FUJITSU SEMICONDUCTOR
DATA SHEET

DS07-13601-5E

16-bit Proprietary Microcontroller

CMOS

F²MC-16L MB90630A Series

MB90632A/634A/P634A

■ DESCRIPTION

The MB90630A series are 16-bit microcontrollers designed for high speed real-time processing in consumer product applications such as controlling video cameras, VCRs, or copiers. The series uses the F²MC*-16L CPU. The chips incorporate an eight channels 10-bit A/D converter, two channels 8-bit D/A converter, UART two channels, two channels serial interface, 8/16-bit up/down counter, 16-bit I/O timer (two channels input capture, four channels output compare, and one channel 16-bit free-run timer).

*: F²MC stands for FUJITSU Flexible Microcontroller.

■ FEATURES

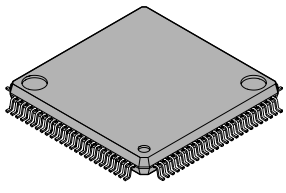
F²MC-16L CPU

- Minimum execution time: 62.5 ns/4 MHz oscillation (Uses PLL clock multiplication), maximum multiplier = 4
- Instruction set optimized for controller applications
  Object code compatibility with F²MC-16(H)
  Wide range of data types (bit, byte, word, and long word)
  Improved instruction cycles provide increased speed
  Additional addressing modes: 23 modes
  High code efficiency
  Access mothods (bank access, linear pointer)

(Continued)

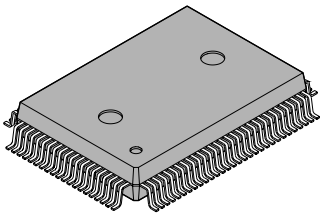
■ PACKAGE

100-pin Plastic LQFP



(FPT-100P-M05)

100-pin Plastic QFP



(FPT-100P-M06)

(Continued)

High precision operations are enhanced by use of a 32-bit accumulator
Extended intelligent I/O service (access area extended to 64 KB)
Maximum memory space: 16 MB

- Enhanced high level language (C) and multitasking support insturctions
  Use of a system stack pointer
  Enhanced pointer indirect instructions
  Barrel shift instructions
- Improved execution speed: Four byte instruction queue
- Powerful interrupt function
- Automatic data transfer function that does not use insturction (IIOS)

Internal peripherals

- ROM: 32 Kbytes (MB90632A)
  64 Kbytes (MB90634A)
  One-time PROM: 64 Kbytes (MB90P634A)
- RAM: 1 Kbytes (MB90632A)
  2 Kbytes (MB90634A)
  3 Kbytes (MB90P634A)
- General-purpose ports: 82 ports max.
- 10-bit A/D converter (RC successive approximation): eight channels (10-bit resolution, conversion time = 5.2 µs at 4 MHz with a × 4 multiplier)
- 8-bit D/A converter two channels (8-bit resolution)
- UART (can also be used as a serial port) two channels
- I/O expansion serial interface two channels
- 8/16-bit PPG (can be set to either 8-bit × two channels or 16-bit × one channel) one channel
- 16-bit I/O timer one channel
  (two channels input capture, four channels output compare, and one channel free-run timer)
- Clock output generator
- Timebase counter/watchdog timer (18-bit)
- Low-power consumption modes
- The device types are classified by the initial value of the oscillation stabilization delay time.
  Oscillation stabilization delay time initial value = 2.05 ms: MB90630A series (MB90632A/634A/P634A)
- Package: LQFP-100 (QFP-100 planned)
- CMOS technology

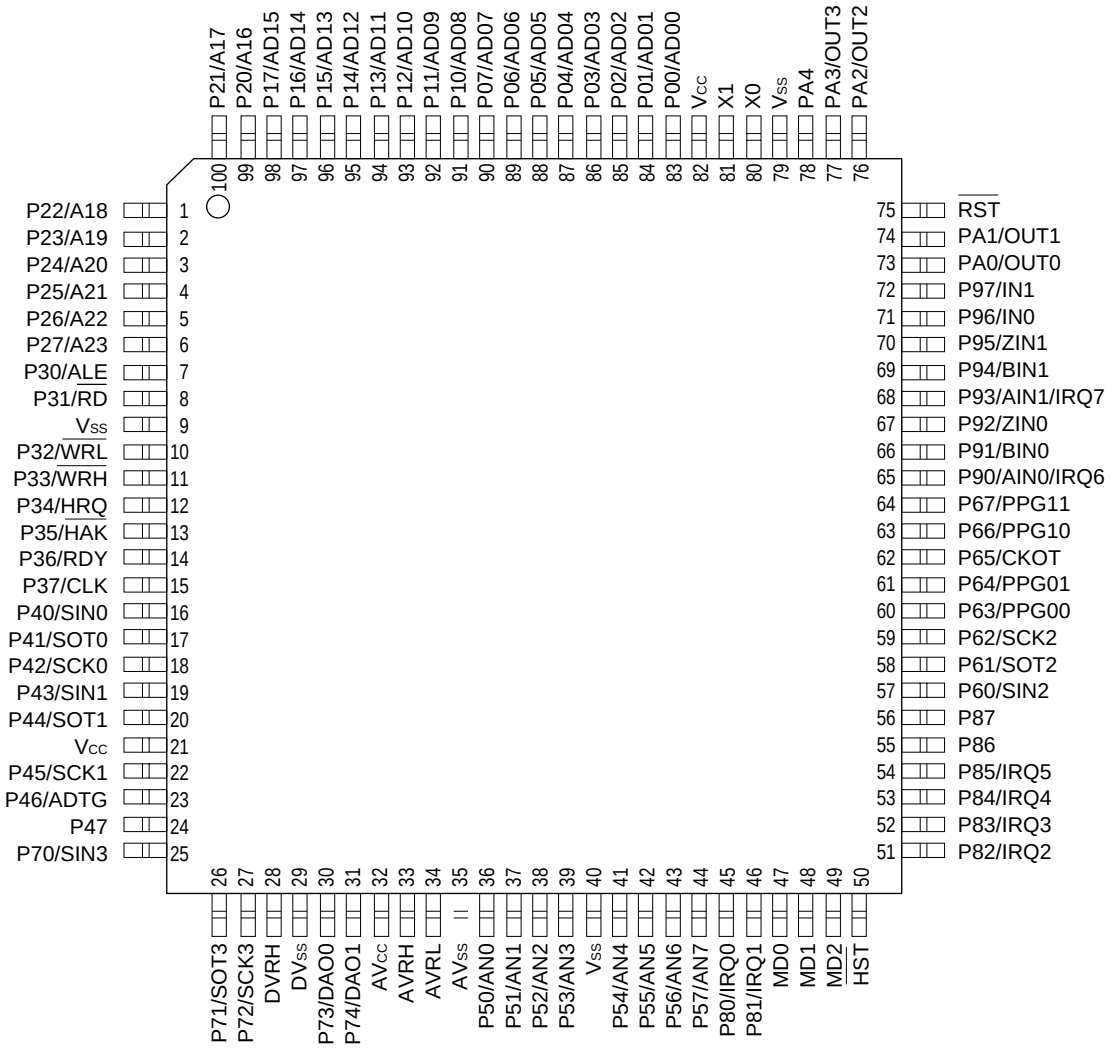
■ PRODUCT LINEUP

Part number / Parameter	MB90P634A	MB90632A	MB90634A
Classification	OTPROM	Mask ROM	
ROM size	64 Kbyte	32 Kbyte	64 Kbyte
RAM size	3 Kbyte	1 Kbyte	2 Kbyte
CPU functions	Number of instructions : 340 Instruction bit length : 8/16 bits Instruction length : 1/7 bytes Data bit length : 1/4/8/16/32 bits Minimum execution time : 62.5 ns/4 MHz (PLL multiplier = 4) Interrupt processing time : 1000 ns/16 MHz (minimum)		
Ports	I/O ports (CMOS/TTL) : 82 ports (Input pull-up resistors available : 24 ports Can be set as open-drain outputs : 8 ports)		
Package	FPT-100P-M05 FPT-100P-M06		
A/D converter	10-bit resolution, 5.2 µs conversion time (at 4 MHz with a ×4 multiplier) RC successive approximation, 8 channels (multiplexed inputs)		
D/A converter	8-bit resolution R-2R type, 2 channels (independent)		
UART	Full-duplex, double-buffered (8-bit), internal baud rate correction circuit that uses the operating clock NRZ-type transfer, supports MIDI frequencies, 2 channels		
Serial interface	8-bit data register. LSB-first or MSB-first operation can be selected. The transfer shift clock can be input externally. The internal shift clock includes a built-in operating clock correction circuit. 1 channel		
8/16-bit PPG	Can operate as two independent channels in 8-bit mode. Can also be used as a single-channel 16-bit PPG. 1 channel		
8/16-bit up/down counter	6 event inputs. Can operate as two independent 8-bit up/down counter channels. Can also be used as a single-channel 16-bit counter. Includes reload and compare functions. 1 channel		
16-bit I/O timer	Consists of 2 × input capture, 4 × output compare, and 1 × free-run timer. 1 channel		
Timer functions	Timebase timer/watchdog timer (18-bit)		
Low-power consumption modes	Includes sleep, stop, and hardware standby functions		
Oscillation stabilization delay time	The initial value of the oscillation stabilization delay time is 64 ms. The oscillation stabilization delay time can also be set to 0 ms, 2.05 ms, 8.19 ms, or 64 ms (for an crystal oscillator). The MB90630A series are for FAR oscillators.		
External interrupt	8 inputs External interrupt mode (Interrupts can be generated from four different types of request signal)		
PLL function	Selectable multiplier: 1/2/3/4 (Set a multiplier that does not exceed the assured operation frequency range.)		
Other	V_{PP} is shared with the MD2 pin (for EPROM programming)	—	—

MB90630A Series

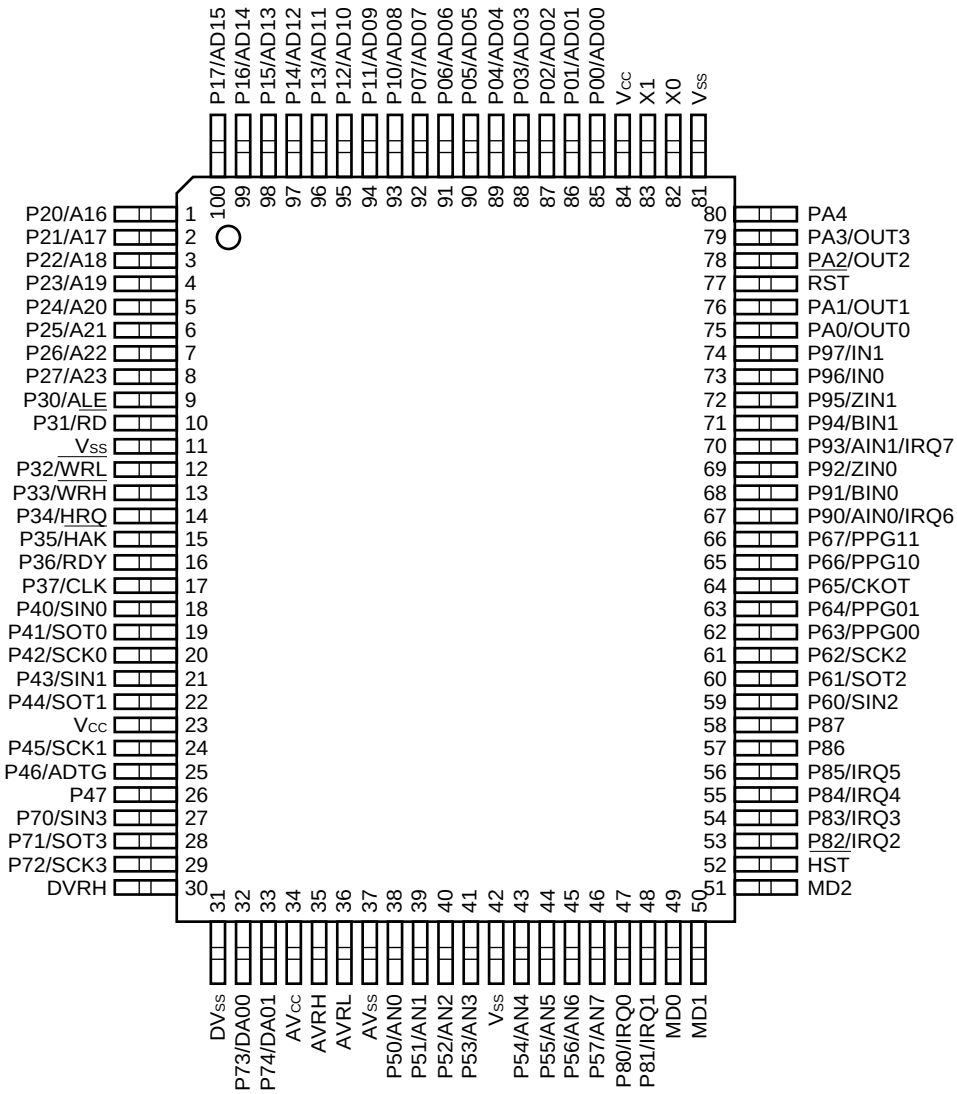
■ PIN ASSIGNMENT

(TOP VIEW)



(FPT-100P-M05)

(TOP VIEW)



(FPT-100P-M06)

■ PIN DESCRIPTION

Pin no.		Pin name	Circuit type	Function
LQFP*1	QFP*2			
80	82	X0	A	Oscillator pin
81	83	X1	A	Oscillator pin
50	52	$\overline{HST}$	C	Hardware standby input pin
75	77	$\overline{RST}$	B	Reset input pin
83 to 90	85 to 92	P00 to P07	D (STBC)	General-purpose I/O ports Pull-up resistors can be set (RD07 to RD00 = “1”) using the pull-up resistor setting register (RDR0). The setting does not apply for ports set as outputs (D07 to D00 = “1”: invalid at the output setting).
		AD00 to AD07		In external bus mode, the pins function as the lower data I/O or lower address outputs (AD00 to AD07).
91 to 98	93 to 100	P10 to P17	D (STBC)	General-purpose I/O ports Pull-up resistors can be set (RD17 to RD10 = “1”) using the pull-up resistor setting register (RDR1). The setting does not apply for ports set as outputs (D17 to D10 = “1”: invalid at the output setting).
		AD08 to AD15		In 16-bit external bus mode, the pins function as the upper data I/O or middle address outputs (AD08 to AD15).
99, 100, 1 to 6	1 to 8	P20 to P27	H (STBC)	General-purpose I/O ports In external bus mode, pins for which the corresponding bit in the HACR register is “0” function as the P20 to P27 pins.
		A16 to A23		In external bus mode, pins for which the corresponding bit in the HACR register is “1” function as the upper address output pins (A16 to A23).
7	9	P30	H (STBC)	General-purpose I/O port Functions as the ALE pin in external bus mode.
		ALE		Functions as the address latch enable signal.
8	10	P31	H (STBC)	General-purpose I/O port Functions as the $\overline{RD}$ pin in external bus mode.
		$\overline{RD}$		Functions as the read strobe output ($\overline{RD}$).
10	12	P32	H (STBC)	General-purpose I/O port Functions as the $\overline{WR}$ pin in external bus mode if the WRE bit in the EPCR register is “1”.
		$\overline{WRL}$		Functions as the lower data write strobe output ($\overline{WRL}$).
11	13	P33	H (STBC)	General-purpose I/O port Functions as the $\overline{WRH}$ pin in 16-bit external bus mode if the WRE bit in the EPCR register is “1”.
		$\overline{WRH}$		Functions as the upper data write strobe output ($\overline{WRH}$).

(Continued)

STBC: Incorporates standby control
*1: LQFP (FPT-100P-M05)
*2: QFP (FPT-100P-M06)

Pin no.		Pin name	Circuit type	Function
LQFP*1	QFP*2			
12	14	P34	H (STBC)	General-purpose I/O port Functions as the HRQ pin in external bus mode if the HDE bit in the EPCR register is “1”.
		HRQ		Functions as the hold request input pin (HRQ).
13	15	P35	H (STBC)	General-purpose I/O port Functions as the $\overline{\text{HAK}}$ pin in external bus mode if the HDE bit in the EPCR register is “1”.
		$\overline{\text{HAK}}$		Functions as the hold acknowledge output ($\overline{\text{HAK}}$) pin.
14	16	P36	H (STBC)	General-purpose I/O port Functions as the RDY pin in external bus mode if the RYE bit in the EPCR register is “1”.
		RDY		Functions as the external ready input (RDY) pin.
15	17	P37	H (STBC)	General-purpose I/O port Functions as the CLK pin in external bus mode if the CKE bit in the EPCR register is “1”.
		CLK		Functions as the machine cycle clock output (CLK) pin.
16	18	P40	G (STBC)	General-purpose I/O port When UART0 is operating, the data at the pin is used as the serial input (SIN0). Can be set as an open-drain output port (OD40 = “1”) by the open-drain control register (ODR4). The setting does not apply for ports set as inputs (D40 = “0”: invalid at the input setting).
		SIN0		Functions as the UART0 serial input (SIN0).
17	19	P41	F (STBC)	General-purpose I/O port Functions as the SOT0 pin if the SOE bit in the UMC register is “1”. Can be set as an open-drain output port (OD41 = “1”) by the open-drain control register (ODR4). The setting does not apply for ports set as inputs (D41 = “0”: invalid at the input setting).
		SOT0		Functions as the UART0 serial data output pin (SOT0).
18	20	P42	G (STBC)	General-purpose I/O port When UART0 is operating in external shift clock mode, the data at the pin is used as the clock input (SCK0). Also, functions as the SCK0 pin if the SOE bit in the UMC register is “1”. Can be set as an open-drain output port (OD42 = “1”) by the open-drain control register (ODR4). The setting does not apply for ports set as inputs (D42 = “0”: invalid at the input setting).
		SCK0		Functions as the UART0 serial clock I/O pin (SCK0).

(Continued)

STBC: Incorporates standby control
*1: LQFP (FPT-100P-M05)
*2: QFP (FPT-100P-M06)

Pin no.		Pin name	Circuit type	Function
LQFP*1	QFP*2			
19	21	P43	G (STBC)	General-purpose I/O port When I/O expansion serial is operating, the data at the pin is used as the serial input (SIN1). Can be set as an open-drain output port (OD43 = “1”) by the open-drain control register (ODR4). The setting does not apply for ports set as inputs (D43 = “0”: invalid at the input setting).
		SIN1		Functions as the serial input for I/O expansion serial data.
20	22	P44	F (STBC)	General-purpose I/O port Functions as the SOT1 pin if the SOE bit in the UMC register is “1”. Can be set as an open-drain output port (OD44 = “1”) by the open-drain control register (ODR4). The setting does not apply for ports set as inputs (D44 = “0”: invalid at the input setting).
		SOT1		Functions as the output pin (SOT1) for I/O expansion serial data.
22	24	P45	G (STBC)	General-purpose I/O port When I/O expansion serial is operating in external shift clock mode, the data at the pin is used as the clock input (SCK1). Also, functions as the SCK1 pin if the SOE bit in the UMC register is “1”. Can be set as an open-drain output port (OD45 = “1”) by the open-drain control register (ODR4). The setting does not apply for ports set as inputs (D45 = “0”: invalid at the input setting).
		SCK1		Functions as the I/O expansion serial clock I/O pin (SCK1).
23	25	P46	F (STBC)	General-purpose I/O port Can be set as an open-drain output port (OD46 = “1”) by the open-drain control register (ODR4). The setting does not apply for ports set as inputs (D46 = “0”: invalid at the input setting).
		ADTG		Functions as the external trigger input pin for the A/D converter.
24	26	P47	F (STBC)	General-purpose I/O port Can be set as an open-drain output port (OD47 = “1”) by the open-drain control register (ODR4). The setting does not apply for ports set as inputs (D47 = “0”: invalid at the input setting).
36 to 39, 41 to 44	38 to 41, 43 to 46	P50 to P57	K (STBC)	General-purpose I/O ports
		AN0 to AN7		The pins are used as analog inputs (AN0 to AN7) when the A/D converter is operating.
25	27	P70	I (STBC)	General-purpose I/O port
		SIN3		Functions as the UART1 serial input (SIN3).
26	28	P71	H (STBC)	General-purpose I/O port
		SOT3		Functions as the UART1 serial data output pin (SOT3).
27	29	P72	I (STBC)	General-purpose I/O port
		SCK3		Functions as the UART1 serial clock I/O pin (SCK0).

(Continued)

STBC: Incorporates standby control
*1: LQFP (FPT-100P-M05)
*2: QFP (FPT-100P-M06)

Pin no.		Pin name	Circuit type	Function
LQFP*1	QFP*2			
30	32	P73	L (STBC)	General-purpose I/O port Functions as a D/A output pin when DAE0 = “1” in the D/A control register (DACR).
		DAO0		Functions as D/A output 0 when the D/A converter is operating.
31	33	P74	L (STBC)	General-purpose I/O port Functions as a D/A output pin when DAE1 = “1” in the D/A control register (DACR).
		DAO1		Functions as D/A output 1 when the D/A converter is operating.
45	47	P80	I	General-purpose I/O port
		IRQ0		Functions as external interrupt request I/O 0.
46	48	P81	I	General-purpose I/O port
		IRQ1		Functions as external interrupt request I/O 1.
51	53	P82	I	General-purpose I/O port
		IRQ2		Functions as external interrupt request I/O 2.
52	54	P83	I	General-purpose I/O port
		IRQ3		Functions as external interrupt request I/O 3.
53	55	P84	I	General-purpose I/O port
		IRQ4		Functions as external interrupt request I/O 4.
54	56	P85	I	General-purpose I/O port
		IRQ5		Functions as external interrupt request I/O 5.
55	57	P86	H (STBC)	General-purpose I/O port This applies in all cases.
56	58	P87	H (STBC)	General-purpose I/O port This applies in all cases.
57	59	P60	E (STBC)	General-purpose I/O port A pull-up resistor can be set (RD60 = “1”) using the pull-up resistor setting register (RDR6). The setting does not apply for ports set as outputs (D60 = “1”: invalid at the output setting).
		SIN2		Functions as a data input pin (SIN2) for I/O expansion serial.
58	60	P61	D (STBC)	General-purpose I/O port Functions as the SOT2 pin if the SOE bit in the UMC register is “1”. A pull-up resistor can be set (RD61 = “1”) using the pull-up resistor setting register (RDR6). The setting does not apply for ports set as outputs (D61 = “1”: invalid at the output setting).
		SOT2		Functions as an output pin (SOT2) for I/O expansion serial data.

(Continued)

STBC: Incorporates standby control
*1: LQFP (FPT-100P-M05)
*2: QFP (FPT-100P-M06)

Pin no.		Pin name	Circuit type	Function
LQFP*1	QFP*2			
59	61	P62	E (STBC)	General-purpose I/O port When I/O expansion serial is operating in external shift clock mode, the data at the pin is used as the clock input (SCK2). Also, functions as the SCK2 pin if the SOE bit in the UMC register is “1”. A pull-up resistor can be set (RD62 = “1”) using the pull-up resistor setting register (RDR6). The setting does not apply for ports set as outputs (D62 = “1”: invalid at the output setting).
		SCK2		Functions as the I/O expansion serial clock I/O pin (SCK2).
60	62	P63	D (STBC)	General-purpose I/O port A pull-up resistor can be set (RD63 = “1”) using the pull-up resistor setting register (RDR6). The setting does not apply for ports set as outputs (D63 = “1”: invalid at the output setting).
		PPG00		Functions as the PPG00 output when PPG output is enabled.
61	63	P64	D (STBC)	General-purpose I/O port A pull-up resistor can be set (RD64 = “1”) using the pull-up resistor setting register (RDR6). The setting does not apply for ports set as outputs (D64 = “1”: invalid at the output setting).
		PPG01		Functions as the PPG01 output when PPG output is enabled.
62	64	P65	D (STBC)	General-purpose I/O port A pull-up resistor can be set (RD65 = “1”) using the pull-up resistor setting register (RDR6). The setting does not apply for ports set as outputs (D65 = “1”: invalid at the output setting).
		CKOT		Functions as the CKOT output when CKOT is operating.
63	65	P66	D (STBC)	General-purpose I/O port A pull-up resistor can be set (RD66 = “1”) using the pull-up resistor setting register (RDR6). The setting does not apply for ports set as outputs (D66 = “1”: invalid at the output setting).
		PPG10		Functions as the PPG10 output when PPG output is enabled.
64	66	P67	D (STBC)	General-purpose I/O port A pull-up resistor can be set (RD67 = “1”) using the pull-up resistor setting register (RDR6). The setting does not apply for ports set as outputs (D67 = “1”: invalid at the output setting).
		PPG11		Functions as the PPG11 output when PPG output is enabled.
65	67	P90	I	General-purpose I/O port
		AIN0		Input to channel 0 of the 8/16-bit up/down timer.
		IRQ6		Functions as an interrupt request input.
66	68	P91	I (STBC)	General-purpose I/O port
		BIN0		Input to channel 0 of the 8/16-bit up/down timer.

(Continued)

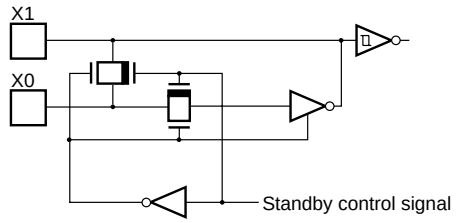
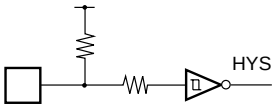
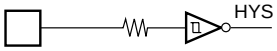
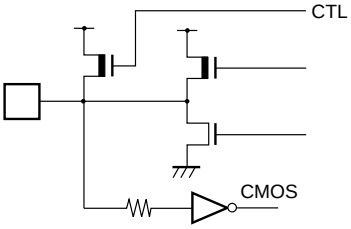
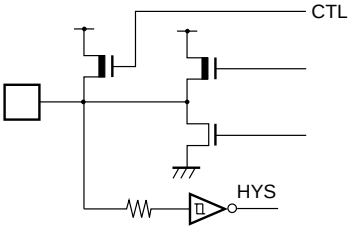
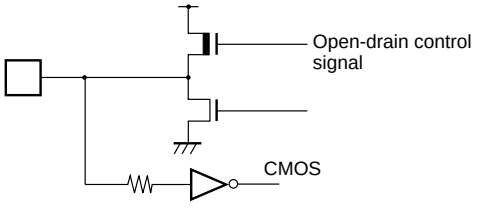
STBC: Incorporates standby control
*1: LQFP (FPT-100P-M05)
*2: QFP (FPT-100P-M06)

(Continued)

Pin no.		Pin name	Circuit type	Function
LQFP*1	QFP*2			
67	69	P92	I (STBC)	General-purpose I/O port
		ZIN0		Input to channel 0 of the 8/16-bit up/down timer.
68	70	P93	I	General-purpose I/O port
		AIN1		Input to channel 1 of the 8/16-bit up/down timer.
		IRQ7		Functions as an interrupt request input.
69	71	P94	I (STBC)	General-purpose I/O port
		BIN1		Input to channel 1 of the 8/16-bit up/down timer.
70	72	P95	I (STBC)	General-purpose I/O port
		ZIN1		Input to channel 1 of the 8/16-bit up/down timer.
71	73	P96	I (STBC)	General-purpose I/O port
		IN0		Trigger input for channel 0 of the input capture.
72	74	P97	I (STBC)	General-purpose I/O port
		IN1		Trigger input for channel 1 of the input capture.
73	75	PA0	H (STBC)	General-purpose I/O port
		OUT0		Event output for channel 0 of the output compare.
74	76	PA1	H (STBC)	General-purpose I/O port
		OUT1		Event output for channel 1 of the output compare.
76	78	PA2	H (STBC)	General-purpose I/O port
		OUT2		Event output for channel 2 of the output compare.
77	79	PA3	H (STBC)	General-purpose I/O port
		OUT3		Event output for channel 3 of the output compare.
78	80	PA4	H (STBC)	General-purpose I/O port
32	34	AV_{CC}	—	A/D converter power supply pin
35	37	AV_{SS}	—	A/D converter power supply pin
33	35	AVRH	—	A/D converter external reference power supply pin
34	36	AVRL	—	A/D converter external reference power supply pin
28	30	DVRH	—	D/A converter external reference power supply pin
29	31	DV_{SS}	—	D/A converter power supply pin
47 to 49	49 to 51	MD0 to MD2	C	Operating mode selection pins. Connect directly to V_{CC} or V_{SS}.
21, 82	23, 84	V_{CC}	—	Power supply (5.0 V) input pin
9, 40, 79	11, 42, 81	V_{SS}	—	Power supply (0.0 V) input pin

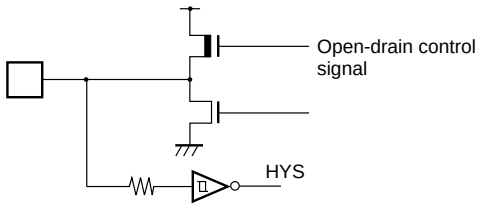
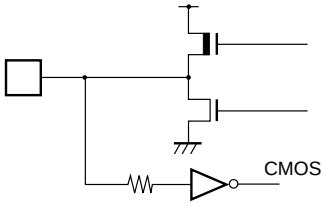
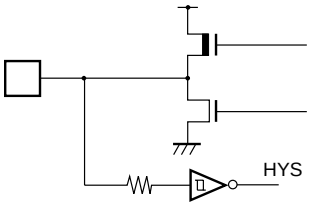
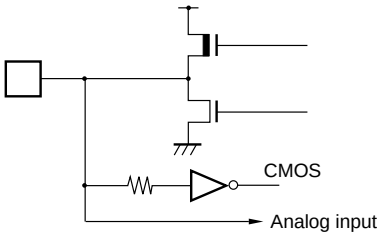
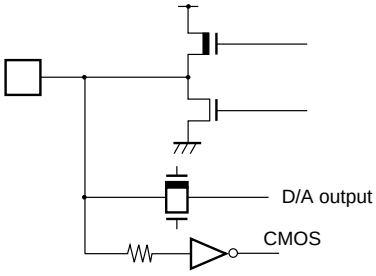
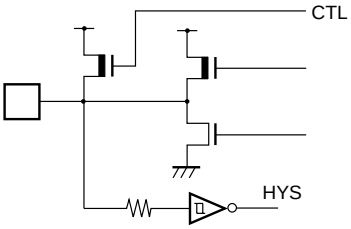
STBC: Incorporates standby control
*1: LQFP (FPT-100P-M05)
*2: QFP (FPT-100P-M06)

■ I/O CIRCUIT TYPE

Type	Circuit	Remarks
A		• Oscillator feedback Registance 1 MΩ (approx.)
B		• Hysteresis input with pull-up Registance 50 kΩ (approx.)
C		• Hysteresis input port
D		• Incorporates pull-up resistor control (for input) Registance 50 kΩ (approx.) • CMOS level I/O
E		• Incorporates pull-up resistor control (for input) Registance 50 kΩ (approx.) • CMOS level output • Hysteresis input
F		• CMOS level I/O • Open-drain control signal

(Continued)

(Continued)

Type	Circuit	Remarks
G		• CMOS level output • Hysteresis input • Incorporates open-drain control
H		• CMOS level I/O
I		• CMOS level output • Hysteresis input
K		• CMOS level I/O • Analog input
L		• CMOS level I/O • Analog output • Shared with D/A outputs
M		• Incorporates pull-up resistor control (for input) Registance 50 kΩ (approx.) • CMOS level output • Hysteresis input

■ HANDLING DEVICES

1. Preventing Latch-up

Latch-up occurs in a CMOS IC if a voltage greater than V_{CC} or less than V_{SS} is applied to an input or output pin or if the voltage applied between V_{CC} and V_{SS} exceeds the rating. If latch-up occurs, the power supply current increases rapidly resulting in thermal damage to circuit elements. Therefore, ensure that maximum ratings are not exceeded in circuit operation.
For the same reason, also ensure that the analog supply voltage does not exceed the digital supply voltage.

2. Treatment of Unused Pins

Leaving unused input pins unconnected can cause misoperation. Always pull-up or pull-down unused pins.

3. External Reset Input

To reliably reset the controller by inputting an “L” level to the $\overline{RST}$ pin, ensure that the “L” level is applied for at least five machine cycles. Take particular note when using an external clock input.

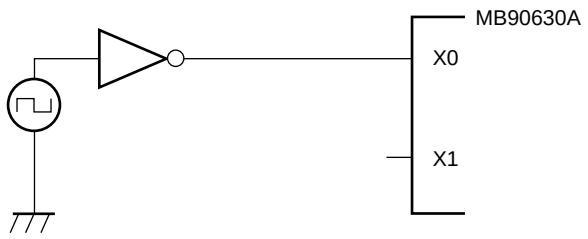
4. V_{CC} and V_{SS} Pins

Ensure that all V_{CC} pins are at the same voltage. The same applies for the V_{SS} pins.

5. Precautions when Using an External Clock

Drive the X0 pin only when using an external clock.

- **Using an external clock**



6. A/D Converter Power Supply and the Turn-on Sequence for Analog Inputs

Always turn off the A/D converter power supply (AV_{CC}, AVRH, AVRL) and analog inputs (AN0 to AN7) before turning off the digital power supply (V_{CC}).
When turning the power on or off, ensure that AVRH does not exceed AV_{CC}.
Also, when using the analog input pins as input ports, ensure that the input voltage does not exceed AV_{CC}.

