FUJITSU SEMICONDUCTOR
DATA SHEET

DS07-13706-1E

16-bit Proprietary Microcontroller

CMOS

F²MC-16LX MB90550A Series

MB90552A/553A/T552A/T553A/F553A/P553A

■ DESCRIPTION

The MB90550A series is a line of general-purpose, high-performance, 16-bit microcontrollers designed for applications which require high-speed real-time processing, such as industrial machines, OA equipment, and process control systems.

While inheriting the AT architecture of the F²MC*-8 family, the instruction set for the MB90550A series incorporates additional instructions for high-level languages, supports extended addressing modes, and contains enhanced multiplication and division instructions as well as a substantial collection of improved bit manipulation instructions. In addition, the MB90550A has an on-chip 32-bit accumulator which enables processing of long-word data.

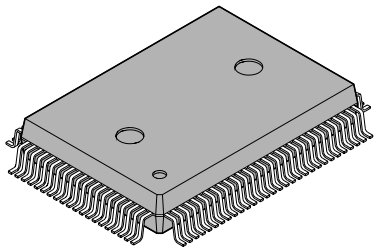
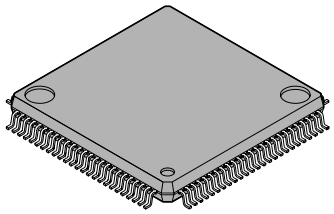
*: F²MC stands for FUJITSU Flexible Microcontroller.

■ FEATURES

- Minimum instruction execution time : 62.5 ns (at oscillation of 4 MHz,×four times the PLL clock)
- Maximum memory space 16 Mbytes
- Instruction set optimized for controller applications
  Supported data types : Bit, byte, word, and long word
  Typical addressing mode : 23 types
  Enhanced precision calculation realized by the 32-bit accumulator
  Enhanced signed multiplication/division instruction and RETI instruction functions

(Continued)

■ PACKAGES

100-pin plastic QFP	100-pin plastic LQFP
	
(FPT-100P-M06)	(FPT-100P-M05)

MB90550A Series

(Continued)

- Instruction set designed for high level language (C) and multi-task operations
  Adoption of system stack pointer
  Symmetrical instruction set and barrel shift instructions
- Address match detection function integrated (for two address pointers)
- Faster execution speed : 4-byte queue
- Powerful interrupt functions (Eight priority levels programmable)
  External interrupt inputs : 8 channels
- Data transfer functions (Intelligent I/O service) : Up to 16 channels
  DTP request inputs : 8 channels
- Embedded ROM size (EPROM, Flash : 128 Kbytes)
  Mask ROM : 64 Kbytes/128 Kbytes
- Embedded RAM size (EPROM, Flash : 4 Kbytes)
  Mask ROM : 2 Kbytes/4 Kbytes
- General-purpose ports :Up to 83 channels
  (Input pull-up resistor settable for : 16 channels
  Open drain settable for : 8 channels
  I/O open drains : 6 channels)
- A/D converter (RC successive approximation type): 8 channels
  (Resolution: 8 or 10 bits selectable; Conversion time of 26.3 μs minimum)
- UART : 1 channel
- Extended I/O serial interface : 2 channels
- I^2C interface : 2 channels
  (Two channels, including one switchable between terminal input and output)
- 16-bit reload timer : 2 channels
- 8/16-bit PPG timer : 3 channels
  (8 bits × 2 channels; 16 bits x 1 channel: Mode switching function provided)
- 16-bit I/O timer
  (Input capture × 4 channels, output compare × 4 channels, free run timer ×1 channel
- Clock monitor function integrated (Delivering the oscillation clock divided by 21 to 28)
- Timebase timer/watchdog timer : 18 bit
- Low power consumption modes (sleep, stop, hardware standby, and CPU intermittent operation modes)
- Package : QFP-100, LQFP-100
- CMOS technology

■ PRODUCT LINEUP

Item \ Part number		MB90552A	MB90553A	MB90F553A	MB90P553A	MB90V550A
Classification		Mask ROM products		Flash ROM products	OTP	Evaluation product
		Mass Product				
ROM size		64 Kbytes	128 Kbytes			None
RAM size		2 Kbytes	4 Kbytes			6 Kbytes
CPU functions		The number of instructions: 340 Instruction bit length: 8 bits, 16 bits Instruction length: 1 byte to 7 bytes Data bit length: 1 bit, 8 bits, 16 bits Minimum execution time: 62.5 ns (at machine clock of 16 MHz) Interrupt processing time: 1.5 ms (at machine clock of 16 MHz, minimum value)				
Ports		General-purpose I/O ports (CMOS output): 53 General-purpose I/O ports (with pull-up resistor): 16 General-purpose I/O ports (N-channel open-drain output): 6 General-purpose I/O ports (N-channel open-drain function selectable): 8 Total: 83				
UART0 (SCI)		Clock synchronized transmission (62.5 kbps to 2 Mbps) Clock asynchronized transmission (62500 bps to 9615 bps) Transmission can be performed by bi-directional serial transmission or by master/slave connection.				
8/10-bit A/D converter		Resolution: 8/10-bit Number of inputs: 8 One-shot conversion mode (converts selected channel only once) Scan conversion mode (converts two or more successive channels and can program up to 8 channels.) Continuous conversion mode (converts selected channel continuously) Stop conversion mode (converts selected channel and stop operation repeatedly)				
8/16-bit PPG timer		Number of channels: 1 (or 8-bit × 2 channels) PPG operation of 8-bit or 16-bit A pulse wave of given intervals and given duty ratios can be output. Pulse interval: 62.5 ns to 1 ms (at oscillation of 4 MHz, machine clock of 16 MHz)				
16-bit I/O timer	16-bit free run timer	Number of channel: 1 Overflow interrupts				
	Output compare (OCU)	Number of channels: 4 Pin input factor: A match signal of compare register				
	Input capture (ICU)	Number of channels: 4 Rewriting a register value upon a pin input (rising, falling, or both edges)				

(Continued)

MB90550A Series

(Continued)

Item \ Part number	MB90552A	MB90553A	MB90F553A	MB90P553A	MB90V550A
DTP/external interrupt circuit	Number of inputs: 8 Started by a rising edge, a falling edge, an “H” level input, or an “L” level input. External interrupt circuit or extended intelligent I/O service (EI^{2}OS) can be used.				
Extended I/O serial interface	Clock synchronized transmission (3125 bps to 1 Mbps) LSB first/MSB first				
I^{2}C interface	Serial I/O port for supporting Inter IC BUS				
Timebase timer	18-bit counter Interrupt interval: 1.024 ms, 4.096 ms, 16.384 ms, 131.072 ms (at oscillation of 4 MHz)				
Watchdog timer	Reset generation interval: 3.58 ms, 14.33 ms, 57.23 ms, 458.75 ms (at oscillation of 4 MHz, minimum value)				
Process	CMOS				
Power supply voltage for operation*	4.5 V to 5.5 V				

*:Varies with conditions such as the operating frequency. (See section “■ ELECTRICAL CHARACTERISTICS”)
Assurance for the MB90V550A is given only for operation with a tool at a power voltage of 4.5 V to 5.5 V, an operating temperature of 0 to +25°C, and an operating frequency of 1 MHz to 16 MHz.

■ PACKAGE AND CORRESPONDING PRODUCTS

Package	MB90552A	MB90553A	MB90F553A	MB90P553A
FPT-100P-M05	○	○	○	×
FPT-100P-M06	○	○	○	○

○ : Available × : Not available

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

■ DIFFERENCES AMONG PRODUCTS

Memory Size

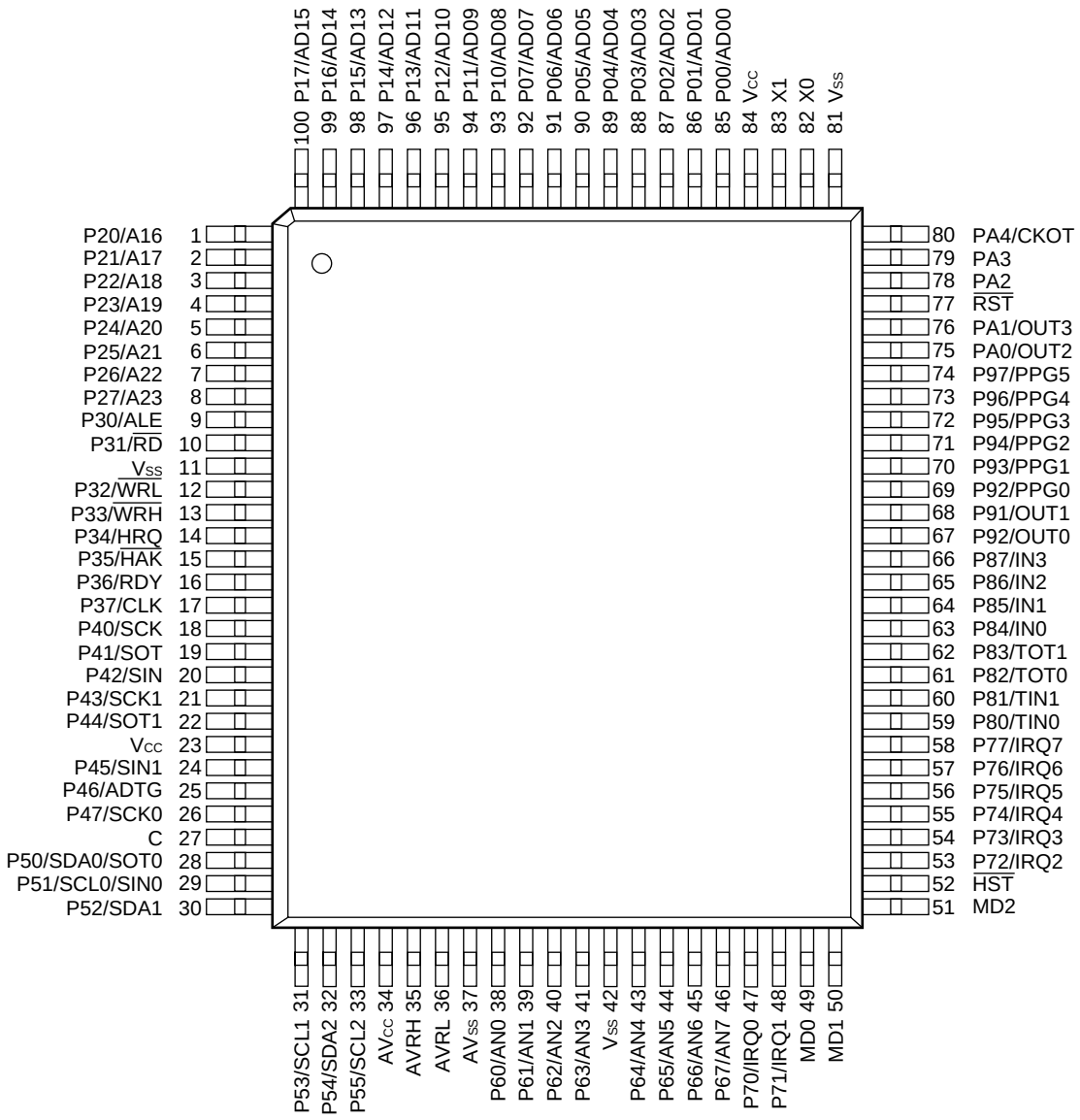
In evaluation with an evaluation product, note the difference between the evaluation product and the product actually used. The following items must be taken into consideration.

- The MB90V550A does not have an internal ROM, however, operations equivalent to chips with an internal ROM can be evaluated by using a dedicated development tool, enabling selection of ROM size by settings of the development tool.
- In the MB90V550, images from FF4000$_H$ to FFFFFF$_H$ are mapped to bank 00, and FE0000$_H$ to FF3FFF$_H$ to mapped to bank FE and FF only. (This setting can be changed by configuring the deveolpment tool.)
- In the MB90F553A/553A/552A, images from FF4000$_H$ to FFFFFF$_H$ are mapped to bank 00, and FF0000$_H$ to FF3FFF$_H$ to bank FF only.

■ PIN ASSIGNMENT

- FPT-100P-M06

(Top view)

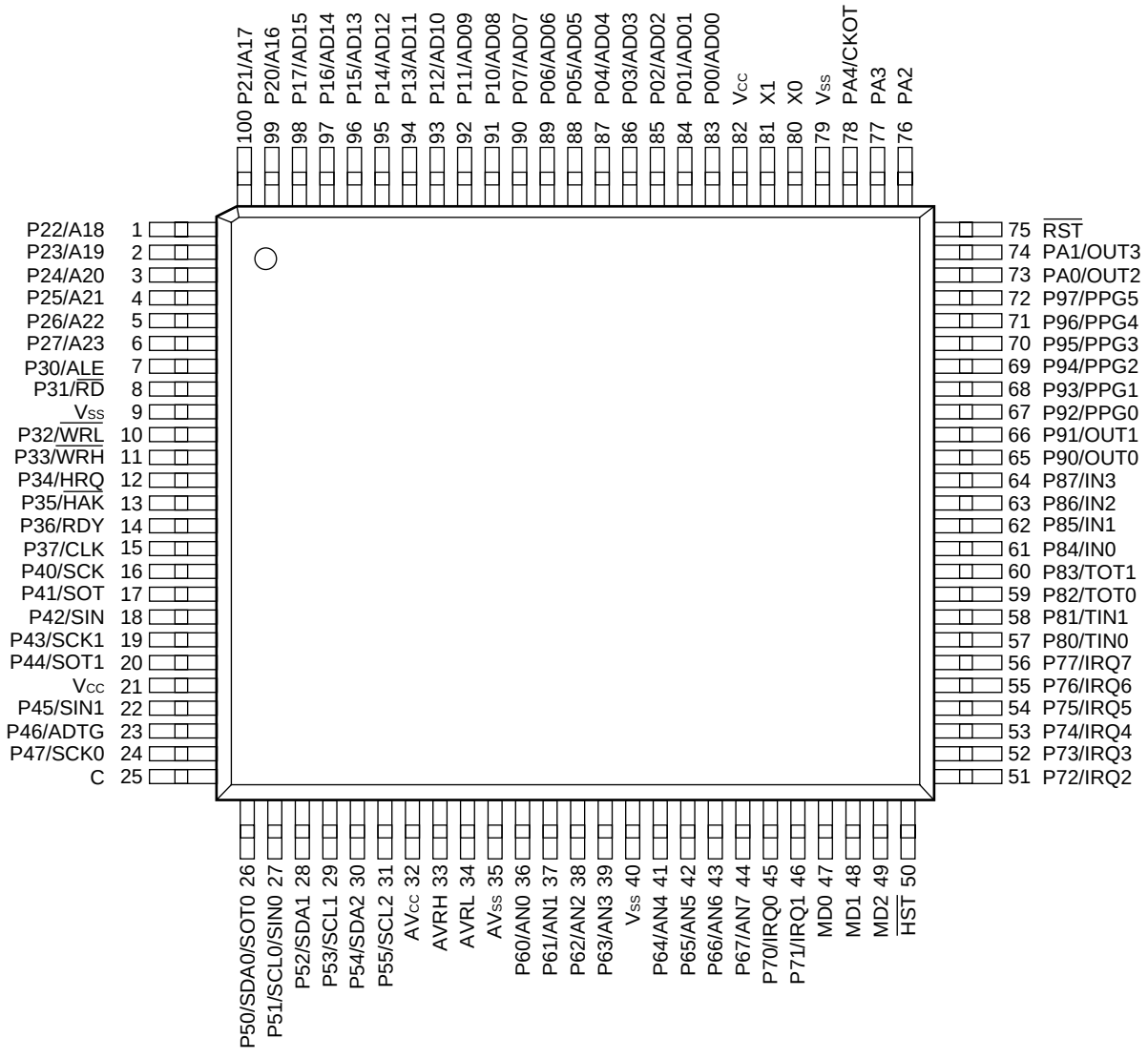


(FPT-100P-M06)

MB90550A Series

- FPT-100P-M05

(Top view)



(FPT-100P-M05)

■ PIN DESCRIPTION

Pin no.		Pin name	Circuit type	Function
QFP	LQFP			
82	80	X0	A	Oscillation pin
83	81	X1	A	Oscillation pin
77	75	$\overline{\text{RST}}$	B	Reset input pin
52	50	$\overline{\text{HST}}$	C	Hardware standby input pin
85 to 92	83 to 90	P00 to P07	D (CMOS)	General-purpose I/O port. A pull-up resistor can be added (RD07 to RD00 = 1) by using the pull-up resistor setting register (RDR0). D07 to D00 = 1: Disabled when the port is set for output.
		AD00 to AD07		Serve as lower data I/O/lower address output (AD00 to AD07) pins in the external bus mode.
93 to 100	91 to 98	P10 to P17	D (CMOS)	General-purpose I/O port. A pull-up resistor can be added (RD17 to RD10 = 1) by using the pull-up resistor setting register (RDR1). D17 to D10 = 1: Disabled when the port is set for output.
		AD08 to AD15		Serve as upper data I/O/middle address output (AD08 to AD15) pins in the 16-bit bus-width, external bus mode.
1 to 8	99,100, 1 to 6	P20 to P27	E (CMOS)	General-purpose I/O port. This function is enabled either in single-chip mode or with the xternal address output control register set to "Port".
		A16 to A23		External address bus A16 to A23 output pins. This function is enabled in an external-bus enabled mode with the external address output register set to "Address".
9	7	P30	E (CMOS)	General-purpose I/O port. This function is enabled in single-chip mode.
		ALE		Address latch enable output pin. This function is enabled in an external-bus enabled mode.
10	8	P31	E (CMOS)	General-purpose I/O port. This function is enabled in single-chip mode.
		$\overline{\text{RD}}$		Read strobe output pin for the data bus. This function is enabled in an external-bus enabled mode.
12	10	P32	E (CMOS)	General-purpose I/O port. This function is enabled in single-chip mode.
		$\overline{\text{WRL}}$		Write strobe output pin for the lower eight bits of the data bus. This function is enabled in an external-bus enabled mode.
13	11	P33	E (CMOS)	General-purpose I/O port. This function is enabled in single-chip mode.
		$\overline{\text{WRH}}$		Write strobe output pin for the upper eight bits of the data bus. This function is enabled in an external-bus enabled mode.

(Continued)

Pin no.		Pin name	Circuit type	Function
QFP	LQFP			
14	12	P34	E (CMOS)	General-purpose I/O port. This function is enabled in single-chip mode
		HRQ		Hold request input pin. This function is enabled in an external-bus enabled mode.
15	13	P35	E (CMOS)	General-purpose I/O port. This function is enabled in single-chip mode.
		$\overline{\text{HAK}}$		Hold acknowledge output pin. This function is enabled in an external-bus enabled mode.
16	14	P36	E (CMOS)	General-purpose I/O port. This function is enabled in single-chip mode.
		RDY		Ready signal input pin. This function is enabled in an external-bus enabled mode.
17	15	P37	E (CMOS)	General-purpose I/O port. This function is enabled in single-chip mode.
		CLK		CLK output pin. This function is enabled in an external-bus enabled mode.
18	16	P40	F (CMOS/H)	General-purpose I/O port. Serves as an open-drain output port (OD40 = 1) depending on the setting of the open-drain control setting register (ODR4). (D40 = 0: Disabled when the port is set for input.)
		SCK		UART serial clock I/O pin. This function is enabled with the UART clock output enabled.
19	17	P41	F (CMOS/H)	General-purpose I/O port. Serves as an open-drain output port (OD41 = 1) depending on the setting of the open-drain control setting register (ODR4). (D41 = 0: Disabled when the port is set for input.)
		SOT		UART serial data output pin. This function is enabled with the UART serial data output enabled.
20	18	P42	F (CMOS/H)	General-purpose I/O port. Serves as an open-drain output port (OD42 = 1) depending on the setting of the open-drain control setting register (ODR4). (D42 = 0: Disabled when the port is set for input.)
		SIN		UART serial data input pin. Since this input is used as required while the UART is operating for input, the output by any other function must be off unless used intentionally.
21	19	P43	F (CMOS/H)	General-purpose I/O port. Serves as an open-drain output port (OD43 = 1) depending on the setting of the open-drain control setting register (ODR4). (D43 = 0: Disabled when the port is set for input.)
		SCK1		Extended I/O serial clock I/O pin. This function is enabled with the extended I/O serial clock output enabled.

