

# 16-bit Proprietary Microcontroller

CMOS

## F<sup>2</sup>MC-16F MB90F243H

## MB90F243H

### ■ DESCRIPTION

The MB90F243H is a 16-bit microcontroller optimized for applications in mechatronics such as HDD units. The architecture of the MB90F243 is based on the MB90242A, and embedded with a 128-Kbyte flash memory.

The instruction set is based on the AT architecture of the F<sup>2</sup>MC-16 and 16H family, with additional high-level language supporting instruction, expanded addressing modes, enhanced multiplication and division instructions, and improved bit processing instructions. In addition, long-word data can now be processed due to the inclusion of a 32-bit accumulator.

The MB90F243H includes a variety of peripherals on chip, such as the device is equipped with 6-channel 8/10-bit A/D converter, UART, 3-channel 16-bit reload timers, 1-channel 16-bit timer, 4-channel 16-bit input capture and 4-channel DTP/external interrupts.

\* : F<sup>2</sup>MC stands for FUJITSU Flexible Microcontroller.

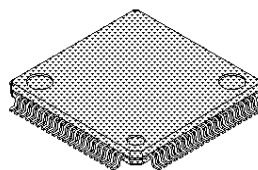
### ■ FEATURES

- Minimum execution time: 50.0 ns at 40 MHz oscillation
- Instruction set optimized for controller applications
  - Variety of data types: bit, byte, word, long-word
  - Expanded addressing modes: 25 types
  - High coding efficiency
  - Improvement of high-precision arithmetic operations through use of 32-bit accumulator
  - Enhanced multiplication and division instructions (signed arithmetic operations)

*(Continued)*

### ■ PACKAGE

80-pin Plastic TQFP



(FPT-80P-M15)

# MB90F243H

(Continued)

- Instruction set supports high-level language (C language) and multitasking
  - Inclusion of system stack pointer
  - Variety of pointers
  - High instruction set symmetry
  - Barrel shift instruction
  - Stack clock function
- Improved execution speed: 8-byte queue
- Powerful interrupt functions
  - Interrupt processing time: 0.8  $\mu$ s at 40 MHz oscillation
  - Priority levels: 8 levels (programmable)
  - External interrupt inputs: 4 channels
- Automatic transfer function independent of CPU
  - Extended intelligent I/O Service: max. 15 channels
- 128-Kbyte flash memory
  - Access time (min.) : 120 ns
  - Sector structure of 16K + 512  $\times$  2 + 7K + 8K + 32K + 64K
  - Program/erase operations from both programmers and CPUs through built-in flash memory interface circuit
  - Built-in program booster
- Internal RAM: 1 Kbyte
  - According to mode settings, data stored on RAM can be executed as CPU instructions.
- General-purpose ports: max. 62 channels (single-chip mode)  
max. 38 channels (external bus mode)
- 18-bit timebase timer
- Watchdog timer
- UART: 8 bits  $\times$  1 channel
- 8/16-bit I/O simple serial interface (max. 10 Mbps): 1 channel
- 8/10-bit A/D converter: analog inputs: 6 channels
  - Resolution: 10 bits (switchable to 8 bits)
  - Conversion time: min. 1.0  $\mu$ s
  - Conversion result store register: 4 channels
- 16-bit free-run timer: 1 channel (operating clock: 0.2  $\mu$ s)
- 16-bit input capture: 4 channels
- 16-bit reload timer: 3 channels
- Low-power consumption modes
  - Sleep mode
  - Stop mode
  - Hardware standby mode
- Package: TQFP-80
- CMOS technology

# MB90F243H

## ■ PRODUCT LINEUP

Item		Part number	MB90F243H	MB90F243	MB90242A	MB90V241
Classification			Flash memory version		External ROM product	For evaluation
CPU core	ROM size		Flash memory 128 Kbytes		None	
	RAM size		1 Kbyte		2 Kbytes	4 Kbytes
	Number of instructions		412 instructions			
	Minimum execution time		50.0 ns at 40 MHz	62.5 ns at 32 MHz		
	Product-sum operation unit		None		On chip	
	Low-power consumption modes		Sleep, stop, hardware standby			
	DTP/external interrupts		Interrupt sources: 23 channels/ external interrupt inputs: 4 channels			
Peripherals	Ports		Output ports (N-channel open-drain): 6 I/O ports (CMOS): 56 Total: 62		Output ports (N-channel open-drain): 6 I/O ports (CMOS): 32 Total: 38	
	Timebase timer		18 bits × 1 channel			
	UART		8 bits × 1 channel			
	8/10-bit A/D converter		8/10-bit resolution × 6 channels			
	8/16-bit I/O simple serial interface		8/16 bits × 1 channel			
	16-bit free-run timer		16 bits × 1 channel			
	16-bit input capture		16 bits × 4 channels			
	16-bit reload timer		3 channels		2 channels	3 channels
Watchdog timer function		On chip				
Characteristics	Power supply voltage*		4.5 V to 5.5 V			
	Operating temperature		0°C to +70°C	-25°C to +85°C	-30°C to +70°C	0°C to +70°C
	System clock frequency		40 MHz (5.0 V ±10%)	32 MHz (5.0 V ±10%)		
Process			CMOS			

\* : Varies with conditions such as the operating frequency. (See section “■ Electrical Characteristics.”)

# MB90F243H

## ■ PACKAGE AND CORRESPONDING PRODUCTS

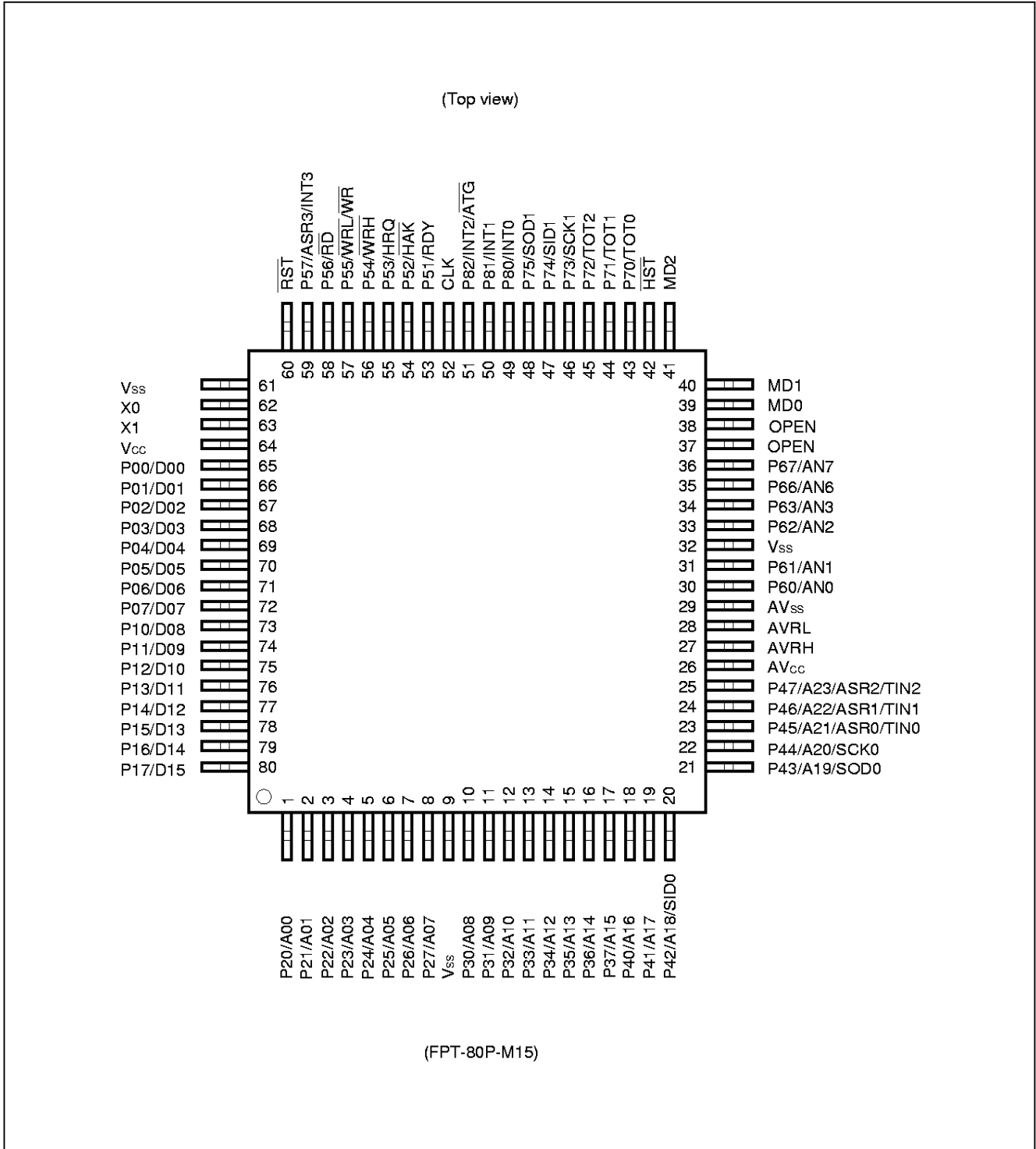
Package	MB90F243H	MB90F243	MB90242A
FPT-80P-M05	×	○	○
FPT-80P-M15	○	○	×

○ : Available    × : Not available

Note: For more information about each package, see section “■ Package Dimensions.”

# MB90F243H

## ■ PIN ASSIGNMENT



# MB90F243H

## ■ PIN DESCRIPTION

Pin no. TQFP-80*	Pin name	Circuit type	Function
62	X0	A	Crystal oscillator pins (40 MHz)
63	X1		
39 to 41	MD0 to MD2	C	Operating mode selection input pins Connect directly to $V_{CC}$ or $V_{SS}$ . In the flash memory mode, these pins are set to be $V_{IO}$ (= 12.0 V) input pins by performing a proper operation.
60	$\overline{RST}$	B	External reset request input pin
42	$\overline{HST}$	D	Hardware standby input pin
65 to 72	P00 to P07	E	General-purpose I/O port
	D00 to D07		I/O pins for the lower 8 bits of the external data bus
73 to 80	P10 to P17	E	General-purpose I/O port This function is valid when the external bus 8-bit mode.
	D08 to D17		I/O pins for the upper 8 bits of the external data bus This function is valid when 16-bit bus mode.
1 to 8	P20 to P27	F	General-purpose I/O port
	A00 to A07		Output pins for the medium 8 bits of the external address bus
10 to 17	P30 to P37	F	General-purpose I/O port This function is valid when the corresponding bit of the middle address control register specification is "port".
	A08 to A15		Output pins for the medium 8 bits of the external address bus This function is valid when the corresponding bit of the middle address control register specification is "port".
18	P40	F	General-purpose I/O port This function is valid when the corresponding bit of the upper address control register specification is "port".
	A16		External address bus output pin of the bit 16 This function is valid when the corresponding bit of the upper address control register specification is "address".
19	P41	F	General-purpose I/O port This function is valid when the upper address control register specification is "port".
	A17		External address bus output pin of the bit 17 This function is valid when the corresponding bit of the upper address control register specification is "address".

\*: FPT-80P-M15

(Continued)

# MB90F243H

Pin no. TQFP-80*	Pin name	Circuit type	Function
20	P42	F	General-purpose I/O port This function is valid when the corresponding bit of the upper address control register specification is "port".
	A18		External address bus output pin of the bit 18 This function is valid when the corresponding bit of the upper address control register specification is "address".
	SID0		UART #0 data input pin During UART #0 input operations, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately.
21	P43	G	General-purpose I/O port This function is valid when the UART #0 data output is disabled and the corresponding bit of the upper address control register specification is "port".
	A19		External address bus output pin of the bit 19 This function is valid when the UART #0 data output is disabled and the corresponding bit of the upper address control register specification is "address".
	SOD0		UART #0 data output pin This function is valid when the UART #0 data output is enabled.
22	P44	G	General-purpose I/O port This function is valid when the UART #0 and SSI #2 clock output are disabled and the corresponding bit of the upper address control register specification is "port".
	A20		External address bus output pin of the bit 20 This function is valid when the UART #0 clock output is disabled and the corresponding bit of the upper address control register specification is "address".
	SCK0		UART #0 clock I/O pin
23	P45	G	General-purpose I/O port This function is valid when the SSI #2 data output is disabled and the corresponding bit of the upper address control register specification is "port".
	A21		External address bus output pin of the bit 21 This function is valid when the SSI #2 data output is disabled and the corresponding bit of the upper address control register specification is "address".
	ASR0		16-bit input capture #0 data input pin During 16-bit input capture #0 input operations, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately.
	TIN0		16-bit timer #0 data input pin During 16-bit timer #0 input operations, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately.

