FUJITSU SEMICONDUCTOR
DATA SHEET

Version 1.3

16-bit Proprietary Microcontroller

CMOS

F²MC-16LX MB90370 Series

MB90372/F372/V370

■ DESCRIPTION

The MB90370 series is a line of general-purpose, 16-bit microcontrollers designed for those applications which require high-speed real-time processing. The instruction set is designed to be optimized for controller applications which inheriting the AT architecture of F²MC-16LX series and allow a wide range of control tasks to be processed efficiently at high speed.

A built-in LPC interface, serial IRQ and PS/2 interface simplifies communication with host CPU and PS/2 devices in computer system. Moreover, SMbus compliant I²C, comparator for battery control and A/D converter implements the smart battery control. With these features, the MB90370 series matches itself as keyboard controller with smart battery control.

While inheriting the AT architecture of the F²MC*1 family, the instruction set for the F²MC-16LX CPU core of the MB90370 series incorporates additional instructions for high-level languages, supports extended addressing modes, and contains enhanced multiplication and division instructions as well as a substantial collection of improved bit manipulation instructions. In addition, the MB90370 has an on-chip 32-bit accumulator which enables processing of long-word data.

Notes: *1: F²MC stands for FUJITSU Flexible Microcontroller, a registered trademark of FUJITSU LIMITED.

*2: Purchase of Fujitsu I²C components conveys a license under the Philips I²C Patent Rights to use these components in an I²C system, provided that the system conforms to the I²C Standard Specification as defined by Philips.

■ FEATURES

- **Clock**
  - Embedded PLL clock multiplication circuit
  - Operating clock (PLL clock) can selected from divided-by-2 of oscillation or one to four times the oscillation (at oscillation of 4 MHz to 16 MHz)
  - Minimum instruction execution time of 62.5 ns (at oscillation of 4 MHz, four times the PLL clock, operation at V_{CC} of 3.3 V)
- **CPU addressing space of 16 Mbytes**
  - Internal 24-bit addressing
- **Instruction set optimized for controller applications**
  - Rich data types (bit, byte, word, long word)
  - Rich addressing mode (23 types)

- High code efficiency
- Enhanced precision calculation realized by the 32-bit accumulator

● **Instruction set designed for high level language (C) and multi-task operations**

- Adoption of system stack pointer
- Enhanced pointer indirect instructions
- Barrel shift instructions

● **Program patch function (2 address pointer)**

● **Improved execution speed**

- 4-byte instruction queue

● **Powerful interrupt function**

- Priority level programmable : 8 levels
- 32 factors of stronger interrupt function

● **Automatic data transmission function independent of CPU operation**

- Extended intelligent I/O service function (EI^{2}OS)
- Maximum 16 channels

● **Low-power consumption (standby) mode**

- Sleep mode (mode in which CPU operating clock is stopped)
- Timebase timer mode (mode in which operations other than timebase timer and watch timer are stopped)
- Stop mode (mode in which all oscillations are stopped)
- CPU intermittent operation mode
- Watch mode

● **Package**

- LQFP-144 (FPT-144P-M12 : 0.4 mm pitch)

● **Process**

- CMOS technology

■ PRODUCT LINEUP

Parameter \ Part number	MB90V370	MB90F372	MB90372
Classification	—	Flash type ROM	Mask ROM
ROM size	—	64K Bytes	
RAM size	15.7K Bytes	6K Bytes	
CPU function	Number of instruction : 351 Minimum execution time : 62.5 ns / 4 MHz (PLL x 4) Addressing mode : 23 Data bit length : 1, 8, 16 bits Maximum memory space : 16 MBytes		
I/O port	I/O port (N-channel) : 16 I/O port (CMOS) : 72 I/O port (CMOS with pull-up control) : 32 Total : 120		
16-bit reload timer	Reload timer : 4 channels Reload mode, single-shot mode or event count mode selectable		
16-bit PPG timer	PPG timer : 3 channels PWM mode or single-shot mode selectable		
Bit decoder	Bit decoder : 1 channel		
Parity generator	Parity generator : 1 channel Selectable odd/even parity		
PS/2 interface	PS/2 interface : 3 channels 4 selectable sampling clocks		
LPC interface	LPC bus interface : 1 channel Universal peripheral Interface : 4 channels GA20 output control : for UPI channel 0 only Data buffer array : 48 bytes		
Serial IRQ controller	Serial IRQ request : 6 channels LPC clock monitor / control		
UART	With full-duplex double buffer (variable data length) Clock asynchronized or clock synchronized transmission (with start and stop bits) can be selectively used		
I²C	I²C (SMbus compliant) : 1 channel Support I²C bus of PHILIPS and the SMbus proposed by Intel I²C bus Selectable packet error check Timeout detection function		
Multi-address I²C	Multi-address I²C (SMbus compliant) : 1 channel Support I²C bus of PHILIPS and the SMbus proposed by Intel I²C bus Selectable packet error check Timeout detection function 6 addresses support ALERT function		
Bridge circuit	Three bus connection routes can be switched by I²C / multi-address I²C		

Parameter \ Part number	MB90V370	MB90F372	MB90372
Comparator	A comparator that can change the hysteresis width is contained Battery voltage, mounting/dismounting and instantaneous interruption can be detected Parallel and serial charging/discharging		

Parameter \ Part number	MB90V370	MB90F372	MB90372
External interrupt	6 independent channels Selectable causes	: Rise/fall edge, fall edge, “L” level or “H” level	
Key-on wake-up interrupt	8 independent channels Causes	: “L” level	
8/10-bit A/D converter	8/10-bit resolution Conversion time	: 12 channels : Less than 6.13 µS (16 MHz internal clock)	
8-bit D/A converter	8-bit resolution	: 2 channels	
LCD controller/driver	Up to 9 SEG x 4 COM Selectable LCD output or CMOS I/O port		
Low-power consumption	Stop mode / Sleep mode / CPU intermittent operation mode / Watch mode		
Process	CMOS		
Package	PGA256	LQFP-144 (FPT-144P-M12: 0.4 mm pitch)	
Operating voltage	3.0~3.6 V @ 16 MHz *		

*: Varies with conditions such as the operating frequency (see Section “■ ELECTRICAL CHARACTERISTICS”). Assurance for the MB90V370 is given only for operation with a tool at power supply voltage of 3.0 V to 3.6 V, an operating temperature of 0 to +25 °C, and an operating frequency of 1 MHz to 16 MHz.

■ PACKAGE AND CORRESPONDING PRODUCTS

Package	MB90V370	MB90F372	MB90372
PGA256	○	X	X
FPT-144P-M12	X	○	○

○ : Available
X : Not available

Note: For more information about each package, see Section “■ PACKAGE DIMENSIONS”.

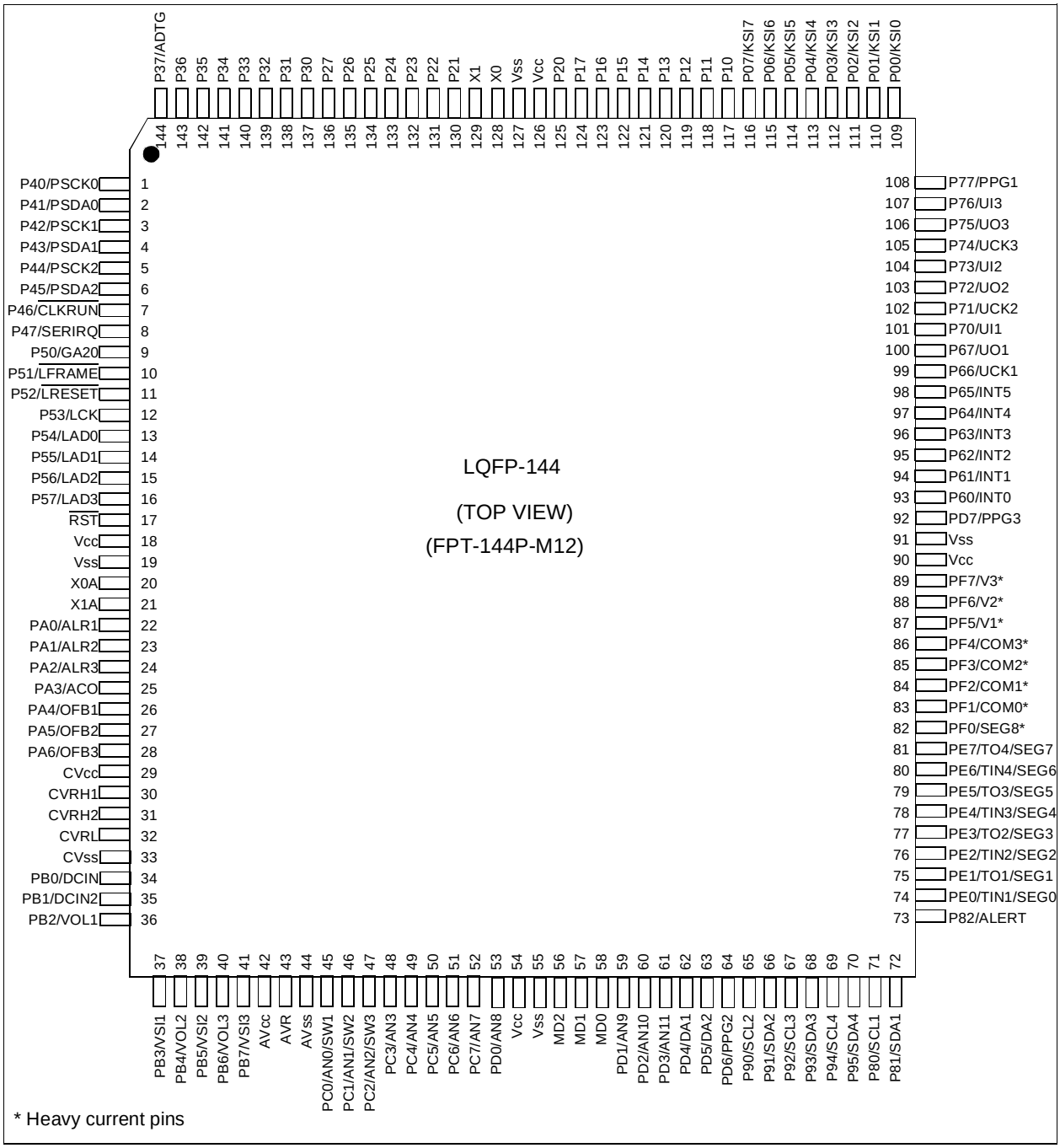
■ DIFFERENCES AMONG PRODUCTS

Memory size

In evaluation with an evaluation product, note the difference between the evaluation product and the product actually used. The following items must be taken into consideration.

- The MB90V370 does not have an internal ROM, however, operations equivalent to chips with an internal ROM can be evaluated by using a dedicated development tool, enabling selection of ROM size by settings of the development tool.
- In the MB90V370, images from $FF4000_H$ to $FFFFFF_H$ are mapped to bank 00, and $FF0000_H$ to $FF3FFF_H$ are mapped to bank FF only. (This setting can be changed by the development tool configuration.)
- In the MB90372/F372, images from $FF4000_H$ to $FFFFFF_H$ are mapped to bank 00, and $FF0000_H$ to $FF3FFF_H$ are mapped to bank FF only.

■ PIN ASSIGNMENT



* Heavy current pins

■ PIN DESCRIPTION

Pin no. LQFP-144	Pin name	I/O circuit	Pin status during reset	Function
128,129	X0,X1	A	Oscillating	Main oscillation input pins.
20,21	X0A,X1A	A	Oscillating	Sub-clock oscillation input pins.
17	$\overline{RST}$	B	Reset input	External reset input pin.
58, 57, 56	MD0 ~ 2	C	Mode input	Input pin for operation mode specification. Connect this pin directly to Vcc or Vss.
109 ~ 116	P00 ~ P07	D	Port input	General-purpose I/O ports.
	KSI0 ~ KSI7			Can be used as key-on wake-up interrupt input channel 0 ~ 7. Input is enabled when 1 is set in EICR: EN0 ~ 7 in standby mode.
117 ~ 124	P10 ~ P17	E		General-purpose I/O ports.
125, 130~136	P20 ~ P27	E		General-purpose I/O ports.
137 ~ 143	P30 ~ P36	E		General-purpose I/O ports.
144	P37	E		General-purpose I/O ports.
	ADTG			External trigger input pin (ADTG) for the A/D converter.
1	P40	F		General-purpose N-ch open-drain I/O port.
	PSCK0			Serial clock I/O pin for PS/2 interface channel 0. This function is selected when PS/2 interface channel 0 is enabled.
2	P41	F		General-purpose N-ch open-drain I/O port.
	PSDA0			Serial data I/O pin for PS/2 interface channel 0. This function is selected when PS/2 interface channel 0 is enabled.
3	P42	F		General-purpose N-ch open-drain I/O port.
	PSCK1			Serial clock I/O pin for PS/2 interface channel 1. This function is selected when PS/2 interface channel 1 is enabled.
4	P43	F		General-purpose N-ch open-drain I/O port.
	PSDA1			Serial data I/O pin for PS/2 interface channel 1. This function is selected when PS/2 interface channel 1 is enabled.
5	P44	F		General-purpose N-ch open-drain I/O port.
	PSCK2			Serial clock I/O pin for PS/2 interface channel 2. This function is selected when PS/2 interface channel 2 is enabled.
6	P45	F		General-purpose N-ch open-drain I/O port.
	PSDA2			Serial data I/O pin for PS/2 interface channel 2. This function is selected when PS/2 interface channel 2 is enabled.
7	P46	G		General-purpose N-ch open-drain I/O port.
	$\overline{CLKRUN}$			LPC clock status / restart request I/O pin for serial IRQ controller. This function is selected when serial IRQ and LPC clock restart request is enabled.
8	P47	H		General-purpose I/O port.
	SERIRQ			Serial IRQ data I/O pin for serial IRQ controller. This function is selected when serial IRQ is enabled.

(Continued)

Pin no. LQFP-144	Pin name	I/O circuit	Pin status during reset	Function
9	P50	H	Port input	General-purpose I/O port.
	GA20			GA20 output for LPC interface. This function is selected when GA20 function is enabled.
10	P51	H		General-purpose I/O port.
	$\overline{\text{LFRAME}}$			LFRAME input for LPC interface. This function is selected when LPC interface is enabled.
11	P52	H		General-purpose I/O port.
	$\overline{\text{LRESET}}$			Reset input for LPC interface. This function is selected when LPC interface is enabled.
12	P53	H		General-purpose I/O port.
	LCK			Clock input for LPC interface. This function is selected when LPC interface is enabled.
13 ~ 16	P54 ~ P57	H		General-purpose I/O ports.
	LAD0 ~ LAD3			Address/Data I/O for LPC interface. This function is selected when LPC interface is enabled.
93 ~ 98	P60 ~ P65	I		General-purpose I/O ports.
	INT0 ~ INT5			Can be used as DTP/external interrupt request input channel 0 ~ 5. Input is enabled when 1 is set in ENIR: EN0 ~ 5 in standby mode.
99	P66	I		General-purpose I/O port.
	UCK1			Serial clock I/O pin for UART channel 1. This function is enabled when UART channel 1 enables clock output.
100	P67	I		General-purpose I/O port.
	UO1			Serial data output pin for UART channel 1. This function is enabled when UART channel 1 enables data output.
101	P70	I		General-purpose I/O port.
	UI1			Serial data input pin for UART channel 1. While UART channel 1 is operating for input, the input of this pin is used as required and must not be used for any other input.
102	P71	I		General-purpose I/O port.
	UCK2			Serial clock I/O pin for UART channel 2. This function is enabled when UART channel 2 enables clock output.
103	P72	I		General-purpose I/O port.
	UO2			Serial data output pin for UART channel 2. This function is enabled when UART channel 2 enables data output.
104	P73	I		General-purpose I/O port.
	UI2			Serial data input pin for UART channel 2. While UART channel 2 is operating for input, the input of this pin is used as required and must not be used for any other input.
105	P74	I		General-purpose I/O port.
	UCK3			Serial clock I/O pin for UART channel 3. This function is enabled when UART channel 3 enables clock output.
106	P75	I		General-purpose I/O port.
	UO3			Serial data output pin for UART channel 3. This function is enabled when UART channel 3 enables data output.

(Continued)

Pin no. LQFP-144	Pin name	I/O circuit	Pin status during reset	Function
107	P76	I	Port input	General-purpose I/O port.
	UI3			Serial data input pin for UART channel 3. While UART channel 3 is operating for input, the input of this pin is used as required and must not be used for any other input.
108	P77	I		General-purpose I/O port.
	PPG1			Output pin for PPG channel 1. This function is enabled when PPG channel 1 output is enabled.
71	P80	T		General-purpose N-ch open-drain I/O port.
	SCL1			Serial clock I/O pin for multi-address I²C.
72	P81	T		General-purpose N-ch open-drain I/O port.
	SDA1			Serial data I/O pin for multi-address I²C.
73	P82	J		General-purpose N-ch open-drain I/O port.
	ALERT			ALERT output pin for multi-address I²C.
65	P90	T		General-purpose N-ch open-drain I/O port.
	SCL2			Serial clock I/O pin for bridge circuit.
66	P91	T		General-purpose N-ch open-drain I/O port.
	SDA2			Serial data I/O pin for bridge circuit.
67	P92	T		General-purpose N-ch open-drain I/O port.
	SCL3			Serial clock I/O pin for bridge circuit.
68	P93	T		General-purpose N-ch open-drain I/O port.
	SDA3			Serial data I/O pin for bridge circuit.
69	P94	T		General-purpose N-ch open-drain I/O port.
	SCL4			Serial clock I/O pin for bridge circuit.
70	P95	T		General-purpose N-ch open-drain I/O port.
	SDA4			Serial data I/O pin for bridge circuit.
22 ~ 24	PA0 ~ PA2	H		General-purpose I/O ports.
	ALR1 ~ ALR3			Alarm signal output when battery 1 ~ 3 run down in comparator circuit.
25	PA3	H		General-purpose I/O port.
	ACO			AC power set signal output in comparator circuit.
26 ~ 28	PA4 ~ PA6	H		General-purpose I/O ports.
	OFB1 ~ OFB3			Battery 1 ~ 3 discharge control signal output in comparator circuit.
34, 35	PB0 ~ PB1	K	Comparator input	General-purpose I/O ports.
	DCIN ~ DCIN2			AC power monitoring input in comparator circuit.
36	PB2	K		General-purpose I/O ports.
	VOL1			Battery 1 power instantaneous interruption monitoring input in comparator circuit.

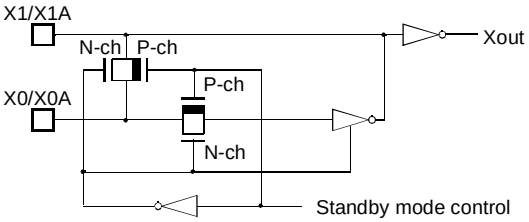
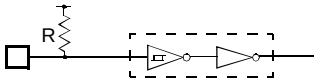
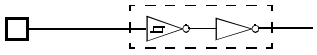
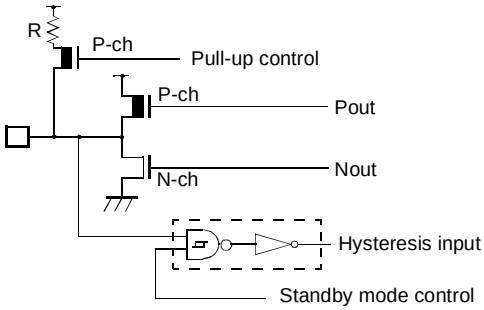
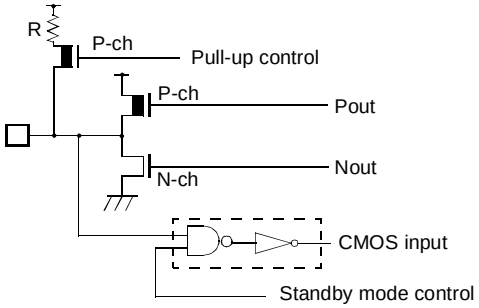
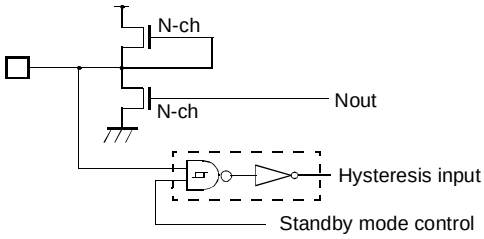
(Continued)

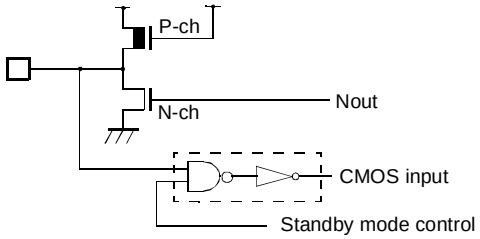
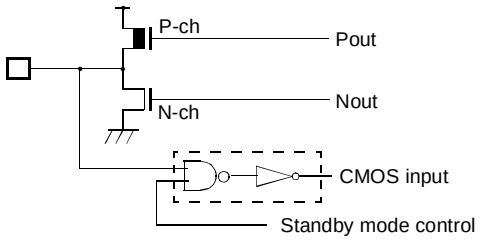
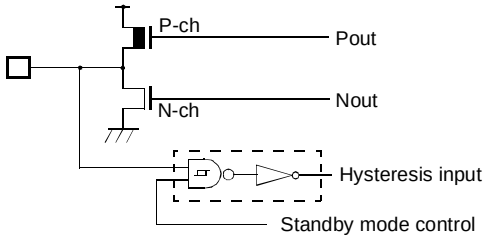
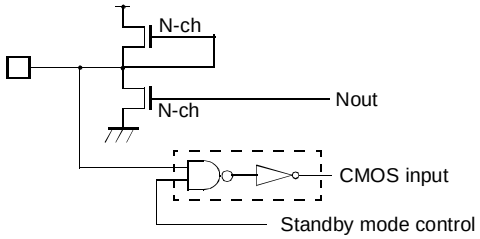
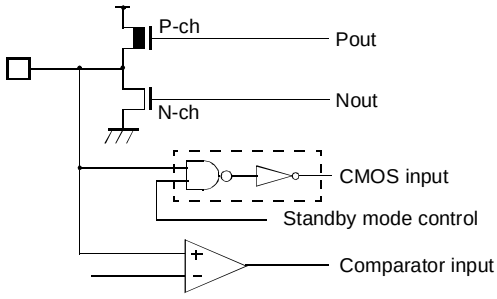
Pin no. LQFP-144	Pin name	I/O circuit	Pin status during reset	Function
37	PB3	K	Comparator input	General-purpose I/O ports.
	VSI1			Battery 1 indicator monitoring input in comparator circuit.
38	PB4	K		General-purpose I/O ports.
	VOL2			Battery 2 power instantaneous interruption monitoring input in comparator circuit.
39	PB5	K		General-purpose I/O ports.
	VSI2			Battery 2 indicator monitoring input in comparator circuit.
40	PB6	K		General-purpose I/O ports.
	VOL3			Battery 3 power instantaneous interruption monitoring input in comparator circuit.
41	PB7	K		General-purpose I/O ports.
	VSI3			Battery 3 indicator monitoring input in comparator circuit.
45 ~ 47	PC0 ~ PC2	L	Comparator input or A/D input	General-purpose I/O ports.
	SW1 ~ SW3			Battery 1 ~ 3 mount / dismount detection input in comparator circuit.
	AN0 ~ AN2			A/D converter analog input pin 0 ~ 2. This function is enabled when the analog input specification is enabled (ADER1).
48 ~ 52	PC3 ~ PC7	M	A/D input	General-purpose I/O ports.
	AN3 ~ AN7			A/D converter analog input pin 3 ~ 7. This function is enabled when the analog input specification is enabled (ADER1).
53, 59 ~ 61	PD0 ~ PD3	M		General-purpose I/O ports.
	AN8 ~ AN11			A/D converter analog input pin 8 ~ 11. This function is enabled when the analog input specification is enabled (ADER2).
62 ~ 63	PD4 ~ PD5	N	Port input	General-purpose I/O ports.
	DA1 ~ DA2			D/A converter analog output 1 ~ 2. This function is selected when D/A converted is enabled.
64, 92	PD6 ~ PD7	H		General-purpose I/O port.
	PPG2 ~ PPG3			Output pin for PPG channel 2 ~ 3. This function is selected when PPG channel 2 ~ 3 output is enabled.
74	PE0	O		General-purpose I/O port.
	SEG0			Segment output pin for LCD controller/driver. This function is selected when LCD segment output is enabled.
	TIN1			External clock input pin for reload timer 1.
75	PE1	O		General-purpose I/O port.
	SEG1			Segment output pin for LCD controller/driver. This function is selected when LCD segment output is enabled.
	TO1			Event output pin for reload timer 1.
76	PE2	O		General-purpose I/O port.
	SEG2			Segment output pin for LCD controller/driver. This function is selected when LCD segment output is enabled.
	TIN2			External clock input pin for reload timer 2.

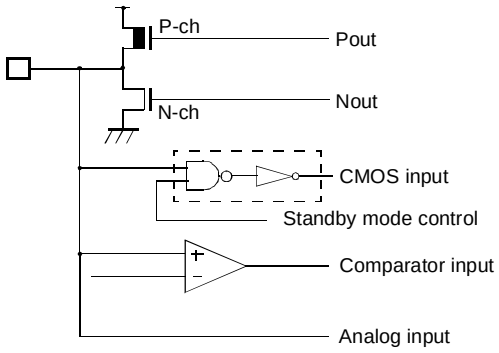
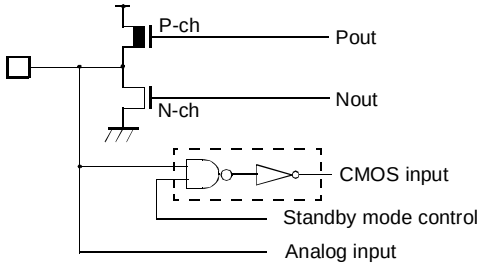
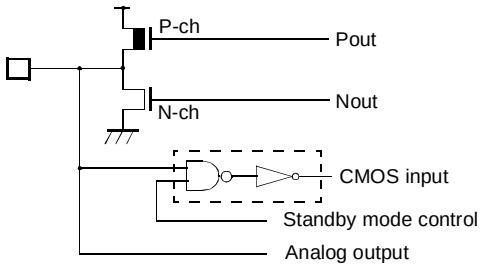
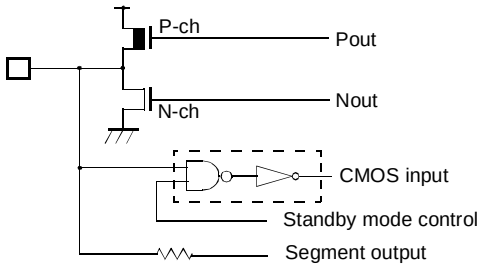
(Continued)

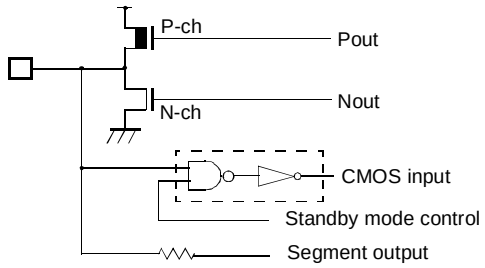
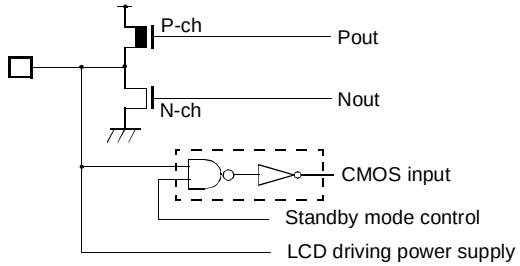
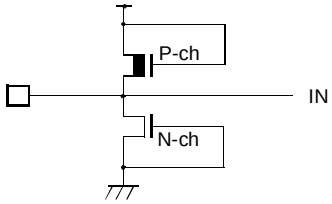
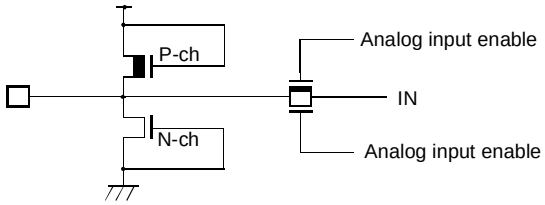
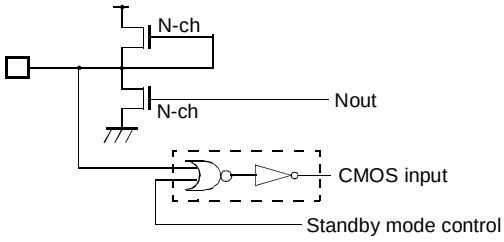
Pin no. LQFP-144	Pin name	I/O circuit	Pin status during reset	Function
77	PE3	O	Port input	General-purpose I/O port.
	SEG3			Segment output pin for LCD controller/driver. This function is selected when LCD segment output is enabled.
	TO2			Event output pin for reload timer 2.
78	PE4	O		General-purpose I/O port.
	SEG4			Segment output pin for LCD controller/driver. This function is selected when LCD segment output is enabled.
	TIN3			External clock input pin for reload timer 3.
79	PE5	O		General-purpose I/O port.
	SEG5			Segment output pin for LCD controller/driver. This function is selected when LCD segment output is enabled.
	TO3			Event output pin for reload timer 3.
80	PE6	O		General-purpose I/O port.
	SEG6			Segment output pin for LCD controller/driver. This function is selected when LCD segment output is enabled.
	TIN4			External clock input pin for reload timer 4.
81	PE7	O		General-purpose I/O port.
	SEG7			Segment output pin for LCD controller/driver. This function is selected when LCD segment output is enabled.
	TO4			Event output pin for reload timer 4.
82	PF0	P		General-purpose I/O port.
	SEG8			Segment output pin for LCD controller/driver. This function is selected when LCD segment output is enabled.
83 ~ 86	PF1 ~ PF4	P		General-purpose I/O port.
	COM0 ~ COM3			COM output pin for LCD controller/driver. This function is selected when LCD COM output is enabled.
87 ~ 89	PF5 ~ PF7	Q	Power input	General-purpose I/O port.
	V1 ~ V3			Power input pin for LCD controller/driver. This function is selected when external voltage divider is enabled.
42	AVCC	R	Power input	Vcc power input pin for analog circuits.
43	AVR	S		Vref+ input pin for the A/D converter. This voltage must not exceed Vcc. Vref- is fixed to AVSS.
44	AVSS	R		Vss power input pin for analog circuits.
29	CVCC	R	Power input	Vcc power input pin for analog circuits.
30	CVRH1	R		Standard power input pin of the comparator.
31	CVRH2	R		
32	CVRL	R		
33	CVSS	R		Vss power input pin for analog circuits.
19,55,91,127	Vss	–	Power input	Power (0 V) input pin.
18,54,90,126	Vcc	–		Power (3.3 V) input pin.

■ I/O CIRCUIT TYPE

Classification	Type	Remarks
A		Main/Sub clock (main/sub clock crystal oscillator) • At an oscillation feedback resistor of approximately 1 MΩ
B		• Hysteresis input • Pull-up resistor approximately 50 kΩ
C		• Hysteresis input
D		• CMOS output • Hysteresis input • Selectable pull-up resistor approximately 50 kΩ • I_{OL} = 4 mA
E		• CMOS output • CMOS input • Selectable pull-up resistor approximately 50 kΩ • I_{OL} = 4 mA
F		• N-ch open-drain output • Hysteresis input • I_{OL} = 4 mA • 5V tolerant

Classification	Type	Remarks
G		• N-ch open-drain output • CMOS input • I_{OL} = 4 mA
H		• CMOS output • CMOS input • I_{OL} = 4 mA
I		• CMOS output • Hysteresis input • I_{OL} = 4 mA
J		• N-ch open-drain output • CMOS input • I_{OL} = 4 mA • 5V tolerant
K		• CMOS output • CMOS input • Comparator input • I_{OL} = 4 mA

Classification	Type	Remarks
L		• CMOS output • CMOS input • Comparator input • A/D analog input • I_{OL} = 4 mA
M		• CMOS output • CMOS input • A/D analog input • I_{OL} = 4 mA
N		• CMOS output • CMOS input • D/A analog output • I_{OL} = 4 mA
O		• CMOS output • CMOS input • Segment output • I_{OL} = 4 mA

Classification	Type	Remarks
P		• CMOS output • CMOS input • Segment output • I_{OL} = 12 mA
Q		• CMOS output • CMOS input • LCD driving power supply • I_{OL} = 12 mA
R		• Power supply input protection circuit
S		• A/D converter reference voltage (AVR) input pin with protection circuit
T		• N-ch open-drain output • CMOS input • I_{OL} = 4 mA • 5V tolerant

■ HANDLING DEVICES

- Be sure that the maximum rated voltage is not exceeded (latch-up prevention).

  A latch-up may occur on a CMOS IC if a voltage higher than V_{CC} or lower than V_{SS} is applied to an input or output pin other than medium-to-high voltage pins. A latch-up may also occur if a voltage higher than the rating is applied between V_{CC} and V_{SS}. A latch-up causes a rapid increase in the power supply current, which can result in thermal damage to an element. Take utmost care that the maximum rated voltage is not exceeded.

  When turning the power on or off to analog circuits, be sure that the analog supply voltages (AV_{CC}, CV_{CC}, AVR, CVRH1, CVRH2 and CVRL) and analog input voltage do not exceed the digital supply voltage (V_{CC}).

- Stabilize the supply voltages

  Even within the operation guarantee range of the V_{CC} supply voltage, a malfunction can be caused if the supply voltage undergoes a rapid change. For voltage stabilization guidelines, the V_{CC} ripple fluctuations (P-P value) at commercial frequencies (50 to 60 Hz) should be suppressed to "10%" or less of the reference V_{CC} value. During a momentary change such as when switching a supply voltage, voltage fluctuations should also be suppressed so that the "transient fluctuation rate" is 0.1 V/ms or less.

- Power-on

  To prevent a malfunction in the built-in voltage drop circuit, secure "50 µs (between 0.2 V and 1.8 V)" or more for the voltage rise time during power-on.

- Treatment of unused input pins

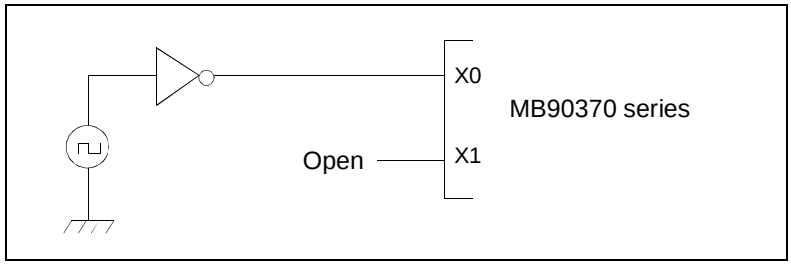
  An unused input pin may cause a malfunction if it is left open. Every unused input pin should be pulled up or down.

- Treatment of A/D converter, D/A converter and comparator power pin

  When the A/D converter, D/A converter and comparator is not used, connect the pins as follows: $AV_{CC} = CV_{CC} = V_{CC}$, $AV_{SS} = AVR = CV_{SS} = CVRL = CVRH1 = CVRH2 = V_{SS}$.

- Notes on external clock

  When an external clock is used, the oscillation stabilization wait time is required at power-on reset or at cancellation of sub-clock mode or stop mode. As shown in diagram below, when an external clock is used, connect only the X0 pin and leave the X1 pin open.



- Power supply pins

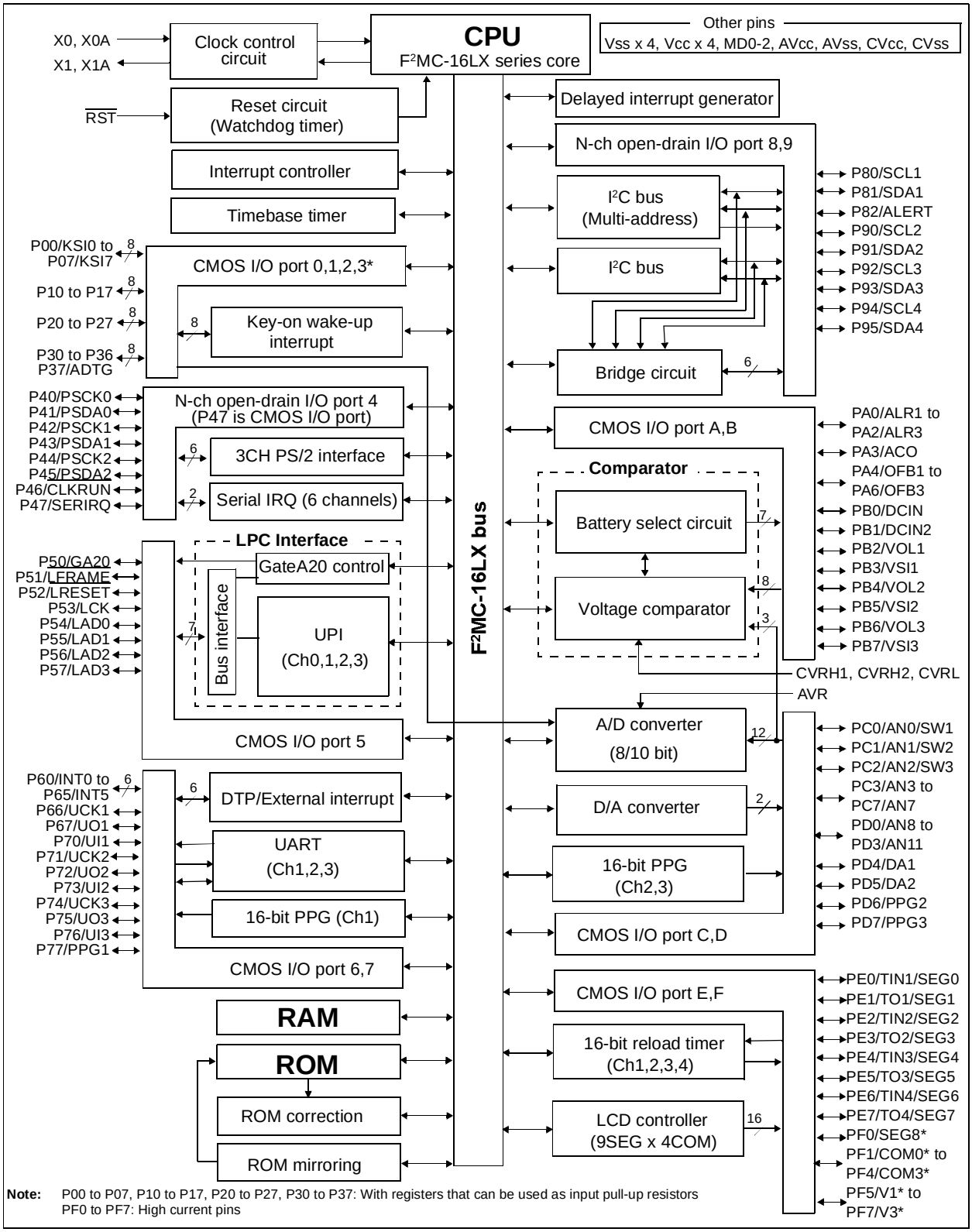
  When a device has two or more V_{CC} or V_{SS} pins, the pins that should have equal potential are connected within the device in order to prevent a latch-up or other malfunction. To reduce extraneous emission, to prevent a malfunction of the strobe signal due to an increase in the group level, and to maintain the local output current rating, connect all these power supply pins to an external power supply and ground them.

  The current source should be connected to the V_{CC} and V_{SS} pins of the device with minimum impedance. It is recommended that a bypass capacitor of about 0.1 µF be connected near the terminals between V_{CC} and V_{SS}.

- Analog power-on sequence of A/D converter, D/A converter and comparator

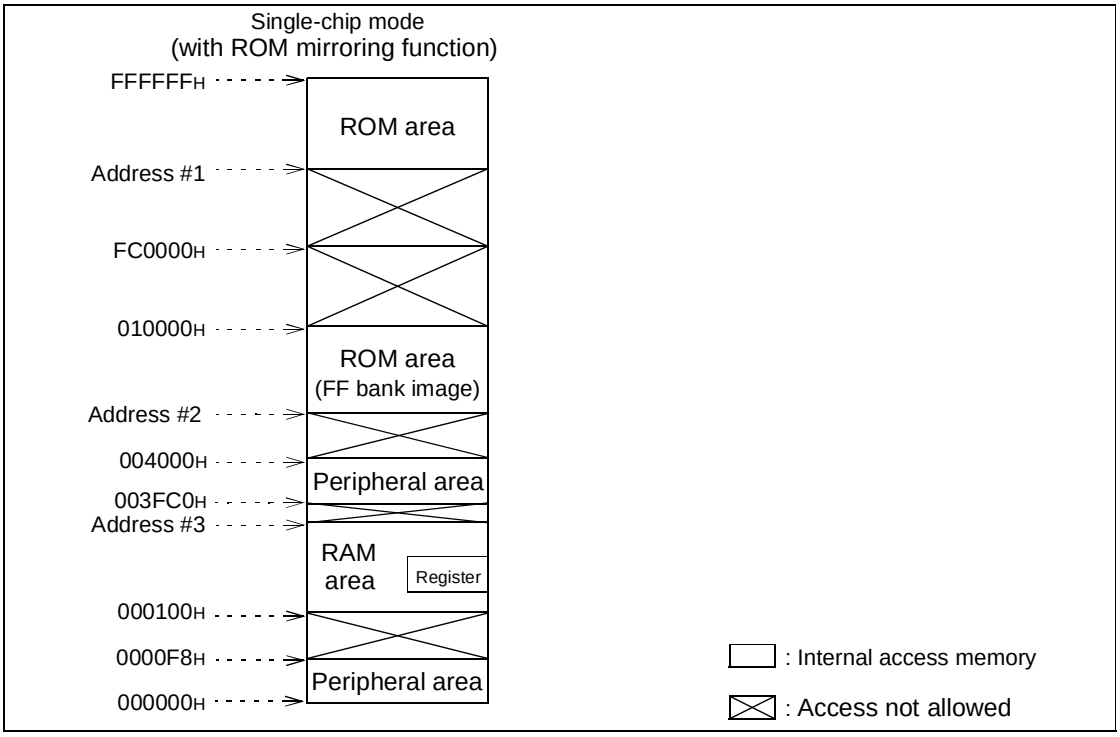
  The power to the A/D converter, D/A converter and comparator (AV_{CC}, CV_{CC}, AVR, CVRH1, CVRH2 and CVRL) and analog inputs (AN0 ~ AN11, VOL1 ~ 3, VSI1 ~ 3, SW1 ~ 3, DCIN and DCIN2) must be turned on after the power to the digital circuits (V_{CC}) is turned on. When turning off the power, turn off the power to the digital circuits (V_{CC}) after turning off the power to the A/D converter, D/A converter, comparator and analog inputs. When the power is turned on or off, AVR should not exceed AV_{CC}. And CVRH1, CVRH2 and CVRL should not exceed CV_{CC}. Also, when a pin that is used for A/D analog input is also used as an input port, the input voltage should not exceed AV_{CC}. And when comparator analog input is also used as an input port, the input voltage should not exceed CV_{CC}. (The power to the analog circuits and the power to the digital circuits can be simultaneously turned on or off.)

■ BLOCK DIAGRAM



Note: P00 to P07, P10 to P17, P20 to P27, P30 to P37: With registers that can be used as input pull-up resistors
PF0 to PF7: High current pins

■ MEMORY MAP



Model	Address #1	Address #2	Address #3
MB90372	FF0000H	004000H	001900H
MB90F372	FF0000H	004000H	001900H
MB90V370	FF0000H*1	004000H*1	003FC0H

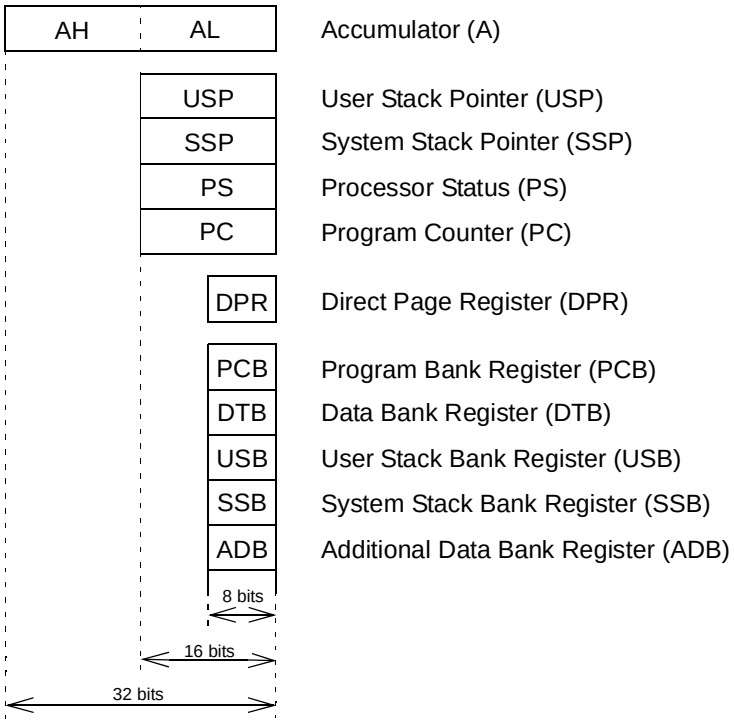
*1: The MB90V370 does not contain ROM. Assume that the development tool uses these area for its ROM decode areas.

Notes:

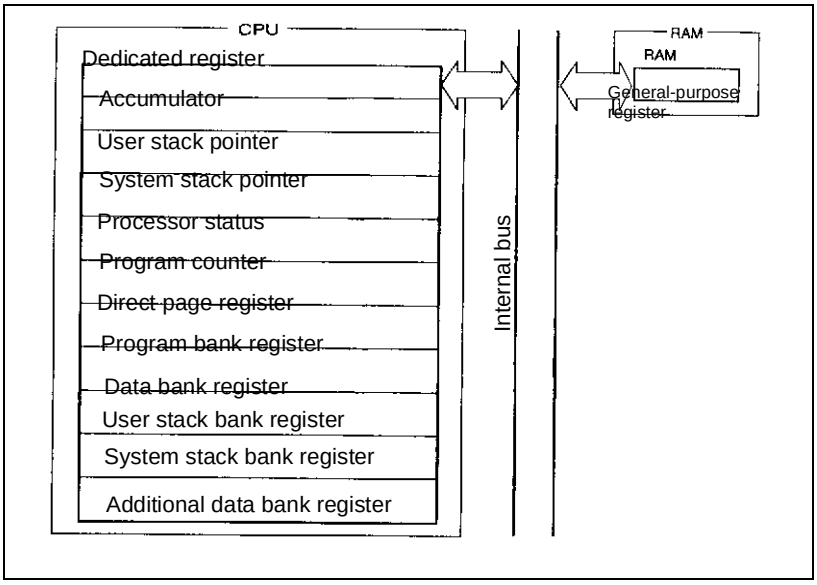
- If single-chip mode (without ROM mirroring function) is selected, see Chapter 31, "ROM Mirroring Function Selection Module" of the MB90370 series H/W manual.
- ROM data in the FF bank can be seen as an image in the higher 00 bank to validate the small model C compiler. Because addresses of the 16 low-order bits in the FF bank are the same, the table in ROM can be referenced without the "far" specification. For example, when 00C000H is accessed, the contents of ROM at FFC000H are actually accessed. The ROM area in the FF bank exceeds 48 kilobytes, and all areas cannot be seen as images in the 00 bank. Because ROM data from FF4000H to FFFFFFH is seen as an image at 004000H to 00FFFFH, the ROM data table should be stored in the area from FF4000H to FFFFFFH.

