



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Table of contents

1. Description .....	1
1.1 Features .....	1
1.2 Applications .....	1
1.3 Pin Configuration .....	3
1.4 Block Diagram .....	4
1.5 Performance Outline .....	5
2. Operation of Functional Blocks .....	9
2.1 Memory .....	9
2.2 CPU .....	13
2.3 Reset .....	16
2.4 Processor Mode .....	20
2.5 Clock Generating Circuit .....	31
2.6 Protection .....	40
2.7 Interrupt .....	41
2.8 Watchdog Timer .....	61
2.9 DMAC .....	63
2.10 Timer .....	73
2.11 Serial I/O .....	91
2.12 A-D Converter .....	132
2.13 D-A Converter .....	142
2.14 CRC Calculation Circuit .....	144
2.15 Expansion Function .....	146
2.16 Programmable I/O .....	207
3. Usage Precaution .....	217
4. Electrical Characteristic .....	222
5. Marking Figure .....	239
6. Package Outline .....	240

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 1.3 Pin Configuration

Figure 1.3.1 shows the pin configuration (top view).

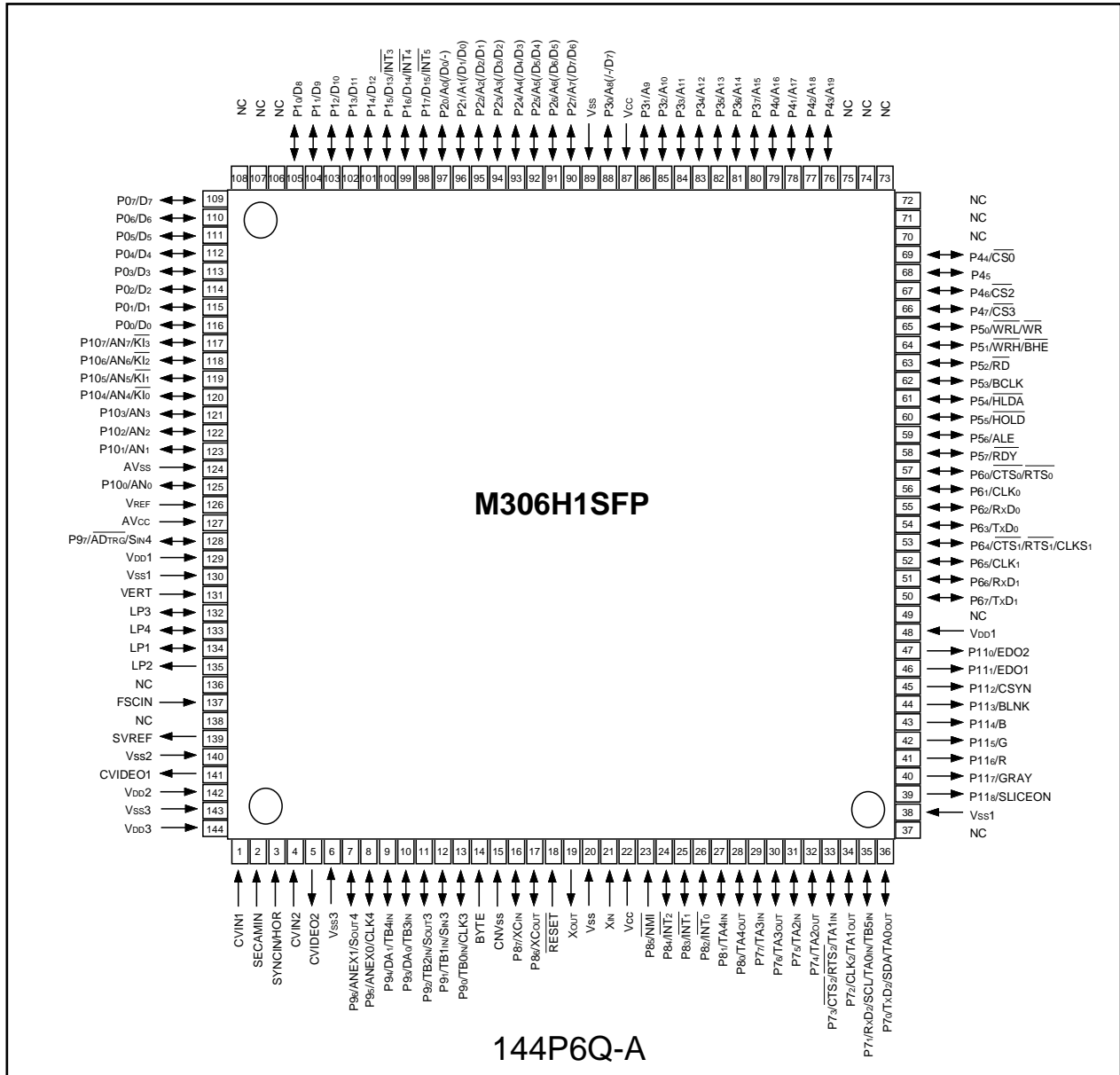


Figure 1.3.1 Pin configuration (top view)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 1.4 Block Diagram

Figure 1.4.1 is a block diagram of the M306H1SFP.

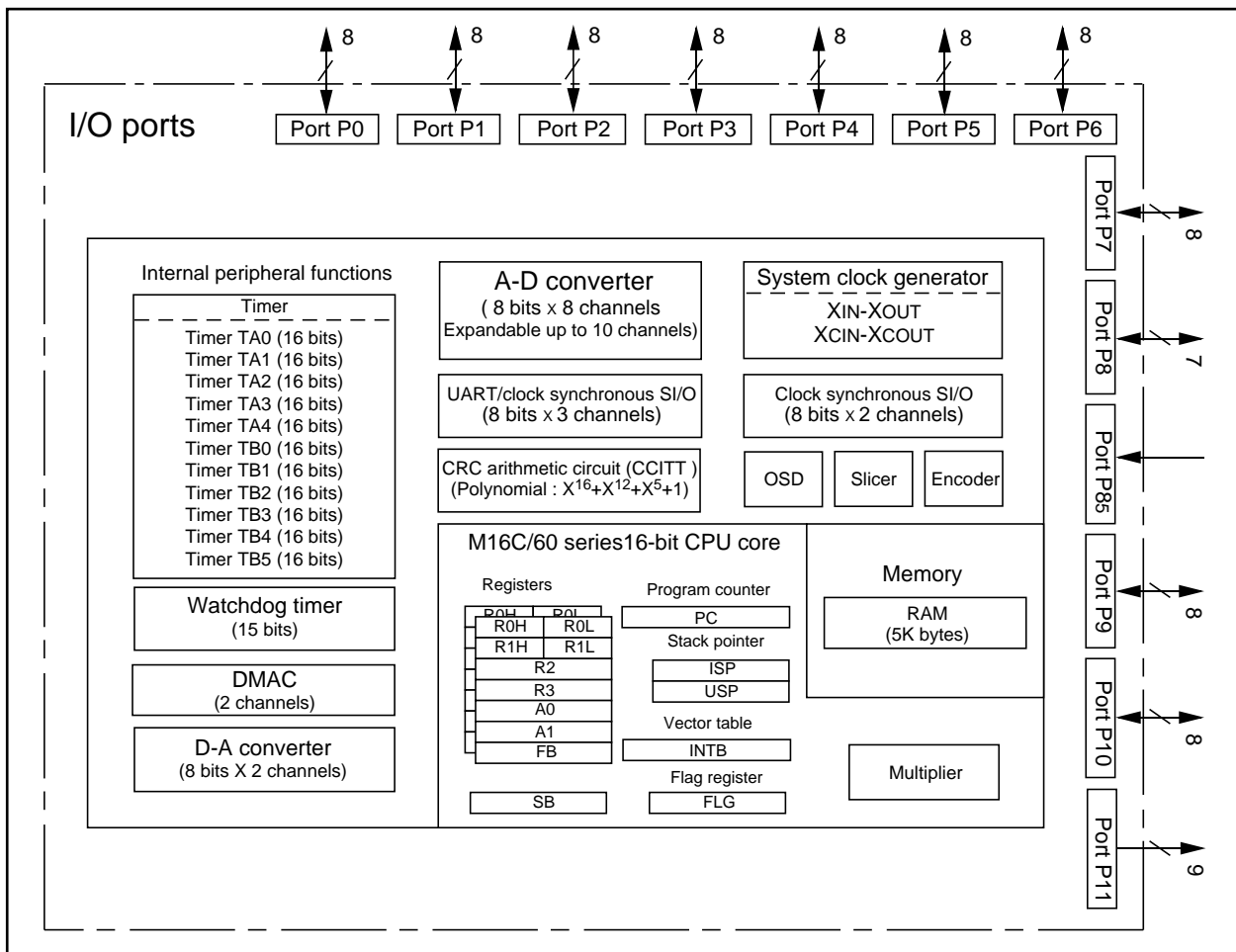


Figure 1.4.1 Block diagram of M306H1SFP

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 1.5 Performance Outline

Table 1.5.1 is a performance outline of M306H1SFP.

**Table 1.5.1 Performance outline of M306H1SFP**

Item		Performance
Number of basic instructions		91 instructions
Shortest instruction execution time		100ns (f(XIN)=10MHz)
Memory capacity	ROM	—
	RAM	5K bytes
I/O port	P0 to P10 (except P85)	8 bits x 10, 7 bits x 1
Input port	P85	1 bit x 1
Output port	P11	9 bit x 1
Multifunction timer	TA0, TA1, TA2, TA3, TA4	16 bits x 5
	TB0, TB1, TB2, TB3, TB4, TB5	16 bits x 6
Serial I/O	UART0, UART1, UART2	(UART or clock synchronous) x 3
	SI/O3, SI/O4	(Clock synchronous) x 2
A-D converter		8 bits x (8 + 2) channels
D-A converter		8 bits x 2 channels
DMAC		2 channels (trigger: 24 sources)
CRC calculation circuit		CRC-CCITT
Watchdog timer		15 bits x 1 (with prescaler)
Interrupt		25 internal and 8 external sources, 4 software sources, 7 levels
Clock generating circuit		2 built-in clock generation circuits (built-in feedback resistor, and external ceramic or crystal oscillator)
Supply voltage		4.75 to 5.25V (f(XIN)=10MHz)
Device configuration		CMOS high performance silicon gate
Package		144-pin plastic mold QFP
OSD function	OSD display RAM	2.75K Bytes (25 x 40 x 22-bit)
	Font RAM	3.84K Bytes (12 x 10 x 256-bit)
	SYRAM	260 Bytes (13 x 10 x 16-bit)
	Screen composition	40 characters x 25 lines
	Character composition	12 x 10 dots matrix
	Character coloring	8 colors choices per character
	Character Background coloring	8 colors choices per character
	Background coloring	8 colors choices per screen
	SYRAM color	8 colors choices per character
	Character Background coloring	8 colors choices per character
	Synchronous signal	PAL
	Video signal	PAL
Data slicer	Slice RAM	864 Bytes (48 x 18 x 8-bit)
	VBIRAM	95 Bytes ((5 + 5 x 18) x 8-bit)
	Data slicer	for PDC, VPS and VBI
	Encoder	for VBI

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Table 1.5.2 Pin Description**

Pin name	Signal name	I/O type	Function
Vcc, Vss	Power supply input		Supply 4.75 to 5.25 V to the Vcc pin. Supply 0 V to the Vss pin.
CNVss	CNVss	Input	This pin switches between processor modes. Connect it to the Vcc pin when in microprocessor mode.
RESET	Reset input	Input	A "L" on this input resets the microcomputer.
XIN XOUT	Clock input Clock output	Input Output	These pins are provided for the main clock generating circuit. Connect a ceramic resonator or crystal between the XIN and the XOUT pins. To use an externally derived clock, input it to the XIN pin and leave the XOUT pin open.
BYTE	External data bus width select input	Input	This pin selects the width of an external data bus. A 16-bit width is selected when this input is "L"; an 8-bit width is selected when this input is "H". This input must be fixed to either "H" or "L".
AVcc	Analog power supply input		This pin is a power supply input for the A-D converter. Connect this pin to Vcc.
AVss	Analog power supply input		This pin is a power supply input for the A-D converter. Connect this pin to Vss.
VREF	Reference voltage input	Input	This pin is a reference voltage input for the A-D converter.
P00 to P07	I/O port P0	Input/output	This is an 8-bit CMOS I/O port. It has an input/output port direction register that allows the user to set each pin for input or output individually.
D0 to D7		Input/output	When set as a separate bus, these pins input and output data (D0–D7).
P10 to P17	I/O port P1	Input/output	This is an 8-bit I/O port equivalent to P0. Pins in this port also function as external interrupt pins as selected by software.
D8 to D15		Input/output	When set as a separate bus, these pins input and output data (D8–D15).
P20 to P27	I/O port P2	Input/output	This is an 8-bit I/O port equivalent to P0.
A0 to A7		Output	These pins output 8 low-order address bits (A0–A7).
A0/D0 to A7/D7		Input/output	If the external bus is set as an 8-bit wide multiplexed bus, these pins input and output data (D0–D7) and output 8 low-order address bits (A0–A7) separated in time by multiplexing.
A0, A1/D0 to A7/D6		Output Input/output	If the external bus is set as a 16-bit wide multiplexed bus, these pins input and output data (D0–D6) and output address (A1–A7) separated in time by multiplexing. They also output address (A0).
P30 to P37	I/O port P3	Input/output	This is an 8-bit I/O port equivalent to P0.
A8 to A15		Output	These pins output 8 middle-order address bits (A8–A15).
A8/D7, A9 to A15		Input/output Output	If the external bus is set as a 16-bit wide multiplexed bus, these pins input and output data (D7) and output address (A8) separated in time by multiplexing. They also output address (A9–A15).
P40 to P47	I/O port P4	Input/output	This is an 8-bit I/O port equivalent to P0.
CS0,CS2,CS3, A16 to A19		Output Output	These pins output CS0,CS2,CS3 signals and A16–A19. CS0,CS2,CS3 are chip select signals used to specify an access space. A16–A19 are 4 high- order address bits.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Table 1.5.3 Pin Description**

Pin name	Signal name	I/O type	Function
P50 to P57	I/O port P5	Input/output	This is an 8-bit I/O port equivalent to P0.
$\overline{\text{WRL}}$ / $\overline{\text{WR}}$ , $\overline{\text{WRH}}$ / $\overline{\text{BHE}}$ , $\overline{\text{RD}}$ , $\overline{\text{BCLK}}$ , $\overline{\text{HLDA}}$ , $\overline{\text{HOLD}}$ ,  $\overline{\text{ALE}}$ , $\overline{\text{RDY}}$		Output Output Output Output Output Input  Output Input	<p>Output <math>\overline{\text{WRL}}</math>, <math>\overline{\text{WRH}}</math> (<math>\overline{\text{WR}}</math> and <math>\overline{\text{BHE}}</math>), <math>\overline{\text{RD}}</math>, <math>\overline{\text{BCLK}}</math>, <math>\overline{\text{HLDA}}</math>, and <math>\overline{\text{ALE}}</math> signals. <math>\overline{\text{WRL}}</math> and <math>\overline{\text{WRH}}</math>, and <math>\overline{\text{BHE}}</math> and <math>\overline{\text{WR}}</math> can be switched using software control.</p> <p>■ <math>\overline{\text{WRL}}</math>, <math>\overline{\text{WRH}}</math>, and <math>\overline{\text{RD}}</math> selected  With a 16-bit external data bus, data is written to even addresses when the <math>\overline{\text{WRL}}</math> signal is "L" and to the odd addresses when the <math>\overline{\text{WRH}}</math> signal is "L". Data is read when <math>\overline{\text{RD}}</math> is "L".</p> <p>■ <math>\overline{\text{WR}}</math>, <math>\overline{\text{BHE}}</math>, and <math>\overline{\text{RD}}</math> selected  Data is written when <math>\overline{\text{WR}}</math> is "L". Data is read when <math>\overline{\text{RD}}</math> is "L". Odd addresses are accessed when <math>\overline{\text{BHE}}</math> is "L". Use this mode when using an 8-bit external data bus.</p> <p>While the input level at the <math>\overline{\text{HOLD}}</math> pin is "L", the microcomputer is placed in the hold state. While in the hold state, <math>\overline{\text{HLDA}}</math> outputs a "L" level. <math>\overline{\text{ALE}}</math> is used to latch the address. While the input level of the <math>\overline{\text{RDY}}</math> pin is "L", the microcomputer is in the ready state.</p>
P60 to P67	I/O port P6	Input/output	This is an 8-bit I/O port equivalent to P0. The port can be set to have or not have a pull-up resistor in units of four bits by software. Pins in this port also function as UART0 and UART1 I/O pins as selected by software.
P70 to P77	I/O port P7	Input/output	This is an 8-bit I/O port equivalent to P6 (P70 and P71 are N channel open-drain output). Pins in this port also function as timer A0–A3, timer B5 or UART2 I/O pins as selected by software.
P80 to P84, P86,  P87,  P85	I/O port P8    I/O port P85	Input/output Input/output  Input/output  Input	<p>P80 to P84, P86, and P87 are I/O ports with the same functions as P6. Using software, they can be made to function as the I/O pins for timer A4 and the input pins for external interrupts. P86 and P87 can be set using software to function as the I/O pins for a sub clock generation circuit. In this case, connect a quartz oscillator between P86 (XCOUT pin) and P87 (XCIN pin). P85 is an input-only port that also functions for NMI. The NMI interrupt is generated when the input at this pin changes from "H" to "L". The NMI function cannot be cancelled using software. The pull-up cannot be set for this pin.</p>
P90 to P97	I/O port P9	Input/output	This is an 8-bit I/O port equivalent to P6. Pins in this port also function as SI/O3, 4 I/O pins, Timer B0–B4 input pins, D-A converter output pins, A-D converter extended input pins, or A-D trigger input pins as selected by software.
P100 to P107	I/O port P10	Input/output	This is an 8-bit I/O port equivalent to P6. Pins in this port also function as A-D converter input pins. Furthermore, P104–P107 also function as input pins for the key input interrupt function.
VDD1	Power supply input		Digital power supply pin. Connect to +5 V.
VDD2	Power supply input		Analog power supply pin. Connect to +5 V.
VDD3	Power supply input		Analog power supply pin. Connect to +5 V.
CVIDEO1	Composite video output 1	Output	This is composite video signal output pin. Output 2 Vp-p composite video signal. In superimpose mode, this pin's signal consists of CVIN1 signal of the display range combined with the character output signal.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Table 1.5.4 Pin Description**

Pin name	Signal name	I/O type	Function
CVIDEO2	Composite video output 2	Output	This is composite video signal output pin. Output 2 Vp-p composite video signal. This pin's signal consists of CVIN2 signal of vertical blanking erase interval combined with the VBI output signal.
SVREF	Synchronous slice level input	Input	When slice the vertical synchronous signal, input slice power.
CVIN1	Composite video signal input 1	Input	This pin inputs the external composite video signal. In superimpose mode, this pin's signal consists of it's composite video signal combined with the character output signal. Data slices this signal internally by setting.
SECAMIN	SECAM input	Input	Carrier input pin for SECAM.
CVIN2	Composite video signal input 2	Input	This pin inputs the external composite video signal. In VBI encode, this pin's signal consists of it's composite video signal combined with the VBI output signal. Data slices this signal internally by setting.
SYNCIN	Composite video signal input 3	Input	This pin inputs the external composite video signal. Synchronous divides this signal internally.
HOR			Input digital horizontal synchronous signal (5 V).
LP1	Filter output 1	Output	This is filter output pin 1 (for display).
LP2	Filter output 2	Output	This is filter output pin 2 (for synchronous).
LP3	Filter output 3	Output	This is filter output pin 3 (for VBI, VPS).
LP4	Filter output 4	Output	This is filter output pin 4 (for PDC).
FSCIN	fsc input pin for synchronous signal generation	Input	Sub-carrier (fsc) input pin for synchronous signal generation.
VERT	Vertical synchronous signal input	Input	Digital vertical synchronous signal input (5 V).
P110 to P118	Output port P11	Output	This is a 9-bit output-only port. Pins in this port also function as EDO2, EDO1, CSYN, BLNK, B, G, R, GRAY, SLICEON output pins as selected by software.



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2. Operation of Functional Blocks

The M306H0SFP accommodates certain units in a single chip. These units include RAM to store instructions and data and the central processing unit (CPU) to execute arithmetic/logic operations. Also included are peripheral units such as timers, serial I/O, D-A converter, DMAC, CRC calculation circuit, A-D converter, OSD circuit, Data slicer circuit, Data encode circuit and I/O ports.

The following explains each unit.

### 2.1 Memory

Figure 2.1.1 is a memory map of the M306H0SFP. The address space extends the 1M bytes from address 00000<sub>16</sub> to FFFFF<sub>16</sub>. From address FFFFF<sub>16</sub> down is ROM. In the M306H0SFP, can use from address from 04000<sub>16</sub> to FFFFF<sub>16</sub> as external ROM area. The vector table for fixed interrupts such as the reset and  $\overline{\text{NMI}}$  are mapped to from address FFFDC<sub>16</sub> to FFFFF<sub>16</sub>. The starting address of the interrupt routine is stored here. The address of the vector table for timer interrupts, etc., can be set as desired using the internal register (INTB). See the section on interrupts for details.

5K bytes of internal RAM is mapped to from address 00400<sub>16</sub> to 017FF<sub>16</sub>. In addition to storing data, the RAM also stores the stack used when calling subroutines and when interrupts are generated.

The SFR area is mapped to from address 00000<sub>16</sub> to 003FF<sub>16</sub>. This area accommodates the control registers for peripheral devices such as I/O ports, A-D converter, serial I/O, and timers, etc. Figures 2.1.2 to 2.1.4 are location of peripheral unit control registers. Any part of the SFR area that is not occupied is reserved and cannot be used for other purposes.

The special page vector table is mapped to from address FFE00<sub>16</sub> to FFFDB<sub>16</sub>. If the starting addresses of subroutines or the destination addresses of jumps are stored here, subroutine call instructions and jump instructions can be used as 2-byte instructions, reducing the number of program steps.

Address 01800<sub>16</sub> to 03FFF<sub>16</sub> and address 28000<sub>16</sub> to 2FFFF<sub>16</sub> are reserved and cannot be used.

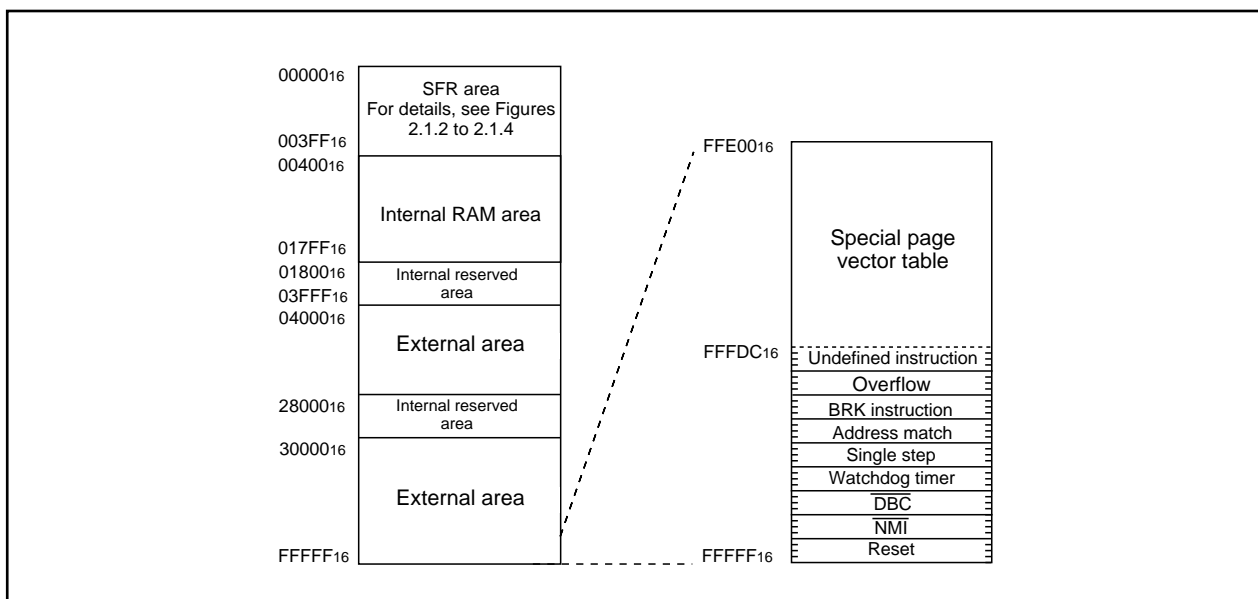


Figure 2.1.1 Memory map

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

0000 <sub>16</sub>		0044 <sub>16</sub>	INT3 interrupt control register(INT3IC)
0001 <sub>16</sub>		0045 <sub>16</sub>	Timer B5 interrupt control register (TB5IC)
0002 <sub>16</sub>		0046 <sub>16</sub>	Timer B4 interrupt control register (TB4IC)
0003 <sub>16</sub>		0047 <sub>16</sub>	Timer B3 interrupt control register (TB3IC)
0004 <sub>16</sub>	Processor mode register 0 (PM0)	0048 <sub>16</sub>	SI/O4 interrupt control register (S4IC)
0005 <sub>16</sub>	Processor mode register 1(PM1)		INT5 interrupt control register(INT5IC)
0006 <sub>16</sub>	System clock control register 0 (CM0)	0049 <sub>16</sub>	SI/O3 interrupt control register (S3IC)
0007 <sub>16</sub>	System clock control register 1 (CM1)		INT4 interrupt control register(INT4IC)
0008 <sub>16</sub>	Chip select control register (CSR)	004A <sub>16</sub>	Bus collision detection interrupt control register (BCNIC)
0009 <sub>16</sub>	Address match interrupt enable register (AIER)	004B <sub>16</sub>	DMA0 interrupt control register (DM0IC)
000A <sub>16</sub>	Protect register (PRCR)	004C <sub>16</sub>	DMA1 interrupt control register (DM1IC)
000B <sub>16</sub>		004D <sub>16</sub>	Key input interrupt control register (KUPIC)
000C <sub>16</sub>		004E <sub>16</sub>	A-D conversion interrupt control register (ADIC)
000D <sub>16</sub>		004F <sub>16</sub>	UART2 transmit interrupt control register (S2TIC)
000E <sub>16</sub>	Watchdog timer start register (WDTS)	0050 <sub>16</sub>	UART2 receive interrupt control register (S2RIC)
000F <sub>16</sub>	Watchdog timer control register (WDC)	0051 <sub>16</sub>	UART0 transmit interrupt control register (S0TIC)
0010 <sub>16</sub>		0052 <sub>16</sub>	UART0 receive interrupt control register (S0RIC)
0011 <sub>16</sub>	Address match interrupt register 0 (RMAD0)	0053 <sub>16</sub>	UART1 transmit interrupt control register (S1TIC)
0012 <sub>16</sub>		0054 <sub>16</sub>	UART1 receive interrupt control register (S1RIC)
0013 <sub>16</sub>		0055 <sub>16</sub>	Timer A0 interrupt control register (TA0IC)
0014 <sub>16</sub>		0056 <sub>16</sub>	Timer A1 interrupt control register (TA1IC)
0015 <sub>16</sub>	Address match interrupt register 1 (RMAD1)	0057 <sub>16</sub>	Timer A2 interrupt control register (TA2IC)
0016 <sub>16</sub>		0058 <sub>16</sub>	Timer A3 interrupt control register (TA3IC)
0017 <sub>16</sub>		0059 <sub>16</sub>	Timer A4 interrupt control register (TA4IC)
0018 <sub>16</sub>		005A <sub>16</sub>	Timer B0 interrupt control register (TB0IC)
0019 <sub>16</sub>		005B <sub>16</sub>	Timer B1 interrupt control register (TB1IC)
001A <sub>16</sub>		005C <sub>16</sub>	Timer B2 interrupt control register (TB2IC)
001B <sub>16</sub>		005D <sub>16</sub>	INT0 interrupt control register (INT0IC)
001C <sub>16</sub>		005E <sub>16</sub>	INT1 interrupt control register (INT1IC)
001D <sub>16</sub>		005F <sub>16</sub>	INT2 interrupt control register (INT2IC)
001E <sub>16</sub>		0060 <sub>16</sub>	
001F <sub>16</sub>		≈	≈
0020 <sub>16</sub>		0200 <sub>16</sub>	
0021 <sub>16</sub>	DMA0 source pointer (SAR0)	0201 <sub>16</sub>	
0022 <sub>16</sub>		0202 <sub>16</sub>	Display RAM address control register
0023 <sub>16</sub>		0203 <sub>16</sub>	
0024 <sub>16</sub>		0204 <sub>16</sub>	Display RAM data control register
0025 <sub>16</sub>	DMA0 destination pointer (DAR0)	0205 <sub>16</sub>	
0026 <sub>16</sub>		0206 <sub>16</sub>	Font RAM address control register
0027 <sub>16</sub>		0207 <sub>16</sub>	
0028 <sub>16</sub>	DMA0 transfer counter (TCR0)	0208 <sub>16</sub>	Font RAM data control register
0029 <sub>16</sub>		0209 <sub>16</sub>	
002A <sub>16</sub>		020A <sub>16</sub>	SYRAM address control register
002B <sub>16</sub>		020B <sub>16</sub>	
002C <sub>16</sub>	DMA0 control register (DM0CON)	020C <sub>16</sub>	SYRAM data control register
002D <sub>16</sub>		020D <sub>16</sub>	
002E <sub>16</sub>		020E <sub>16</sub>	Slice RAM address control register
002F <sub>16</sub>		020F <sub>16</sub>	
0030 <sub>16</sub>		0210 <sub>16</sub>	Slice RAM data control register
0031 <sub>16</sub>	DMA1 source pointer (SAR1)	0211 <sub>16</sub>	
0032 <sub>16</sub>		0212 <sub>16</sub>	VBIRAM address control register
0033 <sub>16</sub>		0213 <sub>16</sub>	
0034 <sub>16</sub>		0214 <sub>16</sub>	VBIRAM data control register
0035 <sub>16</sub>	DMA1 destination pointer (DAR1)	0215 <sub>16</sub>	
0036 <sub>16</sub>		0216 <sub>16</sub>	Address control register for expansion register
0037 <sub>16</sub>		0217 <sub>16</sub>	
0038 <sub>16</sub>	DMA1 transfer counter (TCR1)	0218 <sub>16</sub>	Data control register for expansion register
0039 <sub>16</sub>		0219 <sub>16</sub>	
003A <sub>16</sub>		021A <sub>16</sub>	Humming 8/4 register
003B <sub>16</sub>		021B <sub>16</sub>	
003C <sub>16</sub>	DMA1 control register (DM1CON)	021C <sub>16</sub>	Humming 24/18 register 0
003D <sub>16</sub>		021D <sub>16</sub>	
003E <sub>16</sub>		021E <sub>16</sub>	Humming 24/18 register 1
003F <sub>16</sub>		021F <sub>16</sub>	
0040 <sub>16</sub>		0220 <sub>16</sub>	
0041 <sub>16</sub>		≈	≈
0042 <sub>16</sub>		033F <sub>16</sub>	
0043 <sub>16</sub>			

Figure 2.1.2 Location of peripheral unit control registers (1)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

0340 <sub>16</sub>	Timer B3, 4, 5 count start flag (TBSR)	0380 <sub>16</sub>	Count start flag (TABSR)
0341 <sub>16</sub>		0381 <sub>16</sub>	Clock prescaler reset flag (CPSRF)
0342 <sub>16</sub>		0382 <sub>16</sub>	One-shot start flag (ONSF)
0343 <sub>16</sub>	Timer A1-1 register (TA11)	0383 <sub>16</sub>	Trigger select register (TRGSR)
0344 <sub>16</sub>		0384 <sub>16</sub>	Up-down flag (UDF)
0345 <sub>16</sub>	Timer A2-1 register (TA21)	0385 <sub>16</sub>	
0346 <sub>16</sub>		0386 <sub>16</sub>	
0347 <sub>16</sub>	Timer A4-1 register (TA41)	0387 <sub>16</sub>	Timer A0 (TA0)
0348 <sub>16</sub>	Reserved register	0388 <sub>16</sub>	
0349 <sub>16</sub>	Reserved register	0389 <sub>16</sub>	Timer A1 (TA1)
034A <sub>16</sub>	Reserved register	038A <sub>16</sub>	
034B <sub>16</sub>	Reserved register	038B <sub>16</sub>	Timer A2 (TA2)
034C <sub>16</sub>	Reserved register	038C <sub>16</sub>	
034D <sub>16</sub>	Reserved register	038D <sub>16</sub>	Timer A3 (TA3)
034E <sub>16</sub>		038E <sub>16</sub>	
034F <sub>16</sub>		038F <sub>16</sub>	Timer A4 (TA4)
0350 <sub>16</sub>		0390 <sub>16</sub>	
0351 <sub>16</sub>	Timer B3 register (TB3)	0391 <sub>16</sub>	Timer B0 (TB0)
0352 <sub>16</sub>		0392 <sub>16</sub>	
0353 <sub>16</sub>	Timer B4 register (TB4)	0393 <sub>16</sub>	Timer B1 (TB1)
0354 <sub>16</sub>		0394 <sub>16</sub>	
0355 <sub>16</sub>	Timer B5 register (TB5)	0395 <sub>16</sub>	Timer B2 (TB2)
0356 <sub>16</sub>		0396 <sub>16</sub>	Timer A0 mode register (TA0MR)
0357 <sub>16</sub>		0397 <sub>16</sub>	Timer A1 mode register (TA1MR)
0358 <sub>16</sub>		0398 <sub>16</sub>	Timer A2 mode register (TA2MR)
0359 <sub>16</sub>		0399 <sub>16</sub>	Timer A3 mode register (TA3MR)
035A <sub>16</sub>		039A <sub>16</sub>	Timer A4 mode register (TA4MR)
035B <sub>16</sub>	Timer B3 mode register (TB3MR)	039B <sub>16</sub>	Timer B0 mode register (TB0MR)
035C <sub>16</sub>	Timer B4 mode register (TB4MR)	039C <sub>16</sub>	Timer B1 mode register (TB1MR)
035D <sub>16</sub>	Timer B5 mode register (TB5MR)	039D <sub>16</sub>	Timer B2 mode register (TB2MR)
035E <sub>16</sub>		039E <sub>16</sub>	
035F <sub>16</sub>	Interrupt cause select register (IFSR)	039F <sub>16</sub>	
0360 <sub>16</sub>	SI/O3 transmit/receive register (S3TRR)	03A0 <sub>16</sub>	UART0 transmit/receive mode register (U0MR)
0361 <sub>16</sub>		03A1 <sub>16</sub>	UART0 bit rate generator (U0BRG)
0362 <sub>16</sub>	SI/O3 control register (S3C)	03A2 <sub>16</sub>	
0363 <sub>16</sub>	SI/O3 bit rate generator (S3BRG)	03A3 <sub>16</sub>	UART0 transmit buffer register (U0TB)
0364 <sub>16</sub>	SI/O4 transmit/receive register (S4TRR)	03A4 <sub>16</sub>	UART0 transmit/receive control register 0 (U0C0)
0365 <sub>16</sub>		03A5 <sub>16</sub>	UART0 transmit/receive control register 1 (U0C1)
0366 <sub>16</sub>	SI/O4 control register (S4C)	03A6 <sub>16</sub>	
0367 <sub>16</sub>	SI/O4 bit rate generator (S4BRG)	03A7 <sub>16</sub>	UART0 receive buffer register (U0RB)
0368 <sub>16</sub>		03A8 <sub>16</sub>	UART1 transmit/receive mode register (U1MR)
0369 <sub>16</sub>		03A9 <sub>16</sub>	UART1 bit rate generator (U1BRG)
036A <sub>16</sub>		03AA <sub>16</sub>	
036B <sub>16</sub>		03AB <sub>16</sub>	UART1 transmit buffer register (U1TB)
036C <sub>16</sub>		03AC <sub>16</sub>	UART1 transmit/receive control register 0 (U1C0)
036D <sub>16</sub>		03AD <sub>16</sub>	UART1 transmit/receive control register 1 (U1C1)
036E <sub>16</sub>		03AE <sub>16</sub>	
036F <sub>16</sub>		03AF <sub>16</sub>	UART1 receive buffer register (U1RB)
0370 <sub>16</sub>		03B0 <sub>16</sub>	UART transmit/receive control register 2 (UCON)
0371 <sub>16</sub>		03B1 <sub>16</sub>	
0372 <sub>16</sub>		03B2 <sub>16</sub>	
0373 <sub>16</sub>		03B3 <sub>16</sub>	
0374 <sub>16</sub>		03B4 <sub>16</sub>	
0375 <sub>16</sub>	UART2 special mode register 3(U2SMR3)	03B5 <sub>16</sub>	
0376 <sub>16</sub>	UART2 special mode register 2(U2SMR2)	03B6 <sub>16</sub>	
0377 <sub>16</sub>	UART2 special mode register (U2SMR)	03B7 <sub>16</sub>	
0378 <sub>16</sub>	UART2 transmit/receive mode register (U2MR)	03B8 <sub>16</sub>	DMA0 request cause select register (DM0SL)
0379 <sub>16</sub>	UART2 bit rate generator (U2BRG)	03B9 <sub>16</sub>	
037A <sub>16</sub>		03BA <sub>16</sub>	DMA1 request cause select register (DM1SL)
037B <sub>16</sub>	UART2 transmit buffer register (U2TB)	03BB <sub>16</sub>	
037C <sub>16</sub>	UART2 transmit/receive control register 0 (U2C0)	03BC <sub>16</sub>	
037D <sub>16</sub>	UART2 transmit/receive control register 1 (U2C1)	03BD <sub>16</sub>	CRC data register (CRCD)
037E <sub>16</sub>		03BE <sub>16</sub>	CRC input register (CRCIN)
037F <sub>16</sub>	UART2 receive buffer register (U2RB)	03BF <sub>16</sub>	

Figure 2.1.3 Location of peripheral unit control registers (2)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

03C0 <sub>16</sub>	A-D register 0 (AD0)
03C1 <sub>16</sub>	Reserved register
03C2 <sub>16</sub>	A-D register 1 (AD1)
03C3 <sub>16</sub>	Reserved register
03C4 <sub>16</sub>	A-D register 2 (AD2)
03C5 <sub>16</sub>	Reserved register
03C6 <sub>16</sub>	A-D register 3 (AD3)
03C7 <sub>16</sub>	Reserved register
03C8 <sub>16</sub>	A-D register 4 (AD4)
03C9 <sub>16</sub>	Reserved register
03CA <sub>16</sub>	A-D register 5 (AD5)
03CB <sub>16</sub>	Reserved register
03CC <sub>16</sub>	A-D register 6 (AD6)
03CD <sub>16</sub>	Reserved register
03CE <sub>16</sub>	A-D register 7 (AD7)
03CF <sub>16</sub>	Reserved register
03D0 <sub>16</sub>	
03D1 <sub>16</sub>	
03D2 <sub>16</sub>	
03D3 <sub>16</sub>	
03D4 <sub>16</sub>	A-D control register 2 (ADCON2)
03D5 <sub>16</sub>	
03D6 <sub>16</sub>	A-D control register 0 (ADCON0)
03D7 <sub>16</sub>	A-D control register 1 (ADCON1)
03D8 <sub>16</sub>	D-A register 0 (DA0)
03D9 <sub>16</sub>	
03DA <sub>16</sub>	D-A register 1 (DA1)
03DB <sub>16</sub>	
03DC <sub>16</sub>	D-A control register (DACON)
03DD <sub>16</sub>	
03DE <sub>16</sub>	
03DF <sub>16</sub>	
03E0 <sub>16</sub>	Port P0 (P0)
03E1 <sub>16</sub>	Port P1 (P1)
03E2 <sub>16</sub>	Port P0 direction register (PD0)
03E3 <sub>16</sub>	Port P1 direction register (PD1)
03E4 <sub>16</sub>	Port P2 (P2)
03E5 <sub>16</sub>	Port P3 (P3)
03E6 <sub>16</sub>	Port P2 direction register (PD2)
03E7 <sub>16</sub>	Port P3 direction register (PD3)
03E8 <sub>16</sub>	Port P4 (P4)
03E9 <sub>16</sub>	Port P5 (P5)
03EA <sub>16</sub>	Port P4 direction register (PD4)
03EB <sub>16</sub>	Port P5 direction register (PD5)
03EC <sub>16</sub>	Port P6 (P6)
03ED <sub>16</sub>	Port P7 (P7)
03EE <sub>16</sub>	Port P6 direction register (PD6)
03EF <sub>16</sub>	Port P7 direction register (PD7)
03F0 <sub>16</sub>	Port P8 (P8)
03F1 <sub>16</sub>	Port P9 (P9)
03F2 <sub>16</sub>	Port P8 direction register (PD8)
03F3 <sub>16</sub>	Port P9 direction register (PD9)
03F4 <sub>16</sub>	Port P10 (P10)
03F5 <sub>16</sub>	
03F6 <sub>16</sub>	Port P10 direction register (PD10)
03F7 <sub>16</sub>	
03F8 <sub>16</sub>	
03F9 <sub>16</sub>	
03FA <sub>16</sub>	
03FB <sub>16</sub>	
03FC <sub>16</sub>	Pull-up control register 0 (PUR0)
03FD <sub>16</sub>	Pull-up control register 1 (PUR1)
03FE <sub>16</sub>	Pull-up control register 2 (PUR2)
03FF <sub>16</sub>	Port control register (PCR)

Figure 2.1.4 Location of peripheral unit control registers (3)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.2 Central Processing Unit (CPU)

The CPU has 13 registers shown in Figure 2.2.1. Seven of these registers (R0, R1, R2, R3, A0, A1, and FB) come in two sets; therefore, these have two register banks.

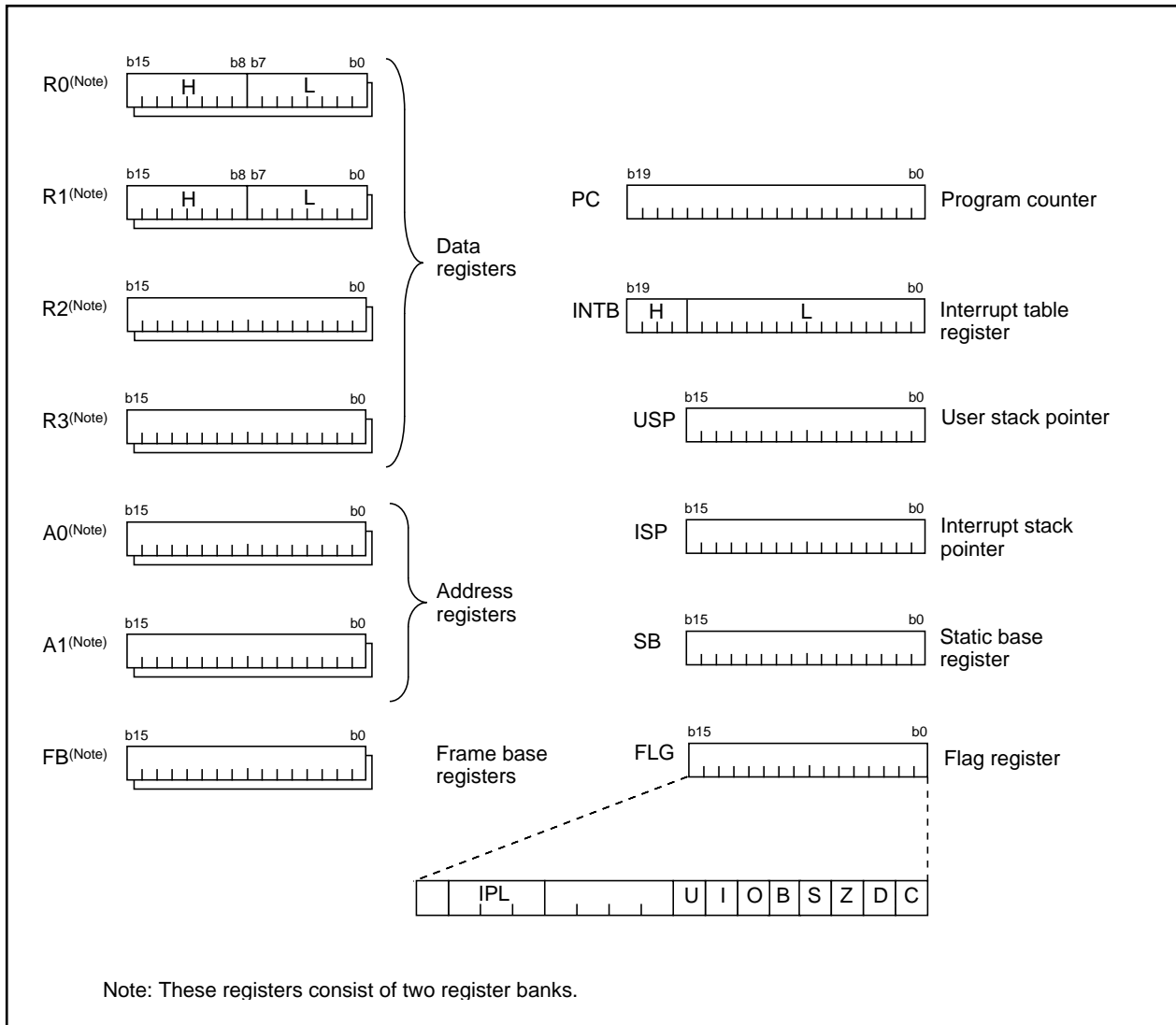


Figure 2.2.1 Central processing unit register

### (1) Data registers (R0, R0H, R0L, R1, R1H, R1L, R2, and R3)

Data registers (R0, R1, R2, and R3) are configured with 16 bits, and are used primarily for transfer and arithmetic/logic operations.

Registers R0 and R1 each can be used as separate 8-bit data registers, high-order bits as (R0H/R1H), and low-order bits as (R0L/R1L). In some instructions, registers R2 and R0, as well as R3 and R1 can use as 32-bit data registers (R2R0/R3R1).

### (2) Address registers (A0 and A1)

Address registers (A0 and A1) are configured with 16 bits, and have functions equivalent to those of data registers. These registers can also be used for address register indirect addressing and address register relative addressing.

In some instructions, registers A1 and A0 can be combined for use as a 32-bit address register (A1A0).

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

### (3) Frame base register (FB)

Frame base register (FB) is configured with 16 bits, and is used for FB relative addressing.

### (4) Program counter (PC)

Program counter (PC) is configured with 20 bits, indicating the address of an instruction to be executed.

### (5) Interrupt table register (INTB)

Interrupt table register (INTB) is configured with 20 bits, indicating the start address of an interrupt vector table.

### (6) Stack pointer (USP/ISP)

Stack pointer comes in two types: user stack pointer (USP) and interrupt stack pointer (ISP), each configured with 16 bits.

Your desired type of stack pointer (USP or ISP) can be selected by a stack pointer select flag (U flag).

This flag is located at the position of bit 7 in the flag register (FLG).

### (7) Static base register (SB)

Static base register (SB) is configured with 16 bits, and is used for SB relative addressing.

### (8) Flag register (FLG)

Flag register (FLG) is configured with 11 bits, each bit is used as a flag. Figure 2.2.2 shows the flag register (FLG). The following explains the function of each flag:

- **Bit 0: Carry flag (C flag)**

This flag retains a carry, borrow, or shift-out bit that has occurred in the arithmetic/logic unit.

- **Bit 1: Debug flag (D flag)**

This flag enables a single-step interrupt.

When this flag is "1", a single-step interrupt is generated after instruction execution. This flag is cleared to "0" when the interrupt is acknowledged.

- **Bit 2: Zero flag (Z flag)**

This flag is set to "1" when an arithmetic operation resulted in 0; otherwise, cleared to "0".

- **Bit 3: Sign flag (S flag)**

This flag is set to "1" when an arithmetic operation resulted in a negative value; otherwise, cleared to "0".

- **Bit 4: Register bank select flag (B flag)**

This flag chooses a register bank. Register bank 0 is selected when this flag is "0"; register bank 1 is selected when this flag is "1".

- **Bit 5: Overflow flag (O flag)**

This flag is set to "1" when an arithmetic operation resulted in overflow; otherwise, cleared to "0".

- **Bit 6: Interrupt enable flag (I flag)**

This flag enables a maskable interrupt.

An interrupt is disabled when this flag is "0", and is enabled when this flag is "1". This flag is cleared to "0" when the interrupt is acknowledged.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

- **Bit 7: Stack pointer select flag (U flag)**

Interrupt stack pointer (ISP) is selected when this flag is “0” ; user stack pointer (USP) is selected when this flag is “1”.

This flag is cleared to “0” when a hardware interrupt is acknowledged or an INT instruction of software interrupt Nos. 0 to 31 is executed.

- **Bits 8 to 11: Reserved area**

- **Bits 12 to 14: Processor interrupt priority level (IPL)**

Processor interrupt priority level (IPL) is configured with three bits, for specification of up to eight processor interrupt priority levels from level 0 to level 7.

If a requested interrupt has priority greater than the processor interrupt priority level (IPL), the interrupt is enabled.

- **Bit 15: Reserved area**

The C, Z, S, and O flags are changed when instructions are executed. See the software manual for details.

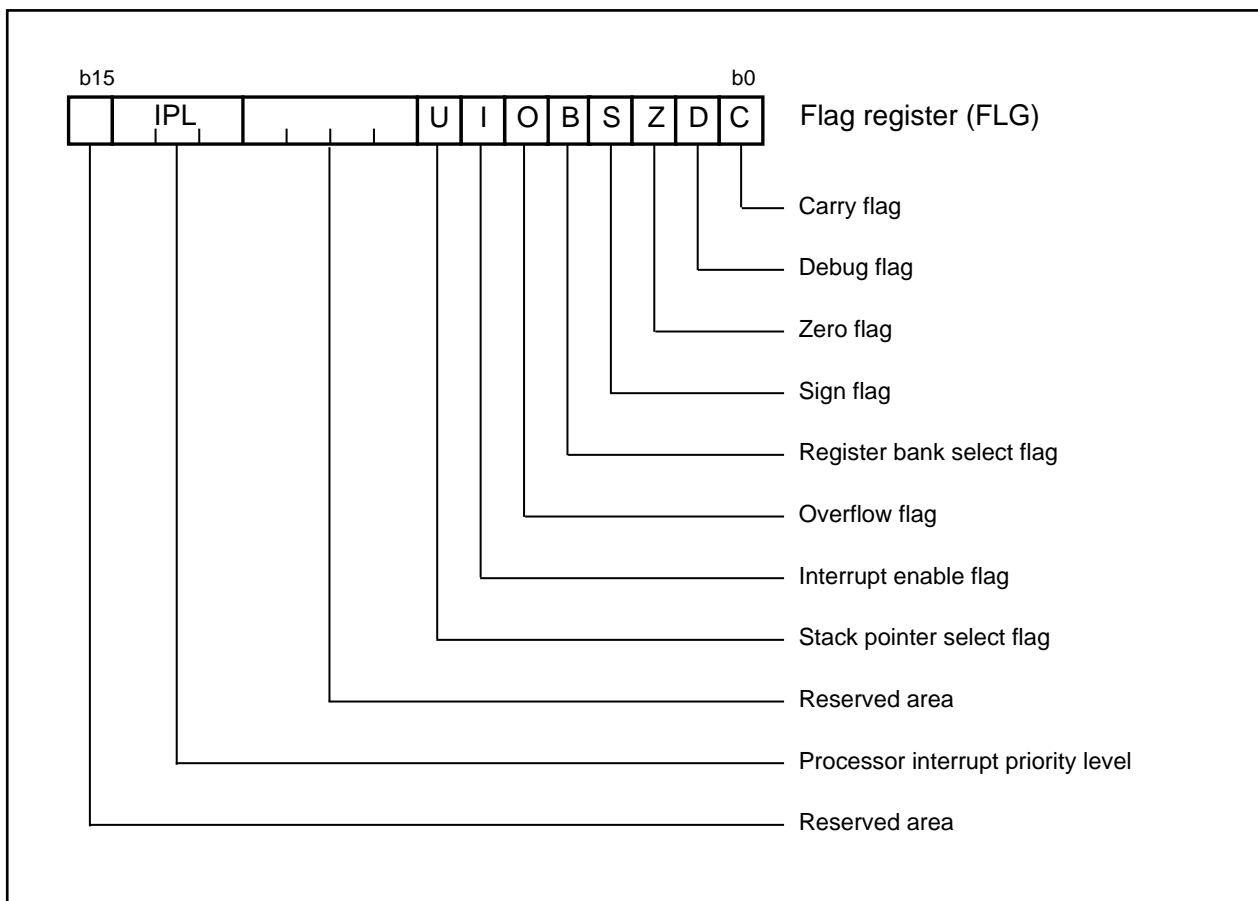


Figure 2.2.2 Flag register (FLG)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.3 Reset

There are two kinds of resets; hardware and software. In both cases, operation is the same after the reset. (See "Software Reset" for details of software resets.) This section explains on hardware resets.

When the supply voltage is in the range where operation is guaranteed, a reset is effected by holding the reset pin level "L" (0.2V<sub>CC</sub> max.) for at least 20 cycles. When the reset pin level is then returned to the "H" level while main clock is stable, the reset status is cancelled and program execution resumes from the address in the reset vector table.

Figure 2.3.1 shows the example reset circuit. Figure 2.3.2 shows the reset sequence.

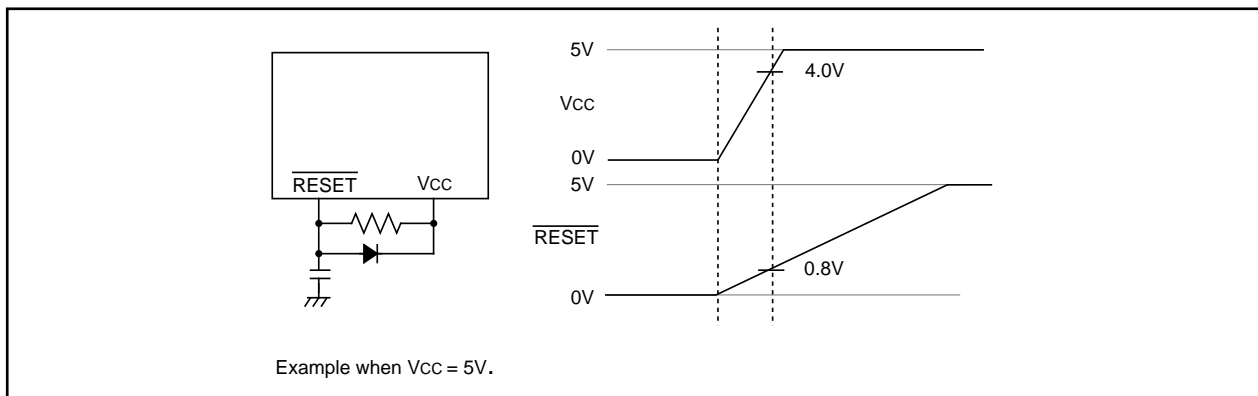


Figure 2.3.1 Example reset circuit

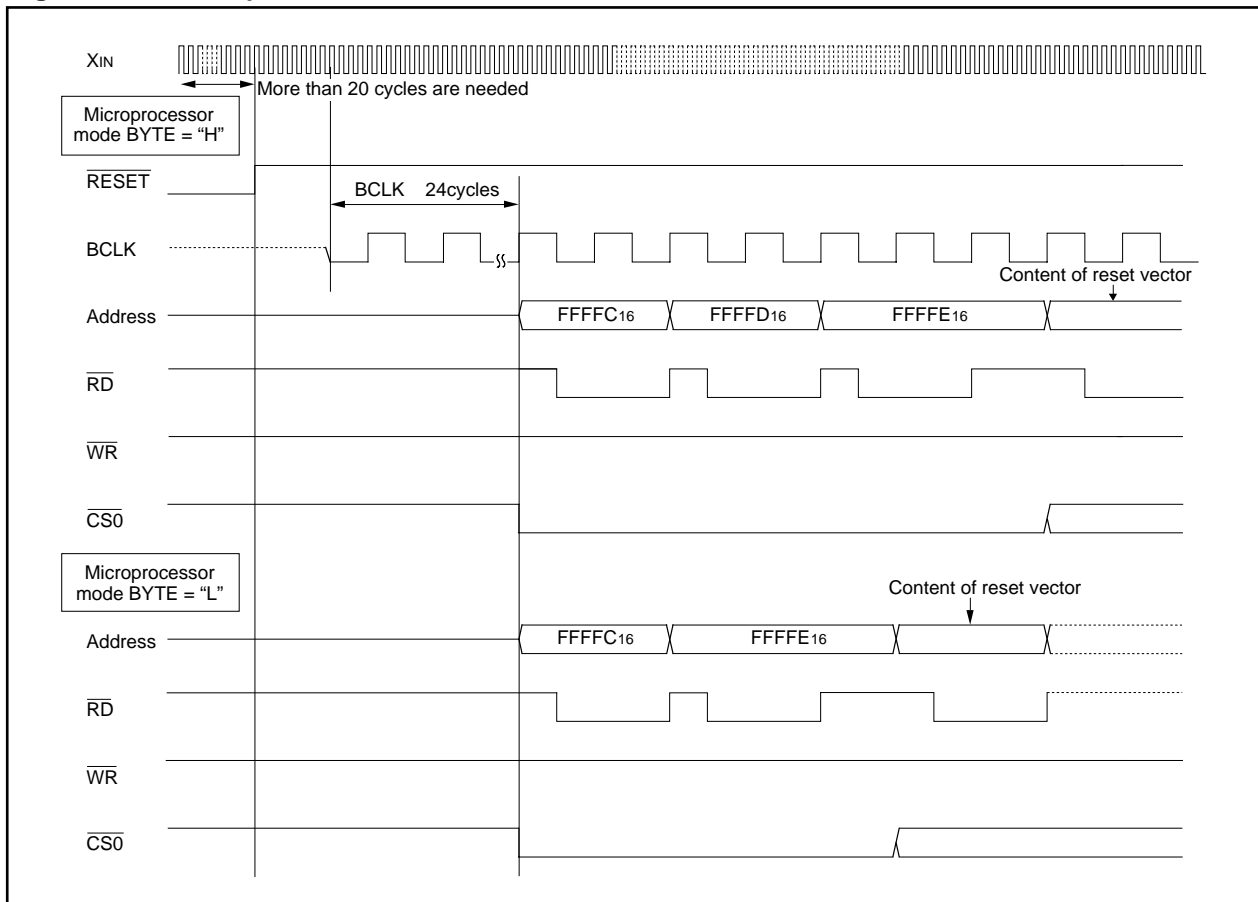


Figure 2.3.2 Reset sequence



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Table 2.3.1 shows the statuses of the other pins while the  $\overline{\text{RESET}}$  pin level is "L". Figures 2.3.3 and 2.3.4 show the internal status of the microcomputer immediately after the reset is cancelled.

**Table 2.3.1 Pin status when  $\overline{\text{RESET}}$  pin level is "L"**

Pin name	Status	
	CNVss = Vcc	
	BYTE = Vss	BYTE = Vcc
P0	Data input (floating)	Data input (floating)
P1	Data input (floating)	Input port (floating)
P2, P3, P40 to P43	Address output (undefined)	Address output (undefined)
P44	$\overline{\text{CS0}}$ output ("H" level is output)	$\overline{\text{CS0}}$ output ("H" level is output)
P45 to P47	Input port (floating) (pull-up resistor is on)	Input port (floating) (pull-up resistor is on)
P50	$\overline{\text{WR}}$ output ("H" level is output)	$\overline{\text{WR}}$ output ("H" level is output)
P51	$\overline{\text{BHE}}$ output (undefined)	$\overline{\text{BHE}}$ output (undefined)
P52	$\overline{\text{RD}}$ output ("H" level is output)	$\overline{\text{RD}}$ output ("H" level is output)
P53	BCLK output	BCLK output
P54	$\overline{\text{HLDA}}$ output (The output value depends on the input to the HOLD pin)	$\overline{\text{HLDA}}$ output (The output value depends on the input to the HOLD pin)
P55	$\overline{\text{HOLD}}$ input (floating)	$\overline{\text{HOLD}}$ input (floating)
P56	$\overline{\text{ALE}}$ output ("L" level is output)	$\overline{\text{ALE}}$ output ("L" level is output)
P57	$\overline{\text{RDY}}$ input (floating)	$\overline{\text{RDY}}$ input (floating)
P6, P7, P80 to P84, P86, P87, P9, P10	Input port (floating)	Input port (floating)
P110 to P118	Output port	Output port
CVIDEO1, CVIDEO2	Output port	Output port
CVIN1, CVIN2, SECAMIN, SVREF, SYNCIN, VERT, FSCIN	Input port	Input port
LP1, LP2, LP3, LP4	Output port	Output port

## 2.3.1 Software Reset

Writing "1" to bit 3 of the processor mode register 0 (address 000416) applies a (software) reset to the microcomputer. A software reset has almost the same effect as a hardware reset. The contents of internal RAM are preserved.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Processor mode register 0 (Note)	(0004 <sub>16</sub> )...	00 <sub>16</sub>	Display RAM address control register	(0202 <sub>16</sub> )...	00 <sub>16</sub>
Processor mode register 1	(0005 <sub>16</sub> )...	0 0 0 0 0 0 x x 0	Display RAM data control register	(0203 <sub>16</sub> )...	00 <sub>16</sub>
System clock control register 0	(0006 <sub>16</sub> )...	0 1 0 0 1 0 0 0	Font RAM address control register	(0204 <sub>16</sub> )...	00 <sub>16</sub>
System clock control register 1	(0007 <sub>16</sub> )...	0 0 1 0 0 0 0 0	Font RAM data control register	(0205 <sub>16</sub> )...	00 <sub>16</sub>
Chip select control register	(0008 <sub>16</sub> )...	0 0 0 0 0 0 0 1	SYRAM address control register	(0206 <sub>16</sub> )...	00 <sub>16</sub>
Address match interrupt enable register	(0009 <sub>16</sub> )...	x x x x x x x 0	SYRAM data control register	(0207 <sub>16</sub> )...	00 <sub>16</sub>
Protect register	(000A <sub>16</sub> )...	x x x x x x 0 0	Slice RAM address control register	(0208 <sub>16</sub> )...	00 <sub>16</sub>
Watchdog timer control register	(000F <sub>16</sub> )...	0 0 0 ? ? ? ? ?	Slice RAM data control register	(0209 <sub>16</sub> )...	00 <sub>16</sub>
Address match interrupt register 0	(0010 <sub>16</sub> )...	00 <sub>16</sub>	VBIRAM address control register	(020A <sub>16</sub> )...	00 <sub>16</sub>
	(0011 <sub>16</sub> )...	00 <sub>16</sub>	VBIRAM data control register	(020B <sub>16</sub> )...	00 <sub>16</sub>
	(0012 <sub>16</sub> )...	x x x x 0 0 0 0	Address control register for expansion register	(020C <sub>16</sub> )...	00 <sub>16</sub>
Address match interrupt register 1	(0014 <sub>16</sub> )...	00 <sub>16</sub>	Data control register for expansion register	(020D <sub>16</sub> )...	00 <sub>16</sub>
	(0015 <sub>16</sub> )...	00 <sub>16</sub>	Humming 8/4	(020E <sub>16</sub> )...	00 <sub>16</sub>
	(0016 <sub>16</sub> )...	x x x x 0 0 0 0	Humming 24/18	(020F <sub>16</sub> )...	00 <sub>16</sub>
DMA0 control register	(002C <sub>16</sub> )...	0 0 0 0 0 ? 0 0	Timer B3,4,5 count start flag	(0210 <sub>16</sub> )...	00 <sub>16</sub>
DMA1 control register	(003C <sub>16</sub> )...	0 0 0 0 0 ? 0 0	Reserved register	(0211 <sub>16</sub> )...	00 <sub>16</sub>
INT3 interrupt control register	(0044 <sub>16</sub> )...	x x 0 0 ? 0 0 0	Reserved register	(0212 <sub>16</sub> )...	00 <sub>16</sub>
Timer B5 interrupt control register	(0045 <sub>16</sub> )...	x x x x ? 0 0 0	Reserved register	(0213 <sub>16</sub> )...	00 <sub>16</sub>
Timer B4 interrupt control register	(0046 <sub>16</sub> )...	x x x x ? 0 0 0	Reserved register	(0214 <sub>16</sub> )...	00 <sub>16</sub>
Timer B3 interrupt control register	(0047 <sub>16</sub> )...	x x x x ? 0 0 0	Reserved register	(0215 <sub>16</sub> )...	00 <sub>16</sub>
SI/O4 interrupt control register	(0048 <sub>16</sub> )...	x x 0 0 ? 0 0 0	Timer B3 mode register	(0216 <sub>16</sub> )...	00 <sub>16</sub>
SI/O3 interrupt control register	(0049 <sub>16</sub> )...	x x 0 0 ? 0 0 0	Timer B4 mode register	(0217 <sub>16</sub> )...	00 <sub>16</sub>
Bus collision detection interrupt control register	(004A <sub>16</sub> )...	x x x x ? 0 0 0	Timer B5 mode register	(0218 <sub>16</sub> )...	00 <sub>16</sub>
DMA0 interrupt control register	(004B <sub>16</sub> )...	x x x x ? 0 0 0	Interrupt cause select register	(0219 <sub>16</sub> )...	00 <sub>16</sub>
DMA1 interrupt control register	(004C <sub>16</sub> )...	x x x x ? 0 0 0	SI/O3 control register	(021A <sub>16</sub> )...	00 <sub>16</sub>
Key input interrupt control register	(004D <sub>16</sub> )...	x x x x ? 0 0 0	SI/O4 control register	(021B <sub>16</sub> )...	00 <sub>16</sub>
A-D conversion interrupt control register	(004E <sub>16</sub> )...	x x x x ? 0 0 0	UART2 special mode register 2	(021C <sub>16</sub> )...	00 <sub>16</sub>
UART2 transmit interrupt control register	(004F <sub>16</sub> )...	x x x x ? 0 0 0	UART2 special mode register	(021D <sub>16</sub> )...	00 <sub>16</sub>
UART2 receive interrupt control register	(0050 <sub>16</sub> )...	x x x x ? 0 0 0	UART2 transmit/receive mode register	(021E <sub>16</sub> )...	00 <sub>16</sub>
UART0 transmit interrupt control register	(0051 <sub>16</sub> )...	x x x x ? 0 0 0	UART2 transmit/receive control register 0	(021F <sub>16</sub> )...	00 <sub>16</sub>
UART0 receive interrupt control register	(0052 <sub>16</sub> )...	x x x x ? 0 0 0	UART2 transmit/receive control register 1	(0340 <sub>16</sub> )...	0 0 0 x x x x x
UART1 transmit interrupt control register	(0053 <sub>16</sub> )...	x x x x ? 0 0 0		(0348 <sub>16</sub> )...	00 <sub>16</sub>
UART1 receive interrupt control register	(0054 <sub>16</sub> )...	x x x x ? 0 0 0		(0349 <sub>16</sub> )...	00 <sub>16</sub>
Timer A0 interrupt control register	(0055 <sub>16</sub> )...	x x x x ? 0 0 0		(034A <sub>16</sub> )...	00 <sub>16</sub>
Timer A1 interrupt control register	(0056 <sub>16</sub> )...	x x x x ? 0 0 0		(034B <sub>16</sub> )...	00 <sub>16</sub>
Timer A2 interrupt control register	(0057 <sub>16</sub> )...	x x x x ? 0 0 0		(035B <sub>16</sub> )...	0 0 ? x 0 0 0 0
Timer A3 interrupt control register	(0058 <sub>16</sub> )...	x x x x ? 0 0 0		(035C <sub>16</sub> )...	0 0 ? x 0 0 0 0
Timer A4 interrupt control register	(0059 <sub>16</sub> )...	x x x x ? 0 0 0		(035D <sub>16</sub> )...	0 0 ? x 0 0 0 0
Timer B0 interrupt control register	(005A <sub>16</sub> )...	x x x x ? 0 0 0		(035F <sub>16</sub> )...	00 <sub>16</sub>
Timer B1 interrupt control register	(005B <sub>16</sub> )...	x x x x ? 0 0 0		(0362 <sub>16</sub> )...	40 <sub>16</sub>
Timer B2 interrupt control register	(005C <sub>16</sub> )...	x x x x ? 0 0 0		(0366 <sub>16</sub> )...	40 <sub>16</sub>
INT0 interrupt control register	(005D <sub>16</sub> )...	x x 0 0 ? 0 0 0		(0376 <sub>16</sub> )...	00 <sub>16</sub>
INT1 interrupt control register	(005E <sub>16</sub> )...	x x 0 0 ? 0 0 0		(0377 <sub>16</sub> )...	00 <sub>16</sub>
INT2 interrupt control register	(005F <sub>16</sub> )...	x x 0 0 ? 0 0 0		(0378 <sub>16</sub> )...	00 <sub>16</sub>
				(037C <sub>16</sub> )...	0 0 0 0 1 0 0 0
				(037D <sub>16</sub> )...	0 0 0 0 0 0 1 0

x : Nothing is mapped to this bit  
? : Undefined

The content of other registers and RAM is undefined when the microcomputer is reset. The initial values must therefore be set.

Figure 2.3.3 Device's internal status after a reset is cleared

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Count start flag	(0380 <sub>16</sub> )...	00 <sub>16</sub>	D-A control register	(03DC <sub>16</sub> )...	00 <sub>16</sub>
Clock prescaler reset flag	(0381 <sub>16</sub> )...	0 x x x x x x x	Port P0 direction register	(03E2 <sub>16</sub> )...	00 <sub>16</sub>
One-shot start flag	(0382 <sub>16</sub> )...	0 0 x 0 0 0 0 0	Port P1 direction register	(03E3 <sub>16</sub> )...	00 <sub>16</sub>
Trigger select flag	(0383 <sub>16</sub> )...	00 <sub>16</sub>	Port P2 direction register	(03E6 <sub>16</sub> )...	00 <sub>16</sub>
Up-down flag	(0384 <sub>16</sub> )...	00 <sub>16</sub>	Port P3 direction register	(03E7 <sub>16</sub> )...	00 <sub>16</sub>
Timer A0 mode register	(0396 <sub>16</sub> )...	00 <sub>16</sub>	Port P4 direction register	(03EA <sub>16</sub> )...	00 <sub>16</sub>
Timer A1 mode register	(0397 <sub>16</sub> )...	00 <sub>16</sub>	Port P5 direction register	(03EB <sub>16</sub> )...	00 <sub>16</sub>
Timer A2 mode register	(0398 <sub>16</sub> )...	00 <sub>16</sub>	Port P6 direction register	(03EE <sub>16</sub> )...	00 <sub>16</sub>
Timer A3 mode register	(0399 <sub>16</sub> )...	00 <sub>16</sub>	Port P7 direction register	(03EF <sub>16</sub> )...	00 <sub>16</sub>
Timer A4 mode register	(039A <sub>16</sub> )...	00 <sub>16</sub>	Port P8 direction register	(03F2 <sub>16</sub> )...	0 0 x 0 0 0 0 0
Timer B0 mode register	(039B <sub>16</sub> )...	0 0 ? x 0 0 0 0	Port P9 direction register	(03F3 <sub>16</sub> )...	00 <sub>16</sub>
Timer B1 mode register	(039C <sub>16</sub> )...	0 0 ? x 0 0 0 0	Port P10 direction register	(03F6 <sub>16</sub> )...	00 <sub>16</sub>
Timer B2 mode register	(039D <sub>16</sub> )...	0 0 ? x 0 0 0 0	Pull-up control register 0	(03FC <sub>16</sub> )...	00 <sub>16</sub>
UART0 transmit/receive mode register	(03A0 <sub>16</sub> )...	00 <sub>16</sub>	Pull-up control register 1(Note)	(03FD <sub>16</sub> )...	00 <sub>16</sub>
UART0 transmit/receive control register 0	(03A4 <sub>16</sub> )...	0 0 0 0 1 0 0 0	Pull-up control register 2	(03FE <sub>16</sub> )...	00 <sub>16</sub>
UART0 transmit/receive control register 1	(03A5 <sub>16</sub> )...	0 0 0 0 0 0 1 0	Port control register	(03FF <sub>16</sub> )...	00 <sub>16</sub>
UART1 transmit/receive mode register	(03A8 <sub>16</sub> )...	00 <sub>16</sub>	Data registers (R0/R1/R2/R3)		0000 <sub>16</sub>
UART1 transmit/receive control register 0	(03AC <sub>16</sub> )...	0 0 0 0 1 0 0 0	Address registers (A0/A1)		0000 <sub>16</sub>
UART1 transmit/receive control register 1	(03AD <sub>16</sub> )...	0 0 0 0 0 0 1 0	Frame base register (FB)		0000 <sub>16</sub>
UART transmit/receive control register 2	(03B0 <sub>16</sub> )...	x 0 0 0 0 0 0 0	Interrupt table register (INTB)		00000 <sub>16</sub>
DMA0 cause select register	(03B8 <sub>16</sub> )...	00 <sub>16</sub>	User stack pointer (USP)		0000 <sub>16</sub>
DMA1 cause select register	(03BA <sub>16</sub> )...	00 <sub>16</sub>	Interrupt stack pointer (ISP)		0000 <sub>16</sub>
A-D control register 2	(03D4 <sub>16</sub> )...	0 0 0 0 x x x 0	Static base register (SB)		0000 <sub>16</sub>
A-D control register 0	(03D6 <sub>16</sub> )...	0 0 0 0 0 ? ? ?	Flag register (FLG)		0000 <sub>16</sub>
A-D control register 1	(03D7 <sub>16</sub> )...	00 <sub>16</sub>			

x : Nothing is mapped to this bit  
? : Undefined

The content of other registers and RAM is undefined when the microcomputer is reset. The initial values must therefore be set.

Note: When the VCC level is applied to the CNVSS pin, it is 02<sub>16</sub> at a reset.

**Figure 2.3.4 Device's internal status after a reset is cleared**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.4 Processor Mode

### (1) Types of Processor Mode

Processor mode can be used at microprocessor mode.

- **Microprocessor mode**

In microprocessor mode, the SFR, internal RAM, and external memory space can be accessed.

In this mode, some of the pins function as the address bus, the data bus, and as control signals. The number of pins assigned to these functions depends on the bus and register settings. (See “2.4.1 Bus Settings” for details.)

### (2) Setting Microprocessor Mode

Microprocessor mode is set using the CNVss pin and the processor mode bits (bits 1 and 0 at address 000416). Set the processor mode bits to “112”.

Regardless of the level of the CNVss pin, the processor mode bits can be changed by software. Therefore, never change the processor mode bits when changing the contents of other bits.

- **Applying Vcc to CNVss pin**

The microcomputer starts to operate in microprocessor mode after being reset.

Figure 2.4.1 shows the processor mode register 0 and 1.

Figure 2.4.2 shows the memory maps applicable for microprocessor mode.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Processor mode register 0 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
								PM0	0004 <sub>16</sub>	00 <sub>16</sub> (Note 2)
								Bit symbol	Bit name	Function
								PM00	Processor mode bit	b1 b0 0 0: Inhibited 0 1: Inhibited 1 0: Inhibited 1 1: Microprocessor mode
								PM01		
								PM02	R/W mode select bit	0: <u>RD</u> , <u>BHE</u> , <u>WR</u> 1: <u>RD</u> , <u>WRH</u> , <u>WRL</u>
								PM03	Software reset bit	The device is reset when this bit is set to "1". The value of this bit is "0" when read.
								PM04	Multiplexed bus space select bit	b5 b4 0 0: Multiplexed bus is not used 0 1: Allocated to CS2 space 1 0: Inhibited 1 1: Inhibited
								PM05		
								PM06	Port P40 to P43 function select bit	0: Address output 1: Port function (Address is not output)
								PM07	BCLK output disable bit	0: BCLK is output 1: BCLK is not output (Pin is left floating)

Notes 1: Set bit 1 of the protect register (address 000A<sub>16</sub>) to "1" when writing new values to this register.

2: If the Vcc voltage is applied to the CNVss, the value of this register when reset is 03<sub>16</sub>. (PM00 and PM01 both are set to "1".)

## Processor mode register 1 (Note)

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
	0	0	0	0	0	0	0	PM1	0005 <sub>16</sub>	00000XX0 <sub>2</sub>
								Bit symbol	Bit name	Function
								Reserved bit		Must always be set to "0"
								Nothing is assigned.		
								In an attempt to write to these bits, write "0". The value, if read, turns out to be indeterminate.		
								Reserved bit		Must always be set to "0"
								Reserved bit		Must always be set to "0"
								PM17	Wait bit	0: No wait state 1: Wait state inserted

Note: Set bit 1 of the protect register (address 000A<sub>16</sub>) to "1" when writing new values to this register.

Figure 2.4.1 Processor mode registers

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

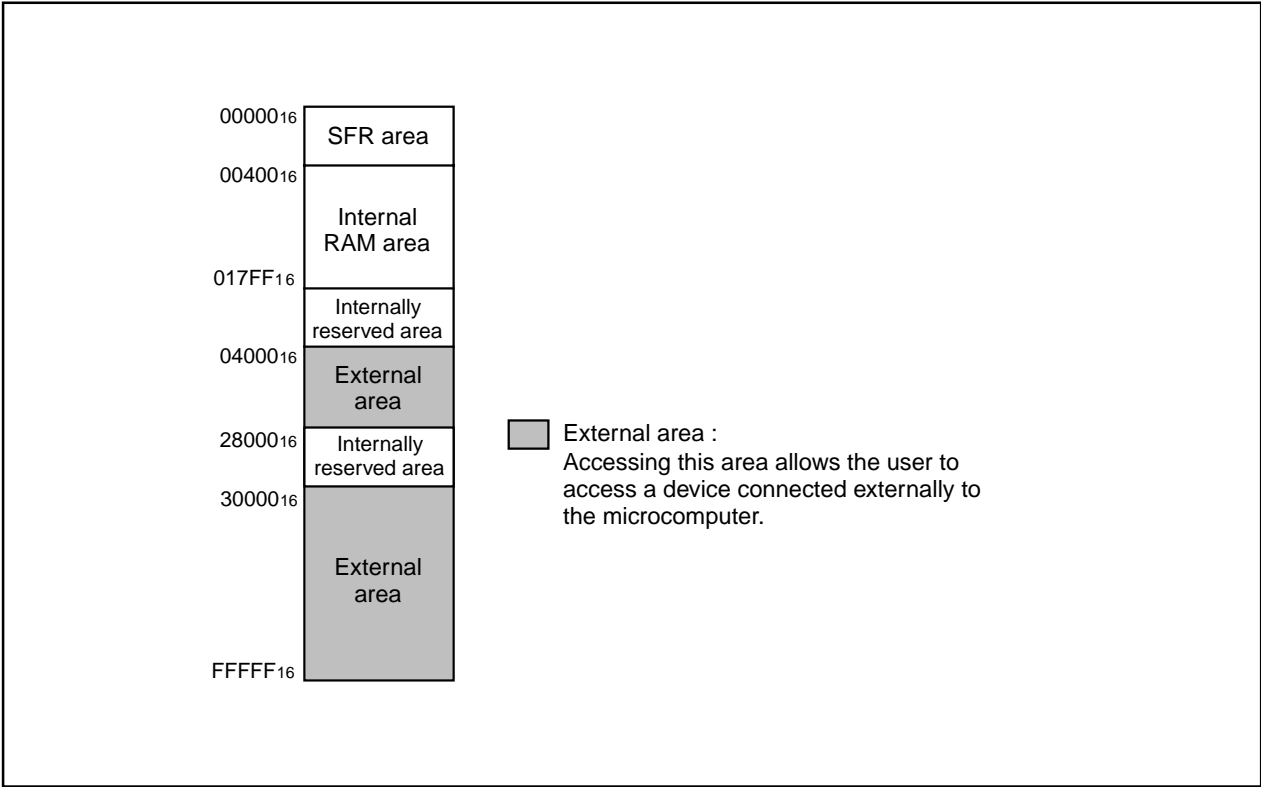


Figure 2.4.2 Memory maps applicable for microprocessor mode

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
 with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.4.1 Bus settings

The BYTE pin and bits 4 to 6 of the processor mode register 0 (address 000416) are used to change the bus settings. Table 2.4.1 shows the factors used to change the bus settings.

**Table 2.4.1 Factors for switching bus settings**

Bus setting	Switching factor
Switching external address bus width	Bit 6 of processor mode register 0
Switching external data bus width	BYTE pin
Switching between separate and multiplex bus	Bits 4 and 5 of processor mode register 0

### (1) Selecting external address bus width

The address bus width for external output in the 1M bytes of address space can be set to 16 bits (64K bytes address space) or 20 bits (1M bytes address space). When bit 6 of the processor mode register 0 is set to "1", the external address bus width is set to 16 bits, and P2 and P3 become part of the address bus. P40 to P43 can be used as programmable I/O ports. When bit 6 of processor mode register 0 is set to "0", the external address bus width is set to 20 bits, and P2, P3, and P40 to P43 become part of the address bus.

### (2) Selecting external data bus width

The external data bus width can be set to 8 or 16 bits. (Note, however, that only the separate bus can be set.) When the BYTE pin is "L", the bus width is set to 16 bits; when "H", it is set to 8 bits. (The internal bus width is permanently set to 16 bits.) While operating, fix the BYTE pin either to "H" or to "L".

### (3) Selecting separate/multiplex bus

The bus format can be set to multiplex or separate bus using bits 4 and 5 of the processor mode register 0.

#### • Separate bus

In this mode, the data and address are input and output separately. The data bus can be set using the BYTE pin to be 8 or 16 bits. When the BYTE pin is "H", the data bus is set to 8 bits and P0 functions as the data bus and P1 as a programmable I/O port. When the BYTE pin is "L", the data bus is set to 16 bits and P0 and P1 are both used for the data bus.

When the separate bus is used for access, a software wait can be selected.

#### • Multiplex bus

In this mode, data and address I/O are time multiplexed. With an 8-bit data bus selected (BYTE pin = "H"), the 8 bits from D0 to D7 are multiplexed with A0 to A7.

With a 16-bit data bus selected (BYTE pin = "L"), the 8 bits from D0 to D7 are multiplexed with A1 to A8. D8 to D15 are not multiplexed. In this case, the external devices connected to the multiplexed bus are mapped to the microcomputer's even addresses (every 2nd address). To access these external devices, access the even addresses as bytes.

The ALE signal latches the address. It is output from P56.

Before using the multiplex bus for access, be sure to insert a software wait.

The processor operates using the separate bus after reset is revoked, so the entire space multiplexed bus cannot be chosen.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Table 2.4.2 Pin functions for processor mode**

Processor mode	Microprocessor modes			
Multiplexed bus space select bit	"01" $\left[ \begin{array}{l} \text{CS2 is for multiplexed bus} \\ \text{and others are for separate} \\ \text{bus} \end{array} \right]$		"00" (separate bus)	
Data bus width BYTE pin level	8 bits "H"	16 bits "L"	8 bits "H"	16 bits "L"
P00 to P07	Data bus	Data bus	Data bus	Data bus
P10 to P17	I/O port	Data bus	I/O port	Data bus
P20	Address bus /data bus (Note)	Address bus	Address bus	Address bus
P21 to P27	Address bus data bus (Note)	Address bus data bus (Note)	Address bus	Address bus
P30	Address bus	Address bus data bus (Note)	Address bus	Address bus
P31 to P37	Address bus	Address bus	Address bus	Address bus
P40 to P43 Port P40 to P43 function select bit = 1	I/O port	I/O port	I/O port	I/O port
P40 to P43 Port P40 to P43 function select bit = 0	Address bus	Address bus	Address bus	Address bus
P44 to P47	$\overline{\text{CS}}$ (chip select) or programmable I/O port (For details, refer to "Bus control")			
P50 to P53	Outputs $\overline{\text{RD}}$ , $\overline{\text{WRL}}$ , $\overline{\text{WRH}}$ , and $\overline{\text{BCLK}}$ or $\overline{\text{RD}}$ , $\overline{\text{BHE}}$ , $\overline{\text{WR}}$ , and $\overline{\text{BCLK}}$ (For details, refer to "Bus control")			
P54	$\overline{\text{HLDA}}$	$\overline{\text{HLDA}}$	$\overline{\text{HLDA}}$	$\overline{\text{HLDA}}$
P55	$\overline{\text{HOLD}}$	$\overline{\text{HOLD}}$	$\overline{\text{HOLD}}$	$\overline{\text{HOLD}}$
P56	ALE	ALE	ALE	ALE
P57	$\overline{\text{RDY}}$	$\overline{\text{RDY}}$	$\overline{\text{RDY}}$	$\overline{\text{RDY}}$

Note : Address bus when in separate bus mode.



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.4.2 Bus Control

The following explains the signals required for accessing external devices and software waits.

### (1) Address bus/data bus

The address bus consists of the 20 pins A0 to A19 for accessing the 1M bytes of address space.

The data bus consists of the pins for data I/O. When the BYTE pin is "H", the 8 ports D0 to D7 function as the data bus. When BYTE is "L", the 16 ports D0 to D15 function as the data bus.

### (2) Chip select signal

The chip select signal is output using the same pins as P44, P46 and to P47. Bits 0, 2 and 3 of the chip select control register (address 0008<sub>16</sub>) set each pin to function as a port or to output the chip select signal.

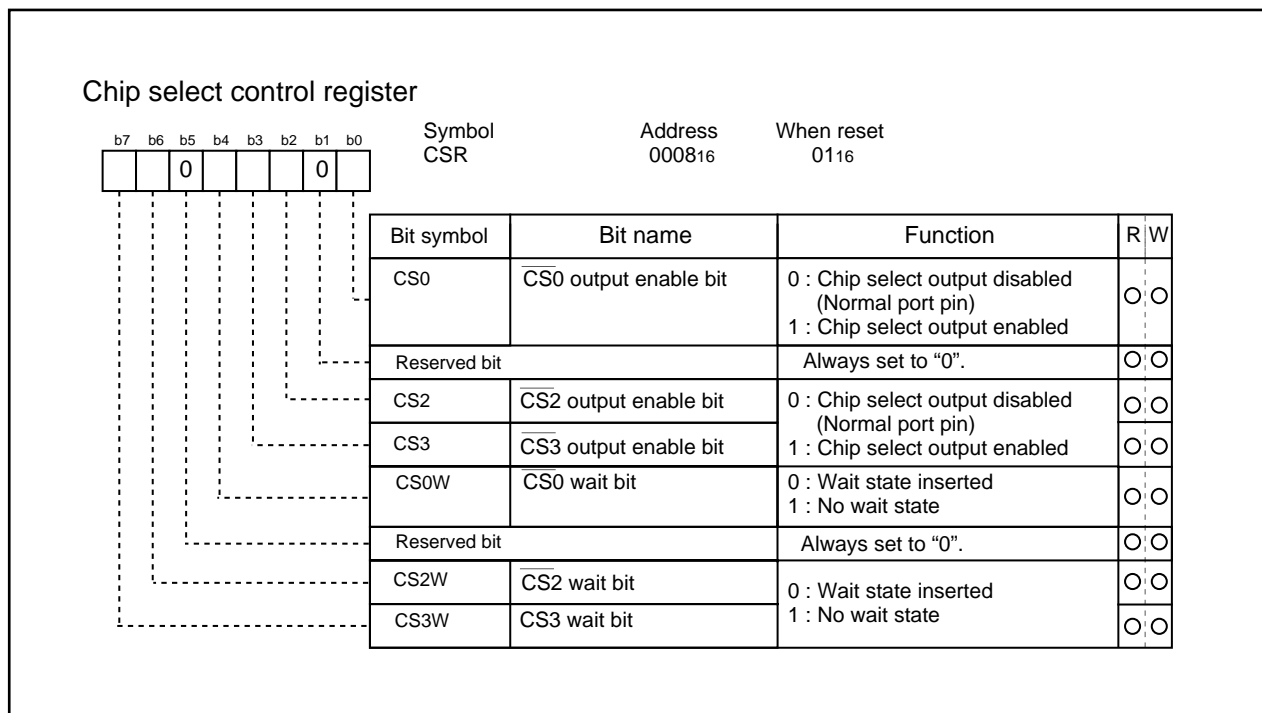
In microprocessor mode, only  $\overline{CS0}$  outputs the chip select signal after the reset state has been cancelled.  $\overline{CS2}$ ,  $\overline{CS3}$  function as input ports. Figure 2.4.3 shows the chip select control register.

The chip select signal can be used to split the external area. Tables 2.4.3 show the external memory areas specified using the chip select signal.

**Table 2.4.3 External areas specified by the chip select signals**

Processor mode	Chip select signal		
	$\overline{CS0}$	$\overline{CS2}$	$\overline{CS3}$
Microprocessor mode	30000 <sub>16</sub> to FFFFF <sub>16</sub> (832K bytes)	08000 <sub>16</sub> to 27FFF <sub>16</sub> (128K bytes)	04000 <sub>16</sub> to 07FFF <sub>16</sub> (16K bytes)

Note : Address 28000<sub>16</sub> to 2FFFF<sub>16</sub> are reserved and cannot be used.



**Figure 2.4.3 Chip select control register**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

### (3) Read/write signals

With a 16-bit data bus (BYTE pin = "L"), bit 2 of the processor mode register 0 (address 0004<sub>16</sub>) select the combinations of  $\overline{RD}$ ,  $\overline{BHE}$ , and  $\overline{WR}$  signals or  $\overline{RD}$ ,  $\overline{WRL}$ , and  $\overline{WRH}$  signals. With an 8-bit data bus (BYTE pin = "H"), use the combination of  $\overline{RD}$ ,  $\overline{WR}$ , and  $\overline{BHE}$  signals. (Set bit 2 of the processor mode register 0 (address 0004<sub>16</sub>) to "0".) Tables 2.4.4 and 2.4.5 show the operation of these signals.

After a reset has been cancelled, the combination of  $\overline{RD}$ ,  $\overline{WR}$ , and  $\overline{BHE}$  signals is automatically selected. When switching to the  $\overline{RD}$ ,  $\overline{WRL}$ , and  $\overline{WRH}$  combination, do not write to external memory until bit 2 of the processor mode register 0 (address 0004<sub>16</sub>) has been set (Note).

Note: Before attempting to change the contents of the processor mode register 0, set bit 1 of the protect register (address 000A<sub>16</sub>) to "1".

**Table 2.4.4 Operation of  $\overline{RD}$ ,  $\overline{WRL}$ , and  $\overline{WRH}$  signals**

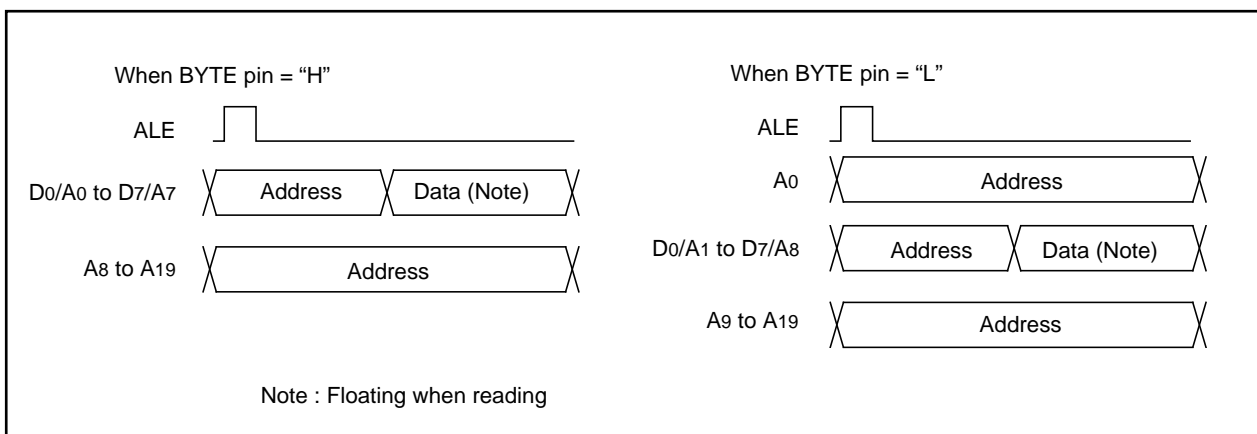
Data bus width	$\overline{RD}$	$\overline{WRL}$	$\overline{WRH}$	Status of external data bus
16-bit (BYTE = "L")	L	H	H	Read data
	H	L	H	Write 1 byte of data to even address
	H	H	L	Write 1 byte of data to odd address
	H	L	L	Write data to both even and odd addresses

**Table 2.4.5 Operation of  $\overline{RD}$ ,  $\overline{WR}$ , and  $\overline{BHE}$  signals**

Data bus width	$\overline{RD}$	$\overline{WR}$	$\overline{BHE}$	A0	Status of external data bus
16-bit (BYTE = "L")	H	L	L	H	Write 1 byte of data to odd address
	L	H	L	H	Read 1 byte of data from odd address
	H	L	H	L	Write 1 byte of data to even address
	L	H	H	L	Read 1 byte of data from even address
	H	L	L	L	Write data to both even and odd addresses
	L	H	L	L	Read data from both even and odd addresses
8-bit (BYTE = "H")	H	L	Not used	H / L	Write 1 byte of data
	L	H	Not used	H / L	Read 1 byte of data

### (4) ALE signal

The ALE signal latches the address when accessing the multiplex bus space. Latch the address when the ALE signal falls.



**Figure 2.4.4 ALE signal and address/data bus**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (5) The $\overline{\text{RDY}}$ signal

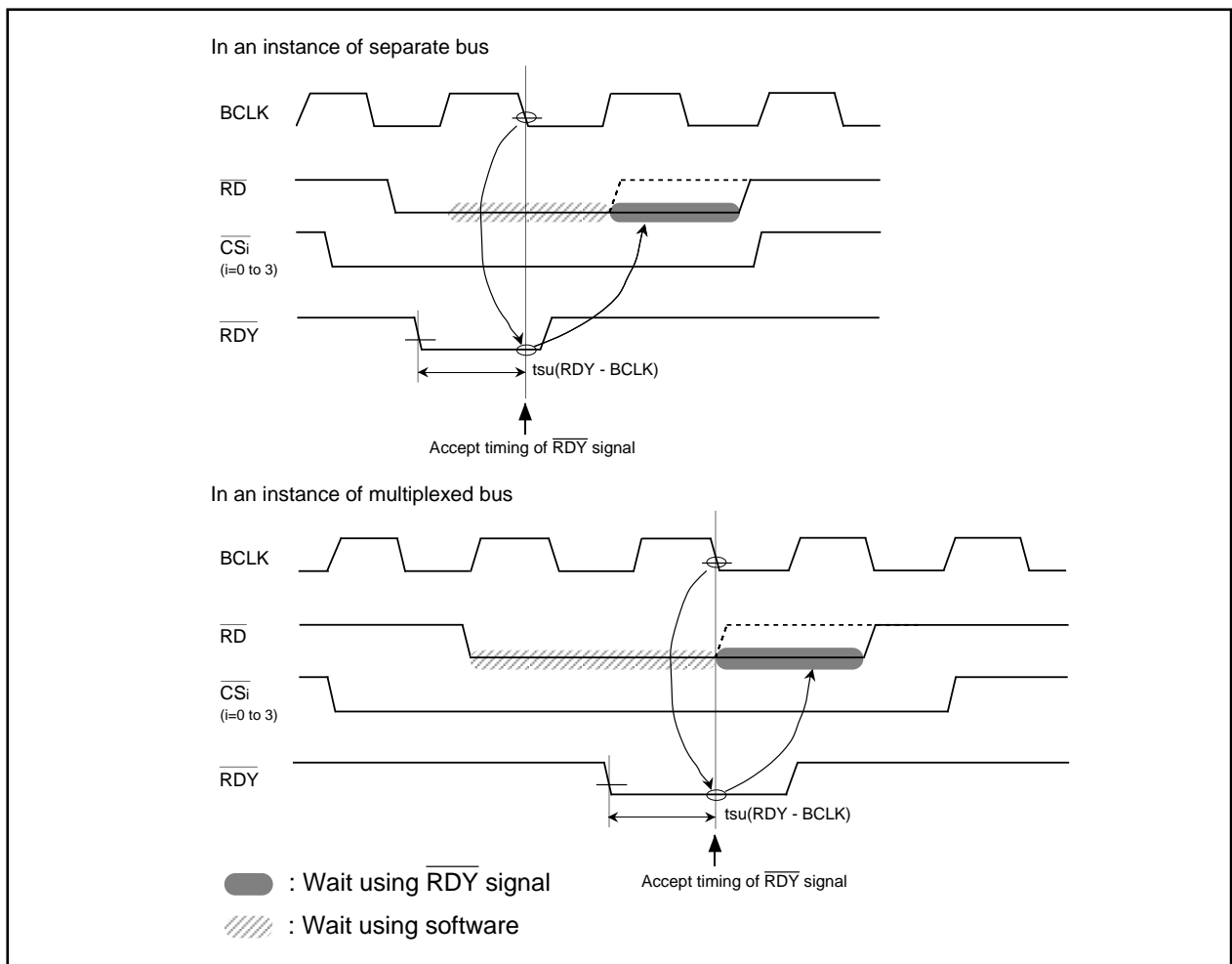
$\overline{\text{RDY}}$  is a signal that facilitates access to an external device that requires long access time. As shown in Figure 2.4.5, if an “L” is being input to the  $\overline{\text{RDY}}$  at the BCLK falling edge, the bus turns to the wait state. If an “H” is being input to the  $\overline{\text{RDY}}$  pin at the BCLK falling edge, the bus cancels the wait state. Table 2.4.6 shows the state of the microcomputer with the bus in the wait state, and Figure 2.4.5 shows an example in which the  $\overline{\text{RD}}$  signal is prolonged by the  $\overline{\text{RDY}}$  signal.

The  $\overline{\text{RDY}}$  signal is valid when accessing the external area during the bus cycle in which bits 4, 6 and 7 of the chip select control register (address 000816) are set to “0”. The  $\overline{\text{RDY}}$  signal is invalid when setting “1” to all bits 4, 6 and 7 of the chip select control register (address 000816), but the  $\overline{\text{RDY}}$  pin should be treated as properly as in non-using.

**Table 2.4.6 Microcomputer status in ready state (Note)**

Item	Status
Oscillation	On
$\overline{\text{R/W}}$ signal, address bus, data bus, $\overline{\text{CS}}$ ALE signal, HLDA, programmable I/O ports	Maintain status when $\overline{\text{RDY}}$ signal received
Internal peripheral circuits	On

Note: The  $\overline{\text{RDY}}$  signal cannot be received immediately prior to a software wait.



**Figure 2.4.5 Example of  $\overline{\text{RD}}$  signal extended by  $\overline{\text{RDY}}$  signal**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (6) Hold signal

The hold signal is used to transfer the bus privileges from the CPU to the external circuits. Inputting "L" to the HOLD pin places the microcomputer in the hold state at the end of the current bus access. This status is maintained and "L" is output from the HLDA pin as long as "L" is input to the HOLD pin. Table 2.4.7 shows the microcomputer status in the hold state.

Bus-using priorities are given to  $\overline{\text{HOLD}}$ , DMAC, and CPU in order of decreasing precedence.

$\overline{\text{HOLD}} > \text{DMAC} > \text{CPU}$

**Figure 2.4.6 Bus-using priorities**

**Table 2.4.7 Microcomputer status in hold state**

Item		Status
Oscillation		ON
R/W signal, address bus, data bus, $\overline{\text{CS}}$ , $\overline{\text{BHE}}$		Floating
Programmable I/O ports	P0, P1, P2, P3, P4, P5	Floating
	P6, P7, P8, P9, P10	Maintains status when hold signal is received
HLDA		Output "L"
Internal peripheral circuits		ON (but watchdog timer stops)
ALE signal		Undefined

## (7) External bus status when the internal area is accessed

Table 2.4.8 shows the external bus status when the internal area is accessed.

**Table 2.4.8 External bus status when the internal area is accessed**

Item		SFR accessed	Internal RAM accessed
Address bus		Address output	Maintain status before accessed address of external area
Data bus	When read	Floating	Floating
	When write	Output data	Undefined
$\overline{\text{RD}}$ , $\overline{\text{WR}}$ , $\overline{\text{WRL}}$ , $\overline{\text{WRH}}$		$\overline{\text{RD}}$ , $\overline{\text{WR}}$ , $\overline{\text{WRL}}$ , $\overline{\text{WRH}}$ output	Output "H"
$\overline{\text{BHE}}$		$\overline{\text{BHE}}$ output	Maintain status before accessed status of external area
$\overline{\text{CS}}$		Output "H"	Output "H"
ALE		Output "L"	Output "L"

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (8) BCLK output

The user can choose the BCLK output by use of bit 7 of processor mode register 0 (0004<sub>16</sub>) (Note). When set to "1", the output floating.

Note: Before attempting to change the contents of the processor mode register 0, set bit 1 of the protectregister (address 000A<sub>16</sub>) to "1".

## (9) Software wait

A software wait can be inserted by setting the wait bit (bit 7) of the processor mode register 1 (address 0005<sub>16</sub>) (Note) and bits 4, 6 and 7 of the chip select control register (address 0008<sub>16</sub>).

A software wait is inserted in the internal RAM area and in the external memory area by setting the wait bit of the processor mode register 1. When set to "0", each bus cycle is executed in one BCLK cycle. When set to "1", each bus cycle is executed in two or three BCLK cycles. After the microcomputer has been reset, this bit defaults to "0". When set to "1", a wait is applied to all memory areas (two or three BCLK cycles), regardless of the contents of bits 4, 6 and 7 of the chip select control register. Set this bit after referring to the recommended operating conditions (main clock input oscillation frequency) of the electric characteristics. However, when the user is using the  $\overline{\text{RDY}}$  signal, the relevant bit in the chip select control register's bits 4, 6 and 7 must be set to "0".

When the wait bit of the processor mode register 1 is "0", software waits can be set independently for each areas selected using the chip select signal. Bits 4, 6 and 7 of the chip select control register correspond to chip selects  $\overline{\text{CS}}_0$ ,  $\overline{\text{CS}}_2$ , and  $\overline{\text{CS}}_3$ . When one of these bits is set to "1", the bus cycle is executed in one BCLK cycle. When set to "0", the bus cycle is executed in two or three BCLK cycles. These bits default to "0" after the microcomputer has been reset.

The SFR area is always accessed in two BCLK cycles regardless of the setting of these control bits. Also, insert a software wait if using the multiplex bus to access the external memory area.

Table 2.4.9 shows the software wait and bus cycles. Figure 2.4.7 shows example bus timing when using software waits.

Note: Before attempting to change the contents of the processor mode register 1, set bit 1 of the protect register (address 000A<sub>16</sub>) to "1".

**Table 2.4.9 Software waits and bus cycles**

Area	Bus status	Wait bit	Bits 4, 6 and 7 of chip select control register	Bus cycle
SFR	———	Invalid	Invalid	2 BCLK cycles
Internal RAM	———	0	Invalid	1 BCLK cycle
	———	1	Invalid	2 BCLK cycles
External memory area	Separate bus	0	1	1 BCLK cycle
	Separate bus	0	0	2 BCLK cycles
	Separate bus	1	0 (Note)	2 BCLK cycles
	Multiplex bus	0	0	3 BCLK cycles
	Multiplex bus	1	0 (Note)	3 BCLK cycles

Note: When using the RDY signal, always set to "0".