\*: FPT-80P-M15

(Continued)

# MB90F243H

Pin no. TQFP-80*	Pin name	Circuit type	Function
24	P46	G	General-purpose I/O port This function is valid when the corresponding bit of the upper address control register specification is "port".
	A22		External address bus output pin of the bit 22 This function is valid when the corresponding bit of the upper address control register specification is "address".
	ASR1		16-bit input capture #1 data input pin During 16-bit input capture #1 input operations, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately.
	TIN1		16-bit timer #1 data input pin During 16-bit timer #1 input operations, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately.
25	P47	G	General-purpose I/O port This function is valid when the corresponding bit of the upper address control register specification is "port".
	A23		External address bus output pin for the bit 23 This function is valid when the corresponding bit of the upper address control register specification is "address".
	ASR2		16-bit input capture #2 data input pin During 16-bit input capture #2 input operations, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately.
	TIN2		16-bit timer #2 data input pin During 16-bit timer #2 input operations, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately.
53	P51	H	General-purpose I/O port This function is valid when the ready function is disabled.
	RDY		Ready input pin This function is valid when the ready function is enabled.
54	P52	H	General-purpose I/O port This function is valid when the hold function is disabled.
	HAK		Hold acknowledge output pin This function is valid when the hold function is enabled.
55	P53	H	General-purpose I/O port This function is valid when the hold function is disabled.
	HRQ		Hold request input pin This function is valid and when the hold function is enabled.

\*: FPT-80P-M15

(Continued)



# MB90F243H

Pin no. TQFP-80*	Pin name	Circuit type	Function
56	P54	F	General-purpose I/O port This function is valid in external bus eight-bit mode, or when $\overline{WRH}$ pin output is disabled.
	$\overline{WRH}$		Write strobe output pin for the upper eight bits of the data bus This function is valid in modes where the external bus 16-bit mode is enabled, and $\overline{WRH}$ pin output is enabled.
57	P55	F	General-purpose I/O port This function is valid when $\overline{WRL}$ pin output is disabled.
	$\overline{WRL} / \overline{WR}$		Write strobe output pin for the lower eight bits of the data bus This function is valid $\overline{WRL}$ pin output is enabled.
58	P56	F	General-purpose I/O port
	$\overline{RD}$		Read strobe output pin for the data bus
59	P57	F	General-purpose I/O port
	ASR3		16-bit input capture #3 data input pin During 16-bit input capture #3 input operations, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately.
	INT3		DTP/external interrupt #3 data input pin During DTP/external interrupt #3 input operations, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately.
30, 31, 33, 34, 35, 36	P60, P61, P62, P63, P66, P67	I	N-ch open-drain type I/O ports When bits corresponding to the ADER are set to "0", reading instructions other than the read-modify-write group returns the pin level. The value written on the data register is output to this pin directly.
	AN0, AN1, AN2, AN3, AN6, AN7		8/10-bit A/D converter analog input pins Use this function after setting bits corresponding to the ADER to "1" and setting corresponding bits of the data register to "1".
43 to 45	P70 to P72	G	General-purpose I/O port This function is valid when the reload timer #0, #1, and #2 output is disabled.
	TOT0 to TOT2		16-bit timer output pins This function is valid when the 16-bit timer #0, #1, and #2 output is enabled.
46	P73	G	General-purpose I/O port This function is valid when the SSI #1 clock output is disabled.
	SCK1		SSI #1 clock output I/O pin

\*: FPT-80P-M15

(Continued)

# MB90F243H

Pin no. TQFP-80*	Pin name	Circuit type	Function
47	P74	G	General-purpose I/O port This function is always valid.
	SID1		SSI #1 data input pin During SSI #1 input operations, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately.
48	P75	G	General-purpose I/O port This function is valid when the SSI #1 data output is disabled.
	SOD1		SSI #1 data output pin This function is valid when the SSI #1 data output is disabled.
49, 50	P80, P81	G	General-purpose I/O port This function is always valid.
	INT0, INT1		DTP/external interrupt input pin When external interrupts are enabled, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately.
51	P82	G	General-purpose I/O port This function is always valid.
	INT2		DTP/external interrupt input pin When external interrupts are enabled, these inputs may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using them for output deliberately. Because an input to this pin is clamped to Low when the CPU stops, use INT0 or INT1 to wake up the system from the stop mode.
	$\overline{\text{ATG}}$		8/10-bit A/D converter trigger input pin When 8/10-bit A/D converter is waiting for activation, this input may be used at any time; therefore, it is necessary to stop output by other functions on this pin, except when using it for output deliberately.
37, 38	OPEN	—	Open pins No internal connections are made.
52	CLK	G	CLK output pin
64	V <sub>cc</sub>	Power supply	Digital circuit power supply pin
9, 32, 61	V <sub>ss</sub>	Power supply	Digital circuit power supply (GND) pin
26	AV <sub>cc</sub>	Power supply	Analog circuit power supply pin This power supply must only be turned on or off when electric potential of AV <sub>cc</sub> or greater is applied to V <sub>cc</sub> .

\*: FPT-80P-M15

(Continued)

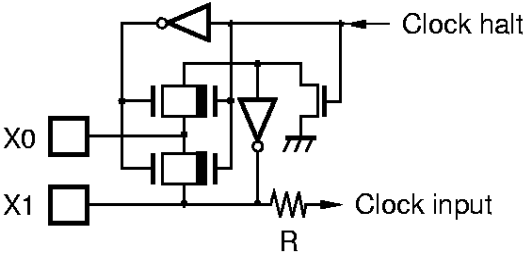
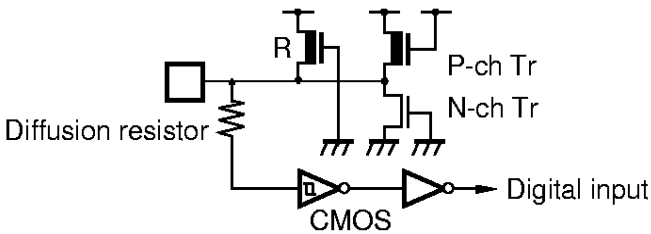
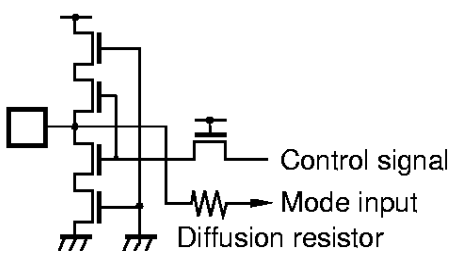
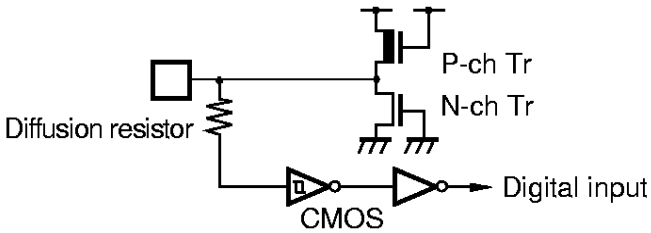
**MB90F243H***(Continued)*

Pin no. TQFP-80*	Pin name	Circuit type	Function
27	AVRH	Power supply	8/10-bit A/D converter external reference voltage input pin This pin must only be turned on or off when electric potential of AVRH or greater is applied to AV <sub>CC</sub> .
28	AVRL	Power supply	8/10-bit A/D converter external reference voltage input pin
29	AV <sub>SS</sub>	Power supply	Analog circuit power supply (GND) pin

\*: FPT-80P-M15

# MB90F243H

## ■ I/O CIRCUIT TYPE

Type	Circuit	Remarks
A		<ul style="list-style-type: none"> <li>• 40 MHz</li> <li>• Oscillation feedback resistor: approximately 1 MΩ</li> </ul>
B		<ul style="list-style-type: none"> <li>• CMOS-level hysteresis input Without standby control</li> <li>• Pull-up resistor: approximately 50 KΩ</li> </ul>
C		<ul style="list-style-type: none"> <li>• CMOS-level input</li> <li>• High voltage control for flash memory testing</li> </ul>
D		<ul style="list-style-type: none"> <li>• CMOS-level hysteresis input (Without standby control)</li> <li>• Optional pull-up resistor</li> </ul>

(Continued)

# MB90F243H

(Continued)

Type	Circuit	Remarks
E		<ul style="list-style-type: none"> <li>• CMOS-level output</li> <li>• TTL-level input (With standby control)</li> <li>• Optional pull-up resistor</li> </ul>
F		<ul style="list-style-type: none"> <li>• CMOS-level output</li> <li>• CMOS-level hysteresis input</li> <li>• TTL-level input (flash memory mode) (With standby control)</li> <li>• Optional pull-up resistor</li> </ul>
G		<ul style="list-style-type: none"> <li>• CMOS-level output</li> <li>• CMOS-level hysteresis input (With standby control)</li> <li>• Optional pull-up resistor</li> </ul>
H		<ul style="list-style-type: none"> <li>• CMOS-level output</li> <li>• TTL-level input (With standby control)</li> <li>• Optional pull-up resistor</li> </ul>
I		<ul style="list-style-type: none"> <li>• N-ch open-drain CMOS-level output</li> <li>• CMOS-level hysteresis input (Analog input) (With analog input control)</li> <li>• Optional pull-up resistor</li> </ul>

# MB90F243H

## ■ HANDLING DEVICES

### 1. Preventing Latchup

Latchup may occur on CMOS ICs if voltage higher than  $V_{CC}$  or lower than  $V_{SS}$  is applied to the input or output pins other than medium-and high-voltage pins or if higher than the voltage which shown on “■ Absolute Maximum Ratings” is applied between  $V_{CC}$  and  $V_{SS}$ .

When latchup occurs, power supply current increases rapidly might thermally damage elements. When using, take great care not to exceed the absolute maximum ratings.

In addition, for the same reasons take care to prevent the analog power supply from exceeding the digital power supply.

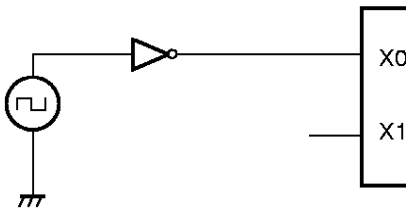
### 2. Treatment of Unused Pins

Leaving unused input pins open could cause malfunctions. They should be connected to a pull-up or pull-down resistors.

### 3. Precautions when Using an External Clock

When an external clock is used, drive X0 only.

- For example an external clock



### 4. Power Supply Pins

When there are several  $V_{CC}$  and  $V_{SS}$  pins, those pins that should have the same electric potential are connected within the device when the device is designed in order to prevent misoperation, such as latch-up. However, all of those pins must be connected to the power supply and ground externally in order to reduce unnecessary emissions, prevent misoperation of strobe signals due to an increase in the ground level, and to observe the total output current standards.

In addition, give a due consideration to the connection in that current supply be connected to  $V_{CC}$  and  $V_{SS}$  with the lowest possible impedance.

Finally, it is recommended to connect a capacitor of about  $0.1 \mu\text{F}$  between  $V_{CC}$  and  $V_{SS}$  near this device as a bypass capacitor.

## 5. Crystal Oscillation Circuit

Noise in the vicinity of the X0 and X1 pins will cause this device to operate incorrectly. Design the printed circuit board so that the bypass capacitor connecting X0 and X1 pins and the crystal oscillator (or ceramic oscillator) to ground is located as close to the device as possible.

In addition, because printed circuit board artwork in which the area around the X0 and X1 pins is surrounded by ground provides stable operation, such an arrangement is strongly recommended.

## 6. Sequence for Applying the A/D Converter Power Supply and the Analog Inputs

Always be sure to apply the digital power supply ( $V_{CC}$ ) before applying the A/D converter power supply ( $AV_{CC}$ ,  $AVRH$ , and  $AVRL$ ) and the analog inputs (AN0 to AN7).

In addition, when the power is turned off, turn off the A/D converter power supply and the analog inputs first, and then turn off the digital power supply. (Turning on or off the analog and digital power supplies simultaneously will not cause any problems.)

Whether applying or cutting off the power, be certain that  $AVRH$  does not exceed  $AV_{CC}$ .

