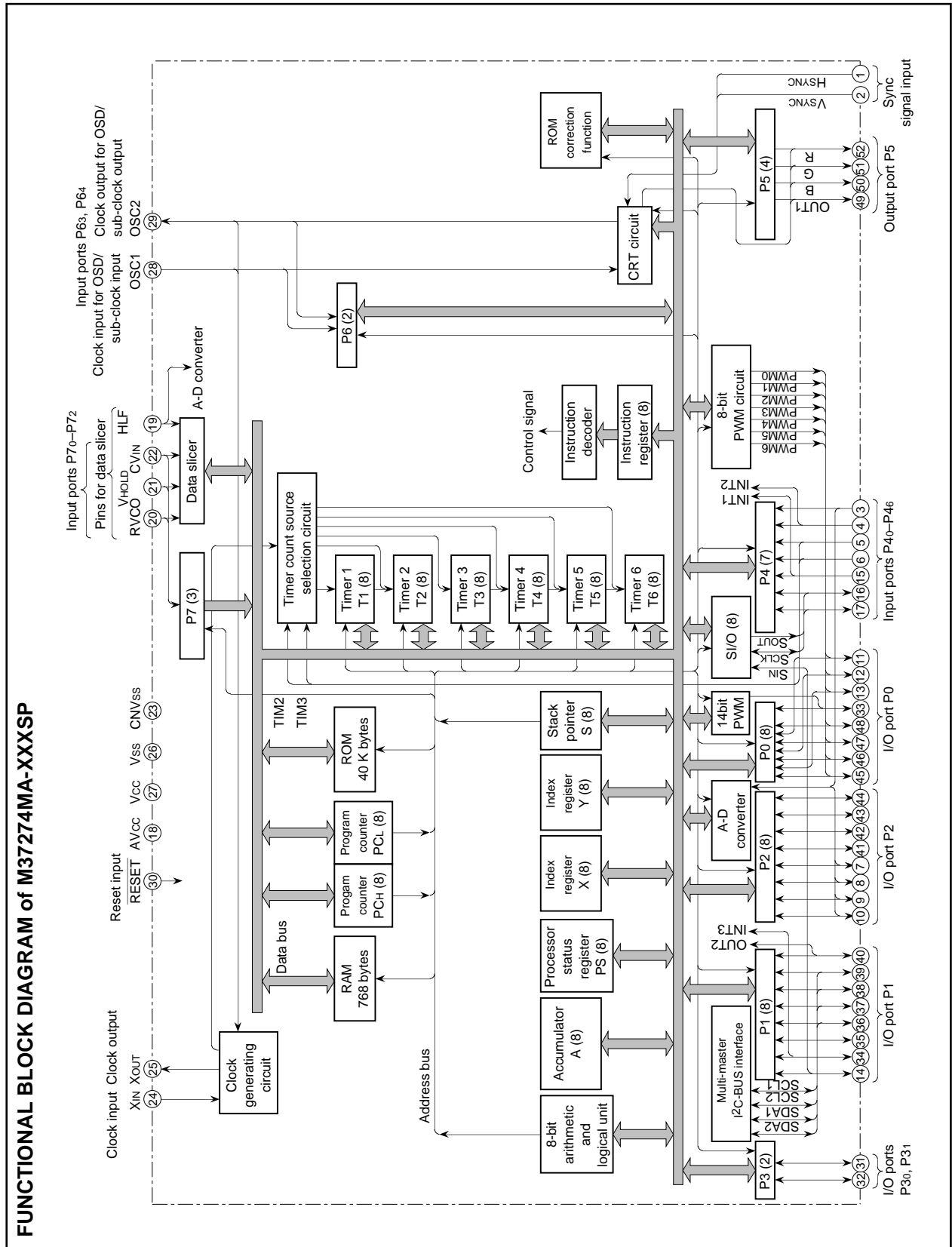


**Outline 52P4B**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**FUNCTIONS**

Parameter		Functions	
Number of basic instructions		71	
Instruction execution time		0.5 $\mu$ s (the minimum instruction execution time, at 8 MHz oscillation frequency)	
Clock frequency		8 MHz (maximum)	
Memory size	ROM	40 K bytes	
	RAM	768 bytes	
	ROM correction memory	64 bytes	
	OSD ROM	11072 bytes	
	OSD RAM	1296 bytes	
Input/Output ports	P00–P02, P04–P07	I/O	7-bit X 1 (N-channel open-drain output structure, can be used as 8-bit PWM output pins)
	P03	I/O	1-bit X 1 (CMOS input/output structure, can be used as 14-bit PWM output pin)
	P10, P15–P17	I/O	4-bit X 1 (CMOS input/output structure, can be used as OSD output pin, INT input pin, serial input pin)
	P11–P14	I/O	4-bit X 1 (N-channel open-drain output structure, can be used as multi-master I <sup>2</sup> C-BUS interface)
	P2	I/O	8-bit X 1 (CMOS input/output structure, can be used as A-D input pins)
	P30, P31	I/O	2-bit X 1 (CMOS input/output structure)
	P40–P44	Input	5-bit X 1 (can be used as A-D input pins, INT input pins, external clock input pins)
	P45, P46	Input	2-bit X 1 (N-channel open-drain output structure when serial I/O is used, can be used as serial I/O pins)
	P52–P55	Output	4-bit X 1 (CMOS output structure, can be used as OSD output)
	P63	Input	1-bit X 1 (can be used as sub-clock input pin, OSD clock input pin)
	P64	Input	1-bit X 1 (CMOS output structure when LC is oscillating, can be used as sub-clock output pin, OSD clock output pin)
	P70–P72	Input	3-bit X 1 (can be used as data slicer input/output)
Serial I/O		8-bit X 1	
Multi-master I <sup>2</sup> C-BUS interface		1	
A-D converter		6 channels (8-bit resolution)	
PWM output circuit		14-bit X 1, 8-bit X 7	
Timers		8-bit timer X 6	
Subroutine nesting		128 levels (maximum)	
Interrupt		External interrupt X 3, Internal timer interrupt X 6, Serial I/O interrupt X 1, OSD interrupt X 1, Multi-master I <sup>2</sup> C-BUS interface interrupt X 1, Data slicer interrupt X 1, f(XIN)/4092 interrupt X 1, VSYNC interrupt X 1, A-D conversion interrupt X 1, BRK instruction interrupt X 1	
Clock generating circuit		2 built-in circuits (externally connected to a ceramic resonator or a quartz-crystal oscillator)	
Data slicer		Built in	

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**FUNCTIONS (continued)**

Parameter		Functions		
OSD function	Number of display characters	36 characters X 12 lines		
	Character display area	CC mode: 16 X 26 dots (dot structure: 16 X 20 dots) OSD mode: 16 X 20 dots EXOSD mode: 16 X 26 dots		
	Kinds of characters	256 kinds (In EXOSDmode, they can be combined with 16 kinds of extra fonts)		
	Kinds of character sizes	CC mode: 2 kinds OSD mode: 14 kinds EXOSD mode: 6 kinds		
	Kinds of character colors	CC mode: 7 kinds (R, G, B) OSD mode: 7 kinds (R, G, B) EXOSD mode: 5 kinds (R, G, B)		
	Display position (horizontal, vertical)	256 levels (horizontal) X 1024 levels (vertical)		
Power source voltage		5 V ± 10 %		
Power dissipation	In high-speed mode	OSD ON	Data slicer ON	165 mW typ. (at oscillation frequency f(X <sub>IN</sub> ) = 8 MHz, fosc = 13 MHz)
		OSD OFF	Data slicer OFF	82.5 mW typ. (at oscillation frequency f(X <sub>IN</sub> ) = 8 MHz)
	In low-speed mode	OSD OFF	Data slicer OFF	0.33mW typ. (at oscillation frequency f(X <sub>CIN</sub> ) = 32 kHz, f(X <sub>IN</sub> ) = stopped)
		In stop mode		0.055 mW (maximum)
Operating temperature range		-10 °C to 70 °C		
Device structure		CMOS silicon gate process		
Package		52-pin shrink plastic molded DIP		

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**PIN DESCRIPTION**

Pin	Name	Input/ Output	Functions
Vcc, AVcc, Vss	Power source		Apply voltage of 5 V ± 10 % (typical) to Vcc and AVcc, and 0 V to Vss.
CNVss	CNVss		Connected to Vss.
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 Vcc conditions). If more time is needed for the quartz-crystal oscillator to stabilize, this "L" condition should be maintained for the required time.
XIN	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 XIN and XOUT. If an external clock is used, the clock source should be connected to the XIN pin and the XOUT pin should be left open.
XOUT	Clock output	Output	
P00/PWM4– P02/PWM6, P03/DA, P04/PWM0– P07/PWM3	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 of P03 is CMOS output, that of P00–P02 and P04–P07 are N-channel open-drain output. See notes at end of Table for full details of port P0 functions.
	DA output	Output	Pin P03 is also used as 14-bit PWM output pin DA. The output structure is CMOS output.
	8-bit PWM output	Output	Pins P00–P02 and P04–P07 are also used as PWM output pins PWM4–PWM6 and PWM0–PWM3 respectively. The output structure is N-channel open-drain output.
P10/OUT2, P11/SCL1, P12/SCL2, P13/SDA1, P14/SDA2, P15, P16/INT3, P17/SIN	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 of P10 and P15–P17 is CMOS output, that of P11–P14 is N-channel open-drain output.
	OSD output	Output	Pin P10 is also used as OSD output pin OUT2. The output structure is CMOS output.
	Multi-master I <sup>2</sup> C-BUS interface	Output	Pin P11 is used as SCL1, SCL2, SDA1 and SDA2 respectively, when multi-master I <sup>2</sup> C-BUS interface is used. The output structure is N-channel open-drain output.
	External interrupt input	Input	Pin P16 is also used as external interrupt input pin INT3.
P20–P23 P24/AD3– P26/AD1, P27/AD5	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.
	Analog input	Input	Pins P24–P26, P27 are also used as analog input pins AD3–AD1, AD5 respectively.
P30, P31	I/O port P3	I/O	Ports P30 and P31 are 2-bit I/O ports and have basically the same functions as port P0. The output structure is CMOS output.
P40/AD4, P41/INT2, P42/TIM2, P43/TIM3, P44/INT1, P45/SOUT, P46/SCLK	Input port P4	Input	Ports P40–P46 are a 7-bit input port.
	Analog input	Input	Pin P40 is also used as analog input pin AD4.
	External interrupt input	Input	Pins P41, P44 are also used as external interrupt input pins INT2, INT1.
	External clock input	Input	Pins P42 and P43 are also used as external clock input pins TIM2, TIM3 respectively.
	Serial I/O data output	Output	Pin P45 is used as serial I/O data output pin SOUT. The output structure is N-channel open-drain output.
	Serial I/O synchronous clock input/output	I/O	Pin P46 is used as serial I/O synchronous clock input/output pin SCLK. The output structure is N-channel open-drain output.
P52/R,P53/G, P54/B, P55/OUT1	Output port P5	Output	Ports P52–P55 are 4-bit output ports. The output structure is CMOS output.
	OSD output	Output	Pins P52–P55 are also used as OSD output pins R, G, B, OUT1 respectively.

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**PIN DESCRIPTION (continued)**

Pin	Name	Input/Output	Functions
P63/OSC1/ XCIN, P64/OSC2/ XCOUT	Input port	Input	Ports P63 and P64 are 2-bit input port.
	Clock input for OSD	Input	Pin P63 is also used as OSD clock input pin OSC1.
	Clock output for OSD	Output	Pin P64 is also used as OSD clock output pin OSC2. The output structure is CMOS output.
	Sub-clock output	Output	Pin P64 is also used as sub-clock output pin XCOUT. The output structure is CMOS output.
	Sub-clock input	Input	Pin P63 is also used as sub-clock input pin XCIN.
P70/CVIN, P71/VHOLD, P72/RVCO	Input port P7	Input	Ports P70–P72 are 3-bit input port.
	Input for data slicer	Input	Pins P70, P71 are also used as data slicer input pins CVIN, VHOLD respectively. When using data slicer, input composite video signal through a capacitor. Connect a capacitor between VHOLD and VSS.
	Input/output for data slicer	I/O	Pins P72 pin is also used as input/output pin for data slicer RVCO. When using data slicer, connect a resistor between RVCO and VSS.
HLF/AD6			When using data slicer, connect a filter using of a capacitor and a resistor between HLF and VSS.
	Analog input	Input	This is an analog input pin AD6.
HSYNC	HSYNC input	Input	This is a horizontal synchronous signal input for OSD.
VSYSN	VSYSN input	Input	This is a vertical synchronous signal input for OSD.

**Note :** As shown in the memory map (Figure 5), port P0 is accessed as a memory at address 00C0<sub>16</sub> of zero page. Port P0 has the port P0 direction register (address 00C1<sub>16</sub> of zero page) which can be used to program each bit as an input ("0") or an output ("1"). The pins programmed as "1" in the direction register are output pins. When pins are programmed as "0," they are input pins. When pins are programmed as output pins, the output data are written into the port latch and then output. When data is read from the output pins, the output pin level is not read but the data of the port latch is read. This allows a previously-output value to be read correctly even if the output "L" 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 data is written into the input pin, it is written only into the port latch, while the pin remains in the floating state.

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

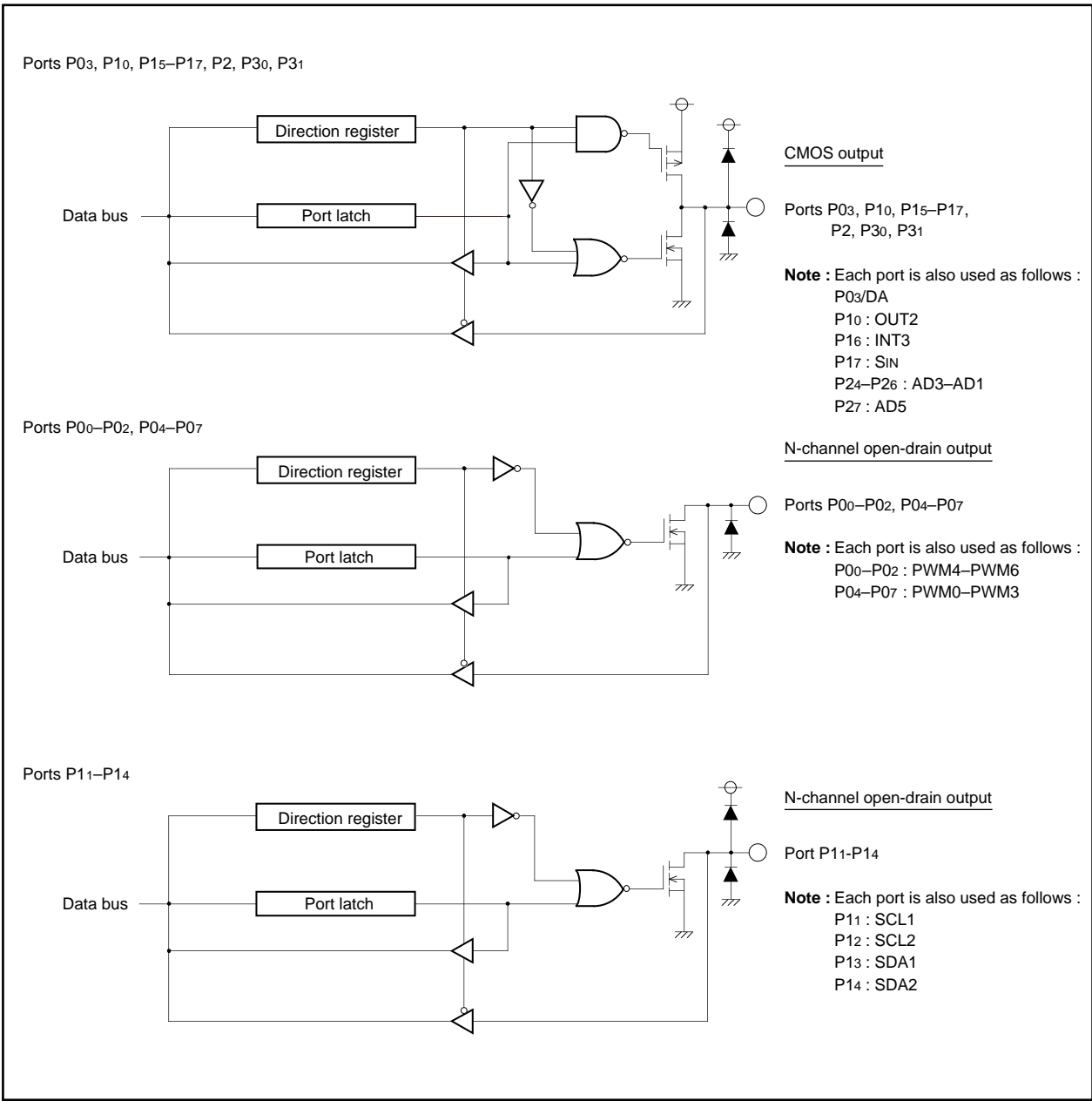


Fig. 1. I/O Pin Block Diagram (1)



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

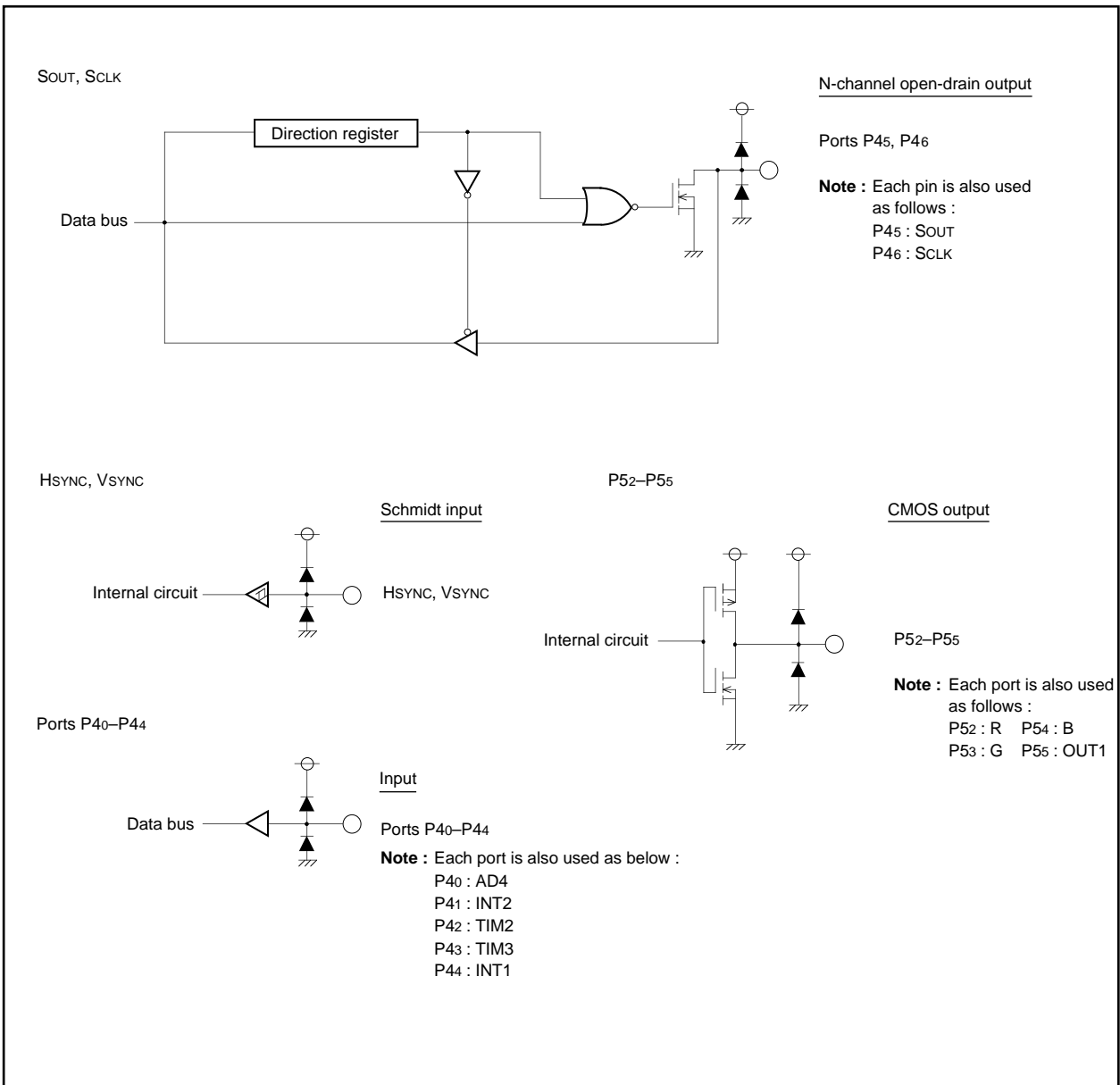


Fig. 2. I/O Pin Block Diagram (2)

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**FUNCTIONAL DESCRIPTION**

**Central Processing Unit (CPU)**

The M37274MA-XXXSP uses the standard 740 Family instruction set. Refer to the table of 740 Family addressing modes and machine instructions or the SERIES 740 <Software> User's Manual for details on the instruction set.

Machine-resident 740 Family instructions are as follows:

The FST, SLW instruction cannot be used.

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

**CPU Mode Register**

The CPU mode register contains the stack page selection bit and internal system clock selection bit. The CPU mode register is allocated at address 00FB16.

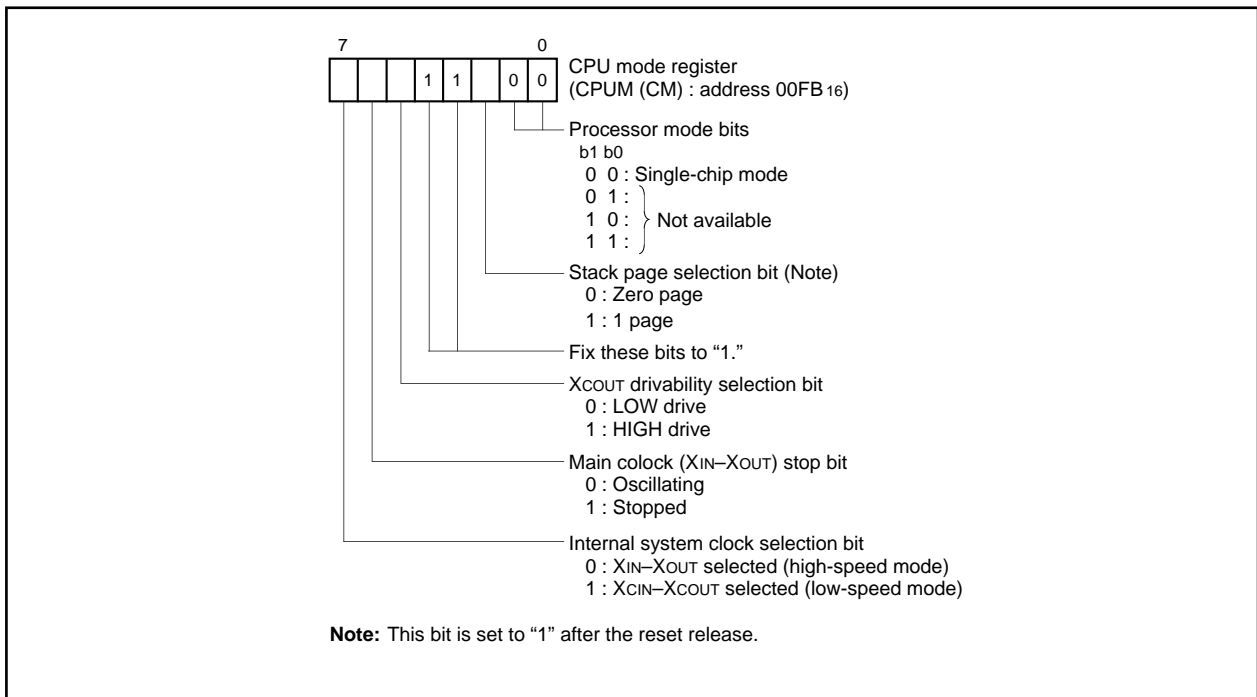


Fig. 3. CPU Mode Register

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MEMORY**

**Special Function Register (SFR) Area**

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

**RAM**

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

**ROM**

ROM is used for storing user programs as well as the interrupt vector area.

**RAM for OSD**

RAM for display is used for specifying the character codes and colors to display.

**ROM for OSD**

ROM for display is used for storing character data.

**Interrupt Vector Area**

The interrupt vector area contains reset and interrupt vectors.

**Zero Page**

The 256 bytes from addresses 0000<sub>16</sub> to 00FF<sub>16</sub> are called the zero page area. The internal RAM and the special function registers (SFR) are allocated to this area.

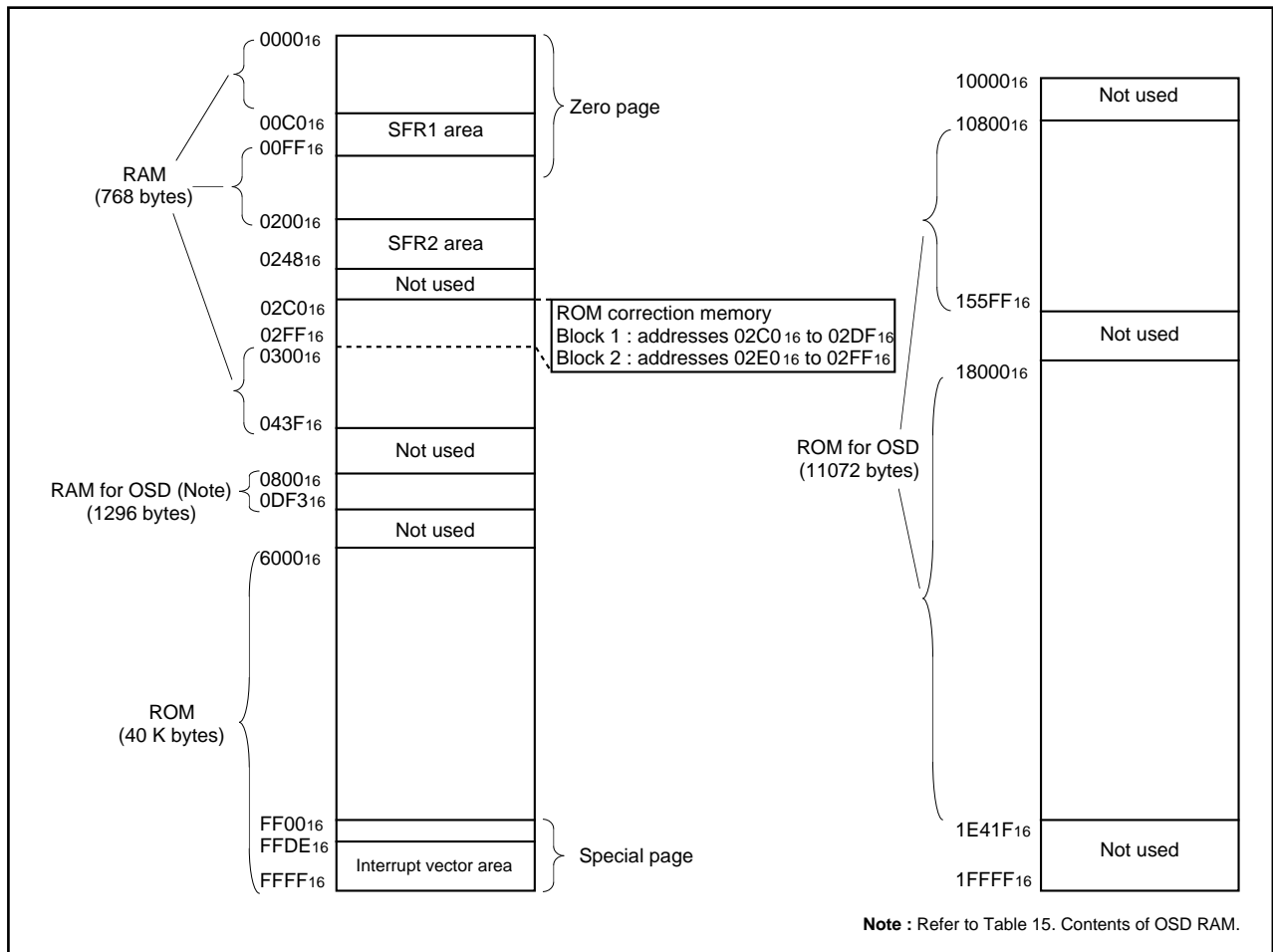
The zero page addressing mode can be used to specify memory and register addresses in the zero page area. Access to this area with only 2 bytes is possible in the zero page addressing mode.

**Special Page**

The 256 bytes from addresses FF00<sub>16</sub> to FFFF<sub>16</sub> are called the special page area. The special page addressing mode can be used to specify memory addresses in the special page area. Access to this area with only 2 bytes is possible in the special page addressing mode.

**ROM Correction Memory (RAM)**

This is used as the program area for ROM correction.



Note : Refer to Table 15. Contents of OSD RAM.

Fig. 4. Memory map

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**■SFR1 area (addresses C0<sub>16</sub>to DF<sub>16</sub>)**

**< Bit allocation >**

: Function bit  
 Name :  
 : No function bit  
 : Fix to this bit to "0"  
 (do not write to "1")  
 : Fix to this bit to "1"  
 (do not write to "0")

**< State immediately after reset >**

: "0" immediately after reset  
 : "1" immediately after reset  
 : Indeterminate immediately after reset

Address	Register	Bit allocation								State immediately after reset								
		b7							b0	b7							b0	
C0 <sub>16</sub>	Port P0 (P0)																?	
C1 <sub>16</sub>	Port P0 direction register (D0)																00 <sub>16</sub>	
C2 <sub>16</sub>	Port P1 (P1)																?	
C3 <sub>16</sub>	Port P1 direction register (D1)																00 <sub>16</sub>	
C4 <sub>16</sub>	Port P2 (P2)																?	
C5 <sub>16</sub>	Port P2 direction register (D2)																00 <sub>16</sub>	
C6 <sub>16</sub>	Port P3 (P3)																?	
C7 <sub>16</sub>	Port P3 direction register (D3)		T3SC														00 <sub>16</sub>	
C8 <sub>16</sub>	Port P4 (P4)																?	
C9 <sub>16</sub>	Port P4 direction register (D4)								0								00 <sub>16</sub>	
CA <sub>16</sub>	Port P5 (P5)																?	
CB <sub>16</sub>	OSD port control register (PF)	0	OUT2	OUT1	B	G	R	0	0								00 <sub>16</sub>	
CC <sub>16</sub>	Port P6 (P6)																?	
CD <sub>16</sub>	Port P7 (P7)									0	0	0	0	0	0	?	?	?
CE <sub>16</sub>	OSD control register (OC)	OC7	OC6	OC5	OC4	OC3	OC2	OC1	OC0								00 <sub>16</sub>	
CF <sub>16</sub>	Horizontal position register (HP)	HP7	HP6	HP5	HP4	HP3	HP2	HP1	HP0								00 <sub>16</sub>	
D0 <sub>16</sub>	Block control register 1 (BC <sub>1</sub> )	BC <sub>1,8</sub>	BC <sub>1,7</sub>	BC <sub>1,6</sub>	BC <sub>1,5</sub>	BC <sub>1,4</sub>	BC <sub>1,3</sub>	BC <sub>1,2</sub>	BC <sub>1,1</sub>								?	
D1 <sub>16</sub>	Block control register 2 (BC <sub>2</sub> )	BC <sub>2,8</sub>	BC <sub>2,7</sub>	BC <sub>2,6</sub>	BC <sub>2,5</sub>	BC <sub>2,4</sub>	BC <sub>2,3</sub>	BC <sub>2,2</sub>	BC <sub>2,1</sub>								?	
D2 <sub>16</sub>	Block control register 3 (BC <sub>3</sub> )	BC <sub>3,8</sub>	BC <sub>3,7</sub>	BC <sub>3,6</sub>	BC <sub>3,5</sub>	BC <sub>3,4</sub>	BC <sub>3,3</sub>	BC <sub>3,2</sub>	BC <sub>3,1</sub>								?	
D3 <sub>16</sub>	Block control register 4 (BC <sub>4</sub> )	BC <sub>4,8</sub>	BC <sub>4,7</sub>	BC <sub>4,6</sub>	BC <sub>4,5</sub>	BC <sub>4,4</sub>	BC <sub>4,3</sub>	BC <sub>4,2</sub>	BC <sub>4,1</sub>								?	
D4 <sub>16</sub>	Block control register 5 (BC <sub>5</sub> )	BC <sub>5,8</sub>	BC <sub>5,7</sub>	BC <sub>5,6</sub>	BC <sub>5,5</sub>	BC <sub>5,4</sub>	BC <sub>5,3</sub>	BC <sub>5,2</sub>	BC <sub>5,1</sub>								?	
D5 <sub>16</sub>	Block control register 6 (BC <sub>6</sub> )	BC <sub>6,8</sub>	BC <sub>6,7</sub>	BC <sub>6,6</sub>	BC <sub>6,5</sub>	BC <sub>6,4</sub>	BC <sub>6,3</sub>	BC <sub>6,2</sub>	BC <sub>6,1</sub>								?	
D6 <sub>16</sub>	Block control register 7 (BC <sub>7</sub> )	BC <sub>7,8</sub>	BC <sub>7,7</sub>	BC <sub>7,6</sub>	BC <sub>7,5</sub>	BC <sub>7,4</sub>	BC <sub>7,3</sub>	BC <sub>7,2</sub>	BC <sub>7,1</sub>								?	
D7 <sub>16</sub>	Block control register 8 (BC <sub>8</sub> )	BC <sub>8,8</sub>	BC <sub>8,7</sub>	BC <sub>8,6</sub>	BC <sub>8,5</sub>	BC <sub>8,4</sub>	BC <sub>8,3</sub>	BC <sub>8,2</sub>	BC <sub>8,1</sub>								?	
D8 <sub>16</sub>	Block control register 9 (BC <sub>9</sub> )	BC <sub>9,8</sub>	BC <sub>9,7</sub>	BC <sub>9,6</sub>	BC <sub>9,5</sub>	BC <sub>9,4</sub>	BC <sub>9,3</sub>	BC <sub>9,2</sub>	BC <sub>9,1</sub>								?	
D9 <sub>16</sub>	Block control register 10 (BC <sub>10</sub> )	BC <sub>10,8</sub>	BC <sub>10,7</sub>	BC <sub>10,6</sub>	BC <sub>10,5</sub>	BC <sub>10,4</sub>	BC <sub>10,3</sub>	BC <sub>10,2</sub>	BC <sub>10,1</sub>								?	
DA <sub>16</sub>	Block control register 11 (BC <sub>11</sub> )	BC <sub>11,8</sub>	BC <sub>11,7</sub>	BC <sub>11,6</sub>	BC <sub>11,5</sub>	BC <sub>11,4</sub>	BC <sub>11,3</sub>	BC <sub>11,2</sub>	BC <sub>11,1</sub>								?	
DB <sub>16</sub>	Block control register 12 (BC <sub>12</sub> )	BC <sub>12,8</sub>	BC <sub>12,7</sub>	BC <sub>12,6</sub>	BC <sub>12,5</sub>	BC <sub>12,4</sub>	BC <sub>12,3</sub>	BC <sub>12,2</sub>	BC <sub>12,1</sub>								?	
DC <sub>16</sub>																	?	
DD <sub>16</sub>																	?	
DE <sub>16</sub>																	?	
DF <sub>16</sub>																	?	

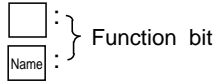
Fig. 5. Memory Map of Special Function Register 1 (SFR1) (1)



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

■ SFR1 area (addresses E0<sub>16</sub> to FF<sub>16</sub>)

< Bit allocation >



■ : No function bit

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

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

< State immediately after reset >

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately after reset

Address	Register	Bit allocation								State immediately after reset								
		b7							b0	b7							b0	
E0 <sub>16</sub>	Caption position register (CP)	1	0	0	CP4	CP3	CP2	CP1	CP0									00 <sub>16</sub>
E1 <sub>16</sub>	Start bit position register (SP)	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0									00 <sub>16</sub>
E2 <sub>16</sub>	Window register (WN)	0	0	WN5	WN4	WN3	WN2	WN1	WN0									00 <sub>16</sub>
E3 <sub>16</sub>	Sync slice register (SSL)	SSL7	0	0	0	0	1	0	1									00 <sub>16</sub>
E4 <sub>16</sub>	Data register 1 (CD1)																	00 <sub>16</sub>
E5 <sub>16</sub>	Data register 2 (CD2)																	00 <sub>16</sub>
E6 <sub>16</sub>	Clock run-in register 1 (CR1)	0	1	0	1	CR13	CR12	CR11	CR10									00 <sub>16</sub>
E7 <sub>16</sub>	Clock run-in register 2 (CR2)	1	0	0	1	1	1	CR21	1									00 <sub>16</sub>
E8 <sub>16</sub>	Clock run-in detect register 1 (CRD1)	CRD17	CRD15	CRD15	CRD15	CRD15												00 <sub>16</sub>
E9 <sub>16</sub>	Clock run-in detect register 2 (CRD2)	CRD27	CRD25	CRD25	CRD25	CRD25	CRD22	CRD21	CRD20									00 <sub>16</sub>
EA <sub>16</sub>	Data slicer control register 1 (DSC1)	DSC17	0	DSC15	0	0	DSC12	DSC11	DSC10	?	0	?	0	0	0	0	0	
EB <sub>16</sub>	Data slicer control register 2 (DSC2)	DSC27	0	DSC25	0	0	DSC22	DSC21	DSC20	?	0	?	0	0	?	0	0	
EC <sub>16</sub>	Data register 3 (CD3)																	00 <sub>16</sub>
ED <sub>16</sub>	Data register 4 (CD4)																	00 <sub>16</sub>
EE <sub>16</sub>	A-D conversion register (AD)																	?
EF <sub>16</sub>	A-D control register (ADCON)	0			ADVREF	ADSTR	ADIN2	ADIN1	ADIN0	0	?	0	0	1	0	0	0	
F0 <sub>16</sub>	Timer 1 (TM1)																	FF <sub>16</sub>
F1 <sub>16</sub>	Timer 2 (TM2)																	07 <sub>16</sub>
F2 <sub>16</sub>	Timer 3 (TM3)																	FF <sub>16</sub>
F3 <sub>16</sub>	Timer 4 (TM4)																	07 <sub>16</sub>
F4 <sub>16</sub>	Timer mode register 1 (TM1)	TM17	TM16	TM15	TM14	TM13	TM12	TM11	TM10									00 <sub>16</sub>
F5 <sub>16</sub>	Timer mode register 2 (TM2)	TM27	TM26	TM25	TM24	TM23	TM22	TM21	TM20									00 <sub>16</sub>
F6 <sub>16</sub>	I <sup>2</sup> C data shift register (S0)																	?
F7 <sub>16</sub>	I <sup>2</sup> C address register (S0D)	SAD6	SAD5	SAD4	SAD3	SAD2	SAD1	SAD0	RBW									00 <sub>16</sub>
F8 <sub>16</sub>	I <sup>2</sup> C status register (S1)	MST	TRX	BB	PIN	AL	AAS	ADO	LRB	0	0	0	1	0	0	0	?	
F9 <sub>16</sub>	I <sup>2</sup> C control register (S1D)	BSEL1	BSEL0	<sup>10</sup> BIT SAD	ALS	ES0	BC2	BC1	BC0									00 <sub>16</sub>
FA <sub>16</sub>	I <sup>2</sup> C clock control register (S2)	ACK BIT	FAST MODE	CCR4	CCR3	CCR2	CCR1	CCR0										00 <sub>16</sub>
FB <sub>16</sub>	CPU mode register (CPUM)	CM7	CM6	CM5	1	1	CM2	0	0	0	0	1	1	1	1	0	0	
FC <sub>16</sub>	Interrupt request register 1 (IREQ1)		ADR	VSCR	CRTR	TM4R	TM3R	TM2R	TM1R									00 <sub>16</sub>
FD <sub>16</sub>	Interrupt request register 2 (IREQ2)	0	T56R	IICR	INT2R	1MSR	SIOR	DSR	INT1R									00 <sub>16</sub>
FE <sub>16</sub>	Interrupt control register 1 (ICON1)		ADE	VSCR	CRTE	TM4E	TM3E	TM2E	TM1E									00 <sub>16</sub>
FF <sub>16</sub>	Interrupt control register 2 (ICON2)	T56S	T56E	IICE	INT2E	1MSE	SIOE	DSE	INT1E									00 <sub>16</sub>

Fig. 6. Memory Map of Special Function Register 1 (SFR2) (2)

■SFR2 area (addresses 200<sub>16</sub> to 21F<sub>16</sub>)

< Bit allocation >

: } Function bit  
 Name : }

: No function bit

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

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

< State immediately after reset >

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately  
 after reset

Address	Register	Bit allocation								State immediately after reset								
		b7							b0	b7							b0	
200 <sub>16</sub>	PWM0 register (PWM0)																	?
201 <sub>16</sub>	PWM1 register (PWM1)																	?
202 <sub>16</sub>	PWM2 register (PWM2)																	?
203 <sub>16</sub>	PWM3 register (PWM3)																	?
204 <sub>16</sub>	PWM4 register (PWM4)																	?
205 <sub>16</sub>	PWM5 register (PWM5)																	?
206 <sub>16</sub>	PWM6 register (PWM6)																	?
207 <sub>16</sub>																		?
208 <sub>16</sub>	Clock run-in detect register 3 (CRD3)	CRD35	CRD34	CRD33	CRD32	CRD31												00 <sub>16</sub>
209 <sub>16</sub>	Clock run-in register (CR3)		CR36	CR35	CR34	CR33	CR32	CR31	CR30									?
20A <sub>16</sub>	PWM mode register 1 (PN)						PN3	PN2	PN1	PN0								? ? ? ? 0 0 0 0
20B <sub>16</sub>	PWM mode register 2 (PW)	0	PW6	PW5	PW4	PW3	PW2	PW1	PW0									00 <sub>16</sub>
20C <sub>16</sub>	Timer 5 (TM5)																	07 <sub>16</sub>
20D <sub>16</sub>	Timer 6 (TM6)																	FF <sub>16</sub>
20E <sub>16</sub>																		00 <sub>16</sub>
20F <sub>16</sub>	Sync pulse counter register (SYC)				SYC5	SYC4	SYC3	SYC2	SYC1	SYC0								00 <sub>16</sub>
210 <sub>16</sub>	Data slicer control register 3 (DSC3)	DSC37	DSC36	DSC35	DSC34	DSC33	DSC32	DSC31	DSC30									00 <sub>16</sub>
211 <sub>16</sub>																		?
212 <sub>16</sub>	Interrupt input polarity register (IP)	AD/INT3 SEL	INT3 POL	0	INT2 POL	INT1 POL	0	0	0									00 <sub>16</sub>
213 <sub>16</sub>	Serial I/O mode register (SM)	0	0	SM5	SM4	SM3	SM2	SM1	SM0									00 <sub>16</sub>
214 <sub>16</sub>	Serial I/O register (SIO)																	?
215 <sub>16</sub>		0																?
216 <sub>16</sub>	Clock source control register (CS)	0	CS6	CS5	CS4	CS3	CS2	CS1	CS0									00 <sub>16</sub>
217 <sub>16</sub>	I/O polarity control register (PC)	PC7	PC6	PC5	PC4		PC2	PC1	PC0									1 0 0 0 0 0 0 0
218 <sub>16</sub>	Raster color register (RC)	RC7	RC6	RC5			RC2	RC1	RC0									00 <sub>16</sub>
219 <sub>16</sub>	Extra font color register (EC)				0	0	EC2	EC1	EC0									00 <sub>16</sub>
21A <sub>16</sub>					0	0	0	0	0									00 <sub>16</sub>
21B <sub>16</sub>	Border color register (FC)				0	0	FC2	FC1	FC0									00 <sub>16</sub>
21C <sub>16</sub>	Window H register 1 (WH1)	WH17	WH16	WH15	WH14	WH13	WH12	WH11	WH10									?
21D <sub>16</sub>	Window L register 1 (WL1)	WL17	WL16	WL15	WL14	WL13	WL12	WL11	WL10									?
21E <sub>16</sub>	Window H register 2 (WH2)							WH21	WH20									?
21F <sub>16</sub>	Window L register 2 (WL2)							WL21	WL20									?

Fig. 7. Memory Map of Special Function Register 2 (SFR2) (1)

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**■ SFR2 area (addresses 220<sub>16</sub> to 248<sub>16</sub>)**

< Bit allocation >

: } Function bit  
 Name : }

: No function bit

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

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

< State immediately after reset >

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately after reset

Address	Register	Bit allocation								State immediately after reset							
		b7							b0	b7							b0
220 <sub>16</sub>	Vertical position register 1 <sub>1</sub> (VP1 <sub>1</sub> )	VP1 <sub>1,8</sub>	VP1 <sub>1,7</sub>	VP1 <sub>1,6</sub>	VP1 <sub>1,5</sub>	VP1 <sub>1,4</sub>	VP1 <sub>1,3</sub>	VP1 <sub>1,2</sub>	VP1 <sub>1,1</sub>								?
221 <sub>16</sub>	Vertical position register 1 <sub>2</sub> (VP1 <sub>2</sub> )	VP1 <sub>2,8</sub>	VP1 <sub>2,7</sub>	VP1 <sub>2,6</sub>	VP1 <sub>2,5</sub>	VP1 <sub>2,4</sub>	VP1 <sub>2,3</sub>	VP1 <sub>2,2</sub>	VP1 <sub>2,1</sub>								?
222 <sub>16</sub>	Vertical position register 1 <sub>3</sub> (VP1 <sub>3</sub> )	VP1 <sub>3,8</sub>	VP1 <sub>3,7</sub>	VP1 <sub>3,6</sub>	VP1 <sub>3,5</sub>	VP1 <sub>3,4</sub>	VP1 <sub>3,3</sub>	VP1 <sub>3,2</sub>	VP1 <sub>3,1</sub>								?
223 <sub>16</sub>	Vertical position register 1 <sub>4</sub> (VP1 <sub>4</sub> )	VP1 <sub>4,8</sub>	VP1 <sub>4,7</sub>	VP1 <sub>4,6</sub>	VP1 <sub>4,5</sub>	VP1 <sub>4,4</sub>	VP1 <sub>4,3</sub>	VP1 <sub>4,2</sub>	VP1 <sub>4,1</sub>								?
224 <sub>16</sub>	Vertical position register 1 <sub>5</sub> (VP1 <sub>5</sub> )	VP1 <sub>5,8</sub>	VP1 <sub>5,7</sub>	VP1 <sub>5,6</sub>	VP1 <sub>5,5</sub>	VP1 <sub>5,4</sub>	VP1 <sub>5,3</sub>	VP1 <sub>5,2</sub>	VP1 <sub>5,1</sub>								?
225 <sub>16</sub>	Vertical position register 1 <sub>6</sub> (VP1 <sub>6</sub> )	VP1 <sub>6,8</sub>	VP1 <sub>6,7</sub>	VP1 <sub>6,6</sub>	VP1 <sub>6,5</sub>	VP1 <sub>6,4</sub>	VP1 <sub>6,3</sub>	VP1 <sub>6,2</sub>	VP1 <sub>6,1</sub>								?
226 <sub>16</sub>	Vertical position register 1 <sub>7</sub> (VP1 <sub>7</sub> )	VP1 <sub>7,8</sub>	VP1 <sub>7,7</sub>	VP1 <sub>7,6</sub>	VP1 <sub>7,5</sub>	VP1 <sub>7,4</sub>	VP1 <sub>7,3</sub>	VP1 <sub>7,2</sub>	VP1 <sub>7,1</sub>								?
227 <sub>16</sub>	Vertical position register 1 <sub>8</sub> (VP1 <sub>8</sub> )	VP1 <sub>8,8</sub>	VP1 <sub>8,7</sub>	VP1 <sub>8,6</sub>	VP1 <sub>8,5</sub>	VP1 <sub>8,4</sub>	VP1 <sub>8,3</sub>	VP1 <sub>8,2</sub>	VP1 <sub>8,1</sub>								?
228 <sub>16</sub>	Vertical position register 1 <sub>9</sub> (VP1 <sub>9</sub> )	VP1 <sub>9,8</sub>	VP1 <sub>9,7</sub>	VP1 <sub>9,6</sub>	VP1 <sub>9,5</sub>	VP1 <sub>9,4</sub>	VP1 <sub>9,3</sub>	VP1 <sub>9,2</sub>	VP1 <sub>9,1</sub>								?
229 <sub>16</sub>	Vertical position register 1 <sub>10</sub> (VP1 <sub>10</sub> )	VP1 <sub>10,8</sub>	VP1 <sub>10,7</sub>	VP1 <sub>10,6</sub>	VP1 <sub>10,5</sub>	VP1 <sub>10,4</sub>	VP1 <sub>10,3</sub>	VP1 <sub>10,2</sub>	VP1 <sub>10,1</sub>								?
22A <sub>16</sub>	Vertical position register 1 <sub>11</sub> (VP1 <sub>11</sub> )	VP1 <sub>11,8</sub>	VP1 <sub>11,7</sub>	VP1 <sub>11,6</sub>	VP1 <sub>11,5</sub>	VP1 <sub>11,4</sub>	VP1 <sub>11,3</sub>	VP1 <sub>11,2</sub>	VP1 <sub>11,1</sub>								?
22B <sub>16</sub>	Vertical position register 1 <sub>12</sub> (VP1 <sub>12</sub> )	VP1 <sub>12,8</sub>	VP1 <sub>12,7</sub>	VP1 <sub>12,6</sub>	VP1 <sub>12,5</sub>	VP1 <sub>12,4</sub>	VP1 <sub>12,3</sub>	VP1 <sub>12,2</sub>	VP1 <sub>12,1</sub>								?
22C <sub>16</sub>																	?
22D <sub>16</sub>																	?
22E <sub>16</sub>																	?
22F <sub>16</sub>																	?
230 <sub>16</sub>	Vertical position register 2 <sub>1</sub> (VP2 <sub>1</sub> )							VP2 <sub>1,2</sub>	VP2 <sub>1,1</sub>								?
231 <sub>16</sub>	Vertical position register 2 <sub>2</sub> (VP2 <sub>2</sub> )							VP2 <sub>2,2</sub>	VP2 <sub>2,1</sub>								?
232 <sub>16</sub>	Vertical position register 2 <sub>3</sub> (VP2 <sub>3</sub> )							VP2 <sub>3,2</sub>	VP2 <sub>3,1</sub>								?
233 <sub>16</sub>	Vertical position register 2 <sub>4</sub> (VP2 <sub>4</sub> )							VP2 <sub>4,2</sub>	VP2 <sub>4,1</sub>								?
234 <sub>16</sub>	Vertical position register 2 <sub>5</sub> (VP2 <sub>5</sub> )							VP2 <sub>5,2</sub>	VP2 <sub>5,1</sub>								?
235 <sub>16</sub>	Vertical position register 2 <sub>6</sub> (VP2 <sub>6</sub> )							VP2 <sub>6,2</sub>	VP2 <sub>6,1</sub>								?
236 <sub>16</sub>	Vertical position register 2 <sub>7</sub> (VP2 <sub>7</sub> )							VP2 <sub>7,2</sub>	VP2 <sub>7,1</sub>								?
237 <sub>16</sub>	Vertical position register 2 <sub>8</sub> (VP2 <sub>8</sub> )							VP2 <sub>8,2</sub>	VP2 <sub>8,1</sub>								?
238 <sub>16</sub>	Vertical position register 2 <sub>9</sub> (VP2 <sub>9</sub> )							VP2 <sub>9,2</sub>	VP2 <sub>9,1</sub>								?
239 <sub>16</sub>	Vertical position register 2 <sub>10</sub> (VP2 <sub>10</sub> )							VP2 <sub>10,2</sub>	VP2 <sub>10,1</sub>								?
23A <sub>16</sub>	Vertical position register 2 <sub>11</sub> (VP2 <sub>11</sub> )							VP2 <sub>11,2</sub>	VP2 <sub>11,1</sub>								?
23B <sub>16</sub>	Vertical position register 2 <sub>12</sub> (VP2 <sub>12</sub> )							VP2 <sub>12,2</sub>	VP2 <sub>12,1</sub>								?
23C <sub>16</sub>																	?
23D <sub>16</sub>																	?
23E <sub>16</sub>																	?
23F <sub>16</sub>																	?
240 <sub>16</sub>	DA-H register (DA-H)																?
241 <sub>16</sub>	DA-L register (DA-L)									0	0	?	?	?	?	?	?
242 <sub>16</sub>	ROM correction address 1 (high-order)																0016
243 <sub>16</sub>	ROM correction address 1 (low-order)																0016
244 <sub>16</sub>	ROM correction address 2 (high-order)																0016
245 <sub>16</sub>	ROM correction address 2 (low-order)																0016
246 <sub>16</sub>	ROM correction enable register (RCR)							0	0	RCR1	RCR0						0016
247 <sub>16</sub>								0016									0016
248 <sub>16</sub>										0	0	0					0016

Fig. 8. Memory Map of Special Function Register 2 (SFR2) (2)

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

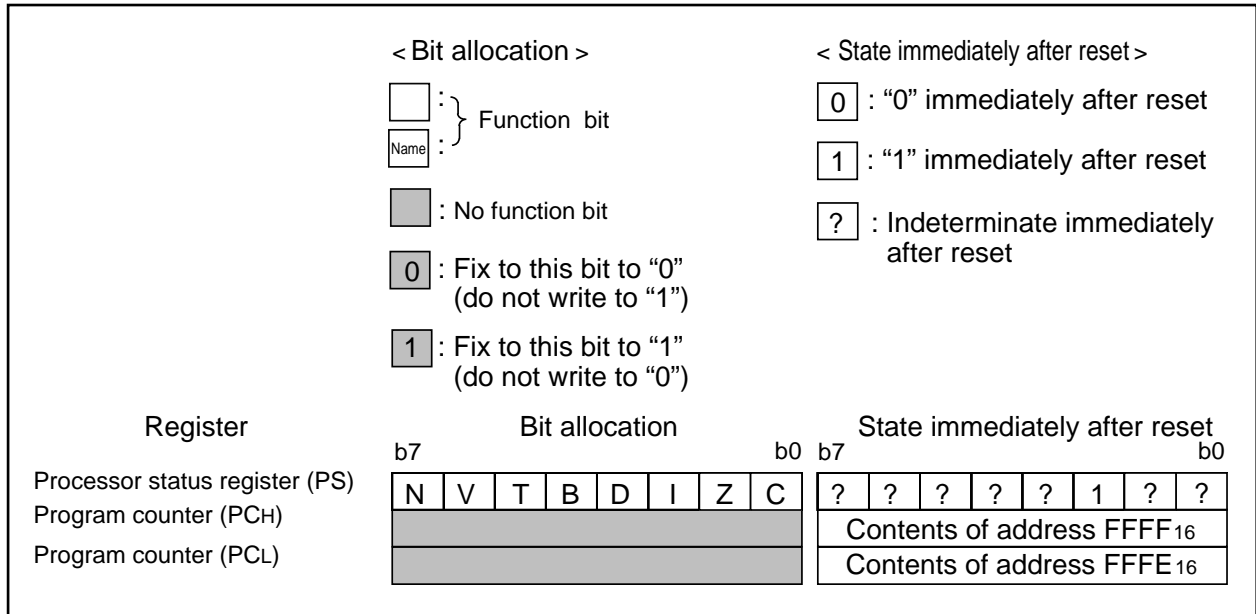


Fig. 9. Internal State of Processor Status Register and Program Counter at Reset



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**INTERRUPTS**

Interrupts can be caused by 18 different sources consisting of 4 external, 12 internal, 1 software, and reset. Interrupts are vectored interrupts with priorities as shown in Table 1. Reset is also included in the table because its operation is similar to an interrupt.

When an interrupt is accepted,

- (1) The contents of the program counter and processor status register are automatically stored into the stack.
- (2) The interrupt disable flag I is set to "1" and the corresponding interrupt request bit is set to "0."
- (3) The jump destination address stored in the vector address enters the program counter.

Other interrupts are disabled when the interrupt disable flag is set to "1."

All interrupts except the BRK instruction interrupt have an interrupt request bit and an interrupt enable bit. The interrupt request bits are in interrupt request registers 1 and 2 and the interrupt enable bits are in interrupt control registers 1 and 2. Figure 11 shows the interrupt-related registers.

Interrupts other than the BRK instruction interrupt and reset are accepted when the interrupt enable bit is "1," interrupt request bit is "1," and the interrupt disable flag is "0." The interrupt request bit can be set to "0" by a program, but not set to "1." The interrupt enable bit can be set to "0" and "1" by a program.

Reset is treated as a non-maskable interrupt with the highest priority. Figure 10 shows interrupt control.

**Interrupt Causes**

- (1) VSYNC and OSD interrupts  
 The VSYNC interrupt is an interrupt request synchronized with the vertical sync signal.  
 The OSD interrupt occurs after character block display to the CRT is completed.
- (2) INT1, INT2, INT3 interrupts  
 With an external interrupt input, the system detects that the level of a pin changes from "L" to "H" or from "H" to "L," and generates an interrupt request. The input active edge can be selected by bits 3, 4 and 6 of the interrupt input polarity register (address 021216) : when this bit is "0," a change from "L" to "H" is detected; when it is "1," a change from "H" to "L" is detected. Note that all bits are cleared to "0" at reset.
- (3) Timer 1, 2, 3 and 4 interrupts  
 An interrupt is generated by an overflow of timer 1, 2, 3 or 4.
- (4) Serial I/O interrupt  
 This is an interrupt request from the clock synchronous serial I/O function.
- (5) f(XIN)/4096 interrupt  
 This interrupt occurs regularly with a f(XIN)/4096 period. Set bit 0 of the PWM mode register 1 to "0."
- (6) Data slicer interrupt  
 An interrupt occurs when slicing data is completed.
- (7) Multi-master I<sup>2</sup>C-BUS interface interrupt  
 This is an interrupt request related to the multi-master I<sup>2</sup>C-BUS interface.
- (8) A-D conversion interrupt  
 An interrupt occurs at the completion of A-D conversion. Since A-D conversion interrupt and the INT3 interrupt share the same vector, an interrupt source is selected by bit 7 of the interrupt interval determination control register (address 021216).