(Continued)

Pin no.		Pin name	Circuit type	Function
QFP	LQFP			
22	20	P44	F (CMOS/H)	General-purpose I/O port. Serves as an open-drain output port (OD44 = 1) depending on the setting of the open-drain control setting register (ODR4). (D44 = 0: Disabled when the port is set for input.)
		SOT1		Extended I/O serial data output pin. This function is enabled with the extended I/O serial data output enabled.
24	22	P45	F (CMOS/H)	General-purpose I/O port. Serves as an open-drain output port (OD45 = 1) depending on the setting of the open-drain control setting register (ODR4). (D45 = 0: Disabled when the port is set for input.)
		SIN1		Extended I/O serial data input pin. Since this input is used as required while the extended I/O serial interface is operating for input, the output by any other function must be off unless used intentionally.
25	23	P46	F (CMOS/H)	General-purpose I/O port. Serves as an open-drain output port (OD46 = 1) depending on the setting of the open-drain control setting register (ODR4). (D46 = 0: Disabled when the port is set for input.)
		ADTG		A/D converter external trigger input pin. Since this input is used as required while the A/D converter is operating for input, the output by any other function must be off unless used intentionally.
26	24	P47	F (CMOS/H)	General-purpose I/O port. Serves as an open-drain output port (OD47 = 1) depending on the setting of the open-drain control setting register (ODR4). D47 = 0: Disabled when the port is set for input.
		SCK0		Extended I/O serial clock I/O pin. This function is enabled with the extended I/O serial clock output enabled.
27	25	C	—	Capacitance pin for regulating the power supply. Connect an external ceramic capacitor of about 0.1 μF.
28	26	P50	G (NchOD/H)	N-channel open-drain I/O port.
		SDA0		I^2C interface data I/O pin. This function is enabled with the I^2C interface enabled for operation. While the I^2C interface is operating, place the port output in the Hi-Z state (PDR = 1).
		SOT0		Extended I/O serial data output pin. This function is enabled with the extended I/O serial data output enabled.

(Continued)

MB90550A Series

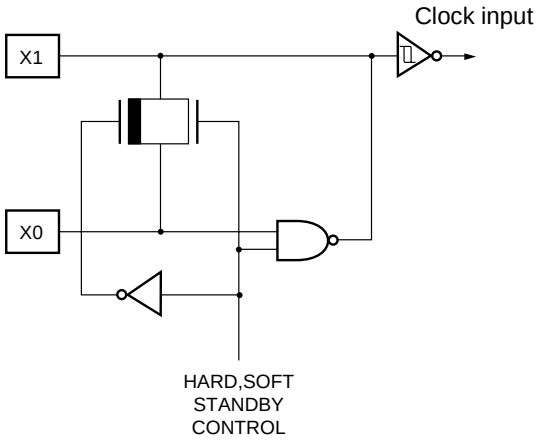
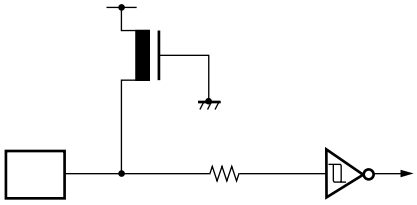
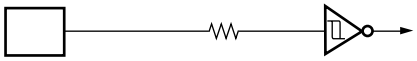
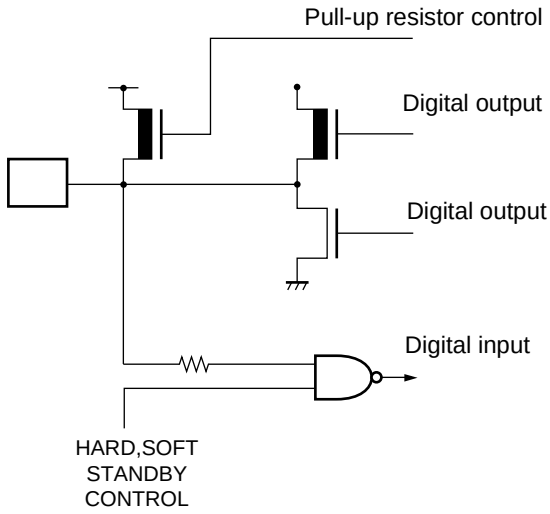
Pin no.		Pin name	Circuit type	Function
QFP	LQFP			
29	27	P51	G (NchOD/H)	N-channel open-drain I/O port.
		SCL0		I^2C interface clock I/O pin. This function is enabled with the I^2C interface enabled for operation. While the I^2C interface is operating, place the port output in the Hi-Z state (PDR = 1).
		SIN0		Extended I/O serial data input pin. Since this input is used as required while the extended I/O serial interface is operating for input, the output by any other function must be off unless used intentionally.
30,32	28,30	P52,P54	G (NchOD/H)	N-channel open-drain I/O port.
		SDA1,SDA2		I^2C interface data I/O pins. This function is enabled with the I^2C interface enabled for operation. While the I^2C interface is operating, place the port output in the Hi-Z state (PDR = 1).
31,33	29,31	P53,P55	G (NchOD/H)	N-channel open-drain I/O port.
		SCL1,SCL2		I^2C interface clock I/O pins. This function is enabled with the I^2C interface enabled for operation. While the I^2C interface is operating, place the port output in the Hi-Z state (PDR = 1).
38 to 41, 43 to 46	36 to 39, 41 to 44	P60 to P67	H (CMOS/H)	General-purpose I/O port.
		AN0 to AN7		A/D converter analog input pin. This function is enabled with the analog input enabled.
47,48, 53 to 58	45,46, 51 to 56	P70 to P77	I (CMOS/H)	General-purpose I/O port.
		IRQ0 to IRQ7		External interrupt request input pins. Since this input is used as required while external interrupts remain enabled, the output by any other function must be off unless used intentionally.
59,60	57,58	P80,P81	J (CMOS/H)	General-purpose I/O port.
		TIN0,TIN1		Reload timer event input pins. Since this input is used as required while the reload timer is operating for input, the output by any other function must be off unless used intentionally.
61,62	59,60	P82,P83	J (CMOS/H)	General-purpose I/O port.
		TOT0,TOT1		Reload timer output pins.
63 to 66	61 to 64	P84 to P87	J (CMOS/H)	General-purpose I/O port.
		IN0 to IN3		Input capture trigger input pin. Since this input is used as required while the input capture unit is operating for input, the output by any other function must be off unless used intentionally.
67,68	65,66	P90,P91	J (CMOS/H)	General-purpose I/O port.
		OUT0,OUT1		Output compare event output pins.

(Continued)

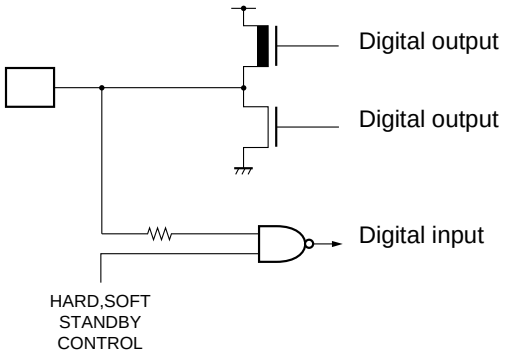
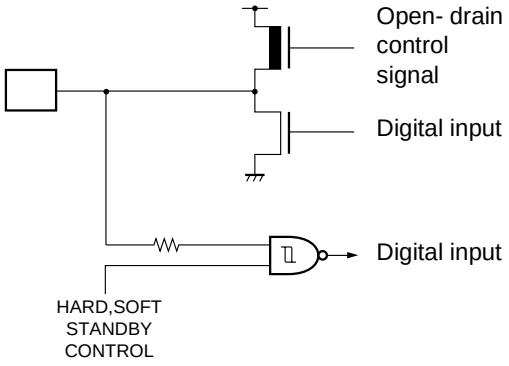
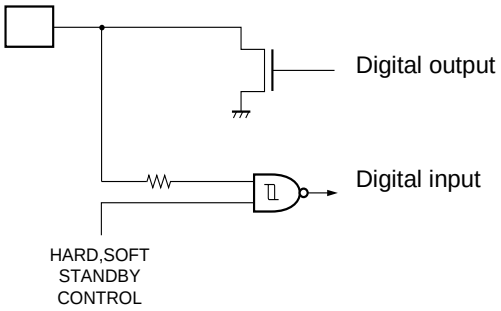
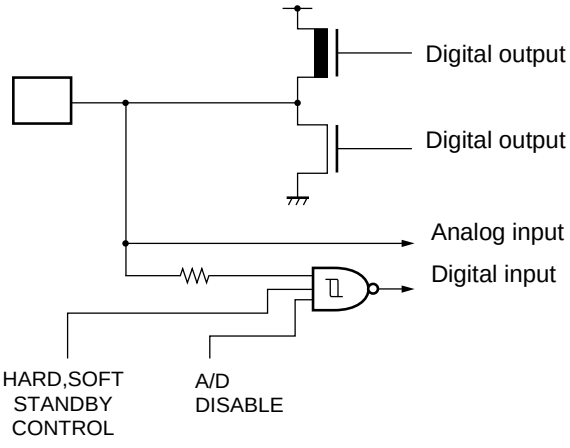
(Continued)

Pin no.		Pin name	Circuit type	Function
QFP	LQFP			
69 to 74	67 to 72	P92 to P97	J (CMOS/H)	General-purpose I/O port.
		PPG0 to PPG5		PPG output pins. This function is enabled with the PPG output enabled.
75,76	73,74	PA0,PA1	J (CMOS/H)	General-purpose I/O port.
		OUT2,OUT3		Output compare event output pins.
78,79	76,77	PA2,PA3	J (CMOS/H)	General-purpose I/O port.
80	78	PA4	J (CMOS/H)	General-purpose I/O port.
		CKOT		Serves as the CKOT output while the CKOT is operating.
34	32	AV_{CC}	—	A/D converter power-supply pin.
35	33	AVRH	—	This is a general purpose I/O port.
36	34	AVRL	—	A/D converter external reference voltage source pin.
37	35	AV_{SS}	—	A/D converter power-supply pin.
49 to 50	47 to 48	MD0,MD1	C	Operation mode setting input pins. Connect these pins directly to Vcc or Vss.
51	49	MD2	K	Operation mode setting input pin. Connect this pin directly to Vcc or Vss. (MB90552A/553A/V550A)
			C	Operation mode setting input pin. Connect this pin directly to Vcc or Vss. (MB90P553A/F553A)
23,84	21,82	V_{CC}	—	Power (5 V) input pin.
11,42, 81	9,40, 79	V_{SS}	—	Power (0 V) input pin.

■ I/O CIRCUIT TYPE

Type	Circuit	Remarks
A		• 3 MHz to 32 MHz • Oscillator recovery resistor approx. 1MΩ
B		• CMOS level hysteresis input • Pull-up resistor provided Resistor : About 50 kΩ
C		• CMOS level hysteresis input
D		• CMOS level output • CMOS level input • Standby control provided • Input pull-up resistor control provided Resistor: About 50 kΩ

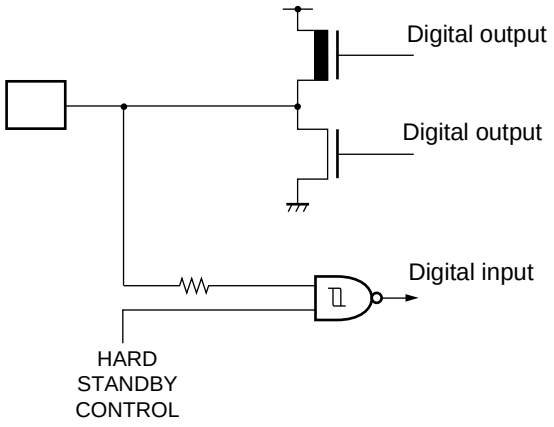
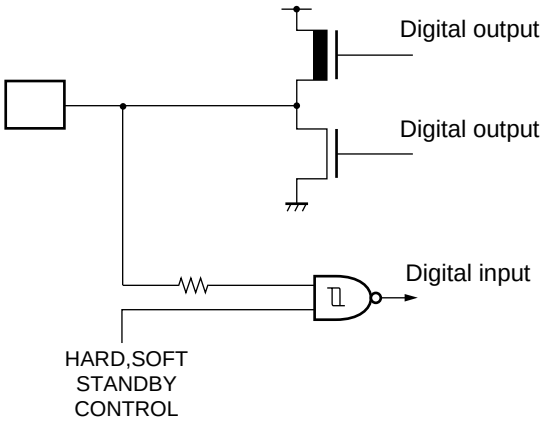
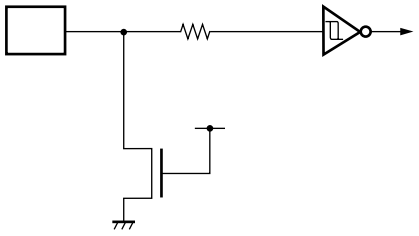
(Continued)

Type	Circuit	Remarks
E		• CMOS level output • CMOS level input • Standby control provided
F		• CMOS level output • CMOS level hysteresis input • Open-drain control provided
G		• N-channel open-drain output • CMOS level hysteresis input • Standby control provided Note: Unlike normal CMOS I/O pins, this pin is not provided with any P-channel transistor. Therefore the pin does not allow a current to flow to the Vcc side even when applied with a voltage from an external device with the IC's power supply left off.
H		• CMOS level output • CMOS level hysteresis input • Standby control provided • Analog input

(Continued)

MB90550A Series

(Continued)

Type	Circuit	Remarks
I		• CMOS level output • CMOS level hysteresis input • Standby control provided
J		• CMOS level output • CMOS level hysteresis input • Standby control provided
K		• CMOS level hysteresis input • Pull-up resistor provided Resistor : About 50kΩ

■ HANDLING DEVICES

1. Preventing Latchup

CMOS ICs may cause latchup in the following situations:

- When a voltage higher than Vcc or lower than Vss is applied to input or output pins.
- When a voltage exceeding the rating is applied between Vcc and Vss.
- When AVcc power is supplied prior to the Vcc voltage.

If latchup occurs, the power supply current increases rapidly, sometimes resulting in thermal breakdown of the device. Use meticulous care not to let it occur.

For the same reason, also be careful not to let the analog power-supply voltage exceed the digital power-supply voltage.

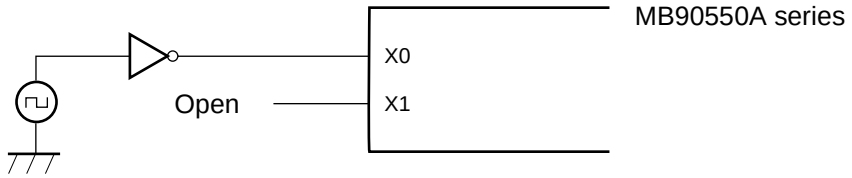
2. Connection of Unused Pins

Leaving unused pins open may result in abnormal operations. Clamp the pin level by connecting it to a pull-up or a pull-down 1kΩ or more resistor.

3. Notes on Using External Clock

In using the external clock, drive X0 pin only and leave X1 pin unconnected.

- **Using external clock**



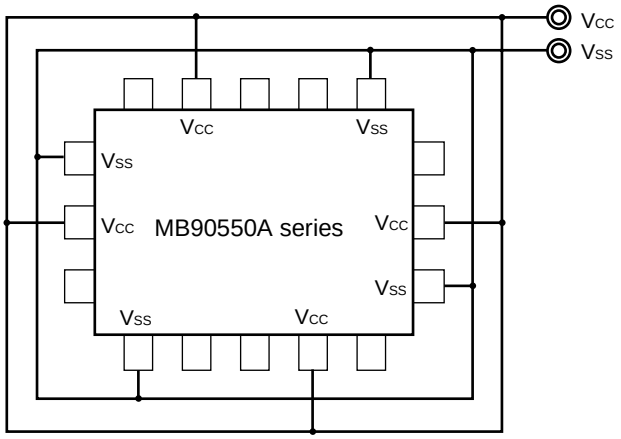
4. Power Supply Pins (V_{CC}/V_{SS})

In products with multiple V_{CC} or V_{SS} pins, the pins of a same potential are internally connected in the device to avoid abnormal operations including latch-up. However, connect the pins external power and ground lines to lower the electro-magnetic emission level and abnormal operation of strobe signals caused by the rise in the ground level, and to conform to the total current rating.

Make sure to connect V_{CC} and V_{SS} pins via lowest impedance to power lines.

It is recommended to provide a bypass capacitor of around 0.1 µF between V_{CC} and V_{SS} pin near the device.

- **Using power supply pins**



5. Crystal Oscillator Circuit

Noises around X0 or X1 pins may be possible causes of abnormal operations. Make sure to provide bypass capacitors via shortest distance from X0, X1 pins, crystal oscillator (or ceramic resonator) and ground lines, and make sure, to the utmost effort, that lines of oscillation circuit not cross the lines of other circuits.

It is highly recommended to provide a printed circuit board art work surrounding X0 and X1 pins with an grand area for stabilizing the operation.

6. Turning-on Sequence of Power Supply to A/D Converter and Analog Inputs

Make sure to turn on the A/D converter power supply, D/A converter power supply (AV_{CC}, AVRH, AVRL) and analog inputs (AN0 to AN7) after turning-on the digital power supply (V_{CC}).

Turn-off the digital power after turning off the A/D converter supply and analog inputs. In this case, make sure that the voltage not exceed AVRH or AV_{CC} (turning on/off the analog and digital power supplies simultaneously is acceptable).

7. Connection of Unused Pins of A/D Converter

Connect unused pin of A/D converter to $AV_{CC} = V_{CC}$, $AV_{SS} = AVRH = AVRL = V_{SS}$.

8. N.C. Pin

The N.C. (internally connected) pin must be opened for use.

9. Notes on Energization

To prevent the internal regulator circuit rom malfunctioning,set the voltage rise time during energization at 50 or more μs.

10. Initialization

In the device, there are internal registers which is initialized only by a power-on reset. To initialize these registers turning on the power again.

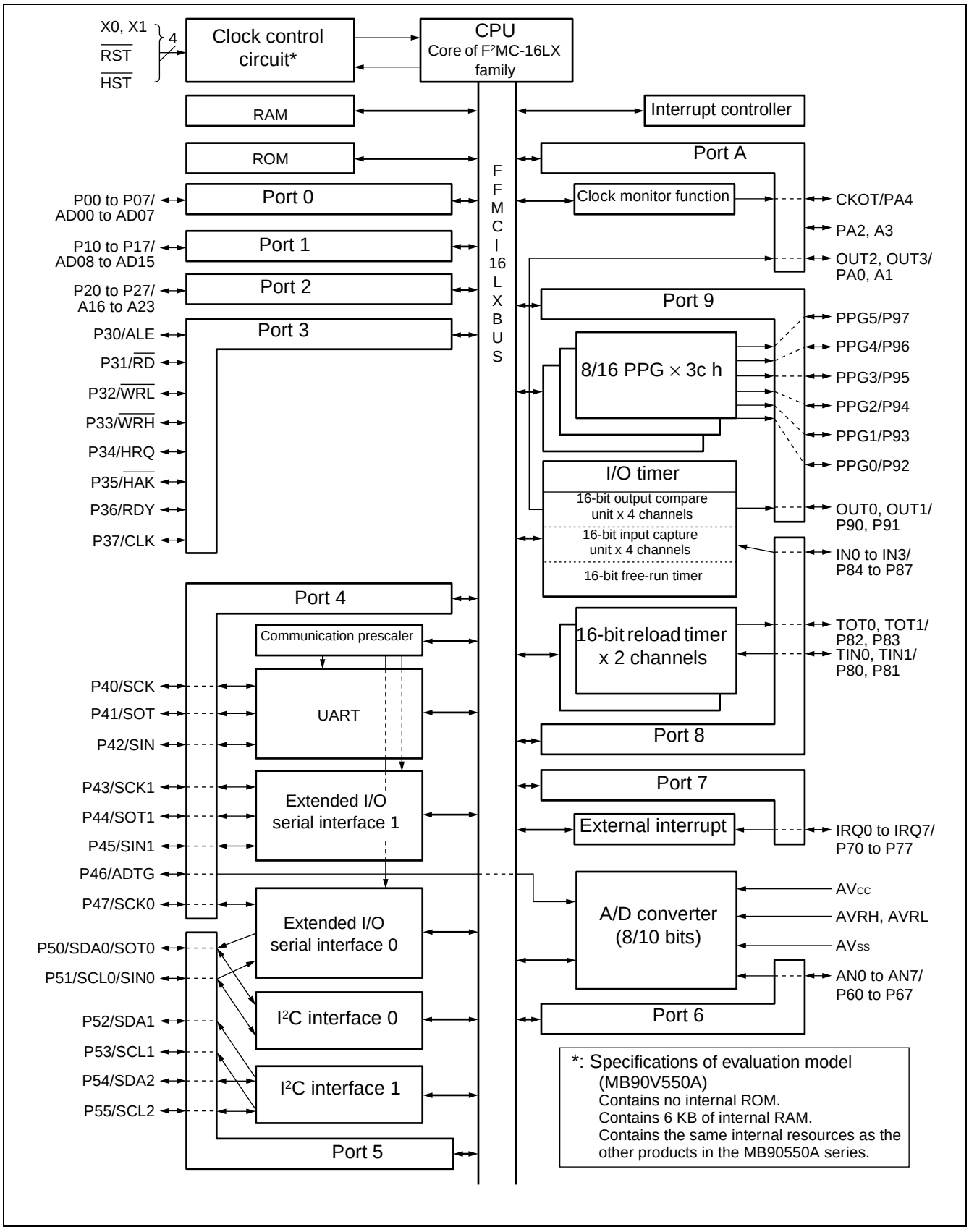
11. Return from standby state

If the power-supply voltage goes below the standby RAM holding voltage in the standby state, the device may fail to return from the standby state. In this case, reset the device via the external reset pin to return to the normal state.