7. Program Mode

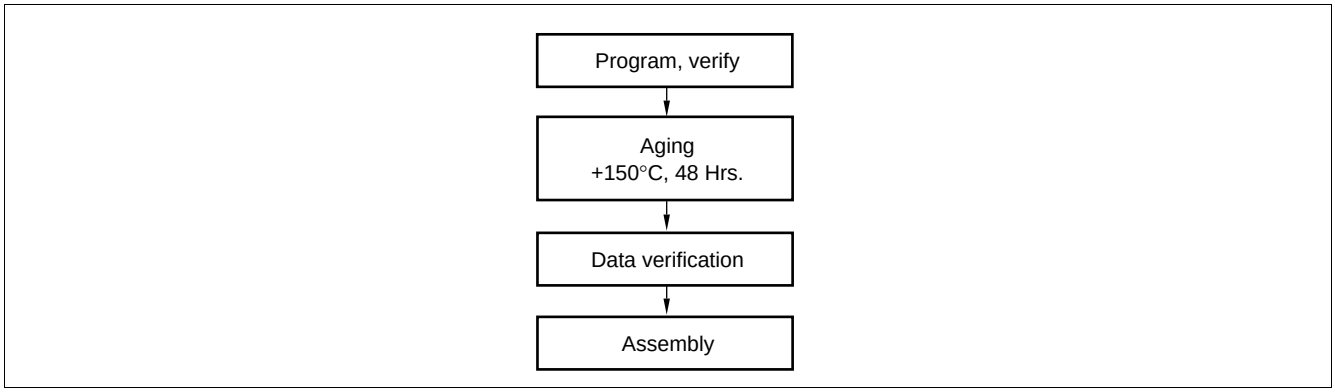
All bits (64 K × 16 bits) in the MB90P634A are “1” on delivery from Fujitsu or after erasing. To write data, selectively program the desired bits to “0”. The value “1” cannot be written electrically.

8. Recommended Screening Conditions

High temperature aging is recommended as the pre-assembly screening procedure.

9. Programming Yield

All bits cannot be programmed at Fujitsu shipping test to a blanked OTPROM microcomputer, due to its nature. For this reason, a programming yield of 100% cannot be assured at all times.



10. Power Supply Voltage Fluctuations

Although V_{CC} power supply voltage is assured to operate within the rated range, a rapid fluctuation of the voltage could cause malfunctions, even if it occurs within the rated range. Stabilizing voltage supplied to the IC is therefore important. As stabilization guidelines, it is recommended to control power so that V_{CC} ripple fluctuations (P-P value) will be less than 10% of the standard V_{CC} value at the commercial frequency (50 to 60 Hz) and the transient fluctuation rate will be less than 0.1 V/ms at the time of 2 momentary fluctuation such as when power is switched.

■ PROGRAMMING THE EPROM IN THE MB90P634A

In EPROM mode, the MB90P634A function as MBM27C1000 equivalents. By using a dedicated adapter socket, the devices can be programmed using a standard EPROM programmer.

1. Pin Assignment in EPROM Mode

- **Pins compatible with the MBM27C1000**

MBM27C1000		MB90P634A	
Pin number	**Pin name**	**Pin number**	**Pin name**
1	V_{PP}	49	MD2 (V_{PP})
2	OE	10	P32
3	A15	98	P17
4	A12	95	P14
5	A07	6	P27
6	A06	5	P26
7	A05	4	P25
8	A04	3	P24
9	A03	2	P23
10	A02	1	P22
11	A01	100	P21
12	A00	99	P20
13	D00	83	P00
14	D01	84	P01
15	D02	85	P02
16	GND	—	—
32	V_{CC}	—	—
31	PGM	11	P33
30	NC	—	—
29	A14	97	P16
28	A13	96	P15
27	A08	91	P10
26	A09	92	P11
25	A11	94	P13
24	A16	7	P30
23	A10	93	P12
22	CE	8	P31
21	D07	90	P07

(Continued)

(Continued)

MBM27C1000		MB90P634A	
Pin number	**Pin name**	**Pin number**	**Pin name**
20	D06	89	P06
19	D05	88	P05
18	D04	87	P04
17	D03	86	P03

- **Power supply and GND connection pins**

Type	Pin number	Pin name
Power supply (V_{CC})	28 50 21, 82	DVRH $\overline{HST}$ V_{CC}
GND	9 34 35 40 29 75 79 12 13 14	V_{SS} AVRL AV_{SS} V_{SS} DV_{SS} $\overline{RST}$ V_{SS} P34 P35 P36

- **Pins other than MBM27C1000-compatible pins**

Pin number	Pin name	Treatment
47 48 80	MD0 MD1 X0	Pull-up (4.7 kΩ)
81	X1	OPEN
15 16 to 20 22 to 24 25 to 27 30 31 36 to 39 41 to 44 45 46 51 to 56 57 to 64 65 to 72 73 74 76 77 78	P37 P40 to P44 P45 to P47 P70 to P72 P73 P74 P50 to P53 P54 to P57 P80 P81 P82 to P87 P60 to P67 P90 to P97 PA0 PA1 PA2 PA3 PA4	Connect pull-up resistors of approximately 1 MΩ to each pin

2. EPROM Programmer Socket Adapter

Part no.	Package	Compatible socket adapter Sun Hayato Co., Ltd.
MB90P634APFV	SQFP-100	ROM-100SQF-32DP-16L

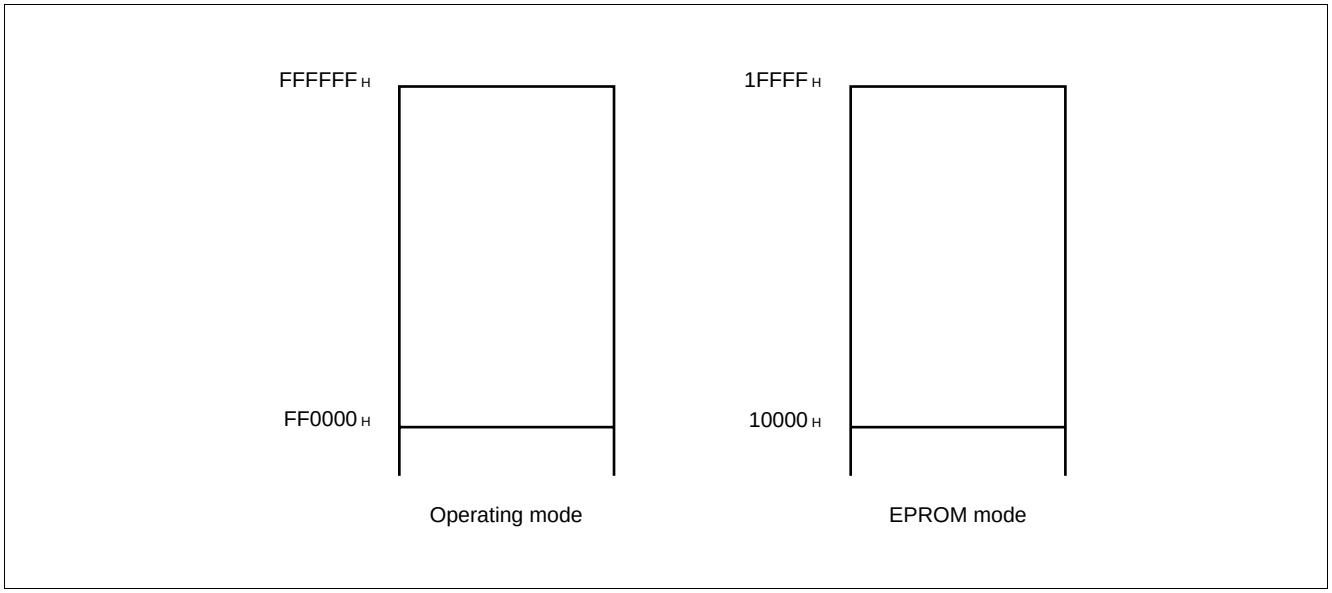
Inquiry: Sun Hayato Co., Ltd.: TEL: (81)-3-3986-0403
FAX: (81)-3-5396-9106

3. Programming Procedure

(1) Set the EPROM programmer for a MBM27C1000.

(2) Load the program data between 10000_H and $1FFFF_H$ in the EPROM programmer.

In the MB90P634A, ROM addresses $FFFFFF_H$ to $FF0000_H$ in operating mode correspond to addresses $1FFFF_H$ to 10000_H in EPROM mode.



(3) Set the MB90P634A, in the adapter socket and connect the adapter socket to the EPROM programmer. Take care to correctly align the device with the adapter.

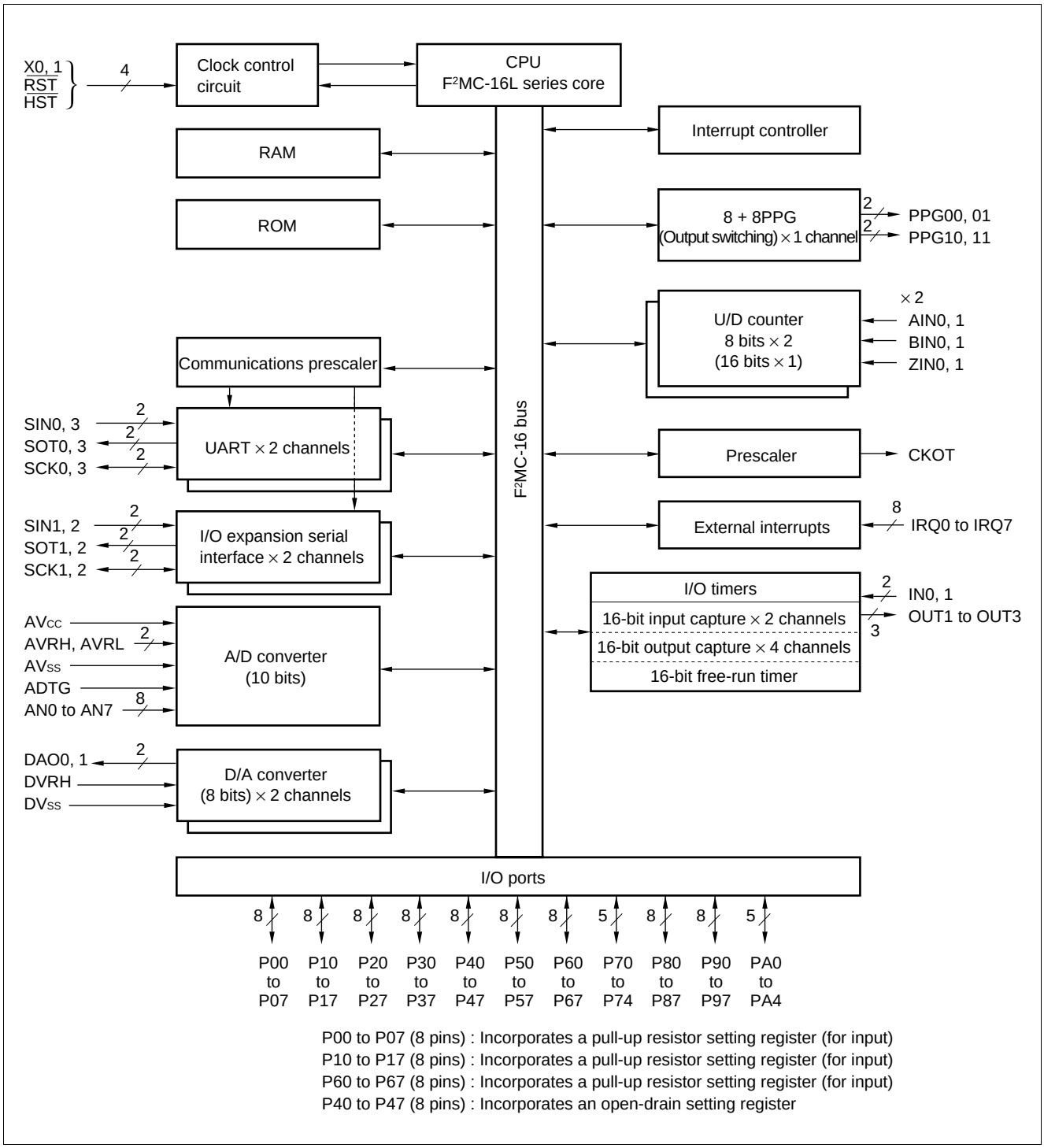
(4) Perform programming.

(5) If programming cannot be performed successfully, connect a 0.1 µF or similar capacitor between V_{CC} and GND and between V_{PP} and GND.

Note: As mask ROM products (MB90632A, 634A) do not support EPROM mode, data cannot be read using an EPROM programmer. Performing a blank check for other than the above addresses results in either non-EPROM addresses being read or the blank check being unable to be performed.

MB90630A Series

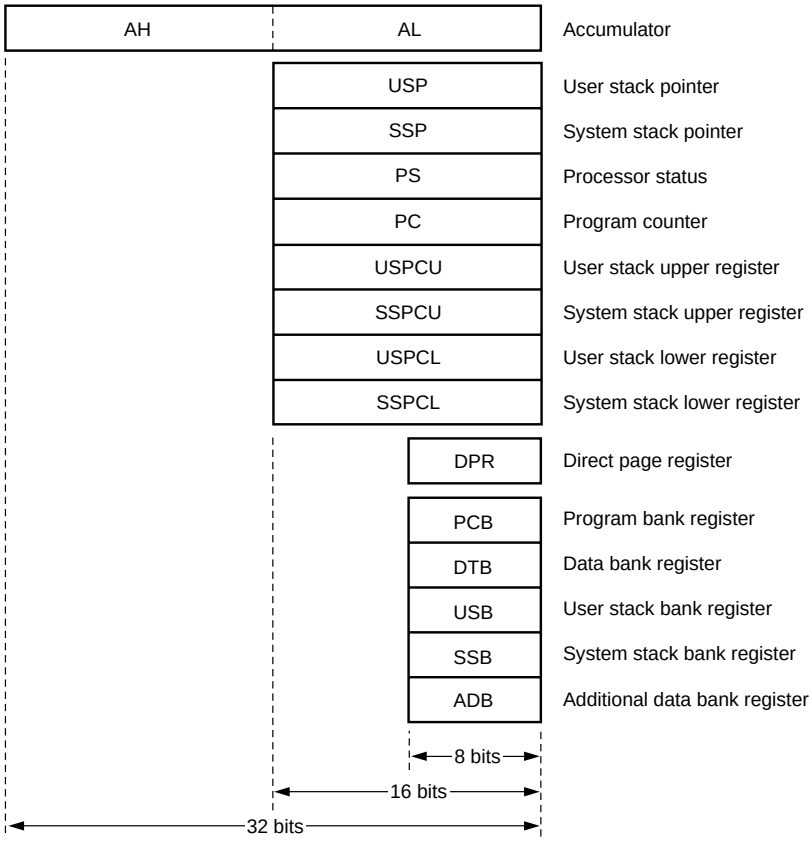
■ BLOCK DIAGRAM



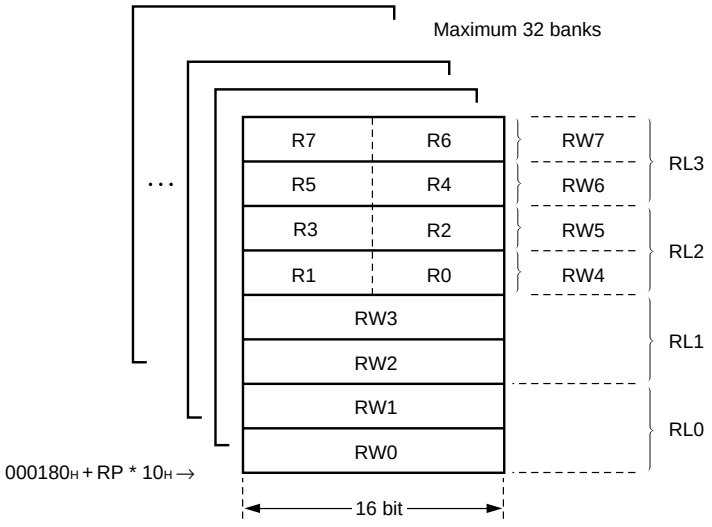
P00 to P07 (8 pins) : Incorporates a pull-up resistor setting register (for input)
P10 to P17 (8 pins) : Incorporates a pull-up resistor setting register (for input)
P60 to P67 (8 pins) : Incorporates a pull-up resistor setting register (for input)
P40 to P47 (8 pins) : Incorporates an open-drain setting register

■ F²MC-16L CPU PROGRAMMING MODEL

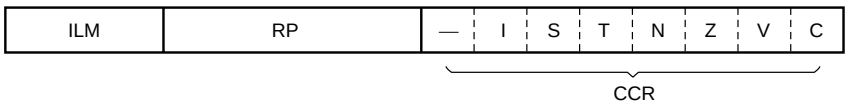
- **Dedicated Registers**



- **General-purpose Registers**

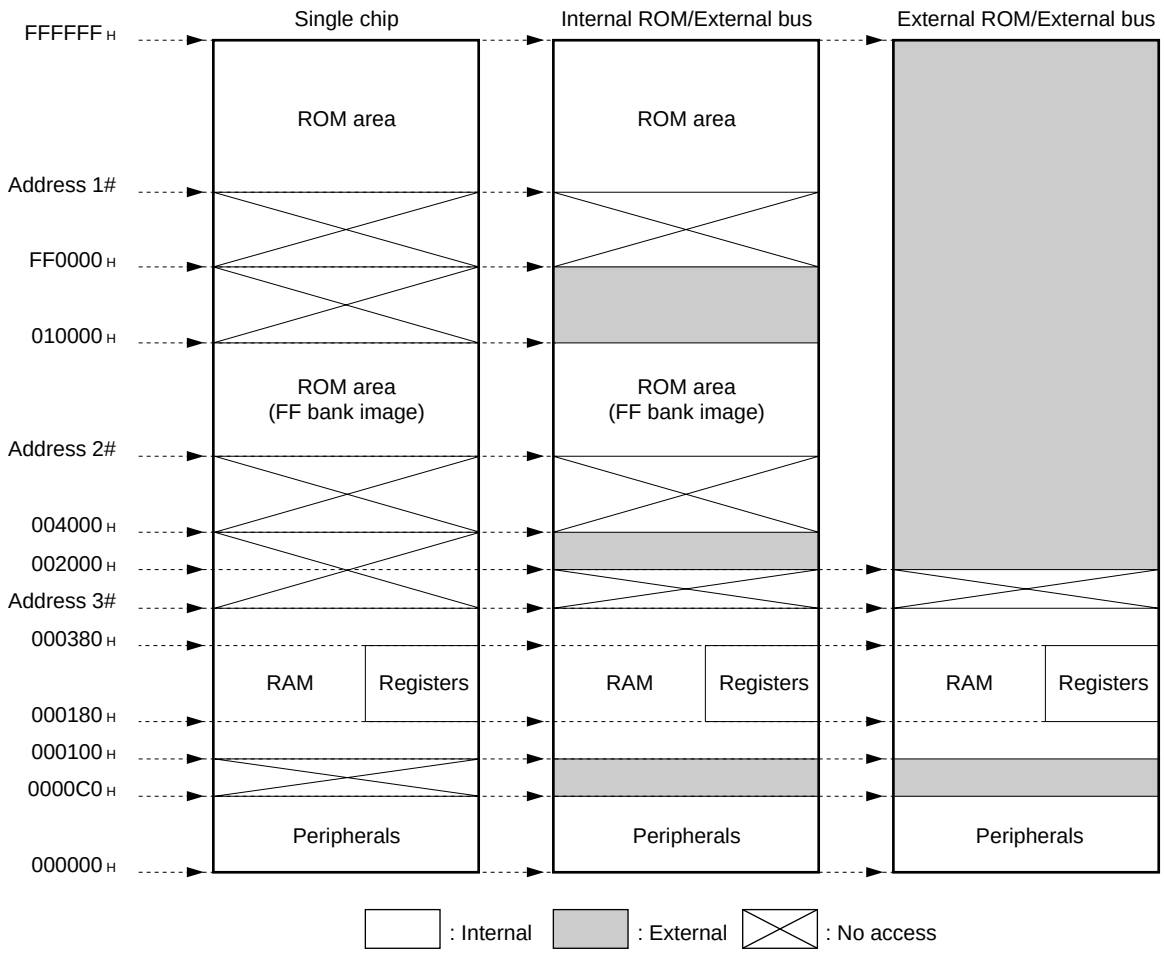


- **Processor Status (PS)**



MB90630A Series

■ MEMORY MAP



Type	Address #1	Address #2	Address #3
MB90632A	FF8000H	008000H	000500H
MB90634A	FF0000H	004000H	000900H
MB90P634A	FF0000H	004000H	000D00H

■ I/O MAP

Address	Register	Register name	Access	Resource	Initial value
00H	Port 0 data register	PDR0	R/W	Port 0	XXXXXXXX
01H	Port 1 data register	PDR1	R/W	Port 1	XXXXXXXX
02H	Port 2 data register	PDR2	R/W	Port 2	XXXXXXXX
03H	Port 3 data register	PDR3	R/W	Port 3	XXXXXXXX
04H	Port 4 data register	PDR4	R/W	Port 4	XXXXXXXX
05H	Port 5 data register	PDR5	R/W	Port 5	XXXXXXXX
06H	Port 6 data register	PDR6	R/W	Port 6	XXXXXXXX
07H	Port 7 data register	PDR7	R/W	Port 7	---XXXXX
08H	Port 8 data register	PDR8	R/W	Port 8	XXXXXXXX
09H	Port 9 data register	PDR9	R/W	Port 9	XXXXXXXX
0AH	Port A data register	PDRA	R/W	Port A	---XXXXX
0B to 0FH	Reserved area				
10H	Port 0 direction register	DDR0	R/W	Port 0	00000000
11H	Port 1 direction register	DDR1	R/W	Port 1	00000000
12H	Port 2 direction register	DDR2	R/W	Port 2	00000000
13H	Port 3 direction register	DDR3	R/W	Port 3	00000000
14H	Port 4 direction register	DDR4	R/W	Port 4	00000000
15H	Port 5 direction register	DDR5	R/W	Port 5	00000000
16H	Port 6 direction register	DDR6	R/W	Port 6	00000000
17H	Port 7 direction register	DDR7	R/W	Port 7	---00000
18H	Port 8 direction register	DDR8	R/W	Port 8	00000000
19H	Port 9 direction register	DDR9	R/W	Port 9	00000000
1AH	Port A direction register	DDRA	R/W	Port A	---00000
1BH	Port 4 pin register	ODR4	R/W	Port 4	00000000
1CH	Port 0 resistance register	RDR0	R/W	Port 0	00000000
1DH	Port 1 resistance register	RDR1	R/W	Port 1	00000000
1EH	Port 6 resistance register	RDR6	R/W	Port 6	00000000
1FH	Analog input enable register	ADER	R/W	Port 5, A/D	11111111
20H	Serial mode register 0	SMR0	R/W	UART0	00000000
21H	Serial control register 0	SCR0	R/W		00000100
22H	Serial input register/ Serial output register 0	SIDR/ SODR0	R/W		XXXXXXXX

(Continued)

Address	Register	Register name	Access	Resource	Initial value
23H	Serial status register 0	SSR0	R/W	UART0	00001–00
24H	Serial mode control status register 0	SMCS0	R/W	I/O expansion serial interface 0	––––0000
25H	Serial mode control status register 0	SMCS0	R/W		00000010
26H	Serial data register 0	SDR0	R/W		XXXXXXXX
27H	Clock division control register	CDCR	R/W	Communications prescaler	0–––1111
28H	Serial mode control status register 1	SMCS1	R/W	I/O expansion serial interface 1	––––0000
29H	Serial mode control status register 1	SMCS1	R/W		00000010
2AH	Serial data register 1	SDR1	R/W		XXXXXXXX
2B to 2FH	Reserved area				
30H	Interrupt/DTP enable register	ENIR	R/W	DTP/External interrupts	00000000
31H	Interrupt/DTP source register	EIRR	R/W		XXXXXXXX
32H	Request level setting register	ELVR	R/W		00000000
33H					00000000
34 to 35H	Reserved area				
36H	Control status register	ADCS1	R/W	A/D converter	00000000
37H		ADCS2			00000000
38H	Data register	ADCR1	R		XXXXXXXX
39H		ADCR2			XXXXXXXX
3AH	D/A converter data register 0	DAT0	R/W	D/A converter	XXXXXXXX
3BH	D/A converter data register 1	DAT1	R/W		XXXXXXXX
3CH	D/A control register 0	DACR0	R/W		–––––––0
3DH	D/A control register 1	DACR1	R/W		–––––––0
3EH	Clock control register	CLKR	R/W	CKOT output	–––––000
3FH	Reserved area				
40H	Reload register L (channel 0)	PRLL0	R/W	8/16 bit PPG	XXXXXXXX
41H	Reload register H (channel 0)	PRLH0	R/W		XXXXXXXX
42H	Reload register L (channel 1)	PRLL1	R/W		XXXXXXXX
43H	Reload register H (channel 1)	PRLH1	R/W		XXXXXXXX
44H	PPG0 operation mode control register	PPGC0	R/W		0X000XX1
45H	PPG1 operation mode control register	PPGC1	R/W		0X000001
46H	PPG0, 1 output control register	PPGOE	R/W		00000000
47 to 4FH	Reserved area				
50H	Lower compare register channel 0	OCCP0	R/W	16-bit I/O timer output compare (channel 0 to 3)	XXXXXXXX

(Continued)

Address	Register	Register name	Access	Resource	Initial value
51H	Upper compare register channel 0	OCCP0	R/W	16-bit I/O timer Output compare (channel 0 to 3)	XXXXXXXX
52H	Lower compare register channel 1	OCCP1	R/W		XXXXXXXX
53H	Upper compare register channel 1				XXXXXXXX
54H	Lower compare register channel 2	OCCP2	R/W		XXXXXXXX
55H	Upper compare register channel 2				XXXXXXXX
56H	Lower compare register channel 3	OCCP3	R/W		XXXXXXXX
57H	Upper compare register channel 3				XXXXXXXX
58H	Compare control status register channel 0	OCS0	R/W		---00000
59H	Compare control status register channel 1	OCS1	R/W		0000--00
5AH	Compare control status register channel 2	OCS2	R/W		---00000
5BH	Compare control status register channel 3	OCS3	R/W		0000--00
5C to 5FH	Reserved area				
60H	Lower input capture register channel 0	IPCP0	R	16-bit I/O timer Input capture (channel 0, 1)	XXXXXXXX
61H	Upper input capture register channel 0		R		XXXXXXXX
62H	Lower input capture register channel 1	IPCP1	R		XXXXXXXX
63H	Upper input capture register channel 1		R		XXXXXXXX
64H	Input capture control status register	ICS	R/W		00000000
65H	Reserved area	—	—		--------
66H	Lower timer data register	TCDTL	R/W	16-bit I/O timer Free-run timer (channel 0, 1)	00000000
67H	Upper timer data register	TCDTH	R/W		00000000
68H	Timer control status register	TCCS	R/W		00000000
69 to 6FH	Reserved area				
70H	Up/down count register channel 0	UDCR0	R	8/16-bit up/down timer/counter	00000000
71H	Up/down count register channel 1	UDCR1			00000000
72H	Reload compare register channel 0	RCR0	W		00000000
73H	Reload compare register channel 1	RCR1			00000000
74H	Counter status register channel 0	CSR0	R/W		00000000
75H	Reserved area	—	—		--------
76H	Counter control register channel 0	CCRL0	R/W		-0000000
77H		CCRH0			00000000
78H	Counter status register channel 1	CSR1	R/W		00000000
79H	Reserved area	—	—		--------
7AH	Counter control register channel 1	CCRL1	R/W		-0000000

(Continued)

Address	Register	Register name	Access	Resource	Initial value
7B$_H$	Counter control register channel 1	CCRH1	R/W	8/16-bit up/down timer/counter	–0000000
7C to 87$_H$	Reserved area				
88$_H$	Serial mode register 1	SMR1	R/W	UART1	00000000
89$_H$	Serial control register 1	SCR1	R/W		00000100
8A$_H$	Serial input register 1/serial output register 1	SIDR1/ SODR1	R/W		XXXXXXXX
8B$_H$	Serial status register 1	SSR1	R/W		00001–00
8C to 9E$_H$	Reserved area (Accessing 90$_H$ to 9E$_H$ is prohibited.)				
9F$_H$	Delayed interrupt generation/ clear register	DIRR	R/W	Delayed interrupt generation module	–––––––0
A0$_H$	Low-power consumption mode register	LPMCR	R/W	Low-power consumption	00011000
A1$_H$	Clock selection register	CKSCR	R/W	Low-power consumption	11001100
A2 to A4$_H$	Reserved area				
A5$_H$	Auto-ready function selection register	ARSR	W	External pins	0011––00
A6$_H$	External address output control register	HACR	W	External pins	––––0000
A7$_H$	Bus control signal selection register	ECSR	W	External pins	0000*00–
A8$_H$	Watchdog timer control register	WDTC	R/W	Watchdog timer	XXXXX111
A9$_H$	Timebase timer control register	TBTC	R/W	Timebase timer	1––00100
AA to AF$_H$	Reserved area				
B0$_H$	Interrupt control register 00	ICR00	R/W	Interrupt controller	00000111
B1$_H$	Interrupt control register 01	ICR01	R/W		00000111
B2$_H$	Interrupt control register 02	ICR02	R/W		00000111
B3$_H$	Interrupt control register 03	ICR03	R/W		00000111
B4$_H$	Interrupt control register 04	ICR04	R/W		00000111
B5$_H$	Interrupt control register 05	ICR05	R/W		00000111
B6$_H$	Interrupt control register 06	ICR06	R/W		00000111
B7$_H$	Interrupt control register 07	ICR07	R/W		00000111
B8$_H$	Interrupt control register 08	ICR08	R/W		00000111
B9$_H$	Interrupt control register 09	ICR09	R/W		00000111
BA$_H$	Interrupt control register 10	ICR10	R/W		00000111
BB$_H$	Interrupt control register 11	ICR11	R/W		00000111
BC$_H$	Interrupt control register 12	ICR12	R/W		00000111
BD$_H$	Interrupt control register 13	ICR13	R/W		00000111

(Continued)

(Continued)

Address	Register	Register name	Access	Resource	Initial value
BE$_H$	Interrupt control register 14	ICR14	R/W	Interrupt controller	00000111
BF$_H$	Interrupt control register 15	ICR15	R/W		00000111
C0 to FF$_H$	Reserved area	—	—	—	—

Initial values
0: The initial value of this bit is “0”.
1: The initial value of this bit is “1”.
* : The initial value of this bit is “0” or “1”.
X: The initial value of this bit is undefined.
–: This bit is not used. The initial value is undefined.

Note: Areas below address 0000FF$_H$ not listed in the table are reserved areas. These addresses are accessed by internal access. No access signals are output on the external bus.

■ INTERRUPT VECTOR AND INTERRUPT CONTROL REGISTER ASSIGNMENTS TO INTERRUPT SOURCES

Interrupt source	I²OS support	Interrupt vector		Interrupt control register	
		Number	Address	ICR	Address
Reset	×	#08	FFFFDC$_H$	—	—
INT 9 instruction	×	#09	FFFFD8$_H$	—	—
Exception	×	#10	FFFFD4$_H$	—	—
A/D converter	○	#11	FFFFD0$_H$	ICR00	0000B0$_H$
DTP 0 (External interrupt 0)	○	#13	FFFFC8$_H$	ICR01	0000B1$_H$
16-bit free-run timer (I/O timer) overflow	○	#14	FFFFC4$_H$		
I/O expansion serial 1	○	#15	FFFFC0$_H$	ICR02	0000B2$_H$
DTP 1 (External interrupt 1)	○	#16	FFFFBC$_H$		
I/O expansion serial 2	○	#17	FFFFB8$_H$	ICR03	0000B3$_H$
DTP 2 (External interrupt 2)	○	#18	FFFFB4$_H$		
DTP 3 (External interrupt 3)	○	#19	FFFFB0$_H$	ICR04	0000B4$_H$
8/16-bit PPG 0 counter borrow	○	#20	FFFFAC$_H$		
8/16-bit U/D counter 0 compare	○	#21	FFFFA8$_H$	ICR05	0000B5$_H$
8/16-bit U/D counter 0 underflow/ overflow, up/down invert	○	#22	FFFFA4$_H$		
8/16-bit PPG 1 counter borrow	○	#23	FFFFA0$_H$	ICR06	0000B6$_H$
DTP 4/5 (External interrupt 4/5)	○	#24	FFFF9C$_H$		
Output compare (channel 2) match (I/O timer)	○	#25	FFFF98$_H$	ICR07	0000B7$_H$
Output compare (channel 3) match (I/O timer)	○	#26	FFFF94$_H$		
DTP 6 (External interrupt 6)	○	#28	FFFF8C$_H$	ICR08	0000B8$_H$
8/16-bit U/D counter 1 compare	○	#29	FFFF88$_H$	ICR09	0000B9$_H$
8/16-bit U/D counter 1 underflow/ overflow, up/down invert	○	#30	FFFF84$_H$		
Input capture (channel 0) read (I/O timer)	○	#31	FFFF80$_H$	ICR10	0000BA$_H$
Input capture (channel 1) read (I/O timer)	○	#32	FFFF7C$_H$		
Output compare (channel 0) match (I/O timer)	○	#33	FFFF78$_H$	ICR11	0000BB$_H$
Output compare (channel 1) match (I/O timer)	○	#34	FFFF74$_H$		
DTP 7 (External interrupt 7)	○	#36	FFFF6C$_H$	ICR12	0000BC$_H$
UART0 receive complete	◎	#37	FFFF68$_H$	ICR13	0000BD$_H$
UART1 receive complete	◎	#38	FFFF64$_H$		
UART0 transmit complete	◎	#39	FFFF60$_H$	ICR14	0000BE$_H$
UART1 transmit complete	◎	#40	FFFF5C$_H$		
Reserved	×	#41	FFFF58$_H$	ICR15	0000BF$_H$
Delayed interrupt	×	#42	FFFF54$_H$		

○: Indicates that the interrupt request flag is cleared by the I²OS interrupt clear signal (no stop request).
◎: Indicates that the interrupt request flag is cleared by the I²OS interrupt clear signal (stop request present).
×: Indicates that the interrupt request flag is not cleared by the I²OS interrupt clear signal.
Note: For resources in which two interrupt sources share the same interrupt number, the I²OS interrupt clear signal clears both interrupt request flags.