**Table 1. Interrupt Vector Addresses and Priority**

Interrupt Source	Priority	Vector Addresses	Remarks
Reset	1	FFFF <sub>16</sub> , FFFE <sub>16</sub>	Non-maskable
OSD interrupt	2	FFFD <sub>16</sub> , FFFC <sub>16</sub>	
INT1 interrupt	3	FFFB <sub>16</sub> , FFFA <sub>16</sub>	Active edge selectable
Data slicer interrupt	4	FFF9 <sub>16</sub> , FFF8 <sub>16</sub>	
Serial I/O interrupt	5	FFF7 <sub>16</sub> , FFF6 <sub>16</sub>	
Timer 4 interrupt	6	FFF5 <sub>16</sub> , FFF4 <sub>16</sub>	
f(XIN)/4096 interrupt	7	FFF3 <sub>16</sub> , FFF2 <sub>16</sub>	
VSYNC interrupt	8	FFF1 <sub>16</sub> , FFF0 <sub>16</sub>	Active edge selectable
Timer 3 interrupt	9	FFEF <sub>16</sub> , FFEE <sub>16</sub>	
Timer 2 interrupt	10	FFED <sub>16</sub> , FFEC <sub>16</sub>	
Timer 1 interrupt	11	FFEB <sub>16</sub> , FFEA <sub>16</sub>	
A-D conversion · INT3 interrupt	12	FFE9 <sub>16</sub> , FFE8 <sub>16</sub>	Software switch by software (See note)/ When selecting INT3 interrupt, active edge selectable.
INT2 interrupt	13	FFE7 <sub>16</sub> , FFE6 <sub>16</sub>	Active edge selectable
Multi-master I <sup>2</sup> C-BUS interface interrupt	14	FFE5 <sub>16</sub> , FFE4 <sub>16</sub>	
Timer 5 · 6 interrupt	15	FFE3 <sub>16</sub> , FFE2 <sub>16</sub>	Software switch by software (See note)
BRK instruction interrupt	16	FFD <sub>16</sub> , FFDE <sub>16</sub>	Non-maskable (software interrupt)

**Note :** Switching a source during a program causes an unnecessary interrupt occurs. Accordingly, set a source at initializing of program.

**PRELIMINARY**

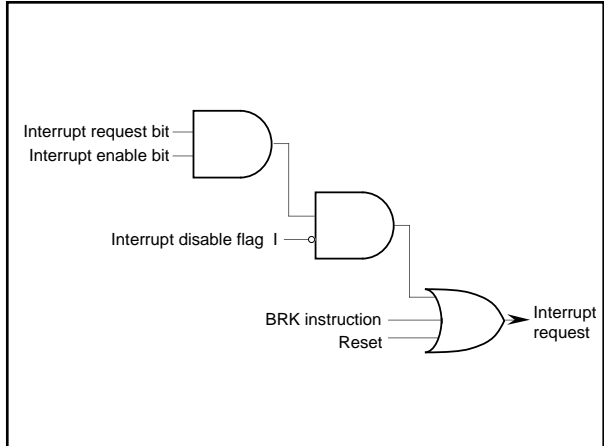
Notice: This is not a final specification.  
Some parametric limits are subject to change.

(9)Timer 5 - 6 interrupt

An interrupt is generated by an overflow of timer 5 or 6. Their priorities are same, and can be switched by software.

(10)BRK instruction interrupt

This software interrupt has the least significant priority. It does not have a corresponding interrupt enable bit, and it is not affected by the interrupt disable flag I (non-maskable).

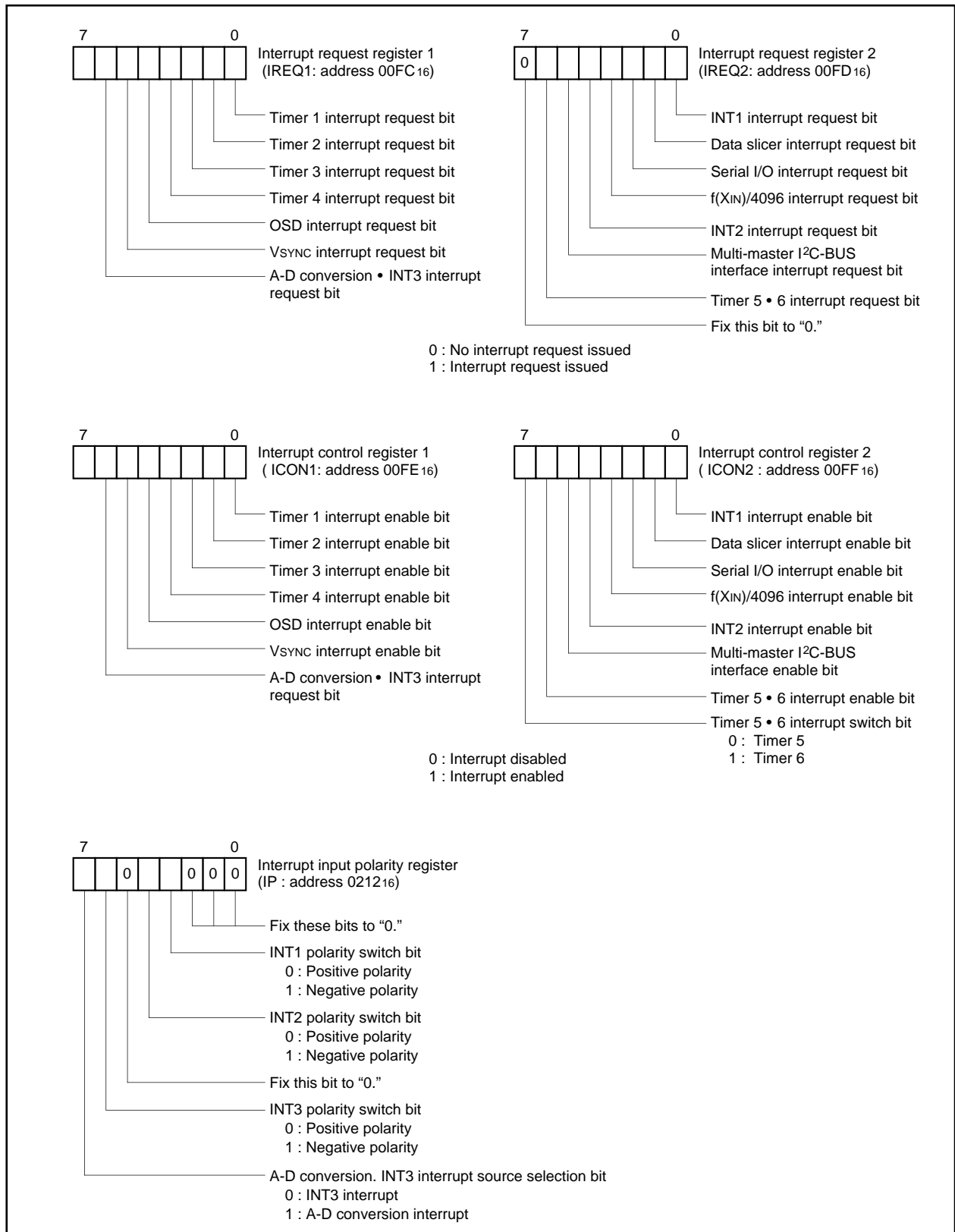


**Fig. 10. Interrupt Control**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER



**Fig. 11. Interrupt-related Registers**

## TIMERS

The M37271MF-XXXSP has 6 timers: timer 1, timer 2, timer 3, timer 4, timer 5, and timer 6. All timers are 8-bit timers with the 8-bit timer latch. The timer block diagram is shown in Figure 13.

All of the timers count down and their divide ratio is  $1/(n+1)$ , where  $n$  is the value of timer latch. By writing a count value to the corresponding timer latch (addresses 00F0<sub>16</sub> to 00F3<sub>16</sub>: timers 1 to 4, addresses 020C<sub>16</sub> and 020D<sub>16</sub>: timers 5 and 6), the value is also set to a timer, simultaneously.

The count value is decremented by 1. The timer interrupt request bit is set to "1" by a timer overflow at the next count pulse, after the count value reaches "00<sub>16</sub>".

### (1) Timer 1

Timer 1 can select one of the following count sources:

- $f(X_{IN})/16$  or  $f(X_{CIN})/16$
- $f(X_{IN})/4096$  or  $f(X_{CIN})/4096$
- External clock from the P42 TIM2 pin

The count source of timer 1 is selected by setting bits 5 and 0 of timer mode register 1 (address 00F4<sub>16</sub>). Either  $f(X_{IN})$  or  $f(X_{CIN})$  is selected by bit 7 of the CPU mode register.

Timer 1 interrupt request occurs at timer 1 overflow.

### (2) Timer 2

Timer 2 can select one of the following count sources:

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

The count source of timer 2 is selected by setting bits 4 and 1 of timer mode register 1 (address 00F4<sub>16</sub>). Either  $f(X_{IN})$  or  $f(X_{CIN})$  is selected by bit 7 of the CPU mode register. When timer 1 overflow signal is a count source for the timer 2, the timer 1 functions as an 8-bit prescaler.

Timer 2 interrupt request occurs at timer 2 overflow.

### (3) Timer 3

Timer 3 can select one of the following count sources:

- $f(X_{IN})/16$  or  $f(X_{CIN})/16$
- $f(X_{CIN})$
- External clock from the TIM3 pin

The count source of timer 3 is selected by setting bit 0 of timer mode register 2 (address 00F5<sub>16</sub>) and bit 6 at address 00C7<sub>16</sub>. Either  $f(X_{IN})$  or  $f(X_{CIN})$  is selected by bit 7 of the CPU mode register.

Timer 3 interrupt request occurs at timer 3 overflow.

### (4) Timer 4

Timer 4 can select one of the following count sources:

- $f(X_{IN})/16$  or  $f(X_{CIN})/16$
- $f(X_{IN})/2$  or  $f(X_{CIN})/2$
- $f(X_{CIN})$

The count source of timer 3 is selected by setting bits 1 and 4 of timer mode register 2 (address 00F5<sub>16</sub>). Either  $f(X_{IN})$  or  $f(X_{CIN})$  is selected by bit 7 of the CPU mode register. When timer 3 overflow signal is a count source for the timer 4, the timer 3 functions as an 8-bit prescaler.

Timer 4 interrupt request occurs at timer 4 overflow.

### (5) Timer 5

Timer 5 can select one of the following count sources:

- $f(X_{IN})/16$  or  $f(X_{CIN})/16$
- Timer 2 overflow signal
- Timer 4 overflow signal

The count source of timer 3 is selected by setting bit 6 of timer mode register 1 (address 00F4<sub>16</sub>) and bit 7 of timer mode register 2 (address 00F5<sub>16</sub>). When overflow of timer 2 or 4 is a count source for timer 5, either timer 2 or 4 functions as an 8-bit prescaler. Either  $f(X_{IN})$  or  $f(X_{CIN})$  is selected by bit 7 of the CPU mode register.

Timer 5 interrupt request occurs at timer 5 overflow.

### (6) Timer 6

Timer 6 can select one of the following count sources:

- $f(X_{IN})/16$  or  $f(X_{CIN})/16$
- Timer 5 overflow signal

The count source of timer 6 is selected by setting bit 7 of timer mode register 1 (address 00F4<sub>16</sub>). Either  $f(X_{IN})$  or  $f(X_{CIN})$  is selected by bit 7 of the CPU mode register. When timer 5 overflow signal is a count source for timer 6, timer 5 functions as an 8-bit prescaler.

Timer 6 interrupt request occurs at timer 6 overflow.

At reset, timers 3 and 4 are connected by hardware and "FF<sub>16</sub>" is automatically set in timer 3; "07<sub>16</sub>" in timer 4. The  $f(X_{IN})^*/16$  is selected as the timer 3 count source. The internal reset is released by timer 4 overflow in this state and the internal clock is connected.

At execution of the STP instruction, timers 3 and 4 are connected by hardware and "FF<sub>16</sub>" is automatically set in timer 3; "07<sub>16</sub>" in timer 4. However, the  $f(X_{IN})^*/16$  is not selected as the timer 3 count source. So set both bit 0 of timer mode register 2 (address 00F5<sub>16</sub>) and bit 6 at address 00C7<sub>16</sub> to "0" before execution of the STP instruction ( $f(X_{IN})^*/16$  is selected as the timer 3 count source). The internal STP state is released by timer 4 overflow in this state and the internal clock is connected.

As a result of the above procedure, the program can start under a stable clock.

\* : When bit 7 of the CPU mode register (CM7) is "1,"  $f(X_{IN})$  becomes  $f(X_{CIN})$ .

The structure of timer-related registers is shown in Figure 12.

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

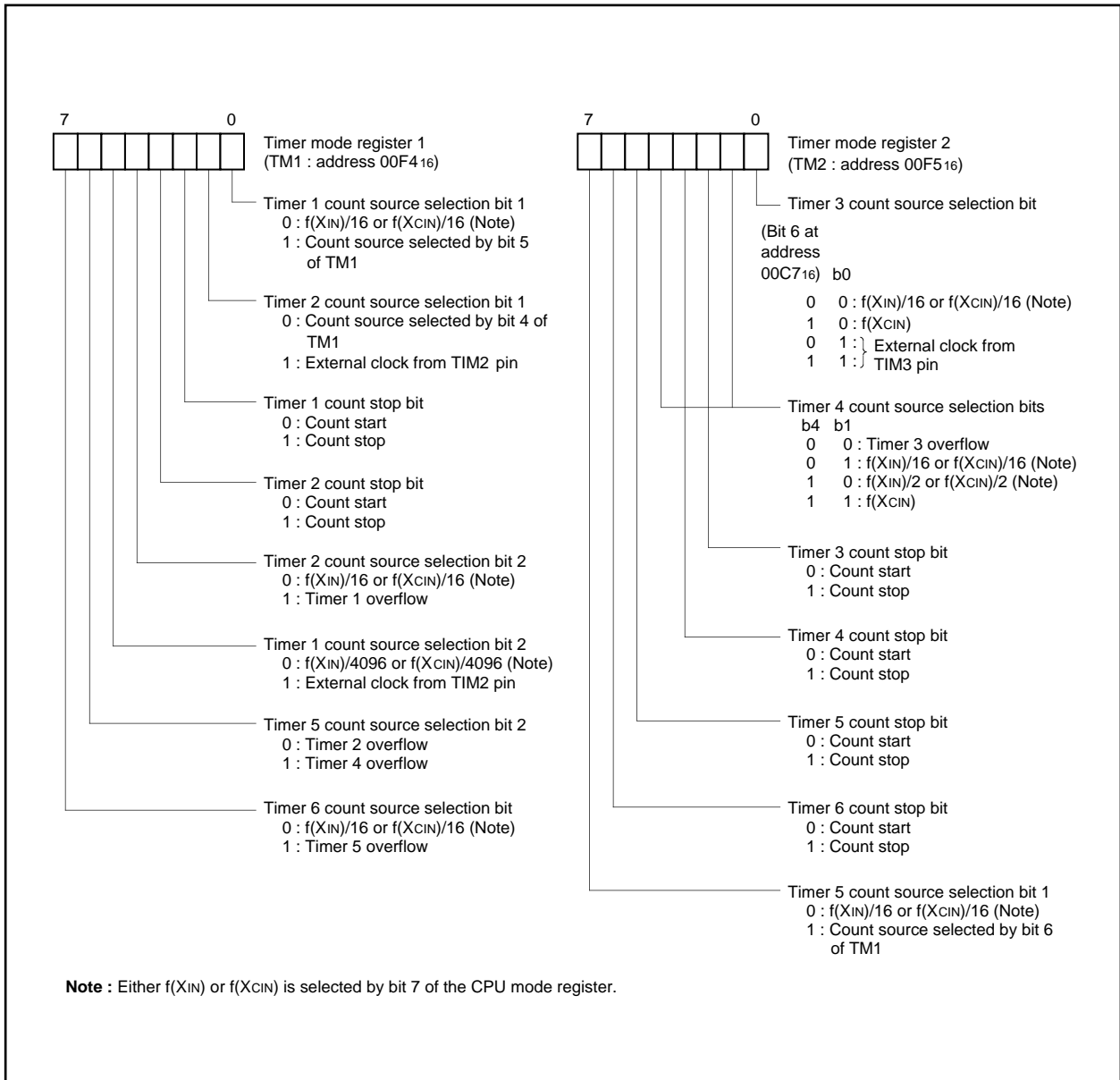
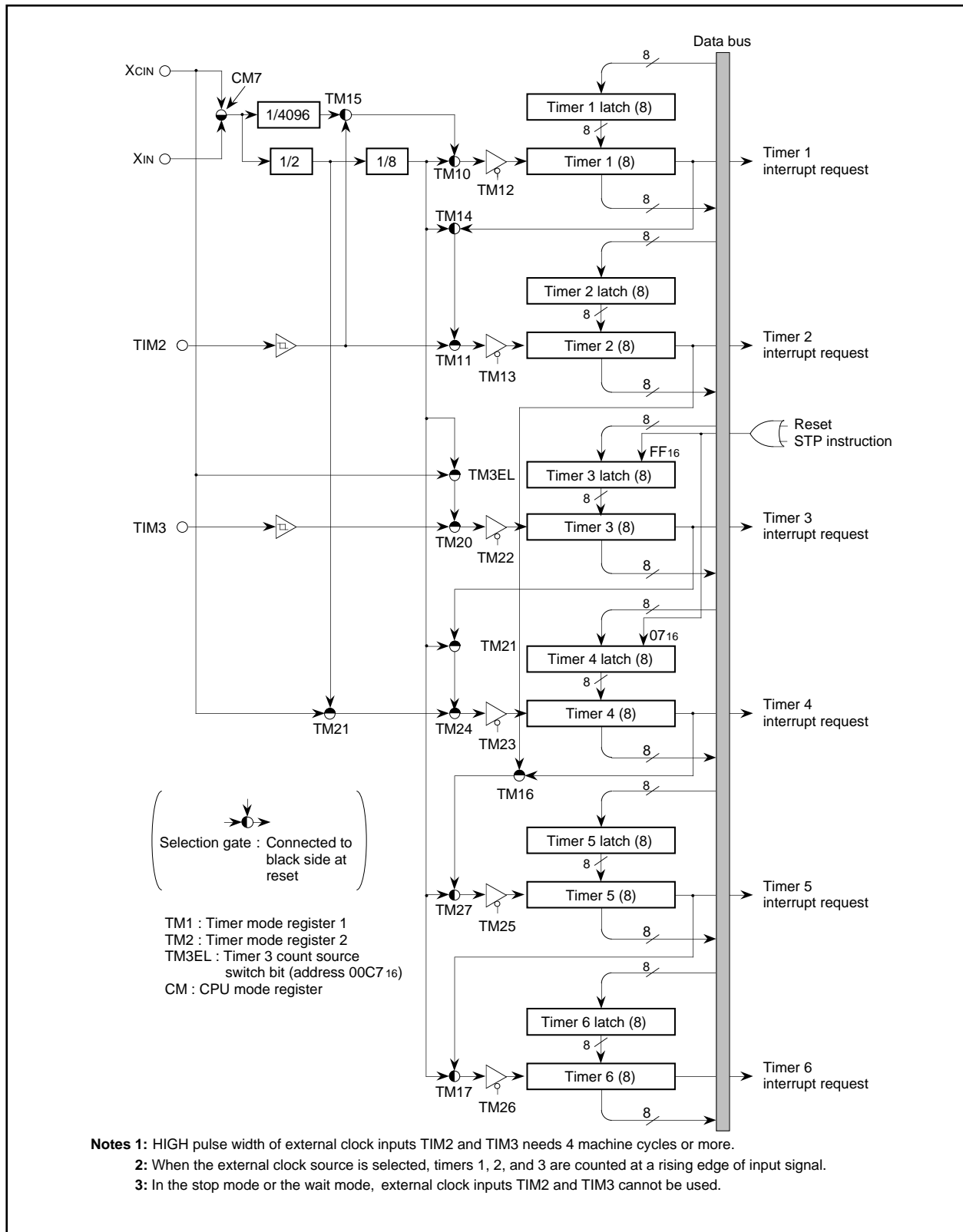


Fig. 12. Timer-related Registers

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER



**Fig. 13. Timer Block Diagram**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**SERIAL I/O**

The M37274MA-XXXSP has a built-in serial I/O which can either transmit or receive 8-bit data serially in the clock synchronous mode. The serial I/O block diagram is shown in Figure 14. The synchronous clock I/O pin (SCLK), and data output pin (SOUT) also function as port P4, data input pin (SIN) also functions as port P1. Bit 2 of the serial I/O mode register (address 021316) selects whether the synchronous clock is supplied internally or externally (from the P46/SCLK pin). When an internal clock is selected, bits 1 and 0 select whether f(XIN) or f(XCIN) is divided by 8, 16, 32, or 64. To use SOUT and P46/SCLK pins for serial I/O, set the corresponding bits of the port P4 direction register (address 00C916) to "0." To use SIN pin for serial I/O, set the corresponding bit of the port P1 direction register (address 00C316) to "0."

The operation of the serial I/O is described below. The operation of the serial I/O differs depending on the clock source; external clock or internal clock.

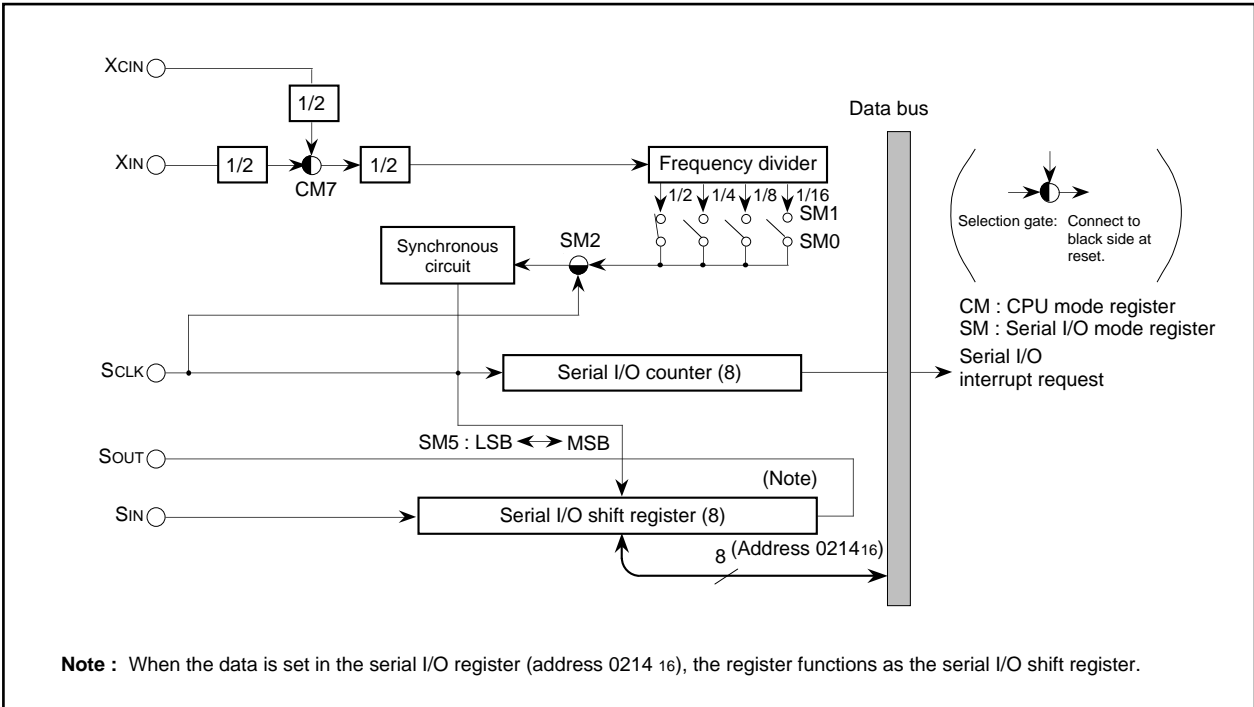


Fig. 14. Serial I/O Block Diagram

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

**SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER**

**Internal clock :** The serial I/O counter is set to "7" during the write cycle into the serial I/O register (address 0214<sub>16</sub>), and the transfer clock goes "H" forcibly. At each falling edge of the transfer clock after the write cycle, serial data is output from the SOUT pin. Transfer direction can be selected by bit 5 of the serial I/O mode register. At each rising edge of the transfer clock, data is input from the SIN pin and data in the serial I/O register is shifted 1 bit.

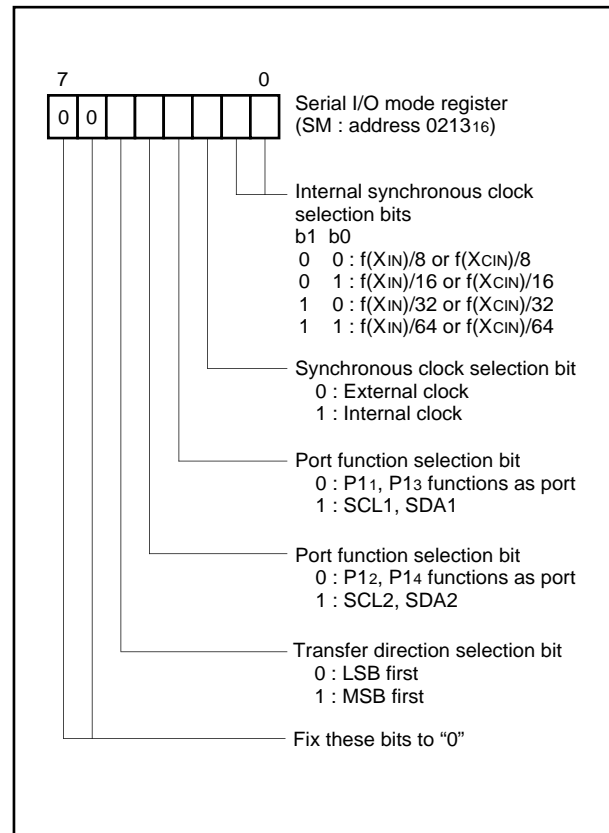
After the transfer clock has counted 8 times, the serial I/O counter becomes "0" and the transfer clock stops at HIGH. At this time the interrupt request bit is set to "1."

**External clock :** The an external clock is selected as the clock source, the interrupt request is set to "1" after the transfer clock has been counted 8 counts. However, transfer operation does not stop, so the clock should be controlled externally. Use the external clock of 500kHz or less with a duty cycle of 50%.

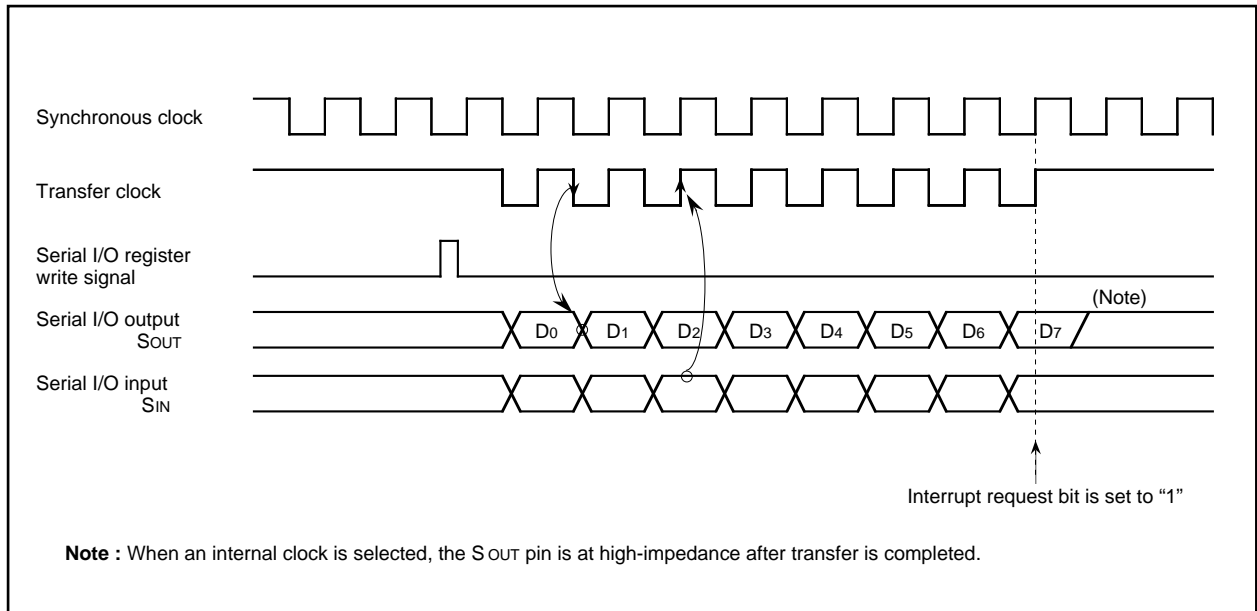
The serial I/O timing is shown in Figure 15. When using an external clock for transfer, the external clock must be held at HIGH for initializing the serial I/O counter. When switching between an internal clock and an external clock, do not switch during transfer. Also, be sure to initialize the serial I/O counter after switching.

**Notes 1:** On programming, note that the serial I/O counter is set by writing to the serial I/O register with the bit managing instructions, such as SEB and CLB.

**2:** When an external clock is used as the synchronous clock, write transmit data to the serial I/O register when the transfer clock input level is HIGH.



**Fig. 16. Serial I/O Mode Register**



**Fig. 15. Serial I/O Timing (for LSB first)**



**PRELIMINARY**  
Notice: This is not a final specification.  
Some parametric limits are subject to change.

## PWM OUTPUT FUNCTION

The M37274MA-XXXSP is equipped with a 14-bit PWM (DA) seven 8-bit PWMs (PWM0–PWM6). DA has a 14-bit resolution with the minimum resolution bit width of  $0.25 \mu\text{s}$  and a repeat period of  $4096 \text{ ms}$  (for  $f(X_{IN}) = 8 \text{ MHz}$ ). PWM0–PWM6 have the same circuit structure and an 8-bit resolution with minimum resolution bit width of  $4 \mu\text{s}$  and repeat period of  $1024 \mu\text{s}$  (for  $f(X_{IN}) = 8 \text{ MHz}$ ).

Figure 17 shows the PWM block diagram. The PWM timing generating circuit applies individual control signals to PWM0–PWM6 using  $f(X_{IN})$  divided by 2 as a reference signal.

### (1) Data Setting

When outputting DA, first set the high-order 8 bits to the DA-H register (address  $0240_{16}$ ), then the low-order 6 bits to the DA-L register (address  $0241_{16}$ ). When outputting PWM0–PWM6, set 8-bit output data to the PWM $i$  register ( $i$  means 0 to 6; addresses  $0200_{16}$  to  $0206_{16}$ ).

### (2) Transmitting Data from Register to PWM circuit

Data transfer from the 8-bit PWM register to the 8-bit PWM circuit is executed at writing data to the register.

The signal output from the 8-bit PWM output pin corresponds to the contents of this register.

Also, data transfer from the DA register (addresses  $0240_{16}$  and  $0241_{16}$ ) to the 14-bit PWM circuit is executed at writing data to the DA-L register (address  $0241_{16}$ ). Reading from the DA-H register (address  $0240_{16}$ ) means reading this transferred data. Accordingly, it is possible to confirm the data being output from the D-A output pin by reading the DA register.

### (3) Operating of 8-bit PWM

The following explains PWM operation.

First, set the bit 0 of PWM mode register 1 (address  $020A_{16}$ ) to "0" (at reset, bit 0 is already set to "0" automatically), so that the PWM count source is supplied.

PWM0–PWM3 are also used as pins P04–P07, PWM4–PWM6 are also used as pins P00–P02, respectively. Set the corresponding bits of the port P0 direction register to "1" (output mode). And select each output polarity by bit 3 of PWM mode register 1 (address  $020A_{16}$ ). Then, set bits 7 to 0 of PWM mode register 2 to "1" (PWM output). The PWM waveform is output from the PWM output pins by setting these registers.

Figure 18 shows the 8-bit PWM timing. One cycle ( $T$ ) is composed of  $256 (2^8)$  segments. The 8 kinds of pulses, relative to the weight of each bit (bits 0 to 7), are output inside the circuit during 1 cycle. Refer to Figure 20 (a). The 8-bit PWM outputs waveform which is the logical sum (OR) of pulses corresponding to the contents of bits 0 to 7 of the 8-bit PWM register. Several examples are shown in Figure 20 (b). 256 kinds of output (HIGH area:  $0/256$  to  $255/256$ ) are selected by changing the contents of the PWM register. A length of entirely HIGH cannot be output, i.e.  $256/256$ .

### (4) Operating of 14-bit PWM

As with 8-bit PWM, set the bit 0 of the PWM mode register 1 (address  $020A_{16}$ ) to "0" (at reset, bit 0 is already set to "0" automatically), so that the PWM count source is supplied. Pin DA is also used as port P03. Select output mode by setting bit 3 of the port P0 direction register. Next, select the output polarity by bit 3 of the PWM mode register 1. Then, the 14-bit PWM outputs from the D-A output pin by setting bit 1 of the PWM mode register 1 to "0" (at reset, this bit already set to "0" automatically) to select the DA output.

The output example of the 14-bit PWM is shown in Figure 19.

The 14-bit PWM divides the data of the DA latch into the low-order 6 bits and the high-order 8 bits.

The fundamental waveform is determined with the high-order 8-bit data "DH." A "H" level area with a length  $\tau \times D_H$  ("H" level area of fundamental waveform) is output every short area of " $t$ " =  $256\tau$  =  $64 \text{ ms}$  ( $\tau$  is the minimum resolution bit width of  $0.25 \mu\text{s}$ ). The "H" level area increase interval ( $t_m$ ) is determined with the low-order 6-bit data "DL." The "H" level are of smaller intervals " $t_m$ " shown in Table 6 is longer by  $\tau$  than that of other smaller intervals in PWM repeat period " $T$ " =  $64t$ . Thus, a rectangular waveform with the different "H" width is output from the D-A pin. Accordingly, the PWM output changes by  $\tau$  unit pulse width by changing the contents of the DA-H and DA-L registers. A length of entirely "H" output cannot be output, i. e.  $256/256$ .

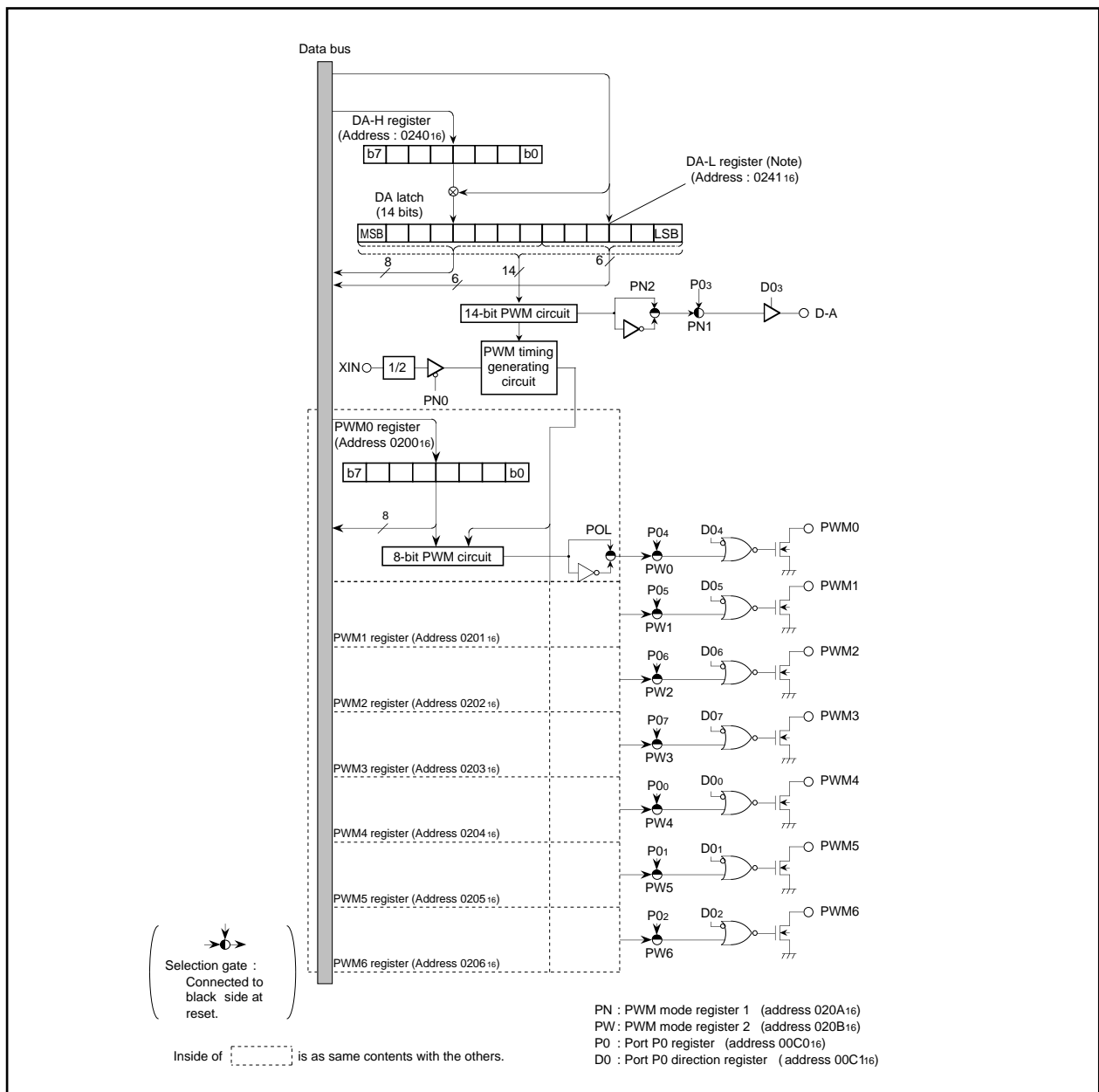
### (5) Output after Reset

At reset, the output of ports P00–P02 and P04–P07 is in the high-impedance state and the contents of the PWM register and the PWM circuit are undefined. Note that after reset, the PWM output is undefined until setting the PWM register.

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**Table 2. Relation Between Low-order 6-bit Data and High-level Area Increase Interval**

Low-order 6 bits of Data	Area Longer by $\tau$ Than That of Other $t_m$ ( $m = 0$ to 63)
0 0 0 0 0 0 <sup>LSB</sup>	Nothing
0 0 0 0 0 1	$m = 32$
0 0 0 0 1 0	$m = 16, 48$
0 0 0 1 0 0	$m = 8, 24, 40, 56$
0 0 1 0 0 0	$m = 4, 12, 20, 28, 36, 44, 52, 60$
0 1 0 0 0 0	$m = 2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62$
1 0 0 0 0 0	$m = 1, 3, 5, 7, \dots, 57, 59, 61, 63$



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

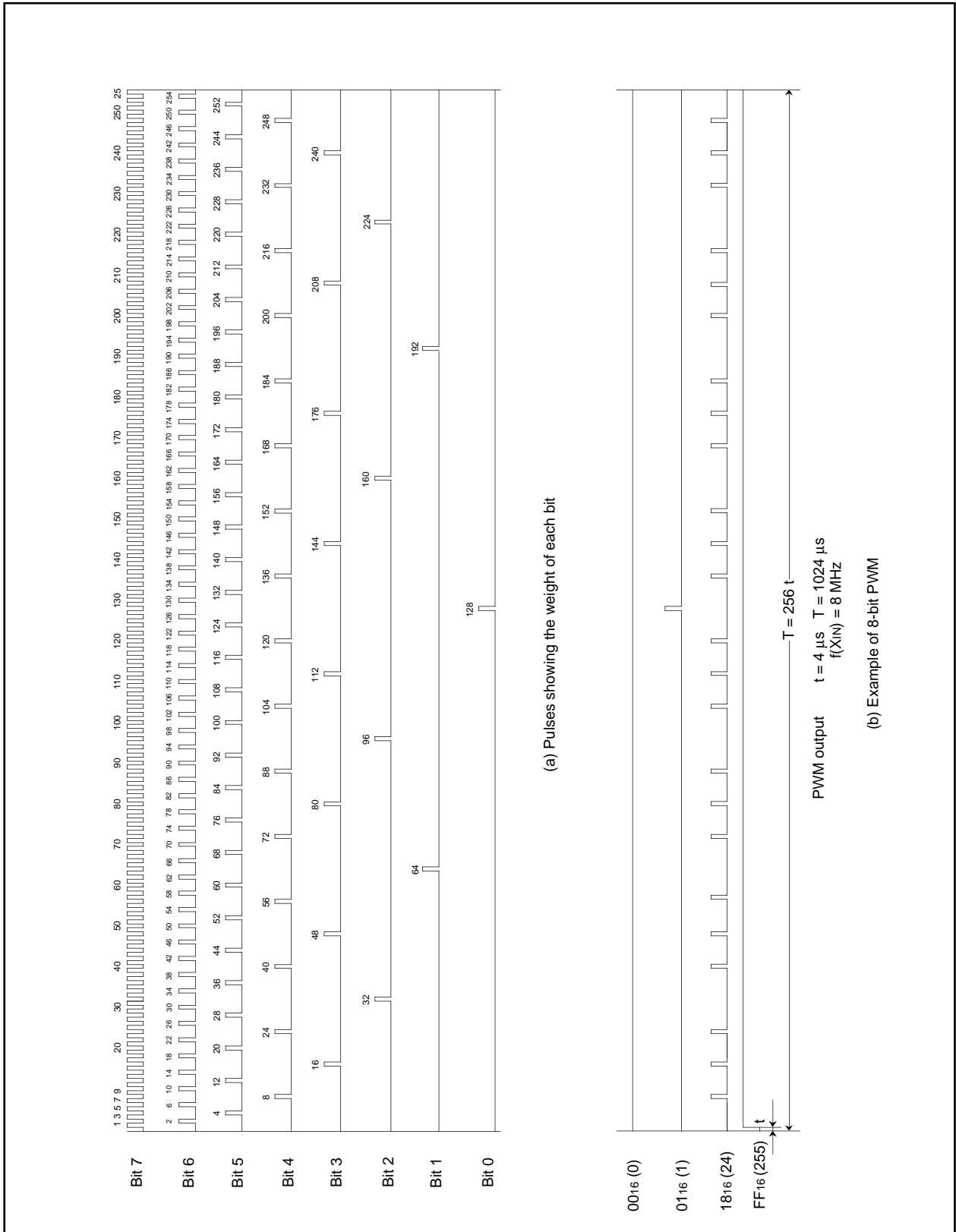


Fig. 18. 8-bit PWM Timing

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

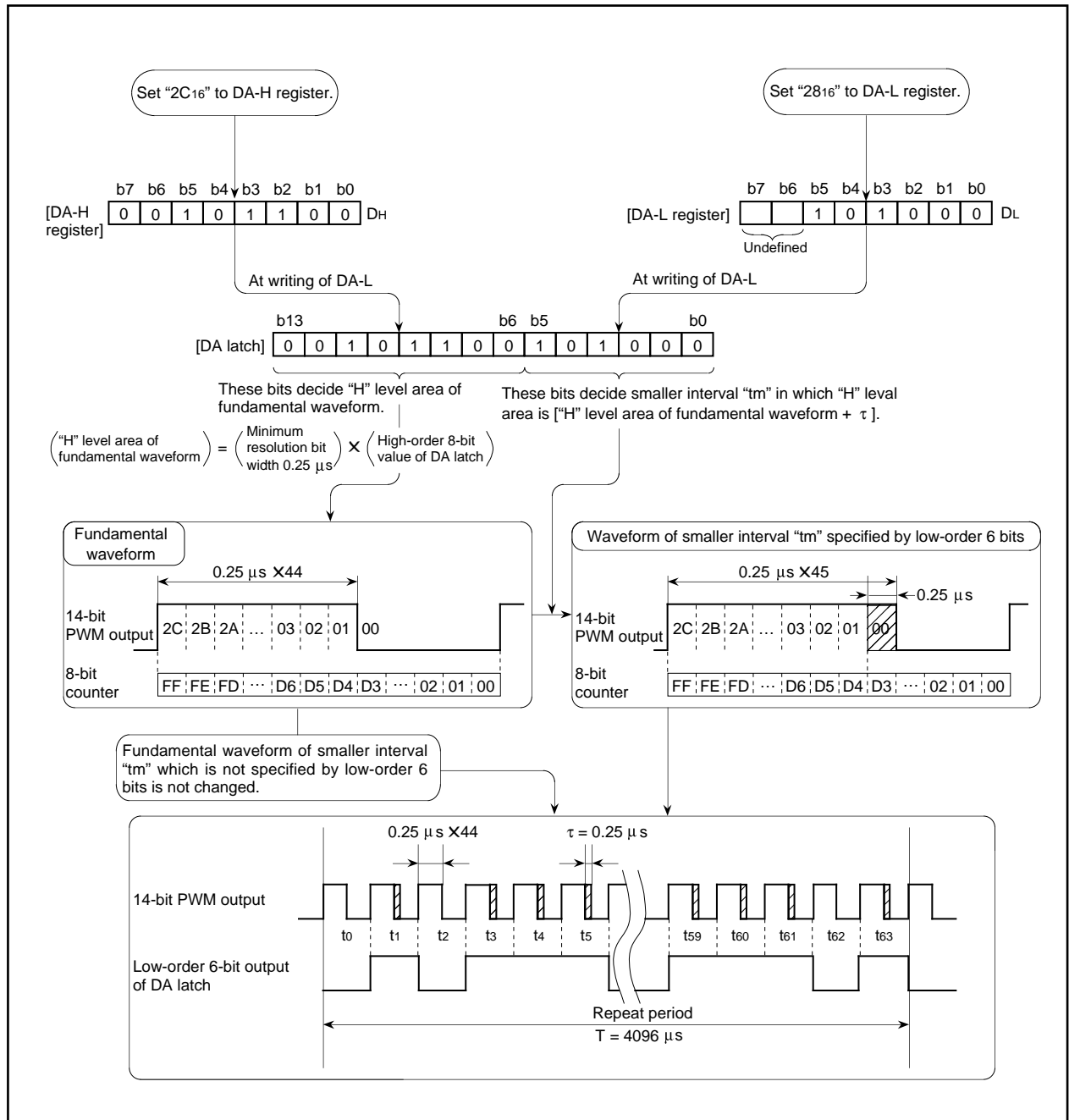


Fig. 19. 14-bit PWM Output Example (f(XIN) = 8MHz)

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

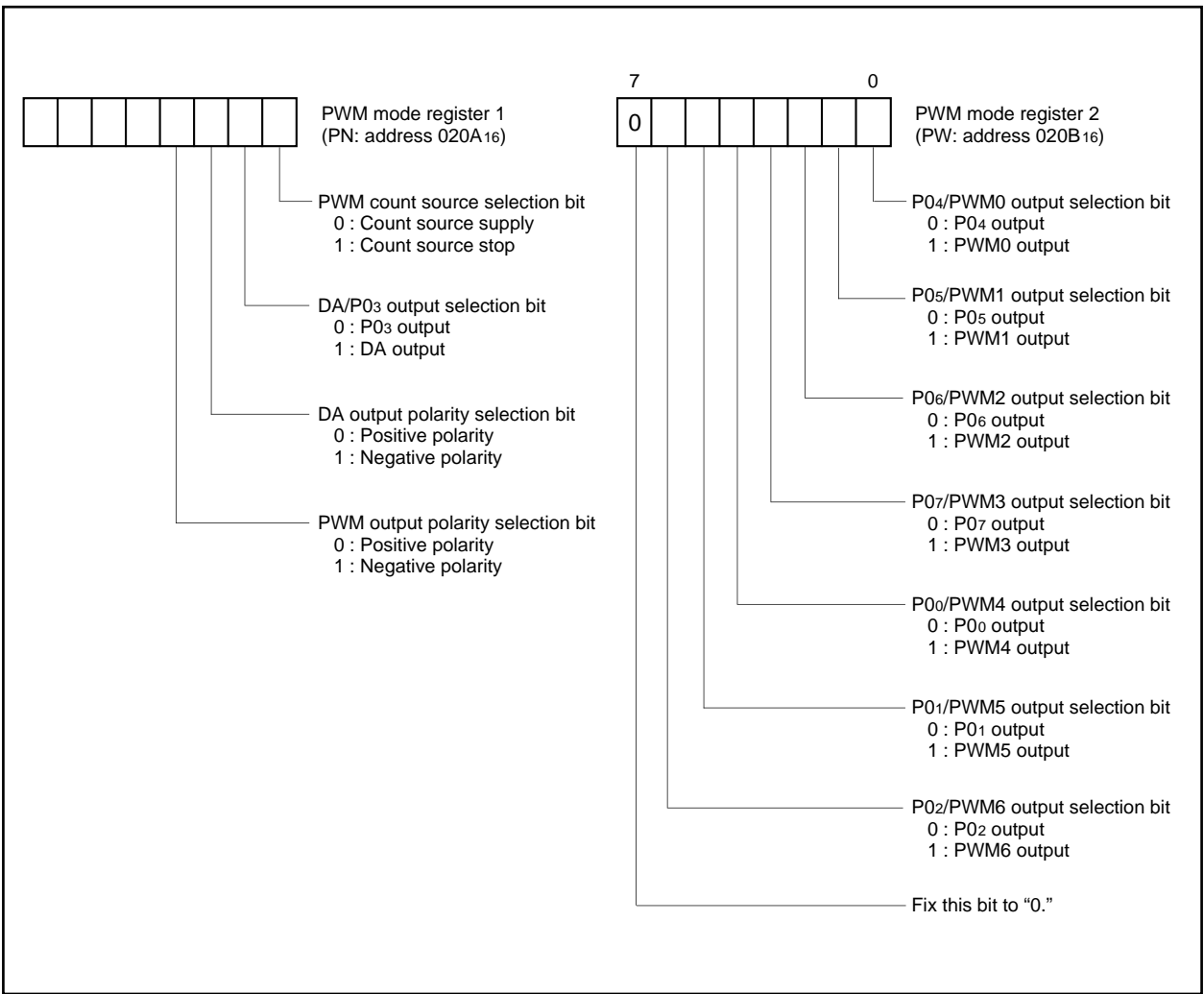


Fig. 20. PWM-related Registers

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**A-D CONVERTER**

**(1) A-D Conversion Register (AD)**

A-D conversion register is a read-only register that stores the result of an A-D conversion. This register should not be read during A-D conversion.

**(2) A-D Control Register (ADCON)**

The A-D control register controls A-D conversion. Bits 1 and 0 of this register select analog input pins. When these pins are not used as analog input pins, they are used as ordinary I/O pins. Bit 3 is the A-D conversion completion bit, A-D conversion is started by writing "0" to this bit. The value of this bit remains at "0" during an A-D conversion, then changes to "1" when the A-D conversion is completed.

Bit 4 controls connection between the resistor ladder and VCC. When not using the A-D converter, the resistor ladder can be cut off from the internal VCC by setting this bit to "0," accordingly providing low-power dissipation.

**(3) Comparison Voltage Generator (Resistor Ladder)**

The voltage generator divides the voltage between VSS and VCC by 256, and outputs the divided voltages to the comparator as the reference voltage Vref.

**(4) Channel Selector**

The channel selector connects an analog input pin, selected by bits 1 and 0 of the A-D control register, to the comparator.

**(5) Comparator and Control Circuit**

The conversion result of the analog input voltage and the reference voltage "Vref" is stored in the A-D conversion register. The A-D conversion completion bit and A-D conversion interrupt request bit are set to "1" at the completion of A-D conversion.

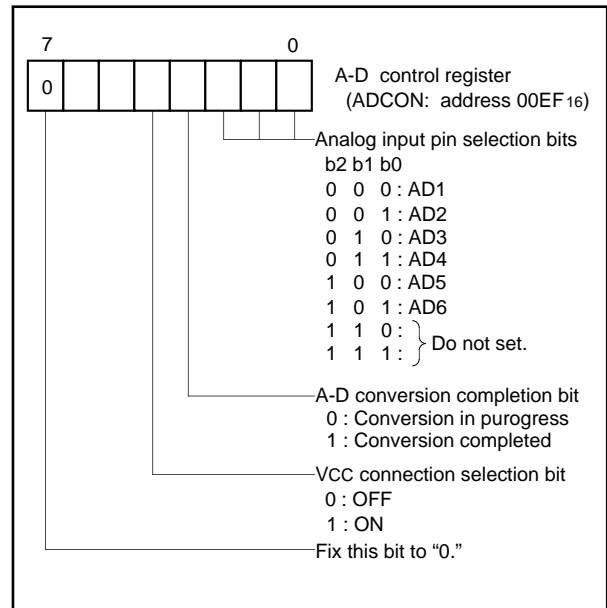


Fig. 21. A-D Control Register

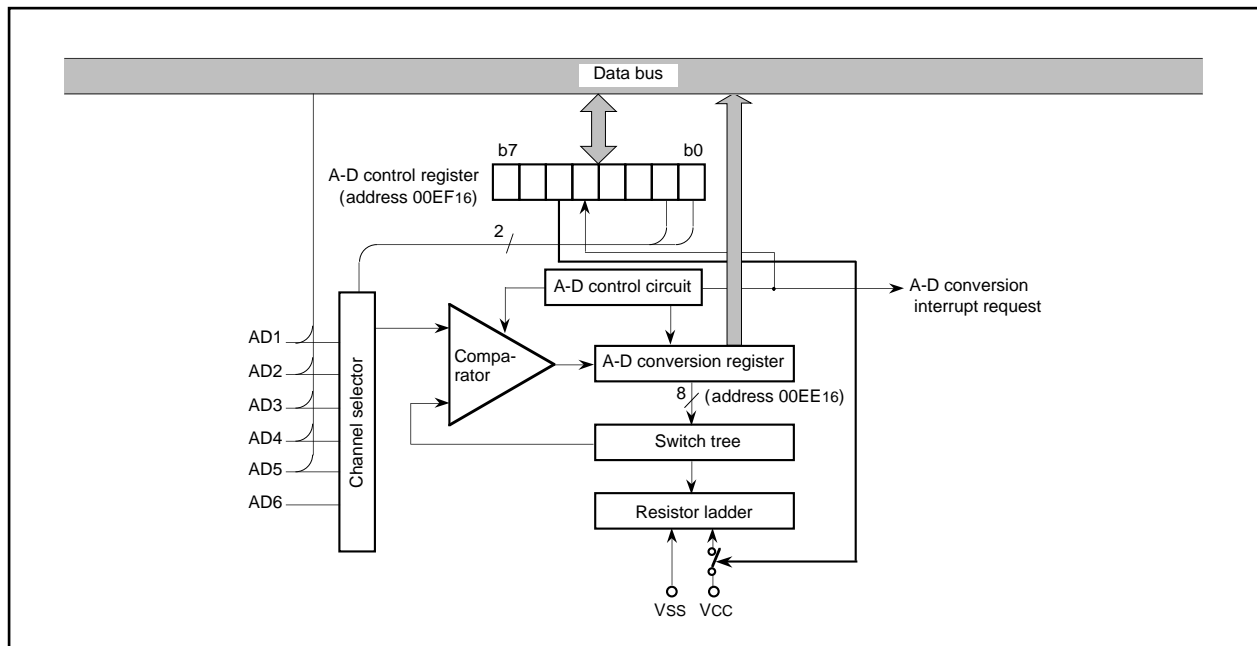


Fig. 22. A-D Comparator Block Diagram

**(6) Conversion Method**

- ① Set bit 7 of the interrupt input polarity register (address 021216) to "1" to generate an interrupt request at completion of A-D conversion.
- ② Set the A-D conversion • INT3 interrupt request bit to "0" (even when A-D conversion is started, the A-D conversion • INT3 interrupt request bit is not set to "0" automatically).
- ③ When using A-D conversion interrupt, enable interrupts by setting A-D conversion • INT3 interrupt request bit to "1" and setting the interrupt disable flag to "0."
- ④ Set the Vcc connection selection bit to "1" to connect Vcc to the resistor ladder.
- ⑤ Select analog input pins by the analog input selection bit of the A-D control register.
- ⑥ Set the A-D conversion completion bit to "0." This write operation starts the A-D conversion. Do not read the A-D conversion register during the A-D conversion.
- ⑦ Verify the completion of the conversion by the state ("1") of the A-D conversion completion bit, the state ("1") of A-D conversion • INT3 interrupt request bit, or the occurrence of an A-D conversion interrupt.
- ⑧ Read the A-D conversion register to obtain the conversion results.

**Note :** When the ladder resistor is disconnect from Vcc, set the Vcc connection selection bit to "0" between steps ⑦ and ⑧.

**(7) Internal Operation**

When the A-D conversion starts, the following operations are automatically performed.

- ① The A-D conversion register is set to "0016."
- ② The most significant bit of the A-D conversion register becomes "1," and the comparison voltage "Vref" is input to the comparator. At this point, Vref is compared with the analog input voltage "VIN."
- ③ Bit 7 is determined by the comparison results as follows.  
 When  $V_{ref} < V_{IN}$  : bit 7 holds "1"  
 When  $V_{ref} > V_{IN}$  : bit 7 becomes "0"

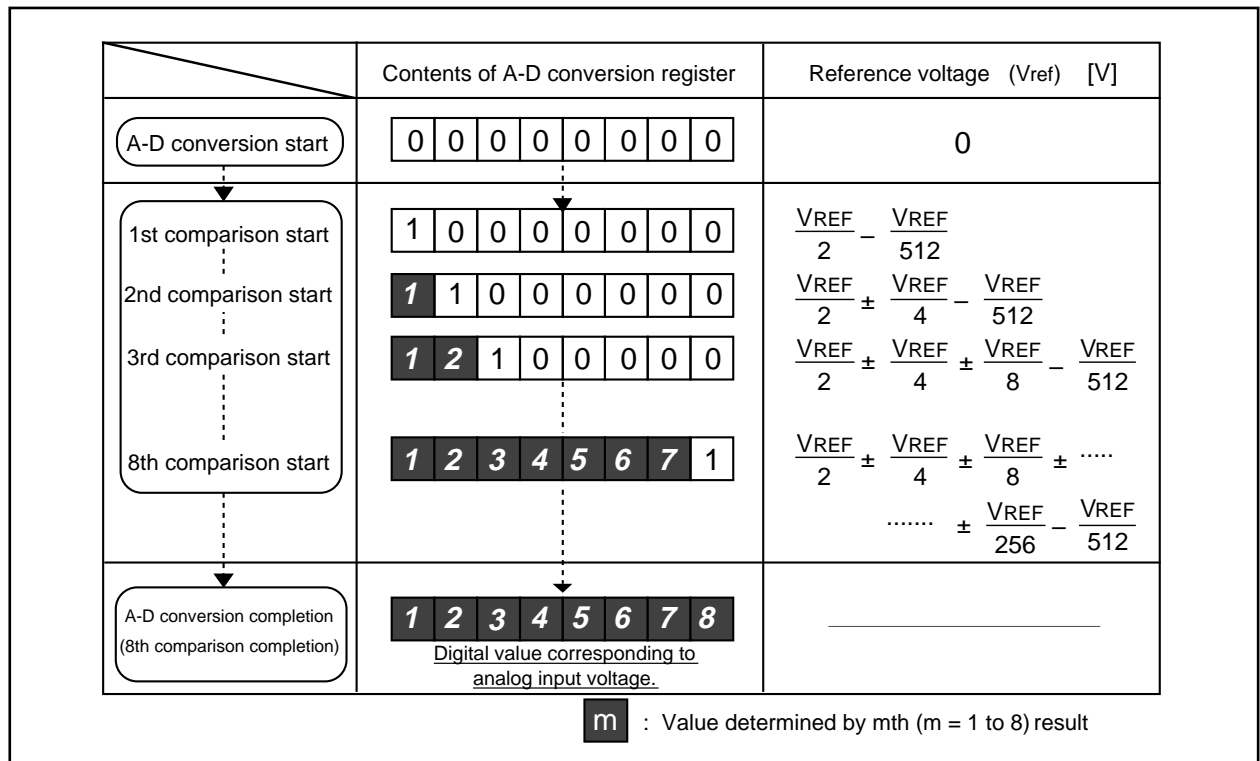
With the above operations, the analog value is converted into a digital value. The A-D conversion terminates in a maximum of 50 machine cycles (12.5 μs at  $f(X_{IN}) = 8$  MHz) after it starts, and the conversion result is stored in the A-D conversion register.