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

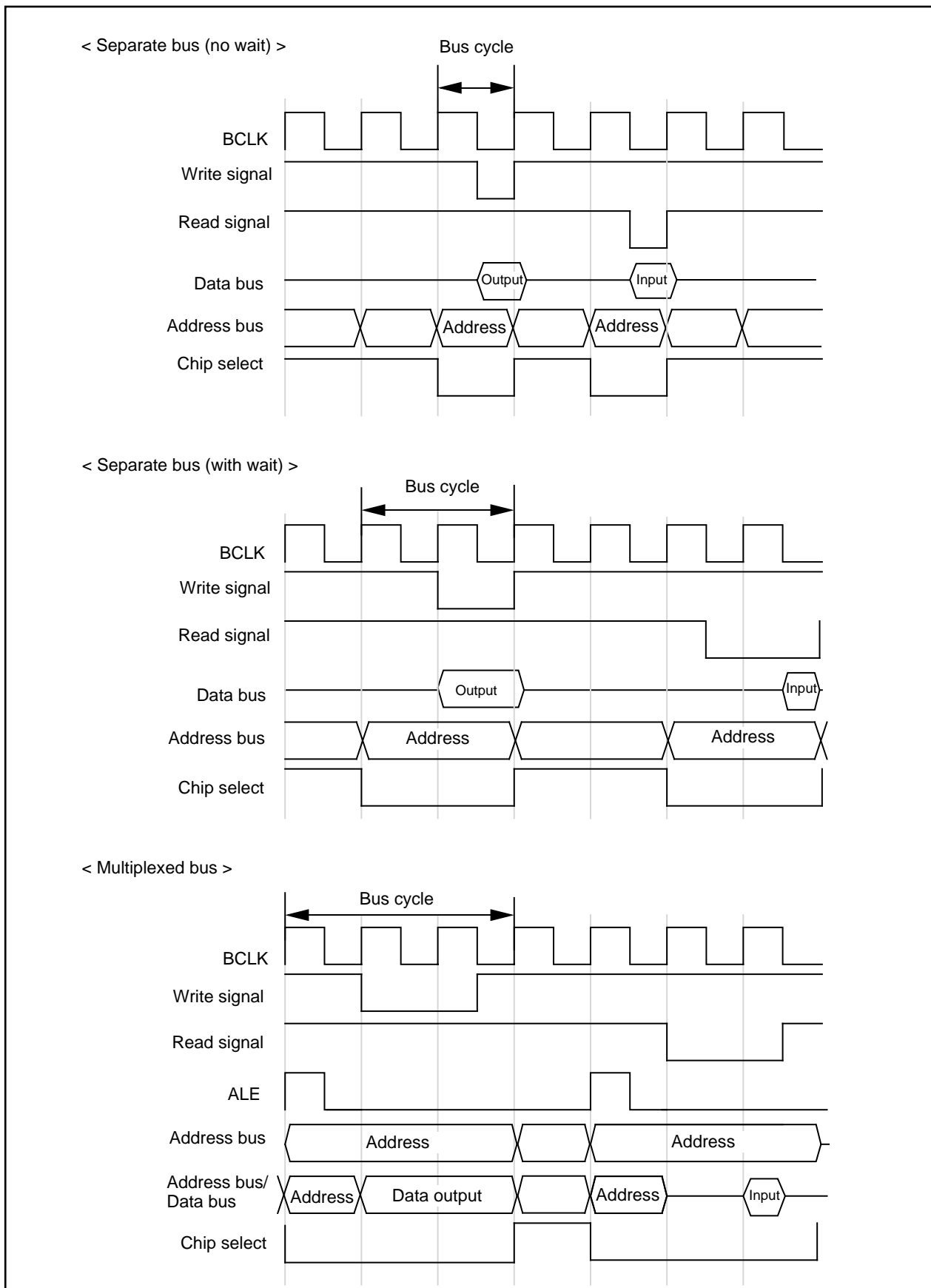


Figure 2.4.7 Typical bus timings using software wait

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.5 Clock Generating Circuit

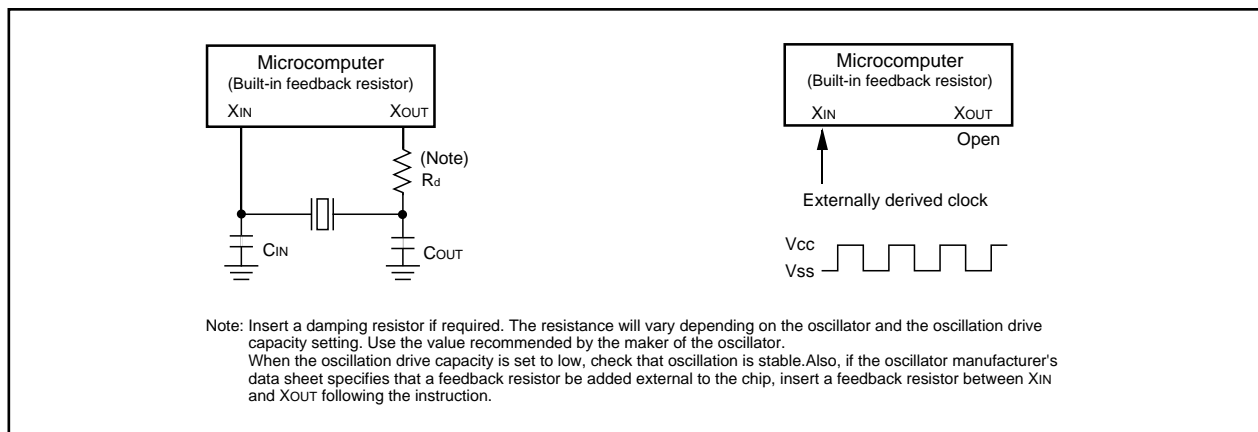
The clock generating circuit contains two oscillator circuits that supply the operating clock sources to the CPU and internal peripheral units.

**Table 2.5.1 Main clock and sub clock generating circuits**

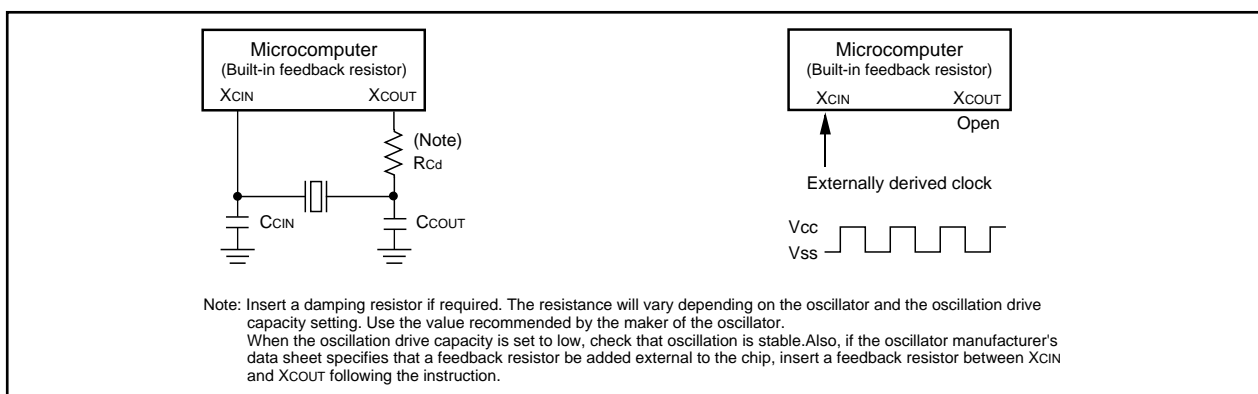
	Main clock generating circuit	Sub clock generating circuit
Use of clock	<ul style="list-style-type: none"> <li>CPU's operating clock source</li> <li>Internal peripheral units' operating clock source</li> </ul>	<ul style="list-style-type: none"> <li>CPU's operating clock source</li> <li>Timer A/B's count clock source</li> </ul>
Usable oscillator	Ceramic or crystal oscillator	Crystal oscillator
Pins to connect oscillator	XIN, XOUT	XCIN, XCOUT
Oscillation stop/restart function	Available	Available
Oscillator status immediately after reset	Oscillating	Stopped
Other	Externally derived clock can be input	

### 2.5.1 Example of oscillator circuit

Figure 2.5.1 shows some examples of the main clock circuit, one using an oscillator connected to the circuit, and the other one using an externally derived clock for input. Figure 2.5.2 shows some examples of sub clock circuits, one using an oscillator connected to the circuit, and the other one using an externally derived clock for input. Circuit constants in Figures 2.5.1 and 2.5.2 vary with each oscillator used. Use the values recommended by the manufacturer of your oscillator.



**Figure 2.5.1 Examples of main clock**



**Figure 2.5.2 Examples of sub clock**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.5.2 Clock Control

Figure 2.5.3 shows the block diagram of the clock generating circuit.

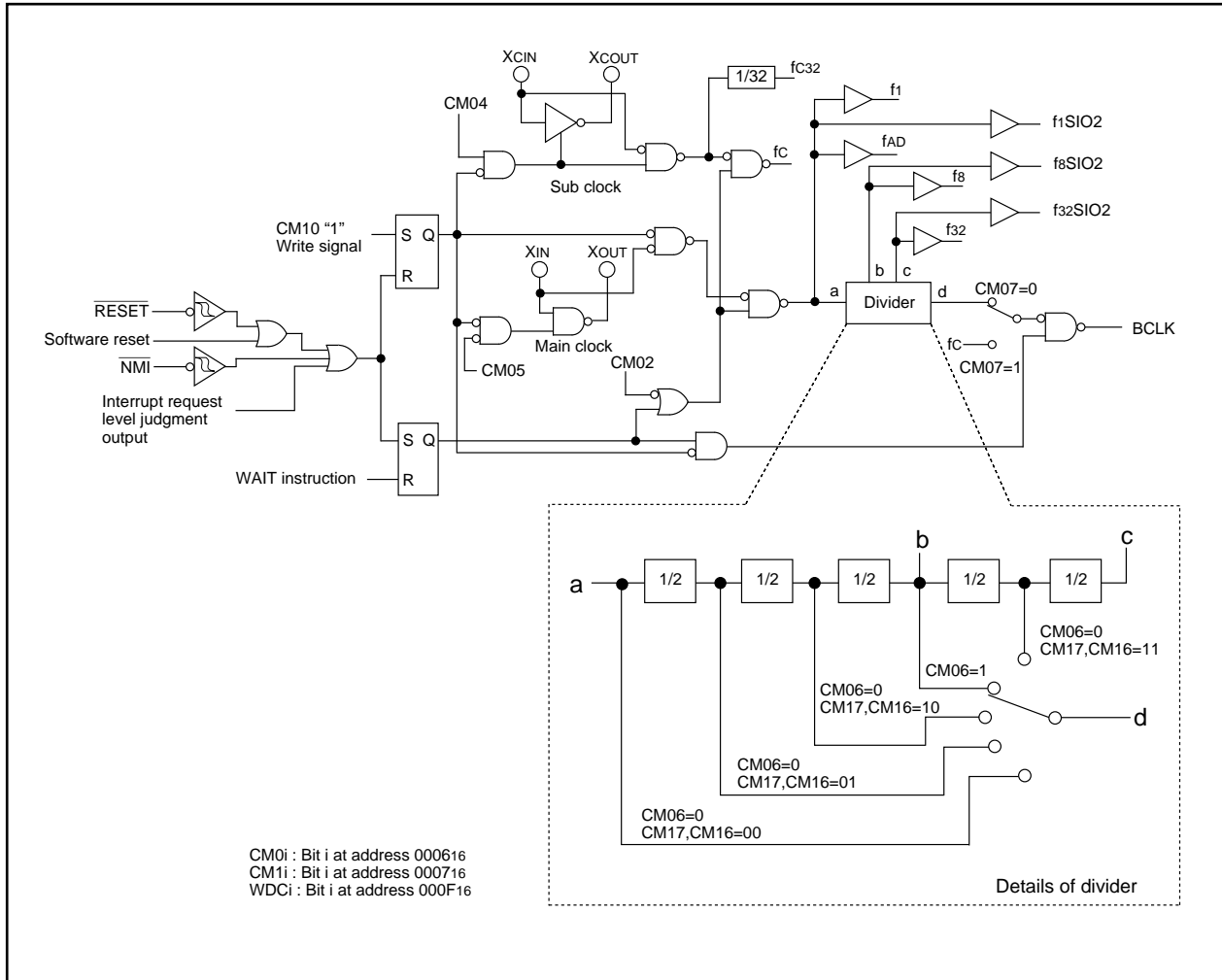


Figure 2.5.3 Clock generating circuit



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

The following paragraphs describes the clocks generated by the clock generating circuit.

## (1) Main clock

The main clock is generated by the main clock oscillation circuit. After a reset, the clock is divided by 8 to the BCLK. The clock can be stopped using the main clock stop bit (bit 5 at address 0006<sub>16</sub>). Stopping the clock, after switching the operating clock source of CPU to the sub-clock, reduces the power dissipation. After the oscillation of the main clock oscillation circuit has stabilized, the drive capacity of the main clock oscillation circuit can be reduced using the XIN-XOUT drive capacity select bit (bit 5 at address 0007<sub>16</sub>). Reducing the drive capacity of the main clock oscillation circuit reduces the power dissipation. This bit changes to "1" when shifting from high-speed/medium-speed mode to stop mode and at a reset. When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained.

## (2) Sub-clock

The sub-clock is generated by the sub-clock oscillation circuit. No sub-clock is generated after a reset. After oscillation is started using the port Xc select bit (bit 4 at address 0006<sub>16</sub>), the sub-clock can be selected as the BCLK by using the system clock select bit (bit 7 at address 0006<sub>16</sub>). However, be sure that the sub-clock oscillation has fully stabilized before switching.

After the oscillation of the sub-clock oscillation circuit has stabilized, the drive capacity of the sub-clock oscillation circuit can be reduced using the XCIN-XCOUT drive capacity select bit (bit 3 at address 0006<sub>16</sub>). Reducing the drive capacity of the sub-clock oscillation circuit reduces the power dissipation. This bit changes to "1" when shifting to stop mode and at a reset.

## (3) BCLK

The BCLK is the clock that drives the CPU, and is fc or the clock is derived by dividing the main clock by 1, 2, 4, 8, or 16. The BCLK is derived by dividing the main clock by 8 after a reset. The BCLK signal can be output from BCLK pin by the BCLK output disable bit (bit 7 at address 0004<sub>16</sub>).

The main clock division select bit 0 (bit 6 at address 0006<sub>16</sub>) changes to "1" when shifting from high-speed/medium-speed to stop mode and at reset. When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained.

## (4) Peripheral function clock(f<sub>1</sub>, f<sub>8</sub>, f<sub>32</sub>, f<sub>1SIO2</sub>, f<sub>8SIO2</sub>, f<sub>32SIO2</sub>, f<sub>AD</sub>)

The clock for the peripheral devices is derived from the main clock or by dividing it by 1, 8, or 32. The peripheral function clock is stopped by stopping the main clock or by setting the WAIT peripheral function clock stop bit (bit 2 at 0006<sub>16</sub>) to "1" and then executing a WAIT instruction.

## (5) fc<sub>32</sub>

This clock is derived by dividing the sub-clock by 32. It is used for the timer A and timer B counts.

## (6) fc

This clock has the same frequency as the sub-clock. It is used for the BCLK and for the watchdog timer.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Figure 2.5.4 shows the system clock control registers 0 and 1.

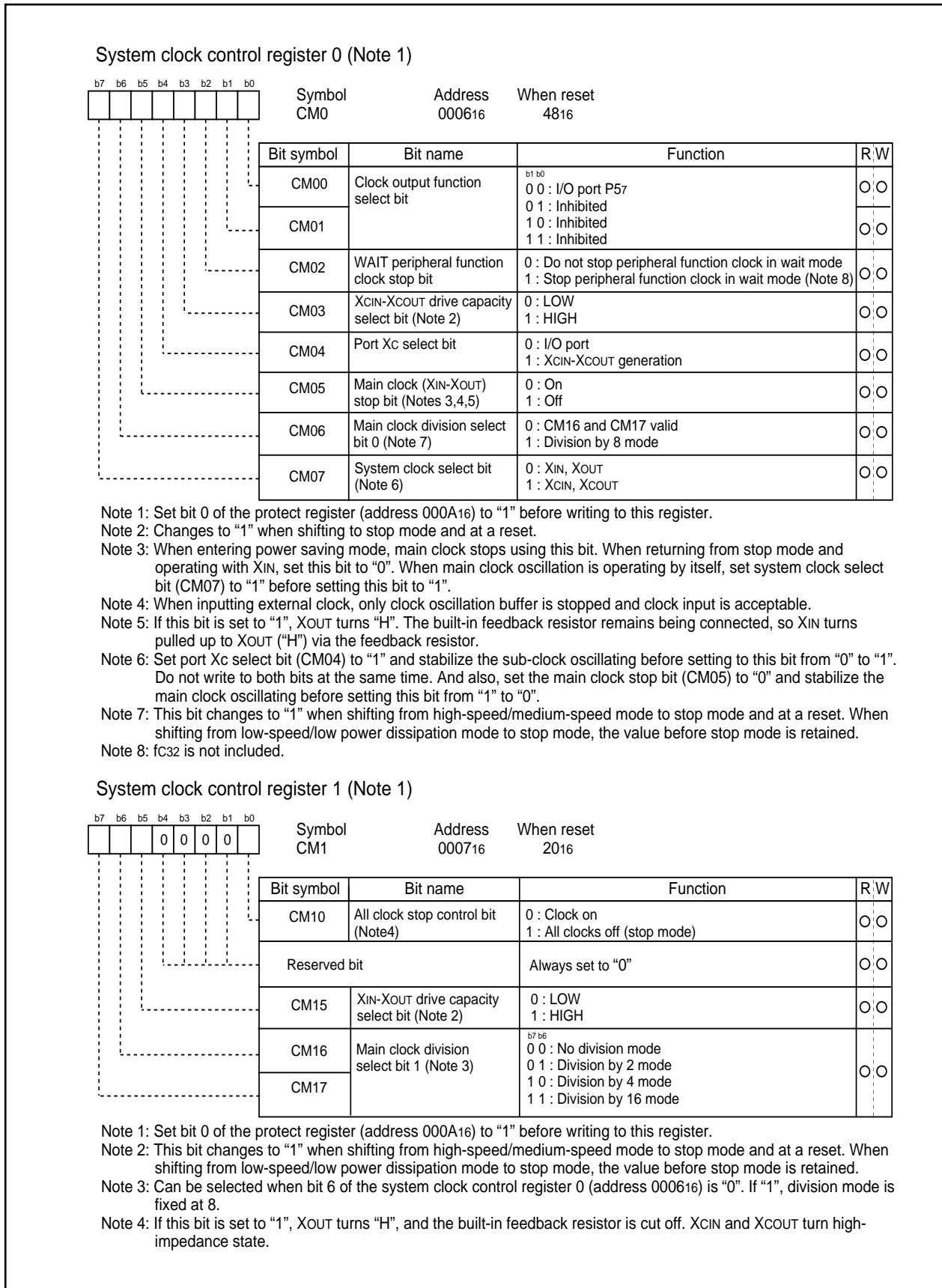


Figure 2.5.4 Clock control registers 0 and 1

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.5.3 Stop Mode

Writing "1" to the main clock and sub-clock stop control bit (bit 0 at address 000716) stops oscillation and the microcomputer enters stop mode. In stop mode, the content of the internal RAM is retained provided that VCC remains above 2V.

The internal oscillator circuit of expansion function (OSD function/ data slice function/ data encode function/ humming function) stops oscillation when expansion register CK\_VCO, XTAL\_VCO, PDC\_VCO\_ON, VPS\_VCO\_ON = "L".

Because the oscillation, BCLK, f1 to f32, f1SIO2 to f32SIO2, fc, fc32, and fAD stops in stop mode, peripheral functions such as the A-D converter and watchdog timer do not function. However, timer A and timer B operate provided that the event counter mode is set to an external pulse, and UARTi(i = 0 to 2) SI/O3,4 functions provided an external clock is selected. Table 2.5.2 shows the status of the ports in stop mode. Stop mode is cancelled by a hardware reset or interrupt. If an interrupt is to be used to cancel stop mode, that interrupt must first have been enabled. If returning by an interrupt, that interrupt routine is executed. When shifting from high-speed/medium-speed mode to stop mode and at a reset, the main clock division select bit 0 (bit 6 at address 000616) is set to "1". When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained.

**Table 2.5.2 Port status during stop mode**

Pin	Microprocessor mode
Address bus, data bus, $\overline{\text{CS0}}$ , $\overline{\text{CS2}}$ , $\overline{\text{CS3}}$	Retains status before stop mode
$\overline{\text{RD}}$ , $\overline{\text{WR}}$ , $\overline{\text{BHE}}$ , $\overline{\text{WRL}}$ , $\overline{\text{WRH}}$	"H"
$\overline{\text{HLDA}}$ , BCLK	"H"
ALE	"H"
Port	Retains status before stop mode

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.5.4 Wait Mode

When a WAIT instruction is executed, the BCLK stops and the microcomputer enters the wait mode. In this mode, oscillation continues but the BCLK and watchdog timer stop. Writing "1" to the WAIT peripheral function clock stop bit and executing a WAIT instruction stops the clock being supplied to the internal peripheral functions, allowing power dissipation to be reduced. Table 2.5.3 shows the status of the ports in wait mode.

Wait mode is cancelled by a hardware reset or an interrupt. If an interrupt is used to cancel wait mode, the microcomputer restarts from the interrupt routine using as BCLK, the clock that had been selected when the WAIT instruction was executed.

**Table 2.5.3 Port status during wait mode**

Pin	Microprocessor mode
Address bus, data bus, CS0, CS2, CS3	Retains status before stop mode
RD, WR, BHE, WRL, WRH	"H"
HLDA, BCLK	"H"
ALE	"H"
Port	Retains status before stop mode

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.5.5 Status Transition Of BCLK

Power dissipation can be reduced and low-voltage operation achieved by changing the count source for BCLK. Table 2.5.4 shows the operating modes corresponding to the settings of system clock control registers 0 and 1.

When reset, the device starts in division by 8 mode. The main clock division select bit 0 (bit 6 at address 000616) changes to "1" when shifting from high-speed/medium-speed to stop mode and at a reset. When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained. The following shows the operational modes of BCLK.

### (1) Division by 2 mode

The main clock is divided by 2 to obtain the BCLK.

### (2) Division by 4 mode

The main clock is divided by 4 to obtain the BCLK.

### (3) Division by 8 mode

The main clock is divided by 8 to obtain the BCLK. When reset, the device starts operating from this mode. Before the user can go from this mode to no division mode, division by 2 mode, or division by 4 mode, the main clock must be oscillating stably. When going to low-speed or lower power consumption mode, make sure the sub-clock is oscillating stably.

### (4) Division by 16 mode

The main clock is divided by 16 to obtain the BCLK.

### (5) No-division mode

The main clock is divided by 1 to obtain the BCLK.

### (6) Low-speed mode

fc is used as the BCLK. Note that oscillation of both the main and sub clocks must have stabilized before transferring from this mode to another or vice versa. At least 2 to 3 seconds are required after the sub clock starts. Therefore, the program must be written to wait until this clock has stabilized immediately after powering up and after stop mode is cancelled.

### (7) Low power dissipation mode

fc is the BCLK and the main clock is stopped.

Note :

Before the count source for BCLK can be changed from XIN to XCIN or vice versa, the clock to which the count source is going to be switched must be oscillating stably. Allow a wait time in software for the oscillation to stabilize before switching over the clock.

**Table 2.5.4 Operating modes dictated by settings of system clock control registers 0 and 1**

CM17	CM16	CM07	CM06	CM05	CM04	Operating mode of BCLK
0	1	0	0	0	Invalid	Division by 2 mode
1	0	0	0	0	Invalid	Division by 4 mode
Invalid	Invalid	0	1	0	Invalid	Division by 8 mode
1	1	0	0	0	Invalid	Division by 16 mode
0	0	0	0	0	Invalid	No-division mode
Invalid	Invalid	1	Invalid	0	1	Low-speed mode
Invalid	Invalid	1	Invalid	1	1	Low power dissipation mode

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.5.6 Power control

The following is a description of the three available power control modes:

### Modes

Power control is available in three modes.

#### (a) Normal operation mode

- **High-speed mode**

Divide-by-1 frequency of the main clock becomes the BCLK. The CPU operates with the internal clock selected. Each peripheral function operates according to its assigned clock.

- **Medium-speed mode**

Divide-by-2, divide-by-4, divide-by-8, or divide-by-16 frequency of the main clock becomes the BCLK. The CPU operates according to the internal clock selected. Each peripheral function operates according to its assigned clock.

- **Low-speed mode**

fc becomes the BCLK. The CPU operates according to the fc clock. The fc clock is supplied by the secondary clock. Each peripheral function operates according to its assigned clock.

- **Low power consumption mode**

The main clock operating in low-speed mode is stopped. The CPU operates according to the fc clock. The fc clock is supplied by the secondary clock. The only peripheral functions that operate are those with the sub-clock selected as the count source.

#### (b) Wait mode

The CPU operation is stopped. The oscillators do not stop.

#### (c) Stop mode

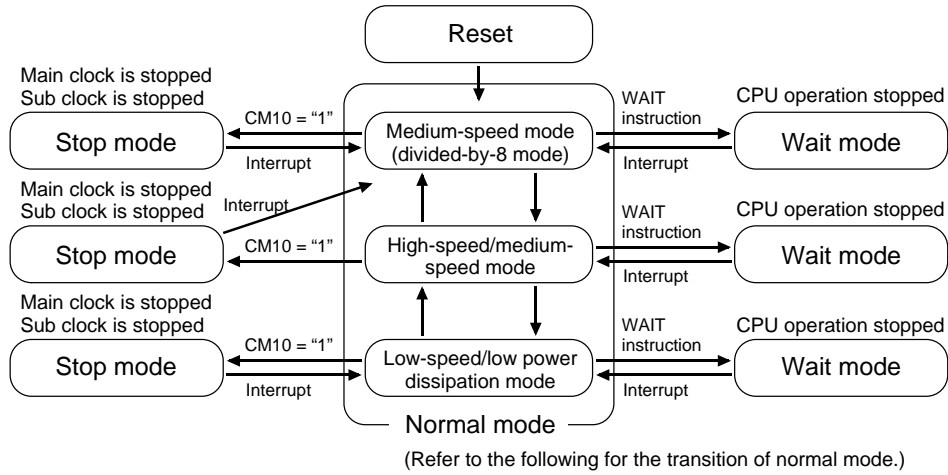
The main clock and the sub-clock oscillators stop. The CPU and all built-in peripheral functions stop. This mode, among the three modes listed here, is the most effective in decreasing power consumption.

Figure 2.5.5 is the state transition diagram of the above modes.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Transition of stop mode, wait mode



## Transition of normal mode

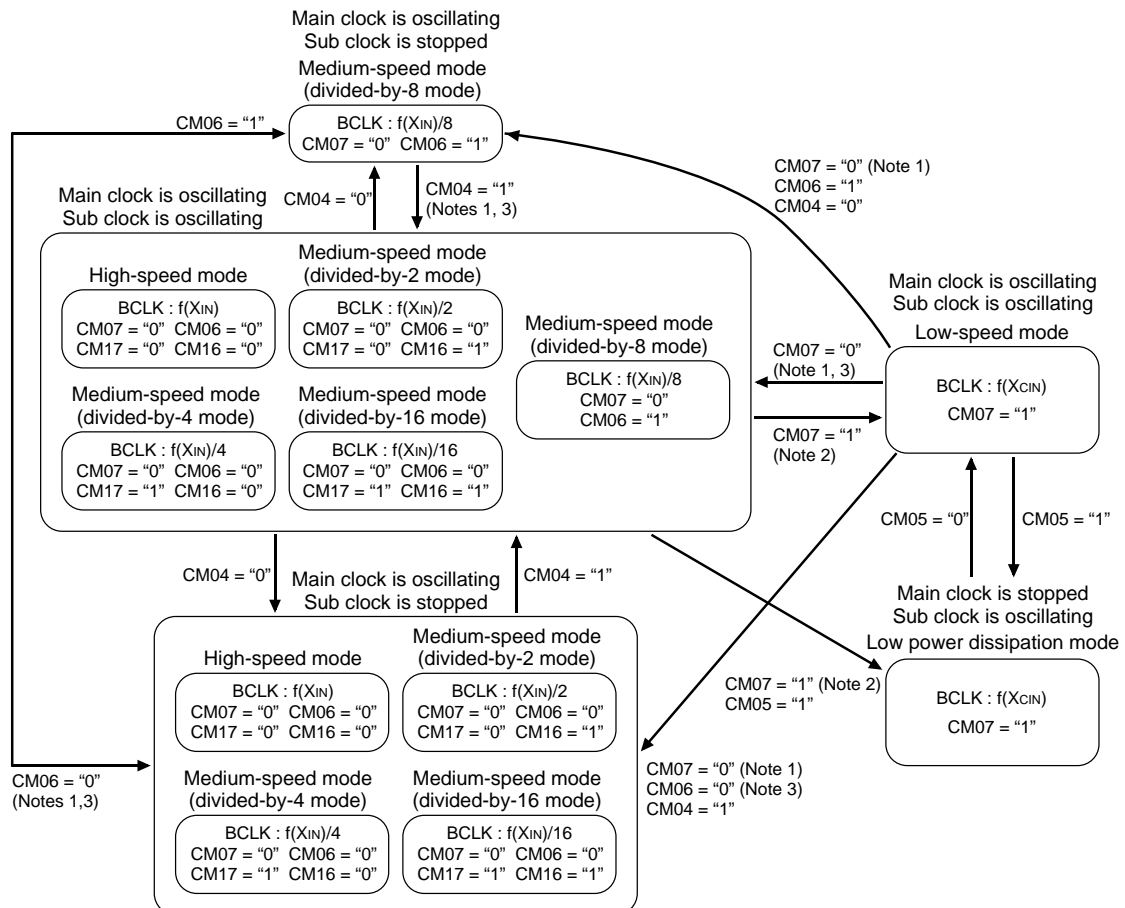


Figure 2.5.5 State transition diagram of Power control mode

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.6 Protection

The protection function is provided so that the values in important registers cannot be changed in the event that the program runs out of control. Figure 2.6.1 shows the protect register. The values in the processor mode register 0 (address 0004<sub>16</sub>), processor mode register 1 (address 0005<sub>16</sub>), system clock control register 0 (address 0006<sub>16</sub>), system clock control register 1 (address 0007<sub>16</sub>), port P9 direction register (address 03F3<sub>16</sub>), SI/O3 control register (address 0362<sub>16</sub>) and SI/O4 control register (address 0366<sub>16</sub>) can only be changed when the respective bit in the protect register is set to "1". Therefore, important outputs can be allocated to port P9.

If, after "1" (write-enabled) has been written to the port P9 direction register and SI/Oi control register (i=3,4) write-enable bit (bit 2 at address 000A<sub>16</sub>), a value is written to any address, the bit automatically reverts to "0" (write-inhibited). However, the system clock control registers 0 and 1 write-enable bit (bit 0 at 000A<sub>16</sub>) and processor mode register 0 and 1 write-enable bit (bit 1 at 000A<sub>16</sub>) do not automatically return to "0" after a value has been written to an address. The program must therefore be written to return these bits to "0".

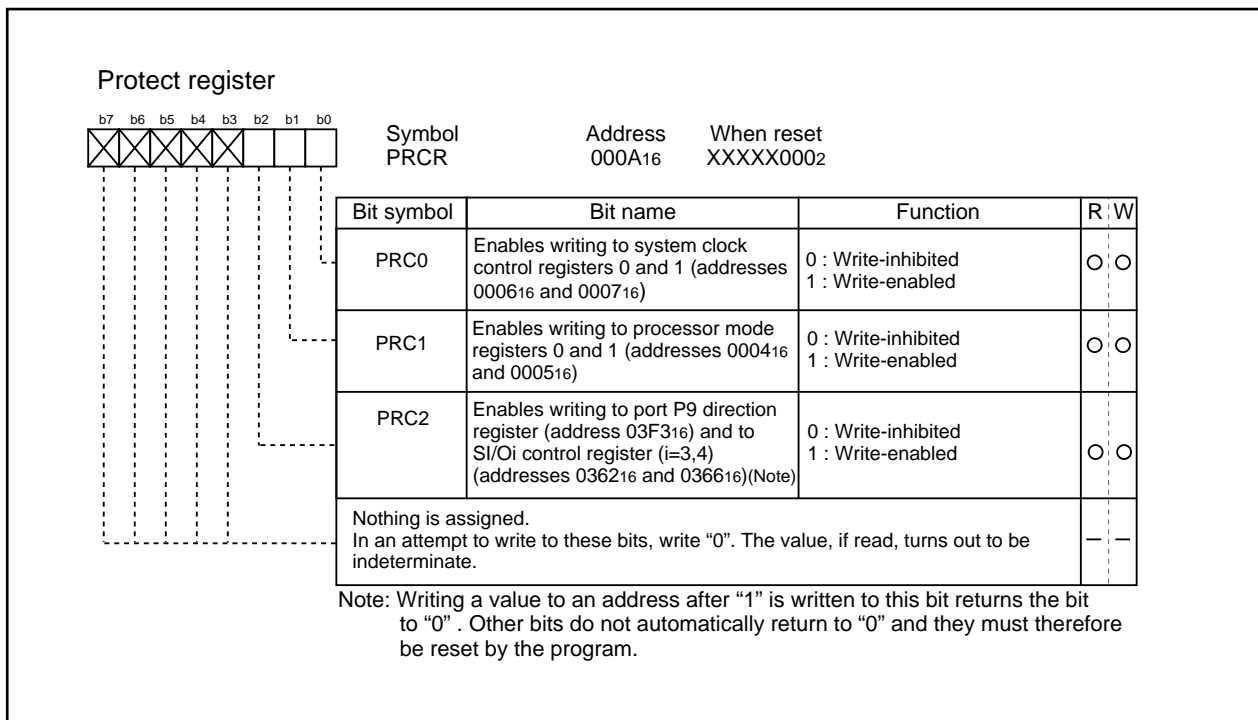


Figure 2.6.1 Protect register



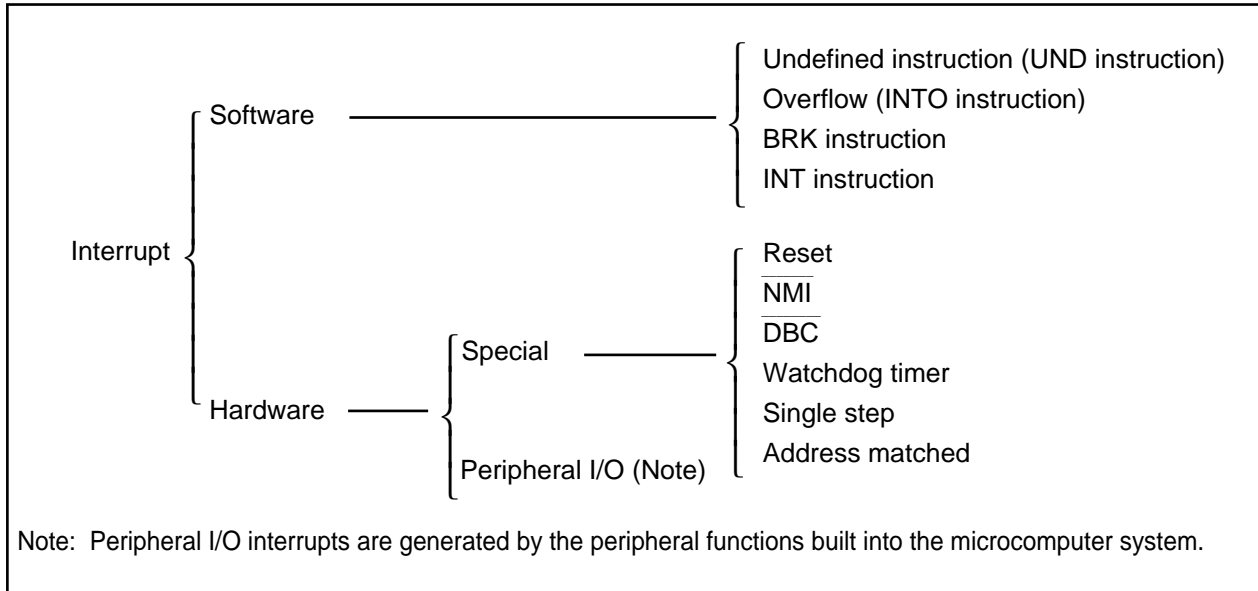
# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.7 Interrupt

### 2.7.1 Interrupt

Figure 2.7.1 lists the types of interrupts.



**Figure 2.7.1 Classification of interrupts**

- Maskable interrupt : An interrupt which can be enabled (disabled) by the interrupt enable flag (I flag) or whose interrupt priority **can be changed** by priority level.
- Non-maskable interrupt : An interrupt which cannot be enabled (disabled) by the interrupt enable flag (I flag) or whose interrupt priority **cannot be changed** by priority level.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.7.2 Software Interrupts

A software interrupt occurs when executing certain instructions. Software interrupts are non-maskable interrupts.

- **Undefined instruction interrupt**

An undefined instruction interrupt occurs when executing the UND instruction.

- **Overflow interrupt**

An overflow interrupt occurs when executing the INTO instruction with the overflow flag (O flag) set to "1". The following are instructions whose O flag changes by arithmetic:

ABS, ADC, ADCF, ADD, CMP, DIV, DIVU, DIVX, NEG, RMPA, SBB, SHA, SUB

- **BRK interrupt**

A BRK interrupt occurs when executing the BRK instruction.

- **INT interrupt**

An INT interrupt occurs when assigning one of software interrupt numbers 0 through 63 and executing the INT instruction. Software interrupt numbers 0 through 31 are assigned to peripheral I/O interrupts, so executing the INT instruction allows executing the same interrupt routine that a peripheral I/O interrupt does.

The stack pointer (SP) used for the INT interrupt is dependent on which software interrupt number is involved.

So far as software interrupt numbers 0 through 31 are concerned, the microcomputer saves the stack pointer assignment flag (U flag) when it accepts an interrupt request. It changes the U flag to "0" and selects the interrupt stack pointer (ISP), and then executes an interrupt sequence. When returning from the interrupt routine, the U flag is returned to the state it was before the acceptance of interrupt request. So far as software numbers 32 through 63 are concerned, the stack pointer does not make a shift.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
 with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.7.3 Hardware Interrupts

Hardware interrupts are classified into two types — special interrupts and peripheral I/O interrupts.

### (1) Special interrupts

Special interrupts are non-maskable interrupts.

- **Reset**

Reset occurs if an “L” is input to the  $\overline{\text{RESET}}$  pin.

- **$\overline{\text{NMI}}$  interrupt**

An  $\overline{\text{NMI}}$  interrupt occurs if an “L” is input to the  $\overline{\text{NMI}}$  pin.

- **$\overline{\text{DBC}}$  interrupt**

This interrupt is exclusively for the debugger, do not use it in other circumstances.

- **Watchdog timer interrupt**

Generated by the watchdog timer.

- **Single-step interrupt**

This interrupt is exclusively for the debugger, do not use it in other circumstances. With the debug flag (D flag) set to “1”, a single-step interrupt occurs after one instruction is executed.

- **Address match interrupt**

An address match interrupt occurs immediately before the instruction held in the address indicated by the address match interrupt register is executed with the address match interrupt enable bit set to “1”.

If an address other than the first address of the instruction in the address match interrupt register is set, no address match interrupt occurs. For address match interrupt, see 2.7.10 Address match Interrupt.

### (2) Peripheral I/O interrupts

A peripheral I/O interrupt is generated by one of built-in peripheral functions. Built-in peripheral functions are dependent on classes of products, so the interrupt factors too are dependent on classes of products. The interrupt vector table is the same as the one for software interrupt numbers 0 through 31 the INT instruction uses. Peripheral I/O interrupts are maskable interrupts.

- **Bus collision detection interrupt**

This is an interrupt that the serial I/O bus collision detection generates.

- **DMA0 interrupt, DMA1 interrupt**

These are interrupts that DMA generates.

- **Key-input interrupt**

A key-input interrupt occurs if an “L” is input to the  $\overline{\text{KI}}$  pin.

- **A-D conversion interrupt**

This is an interrupt that the A-D converter generates.

- **UART0, UART1, UART2/NACK, SI/O3 and SI/O4 transmission interrupt**

These are interrupts that the serial I/O transmission generates.

- **UART0, UART1, UART2/ACK, SI/O3 and SI/O4 reception interrupt**

These are interrupts that the serial I/O reception generates.

- **Timer A0 interrupt through timer A4 interrupt**

These are interrupts that timer A generates

- **Timer B0 interrupt through timer B5 interrupt**

These are interrupts that timer B generates.

- **$\overline{\text{INT0}}$  interrupt through  $\overline{\text{INT5}}$  interrupt**

An  $\overline{\text{INT}}$  interrupt occurs if either a rising edge or a falling edge or a both edge is input to the  $\overline{\text{INT}}$  pin.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.7.4 Interrupts and Interrupt Vector Tables

If an interrupt request is accepted, a program branches to the interrupt routine set in the interrupt vector table. Set the first address of the interrupt routine in each vector table. Figure 2.7.2 shows the format for specifying the address.

Two types of interrupt vector tables are available — fixed vector table in which addresses are fixed and variable vector table in which addresses can be varied by the setting.

	MSB	LSB
Vector address + 0	Low address	
Vector address + 1	Mid address	
Vector address + 2	0 0 0 0	High address
Vector address + 3	0 0 0 0	0 0 0 0

**Figure 2.7.2 Format for specifying interrupt vector addresses**

### • Fixed vector tables

The fixed vector table is a table in which addresses are fixed. The vector tables are located in an area extending from FFFDC<sub>16</sub> to FFFFF<sub>16</sub>. One vector table comprises four bytes. Set the first address of interrupt routine in each vector table. Table 2.7.1 shows the interrupts assigned to the fixed vector tables and addresses of vector tables.

**Table 2.7.1 Interrupts assigned to the fixed vector tables and addresses of vector tables**

Interrupt source	Vector table addresses Address (L) to address (H)	Remarks
Undefined instruction	FFFD <sub>C16</sub> to FFFD <sub>F16</sub>	Interrupt on UND instruction
Overflow	FFFE <sub>016</sub> to FFFE <sub>316</sub>	Interrupt on INTO instruction
BRK instruction	FFFE <sub>416</sub> to FFFE <sub>716</sub>	If the vector contains FF <sub>16</sub> , program execution starts from the address shown by the vector in the variable vector table
Address match	FFFE <sub>816</sub> to FFFE <sub>B16</sub>	There is an address-matching interrupt enable bit
Single step (Note)	FFFE <sub>C16</sub> to FFFE <sub>F16</sub>	Do not use
Watchdog timer	FFFF <sub>016</sub> to FFFF <sub>316</sub>	
DBC (Note)	FFFF <sub>416</sub> to FFFF <sub>716</sub>	Do not use
NMI	FFFF <sub>816</sub> to FFFF <sub>B16</sub>	External interrupt by input to NMI pin
Reset	FFFF <sub>C16</sub> to FFFF <sub>F16</sub>	

Note: Interrupts used for debugging purposes only.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## • Variable vector tables

The addresses in the variable vector table can be modified, according to the user's settings. Indicate the first address using the interrupt table register (INTB). The 256-byte area subsequent to the address the INTB indicates becomes the area for the variable vector tables. One vector table comprises four bytes. Set the first address of the interrupt routine in each vector table. Table 2.7.2 shows the interrupts assigned to the variable vector tables and addresses of vector tables.

**Table 2.7.2 Interrupts assigned to the variable vector tables and addresses of vector tables**

Software interrupt number	Vector table address Address (L) to address (H)	Interrupt source	Remarks
Software interrupt number 0	+0 to +3 (Note 1)	BRK instruction	Cannot be masked I flag
Software interrupt number 4	+16 to +19 (Note 1)	INT3	
Software interrupt number 5	+20 to +23 (Note 1)	Timer B5	
Software interrupt number 6	+24 to +27 (Note 1)	Timer B4	
Software interrupt number 7	+28 to +31 (Note 1)	Timer B3	
Software interrupt number 8	+32 to +35 (Note 1)	SI/O4/INT5 (Note 2)	
Software interrupt number 9	+36 to +39 (Note 1)	SI/O3/INT4 (Note 2)	
Software interrupt number 10	+40 to +43 (Note 1)	Bus collision detection	
Software interrupt number 11	+44 to +47 (Note 1)	DMA0	
Software interrupt number 12	+48 to +51 (Note 1)	DMA1	
Software interrupt number 13	+52 to +55 (Note 1)	Key input interrupt	
Software interrupt number 14	+56 to +59 (Note 1)	A-D	
Software interrupt number 15	+60 to +63 (Note 1)	UART2 transmit/NACK (Note 3)	
Software interrupt number 16	+64 to +67 (Note 1)	UART2 receive/ACK (Note 3)	
Software interrupt number 17	+68 to +71 (Note 1)	UART0 transmit	
Software interrupt number 18	+72 to +75 (Note 1)	UART0 receive	
Software interrupt number 19	+76 to +79 (Note 1)	UART1 transmit	
Software interrupt number 20	+80 to +83 (Note 1)	UART1 receive	
Software interrupt number 21	+84 to +87 (Note 1)	Timer A0	
Software interrupt number 22	+88 to +91 (Note 1)	Timer A1	
Software interrupt number 23	+92 to +95 (Note 1)	Timer A2	
Software interrupt number 24	+96 to +99 (Note 1)	Timer A3	
Software interrupt number 25	+100 to +103 (Note 1)	Timer A4	
Software interrupt number 26	+104 to +107 (Note 1)	Timer B0	
Software interrupt number 27	+108 to +111 (Note 1)	Timer B1	
Software interrupt number 28	+112 to +115 (Note 1)	Timer B2	
Software interrupt number 29	+116 to +119 (Note 1)	INT0	
Software interrupt number 30	+120 to +123 (Note 1)	INT1	
Software interrupt number 31	+124 to +127 (Note 1)	INT2	
Software interrupt number 32 to Software interrupt number 63	+128 to +131 (Note 1) to +252 to +255 (Note 1)	Software interrupt	Cannot be masked I flag

Note 1: Address relative to address in interrupt table register (INTB).

Note 2: It is selected by interrupt request cause bit (bit 6, 7 in address 035F16).

Note 3: When IIC mode is selected, NACK and ACK interrupts are selected.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.7.5 Interrupt Control

Descriptions are given here regarding how to enable or disable maskable interrupts and how to set the priority to be accepted. What is described here does not apply to non-maskable interrupts.

Enable or disable a maskable interrupt using the interrupt enable flag (I flag), interrupt priority level selection bit, or processor interrupt priority level (IPL). Whether an interrupt request is present or absent is indicated by the interrupt request bit. The interrupt request bit and the interrupt priority level selection bit are located in the interrupt control register of each interrupt. Also, the interrupt enable flag (I flag) and the IPL are located in the flag register (FLG).

Figure 2.7.3 shows the memory map of the interrupt control registers.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

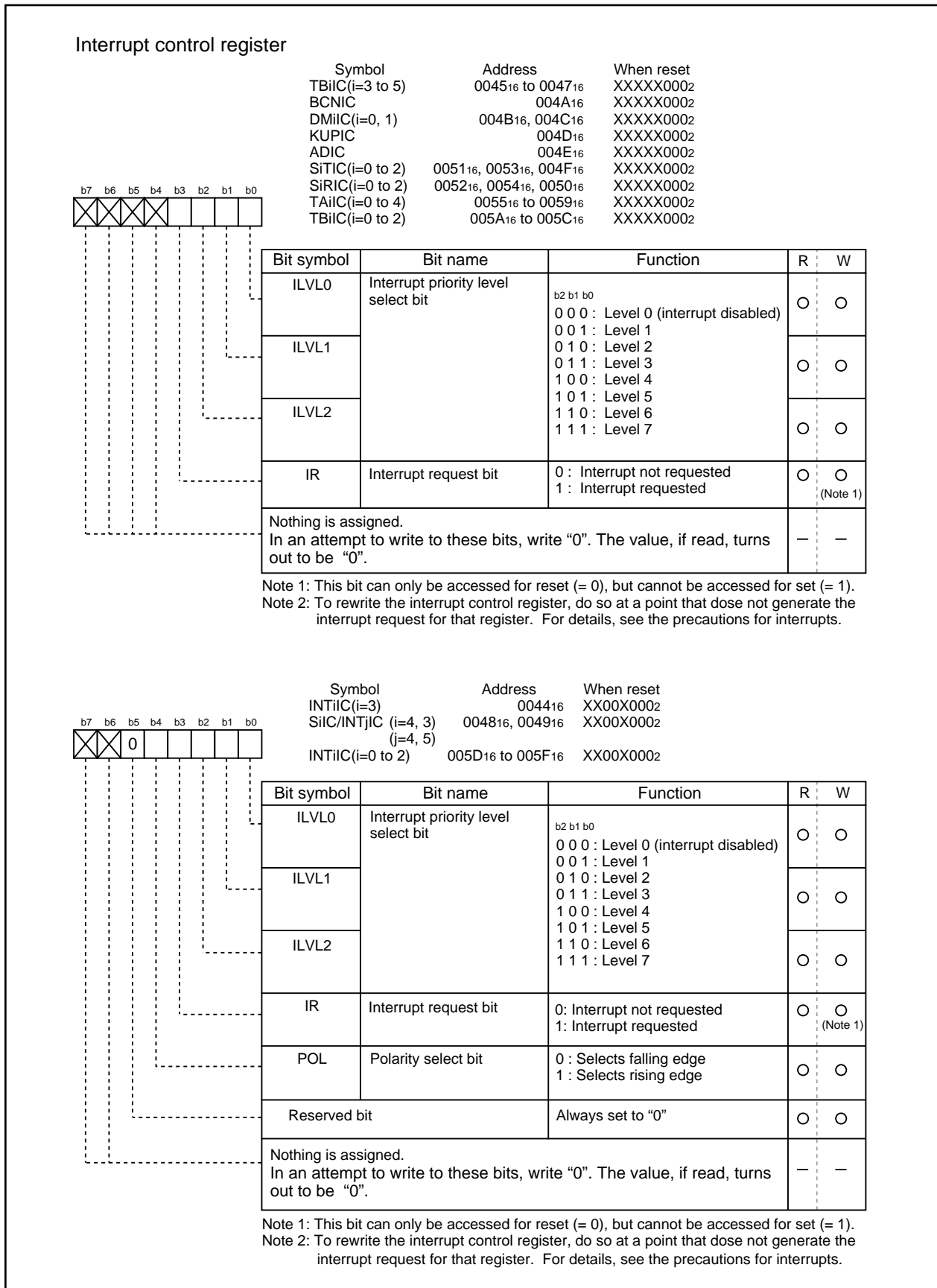


Figure 2.7.3 Interrupt control registers

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (1) Interrupt Enable Flag (I flag)

The interrupt enable flag (I flag) controls the enabling and disabling of maskable interrupts. Setting this flag to "1" enables all maskable interrupts; setting it to "0" disables all maskable interrupts. This flag is set to "0" after reset.

## (2) Interrupt Request Bit

The interrupt request bit is set to "1" by hardware when an interrupt is requested. After the interrupt is accepted and jumps to the corresponding interrupt vector, the request bit is set to "0" by hardware. The interrupt request bit can also be set to "0" by software. (Do not set this bit to "1").

## (3) Interrupt Priority Level Select Bit and Processor Interrupt Priority Level (IPL)

Set the interrupt priority level using the interrupt priority level select bit, which is one of the component bits of the interrupt control register. When an interrupt request occurs, the interrupt priority level is compared with the IPL. The interrupt is enabled only when the priority level of the interrupt is higher than the IPL. Therefore, setting the interrupt priority level to "0" disables the interrupt.

Table 2.7.3 shows the settings of interrupt priority levels and Table 2.7.4 shows the interrupt levels enabled, according to the consist of the IPL.

The following are conditions under which an interrupt is accepted:

- interrupt enable flag (I flag) = 1
- interrupt request bit = 1
- interrupt priority level > IPL

The interrupt enable flag (I flag), the interrupt request bit, the interrupt priority select bit, and the IPL are independent, and they are not affected by one another.

**Table 2.7.3 Settings of interrupt priority levels**

Interrupt priority level select bit	Interrupt priority level	Priority order
b2 b1 b0 0 0 0	Level 0 (interrupt disabled)	———
0 0 1	Level 1	<div style="text-align: center;">                     Low                      ↓                      High                 </div>
0 1 0	Level 2	
0 1 1	Level 3	
1 0 0	Level 4	
1 0 1	Level 5	
1 1 0	Level 6	
1 1 1	Level 7	

**Table 2.7.4 Interrupt levels enabled according to the contents of the IPL**

IPL	Enabled interrupt priority levels
IPL <sub>2</sub> IPL <sub>1</sub> IPL <sub>0</sub> 0 0 0	Interrupt levels 1 and above are enabled
0 0 1	Interrupt levels 2 and above are enabled
0 1 0	Interrupt levels 3 and above are enabled
0 1 1	Interrupt levels 4 and above are enabled
1 0 0	Interrupt levels 5 and above are enabled
1 0 1	Interrupt levels 6 and above are enabled
1 1 0	Interrupt levels 7 and above are enabled
1 1 1	All maskable interrupts are disabled



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (4) Rewrite the interrupt control register

To rewrite the interrupt control register, do so at a point that does not generate the interrupt request for that register. If there is possibility of the interrupt request occur, rewrite the interrupt control register after the interrupt is disabled. The program examples are described as follow:

### Example 1:

```
INT_SWITCH1:
  FCLR  I           ; Disable interrupts.
  AND.B #00h, 0055h ; Clear TA0IC int. priority level and int. request bit.
  NOP                     ; Four NOP instructions are required when using HOLD function.
  NOP
  FSET  I           ; Enable interrupts.
```

### Example 2:

```
INT_SWITCH2:
  FCLR  I           ; Disable interrupts.
  AND.B #00h, 0055h ; Clear TA0IC int. priority level and int. request bit.
  MOV.W MEM, R0     ; Dummy read.
  FSET  I           ; Enable interrupts.
```

### Example 3:

```
INT_SWITCH3:
  PUSHC FLG         ; Push Flag register onto stack
  FCLR  I           ; Disable interrupts.
  AND.B #00h, 0055h ; Clear TA0IC int. priority level and int. request bit.
  POPC  FLG         ; Enable interrupts.
```

The reason why two NOP instructions (four when using the HOLD function) or dummy read are inserted before FSET I in Examples 1 and 2 is to prevent the interrupt enable flag I from being set before the interrupt control register is rewritten due to effects of the instruction queue.

When a instruction to rewrite the interrupt control register is executed but the interrupt is disabled, the interrupt request bit is not set sometimes even if the interrupt request for that register has been generated. This will depend on the instruction. If this creates problems, use the below instructions to change the register.

Instructions : AND, OR, BCLR, BSET

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.7.6 Interrupt Sequence

An interrupt sequence — what are performed over a period from the instant an interrupt is accepted to the instant the interrupt routine is executed — is described here.

If an interrupt occurs during execution of an instruction, the processor determines its priority when the execution of the instruction is completed, and transfers control to the interrupt sequence from the next cycle. If an interrupt occurs during execution of either the SMOVB, SMOVF, SSTR or RMPA instruction, the processor temporarily suspends the instruction being executed, and transfers control to the interrupt sequence.

In the interrupt sequence, the processor carries out the following in sequence given:

- (a) CPU gets the interrupt information (the interrupt number and interrupt request level) by reading address 00000<sub>16</sub>.
- (b) Saves the content of the flag register (FLG) as it was immediately before the start of interrupt sequence in the temporary register (Note) within the CPU.
- (c) Sets the interrupt enable flag (I flag), the debug flag (D flag), and the stack pointer select flag (U flag) to "0" (the U flag, however does not change if the INT instruction, in software interrupt numbers 32 through 63, is executed)
- (d) Saves the content of the temporary register (Note) within the CPU in the stack area.
- (e) Saves the content of the program counter (PC) in the stack area.
- (f) Sets the interrupt priority level of the accepted instruction in the IPL.

After the interrupt sequence is completed, the processor resumes executing instructions from the first address of the interrupt routine.

Note: This register cannot be utilized by the user.

### (1) Interrupt Response Time

'Interrupt response time' is the period between the instant an interrupt occurs and the instant the first instruction within the interrupt routine has been executed. This time comprises the period from the occurrence of an interrupt to the completion of the instruction under execution at that moment (a) and the time required for executing the interrupt sequence (b). Figure 2.7.4 shows the interrupt response time.

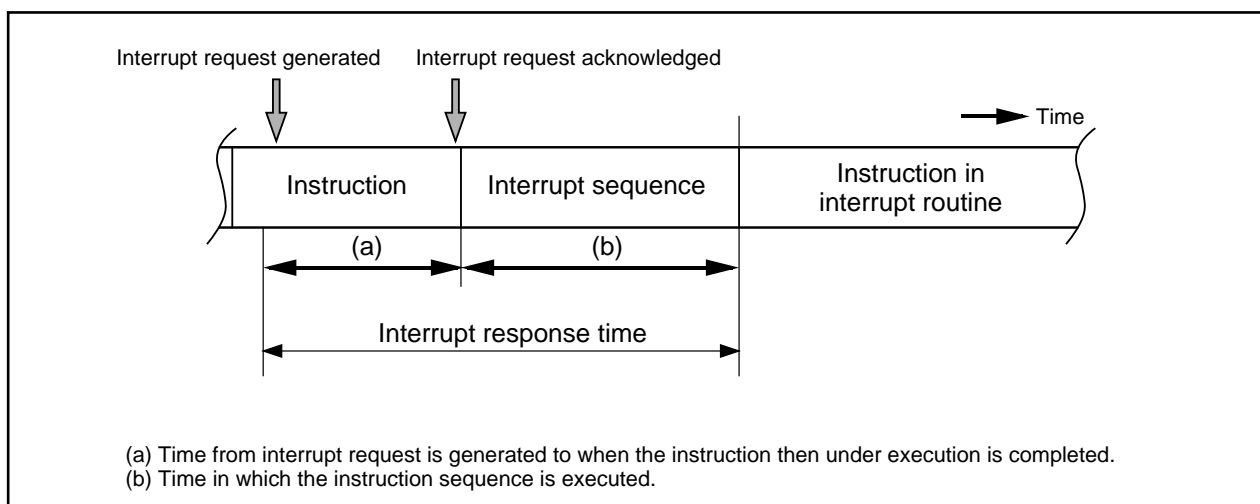


Figure 2.7.4 Interrupt response time

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Time (a) is dependent on the instruction under execution. Thirty cycles is the maximum required for the DIVX instruction (without wait).

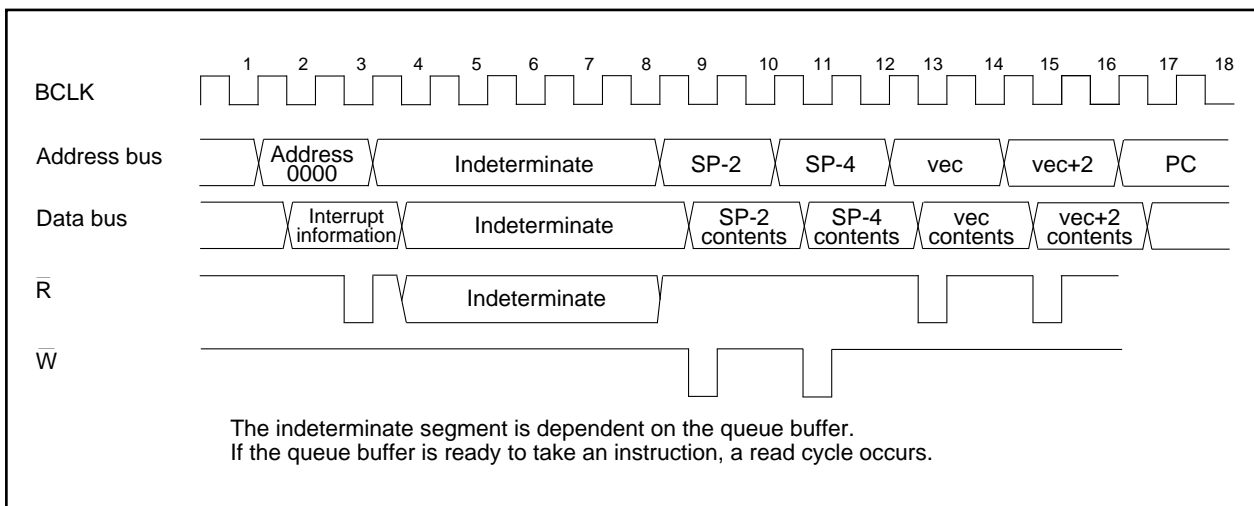
Time (b) is as shown in Table 2.7.5

**Table 2.7.5 Time required for executing the interrupt sequence**

Interrupt vector address	Stack pointer (SP) value	16-Bit bus, without wait	8-Bit bus, without wait
Even	Even	18 cycles (Note 1)	20 cycles (Note 1)
Even	Odd	19 cycles (Note 1)	20 cycles (Note 1)
Odd (Note 2)	Even	19 cycles (Note 1)	20 cycles (Note 1)
Odd (Note 2)	Odd	20 cycles (Note 1)	20 cycles (Note 1)

Notes 1: Add 2 cycles in the case of a DBC interrupt; add 1 cycle in the case either of an address coincidence interrupt or of a single-step interrupt.

Notes 2: Locate an interrupt vector address in an even address, if possible.



**Figure 2.7.5 Time required for executing the interrupt sequence**

## (2) Variation of IPL when Interrupt Request is Accepted

If an interrupt request is accepted, the interrupt priority level of the accepted interrupt is set in the IPL.

If an interrupt request, that does not have an interrupt priority level, is accepted, one of the values shown in Table 2.7.6 is set in the IPL.

**Table 2.7.6 Relationship between interrupts without interrupt priority levels and IPL**

Interrupt sources without priority levels	Value set in the IPL
Watchdog timer, NMI	7
Reset	0
Other	Not changed

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (3) Saving Registers

In the interrupt sequence, only the contents of the flag register (FLG) and that of the program counter (PC) are saved in the stack area.

First, the processor saves the four higher-order bits of the program counter, and 4 upper-order bits and 8 lower-order bits of the FLG register, 16 bits in total, in the stack area, then saves 16 lower-order bits of the program counter. Figure 2.7.6 shows the state of the stack as it was before the acceptance of the interrupt request, and the state the stack after the acceptance of the interrupt request.

Save other necessary registers at the beginning of the interrupt routine using software. Using the PUSHM instruction alone can save all the registers except the stack pointer (SP).

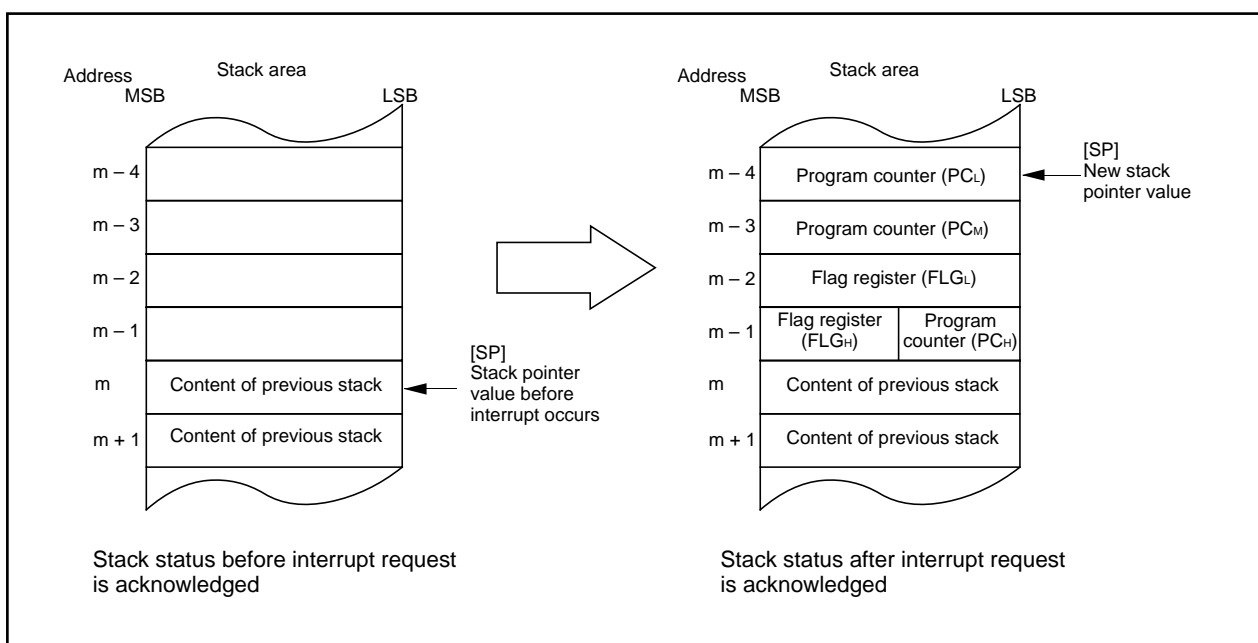


Figure 2.7.6 State of stack before and after acceptance of interrupt request

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

The operation of saving registers carried out in the interrupt sequence is dependent on whether the content of the stack pointer, at the time of acceptance of an interrupt request, is even or odd. If the content of the stack pointer (Note) is even, the content of the flag register (FLG) and the content of the program counter (PC) are saved, 16 bits at a time. If odd, their contents are saved in two steps, 8 bits at a time. Figure 2.7.7 shows the operation of the saving registers.

Note: Stack pointer indicated by U flag.

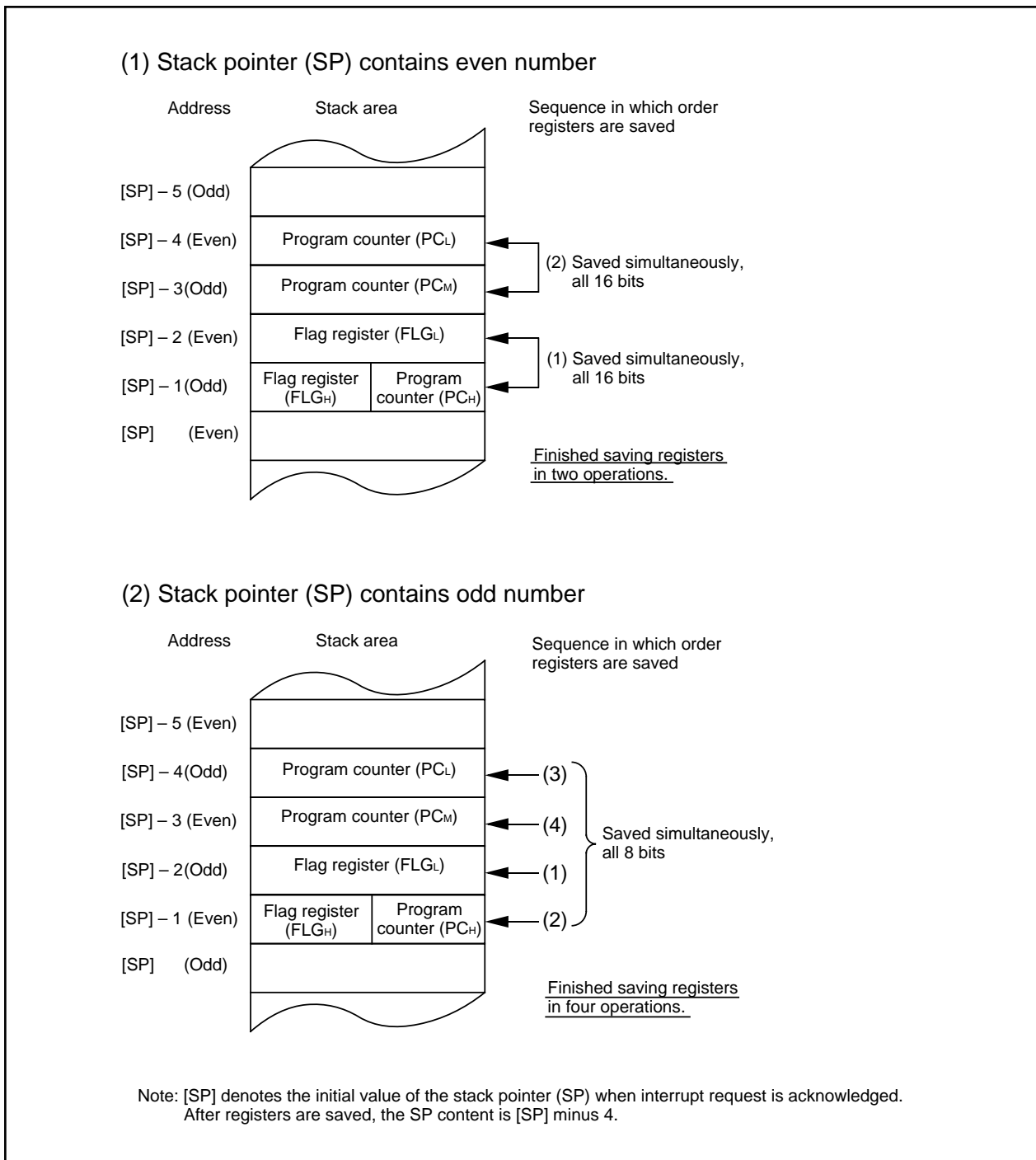


Figure 2.7.7 Operation of saving registers

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (4) Returning from an Interrupt Routine

Executing the REIT instruction at the end of an interrupt routine returns the contents of the flag register (FLG) as it was immediately before the start of interrupt sequence and the contents of the program counter (PC), both of which have been saved in the stack area. Then control returns to the program that was being executed before the acceptance of the interrupt request, so that the suspended process resumes.

Return the other registers saved by software within the interrupt routine using the POPM or similar instruction before executing the REIT instruction.

## (5) Interrupt Priority

If there are two or more interrupt requests occurring at a point in time within a single sampling (checking whether interrupt requests are made), the interrupt assigned a higher priority is accepted.

Assign an arbitrary priority to maskable interrupts (peripheral I/O interrupts) using the interrupt priority level select bit. If the same interrupt priority level is assigned, however, the interrupt assigned a higher hardware priority is accepted.

Priorities of the special interrupts, such as Reset (dealt with as an interrupt assigned the highest priority), watchdog timer interrupt, etc. are regulated by hardware.

Figure 2.7.8 shows the priorities of hardware interrupts.

Software interrupts are not affected by the interrupt priority. If an instruction is executed, control branches invariably to the interrupt routine.

Reset >  $\overline{\text{NMI}}$  >  $\overline{\text{DBC}}$  > Watchdog timer > Peripheral I/O > Single step > Address match

**Figure 2.7.8 Hardware interrupts priorities**

## (6) Interrupt resolution circuit

When two or more interrupts are generated simultaneously, this circuit selects the interrupt with the highest priority level. Figure 2.7.9 shows the circuit that judges the interrupt priority level.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

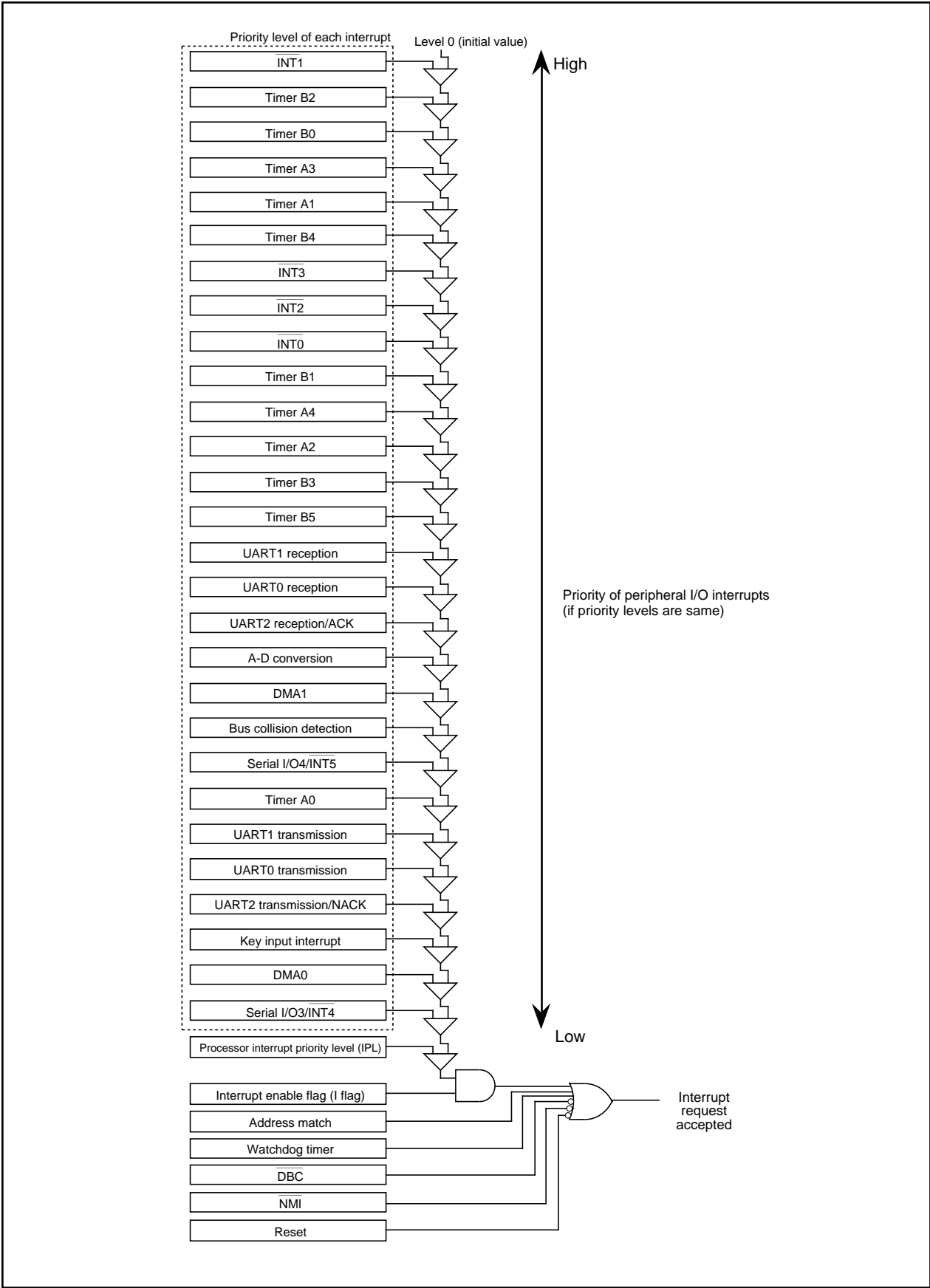


Figure 2.7.9 Maskable interrupts priorities (peripheral I/O interrupts)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.7.7 INT Interrupt

INT0 to INT5 are triggered by the edges of external inputs. The edge polarity is selected using the polarity select bit.

Of interrupt control registers, 0048<sub>16</sub> is used both as serial I/O4 and external interrupt INT5 input control register, and 0049<sub>16</sub> is used both as serial I/O3 and as external interrupt INT4 input control register. Use the interrupt request cause select bits - bits 6 and 7 of the interrupt request cause select register (035F<sub>16</sub>) - to specify which interrupt request cause to select. After having set an interrupt request cause, be sure to clear the corresponding interrupt request bit before enabling an interrupt. Either of the interrupt control registers - 0048<sub>16</sub>, 0049<sub>16</sub> - has the polarity-switching bit. Be sure to set this bit to "0" to select an serial I/O as the interrupt request cause.

As for external interrupt input, an interrupt can be generated both at the rising edge and at the falling edge by setting "1" in the INT<sub>i</sub> interrupt polarity switching bit of the interrupt request cause select register (035F<sub>16</sub>). To select both edges, set the polarity switching bit of the corresponding interrupt control register to 'falling edge' ("0").

Figure 2.7.10 shows the Interrupt request cause select register.

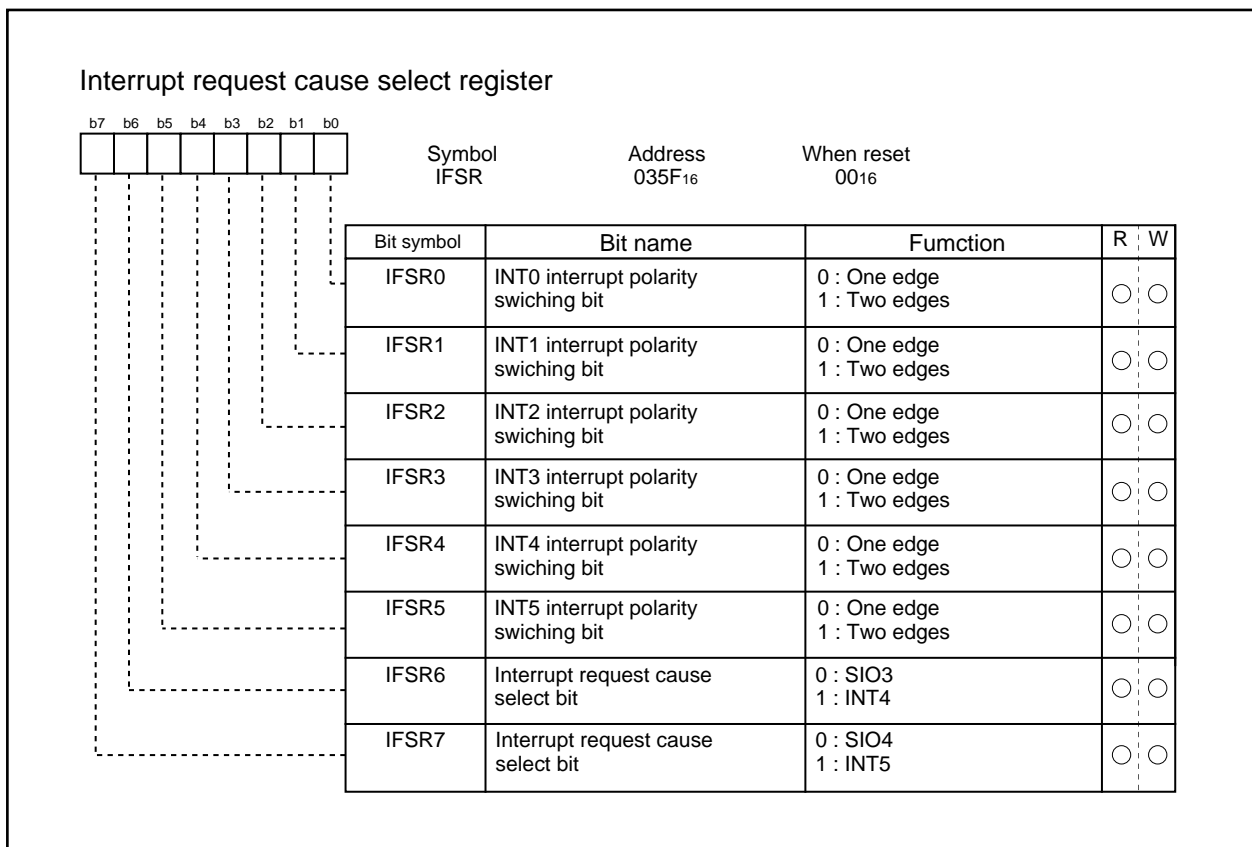


Figure 2.7.10 Interrupt request cause select register



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.7.8 $\overline{\text{NMI}}$ Interrupt

An  $\overline{\text{NMI}}$  interrupt is generated when the input to the P85/ $\overline{\text{NMI}}$  pin changes from "H" to "L". The  $\overline{\text{NMI}}$  interrupt is a non-maskable external interrupt. The pin level can be checked in the port P85 register (bit 5 at address 03F0<sub>16</sub>).

This pin cannot be used as a normal port input.

## 2.7.9 Key Input Interrupt

If the direction register of any of P104 to P107 is set for input and a falling edge is input to that port, a key input interrupt is generated. A key input interrupt can also be used as a key-on wakeup function for cancelling the wait mode or stop mode. However, if you intend to use the key input interrupt, do not use P104 to P107 as A-D input ports. Figure 2.7.11 shows the block diagram of the key input interrupt. Note that if an "L" level is input to any pin that has not been disabled for input, inputs to the other pins are not detected as an interrupt.

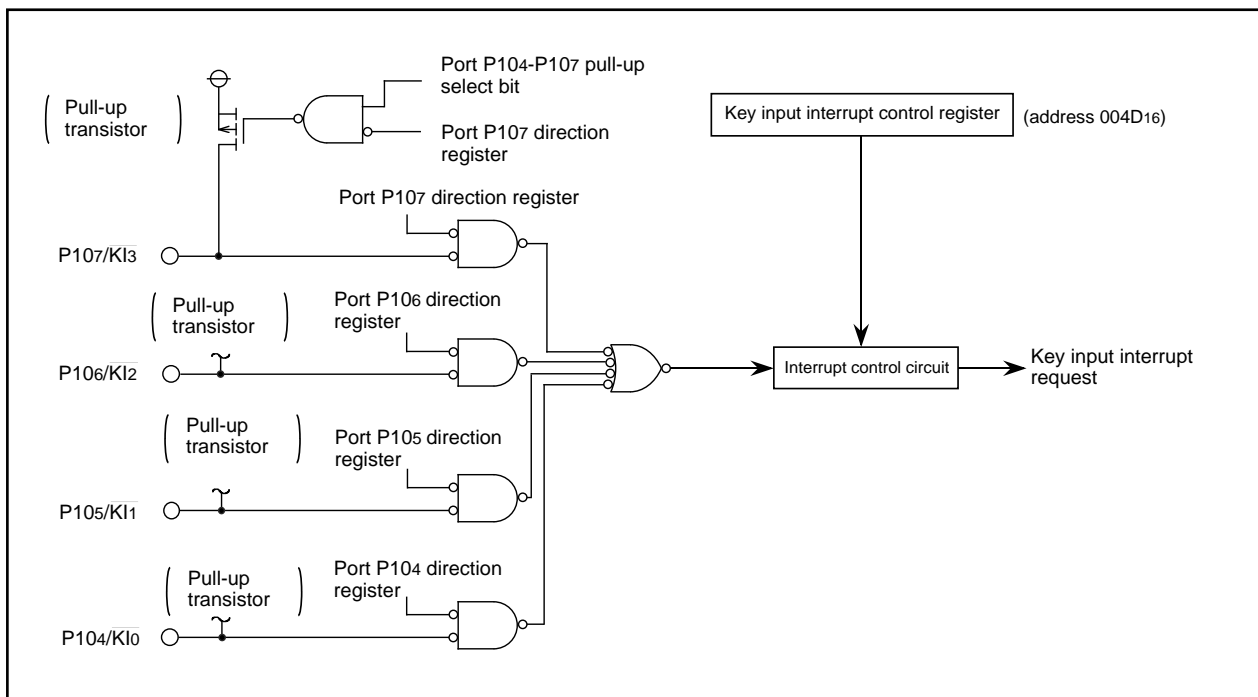


Figure 2.7.11 Block diagram of key input interrupt

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

2.7.10 Address Match Interrupt

An address match interrupt is generated when the address match interrupt address register contents match the program counter value. Two address match interrupts can be set, each of which can be enabled and disabled by an address match interrupt enable bit. Address match interrupts are not affected by the interrupt enable flag (I flag) and processor interrupt priority level (IPL). The value of the program counter (PC) for an address match interrupt varies depending on the instruction being executed. Figure 2.7.12 shows the address match interrupt-related registers.

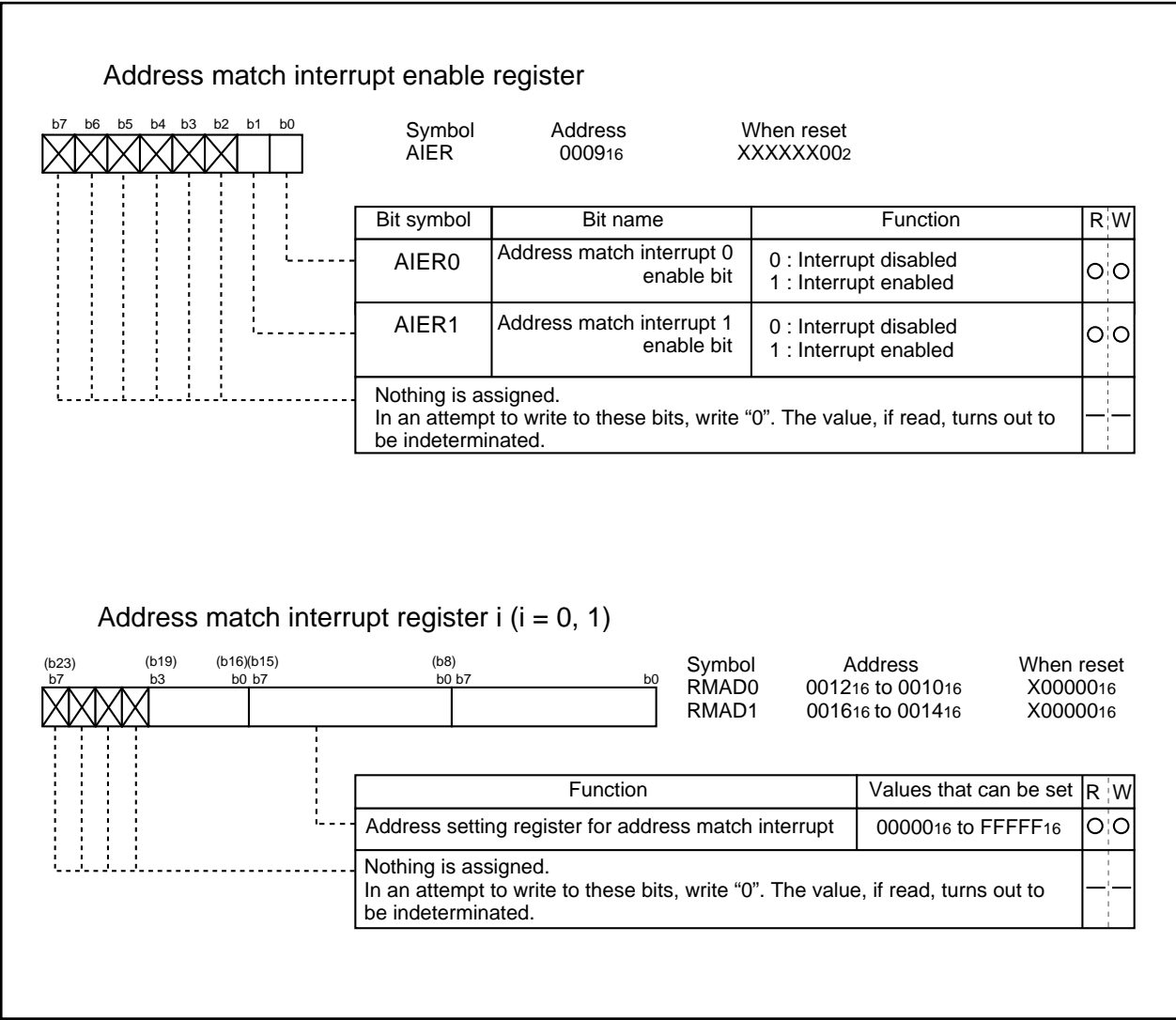


Figure 2.7.12 Address match interrupt-related registers

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.7.11 Precautions for Interrupts

### (1) Reading address 00000<sub>16</sub>

- When maskable interrupt is occurred, CPU read the interrupt information (the interrupt number and interrupt request level) in the interrupt sequence. The interrupt request bit of the certain interrupt written in address 00000<sub>16</sub> will then be set to "0". Reading address 00000<sub>16</sub> by software sets enabled highest priority interrupt source request bit to "0".  
Though the interrupt is generated, the interrupt routine may not be executed.  
Do not read address 00000<sub>16</sub> by software.

### (2) Setting the stack pointer

- The value of the stack pointer immediately after reset is initialized to 0000<sub>16</sub>. Accepting an interrupt before setting a value in the stack pointer may become a factor of runaway. Be sure to set a value in the stack pointer before accepting an interrupt. When using the  $\overline{\text{NMI}}$  interrupt, initialize the stack point at the beginning of a program. Concerning the first instruction immediately after reset, generating any interrupts including the  $\overline{\text{NMI}}$  interrupt is prohibited.

### (3) The $\overline{\text{NMI}}$ interrupt

- As for the  $\overline{\text{NMI}}$  interrupt pin, an interrupt cannot be disabled. Connect it to the Vcc pin via a resistor (pull-up) if unused. Be sure to work on it.
- The  $\overline{\text{NMI}}$  pin also serves as P85, which is exclusively input. Reading the contents of the P8 register allows reading the pin value. Use the reading of this pin only for establishing the pin level at the time when the  $\overline{\text{NMI}}$  interrupt is input.
- Do not reset the CPU with the input to the  $\overline{\text{NMI}}$  pin being in the "L" state.
- Do not attempt to go into stop mode with the input to the  $\overline{\text{NMI}}$  pin being in the "L" state. With the input to the  $\overline{\text{NMI}}$  being in the "L" state, the CM10 is fixed to "0", so attempting to go into stop mode is turned down.
- Do not attempt to go into wait mode with the input to the  $\overline{\text{NMI}}$  pin being in the "L" state. With the input to the  $\overline{\text{NMI}}$  pin being in the "L" state, the CPU stops but the oscillation does not stop, so no power is saved. In this instance, the CPU is returned to the normal state by a later interrupt.
- Signals input to the  $\overline{\text{NMI}}$  pin require an "L" level of 1 clock or more, from the operation clock of the CPU.

### (4) External interrupt

- Either an "L" level or an "H" level of at least 250 ns width is necessary for the signal input to pins  $\overline{\text{INT0}}$  through  $\overline{\text{INT5}}$  regardless of the CPU operation clock.
- When the polarity of the  $\overline{\text{INT0}}$  to  $\overline{\text{INT5}}$  pins is changed, the interrupt request bit is sometimes set to "1". After changing the polarity, set the interrupt request bit to "0". Figure 2.7.13 shows the procedure for changing the  $\overline{\text{INT}}$  interrupt generate factor.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

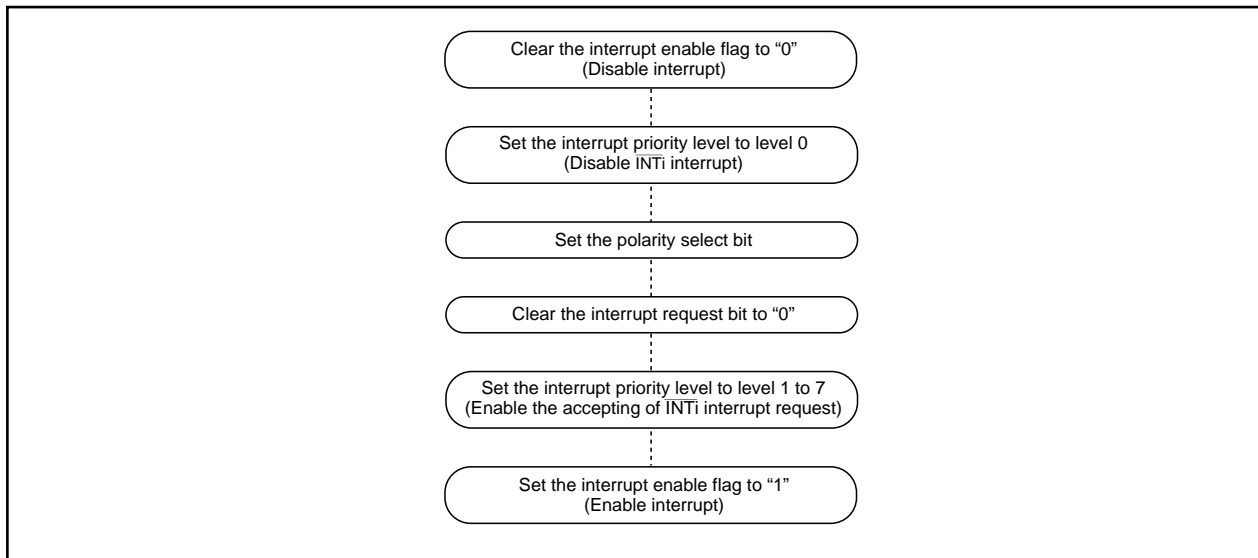


Figure 2.7.13 Switching condition of INT interrupt request

## (5) Rewrite the interrupt control register

- To rewrite the interrupt control register, do so at a point that does not generate the interrupt request for that register. If there is possibility of the interrupt request occur, rewrite the interrupt control register after the interrupt is disabled. The program examples are described as follow:

### Example 1:

```

INT_SWITCH1:
  FCLR  I          ; Disable interrupts.
  AND.B #00h, 0055h ; Clear TA0IC int. priority level and int. request bit.
  NOP                      ; Four NOP instructions are required when using HOLD function.
  NOP
  FSET  I          ; Enable interrupts.
  
```

### Example 2:

```

INT_SWITCH2:
  FCLR  I          ; Disable interrupts.
  AND.B #00h, 0055h ; Clear TA0IC int. priority level and int. request bit.
  MOV.W MEM, R0     ; Dummy read.
  FSET  I          ; Enable interrupts.
  
```

### Example 3:

```

INT_SWITCH3:
  PUSHC FLG        ; Push Flag register onto stack
  FCLR  I          ; Disable interrupts.
  AND.B #00h, 0055h ; Clear TA0IC int. priority level and int. request bit.
  POPC  FLG        ; Enable interrupts.
  
```

The reason why two NOP instructions (four when using the HOLD function) or dummy read are inserted before FSET I in Examples 1 and 2 is to prevent the interrupt enable flag I from being set before the interrupt control register is rewritten due to effects of the instruction queue.

- When a instruction to rewrite the interrupt control register is executed but the interrupt is disabled, the interrupt request bit is not set sometimes even if the interrupt request for that register has been generated. This will depend on the instruction. If this creates problems, use the below instructions to change the register.

Instructions : AND, OR, BCLR, BSET

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.8 Watchdog Timer

The watchdog timer has the function of detecting when the program is out of control. The watchdog timer is a 15-bit counter which down-counts the clock derived by dividing the BCLK using the prescaler. A watchdog timer interrupt is generated when an underflow occurs in the watchdog timer. When XIN is selected for the BCLK, bit 7 of the watchdog timer control register (address 000F<sub>16</sub>) selects the prescaler division ratio (by 16 or by 128). When XCIN is selected as the BCLK, the prescaler is set for division by 2 regardless of bit 7 of the watchdog timer control register (address 000F<sub>16</sub>). Thus the watchdog timer's period can be calculated as given below. The watchdog timer's period is, however, subject to an error due to the pre-scaler.

With XIN chosen for BCLK

$$\text{Watchdog timer period} = \frac{\text{pre-scaler dividing ratio (16 or 128)} \times \text{watchdog timer count (32768)}}{\text{BCLK}}$$

With XCIN chosen for BCLK

$$\text{Watchdog timer period} = \frac{\text{pre-scaler dividing ratio (2)} \times \text{watchdog timer count (32768)}}{\text{BCLK}}$$

For example, suppose that BCLK runs at 10 MHz and that 16 has been chosen for the dividing ratio of the pre-scaler, then the watchdog timer's period becomes approximately 52.4 ms.

The watchdog timer is initialized by writing to the watchdog timer start register (address 000E<sub>16</sub>) and when a watchdog timer interrupt request is generated. The prescaler is initialized only when the microcomputer is reset. After a reset is cancelled, the watchdog timer and prescaler are both stopped. The count is started by writing to the watchdog timer start register (address 000E<sub>16</sub>).

Figure 2.8.1 shows the block diagram of the watchdog timer. Figure 2.8.2 shows the watchdog timer-related registers.

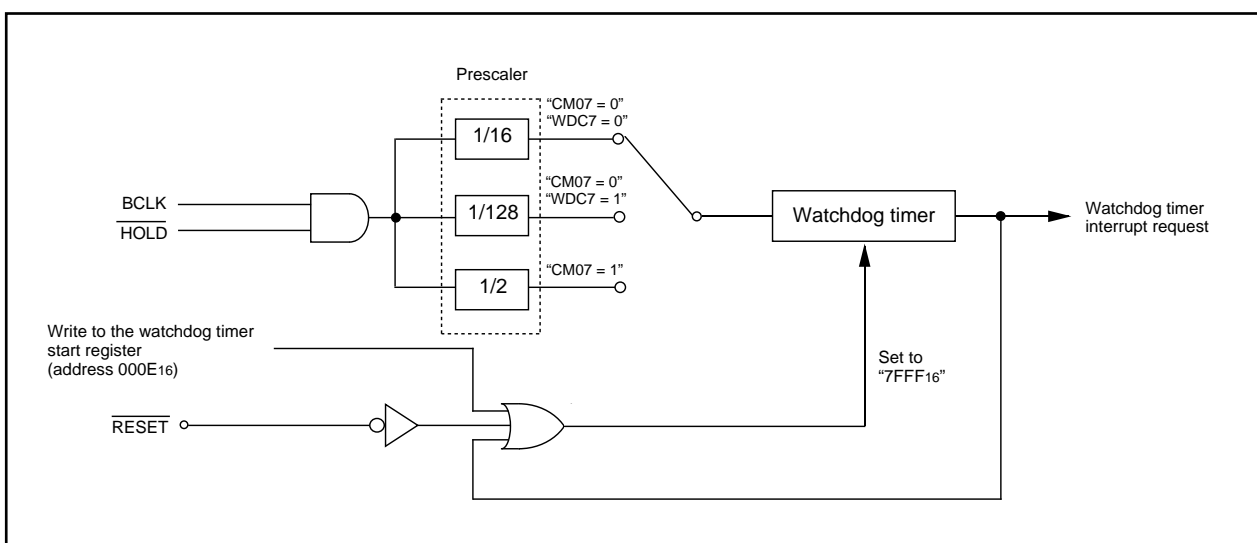


Figure 2.8.1 Block diagram of watchdog timer

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

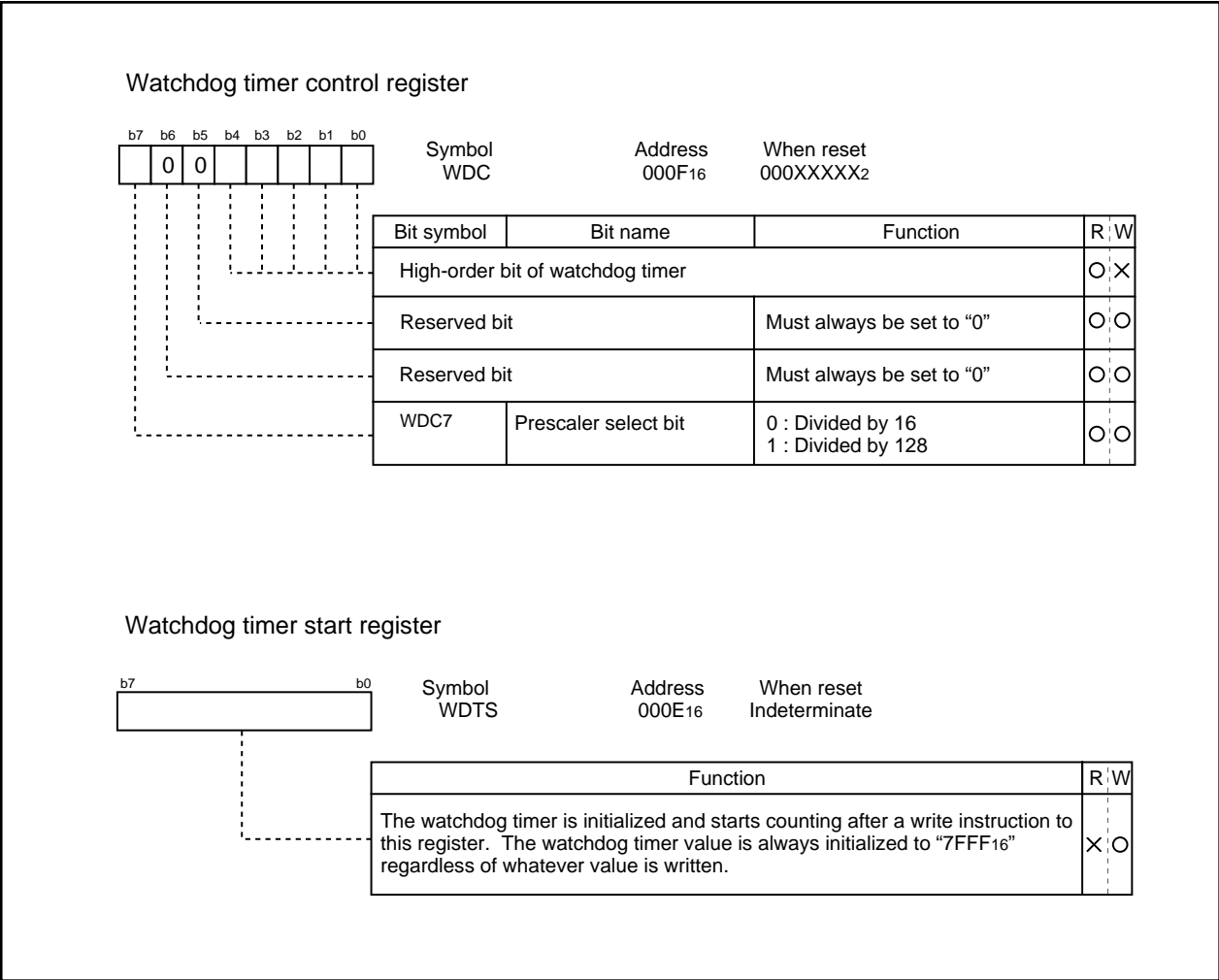


Figure 2.8.2 Watchdog timer control and start registers

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.9 DMAC

This microcomputer has two DMAC (direct memory access controller) channels that allow data to be sent to memory without using the CPU. DMAC shares the same data bus with the CPU. The DMAC is given a higher right of using the bus than the CPU, which leads to working the cycle stealing method. On this account, the operation from the occurrence of DMA transfer request signal to the completion of 1-word (16-bit) or 1-byte (8-bit) data transfer can be performed at high speed. Figure 2.9.1 shows the block diagram of the DMAC. Table 2.9.1 shows the DMAC specifications. Figures 2.9.2 to 2.9.4 show the registers used by the DMAC.

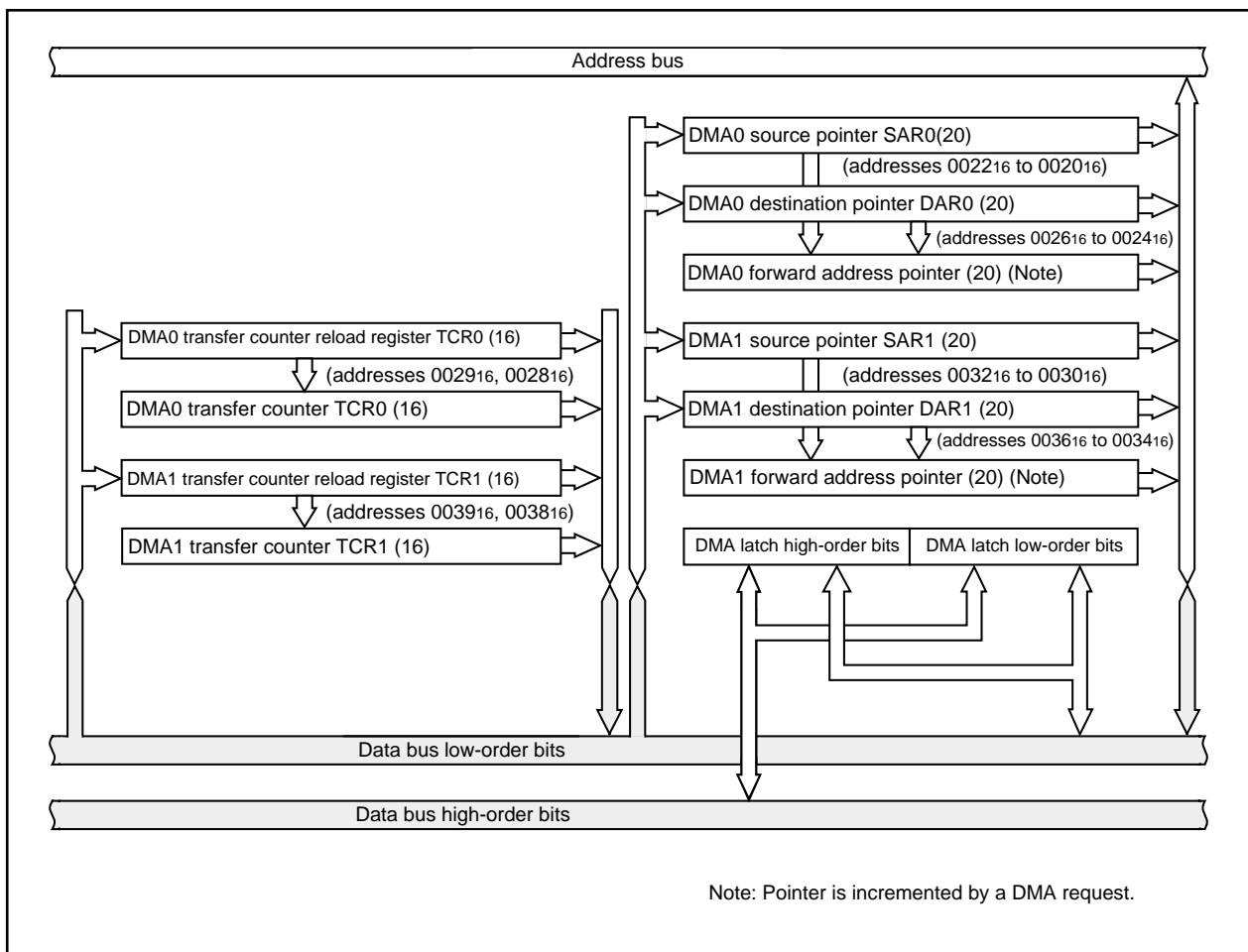


Figure 2.9.1 Block diagram of DMAC

Either a write signal to the software DMA request bit or an interrupt request signal is used as a DMA transfer request signal. But the DMA transfer is affected neither by the interrupt enable flag (I flag) nor by the interrupt priority level. The DMA transfer doesn't affect any interrupts either.

If the DMAC is active (the DMA enable bit is set to 1), data transfer starts every time a DMA transfer request signal occurs. If the cycle of the occurrences of DMA transfer request signals is higher than the DMA transfer cycle, there can be instances in which the number of transfer requests doesn't agree with the number of transfers. For details, see the description of the DMA request bit.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Table 2.9.1 DMAC specifications**

Item	Specification
No. of channels	2 (cycle steal method)
Transfer memory space	<ul style="list-style-type: none"> <li>• From any address in the 1M bytes space to a fixed address</li> <li>• From a fixed address to any address in the 1M bytes space</li> <li>• From a fixed address to a fixed address</li> </ul> (Note that DMA-related registers [0020 <sub>16</sub> to 003F <sub>16</sub> ] cannot be accessed)
Maximum No. of bytes transferred	128K bytes (with 16-bit transfers) or 64K bytes (with 8-bit transfers)
DMA request factors (Note)	Falling edge of INT0 or INT1 (INT0 can be selected by DMA0, INT1 by DMA1) or both edge Timer A0 to timer A4 interrupt requests Timer B0 to timer B5 interrupt requests UART0 transfer and reception interrupt requests UART1 transfer and reception interrupt requests UART2 transfer and reception interrupt requests Serial I/O3, 4 interrupt requests A-D conversion interrupt requests Software triggers
Channel priority	DMA0 takes precedence if DMA0 and DMA1 requests are generated simultaneously
Transfer unit	8 bits or 16 bits
Transfer address direction	forward/fixed (forward direction cannot be specified for both source and destination simultaneously)
Transfer mode	<ul style="list-style-type: none"> <li>• Single transfer mode After the transfer counter underflows, the DMA enable bit turns to "0", and the DMAC turns inactive</li> <li>• Repeat transfer mode After the transfer counter underflows, the value of the transfer counter reload register is reloaded to the transfer counter. The DMAC remains active unless a "0" is written to the DMA enable bit.</li> </ul>
DMA interrupt request generation timing	When an underflow occurs in the transfer counter
Active	When the DMA enable bit is set to "1", the DMAC is active. When the DMAC is active, data transfer starts every time a DMA transfer request signal occurs.
Inactive	<ul style="list-style-type: none"> <li>• When the DMA enable bit is set to "0", the DMAC is inactive.</li> <li>• After the transfer counter underflows in single transfer mode</li> </ul>
Forward address pointer and reload timing for transfer counter	At the time of starting data transfer immediately after turning the DMAC active, the value of one of source pointer and destination pointer - the one specified for the forward direction - is reloaded to the forward direction address pointer, and the value of the transfer counter reload register is reloaded to the transfer counter.
Writing to register	Registers specified for forward direction transfer are always write enabled. Registers specified for fixed address transfer are write-enabled when the DMA enable bit is "0".
Reading the register	Can be read at any time. However, when the DMA enable bit is "1", reading the register set up as the forward register is the same as reading the value of the forward address pointer.