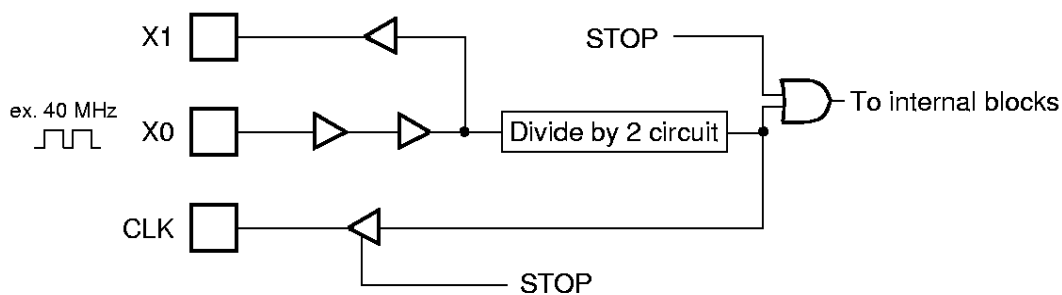
## 7. External Reset Input

To reliably reset the controller by inputting an "L" level to the  $\overline{RST}$  pin, ensure that the "L" level is applied for at least five machine cycles.

## 8. $\overline{HST}$ Pin

When turning on the system, be sure to set the  $\overline{HST}$  pin to "H" level. Never set the  $\overline{HST}$  pin to "L" level while the  $\overline{RST}$  pin is in "L" level.

## 9. CLK Pin

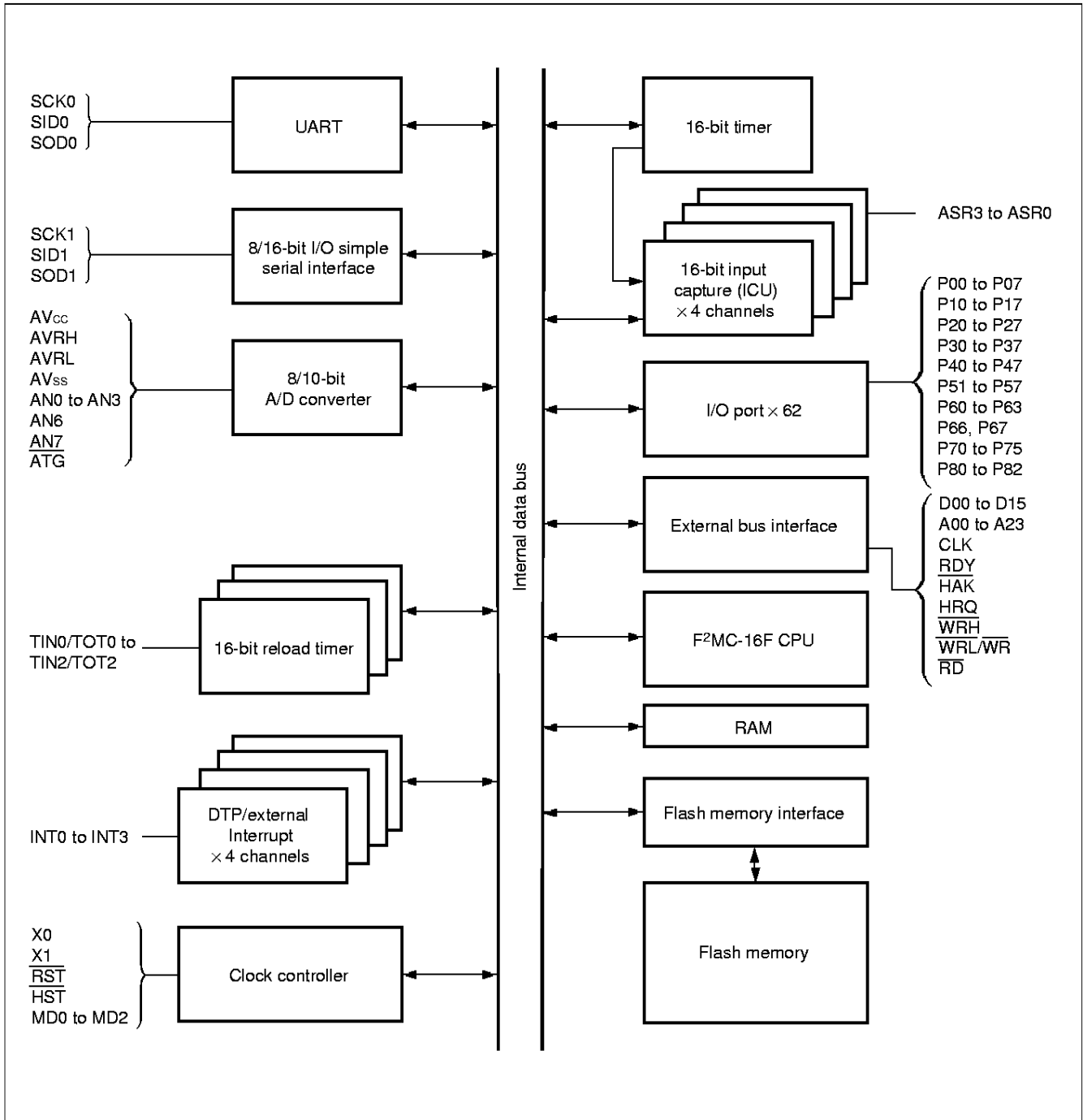


Note: CLK pin cannot use as I/O port.

Care must be taken that this is different from standard specification for F<sup>2</sup>MC-16F family.