An A-D conversion interrupt request occurs at the same time as A-D conversion completion, the A-D conversion • INT3 interrupt request bit becomes "1." The A-D conversion completion bit also becomes "1."

**Table 3. Expression for Vref and VREF**

A-D conversion register contents "n" (decimal notation)	Vref (V)
0	0
1 to 255	$\frac{V_{REF}}{256} \times (n - 0.5)$

**Note:** VREF indicates the voltage of internal Vcc.



**Fig. 23. Changes in A-D Conversion Register and Comparison Voltage during A-D Conversion**

**(8) Definition of A-D Conversion Accuracy**

The definition of A-D conversion accuracy is described below.

① Relative accuracy

• **Zero transition error (V<sub>0T</sub>)**

The deviation of the input voltage at which A-D conversion output data changes from "0" to "1," from the corresponding ideal A-D conversion characteristics between 0 and V<sub>REF</sub>.

$$V_{0T} = \frac{(V_0 - 1/2 \times V_{REF}/256)}{1LSB} \quad [LSB]$$

• **Full-scale transition error (V<sub>FST</sub>)**

The deviation of the input voltage at which A-D conversion output data changes from "255" to "254," from the corresponding ideal A-D conversion characteristics between 0 and V<sub>REF</sub>.

$$V_{FST} = \frac{(V_{REF} - 3/2 \times V_{REF}/256) - V_{254}}{1LSB} \quad [LSB]$$

• **Non-linearity error**

The deviation of the actual A-D conversion characteristics, from the ideal A-D conversion characteristics between V<sub>0</sub> and V<sub>254</sub>.

$$\text{Non-linearity error} = \frac{V_n - (1LSB \times n + V_0)}{1LSB} \quad [LSB]$$

• **Differential non-linearity error**

The deviation of the input voltage required to change output data by "1," from the corresponding ideal A-D conversion characteristics between 0 and V<sub>REF</sub>.

$$\text{Differential non-linearity error} = \frac{(V_{n+1} - V_n) - 1LSB}{1LSB} [LSB]$$

② Absolute accuracy

• **Absolute accuracy error**

The deviation of the actual A-D conversion characteristics, from the ideal A-D conversion characteristics between 0 and V<sub>REF</sub>.

$$\text{Absolute accuracy error} = \frac{V_n - 1LSB \times (n + 1/2)}{1LSBA} \quad [LSB]$$

**Note:** The analog input voltage "V<sub>n</sub>" at which A-D conversion output data changes from "n" to "n + 1" (n ; 0 to 254) is as follows (refer to Figure 24) :

$$1LSB \text{ with respect to relative accuracy} = \frac{V_{254} - V_0}{254} [V]$$

$$1LSBA \text{ with respect to absolute accuracy} = \frac{V_{REF}}{256} [V]$$

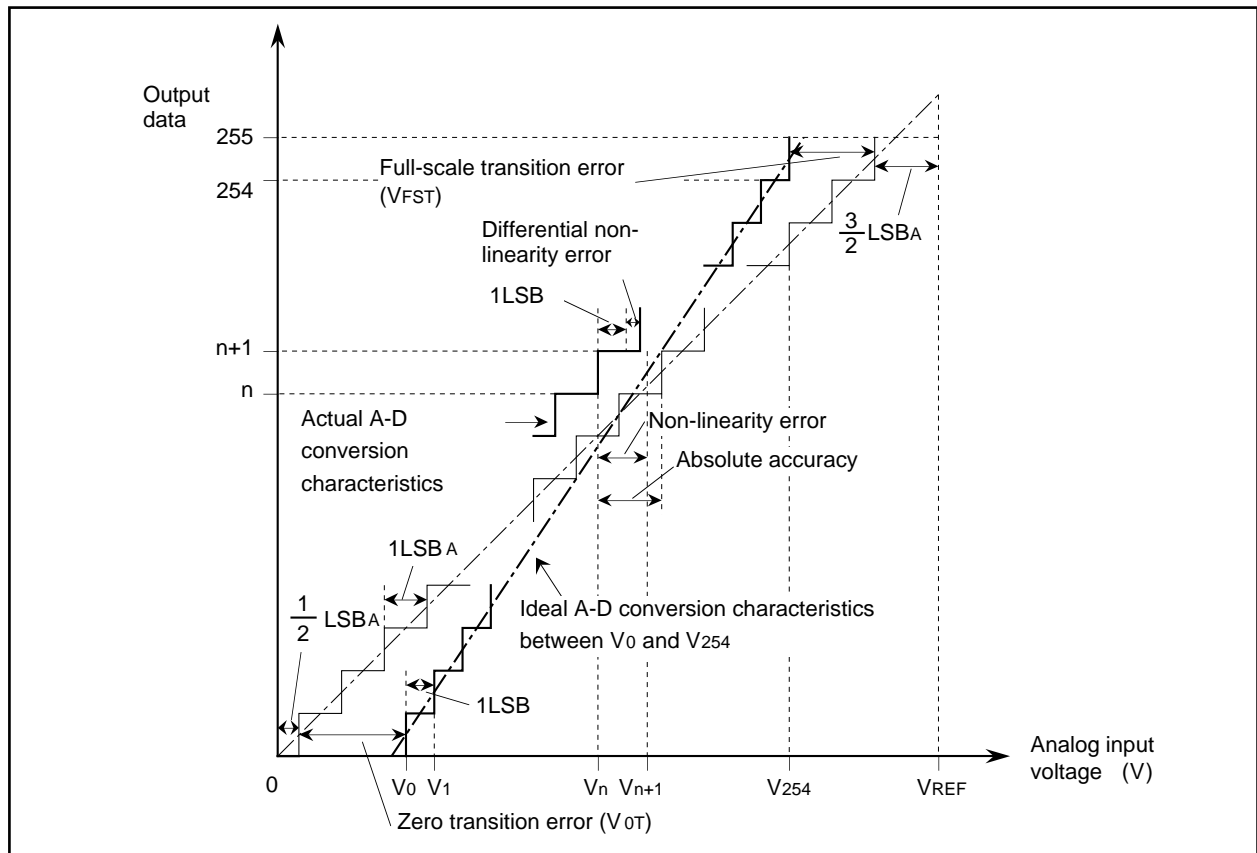


Fig. 24. Definition of A-D Conversion Accuracy



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**DATA SLICER**

The M37274MA-XXXSP includes the data slicer function for the closed caption decoder (referred to as the CCD). This function takes out the caption data superimposed in the vertical blanking interval of a composite video signal. A composite video signal which makes the sync chip's polarity negative is input to the CV<sub>IN</sub> pin.

When the data slicer function is not used, the data slicer circuit can be cut off by setting bit 0 of data slicer control register 1 (address 00EA<sub>16</sub>) to "0." Also, the timing signal generating circuit can be cut off by setting bit 0 of data slicer control register 2 (address 00EB<sub>16</sub>) to "0." These settings can realize the low-power dissipation.

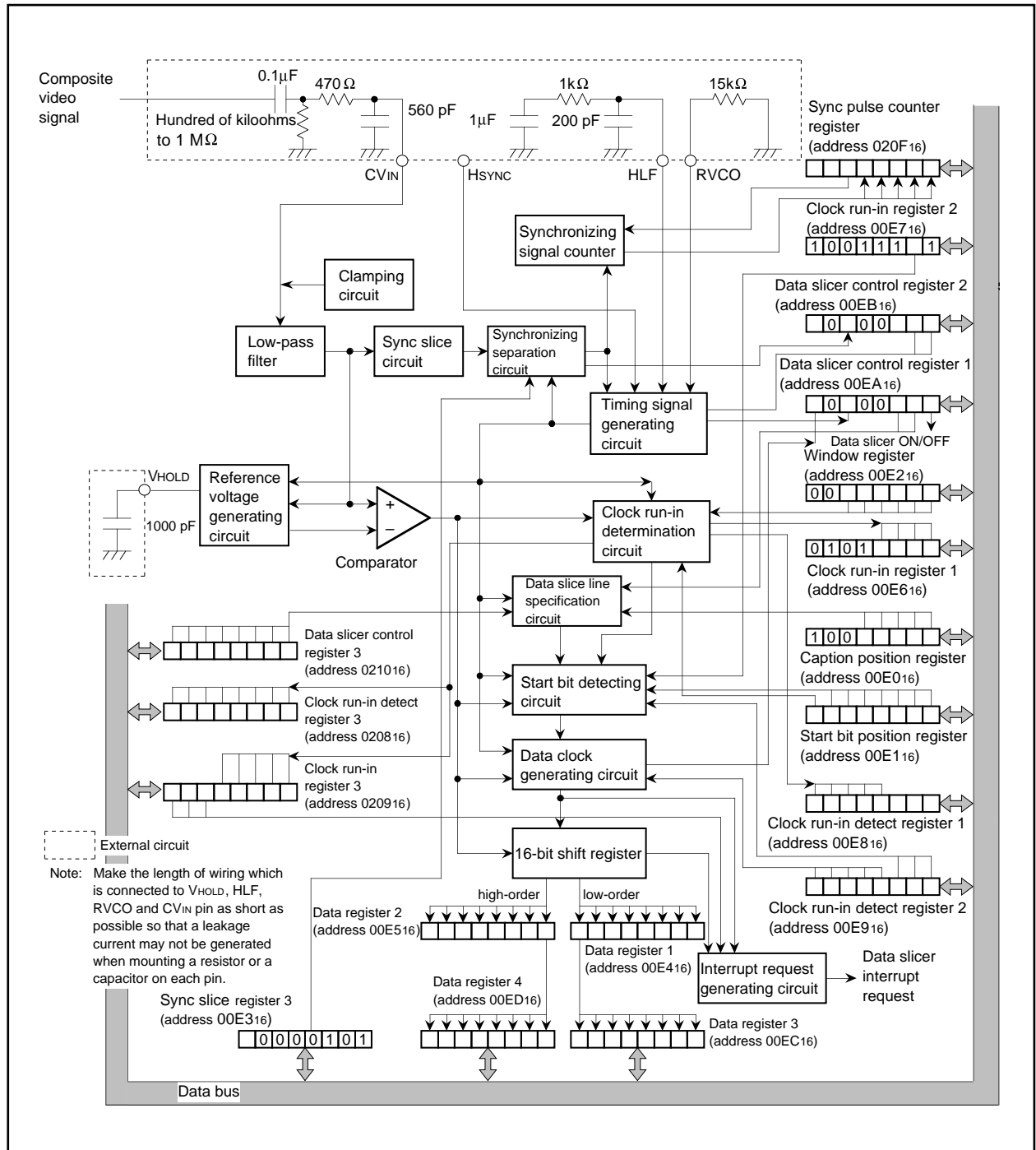


Fig. 25. Data Slicer Block Diagram

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

**SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER**

Figure 26 shows the data slicer control registers.

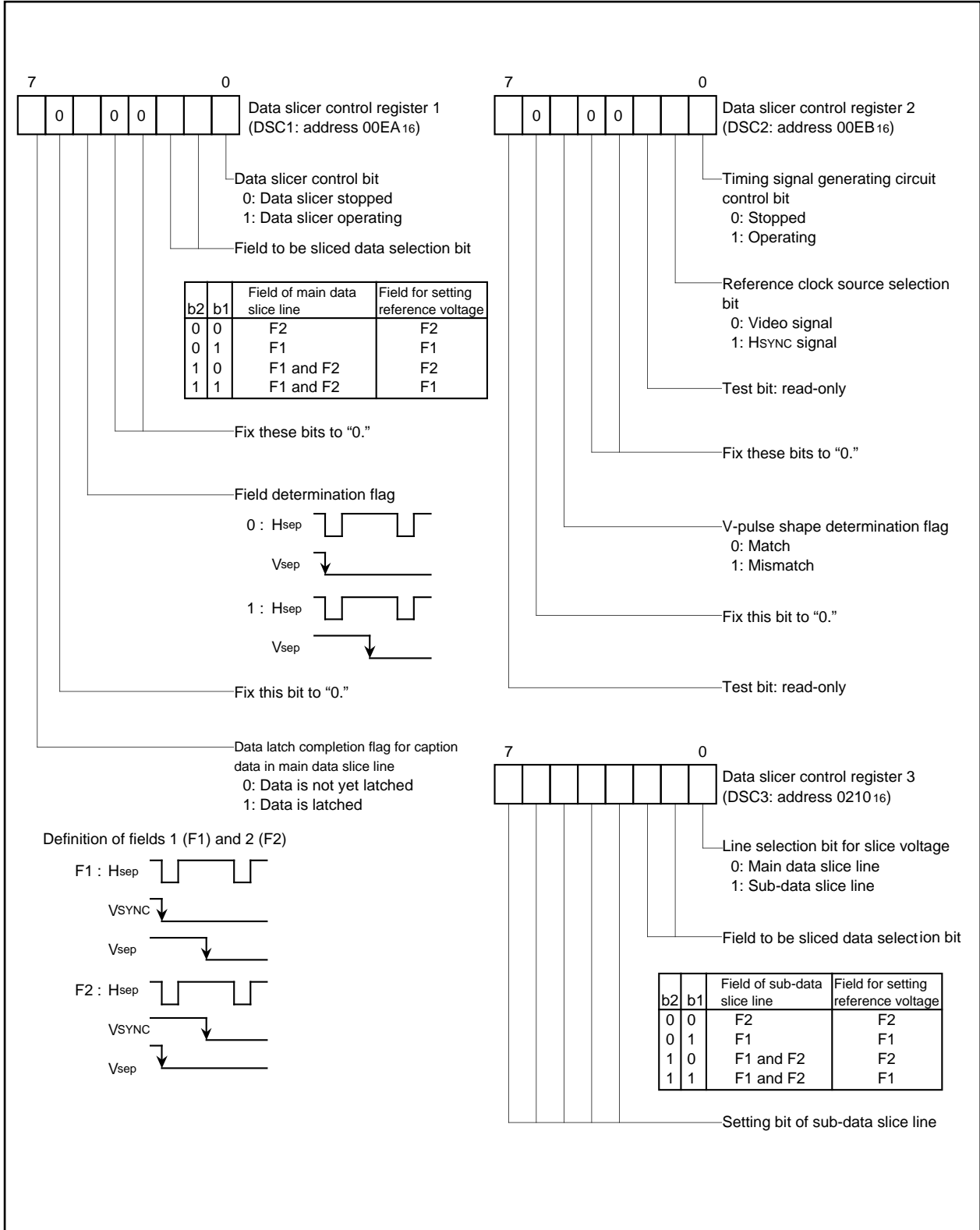


Fig. 26. Data Slicer Control Registers

**(1) Clamping Circuit and Low-pass Filter**

This filter attenuates the noise of the composite video signal input from the CV<sub>IN</sub> pin. The CV<sub>IN</sub> pin to which composite video signal is input requires a capacitor (0.1 μF) coupling outside. Pull down the CV<sub>IN</sub> pin with a resistor of hundreds of kilohms to 1 M. In addition, we recommend to install externally a simple low-pass filter using a resistor and a capacitor at the CV<sub>IN</sub> pin (refer to Figure 25).

**(2) Sync Slice Circuit**

This circuit takes out a composite sync signal from the output signal of the low-pass filter. Figure 27 shows the structure of the sync slice register.

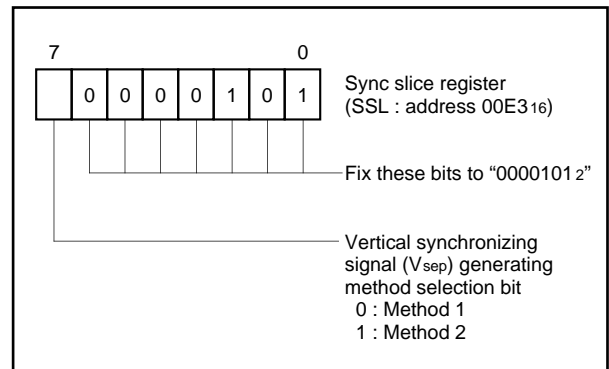


Fig. 27. Sync Slice Register

**(3) Synchronous Signal Separation Circuit**

This circuit separates a horizontal synchronous signal and a vertical synchronous signal from the composite sync signal taken out in the sync slice circuit.

① Horizontal synchronous signal (H<sub>sep</sub>)

A one-shot horizontal synchronous signal H<sub>sep</sub> is generated at the falling edge of the composite sync signal.

② Vertical synchronous signal (V<sub>sep</sub>)

As a V<sub>sep</sub> signal generating method, it is possible to select one of the following 2 methods by using bit 7 of the sync slice register (address 00E316).

•Method 1 The "L" level width of the composite sync signal is measured. If this width exceeds a certain time, a V<sub>sep</sub> signal is generated in synchronization with the rising of the timing signal immediately after this "L" level.

•Method 2 The "L" level width of the composite sync signal is measured. If this width exceeds a certain time, it is detected whether a falling of the composite sync signal exits or not in the "L" level period of the timing signal immediately after this "L" level. If a falling exists, a V<sub>sep</sub> signal is generated in synchronization with the rising of the timing signal (refer to Figure 28).

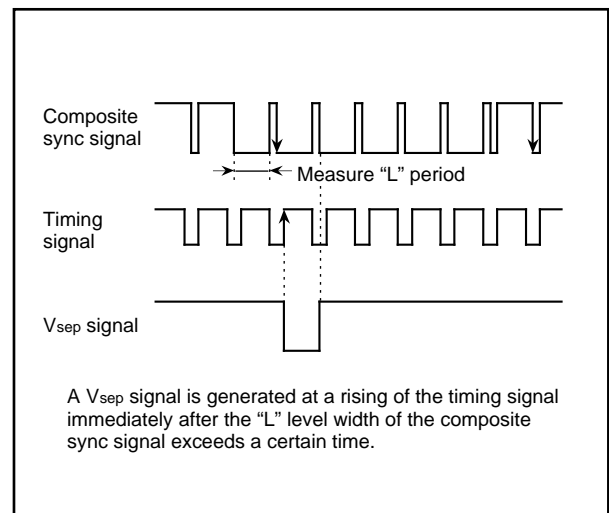


Fig. 28. Vsep Generating Timing (method 2)

Figure 28 shows a V<sub>sep</sub> generating timing. The timing signal shown in the figure is generated from the reference clock which the timing generating circuit outputs.

Reading bit 5 of data slicer control register 2 permits determining the shape of the V-pulse portion of the composite sync signal. As shown in Figure 29, when the A level matches the B level, this bit is "0." In the case of a mismatch, the bit is "1."

For the pins RVCO and the HLF, connect a resistor and a capacitor as shown in Figure 25. Make the length of wiring which is connected to these pins as short as possible so that a leakage current may not be generated.

**Note:** It takes a few tens of milliseconds until the reference clock becomes stable after the data slicer and the timing signal generating circuit are started. In this period, various timing signals, H<sub>sep</sub> signals and V<sub>sep</sub> signals become unstable. For this reason, take stabilization time into consideration when programming.

**PRELIMINARY**  
Notice: This is not a final specification.  
Some parametric limits are subject to change.

**(4) Timing Signal Generating Circuit**

This circuit generates a reference clock which is 832 times as large as the horizontal synchronous signal frequency. It also generates various timing signals on the basis of the reference clock, horizontal synchronous signal and vertical synchronizing signal. The circuit operates by setting bit 0 of data slicer control register 2 (address 00EB16) to "1."

The reference clock can be used as a display clock for OSD function in addition to the data slicer. The Hsync signal can be used as a count source instead of the composite sync signal. However, when the Hsync signal is selected, the data slicer cannot be used. A count source of the reference clock can be selected by bit 1 of data slicer control register 2 (address 00EB16).

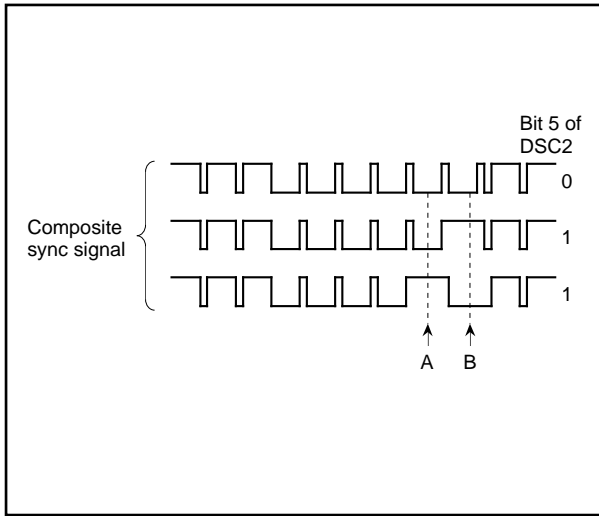


Fig. 29. Determination of V-pulse Waveform

**(5) Data Slice Line Specification Circuit**

① Specification of data slice line

M37274MA-XXXSP has 2 data slice line specification circuits for slicing arbitrary 2  $H_{sep}$  in 1 field. The following 2 data slice lines are specified .

<Main data slice line>

This line is specified by the caption position register (address 00E0<sub>16</sub>).

<Sub-data slice line>

This line is specified by the data slicer control register 3 (address 00EB<sub>16</sub>).

The counter is reset at the falling edge of  $V_{sep}$  and is incremented by 1 every  $H_{sep}$  pulse. When the counter value matched the value specified by bits 4 to 0 of the caption position register (in case of the sub-data slice line, by bits 3 to 7 of the data slicer control register 3), this  $H_{sep}$  is sliced.

The values of "00<sub>16</sub>" to "1F<sub>16</sub>" can be set in the caption position register. Bit 7 to bit 5 are used for testing. Set "100<sub>2</sub>." Figure 30 shows the signals in the vertical blanking interval. Figure 31 shows the structure of the caption position register.

② Selection of field to be sliced data

In the case of the main data slice line, the field to be sliced data is selected by bits 2 and 1 of the data slicer control register 1 (address 00EA<sub>16</sub>). In the case of the sub-data slice line, the field is selected by bits 2 and 1 of the data slicer control register 3. When bit 2 of the data slicer control register 1 is set to "1," it is possible to slice data of both fields (refer to Figure 26).

③ Specification of line to set slice voltage

The reference voltage for slicing (slice voltage) is generated by integrating the amplitude of the clock run-in pulse in the particular line (refer to Table 4).

④ Field determination

The field determination flag can be read out by bit 5 of the data slicer control register 1. This flag charge at the falling edge of  $V_{sep}$ .

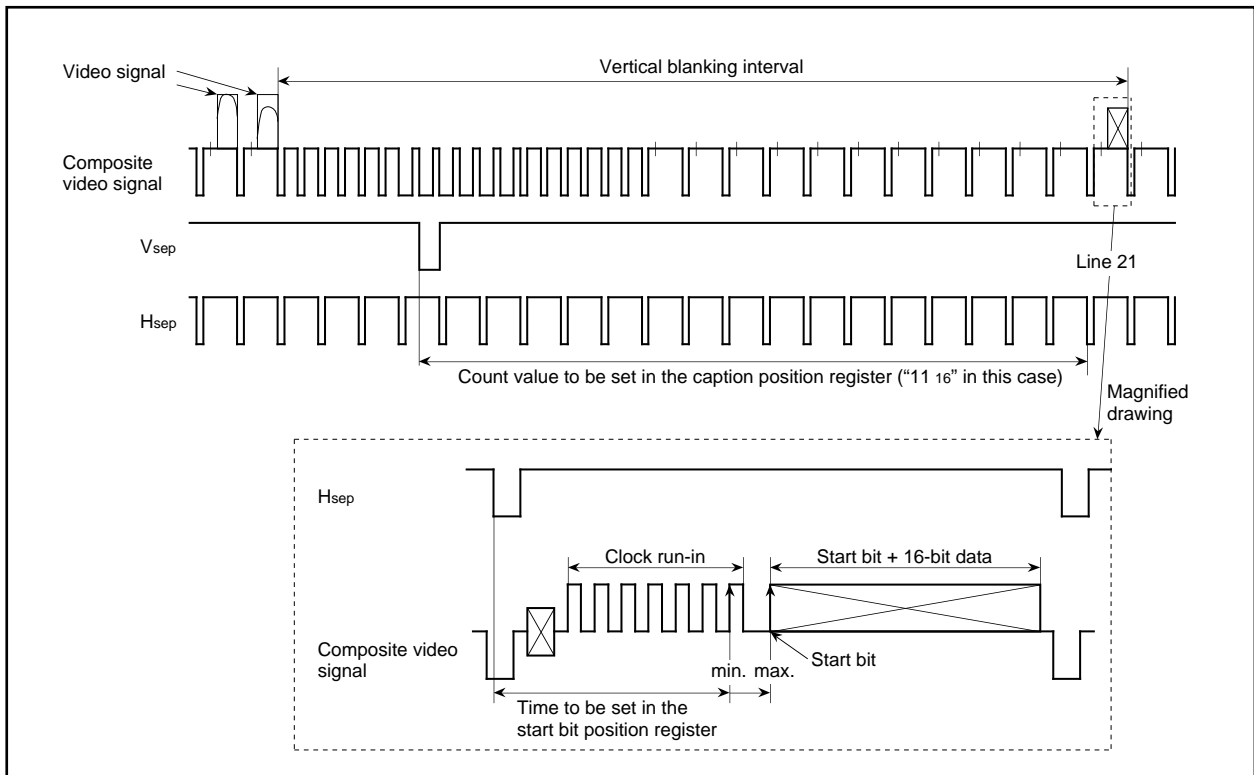
**Table 4. Specifying of Field Whose Sets Reference Voltage**

Bit 0 of DSC3	Field	Line
0	Field specified by bit 1 of DSC1    0: F2    1: F1	Line specified by bits 4 to 0 of CP (Main data slice line)
1	Field specified by bit 1 of DSC3    0: F2    1: F1	Line specified by bits 7 to 3 of DSC3 (Sub-data slice line)

DSC1 : Data slice control register 1

DSC3 : Data slice control register 3

CP : Caption position register



**Fig. 30. Signals in Vertical Blanking Interval**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**(6) Reference Voltage Generating Circuit and Comparator**

The composite video signal clamped by the clamping circuit is input to the reference voltage generating circuit and the comparator.

① Reference voltage generating circuit

This circuit generates a reference voltage (slice voltage) by using the amplitude of the clock run-in pulse in line specified by the data slice line specification circuit. Connect a capacitor between the V<sub>HOLD</sub> pin and the V<sub>SS</sub> pin, and make the length of wiring as short as possible so that a leakage current may not be generated.

② Comparator

The comparator compares the voltage of the composite video signal with the voltage (reference voltage) generated in the reference voltage generating circuit, and converts the composite video signal into a digital value.

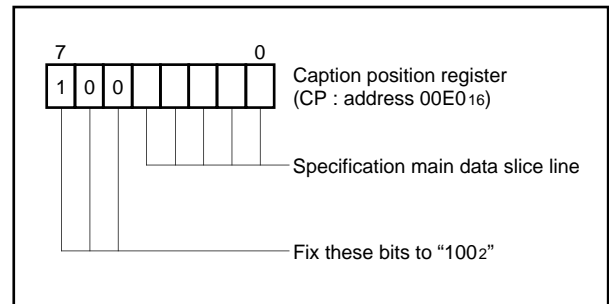


Fig. 31. Caption Position Register

**(7) Start Bit Detecting Circuit**

This circuit detects a start bit at line decided in the data slice line specification circuit. For start bit detection, it is possible to select one of the following two types by using bit 1 of clock run-in register 2 (address 00E716).

① After the lapse of the time corresponding to the set value of the start bit position register (address 00E116), the first rising of the composite video signal is detected as a start bit.

The time is set in bits 0 to 6 of the start bit position register (address 00E116) (refer to Figure 32). Set a value fit for the following conditions.

Figure 32 shows the structure of the start bit position register.

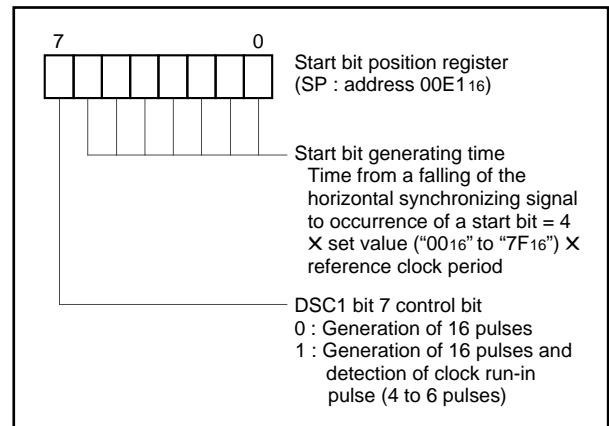
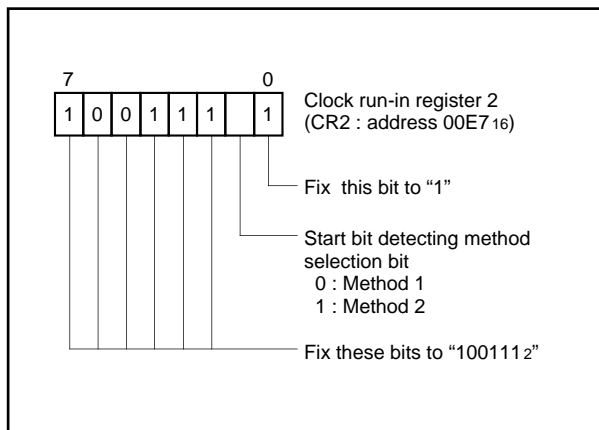


Fig. 32. Start Bit Position Register

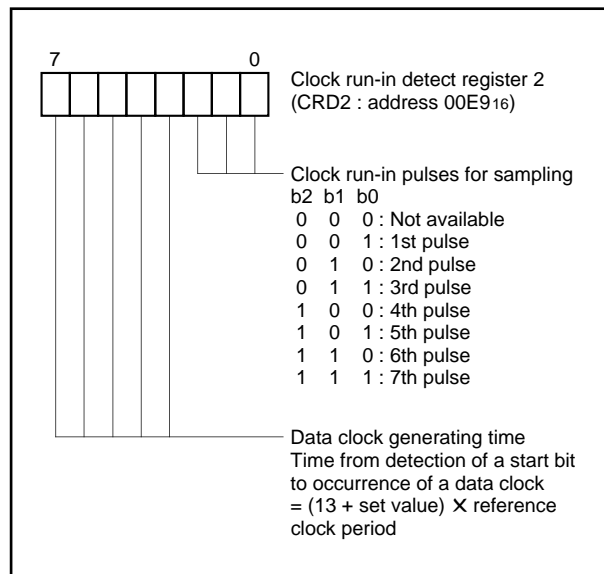
$$\left[ \begin{array}{l} \text{Time from the falling of the horizontal} \\ \text{synchronizing signal to the last rising} \\ \text{of the clock run-in} \end{array} \right] < \left[ \begin{array}{l} 4 \times \text{set value of the start bit position} \\ \text{register} \times \text{reference clock period} \end{array} \right] < \left[ \begin{array}{l} \text{Time from the falling of the horizontal} \\ \text{synchronous signal to occurrence of} \\ \text{the start bit} \end{array} \right]$$

② After a falling of the clock run-in pulse set in bits 2 to 0 of clock run-in detect register 2 (address 00E9<sub>16</sub>) is detected, a start bit is detected by sampling a comparator output. A sampling clock for sampling is obtained by dividing the reference clock generated in the timing signal generating circuit by 13.

Figure 34 shows the structure of clock run-in detect register 2. The contents of bits 2 to 0 of clock run-in detect register 2 and bit 1 of clock run-in register 2 are written at a falling of the horizontal synchronous signal. For this reason, even if an instruction for setting is executed, the contents of the register cannot be rewritten until a falling of the horizontal synchronous signal.



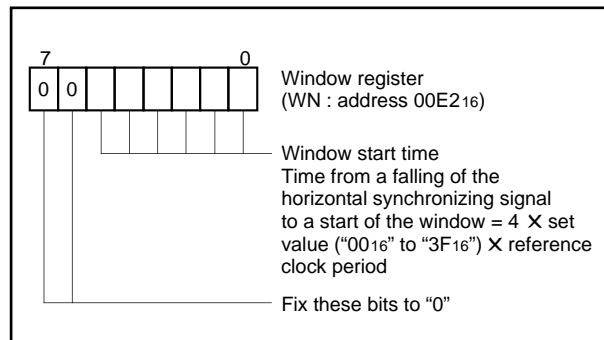
**Fig. 33. Clock Run-in Register 2**



**Fig. 34. Clock Run-in Detect Register 2**

**(8) Clock run-in determination circuit**

This circuit sets a window in the clock run-in portion in the composite video signal, and then determinates clock run-in by counting the number of pulses in this window. Set the time from a falling of the horizontal synchronizing signal to a start of the window by bits 0 to 5 of the window register (address 00E2<sub>16</sub>; refer to Figure 35). The window ends according to the contents of the setting of the start bit position register (refer to Figure 32).



**Fig. 35. Window Register**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

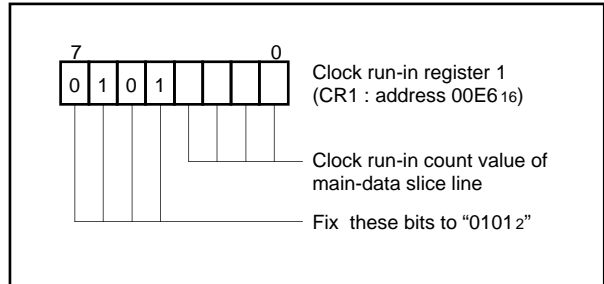
**SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER and ON-SCREEN DISPLAY CONTROLLER**

For the main data slice line, the count value of pulses in the window is stored in clock run-in register 1 (address 00E6<sub>16</sub>; refer to Figure 36). For the sub-data slice line, the count value of pulses in the window is stored in clock run-in register 3 (address 0209<sub>16</sub>; refer to Figure 37). When this count value is 4 to 6, it is determined as a clock run-in. Accordingly, set the count value so that the window may start after the first pulse of the clock run-in (refer to Figure 38).

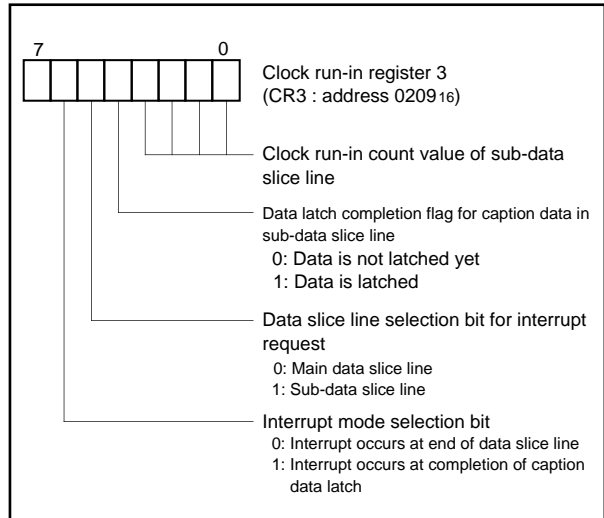
The contents to be set in the window register are written at a falling of the horizontal synchronous signal. For this reason, even if an instruction for setting is executed, the contents of the register cannot be rewritten until a falling of the horizontal synchronous signal.

For the main data slice line, reference clock is counted in the period from a falling of the clock pulse set in bits 0 to 2 of clock run-in detect register 2 (address 00E9<sub>16</sub>) to the next falling. The count value is stored in bits 3 to 7 of clock run-in detect register 1 (address 00E8<sub>16</sub>) (When the count value exceeds "1F<sub>16</sub>," "1F<sub>16</sub>" is held). For the sub-data slice line, the count value is stored in bits 3 to 7 of clock run-in detect register 3 (address 0208<sub>16</sub>). Read out these bits after the occurrence of a data slicer interrupt (refer to (11) Interrupt Request Generating Circuit).

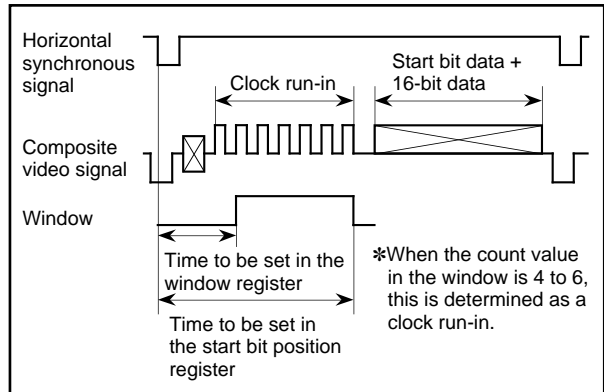
Figure 39 shows the structure of clock run-in detect registers 1 and 3.



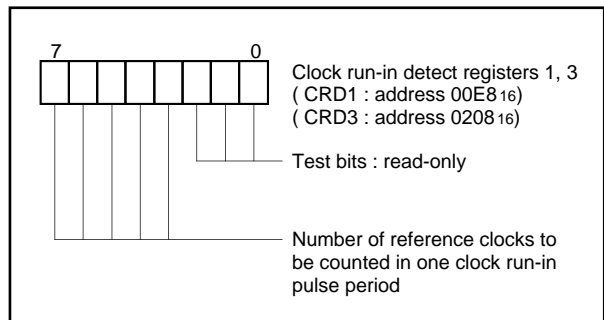
**Fig. 36. Clock Run-in Register 1**



**Fig. 37. Clock Run-in Register 3**



**Fig. 38. Window Setting**



**Fig. 39. Clock Run-in Detect Registers 1 and 3**



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**(9) Data clock generating circuit**

This circuit generates a data clock synchronized with the start bit detected in the start bit detecting circuit.

Set the time from detection of the start bit to occurrence of the data clock in bits 3 to 7 of clock run-in detect register 2 (address 00E9<sub>16</sub>). The time to be set is represented by the following expression:

$$\text{Time} = (13 + \text{set value}) \times \text{reference clock period}$$

For a data clock, 16 pulses are generated. When just 16 pulses have been generated, bit 7 of the data slicer control register is set to "1" (refer to Figure 26). When method 1 is already selected as a start bit detecting method, this bit becomes a logical product (AND) value with a clock run-in determination result by setting bit 7 of the start bit position register to "1."

When method 2 is already selected as a start bit detecting method and 16 pulses are generated of a data clock regardless of bit 7 of the start bit position register, this bit is set to "1." The contents of this bit are reset at a falling of the vertical synchronizing signal (V<sub>sep</sub>).

**Table 5. Setting Conditions for Caption Data Latch Completion Flag**

Bit 7 of SP	Conditions for Setting Bit 7 of DSC1 to "1"	Conditions for Setting Bit 4 of DSC3 to "1"
0	Data clock of 16 pulses has occurred in main data slice line	Data clock of 16 pulses has occurred in sub-data slice line
1	Data clock of 16 pulses has occurred in main data slice line AND Clock run-in pulse are detected 4 to 6 times	Data clock of 16 pulses has occurred in sub-data slice line AND Clock run-in pulse are detected 4 to 6 times

**(10) 16-bit Shift Register**

The caption data converted into a digital value by the comparator is stored into the 16-bit shift register in synchronization with the data clock. For the main data slice line, the contents of the high-order 8 bits of the stored caption data and the contents of the low-order 8 bits of the same data can be obtained by reading out data register 2 (address 00E5<sub>16</sub>) and data register 1 (address 00E4<sub>16</sub>), respectively. For the sub-data slice line, the contents of the high-order 8 bits and the contents of the low-order 8 bits can be obtained by reading out the data register 4 (address 00ED<sub>16</sub>) and data register 3 (address 00EC<sub>16</sub>), respectively. These registers are reset to "0" at a falling of V<sub>sep</sub>. Read out data registers 1 and 2 after the occurrence of a data slicer interrupt (refer to (11) Interrupt Request Generating Circuit).

**(11) Interrupt Request Generating Circuit**

The interrupt requests as shown in Table 6 are generated by combination of the following bits; bits 5 and 6 of the clock run-in register 3 (address 0209<sub>16</sub>), bit 1 of the clock run-in register 2 (address 00E7<sub>16</sub>). Read out the contents of data registers 1 to 4 and the contents of bits 3 to 7 of clock run-in detect registers 1 and 3 after the occurrence of a data slicer interrupt request.

**Table 6. Occurrence Sources of Interrupt Request**

CR3		CR2	Occurrence Sources of Interrupt Request	
b5	b6	b1	Slice line	Sources
0	0	0	Main data slice line	At end of data slice line
		1		Data clock of 16 pulses has occurred AND Clock run-in pulse are detected 4 to 6 times
	1	0		Data clock of 16 pulses has occurred
		1		
1	0	0	Sub-data slice line	At end of data slice line
		1		Data clock of 16 pulses has occurred AND Clock run-in pulse are detected 4 to 6 times
	1	0		Data clock of 16 pulses has occurred
		1		

**(12) Synchronous Signal Counter**

The synchronous signal counter counts the composite sync signal taken out from a video signal in the data slicer circuit or the vertical synchronous signal  $V_{sep}$  as a count source.

The count value in a certain time (T time) generated by  $f(X_{IN})/2^{13}$  or  $f(X_{IN})/2^{13}$  is stored into the 5-bit latch. Accordingly, the latch value changes in the cycle of T time. When the count value exceeds "1F<sub>16</sub>," "1F<sub>16</sub>" is stored into the latch.

The latch value can be obtained by reading out the sync pulse counter register (address 020F<sub>16</sub>). A count source is selected by bit 5 of the sync pulse counter register.

The synchronous signal counter is used when bit 0 of PWM mode register 1 (address 02EA<sub>16</sub>).

Figure 40 shows the structure of the sync pulse counter and Figure 41 shows the synchronous signal counter block diagram.

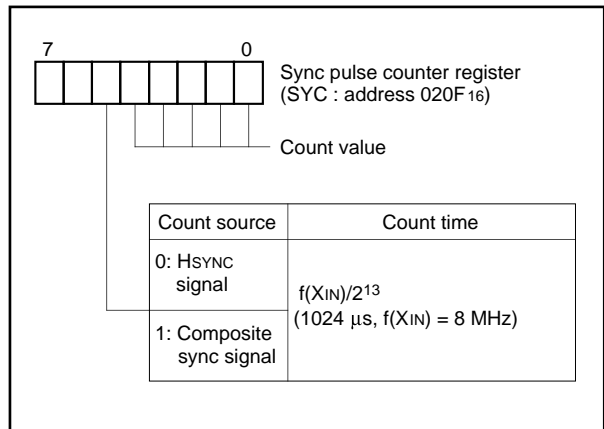


Fig. 40. Sync Pulse Counter Register

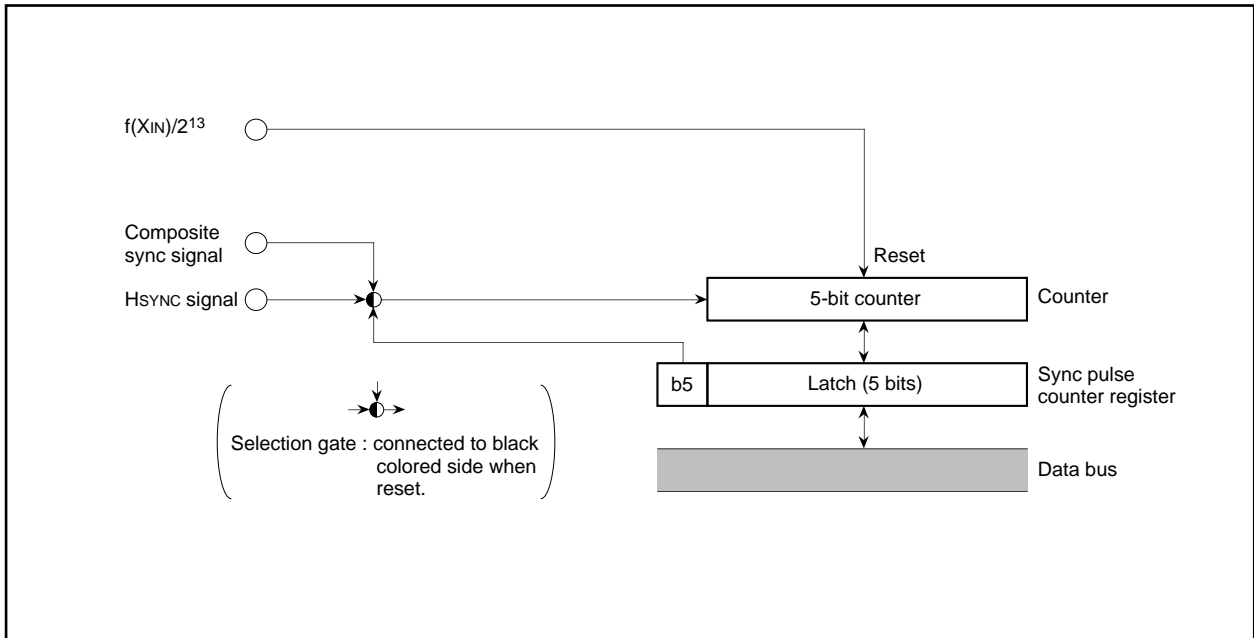


Fig. 41. Synchronous Signal Counter Block Diagram

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MULTI-MASTER I<sup>2</sup>C-BUS INTERFACE**

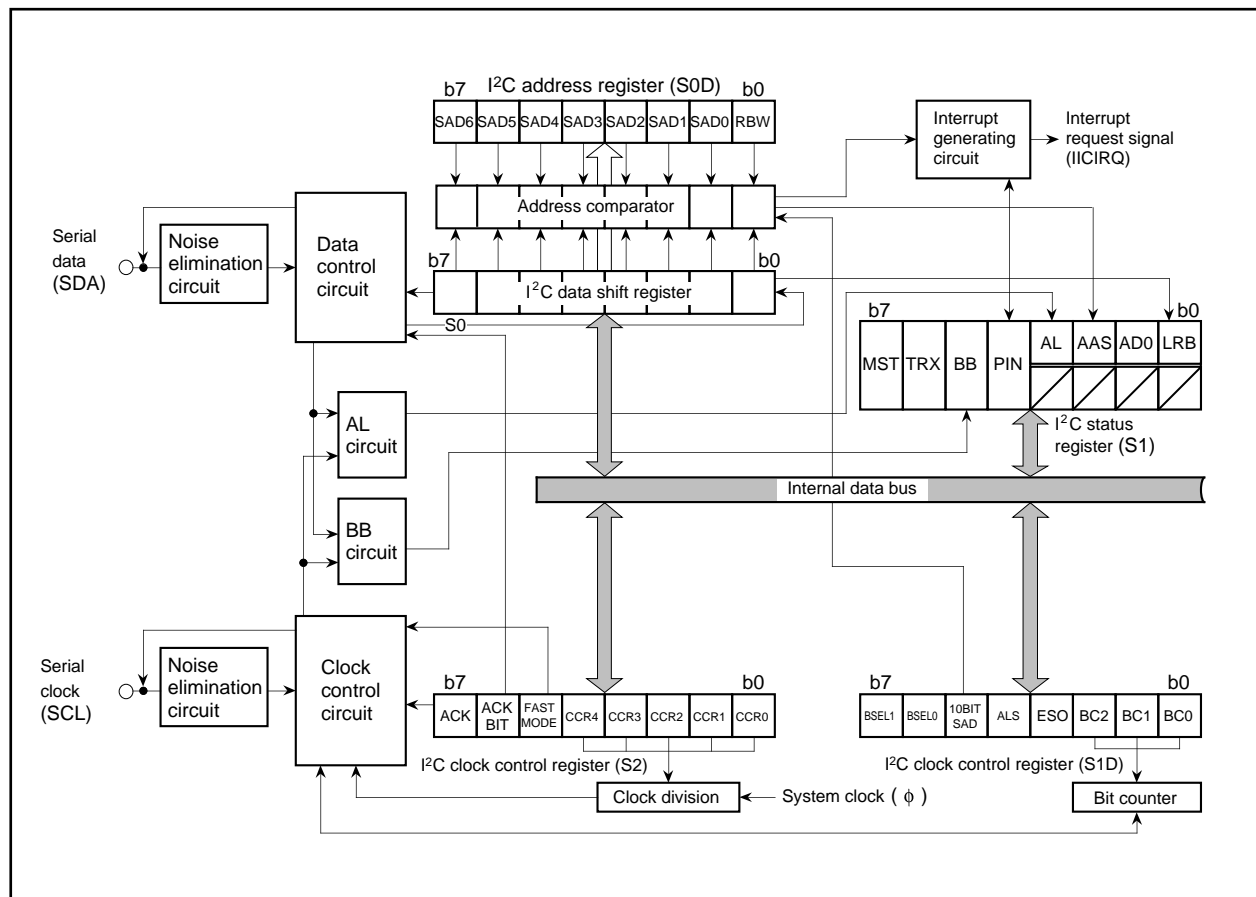
The multi-master I<sup>2</sup>C-BUS interface is a serial communications circuit, conforming to the Philips I<sup>2</sup>C-BUS data transfer format. This interface, offering both arbitration lost detection and a synchronous functions, is useful for the multi-master serial communications. Figure 42 shows a block diagram of the multi-master I<sup>2</sup>C-BUS interface and Table 7 shows multi-master I<sup>2</sup>C-BUS interface functions. This multi-master I<sup>2</sup>C-BUS interface consists of the I<sup>2</sup>C address register, the I<sup>2</sup>C data shift register, the I<sup>2</sup>C clock control register, the I<sup>2</sup>C status register and other control circuits.

**Table 7. Multi-master I<sup>2</sup>C-BUS Interface Functions**

Item	Function
Format	In conformity with Philips I <sup>2</sup> C-BUS standard: 10-bit addressing format 7-bit addressing format High-speed clock mode Standard clock mode
Communication mode	In conformity with Philips I <sup>2</sup> C-BUS standard: Master transmission Master reception Slave transmission Slave reception
SCL clock frequency	16.1 kHz to 400 kHz (at $\phi = 4$ MHz)

$\phi$  : System clock =  $f(XIN)/2$

**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 I<sup>2</sup>C control register at address 00F916) for connections between the I<sup>2</sup>C-BUS interface and ports (SCL1, SCL2, SDA1, SDA2).



**Fig. 42. Block Diagram of Multi-master I<sup>2</sup>C-BUS Interface**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**(1) I<sup>2</sup>C Data Shift Register**

The I<sup>2</sup>C data shift register (S0 : address 00F616) 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 I<sup>2</sup>C data shift register is in a write enable status only when the ESO bit of the I<sup>2</sup>C control register (address 00F916) is "1." The bit counter is reset by a write instruction to the I<sup>2</sup>C data shift register. When both the ESO bit and the MST bit of the I<sup>2</sup>C status register (address 00F816) are "1," the SCL is output by a write instruction to the I<sup>2</sup>C data shift register. Reading data from the I<sup>2</sup>C data shift register is always enabled regardless of the ESO bit value.

**Note:** To write data into the I<sup>2</sup>C data shift register after setting the MST bit to "0" (slave mode), keep an interval of 8 machine cycles or more.

**(2) I<sup>2</sup>C Address Register**

The I<sup>2</sup>C address register (address 00F716) consists of a 7-bit slave address and a 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: 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 I<sup>2</sup>C 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.

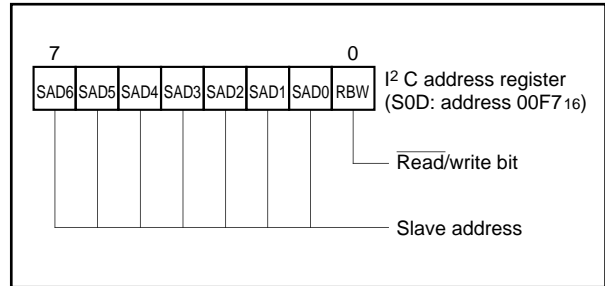


Fig. 43. I<sup>2</sup>C Address Register

**(3) I<sup>2</sup>C Clock Control Register**

The I<sup>2</sup>C clock control register (address 00FA16) 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 7.

■ 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).

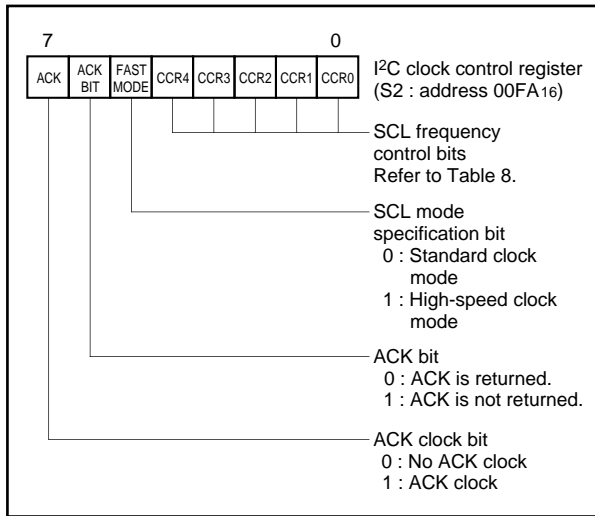
\*ACK clock: Clock for acknowledgement

■ 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.

**Note:** Do not write data into the I<sup>2</sup>C clock control register during transmission. If data is written during transmission, the I<sup>2</sup>C clock generator is reset, so that data cannot be transmitted normally.

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.



**Fig. 44. I<sup>2</sup>C Clock Control Register**

**Table 8. Set Values of I<sup>2</sup>C Clock Control Register and SCL Frequency**

Setting value of CCR4-CCR0					SCL frequency (at $\phi = 4\text{MHz}$ , unit : kHz)	
CCR4	CCR3	CCR2	CCR1	CCR0	Standard clock mode	High-speed clock mode
0	0	0	0	0	Setup disabled	Setup disabled
0	0	0	0	1	Setup disabled	Setup disabled
0	0	0	1	0	Setup disabled	Setup disabled
0	0	0	1	1	Setup disabled	333
0	0	1	0	0	Setup disabled	250
0	0	1	0	1	100	400(Note)
0	0	1	1	0	83.3	166
⋮	⋮	⋮	⋮	⋮	500/CCR value	1000/CCR value
1	1	1	0	1	17.2	34.5
1	1	1	1	0	16.6	33.3
1	1	1	1	1	16.1	32.3

**Note:** At 400 kHz in the high-speed clock mode, the duty is as below.

LOW period : HIGH period = 3 : 2  
 In the other cases, the duty is 50%.

**(4) I<sup>2</sup>C Control Register**

The I<sup>2</sup>C control register (address 00F916) 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 "0002" and the address data is always transmitted and received in 8 bits.

■ Bit 3: I<sup>2</sup>C Interface Use Enable Bit (ESO)

This bit enables usage of the multimaster I<sup>2</sup>C 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 I<sup>2</sup>C status register at address 00F816).

● Writing data to the I<sup>2</sup>C data shift register (address 00F616) is disabled.

■ Bit 4: Data Format Selection Bit (ALS)

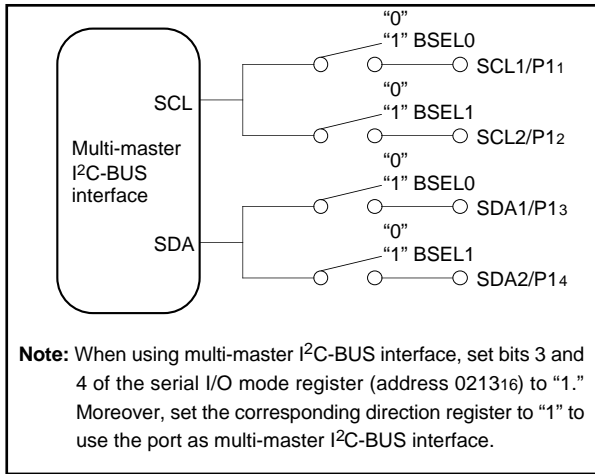
This bit decides whether or not to recognize slave addresses. When this bit is set to "0," the addressing format is selected, so that address data is recognized. When a match is found between a slave address and address data as a result of comparison or when a general call (refer to "(5) I<sup>2</sup>C 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 I<sup>2</sup>C address register (address 00F716) are compared with address data. When this bit is set to "1," the 10-bit addressing format is selected, all the bits of the I<sup>2</sup>C address register are compared with address data.

■ Bits 6 and 7: Connection Control Bits between I<sup>2</sup>C-BUS Interface and Ports (BSEL0, BSEL1)

These bits controls the connection between SCL and ports or SDA and ports (refer to Figure 46).



**Fig. 45. Connection Port Control by BSEL0 and BSEL1**

**(5) I<sup>2</sup>C Status Register**

The I<sup>2</sup>C status register (address 00F8<sub>16</sub>) controls the I<sup>2</sup>C-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 I<sup>2</sup>C data shift register (address 00F6<sub>16</sub>).

■ Bit 1: General Call Detecting Flag (AD0)

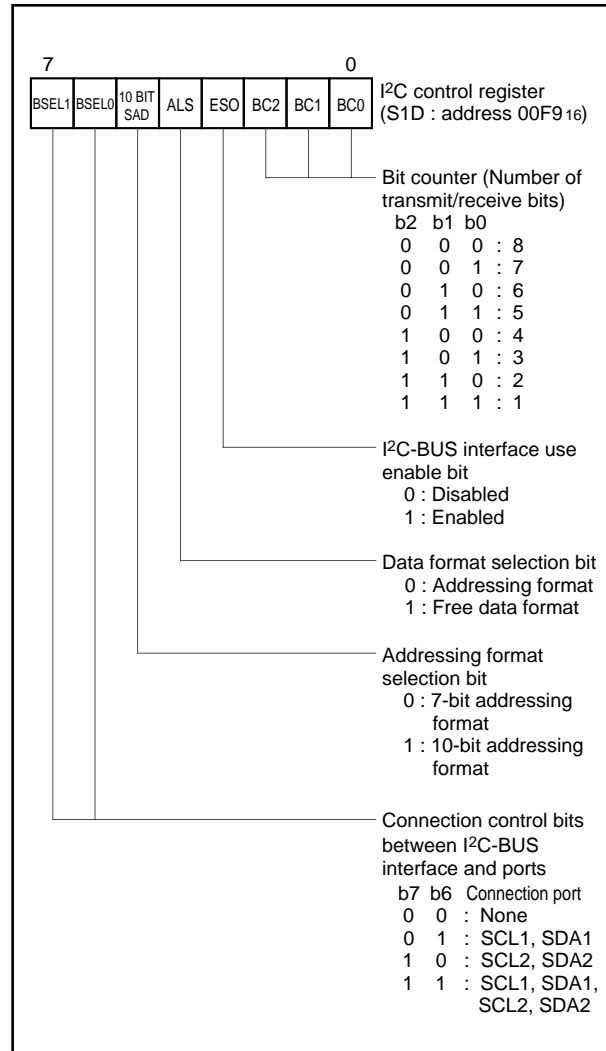
This bit is set to "1" when a general call\* whose address data is all "0" is received in the slave mode. By a general call of the master device, every slave device receives control data after the general call. The AD0 bit is set to "0" by detecting the STOP condition or START condition.