Note: DMA transfer is not effective to any interrupt. DMA transfer is affected neither by the interrupt enable flag (I flag) nor by the interrupt priority level.



M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

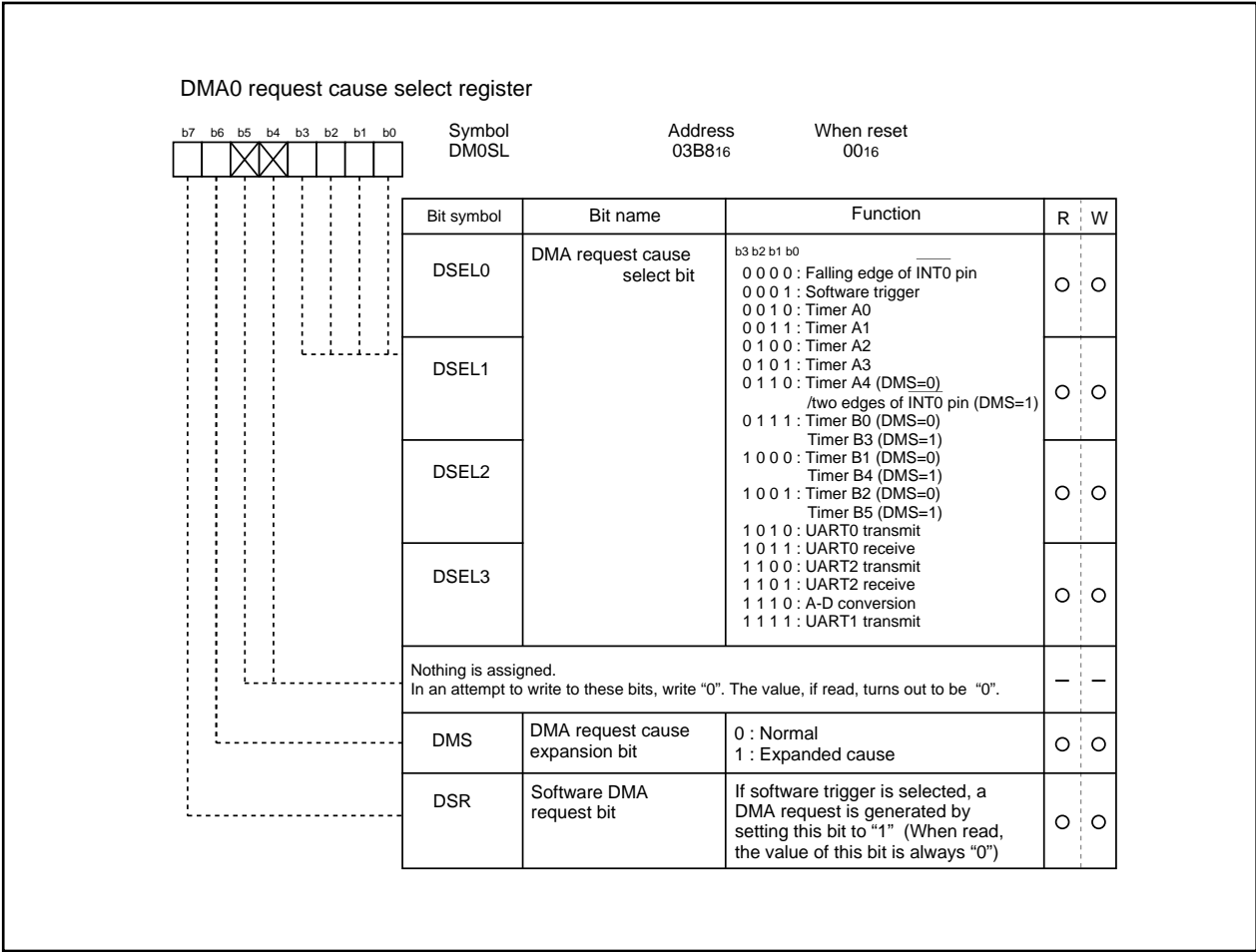


Figure 2.9.2 DMAC register (1)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

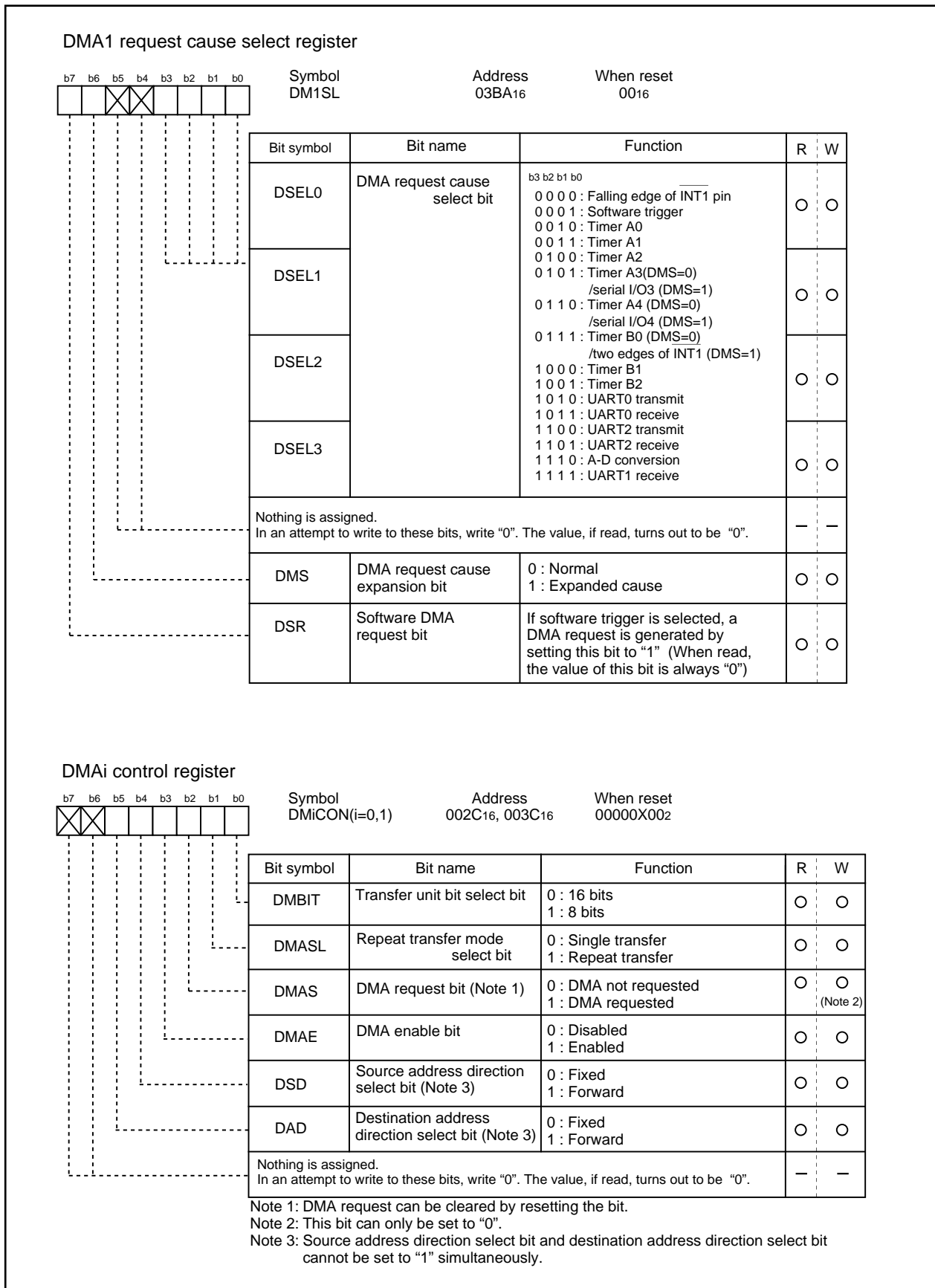


Figure 2.9.3 DMAC register (2)

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

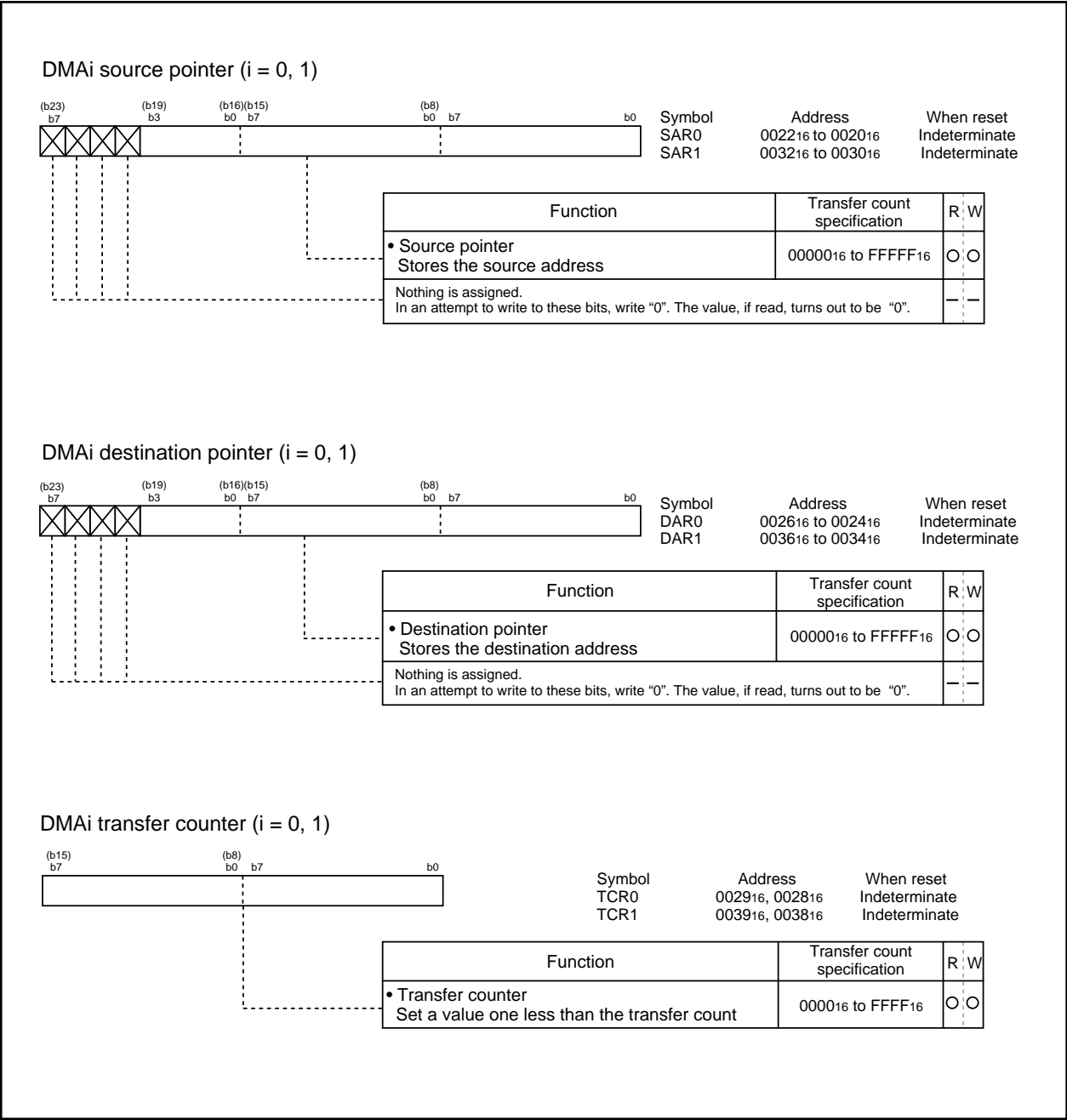


Figure 2.9.4 DMAC register (3)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

---

## (1) Transfer cycle

The transfer cycle consists of the bus cycle in which data is read from memory or from the SFR area (source read) and the bus cycle in which the data is written to memory or to the SFR area (destination write). The number of read and write bus cycles depends on the source and destination addresses and, the level of the BYTE pin. Also, the bus cycle itself is longer when software waits are inserted.

### (a) Effect of source and destination addresses

When 16-bit data is transferred on a 16-bit data bus, and the source and destination both start at odd addresses, there are one more source read cycle and destination write cycle than when the source and destination both start at even addresses.

### (b) Effect of BYTE pin level

When transferring 16-bit data over an 8-bit data bus (BYTE pin = "H"), the 16 bits of data are sent in two 8-bit blocks. Therefore, two bus cycles are required for reading the data and two are required for writing the data. Also, in contrast to when the CPU accesses internal memory, when the DMAC accesses internal memory (internal RAM, and SFR), these areas are accessed using the data size selected by the BYTE pin.

### (c) Effect of software wait

When the SFR area or a memory area with a software wait is accessed, the number of cycles is increased for the wait by 1 bus cycle. The length of the cycle is determined by BCLK.

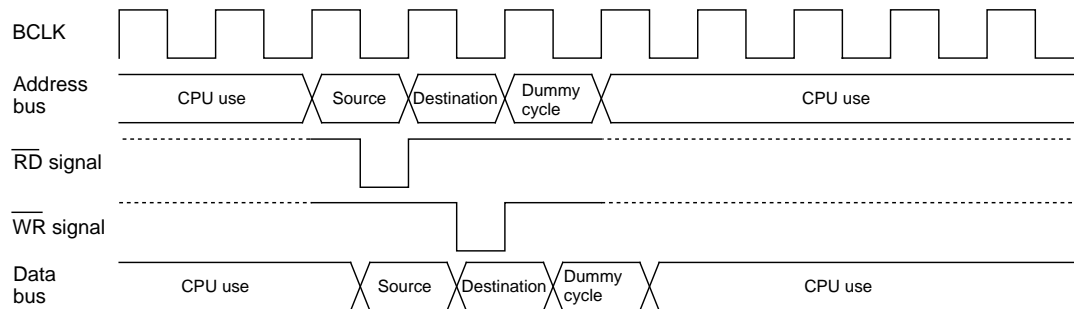
Figure 2.9.5 shows the example of the transfer cycles for a source read. For convenience, the destination write cycle is shown as one cycle and the source read cycles for the different conditions are shown. In reality, the destination write cycle is subject to the same conditions as the source read cycle, with the transfer cycle changing accordingly. When calculating the transfer cycle, remember to apply the respective conditions to both the destination write cycle and the source read cycle. For example (2) in Figure 2.9.5, if data is being transferred in 16-bit units on an 8-bit bus, two bus cycles are required for both the source read cycle and the destination write cycle.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

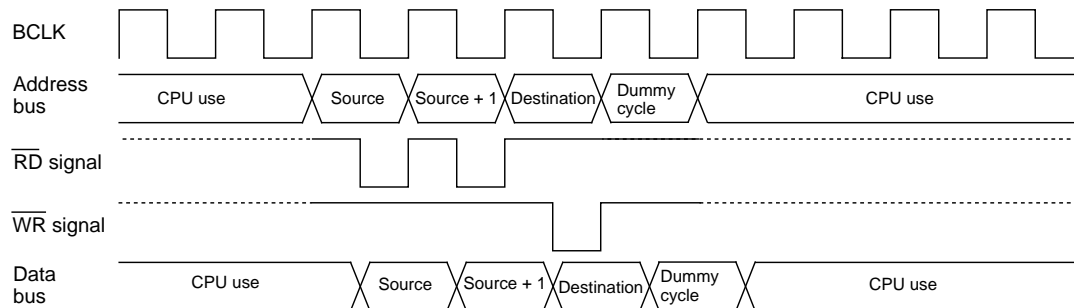
## (1) 8-bit transfers

16-bit transfers from even address and the source address is even.

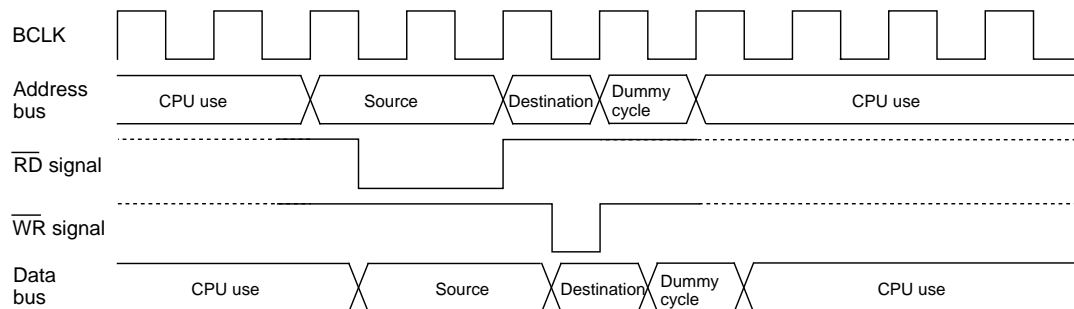


## (2) 16-bit transfers and the source address is odd

Transferring 16-bit data on an 8-bit data bus (In this case, there are also two destination write cycles).

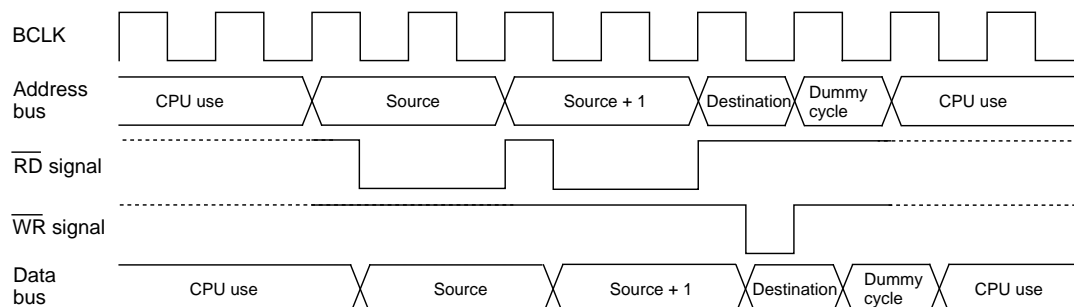


## (3) One wait is inserted into the source read under the conditions in (1)



## (4) One wait is inserted into the source read under the conditions in (2)

(When 16-bit data is transferred on an 8-bit data bus, there are two destination write cycles).



Note: The same timing changes occur with the respective conditions at the destination as at the source.

Figure 2.9.5 Example of the transfer cycles for a source read

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (2) DMAC transfer cycles

Any combination of even or odd transfer read and write addresses is possible. Table 2.9.2 shows the number of DMAC transfer cycles.

The number of DMAC transfer cycles can be calculated as follows:

$$\text{No. of transfer cycles per transfer unit} = \text{No. of read cycles} \times j + \text{No. of write cycles} \times k$$

**Table 2.9.2 No. of DMAC transfer cycles**

Transfer unit	Bus width	Access address	Microprocessor mode	
			No. of read cycles	No. of write cycles
8-bit transfers (DMBIT= "1")	16-bit (BYTE= "L")	Even	1	1
		Odd	1	1
	8-bit (BYTE = "H")	Even	1	1
		Odd	1	1
16-bit transfers (DMBIT= "0")	16-bit (BYTE = "L")	Even	1	1
		Odd	2	2
	8-bit (BYTE = "H")	Even	2	2
		Odd	2	2

### Coefficient j, k

Internal memory			External memory		
Internal RAM No wait	Internal RAM With wait	SFR area	Separate bus No wait	Separate bus With wait	Multiplex bus
1	2	2	1	2	3

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.9.1 DMA enable bit

Setting the DMA enable bit to "1" makes the DMAC active. The DMAC carries out the following operations at the time data transfer starts immediately after DMAC is turned active.

- (1) Reloads the value of one of the source pointer and the destination pointer - the one specified for the forward direction - to the forward direction address pointer.
- (2) Reloads the value of the transfer counter reload register to the transfer counter.

Thus overwriting "1" to the DMA enable bit with the DMAC being active carries out the operations given above, so the DMAC operates again from the initial state at the instant "1" is overwritten to the DMA enable bit.

## 2.9.2 DMA request bit

The DMAC can generate a DMA transfer request signal triggered by a factor chosen in advance out of DMA request factors for each channel.

DMA request factors include the following.

- \* Factors effected by using the interrupt request signals from the built-in peripheral functions and software DMA factors (internal factors) effected by a program.
- \* External factors effected by utilizing the input from external interrupt signals.

For the selection of DMA request factors, see the descriptions of the DMA<sub>i</sub> factor selection register. The DMA request bit turns to "1" if the DMA transfer request signal occurs regardless of the DMAC's state (regardless of whether the DMA enable bit is set "1" or to "0"). It turns to "0" immediately before data transfer starts.

In addition, it can be set to "0" by use of a program, but cannot be set to "1".

There can be instances in which a change in DMA request factor selection bit causes the DMA request bit to turn to "1". So be sure to set the DMA request bit to "0" after the DMA request factor selection bit is changed.

The DMA request bit turns to "1" if a DMA transfer request signal occurs, and turns to "0" immediately before data transfer starts. If the DMAC is active, data transfer starts immediately, so the value of the DMA request bit, if read by use of a program, turns out to be "0" in most cases. To examine whether the DMAC is active, read the DMA enable bit.

Here follows the timing of changes in the DMA request bit.

### (1) Internal factors

Except the DMA request factors triggered by software, the timing for the DMA request bit to turn to "1" due to an internal factor is the same as the timing for the interrupt request bit of the interrupt control register to turn to "1" due to several factors.

Turning the DMA request bit to "1" due to an internal factor is timed to be effected immediately before the transfer starts.

### (2) External factors

An external factor is a factor caused to occur by the leading edge of input from the INT<sub>i</sub> pin (i depends on which DMAC channel is used).

Selecting the INT<sub>i</sub> pins as external factors using the DMA request factor selection bit causes input from these pins to become the DMA transfer request signals.

The timing for the DMA request bit to turn to "1" when an external factor is selected synchronizes with the signal's edge applicable to the function specified by the DMA request factor selection bit (synchronizes with the trailing edge of the input signal to each INT<sub>i</sub> pin, for example).

With an external factor selected, the DMA request bit is timed to turn to "0" immediately before data transfer starts similarly to the state in which an internal factor is selected.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

### (3) The priorities of channels and DMA transfer timing

If a DMA transfer request signal falls on a single sampling cycle (a sampling cycle means one period from the leading edge to the trailing edge of BCLK), the DMA request bits of applicable channels concurrently turn to "1". If the channels are active at that moment, DMA0 is given a high priority to start data transfer. When DMA0 finishes data transfer, it gives the bus right to the CPU. When the CPU finishes single bus access, then DMA1 starts data transfer and gives the bus right to the CPU.

An example in which DMA transfer is carried out in minimum cycles at the time when DMA transfer request signals due to external factors concurrently occur.

Figure 2.9.6 An example of DMA transfer effected by external factors.

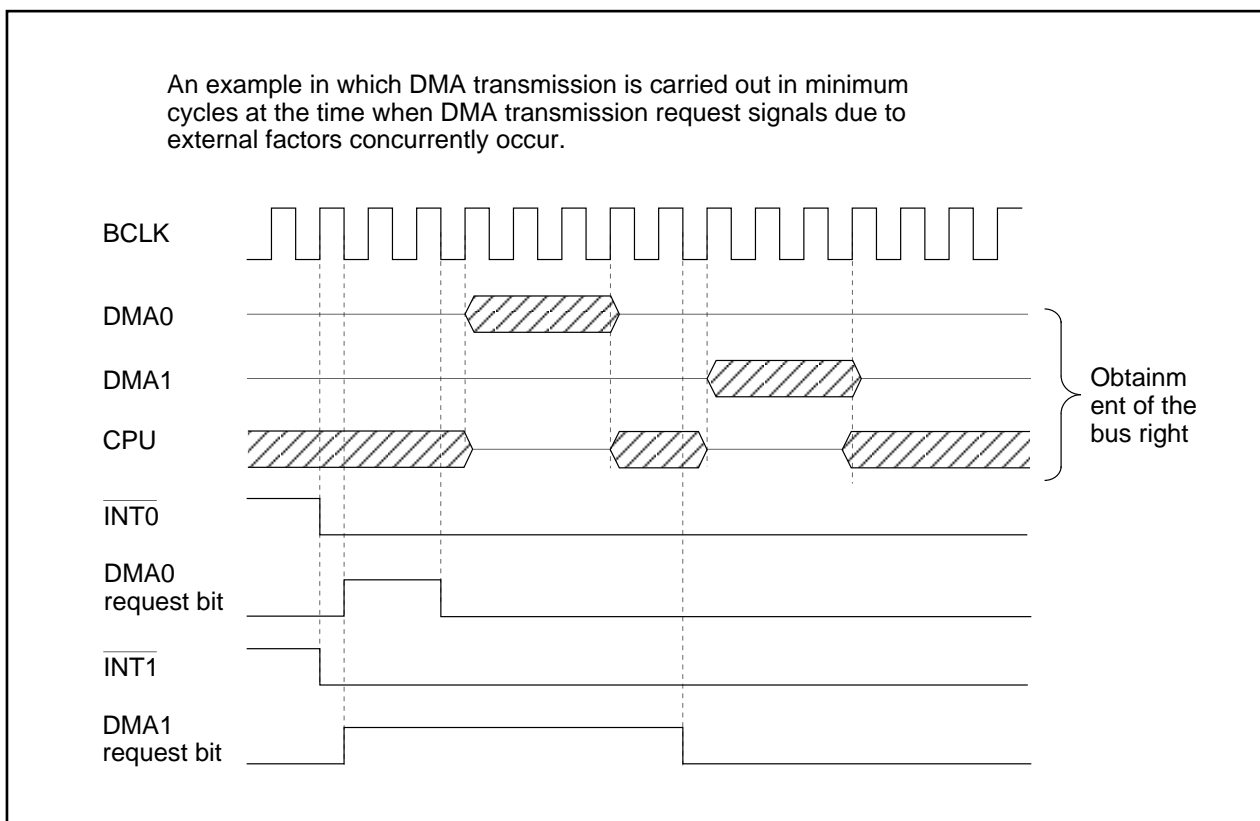


Figure 2.9.6 An example of DMA transfer effected by external factors



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.10 Timer

There are eleven 16-bit timers. These timers can be classified by function into timers A (five) and timers B (six). All these timers function independently.

Figures 2.10.1 and 2.10.2 show the block diagram of timers.

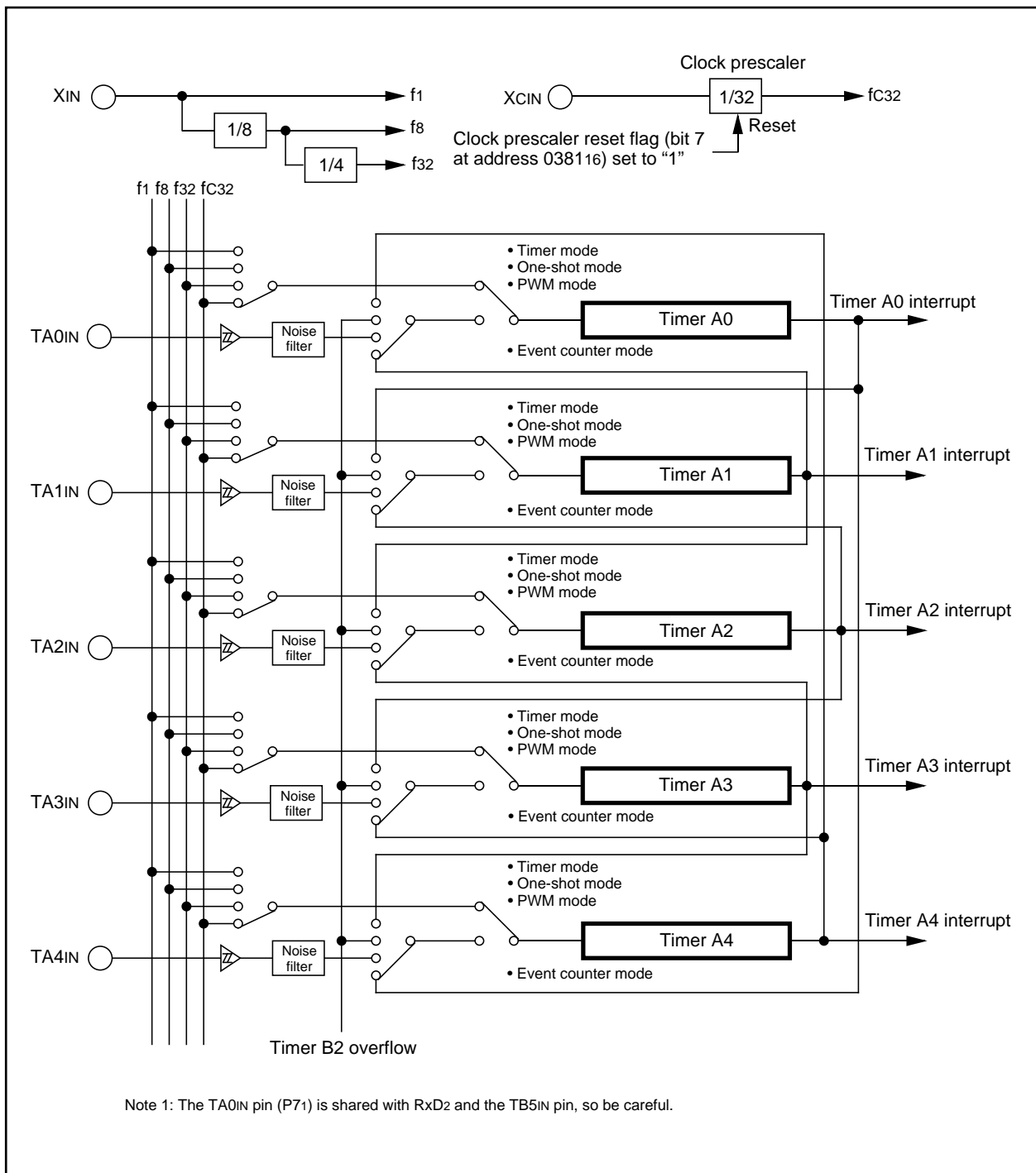


Figure 2.10.1 Timer A block diagram

## M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

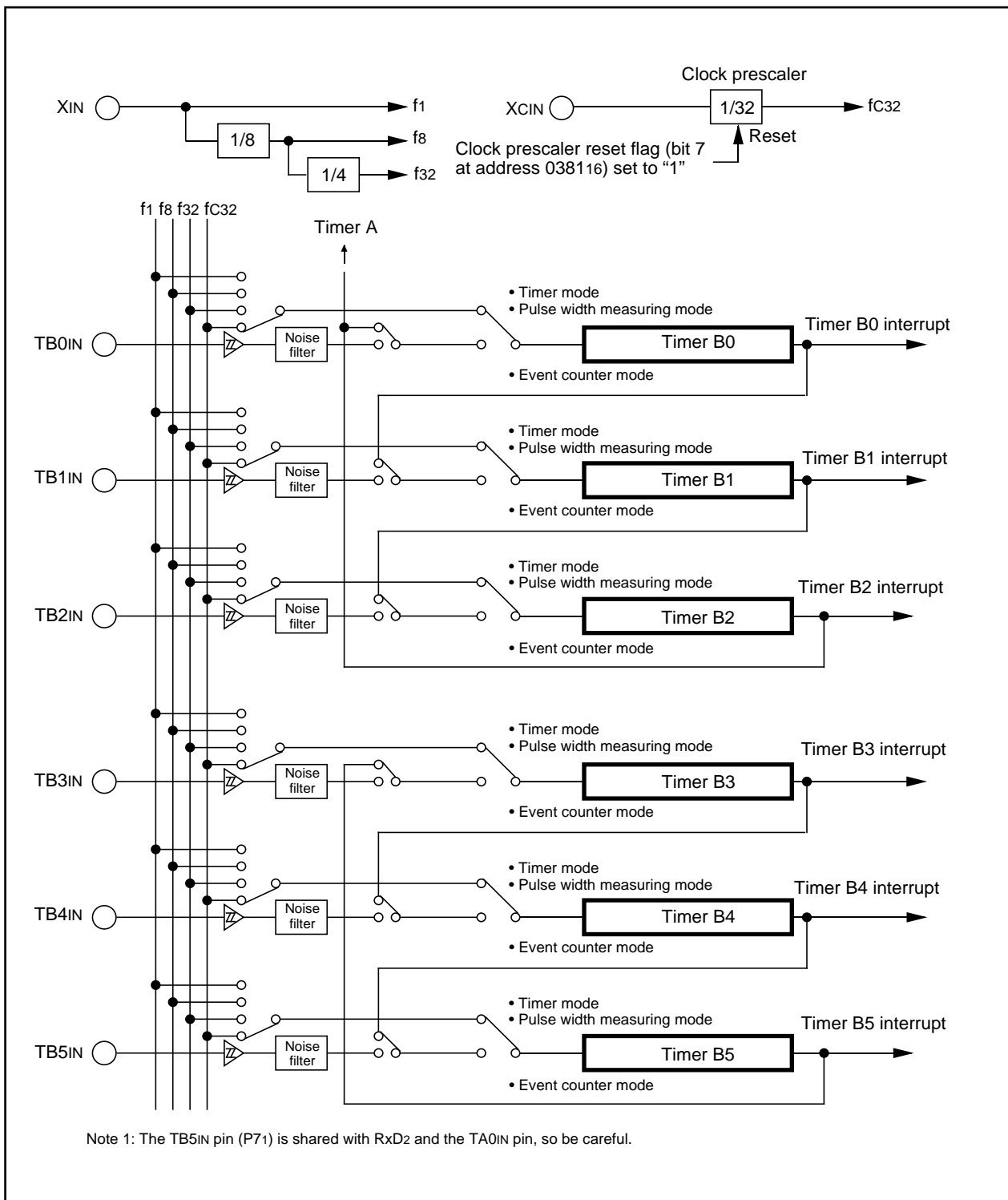


Figure 2.10.2 Timer B block diagram

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

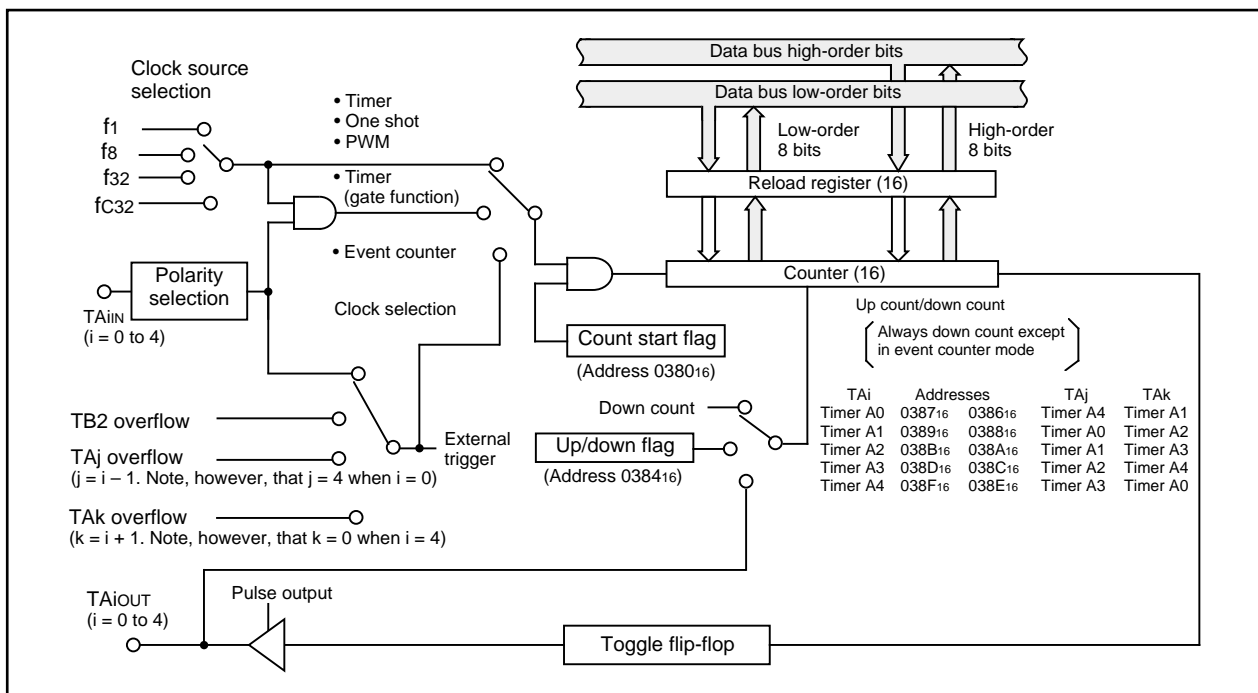
### 2.10.1 Timer A

Figure 2.10.3 shows the block diagram of timer A. Figures 2.10.4 to 2.10.6 show the timer A-related registers.

Except in event counter mode, timers A0 through A4 all have the same function. Use the timer Ai mode register (i = 0 to 4) bits 0 and 1 to choose the desired mode.

Timer A has the four operation modes listed as follows:

- **Timer mode:** The timer counts an internal count source.
- **Event counter mode:** The timer counts pulses from an external source or a timer over flow.
- **One-shot timer mode:** The timer stops counting when the count reaches "000016".
- **Pulse width modulation (PWM) mode:** The timer outputs pulses of a given width.



**Figure 2.10.3 Block diagram of timer A**

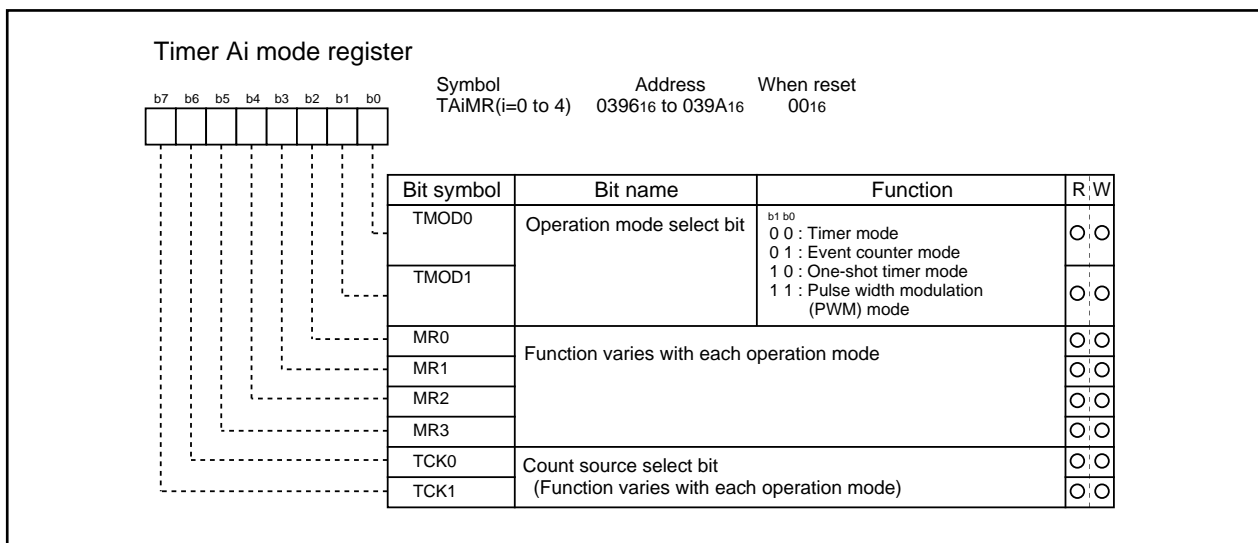
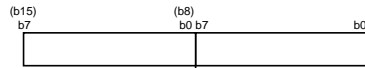


Figure 2.10.4 Timer A-related registers (1)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Timer Ai register (Note)



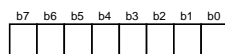
Symbol	Address	When reset
TA0	0387 <sub>16</sub> , 0386 <sub>16</sub>	Indeterminate
TA1	0389 <sub>16</sub> , 0388 <sub>16</sub>	Indeterminate
TA2	038B <sub>16</sub> , 038A <sub>16</sub>	Indeterminate
TA3	038D <sub>16</sub> , 038C <sub>16</sub>	Indeterminate
TA4	038F <sub>16</sub> , 038E <sub>16</sub>	Indeterminate

Function	Values that can be set	R/W
• Timer mode Counts an internal count source	0000 <sub>16</sub> to FFFF <sub>16</sub>	○:○
• Event counter mode Counts pulses from an external source or timer overflow	0000 <sub>16</sub> to FFFF <sub>16</sub>	○:○
• One-shot timer mode Counts a one shot width	0000 <sub>16</sub> to FFFF <sub>16</sub>	×:○
• Pulse width modulation mode (16-bit PWM) Functions as a 16-bit pulse width modulator	0000 <sub>16</sub> to FFFE <sub>16</sub>	×:○
• Pulse width modulation mode (8-bit PWM) Timer low-order address functions as an 8-bit prescaler and high-order address functions as an 8-bit pulse width modulator	00 <sub>16</sub> to FE <sub>16</sub> (Both high-order and low-order addresses)	×:○

Note 1: Read and write data in 16-bit units.

Note 2: In the case of using "Event counter mode" as "Free-Run type", the timer register contents may be unknown when counting begins. (Refer 3. Usage Precaution.)

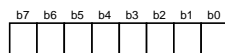
## Count start flag



Symbol	Address	When reset
TABSR	0380 <sub>16</sub>	00 <sub>16</sub>

Bit symbol	Bit name	Function	R/W
TA0S	Timer A0 count start flag	0 : Stops counting 1 : Starts counting	○:○
TA1S	Timer A1 count start flag		○:○
TA2S	Timer A2 count start flag		○:○
TA3S	Timer A3 count start flag		○:○
TA4S	Timer A4 count start flag		○:○
TB0S	Timer B0 count start flag		○:○
TB1S	Timer B1 count start flag		○:○
TB2S	Timer B2 count start flag		○:○

## Up/down flag



Symbol	Address	When reset
UDF	0384 <sub>16</sub>	00 <sub>16</sub>

Bit symbol	Bit name	Function	R/W
TA0UD	Timer A0 up/down flag	0 : Down count 1 : Up count	○:○
TA1UD	Timer A1 up/down flag		○:○
TA2UD	Timer A2 up/down flag	This specification becomes valid when the up/down flag content is selected for up/down switching cause	○:○
TA3UD	Timer A3 up/down flag		○:○
TA4UD	Timer A4 up/down flag		○:○
TA2P	Timer A2 two-phase pulse signal processing select bit	0 : two-phase pulse signal processing disabled 1 : two-phase pulse signal processing enabled	×:○
TA3P	Timer A3 two-phase pulse signal processing select bit		×:○
TA4P	Timer A4 two-phase pulse signal processing select bit	When not using the two-phase pulse signal processing function, set the select bit to "0"	×:○

Figure 2.10.5 Timer A-related registers (2)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

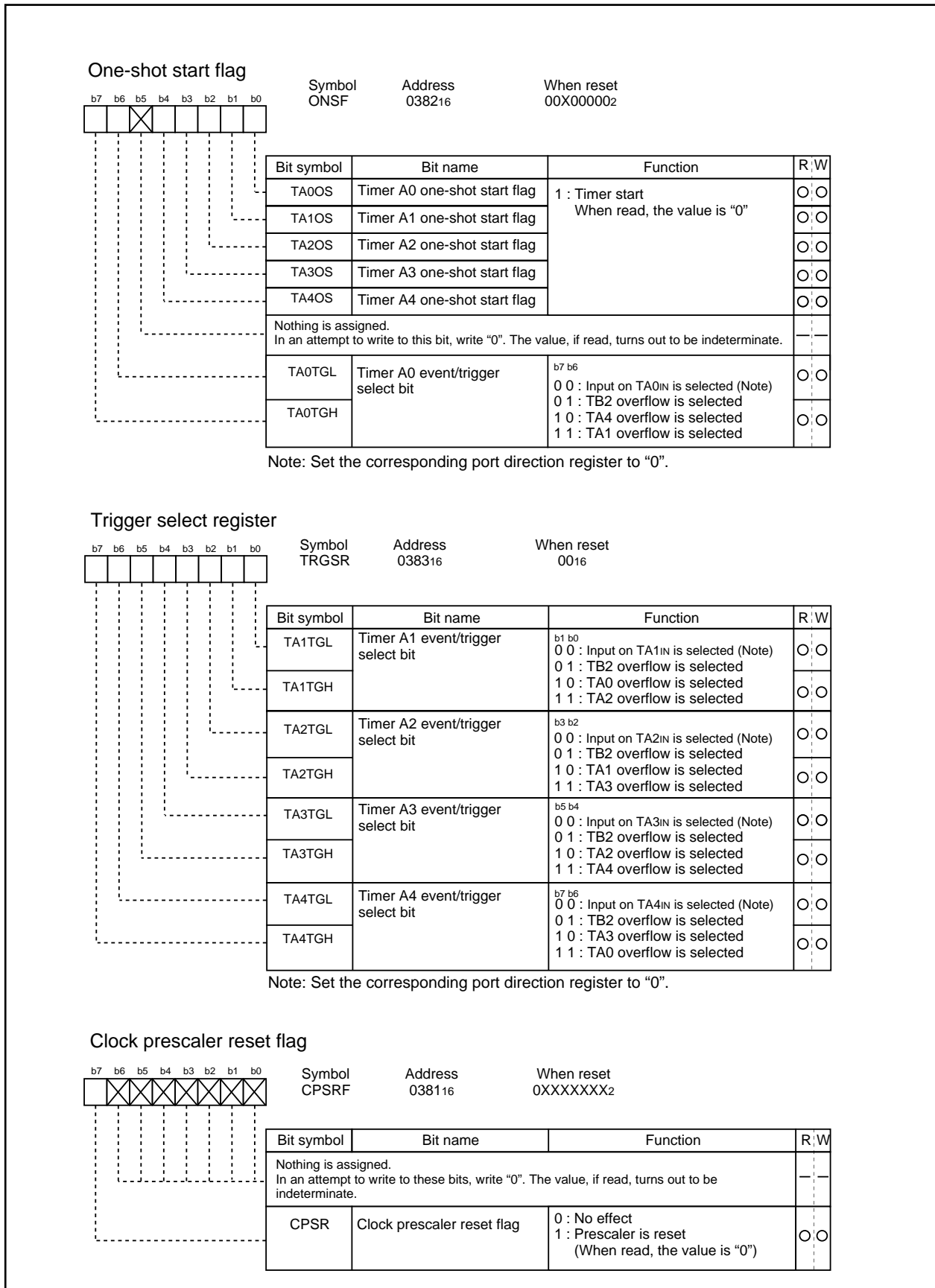


Figure 2.10.6 Timer A-related registers (3)

# M306H1SFP

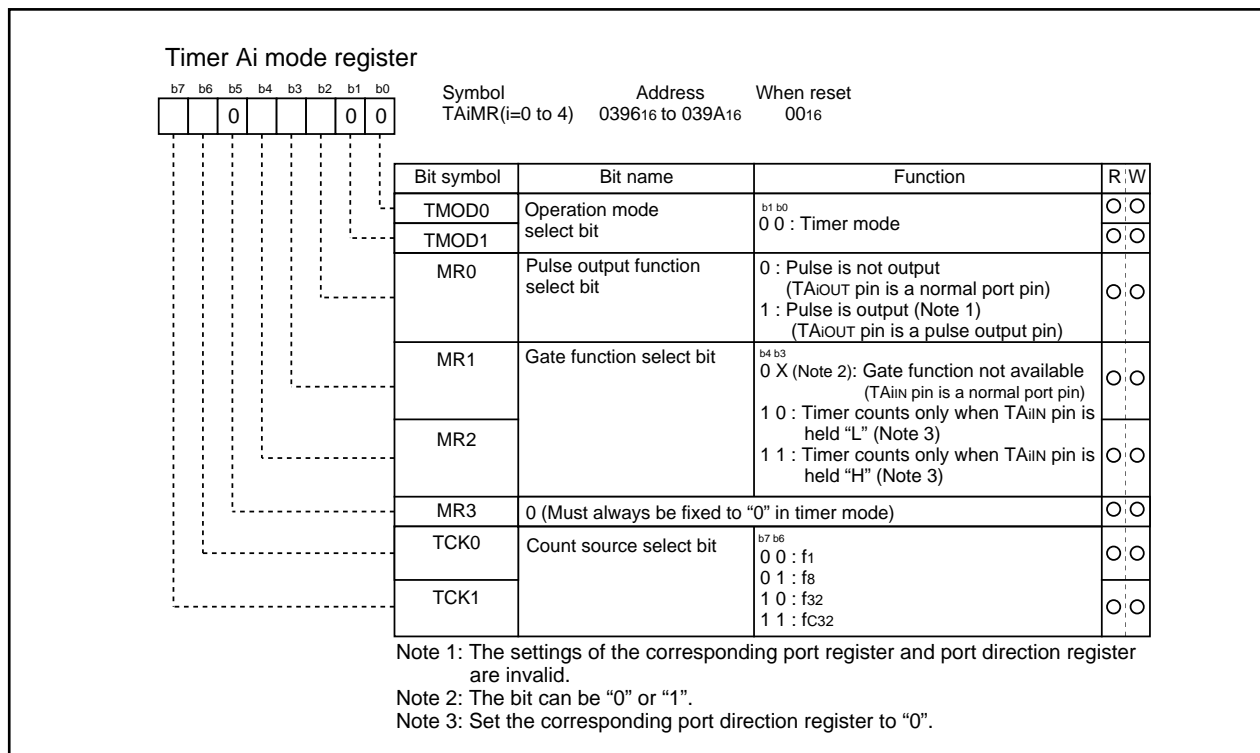
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (1) Timer mode

In this mode, the timer counts an internally generated count source. (See Table 2.10.1) Figure 2.10.7 shows the timer Ai mode register in timer mode.

**Table 2.10.1 Specifications of timer mode**

Item	Specification
Count source	f <sub>1</sub> , f <sub>8</sub> , f <sub>32</sub> , f <sub>C32</sub>
Count operation	<ul style="list-style-type: none"> <li>Down count</li> <li>When the timer underflows, it reloads the reload register contents before continuing counting</li> </ul>
Divide ratio	1/(n+1)    n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	When the timer underflows
TAiIN pin function	Programmable I/O port or gate input
TAiOUT pin function	Programmable I/O port or pulse output
Read from timer	Count value can be read out by reading timer Ai register
Write to timer	<ul style="list-style-type: none"> <li>When counting stopped When a value is written to timer Ai register, it is written to both reload register and counter</li> <li>When counting in progress When a value is written to timer Ai register, it is written to only reload register (Transferred to counter at next reload time)</li> </ul>
Select function	<ul style="list-style-type: none"> <li>Gate function Counting can be started and stopped by the TAiIN pin's input signal</li> <li>Pulse output function Each time the timer underflows, the TAiOUT pin's polarity is reversed</li> </ul>



**Figure 2.10.7 Timer Ai mode register in timer mode**

## M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**(2) Event counter mode**

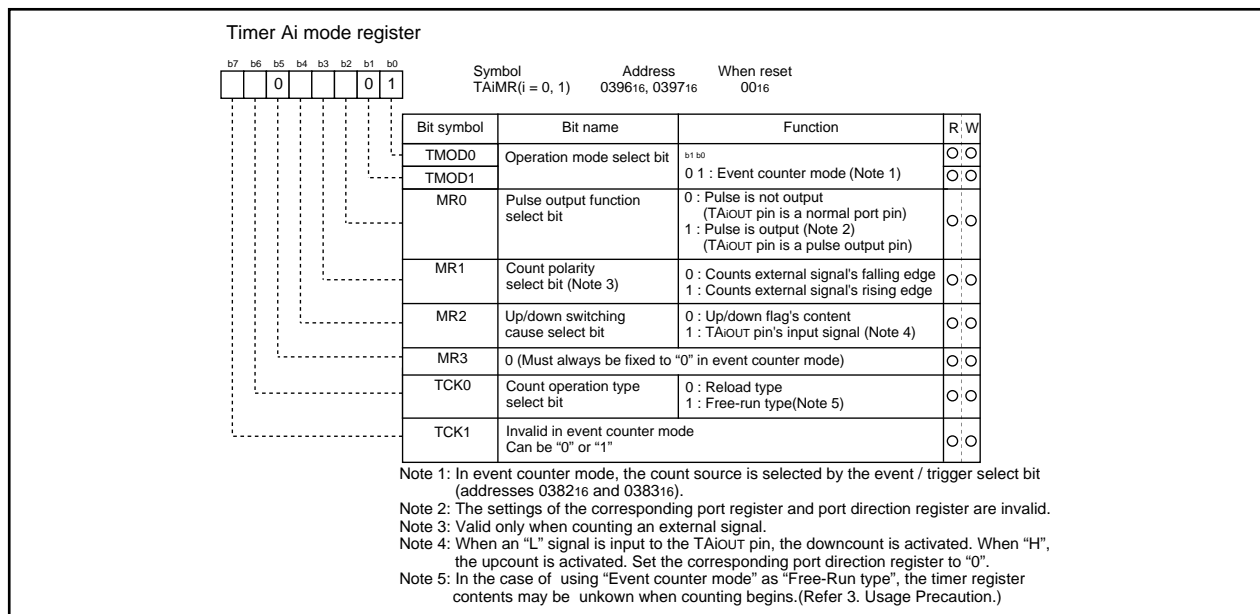
In this mode, the timer counts an external signal or an internal timer's overflow. Timers A0 and A1 can count a single-phase external signal. Timers A2, A3, and A4 can count a single-phase and a two-phase external signal. Table 2.10.2 lists timer specifications when counting a single-phase external signal. Figure 2.10.8 shows the timer Ai mode register in event counter mode.

Table 2.10.3 lists timer specifications when counting a two-phase external signal. Figure 2.10.9 shows the timer Ai mode register in event counter mode.

**Table 2.10.2 Timer specifications in event counter mode (when not processing two-phase pulse signal)**

Item	Specification
Count source	<ul style="list-style-type: none"> <li>External signals input to TAIin pin (effective edge can be selected by software)</li> <li>TB2 overflow, TAJ overflow</li> </ul>
Count operation	<ul style="list-style-type: none"> <li>Up count or down count can be selected by external signal or software</li> <li>When the timer overflows or underflows, it reloads the reload register contents before continuing counting (Note)</li> </ul>
Divide ratio	$1 / (FFFF_{16} - n + 1)$ for up count $1 / (n + 1)$ for down count      n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	The timer overflows or underflows
TAIin pin function	Programmable I/O port or count source input
TAIOUT pin function	Programmable I/O port, pulse output, or up/down count select input
Read from timer	Count value can be read out by reading timer Ai register
Write to timer	<ul style="list-style-type: none"> <li>When counting stopped When a value is written to timer Ai register, it is written to both reload register and counter</li> <li>When counting in progress When a value is written to timer Ai register, it is written to only reload register (Transferred to counter at next reload time)</li> </ul>
Select function	<ul style="list-style-type: none"> <li>Free-run count function Even when the timer overflows or underflows, the reload register content is not reloaded to it</li> <li>Pulse output function Each time the timer overflows or underflows, the TAIOUT pin's polarity is reversed</li> </ul>

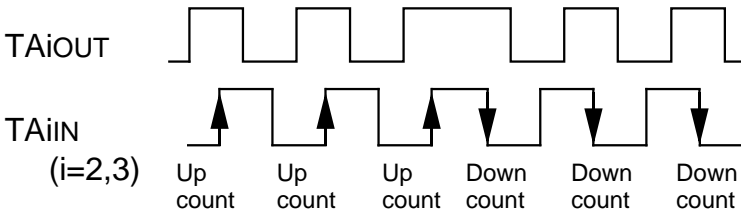
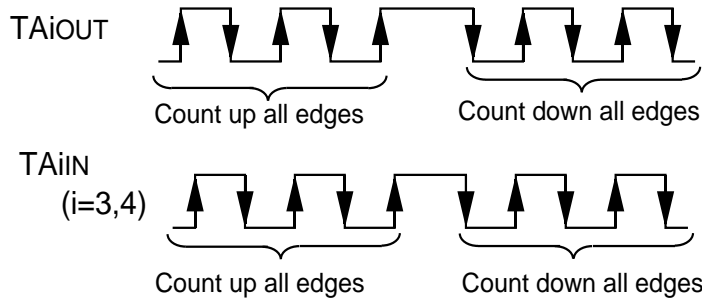
Note: This does not apply when the free-run function is selected.

**Figure 2.10.8 Timer Ai mode register in event counter mode**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Table 2.10.3 Timer specifications in event counter mode (when processing two-phase pulse signal with timers A2, A3, and A4)

Item	Specification
Count source	<ul style="list-style-type: none"> <li>Two-phase pulse signals input to TAIIN or TAIOUT pin</li> </ul>
Count operation	<ul style="list-style-type: none"> <li>Up count or down count can be selected by two-phase pulse signal</li> <li>When the timer overflows or underflows, the reload register content is reloaded and the timer starts over again (Note)</li> </ul>
Divide ratio	$1 / (FFFF_{16} - n + 1)$ for up count $1 / (n + 1)$ for down count $n$ : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	Timer overflows or underflows
TAiIN pin function	Two-phase pulse input
TAiOUT pin function	Two-phase pulse input
Read from timer	Count value can be read out by reading timer A2, A3, or A4 register
Write to timer	<ul style="list-style-type: none"> <li>When counting stopped When a value is written to timer A2, A3, or A4 register, it is written to both reload register and counter</li> <li>When counting in progress When a value is written to timer A2, A3, or A4 register, it is written to only reload register. (Transferred to counter at next reload time.)</li> </ul>
Select function	<ul style="list-style-type: none"> <li>Normal processing operation The timer counts up rising edges or counts down falling edges on the TAIIN pin when input signal on the TAIOUT pin is "H"</li> </ul>  <ul style="list-style-type: none"> <li>Multiply-by-4 processing operation If the phase relationship is such that the TAIIN pin goes "H" when the input signal on the TAIOUT pin is "H", the timer counts up rising and falling edges on the TAIOUT and TAIIN pins. If the phase relationship is such that the TAIIN pin goes "L" when the input signal on the TAIOUT pin is "H", the timer counts down rising and falling edges on the TAIOUT and TAIIN pins.</li> </ul> 

Note: This does not apply when the free-run function is selected.



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Timer Ai mode register (When not using two-phase pulse signal processing)

Symbol								Address		When reset		
TAiMR(i = 2 to 4)								0398 <sub>16</sub> to 039A <sub>16</sub>		00 <sub>16</sub>		
b7	b6	b5	b4	b3	b2	b1	b0					
		0				0	1					
								Bit symbol	Bit name	Function	R	W
								TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode	○	○
								TMOD1		○	○	
								MR0	Pulse output function select bit	0 : Pulse is not output (TAiOUT pin is a normal port pin) 1 : Pulse is output (Note 1) (TAiOUT pin is a pulse output pin)	○	○
								MR1	Count polarity select bit (Note 2)	0 : Counts external signal's falling edges 1 : Counts external signal's rising edges	○	○
								MR2	Up/down switching cause select bit	0 : Up/down flag's content 1 : TAiOUT pin's input signal (Note 3)	○	○
								MR3	0 : (Must always be "0" in event counter mode)		○	○
								TCK0	Count operation type select bit	0 : Reload type 1 : Free-run type(Note 6)	○	○
								TCK1	Two-phase pulse signal processing operation select bit (Note 4)(Note 5)	0 : Normal processing operation 1 : Multiply-by-4 processing operation	○	○

Note 1: The settings of the corresponding port register and port direction register are invalid.

Note 2: This bit is valid when only counting an external signal.

Note 3: Set the corresponding port direction register to "0".

Note 4: This bit is valid for the timer A3 mode register.

For timer A2 and A4 mode registers, this bit can be "0" or "1".

Note 5: When performing two-phase pulse signal processing, make sure the two-phase pulse signal processing operation select bit (address 0384<sub>16</sub>) is set to "1". Also, always be sure to set the event/trigger select bit (addresses 0382<sub>16</sub> and 0383<sub>16</sub>) to "00".

Note 6: In the case of using "Event counter mode" as "Free-Run type", the timer register contents may be unknown when counting begins.(Refer 3. Usage Precaution.)

## Timer Ai mode register (When using two-phase pulse signal processing)

b7b6b5b4b3b2b1b0								Symbol	Address	When reset		
		0	1	0	0	0	1	TAiMR(i = 2 to 4)	0398 <sub>16</sub> to 039A <sub>16</sub>	00 <sub>16</sub>		
								Bit symbol	Bit name	Function	R	W
								TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode	○	○
								TMOD1			○	○
								MR0	0 (Must always be "0" when using two-phase pulse signal processing)		○	○
								MR1	0 (Must always be "0" when using two-phase pulse signal processing)		○	○
								MR2	1 (Must always be "1" when using two-phase pulse signal processing)		○	○
								MR3	0 (Must always be "0" when using two-phase pulse signal processing)		○	○
								TCK0	Count operation type select bit	0 : Reload type 1 : Free-run type(Note 3)	○	○
								TCK1	Two-phase pulse processing operation select bit (Note 1)(Note 2)	0 : Normal processing operation 1 : Multiply-by-4 processing operation	○	○

Note 1: This bit is valid for timer A3 mode register.

For timer A2 and A4 mode registers, this bit can be "0" or "1".

Note 2: When performing two-phase pulse signal processing, make sure the two-phase pulse signal processing operation select bit (address 0384<sub>16</sub>) is set to "1". Also, always be sure to set the event/trigger select bit (addresses 0382<sub>16</sub> and 0383<sub>16</sub>) to "00".

Note 3: In the case of using "Event counter mode" as "Free-Run type", the timer register contents may be unknown when counting begins.(Refer 3. Usage Precaution.)

Figure 2.10.9 Timer Ai mode register in event counter mode

# M306H1SFP

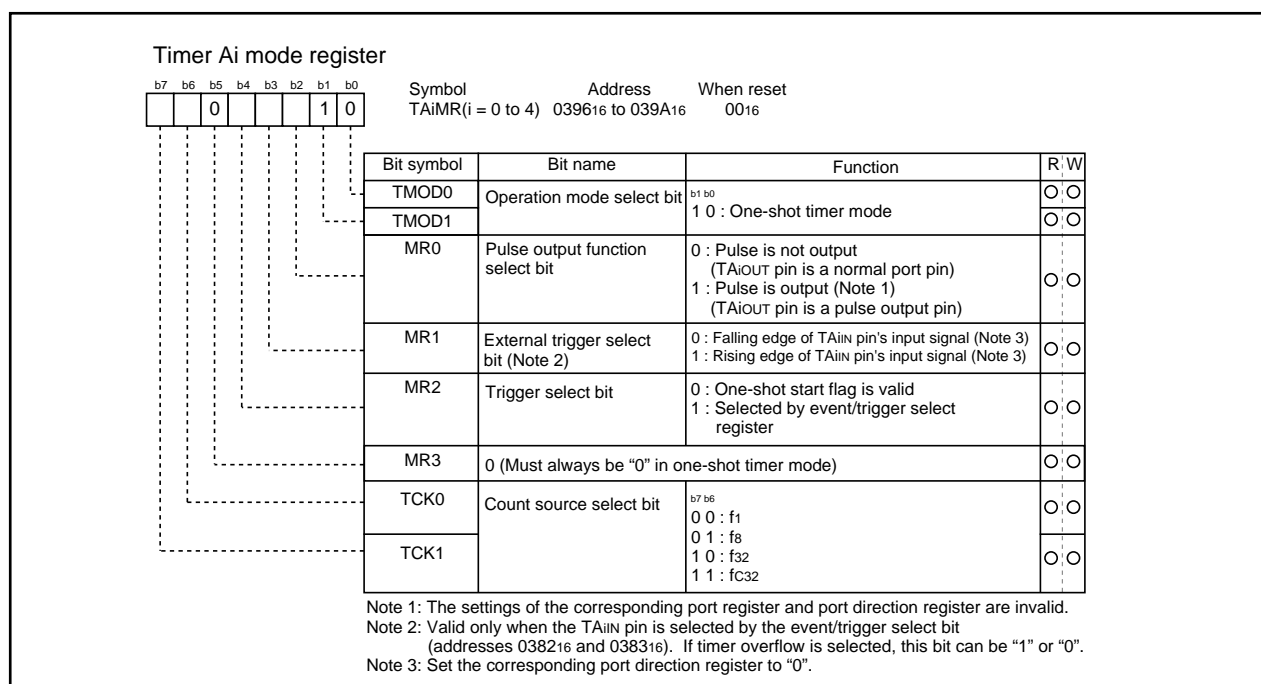
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

### (3) One-shot timer mode

In this mode, the timer operates only once. (See Table 2.10.4) When a trigger occurs, the timer starts up and continues operating for a given period. Figure 2.10.10 shows the timer Ai mode register in one-shot timer mode.

**Table 2.10.4 Timer specifications in one-shot timer mode**

Item	Specification
Count source	f <sub>1</sub> , f <sub>8</sub> , f <sub>32</sub> , f <sub>C32</sub>
Count operation	<ul style="list-style-type: none"> <li>The timer counts down</li> <li>When the count reaches 0000<sub>16</sub>, the timer stops counting after reloading a new count</li> <li>If a trigger occurs when counting, the timer reloads a new count and restarts counting</li> </ul>
Divide ratio	1/n    n : Set value
Count start condition	<ul style="list-style-type: none"> <li>An external trigger is input</li> <li>The timer overflows</li> <li>The one-shot start flag is set (= 1)</li> </ul>
Count stop condition	<ul style="list-style-type: none"> <li>A new count is reloaded after the count has reached 0000<sub>16</sub></li> <li>The count start flag is reset (= 0)</li> </ul>
Interrupt request generation timing	The count reaches 0000 <sub>16</sub>
TAiIN pin function	Programmable I/O port or trigger input
TAiOUT pin function	Programmable I/O port or pulse output
Read from timer	When timer Ai register is read, it indicates an indeterminate value
Write to timer	<ul style="list-style-type: none"> <li>When counting stopped When a value is written to timer Ai register, it is written to both reload register and counter</li> <li>When counting in progress When a value is written to timer Ai register, it is written to only reload register (Transferred to counter at next reload time)</li> </ul>



**Figure 2.10.10 Timer Ai mode register in one-shot timer mode**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

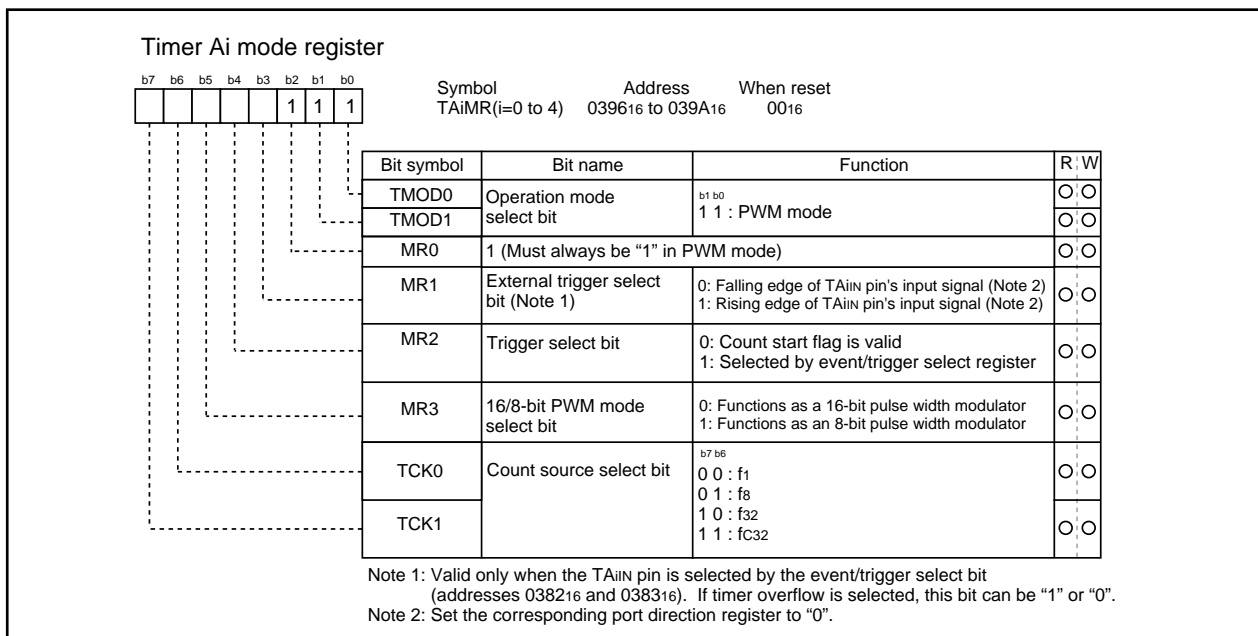
## (4) Pulse width modulation (PWM) mode

In this mode, the timer outputs pulses of a given width in succession. (See Table 2.10.5) In this mode, the counter functions as either a 16-bit pulse width modulator or an 8-bit pulse width modulator.

Figure 2.10.11 shows the timer Ai mode register in pulse width modulation mode. Figure 2.10.12 shows the example of how a 16-bit pulse width modulator operates. Figure 2.10.13 shows the example of how an 8-bit pulse width modulator operates.

**Table 2.10.5 Timer specifications in pulse width modulation mode**

Item	Specification
Count source	f <sub>1</sub> , f <sub>8</sub> , f <sub>32</sub> , f <sub>C32</sub>
Count operation	<ul style="list-style-type: none"> <li>The timer counts down (operating as an 8-bit or a 16-bit pulse width modulator)</li> <li>The timer reloads a new count at a rising edge of PWM pulse and continues counting</li> <li>The timer is not affected by a trigger that occurs when counting</li> </ul>
16-bit PWM	<ul style="list-style-type: none"> <li>High level width <math>n / f_i</math> <math>n</math>: Set value</li> <li>Cycle time <math>(2^{16}-1) / f_i</math> fixed</li> </ul>
8-bit PWM	<ul style="list-style-type: none"> <li>High level width <math>n \times (m+1) / f_i</math> <math>n</math>: values set to timer Ai register's high-order address</li> <li>Cycle time <math>(2^8-1) \times (m+1) / f_i</math> <math>m</math>: values set to timer Ai register's low-order address</li> </ul>
Count start condition	<ul style="list-style-type: none"> <li>External trigger is input</li> <li>The timer overflows</li> <li>The count start flag is set (= 1)</li> </ul>
Count stop condition	<ul style="list-style-type: none"> <li>The count start flag is reset (= 0)</li> </ul>
Interrupt request generation timing	PWM pulse goes "L"
TAiIN pin function	Programmable I/O port or trigger input
TAiOUT pin function	Pulse output
Read from timer	When timer Ai register is read, it indicates an indeterminate value
Write to timer	<ul style="list-style-type: none"> <li>When counting stopped When a value is written to timer Ai register, it is written to both reload register and counter</li> <li>When counting in progress When a value is written to timer Ai register, it is written to only reload register (Transferred to counter at next reload time)</li> </ul>



**Figure 2.10.11 Timer Ai mode register in pulse width modulation mode**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

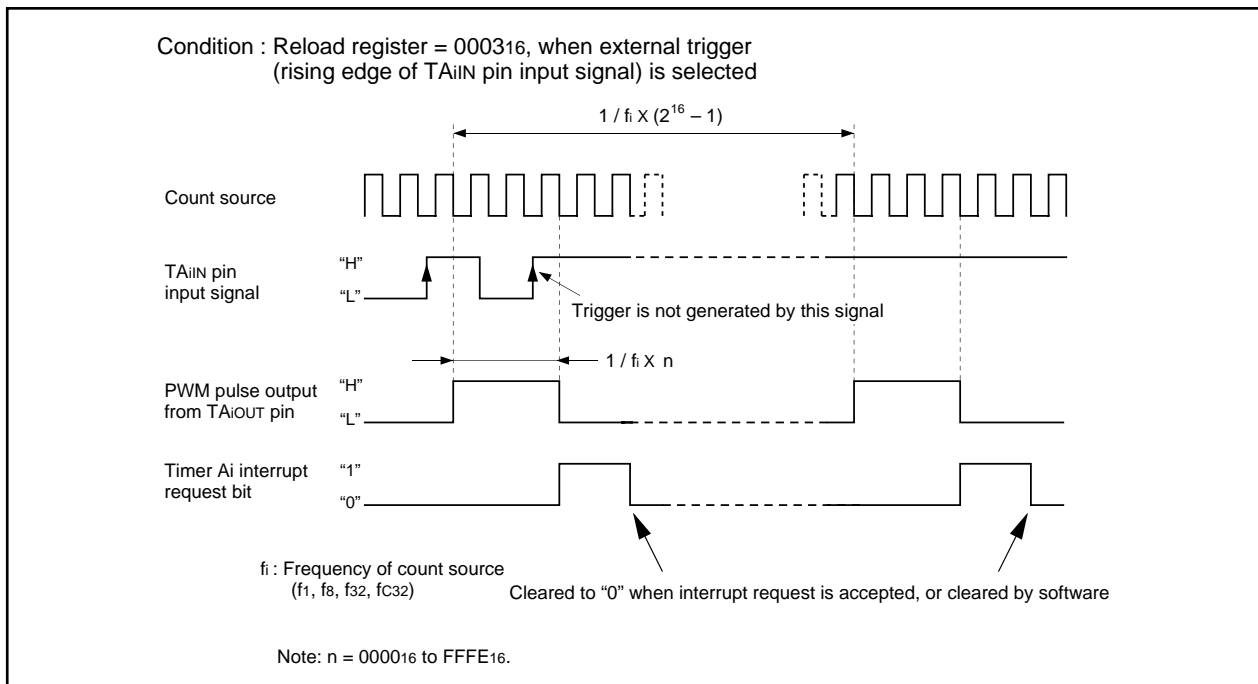


Figure 2.10.12 Example of how a 16-bit pulse width modulator operates

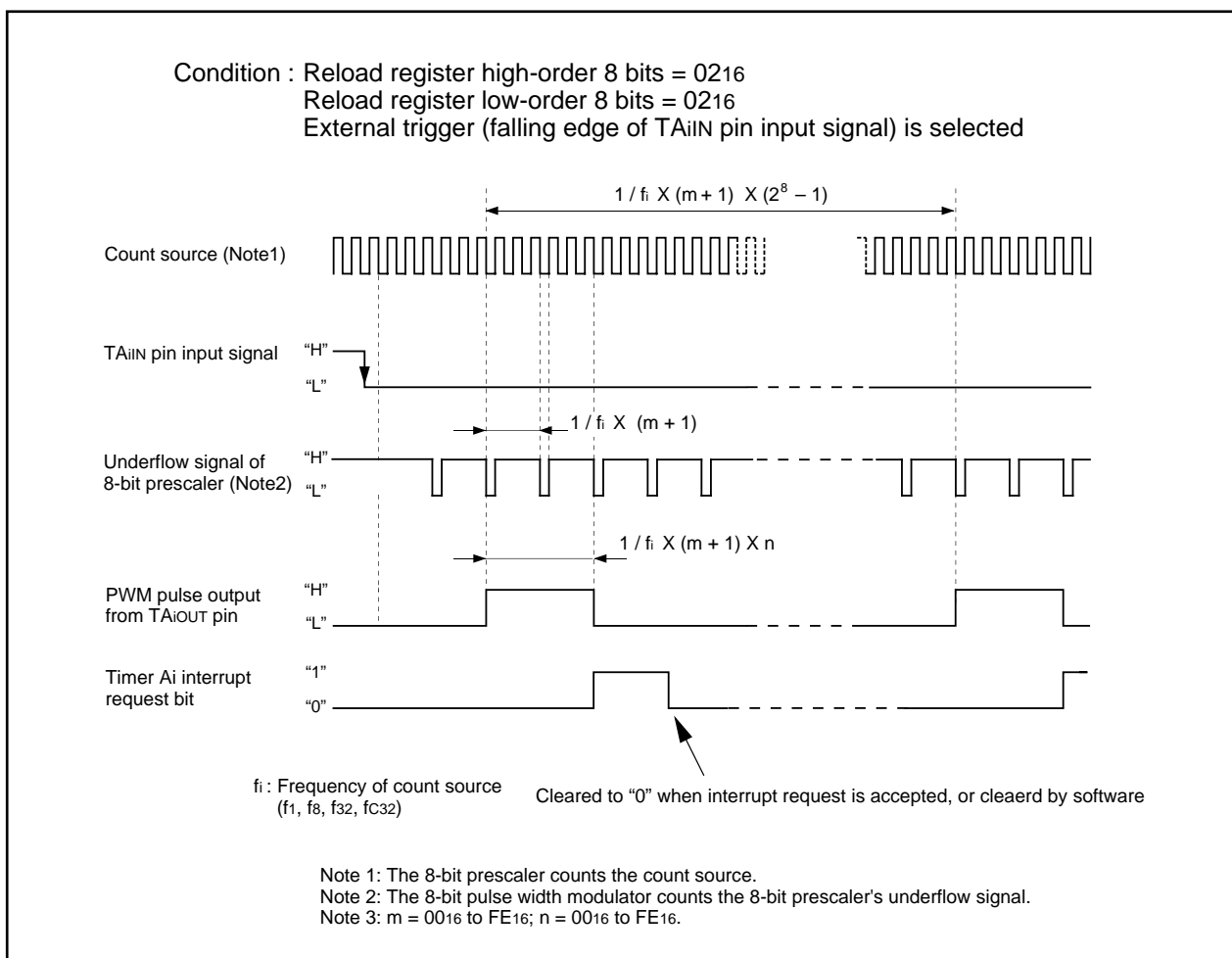


Figure 2.10.13 Example of how an 8-bit pulse width modulator operates

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.10.2 Timer B

Figure 2.10.14 shows the block diagram of timer B. Figures 2.10.15 and 2.10.16 show the timer B-related registers.

Use the timer Bi mode register (i = 0 to 2) bits 0 and 1 to choose the desired mode.

Timer B has three operation modes listed as follows:

- Timer mode: The timer counts an internal count source.
- Event counter mode: The timer counts pulses from an external source or a timer overflow.
- Pulse period/pulse width measuring mode: The timer measures an external signal's pulse period or pulse width.

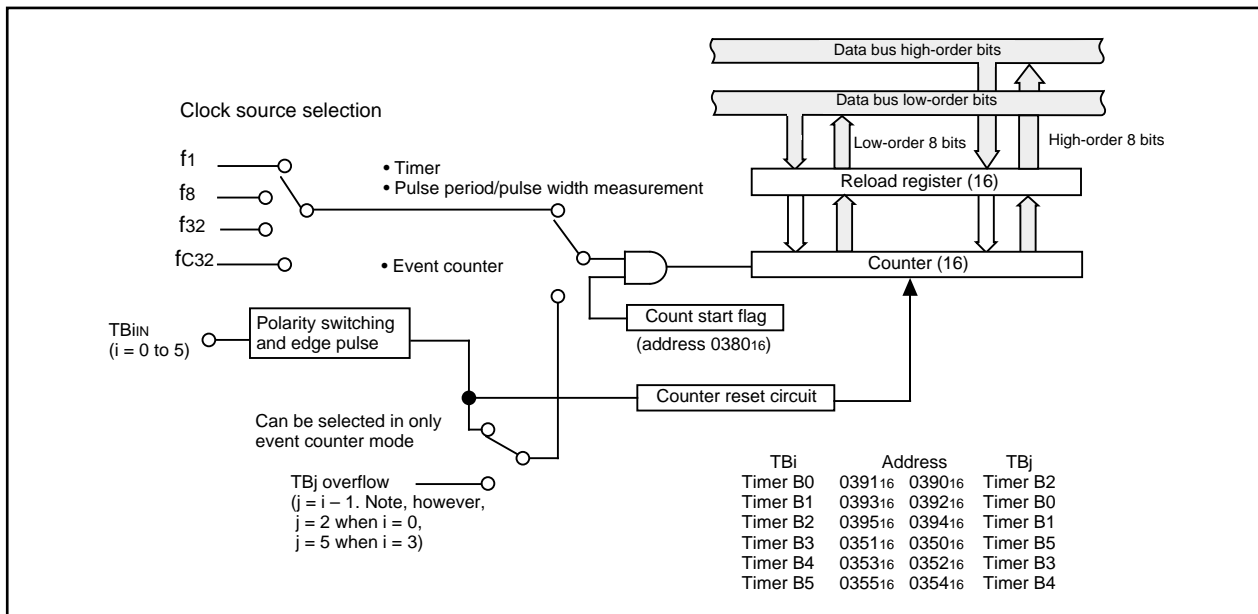


Figure 2.10.14 Block diagram of timer B

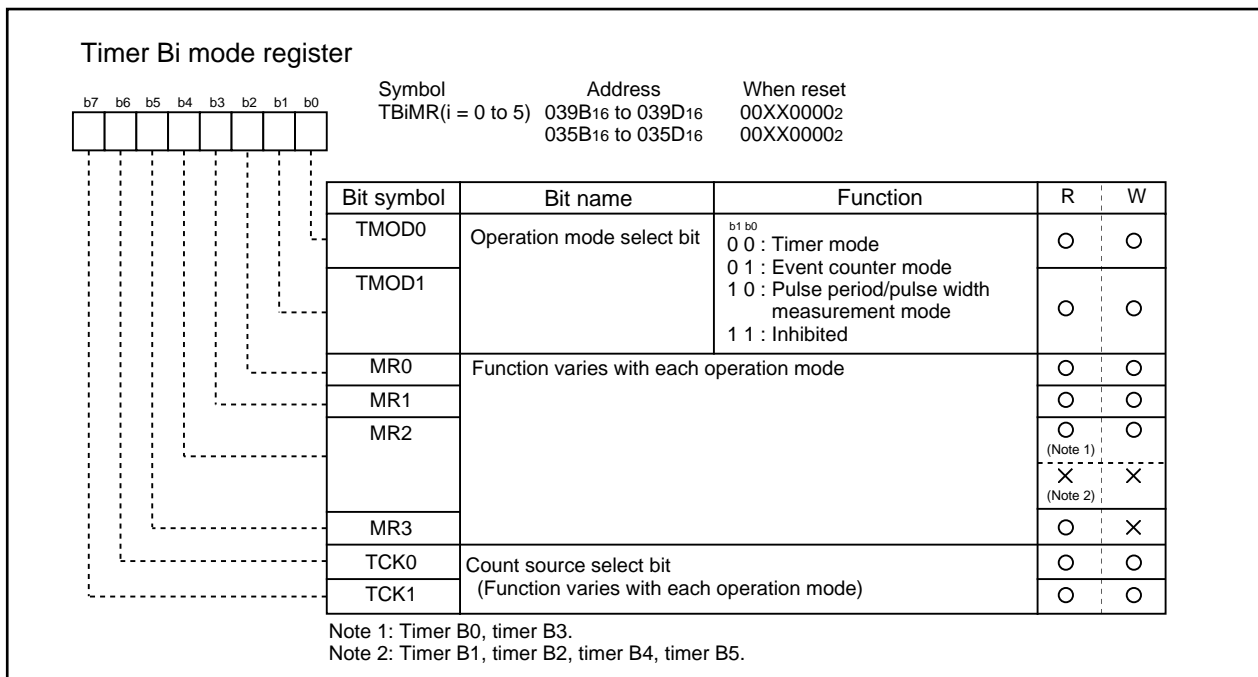
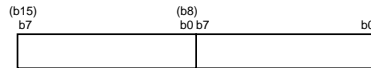


Figure 2.10.15 Timer B-related registers (1)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Timer Bi register (Note)

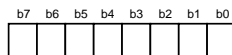


Symbol	Address	When reset
TB0	0391 <sub>16</sub> , 0390 <sub>16</sub>	Indeterminate
TB1	0393 <sub>16</sub> , 0392 <sub>16</sub>	Indeterminate
TB2	0395 <sub>16</sub> , 0394 <sub>16</sub>	Indeterminate
TB3	0351 <sub>16</sub> , 0350 <sub>16</sub>	Indeterminate
TB4	0353 <sub>16</sub> , 0352 <sub>16</sub>	Indeterminate
TB5	0355 <sub>16</sub> , 0354 <sub>16</sub>	Indeterminate

Function	Values that can be set	R	W
• Timer mode Counts the timer's period	0000 <sub>16</sub> to FFFF <sub>16</sub>	○	○
• Event counter mode Counts external pulses input or a timer overflow	0000 <sub>16</sub> to FFFF <sub>16</sub>	○	○
• Pulse period / pulse width measurement mode Measures a pulse period or width	—	○	×

Note: Read and write data in 16-bit units.

## Count start flag



Symbol	Address	When reset
TABSR	0380 <sub>16</sub>	00 <sub>16</sub>

Bit symbol	Bit name	Function	R	W
TA0S	Timer A0 count start flag	0 : Stops counting 1 : Starts counting	○	○
TA1S	Timer A1 count start flag		○	○
TA2S	Timer A2 count start flag		○	○
TA3S	Timer A3 count start flag		○	○
TA4S	Timer A4 count start flag		○	○
TB0S	Timer B0 count start flag		○	○
TB1S	Timer B1 count start flag		○	○
TB2S	Timer B2 count start flag		○	○

## Timer B3, 4, 5 count start flag



Symbol	Address	When reset
TBSR	0340 <sub>16</sub>	000XXXXX <sub>2</sub>

Bit symbol	Bit name	Function	R	W
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0".			—	—
TB3S	Timer B3 count start flag	0 : Stops counting 1 : Starts counting	○	○
TB4S	Timer B4 count start flag		○	○
TB5S	Timer B5 count start flag		○	○

## Clock prescaler reset flag



Symbol	Address	When reset
CPSRF	0381 <sub>16</sub>	0XXXXXXX <sub>2</sub>

Bit symbol	Bit name	Function	R	W
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0".			—	—
CPSR	Clock prescaler reset flag	0 : No effect 1 : Prescaler is reset (When read, the value is "0")	○	○

Figure 2.10.16 Timer B-related registers (2)

# M306H1SFP

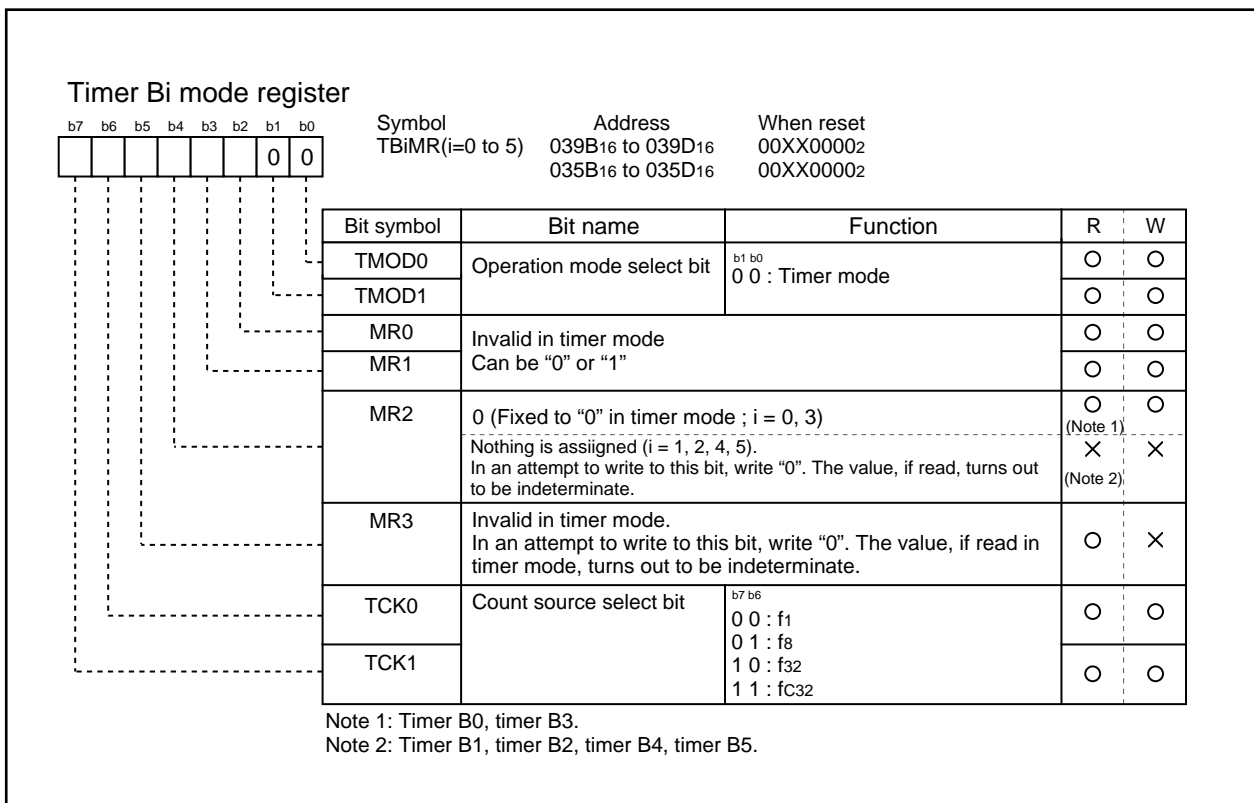
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (1) Timer mode

In this mode, the timer counts an internally generated count source. (See Table 2.10.6) Figure 2.10.17 shows the timer Bi mode register in timer mode.

**Table 2.10.6 Timer specifications in timer mode**

Item	Specification
Count source	f1, f8, f32, fc32
Count operation	<ul style="list-style-type: none"> <li>Counts down</li> <li>When the timer underflows, it reloads the reload register contents before continuing counting</li> </ul>
Divide ratio	1/(n+1)    n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	The timer underflows
TBiIN pin function	Programmable I/O port
Read from timer	Count value is read out by reading timer Bi register
Write to timer	<ul style="list-style-type: none"> <li>When counting stopped When a value is written to timer Bi register, it is written to both reload register and counter</li> <li>When counting in progress When a value is written to timer Bi register, it is written to only reload register (Transferred to counter at next reload time)</li> </ul>



**Figure 2.10.17 Timer Bi mode register in timer mode**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

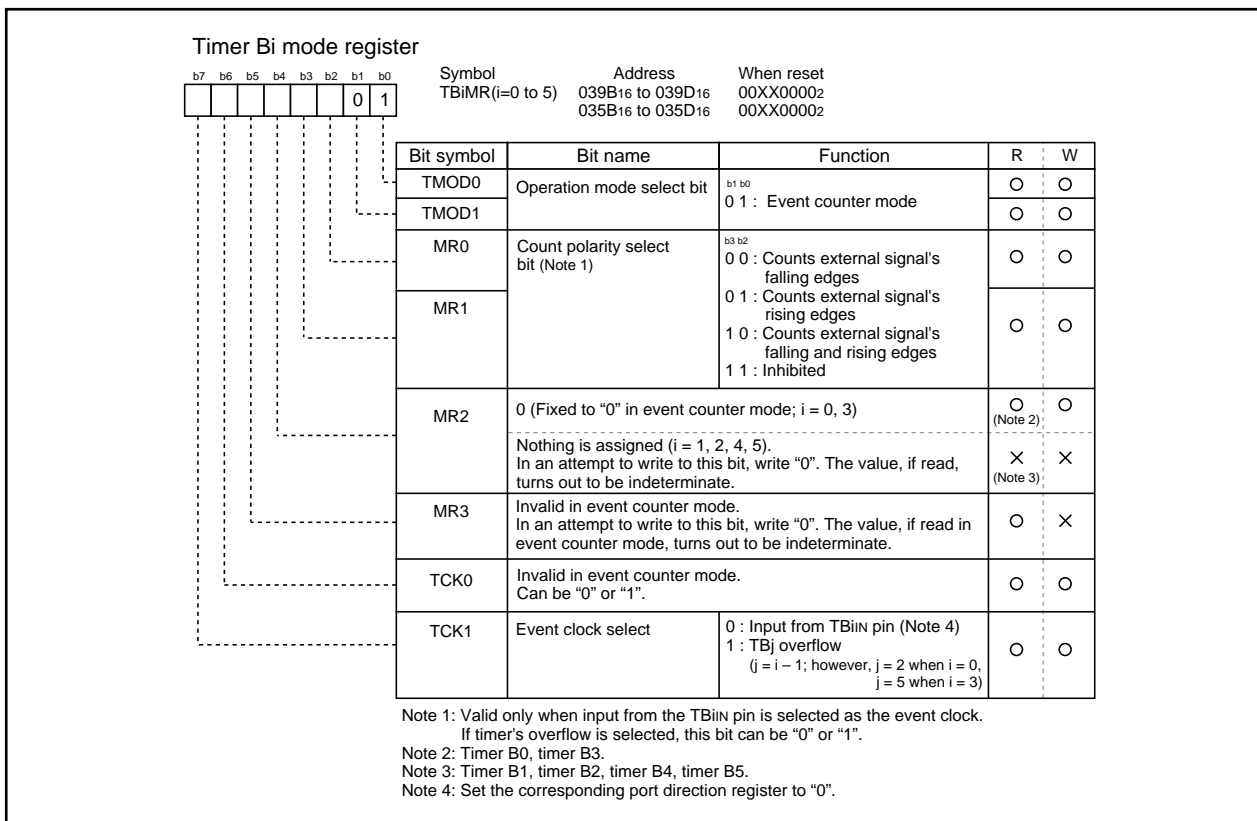
## (2) Event counter mode

In this mode, the timer counts an external signal or an internal timer's overflow. (See Table 2.10.7)

Figure 2.10.18 shows the timer Bi mode register in event counter mode.

### Table 2.10.7 Timer specifications in event counter mode

Item	Specification
Count source	<ul style="list-style-type: none"> <li>• External signals input to TBIIN pin</li> <li>• Effective edge of count source can be a rising edge, a falling edge, or falling and rising edges as selected by software</li> </ul>
Count operation	<ul style="list-style-type: none"> <li>• Counts down</li> <li>• When the timer underflows, it reloads the reload register contents before continuing counting</li> </ul>
Divide ratio	$1/(n+1)$ n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	The timer underflows
TBIIN pin function	Count source input
Read from timer	Count value can be read out by reading timer Bi register
Write to timer	<ul style="list-style-type: none"> <li>• When counting stopped When a value is written to timer Bi register, it is written to both reload register and counter</li> <li>• When counting in progress When a value is written to timer Bi register, it is written to only reload register (Transferred to counter at next reload time)</li> </ul>



**Figure 2.10.18 Timer Bi mode register in event counter mode**



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

### (3) Pulse period/pulse width measurement mode

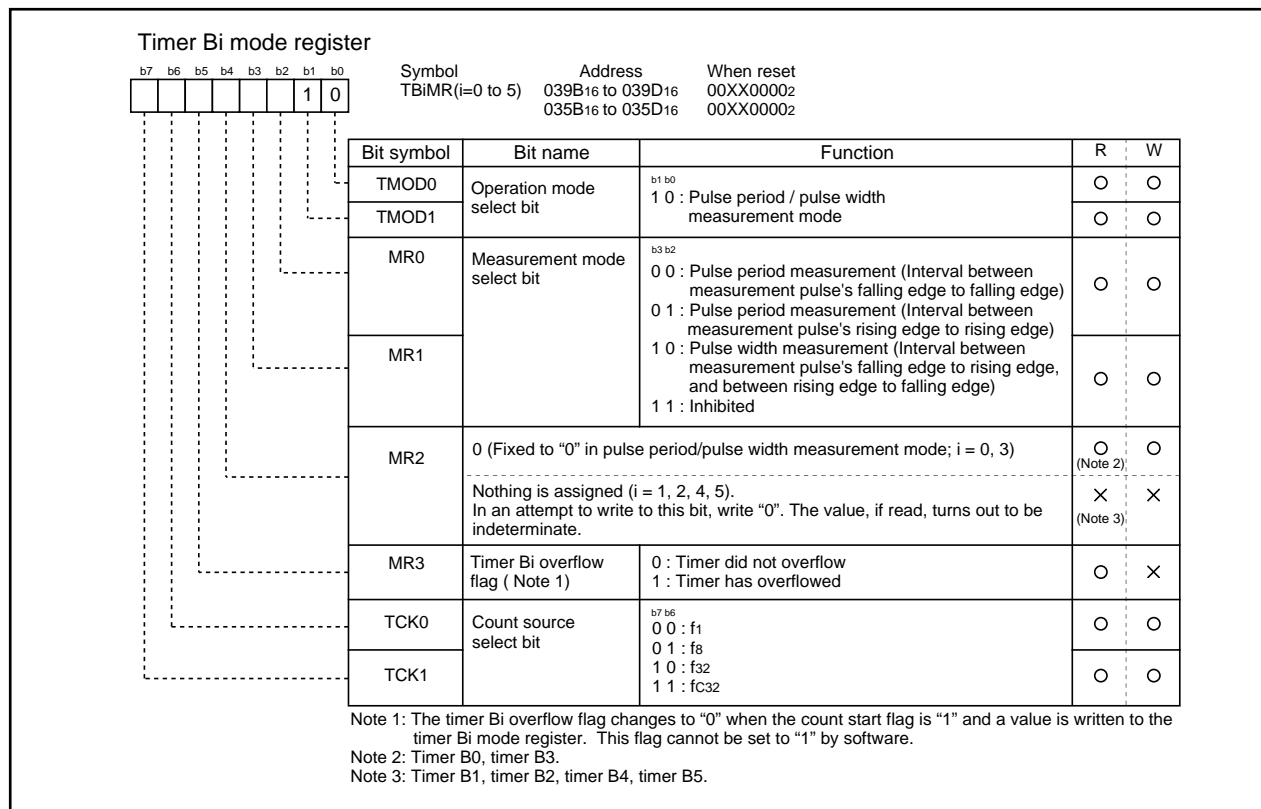
In this mode, the timer measures the pulse period or pulse width of an external signal. (See Table 2.10.8) Figure 2.10.19 shows the timer Bi mode register in pulse period/pulse width measurement mode. Figure 2.10.20 shows the operation timing when measuring a pulse period. Figure 2.10.21 shows the operation timing when measuring a pulse width.

**Table 2.10.8 Timer specifications in pulse period/pulse width measurement mode**

Item	Specification
Count source	f1, f8, f32, fc32
Count operation	<ul style="list-style-type: none"> <li>Up count</li> <li>Counter value "0000<sub>16</sub>" is transferred to reload register at measurement pulse's effective edge and the timer continues counting</li> </ul>
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	<ul style="list-style-type: none"> <li>When measurement pulse's effective edge is input (Note 1)</li> <li>When an overflow occurs. (Simultaneously, the timer Bi overflow flag changes to "1". The timer Bi overflow flag changes to "0" when the count start flag is "1" and a value is written to the timer Bi mode register.)</li> </ul>
TBiIN pin function	Measurement pulse input
Read from timer	When timer Bi register is read, it indicates the reload register's content (measurement result) (Note 2)
Write to timer	Cannot be written to

Note 1: An interrupt request is not generated when the first effective edge is input after the timer has started counting.

Note 2: The value read out from the timer Bi register is indeterminate until the second effective edge is input after the timer.



**Figure 2.10.19 Timer Bi mode register in pulse period/pulse width measurement mode**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

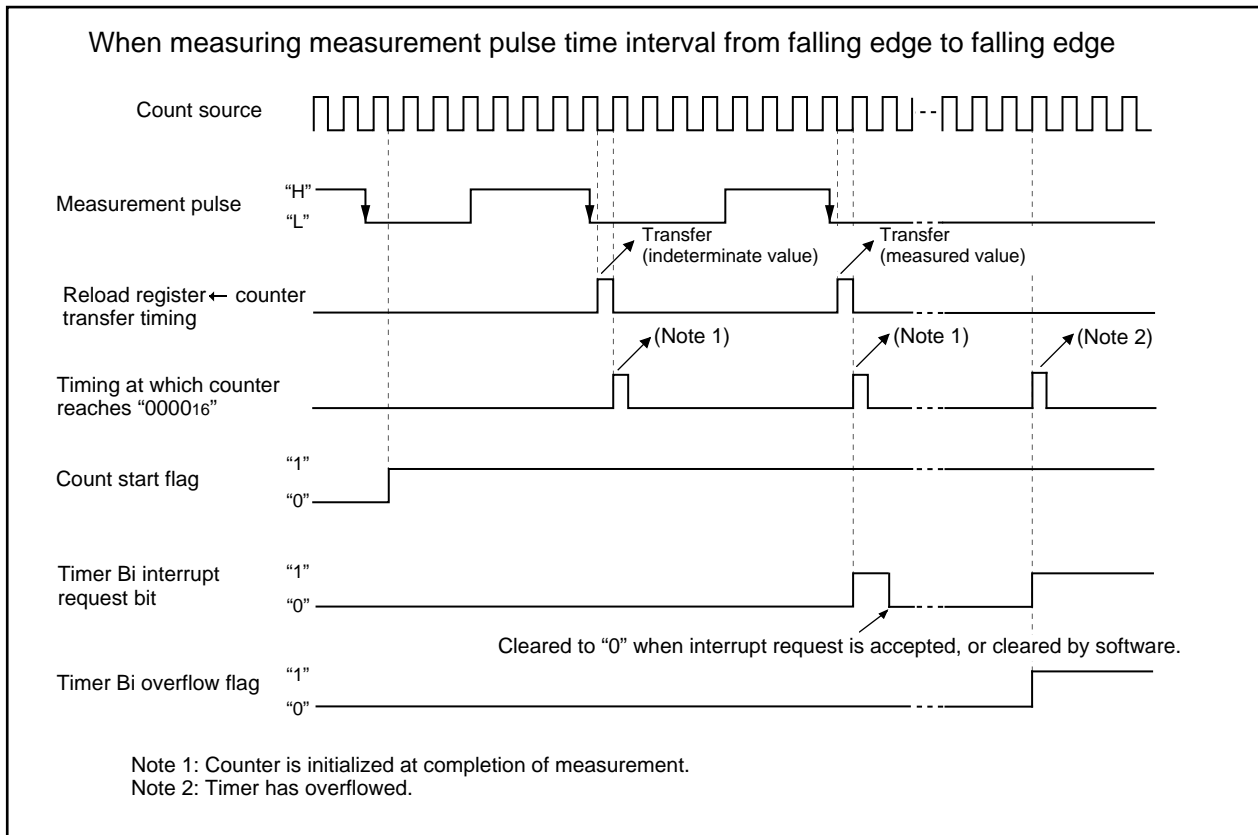


Figure 2.10.20 Operation timing when measuring a pulse period

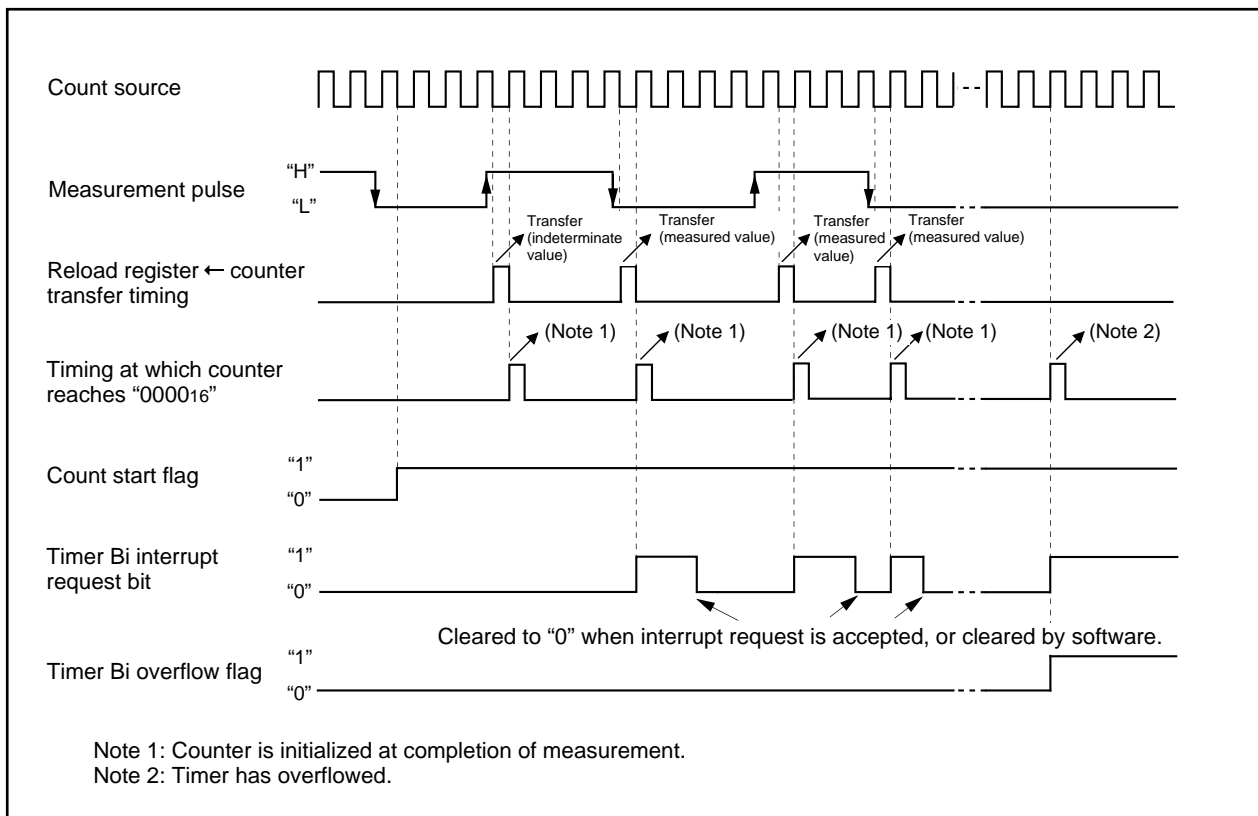


Figure 2.10.21 Operation timing when measuring a pulse width

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.11 Serial I/O

Serial I/O is configured as five channels: UART0, UART1, UART2, S I/O3 and S I/O4.

### 2.11.1 UART0 to 2

UART0, UART1 and UART2 each have an exclusive timer to generate a transfer clock, so they operate independently of each other.

Figure 2.11.1 shows the block diagram of UART0, UART1 and UART2. Figures 2.11.2 and 2.11.3 show the block diagram of the transmit/receive unit.