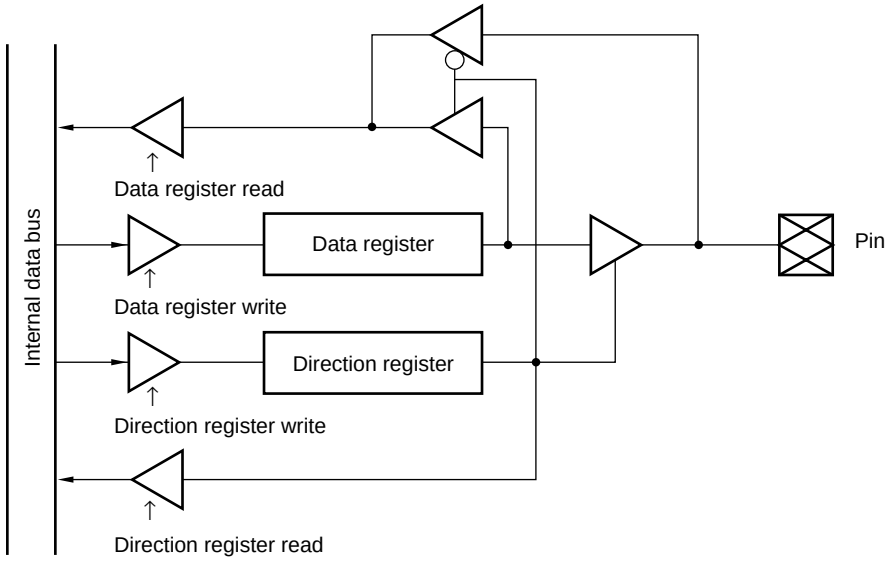
■ PERIPHERAL RESOURCES

1. Parallel Ports

(1) I/O Ports

Each port pin can be specified as either an input or output by its corresponding direction register when the pin is not set for use by a peripheral. When a port is set as an input, reading the data register always reads the value corresponding to the pin level. When a port is set as an output, reading the data register reads the data register latch value. The same applies when reading using a read-modify-write instruction.
When used as control outputs, reading the data register reads the control output value, irrespective of the direction register value.
Note that if a read-modify-write instruction (set bit or similar instruction) is used to set output data in the data register before switching a pin from input to output, the instruction reads the input level at the pin and not the data register latch value.

- **Block Diagram**



(2) Register Configuration

bit	15/7	14/6	13/5	12/4	11/3	10/2	9/1	8/0	
Address: 000000H	P07	P06	P05	P04	P03	P02	P01	P00	Port 0 data register (PDR0)
Address: 000001H	P17	P16	P15	P14	P13	P12	P11	P10	Port 1 data register (PDR1)
Address: 000002H	P27	P26	P25	P24	P23	P22	P21	P20	Port 2 data register (PDR2)
Address: 000003H	P37	P36	P35	P34	P33	P32	P31	P30	Port 3 data register (PDR3)
Address: 000004H	P47	P46	P45	P44	P43	P42	P41	P40	Port 4 data register (PDR4)
Address: 000005H	P57	P56	P55	P54	P53	P52	P51	P50	Port 5 data register (PDR5)
Address: 000006H	P67	P66	P65	P64	P63	P62	P61	P60	Port 6 data register (PDR6)
Address: 000007H	—	—	—	P74	P73	P72	P71	P70	Port 7 data register (PDR7)
Address: 000008H	P87	P86	P85	P84	P83	P82	P81	P80	Port 8 data register (PDR8)
Address: 000009H	P97	P96	P95	P94	P93	P92	P91	P90	Port 9 data register (PDR9)
Address: 00000AH	—	—	—	PA4	PA3	PA2	PA1	PA0	Port A data register (PDRA)

bit	15/7	14/6	13/5	12/4	11/3	10/2	9/1	8/0	
Address: 000010H	D07	D06	D05	D04	D03	D02	D01	D00	Port 0 direction register (DDR0)
Address: 000011H	D17	D16	D15	D14	D13	D12	D11	D10	Port 1 direction register (DDR1)
Address: 000012H	D27	D26	D25	D24	D23	D22	D21	D20	Port 2 direction register (DDR2)
Address: 000013H	D37	D36	D35	D34	D33	D32	D31	D30	Port 3 direction register (DDR3)
Address: 000014H	D47	D46	D45	D44	D43	D42	D41	D40	Port 4 direction register (DDR4)
Address: 000015H	D57	D56	D55	D54	D53	D52	D51	D50	Port 5 direction register (DDR5)
Address: 000016H	D67	D66	D65	D64	D63	D62	D61	D60	Port 6 direction register (DDR6)
Address: 000017H	—	—	—	D74	D73	D72	D71	D70	Port 7 direction register (DDR7)
Address: 000018H	D87	D86	D85	D84	D83	D82	D81	D80	Port 8 direction register (DDR8)
Address: 000019H	D97	D96	D95	D94	D93	D92	D91	D90	Port 9 direction register (DDR9)
Address: 00001AH	—	—	—	DA4	DA3	DA2	DA1	DA0	Port A direction register (DDRA)

bit	15	14	13	12	11	10	9	8	
Address: 00001BH	OD47	OD46	OD45	OD44	OD43	OD42	OD41	OD40	Port 4 pin register (ODR4)

bit	15/7	14/6	13/5	12/4	11/3	10/2	9/1	8/0	
Address: 00001CH	RD07	RD06	RD05	RD04	RD03	RD02	RD01	RD00	Port 0 resistor register (RDR0)
Address: 00001DH	RD17	RD16	RD15	RD14	RD13	RD12	RD11	RD10	Port 1 resistor register (RDR1)
Address: 00001EH	RD67	RD66	RD65	RD64	RD63	RD62	RD61	RD60	Port 6 resistor register (RDR6)

bit	15	14	13	12	11	10	9	8	
Address: 00001FH	ADE7	ADE6	ADE5	ADE4	ADE3	ADE2	ADE1	ADE0	Port 5 analog input enable register (ADER)

(3) Register Details

- **Port Data Registers**

	bit	7	6	5	4	3	2	1	0	Initial value	Access
PDR0 Address: 000000H		P07	P06	P05	P04	P03	P02	P01	P00	Undefined	R/W*

	bit	15	14	13	12	11	10	9	8		
PDR1 Address: 000001H		P17	P16	P15	P14	P13	P12	P11	P10	Undefined	R/W*

	bit	7	6	5	4	3	2	1	0		
PDR2 Address: 000002H		P27	P26	P25	P24	P23	P22	P21	P20	Undefined	R/W*

	bit	15	14	13	12	11	10	9	8		
PDR3 Address: 000003H		P37	P36	P35	P34	P33	P32	P31	P30	Undefined	R/W*

	bit	7	6	5	4	3	2	1	0		
PDR4 Address: 000004H		P47	P46	P45	P44	P43	P42	P41	P40	Undefined	R/W*

	bit	15	14	13	12	11	10	9	8		
PDR5 Address: 000005H		P57	P56	P55	P54	P53	P52	P51	P50	Undefined	R/W*

	bit	7	6	5	4	3	2	1	0		
PDR6 Address: 000006H		P67	P66	P65	P64	P63	P62	P61	P60	Undefined	R/W*

	bit	15	14	13	12	11	10	9	8		
PDR7 Address: 000007H		—	—	—	P74	P73	P72	P71	P70	Undefined	R/W*

	bit	7	6	5	4	3	2	1	0		
PDR8 Address: 000008H		P87	P86	P85	P84	P83	P82	P81	P80	Undefined	R/W*

	bit	15	14	13	12	11	10	9	8		
PDR9 Address: 000009H		P97	P96	P95	P94	P93	P92	P91	P90	Undefined	R/W*

	bit	7	6	5	4	3	2	1	0		
PDRA Address: 00000AH		—	—	—	PA4	PA3	PA2	PA1	PA0	Undefined	R/W*

* : The operation of reading or writing to I/O ports is slightly different from reading or writing to memory, as follows.
- Input mode
  Read: Reads the corresponding pin level.
  Write: Writes to the output latch.
- Output mode
  Read: Reads the value of the data register latch.
  Write: The value is output from the corresponding pin.

- **Port Direction Registers**

	bit	7	6	5	4	3	2	1	0	Initial value	Access
DDR0 Address: 000010H		D07	D06	D05	D04	D03	D02	D01	D00	00000000B	R/W
DDR1 Address: 000011H	bit	15	14	13	12	11	10	9	8		
		D17	D16	D15	D14	D13	D12	D11	D10	00000000B	R/W
DDR2 Address: 000012H	bit	7	6	5	4	3	2	1	0		
		D27	D26	D25	D24	D23	D22	D21	D20	00000000B	R/W
DDR3 Address: 000013H	bit	15	14	13	12	11	10	9	8		
		D37	D36	D35	D34	D33	D32	D31	D30	00000000B	R/W
DDR4 Address: 000014H	bit	7	6	5	4	3	2	1	0		
		D47	D46	D45	D44	D43	D42	D41	D40	00000000B	R/W
DDR5 Address: 000015H	bit	15	14	13	12	11	10	9	8		
		D57	D56	D55	D54	D53	D52	D51	D50	00000000B	R/W
DDR6 Address: 000016H	bit	7	6	5	4	3	2	1	0		
		D67	D66	D65	D64	D63	D62	D61	D60	00000000B	R/W
DDR7 Address: 000017H	bit	15	14	13	12	11	10	9	8		
		—	—	—	D74	D73	D72	D71	D70	-----000B	R/W
DDR8 Address: 000018H	bit	7	6	5	4	3	2	1	0		
		D87	D86	D85	D84	D83	D82	D81	D80	00000000B	R/W
DDR9 Address: 000019H	bit	15	14	13	12	11	10	9	8		
		D97	D96	D95	D94	D93	D92	D91	D90	00000000B	R/W
DDRA Address: 00001AH	bit	7	6	5	4	3	2	1	0		
		—	—	—	DA4	DA3	DA2	DA1	DA0	---00000B	R/W

When pins are used as ports, the register bits control the corresponding pins as follows.

0: Input mode

1: Output mode

Bits are set to “0” by a reset.

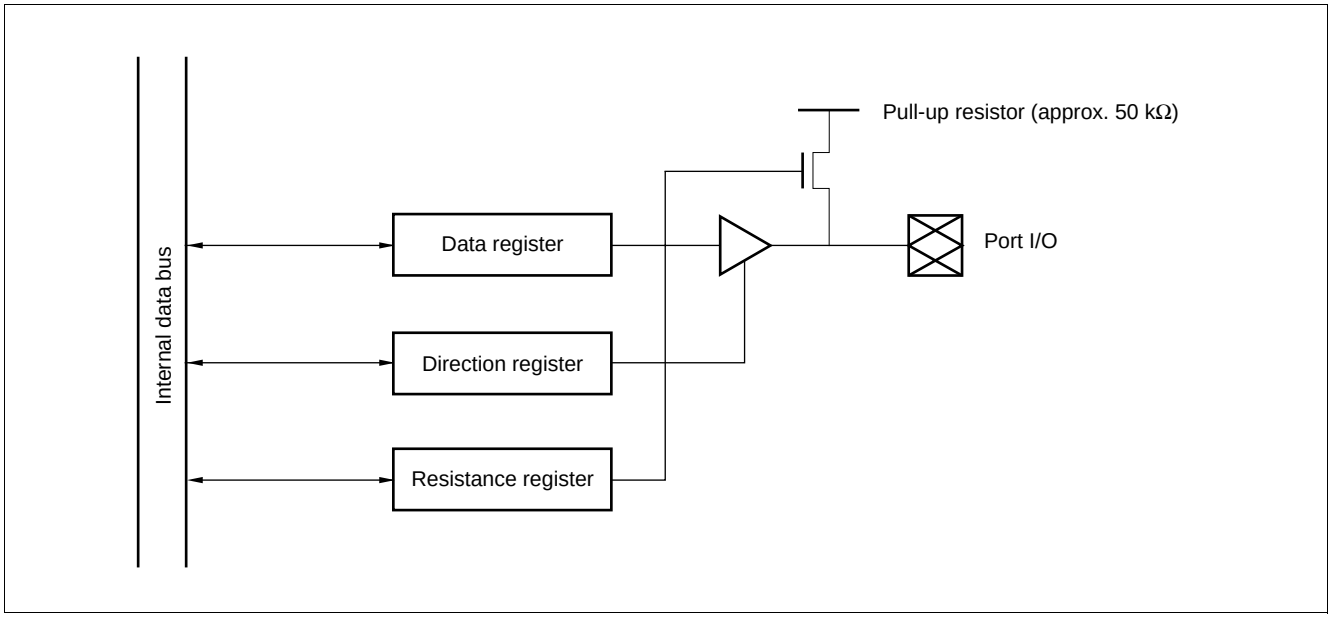
- **Port Resistance Registers**

	bit	7	6	5	4	3	2	1	0	
RDR0 Address: 00001C$_H$		RD07	RD06	RD05	RD04	RD03	RD02	RD01	RD00	Initial value 00000000$_B$

	bit	15	14	13	12	11	10	9	8	
RDR1 Address: 00001D$_H$		RD17	RD16	RD15	RD14	RD13	RD12	RD11	RD10	00000000$_B$

	bit	7	6	5	4	3	2	1	0	
RDR6 Address: 00001E$_H$		RD67	RD66	RD65	RD64	RD63	RD62	RD61	RD60	00000000$_B$

- **Block Diagram**

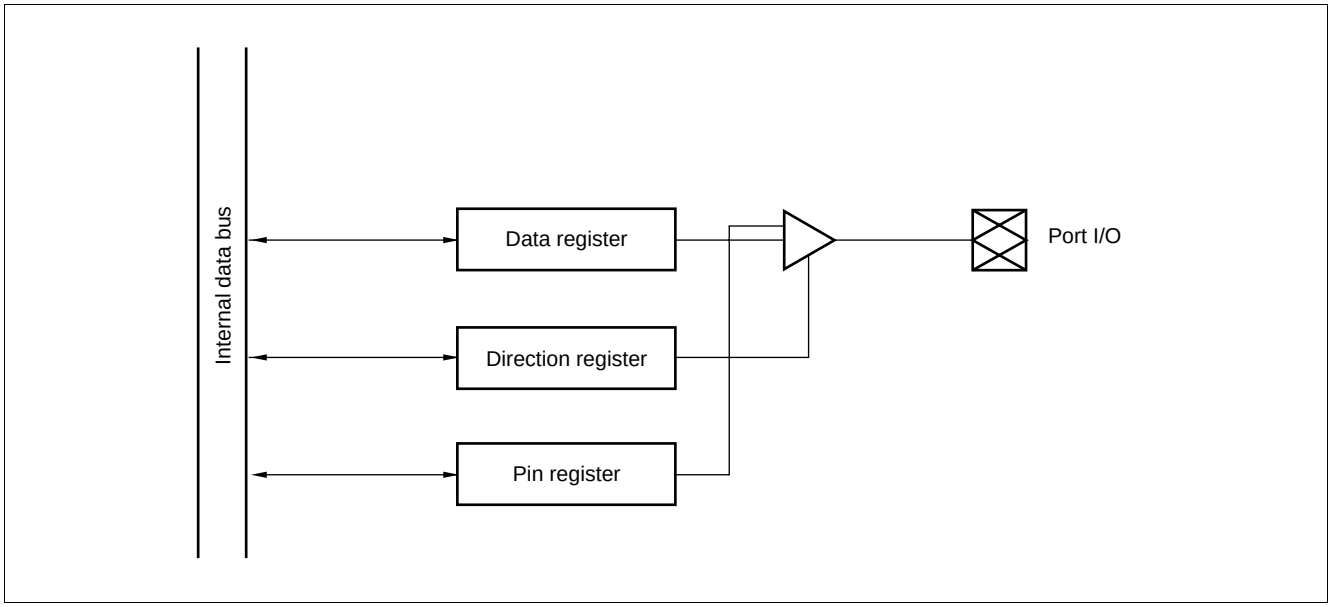


Notes: • Input resistance register R/W
Controls the pull-up resistor in input mode.
0: Pull-up resistor disconnected in input mode.
1: Pull-up resistor connected in input mode.
The setting has no meaning in output mode (pull-up resistor disconnected).
The direction register (DDR) sets input or output mode.
• The pull-up resistor is disconnected in hardware standby or stop mode (SPL = 1) (high impedance).
• This function is disabled when using an external bus. In this case, do not write to this register.

- **Port Pin Register**

bit	7	6	5	4	3	2	1	0	
ODR4 Address: 00001B$_H$	OD47	OD46	OD45	OD44	OD43	OD42	OD41	OD40	Initial value 00000000$_B$

- **Block Diagram**



Notes: • Pin register R/W
Performs open-drain control in output mode.
0: Operate as a standard output port in output mode.
1: Operate as an open-drain output port in output mode.
The setting has no meaning in input mode (output Hi-z).
The direction register (DDR) sets input or output mode
- The pull-up resistor is disconnected in hardware standby or stop mode (SPL = 1) (high impedance).
- This function is disabled when using an external bus. In this case, do not write to this register.

- **Analog Input Enable Register**

bit	15	14	13	12	11	10	9	8	
ADER Address: 00001F$_H$	ADE7	ADE6	ADE5	ADE4	ADE3	ADE2	ADE1	ADE0	Initial value 11111111$_B$
	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	

Controls each port 5 pin as follows.
0: Port input mode
1: Analog input mode
Set to “1” by a reset.

2. UART

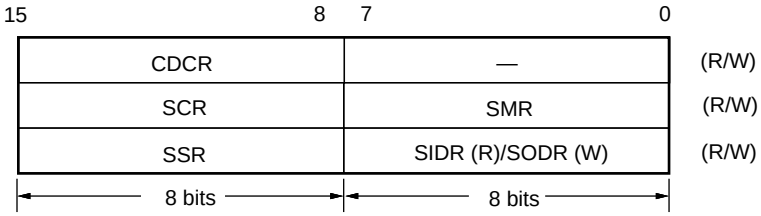
The UART is a serial I/O port that can be used for CLK asynchronous (start-stop synchronization) or CLK synchronous communications. The UART has the following features.

- Full duplex, double buffered
- Supports asynchronous (start-stop synchronization) and CLK synchronous data transfer
- Supports multi-processor mode
- Built-in dedicated baud rate generator

  Asynchronous: 9615, 31250, 4808, 2404, 1202 bps
  CLK synchronous: 1 Mbps, 500 Kbps, 250 Kbps, 125 Kbps, and 62.5 } For a 6, 8, 10, 12, or 16 MHz clock.

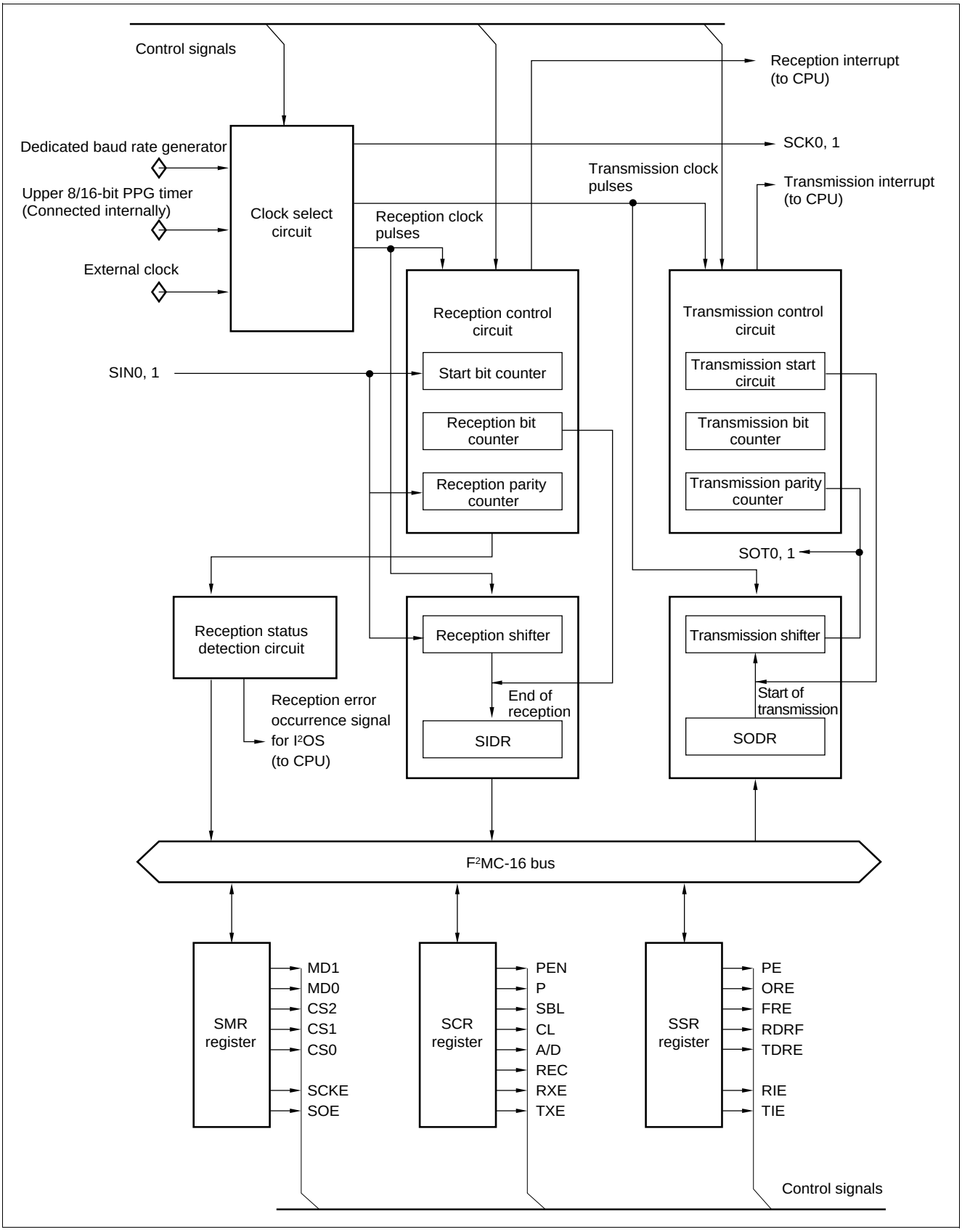
- Supports flexible baud rate setting using an external clock
- Error detect function (parity, framing, and overrun)
- NRZ type transmission signal
- Intelligent I/O service support

(1) Register Configuration



	bit	7	6	5	4	3	2	1	0	
Address: 000020H 000088H		MD1	MD0	CS2	CS1	CS0	Reserved	SCKE	SOE	Serial mode register 0, 1 (SMR0, 1)
	bit	15	14	13	12	11	10	9	8	
Address: 000021H 000089H		PEN	P	SBL	CL	A/D	REC	RXE	TXE	Serial control register 0, 1 (SCR0, 1)
	bit	7	6	5	4	3	2	1	0	
Address: 000022H 00008AH		D7	D6	D5	D4	D3	D2	D1	D0	Serial input register/ Serial output register 0, 1 (SIDR/SODR0, 1)
	bit	15	14	13	12	11	10	9	8	
Address: 000023H 00008BH		PE	ORE	FRE	RDRF	TDRE	—	RIE	TIE	Serial status register 0, 1 (SSR0, 1)
	bit	15	14	13	12	11	10	9	8	
Address: 000027H		MD	—	—	—	DIV3	DIV2	DIV1	DIV0	Clock division control register (CDCR)

(2) Block Diagram



3. I/O Expansion Serial Interface

This block consists of an 8-bit serial I/O interface that can perform clock synchronous data transfer. Either LSB-first or MSB-first data transfer can be selected.
The following two serial I/O operation modes are available.

- Internal shift clock mode: Data transfer is synchronized with the internal clock.
- External shift clock mode: Data transfer is synchronized with the clock input from the external pin (SCK). By manipulating the general-purpose port that shares the external pin (SCK), this mode also enables the data transfer operation to be driven by CPU instructions.

(1) Register Configuration

	bit	15	14	13	12	11	10	9	8	
Address: 000025H 000029H		SMD2	SMD1	SMD0	SIE	SIR	BUSY	STOP	STRT	
Address: 000024H 000028H	bit	7	6	5	4	3	2	1	0	Serial mode control status registers 0, 1 (SMCS0, 1)
		—	—	—	—	MODE	BDS	SOE	SCOE	
Address: 000026H 00002AH	bit	7	6	5	4	3	2	1	0	Serial data registers 0, 1 (SDR0, 1)
		D7	D6	D5	D4	D3	D2	D1	D0	

(2) Register Details

- **Serial Mode Control Status Register (SMCS)**

	bit	15	14	13	12	11	10	9	8	
SMCS Address: 000025H 000029H		SMD2	SMD1	SMD0	SIE	SIR	BUSY	STOP	STRT	Initial value 00000010B
		(R/W)	(R/W)	(R/W)	(R/W)	(R/W*1)	(R)	(R/W)	(R/W*2)	
SMCS Address: 000024H 000028H	bit	7	6	5	4	3	2	1	0	Initial value ----0000B
		—	—	—	—	MODE	BDS	SOE	SCOE	
						(R/W)	(R/W)	(R/W)	(R/W)	

*1: Only “0” can be written.
*2: Only “1” can be written. Reading always returns “0”.

This register controls the transfer operation mode of the serial I/O. The following describes the function of each bit.

(a) [bit 3] Serial mode selection bit (MODE)
This bit selects the conditions for starting operation from the halted state. Changing the mode during operation is prohibited.

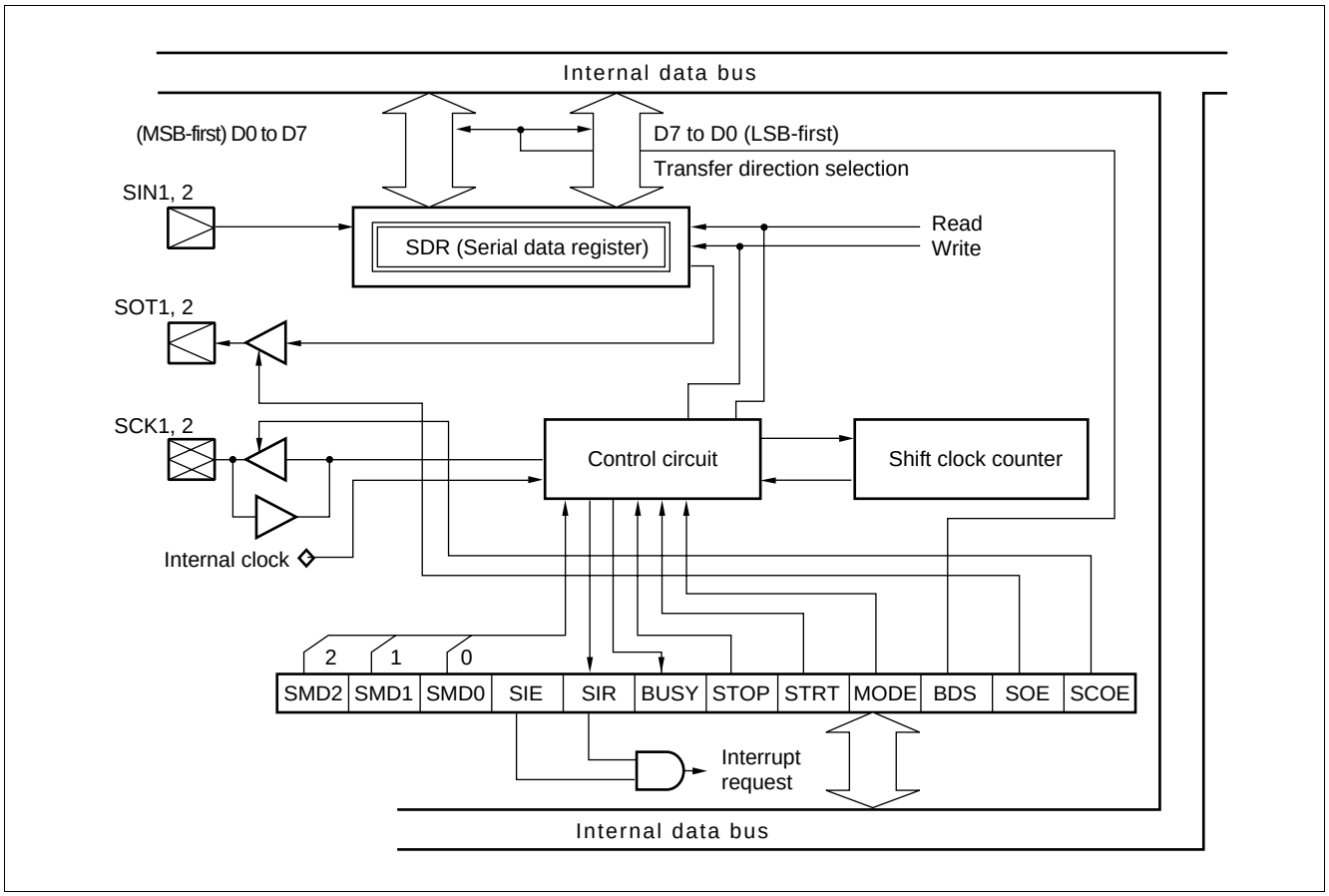
MODE	Operation
0	Start when STRT is set to “1”. [Initial value]
1	Start on reading from or writing to the serial data register.

The bit is initialized to “0” by a reset. The bit is readable and writable. Set to “1” when using the intelligent I/O service.

(b) [bit 2] Transfer direction selection bit (BDS: Bit Direction Select)
Selects as follows at the time of serial data input and output whether the data are to be transferred in the order from LSB to MSB or vice versa.

MODE	Operation
0	LSB-first [Initial value]
1	MSB-first

(3) Block Diagram



4. A/D Converter

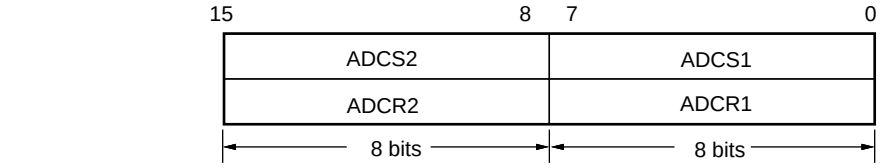
The A/D converter converts analog input voltages to digital values. The A/D converter has the following features.

- Conversion time: Minimum of 5.2 µs per channel (for a 16 MHz machine clock)
- Uses RC-type successive approximation conversion with a sample and hold circuit.
- 10-bit resolution
- Eight program-selectable analog input channels

  Single conversion mode : Selectively convert a one channel.
  Scan conversion mode : Continuously convert multiple channels. Maximum of 8 program-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.)
- An A/D conversion completion interrupt request to the CPU can be generated on the completion of A/D conversion. This interrupt can activate I²OS to transfer the result of A/D conversion to memory and is suitable for continuous operation.
- Activation by software, external trigger (falling edge), or timer (rising edge) can be selected.