■ F²MC-16LX CPU PROGRAMMING MODEL

• Dedicated registers



• General-purpose registers



• Processor status (PS)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PS	ILM			RP					CCR							
Default value ⇒	0	0	0	0	0	0	0	0	–	1	0	X	X	X	X	X

: CCR

: RP

: ILM

- : Not used
X : Undefined

■ I/O MAP

Address	Abbreviation	Register	Byte access	Word access	Resource name	Initial value
000000H	PDR0	Port 0 data register	R/W	R/W	Port 0	XXXXXXXXB
000001H	PDR1	Port 1 data register	R/W	R/W	Port 1	XXXXXXXXB
000002H	PDR2	Port 2 data register	R/W	R/W	Port 2	XXXXXXXXB
000003H	PDR3	Port 3 data register	R/W	R/W	Port 3	XXXXXXXXB
000004H	PDR4	Port 4 data register	R/W	R/W	Port 4	X1111111B
000005H	PDR5	Port 5 data register	R/W	R/W	Port 5	XXXXXXXXB
000006H	PDR6	Port 6 data register	R/W	R/W	Port 6	XXXXXXXXB
000007H	PDR7	Port 7 data register	R/W	R/W	Port 7	XXXXXXXXB
000008H	PDR8	Port 8 data register	R/W	R/W	Port 8	-----111B
000009H	PDR9	Port 9 data register	R/W	R/W	Port 9	--111111B
00000AH	PDRA	Port A data register	R/W	R/W	Port A	-XXXXXXXB
00000BH	PDRB	Port B data register	R/W	R/W	Port B	XXXXXXXXB
00000CH	PDRC	Port C data register	R/W	R/W	Port C	XXXXXXXXB
00000DH	PDRD	Port D data register	R/W	R/W	Port D	XXXXXXXXB
00000EH	PDRE	Port E data register	R/W	R/W	Port E	XXXXXXXXB
00000FH	PDRF	Port F data register	R/W	R/W	Port F	XXXXXXXXB
000010H	DDR0	Port 0 direction register	R/W	R/W	Port 0	00000000B
000011H	DDR1	Port 1 direction register	R/W	R/W	Port 1	00000000B
000012H	DDR2	Port 2 direction register	R/W	R/W	Port 2	00000000B
000013H	DDR3	Port 3 direction register	R/W	R/W	Port 3	00000000B
000014H	DDR4	Port 4 direction register	R/W	R/W	Port 4	0-------B
000015H	DDR5	Port 5 direction register	R/W	R/W	Port 5	00000000B
000016H	DDR6	Port 6 direction register	R/W	R/W	Port 6	00000000B
000017H	DDR7	Port 7 direction register	R/W	R/W	Port 7	00000000B
000018H	PGDR	Parity generator data register	R/W	R/W	Parity generator	XXXXXXXXB
000019H	PGCSR	Parity generator control status register	R/W	R/W		X------0B
00001AH	DDRA	Port A direction register	R/W	R/W	Port A	-0000000B
00001BH	DDRB	Port B direction register	R/W	R/W	Port B	00000000B
00001CH	DDRC	Port C direction register	R/W	R/W	Port C	00000000B
00001DH	DDRD	Port D direction register	R/W	R/W	Port D	00000000B

(Continued)

Address	Abbreviation	Register	Byte access	Word access	Resource name	Initial value
00001E$_H$	DDRE	Port E direction register	R/W	R/W	Port E	00000000$_B$
00001F$_H$	DDRF	Port F direction register	R/W	R/W	Port F	00000000$_B$
000020$_H$	SMR1	Serial mode register 1	R/W	R/W	UART1	00000-00$_B$
000021$_H$	SCR1	Serial control register 1	R/W	R/W		00000100$_B$
000022$_H$	SIDR1/ SODR1	Input data register 1 / Output data register 1	R/W	R/W		XXXXXXXX$_B$
000023$_H$	SSR1	Serial status register 1	R/W	R/W		00001000$_B$
000024$_H$	M2CR1	Mode 2 control register 1	R/W	R/W		----1000$_B$
000025$_H$	CDCR1	Clock division control register 1	R/W	R/W	Communication prescaler 1	00--0000$_B$
000026$_H$	ENIR	Interrupt / DTP enable register	R/W	R/W	DTP/external interrupt	--000000$_B$
000027$_H$	EIRR	Interrupt / DTP cause register	R/W	R/W		--XXXXXX$_B$
000028$_H$	ELVR	Request level setting register	R/W	R/W		00000000$_B$
000029$_H$			R/W	R/W		----0000$_B$
00002A$_H$	ADER1	Analog input enable register 1	R/W	R/W	Port C, A/D	11111111$_B$
00002B$_H$	ADER2	Analog input enable register 2	R/W	R/W	Port D, A/D	----1111$_B$
00002C$_H$	BRSR	Bridge circuit selection register	R/W	R/W	Bridge circuit	--000000$_B$
00002D$_H$	ADC0	A/D control register	R/W	R/W	8/10-bit A/D converter	00000000$_B$
00002E$_H$	ADCR0	A/D data register	R	R		XXXXXXXX$_B$
00002F$_H$	ADCR1		R/W	R/W		00000-XX$_B$
000030$_H$	ADCS0	A/D control status register	R/W	R/W		00--------$_B$
000031$_H$	ADCS1		R/W	R/W		00000000$_B$
000032$_H$	SICRL	Serial interrupt request register	R/W	R/W	Serial IRQ	00000000$_B$
000033$_H$	SICRH	Serial interrupt control register	R/W	R/W		00000000$_B$
000034$_H$	SIFR1	Serial interrupt frame number register 1	R/W	R/W		--000000$_B$
000035$_H$	SIFR2	Serial interrupt frame number register 2	R/W	R/W		--000000$_B$
000036$_H$	SIFR3	Serial interrupt frame number register 3	R/W	R/W		--000000$_B$
000037$_H$	SIFR4	Serial interrupt frame number register 4	R/W	R/W		--000000$_B$
000038$_H$	PDCRL1	PPG1 down counter register	-	R	16-bit PPG timer (CH1)	11111111$_B$
000039$_H$	PDCRH1		-	R		11111111$_B$
00003A$_H$	PCSRL1	PPG1 period setting register	-	W		XXXXXXXX$_B$
00003B$_H$	PCSRH1		-	W		XXXXXXXX$_B$
00003C$_H$	PDUTL1	PPG1 duty setting register	-	W		XXXXXXXX$_B$
00003D$_H$	PDUTH1		-	W		XXXXXXXX$_B$
00003E$_H$	PCNTL1	PPG1 control status register	R/W	R/W		--000000$_B$
00003F$_H$	PCNTH1		R/W	R/W		00000000$_B$

(Continued)

Address	Abbreviation	Register	Byte access	Word access	Resource name	Initial value
000040H	PDCRL2	PPG2 down counter register	-	R	16-bit PPG timer (CH2)	11111111B
000041H	PDCRH2		-	R		11111111B
000042H	PCSRL2	PPG2 period setting register	-	W		XXXXXXXXB
000043H	PCSRH2		-	W		XXXXXXXXB
000044H	PDUTL2	PPG2 duty setting register	-	W		XXXXXXXXB
000045H	PDUTH2		-	W		XXXXXXXXB
000046H	PCNTL2	PPG2 control status register	R/W	R/W		--000000B
000047H	PCNTH2		R/W	R/W		00000000B
000048H	PDCRL3	PPG3 down counter register	-	R	16-bit PPG timer (CH3)	11111111B
000049H	PDCRH3		-	R		11111111B
00004AH	PCSRL3	PPG3 period setting register	-	W		XXXXXXXXB
00004BH	PCSRH3		-	W		XXXXXXXXB
00004CH	PDUTL3	PPG3 duty setting register	-	W		XXXXXXXXB
00004DH	PDUTH3		-	W		XXXXXXXXB
00004EH	PCNTL3	PPG3 control status register	R/W	R/W		--000000B
00004FH	PCNTH3		R/W	R/W		00000000B
000050H	PSCR0	PS/2 interface control register 0	R/W	R/W	3-channel PS/2 interface	0--00000B
000051H	PSSR0	PS/2 interface status register 0	R/W	R/W		00000000B
000052H	PSCR1	PS/2 interface control register 1	R/W	R/W		0--00000B
000053H	PSSR1	PS/2 interface status register 1	R/W	R/W		00000000B
000054H	PSCR2	PS/2 interface control register 2	R/W	R/W		0--00000B
000055H	PSSR2	PS/2 interface status register 2	R/W	R/W		00000000B
000056H	PSDR0	PS/2 interface data register 0	R/W	R/W		00000000B
000057H	PSDR1	PS/2 interface data register 1	R/W	R/W		00000000B
000058H	PSDR2	PS/2 interface data register 2	R/W	R/W		00000000B
000059H	PSMR	PS/2 interface mode register	R/W	R/W		----0000B
00005AH	DAT0	D/A converter data register 0	R/W	R/W	D/A converter	XXXXXXXXB
00005BH	DAT1	D/A converter data register 1	R/W	R/W		XXXXXXXXB
00005CH	DACR0	D/A control register 0	R/W	R/W		-------0B
00005DH	DACR1	D/A control register 1	R/W	R/W		-------0B

(Continued)

Address	Abbreviation	Register	Byte access	Word access	Resource name	Initial value
00005E$_H$	UPAL1	UPI1 address register (lower)	R/W	R/W	LPC interface	XXXXXXXX$_B$
00005F$_H$	UPAH1	UPI1 address register (upper)	R/W	R/W		XXXXXXXX$_B$
000060$_H$	UPAL2	UPI2 address register (lower)	R/W	R/W		XXXXXXXX$_B$
000061$_H$	UPAH2	UPI2 address register (upper)	R/W	R/W		XXXXXXXX$_B$
000062$_H$	UPAL3	UPI3 address register (lower)	R/W	R/W		XXXXXXXX$_B$
000063$_H$	UPAH3	UPI3 address register (upper)	R/W	R/W		XXXXXXXX$_B$
000064$_H$	UPCL	UPI control register (lower)	R/W	R/W		00000000$_B$
000065$_H$	UPCH	UPI control register (upper)	R/W	R/W		-000-000$_B$
000066$_H$	UPDI0/ UPDO0	UPI0 data input register / data output register	R/W	R/W		XXXXXXXX$_B$
000067$_H$	UPS0	UPI0 status register	R/W	R/W		00000000$_B$
000068$_H$	UPDI1/ UPDO1	UPI1 data input register / data output register	R/W	R/W		XXXXXXXX$_B$
000069$_H$	UPS1	UPI1 status register	R/W	R/W		00000000$_B$
00006A$_H$	UPDI2/ UPDO2	UPI2 data input register / data output register	R/W	R/W		XXXXXXXX$_B$
00006B$_H$	UPS2	UPI2 status register	R/W	R/W		00000000$_B$
00006C$_H$	UPDI3/ UPDO3	UPI3 data input register / data output register	R/W	R/W		XXXXXXXX$_B$
00006D$_H$	UPS3	UPI3 status register	R/W	R/W		00000000$_B$
00006E$_H$	LCR	LPC control register	R/W	R/W		-----000$_B$
00006F$_H$	ROMM	ROM mirroring function selection register	W	W	ROM mirroring function	-------1$_B$
000070$_H$	TMCSRL1	Timer control status register CH1 (lower)	R/W	R/W	16-bit reload timer (CH1)	00000000$_B$
000071$_H$	TMCSRH1	Timer control status register CH1 (upper)	R/W	R/W		----0000$_B$
000072$_H$	TMR1/ TMRD1	16-bit timer/reload register CH1	-	R/W		XXXXXXXX$_B$
000073$_H$			-	R/W		XXXXXXXX$_B$
000074$_H$	TMCSRL2	Timer control status register CH2 (lower)	R/W	R/W	16-bit reload timer (CH2)	00000000$_B$
000075$_H$	TMCSRH2	Timer control status register CH2 (upper)	R/W	R/W		----0000$_B$
000076$_H$	TMR2/ TMRD2	16-bit timer/reload register CH2	-	R/W		XXXXXXXX$_B$
000077$_H$			-	R/W		XXXXXXXX$_B$

(Continued)

Address	Abbreviation	Register	Byte access	Word access	Resource name	Initial value
000078H	TMCSRL3	Timer control status register CH3 (lower)	R/W	R/W	16-bit reload timer (CH3)	00000000B
000079H	TMCSRH3	Timer control status register CH3 (upper)	R/W	R/W		----0000B
00007AH	TMR3/TMRD3	16-bit timer/reload register CH3	-	R/W		XXXXXXXXB
00007BH			-	R/W		XXXXXXXXB
00007CH	TMCSRL4	Timer control status register CH4 (lower)	R/W	R/W	16-bit reload timer (CH4)	00000000B
00007DH	TMCSRH4	Timer control status register CH4 (upper)	R/W	R/W		----0000B
00007EH	TMR4/TMRD4	16-bit timer/reload register CH4	-	R/W		XXXXXXXXB
00007FH			-	R/W		XXXXXXXXB
000080H	IBCRL	I²C bus control register (lower)	R/W	R/W	I²C	----0000B
000081H	IBCRH	I²C bus control register (upper)	R/W	R/W		00000000B
000082H	IBSRL	I²C bus status register (lower)	R	R		00000000B
000083H	IBSRH	I²C bus status register (upper)	R/W	R/W		--000000B
000084H	IDAR	I²C data register	R/W	R/W		XXXXXXXXB
000085H	IADR	I²C address register	R/W	R/W		-XXXXXXXB
000086H	ICCR	I²C clock control register	R/W	R/W		0-000000B
000087H	ITCR	I²C timeout control register	R/W	R/W		-0-00000B
000088H	ITOC	I²C timeout clock register	R/W	R/W		00000000B
000089H	ITOD	I²C timeout data register	R/W	R/W		00000000B
00008AH	ISTO	I²C slave timeout register	R/W	R/W		00000000B
00008BH	IMTO	I²C master timeout register	R/W	R/W		00000000B
00008CH	RDR0	Port 0 pull-up resistor setting register	R/W	R/W	Port 0	00000000B
00008DH	RDR1	Port 1 pull-up resistor setting register	R/W	R/W	Port 1	00000000B
00008EH	RDR2	Port 2 pull-up resistor setting register	R/W	R/W	Port 2	00000000B
00008FH	RDR3	Port 3 pull-up resistor setting register	R/W	R/W	Port 3	00000000B
000090H ~ 9DH	Prohibited area					
00009EH	PACSR	Program address detect control status register	R/W	R/W	Address match detection	00000000B
00009FH	DIRR	Delayed interrupt cause / clear register	R/W	R/W	Delayed interrupt	-------0B

(Continued)

Address	Abbreviation	Register	Byte access	Word access	Resource name	Initial value
0000A0H	LPMCR	Low-power consumption mode register	R/W	R/W	Low-power consumption control register	00011000B
0000A1H	CKSCR	Clock selection register	R/W	R/W		11111100B
0000A2H ~ A3H	Prohibited area					
0000A4H	CKMC	Clock modulation control register	R/W	R/W	Clock modulation	-------0B
0000A5H ~ A7H	Prohibited area					
0000A8H	WDTC	Watchdog control register	R/W	R/W	Watchdog timer	X-XXX111B
0000A9H	TBTC	Timebase timer control register	R/W	R/W	Timebase timer	1--00100B
0000AAH	WTC	Watch timer control register	R/W	R/W	Watch timer	10001000B
0000ABH	Prohibited area					
0000ACH	EICR	Wake-up interrupt control register	R/W	R/W	Wake-up interrupt	00000000B
0000ADH	EIFR	Wake-up interrupt flag register	R/W	R/W		-------0B
0000AEH	FMCS	Flash memory control status register	R/W	R/W	Flash memory interface circuit	00010000B
0000AFH	Prohibited area					
0000B0H	ICR00	Interrupt control register 00	R/W	R/W	Interrupt controller	00000111B
0000B1H	ICR01	Interrupt control register 01	R/W	R/W		00000111B
0000B2H	ICR02	Interrupt control register 02	R/W	R/W		00000111B
0000B3H	ICR03	Interrupt control register 03	R/W	R/W		00000111B
0000B4H	ICR04	Interrupt control register 04	R/W	R/W		00000111B
0000B5H	ICR05	Interrupt control register 05	R/W	R/W		00000111B
0000B6H	ICR06	Interrupt control register 06	R/W	R/W		00000111B
0000B7H	ICR07	Interrupt control register 07	R/W	R/W		00000111B
0000B8H	ICR08	Interrupt control register 08	R/W	R/W		00000111B
0000B9H	ICR09	Interrupt control register 09	R/W	R/W		00000111B
0000BAH	ICR10	Interrupt control register 10	R/W	R/W		00000111B
0000BBH	ICR11	Interrupt control register 11	R/W	R/W		00000111B
0000BCH	ICR12	Interrupt control register 12	R/W	R/W		00000111B
0000BDH	ICR13	Interrupt control register 13	R/W	R/W		00000111B
0000BEH	ICR14	Interrupt control register 14	R/W	R/W		00000111B
0000BFH	ICR15	Interrupt control register 15	R/W	R/W		00000111B

(Continued)

Address	Abbreviation	Register	Byte access	Word access	Resource name	Initial value
0000C0H	MBCRL	MI²C bus control register (lower)	R/W	R/W	MI²C	----0000B
0000C1H	MBCRH	MI²C bus control register (upper)	R/W	R/W		00000000B
0000C2H	MBSRL	MI²C bus status register (lower)	R	R		00000000B
0000C3H	MBSRH	MI²C bus status register (upper)	R/W	R/W		--000000B
0000C4H	MDAR	MI²C data register	R/W	R/W		XXXXXXXXB
0000C5H	MALR	MI²C alert register	R/W	R/W		----0000B
0000C6H	MADR1	MI²C address register 1	R/W	R/W		-XXXXXXXB
0000C7H	MADR2	MI²C address register 2	R/W	R/W		-XXXXXXXB
0000C8H	MADR3	MI²C address register 3	R/W	R/W		-XXXXXXXB
0000C9H	MADR4	MI²C address register 4	R/W	R/W		-XXXXXXXB
0000CAH	MADR5	MI²C address register 5	R/W	R/W		-XXXXXXXB
0000CBH	MADR6	MI²C address register 6	R/W	R/W		-XXXXXXXB
0000CCH	MCCR	MI²C clock control register	R/W	R/W		0-000000B
0000CDH	MTCR	MI²C timeout control register	R/W	R/W		-0-00000B
0000CEH	MTOC	MI²C timeout clock register	R/W	R/W		00000000B
0000CFH	MTOD	MI²C timeout data register	R/W	R/W		00000000B
0000D0H	MSTO	MI²C slave timeout register	R/W	R/W		00000000B
0000D1H	MMTO	MI²C master timeout register	R/W	R/W		00000000B
0000D2H	SMR2	Serial mode register 2	R/W	R/W	UART2	00000-00B
0000D3H	SCR2	Serial control register 2	R/W	R/W		00000100B
0000D4H	SIDR2/ SODR2	Input data register 2 / output data register 2	R/W	R/W		XXXXXXXXB
0000D5H	SSR2	Status register 2	R/W	R/W		00001000B
0000D6H	M2CR2	Mode 2 control register 2	R/W	R/W		----1000B
0000D7H	CDCR2	Clock division control register 2	R/W	R/W	Communication prescaler 2	00--0000B

(Continued)

Address	Abbreviation	Register	Byte access	Word access	Resource name	Initial value
0000D8H	COCRL	Comparator control register (lower)	R/W	R/W	Voltage comparator	--000000B
0000D9H	COCRH	Comparator control register (upper)	R/W	R/W		00011111B
0000DAH	COSRL1	Comparator status register 1 (lower)	R/W	R/W		00000000B
0000DBH	COSRH1	Comparator status register 1 (upper)	R/W	R/W		--000000B
0000DCH	CICRL	Comparator interrupt control register (lower)	R/W	R/W		00000000B
0000DDH	CICRH	Comparator interrupt control register (upper)	R/W	R/W		--000000B
0000DEH	COSRL2	Comparator status register 2 (lower)	R	R		XXXXXXXXB
0000DFH	COSRH2	Comparator status register 2 (upper)	R	R		--XXXXXXB
0000E0H	CIER	Comparator input enable register	R/W	R/W		---11111B
0000E1H	BDR	Bit data register	R/W	R/W	Bit decoder	----XXXXB
0000E2H	BRRL	Bit result register (lower)	R	R		XXXXXXXXB
0000E3H	BRRH	Bit result register (upper)	R	R		XXXXXXXXB
0000E4H	SMR3	Serial mode register 3	R/W	R/W	UART3	00000-00B
0000E5H	SCR3	Serial control register 3	R/W	R/W		00000100B
0000E6H	SIDR3 / SODR3	Input data register 3 / output data register 3	R/W	R/W		XXXXXXXXB
0000E7H	SSR3	Status register 3	R/W	R/W		00001000B
0000E8H	M2CR3	Mode 2 control register 3	R/W	R/W		----1000B
0000E9H	CDCR3	Clock division control register 3	R/W	R/W	Communication prescaler 3	00--0000B
0000EAH	PDL3	Port 3 data latch register	R/W	R/W	Port 3 data latch	00000000B
0000EBH ~ EDH	Prohibited area					
0000EEH	LCRL	LCD control register 0	R/W	R/W	LCD controller / driver	00010000B
0000EFH	LCRH	LCD control register 1	R/W	R/W		00000000B
0000F0H ~ F4H	VRAM	LCD display RAM	R/W	-		XXXXXXXXB
0000F5H ~ F7H	Prohibited area					
0000F8H ~ FFH	External area					

(Continued)

Address	Abbreviation	Register	Byte access	Word access	Resource name	Initial value
001FF0H	PADR0	Program address detection register 0	R/W	R/W	Address match detection	XXXXXXXXB
001FF1H		Program address detection register 1	R/W	R/W		XXXXXXXXB
001FF2H		Program address detection register 2	R/W	R/W		XXXXXXXXB
001FF3H	PADR1	Program address detection register 3	R/W	R/W		XXXXXXXXB
001FF4H		Program address detection register 4	R/W	R/W		XXXXXXXXB
001FF5H		Program address detection register 5	R/W	R/W		XXXXXXXXB

(Continued)

Address	Abbreviation	Register	Byte access	Word access	Resource name	Initial value
003FC0H	UDRL0	UP data register 0 (lower)	R/W	R/W	LPC data buffer array	XXXXXXXXB
003FC1H	UDRH0	UP data register 0 (upper)	R/W	R/W		XXXXXXXXB
003FC2H	UDRL1	UP data register 1 (lower)	R/W	R/W		XXXXXXXXB
003FC3H	UDRH1	UP data register 1 (upper)	R/W	R/W		XXXXXXXXB
003FC4H	UDRL2	UP data register 2 (lower)	R/W	R/W		XXXXXXXXB
003FC5H	UDRH2	UP data register 2 (upper)	R/W	R/W		XXXXXXXXB
003FC6H	UDRL3	UP data register 3 (lower)	R/W	R/W		XXXXXXXXB
003FC7H	UDRH3	UP data register 3 (upper)	R/W	R/W		XXXXXXXXB
003FC8H	UDRL4	UP data register 4 (lower)	R/W	R/W		XXXXXXXXB
003FC9H	UDRH4	UP data register 4 (upper)	R/W	R/W		XXXXXXXXB
003FCAH	UDRL5	UP data register 5 (lower)	R/W	R/W		XXXXXXXXB
003FCBH	UDRH5	UP data register 5 (upper)	R/W	R/W		XXXXXXXXB
003FCCH	UDRL6	UP data register 6 (lower)	R/W	R/W		XXXXXXXXB
003FCDH	UDRH6	UP data register 6 (upper)	R/W	R/W		XXXXXXXXB
003FCEH	UDRL7	UP data register 7 (lower)	R/W	R/W		XXXXXXXXB
003FCFH	UDRH7	UP data register 7 (upper)	R/W	R/W		XXXXXXXXB
003FD0H	UDRL8	UP data register 8 (lower)	R/W	R/W		XXXXXXXXB
003FD1H	UDRH8	UP data register 8 (upper)	R/W	R/W		XXXXXXXXB
003FD2H	UDRL9	UP data register 9 (lower)	R/W	R/W		XXXXXXXXB
003FD3H	UDRH9	UP data register 9 (upper)	R/W	R/W		XXXXXXXXB
003FD4H	UDRLA	UP data register A (lower)	R/W	R/W		XXXXXXXXB
003FD5H	UDRHA	UP data register A (upper)	R/W	R/W		XXXXXXXXB
003FD6H	UDRLB	UP data register B (lower)	R/W	R/W		XXXXXXXXB
003FD7H	UDRHB	UP data register B (upper)	R/W	R/W		XXXXXXXXB
003FD8H	UDRLC	UP data register C (lower)	R/W	R/W		XXXXXXXXB
003FD9H	UDRHC	UP data register C (upper)	R/W	R/W		XXXXXXXXB
003FDAH	UDRLD	UP data register D (lower)	R/W	R/W		XXXXXXXXB
003FDBH	UDRHD	UP data register D (upper)	R/W	R/W		XXXXXXXXB
003FDCH	UDRLE	UP data register E (lower)	R/W	R/W		XXXXXXXXB
003FDDH	UDRHE	UP data register E (upper)	R/W	R/W		XXXXXXXXB
003FDEH	UDRLF	UP data register F (lower)	R/W	R/W		XXXXXXXXB
003FDFH	UDRHF	UP data register F (upper)	R/W	R/W		XXXXXXXXB
003FE0H	DNDL0	DOWN data register 0 (lower)	R	R		XXXXXXXXB
003FE1H	DNDH0	DOWN data register 0 (upper)	R	R		XXXXXXXXB
003FE2H	DNDL1	DOWN data register 1 (lower)	R	R		XXXXXXXXB
003FE3H	DNDH1	DOWN data register 1 (upper)	R	R		XXXXXXXXB

(Continued)

Address	Abbreviation	Register	Byte access	Word access	Resource name	Initial value
003FE4H	DNDL2	DOWN data register 2 (lower)	R	R	LPC data buffer array	XXXXXXXXB
003FE5H	DNDH2	DOWN data register 2 (upper)	R	R		XXXXXXXXB
003FE6H	DNDL3	DOWN data register 3 (lower)	R	R		XXXXXXXXB
003FE7H	DNDH3	DOWN data register 3 (upper)	R	R		XXXXXXXXB
003FE8H	DNDL4	DOWN data register 4 (lower)	R	R		XXXXXXXXB
003FE9H	DNDH4	DOWN data register 4 (upper)	R	R		XXXXXXXXB
003FEAH	DNDL5	DOWN data register 5 (lower)	R	R		XXXXXXXXB
003FEBH	DNDH5	DOWN data register 5 (upper)	R	R		XXXXXXXXB
003FECH	DNDL6	DOWN data register 6 (lower)	R	R		XXXXXXXXB
003FEDH	DNDH6	DOWN data register 6 (upper)	R	R		XXXXXXXXB
003FEEH	DNDL7	DOWN data register 7 (lower)	R	R		XXXXXXXXB
003FEFH	DNDH7	DOWN data register 7 (upper)	R	R		XXXXXXXXB
003FF0H	DBAAL	Data buffer array address register (lower)	R/W	R/W		XXXXXXXXB
003FF1H	DBAAH	Data buffer array address register (upper)	R/W	R/W		XXXXXXXXB
003FF2H ~ 003FFFH	Prohibited area					

- **Meaning of abbreviations used for reading and writing**

  R/W: Read and write enabled

  R: Read-only

  W: Write-only

- **Explanation of initial values**

  0: The bit is initialized to 0.

  1: The bit is initialized to 1.

  X: The initial value of the bit is undefined.

  -: The bit is not used. Its initial value is undefined.

- **Instruction using IO addressing e.g. MOV A, io, is not supported for registers area 003FC0H to 003FFFH.**

■ INTERRUPT FACTORS, INTERRUPT VECTORS, INTERRUPT CONTROL REGISTER

Interrupt cause	EI²OS support	Interrupt vector			Interrupt control register		Priority *2
		Number		Address	ICR	Address	
Reset	X	#08	08H	FFFFDCH	-	-	High
INT9 instruction	X	#09	09H	FFFFD8H	-	-	
Exception processing	X	#10	0AH	FFFFD4H	-	-	
A/D converter conversion termination	O	#11	0BH	FFFFD0H	ICR00	0000B0H*1	
Timebase timer	Δ	#12	0CH	FFFFCCH			
UPI0 IBF / LPC reset	Δ	#13	0DH	FFFFC8H	ICR01	0000B1H*1	
UPI1 IBF	Δ	#14	0EH	FFFFC4H			
UPI2 IBF	Δ	#15	0FH	FFFFC0H	ICR02	0000B2H*1	
UPI3 IBF	Δ	#16	10H	FFFFBCH			
DTP/ext. interrupt channels 0/1 detection	O	#17	11H	FFFFB8H	ICR03	0000B3H*1	
DTP/ext. interrupt channels 2/3 detection	O	#18	12H	FFFFB4H			
DTP/ext. interrupt channels 4/5 detection	O	#19	13H	FFFFB0H	ICR04	0000B4H*1	
Wake-up interrupt detection	Δ	#20	14H	FFFFACH			
UPI0/1/2/3 OBE	Δ	#21	15H	FFFFA8H	ICR05	0000B5H*2	
16-bit PPG timer 1	O	#22	16H	FFFFA4H			
PS/2 interface 0/1	Δ	#23	17H	FFFFA0H	ICR06	0000B6H*1	
PS/2 interface 2	Δ	#24	18H	FFFF9CH			
Watch timer	Δ	#25	19H	FFFF98H	ICR07	0000B7H*1	
I²C transfer complete / bus error	Δ	#26	1AH	FFFF94H			
16-bit PPG timer 2/3	O	#27	1BH	FFFF90H	ICR08	0000B8H*1	
Voltage comparator 1	Δ	#28	1CH	FFFF8CH			
MI²C transfer complete / bus error	Δ	#29	1DH	FFFF88H	ICR09	0000B9H*1	
Voltage comparator 2	Δ	#30	1EH	FFFF84H			
I²C timeout / standby wake-up	Δ	#31	1FH	FFFF80H	ICR10	0000BAH*1	
16-bit reload timer 1/2 underflow	O	#32	20H	FFFF7CH			
MI²C timeout / standby wake-up	Δ	#33	21H	FFFF78H	ICR11	0000BBH*1	
16-bit reload timer 3/4 underflow	O	#34	22H	FFFF74H			
UART1 receive	⊙	#35	23H	FFFF70H	ICR12	0000BCH*1	
UART1 send	Δ	#36	24H	FFFF6CH			
UART2 receive	⊙	#37	25H	FFFF68H	ICR13	0000BDH*1	
UART2 send	Δ	#38	26H	FFFF64H			
UART3 receive	⊙	#39	27H	FFFF60H	ICR14	0000BEH*1	
UART3 send	Δ	#40	28H	FFFF5CH			
Flash memory status	Δ	#41	29H	FFFF58H	ICR15	0000BFH*1	
Delayed interrupt generator module	Δ	#42	2AH	FFFF54H			Low

O: Can be used and interrupt request flag is cleared by EI²OS interrupt clear signal.

X: Cannot be used.

⊙: Can be used and support the EI²OS stop request.

Δ: Can be used.

*1: - For peripheral functions that share the ICR register, the interrupt level will be the same.

- If the extended intelligent I/O service is to be used with a peripheral function that shares the ICR register with another peripheral function, the service can be started by either of the function. And if EI^{2}OS clear is supported, both interrupt request flags for the two interrupt causes are cleared by EI^{2}OS interrupt clear signal. It is recommended to mask either of the interrupt request during the use of EI^{2}OS.
- EI^{2}OS service cannot be started multiple times simultaneously. Interrupt other than the operating interrupt is masked during EI^{2}OS operation. It is recommended to mask either of the interrupt requests during the use of EI^{2}OS.

*2: This priority is applied when interrupts of the same level occur simultaneously.

■ PERIPHERAL RESOURCES

1. Low-power Consumption Control Circuit

The MB90370 series has the following CPU operating mode selected by the configuration of an operating clock and clock operation control.

● **Clock Mode**

- PLL clock mode

  In this mode, a PLL clock that is a multiple of the oscillation clock (HCLK) is used to operate the CPU and peripheral functions.

- Main clock mode

  In this mode, the main clock, with the oscillation clock (HCLK) frequency divided by 2 is used to operate the CPU and peripheral functions. In the main clock mode, the PLL multiplier circuit is inactive.

- Sub-clock mode

  In this mode, the sub-clock, with the sub-clock (SCLK) frequency divided by 4 is used to operate the CPU and peripheral functions. In the sub-clock mode, the main clock and PLL multiplier circuit are inactive.

Reference

For the clock mode, see Section 4.4 "Clock Mode" of the MB90370 series H/W manual.

● **CPU Intermittent Operating Mode**

In this mode, the CPU is operated intermittently while high-speed clock pluses are supplied to peripheral functions, thereby reducing power consumption. In this mode, intermittent clock pulses are supplied only to the CPU while it is accessing a register, internal memory, peripheral function, or external unit.

● **Standby Mode**

In this mode, the low-power consumption control circuit stops supplying the clock to the CPU (sleep mode) or the CPU and peripheral functions (timebase timer mode) or stops the oscillation clock itself (stop mode), thereby reducing power consumption.

- PLL sleep mode

  The PLL sleep mode is activated to stop the CPU operating clock in the PLL clock mode. Components excluding the CPU operate on the PLL clock.

- Main sleep mode

  The main sleep mode is activated to stop the CPU operating clock in the main clock mode. Components excluding the CPU operate on the main clock.

- Sub-sleep mode

  The sub-sleep mode is activated to stop the CPU operating clock in the sub-clock mode. Components excluding the CPU operate on the divided-by-four sub-clock.

- Timebase timer mode

  The timebase timer mode causes the operation of functions, excluding the oscillation clock, timebase timer, and watch timer, to stop. All functions other than the timebase timer and watch timer are inactivated.

- Watch mode and main watch mode

  The watch mode and main watch mode operates the watch timer only. The sub-clock operates but the main clock and PLL multiplier circuit stop.

- Stop mode

  The stop mode causes the oscillation to stop. All functions are inactivated.

Note

Because the stop mode turns the oscillation clock off, data can be retained by the lowest power consumption.

(1) Register configuration

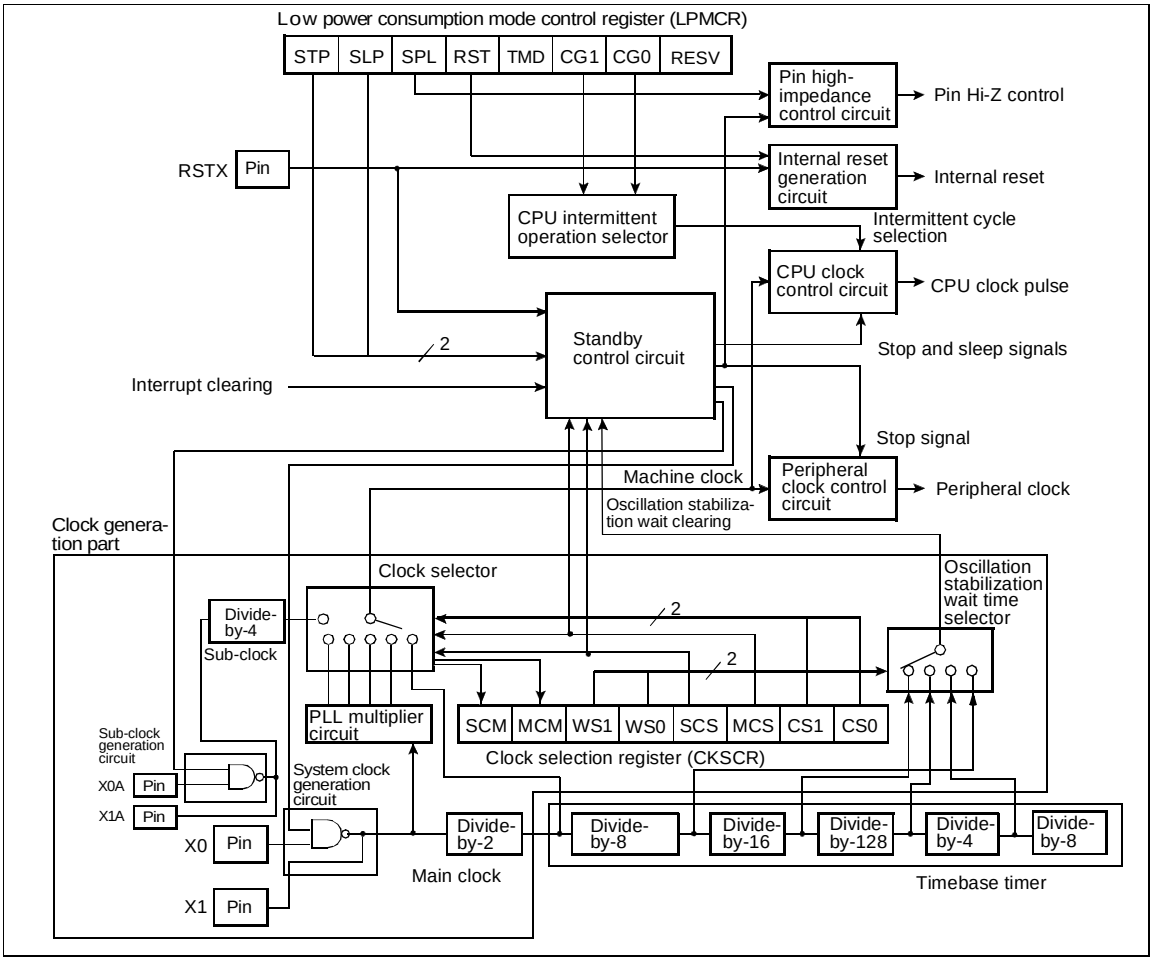
Clock Selection Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 0000A1H	SCM	MCM	WS1	WS0	SCS	MCS	CS1	CS0	CKSCR
Read/write ⇨	R	R	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	1	1	1	1	1	1	0	0	

Lower Power Consumption Mode Control Register

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 0000A0H	STP	SLP	SPL	RST	TMD	CG1	CG0	Reserved	LPMCR
Read/write ⇨	W	W	R/W	W	W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	1	1	0	0	0	

(2) Block diagram



2. I/O Ports

(1) Outline of I/O ports

Each I/O port outputs data from the CPU to the I/O pins or inputs signals from the I/O pins to the CPU as directed by the port data register (PDR). Each CMOS I/O port can also designate the direction of a data flow (input or output) at the I/O pins in bit units using the port data direction register (DDR). Or N-channel open-drain port can designate the direction of a data flow (input or output) at the I/O pins in bit units using the port data register (PDR). The function of each port and the resources using it are described below:

- Port 0 : General-purpose I/O port/resource (Key-on wake-up interrupt)
- Port 1 : General-purpose I/O port
- Port 2 : General-purpose I/O port
- Port 3 : General-purpose I/O port/resource (A/D converter external trigger)
- Port 4 : General-purpose I/O port/resource (PS/2 interface / serial IRQ controller)
- Port 5 : General-purpose I/O port/resource (LPC interface)
- Port 6 : General-purpose I/O port/resource (DTP / UART1)
- Port 7 : General-purpose I/O port/resource (UART1 / UART2 / UART3 / PPG1)
- Port 8 : General-purpose I/O port/resource (Multi-address I^2C)
- Port 9 : General-purpose I/O port/resource (I^2C / Multi-address I^2C)
- Port A : General-purpose I/O port/resource (Comparator)
- Port B : General-purpose I/O port/resource (Comparator)
- Port C : General-purpose I/O port/resource (Comparator / A/D converter)
- Port D : General-purpose I/O port/resource (A/D converter / D/A converter / PPG2 / PPG3)
- Port E : General-purpose I/O port/resource (Reload timer1 ~ 4 / LCD controller)
- Port F : General-purpose I/O port/resource (LCD controller)

(2) Register configuration

Register	Read/Write	Address	Initial value
Port 0 data register (PDR0)	R/W	000000_H	$XXXXXXXX_B$
Port 1 data register (PDR1)	R/W	000001_H	$XXXXXXXX_B$
Port 2 data register (PDR2)	R/W	000002_H	$XXXXXXXX_B$
Port 3 data register (PDR3)	R/W	000003_H	$XXXXXXXX_B$
Port 4 data register (PDR4)	R/W	000004_H	$X1111111_B$
Port 5 data register (PDR5)	R/W	000005_H	$XXXXXXXX_B$
Port 6 data register (PDR6)	R/W	000006_H	$XXXXXXXX_B$
Port 7 data register (PDR7)	R/W	000007_H	$XXXXXXXX_B$
Port 8 data register (PDR8)	R/W	000008_H	$-----111_B$
Port 9 data register (PDR9)	R/W	000009_H	$--111111_B$
Port A data register (PDRA)	R/W	$00000A_H$	$-XXXXXXX_B$
Port B data register (PDRB)	R/W	$00000B_H$	$XXXXXXXX_B$
Port C data register (PDRC)	R/W	$00000C_H$	$XXXXXXXX_B$
Port D data register (PDRD)	R/W	$00000D_H$	$XXXXXXXX_B$
Port E data register (PDRE)	R/W	$00000E_H$	$XXXXXXXX_B$
Port F data register (PDRF)	R/W	$00000F_H$	$XXXXXXXX_B$
Port 0 data direction register (DDR0)	R/W	000010_H	00000000_B

Register	Read/Write	Address	Initial value
Port 1 data direction register (DDR1)	R/W	000011H	00000000B
Port 2 data direction register (DDR2)	R/W	000012H	00000000B
Port 3 data direction register (DDR3)	R/W	000013H	00000000B
Port 4 data direction register (DDR4)	R/W	000014H	0-------B
Port 5 data direction register (DDR5)	R/W	000015H	00000000B
Port 6 data direction register (DDR6)	R/W	000016H	00000000B
Port 7 data direction register (DDR7)	R/W	000017H	00000000B
Port A data direction register (DDRA)	R/W	00001AH	-0000000B
Port B data direction register (DDRB)	R/W	00001BH	00000000B
Port C data direction register (DDRC)	R/W	00001CH	00000000B
Port D data direction register (DDRD)	R/W	00001DH	00000000B
Port E data direction register (DDRE)	R/W	00001EH	00000000B
Port F data direction register (DDRF)	R/W	00001FH	00000000B
Analog data input enable register (ADER1)	R/W	00002AH	11111111B
Analog data input enable register (ADER2)	R/W	00002BH	----1111B
Comparator input enable register (CIER)	R/W	0000E0H	---11111B
LCD control register 1 (LCRH)	R/W	0000EFH	00000000B
Port 0 pull-up resistor setting register (RDR0)	R/W	00008CH	00000000B
Port 1 pull-up resistor setting register (RDR1)	R/W	00008DH	00000000B
Port 2 pull-up resistor setting register (RDR2)	R/W	00008EH	00000000B
Port 3 pull-up resistor setting register (RDR3)	R/W	00008FH	00000000B
Port 3 data latch register (PDL3)	R/W	0000EAH	00000000B