UARTi (i = 0 to 2) has two operation modes: a clock synchronous serial I/O mode and a clock asynchronous serial I/O mode (UART mode). The contents of the serial I/O mode select bits (bits 0 to 2 at addresses 03A0<sub>16</sub>, 03A8<sub>16</sub> and 0378<sub>16</sub>) determine whether UARTi is used as a clock synchronous serial I/O or as a UART. Although a few functions are different, UART0, UART1 and UART2 have almost the same functions.

UART0 through UART2 are almost equal in their functions with minor exceptions. UART2, in particular, is compliant with the SIM interface with some extra settings added in clock-asynchronous serial I/O mode (Note). It also has the bus collision detection function that generates an interrupt request if the TxD pin and the RxD pin are different in level.

Table 2.11.1 shows the comparison of functions of UART0 through UART2, and Figures 2.11.4 to 2.11.8 show the registers related to UARTi.

Note: SIM : Subscriber Identity Module

**Table 2.11.1 Comparison of functions of UART0 through UART2**

Function	UART0	UART1	UART2
CLK polarity selection	Possible (Note 1)	Possible (Note 1)	Possible (Note 1)
LSB first / MSB first selection	Possible (Note 1)	Possible (Note 1)	Possible (Note 2)
Continuous receive mode selection	Possible (Note 1)	Possible (Note 1)	Possible (Note 1)
Transfer clock output from multiple pins selection	Impossible	Possible (Note 1)	Impossible
Serial data logic switch	Impossible	Impossible	Possible (Note 4)
Sleep mode selection	Possible (Note 3)	Possible (Note 3)	Impossible
TxD, RxD I/O polarity switch	Impossible	Impossible	Possible
TxD, RxD port output format	CMOS output	CMOS output	N-channel open-drain output
Parity error signal output	Impossible	Impossible	Possible (Note 4)
Bus collision detection	Impossible	Impossible	Possible

Note 1: Only when clock synchronous serial I/O mode.

Note 2: Only when clock synchronous serial I/O mode and 8-bit UART mode.

Note 3: Only when UART mode.

Note 4: Using for SIM interface.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

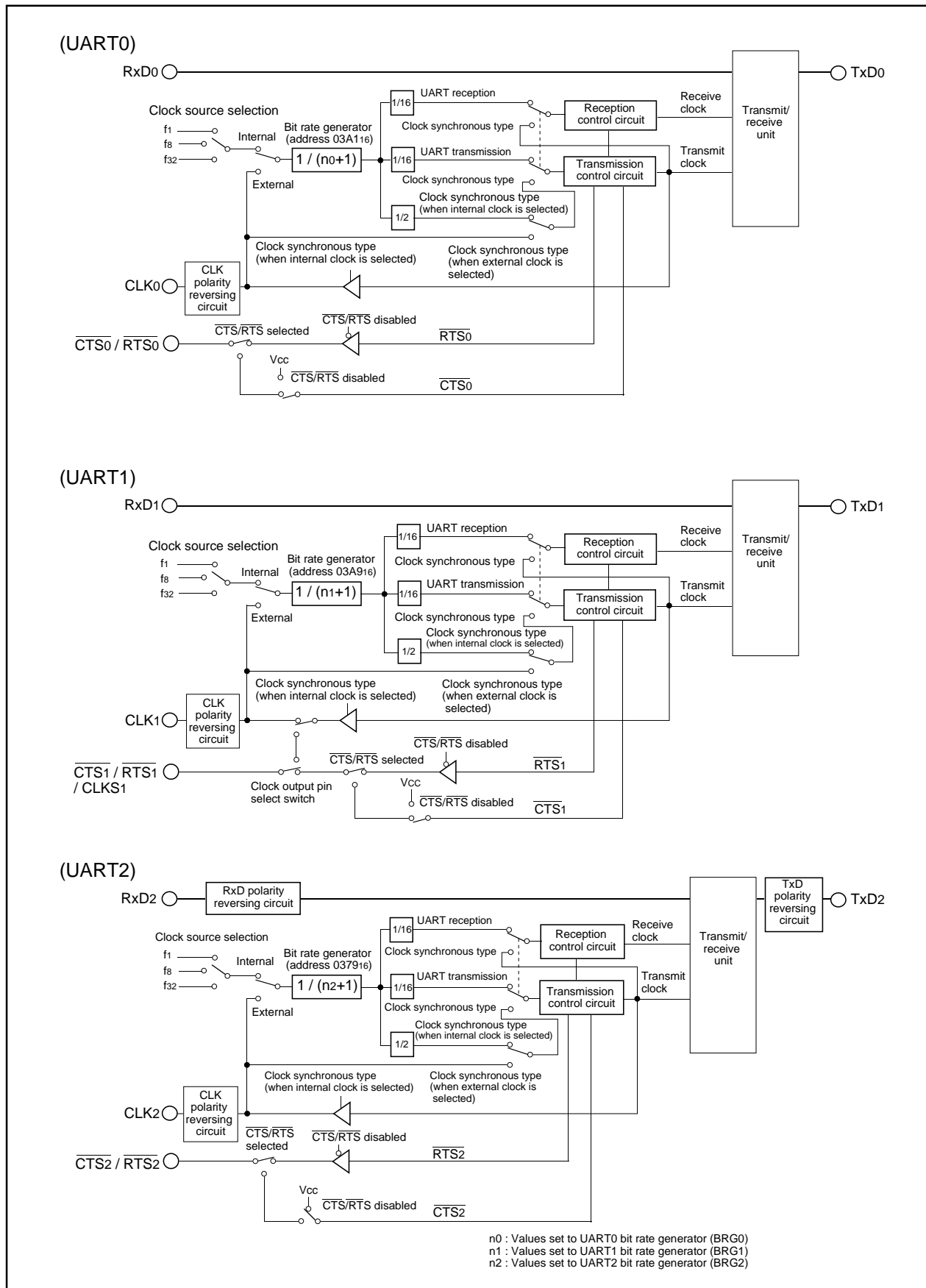


Figure 2.11.1 Block diagram of UARTi (i = 0 to 2)

**M306H1SFP**

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

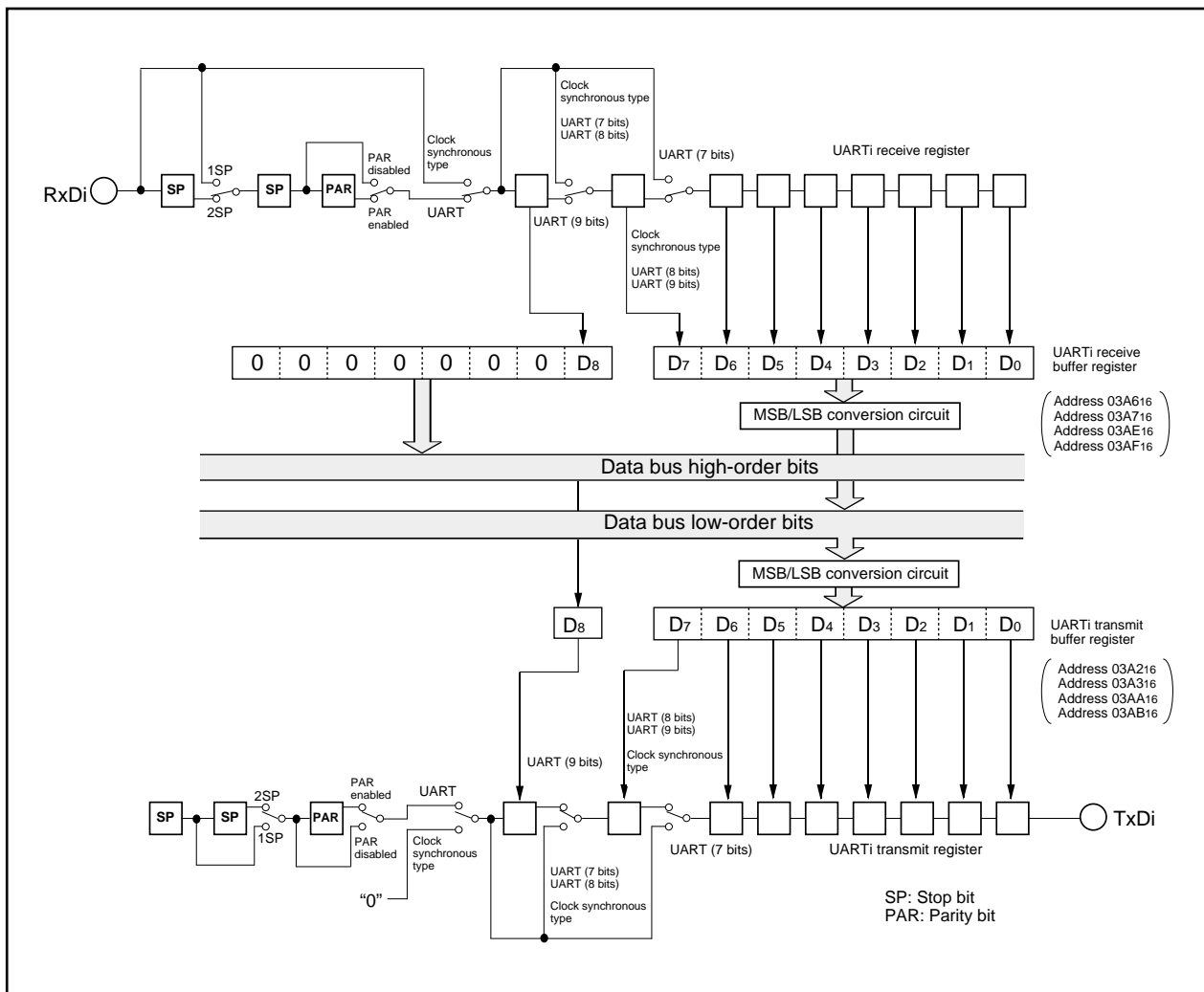


Figure 2.11.2 Block diagram of UARTi (i = 0, 1) transmit/receive unit

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

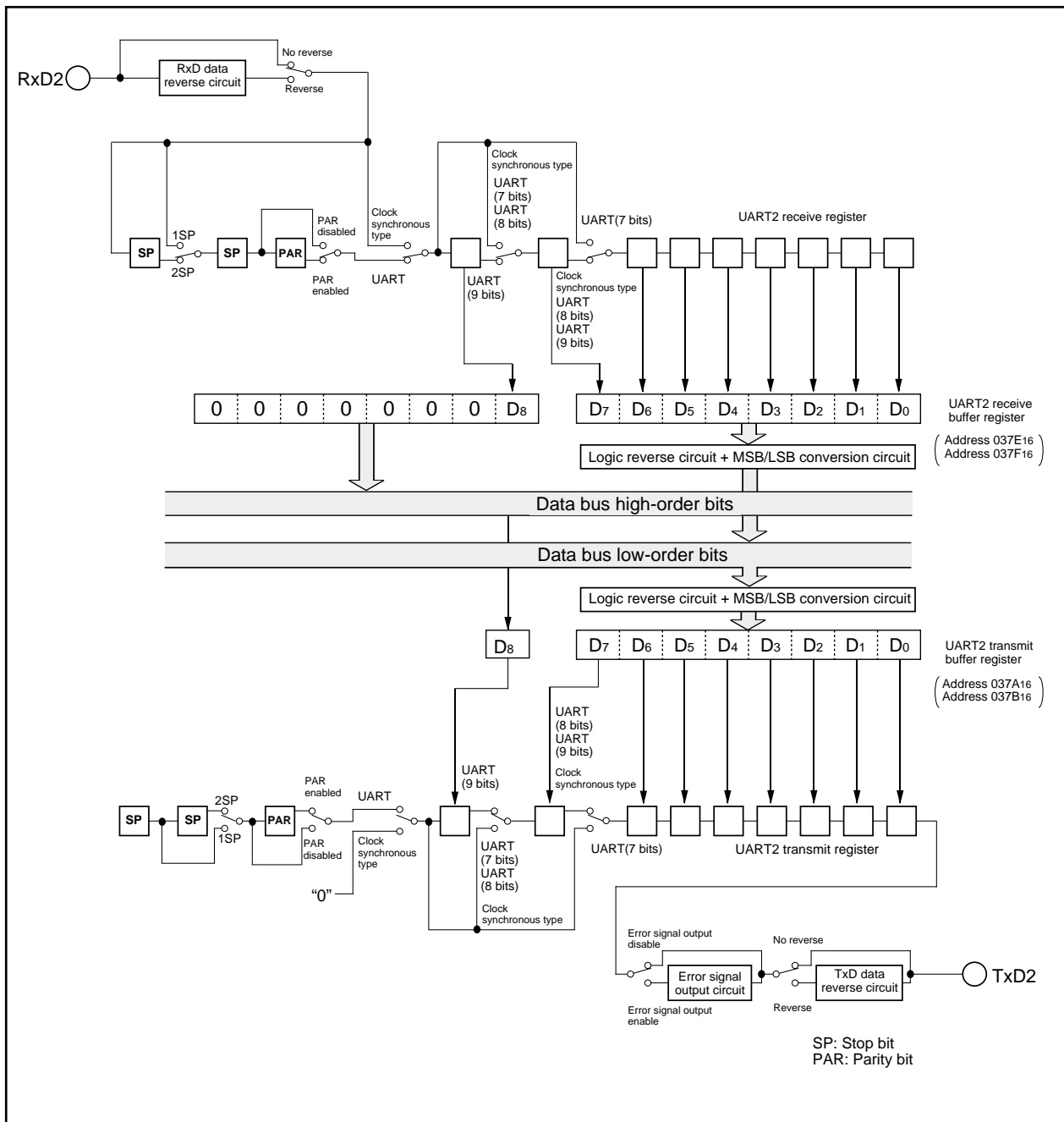


Figure 2.11.3 Block diagram of UART2 transmit/receive unit

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

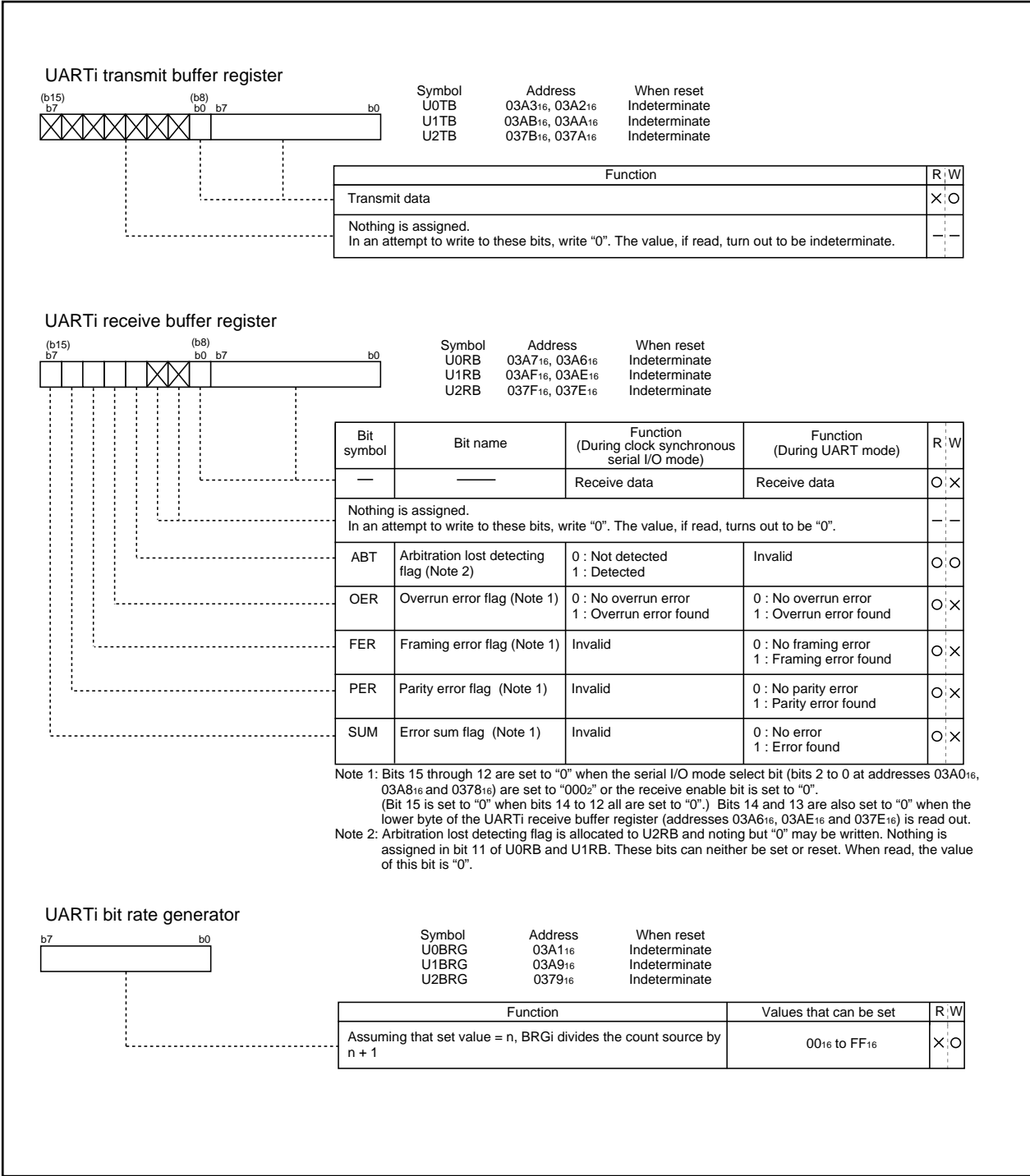


Figure 2.11.4 UARTi I/O-related registers (1)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

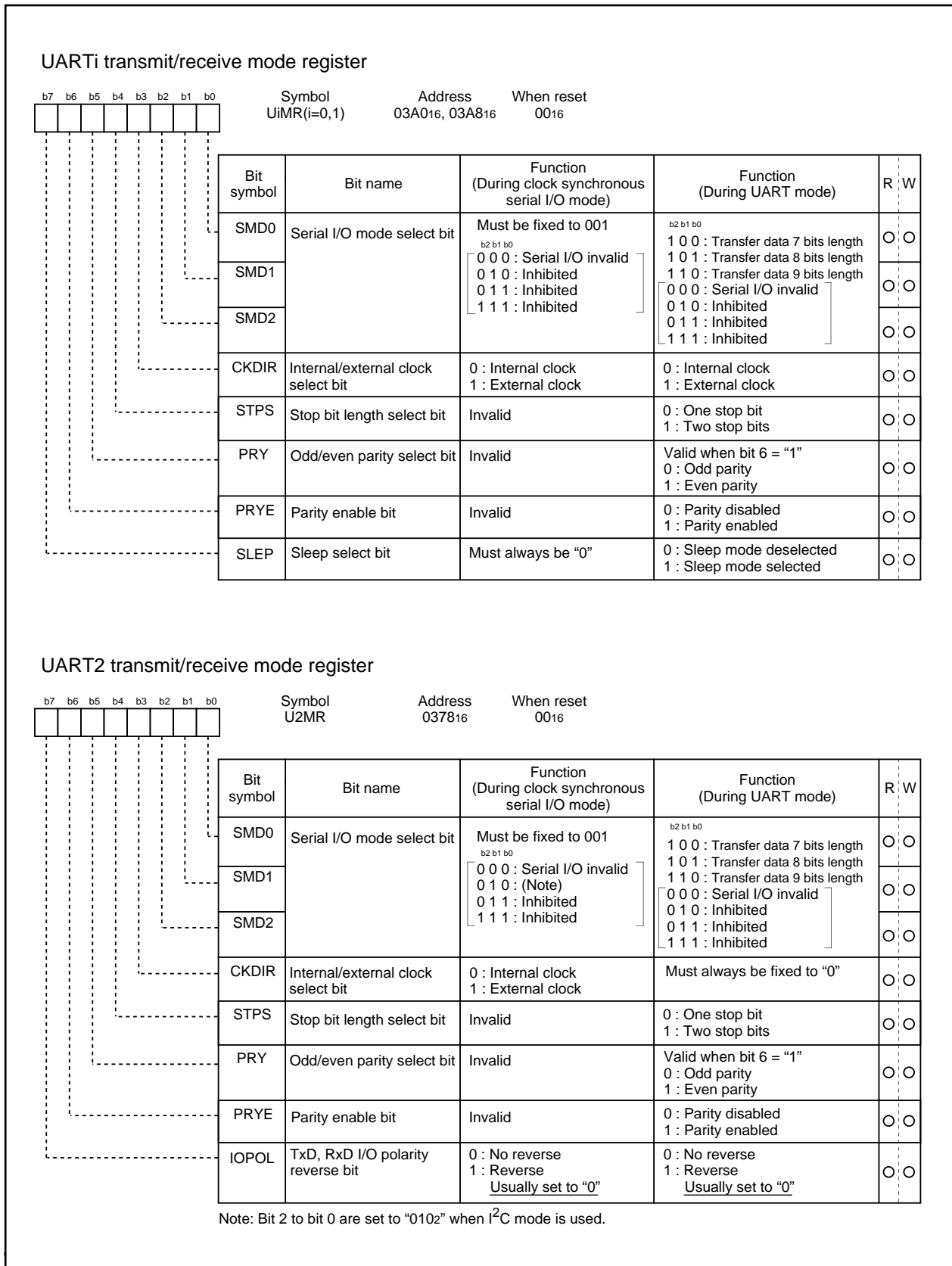


Figure 2.11.5 UARTi I/O-related registers (2)



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

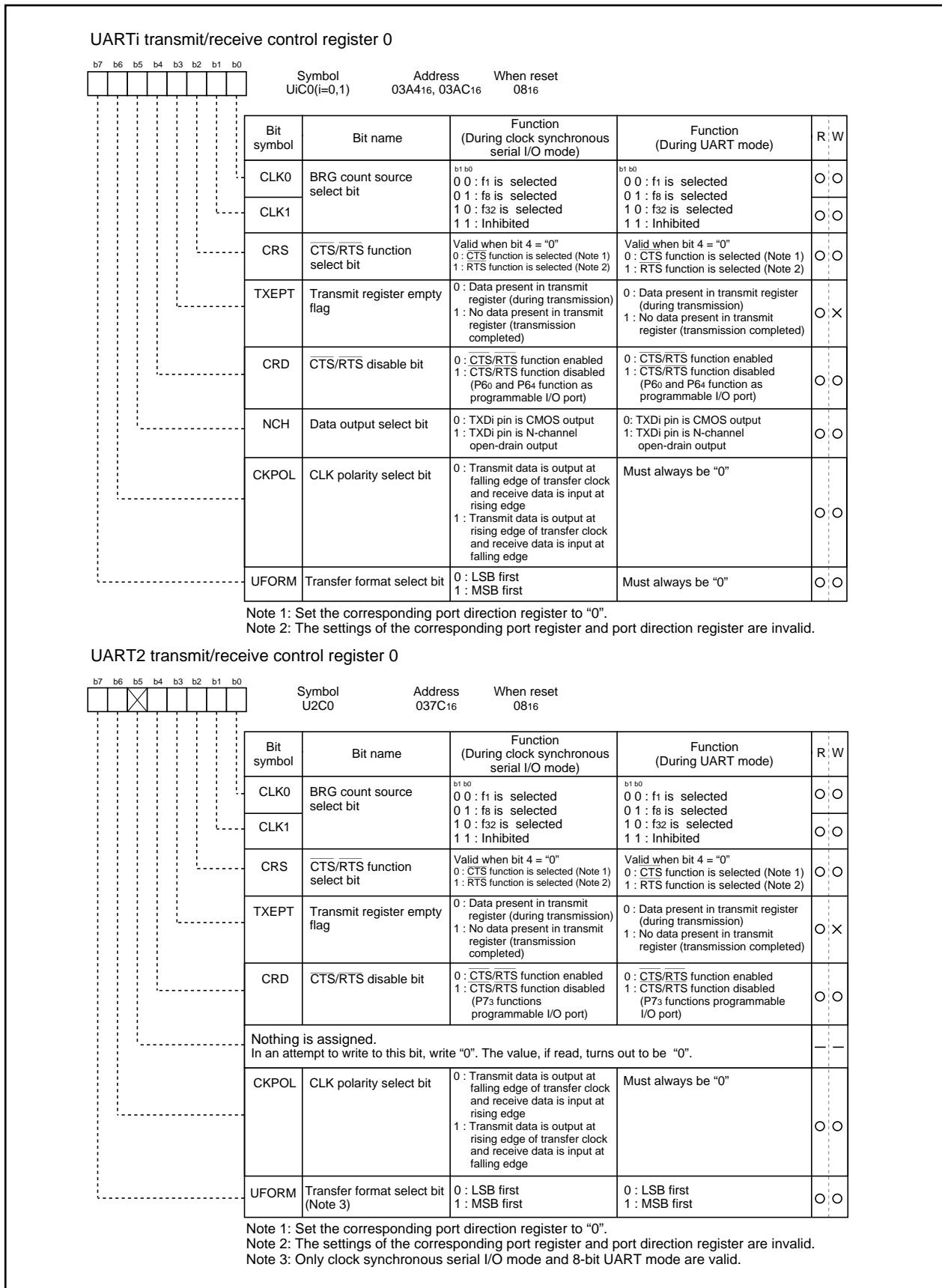


Figure 2.11.6 UARTi I/O-related registers (3)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

UARTi transmit/receive control register 1

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
TE	Transmit enable bit	0 : Transmission disabled 1 : Transmission enabled	0 : Transmission disabled 1 : Transmission enabled	○	○
TI	Transmit buffer empty flag	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	○	×
RE	Receive enable bit	0 : Reception disabled 1 : Reception enabled	0 : Reception disabled 1 : Reception enabled	○	○
RI	Receive complete flag	0 : No data present in receive buffer register 1 : Data present in receive buffer register	0 : No data present in receive buffer register 1 : Data present in receive buffer register	○	×
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0".				—	—

UART2 transmit/receive control register 1

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
TE	Transmit enable bit	0 : Transmission disabled 1 : Transmission enabled	0 : Transmission disabled 1 : Transmission enabled	○	○
TI	Transmit buffer empty flag	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	○	×
RE	Receive enable bit	0 : Reception disabled 1 : Reception enabled	0 : Reception disabled 1 : Reception enabled	○	○
RI	Receive complete flag	0 : No data present in receive buffer register 1 : Data present in receive buffer register	0 : No data present in receive buffer register 1 : Data present in receive buffer register	○	×
U2IRS	UART2 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmit is completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmit is completed (TXEPT = 1)	○	○
U2RRM	UART2 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enabled	Invalid	○	○
U2LCH	Data logic select bit	0 : No reverse 1 : Reverse	0 : No reverse 1 : Reverse	○	○
U2ERE	Error signal output enable bit	Must be fixed to "0"	0 : Output disabled 1 : Output enabled	○	○

Figure 2.11.7 UARTi I/O-related registers (4)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## UART transmit/receive control register 2

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	UCON	03B0 <sub>16</sub>	X0000000 <sub>2</sub>

Note: When using multiple pins to output the transfer clock, the following requirements must be met:  
• UART1 internal/external clock select bit (bit 3 at address 03A8<sub>16</sub>) = "0".

## UART2 special mode register

<div><div>b7b6b5b4b3b2b1b0</div><div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div></div>								Symbol	Address	When reset		
								U2SMR	0377 <sub>16</sub>	00 <sub>16</sub>		
Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W							
IICM	I <sup>2</sup> C mode selection bit	0 : Normal mode 1 : I <sup>2</sup> C mode	Must always be "0"	○	○							
ABC	Arbitration lost detecting flag control bit	0 : Update per bit 1 : Update per byte	Must always be "0"	○	○							
BBS	Bus busy flag	0 : STOP condition detected 1 : START condition detected	Must always be "0"	○	○	(Note 1)						
LSYN	SCLL sync output enable bit	0 : Disabled 1 : Enabled	Must always be "0"	○	○							
ABSCS	Bus collision detect sampling clock select bit	Must always be "0"	0 : Rising edge of transfer clock 1 : Underflow signal of timer A0	○	○							
ACSE	Auto clear function select bit of transmit enable bit	Must always be "0"	0 : No auto clear function 1 : Auto clear at occurrence of bus collision	○	○							
SSS	Transmit start condition select bit	Must always be "0"	0 : Ordinary 1 : Falling edge of Rx/D2	○	○							
SDDS	SDA digital delay selection bit (Notes 2 and 3)	0 : Analog delay output selection 1 : Digital delay output selection	Must always be "0"	○	○							

Notes 1: Nothing but "0" may be written.  
2: Do not write "1" except at I<sup>2</sup>C mode. Must always be "0" at normal mode.  
Bit 7 to bit5 (DL2 to DL0 = SDA digital delay value setting bit) of UART2 special mode register 3 (U2SMR3/address 0375<sub>16</sub>) are initialized and become "000" when this bit is "0", analog delay circuit is selected. Reading and writing U2SMR are enable when SDDS = "0".  
3: Delaying; Only analog delay value when analog delay is selected, and only digital delay value when digital delay is selected.

Figure 2.11.8 UARTi I/O-related registers (5)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## UART2 special mode register 2 (I<sup>2</sup>C bus exclusive register)

<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> <div>b7b6b5b4b3b2b1b0</div>								Symbol	Address	When reset
								U2SMR2	0376 <sub>16</sub>	00 <sub>16</sub>
</										

## UART2 special mode register 3 (I<sup>2</sup>C bus exclusive register)

<div><div>b7b6b5b4b3b2b1b0</div><div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div></div>								Symbol U2SMR3	Address 0375 <sub>16</sub>	When reset Indeterminate  (initializing value is "00 <sub>16</sub> " at SDDS = "1")																		
<table><tr><th>Bit symbol</th><th>Bit name</th><th>Function (I<sup>2</sup>C bus exclusive)</th><th>R</th><th>W</th></tr><tr><td colspan="3">Nothing is assigned. In an attempt to write to this bit, write "0". The value, if read, turns out to be "0". "0" is read out when SDDS = 1.</td><td>—</td><td>—</td></tr><tr><td>DL0</td><td rowspan="3">SDA digital delay value setting bit</td><td>b7 b6 b5 0 0 0 : Analog delay 0 0 1 : 2 cycle of 1/f (Xin)(Digital delay) 0 1 0 : 3 cycle of 1/f (Xin)(Digital delay) 0 1 1 : 4 cycle of 1/f (Xin)(Digital delay) 1 0 0 : 5 cycle of 1/f (Xin)(Digital delay) 1 0 1 : 6 cycle of 1/f (Xin)(Digital delay) 1 1 0 : 7 cycle of 1/f (Xin)(Digital delay) 1 1 1 : 8 cycle of 1/f (Xin)(Digital delay)</td><td>○</td><td>○</td></tr><tr><td>DL1</td><td>○</td><td>○</td></tr><tr><td>DL2</td><td>○</td><td>○</td></tr></table>								Bit symbol	Bit name	Function (I <sup>2</sup> C bus exclusive)	R	W	Nothing is assigned. In an attempt to write to this bit, write "0". The value, if read, turns out to be "0". "0" is read out when SDDS = 1.			—	—	DL0	SDA digital delay value setting bit	b7 b6 b5 0 0 0 : Analog delay 0 0 1 : 2 cycle of 1/f (Xin)(Digital delay) 0 1 0 : 3 cycle of 1/f (Xin)(Digital delay) 0 1 1 : 4 cycle of 1/f (Xin)(Digital delay) 1 0 0 : 5 cycle of 1/f (Xin)(Digital delay) 1 0 1 : 6 cycle of 1/f (Xin)(Digital delay) 1 1 0 : 7 cycle of 1/f (Xin)(Digital delay) 1 1 1 : 8 cycle of 1/f (Xin)(Digital delay)	○	○	DL1	○	○	DL2	○	○
Bit symbol	Bit name	Function (I <sup>2</sup> C bus exclusive)	R	W																								
Nothing is assigned. In an attempt to write to this bit, write "0". The value, if read, turns out to be "0". "0" is read out when SDDS = 1.			—	—																								
DL0	SDA digital delay value setting bit	b7 b6 b5 0 0 0 : Analog delay 0 0 1 : 2 cycle of 1/f (Xin)(Digital delay) 0 1 0 : 3 cycle of 1/f (Xin)(Digital delay) 0 1 1 : 4 cycle of 1/f (Xin)(Digital delay) 1 0 0 : 5 cycle of 1/f (Xin)(Digital delay) 1 0 1 : 6 cycle of 1/f (Xin)(Digital delay) 1 1 0 : 7 cycle of 1/f (Xin)(Digital delay) 1 1 1 : 8 cycle of 1/f (Xin)(Digital delay)	○	○																								
DL1		○	○																									
DL2		○	○																									

- Notes 1: Reading and writing is possible when bit7 (SDDS = SDA digital delay selection bit) of UART2 special mode register (U2SMR/address 0377<sub>16</sub>) is "1". When set SDDS = "1" and read out initialized value of UART2 special mode register 3(U2SMR3), this value is "00<sub>16</sub>". When set SDDS = "1" and write to UART2 special mode register 3(U2SMR3), set "0" to bit 0 to bit 4. When SDDS = "0", writing is enable. When read out, this value is indeterminate.
- 2: When SDDS = "0", this bit is initialized and become "000", selected analog delay circuit. This bit is become "000" after end reset released, and selected analog delay circuit. Reading out is possible when only SDDS = "1". when SDDS = "0", value which was read out is indeterminate.
- 3: Delaying ; Only analog delay value when analog delay is selected, and only digital delay value when digital delay is selected.
- 4: Delay level depends on SCL pin and SDA pin. And, when use external clock, delay is increase around 100ns. So test first, and use this.

Figure 2.11.9 UARTi-related registers (6)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.11.2 Clock synchronous serial I/O mode

The clock synchronous serial I/O mode uses a transfer clock to transmit and receive data. Tables 2.11.2 and 2.11.3 list the specifications of the clock synchronous serial I/O mode. Figur 2.11.10 shows the UARTi transmit/receive mode register.

**Table 2.11.2 Specifications of clock synchronous serial I/O mode (1)**

Item	Specification
Transfer data format	<ul style="list-style-type: none"> <li>Transfer data length: 8 bits</li> </ul>
Transfer clock	<ul style="list-style-type: none"> <li>When internal clock is selected (bit 3 at addresses 03A0<sub>16</sub>, 03A8<sub>16</sub>, 0378<sub>16</sub> = "0") : <math>f_i / 2(n+1)</math> (Note 1) <math>f_i = f_1, f_8, f_{32}</math></li> <li>When external clock is selected (bit 3 at addresses 03A0<sub>16</sub>, 03A8<sub>16</sub>, 0378<sub>16</sub> = "1") : Input from CLKi pin</li> </ul>
Transmission/reception control	<ul style="list-style-type: none"> <li>CTS function/RTS function/CTS, RTS function chosen to be invalid</li> </ul>
Transmission start condition	<ul style="list-style-type: none"> <li>To start transmission, the following requirements must be met: <ul style="list-style-type: none"> <li>Transmit enable bit (bit 0 at addresses 03A5<sub>16</sub>, 03AD<sub>16</sub>, 037D<sub>16</sub>) = "1"</li> <li>Transmit buffer empty flag (bit 1 at addresses 03A5<sub>16</sub>, 03AD<sub>16</sub>, 037D<sub>16</sub>) = "0"</li> <li>When CTS function selected, CTS input level = "L"</li> </ul> </li> <li>Furthermore, if external clock is selected, the following requirements must also be met: <ul style="list-style-type: none"> <li>CLKi polarity select bit (bit 6 at addresses 03A4<sub>16</sub>, 03AC<sub>16</sub>, 037C<sub>16</sub>) = "0": CLKi input level = "H"</li> <li>CLKi polarity select bit (bit 6 at addresses 03A4<sub>16</sub>, 03AC<sub>16</sub>, 037C<sub>16</sub>) = "1": CLKi input level = "L"</li> </ul> </li> </ul>
Reception start condition	<ul style="list-style-type: none"> <li>To start reception, the following requirements must be met: <ul style="list-style-type: none"> <li>Receive enable bit (bit 2 at addresses 03A5<sub>16</sub>, 03AD<sub>16</sub>, 037D<sub>16</sub>) = "1"</li> <li>Transmit enable bit (bit 0 at addresses 03A5<sub>16</sub>, 03AD<sub>16</sub>, 037D<sub>16</sub>) = "1"</li> <li>Transmit buffer empty flag (bit 1 at addresses 03A5<sub>16</sub>, 03AD<sub>16</sub>, 037D<sub>16</sub>) = "0"</li> </ul> </li> <li>Furthermore, if external clock is selected, the following requirements must also be met: <ul style="list-style-type: none"> <li>CLKi polarity select bit (bit 6 at addresses 03A4<sub>16</sub>, 03AC<sub>16</sub>, 037C<sub>16</sub>) = "0": CLKi input level = "H"</li> <li>CLKi polarity select bit (bit 6 at addresses 03A4<sub>16</sub>, 03AC<sub>16</sub>, 037C<sub>16</sub>) = "1": CLKi input level = "L"</li> </ul> </li> </ul>
Interrupt request generation timing	<ul style="list-style-type: none"> <li>When transmitting <ul style="list-style-type: none"> <li>Transmit interrupt cause select bit (bits 0, 1 at address 03B0<sub>16</sub>, bit 4 at address 037D<sub>16</sub>) = "0": Interrupts requested when data transfer from UARTi transfer buffer register to UARTi transmit register is completed</li> <li>Transmit interrupt cause select bit (bits 0, 1 at address 03B0<sub>16</sub>, bit 4 at address 037D<sub>16</sub>) = "1": Interrupts requested when data transmission from UARTi transfer register is completed</li> </ul> </li> <li>When receiving <ul style="list-style-type: none"> <li>Interrupts requested when data transfer from UARTi receive register to UARTi receive buffer register is completed</li> </ul> </li> </ul>
Error detection	<ul style="list-style-type: none"> <li>Overrun error (Note 2) This error occurs when the next data is ready before contents of UARTi receive buffer register are read out</li> </ul>

Note 1: "n" denotes the value 00<sub>16</sub> to FF<sub>16</sub> that is set to the UART bit rate generator.

Note 2: If an overrun error occurs, the UARTi receive buffer will have the next data written in. Note also that the UARTi receive interrupt request bit is not set to "1".

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Table 2.11.3 Specifications of clock synchronous serial I/O mode (2)**

Item	Specification
Select function	<ul style="list-style-type: none"> <li>• CLK polarity selection Whether transmit data is output/input at the rising edge or falling edge of the transfer clock can be selected</li> <li>• LSB first/MSB first selection Whether transmission/reception begins with bit 0 or bit 7 can be selected</li> <li>• Continuous receive mode selection Reception is enabled simultaneously by a read from the receive buffer register</li> <li>• Transfer clock output from multiple pins selection (UART1) (Note) UART1 transfer clock can be chosen by software to be output from one of the two pins set</li> <li>• Switching serial data logic (UART2) Whether to reverse data in writing to the transmission buffer register or reading the reception buffer register can be selected.</li> <li>• TxD, RxD I/O polarity reverse (UART2) This function is reversing TxD port output and RxD port input. All I/O data level is reversed.</li> </ul>

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## UARTi transmit/receive mode registers

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset				
0					0	0	1	UiMR(i=0,1)	03A0 <sub>16</sub> , 03A8 <sub>16</sub>	00 <sub>16</sub>				
								Bit symbol	Bit name	Function	R/W			
								SMD0	Serial I/O mode select bit	b2 b1 b0 0 0 1 : Clock synchronous serial I/O mode	○	○		
								SMD1			○	○		
								SMD2			○	○		
								CKDIR	Internal/external clock select bit	0 : Internal clock 1 : External clock	○	○		
								STPS	Invalid in clock synchronous serial I/O mode				○	○
								PRY					○	○
								PRYE					○	○
								SLEP	0 (Must always be “0” in clock synchronous serial I/O mode)				○	○

## UART2 transmit/receive mode register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset	
0					0	0	1	U2MR	0378 <sub>16</sub>	00 <sub>16</sub>	
								Bit symbol	Bit name	Function	R/W
								SMD0	Serial I/O mode select bit	b2 b1 b0 0 0 1 : Clock synchronous serial I/O mode	○ ○
								SMD1			○ ○
								SMD2			○ ○
								CKDIR	Internal/external clock select bit	0 : Internal clock 1 : External clock	○ ○
								STPS	Invalid in clock synchronous serial I/O mode		○ ○
								PRY			○ ○
								PRYE			○ ○
								IOPOL	TxD, RxD I/O polarity reverse bit (Note)	0 : No reverse 1 : Reverse	○ ○

Note: Usually set to "0".

Figure 2.11.10 UARTi transmit/receive mode register in clock synchronous serial I/O mode

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Table 2.11.4 lists the functions of the input/output pins during clock synchronous serial I/O mode. Note that for a period from when the UARTi operation mode is selected to when transfer starts, the TxDi pin outputs a "H". (If the N-channel open-drain is selected, this pin is in floating state.)

**Table 2.11.4 Input/output pin functions in clock synchronous serial I/O mode**

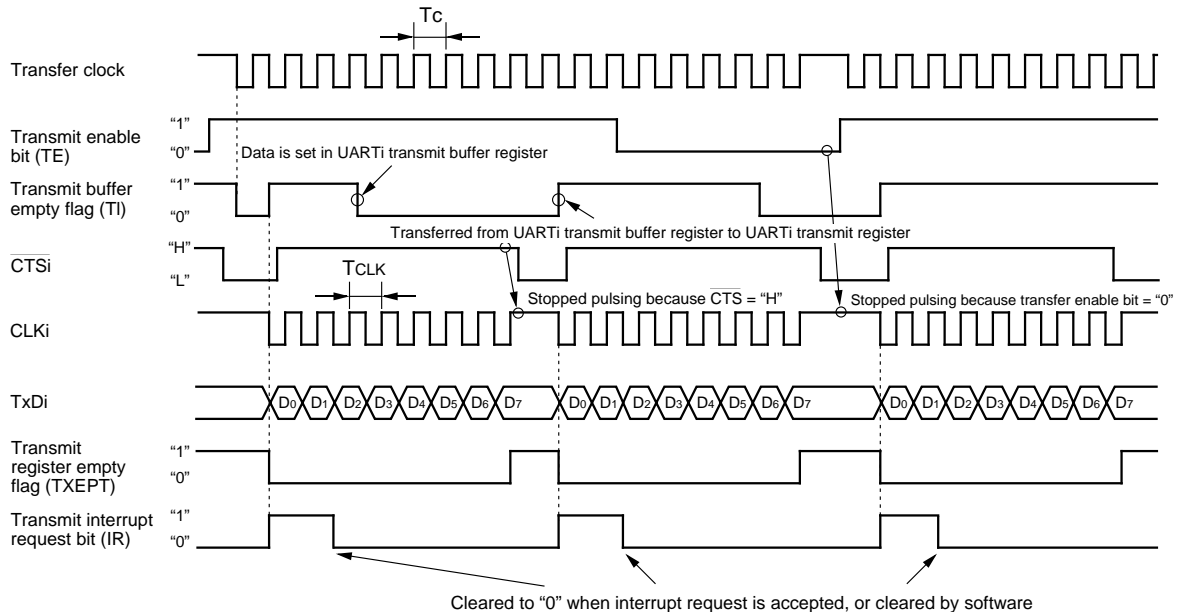
Pin name	Function	Method of selection
TxDi (P63, P67, P70)	Serial data output	(Outputs dummy data when performing reception only)
RxDi (P62, P66, P71)	Serial data input	Port P62, P66 and P71 direction register (bits 2 and 6 at address 03EE16, bit 1 at address 03EF16) = "0" (Can be used as an input port when performing transmission only)
CLKi (P61, P65, P72)	Transfer clock output	Internal/external clock select bit (bit 3 at address 03A016, 03A816, 037816) = "0"
	Transfer clock input	Internal/external clock select bit (bit 3 at address 03A016, 03A816, 037816) = "1" Port P61, P65 and P72 direction register (bits 1 and 5 at address 03EE16, bit 2 at address 03EF16) = "0"
CTS <sub>i</sub> /RTS <sub>i</sub> (P60, P64, P73)	CTS input	CTS/RTS disable bit (bit 4 at address 03A416, 03AC16, 037C16) = "0" CTS/RTS function select bit (bit 2 at address 03A416, 03AC16, 037C16) = "0" Port P60, P64 and P73 direction register (bits 0 and 4 at address 03EE16, bit 3 at address 03EF16) = "0"
	RTS output	CTS/RTS disable bit (bit 4 at address 03A416, 03AC16, 037C16) = "0" CTS/RTS function select bit (bit 2 at address 03A416, 03AC16, 037C16) = "1"
	Programmable I/O port	CTS/RTS disable bit (bit 4 at address 03A416, 03AC16, 037C16) = "1"



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## • Example of transmit timing (when internal clock is selected)



## • Example of receive timing (when external clock is selected)

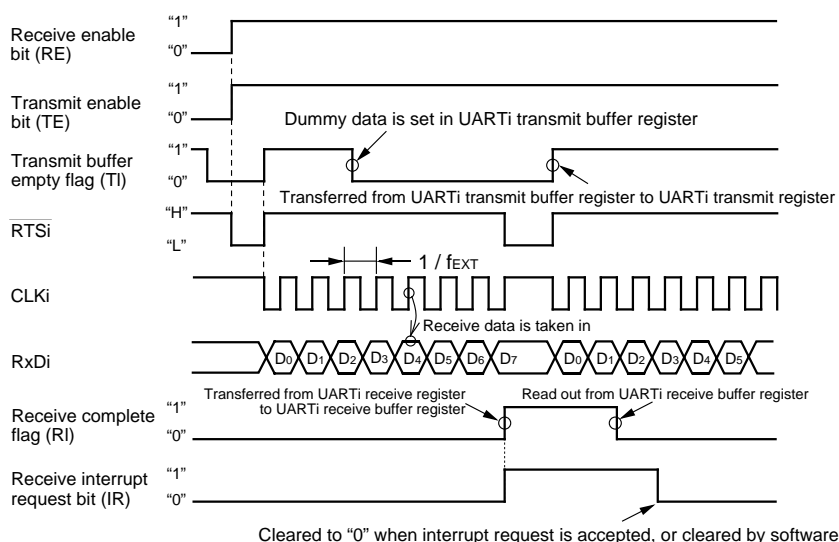


Figure 2.11.11 Typical transmit/receive timings in clock synchronous serial I/O mode

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
 with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (1) Polarity select function

As shown in Figure 2.11.12 the CLK polarity select bit (bit 6 at addresses 03A4<sub>16</sub>, 03AC<sub>16</sub>, 037C<sub>16</sub>) allows selection of the polarity of the transfer clock.

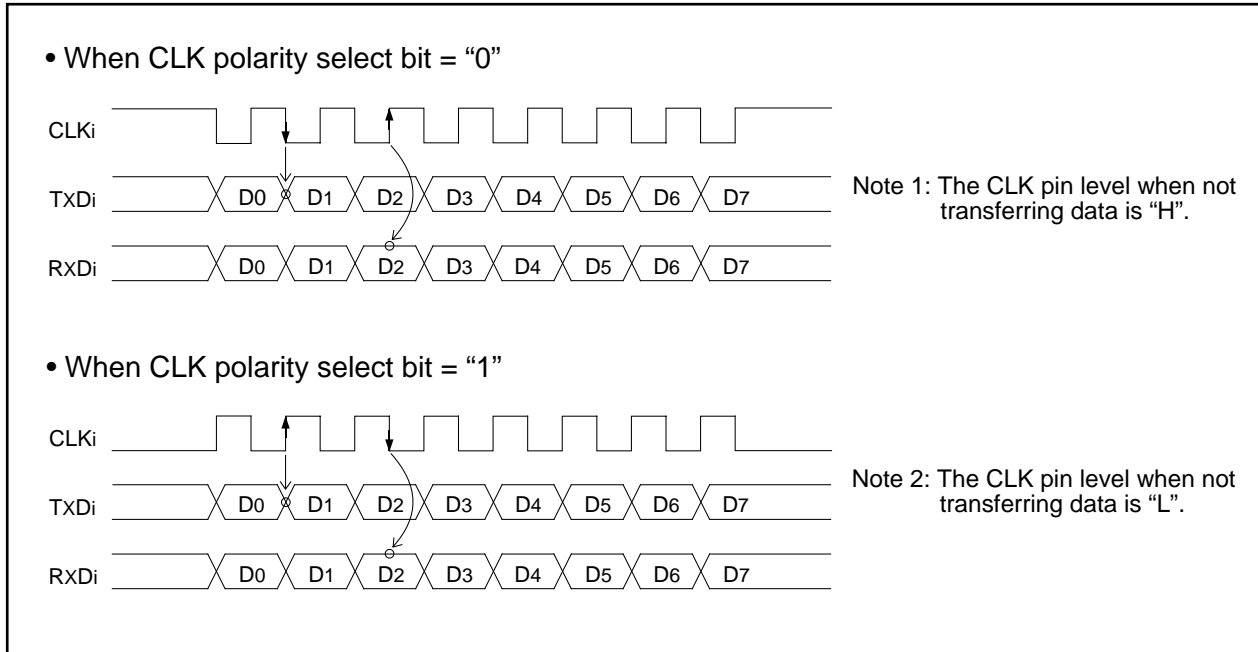


Figure 2.11.12 Polarity of transfer clock

## (2) LSB first/MSB first select function

As shown in Figure 2.11.13, when the transfer format select bit (bit 7 at addresses 03A4<sub>16</sub>, 03AC<sub>16</sub>, 037C<sub>16</sub>) = "0", the transfer format is "LSB first"; when the bit = "1", the transfer format is "MSB first".

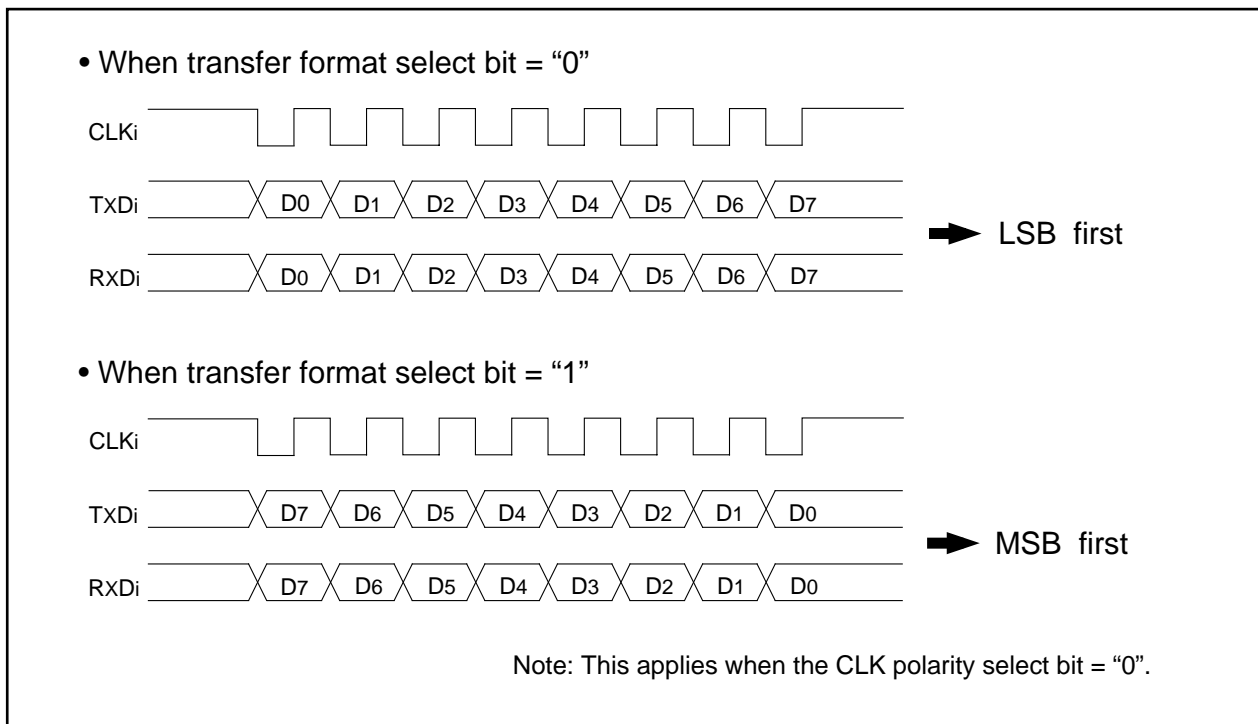


Figure 2.11.13 Transfer format

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

### (3) Transfer clock output from multiple pins function (UART1)

This function allows the setting two transfer clock output pins and choosing one of the two to output a clock by using the CLK and CLKS select bit (bits 4 and 5 at address 03B016). (See Figure 2.11.14) The multiple pins function is valid only when the internal clock is selected for UART1. Note that when this function is selected, UART1  $\overline{\text{CTS}}/\overline{\text{RTS}}$  function cannot be used.

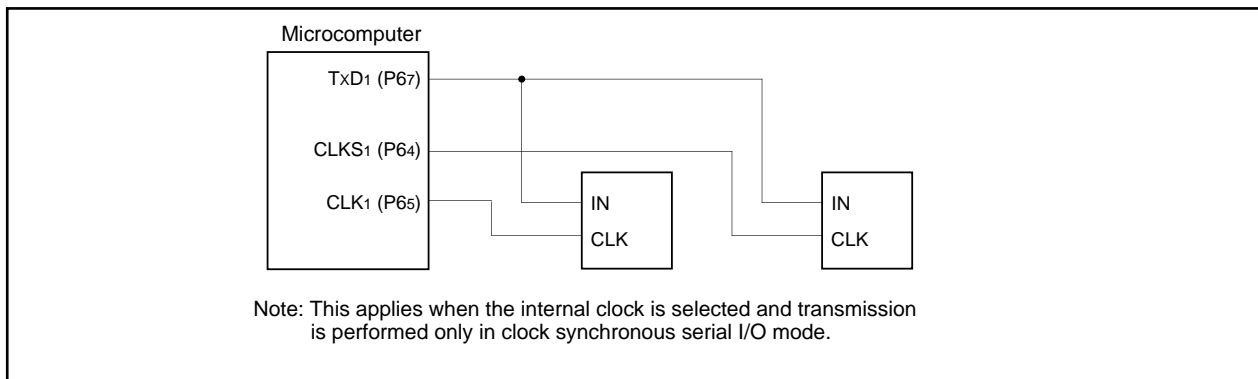


Figure 2.11.14 The transfer clock output from the multiple pins function usage

### (4) Continuous receive mode

If the continuous receive mode enable bit (bits 2 and 3 at address 03B016, bit 5 at address 037D16) is set to "1", the unit is placed in continuous receive mode. In this mode, when the receive buffer register is read out, the unit simultaneously goes to a receive enable state without having to set dummy data to the transmit buffer register back again.

### (5) Serial data logic switch function (UART2)

When the data logic select bit (bit6 at address 037D16) = "1", and writing to transmit buffer register or reading from receive buffer register, data is reversed. Figure 2.11.15 shows the example of serial data logic switch timing.

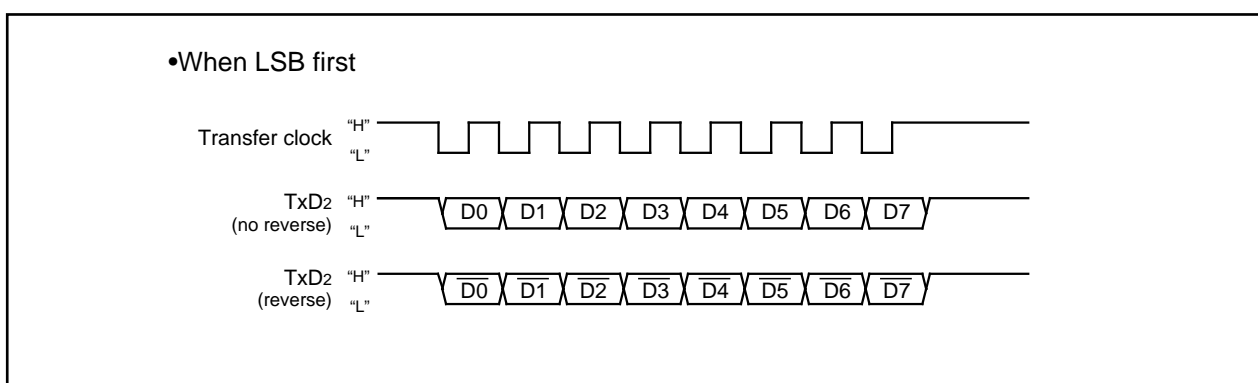


Figure 1.11.15 Serial data logic switch timing

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.11.3 Clock asynchronous serial I/O (UART) mode

The UART mode allows transmitting and receiving data after setting the desired transfer rate and transfer data format. Tables 2.11.5 and 2.11.6 list the specifications of the UART mode. Figure 2.11.16 shows the UARTi transmit/receive mode register.

**Table 2.11.5 Specifications of UART Mode (1)**

Item	Specification
Transfer data format	<ul style="list-style-type: none"> <li>• Character bit (transfer data): 7 bits, 8 bits, or 9 bits as selected</li> <li>• Start bit: 1 bit</li> <li>• Parity bit: Odd, even, or nothing as selected</li> <li>• Stop bit: 1 bit or 2 bits as selected</li> </ul>
Transfer clock	<ul style="list-style-type: none"> <li>• When internal clock is selected (bit 3 at addresses 03A0<sub>16</sub>, 03A8<sub>16</sub>, 0378<sub>16</sub> = "0") : <math>f_i/16(n+1)</math> (Note 1) <math>f_i = f_1, f_8, f_{32}</math></li> <li>• When external clock is selected (bit 3 at addresses 03A0<sub>16</sub> and 03A8<sub>16</sub> = "1") : <math>f_{EXT}/16(n+1)</math> (Note 1) (Note 2) (Do not set external clock for UART2)</li> </ul>
Transmission/reception control	<ul style="list-style-type: none"> <li>• CTS function/RTS function/CTS, RTS function chosen to be invalid</li> </ul>
Transmission start condition	<ul style="list-style-type: none"> <li>• To start transmission, the following requirements must be met: <ul style="list-style-type: none"> <li>- Transmit enable bit (bit 0 at addresses 03A5<sub>16</sub>, 03AD<sub>16</sub>, 037D<sub>16</sub>) = "1"</li> <li>- Transmit buffer empty flag (bit 1 at addresses 03A5<sub>16</sub>, 03AD<sub>16</sub>, 037D<sub>16</sub>) = "0"</li> <li>- When CTS function selected, CTS input level = "L"</li> </ul> </li> </ul>
Reception start condition	<ul style="list-style-type: none"> <li>• To start reception, the following requirements must be met: <ul style="list-style-type: none"> <li>- Receive enable bit (bit 2 at addresses 03A5<sub>16</sub>, 03AD<sub>16</sub>, 037D<sub>16</sub>) = "1"</li> <li>- Start bit detection</li> </ul> </li> </ul>
Interrupt request generation timing	<ul style="list-style-type: none"> <li>• When transmitting <ul style="list-style-type: none"> <li>- Transmit interrupt cause select bits (bits 0,1 at address 03B0<sub>16</sub>, bit4 at address 037D<sub>16</sub>) = "0": Interrupts requested when data transfer from UARTi transfer buffer register to UARTi transmit register is completed</li> <li>- Transmit interrupt cause select bits (bits 0, 1 at address 03B0<sub>16</sub>, bit4 at address 037D<sub>16</sub>) = "1": Interrupts requested when data transmission from UARTi transfer register is completed</li> </ul> </li> <li>• When receiving <ul style="list-style-type: none"> <li>- Interrupts requested when data transfer from UARTi receive register to UARTi receive buffer register is completed</li> </ul> </li> </ul>
Error detection	<ul style="list-style-type: none"> <li>• Overrun error (Note 3) This error occurs when the next data is ready before contents of UARTi receive buffer register are read out</li> <li>• Framing error This error occurs when the number of stop bits set is not detected</li> <li>• Parity error This error occurs when if parity is enabled, the number of 1's in parity and character bits does not match the number of 1's set</li> <li>• Error sum flag This flag is set (= 1) when any of the overrun, framing, and parity errors is encountered</li> </ul>

Note 1: 'n' denotes the value 00<sub>16</sub> to FF<sub>16</sub> that is set to the UARTi bit rate generator.

Note 2: f<sub>EXT</sub> is input from the CLKi pin.

Note 3: If an overrun error occurs, the UARTi receive buffer will have the next data written in. Note also that the UARTi receive interrupt request bit is not set to "1".

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

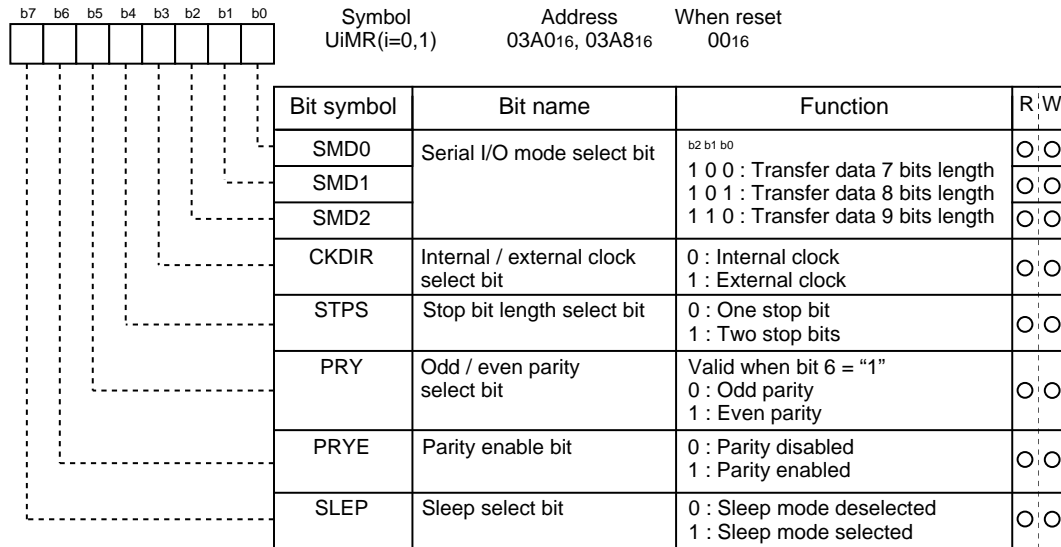
**Table 2.11.6 Specifications of UART Mode (2)**

Item	Specification
Select function	<ul style="list-style-type: none"> <li>• Sleep mode selection (UART0, UART1) This mode is used to transfer data to and from one of multiple slave micro-computers</li> <li>• Serial data logic switch (UART2) This function is reversing logic value of transferring data. Start bit, parity bit and stop bit are not reversed.</li> <li>• TxD, RxD I/O polarity switch (UART2) This function is reversing TxD port output and RxD port input. All I/O data level is reversed.</li> </ul>

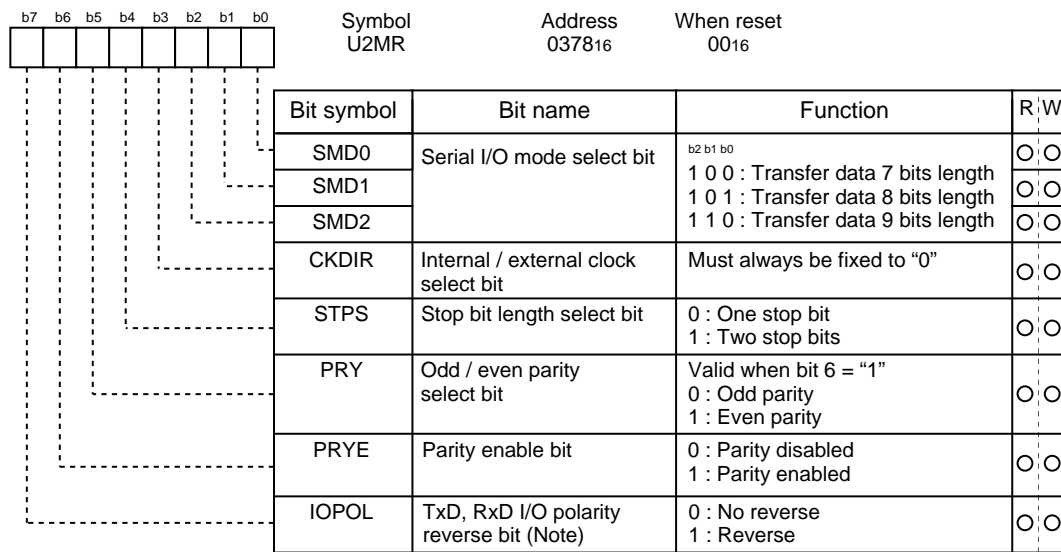
# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## UARTi transmit / receive mode registers



## UART2 transmit / receive mode register



Note: Usually set to "0".

**Figure 2.11.16 UARTi transmit/receive mode register in UART mode**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Table 2.11.7 lists the functions of the input/output pins during UART mode. Note that for a period from when the UARTi operation mode is selected to when transfer starts, the TxDi pin outputs a "H". (If the N-channel open-drain is selected, this pin is in floating state.)

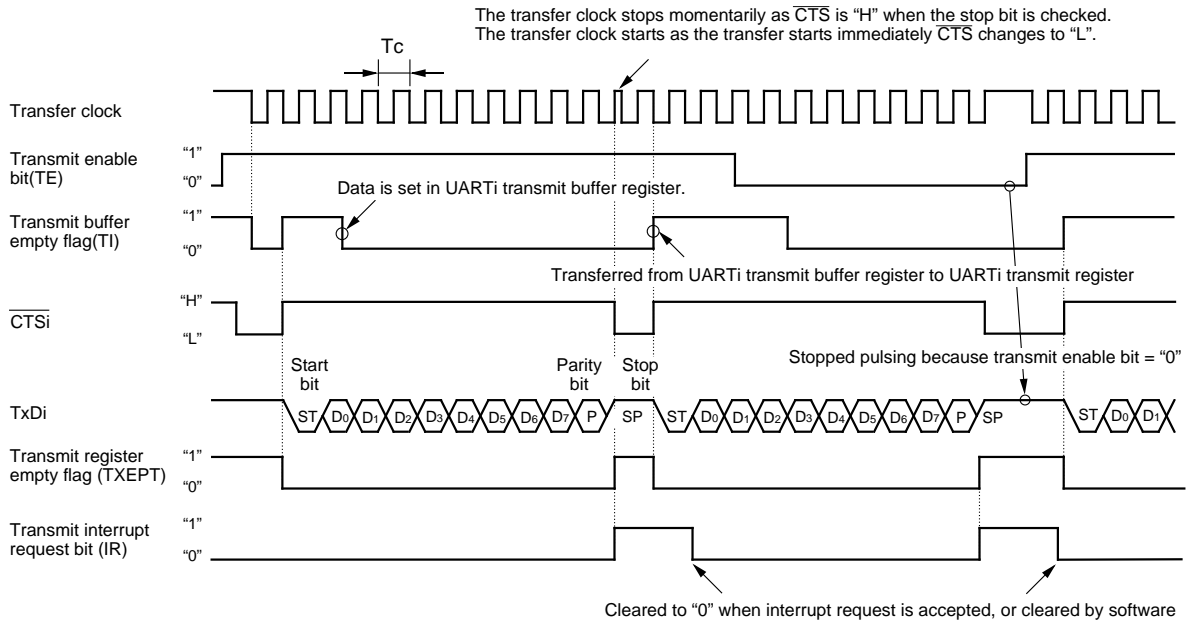
**Table 2.11.7 Input/output pin functions in UART mode**

Pin name	Function	Method of selection
TxDi (P63, P67, P70)	Serial data output	
RxDi (P62, P66, P71)	Serial data input	Port P62, P66 and P71 direction register (bits 2 and 6 at address 03EE16, bit 1 at address 03EF16) = "0" (Can be used as an input port when performing transmission only)
CLKi (P61, P65, P72)	Programmable I/O port	Internal/external clock select bit (bit 3 at address 03A016, 03A816, 037816) = "0"
	Transfer clock input	Internal/external clock select bit (bit 3 at address 03A016, 03A816) = "1" Port P61, P65 direction register (bits 1 and 5 at address 03EE16) = "0" (Do not set external clock for UART2)
CTS <sub>i</sub> /RTS <sub>i</sub> (P60, P64, P73)	CTS input	CTS/RTS disable bit (bit 4 at address 03A416, 03AC16, 037C16) = "0" CTS/RTS function select bit (bit 2 at address 03A416, 03AC16, 037C16) = "0" Port P60, P64 and P73 direction register (bits 0 and 4 at address 03EE16, bit 3 at address 03EF16) = "0"
	RTS output	CTS/RTS disable bit (bit 4 at address 03A416, 03AC16, 037C16) = "0" CTS/RTS function select bit (bit 2 at address 03A416, 03AC16, 037C16) = "1"
	Programmable I/O port	CTS/RTS disable bit (bit 4 at address 03A416, 03AC16, 037C16) = "1"

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## • Example of transmit timing when transfer data is 8 bits long (parity enabled, one stop bit)



## • Example of transmit timing when transfer data is 9 bits long (parity disabled, two stop bits)

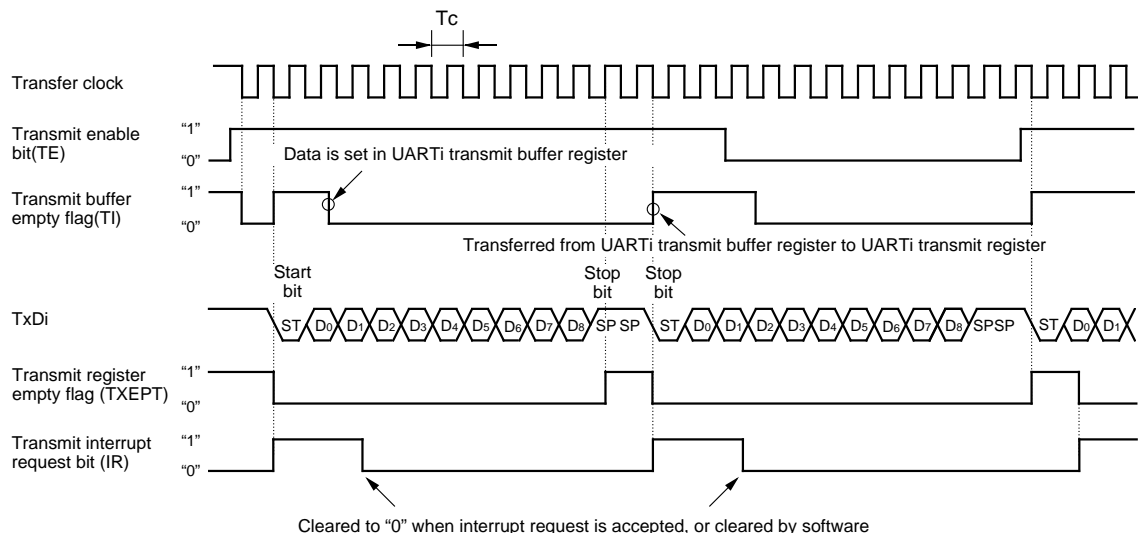


Figure 2.11.17 Typical transmit timings in UART mode(UART0,UART1)



M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

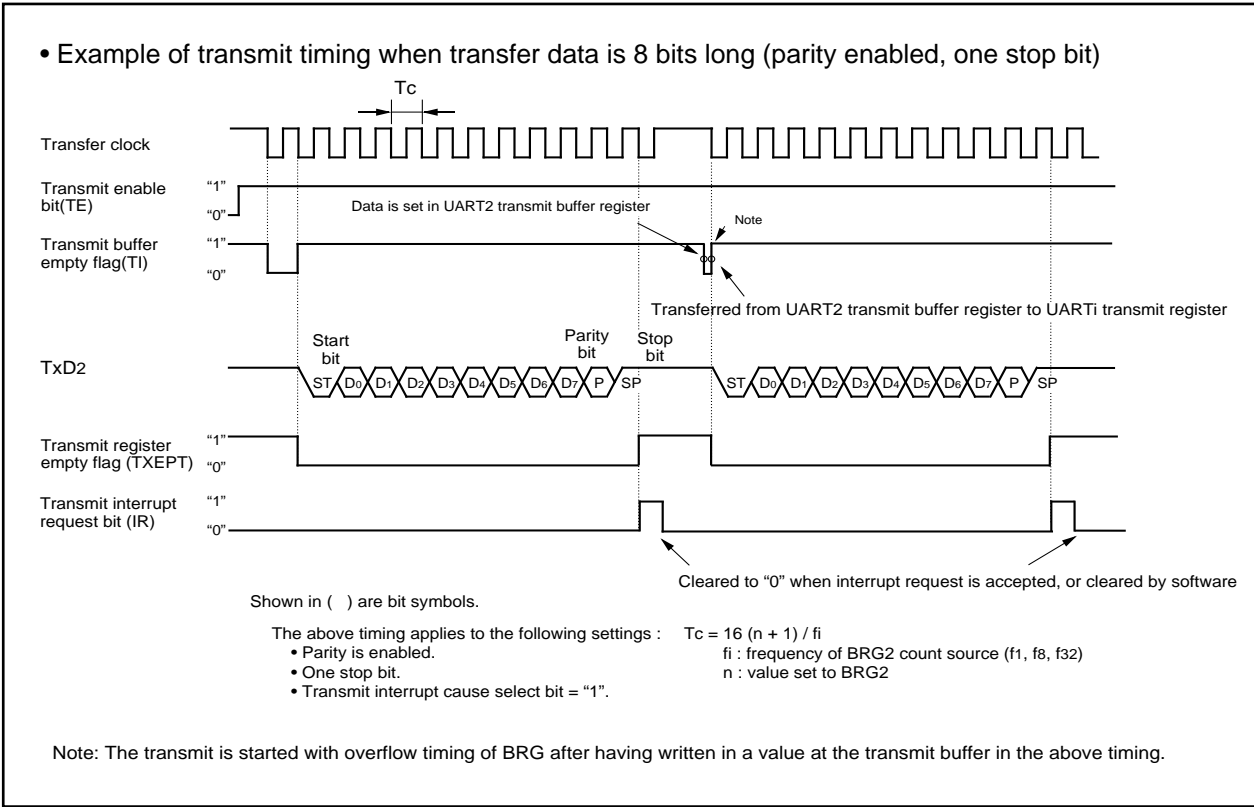


Figure 2.11.18 Typical transmit timings in UART mode(UART2)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

- Example of receive timing when transfer data is 8 bits long (parity disabled, one stop bit)

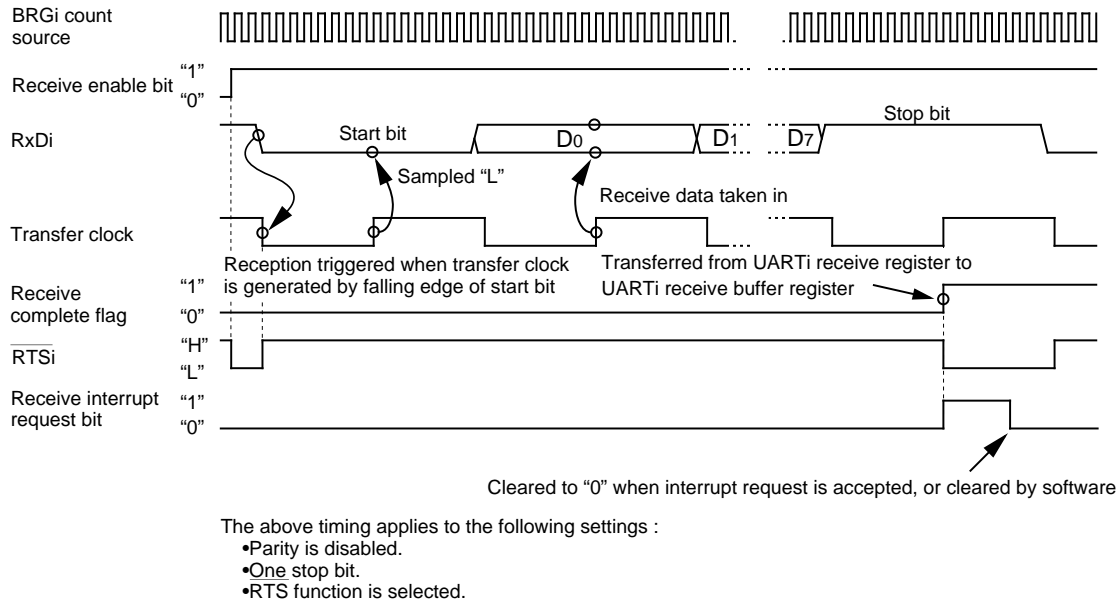


Figure 2.11.19 Typical receive timing in UART mode

## (1) Sleep mode (UART0, UART1)

This mode is used to transfer data between specific microcomputers among multiple microcomputers connected using UARTi. The sleep mode is selected when the sleep select bit (bit 7 at addresses 03A016, 03A816) is set to "1" during reception. In this mode, the unit performs receive operation when the MSB of the received data = "1" and does not perform receive operation when the MSB = "0".

## (2) Function for switching serial data logic (UART2)

When the data logic select bit (bit 6 of address 037D16) is assigned 1, data is inverted in writing to the transmission buffer register or reading the reception buffer register. Figure 2.11.20 shows the example of timing for switching serial data logic.

- When LSB first, parity enabled, one stop bit

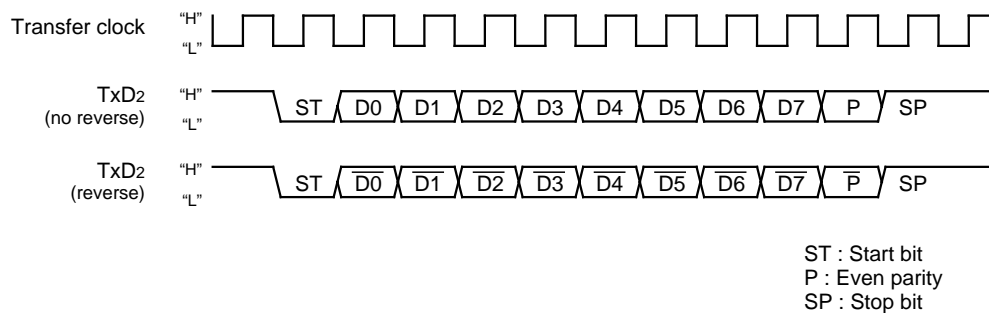


Figure 2.11.20 Timing for switching serial data logic

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

### (3) TxD, RxD I/O polarity reverse function (UART2)

This function is to reverse TxD pin output and RxD pin input. The level of any data to be input or output (including the start bit, stop bit(s), and parity bit) is reversed. Set this function to "0" (not to reverse) for usual use.

### (4) Bus collision detection function (UART2)

This function is to sample the output level of the TxD pin and the input level of the RxD pin at the rising edge of the transfer clock; if their values are different, then an interrupt request occurs. Figure 2.11.21 shows the example of detection timing of a buss collision (in UART mode).

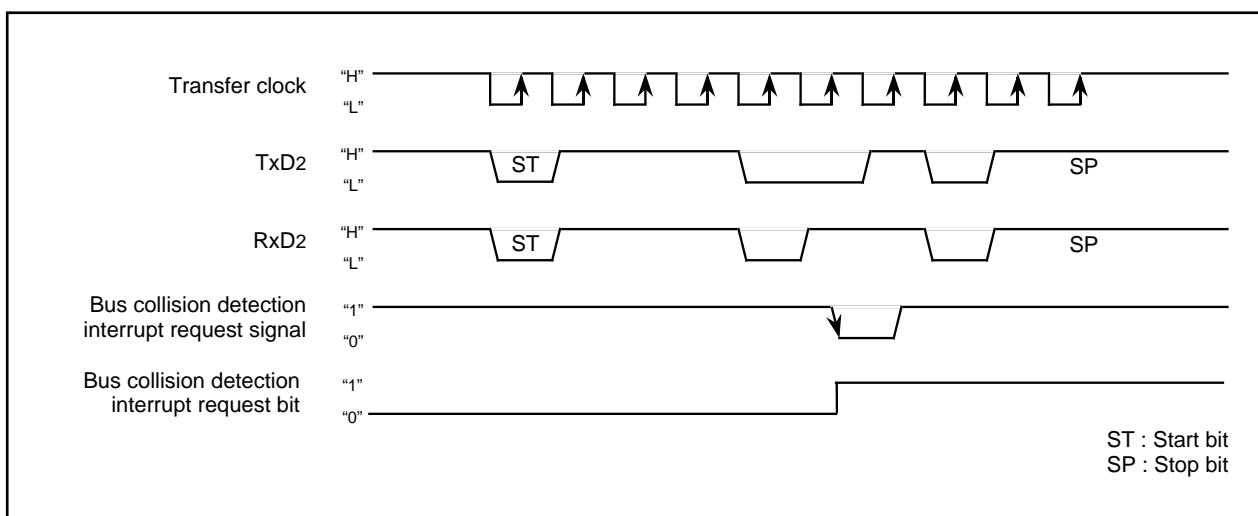


Figure 2.11.21 Detection timing of a bus collision (in UART mode)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.11.4 Clock-asynchronous serial I/O mode (compliant with the SIM interface)

The SIM interface is used for connecting the microcomputer with a memory card or the like; adding some extra settings in UART2 clock-asynchronous serial I/O mode allows the user to effect this function. Table 2.11.8 shows the specifications of clock-asynchronous serial I/O mode (compliant with the SIM interface).

**Table 2.11.8 Specifications of clock-asynchronous serial I/O mode (compliant with the SIM interface)**

Item	Specification
Transfer data format	<ul style="list-style-type: none"> <li>• Transfer data 8-bit UART mode (bit 2 through bit 0 of address 0378<sub>16</sub> = "1012")</li> <li>• One stop bit (bit 4 of address 0378<sub>16</sub> = "0")</li> <li>• With the direct format chosen <ul style="list-style-type: none"> <li>Set parity to "even" (bit 5 and bit 6 of address 0378<sub>16</sub> = "1" and "1" respectively)</li> <li>Set data logic to "direct" (bit 6 of address 037D<sub>16</sub> = "0").</li> <li>Set transfer format to LSB (bit 7 of address 037C<sub>16</sub> = "0").</li> </ul> </li> <li>• With the inverse format chosen <ul style="list-style-type: none"> <li>Set parity to "odd" (bit 5 and bit 6 of address 0378<sub>16</sub> = "0" and "1" respectively)</li> <li>Set data logic to "inverse" (bit 6 of address 037D<sub>16</sub> = "1")</li> <li>Set transfer format to MSB (bit 7 of address 037C<sub>16</sub> = "1")</li> </ul> </li> </ul>
Transfer clock	<ul style="list-style-type: none"> <li>• With the internal clock chosen (bit 3 of address 0378<sub>16</sub> = "0") : <math>f_i / 16 (n + 1)</math> (Note 1) : <math>f_i = f_1, f_8, f_{32}</math> (Do not set external clock)</li> </ul>
Transmission / reception control	<ul style="list-style-type: none"> <li>• Disable the CTS and RTS function (bit 4 of address 037C<sub>16</sub> = "1")</li> </ul>
Other settings	<ul style="list-style-type: none"> <li>• The sleep mode select function is not available for UART2</li> <li>• Set transmission interrupt factor to "transmission completed" (bit 4 of address 037D<sub>16</sub> = "1")</li> </ul>
Transmission start condition	<ul style="list-style-type: none"> <li>• To start transmission, the following requirements must be met: <ul style="list-style-type: none"> <li>- Transmit enable bit (bit 0 of address 037D<sub>16</sub>) = "1"</li> <li>- Transmit buffer empty flag (bit 1 of address 037D<sub>16</sub>) = "0"</li> </ul> </li> </ul>
Reception start condition	<ul style="list-style-type: none"> <li>• To start reception, the following requirements must be met: <ul style="list-style-type: none"> <li>- Reception enable bit (bit 2 of address 037D<sub>16</sub>) = "1"</li> <li>- Detection of a start bit</li> </ul> </li> </ul>
Interrupt request generation timing	<ul style="list-style-type: none"> <li>• When transmitting <ul style="list-style-type: none"> <li>When data transmission from the UART2 transfer register is completed (bit 4 of address 037D<sub>16</sub> = "1")</li> </ul> </li> <li>• When receiving <ul style="list-style-type: none"> <li>When data transfer from the UART2 receive register to the UART2 receive buffer register is completed</li> </ul> </li> </ul>
Error detection	<ul style="list-style-type: none"> <li>• Overrun error (see the specifications of clock-asynchronous serial I/O) (Note 2)</li> <li>• Framing error (see the specifications of clock-asynchronous serial I/O)</li> <li>• Parity error (see the specifications of clock-asynchronous serial I/O) <ul style="list-style-type: none"> <li>- On the reception side, an "L" level is output from the TxD<sub>2</sub> pin by use of the parity error signal output function (bit 7 of address 037D<sub>16</sub> = "1") when a parity error is detected</li> <li>- On the transmission side, a parity error is detected by the level of input to the RxD<sub>2</sub> pin when a transmission interrupt occurs</li> </ul> </li> <li>• The error sum flag (see the specifications of clock-asynchronous serial I/O)</li> </ul>

Note 1: 'n' denotes the value 00<sub>16</sub> to FF<sub>16</sub> that is set to the UARTi bit rate generator.

Note 2: If an overrun error occurs, the UART2 receive buffer will have the next data written in. Note also that the UARTi receive interrupt request bit is not set to "1".

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

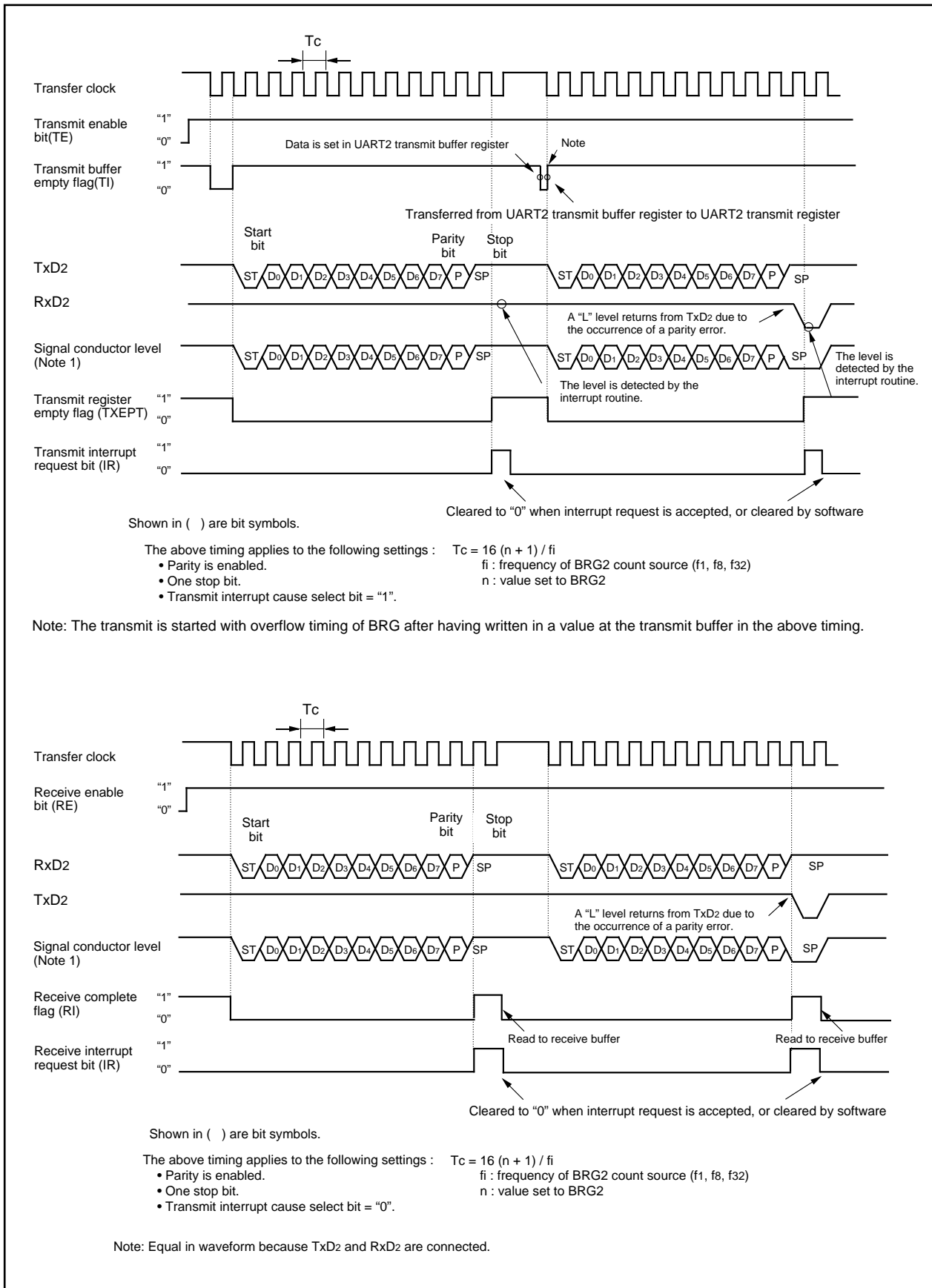


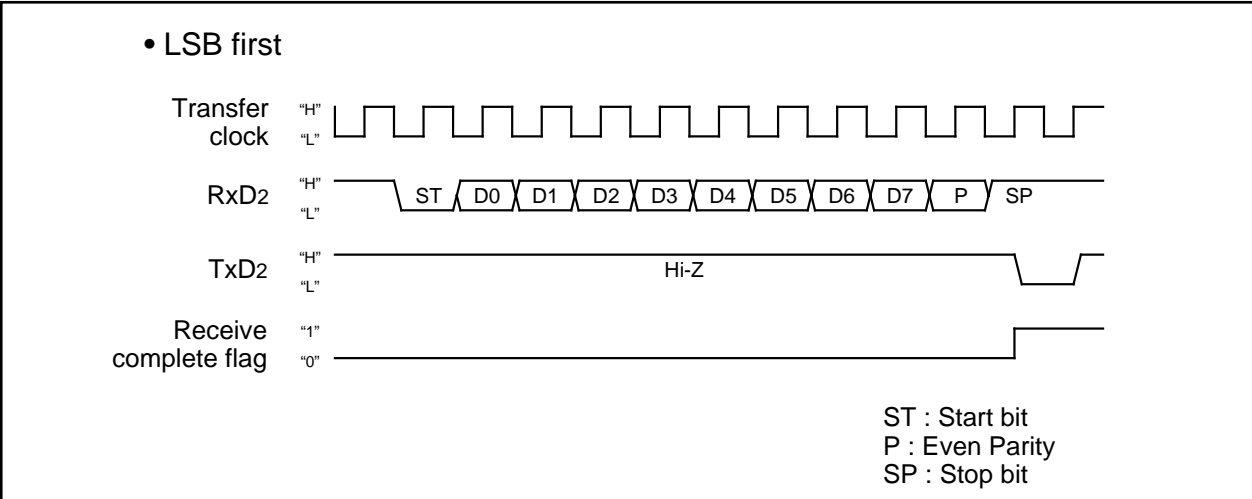
Figure 2.11.22 Typical transmit/receive timing in UART mode (compliant with the SIM interface)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**(1) Function for outputting a parity error signal**

With the error signal output enable bit (bit 7 of address 037D16) assigned “1”, you can output an “L” level from the TxD2 pin when a parity error is detected. In step with this function, the generation timing of a transmission completion interrupt changes to the detection timing of a parity error signal. Figure 2.11.23 shows the output timing of the parity error signal.

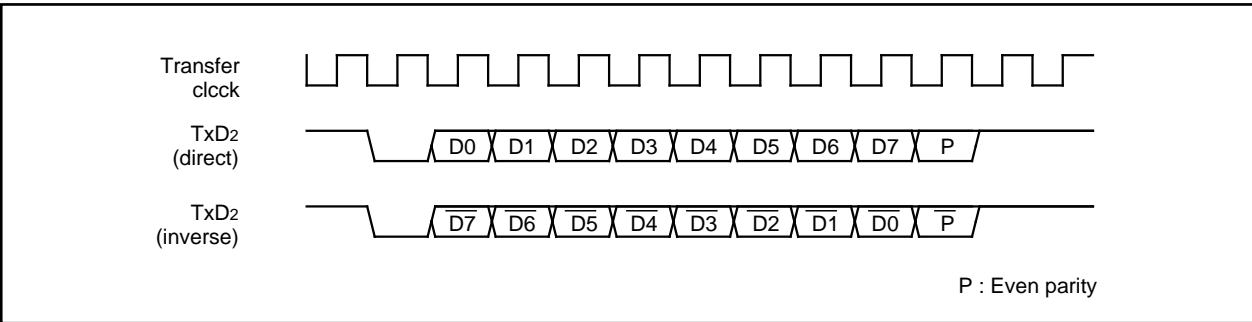


**Figure 2.11.23 Output timing of the parity error signal**

**(2) Direct format/inverse format**

Connecting the SIM card allows you to switch between direct format and inverse format. If you choose the direct format, D0 data is output from TxD2. If you choose the inverse format, D7 data is inverted and output from TxD2.

Figure 2.11.24 shows the SIM interface format.

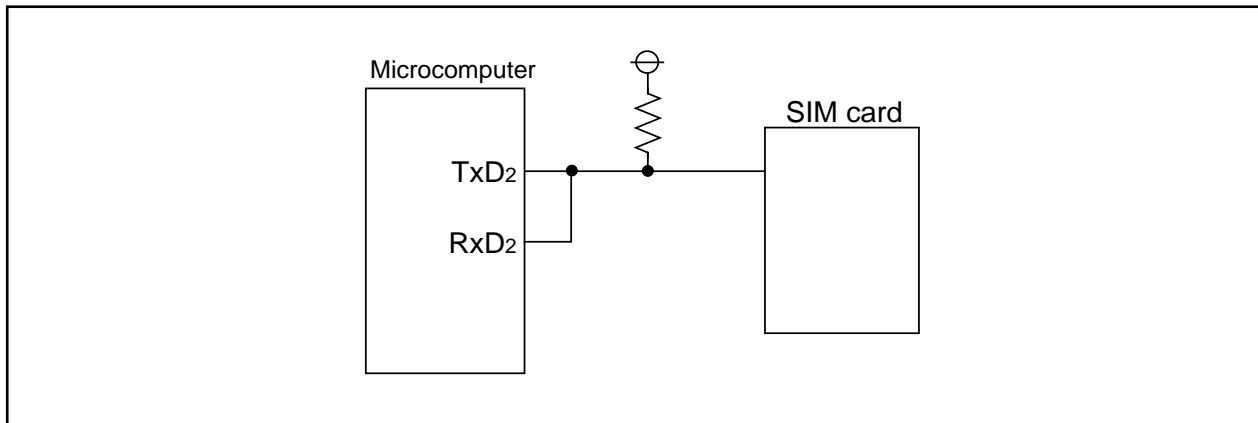


**Figure 2.11.24 SIM interface format**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
 with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Figure 2.11.25 shows the example of connecting the SIM interface. Connect TxD<sub>2</sub> and RxD<sub>2</sub> and apply pull-up.



**Figure 2.11.25 Connecting the SIM interface**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.11.5 UART2 Special Mode Register

The UART2 special mode register (address 0377<sub>16</sub>) is used to control UART2 in various ways.

Figure 2.11.26 shows the UART2 special mode register.

In the first place, the control bits related to the I<sup>2</sup>C bus(simplified I<sup>2</sup>C bus) interface are explained.

Bit 0 of the UART special mode register (0377<sub>16</sub>) is used as the I<sup>2</sup>C mode selection bit.

Setting "1" in the I<sup>2</sup>C mode select bit (bit 0) goes the circuit to achieve the I<sup>2</sup>C bus interface effective.

Since this function uses clock-synchronous serial I/O mode, set this bit to "0" in UART mode.

UART2 special mode register

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
b7	IICM	I <sup>2</sup> C mode selection bit 0 : Normal mode 1 : I <sup>2</sup> C mode	Must always be "0"	○	○
b6	ABC	Arbitration lost detecting flag control bit 0 : Update per bit 1 : Update per byte	Must always be "0"	○	○
b5	BBS	Bus busy flag 0 : STOP condition detected 1 : START condition detected	Must always be "0"	○	○
b4	LSYN	SCLL sync output enable bit 0 : Disabled 1 : Enabled	Must always be "0"	○	○
b3	ABSCS	Bus collision detect sampling clock select bit	Must always be "0"	○	○
b2	ACSE	Auto clear function select bit of transmit enable bit	Must always be "0"	○	○
b1	SSS	Transmit start condition select bit	0 : Ordinary 1 : Falling edge of Rx/D2	○	○
b0	SDDS	SDA digital delay select bit (Notes 2 and 3) 0 : Selects analog delay output 1 : Selects digital delay output (Must always be "0" except at I <sup>2</sup> C mode)	Must always be "0"	○	○

- Notes 1: Nothing but "0" may be written.  
2: Do not write "1" except at I<sup>2</sup>C mode. Must always be "0" at normal mode.  
Bit 7 to bit5 (DL2 to DL0 = SDA digital delay value setting bit) of UART2 special mode register 3 (U2SMR3/address 0375<sub>16</sub>) are initialized and become "000" when this bit is "0", analog delay circuit is selected. Reading and writing U2SMR are enable when SDDS = "0".  
3: Delaying ; Only analog delay value when analog delay is selected, and only digital delay value when digital delay is selected.

UART2 special mode register 3 (I<sup>2</sup>C bus exclusive register)

Bit symbol	Bit name	Function (I <sup>2</sup> C bus exclusive)	R	W
b7	DL0	SDA digital delay value set bit b7 b6 b5 0 0 0 : Selects analog delay 0 0 1 : 2 cycle of 1/f (Xin)(Digital delay) 0 1 0 : 3 cycle of 1/f (Xin)(Digital delay) 0 1 1 : 4 cycle of 1/f (Xin)(Digital delay) 1 0 0 : 5 cycle of 1/f (Xin)(Digital delay) 1 0 1 : 6 cycle of 1/f (Xin)(Digital delay) 1 1 0 : 7 cycle of 1/f (Xin)(Digital delay) 1 1 1 : 8 cycle of 1/f (Xin)(Digital delay)	○	○
b6	DL1		○	○
b5	DL2		○	○

- Notes 1: Reading and writing is possible when bit7 (SDDS = SDA digital delay selection bit) of UART2 special mode register (U2SMR/address 0377<sub>16</sub>) is "1". When set SDDS = "1" and read out initialized value of UART2 special mode register 3(U2SMR3), this value is "000".When set SDDS = "1" and write to UART2 special mode register 3(U2SMR3), set "0" to bit 0 to bit 4. When SDDS = "0", writing is enable. When read out, this value is indeterminate.  
2: When SDDS = "0", this bit is initialized and become "000", selected analog delay circuit. This bit is become "000" after end reset released, and selected analog delay circuit. Reading out is possible when only SDDS = "1". when SDDS = "0", value which was read out is indeterminate.  
3: Delaying ; Only analog delay value when analog delay is selected, and only digital delay value when digital delay is selected.  
4: Delay level depends on SCL pin and SDA pin. And, when use external clock, delay is increase around 100ns. So test first, and use this.

Figure 2.11.26 UART2 special mode register



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

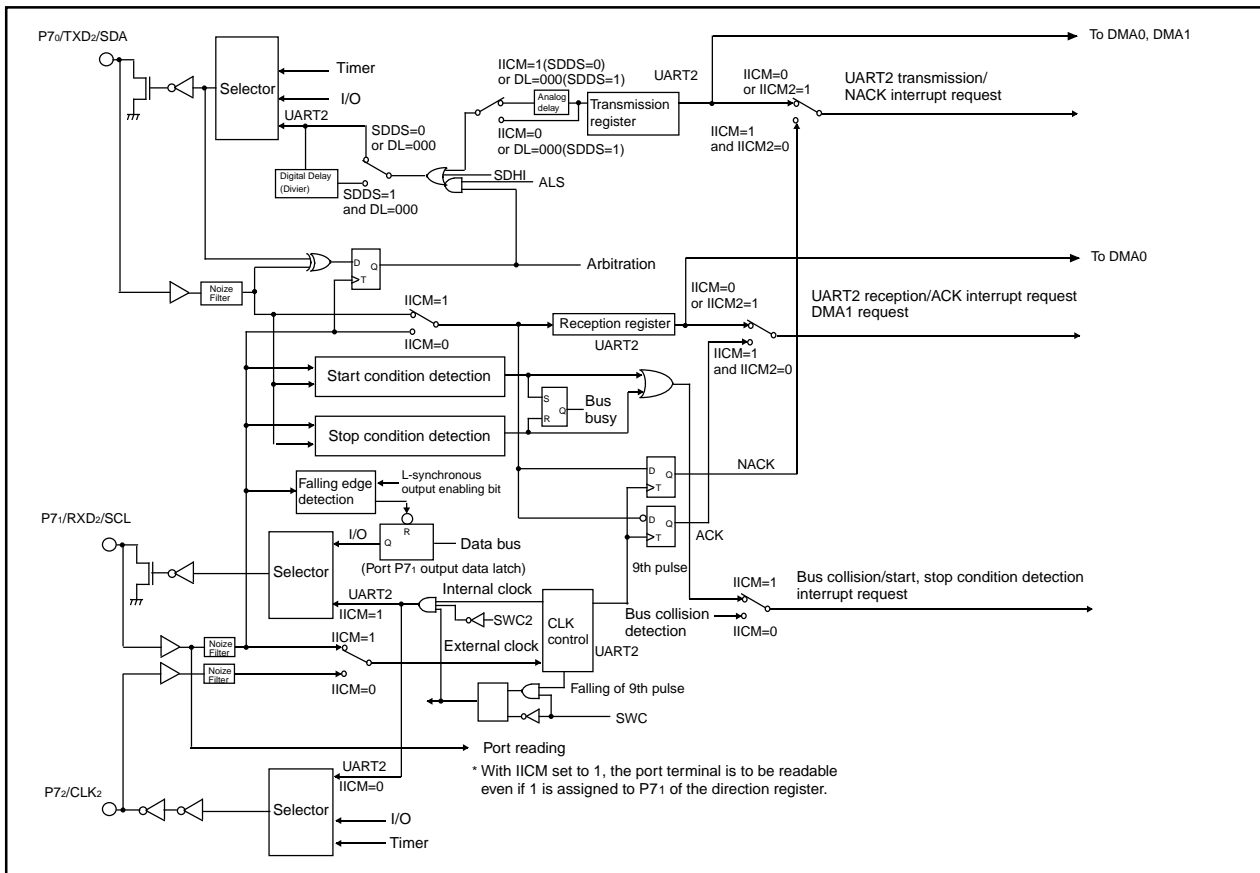


Figure 2.11.27 Functional block diagram for I²C mode

Table 2.11.9 Features in I²C mode

	Function	Normal mode	I²C mode (Note 1)
1	Factor of interrupt number 10 (Note 2)	Bus collision detection	Start condition detection or stop condition detection
2	Factor of interrupt number 15 (Note 2)	UART2 transmission	No acknowledgment detection (NACK)
3	Factor of interrupt number 16 (Note 2)	UART2 reception	Acknowledgment detection (ACK)
4	UART2 transmission output delay	Not delayed	Delayed
5	P70 at the time when UART2 is in use	TxD2 (output)	SDA (input/output) (Note 3)
6	P71 at the time when UART2 is in use	RxD2 (input)	SCL (input/output)
7	P72 at the time when UART2 is in use	CLK2	P72
8	DMA1 factor at the time when 1 1 0 1 is assigned to the DMA request factor selection bits	UART2 reception	Acknowledgment detection (ACK)
9	Noise filter width	15ns	50ns
10	Reading P71	Reading the terminal when 0 is assigned to the direction register	Reading the terminal regardless of the value of the direction register
11	Initial value of UART2 output	H level (when 0 is assigned to the CLK polarity select bit)	The value set in latch P70 when the port is selected

Note 1: Make the settings given below when I²C mode is in use.

Set 0 1 0 in bits 2, 1, 0 of the UART2 transmission/reception mode register.

Disable the RTS/CTS function. Choose the MSB First function.

Note 2: Follow the steps given below to switch from a factor to another.

1. Disable the interrupt of the corresponding number.
2. Switch from a factor to another.
3. Reset the interrupt request flag of the corresponding number.
4. Set an interrupt level of the corresponding number.

Note 3: Set an initial value of SDA transmission output when serial I/O is invalid.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Figure 2.11.27 shows the functional block diagram for I<sup>2</sup>C mode. Setting "1" in the I<sup>2</sup>C mode selection bit (IICM) causes ports P70, P71, and P72 to work as data transmission-reception terminal SDA, clock input-output terminal SCL, and port P72 respectively. A delay circuit is added to the SDA transmission output, so the SDA output changes after SCL fully goes to "L". Can select analog delay or digital delay by SDA digital delay selection bit (7 bit of address 037716). When select digital delay, can select delay to 2 cycle to 8 cycle of f1 by UART2 special mode register 3 (address 037516). Functions changed by I<sup>2</sup>C mode selection bit 2 is shown in below.

**Table 2.11.10 Delay circuit selection condition**

	Register value			Contents
	IICM	SDDS	DL	
Digital delay selection	1	1	001 ? 111	When select digital delay, analog delay is not added. Only digital delay.
Analog delay selection	1	1	000	When select DL="000", analog delay is chosen regardless of the value of SDDS.
		0	(000)	When SDDS="0", DL is initialized and DL="000".
No delay	0	0	(000)	Delay circuit is not selected when IICM="0". But, must set SDDS="0" when IICM="0".

An attempt to read Port P71 (SCL) results in getting the terminal's level regardless of the content of the port direction register. The initial value of SDA transmission output in this mode goes to the value set in port P70. The interrupt factors of the bus collision detection interrupt, UART2 transmission interrupt, and of UART2 reception interrupt turn to the start/stop condition detection interrupt, acknowledgment non-detection interrupt, and acknowledgment detection interrupt respectively.