12. Precautions for Use of 'DIV A, Ri,' and 'DIVW A, Ri' Instructions

The signed multiplication-division instructions 'DIV A, Ri,' and 'DIVW A, RWi' should be used when the corresponding bank registers (DTB, ADB, USB, SSB) are set to value '00h.' If the corresponding bank registers (DTB, ADB, USB, SSB) are set to a value other than '00h,' then the remainder obtained after the execution of the instruction will not be placed in the instruction operand register.

■ BLOCK DIAGRAM



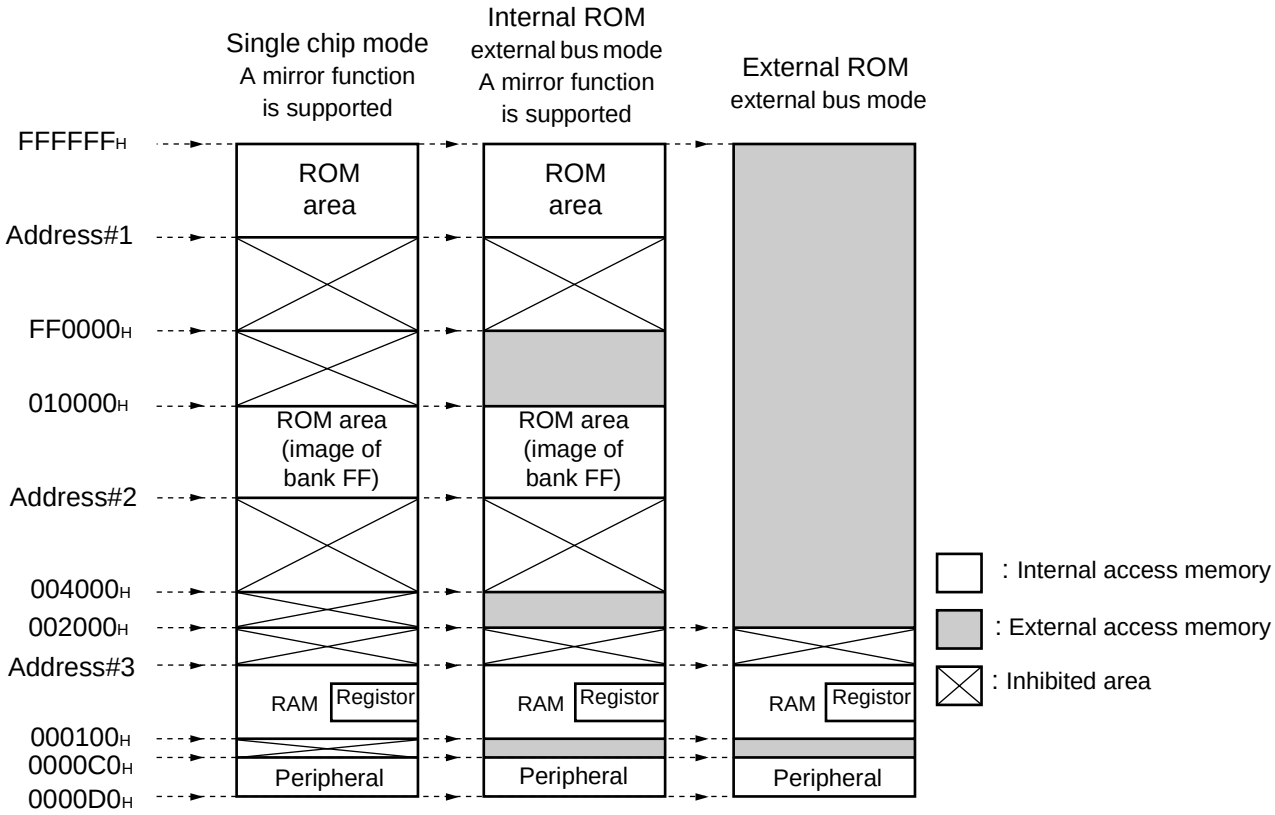
Note: The clock control circuit contains a watchdog timer, time-base timer, and a low power consumption control circuit.

P00 to P07 (8 pins): Input pull-up resistor setting register provided
P10 to P17 (8 pins): Input pull-up resistor setting register provided
P40 to P47 (8 pins): Open-drain control setting register provided
P50 to P55 (6 pins): N-channel open drain
Ports 0, 1, 2, 3, 4, 6, 7, 8, 9, and A are CMOS level input/output ports.

■ MEMORY MAP

The ROM data of bank FF is reflected in the upper address of bank 00, realizing effective use of the C compiler small model. The lower 16-bit of bank FF and the lower 16-bit of bank 00 is assigned to the same address, enabling reference of the table on the ROM without stating “far”.

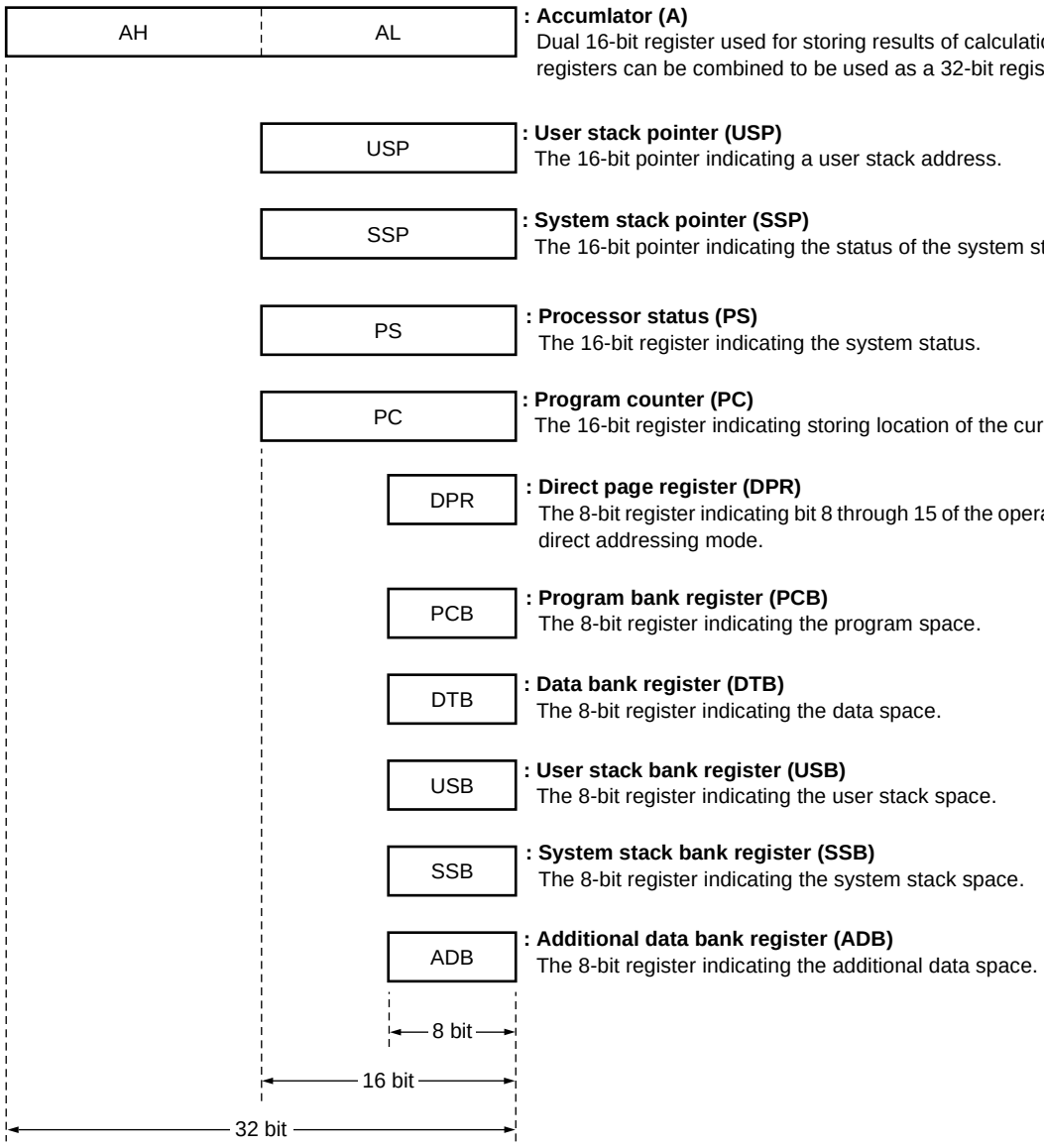
For example, if an attempt has been made to access 00C000H, the contents of the ROM at FFC000H are accessed actually. Since the ROM area of the FF bank exceeds 48 Kbytes, the whole area cannot be reflected in the image for the 00 bank. The ROM data at FF4000H to FFFFFFH looks, therefore, as if it were the image for 004000H to 00FFFFH. Thus, it is recommended that the ROM data table be stored in the area of FF4000H to FFFFFFH.



Parts No.	Address#1	Address#2	Address#3
MB90552A	FF0000H	004000H	000900H
MB90553A	FE0000H	004000H	001100H
MB90F553A	FE0000H	004000H	001100H
MB90P553A	FE0000H	004000H	001100H
MB90V550A	(FE0000H)	004000H	001900H

■ F²MC-16LX CPU PROGRAMMING MODEL

- Dedicated registers



■ I/O MAP

Address	Register name	Abbreviated register name	Read/write	Resource name	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	_ _ 1 1 1 1 1 1
06H	Port 6 data register	PDR6	R/W	Port 6	XXXXXXXX
07H	Port 7 data register	PDR7	R/W	Port 7	XXXXXXXX
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
0BH to 0FH	(Disabled)				
10H	Port 0 direction register	DDR0	R/W	Port 0	0 0 0 0 0 0 0 0
11H	Port 1 direction register	DDR1	R/W	Port 1	0 0 0 0 0 0 0 0
12H	Port 2 direction register	DDR2	R/W	Port 2	0 0 0 0 0 0 0 0
13H	Port 3 direction register	DDR3	R/W	Port 3	0 0 0 0 0 0 0 0
14H	Port 4 direction register	DDR4	R/W	Port 4	0 0 0 0 0 0 0 0
15H	(Disabled)				
16H	Port 6 direction register	DDR6	R/W	Port 6	0 0 0 0 0 0 0 0
17H	Port 7 direction register	DDR7	R/W	Port 7	0 0 0 0 0 0 0 0
18H	Port 8 direction register	DDR8	R/W	Port 8	0 0 0 0 0 0 0 0
19H	Port 9 direction register	DDR9	R/W	Port 9	0 0 0 0 0 0 0 0
1AH	Port A direction register	DDRA	R/W	Port A	_ _ _ 0 0 0 0 0
1BH	Port 4 output pin register	ODR4	R/W	Port 4	0 0 0 0 0 0 0 0
1CH	Port 0 resistor setting register	RDR0	R/W	Port 0	0 0 0 0 0 0 0 0
1DH	Port 1 resistor setting register	RDR1	R/W	Port 1	0 0 0 0 0 0 0 0
1EH	(Disabled)				
1FH	Analog input enable register	ADER	R/W	Port 6, A/D converter	1 1 1 1 1 1 1 1
20H	Serial mode register	SMR	R/W	UART	0 0 0 0 0 0 0 0
21H	Serial control register	SCR	R/W		0 0 0 0 0 1 0 0
22H	Serial input data register / serial output data register	SIDR/SODR	R/W		XXXXXXXX
23H	Serial status register	SSR	R/W		0 0 0 0 1 _ 0 0

(Continued)

Address	Register name	Abbreviated register name	Read/write	Resource name	Initial value
24H	Serial mode control status register 0	SMCS0	R/W	Extended I/O serial interface 0	_ _ _ _ 0 0 0 0
25H	Serial mode control status register 0		R/W!		0 0 0 0 0 0 1 0
26H	Serial data register 0	SDR0	R/W		XXXXXXXX
27H	Clock frequency-divider control register	CDCR	R/W	Communication prescaler	0 _ _ _ 1 1 1 1
28H	Serial mode control status register 1	SMCS1	R/W	Extended I/O serial interface 1	_ _ _ _ 0 0 0 0
29H	Serial mode control status register 1		R/W!		0 0 0 0 0 0 1 0
2AH	Serial data register 1	SDR1	R/W		XXXXXXXX
2BH	(Disabled)				
2CH	I²C bus status register 0	IBSR0	R	I²C interface 0	0 0 0 0 0 0 0 0
2DH	I²C bus control register 0	IBCR0	R/W		0 0 0 0 0 0 0 0
2EH	I²C bus clock select register 0	ICCR0	R/W		_ _ 0XXXXX
2FH	I²C bus address register 0	IADR0	R/W		_ XXXXXXX
30H	I²C bus data register 0	IDAR0	R/W		XXXXXXXX
31H	(Disabled)				
32H	I²C bus status register 1	IBSR1	R	I²C interface 1	0 0 0 0 0 0 0 0
33H	I²C bus control register 1	IBCR1	R/W		0 0 0 0 0 0 0 0
34H	I²C bus clock select register 1	ICCR1	R/W		_ _ 0XXXXX
35H	I²C bus address register 1	IADR1	R/W		_ XXXXXXX
36H	I²C bus data register 1	IDAR1	R/W		XXXXXXXX
37H	I²C bus port select register	ISEL	R/W		_ _ _ _ _ _ _ 0
38H	Interrupt/DTP enable register	ENIR	R/W	DTP/externalint interrupt	0 0 0 0 0 0 0 0
39H	Interrupt/DTP factor register	EIRR	R/W		XXXXXXXX
3AH	Request level setting register	ELVR	R/W		0 0 0 0 0 0 0 0
3BH					0 0 0 0 0 0 0 0
3CH	Control status register	ADCS0	R/W	A/D convertor	0 0 0 0 0 0 0 0
3DH		ADCS1	R/W		0 0 0 0 0 0 0 0
3EH	Data register	ADCR0	R/W!		XXXXXXXX
3FH		ADCR1	R/W		XXXXXXXX

(Continued)

Address	Register name	Abbreviated register name	Read/write	Resource name	Initial value
40H	Reload register L (ch.0)	PRLL0	R/W	8/16 bit PPG0/1	XXXXXXXX
41H	Reload register H (ch.0)	PRLH0	R/W		XXXXXXXX
42H	Reload register L (ch.1)	PRLL1	R/W		XXXXXXXX
43H	Reload register H (ch.1)	PRLH1	R/W		XXXXXXXX
44H	PPG0 operating mode control register	PPGC0	R/W		0_000__1
45H	PPG1 operating mode control register	PPGC1	R/W		0_000001
46H	PPG0 and 1 output control register	PPGE1	R/W		00000000
47H	(Disabled)				
48H	Reload register L (ch.2)	PRLL2	R/W	8/16 bit PPG2/3	XXXXXXXX
49H	Reload register H (ch.2)	PRLH2	R/W		XXXXXXXX
4AH	Reload register L (ch.3)	PRLL3	R/W		XXXXXXXX
4BH	Reload register H (ch.3)	PRLH3	R/W		XXXXXXXX
4CH	PPG2 operating mode control register	PPGC2	R/W		0_000__1
4DH	PPG3 operating mode control register	PPGC3	R/W		0_000001
4EH	PPG2 and 3 output control register	PPGE2	R/W		00000000
4FH	(Disabled)				
50H	Reload register L (ch.4)	PRLL4	R/W	8/16 bit PPG4/5	XXXXXXXX
51H	Reload register H (ch.4)	PRLH4	R/W		XXXXXXXX
52H	Reload register L (ch.5)	PRLL5	R/W		XXXXXXXX
53H	Reload register H (ch.5)	PRLH5	R/W		XXXXXXXX
54H	PPG4 operating mode control register	PPGC4	R/W		0_000__1
55H	PPG5 operating mode control register	PPGC5	R/W		0_000001
56H	PPG4 and 5 output control register	PPGE3	R/W		00000000
57H	(Disabled)				
58H	Clock output enable register	CLKR	R/W	Clock monitor function	____0000
59H	(Disabled)				

(Continued)

Address	Register name	Abbreviated register name	Read/write	Resource name	Initial value
5AH	Control status register 0	TMCSR0	R/W	16 bit reload timer 0	0 0 0 0 0 0 0 0
5BH					_ _ _ _ 0 0 0 0
5CH	16 bit timer register 0/ 16 bit reload register 0	TMR0/ TMRLR0	R/W		XXXXXXXX
5DH					XXXXXXXX
5EH	Control status register 1	TMCSR1	R/W	16 bit reload timer 1	0 0 0 0 0 0 0 0
5FH					_ _ _ _ 0 0 0 0
60H	16 bit timer register 1/ 16 bit reload register 1	TMR1/ TMRLR1	R/W		XXXXXXXX
61H					XXXXXXXX
62H	Input capture register, channel-0 lower bits	IPCP0	R	16 bit I/O timer Input capture (ch.0 to ch.3)	XXXXXXXX
63H	Input capture register, channel-0 upper bits				XXXXXXXX
64H	Input capture register, channel-1 lower bits	IPCP1	R		XXXXXXXX
65H	Input capture register, channel-1 upper bits				XXXXXXXX
66H	Input capture register, channel-2 lower bits	IPCP2	R		XXXXXXXX
67H	Input capture register, channel-2 upper bits				XXXXXXXX
68H	Input capture register, channel-3 lower bits	IPCP3	R		XXXXXXXX
69H	Input capture register, channel-3 upper bits				XXXXXXXX
6AH	Input capture control status register	ICS01	R/W		0 0 0 0 0 0 0 0
6BH	Input capture control status register	ICS23	R/W		0 0 0 0 0 0 0 0
6CH	Timer data register, lower bits	TCDT	R/W	16 bit I/O timer free run timer	0 0 0 0 0 0 0 0
6DH	Timer data register, upper bits		R/W		0 0 0 0 0 0 0 0
6EH	Timer control status register	TCCS	R/W		0 0 0 0 0 0 0 0
6FH	ROM mirroring function selection register	ROMM	W	ROM mirroring function	_ _ _ _ _ _ _ 1

(Continued)