\*General call: The master transmits the general call address "00<sub>16</sub>" to all slaves.

■ Bit 2: Slave Address Comparison Flag (AAS)

This flag indicates a comparison result of address data.

- ① In 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 I<sup>2</sup>C address register (address 00F7<sub>16</sub>).
  - A general call is received.
- ② In 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 I<sup>2</sup>C address register (8 bits consists of slave address and RBW), the first bytes match.
- ③ The state of this bit is changed from "1" to "0" by executing a write instruction to the I<sup>2</sup>C data shift register (address 00F6<sub>16</sub>).



**Fig. 46. I<sup>2</sup>C Control Register**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

■ Bit 3: Arbitration Lost\* Detecting Flag (AL)

In the master transmission mode, when a device other than the microcomputer sets the SDA to "L", 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: I<sup>2</sup>C-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 40 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 I<sup>2</sup>C data shift register (address 00F616).
- 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 I<sup>2</sup>C control register (address 00F916) 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 I<sup>2</sup>C control register (address 00F916) is "0" in the slave reception mode is selected, the TRX bit is set to "1" (transmit) if the least significant bit ( $\overline{R/W}$  bit) of the address data trans-

mitted by the master is "1." When the ALS bit is "0" and the  $\overline{R/W}$  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

**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.

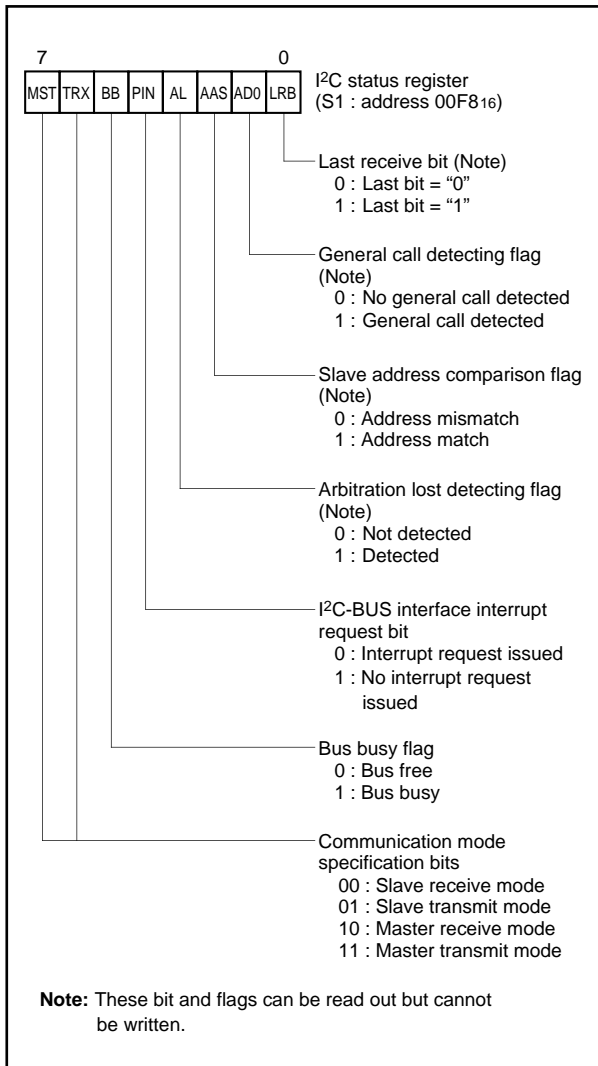


Fig. 47. I<sup>2</sup>C Status Register

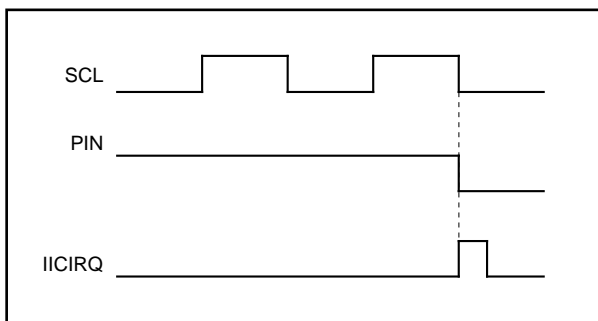


Fig. 48. Interrupt Request Signal Generation Timing

**(6) START Condition Generation Method**

When the ESO bit of the I<sup>2</sup>C control register (address 00F916) is "1," execute a write instruction to the I<sup>2</sup>C status register (address 00F816) to set the MST, TRX and BB bits to "1." A START condition will then be generated. After that, the bit counter becomes "0002" and an SCL for 1 byte is output. The START condition generation timing and BB bit set timing are different in the standard clock mode and the high-speed clock mode. Refer to Figure 49 for the START condition generation timing diagram, and Table 9 for the START condition/STOP condition generation timing table.

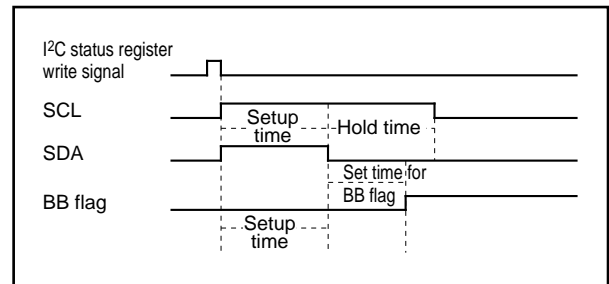


Fig. 49. START Condition Generation Timing Diagram

**(7) RESTART Condition Generation Method**

To generate the RESTART condition, take the following sequence:

- ① Set "2016" to the I<sup>2</sup>C status register (S1).
- ② Write a transmit data to the I<sup>2</sup>C data shift register.
- ③ Set "F016" to the I<sup>2</sup>C status register (S1) again.

**<Example of Setting of RESTART Condition>**

I<sup>2</sup>C status register ; S1 = 2016  
 I<sup>2</sup>C data shift register ; S0 = transmit data after restart  
 I<sup>2</sup>C status register ; S1 = F016



**(8) STOP Condition Generation Method**

When the ES0 bit of the I<sup>2</sup>C control register (address 00F916) is "1," execute a write instruction to the I<sup>2</sup>C status register (address 00F816) 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 50 for the STOP condition generation timing diagram, and Table 9 for the START condition/STOP condition generation timing table.

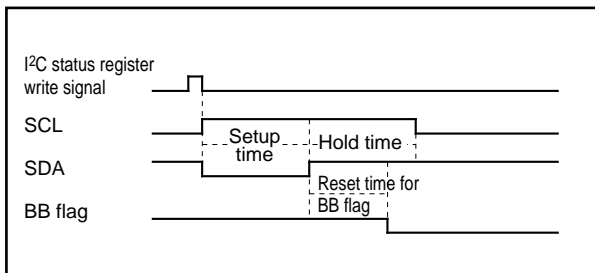


Fig. 50. STOP Condition Generation Timing Diagram

**(9) START/STOP Condition Detect Conditions**

The START/STOP condition detect conditions are shown in Figure 51 and Table 10. 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.

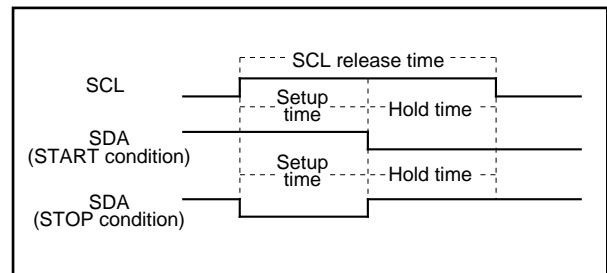


Fig. 51. START Condition/STOP Condition Detect Timing Diagram

Table 9. 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.

Table 10. 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.

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**(10) 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 I<sup>2</sup>C control register (address 00F9<sub>16</sub>) 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 I<sup>2</sup>C address register (address 00F7<sub>16</sub>). At the time of this comparison, address comparison of the RBW bit of the I<sup>2</sup>C address register (address 00F7<sub>16</sub>) is not made. For the data transmission format when the 7-bit addressing format is selected, refer to Figure 52, (1) and (2).

② 10-bit addressing format

To meet the 10-bit addressing format, set the 10BIT SAD bit of the I<sup>2</sup>C control register (address 00F9<sub>16</sub>) 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 I<sup>2</sup>C address register (address 00F7<sub>16</sub>). At the time of this comparison, an address comparison between the RBW bit of the I<sup>2</sup>C address register (address 00F7<sub>16</sub>) and the R/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/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 I<sup>2</sup>C status register (address 00F8<sub>16</sub>) is set to "1." After the second-byte address data is stored into the I<sup>2</sup>C data shift register (address 00F6<sub>16</sub>), make an address comparison between the second-byte data and the slave address by software. When the address data of the 2nd bytes matches the slave address, set the RBW bit of the I<sup>2</sup>C address register (address 00F7<sub>16</sub>) to "1" by software. This processing can match the 7-bit slave address and R/W data, which are received after a RESTART condition is detected, with the value of the I<sup>2</sup>C address register (address 00F7<sub>16</sub>). For the data transmission format when the 10-bit addressing format is selected, refer to Figure 52, (3) and (4).

**(11) Example of Master Transmission**

An example of master transmission in the standard clock mode, at the SCL frequency of 100 kHz and in the ACK return mode is shown below.

- ① Set a slave address in the high-order 7 bits of the I<sup>2</sup>C address register (address 00F7<sub>16</sub>) and "0" in the RBW bit.
- ② Set the ACK return mode and SCL = 100 kHz by setting "85<sub>16</sub>" in the I<sup>2</sup>C clock control register (address 00FA<sub>16</sub>).
- ③ Set "10<sub>16</sub>" in the I<sup>2</sup>C status register (address 00F8<sub>16</sub>) and hold the SCL at the HIGH.
- ④ Set a communication enable status by setting "48<sub>16</sub>" in the I<sup>2</sup>C control register (address 00F9<sub>16</sub>).
- ⑤ Set the address data of the destination of transmission in the high-order 7 bits of the I<sup>2</sup>C data shift register (address 00F6<sub>16</sub>) and set "0" in the least significant bit.
- ⑥ Set "F0<sub>16</sub>" in the I<sup>2</sup>C status register (address 00F8<sub>16</sub>) to generate a START condition. At this time, an SCL for 1 byte and an ACK clock automatically occurs.

- ⑦ Set transmit data in the I<sup>2</sup>C data shift register (address 00F6<sub>16</sub>). At this time, an SCL and an ACK clock automatically occurs.
- ⑧ When transmitting control data of more than 1 byte, repeat step ⑦.
- ⑨ Set "D0<sub>16</sub>" in the I<sup>2</sup>C status register (address 00F8<sub>16</sub>). After this, if ACK is not returned or transmission ends, a STOP condition will be generated.

**(12) Example of Slave Reception**

An example of slave reception in the high-speed clock mode, at the SCL frequency of 400 kHz, in the ACK non-return mode, using the addressing format, is shown below.

- ① Set a slave address in the high-order 7 bits of the I<sup>2</sup>C address register (address 00F7<sub>16</sub>) and "0" in the RBW bit.
- ② Set the no ACK clock mode and SCL = 400 kHz by setting "25<sub>16</sub>" in the I<sup>2</sup>C clock control register (address 00FA<sub>16</sub>).
- ③ Set "10<sub>16</sub>" in the I<sup>2</sup>C status register (address 00F8<sub>16</sub>) and hold the SCL at the HIGH.
- ④ Set a communication enable status by setting "48<sub>16</sub>" in the I<sup>2</sup>C control register (address 00F9<sub>16</sub>).
- ⑤ When a START condition is received, an address comparison is made.
- ⑥ •When all transmitted addresses are "0" (general call) :  
 AD0 of the I<sup>2</sup>C status register (address 00F8<sub>16</sub>) is set to "1" and an interrupt request signal occurs.  
 •When the transmitted addresses match the address set in ①:  
 AAS of the I<sup>2</sup>C status register (address 00F8<sub>16</sub>) is set to "1" and an interrupt request signal occurs.  
 •In the cases other than the above :  
 AD0 and AAS of the I<sup>2</sup>C status register (address 00F8<sub>16</sub>) are set to "0" and no interrupt request signal occurs.
- ⑦ Set dummy data in the I<sup>2</sup>C data shift register (address 00F6<sub>16</sub>).
- ⑧ When receiving control data of more than 1 byte, repeat step ⑦.
- ⑨ When a STOP condition is detected, the communication ends.

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

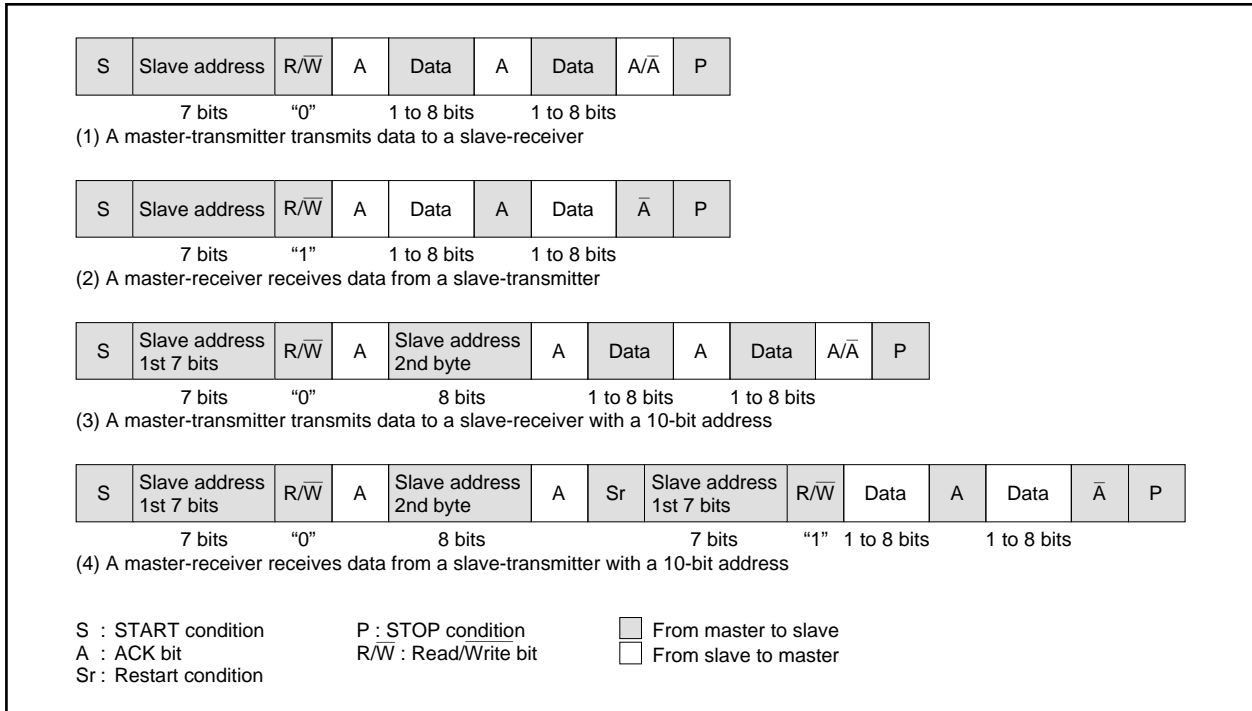


Fig. 52. Address Data Communication Format

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**OSD FUNCTIONS**

Table 11 outlines the OSD functions of the M37271MF-XXXSP.  
 The M37274MA-XXXSP incorporates an OSD circuit of 36 characters X 12 lines. OSD is controlled by the OSD control register. There are 3 display modes and they are selected by a block unit. The display modes are selected by block control register i (i = 1 to 12). The features of each mode are described below.

**Note :** Note that EPROM version has 40 characters X 16 lines when programming.

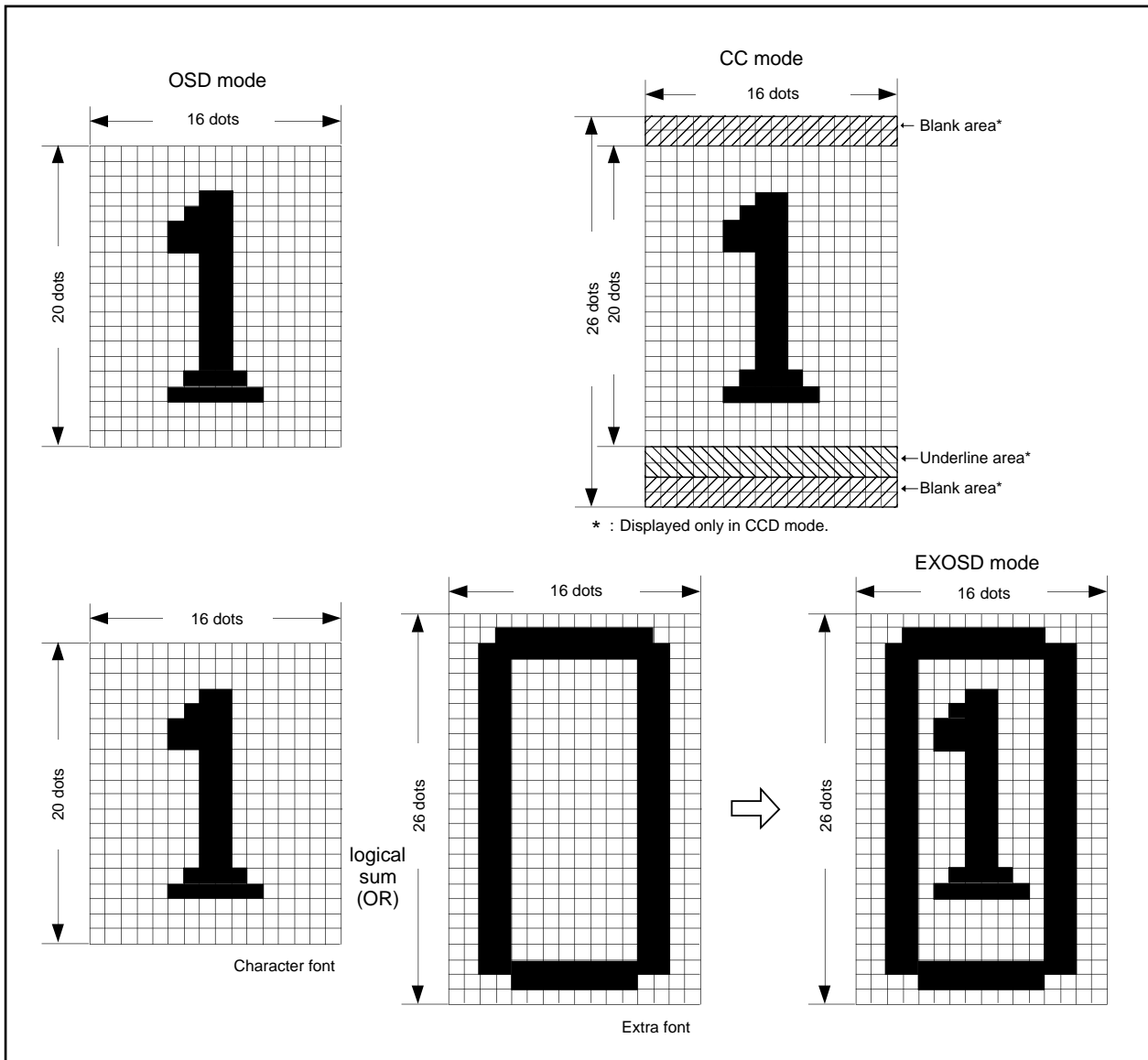
**Table 11. Features of Each Display Mode**

Parameter	Display Mode		
	CC Mode (Closed caption mode)	OSD Mode (On-screen display mode)	EXOSD Mode (Extra on-screen display mode)
Number of display characters	36 characters X 12 lines	36 characters X 12 lines	36 characters X 12 lines
Character display area	16 X 26 dots (character dot structure : 20 X 16 dots)	16 X 20 dots	16 X 26 dots
Kinds of characters	256 kinds (In EXOSD mode, they can be combined with 16 kinds of extra fonts)		
Kinds of character sizes	2 kinds	14 kinds	6 kinds
Pre-divide ratio (Note)	X 1, X 2	X 1, X 2, X 3	X 1, X 2, X 3
Dot size	1Tc X 1/2H	1Tc X 1/2H, 1Tc X 1H, 1.5Tc X 1/2H, 1.5Tc X 1H, 2Tc X 2H, 3Tc X 3H	1Tc X 1/2H, 1Tc X 1H
Attribute	Smooth italic, under line, flash	Border	Border, extra font (16 kinds)
Character font coloring	1 screen : 7 kinds, Max. 7 kinds (a character unit)	1 screen : 7 kinds, Max. 7 kinds (a character unit)	1 screen : 5 kinds, Max. 5 kinds (a character unit)
Raster coloring	Possible (a screen unit, 1 screen : 7 kinds, max. 7 kinds)	Possible (a screen unit, 1 screen : 7 kinds, max. 7 kinds)	Possible (a screen unit, 1 screen : 7 kinds, max. 7 kinds)
Character background coloring	Possible (a character unit, 1 screen : 7 kinds, max. 7 kinds)	Possible (a character unit, 1 screen : 7 kinds, max. 7 kinds)	Possible (a character unit, 1 screen : 7 kinds, max. 7 kinds)
Border coloring	_____	Possible (a screen unit, 1 screen : 7 kinds, max. 7 kinds)	Possible (a screen unit, 1 screen : 7 kinds, max. 7 kinds)
Extra font coloring	_____	_____	Possible (a screen unit, 1 screen : 7 kinds, max. 7 kinds)
OSD output	R, G, B, OUT1, OUT2	R, G, B, OUT1, OUT2	R, G, B, OUT1, OUT2
Function	Auto solid space function Window function Dual layer OSD function (layer 1)	Dual layer OSD function (layer 2)	_____
Display expansion (multiline display)	Possible	Possible	Possible

**Notes 1:** The divide ratio of the frequency divider (the pre-divide circuit) is referred as "pre-divide ratio" hereafter.  
**2:** The character size is specified with dot size and pre-divide ratio (refer to (3) Dote size).

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

The OSD circuit has an extended display mode. This mode allows multiple lines (16 lines or more) to be displayed on the screen by interrupting the display each time one line is displayed and rewriting data in the block for which display is terminated by software. Figure 53 shows the configuration of OSD character. Figure 54 shows the block diagram of the OSD circuit. Figure 55 shows the structure of the OSD control register. Figure 56 shows the structure of the block control register.



**Fig. 53. Configuration of OSD Character Display Area**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

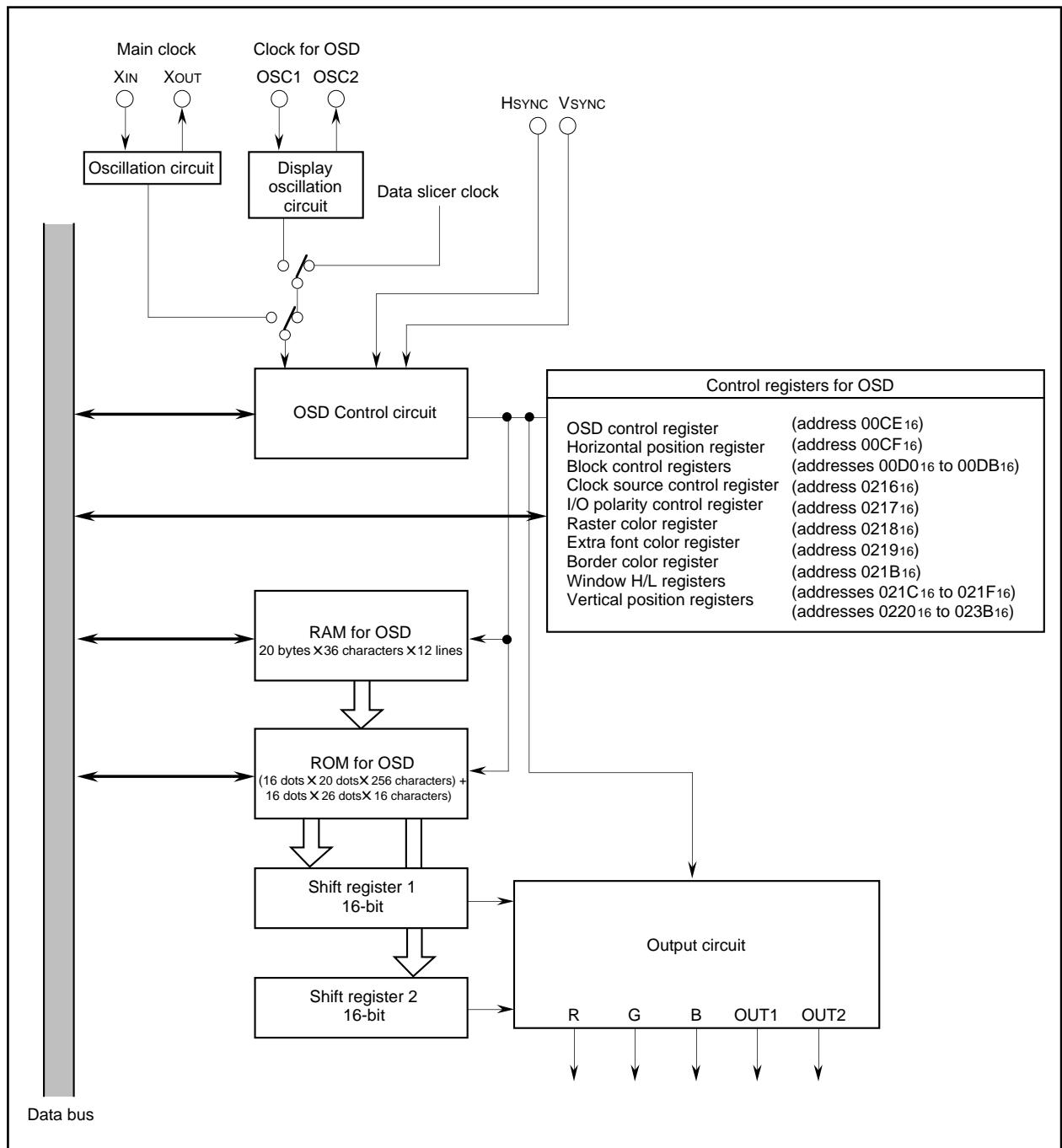
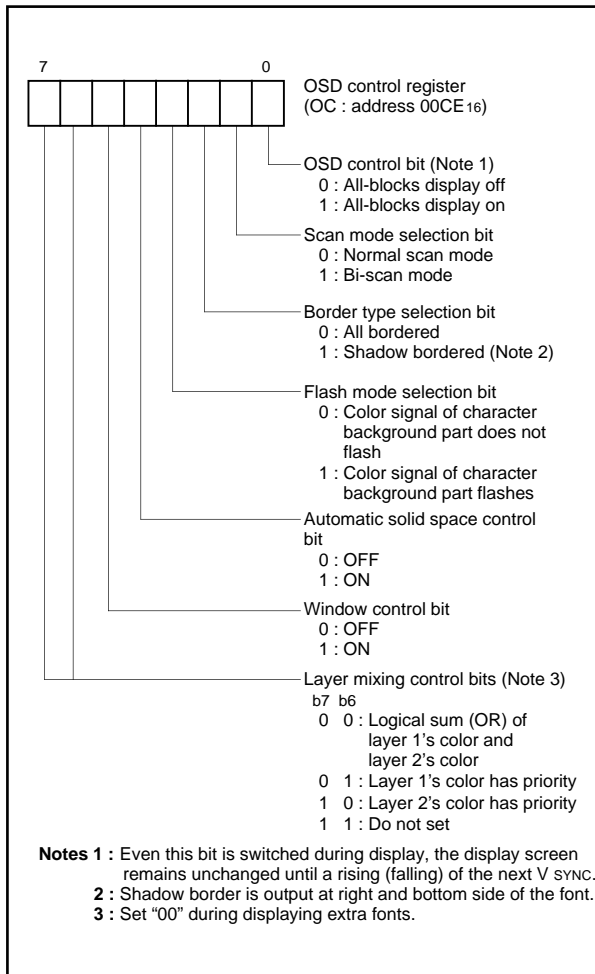


Fig. 54. Block Diagram of OSD Circuit

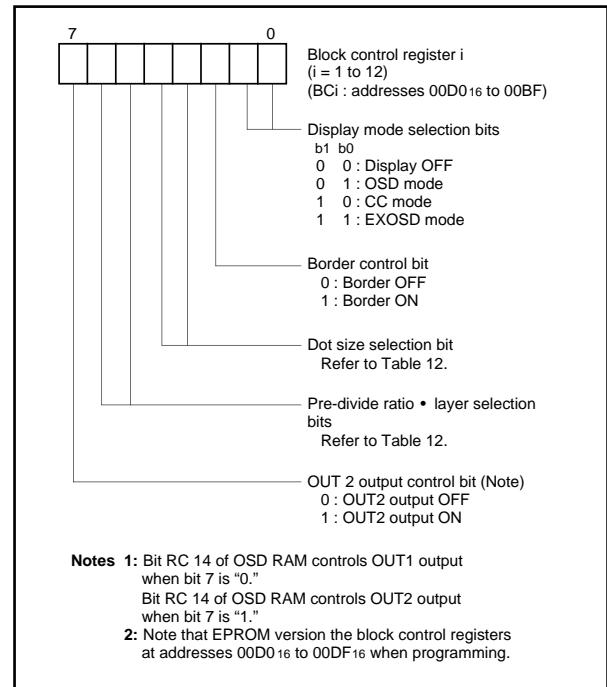
**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

**SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER**



**Fig. 55. OSD Control Register**



**Fig. 56. Block Control Registers**

**Table 12. Setting Value of Block Control Registers**

b6	b5	b4	b3	CS6	Pre-divide Ratio	Dot Size	Display Layer
0	0	0	0	—	X 1	1Tc X 1/2H	Layer 1
		0	1			1Tc X 1H	
		1	0			2Tc X 2H	
		1	1			3Tc X 3H	
0	1	0	0	—	X 2	1Tc X 1/2H	
		0	1			1Tc X 1H	
		1	0			2Tc X 2H	
		1	1			3Tc X 3H	
1	0	0	0	—	X 3	1Tc X 1/2H	
		0	1			1Tc X 1H	
		1	0			2Tc X 2H	
		1	1			3Tc X 3H	
1	1	—	0	0	X 1	1Tc X 1/2H	Layer 2
		—	1			1Tc X 1H	
1	1	0	0	1	X 2	1Tc X 1/2H	
		0	1			1Tc X 1H	
		1	0			1.5Tc X 1/2H	
		1	1			1.5Tc X 1H	

**Notes 1 :** CS6 : Bit 6 of clock control register (Address 021616)

**2 :** Tc : OSD clock cycle divided in the pre-divide circuit

**3 :** H : Hsync

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

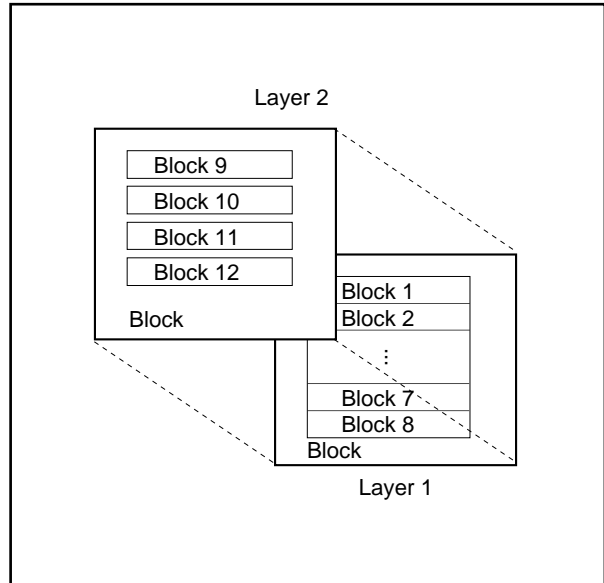
**(1) Dual Layer OSD**

M37274MA-XXXSP has 2 layers; layer 1 and layer 2. These layers display the OSD for controlling TV and the closed caption display at the same time and overlaid on each other.

Each block can be assigned to either layer by bits 6 and 5 of the block control register (refer to Figure 56). For example, only when both bits 5 and 6 are "1," the block is assigned to layer 2. Other bit combinations assign the block to layer 1.

When a block of layer 1 is overlapped with that of layer 2, a screen is combined (refer to Figure 58) by bits 7 and 6 of the OSD control register (refer to Figure 55).

**Note:** When using the dual layer OSD, note Table 13.

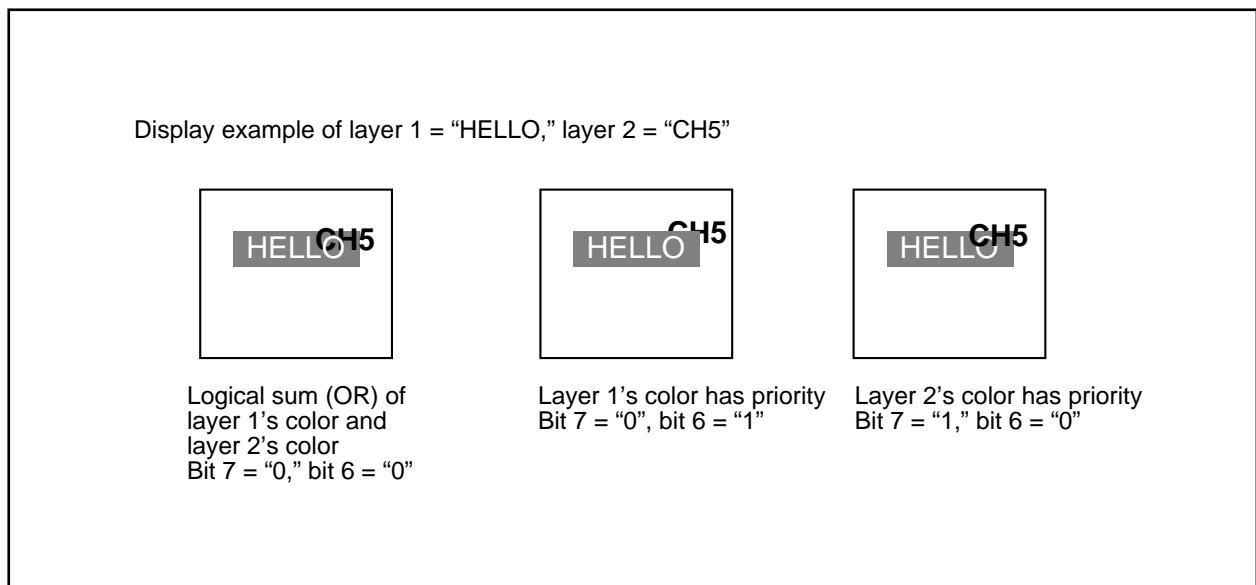


**Fig. 57. Image of Dual Layer OSD**

**Table 13. Conditions of Dual Layer**

Parameter \ Block	Block in Layer 1	Block in Layer 2	
Display mode	CC mode	OSD mode	
OSD Clock source	Data slicer clock or OSC1 or main clock	Same as layer 1	
Pre-divide ratio	X 1 or X 2 (all blocks)	Same as layer 1 (Note)	
Dot size	1Tc X 1/2H	Pre-divide ratio = 1	Pre-divide ratio = 2
		1Tc X 1/2H	1Tc X 1/2H, 1.5Tc X 1/2H
		1Tc X 1H	1Tc X 1H, 1.5Tc X 1H
Horizontal display start position	Arbitrary	Same position as layer 1	

**Note:** For the pre-divide ratio of the layer 2, select the same as the layer 1's ratio by bit 6 of the clock control register.



**Fig. 58. Display Example of Dual Layer OSD**



**(2) Display Position**

The display positions of characters are specified by a block. There are 12 blocks, blocks 1 to 12. Up to 36 characters can be displayed in each block (refer to (6) Memory for OSD).

The display position of each block can be set in both horizontal and vertical directions by software.

The display position in the horizontal direction can be selected for all blocks in common from 256-step display positions in units of 4 T<sub>osc</sub> (T<sub>osc</sub> = OSD oscillation cycle).

The display position in the vertical direction for each block can be selected from 1024-step display positions in units of 1 T<sub>H</sub> (T<sub>H</sub> = H<sub>sync</sub> cycle).

Blocks are displayed in conformance with the following rules:

- ① When the display position is overlapped with another block (Figure 59, (b)), a lower block number (1 to 12) is displayed on the front.
- ② When another block display position appears while one block is displayed (Figure 59 (c)), the block with a larger set value as the vertical display start position is displayed. However, do not display block with the dot size of 2T<sub>c</sub> X 2H or 3T<sub>c</sub> X 3H during display period ( \* ) of another block.
- \* In the case of OSD mode block: 20 dots in vertical from the vertical display start position.
- \* In the case of CC or EXOSD mode block: 26 dots in vertical from the vertical display start position.

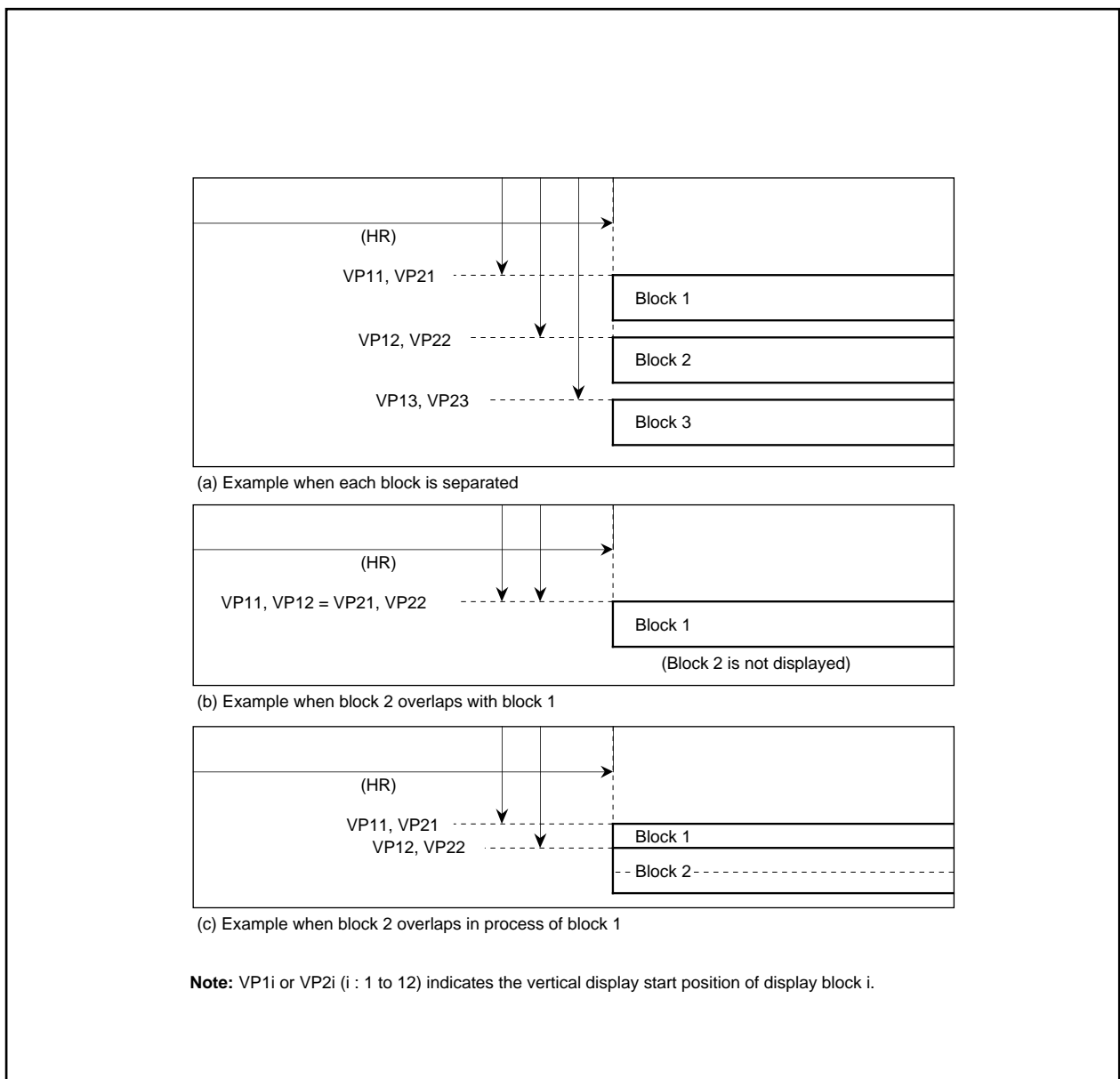
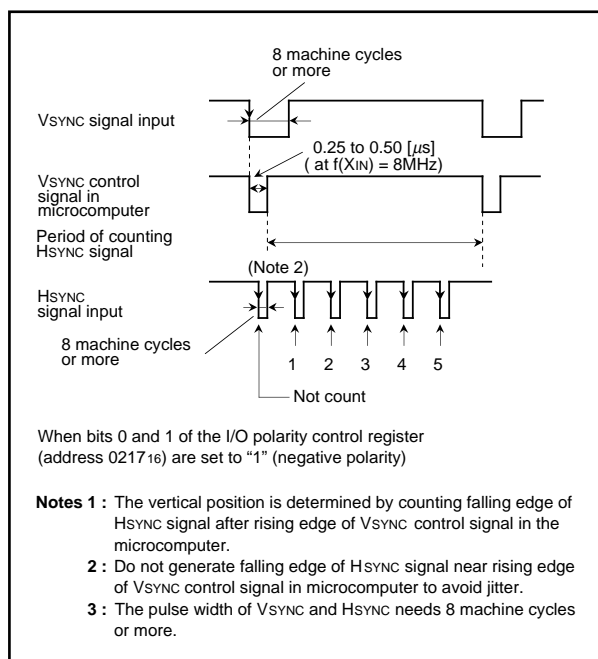


Fig. 59. Display Position

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

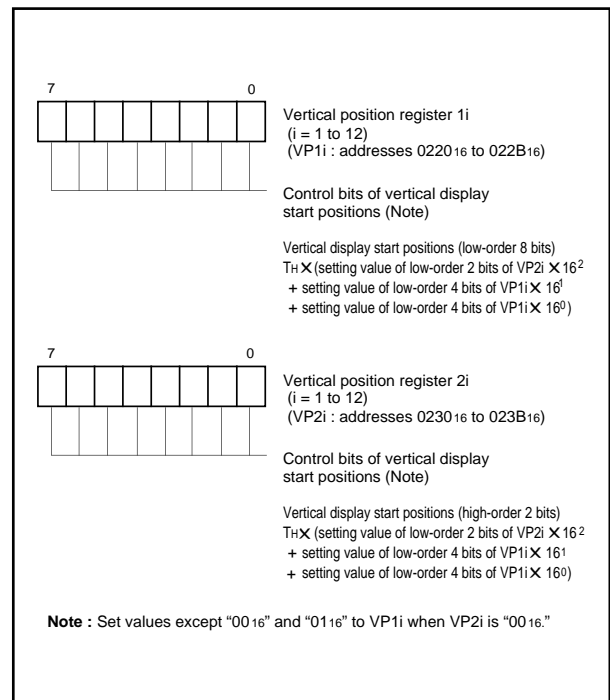
SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

The display position in the vertical direction is determined by counting the horizontal sync signal (HSYNC). At this time, when VSYNC and HSYNC are positive polarity (negative polarity), it starts to count the rising edge (falling edge) of HSYNC signal from after fixed cycle of rising edge (falling edge) of VSYNC signal. So interval from rising edge (falling edge) of VSYNC signal to rising edge (falling edge) of HSYNC signal needs enough time (2 machine cycles or more) for avoiding jitter. The polarity of HSYNC and VSYNC signals can select with the I/O polarity control register (address 0217<sub>16</sub>).



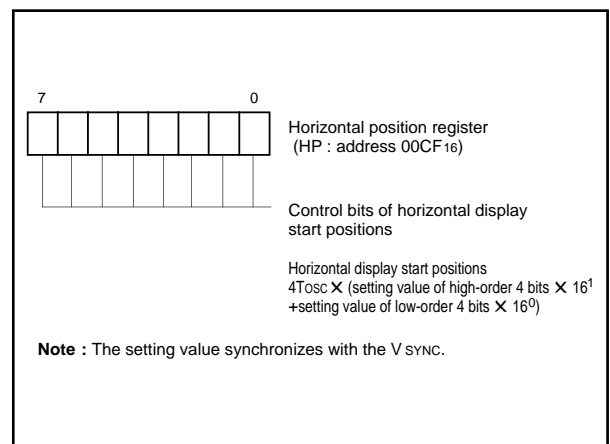
**Fig. 60. Supplement Explanation for Display Position**

The vertical position for each block can be set in 1024 steps (where each step is 1T<sub>H</sub> (T<sub>H</sub>: HSYNC cycle)) as values "00<sub>16</sub>" to "FF<sub>16</sub>" in vertical position register 1i (i = 1 to 12) (addresses 0220<sub>16</sub> to 022B<sub>16</sub>) and values "00<sub>16</sub>" to "03<sub>16</sub>" in vertical position register 2i (i = 1 to 12) (addresses 0230<sub>16</sub> to 023B<sub>16</sub>). The structure of the vertical position registers is shown in Figure 61.



**Fig. 61. Vertical Position Registers**

The horizontal position is common to all blocks, and can be set in 256 steps (where 1 step is 4T<sub>osc</sub>, T<sub>osc</sub> being the oscillating cycle for display) as values "00<sub>16</sub>" to "FF<sub>16</sub>" in bits 0 to 7 of the horizontal position register (address 00CF<sub>16</sub>). The structure of the horizontal position register is shown in Figure 62.



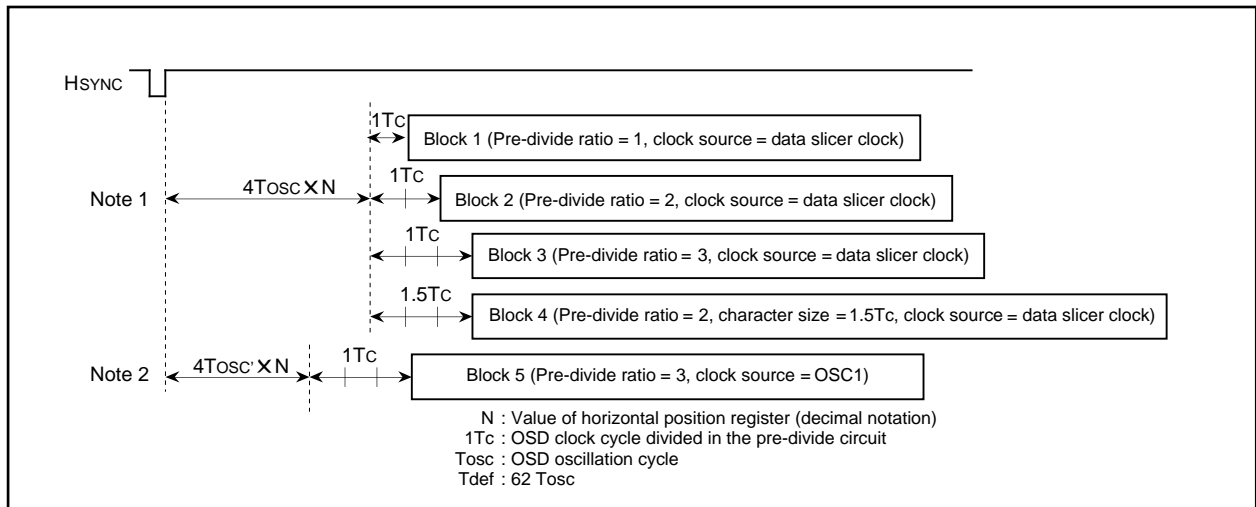
**Fig. 62. Horizontal Position Register**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER**

**Notes 1 :**  $1T_c$  ( $T_c$  : OSD clock cycle divided by prescaler) gap occurs between the horizontal display start position set by the horizontal position register and the most left dot of the 1st block. Accordingly, when 2 blocks have different pre-divide ratios, their horizontal display start position will not match.  
 Ordinarily, this gap is  $1T_c$  regardless of character sizes, however, the gap is  $1.5T_c$  only when the character size is  $1.5T_c$ .

**2 :** The horizontal start position is based on the OSD clock source cycle selected for each block. Accordingly, when 2 blocks have different OSD clock source cycles, their horizontal display start position will not match.



**Fig. 63. Notes on Horizontal Display Start Position**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**(3) Dot Size**

The dot size can be selected by a block unit. The dot size in vertical direction is determined by dividing HSYNC in the vertical dot size control circuit. The dot size in horizontal is determined by dividing the following clock in the horizontal dot size control circuit : the clock gained by dividing the OSD clock source (data slicer clock, OSC1, main clock) in the pre-divide circuit. The clock cycle divided in the pre-divide circuit is defined as 1Tc.

The dot size of the layer 1 is specified by bits 6 to 3 of the block control register.

The dot size of the layer 2 is specified by the following bits : bits 3 and 4 of the block control register, bit 6 of the clock source control register. Refer to Figure 56 (the structure of the block control reg-

ter), refer to Figure 65 (the structure of the clock source control register).

The block diagram of dot size control circuit is shown in Figure 64.

**Notes 1 :** The pre-divide ratio = 3 cannot be used in the CC mode.

**2 :** The pre-divide ratio of the OSD mode block on the layer 2 must be same as that of the CC mode block on the layer 1 by bit 6 of the clock source control register.

**3 :** In the bi-scan mode, the dot size in the vertical direction is 2 times as compared with the normal mode. Refer to "(13) Scan Mode" about the scan mode.

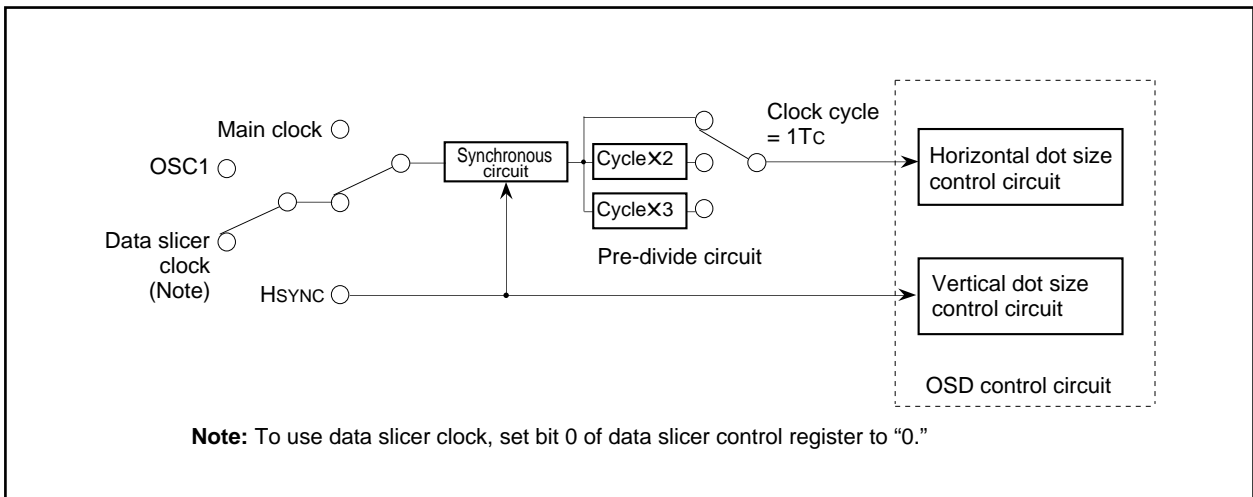


Fig. 64. Block Diagram of Dot Size Control Circuit

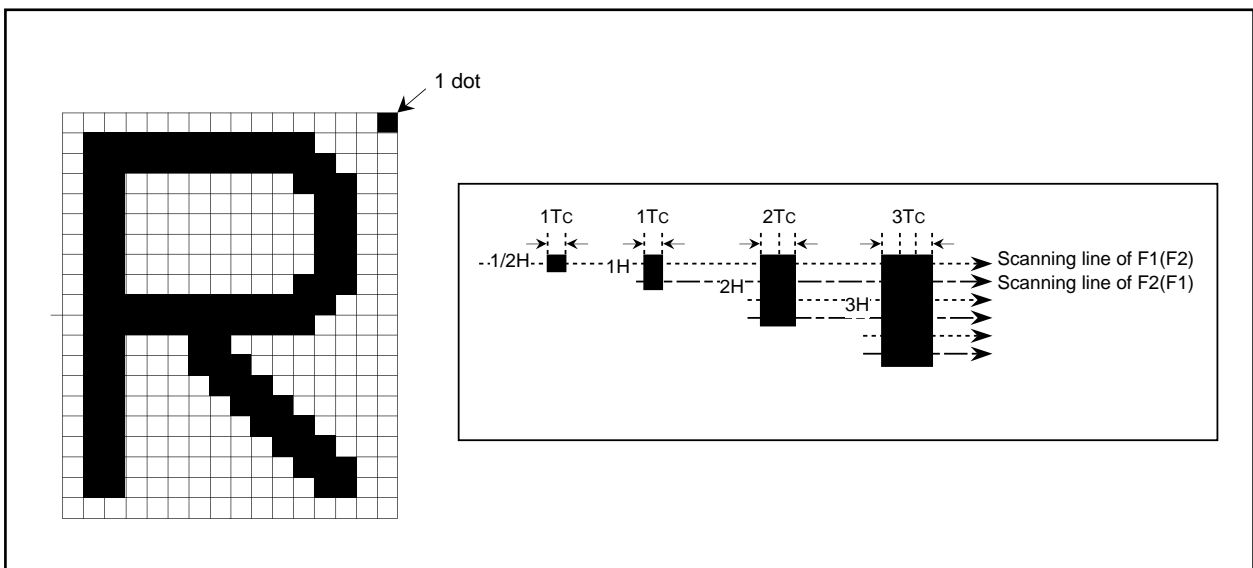


Fig. 65. Definition of Dot Sizes

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**(4) Clock for OSD**

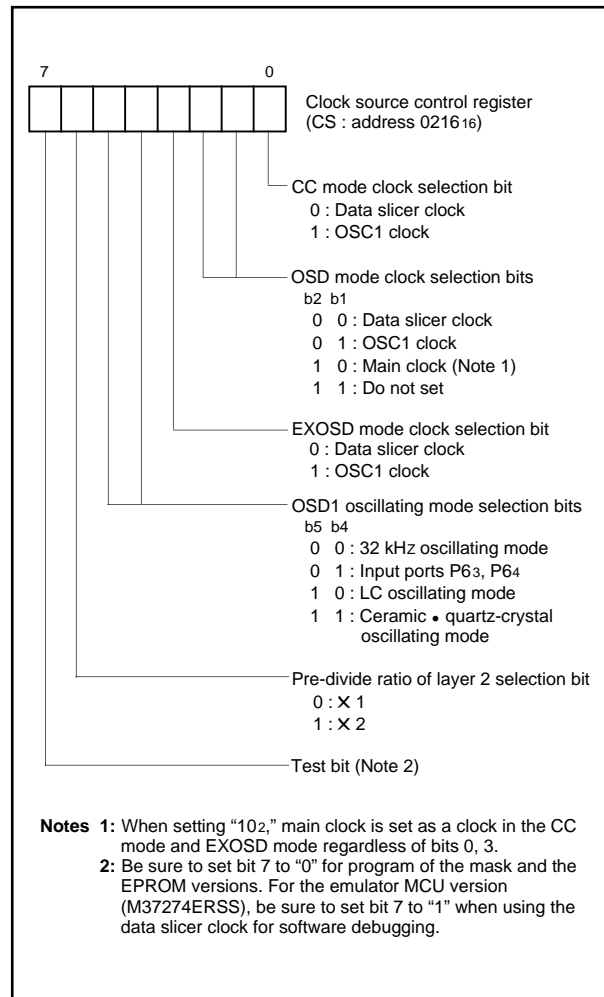
As a clock for display to be used for OSD, it is possible to select one of the following 4 types.

- Main clock (8 MHz)
- Data slicer clock output from the data slicer (approximately 26 MHz)
- Clock from the LC oscillator supplied from the pins OSC1 and OSC2
- Clock from the ceramic resonator or the quartz-crystal oscillator from the pins OSC1 and OSC2

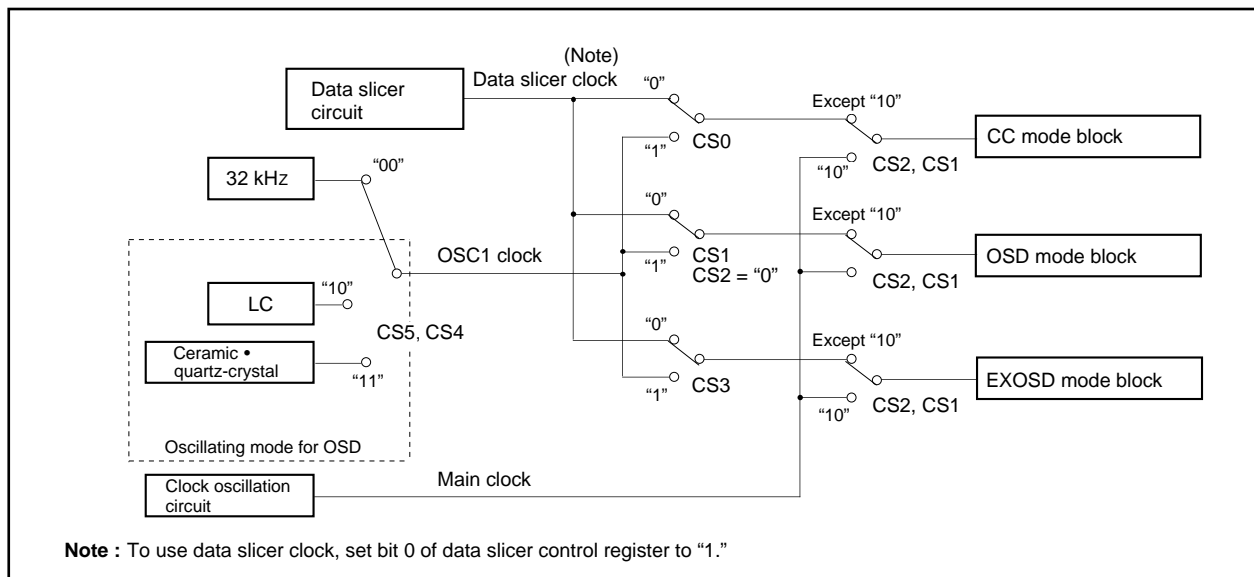
This OSD clock for each block can be selected by the following bits : bit 7 of the port P3 direction register, bits 5 and 4 of the clock source control register (addresses 0216<sub>16</sub>). A variety of character sizes can be obtained by combining dot sizes with OSD clocks. When not using the pins OSC1 and OSC2 for the OSD clock I/O pins, the pins can be used as sub-clock I/O pins or port P6.