The start condition detection interrupt refers to the interrupt that occurs when the falling edge of the SDA terminal (P70) is detected with the SCL terminal (P71) staying "H". The stop condition detection interrupt refers to the interrupt that occurs when the rising edge of the SDA terminal (P70) is detected with the SCL terminal (P71) staying "H". The bus busy flag (bit 2 of the UART2 special mode register) is set to "1" by the start condition detection, and set to "0" by the stop condition detection.

The acknowledgment non-detection interrupt refers to the interrupt that occurs when the SDA terminal level is detected still staying "H" at the rising edge of the 9th transmission clock. The acknowledgment detection interrupt refers to the interrupt that occurs when SDA terminal's level is detected already went to "L" at the 9th transmission clock. Also, assigning 1101(UART2 reception) to the DMA1 request factor select bits provides the means to start up the DMA transfer by the effect of acknowledgment detection.

Bit 1 of the UART2 special mode register (037716) is used as the arbitration loss detecting flag control bit. Arbitration means the act of detecting the nonconformity between transmission data and SDA terminal data at the timing of the SCL rising edge. This detecting flag is located at bit 3 of the UART2 reception buffer register (037F16), and "1" is set in this flag when nonconformity is detected. Use the arbitration lost detecting flag control bit to choose which way to use to update the flag, bit by bit or byte by byte. When setting this bit to "1" and updated the flag byte by byte if nonconformity is detected, the arbitration lost detecting flag is set to "1" at the falling edge of the 9th transmission clock.

If update the flag byte by byte, must judge and clear ("0") the arbitration lost detecting flag after completing the first byte acknowledge detect and before starting the next one byte transmission.

Bit 3 of the UART2 special mode register is used as SCL- and L-synchronous output enable bit. Setting this bit to "1" goes the P71 data register to "0" in synchronization with the SCL terminal level going to "L".

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Some other functions added are explained here. Figure 2.11.28 shows their workings.

Bit 4 of the UART2 special mode register is used as the bus collision detect sampling clock select bit. The bus collision detect interrupt occurs when the RxD2 level and TxD2 level do not match, but the nonconformity is detected in synchronization with the rising edge of the transfer clock signal if the bit is set to "0". If this bit is set to "1", the nonconformity is detected at the timing of the overflow of timer A0 rather than at the rising edge of the transfer clock.

Bit 5 of the UART2 special mode register is used as the auto clear function select bit of transmit enable bit. Setting this bit to "1" automatically resets the transmit enable bit to "0" when "1" is set in the bus collision detect interrupt request bit (nonconformity).

Bit 6 of the UART2 special mode register is used as the transmit start condition select bit. Setting this bit to "1" starts the TxD transmission in synchronization with the falling edge of the RxD terminal.

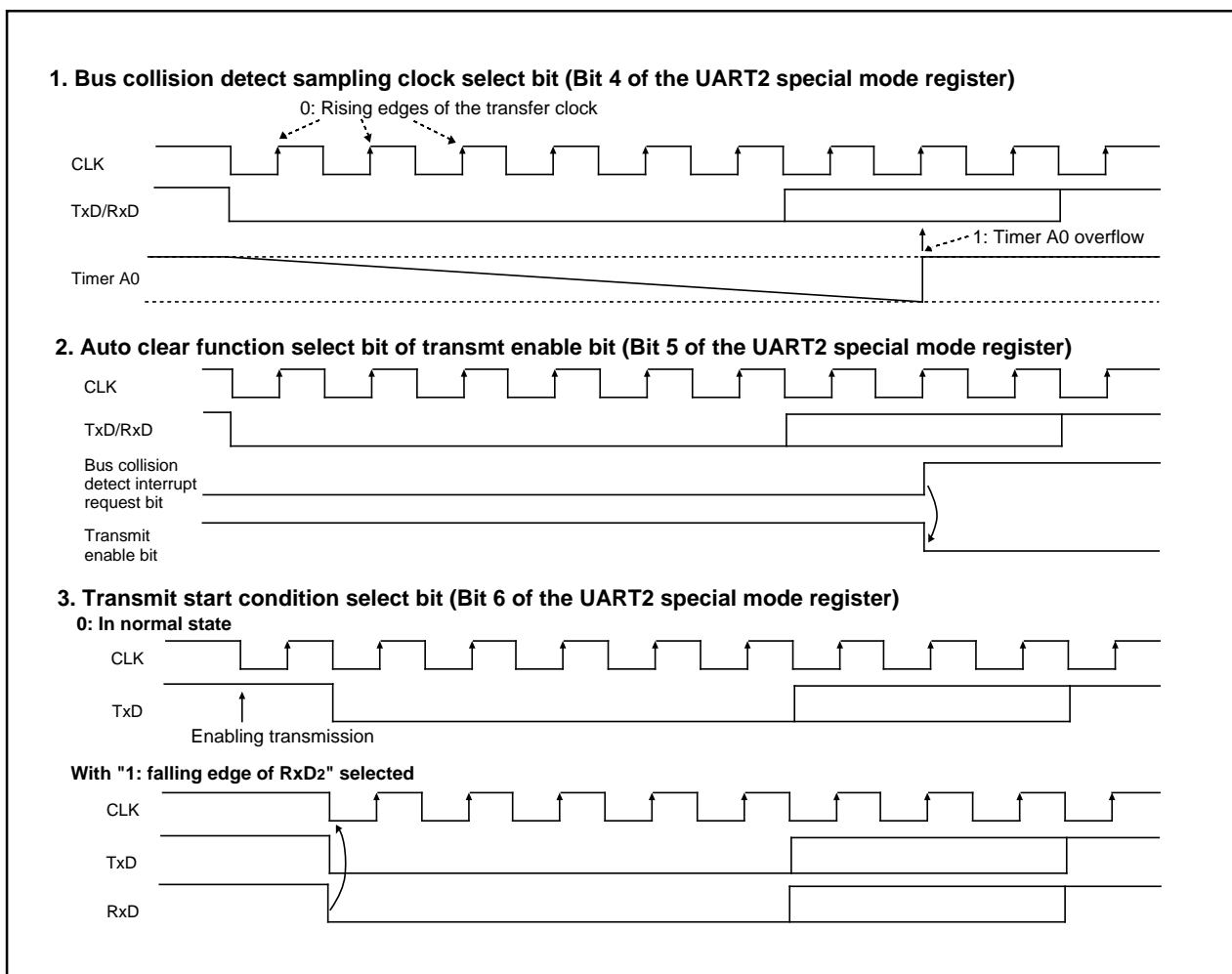


Figure 2.11.28 Some other functions added

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.11.6 UART2 Special Mode Register 2

UART2 special mode register 2 (address 0376<sub>16</sub>) is used to further control UART2 in I<sup>2</sup>C mode. Figure 2.11.29 shows the UART2 special mode register 2.

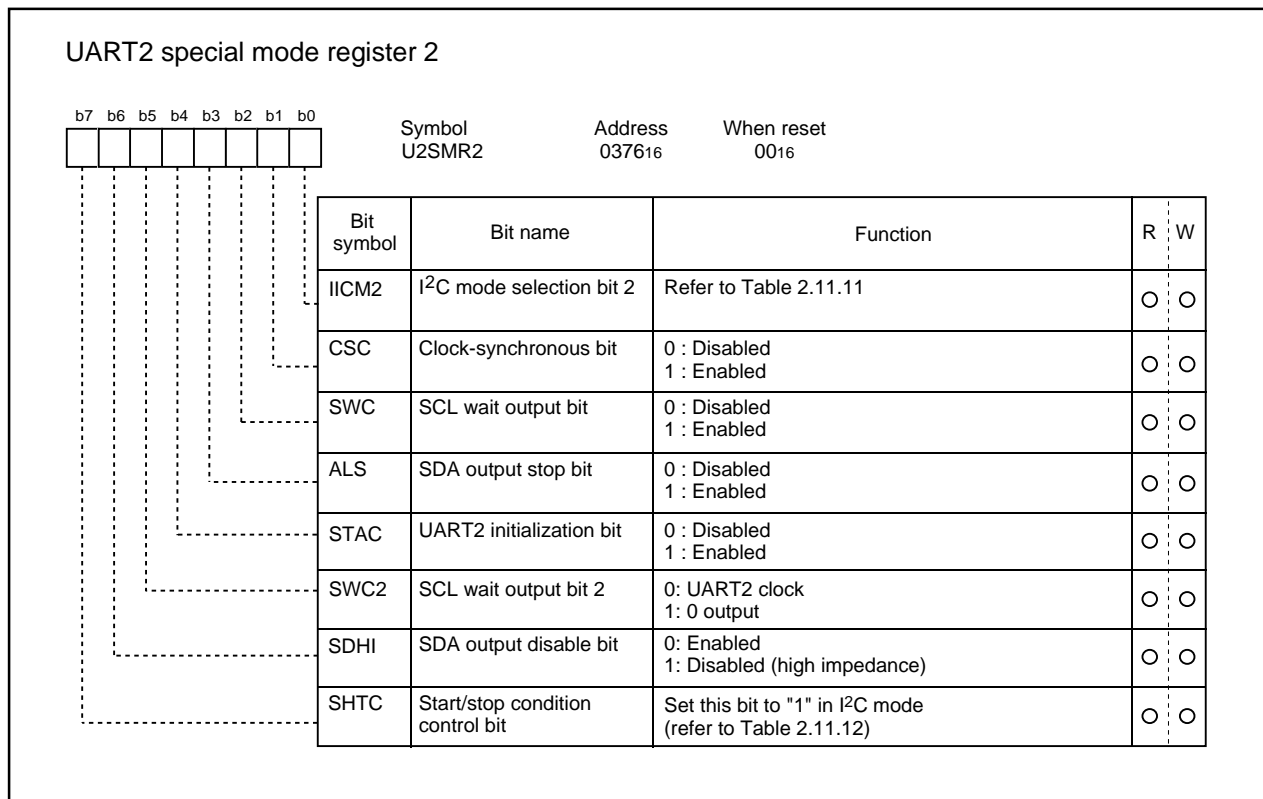


Figure 2.11.29 UART2 special mode register 2

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Bit 0 of the UART2 special mode register 2 (address 0376<sub>16</sub>) is used as the I<sup>2</sup>C mode selection bit 2. Table 2.11.11 shows the types of control to be changed by I<sup>2</sup>C mode selection bit 2 when the I<sup>2</sup>C mode selection bit is set to "1". Table 2.11.12 shows the timing characteristics of detecting the start condition and the stop condition. Set the start/stop condition control bit (bit 7 of UART2 special mode register 2) to "1" in I<sup>2</sup>C mode.

**Table 2.11.11 Functions changed by I<sup>2</sup>C mode selection bit 2**

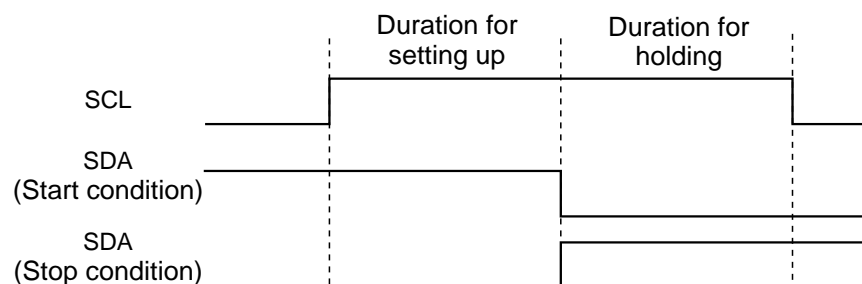
	Function	IICM2 = 0	IICM2 = 1
1	Factor of interrupt number 15	No acknowledgment detection (NACK)	UART2 transmission (the rising edge of the final bit of the clock)
2	Factor of interrupt number 16	Acknowledgment detection (ACK)	UART2 reception (the falling edge of the final bit of the clock)
3	DMA1 factor at the time when 1 1 0 1 is assigned to the DMA request factor selection bits	Acknowledgment detection (ACK)	UART2 reception (the falling edge of the final bit of the clock)
4	Timing for transferring data from the UART2 reception shift register to the reception buffer.	The rising edge of the final bit of the reception clock	The falling edge of the final bit of the reception clock
5	Timing for generating a UART2 reception/ACK interrupt request	The rising edge of the final bit of the reception clock	The falling edge of the final bit of the reception clock

**Table 2.11.12 Timing characteristics of detecting the start condition and the stop condition(Note1)**

3 to 6 cycles < duration for setting-up (Note2)
3 to 6 cycles < duration for holding (Note2)

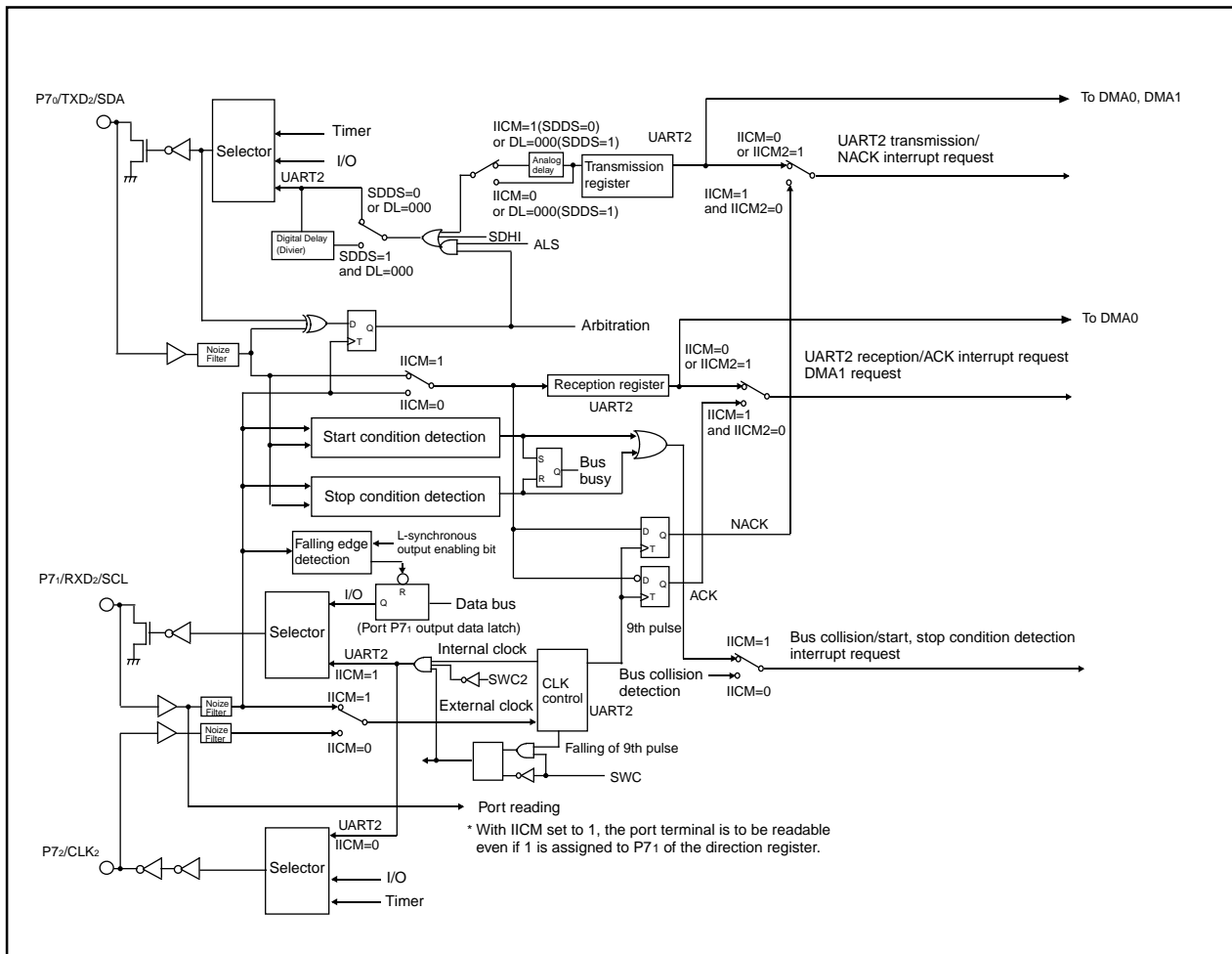
Note 1 : When the start/stop condition count bit is "1" .

Note 2 : "cycles" is in terms of the input oscillation frequency f(X<sub>IN</sub>) of the main clock.



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER



**Figure 2.11.30 Functional block diagram for I<sup>2</sup>C mode**

Functions available in I<sup>2</sup>C mode are shown in Figure 2.11.30—a functional block diagram.

Bit 3 of the UART2 special mode register 2 (address 0376<sub>16</sub>) is used as the SDA output stop bit. Setting this bit to "1" causes an arbitration loss to occur, and the SDA pin turns to high-impedance state the instant when the arbitration loss detection flag is set to "1".

Bit 1 of the UART2 special mode register 2 (address 036716) is used as the clock synchronization bit. With this bit set to "1" at the time when the internal SCL is set to "H", the internal SCL turns to "L" if the falling edge is found in the SCL pin; and the baud rate generator reloads the set value, and start counting within the "L" interval. When the internal SCL changes from "L" to "H" with the SCL pin set to "L", stops counting the baud rate generator, and starts counting it again when the SCL pin turns to "H". Due to this function, the UART2 transmission-reception clock becomes the logical product of the signal flowing through the internal SCL and that flowing through the SCL pin. This function operates over the period from the moment earlier by a half cycle than falling edge of the UART2 first clock to the rising edge of the ninth bit. To use this function, choose the internal clock for the transfer clock.

Bit 2 of the UART2 special mode register 2 (0376<sub>16</sub>) is used as the SCL wait output bit. Setting this bit to "1" causes the SCL pin to be fixed to "L" at the falling edge of the ninth bit of the clock. Setting this bit to "0" frees the output fixed to "L".

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

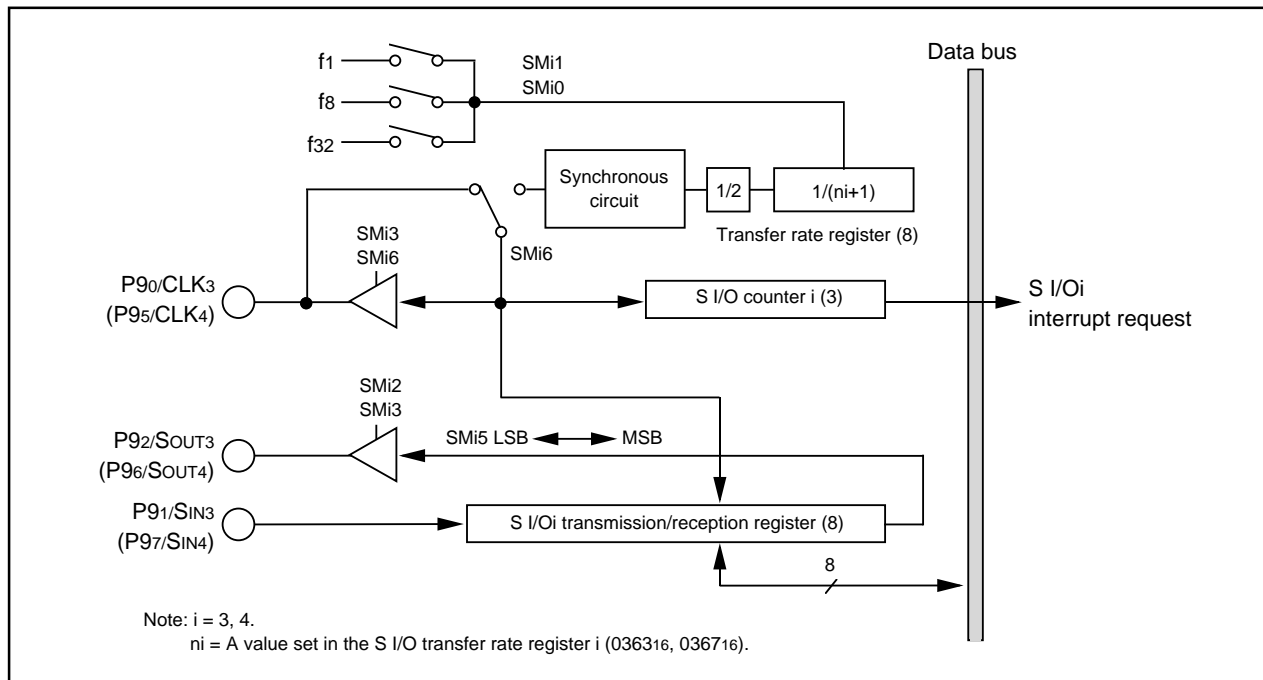
Bit 4 of the UART2 special mode register 2 (address 0376<sub>16</sub>) is used as the UART2 initialization bit. Setting this bit to "1", and when the start condition is detected, the microcomputer operates as follows.

- (1) The transmission shift register is initialized, and the content of the transmission register is transferred to the transmission shift register. This starts transmission by dealing with the clock entered next as the first bit. The UART2 output value, however, doesn't change until the first bit data is output after the entrance of the clock, and remains unchanged from the value at the moment when the microcomputer detected the start condition.
- (2) The reception shift register is initialized, and the microcomputer starts reception by dealing with the clock entered next as the first bit.
- (3) The SCL wait output bit turns to "1". This turns the SCL pin to "L" at the falling edge of the ninth bit of the clock.

Starting to transmit/receive signals to/from UART2 using this function doesn't change the value of the transmission buffer empty flag. To use this function, choose the external clock for the transfer clock. Bit 5 of the UART2 special mode register 2 (0376<sub>16</sub>) is used as the SCL pin wait output bit 2. Setting this bit to "1" with the serial I/O specified allows the user to forcibly output an "L" from the SCL pin even if UART2 is in operation. Setting this bit to "0" frees the "L" output from the SCL pin, and the UART2 clock is input/output.

Bit 6 of the UART2 special mode register 2 (0376<sub>16</sub>) is used as the SDA output enable bit. Setting this bit to "1" forces the SDA pin to turn to the high-impedance state. Refrain from changing the value of this bit at the rising edge of the UART2 transfer clock. There can be instances in which arbitration lost detection flag is turned on.

Figure 2.11.31 shows the S I/O3, 4 block diagram, and Figure 2.11.32 shows the S I/O3, 4 control register. Table 2.11.13 shows the specifications of S I/O3, 4.



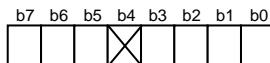
**Figure 2.11.31 S I/O3, 4 block diagram**



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## S I/Oi control register (i = 3, 4) (Note 1)



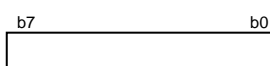
Symbol	Address	When reset
SiC	0362 <sub>16</sub> , 0366 <sub>16</sub>	40 <sub>16</sub>

Bit symbol	Bit name	Description	R	W
SMi0	Internal synchronous clock select bit	b1 b0 0 0 : Selecting f <sub>1</sub> 0 1 : Selecting f <sub>8</sub> 1 0 : Selecting f <sub>32</sub> 1 1 : Not to be used	○	○
SMi1			○	○
SMi2	Souri output disable bit	0 : Souri output 1 : Souri output disable(high impedance)	○	○
SMi3	S I/Oi port select bit (Note 2)	0 : Input-output port 1 : Souri output, CLK function	○	○
	Nothing is assigned. In an attempt to write to this bit, write "0". The value, if read, turns out to be "0".		—	—
SMi5	Transfer direction lect bit	0 : LSB first 1 : MSB first	○	○
SMi6	Synchronous clock select bit (Note 2)	0 : External clock 1 : Internal clock	○	○
SMi7	Souri initial value set bit	Effective when SMi3 = 0 0 : L output 1 : H output	○	○

Note 1: Set "1" in bit 2 of the protection register (000A<sub>16</sub>) in advance to write to the S I/Oi control register (i = 3, 4).

Note 2: When using the port as an input/output port by setting the SI/Oi port select bit (i = 3, 4) to "1", be sure to set the sync clock select bit to "1".

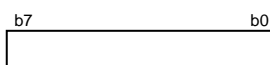
## SI/Oi bit rate generator



Symbol	Address	When reset
S3BRG	0363 <sub>16</sub>	Indeterminate
S4BRG	0367 <sub>16</sub>	Indeterminate

Indeterminate	Values that can be set	R	W
Assuming that set value = n, BRGi divides the count source by n + 1	00 <sub>16</sub> to FF <sub>16</sub>	○	○

## SI/Oi transmit/receive register



Symbol	Address	When reset
S3TRR	0360 <sub>16</sub>	Indeterminate
S4TRR	0364 <sub>16</sub>	Indeterminate

Indeterminate	R	W
Transmission/reception starts by writing data to this register. After transmission/reception finishes, reception data is input.	○	○

Figure 2.11.32 S I/O3, 4 related register

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Table 2.11.13 Specifications of S I/O3, 4**

Item	Specifications
Transfer data format	<ul style="list-style-type: none"> <li>Transfer data length: 8 bits</li> </ul>
Transfer clock	<ul style="list-style-type: none"> <li>With the internal clock selected (bit 6 of 0362<sub>16</sub>, 0366<sub>16</sub> = "1"): f<sub>1/2</sub>(n<sub>i</sub>+1), f<sub>8/2</sub>(n<sub>i</sub>+1), f<sub>32/2</sub>(n<sub>i</sub>+1) (Note 1)</li> <li>With the external clock selected (bit 6 of 0362<sub>16</sub>, 0366<sub>16</sub> = 0): Input from the CLK<sub>i</sub> terminal (Note 2)</li> </ul>
Conditions for transmission/reception start	<ul style="list-style-type: none"> <li>To start transmit/reception, the following requirements must be met: <ul style="list-style-type: none"> <li>Select the synchronous clock (use bit 6 of 0362<sub>16</sub>, 0366<sub>16</sub>).</li> <li>Select a frequency dividing ratio if the internal clock has been selected (use bits 0 and 1 of 0362<sub>16</sub>, 0366<sub>16</sub>).</li> <li>SOUT<sub>i</sub> initial value set bit (use bit 7 of 0362<sub>16</sub>, 0366<sub>16</sub>) = 1.</li> <li>S I/O<sub>i</sub> port select bit (bit 3 of 0362<sub>16</sub>, 0366<sub>16</sub>) = 1.</li> <li>Select the transfer direction (use bit 5 of 0362<sub>16</sub>, 0366<sub>16</sub>)</li> <li>Write transfer data to SI/O<sub>i</sub> transmit/receive register (0360<sub>16</sub>, 0364<sub>16</sub>)</li> </ul> </li> <li>To use S I/O<sub>i</sub> interrupts, the following requirements must be met: <ul style="list-style-type: none"> <li>Clear the SI/O<sub>i</sub> interrupt request bit before writing transfer data to the SI/O<sub>i</sub> transmit/receive register (bit 3 of 0049<sub>16</sub>, 0048<sub>16</sub>) = 0.</li> </ul> </li> </ul>
Interrupt request generation timing	<ul style="list-style-type: none"> <li>Rising edge of the last transfer clock. (Note 3)</li> </ul>
Select function	<ul style="list-style-type: none"> <li>LSB first or MSB first selection Whether transmission/reception begins with bit 0 (LSB) or bit 7 (MSB) can be selected.</li> <li>Function for setting an SOUT<sub>i</sub> initial value selection When using an external clock for the transfer clock, the user can choose the SOUT<sub>i</sub> pin output level during a non-transfer time. For details on how to set, see Figure 2.11.33.</li> </ul>
Precaution	<ul style="list-style-type: none"> <li>Unlike UART0–2, SI/O<sub>i</sub> (i = 3, 4) is not divided for transfer register and buffer. Therefore, do not write the next transfer data to the SI/O<sub>i</sub> transmit/receive register (addresses 0360<sub>16</sub>, 0364<sub>16</sub>) during a transfer. When the internal clock is selected for the transfer clock, SOUT<sub>i</sub> holds the last data for a 1/2 transfer clock period after it finished transferring and then goes to a high-impedance state. However, if the transfer data is written to the SI/O<sub>i</sub> transmit/receive register (addresses 0360<sub>16</sub>, 0364<sub>16</sub>) during this time, SOUT<sub>i</sub> is placed in the high-impedance state immediately upon writing and the data hold time is thereby reduced.</li> </ul>

Note 1: n is a value from 00<sub>16</sub> through FF<sub>16</sub> set in the S I/O<sub>i</sub> transfer rate register (i = 3, 4).

Note 2: With the external clock selected:

- Before data can be written to the SI/O<sub>i</sub> transmit/receive register (addresses 0360<sub>16</sub>, 0364<sub>16</sub>), the CLK<sub>i</sub> pin input must be in the low state. Also, before rewriting the SI/O<sub>i</sub> Control Register (addresses 0362<sub>16</sub>, 0366<sub>16</sub>)'s bit 7 (SOUT<sub>i</sub> initial value set bit), make sure the CLK<sub>i</sub> pin input is held low.
- The S I/O<sub>i</sub> circuit keeps on with the shift operation as long as the synchronous clock is entered in it, so stop the synchronous clock at the instant when it counts to eight. The internal clock, if selected, automatically stops.

Note 3: If the internal clock is used for the synchronous clock, the transfer clock signal stops at the "H" state.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (1) Functions for setting an Souti initial value

When using an external clock for the transfer clock, the SOUTi pin output level during a non-transfer time can be set to the high or the low state. Figure 2.11.33 shows the timing chart for setting an SOUTi initial value and how to set it.

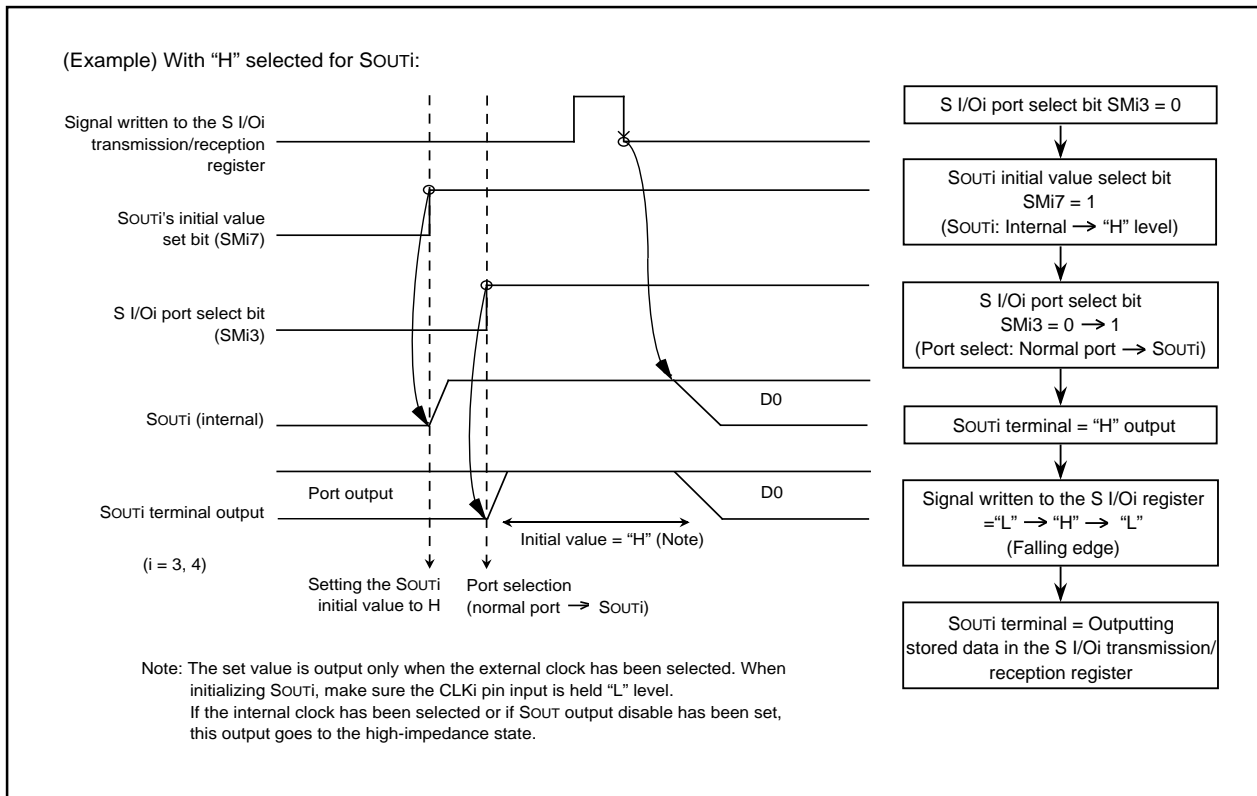


Figure 2.11.33 Timing chart for setting SOUTi's initial value and how to set it

## (2) S I/Oi operation timing

Figure 2.11.34 shows the S I/Oi operation timing

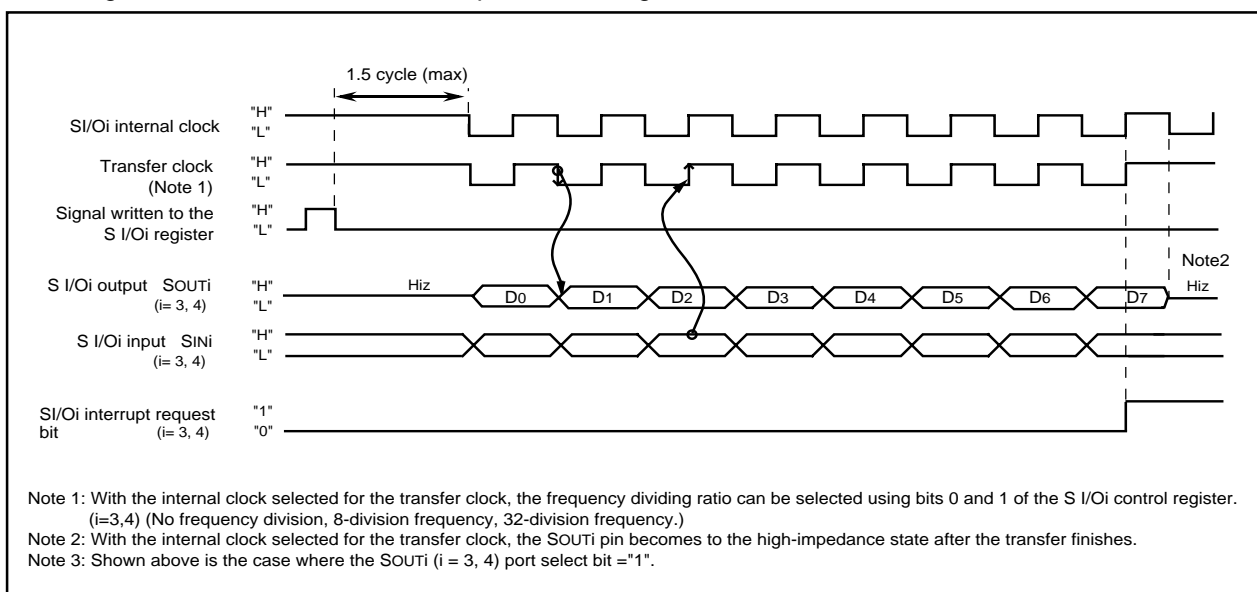


Figure 22.11.34 S I/Oi operation timing chart

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.12 A-D Converter

The A-D converter consists of one 8-bit successive approximation A-D converter circuit with a capacitive coupling amplifier. Pins P100 to P107, P95, and P96 also function as the analog signal input pins. The direction registers of these pins for A-D conversion must therefore be set to input. The Vref connect bit (bit 5 at address 03D716) can be used to isolate the resistance ladder of the A-D converter from the reference voltage input pin (VREF) when the A-D converter is not used. Doing so stops any current flowing into the resistance ladder from VREF, reducing the power dissipation. When using the A-D converter, start A-D conversion only after setting bit 5 of 03D716 to connect VREF.

The result of A-D conversion is stored in the A-D registers of the selected pins.

Table 2.12.1 shows the performance of the A-D converter. Figure 2.12.1 shows the block diagram of the A-D converter, and Figures 2.12.2 and 2.12.3 show the A-D converter-related registers.

**Table 2.12.1 Performance of A-D converter**

Item	Performance
Method of A-D conversion	Successive approximation (capacitive coupling amplifier)
Analog input voltage (Note 1)	0V to AVCC (VCC)
Operating clock $\phi_{AD}$ (Note 2)	$f_{AD}/\text{divide-by-2}$ of $f_{AD}/\text{divide-by-4}$ of $f_{AD}$ , $f_{AD}=f(XIN)$
Resolution	8-bit
Absolute precision	<ul style="list-style-type: none"> <li>• Without sample and hold function <math>\pm 3\text{LSB}</math></li> <li>• With sample and hold function <math>\pm 2\text{LSB}</math></li> </ul>
Operating modes	One-shot mode, repeat mode, single sweep mode, repeat sweep mode 0, and repeat sweep mode 1
Analog input pins	8pins (AN0 to AN7) + 2pins (ANEX0 and ANEX1)
A-D conversion start condition	<ul style="list-style-type: none"> <li>• Software trigger A-D conversion starts when the A-D conversion start flag changes to "1"</li> <li>• External trigger (can be retriggered) A-D conversion starts when the A-D conversion start flag is "1" and the ADTRG/P97 input changes from "H" to "L"</li> </ul>
Conversion speed per pin	<ul style="list-style-type: none"> <li>• Without sample and hold function 49 <math>\phi_{AD}</math> cycles</li> <li>• With sample and hold function 28 <math>\phi_{AD}</math> cycles</li> </ul>

Note 1: Does not depend on use of sample and hold function.

Note 2: Without sample and hold function, set the  $\phi_{AD}$  frequency to 250kHz min.

With the sample and hold function, set the  $\phi_{AD}$  frequency to 1MHz min.

The block diagram illustrates the internal architecture of the A-D converter. At the top, the A-D conversion rate selection is controlled by CKS0 and CKS1, which determine the clock source (fAD or φAD) and the conversion rate (1/2 or 1). The Resistor ladder is connected to VREF and AVSS, with VCUT=0 and VCUT=1 inputs. The Successive conversion register receives data from the Resistor ladder and outputs to the A-D control registers (A-D control register 1 at address 03D716 and A-D control register 0 at address 03D616). The A-D registers (A-D register 0(8) to A-D register 7(8)) are addressed by the Successive conversion register and output to the Data bus. The Decoder receives inputs from the A-D control registers and outputs to the A-D registers. The Comparator compares Vref and VIN. The input channels (AN0-AN7) are connected to the A-D registers via CH2, CH1, and CH0. The output channels (OPA1, OPA0) are controlled by OPA1 and OPA0 inputs, which determine the output mode (Normal operation, ANEX0, ANEX1, or External op-amp mode).



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

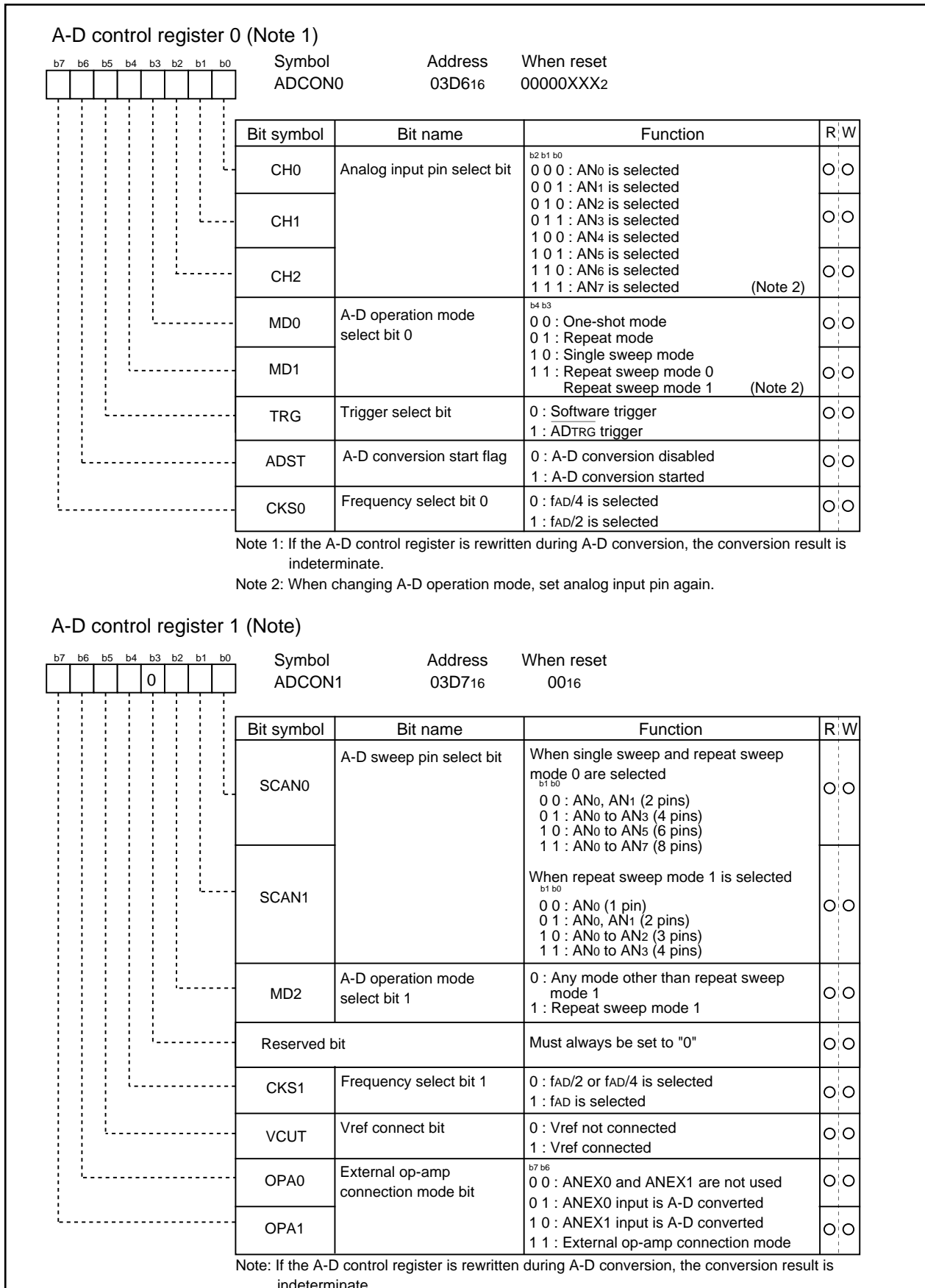


Figure 2.12.2 A-D converter-related registers (1)

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

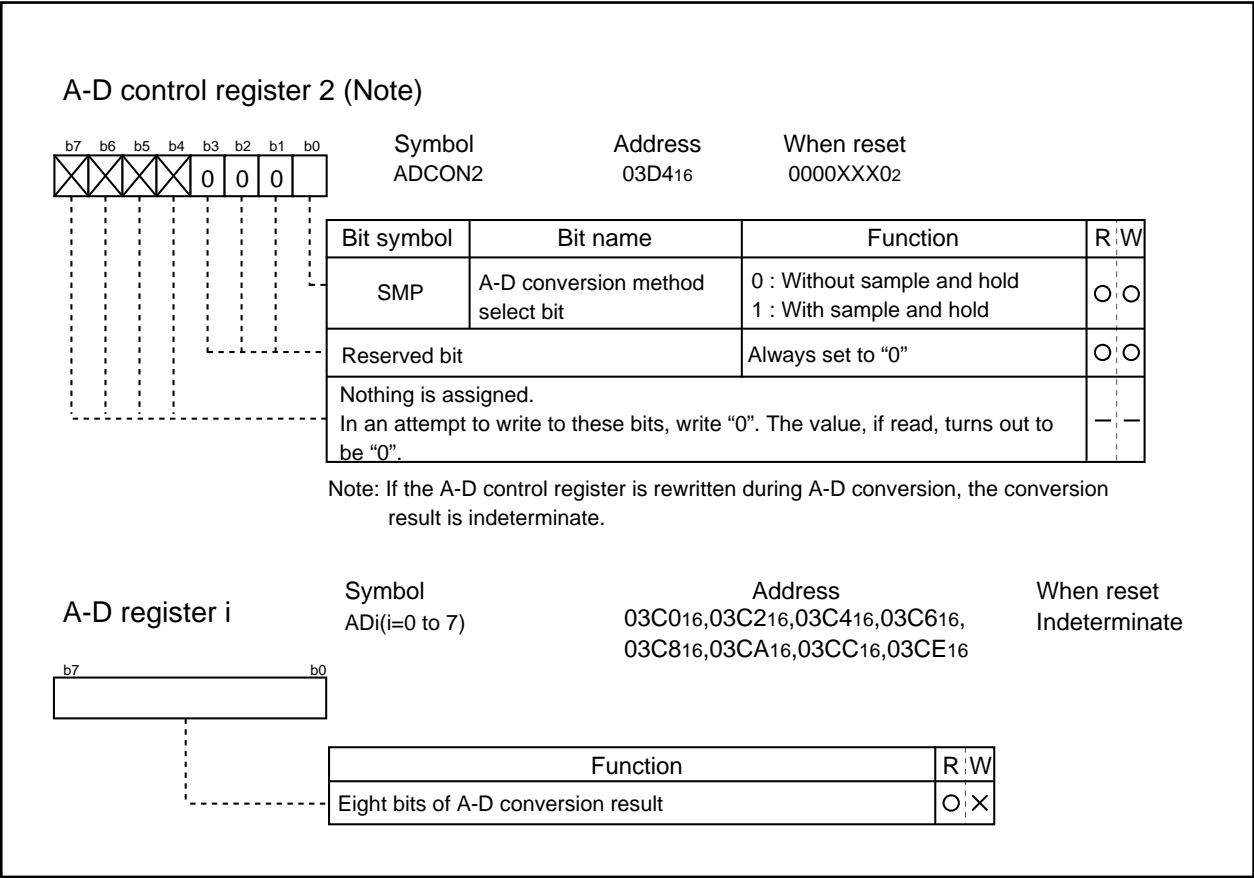


Figure 2.12.3 A-D converter-related registers (2)

# M306H1SFP

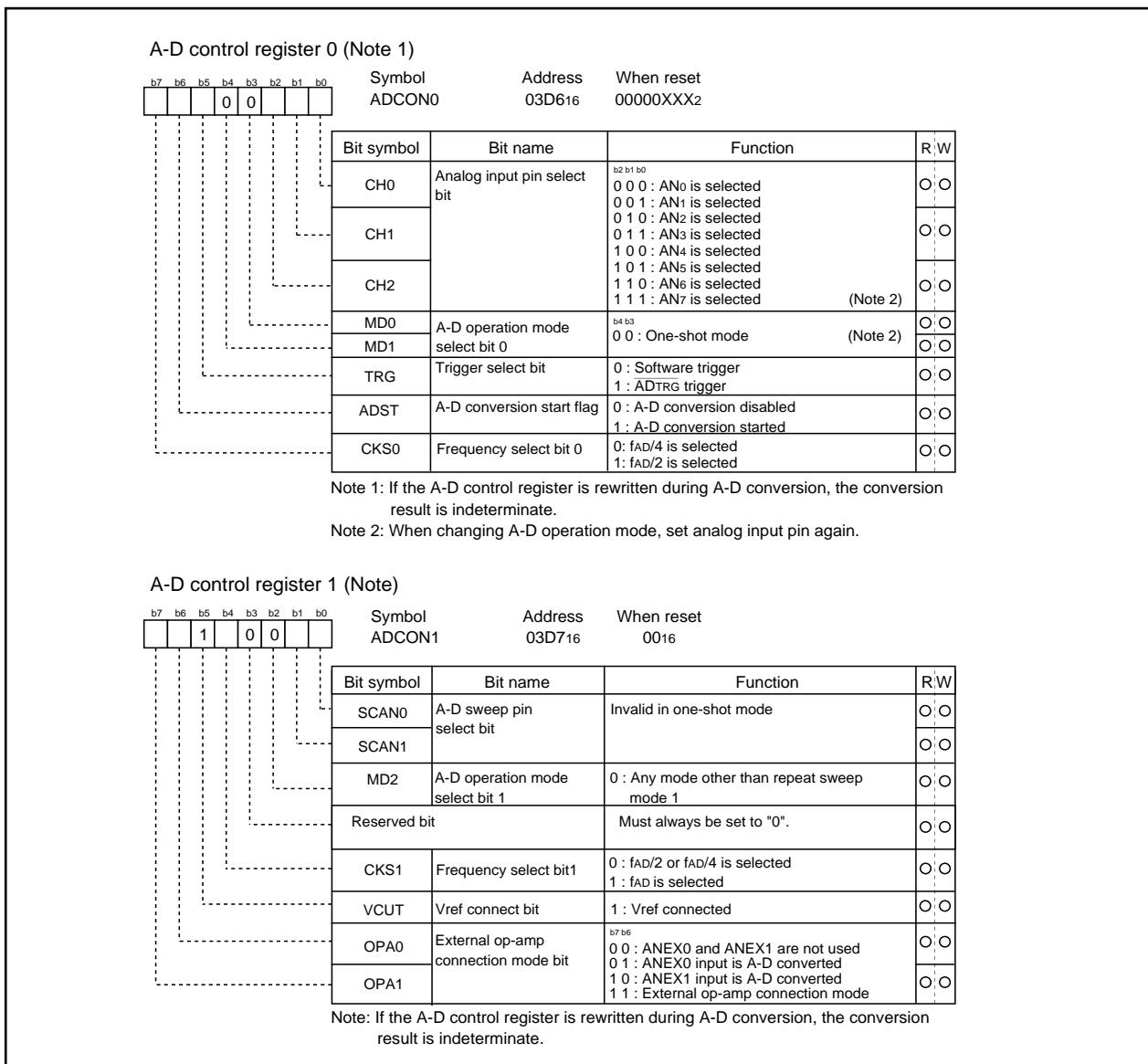
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (1) One-shot mode

In one-shot mode, the pin selected using the analog input pin select bit is used for one-shot A-D conversion. Table 2.12.2 shows the specifications of one-shot mode. Figure 2.12.4 shows the A-D control register in one-shot mode.

**Table 2.12.2 One-shot mode specifications**

Item	Specification
Function	The pin selected by the analog input pin select bit is used for one A-D conversion
Start condition	Writing "1" to A-D conversion start flag
Stop condition	<ul style="list-style-type: none"> <li>End of A-D conversion (A-D conversion start flag changes to "0", except when external trigger is selected)</li> <li>Writing "0" to A-D conversion start flag</li> </ul>
Interrupt request generation timing	End of A-D conversion
Input pin	One of AN <sub>0</sub> to AN <sub>7</sub> , as selected
Reading of result of A-D converter	Read A-D register corresponding to selected pin



**Figure 2.12.4 A-D conversion register in one-shot mode**



# M306H1SFP

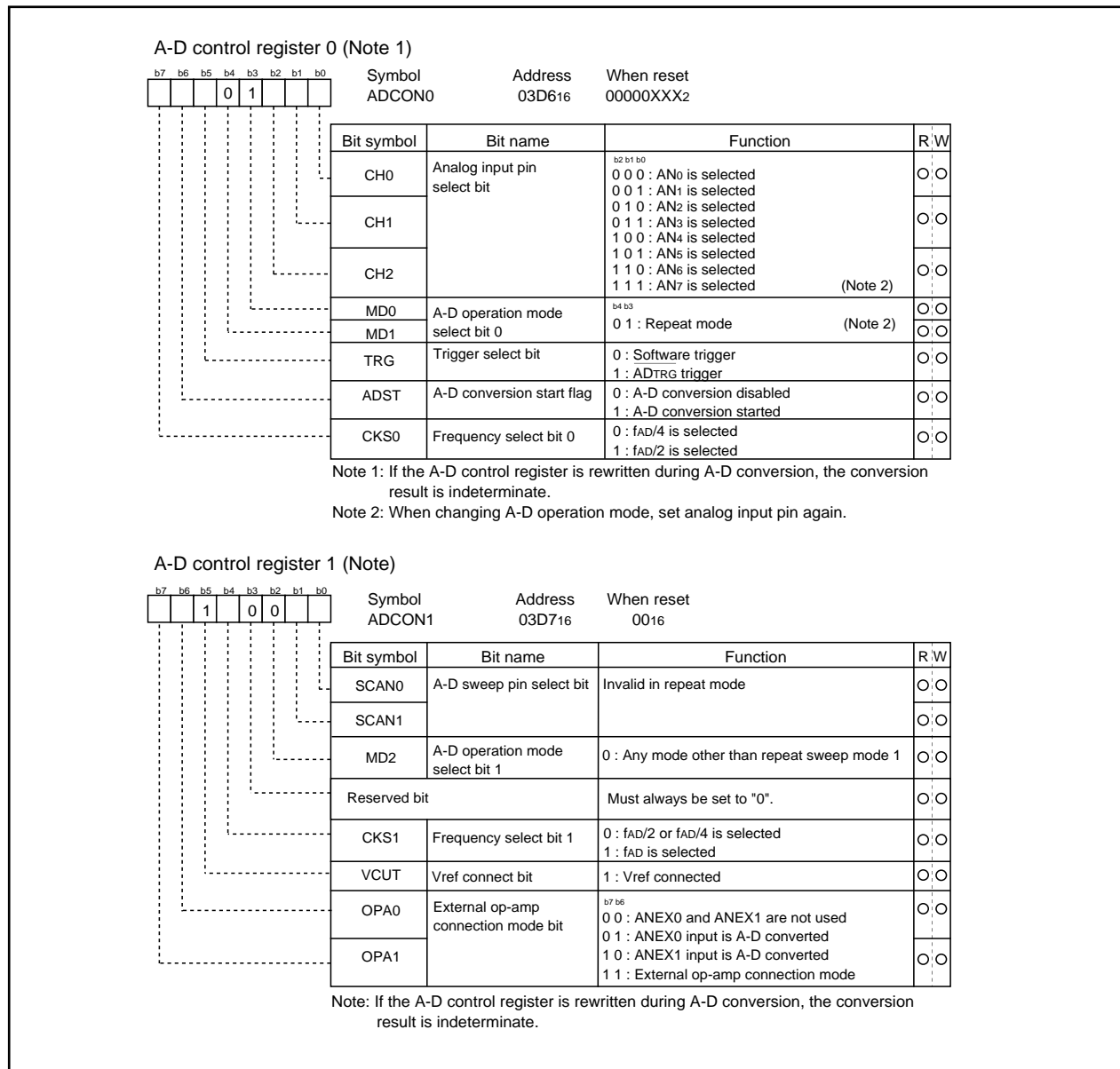
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (2) Repeat mode

In repeat mode, the pin selected using the analog input pin select bit is used for repeated A-D conversion. Table 2.12.3 shows the specifications of repeat mode. Figure 2.12.5 shows the A-D control register in repeat mode.

**Table 2.12.3 Repeat mode specifications**

Item	Specification
Function	The pin selected by the analog input pin select bit is used for repeated A-D conversion
Star condition	Writing "1" to A-D conversion start flag
Stop condition	Writing "0" to A-D conversion start flag
Interrupt request generation timing	None generated
Input pin	One of AN0 to AN7, as selected
Reading of result of A-D converter	Read A-D register corresponding to selected pin



**Figure 2.12.5 A-D conversion register in repeat mode**

# M306H1SFP

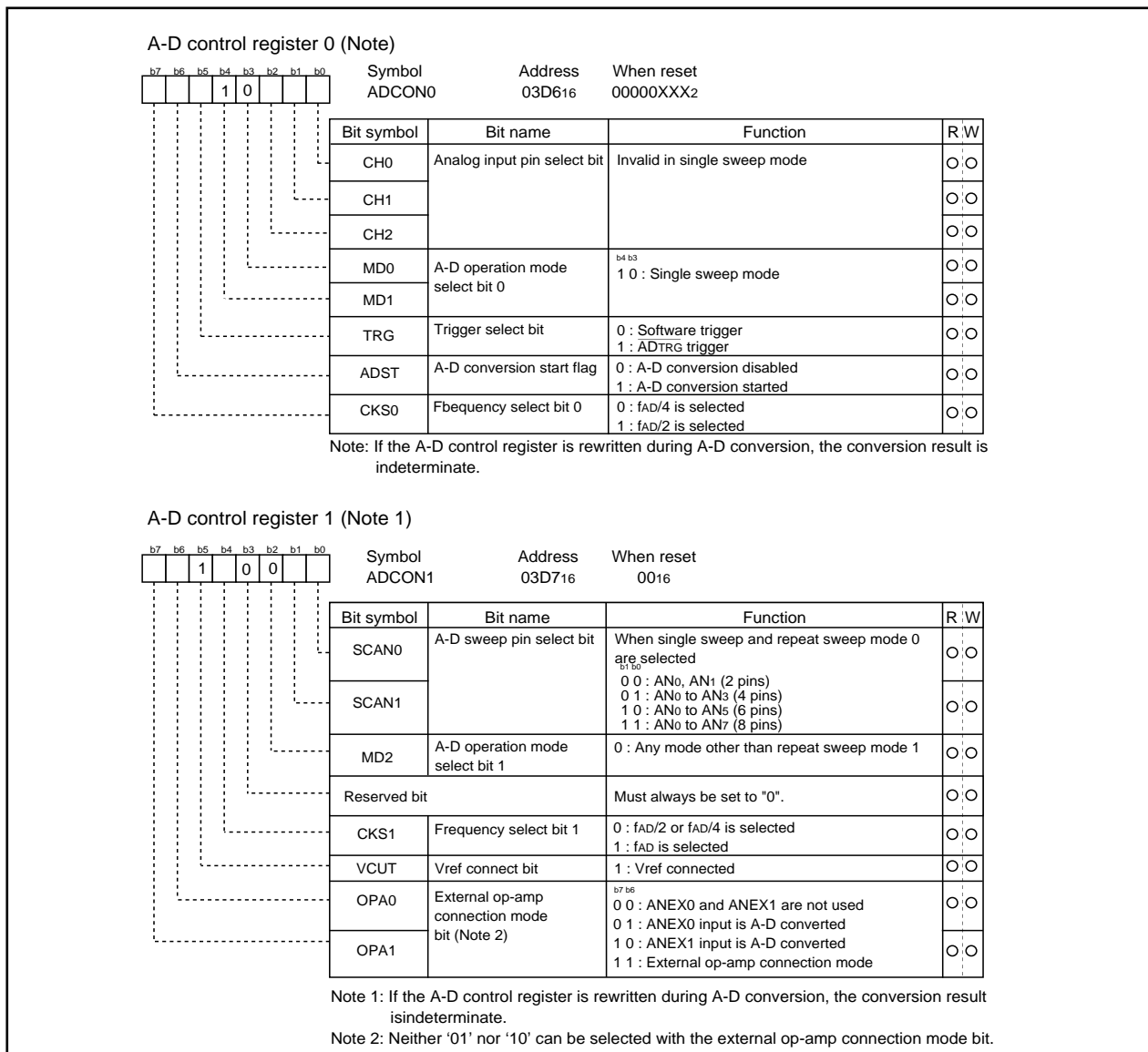
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

### (3) Single sweep mode

In single sweep mode, the pins selected using the A-D sweep pin select bit are used for one-by-one A-D conversion. Table 2.12.4 shows the specifications of single sweep mode. Figure 2.12.6 shows the A-D control register in single sweep mode.

**Table 2.12.4 Single sweep mode specifications**

Item	Specification
Function	The pins selected by the A-D sweep pin select bit are used for one-by-one A-D conversion
Start condition	Writing "1" to A-D converter start flag
Stop condition	<ul style="list-style-type: none"> <li>End of A-D conversion (A-D conversion start flag changes to "0", except when external trigger is selected)</li> <li>Writing "0" to A-D conversion start flag</li> </ul>
Interrupt request generation timing	End of A-D conversion
Input pin	AN0 and AN1 (2 pins), AN0 to AN3 (4 pins), AN0 to AN5 (6 pins), or AN0 to AN7 (8 pins)
Reading of result of A-D converter	Read A-D register corresponding to selected pin



**Figure 2.12.6 A-D conversion register in single sweep mode**

# M306H1SFP

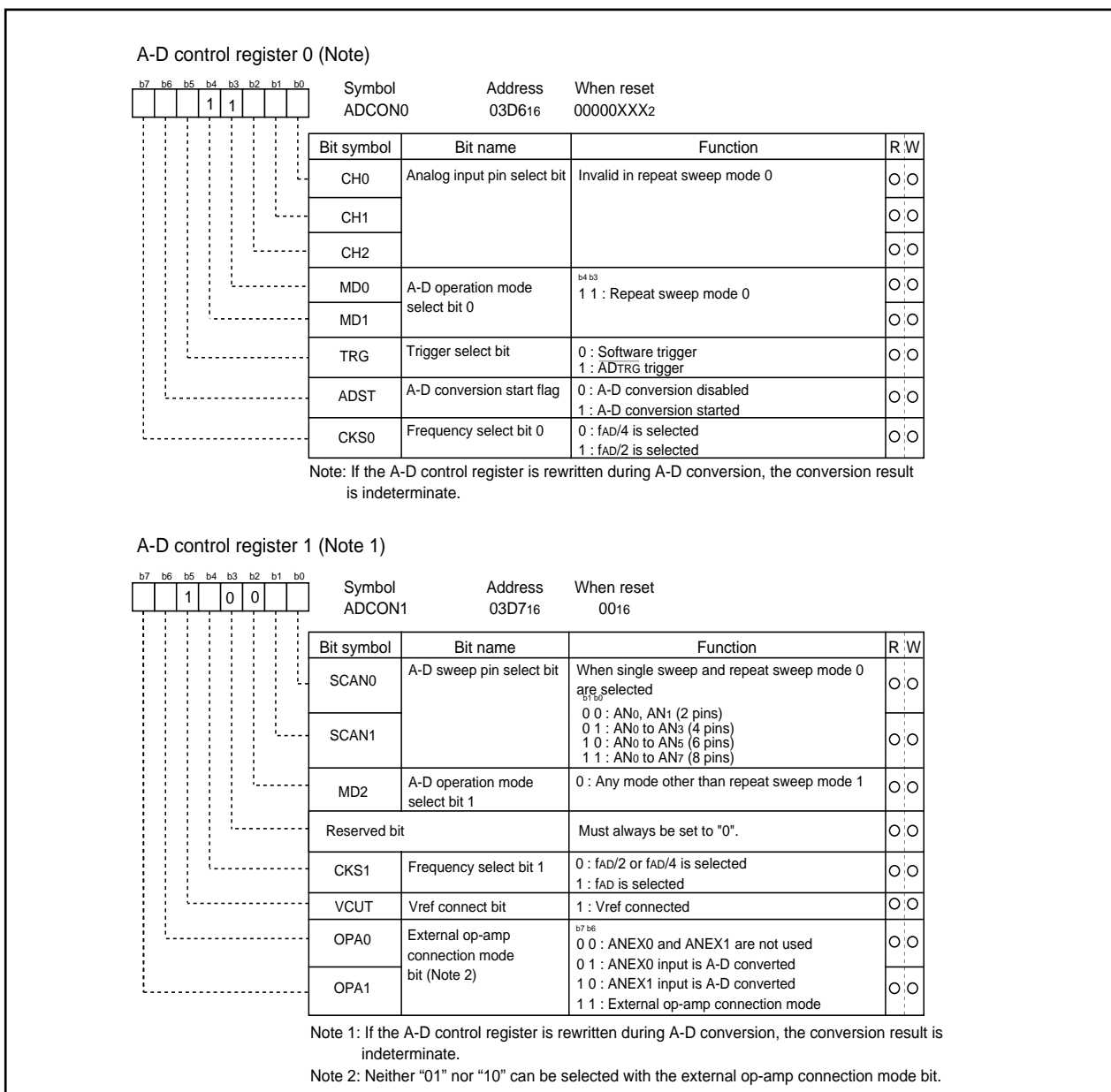
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (4) Repeat sweep mode 0

In repeat sweep mode 0, the pins selected using the A-D sweep pin select bit are used for repeat sweep A-D conversion. Table 2.12.5 shows the specifications of repeat sweep mode 0. Figure 2.12.7 shows the A-D control register in repeat sweep mode 0.

**Table 2.12.5 Repeat sweep mode 0 specifications**

Item	Specification
Function	The pins selected by the A-D sweep pin select bit are used for repeat sweep A-D conversion
Start condition	Writing "1" to A-D conversion start flag
Stop condition	Writing "0" to A-D conversion start flag
Interrupt request generation timing	None generated
Input pin	AN <sub>0</sub> and AN <sub>1</sub> (2 pins), AN <sub>0</sub> to AN <sub>3</sub> (4 pins), AN <sub>0</sub> to AN <sub>5</sub> (6 pins), or AN <sub>0</sub> to AN <sub>7</sub> (8 pins)
Reading of result of A-D converter	Read A-D register corresponding to selected pin (at any time)



**Figure 2.12.7 A-D conversion register in repeat sweep mode 0**

## M306H1SFP

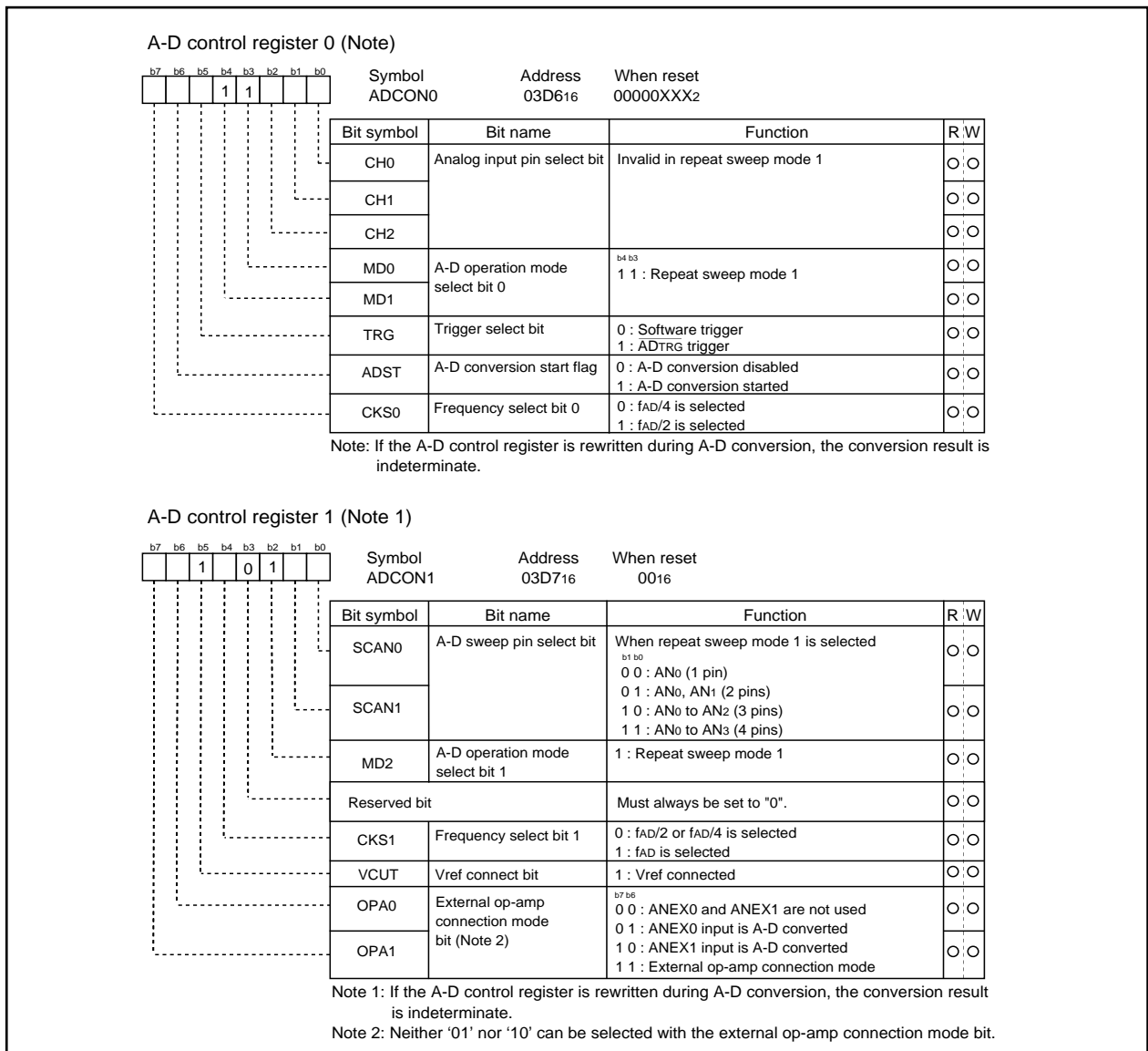
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**(5) Repeat sweep mode 1**

In repeat sweep mode 1, all pins are used for A-D conversion with emphasis on the pin or pins selected using the A-D sweep pin select bit. Table 2.12.6 shows the specifications of repeat sweep mode 1. Figure 2.12.8 shows the A-D control register in repeat sweep mode 1.

**Table 2.12.6 Repeat sweep mode 1 specifications**

Item	Specification
Function	All pins perform repeat sweep A-D conversion, with emphasis on the pin or pins selected by the A-D sweep pin select bit Example : AN <sub>0</sub> selected AN <sub>0</sub> → AN <sub>1</sub> → AN <sub>0</sub> → AN <sub>2</sub> → AN <sub>0</sub> → AN <sub>3</sub> , etc
Start condition	Writing "1" to A-D conversion start flag
Stop condition	Writing "0" to A-D conversion start flag
Interrupt request generation timing	None generated
Input pin	AN <sub>0</sub> (1 pin), AN <sub>0</sub> and AN <sub>1</sub> (2 pins), AN <sub>0</sub> to AN <sub>2</sub> (3 pins), AN <sub>0</sub> to AN <sub>3</sub> (4 pins)
Reading of result of A-D converter	Read A-D register corresponding to selected pin (at any time)

**Figure 2.12.8 A-D conversion register in repeat sweep mode 1**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (a) Sample and hold

Sample and hold is selected by setting bit 0 of the A-D control register 2 (address 03D4<sub>16</sub>) to "1". When sample and hold is selected, the rate of conversion of each pin increases. As a result, a 28 fAD cycle is achieved. Sample and hold can be selected in all modes. However, in all modes, be sure to specify before starting A-D conversion whether sample and hold is to be used.

## (b) Extended analog input pins

In one-shot mode and repeat mode, the input via the extended analog input pins ANEX0 and ANEX1 can also be converted from analog to digital.

When bit 6 of the A-D control register 1 (address 03D7<sub>16</sub>) is "1" and bit 7 is "0", input via ANEX0 is converted from analog to digital. The result of conversion is stored in A-D register 0.

When bit 6 of the A-D control register 1 (address 03D7<sub>16</sub>) is "0" and bit 7 is "1", input via ANEX1 is converted from analog to digital. The result of conversion is stored in A-D register 1.

## (c) External operation amp connection mode

In this mode, multiple external analog inputs via the extended analog input pins, ANEX0 and ANEX1, can be amplified together by just one operation amp and used as the input for A-D conversion.

When bit 6 of the A-D control register 1 (address 03D7<sub>16</sub>) is "1" and bit 7 is "1", input via AN<sub>0</sub> to AN<sub>7</sub> is output from ANEX0. The input from ANEX1 is converted from analog to digital and the result stored in the corresponding A-D register. The speed of A-D conversion depends on the response of the external operation amp. Do not connect the ANEX0 and ANEX1 pins directly. Figure 2.12.9 is an example of how to connect the pins in external operation amp mode.

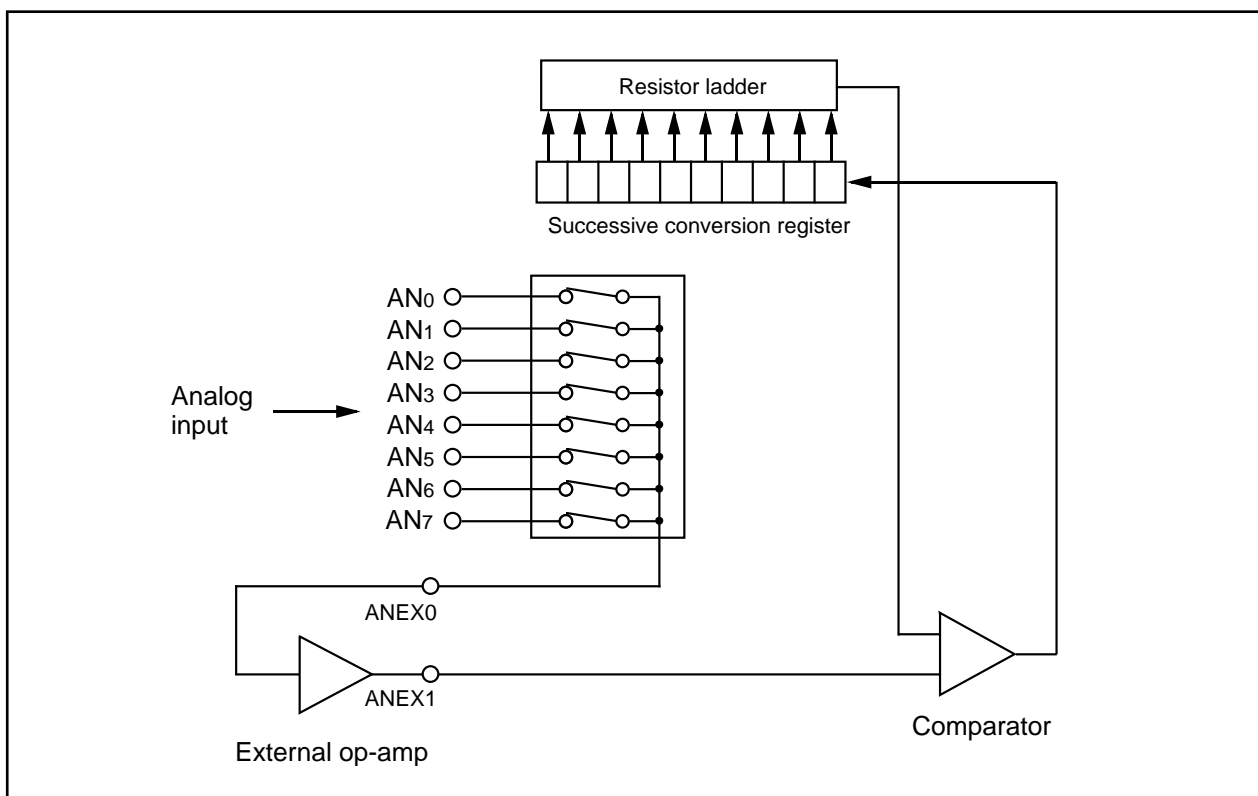


Figure 2.12.9 Example of external op-amp connection mode

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.13 D-A Converter

This is an 8-bit, R-2R type D-A converter. The microcomputer contains two independent D-A converters of this type.

D-A conversion is performed when a value is written to the corresponding D-A register. Bits 0 and 1 (D-A output enable bits) of the D-A control register decide if the result of conversion is to be output. Do not set the target port to output mode if D-A conversion is to be performed.

Output analog voltage (V) is determined by a set value (n : decimal) in the D-A register.

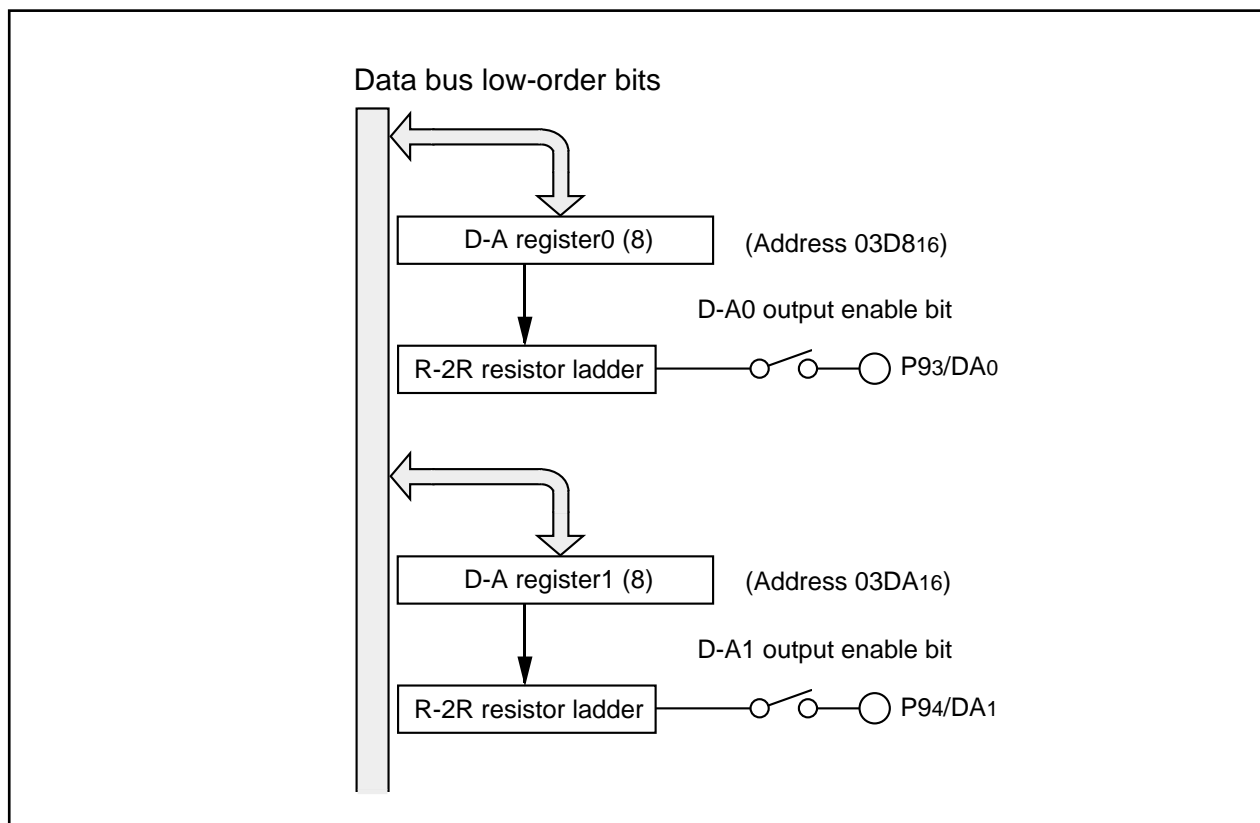
$$V = V_{REF} \times n / 256 \quad (n = 0 \text{ to } 255)$$

$V_{REF}$  : reference voltage

Table 2.13.1 lists the performance of the D-A converter. Figure 2.13.1 shows the block diagram of the D-A converter. Figure 2.13.2 shows the D-A control register. Figure 2.13.3 shows the D-A converter equivalent circuit.

**Table 2.13.1 Performance of D-A converter**

Item	Performance
Conversion method	R-2R method
Resolution	8 bits
Analog output pin	2 channels



**Figure 2.13.1 Block diagram of D-A converter**

**D-A control register**

Symbol: DACon      Address: 03DC<sub>16</sub>      When reset: 00<sub>16</sub>

Bit symbol	Bit name	Function	R	W
DA0E	D-A0 output enable bit	0 : Output disabled 1 : Output enabled	○	○
DA1E	D-A1 output enable bit	0 : Output disabled 1 : Output enabled	○	○
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0"			—	—

**D-A register**

Symbol: DAi (i = 0, 1)      Address: 03D8<sub>16</sub>, 03DA<sub>16</sub>      When reset: Indeterminate

Function	R	W
Output value of D-A conversion	○	○

The diagram illustrates a 10-bit digital-to-analog converter (D-A) circuit. At the top, a "D-A output enable bit" is shown with a switch labeled "0" and "1". The DA0 input is connected to a resistor network. The network consists of a series of resistors: a 2R resistor followed by a series of R resistors, and finally a 2R resistor at the end. The D-A register0 is a 10-bit register that controls the output. The MSB (Most Significant Bit) is on the left, and the LSB (Least Significant Bit) is on the right. The output lines are AVss and VREF.

Note 1: The above diagram shows an instance in which the D-A register is assigned 2A16.

Note 2: The same circuit as this is also used for D-A1.

Note 3: To reduce the current consumption when the D-A converter is not used, set the D-A output enable bit to 0 and set the D-A register to 0016 so that no current flows in the resistors Rs and 2Rs.



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.14 CRC Calculation Circuit

The Cyclic Redundancy Check (CRC) calculation circuit detects an error in data blocks. The microcomputer uses a generator polynomial of CRC\_CCITT ( $X^{16} + X^{12} + X^5 + 1$ ) to generate CRC code. The CRC code is a 16-bit code generated for a block of a given data length in multiples of 8 bits. The CRC code is set in a CRC data register each time one byte of data is transferred to a CRC input register after writing an initial value into the CRC data register. Generation of CRC code for one byte of data is completed in two machine cycles.

Figure 2.14.1 shows the block diagram of the CRC circuit. Figure 2.14.2 shows the CRC-related registers. Figure 2.14.3 shows the calculation example using the CRC calculation circuit

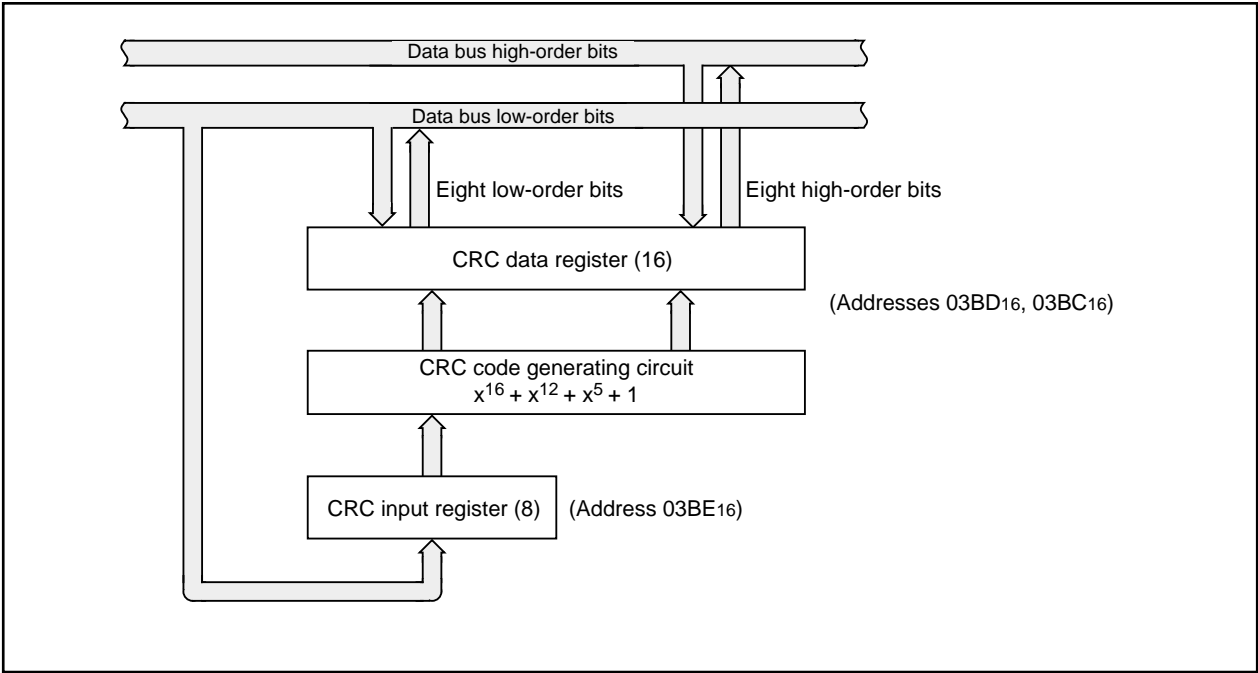


Figure 2.14.1 Block diagram of CRC circuit

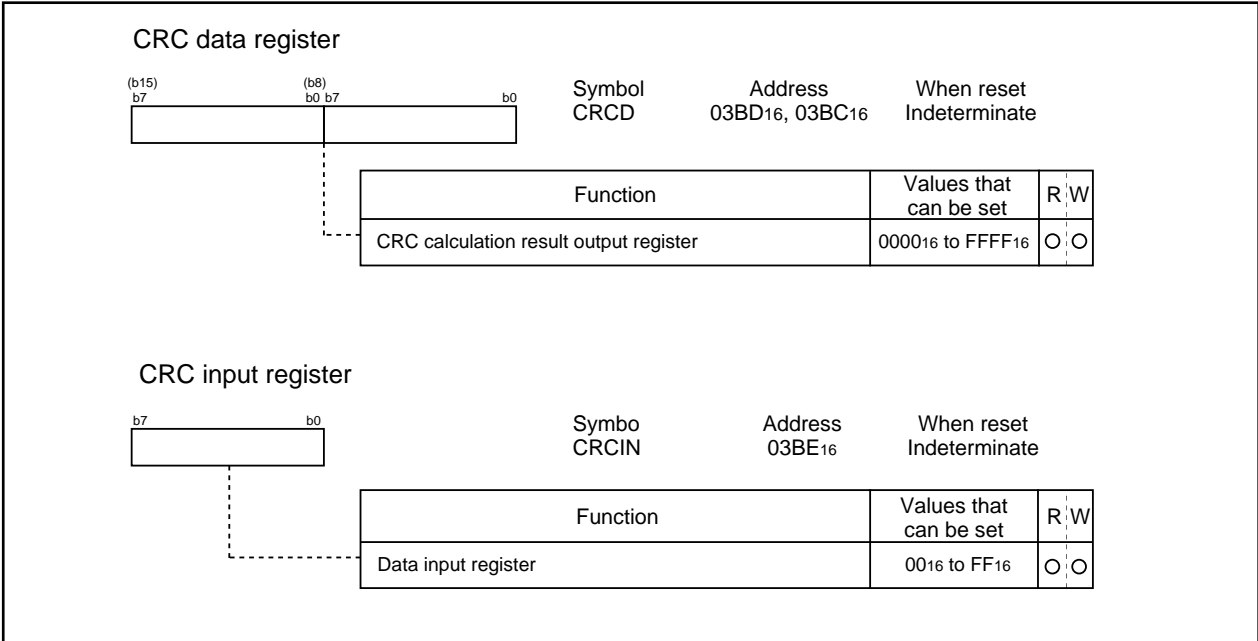


Figure 2.14.2 CRC-related registers



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

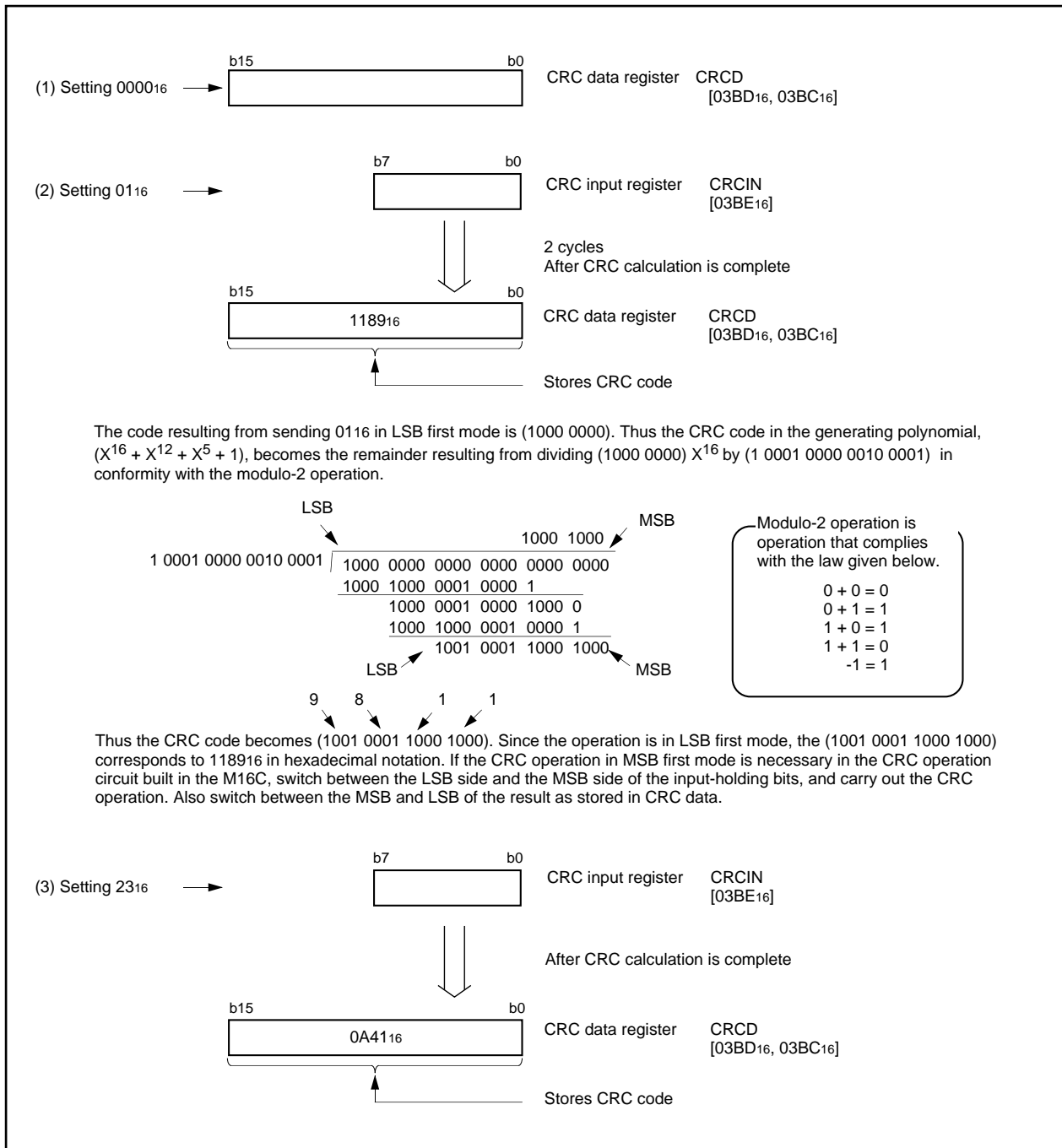


Figure 2.14.3 Calculation example using the CRC calculation circuit

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.15 Expansion Function

### 2.15.1 Expansion function description

Expansion function consists of OSD display function, data slicer function, data encoder function and humming decoder function. Each function is controlled by expansion memories.

#### (1) OSD function

Character is consisted of 12 X 10 dots, can display 40 (horizontal) X 25 (vertical) on the fixed line. And also, can be written over with built-in composite RAM.

M306H0SFP can be reduced external circuit by built-in SYNC-SEP (synchronous separate) and synchronous correction circuit. And it also can reduce error of character display at superimpose.

**Table 2.15.1 OSD function outline**

Screen composition	40 characters X 25 lines Fixed line display (at scrolling 40 characters X 24 lines)
Number of characters displayed	1000 (Max.)
Character composition	12 X 10 dot matrix (horizontal direction : 12 dots, vertical direction : 10 dots)
Characters available	Font RAM : 256 characters Composite RAM(SYRAM) : 15 characters
Character sizes available	Horizontal : one time, two times Vertical : one time, two times setting by every line
Display locations available	Horizontal direction : 486 locations Vertical direction : 235 locations
Blinking	Character units Cycle : approximately 1 second, or approximately 0.5 seconds (per screen) Duty 25%, 50% or 75% (per screen)
Coloring	Character coloring : 8 colors choices per character Character Background coloring : 8 colors choices per character Background coloring : 8 colors choices per screen
Blanking	Character blanking Matrix-outline Halftone blanking Can be set by every line
Superimpose	Can be displayed (PAL/SECAM)(monotone display)
Synchronous signal	Composite synchronous signal generate (only PAL) Composite video signal generate (only PAL)
Scrolling	The top and bottom smooth scroll of the soft control
General-purpose output ports	Combined port output : 9 (switching to R,G,B,GRAY,BLNK,CSYN,SLICEON, EDO1, EDO2 output)
Synchronous correction circuit	Built-in
Synchronous separation circuit	Built-in

#### (2) Data slicer function

Corresponds to TELETEXT, VPS, and VBI data

#### (3) Data encoder function

Encode VBI data

#### (4) Humming decoder function

8/4 humming and 24/18 humming

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

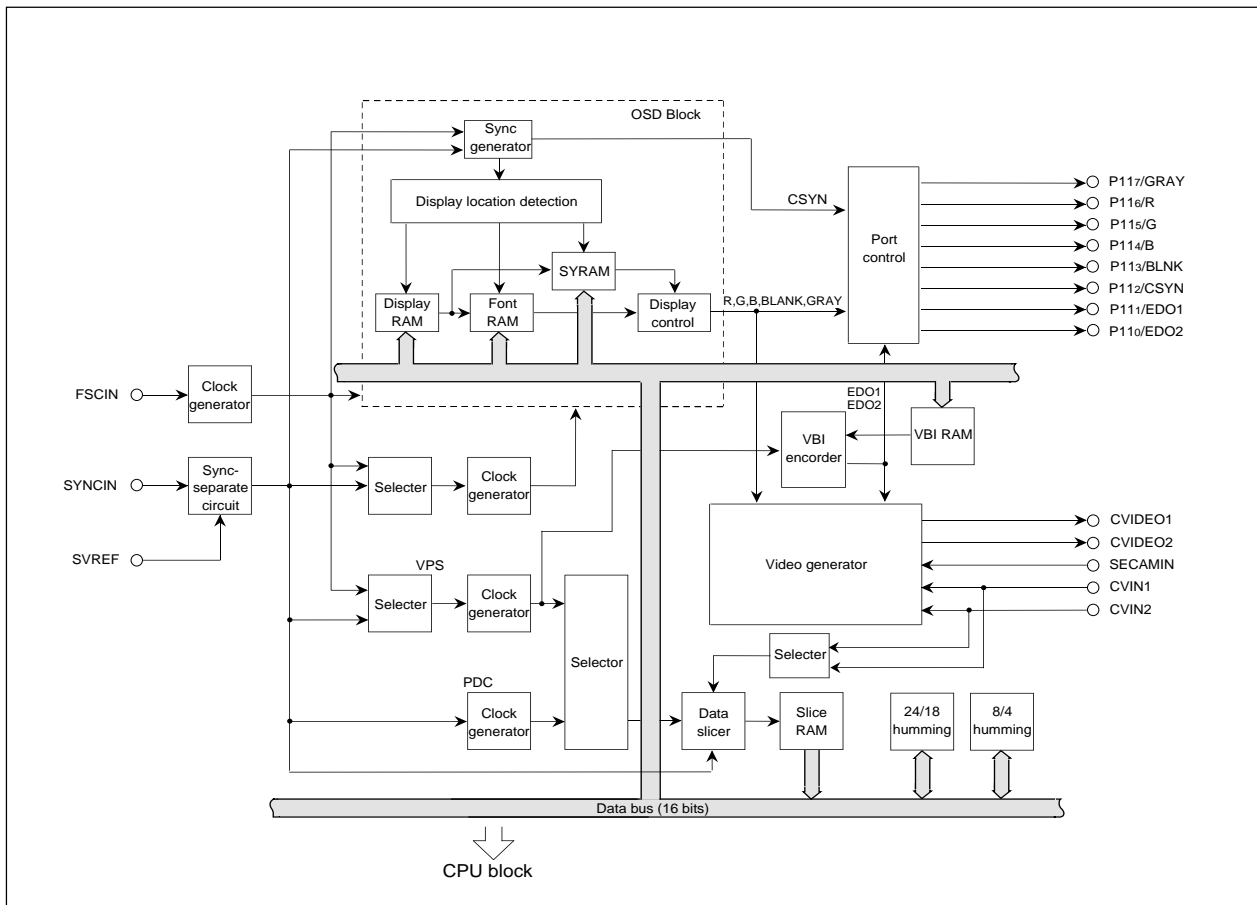


Figure 2.15.1 Block diagram of expansion function

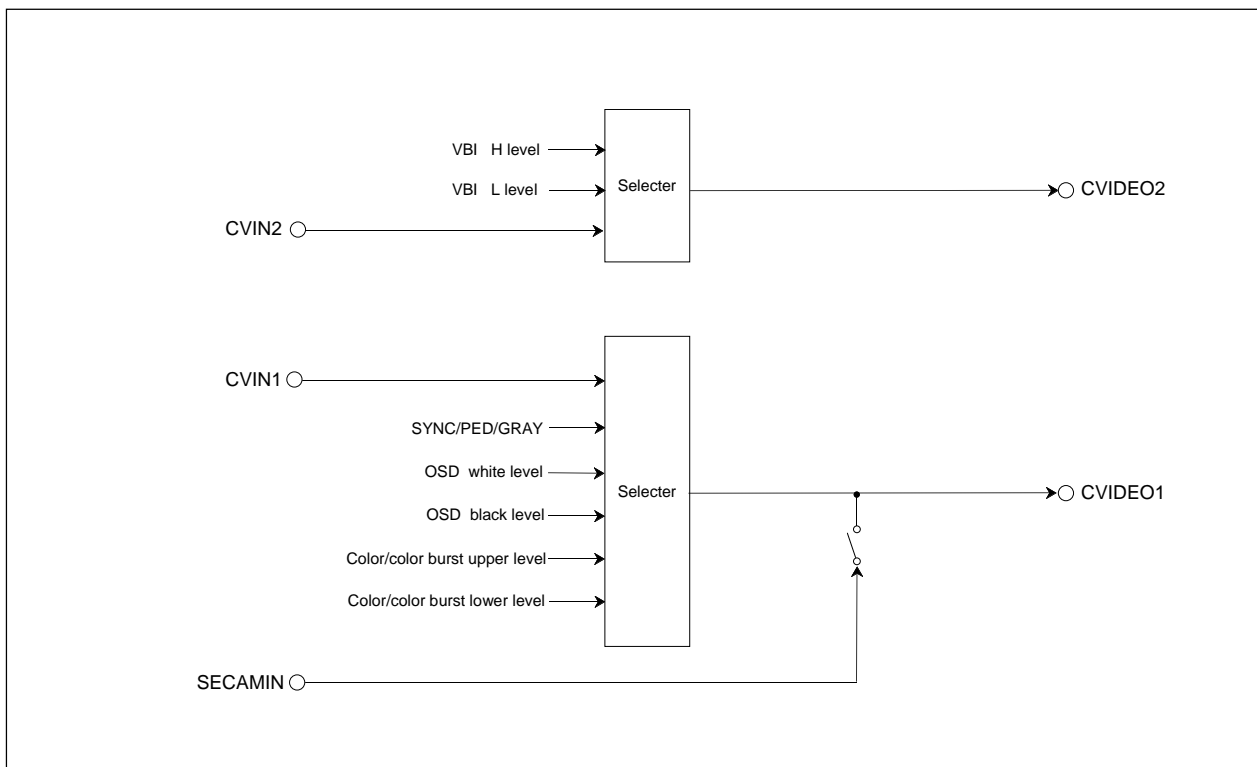


Figure 2.15.2 Block diagram of video generator

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.15.2 Expansion memory

Expansion function memory is divided by 6 patterns ; display RAM, Font RAM, SYRAM, Slice RAM, VBIRAM and expansion register. (Humming decoder operates by the register placed on SFR). Data writing and read out to these RAM and the expansion register are carried out 16 bit unit by the data setting register (addresses 0202<sub>16</sub> to 0218<sub>16</sub>) placed on SFR.

Contents of each memory and data setting register are shown in Table 2.15.2.

**Table 2.15.2 Expansion memory composition**

Expansion memory	Contents	Data setting register
Display RAM	1 screen (40 characters X 25 lines) display character setting. RAM font (character code), character color, character background color, blinking, SYRAM font (character code) and SYRAM character color are specified by 1 character unit.	Display RAM address control register (0202 <sub>16</sub> ) Display RAM data control register (0204 <sub>16</sub> )
Font RAM	255 character fonts setting.	Font RAM address control register (0206 <sub>16</sub> ) Font RAM data control register (0208 <sub>16</sub> )
SYRAM	15 composite character fonts setting.	SYRAM address control register (020A <sub>16</sub> ) SYRAM data control register (020C <sub>16</sub> )
Slice RAM	Store slice data.	Slice RAM address control register (020E <sub>16</sub> ) Slice RAM data control register (0210 <sub>16</sub> )
VBIRAM	VBI encode data setting.	VBIRAM address control register (0212 <sub>16</sub> ) VBIRAM data control register (0214 <sub>16</sub> )
Expansion register	This register controls OSD display, data slicer and VBI encoder.	OSD register address control register (0216 <sub>16</sub> ) OSD register data control register (0218 <sub>16</sub> )

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.15.3 Display RAM

Set 1 screen (40 characters X 25 lines) display character.

1 character display character setting is consists is 2 addresses (even address 16 bits + odd address 8 bits), set characters available, character color, blinking, character background color, SYRAM available and SYRAM color. Display RAM composition is shown in Table 2.15.3.

**Table 2.15.3 Display RAM composition**

Address (CA10 to CA0)	CD15	CD14	CD13	CD12	CD11	CD10	CD9	CD8	CD7	CD6	CD5	CD4	CD3	CD2	CD1	CD0	Remarks
000 <sub>16</sub>	0	BB	BG	BR	BLINK	CB	CG	CR	C7	C6	C5	C4	C3	C2	C1	C0	Character setting of the 0th
		Character background			Blinking	Character color			Font RAM character code								character of the 0th line.
001 <sub>16</sub>	0	0	0	0	0	0	0	0	SB	SG	SR	0	SYC3	SYC2	SYC1	SYC0	
									SYRAM character code(Note)				SYRAM character code(Note)				
002 <sub>16</sub>	0	BB	BG	BR	BLINK	CB	CG	CR	C7	C6	C5	C4	C3	C2	C1	C0	Character setting of the first
003 <sub>16</sub>	0	0	0	0	0	0	0	0	SB	SG	SR	0	SYC3	SYC2	SYC1	SYC0	character of the 0th line.
004 <sub>16</sub>	0	BB	BG	BR	BLINK	CB	CG	CR	C7	C6	C5	C4	C3	C2	C1	C0	Character setting of the second
005 <sub>16</sub>	0	0	0	0	0	0	0	0	SB	SG	SR	0	SYC3	SYC2	SYC1	SYC0	character of the 0th line.
006 <sub>16</sub>	:																Character setting of the third
:																	character of the 0th line.
7CB <sub>16</sub>																	Character setting of the 37th
																	character of the 24th line.
7CC <sub>16</sub>	0	BB	BG	BR	BLINK	CB	CG	CR	C7	C6	C5	C4	C3	C2	C1	C0	Character setting of the 38th
7CD <sub>16</sub>	0	0	0	0	0	0	0	0	SB	SG	SR	0	SYC3	SYC2	SYC1	SYC0	character of the 24th line.
7CE <sub>16</sub>	0	BB	BG	BR	BLINK	CB	CG	CR	C7	C6	C5	C4	C3	C2	C1	C0	Character setting of the 39th
7CF <sub>16</sub>	0	0	0	0	0	0	0	0	SB	SG	SR	0	SYC3	SYC2	SYC1	SYC0	Character of the 24th line.

Note: SYRAM setting bit is G1character setting bit when set 001<sub>16</sub> to font RAM character code.

(Refer to Teletext G1 character display for detail)

Set accessing address (CA10 to CA0) (shown in Table 2.15.3) to display RAM address control register (address 0202<sub>16</sub>), and write data (CD15 to CD0) from display RAM data control register (address 0204<sub>16</sub>). After data accessing fixed, display RAM address control register increments address automatically. Then, writing next address data is possible.

Display RAM bit composition is shown in Figure 2.15.3, Display RAM access registers are shown in Figure 2.15.4, Display RAM data access block diagram is shown in Figure 2.15.5, and Address map is shown in Figure 2.15.6 and Figure 2.15.7.