Address	Register name	Abbreviated register name	Read/write	Resource name	Initial value
70H	Compare register, channel-0 lower bits	OCCP0	R/W	16 bit I/O timer output compare (ch.0 to ch.3)	XXXXXXXX
71H	Compare register, channel-0 upper bits				XXXXXXXX
72H	Compare register, channel-1 lower bits	OCCP1	R/W		XXXXXXXX
73H	Compare register, channel-1 upper bits				XXXXXXXX
74H	Compare register, channel-2 lower bits	OCCP2	R/W		XXXXXXXX
75H	Compare register, channel-2 upper bits				XXXXXXXX
76H	Compare register, channel-3 lower bits	OCCP3	R/W		XXXXXXXX
77H	Compare register, channel-3 upper bits				XXXXXXXX
78H	Compare control status register, channel-0	OCS0	R/W		0000__00
79H	Compare control status register, channel-1	OCS1	R/W		___00000
7AH	Compare control status register, channel-2	OCS2	R/W		0000__00
7BH	Compare control status register, channel-3	OCS3	R/W		___00000
7CH to 9DH	(Disabled)				
9EH	Program address detection control register	PACSR	R/W	Address match detection function	00000000
9FH	Delayed interrupt factor generation/cancellation register	DIRR	R/W	Delayed interrupt	_______0
A0H	Low-power consumption mode control register	LPMCR	R/W!	Low power consumption control circuit	00011000
A1H	Clock select register	CKSCR	R/W!		11111100
A2H to A4H	(Disabled)				
A5H	Automatic ready function select register	ARSR	W	External bus pin control circuit	0011__00
A6H	External address output control register	HACR	W		00000000
A7H	Bus control signal select register	ECSR	W		0000000_

(Continued)

Address	Register name	Abbreviated register name	Read/write	Resource name	Initial value
A8H	Watchdog timer control register	WDTC	R/W!	Watchdog timer	XXXXX 1 1 1
A9H	Timebase timer control register	TBTC	R/W!	Timebase timer	1 _ _ 0 0 1 0 0
AAH to ADH	(Disabled)				
AEH	Flash control status register	FMCS	R/W	Flash interface circuit	0 0 0 0 0 _ _ 0
AFH	(Disabled)				
B0H	Interrupt control register 00	ICR00	R/W!	Interrupt controller	0 0 0 0 0 1 1 1
B1H	Interrupt control register 01	ICR01	R/W!		0 0 0 0 0 1 1 1
B2H	Interrupt control register 02	ICR02	R/W!		0 0 0 0 0 1 1 1
B3H	Interrupt control register 03	ICR03	R/W!		0 0 0 0 0 1 1 1
B4H	Interrupt control register 04	ICR04	R/W!		0 0 0 0 0 1 1 1
B5H	Interrupt control register 05	ICR05	R/W!		0 0 0 0 0 1 1 1
B6H	Interrupt control register 06	ICR06	R/W!		0 0 0 0 0 1 1 1
B7H	Interrupt control register 07	ICR07	R/W!		0 0 0 0 0 1 1 1
B8H	Interrupt control register 08	ICR08	R/W!		0 0 0 0 0 1 1 1
B9H	Interrupt control register 09	ICR09	R/W!		0 0 0 0 0 1 1 1
BAH	Interrupt control register 10	ICR10	R/W!		0 0 0 0 0 1 1 1
BBH	Interrupt control register 11	ICR11	R/W!		0 0 0 0 0 1 1 1
BCH	Interrupt control register 12	ICR12	R/W!		0 0 0 0 0 1 1 1
BDH	Interrupt control register 13	ICR13	R/W!		0 0 0 0 0 1 1 1
BEH	Interrupt control register 14	ICR14	R/W!		0 0 0 0 0 1 1 1
BFH	Interrupt control register 15	ICR15	R/W!		0 0 0 0 0 1 1 1
C0H to FFH	(External area)				
100H to #H	(RAM area)				
#H to 1FEFH	(Reserved area)				

(Continued)

(Continued)

Address	Register name	Abbreviated register name	Read/write	Resource name	Initial value
1FF0H	Program address detection register 0	PADR0	R/W	Address match detection function	XXXXXXXX
1FF1H	Program address detection register 1		R/W		XXXXXXXX
1FF2H	Program address detection register 2		R/W		XXXXXXXX
1FF3H	Program address detection register 3	PADR1	R/W		XXXXXXXX
1FF4H	Program address detection register 4		R/W		XXXXXXXX
1FF5H	Program address detection register 5		R/W		XXXXXXXX
1FF6H to 1FFFH	(Reserved area)				

- Initial value representations
  0: Initial value of 0
  1: Initial value of 1
  X: Initial value undefined
  –: Initial value undefined (none)
- Addresses that follow 00FFH are a reserved area.
- The boundary #H between the RAM and reserved areas is different depending on each product.

Note : For writable bits, the initial value column contains the initial value to which the bit is initialized at a reset. Notice that it is not the value read from the bit.
The LPMCR, CKSCR, and WDTC registers may be initialized or not at a reset, depending on the type of the reset. Their initial values in the above list are those to which the registers are initialized, of course.
“R/W!” in the access column indicates that the register contains read-only or write-only bits.
If a read-modify-write instruction (such as a bit setting instruction) is used to access a register marked “R/W!” “R/W*”, or “W” in the access column, the bit focused on by the instruction is set to the desired value but a malfunction occurs if the other bits contains a write-only bit. Do not use such instructions to access those registers.

■ INTERRUPT FACTORS

INTERRUPT VECTORS, INTERRUPT CONTROL REGISTERS

Interrupt source	EI²OS support	Interrupt vectors		Interrupt control registers	
		Number	Address	ICR	Address
Reset	×	# 08	FFFFDC$_H$	—	—
INT9 instruction	×	# 09	FFFFD8$_H$	—	—
Exception	×	# 10	FFFFD4$_H$	—	—
A/D converter	○	# 11	FFFFD0$_H$	ICR00	0000B0$_H$
Timebase timer	×	# 12	FFFFCC$_H$		
DTP0 (external interrupt 0)	○	# 13	FFFFC8$_H$	ICR01	0000B1$_H$
DTP4/5 (external interrupt 4/5)	○	# 14	FFFFC4$_H$		
DTP1 (external interrupt 1)	○	# 15	FFFFC0$_H$	ICR02	0000B2$_H$
8/16-bit PPG timer0 counter borrow	×	# 16	FFFFBC$_H$		
DTP2 (external interrupt 2)	○	# 17	FFFFB8$_H$	ICR03	0000B3$_H$
8/16-bit PPG timer 1 counter borrow	×	# 18	FFFFB4$_H$		
DTP3 (external interrupt 3)	○	# 19	FFFFB0$_H$	ICR04	0000B4$_H$
8/16-bit PPG timer 2 counter borrow	×	# 20	FFFFAC$_H$		
Extended I/O serial interface 0	○	# 21	FFFFA8$_H$	ICR05	0000B5$_H$
8/16-bit PPG timer 3 counter borrow	×	# 22	FFFFA4$_H$		
Extended I/O serial interface 1	○	# 23	FFFFA0$_H$	ICR06	0000B6$_H$
16-bit free-run timer (I/O timer) overflow	○	# 24	FFFF9C$_H$		
16-bit re-load timer 0	○	# 25	FFFF98$_H$	ICR07	0000B7$_H$
DTP6/7 (external interrupt 6/7)	○	# 26	FFFF94$_H$		
16-bit re-load timer 1	○	# 27	FFFF90$_H$	ICR08	0000B8$_H$
8/16-bit PPG timer 4/5 counter borrow	×	# 28	FFFF8C$_H$		
Input capture (ch.0) include (I/O timer)	○	# 29	FFFF88$_H$	ICR09	0000B9$_H$
Input capture (ch.1) include (I/O timer)	○	# 30	FFFF84$_H$		
Input capture (ch.2) include (I/O timer)	○	# 31	FFFF80$_H$	ICR10	0000BA$_H$
Input capture (ch.3) include (I/O timer)	○	# 32	FFFF7C$_H$		
Output compare (ch.0) match (Output timer)	○	#33	FFFF78$_H$	ICR11	0000BB$_H$
Output compare (ch.1) match (Output timer)	○	# 34	FFFF74$_H$		
Output compare (ch.2) match (Output timer)	○	# 35	FFFF70$_H$	ICR12	0000BC$_H$
Output compare (ch.3) match (Output timer)	○	# 36	FFFF6C$_H$		
UART0 transmission complete	○	# 37	FFFF68$_H$	ICR13	0000BD$_H$
I²C interface 0	×	# 38	FFFF64$_H$		
UART0 reception complete	◎	# 39	FFFF60$_H$	ICR14	0000BE$_H$
I²C interface 1	×	# 40	FFFF5C$_H$		
Flash memory status	×	# 41	FFFF58$_H$	ICR15	0000BF$_H$
Delayed interrupt generation module	×	# 42	FFFF54$_H$		

○ : The interrupt request flag is cleared by the EI²OS interrupt clear signal.

× : The interrupt request flag is not cleared by the EI²OS interrupt clear signal.

◎ : The interrupt request flag is cleared by the EI²OS interrupt clear signal. The stop request is available.

Note: On using the EI²OS Function with Extended I/O Serial Interface 2

If a resource has two interrupt sources for the same interrupt number, both of the interrupt request flags are cleared by the EI²OS interrupt clear signal. When the EI²OS function is used for one of the two interrupt sources, therefore, the other interrupt function cannot be used. Set the interrupt request enable bit for the relevant resource to 0 for software polling processing.

Interrupt source	Interrupt No.	Interrupt control register	Resource interrupt request
Extended I/O serial interface 1	# 23	ICR06	Enabled
16-bit free-run timer (I/O timer) overflow	# 24		Disabled

■ 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} + 6.0	V	
	AV_{CC}	V_{SS} – 0.3	V_{SS} + 6.0	V	$V_{CC} \geq AV_{CC}$ *1
	AVRH	V_{SS} – 0.3	V_{SS} + 6.0	V	$AV_{CC} \geq AVRH \geq AVRL$
	AVRL	V_{SS} – 0.3	V_{SS} + 6.0	V	
Input voltage	V_I	V_{SS} – 0.3	V_{CC} + 0.3	V	
Output voltage	V_O	V_{SS} – 0.3	V_{CC} + 0.3	V	
“L” level maximum output current*2	I_{OL1}	—	10	mA	Other than P20 to P27
	I_{OL2}	—	20	mA	P20 to P27
“L” level average output current	I_{OLAV1}	—	4	mA	Other than P20 to P27
	I_{OLAV2}	—	12	mA	P20 to P27
“L” level total maximum output current	$\sum I_{OL}$	—	150	mA	
“L” level total average output current	$\sum I_{OLAV}$	—	80	mA	*5
“H” level maximum output current	I_{OH}*2	—	–15	mA	
“H” level average output current	I_{OHAV}*3	—	–4	mA	*5
“H” level total maximum output current	$\sum I_{OH}$	—	–100	mA	
“H” level total average output current	$\sum I_{OHAV}$*4	—	–50	mA	*5
Power consumption	P_D	—	500	mW	
Operating temperature	T_A	–40	+85	°C	
Storage temperature	T_{STG}	–55	+150	°C	

*1 : Be careful not to let AVcc exceed Vcc, for example, when the power supply is turned on.

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

*3 : Average output current is an average current value observed for a 100 ms period for a corresponding pin.

*4 : Total average current is an average current value observed for a 100 ms period for all corresponding pins.

*5 : Average output current = operating current × operating efficiency

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

2. Recommended Operating Conditions

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

Parameter	Symbol	Value		Unit	Remarks
		Min.	Max.		
Power supply voltage	V_{CC} AV_{CC}	4.5	5.5	V	Normal operation (MB90F553A, MB90P553A, MB90V550A)
		3.5	5.5	V	Normal operation (MB90553A, MB90552A)
		3.5	5.5	V	Retains status at the time of operation stop
"H" level input voltage	V_{IH}	0.7V_{CC}	V_{CC}+0.3	V	CMOS input pin*1
	V_{IHS}	0.8V_{CC}	V_{CC}+0.3	V	CMOS hysteresys input pin*2
	V_{IHM}	V_{CC} – 0.3	V_{CC}+0.3	V	MD pin input*3
"L" level input voltage	V_{IL}	V_{SS} – 0.3	0.3V_{CC}	V	CMOS input pin*1
	V_{ILS}	V_{SS} – 0.3	0.2V_{CC}	V	CMOS hysteresys input pin*2
	V_{ILM}	V_{SS} – 0.3	V_{SS} +0.3	V	MD pin input*3
Smoothing capacitor*4	C_S	0.1	1.0	µF	*5
Operating temperature	T_A	–40	+85	°C	

*1 : P00 to P07, P10 to P17, P20 to P27, P30 to P37

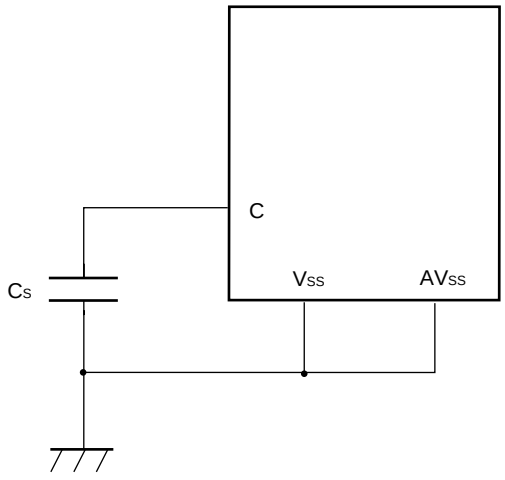
*2 : X0, $\overline{HST}$, $\overline{RST}$, P40 to P47, P50 to P55, P60 to P67, P70 to P77, P80 to P87, P90 to P97, PA0 to PA4

*3 : MD0, MD1, MD2

*4 : For connecting smoothing capacitor C_S, see the diagram below:

*5 : Use a ceramic capacitor or a capacitor with equivqlent frequency characteristics. The smoothing capacitor to be connected to the V_{CC} pin must have a capacitance value higher than C_S.

- **C pin connection circuit**



WARNING: The recommended operating conditions are required in order to ensure the normal operation of the semiconductor device. All of the device's electrical characteristics are warranted when the device is operated within these ranges.
Always use semiconductor devices within their recommended operating condition ranges. Operation outside these ranges may adversely affect reliability and could result in device failure.
No warranty is made with respect to uses, operating conditions, or combinations not represented on the data sheet. Users considering application outside the listed conditions are advised to contact their FUJITSU representatives beforehand.

3. DC Characteristics

(V_{CC} = 5.0 V ± 10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
Open-drain output pin voltage	V_D	P50 to P55	—	V_{SS} – 0.3	—	V_{SS} + 6.0	V	
"H" level output voltage	V_{OH}	Other than P50 to P55	V_{CC} = 4.5V, I_{OH} = –4.0mA	V_{CC} – 0.5	—	—	V	
"L" level output voltage 1	V_{OL1}	Other than P20 to P27	V_{CC} = 4.5V, I_{OL} = 4.0mA	—	—	0.4	V	
"L" level output voltage 2	V_{OL2}	P20 to P27	V_{CC} = 4.5V, I_{OL} = 12.0mA	—	—	0.4	V	
Input leakage current	I_{IL}	All output pins	V_{CC} = 5.5V, V_{SS} < V_I < V_{CC}	–5	—	5	µA	
Power supply current*	I_{CC}	V_{CC}	Internal operation at 16 MHz V_{CC} = 5.5 V Normal operation	—	30	40	mA	MB90V550A
				—	80	110	mA	MB90P553A
				—	60	90	mA	MB90F553A
				—	30	40	mA	MB90553A
				—	25	35	mA	MB90552A
			When data written in flash mode	—	100	150	mA	MB90F553A
	I_{CCS}		Internal operation at 16 MHz V_{CC} = 5.5 V In sleep mode	—	7	10	mA	MB90V550A
				—	25	30	mA	MB90P553A
				—	10	20	mA	MB90F553A
				—	7	10	mA	MB90553A
				—	7	10	mA	MB90552A
	I_{CCH}		V_{CC} = 5.5V, T_A = +25°C In stop mode	—	5	20	µA	MB90V550A
				—	0.1	10	µA	MB90P553A
				—	5	20	µA	MB90F553A
				—	5	20	µA	MB90553A
				—	5	20	µA	MB90552A
Input capacitance	C_{IN}	Other than AV_{CC}, AV_{SS}, C, V_{CC} and V_{SS}	—	—	10	—	pF	
Open-drain output leakage current	I_{leak}	P50 to P55	—	—	0.1	5	µA	
Pull-up resistance	R_{UP}	P00 to P07 and P10 to P17 (In pull-up setting), $\overline{RST}$	—	25	50	100	kΩ	Other than MB90V550A
				20	40	100	kΩ	MB90V550A

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

4. AC Characteristics

(1) Clock Timing

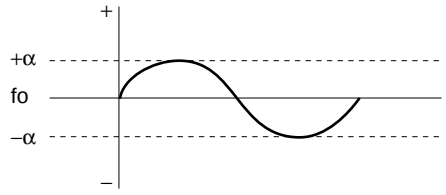
(V_{CC} = 5.0 V±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Value			Unit	Unit
			Min.	Typ.	Max.		
Oscillation clock frequency	F_C	X0, X1	3	—	16	MHz	
Oscillation clock cycle time	t_C	X0, X1	62.5	—	333	ns	
Frequency fluctuation rate locked*	Δf	—	—	—	5	%	
Input clock pulse width	P_{WH} P_{WL}	X0	10	—	—	ns	Recommended duty ratio of 40% to 60%
Input clock rising/falling time	t_{CR}, t_{CF}	X0	—	—	5	ns	External clock operation
Internal operating clock frequency	F_{CP}	—	8.0	—	16	MHz	PLL operation
			1.5	—	16	MHz	When PLL is not used
Internal operating clock cycle time	t_{CP}	—	62.5	—	125	ns	PLL operation
			62.5	—	666	ns	When PLL is not used

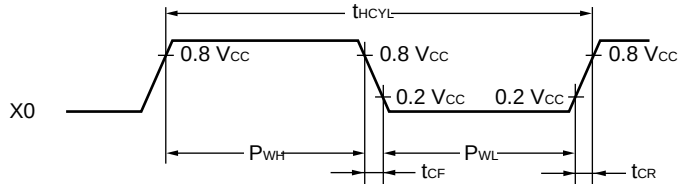
* :The frequency fluctuation rate is the maximum deviation rate of the preset center frequency when the multiplied PLL signal is locked.