R/W: Read/write enabled

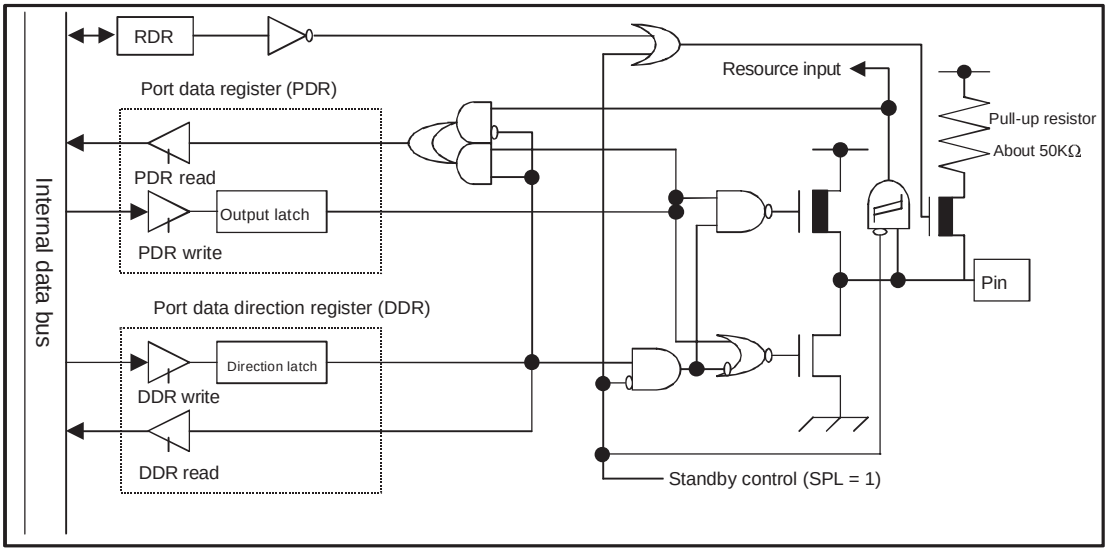
R : Read-only

X : Undefined

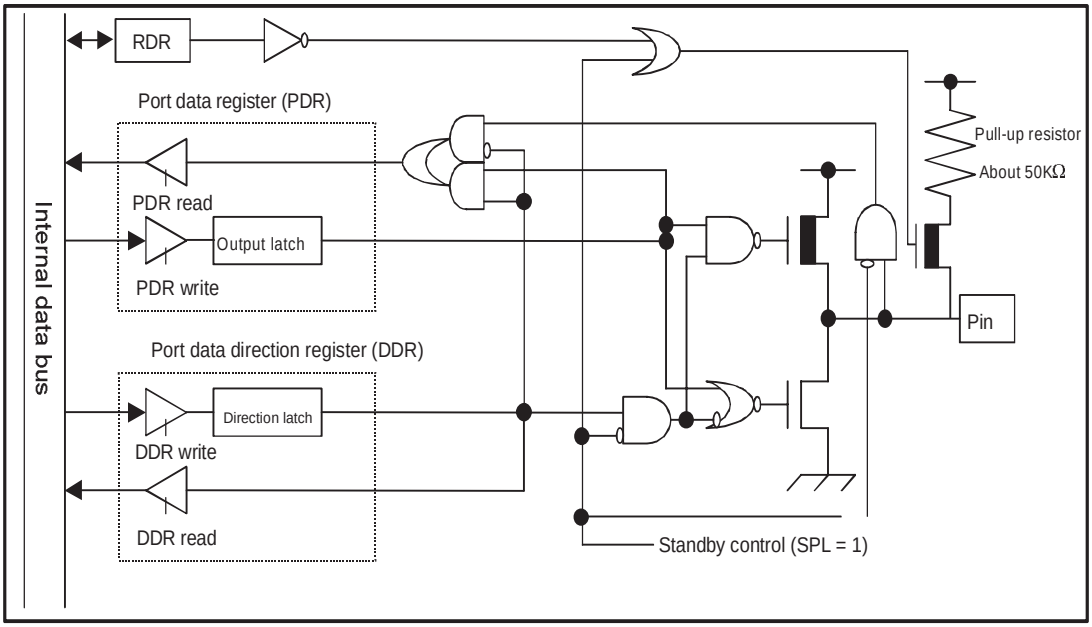
– : Not used

(3) Block diagram of I/O ports

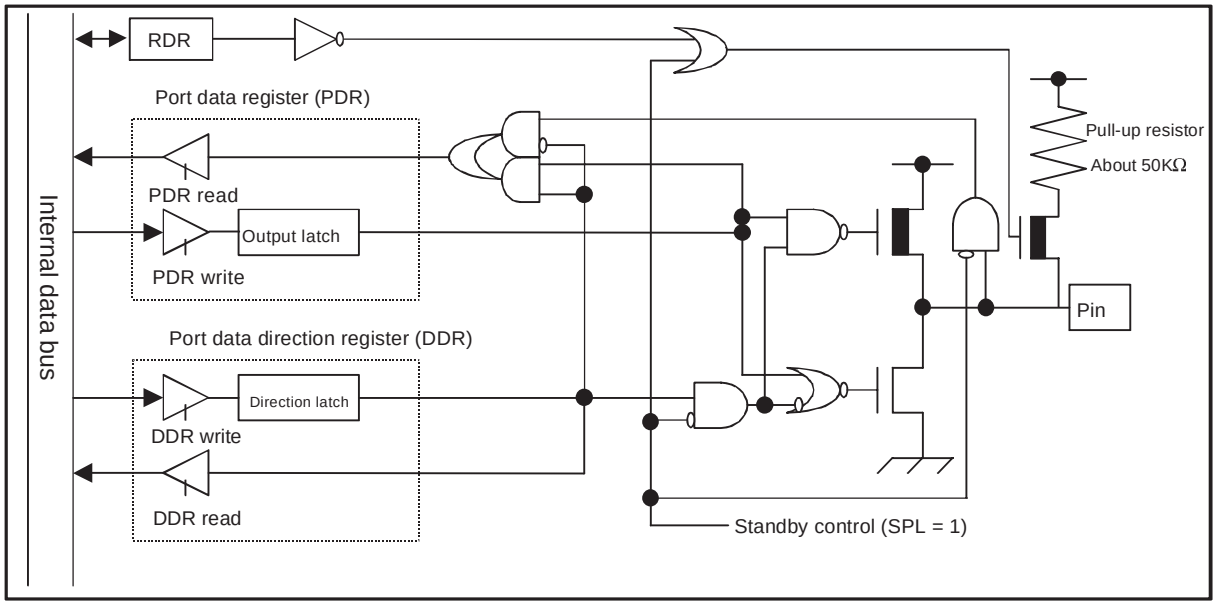
- Block diagram of port 0 pins



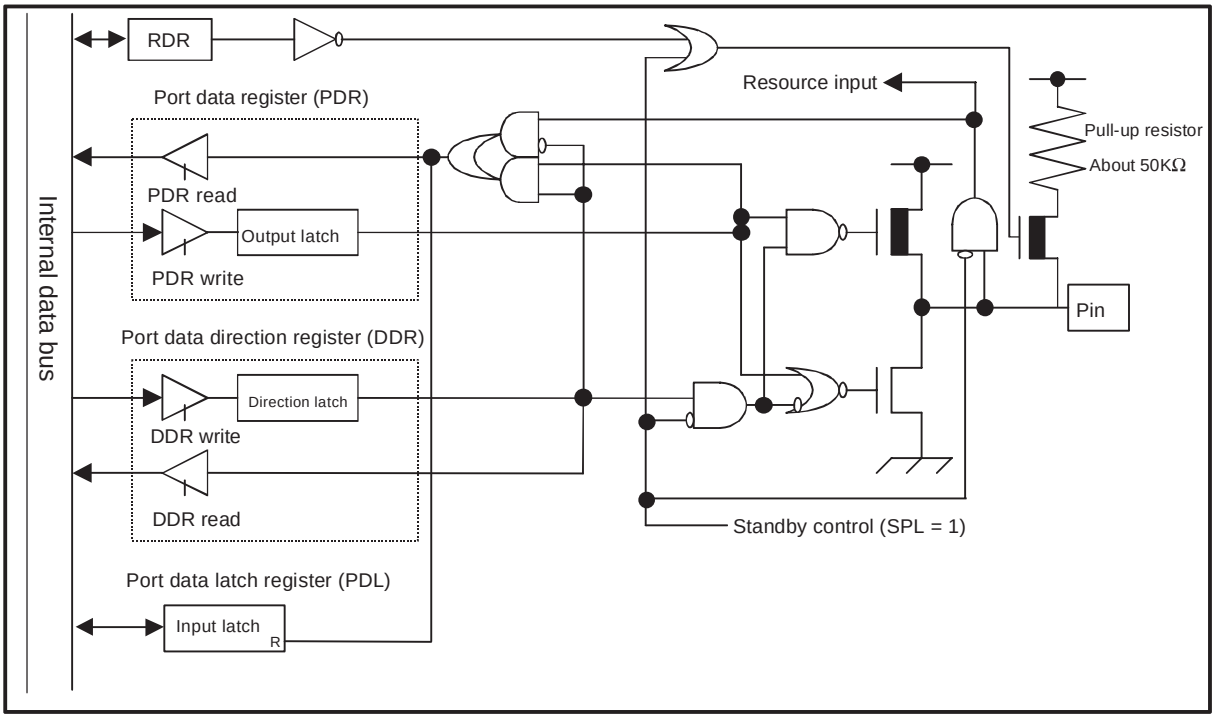
- Block diagram of port 1 pins



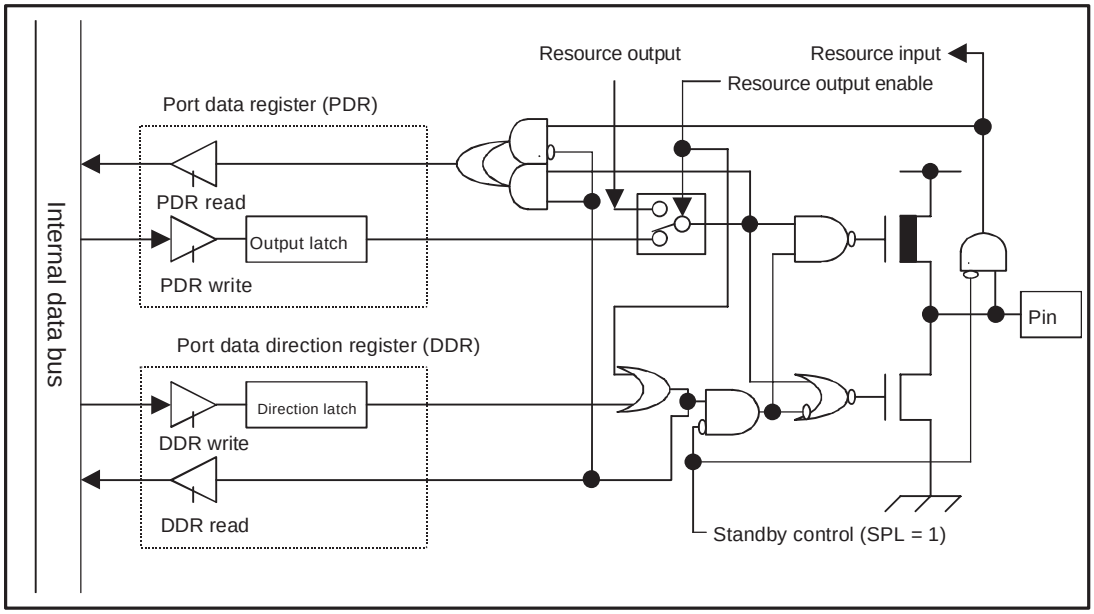
• Block diagram of port 2 pins



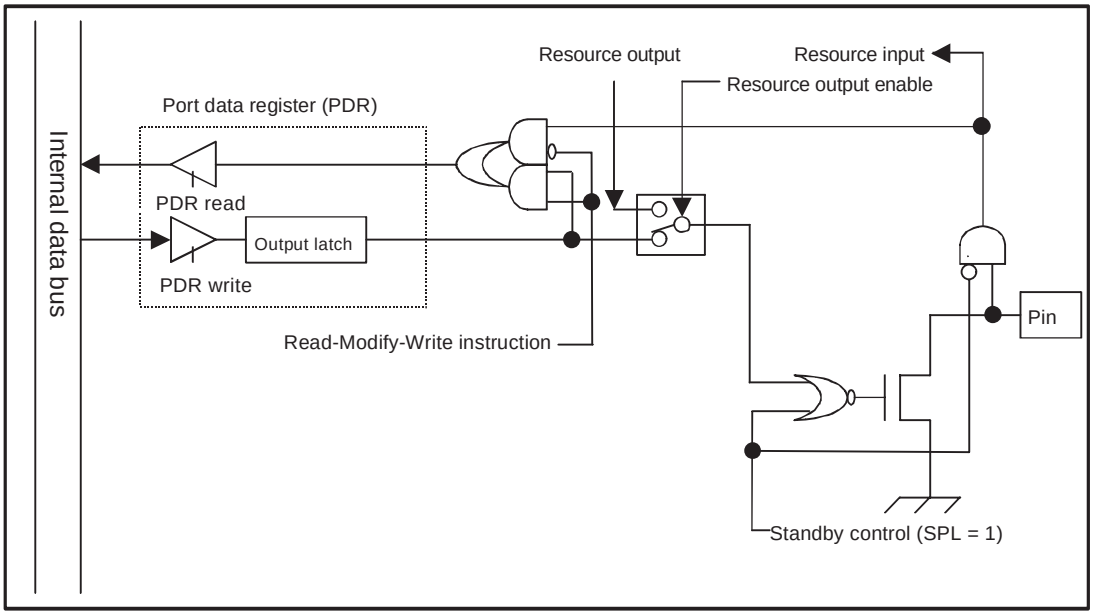
• Block diagram of port 3 pins



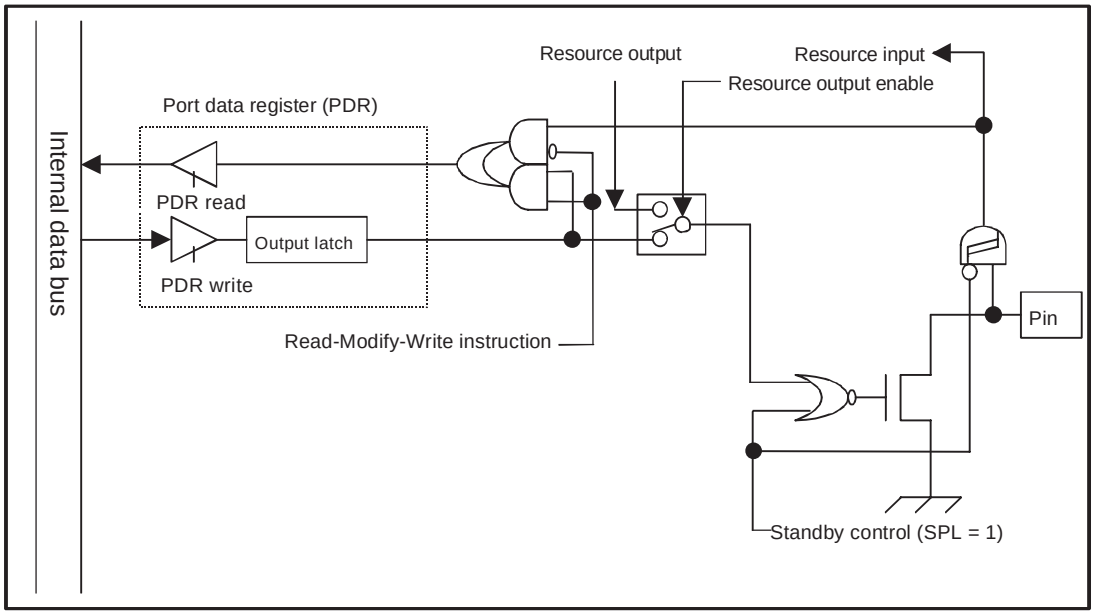
• Block diagram of port 47 pin



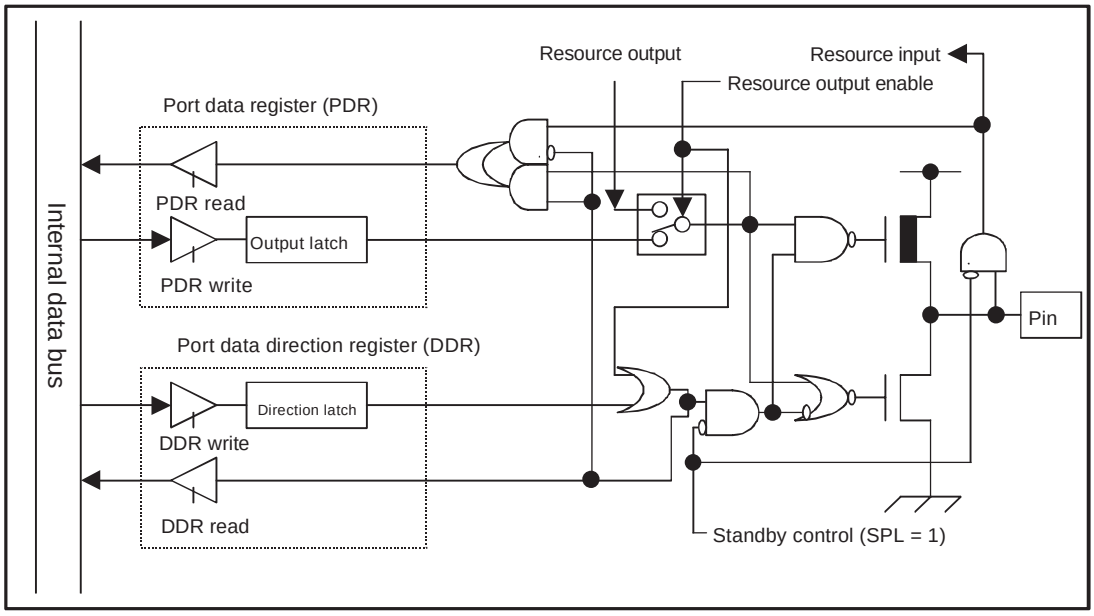
• Block diagram of port 46 pin



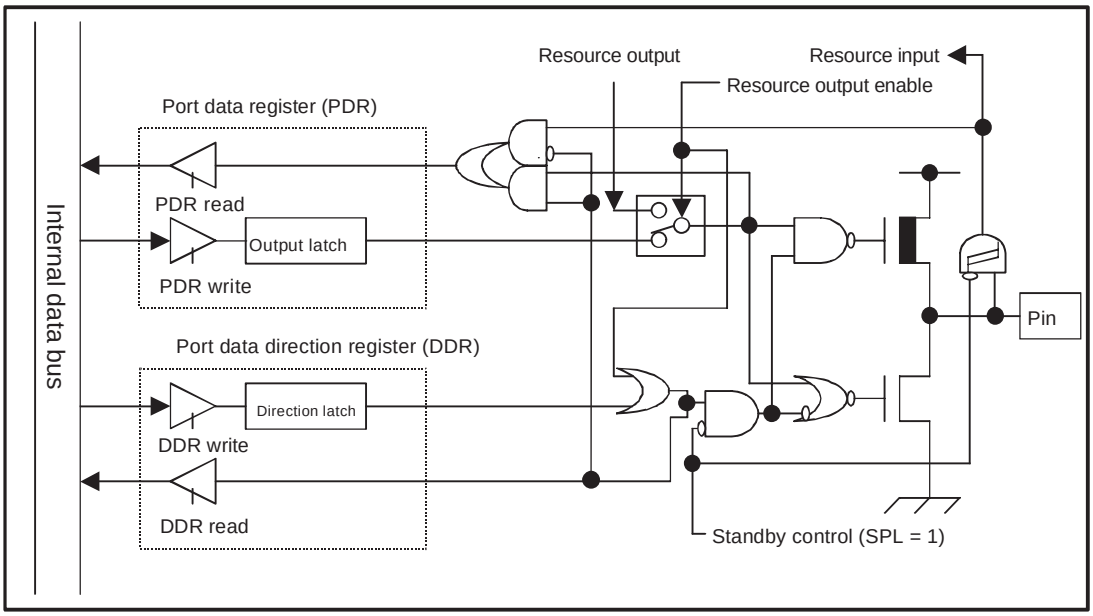
- Block diagram of port 45 ~ 40 pins



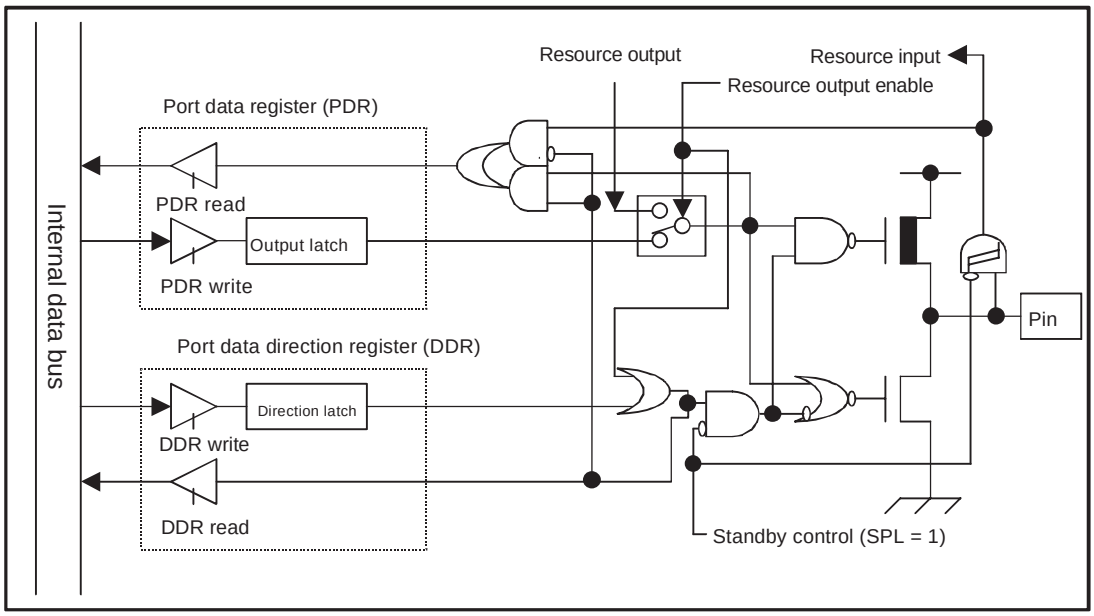
- Block diagram of port 5 pins



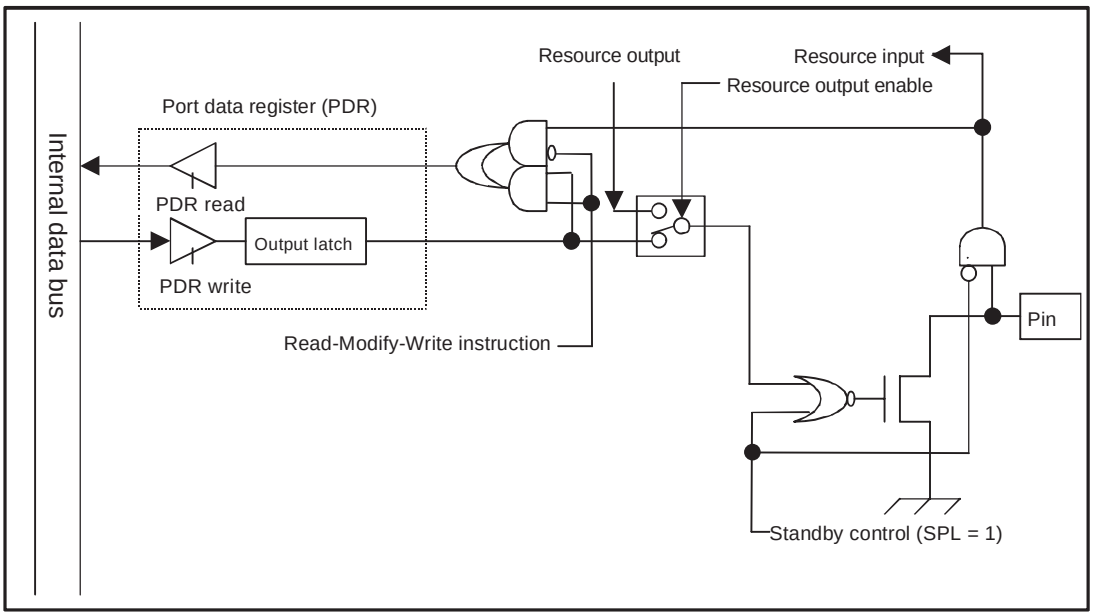
- Block diagram of port 6 pins



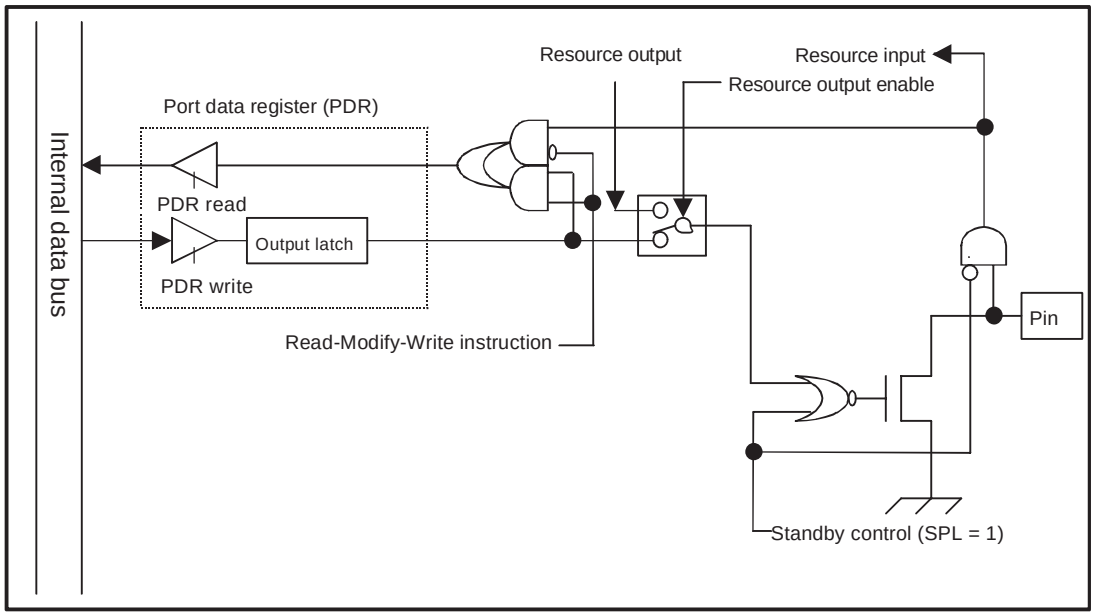
- Block diagram of port 7 pins



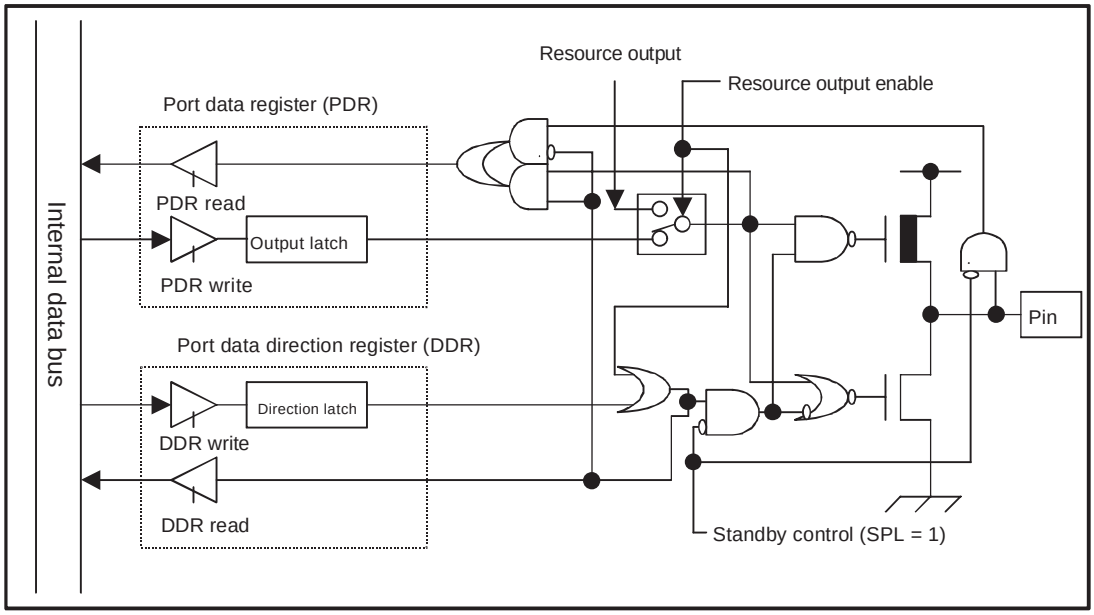
• Block diagram of port 8 pins



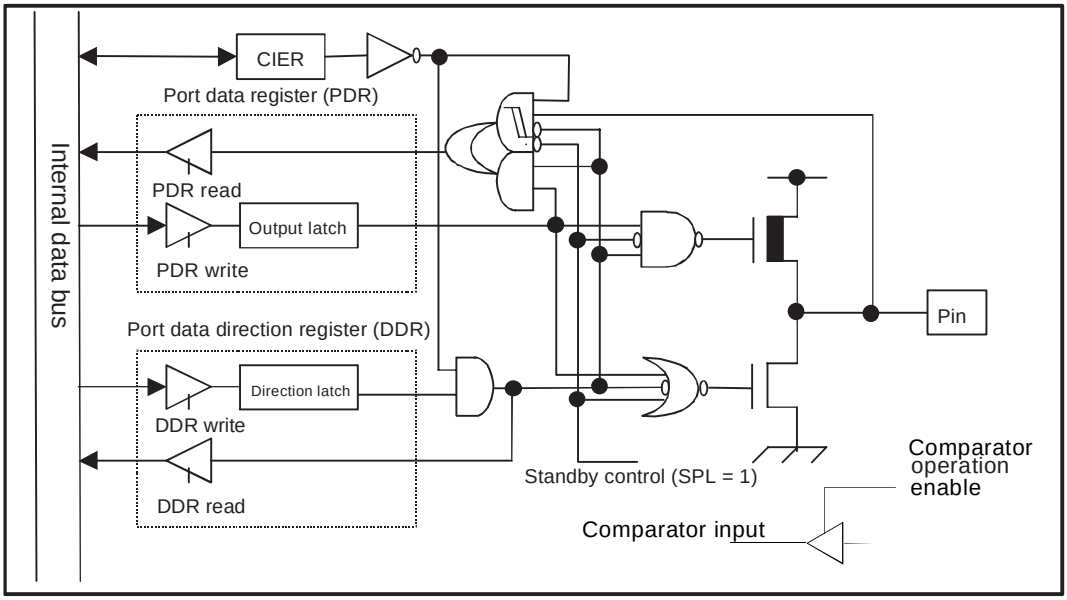
• Block diagram of port 9 pins



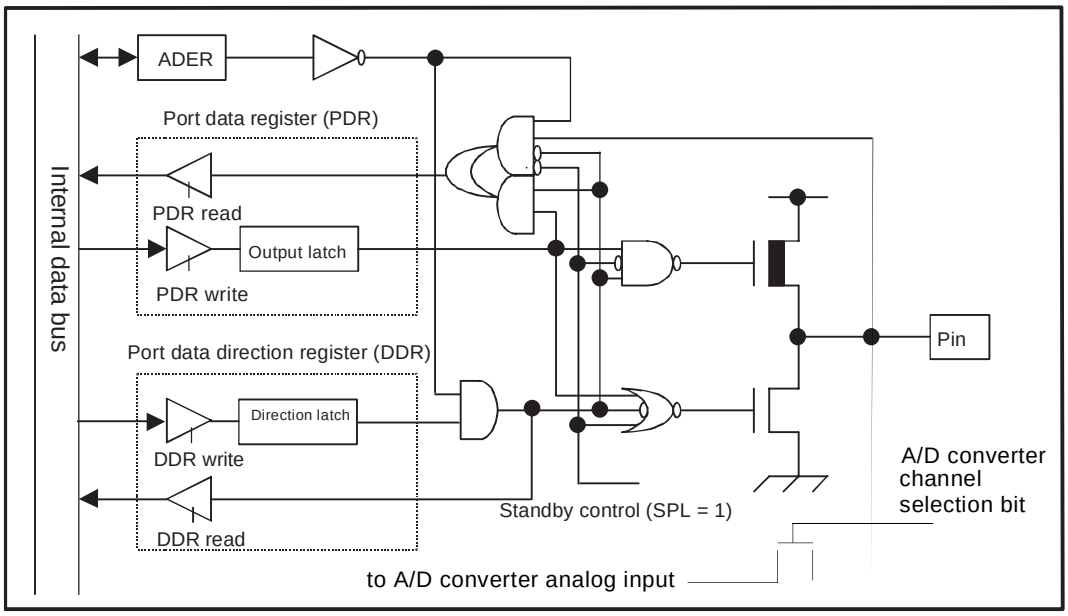
• Block diagram of port A pins



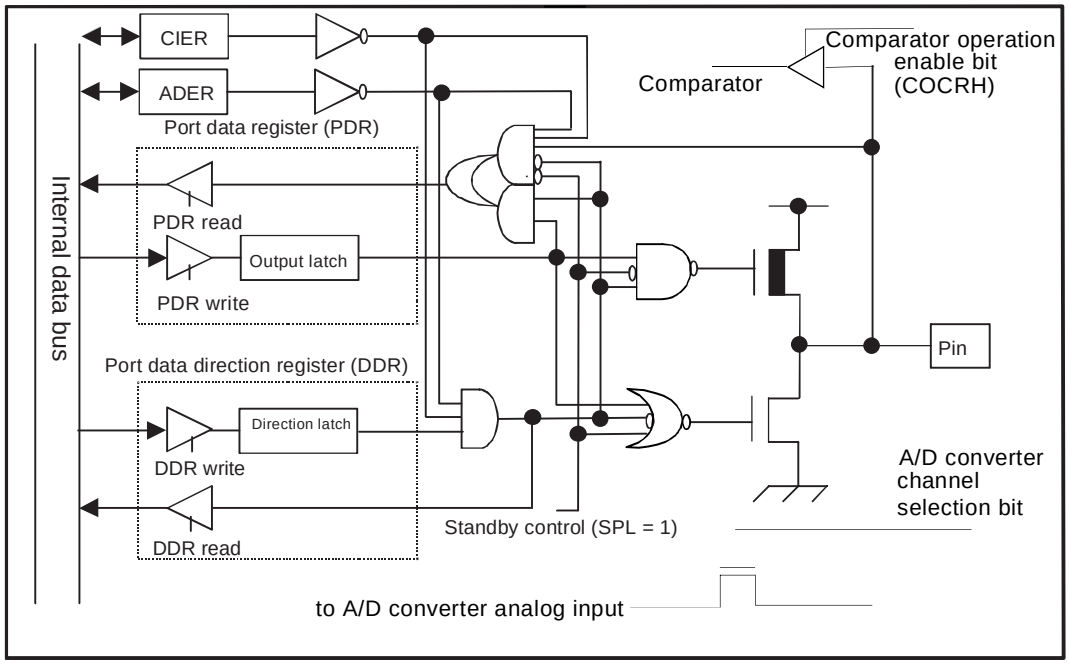
• Block diagram of port B pins



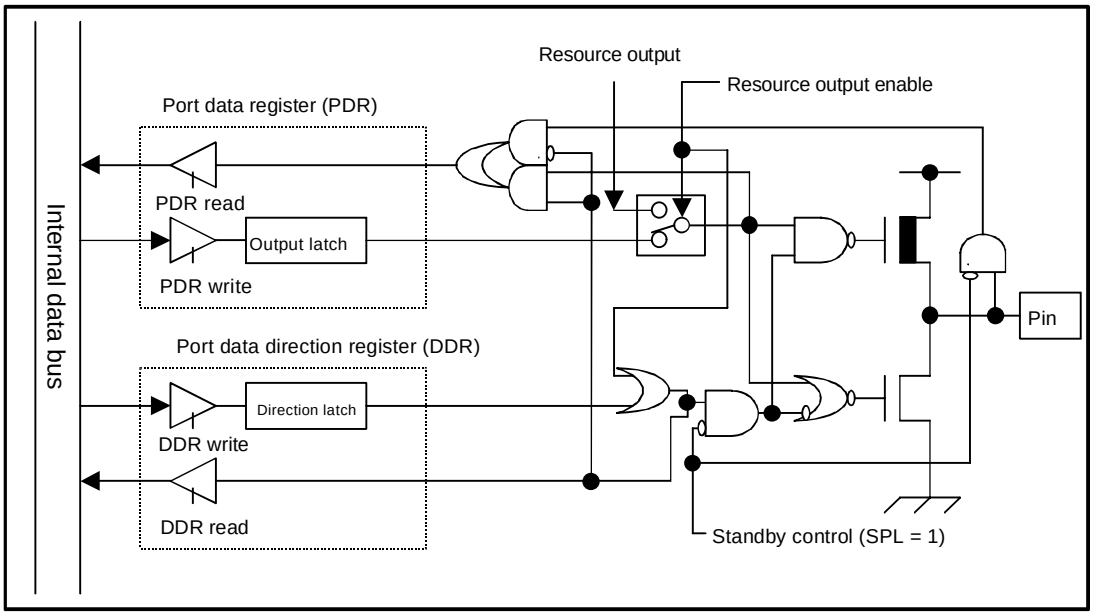
• Block diagram of port C7 ~ C3 pins



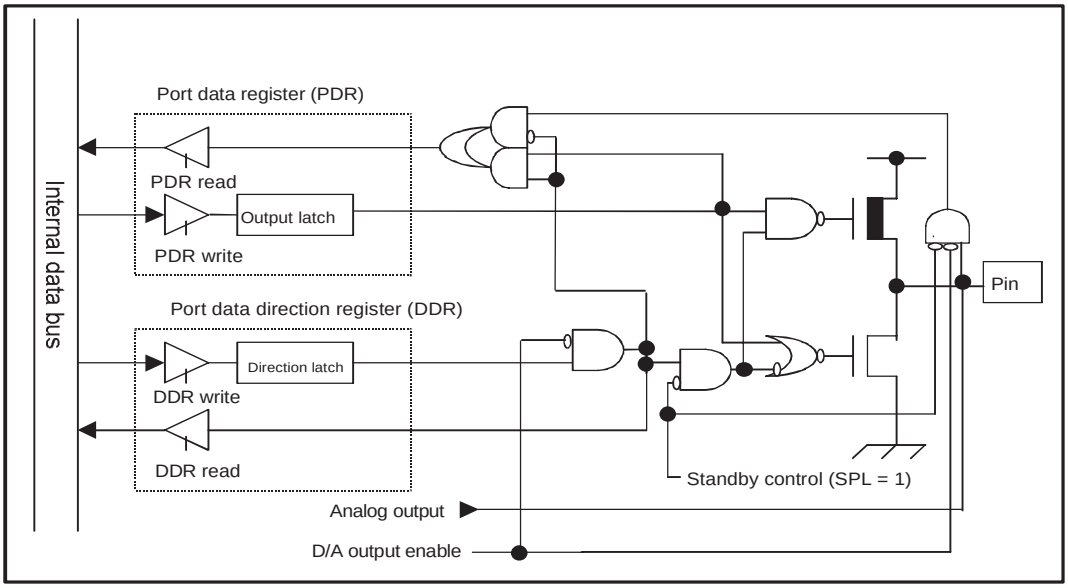
• Block diagram of port C2 ~ C0 pins



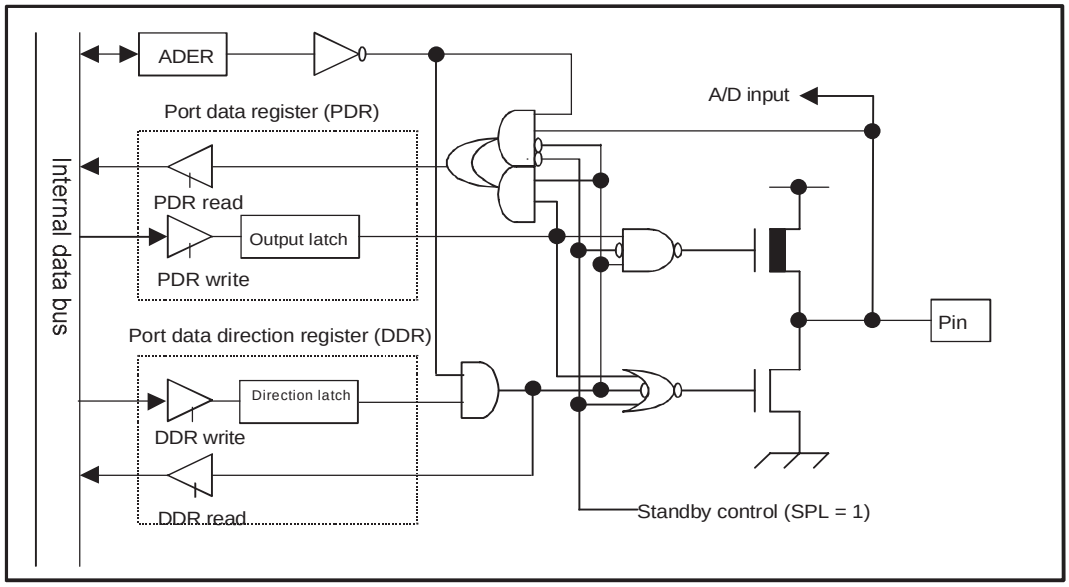
• Block diagram of port D7 ~ D6 pins



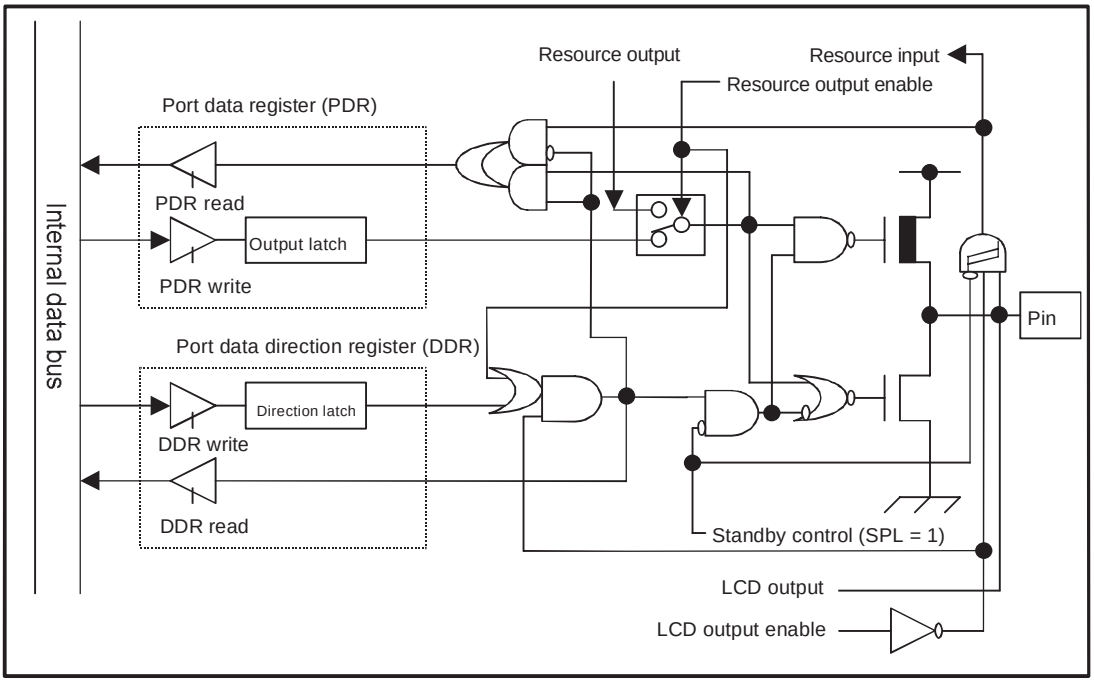
• Block diagram of port D5 ~ D4 pins



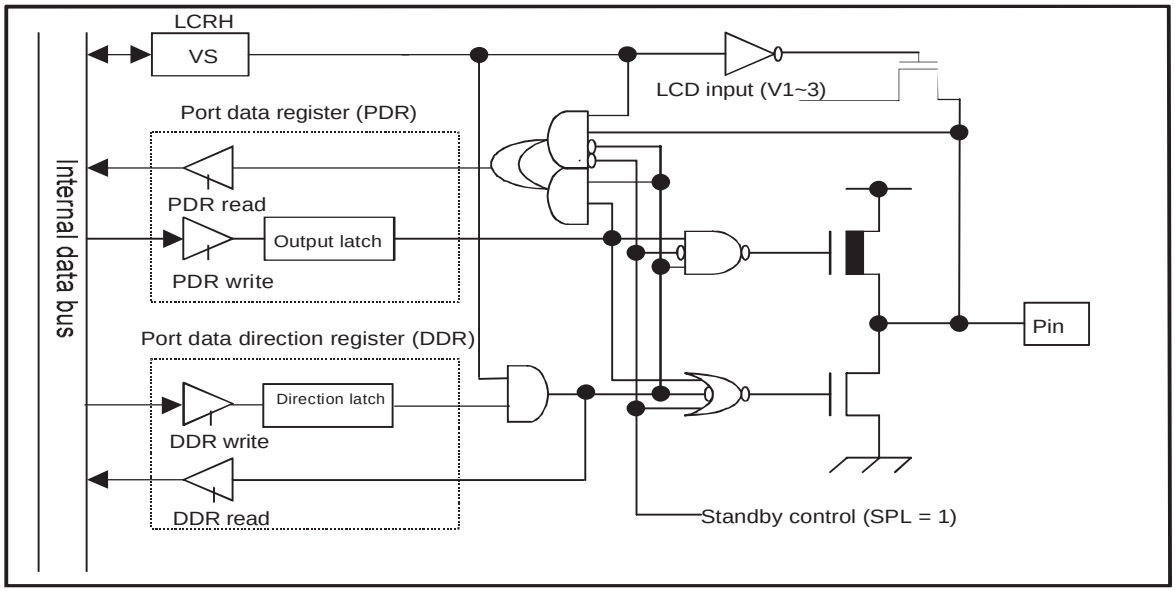
• Block diagram of port D3 ~ D0 pins



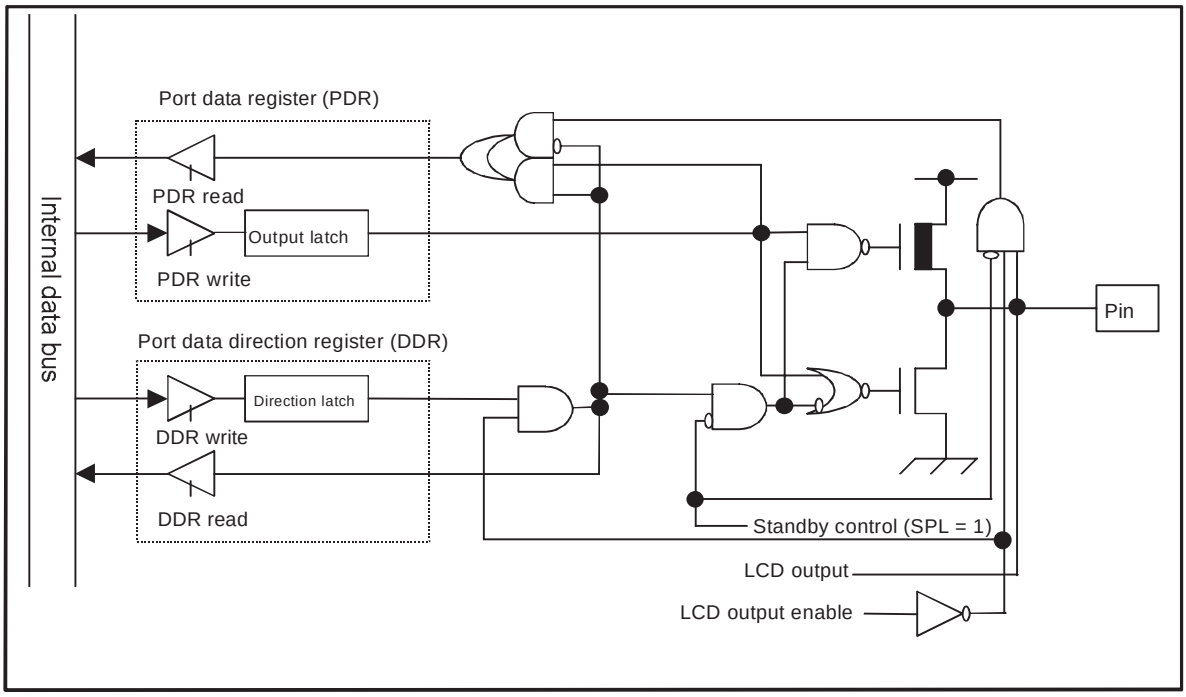
• Block diagram of port E pins



• Block diagram of port F7 ~ F5 pins



• Block diagram of port F4 ~ F0 pins



3. Timebase timer

The timebase timer is an 18-bit free-running counter (timebase counter) that counts up in synchronization with the internal count clock (one-half of the source oscillation).

Features of timebase timer :

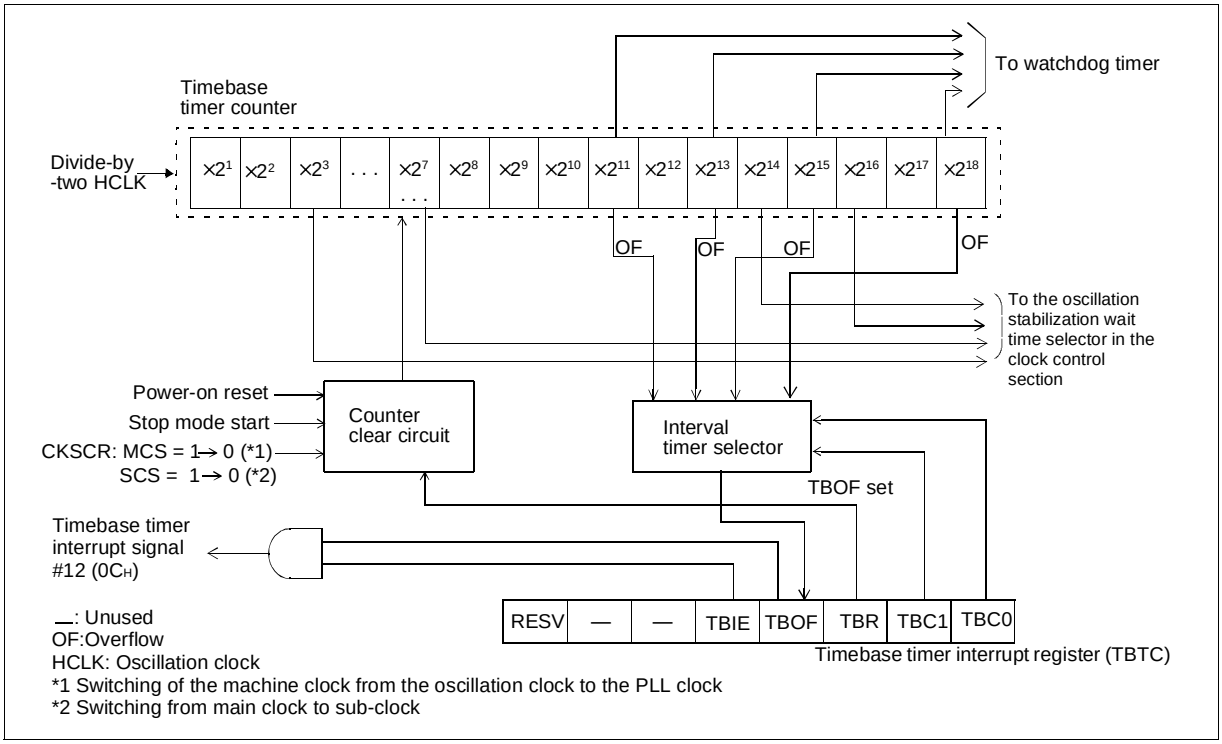
- Interrupt generated when counter overflow
- EI²OS supported
- Interval timer function :
  An interrupt generated at four different time intervals
- Clock supply function :
  Four different clock can be selected as watchdog timer's count clock
  Supply clock for oscillation stabilization

(1) Register configuration

Timebase Timer Control Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 0000A9H	Reserved	—	—	TBIE	TBOF	TBR	TBC1	TBC0	TBTC
Read/write ⇨	R/W	—	—	R/W	R/W	W	R/W	R/W	
Initial value ⇨	1	—	—	0	0	1	0	0	

(2) Block diagram of timebase timer



4. Watchdog timer

The watchdog timer is a 2-bit counter that uses the timebase timer's supply clock as the count clock. After activation, if the watchdog timer is not cleared within a given period, the CPU will be reset.

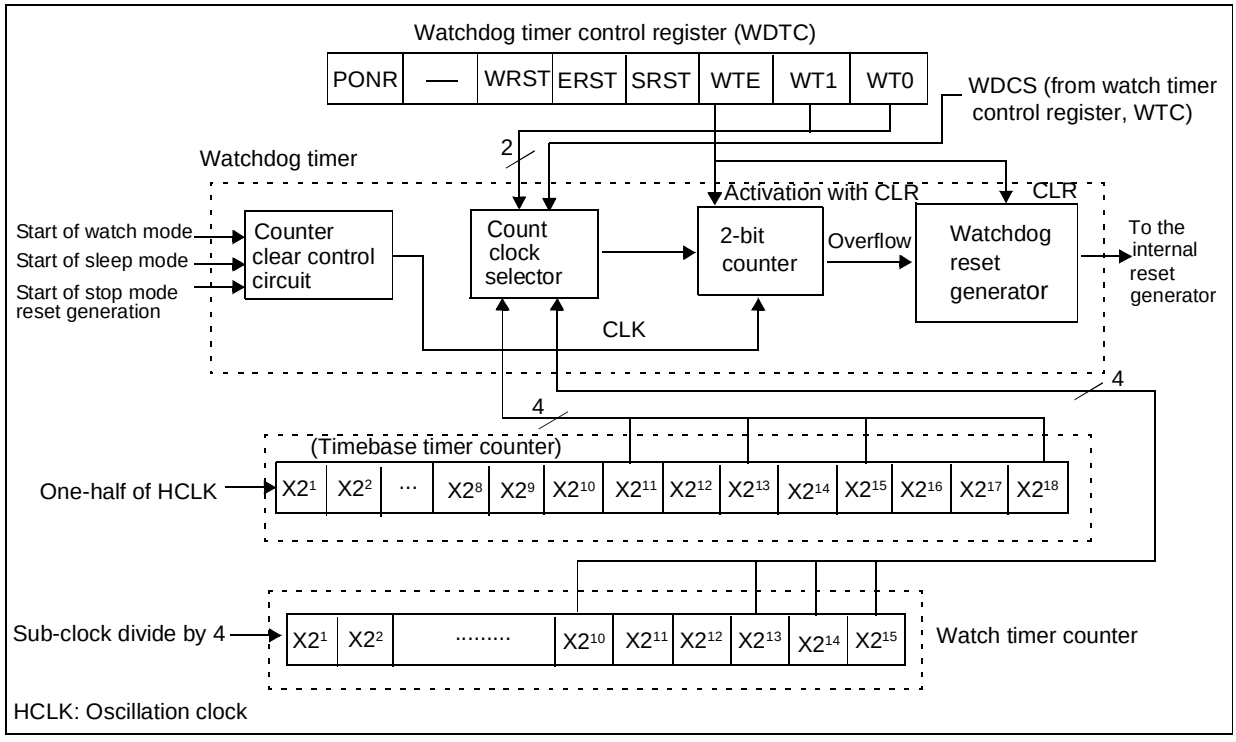
- Features of watchdog timer :
  Reset CPU at four different time intervals
  Status bits to indicate the reset causes

(1) Register configuration of watchdog timer

Watchdog Timer Control Register

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 0000A8H	PONR	—	WRST	ERST	SRST	WTE	WT1	WT0	WDTC
Read/write ⇨	R	—	R	R	R	W	W	W	
Initial value ⇨	X	—	X	X	X	1	1	1	

(2) Block diagram of watchdog timer



5. Watch timer

The watch timer is a 15-bit timer that uses sub-clocks and can generate an interval interrupt. It can also be used as the watchdog timer clock source and sub-clock oscillation wait time.

Features of the watch timer :

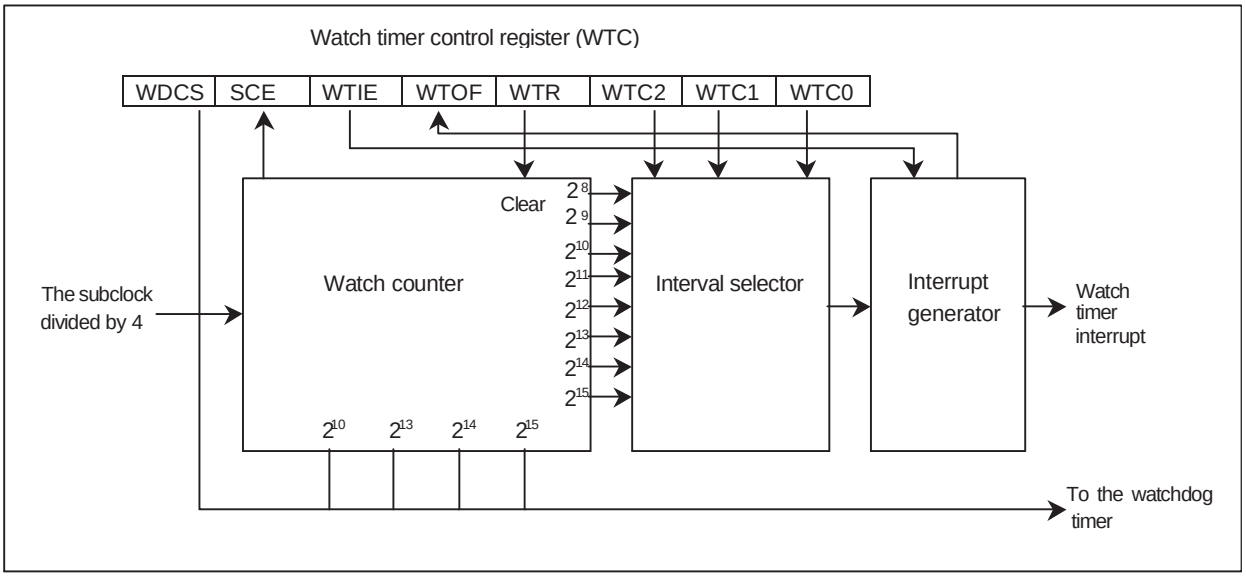
- Provides the watchdog timer clock source
- Sub-clock oscillation stabilization wait timer function
- Interval timer function that generates interrupts in a given cycle

(1) Register configuration of watch timer

Watch Timer Control Register

	7	6	5	4	3	2	1	0	
									⇦ Bit number
Address: 0000AA$_H$	WDCS	SCE	WTIE	WTOF	WTR	WTC2	WTC1	WTC0	WTC
Read/write ⇨	R/W	R	R/W	R/W	W	R/W	R/W	R/W	
Initial value ⇨	1	0	0	0	1	0	0	0	

(2) Block diagram of watch timer



6. 16-bit PPG timer (x 3)

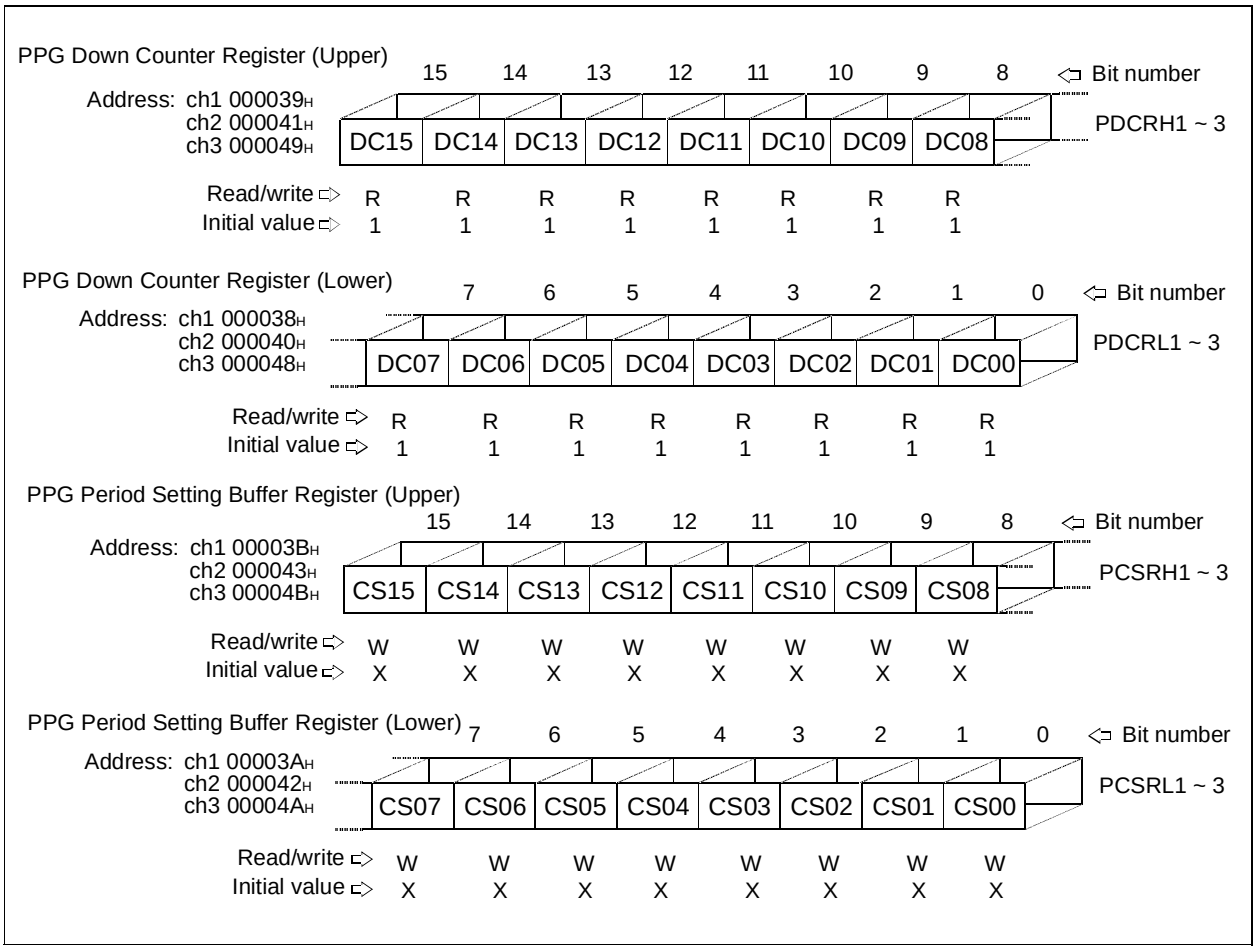
The 16-bit PPG (Programmable Pulse Generator) timer consists of a 16-bit down counter, prescaler, 16-bit period setting register, 16-bit duty setting register, 16-bit control register and a PPG output pin.

Features of 16-bit PPG timer :

- 8 types of counter operation clock (ϕ, ϕ/2, ϕ/4, ϕ/8, ϕ/16, ϕ/32, ϕ/64, ϕ/128) can be selected (ϕ is the machine clock)
- An interrupt is generated when there is a trigger or an counter borrow or when PPG rising (normal polarity) / PPG falling (inverted polarity)
- PPG output operation

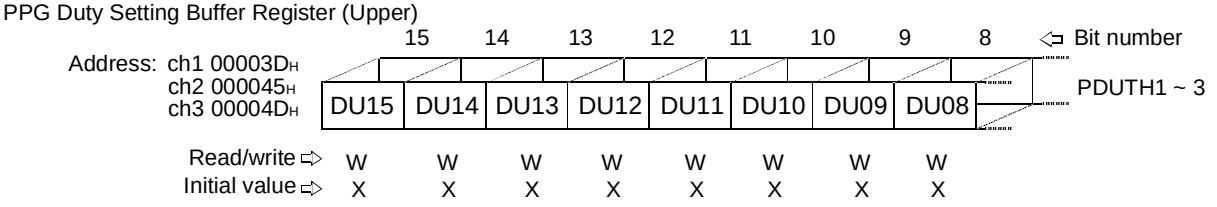
The 16-bit PPG timer can output pulse waveforms with variable period and duty ratio. Also, it can be used as D/A converter in conjunction with an external circuit.