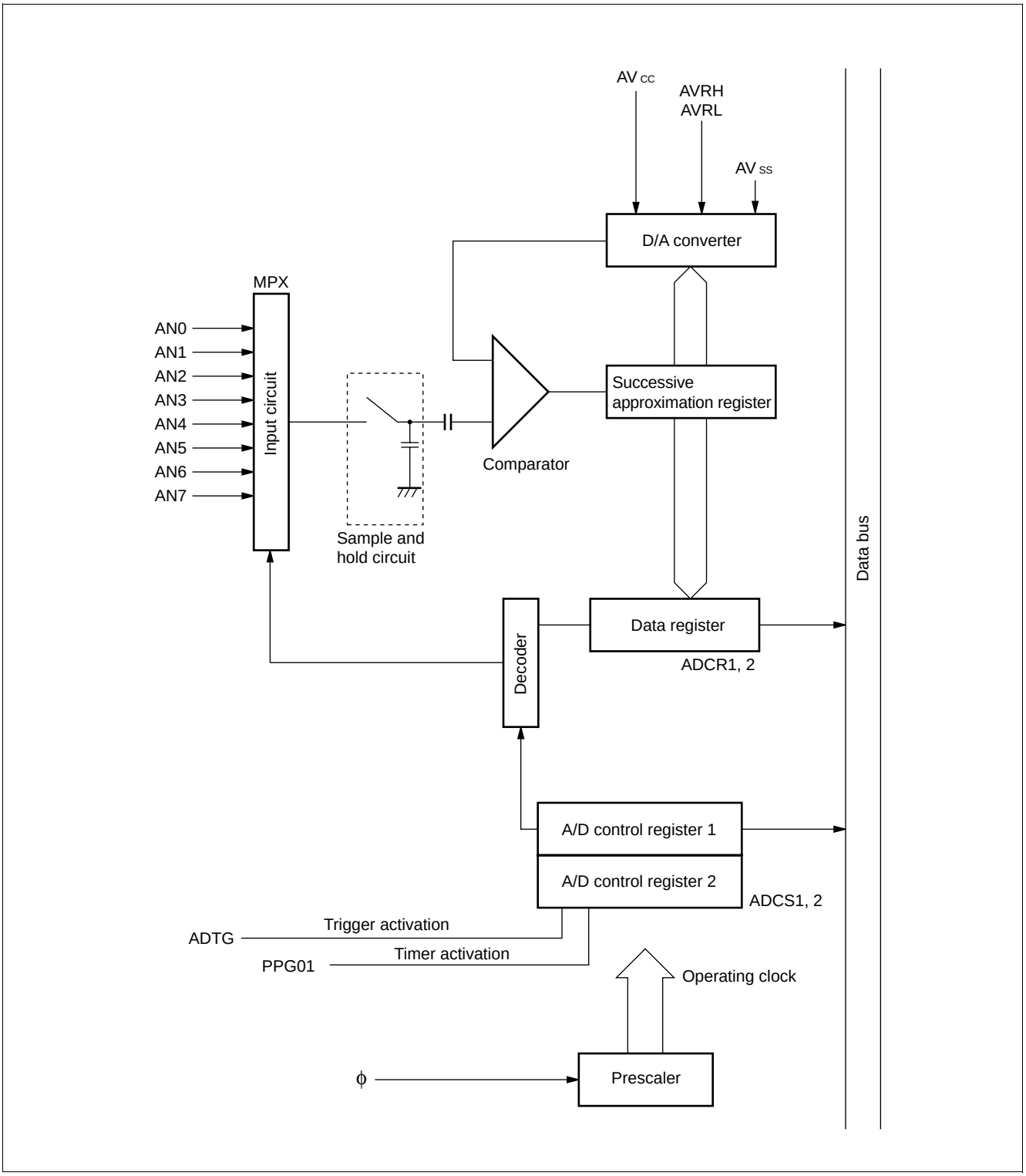
(1) Register Configuration

The A/D converter has the following registers.



	bit	7	6	5	4	3	2	1	0	
Address: 000036H		MD1	MD0	ANS2	ANS1	ANS0	ANE2	ANE1	ANE0	Control status register (ADCS1, ADCS2)
	bit	15	14	13	12	11	10	9	8	
Address: 000037H		BUSY	INT	INTE	PAUS	STS1	STS0	STRT	DA	
	bit	7	6	5	4	3	2	1	0	
Address: 000038H		7	6	5	4	3	2	1	0	Data register (ADCR1, ADCR2)
	bit	15	14	13	12	11	10	9	8	
Address: 000039H		—	—	—	—	—	—	9	8	

(2) Block Diagram



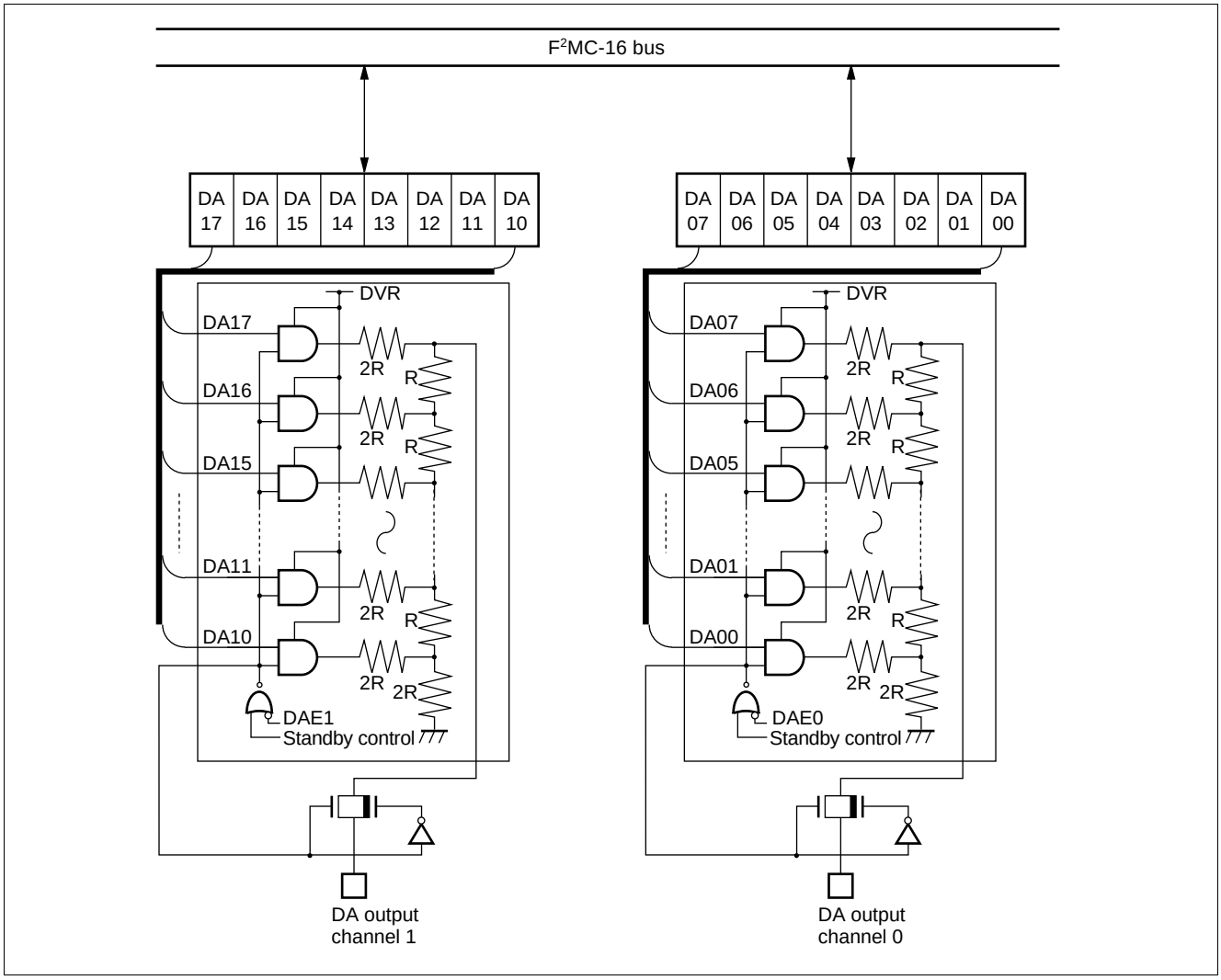
5. D/A Converter

This block is an R-2R type D/A converter with 8-bit resolution. The device contains two D/A converters. The D/A control register controls the output of the two D/A converters independently.

(1) Register Configuration

	bit	7	6	5	4	3	2	1	0	
Address: 00003A$_H$		DA07	DA06	DA05	DA04	DA03	DA02	DA01	DA00	D/A converter data register 0 (DAT0)
	bit	15	14	13	12	11	10	9	8	
Address: 00003B$_H$		DA17	DA16	DA15	DA14	DA13	DA12	DA11	DA10	D/A converter data register 0 (DAT1)
	bit	7	6	5	4	3	2	1	0	
Address: 00003C$_H$		—	—	—	—	—	—	—	DAE0	D/A control register 0 (DACR0)
	bit	15	14	13	12	11	10	9	8	
Address: 00003D$_H$		—	—	—	—	—	—	—	DAE1	D/A control register 1 (DACR1)

(2) Block Diagram



6. 8/16-bit PPG

This block is an 8-bit reload timer module. The block performs PPG output in which the pulse output is controlled by the operation of the timer.
The hardware consists of two 8-bit down-counters, four 8-bit reload registers, one 16-bit control register, two external pulse output pins, and two interrupt outputs. The PPG has the following functions.

- 8-bit PPG output in two channels independent operation mode:
  Two independent PPG output channels are available.
- 16-bit PPG output operation mode : One 16-bit PPG output channel is available.
- 8+8-bit PPG output operation mode : Variable-period 8-bit PPG output operation is available by using the output of channel 0 as the clock input to channel 1.
- PPG output operation : Outputs pulse waveforms with variable period and duty ratio. Can be used as a D/A converter in conjunction with an external circuit.

(1) Register Configuration

PPG0 operation mode control
Address: channel 0 000044H

	7	6	5	4	3	2	1	0	
	PEN0	—	PE00	PIE0	PUF0	—	—	Reserved	PPGC0
Read/write →	(R/W)	(—)	(R/W)	(R/W)	(R/W)	(—)	(—)	(—)	
Initial value →	(0)	(X)	(0)	(0)	(0)	(X)	(X)	(1)	

PPG1 operation mode control
Address: channel 1 000045H

	15	14	13	12	11	10	9	8	
	PEN1	—	PE10	PIE1	PUF1	MD1	MD0	Reserved	PPGC1
Read/write →	(R/W)	(—)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(—)	
Initial value →	(0)	(X)	(0)	(0)	(0)	(0)	(0)	(1)	

PPG0, 1 output control register
Address: channel 0,1 000046H

	7	6	5	4	3	2	1	0	
	PCS2	PCS1	PCS0	PCM2	PCM1	PCM0	PE11	PE01	PPGOE
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)	

Reload register H
Address: channel 0 000041H
channel 1 000043H

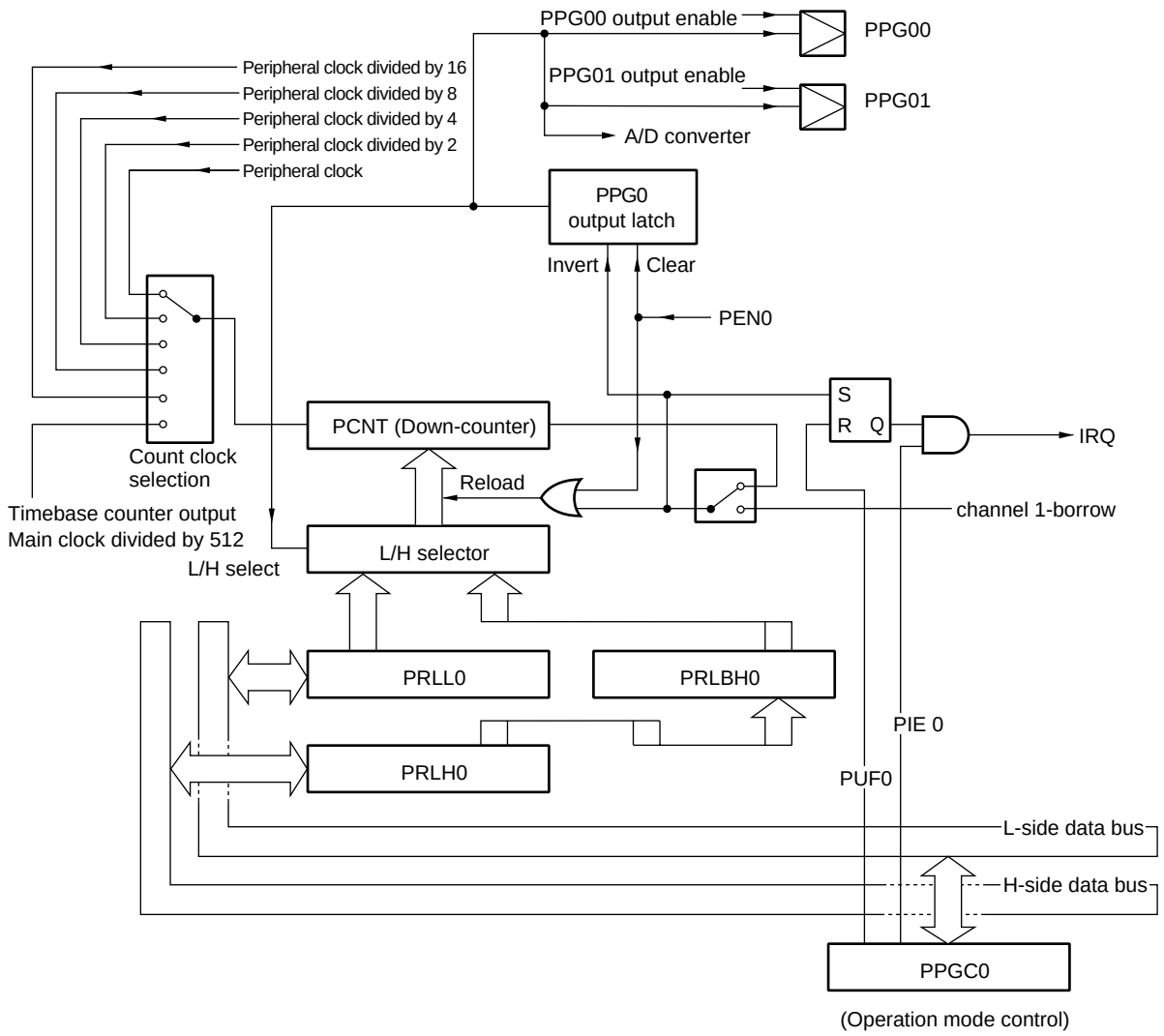
	15	14	13	12	11	10	9	8	
									PRLH0, 1
Read/write →	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
Initial value →	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

Reload register L
Address: channel 0 000040H
channel 1 000042H

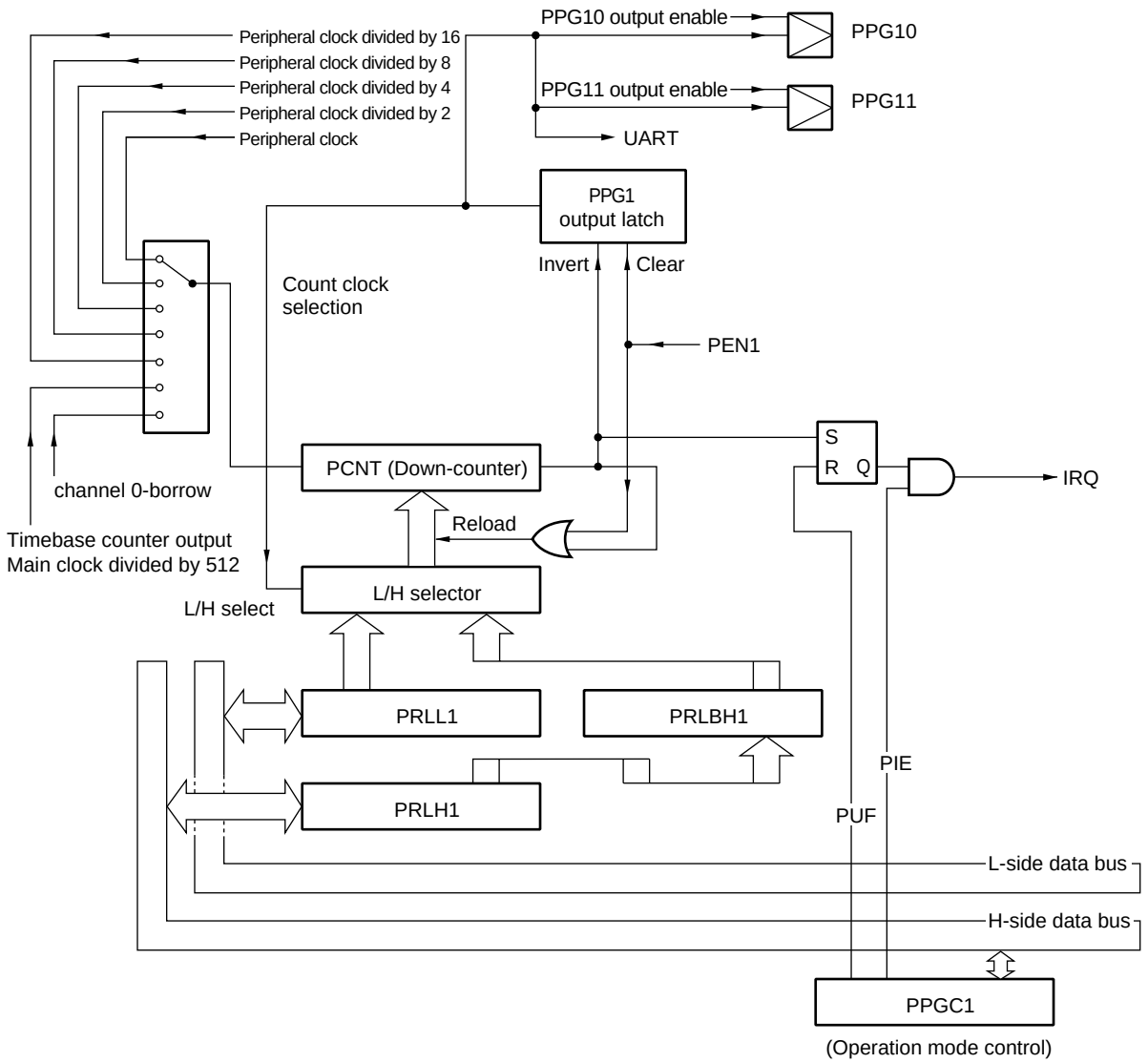
	7	6	5	4	3	2	1	0	
									PRLL0, 1
Read/write →	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
Initial value →	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

(2) Block Diagram

- **8/16-bit PPG (channel 0)**



- **8/16-bit PPG (channel 1)**



7. 8/16-bit Up/Down Counter/Timer

This block is an up/down counter/timer and consists of six event input pins, two 8-bit up/down counters, two 8-bit reload/compare registers, and their control circuits.

(1) Main Functions

- The 8-bit count register can count in the range 0 to 256D (or 0 to 65535D in 1 × 16-bit operation mode).
- The count clock selection can select between four different count modes.

  Count modes
  - Timer mode
  - Up/down counter mode
  - Phase difference count mode (× 2)
  - Phase difference count mode (× 8)

- Two different internal count clocks are available in timer mode.

  Count clock (at 16 MHz operation)
  - 125 ns (8 MHz: Divide by 2)
  - 1.0 µs (1 MHz: Divide by 8)

- In up/down count mode, you can select which edge to detect on the external pin input signal.

  Detected edge
  - Detect falling edges
  - Detect rising edges
  - Detect both rising and falling edges
  - Edge detection disabled

- Phase difference count mode is suitable for motor encoder counting. By inputting the A, B, and Z phase outputs from the encoder, a high-precision rotational angle, speed, or similar count can be implemented simply.
- Two different functions can be selected for the ZIN pin.

  ZIN pin
  - Counter clear function
  - Gate function

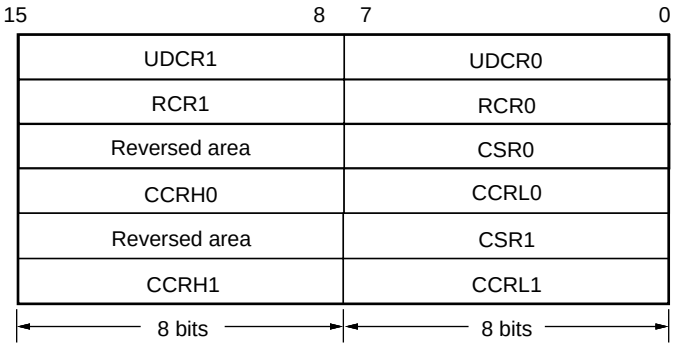
- Compare and reload functions are available and can be used either independently or together. A variable-width up/down count can be performed by activating both functions.

  Compare/reload function
  - Compare function (Output an interrupt when a compare occurs.)
  - Compare function (Output an interrupt and clear the counter when a compare occurs.)
  - Reload function (Output an interrupt and reload when an underflow occurs.)
  - Compare/reload function (Output an interrupt and clear the counter when a compare occurs. Output an interrupt and reload when an underflow occurs.)
  - Compare/reload disabled

- Whether or not to generate an interrupt when a compare, reload (underflow), or overflow occurs can be set independently.
- The previous count direction can be determined from the count direction flag.
- An interrupt can be generated when the count direction changes.

(2) Register Configuration

The 8/16-bit up/down counter/timer has the following registers.



	bit	7	6	5	4	3	2	1	0	
Address: 000070H		D07	D06	D05	D04	D03	D02	D01	D00	Up/down count register channel 0 (UDCR0)

	bit	15	14	13	12	11	10	9	8	
Address: 000071H		D17	D16	D15	D14	D13	D12	D11	D10	Up/down count register channel 1 (UDCR1)

	bit	7	6	5	4	3	2	1	0	
Address: 000072H		D07	D06	D05	D04	D03	D02	D01	D00	Reload compare register channel 0 (RCR1)

	bit	15	14	13	12	11	10	9	8	
Address: 000073H		D17	D16	D15	D14	D13	D12	D11	D10	Reload compare register channel 1 (RCR1)

	bit	7	6	5	4	3	2	1	0	
Address: 000074H 000078H		CSTR	CITE	UDIE	CMPF	OVFF	UDFF	UDF1	UDF0	Counter status register channel 0, 1 (CSR0, 1)

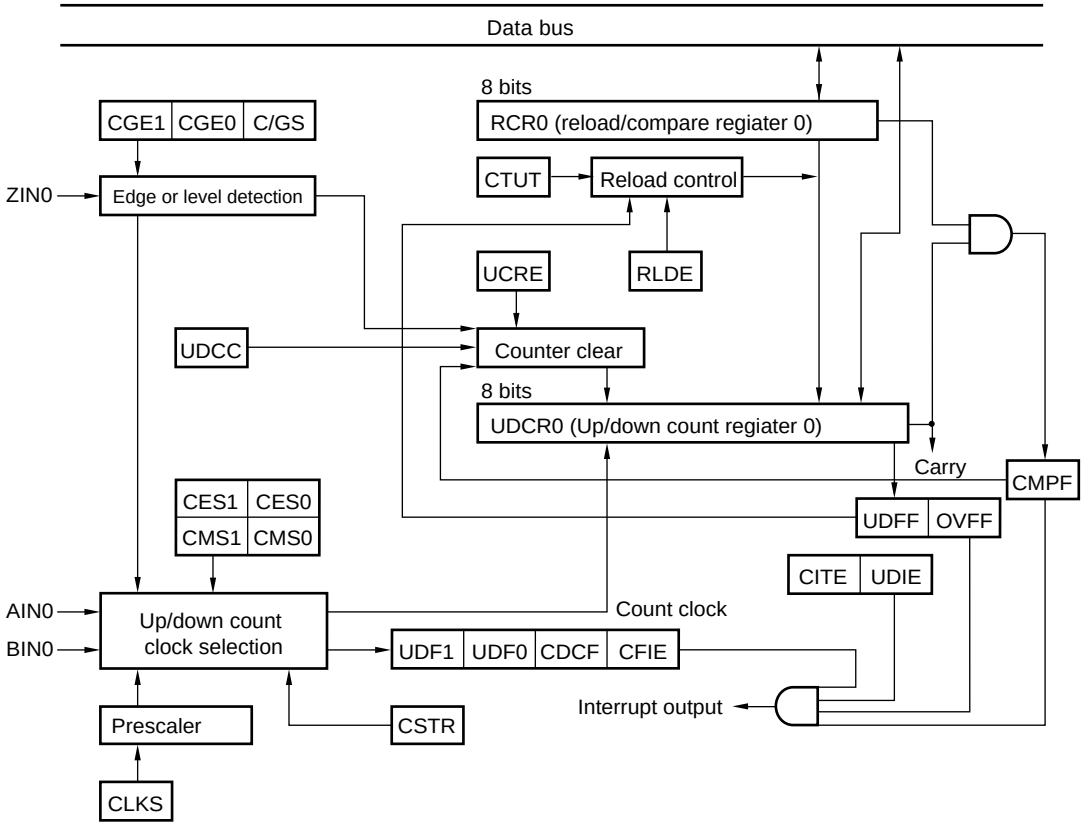
	bit	7	6	5	4	3	2	1	0	
Address: 000076H 00007AH		—	CTUT	UCRE	RLDE	UDCC	CGSC	CGE1	CGE0	Counter status register channel 0, 1 (CCRL0, 1)

	bit	15	14	13	12	11	10	9	8	
Address: 000077H		M16E	CDCF	CFIE	CLKS	CMS1	CMS0	CES1	CES0	Counter control register channel 0 (CCRH0)

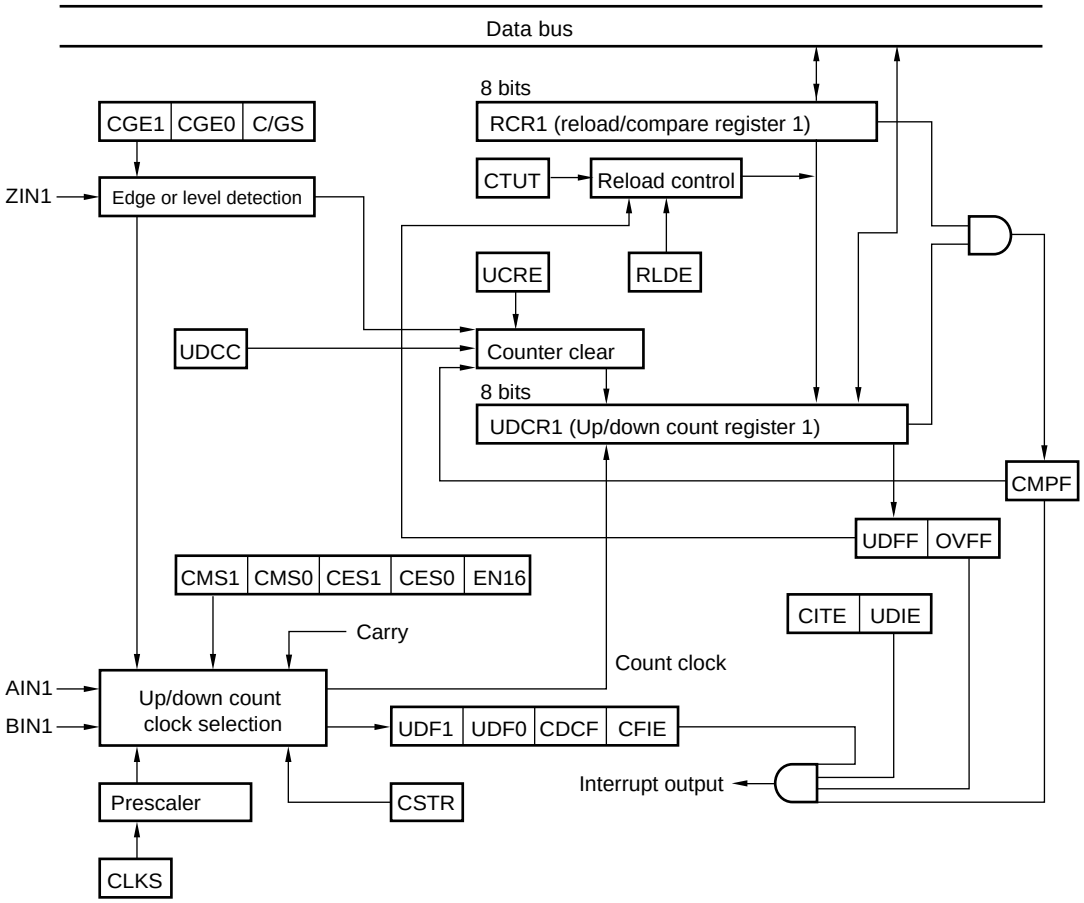
	bit	7	6	5	4	3	2	1	0	
Address: 00007BH		—	CDCF	CFIE	CLKS	CMS1	CMS0	CES1	CES0	Counter control register channel 1 (CCRH1)

(3) Block Diagram

- **8/16-bit Up/Down Counter/Timer (channel 0)**



- **8/16-bit Up/Down Counter/Timer (channel 1)**



8. Clock Output Control Register

The clock output outputs the divided machine clock.

(1) Register Configuration

Clock control register
Address: 0003E$_H$

bit	7	6	5	4	3	2	1	0	
	—	—	—	—	CKEN	FRQ2	FRQ1	FRQ0	CLKR
Read/write →					(R/W)	(R/W)	(R/W)	(R/W)	
Initial value →	(—)	(—)	(—)	(—)	(0)	(0)	(0)	(0)	

(a) [bit 3] CKEN
CKOT output enable bit

MODE	Operation
0	Operate as a standard port.
1	Operate as the CKOT output.

(b) [bits 2, 1, 0] FRQ2, FRQ1, FRQ0
These bits select the output frequency of the clock.

FRQ2	FRQ1	FRQ0	Output clock	ϕ = 16 MHz	ϕ = 8 MHz	ϕ = 4 MHz
0	0	0	$\phi/2^1$	125 ns	250 ns	500 ns
0	0	1	$\phi/2^2$	250 ns	500 ns	1 µs
0	1	0	$\phi/2^3$	500 ns	1 µs	2 µs
0	1	1	$\phi/2^4$	1 µs	2 µs	4 µs
1	0	0	$\phi/2^5$	2 µs	4 µs	8 µs
1	0	1	$\phi/2^6$	4 µs	8 µs	16 µs
1	1	0	$\phi/2^7$	8 µs	16 µs	32 µs
1	1	1	$\phi/2^8$	16 µs	32 µs	64 µs

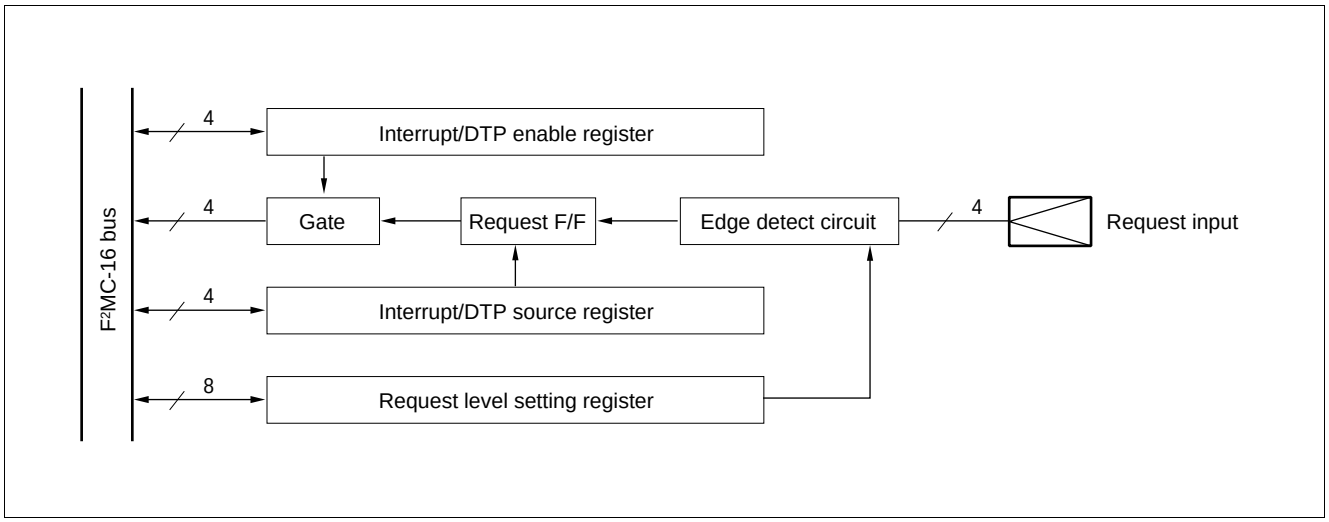
9. DTP/External Interrupts

The DTP (Data Transfer Peripheral) is a peripheral block that interfaces external peripherals to the F²MC-16L CPU. The DTP receives DMA and interrupt processing requests from external peripherals and passes the requests to the F²MC-16L CPU to activate the intelligent I/O service or interrupt processing. Two request levels (“H” and “L”) are provided for the intelligent I/O service. For external interrupt requests, generation of interrupts on a rising or falling edge as well as on “H” and “L” levels can be selected, giving a total of four types.

(1) Register Configuration

	bit	7	6	5	4	3	2	1	0	
Address: 000030H		EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0	Interrupt/DTP enable register (ENIR)
	bit	15	14	13	12	11	10	9	8	
Address: 000031H		ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0	Interrupt/DTP source register (EIRR)
	bit	7	6	5	4	3	2	1	0	
Address: 000032H		LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0	Request level setting register (ELVR)
	bit	15	14	13	12	11	10	9	8	
Address: 000033H		LB7	LA7	LB6	LA6	LB5	LA5	LB4	LA4	Request level setting register (ELVR)

(2) Block Diagram



10. 16-bit I/O Timer

The 16-bit I/O timer consists of one 16-bit free-run timer, four output compare, and two input capture modules. Based on the 16-bit free-run timer, these functions can be used to generate two independent waveform outputs and to measure input pulse widths and external clock periods.

(1) A Summary of Each Function

- 16-bit free-run timer (× 1)

The 16-bit free-run timer consists of a 16-bit up-counter, a control register, and a prescaler. The output of the timer/counter is used as the base time for the input capture and output compare.

(a) The operating clock for the counter can be selected from four different clocks.
Four internal clocks (ϕ/4, ϕ/16, ϕ/32, ϕ/64)

(b) Interrupts can be generated when a counter value overflow or compare match with compare register 0 occurs (the appropriate mode must be set for a compare match).

(c) The counter can be initialized to 0000_H by a reset, software clear, or compare match with compare register 0.

- Output compare (× 4)

The output compare consists of two 16-bit compare registers, compare output latches, and control registers. The modules can invert the output level and generate an interrupt when the 16-bit free-run timer value matches the compare register value.

(a) The four compare registers can be operated independently.
Each compare register has a corresponding output pin and interrupt flag.