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

Center frequency

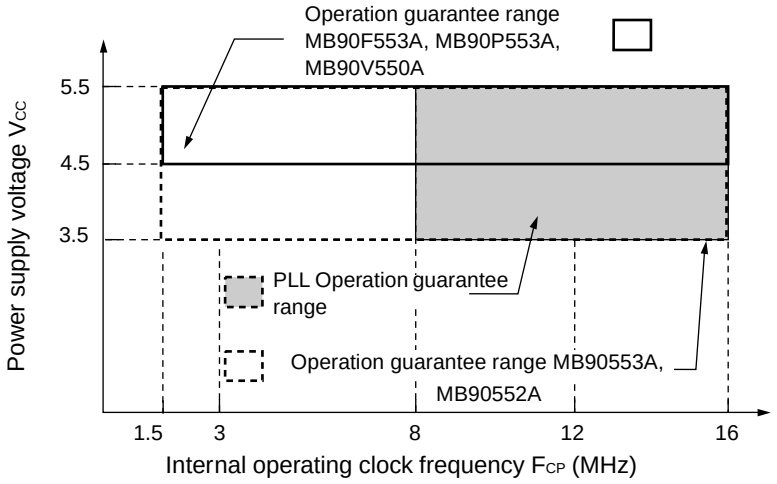


- **X0, X1 clock timing**

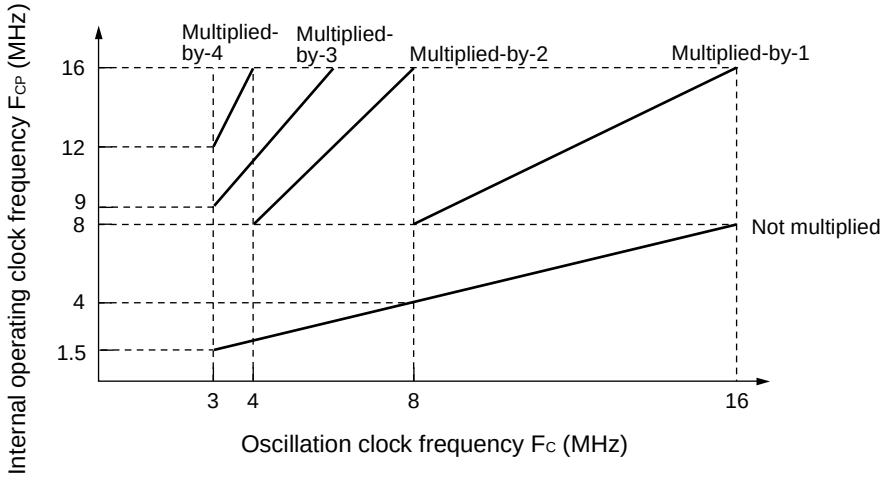


- **PLL operation guarantee range**

Relationship between internal operating clock frequency and power supply voltage



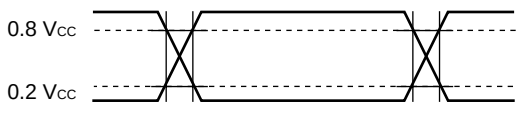
Relationship between oscillation clock frequency and internal operating clock frequency



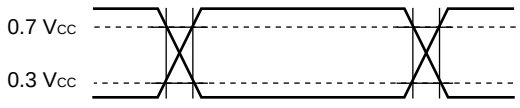
The AC ratings are measured for the following measurement reference voltages

- **Input signal waveform**

Hystheresis input pin

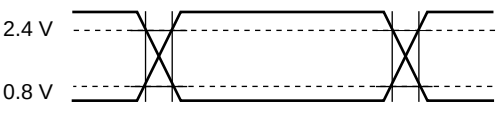


Pins other than hystheresis input/MD input



- **Output signal waveform**

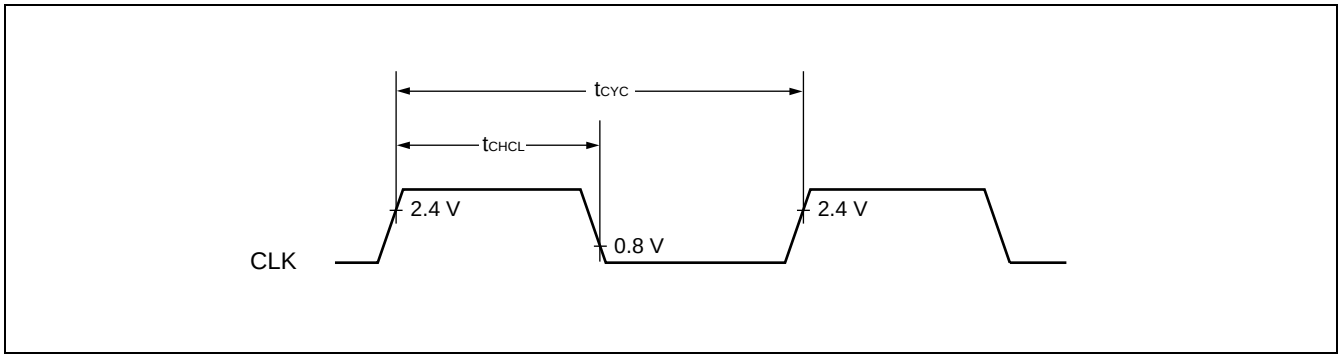
Output pin



(2) Clock Output Timing

(V_{CC} = 5.0 V ±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

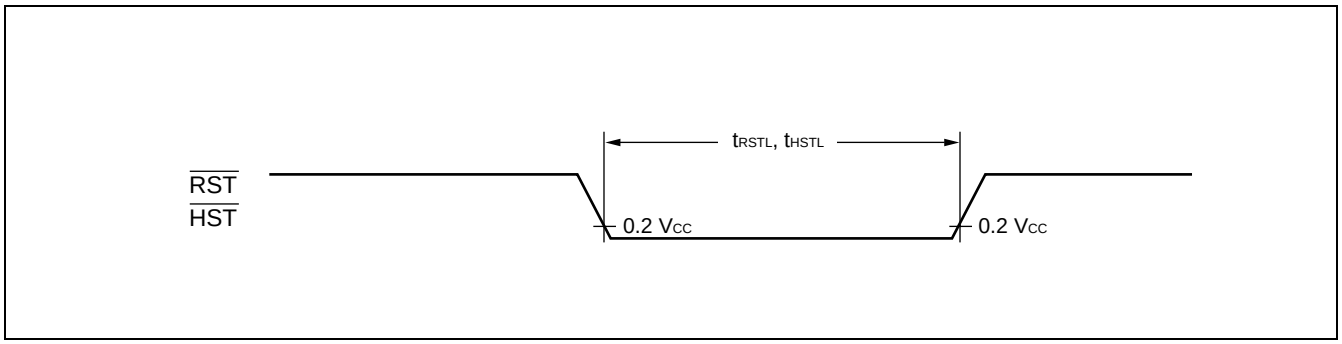
Parameter	Symbol	Pin name	Value		Unit	Remarks
			Min.	Max.		
Cycle time	t_{CYC}	CLK	t_{CP}	—	ns	
CLK ↑ → CLK ↓ time	t_{CHCL}		$t_{CP}/2 - 20$	$t_{CP}/2+20$	ns	



(3) Reset, Hardware Standby Input Timing

(V_{CC} = 5.0 V ±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

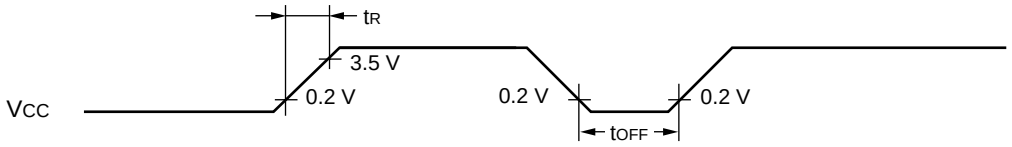
Parameter	Symbol	Pin name	Value		Unit	Remarks
			Min.	Max.		
Reset input time	t_{RSTL}	RST	16 t_{CP}	—	ns	
Hardware standby input time	t_{HSTL}	HST	16 t_{CP}	—	ns	



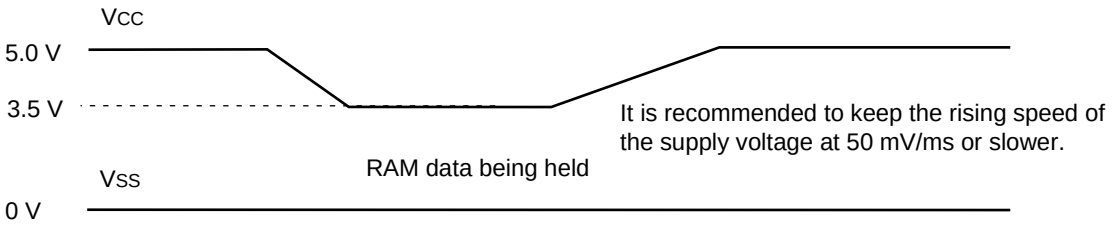
(4) Specification for Power-on Reset

(V_{CC} = 5.0 V ± 10 %, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Value		Unit	Remarks
			Min.	Max.		
Power supply rising time	t_R	V_{CC}	0.066	30	ms	
Power-supply start voltage	V_{OFF}		—	0.2	V	
Power-supply end voltage	V_{ON}		3.5	—	V	
Power supply cut-off time	t_{OFF}		4	—	ms	Due to repeated operations



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

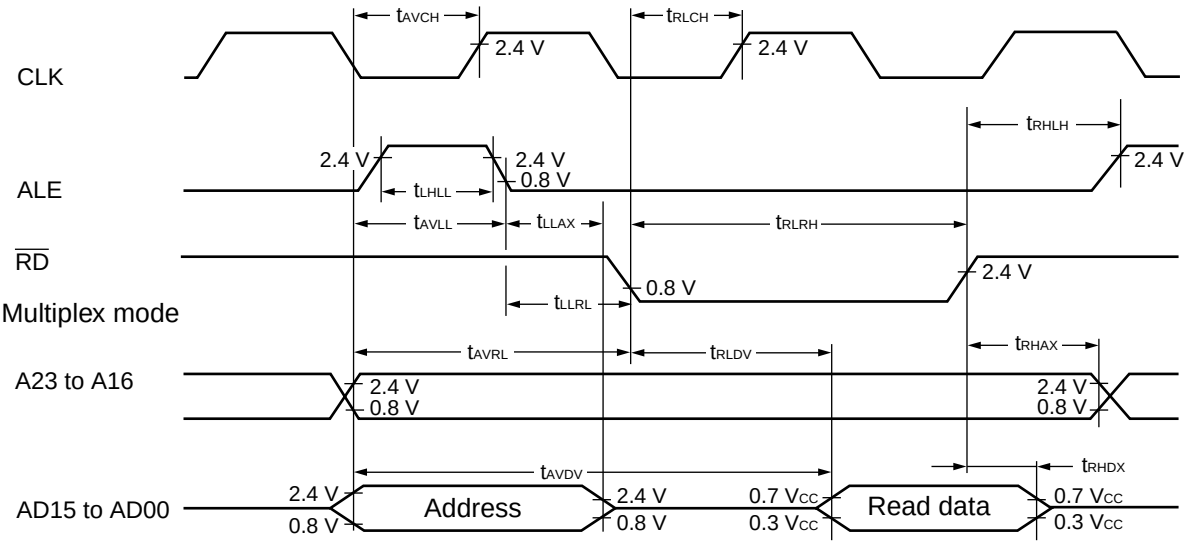


(5) Bus Read Timing

(V_{CC} = 5.0 V±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Value		Unit	Remarks
			Min.	Max.		
ALE pulse width	t_{LHLL}	ALE	$t_{CP}/2 - 20$		ns	
Effective address → ALE ↓ time	t_{AVLL}	ALE, A23 to A16, AD15 to AD00	$t_{CP}/2 - 20$	—	ns	
ALE ↓ → address effective time	t_{LLAX}	ALE, AD15 to AD00	$t_{CP}/2 - 15$	—	ns	
Effective address → $\overline{RD}$ ↓ time	t_{AVRL}	A23 to A16, AD15 to AD00, $\overline{RD}$	$t_{CP} - 15$	—	ns	
Effective address → valid data input	t_{AVDV}	A23 to A16, AD15 to AD00	—	$5\ t_{CP}/2 - 60$	ns	
$\overline{RD}$ pulse width	t_{RLRH}	$\overline{RD}$	$3\ t_{CP}/2 - 20$	—	ns	
$\overline{RD}$ ↓ → valid data input	t_{RLDV}	$\overline{RD}$, AD1 to AD00	—	$3\ t_{CP}/2 - 60$	ns	
$\overline{RD}$ ↑ → data hold time	t_{RHDX}	$\overline{RD}$, AD15 to AD00	0	—	ns	
$\overline{RD}$ ↑ → ALE ↑ time	t_{RHLH}	$\overline{RD}$, ALE	$t_{CP}/2 - 15$	—	ns	
$\overline{RD}$ ↑ → address effective time	t_{RHAX}	$\overline{ALE}$, A23 to A16	$t_{CP}/2 - 10$	—	ns	
Effective address → CLK ↑ time	t_{AVCH}	A23 to A16, AD15 to AD00, CLK	$t_{CP}/2 - 20$	—	ns	
$\overline{RD}$ ↓ → CLK ↑ time	t_{RLCH}	$\overline{RD}$, CLK	$t_{CP}/2 - 20$	—	ns	
ALE ↓ → $\overline{RD}$ ↓ time	t_{LLRL}	ALE, $\overline{RD}$	$t_{CP}/2 - 15$	—	ns	

- **Bus read timing**



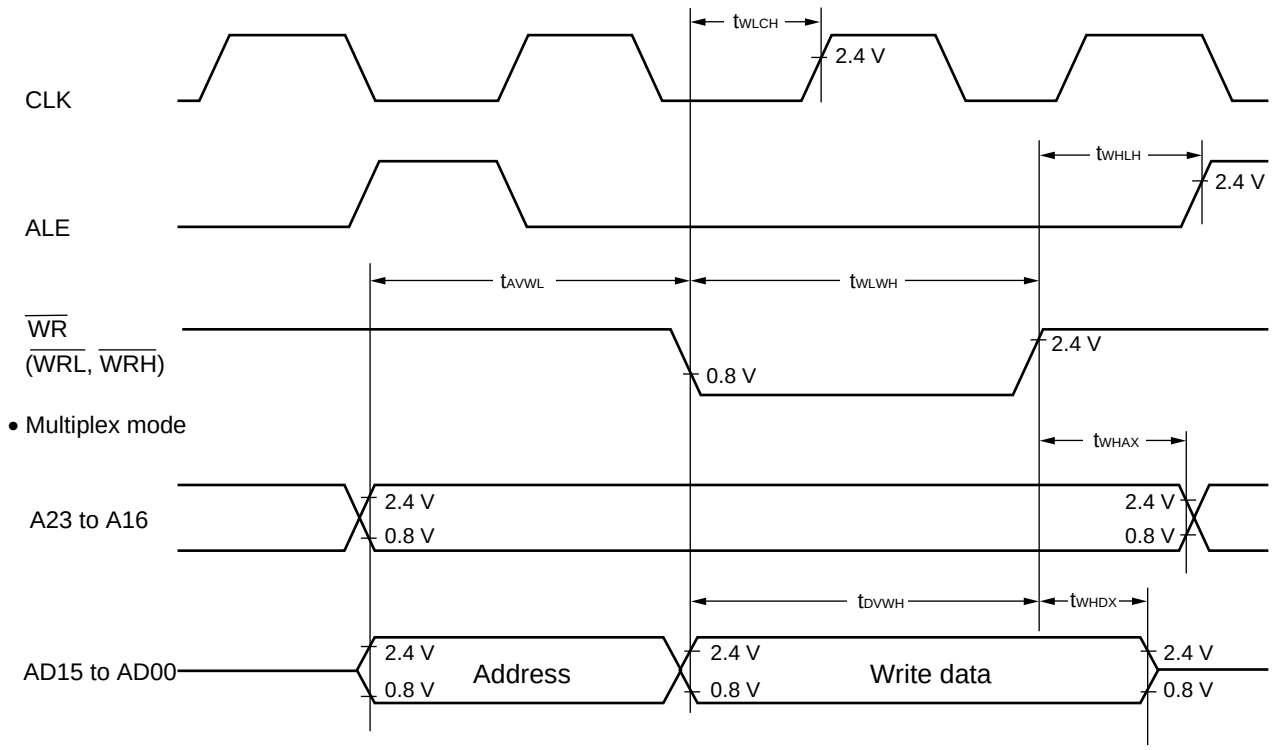
- Multiplex mode

(6) Bus Write Timing

(V_{CC} = 5.0 V±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Value		Unit	Remarks
			Min.	Max.		
Effective address → $\overline{WR}$ ↓ time	t_{AVWL}	A23 to A16, AD15 to AD00, $\overline{WRH}$, $\overline{WRL}$	t_{CP} – 15	—	ns	
$\overline{WR}$ pulse width	t_{WLWH}	$\overline{WRH}$, $\overline{WRL}$	3 t_{CP}/2 – 20	—	ns	
valid data output→ $\overline{WR}$ ↑ time	t_{DVWH}	AD15 to AD00, $\overline{WRH}$, $\overline{WRL}$	3 t_{CP}/2 – 20	—	ns	
$\overline{WR}$ ↑ → data hold time	t_{WHDX}	AD15 to AD00, $\overline{WRH}$, $\overline{WRL}$	20	—	ns	Multiplex mode
$\overline{WR}$ ↑ → address effective time	t_{WHAX}	A23 to A16, $\overline{WRH}$, $\overline{WRL}$	t_{CP}/2 – 10	—	ns	
$\overline{WR}$ ↑ → ALE ↑ time	t_{WHLH}	$\overline{WRH}$, $\overline{WRL}$, ALE	t_{CP}/2 – 15	—	ns	
$\overline{WR}$ ↓ → CLK ↑ time	t_{WLCH}	$\overline{WRH}$, $\overline{WRL}$, CLK	t_{CP}/2 – 20	—	ns	

- **Bus write timing**



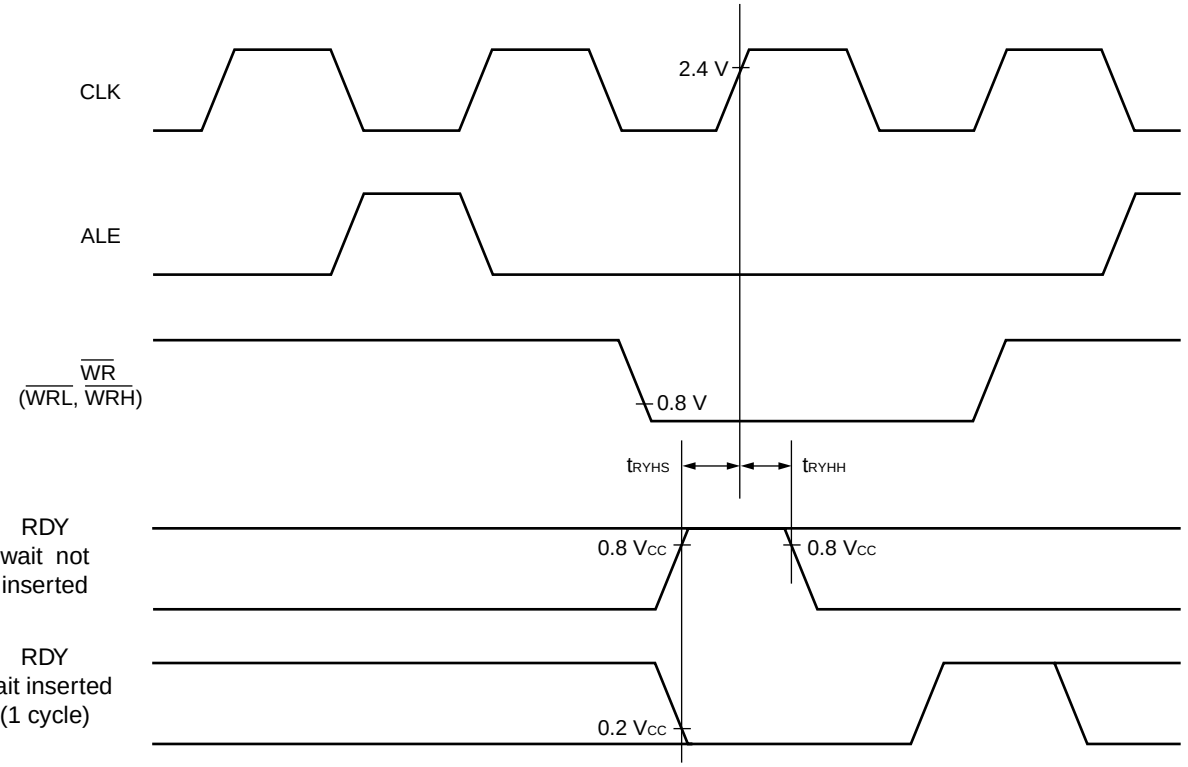
(7) Ready Input Timing

(V_{CC} = 5.0 V±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Value		Unit	Remarks
			Min.	Max.		
RDY setup time	t_{RYHS}	RDY CLK	45	—	ns	
RDY hold time	t_{RYHH}		0	—	ns	

Note : Use the automatic ready function when the setup time for the rising edge of the RDY signal is not sufficient.

- **Ready input timing**



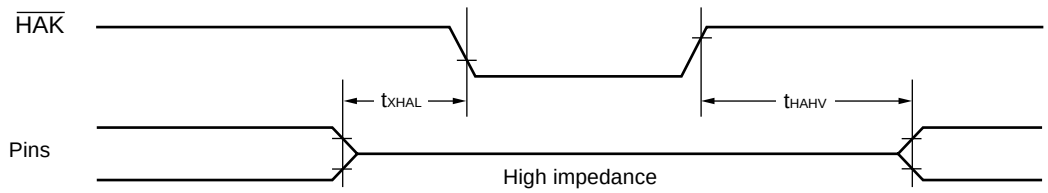
(8) Hold Timing

(V_{CC} = 5.0 V±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

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

Note : More than 1 machine cycle is needed before $\overline{HAK}$ changes after HRQ pin is fetched.