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

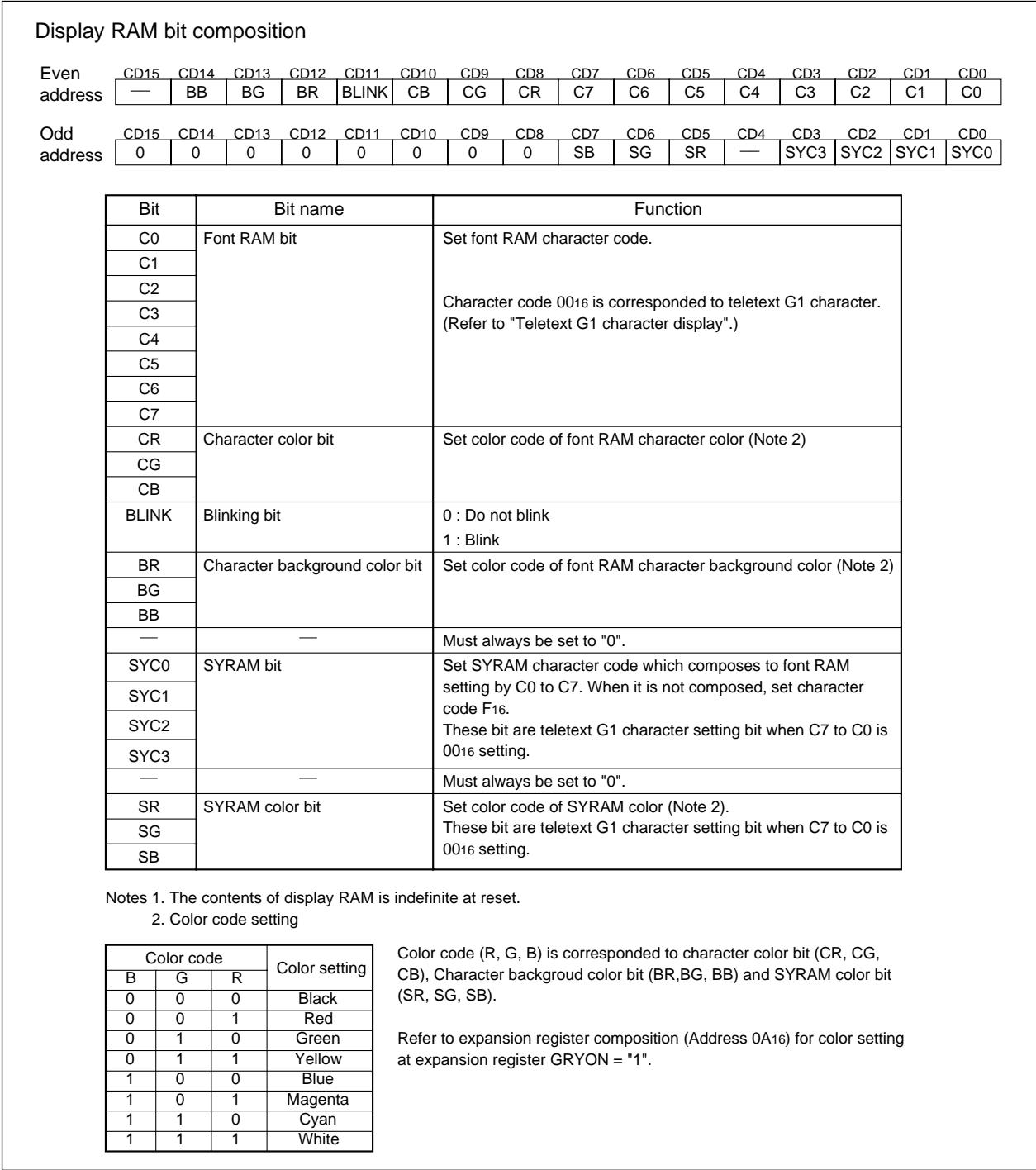


Figure 2.15.3 Display RAM bit composition

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

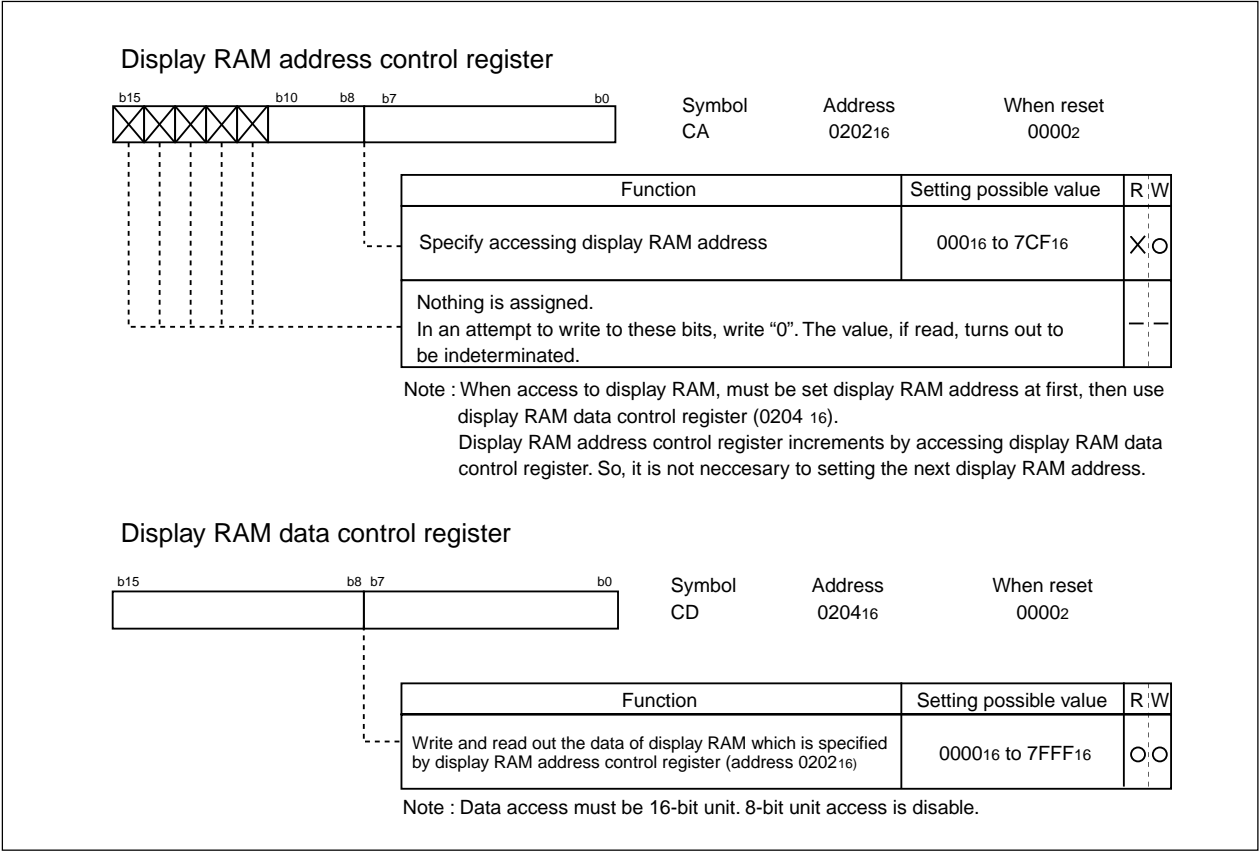


Figure 2.15.4 Display RAM access registers

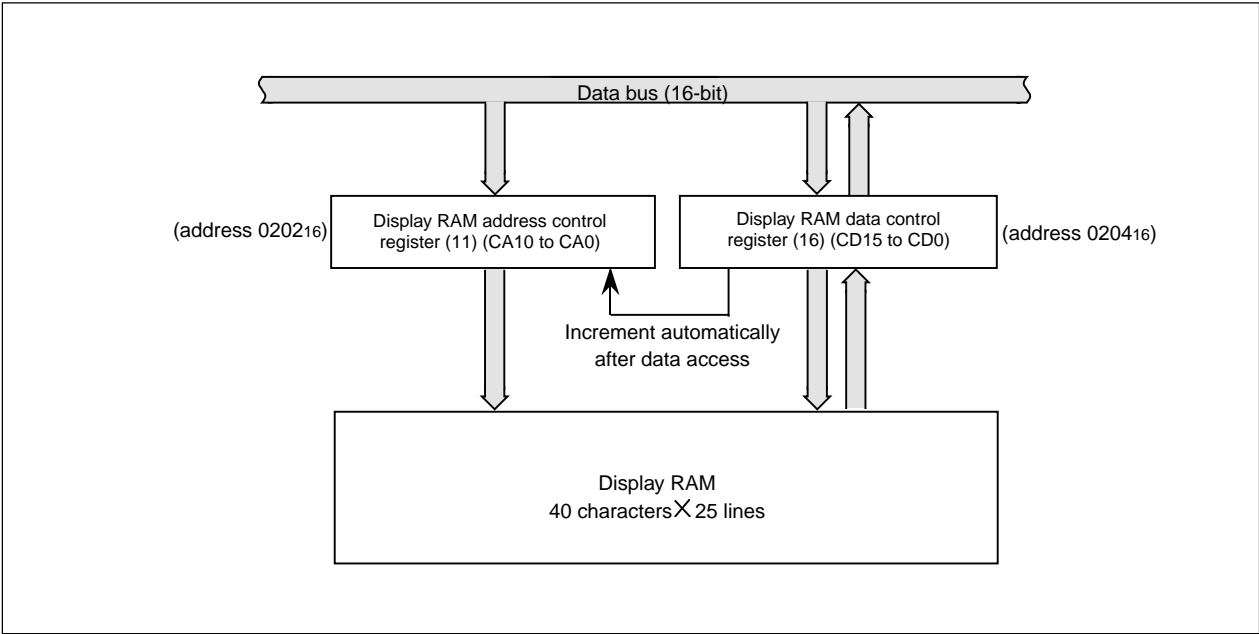


Figure 2.15.5 Display RAM access block diagram

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Character 0		Character 39	
Line 0	000 002 004 006 008 00A 00C 00E 010 012 014 016 018 01A 01C 01E 020 022 024 026 028 02A 02C 02E 030 032 034 036 038 03A 03C 03E 040 042 044 046 048 04A 04C 04E	001 003 005 007 009 00B 00D 00F 011 013 015 017 019 01B 01D 01F 021 023 025 027 029 02B 02D 02F 031 033 035 037 039 03B 03D 03F 041 043 045 047 049 04B 04D 04F	002 004 006 008 00A 00C 00E 010 012 014 016 018 01A 01C 01E 020 022 024 026 028 02A 02C 02E 030 032 034 036 038 03A 03C 03E 040 042 044 046 048 04A 04C 04E
Line 1	050 052 054 056 058 05A 05C 05E 060 062 064 066 068 06A 06C 06E 070 072 074 076 078 07A 07C 07E 080 082 084 086 088 08A 08C 08E 090 092 094 096 098 09A 09C 09E	051 053 055 057 059 05B 05D 05F 061 063 065 067 069 06B 06D 06F 071 073 075 077 079 07B 07D 07F 081 083 085 087 089 08B 08D 08F 091 093 095 097 099 09B 09D 09F	052 054 056 058 05A 05C 05E 060 062 064 066 068 06A 06C 06E 070 072 074 076 078 07A 07C 07E 080 082 084 086 088 08A 08C 08E 090 092 094 096 098 09A 09C 09E
Line 2	0A0 0A2 0A4 0A6 0A8 0AA 0AC 0AE 0B0 0B2 0B4 0B6 0B8 0BA 0BC 0BE 0C0 0C2 0C4 0C6 0C8 0CA 0CC 0CE 0D0 0D2 0D4 0D6 0D8 0DA 0DC 0DE 0E0 0E2 0E4 0E6 0E8 0EA 0EC 0EE	0A1 0A3 0A5 0A7 0A9 0AB 0AD 0AF 0B1 0B3 0B5 0B7 0B9 0BB 0BD 0BF 0C1 0C3 0C5 0C7 0C9 0CB 0CD 0CF 0D1 0D3 0D5 0D7 0D9 0DB 0DD 0DF 0E1 0E3 0E5 0E7 0E9 0EB 0ED 0EF	0A2 0A4 0A6 0A8 0AA 0AC 0AE 0B0 0B2 0B4 0B6 0B8 0BA 0BC 0BE 0C0 0C2 0C4 0C6 0C8 0CA 0CC 0CE 0D0 0D2 0D4 0D6 0D8 0DA 0DC 0DE 0E0 0E2 0E4 0E6 0E8 0EA 0EC 0EE
Line 3	0F0 0F2 0F4 0F6 0F8 0FA 0FC 0FE 100 102 104 106 108 10A 10C 10E 110 112 114 116 118 11A 11C 11E 120 122 124 126 128 12A 12C 12E 130 132 134 136 138 13A 13C 13E	0F1 0F3 0F5 0F7 0F9 0FB 0FD 0FF 101 103 105 107 109 10B 10D 10F 111 113 115 117 119 11B 11D 11F 121 123 125 127 129 12B 12D 12F 131 133 135 137 139 13B 13D 13F	0F2 0F4 0F6 0F8 0FA 0FC 0FE 100 102 104 106 108 10A 10C 10E 110 112 114 116 118 11A 11C 11E 120 122 124 126 128 12A 12C 12E 130 132 134 136 138 13A 13C 13E
Line 4	140 142 144 146 148 14A 14C 14E 150 152 154 156 158 15A 15C 15E 160 162 164 166 168 16A 16C 16E 170 172 174 176 178 17A 17C 17E 180 182 184 186 188 18A 18C 18E	141 143 145 147 149 14B 14D 14F 151 153 155 157 159 15B 15D 15F 161 163 165 167 169 16B 16D 16F 171 173 175 177 179 17B 17D 17F 181 183 185 187 189 18B 18D 18F	142 144 146 148 14A 14C 14E 150 152 154 156 158 15A 15C 15E 160 162 164 166 168 16A 16C 16E 170 172 174 176 178 17A 17C 17E 180 182 184 186 188 18A 18C 18E
Line 5	190 192 194 196 198 19A 19C 19E 1A0 1A2 1A4 1A6 1A8 1AA 1AC 1AE 1B0 1B2 1B4 1B6 1B8 1BA 1BC 1BE 1C0 1C2 1C4 1C6 1C8 1CA 1CC 1CE 1D0 1D2 1D4 1D6 1D8 1DA 1DC 1DE	191 193 195 197 199 19B 19D 19F 1A1 1A3 1A5 1A7 1A9 1AB 1AD 1AF 1B1 1B3 1B5 1B7 1B9 1BB 1BD 1BF 1C1 1C3 1C5 1C7 1C9 1CB 1CD 1CF 1D1 1D3 1D5 1D7 1D9 1DB 1DD 1DF	192 194 196 198 19A 19C 19E 1A0 1A2 1A4 1A6 1A8 1AA 1AC 1AE 1B0 1B2 1B4 1B6 1B8 1BA 1BC 1BE 1C0 1C2 1C4 1C6 1C8 1CA 1CC 1CE 1D0 1D2 1D4 1D6 1D8 1DA 1DC 1DE
Line 6	1E0 1E2 1E4 1E6 1E8 1EA 1EC 1EE 1F0 1F2 1F4 1F6 1F8 1FA 1FC 1FE 200 202 204 206 208 20A 20C 20E 210 212 214 216 218 21A 21C 21E 220 222 224 226 228 22A 22C 22E	1E1 1E3 1E5 1E7 1E9 1EB 1ED 1EF 1F1 1F3 1F5 1F7 1F9 1FB 1FD 1FF 201 203 205 207 209 20B 20D 20F 211 213 215 217 219 21B 21D 21F 221 223 225 227 229 22B 22D 22F	1E2 1E4 1E6 1E8 1EA 1EC 1EE 1F0 1F2 1F4 1F6 1F8 1FA 1FC 1FE 200 202 204 206 208 20A 20C 20E 210 212 214 216 218 21A 21C 21E 220 222 224 226 228 22A 22C 22E
Line 7	230 232 234 236 238 23A 23C 23E 240 242 244 246 248 24A 24C 24E 250 252 254 256 258 25A 25C 25E 260 262 264 266 268 26A 26C 26E 270 272 274 276 278 27A 27C 27E	231 233 235 237 239 23B 23D 23F 241 243 245 247 249 24B 24D 24F 251 253 255 257 259 25B 25D 25F 261 263 265 267 269 26B 26D 26F 271 273 275 277 279 27B 27D 27F	232 234 236 238 23A 23C 23E 240 242 244 246 248 24A 24C 24E 250 252 254 256 258 25A 25C 25E 260 262 264 266 268 26A 26C 26E 270 272 274 276 278 27A 27C 27E
Line 8	280 282 284 286 288 28A 28C 28E 290 292 294 296 298 29A 29C 29E 2A0 2A2 2A4 2A6 2A8 2AA 2AC 2AE 2B0 2B2 2B4 2B6 2B8 2BA 2BC 2BE 2C0 2C2 2C4 2C6 2C8 2CA 2CC 2CE	281 283 285 287 289 28B 28D 28F 291 293 295 297 299 29B 29D 29F 2A1 2A3 2A5 2A7 2A9 2AB 2AD 2AF 2B1 2B3 2B5 2B7 2B9 2BB 2BD 2BF 2C1 2C3 2C5 2C7 2C9 2CB 2CD 2CF	282 284 286 288 28A 28C 28E 290 292 294 296 298 29A 29C 29E 2A0 2A2 2A4 2A6 2A8 2AA 2AC 2AE 2B0 2B2 2B4 2B6 2B8 2BA 2BC 2BE 2C0 2C2 2C4 2C6 2C8 2CA 2CC 2CE
Line 9	2D0 2D2 2D4 2D6 2D8 2DA 2DC 2DE 2E0 2E2 2E4 2E6 2E8 2EA 2EC 2EE 2F0 2F2 2F4 2F6 2F8 2FA 2FC 2FE 300 302 304 306 308 30A 30C 30E 310 312 314 316 318 31A 31C 31E	2D1 2D3 2D5 2D7 2D9 2DB 2DD 2DF 2E1 2E3 2E5 2E7 2E9 2EB 2ED 2EF 2F1 2F3 2F5 2F7 2F9 2FB 2FD 2FF 301 303 305 307 309 30B 30D 30F 311 313 315 317 319 31B 31D 31F	2D2 2D4 2D6 2D8 2DA 2DC 2DE 2E0 2E2 2E4 2E6 2E8 2EA 2EC 2EE 2F0 2F2 2F4 2F6 2F8 2FA 2FC 2FE 300 302 304 306 308 30A 30C 30E 310 312 314 316 318 31A 31C 31E
Line 10	320 322 324 326 328 32A 32C 32E 330 332 334 336 338 33A 33C 33E 340 342 344 346 348 34A 34C 34E 350 352 354 356 358 35A 35C 35E 360 362 364 366 368 36A 36C 36E	321 323 325 327 329 32B 32D 32F 331 333 335 337 339 33B 33D 33F 341 343 345 347 349 34B 34D 34F 351 353 355 357 359 35B 35D 35F 361 363 365 367 369 36B 36D 36F	322 324 326 328 32A 32C 32E 330 332 334 336 338 33A 33C 33E 340 342 344 346 348 34A 34C 34E 350 352 354 356 358 35A 35C 35E 360 362 364 366 368 36A 36C 36E
Line 11	370 372 374 376 378 37A 37C 37E 380 382 384 386 388 38A 38C 38E 390 392 394 396 398 39A 39C 39E 3A0 3A2 3A4 3A6 3A8 3AA 3AC 3AE 3B0 3B2 3B4 3B6 3B8 3BA 3BC 3BE	371 373 375 377 379 37B 37D 37F 381 383 385 387 389 38B 38D 38F 391 393 395 397 399 39B 39D 39F 3A1 3A3 3A5 3A7 3A9 3AB 3AD 3AF 3B1 3B3 3B5 3B7 3B9 3BB 3BD 3BF	372 374 376 378 37A 37C 37E 380 382 384 386 388 38A 38C 38E 390 392 394 396 398 39A 39C 39E 3A0 3A2 3A4 3A6 3A8 3AA 3AC 3AE 3B0 3B2 3B4 3B6 3B8 3BA 3BC 3BE
Line 12	3C0 3C2 3C4 3C6 3C8 3CA 3CC 3CE 3D0 3D2 3D4 3D6 3D8 3DA 3DC 3DE 3E0 3E2 3E4 3E6 3E8 3EA 3EC 3EE 3F0 3F2 3F4 3F6 3F8 3FA 3FC 3FE 400 402 404 406 408 40A 40C 40E	3C1 3C3 3C5 3C7 3C9 3CB 3CD 3CF 3D1 3D3 3D5 3D7 3D9 3DB 3DD 3DF 3E1 3E3 3E5 3E7 3E9 3EB 3ED 3EF 3F1 3F3 3F5 3F7 3F9 3FB 3FD 3FF 401 403 405 407 409 40B 40D 40F	3C2 3C4 3C6 3C8 3CA 3CC 3CE 3D0 3D2 3D4 3D6 3D8 3DA 3DC 3DE 3E0 3E2 3E4 3E6 3E8 3EA 3EC 3EE 3F0 3F2 3F4 3F6 3F8 3FA 3FC 3FE 400 402 404 406 408 40A 40C 40E
Line 13	410 412 414 416 418 41A 41C 41E 420 422 424 426 428 42A 42C 42E 430 432 434 436 438 43A 43C 43E 440 442 444 446 448 44A 44C 44E 450 452 454 456 458 45A 45C 45E	411 413 415 417 419 41B 41D 41F 421 423 425 427 429 42B 42D 42F 431 433 435 437 439 43B 43D 43F 441 443 445 447 449 44B 44D 44F 451 453 455 457 459 45B 45D 45F	412 414 416 418 41A 41C 41E 420 422 424 426 428 42A 42C 42E 430 432 434 436 438 43A 43C 43E 440 442 444 446 448 44A 44C 44E 450 452 454 456 458 45A 45C 45E
Line 14	460 462 464 466 468 46A 46C 46E 470 472 474 476 478 47A 47C 47E 480 482 484 486 488 48A 48C 48E 490 492 494 496 498 49A 49C 49E 4A0 4A2 4A4 4A6 4A8 4AA 4AC 4AE	461 463 465 467 469 46B 46D 46F 471 473 475 477 479 47B 47D 47F 481 483 485 487 489 48B 48D 48F 491 493 495 497 499 49B 49D 49F 4A1 4A3 4A5 4A7 4A9 4AB 4AD 4AF	462 464 466 468 46A 46C 46E 470 472 474 476 478 47A 47C 47E 480 482 484 486 488 48A 48C 48E 490 492 494 496 498 49A 49C 49E 4A0 4A2 4A4 4A6 4A8 4AA 4AC 4AE
Line 15	4B0 4B2 4B4 4B6 4B8 4BA 4BC 4BE 4C0 4C2 4C4 4C6 4C8 4CA 4CC 4CE 4D0 4D2 4D4 4D6 4D8 4DA 4DC 4DE 4E0 4E2 4E4 4E6 4E8 4EA 4EC 4EE 4F0 4F2 4F4 4F6 4F8 4FA 4FC 4FE	4B1 4B3 4B5 4B7 4B9 4BB 4BD 4BF 4C1 4C3 4C5 4C7 4C9 4CB 4CD 4CF 4D1 4D3 4D5 4D7 4D9 4DB 4DD 4DF 4E1 4E3 4E5 4E7 4E9 4EB 4ED 4EF 4F1 4F3 4F5 4F7 4F9 4FB 4FD 4FF	4B2 4B4 4B6 4B8 4BA 4BC 4BE 4C0 4C2 4C4 4C6 4C8 4CA 4CC 4CE 4D0 4D2 4D4 4D6 4D8 4DA 4DC 4DE 4E0 4E2 4E4 4E6 4E8 4EA 4EC 4EE 4F0 4F2 4F4 4F6 4F8 4FA 4FC 4FE
Line 16	500 502 504 506 508 50A 50C 50E 510 512 514 516 518 51A 51C 51E 520 522 524 526 528 52A 52C 52E 530 532 534 536 538 53A 53C 53E 540 542 544 546 548 54A 54C 54E	501 503 505 507 509 50B 50D 50F 511 513 515 517 519 51B 51D 51F 521 523 525 527 529 52B 52D 52F 531 533 535 537 539 53B 53D 53F 541 543 545 547 549 54B 54D 54F	502 504 506 508 50A 50C 50E 510 512 514 516 518 51A 51C 51E 520 522 524 526 528 52A 52C 52E 530 532 534 536 538 53A 53C 53E 540 542 544 546 548 54A 54C 54E

Notes 1. The hexadecimal numbers in the boxes show the display RAM address.  
2. A character is set in 2 addresses (even address (upper stage in the figure 1) 16 bit + odd number address (lower step in the figure 1) 8 bits).

Figure 2.15.6 Address map 1 (continued)



M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Character 0		Character 39	
Line 17	550	552	554
	551	553	555
Line 18	5A0	5A2	5A4
	5A1	5A3	5A5
Line 19	5F0	5F2	5F4
	5F1	5F3	5F5
Line 20	640	642	644
	641	643	645
Line 21	690	692	694
	691	693	695
Line 22	6E0	6E2	6E4
	6E1	6E3	6E5
Line 23	730	732	734
	731	733	735
Line 24	780	782	784
	781	783	785
Notes 1. The hexadecimal numbers in the boxes show the display RAM address. 2. A character is set in 2 addresses (even address (upper stage in the figure 1) 16 bit + odd number address (lower step in the figure 1) 8 bits).			

Figure 2.15.7 Address map 2

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Teletext G1 character display

Can display teletext G1 character by setting character code 00<sub>16</sub> to font RAM bit (C7 to C0) of display RAM. SYRAM setting is invalid when set 00<sub>16</sub> to font RAM bit (C7 to C0), set G1 character by G1 character bit (G0 to G5) and G1 character form bit(G6). At the time, set 0 to all addresses of font RAM code 00<sub>16</sub> (font RAM addresses 000<sub>16</sub> to 009<sub>16</sub>).

Display RAM composition at G1 character display is shown in Figure 2.15.8.

Even address	CD15	CD14	CD13	CD12	CD11	CD10	CD9	CD8	CD7	CD6	CD5	CD4	CD3	CD2	CD1	CD0
	—	BB	BG	BR	BLINK	CB	CG	CR	0 <sub>(C7)</sub>	0 <sub>(C6)</sub>	0 <sub>(C5)</sub>	0 <sub>(C4)</sub>	0 <sub>(C3)</sub>	0 <sub>(C2)</sub>	0 <sub>(C1)</sub>	0 <sub>(C0)</sub>
Odd address	CD15	CD14	CD13	CD12	CD11	CD10	CD9	CD8	CD7	CD6	CD5	CD4	CD3	CD2	CD1	CD0
	0	0	0	0	0	0	0	0	G6 <sub>(SB)</sub>	G5 <sub>(SG)</sub>	G4 <sub>(SR)</sub>	—	G3 <sub>(SYC3)</sub>	G2 <sub>(SYC2)</sub>	G1 <sub>(SYC1)</sub>	G0 <sub>(SYC0)</sub>

Bit	Bit name	Function
0(C0)	Font RAM bit	Set 00 <sub>16</sub> when display teletext G1 character. At the time , set space in font RAM(00 <sub>16</sub> ).
0(C1)		
0(C2)		
0(C3)		
0(C4)		
0(C5)		
0(C6)		
0(C7)		
CR	G1 character color bit	Set color code of G1 character color.
CG		
CB		
BLINK	Blinking bit	0 : Do not blink 1 : Blink
BR	G1 character background color bit	Set color code of G1 character background color.
BG		
BB		
—	—	Must always be set to "0".
G0(SYC0)	G1 character bit(1)	Set G1 character by G0 to G5. (Refer to the next page.)
G1(SYC1)		
G2(SYC2)		
G3(SYC3)		
—	—	Must always be set to "0".
G4(SR)	G1 character bit(1)	Set G1 character by G0 to G5. (Refer to the next page.)
G5(SG)		
G6(SB)	G1 character form bit	0 : Contiguous form (Refer to the next page) 1 : Separated form

Figure 2.15.8 Display RAM bit composition(at G1 character displaying )

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

G1 character composition

Set G1 character by display RAM G1 character bit (G0 to G5) and G1 character form bit(G6). G1 character composition is shown in Figure 2.15.9. G1 character is divided to 6 blocks (refer to Figure 2.15.9), and set character by G0 to G5 in each block. Also, G1 character form is set by G6.  
Can display 64 patterns G1 character by using G0 to G5. G1 character composition is shown in Figure 2.15.10.

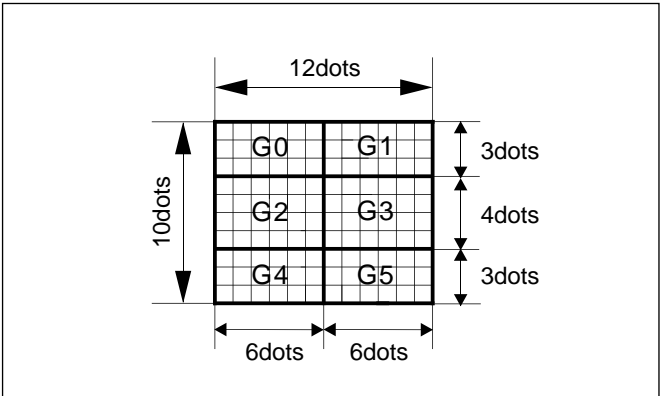


Figure 2.15.9 G1 character composition

	Example1:G0,G1,G5=1,G2,G3,G4=0	Example2:G3,G4=1,G0,G1,G2,G5=0
G6=0 (Contiguous form)		
G6=1 (Separated form)		

Figure 2.15.10 G1 character setting

Set 0 to G0 to G5 when use font RAM code 00<sub>16</sub> as normal character.  
However, SYRAM can not be displayed.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.15.4 Font RAM

Character font composited horizontal direction 12 dots X vertical direction 10 dots is set to font RAM code 00<sub>16</sub> to FF<sub>16</sub> (255 available, 7F<sub>16</sub>:blank code).

1 character setting is 10 address composite (12-bit X 10 addresses).

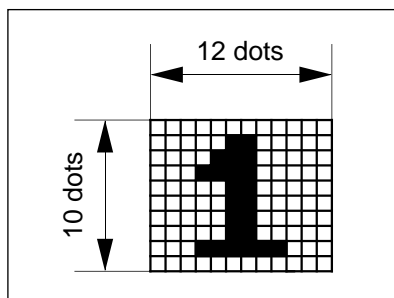
Setting character is displayed by specifying font RAM code to font RAM bit of display RAM. Font RAM code 00<sub>16</sub> is corresponds to Teletext G1 character. Then, font RAM code 7F<sub>16</sub> is fixed by blank, character font setting to this code is disable. Font RAM composition is shown in Table 2.15.4.

**Table 2.15.4 Font RAM composition**

Font RAM addresses (FA11 to FA0)	FD11	FD10	FD9	FD8	FD7	FD6	FD5	FD4	FD3	FD2	FD1	FD0	Remarks
00 <sub>16</sub>	F0B	F0A	F09	F08	F07	F06	F05	F04	F03	F02	F01	F00	Font RAM code (00 <sub>16</sub> )
001 <sub>16</sub>	F1B	F1A	F19	F18	F17	F16	F15	F14	F13	F12	F11	F10	
002 <sub>16</sub>	F2B	F2A	F29	F28	F27	F26	F25	F24	F23	F22	F21	F20	
003 <sub>16</sub>	F3B	F3A	F39	F38	F37	F36	F35	F34	F33	F32	F31	F30	
004 <sub>16</sub>	F4B	F4A	F49	F48	F47	F46	F45	F44	F43	F42	F41	F40	
005 <sub>16</sub>	F5B	F5A	F59	F58	F57	F56	F55	F54	F53	F52	F51	F50	
006 <sub>16</sub>	F6B	F6A	F69	F68	F67	F66	F65	F64	F63	F62	F61	F60	
007 <sub>16</sub>	F7B	F7A	F79	F78	F77	F76	F75	F74	F73	F72	F71	F70	
008 <sub>16</sub>	F8B	F8A	F89	F88	F87	F86	F85	F84	F83	F82	F81	F80	
009 <sub>16</sub>	F9B	F9A	F99	F98	F97	F96	F95	F94	F93	F92	F91	F90	
00A <sub>16</sub> ⋮ 00F <sub>16</sub>	Unused area												
010 <sub>16</sub> ⋮ 019 <sub>16</sub>	F0B ⋮ F9B	F0A ⋮ F9A	F09 ⋮ F99	F08 ⋮ F98	F07 ⋮ F97	F06 ⋮ F96	F05 ⋮ F95	F04 ⋮ F94	F03 ⋮ F93	F02 ⋮ F92	F01 ⋮ F91	F00 ⋮ F90	Font RAM code (01 <sub>16</sub> )
020 <sub>16</sub> ⋮ FD9 <sub>16</sub>	⋮												Font RAM code (02 <sub>16</sub> ) ⋮ Font RAM code (FD <sub>16</sub> )
FE0 <sub>16</sub> ⋮ FE9 <sub>16</sub>	F0B ⋮ F9B	F0A ⋮ F9A	F09 ⋮ F99	F08 ⋮ F98	F07 ⋮ F97	F06 ⋮ F96	F05 ⋮ F95	F04 ⋮ F94	F03 ⋮ F93	F02 ⋮ F92	F01 ⋮ F91	F00 ⋮ F90	Font RAM code (FE <sub>16</sub> )
FF0 <sub>16</sub> ⋮ FF9 <sub>16</sub>	F0B ⋮ F9B	F0A ⋮ F9A	F09 ⋮ F99	F08 ⋮ F98	F07 ⋮ F97	F06 ⋮ F96	F05 ⋮ F95	F04 ⋮ F94	F03 ⋮ F93	F02 ⋮ F92	F01 ⋮ F91	F00 ⋮ F90	Font RAM code (FF <sub>16</sub> )

For accessing to font RAM data, set accessing address (FA11 to FA0) (shown in Table 2.15.4) to font RAM address control register (0206<sub>16</sub>). Then write data (FD11 to FD0) by font RAM data control register (0208<sub>16</sub>). After data accessing fixed, font RAM address control register increments address automatically. Then, next address data writing is possible. Do not access to unused area (addresses xA<sub>16</sub> to xF<sub>16</sub>) of each Font RAM codes. But, when write data in succession, jump unused area and increments address automatically. (ex. increment automatically from address 009<sub>16</sub> to 010<sub>16</sub>).

Font composition is shown in Figure 2.15.11, Setting example is shown in Figure 2.15.12, Font RAM access registers are shown in Figure 2.15.13 and Font RAM access block diagram is shown in Figure 2.15.14.



**Figure 2.15.11 Font composition**

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

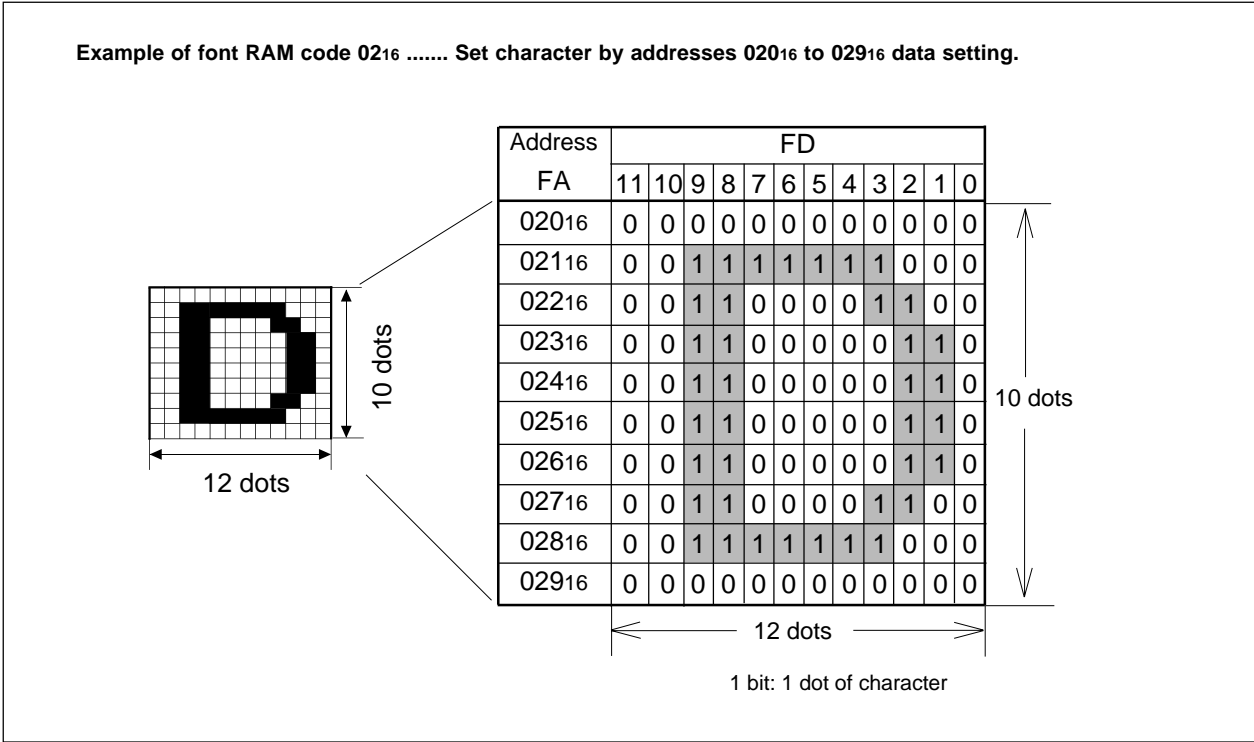


Figure 2.15.12 Setting example of font RAM

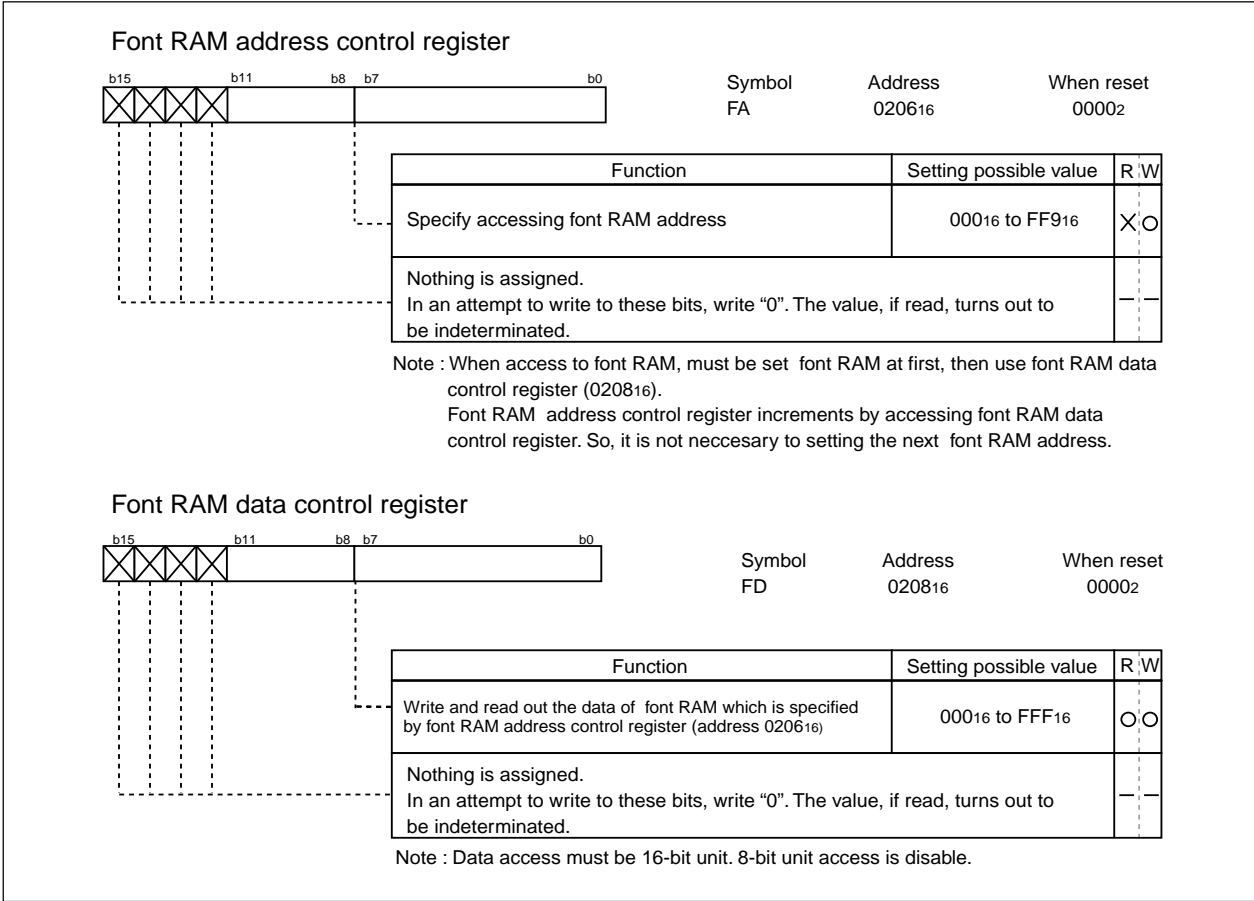
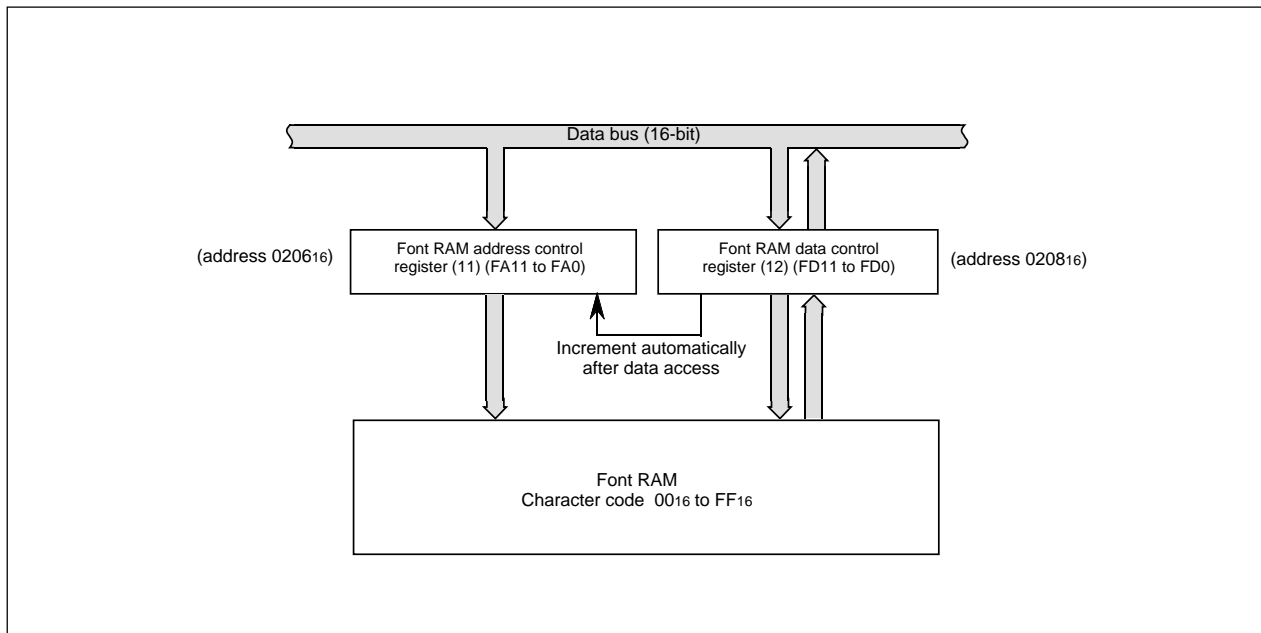


Figure 2.15.13 Font RAM access registers

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER



**Figure 2.15.14 Font RAM access block diagram**

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.15.5 SYRAM

Character font composite horizontal direction 12 dots X vertical direction 10 dots is set to SYRAM code 0<sub>16</sub> to E<sub>16</sub> (15 available).

Setting composite character is composed to font RAM by specifying SYRAM code to SYRAM bit of display RAM. Then, SYRAM code F<sub>16</sub> is fixed by blank, character font setting to this code is disable.

Use F<sub>16</sub> when SYRAM is not composed to character.

SYRAM composite is shown in Table 2.15.5.

**Table 2.15.5 SYRAM composition**

Font RAM addresses (FA <sub>10</sub> to FA <sub>0</sub> )	YD <sub>12</sub>	YD <sub>11</sub>	YD <sub>10</sub>	YD <sub>9</sub>	YD <sub>8</sub>	YD <sub>7</sub>	YD <sub>6</sub>	YD <sub>5</sub>	YD <sub>4</sub>	YD <sub>3</sub>	YD <sub>2</sub>	YD <sub>1</sub>	YD <sub>0</sub>	Remarks
00 <sub>16</sub>	SYEX <sub>0</sub>	SY0B	SY0A	SY09	SY08	SY07	SY06	SY05	SY04	SY03	SY02	SY01	SY00	SYRAM code (0 <sub>16</sub> )
01 <sub>16</sub>	SYEX <sub>1</sub>	SY1B	SY1A	SY19	SY18	SY17	SY16	SY15	SY14	SY13	SY12	SY11	SY10	
02 <sub>16</sub>	SYEX <sub>2</sub>	SY2B	SY2A	SY29	SY28	SY27	SY26	SY25	SY24	SY23	SY22	SY21	SY20	
03 <sub>16</sub>	SYEX <sub>3</sub>	SY3B	SY3A	SY39	SY38	SY37	SY36	SY35	SY34	SY33	SY32	SY31	SY30	
04 <sub>16</sub>	SYEX <sub>4</sub>	SY4B	SY4A	SY49	SY48	SY47	SY46	SY45	SY44	SY43	SY42	SY41	SY40	
05 <sub>16</sub>	SYEX <sub>5</sub>	SY5B	SY5A	SY59	SY58	SY57	SY56	SY55	SY54	SY53	SY52	SY51	SY50	
06 <sub>16</sub>	SYEX <sub>6</sub>	SY6B	SY6A	SY69	SY68	SY67	SY66	SY65	SY64	SY63	SY62	SY61	SY60	
07 <sub>16</sub>	SYEX <sub>7</sub>	SY7B	SY7A	SY79	SY78	SY77	SY76	SY75	SY74	SY73	SY72	SY71	SY70	
08 <sub>16</sub>	SYEX <sub>8</sub>	SY8B	SY8A	SY89	SY88	SY87	SY86	SY85	SY84	SY83	SY82	SY81	SY80	
09 <sub>16</sub>	SYEX <sub>9</sub>	SY9B	SY9A	SY99	SY98	SY97	SY96	SY95	SY94	SY93	SY92	SY91	SY90	
0A <sub>16</sub> ⋮ 0F <sub>16</sub>	Unused area													
10 <sub>16</sub> ⋮ 19 <sub>16</sub>	SYEX <sub>0</sub> ⋮ SYEX <sub>9</sub>	SY0B ⋮ SY9B	SY0A ⋮ SY9A	SY09 ⋮ SY99	SY08 ⋮ SY98	SY07 ⋮ SY97	SY06 ⋮ SY96	SY05 ⋮ SY95	SY04 ⋮ SY94	SY03 ⋮ SY93	SY02 ⋮ SY92	SY01 ⋮ SY91	SY00 ⋮ SY90	SYRAM code (1 <sub>16</sub> )
20 <sub>16</sub> ⋮ C9 <sub>16</sub>	⋮													SYRAM code (2 <sub>16</sub> ) ⋮ SYRAM code (C <sub>16</sub> )
D0 <sub>16</sub> ⋮ D9 <sub>16</sub>	SYEX <sub>0</sub> ⋮ SYEX <sub>9</sub>	SY0B ⋮ SY9B	SY0A ⋮ SY9A	SY09 ⋮ SY99	SY08 ⋮ SY98	SY07 ⋮ SY97	SY06 ⋮ SY96	SY05 ⋮ SY95	SY04 ⋮ SY94	SY03 ⋮ SY93	SY02 ⋮ SY92	SY01 ⋮ SY91	SY00 ⋮ SY90	SYRAM code (D <sub>16</sub> )
E0 <sub>16</sub> ⋮ E9 <sub>16</sub>	SYEX <sub>0</sub> ⋮ SYEX <sub>9</sub>	SY9B	SY0A	SY09	SY08	SY07	SY06	SY05	SY04	SY03	SY02	SY01	SY00	SYRAM code (E <sub>16</sub> )

For accessing to SYRAM data, set accessing address (YA<sub>7</sub> to YA<sub>0</sub>) (shown in Table 2.15.5) to SYRAM address control register (020A<sub>16</sub>). Then write data (YD<sub>12</sub> to YD<sub>0</sub>) by SYRAM data control register (020C<sub>16</sub>). When end the accessing, SYRAM address control register increments address automatically. Then, next address data writing is possible. Do not access to unused area (addresses xA<sub>16</sub> to xF<sub>16</sub>) of each SYRAM codes. But, when write data in succession, jump unused area and increments address automatically. (ex. increment automatically from address 09<sub>16</sub> to 10<sub>16</sub>).

Setting example is shown in Figure 2.15.15, SYRAM access registers are shown in Figure 2.15.16 and SYRAM access block diagram is shown in Figure 2.15.17.

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

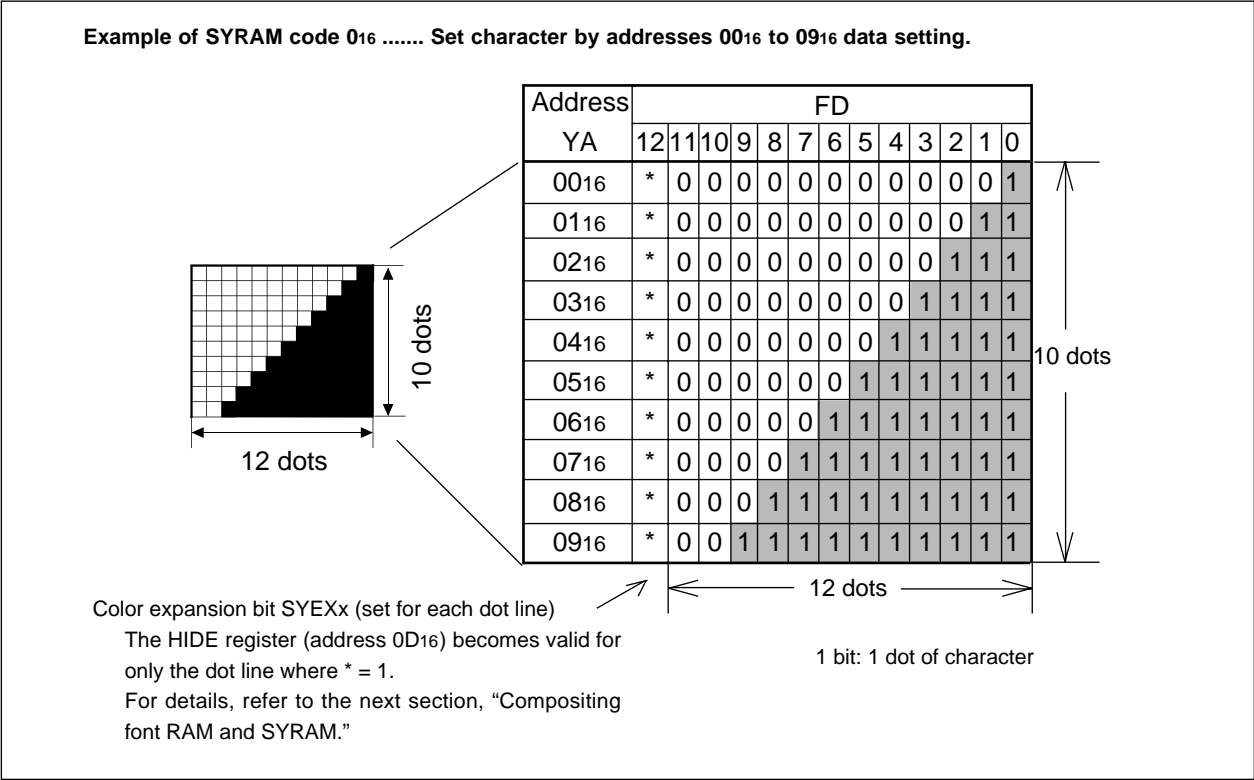


Figure 2.15.15 Setting example of SYRAM

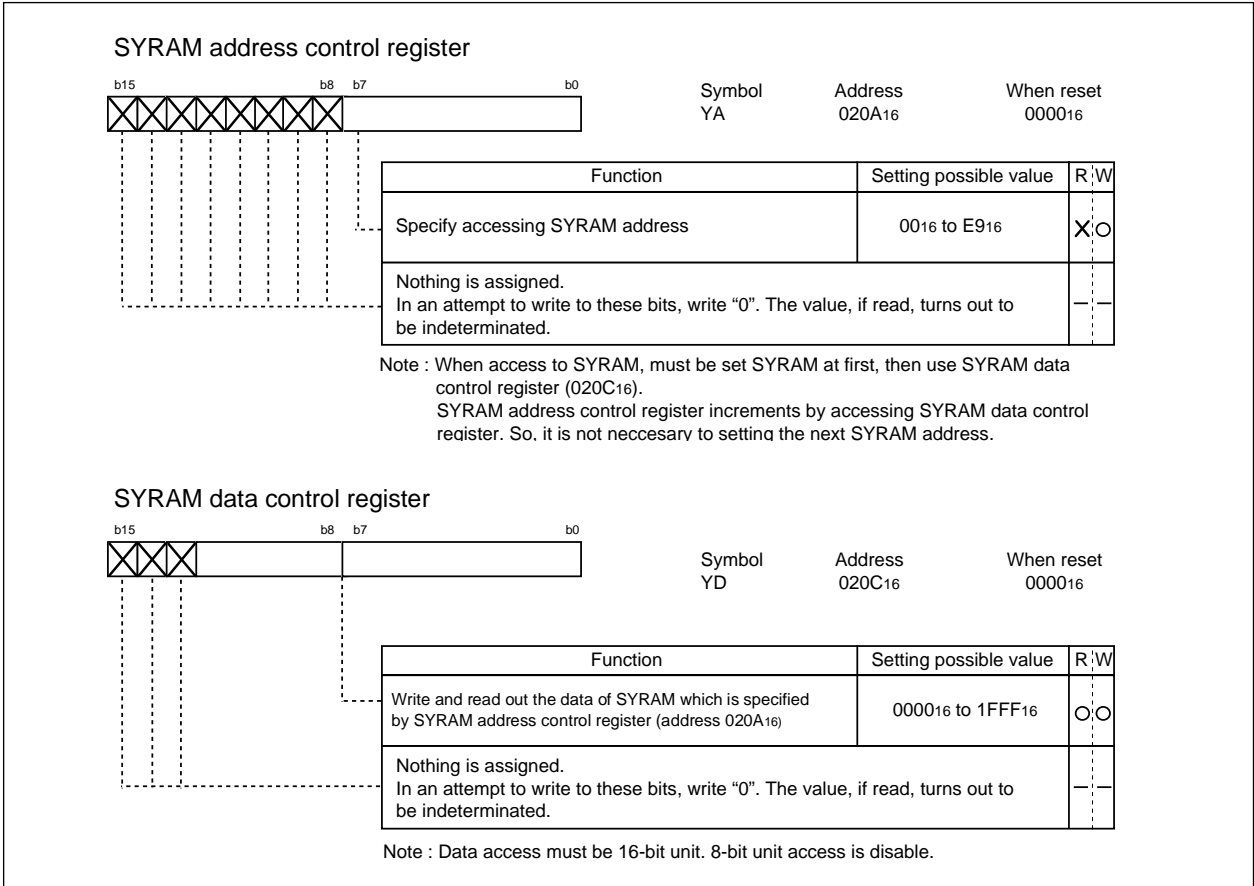
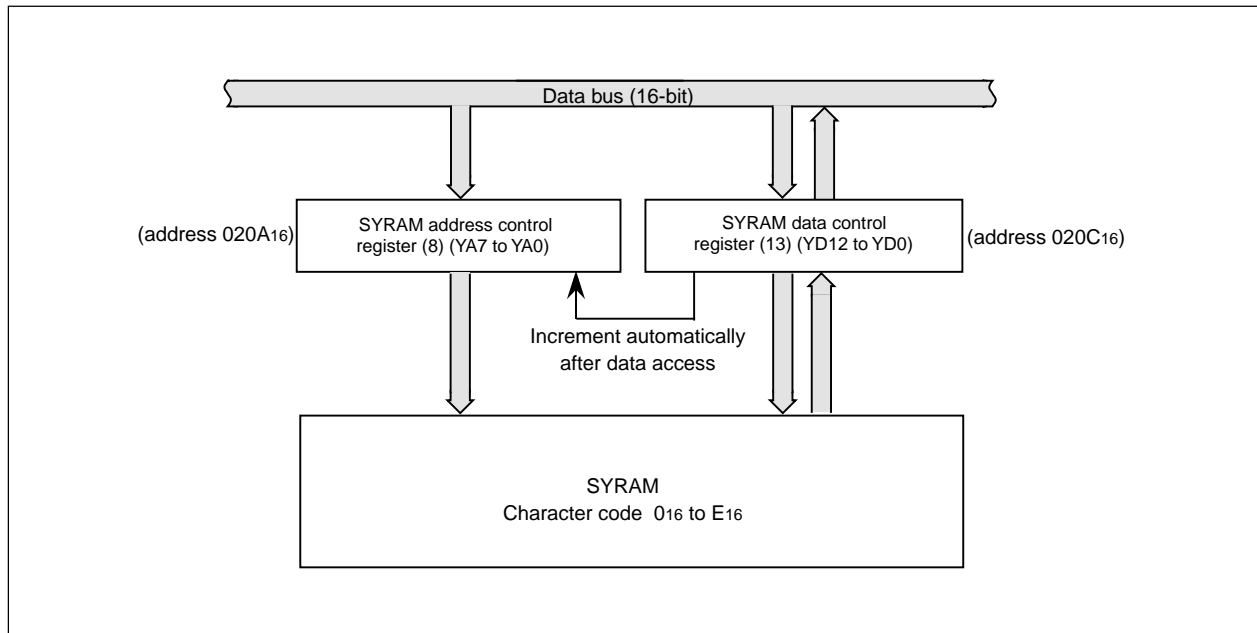


Figure 2.15.16 SYRAM access registers



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER



**Figure 2.15.17 SYRAM access block diagram**

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Compositing font RAM and SYRAM

Can composite characters in font RAM with SYRAM.

The compositing method is determined by the SYEXx color expansion bit and the HIDE register (address 0D16).

For dot lines where SYEXx = 0, the SYRAM color is set by the display RAM's SR, SG, and SB irrespective of the HIDE register's content.

If the HIDE register's content is 0, the SYRAM color for dot lines where SYEXx = 1 is set by the registers LINER, LINEG, and LINEB (address 0816).

If the HIDE register's content is 1, the font RAM part of the dot lines where SYEXx = 1 is overwritten in HIDE mode with colors set by the registers LINER, LINEG, and LINEB irrespective of the font RAM's content and color. The color of the SYRAM part is set by the display RAM's SR, SG, and SB as in the case of dot lines where SYEXx = 0.

Figure 2.15.18 shows an example for each instance of compositing.

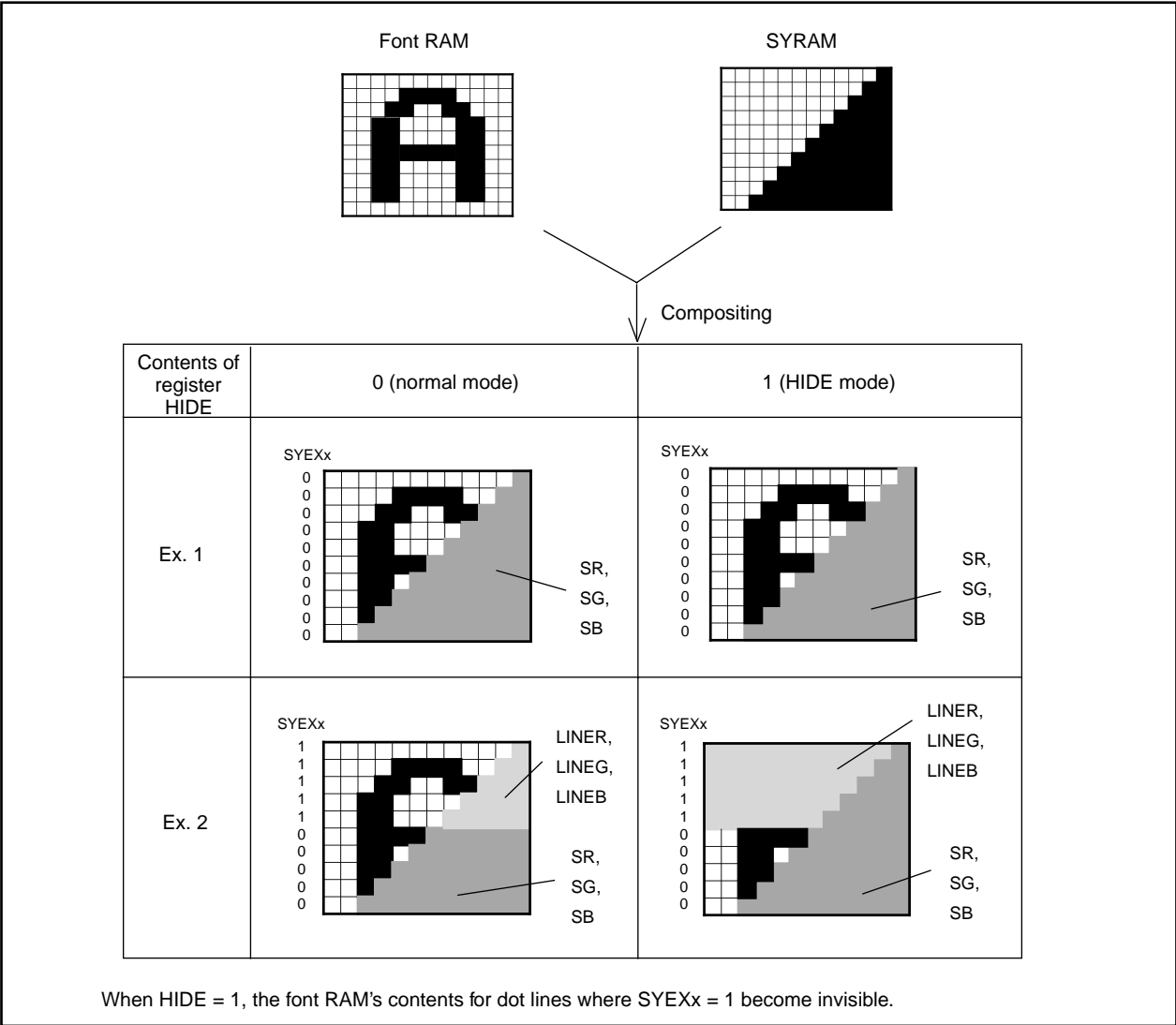


Figure 2.15.18 Compositing example

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.15.6 Slice RAM

Store 18-line slice data. There are 3 types of Slice data : PDC, VPS and VBI. All data are stored to addresses which corresponds to slicing line (ex. 22 line' data is stored to addresses 200<sub>16</sub> to 217<sub>16</sub>). 24 addresses (SR00x to SR17x) are prepared for 1 line, slice data is stored in order from LSB side. Then, slice datas and field information are stored to the top address of each line. Slice RAM composite is shown in Table 2.15.6.

**Table 2.15.6 Slice RAM composition**

Slice RAM addresses (SA9 to SA0)	SD15	SD14	SD13	SD12	SD11	SD10	SD9	SD8	SD7	SD6	SD5	SD4	SD3	SD2	SD1	SD0	Remarks
000 <sub>16</sub> 001 <sub>16</sub> ⋮ 016 <sub>16</sub> 017 <sub>16</sub>	SR00F SR01F ⋮ SR16F SR17F	SR00E SR01E ⋮ SR16E SR17E	SR00D SR01D ⋮ SR16D SR17D	SR00C SR01C ⋮ SR16C SR17C	SR00B SR01B ⋮ SR16B SR17B	SR00A SR01A ⋮ SR16A SR17A	SR009 SR019 ⋮ SR169 SR179	SR008 SR018 ⋮ SR168 SR178	SR007 SR017 ⋮ SR167 SR177	SR006 SR016 ⋮ SR166 SR176	SR005 SR015 ⋮ SR165 SR175	SR004 SR014 ⋮ SR164 SR174	SR003 SR013 ⋮ SR163 SR173	SR002 SR012 ⋮ SR162 SR172	SR001 SR011 ⋮ SR161 SR171	SR000 SR010 ⋮ SR160 SR170	6th line or 318th line slice data
018 <sub>16</sub> ⋮ 01F <sub>16</sub>	Unused area																
020 <sub>16</sub> ⋮ 037 <sub>16</sub>	SR00F ⋮ SR17F	SR00E ⋮ SR17E	SR00D ⋮ SR17D	SR00C ⋮ SR17C	SR00B ⋮ SR17B	SR00A ⋮ SR17A	SR009 ⋮ SR179	SR008 ⋮ SR178	SR007 ⋮ SR177	SR006 ⋮ SR176	SR005 ⋮ SR175	SR004 ⋮ SR174	SR003 ⋮ SR173	SR002 ⋮ SR172	SR001 ⋮ SR171	SR000 ⋮ SR170	7th line or 319 th line slice data
040 <sub>16</sub> ⋮ 1F7 <sub>16</sub>	⋮																8th line to 21th line or 320th line to 333 line slice data
200 <sub>16</sub> ⋮ 217 <sub>16</sub>	SR00F ⋮ SR17F	SR00E ⋮ SR17E	SR00D ⋮ SR17D	SR00C ⋮ SR17C	SR00B ⋮ SR17B	SR00A ⋮ SR17A	SR009 ⋮ SR179	SR008 ⋮ SR178	SR007 ⋮ SR177	SR006 ⋮ SR176	SR005 ⋮ SR175	SR004 ⋮ SR174	SR003 ⋮ SR173	SR002 ⋮ SR172	SR001 ⋮ SR171	SR000 ⋮ SR170	22th line or 334th line slice data
220 <sub>16</sub> ⋮ 237 <sub>16</sub>	SR00F ⋮ SR17F	SR00E ⋮ SR17E	SR00D ⋮ SR17D	SR00C ⋮ SR17C	SR00B ⋮ SR17B	SR00A ⋮ SR17A	SR009 ⋮ SR179	SR008 ⋮ SR178	SR007 ⋮ SR177	SR006 ⋮ SR176	SR005 ⋮ SR175	SR004 ⋮ SR174	SR003 ⋮ SR173	SR002 ⋮ SR172	SR001 ⋮ SR171	SR000 ⋮ SR170	23th line or 335th line slice data

For accessing to slice RAM data, set accessing address (SA9 to SA0) (shown in Table 2.15.6) to slice RAM address control register (address 020E<sub>16</sub>). Then read out data from slice RAM data control register (address 0210<sub>16</sub>). When end the data reading, slice RAM address control register increments address automatically. Then, next address data reading is possible. Do not access to unused area of each character codes. Must set address to each line because unused area has no address' automatically increment.

Slice RAM bit composition is shown in Figure 2.15.19, Slice RAM access registers are shown in Figure 2.15.20 and Slice RAM access block diagram is shown in Figure 2.15.21.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

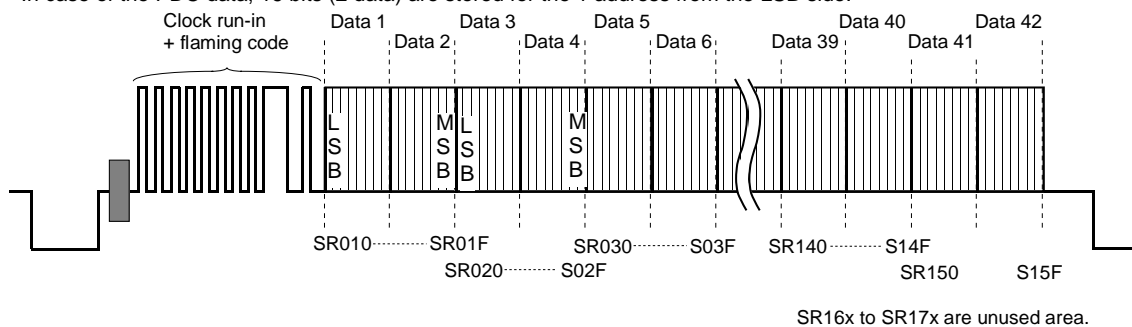
The each head address of the address is corresponded to slicing line has stored next slice information.

	SR00F to SR004	SR003	SR002	SR001	SR000
PDC	0	field * (Note)	0	0	1
VPS	0	field * (Note)	0	1	0
VBI	0	field * (Note)	1	0	0
Other	0	0	0	0	0

Note : \* the first field : 1  
the second field : 0

## (1) PDC

In case of the PDC data, 16 bits (2 data) are stored for the 1 address from the LSB side.

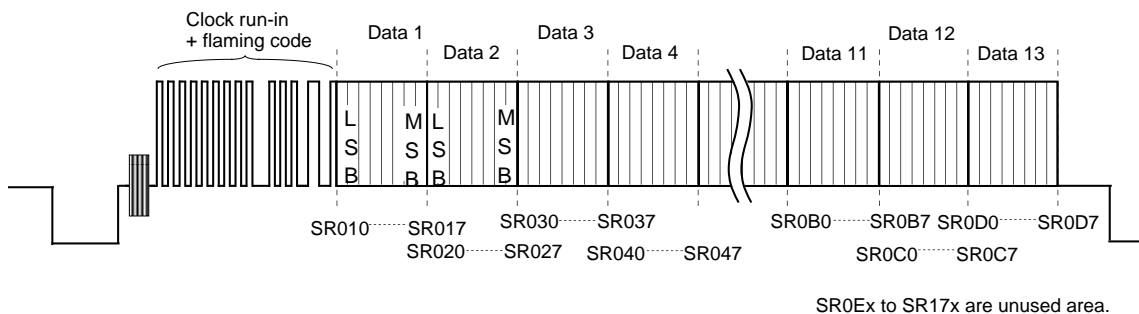


## (2) VPS

In case of the VPS data, 8 bits (a data) are stored for an address from the LSB side.

Low-order 8 bits stores the slice data. And, high-order 8 bits become warning bit, when the send data is not recognized as bi-phase type.

The case of bi-phase data = "1,0" or "0,1" (the bi-phase type) becomes "0" for this warning bit, and it becomes "1" in bi-phase data = "0,0" or "1,1" (it is not the bi-phase type). (For example, bi-phase data of SR011 is "0,0" or "1,1", "1" is set to SR019.)



## (3) VBI

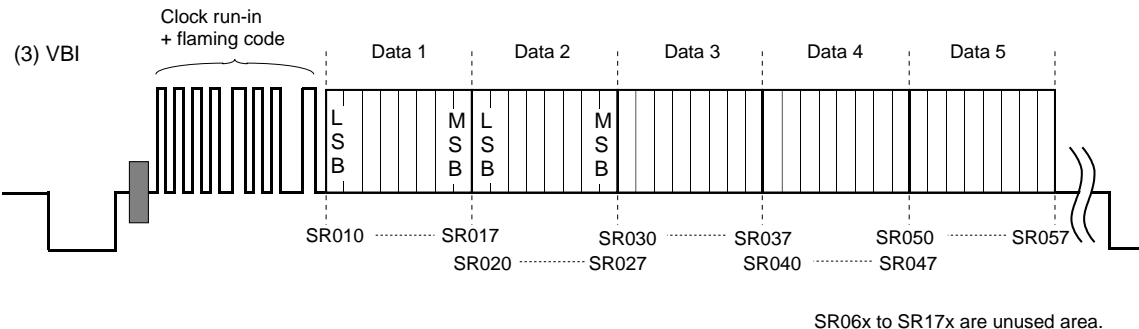


Figure 2.15.19 Slice RAM bit composition

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

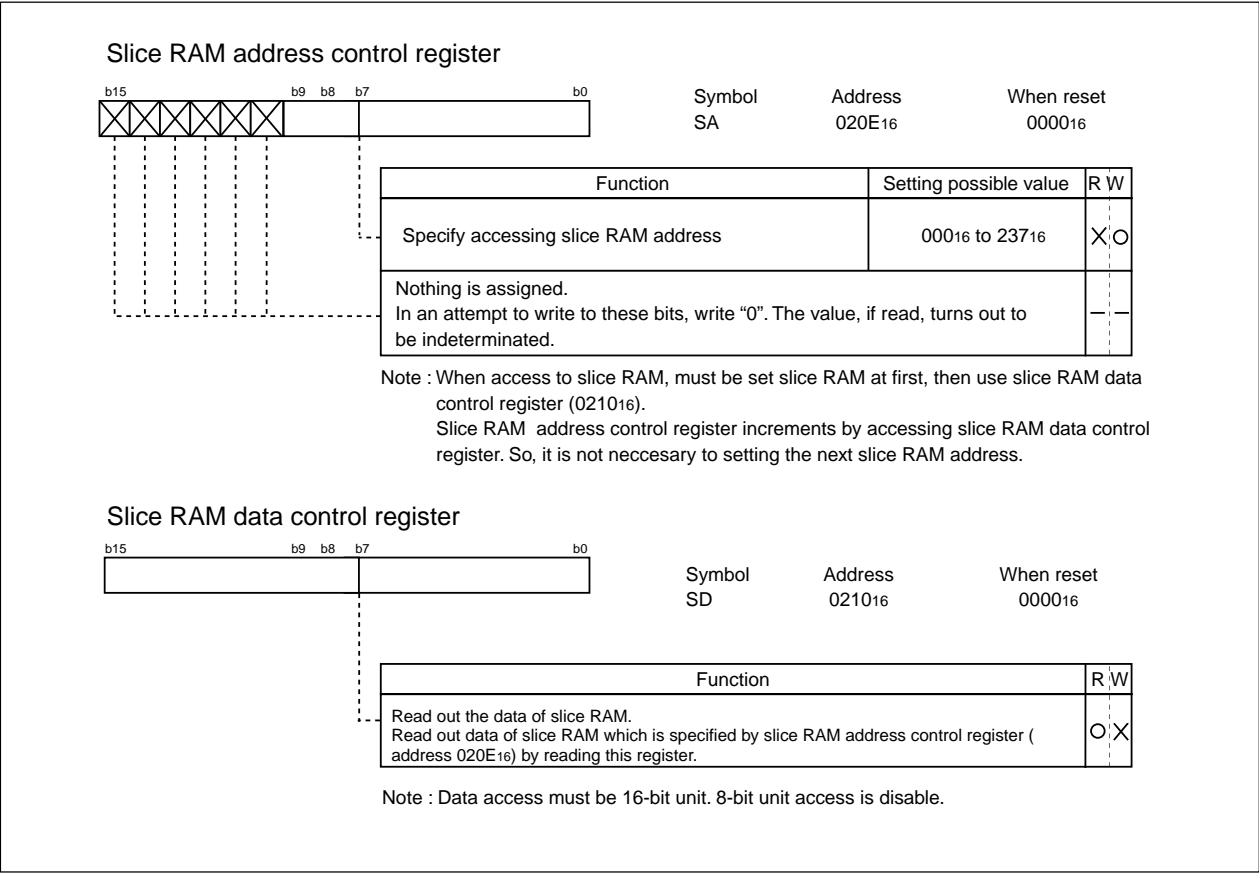


Figure 2.15.20 Slice RAM access registers

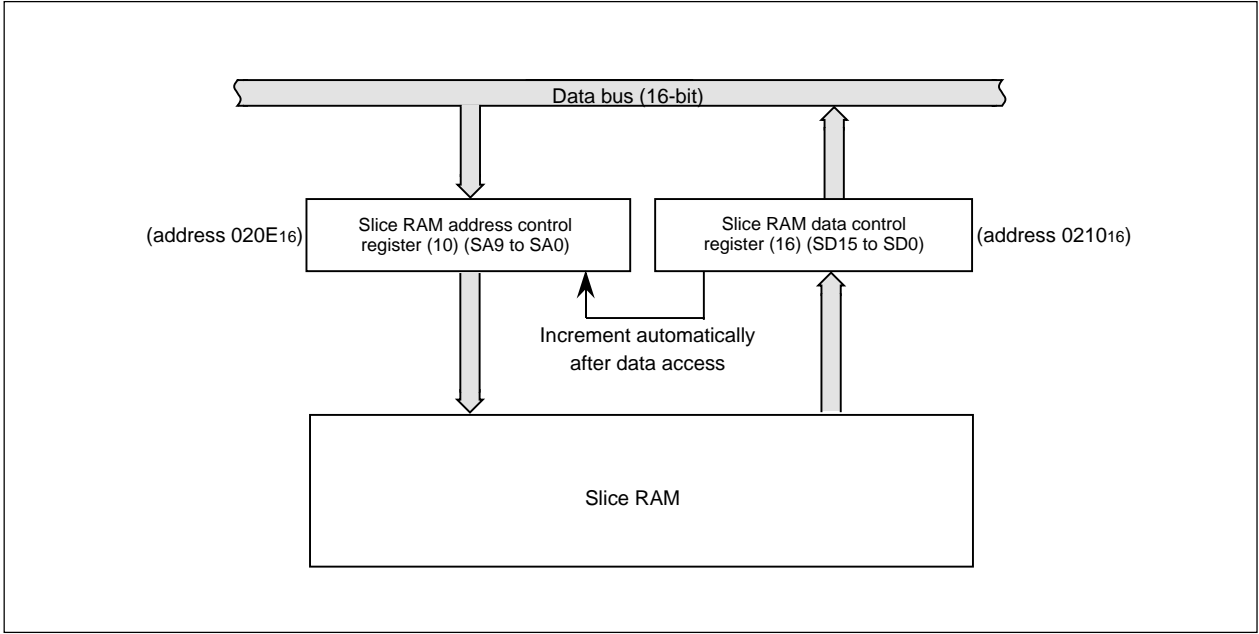


Figure 2.15.21 Slice RAM access block diagram

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

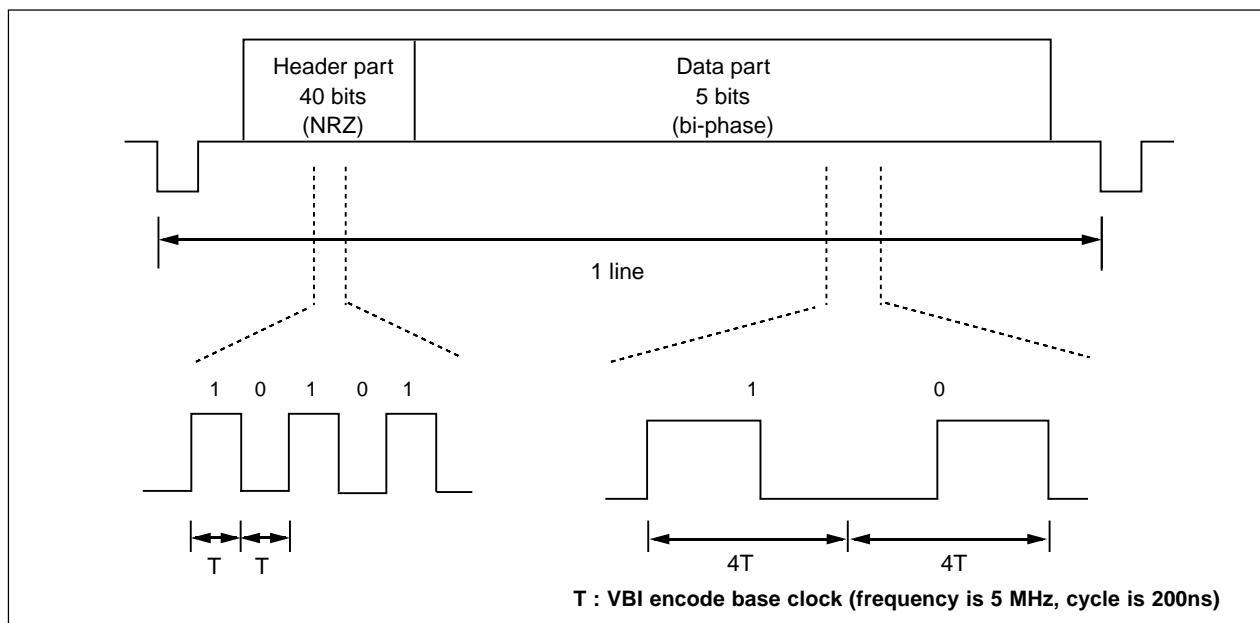
## 2.15.7 VBIRAM

Set 18-line VBI encode data. 5 addresses (8-bit X 5) are prepared for 1 line, output data in order from LSB side in bi-phase type. Specify output pattern (the NRZ type) of header (clock-run in and framing code) (each line command) at addresses 00<sub>16</sub> to 04<sub>16</sub>.

VBIRAM composite is shown in Table 2.15.7, VBI encode data composite is shown in Figure 2.15.20.

**Table 2.15.7 VBIRAM composition**

VBIRAM addresses (EA6 to EA0)	ED7	ED6	ED5	ED4	ED3	ED2	ED1	ED0	Remarks
00 <sub>16</sub>	VF07	VF06	VF05	VF04	VF03	VF02	VF01	VF00	Specify Clock-run in and Framing code pattern. 1-bit corresponds to 1T (Max.40 bits). Outputs before data in each line (each line common).
01 <sub>16</sub>	VF17	VF16	VF15	VF14	VF13	VF12	VF11	VF10	
02 <sub>16</sub>	VF27	VF26	VF25	VF24	VF23	VF22	VF21	VF20	
03 <sub>16</sub>	VF37	VF36	VF35	VF34	VF33	VF32	VF31	VF30	
04 <sub>16</sub>	VF47	VF46	VF45	VF44	VF43	VF42	VF41	VF40	
05 <sub>16</sub>	VR07	VR06	VR05	VR04	VR03	VR02	VR01	VF00	Specify output data of 6th line and 318th line. 1-bit corresponds to bi-phase 1-bit (4T).
06 <sub>16</sub>	VR17	VR16	VR15	VR14	VR13	VR12	VR11	VF10	
07 <sub>16</sub>	VR27	VR26	VR25	VR24	VR23	VR22	VR21	VF20	
08 <sub>16</sub>	VR37	VR36	VR35	VR34	VR33	VR32	VR31	VF30	
09 <sub>16</sub>	VR47	VR46	VR45	VR44	VR43	VR42	VR41	VF40	
0A <sub>16</sub>	VR07	VR06	VR05	VR04	VR03	VR02	VR01	VF00	Specify output data of 7th line and 319th line. 1-bit corresponds to bi-phase 1-bit (4T).
0B <sub>16</sub>	VR17	VR16	VR15	VR14	VR13	VR12	VR11	VF10	
0C <sub>16</sub>	VR27	VR26	VR25	VR24	VR23	VR22	VR21	VF20	
0D <sub>16</sub>	VR37	VR36	VR35	VR34	VR33	VR32	VR31	VF30	
0E <sub>16</sub>	VR47	VR46	VR45	VR44	VR43	VR42	VR41	VF40	
...	...	...	...	...	...	...	...	...	...
55 <sub>16</sub>	VR07	VR06	VR05	VR04	VR03	VR02	VR01	VF00	Specify output data of 22th line and 334th line. 1-bit corresponds to bi-phase 1-bit (4T).
56 <sub>16</sub>	VR17	VR16	VR15	VR14	VR13	VR12	VR11	VF10	
57 <sub>16</sub>	VR27	VR26	VR25	VR24	VR23	VR22	VR21	VF20	
58 <sub>16</sub>	VR37	VR36	VR35	VR34	VR33	VR32	VR31	VF30	
59 <sub>16</sub>	VR47	VR46	VR45	VR44	VR43	VR42	VR41	VF40	
5A <sub>16</sub>	VR07	VR06	VR05	VR04	VR03	VR02	VR01	VF00	Specify output data of 23th line and 335th line. 1-bit corresponds to bi-phase 1-bit (4T).
5B <sub>16</sub>	VR17	VR16	VR15	VR14	VR13	VR12	VR11	VF10	
5C <sub>16</sub>	VR27	VR26	VR25	VR24	VR23	VR22	VR21	VF20	
5D <sub>16</sub>	VR37	VR36	VR35	VR34	VR33	VR32	VR31	VF30	
5E <sub>16</sub>	VR47	VR46	VR45	VR44	VR43	VR42	VR41	VF40	



**Figure 2.15.22 VBIRAM encode data composition**

For accessing to VBIRAM data, set accessing address (EA) (shown in Table 2.15.7) to VBIRAM address control register (address 0212<sub>16</sub>). Then write data (ED) from VBIRAM data control register (address 0214<sub>16</sub>). When end the data accessing, VBIRAM address control register increments address automatically. Then, next address data writing is possible.

VBIRAM access registers are shown in Figure 2.15.23 and VBIRAM access block diagram is shown in Figure 2.15.24.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

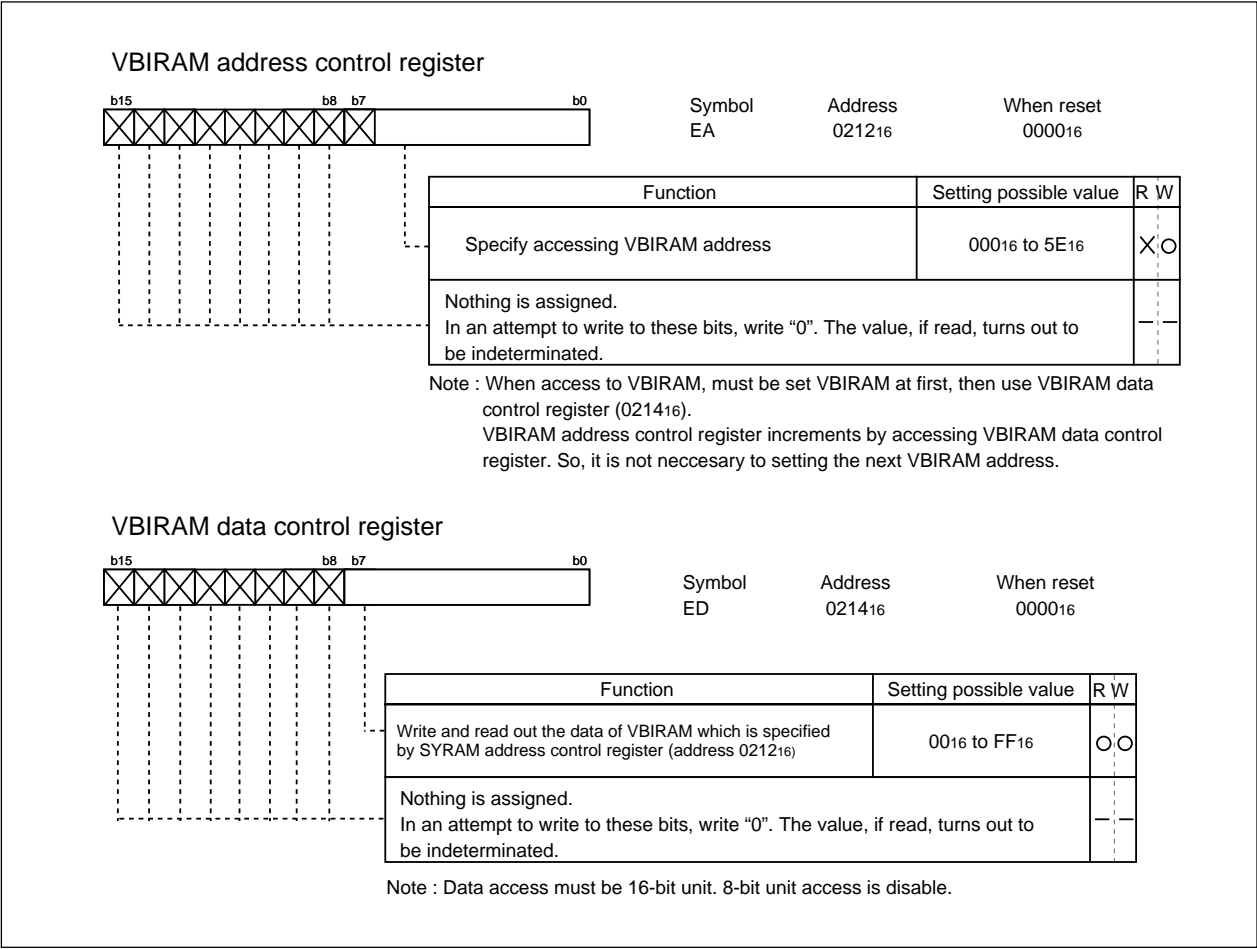


Figure 2.15.23 VBIRAM access registers

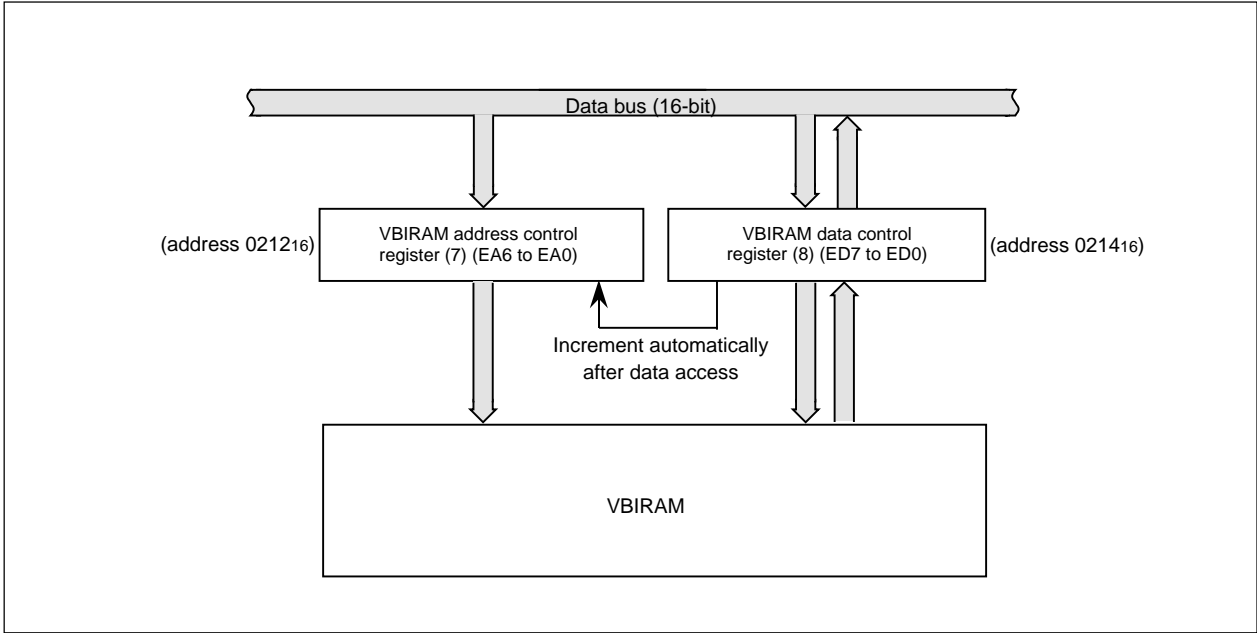


Figure 2.15.24 VBIRAM access block

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**(1) Setting of Clock-run in and Flaming code**

Specify clock-run in and flaming code output pattern at VBIRAM addresses 00<sub>16</sub> to 04<sub>16</sub> (40 bits). Data 1-bit corresponds to 1T, every byte is output at LSB first.  
When clock-run in and flaming code are less than 40 bits (40T), put "0" to the top (from the end, set "0" to unused bit). This pattern of every line is common, outputting before data of every line. Example of setting is shown in Figure 2.15.25.

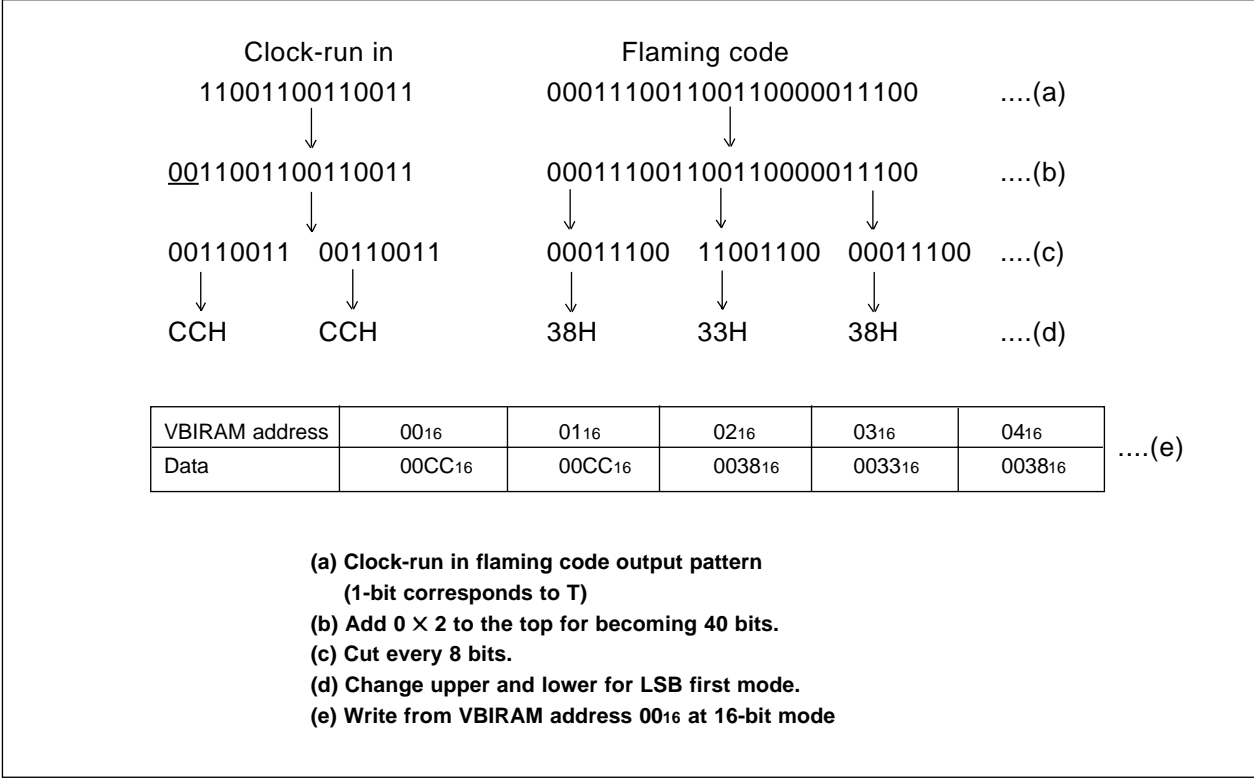


Figure 2.15.25 Example of setting

**(2) Data setting**

Set 5 bytes data for 1 line. Setting data is output in bi-phase method. VBI data 1 bit is corresponds to output bi-phase 1 bit (4T). Data specifying is set to RAM which is corresponds to RAM corresponding to the line specifying composition at expansion register VBIL0 to VBIL17. When set to RAM of unspecific line, output is disable.



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

### (3) EDO2 (VBI-BLKN) signal output specification

EDO2 signal (BLNK signal for VBI signal) output including former 1.8 $\mu$ s and outer 6.4 $\mu$ s of VBI encode data. Example of output timing is shown in Figure 2.15.26.

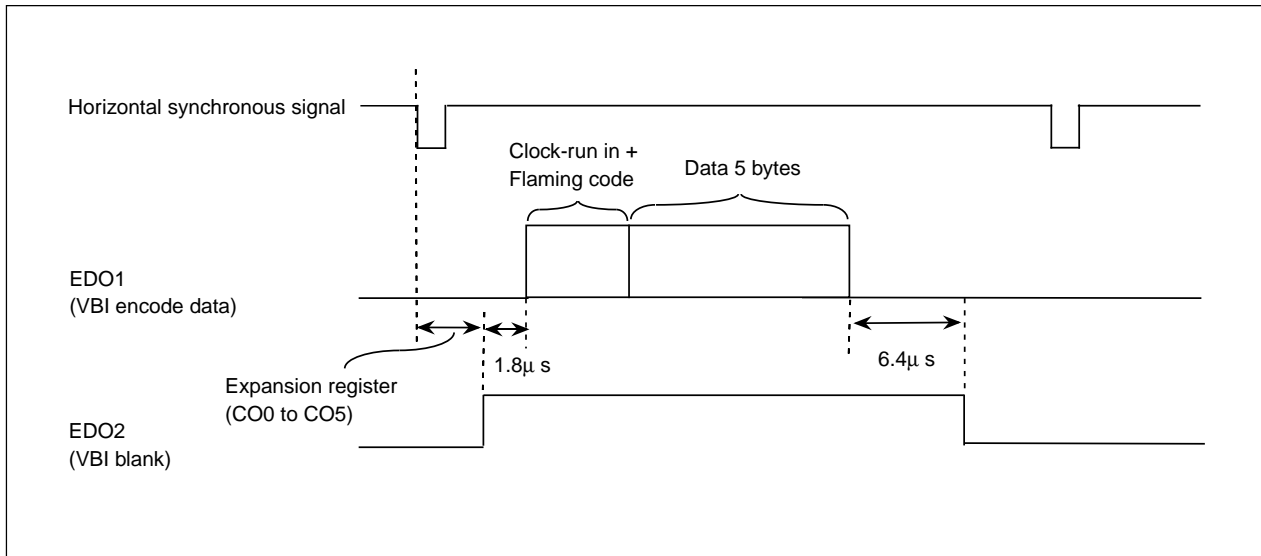


Figure 2.15.26 Example of output timing

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.15.8 Expansion Register

Control function of OSD function, Data slicer function and VBI encoder function. Expansion register composition is shown in Table 2.15.8.

**Table 2.15.8 Expansion register composition**

DA5 to DA0	DD15	DD14	DD13	DD12	DD11	DD10	DD9	DD8	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0	Remarks
00 <sup>16</sup>	PTC6	PTC5	PTC4	PTC3	PTC2	PTC1	PTC0	STBY0	VP7	VP6	VP5	VP4	VP3	VP2	VP1	VP0	Vertical display position, Port setting
01 <sup>16</sup>	PTD6	PTD5	PTD4	PTD3	PTD2	PTD1	PTD0	HP8	HP7	HP6	HP5	HP4	HP3	HP2	HP1	HP0	Horizontal display position, Port setting
02 <sup>16</sup>	HSZ15	HSZ14	HSZ13	HSZ12	HSZ11	HSZ10	HSZ9	HSZ8	HSZ7	HSZ6	HSZ5	HSZ4	HSZ3	HSZ2	HSZ1	HSZ0	Horizontal size setting
03 <sup>16</sup>	—	—	—	TEST2	TEST1	TEST0	BCOL	HSZ24	HSZ23	HSZ22	HSZ21	HSZ20	HSZ19	HSZ18	HSZ17	HSZ16	Horizontal size setting
04 <sup>16</sup>	VSZ15	VSZ14	VSZ13	VSZ12	VSZ11	VSZ10	VSZ9	VSZ8	VSZ7	VSZ6	VSZ5	VSZ4	VSZ3	VSZ2	VSZ1	VSZ0	Vertical size setting
05 <sup>16</sup>	—	—	—	—	BLINK2	BLINK1	BLINK0	VSZ24	VSZ23	VSZ22	VSZ21	VSZ20	VSZ19	VSZ18	VSZ17	VSZ16	Vertical size and blinking setting
06 <sup>16</sup>	DSP015	DSP014	DSP013	DSP012	DSP011	DSP010	DSP09	DSP08	DSP07	DSP06	DSP05	DSP04	DSP03	DSP02	DSP01	DSP00	Display mode setting
07 <sup>16</sup>	—	C05	C04	C03	C02	C01	C00	DSP024	DSP023	DSP022	DSP021	DSP020	DSP019	DSP018	DSP017	DSP016	Display mode setting
08 <sup>16</sup>	DSP115	DSP114	DSP113	DSP112	DSP111	DSP110	DSP19	DSP18	DSP17	DSP16	DSP15	DSP14	DSP13	DSP12	DSP11	DSP10	Display mode setting
09 <sup>16</sup>	—	—	—	—	—	—	—	DSP124	DSP123	DSP122	DSP121	DSP120	DSP119	DSP118	DSP117	DSP116	Display mode setting
0A <sup>16</sup>	—	—	—	GRYB	GRYG	GRYR	GRYON	SLIN4	SLIN3	SLIN2	SLIN1	SLIN0	SBIT3	SBIT2	SBIT1	SBIT0	Gray, scroll setting
0B <sup>16</sup>	—	PTD8	PTD7	SEND4	SEND3	SEND2	SEND1	SEND0	—	PTC8	PTC7	SST4	SST3	SST2	SST1	SST0	Scroll, port setting
0C <sup>16</sup>	SECAM	—	INO	TIMBAS	—	—	YON1	—	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	Display frequency setting
0D <sup>16</sup>	SEFLD	—	MPAL	NXP	EQP	HIDE	LEVEL0	INTNON	PALH	—	—	—	DSPONV	DSPON	—	EX	Display control setting
0E <sup>16</sup>	—	—	—	YON0	ALL24	—	—	—	—	LBLACK	LINEB	LINEG	LINER	PHASE2	PHASE1	PHASE0	Color setting
0F <sup>16</sup>	—	—	—	—	—	—	—	ADON	—	—	SEL_PDC	—	—	—	—	—	Slicer control setting
10 <sup>16</sup>	SELSLI	—	—	—	—	—	—	—	SLSLVL	SLI_VP2	SLI_VP1	SLI_VP0	—	—	VPS_SUB	—	Sync separation, slice setting
11 <sup>16</sup>	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
12 <sup>16</sup>	SEL_VPSH	—	—	—	—	—	—	—	—	—	SEK15	SEK14	SEK13	SEK12	SEK11	SEK10	Slice setting
13 <sup>16</sup>	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
14 <sup>16</sup>	—	—	—	—	—	—	—	—	—	INI	—	—	—	—	—	—	Display setting
15 <sup>16</sup>	—	—	STBY1	—	—	—	—	—	—	PDC_VCO_ON	—	—	XTAL_VCO	—	—	CK_VCO	Oscillation ON/OFF setting
16 <sup>16</sup>	RGBWH	—	—	—	—	PD2	PD1	—	PDC_HP10	PDC_HP9	PDC_HP8	PDC_HP7	PDC_HP6	PDC_HP5	PDC_HP4	PDC_HP3	PDC slice position setting
17 <sup>16</sup>	HGSL	HGSL5	—	CCD	—	—	—	—	VPS_HP10	VPS_HP9	VPS_HP8	VPS_HP7	VPS_HP6	VPS_HP5	VPS_HP4	VPS_HP3	VPS slice position setting
18 <sup>16</sup>	VBI15	VBI14	VBI13	VBI12	VBI11	VBI10	VBI9	VBI8	VBI7	VBI6	VBI5	VBI4	VBI3	VBI2	VBI1	VBI0	VBI encode setting
19 <sup>16</sup>	—	VBI17	VBI16	VPS_LINE4	VPS_LINE3	VPS_LINE2	VPS_LINE1	VPS_LINE0	ENC2	ENC1	VBI2	VBI1	VPSF2	VPSF1	PDCF2	PDCF1	Slice setting
1A <sup>16</sup>	VPS_FL07	VPS_FL08	VPS_FL05	VPS_FL04	VPS_FL03	VPS_FL02	VPS_FL01	VPS_FL00	PDC_FL07	PDC_FL06	PDC_FL05	PDC_FL04	PDC_FL03	PDC_FL02	PDC_FL01	PDC_FL00	PDC, VPS flaring setting
1B <sup>16</sup>	—	—	CHK_VPS5	—	—	—	—	—	—	—	CHK_PDC5	—	—	—	—	—	—
1C <sup>16</sup>	—	—	—	SELPEEK	—	DIV_PDC7	DIV_PDC6	DIV_PDC5	DIV_PDC4	DIV_PDC3	DIV_PDC2	DIV_PDC1	DIV_PDC0	DIV_PDCS2	DIV_PDCS1	DIV_PDCS0	PDC frequency setting
1D <sup>16</sup>	—	—	—	—	DIV_VPS8	DIV_VPS7	DIV_VPS6	DIV_VPS5	DIV_VPS4	DIV_VPS3	DIV_VPS2	DIV_VPS1	DIV_VPS0	DIV_VPS22	DIV_VPS21	DIV_VPS20	VPS frequency setting
1E <sup>16</sup>	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
1F <sup>16</sup>	—	—	—	—	—	—	—	—	MACRON	—	—	FLD	—	—	—	—	Macro, field flag
20 <sup>16</sup>	—	—	—	—	MIN3	MIN2	MIN1	MIN0	—	—	MAX5	MAX4	MAX3	MAX2	MAX1	MAX0	Slice setting
21 <sup>16</sup>	—	—	—	—	—	—	—	—	—	DBL_HEIGHT	—	—	—	—	—	—	—
22 <sup>16</sup>	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

For accessing to expansion register data, set accessing address (DA5 to DA0) (shown in Table 2.15.8) to expansion register address control register (address 0216<sub>16</sub>). Then write data (DD15 to DD0) by expansion register data control register (address 0218<sub>16</sub>). When end the data accessing, expansion register address control register increments address automatically. Then, next address data writing is possible.

Expansion register access registers are shown in Figure 2.15.27, expansion register access block diagram is shown in Figure 2.15.28, and expansion register bit compositions are shown in p172 to p197.