(b) The four compare registers can be paired to control the output pins.
Invert the output pins using the four compare registers.

(c) Initial values can be set for the output pins.

(d) An interrupt can be generated when a compare match occurs.

- Input capture (× 2)

The input capture consists of two independent external input pins, their corresponding capture registers, and a control register. The value of the 16-bit free-run timer can be stored in the capture register and an interrupt generated when the specified edge is detected on the signal from the external input pin.

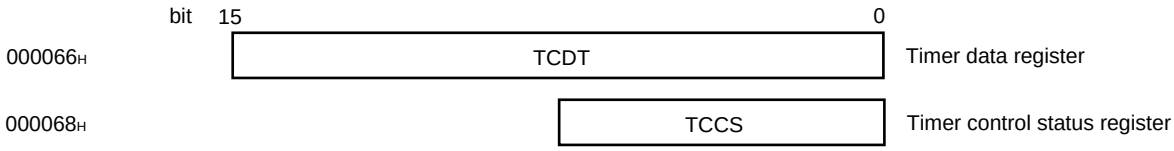
(a) The edge to detect on the external input signal is selectable.
Detection of rising edges, falling edges, or either edge can be specified.

(b) The two input capture channels can operate independently.

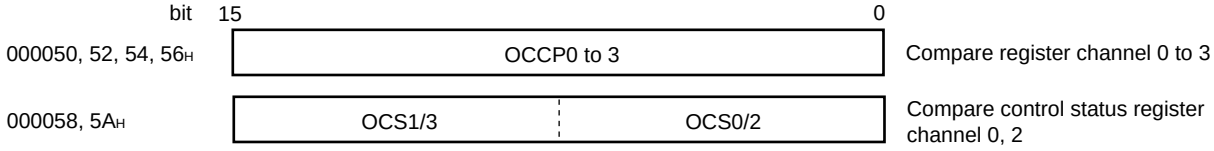
(c) An interrupt can be generated on detection of the specified edge on the external input signal.
The input capture interrupt can activate the intelligent I/O service.

(2) Register Configuration for the Entire 16-bit I/O Timer

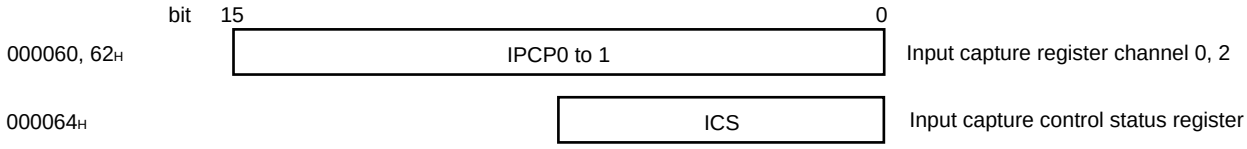
- **16-bit free-run timer**



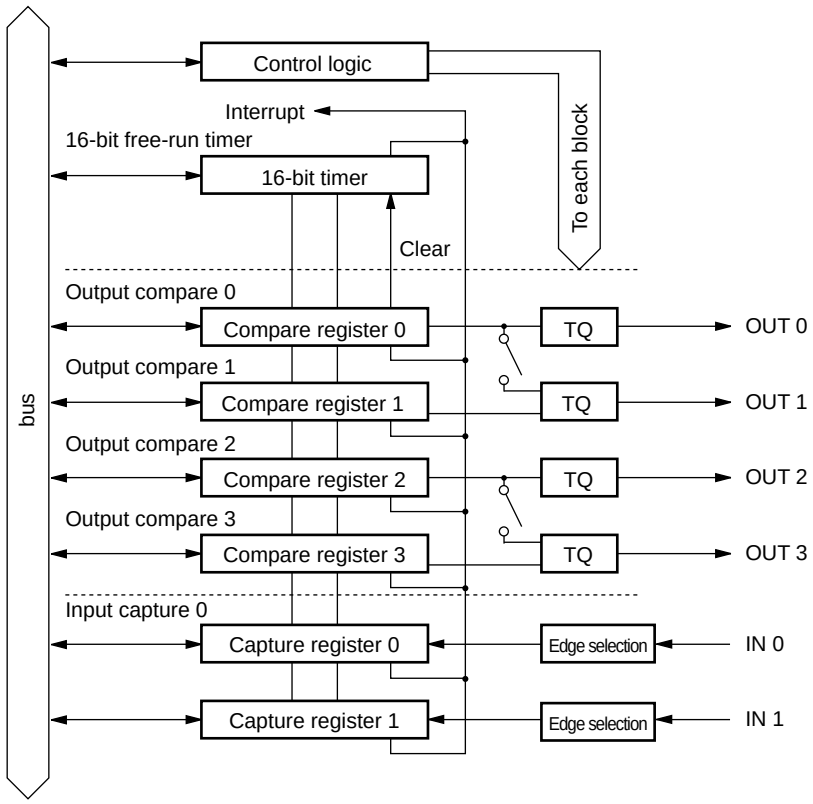
- **16-bit output compare**



- **16-bit input capture**



- **Overall Block Diagram of the 16-bit I/O timer**



(3) 16-bit Free-run Timer

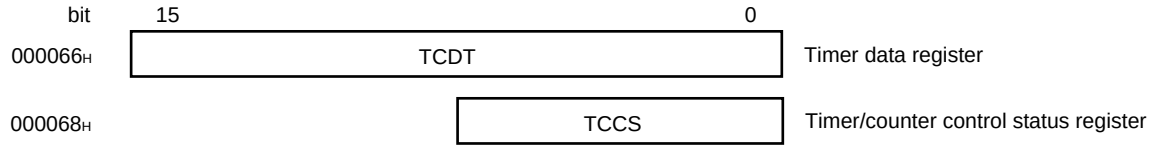
The 16-bit free-run timer consists of a 16-bit up-counter and a control status register. The count value of the timer is used as the base time for the input capture and output compare.

(a) The count clock can be selected from four different clocks.

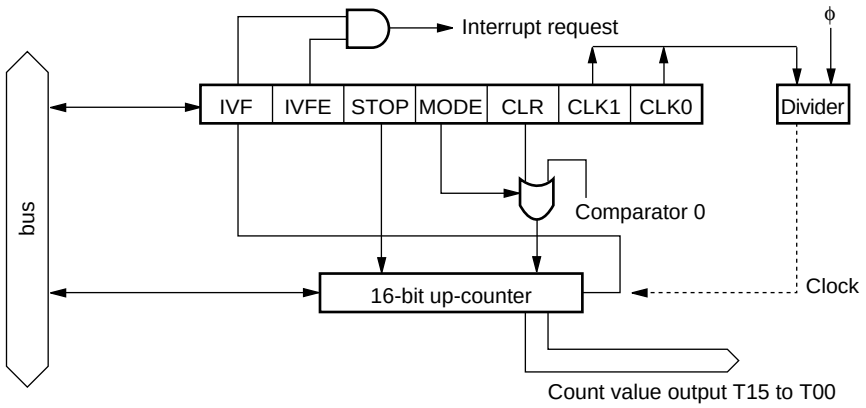
(b) Interrupts can be generated when a counter value overflow occurs.

(c) Depending on the mode setting, the counter can be initialized when a match occurs with compare register 0 of the output compare.

- **Register Configuration**



- **Block Diagram**



- **Register Details**

Data Register

bit	15	14	13	12	11	10	9	8
Address: 000067H	T15	T14	T13	T12	T11	T10	T09	T08
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)

bit	7	6	5	4	3	2	1	0
Address: 000066H	T07	T06	T05	T04	T03	T02	T01	T00
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)

The count value of the 16-bit free-run timer can be read from this register. The count is cleared to "0000H" by a reset. Writing to this register sets the timer value. However, only write to the register when the timer is halted (STOP = "1"). Always use word access.
The 16-bit free-run timer is initialized by the following.

(a) Reset

(b) The clear bit (CLR) of the control status register

(c) A match between the timer/counter value and compare register 0 of the output compare (if the appropriate mode is set)

(4) Output Compare

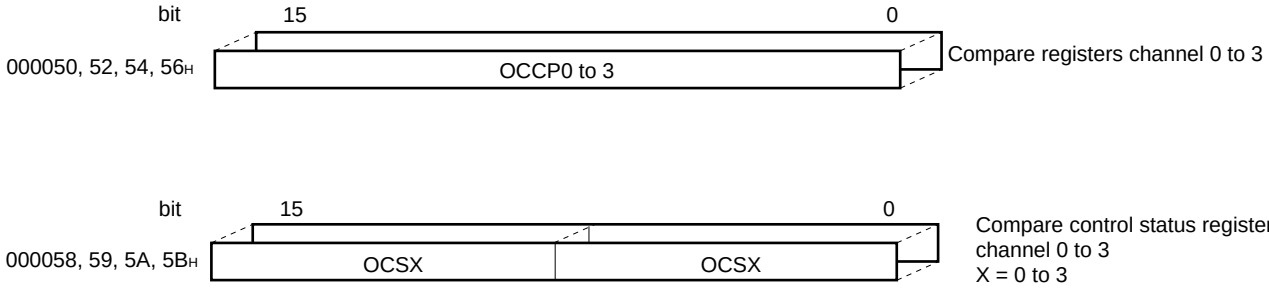
The output compare consists of 16-bit compare registers, compare output pins, and a control register. The module can invert the output level and generate an interrupt when the 16-bit free-run timer value matches a compare register value.

(a) The two compare registers can be operated independently.
The output compare can also be set to control pin output using two compare registers.

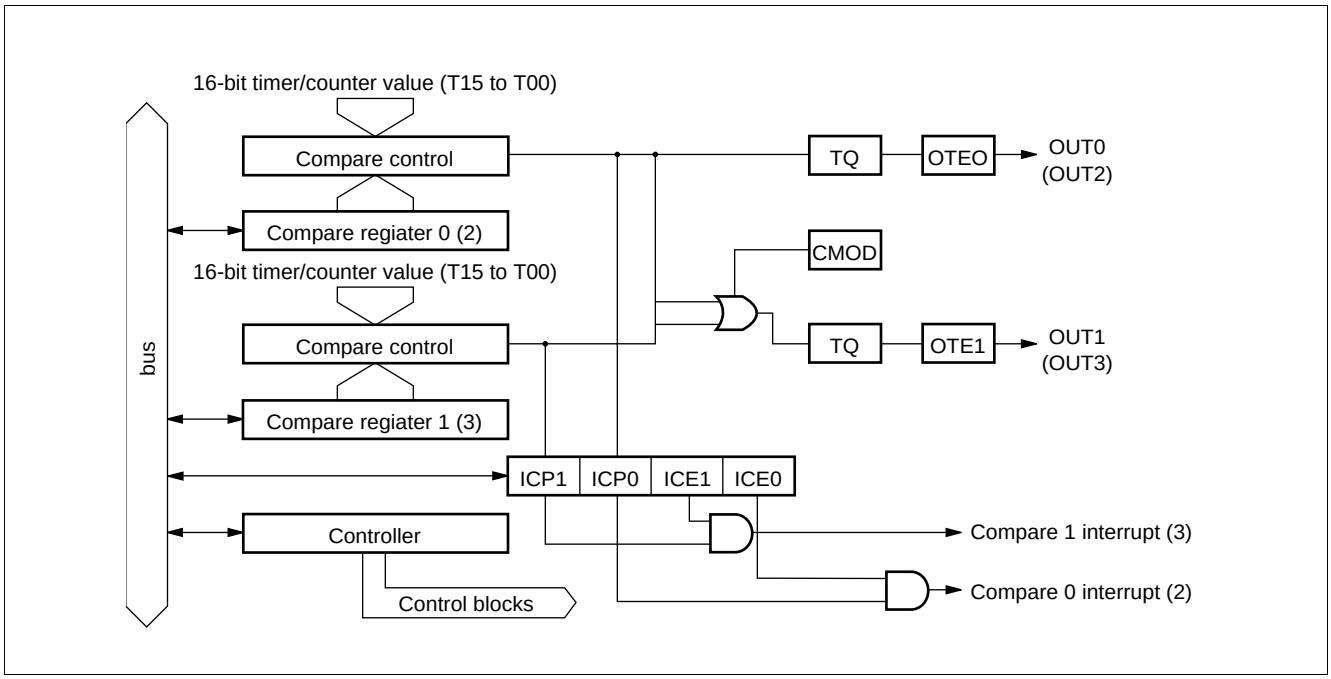
(b) The initial value of the output pins can be set.

(c) An interrupt can be generated when a compare match occurs.

- **Register Configuration**



- **Block Diagram**



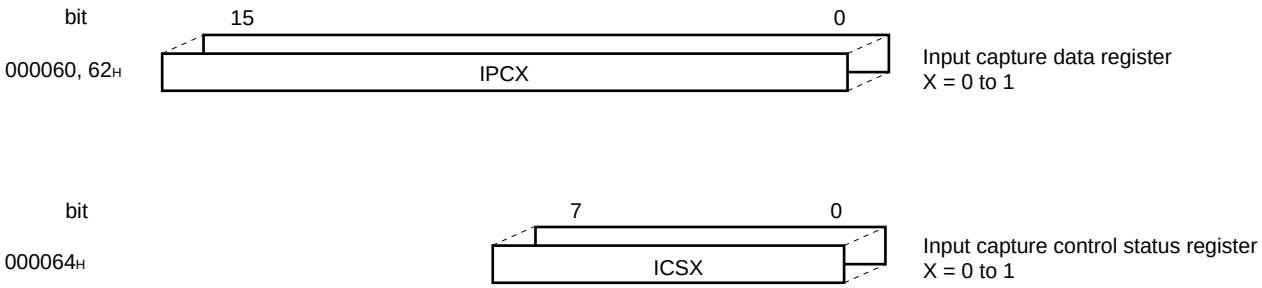
(5) Input Capture

The function of this module is to store the value of the 16-bit free-run timer in a register when the specified edge (rising, falling, or either edge) is detected on the external input signal. The module can also generate an interrupt on detection of the edge. The input capture contains input capture data registers and a control register. Each input capture has a corresponding external input pin.

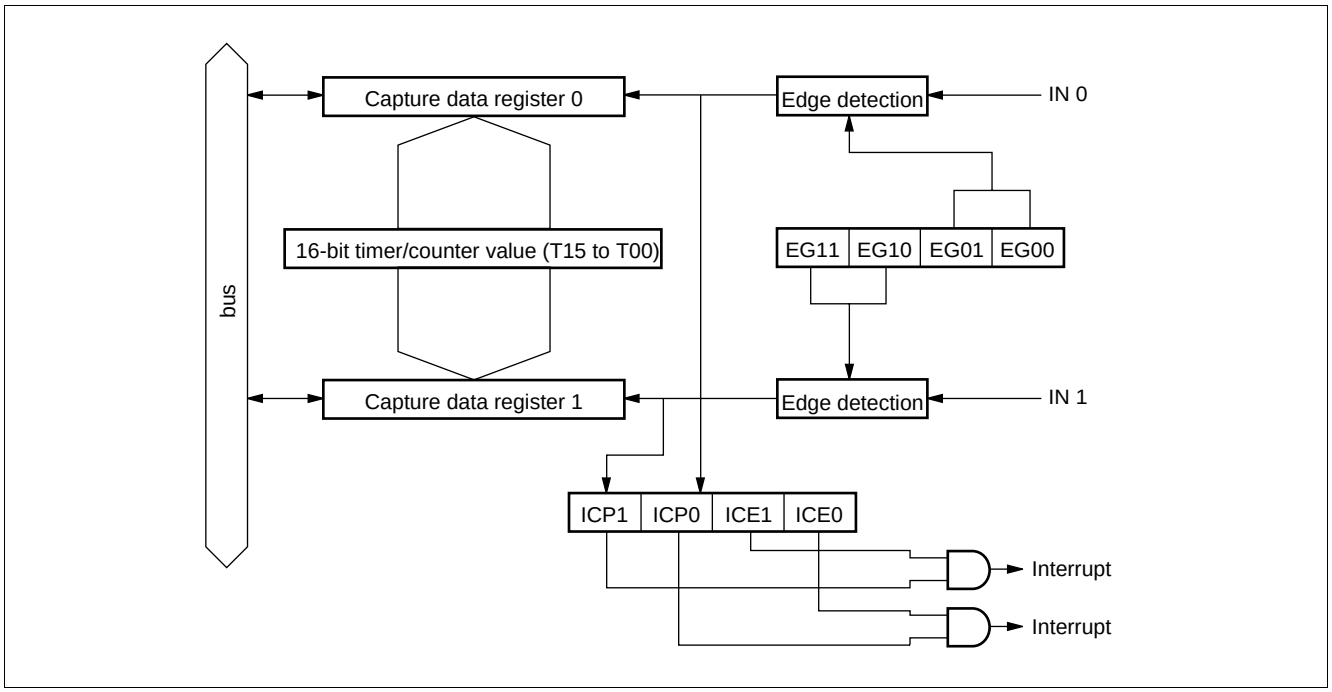
(a) Three different types of edge detection can be selected.
Rising edges (↑), falling edges (↓), or either edge (↑ ↓).

(b) An interrupt can be generated on detection of the specified edge on the external input.

- **Register Configuration (for the entire input capture)**



- **Block Diagram**



- **Register Details**

Input capture data register

bit	15	14	13	12	11	10	9	8
000060, 62H	CP15	CP14	CP13	CP12	CP11	CP10	CP09	CP08
Read/write →	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)
Initial value →	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)

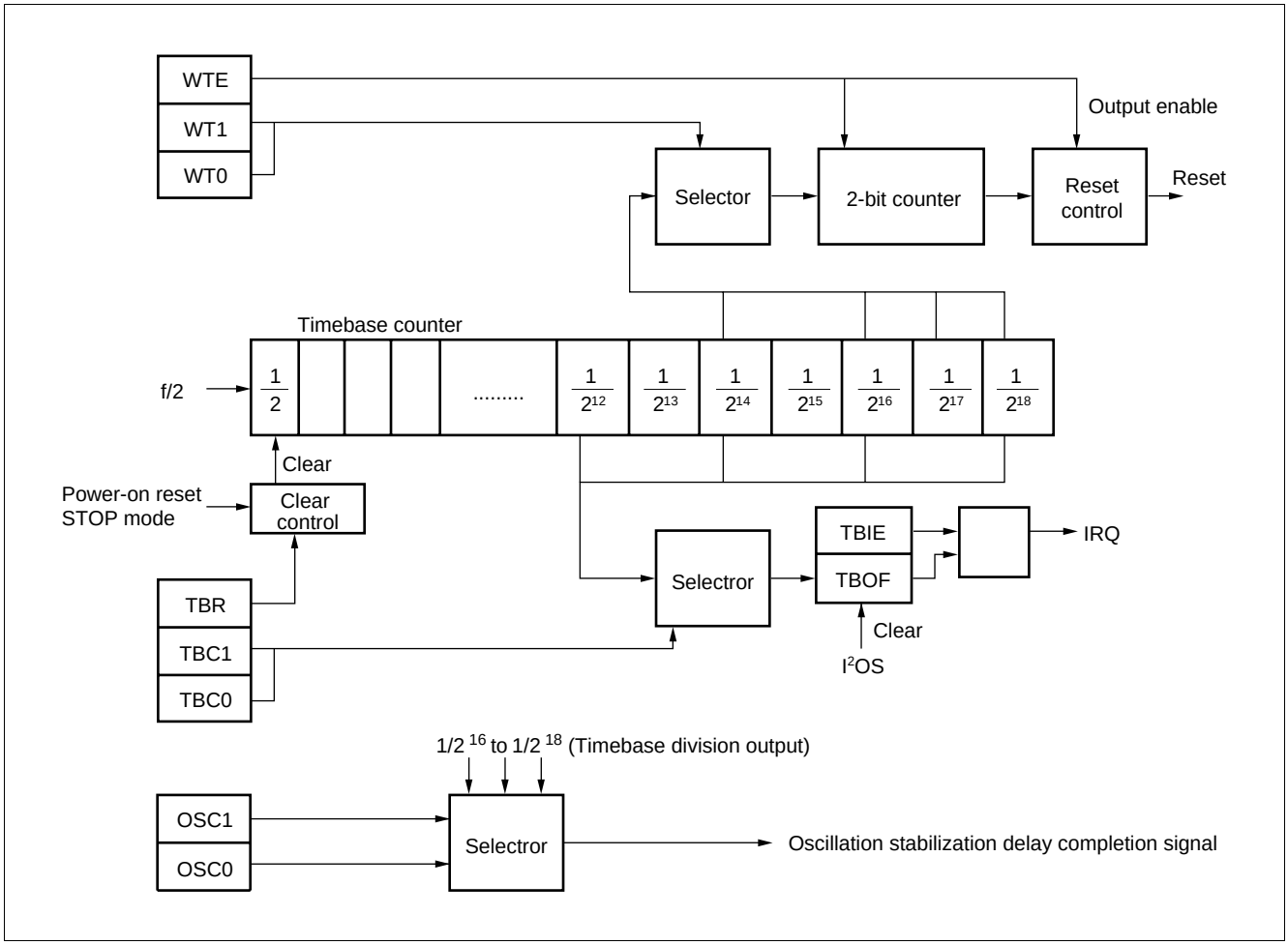
bit	7	6	5	4	3	2	1	0
	CP07	CP06	CP05	CP04	CP03	CP02	CP01	CP00
Read/write →	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)
Initial value →	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)

The 16-bit free-run timer value is stored in these registers when the specified edge is detected on the input waveform from the corresponding external pin. (Always use word access. Writing is prohibited.)

11. Watchdog Timer

The watchdog timer consists of a 2-bit watchdog counter that uses the carry signal from the 18-bit timebase counter as its clock source, a control register, and a watchdog reset controller. The following block diagram shows the structure of both the watchdog timer and timebase timer (see “12. Timebase Timer”).

(1) Block Diagram



(2) Register Configuration

bit	7	6	5	4	3	2	1	0	
Address: 0000A8H	PONR	STBR	WRST	ERST	SRST	WTE	WT1	WT0	Watchdog timer control register (WDTC)

12. Timebase Timer

The timebase timer consists of an 18-bit timebase counter (which divides the system clock) and a control register. The carry signal of the timebase counter can generate a fixed period interrupt.
All bits of the timebase counter are cleared to zero at power-on, when stop mode is set, or by software (by writing “0” to the TBR bit). The timebase counter continuously increments while an oscillation is input.
The timebase counter is also used as the clock source for the watchdog timer and as a timer for the oscillation stabilization delay time.

(1) Block Diagram

See “(1) Block diagram” in “11. Watchdog Timer” for the block diagram of the timebase timer.

(2) Register Configuration

	bit	15	14	13	12	11	10	9	8	
Address: 0000A9H		Reserved	—	—	TBIE	TBCF	TBR	TBC1	TBC0	Timebase timer control register (TBTC)

(3) Register Details

- **TBTC (Timebase timer control register)**

	bit	15	14	13	12	11	10	9	8	
Address: 0000A9H		Reserved	—	—	TBIE	TBCF	TBR	TBC1	TBC0	Initial value X--00000B
		(W)			(R/W)	(R/W)	(W)	(R/W)	(R/W)	

(a) [bit 15] Reserved
A reserved bit. Always set to “1” when writing data to the register.

(b) [bit 12] TBIE
Interval interrupt enable bit for the timebase timer. The interrupt is enabled when TBIE is “1” and disabled when TBIE is “0”. Initialized to “0” by a reset. The bit is readable and writable.

(c) [bit 11] TBOF
Interrupt request flag for the timebase timer. An interrupt request is generated if TBCF goes to “1” when TBIE is “1”. The bit is set to “1” at fixed intervals set by the TBC1 and 0 bits. Clear by writing “0”, transition to stop or hardware standby mode, or a reset. Writing “1” has no meaning.
Read as “1” by read-modify-write instructions.

(d) [bit 10] TBR
Clears all bits of the timebase counter to “0”. Writing “0” to the TBR bit clears the timebase counter. Writing “1” to the TBR bit is meaningless. Reading from the TBR bit results in “1”.

(e) [bit 9, 8] TBC1, 0
Set a timebase timer interval. The bits are initialized to “00” by resetting. These bits are readable and writable.

Setting of timebase timer interval

TBC1	TBC0	Interval time when base frequency is 4 MHz
0	0	1.024 ms
0	1	4.096 ms
1	0	16.384 ms
1	1	131.072 ms

13. External Bus Pin Control Circuit

The external bus pin control circuit controls the external bus pins required to extend the CPU's address/data bus outside the device.

(1) Register Configuration

Auto-ready function selection register (ARSR)
Address: 0000A5H

bit	15	14	13	12	11	10	9	8
	ICR1	ICR0	HMR1	HMR0	—	—	LMR1	LMR0
Read/write →	(W)	(W)	(W)	(W)	(—)	(—)	(W)	(W)
Initial value →	(0)	(0)	(1)	(1)	(—)	(—)	(0)	(0)

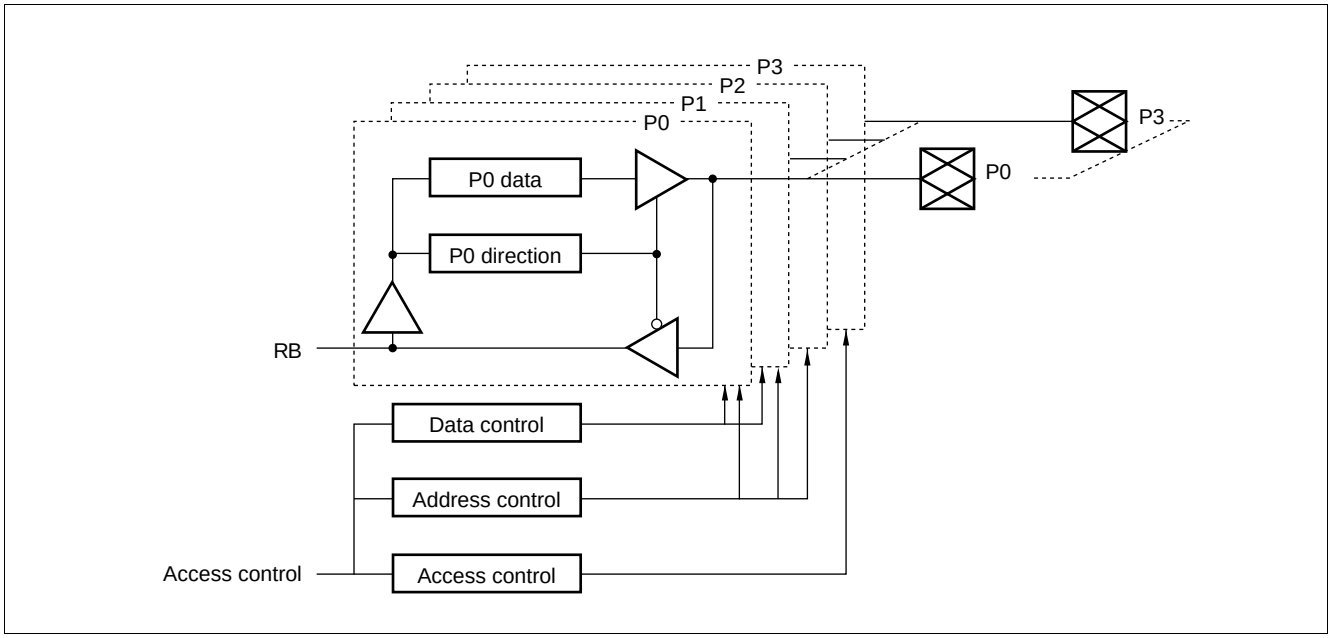
External address output control register (HACR)
Address: 0000A6H

bit	7	6	5	4	3	2	1	0
	E23	E22	E21	E20	E19	E18	E17	E16
Read/write →	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)
Initial value →	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)

Bus control signal selection register (EPCR)
Address: 0000A7H

bit	15	14	13	12	11	10	9	8
	CKE	RYE	HDE	ICBS	HMBS	WRE	LMBS	—
Read/write →	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(—)
Initial value →	(0)	(0)	(0)	(0)	(1/0)	(0)	(0)	(—)

(2) Block Diagram



14. Low-Power Control Circuits (CPU Intermittent Operation Function, Oscillation Stabilization Delay Time, and Clock Multiplier Function)

The following operation modes are available: PLL clock mode, PLL sleep mode, timer mode, main clock mode, main sleep mode, stop mode, and hardware standby mode. Operation modes other than PLL clock mode are classified as low power consumption modes.
In main clock mode and main sleep mode, the device operates on the main clock only (OSC oscillator clock). The PLL clock (VCO oscillator clock) is stopped in these modes and the main clock divided by 2 is used as the operating clock.
In PLL sleep mode and main sleep mode, the CPU's operating clock only is stopped and other elements continue to operate.
In timer mode, only the timebase timer operates.
Stop mode and hardware standby mode stop the oscillator. These modes maintain existing data with minimum power consumption.
The CPU intermittent operation function provides an intermittent clock to the CPU when register, internal memory, internal resource, or external bus access is performed. This function reduces power consumption by lowering the CPU execution speed while still providing a high-speed clock to internal resources.
The PLL clock multiplier ratio can be set to 1, 2, 3, or 4 by the CS1, 0 bits.
The WS1, 0 bits set the delay time to wait for the main clock oscillation to stabilize when recovering from stop mode or hardware standby mode.