**Table 14. Setting for P6<sub>3</sub>/OSC1/X<sub>CIN</sub>, P6<sub>4</sub>/OSC2/X<sub>COUT</sub>**

Register	Function	OSD Clock I/O Pin	Sub-clock I/O Pin	Input Port
b7 of port P3 direction register		0	0	1
Clock source control register	b5	1	1	0
	b4	0	1	1



**Fig. 66. Clock Control Register**



**Fig. 67. Block Diagram of OSD Selection Circuit**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**(5) Field Determination Display**

To display the block with vertical dot size of 1/2H, whether an even field or an odd field is determined through differences in a synchronizing signal waveform of interlacing system. The dot line 0 or 1 (refer to Figure 69) corresponding to the field is displayed alternately.

In the following, the field determination standard for the case where both the horizontal sync signal and the vertical sync signal are negative-polarity inputs will be explained. A field determination is determined by detecting the time from a falling edge of the horizontal sync signal until a falling edge of the VSYNC control signal (refer to Figure 60) in the microcomputer and then comparing this time with the time of the previous field. When the time is longer than the comparing time, it is regarded as even field. When the time is shorter, it is regarded as odd field

The contents of this field can be read out by the field determination flag (bit 7 of the I/O polarity control register at address 021716). A dot line is specified by bit 6 of the I/O polarity control register (refer to Figure 69).

However, the field determination flag read out from the CPU is fixed to "0" at even field or "1" at odd field, regardless of bit 6.

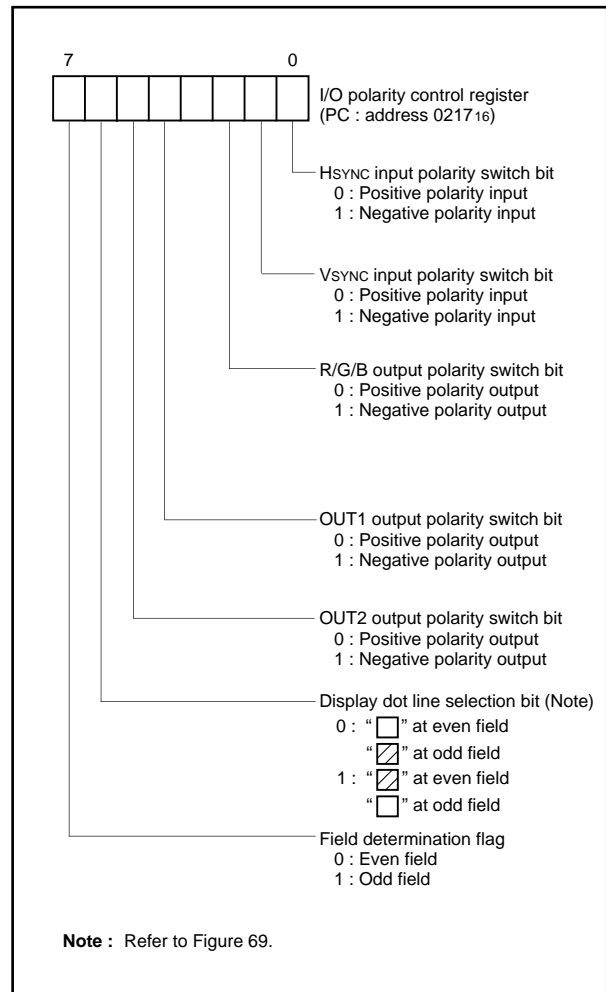


Fig. 68. I/O Polarity Control Register

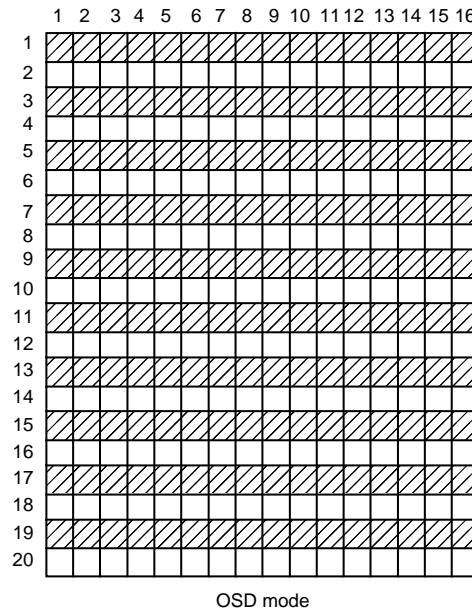
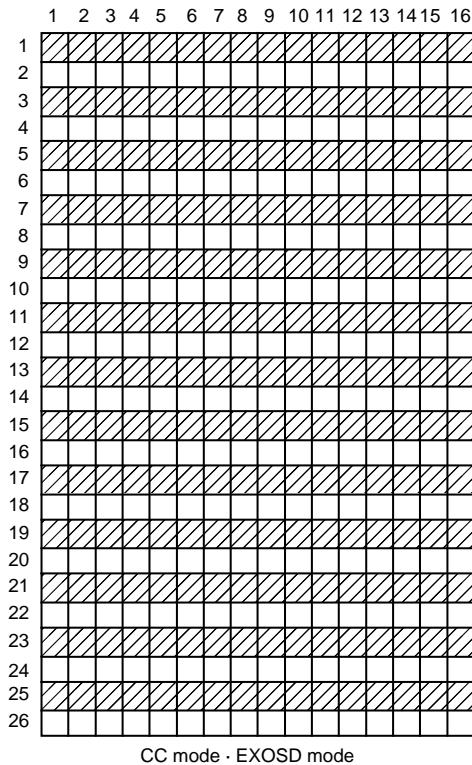
**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER and ON-SCREEN DISPLAY CONTROLLER

Both Hsync signal and Vsync signal are negative-polarity input

Hsync		Field	Field determination flag(Note)	Display dot line selection bit	Display dot line
Vsync and Vsync control signal in micro-computer  Upper : Vsync signal  Lower : Vsync control signal in micro-computer		Odd	/	/	/
		Even	0 (T2 > T1)	0	Dot line 1 <input type="checkbox"/>
		Odd	1 (T3 < T2)	1	Dot line 0 <input checked="" type="checkbox"/>

When using the field determination flag, be sure to set bit 0 of the PWM mode register 1 (address 020A 16) to "0."



When the display dot line selection bit is "0," the "□" font is displayed at even field, the "▨" font is displayed at odd field. Bit 7 of the I/O polarity control register can be read as the field determination flag : "1" is read at odd field, "0" is read at even field.

OSD ROM font configuration diagram

**Note :** The field determination flag changes at a rising edge of the V sync control signal (negative-polarity input) in the microcomputer.

Fig. 69. Relation between Field Determination Flag and Display Font

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**(6) Memory for OSD**

There are 2 types of memory for OSD : ROM for OSD (addresses 10800<sub>16</sub> to 155FF<sub>16</sub>, 18000<sub>16</sub> to 1E41F<sub>16</sub>) used to store character dot data (masked) and RAM for OSD (addresses 0800<sub>16</sub> to 0DF3<sub>16</sub>) used to specify the characters and colors to be displayed. The following describes each type of memory.

① **ROM for OSD (addresses 10800<sub>16</sub> to 155FF<sub>16</sub>, 18000<sub>16</sub> to 1E43F<sub>16</sub>)**

The ROM for OSD contains dot pattern data for characters to be displayed. To actually display the character code and the extra code 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 the ROM for OSD) into the RAM for OSD.

The OSD ROM of the character font has a capacity of 11072 bytes. Since 40 bytes are required for 1 character data, the ROM can store up to 256 kinds of characters. The OSD ROM of the extra font has a capacity of 832 bytes. Since 52 bytes are required for 1 character data, the ROM can store up to 16 kinds of characters. Data of the character font and extra font is specified shown in Figure 70.

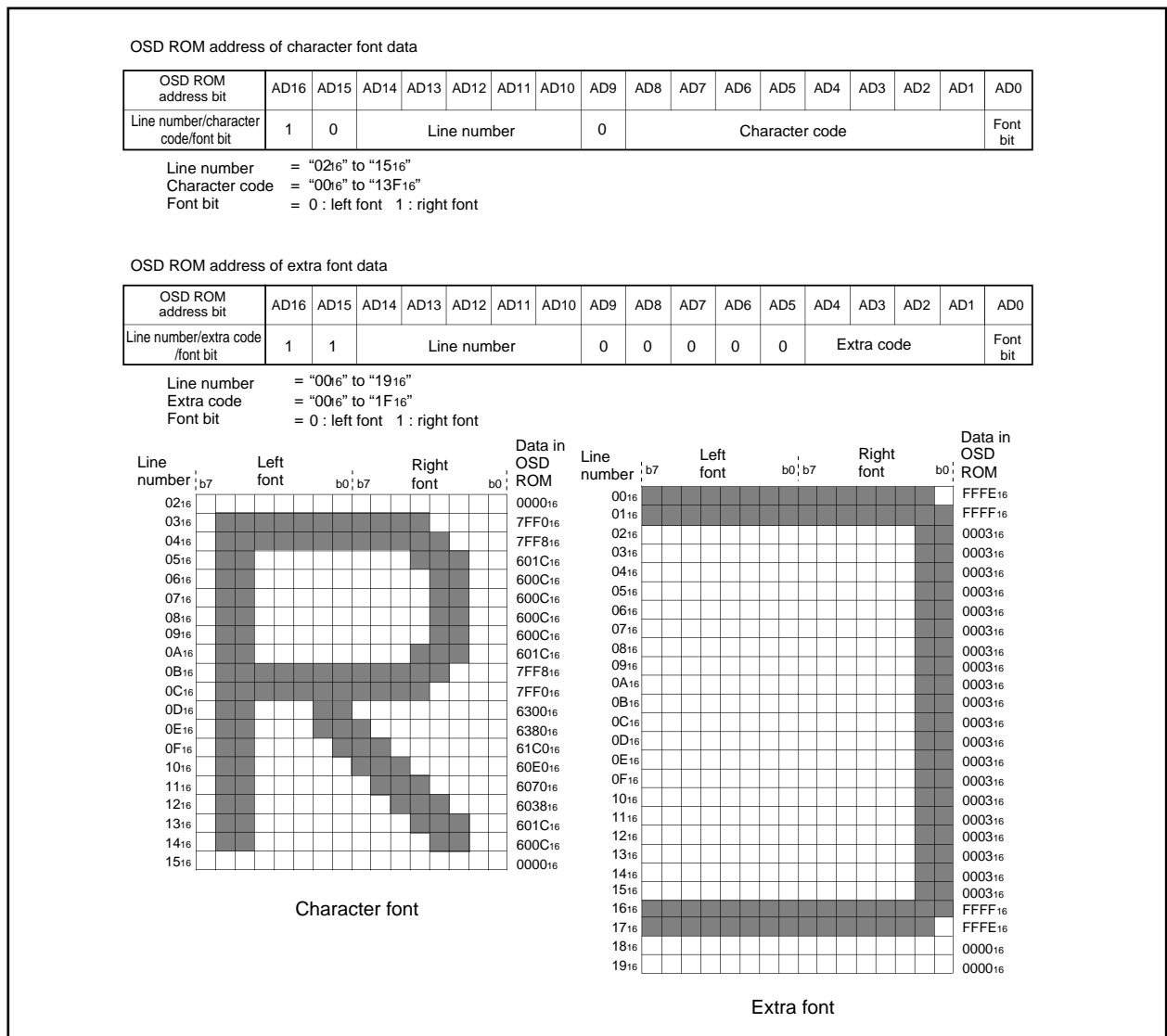


Fig. 70. OSD Character Data Storing Form



**SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER**

**2 RAM for OSD (addresses 0800<sub>16</sub> to 0DF3<sub>16</sub>)**

The RAM for OSD is allocated at addresses 0800<sub>16</sub> to 0DF3<sub>16</sub>, and is divided into a display character code specification part, color code 1 specification part, and color code 2 specification part for each block. Table 15 shows the contents of the RAM for OSD.

For example, to display 1 character position (the left edge) in block 1, write the character code in address 0800<sub>16</sub>, write color code 1 at 0840<sub>16</sub>, and write color code 2 at 0828<sub>16</sub>.

The structure of the RAM for OSD is shown in Figure 72.

**Note:** For the OSD mode block with dot size of 1.5Tc X 1/2H and 1.5Tc X 1H, the 3nth (n = 1 to 13) character is skipped as compared with ordinary block\*. Accordingly, maximum 24 characters are only displayed in 1 block. The RAM data for the 3nth character does not effect the display. Any character data can be stored here (refer to Figure 71).

Note that EPROM version has maximum 27 characters (The right 1/3 part of the 27th's character area is not displayed.) in 1 block when programming.

\* Blocks with dot size of 1Tc X 1/2H and 1Tc X 1H, or blocks on the layer 1

**Table 15. Contents of OSD RAM**

Block	Display Position (from left)	Character Code Specification	Color Code 1 Specification	Color Code 2 Specification
Block 1	1st character	0800 <sub>16</sub>	0840 <sub>16</sub>	0828 <sub>16</sub>
	2nd character	0801 <sub>16</sub>	0841 <sub>16</sub>	0829 <sub>16</sub>
	:	:	:	:
	24th character	0817 <sub>16</sub>	0857 <sub>16</sub>	083F <sub>16</sub>
	25th character	0818 <sub>16</sub>	0858 <sub>16</sub>	0868 <sub>16</sub>
	:	:	:	:
Block 2	35th character	0822 <sub>16</sub>	0862 <sub>16</sub>	0872 <sub>16</sub>
	36th character	0823 <sub>16</sub>	0863 <sub>16</sub>	0873 <sub>16</sub>
	1st character	0880 <sub>16</sub>	08C0 <sub>16</sub>	08A8 <sub>16</sub>
	2nd character	0881 <sub>16</sub>	08C1 <sub>16</sub>	08A9 <sub>16</sub>
	:	:	:	:
	24th character	0897 <sub>16</sub>	08D7 <sub>16</sub>	08BF <sub>16</sub>
Block 3	25th character	0E98 <sub>16</sub>	08D8 <sub>16</sub>	08E8 <sub>16</sub>
	:	:	:	:
	35th character	08A2 <sub>16</sub>	08E2 <sub>16</sub>	08F2 <sub>16</sub>
	36th character	08A3 <sub>16</sub>	08E3 <sub>16</sub>	08F3 <sub>16</sub>
	1st character	0900 <sub>16</sub>	0940 <sub>16</sub>	0928 <sub>16</sub>
	2nd character	0901 <sub>16</sub>	0941 <sub>16</sub>	0929 <sub>16</sub>
Block 4	:	:	:	:
	24th character	0917 <sub>16</sub>	0957 <sub>16</sub>	093F <sub>16</sub>
	25th character	0918 <sub>16</sub>	0958 <sub>16</sub>	0968 <sub>16</sub>
	:	:	:	:
	35th character	0922 <sub>16</sub>	0962 <sub>16</sub>	0972 <sub>16</sub>
	36th character	0923 <sub>16</sub>	0963 <sub>16</sub>	0973 <sub>16</sub>
Block 5	1st character	0980 <sub>16</sub>	09C0 <sub>16</sub>	09A8 <sub>16</sub>
	2nd character	0981 <sub>16</sub>	09C1 <sub>16</sub>	09A9 <sub>16</sub>
	:	:	:	:
	24th character	0997 <sub>16</sub>	09D7 <sub>16</sub>	09BF <sub>16</sub>
	25th character	0998 <sub>16</sub>	08D8 <sub>16</sub>	09E8 <sub>16</sub>
	:	:	:	:
Block 6	35th character	09A2 <sub>16</sub>	09E2 <sub>16</sub>	09F2 <sub>16</sub>
	36th character	09A3 <sub>16</sub>	09E3 <sub>16</sub>	09F3 <sub>16</sub>
	1st character	0A00 <sub>16</sub>	0A40 <sub>16</sub>	0A28 <sub>16</sub>
	2nd character	0A01 <sub>16</sub>	0A41 <sub>16</sub>	0A29 <sub>16</sub>
	:	:	:	:
	24th character	0A17 <sub>16</sub>	0A57 <sub>16</sub>	0A3F <sub>16</sub>
Block 7	25th character	0A18 <sub>16</sub>	0A58 <sub>16</sub>	0A68 <sub>16</sub>
	:	:	:	:
	35th character	0A22 <sub>16</sub>	0A62 <sub>16</sub>	0A72 <sub>16</sub>
	36th character	0A23 <sub>16</sub>	0A63 <sub>16</sub>	0A73 <sub>16</sub>

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

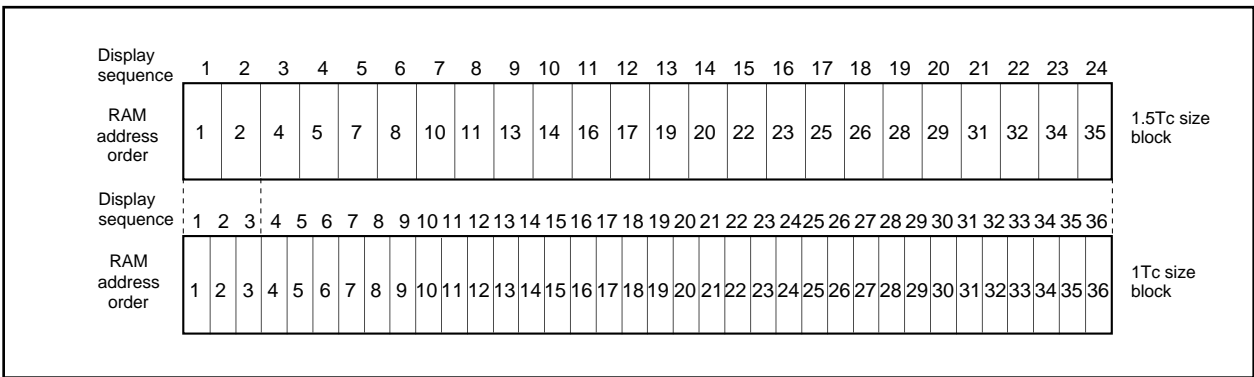
SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**Table 15. Contents of OSD RAM (continued)**

Block	Display Position (from left)	Character Code Specification	Color Code 1 Specification	Color Code 2 Specification
Block 6	1st character	0A80 <sub>16</sub>	0AC0 <sub>16</sub>	0AA8 <sub>16</sub>
	2nd character	0A81 <sub>16</sub>	0AC1 <sub>16</sub>	0AA9 <sub>16</sub>
	⋮	⋮	⋮	⋮
	24th character	0A97 <sub>16</sub>	0AD7 <sub>16</sub>	0ABF <sub>16</sub>
	25th character	0A98 <sub>16</sub>	0AD8 <sub>16</sub>	0AE8 <sub>16</sub>
	⋮	⋮	⋮	⋮
Block 7	35th character	0AA2 <sub>16</sub>	0AE2 <sub>16</sub>	0AF2 <sub>16</sub>
	36th character	0AA3 <sub>16</sub>	0AE3 <sub>16</sub>	0AF3 <sub>16</sub>
	1st character	0B00 <sub>16</sub>	0B40 <sub>16</sub>	0B28 <sub>16</sub>
	2nd character	0B01 <sub>16</sub>	0B41 <sub>16</sub>	0B29 <sub>16</sub>
	⋮	⋮	⋮	⋮
	24th character	0B17 <sub>16</sub>	0B57 <sub>16</sub>	0B3F <sub>16</sub>
Block 8	25th character	0B18 <sub>16</sub>	0B58 <sub>16</sub>	0B68 <sub>16</sub>
	⋮	⋮	⋮	⋮
	35th character	0B22 <sub>16</sub>	0B62 <sub>16</sub>	0B72 <sub>16</sub>
	36th character	0B23 <sub>16</sub>	0B63 <sub>16</sub>	0B73 <sub>16</sub>
	1st character	0B80 <sub>16</sub>	0BC0 <sub>16</sub>	0BA8 <sub>16</sub>
	2nd character	0B81 <sub>16</sub>	0BC1 <sub>16</sub>	0BA9 <sub>16</sub>
Block 9	⋮	⋮	⋮	⋮
	24th character	0B97 <sub>16</sub>	0BD7 <sub>16</sub>	0BBF <sub>16</sub>
	25th character	0B98 <sub>16</sub>	0BD8 <sub>16</sub>	0BE8 <sub>16</sub>
	⋮	⋮	⋮	⋮
	35th character	0BA2 <sub>16</sub>	0BE2 <sub>16</sub>	0BF2 <sub>16</sub>
	36th character	0BA3 <sub>16</sub>	0BE3 <sub>16</sub>	0BF3 <sub>16</sub>
Block 10	1st character	0C00 <sub>16</sub>	0C40 <sub>16</sub>	0C28 <sub>16</sub>
	2nd character	0C01 <sub>16</sub>	0C41 <sub>16</sub>	0C29 <sub>16</sub>
	⋮	⋮	⋮	⋮
	24th character	0C17 <sub>16</sub>	0C57 <sub>16</sub>	0C3F <sub>16</sub>
	25th character	0C18 <sub>16</sub>	0C58 <sub>16</sub>	0C68 <sub>16</sub>
	⋮	⋮	⋮	⋮
Block 11	35th character	0C22 <sub>16</sub>	0C62 <sub>16</sub>	0C72 <sub>16</sub>
	36th character	0C23 <sub>16</sub>	0C63 <sub>16</sub>	0C73 <sub>16</sub>
	1st character	0C80 <sub>16</sub>	0CC0 <sub>16</sub>	0CA8 <sub>16</sub>
	2nd character	0C81 <sub>16</sub>	0CC1 <sub>16</sub>	0CA9 <sub>16</sub>
	⋮	⋮	⋮	⋮
	24th character	0C97 <sub>16</sub>	0CD7 <sub>16</sub>	0CBF <sub>16</sub>
Block 12	25th character	0C98 <sub>16</sub>	0CD8 <sub>16</sub>	0CE8 <sub>16</sub>
	⋮	⋮	⋮	⋮
	35th character	0CA2 <sub>16</sub>	0CE2 <sub>16</sub>	0CF2 <sub>16</sub>
	36th character	0CA3 <sub>16</sub>	0CE3 <sub>16</sub>	0CF3 <sub>16</sub>
	1st character	0D00 <sub>16</sub>	0D40 <sub>16</sub>	0D28 <sub>16</sub>
	2nd character	0D01 <sub>16</sub>	0D41 <sub>16</sub>	0D29 <sub>16</sub>
Block 13	⋮	⋮	⋮	⋮
	24th character	0D17 <sub>16</sub>	0D57 <sub>16</sub>	0D3F <sub>16</sub>
	25th character	0D18 <sub>16</sub>	0D58 <sub>16</sub>	0D68 <sub>16</sub>
	⋮	⋮	⋮	⋮
	35th character	0D22 <sub>16</sub>	0D62 <sub>16</sub>	0D72 <sub>16</sub>
	36th character	0D23 <sub>16</sub>	0D63 <sub>16</sub>	0D73 <sub>16</sub>
Block 14	1st character	0D80 <sub>16</sub>	0DC0 <sub>16</sub>	0DA8 <sub>16</sub>
	2nd character	0D81 <sub>16</sub>	0DC1 <sub>16</sub>	0DA9 <sub>16</sub>
	⋮	⋮	⋮	⋮
	24th character	0D97 <sub>16</sub>	0DD7 <sub>16</sub>	0DBF <sub>16</sub>
	25th character	0D98 <sub>16</sub>	0DD8 <sub>16</sub>	0DE8 <sub>16</sub>
	⋮	⋮	⋮	⋮
Block 15	35th character	0DA2 <sub>16</sub>	0DE2 <sub>16</sub>	0DF2 <sub>16</sub>
	36th character	0DA3 <sub>16</sub>	0DE3 <sub>16</sub>	0DF3 <sub>16</sub>

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER



**Fig. 71. RAM Data for 3nth Character**

**Note:** Do not read from and write to addresses shown in Table 16.

**Table 16. List of Access Disable Addresses**

0878 <sub>16</sub>	0879 <sub>16</sub>	087A <sub>16</sub>
08F8 <sub>16</sub>	08F9 <sub>16</sub>	08FA <sub>16</sub>
0978 <sub>16</sub>	0979 <sub>16</sub>	097A <sub>16</sub>
09F8 <sub>16</sub>	09F9 <sub>16</sub>	09FA <sub>16</sub>
0A78 <sub>16</sub>	0A79 <sub>16</sub>	0A7A <sub>16</sub>
0AF8 <sub>16</sub>	0AF9 <sub>16</sub>	0AFA <sub>16</sub>
0B78 <sub>16</sub>	0B79 <sub>16</sub>	0B7A <sub>16</sub>
0BF8 <sub>16</sub>	0BF9 <sub>16</sub>	0BFA <sub>16</sub>
0C78 <sub>16</sub>	0C79 <sub>16</sub>	0C7A <sub>16</sub>
0CF8 <sub>16</sub>	0CF9 <sub>16</sub>	0CFA <sub>16</sub>
0D78 <sub>16</sub>	0D79 <sub>16</sub>	0D7A <sub>16</sub>
0DF8 <sub>16</sub>	0DF9 <sub>16</sub>	0DFA <sub>16</sub>
0E78 <sub>16</sub>	0E79 <sub>16</sub>	0E7A <sub>16</sub>
0EF8 <sub>16</sub>	0EF9 <sub>16</sub>	0EFA <sub>16</sub>
0F78 <sub>16</sub>	0F79 <sub>16</sub>	0F7A <sub>16</sub>
0FF8 <sub>16</sub>	0FF9 <sub>16</sub>	0FFA <sub>16</sub>

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

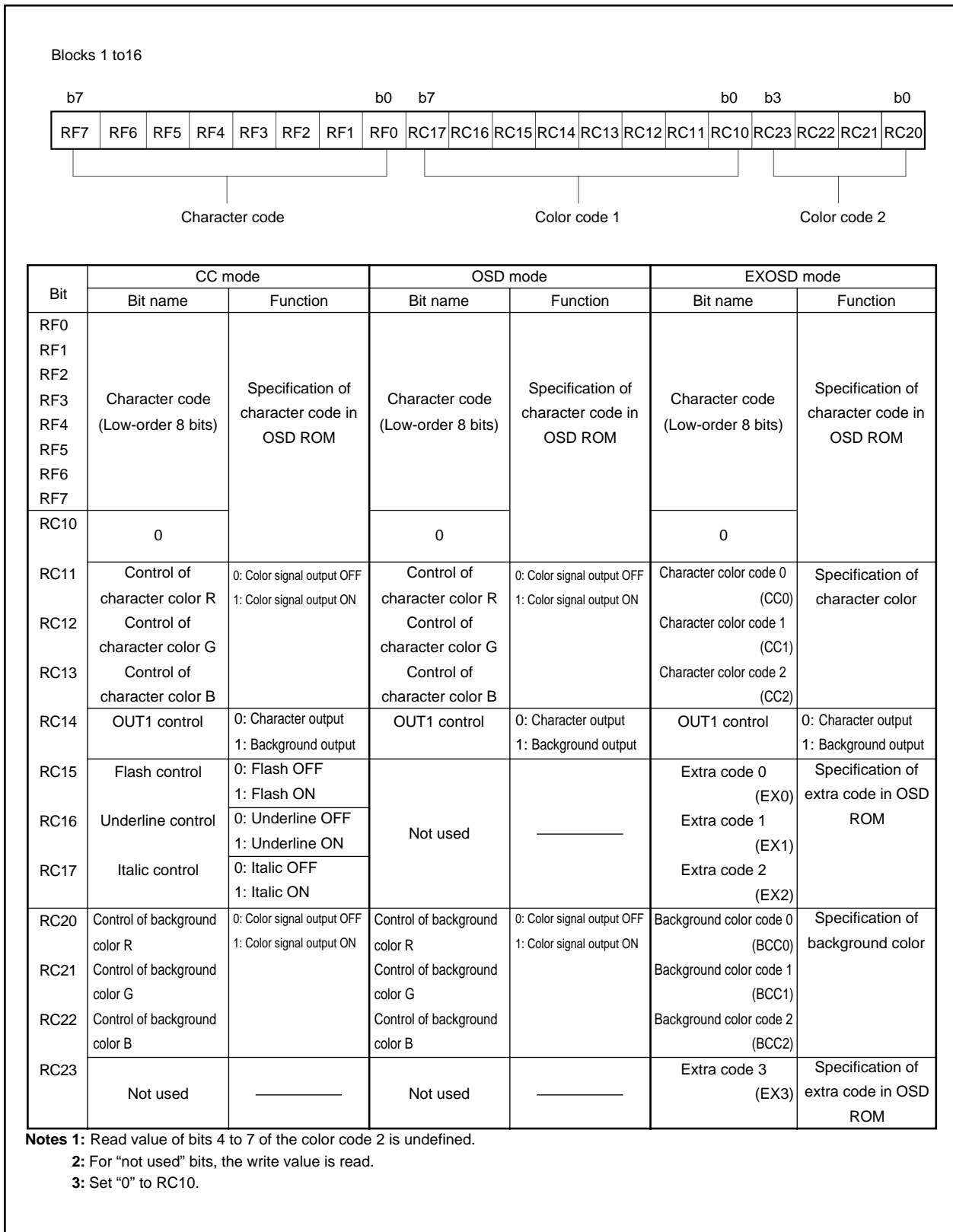


Fig. 72. Structure of OSD RAM

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**(7) Character color**

The color for each character is displayed by the color code 1. The kinds and specification method of character color are different depending on each mode.

- CC mode ..... 7 kinds  
 Specified by bits 1 (R), 2 (G), and 3 (B) of color code 1
- OSD mode ..... 7 kinds  
 Specified by bits 1 (R), 2 (G) and 3 (B) of color code 1
- EXOSD mode ..... 5 kinds  
 Specified by bits 1 (CC0), 2 (CC1), and 3 (CC2) of color code 1

The correspondence Table of color code 1 and color signal output in the EXOSD mode is shown in Table 17.

**Table 17. Correspondence Table of Color Code 1 and Color Signal Output in EXOSD Mode**

Color Code 1			Color Signal Output		
Bit 3 CC2	Bit 2 CC1	Bit 1 CC0	R	G	B
0	0	0	0	0	0
0	0	1	1	0	0
0	1	0	0	1	0
0	1	1	1	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	0	1	1
1	1	1	1	1	1

**(8) Character background color**

The character background color can be displayed in the character display area. The character background color for each character is specified by color code 2. The kinds and specification method of character background color are different depending on each mode.

- CC mode ..... 7 kinds  
 Specified by bits 0 (R), 1 (G), and 2 (B) of color code 2
- OSD mode ..... 7 kinds  
 Specified by bits 0 (R), 1 (G), and 2 (B) of color code 2
- EXOSD mode ..... 5 kinds  
 Specified by bits 0 (BCC0), 1 (BCC1), and 2 (BCC2) of color code 2

The correspondence table of the color code 2 and color signal output in the EXOSD mode is shown in Table 18.

**Table 18. Correspondence Table of Color Code 2 and Color Signal Output in EXOSD Mode**

Color Code 2			Color Signal Output		
Bit 2 BCC2	Bit 1 BCC1	Bit 0 BCC0	R	G	B
0	0	0	0	0	0
0	0	1	1	0	0
0	1	0	0	1	0
0	1	1	1	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	0	1	1
1	1	1	1	1	1

**Note :** The character background color is displayed in the following part :  
 (character display area)–(character font)–(border)–(extra font).  
 Accordingly, the character background color does not mix with these color signal.

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**(9) OUT1, OUT2 signals**

The OUT1, OUT2 signals are used to control the luminance of the video signal. The output waveform of the OUT1, OUT2 signals is controlled by bit 4 of color code 1 (refer to Figure 72), bits 2 and 7 of

the block control register (refer to Figure 56). The setting values for controlling OUT1, OUT2 and the corresponding output waveform is shown in Figure 73.

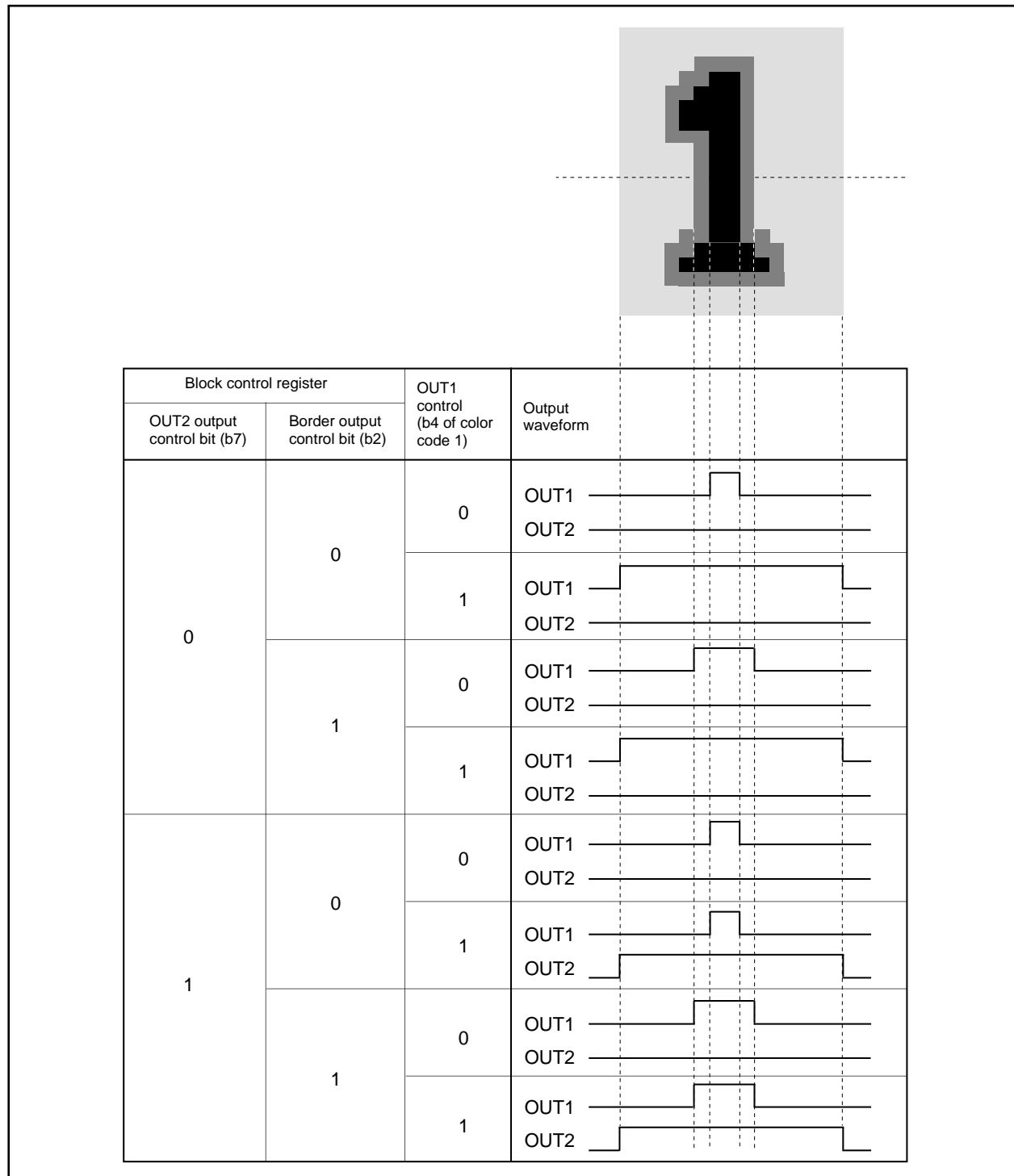


Fig. 73. Setting Value for Controlling OUT1, OUT2 and Corresponding Output Waveform

**PRELIMINARY**  
Notice: This is not a final specification.  
Some parametric limits are subject to change.

**(10) Attribute**

The attributes (flash, underline, italic) are controlled to the character font. The attributes for each character are specified by the color codes 1 and 2 (refer to Figure 71). The attributes to be controlled are different depending on each mode.

- CC mode ..... Flash, underline, italic
- OSD mode ..... Border (all bordered, shadow bordered can be selected)
- EXOSD mode ..... Border (all bordered, shadow bordered can be selected) , extra font (16 kinds)

① **Under line**

The underline is output at the 23th and 24th dots in vertical direction only in the CC mode. The underline is controlled by bit 6 of color code 1. The color of underline is the same color as that of the character font.

② **Flash**

The parts of the character font, the underline, and the character background are flashed only in the CC mode. The color signals (R, G, B, OUT1) of the character font and the underline are controlled by bit 5 of color code 1. All of the color signals for the character font flash. However, the color signal for the character background can be controlled by bit 3 of the OSD control register (refer to Figure 52). The flash cycle bases on the VSYNC count.

- VSYNC cycle X 48 ≒ 800 ms (at flash ON)
- VSYNC cycle X 16 ≒ 267 ms (at flash OFF)

③ **Italic**

The italic is made by slanting the font stored in OSD ROM to the right only in the CC mode. The italic is controlled by bit 7 of color code 1.

The display example of the italic and underline is shown in Figure 75. In this case, "R" is displayed.

**Notes 1:** When setting both the italic and the flash, the italic character flashes.

- 2:** When the pre-divide ratio = 1, the italic character with slant of 1 dot X 5 steps is displayed (refer to Figure 74 (c)). When the pre-divide ratio = 2, the italic character with slant of 1/2 dot X 10 steps is displayed (refer to Figure 74 (d)).
- 3:** The boundary of character color is displayed in italic. However, the boundary of character background color is not affected by the italic (refer to Figure 75).
- 4:** The adjacent character (one side or both side) to an italic character is displayed in italic even when the character is not specified to display in italic (refer to Figure 75).
- 5:** When displaying the italic character in the block with the pre-divide ratio = 1, set the OSD clock frequency to 11 MHz to 14 MHz.

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

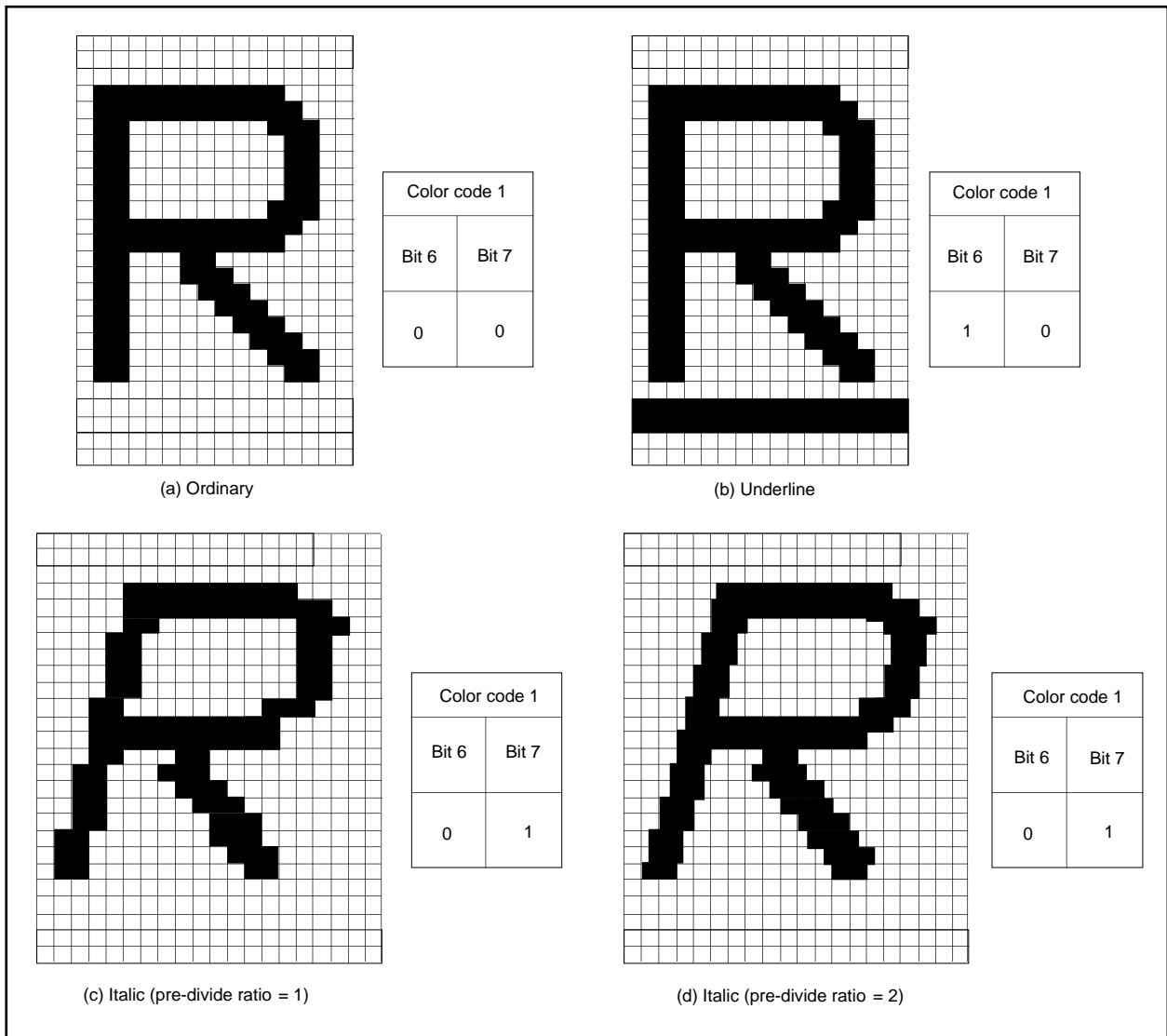


Fig. 74. Example of Attribute Display (in CC mode)

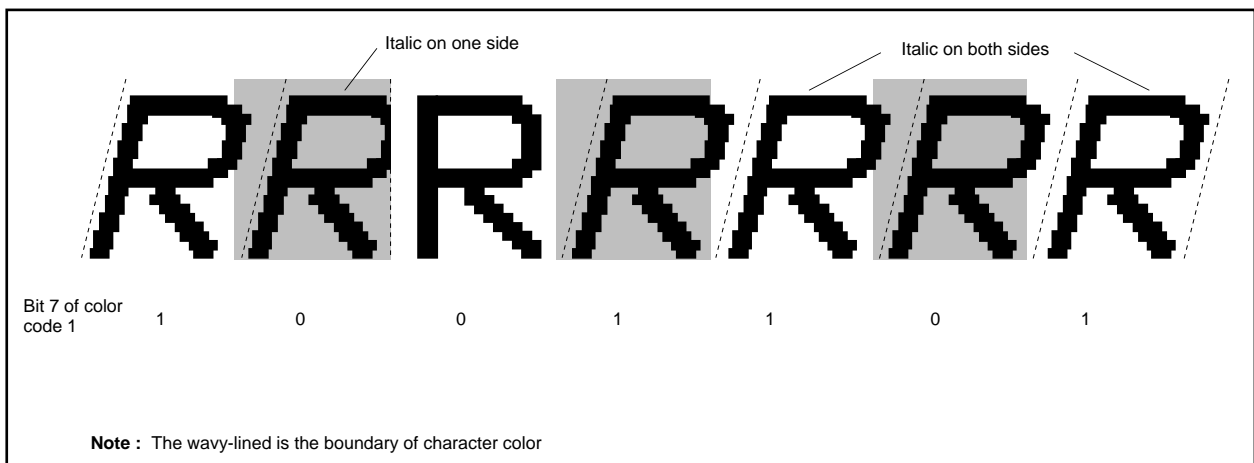


Fig. 75. Example of Italic Display



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

④ **Extra font**

There are 16 kinds of the extra fonts configured with 16 X 26 dots in OSD ROM. This 16 kinds fonts can be displayed by ORed with the character font by a character unit (refer to Figure 53). In only the EXOSD mode, the extra font is controlled the following : bits 7 to 5 of the color code 1 and bit 3 of the color code 2.

The extra font color for each screen is specified by the extra color register. When the character font overlaps with the extra font, the color of the area becomes the ORed color of both fonts.

- Notes 1 :** When using the extra font, set bits 7 and 6 of the OSD control register to "0" (refer to Figure 55).  
**2 :** Extra fonts are always displayed by ORed with the character font. Accordingly, when displaying only a extra font, set a blank for a character font and OR with it.

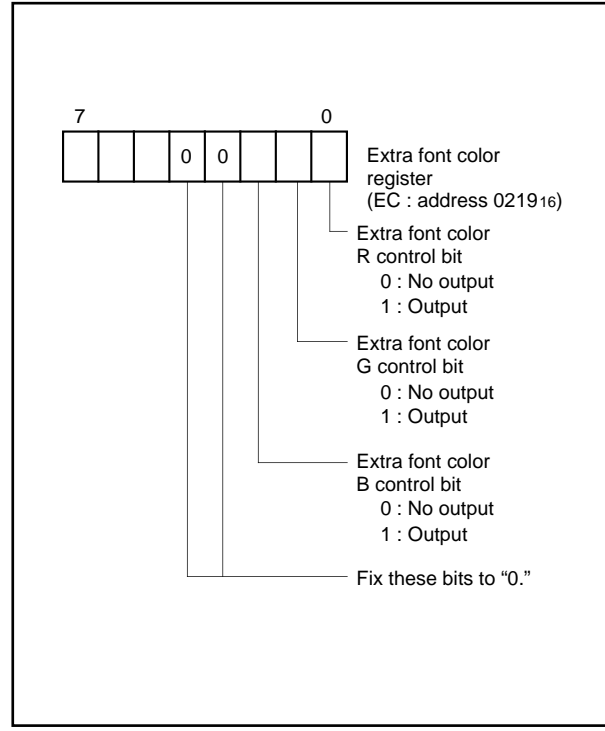


Fig. 76. Extra Font Color Register

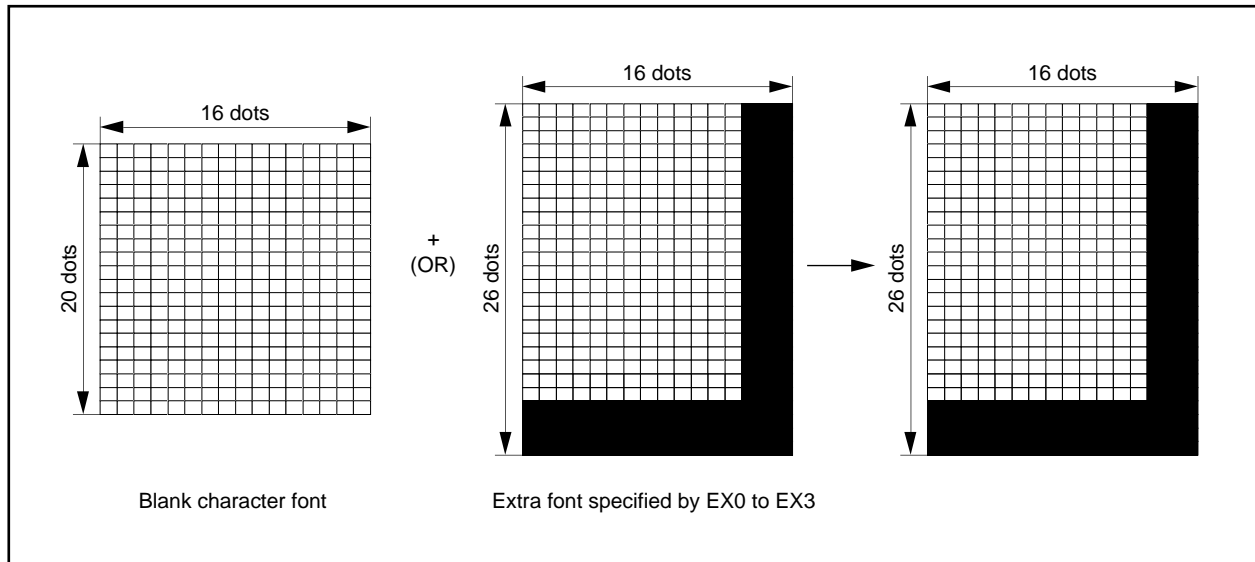


Fig. 77. Display Example of Only Extra Font

⑤ **Border**

The border is output in the OSD mode and the EXOSD mode. The all bordered (bordering around of character font) and the shadow bordered (bordering right and bottom sides of character font) are selected (refer to Figure 77) by bit 2 of the OSD control register (refer to Figure 55). The border ON/OFF is controlled by bit 2 of the block control register (refer to Figure 56).

The OUT1 signal is used for border output. The border color for each screen is specified by the border color register.

The horizontal size (x) of border is 1Tc (OSD clock cycle divided in the pre-divide circuit) regardless of the character font dot size. However, only when the pre-divide ratio = 2 and character size = 1.5Tc, the horizontal size is 1.5Tc. The vertical size (y) different depending on the screen scan mode and the vertical dot size of character font.

**Notes 1 :** There is no border for the extra font.

**2 :** The border dot area is the shaded area as shown in Figure 79. In the EXOSD mode, top and bottom of character font display area is not bordered.

**3 :** When the border dot overlaps on the next character font, the character font has priority (refer to Figure 80 A). When the border dot overlaps on the next character background, the border has priority (refer to Figure 80 B).

**4 :** The border is not displayed at right side of the most right dot in the display area of the 36th character (the character located at the most right of the block). However, note that EPROM version can display the border above when programming.

(The border for the right edge dots of the 40th's character area is not displayed.)

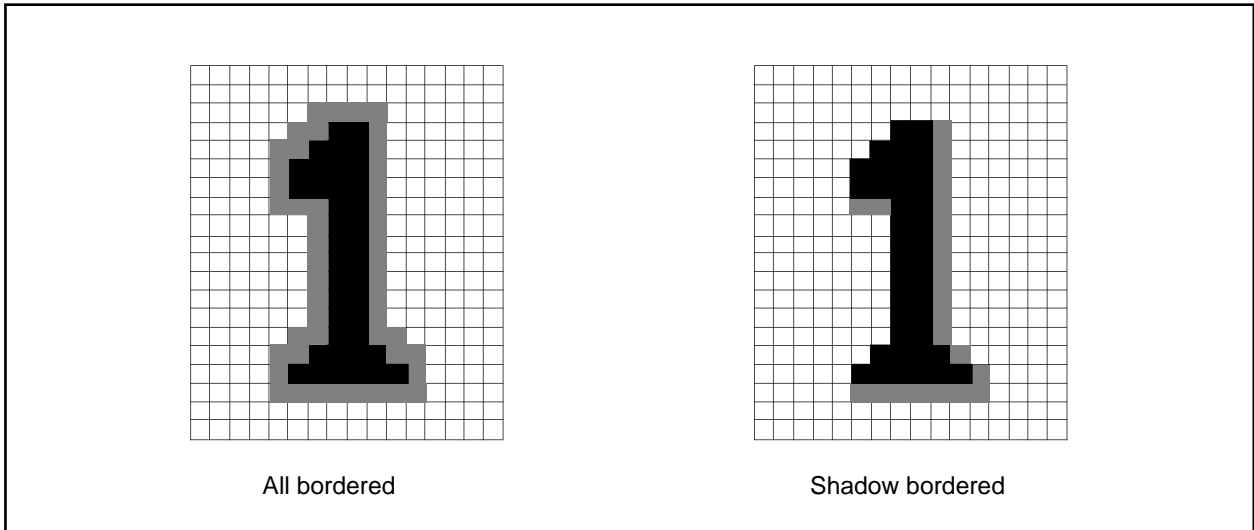


Fig. 78. Example of Border Display

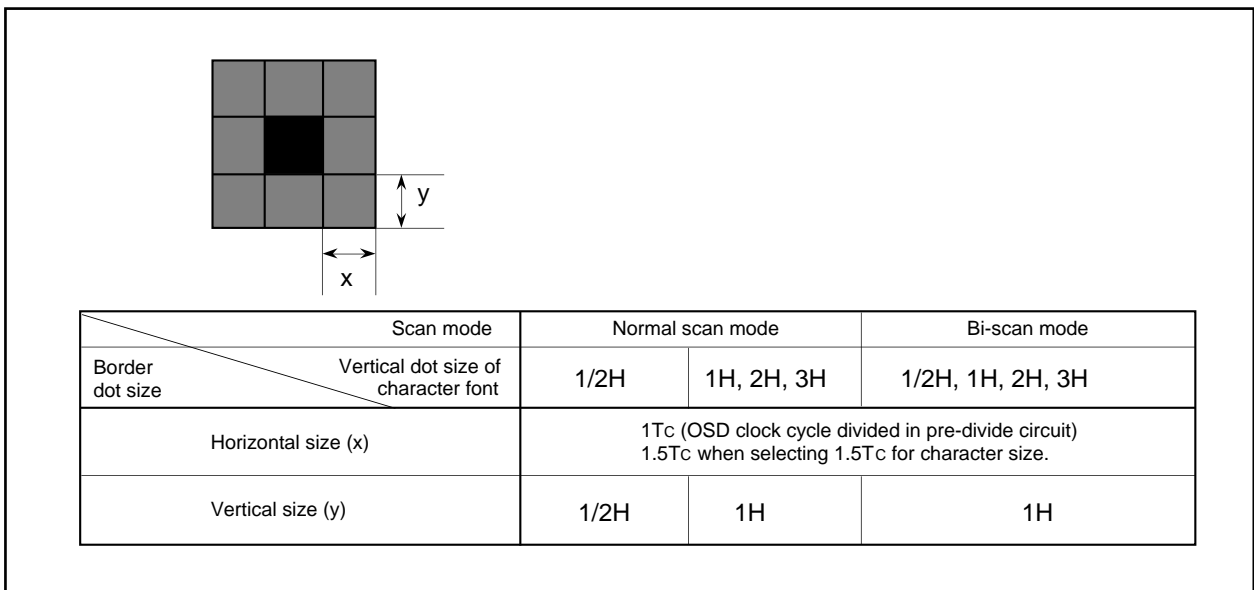


Fig. 79. Horizontal and Vertical Size of Border

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

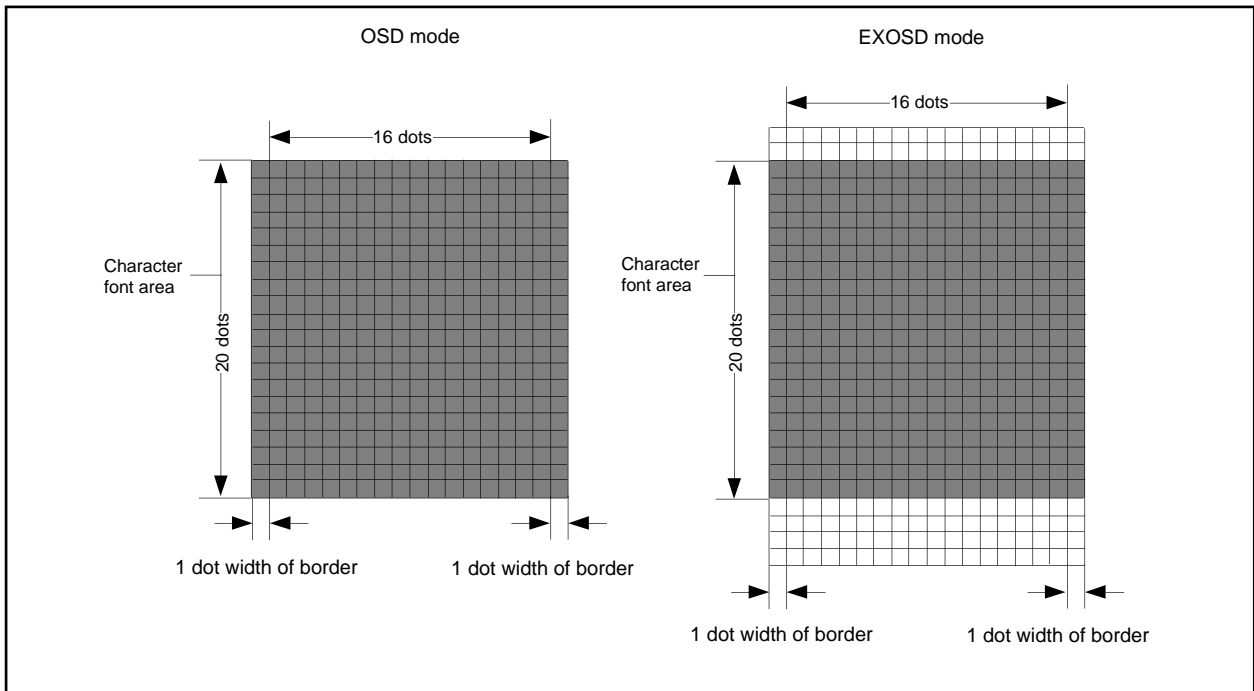


Fig. 80. Border Area

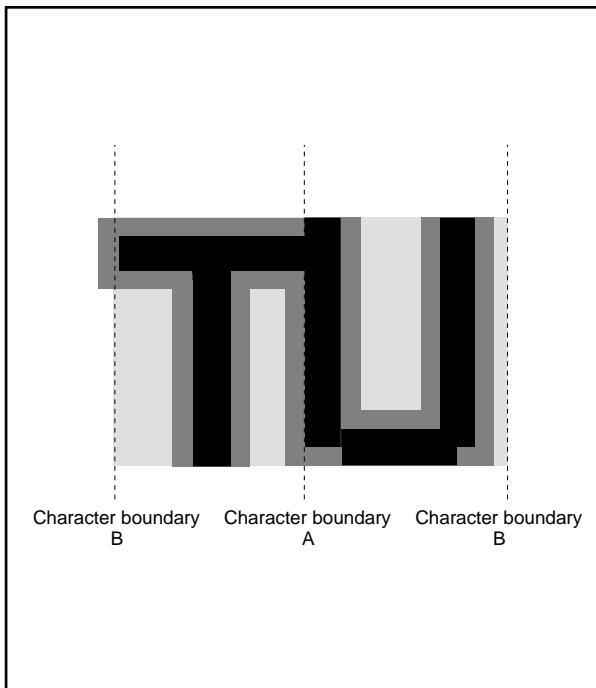


Fig. 81. Border Priority

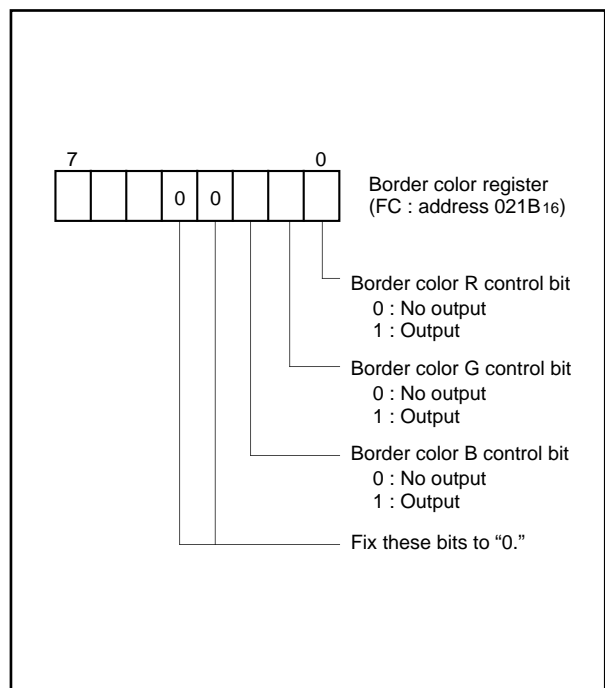


Fig. 82. Border Color Register

**(11) Multiline Display**

The M37274MA-XXXSP can ordinarily display 16 lines on the CRT screen by displaying 16 blocks at different vertical positions. In addition, it can display up to 12 lines by using OSD interrupts.