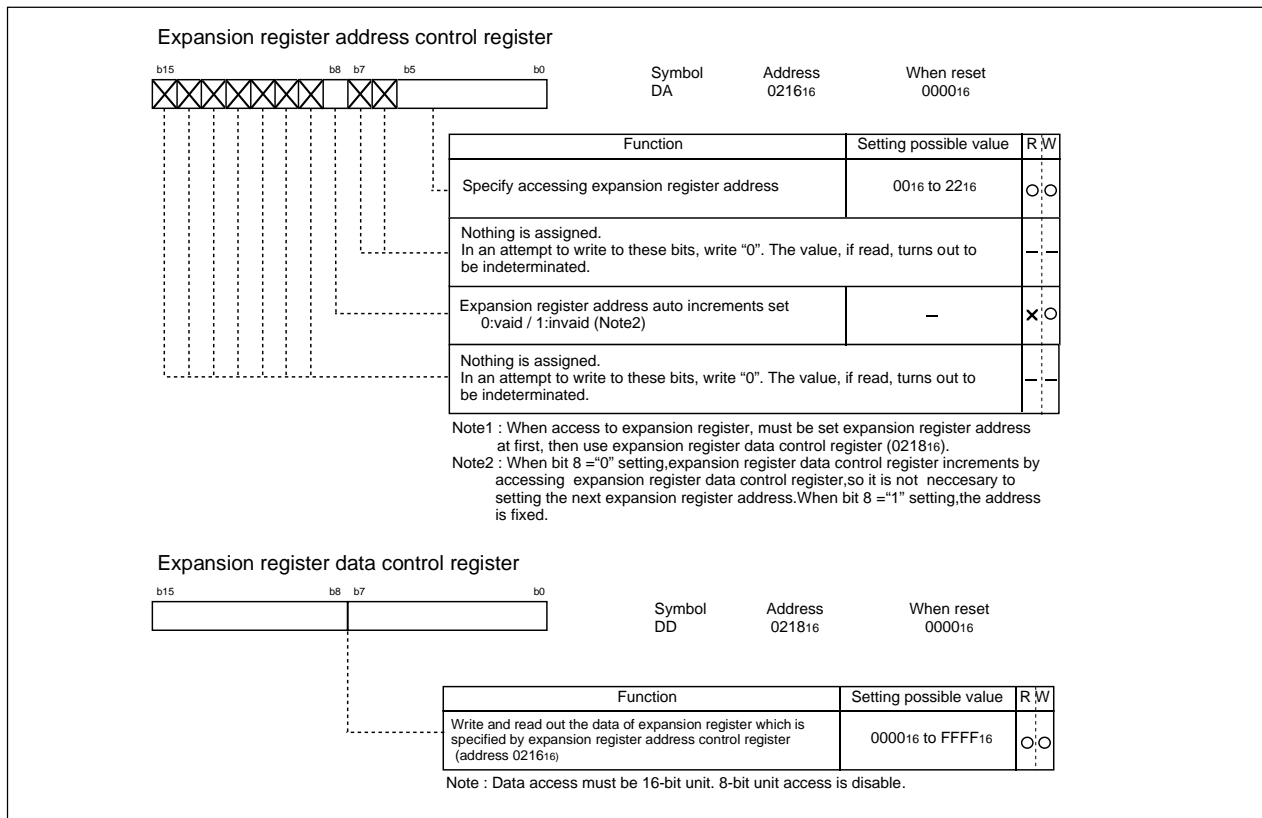


Figure 2.15.27 Expansion register access registers composition

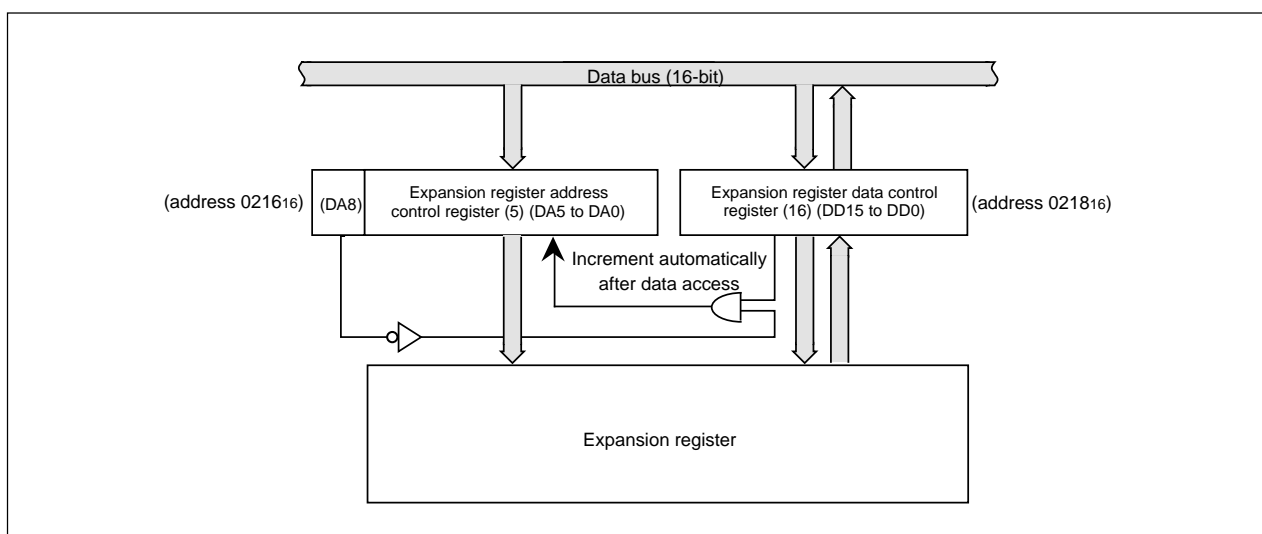


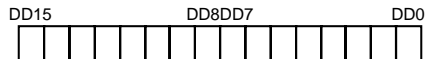
Figure 2.15.28 Expansion register access block diagram

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Expansion register construction

(1) Address 00<sub>16</sub> (= DA5 to 0)

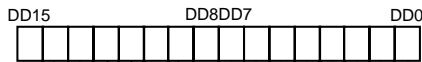


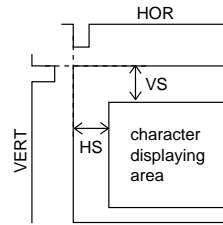
Bit symbol	Bit name	Function	R	W
VP0	Vertical display start position selection bit	<p>If VS is the vertical display start position,  <math display="block">VS = H \times \sum_{n=0}^7 2^n VP_n</math> H: Cycle with the horizontal synchronizing pulse</p> <p>VP7 to VP0 ≤ (00001110<sub>2</sub>) are disable</p>	○	○
VP1				
VP2				
VP3				
VP4				
VP5				
VP6				
VP7				
STBY0	Stand-by mode selection bit	0 Normal mode	○	○
		1 Stand-by mode		
PTC0	Port P11 <sub>0</sub> output selection bit	0 P0 output	○	○
		1 EDO2 output		
PTC1	Port P11 <sub>1</sub> output selection bit	0 P1 output	○	○
		1 EDO1 output		
PTC2	Port P11 <sub>2</sub> output selection bit	0 P2 output	○	○
		1 CSYN output		
PTC3	Port P11 <sub>3</sub> output selection bit	0 P3 output	○	○
		1 BLNK output		
PTC4	Port P11 <sub>4</sub> output selection bit	0 P4 output	○	○
		1 B output		
PTC5	Port P11 <sub>5</sub> output selection bit	0 P5 output	○	○
		1 G output		
PTC6	Port P11 <sub>6</sub> output selection bit	0 P6 output	○	○
		1 R output		

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (2) Address 0116 (= DA5 to 0)

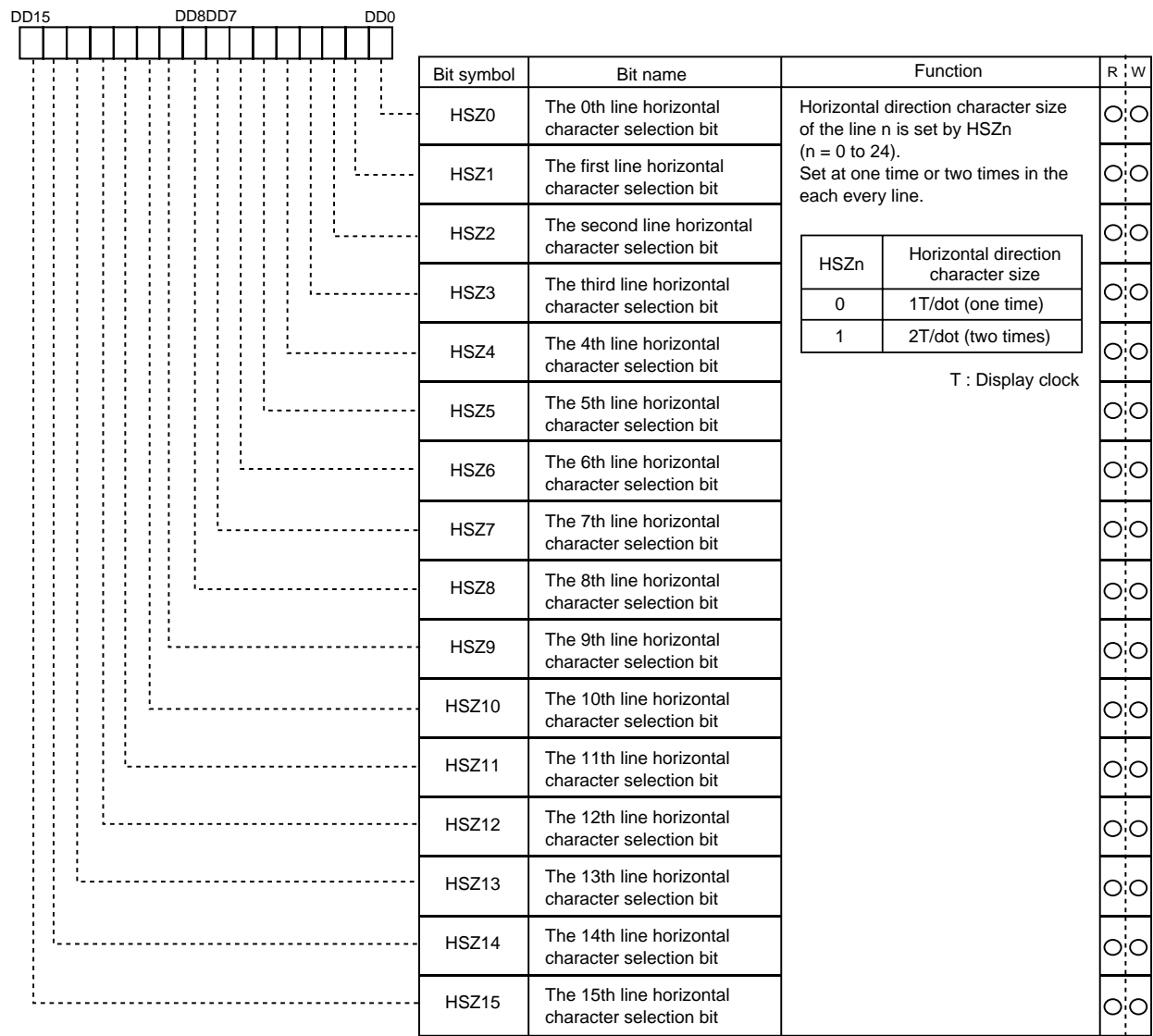


Bit symbol	Bit name	Function	R	W
HP0	Horizontal display start position selection bit	If HS is the vertical display start position, $HS = T1 \times \sum_{n=0}^8 2^n HP_n + 9$ $T1 : \text{Cycle with the display clock}$  <p>HP8 to HP0 ≤ (0000100112) are disable</p>		
HP1				
HP2				
HP3				
HP4				
HP5				
HP6				
HP7				
HP8				
PTD0	Port P110 data selection bit	0 When port output : fixed to L , when EDO2 output : specified negative polarity.		
		1 When port output : fixed to H , when EDO2 output : specified positive polarity.		
PTD1	Port P111 data selection bit	0 When port output : fixed to L , when EDO1 output : specified negative polarity.		
		1 When port output : fixed to H , when EDO1 output : specified positive polarity.		
PTD2	Port P112 data selection bit	0 When port output : fixed to L , when CSYN output : specified negative polarity.		
		1 When port output : fixed to H , when CSYN output : specified positive polarity.		
PTD3	Port P113 data selection bit	0 When port output : fixed to L , when BLNK output : specified negative polarity.		
		1 When port output : fixed to H , when BLNK output : specified positive polarity.		
PTD4	Port P114 data selection bit	0 When port output : fixed to L , when B output : specified negative polarity.		
		1 When port output : fixed to H , when B output : specified positive polarity.		
PTD5	Port P115 data selection bit	0 When port output : fixed to L , when G output : specified negative polarity.		
		1 When port output : fixed to H , when G output : specified positive polarity.		
PTD6	Port P116 data selection bit	0 When port output : fixed to L , when R output : specified negative polarity.		
		1 When port output : fixed to H , when R output : specified positive polarity.		

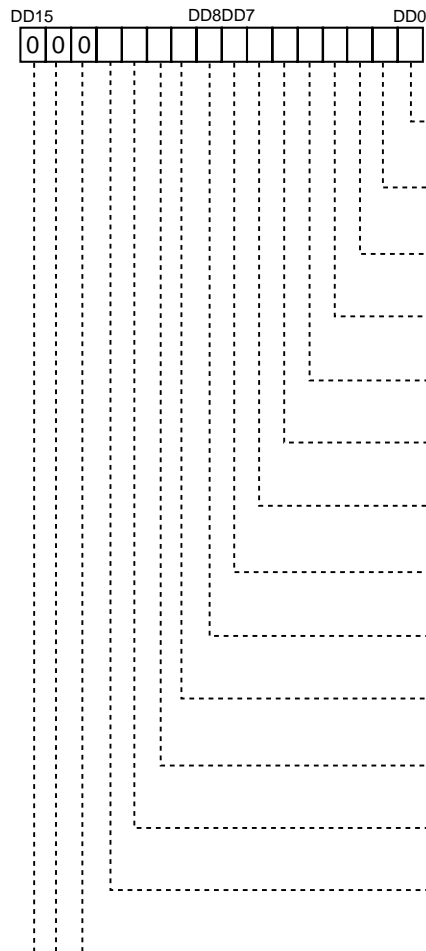
# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

(3) Address 0216 ( = DA5 to 0)



**(4) Address 03<sub>16</sub> ( = DA5 to 0)**

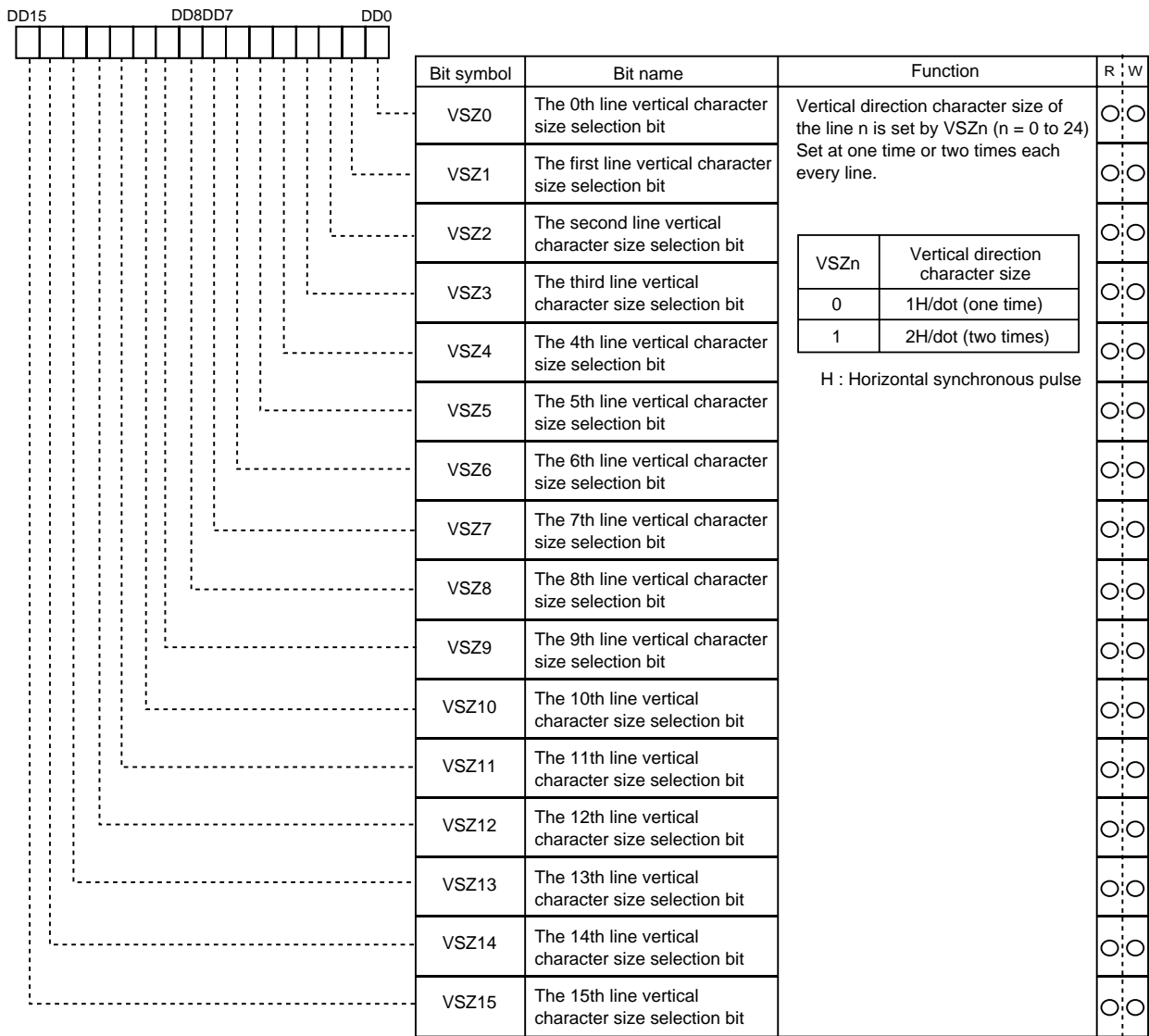


Bit symbol	Bit name	Function		R	W						
HSZ16	The 16th line horizontal character selection bit	<div>Horizontal direction character size of the line n is set by HSZn (n = 0 to 24). Set at one time or two times in the each every line.</div> <table><tr><td>HSZn</td><td>Horizontal direction character size</td></tr><tr><td>0</td><td>1T/dot (one time)</td></tr><tr><td>1</td><td>2T/dot (two times)</td></tr></table> <div>T : Display clock</div>		HSZn	Horizontal direction character size	0	1T/dot (one time)	1	2T/dot (two times)	<input type="radio"/>	<input type="radio"/>
HSZn	Horizontal direction character size										
0	1T/dot (one time)										
1	2T/dot (two times)										
HSZ17	The 17th line horizontal character selection bit			<input type="radio"/>	<input type="radio"/>						
HSZ18	The 18th line horizontal character selection bit			<input type="radio"/>	<input type="radio"/>						
HSZ19	The 19th line horizontal character selection bit			<input type="radio"/>	<input type="radio"/>						
HSZ20	The 20th line horizontal character selection bit			<input type="radio"/>	<input type="radio"/>						
HSZ21	The 21th line horizontal character selection bit			<input type="radio"/>	<input type="radio"/>						
HSZ22	The 22th line horizontal character selection bit			<input type="radio"/>	<input type="radio"/>						
HSZ23	The 23th line horizontal character selection bit	<input type="radio"/>	<input type="radio"/>								
HSZ24	The 24th line horizontal character selection bit	<input type="radio"/>	<input type="radio"/>								
BCOL	All blanking selection bit	0	Blanking of DSP1n and DSP0n	<input type="radio"/>	<input type="radio"/>						
		1	All raster blanking								
TEST0	Test bit	Must always be set to "0".		<input type="radio"/>	<input type="radio"/>						
TEST1				<input type="radio"/>	<input type="radio"/>						
TEST2				<input type="radio"/>	<input type="radio"/>						
Reserved bit		Must always be set to "0".		<input checked="" type="radio"/>	<input type="radio"/>						

M306H1SFP

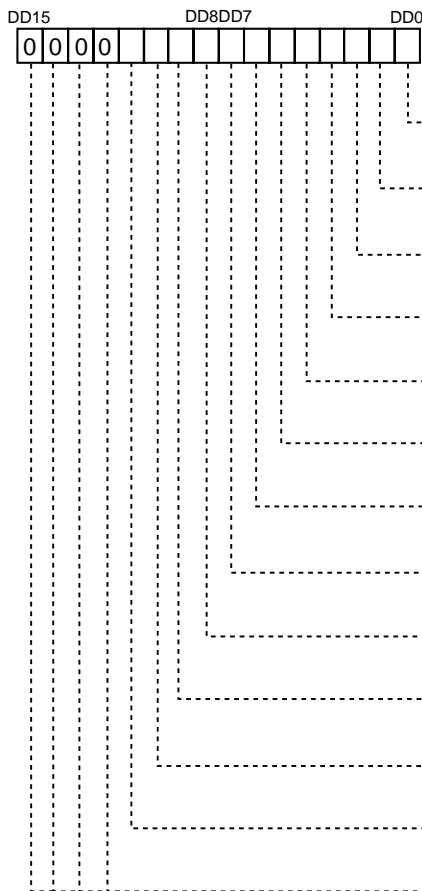
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

(5) Address 0416 ( = DA5 to 0)





**(6) Address 05<sub>16</sub> ( = DA5 to 0)**

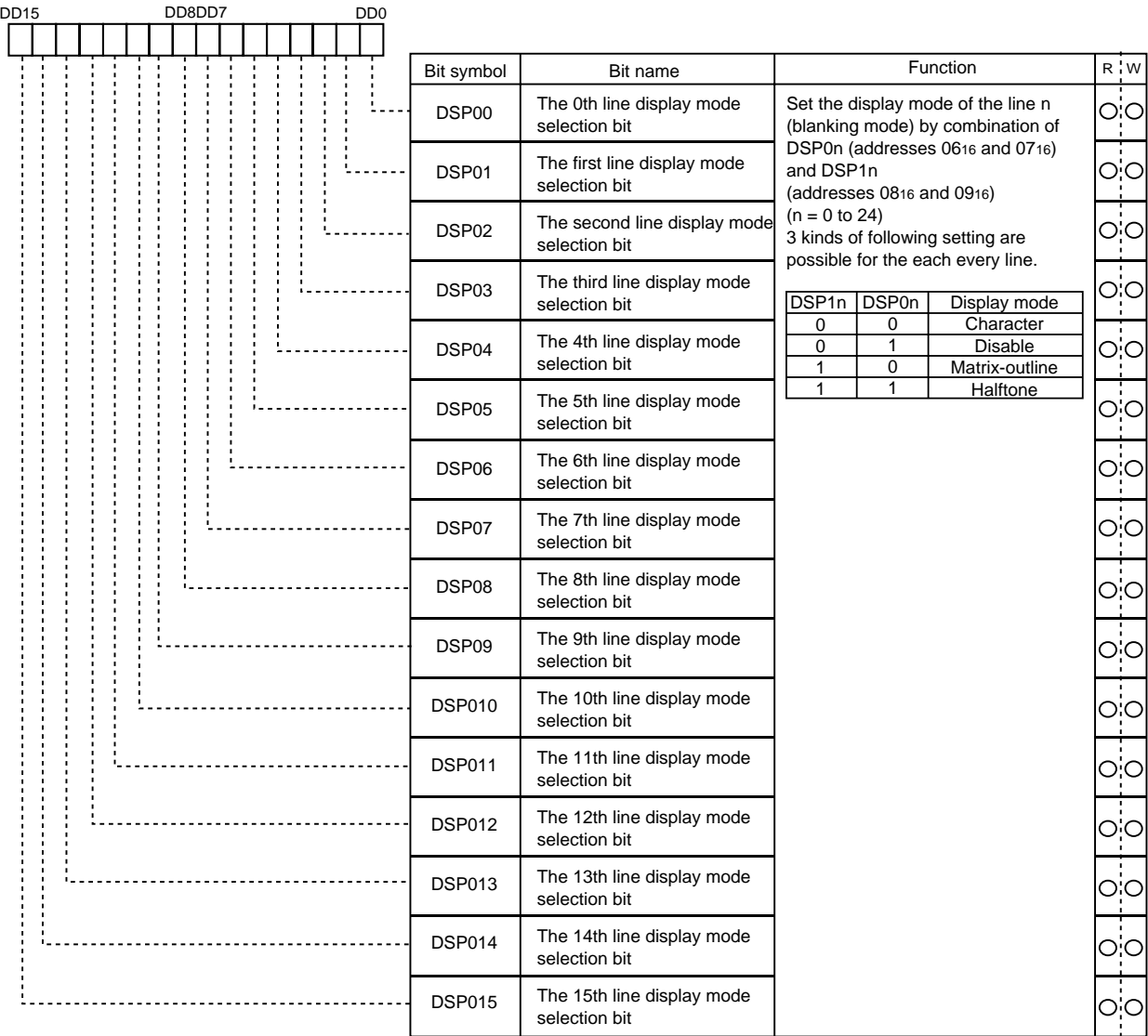


Bit symbol	Bit name	Function	R	W															
VSZ16	The 16th line vertical character size selection bit	<div>Vertical direction character size of the line n is set by VSZn (n = 0 to 24) Set at one time or two times in the each every line.</div> <table><tr><td>VSZn</td><td>Vertical direction character size</td></tr><tr><td>0</td><td>1H/dot (one time)</td></tr><tr><td>1</td><td>2H/dot (two times)</td></tr></table> <div>H : Horizontal synchronous pulse</div>	VSZn	Vertical direction character size	0	1H/dot (one time)	1	2H/dot (two times)	<div>○</div>	<div>○</div>									
VSZn	Vertical direction character size																		
0	1H/dot (one time)																		
1	2H/dot (two times)																		
VSZ17	The 17th line vertical character size selection bit		<div>○</div>	<div>○</div>															
VSZ18	The 18th line vertical character size selection bit		<div>○</div>	<div>○</div>															
VSZ19	The 19th line vertical character size selection bit		<div>○</div>	<div>○</div>															
VSZ20	The 20th line vertical character size selection bit		<div>○</div>	<div>○</div>															
VSZ21	The 21th line vertical character size selection bit		<div>○</div>	<div>○</div>															
VSZ22	The 22th line vertical character size selection bit	<div>○</div>	<div>○</div>																
VSZ23	The 23th line vertical character size selection bit	<div>○</div>	<div>○</div>																
VSZ24	The 24th line vertical character size selection bit	<div>○</div>	<div>○</div>																
BLINK0	Blinking duty selection bit	<table><tr><td>BLINK1</td><td>BLINK0</td><td>DUTY</td></tr><tr><td>0</td><td>0</td><td>Blinking off</td></tr><tr><td>0</td><td>1</td><td>25%</td></tr><tr><td>1</td><td>1</td><td>50%</td></tr><tr><td>1</td><td>0</td><td>75%</td></tr></table>	BLINK1	BLINK0	DUTY	0	0	Blinking off	0	1	25%	1	1	50%	1	0	75%	<div>○</div>	<div>○</div>
BLINK1		BLINK0	DUTY																
0		0	Blinking off																
0		1	25%																
1	1	50%																	
1	0	75%																	
BLINK1	<div>○</div>	<div>○</div>																	
BLINK2	Blinking cycle selection bit	0	Cycle approximatery 1 second.	<div>○</div>	<div>○</div>														
		1	Cycle approximatery 0.5 second.	<div>○</div>	<div>○</div>														
Reserved bit		Must always be set to "0".	<div>×</div>	<div>○</div>															

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

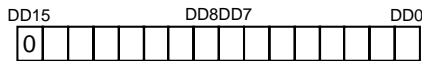
(7) Address 0616 ( = DA5 to 0)

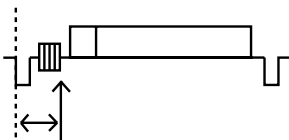


# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (8) Address 07<sub>16</sub> (= DA<sub>5</sub> to 0)

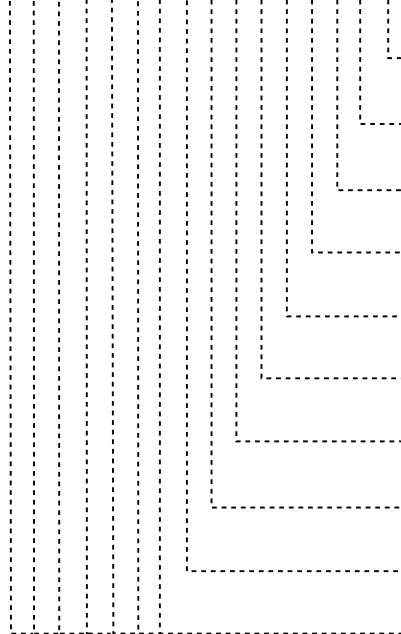


Bit symbol	Bit name	Function	R	W															
DSP016	The 16th line display mode selection bit	<p>Set the display mode of the line n (blanking mode) by combination of DSP0n (addresses 06<sub>16</sub> and 07<sub>16</sub>) and DSP1n (addresses 08<sub>16</sub> and 09<sub>16</sub>) (n = 0 to 24)</p> <p>3 kinds of following setting are possible for the each every line.</p> <table><tr><th>DSP1n</th><th>DSP0n</th><th>Display mode</th></tr><tr><td>0</td><td>0</td><td>Character</td></tr><tr><td>0</td><td>1</td><td>Disable</td></tr><tr><td>1</td><td>0</td><td>Matrix-outline</td></tr><tr><td>1</td><td>1</td><td>Halftone</td></tr></table>	DSP1n	DSP0n	Display mode	0	0	Character	0	1	Disable	1	0	Matrix-outline	1	1	Halftone	○	○
DSP1n	DSP0n		Display mode																
0	0		Character																
0	1		Disable																
1	0		Matrix-outline																
1	1		Halftone																
DSP017	The 17th line display mode selection bit		○	○															
DSP018	The 18th line display mode selection bit		○	○															
DSP019	The 19th line display mode selection bit		○	○															
DSP020	The 20th line display mode selection bit	○	○																
DSP021	The 21th line display mode selection bit	○	○																
DSP022	The 22th line display mode selection bit	○	○																
DSP023	The 23th line display mode selection bit	○	○																
DSP024	The 24th line display mode selection bit	○	○																
CO0	VBI encode horizontal start position selection bit	<p>VBI encode horizontal start position</p>  <p>CO0 to CO5</p> <p>(Each line are set to common)</p>	○	○															
CO1			○	○															
CO2			○	○															
CO3			○	○															
CO4			○	○															
CO5			○	○															
Reserved bit		Must always be set to "0".	x	○															

**(9) Address 08<sub>16</sub> ( = DA5 to 0)**



**(10) Address 09<sub>16</sub> ( = DA5 to 0)**

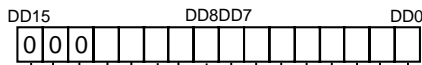


Bit symbol	Bit name	Function	R	W															
DSP116	The 16th line display mode selection bit	Set the display mode of the line n (blanking mode) by combination of DSP0n (addresses 06 <sub>16</sub> and 07 <sub>16</sub> ) and DSP1n (addresses 08 <sub>16</sub> and 09 <sub>16</sub> ) (n = 0 to 24) 3 kinds of following setting are possible for the each every line. <table><tr><th>DSP1n</th><th>DSP0n</th><th>Display mode</th></tr><tr><td>0</td><td>0</td><td>Character</td></tr><tr><td>0</td><td>1</td><td>Disable</td></tr><tr><td>1</td><td>0</td><td>Matrix-outline</td></tr><tr><td>1</td><td>1</td><td>Halftone</td></tr></table>	DSP1n	DSP0n	Display mode	0	0	Character	0	1	Disable	1	0	Matrix-outline	1	1	Halftone	<input type="radio"/>	<input type="radio"/>
DSP1n	DSP0n		Display mode																
0	0		Character																
0	1		Disable																
1	0		Matrix-outline																
1	1		Halftone																
DSP117	The 17th line display mode selection bit		<input type="radio"/>	<input type="radio"/>															
DSP118	The 18th line display mode selection bit		<input type="radio"/>	<input type="radio"/>															
DSP119	The 19th line display mode selection bit		<input type="radio"/>	<input type="radio"/>															
DSP120	The 20th line display mode selection bit		<input type="radio"/>	<input type="radio"/>															
DSP121	The 21th line display mode selection bit	<input type="radio"/>	<input type="radio"/>																
DSP122	The 22th line display mode selection bit	<input type="radio"/>	<input type="radio"/>																
DSP123	The 23th line display mode selection bit	<input type="radio"/>	<input type="radio"/>																
DSP124	The 24th line display mode selection bit	<input type="radio"/>	<input type="radio"/>																
Reserved bit		Must always be set to "0".	<input checked="" type="checkbox"/>	<input type="radio"/>															

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

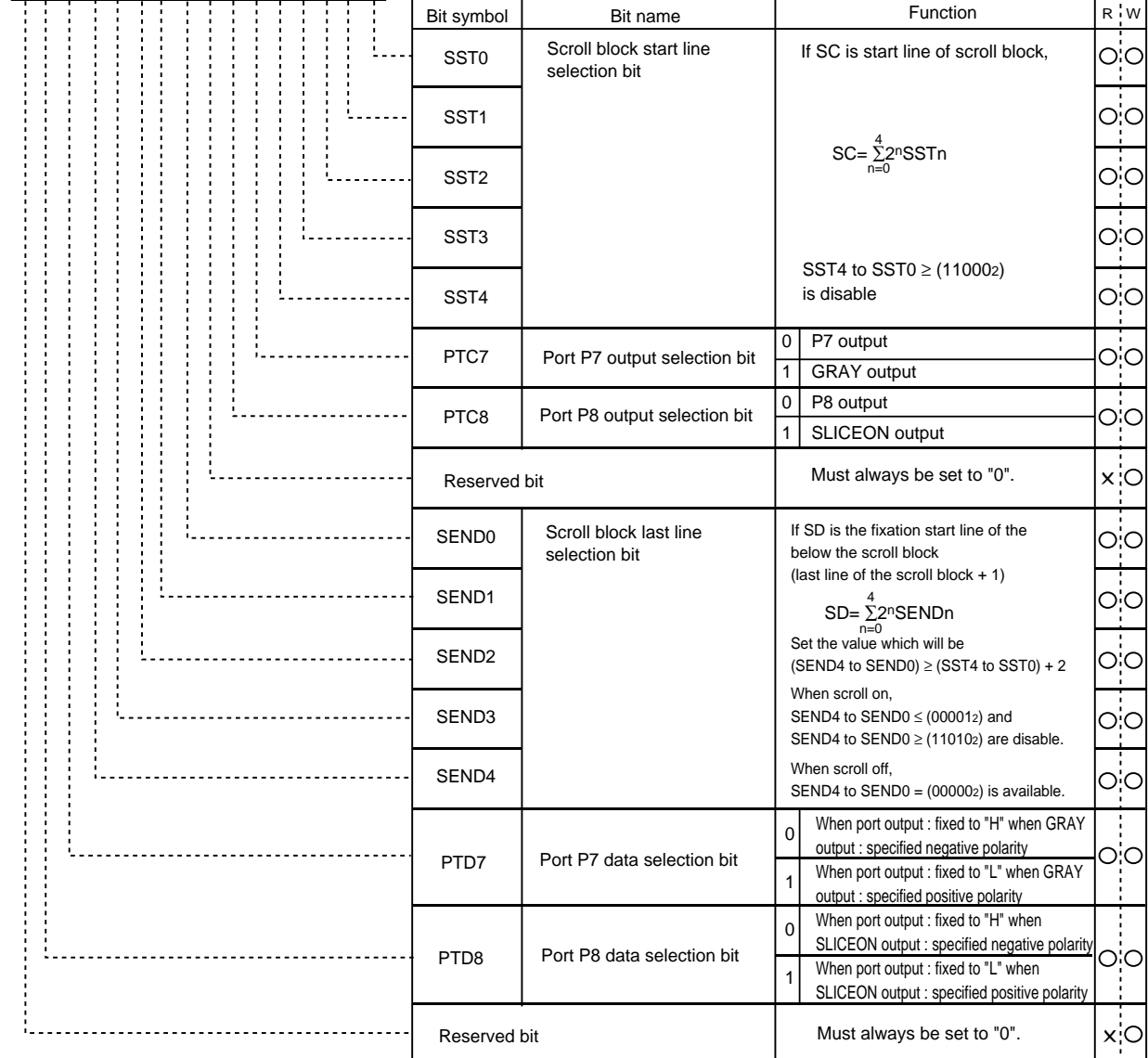
## (11) Address 0A<sub>16</sub> (= DA5 to 0)



Bit symbol	Bit name	Function	R	W																																				
SBIT0	Scroll display start dot selection bit	If SA is display start dot of scroll block,  $SA = \sum_{n=0}^3 2^n SBITn$  SBIT3 to SBIT0 ≥ (1010 <sub>2</sub> ) is disable	○	○																																				
SBIT1																																								
SBIT2																																								
SBIT3																																								
SLIN0	Scroll display start dot selection bit	If SB is display start dot of scroll block,  $SB = \sum_{n=0}^4 2^n SBITn$  SLIN4 to SLIN0 ≥ (11001 <sub>2</sub> ) is disable. Set the value which is satisfies with shown below : SST4 to SST0 ≤ SLIN4 to SLIN0 < SEND4 to SEND0	○	○																																				
SLIN1																																								
SLIN2																																								
SLIN3																																								
SLIN4																																								
GRYON	Gray display selection bit	0	○	○																																				
		1																																						
GRYR	Gray display color selection bit	<table><tr><th>GRYB</th><th>GRYG</th><th>GRYR</th><th>Color</th></tr><tr><td>0</td><td>0</td><td>0</td><td>Black</td></tr><tr><td>0</td><td>0</td><td>1</td><td>Red</td></tr><tr><td>0</td><td>1</td><td>0</td><td>Green</td></tr><tr><td>0</td><td>1</td><td>1</td><td>Yellow</td></tr><tr><td>1</td><td>0</td><td>0</td><td>Blue</td></tr><tr><td>1</td><td>0</td><td>1</td><td>Magenta</td></tr><tr><td>1</td><td>1</td><td>0</td><td>Cyan</td></tr><tr><td>1</td><td>1</td><td>1</td><td>White</td></tr></table>	GRYB	GRYG	GRYR	Color	0	0	0	Black	0	0	1	Red	0	1	0	Green	0	1	1	Yellow	1	0	0	Blue	1	0	1	Magenta	1	1	0	Cyan	1	1	1	White	○	○
GRYB		GRYG	GRYR	Color																																				
0		0	0	Black																																				
0		0	1	Red																																				
0		1	0	Green																																				
0		1	1	Yellow																																				
1		0	0	Blue																																				
1		0	1	Magenta																																				
1		1	0	Cyan																																				
1		1	1	White																																				
GRYG																																								
GRYB	Gray color is set by this register Valid only ot GRYON = "1"	○	○																																					
Reserved bit				Must always be set to "0".	x	○																																		

Note 1. Refer to register RGBWH (Address 16<sub>16</sub>) about RGB output.

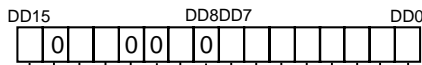
**(12) Address 0B<sub>16</sub> ( = DA5 to 0)**



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (13) Address 0C<sub>16</sub> (= DA5 to 0)



Bit symbol	Bit name	Function		R	W
PC0	Display frequency selection bit	Control display frequency $f_r$ ,		○	○
PC1		$f_r = f_H \times \left\{ \sum_{n=0}^7 PC_n + 512 \right\}$		○	○
PC2		$f_H$ : Horizontal synchronous signal frequency		○	○
PC3		PC7 to PC0 ≤ (011111112) is disable.		○	○
PC4				○	○
PC5				○	○
PC6		Set PC7 to PC0 = (111101012), normally.		○	○
PC7				○	○
Reserved bit		Must always be set to "0".		x	○
YON1	Color burst at internal synchronous selection bit (Note)	0	Color burst ON	○	○
		1	Color burst OFF		
Reserved bit		Must always be set to "0".		x	○
TIMBAS	Time base selection bit	0	Time base ON	○	○
		1	Time base OFF		
IN0	Internal synchronous selection bit	0	External synchronous setting	○	○
		1	Internal synchronous setting		
Reserved bit		Must always be set to "0".		x	○
SECAM	Combination selection bit from SECAMIN pin	0	Do not superimpose the carrier from SECAMIN pin.	○	○
		1	Superimpose the carrier from SECAMIN pin.		

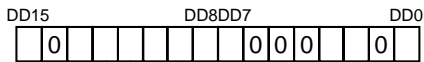
Note1. When moto-tone display (YON0(address 0E<sub>16</sub>)= "1") setting, must be set to "1".



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (14) Address 0D<sub>16</sub> (= DA<sub>5</sub> to 0)



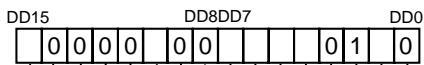
Bit symbol	Bit name	Function			R	W															
EX	External/internal synchronous selection bit	0	External synchronization		○	○															
		1	Internal synchronization																		
Reserved bit		Must always be set to "0".			x	○															
DSPON	Digital display selection bit	0	Digital output display OFF.		○	○															
		1	Digital outoput display ON.																		
DSPONV	Analog display selection bit	0	Composite video signal output display OFF.		○	○															
		1	Composite video signal output display ON.																		
Reserved bit		Must always be set to "0".			x	○															
PALH	Number of scanning line selection bit	<table><tr><td>PALH</td><td>INT/NON</td><td>Number of scanning line</td></tr><tr><td>0</td><td>0</td><td>625H</td></tr><tr><td>0</td><td>1</td><td>626H</td></tr><tr><td>1</td><td>0</td><td>624H</td></tr><tr><td>1</td><td>1</td><td>628H</td></tr></table>			PALH	INT/NON	Number of scanning line	0	0	625H	0	1	626H	1	0	624H	1	1	628H	○	○
PALH		INT/NON	Number of scanning line																		
0		0	625H																		
0		1	626H																		
1		0	624H																		
1	1	628H																			
INTNON				○	○																
LEVEL0	Video signal generation selection bit	0	Composite video signal generation circuit OFF.		○	○															
		1	Composite video signal generation circuit ON.																		
HIDE	SYRAM expansion display selection bit	0	SYRAM writing over		○	○															
		1	SYRAM writing over or character erasing																		
EQP	Equivalent pulse selection bit	0	Do not include equivalent pulse.		○	○															
		1	Includes equivalent pulse.																		
NXP	Broadcast method selection bit	<table><tr><td>N/P</td><td>MPAL</td><td>Broadcasting method</td></tr><tr><td>0</td><td>0</td><td>NTSC</td></tr><tr><td>0</td><td>1</td><td>M-PAL</td></tr><tr><td>1</td><td>0</td><td>PAL</td></tr><tr><td>1</td><td>1</td><td>Disable</td></tr></table>			N/P	MPAL	Broadcasting method	0	0	NTSC	0	1	M-PAL	1	0	PAL	1	1	Disable	○	○
N/P		MPAL	Broadcasting method																		
0		0	NTSC																		
0		1	M-PAL																		
1		0	PAL																		
1	1	Disable																			
MPAL				○	○																
Reserved bit		Must always be set to "0".			x	○															
SELFLD	Field at non interlace selection bit	0	The second field.		○	○															
		1	The first field.																		



# M306H1SFP

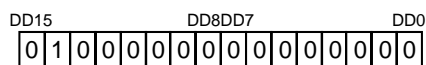
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**(17) Address<sub>1016</sub> ( = DA5 to 0)**



Bit symbol	Bit name	Function		R	W
Reserved bit		Must always be set to "0".		x	0
VPS_SUB	Flaming code check selection bit for VPS data.	0	Later 8bits of flaming code 16bits		0
		1	Former 4bits and later 4bits of flaming code 16bits (Select 8bits which is set in VPS_FLC0 to 7)		0
Reserved bit		Must always be set to "1".		x	0
Reserved bit		Must always be set to "0".		x	0
SLI_VP0	Slice start line selection bit (Field 1 and 2 are common) Stores data for 18 lines from the 6th line, normally. (SLI_VP2 to SLI_VPO = "316" fixed)	If the slice start line is SLI_VS, $\text{<Field 1> SLI\_VS} = \sum_{n=0}^2 2^n \text{SLI\_VPn} + 3$ $\text{<Field 2> SLI\_VS} = \sum_{n=0}^2 2^n \text{SLI\_VPn} + 315$ Stores data for 18 lines from line which is set by this register to slice RAM.			0
SLI_VP1					0
SLI_VP2					0
SLSLVL	Slice level control bit	0	Auto level for data slice		0
		1	Fix level for data slice		0
Reserved bit		Must always be set to "0".		x	0
SYNCSEP_ON0	Synchronous separation control bit	0	Sync-sep circuit OFF		0
		1	Sync-sep circuit ON		0
Reserved bit		Must always be set to "0".		x	0
SELSLI	Slice signal input pin selection bit	0	CVIN1 pin		0
		1	CVIN2 pin		0

**(18) Address 11<sub>16</sub>( = DA5 to 0)**



Bit symbol	Bit name	Function	R	W
	Reserved bit	Must always be set to "0".	x	0
	Reserved bit	Must always be set to "1".	x	0
	Reserved bit	Must always be set to "0".	x	0

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

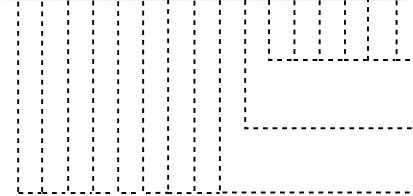
(19) Address 1216 ( = DA5 to 0)

15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
----	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

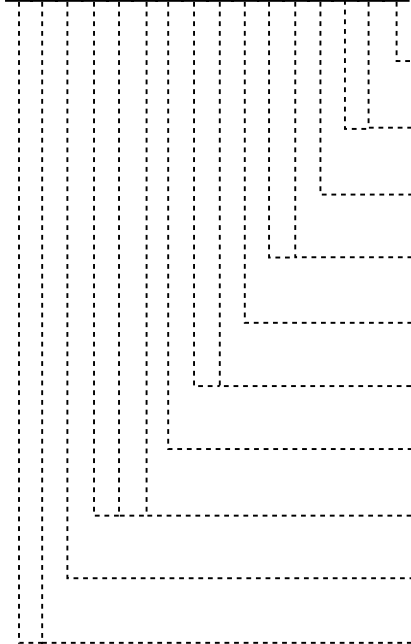
(20) Address 1316 ( = DA5 to 0)

DD15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	DD0
DD8DD7																
Reserved bit				Must always be set to "0".												x : ○

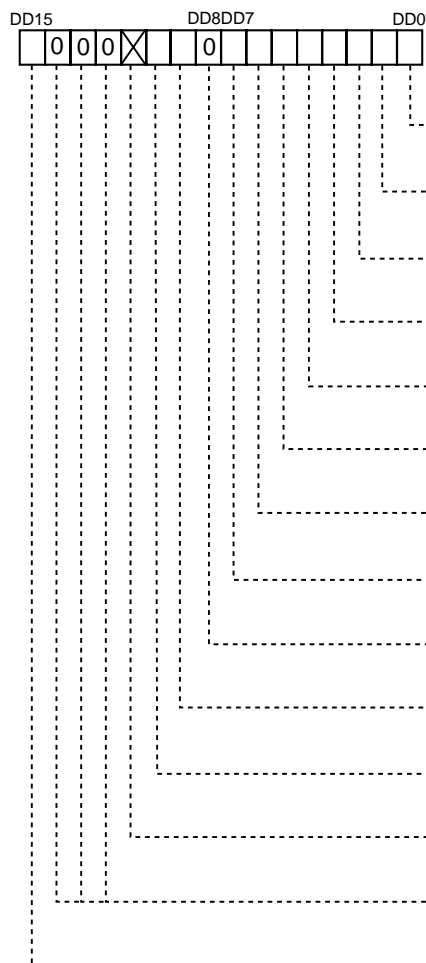
**(21) Address 14<sub>16</sub> ( = DA5 to 0)**

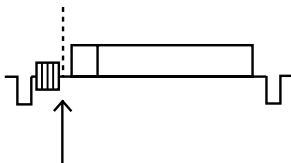


**(22) Address 15<sub>16</sub> (= DA5 to 0)**



**(23) Address 16<sub>16</sub> ( = DA5 to 0)**

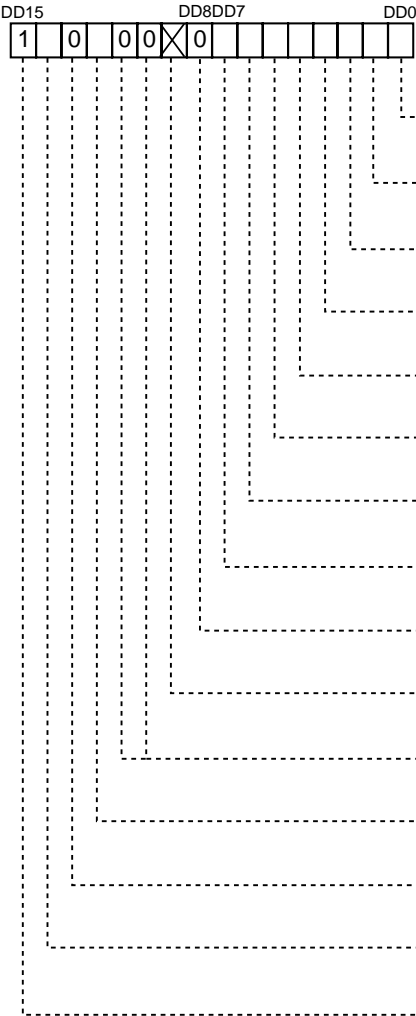


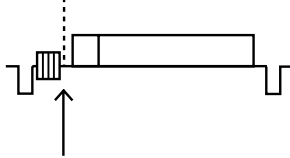
Bit symbol	Bit name	Function	R	W
PDC_HP3	PDC slice check start position selection bit	If the PDC slice check start position is PDC_HS, $PDC\_HS = T3 \times \sum_{n=3}^{10} 2^{(n-3)} PDC\_HPn$  T3 : PDC clock run-in cycle +2    Set to flaming code check start position  Set by the 144ns (1bit)	○	○
PDC_HP4			○	○
PDC_HP5			○	○
PDC_HP6			○	○
PDC_HP7			○	○
PDC_HP8			○	○
PDC_HP9			○	○
PDC_HP10			○	○
Reserved bit			Must always be set to "0".	x
PD1	PDC, VPS, VBI clock phase control bit	Adjust clock phase for Data slicer. Normally, PD2 to PD1=(10) <sub>2</sub> fixed.	○	○
PD2			○	○
Nothing is assigned.			x	x
Reserved bit		Must always be set to "0".	x	○
RGBWH	RGB out put (gray display) selection bit	0	Normal	
		1	RGB output of gray display color is white.	

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

(24) Address 17<sub>16</sub> (= DA5 to 0)

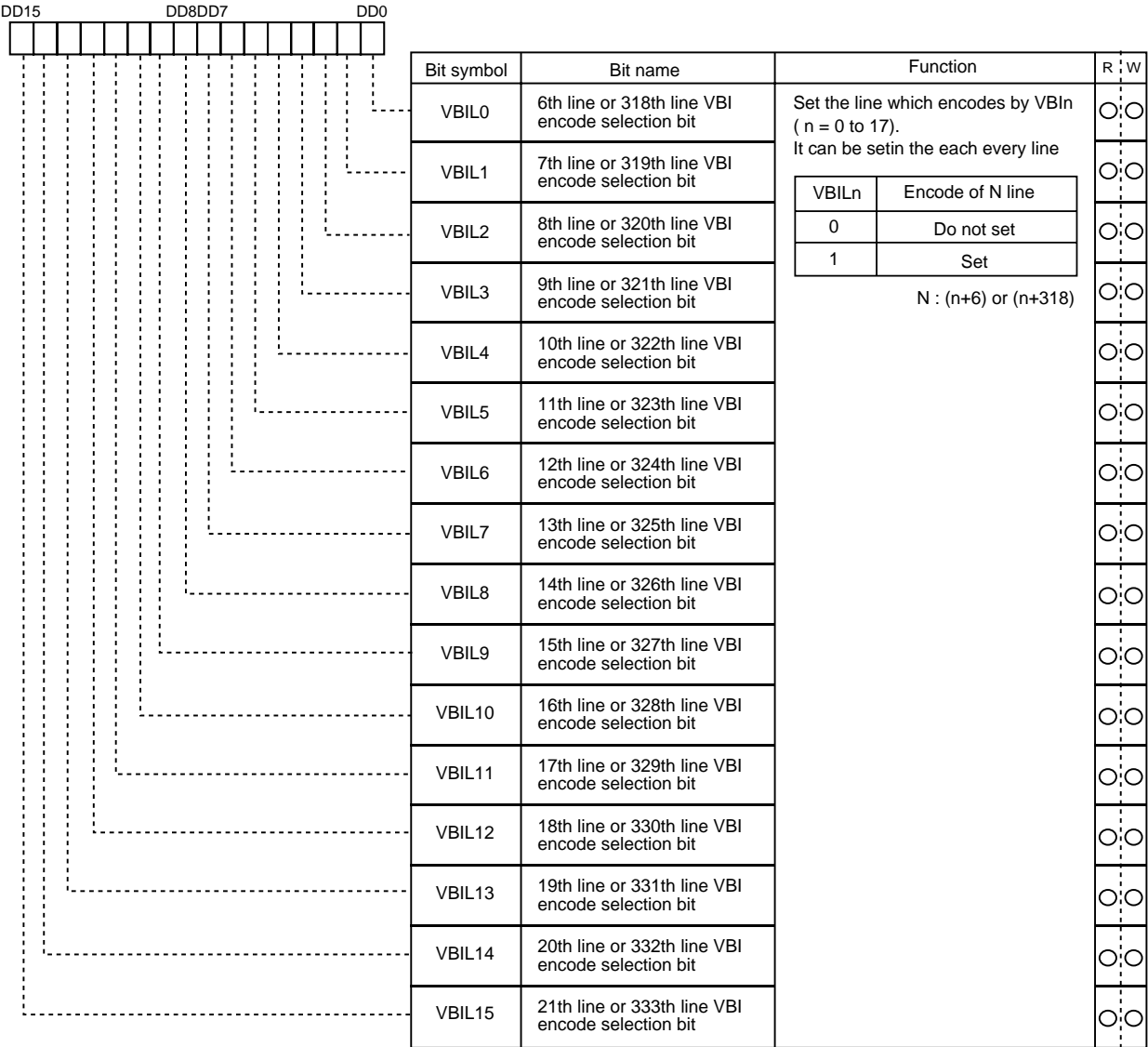


Bit symbol	Bit name		Function	R	W
VPS_HP3	VPS and VBI slice check start position selection bit		If VPS and VBI slice check start position is VPS_HS, $VPS\_HS= T2 \times \sum_{n=3}^{10} 2^{(n-3)} VPS\_HPn$ T2 : VPS or VBI clock run-in cycle +2  Set to flaming code check start position Set by the 200ns (1bit)....VPS Set by the 800ns (1bit)....VBI	○	○
VPS_HP4				○	○
VPS_HP5				○	○
VPS_HP6				○	○
VPS_HP7				○	○
VPS_HP8				○	○
VPS_HP9				○	○
VPS_HP10				○	○
Reserved bit			Must always be set to "0".	x	○
Nothing is assigned.				x	x
Reserved bit			Must always be set to "0".	x	○
CCD	CCD slicer selection bit	0	PDC, VPS, VBI	○	○
		1	CCD		
Reserved bit			Must always be set to "0".	x	○
HGSLs	Data slicer control bit	0	PDC, VPS	○	○
		1	VBI		
HGSL	Data slicer control bit	Must always be set to "1".		○	○

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

(25) Address 1816 ( = DA5 to 0)

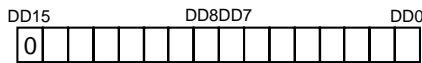




# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (26) Address 1916 (= DA5 to 0)



Bit symbol	Bit name	Function	R	W
PDCF1	PDC data slice selection bit (field1)	0 Do not slice field 1 PDC data	○	○
		1 Slice field 1 PDC data		
PDCF2	PDC data slice selection bit (field2)	0 Do not slice field 2 PDC data	○	○
		1 Slice field 2 PDC data		
VPSF1	VPS data slice selection bit (field1)	0 Do not slice field 1 VPS data	○	○
		1 Slice field 1 VPS data		
VPSF2	VPS data slice selection bit (field2)	0 Do not slice field 2 VPS data	○	○
		1 Slice field 2 VPS data		
VBIF1	VBI data slice selection bit (field1)	0 Do not slice field 1 VBI data	○	○
		1 Slice field 1 VBI data		
VBIF2	VBI data slice selection bit (field2)	0 Do not slice field 2 VBI data	○	○
		1 Slice field 2 VBI data		
ENCF1	VBI data encode selection bit (field1)	0 Do not slice field 1 VBI data	○	○
		1 Slice field 1 VBI data		
ENCF2	VBI data encode selection bit (field2)	0 Do not slice field 2 VBI data	○	○
		1 Slice field 2 VBI data		
VPSF_LINE0	VPS data slice line selection bit	When VPS data slice line is VPS_LINES, $VPS\_LINES = \sum_{n=0}^4 2^n VPS\_LINE_n + 7$ Fix to 16th line normally.  (VPS_LINE4 to VPS_LINE0 = "010012" fixed) Setting value from 000002 to 100002 (7th line to 23 line)	○	○
VPSF_LINE1			○	○
VPSF_LINE2			○	○
VPSF_LINE3			○	○
VPSF_LINE4			○	○
VBIL16	22th line or 334th line VBI encode selection bit	Set encode line by VBILn (n = 0 to 17) Refer to address 1816	○	○
VBIL17	23th line or 335th line VBI encode selection bit		○	○
Reserved bit		Must always be set to "0".	×	○

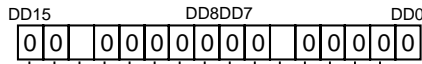
**(27) Addrres 1A<sub>16</sub> ( = DA5 to 0)**



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (28) Address 1B<sub>16</sub> (= DA<sub>5</sub> to 0)



Bit symbol	Bit name	Function	R	W
	Reserved bit	Must always be set to "0".	x	0
CHK_PDC5	Flaming code check selection bit	0 PDC_FLC5 valid	0	0
		1 PDC_FLC5 invalid (Note1)		
	Reserved bit	Must always be set to "0".	x	0
CHK_VPS5	Flaming code check selection bit	0 VPS_FLC5 valid	0	0
		1 VPS_FLC5 invalid (Note1)		
	Reserved bit	Must always be set to "0".	x	0

Note1. At VBI slice, must be set to "1".

## (29) Address 1C<sub>16</sub> (= DA<sub>5</sub> to 0)

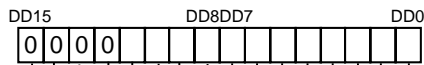


Bit symbol	Bit name	Function	R	W
DIV_PDCS0	PLL control bit for PDC	Control the slice clock frequency f <sub>PDC</sub> for PDC. $f_{PDC} = f_H \times \left( \sum_{n=0}^8 2^n \text{DIV\_VPS}_n + \sum_{m=0}^3 2^{m-3} \text{DIV\_PDCS}_m \right)$	0	0
DIV_PDCS1				
DIV_PDCS2				
DIV_PDC0	PLL divided value selection bit for PDC	f <sub>H</sub> : Horizontal synchronized signal frequency  When SEL_PDCH (address 0F <sub>16</sub> ) = "0", DIV_PDC8 to DIV_PDC0 = (110111011) <sub>2</sub> DIV_PDC2 to DIV_PDC0 = (110) <sub>2</sub>  When SEL_PDCH = "1" DIV_PDC8 to DIV_PDC0 = (000010010) <sub>2</sub> DIV_PDC2 to DIV_PDC0 = (101) <sub>2</sub>	0	0
DIV_PDC1				
DIV_PDC2				
DIV_PDC3				
DIV_PDC4				
DIV_PDC5				
DIV_PDC6				
DIV_PDC7				
DIV_PDC8				
SELPEEK	Peek point detect selection bit	0 Detect from A/D data	0	0
		1 Detect from data of digital calculation after normally "1" setting.		
	Reserved bit	Must always be set to "0".	x	0

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (30) Address 1D<sub>16</sub> (= DA<sub>5</sub> to 0)



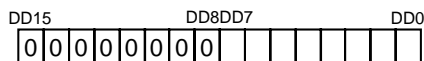
Bit symbol	Bit name	Function	R	W
DIV_VPSS0	PLL control bit for VPS and VBI	Control the slice clock frequency f <sub>VPS</sub> for VPS and VBI. $f_{PDC} = f_H \times \left( \sum_{n=0}^8 2^n \text{DIV\_VPSn} + \sum_{m=0}^2 2^{m-3} \text{DIV\_VPSSm} \right)$	○	○
DIV_VPSS1			○	○
DIV_VPSS2			○	○
DIV_VPS0	PLL divided value selection bit for VPS and VBI	f <sub>H</sub> : Horizontal synchronized signal frequency  When SEL_VPSH (address 12 <sub>16</sub> ) = "0", DIV_VPS8 to DIV_VPS0 = (10011111) <sub>2</sub> DIV_VPSS2 to DIV_VPSS0 = (110) <sub>2</sub>  When SEL_VPSH = "1", DIV_VPS8 to DIV_VPS0 = (00001111) <sub>2</sub> DIV_VPSS2 to DIV_VPSS0 = (110) <sub>2</sub>	○	○
DIV_VPS1			○	○
DIV_VPS2			○	○
DIV_VPS3			○	○
DIV_VPS4			○	○
DIV_VPS5			○	○
DIV_VPS6			○	○
DIV_VPS7			○	○
DIV_VPS8			○	○
Reserved bit			Must always be set to "0".	x

## (31) Address 1E<sub>16</sub> (= DA<sub>5</sub> to 0)



Bit symbol	Bit name	Function	R	W
Reserved bit		Writing is disable. Reading exclusive bit.	x	x

## (32) Address 1F<sub>16</sub> (= DA<sub>5</sub> to 0)

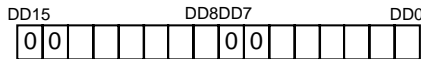


Bit symbol	Bit name	Function		R	W
Reserved bit		Writing is disable. Reading exclusive bit.		X	X
FLD	Fild flag	0	The second field.	○	X
		1	The first field.		
Reserved bit		Writing is disable. Reading exclusive bit.		X	X
MACRON	Macro vision flag	0	No macro vision.	○	X
		1	Macro vision		
Reserved bit		Writing is disable. Reading exclusive bit.		X	X

# M306H1SFP

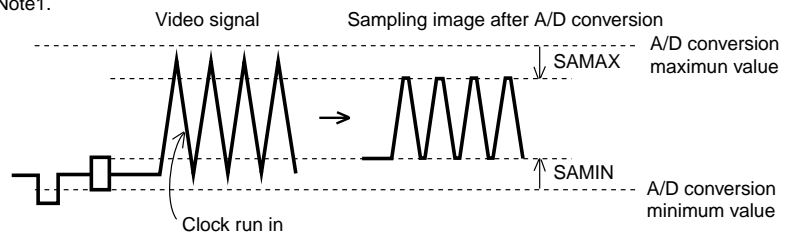
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (33) Address 2016 (= DA5 to 0)

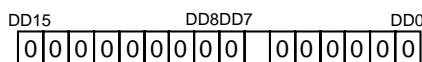


Bit symbol	Bit name	Function	R	W
MAX0	Slice data sampling maximum value selection bit	Set slice data sampling maximum value after A/D conversion. $SAMAX = \sum_{n=0}^5 2^n \times MAXn$ (Note1)		
MAX1				
MAX2				
MAX3				
MAX4				
MAX5				
Reserved bit		Must always be set to "0".	X	0
MIN0	Slice data sampling minimum value selection bit	Set slice data sampling minimum value after A/D conversion. $SAMIN = \sum_{n=0}^5 2^n \times MINn$ (Note1)		
MIN1				
MIN2				
MIN3				
MIN4				
MIN5				
Reserved bit		Must always be set to "0".	X	0

Note1.

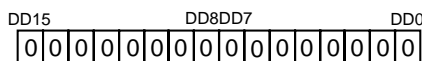


## (34) Address 2116 (= DA5 to 0)



Bit symbol	Bit name	Function	R	W
Reserved bit		Must always be set to "0".	×	○
DBL_HEIGHT	Double height display selection bit	0 Display next line, when vertical direction character size is two times. 1 Do not display next line, when vertical direction character size is two times.	○	○
Reserved bit		Must always be set to "0".	×	○

## (35) Address 2216 (= DA5 to 0)



Bit symbol	Bit name	Function	R	W
Reserved bit		Must always be set to "0".	×	○

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.15.9 Expansion Register Construction Composition

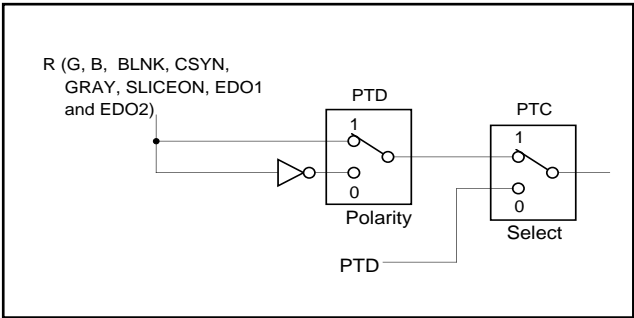


Figure 2.15.29 Switching of port output, R, G and B output

Table 2.15.9 Video signal level

Color name	Phase (rad)	Luminance level (V) (Note1)			Chroma level (mV) (Note1)			Chroma amplitude (Notes 1 and 2)		
		Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.
Sync-chip	—	0.90	1.00	1.10	—	—	—	—	—	—
Pedestal	—	1.50	1.60	1.70	—	—	—	—	—	—
Color burst	$\pm 4\pi/16$	1.50	1.60	1.70	480	600	720	—	1.00	—
Black	—	1.50	1.60	1.70	—	—	—	—	—	—
Red	$\pm 7\pi/16 \pm 2\pi/16$	1.70	1.80	1.90	1020	1200	1380	1.70	2.00	2.30
Green	$\mp 5\pi/16 \pm 2\pi/16$	1.95	2.05	2.15	930	1100	1270	1.55	1.83	2.11
Yellow	$\pm \pi/16 \pm 2\pi/16$	2.25	2.35	2.45	670	800	920	1.13	1.33	1.53
Blue	$\mp 15\pi/16 \pm 2\pi/16$	1.60	1.70	1.80	670	800	920	1.13	1.33	1.53
Magenta	$\mp 11\pi/16 \pm 2\pi/16$	1.80	1.90	2.00	930	1100	1270	1.55	1.83	2.11
Cyan	$\mp 9\pi/16 \pm 2\pi/16$	2.10	2.20	2.30	1020	1200	1380	1.70	2.00	2.30
Gray	—	2.10	2.20	2.30	—	—	—	—	—	—
White	—	2.40	2.50	2.60	—	—	—	—	—	—

Notes. 1 The luminance level and the chroma amplitude of this video signal are ruled only for PAL method.

2 The chroma amplitude is ruled as shown below,  
[Each color's chroma ÷ Color burst's chroma]

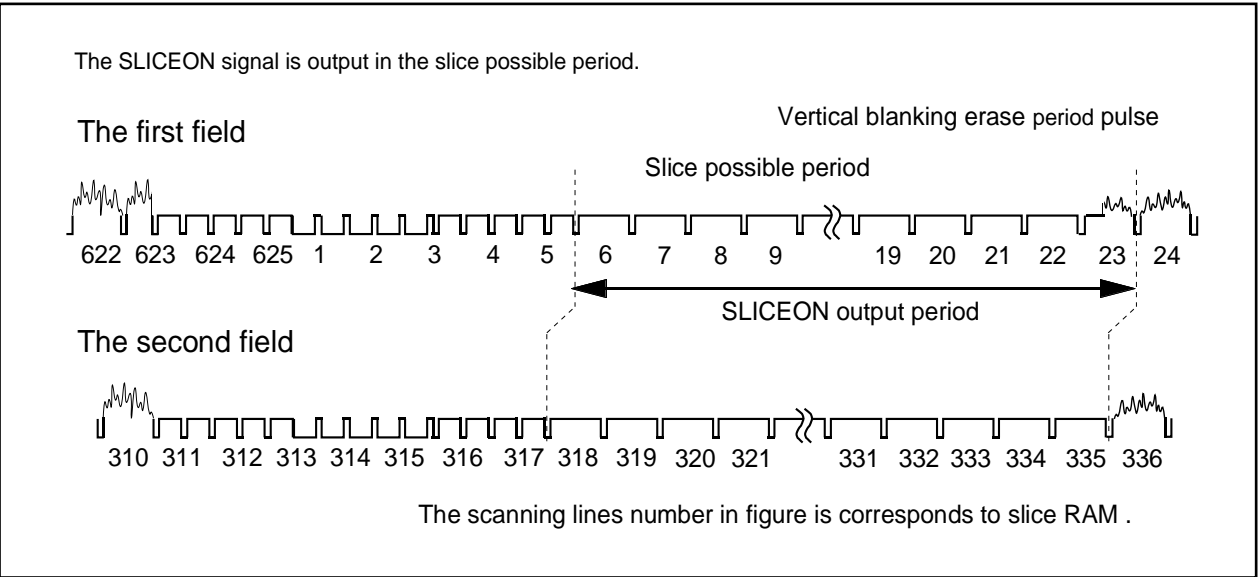


Figure 2.15.30 Slice timing

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.15.10 Display Forms

### (1) Blanking mode

Display forms are shown in Table 2.15.10, display forms at each display mode are shown in Figure 2.15.31.

Table 2.15.26 Display forms

Display mode	DSP1 xx (Addresses 08 <sub>16</sub> and 09 <sub>16</sub> )	DSP0 xx (Addresses 06 <sub>16</sub> and 07 <sub>16</sub> )	BLNK output
Character	0	0	Character size
Disable	0	1	—
Matrix-outline	1	0	All blanking
Halftone	1	1	Blanking OFF

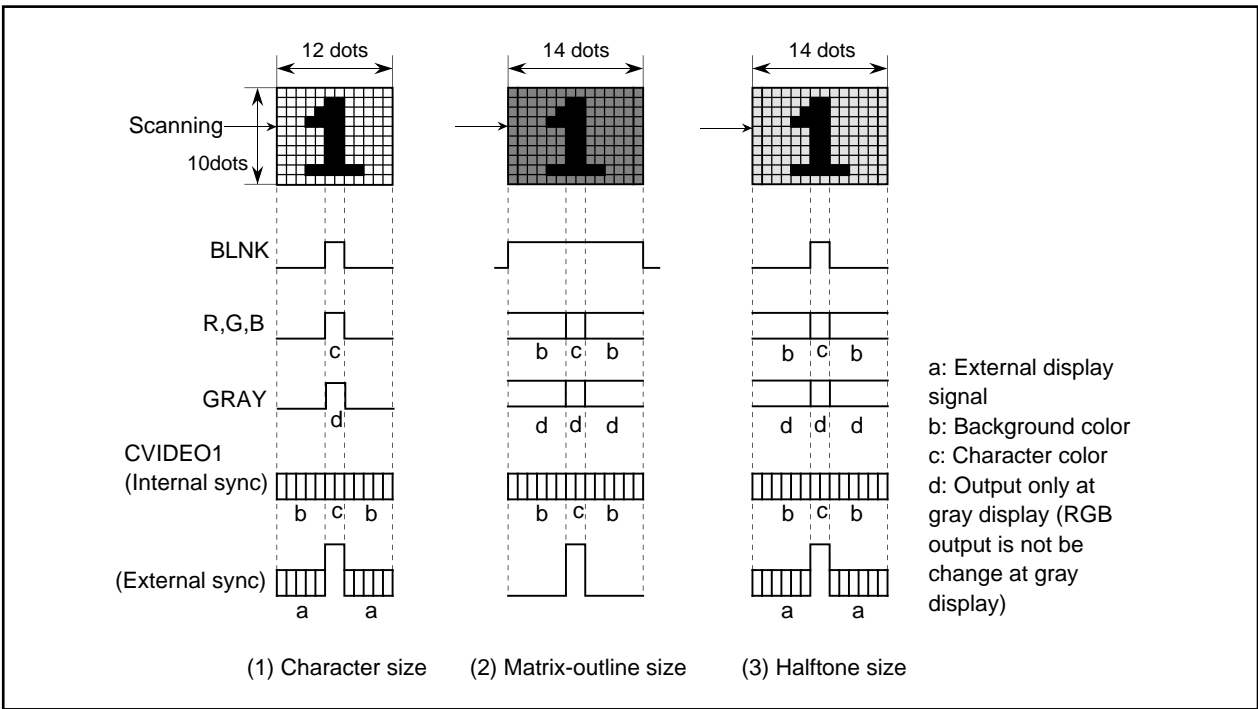


Figure 2.15.31 Blanking mode display

For matrix and halftone, a character's number of dots in the horizontal direction increases to 14. Figure 2.15.32 shows a display example for a case where adjacent characters have different background colors and for character code 7F<sub>16</sub>.

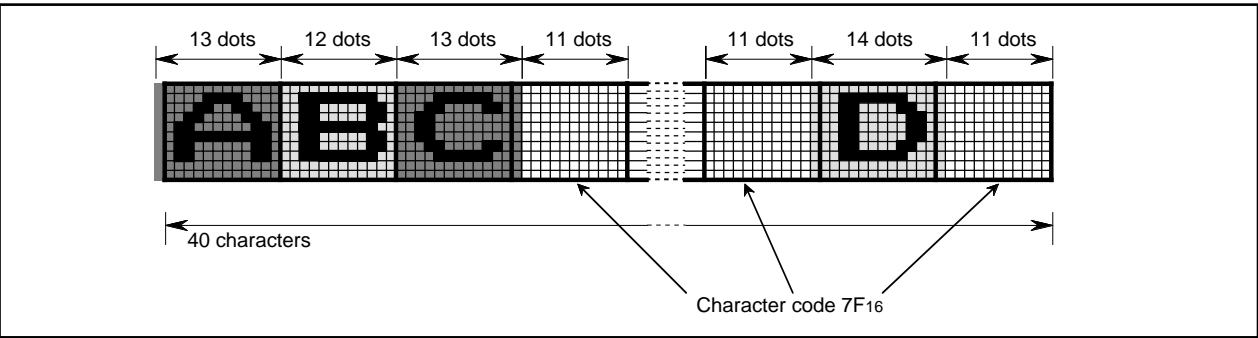


Figure 2.15.32 Number of dots in the horizontal direction at matrix-outline or halftone

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**(2) Setting matrix outline**

Matrix outline is set by using register ALL24 (address 0E16).Matrix outline can be set for each line by using the register DSP1xx (addresses 0816 and 0916) .

However, this setting is disabled if the register EX (address 0D16) is 0 (external sync). An example of setting example of all matrix-outline area is shown in Figure 2.15.33.

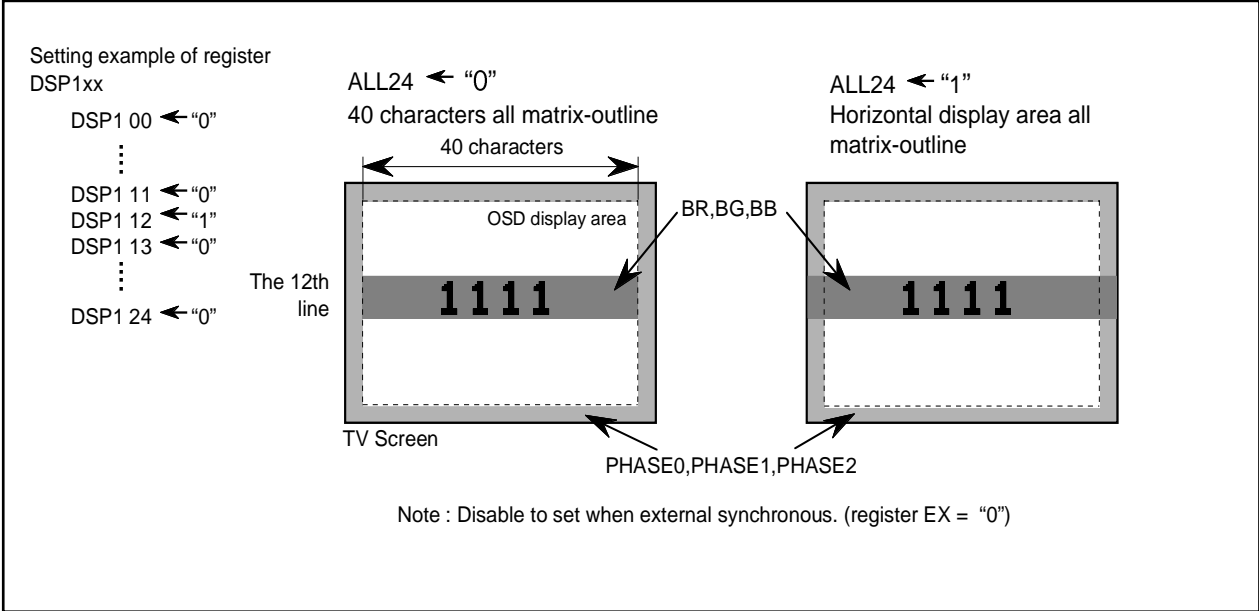


Figure 2.15.33 Setting example of all matrix-outline area

**(3) Blinking mode**

Blinking by BLINK bit of display RAM.

And, use registers BLINK0, 1, and 2 (address 0516) to set the duty ratio and period that determines the blinking time.

Blinking mode is shown in Table 2.15.11(SYRAM do not blink).

The register settings and the duty ratio and period are shown in tables 2.15.12 and 2.15.13.

Table 2.15.11 Blinking mode

Blinking mode	at blinking OFF
Blinking 	

Table 2.15.12 Setting of duty ratio

BLINK1	BLINK0	0	1
0		Blink OFF	Duty 25%
1		Duty 50%	Duty 75%

Table 2.15.13 Setting of cycle

BLINK2	Cycle
0	Approximately 1 second (Vertical sync divided into 1/64)
1	Approximately 0.5 second (Vertical sync divided into 1/32)



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (4) Scroll display mode

The scroll display mode is entered by setting registers SBIT0 to 3 (SA), SLIN0 to 4 (SB) (address 0A16), SST0 to 4 (SC), and SEND0 to 4 (SD) (address 0B16). (Scroll is turned off when SD = 0.)

The screen is scrolled in the range from the (SC)'th line to the (SD-1)'th line, and sections above and below this range are fixed. The beginning line and beginning dot of scroll are the (SA)'th dot on the (SB)'th line.

The screen can be scrolled up or down by successively incrementing or decrementing SA and SB.

Figure 2.15.34 shows examples of how the display is scrolled. The scroll range in these examples contains 20 lines (second to the 21th lines). However, the screen can display only 19 lines at a time, and the remaining one line is handled as a dummy line and not displayed.

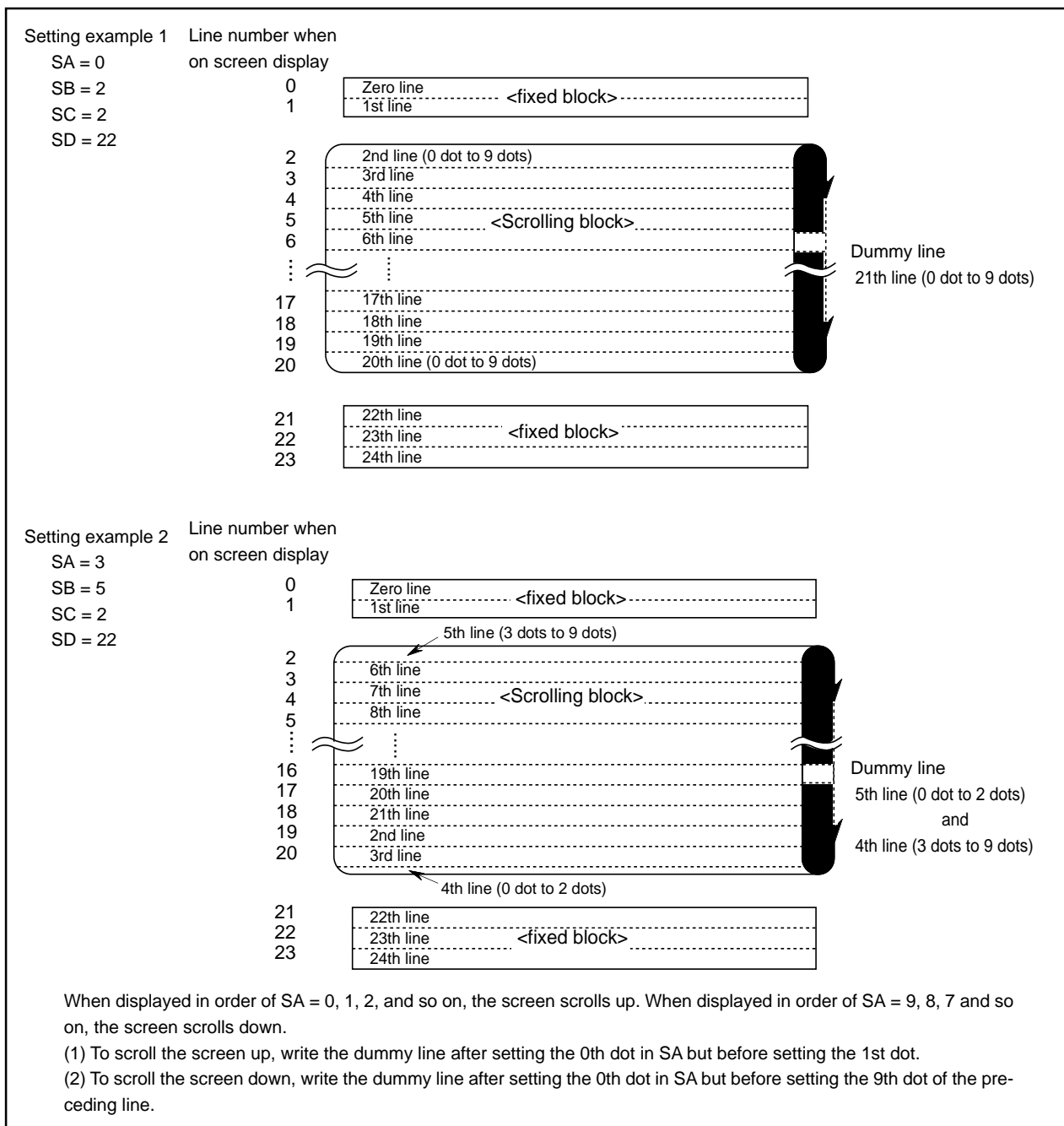


Figure 2.15.34 Scrolling example

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

2.15.11 8/4 Humming Decoder

8/4 humming decoder opetates only by written the data which 8/4 humming- decoded to 8/4 humming register (address 021A16). 8/4 humming register consists of 16 bits, can decode two data at a time. Can obtain the decoded result by reading 8/4 humming register, and the decoded value and error information are output. Corrects and outputs the decoded value for single error, and outputs only error information for double error. Decoded result is shown in Figure 2.15.35 and humming 8/4 register composition is shown in Figure 2.15.36.

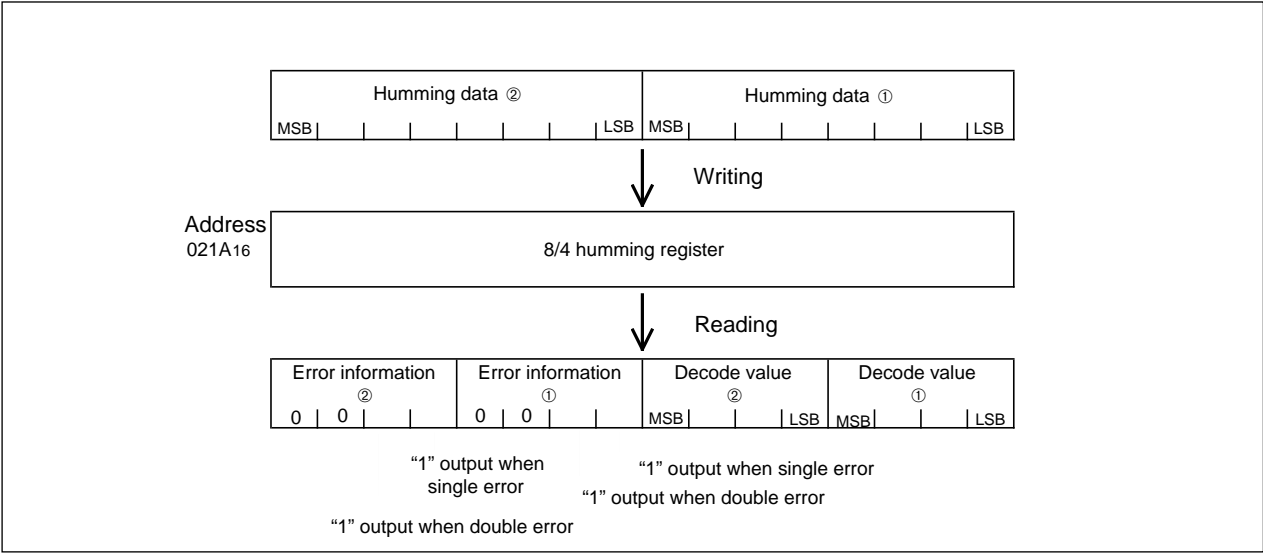


Figure 2.15.35 Decoded result

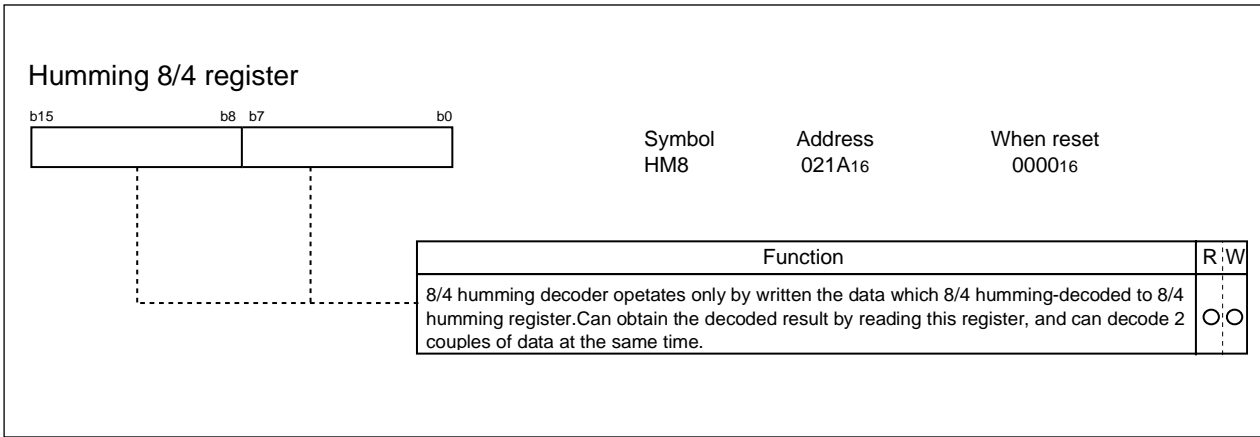


Figure 2.15.36 Humming 8/4 register composition

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

2.15.12 24/18Humming Decoder

24/18 humming decoder operates only by written the data which 24/18 humming-encoded to 24/18 humming register 0 (address 021C16) and 1 (address 021E16). Can obtain the decoded result by reading the same 24/18 humming register. Decoded result is shown in Figure 2.15.37 and humming 24/18 register composition is shown in Figure 2.15.38.

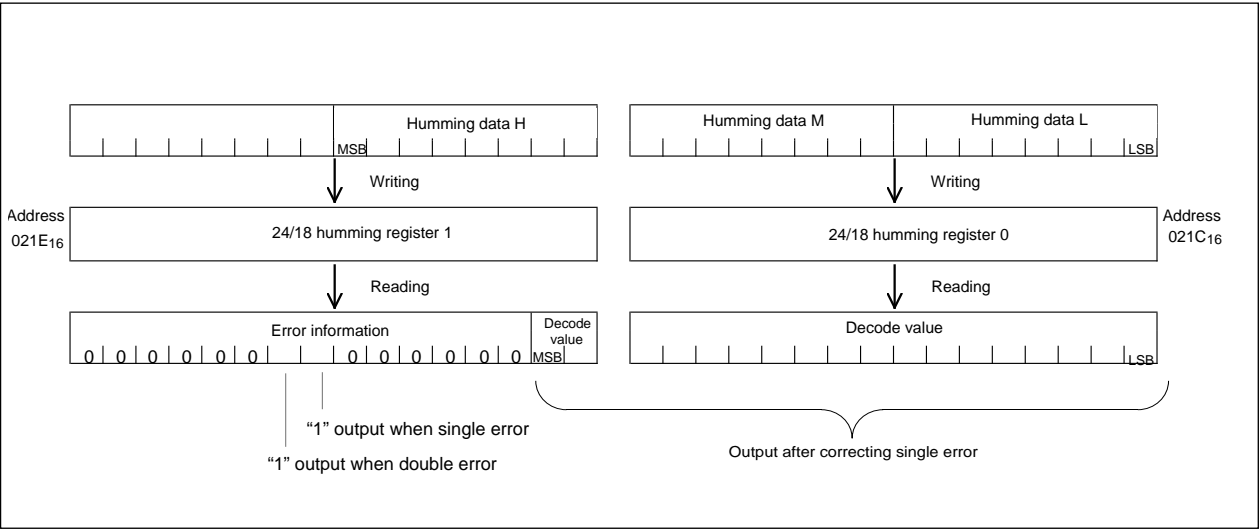


Figure 2.15.37 Decoded result

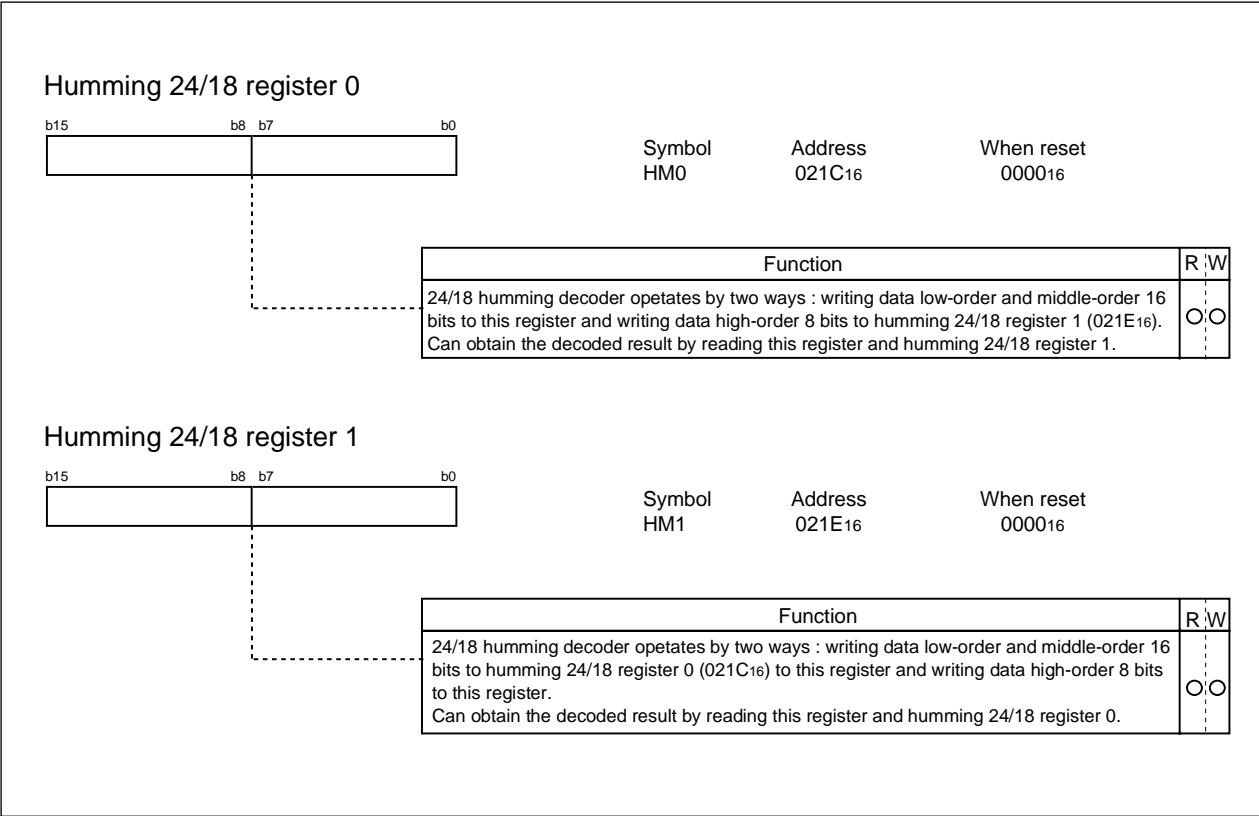


Figure 2.15.38 Humming 24/18 register composition

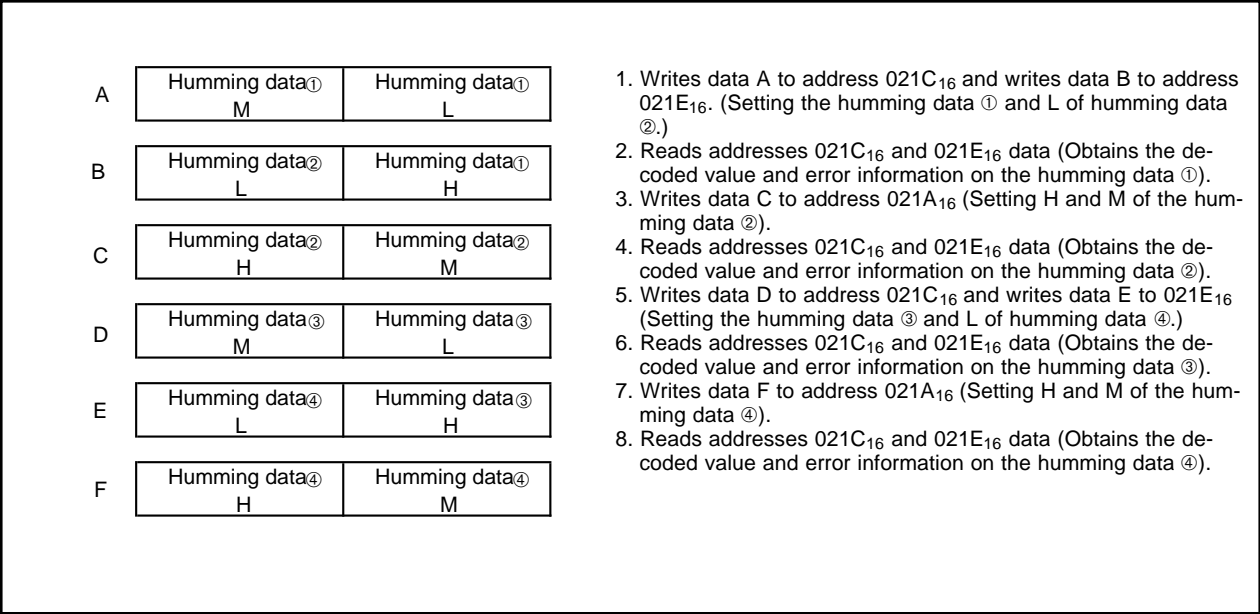
# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Continuous error correction**

When uses humming 8/4 (address 021A<sub>16</sub>) at tha same time as humming 24/18, can do the continu-  
ous error correction.

Continuous error correction sequence is shown in Figure 2.15.39.



**Figure 2.15.39 Continuous error correction sequence**

Then, because using a part of circuit of humming 8/4 about this operation, cannot use this operation at the same time.

When using the humming circuit, do the decoded result reading operation at once after the setting data of humming. And do not access other memories (Including the humming circuit) before reading of the decoded result.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.15.13 I/O Composition of pins for Expansion Memory

Figure 2.15.40 and figure 2.15.41 show pins for expansion memory.

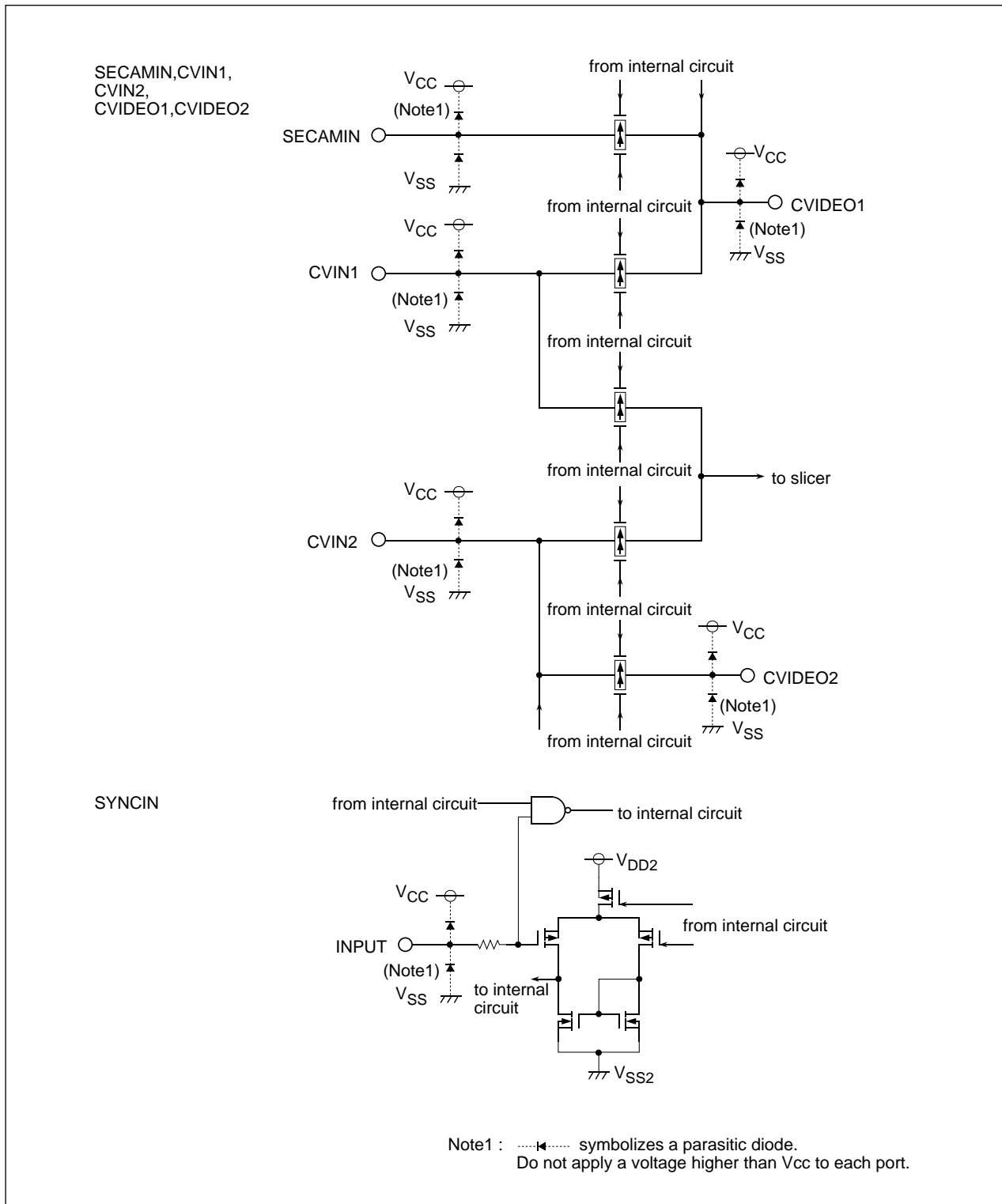


Figure 2.15.40 Pins for expansion memory(1)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

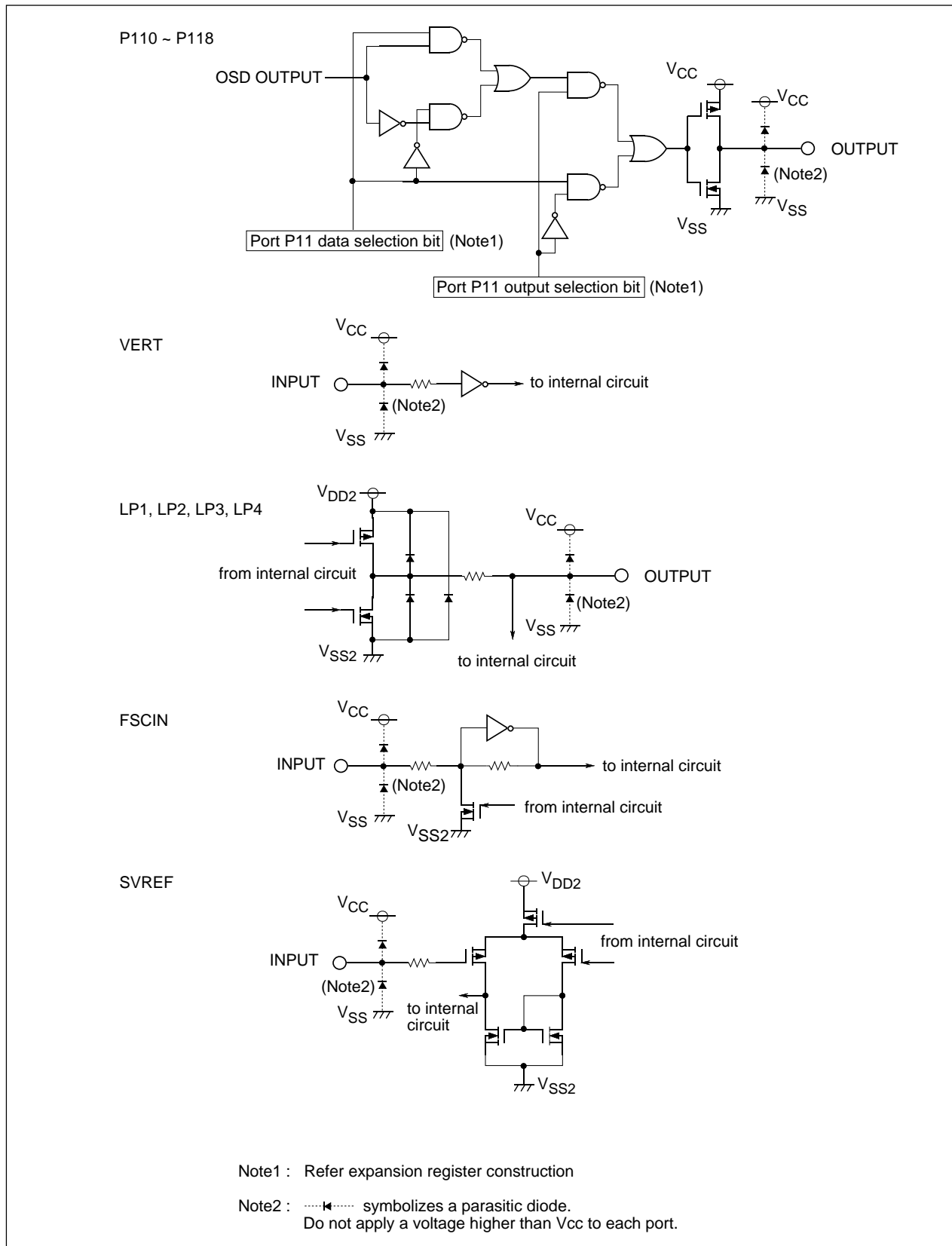


Figure 2.15.41 Pins for expansion memory(2)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 2.16 Programmable I/O Ports

There are 87 programmable I/O ports: P0 to P10 (excluding P85). Each port can be set independently for input or output using the direction register. A pull-up resistance for each block of 4 ports can be set. P85 is an input-only port and has no built-in pull-up resistance.

Figures 2.16.1 to 2.16.4 show the programmable I/O ports. Figure 2.16.5 shows the I/O pins.

Each pin functions as a programmable I/O port and as the I/O for the built-in peripheral devices.

To use the pins as the inputs for the built-in peripheral devices, set the direction register of each pin to input mode. When the pins are used as the outputs for the built-in peripheral devices (other than the D-A converter), they function as outputs regardless of the contents of the direction registers. When pins are to be used as the outputs for the D-A converter, do not set the direction registers to output mode. See the descriptions of the respective functions for how to set up the built-in peripheral devices.

### (1) Direction registers

Figure 2.16.6 shows the direction registers.

These registers are used to choose the direction of the programmable I/O ports. Each bit in these registers corresponds one for one to each I/O pin.

Note: There is no direction register bit for P85.

### (2) Port registers

Figure 2.16.7 shows the port registers.

These registers are used to write and read data for input and output to and from an external device. A port register consists of a port latch to hold output data and a circuit to read the status of a pin. Each bit in port registers corresponds one for one to each I/O pin.

### (3) Pull-up control registers

Figure 2.16.8 shows the pull-up control registers.

The pull-up control register can be set to apply a pull-up resistance to each block of 4 ports. When ports are set to have a pull-up resistance, the pull-up resistance is connected only when the direction register is set for input.

However, the pull-up control register of P0 to P5 is invalid.

### (4) Port control register

Figure 2.16.9 shows the port control register.

The bit 0 of port control register is used to read port P1 as follows:

0 : When port P1 is input port, port input level is read.

When port P1 is output port, the contents of port P1 register is read.

1 : The contents of port P1 register is read always.