(1) Register configuration of PPG timer

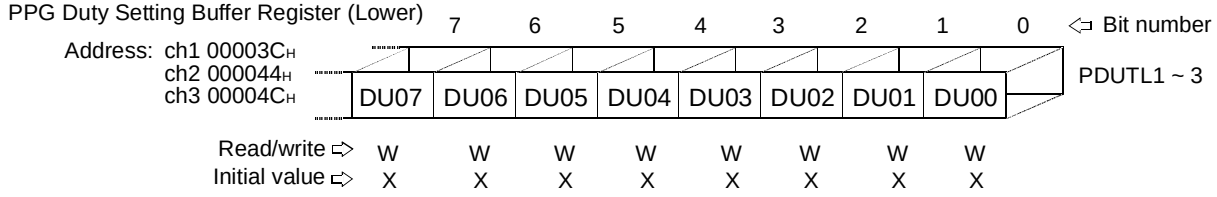


(Continued)

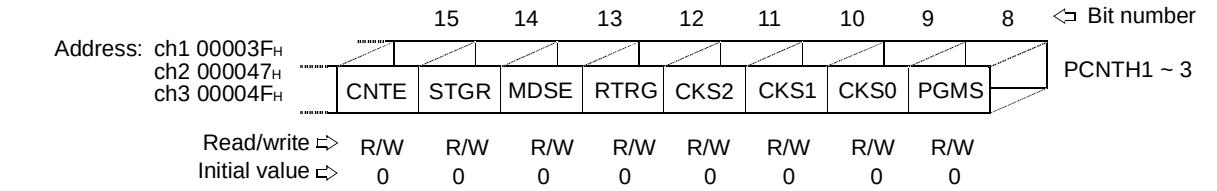
PPG Duty Setting Buffer Register (Upper)



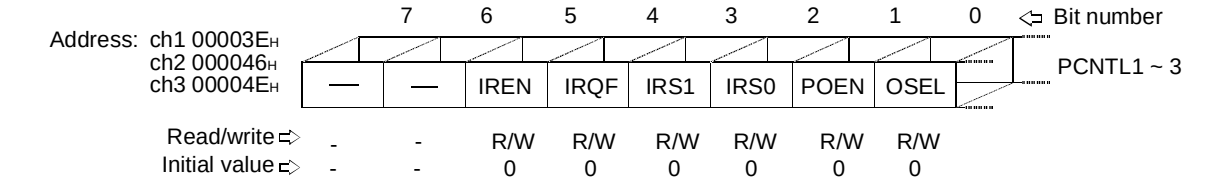
PPG Duty Setting Buffer Register (Lower)



PPG Control Status Register (Upper)

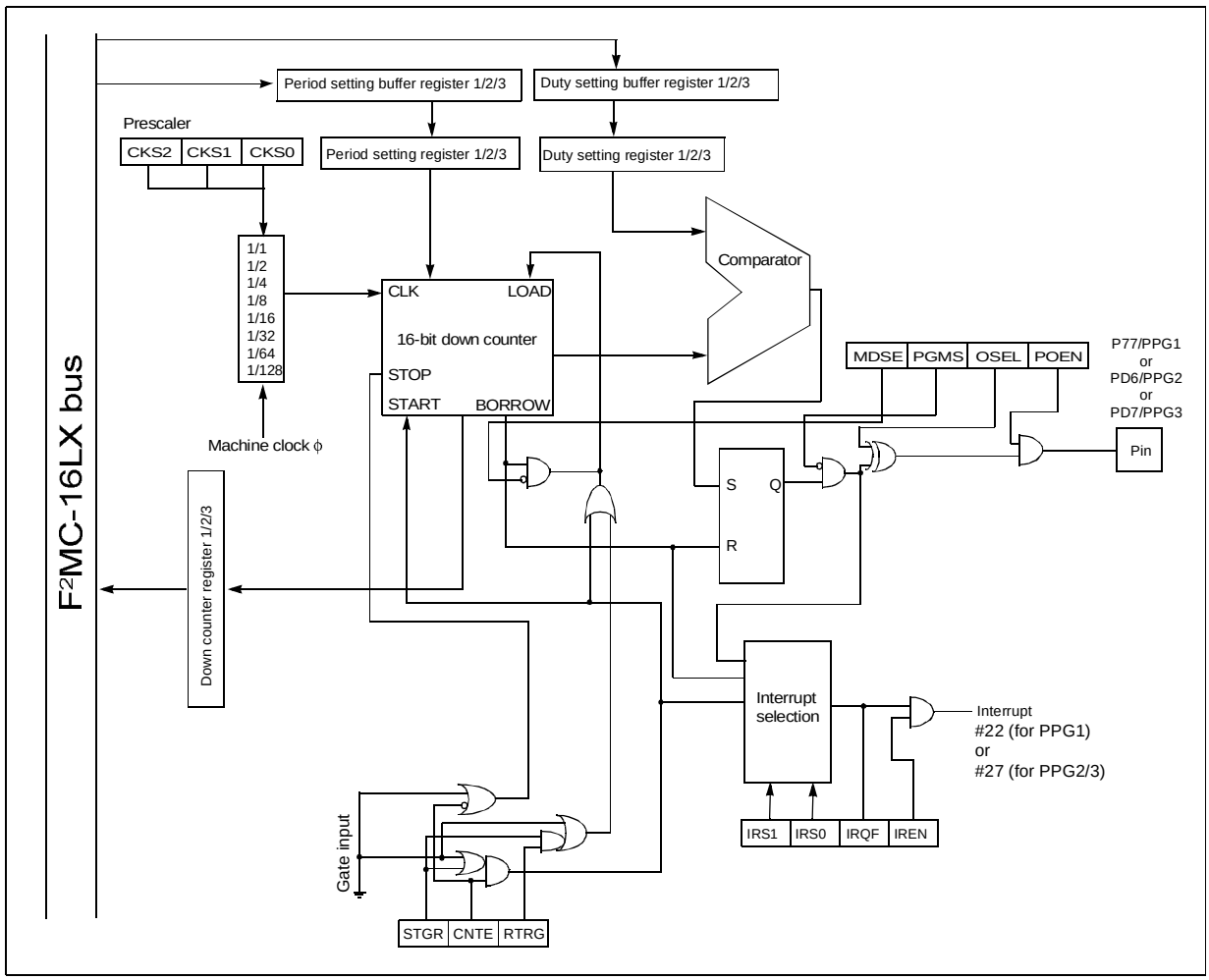


PPG Control Status Register (Lower)



Note : Registers PDCR1 ~ 3, PCSR1 ~ 3 and PDUT1 ~ 3 are word access only

(2) Block diagram of PPG timer



7. 16-bit reload timer (x 4)

The 16-bit reload timer provides two operating mode, internal clock mode and event count mode. In each operating mode, the 16-bit down counter can be reloaded (reload mode) or stopped when underflow (one-shot mode).
Output pins TO1 ~ TO4 are able to output different waveform according to the counter operating mode. TO1 ~ TO4 toggles when counter underflow if counter is operated as reload mode. TO1 ~ TO4 output specified level ("H" or "L") when counter is counting if the counter is in one-shot mode.

Features of the 16-bit reload timer :

- Interrupt generated when timer underflow
- EI²OS supported
- Internal clock operating mode :
  Three internal count clocks can be selected
  Counter can be activated by software or external trigger (signal at TIN1 ~ TIN4 pin)
  Counter can be reloaded or stopped when underflow after activated
- Event count operating mode :
  Counter counts down by one when specified edge at TIN1 ~ TIN4 pin
  Counter can be reloaded or stopped when underflow

(1) Register configuration of reload timer

Timer Control Status Register (Upper)

Address: ch1 000071H / ch2 000075H / ch3 000079H / ch4 00007DH	15	14	13	12	11	10	9	8	⇦ Bit number
	—	—	—	—	CSL1	CSL0	MOD2	MOD1	TMCSRH1 ~ 4
Read/write ⇨	—	—	—	—	R/W	R/W	R/W	R/W	
Initial value ⇨	—	—	—	—	0	0	0	0	

Timer Control Status Register (Lower)

Address: ch1 000070H / ch2 000074H / ch3 000078H / ch4 00007CH	7	6	5	4	3	2	1	0	⇦ Bit number
	MOD0	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG	TMCSRL1 ~ 4
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

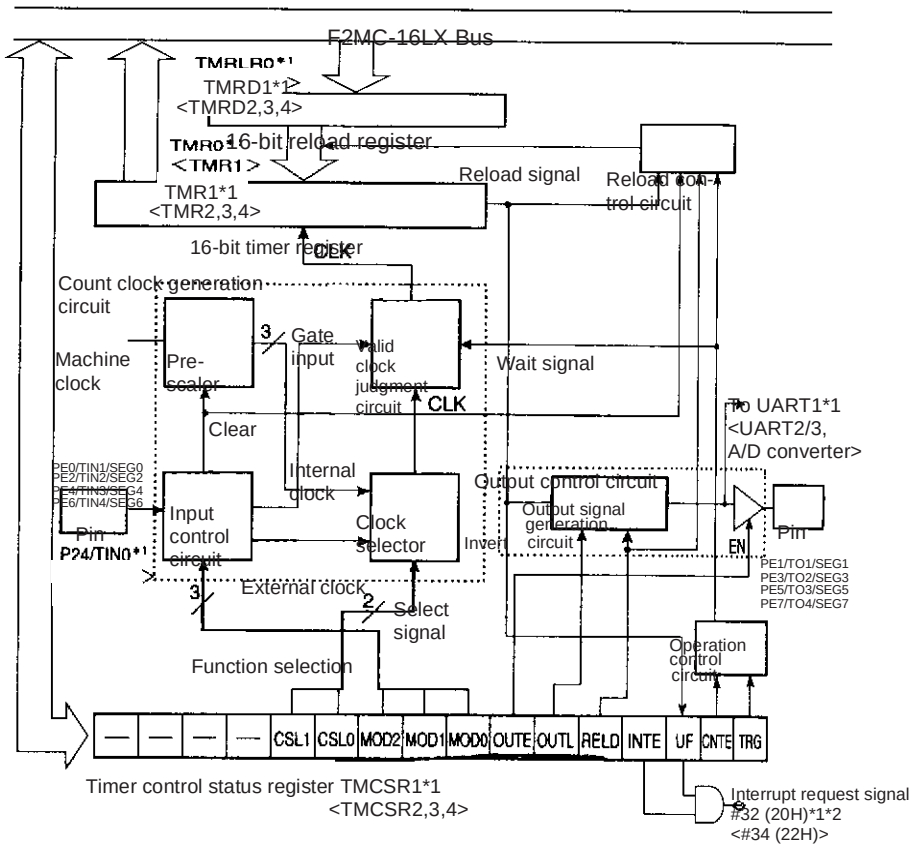
16-bit Timer Register / 16-bit Reload Register (Upper)

Address: ch1 000073H / ch2 000077H / ch3 00007BH / ch4 00007FH	15	14	13	12	11	10	9	8	⇦ Bit number
	D15	D14	D13	D12	D11	D10	D09	D08	TMR1 ~ 4 / TMRD1 ~ 4
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	X	X	X	X	X	X	X	X	

16-bit Timer Register / 16-bit Reload Register (Lower)

Address: ch1 000072H / ch2 000076H / ch3 00007AH / ch4 00007EH	7	6	5	4	3	2	1	0	⇦ Bit number
	D07	D06	D05	D04	D03	D02	D01	D00	TMR1 ~ 4 / TMRD1 ~ 4
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	X	X	X	X	X	X	X	X	

(2) Block diagram of reload timer



*1 This register includes channel 1,2,3 and 4. The register enclosed in < and > indicates the channel 2,3 and 4 register.

*2 Interrupt number, channel 1 and 2 share one interrupt number, channel 3 and 4 share another

8. I²C

The I²C (Inter IC Bus) interface is a simple structure bidirectional bus consisting of two wires : a serial data line (SDA) and a serial clock line (SCL). Among the devices connected with these two wires, information is transmitted to one another. By recognizing the unique address of each device, it can operate as a transmitting or receiving device in accordance with the function of each device. Among these devices, the master/slave relation is established.

The I²C interface can connect two or more devices to the bus provided the upper limit of the bus capacitance does not exceed 400 pF. It is a full-fledged multi-master bus equipped with collision detection and communication adjustment procedures designed to avoid the destruction of data if two or more masters attempt to start data transfer simultaneously.

The communication adjustment procedure permits only one master to control the bus when two or more masters attempt to control the bus so that messages are not lost or the contents of messages are not changed. Multi-master means that multiple masters attempt to control the bus simultaneously without losing messages.

This I²C interface includes MCU standby mode wake-up function, and a CRC-8 calculator that performs automatic Packet Error Code (PEC) generation and verification.

(1) Register configuration of I²C

I²C Bus Control Register (Lower)

	7	6	5	4	3	2	1	0	
Address: 000080H	—	—	—	—	RES	PECE	LBT	WUE	IBCRL
Read/write	-	-	-	-	R/W	R/W	R/W	R/W	
Initial value	-	-	-	-	0	0	0	0	

I²C Bus Control Register (Upper)

	15	14	13	12	11	10	9	8	
Address: 000081H	BER	BEIE	SCC	MSS	ACK	GCAA	INTE	INT	IBCRH
Read/write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

I²C Bus Status Register (Lower)

	7	6	5	4	3	2	1	0	
Address: 000082H	BB	RSC	AL	LRB	TRX	AAS	GCA	FBT	IBSRL
Read/write	R	R	R	R	R	R	R	R	
Initial value	0	0	0	0	0	0	0	0	

I²C Bus Status Register (Upper)

	15	14	13	12	11	10	9	8	
Address: 000083H	—	—	PMATCH	WUF	TDR	TCR	MTR	STR	IBSRH
Read/write	-	-	R	R/W	R/W	R/W	R/W	R/W	
Initial value	-	-	0	0	0	0	0	0	

I²C Data Register

	7	6	5	4	3	2	1	0	
Address: 000084H	D7	D6	D5	D4	D3	D2	D1	D0	IDAR
Read/write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	X	X	X	X	X	X	X	X	

(Continued)

(Continued)

I²C Address Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 000085H	—	A6	A5	A4	A3	A2	A1	A0	IADR
Read/write ⇨	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	-	X	X	X	X	X	X	X	

I²C Clock Control Register

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 000086H	DMBP	—	EN	CS4	CS3	CS2	CS1	CS0	ICCR
Read/write ⇨	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	-	0	0	0	0	0	0	

I²C Timeout Control Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 000087H	—	AAC	—	TOE	EXT	TS2	TS1	TS0	ITCR
Read/write ⇨	-	R/W	-	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	-	0	-	0	0	0	0	0	

I²C Timeout Clock Register

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 000088H	C7	C6	C5	C4	C3	C2	C1	C0	ITOC
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

I²C Timeout Data Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 000089H	D7	D6	D5	D4	D3	D2	D1	D0	ITOD
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

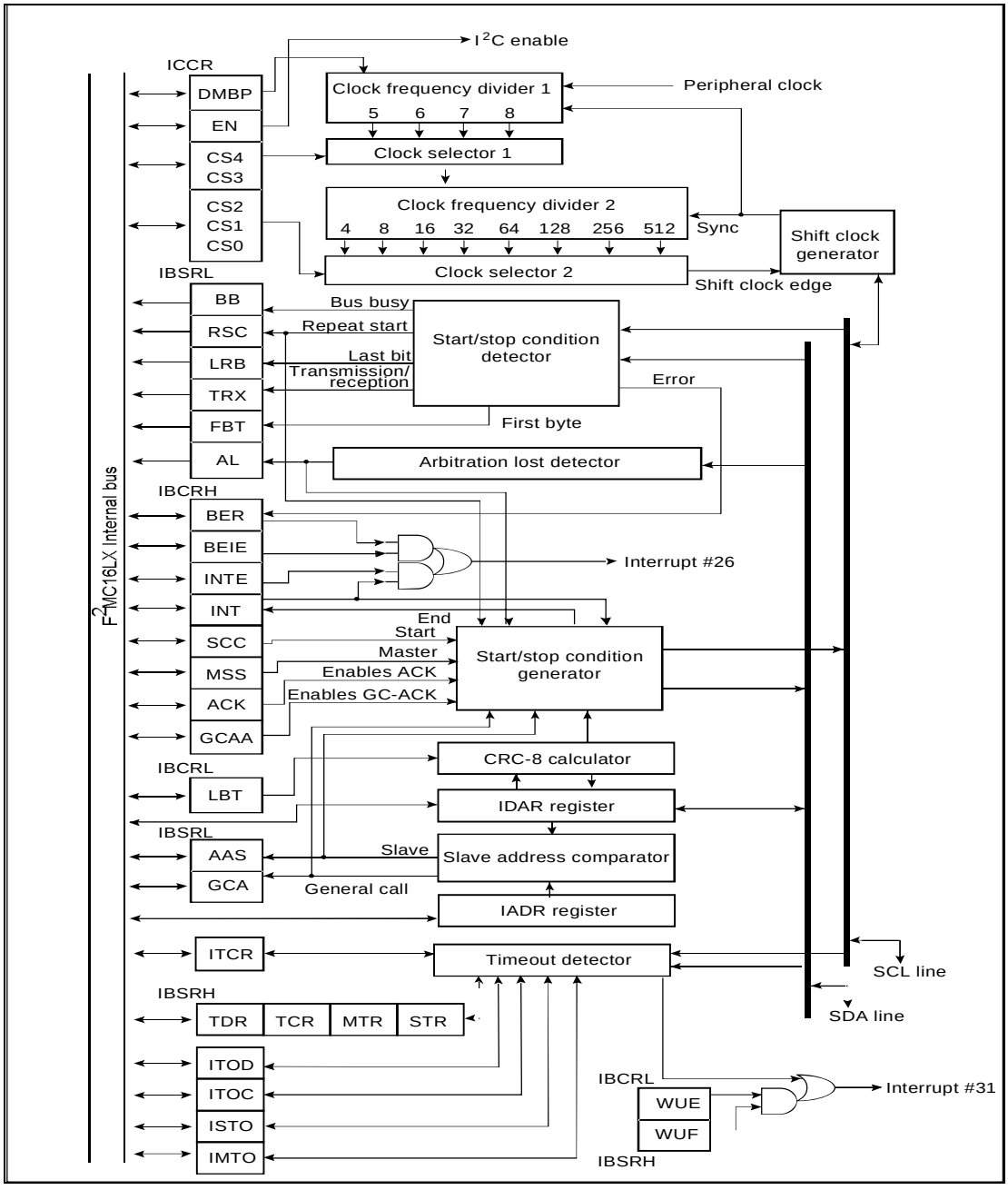
I²C Slave Timeout Register

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 00008AH	S6	S6	S5	S4	S3	S2	S1	S0	ISTO
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

I²C Master Timeout Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 00008BH	M7	M6	M5	M4	M3	M2	M1	M0	IMTO
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

(2) Block diagram of I²C



9. MI²C

The Multi-address I²C (Inter IC Bus) interface is a simple structure bidirectional bus consisting of two wires : a serial data line (SDA) and a serial clock line (SCL). Among the devices connected with these two wires, information is transmitted to one another. By recognizing the unique address of each device, it can operate as a transmitting or receiving device in accordance with the function of each device. Among these devices, the master/slave relation is established.

The Multi-address I²C interface can connect two or more devices to the bus provided the upper limit of the bus capacitance does not exceed 400 pF. It is a full-fledged multi-master bus equipped with collision detection and communication adjustment procedures designed to avoid the destruction of data if two or more masters attempt to start data transfer simultaneously. This macro provides 6 addresses to implement the multi-address function.

The communication adjustment procedure permits only one master to control the bus when two or more masters attempt to control the bus so that messages are not lost or the contents of messages are not changed. Multi-master means that multiple masters attempt to control the bus simultaneously without losing messages.

This Multi-address I²C interface includes MCU standby mode wake-up function, and a CRC-8 calculator that performs automatic Packet Error Code (PEC) generation and verification.

(1) Register configuration of MI²C

Multi-address I²C Bus Control Register (Lower)

	7	6	5	4	3	2	1	0	
									⇦ Bit number
Address: 0000C0H	—	—	—	—	RES	PECE	LBT	WUE	MBCRL
Read/write ⇨	-	-	-	-	R/W	R/W	R/W	R/W	
Initial value ⇨	-	-	-	-	0	0	0	0	

Multi-address I²C Bus Control Register (Upper)

	15	14	13	12	11	10	9	8	
									⇦ Bit number
Address: 0000C1H	BER	BEIE	SCC	MSS	ACK	GCAA	INTE	INT	MBCRH
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

Multi-address I²C Bus Status Register (Lower)

	7	6	5	4	3	2	1	0	
									⇦ Bit number
Address: 0000C2H	BB	RSC	AL	LRB	TRX	AAS	GCA	FBT	MBSRL
Read/write ⇨	R	R	R	R	R	R	R	R	
Initial value ⇨	0	0	0	0	0	0	0	0	

Multi-address I²C Bus Status Register (Upper)

	15	14	13	12	11	10	9	8	
									⇦ Bit number
Address: 0000C3H	—	—	PMATCH	WUF	TDR	TCR	MTR	STR	MBSRH
Read/write ⇨	-	-	R	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	-	-	0	0	0	0	0	0	

Multi-address I²C Data Register

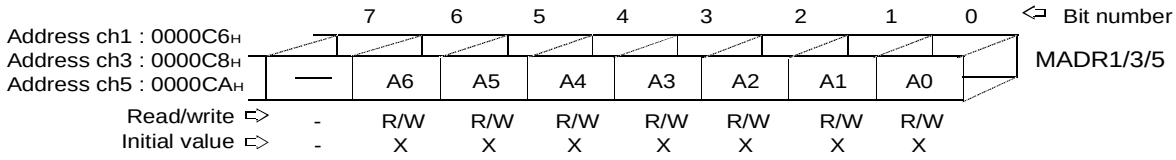
	7	6	5	4	3	2	1	0	
									⇦ Bit number
Address: 0000C4H	D7	D6	D5	D4	D3	D2	D1	D0	MDAR
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	X	X	X	X	X	X	X	X	

(Continued)

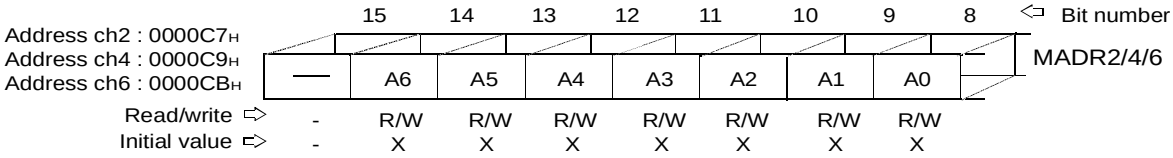
(Continued)

Multi-address I²C Alert Register

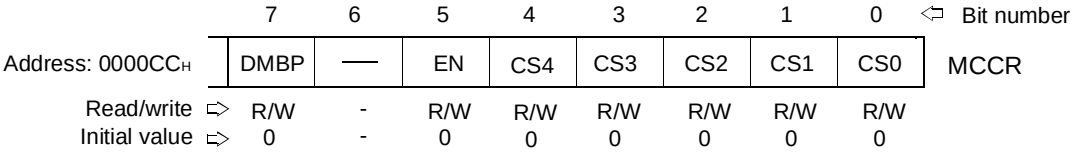
Multi-address I²C Address Register 1/3/5



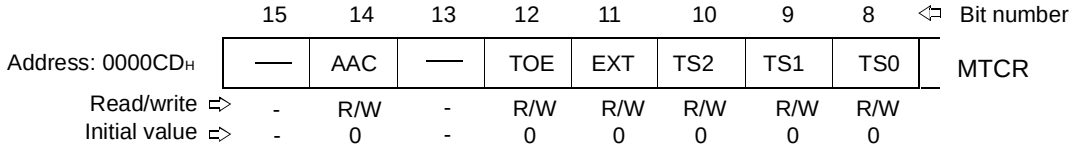
Multi-address I²C Address Register 2/4/6



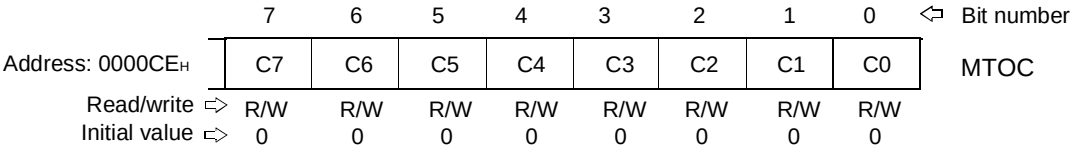
Multi-address I²C Clock Control Register



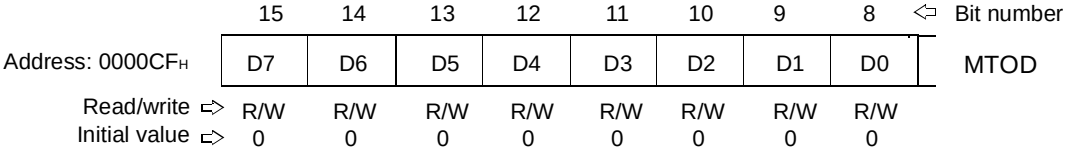
Multi-address I²C Timeout Control Register



Multi-address I²C Timeout Clock Register



Multi-address I²C Timeout Data Register



(Continued)

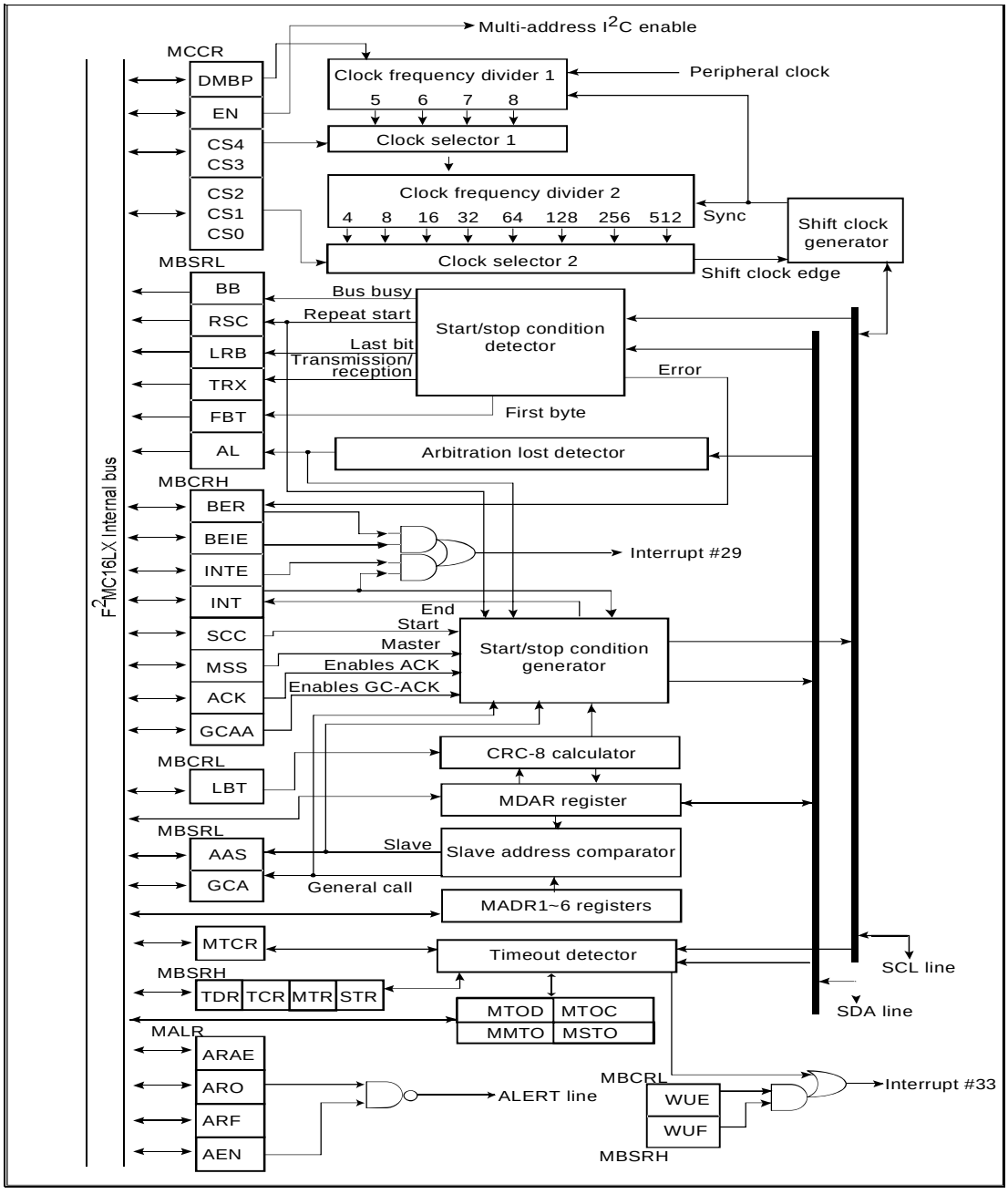
Multi-address I²C Slave Timeout Register

	7	6	5	4	3	2	1	0	
	⇦ Bit number								
Address: 0000D0H	S6	S6	S5	S4	S3	S2	S1	S0	MSTO
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

Multi-address I²C Master Timeout Register

	15	14	13	12	11	10	9	8	
	⇦ Bit number								
Address: 0000D1H	M7	M6	M5	M4	M3	M2	M1	M0	MMTO
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

(2) Block diagram of MI²C



10. Bridge circuit

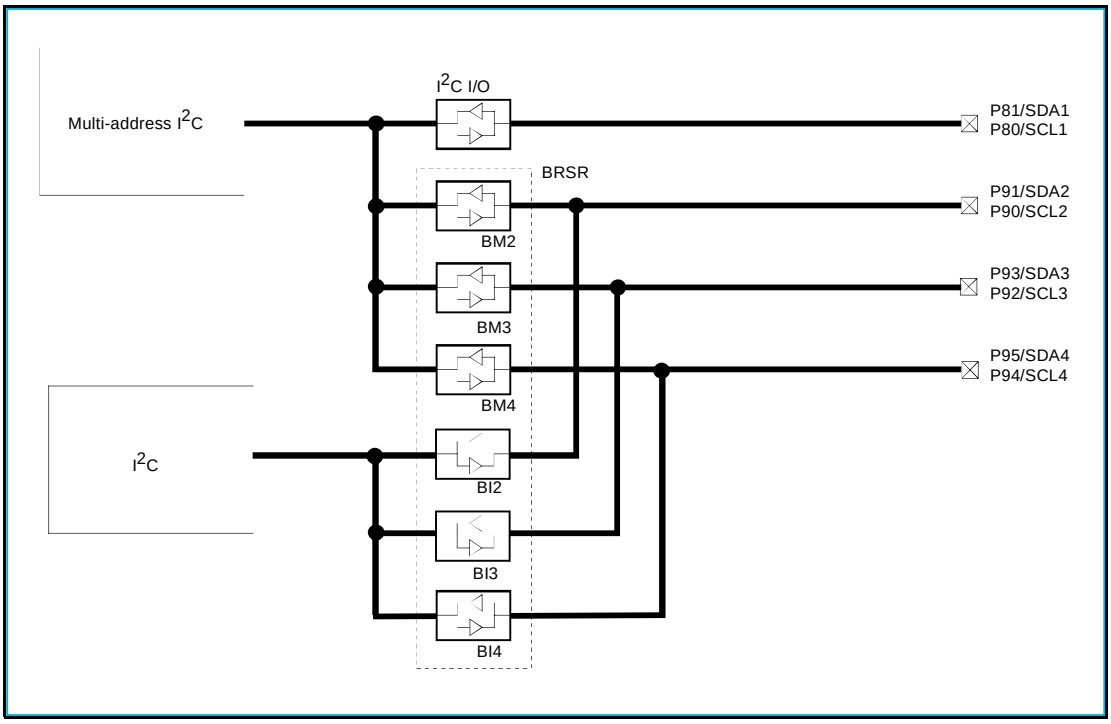
The bridge circuit can switch the I/O path of each port to I²C or Multi-address I²C.

(1) Register configuration of bridge circuit

Bridge Circuit Selection Register

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 00002C$_H$	—	—	BM4	BI4	BM3	BI3	BM2	BI2	BRSR
Read/write ⇨	—	—	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	—	—	0	0	0	0	0	0	

(2) Block diagram of bridge circuit



11. Comparator

This comparator circuit monitors voltage of up to three batteries and automatically controls electric discharge. Either parallel discharge or sequential discharge can be selected.

• Parallel discharge control

In parallel discharge control, all batteries are allowed to discharge when power is not being supplied from the AC adapter.

• If power is being supplied from the AC adapter, the permission/prohibition of discharge for batteries is controlled by software.

• Sequential discharge control

In sequential discharge control, the comparator controls discharge in a specified order, while monitoring intermittent interruption of power, voltage level, and mount/dismount of batteries, when power is not being supplied from the AC adapter.

• If power is being supplied from the AC adapter, the permission/prohibition of discharge for batteries is controlled by software.
• Up to three batteries can be controlled, and the order of discharge can be selected.

- The affect of intermittent interruption of power is automatically filtered.
- Mount/dismount of batteries is automatically detected and discharge is controlled.
- Battery voltage is monitored, and if battery voltage is below the specified voltage, change over to the next battery is automatically done.

(1) Register configuration of comparator

Comparator Control Register (Lower)

	7	6	5	4	3	2	1	0	
									⇦ Bit number
Address: 0000D8H	—	—	BOF3	BOF2	BOF1	SPM2	SPM1	SPM0	COCRL
Read/write ⇨	-	-	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	-	-	0	0	0	0	0	0	

Comparator Control Register (Upper)

	15	14	13	12	11	10	9	8	
									⇦ Bit number
Address: 0000D9H	SPL3	SPL2	SPL1	B3	B2	B1	DC2	DC1	COCRH
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	1	1	1	1	1	

Comparator Status Register 1 (Lower)

	7	6	5	4	3	2	1	0	
									⇦ Bit number
Address: 0000DAH	COR8	COR7	COR6	COR5	COR4	COR3	COR2	COR1	COSRL1
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

Comparator Status Register 1 (Upper)

	15	14	13	12	11	10	9	8	
									⇦ Bit number
Address: 0000DBH	—	—	SWR3	SWR2	SW1	VAR3	VAR2	VAR1	COSRH1
Read/write ⇨	-	-	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	-	-	0	0	0	0	0	0	

Comparator Interrupt Control Register (Lower)

	7	6	5	4	3	2	1	0	
									⇦ Bit number
Address: 0000DCH	CEN8	CEN7	CEN6	CEN5	CEN4	CEN3	CEN2	CEN1	CICRL
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

(Continued)

(Continued)

Comparator Interrupt Control Register (Upper)

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 0000DD$_H$	—	—	SEN3	SEN2	SEN1	VEN3	VEN2	VEN1	CICRH
Read/write ⇨	-	-	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	-	-	0	0	0	0	0	0	

Comparator Status Register 2 (Lower)

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 0000DE$_H$	COS8	COS7	COS6	COS5	COS4	COS3	COS2	COS1	COSRL2
Read/write ⇨	R	R	R	R	R	R	R	R	
Initial value ⇨	X	X	X	X	X	X	X	X	

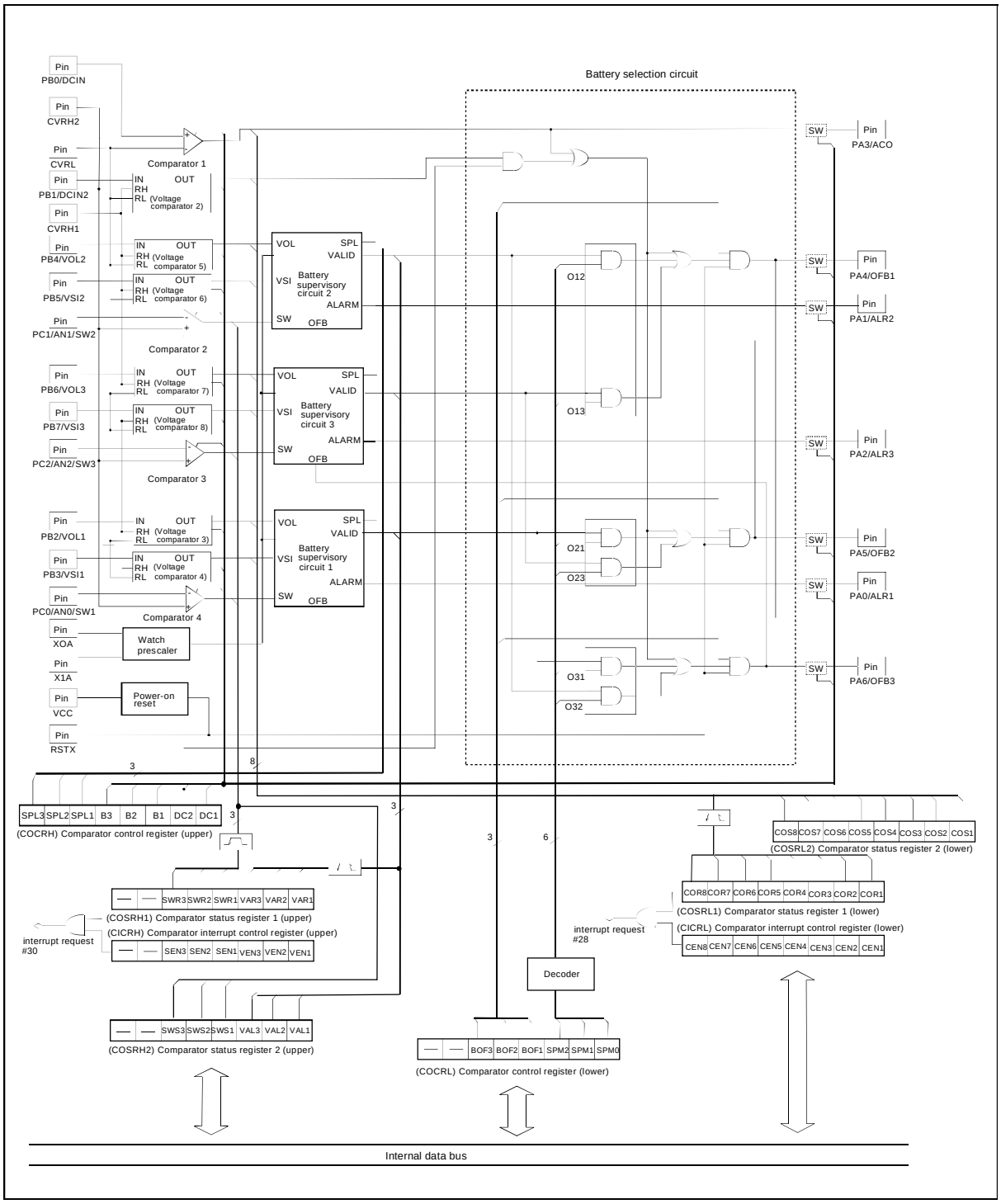
Comparator Status Register 2 (Upper)

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 0000DF$_H$	—	—	SWS3	SWS2	SWS1	VAL3	VAL2	VAL1	COSRH2
Read/write ⇨	-	-	R	R	R	R	R	R	
Initial value ⇨	-	-	X	X	X	X	X	X	

Comparator Input Enable Register

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 0000E0$_H$	—	—	—	BIE3	BIE2	BIE1	DIE2	DIE1	CIER
Read/write ⇨	-	-	-	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	-	-	-	1	1	1	1	1	

(2) Block diagram of comparator



12. UART (x 3)

The UART (Universal Asychronous Receiver Transmitter) is a serial I/O port for asynchronous (start-stop) communication or clock-synchronous communication.

The UART has the following features :

- Full-duplex double buffering
- Capable of asynchronous (start-stop bit) and CLK-synchronous communications
- Support for the multiprocessor mode
- Various method of baud rate generation :
  - External clock input possible
  - Internal clock (a clock supplied from 16-bit reload timer can be used)
  - Embedded dedicated baud rate generator

Operation	Baud rate
Asynchronous	76923 / 38461 / 19230 / 9615 / 500K / 250K bps
CLK synchronous	16M / 8M / 4M / 2M / 1M / 500K bps

- Error detection functions (parity, framing, overrun)
- NRZ (Non Return to Zero) signal format
- Interrupt request :
  - Receive interrupt (receive complete, receive error detection)
  - Transmit interrupt (transmission complete)
  - Transmit / receive conforms to extended intelligent I/O service (EI^{2}OS)

(1) Register configuration of UART

Serial Mode Register

Address: ch1 000020H
ch2 0000D2H
ch3 0000E4H

Bit number	7	6	5	4	3	2	1	0
	MD1	MD0	CS2	CS1	CS0	—	SCKE	SOE
Read/write	R/W	R/W	R/W	R/W	R/W	—	R/W	R/W
Initial value	0	0	0	0	0	—	0	0

SMR1/2/3

Serial Control Register

Address: ch1 000021H
ch2 0000D3H
ch3 0000E5H

Bit number	15	14	13	12	11	10	9	8
	PEN	P	SBL	CL	A/D	REC	RXE	TXE
Read/write	R/W	R/W	R/W	R/W	R/W	W	R/W	R/W
Initial value	0	0	0	0	0	1	0	0

SCR1/2/3

UART Input Data Register / Output Data Register

Address: ch1 000022H
ch2 0000D4H
ch3 0000E6H

Bit number	7	6	5	4	3	2	1	0
	D7	D6	D5	D4	D3	D2	D1	D0
Read/write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	X	X	X	X	X	X	X	X

SIDR1/2/3
SODR1/2/3

UART Status Register

Address: ch1 000023H
ch2 0000D5H
ch3 0000E7H

Bit number	15	14	13	12	11	10	9	8
	PE	ORE	FRE	RDRF	TDRE	BDS	RIE	TIE
Read/write	R	R	R	R	R	R/W	R/W	R/W
Initial value	0	0	0	0	1	0	0	0

SSR1/2/3

Clock Division Control Register

Address: ch1 000025H
ch2 0000D7H
ch3 0000E9H

Bit number	15	14	13	12	11	10	9	8
	MD	SRST	—	—	DIV3	DIV2	DIV1	DIV0
Read/write	R/W	R/W	—	—	R/W	R/W	R/W	R/W
Initial value	0	0	—	—	0	0	0	0

CDCR1/2/3

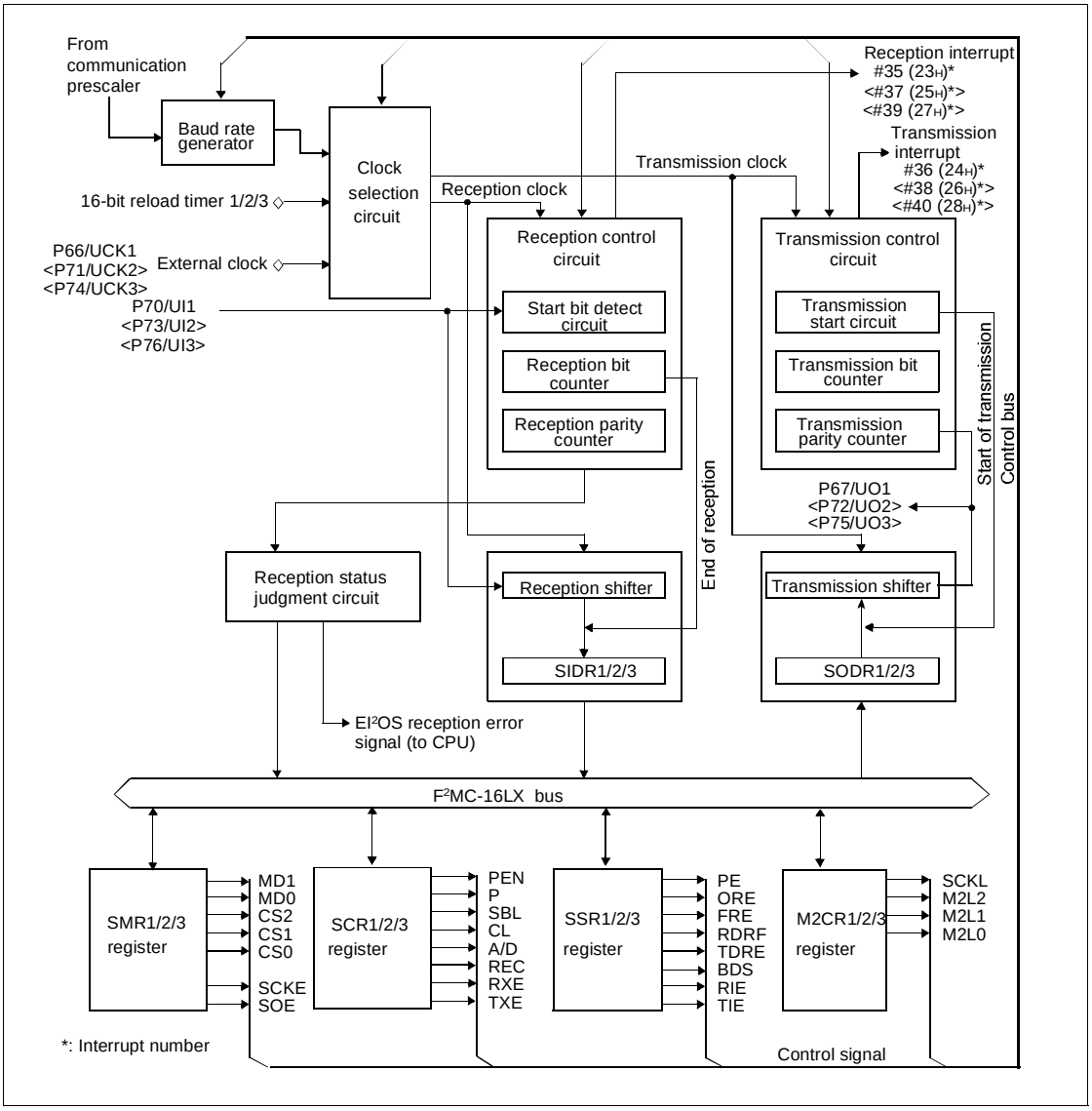
Mode 2 Control Register

Address: ch1 000024H
ch2 0000D6H
ch3 0000E8H

Bit number	7	6	5	4	3	2	1	0
	—	—	—	—	SCKL	M2L2	M2L1	M2L0
Read/write	—	—	—	—	R/W	R/W	R/W	R/W
Initial value	—	—	—	—	1	0	0	0

M2CR1/2/3

(2) Block diagram of UART



13. LCD controller/driver

The LCD (Liquid Crystal Display) controller/driver function displays the contents of a display data memory directly to the LCD panel by segment and common outputs.

- Up to nine segment outputs (SEG0 to SEG8) and four common outputs (COM0 to COM3) may be used.
- Built-in display RAM.
- Three selectable duty ratios (1/2, 1/3, and 1/4). Not all duty ratios are available with all bias settings, however.
- Either the main or sub-clock can be selected as the drive clock.
- LCD can be driven directly.

Table below shows the duty ratios available with each bias setting.

Part number	Bias	1/2 duty ratio	1/3 duty ratio	1/4 duty ratio
MB90370 series	1/2 bias	○	X	X
	1/3 bias	X	○	○

○: Recommended mode

X : Do not use

(1) Register configuration of LCD

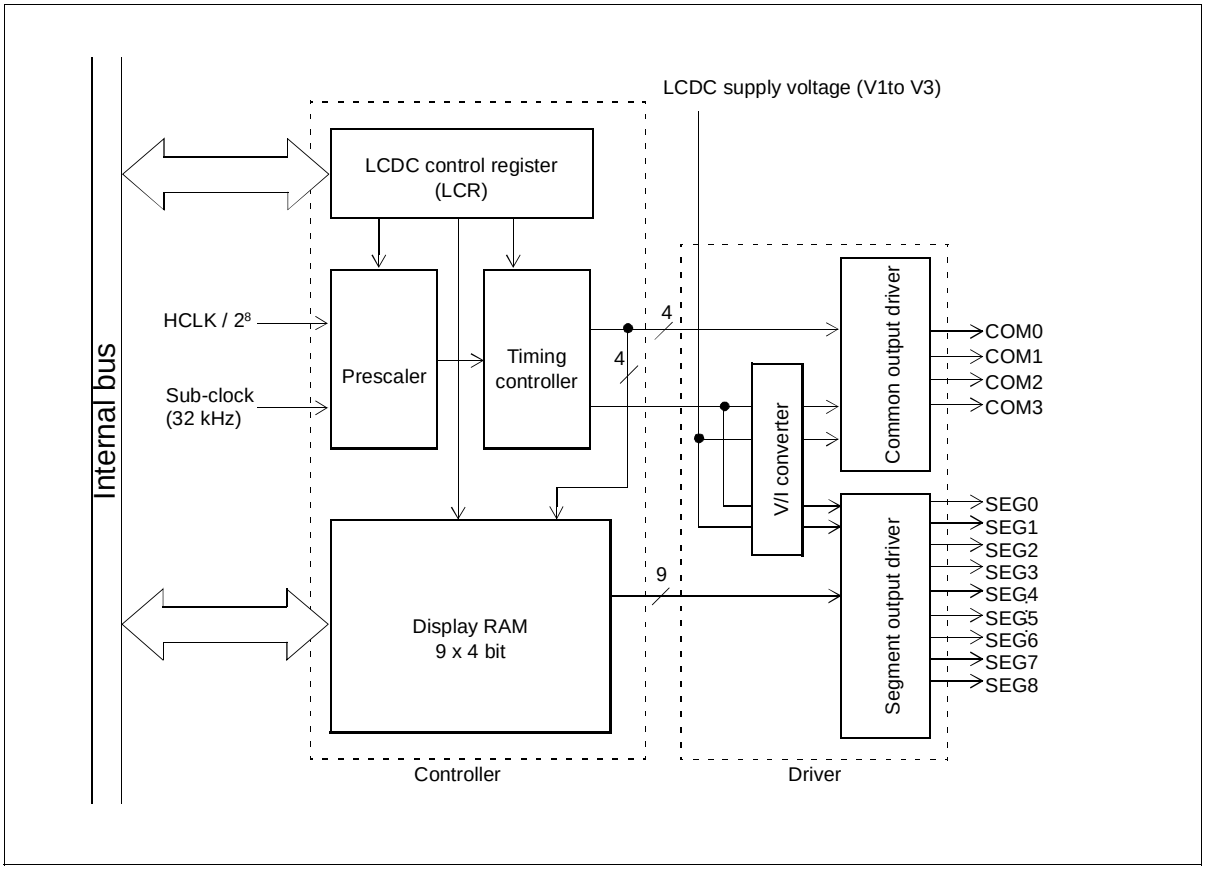
LCDC Control Register (Upper)

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 0000EF$_H$	SS4	VS	CS1	CS0	SS3	SS2	SS1	SS0	LCRH
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

LCDC Control Register (Lower)

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 0000EE$_H$	CSS	LCEN	VSEL	BK	MS1	MS0	FP1	FP0	LCRL
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	1	0	0	0	0	

(2) Block diagram of LCD



14. A/D converter

The A/D (Analog to Digital) converter converts the analog voltage input to an analog input pin (input voltage) to a digital value.

The converter has the following features :

- The minimum conversion time is 6.13 µs (for a machine clock of 16 MHz; includes the sampling time).
- The minimum sampling time is 3.75 µs (for a machine clock of 16 MHz).
- The converter uses the RC-type successive approximation conversion method with a sample and hold circuit.
- A resolution of 10 bits or 8 bits can be selected.
- Up to twelve channels for analog input pins can be selected by a program.
- Various conversion mode :
  - Single conversion mode : Selectively convert one channel.
  - Scan conversion mode : Continuously convert multiple channels. Maximum of 12 selectable channels.
  - Continuous conversion mode : Repeatedly convert specified channels.
  - Stop conversion mode : Convert one channel then halt until the next activation. (Enables synchronization of the conversion start timing.)
- At the end of A/D conversion, an interrupt request can be generated and EI²OS can be activated.
- In the interrupt-enabled state, the conversion data protection function prevents any part of the data from being lost through continuous conversion.
- The conversion can be activated by software, 16-bit reload timer 4 (rise edge) and ADTG.

(1) Register configuration of A/D converter

Analog Input Enable Register 2

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 00002B$_H$	—	—	—	—	ADE11	ADE10	ADE9	ADE8	ADER2
Read/write ⇨	—	—	—	—	R/W	R/W	R/W	R/W	
Initial value ⇨	—	—	—	—	1	1	1	1	

Analog Input Enable Register 1

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 00002A$_H$	ADE7	ADE6	ADE5	ADE4	ADE3	ADE2	ADE1	ADE0	ADER1
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	1	1	1	1	1	1	1	1	

A/D Control Status Register 1

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 000031$_H$	BUSY	INT	INTE	PAUS	STS1	STS0	STRT	RESV	ADCS1
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

A./D Control Status Register 0

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 000030$_H$	MD1	MD0	—	—	—	—	—	—	ADCS0
Read/write ⇨	R/W	R/W	—	—	—	—	—	—	
Initial value ⇨	0	0	—	—	—	—	—	—	

(Continued)

(Continued)

A/D Control Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 00002D$_H$	ANS3	ANS2	ANS1	ANS0	ANE3	ANE2	ANE1	ANE0	ADC0
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

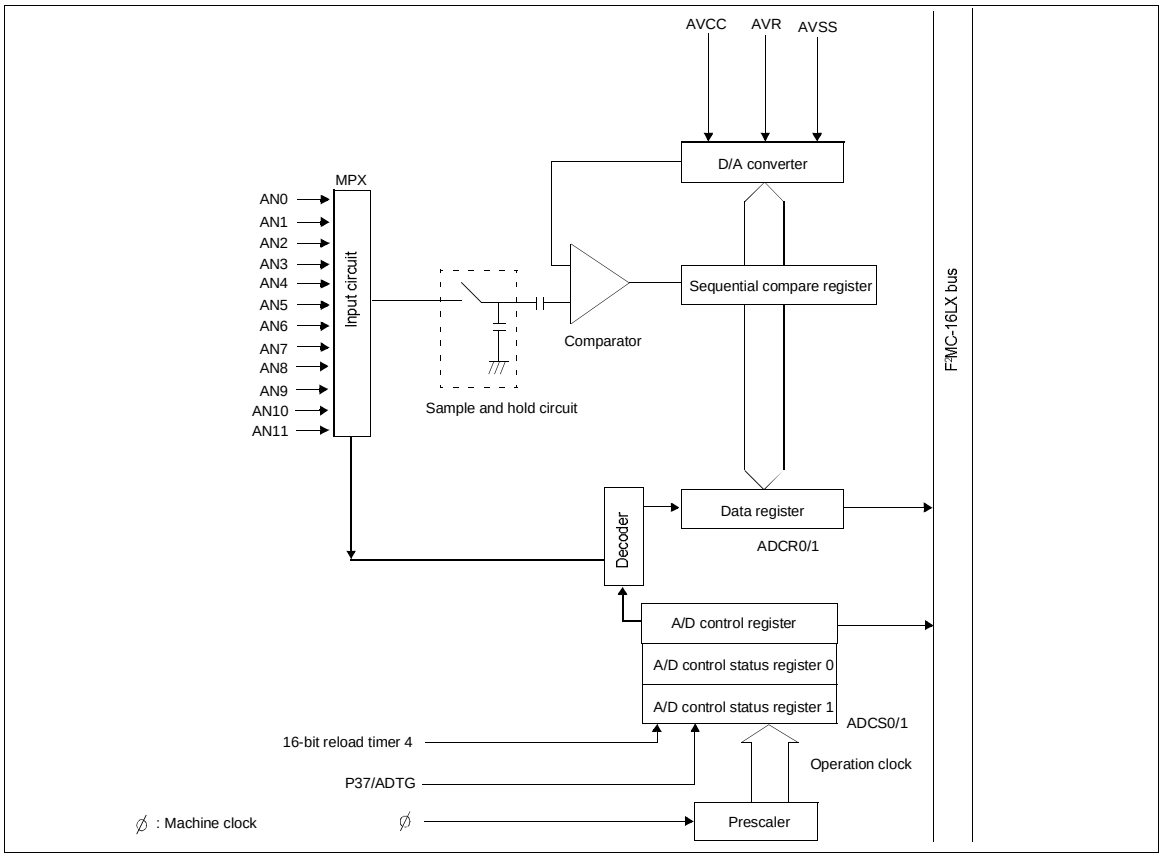
A/D Data Register (Upper)

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 00002F$_H$	S10	ST1	ST0	CT1	CT0	—	D9	D8	ADCR1
Read/write ⇨	R/W	W	W	W	W	—	R	R	
Initial value ⇨	0	0	0	0	0	—	X	X	

A/D Data Register (Lower)

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 00002E$_H$	D7	D6	D5	D4	D3	D2	D1	D0	ADCR0
Read/write ⇨	R	R	R	R	R	R	R	R	
Initial value ⇨	X	X	X	X	X	X	X	X	

(2) Block diagram of A/D converter



15. D/A converter

The D/A (Digital to Analog) converter is used to generate an analog output from an 8-bit digital input. By setting the enable bit in the D/A control register (DACR) to 1, it will enable the corresponding D/A output channel. Hence, setting this bit to 0 will disable that channel.