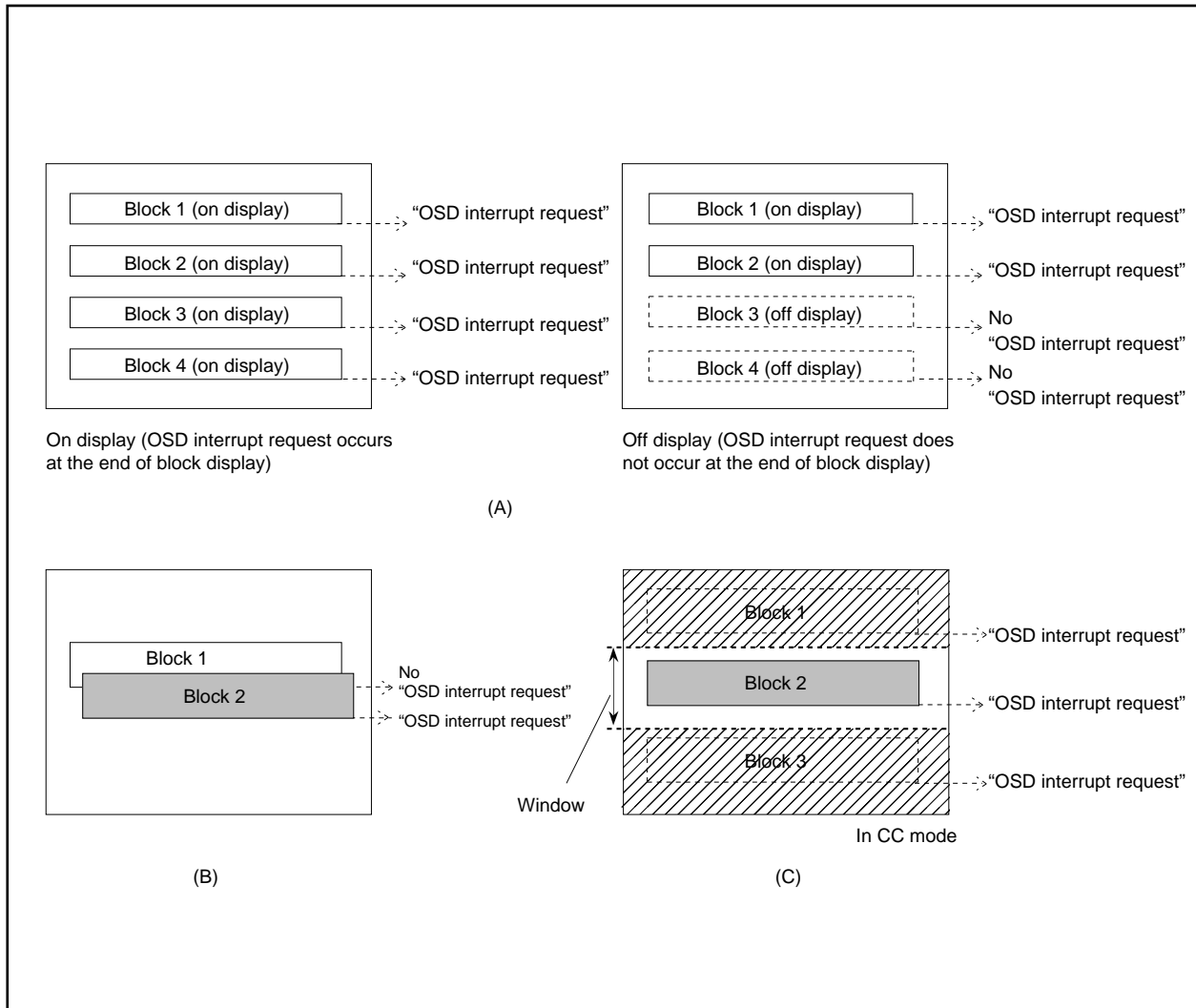
An OSD interrupt request occurs at the point at which display of each block has been completed. In other words, when a scanning line reaches the point of the display position (specified by the vertical position registers) of a certain block, the character display of that block starts, and an interrupt occurs at the point at which the scanning line exceeds the block. The mode in which an OSD interrupt occurs is different depending on the setting of the raster color register (refer to Figure 89).

- When bit 7 of the raster color register is "0"  
 An OSD interrupt occurs at the end of block display in the OSD and the EXOSD mode.
- When bit 7 of the raster color register is "1"  
 An OSD interrupt occurs at the end of block display in the CC mode.

**Notes 1:** An OSD interrupt 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 by the display control bit of the block control register (addresses 00D0<sub>16</sub> to 00DB<sub>16</sub>), an OSD interrupt request does not occur (refer to Figure 83 (A)).

**2:** When another block display appears while one block is displayed, an OSD interrupt request occurs only once at the end of the another block display (refer to Figure 83 (B)).

**3:** On the screen setting window, an OSD interrupt occurs even at the end of the CC mode block (off display) out of window (refer to Figure 83 (C)).



**Fig. 83. Note on Occurrence of OSD Interrupt**

**(12) Automatic Solid Space Function**

This function generates automatically the solid space (OUT1 or OUT2 blank output) of the character area in the CC mode.

The solid space is output in the following area :

- the character area except character code "0916"
- the character area on the left and right sides

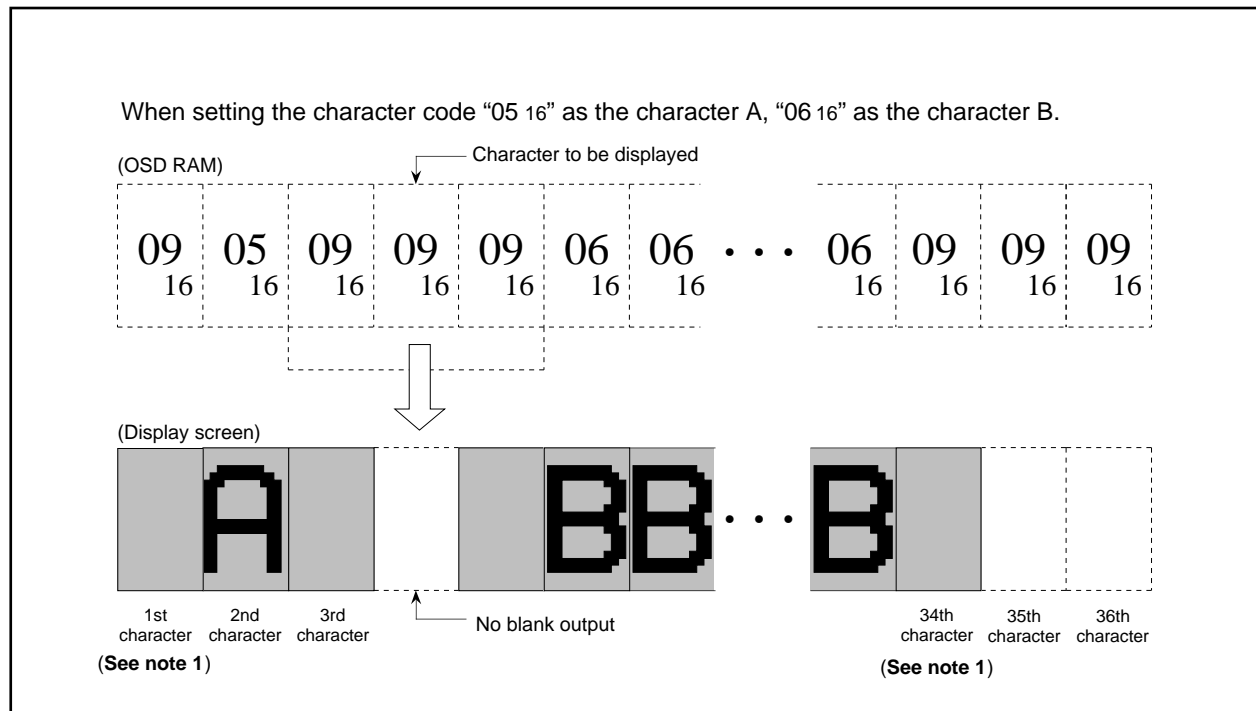
This function is turned on and off by bit 4 of the OSD control register (refer to Figure 55).

**Note :** When using this function, set "0916" to the character below :

- The 1st character
- The 34th character and the following character.

**Table 19. Setting for Automatic Solid Space**

Bit 4 of OSD control register	0				1			
Bit 7 of block control register	0		1		0		1	
Bit 4 of color code 1	0	1	0	1	0	1	0	1
OUT1 output signal	Character font part	Character display area	Character font part		Solid space		Character font part	
OUT2 output signal	OFF		OFF	Character display area	OFF		Solid space	



**Fig. 84. Display Screen Example of Automatic Solid Space**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**(13) Scan Mode**

M37271MF-XXXSP has the bi-scan mode for corresponding to Hsync of double speed frequency. In the bi-scan mode, the vertical start display position and the vertical size is two times as compared with the normal scan mode. The scan mode is selected by bit 1 of the OSD control register (refer to Figure 55).

**Table 20. Setting for Scan Mode**

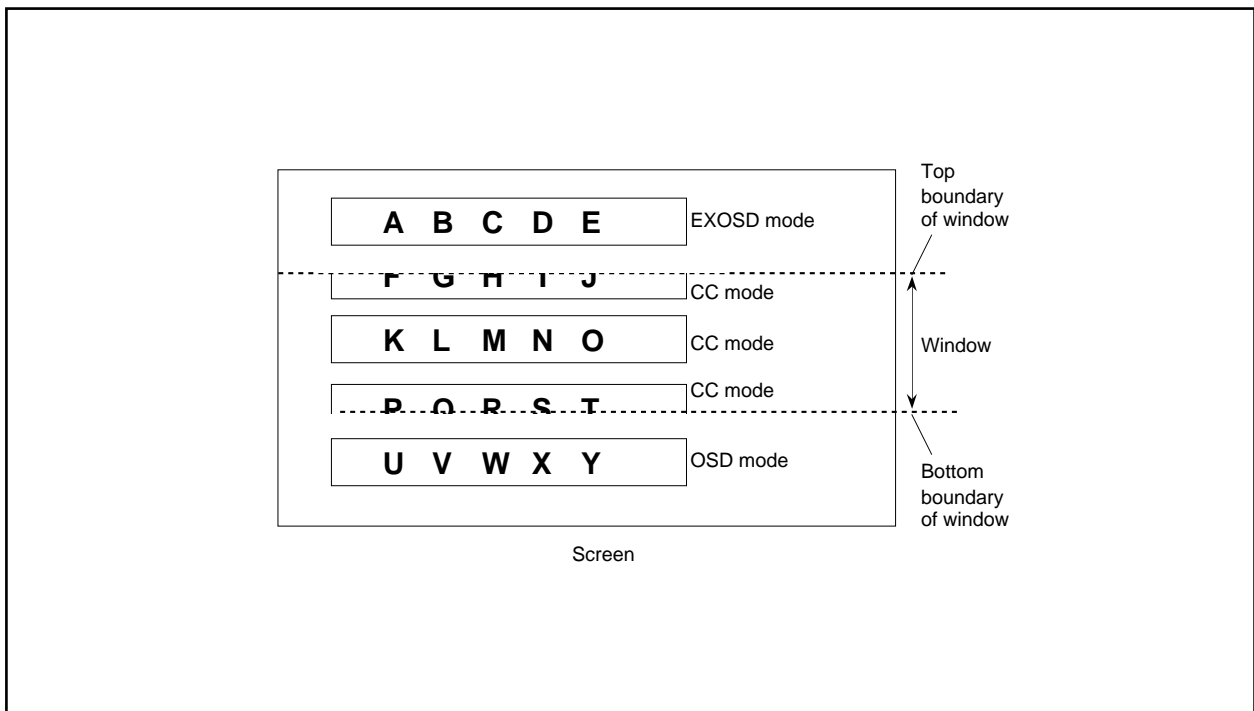
Parameter	Scan Mode	Normal Scan	Bi-Scan
Bit 1 of OSD control register		0	1
Vertical display start position		Value of vertical position register X 1H	Value of vertical position register X 2H
Vertical dot size		1Tc X 1/2H	1Tc X 1H
		1Tc X 1H	1Tc X 2H
		2Tc X 2H	2Tc X 4H
		3Tc X 3H	3Tc X 6H

**(14) Window Function**

This function sets the top and bottom boundary of display limit on a screen. The window function is valid only in the CC mode. The top boundary is set by window H registers 1 and 2. The bottom boundary is set by window L registers 1 and 2. This function is turned on and off by bit 5 of the OSD control register (refer to Figure 55). The structure of window H registers 1 and 2 is shown in Figure 86, the structure of window L registers 1 and 2 is shown in Figure 87.

**Notes 1:** Set values except "0016" and "0116" to the window H register 1 when the window H register 2 is "0016."

**2:** Set the register value fit for the following condition :  
 $(WH1 + WH2) < (WL1 + WL2)$



**Fig. 85. Example of Window Function**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

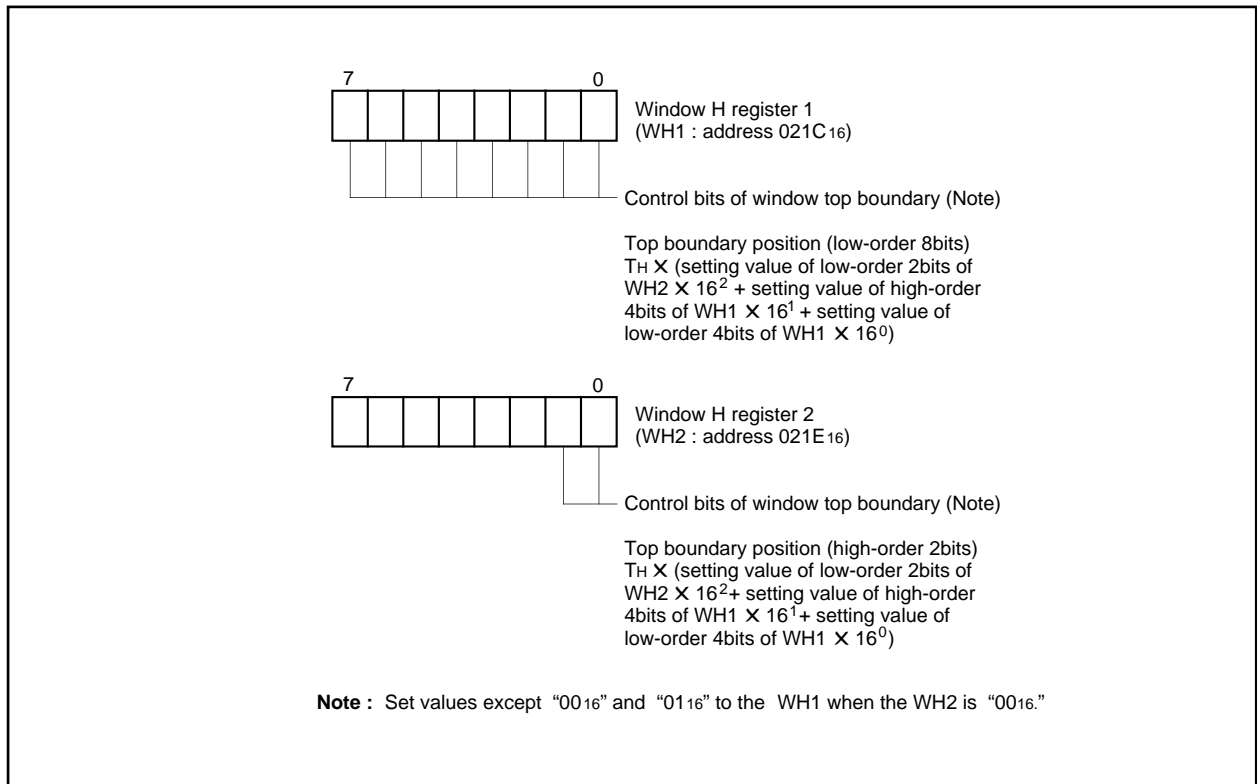


Fig. 86. Window H Registers

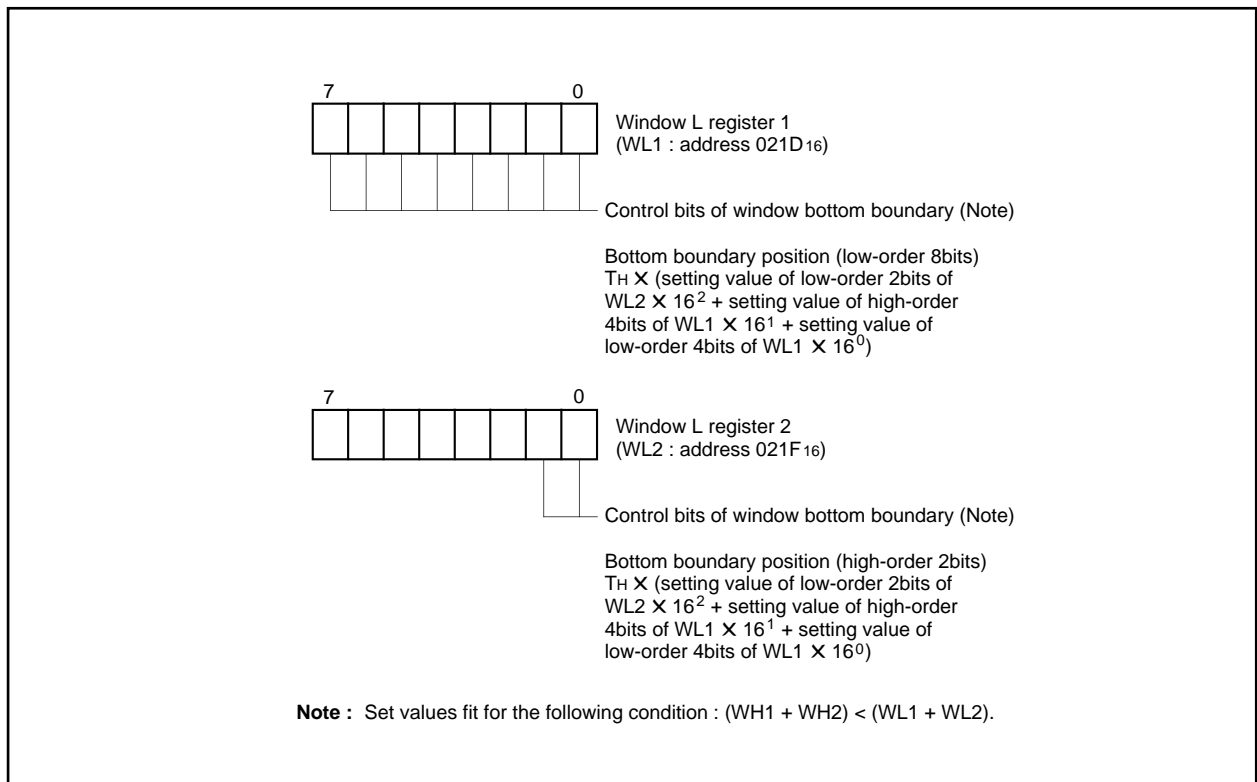


Fig. 87. Window L Registers

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**(15) OSD Output Pin Control**

The OSD output pins R, G, B, and OUT1 can also function as ports P52, P53, P54 and P55. Set the corresponding bit of the OSD port control register (address 00CB16) to "0" to specify these pins as OSD output pins, or set it to "1" to specify it as a general-purpose port P5 pins. The OUT2 can also function as port P10. Set the corresponding bit of the port P1 direction register (address 00C316) to "1" (output mode). After that, switch between the OSD output function and the port function by the OSD port control register. Set the corresponding bit to "1" to specify the pin as OSD output pin, or set it to "0" to specify as port P1 pin.

The input polarity of the HSYNC, VSYNC and output polarity of signals R, G, B, OUT1 and OUT2 can be specified with the I/O polarity control register (address 021716) . Set a bit to "0" to specify positive polarity; set it to "1" to specify negative polarity (refer to Figure 68). The OSD port control register is shown in Figure 88.

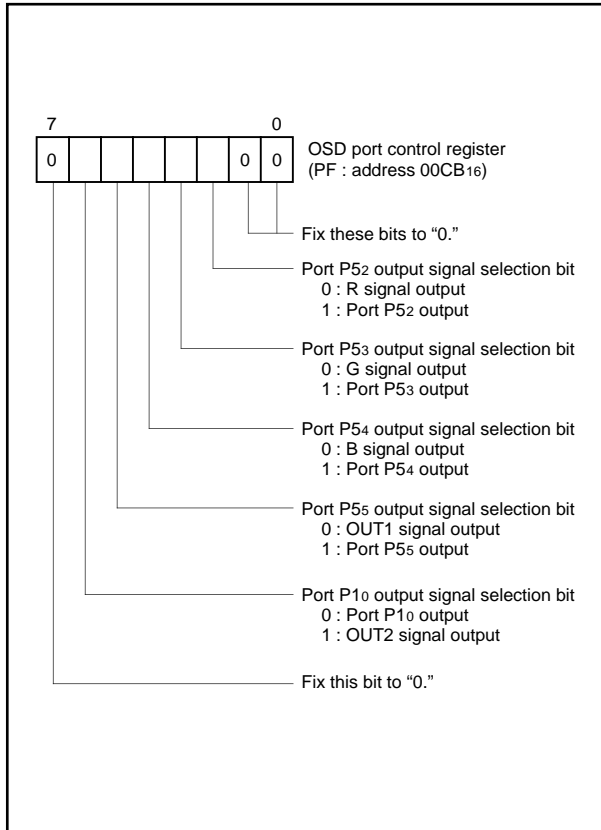


Fig. 88. OSD Port Control Register



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**(16) Raster Coloring Function**

An entire screen (raster) can be colored by setting the bits 6 to 0 of the raster color register. Since each of the R, G, B, OUT1, and OUT2 pins can be switched to raster coloring output, 7 raster colors can be obtained.

If the OUT1 pin has been set to raster coloring output, a raster coloring signal is always output during 1 horizontal scanning period. This setting is necessary for erasing a background TV image.

If the R, G, and B pins have been set to output, a raster coloring signal is output in the part except a no-raster colored character (in Figure 90, a character "1") and the character background output during 1 horizontal scanning period. This ensures that the character color/ the character background color is not mixed with the raster color.

The structure of the raster color register is shown in Figure 89, the example of raster coloring is shown in Figure 90.

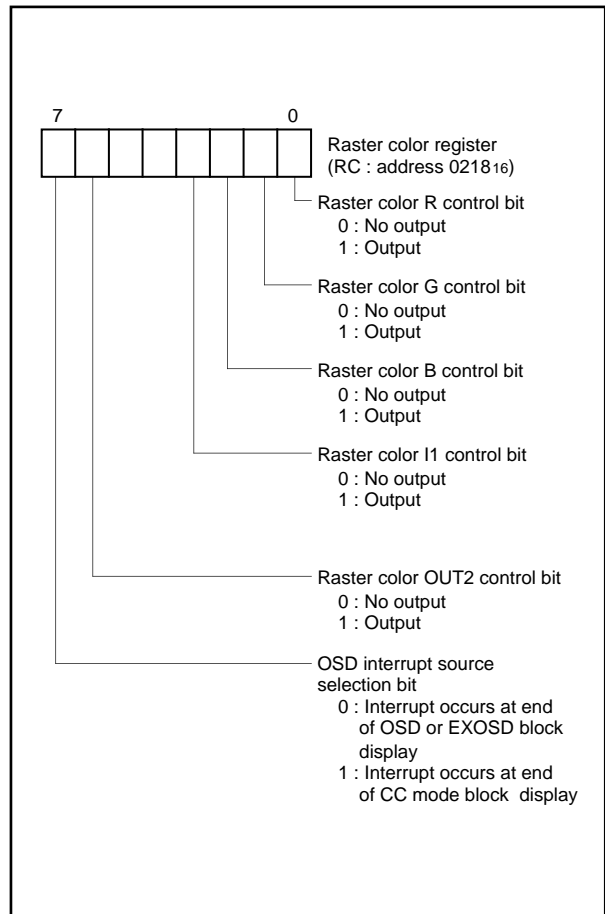


Fig. 89. Raster Color Register

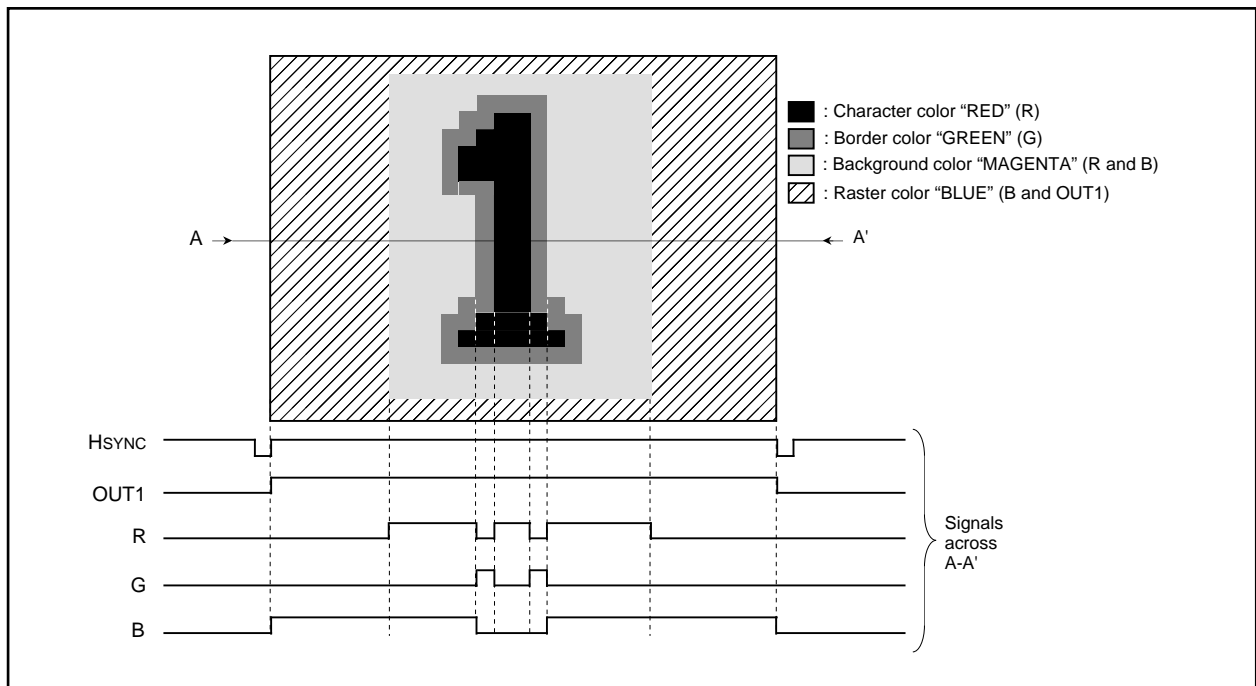


Fig. 90. Example of Raster Coloring

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

### ROM CORRECTION FUNCTION

This can correct 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 X 2 blocks.

Block 1 : addresses 02C0<sub>16</sub> to 02DF<sub>16</sub>

Block 2 : addresses 02E0<sub>16</sub> to 02FF<sub>16</sub>

Set the 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 memory for correction. 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 main program to the correction program.

**3 :** Do not set the same ROM correction address to blocks 1 and 2.

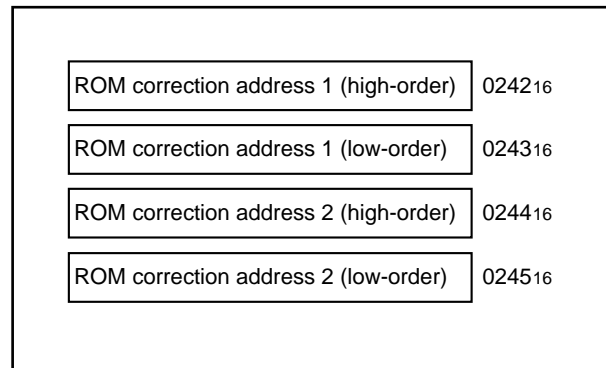


Fig. 91. ROM Correction Address Registers

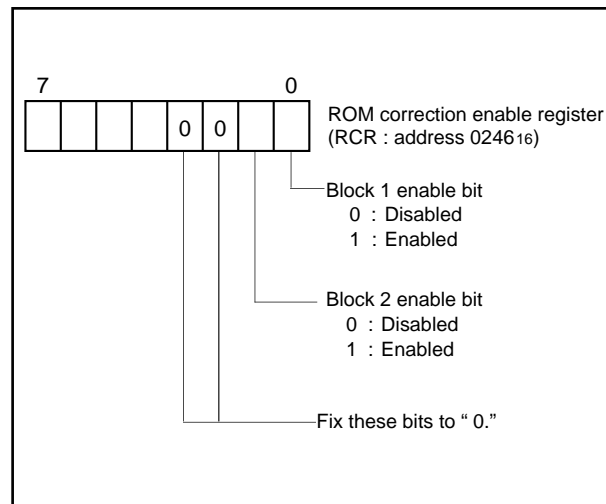


Fig. 92. ROM Correction Enable Register

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**RESET CIRCUIT**

When the oscillation of a quartz-crystal oscillator or a ceramic resonator is stable and the power source voltage is  $5\text{ V} \pm 10\%$ , hold the RESET pin at LOW for  $2\ \mu\text{s}$  or more, then return it to HIGH. Then, as shown in Figure 94, reset is released and the program starts from the address formed by using the content of address FFFF<sub>16</sub> as the high-order address and the content of address FFFE<sub>16</sub> as the low-order address. The internal state of microcomputer at reset are shown in Figures 5 to 9.

An example of the reset circuit is shown in Figure 93.

The reset input voltage must be kept 0.9 V or less until the power source voltage surpasses 4.5 V.

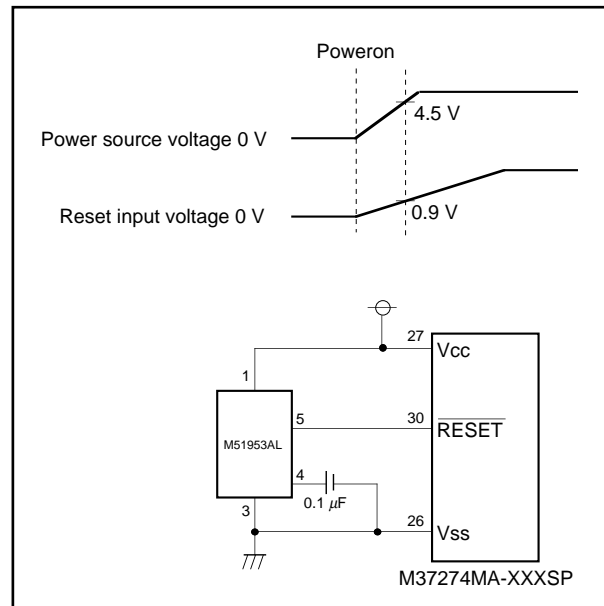


Fig. 93. Example of Reset Circuit

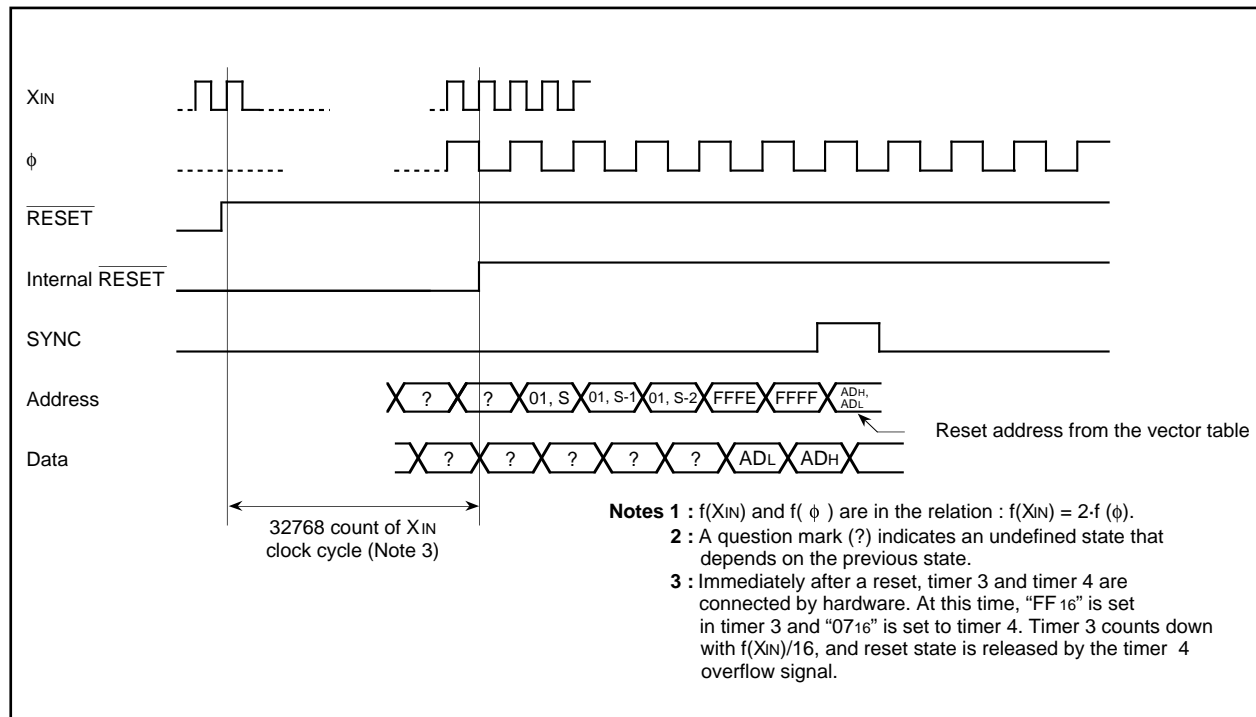


Fig. 94. Reset Sequence

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**CLOCK GENERATING CIRCUIT**

The M37274MA-XXXSP has 2 built-in oscillation circuits. An oscillation circuit can be formed by connecting a resonator between XIN and XOUT (XCIN and XCOUT). Use the circuit constants in accordance with the resonator manufacturer's recommended values. No external resistor is needed between XIN and XOUT since a feed-back resistor exists on-chip. However, an external feed-back resistor is needed between XCIN and XCOUT. When using XCIN-XCOUT as sub-clock, clear bits 5 and 4 of the clock source control register to "0." To supply a clock signal externally, input it to the XIN (XCIN) pin and make the XOUT (XCOUT) pin open. When not using XCIN clock, connect the XCIN to VSS and make the XCOUT pin open.

After reset has completed, the internal clock  $\phi$  is half the frequency of XIN. Immediately after poweron, both the XIN and XCIN clock start oscillating. To set the internal clock  $\phi$  to low-speed operation mode, set bit 7 of the CPU mode register (address 00FB16) to "1."

**Oscillation Control**

**(1) Stop mode**

The built-in clock generating circuit is shown in Figure 95. When the STP instruction is executed, the internal clock  $\phi$  stops at HIGH. At the same time, timers 3 and 4 are connected by hardware and "FF16" is set in timer 3 and "0716" is set in timer 4. Select  $f(XIN)/16$  or  $f(XCIN)/16$  as the timer 3 count source (set both bit 0 of the timer mode register 2 and bit 6 at address 00C716 to "0" before the execution of the STP instruction). Moreover, set the timer 3 and timer 4 interrupt enable bits to disabled ("0") before execution of the STP instruction. The oscillator restarts when external interrupt is accepted. However, the internal clock  $\phi$  keeps its "H" level until timer 4 overflows, allowing time for oscillation stabilization when a ceramic resonator or a quartz-crystal oscillator is used.

**(2) Wait mode**

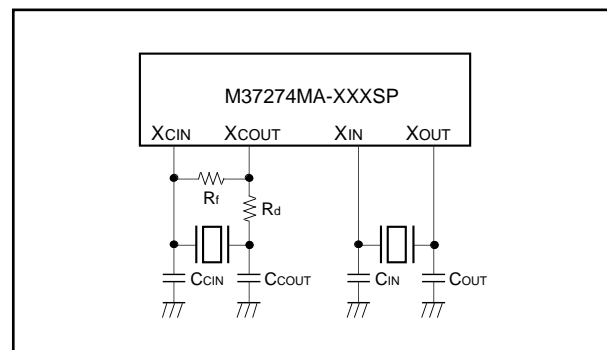
When the WIT instruction is executed, the internal clock  $\phi$  stops in the "H" level but the oscillator continues running. This wait state is released at reset or when an interrupt is accepted (Note). Since the oscillator does not stop, the next instruction can be executed at once.

**Note:** In the wait mode, the following interrupts are invalid.

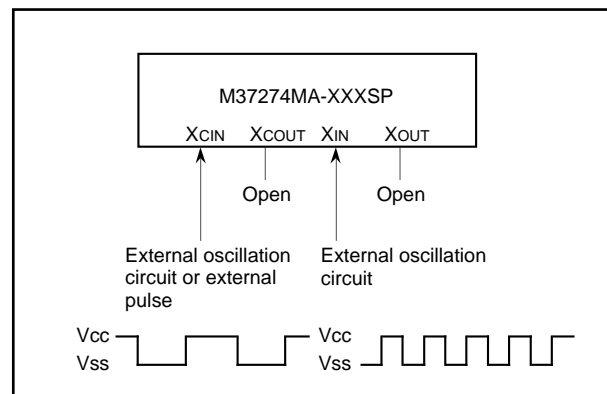
- (1) VSYNC interrupt
- (2) OSD interrupt
- (3) Timers 1 and 2 interrupts using TIM2 pin input as count source
- (4) Timer 3 interrupt using TIM3 pin input as count source
- (5) Data slicer interrupt
- (6) Multi-master I<sup>2</sup>C-BUS interface interrupt
- (7)  $f(XIN)/4096$  interrupt
- (8) All timer interrupts using  $f(XIN)/2$  or  $f(XCIN)/2$  as count source
- (9) All timer interrupts using  $f(XIN)/4096$  or  $f(XCIN)/4096$  as count source
- (10) A-D conversion interrupt

**(3) Low-Speed Mode**

If the internal clock is generated from the sub-clock (XCIN), a low power consumption operation can be realized by stopping only the main clock XIN. To stop the main clock, set bit 6 (CM6) of the CPU mode register (00FB16) to "1." When the main clock XIN is restarted, the program must allow enough time to for oscillation to stabilize. Note that in low-power-consumption mode the XCIN-XCOUT drivability can be reduced, allowing even lower power consumption. To reduce the XCIN-XCOUT drivability, clear bit 5 (CM5) of the CPU mode register (00FB16) to "0." At reset, this bit is set to "1" and strong drivability is selected to help the oscillation to start. When an STP instruction is executed, set this bit to "1" by software before executing.



**Fig. 95. Ceramic Resonator Circuit Example**

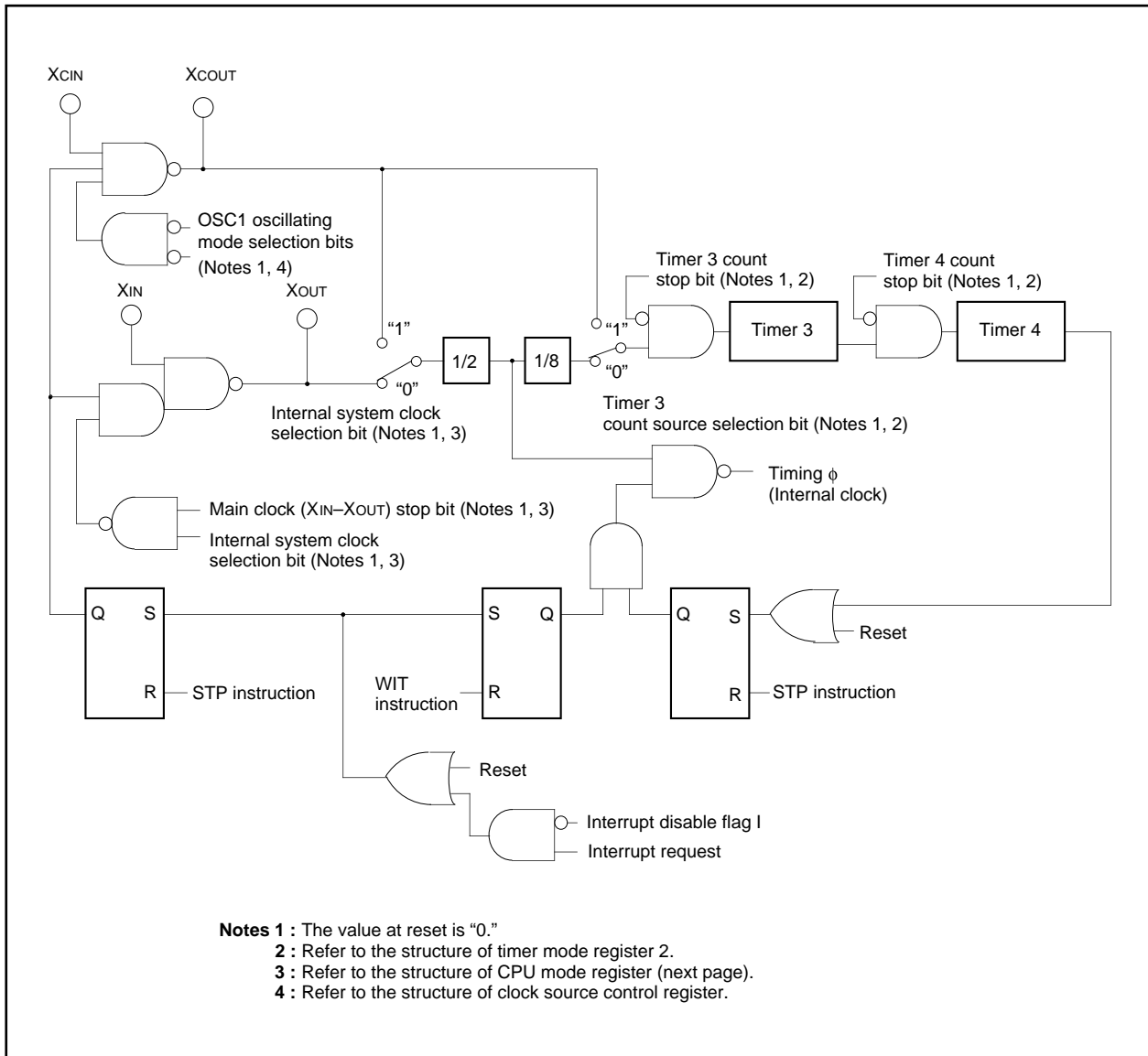


**Fig. 96. External Clock Input Circuit Example**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

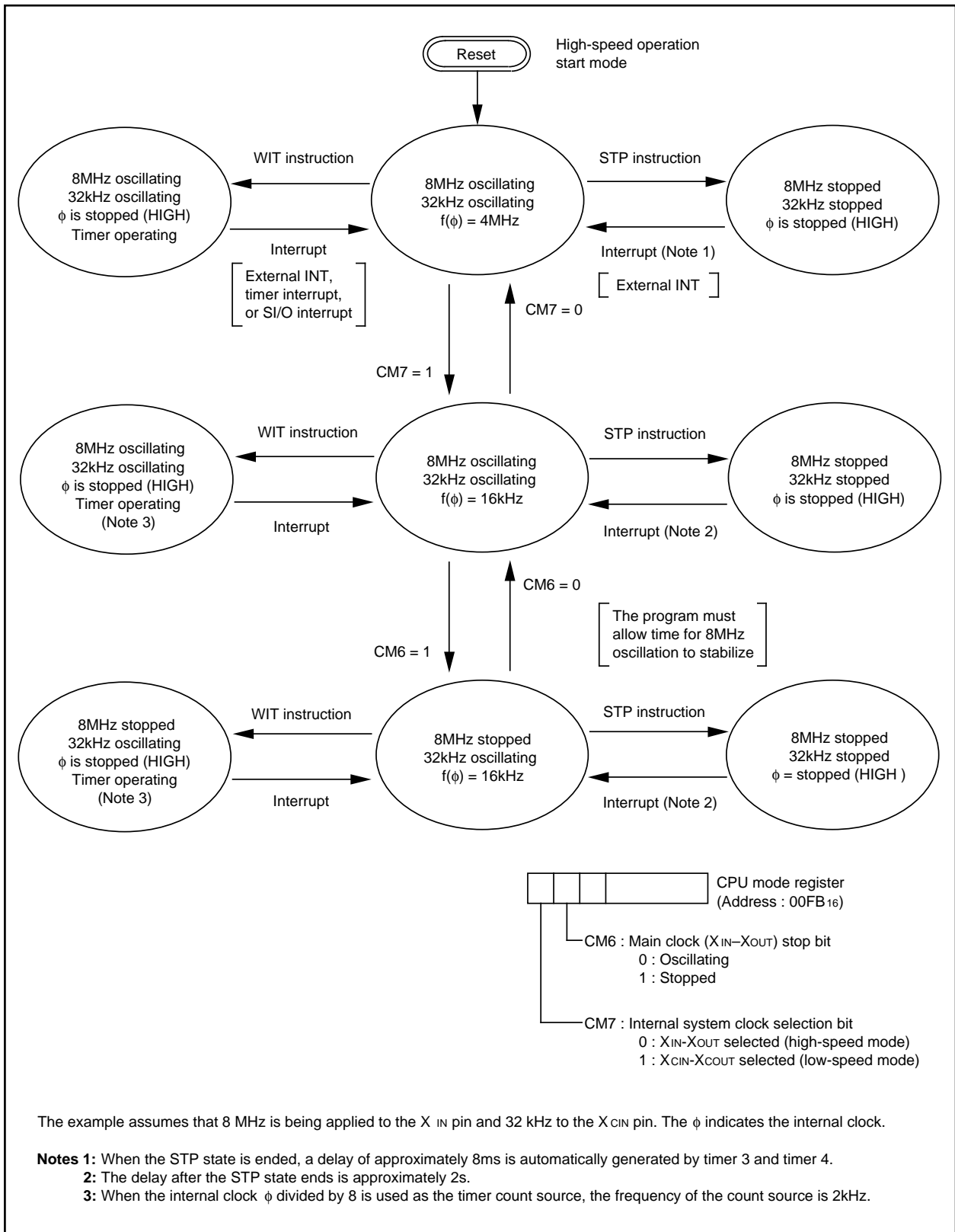
SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER



**Fig. 97. Clock Generating Circuit Block Diagram**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

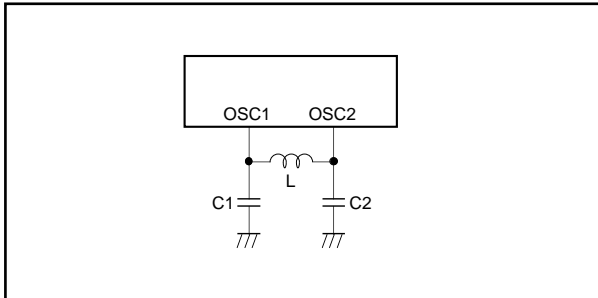


**Fig. 98. State Transitions of System Clock**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**DISPLAY OSCILLATION CIRCUIT**

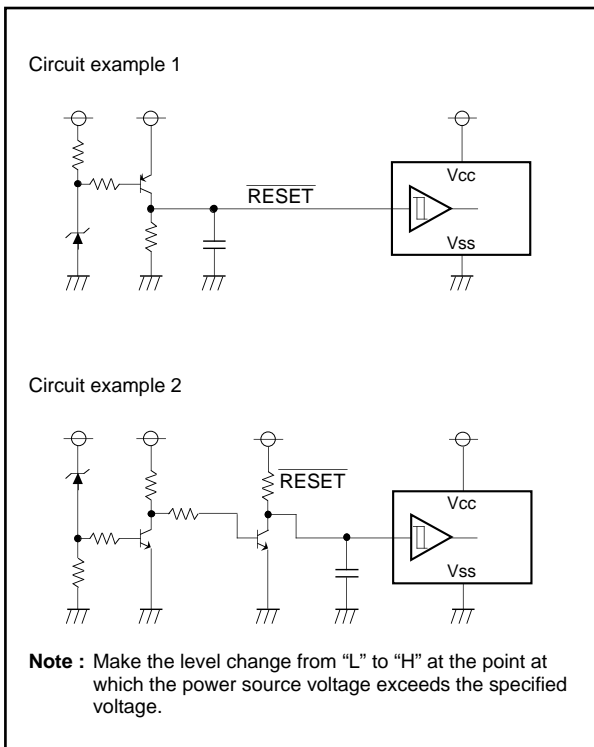
The OSD oscillation circuit has a built-in clock oscillation circuits, so that a clock for OSD can be obtained simply by connecting an LC, a ceramic resonator, or a quartz-crystal oscillator across the pins OSC1 and OSC2. Which of the sub-clock or the OSD oscillation circuit is selected by setting bits 5 and 4 of the clock source control register (address 021616).



**Fig. 99. Display Oscillation Circuit**

**AUTO-CLEAR CIRCUIT**

When a power source is supplied, the auto-clear function will operate by connecting the following circuit to the RESET pin.



**Fig. 100. Auto-clear Circuit Example**

**ADDRESSING MODE**

The memory access is reinforced with 17 kinds of addressing modes. Refer to SERIES 740 <Software> User's Manual for details.

**MACHINE INSTRUCTIONS**

There are 71 machine instructions. Refer to SERIES 740 <Software> User's Manual for details.

**PROGRAMMING NOTES**

- (1) The divide ratio of the timer is 1/(n+1).
- (2) Even though the BBC and BBS instructions are executed immediately after the interrupt request bits are modified (by the program), those instructions are only valid for the contents before the modification. At least one instruction cycle is needed (such as an NOP) between the modification of the interrupt request bits and the execution of the BBC and BBS instructions.
- (3) After the ADC and SBC instructions are executed (in the decimal mode), one instruction cycle (such as an NOP) is needed before the SEC, CLC, or CLD instruction is executed.
- (4) An NOP instruction is needed immediately after the execution of a PLP instruction.
- (5) In order to avoid noise and latch-up, connect a bypass capacitor ( $\approx 0.1 \text{ mF}$ ) directly between the Vcc pin-Vss pin, AVcc pin-Vss pin, and the Vcc pin-CNvss pin, using a thick wire.

**DATA REQUIRED FOR MASK ORDERS**

The following are necessary when ordering a mask ROM production:

- (1) Mask ROM Order Confirmation Form
- (2) Mask Specification Form
- (3) Data to be written to ROM, in EPROM form (32-pin DIP Type 27C101, three identical copies)

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**ABSOLUTE MAXIMUM RATINGS**

Symbol	Parameter	Conditions	Ratings	Unit
V <sub>CC</sub> , AV <sub>CC</sub>	Power source voltage V <sub>CC</sub> , AV <sub>CC</sub>	All voltages are based on V <sub>SS</sub> . Output transistors are cut off.	-0.3 to 6	V
V <sub>I</sub>	Input voltage CNV <sub>SS</sub>		-0.3 to 6	V
V <sub>I</sub>	Input voltage P00-P07, P10-P17, P20-P27, P30, P31, P40-P46, P64, P63, P70-P72, X <sub>IN</sub> , H <sub>SYNC</sub> , V <sub>SYNC</sub> , RESET		-0.3 to V <sub>CC</sub> + 0.3	V
V <sub>O</sub>	Output voltage P03, P10-P17, P20-P27, P30, P31, P52-P55, SOUT, SCLK, XOUT, OSC2		-0.3 to V <sub>CC</sub> + 0.3	V
V <sub>O</sub>	Output voltage P00-P02, P04-P07		-0.3 to 13	V
I <sub>OH</sub>	Circuit current P52-P55, P10, P03, P15-P17, P20-P27, P30, P31		0 to 1 (Note 1)	mA
I <sub>OL1</sub>	Circuit current P52-P55, P10, P03, P15-P17, P20-P27, SOUT, SCLK		0 to 2 (Note 2)	mA
I <sub>OL2</sub>	Circuit current P11-P14		0 to 6 (Note 2)	mA
I <sub>OL3</sub>	Circuit current P00-P02, P04-P07		0 to 1 (Note 2)	mA
I <sub>OL4</sub>	Circuit current P30, P31		0 to 10 (Note 3)	mA
P <sub>d</sub>	Power dissipation	T <sub>a</sub> = 25 °C	550	mW
T <sub>opr</sub>	Operating temperature		-10 to 70	°C
T <sub>stg</sub>	Storage temperature		-40 to 125	°C

**RECOMMENDED OPERATING CONDITIONS** (T<sub>a</sub> = -10 °C to 70 °C, V<sub>CC</sub> = 5 V ± 10 %, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
V <sub>CC</sub> , AV <sub>CC</sub>	Power source voltage (Note 4), During CPU, OSD, data slicer operation	4.5	5.0	5.5	V
V <sub>CC</sub> , AV <sub>CC</sub>	RAM hold voltage (when clock is stopped)	2.0		5.5	V
V <sub>SS</sub>	Power source voltage	0	0	0	V
V <sub>IH1</sub>	HIGH input voltage P00-P07, P10-P17, P20-P27, P30, P31, P40-P46, P64, P70-P72, H <sub>SYNC</sub> , V <sub>SYNC</sub> , RESET, X <sub>IN</sub> , P63	0.8V <sub>CC</sub>		V <sub>CC</sub>	V
V <sub>IH2</sub>	HIGH input voltage SCL1, SCL2, SDA1, SDA2	0.7V <sub>CC</sub>		V <sub>CC</sub>	V
V <sub>IL1</sub>	LOW input voltage P00-P07, P10-P17, P20-P27, P30, P31, P40-P46, P63, P64, P70-P72	0		0.4 V <sub>CC</sub>	V
V <sub>IL2</sub>	LOW input voltage SCL1, SCL2, SDA1, SDA2	0		0.3 V <sub>CC</sub>	V
V <sub>IL3</sub>	LOW input voltage (Note 6) RESET, X <sub>IN</sub> , OSC1, H <sub>SYNC</sub> , V <sub>SYNC</sub> , INT1, INT2, INT3, TIM2, TIM3, SCLK, S <sub>IN</sub>	0		0.2 V <sub>CC</sub>	V
I <sub>OH</sub>	HIGH average output current (Note 1) P52-P55, P10, P03, P15-P17, P20-P27, P30, P31			1	mA
I <sub>OL1</sub>	LOW average output current (Note 2) P52-P55, P10, P03, P15-P17, P20-P27, SOUT, SCLK			2	mA
I <sub>OL2</sub>	LOW average output current (Note 2) P11-P14			6	mA
I <sub>OL3</sub>	LOW average output current (Note 2) P00-P02, P04-P07			1	mA
I <sub>OL4</sub>	LOW average output current (Note 3) P30, P31			10	mA
f(X <sub>IN</sub> )	Oscillation frequency (for CPU operation) (Note 5) X <sub>IN</sub>	7.9	8.0	8.1	MHz
f(X <sub>CIN</sub> )	Oscillation frequency (for sub-clock operation) X <sub>CIN</sub>	29	32	35	kHz
f <sub>OSC</sub>	Oscillation frequency (for OSD) OSC1				MHz
		LC oscillating mode	11.0	27.0	
		Ceramic oscillating mode	26.5	27.0	27.5
f <sub>hs1</sub>	Input frequency TIM2, TIM3, INT1, INT2, INT3			100	kHz
f <sub>hs2</sub>	Input frequency SCLK			1	MHz
f <sub>hs3</sub>	Input frequency SCL1, SCL2			400	kHz
f <sub>hs4</sub>	Input frequency Horizontal sync. signal of video signal	15.262	15.734	16.206	kHz
V <sub>I</sub>	Input amplitude video signal CV <sub>IN</sub>	1.5	2.0	2.5	V



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**ELECTRIC CHARACTERISTICS** ( $V_{CC} = 5\text{ V} \pm 10\%$ ,  $V_{SS} = 0\text{ V}$ ,  $f(X_{IN}) = 8\text{ MHz}$ ,  $T_a = -10\text{ }^\circ\text{C}$  to  $70\text{ }^\circ\text{C}$ , unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min.	Typ.	Max.	
I <sub>CC</sub>	Power source current	System operation	V <sub>CC</sub> = 5.5 V, f(X <sub>IN</sub> ) = 8 MHz CRT OFF Data slicer OFF		15	30	mA
						30	
			V <sub>CC</sub> = 5.5 V, f(X <sub>IN</sub> ) = 0, f(X <sub>CIN</sub> ) = 32kHz, OSD OFF, Data slicer OFF, Low-power dissipation mode set (CM <sub>5</sub> = "0", CM <sub>6</sub> = "1")		60	200	μA
		Wait mode	V <sub>CC</sub> = 5.5 V, f(X <sub>IN</sub> ) = 8 MHz		2	4	mA
			V <sub>CC</sub> = 5.5 V, f(X <sub>IN</sub> ) = 0, f(X <sub>CIN</sub> ) = 32kHz, Low-power dissipation mode set (CM <sub>5</sub> = "0", CM <sub>6</sub> = "1")		25	100	μA
			Stop mode	V <sub>CC</sub> = 5.5 V, f(X <sub>IN</sub> ) = 0, f(X <sub>CIN</sub> ) = 0		1	10
V <sub>OH</sub>	HIGH output voltage	P52–P55, P10, P03, P15–P17, P20–P27, P30, P31	V <sub>CC</sub> = 4.5 V I <sub>OH</sub> = –0.5 mA	2.4			V
V <sub>OL</sub>	LOW output voltage	P52–P55, P10, SOUT, SCLK, P00–P07, P15–P17, P20–P27	V <sub>CC</sub> = 4.5 V I <sub>OL</sub> = 0.5 mA			0.4	V
		P30, P31	V <sub>CC</sub> = 4.5 V I <sub>OL</sub> = 10.0 mA			3.0	
	P11–P14	V <sub>CC</sub> = 4.5 V I <sub>OL</sub> = 3 mA			0.4		
			I <sub>OL</sub> = 6 mA			0.6	
V <sub>T+</sub> –V <sub>T–</sub>	Hysteresis (Note 6)	RESET, HSYNC, VSYNC, INT1, INT2, INT3, TIM2, TIM3, SIN, SCLK, SCL1, SCL2, SDA1, SDA2	V <sub>CC</sub> = 5.0 V		0.5	1.3	V
I <sub>IZH</sub>	HIGH input leak current	RESET, P03, P10–P17, P20–P27, P30, P31, P40–P46, P63, P64, P70–P72, HSYNC, VSYNC	V <sub>CC</sub> = 5.5 V V <sub>I</sub> = 5.5 V			5	μA
		P00–P02, P04–P07	V <sub>CC</sub> = 5.5 V V <sub>I</sub> = 12 V			10	
I <sub>IZL</sub>	LOW input leak current	RESET, P00–P07, P10–P17, P20–P27, P30, P31, P40–P46, P63, P64, P70–P72, HSYNC, VSYNC	V <sub>CC</sub> = 5.5 V V <sub>I</sub> = 0 V			5	μA
R <sub>BS</sub>	I <sup>2</sup> C-BUS-BUS switch connection resistor (between SCL1 and SCL2, SDA1 and SDA2)		V <sub>CC</sub> = 4.5 V			130	Ω

- Notes 1:** The total current that flows out of the IC must be 20 or less.  
**2:** The total input current to IC (I<sub>OL1</sub> + I<sub>OL2</sub> + I<sub>OL3</sub>) must be 20 mA or less.  
**3:** The total average input current for ports P30, P31 to IC must be 10 mA or less.  
**4:** Connect 0.1 μF or more capacitor externally between the power source pins V<sub>CC</sub>–V<sub>SS</sub> and AV<sub>CC</sub>–V<sub>SS</sub> so as to reduce power source noise.  
 Also connect 0.1 μF or more capacitor externally between the pins V<sub>CC</sub>–CNV<sub>SS</sub>.  
**5:** Use a quartz-crystal oscillator or a ceramic resonator for the CPU oscillation circuit. When using the data slicer, use 8 MHz.  
**6:** P16, P41–P44 have the hysteresis when these pins are used as interrupt input pins or timer input pins. P11–P14 have the hysteresis when these pins are used as multi-master I<sup>2</sup>C-BUS interface ports. P17 and P46 have the hysteresis when these pins are used as serial I/O pins.  
**7:** When using the sub-clock, set f<sub>CLK</sub> < f<sub>CPU</sub>/3.  
**8:** Pin names in each parameter is described as below.  
 (1) Dedicated pins: dedicated pin names.  
 (2) Duple-/triple-function ports  
 • When the same limits: I/O port name.  
 • When the limits of functions except ports are different from I/O port limits: function pin name.