This register is valid in the following:

- External bus width is 8 bits.
- Port P1 can be used as a port in multiplexed bus for the entire space.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

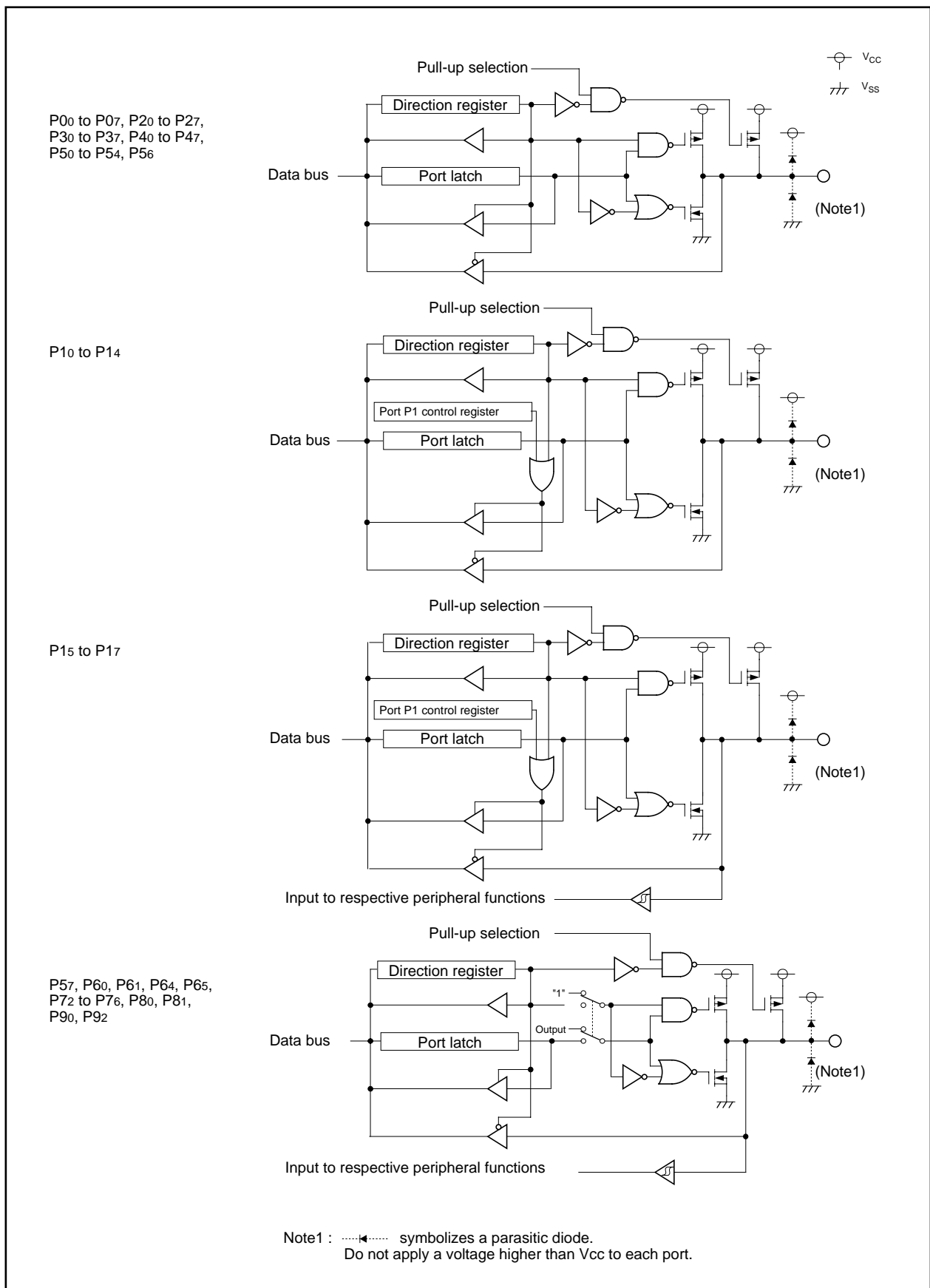


Figure 2.16.1 Programmable I/O ports (1)



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

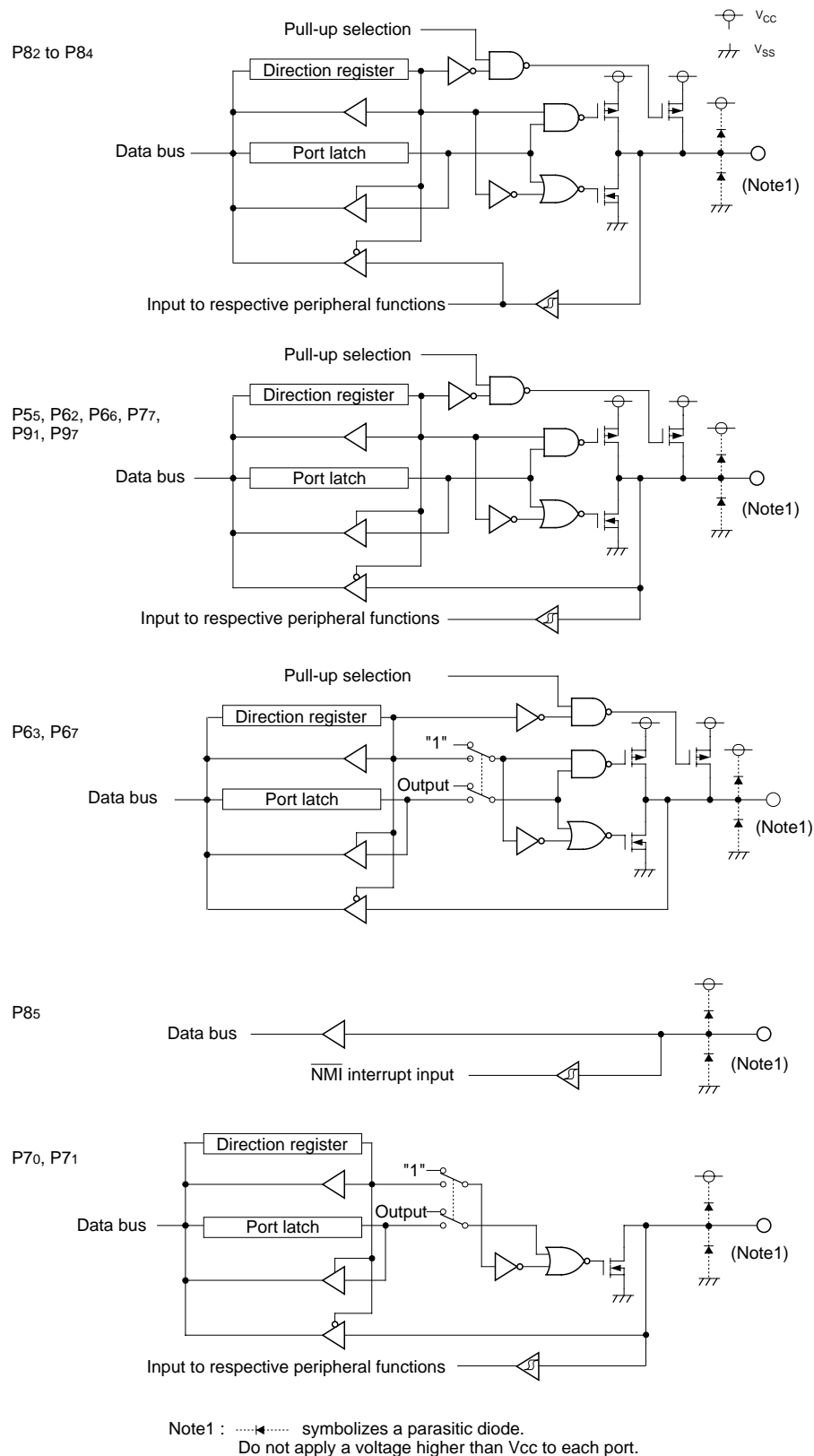


Figure 2.16.2 Programmable I/O ports (2)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

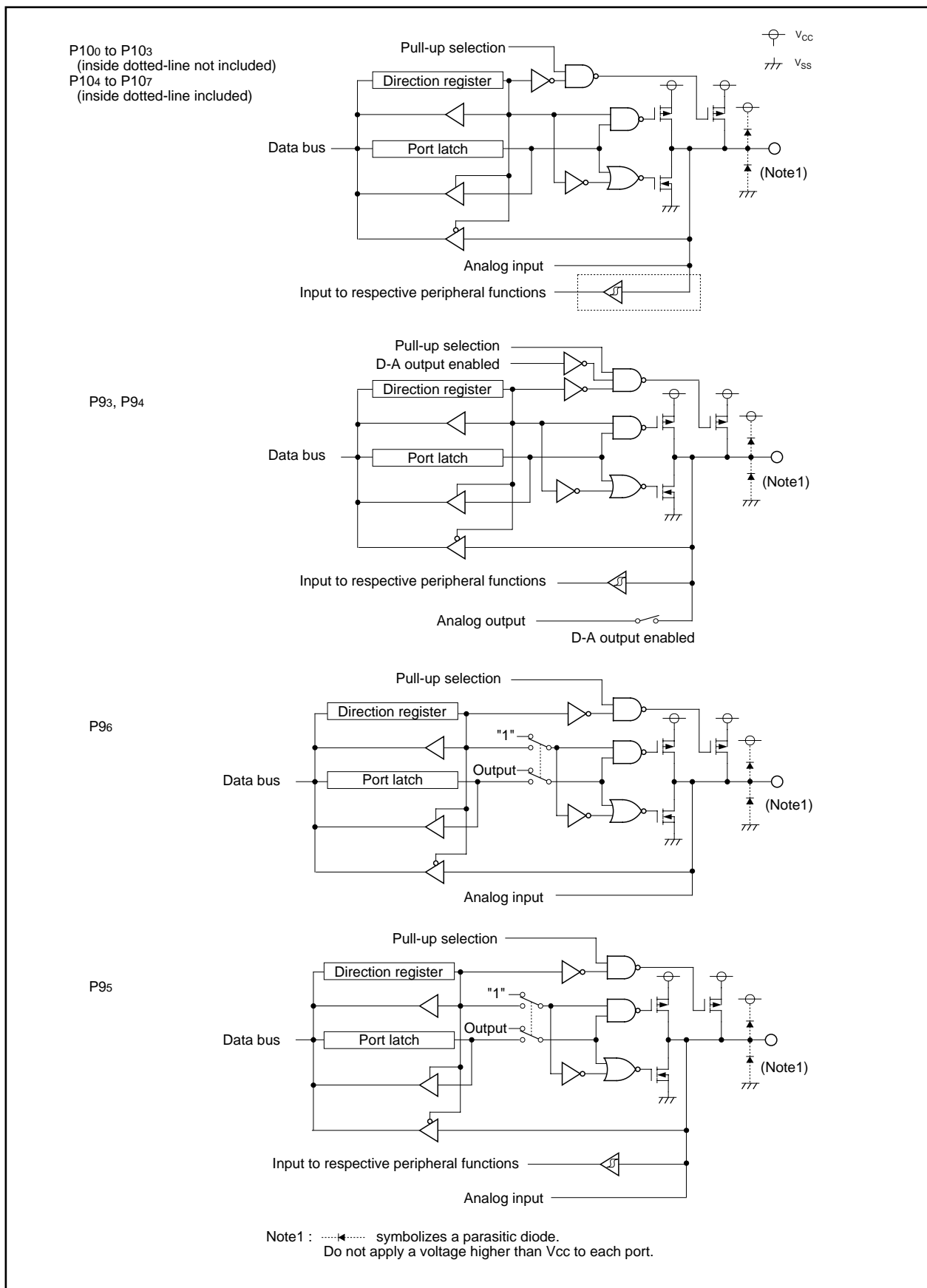


Figure 2.16.3 Programmable I/O ports (3)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

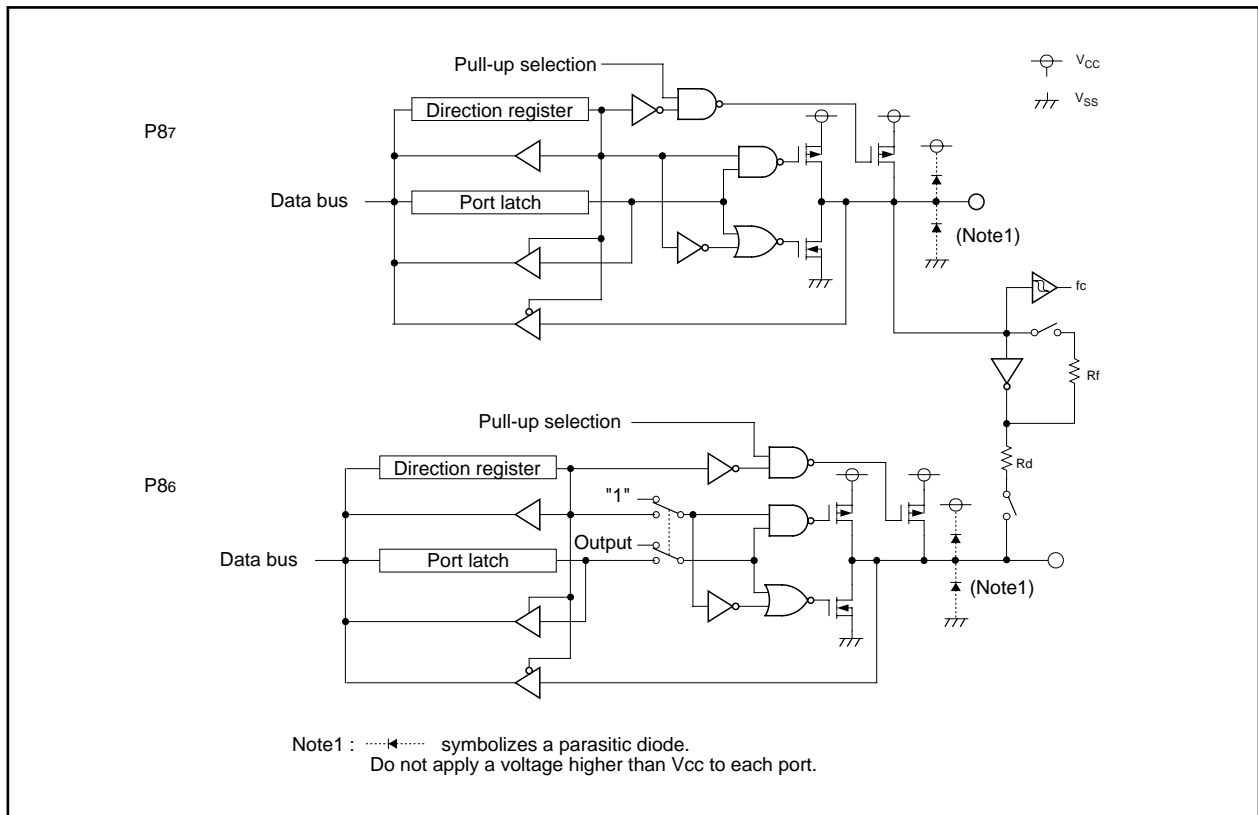


Figure 2.16.4 Programmable I/O ports (4)

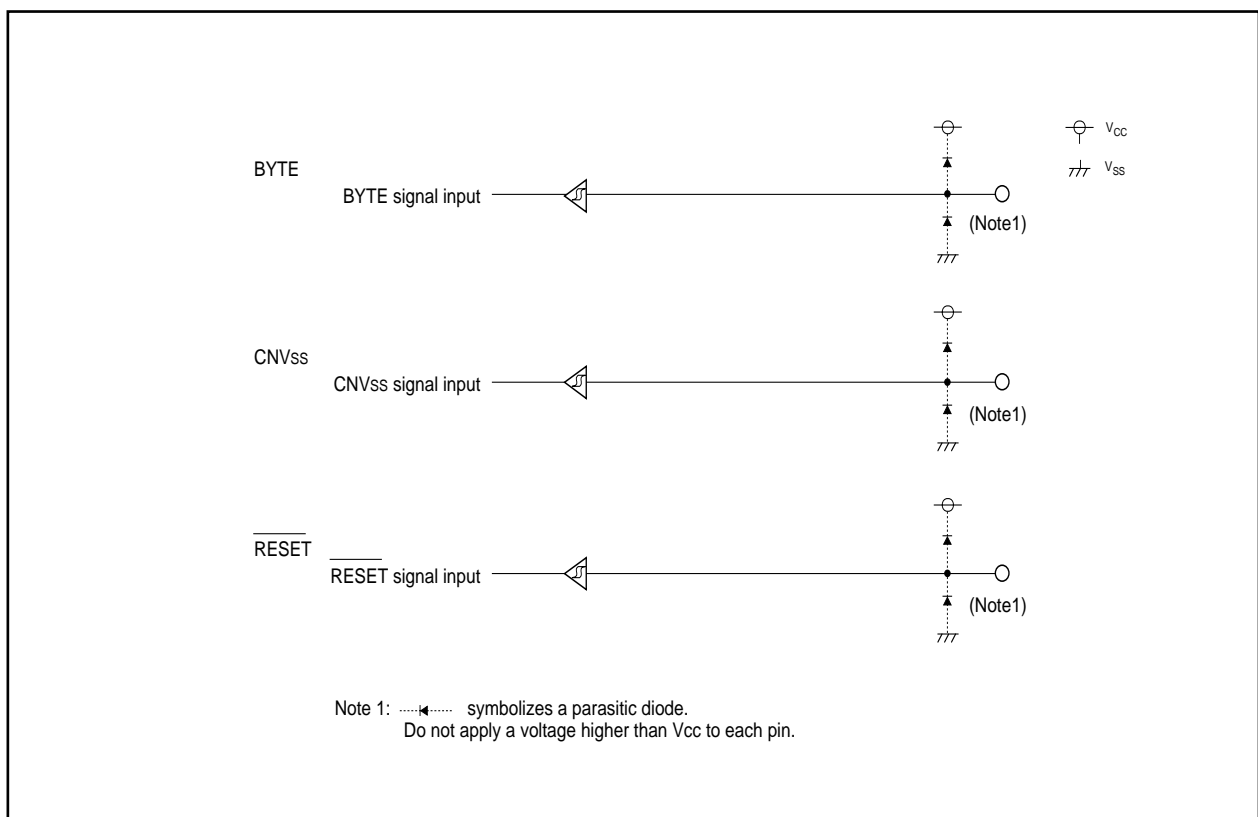


Figure 2.16.5 I/O pins

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

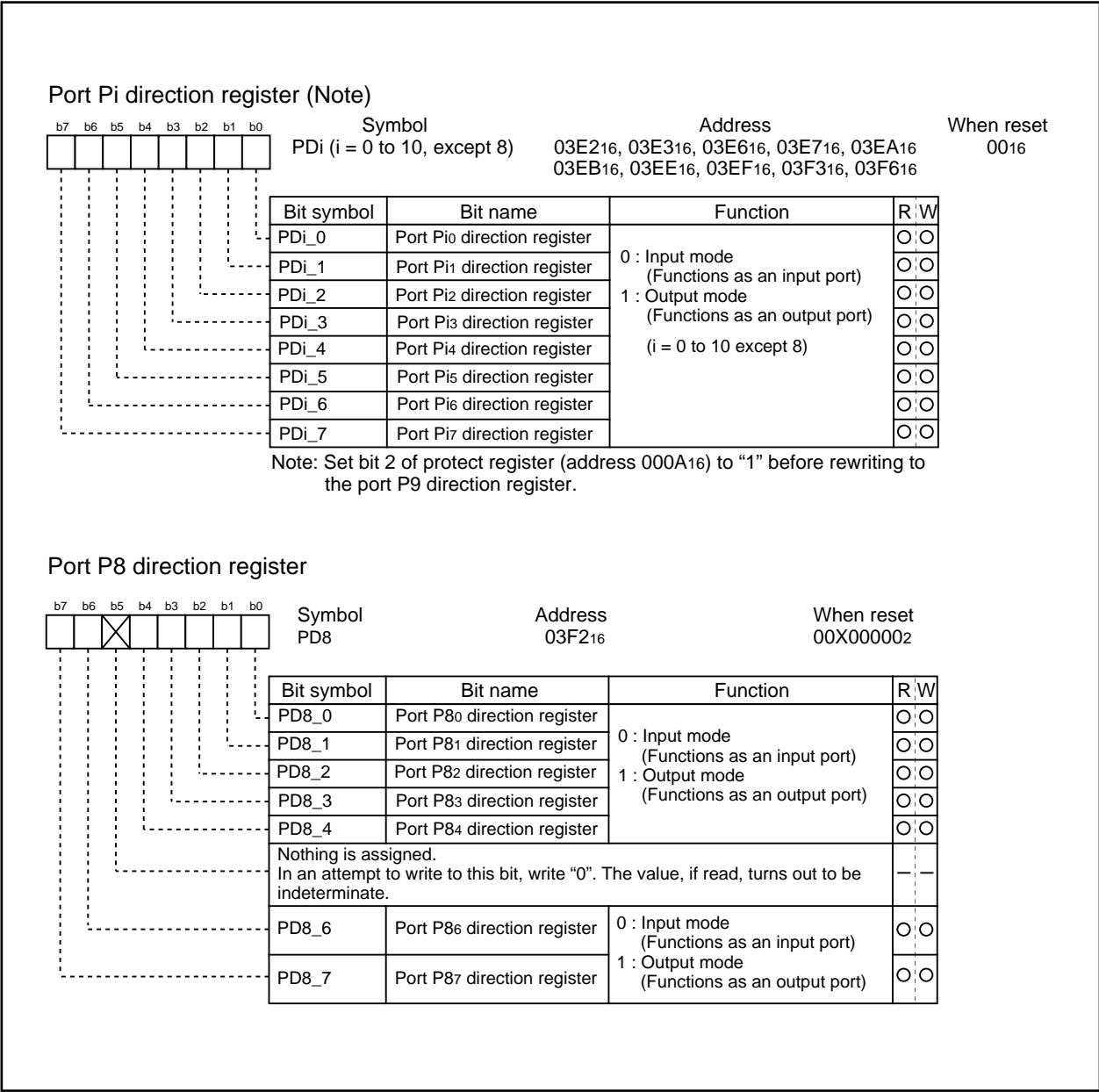


Figure 2.16.6 Direction register

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

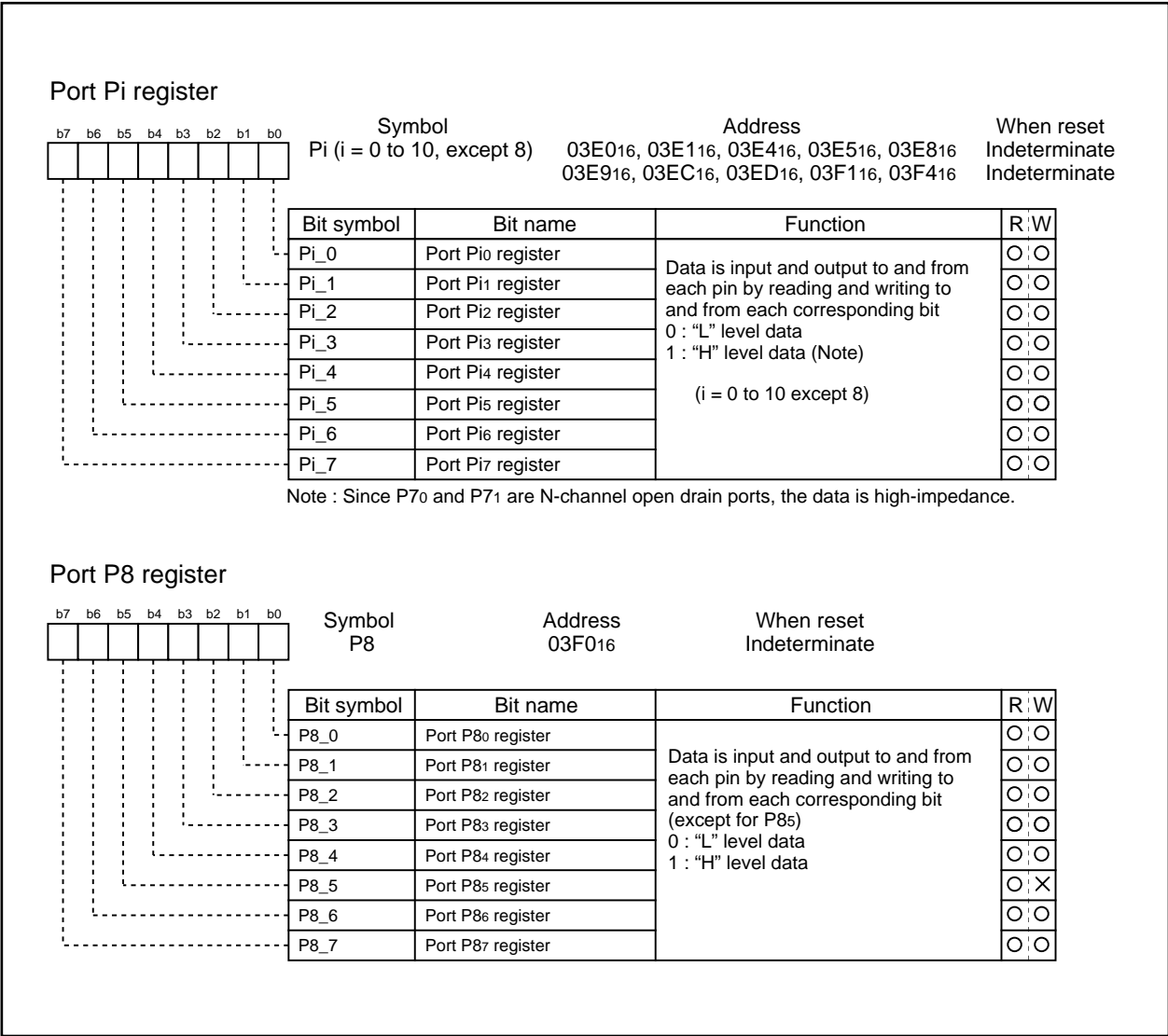


Figure 2.16.7 Port register

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Pull-up control register 0

b7	b6	b5	b4	b3	b2	b1	b0	Symbol PUR0	Address 03FC <sub>16</sub>	When reset 00 <sub>16</sub>	
								Bit symbol	Bit name	Function	R W
								PU00	P00 to P03 pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Inhibited	○ ○
								PU01	P04 to P07 pull-up		○ ○
								PU02	P10 to P13 pull-up		○ ○
								PU03	P14 to P17 pull-up		○ ○
								PU04	P20 to P23 pull-up		○ ○
								PU05	P24 to P27 pull-up		○ ○
								PU06	P30 to P33 pull-up		○ ○
								PU07	P34 to P37 pull-up		○ ○

## Pull-up control register 1

b7	b6	b5	b4	b3	b2	b1	b0	Symbol PUR1	Address 03FD <sub>16</sub>	When reset 00 <sub>16</sub> (Note 2)	
								Bit symbol	Bit name	Function	R W
								PU10	P40 to P43 pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Inhibited	○ ○
								PU11	P44 to P47 pull-up		○ ○
								PU12	P50 to P53 pull-up		○ ○
								PU13	P54 to P57 pull-up		○ ○
								PU14	P60 to P63 pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○ ○
								PU15	P64 to P67 pull-up		○ ○
								PU16	P70 to P73 pull-up (Note 1)		○ ○
								PU17	P74 to P77 pull-up		○ ○

Note 1: Since P70 and P71 are N-channel open drain ports, pull-up is not available for them.

Note 2: When the Vcc level is being impressed to the CNVss terminal, this register becomes to 02<sub>16</sub> when reset (PU11 becomes to "1").

## Pull-up control register 2

b7	b6	b5	b4	b3	b2	b1	b0	Symbol PUR2	Address 03FE <sub>16</sub>	When reset 00 <sub>16</sub>	
								Bit symbol	Bit name	Function	R W
								PU20	P80 to P83 pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○ ○
								PU21	P84 to P87 pull-up (Except P85)		○ ○
								PU22	P90 to P93 pull-up		○ ○
								PU23	P94 to P97 pull-up		○ ○
								PU24	P100 to P103 pull-up		○ ○
								PU25	P104 to P107 pull-up		○ ○
								Nothing is assigned.			— —
								In an attempt to write to these bits, write "0". The value, if read, turns out to be "0".			— —

Figure 2.16.8 Pull-up control register

M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

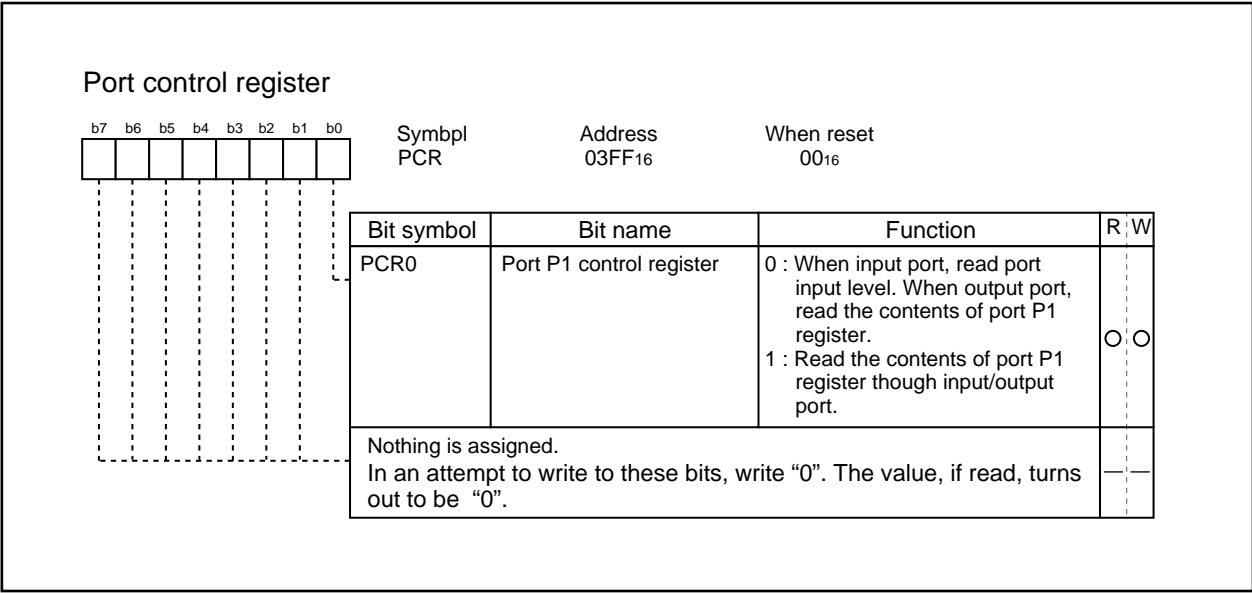


Figure 2.16.9 Port control register

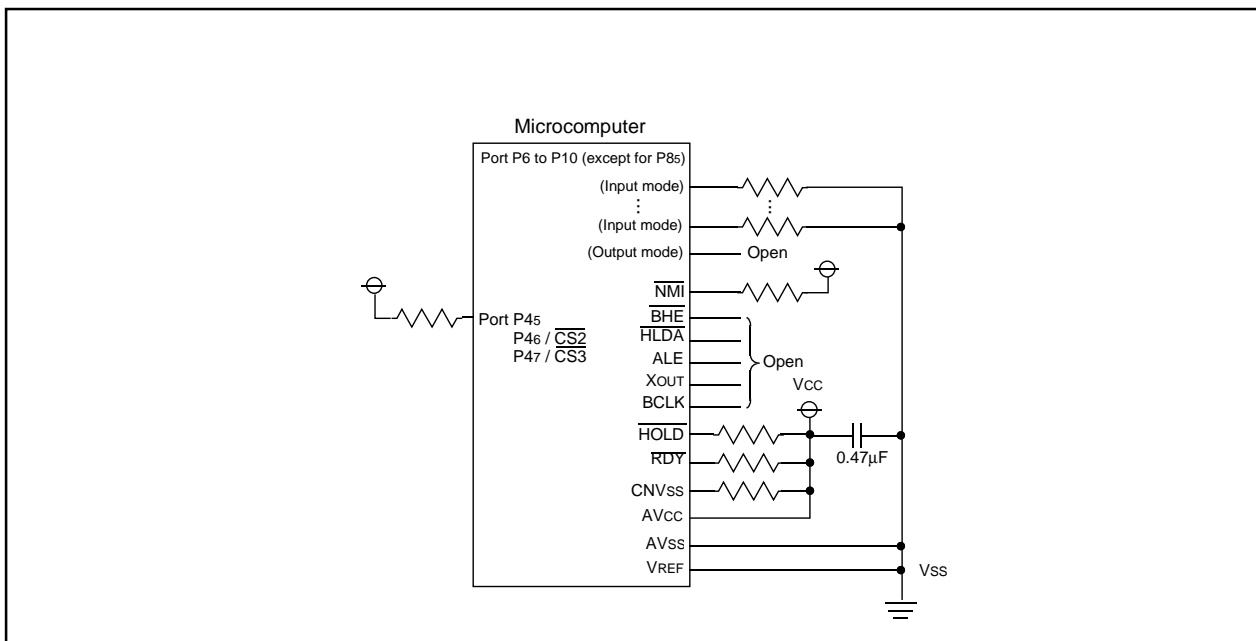
# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Table 2.16.1 Example connection of unused pins.**

Pin name	Connection
Ports P6 to P10 (excluding P85)	After setting for input mode, connect every pin to Vss or Vcc via a resistor; or after setting for output mode, leave these pins open.
P45, P46/CS2, P47/CS3	Sets ports to input mode, sets bits CS2, CS3 to 0, and connects to Vcc via resistors (pull-up).
BHE, ALE, HLDA, XOUT(Note), BCLK	Open
HOLD, RDY, NMI	Connect via resistor to Vcc (pull-up)
AVcc	Connect to Vcc
AVss, VREF	Connect to Vss
CNVss	Connect via resistor to Vcc (pull-up)

Note: With external clock input to XIN pin.



**Figure 2.16.10 Example connection of unused pins**



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 3. Usage Precaution

### Timer A (timer mode)

- (1) Reading the timer Ai register while a count is in progress allows reading, with arbitrary timing, the value of the counter. Reading the timer Ai register with the reload timing gets "FFFF<sub>16</sub>". Reading the timer Ai register after setting a value in the timer Ai register with a count halted but before the counter starts counting gets a proper value.

### Timer A (event counter mode)

- (1) Reading the timer Ai register while a count is in progress allows reading, with arbitrary timing, the value of the counter. Reading the timer Ai register with the reload timing gets "FFFF<sub>16</sub>" by underflow or "0000<sub>16</sub>" by overflow. Reading the timer Ai register after setting a value in the timer Ai register with a count halted but before the counter starts counting gets a proper value.
- (2) When stop counting in free run type, set timer again.
- (3) In the case of using "Event counter mode" as "Free-Run type" for timer A, the timer register contents may be unknown when counting begins. If the timer register is set before counting has started, then the starting value will be unknown.  
This issue will occur only for the "Event counter mode" operating as "Free-Run type". The value of the timer register will not be unknown during counting.

### Timer A (one-shot timer mode)

- (1) Setting the count start flag to "0" while a count is in progress causes as follows:
  - The counter stops counting and a content of reload register is reloaded.
  - The TAIOUT pin outputs "L" level.
  - The interrupt request generated and the timer Ai interrupt request bit goes to "1".
- (2) The timer Ai interrupt request bit goes to "1" if the timer's operation mode is set using any of the following procedures:
  - Selecting one-shot timer mode after reset.
  - Changing operation mode from timer mode to one-shot timer mode.
  - Changing operation mode from event counter mode to one-shot timer mode.

Therefore, to use timer Ai interrupt (interrupt request bit), set timer Ai interrupt request bit to "0" after the above listed changes have been made.

### Timer A (pulse width modulation mode)

- (1) The timer Ai interrupt request bit becomes "1" if setting operation mode of the timer in compliance with any of the following procedures:
  - Selecting PWM mode after reset.
  - Changing operation mode from timer mode to PWM mode.
  - Changing operation mode from event counter mode to PWM mode.

Therefore, to use timer Ai interrupt (interrupt request bit), set timer Ai interrupt request bit to "0" after the above listed changes have been made.
- (2) Setting the count start flag to "0" while PWM pulses are being output causes the counter to stop counting. If the TAIOUT pin is outputting an "H" level in this instance, the output level goes to "L", and the timer Ai interrupt request bit goes to "1". If the TAIOUT pin is outputting an "L" level in this instance, the level does not change, and the timer Ai interrupt request bit does not become "1".

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Timer B (timer mode, event counter mode)

- (1) Reading the timer Bi register while a count is in progress allows reading , with arbitrary timing, the value of the counter. Reading the timer Bi register with the reload timing gets "FFFF<sub>16</sub>". Reading the timer Bi register after setting a value in the timer Bi register with a count halted but before the counter starts counting gets a proper value.

## Timer B (pulse period/pulse width measurement mode)

- (1) If changing the measurement mode select bit is set after a count is started, the timer Bi interrupt request bit goes to "1".
- (2) When the first effective edge is input after a count is started, an indeterminate value is transferred to the reload register. At this time, timer Bi interrupt request is not generated.

## A-D Converter

- (1) Write to each bit (except bit 6) of A-D control register 0, to each bit of A-D control register 1, and to bit 0 of A-D control register 2 when A-D conversion is stopped (before a trigger occurs).  
In particular, when the Vref connection bit is changed from "0" to "1", start A-D conversion after an elapse of 1  $\mu$ s or longer.
- (2) When changing A-D operation mode, select analog input pin again.
- (3) Using one-shot mode or single sweep mode  
Read the correspondence A-D register after confirming A-D conversion is finished. (It is known by A-D conversion interrupt request bit.)
- (4) Using repeat mode, repeat sweep mode 0 or repeat sweep mode 1  
Use the undivided main clock as the internal CPU clock.

## Stop Mode and Wait Mode

- (1) When returning from stop mode by hardware reset,  $\overline{\text{RESET}}$  pin must be set to "L" level until main clock oscillation is stabilized.
- (2) When switching to either wait mode or stop mode, instructions occupying four bytes either from the WAIT instruction or from the instruction that sets the every-clock stop bit to "1" within the instruction queue are prefetched and then the program stops. So put at least four NOPs in succession either to the WAIT instruction or to the instruction that sets the every-clock stop bit to "1".

## Interrupts

- (1) Reading address 00000<sub>16</sub>
  - When maskable interrupt is occurred, CPU read the interrupt information (the interrupt number and interrupt request level) in the interrupt sequence.  
The interrupt request bit of the certain interrupt written in address 00000<sub>16</sub> will then be set to "0".  
Reading address 00000<sub>16</sub> by software sets enabled highest priority interrupt source request bit to "0".  
Though the interrupt is generated, the interrupt routine may not be executed.  
Do not read address 00000<sub>16</sub> by software.
- (2) Setting the stack pointer
  - The value of the stack pointer immediately after reset is initialized to 0000<sub>16</sub>. Accepting an interrupt before setting a value in the stack pointer may become a factor of runaway. Be sure to set a value in the stack pointer before accepting an interrupt.  
When using the  $\overline{\text{NMI}}$  interrupt, initialize the stack point at the beginning of a program. Concerning the first instruction immediately after reset, generating any interrupts including the  $\overline{\text{NMI}}$  interrupt is prohibited.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
 with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## (3) The $\overline{\text{NMI}}$ interrupt

- As for the  $\overline{\text{NMI}}$  interrupt pin, an interrupt cannot be disabled. Connect it to the VCC pin via a resistor (pull-up) if unused. Be sure to work on it.
- Do not get either into stop mode with the  $\overline{\text{NMI}}$  pin set to "L".

## (4) External interrupt

- When the polarity of the INT0 to  $\overline{\text{INT5}}$  pins is changed, the interrupt request bit is sometimes set to "1". After changing the polarity, set the interrupt request bit to "0".

## (5) Rewrite the interrupt control register

- To rewrite the interrupt control register, do so at a point that does not generate the interrupt request for that register. If there is possibility of the interrupt request occur, rewrite the interrupt control register after the interrupt is disabled. The program examples are described as follow:

### Example 1:

```

INT_SWITCH1:
  FCLR  I           ; Disable interrupts.
  AND.B #00h, 0055h ; Clear TA0IC int. priority level and int. request bit.
  NOP                      ; Four NOP instructions are required when using HOLD function.
  NOP
  FSET  I           ; Enable interrupts.

```

### Example 2:

```

INT_SWITCH2:
  FCLR  I           ; Disable interrupts.
  AND.B #00h, 0055h ; Clear TA0IC int. priority level and int. request bit.
  MOV.W MEM, R0     ; Dummy read.
  FSET  I           ; Enable interrupts.

```

### Example 3:

```

INT_SWITCH3:
  PUSHC FLG        ; Push Flag register onto stack
  FCLR  I           ; Disable interrupts.
  AND.B #00h, 0055h ; Clear TA0IC int. priority level and int. request bit.
  POPC  FLG        ; Enable interrupts.

```

The reason why two NOP instructions (four when using the HOLD function) or dummy read are inserted before FSET I in Examples 1 and 2 is to prevent the interrupt enable flag I from being set before the interrupt control register is rewritten due to effects of the instruction queue.

- When a instruction to rewrite the interrupt control register is executed but the interrupt is disabled, the interrupt request bit is not set sometimes even if the interrupt request for that register has been generated. This will depend on the instruction. If this creates problems, use the below instructions to change the register.

Instructions : AND, OR, BCLR, BSET

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Other Notes

### (1) Timing of power supplying

The power need to supply to VCC, VDD1, VDD2, VDD3 and AVCC at a time. While operating, must set same voltage.

### (2) Power supply noise and latch-up

In order to avoid power supply noise and latch-up, connect a bypass capacitor (more than 0.1μF) directly between the VCC pin and VSS pin, VDD1 pin and VSS1 pin, VDD2 pin and VSS2 pin, VDD3 pin and VSS3 pin, AVCC pin and AVSS pin using a heavy wire.

### (3) After the reset

After the reset, until the oscillator circuit stabilizes, data is sometimes not set correctly in the display RAM, font RAM, SYRAM and VBIRAM. Therefore, use the following start-up procedure.

- (a) Reset release.
- (b) Set expansion register CK\_VCO, XTAL\_VCO, PDC\_VCO\_ON, VPS\_VCO\_ON = "H". (oscillation start)
- (c) Set expansion register SYNCSEP\_ON0 = "H".
- (d) Set expansion register NXP = "H".
- (e) Set expansion register PCn, DIV\_PDCn, DIV\_PDCSn, DIV\_VPSn, DIV\_VPSSn.
- (f) Disable data input for a 20 m sec (time enough to allow the internal oscillator circuit to stabilize).
- (g) Set other expansion registers.
- (h) Set the SYRAM.
- (i) Set the display RAM.
- (j) Set expansion register DSPON and DSPONV to display ON.
- (k) Possible to access slice RAM.

### (4) When resuming internal oscillation from the off state

The each internal oscillator circuit of expansion function stops oscillating when expansion register CK\_VCO, XTAL\_VCO, PDC\_VCO\_ON, VPS\_VCO\_ON = "L".

When resuming internal oscillation from the off state, up until the oscillator circuit stabilizes, data is sometimes not set correctly in the display RAM, font RAM, SYRAM and VBIRAM. Therefore, start oscillation as follows.

- (a) Set expansion register CK\_VCO = "H".
- (b) Set expansion register XTAL\_VCO = "H".
- (c) Set expansion register PDC\_VCO\_ON = "H", VPS\_VCO\_ON = "H".  
(Necessity none when data slicer is not used)
- (d) Wait for a 20 m sec. (time enough to allow the internal oscillator circuit to stabilize)
- (e) Access the other memories.

Especially, set expansion register XTAL\_VCO = "H" when access to display RAM, font RAM, SYRAM, VBIRAM and slice RAM. And input 4.43 MHz sub carrier frequency clock from the FSCIN pin.

Access the memory after waiting for 20ms certainly when resuming synchronous oscillation from the off state, and begin to input clock into the FSCIN pin.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

(5) Other notes on oscillation

Make note of the fact that the internal oscillator circuit cannot stabilize in the below situations.

- (a) When the external composite video signal is discontinuous. (when changing channels, etc.)
- (b) When expansion register PCn setting is changed.
- (c) When expansion register SYNCSEP\_ON0 setting is changed.

Before changing settings, turn expansion registers DSPON and DSPONV off. Also, disable data input for 20 m sec after making settings.

(6) When no external composite video signal is input

Without a signal, characters cannot be displayed by external synchronization. Therefore, switch to internal synchronization.

(7) When signal level of the external composite video signal is extremely poor

With a weak electric field, character display is uncontrollable by external synchronization. Therefore, switch to internal synchronization.

(8) When oscillation circuit stop for data slicer

Expansion register PDC\_VCO\_ON,VPS\_VCO\_ON is set at "L", when the data slicer is not used, and the oscillation is stopped. When starting oscillation again, set data at the following order.

- (a) Set expansion register PDC\_VCO\_ON,VPS\_VCO\_ON = "L".
- (b) Set expansion register PDC\_VCO\_ON,VPS\_VCO\_ON = "H".
- (c) 60 ms or more is a waiting state (stability period of internal oscillation circuit + data slice preparation).

To operate slice RAM , set expansion register XTAL\_VCO = "H". And input 4.43 MHz sub carrier frequency clock from the FSCIN pin.

Access the memories after waiting for 20 ms certainly when resuming synchronous oscillation from the off state , and begin to input clock into the FSCIN pin.

(9) When the data slicer is used without displaying OSD

If expansion register DSPON is set in "L", the OSD display is turned off.

Expansion register CK\_VCO must be set "H" in that case

(10) At stop mode (clock is stopped)

Set each input pins to as follows.

- (a) Set VERT pin = VSS.
- (b) Stop the FSCIN pin input.
- (c) Set expansion register STBY0 and STBY1 = "H".

Set all expansion registers to "L" except for the superscription register.

(11) When operation start from stop mode (clock is stopped)

Input FSCIN pin clock after set "L" to register STBY0 and STBY1.

At next, set expansion register as notes (4).

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 4. Electrical characteristic

**Table 4.1 Absolute maximum ratings**

Symbol	Parameter		Condition	Rated value	Unit
V <sub>cc</sub>	Supply voltage		V <sub>cc</sub> =AV <sub>cc</sub>	-0.3 to 5.75	V
AV <sub>cc</sub>	Analog supply voltage		V <sub>cc</sub> =AV <sub>cc</sub>	-0.3 to 5.75	V
V <sub>i</sub>	Input voltage	RESET, CNV <sub>ss</sub> , BYTE, P0 <sub>0</sub> to P0 <sub>7</sub> , P1 <sub>0</sub> to P1 <sub>7</sub> , P2 <sub>0</sub> to P2 <sub>7</sub> , P3 <sub>0</sub> to P3 <sub>7</sub> , P4 <sub>0</sub> to P4 <sub>7</sub> , P5 <sub>0</sub> to P5 <sub>7</sub> , P6 <sub>0</sub> to P6 <sub>7</sub> , P7 <sub>2</sub> to P7 <sub>7</sub> , P8 <sub>0</sub> to P8 <sub>7</sub> , P9 <sub>0</sub> to P9 <sub>7</sub> , P10 <sub>0</sub> to P10 <sub>7</sub> , VREF, XIN, HOR, VERT		-0.3 to V <sub>cc</sub> +0.3	V
		P7 <sub>0</sub> , P7 <sub>1</sub>		-0.3 to 5.75	V
V <sub>o</sub>	Output voltage	P0 <sub>0</sub> to P0 <sub>7</sub> , P1 <sub>0</sub> to P1 <sub>7</sub> , P2 <sub>0</sub> to P2 <sub>7</sub> , P3 <sub>0</sub> to P3 <sub>7</sub> , P4 <sub>0</sub> to P4 <sub>7</sub> , P5 <sub>0</sub> to P5 <sub>7</sub> , P6 <sub>0</sub> to P6 <sub>7</sub> , P7 <sub>2</sub> to P7 <sub>7</sub> , P8 <sub>0</sub> to P8 <sub>4</sub> , P8 <sub>6</sub> , P8 <sub>7</sub> , P9 <sub>0</sub> to P9 <sub>7</sub> , P10 <sub>0</sub> to P10 <sub>7</sub> , XOUT, P11 <sub>0</sub> to P11 <sub>8</sub>		-0.3 to V <sub>cc</sub> +0.3	V
		P7 <sub>0</sub> , P7 <sub>1</sub>		-0.3 to 5.75	V
P <sub>d</sub>	Power dissipation		T <sub>a</sub> =25 °C	1000	mW
T <sub>opr</sub>	Operating ambient temperature			-20 to 70	°C
T <sub>stg</sub>	Storage temperature			-40 to 125	°C

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Tabl 4.2 Recommended operating conditions (referenced to Vcc = 4.75V to 5.25V at Ta = – 20 to 70°C unless otherwise specified)**

Symbol	Parameter		Standard			Unit
			Min	Typ.	Max.	
Vcc	Supply voltage		4.75	5.0	5.25	V
AVcc	Analog supply voltage			Vcc		V
Vss	Supply voltage			0		V
AVss	Analog supply voltage			0		V
VIH	HIGH input voltage	P31 to P37, P40 to P47, P50 to P57, P60 to P67, P70 to P77, P80 to P87, P90 to P97, P100 to P107, XIN, RESET, CNVss, BYTE, HOR, VERT	0.8Vcc		Vcc	V
		P00 to P07, P10 to P17, P20 to P27, P30	0.5Vcc		Vcc	V
VIL	LOW input voltage	P31 to P37, P40 to P47, P50 to P57, P60 to P67, P70 to P77, P80 to P87, P90 to P97, P100 to P107, XIN, RESET, CNVss, BYTE, HOR, VERT	0		0.2Vcc	V
		P00 to P07, P10 to P17, P20 to P27, P30	0		0.16Vcc	V
VCVIN	Composite video input voltage CVIN1, CVIN2			2V P-P		V
VFSCIN	Input voltage FSCIN(Note 1)		0.3V P-P		4.0V P-P	V
IOH (peak)	HIGH peak output current (Note 2.3)	P00 to P07, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P60 to P67, P72 to P77, P80 to P84, P86, P87, P90 to P97, P100 to P107, P110 to P118			-10.0	mA
IOH (avg)	HIGH average output current	P00 to P07, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P60 to P67, P72 to P77, P80 to P84, P86, P87, P90 to P97, P100 to P107, P110 to P118			-5.0	mA
IOL (peak)	LOW peak output current	P00 to P07, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P60 to P67, P72 to P77, P80 to P84, P86, P87, P90 to P97, P100 to P107, P110 to P118			10.0	mA
IOL (avg)	LOW average output current	P00 to P07, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P60 to P67, P72 to P77, P80 to P84, P86, P87, P90 to P97, P100 to P107, P110 to P118			5.0	mA
f (XIN)	Main clock input oscillation frequency	No wait with wait Vcc=4.75V to 5.25V	0		10	MHz
f (XCIN)	Subclock oscillation frequency			32.768	50	kHz
f (FSCIN)	Oscillation frequency for synchronous signal(Duty 40% to 60%)			4.434		MHz

Note 1: Noise component is within 30mV.

Note 2: The mean output current is the mean value within 100ms.

Note 3: The total IOL (peak) for ports P0, P1, P2, P86, P87, P9, and P10 must be 80mA max. The total IOH (peak) for ports P0, P1, P2, P86, P87, P9, and P10 must be 80mA max. The total IOL (peak) for ports P3, P4, P5, P6, P7, and P80 to P84 must be 80mA max. The total IOH (peak) for ports P3, P4, P5, P6, P72 to P77, and P80 to P84 must be 80mA max.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Table 4.3 Electrical characteristics (referenced to  $V_{CC} = 5V$ ,  $V_{SS} = 0V$  at  $T_a = 25^\circ C$ ,  $f(X_{IN}) = 10MHz$  unless otherwise specified)**

Symbol	Parameter		Measuring condition	Standard			Unit
				Min	Typ.	Max.	
$V_{OH}$	HIGH output voltage	P0 <sub>0</sub> to P0 <sub>7</sub> , P1 <sub>0</sub> to P1 <sub>7</sub> , P2 <sub>0</sub> to P2 <sub>7</sub> , P3 <sub>0</sub> to P3 <sub>7</sub> , P4 <sub>0</sub> to P4 <sub>7</sub> , P5 <sub>0</sub> to P5 <sub>7</sub> , P6 <sub>0</sub> to P6 <sub>7</sub> , P7 <sub>2</sub> to P7 <sub>7</sub> , P8 <sub>0</sub> to P8 <sub>4</sub> , P8 <sub>6</sub> , P8 <sub>7</sub> , P9 <sub>0</sub> to P9 <sub>7</sub> , P10 <sub>0</sub> to P10 <sub>7</sub> , P11 <sub>0</sub> to P11 <sub>8</sub>	$I_{OH} = -5mA$	3.0			V
$V_{OH}$	HIGH output voltage	P0 <sub>0</sub> to P0 <sub>7</sub> , P1 <sub>0</sub> to P1 <sub>7</sub> , P2 <sub>0</sub> to P2 <sub>7</sub> , P3 <sub>0</sub> to P3 <sub>7</sub> , P4 <sub>0</sub> to P4 <sub>7</sub> , P5 <sub>0</sub> to P5 <sub>7</sub> , P6 <sub>0</sub> to P6 <sub>7</sub> , P7 <sub>2</sub> to P7 <sub>7</sub> , P8 <sub>0</sub> to P8 <sub>4</sub> , P8 <sub>6</sub> , P8 <sub>7</sub> , P9 <sub>0</sub> to P9 <sub>7</sub> , P10 <sub>0</sub> to P10 <sub>7</sub> , P11 <sub>0</sub> to P11 <sub>8</sub>	$I_{OH} = -200\mu A$	4.7			V
$V_{OH}$	HIGH output voltage	LP1 to LP4	$V_{CC} = 4.75V$ , $I_{OH} = -0.5mA$	3.75			V
$V_{OH}$	HIGH output voltage	$X_{OUT}$	HIGHPOWER	$I_{OH} = -1mA$	3.0		V
			LOWPOWER	$I_{OH} = -0.5mA$	3.0		
	HIGH output voltage	$X_{COUT}$	HIGHPOWER	With no load applied		3.0	V
			LOWPOWER	With no load applied		1.6	
$V_{OL}$	LOW output voltage	P0 <sub>0</sub> to P0 <sub>7</sub> , P1 <sub>0</sub> to P1 <sub>7</sub> , P2 <sub>0</sub> to P2 <sub>7</sub> , P3 <sub>0</sub> to P3 <sub>7</sub> , P4 <sub>0</sub> to P4 <sub>7</sub> , P5 <sub>0</sub> to P5 <sub>7</sub> , P6 <sub>0</sub> to P6 <sub>7</sub> , P7 <sub>0</sub> to P7 <sub>7</sub> , P8 <sub>0</sub> to P8 <sub>4</sub> , P8 <sub>6</sub> , P8 <sub>7</sub> , P9 <sub>0</sub> to P9 <sub>7</sub> , P10 <sub>0</sub> to P10 <sub>7</sub> , P11 <sub>0</sub> to P11 <sub>8</sub>	$I_{OL} = 5mA$			2.0	V
$V_{OL}$	LOW output voltage	P0 <sub>0</sub> to P0 <sub>7</sub> , P1 <sub>0</sub> to P1 <sub>7</sub> , P2 <sub>0</sub> to P2 <sub>7</sub> , P3 <sub>0</sub> to P3 <sub>7</sub> , P4 <sub>0</sub> to P4 <sub>7</sub> , P5 <sub>0</sub> to P5 <sub>7</sub> , P6 <sub>0</sub> to P6 <sub>7</sub> , P7 <sub>0</sub> to P7 <sub>7</sub> , P8 <sub>0</sub> to P8 <sub>4</sub> , P8 <sub>6</sub> , P8 <sub>7</sub> , P9 <sub>0</sub> to P9 <sub>7</sub> , P10 <sub>0</sub> to P10 <sub>7</sub> , P11 <sub>0</sub> to P11 <sub>8</sub>	$I_{OL} = 200\mu A$			0.45	V
$V_{OL}$	LOW output voltage	LP1 to LP4	$V_{CC} = 4.75V$ , $I_{OH} = -0.5mA$			0.4	V
$V_{OL}$	LOW output voltage	$X_{OUT}$	HIGHPOWER	$I_{OL} = 1mA$		2.0	V
			LOWPOWER	$I_{OL} = 0.5mA$		2.0	
	LOW output voltage	$X_{COUT}$	HIGHPOWER	With no load applied		0	V
			LOWPOWER	With no load applied		0	
$V_{T+} - V_{T-}$	Hysteresis	HOLD, RDY, TA0 <sub>IN</sub> to TA4 <sub>IN</sub> , TB0 <sub>IN</sub> to TB2 <sub>IN</sub> , INT <sub>0</sub> to INT <sub>5</sub> , ADTRG, CTS <sub>1</sub> , CLK <sub>1</sub> , NMI, TA2 <sub>OUT</sub> to TA4 <sub>OUT</sub> , K <sub>10</sub> to K <sub>13</sub>		0.2		0.8	V
$V_{T+} - V_{T-}$	Hysteresis	CTS <sub>0</sub> , CLK <sub>0</sub>		0.2		1.4	V
$V_{T+} - V_{T-}$	Hysteresis	RESET		0.2		1.8	V
$I_{IH}$	HIGH input current	P0 <sub>0</sub> to P0 <sub>7</sub> , P1 <sub>0</sub> to P1 <sub>7</sub> , P2 <sub>0</sub> to P2 <sub>7</sub> , P3 <sub>0</sub> to P3 <sub>7</sub> , P4 <sub>0</sub> to P4 <sub>7</sub> , P5 <sub>0</sub> to P5 <sub>7</sub> , P6 <sub>0</sub> to P6 <sub>7</sub> , P7 <sub>0</sub> to P7 <sub>7</sub> , P8 <sub>0</sub> to P8 <sub>7</sub> , P9 <sub>0</sub> to P9 <sub>7</sub> , P10 <sub>0</sub> to P10 <sub>7</sub> , X <sub>IN</sub> , RESET, CNV <sub>SS</sub> , BYTE, HOR, VERT	$V_I = 5V$			5.0	$\mu A$
$I_{IL}$	LOW input current	P0 <sub>0</sub> to P0 <sub>7</sub> , P1 <sub>0</sub> to P1 <sub>7</sub> , P2 <sub>0</sub> to P2 <sub>7</sub> , P3 <sub>0</sub> to P3 <sub>7</sub> , P4 <sub>0</sub> to P4 <sub>7</sub> , P5 <sub>0</sub> to P5 <sub>7</sub> , P6 <sub>0</sub> to P6 <sub>7</sub> , P7 <sub>0</sub> to P7 <sub>7</sub> , P8 <sub>0</sub> to P8 <sub>7</sub> , P9 <sub>0</sub> to P9 <sub>7</sub> , P10 <sub>0</sub> to P10 <sub>7</sub> , X <sub>IN</sub> , RESET, CNV <sub>SS</sub> , BYTE, HOR, VERT	$V_I = 0V$			-5.0	$\mu A$
$R_{PULLUP}$	Pull-up resistance	P0 <sub>0</sub> to P0 <sub>7</sub> , P1 <sub>0</sub> to P1 <sub>7</sub> , P2 <sub>0</sub> to P2 <sub>7</sub> , P3 <sub>0</sub> to P3 <sub>7</sub> , P4 <sub>0</sub> to P4 <sub>7</sub> , P5 <sub>0</sub> to P5 <sub>7</sub> , P6 <sub>0</sub> to P6 <sub>7</sub> , P7 <sub>2</sub> to P7 <sub>7</sub> , P8 <sub>0</sub> to P8 <sub>4</sub> , P8 <sub>6</sub> , P8 <sub>7</sub> , P9 <sub>0</sub> to P9 <sub>7</sub> , P10 <sub>0</sub> to P10 <sub>7</sub>	$V_I = 0V$	30.0	50.0	167.0	k $\Omega$
$V_{SYNCIN}$	Sync voltage amplitude			0.3	0.6	1.2	V
$V_{dat(text)}$	Teletext data voltage amplitude			0.6	0.9	1.4	V
$\Delta f / f$	Range for display oscillator circuit			$\pm 7$			%
$f_H$	Horizontal synchronous signal frequency			14.6	15.625	17.0	kHz



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Table 4.4 Electrical characteristics (referenced to  $V_{CC} = 5V$ ,  $V_{SS} = 0V$  at  $T_a = 25^\circ C$ ,  $f(X_{IN}) = 10MHz$  unless otherwise specified)**

Symbol	Parameter	Measuring condition	Standard			Unit
			Min	Typ.	Max.	
$R_{fXIN}$	Feedback resistance $X_{IN}$			1.0		$M\Omega$
$R_{fXCIN}$	Feedback resistance $X_{CIN}$			6.0		$M\Omega$
$V_{RAM}$	RAM retention voltage	When clock is stopped	2.0			V
$I_{CC}$	Power supply current	When OSD operate, $f(X_{IN})=10MHz$		150	180	mA
		When clock is stopped			3	mA

**Tabl 4.5 Video signal input conditions ( $V_{CC} = 5.0V$ ,  $T_a = -20$  to  $70^\circ C$ )**

Symbol	Parameter	Measuring condition	Standard			Unit
			Min	Typ.	Max.	
$V_{IN-cu}$	Composite video signal input clamp voltage	Sync-chip voltage		1.0		V

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Table 4.6 A-D conversion characteristics (referenced to  $V_{CC} = AV_{CC} = V_{REF} = 5V$ ,  $V_{SS} = AV_{SS} = 0V$  at  $T_a = 25^\circ C$ ,  $f(X_{IN}) = 10MHz$  unless otherwise specified)**

Symbol	Parameter		Measuring condition	Standard			Unit
				Min.	Typ.	Max.	
—	Resolution		$V_{REF} = V_{CC}$			8	Bits
—	Absolute accuracy	Sample & hold function not available	$V_{REF} = V_{CC} = 5V$			$\pm 3$	LSB
		Sample & hold function available(8bit)	$V_{REF} = V_{CC} = 5V$			$\pm 2$	LSB
RLADDER	Ladder resistance		$V_{REF} = V_{CC}$	10		40	k $\Omega$
tCONV	Conversion time(8bit)			2.8			$\mu s$
tsAMP	Sampling time			0.3			$\mu s$
VREF	Reference voltage			2		$V_{CC}$	V
VIA	Analog input voltage			0		$V_{REF}$	V

**Table 4.7 D-A conversion characteristics (referenced to  $V_{CC} = 5V$ ,  $V_{SS} = AV_{SS} = 0V$ ,  $V_{REF} = 5V$  at  $T_a = 25^\circ C$ ,  $f(X_{IN}) = 10MHz$  unless otherwise specified)**

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
—	Resolution				8	Bits
—	Absolute accuracy				1.0	%
t <sub>SU</sub>	Setup time				3	$\mu s$
R <sub>O</sub>	Output resistance		4	10	20	k $\Omega$
I <sub>VREF</sub>	Reference power supply input current	(Note)			1.5	mA

Note: This applies when using one D-A converter, with the D-A register for the unused D-A converter set to "0016".

The A-D converter's ladder resistance is not included.

Also, when the Vref is unconnected at the A-D control register, IVREF is sent.

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Timing requirements (referenced to V<sub>CC</sub> = 5V, V<sub>SS</sub> = 0V at Ta = 25°C unless otherwise specified)

**Table 4.8 External clock input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub>	External clock input cycle time	100		ns
t <sub>w(H)</sub>	External clock input HIGH pulse width	40		ns
t <sub>w(L)</sub>	External clock input LOW pulse width	40		ns
t <sub>r</sub>	External clock rise time		18	ns
t <sub>f</sub>	External clock fall time		18	ns

**Table 4.9 RDY, HOLD, HLDA input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>ac1</sub> (RD-DB)	Data input access time (no wait)		(Note)	ns
t <sub>ac2</sub> (RD-DB)	Data input access time (with wait)		(Note)	ns
t <sub>ac3</sub> (RD-DB)	Data input access time (when accessing multiplex bus area)		(Note)	ns
t <sub>su</sub> (DB-RD)	Data input setup time	40		ns
t <sub>su</sub> (RDY-BCLK)	RDY input setup time	30		ns
t <sub>su</sub> (HOLD-BCLK)	HOLD input setup time	40		ns
t <sub>h</sub> (RD-DB)	Data input hold time	0		ns
t <sub>h</sub> (BCLK-RDY)	RDY input hold time	0		ns
t <sub>h</sub> (BCLK-HOLD)	HOLD input hold time	0		ns
t <sub>d</sub> (BCLK-HLDA)	HLDA output delay time		40	ns

Note: Calculated according to the BCLK frequency as follows:

$$t_{ac1}(RD - DB) = \frac{10^9}{f(BCLK) \times 2} - 45 \quad [ns]$$

$$t_{ac2}(RD - DB) = \frac{3 \times 10^9}{f(BCLK) \times 2} - 45 \quad [ns]$$

$$t_{ac3}(RD - DB) = \frac{3 \times 10^9}{f(BCLK) \times 2} - 45 \quad [ns]$$

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Timing requirements (referenced to  $V_{CC} = 5V$ ,  $V_{SS} = 0V$  at  $T_a = 25^{\circ}C$  unless otherwise specified)**

**Table 4.10 Timer A input (counter input in event counter mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_c(TA)$	TAiN input cycle time	100		ns
$t_w(TAH)$	TAiN input HIGH pulse width	40		ns
$t_w(TAL)$	TAiN input LOW pulse width	40		ns

**Table 4.11 Timer A input (gating input in timer mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_c(TA)$	TAiN input cycle time	400		ns
$t_w(TAH)$	TAiN input HIGH pulse width	200		ns
$t_w(TAL)$	TAiN input LOW pulse width	200		ns

**Table 4.12 Timer A input (external trigger input in one-shot timer mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_c(TA)$	TAiN input cycle time	200		ns
$t_w(TAH)$	TAiN input HIGH pulse width	100		ns
$t_w(TAL)$	TAiN input LOW pulse width	100		ns

**Table 4.13 Timer A input (external trigger input in pulse width modulation mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_w(TAH)$	TAiN input HIGH pulse width	100		ns
$t_w(TAL)$	TAiN input LOW pulse width	100		ns

**Table 4.14 Timer A input (up/down input in event counter mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_c(UP)$	TAiOUT input cycle time	2000		ns
$t_w(UPH)$	TAiOUT input HIGH pulse width	1000		ns
$t_w(UPL)$	TAiOUT input LOW pulse width	1000		ns
$t_{su}(UP-TIN)$	TAiOUT input setup time	400		ns
$t_h(TIN-UP)$	TAiOUT input hold time	400		ns

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Timing requirements (referenced to  $V_{CC} = 5V$ ,  $V_{SS} = 0V$  at  $T_a = 25^\circ C$  unless otherwise specified)

**Table 4.15 Timer B input (counter input in event counter mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiN input cycle time (counted on one edge)	100		ns
$t_{w(TBH)}$	TBiN input HIGH pulse width (counted on one edge)	40		ns
$t_{w(TBL)}$	TBiN input LOW pulse width (counted on one edge)	40		ns
$t_{c(TB)}$	TBiN input cycle time (counted on both edges)	200		ns
$t_{w(TBH)}$	TBiN input HIGH pulse width (counted on both edges)	80		ns
$t_{w(TBL)}$	TBiN input LOW pulse width (counted on both edges)	80		ns

**Table 4.16 Timer B input (pulse period measurement mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiN input cycle time	400		ns
$t_{w(TBH)}$	TBiN input HIGH pulse width	200		ns
$t_{w(TBL)}$	TBiN input LOW pulse width	200		ns

**Table 4.17 Timer B input (pulse width measurement mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiN input cycle time	400		ns
$t_{w(TBH)}$	TBiN input HIGH pulse width	200		ns
$t_{w(TBL)}$	TBiN input LOW pulse width	200		ns

**Table 4.18 A-D trigger input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(AD)}$	ADTRG input cycle time (trigger able minimum)	1000		ns
$t_{w(ADL)}$	ADTRG input LOW pulse width	125		ns

**Table 4.19 Serial I/O**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CK)}$	CLKi input cycle time	200		ns
$t_{w(CKH)}$	CLKi input HIGH pulse width	100		ns
$t_{w(CKL)}$	CLKi input LOW pulse width	100		ns
$t_{d(C-Q)}$	TxDi output delay time		80	ns
$t_{h(C-Q)}$	TxDi hold time	0		ns
$t_{su(D-C)}$	RxDi input setup time	30		ns
$t_{h(C-D)}$	RxDi input hold time	90		ns

**Table 4.20 External interrupt  $\overline{INT_i}$  inputs**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(INH)}$	$\overline{INT_i}$ input HIGH pulse width	250		ns
$t_{w(INL)}$	$\overline{INT_i}$ input LOW pulse width	250		ns

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Switching characteristics (referenced to V<sub>CC</sub> = 5V, V<sub>SS</sub> = 0V at Ta = 25°C, CM15 = "1" unless otherwise specified)**

**Table 4.21 No wait**

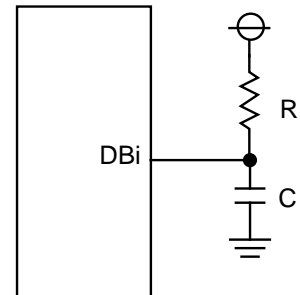
Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
t <sub>d</sub> (BCLK-AD)	Address output delay time	Figure 4.1		25	ns
t <sub>h</sub> (BCLK-AD)	Address output hold time (BCLK standard)		4		ns
t <sub>h</sub> (RD-AD)	Address output hold time (RD standard)		0		ns
t <sub>h</sub> (WR-AD)	Address output hold time (WR standard)		0		ns
t <sub>d</sub> (BCLK-CS)	Chip select output delay time			25	ns
t <sub>h</sub> (BCLK-CS)	Chip select output hold time (BCLK standard)		4		ns
t <sub>d</sub> (BCLK-ALE)	ALE signal output delay time			25	ns
t <sub>h</sub> (BCLK-ALE)	ALE signal output hold time		- 4		ns
t <sub>d</sub> (BCLK-RD)	RD signal output delay time			25	ns
t <sub>h</sub> (BCLK-RD)	RD signal output hold time		0		ns
t <sub>d</sub> (BCLK-WR)	WR signal output delay time			25	ns
t <sub>h</sub> (BCLK-WR)	WR signal output hold time		0		ns
t <sub>d</sub> (BCLK-DB)	Data output delay time (BCLK standard)			40	ns
t <sub>h</sub> (BCLK-DB)	Data output hold time (BCLK standard)		4		ns
t <sub>d</sub> (DB-WR)	Data output delay time (WR standard)		(Note1)		ns
t <sub>h</sub> (WR-DB)	Data output hold time (WR standard)(Note2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

$$t_d(\text{DB} - \text{WR}) = \frac{10^9}{f(\text{BCLK}) \times 2} - 40 \quad [\text{ns}]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.  
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.  
Hold time of data bus is expressed in  
 $t = -CR \times \ln(1 - V_{OL} / V_{CC})$   
by a circuit of the right figure.  
For example, when  $V_{OL} = 0.2V_{CC}$ ,  $C = 30\text{pF}$ ,  $R = 1\text{k}\Omega$ , hold time of output "L" level is

$$t = -30\text{pF} \times 1\text{k}\Omega \times \ln(1 - 0.2V_{CC} / V_{CC}) \\ = 6.7\text{ns}.$$



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

Switching characteristics (refer to  $V_{CC} = 5V$ ,  $V_{SS} = 0V$  at  $T_a = 25^\circ C$ ,  $CM15 = "1"$  unless otherwise specified)

Table 4.22 With wait, accessing external memory

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_d(BCLK-AD)$	Address output delay time	Figure 4.1		25	ns
$t_h(BCLK-AD)$	Address output hold time (BCLK standard)		4		ns
$t_h(RD-AD)$	Address output hold time (RD standard)		0		ns
$t_h(WR-AD)$	Address output hold time (WR standard)		0		ns
$t_d(BCLK-CS)$	Chip select output delay time			25	ns
$t_h(BCLK-CS)$	Chip select output hold time (BCLK standard)		4		ns
$t_d(BCLK-ALE)$	ALE signal output delay time			25	ns
$t_h(BCLK-ALE)$	ALE signal output hold time		-4		ns
$t_d(BCLK-RD)$	RD signal output delay time			25	ns
$t_h(BCLK-RD)$	RD signal output hold time		0		ns
$t_d(BCLK-WR)$	WR signal output delay time			25	ns
$t_h(BCLK-WR)$	WR signal output hold time		0		ns
$t_d(BCLK-DB)$	Data output delay time (BCLK standard)			40	ns
$t_h(BCLK-DB)$	Data output hold time (BCLK standard)		4		ns
$t_d(DB-WR)$	Data output delay time (WR standard)		(Note1)		ns
$t_h(WR-DB)$	Data output hold time (WR standard)(Note2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

$$t_d(DB - WR) = \frac{10^9}{f(BCLK)} - 40 \quad [ns]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.  
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.

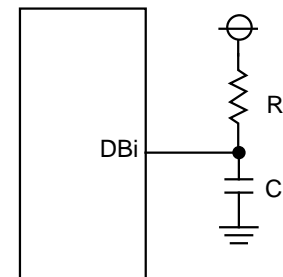
Hold time of data bus is expressed in

$$t = -CR \times \ln(1 - V_{OL} / V_{CC})$$

by a circuit of the right figure.

For example, when  $V_{OL} = 0.2V_{CC}$ ,  $C = 30pF$ ,  $R = 1k\Omega$ , hold time of output "L" level is

$$t = -30pF \times 1k\Omega \times \ln(1 - 0.2V_{CC} / V_{CC}) \\ = 6.7ns.$$



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

**Switching characteristics (referenced to  $V_{CC} = 5V$ ,  $V_{SS} = 0V$  at  $T_a = 25^\circ C$ ,  $CM15 = "1"$  unless otherwise specified)**

**Table 4.23 With wait, accessing external memory, multiplex bus area selected**

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_d(BCLK-AD)$	Address output delay time	Figure 4.1		25	ns
$t_h(BCLK-AD)$	Address output hold time (BCLK standard)		4		ns
$t_h(RD-AD)$	Address output hold time (RD standard)		(Note)		ns
$t_h(WR-AD)$	Address output hold time (WR standard)		(Note)		ns
$t_d(BCLK-CS)$	Chip select output delay time			25	ns
$t_h(BCLK-CS)$	Chip select output hold time (BCLK standard)		4		ns
$t_h(RD-CS)$	Chip select output hold time (RD standard)		(Note)		ns
$t_h(WR-CS)$	Chip select output hold time (WR standard)		(Note)		ns
$t_d(BCLK-RD)$	RD signal output delay time			25	ns
$t_h(BCLK-RD)$	RD signal output hold time		0		ns
$t_d(BCLK-WR)$	WR signal output delay time			25	ns
$t_h(BCLK-WR)$	WR signal output hold time		0		ns
$t_d(BCLK-DB)$	Data output delay time (BCLK standard)			40	ns
$t_h(BCLK-DB)$	Data output hold time (BCLK standard)		4		ns
$t_d(DB-WR)$	Data output delay time (WR standard)		(Note)		ns
$t_h(WR-DB)$	Data output hold time (WR standard)		(Note)		ns
$t_d(BCLK-ALE)$	ALE signal output delay time (BCLK standard)			25	ns
$t_h(BCLK-ALE)$	ALE signal output hold time (BCLK standard)		- 4		ns
$t_d(AD-ALE)$	ALE signal output delay time (Address standard)		(Note)		ns
$t_h(ALE-AD)$	ALE signal output hold time (Address standard)		50		ns
$t_d(AD-RD)$	Post-address RD signal output delay time		0		ns
$t_d(AD-WR)$	Post-address WR signal output delay time		0		ns
$t_d(RD-AD)$	Address output floating start time			8	ns

Note: Calculated according to the BCLK frequency as follows:

$$t_h(RD - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$t_h(WR - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$t_h(RD - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$t_h(WR - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$t_d(DB - WR) = \frac{10^9 \times 3}{f(BCLK) \times 2} - 40 \quad [ns]$$

$$t_h(WR - DB) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$t_d(AD - ALE) = \frac{10^9}{f(BCLK) \times 2} - 25 \quad [ns]$$



# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

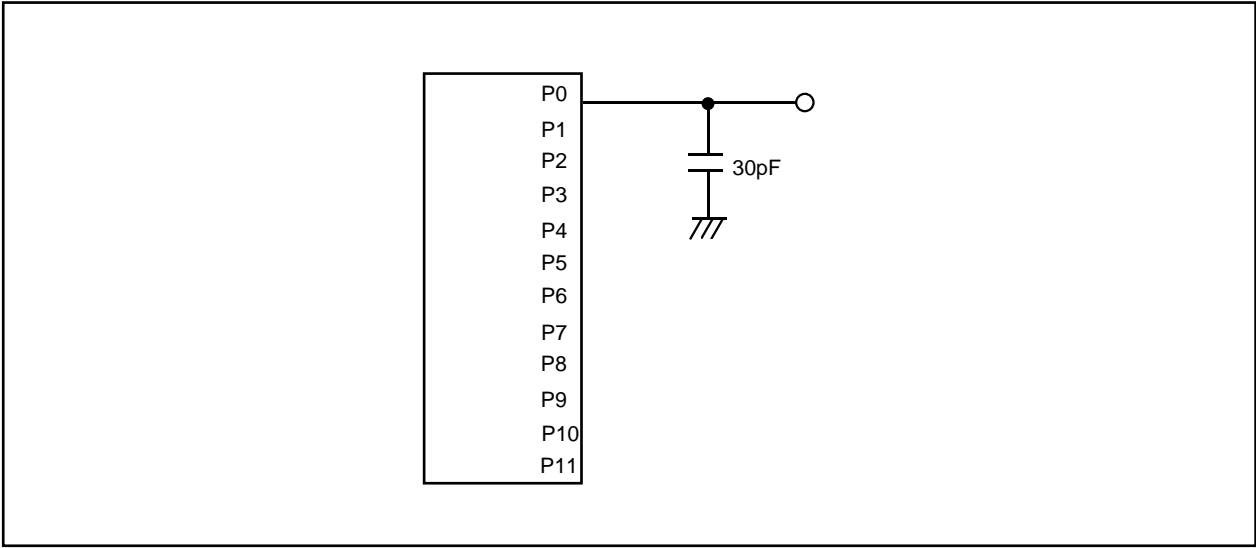


Figure 4.1 Port P0 to P11 measurement circuit

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

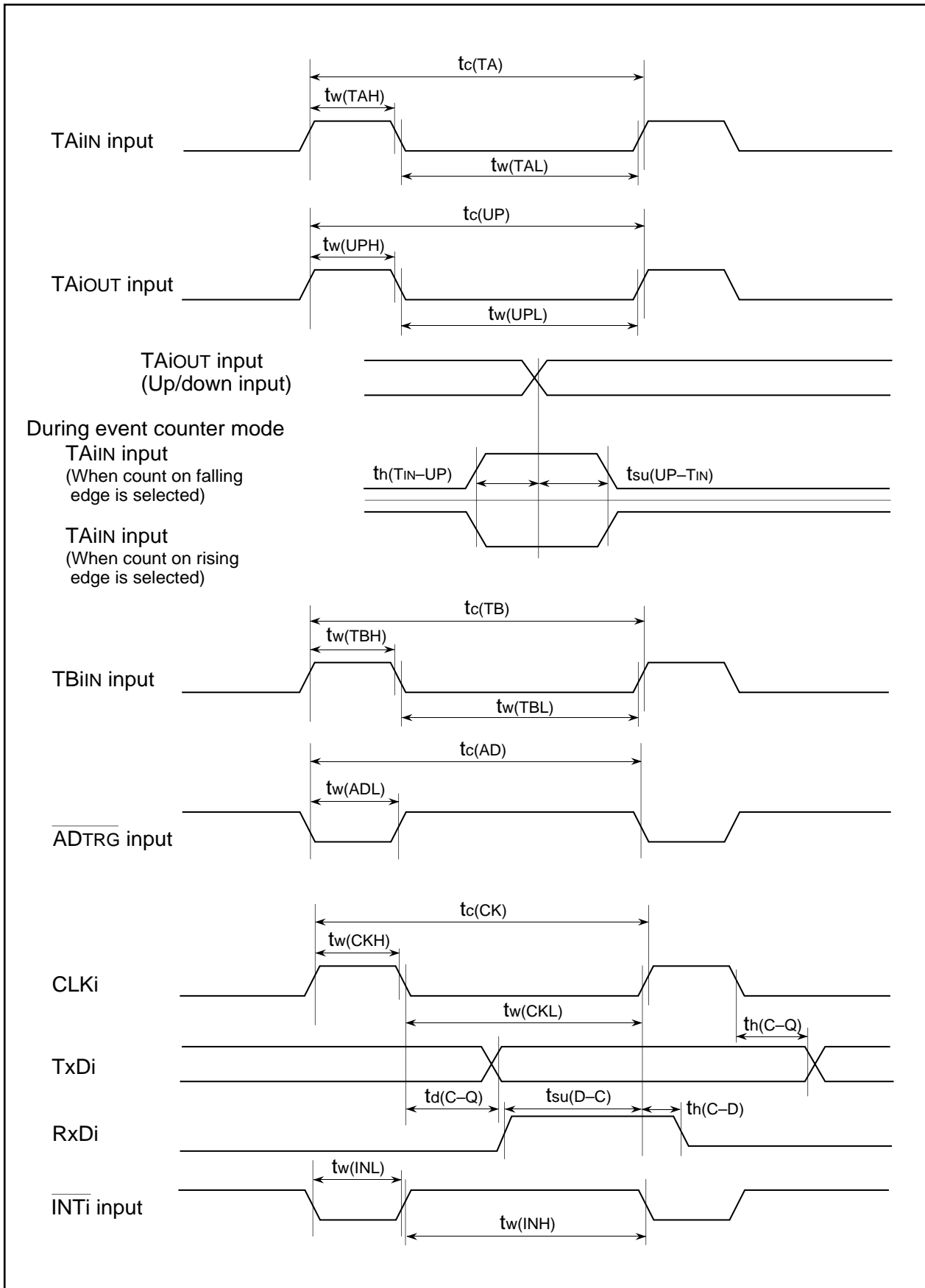
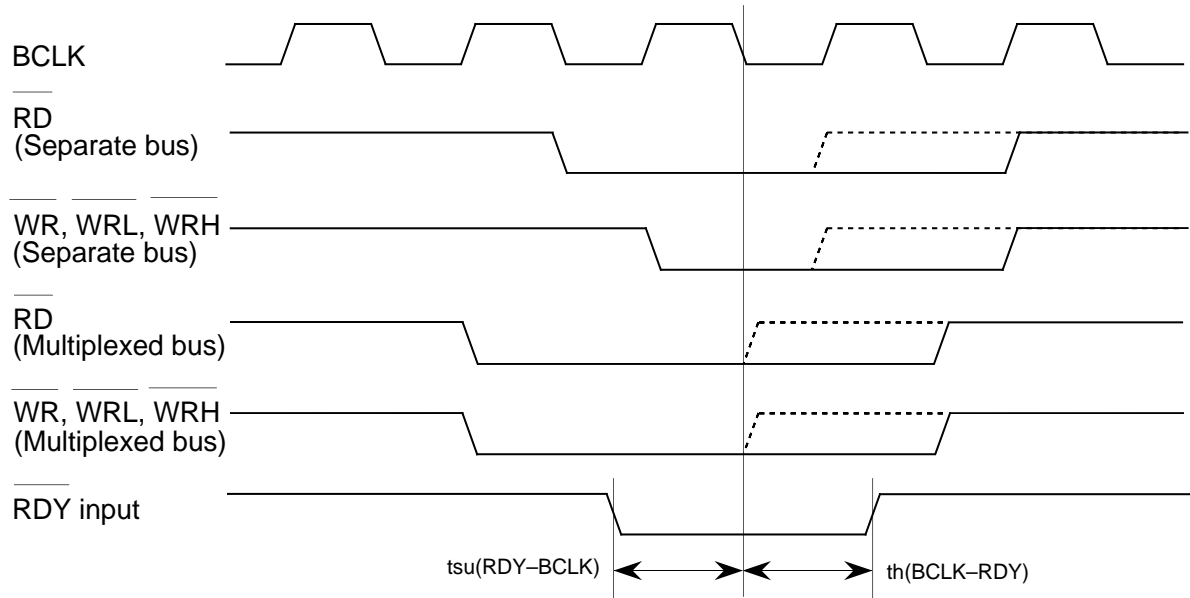


Figure 4.2 Timing diagram (1)

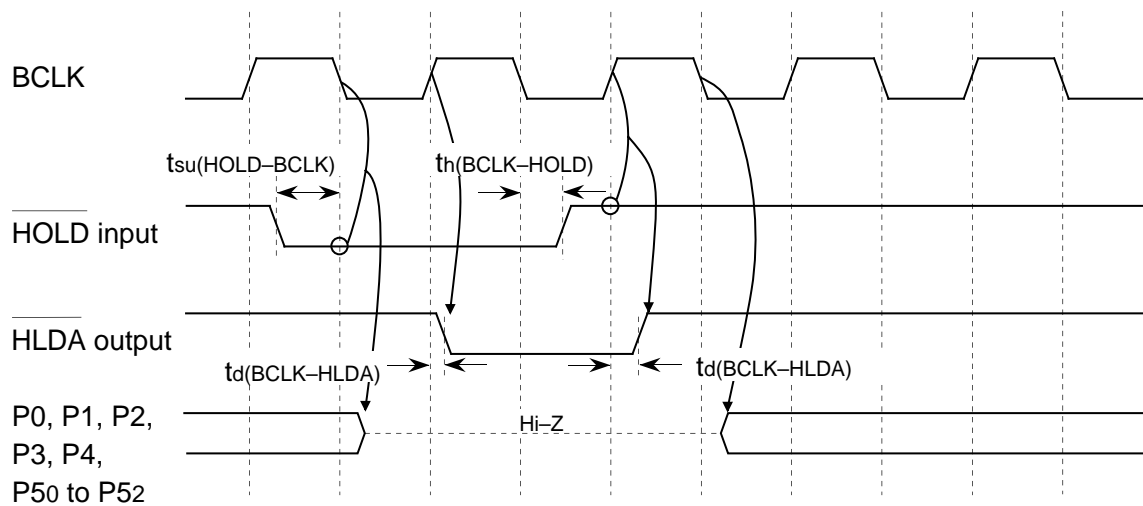
# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## Valid Only With Wait



## Valid With Or Without Wait



Note: The above pins are set to high-impedance regardless of the input level of the BYTE pin and bit (PM06) of processor mode register 0 selects the function of ports P40 to P43.

Measuring conditions :

- $V_{CC}=5V$
- Input timing voltage : Determined with  $V_{IL}=1.0V$ ,  $V_{IH}=4.0V$
- Output timing voltage : Determined with  $V_{OL}=2.5V$ ,  $V_{OH}=2.5V$

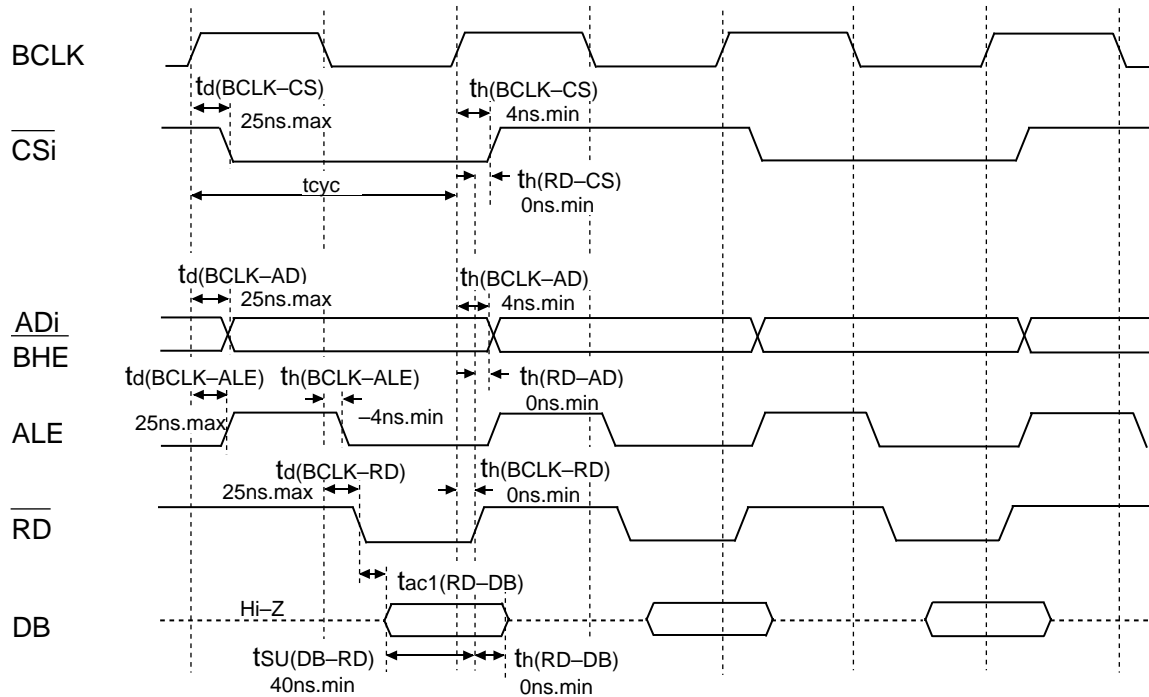
Figure 4.3 Timing diagram (2)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## With No Wait

### Read timing



### Write timing

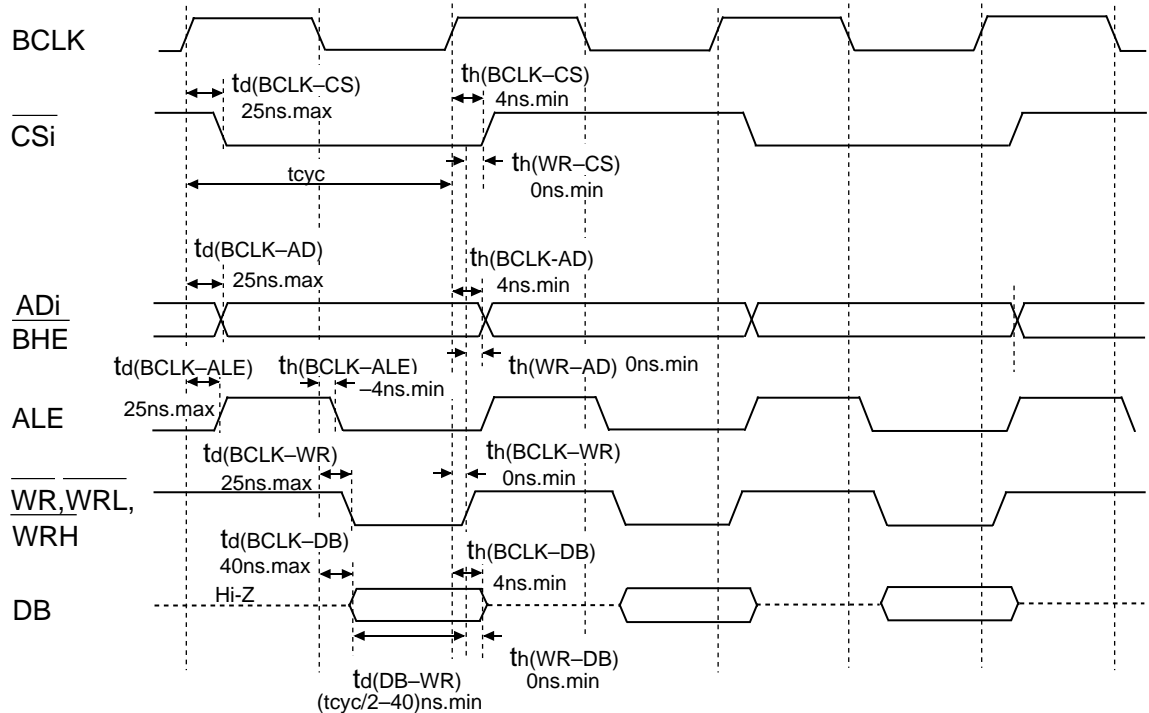


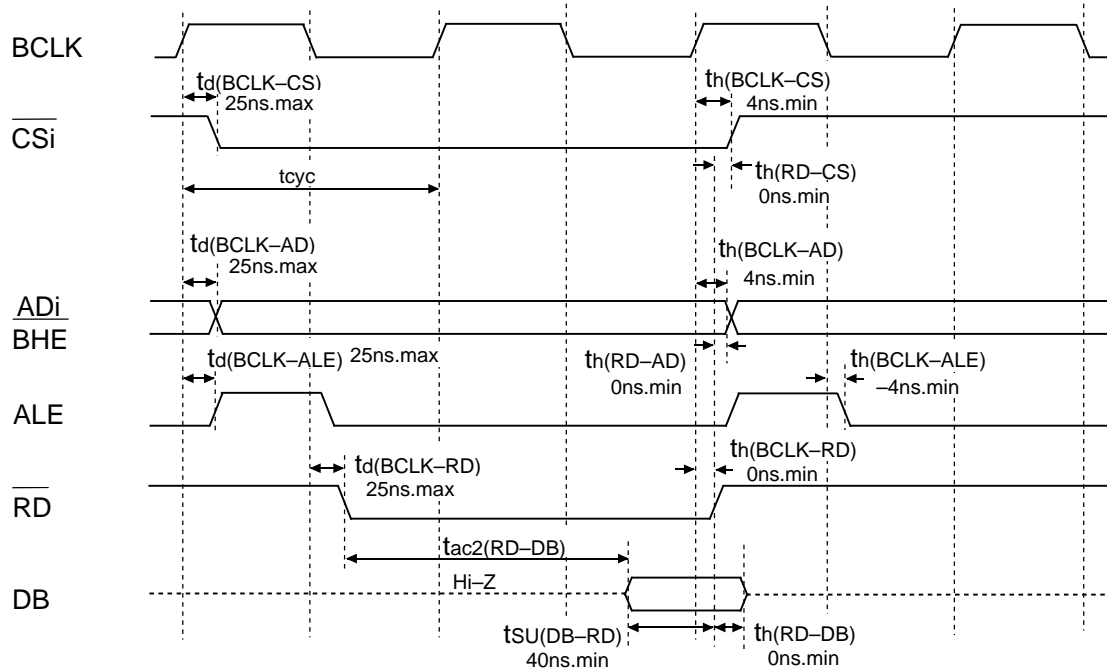
Figure 4.4 Timing diagram (3)

# M306H1SFP

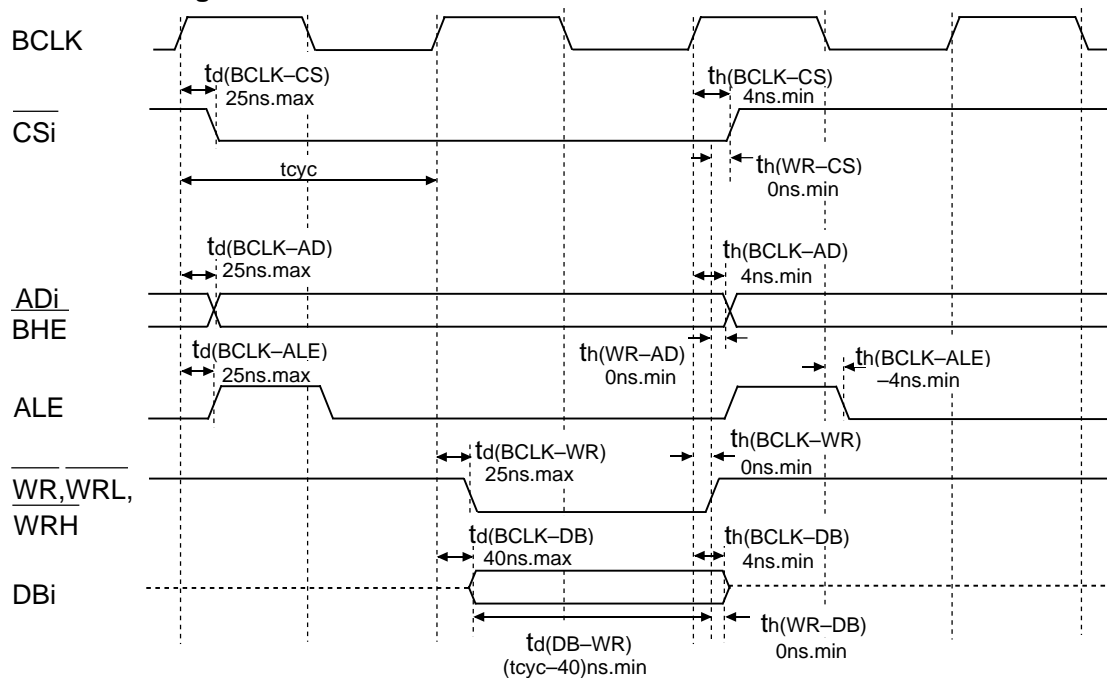
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## When Accessing External Memory Area With Wait

### Read timing



### Write timing



Measuring conditions :

- VCC=5V
- Input timing voltage : Determined with:  $V_{IL}=0.8V$ ,  $V_{IH}=2.5V$
- Output timing voltage : Determined with:  $V_{OL}=0.8V$ ,  $V_{OH}=2.0V$

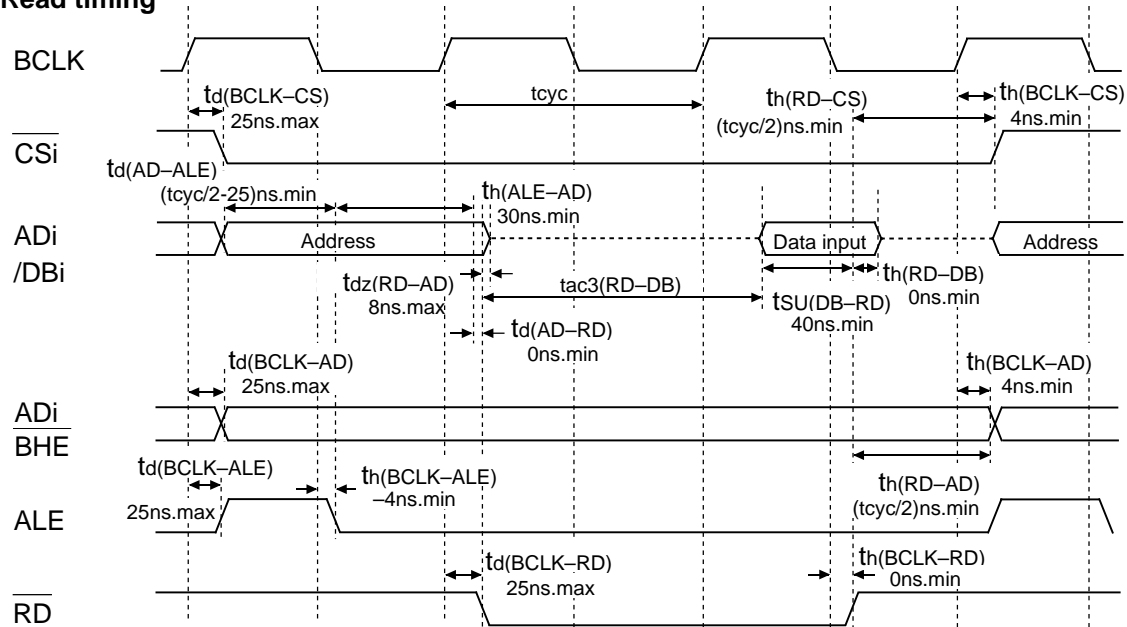
Figure 4.5 Timing diagram (4)

## M306H1SFP

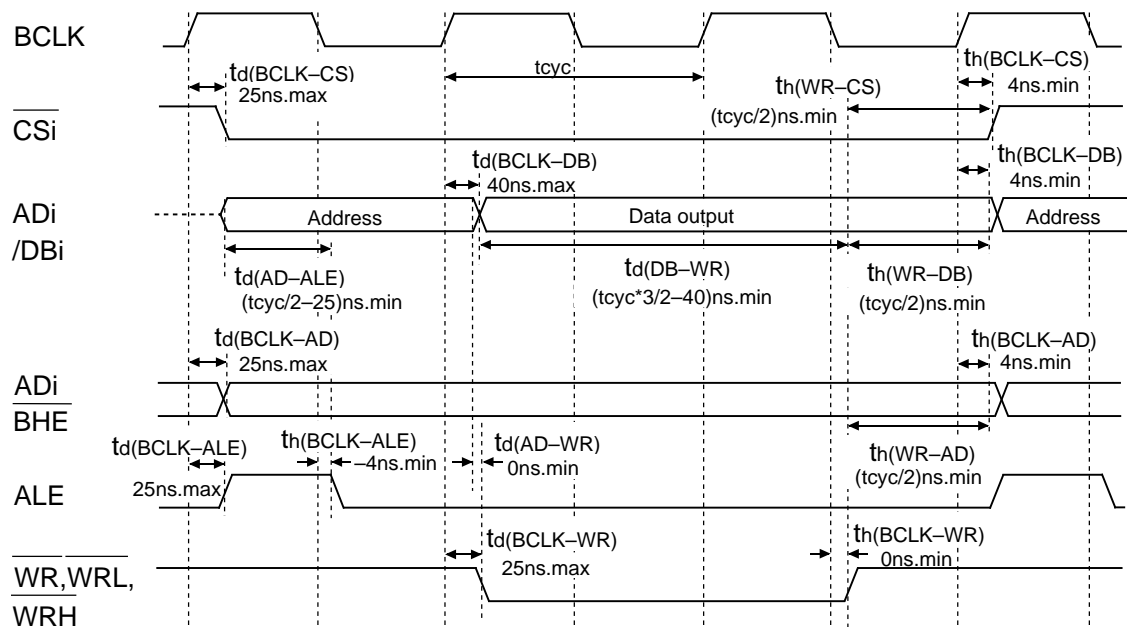
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

### When Accessing External Memory Area With Wait, And Select Multiplexed bus

#### Read timing



#### Write timing



Measuring conditions :

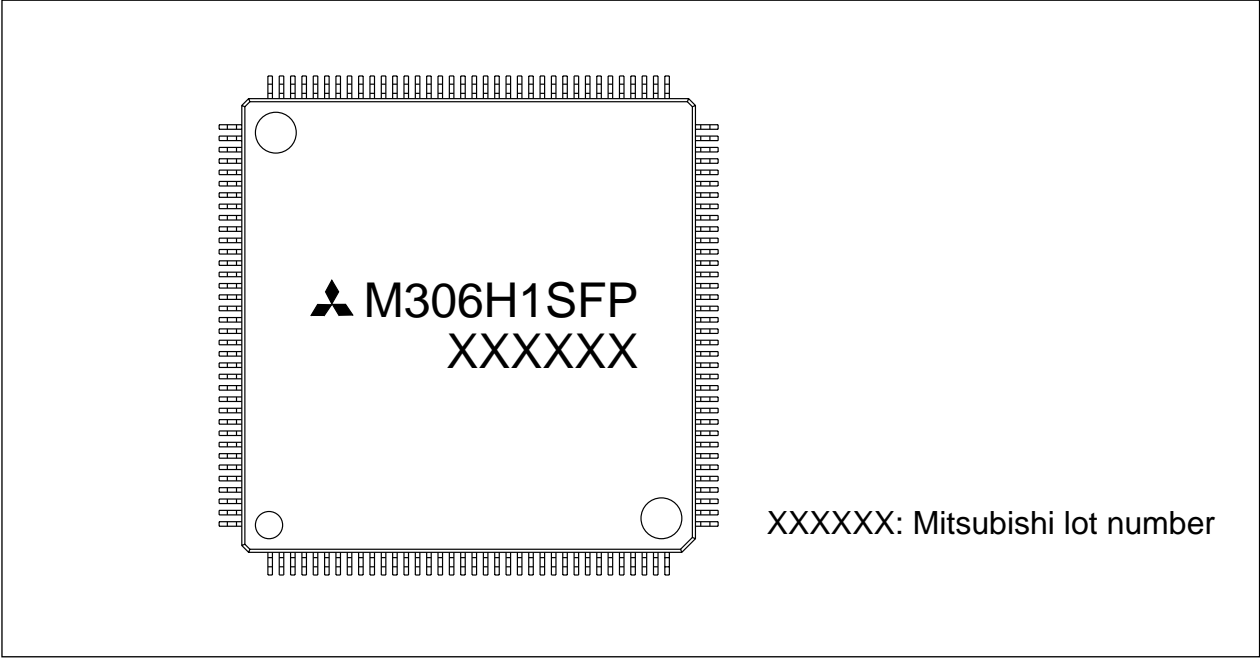
- $V_{CC}=5V$
- Input timing voltage : Determined with  $V_{IL}=0.8V$ ,  $V_{IH}=2.5V$
- Output timing voltage : Determined with  $V_{OL}=0.8V$ ,  $V_{OH}=2.0V$

Figure 4.6 Timing diagram (5)

# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

## 5. Marking Figure



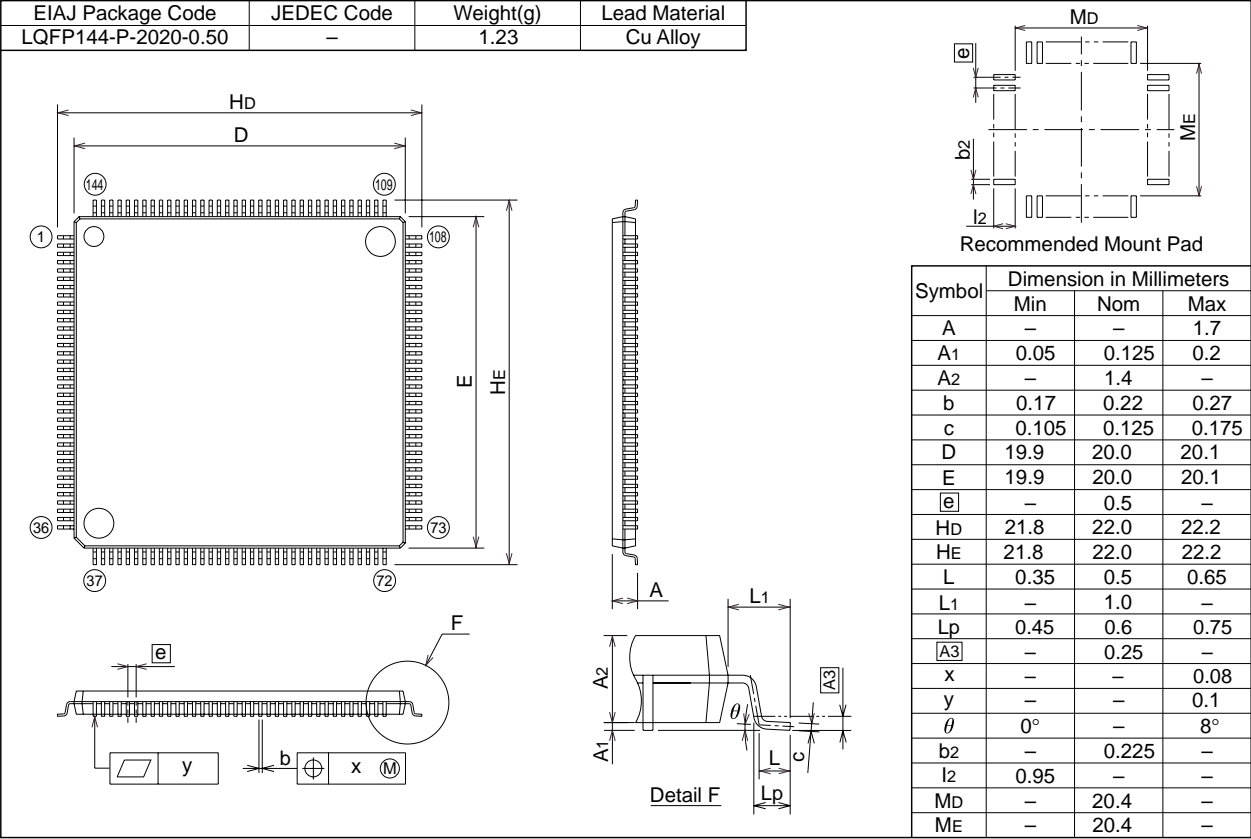
M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

6. Package Outline

144P6Q-A

Plastic 144pin 20X20mm body LQFP





# M306H1SFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER  
with DATA SLICER and ON-SCREEN DISPLAY CONTROLLER

---



#### 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, programs, algorithms, or circuit application examples contained in these materials.
- All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Mitsubishi Electric Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor for the latest product information before purchasing a product listed herein.  
The information described here may contain technical inaccuracies or typographical errors. Mitsubishi Electric Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors. Please also pay attention to information published by Mitsubishi Electric Corporation by various means, including the Mitsubishi Semiconductor home page (<http://www.mitsubishichips.com>).
- When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Mitsubishi Electric Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
- Mitsubishi Electric Corporation semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
- The prior written approval of Mitsubishi Electric Corporation is necessary to reprint or reproduce in whole or in part these materials.
- If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.  
Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
- Please contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor for further details on these materials or the products contained therein.

## REVISION HISTORY

## M306H1SFP (Rev.1.1) DATA SHEET

Rev. No.	Revision Description	Rev. date
1.0	PDF First Edition	0006
1.1	<ul style="list-style-type: none"><li>• Expansion register construction corrected (28) Address 1B<sub>16</sub> ( = DA5 to 0) (page 195) (29) Address 1C<sub>16</sub> ( = DA5 to 0) (page 195) (34) Address 21<sub>16</sub> ( = DA5 to 0) (page 197) (35) Address 22<sub>16</sub> ( = DA5 to 0) (page 197)</li><li>• The change of the page layout Usage precaution (page 219 and 220)</li></ul>	0010