- **Hold timing**



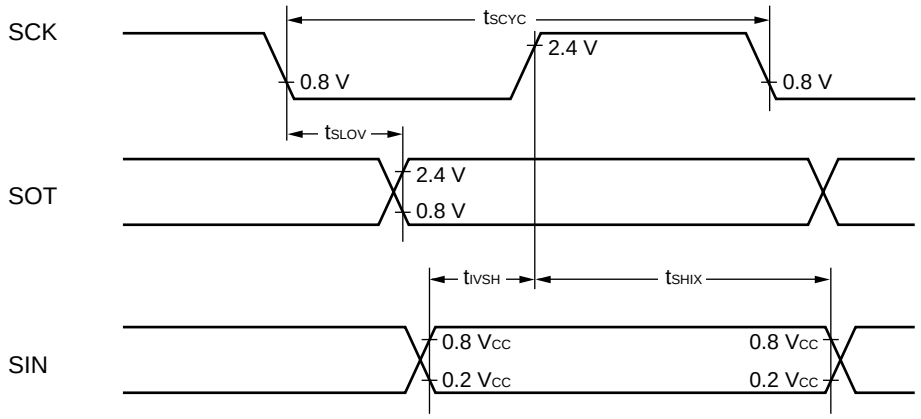
(9) UART, Extended I/O Sirial 0, 1 Timing

(V_{CC} = 5.0 V±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

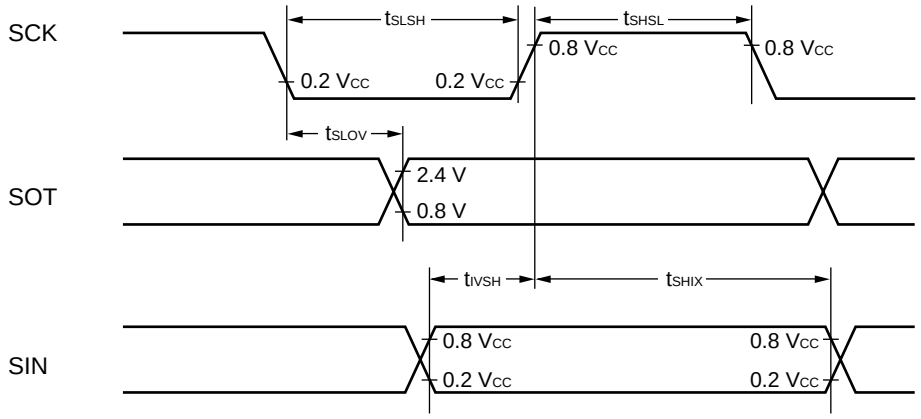
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Serial clock cycle time	t_{SCYC}	SCK0 to SCK2	Internal shift clock mode C_L = 80 pF + 1 TTL for an output pin	8 t_{CP}	—	ns	
SCK ↓ → SOT delay time	t_{SLOV}	SCK0 to SCK2, SOT0 to SOT2		–80	80	ns	
Valid SIN → SCK ↑	t_{IVSH}	SCK0 to SCK2, SIN0 to SIN2		100	—	ns	
SCK ↑ → valid SIN hold time	t_{SHIX}	SCK0 to SCK2, SIN0 to SIN2		t_{CP}	—	ns	
Serial clock “H” pulse width	t_{SHSL}	SCK0 to SCK2	External shift clock mode C_L = 80 pF + 1 TTL for an output pin	4 t_{CP}	—	ns	
Serial clock “L” pulse width	t_{SLSH}	SCK0 to SCK2		4 t_{CP}	—	ns	
SCK ↓ → SOT delay time	t_{SLOV}	SCK0 to SCK2, SOT0 to SOT2		—	150	ns	
Valid SIN → SCK ↑	t_{IVSH}	SCK0 to SCK2, SIN0 to SIN2		60	—	ns	
SCK ↑ → valid SIN hold time	t_{SHIX}	SCK0 to SCK2, SIN0 to SIN2		60	—	ns	

Notes: • These are AC ratings in the CLK synchronous mode.
• C_L is the load capacitance value connected to pins while testing.

- **Internal shift clock mode**



- **External shift clock mode**

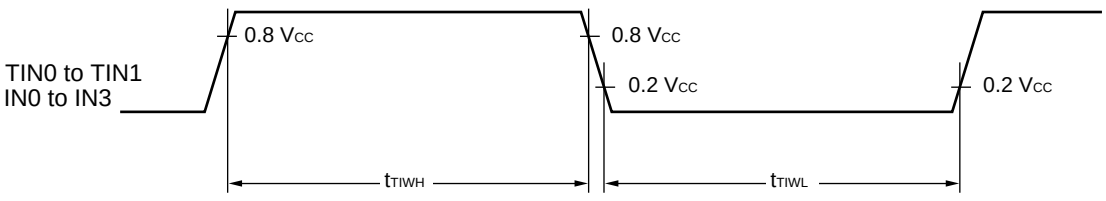


(10) Timer Input Timing

(V_{CC} = 5.0 V±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Value		Unit	Remarks
			Min.	Max.		
Input pulse width	t_{TIWH} t_{TIWL}	TIN0, TIN1 IN0 to IN3	4 t_{CP}	—	ns	

- **Timer input timing**

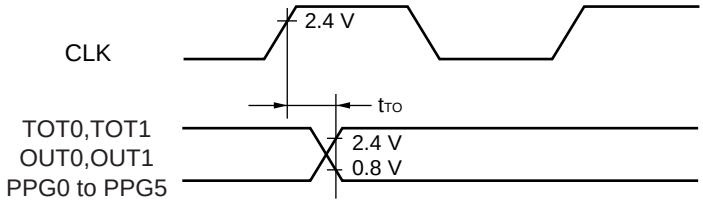


(11) Timer Output Timing

(V_{CC} = 5.0 V±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Value		Unit	Remarks
			Min.	Max.		
CLK ↑ → T_{OUT} transition time	t_{TO}	TOT0,TOT1,OUT0, OUT1,PPG0 to PPG5	30	—	ns	

- **Timer output timing**

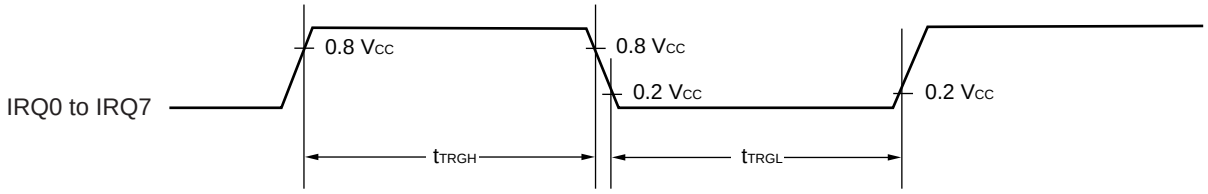


(12) Trigger Input Timing

(V_{CC} = 5.0 V±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Value		Unit	Remarks
			Min.	Max.		
Input pulse width	t_{TRGL}	IRQ0 to IRQ7	5 t_{CP}	—	ns	

- **Trigger input timing**



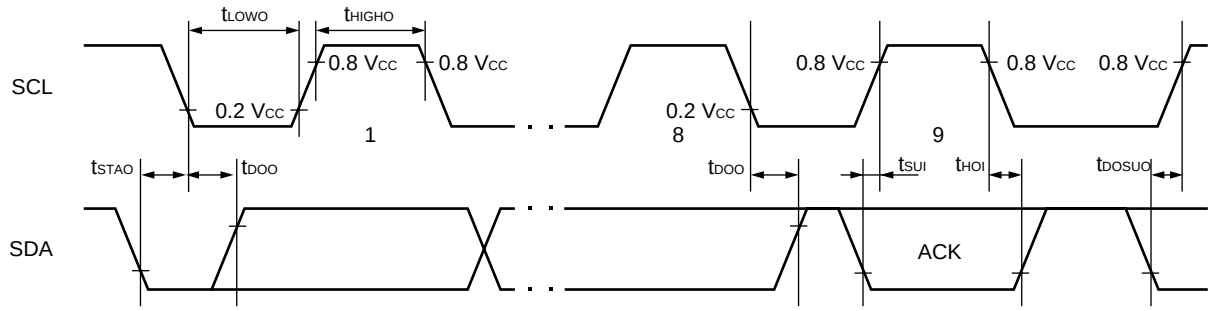
(13) I²C Interface

(V_{CC} = 5.0 V±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

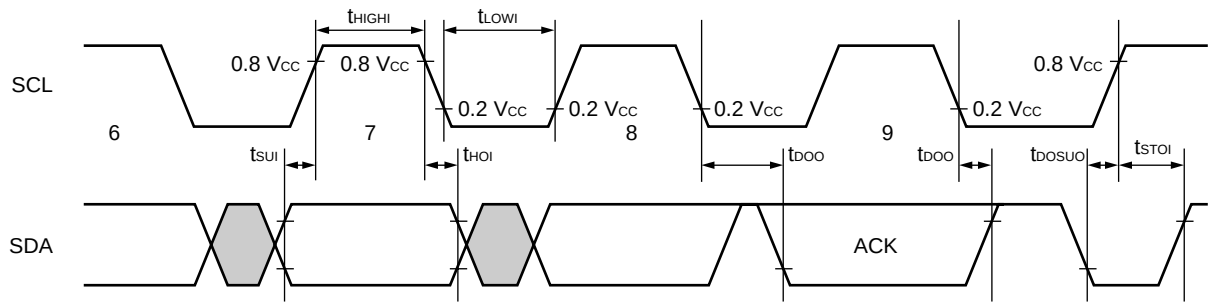
Parameter	Symbol	Pin name	Value		Unit	Remarks
			Min.	Max.		
Internal clock cycle time	t_{CP}	—	62.5	666	ns	All products
Start condition output	t_{STAO}	SDA0 to SDA2 SCL0 toSCL2	$t_{CP} \times m \times n/2 - 20$	$t_{CP} \times m \times n/2 + 20$	ns	Only as master
Stop condition output	t_{STOO}		t_{CP} (m × n/2 + 4) – 20	t_{CP} (m × n/2 + 4) + 20	ns	
Start condition detection	t_{STAI}		3 t_{CP} + 40	—	ns	Only as slave
Stop condition detection	t_{STOI}		3 t_{CP} + 40	—	ns	
SCL output “L” width	t_{LOWO}	SCL0 to SCL2	$t_{CP} \times m \times n/2 - 20$	$t_{CP} \times m \times n/2 + 20$	ns	Only as master
SCL output “H” width	t_{HIGHO}		t_{CP} (m × n/2 + 4) – 20	t_{CP} (m × n/2 + 4) + 20	ns	
SDA output delay time	t_{DOO}	SDA0 to SDA2 SCL0 to SCL2	2 t_{CP} – 20	2 t_{CP} + 20	ns	
Setup after SDA output interrupt period	t_{DOSUO}		4 t_{CP} – 20	—	ns	
SCL input “L” width	t_{LOWI}	SCL0 to SCL2	3 t_{CP} + 40	—	ns	
SCL input “H” width	t_{HIGHI}		t_{CP} + 40	—	ns	
SDA input setup time	t_{SUI}	SDA0 to SDA2 SCL0 to SCL2	40	—	ns	
SDA input hold time	t_{HOI}		0	—	ns	

Notes: • “m” and“n” in the above table represent the values of shift clock frequency setting bits (CS4 to CS0) in the clock control register “ICCR”. For details, refer to the register description in the hardware manual.
• t_{DOSUO} represents the minimum value when the interrupt period is equal to or greater than the SCL “L” width.
• The SDA and SCL output values indicate that that rise time is 0 ns.

- **I²C interface [data transmitter (master/slave)]**



- **I²C interface [data receiver (master/slave)]**



5. A/D Converter

(1)Electrical Characteristics

(4.5 V ≤ AVRH – AVRL, V_{CC} = AV_{CC} = 5.0 V±10%, V_{SS} = AV_{SS} = 0.0 V, T_A = –40 °C to +85 °C)

Parameter	Symbol	Pin name	Value			Unit	Remarks
			Min.	Typ.	Max.		
Resolution	—	—	—	10	—	bit	
Total error	—	—	—	—	±5.0	LSB	
Non-linear error	—	—	—	—	±2.5	LSB	
Differential linearity error	—	—	—	—	±1.9	LSB	
Zero transition voltage	V_{OT}	AN0 to AN7	AVRL–3.5LSB	AVRL+0.5LSB	AVRL+4.5LSB	V	1LSB= (AVRH–AVRL) /1024
Full-scale transition voltage	V_{FST}	AN0 to AN7	AVRH–6.5LSB	AVRH–1.5LSB	AVRH+1.5LSB	V	
Sampling period	t_{SMP}	—	64	—	4096	t_{CP}	
Compare time	t_{CMP}	—	22	—	—	µs	*1
A/D Conversion time	t_{CNV}	—	26.3	—	—	µs	*2
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	—	AV_{CC}	V	
	—	AVRL	0	—	AVRH	V	
Power supply current	I_A	AV_{CC}	—	3.5	7.0	mA	
	I_{AH}		—	—	5	µA	*3
Reference voltage supply current	I_R	AVRH	—	300	500	µA	
	I_{RH}		—	—	5	µA	*3
Offset between channels	—	AN0 to AN7	—	—	4	LSB	

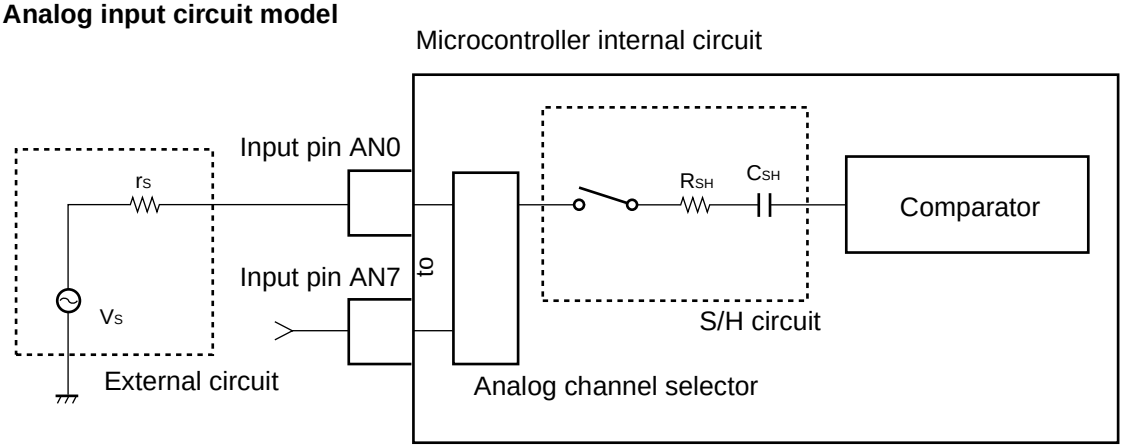
*1: When F_{CP} = 8 MHz, $t_{CMP} = 176 \times t_{CP}$. When F_{CP} = 16 MHz, $t_{CMP} = 352 \times t_{CP}$.

*2: Equivalent to the time for conversion per channel if "$t_{SMP} = 64 \times t_{CP}$" or "$t_{CMP} = 352 \times t_{CP}$" is selected when F_{CP} = 16 MHz.

*3: Specifies the power-supply current (Vcc = AVcc = AVRH = 5.0 V) when the A/D converter is inactive and the CPU has been stopped.

Notes:
- The error becomes larger relatively as |AVRH-AVRL| becomes smaller.
- Use the output impedance r_S of the external circuit for analog input under the following condition: External circuit output impedance r_S = 10 kΩ max.
- If the output impedance of the external circuit is too high, the analog voltage sampling time may be insufficient.
- If you insert a DC-blocking capacitor between the external circuit and the input pin, select the capacitance about several thousands times the sampling capacitance C_{SH} in the chip to suppress the effect of capacity potential division with C_{SH}.

- **Analog input circuit model**



<Recommended/reference values for device parameters>

r_S = 10 kΩ or less

R_{SH} = About 3 kΩ

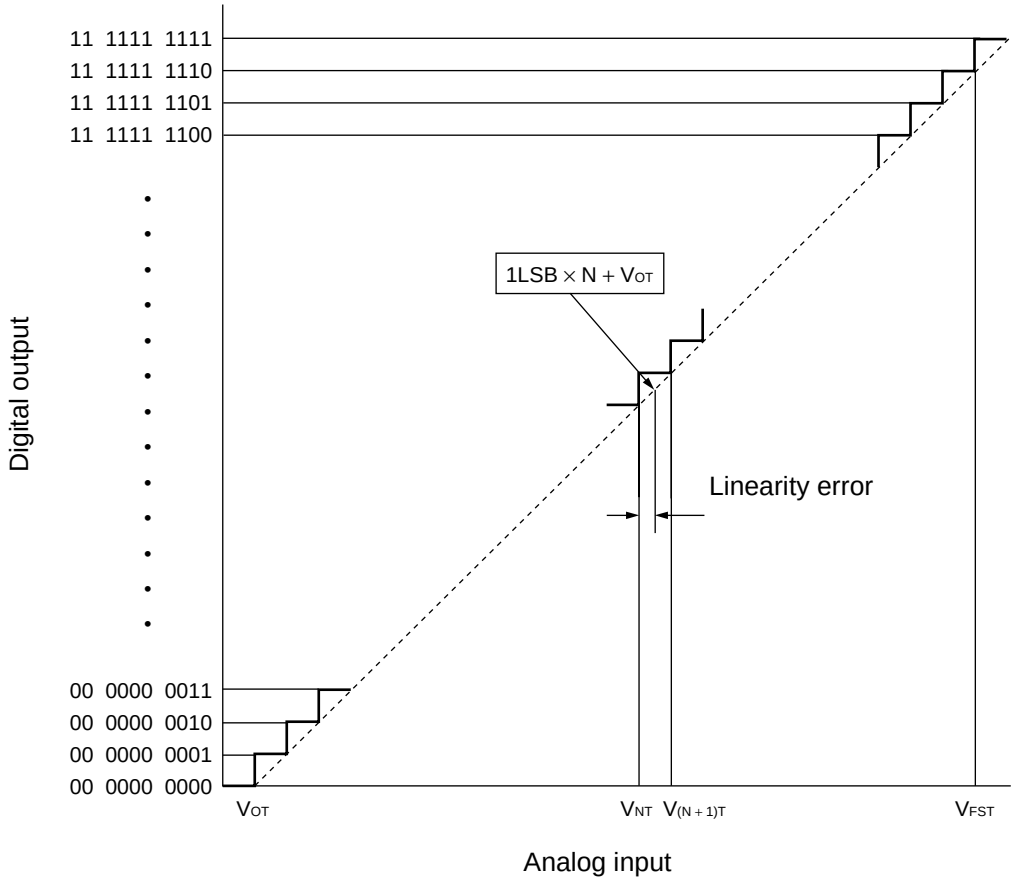
C_{SH} = About 25 pF

Note: Device parameter values are provided as reference values for design purposes; they are not guaranteed.