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**A-D CONVERTER CHARACTERISTICS**

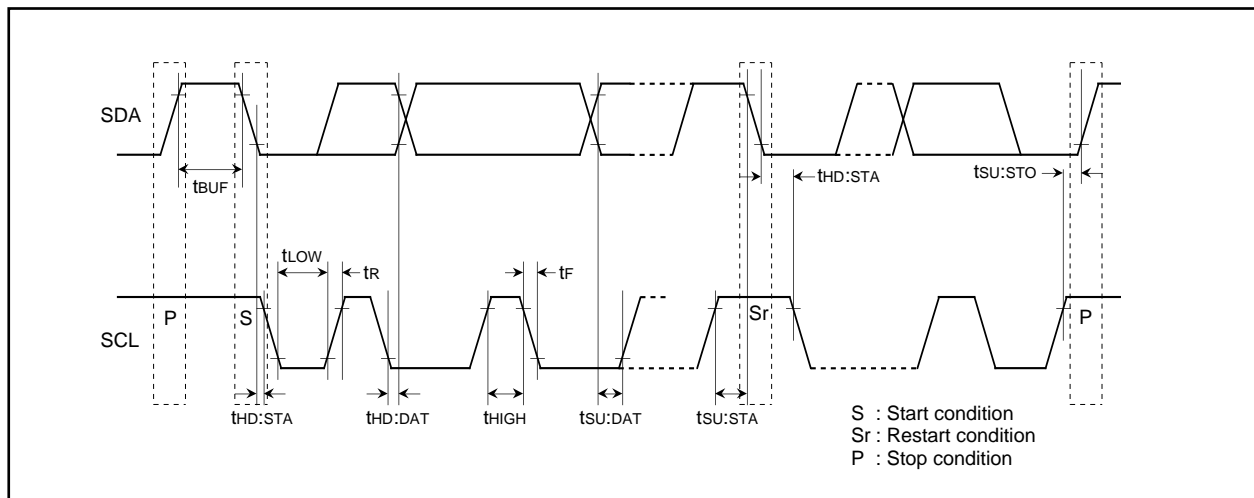
(VCC = 5 V ± 10 %, VSS = 0 V, f(XIN) = 8 MHz, Ta = -10 °C to 70 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
—	Resolution				8	bits
—	Non-linearity error		0		±2	LSB
—	Differential non-linearity error		0		±0.9	LSB
VOT	Zero transition error	VCC = 5.12V IOL (SUM) = 0mA	0		2	LSB
VFST	Full-scale transition error	VCC = 5.12V	0		4	LSB
TCONV	Conversion time		12.25		12.5	μs
VREF	Reference voltage				VCC	V
RLADDER	Ladder resistor			25		kΩ
VIA	Analog input current		0		VREF	V

**MULTI-MASTER I<sup>2</sup>C-BUS BUS LINE CHARACTERISTICS**

Symbol	Parameter	Standard clock mode		High-speed clock mode		Unit
		Min.	Max.	Min.	Max.	
tBUF	Bus free time	4.7		1.3		μs
tHD:STA	Hold time for START condition	4.0		0.6		μs
tLOW	LOW period of SCL clock	4.7		1.3		μs
tR	Rising time of both SCL and SDA signals		1000	20+0.1Cb	300	ns
tHD:DAT	Data hold time	0		0	0.9	μs
tHIGH	HIGH period of SCL clock	4.0		0.6		μs
tF	Falling time of both SCL and SDA signals		300	20+0.1Cb	300	ns
tSU:DAT	Data set-up time	250		100		ns
tSU:STA	Set-up time for repeated START condition	4.7		0.6		μs
tSU:STO	Set-up time for STOP condition	4.0		0.6		μs

**Note:** Cb = total capacitance of 1 bus line



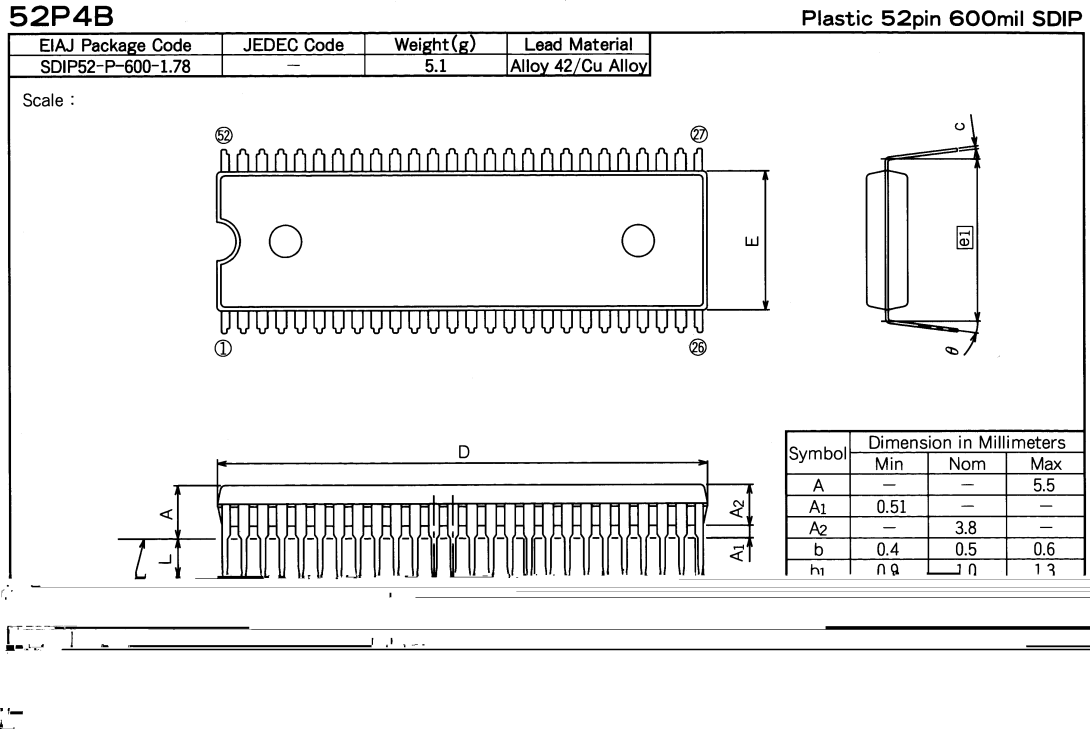
**Fig. 101. Definition Diagram of Timing on Multi-master I<sup>2</sup>C-BUS**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**PACKAGE OUTLINE**



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

GZZ-SH11-19B < 69A0 >

**740 FAMILY MASK ROM CONFIRMATION FORM**  
**SINGLE-CHIP MICROCOMPUTER M37274MA-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)

<input type="checkbox"/> <b>27C101</b>
EPROM address
0000 <sub>16</sub> Product name ASCII code : 'M37274MA-'
0000F <sub>16</sub>
0600 <sub>16</sub> ROM (40 K)
0FFFF <sub>16</sub>
10800 <sub>16</sub> OSD ROM
1E41F <sub>16</sub>
1FFFF <sub>16</sub>

- (1) Set "FF<sub>16</sub>" in the shaded area and in the addresses at which OSD ROM data does not present (refer to page 4/4).
- (2) Write the ASCII codes that indicates the product name of "M37274MA-" to addresses 0000<sub>16</sub> to 0000F<sub>16</sub>.

EPROM data check item (Refer the EPROM data and check "✓" in the appropriate box)

- Do you set "FF<sub>16</sub>" in the shaded area and in the addresses at which OSD ROM data does not present (refer to page 4/4)? → Yes
- Do you write the ASCII codes that indicates the product name of "M37274MA-" to addresses 0000<sub>16</sub> to 0000F<sub>16</sub>? → 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 (52P4B for M37274MA-XXXSP) and attach to the mask ROM confirmation form.

**PRELIMINARY**

Notice: This is not a final specification.  
Some parametric limits are subject to change.

GZZ-SH11-19B < 69A0 >

**740 FAMILY MASK ROM CONFIRMATION FORM  
SINGLE-CHIP MICROCOMPUTER M37274MA-XXXSP  
MITSUBISHI ELECTRIC**

**Writing the product name and character ROM data onto EPROMs**

Addresses 00000<sub>16</sub> to 0000F<sub>16</sub> store the product name, and addresses 10800<sub>16</sub> to 1FFFF<sub>16</sub> 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 'M37274MA-' are listed on the right.  
The addresses and data are in hexadecimal notation.

Address		Address	
0000 <sub>16</sub>	'M' = 4 D <sub>16</sub>	0008 <sub>16</sub>	'-' = 2 D <sub>16</sub>
0001 <sub>16</sub>	'3' = 3 3 <sub>16</sub>	0009 <sub>16</sub>	F F <sub>16</sub>
0002 <sub>16</sub>	'7' = 3 7 <sub>16</sub>	000A <sub>16</sub>	F F <sub>16</sub>
0003 <sub>16</sub>	'2' = 3 2 <sub>16</sub>	000B <sub>16</sub>	F F <sub>16</sub>
0004 <sub>16</sub>	'7' = 3 7 <sub>16</sub>	000C <sub>16</sub>	F F <sub>16</sub>
0005 <sub>16</sub>	'4' = 3 4 <sub>16</sub>	000D <sub>16</sub>	F F <sub>16</sub>
0006 <sub>16</sub>	'M' = 4 D <sub>16</sub>	000E <sub>16</sub>	F F <sub>16</sub>
0007 <sub>16</sub>	'A' = 4 1 <sub>16</sub>	000F <sub>16</sub>	F F <sub>16</sub>

2. Inputting the character ROM  
Input the character ROM data to character ROM. For the character ROM data, see the next page and on.

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

GZZ-SH11-19B < 69A0 >

**SERIES MELPS 740 MASK ROM CONFIRMATION FORM**  
**SINGLE-CHIP MICROCOMPUTER M37274MA-XXXSP**  
**MITSUBISHI ELECTRIC**


Mask ROM number	
-----------------	--

Font data must be stored in the proper OSD ROM address according to the following table.

(1) OSD ROM address of character font data

OSD ROM address bit	AD16	AD15	AD14	AD13	AD12	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
Line number / Character code / Font bit	1	0	Line number						0	Character code							Font bit


Line number = 02<sub>16</sub> to 15<sub>16</sub>  
 Character code = 00<sub>16</sub> to FF<sub>16</sub>  
 Font bit = 0 : Left font  
           1 : Right font

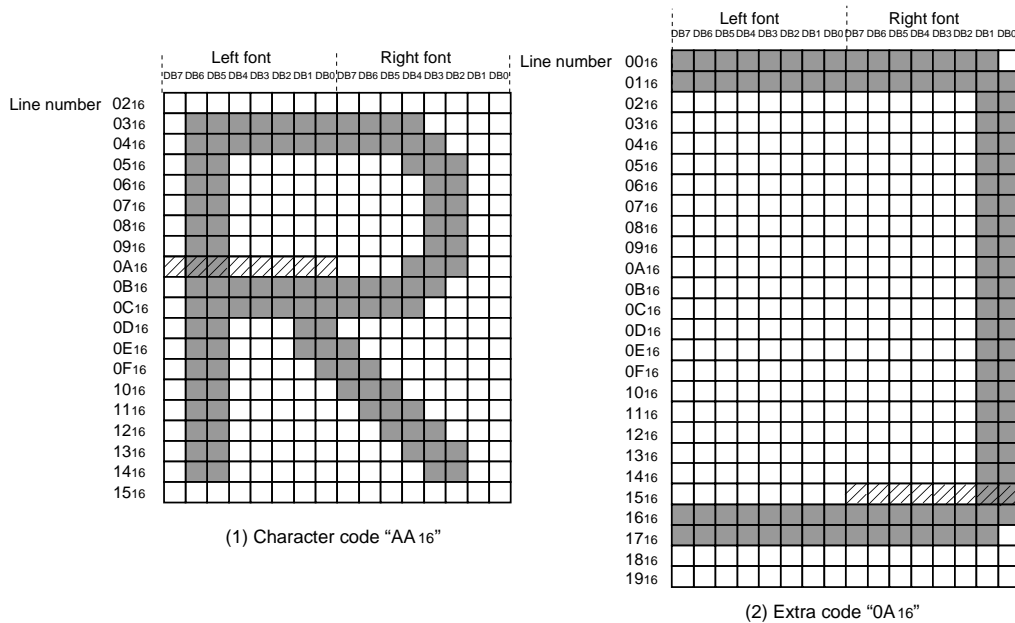
Example) The font data "60" (shaded area ) of the character code "AA 16" is stored in address  
 1:0 0 1 0:1 0 0 1:0 1 0 1:0 1 0 0:2 =12954<sub>16</sub>.

(2) OSD ROM address of extra font data

OSD ROM address bit	AD16	AD15	AD14	AD13	AD12	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
Line number / Extra code / Font bit	1	1	Line number						0	0	0	0	Extra code				Font bit

Line number = 00<sub>16</sub> to 19<sub>16</sub>  
 Extra code = 00<sub>16</sub> to 0F<sub>16</sub>  
 Font bit = 0 : Left font  
           1 : Right font

Example) The font data "03" (shaded area ) of the extra code "0A 16" is stored in address  
 1:1 1 0 1:0 1 0 0:0 0 0 1:0 1 0 1:2 =1D415<sub>16</sub>.



(3/4)

**PRELIMINARY**

Notice: This is not a final specification.  
Some parametric limits are subject to change.

GZZ-SH11-19B < 69A0 >

**740 FAMILY MASK ROM CONFIRMATION FORM  
SINGLE-CHIP MICROCOMPUTER M37274MA-XXXSP  
MITSUBISHI ELECTRIC**

The following OSD ROM addresses must be set "FF." There are no font data in these addresses.

10A00 <sub>16</sub> to 10BFF <sub>16</sub>	13200 <sub>16</sub> to 133FF <sub>16</sub>	18020 <sub>16</sub> to 183FF <sub>16</sub>	1B020 <sub>16</sub> to 1B3FF <sub>16</sub>
10E00 <sub>16</sub> to 10FFF <sub>16</sub>	13600 <sub>16</sub> to 137FF <sub>16</sub>	18420 <sub>16</sub> to 187FF <sub>16</sub>	1B420 <sub>16</sub> to 1B7FF <sub>16</sub>
11200 <sub>16</sub> to 113FF <sub>16</sub>	13A00 <sub>16</sub> to 13BFF <sub>16</sub>	18820 <sub>16</sub> to 18BFF <sub>16</sub>	1B820 <sub>16</sub> to 1BBFF <sub>16</sub>
11600 <sub>16</sub> to 117FF <sub>16</sub>	13E00 <sub>16</sub> to 13FFF <sub>16</sub>	18C20 <sub>16</sub> to 18FFF <sub>16</sub>	1BC20 <sub>16</sub> to 1BFFF <sub>16</sub>
11A00 <sub>16</sub> to 11BFF <sub>16</sub>	14200 <sub>16</sub> to 143FF <sub>16</sub>	19020 <sub>16</sub> to 193FF <sub>16</sub>	1C020 <sub>16</sub> to 1C3FF <sub>16</sub>
11E00 <sub>16</sub> to 11FFF <sub>16</sub>	14600 <sub>16</sub> to 147FF <sub>16</sub>	19420 <sub>16</sub> to 197FF <sub>16</sub>	1C420 <sub>16</sub> to 1C7FF <sub>16</sub>
12200 <sub>16</sub> to 123FF <sub>16</sub>	14A00 <sub>16</sub> to 14BFF <sub>16</sub>	19820 <sub>16</sub> to 19BFF <sub>16</sub>	1C820 <sub>16</sub> to 1CBFF <sub>16</sub>
12600 <sub>16</sub> to 127FF <sub>16</sub>	14E00 <sub>16</sub> to 14FFF <sub>16</sub>	19C20 <sub>16</sub> to 19FFF <sub>16</sub>	1CC20 <sub>16</sub> to 1CFFF <sub>16</sub>
12A00 <sub>16</sub> to 12BFF <sub>16</sub>	15200 <sub>16</sub> to 153FF <sub>16</sub>	1A020 <sub>16</sub> to 1A3FF <sub>16</sub>	1D020 <sub>16</sub> to 1D3FF <sub>16</sub>
12E00 <sub>16</sub> to 12FFF <sub>16</sub>	15600 <sub>16</sub> to 17FFF <sub>16</sub>	1A420 <sub>16</sub> to 1A7FF <sub>16</sub>	1D420 <sub>16</sub> to 1D7FF <sub>16</sub>
		1A820 <sub>16</sub> to 1ABFF <sub>16</sub>	1D820 <sub>16</sub> to 1DBFF <sub>16</sub>
		1AC20 <sub>16</sub> to 1AFF <sub>16</sub>	1DC20 <sub>16</sub> to 1DFFF <sub>16</sub>
			1E020 <sub>16</sub> to 1E3FF <sub>16</sub>

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

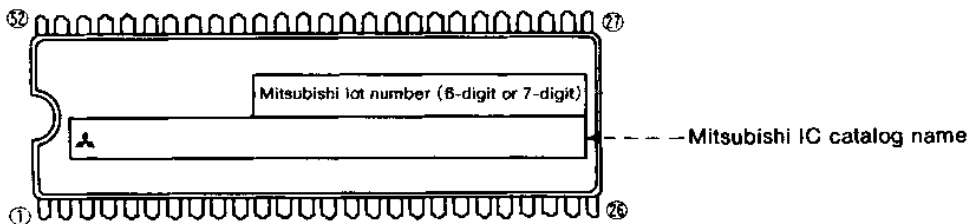
SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**52P4B (52-PIN SHRINK DIP) MARK SPECIFICATION FORM**

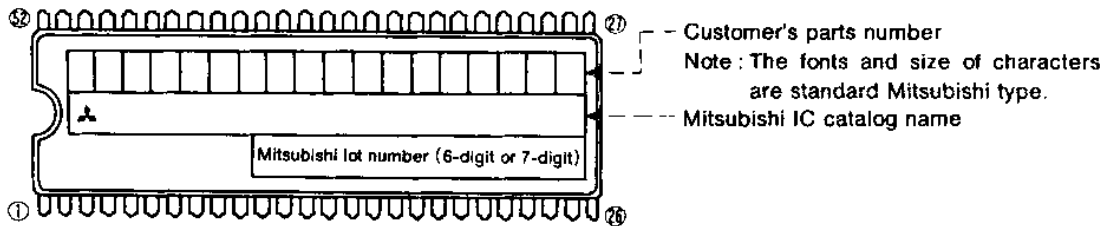
Mitsubishi IC catalog name

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.

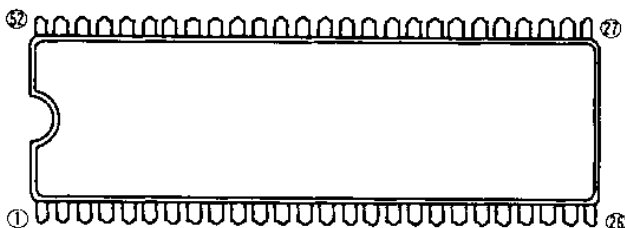
3: Customer's parts number can be up to 18 characters:

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.



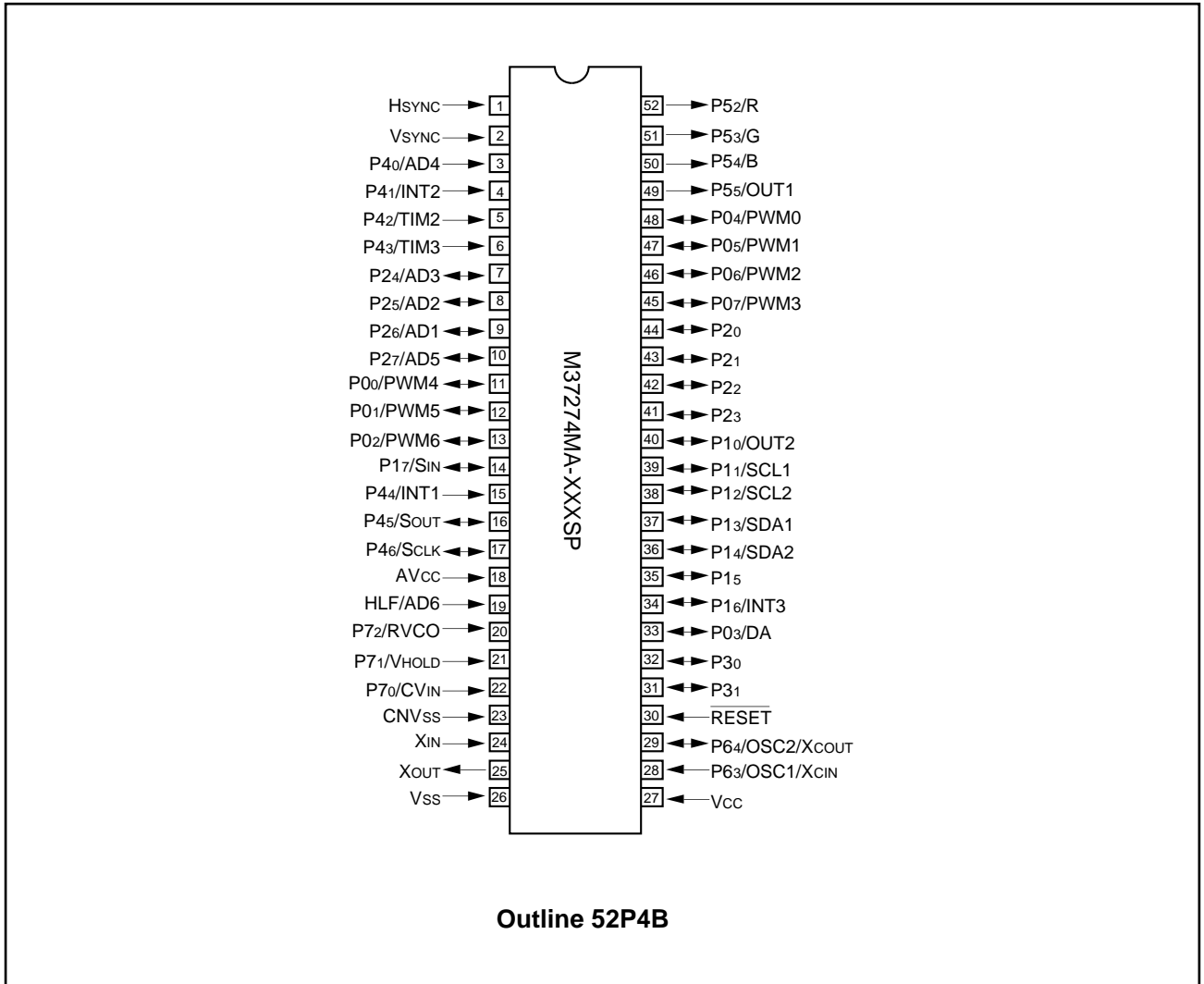
**PRELIMINARY**

Notice: This is not a final specification.  
Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
and ON-SCREEN DISPLAY CONTROLLER

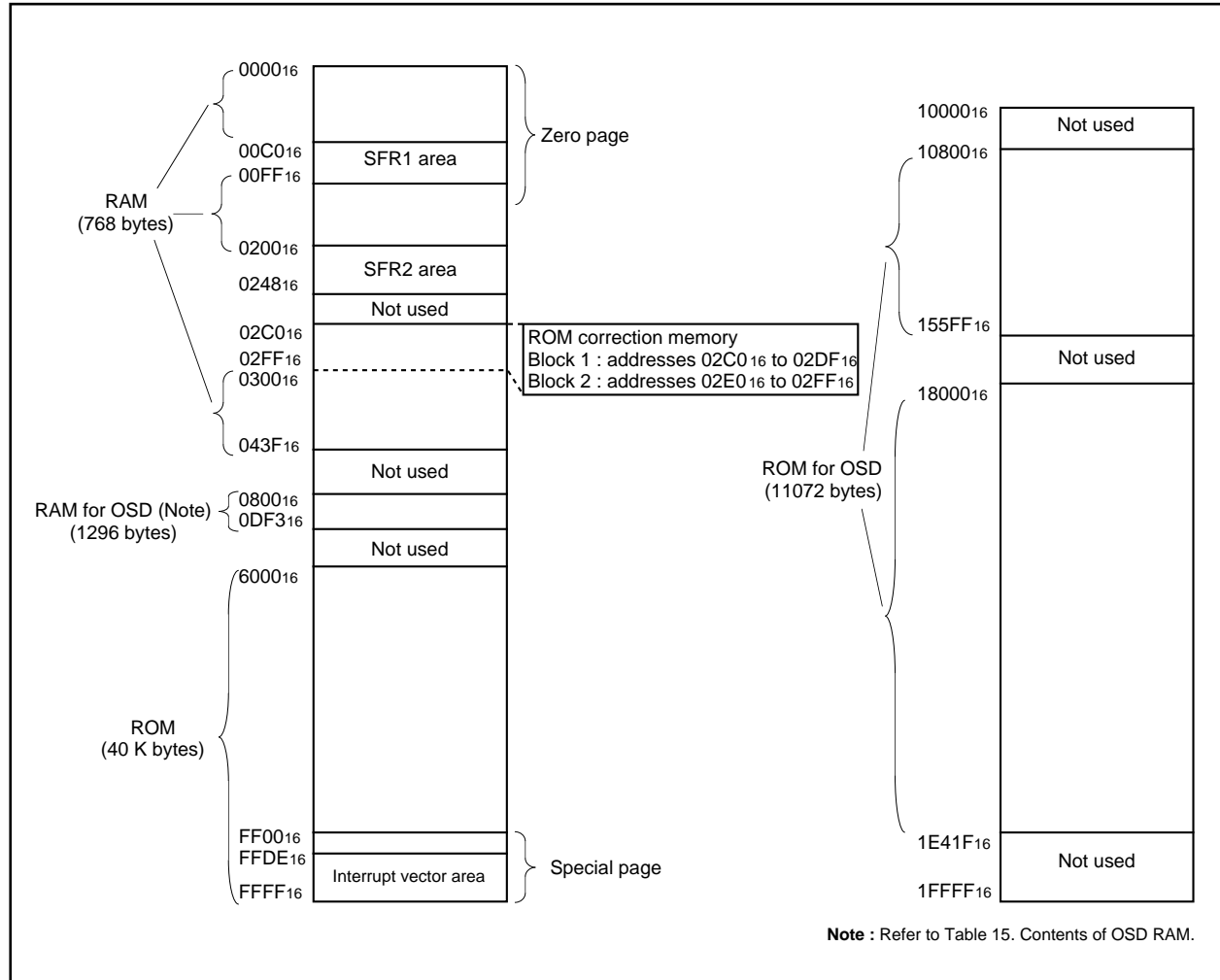
**APPENDIX**

**Pin Configuration (TOP VIEW)**



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**Memory Map**



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**Memory Map of Special Function Register (SFR)**

■ SFR1 area (addresses C0<sub>16</sub> to DF<sub>16</sub>)

< Bit allocation >



- ☐ : No function bit
- 0 : Fix to this bit to "0"  
(do not write to "1")
- 1 : Fix to this bit to "1"  
(do not write to "0")

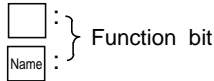
< State immediately after reset >

- 0 : "0" immediately after reset
- 1 : "1" immediately after reset
- ? : Indeterminate immediately after reset

Address	Register	Bit allocation								State immediately after reset								
		b7							b0	b7							b0	
C0 <sub>16</sub>	Port P0 (P0)																	
C1 <sub>16</sub>	Port P0 direction register (D0)									00 <sub>16</sub>								
C2 <sub>16</sub>	Port P1 (P1)																	
C3 <sub>16</sub>	Port P1 direction register (D1)									00 <sub>16</sub>								
C4 <sub>16</sub>	Port P2 (P2)																	
C5 <sub>16</sub>	Port P2 direction register (D2)									00 <sub>16</sub>								
C6 <sub>16</sub>	Port P3 (P3)																	
C7 <sub>16</sub>	Port P3 direction register (D3)		T3SC							00 <sub>16</sub>								
C8 <sub>16</sub>	Port P4 (P4)																	
C9 <sub>16</sub>	Port P4 direction register (D4)								0	00 <sub>16</sub>								
CA <sub>16</sub>	Port P5 (P5)																	
CB <sub>16</sub>	OSD port control register (PF)	0	OUT2	OUT1	B	G	R	0	0	00 <sub>16</sub>								
CC <sub>16</sub>	Port P6 (P6)																	
CD <sub>16</sub>	Port P7 (P7)									0	0	0	0	0	0	?	?	?
CE <sub>16</sub>	OSD control register (OC)	OC7	OC6	OC5	OC4	OC3	OC2	OC1	OC0	00 <sub>16</sub>								
CF <sub>16</sub>	Horizontal position register (HP)	HP7	HP6	HP5	HP4	HP3	HP2	HP1	HP0	00 <sub>16</sub>								
D0 <sub>16</sub>	Block control register 1 (BC <sub>1</sub> )	BC <sub>18</sub>	BC <sub>17</sub>	BC <sub>16</sub>	BC <sub>15</sub>	BC <sub>14</sub>	BC <sub>13</sub>	BC <sub>12</sub>	BC <sub>11</sub>									
D1 <sub>16</sub>	Block control register 2 (BC <sub>2</sub> )	BC <sub>28</sub>	BC <sub>27</sub>	BC <sub>26</sub>	BC <sub>25</sub>	BC <sub>24</sub>	BC <sub>23</sub>	BC <sub>22</sub>	BC <sub>21</sub>									
D2 <sub>16</sub>	Block control register 3 (BC <sub>3</sub> )	BC <sub>38</sub>	BC <sub>37</sub>	BC <sub>36</sub>	BC <sub>35</sub>	BC <sub>34</sub>	BC <sub>33</sub>	BC <sub>32</sub>	BC <sub>31</sub>									
D3 <sub>16</sub>	Block control register 4 (BC <sub>4</sub> )	BC <sub>48</sub>	BC <sub>47</sub>	BC <sub>46</sub>	BC <sub>45</sub>	BC <sub>44</sub>	BC <sub>43</sub>	BC <sub>42</sub>	BC <sub>41</sub>									
D4 <sub>16</sub>	Block control register 5 (BC <sub>5</sub> )	BC <sub>58</sub>	BC <sub>57</sub>	BC <sub>56</sub>	BC <sub>55</sub>	BC <sub>54</sub>	BC <sub>53</sub>	BC <sub>52</sub>	BC <sub>51</sub>									
D5 <sub>16</sub>	Block control register 6 (BC <sub>6</sub> )	BC <sub>68</sub>	BC <sub>67</sub>	BC <sub>66</sub>	BC <sub>65</sub>	BC <sub>64</sub>	BC <sub>63</sub>	BC <sub>62</sub>	BC <sub>61</sub>									
D6 <sub>16</sub>	Block control register 7 (BC <sub>7</sub> )	BC <sub>78</sub>	BC <sub>77</sub>	BC <sub>76</sub>	BC <sub>75</sub>	BC <sub>74</sub>	BC <sub>73</sub>	BC <sub>72</sub>	BC <sub>71</sub>									
D7 <sub>16</sub>	Block control register 8 (BC <sub>8</sub> )	BC <sub>88</sub>	BC <sub>87</sub>	BC <sub>86</sub>	BC <sub>85</sub>	BC <sub>84</sub>	BC <sub>83</sub>	BC <sub>82</sub>	BC <sub>81</sub>									
D8 <sub>16</sub>	Block control register 9 (BC <sub>9</sub> )	BC <sub>98</sub>	BC <sub>97</sub>	BC <sub>96</sub>	BC <sub>95</sub>	BC <sub>94</sub>	BC <sub>93</sub>	BC <sub>92</sub>	BC <sub>91</sub>									
D9 <sub>16</sub>	Block control register 10 (BC <sub>10</sub> )	BC <sub>108</sub>	BC <sub>107</sub>	BC <sub>106</sub>	BC <sub>105</sub>	BC <sub>104</sub>	BC <sub>103</sub>	BC <sub>102</sub>	BC <sub>101</sub>									
DA <sub>16</sub>	Block control register 11 (BC <sub>11</sub> )	BC <sub>118</sub>	BC <sub>117</sub>	BC <sub>116</sub>	BC <sub>115</sub>	BC <sub>114</sub>	BC <sub>113</sub>	BC <sub>112</sub>	BC <sub>111</sub>									
DB <sub>16</sub>	Block control register 12 (BC <sub>12</sub> )	BC <sub>128</sub>	BC <sub>127</sub>	BC <sub>126</sub>	BC <sub>125</sub>	BC <sub>124</sub>	BC <sub>123</sub>	BC <sub>122</sub>	BC <sub>121</sub>									
DC <sub>16</sub>																		
DD <sub>16</sub>																		
DE <sub>16</sub>																		
DF <sub>16</sub>																		

■ SFR1 area (addresses E0<sub>16</sub> to FF<sub>16</sub>)

< Bit allocation >



☐ : No function bit

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

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

< State immediately after reset >

0 : "0" immediately after reset

1 : "1" immediately after reset

? : Indeterminate immediately  
 after reset

Address	Register	Bit allocation								State immediately after reset								
		b7							b0	b7							b0	
E0 <sub>16</sub>	Caption position register (CP)	1	0	0	CP4	CP3	CP2	CP1	CP0									00 <sub>16</sub>
E1 <sub>16</sub>	Start bit position register (SP)	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0									00 <sub>16</sub>
E2 <sub>16</sub>	Window register (WN)	0	0	WN5	WN4	WN3	WN2	WN1	WN0									00 <sub>16</sub>
E3 <sub>16</sub>	Sync slice register (SSL)	SSL7	0	0	0	0	1	0	1									00 <sub>16</sub>
E4 <sub>16</sub>	Data register 1 (CD1)																	00 <sub>16</sub>
E5 <sub>16</sub>	Data register 2 (CD2)																	00 <sub>16</sub>
E6 <sub>16</sub>	Clock run-in register 1 (CR1)	0	1	0	1	CR13	CR12	CR11	CR10									00 <sub>16</sub>
E7 <sub>16</sub>	Clock run-in register 2 (CR2)	1	0	0	1	1	1	CR21	1									00 <sub>16</sub>
E8 <sub>16</sub>	Clock run-in detect register 1 (CRD1)	CRD17	CRD15	CRD15	CRD15	CRD15												00 <sub>16</sub>
E9 <sub>16</sub>	Clock run-in detect register 2 (CRD2)	CRD27	CRD25	CRD25	CRD25	CRD25	CRD22	CRD21	CRD20									00 <sub>16</sub>
EA <sub>16</sub>	Data slicer control register 1 (DSC1)	DSC17	0	DSC15	0	0	DSC12	DSC11	DSC10	?	0	?	0	0	0	0	0	
EB <sub>16</sub>	Data slicer control register 2 (DSC2)	DSC27	0	DSC25	0	0	DSC22	DSC21	DSC20	?	0	?	0	0	?	0	0	
EC <sub>16</sub>	Data register 3 (CD3)																	00 <sub>16</sub>
ED <sub>16</sub>	Data register 4 (CD4)																	00 <sub>16</sub>
EE <sub>16</sub>	A-D conversion register (AD)																	?
EF <sub>16</sub>	A-D control register (ADCON)	0			ADVREF	ADSTR	ADIN2	ADIN1	ADIN0	0	?	0	0	1	0	0	0	
F0 <sub>16</sub>	Timer 1 (TM1)																	FF <sub>16</sub>
F1 <sub>16</sub>	Timer 2 (TM2)																	07 <sub>16</sub>
F2 <sub>16</sub>	Timer 3 (TM3)																	FF <sub>16</sub>
F3 <sub>16</sub>	Timer 4 (TM4)																	07 <sub>16</sub>
F4 <sub>16</sub>	Timer mode register 1 (TM1)	TM17	TM16	TM15	TM14	TM13	TM12	TM11	TM10									00 <sub>16</sub>
F5 <sub>16</sub>	Timer mode register 2 (TM2)	TM27	TM26	TM25	TM24	TM23	TM22	TM21	TM20									00 <sub>16</sub>
F6 <sub>16</sub>	I <sup>2</sup> C data shift register (S0)																	?
F7 <sub>16</sub>	I <sup>2</sup> C address register (S0D)	SAD6	SAD5	SAD4	SAD3	SAD2	SAD1	SAD0	RBW									00 <sub>16</sub>
F8 <sub>16</sub>	I <sup>2</sup> C status register (S1)	MST	TRX	BB	PIN	AL	AAS	ADO	LRB	0	0	0	1	0	0	0	?	
F9 <sub>16</sub>	I <sup>2</sup> C control register (S1D)	BSEL1	BSEL0	<sup>10 BIT</sup> SAD	ALS	ES0	BC2	BC1	BC0									00 <sub>16</sub>
FA <sub>16</sub>	I <sup>2</sup> C clock control register (S2)	ACK	ACK BIT	FAST MODE	CCR4	CCR3	CCR2	CCR1	CCR0									00 <sub>16</sub>
FB <sub>16</sub>	CPU mode register (CPUM)	CM7	CM6	CM5	1	1	CM2	0	0	0	0	1	1	1	1	0	0	
FC <sub>16</sub>	Interrupt request register 1 (IREQ1)		ADR	VSCR	CRTR	TM4R	TM3R	TM2R	TM1R									00 <sub>16</sub>
FD <sub>16</sub>	Interrupt request register 2 (IREQ2)	0	T56R	IICR	INT2R	IMSR	SIOR	DSR	INT1R									00 <sub>16</sub>
FE <sub>16</sub>	Interrupt control register 1 (ICON1)		ADE	VSCR	CRTE	TM4E	TM3E	TM2E	TM1E									00 <sub>16</sub>
FF <sub>16</sub>	Interrupt control register 2 (ICON2)	T56S	T56E	IICE	INT2E	IMSE	SIOE	DSE	INT1E									00 <sub>16</sub>

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

■ SFR2 area (addresses 200<sub>16</sub> to 21F<sub>16</sub>)

< Bit allocation >

: } Function bit  
 Name : }

: No function bit

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

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

< State immediately after reset >

: "0" immediately after reset

: "1" immediately after reset

: Indeterminate immediately  
 after reset

Address	Register	Bit allocation								State immediately after reset								
		b7	b6	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0	
200 <sub>16</sub>	PWM0 register (PWM0)																	?
201 <sub>16</sub>	PWM1 register (PWM1)																	?
202 <sub>16</sub>	PWM2 register (PWM2)																	?
203 <sub>16</sub>	PWM3 register (PWM3)																	?
204 <sub>16</sub>	PWM4 register (PWM4)																	?
205 <sub>16</sub>	PWM5 register (PWM5)																	?
206 <sub>16</sub>	PWM6 register (PWM6)																	?
207 <sub>16</sub>																		?
208 <sub>16</sub>	Clock run-in detect register 3 (CRD3)	CRD35	CRD34	CRD33	CRD32	CRD31												00 <sub>16</sub>
209 <sub>16</sub>	Clock run-in register (CR3)		CR36	CR35	CR34	CR33	CR32	CR31	CR30									?
20A <sub>16</sub>	PWM mode register 1 (PN)					PN3	PN2	PN1	PN0	?	?	?	?	0	0	0	0	
20B <sub>16</sub>	PWM mode register 2 (PW)	0	PW6	PW5	PW4	PW3	PW2	PW1	PW0									00 <sub>16</sub>
20C <sub>16</sub>	Timer 5 (TM5)																	07 <sub>16</sub>
20D <sub>16</sub>	Timer 6 (TM6)																	FF <sub>16</sub>
20E <sub>16</sub>																		00 <sub>16</sub>
20F <sub>16</sub>	Sync pulse counter register (SYC)			SYC5	SYC4	SYC3	SYC2	SYC1	SYC0									00 <sub>16</sub>
210 <sub>16</sub>	Data slicer control register 3 (DSC3)	DSC37	DSC36	DSC35	DSC34	DSC33	DSC32	DSC31	DSC30									00 <sub>16</sub>
211 <sub>16</sub>																		?
212 <sub>16</sub>	Interrupt input polarity register (IP)	AD/INT3 SEL	INT3 POL	0	INT2 POL	INT1 POL	0	0	0									00 <sub>16</sub>
213 <sub>16</sub>	Serial I/O mode register (SM)	0	0	SM5	SM4	SM3	SM2	SM1	SM0									00 <sub>16</sub>
214 <sub>16</sub>	Serial I/O register (SIO)																	?
215 <sub>16</sub>		0																?
216 <sub>16</sub>	Clock source control register (CS)	0	CS6	CS5	CS4	CS3	CS2	CS1	CS0									00 <sub>16</sub>
217 <sub>16</sub>	I/O polarity control register (PC)	PC7	PC6	PC5	PC4		PC2	PC1	PC0	1	0	0	0	0	0	0	0	
218 <sub>16</sub>	Raster color register (RC)	RC7	RC6	RC5			RC2	RC1	RC0									00 <sub>16</sub>
219 <sub>16</sub>	Extra font color register (EC)				0	0	EC2	EC1	EC0									00 <sub>16</sub>
21A <sub>16</sub>					0	0	0	0	0									00 <sub>16</sub>
21B <sub>16</sub>	Border color register (FC)				0	0	FC2	FC1	FC0									00 <sub>16</sub>
21C <sub>16</sub>	Window H register 1 (WH1)	WH17	WH16	WH15	WH14	WH13	WH12	WH11	WH10									?
21D <sub>16</sub>	Window L register 1 (WH1)	WL17	WL16	WL15	WL14	WL13	WL12	WL11	WL10									?
21E <sub>16</sub>	Window H register 2 (WH2)							WH21	WH20									?
21F <sub>16</sub>	Window L register 2 (WH2)							WL21	WL20									?

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

■ SFR2 area (addresses 220<sub>16</sub> to 248<sub>16</sub>)

< Bit allocation >

: } Function bit  
 Name : }

: No function bit

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

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

< State immediately after reset >

0 : "0" immediately after reset

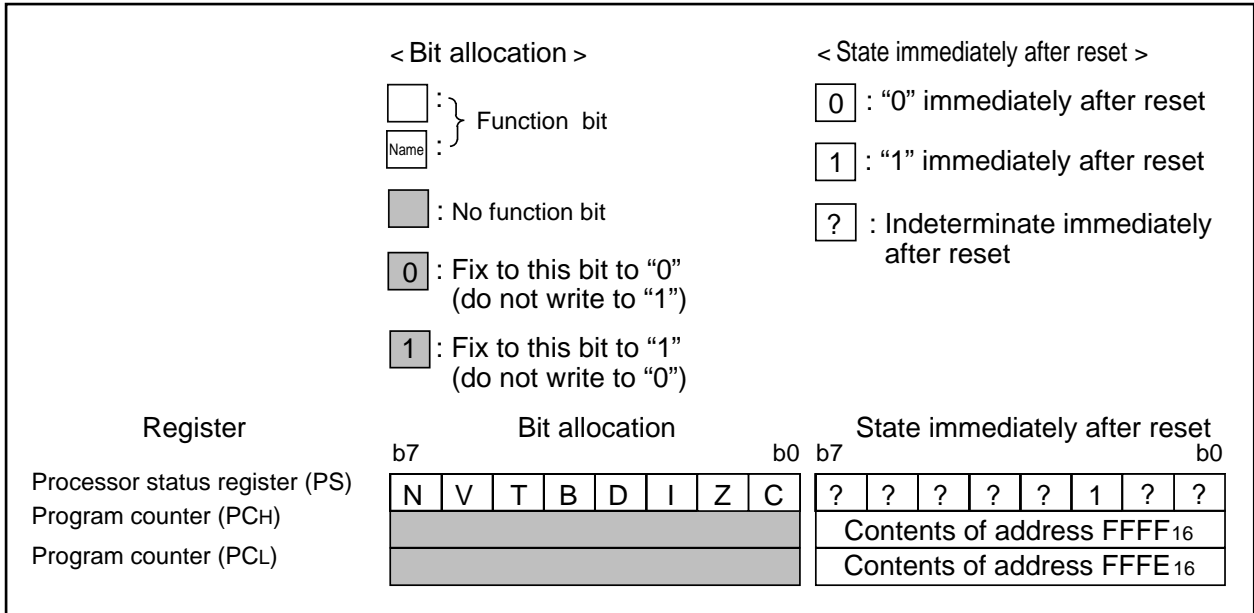
1 : "1" immediately after reset

? : Indeterminate immediately  
 after reset

Address	Register	Bit allocation								State immediately after reset							
		b7							b0	b7							b0
220 <sub>16</sub>	Vertical position register 1 <sub>1</sub> (VP1 <sub>1</sub> )	VP1 <sub>18</sub>	VP1 <sub>17</sub>	VP1 <sub>16</sub>	VP1 <sub>15</sub>	VP1 <sub>14</sub>	VP1 <sub>13</sub>	VP1 <sub>12</sub>	VP1 <sub>11</sub>	?							?
221 <sub>16</sub>	Vertical position register 1 <sub>2</sub> (VP1 <sub>2</sub> )	VP1 <sub>28</sub>	VP1 <sub>27</sub>	VP1 <sub>26</sub>	VP1 <sub>25</sub>	VP1 <sub>24</sub>	VP1 <sub>23</sub>	VP1 <sub>22</sub>	VP1 <sub>21</sub>	?							?
222 <sub>16</sub>	Vertical position register 1 <sub>3</sub> (VP1 <sub>3</sub> )	VP1 <sub>38</sub>	VP1 <sub>37</sub>	VP1 <sub>36</sub>	VP1 <sub>35</sub>	VP1 <sub>34</sub>	VP1 <sub>33</sub>	VP1 <sub>32</sub>	VP1 <sub>31</sub>	?							?
223 <sub>16</sub>	Vertical position register 1 <sub>4</sub> (VP1 <sub>4</sub> )	VP1 <sub>48</sub>	VP1 <sub>47</sub>	VP1 <sub>46</sub>	VP1 <sub>45</sub>	VP1 <sub>44</sub>	VP1 <sub>43</sub>	VP1 <sub>42</sub>	VP1 <sub>41</sub>	?							?
224 <sub>16</sub>	Vertical position register 1 <sub>5</sub> (VP1 <sub>5</sub> )	VP1 <sub>58</sub>	VP1 <sub>57</sub>	VP1 <sub>56</sub>	VP1 <sub>55</sub>	VP1 <sub>54</sub>	VP1 <sub>53</sub>	VP1 <sub>52</sub>	VP1 <sub>51</sub>	?							?
225 <sub>16</sub>	Vertical position register 1 <sub>6</sub> (VP1 <sub>6</sub> )	VP1 <sub>68</sub>	VP1 <sub>67</sub>	VP1 <sub>66</sub>	VP1 <sub>65</sub>	VP1 <sub>64</sub>	VP1 <sub>63</sub>	VP1 <sub>62</sub>	VP1 <sub>61</sub>	?							?
226 <sub>16</sub>	Vertical position register 1 <sub>7</sub> (VP1 <sub>7</sub> )	VP1 <sub>78</sub>	VP1 <sub>77</sub>	VP1 <sub>76</sub>	VP1 <sub>75</sub>	VP1 <sub>74</sub>	VP1 <sub>73</sub>	VP1 <sub>72</sub>	VP1 <sub>71</sub>	?							?
227 <sub>16</sub>	Vertical position register 1 <sub>8</sub> (VP1 <sub>8</sub> )	VP1 <sub>88</sub>	VP1 <sub>87</sub>	VP1 <sub>86</sub>	VP1 <sub>85</sub>	VP1 <sub>84</sub>	VP1 <sub>83</sub>	VP1 <sub>82</sub>	VP1 <sub>81</sub>	?							?
228 <sub>16</sub>	Vertical position register 1 <sub>9</sub> (VP1 <sub>9</sub> )	VP1 <sub>98</sub>	VP1 <sub>97</sub>	VP1 <sub>96</sub>	VP1 <sub>95</sub>	VP1 <sub>94</sub>	VP1 <sub>93</sub>	VP1 <sub>92</sub>	VP1 <sub>91</sub>	?							?
229 <sub>16</sub>	Vertical position register 1 <sub>10</sub> (VP1 <sub>10</sub> )	VP1 <sub>108</sub>	VP1 <sub>107</sub>	VP1 <sub>106</sub>	VP1 <sub>105</sub>	VP1 <sub>104</sub>	VP1 <sub>103</sub>	VP1 <sub>102</sub>	VP1 <sub>101</sub>	?							?
22A <sub>16</sub>	Vertical position register 1 <sub>11</sub> (VP1 <sub>11</sub> )	VP1 <sub>118</sub>	VP1 <sub>117</sub>	VP1 <sub>116</sub>	VP1 <sub>115</sub>	VP1 <sub>114</sub>	VP1 <sub>113</sub>	VP1 <sub>112</sub>	VP1 <sub>111</sub>	?							?
22B <sub>16</sub>	Vertical position register 1 <sub>12</sub> (VP1 <sub>12</sub> )	VP1 <sub>128</sub>	VP1 <sub>127</sub>	VP1 <sub>126</sub>	VP1 <sub>125</sub>	VP1 <sub>124</sub>	VP1 <sub>123</sub>	VP1 <sub>122</sub>	VP1 <sub>121</sub>	?							?
22C <sub>16</sub>																	?
22D <sub>16</sub>																	?
22E <sub>16</sub>																	?
22F <sub>16</sub>																	?
230 <sub>16</sub>	Vertical position register 2 <sub>1</sub> (VP2 <sub>1</sub> )							VP2 <sub>12</sub>	VP2 <sub>11</sub>								?
231 <sub>16</sub>	Vertical position register 2 <sub>2</sub> (VP2 <sub>2</sub> )							VP2 <sub>22</sub>	VP2 <sub>21</sub>								?
232 <sub>16</sub>	Vertical position register 2 <sub>3</sub> (VP2 <sub>3</sub> )							VP2 <sub>32</sub>	VP2 <sub>31</sub>								?
233 <sub>16</sub>	Vertical position register 2 <sub>4</sub> (VP2 <sub>4</sub> )							VP2 <sub>42</sub>	VP2 <sub>41</sub>								?
234 <sub>16</sub>	Vertical position register 2 <sub>5</sub> (VP2 <sub>5</sub> )							VP2 <sub>52</sub>	VP2 <sub>51</sub>								?
235 <sub>16</sub>	Vertical position register 2 <sub>6</sub> (VP2 <sub>6</sub> )							VP2 <sub>62</sub>	VP2 <sub>61</sub>								?
236 <sub>16</sub>	Vertical position register 2 <sub>7</sub> (VP2 <sub>7</sub> )							VP2 <sub>72</sub>	VP2 <sub>71</sub>								?
237 <sub>16</sub>	Vertical position register 2 <sub>8</sub> (VP2 <sub>8</sub> )							VP2 <sub>82</sub>	VP2 <sub>81</sub>								?
238 <sub>16</sub>	Vertical position register 2 <sub>9</sub> (VP2 <sub>9</sub> )							VP2 <sub>92</sub>	VP2 <sub>91</sub>								?
239 <sub>16</sub>	Vertical position register 2 <sub>10</sub> (VP2 <sub>10</sub> )							VP2 <sub>102</sub>	VP2 <sub>101</sub>								?
23A <sub>16</sub>	Vertical position register 2 <sub>11</sub> (VP2 <sub>11</sub> )							VP2 <sub>112</sub>	VP2 <sub>111</sub>								?
23B <sub>16</sub>	Vertical position register 2 <sub>12</sub> (VP2 <sub>12</sub> )							VP2 <sub>122</sub>	VP2 <sub>121</sub>								?
23C <sub>16</sub>																	?
23D <sub>16</sub>																	?
23E <sub>16</sub>																	?
23F <sub>16</sub>																	?
240 <sub>16</sub>	DA-H register (DA-H)																?
241 <sub>16</sub>	DA-L register (DA-L)									0	0	?	?	?	?	?	?
242 <sub>16</sub>	ROM correction address 1 (high-order)																0016
243 <sub>16</sub>	ROM correction address 1 (low-order)																0016
244 <sub>16</sub>	ROM correction address 2 (high-order)																0016
245 <sub>16</sub>	ROM correction address 2 (low-order)																0016
246 <sub>16</sub>	ROM correction enable register (RCR)					0	0	RCR1	RCR0								0016
247 <sub>16</sub>																	0016
248 <sub>16</sub>								0	0	0							0016

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

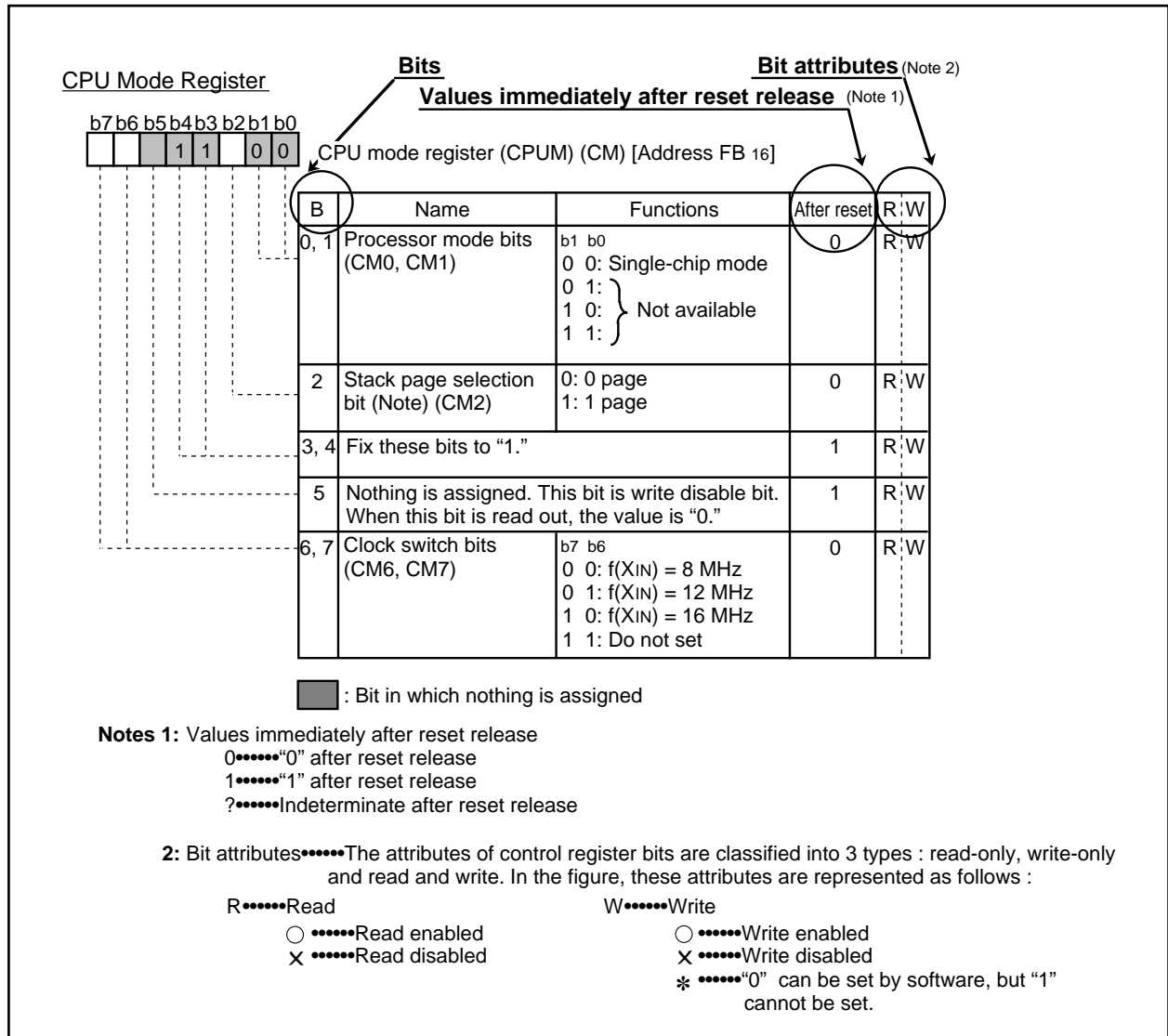
**Internal State of Processor Status Register and  
 Program Counter at Reset**



**Structure of Register**

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

**Note :** The following registers are the mask version's registers.



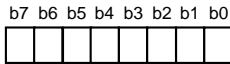


**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**Port Pi Direction Register**



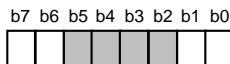
Port Pi direction register (Di) (i=0,1,2) [Addresses 00C1<sub>16</sub>, 00C3<sub>16</sub>, 00C5<sub>16</sub>]

B	Name	Functions	After reset	R	W
0	Port Pi direction register	0 : Port Pi <sub>0</sub> input mode 1 : Port Pi <sub>0</sub> output mode	0	R	W
1		0 : Port Pi <sub>1</sub> input mode 1 : Port Pi <sub>1</sub> output mode	0	R	W
2		0 : Port Pi <sub>2</sub> input mode 1 : Port Pi <sub>2</sub> output mode	0	R	W
3		0 : Port Pi <sub>3</sub> input mode 1 : Port Pi <sub>3</sub> output mode	0	R	W
4		0 : Port Pi <sub>4</sub> input mode 1 : Port Pi <sub>4</sub> output mode	0	R	W
5		0 : Port Pi <sub>5</sub> input mode 1 : Port Pi <sub>5</sub> output mode	0	R	W
6		0 : Port Pi <sub>6</sub> input mode 1 : Port Pi <sub>6</sub> output mode	0	R	W
7		0 : Port Pi <sub>7</sub> input mode 1 : Port Pi <sub>7</sub> output mode	0	R	W

Port Pi Direction Register

**Addresses 00C1<sub>16</sub>, 00C3<sub>16</sub>, 00C5<sub>16</sub>**

**Port P3 Direction Register**



Port P3 direction register (D3) [Address 00C7<sub>16</sub>]

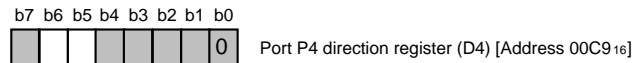
B	Name	Functions	After reset	R	W
0	Port P3 direction register	0 : Port P3 <sub>0</sub> input mode 1 : Port P3 <sub>0</sub> output mode	0	R	W
1		0 : Port P3 <sub>1</sub> input mode 1 : Port P3 <sub>1</sub> output mode	0	R	W
2 to 5	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—
6	Timer 3 count source selection bit (T3SC)	Refer to Timer mode register 2 (address 00F5 <sub>16</sub> ).	0	R	W
7	Ports P6 <sub>3</sub> , P6 <sub>4</sub> selection bit	Refer to clock source control register (address 0216 <sub>16</sub> ).	0	R	W

Port P3 Direction Register

**Address 00C7<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**Port P4 Direction Register**

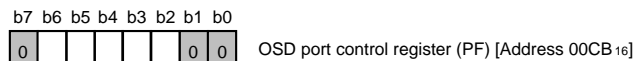


B	Name	Functions	After reset	R	W
0	Fix this bit to "0."		0	R	W
1 to 4, 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R	—
5	Port P4 <sub>5</sub> selection bit	0: SOUT pin 1: Input port P4 <sub>5</sub>	0	R	W
6	Port P4 <sub>6</sub> selection bit	0: SCLK pin 1: Input port P4 <sub>6</sub>	0	R	W

Port P4 Direction Register

**Address 00C9<sub>16</sub>**

**OSD Port Control Register**



B	Name	Functions	After reset	R	W
0, 1, 7	Fix these bits to "0."		0	R	W
2	Port P5 <sub>2</sub> output signal selection bit (R)	0 : R signal output 1 : Port P5 <sub>2</sub> output	0	R	W
3	Port P5 <sub>3</sub> output signal selection bit (G)	0 : G signal output 1 : Port P5 <sub>3</sub> output	0	R	W
4	Port P5 <sub>4</sub> output signal selection bit (B)	0 : B signal output 1 : Port P5 <sub>4</sub> output	0	R	W
5	Port P5 <sub>5</sub> output signal selection bit (OUT1)	0 : OUT1 signal output 1 : Port P5 <sub>5</sub> output	0	R	W
6	Port P1 <sub>0</sub> output signal selection bit (OUT2)	0 : Port P1 <sub>0</sub> output 1 : OUT2 signal output	0	R	W

OSD Port Control Register

**Address 00CB<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**OSD Control Register**

b7 b6 b5 b4 b3 b2 b1 b0

OSD control register (OC) [Address 00CE 16]

B	Name	Functions	After reset	R/W
0	OSD control bit (OC0) (See note 1)	0 : All-blocks display off 1 : All-blocks display on	0	R/W
1	Scan mode selection bit (OC1)	0 : Normal scan mode 1 : Bi-scan mode	0	R/W
2	Border type selection bit (OC2)	0 : All bordered 1 : Shadow bordered (See note 2)	0	R/W
3	Flash mode selection bit (OC3)	0 : Color signal of character background part does not flash 1 : Color signal of character background part flashes	0	R/W
4	Automatic solid space control bit (OC4)	0 : OFF 1 : ON	0	R/W
5	Window control bit (OC5)	0 : OFF 1 : ON	0	R/W
6, 7	Layer mixing control bits (OC6, OC7) (See note 3)	b7 b6 0 0: Logic sum (OR) of layer 1's color and layer 2's color 0 1: Layer 1's color has priority 1 0: Layer 2's color has priority 1 1: Do not set.	0	R/W

**Notes 1 :** Even this bit is switched during display, the display screen remains unchanged until a rising (falling) of the next V SYNC.  
**2 :** Shadow border is output at right and bottom side of the font.  
**3 :** Set "00" during displaying extra fonts.