# MB90F243H

## ■ BLOCK DIAGRAM

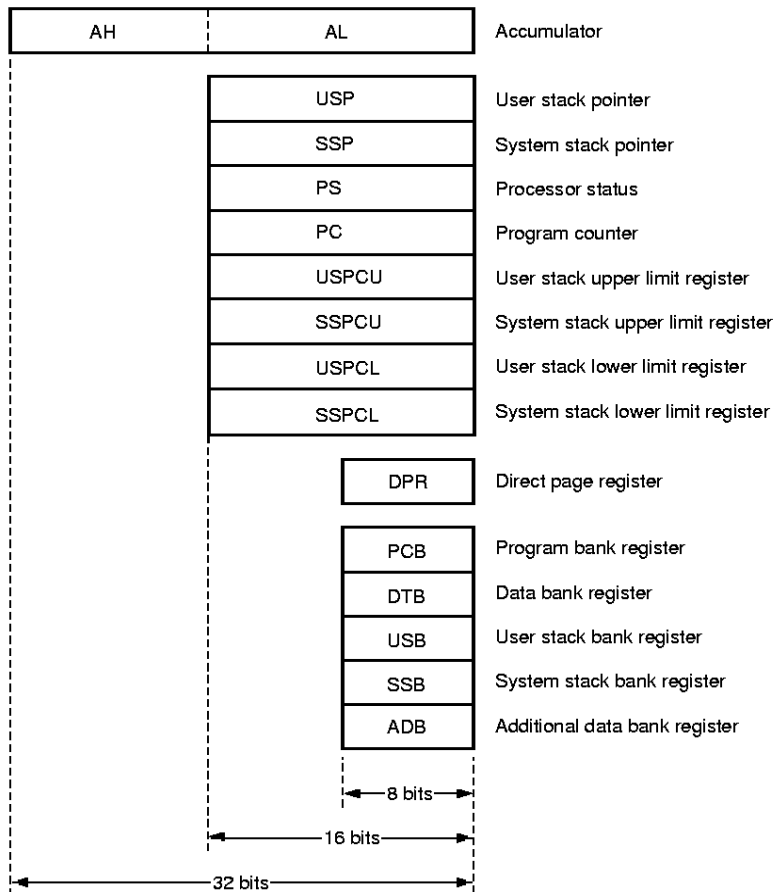




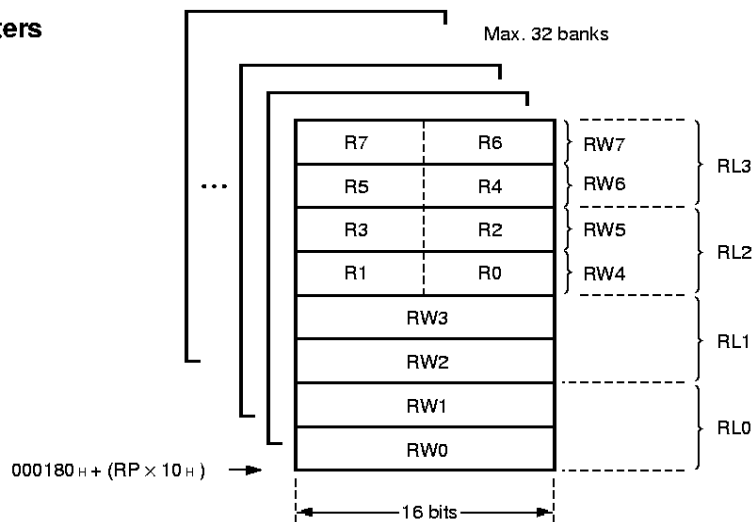
# MB90F243H

## ■ F<sup>2</sup>MC-16L CPU PROGRAMMING MODEL

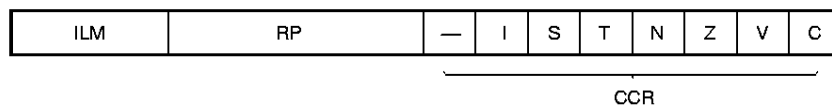
### • Dedicated registers



### • General-purpose registers

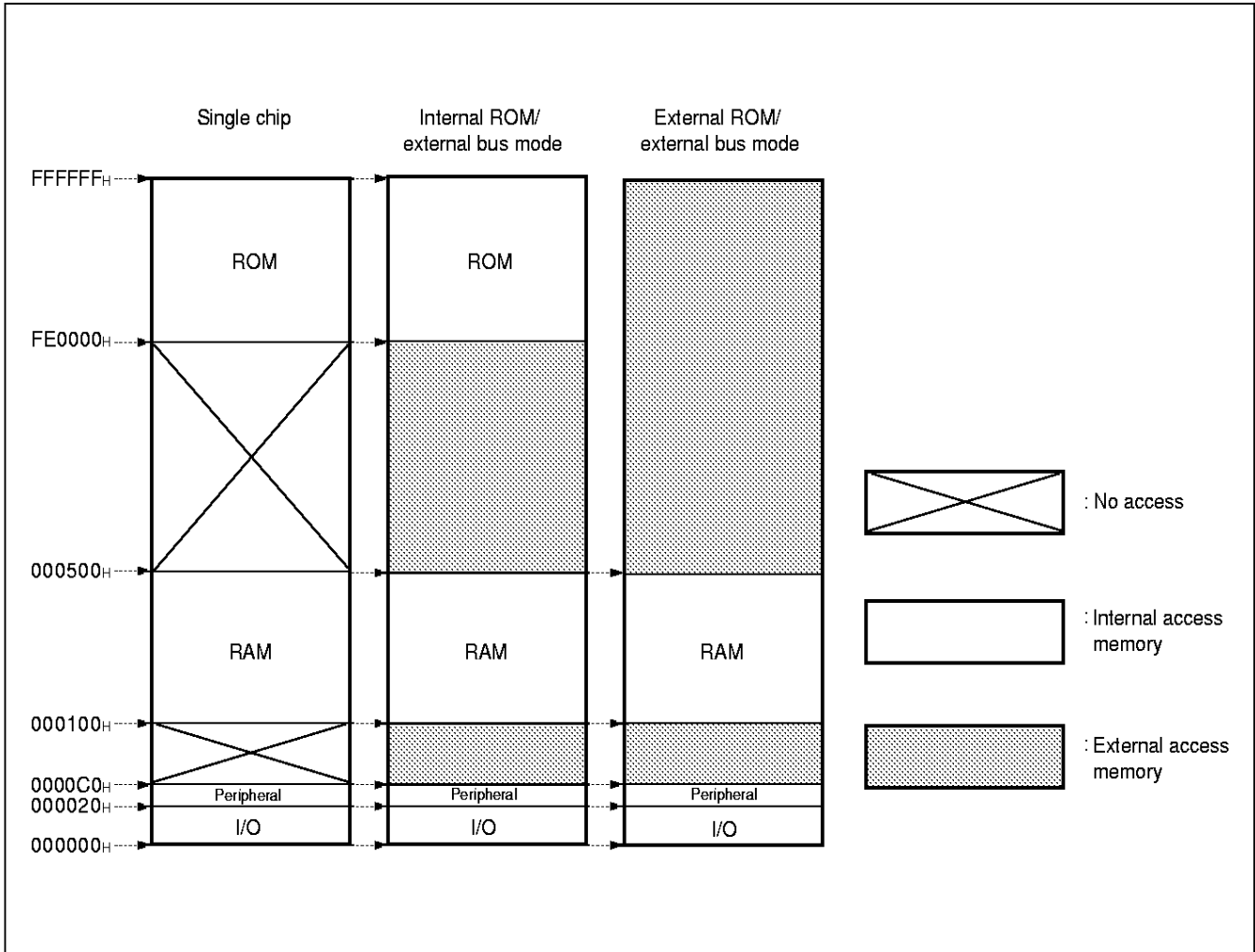


### • Processor status (PS)



# MB90F243H

## MEMORY MAP



# MB90F243H

## ■ I/O MAP

Address	Register name	Register	Read/write	Resource name	Initial value
00000H	PDR0	Port 0 data register	R/W	Port 0	XXXXXXXX
00001H	PDR1	Port 1 data register	R/W	Port 1	XXXXXXXX
00002H	PDR2	Port 2 data register	R/W	Port 2	XXXXXXXX
00003H	PDR3	Port 3 data register	R/W	Port 3	XXXXXXXX
00004H	PDR4	Port 4 data register	R/W	Port 4	XXXXXXXX
00005H	PDR5	Port 5 data register	R/W	Port 5	XXXXXX-
00006H	PDR6	Port 6 data register	R/W	Port 6	11--1111
00007H	PDR7	Port 7 data register	R/W	Port 7	--XXXXXX
00008H	PDR8	Port 8 data register	R/W	Port 8	-----XXX
00009H to 0000FH	(Vacancy)				
00010H	DDR0	Port 0 direction register	R/W	Port 0	00000000
00011H	DDR1	Port 1 data direction register	R/W	Port 1	00000000
00012H	DDR2	Port 2 direction register	R/W	Port 2	00000000
00013H	DDR3	Port 3 direction register	R/W	Port 3	00000000
00014H	DDR4	Port 4 data direction register	R/W	Port 4	00000000
00015H	DDR5	Port 5 data direction register	R/W	Port 5	0000000-
00016H	ADER	Analog input enable register	R/W	Analog input enabled	11--1111
00017H	DDR7	Port 7 data direction register	R/W	Port 7	--000000
00018H	DDR8	Port 8 data direction register	R/W	Port 8	-----000
00019H to 0001FH	(Vacancy)				
00020H	SCR1	Serial control status register 1	R/W	8/16-bit I/O simple serial interface ch. 1	10000000
00021H	SSR1	Serial status register 1	R/W		-----00
00022H	SDR1L	Serial data register 1 (L)	R/W		XXXXXXXX
00023H	SDR1H	Serial data register 1 (H)	R/W		XXXXXXXX
00024H to 00027H	(Vacancy)				
00028H	UMC0	Mode control register 0	R/W	UART ch. 0	0000100
00029H	USR0	Status register 0	R/W		00010000
0002AH	UIDR0/ UODR0	Input data register 0/ output data register 0	R/W		XXXXXXXX
0002BH	URD0	Rate and data register 0	R/W		00000000
0002CH to 0002FH	(Vacancy)				

(Continued)

# MB90F243H

Address	Register name	Register	Read/write	Resource name	Initial value
000030H	ENIR	DTP/interrupt enable register	R/W	DTP/external interrupt	----0000
000031H	EIRR	DTP/interrupt source register	R/W		----0000
000032H	ELVR	Request level setting register	R/W		00000000
000033H to 00003FH	(Vacancy)				
000040H	TMCSR0	Timer control status register #0	R/W	16-bit timer #0	00000000
000041H			R/W		XXXX0000
000042H	TMR0	16-bit timer register #0	R		XXXXXXXX
000043H			R		XXXXXXXX
000044H	TMRLR0	16-bit reload register #0	W		XXXXXXXX
000045H			W		XXXXXXXX
000046H 000047H	(Vacancy)				
000048H	TMCSR1	Timer control status register #1	R/W	16-bit timer #1	00000000
000049H			R/W		XXXX0000
00004AH	TMR1	16-bit timer register #1	R		XXXXXXXX
00004BH			R		XXXXXXXX
00004CH	TMRLR1	16-bit reload register #1	W		XXXXXXXX
00004DH			W		XXXXXXXX
00004EH 00004FH	(Vacancy)				
000050H	TMCSR2	Timer control status register #2	R/W	16-bit timer #2	00000000
000051H			R/W		XXXX0000
000052H	TMR2	16-bit timer register #2	R		XXXXXXXX
000053H			R		XXXXXXXX
000054H	TMRLR2	16-bit reload register #2	W		XXXXXXXX
000055H			W		XXXXXXXX
000056H to 00005FH	(Vacancy)				
000060H	ICP0	Input capture register 0	R	16-bit input capture 0 and 1	XXXXXXXX
000061H			R		XXXXXXXX
000062H	ICP1	Input capture register 1	R		XXXXXXXX
000063H			R		XXXXXXXX
000064H	ICS0	Input capture control status register 0 and 1	R/W	00000000	
000065H	(Vacancy)				

(Continued)

# MB90F243H

Address	Register name	Register	Read/write	Resource name	Initial value
000066 <sub>H</sub>	ICP2	Input capture register 2	R	16-bit input capture 2 and 3	XXXXXXXX
000067 <sub>H</sub>			R		XXXXXXXX
000068 <sub>H</sub>	ICP3	Input capture register 3	R		XXXXXXXX
000069 <sub>H</sub>			R		XXXXXXXX
00006A <sub>H</sub>	ICS1	Input capture control status register 2 and 3	R/W		
00006B <sub>H</sub>	(Vacancy)				
00006C <sub>H</sub>	TCDT	Timer data register	R/W	16-bit free-run timer	00000000
00006D <sub>H</sub>			R/W		00000000
00006E <sub>H</sub>	TCCS	Timer control status register	R/W		00000000
00006F <sub>H</sub>	(Vacancy)				
000070 <sub>H</sub>	ADCS1	A/D converter control register 1	R/W	8/10-bit A/D converter	000-0000
000071 <sub>H</sub>	ADCS2	A/D converter control register 2	R/W		-000--00
000072 <sub>H</sub>	ADCT1	Conversion time setting register 1	R/W		XXXXXXXX
000073 <sub>H</sub>	ADCT2	Conversion time setting register 2	R/W		XXXXXXXX
000074 <sub>H</sub>	ADTL0	A/D data register 0 (L)	R		XXXXXXXX
000075 <sub>H</sub>	ADTH0	A/D data register 0 (H)	R		-----XX
000076 <sub>H</sub>	ADTL1	A/D data register 1 (L)	R		XXXXXXXX
000077 <sub>H</sub>	ADTH1	A/D data register 1 (H)	R		-----XX
000078 <sub>H</sub>	ADTL2	A/D data register 2 (L)	R		XXXXXXXX
000079 <sub>H</sub>	ADTH2	A/D data register 2 (H)	R		-----XX
00007A <sub>H</sub>	ADTL3	A/D data register 3 (L)	R		XXXXXXXX
00007B <sub>H</sub>	ADTH3	A/D data register 3 (H)	R		-----XX
00007C <sub>H</sub> to 00008F <sub>H</sub>	(Vacancy)				
000090 <sub>H</sub> to 00009E <sub>H</sub>	(System reserved area)*1				
00009F <sub>H</sub>	DIRR	Delayed interrupt source generation/ release register	R/W	Delayed interrupt generation module	-----0
0000A0 <sub>H</sub>	STBYC	Standby control register	R/W	Low-power consumption mode	0001XXXX
0000A3 <sub>H</sub>	MACR	Middle address control register	W	External pin	*2
0000A4 <sub>H</sub>	HACR	Upper address control register	W		*2
0000A5 <sub>H</sub>	EPCR	External pin control register	W		*2
0000A8 <sub>H</sub>	WTC	Watchdog timer control register	R/W	Watchdog timer	XXXXXXXX
0000A9 <sub>H</sub>	TBTC	Timebase timer control register	R/W	Timebase timer	0XX00000

(Continued)

# MB90F243H

(Continued)

Address	Register name	Register	Read/write	Resource name	Initial value
0000AE <sub>H</sub>	FMCS	Control status register	R/W	Flash memory	000X0--0
0000B0 <sub>H</sub>	ICR00	Interrupt control register 00	R/W <sup>*3</sup>	Interrupt controller	00000111
0000B1 <sub>H</sub>	ICR01	Interrupt control register 01	R/W <sup>*3</sup>		00000111
0000B2 <sub>H</sub>	ICR02	Interrupt control register 02	R/W <sup>*3</sup>		00000111
0000B3 <sub>H</sub>	ICR03	Interrupt control register 03	R/W <sup>*3</sup>		00000111
0000B4 <sub>H</sub>	ICR04	Interrupt control register 04	R/W <sup>*3</sup>		00000111
0000B5 <sub>H</sub>	ICR05	Interrupt control register 05	R/W <sup>*3</sup>		00000111
0000B6 <sub>H</sub>	ICR06	Interrupt control register 06	R/W <sup>*3</sup>		00000111
0000B7 <sub>H</sub>	ICR07	Interrupt control register 07	R/W <sup>*3</sup>		00000111
0000B8 <sub>H</sub>	ICR08	Interrupt control register 08	R/W <sup>*3</sup>		00000111
0000B9 <sub>H</sub>	ICR09	Interrupt control register 09	R/W <sup>*3</sup>		00000111
0000BA <sub>H</sub>	ICR10	Interrupt control register 10	R/W <sup>*3</sup>		00000111
0000BB <sub>H</sub>	ICR11	Interrupt control register 11	R/W <sup>*3</sup>		00000111
0000BC <sub>H</sub>	ICR12	Interrupt control register 12	R/W <sup>*3</sup>		00000111
0000BD <sub>H</sub>	ICR13	Interrupt control register 13	R/W <sup>*3</sup>		00000111
0000BE <sub>H</sub>	ICR14	Interrupt control register 14	R/W <sup>*3</sup>		00000111
0000BF <sub>H</sub>	ICR15	Interrupt control register 15	R/W <sup>*3</sup>		00000111
0000C0 <sub>H</sub> to 0000FF <sub>H</sub>	(External area) <sup>*3</sup>				

Explanation of read/write

R/W : Readable and writable  
 R : Read only  
 W : Write only

Explanation of initial values

0 : The initial value of this bit is "0".  
 1 : The initial value of this bit is "1".  
 X : The initial value of this bit is undefined.  
 - : This bit is unused. No initial value is defined.

\*1: Access prohibited.

\*2: The initial values are changed depending on a bus mode.

\*3: The only area available for the external access below address 0000FF<sub>H</sub> is this area. Addresses not explained in the table are "(reserved area)"; accesses to these areas are handled accesses to internal areas. No access signal is generated for the external bus.

Note: Do not use any "(Vacancy)".

# MB90F243H

## ■ ELECTRICAL CHARACTERISTICS

### 1. Absolute Maximum Ratings

( $V_{SS} = V_{SS} = 0.0\text{ V}$ )

Parameter	Symbol	Value		Unit	Remarks
		Min.	Max.		
Power supply voltage	$V_{CC}$	$V_{SS} - 0.3$	$V_{SS} + 7.0$	V	
	$AV_{CC}$	$V_{CC} - 0.3$	$V_{CC} + 7.0$	V	
	$AV_{RH}$	$V_{SS} - 0.3$	$V_{SS} + 7.0$	V	*1
	$AV_{RL}$	$V_{SS} - 0.3$	$V_{SS} + 7.0$	V	
Input voltage	$V_I$	$V_{SS} - 0.3$	$V_{CC} + 0.3$	V	*2
Output voltage	$V_O$	$V_{SS} - 0.3$	$V_{CC} + 0.3$	V	*2
"L" level maximum output current	$I_{OL}$	—	15	mA	
"L" level average output current	$I_{OLAV}$	—	4	mA	
"L" level total maximum output current	$\Sigma I_{OL}$	—	100	mA	
"L" level total average output current	$\Sigma I_{OLAV}$	—	50	mA	
"H" level maximum output current	$I_{OH}$	—	-15	mA	
"H" level average output current	$I_{OHAV}$	—	-4	mA	
"H" level total maximum output current	$\Sigma I_{OH}$	—	-100	mA	
"H" level total average output current	$\Sigma I_{OHAV}$	—	-50	mA	
Power consumption	$P_D$	—	+600	mW	
Operating temperature	$T_A$	0	+70	°C	
Storage temperature	$T_{stg}$	-55	+125	°C	

\*1:  $AV_{CC}$ ,  $AV_{RH}$  and  $AV_{RL}$  must not exceed  $V_{CC}$ .

\*2:  $V_I$  and  $V_O$  must not exceed  $V_{CC} + 0.3\text{ V}$ .