(1) Register Configuration

Low-power consumption mode register

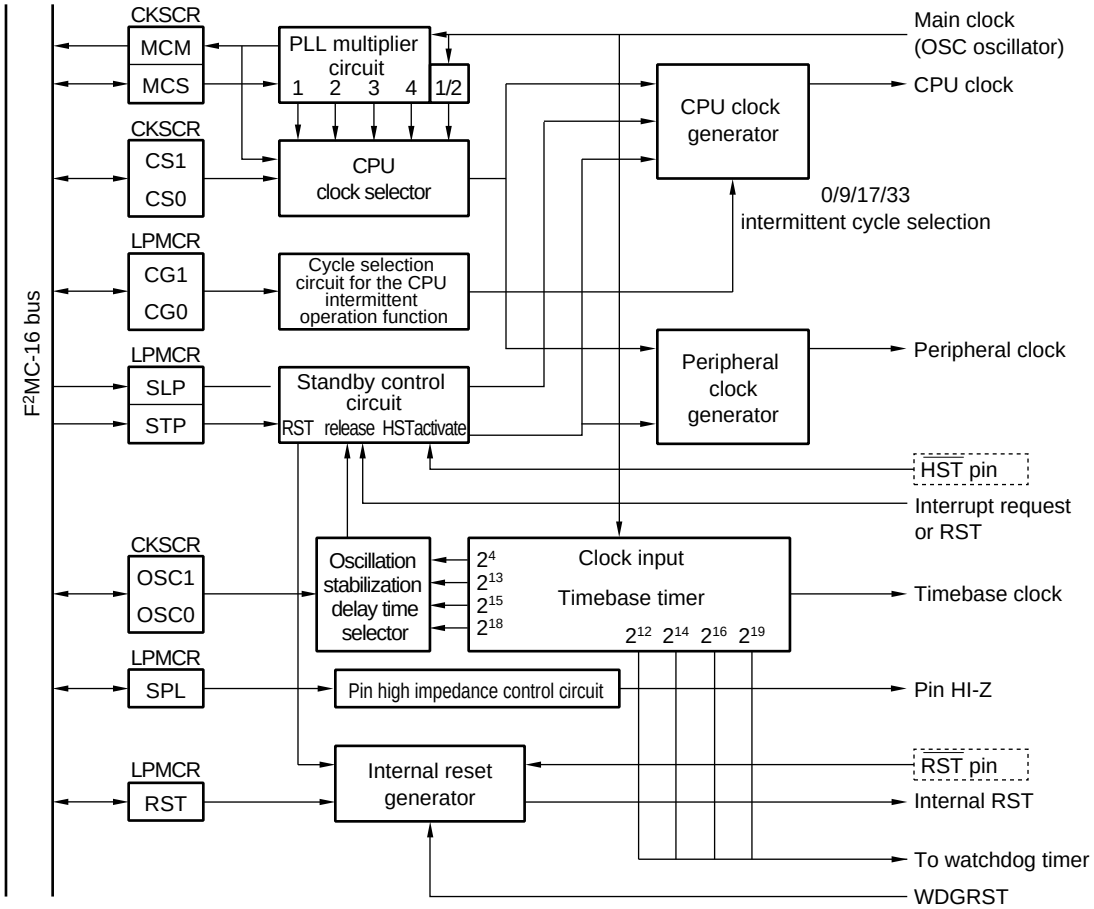
Address: 0000A0H	bit 7	6	5	4	3	2	1	0	
	STP	SLP	SPL	RST	Reserved	CG1	CG0	Reserved	LPMCR
Read/write →	(W)	(W)	(R/W)	(W)	(—)	(R/W)	(R/W)	(—)	
Initial value →	(0)	(0)	(0)	(1)	(1)	(0)	(0)	(0)	

Clock select register

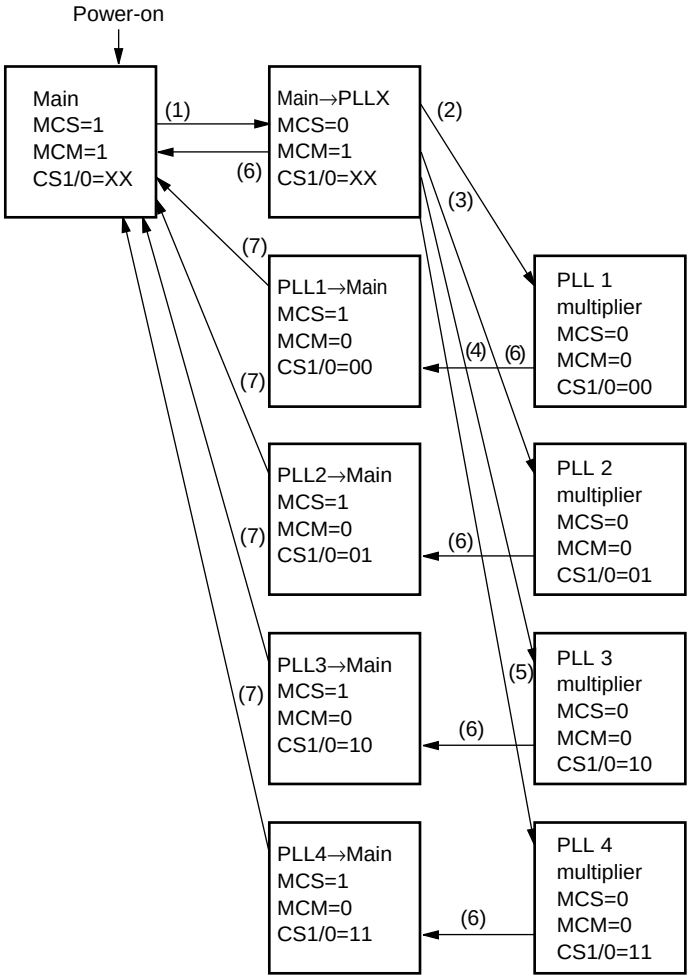
Address: 0000A1H	bit 15	14	13	12	11	10	9	8	
	Reserved	MCM	WS1	WS0	Reserved	MCS	CS1	CS0	CKSCR
Read/write →	(—)	(R)	(R/W)	(R/W)	(—)	(R/W)	(R/W)	(R/W)	
Initial value →	(1)	(1)	(1)	(1)	(1)	(1)	(0)	(0)	

(2) Block Diagram

- **Low-Power Consumption Control Circuit and Clock Generator**



- **State Transition Diagram for Clock Selection**



(1) MCS bit cleared
(2) PLL clock oscillation stabilization delay complete and CS1/0="00"
(3) PLL clock oscillation stabilization delay complete and CS1/0="01"
(4) PLL clock oscillation stabilization delay complete and CS1/0="10"
(5) PLL clock oscillation stabilization delay complete and CS1/0="11"
(6) MCS bit set (including a hardware standby or watchdog reset)
(7) PLL clock and main clock synchronized timing

15. Delayed Interrupt Generation Module

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

(1) Register Configuration

	bit	15	14	13	12	11	10	9	8	
Address:	00009F$_H$	—	—	—	—	—	—	—	R0	Delayed interrupt request register (DIRR)

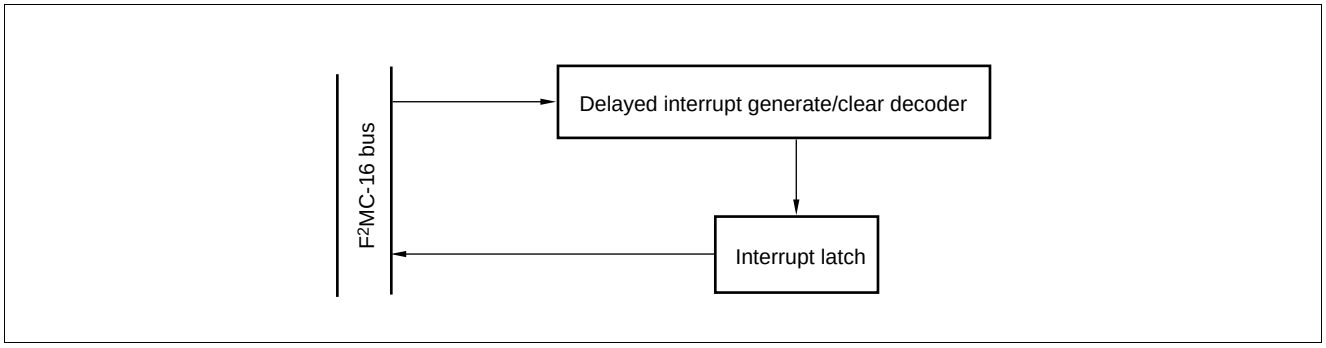
(2) Register Details

Delayed interrupt request register (DIRR)

	bit	15	14	13	12	11	10	9	8	
Address:	00009F$_H$	—	—	—	—	—	—	—	R0	Initial value -------0$_B$
									(R/W)	

The DIRR register controls generation and clearing of delayed interrupt requests. Writing “1” to the register generates a delayed interrupt request. Writing “0” to the register clears the delayed interrupt request. The register is set to the interrupt cleared state by a reset. Either “0” or “1” can be written to the reserved bits. However, considering possible future extensions, it is recommended that the set bit and clear bit instructions are used for register access.

(3) Block Diagram



■ ELECTRICAL CHARACTERISTICS

1. Absolute Maximum Ratings

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

Parameter	Symbol	Value		Unit	Remarks
		Min.	Max.		
Power supply voltage	V_{CC}	V_{SS} – 0.3	V_{SS} + 7.0	V	
	AV_{CC}*1	V_{SS} – 0.3	V_{SS} + 7.0	V	
	AVRH, AVRL*1	V_{SS} – 0.3	V_{SS} + 7.0	V	
Program voltage	V_{PP}	V_{SS} – 0.3	—	V	
Input voltage*2	V_I	V_{SS} – 0.3	V_{CC} + 0.3	V	
Output voltage*2	V_O	V_{SS} – 0.3	V_{CC} + 0.3	V	
“L” level (maximum) output current*3	I_{OL}	—	15	mA	
“L” level (average) output current*4	I_{OLAV}	—	50	mA	
“L” level total (maximum) output current	ΣI_{OL}	—	100	mA	
“L” level total (average) output current*5	ΣI_{OLAV}	—	50	mA	
“H” level (maximum) output current*3	I_{OH}	—	–15	mA	
“H” level (average) output current*4	I_{OHAV}	—	–50	mA	
“H” level total (maximum) output current	ΣI_{OH}	—	–100	mA	
“H” level total (average) output current*5	ΣI_{OHAV}	—	–50	mA	
Power consumption	P_d	—	+400	mW	
Operating temperature	T_A	–40	+85	°C	
Storage temperature	T_{stg}	–55	+150	°C	

*1: AV_{CC}, AVRH, and AVRL must not exceed V_{CC}. Similarly, it must not exceed AVRH and AVRL.
*2: V_I and V_O must not exceed V_{CC} + 0.3 V.
*3: The maximum output current must not be exceeded at any individual pin.
*4: The average output current is the rating for the current from an individual pin averaged over 100 ms.
*5: The average total output current is the rating for the current from all pins averaged over 100 ms.

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} = 0.0 V)

Parameter	Symbol	Value		Unit	Remarks
		Min.	Max.		
Power supply voltage	V_{CC}	2.7	5.5	V	For normal operation
		2.7	5.5	V	To maintain statuses in stop mode
"H" level input voltage	V_{IH}	0.7 V_{CC}	V_{CC} + 0.3	V	Other than V_{IHS}
	V_{IHS}	0.8 V_{CC}	V_{CC} + 0.3	V	Hysteresis inputs
	V_{IHM}	V_{CC} – 0.3	V_{CC} + 0.3	V	
"L" level input voltage	V_{IL}	V_{SS} – 0.3	0.3 V_{CC}	V	Other than V_{ILS}
	V_{ILS}	V_{SS} – 0.3	0.2 V_{CC}	V	Hysteresis inputs
	V_{ILM}	V_{SS} – 0.3	V_{SS} + 0.3	V	
Operating temperature	T_A	–40	+85	°C	

WARNING: Recommended operating conditions are normal operating ranges for the semiconductor device. All the device's electrical characteristics are warranted when operated within these ranges.

Always use semiconductor devices within the recommended operating conditions. 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 representative beforehand.

3. DC Characteristics

(V_{CC} = +2.7 V to +5.5 V, V_{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}	—	V_{CC} = +5.0 V±10%	0.7 V_{CC}	—	V_{CC} + 0.3	V	
	V_{IHS}			0.8 V_{CC}	—	V_{CC} + 0.3	V	*1
	V_{IHM}		—	V_{CC} – 0.3	—	V_{CC} + 0.3	V	
"L" level input voltage	V_{IL}	—	V_{CC} = +5.0 V±10%	0.7 V_{CC}	—	V_{CC} + 0.3	V	
	V_{ILS}			0.8 V_{CC}	—	V_{CC} + 0.3	V	*1
	V_{ILM}		—	V_{SS} – 0.3	—	V_{SS} + 0.3	V	
"H" level output voltage	V_{OH}	—	V_{CC} = +4.5 V±10% I_{OH} = –4.0 mA	V_{CC} – 0.5	—	—	V	
			V_{CC} = +2.7 V I_{OH} = –1.6 mA	V_{CC} – 0.3	—	—	V	
"L" level output voltage	V_{OL}	—	V_{CC} = +4.5 V±10% I_{OH} = –4.0 mA	—	—	0.4	V	
			V_{CC} = +2.7 V I_{OH} = –2.0 mA	—	—	0.4	V	
Pull-up resistor	R_{pull}	$\overline{RST}$	—	22	—	110	kΩ	
Power supply current*2	I_{CC}	V_{CC}	V_{CC} = +5.0 V±10% F_C = 16 MHz	—	60	80	mA	
	I_{CCS}			—	20	35	mA	
	I_{CC}	V_{CC}	V_{CC} = +3.0 V±10% F_C = 10 MHz	—	15	40	mA	
	I_{CCS}			—	10	15	mA	
	I_{CCH}		V_{CC} = +5.0 V±10%	—	—	20	µA	
Input pin capacitance	C_{IN}	Other than V_{CC} and V_{SS}	—	—	10	—	pF	
Input leak current	I_{IL}	P73, 74 P86, 87	V_{CC} = 5.5 V V_{SS} < V_I < V_{CC}	–10	—	10	µA	
Leak current for open-drain outputs	I_{leak}	P50 to P57	—	—	0.1	10	µA	

*1: Hysteresis input pins: $\overline{RST}$, $\overline{HST}$

*2: Current values are provisional and are subject to change without notice to allow for improvements to the characteristics and similar.

4. AC Characteristics

(1) Clock Timing

- **When V_{CC} = 5.0 V±10%**

(V_{CC} = 4.5 V to +5.0 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

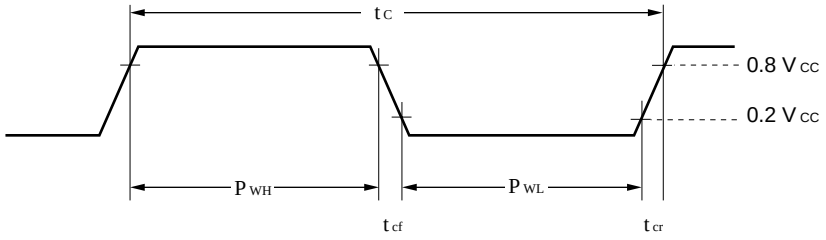
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Clock frequency	F_C	X0, X1	—	3	16	MHz	
Clock cycle time	t_C	X0, X1	—	62.5	333	ns	
Input clock pulse width	P_{WH}, P_{WL}	X0	—	10	—	ns	The duty ratio should be in the range 30 to 70%
Input clock rise time and fall time	t_{cr}, t_{cf}	X0	—	—	5	ns	
Internal operating clock frequency	f_{CP}	—	—	1.5	16	MHz	
Internal operating clock cycle time	t_{CP}	—	—	62.5	333	ns	

- **When V_{CC} = 2.7 V (min.)**

(V_{CC} = 4.5 V to +5.0 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

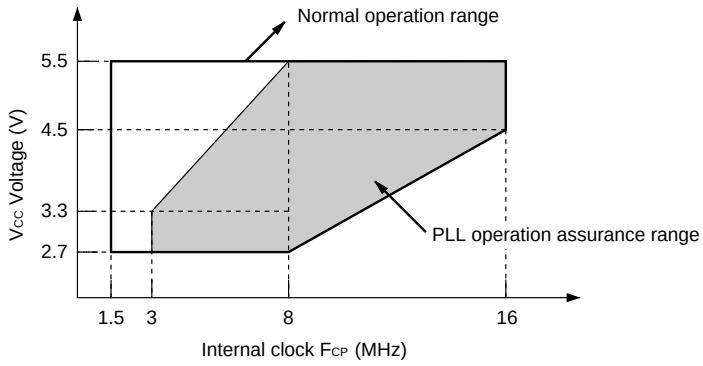
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Clock frequency	F_C	X0, X1	—	3	10	MHz	
Clock cycle time	t_C	X0, X1	—	100	333	ns	
Input clock pulse width	P_{WH}, P_{WL}	X0	—	20	—	ns	The duty ratio should be in the range 30 to 70%
Input clock rise time and fall time	t_{cr}, t_{cf}	X0	—	—	5	ns	
Internal operating clock frequency	f_{CP}	—	—	1.5	8	MHz	
Internal operating clock cycle time	t_{CP}	—	—	100	333	ns	

- **Clock Timing**

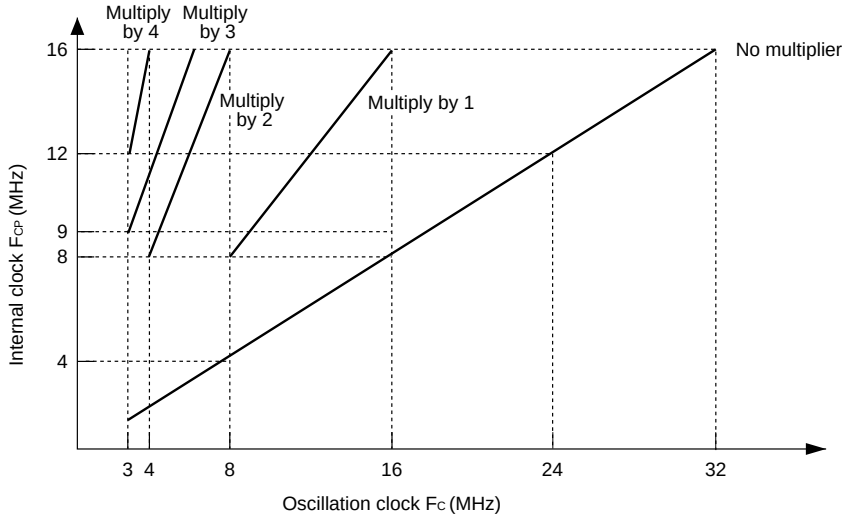


- **PLL Operation Assurance Range**

Relationship between the internal operating clock frequency and suply voltage



Relationship between the oscillation frequency and internal operating clock frequency

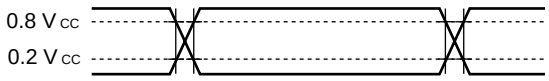


Note: Low voltage operation down to 2.7 V is also assured for the evaluation tools.

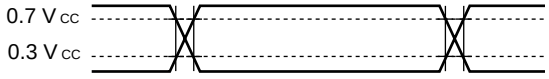
The AC characteristics are for the following measurement reference voltages.

- **Input Signal Waveform**

Hysteresis input pins

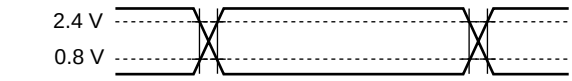


Other than hysteresis or MD input pins



- **Output Signal Waveform**

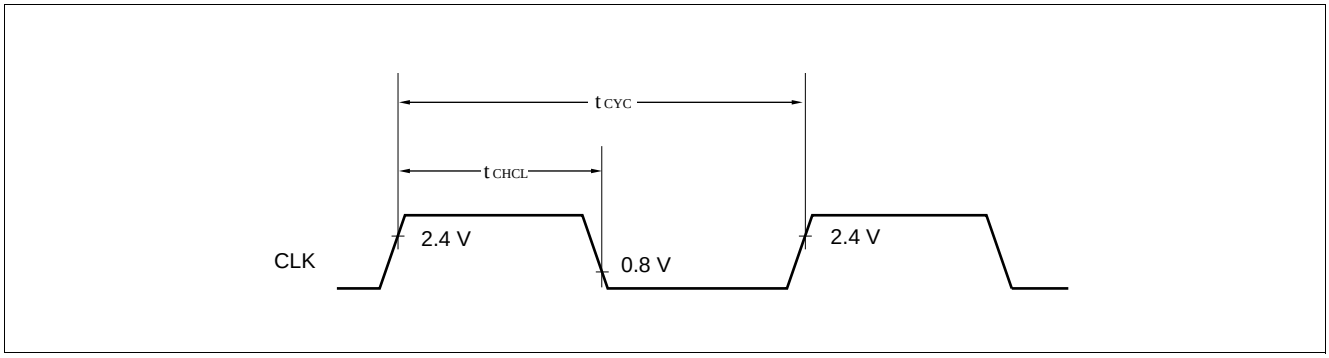
Output pins



(2) Clock Output Timing

(V_{CC} = +2.7 V to +5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

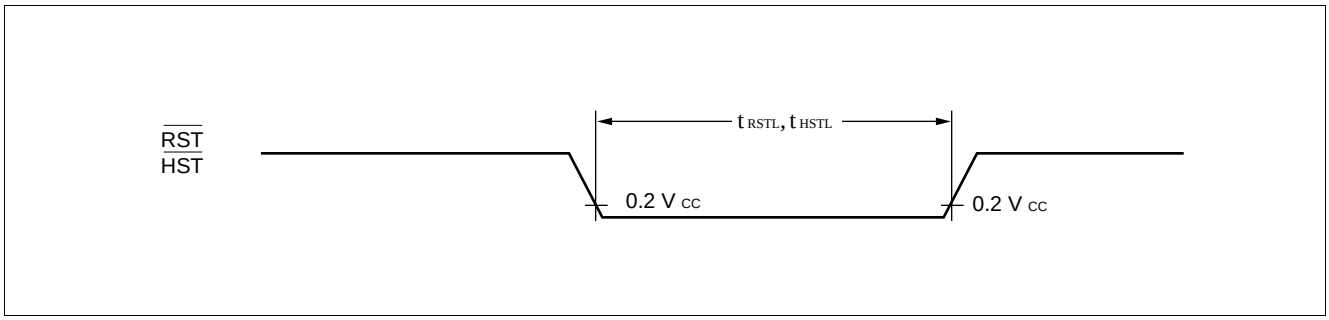
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Cycle time	t_{CYC}	CLK	V_{CC} = 5.0 V±10%	62.5	—	ns	
CLK ↑ → CLK↓	t_{CHCL}			20	—	ns	



(3) Reset and Hardware Standby Inputs

(V_{CC} = +2.7 V to +5.5 V, V_{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}	$\overline{RST}$	—	4	—	Machine cycle	
Hardware standby input time	t_{HSTL}	$\overline{HST}$		4	—	Machine cycle	

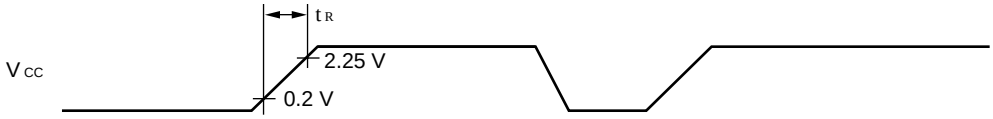


(4) Power-on Reset

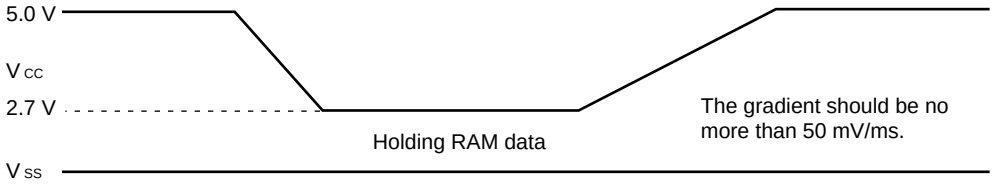
(V_{CC} = +2.7 V to +5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Power supply rising time	t_R	V_{CC}	—	—	30	ms	
Power supply cut-off time	t_{OFF}	V_{CC}		1	—	ms	

Note: The above values are the values required for a power-on reset.



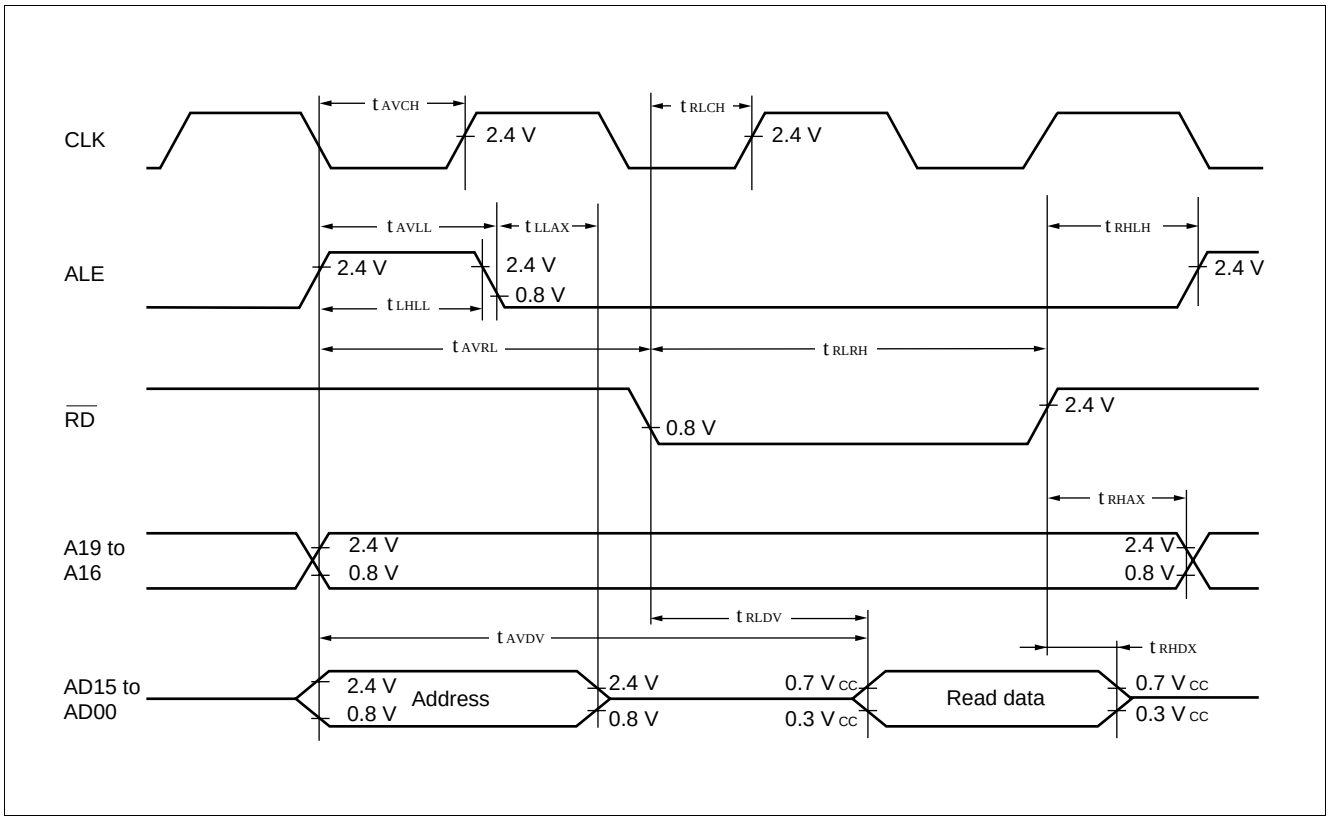
Abrupt changes in the power supply voltage may cause a power-on reset.
When changeing the power supply voltage during operation, suppress variations in the voltage and ensure that the voltage rises smoothly, as shown in the following figure. Also, do not use the PLL clock when varying the voltage. However, the supply voltage can be changed when using the PLL clock if the voltage drops by less than 1 mV/s.



(5) Bus Timing (Read)

(V_{CC} = +2.7 V to +5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

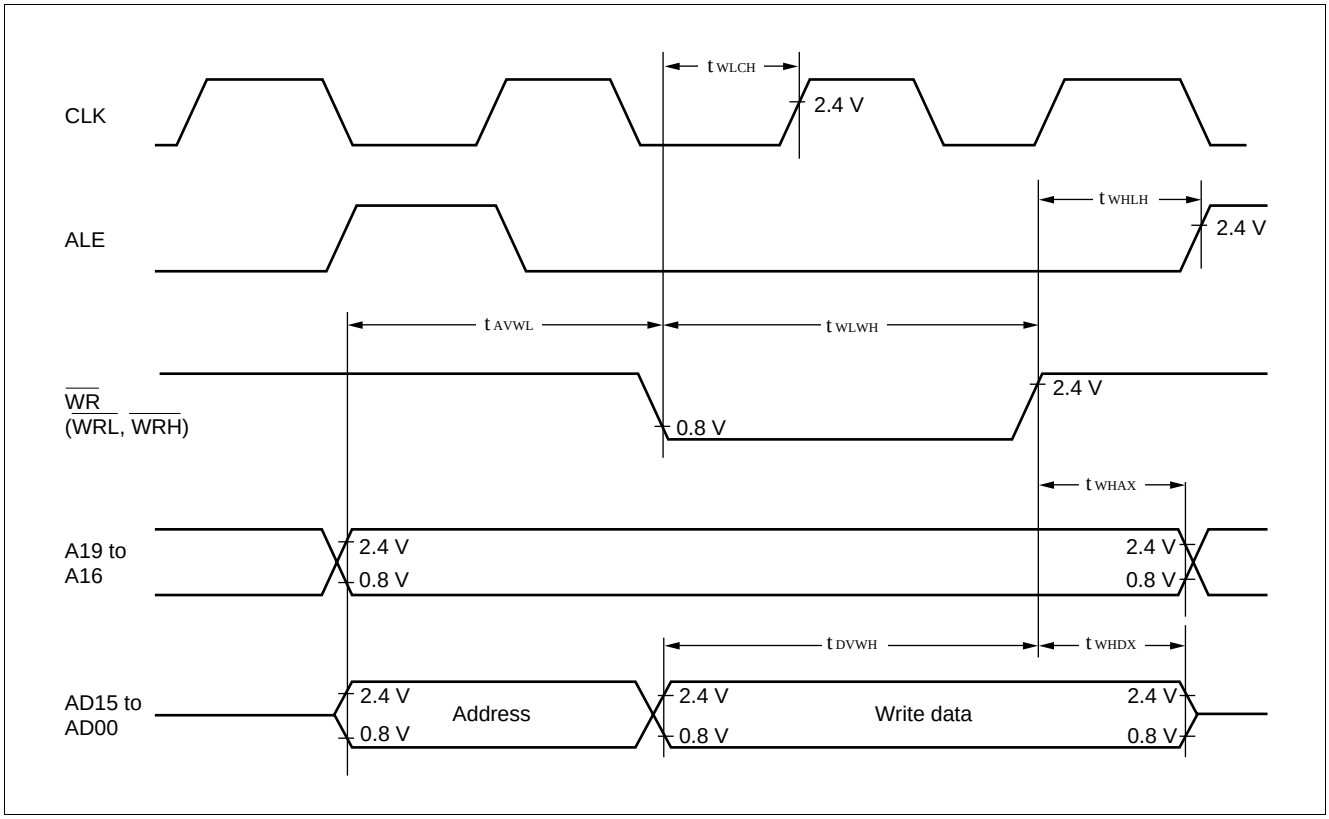
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
ALE pulse width	t_{LHLL}	ALE	—	$t_{CP}/2$ –20	—	ns	
Valid address → ALE ↓ time	t_{AVLL}	Multiplexed address		$t_{CP}/2$ –25	—		
ALE ↓ → address valid time	t_{LLAX}	Multiplexed address		$t_{CP}/2$ –15	—	ns	
Valid address → $\overline{RD}$ ↓ time	t_{AVRL}	Multiplexed address		t_{CP} –15	—		
Valid address→ valid data input	t_{AVDV}	Multiplexed address		—	5 $t_{CP}/2$ –60	ns	
$\overline{RD}$ pulse width	t_{RLRH}	$\overline{RD}$		3 $t_{CP}/2$ –20	—	ns	
$\overline{RD}$ ↓ → valid data input	t_{RLDV}	D15 to D00		—	3 $t_{CP}/2$ –60	ns	
$\overline{RD}$ ↑ → data hold time	t_{RHDX}			0	—	ns	
Valid address → valid data input	t_{AVDV}			0	—	ns	
$\overline{RD}$ ↑ → ALE ↑ time	t_{RHLH}	$\overline{RD}$, ALE		$t_{CP}/2$ –15	—	ns	
$\overline{RD}$ ↑ → address valid time	t_{RHAX}	Address, $\overline{RD}$		$t_{CP}/2$ –10	—	ns	
Valid address → CLK ↑ time	t_{AVCH}	Address, CLK		$t_{CP}/2$ –20	—	ns	
$\overline{RD}$ ↓ → CLK ↑ time	t_{RLCH}	$\overline{RD}$, CLK		$t_{CP}/2$ –20	—	ns	



(6) Bus Timing (Write)

(V_{CC} = +2.7 V to +5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Valid address → $\overline{WR}$ ↓ time	t_{AVWL}	A19 to A00	—	t_{CP}–15	—	ns	
Valid address → $\overline{RD}$ ↓ time	t_{AVRL}	A23 to A00		t_{CP}/2 –15	—	ns	
$\overline{WR}$ pulse width	t_{WLWH}	$\overline{WR}$		3 t_{CP}/2 –20	—	ns	
$\overline{RD}$ pulse width	t_{RLRH}	$\overline{RD}$		3 t_{CP}/2 –20	—	ns	
Valid data output → $\overline{WR}$ ↑ time	t_{DVWH}	D15 to D00		3 t_{CP}/2 –20	—	ns	
$\overline{WR}$ ↑ → data hold time	t_{WHDX}	D15 to D00		20	—	ns	
$\overline{WR}$ ↑ → address valid time	t_{WHAX}	A19 to A00		t_{CP}/2 –10	—	ns	
$\overline{WR}$ ↑ → ALE ↑ time	t_{WHLH}	$\overline{WR}$, ALE		t_{CP}/2 –15	—	ns	
$\overline{WR}$ ↓ → CLK ↑ time	t_{WLCH}	$\overline{WRL}$, $\overline{WRH}$, CLK		t_{CP}/2 –20	—	ns	

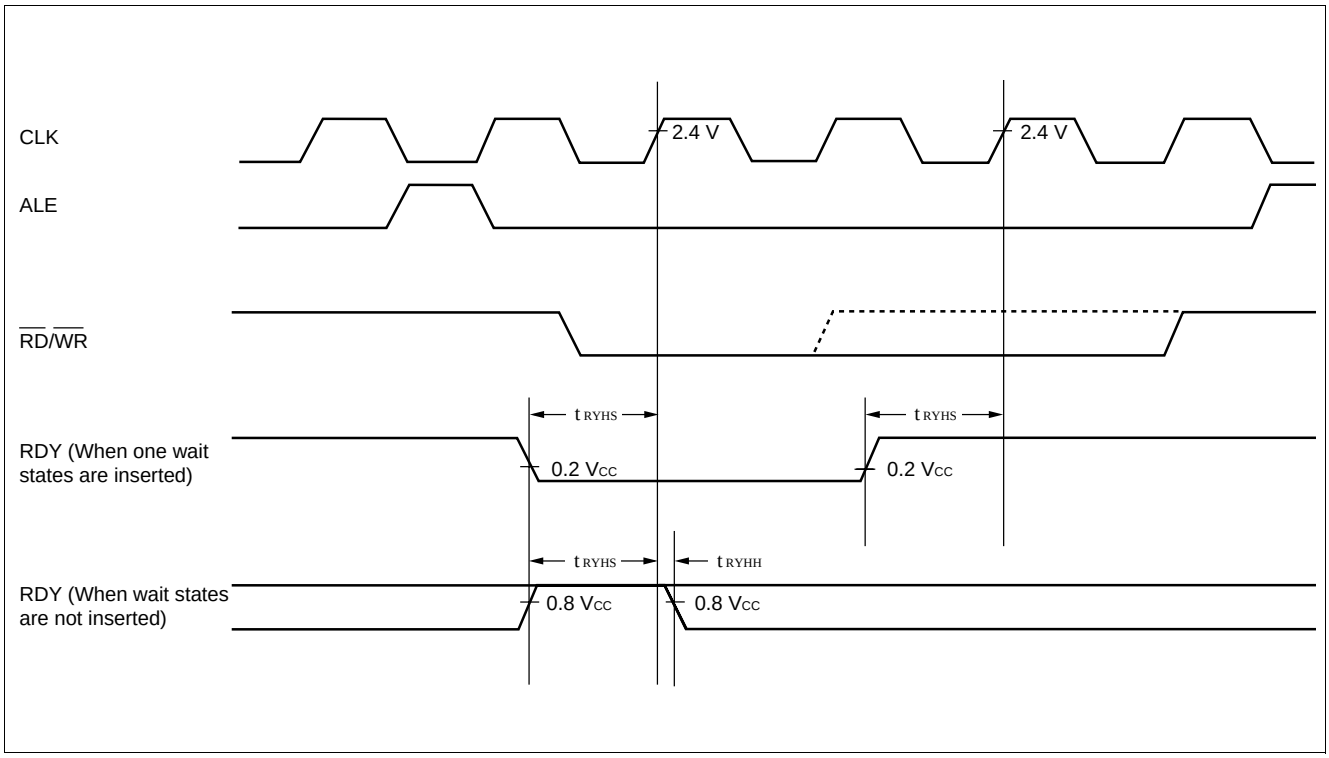


(7) Ready Input Timing

(V_{CC} = +2.7 V to +5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
RDY setup time	t_{RYHS}	RDY	V_{CC} = 5.0 V ±10%	45	—	ns	
			V_{CC} = 3.0 V ±10%	70	—	ns	
RDY hold time	t_{RYHH}		—	0	—	ns	

Note: Use the auto-ready function if the RDY setup time is too short.