If D/A output is disabled, the analog switch inserted to the output of each D/A converter channel in series is turned off. In the D/A converter, the bit is cleared to 0 and the direct-current path is shut off. The above is also true in the stop mode.

The output voltage of the D/A converter ranges from 0 V to 255/256 x DVR. To change the output voltage range, adjust the DVR voltage externally.

The D/A converter output does not have the internal buffer amplifier. The analog switch (= 100 Ω) is inserted to the output in series. To apply load to the output externally, estimate a sufficient stabilization time.

Table below lists the theoretical values of output voltage of the D/A converter.

Value written to DA07 to DA00 and DA17 to DA10	Theoretical value of output voltage
00_H	0/256 × DVR (= 0 V)
01_H	1/256 × DVR
02_H	2/256 × DVR
:	:
FD_H	253/256 × DVR
FE_H	254/256 × DVR
FF_H	255/256 × DVR

(1) Register configuration of D/A converter

D/A converter register 1

Bit	15	14	13	12	11	10	9	8	
Address:00005BH	DA17	DA16	DA15	DA14	DA13	DA12	DA11	DA10	DAT1
Read/write →	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value →	X	X	X	X	X	X	X	X	

D/A converter register 0

Bit	7	6	5	4	3	2	1	0	
Address:00005AH	DA07	DA06	DA05	DA04	DA03	DA02	DA01	DA00	DAT0
Read/write →	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value →	X	X	X	X	X	X	X	X	

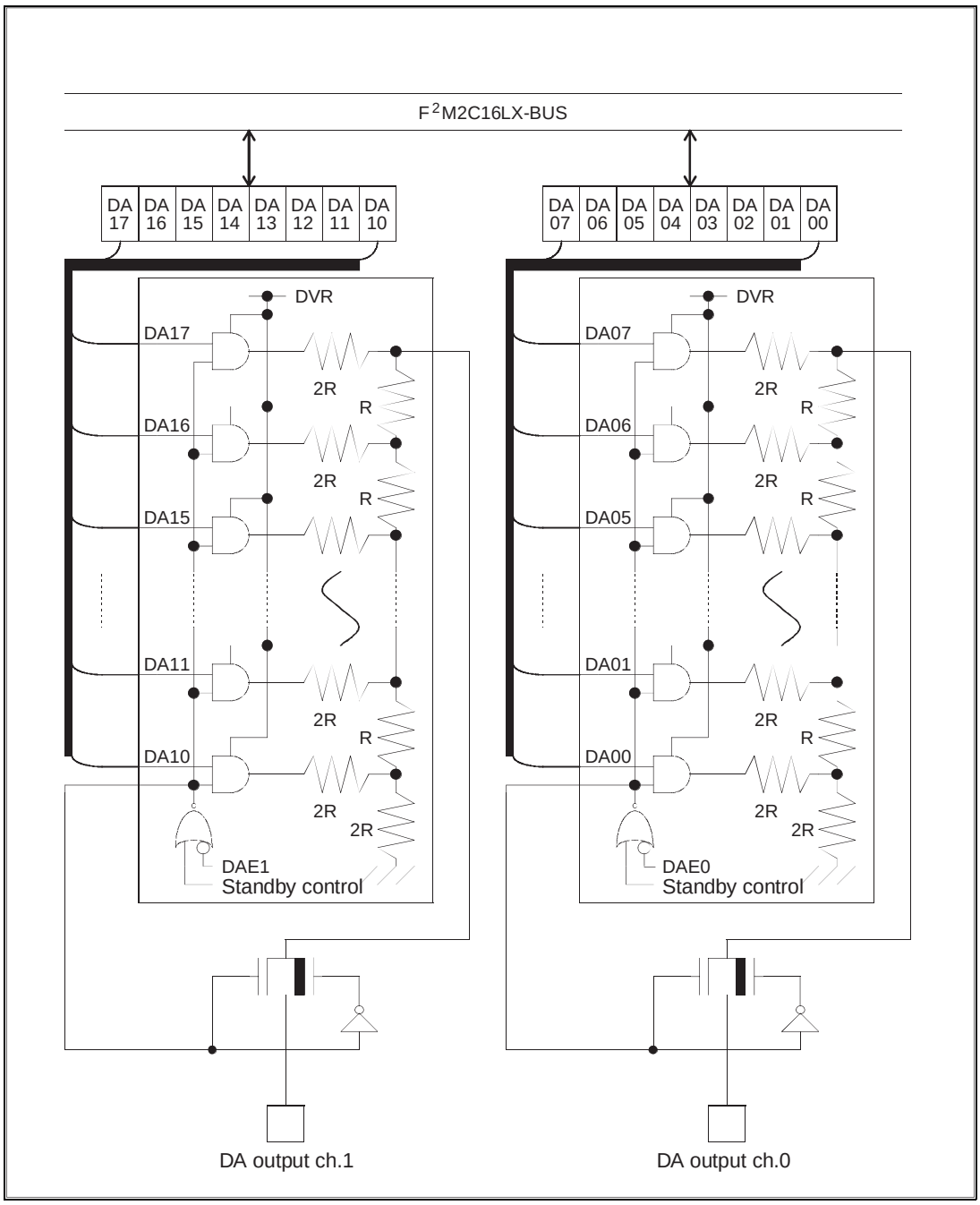
D/A control register 1

Bit	15	14	13	12	11	10	9	8	
Address:00005DH	-	-	-	-	-	-	-	DAE1	DACR1
Read/write →	-	-	-	-	-	-	-	R/W	
Initial value →	-	-	-	-	-	-	-	0	

D/A control register 0

Bit	7	6	5	4	3	2	1	0	
Address:00005CH	-	-	-	-	-	-	-	DAE0	DACR0
Read/write →	-	-	-	-	-	-	-	R/W	
Initial value →	-	-	-	-	-	-	-	0	

(2) Block diagram of D/A converter



16. LPC interface

The LPC (Low Pin Count) interface consists of an LPC bus interface, universal parallel interface (UPI x 4 channels), gate address A20 function and LPC data buffer array. By using the LPC bus interface and UPI, data can be exchanged with an external host CPU synchronously via an external LPC bus.

- LPC bus interface

  The LPC bus interface provides direct access of host CPU to UPI.
  - It supports I/O read and I/O write cycle only. Other cycle types will be ignored.
  - It supports LPC clock running at 33 MHz.

- Universal parallel interface, UPI x 4 channels

  The UPI is used to exchange parallel data to serial data in LPC bus with host CPU.
  - An 8-bit data will be transmitted or received.
  - A buffer function is available for independent input and output.
  - The I/O buffer status can be output externally through LPC bus interface.

- Gate address A20 function for UPI channel 0

  The GA20 (Gate Address A20) is intended to implement the memory management in a PC architecture. This allows the access to the extended memory needed by the operating system. On-chip logic is provided to speed up the generation of GA20.

- Data buffer array

  The data buffer array is consisted of 32 bytes UP data register and 16 bytes DOWN data register to speed up the data transfer between MCU and external host through LPC bus.

(1) Register configuration of LPC bus interface register

LPC Control Register

		7	6	5	4	3	2	1	0	⇦ Bit number
Address:	00006E$_H$	—	—	—	—	—	LRF	LRIE	LPE	LCR
	Read/write ⇨	—	—	—	—	—	R/W	R/W	R/W	
	Initial value ⇨	—	—	—	—	—	0	0	0	

(2) Register configuration of UPI registers

UPI Address Register (Upper)

Address: ch1 00005F$_H$
ch2 000061$_H$
ch3 000063$_H$

	15	14	13	12	11	10	9	8	
	UPA15	UPA14	UPA13	UPA12	UPA11	UPA10	UPA09	UPA08	⇦ Bit number
									UPAH1 ~ 3
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	X	X	X	X	X	X	X	X	

UPI Address Register (Lower)

Address: ch1 00005E$_H$
ch2 000060$_H$
ch3 000062$_H$

	7	6	5	4	3	2	1	0	
	UPA07	UPA06	UPA05	UPA04	UPA03	UPA02	UPA01	UPA00	⇦ Bit number
									UPAL1 ~ 3
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	X	X	X	X	X	X	X	X	

UPI Control Register (Upper)

Address: 000065$_H$

	15	14	13	12	11	10	9	8	
	—	UPE3	IBFE3	OBEE3	—	UPE2	IBFE2	OBEE2	⇦ Bit number
									UPCH
Read/write ⇨	—	R/W	R/W	R/W	—	R/W	R/W	R/W	
Initial value ⇨	—	0	0	0	—	0	0	0	

UPI Control Register (Lower)

Address: 000064$_H$

	7	6	5	4	3	2	1	0	
	DBAE	UPE1	IBFE1	OBEE1	GA20E	UPE0	IBFE0	OBEE0	⇦ Bit number
									UPCL
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

UPI Status Register

Address: ch0 000067$_H$
ch1 000069$_H$
ch2 00006B$_H$
ch3 00006D$_H$

	15	14	13	12	11	10	9	8	
	UF4	UF3	UF2	UF1	A2	UF0	IBF	OBF	⇦ Bit number
									UPS0 ~ 3
Read/write ⇨	R/W	R/W	R/W	R/W	R	R/W	R	R	
Initial value ⇨	0	0	0	0	0	0	0	0	

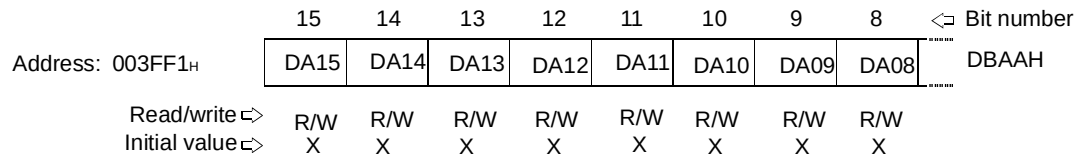
UPI Data Input Register / Data Output Register

Address: ch0 000066$_H$
ch1 000068$_H$
ch2 00006A$_H$
ch3 00006C$_H$

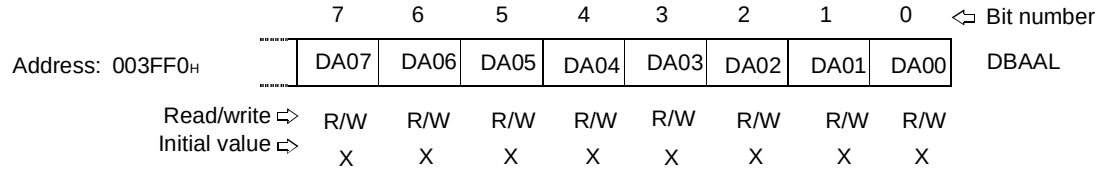
	7	6	5	4	3	2	1	0	
	UPD7	UPD6	UPD5	UPD4	UPD3	UPD2	UPD1	UPD0	⇦ Bit number
									UPDI0 ~ 3 / UPDO0 ~ 3
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	X	X	X	X	X	X	X	X	

(3) Register configuration of LPC data buffer registers

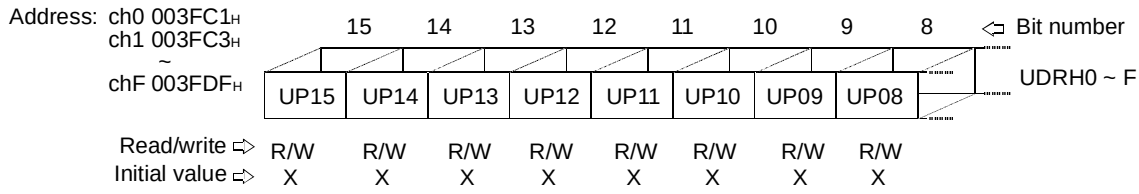
Data Buffer Array Address Register (Upper)



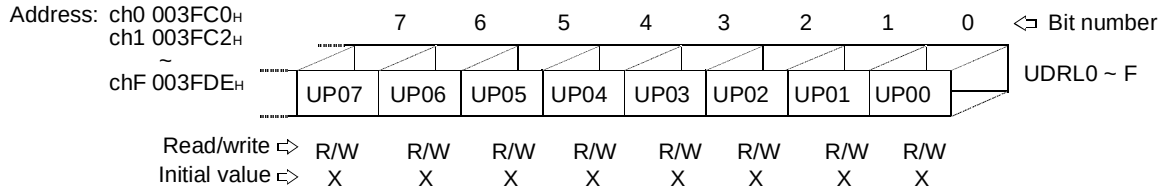
Data Buffer Array Address Register (Lower)



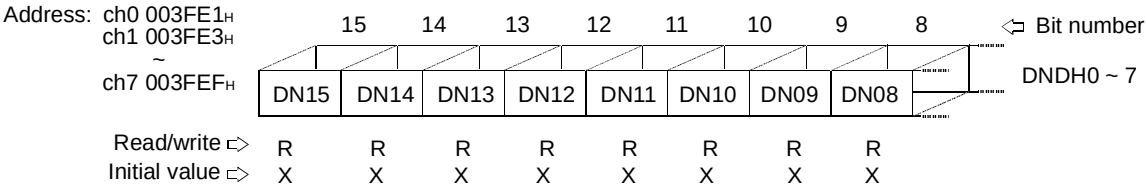
UP Data Register (upper)



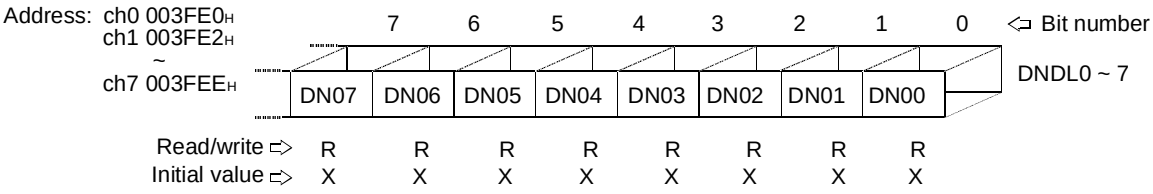
UP Data Register (lower)



DOWN Data Register (upper)



DOWN Data Register (lower)



(Continued)

(Continued)

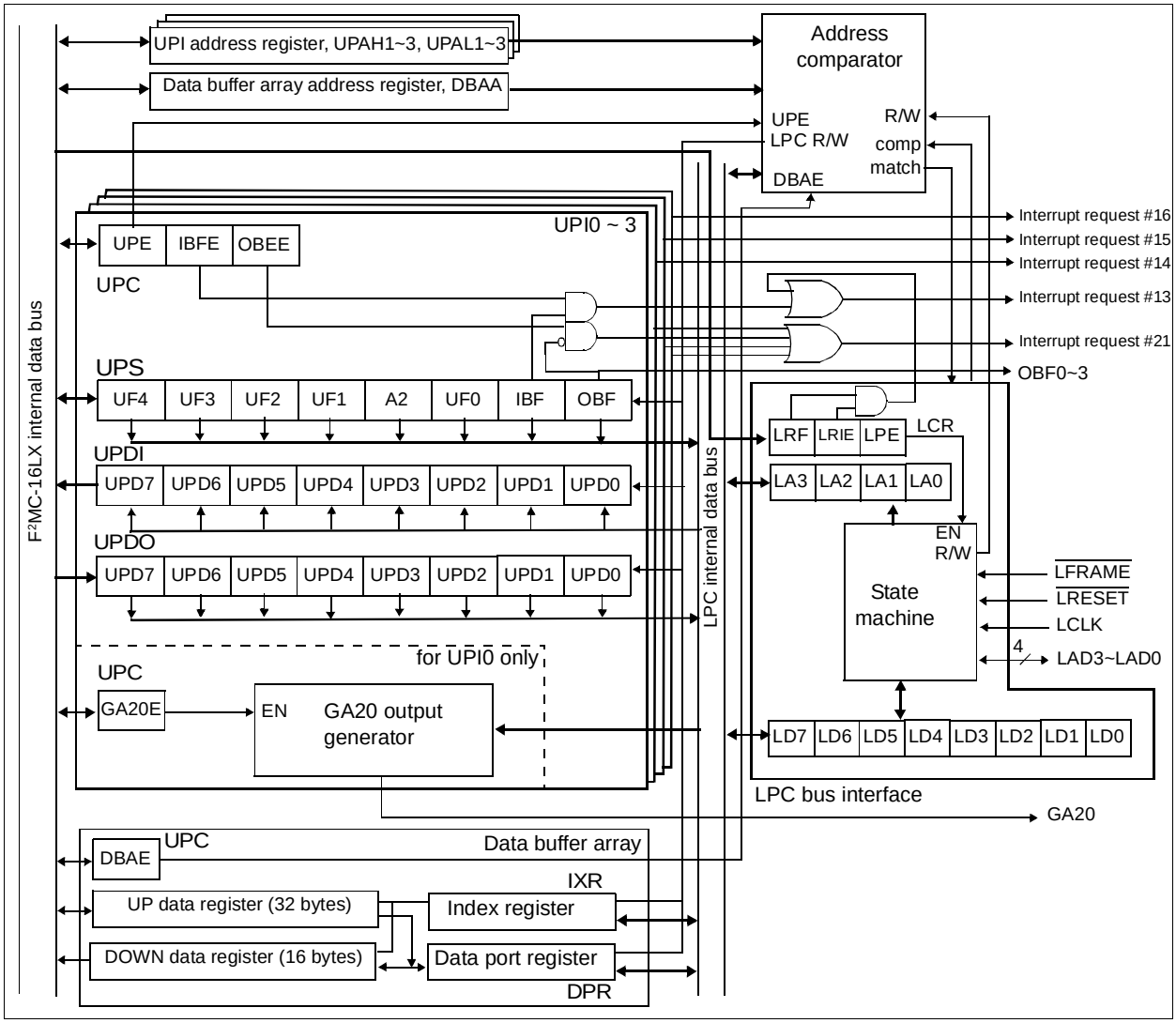
Index Register

	7	6	5	4	3	2	1	0	
Address: —	—	—	IX05	IX04	IX03	IX02	IX01	IX00	IXR
Read/write ⇨	—	—	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	—	—	0	0	0	0	0	0	

Data Port Register

	7	6	5	4	3	2	1	0	
Address: —	DP07	DP06	DP05	DP04	DP03	DP02	DP01	DP00	DPR
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	X	X	X	X	X	X	X	X	

(4) Block diagram of LPC interface



17. Serial IRQ controller

The serial IRQ controller consists of a 6-channel serial IRQ control circuit and an LPC clock monitor / control circuit. By using this serial IRQ controller, host interrupt requests can be transferred serially through a single signal wire (SERIRQ), synchronized with the LPC clock.

- 6-channel serial IRQ control circuit

  - The 6-channel serial IRQ control circuit consists of a serial interrupt control register (SICR), 4 serial interrupt frame number registers (SIFR1 ~ 4), a protocol state machine and a serial interrupt data latch and output control.
  - For channel 0A, 0B and 1 ~ 3, if SICR : OBE bit (OBF controlled enable bit) = 0, then serial IRQ can be controlled by software setting of SICR : IRR bit. If SICR : OBE bit = 1, then software control is disabled and serial IRQ is controlled by OBF flag (Output buffer full flag) from LPC UPI0 ~ 3.
  - For channel 4, serial IRQ can be controlled by software setting of SICR : IRR bit.
  - For channel 0A and 0B, additional enable bit (SICR : EN0A/0B bit) can be used to latch and keep the OBF0 or IRR0A/0B bit status.
  - The serial interrupt data latch transfers serial IRQs serially according to their frame number. The frame number for channel 0A is fixed to “IRQ1”, for channel 0B is fixed to “IRQ12”, and the frame number for channel 1 ~ 4 are software programmable (IRQ1 ~ 15, and IRQ21 ~ 31) by setting the SIFR1 ~ 4.
  - By monitoring the SERIRQ and the LPC clock pin, the protocol state machine can detect the START frame condition. Then it starts counting the DATA frame and transfers its serial IRQs through SERIRQ. Finally it can switch to continuous/quiet mode operation by determine the STOP frame condition.
  - The serial interrupt output control support both continuous and quiet mode operation. In continuous mode operation, only the host can initiate the serial IRQs transfer; In quiet mode operation, both the host and slave (e.g. the serial IRQ controller) can initiate the serial IRQs transfer.

- LPC clock monitor / control circuit

  - The LPC clock monitor / control circuit consists of a clock-run monitor / control circuit. By monitoring the clock-run pin ($\overline{\text{CLKRUN}}$), the clock monitor / control circuit can determine whether the host has stopped LPC clock in quiet mode operation or not. If LPC clock is stopped and the controller want to initiate the serial IRQs transfer, then it can request the host to restart the LPC clock by controlling the $\overline{\text{CLKRUN}}$ pin.

(1) Register configuration of serial IRQ controller

Serial Interrupt Control Register (Lower)

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 000032H	EN0B	EN0A	IRR4	IRR3	IRR2	IRR1	IRR0B	IRR0A	SICRL
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

Serial Interrupt Control Register (Upper)

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 000033H	IRQEN	RSEN	BUSY	OBE3	OBE2	OBE1	OBE0B	OBE0A	SICRH
Read/write ⇨	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

Serial Interrupt Frame Number Register 1

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 000034H	-	-	LV1	FR14	FR13	FR12	FR11	FR10	SIFR1
Read/write ⇨	-	-	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	-	-	0	0	0	0	0	0	

Serial Interrupt Frame Number Register 2

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 000035H	-	-	LV2	FR24	FR23	FR22	FR21	FR20	SIFR2
Read/write ⇨	-	-	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	-	-	0	0	0	0	0	0	

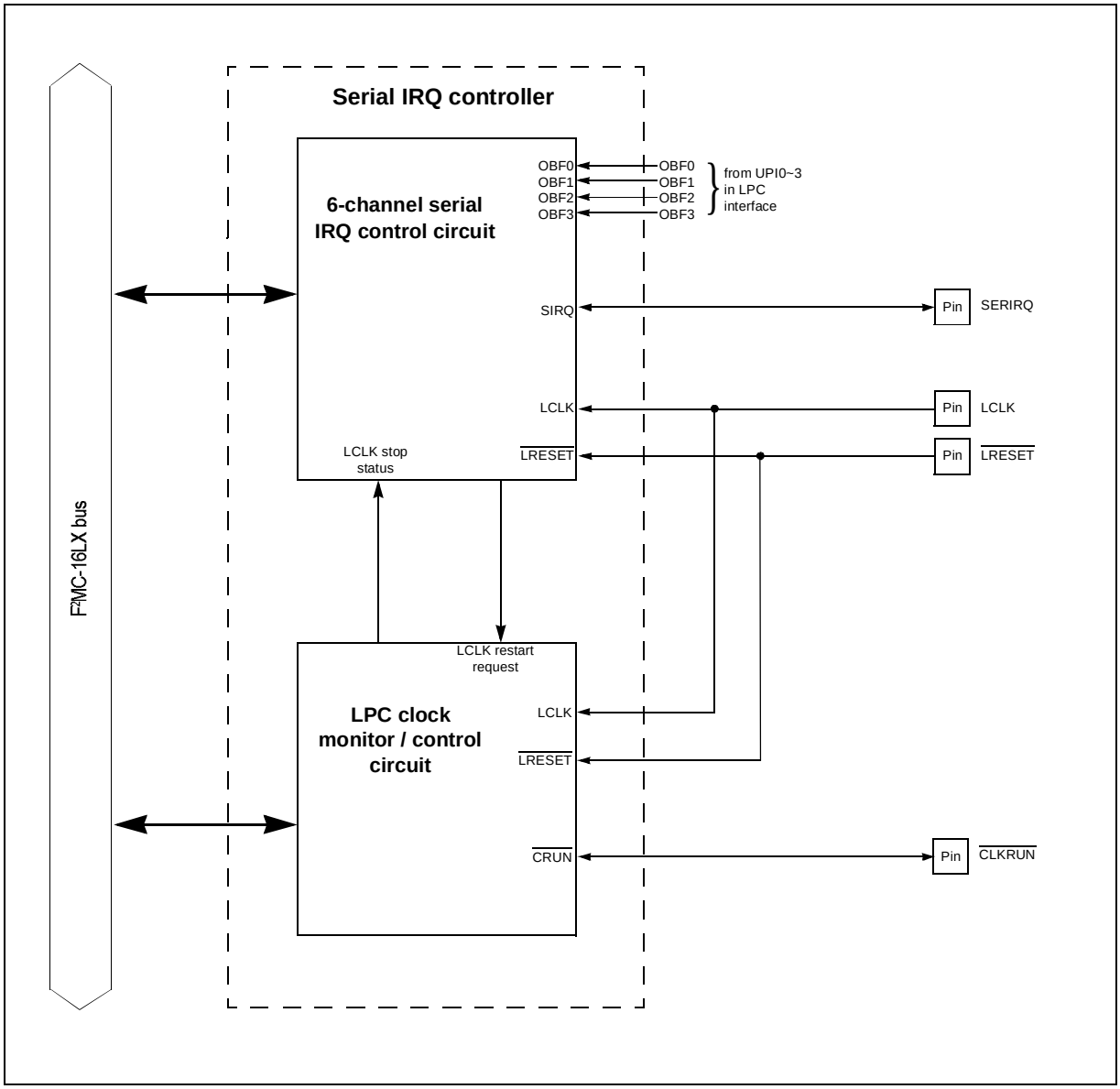
Serial Interrupt Frame Number Register 3

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 000036H	-	-	LV3	FR34	FR33	FR32	FR31	FR30	SIFR3
Read/write ⇨	-	-	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	-	-	0	0	0	0	0	0	

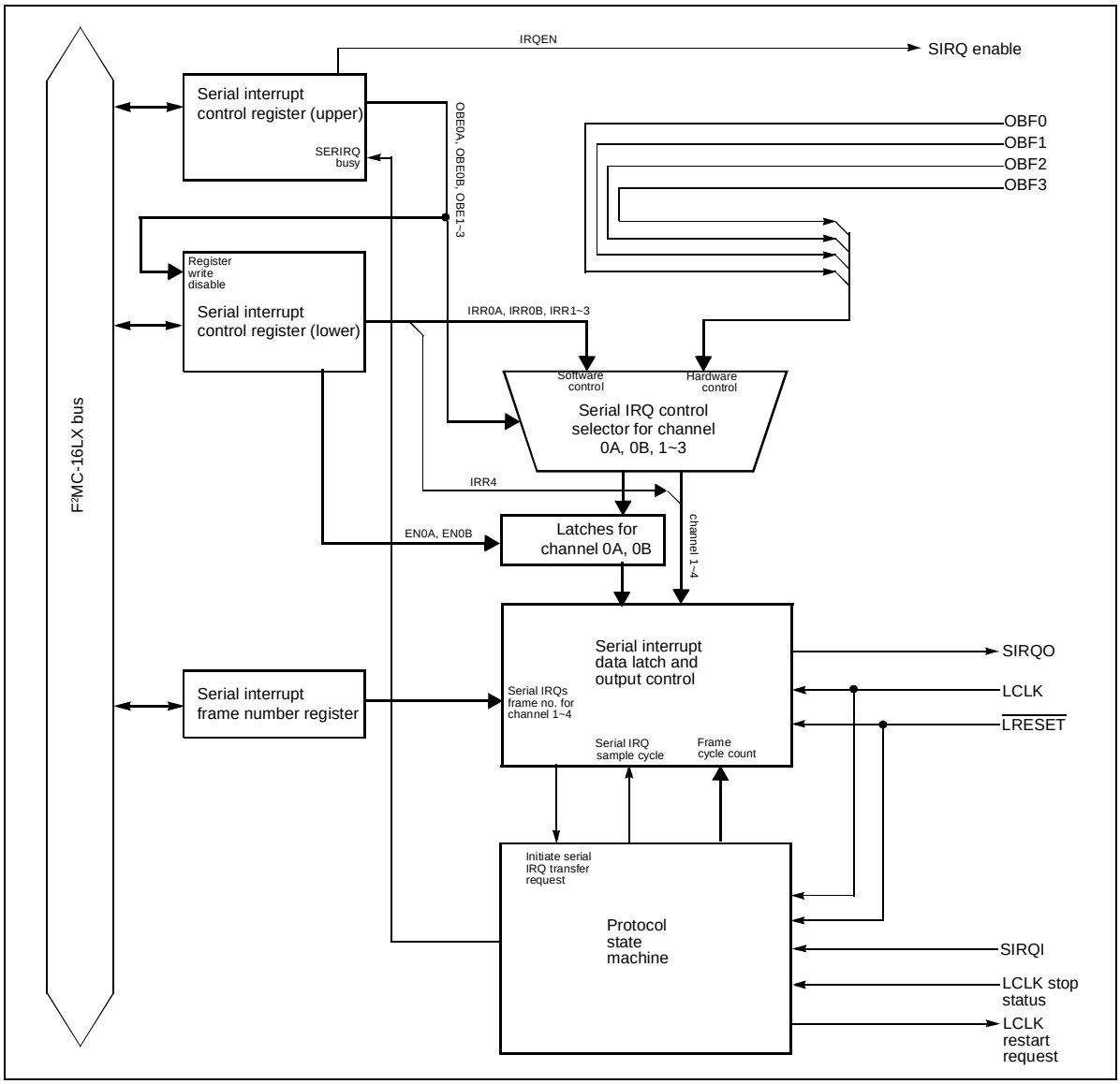
Serial Interrupt Frame Number Register 4

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 000037H	-	-	LV4	FR44	FR43	FR42	FR41	FR40	SIFR4
Read/write ⇨	-	-	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	-	-	0	0	0	0	0	0	

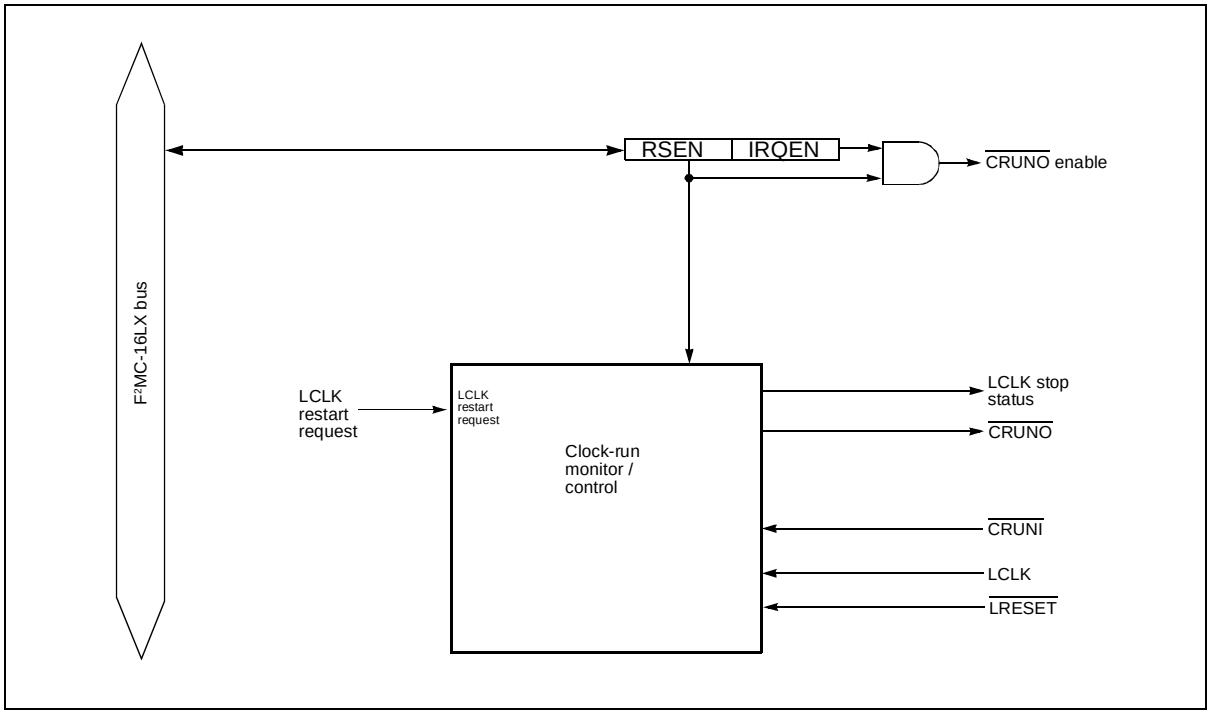
(2) Block diagram of the serial IRQ controller



(3) Block diagram of the 6-channel serial IRQ control circuit



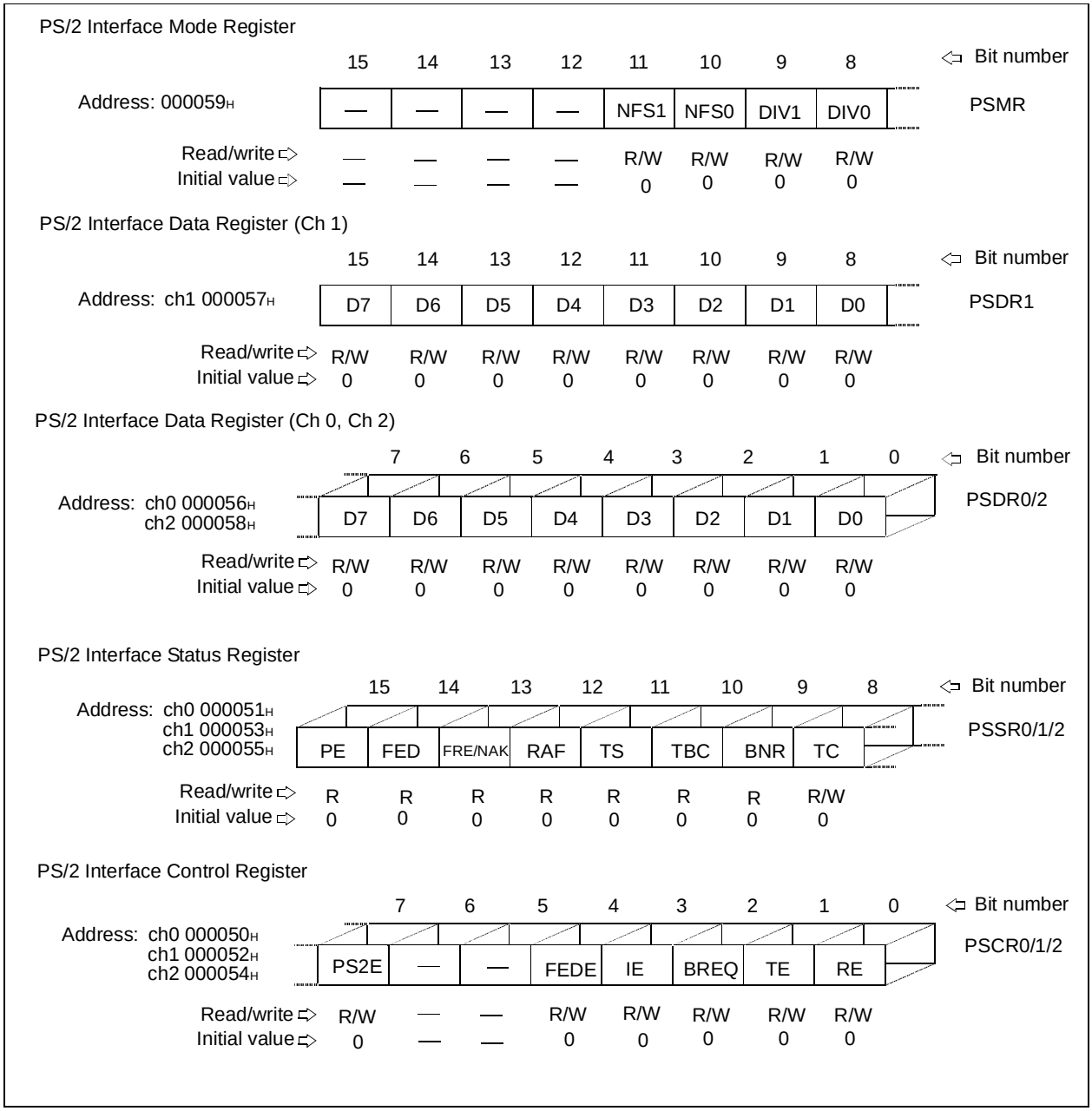
(4) Block diagram of the LPC clock monitor / control circuit



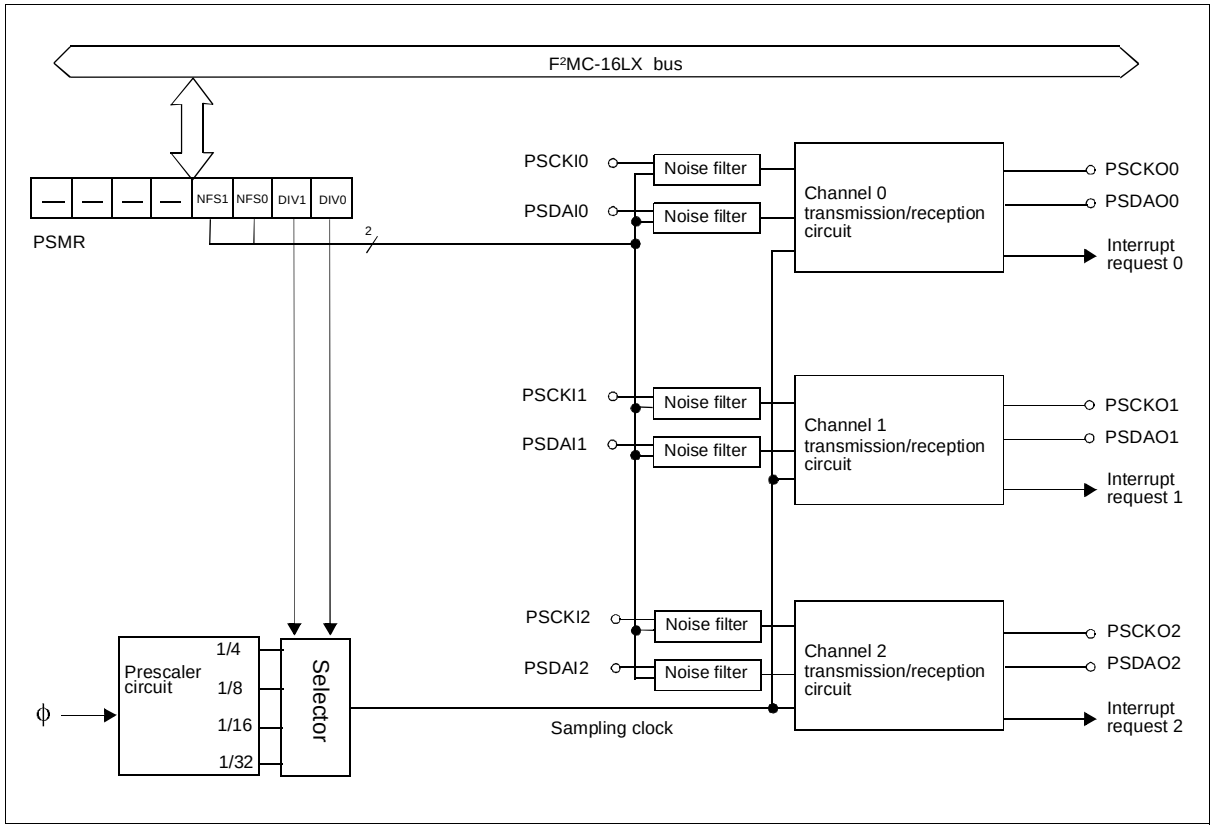
18. 3-channel PS/2 interface

The 3-channel PS/2 interface consists of 3 individual channels of PS/2 interface that can be operated concurrently. PS/2 interface is a two wires, bidirectional serial bus providing economical way for data exchange between host (keyboard controller) and device (keyboard / mouse etc).

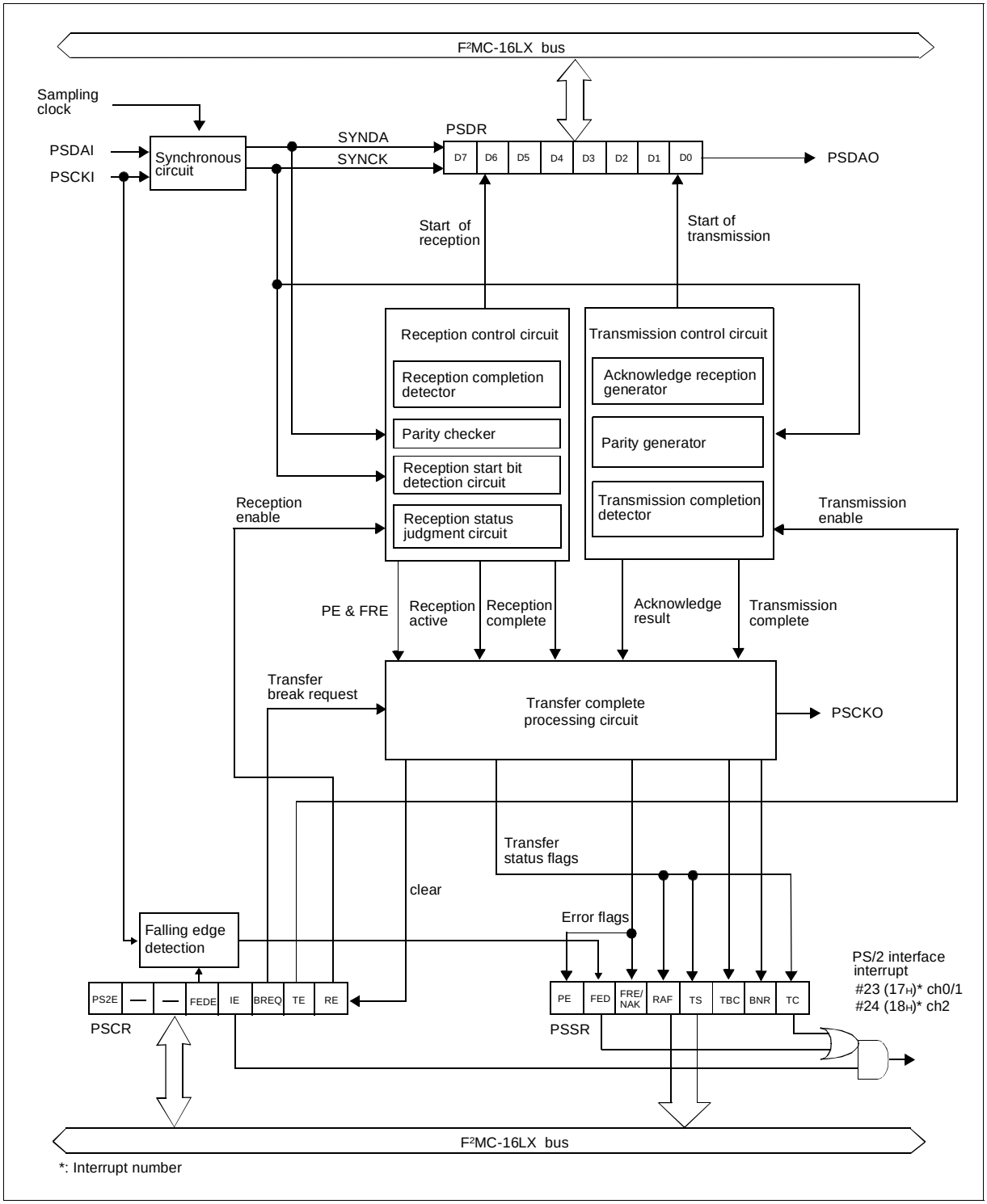
(1) Register configuration of 3-channel PS/2 interface



(2) Block diagram of 3-channel PS/2 interface



(3) Block diagram of PS/2 interface transmission/reception circuit (1 channel)



19. Parity generator

The parity generator is a simple circuit that generates odd / even parity based on the input data. It consists of a parity generator data register (PGDR), an odd / even parity generation logic and a parity generator control status register (PGCSR).

An 8-bit data can be loaded into PGDR, then the parity generator will generate odd / even parity based on the input data. Either odd or even parity can be generated by setting the PGCSR.

For odd parity generation, if the number of “1”s in the PGDR is even number, then the parity bit in PGCSR will be set to “1”, otherwise the parity bit will be set to “0”.

For even parity generation, if the number of “1”s in the PGDR is even number, then the parity bit in PGCSR will be set to “0”, otherwise the parity bit will be set to “1”.

Table shows some examples of odd / even parity generation.

Input data	Parity bit (odd parity)	Parity bit (even parity)
$0000\ 0000_B$	1	0
$0101\ 0101_B$	1	0
$1000\ 0000_B$	0	1
$1010\ 1011_B$	0	1

(1) Register configuration of parity generator

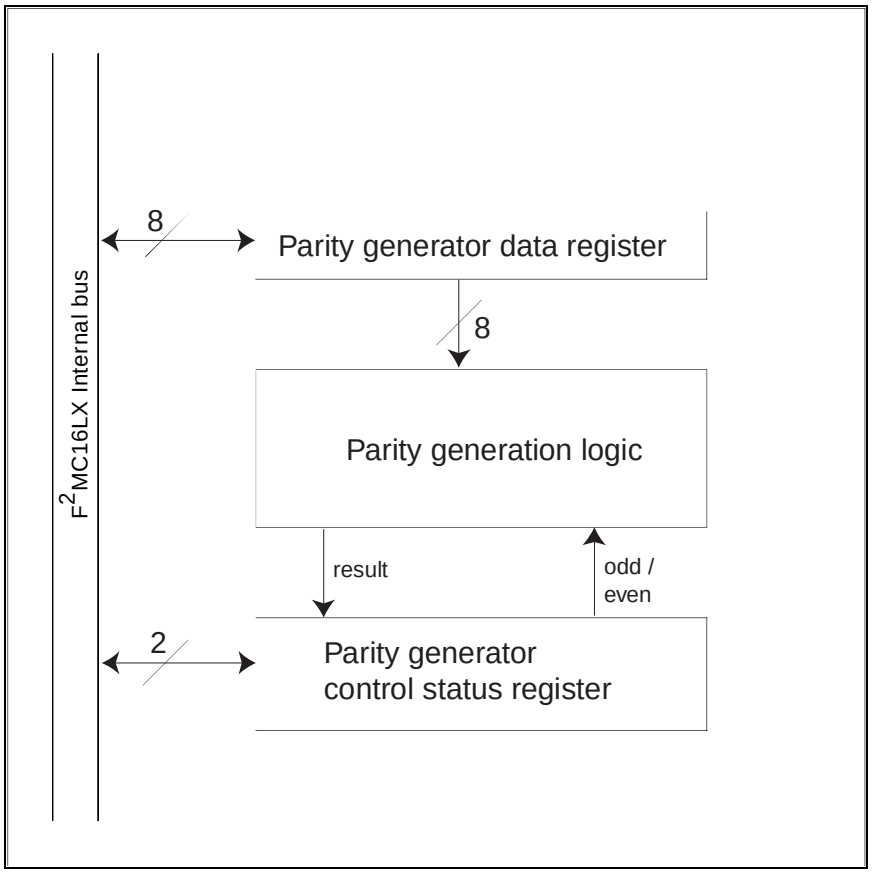
Parity Generator Data Register

	7	6	5	4	3	2	1	0	⇦ Bit number
Address : 000018_H	D7	D6	D5	D4	D3	D2	D1	D0	PGDR
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	X	X	X	X	X	X	X	X	

Parity Generator Control Status Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address : 000019_H	PRTY	—	—	—	—	—	—	PSEL	PGCSR
Read/write ⇨	R	-	-	-	-	-	-	R/W	
Initial value ⇨	X	-	-	-	-	-	-	0	

(2) Block diagram of parity generator



20. Bit decoder

The bit decoder is a simple one-hot decoder that can be used together with the keyscan inputs. It consists of a bit data register (BDR), a decoder logic and a bit result register (BRR). A 4-bit encoded data can be loaded into BDR, then the decoder logic will decode the data and store the 16-bit resulted data into BRR. Below shows the decoder's logic table.

4-bit encoded data	16-bit resulted data
0_H	$0000\ 0000\ 0000\ 0001_B$
1_H	$0000\ 0000\ 0000\ 0010_B$
2_H	$0000\ 0000\ 0000\ 0100_B$
3_H	$0000\ 0000\ 0000\ 1000_B$
4_H	$0000\ 0000\ 0001\ 0000_B$
5_H	$0000\ 0000\ 0010\ 0000_B$
6_H	$0000\ 0000\ 0100\ 0000_B$
7_H	$0000\ 0000\ 1000\ 0000_B$
8_H	$0000\ 0001\ 0000\ 0000_B$
9_H	$0000\ 0010\ 0000\ 0000_B$
A_H	$0000\ 0100\ 0000\ 0000_B$
B_H	$0000\ 1000\ 0000\ 0000_B$
C_H	$0001\ 0000\ 0000\ 0000_B$
D_H	$0010\ 0000\ 0000\ 0000_B$
E_H	$0100\ 0000\ 0000\ 0000_B$
F_H	$1000\ 0000\ 0000\ 0000_B$

(1) Register configuration of bit decoder

Bit Data Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address : $0000E1_H$	—	—	—	—	D3	D2	D1	D0	BDR
Read/write ⇨	-	-	-	-	R/W	R/W	R/W	R/W	
Initial value ⇨	-	-	-	-	X	X	X	X	

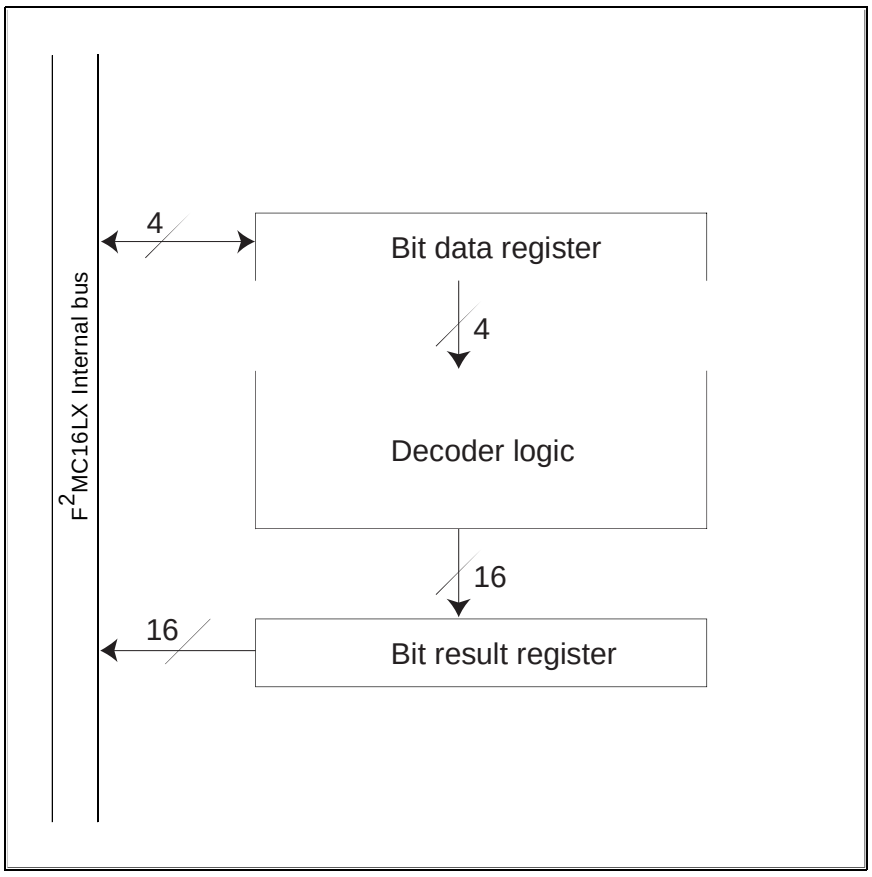
Bit Result Register (Upper)

	15	14	13	12	11	10	9	8	⇦ Bit number
Address : $0000E3_H$	R15	R14	R13	R12	R11	R10	R9	R8	BRRH
Read/write ⇨	R	R	R	R	R	R	R	R	
Initial value ⇨	X	X	X	X	X	X	X	X	

Bit Result Register (Lower)

	7	6	5	4	3	2	1	0	⇦ Bit number
Address : $0000E2_H$	R7	R6	R5	R4	R3	R2	R1	R0	BRRL
Read/write ⇨	R	R	R	R	R	R	R	R	
Initial value ⇨	X	X	X	X	X	X	X	X	

(2) Block diagram of bit decoder



21. Wake-up interrupt

The wake-up interrupt circuit detects the signals of the “L” levels input to the external interrupt pins and to generate interrupt request to the CPU. These interrupts can wake up the CPU from standby mode.