**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.

# MB90F243H

## 2. Recommended Operating Conditions

( $V_{SS} = V_{SS} = 0.0\text{ V}$ )

Parameter	Symbol	Value		Unit	Remarks
		Min.	Max.		
Power supply voltage	$V_{CC}$	4.5	5.5	V	Normal operation
	$V_{CC}$	3.0	5.5	V	Maintaining the stop status
“H” level input voltage	$V_{IH1}$	$0.7 V_{CC}$	$V_{CC} + 0.3$	V	CMOS input
	$V_{IH2}$	2.2	$V_{CC} + 0.3$	V	TTL input
	$V_{IH1S}$	$0.8 V_{CC}$	$V_{CC} + 0.3$	V	Hysteresis input
	$V_{IHM}$	$V_{SS} - 0.3$	$V_{CC} + 0.3$	V	MD0 to MD2
“L” level input voltage	$V_{IL1}$	$V_{SS} - 0.3$	$0.3 V_{CC}$	V	CMOS input
	$V_{IL2}$	$V_{SS} - 0.3$	0.8	V	TTL input
	$V_{IL1S}$	$V_{SS} - 0.3$	$0.2 V_{CC}$	V	Hysteresis input
	$V_{ILM}$	$V_{SS} - 0.3$	$V_{SS} + 0.3$	V	MD0 to MD2
Operating temperature	$T_A$	0	+70	°C	

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

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

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



# MB90F243H

## 3. DC Characteristics

( $AV_{CC} = V_{CC} = 5.0\text{ V} \pm 10\%$ ,  $AV_{SS} = V_{SS} = 0.0\text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
V <sub>ID</sub> input voltage	V <sub>ID</sub>	—	—	11.5	—	12.5	V	
“H” level output voltage	V <sub>OH</sub>	All ports except port 6	V <sub>CC</sub> = 4.5 V I <sub>OH</sub> = -4.0 mA	V <sub>CC</sub> - 0.5	—	—	V	
“L” level output voltage	V <sub>OL</sub>	All ports	V <sub>CC</sub> = 4.5 V I <sub>OL</sub> = 4.0 mA	—	—	0.4	V	
“H” level input current	I <sub>IH1</sub>	Except $\overline{\text{RST}}$	V <sub>CC</sub> = 5.5 V V <sub>IH</sub> = 0.7 V <sub>CC</sub>	—	—	-10	μA	CMOS input
	I <sub>IH2</sub>	—	V <sub>CC</sub> = 5.5 V V <sub>IH</sub> = 2.2 V <sub>CC</sub>	—	—	-10	μA	TTL input
	I <sub>IH3</sub>	—	V <sub>CC</sub> = 5.5 V V <sub>IH</sub> = 0.8 V <sub>CC</sub>	—	—	-10	μA	Hysteresis input
“L” level input current	I <sub>IL1</sub>	Except $\overline{\text{RST}}$	V <sub>CC</sub> = 5.5 V V <sub>IH</sub> = 0.3 V <sub>CC</sub>	—	—	10	μA	CMOS input
	I <sub>IL2</sub>	—	V <sub>CC</sub> = 5.5 V V <sub>IH</sub> = 0.8 V <sub>CC</sub>	—	—	10	μA	TTL input
	I <sub>IL3</sub>	—	V <sub>CC</sub> = 5.5 V V <sub>IH</sub> = 0.2 V <sub>CC</sub>	—	—	10	μA	Hysteresis input
Pull-up resistor	R <sub>PULL</sub>	$\overline{\text{RST}}$	—	22	—	110	KΩ	
Power supply current*1	I <sub>CC1</sub>	V <sub>CC</sub>	CPU normal mode internal 20 MHz operation	—	—	130	mA	Flash memory read state
	I <sub>CC2</sub>	V <sub>CC</sub>		—	—	150	mA	Flash memory program/erase state
	I <sub>CCS</sub>	V <sub>CC</sub>	CPU sleep mode internal 20 MHz operation	—	—	40	mA	
	I <sub>CCH</sub>	V <sub>CC</sub>	CPU stop mode T <sub>A</sub> = +25°C	—	—	100	μA	
Input capacitance	C <sub>IN</sub>	Other than V <sub>CC</sub> , V <sub>SS</sub>	—	10	—	pF		
Open-drain output leakage current	I <sub>LEAK</sub>	Port 6	—	—	10	μA		
Low V <sub>CC</sub> voltage*2	V <sub>LKO</sub>	—	—	3.2	—	4.2	V	

\*1: Because the current values are tentative values, they are subject to change without notice due to our efforts to improve the characteristics of these devices.

\*2: To prevent improper commands from being activated during rise and fall of V<sub>CC</sub>, the internal V<sub>CC</sub> detection circuit of the flash memory allows only read accesses and ignores write accesses while V<sub>CC</sub> is lower than V<sub>LKO</sub>.

# MB90F243H

## 4. Flash Memory Program/Erase Characteristics

( $A_{V_{CC}} = V_{CC} = 5.0 \text{ V} \pm 10\%$ ,  $A_{V_{SS}} = V_{SS} = 0.0 \text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

Parameter	Condition	Value			Unit	Remarks
		Min.	Typ.	Max.		
Sector erase time	$T_A = +25^\circ\text{C}$ , $V_{CC} = 5.0 \text{ V}$ , 100 cycles	—	1.5	13.5	sec	Except for the write time before internal erase operation
Chip erase time		—	—	27.0	sec	Except for the write time before internal erase operation
Byte program time	—	—	16	1000*	$\mu\text{s}$	Except for the over head time of the system
Chip program time	$T_A = +25^\circ\text{C}$ , $V_{CC} = 5.0 \text{ V}$ , 100 cycles	—	2.1	12.5	sec	Except for the over head time of the system
Erase/program cycle	—	100	—	—	cycles	

\* : The internal automatic algorithm continues operations for up to 48 ms, for each 1-byte writing operation.

# MB90F243H

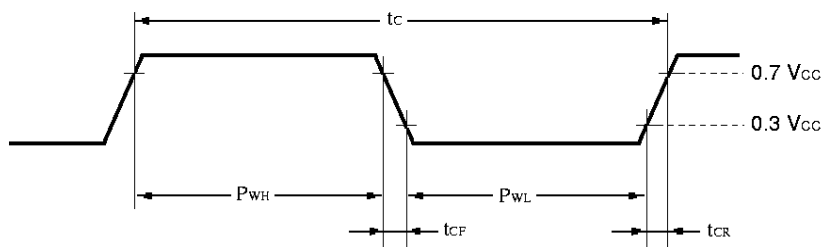
## 5. AC Characteristics

### (1) Clock Timing

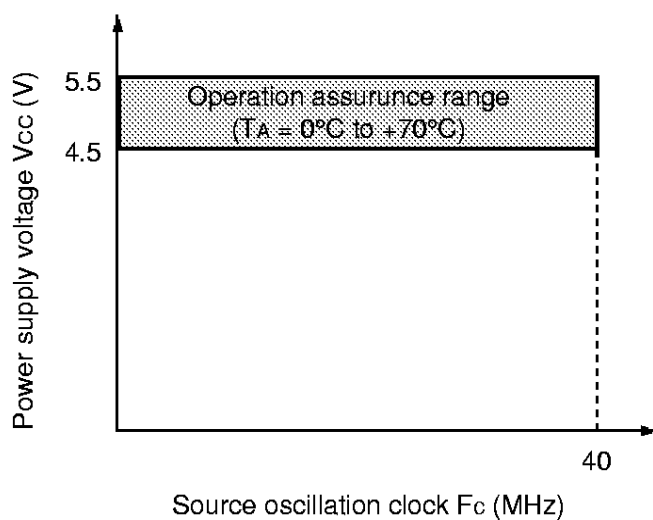
( $A_{V_{SS}} = V_{SS} = 0.0\text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Clock frequency	$F_c$	X0, X1	—	—	40	MHz	
Clock cycle time	$t_c$	X0, X1		$1/F_c$	—	ns	
Input clock pulse width	$P_{WH}$ , $P_{WL}$	X0		10	—	ns	
Input clock rising/falling time	$t_{CR}$ , $t_{CF}$	X0		—	8	ns	

#### • Clock timing



#### • Relationship between clock frequency and power supply voltage



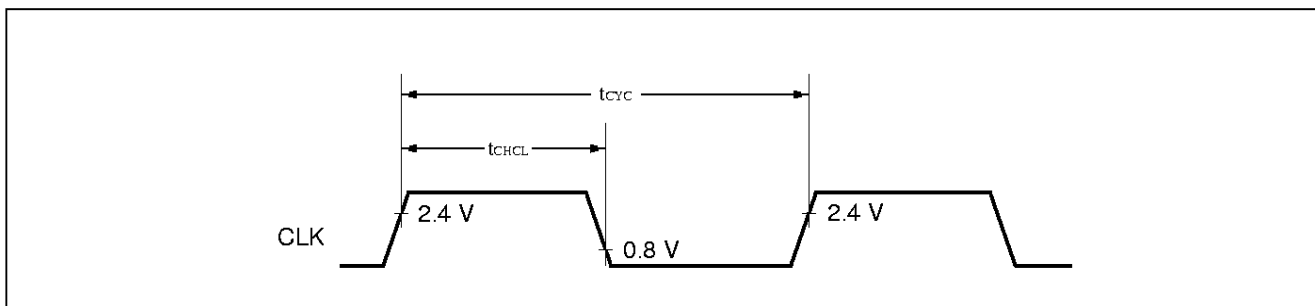
# MB90F243H

## (2) Clock Output Timing

( $AV_{CC} = V_{CC} = 5.0 \text{ V} \pm 10\%$ ,  $AV_{SS} = V_{SS} = 0.0 \text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Cycle time	$t_{CYC}$	CLK	—	$2 t_c^*$	—	ns	
CLK $\uparrow \rightarrow$ CLK $\downarrow$	$t_{CHCL}$	CLK		$1 t_{CYC}/2 - 20$	$1 t_{CYC}/2 + 20$	ns	

\* : For information on  $t_c$  (clock cycle time), see "(1) Clock Timing."



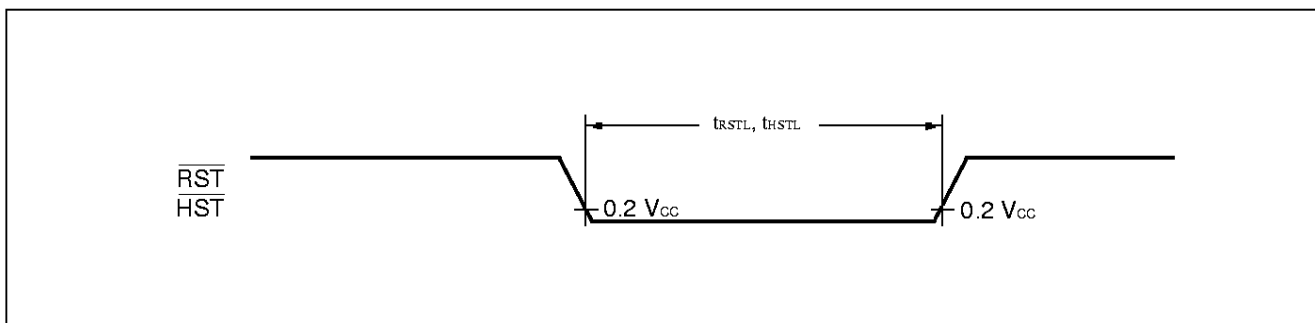
## (3) Reset and Hardware Standby Input

( $AV_{CC} = V_{CC} = 5.0 \text{ V} \pm 10\%$ ,  $AV_{SS} = V_{SS} = 0.0 \text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Reset input time	$t_{RSTL}$	$\overline{\text{RST}}$	—	$5 t_{CYC}^*$	—	ns	
Hardware standby input time	$t_{HSTL}$	$\overline{\text{HST}}$		$5 t_{CYC}^*$	—	ns	

\* : For information on  $t_{CYC}$  (cycle time), see "(2) Clock Output Timing."

Note: When hardware standby input is given, the machine cycle is simultaneously selected to be divide-by-32.