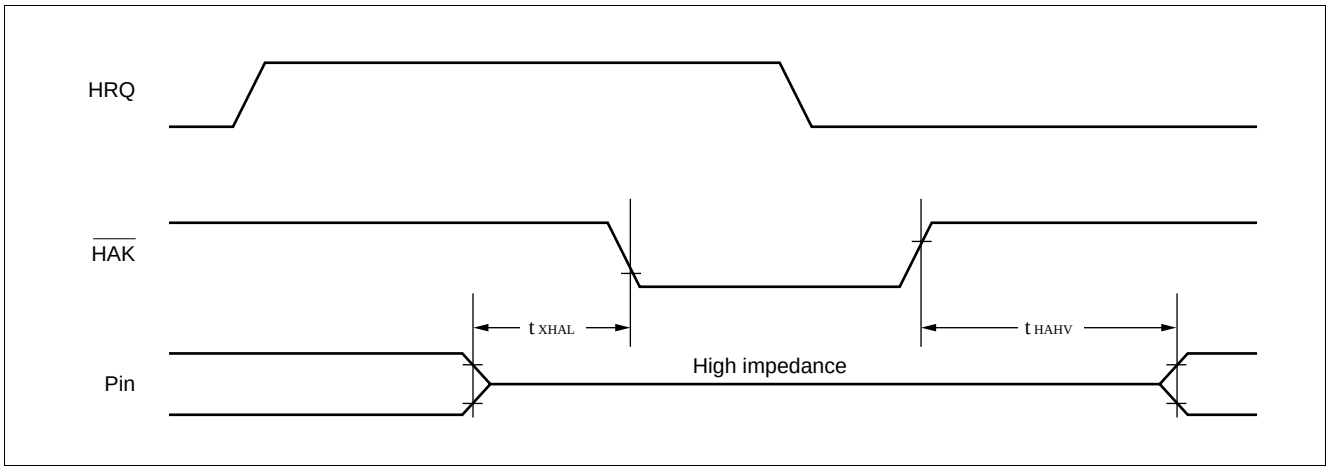


(8) Hold Timing

(V_{CC} = +2.7 V to +5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Pin floating → $\overline{HAK}$ ↓ time	t_{XHAL}	$\overline{HAK}$	—	30	t_{CP}	ns	
$\overline{HAK}$ ↑ → pin valid time	t_{HAHV}	$\overline{HAK}$	—	t_{CP}	2 t_{CP}	ns	

Note: After reading HRQ, more than one cycle is required before changing $\overline{HAK}$.



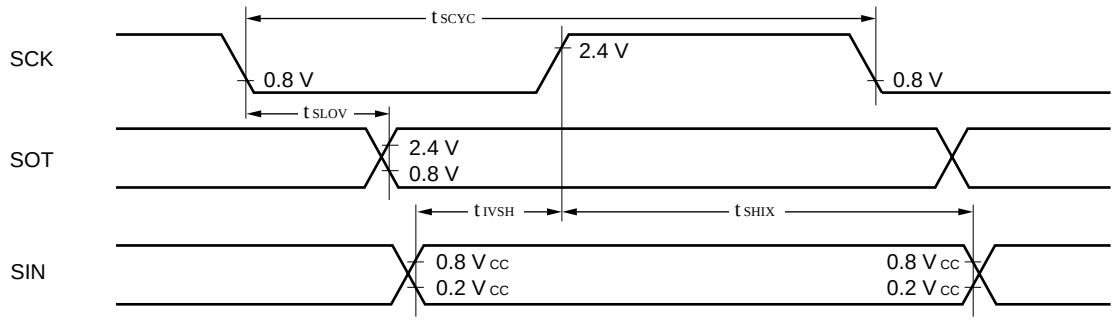
(9) UART Timing

(V_{CC} = +2.7 V to +5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

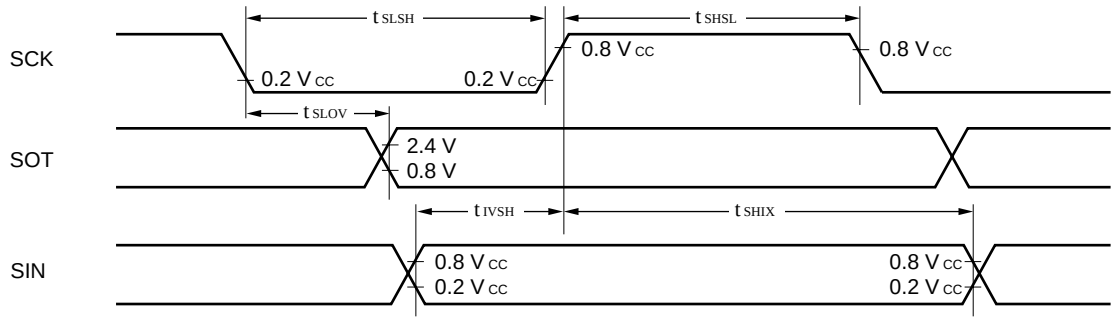
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Serial clock cycle time	t_{SCYC}	—	—	8 t_{CP}	—	ns	
SCK ↓ → SOT delay time	t_{SLOV}	—	V_{CC} = +5.0 V ±10%	–80	80	ns	C_L = 80 pF+1TTL for the internal shift clock mode output pin
			V_{CC} = +3.0 V ±10%	–120	120	ns	
Valid SIN → SCK ↑	t_{IVSH}		V_{CC} = +5.0 V ±10%	100	—	ns	
			V_{CC} = +3.0 V ±10%	200	—	ns	
SCK ↑ → valid SIN hold time	t_{SHIX}		V_{CC} = +5.0 V ±10%	60	—	ns	
			V_{CC} = +3.0 V ±10%	120	—	ns	
Serial clock “H” pulse width	t_{SHSL}		—	4 t_{CP}	—	ns	C_L = 80 pF+1TTL for the external shift clock mode output pin
Serial clock “L” pulse width	t_{SLSH}		—	4 t_{CP}	—	ns	
SCK ↓ → SOT delay time	t_{SLOV}		V_{CC} = +5.0 V ±10%	—	150	ns	
			V_{CC} = +3.0 V ±10%	—	200	ns	
Valid SIN → SCK ↑	t_{IVSH}		V_{CC} = +5.0 V ±10%	60	—	ns	
			V_{CC} = +3.0 V ±10%	120	—	ns	
SCK ↑ → valid SIN hold time	t_{SHIX}		V_{CC} = +5.0 V ±10%	60	—	ns	
			V_{CC} = +3.0 V ±10%	120	—	ns	

Notes: • These are the AC characteristics for CLK synchronous mode.
• C_L is the load capacitance connected to the pin at testing.
• t_{CP} is the machine cycle period (unit: ns).

- **Internal Shift Clock Mode**



- **External Shift Clock Mode**



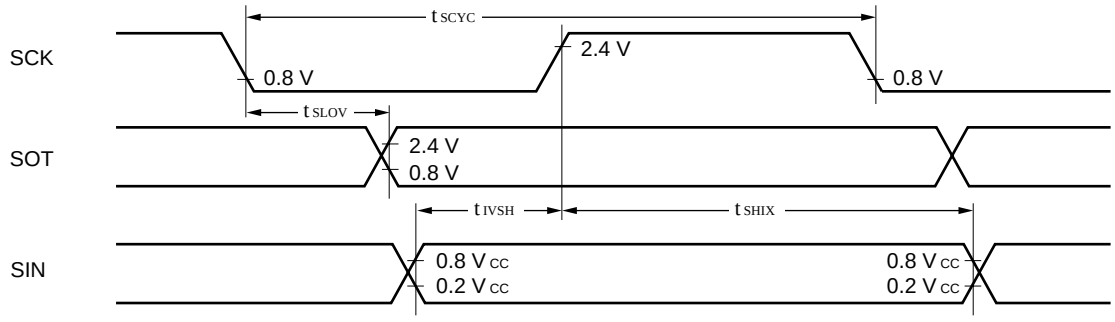
(10) I/O Extended Serial Timing

(V_{CC} = +2.7 V to +5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

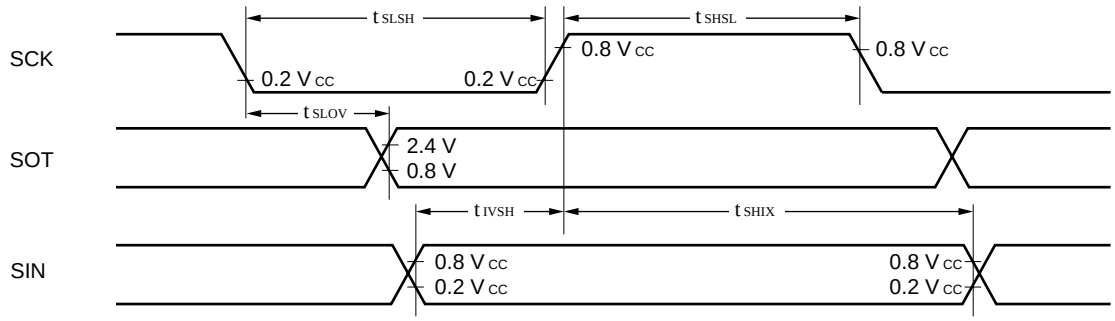
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Serial clock cycle time	t_{SCYC}	—	—	8 t_{CP}	—	ns	C_L = 80 pF+1TTL for the internal shift clock mode output pin
SCK ↓ → SOT delay time	t_{SLOV}	—	V_{CC} = +5.0 V ±10%	—	80	ns	
			V_{CC} = +3.0 V ±10%	—	160	ns	
Valid SIN → SCK ↑	t_{IVSH}	—	—	t_{CP}	—	ns	
SCK ↑ → valid SIN hold time	t_{SHIX}	—	—	t_{CP}	—	ns	
Serial clock “H” pulse width	t_{SHSL}	—	V_{CC} = +5.0 V ±10%	230	—	ns	C_L = 80 pF+1TTL for the external shift clock mode output pin Max. 2 MHz
			V_{CC} = +3.0 V ±10%	460	—	ns	
Serial clock “L” pulse width	t_{SLSH}	—	V_{CC} = +5.0 V ±10%	230	—	ns	
			V_{CC} = +3.0 V ±10%	460	—	ns	
SCK ↓ → SOT delay time	t_{SLOV}	—	—	2 t_{CP}	—	ns	
Valid SIN →SCK ↑	t_{IVSH}	—	—	t_{CP}	—	ns	
SCK ↑ → valid SIN hold time	t_{SHIX}	—	—	2 t_{CP}	—	ns	

Notes: • These are the AC characteristics for CLK synchronous mode.
• C_L is the load capacitance connected to the pin at testing.
• t_{CP} is the machine cycle period (unit: ns).
• The values in the table are target values.

- **Internal Shift Clock Mode**



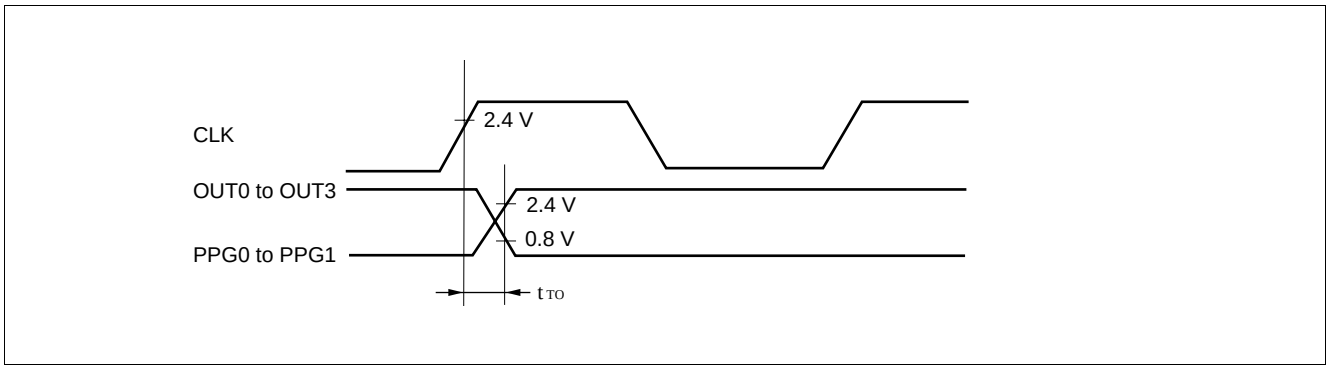
- **External Shift Clock Mode**



(11) Timer Output Timing

(V_{CC} = +2.7 V to +5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

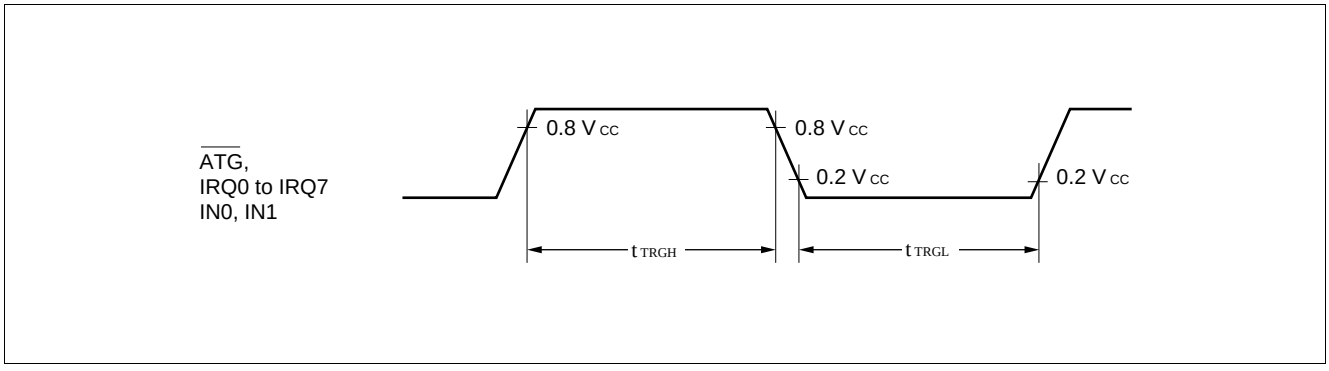
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
SCK ↑ → T_{OUT} change time	t_{TO}	OUT0 to OUT3	V_{CC} = +5.0 V ±10%	30	—	ns	
		PPG00 to PPG11	V_{CC} = +3.0 V ±10%	80	—	ns	



(12) Trigger Input Timing

(V_{CC} = +2.7 V to +5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

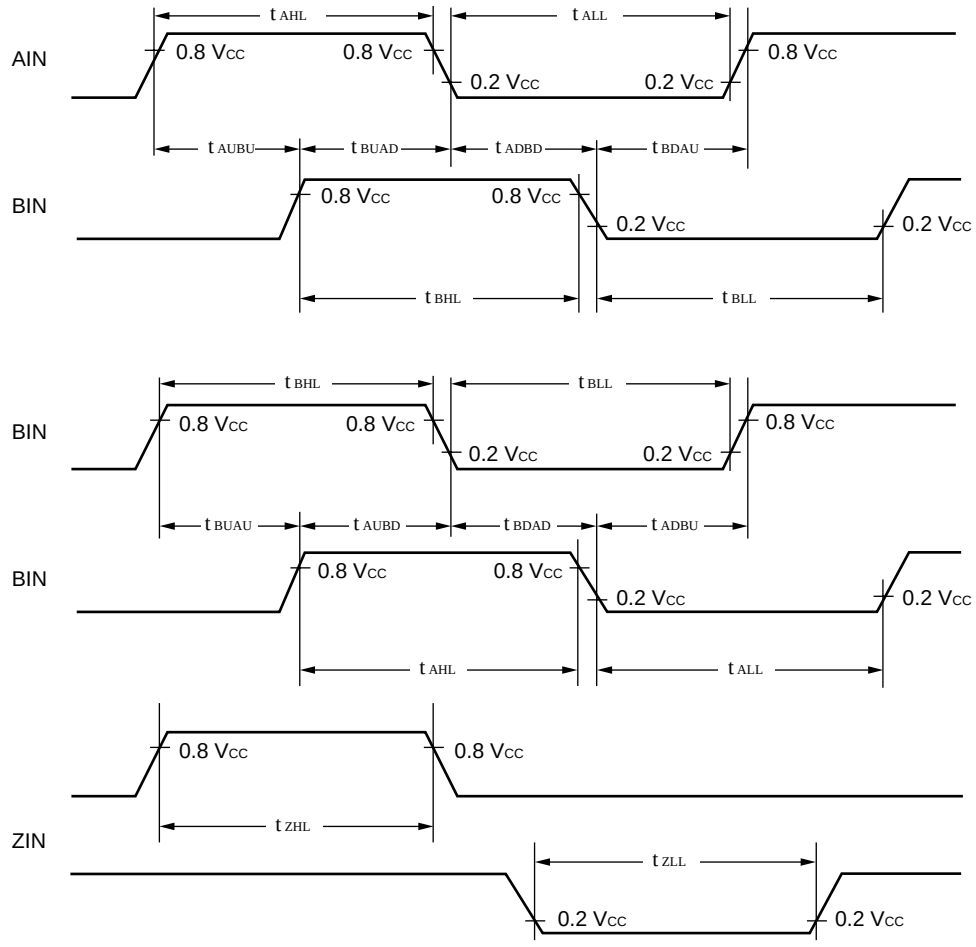
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Input pulse width	t_{TRGH} t_{TRGL}	$\overline{ATG}$, IRQ0 to IRQ7 IN0, IN1	—	5 t_{CP}	—	ns	



(13) Up/down Counter

(V_{CC} = +2.7 V to +5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +85°C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
AIN input “1” pulse width	t_{AHL}	AIN0, AIN1 BIN0, BIN1	—	8 t_{CYL}	—	ns	
AIN input “0” pulse width	t_{ALL}			8 t_{CYL}	—	ns	
BIN input “1” pulse width	t_{BHL}			8 t_{CYL}	—	ns	
BIN input “0” pulse width	t_{BLL}			8 t_{CYL}	—	ns	
AIN ↑ → BIN ↑ time	t_{AUBU}			4 t_{CYL}	—	ns	
BIN ↑ → AIN ↓ time	t_{BUAD}			4 t_{CYL}	—	ns	
AIN ↓ → BIN ↓ time	t_{ADBD}			4 t_{CYL}	—	ns	
BIN ↓ → AIN ↑ time	t_{BDAU}			4 t_{CYL}	—	ns	
BIN ↑ → AIN ↑ time	t_{BUAU}			4 t_{CYL}	—	ns	
AIN ↑ → BIN ↓ time	t_{AUBD}			4 t_{CYL}	—	ns	
BIN ↓ → AIN ↓ time	t_{BDAD}			4 t_{CYL}	—	ns	
AIN ↓ → BIN ↑ time	t_{ADBU}			4 t_{CYL}	—	ns	
ZIN input “1” pulse width	t_{ZHL}	ZIN0, ZIN1		4 t_{CYL}	—	ns	
ZIN input “0” pulse width	t_{ZLL}			4 t_{CYL}	—	ns	


AIN
BIN
BIN
BIN
ZIN
tAHL
tALL
tAUBU
tBUAD
tADBD
tBDAU
tBHL
tBLL
tBUAU
tAUBD
tBDAD
tADBU
tZHL
tZLL
0.8 VCC
0.2 VCC

5. A/D Converter Electrical Characteristics

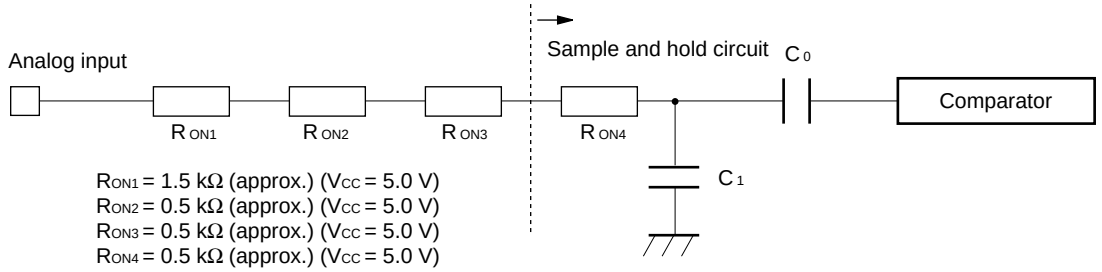
($AV_{CC} = V_{CC}$ = +2.7 V to +5.5 V, $AV_{SS} = V_{SS}$ = 0.0 V, 2.7 V ≤ AVRH – AVRL, T_A = –40°C to +85°C)

Parameter	Symbol	Pin name	Value			Unit
			Min.	Typ.	Max.	
Resolution	—	—	—	10	10	bit
Total error	—	—	—	—	±3.0	LSB
Linearity error	—	—	—	—	±2.0	LSB
Differential linearity error	—	—	—	—	±1.5	LSB
Zero transition error	V_{OT}	AN0 to AN7	–1.5	+0.5	+2.5	LSB
Full scale transition error	V_{FST}	AN0 to AN7	AVRH –3.5	AVRL –1.5	AVRH +0.5	LSB
Conversion time	—	—	5.12*1	—	—	µs
			8.12*2	—	—	µs
Analog port input current	I_{AIN}	AN0 to AN7	—	—	10	µA
Analog input voltage	V_{AIN}	AN0 to AN7	AVRL	—	AVRH	V
Reference voltage	—	AVRH	AVRL + 2.7	—	AV_{CC}	V
	—	AVRL	0	—	AVRH – 2.7	V
Power supply current	I_A	AV_{CC}	—	5	—	mA
	I_{AH}	AV_{CC}	—	—	5*3	µA
Reference voltage supply current	I_R	AVRH	—	200	—	µA
	I_{RH}	AVRH	—	—	5*3	µA
Variation between channels	—	AN0 to AN7	—	—	4	LSB

*1: For V_{CC} = +5.0 V ±10% and a 16 MHz machine clock
*2: For V_{CC} = +3.0 V ±10% and an 8 MHz machine clock
*3: The current when the A/D converter is not operating or the CPU is in stop mode (for $V_{CC} = AV_{CC}$ = AVRH = +5.0 V).

Notes: • The error increases proportionally as |AVRH – AVRL| decreases.
• The output impedance of the external circuits connected to the analog inputs should be in the following range.
Output impedance of external circuit < approx. 10 kΩ
• If the output impedance of the external circuit is too high, the sampling time for the analog voltage may be too short. (Sampling time = 3.8 µs (corresponds to 16 MHz internal operation if the multiplier is 4.))

• **Model of the Analog Input Circuit**

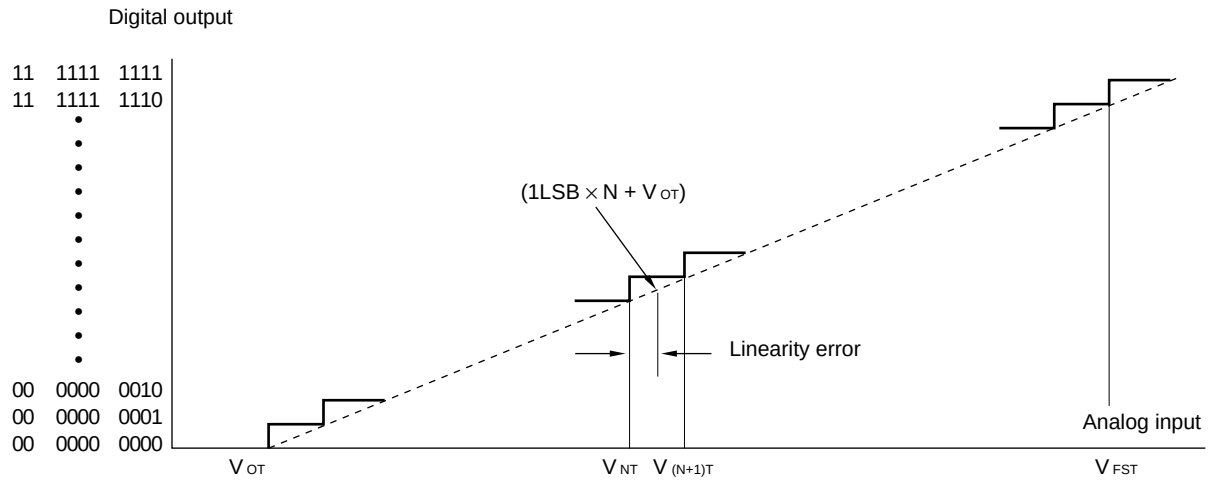


R_{ON1} = 1.5 kΩ (approx.) (V_{CC} = 5.0 V)
R_{ON2} = 0.5 kΩ (approx.) (V_{CC} = 5.0 V)
R_{ON3} = 0.5 kΩ (approx.) (V_{CC} = 5.0 V)
R_{ON4} = 0.5 kΩ (approx.) (V_{CC} = 5.0 V)
C_0 = 60 pF (approx.)
C_1 = 4 pF (approx.)

Note: The above values are for reference only.

6. A/D Converter Glossary

- Resolution
  The change in analog voltage that can be recognized by the A/D converter.
  If the resolution is 10 bits, the analog voltage can be resolved into 2^{10} = 1024 steps.
- Total error
  The deviation between the actual and logic value attributable to offset error, gain error, non-linearity error, and noise.
- Linearity error
  The deviation between the actual conversion characteristic of the device and the line linking the zero transition point (00 0000 0000 ↔ 00 0000 0001) and the full scale transition point (11 1111 1110 ↔ 11 1111 1111).
- Differential linearity error
  The variation from the ideal input voltage required to change the output code by 1 LSB.



$$1LSB = \frac{V_{FST} - V_{OT}}{1022}$$

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

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

7. 8-bit D/A Converter Electrical Characteristics

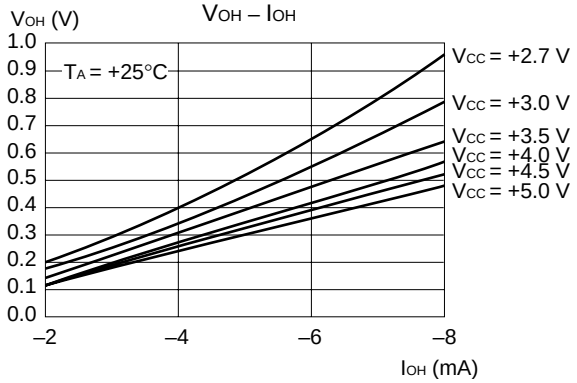
(V_{CC} = 2.7 to 5.5 V, V_{SS} = 0.0 V, T_A = –40°C to +70°C)

Parameter	Symbol	Pin name	Value			Unit	Remarks
			Min.	Typ.	Max.		
Resolution	—	—	—	8	8	bit	
Differential linearity error	—	—	–0.9	—	0.9	LSB	
Absolute accuracy	—	—	—	—	1.2	%	
Conversion time	—	—	—	10	20	µS	The load capacitance = 20 pF
Analog reference power supply voltage	—	DVRH	V_{SS} + 1.7	—	V_{CC}	V	DV_{SS} = V_{SS} = 0.0 V
Reference power supply current (when operating)	I_D	DVRH	—	1.0	1.5	mA	Current consumption at conversion
Reference power supply current (when stopped)	I_{DH}	DVRH	—	—	10	µA	Current consumption when stopped
Analog output impedance	—	DA0	—	28	—	kΩ	

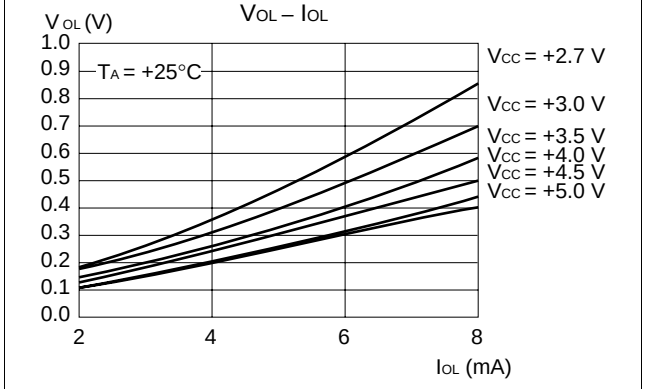
Note: DV_{SS} must be connected at V_{SS} = 0.0 V.

■ EXAMPLE CHARACTERISTICS

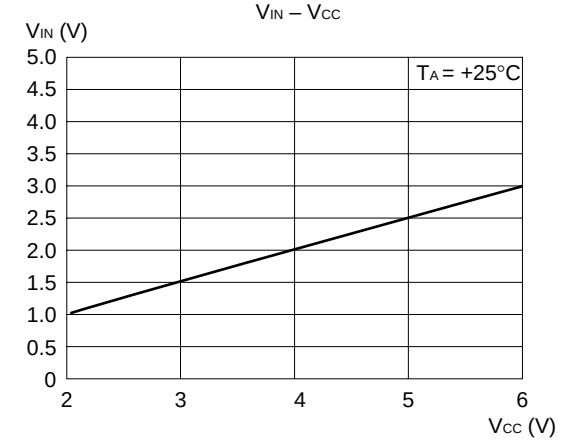
(1) “H” Level Output Voltage



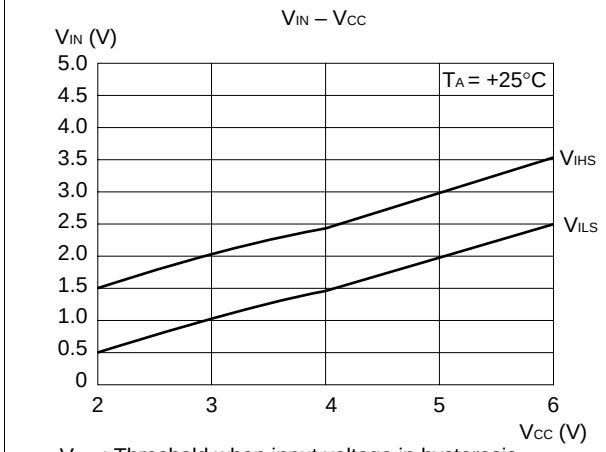
(2) “L” Level Output Voltage



(3) “H” Level Input Voltage/“L” Level Input Voltage (CMOS Input)



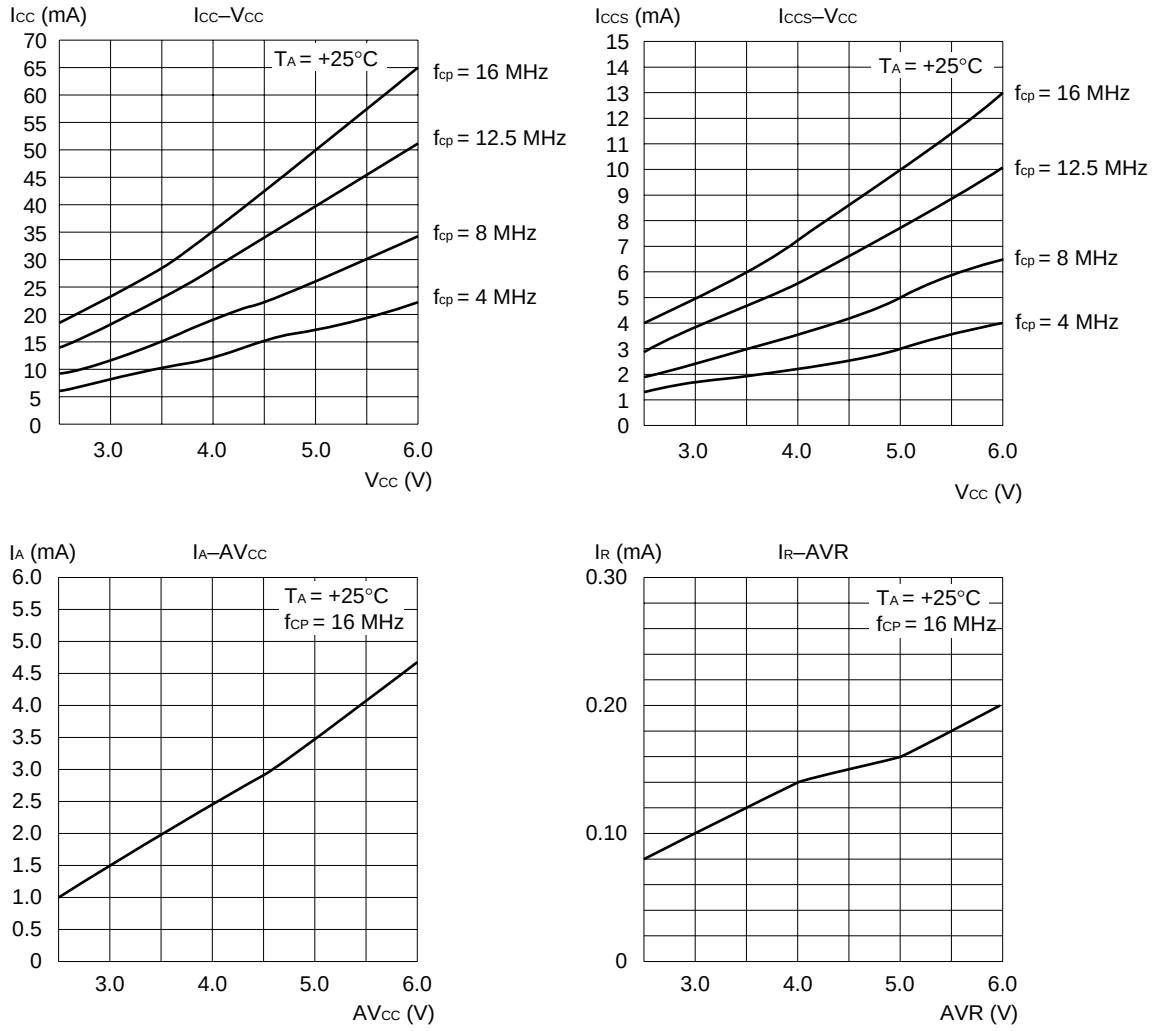
(4) “H” Level Input Voltage/“L” Level Input Voltage (Hysteresis Input)



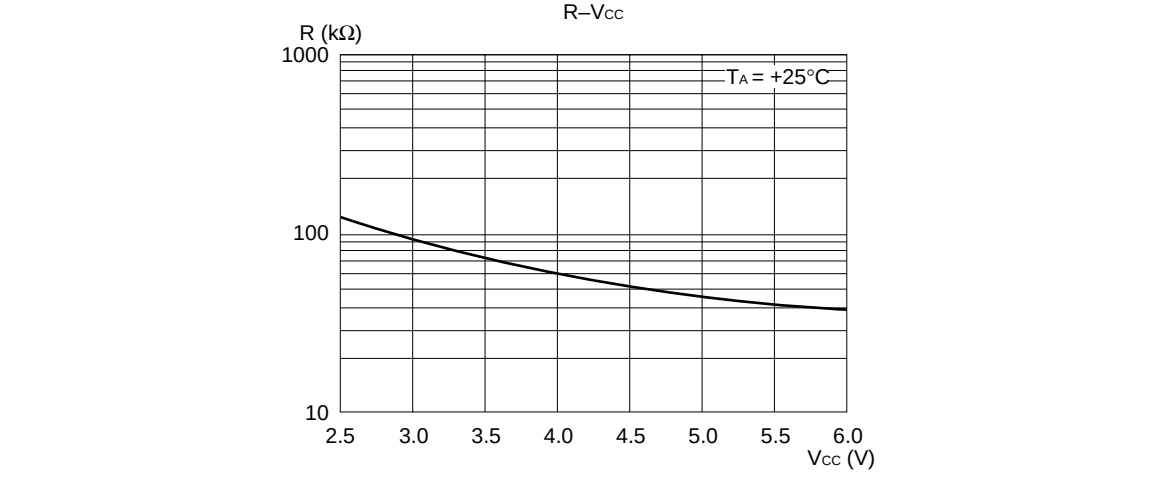
VIHS : Threshold when input voltage in hysteresis characteristics is set to “H” level

VILS : Threshold when input voltage in hysteresis characteristics is set to “L” level

(5) Power Supply Current (f_{CP} = Internal Operating Clock Frequency)



(5) Pull-up Resistance



■ INSTRUCTIONS (340 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.
#	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	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.
S	
T	
N	
Z	
V	
C	
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.