Wake-up interrupt pins:	8 pins (P00/KSI0 to P07/KSI7).
Wake-up interrupt sources:	“L” level signal input to a wake-up interrupt pin.
Interrupt control:	Enables or disables to input wake-up interrupt controlled by wake-up interrupt control register (EICR).
Interrupt flag:	IRQ flag bit of wake-up interrupt flag register (EIFR). Flag set when there is an IRQ.
Interrupt request:	Interrupt request #20 is generated if any enabled external interrupt pin goes LOW.

(1) Register configuration of wake-up interrupt

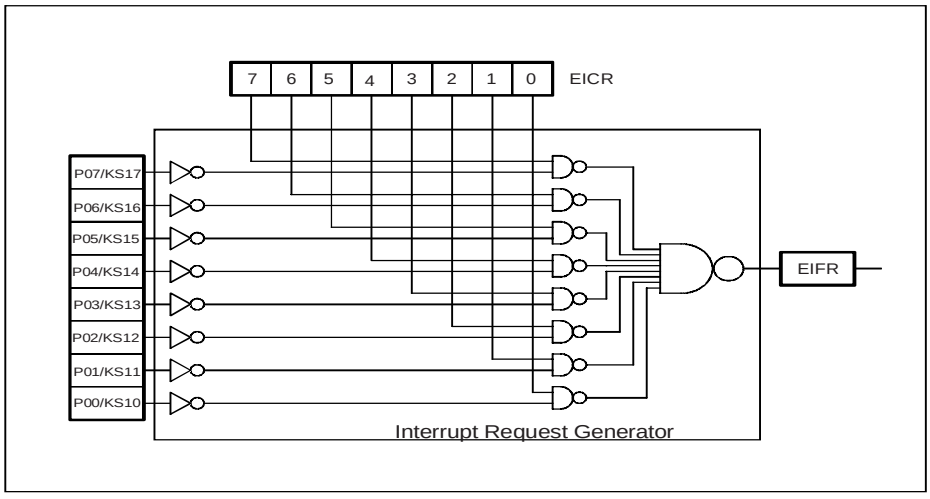
Wake-up Interrupt Flag Register

	15	14	13	12	11	10	9	8	
									⇦ Bit number
Address: 0000AD$_H$	—	—	—	—	—	—	—	WIF	EIFR
Read/write ⇨	—	—	—	—	—	—	—	R/W	
Initial value ⇨	—	—	—	—	—	—	—	0	

Wake-up Interrupt Control Register

	7	6	5	4	3	2	1	0	
									⇦ Bit number
Address: 0000AC$_H$	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0	EICR
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

(2) Block diagram of wake-up interrupt



22. DTP/External interrupts

The DTP (Data Transfer Peripheral)/external interrupt circuit is activated by the signal supplied to a DTP/external interrupt pin. The CPU accepts the signal using the same procedure it uses for normal hardware interrupts and generates external interrupts or activates the extended intelligent I/O service (EI^{2}OS).

Features of DTP/External interrupt :

- Total 6 external interrupt channels
- Two request levels (“H” and “L”) are provided for the intelligent I/O service
- Four request levels (rise/fall edge, fall edge, “H” level and “L” level) are provided for external interrupt requests

(1) Register configuration

DTP/Interrupt Source Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 000027$_H$	—	—	ER5	ER4	ER3	ER2	ER1	ER0	EIRR
Read/write ⇨	—	—	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	—	—	0	0	0	0	0	0	

DTP/Interrupt Enable Register

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 000026$_H$	—	—	EN5	EN4	EN3	EN2	EN1	EN0	ENIR
Read/write ⇨	—	—	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	—	—	0	0	0	0	0	0	

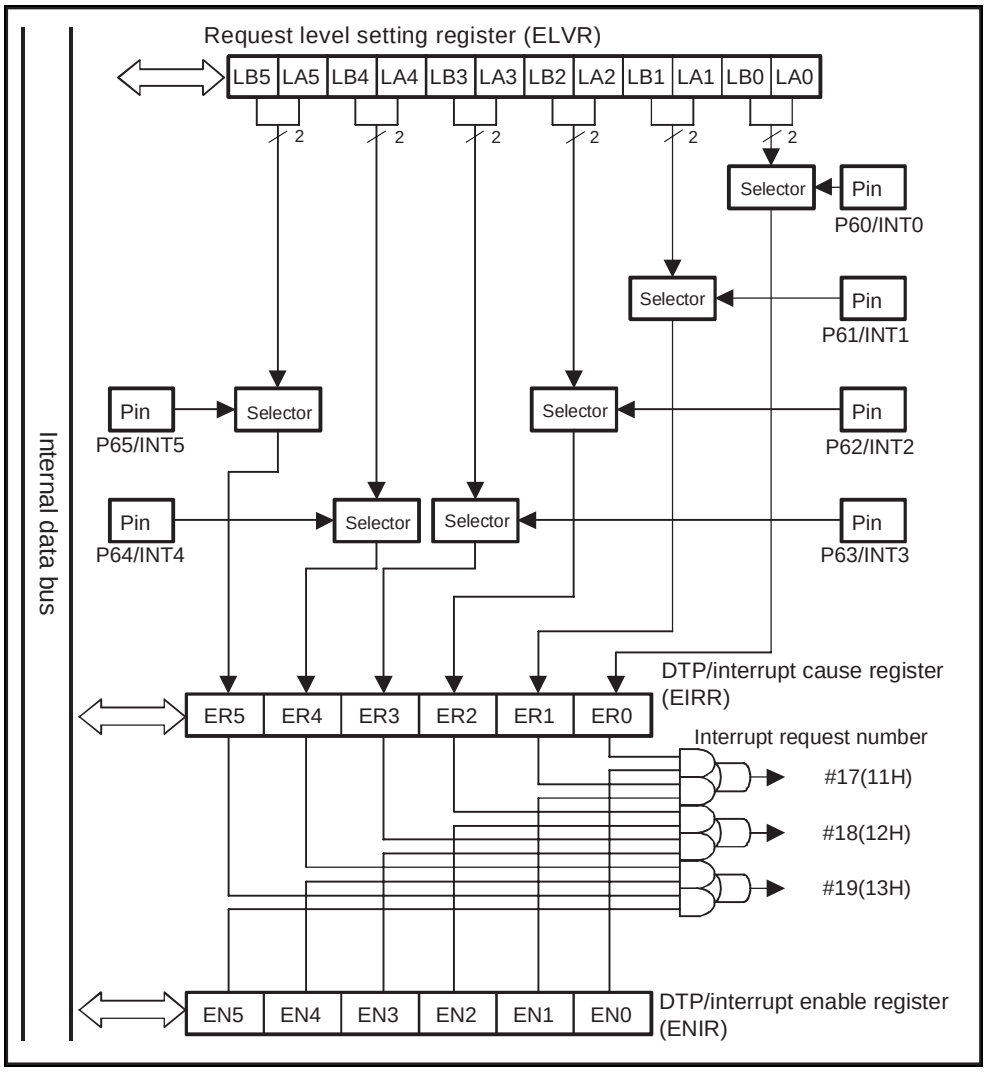
Request Level Setting Register (Upper)

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 000029$_H$	—	—	—	—	LB5	LA5	LB4	LA4	ELVRH
Read/write ⇨	—	—	—	—	R/W	R/W	R/W	R/W	
Initial value ⇨	—	—	—	—	0	0	0	0	

Request Level Setting Register (Lower)

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: 000028$_H$	LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0	ELVRL
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	0	0	0	0	0	0	0	0	

(2) Block diagram of DTP/External interrupts



23. Delayed interrupt generation module

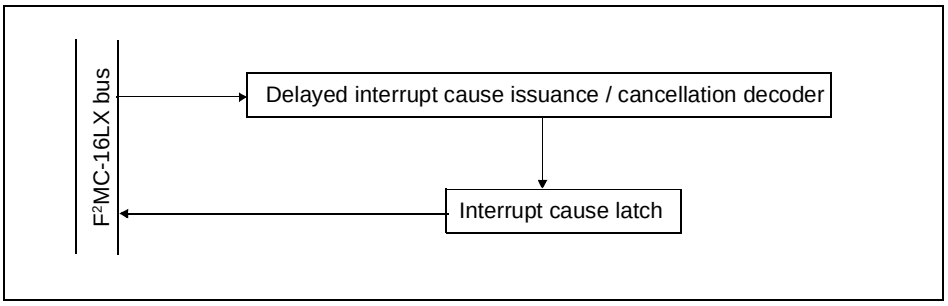
The delayed interrupt generation module is used to generate a task switching interrupt. Interrupt requests to the F²MC-16LX CPU can be generated and cleared by software using this module.

(1) Register configuration

Delayed Interrupt Generator Module Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: 00009F$_H$	—	—	—	—	—	—	—	R0	DIRR
Read/write ⇨	—	—	—	—	—	—	—	R/W	
Initial value ⇨	—	—	—	—	—	—	—	0	

(2) Block diagram



24. ROM correction function

When an address matches the value set in the address detection register, the instruction code to be loaded into the CPU is forced to be replaced with the INT9 instruction code (01_H). When executing a set instruction, the CPU executes the INT9 instruction. The address match detection function is implemented by processing using the INT9 interrupt routine.

The device contains two address detection registers, each provided with a compare enable bit. When the value set in the address detection register matches an address and the interrupt enable bit is “1”, the instruction code to be loaded into the CPU is forced to be replaced with the INT9 instruction code.

(1) Register configuration

Program Address Detection Control / Status Register

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: $00009E_H$	—	—	—	—	AD1E	AD1D	AD0E	AD0D	PACSR
Read/write ⇨	—	—	—	—	R/W	R/W	R/W	R/W	
Initial value ⇨	—	—	—	—	0	0	0	0	

Program Address Detection Register 0 (Upper Byte)

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: $001FF2_H$									PADRH0
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	X	X	X	X	X	X	X	X	

Program Address Detection Register 0 (Middle Byte)

	15	14	13	12	11	10	9	8	⇦ Bit number
Address: $001FF1_H$									PADRM0
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	X	X	X	X	X	X	X	X	

Program Address Detection Register 0 (Lower Byte)

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: $001FF0_H$									PADRL0
Read/write ⇨	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value ⇨	X	X	X	X	X	X	X	X	

(Continued)

(Continued)

Program Address Detection Register 1 (Upper Byte)

	15	14	13	12	11	10	9	8	
Address: 001FF5H									PADRH1
Read/write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	X	X	X	X	X	X	X	X	

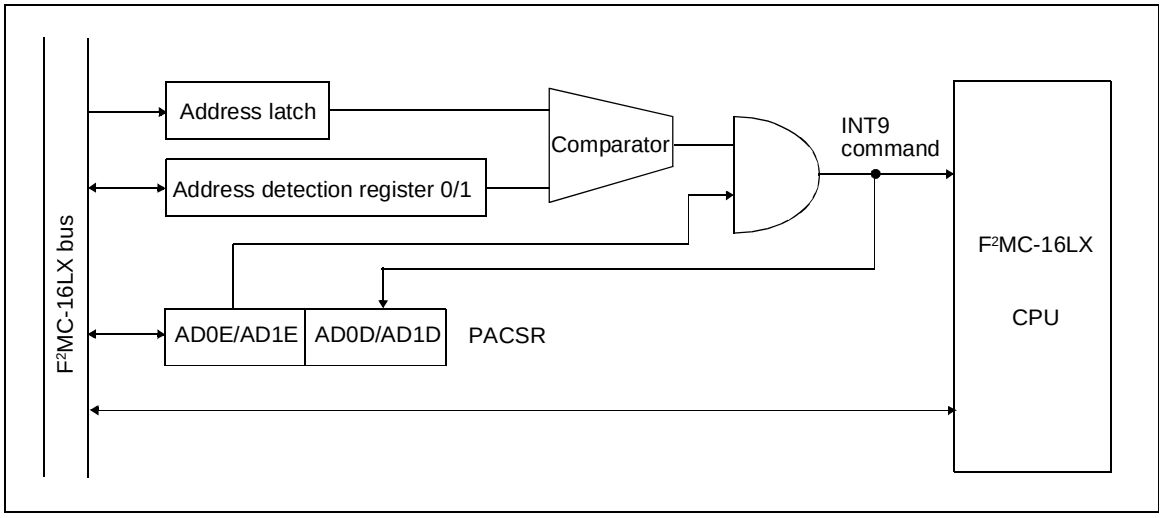
Program Address Detection Register 1 (Middle Byte)

	7	6	5	4	3	2	1	0	
Address: 001FF4H									PADRM1
Read/write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	X	X	X	X	X	X	X	X	

Program Address Detection Register 1 (Lower Byte)

	15	14	13	12	11	10	9	8	
Address: 001FF3H									PADRL1
Read/write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	X	X	X	X	X	X	X	X	

(2) Block diagram



25. ROM mirroring function selection module

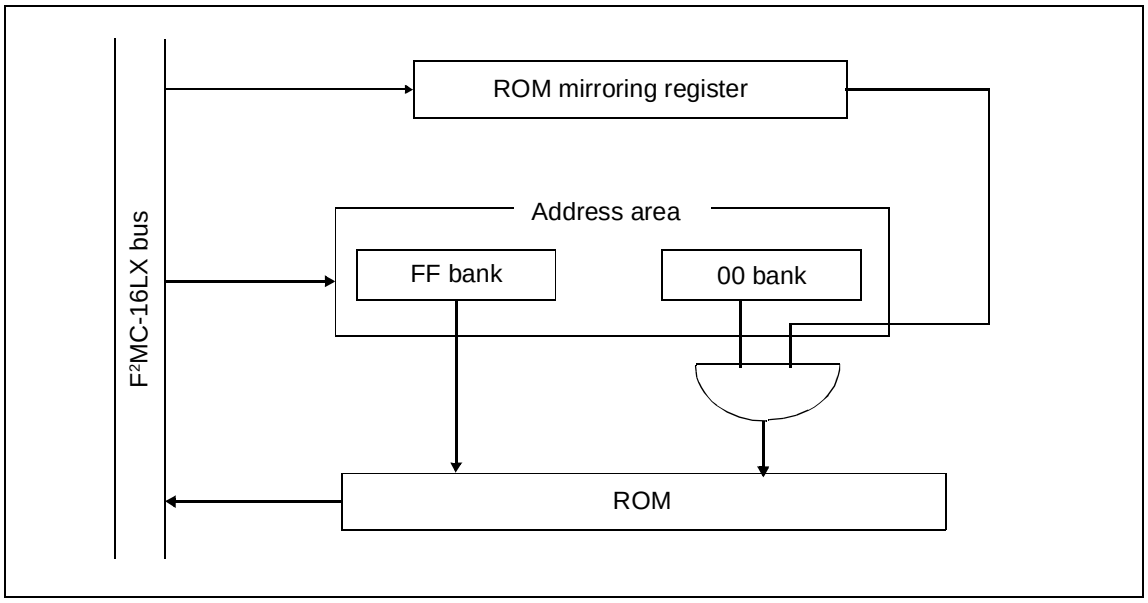
The ROM mirroring function selection module can select what the FF bank allocated the ROM sees through the 00 bank according to register settings.

(1) Register configuration

ROM Mirror Function Selection Register

	15	14	13	12	11	10	9	8	⇦ Bit number
Address : 0006F$_H$	—	—	—	—	—	—	—	M1	ROMM
Read/write ⇨	—	—	—	—	—	—	—	W	
Initial value ⇨	—	—	—	—	—	—	—	1	

(2) Block diagram



26. 512K bit flash memory

The 512K bit flash memory is allocated in the FE_H to FF_H banks on the CPU memory map. Like masked ROM, flash memory is read-accessible and program-accessible to the CPU using the flash memory interface circuit. The flash memory can be programmed/erased by the instruction from the CPU via the flash memory interface circuit. The flash memory can therefore be reprogrammed (updated) while still on the circuit board under integrated CPU control, allowing program code and data to be improved efficiently.
Note that sector operations such as “enable sector protect” cannot be used.

Features of 512K bit flash memory :

- 64K words x 8 bits / 32K words x 16 bits (16K + 8K + 8K + 32K) sector configuration
- Automatic program algorithm (same as the Embedded Algorithm* : MBM29F400TA)
- Installation of the deletion temporary stop/delete restart function
- Write/delete completion detected by the data polling or toggle bit
- Write/delete completion detected by the CPU interrupt
- Compatibility with the JEDEC standard-type command
- Each sector deletion can be executed (Sectors can be freely combined)
- Number of write/delete operations 10,000 times guaranteed

* : Embedded Algorithm is a trademark of Advanced Micro Devices, Inc.

(1) Register configuration

Flash Memory Control Status Register

	7	6	5	4	3	2	1	0	⇦ Bit number
Address: $0000AE_H$	INTE	RDYINT	WE	RDY	Reserved	LPM1	Reserved	LPM0	FMCS
Read/write ⇨	R/W	R/W	R/W	R	W	R/W	W	R/W	
Initial value ⇨	0	0	0	1	0	0	0	0	

(2) Sector configuration of 512K bit flash memory

The 512K bit flash memory has the sector configuration illustrated below. The addresses in the illustration are the upper and lower addresses of each sector.
When accessed from the CPU, SA0 and SA1 to SA3 are allocated in the FF bank registers, respectively.

Flash memory	CPU address	*Writer address
SA3 (16 Kbytes)	$FFFFFF_H$	$7FFFF_H$
	$FFC000_H$	$7C000_H$
SA2 (8 Kbytes)	$FFBFFF_H$	$7BFFF_H$
	$FFA000_H$	$7A000_H$
SA1 (8 Kbytes)	$FF9FFF_H$	$79FFF_H$
	$FF8000_H$	78000_H
SA0 (32 Kbytes)	$FF7FFF_H$	$77FFF_H$
	$FF0000_H$	70000_H

* : Writer addresses correspond to CPU addresses when data is programmed in flash memory by a parallel writer. Writer addresses are used to program/erase data using a general-purpose writer.

■ ELECTRICAL CHARACTERISTICS

1. Absolute Maximum Ratings

($V_{SS} = AV_{SS} = CV_{SS}$ = 0.0 V)

Parameter	Symbol	Value		Unit	Remarks
		Min.	Max.		
Power supply voltage	V_{CC}	V_{SS} – 0.3	V_{SS} + 4.0	V	
	CV_{CC}	V_{SS} – 0.3	V_{SS} + 4.0	V	$V_{CC} \geq CV_{CC}$ *1
	AV_{CC}	V_{SS} – 0.3	V_{SS} + 4.0	V	$V_{CC} \geq AV_{CC}$ *1
A/D converter reference input voltage	AVR	V_{SS} – 0.3	V_{SS} + 4.0	V	$AV_{CC} \geq AVR$, $AVR \geq AV_{SS}$
Comparator reference input voltage	CVRH1 CVRH2 CVRL	V_{SS} – 0.3	V_{SS} + 4.0	V	$CV_{CC} \geq CVRH1$, $CVRH1 \geq CV_{SS}$ $CV_{CC} \geq CVRH2$, $CVRH2 \geq CV_{SS}$ $CV_{CC} \geq CVRL$, $CVRL \geq CV_{SS}$
LCD power supply voltage	V1 ~ V3	V_{SS} – 0.3	V_{SS} + 4.0	V	V1 to V3 must not exceed V_{CC}
Input voltage	V_{I1}	V_{SS} – 0.3	V_{SS} + 4.0	V	All pins except P40 ~ P45, P80 ~ P82, P90 ~ P95 *2
	V_{I2}	V_{SS} – 0.3	V_{SS} + 6.0	V	P40 ~ P45, P80 ~ P82, P90 ~ P95
Output voltage	V_O	V_{SS} – 0.3	V_{SS} + 4.0	V	*2
Maximum clamp current	I_{CLAMP}	-2.0	+2.0	mA	*4
Total maximum clamp current	$\Sigma\lvert I_{CLAMP}\rvert$	—	20	mA	*4
“L” level maximum output current	I_{OL1}	—	10	mA	All pins except PF0 ~ PF7*3
	I_{OL2}	—	20	mA	PF0 ~ PF7*3
“L” level average output current	I_{OLAV1}	—	4	mA	All pins except PF0 ~ PF7 Average output current = operating current × operating efficiency
	I_{OLAV2}	—	12	mA	PF0 ~ PF7 Average output current = operating current × operating efficiency
“L” level total maximum output current	ΣI_{OL}	—	100	mA	
“L” level total average output current	ΣI_{OLAV}	—	50	mA	Average output current = operating current × operating efficiency
“H” level maximum output current	I_{OH}	—	–10	mA	*3
“H” level average output current	I_{OHAV}	—	–3	mA	Average output current = operating current × operating efficiency
“H” level total maximum output current	ΣI_{OH}	—	–100	mA	
“H” level total average output current	ΣI_{OHAV}	—	–50	mA	Average output current = operating current × operating efficiency
Power consumption	P_D	—	200	mW	
Operating temperature	T_A	–40	+85	°C	

Parameter	Symbol	Value		Unit	Remarks
		Min.	Max.		
Storage temperature	Tstg	–55	+150	°C	

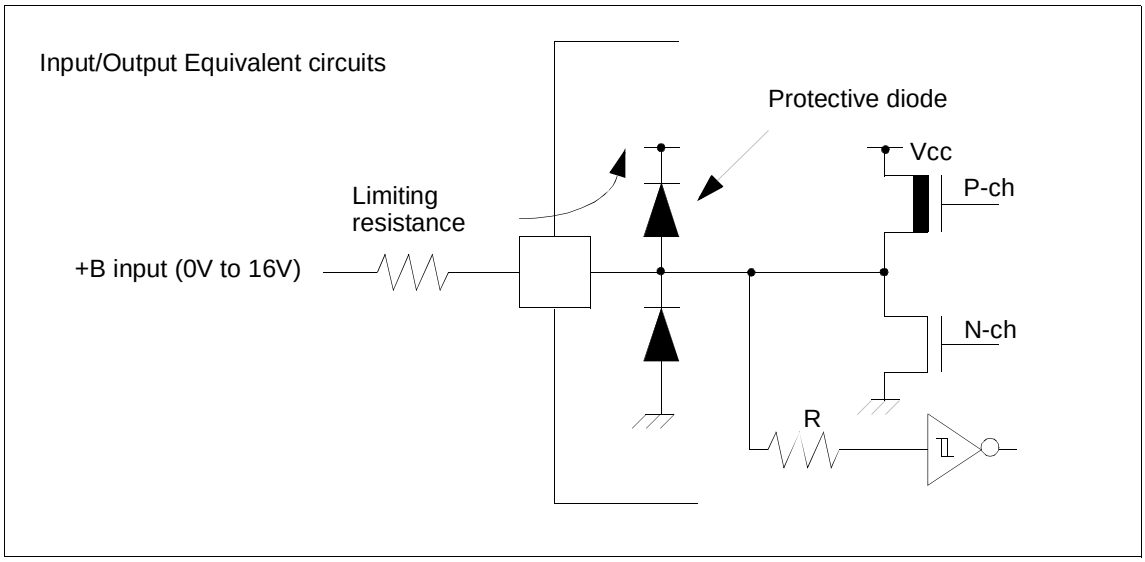
*1 : Set AV_{CC}, CV_{CC} and V_{CC} at the same voltage. Take care so that AVR, CVRH1, CVRH2 and CVRL do not exceed V_{CC} + 0.3 V when the power is turned on.

*2 : V_I and V_O shall never exceed V_{CC} + 0.3 V.

*3 : The maximum output current is a peak value for a corresponding pin.

*4 : - Use within recommended operating conditions.

- Use at DC voltage (current).
- The +B signal should always be applied a limiting resistance placed between the +B signal and the microcontroller.
- The value of the limiting resistance should be set so that when the +B signal is applied the input current to the microcontroller pin does not exceed rated values, either instantaneously or for prolonged periods.
- Note that when the microcontroller drive current is low, such as in the power saving modes, the +B input potential may pass through the protective diode and increase the potential at the Vcc pin, and this may affect other devices.
- Note that if a +B signal is input when the microcontroller power supply is off (not fixed at 0V), the power supply is provided from the pins, so that incomplete operation may result.
- Note that if the +B input is applied during power-on, the power supply is provided from the pins and the resulting supply voltage may not be sufficient to poerate the power-on reset.
- Care must be taken not to leave the +B input pin open.
- Note that analog system input/output pins other than the A/D input pins (LCD drive pins, comparator input pins, etc.) cannot accept +B signal input.
- Sample recommended circuits:



WARNING: Semiconductor devices can be permanently damaged by application of stress (voltage, current, temperature, etc.) in excess of absolute maximum ratings. Do not exceed these ratings.

2. Recommended Operating Conditions

(V_{SS} = AV_{SS} = CV_{SS} = 0.0 V)

Parameter	Symbol	Value		Unit	Remarks
		Min.	Max.		
Power supply voltage *2	V_{CC}	3.0 *1	3.6	V	Normal operation assurance range
	CV_{CC}	3.3	3.6	V	
	V_{CC}	1.8	3.6	V	Retains the RAM state in stop mode
A/D converter reference input voltage *3	AVR	0	AV_{CC}	V	Normal operation assurance range
LCD power supply voltage	V1 ~ V3	V_{SS}	V_{CC}	V	V1 ~ V3 pins (The optimum value is dependent on the LCD element in use.)
Operating temperature	T_A	–40	+85	°C	

*1 : The operating voltage varies with the operation frequency.
*2 : Set AV_{CC}, CV_{CC} and V_{CC} at the same voltage.
*3 : Take care so that AVR, CVRH1, CVRH2 and CVRL do not exceed V_{CC} + 0.3 V when power is turned on.

WARNING: The recommended operating conditions are required in order to ensure the normal operation of the semiconductor device. All of the device's electrical characteristics are warranted when the device is operated within these ranges.

Always use semiconductor devices within their recommended operating condition ranges. Operation outside these ranges may adversely affect reliability and could result in device failure.

No warranty is made with respect to uses, operating conditions, or combinations not represented on the data sheet. Users considering application outside the listed conditions are advised to contact their FUJITSU representatives beforehand.

3. DC Characteristics

(V_{CC} = AV_{CC} = CV_{CC} = 3.0 V to 3.6 V, V_{SS} = AV_{SS} = CV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
"H" level input voltage	V_{IH}	P10 ~ P17 P20 ~ P27 P30 ~ P37 P46 ~ P47 P50 ~ P57 PA0 ~ PA6 PB0 ~ PB7 PC0 ~ PC7 PD0 ~ PD7 PF0 ~ PF7	—	0.7 V_{CC}	—	V_{CC} + 0.3	V	CMOS input pins
	V_{IHS}	P00 ~ P07 P60 ~ P67 P70 ~ P77 $\overline{PE0}$ ~ PE7 $\overline{RST}$		0.8 V_{CC}	—	V_{CC} + 0.3	V	CMOS hysteresis input pins
	V_{IHS5}	P40 ~ P45		0.8 V_{CC}	—	V_{SS} + 5.5	V	5 V tolerant CMOS hysteresis input pins
	V_{IH5}	P82		0.7 V_{CC}	—	V_{SS} + 5.5	V	5 V tolerant CMOS input pin
	V_{IHSM}	P80 ~ P81 P90 ~ P95		2.1	—	V_{SS} + 5.5	V	SMbus input pins
	V_{IHM}	MD0 ~ MD2		V_{CC} – 0.3	—	V_{CC} + 0.3	V	Mode pins
"L" level input voltage	V_{IL}	P10 ~ P17 P20 ~ P27 P30 ~ P37 P46 ~ P47 P50 ~ P57 P82 PA0 ~ PA6 PB0 ~ PB7 PC0 ~ PC7 PD0 ~ PD7 PF0 ~ PF7		V_{SS} – 0.3	—	0.3 V_{CC}	V	CMOS input pins
	V_{ILS}	P00 ~ P07 P40 ~ P45 P60 ~ P67 P70 ~ P77 $\overline{PE0}$ ~ PE7 $\overline{RST}$		V_{SS} – 0.3	—	0.2 V_{CC}	V	CMOS hysteresis input pins
	V_{ILSM}	P80 ~ P81 P90 ~ P95		V_{SS} – 0.3	—	0.8	V	SMbus input pins
	V_{ILM}	MD0 ~ MD2		V_{SS} – 0.3	—	V_{SS} + 0.3	V	Mode pins

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
Open-drain output pin application voltage	V_{D5}	P40 ~ P45 P80 ~ P82 P90 ~ P95	—	V_{SS} – 0.3	—	V_{SS} + 5.5	V	
	V_D	P46		V_{SS} – 0.3	—	V_{CC} + 0.3	V	
“H” level output voltage	V_{OH1}	All port pins except P40 ~ P46 P80 ~ P82 P90 ~ P95 PF0 ~ PF7	V_{CC} = 3.0 V I_{OH1} = –4.0 mA	V_{CC} – 0.5	—	—	V	
	V_{OH2}	PF0 ~ PF7	V_{CC} = 3.0 V I_{OH2} = –8.0 mA	V_{CC} – 0.5	—	—	V	
“L” level output voltage	V_{OL1}	All port pins except PF0 ~ PF7	I_{OL1} = 4.0 mA	—	—	0.4	V	
	V_{OL2}	PF0 ~ PF7	I_{OL2} = 12.0 mA	—	—	0.4	V	
Input leakage current (Hi-Z output leakage current)	I_{IL}	All input pins	V_{CC} = 3.3 V, $V_{SS} < V_I < V_{CC}$	–5	—	5	µA	
Open-drain output leakage current	I_{LEAK}	P40 ~ P46 P80 ~ P82 P90 ~ P95	—	—	—	5	µA	

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
Power supply current*	I_{CC}	V_{CC}	V_{CC} = 3.3 V, Internal operation at 16 MHz	—	37	45	mA	MB90F372
				—	30	TBD	mA	MB90372
	I_{CCS}		V_{CC} = 3.3 V, Internal operation at 16 MHz, In sleep mode	—	15	20	mA	
	I_{CCL}		V_{CC} = 3.3 V, External 32 kHz, Internal operation at 8 kHz, In sub-clock mode, T_A = 25 °C	—	23	80	µA	
	I_{CCLS}		V_{CC} = 3.3 V, External 32 kHz, Internal operation at 8 kHz, In sub-clock sleep mode, T_A = 25 °C	—	10	50	µA	
	I_{CCWAT}		V_{CC} = 3.3 V, External 32 kHz, Internal operation at 8 kHz, In watch mode, T_A = 25 °C	—	1.5	30	µA	
Power supply current*	I_{CCT}	V_{CC}	V_{CC} = 3.3 V, Internal operation at 16 MHz, In timebase timer mode	—	1.3	2	mA	
	I_{CCH}		V_{CC} = 3.3 V, In stop mode, T_A = 25 °C	—	1	20	µA	
Input capacitance	C_{IN}	All input pins except V_{CC}, AV_{CC}, CV_{CC}, V_{SS}, AV_{SS}, CV_{SS}	—	—	10	80	pF	
LCD divided resistance	R_{LCD}	—	Between V_{CC} and V3 at V_{CC} = 3.3 V	100	200	400	kΩ	
			Between V3 and V2 Between V2 and V1 Between V1 and V_{SS} at V_{CC} = 3.3 V	50	100	200		

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
COM0 ~ COM3 output impedance	R_{VCOM}	COM0 ~ COM3	V1 ~ V3 = 3.3 V	—	—	5	kΩ	
SEG0 ~ SEG8 output impedance	R_{VSEG}	SEG0 ~ SEG8		—	—	5	kΩ	
LCD leakage current	L_{LCDL}	V1 ~ V3 COM0 ~ COM3 SEG0 ~ SEG8	—	—	—	±1	μA	
Pull-up resistance	R_{UP}	P00 ~ P07 P10 ~ P17 P20 ~ P27 P30 ~ P37 $\overline{RST}$	—	25	50	100	kΩ	
Pull-down resistance	R_{DOWN}	MD2	—	25	50	100	kΩ	MB90V370, MB90372 only

* : The current value is preliminary value and may be subject to change for enhanced characteristics without previous notice. The power supply current is measured with an external clock.

4. AC Characteristics

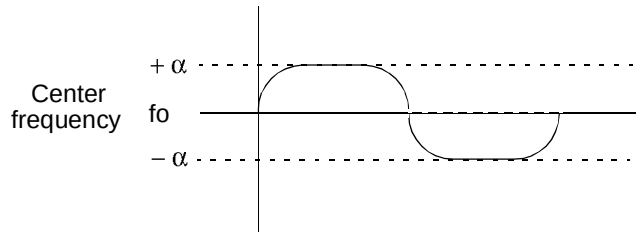
(1) Clock Timings

(V_{CC} = AV_{CC} = CV_{CC} = 3.0 V to 3.6 V, V_{SS} = AV_{SS} = CV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

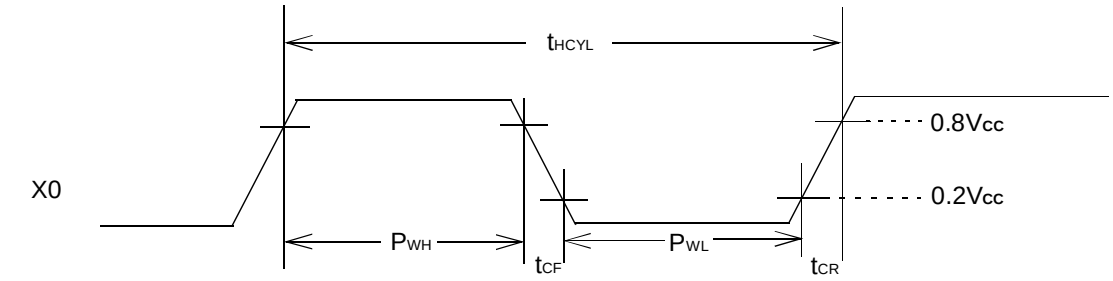
Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
Clock frequency	F_{CH}	X0, X1	—	3	—	16	MHz	Crystal oscillator
	F_{CH}	X0, X1		3	—	32	MHz	External clock
	F_{CL}	X0A, X1A		—	32.768	—	kHz	
Clock cycle time	t_{HCYL}	X0, X1		31.25	—	333	ns	
	t_{LCYL}	X0A, X1A		—	30.5	—	µs	
Frequency fluctuation rate locked*	Δf	—		—	—	5	%	
Input clock pulse width	P_{WH} P_{WL}	X0		5	—	—	ns	Recommend duty ratio of 30% to 70%
	P_{WHL} P_{WLL}	X0A		—	15.2	—	µs	Recommend duty ratio of 30% to 70%
Input clock rise/fall time	t_{CR} t_{CF}	X0		—	—	5	ns	External clock operation
Internal operating clock frequency	f_{CP}	—		1.5	—	16	MHz	Main clock operation
	f_{LCP}	—		—	8.192	—	kHz	Sub-clock operation
Internal operating clock cycle time	t_{CP}	—		62.5	—	666	ns	Main clock operation
	t_{LCP}	—		—	122.1	—	µs	Sub-clock operation

*: The frequency fluctuation rate is the maximum deviation rate of the preset center frequency when the multiplied PLL signal is locked.

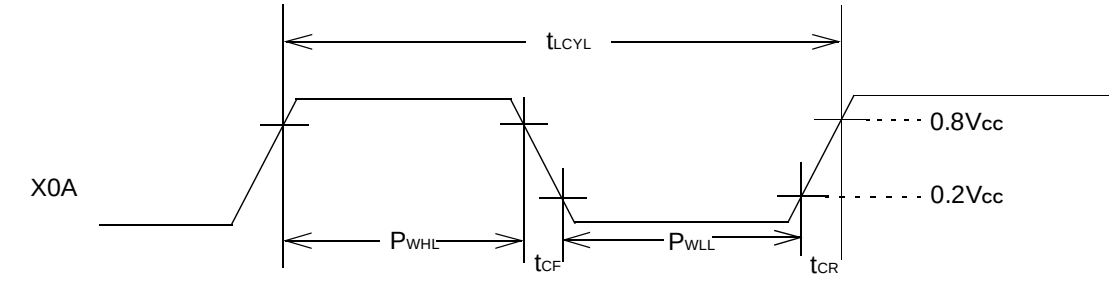
$$\Delta f = \frac{|\alpha|}{fo} \times 100\ (\%)$$



X0, X1 clock timing

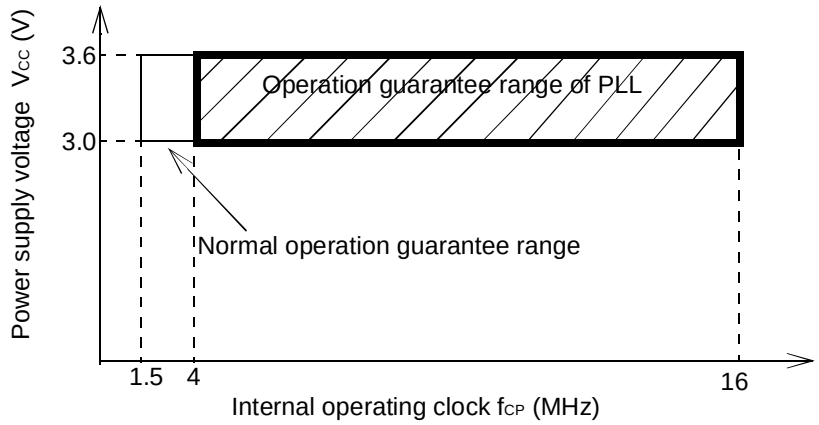


X0A, X1A clock timing

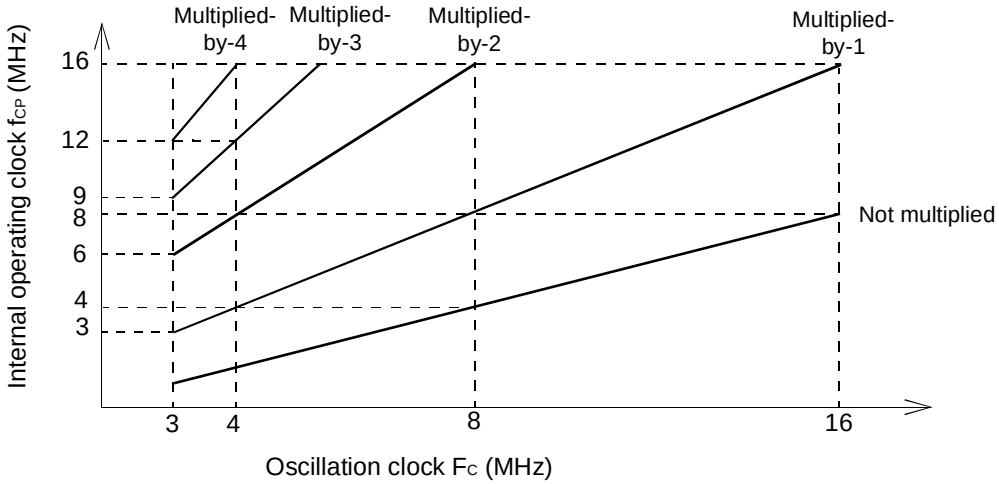


- PLL operation guarantee range

Relationship between internal operating clock frequency and power supply voltage



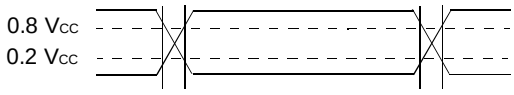
Relationship between oscillating frequency and internal operating clock frequency



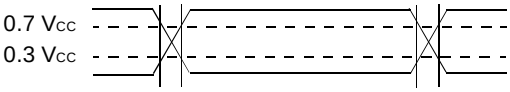
The AC ratings are measured for the following measurement reference voltages:

- Input signal waveform

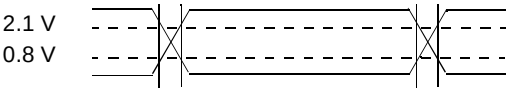
  Hysteresis input pin



  CMOS input pin

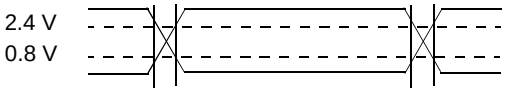


  SMbus input pin



- Output signal waveform

  Output pin

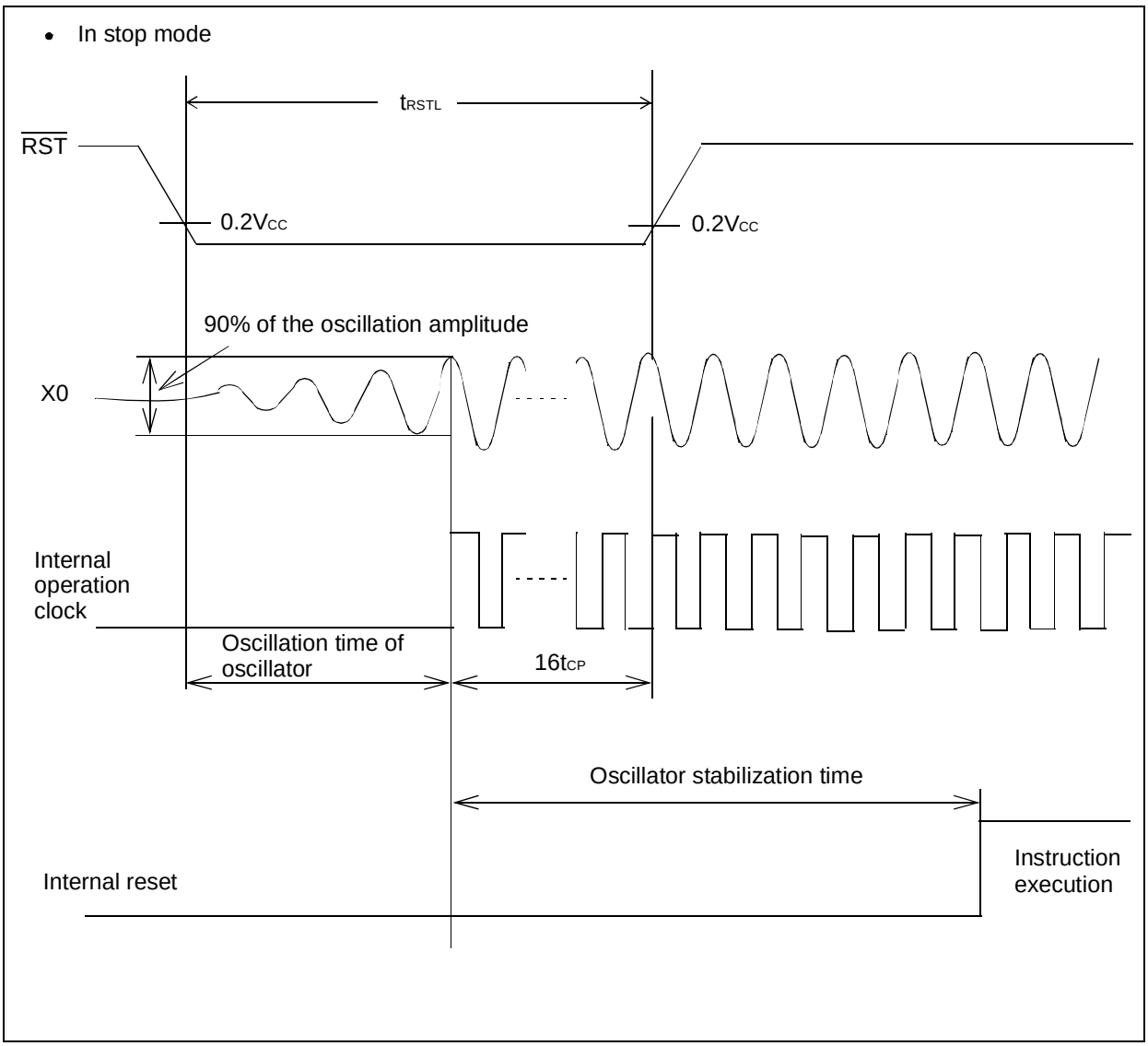


(2) Reset Input Timing

(V_{CC} = AV_{CC} = CV_{CC} = 3.0 V to 3.6 V, V_{SS} = AV_{SS} = CV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Reset input time	t_{RSTL}	$\overline{RST}$	—	16 t_{CP}	—	ns	Normal operation
				Oscillation time of oscillator* + 16 t_{CP}	—	ms	In stop mode and sub-clock mode

* : Oscillation time of oscillator is the time to reach to 90% of the oscillation amplitude from stand still. In the crystal oscillator, the oscillation time is between several ms to tens of ms. In FAR/ceramic oscillator, the oscillation time is between hundreds of µs to several ms. In the external clock, the oscillation time is 0 ms.



(3) Power-on Reset

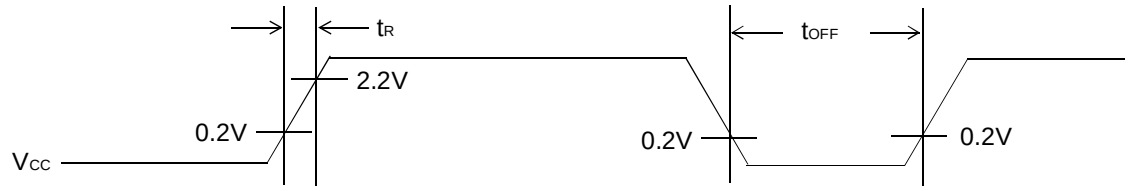
(V_{CC} = AV_{CC} = CV_{CC} = 3.0 V to 3.6 V, V_{SS} = AV_{SS} = CV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Power supply rise time	t_R	V_{CC}*	—	—	50	ms	
Power supply cut-off time	t_{OFF}	V_{CC}*		1	—	ms	Due to repeated operations

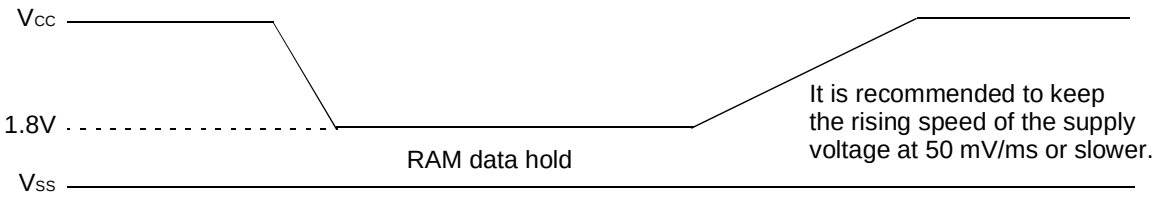
* : V_{CC} must be kept lower than 0.2 V before power-on.

Note: The above values are used for causing a power-on reset.
Some registers in the device are initialized only upon a power-on reset. To initialize these registers, turn on the power supply using the above values.

Note: Make sure that power supply rises within the selected oscillation stabilization time. If the power supply voltage needs to be varied in the course of operation, a smooth voltage rise is recommended.



Sudden changes in the power supply voltage may cause a power-on reset.
To change the power supply voltage while the device is in operation, it is recommneded to raise the voltage smoothly to suppress fluctuations as shown below. In this case, change the supply voltage with the PLL clock not used. If the voltage drop is 1 mV or fewer per second, however, you can use the PLL clock.



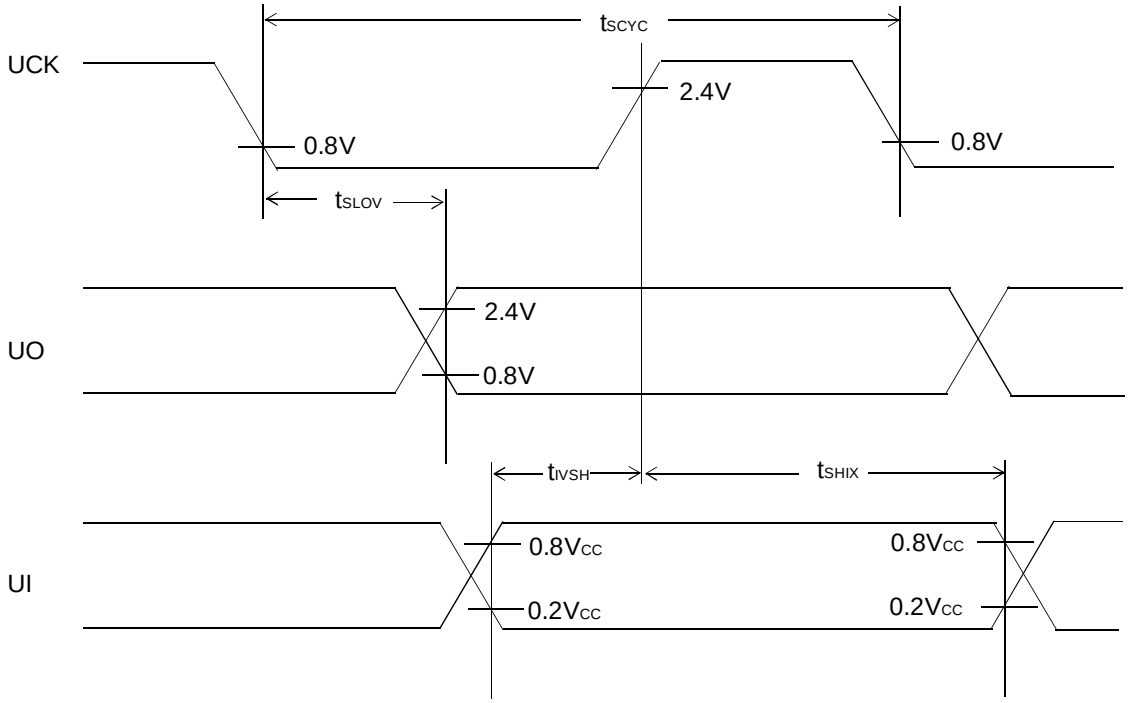
(4) UART1 to UART3

(V_{CC} = AV_{CC} = CV_{CC} = 3.0 V to 3.6 V, V_{SS} = AV_{SS} = CV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

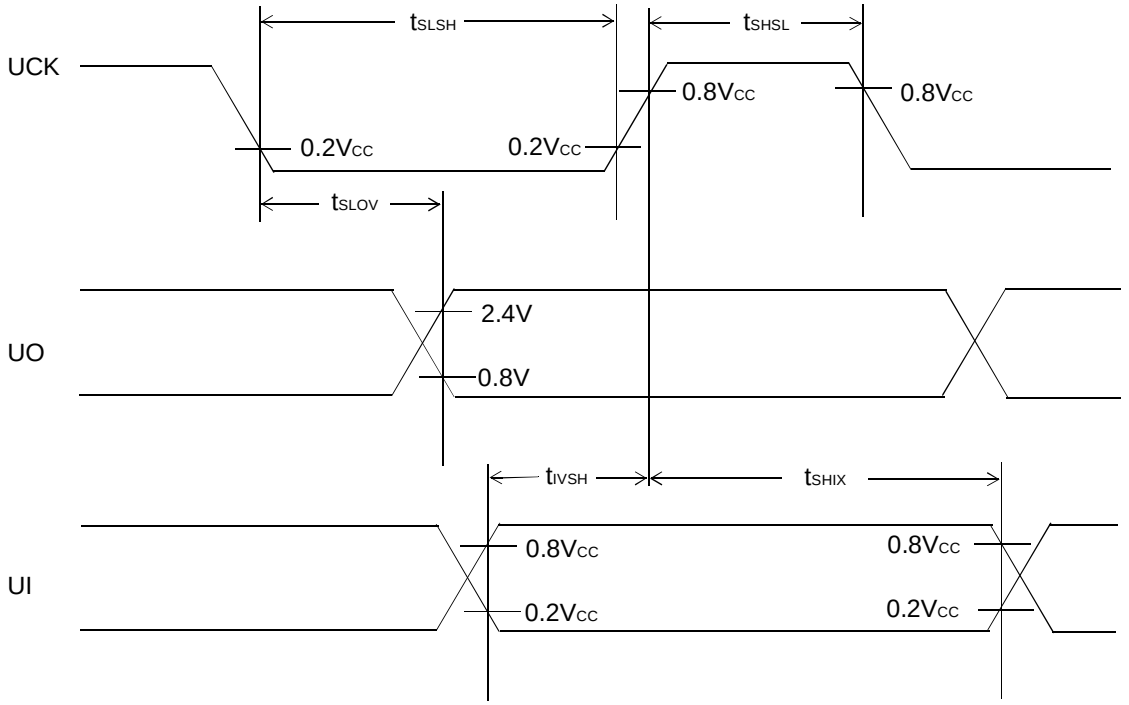
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Serial clock cycle time	t_{SCYC}	UCK1 ~ UCK3	C_L = 80 pF + 1 TTL for an output pin of internal shift clock mode	8 t_{CP}	—	ns	
UCK ↓ → UO delay time	t_{SLOV}	UCK1 ~ UCK3 UO1 ~ UO3		–80	80	ns	
Valid UI → UCK ↑	t_{IVSH}	UCK1 ~ UCK3 UI1 ~ UI3		100	—	ns	
UCK ↑ → valid UI hold time	t_{SHIX}	UCK1 ~ UCK3 UI1 ~ UI3		t_{CP}	—	ns	
Serial clock “H” pulse width	t_{SHSL}	UCK1 ~ UCK3	C_L = 80 pF + 1 TTL for an output pin of external shift clock mode	4 t_{CP}	—	ns	
Serial clock “L” pulse width	t_{SLSH}	UCK1 ~ UCK3		4 t_{CP}	—	ns	
UCK ↓ → UO delay time	t_{SLOV}	UCK1 ~ UCK3 UO1 ~ UO3		—	150	ns	
Valid UI → UCK ↑	t_{IVSH}	UCK1 ~ UCK3 UI1 ~ UI3		60	—	ns	
UCK ↑ → valid UI hold time	t_{SHIX}	UCK1 ~ UCK3 UI1 ~ UI3		60	—	ns	

Note : • These are AC ratings in the CLK synchronous mode.
• C_L is the load capacitance value connected to pins while testing.
• t_{CP} is the internal operating clock cycle time.