# MB90F243H

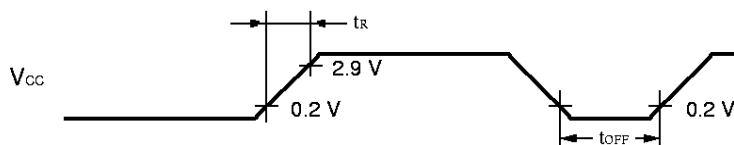
## (4) Power on Supply Specifications (Power-on Reset)

( $V_{SS} = V_{SS} = 0.0 \text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

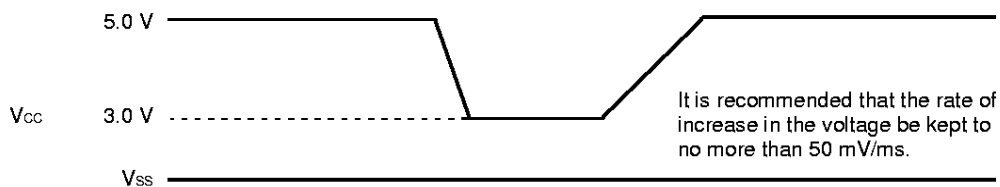
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Power supply rising time	$t_{\text{R}}$	$V_{\text{CC}}$	—	—	30	ms	*
Power supply cut-off time	$t_{\text{OFF}}$	$V_{\text{CC}}$	—	1	—	ms	

\* : Before the power rising,  $V_{\text{CC}}$  must be less than 0.2 V.

Note: The above standards are the values needed in order to activate a power-on reset.



If power supply voltage needs to be changed in the course of operation, a smooth voltage rise is recommended by suppressing the voltage variation as shown below.



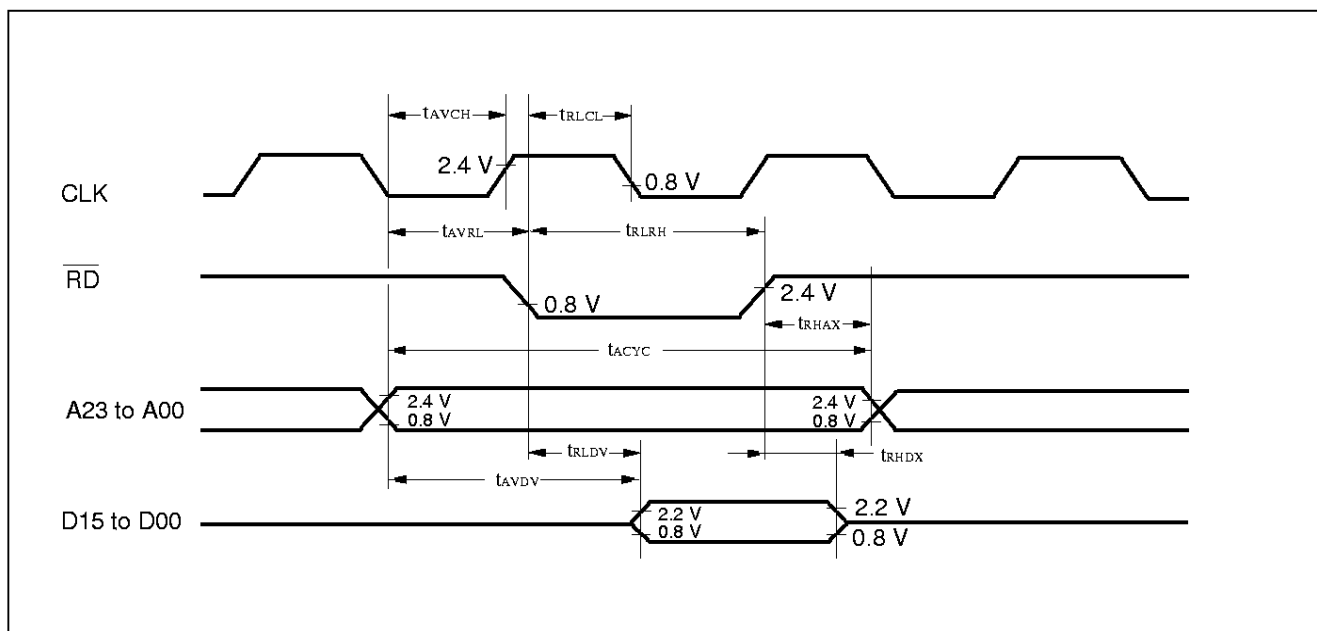
# MB90F243H

## (5) Bus Read Timing

( $AV_{CC} = V_{CC} = 5.0 \text{ V} \pm 10\%$ ,  $AV_{SS} = V_{SS} = 0.0 \text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Address cycle time	$t_{ACYC}$	A23 to A00	—	$2 t_{CYC}^* - 10$	—	ns	
Valid address $\rightarrow \overline{RD} \downarrow$ time	$t_{AVRL}$	A23 to A00		$1 t_{CYC}^*/2 - 15$	—	ns	
$\overline{RD}$ pulse width	$t_{RLRH}$	$\overline{RD}$		$1 t_{CYC}^* - 15$	—	ns	
$\overline{RD} \downarrow \rightarrow$ data read time	$t_{RLDV}$	D15 to D00		—	$1 t_{CYC}^* - 20$	ns	
Valid address $\rightarrow$ data read time	$t_{AVDV}$	D15 to D00		—	$3 t_{CYC}^*/2 - 40$	ns	
$\overline{RD} \uparrow \rightarrow$ data hold time	$t_{RHDX}$	D15 to D00		0	—	ns	
$\overline{RD} \uparrow \rightarrow$ address valid time	$t_{RHAX}$	A23 to A00		$1 t_{CYC}^*/2 - 20$	—	ns	
Valid address $\rightarrow$ CLK $\uparrow$ time	$t_{AVCH}$	A23 to A00, CLK		$1 t_{CYC}^*/2 - 25$	—	ns	
$\overline{RD} \downarrow \rightarrow$ CLK $\downarrow$ time	$t_{RLCL}$	$\overline{RD}$ , CLK		$1 t_{CYC}^*/2 - 25$	—	ns	

\* : For information on  $t_{CYC}$  (cycle time), see “(2) Clock Output Timing.”

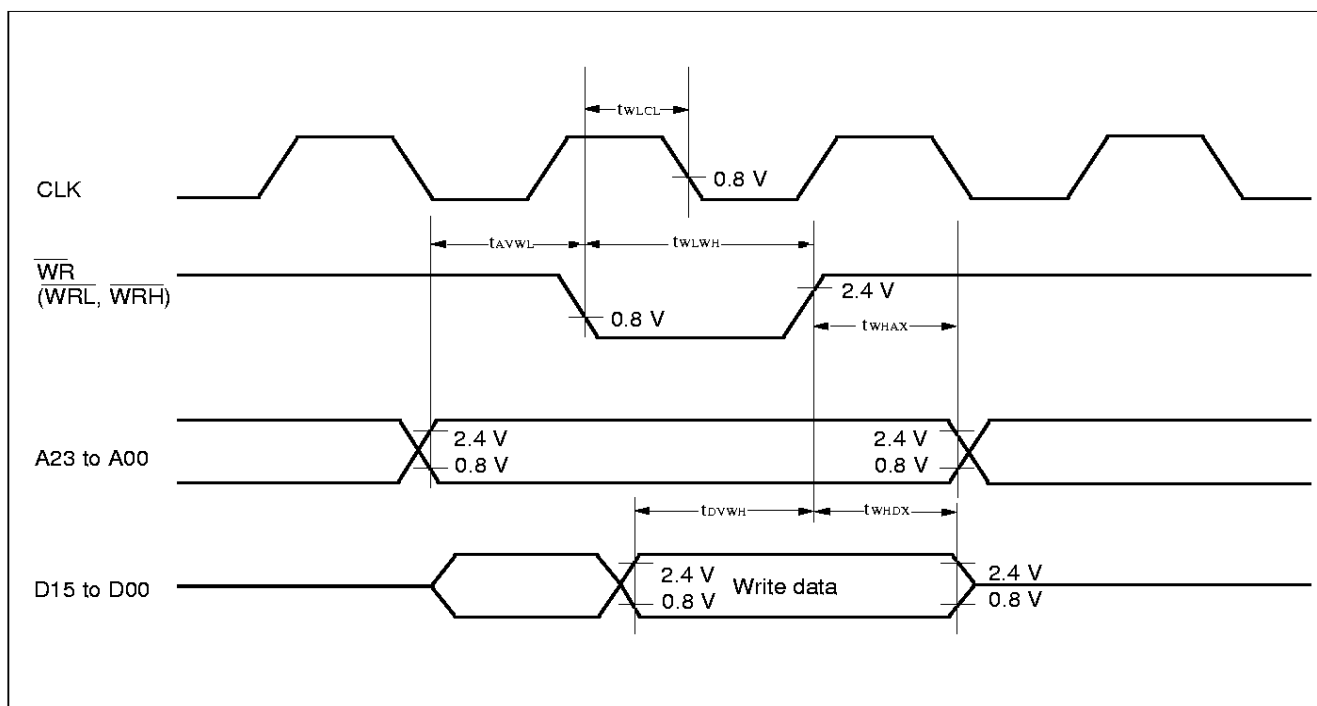


## MB90F243H

## (6) Bus Write Timing

(AV<sub>CC</sub> = V<sub>CC</sub> = 5.0 V ±10%, AV<sub>SS</sub> = V<sub>SS</sub> = 0.0 V, T<sub>A</sub> = 0°C to +70°C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Valid address → $\overline{WR}$ ↓ time	t <sub>AVWL</sub>	A23 to A00	—	1 t <sub>cyt</sub> * / 2 – 15	—	ns	
$\overline{WR}$ pulse width	t <sub>WLWH</sub>	$\overline{WRL}$ , $\overline{WRH}$		1 t <sub>cyt</sub> * – 25	—	ns	
Write data → $\overline{WR}$ ↑ time	t <sub>DVWH</sub>	D15 to D00		1 t <sub>cyt</sub> * – 40	—	ns	
$\overline{WR}$ ↑ → Data hold time	t <sub>WHDX</sub>	D15 to D00		1 t <sub>cyt</sub> * / 2 – 15	—	ns	
$\overline{WR}$ ↑ → Address valid time	t <sub>WHAX</sub>	A23 to A00		1 t <sub>cyt</sub> * / 2 – 15	—	ns	
$\overline{WR}$ ↑ → CLK ↓ time	t <sub>WLCL</sub>	$\overline{WRL}$ , $\overline{WRH}$ , CLK		1 t <sub>cyt</sub> * / 2 – 25	—	ns	

\* : For information on t<sub>cyt</sub> (cycle time), see “(2) Clock Output Timing.”

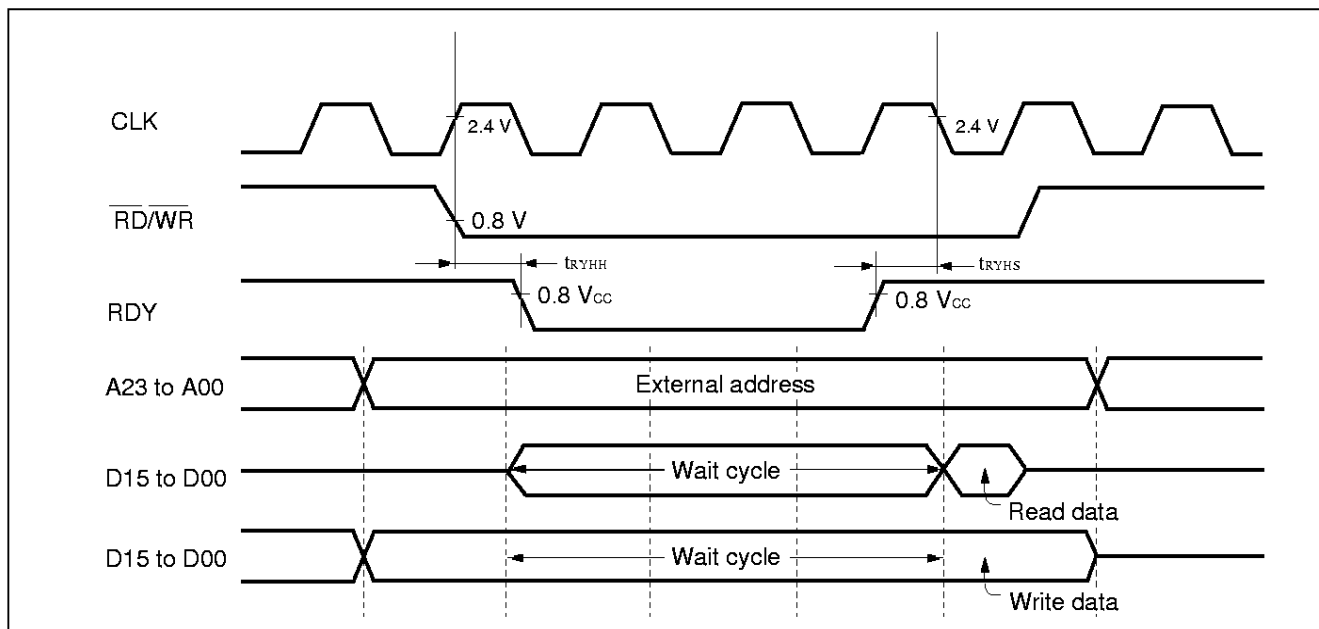
# MB90F243H

## (7) Ready Input Timing

( $A_{V_{CC}} = V_{CC} = 5.0 \text{ V} \pm 10\%$ ,  $A_{V_{SS}} = V_{SS} = 0.0 \text{ V}$ ,  $T_A = 0^\circ\text{C to } +70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
RDY setup time	$t_{RYHS}$	RDY	Source oscillation 40 MHz	15	47	ns	
RDY hold time	$t_{RYHH}$	RDY		0	47	ns	

Note: If the RDY setup time is insufficient, use the auto ready function.