(2) Definitions of Terms

- Resolution: Analog transition identifiable by the A/D converter.
  Analog voltage can be divided into 1024 (2^{10}) components at 10-bit resolution.
- Total error: Difference between actual and logical values. This error is the sum of an offset error, gain error, non-linearity error, and an error caused by noise.
- Linearity error: Deviation of the straight line drawn between the zero transition point (00 0000 0000 <-> 00 0000 0001) and the full-scale transition point (11 1111 1110 <-> 11 1111 1111) of the device from actual conversion characteristics
- Differential linearity error: Deviation from the ideal input voltage required to shift output code by one LSB

- **10-bit A/D converter conversion characteristics**



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

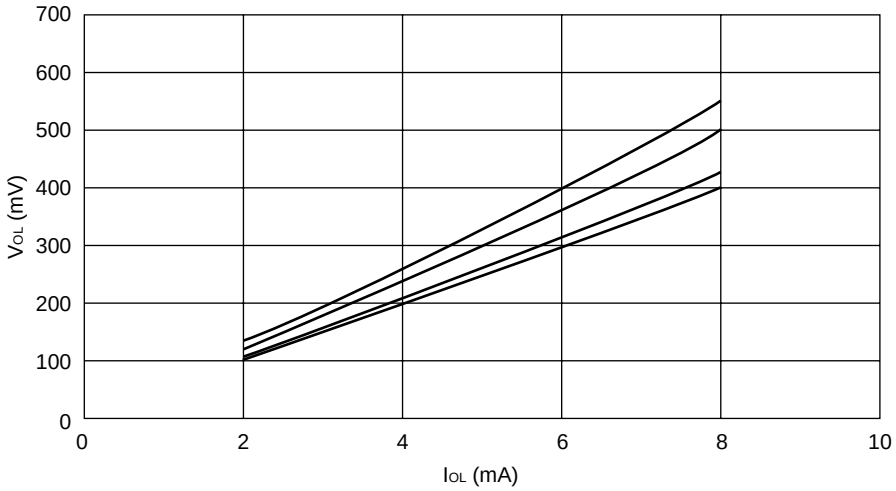
$$\text{Linearity error} = \frac{V_{NT} - (1LSB \times N + V_{OT})}{1LSB} \quad [\text{ LSB }]$$

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

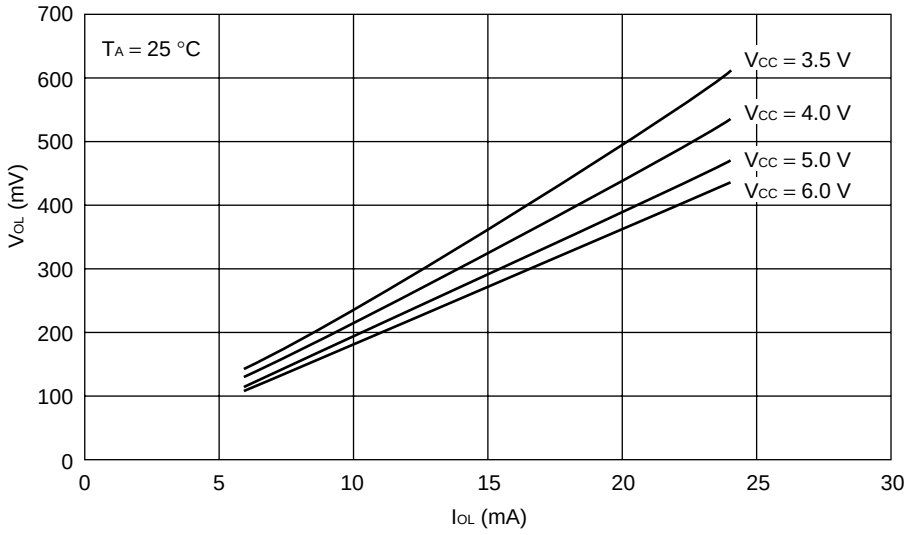
■ EXAMPLE CHARACTERISTICS

1. “L” level output voltage

V_{OL} – I_{OL}
Other than P20 to P27

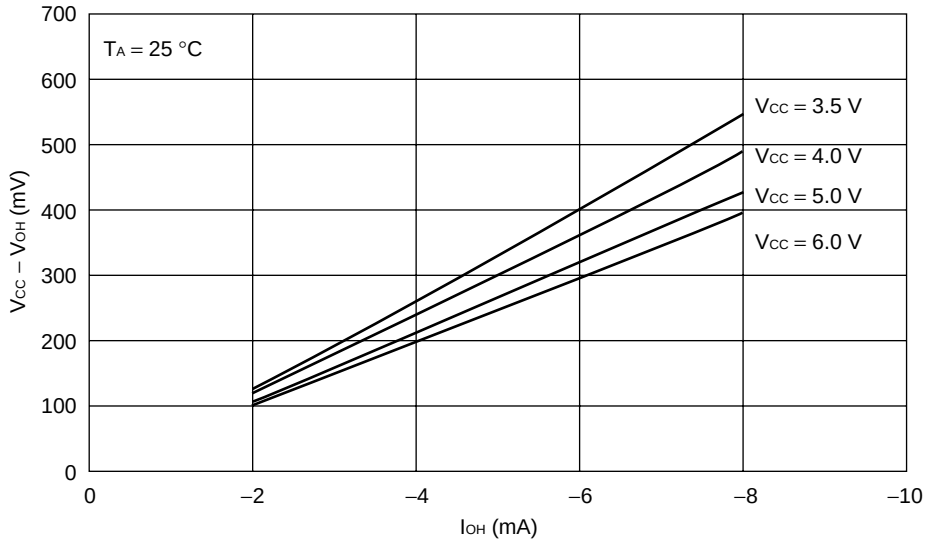


V_{OL} – I_{OL}
P20 to P27



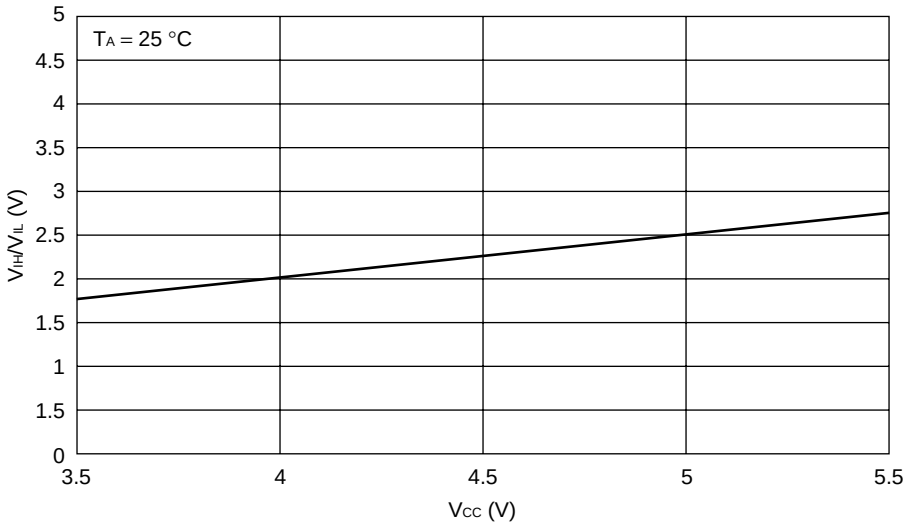
2. “H” level output voltage

$(V_{CC} - V_{OH})$ – I_{OH}
Other than P50 to P55

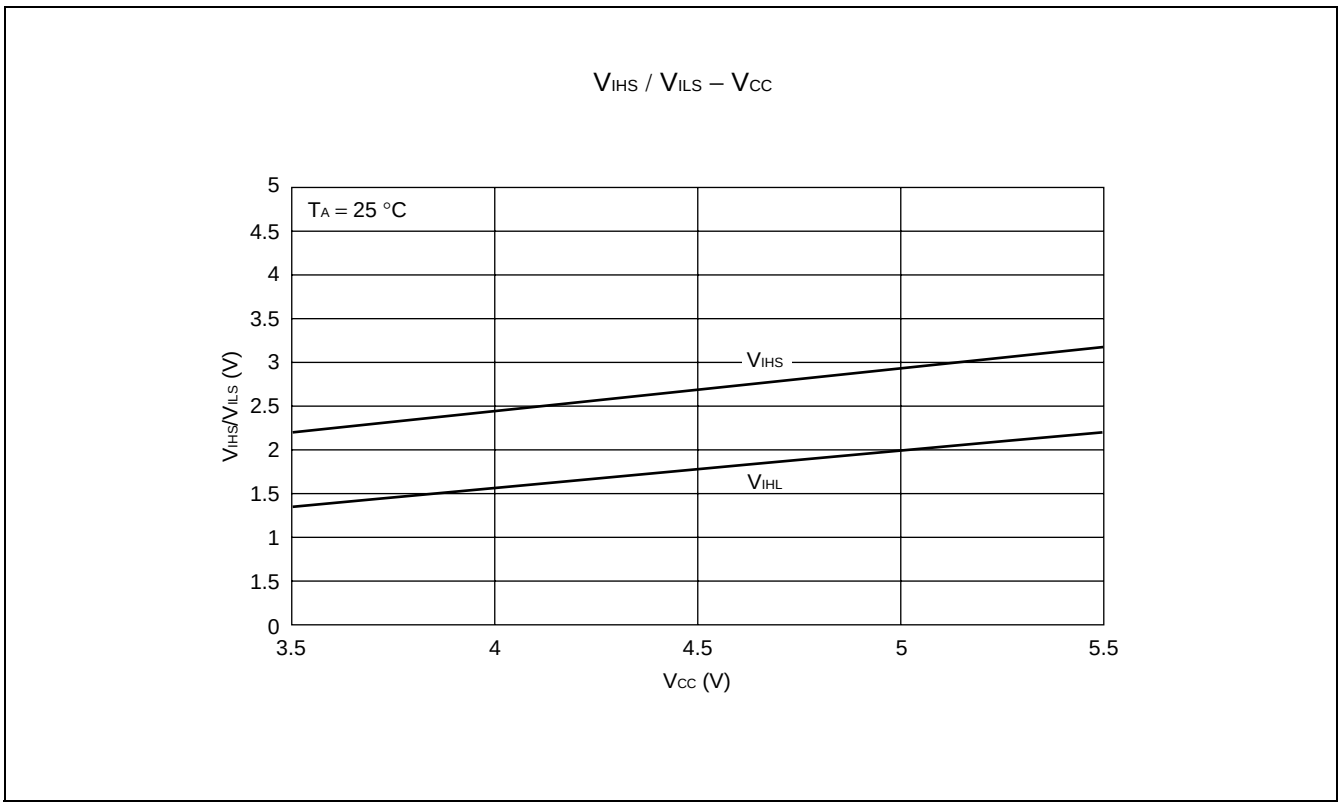


3. “H” level input voltage / “L” level input voltage (CMOS input)

V_{IH} / V_{IL} – V_{CC}



4. “H” level input voltage / “L” level input voltage (CMOS hysteresis input)

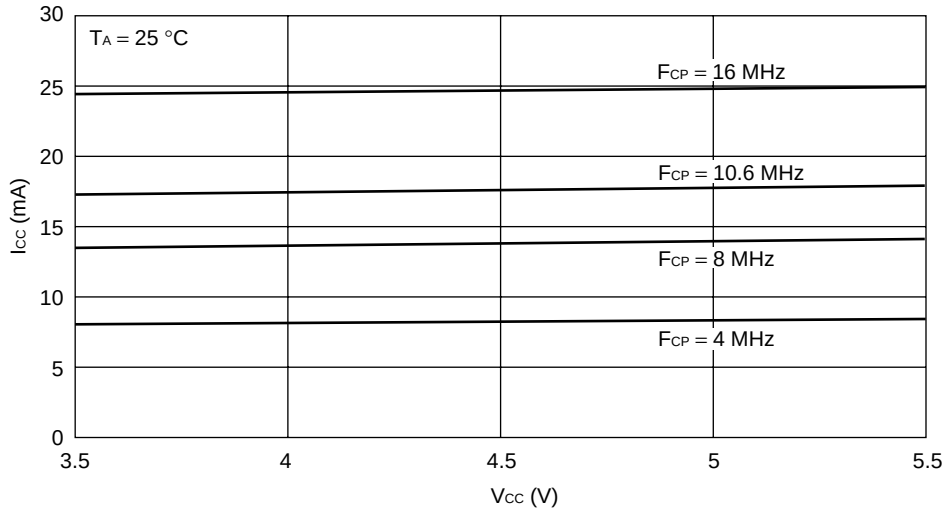


5. Power supply current

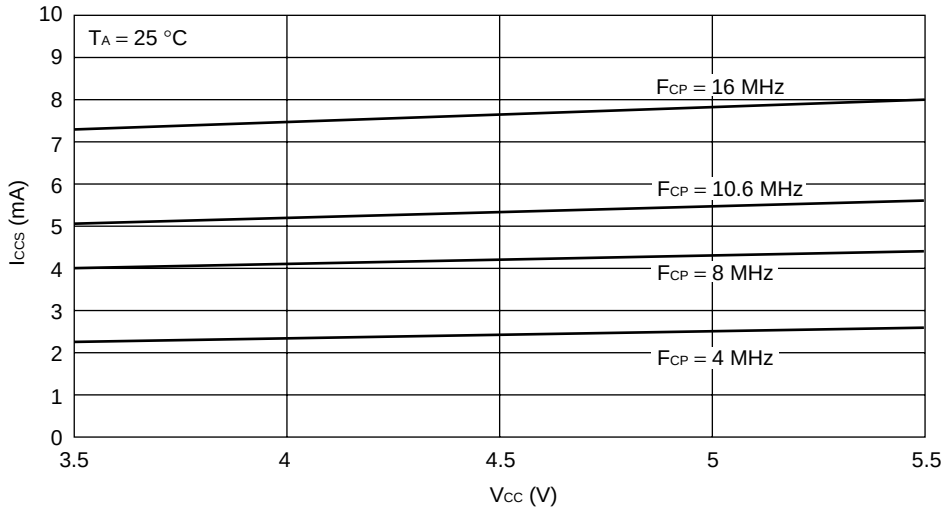
(F_{CP} = internal operating clock frequency)

- MB90552A
- Measurement conditions : External clock mode, ROM read loop operation, without resource operation, Typ. sample, internal operating frequency = 4MHz (external rectangular wave clock at 8MHz), T_A = 25 °C

I_{CC} – V_{CC}

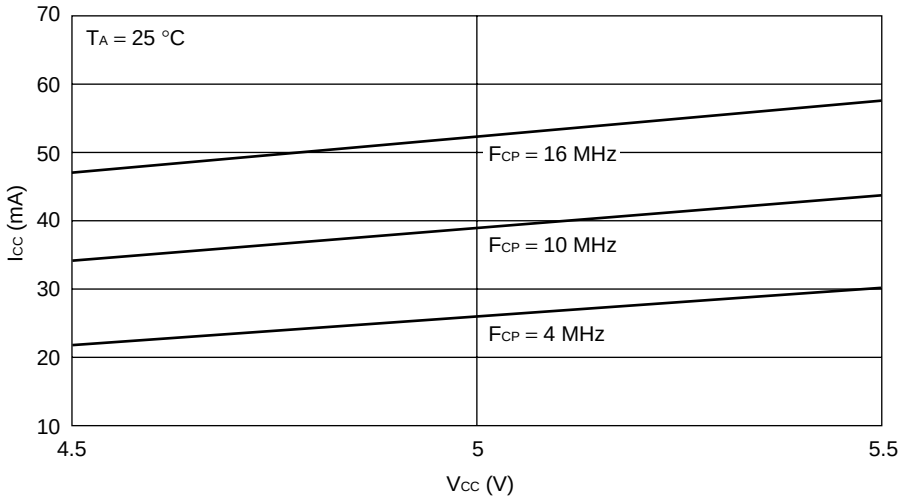


I_{CCS} – V_{CC}

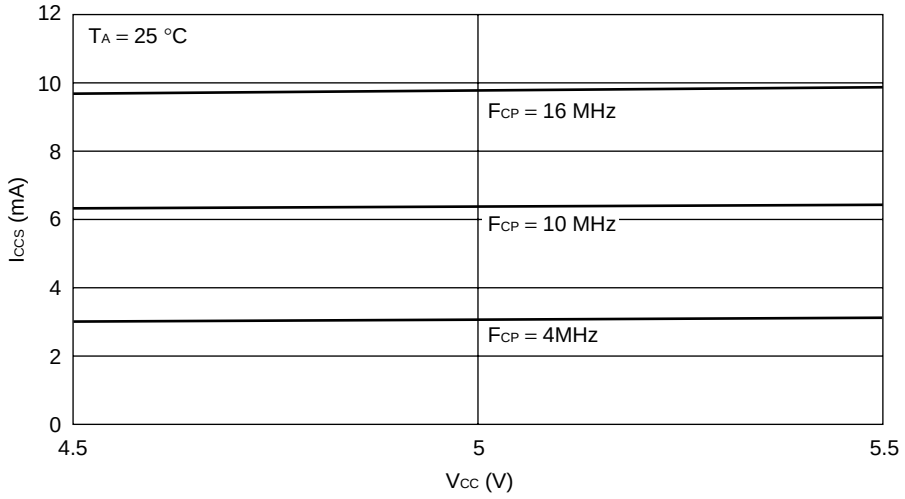


- MB90F553A
- Measurement conditions : External clock mode, ROM read loop operation,
  without resource operation, Typ. sample,
  internal operating frequency = 4MHz (external rectangular wave
  clock at 8MHz), T_A = 25 °C

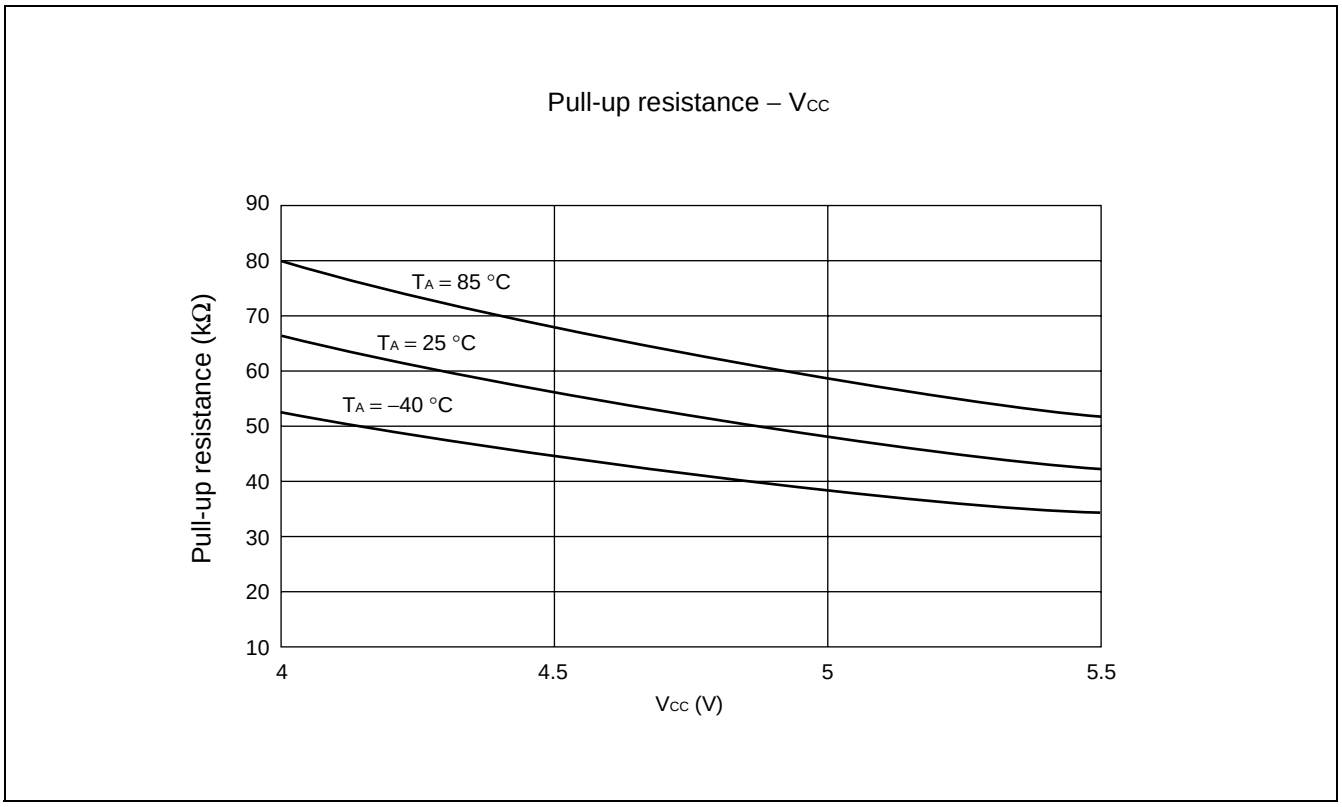
I_{CC} – V_{CC}



I_{CCS} – V_{CC}



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

- **Number of execution cycles**

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

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

Table 2 Explanation of Symbols in Tables of Instructions

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

Table 3 Effective Address Fields

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

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

Table 4 Number of Execution Cycles for Each Type of Addressing

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

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

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

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

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

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

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

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

Table 7 Transfer Instructions (Byte) [41 Instructions]