● Internal shift clock mode



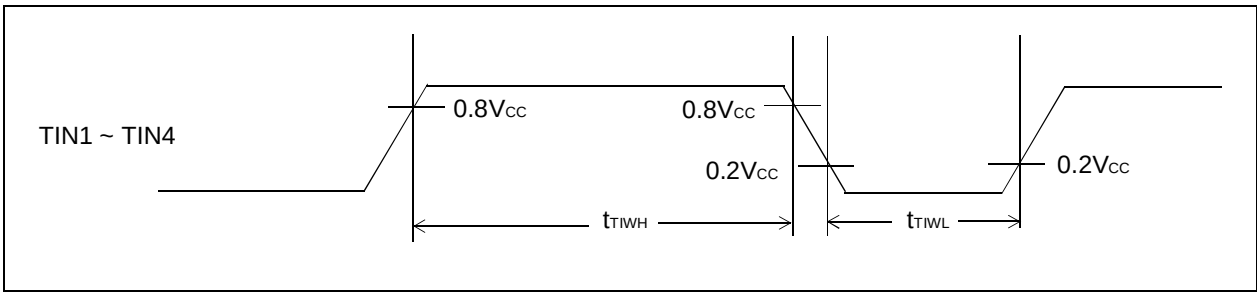
● External shift clock mode



(5) Resources Input Timing

(V_{CC} = AV_{CC} = CV_{CC} = 3.0 V to 3.6 V, V_{SS} = AV_{SS} = CV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

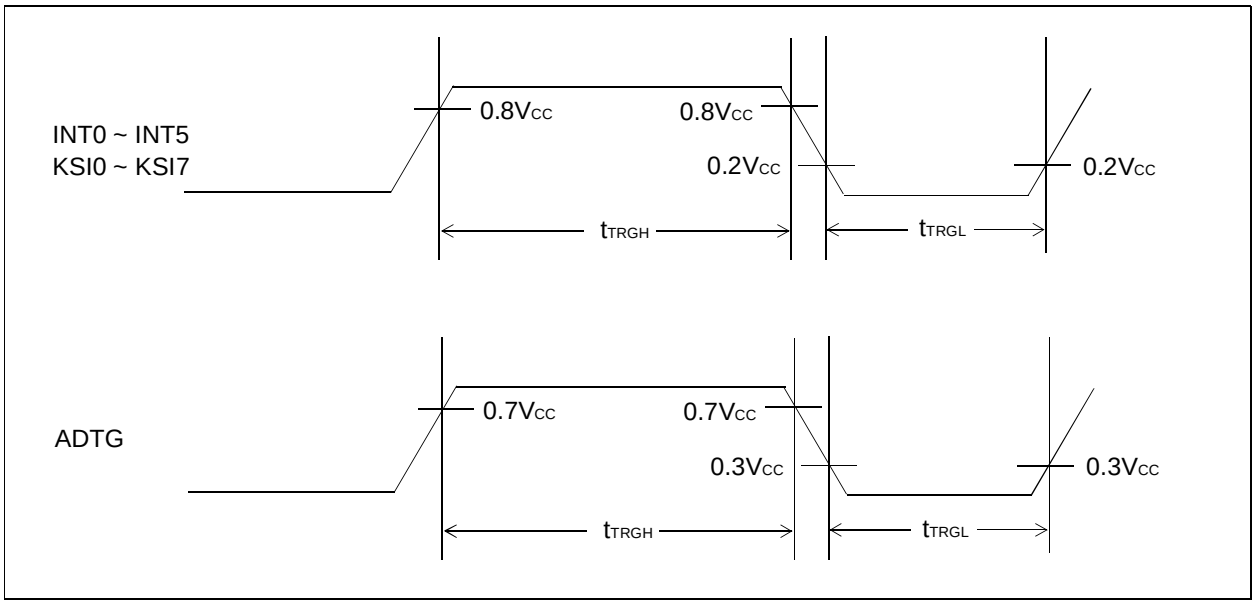
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Timer input pulse width	t_{TIWH} t_{TIWL}	TIN1 ~ TIN4	—	4 t_{CP}	—	ns	



(6) Trigger Input Timing

(V_{CC} = AV_{CC} = CV_{CC} = 3.0 V to 3.6 V, V_{SS} = AV_{SS} = CV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Input pulse width	t_{TRGH} t_{TRGL}	ADTG INT0 ~ INT5 KSI0 ~ KSI7	—	5 t_{CP}	—	ns	Normal operation
				1	—	µs	Stop mode



(7) I²C / MI²C Timing

(V_{CC} = AV_{CC} = CV_{CC} = 3.0 V to 3.6 V, V_{SS} = AV_{SS} = CV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Value		Unit	Remarks
			Min.	Max.		
Start condition output	t_{STA}	SCL SDA	t_{CP} (m x n/2 -1) - 20	t_{CP} (m x n/2 -1) + 20	ns	Master mode
Stop condition output	t_{STO}	SCL SDA	t_{CP} (m x n/2 + 3) - 20	t_{CP} (m x n/2 + 3) + 20	ns	Master mode
Start condition detect	t_{STA}	SCL SDA	t_{CP} + 40	—	ns	
Stop condition detect	t_{STO}	SCL SDA	t_{CP} + 40	—	ns	
Restart condition output	t_{STASU}	SCL SDA	t_{CP} (m x n/2 + 3) - 20	t_{CP} (m x n/2 +3) + 20	ns	Master mode
Restart condition detect	t_{STASU}	SCL SDA	t_{CP} + 40	—	ns	
SCL output “L” width	t_{LOW}	SCL	t_{CP} x m x n/2 - 20	t_{CP} x m x n/2 + 20	ns	Master mode
SCL output “H” width	t_{HIGH}	SCL	t_{CP} (m x n/2 + 2) - 20	t_{CP} (m x n/2 + 2) + 20	ns	Master mode
SDA output delay	t_{DO}	SDA	t_{CP} x 3 - 20	t_{CP} x 3 + 20	ns	
SDA output setup time after interrupt	t_{DOSU}	SDA	t_{CP} x m x n/2 - 20	—	ns	*1
			t_{CP} x 4 - 20	—	ns	*2
SCL input “L” pulse	t_{LOW}	SCL	t_{CP} x 3 + 40	—	ns	
SCL input “H” pulse	t_{HIGH}	SCL	t_{CP} + 40	—	ns	
SDA output setup time	t_{SU}	SDA	40	—	ns	
SDA hold time	t_{HO}	SDA	0	—	ns	

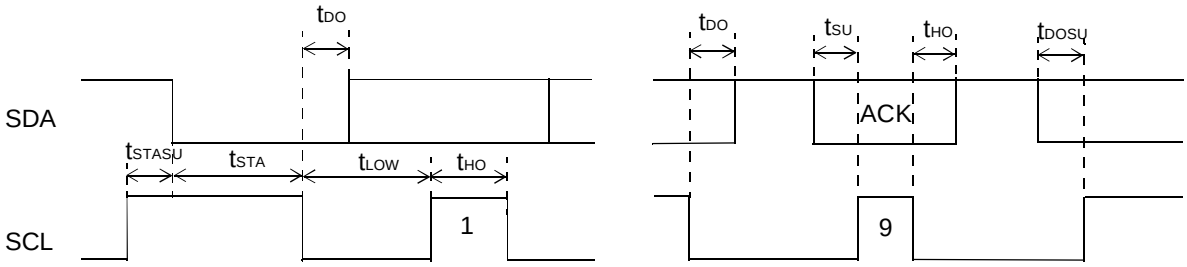
Note

- t_{CP} is the internal operating clock cycle time.
- m is the setting bit of shift clock oscillation defined in the “ICCR register (CS4 ~ CS3)” and “MCCR register (CS4 ~ CS3)”. Please refer to the MB90370 series H/W manual for details.
- n is the setting bit of shift clock oscillation defined in the “ICCR register (CS2 ~ CS0)” and “MCCR register (CS2 ~ CS0)”. Please refer to the MB90370 series H/W manual for details.
- t_{DOSU} is shown in the interrupt time is longer than the “L” width of SCL.
- SDA and SCL output value is specified on condition that the rise/fall time is “0 ns”.

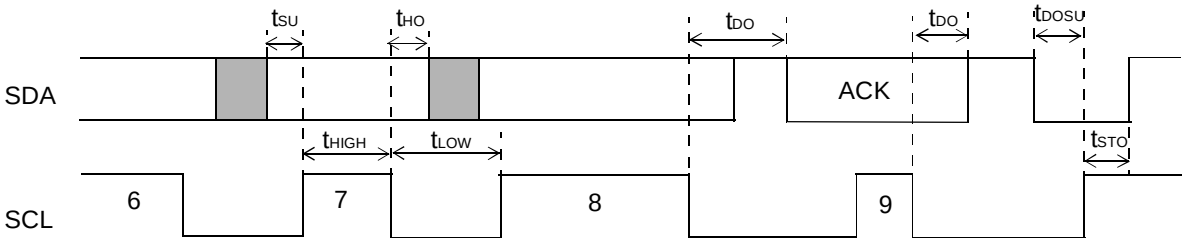
*1: At the stop condition or transferring of next byte.

*2: After setting register bit IBCRH : SCC at restart.

- Data transmit (master / slave)



- Data receive (master / slave)

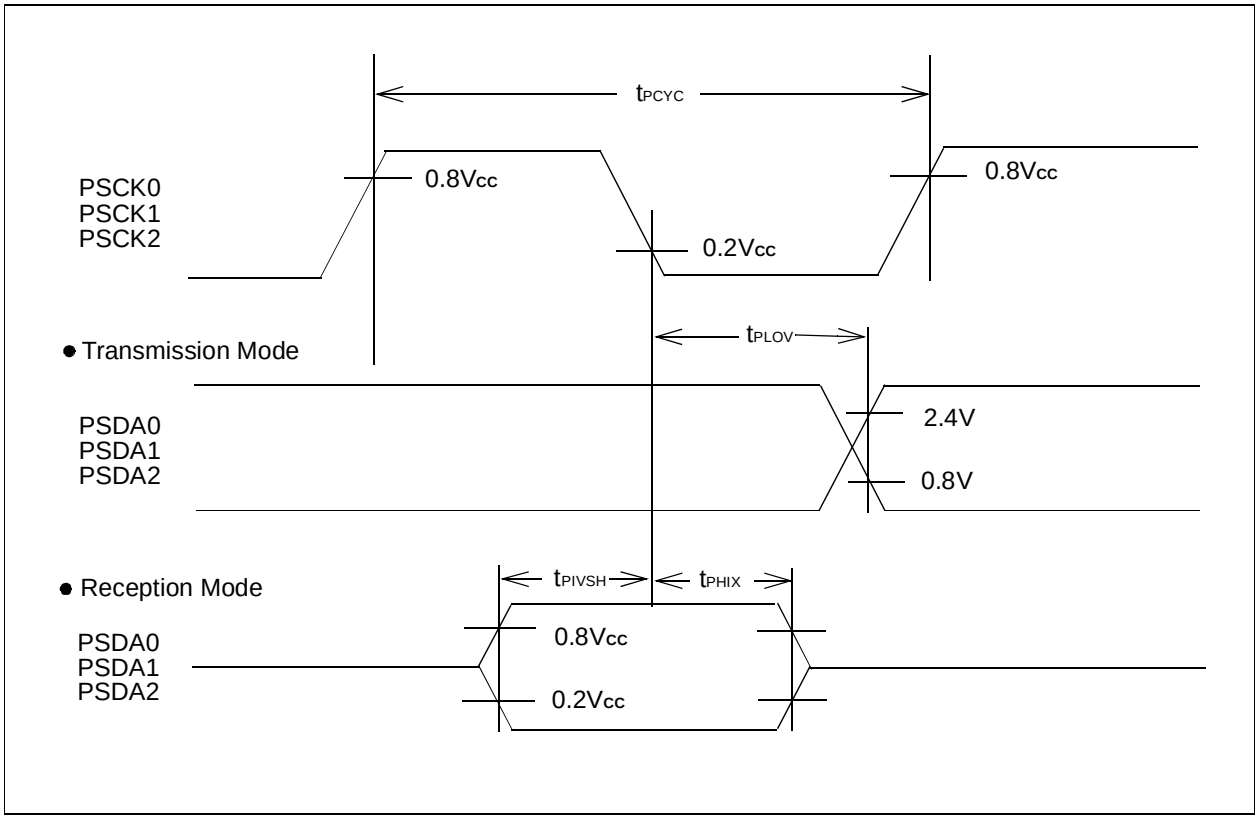


(8) PS/2 Interface Timing

($V_{CC} = AV_{CC} = CV_{CC} = 3.0$ V to 3.6 V, $V_{SS} = 0.0$ V, $T_A = -40$ °C to +85 °C)

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
PSCK clock cycle time	t_{PCYC}	PSCK0 ~ 2 PSDA0 ~ 2	—	4 t_{CP}	—	—	ns	
PSCK↓ → PSDA	t_{PLOV}	PSCK0 ~ 2 PSDA0 ~ 2	Transmission Mode	2 t_{CP}	—	—	ns	
Valid PSDA → PSCK↓	t_{PIVSH}	PSCK0 ~ 2 PSDA0 ~ 2	Reception Mode	1 t_{CP}	—	—	ns	
PSCK↓ → valid PSDA hold time	t_{PHIX}	PSCK0 ~ 2 PSDA0 ~ 2		1 t_{CP}	—	—	ns	
PSCK clock “H” pulse width	t_{PHSL}	PSCK0 ~ 2 PSDA0 ~ 2	—	2 t_{CP}	—	—	ns	
PSCK clock “L” pulse width	t_{PLSH}	PSCK0 ~ 2 PSDA0 ~ 2		2 t_{CP}	—	—	ns	

Note: t_{CP} is the internal operating clock cycle time.

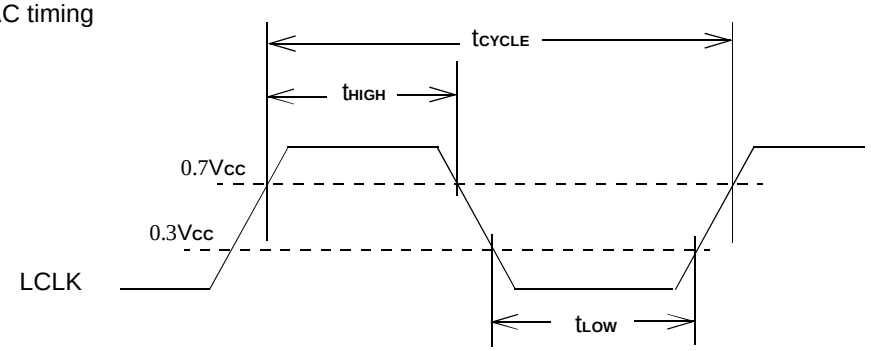


(9) LPC Timing

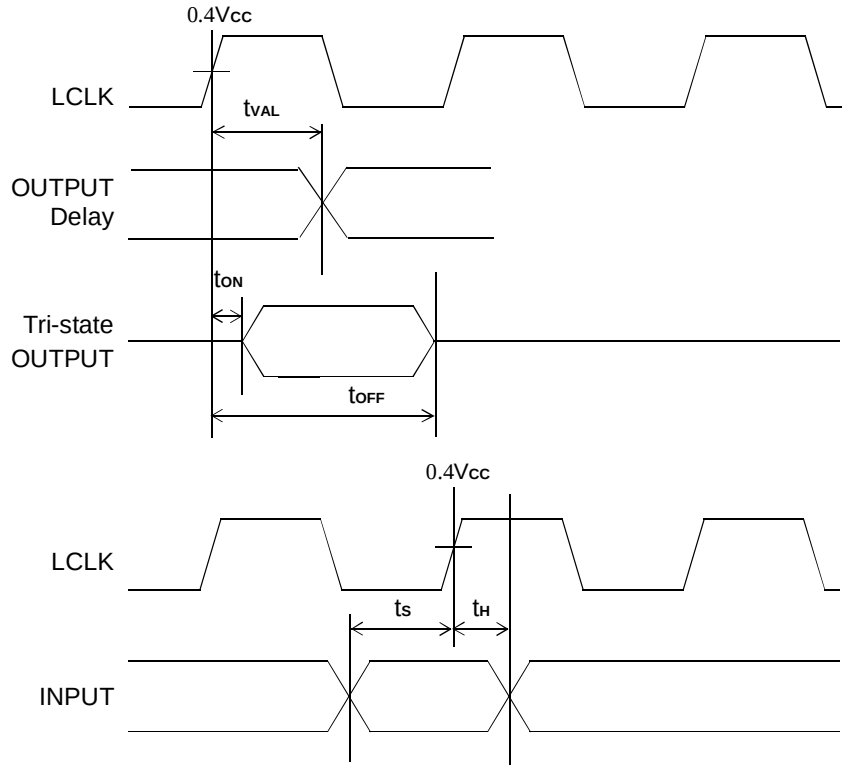
(V_{CC} = AV_{CC} = CV_{CC} = 3.0 V to 3.6 V, V_{SS} = AV_{SS} = CV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
LCLK cycle time	t_{CYCLE}	—	—	30	—	—	ns	
LCLK high time	t_{HIGH}	—	—	12	—	—	ns	
LCLK low time	t_{LOW}	—	—	12	—	—	ns	

LCLK AC timing



LAD, $\overline{\text{LFRAME}}$, GA20 AC timing



5. A/D Converter Electrical Characteristics

(2.7 V ≤ AVR – AV_{SS}, V_{CC} = AV_{CC} = CV_{CC} = 3.0 V to 3.6 V, V_{SS} = AV_{SS} = CV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Value			Unit	Remarks
			Min.	Typ.	Max.		
Resolution	—	—	—	—	10	bit	
Total error	—	—	—	—	±3.0	LSB	
Non-linear error	—	—	—	—	±2.5	LSB	
Differential linearity error	—	—	—	—	±1.9	LSB	
Zero transition voltage	V_{OT}	AN0 ~ AN11	AV_{SS} – 1.5 LSB	AV_{SS} + 0.5 LSB	AV_{SS} + 5.5 LSB	mV	For MB90V370
					AV_{SS} + 2.5 LSB		For MB90F372/372
Full-scale transition voltage	V_{FST}	AN0 ~ AN11	AVR – 3.5 LSB	AVR – 1.5 LSB	AVR + 0.5 LSB	mV	
Conversion time	—	—	3.1	—	—	µs	Actual value is specified as a sum of values specified in ADCR0 : CT1, CT0 and ADCR0 : ST1, ST0. Be sure that the setting value is greater than the min value
Sampling period	—	—	2	—	—	µs	Actual value is specified in ADCR0 : ST1, ST0 bits. Be sure that the setting value is greater than the min value
Analog port input current	I_{AIN}	AN0 ~ AN11	—	0.1	10	µA	
Analog input voltage	V_{AIN}	AN0 ~ AN11	AV_{SS}	—	AVR	V	
Reference voltage	—	AVR	AV_{SS} + 2.7	—	AV_{CC}	V	
Power supply current	I_A	AV_{CC}	—	1.4	6.4	mA	
	I_{AH}		—	—	5	µA	*
Reference voltage supply current	I_R	AVR	—	94	300	µA	
	I_{RH}		—	—	5	µA	*
Offset between channels	—	AN0 ~ AN11	—	—	4	LSB	

*: The current when the A/D converter is not operating or the CPU is in stop mode (for V_{CC} = AV_{CC} = AVR = 3.0 V).

www.DataSheet4U.com

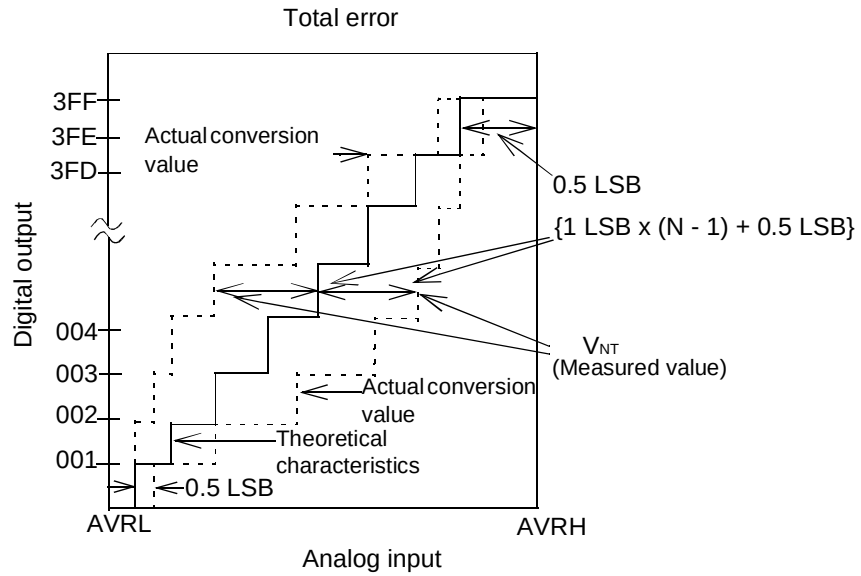
6. A/D Converter Glossary

Resolution : Analog changes that are identifiable with the A/D converter.

Linearity error : The deviation of the straight line connecting the zero transition point (“00 0000 0000” ↔ “00 0000 0001”) with the full-scale transition point (“11 1111 1110” ↔ “11 1111 1111”) from actual conversion characteristics.

Differential linearity error : The deviation of input voltage needed to change the output code by 1 LSB from the theoretical value.

Total error : The total error is defined as a difference between the actual value and the theoretical value, which includes zero-transition error/full-scale transition error and linearity error.



$$\text{Total error for digital output N} = \frac{V_{NT} - \{1\ \text{LSB} \times (N - 1) + 0.5\ \text{LSB}\}}{1\ \text{LSB}}\ [\text{LSB}]$$

$$1\ \text{LSB} = (\text{Theoretical value})\ \frac{\text{AVR} - \text{AVss}}{1024}\ [\text{V}]$$

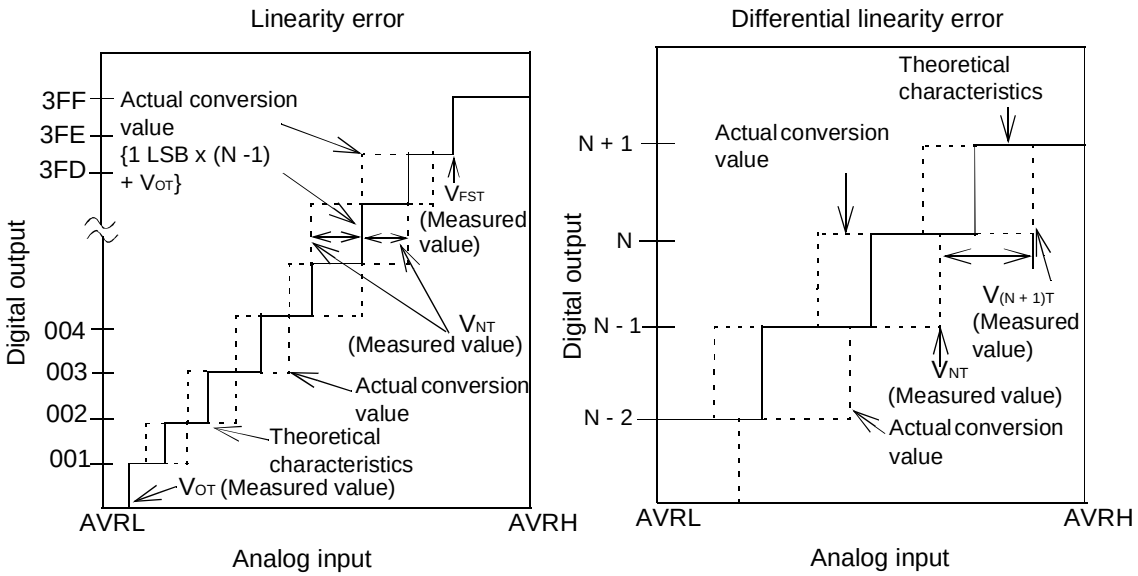
V_{OT} (Theoretical value) = AVss + 0.5 LSB [V]

V_{FST} (Theoretical value) = AVR – 1.5 LSB [V]

V_{NT} : Voltage at a transition of digital output from (N - 1) to N

(Continued)

(Continued)



$$\text{Linearity error of digital output N} = \frac{V_{NT} - \{1\ \text{LSB} \times (N-1) + V_{OT}\}}{1\ \text{LSB}}\ [\text{LSB}]$$

$$\text{Differential linearity error of digital output N} = \frac{V_{(N+1)T} - V_{NT}}{1\ \text{LSB}} - 1\ [\text{LSB}]$$

$$1\ \text{LSB} = \frac{V_{FST} - V_{OT}}{1022}\ [\text{V}]$$

V_{OT} : Voltage at transition of digital output from "000_H" to "001_H"
V_{FST} : Voltage at transition of digital output from "$3FE_H$" to "$3FF_H$"

7. Notes on Using A/D Converter

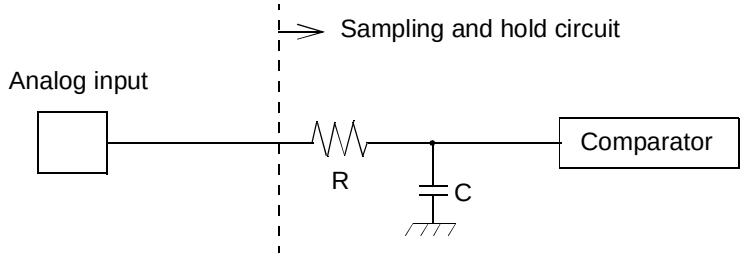
Select the output impedance value for the external circuit of analog input according to the following conditions.

Output impedance values of the external circuit of 4 kΩ or lower are recommended.

When capacitors are connected to external pins, the capacitance of several thousand times the internal capacitor value is recommended to minimized the effect of voltage distribution between the external capacitor and internal capacitor.

When the output impedance of the external circuit is too high, the sampling period for analog voltages may not be sufficient.

● Equipment of analog input circuit model



R : about 1.9 kΩ

C : about 32.3 pF

Note: Listed values must be considered as standards.

• Error

The smaller the | AVR - AV_{SS} |, the greater the error would become relatively.

8. D/A Electrical Characteristics

(V_{CC} = AV_{CC} = CV_{CC} = 3.0 V to 3.6V, V_{SS} = AV_{SS} = CV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
Resolution	—	—	—	—	8	—	bit	
Differential linearity error	—	—		—	—	±0.9	LSB	
Non-linearity error	—	—		—	—	±1.5	LSB	
Conversion time	—	—		—	0.6	—	µs	*
Analog output impedance	—	—		2.0	2.9	3.8	kΩ	
Power supply	I_{DVR}	AV_{CC}		—	—	460	µA	
Current	I_{DVRS}	AV_{CC}		—	0.1	—	µA	D/A stops

* : With load capacitance is 20 pF.

9. Comparator Electrical Characteristics

($V_{CC} = AV_{CC} = CV_{CC} = 3.3$ V to 3.6 V, $V_{SS} = AV_{SS} = CV_{SS} = 0.0$ V, $T_A = -40$ °C to +85 °C)

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
Reference voltage	—	CVRH2	—	1.1	—	2.9	V	
		CVRH1		CVRL	—	2.9	V	
		CVRL		1.1	—	CVRH1	V	
Reference voltage supply current	I_{CR}	CVRH2 CVRH1 CVRL	—	—	—	±1	µA	
Comparator supply current	I_{CV}	CV_{CC}	—	—	—	50	µA	active
				—	—	10	µA	inactive
Analog input voltage	V_{IH}	DCIN DCIN2 VOL1 ~ 3 VSI1 ~ 3	—	CV_{SS}	—	CV_{CC}	V	

10. Serial IRQ Electrical Characteristics

($V_{CC} = AV_{CC} = CV_{CC} = 3.0$ V to 3.6 V, $V_{SS} = AV_{SS} = CV_{SS} = 0.0$ V, $T_A = -40$ °C to +85 °C)

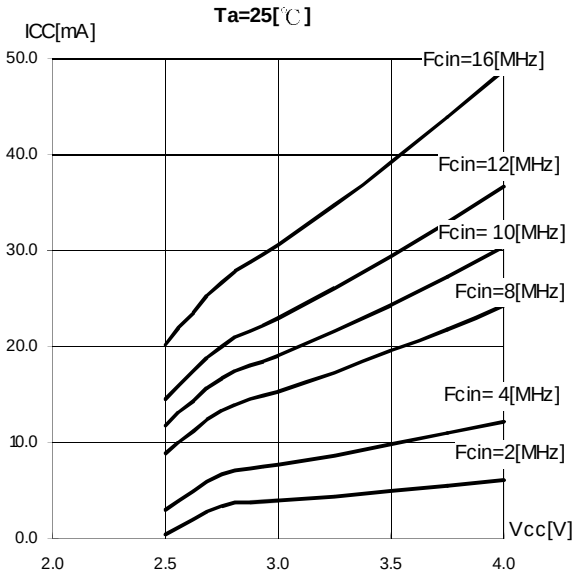
Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
“H” level input voltage	V_{IH}	—	—	$0.7V_{CC}$	—	V_{CC}	V	
“L” level input voltage	V_{IL}	—	—	V_{SS}	—	$0.3V_{CC}$	V	
“H” level output voltage	V_{OH}	—	—	V_{CC} - 0.5	—	—	V	
“L” level output voltage	V_{OL}	—	—	—	—	0.4	V	

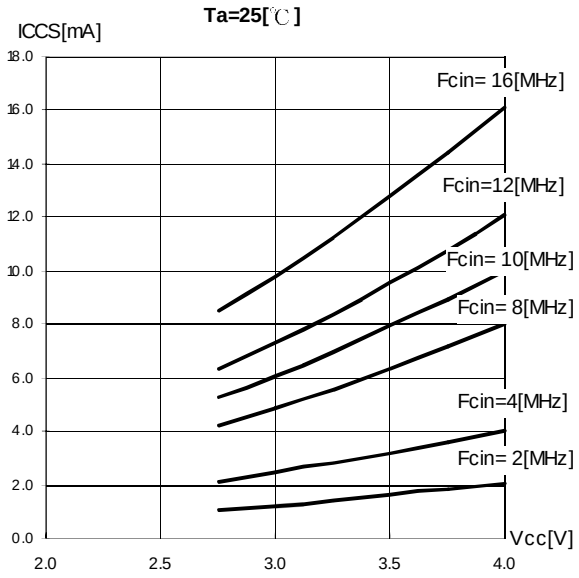
11. Flash Memory Program/Erase Characteristics

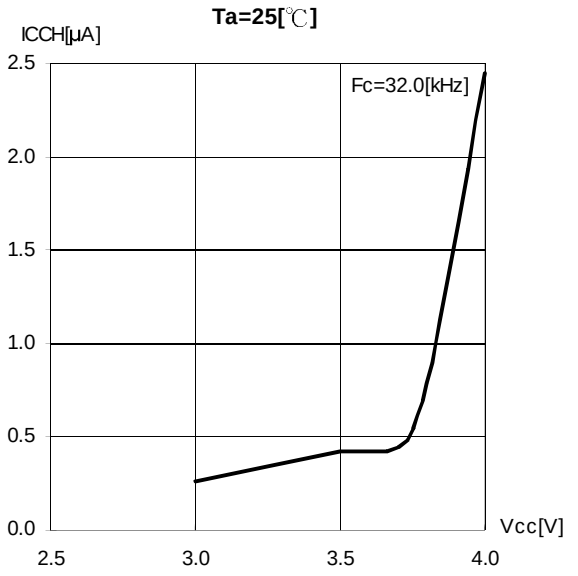
Parameter	Condition	Value			Unit	Remarks
		Min.	Typ.	Max.		
Sector erase time	$T_A = +25$ °C $V_{CC} = 3.0$ V	—	1	15	s	Excludes 00H programming prior to erasure
Chip erase time		—	4	—	s	Excludes 00H programming prior to erasure
Word (16 bit width) programing time		—	16	3,600	µs	Except for the over head time of the system
Program/Erase cycle	—	10,000	—	—	V	

■ EXAMPLE CHARACTERISTICS (MB90F372)

• Power Supply Current

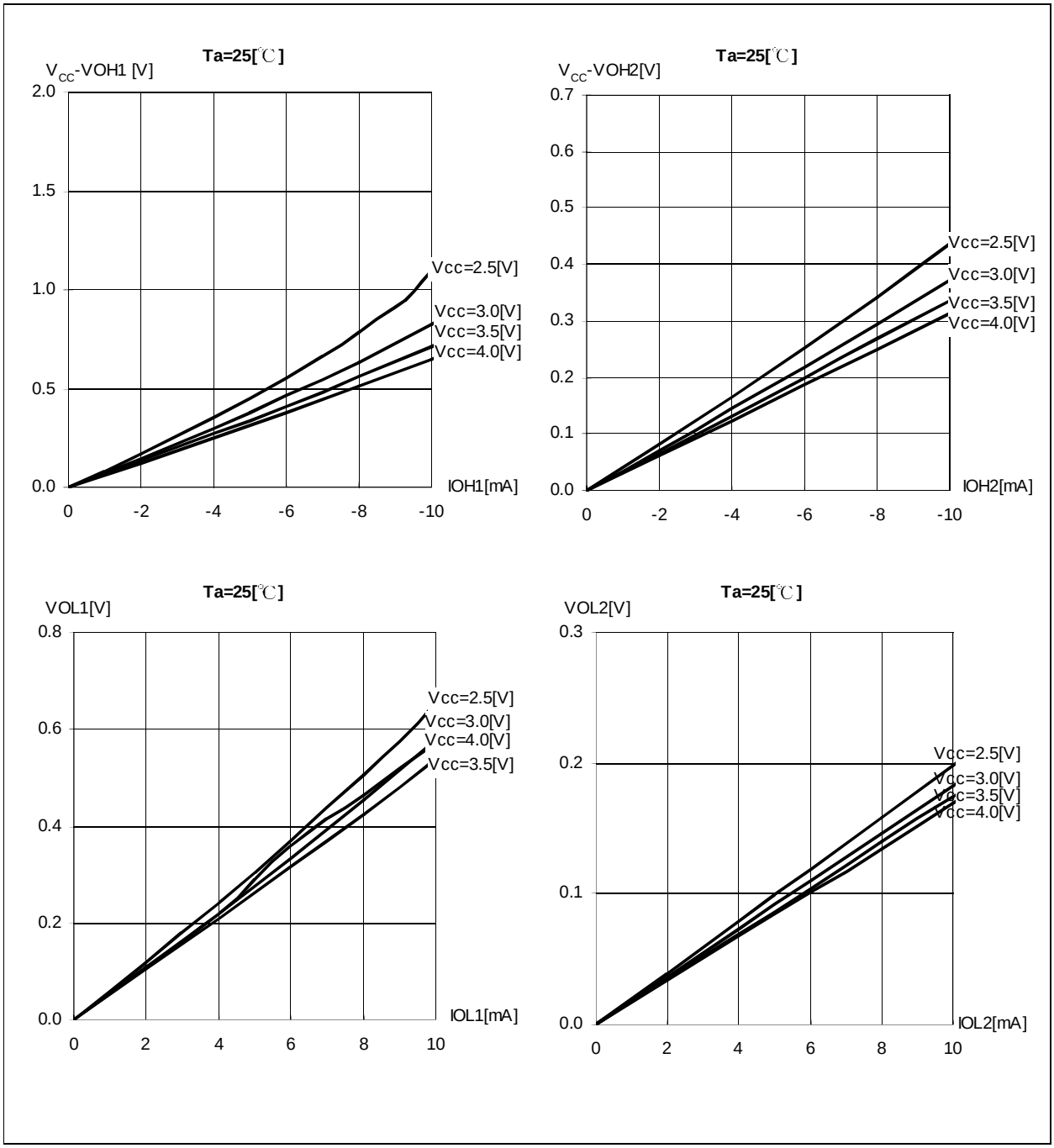






(Continued)

(Continued)


Ta=25[℃]
VCC-VOH1 [V]
2.0
1.5
1.0
0.5
0.0
0
-2
-4
-6
-8
-10
IOH1[mA]
Vcc=2.5[V]
Vcc=3.0[V]
Vcc=3.5[V]
Vcc=4.0[V]
Ta=25[℃]
VCC-VOH2[V]
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
IOH2[mA]
Vcc=2.5[V]
Vcc=3.0[V]
Vcc=3.5[V]
Vcc=4.0[V]
Ta=25[℃]
VOL1[V]
0.8
0.6
0.4
0.2
0.0
0
2
4
6
8
10
IOL1[mA]
Vcc=2.5[V]
Vcc=3.0[V]
Vcc=4.0[V]
Vcc=3.5[V]
Ta=25[℃]
VOL2[V]
0.3
0.2
0.1
0.0
IOL2[mA]
Vcc=2.5[V]
Vcc=3.0[V]
Vcc=3.5[V]
Vcc=4.0[V]

■ INSTRUCTIONS (351 INSTRUCTIONS)

Table 1 Explanation of Items in Tables of Instructions

Item	Meaning
Mnemonic	Upper-case letters and symbols: Represented as they appear in assembler. Lower-case letters: Replaced when described in assembler. Numbers after lower-case letters: Indicate the bit width within the instruction code.
#	Indicates the number of bytes.
~	Indicates the number of cycles. m : When branching n : When not branching See Table 4 for details about meanings of other letters in items.
RG	Indicates the number of accesses to the register during execution of the instruction. It is used calculate a correction value for intermittent operation of CPU.
B	Indicates the correction value for calculating the number of actual cycles during execution of the instruction. (Table 5) The number of actual cycles during execution of the instruction is the correction value summed with the value in the "~" column.
Operation	Indicates the operation of instruction.
LH	Indicates special operations involving the upper 8 bits of the lower 16 bits of the accumulator. Z : Transfers "0". X : Extends with a sign before transferring. – : Transfers nothing.
AH	Indicates special operations involving the upper 16 bits in the accumulator. * : Transfers from AL to AH. – : No transfer. Z : Transfers 00_H to AH. X : Transfers 00_H or FF_H to AH by signing and extending AL.
I S T N Z V C	Indicates the status of each of the following flags: I (interrupt enable), S (stack), T (sticky bit), N (negative), Z (zero), V (overflow), and C (carry). * : Changes due to execution of instruction. – : No change. S : Set by execution of instruction. R : Reset by execution of instruction.
RMW	Indicates whether the instruction is a read-modify-write instruction. (a single instruction that reads data from memory, etc., processes the data, and then writes the result to memory.) * : Instruction is a read-modify-write instruction. – : Instruction is not a read-modify-write instruction. Note: A read-modify-write instruction cannot be used on addresses that have different meanings depending on whether they are read or written.

- **Number of execution cycles**

  The number of cycles required for instruction execution is acquired by adding the number of cycles for each instruction, a corrective value depending on the condition, and the number of cycles required for program fetch. Whenever the instruction being executed exceeds the two-byte (word) boundary, a program on an internal ROM connected to a 16-bit bus is fetched. If data access is interfered with, therefore, the number of execution cycles is increased.

  For each byte of the instruction being executed, a program on a memory connected to an 8-bit external data bus is fetched. If data access in interfered with, therefore, the number of execution cycles is increased. When a general-purpose register, an internal ROM, an internal RAM, an internal I/O device, or an external bus is accessed during intermittent CPU operation, the CPU clock is suspended by the number of cycles specified by the CG1/0 bit of the low-power consumption mode control register. When determining the number of cycles required for instruction execution during intermittent CPU operation, therefore, add the value of the number of times access is done × the number of cycles suspended as the corrective value to the number of ordinary execution cycles.

www.DataSheet4U.com

Table 2 Explanation of Symbols in Tables of Instructions

Symbol	Meaning
A	32-bit accumulator The bit length varies according to the instruction. Byte : Lower 8 bits of AL Word : 16 bits of AL Long : 32 bits of AL and AH
AH AL	Upper 16 bits of A Lower 16 bits of A
SP	Stack pointer (USP or SSP)
PC	Program counter
PCB	Program bank register
DTB	Data bank register
ADB	Additional data bank register
SSB	System stack bank register
USB	User stack bank register
SPB	Current stack bank register (SSB or USB)
DPR	Direct page register
brg1	DTB, ADB, SSB, USB, DPR, PCB, SPB
brg2	DTB, ADB, SSB, USB, DPR, SPB
Ri	R0, R1, R2, R3, R4, R5, R6, R7
RWi	RW0, RW1, RW2, RW3, RW4, RW5, RW6, RW7
RWj	RW0, RW1, RW2, RW3
RLi	RL0, RL1, RL2, RL3
dir	Compact direct addressing
addr16 addr24 ad24 0 to 15 ad24 16 to 23	Direct addressing Physical direct addressing Bit 0 to bit 15 of addr24 Bit 16 to bit 23 of addr24
io	I/O area (000000_H to $0000FF_H$)
imm4 imm8 imm16 imm32 ext (imm8)	4-bit immediate data 8-bit immediate data 16-bit immediate data 32-bit immediate data 16-bit data signed and extended from 8-bit immediate data
disp8 disp16	8-bit displacement 16-bit displacement
bp	Bit offset
vct4 vct8	Vector number (0 to 15) Vector number (0 to 255)
( )b	Bit address
rel	PC relative addressing
ear eam	Effective addressing (codes 00 to 07) Effective addressing (codes 08 to 1F)
rlst	Register list

Table 3 Effective Address Fields

Code	Notation			Address format	Number of bytes in address extension *
00 01 02 03 04 05 06 07	R0 R1 R2 R3 R4 R5 R6 R7	RW0 RW1 RW2 RW3 RW4 RW5 RW6 RW7	RL0 (RL0) RL1 (RL1) RL2 (RL2) RL3 (RL3)	Register direct “ea” corresponds to byte, word, and long-word types, starting from the left	—
08 09 0A 0B	@RW0 @RW1 @RW2 @RW3			Register indirect	0
0C 0D 0E 0F	@RW0 + @RW1 + @RW2 + @RW3 +			Register indirect with post-increment	0
10 11 12 13 14 15 16 17	@RW0 + disp8 @RW1 + disp8 @RW2 + disp8 @RW3 + disp8 @RW4 + disp8 @RW5 + disp8 @RW6 + disp8 @RW7 + disp8			Register indirect with 8-bit displacement	1
18 19 1A 1B	@RW0 + disp16 @RW1 + disp16 @RW2 + disp16 @RW3 + disp16			Register indirect with 16-bit displacement	2
1C 1D 1E 1F	@RW0 + RW7 @RW1 + RW7 @PC + disp16 addr16			Register indirect with index Register indirect with index PC indirect with 16-bit displacement Direct address	0 0 2 2

Note : The number of bytes in the address extension is indicated by the “+” symbol in the “#” (number of bytes) column in the tables of instructions.

Table 4 Number of Execution Cycles for Each Type of Addressing

Code	Operand	(a) Number of execution cycles for each type of addressing	Number of register accesses for each type of addressing
00 to 07	Ri RWi RLi	Listed in tables of instructions	Listed in tables of instructions
08 to 0B	@RWj	2	1
0C to 0F	@RWj +	4	2
10 to 17	@RWi + disp8	2	1
18 to 1B	@RWj + disp16	2	1
1C 1D 1E 1F	@RW0 + RW7 @RW1 + RW7 @PC + disp16 addr16	4 4 2 1	2 2 0 0

Note : “(a)” is used in the “~” (number of states) column and column B (correction value) in the tables of instructions.

Table 5 Compensation Values for Number of Cycles Used to Calculate Number of Actual Cycles

Operand	(b) byte		(c) word		(d) long	
	Cycles	Access	Cycles	Access	Cycles	Access
Internal register	+0	1	+0	1	+0	2
Internal memory even address Internal memory odd address	+0 +0	1 1	+0 +2	1 2	+0 +4	2 4
Even address on external data bus (16 bits) Odd address on external data bus (16 bits)	+1 +1	1 1	+1 +4	1 2	+2 +8	2 4
External data bus (8 bits)	+1	1	+4	2	+8	4

Notes: • “(b)”, “(c)”, and “(d)” are used in the “~” (number of states) column and column B (correction value) in the tables of instructions.
• When the external data bus is used, it is necessary to add in the number of wait cycles used for ready input and automatic ready.

Table 6 Correction Values for Number of Cycles Used to Calculate Number of Program Fetch Cycles

Instruction	Byte boundary	Word boundary
Internal memory	—	+2
External data bus (16 bits)	—	+3
External data bus (8 bits)	+3	—

Notes: • When the external data bus is used, it is necessary to add in the number of wait cycles used for ready input and automatic ready.
• Because instruction execution is not slowed down by all program fetches in actuality, these correction values should be used for “worst case” calculations.