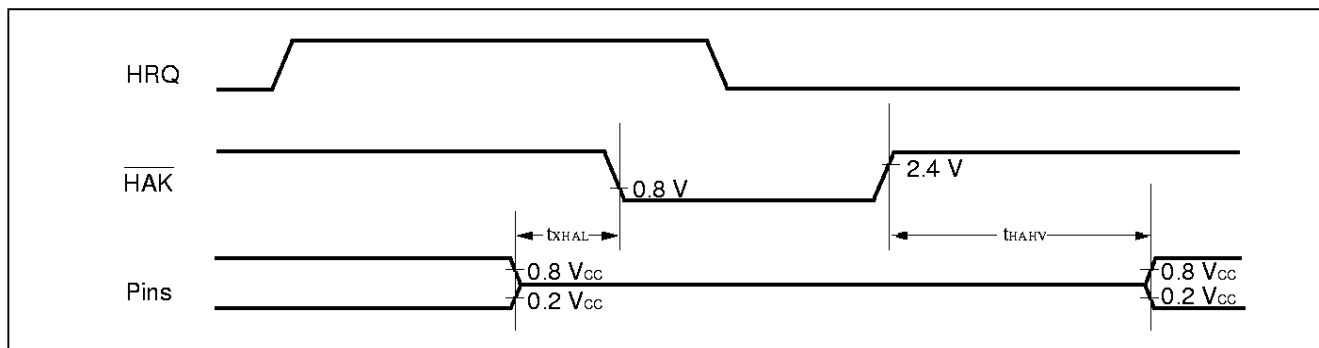
## (8) Hold Timing

( $A_{V_{CC}} = V_{CC} = 5.0 \text{ V} \pm 10\%$ ,  $A_{V_{SS}} = V_{SS} = 0.0 \text{ V}$ ,  $T_A = 0^\circ\text{C to } +70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Pin floating → $\overline{\text{HAK}} \downarrow$ time	$t_{XHAL}$	$\overline{\text{HAK}}$	—	30	1 $t_{CYC}^*$	ns	
$\overline{\text{HAK}} \uparrow$ time → Pin valid time	$t_{HAHV}$	$\overline{\text{HAK}}$		1 $t_{CYC}^*$	2 $t_{CYC}^*$	ns	

\* : For information on  $t_{CYC}$  (cycle time), see “(2) Clock Output Timing.”

Note: At least one cycle is required from the time when HRQ is fetched until  $\overline{\text{HAK}}$  changes.





# MB90F243H

## (9) UART Timing

( $AV_{CC} = V_{CC} = 5.0\text{ V} \pm 10\%$ ,  $AV_{SS} = V_{SS} = 0.0\text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

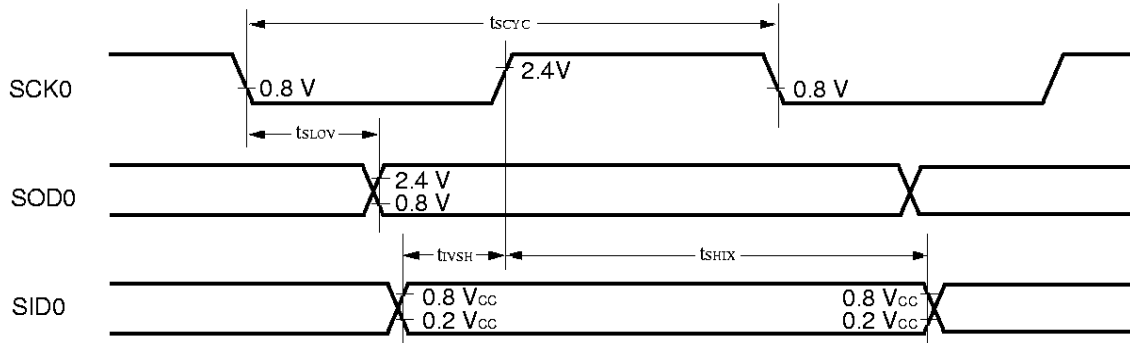
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Serial clock cycle time	tscyc	—	For internal shift clock mode output pin, $C_L = 80\text{ pF} + 1\text{ TTL}$	8 tcyc*	—	ns	
SCK ↓ → SOD delay time	tslov	—		-80	80	ns	
Valid SID → SCK ↑	tivsh	—		100	—	ns	
SCK ↑ → Valid SID hold time	tshix	—		60	—	ns	
Serial clock "H" pulse width	tshsl	—	For external shift clock mode output pin, $C_L = 80\text{ pF} + 1\text{ TTL}$	4 tcyc*	—	ns	
Serial clock "L" pulse width	tslsh	—		4 tcyc*	—	ns	
SCK ↓ → SOD delay time delay time	tslov	—		—	150	ns	
Valid SID → SCK ↑	tivsh	—		60	—	ns	
SCK ↑ → Valid SID hold time	tshix	—		60	—	ns	

\* : For information on tcyc (cycle time), see "(2) Clock Output Timing."

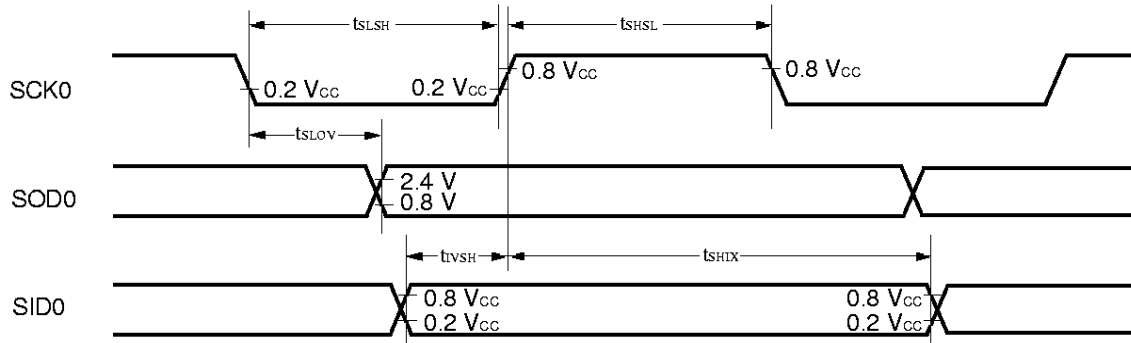
Notes: • These are the AC characteristics for CLK synchronous mode.  
•  $C_L$  is the load capacitance added to pins during testing.

# MB90F243H

## • Internal shift clock mode



## • External shift clock mode



# MB90F243H

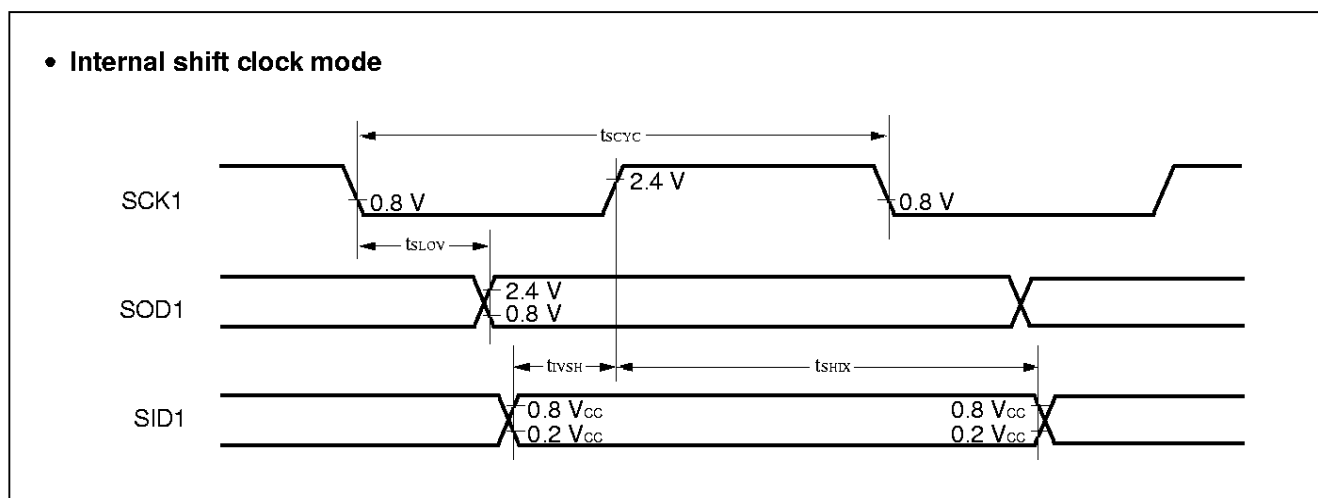
## (10) Serial I/O Timing

( $A_{V_{CC}} = V_{CC} = 5.0 \text{ V} \pm 10\%$ ,  $A_{V_{SS}} = V_{SS} = 0.0 \text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Serial clock cycle time	$t_{SCYC}$	—	For internal shift clock mode output pin, $C_L = 80 \text{ pF}$	$2 t_{CYC}^*$	—	ns	
SCK $\uparrow$ $\rightarrow$ SOD delay time	$t_{SLOV}$	—		—	$t_{CYC}^*/2$	ns	
Valid SID $\rightarrow$ SCK $\uparrow$	$t_{VSH}$	—		$1 t_{CYC} - 15$	—	ns	
SCK $\uparrow$ $\rightarrow$ Valid SID hold time	$t_{SHIX}$	—		$1 t_{CYC}^*$	—	ns	

\* : For information on  $t_{CYC}$  (cycle time), see “(2) Clock Output Timing.”

Note:  $C_L$  is the load capacitance added to pins during testing.



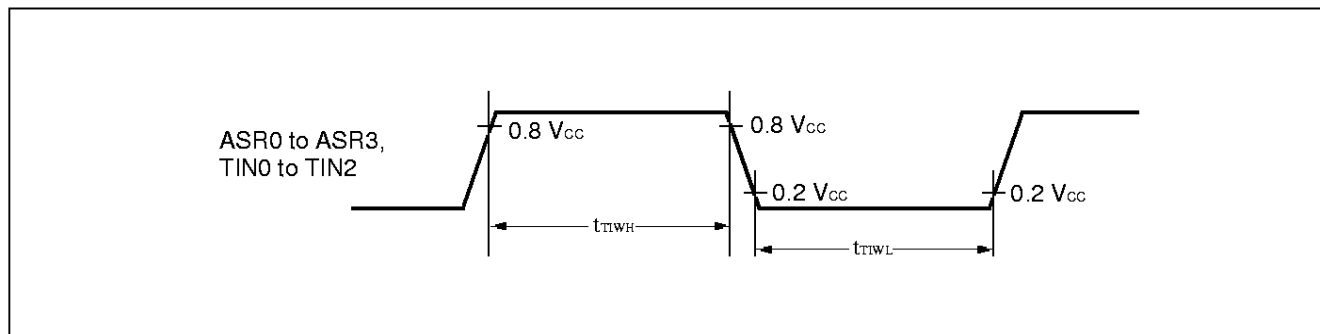
# MB90F243H

## (11) Timer Input Timing

( $AV_{CC} = V_{CC} = 5.0\text{ V} \pm 10\%$ ,  $AV_{SS} = V_{SS} = 0.0\text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Input pulse width	$t_{PIWH}$ , $t_{PIWL}$	ASR0 to ASR3, TIN0 to TIN2	—	4 $t_{CYC}$ *	—	ns	

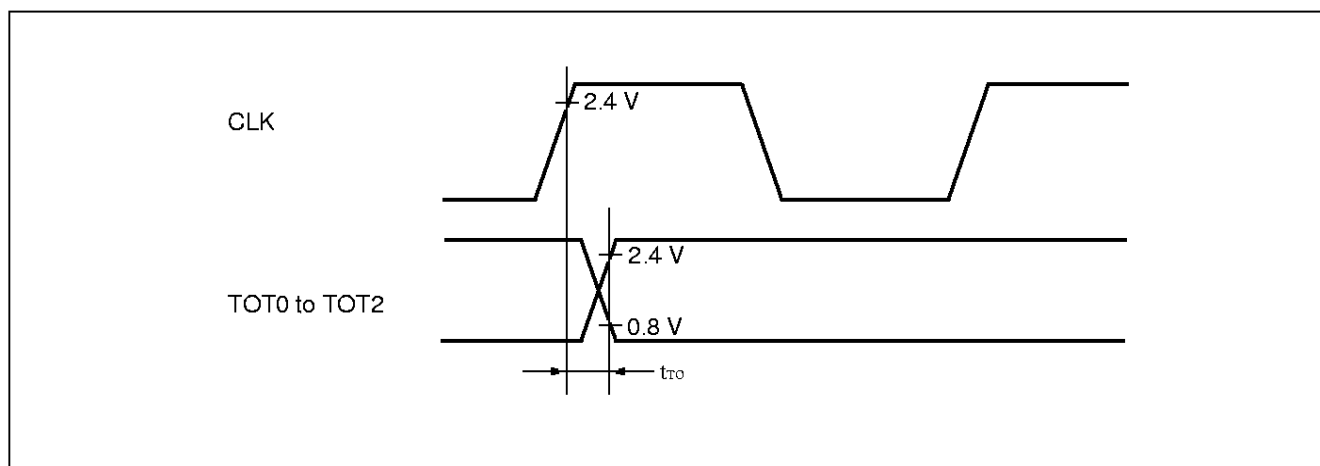
\* : For information on  $t_{CYC}$  (cycle time), see “(2) Clock Output Timing.”