OSD Control Register

**Address 00CE<sub>16</sub>**

**Horizontal Position Register**

b7 b6 b5 b4 b3 b2 b1 b0

Horizontal position register (HP) [Address 00CF 16]

B	Name	Functions	After reset	R/W
0 to 7	Control bits of horizontal display start positions (HP0 to HP7)	Horizontal display start positions 4Tosc X (setting value of high-order 4 bits X 16 <sup>1</sup> + setting value of low-order 4 bits X 16 <sup>0</sup> )	0	R/W

**Note:** The setting value synchronizes with the V SYNC.

Horizontal Position Register

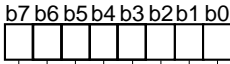
**Address 00CF<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**Block Control register i**



Block control register i (BCi) (i=1 to 12) [Addresses 00D0<sub>16</sub> to 00BF<sub>16</sub>](See note 1)

B	Name	Functions	After reset	R	W
0, 1	Display mode selection bits (BCi0, BCi1) (See note)	b1 b0 0 0: Display OFF 0 1: OSD mode 1 0: CC mode 1 1: EXOSD mode	Indeterminate	R	W
2	Border control bit (BCi2)	0: Border OFF 1: Border ON	Indeterminate	R	W
3, 4	Dot size selection bits (BCi3, BCi4)	b6 b5 b4 b3 CS6 Pre-divide ratio Dot size Display layer 0 0 1 1 — X 1 1Tc X 1/2H 1Tc X 1H 2Tc X 2H 3Tc X 3H 0 1 1 0 — X 2 1Tc X 1/2H 1Tc X 1H 2Tc X 2H 3Tc X 3H Layer1	Indeterminate	R	W
5, 6	Pre-divide ratio • layer selection bit (BCi5, BCi6)	0 0 0 0 — X 3 1Tc X 1/2H 0 1 1 1 — X 3 1Tc X 1H 1 0 1 1 — X 3 2Tc X 2H 3Tc X 3H 1 1 — 0 0 X 1 1Tc X 1/2H 1Tc X 1H 1 1 0 0 1 X 2 1Tc X 1/2H 1Tc X 1H 1 0 1 1 1.5Tc X 1/2H 1.5Tc X 1H Layer2	Indeterminate	R	W
7	OUT2 output control bit (BCi7) (See note 2)	BC17: Window top boundary BC27: Window bottom boundary	Indeterminate	R	W

- Notes 1:** Note that EPROM version the block control registers at addresses 00D0<sub>16</sub> to 00DF<sub>16</sub> when programming.  
**2:** Bit 4 of the color code 1 controls OUT1 output when bit 7 is "0."  
 Bit 4 of the color code 1 controls OUT2 output when bit 7 is "1."

Block Control Register i

**Addresses 00D0<sub>16</sub> to 00DB<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER and ON-SCREEN DISPLAY CONTROLLER

**Caption Position Register**

b7 b6 b5 b4 b3 b2 b1 b0

1	0	0					
---	---	---	--	--	--	--	--

Caption Position Register (CPS) [Address 00E0<sub>16</sub>]

B	Name	Functions	After reset	R	W
0 to 4	Specification main data slice line (CP0 to CP4)		0	R	W
5, 6	Fix these bits to "0."		0	R	W
7	Fix this bit to "0."		0	R	W

Caption Position Register

**Address 00E0<sub>16</sub>**

**Start Bit Position Register**

b7 b6 b5 b4 b3 b2 b1 b0

--	--	--	--	--	--	--	--

Start bit position register (SP) [Address 00E1<sub>16</sub>]

B	Name	Functions	After reset	R	W
0 to 6	Start bit generating time (SP0 to SP6)	Time from a falling of the horizontal synchronous signal to occurrence of a start bit = 4 X set value ("00 <sub>16</sub> " to "7F <sub>16</sub> ") X reference clock period	0	R	W
7	DSC1 bit 7 control bit (SP7)	0 : Generation of 16 pulses 1 : Generation of 16 pulses and detection of clock run-in pulse (4 to 6 pulses)	0	R	W

Start Bit Position Register

**Address 00E1<sub>16</sub>**

**Window Register**

b7 b6 b5 b4 b3 b2 b1 b0

0	0						
---	---	--	--	--	--	--	--

Window register (WN) [Address 00E2<sub>16</sub>]

B	Name	Functions	After reset	R	W
0 to 5	Window start time (WN0 to WN5)	Time from a falling of the horizontal synchronous signal to start of the window = 4 X set value ("00 <sub>16</sub> " to "3F <sub>16</sub> ") X reference clock period	0	R	W
6, 7	Fix these bits to "0."		0	R	W

Window Register

**Address 00E2<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

### Sync Slice Register

b7 b6 b5 b4 b3 b2 b1 b0

	0	0	0	0	1	0	1
--	---	---	---	---	---	---	---

Sync slice register (SSL) [Address 00E3<sub>16</sub>]

B	Name	Functions	After reset	R : W
0, 2	Fix these bits to "1."		0	R : W
1, 3 to 6	Fix these bits to "0."		0	R : W
7	Vertical synchronous signal ( $V_{sep}$ ) generating method selection bit (SSL7)	0: Method 1 1: Method 2	0	R : W

Sync Slice Register

**Address 00E3<sub>16</sub>**

### Clock Run-in Register 1

b7 b6 b5 b4 b3 b2 b1 b0

0	1	0	1				
---	---	---	---	--	--	--	--

Clock run-in register 1 (CR1) [Address 00E6<sub>16</sub>]

B	Name	Functions	After reset	R : W
0 to 3	Clock run-in count value of main-data slice line (CR10 to CR13)		0	R : W
4, 6	Fix these bits to "1."		0	R : W
5, 7	Fix these bits to "0."		0	R : W

Clock Run-in Register 1

**Address 00E6<sub>16</sub>**

### Clock Run-in Register 2

b7 b6 b5 b4 b3 b2 b1 b0

1	0	0	1	1	1		1
---	---	---	---	---	---	--	---

Clock run-in register 2 (CR2) [Address 00E7<sub>16</sub>]

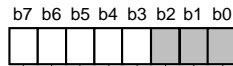
B	Name	Functions	After reset	R : W
0, 2 to 4, 7	Fix these bits to "1."		0	R : W
1	Start bit detecting method selection bit (CR21)	0: Method 1 1: Method 2	0	R : W
5, 6	Fix these bits to "0."		0	R : W

Clock Run-in Register 2

**Address 00E7<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**Clock Run-in Detect Register i**



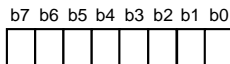
Clock run-in detect register i (CRDi) (i=1, 3) [Addresses 00E8<sub>16</sub>, 0208<sub>16</sub>]

B	Name	Functions	After reset	R	W
0 to 2	Test bits	Read-only	0	R	W
3 to 7	Clock run-in detection bits (CRDi3 to CRDi7)	Number of reference clock s to be counted one clock run-in pulse period	0	R	

Clock Run-in detect Register i

**Addresses 00E8<sub>16</sub>, 0208<sub>16</sub>**

**Clock Run-in Detect Register 2**



Clock run-in detect register 2 (CRD2) [Address 00E9<sub>16</sub>]

B	Name	Functions	After reset	R	W
0 to 2	Clock run-in pulses for sampling (CRD20 to CRD22)	b2 b1 b0 0 0 0 : Not available 0 0 1 : 1st pulse 0 1 0 : 2nd pulse 0 1 1 : 3rd pulse 1 0 0 : 4th pulse 1 0 1 : 5th pulse 1 1 0 : 6th pulse 1 1 1 : 7th pulse	0	R	W
3 to 7	Data clock generating time (CRD23 to CRD27)	Time from detection of a start bit to occurrence of a data clock = (13 + set value) X reference clock period	0	R	W

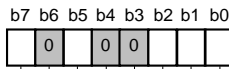
Clock Run-in detect Register 2

**Address 00E9<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

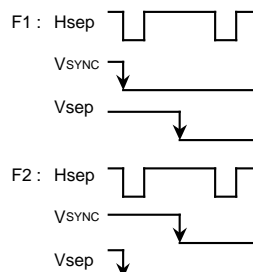
**Data Slicer Control Register 1**



Data slicer control register 1(DSC1) [Address 00EA<sub>16</sub>]

B	Bit	Functions	After reset	R	W																				
0	Data slicer control bit (DSC10)	0: Data slicer stopped 1: Data slicer operating	0	R	W																				
1, 2	Field to be sliced data selection bit (DSC11, DSC12)	<table border="1"> <thead> <tr> <th>b2</th> <th>b1</th> <th>Field of main data slice line</th> <th>Field for setting reference voltage</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>F2</td> <td>F2</td> </tr> <tr> <td>0</td> <td>1</td> <td>F1</td> <td>F1</td> </tr> <tr> <td>1</td> <td>0</td> <td>F1 and F2</td> <td>F2</td> </tr> <tr> <td>1</td> <td>1</td> <td>F1 and F2</td> <td>F1</td> </tr> </tbody> </table>	b2	b1	Field of main data slice line	Field for setting reference voltage	0	0	F2	F2	0	1	F1	F1	1	0	F1 and F2	F2	1	1	F1 and F2	F1	0	R	W
b2	b1	Field of main data slice line	Field for setting reference voltage																						
0	0	F2	F2																						
0	1	F1	F1																						
1	0	F1 and F2	F2																						
1	1	F1 and F2	F1																						
3, 4, 6	Fix these bits to "0."		0	R	W																				
5	Field determination flag (DSC15)	0: Hsep 1: Hsep	Indeterminate	R	—																				
7	Data latch completion flag for caption data in main data slice line (DSC17)	0: Data is not yet latched 1: Data is latched	Indeterminate	R	W																				

Definition of fields 1 (F1) and 2 (F2)



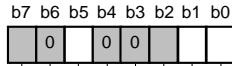
Data Slicer Control Register 1

**Address 00EA<sub>16</sub>**



**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**Data slicer Control Register 2**



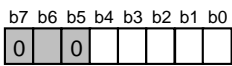
Data slicer Control register 2 (DSC2) [Address 00EB<sub>16</sub>]

B	Name	Functions	After reset	R	W
0	Timing signal generating circuit control bit (DSC20)	0: Stopped 1: Operating	0	R	W
1	Reference clock source selection bit (DSC21)	0: Video signal 1: Hsync signal	0	R	W
2, 7	Test bit	Read-only	Indeterminate	R	—
3, 4, 6	Fix these bits to "0."		0	R	W
5	V-pulse shape determination flag (DSC25)	0: Match 1: Mis match	Indeterminate	R	—

Data Slicer Control Register 2

**Address 00EB<sub>16</sub>**

**A-D Control Register**



A-D control register (ADCON) [Address 00EF<sub>16</sub>]

B	Name	Functions	After reset	R	W
0 to 2	Analog input pin selection bits (ADIN0 to ADIN2)	b2 b1 b0 0 0 0 : AD1 0 0 1 : AD2 0 1 0 : AD3 0 1 1 : AD4 1 0 0 : AD5 1 0 1 : AD6 1 1 0 : } Do not set. 1 1 1 : }	0	R	W
3	A-D conversion completion bit (ADSTR)	0: Conversion in progress 1: Conversion completed	Indeterminate	R	W
4	Vcc connection selection bit (ADVREF)	0: OFF 1: ON	Indeterminate	R	W
6	Nothing is assigned. This bit is a write disable bit. When this bit is read out, the value is indeterminate.		Indeterminate	R	—
5, 7	Fix these bits to "0."		0	R	—

A-D Control Register

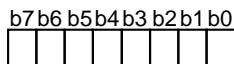
**Address 00EF<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**MITSUBISHI MICROCOMPUTERS**  
**M37274MA-XXXSP**

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

Timer Mode Register 1



Timer mode register 1 (TM1) [Address 00F4<sub>16</sub>]

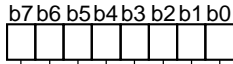
B	Name	Functions	After reset	R	W
0	Timer 1 count source selection bit 1 (TM10)	0: f(X <sub>IN</sub> )/16 or f(X <sub>CIN</sub> )/16 (Note) 1: Count source selected by bit 5 of TM1	0	R	W
1	Timer 2 count source selection bit 1 (TM11)	0: Count source selected by bit 4 of TM1 1: External clock from TIM2 pin	0	R	W
2	Timer 1 count stop bit (TM12)	0: Count start 1: Count stop	0	R	W
3	Timer 2 count stop bit (TM13)	0: Count start 1: Count stop	0	R	W
4	Timer 2 count source selection bit 2 (TM14)	0: f(X <sub>IN</sub> )/16 or f(X <sub>CIN</sub> )/16 (See note) 1: Timer 1 overflow	0	R	W
5	Timer 1 count source selection bit 2 (TM15)	0: f(X <sub>IN</sub> )/4096 or f(X <sub>CIN</sub> )/4096 (See note) 1: External clock from TIM2 pin	0	R	W
6	Timer 5 count source selection bit 2 (TM16)	0: Timer 2 overflow 1: Timer 4 overflow	0	R	W
7	Timer 6 internal count source selection bit (TM17)	0: f(X <sub>IN</sub> )/16 or f(X <sub>CIN</sub> )/16 (See note) 1: Timer 5 overflow	0	R	W

Timer Mode Register 1

**Address 00F4<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**Timer Mode Register 2**



Timer mode register 2 (TM2) [Address 00F5<sub>16</sub>]

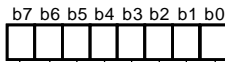
B	Name	Functions	After reset	R	W
0	Timer 3 count source selection bit (TM20)	(b6 at address 00C7 <sub>16</sub> ) b0 0 0 : f(X <sub>IN</sub> )/16 or f(X <sub>CIN</sub> )/16 (See note) 0 1 : f(X <sub>CIN</sub> ) 1 0 : } 1 1 : } External clock from TIM3 pin	0	R	W
1, 4	Timer 4 count source selection bits (TM21, TM24)	b4 b1 0 0 : Timer 3 overflow signal 0 1 : f(X <sub>IN</sub> )/16 or f(X <sub>CIN</sub> )/16 (See note) 1 0 : f(X <sub>IN</sub> )/2 or f(X <sub>CIN</sub> )/2 (See note) 1 1 : f(X <sub>CIN</sub> )	0	R	W
2	Timer 3 count stop bit (TM22)	0: Count start 1: Count stop	0	R	W
3	Timer 4 count stop bit (TM23)	0: Count start 1: Count stop	0	R	W
5	Timer 5 count stop bit (TM25)	0: Count start 1: Count stop	0	R	W
6	Timer 6 count stop bit (TM26)	0: Count start 1: Count stop	0	R	W
7	Timer 5 count source selection bit 1 (TM27)	0: f(X <sub>IN</sub> )/16 or f(X <sub>CIN</sub> )/16 (See note) 1: Count source selected by bit 6 of TM1	0	R	W

**Note:** Either f(X<sub>IN</sub>) or f(X<sub>CIN</sub>) is selected by bit 7 of the CPU mode register.

Timer Mode Register 2

**Address 00F5<sub>16</sub>**

**I<sup>2</sup>C Data Shift Register**



I<sup>2</sup>C data shift register1(S0) [Address 00F6<sub>16</sub>]

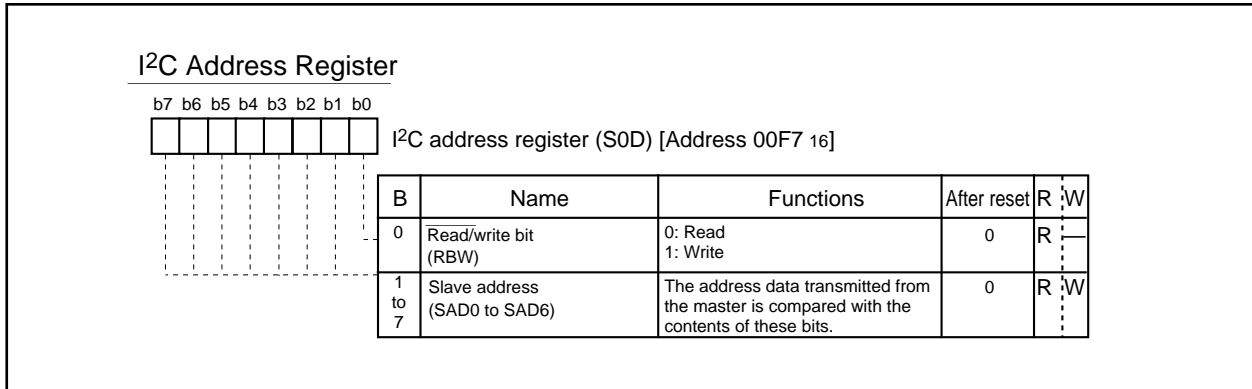
B	Name	Functions	After reset	R	W
0 to 7	D0 to D7	This is an 8-bit shift register to store receive data and write transmit data.	Indeterminate	R	W

**Note:** To write data into the I<sup>2</sup>C data shift register after setting the MST bit to "0" (slave mode), keep an interval of 8 machine cycles or more.

I<sup>2</sup>C Data Shift Register

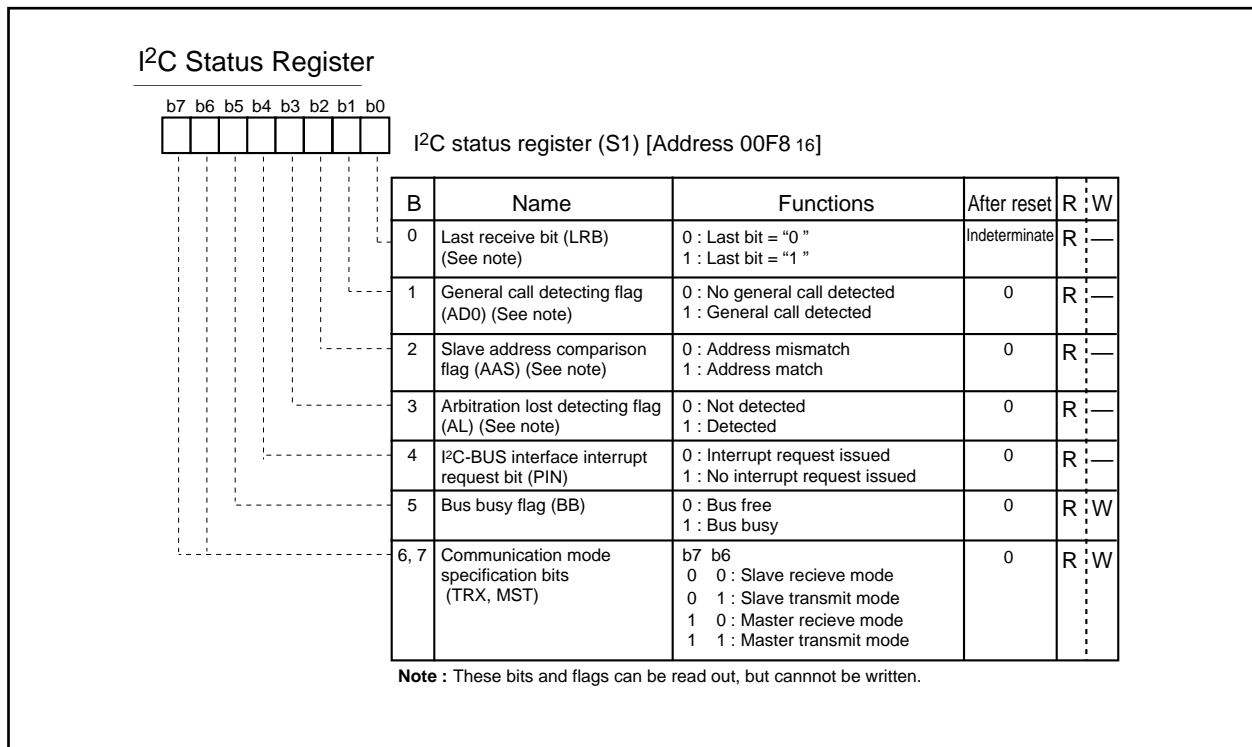
**Address 00F6<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.



I<sup>2</sup>C Address Register

**Address 00F7<sub>16</sub>**



I<sup>2</sup>C Status Register

**Address 00F8<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

**I<sup>2</sup>C Control Register**

b7 b6 b5 b4 b3 b2 b1 b0  
 I<sup>2</sup>C control register (S1D : address 00F9<sub>16</sub>)

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	I <sup>2</sup> C-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 I <sup>2</sup> C-BUS interface and ports	b7 b6 Connection port (See note) 0 0 : None 0 1 : SCL1, SDA1 1 0 : SCL2, SDA2 1 1 : SCL1, SDA1 SCL2, SDA2	0	R : W

**Note:** When using ports P1<sub>1</sub>-P1<sub>4</sub> as I<sup>2</sup>C-BUS interface, the output structure changes automatically from CMOS output to N-channel open-drain output.

I<sup>2</sup>C Control Register

**Address 00F9<sub>16</sub>**

**I<sup>2</sup>C Clock Control Register**

b7 b6 b5 b4 b3 b2 b1 b0  
 I<sup>2</sup>C clock control register (S2 : address 00FA<sub>16</sub>)

B	Name	Functions	After reset	R : W																														
0 to 4	SCL frequency control bits (CCR0 to CCR4)	<table border="1"> <thead> <tr> <th>Setup value of CCR4-CCR0</th> <th>Standard clock mode</th> <th>High speed clock mode</th> </tr> </thead> <tbody> <tr> <td>00 to 02</td> <td>Setup disabled</td> <td>Setup disabled</td> </tr> <tr> <td>03</td> <td>Setup disabled</td> <td>333</td> </tr> <tr> <td>04</td> <td>Setup disabled</td> <td>250</td> </tr> <tr> <td>05</td> <td>100</td> <td>400 (See note)</td> </tr> <tr> <td>06</td> <td>83.3</td> <td>166</td> </tr> <tr> <td>:</td> <td>500/CCR value</td> <td>1000/CCR value</td> </tr> <tr> <td>1D</td> <td>17.2</td> <td>34.5</td> </tr> <tr> <td>1E</td> <td>16.6</td> <td>33.3</td> </tr> <tr> <td>1F</td> <td>16.1</td> <td>32.3</td> </tr> </tbody> </table> (at $\phi = 4$ MHz, unit : kHz)	Setup value of CCR4-CCR0	Standard clock mode	High speed clock mode	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	0	R : W
Setup value of CCR4-CCR0	Standard clock mode	High speed clock mode																																
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																																
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

I<sup>2</sup>C Clock Control Register

**Address 00FA<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**CPU Mode Register**

b7 b6 b5 b4 b3 b2 b1 b0

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

B	Name	Functions	After reset	R/W
0, 1	Processor mode bits (CM0, CM1)	b1 b0 0 0: Single-chip mode 0 1: 1 0: 1 1: } Not available	0	R/W
2	Stack page selection bit (CM2) (See note)	0: 0 page 1: 1 page	1	R/W
3, 4	Fix these bits to "1."		1	R/W
5	XCOUNT drivability selection bit (CM5)	0: LOW drive 1: HIGH drive	1	R/W
6	Main Clock (XIN-XOUT) stop bit (CM6)	0: Oscillating 1: Stopped	0	R/W
7	Internal system clock selection bit (CM7)	0: XIN-XOUT selected (high-speed mode) 1: XCIN-XCOUT selected (high-speed mode)	0	R/W

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

CPU Mode Register

**Address 00FB16**

**Interrupt Request Register 1**

b7 b6 b5 b4 b3 b2 b1 b0

Interrupt request register 1 (IREQ1) [Address 00FC 16]

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	OSD 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	A-D conversion • INT3 interrupt request bit (ADR)	0: No interrupt request issued 1: Interrupt request issued	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 —

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

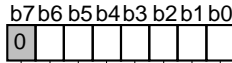
Interrupt Request Register 1

**Address 00FC16**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

Interrupt Request Register 2



Interrupt request register 2 (IREQ2) [Address 00FD<sub>16</sub>]

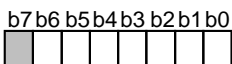
B	Name	Functions	After reset	R : W
0	INT1 interrupt request bit (INT1R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R : *
1	Data slicer interrupt request bit (DSR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R : *
2	Serial I/O interrupt request bit (SIOR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R : *
3	f(X <sub>IN</sub> )/4096 interrupt request bit (1MSR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R : *
4	INT2 interrupt request bit (INT2R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R : *
5	Multi-master I <sup>2</sup> C-BUS interrupt request bit (IICR)	0 : No interrupt request issued 1 : Interrupt request issued	0	R : *
6	Timer 5 • 6 interrupt request bit (T56R)	0 : No interrupt request issued 1 : Interrupt request issued	0	R : *
7	Fix this bit to "0."		0	R : W

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

Interrupt Request Register 2

**Address 00FD<sub>16</sub>**

Interrupt Control Register 1



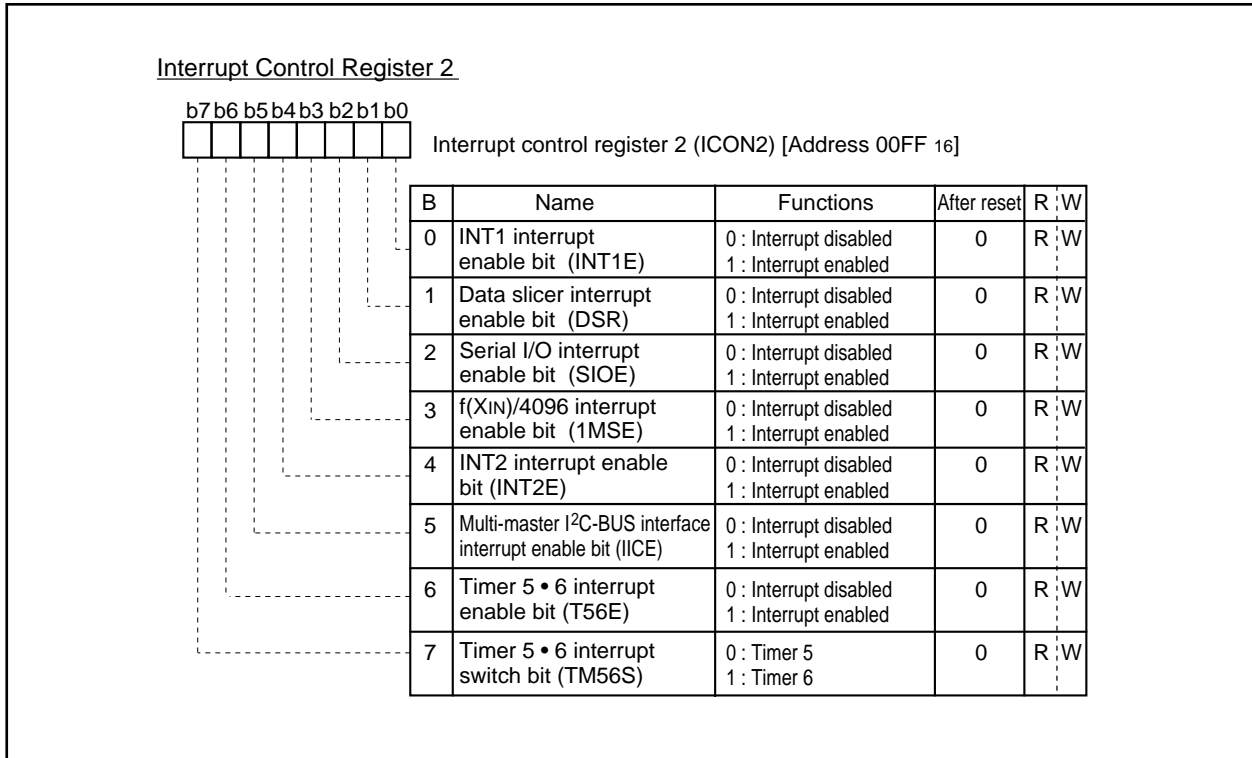
Interrupt control register 1 (ICON1) [Address 00FE<sub>16</sub>]

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	OSD interrupt enable bit (CRTE)	0 : Interrupt disabled 1 : Interrupt enabled	0	R : W
5	VSYNC interrupt enable bit (VSCR)	0 : Interrupt disabled 1 : Interrupt enabled	0	R : W
6	A-D conversion • INT3 interrupt enable bit (ADE)	0 : Interrupt disabled 1 : Interrupt enabled	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 : —

Interrupt Control Register 1

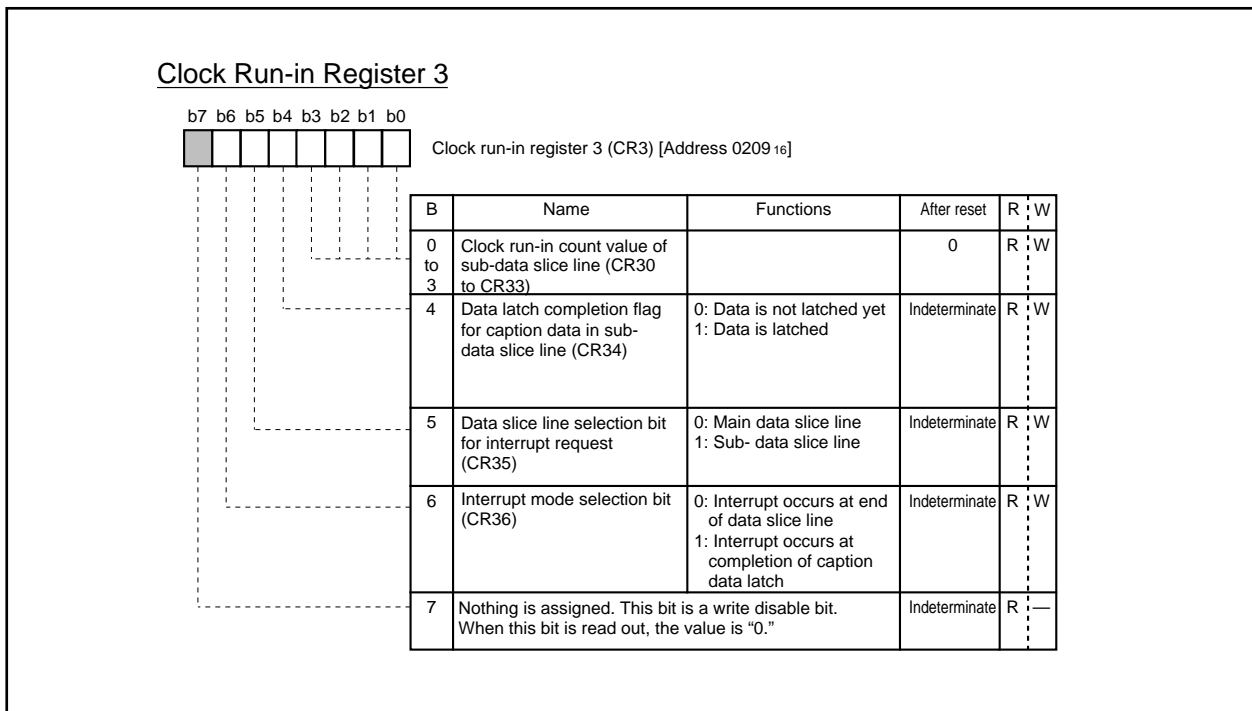
**Address 00FE<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.



Interrupt Control Register 2

**Address 00FF<sub>16</sub>**



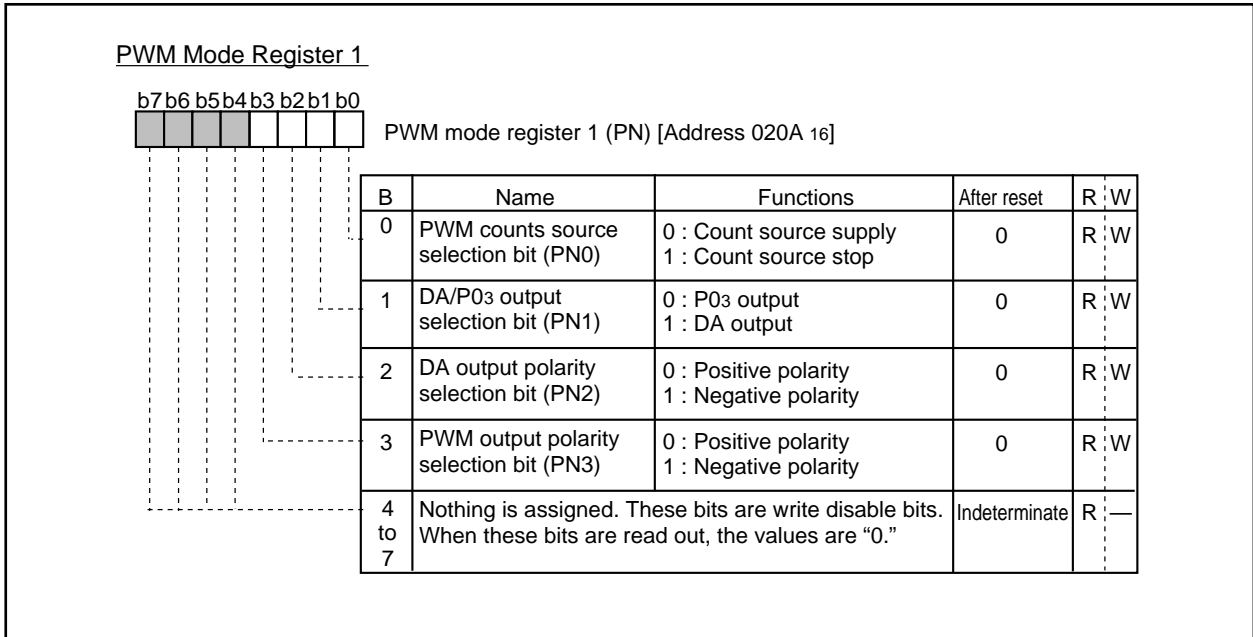
Clock Run-in Register 3

**Address 0209<sub>16</sub>**



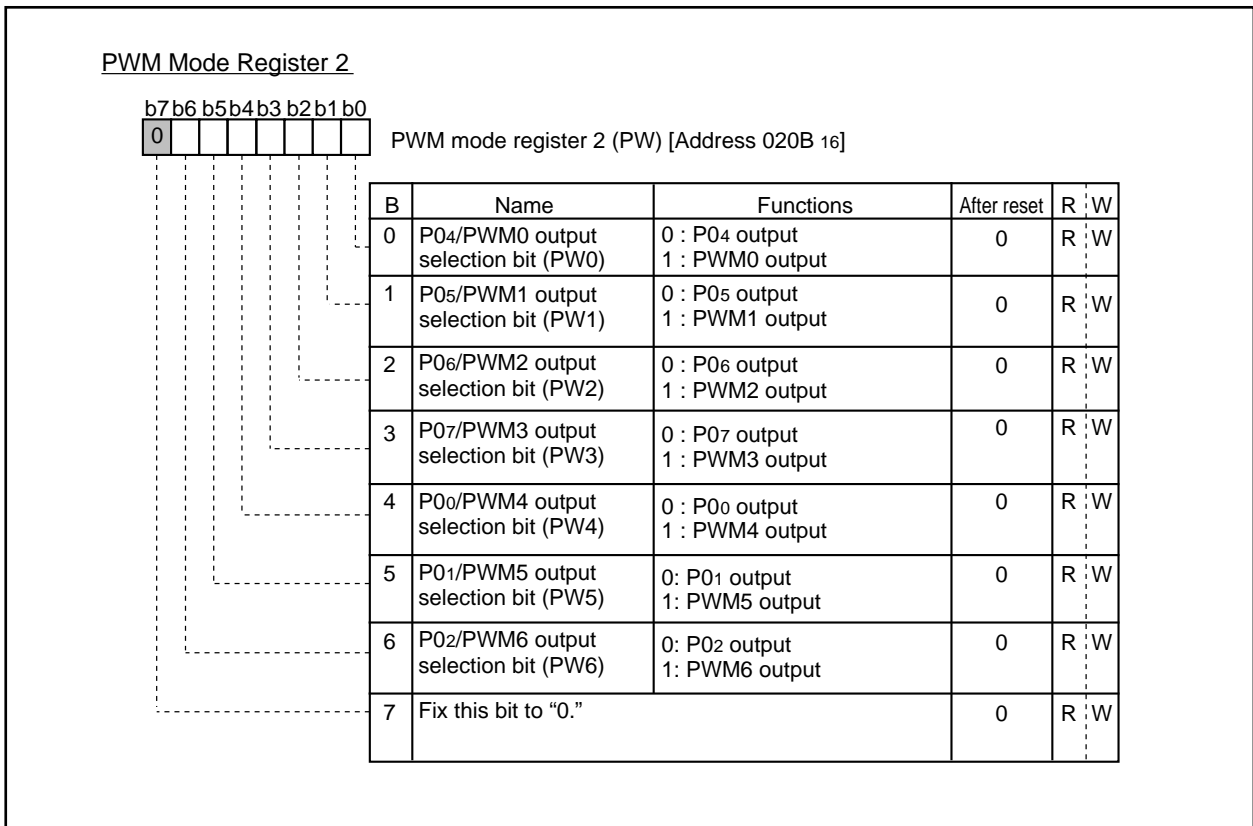
**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER



PWM Mode Register 1

**Address 020A<sub>16</sub>**

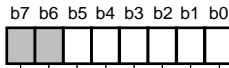


PWM Mode Register 2

**Address 020B<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**Sync Pulse Counter Register**



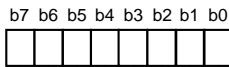
Sync pulse counter register (SYC) [Address 020F<sub>16</sub>]

B	Name	Functions	After reset	R	W
0 to 4	Count value (SYC0 to SYC4)		0	R	—
5	Count source (SYC5)	0: Hsync signal 1: Composite sync signal	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	—

Sync Pulse Counter Register

**Address 020F<sub>16</sub>**

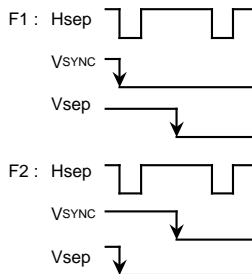
**Data Slicer Control Register 3**



Data slicer control register 3 (DSC3) [Address 0210<sub>16</sub>]

B	Bit	Functions	After reset	R	W		
0	Line selection bit for slice voltage (DSC30)	0: Main data slice line 1: Sub-data slice line	0	R	W		
1, 2	Field to be sliced data selection bit (DSC31, DSC32)	Field of sub-data slice line		0	R		
		Field for setting reference voltage					
		b2	b1			F2	F2
		b2	b1			F1	F1
3 to 7	Setting bit of sub-data slice line (DSC33 to DSC37)		0	R	W		

Definition of fields 1 (F1) and 2 (F2)



Data Slicer Control Register 3

**Address 0210<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER  
 and ON-SCREEN DISPLAY CONTROLLER

Interrupt Input Polarity Register

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

Interrupt input polarity register (IP) [Address 0212<sub>16</sub>]

B	Name	Functions	After reset	R	W
0 to 2, 5	Fix these bits to "0."		0	R	W
3	INT1 polarity switch bit (INT1POL)	0 : Positive polarity 1 : Negative polarity	0	R	W
4	INT2 polarity switch bit (INT2POL)	0 : Positive polarity 1 : Negative polarity	0	R	W
6	INT3 polarity switch bit (INT3POL)	0 : Positive polarity 1 : Negative polarity	0	R	W
7	A-D conversion • INT3 interrupt source selection bit (RE7)	0 : Positive polarity 1 : Negative polarity	0	R	W

Interrupt Input Polarity Register

Address 0212<sub>16</sub>

Serial I/O Mode Register

b7	b6	b5	b4	b3	b2	b1	b0
0	0						

Serial I/O mode register (SM) [Address 0213<sub>16</sub>]

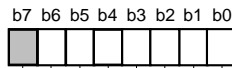
B	Name	Functions	After reset	R	W
0, 1	Internal synchronous clock selection bits (SM0, SM1)	b1 b0 0 0: f(X <sub>IN</sub> )/4 or f(X <sub>CIN</sub> )/4 0 1: f(X <sub>IN</sub> )/16 or f(X <sub>CIN</sub> )/16 1 0: f(X <sub>IN</sub> )/32 or f(X <sub>CIN</sub> )/32 1 1: f(X <sub>IN</sub> )/64 or f(X <sub>CIN</sub> )/64	0	R	W
2	Synchronous clock selection bit (SM2)	0: External clock 1: Internal clock	0	R	W
3	Port function selection bit (SM3)	0: P11, P13 1: SCL1, SDA1	0	R	W
4	Port function selection bit (SM4)	0: P12, P14 1: SCL2, SDA2	0	R	W
5	Transfer direction selection bit (SM5)	0: LSB first 1: MSB first	0	R	W
6, 7	Fix these bits to "0."		0	R	W

Serial I/O Mode Register

Address 0213<sub>16</sub>

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

Clock Source Control Register



Clock source control register (CS) [Address 0216<sub>16</sub>]

B	Name	Functions	After reset	R	W
0	CC mode clock selection bit (CS0)	0: Data slicer clock 1: OSC1 clock	0	R	W
1, 2	OSD mode clock selection bits (CS1, CS2)	b <sub>2</sub> b <sub>1</sub> 0 0: Data slicer clock 0 1: OSC1 clock 1 0: Main clock (See note 1) 1 1: Do not set	0	R	W
3	EXOSD mode clock selection bit (CS3)	0: Data slicer clock 1: OSC1 clock	0	R	W
4, 5	OSD oscillating mode selection bits (CS4, CS5)	b <sub>5</sub> b <sub>4</sub> 0 0: 32 kHz oscillating mode 0 1: Input ports P6 <sub>3</sub> , P6 <sub>4</sub> (See note 2) 1 0: LC oscillating mode 1 1: Ceramic • quartz-crystal oscillating mode	0	R	W
6	Pre-divide ratio of layer 2 selection bit (CS6)	0: X 1 1: X 2	0	R	W
7	Test bit (See note 3)		0	R	W

- Notes**
- 1: When setting "10<sub>2</sub>," main clock is set as a clock in the CC mode and EXOSD mode regardless of bits 0, 3.
  - 2: When selecting input ports P6<sub>3</sub> and P6<sub>4</sub>, set bit 7 at address 00C7<sub>16</sub> to "1."
  - 3: Be sure to set bit 7 to "0" for program of the mask and the EPROM versions. For the emulator MCU version (M37274ERSS), be sure to set bit 7 to "1" when using the data slicer clock for software debugging.

Clock Source Control Register

**Address 0216<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

SINGLE-CHIP 8-BIT CMOS MICROCOMPUTER with CLOSED CAPTION DECODER and ON-SCREEN DISPLAY CONTROLLER

### I/O Polarity Control Register

b7 b6 b5 b4 b3 b2 b1 b0

--	--	--	--	--	--	--	--

I/O polarity control register (PC) [Address 0217<sub>16</sub>]

B	Name	Functions	After reset	R : W
0	Hsync input polarity switch bit (PC0)	0 : Positive polarity input 1 : Negative polarity input	0	R : W
1	Vsync input polarity switch bit (PC1)	0 : Positive polarity input 1 : Negative polarity input	0	R : W
2	R, G, B output polarity switch bit (PC2)	0 : Positive polarity output 1 : Negative polarity output	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	OUT1 output polarity switch bit (PC4)	0 : Positive polarity output 1 : Negative polarity output	0	R : W
5	OUT2 output polarity switch bit (PC5)	0 : Positive polarity output 1 : Negative polarity output	0	R : W
6	Display dot line selection bit (PC6) (See note)	0 : " <input type="checkbox"/> " at even field " <input checked="" type="checkbox"/> " at odd field 1 : " <input checked="" type="checkbox"/> " at even field " <input type="checkbox"/> " at odd field	0	R : W
7	Field determination flag (PC7)	0 : Even field 1 : Odd field	1	R : —

**Note:** Refer to the corresponding figure (P62).

I/O Polarity Control Register

**Address 0217<sub>16</sub>**

### Raster Color Register

b7 b6 b5 b4 b3 b2 b1 b0

--	--	--	--	--	--	--	--

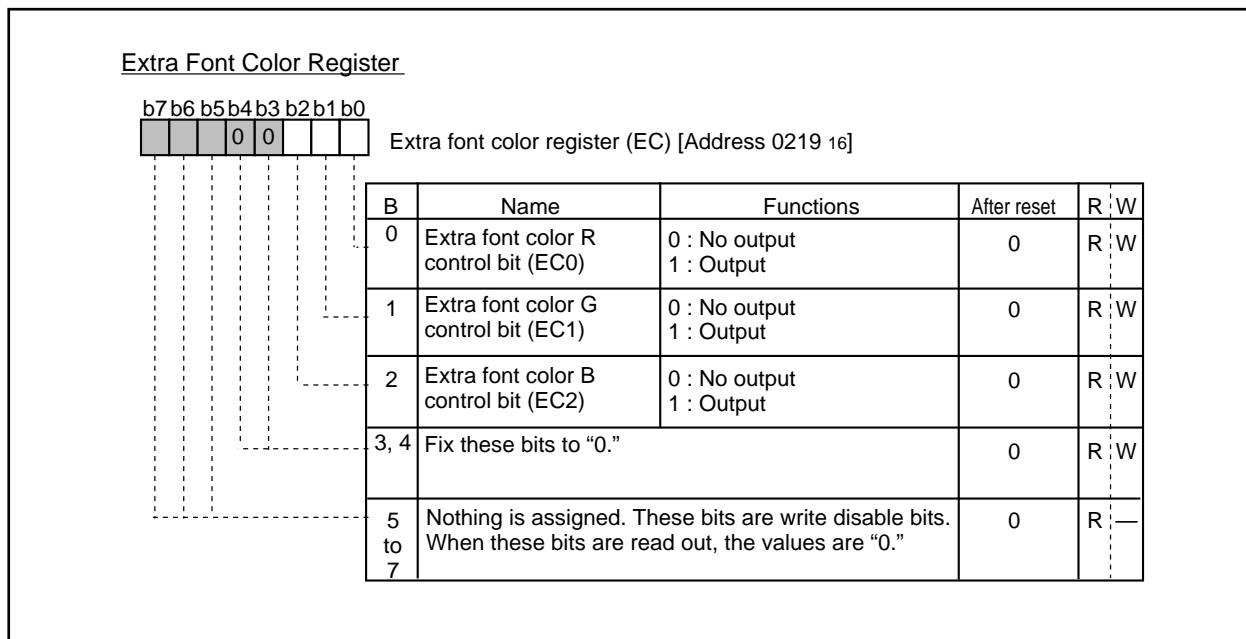
Raster color register (RC) [Address 0218<sub>16</sub>]

B	Name	Functions	After reset	R : W
0	Raster color R control bit (RC0)	0 : No output 1 : Output	0	R : W
1	Raster color G control bit (RC1)	0 : No output 1 : Output	0	R : W
2	Raster color B control bit (RC2)	0 : No output 1 : Output	0	R : W
3, 4	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are "0."		0	R : —
5	Raster color OUT1 control bit (RC5)	0 : No output 1 : Output	0	R : W
6	Raster color OUT2 control bit (RC6)	0 : No output 1 : Output	0	R : W
7	OSD interrupt source selection bit (RC7)	0 : Interrupt occurs at end of OSD or EXOSD block display 1 : Interrupt occurs at end of CC mode block display	0	R : W

Raster Color Register

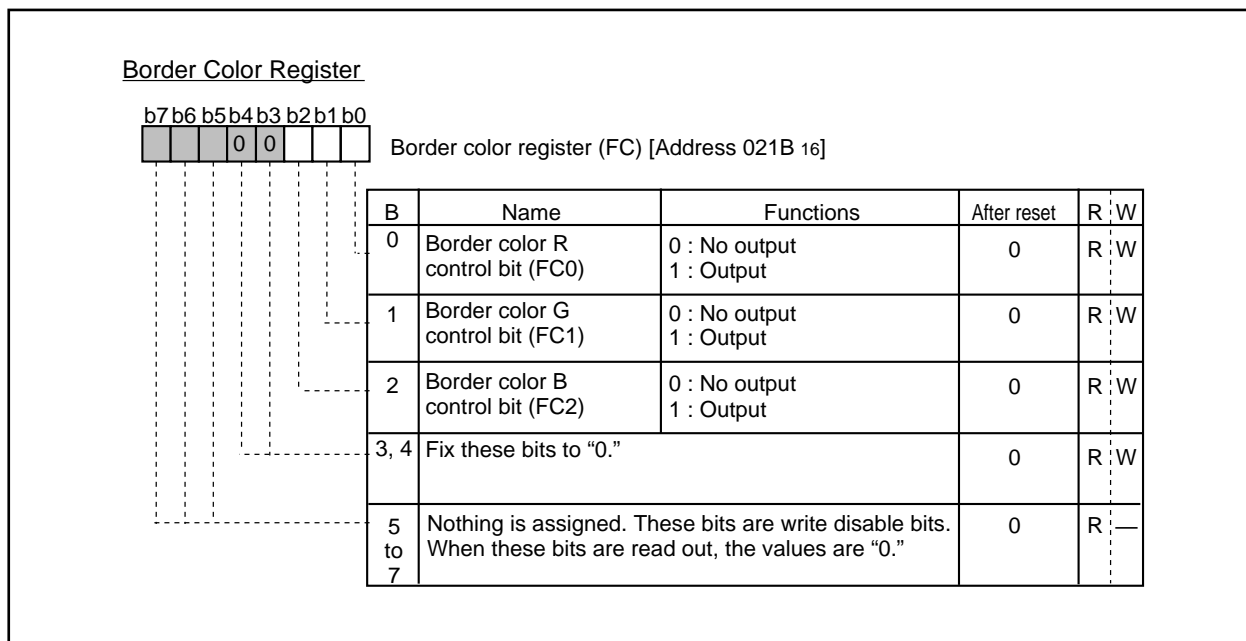
**Address 0218<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.



Extra Font Color Register

**Address 0219<sub>16</sub>**



Border Color Register

**Address 021B<sub>16</sub>**

**PRELIMINARY**  
 Notice: This is not a final specification.  
 Some parametric limits are subject to change.

**Window H Register 1**

Window H register 1 (WH1) [Address 021C 16]

B	Name	Functions	After reset	R;W
0 to 7	Control bits of window top boundary (WN10 to WN17) (See note 1)	Top boundary position (low-order 8 bits) $TH \times$ (setting value of low-order 2 bits of WH2 $\times 16^2$ + setting value of high-order 4 bits of WH1 $\times 16^1$ + setting value of low-order 4 bits of WH1 $\times 16^0$ )	Indeterminate	R;W

**Notes 1:** Set values except "00 16" to the WH1 when WH2 is "00 16."  
**2:** TH is cycle of HSYNC.

Window H Register 1

**Address 021C<sub>16</sub>**

**Window L Register 1**

Window L register 1 (WL1) [Address 021D 16]

B	Name	Functions	After reset	R;W
0 to 7	Control bits of window top boundary (WL10 to WL17) (See note 1)	Top boundary position (low-order 8 bits) $TH \times$ (setting value of low-order 2 bits of WL2 $\times 16^2$ + setting value of high-order 4 bits of WL1 $\times 16^1$ + setting value of low-order 4 bits of WL1 $\times 16^0$ )	Indeterminate	R;W

**Notes 1:** Set values fit for the following condition: (WH1+WH2)<(WL1+WL2)  
**2:** TH is cycle of HSYNC.

Window L Register 1

**Address 021D<sub>16</sub>**

**Window H Register 2**

Window H register 2 (WH2) [Address 021E 16]

B	Name	Functions	After reset	R;W
0, 1	Control bits of window top boundary (WN20 to WN27) (See note 1)	Top boundary position (high-order 2 bits) $TH \times$ (setting value of low-order 2 bits of WH2 $\times 16^2$ + setting value of high-order 4 bits of WH1 $\times 16^1$ + setting value of low-order 4 bits of WH1 $\times 16^0$ )	Indeterminate	R;W
2 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are indeterminate.		Indeterminate	R;—

**Notes 1:** Set values except "00 16" to the WH1 when WH2 is "00 16."  
**2:** TH is cycle of HSYNC.

Window H Register 2

**Address 021E<sub>16</sub>**

**Window L Register 2**

b7 b6 b5 b4 b3 b2 b1 b0

Window L register 2 (WL2) [Address 021F<sub>16</sub>]

B	Name	Functions	After reset	R	W
0, 1	Control bits of window top boundary (WL20 to WL27) (See note 1)	Top boundary position (high-order 2 bits) TH X (setting value of low-order 2 bits of WL2 × 16 <sup>2</sup> + setting value of high-order 4 bits of WL1 × 16 <sup>1</sup> + setting value of low-order 4 bits of WL1 × 16 <sup>0</sup> )	Indeterminate	R	W
2 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are indeterminate.		Indeterminate	R	—

**Notes 1:** Set values fit for the following condition: (WH1+WH2)<(WL1+WL2)  
**2:** TH is cycle of Hsync.

Window L Register 2

**Address 021F<sub>16</sub>**

**Vertical Position Register 1i**

b7 b6 b5 b4 b3 b2 b1 b0

Vertical position register 1i (VP1i) (i = 1 to 12) [Addresses 0220<sub>16</sub> to 022B<sub>16</sub>]

B	Name	Functions	After reset	R	W
0 to 7	Control bits of vertical display start positions (VP1i0 to VP1i7) (See note 1)	Vertical display start positions (low-order 8 bits) TH X (setting value of low-order 2 bits of VP2i × 16 <sup>2</sup> + setting value of low-order 4 bits of VP1i × 16 <sup>1</sup> + setting value of low-order 4 bits of VP1i × 16 <sup>0</sup> )	Indeterminate	R	W

**Notes 1:** Set values except "00<sub>16</sub>" "01<sub>16</sub>" to VP1i when VP2i is "00<sub>16</sub>."  
**2:** TH is cycle of Hsync.

Vertical Position Register 1i

**Addresses 0220<sub>16</sub> to 022B<sub>16</sub>**

**Vertical Position Register 2i**

b7 b6 b5 b4 b3 b2 b1 b0

Vertical position register 2i (VP2i) (i = 1 to 12) [Addresses 0230<sub>16</sub> to 023B<sub>16</sub>]

B	Name	Functions	After reset	R	W
0, 1	Control bits of vertical display start positions (VP1i0 to VP1i7) (See note 1)	Vertical display start positions (high-order 2 bits) TH X (setting value of low-order 2 bits of VP2i × 16 <sup>2</sup> + setting value of low-order 4 bits of VP1i × 16 <sup>1</sup> + setting value of low-order 4 bits of VP1i × 16 <sup>0</sup> )	Indeterminate	R	W
2 to 7	Nothing is assigned. These bits are write disable bits. When these bits are read out, the values are indeterminate.		Indeterminate	R	—

**Notes 1:** Set values except "00<sub>16</sub>" "01<sub>16</sub>" to VP1i when VP2i is "00<sub>16</sub>."  
**2:** TH is cycle of Hsync.

Vertical Position Register 2i

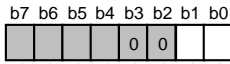
**Addresses 0230<sub>16</sub> to 023B<sub>16</sub>**



**PRELIMINARY**

Notice: This is not a final specification.  
Some parametric limits are subject to change.

**ROM Correction Enable Register**



ROM correction enable register (RCR) [Address 0246<sub>16</sub>]

B	Name	Functions	After reset	R : W
0	Block 1 enable bit (RC0)	0: Disabled 1: Enabled	0	R : W
1	Block 2 enable bit (RC1)	0: Disabled 1: Enabled	0	R : W
2, 3	Fix these 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 : —

ROM Correction Enable Register

**Address 0246<sub>16</sub>**

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.