Table 7 Transfer Instructions (Byte) [41 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOV A, dir	2	3	0	(b)	byte (A) ← (dir)	Z	*	–	–	–	*	*	–	–	–
MOV A, addr16	3	4	0	(b)	byte (A) ← (addr16)	Z	*	–	–	–	*	*	–	–	–
MOV A, Ri	1	2	1	0	byte (A) ← (Ri)	Z	*	–	–	–	*	*	–	–	–
MOV A, ear	2	2	1	0	byte (A) ← (ear)	Z	*	–	–	–	*	*	–	–	–
MOV A, eam	2+	3+ (a)	0	(b)	byte (A) ← (eam)	Z	*	–	–	–	*	*	–	–	–
MOV A, io	2	3	0	(b)	byte (A) ← (io)	Z	*	–	–	–	*	*	–	–	–
MOV A, #imm8	2	2	0	0	byte (A) ← imm8	Z	*	–	–	–	*	*	–	–	–
MOV A, @A	2	3	0	(b)	byte (A) ← ((A))	Z	–	–	–	–	*	*	–	–	–
MOV A, @RLi+disp8	3	10	2	(b)	byte (A) ← ((RLi)+disp8)	Z	*	–	–	–	*	*	–	–	–
MOVN A, #imm4	1	1	0	0	byte (A) ← imm4	Z	*	–	–	–	R	*	–	–	–
MOVX A, dir	2	3	0	(b)	byte (A) ← (dir)	X	*	–	–	–	*	*	–	–	–
MOVX A, addr16	3	4	0	(b)	byte (A) ← (addr16)	X	*	–	–	–	*	*	–	–	–
MOVX A, Ri	2	2	1	0	byte (A) ← (Ri)	X	*	–	–	–	*	*	–	–	–
MOVX A, ear	2	2	1	0	byte (A) ← (ear)	X	*	–	–	–	*	*	–	–	–
MOVX A, eam	2+	3+ (a)	0	(b)	byte (A) ← (eam)	X	*	–	–	–	*	*	–	–	–
MOVX A, io	2	3	0	(b)	byte (A) ← (io)	X	*	–	–	–	*	*	–	–	–
MOVX A, #imm8	2	2	0	0	byte (A) ← imm8	X	*	–	–	–	*	*	–	–	–
MOVX A, @A	2	3	0	(b)	byte (A) ← ((A))	X	–	–	–	–	*	*	–	–	–
MOVX A,@RWi+disp8	2	5	1	(b)	byte (A) ←	X	*	–	–	–	*	*	–	–	–
MOVX A, @RLi+disp8	3	10	2	(b)	((RWi)+disp8) byte (A) ← ((RLi)+disp8)	X	*	–	–	–	*	*	–	–	–
MOV dir, A	2	3	0	(b)	byte (dir) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV addr16, A	3	4	0	(b)	byte (addr16) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV Ri, A	1	2	1	0	byte (Ri) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV ear, A	2	2	1	0	byte (ear) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV eam, A	2+	3+ (a)	0	(b)	byte (eam) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV io, A	2	3	0	(b)	byte (io) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV @RLi+disp8, A	3	10	2	(b)	byte ((RLi) +disp8) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV Ri, ear	2	3	2	0	byte (Ri) ← (ear)	–	–	–	–	–	*	*	–	–	–
MOV Ri, eam	2+	4+ (a)	1	(b)	byte (Ri) ← (eam)	–	–	–	–	–	*	*	–	–	–
MOV ear, Ri	2	4	2	0	byte (ear) ← (Ri)	–	–	–	–	–	*	*	–	–	–
MOV eam, Ri	2+	5+ (a)	1	(b)	byte (eam) ← (Ri)	–	–	–	–	–	*	*	–	–	–
MOV Ri, #imm8	2	2	1	0	byte (Ri) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV io, #imm8	3	5	0	(b)	byte (io) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV dir, #imm8	3	5	0	(b)	byte (dir) ← imm8	–	–	–	–	–	*	*	–	–	–
MOV ear, #imm8	3	2	1	0	byte (ear) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV eam, #imm8	3+	4+ (a)	0	(b)	byte (eam) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV @AL, AH /MOV @A, T	2	3	0	(b)	byte ((A)) ← (AH)	–	–	–	–	–	*	*	–	–	–
XCH A, ear	2	4	2	0		Z	–	–	–	–	–	–	–	–	–
XCH A, eam	2+	5+ (a)	0	2× (b)	byte (A) ↔ (ear)	Z	–	–	–	–	–	–	–	–	–
XCH Ri, ear	2	7	4	0	byte (A) ↔ (eam)	–	–	–	–	–	–	–	–	–	–
XCH Ri, eam	2+	9+ (a)	2	2× (b)	byte (Ri) ↔ (ear) byte (Ri) ↔ (eam)	–	–	–	–	–	–	–	–	–	–

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 8 Transfer Instructions (Word/Long Word) [38 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOVW A, dir	2	3	0	(c)	word (A) ← (dir)	–	*	–	–	–	*	*	–	–	–
MOVW A, addr16	3	4	0	(c)	word (A) ← (addr16)	–	*	–	–	–	*	*	–	–	–
MOVW A, SP	1	1	0	0	word (A) ← (SP)	–	*	–	–	–	*	*	–	–	–
MOVW A, RWi	1	2	1	0	word (A) ← (RWi)	–	*	–	–	–	*	*	–	–	–
MOVW A, ear	2	2	1	0	word (A) ← (ear)	–	*	–	–	–	*	*	–	–	–
MOVW A, eam	2+	3+ (a)	0	(c)	word (A) ← (eam)	–	*	–	–	–	*	*	–	–	–
MOVW A, io	2	3	0	(c)	word (A) ← (io)	–	*	–	–	–	*	*	–	–	–
MOVW A, @A	2	3	0	(c)	word (A) ← ((A))	–	–	–	–	–	*	*	–	–	–
MOVW A, #imm16	3	2	0	0	word (A) ← imm16	–	*	–	–	–	*	*	–	–	–
MOVW A, @RWi+disp8	2	5	1	(c)	word (A) ← ((RWi) +disp8)	–	*	–	–	–	*	*	–	–	–
MOVW A, @RLi+disp8	3	10	2	(c)	word (A) ← ((RLi) +disp8)	–	*	–	–	–	*	*	–	–	–
MOVW dir, A	2	3	0	(c)	word (dir) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW addr16, A	3	4	0	(c)	word (addr16) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW SP, A	1	1	0	0	word (SP) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, A	1	2	1	0	word (RWi) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW ear, A	2	2	1	0	word (ear) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW eam, A	2+	3+ (a)	0	(c)	word (eam) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW io, A	2	3	0	(c)	word (io) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW @RWi+disp8, A	2	5	1	(c)	word ((RWi) +disp8) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW @RLi+disp8, A	3	10	2	(c)	word ((RLi) +disp8) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, ear	2	3	2	(0)	word (RWi) ← (ear)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, eam	2+	4+ (a)	1	(c)	word (RWi) ← (eam)	–	–	–	–	–	*	*	–	–	–
MOVW ear, RWi	2	4	2	0	word (ear) ← (RWi)	–	–	–	–	–	*	*	–	–	–
MOVW eam, RWi	2+	5+ (a)	1	(c)	word (eam) ← (RWi)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, #imm16	3	2	1	0	word (RWi) ← imm16	–	–	–	–	–	*	*	–	–	–
MOVW io, #imm16	4	5	0	(c)	word (io) ← imm16	–	–	–	–	–	–	–	–	–	–
MOVW ear, #imm16	4	2	1	0	word (ear) ← imm16	–	–	–	–	–	*	*	–	–	–
MOVW eam, #imm16	4+	4+ (a)	0	(c)	word (eam) ← imm16	–	–	–	–	–	–	–	–	–	–
MOVW @AL, AH /MOVW@A, T	2	3	0	(c)	word ((A)) ← (AH)	–	–	–	–	–	*	*	–	–	–
XCHW A, ear	2	4	2	0	word (A) ↔ (ear)	–	–	–	–	–	–	–	–	–	–
XCHW A, eam	2+	5+ (a)	0	2× (c)	word (A) ↔ (eam)	–	–	–	–	–	–	–	–	–	–
XCHW RWi, ear	2	7	4	0	word (RWi) ↔ (ear)	–	–	–	–	–	–	–	–	–	–
XCHW RWi, eam	2+	9+ (a)	2	2× (c)	word (RWi) ↔ (eam)	–	–	–	–	–	–	–	–	–	–
MOVL A, ear	2	4	2	0	long (A) ← (ear)	–	–	–	–	–	*	*	–	–	–
MOVL A, eam	2+	5+ (a)	0	(d)	long (A) ← (eam)	–	–	–	–	–	*	*	–	–	–
MOVL A, #imm32	5	3	0	0	long (A) ← imm32	–	–	–	–	–	*	*	–	–	–
MOVL ear, A	2	4	2	0	long (ear) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVL eam, A	2+	5+ (a)	0	(d)	long (eam) ← (A)	–	–	–	–	–	*	*	–	–	–

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 9 Addition and Subtraction Instructions (Byte/Word/Long Word) [42 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
ADD A,#imm8	2	2	0	0	byte (A) ← (A) +imm8	Z	–	–	–	–	*	*	*	*	–
ADD A, dir	2	5	0	(b)	byte (A) ← (A) +(dir)	Z	–	–	–	–	*	*	*	*	–
ADD A, ear	2	3	1	0	byte (A) ← (A) +(ear)	Z	–	–	–	–	*	*	*	*	–
ADD A, eam	2+	4+ (a)	0	(b)	byte (A) ← (A) +(eam)	Z	–	–	–	–	*	*	*	*	–
ADD ear, A	2	3	2	0	byte (ear) ← (ear) + (A)	–	–	–	–	–	*	*	*	*	–
ADD eam, A	2+	5+ (a)	0	2× (b)	byte (eam) ← (eam) + (A)	Z	–	–	–	–	*	*	*	*	*
ADDC A	1	2	0	0	byte (A) ← (AH) + (AL) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDC A, ear	2	3	1	0	byte (A) ← (A) + (ear) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDC A, eam	2+	4+ (a)	0	(b)	byte (A) ← (A) + (eam) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDDC A	1	3	0	0	byte (A) ← (AH) + (AL) + (C) (decimal)	Z	–	–	–	–	*	*	*	*	–
SUB A, #imm8	2	2	0	0	byte (A) ← (A) –imm8	Z	–	–	–	–	*	*	*	*	–
SUB A, dir	2	5	0	(b)	byte (A) ← (A) – (dir)	Z	–	–	–	–	*	*	*	*	–
SUB A, ear	2	3	1	0	byte (A) ← (A) – (ear)	Z	–	–	–	–	*	*	*	*	–
SUB A, eam	2+	4+ (a)	0	(b)	byte (A) ← (A) – (eam)	Z	–	–	–	–	*	*	*	*	–
SUB ear, A	2	3	2	0	byte (ear) ← (ear) – (A)	–	–	–	–	–	*	*	*	*	–
SUB eam, A	2+	5+ (a)	0	2× (b)	byte (eam) ← (eam) – (A)	–	–	–	–	–	*	*	*	*	*
SUBC A	1	2	0	0	byte (A) ← (AH) – (AL) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBC A, ear	2	3	1	0	byte (A) ← (A) – (ear) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBC A, eam	2+	4+ (a)	0	(b)	byte (A) ← (A) – (eam) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBDC A	1	3	0	0	byte (A) ← (AH) – (AL) – (C) (decimal)	Z	–	–	–	–	*	*	*	*	–
ADDW A	1	2	0	0	word (A) ← (AH) + (AL)	–	–	–	–	–	*	*	*	*	–
ADDW A, ear	2	3	1	0	word (A) ← (A) +(ear)	–	–	–	–	–	*	*	*	*	–
ADDW A, eam	2+	4+ (a)	0	(c)	word (A) ← (A) +(eam)	–	–	–	–	–	*	*	*	*	–
ADDW A, #imm16	3	2	0	0	word (A) ← (A) +imm16	–	–	–	–	–	*	*	*	*	–
ADDW ear, A	2	3	2	0	word (ear) ← (ear) + (A)	–	–	–	–	–	*	*	*	*	–
ADDW eam, A	2+	5+ (a)	0	2× (c)	word (eam) ← (eam) + (A)	–	–	–	–	–	*	*	*	*	*
ADDCW A, ear	2	3	1	0	word (A) ← (A) + (ear) + (C)	–	–	–	–	–	*	*	*	*	–
ADDCW A, eam	2+	4+ (a)	0	(c)	word (A) ← (A) + (eam) +	–	–	–	–	–	*	*	*	*	–
SUBW A	1	2	0	0	(C)	–	–	–	–	–	*	*	*	*	–
SUBW A, ear	2	3	1	0	word (A) ← (AH) – (AL)	–	–	–	–	–	*	*	*	*	–
SUBW A, eam	2+	4+ (a)	0	(c)	word (A) ← (A) – (ear)	–	–	–	–	–	*	*	*	*	–
SUBW A, #imm16	3	2	0	0	word (A) ← (A) – (eam)	–	–	–	–	–	*	*	*	*	–
SUBW ear, A	2	3	2	0	word (A) ← (A) –imm16	–	–	–	–	–	*	*	*	*	–
SUBW eam, A	2+	5+ (a)	0	2× (c)	word (ear) ← (ear) – (A)	–	–	–	–	–	*	*	*	*	*
SUBCW A, ear	2	3	1	0	word (eam) ← (eam) – (A)	–	–	–	–	–	*	*	*	*	–
SUBCW A, eam	2+	4+ (a)	0	(c)	word (A) ← (A) – (ear) – (C) word (A) ← (A) – (eam) – (C)	–	–	–	–	–	*	*	*	*	–
ADDL A, ear	2	6	2	0	long (A) ← (A) + (ear)	–	–	–	–	–	*	*	*	*	–
ADDL A, eam	2+	7+ (a)	0	(d)	long (A) ← (A) + (eam)	–	–	–	–	–	*	*	*	*	–
ADDL A, #imm32	5	4	0	0	long (A) ← (A) +imm32	–	–	–	–	–	*	*	*	*	–
SUBL A, ear	2	6	2	0	long (A) ← (A) – (ear)	–	–	–	–	–	*	*	*	*	–
SUBL A, eam	2+	7+ (a)	0	(d)	long (A) ← (A) – (eam)	–	–	–	–	–	*	*	*	*	–
SUBL A, #imm32	5	4	0	0	long (A) ← (A) –imm32	–	–	–	–	–	*	*	*	*	–

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 10 Increment and Decrement Instructions (Byte/Word/Long Word) [12 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
INC ear	2	2	2	0	byte (ear) ← (ear) +1	–	–	–	–	–	*	*	*	–	–
INC eam	2+	5+ (a)	0	2× (b)	byte (eam) ← (eam) +1	–	–	–	–	–	*	*	*	–	*
DEC ear	2	3	2	0	byte (ear) ← (ear) –1	–	–	–	–	–	*	*	*	–	–
DEC eam	2+	5+ (a)	0	2× (b)	byte (eam) ← (eam) –1	–	–	–	–	–	*	*	*	–	*
INCW ear	2	3	2	0	word (ear) ← (ear) +1	–	–	–	–	–	*	*	*	–	–
INCW eam	2+	5+ (a)	0	2× (c)	word (eam) ← (eam) +1	–	–	–	–	–	*	*	*	–	*
DECW ear	2	3	2	0	word (ear) ← (ear) –1	–	–	–	–	–	*	*	*	–	–
DECW eam	2+	5+ (a)	0	2× (c)	word (eam) ← (eam) –1	–	–	–	–	–	*	*	*	–	*
INCL ear	2	7	4	0	long (ear) ← (ear) +1	–	–	–	–	–	*	*	*	–	–
INCL eam	2+	9+ (a)	0	2× (d)	long (eam) ← (eam) +1	–	–	–	–	–	*	*	*	–	*
DECL ear	2	7	4	0	long (ear) ← (ear) –1	–	–	–	–	–	*	*	*	–	–
DECL eam	2+	9+ (a)	0	2× (d)	long (eam) ← (eam) –1	–	–	–	–	–	*	*	*	–	*

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 5, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 11 Compare Instructions (Byte/Word/Long Word) [11 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
CMP A	1	1	0	0	byte (AH) – (AL)	–	–	–	–	–	*	*	*	*	–
CMP A, ear	2	2	1	0	byte (A) ← (ear)	–	–	–	–	–	*	*	*	*	–
CMP A, eam	2+	3+ (a)	0	(b)	byte (A) ← (eam)	–	–	–	–	–	*	*	*	*	–
CMP A, #imm8	2	2	0	0	byte (A) ← imm8	–	–	–	–	–	*	*	*	*	–
CMPW A	1	1	0	0	word (AH) – (AL)	–	–	–	–	–	*	*	*	*	–
CMPW A, ear	2	2	1	0	word (A) ← (ear)	–	–	–	–	–	*	*	*	*	–
CMPW A, eam	2+	3+ (a)	0	(c)	word (A) ← (eam)	–	–	–	–	–	*	*	*	*	–
CMPW A, #imm16	3	2	0	0	word (A) ← imm16	–	–	–	–	–	*	*	*	*	–
CMPL A, ear	2	6	2	0	word (A) ← (ear)	–	–	–	–	–	*	*	*	*	–
CMPL A, eam	2+	7+ (a)	0	(d)	word (A) ← (eam)	–	–	–	–	–	*	*	*	*	–
CMPL A, #imm32	5	3	0	0	word (A) ← imm32	–	–	–	–	–	*	*	*	*	–

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 12 Multiplication and Division Instructions (Byte/Word/Long Word) [11 Instructions]

Mnemonic		#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
DIVU	A	1	*1	0	0	word (AH) /byte (AL) Quotient → byte (AL) Remainder → byte (AH)	–	–	–	–	–	–	–	*	*	–
DIVU	A, ear	2	*2	1	0	word (A)/byte (ear) Quotient → byte (A) Remainder → byte (ear)	–	–	–	–	–	–	–	*	*	–
DIVU	A, eam	2+	*3	0	*6	word (A)/byte (eam) Quotient → byte (A) Remainder → byte (eam)	–	–	–	–	–	–	–	*	*	–
DIVUW	A, ear	2	*4	1	0	long (A)/word (ear) Quotient → word (A) Remainder → word (ear)	–	–	–	–	–	–	–	*	*	–
DIVUW	A, eam	2+	*5	0	*7	long (A)/word (eam) Quotient → word (A) Remainder → word (ear)	–	–	–	–	–	–	–	*	*	–
MULU	A	1	*8	0	0	byte (AH) *byte (AL) → word (A)	–	–	–	–	–	–	–	–	–	–
MULU	A, ear	2	*9	1	0	byte (A) *byte (ear) → word (A)	–	–	–	–	–	–	–	–	–	–
MULU	A, eam	2+	*10	0	(b)	byte (A) *byte (eam) → word (A)	–	–	–	–	–	–	–	–	–	–
MULUW	A	1	*11	0	0	word (AH) *word (AL) → long (A)	–	–	–	–	–	–	–	–	–	–
MULUW	A, ear	2	*12	1	0	word (A) *word (ear) → long (A)	–	–	–	–	–	–	–	–	–	–
MULUW	A, eam	2+	*13	0	(c)	word (A) *word (eam) → long (A)	–	–	–	–	–	–	–	–	–	–

*1: 3 when the result is zero, 7 when an overflow occurs, and 15 normally.
*2: 4 when the result is zero, 8 when an overflow occurs, and 16 normally.
*3: 6 + (a) when the result is zero, 9 + (a) when an overflow occurs, and 19 + (a) normally.
*4: 4 when the result is zero, 7 when an overflow occurs, and 22 normally.
*5: 6 + (a) when the result is zero, 8 + (a) when an overflow occurs, and 26 + (a) normally.
*6: (b) when the result is zero or when an overflow occurs, and 2 × (b) normally.
*7: (c) when the result is zero or when an overflow occurs, and 2 × (c) normally.
*8: 3 when byte (AH) is zero, and 7 when byte (AH) is not zero.
*9: 4 when byte (ear) is zero, and 8 when byte (ear) is not zero.
*10: 5 + (a) when byte (eam) is zero, and 9 + (a) when byte (eam) is not 0.
*11: 3 when word (AH) is zero, and 11 when word (AH) is not zero.
*12: 4 when word (ear) is zero, and 12 when word (ear) is not zero.
*13: 5 + (a) when word (eam) is zero, and 13 + (a) when word (eam) is not zero.

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 13 Signed Multiplication and Division Instructions (Byte/Word/Long Word) [11 Instructions]

Mnemonic		#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
DIV	A	2	*1	0	0	word (AH) /byte (AL) Quotient → byte (AL) Remainder → byte (AH)	Z	–	–	–	–	–	–	*	*	–
DIV	A, ear	2	*2	1	0	word (A)/byte (ear) Quotient → byte (A) Remainder → byte (ear)	Z	–	–	–	–	–	–	*	*	–
DIV	A, eam	2 +	*3	0	*6	word (A)/byte (eam) Quotient → byte (A) Remainder → byte (eam)	Z	–	–	–	–	–	–	*	*	–
DIVW	A, ear	2	*4	1	0	long (A)/word (ear) Quotient → word (A) Remainder → word (ear)	–	–	–	–	–	–	–	*	*	–
DIVW	A, eam	2+	*5	0	*7	long (A)/word (eam) Quotient → word (A) Remainder → word (eam)	–	–	–	–	–	–	–	*	*	–
MULU	A	2	*8	0	0	byte (AH) *byte (AL) → word (A)	–	–	–	–	–	–	–	–	–	–
MULU	A, ear	2	*9	1	0	byte (A) *byte (ear) → word (A)	–	–	–	–	–	–	–	–	–	–
MULU	A, eam	2 +	*10	0	(b)	byte (A) *byte (eam) → word (A)	–	–	–	–	–	–	–	–	–	–
MULUW	A	2	*11	0	0	word (AH) *word (AL) → long (A)	–	–	–	–	–	–	–	–	–	–
MULUW	A, ear	2	*12	1	0	word (A) *word (ear) → long (A)	–	–	–	–	–	–	–	–	–	–
MULUW	A, eam	2 +	*13	0	(c)	word (A) *word (eam) → long (A)	–	–	–	–	–	–	–	–	–	–

*1: Set to 3 when the division-by-0, 8 or 18 for an overflow, and 18 for normal operation.
*2: Set to 3 when the division-by-0, 10 or 21 for an overflow, and 22 for normal operation.
*3: Set to 4 + (a) when the division-by-0, 11 + (a) or 22 + (a) for an overflow, and 23 + (a) for normal operation.
*4: Positive dividend: Set to 4 when the division-by-0, 10 or 29 for an overflow, and 30 for normal operation.
Negative dividend: Set to 4 when the division-by-0, 11 or 30 for an overflow and 31 for normal operation.
*5: Positive dividend: Set to 4 + (a) when the division-by-0, 11 + (a) or 30 + (a) for an overflow, and 31 + (a) for normal operation.
Negative dividend: Set to 4 + (a) when the division-by-0, 12 + (a) or 31 + (a) for an overflow, and 32 + (a) for normal operation.
*6: When the division-by-0, (b) for an overflow, and 2 × (b) for normal operation.
*7: When the division-by-0, (c) for an overflow, and 2 × (c) for normal operation.
*8: Set to 3 when byte (AH) is zero, 12 when the result is positive, and 13 when the result is negative.
*9: Set to 3 when byte (ear) is zero, 12 when the result is positive, and 13 when the result is negative.
*10: Set to 4 + (a) when byte (eam) is zero, 13 + (a) when the result is positive, and 14 + (a) when the result is negative.
*11: Set to 3 when word (AH) is zero, 12 when the result is positive, and 13 when the result is negative.
*12: Set to 3 when word (ear) is zero, 16 when the result is positive, and 19 when the result is negative.
*13: Set to 4 + (a) when word (eam) is zero, 17 + (a) when the result is positive, and 20 + (a) when the result is negative.

Notes:
- When overflow occurs during DIV or DIVW instruction execution, the number of execution cycles takes two values because of detection before and after an operation.
- When overflow occurs during DIV or DIVW instruction execution, the contents of AL are destroyed.
- For (a) to (d), refer to “Table 4 Number of Execution Cycles for Effective Address in Addressing Modes” and “Table 6 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

Table 14 Logical 1 Instructions (Byte/Word) [39 Instructions]

Mnemonic		#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
AND	A, #imm8	2	2	0	0	byte (A) ← (A) and imm8	–	–	–	–	–	*	*	R	–	–
AND	A, ear	2	3	1	0	byte (A) ← (A) and (ear)	–	–	–	–	–	*	*	R	–	–
AND	A, eam	2+	4+ (a)	0	(b)	byte (A) ← (A) and (eam)	–	–	–	–	–	*	*	R	–	–
AND	ear, A	2	3	2	0	byte (ear) ← (ear) and (A)	–	–	–	–	–	*	*	R	–	–
AND	eam, A	2+	5+ (a)	0	2× (b)	byte (eam) ← (eam) and (A)	–	–	–	–	–	*	*	R	–	*
OR	A, #imm8	2	2	0	0	byte (A) ← (A) or imm8	–	–	–	–	–	*	*	R	–	–
OR	A, ear	2	3	1	0	byte (A) ← (A) or (ear)	–	–	–	–	–	*	*	R	–	–
OR	A, eam	2+	4+ (a)	0	(b)	byte (A) ← (A) or (eam)	–	–	–	–	–	*	*	R	–	–
OR	ear, A	2	3	2	0	byte (ear) ← (ear) or (A)	–	–	–	–	–	*	*	R	–	–
OR	eam, A	2+	5+ (a)	0	2× (b)	byte (eam) ← (eam) or (A)	–	–	–	–	–	*	*	R	–	*
XOR	A, #imm8	2	2	0	0	byte (A) ← (A) xor imm8	–	–	–	–	–	*	*	R	–	–
XOR	A, ear	2	3	1	0	byte (A) ← (A) xor (ear)	–	–	–	–	–	*	*	R	–	–
XOR	A, eam	2+	4+ (a)	0	(b)	byte (A) ← (A) xor (eam)	–	–	–	–	–	*	*	R	–	–
XOR	ear, A	2	3	2	0	byte (ear) ← (ear) xor (A)	–	–	–	–	–	*	*	R	–	–
XOR	eam, A	2+	5+ (a)	0	2× (b)	byte (eam) ← (eam) xor (A)	–	–	–	–	–	*	*	R	–	*
NOT	A	1	2	0	0	byte (A) ← not (A)	–	–	–	–	–	*	*	R	–	–
NOT	ear	2	3	2	0	byte (ear) ← not (ear)	–	–	–	–	–	*	*	R	–	–
NOT	eam	2+	5+ (a)	0	2× (b)	byte (eam) ← not (eam)	–	–	–	–	–	*	*	R	–	*
ANDW	A	1	2	0	0	word (A) ← (AH) and (A)	–	–	–	–	–	*	*	R	–	–
ANDW	A, #imm16	3	2	0	0	word (A) ← (A) and imm16	–	–	–	–	–	*	*	R	–	–
ANDW	A, ear	2	3	1	0	word (A) ← (A) and (ear)	–	–	–	–	–	*	*	R	–	–
ANDW	A, eam	2+	4+ (a)	0	(c)	word (A) ← (A) and (eam)	–	–	–	–	–	*	*	R	–	–
ANDW	ear, A	2	3	2	0	word (ear) ← (ear) and (A)	–	–	–	–	–	*	*	R	–	–
ANDW	eam, A	2+	5+ (a)	0	2× (c)	word (eam) ← (eam) and (A)	–	–	–	–	–	*	*	R	–	*
ORW	A	1	2	0	0	word (A) ← (AH) or (A)	–	–	–	–	–	*	*	R	–	–
ORW	A, #imm16	3	2	0	0	word (A) ← (A) or imm16	–	–	–	–	–	*	*	R	–	–
ORW	A, ear	2	3	1	0	word (A) ← (A) or (ear)	–	–	–	–	–	*	*	R	–	–
ORW	A, eam	2+	4+ (a)	0	(c)	word (A) ← (A) or (eam)	–	–	–	–	–	*	*	R	–	–
ORW	ear, A	2	3	2	0	word (ear) ← (ear) or (A)	–	–	–	–	–	*	*	R	–	–
ORW	eam, A	2+	5+ (a)	0	2× (c)	word (eam) ← (eam) or (A)	–	–	–	–	–	*	*	R	–	*
XORW	A	1	2	0	0	word (A) ← (AH) xor (A)	–	–	–	–	–	*	*	R	–	–
XORW	A, #imm16	3	2	0	0	word (A) ← (A) xor imm16	–	–	–	–	–	*	*	R	–	–
XORW	A, ear	2	3	1	0	word (A) ← (A) xor (ear)	–	–	–	–	–	*	*	R	–	–
XORW	A, eam	2+	4+ (a)	0	(c)	word (A) ← (A) xor (eam)	–	–	–	–	–	*	*	R	–	–
XORW	ear, A	2	3	2	0	word (ear) ← (ear) xor (A)	–	–	–	–	–	*	*	R	–	–
XORW	eam, A	2+	5+ (a)	0	2× (c)	word (eam) ← (eam) xor (A)	–	–	–	–	–	*	*	R	–	*
NOTW	A	1	2	0	0	word (A) ← not (A)	–	–	–	–	–	*	*	R	–	–
NOTW	ear	2	3	2	0	word (ear) ← not (ear)	–	–	–	–	–	*	*	R	–	–
NOTW	eam	2+	5+ (a)	0	2× (c)	word (eam) ← not (eam)	–	–	–	–	–	*	*	R	–	*

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 15 Logical 2 Instructions (Long Word) [6 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
ANDL A, ear	2	6	2	0	long (A) ← (A) and (ear)	–	–	–	–	–	*	*	R	–	–
ANDL A, eam	2+	7+ (a)	0	(d)	long (A) ← (A) and (eam)	–	–	–	–	–	*	*	R	–	–
ORL A, ear	2	6	2	0	long (A) ← (A) or (ear)	–	–	–	–	–	*	*	R	–	–
ORL A, eam	2+	7+ (a)	0	(d)	long (A) ← (A) or (eam)	–	–	–	–	–	*	*	R	–	–
XORL A, ea	2	6	2	0	long (A) ← (A) xor (ear)	–	–	–	–	–	*	*	R	–	–
XORL A, eam	2+	7+ (a)	0	(d)	long (A) ← (A) xor (eam)	–	–	–	–	–	*	*	R	–	–

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 16 Sign Inversion Instructions (Byte/Word) [6 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
NEG A	1	2	0	0	byte (A) ← 0 – (A)	X	–	–	–	–	*	*	*	*	–
NEG ear	2	3	2	0	byte (ear) ← 0 – (ear)	–	–	–	–	–	*	*	*	*	–
NEG eam	2+	5+ (a)	0	2× (b)	byte (eam) ← 0 – (eam)	–	–	–	–	–	*	*	*	*	*
NEGW A	1	2	0	0	word (A) ← 0 – (A)	–	–	–	–	–	*	*	*	*	–
NEGW ear	2	3	2	0	word (ear) ← 0 – (ear)	–	–	–	–	–	*	*	*	*	–
NEGW eam	2+	5+ (a)	0	2× (c)	word (eam) ← 0 – (eam)	–	–	–	–	–	*	*	*	*	*

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 17 Normalize Instruction (Long Word) [1 Instruction]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
NRML A, R0	2	*1	1	0	long (A) ← Shift until first digit is “1” byte (R0) ← Current shift count	–	–	–	–	–	–	*	–	–	–

*1: 4 when the contents of the accumulator are all zeroes, 6 + (R0) in all other cases (shift count).

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 18 Shift Instructions (Byte/Word/Long Word) [18 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
RORC A	2	2	0	0	byte (A) ← Right rotation with carry	–	–	–	–	–	*	*	–	*	–
ROLC A	2	2	0	0	byte (A) ← Left rotation with carry	–	–	–	–	–	*	*	–	*	–
RORC ear	2	3	2	0	byte (ear) ← Right rotation with carry	–	–	–	–	–	*	*	–	*	–
RORC eam	2+	5+ (a)	0	2× (b)	byte (eam) ← Right rotation with carry	–	–	–	–	–	*	*	–	*	*
ROLC ear	2	3	2	0	byte (ear) ← Left rotation with carry	–	–	–	–	–	*	*	–	*	–
ROLC eam	2+	5+ (a)	0	2× (b)	byte (eam) ← Left rotation with carry	–	–	–	–	–	*	*	–	*	*
ASR A, R0	2	*1	1	0	byte (A) ← Arithmetic right barrel shift (A, R0)	–	–	–	–	*	*	*	–	*	–
LSR A, R0	2	*1	1	0	byte (A) ← Logical right barrel shift (A, R0)	–	–	–	–	*	*	*	–	*	–
LSL A, R0	2	*1	1	0	byte (A) ← Logical left barrel shift (A, R0)	–	–	–	–	–	*	*	–	*	–
ASRW A	1	2	0	0	word (A) ←Arithmetic right shift (A, 1 bit)	–	–	–	–	*	*	*	–	*	–
LSRW A/SHRW A	1	2	0	0	word (A) ← Logical right shift (A, 1 bit)	–	–	–	–	*	R	*	–	*	–
LSLW A/SHLW A	1	2	0	0	word (A) ← Logical left shift (A, 1 bit)	–	–	–	–	–	*	*	–	*	–
ASRW A, R0	2	*1	1	0	word (A) ← Arithmetic right barrel shift (A,	–	–	–	–	*	*	*	–	*	–
LSRW A, R0	2	*1	1	0	R0)	–	–	–	–	*	*	*	–	*	–
LSLW A, R0	2	*1	1	0	word (A) ← Logical right barrel shift (A, R0) word (A) ← Logical left barrel shift (A, R0)	–	–	–	–	–	*	*	–	*	–
ASRL A, R0	2	*2	1	0	long (A) ← Arithmetic right shift (A, R0)	–	–	–	–	*	*	*	–	*	–
LSRL A, R0	2	*2	1	0	long (A) ← Logical right barrel shift (A, R0)	–	–	–	–	*	*	*	–	*	–
LSLL A, R0	2	*2	1	0	long (A) ← Logical left barrel shift (A, R0)	–	–	–	–	–	*	*	–	*	–

*1: 6 when R0 is 0, 5 + (R0) in all other cases.

*2: 6 when R0 is 0, 6 + (R0) in all other cases.

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 19 Branch 1 Instructions [31 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
BZ/BEQ rel	2	*1	0	0	Branch when (Z) = 1	–	–	–	–	–	–	–	–	–	–
BNZ/BNE rel	2	*1	0	0	Branch when (Z) = 0	–	–	–	–	–	–	–	–	–	–
BC/BLO rel	2	*1	0	0	Branch when (C) = 1	–	–	–	–	–	–	–	–	–	–
BNC/BHS rel	2	*1	0	0	Branch when (C) = 0	–	–	–	–	–	–	–	–	–	–
BN rel	2	*1	0	0	Branch when (N) = 1	–	–	–	–	–	–	–	–	–	–
BP rel	2	*1	0	0	Branch when (N) = 0	–	–	–	–	–	–	–	–	–	–
BV rel	2	*1	0	0	Branch when (V) = 1	–	–	–	–	–	–	–	–	–	–
BNV rel	2	*1	0	0	Branch when (V) = 0	–	–	–	–	–	–	–	–	–	–
BT rel	2	*1	0	0	Branch when (T) = 1	–	–	–	–	–	–	–	–	–	–
BNT rel	2	*1	0	0	Branch when (T) = 0	–	–	–	–	–	–	–	–	–	–
BLT rel	2	*1	0	0	Branch when (V) xor (N) = 1	–	–	–	–	–	–	–	–	–	–
BGE rel	2	*1	0	0	Branch when (V) xor (N) = 0	–	–	–	–	–	–	–	–	–	–
BLE rel	2	*1	0	0	Branch when ((V) xor (N)) or (Z) = 1	–	–	–	–	–	–	–	–	–	–
BGT rel	2	*1	0	0	Branch when ((V) xor (N)) or (Z) = 0	–	–	–	–	–	–	–	–	–	–
BLS rel	2	*1	0	0	Branch when (C) or (Z) = 1	–	–	–	–	–	–	–	–	–	–
BHI rel	2	*1	0	0	Branch when (C) or (Z) = 0	–	–	–	–	–	–	–	–	–	–
BRA rel	2	*1	0	0	Branch unconditionally	–	–	–	–	–	–	–	–	–	–
JMP @A	1	2	0	0	word (PC) ← (A)	–	–	–	–	–	–	–	–	–	–
JMP addr16	3	3	0	0	word (PC) ← addr16	–	–	–	–	–	–	–	–	–	–
JMP @ear	2	3	1	0	word (PC) ← (ear)	–	–	–	–	–	–	–	–	–	–
JMP @eam	2+	4+ (a)	0	(c)	word (PC) ← (eam)	–	–	–	–	–	–	–	–	–	–
JMPP @ear *3	2	5	2	0	word (PC) ← (ear), (PCB) ← (ear +2)	–	–	–	–	–	–	–	–	–	–
JMPP @eam *3	2+	6+ (a)	0	(d)	word (PC) ← (eam), (PCB) ← (eam +2)	–	–	–	–	–	–	–	–	–	–
JMPP addr24	4	4	0	0	word (PC) ← ad24 0 to 15, (PCB) ← ad24 16 to 23	–	–	–	–	–	–	–	–	–	–
CALL @ear *4	2	6	1	(c)	word (PC) ← (ear)	–	–	–	–	–	–	–	–	–	–
CALL @eam *4	2+	7+ (a)	0	2× (c)	word (PC) ← (eam)	–	–	–	–	–	–	–	–	–	–
CALL addr16 *5	3	6	0	(c)	word (PC) ← addr16	–	–	–	–	–	–	–	–	–	–
CALLV #vct4 *5	1	7	0	2× (c)	Vector call instruction	–	–	–	–	–	–	–	–	–	–
CALLP @ear *6	2	10	2	2× (c)	word (PC) ← (ear) 0 to 15, (PCB) ← (ear) 16 to 23	–	–	–	–	–	–	–	–	–	–
CALLP @eam *6	2+	11+ (a)	0	*2	word (PC) ← (eam) 0 to 15, (PCB) ← (eam) 16 to 23	–	–	–	–	–	–	–	–	–	–
CALLP addr24 *7	4	10	0	2× (c)	word (PC) ← addr0 to 15, (PCB) ← addr16 to 23	–	–	–	–	–	–	–	–	–	–

*1: 4 when branching, 3 when not branching.
*2: (b) + 3 × (c)
*3: Read (word) branch address.
*4: W: Save (word) to stack; R: read (word) branch address.
*5: Save (word) to stack.
*6: W: Save (long word) to W stack; R: read (long word) R branch address.
*7: Save (long word) to stack.

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 20 Branch 2 Instructions [19 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
CBNE A, #imm8, rel	3	*1	0	0	Branch when byte (A) ≠ imm8	–	–	–	–	–	*	*	*	*	–
CWBNE A, #imm16, rel	4	*1	0	0	Branch when word (A) ≠ imm16	–	–	–	–	–	*	*	*	*	–
CBNE ear, #imm8, rel	4	*2	1	0	Branch when byte (ear) ≠ imm8	–	–	–	–	–	*	*	*	*	–
CBNE eam, #imm8, rel*10	4+	*3	0	(b)	Branch when byte (eam) ≠ imm8	–	–	–	–	–	*	*	*	*	–
CWBNE ear, #imm16, rel	5	*4	1	0	Branch when word (ear) ≠ imm16	–	–	–	–	–	*	*	*	*	–
CWBNE eam, #imm16, rel*10	5+	*3	0	(c)	Branch when word (eam) ≠ imm16	–	–	–	–	–	*	*	*	*	–
DBNZ ear, rel	3	*5	2	0	Branch when byte (ear) = (ear) – 1, and (ear) ≠ 0	–	–	–	–	–	*	*	*	–	–
DBNZ eam, rel	3+	*6	2	2× (b)	Branch when byte (eam) = (eam) – 1, and (eam) ≠ 0	–	–	–	–	–	*	*	*	–	*
DWBNZ ear, rel	3	*5	2	0	Branch when word (ear) = (ear) – 1, and (ear) ≠ 0	–	–	–	–	–	*	*	*	–	–
DWBNZ eam, rel	3+	*6	2	2× (c)	Branch when word (eam) = (eam) – 1, and (eam) ≠ 0	–	–	–	–	–	*	*	*	–	*
INT #vct8	2	20	0	8× (c)	Software interrupt	–	–	R	S	–	–	–	–	–	–
INT addr16	3	16	0	6× (c)	Software interrupt	–	–	R	S	–	–	–	–	–	–
INTP addr24	4	17	0	6× (c)	Software interrupt	–	–	R	S	–	–	–	–	–	–
INT9	1	20	0	8× (c)	Software interrupt	–	–	R	S	–	–	–	–	–	–
RETI	1	15	0	*7	Return from interrupt	–	–	*	*	*	*	*	*	*	–
LINK #local8	2	6	0	(c)	At constant entry, save old frame pointer to stack, set new frame pointer, and allocate local pointer area	–	–	–	–	–	–	–	–	–	–
UNLINK	1	5	0	(c)	At constant entry, retrieve old frame pointer from stack.	–	–	–	–	–	–	–	–	–	–
RET *8	1	4	0	(c)	Return from subroutine	–	–	–	–	–	–	–	–	–	–
RETP *9	1	6	0	(d)	Return from subroutine	–	–	–	–	–	–	–	–	–	–

*1: 5 when branching, 4 when not branching
*2: 13 when branching, 12 when not branching
*3: 7 + (a) when branching, 6 + (a) when not branching
*4: 8 when branching, 7 when not branching
*5: 7 when branching, 6 when not branching
*6: 8 + (a) when branching, 7 + (a) when not branching
*7: Set to 3 × (b) + 2 × (c) when an interrupt request occurs, and 6 × (c) for return.
*8: Retrieve (word) from stack
*9: Retrieve (long word) from stack
*10: In the CBNE/CWBNE instruction, do not use the RWj+ addressing mode.

Note : For an explanation of "(a)" to "(d)", refer to Table 4, "Number of Execution Cycles for Each Type of Addressing," and Table 6, "Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles."

Table 21 Other Control Instructions (Byte/Word/Long Word) [28 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
PUSHW A	1	4	0	(c)	word (SP) ← (SP) –2, ((SP)) ← (A)	–	–	–	–	–	–	–	–	–	–
PUSHW AH	1	4	0	(c)	word (SP) ← (SP) –2, ((SP)) ← (AH)	–	–	–	–	–	–	–	–	–	–
PUSHW PS	1	4	0	(c)	word (SP) ← (SP) –2, ((SP)) ← (PS)	–	–	–	–	–	–	–	–	–	–
PUSHW rlst	2	*3	*5	*4	(SP) ← (SP) –2n, ((SP)) ← (rlst)	–	–	–	–	–	–	–	–	–	–
POPW A	1	3	0	(c)	word (A) ← ((SP)), (SP) ← (SP) +2	–	*	–	–	–	–	–	–	–	–
POPW AH	1	3	0	(c)	word (AH) ← ((SP)), (SP) ← (SP) +2	–	–	–	–	–	–	–	–	–	–
POPW PS	1	4	0	(c)	word (PS) ← ((SP)), (SP) ← (SP) +2	–	–	*	*	*	*	*	*	*	–
POPW rlst	2	*2	*5	*4	(rlst) ← ((SP)), (SP) ← (SP) +2n	–	–	–	–	–	–	–	–	–	–
JCTX @A	1	14	0	6× (c)	Context switch instruction	–	–	*	*	*	*	*	*	*	–
AND CCR, #imm8	2	3	0	0	byte (CCR) ← (CCR) and imm8	–	–	*	*	*	*	*	*	*	–
OR CCR, #imm8	2	3	0	0	byte (CCR) ← (CCR) or imm8	–	–	*	*	*	*	*	*	*	–
MOV RP, #imm8	2	2	0	0	byte (RP) ←imm8	–	–	–	–	–	–	–	–	–	–
MOV ILM, #imm8	2	2	0	0	byte (ILM) ←imm8	–	–	–	–	–	–	–	–	–	–
MOVEA RWi, ear	2	3	1	0	word (RWi) ←ear	–	–	–	–	–	–	–	–	–	–
MOVEA RWi, eam	2+	2+ (a)	1	0	word (RWi) ←eam	–	–	–	–	–	–	–	–	–	–
MOVEA A, ear	2	1	0	0	word(A) ←ear	–	*	–	–	–	–	–	–	–	–
MOVEA A, eam	2+	1+ (a)	0	0	word (A) ←eam	–	*	–	–	–	–	–	–	–	–
ADDSP #imm8	2	3	0	0	word (SP) ← (SP) +ext (imm8)	–	–	–	–	–	–	–	–	–	–
ADDSP #imm16	3	3	0	0	word (SP) ← (SP) +imm16	–	–	–	–	–	–	–	–	–	–
MOV A, brgl	2	*1	0	0	byte (A) ← (brgl)	Z	*	–	–	–	*	*	–	–	–
MOV brg2, A	2	1	0	0	byte (brg2) ← (A)	–	–	–	–	–	*	*	–	–	–
NOP	1	1	0	0	No operation	–	–	–	–	–	–	–	–	–	–
ADB	1	1	0	0	Prefix code for accessing AD space	–	–	–	–	–	–	–	–	–	–
DTB	1	1	0	0	Prefix code for accessing DT space	–	–	–	–	–	–	–	–	–	–
PCB	1	1	0	0	Prefix code for accessing PC space	–	–	–	–	–	–	–	–	–	–
SPB	1	1	0	0	Prefix code for accessing SP space	–	–	–	–	–	–	–	–	–	–
NCC	1	1	0	0	Prefix code for no flag change	–	–	–	–	–	–	–	–	–	–
CMR	1	1	0	0	Prefix code for common register bank	–	–	–	–	–	–	–	–	–	–

*1: PCB, ADB, SSB, USB, and SPB : 1 state
DTB, DPR : 2 states

*2: 7 + 3 × (pop count) + 2 × (last register number to be popped), 7 when rlst = 0 (no transfer register)

*3: 29 + (push count) – 3 × (last register number to be pushed), 8 when rlst = 0 (no transfer register)

*4: Pop count × (c), or push count × (c)

*5: Pop count or push count.

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 22 Bit Manipulation Instructions [21 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOVB A, dir:bp	3	5	0	(b)	byte (A) ← (dir:bp) b	Z	*	–	–	–	*	*	–	–	–
MOVB A, addr16:bp	4	5	0	(b)	byte (A) ← (addr16:bp) b	Z	*	–	–	–	*	*	–	–	–
MOVB A, io:bp	3	4	0	(b)	byte (A) ← (io:bp) b	Z	*	–	–	–	*	*	–	–	–
MOVB dir:bp, A	3	7	0	2× (b)	bit (dir:bp) b ← (A)	–	–	–	–	–	*	*	–	–	*
MOVB addr16:bp, A	4	7	0	2× (b)	bit (addr16:bp) b ← (A)	–	–	–	–	–	*	*	–	–	*
MOVB io:bp, A	3	6	0	2× (b)	bit (io:bp) b ← (A)	–	–	–	–	–	*	*	–	–	*
SETB dir:bp	3	7	0	2× (b)	bit (dir:bp) b ← 1	–	–	–	–	–	–	–	–	–	*
SETB addr16:bp	4	7	0	2× (b)	bit (addr16:bp) b ← 1	–	–	–	–	–	–	–	–	–	*
SETB io:bp	3	7	0	2× (b)	bit (io:bp) b ← 1	–	–	–	–	–	–	–	–	–	*
CLRB dir:bp	3	7	0	2× (b)	bit (dir:bp) b ← 0	–	–	–	–	–	–	–	–	–	*
CLRB addr16:bp	4	7	0	2× (b)	bit (addr16:bp) b ← 0	–	–	–	–	–	–	–	–	–	*
CLRB io:bp	3	7	0	2× (b)	bit (io:bp) b ← 0	–	–	–	–	–	–	–	–	–	*
BBC dir:bp, rel	4	*1	0	(b)	Branch when (dir:bp) b = 0	–	–	–	–	–	–	*	–	–	–
BBC addr16:bp, rel	5	*1	0	(b)	Branch when (addr16:bp) b = 0	–	–	–	–	–	–	*	–	–	–
BBC io:bp, rel	4	*2	0	(b)	Branch when (io:bp) b = 0	–	–	–	–	–	–	*	–	–	–
BBS dir:bp, rel	4	*1	0	(b)	Branch when (dir:bp) b = 1	–	–	–	–	–	–	*	–	–	–
BBS addr16:bp, rel	5	*1	0	(b)	Branch when (addr16:bp) b = 1	–	–	–	–	–	–	*	–	–	–
BBS io:bp, rel	4	*2	0	(b)	Branch when (io:bp) b = 1	–	–	–	–	–	–	*	–	–	–
SBBS addr16:bp, rel	5	*3	0	2× (b)	Branch when (addr16:bp) b = 1, bit = 1	–	–	–	–	–	–	*	–	–	*
WBTS io:bp	3	*4	0	*5	Wait until (io:bp) b = 1	–	–	–	–	–	–	–	–	–	–
WBTC io:bp	3	*4	0	*5	Wait until (io:bp) b = 0	–	–	–	–	–	–	–	–	–	–

*1: 8 when branching, 7 when not branching
*2: 7 when branching, 6 when not branching
*3: 10 when condition is satisfied, 9 when not satisfied
*4: Undefined count
*5: Until condition is satisfied

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 23 Accumulator Manipulation Instructions (Byte/Word) [6 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
SWAP	1	3	0	0	byte (A) 0 to 7 ↔ (A) 8 to 15	–	–	–	–	–	–	–	–	–	–
SWAPW	1	2	0	0	word (AH) ↔ (AL)	–	*	–	–	–	–	–	–	–	–
EXT	1	1	0	0	byte sign extension	X	–	–	–	–	*	*	–	–	–
EXTW	1	2	0	0	word sign extension	–	X	–	–	–	*	*	–	–	–
ZEXT	1	1	0	0	byte zero extension	Z	–	–	–	–	R	*	–	–	–
ZEXTW	1	1	0	0	word zero extension	–	Z	–	–	–	R	*	–	–	–

Note: For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 6, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 24 String Instructions [10 Instructions]

Mnemonic	#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOVS/MOVSI	2	*2	*5	*3	Byte transfer @AH+ ← @AL+, counter = RW0	–	–	–	–	–	–	–	–	–	–
MOVSD	2	*2	*5	*3	Byte transfer @AH– ← @AL–, counter = RW0	–	–	–	–	–	–	–	–	–	–
SCEQ/SCEQI	2	*1	*5	*4	Byte retrieval (@AH+) – AL, counter = RW0	–	–	–	–	–	*	*	*	*	–
SCEQD	2	*1	*5	*4	Byte retrieval (@AH–) – AL, counter = RW0	–	–	–	–	–	*	*	*	*	–
FISL/FILSI	2	6m +6	*5	*3	Byte filling @AH+ ← AL, counter = RW0	–	–	–	–	–	*	*	–	–	–
MOVSW/MOVSWI	2	*2	*8	*6	Word transfer @AH+ ← @AL+, counter = RW0	–	–	–	–	–	–	–	–	–	–
MOVSWD	2	*2	*8	*6	Word transfer @AH– ← @AL–, counter = RW0	–	–	–	–	–	–	–	–	–	–
SCWEQ/SCWEQI	2	*1	*8	*7	Word retrieval (@AH+) – AL, counter = RW0	–	–	–	–	–	*	*	*	*	–
SCWEQD	2	*1	*8	*7	Word retrieval (@AH–) – AL, counter = RW0	–	–	–	–	–	*	*	*	*	–
FILSW/FILSWI	2	6m +6	*8	*6	Word filling @AH+ ← AL, counter = RW0	–	–	–	–	–	*	*	–	–	–

m: RW0 value (counter value)
n: Loop count
*1: 5 when RW0 is 0, 4 + 7 × (RW0) for count out, and 7 × n + 5 when match occurs
*2: 5 when RW0 is 0, 4 + 8 × (RW0) in any other case
*3: (b) × (RW0) + (b) × (RW0) when accessing different areas for the source and destination, calculate (b) separately for each.
*4: (b) × n
*5: 2 × (RW0)
*6: (c) × (RW0) + (c) × (RW0) when accessing different areas for the source and destination, calculate (c) separately for each.
*7: (c) × n
*8: 2 × (RW0)

Note : For an explanation of "(a)" to "(d)", refer to Table 4, "Number of Execution Cycles for Each Type of Addressing," and Table 6, "Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles."

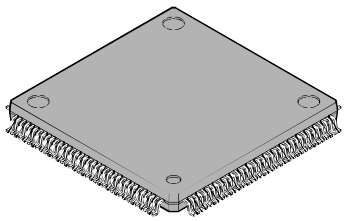
■ ORDERING INFORMATION

Part number	Package	Remarks
MB90F372PMT-G MB90372PMT-G-XXX	144-pin Plastic LQFP (FPT-144P-M12)	XXX is the ROM release number.

■ PACKAGE DIMENSIONS

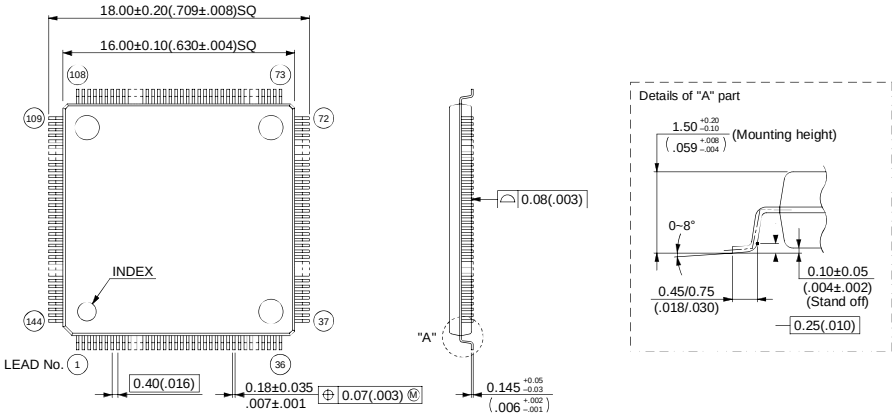
144-pin plastic LQFP
(FPT-144P-M12)

144-pin plastic LQFP



(FPT-144P-M12)

Lead pitch	0.40 mm
Package width x package length	16.0 x 16.0 mm
Lead shape	Gullwing
Sealing method	Plastic mold
Mounting height	1.70 mm MAX
Weight	0.88g



© 1998 FUJITSU LIMITED F144024S-2C-2

Dimensions mm (inches)

Dimensions in mm (inches)

FUJITSU LIMITED

For further information please contact:

Japan

FUJITSU LIMITED
Marketing Division
Electronic Devices
Shinjuku Dai-Ichi Seimei Bldg. 7-1,
Nishishinjuku 2-chome, Shinjuku-ku,
Tokyo 163-0721, Japan
Tel: +81-3-5322-3353
Fax: +81-3-5322-3386

http://edevice.fujitsu.com/

North and South America

FUJITSU MICROELECTRONICS AMERICA, INC.
3545 North First Street,
San Jose, CA 95134-1804, U.S.A.
Tel: +1-408-922-9000
Fax: +1-408-922-9179

Customer Response Center
Mon. - Fri.: 7 am - 5 pm (PST)
Tel: +1-800-866-8608
Fax: +1-408-922-9179

http://www.fma.fujitsu.com/

Europe

FUJITSU MICROELECTRONICS EUROPE GmbH
Am Siebenstein 6-10,
D-63303 Dreieich-Buchschlag,
Germany
Tel: +49-6103-690-0
Fax: +49-6103-690-122

http://www.fme.fujitsu.com/

Asia Pacific

FUJITSU MICROELECTRONICS ASIA PTE LTD.
#05-08, 151 Lorong Chuan,
New Tech Park,
Singapore 556741
Tel: +65-6281-0770
Fax: +65-6281-0220

http://www.fmal.fujitsu.com/

Korea

FUJITSU MICROELECTRONICS KOREA LTD.
1702 KOSMO TOWER, 1002 Daechi-Dong,
Kangnam-Gu,Seoul 135-280
Korea
Tel: +82-2-3484-7100
Fax: +82-2-3484-7111
http://www.fmk.fujitsu.com/

F0208
© FUJITSU LIMITED Printed in Japan

All Rights Reserved.

The contents of this document are subject to change without notice.
Customers are advised to consult with FUJITSU sales representatives before ordering.

The information and circuit diagrams in this document are presented as examples of semiconductor device applications, and are not intended to be incorporated in devices for actual use. Also, FUJITSU is unable to assume responsibility for infringement of any patent rights or other rights of third parties arising from the use of this information or circuit diagrams.

The products described in this document are designed, developed and manufactured as contemplated for general use, including without limitation, ordinary industrial use, general office use, personal use, and household use, but are not designed, developed and manufactured as contemplated (1) for use accompanying fatal risks or dangers that, unless extremely high safety is secured, could have a serious effect to the public, and could lead directly to death, personal injury, severe physical damage or other loss (i.e., nuclear reaction control in nuclear facility, aircraft flight control, air traffic control, mass transport control, medical life support system, missile launch control in weapon system), or (2) for use requiring extremely high reliability (i.e., submersible repeater and artificial satellite).
Please note that Fujitsu will not be liable against you and/or any third party for any claims or damages arising in connection with above-mentioned uses of the products.

Any semiconductor devices have an inherent chance of failure. You must protect against injury, damage or loss from such failures by incorporating safety design measures into your facility and equipment such as redundancy, fire protection, and prevention of over-current levels and other abnormal operating conditions.

If any products described in this document represent goods or technologies subject to certain restrictions on export under the Foreign Exchange and Foreign Trade Law of Japan, the prior authorization by Japanese government will be required for export of those products from Japan.