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: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

(Continued)

(Continued)

Symbol	Meaning
rel	Branch specification relative to PC
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 RW0 RL0 R1 RW1 (RL0) R2 RW2 RL1 R3 RW3 (RL1) R4 RW4 RL2 R5 RW5 (RL2) R6 RW6 RL3 R7 RW7 (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 Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles

Operand	(b) byte		(c) word		(d) long	
	Number of cycles	Number of access	Number of cycles	Number of access	Number of cycles	Number of 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) ←	Z	*	–	–	–	*	*	–	–	–
MOVN A, #imm4	1	1	0	0	((RLi)+disp8) byte (A) ← imm4	Z	*	–	–	–	R	*	–	–	–
MOVX A, dir	2	3	0	(b)		X	*	–	–	–	*	*	–	–	–
MOVX A, addr16	3	4	0	(b)	byte (A) ← (dir)	X	*	–	–	–	*	*	–	–	–
MOVX A, Ri	2	2	1	0	byte (A) ← (addr16)	X	*	–	–	–	*	*	–	–	–
MOVX A, ear	2	2	1	0	byte (A) ← (Ri)	X	*	–	–	–	*	*	–	–	–
MOVX A, eam	2+	3+ (a)	0	(b)	byte (A) ← (ear)	X	*	–	–	–	*	*	–	–	–
MOVX A, io	2	3	0	(b)	byte (A) ← (eam)	X	*	–	–	–	*	*	–	–	–
MOVX A, #imm8	2	2	0	0	byte (A) ← (io)	X	*	–	–	–	*	*	–	–	–
MOVX A, @A	2	3	0	(b)	byte (A) ← imm8	X	–	–	–	–	*	*	–	–	–
MOVX A,@RWi+disp8	2	5	1	(b)	byte (A) ← ((A))	X	*	–	–	–	*	*	–	–	–
MOVX A, @RLi+disp8	3	10	2	(b)	byte (A) ← ((RWi)+disp8)	X	*	–	–	–	*	*	–	–	–
MOV dir, A	2	3	0	(b)	byte (A) ←	–	–	–	–	–	*	*	–	–	–
MOV addr16, A	3	4	0	(b)	((RLi)+disp8)	–	–	–	–	–	*	*	–	–	–
MOV Ri, A	1	2	1	0		–	–	–	–	–	*	*	–	–	–
MOV ear, A	2	2	1	0	byte (dir) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV eam, A	2+	3+ (a)	0	(b)	byte (addr16) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV io, A	2	3	0	(b)	byte (Ri) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV @RLi+disp8, A	3	10	2	(b)	byte (ear) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV Ri, ear	2	3	2	0	byte (eam) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV Ri, eam	2+	4+ (a)	1	(b)	byte (io) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV ear, Ri	2	4	2	0	byte ((RLi) +disp8) ←	–	–	–	–	–	*	*	–	–	–
MOV eam, Ri	2+	5+ (a)	1	(b)	(A)	–	–	–	–	–	*	*	–	–	–
MOV Ri, #imm8	2	2	1	0	byte (Ri) ← (ear)	–	–	–	–	–	*	*	–	–	–
MOV io, #imm8	3	5	0	(b)	byte (Ri) ← (eam)	–	–	–	–	–	–	–	–	–	–
MOV dir, #imm8	3	5	0	(b)	byte (ear) ← (Ri)	–	–	–	–	–	–	–	–	–	–
MOV ear, #imm8	3	2	1	0	byte (eam) ← (Ri)	–	–	–	–	–	*	*	–	–	–
MOV eam, #imm8	3+	4+ (a)	0	(b)	byte (Ri) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV @AL, AH /MOV @A, T	2	3	0	(b)	byte (io) ← imm8 byte (dir) ← imm8 byte (ear) ← imm8	–	–	–	–	–	*	*	–	–	–
XCH A, ear	2	4	2	0	byte (eam) ← imm8	Z	–	–	–	–	–	–	–	–	–
XCH A, eam	2+	5+ (a)	0	2× (b)	byte ((A)) ← (AH)	Z	–	–	–	–	–	–	–	–	–
XCH Ri, ear	2	7	4	0		–	–	–	–	–	–	–	–	–	–
XCH Ri, eam	2+	9+ (a)	2	2× (b)	byte (A) ↔ (ear) byte (A) ↔ (eam) 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 5, "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)	–	*	–	–	–	*	*	–	–	–
MOVW A, @RLi+disp8	3	10	2	(c)	+disp8) word (A) ← ((RLi)	–	*	–	–	–	*	*	–	–	–
MOVW dir, A	2	3	0	(c)	+disp8)	–	–	–	–	–	*	*	–	–	–
MOVW addr16, A	3	4	0	(c)		–	–	–	–	–	*	*	–	–	–
MOVW SP, A	1	1	0	0	word (dir) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, A	1	2	1	0	word (addr16) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW ear, A	2	2	1	0	word (SP) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW eam, A	2+	3+ (a)	0	(c)	word (RWi) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW io, A	2	3	0	(c)	word (ear) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW @RWi+disp8, A	2	5	1	(c)	word (eam) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW @RLi+disp8, A	3	10	2	(c)	word (io) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, ear	2	3	2	(0)	word ((RWi) +disp8) ←	–	–	–	–	–	*	*	–	–	–
MOVW RWi, eam	2+	4+ (a)	1	(c)	(A)	–	–	–	–	–	*	*	–	–	–
MOVW ear, RWi	2	4	2	0	word ((RLi) +disp8) ←	–	–	–	–	–	*	*	–	–	–
MOVW eam, RWi	2+	5+ (a)	1	(c)	(A)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, #imm16	3	2	1	0	word (RWi) ← (ear)	–	–	–	–	–	*	*	–	–	–
MOVW io, #imm16	4	5	0	(c)	word (RWi) ← (eam)	–	–	–	–	–	–	–	–	–	–
MOVW ear, #imm16	4	2	1	0	word (ear) ← (RWi)	–	–	–	–	–	*	*	–	–	–
MOVW eam, #imm16	4+	4+ (a)	0	(c)	word (eam) ← (RWi) word (RWi) ← imm16	–	–	–	–	–	–	–	–	–	–
MOVW AL, AH /MOVW @A, T	2	3	0	(c)	word (io) ← imm16 word (ear) ← imm16 word (eam) ← imm16	–	–	–	–	–	*	*	–	–	–
XCHW A, ear	2	4	2	0		–	–	–	–	–	–	–	–	–	–
XCHW A, eam	2+	5+ (a)	0	2× (c)	word ((A)) ← (AH)	–	–	–	–	–	–	–	–	–	–
XCHW RWi, ear	2	7	4	0		–	–	–	–	–	–	–	–	–	–
XCHW RWi, eam	2+	9+ (a)	2	2× (c)		–	–	–	–	–	–	–	–	–	–
					word (A) ↔ (ear) word (A) ↔ (eam) word (RWi) ↔ (ear) 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 5, “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	–	–	–	–	*	*	*	*	–
	2	5	0	(b)	byte (A) ← (A) +(dir)	Z	–	–	–	–	*	*	*	*	–
ADD A, dir	2	3	1	0	byte (A) ← (A) +(ear)	Z	–	–	–	–	*	*	*	*	–
ADD A, ear	2+	4+ (a)	0	(b)	byte (A) ← (A) +(eam)	Z	–	–	–	–	*	*	*	*	–
ADD A, eam	2	3	2	0	byte (ear) ← (ear) + (A)	–	–	–	–	–	*	*	*	*	–
ADD ear, A	2+	5+ (a)	0	2× (b)	byte (eam) ← (eam) + (A)	Z	–	–	–	–	*	*	*	*	*
ADD eam, A	1	2	0	0	byte (A) ← (AH) + (AL) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDC A	2	3	1	0	byte (A) ← (A) + (ear) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDC A, ear	2+	4+ (a)	0	(b)	byte (A) ← (A) + (eam) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDC A, eam	1	3	0	0	byte (A) ← (AH) + (AL) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDDC A	2	2	0	0	(decimal)	Z	–	–	–	–	*	*	*	*	–
SUB A,	2	5	0	(b)	byte (A) ← (A) –imm8	Z	–	–	–	–	*	*	*	*	–
#imm8	2	3	1	0	byte (A) ← (A) – (dir)	Z	–	–	–	–	*	*	*	*	–
SUB A, dir	2+	4+ (a)	0	(b)	byte (A) ← (A) – (ear)	Z	–	–	–	–	*	*	*	*	–
SUB A, ear	2	3	2	0	byte (A) ← (A) – (eam)	–	–	–	–	–	*	*	*	*	–
SUB A, eam	2+	5+ (a)	0	2× (b)	byte (ear) ← (ear) – (A)	–	–	–	–	–	*	*	*	*	*
SUB ear, A	1	2	0	0	byte (eam) ← (eam) – (A)	Z	–	–	–	–	*	*	*	*	–
SUB eam, A	2	3	1	0	byte (A) ← (AH) – (AL) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBC A	2+	4+ (a)	0	(b)	byte (A) ← (A) – (ear) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBC A, ear	1	3	0	0	byte (A) ← (A) – (eam) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBC A, eam					byte (A) ← (AH) – (AL) – (C)										
SUBDC A					(decimal)										
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,	3	2	0	0	word (A) ← (A) +imm16	–	–	–	–	–	*	*	*	*	–
#imm16	2	3	2	0	word (ear) ← (ear) + (A)	–	–	–	–	–	*	*	*	*	–
ADDW ear, A	2+	5+ (a)	0	2× (c)	word (eam) ← (eam) + (A)	–	–	–	–	–	*	*	*	*	*
ADDW eam, A	2	3	1	0	word (A) ← (A) + (ear) + (C)	–	–	–	–	–	*	*	*	*	–
ADDCW A, ear	2+	4+ (a)	0	(c)	word (A) ← (A) + (eam) + (C)	–	–	–	–	–	*	*	*	*	–
ADDCW A, eam	1	2	0	0	word (A) ← (AH) – (AL)	–	–	–	–	–	*	*	*	*	–
SUBW A	2	3	1	0	word (A) ← (A) – (ear)	–	–	–	–	–	*	*	*	*	–
SUBW A, ear	2+	4+ (a)	0	(c)	word (A) ← (A) – (eam)	–	–	–	–	–	*	*	*	*	–
SUBW A, eam	3	2	0	0	word (A) ← (A) –imm16	–	–	–	–	–	*	*	*	*	–
SUBW A,	2	3	2	0	word (ear) ← (ear) – (A)	–	–	–	–	–	*	*	*	*	–
#imm16	2+	5+ (a)	0	2× (c)	word (eam) ← (eam) – (A)	–	–	–	–	–	*	*	*	*	*
SUBW ear, A	2	3	1	0	word (A) ← (A) – (ear) – (C)	–	–	–	–	–	*	*	*	*	–
SUBW eam, A	2+	4+ (a)	0	(c)	word (A) ← (A) – (eam) – (C)	–	–	–	–	–	*	*	*	*	–
SUBCW A, ear															
SUBCW A, eam															
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,	5	4	0	0	long (A) ← (A) +imm32	–	–	–	–	–	*	*	*	*	–
#imm32	2	6	2	0	long (A) ← (A) – (ear)	–	–	–	–	–	*	*	*	*	–
SUBL A, ear	2+	7+ (a)	0	(d)	long (A) ← (A) – (eam)	–	–	–	–	–	*	*	*	*	–
SUBL A, eam	5	4	0	0	long (A) ← (A) –imm32	–	–	–	–	–	*	*	*	*	–
SUBL 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 5, “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 5, “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 (eam)	–	–	–	–	–	–	–	*	*	–
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 5, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 13 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 5, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 14 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	–	–

Table 15 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)	–	–	–	–	–	*	*	*	*	*

Table 16 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 5, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 17 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+	0	2× (b)	byte (eam) ← Right rotation with	–	–	–	–	–	*	*	–	*	*
ROLC ear	2	(a)	2	0	carry	–	–	–	–	–	*	*	–	*	–
ROLC eam	2+	3	0	2× (b)	byte (ear) ← Left rotation with carry	–	–	–	–	–	*	*	–	*	*
		5+			byte (eam) ← Left rotation with carry										
ASR A, R0	2	(a)	1	0		–	–	–	–	*	*	*	–	*	–
LSR A, R0	2		1	0	byte (A) ← Arithmetic right barrel shift (A,	–	–	–	–	*	*	*	–	*	–
LSL A, R0	2	*1	1	0	R0)	–	–	–	–	–	*	*	–	*	–
		*1			byte (A) ← Logical right barrel shift (A, R0)										
		*1			byte (A) ← Logical left barrel shift (A, R0)										
ASRW A	1	2	0	0	word (A) ← Arithmetic right shift (A, 1	–	–	–	–	*	*	*	–	*	–
LSRW A/	1	2	0	0	bit)	–	–	–	–	*	R	*	–	*	–
SHRW A	1	2	0	0	word (A) ← Logical right shift (A, 1	–	–	–	–	–	*	*	–	*	–
LSLW A/					bit)										
SHLW A	2	*1	1	0	word (A) ← Logical left shift (A, 1 bit)	–	–	–	–	*	*	*	–	*	–
	2	*1	1	0		–	–	–	–	*	*	*	–	*	–
ASRW A, R0	2	*1	1	0	word (A) ← Arithmetic right barrel shift (A,	–	–	–	–	–	*	*	–	*	–
LSRW A, R0					R0)										
LSLW A, R0					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,	–	–	–	–	*	*	*	–	*	–
LSRL A, R0	2	*2	1	0	R0)	–	–	–	–	*	*	*	–	*	–
LSLL A, R0	2	*2	1	0	long (A) ← Logical right barrel shift (A, R0)	–	–	–	–	–	*	*	–	*	–
					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 5, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 18 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	–	–	–	–	–	–	–	–	–	–
BGT rel	2	*1	0	0	(Z) = 1	–	–	–	–	–	–	–	–	–	–
BLS rel	2	*1	0	0	Branch when ((V) xor (N)) or	–	–	–	–	–	–	–	–	–	–
BHI rel	2	*1	0	0	(Z) = 0	–	–	–	–	–	–	–	–	–	–
BRA rel	2	*1	0	0	Branch when (C) or (Z) = 1 Branch when (C) or (Z) = 0	–	–	–	–	–	–	–	–	–	–
JMP @A	1	2	0	0	Branch unconditionally	–	–	–	–	–	–	–	–	–	–
JMP addr16	3	3	0	0		–	–	–	–	–	–	–	–	–	–
JMP @ear	2	3	1	0	word (PC) ← (A)	–	–	–	–	–	–	–	–	–	–
JMP @eam	2+	4+ (a)	0	(c)	word (PC) ← addr16	–	–	–	–	–	–	–	–	–	–
JMPP @ear *3	2	5	2	0	word (PC) ← (ear)	–	–	–	–	–	–	–	–	–	–
JMPP @eam *3	2+	6+ (a)	0	(d)	word (PC) ← (eam)	–	–	–	–	–	–	–	–	–	–
JMPP addr24	4	4	0	0	word (PC) ← (ear), (PCB) ← (ear +2)	–	–	–	–	–	–	–	–	–	–
CALL @ear *4	2	6	1	(c)	word (PC) ← (eam), (PCB) ←	–	–	–	–	–	–	–	–	–	–
CALL @eam *4	2+	7+ (a)	0	2× (c)	(eam +2)	–	–	–	–	–	–	–	–	–	–
CALL addr16 *5	3	6	0	(c)	word (PC) ← ad24 0 to 15,	–	–	–	–	–	–	–	–	–	–
CALLV #vct4 *5	1	7	0	2× (c)	(PCB) ← ad24 16 to 23	–	–	–	–	–	–	–	–	–	–
CALLP @ear *6	2	10	2	2× (c)	word (PC) ← (ear) word (PC) ← (eam)	–	–	–	–	–	–	–	–	–	–
CALLP @eam *6	2+	11+ (a)	0	*2	word (PC) ← addr16 Vector call instruction	–	–	–	–	–	–	–	–	–	–
CALLP addr24 *7	4	10	0	2× (c)	word (PC) ← (ear) 0 to 15 (PCB) ← (ear) 16 to 23 word (PC) ← (eam) 0 to 15 (PCB) ← (eam) 16 to 23 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 5, "Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles."

Table 19 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) ≠	–	–	–	–	–	*	*	*	*	–
CWBNEA, #imm16, rel	4	*1	0	0	imm8	–	–	–	–	–	*	*	*	*	–
					Branch when word (A) ≠										
CBNE ear, #imm8, rel	4	*2	1	0	imm16	–	–	–	–	–	*	*	*	*	–
CBNE eam, #imm8,	4+	*3	0	(b)		–	–	–	–	–	*	*	*	*	–
rel*9	5	*4	1	0	Branch when byte (ear) ≠	–	–	–	–	–	*	*	*	*	–
CWBNEear, #imm16,	5+	*3	0	(c)	imm8	–	–	–	–	–	*	*	*	*	–
rel					Branch when byte (eam) ≠										
CWBNEeam, #imm16,	3	*5	2	0	imm8	–	–	–	–	–	*	*	*	–	–
rel*9					Branch when word (ear) ≠										
	3+	*6	2	2× (b)	imm16	–	–	–	–	–	*	*	*	–	*
DBNZ ear, rel					Branch when word (eam) ≠										
					imm16										
DBNZ eam, rel	3	*5	2	0		–	–	–	–	–	*	*	*	–	–
					Branch when byte (ear) =										
	3+	*6	2	2× (c)	(ear) – 1, and (ear) ≠ 0	–	–	–	–	–	*	*	*	–	*
DWBNZ ear, rel					Branch when byte (eam) =										
					(eam) – 1, and (eam) ≠ 0										
DWBNZ eam, rel	2	20	0	8× (c)		–	–	R	S	–	–	–	–	–	–
	3	16	0	6× (c)	Branch when word (ear) =	–	–	R	S	–	–	–	–	–	–
	4	17	0	6× (c)	(ear) – 1, and (ear) ≠ 0	–	–	R	S	–	–	–	–	–	–
INT #vct8	1	20	0	8× (c)	Branch when word (eam) =	–	–	R	S	–	–	–	–	–	–
INT addr16	1	15	0	6× (c)	(eam) – 1, and (eam) ≠ 0	–	–	*	*	*	*	*	*	*	–
INTP addr24															
INT9	2	6	0	(c)	Software interrupt	–	–	–	–	–	–	–	–	–	–
RETI					Software interrupt										
					Software interrupt										
LINK #local8					Software interrupt										
	1	5	0	(c)	Return from interrupt	–	–	–	–	–	–	–	–	–	–
					At constant entry, save old										
UNLINK	1	4	0	(c)	frame pointer to stack, set	–	–	–	–	–	–	–	–	–	–
	1	6	0	(d)	new frame pointer, and	–	–	–	–	–	–	–	–	–	–
					allocate local pointer area										
RET *7					At constant entry, retrieve										
RETP *8					old frame pointer from stack.										
					Return from subroutine										
					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: Retrieve (word) from stack
*8: Retrieve (long word) from stack
*9: 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 5, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 20 Other Control Instructions (Byte/Word/Long Word) [36 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 5, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 21 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	*	–	–	–	*	*	–	–	–
	3	7	0	2× (b)	bit (dir:bp) b ← (A)	–	–	–	–	–	*	*	–	–	*
MOVB dir:bp, A	4	7	0	2× (b)	bit (addr16:bp) b ← (A)	–	–	–	–	–	*	*	–	–	*
MOVB addr16:bp, A	3	6	0	2× (b)	bit (io:bp) b ← (A)	–	–	–	–	–	*	*	–	–	*
MOVB io:bp, A	3	7	0	2× (b)	bit (dir:bp) b ← 1	–	–	–	–	–	–	–	–	–	*
	4	7	0	2× (b)	bit (addr16:bp) b ← 1	–	–	–	–	–	–	–	–	–	*
SETB dir:bp	3	7	0	2× (b)	bit (io:bp) b ← 1	–	–	–	–	–	–	–	–	–	*
SETB addr16:bp															
SETB io:bp	3	7	0	2× (b)	bit (dir:bp) b ← 0	–	–	–	–	–	–	–	–	–	*
	4	7	0	2× (b)	bit (addr16:bp) b ← 0	–	–	–	–	–	–	–	–	–	*
CLRB dir:bp	3	7	0	2× (b)	bit (io:bp) b ← 0	–	–	–	–	–	–	–	–	–	*
CLRB addr16:bp															
CLRB io:bp	4	*1	0	(b)	Branch when (dir:bp) b = 0	–	–	–	–	–	–	*	–	–	–
	5	*1	0	(b)	Branch when (addr16:bp) b = 0	–	–	–	–	–	–	*	–	–	–
BBC dir:bp, rel	4	*2	0	(b)	Branch when (io:bp) b = 0	–	–	–	–	–	–	*	–	–	–
BBC addr16:bp, rel	4	*1	0	(b)	Branch when (dir:bp) b = 1	–	–	–	–	–	–	*	–	–	–
BBC io:bp, rel	5	*1	0	(b)	Branch when (addr16:bp) b = 1	–	–	–	–	–	–	*	–	–	–
	4	*2	0	(b)	Branch when (io:bp) b = 1	–	–	–	–	–	–	*	–	–	–
BBS dir:bp, rel															
BBS addr16:bp, rel	5	*3	0	2× (b)	Branch when (addr16:bp) b = 1, bit = 1	–	–	–	–	–	–	*	–	–	*
BBS io:bp, rel	3	*4	0	*5	Wait until (io:bp) b = 1	–	–	–	–	–	–	–	–	–	–
SBBS addr16:bp, rel	3	*4	0	*5	Wait until (io:bp) b = 0	–	–	–	–	–	–	–	–	–	–
WBTS io:bp															
WBTC io:bp															

*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 5, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

Table 22 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/XCHW AL, AH	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	*	–	–	–

Table 23 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	–	–	–	–	–	–	–	–	–	–
MOVSD	2	*2	*5	*3	= RW0	–	–	–	–	–	–	–	–	–	–
					Byte transfer @AH– ← @AL–, counter										
SCEQ/SCEQI	2	*1	*5	*4	= RW0	–	–	–	–	–	*	*	*	*	–
SCEQD	2	*1	*5	*4		–	–	–	–	–	*	*	*	*	–
					Byte retrieval (@AH+) – AL, counter =										
FISL/FILSI	2	6m +6	*5	*3	RW0	–	–	–	–	–	*	*	–	–	–
					Byte retrieval (@AH–) – AL, counter = RW0										
					Byte filling @AH+ ← AL, counter = RW0										
MOVSW/	2	*2	*8	*6	Word transfer @AH+ ← @AL+, counter	–	–	–	–	–	–	–	–	–	–
MOVSWI	2	*2	*8	*6	= RW0	–	–	–	–	–	–	–	–	–	–
MOVSWD					Word transfer @AH– ← @AL–, counter										
	2	*1	*8	*7	= RW0	–	–	–	–	–	*	*	*	*	–
SCWEQ/	2	*1	*8	*7		–	–	–	–	–	*	*	*	*	–
SCWEQI					Word retrieval (@AH+) – AL, counter =										
SCWEQD	2	6m +6	*8	*6	RW0	–	–	–	–	–	*	*	–	–	–
					Word retrieval (@AH–) – AL, counter =										
FILSW/FILSWI					RW0										
					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 5, "Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles."

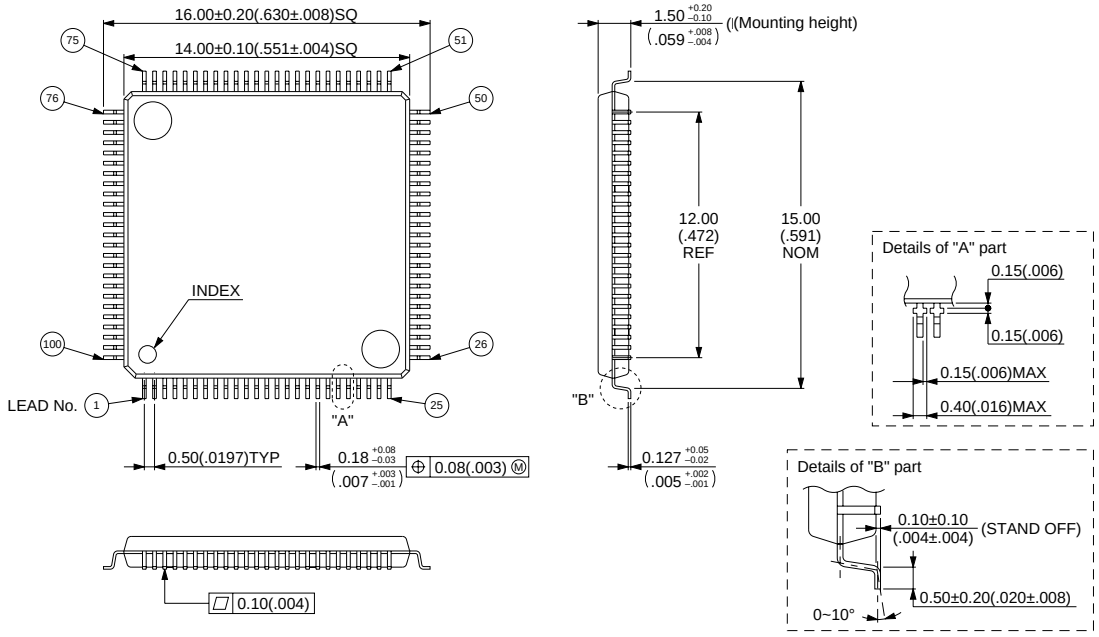
■ ORDERING INFORMATION

Model	Package	Remarks
MB90632APFV MB90634APFV MB90P634APFV	100-pin Plastic LQFP (FPT-100P-M05)	
MB90632APF MB90634APF MB90P634APF	100-pin Plastic QFP (FPT-100P-M06)	MB90P634A supports ES alone.

MB90630A Series

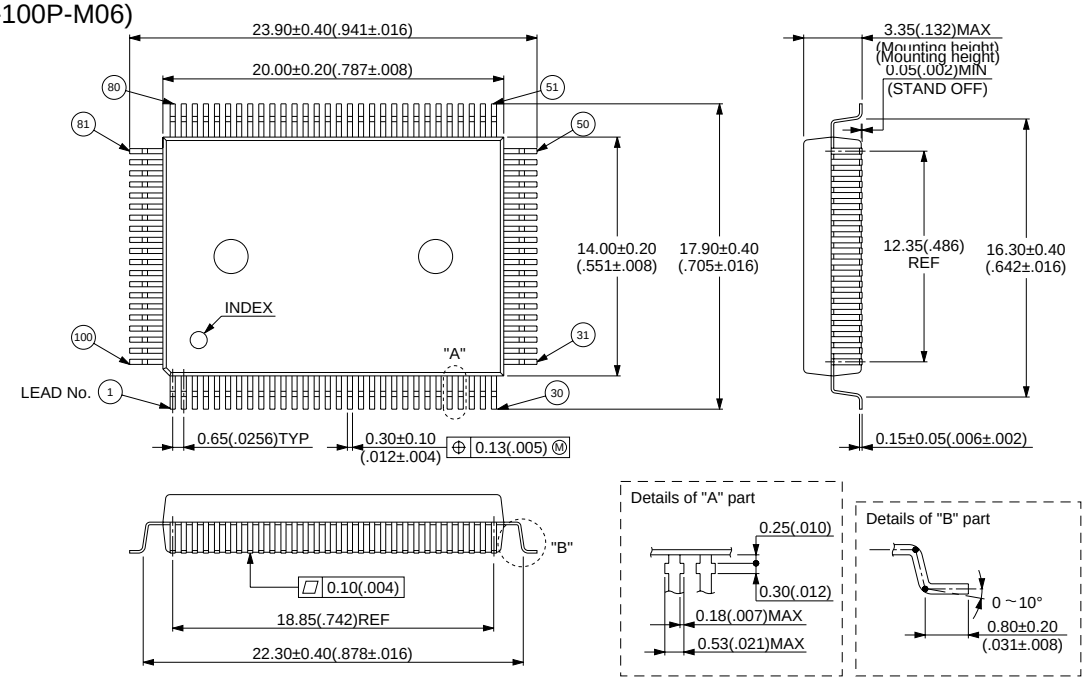
■ PACKAGE DIMENSIONS

100-pin Plastic LQFP
(FPT-100P-M05)



Dimensions in mm (inches)

100-pin Plastic QFP
(FPT-100P-M06)



Dimensions in mm (inches)

FUJITSU LIMITED

For further information please contact:

Japan
FUJITSU LIMITED
Corporate Global Business Support Division
Electronic Devices
KAWASAKI PLANT, 4-1-1, Kamikodanaka
Nakahara-ku, Kawasaki-shi
Kanagawa 211-8588, Japan
Tel: (044) 754-3763
Fax: (044) 754-3329

North and South America
FUJITSU MICROELECTRONICS, INC.
Semiconductor Division
3545 North First Street
San Jose, CA 95134-1804, U.S.A.
Tel: (408) 922-9000
Fax: (408) 432-9044/9045

Europe
FUJITSU MIKROELEKTRONIK GmbH
Am Siebenstein 6-10
63303 Dreieich-Buchschlag
Germany
Tel: (06103) 690-0
Fax: (06103) 690-122

Asia Pacific
FUJITSU MICROELECTRONICS ASIA PTE. LIMITED
#05-08, 151 Lorong Chuan
New Tech Park
Singapore 556741
Tel: (65) 281-0770
Fax: (65) 281-0220

F9803

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 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.

FUJITSU semiconductor devices are intended for use in standard applications (computers, office automation and other office equipment, industrial, communications, and measurement equipment, personal or household devices, etc.).
CAUTION:
Customers considering the use of our products in special applications where failure or abnormal operation may directly affect human lives or cause physical injury or property damage, or where extremely high levels of reliability are demanded (such as aerospace systems, atomic energy controls, sea floor repeaters, vehicle operating controls, medical devices for life support, etc.) are requested to consult with FUJITSU sales representatives before such use. The company will not be responsible for damages arising from such use without prior approval.

Any semiconductor devices have inherently a certain rate 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 Control Law of Japan, the prior authorization by Japanese government should be required for export of those products from Japan.