Mnemonic		#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOV	A, dir	2	3	0	(b)	byte (A) ← (dir)	Z	*	–	–	–	*	*	–	–	–
MOV	A, addr16	3	4	0	(b)	byte (A) ← (addr16)	Z	*	–	–	–	*	*	–	–	–
MOV	A, Ri	1	2	1	0	byte (A) ← (Ri)	Z	*	–	–	–	*	*	–	–	–
MOV	A, ear	2	2	1	0	byte (A) ← (ear)	Z	*	–	–	–	*	*	–	–	–
MOV	A, eam	2+	3+ (a)	0	(b)	byte (A) ← (eam)	Z	*	–	–	–	*	*	–	–	–
MOV	A, io	2	3	0	(b)	byte (A) ← (io)	Z	*	–	–	–	*	*	–	–	–
MOV	A, #imm8	2	2	0	0	byte (A) ← imm8	Z	*	–	–	–	*	*	–	–	–
MOV	A, @A	2	3	0	(b)	byte (A) ← ((A))	Z	–	–	–	–	*	*	–	–	–
MOV	A, @RLi+disp8	3	10	2	(b)	byte (A) ← ((RLi)+disp8)	Z	*	–	–	–	*	*	–	–	–
MOVN	A, #imm4	1	1	0	0	byte (A) ← imm4	Z	*	–	–	–	R	*	–	–	–
MOVX	A, dir	2	3	0	(b)	byte (A) ← (dir)	X	*	–	–	–	*	*	–	–	–
MOVX	A, addr16	3	4	0	(b)	byte (A) ← (addr16)	X	*	–	–	–	*	*	–	–	–
MOVX	A, Ri	2	2	1	0	byte (A) ← (Ri)	X	*	–	–	–	*	*	–	–	–
MOVX	A, ear	2	2	1	0	byte (A) ← (ear)	X	*	–	–	–	*	*	–	–	–
MOVX	A, eam	2+	3+ (a)	0	(b)	byte (A) ← (eam)	X	*	–	–	–	*	*	–	–	–
MOVX	A, io	2	3	0	(b)	byte (A) ← (io)	X	*	–	–	–	*	*	–	–	–
MOVX	A, #imm8	2	2	0	0	byte (A) ← imm8	X	*	–	–	–	*	*	–	–	–
MOVX	A, @A	2	3	0	(b)	byte (A) ← ((A))	X	–	–	–	–	*	*	–	–	–
MOVX	A,@RWi+disp8	2	5	1	(b)	byte (A) ← ((RWi)+disp8)	X	*	–	–	–	*	*	–	–	–
MOVX	A, @RLi+disp8	3	10	2	(b)	byte (A) ← ((RLi)+disp8)	X	*	–	–	–	*	*	–	–	–
MOV	dir, A	2	3	0	(b)	byte (dir) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV	addr16, A	3	4	0	(b)	byte (addr16) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV	Ri, A	1	2	1	0	byte (Ri) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV	ear, A	2	2	1	0	byte (ear) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV	eam, A	2+	3+ (a)	0	(b)	byte (eam) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV	io, A	2	3	0	(b)	byte (io) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV	@RLi+disp8, A	3	10	2	(b)	byte ((RLi) +disp8) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV	Ri, ear	2	3	2	0	byte (Ri) ← (ear)	–	–	–	–	–	*	*	–	–	–
MOV	Ri, eam	2+	4+ (a)	1	(b)	byte (Ri) ← (eam)	–	–	–	–	–	*	*	–	–	–
MOV	ear, Ri	2	4	2	0	byte (ear) ← (Ri)	–	–	–	–	–	*	*	–	–	–
MOV	eam, Ri	2+	5+ (a)	1	(b)	byte (eam) ← (Ri)	–	–	–	–	–	*	*	–	–	–
MOV	Ri, #imm8	2	2	1	0	byte (Ri) ← imm8	–	–	–	–	–	*	*	–	–	–
MOV	io, #imm8	3	5	0	(b)	byte (io) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV	dir, #imm8	3	5	0	(b)	byte (dir) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV	ear, #imm8	3	2	1	0	byte (ear) ← imm8	–	–	–	–	–	*	*	–	–	–
MOV	eam, #imm8	3+	4+ (a)	0	(b)	byte (eam) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV	@AL, AH															
/MOV	@A, T	2	3	0	(b)	byte ((A)) ← (AH)	–	–	–	–	–	*	*	–	–	–
XCH	A, ear	2	4	2	0	byte (A) ↔ (ear)	Z	–	–	–	–	–	–	–	–	–
XCH	A, eam	2+	5+ (a)	0	2× (b)	byte (A) ↔ (eam)	Z	–	–	–	–	–	–	–	–	–
XCH	Ri, ear	2	7	4	0	byte (Ri) ↔ (ear)	–	–	–	–	–	–	–	–	–	–
XCH	Ri, eam	2+	9+ (a)	2	2× (b)	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) +disp8)	–	*	–	–	–	*	*	–	–	–
MOVW A, @RLi+disp8	3	10	2	(c)	word (A) ← ((RLi) +disp8)	–	*	–	–	–	*	*	–	–	–
MOVW dir, A	2	3	0	(c)	word (dir) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW addr16, A	3	4	0	(c)	word (addr16) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW SP, A	1	1	0	0	word (SP) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, A	1	2	1	0	word (RWi) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW ear, A	2	2	1	0	word (ear) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW eam, A	2+	3+ (a)	0	(c)	word (eam) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW io, A	2	3	0	(c)	word (io) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW @RWi+disp8, A	2	5	1	(c)	word ((RWi) +disp8) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW @RLi+disp8, A	3	10	2	(c)	word ((RLi) +disp8) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, ear	2	3	2	(0)	word (RWi) ← (ear)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, eam	2+	4+ (a)	1	(c)	word (RWi) ← (eam)	–	–	–	–	–	*	*	–	–	–
MOVW ear, RWi	2	4	2	0	word (ear) ← (RWi)	–	–	–	–	–	*	*	–	–	–
MOVW eam, RWi	2+	5+ (a)	1	(c)	word (eam) ← (RWi)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, #imm16	3	2	1	0	word (RWi) ← imm16	–	–	–	–	–	*	*	–	–	–
MOVW io, #imm16	4	5	0	(c)	word (io) ← imm16	–	–	–	–	–	–	–	–	–	–
MOVW ear, #imm16	4	2	1	0	word (ear) ← imm16	–	–	–	–	–	*	*	–	–	–
MOVW eam, #imm16	4+	4+ (a)	0	(c)	word (eam) ← imm16	–	–	–	–	–	–	–	–	–	–
MOVW @AL, AH /MOVW@A, T	2	3	0	(c)	word ((A)) ← (AH)	–	–	–	–	–	*	*	–	–	–
XCHW A, ear	2	4	2	0	word (A) ↔ (ear)	–	–	–	–	–	–	–	–	–	–
XCHW A, eam	2+	5+ (a)	0	2× (c)	word (A) ↔ (eam)	–	–	–	–	–	–	–	–	–	–
XCHW RWi, ear	2	7	4	0	word (RWi) ↔ (ear)	–	–	–	–	–	–	–	–	–	–
XCHW RWi, eam	2+	9+ (a)	2	2× (c)	word (RWi) ↔ (eam)	–	–	–	–	–	–	–	–	–	–
MOVL A, ear	2	4	2	0	long (A) ← (ear)	–	–	–	–	–	*	*	–	–	–
MOVL A, eam	2+	5+ (a)	0	(d)	long (A) ← (eam)	–	–	–	–	–	*	*	–	–	–
MOVL A, #imm32	5	3	0	0	long (A) ← imm32	–	–	–	–	–	*	*	–	–	–
MOVL ear, A	2	4	2	0	long (ear) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVL eam, A	2+	5+ (a)	0	(d)	long (eam) ← (A)	–	–	–	–	–	*	*	–	–	–

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 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	–	–	–	–	*	*	*	*	–
ADD A, dir	2	5	0	(b)	byte (A) ← (A) +(dir)	Z	–	–	–	–	*	*	*	*	–
ADD A, ear	2	3	1	0	byte (A) ← (A) +(ear)	Z	–	–	–	–	*	*	*	*	–
ADD A, eam	2+	4+ (a)	0	(b)	byte (A) ← (A) +(eam)	Z	–	–	–	–	*	*	*	*	–
ADD ear, A	2	3	2	0	byte (ear) ← (ear) + (A)	–	–	–	–	–	*	*	*	*	–
ADD eam, A	2+	5+ (a)	0	2× (b)	byte (eam) ← (eam) + (A)	Z	–	–	–	–	*	*	*	*	*
ADDC A	1	2	0	0	byte (A) ← (AH) + (AL) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDC A, ear	2	3	1	0	byte (A) ← (A) + (ear) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDC A, eam	2+	4+ (a)	0	(b)	byte (A) ← (A) + (eam) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDDC A	1	3	0	0	byte (A) ← (AH) + (AL) + (C) (decimal)	Z	–	–	–	–	*	*	*	*	–
SUB A, #imm8	2	2	0	0	byte (A) ← (A) –imm8	Z	–	–	–	–	*	*	*	*	–
SUB A, dir	2	5	0	(b)	byte (A) ← (A) – (dir)	Z	–	–	–	–	*	*	*	*	–
SUB A, ear	2	3	1	0	byte (A) ← (A) – (ear)	Z	–	–	–	–	*	*	*	*	–
SUB A, eam	2+	4+ (a)	0	(b)	byte (A) ← (A) – (eam)	Z	–	–	–	–	*	*	*	*	–
SUB ear, A	2	3	2	0	byte (ear) ← (ear) – (A)	–	–	–	–	–	*	*	*	*	–
SUB eam, A	2+	5+ (a)	0	2× (b)	byte (eam) ← (eam) – (A)	–	–	–	–	–	*	*	*	*	*
SUBC A	1	2	0	0	byte (A) ← (AH) – (AL) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBC A, ear	2	3	1	0	byte (A) ← (A) – (ear) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBC A, eam	2+	4+ (a)	0	(b)	byte (A) ← (A) – (eam) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBDC A	1	3	0	0	byte (A) ← (AH) – (AL) – (C) (decimal)	Z	–	–	–	–	*	*	*	*	–
ADDW A	1	2	0	0	word (A) ← (AH) + (AL)	–	–	–	–	–	*	*	*	*	–
ADDW A, ear	2	3	1	0	word (A) ← (A) +(ear)	–	–	–	–	–	*	*	*	*	–
ADDW A, eam	2+	4+ (a)	0	(c)	word (A) ← (A) +(eam)	–	–	–	–	–	*	*	*	*	–
ADDW A, #imm16	3	2	0	0	word (A) ← (A) +imm16	–	–	–	–	–	*	*	*	*	–
ADDW ear, A	2	3	2	0	word (ear) ← (ear) + (A)	–	–	–	–	–	*	*	*	*	–
ADDW eam, A	2+	5+ (a)	0	2× (c)	word (eam) ← (eam) + (A)	–	–	–	–	–	*	*	*	*	*
ADDCW A, ear	2	3	1	0	word (A) ← (A) + (ear) + (C)	–	–	–	–	–	*	*	*	*	–
ADDCW A, eam	2+	4+ (a)	0	(c)	word (A) ← (A) + (eam) + (C)	–	–	–	–	–	*	*	*	*	–
SUBW A	1	2	0	0	word (A) ← (AH) – (AL)	–	–	–	–	–	*	*	*	*	–
SUBW A, ear	2	3	1	0	word (A) ← (A) – (ear)	–	–	–	–	–	*	*	*	*	–
SUBW A, eam	2+	4+ (a)	0	(c)	word (A) ← (A) – (eam)	–	–	–	–	–	*	*	*	*	–
SUBW A, #imm16	3	2	0	0	word (A) ← (A) –imm16	–	–	–	–	–	*	*	*	*	–
SUBW ear, A	2	3	2	0	word (ear) ← (ear) – (A)	–	–	–	–	–	*	*	*	*	–
SUBW eam, A	2+	5+ (a)	0	2× (c)	word (eam) ← (eam) – (A)	–	–	–	–	–	*	*	*	*	*
SUBCW A, ear	2	3	1	0	word (A) ← (A) – (ear) – (C)	–	–	–	–	–	*	*	*	*	–
SUBCW A, eam	2+	4+ (a)	0	(c)	word (A) ← (A) – (eam) – (C)	–	–	–	–	–	*	*	*	*	–
ADDL A, ear	2	6	2	0	long (A) ← (A) + (ear)	–	–	–	–	–	*	*	*	*	–
ADDL A, eam	2+	7+ (a)	0	(d)	long (A) ← (A) + (eam)	–	–	–	–	–	*	*	*	*	–
ADDL A, #imm32	5	4	0	0	long (A) ← (A) +imm32	–	–	–	–	–	*	*	*	*	–
SUBL A, ear	2	6	2	0	long (A) ← (A) – (ear)	–	–	–	–	–	*	*	*	*	–
SUBL A, eam	2+	7+ (a)	0	(d)	long (A) ← (A) – (eam)	–	–	–	–	–	*	*	*	*	–
SUBL A, #imm32	5	4	0	0	long (A) ← (A) –imm32	–	–	–	–	–	*	*	*	*	–

Note : For an explanation of “(a)” to “(d)”, refer to Table 4, “Number of Execution Cycles for Each Type of Addressing,” and Table 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 (ear)	–	–	–	–	–	–	–	*	*	–
MULU	A	1	*8	0	0	byte (AH) *byte (AL) → word (A)	–	–	–	–	–	–	–	–	–	–
MULU	A, ear	2	*9	1	0	byte (A) *byte (ear) → word (A)	–	–	–	–	–	–	–	–	–	–
MULU	A, eam	2+	*10	0	(b)	byte (A) *byte (eam) → word (A)	–	–	–	–	–	–	–	–	–	–
MULUW	A	1	*11	0	0	word (AH) *word (AL) → long (A)	–	–	–	–	–	–	–	–	–	–
MULUW	A, ear	2	*12	1	0	word (A) *word (ear) → long (A)	–	–	–	–	–	–	–	–	–	–
MULUW	A, eam	2+	*13	0	(c)	word (A) *word (eam) → long (A)	–	–	–	–	–	–	–	–	–	–

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

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

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

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

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

Notes:
- When overflow occurs during DIV or DIVW instruction execution, the number of execution cycles takes two values because of detection before and after an operation.
- When overflow occurs during DIV or DIVW instruction execution, the contents of AL are destroyed.
- For (a) to (d), refer to “Table 4 Number of Execution Cycles for Effective Address in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Table 19 Branch 1 Instructions [31 Instructions]

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

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

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

Table 20 Branch 2 Instructions [19 Instructions]

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

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

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

Table 21 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 22 Bit Manipulation Instructions [21 Instructions]

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

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

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

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

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

Table 24 String Instructions [10 Instructions]

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

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

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

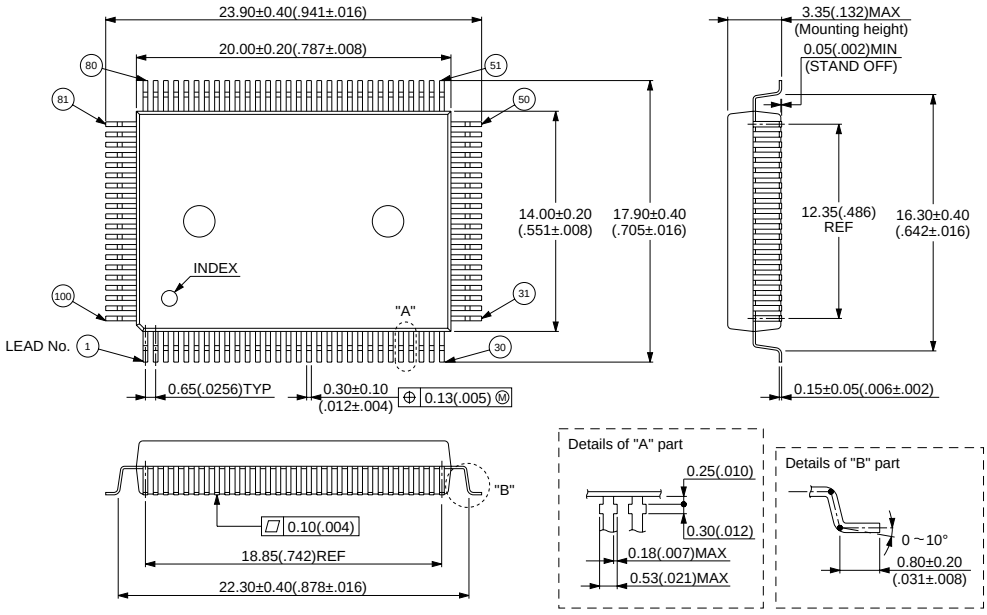
MB90550A Series

■ ORDERING INFORMATION

Part number	Package	Remarks
MB90552APF MB90553APF MB90T552APF MB90T553APF MB90F553APF MB90P553APF	100-pin plastic QFP (FPT-100P-M06)	
MB90552APF MB90553APF MB90T552APF MB90T553APF MB90F553APF MB90P553APF	100-pin plastic LQFP (FPT-100P-M05)	

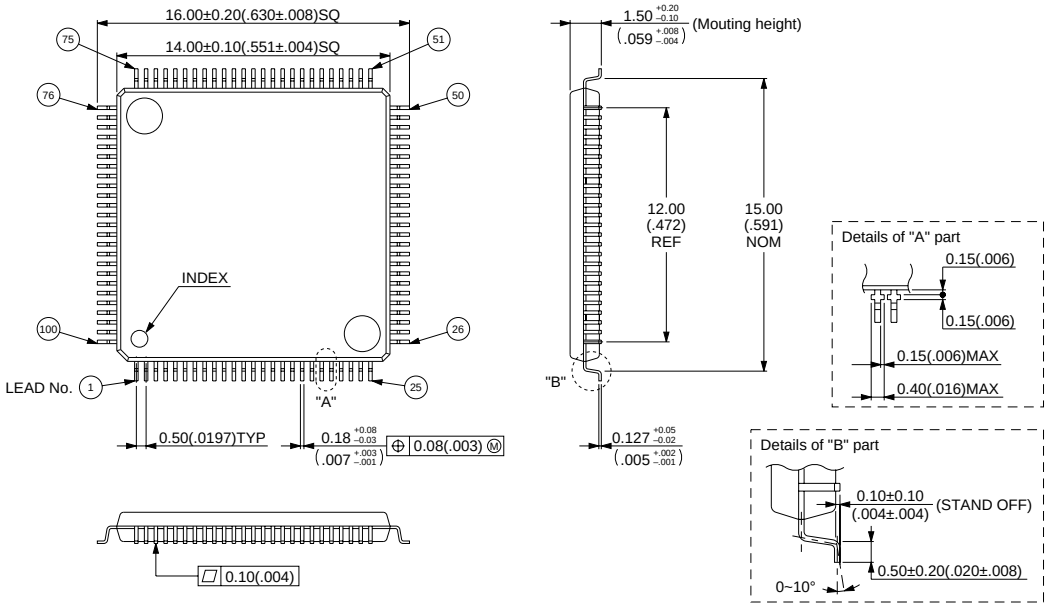
■ PACKAGE DIMENSIONS

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



Dimensions in mm (inches)

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



Dimensions in mm (inches)

MB90550A Series

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: 81(44) 754-3763
Fax: 81(44) 754-3329

http://www.fujitsu.co.jp/

North and South America
FUJITSU MICROELECTRONICS, INC.
Semiconductor Division
3545 North First Street
San Jose, CA 95134-1804, USA
Tel: (408) 922-9000
Fax: (408) 922-9179

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

http://www.fujitsumicro.com/

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

http://www.fujitsu-ede.com/

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

http://www.fmap.com.sg/

F9910

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

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

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

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