## (12) Timer Output Timing

( $AV_{CC} = V_{CC} = 5.0\text{ V} \pm 10\%$ ,  $AV_{SS} = V_{SS} = 0.0\text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
CLK $\uparrow$ → Change time	$t_{TO}$	TOT0 to TOT2	$V_{CC} = 5.0\text{ V} \pm 10\%$	—	40	ns	



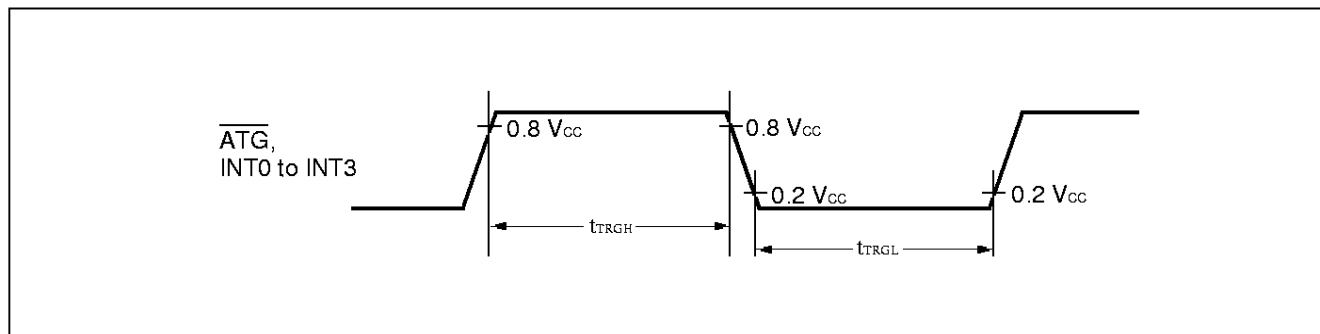
# MB90F243H

## (13) Trigger Input Timing

( $A_{V_{CC}} = V_{CC} = 5.0 \text{ V} \pm 10\%$ ,  $A_{V_{SS}} = V_{SS} = 0.0 \text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Input pulse width	$t_{TRGH}$ , $t_{TRGL}$	$\overline{\text{ATG}}$ , INT0 to INT3	—	5 $t_{CYC}$ *	—	ns	

\* : For information on  $t_{CYC}$  (cycle time), see “(2) Clock Output Timing.”



# MB90F243H

## 6. A/D Converter Electrical Characteristics

( $AV_{CC} = V_{CC} = 5.0\text{ V} \pm 10\%$ ,  $AV_{SS} = V_{SS} = 0.0\text{ V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ )

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
Resolution	—	—	—	—	8	10	bit	
	—	—		—	10	10		
Total error	—	—	—	—	—	$\pm 3.0$	LSB	
Linearity error	—	—	—	—	—	$\pm 2.0$	LSB	
Differential linearity error	—	—	—	—	—	$\pm 1.9$	LSB	
Zero transition voltage	$V_{0T}$	AN0 to AN3, AN6, AN7	—	AVRL – 1.0 LSB	AVRL + 1.0 LSB	AVRL + 3.0 LSB	mV	
Full-scale transition voltage	$V_{FST}$	AN0 to AN3, AN6, AN7	—	AVRH – 4.0 LSB	AVRH – 1.0 LSB	AVRH + 1.0 LSB	mV	
Conversion time	—	—	Setup by ADCT register $V_{CC} = 5.0\text{ V} \pm 10\%^{*1}$	1.0	—	—	$\mu\text{s}$	
Sampling period	—	—		450	—	—	ns	
Conversion period a	—	—		100	—	—	ns	
Conversion period b	—	—		100	—	—	ns	
Conversion period c	—	—		200	—	—	ns	
Analog port input current	$I_{AIN}$	AN0 to AN3, AN6, AN7	—	—	0.1	3	$\mu\text{A}$	
Analog input voltage	—	AN0 to AN3, AN6, AN7	—	AVRL	—	AVRH	V	
Reference voltage	—	AVRH	$AVRH - AVRL \geq 2.7$	AVRL + 2.7	—	$AV_{CC}$	V	
	—	AVRL		0	—	$AVRH - 2.7$	V	
Power supply current	$I_A$	$AV_{CC}$	$AV_{CC} = 5.5\text{ V}$ Stop mode	—	15	25	mA	
	$I_{AS}^{*2}$	$AV_{CC}$		—	—	5	$\mu\text{A}$	
Reference voltage supply current	$I_R$	AVRH	$AV_{CC} = 5.5\text{ V}$ Stop mode	—	1.5	2	mA	
	$I_{RS}^{*2}$	AVRH		—	—	5	$\mu\text{A}$	
Interchannel disparity	—	AN0 to AN3, AN6, AN7	—	—	—	4	LSB	

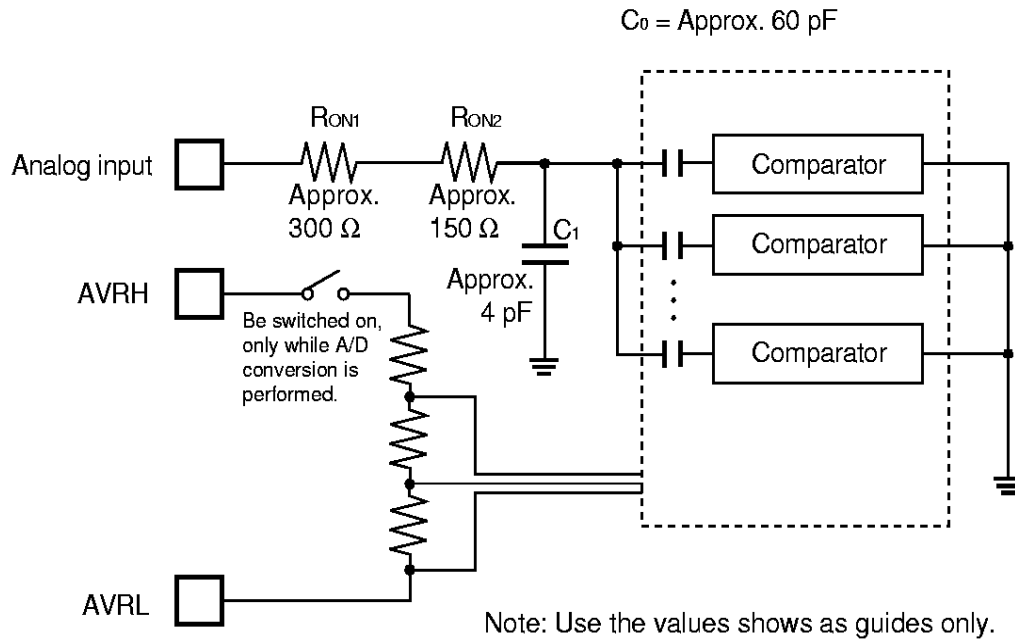
\*1: When  $F_c = 40\text{ MHz}$  (frequency), and the machine cycle is  $50.0\text{ ns}$ .

\*2: Current when the A/D converter is not operating and the CPU is stopped.

Notes: • The smaller  $|AVRH - AVRL|$ , the greater the error would become relatively.

- If the output impedance of the external circuit for the analog input is high, sampling period might be insufficient. When the sampling period set at near the minimum value, the output impedance of the external circuit should be less than approximately  $300\ \Omega$ .

- Analog input circuit model diagram



# MB90F243H

## 7. A/D Converter Glossary

### Resolution

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

When the number of bits is 10, analog voltage can be divide into  $2^{10}$ .

### Linearity error

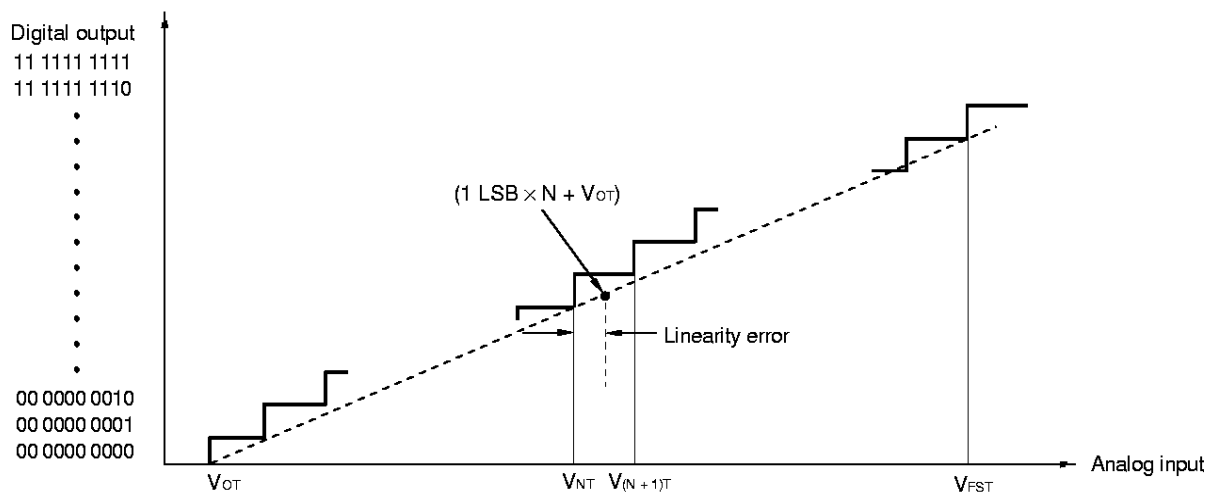
The deviation of the straight line connecting the zero transition point ("00 0000 0000" ↔ "00 0000 0001") with the full-scale transition point ("11 1111 1110" ↔ "11 1111 1111") from actual conversion characteristics

### Differential linearity error

The deviation of input voltage needed to change the output code by 1 LSB from the theoretical value

### Total error (unit: LSB)

The difference between theoretical and actual conversion values caused by the zero transition error, full-scale transition error, non-linearity error, differential linearity error, and noise



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

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

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

$V_{\text{OT}}$ : Voltage for digital output transit from "000<sub>H</sub>" to "001<sub>H</sub>"

$V_{\text{FST}}$ : Voltage for digital output transit from "3FE<sub>H</sub>" to "3FF<sub>H</sub>"



## ■ INSTRUCTION SET (412 INSTRUCTIONS)

Table 1 Explanation of Items in Table of Instructions

Item	Explanation
Mnemonic	Upper-case letters and symbols: Represented as they appear in assembler Lower-case letters: Replaced when described in assembler. Numbers after lower-case letters: Indicate the bit width within the instruction.
#	Indicates the number of bytes.
~	Indicates the number of cycles. See Table 4 for details about meanings of letters in items.
B	Indicates the correction value for calculating the number of actual cycles during execution of instruction. The number of actual cycles during execution of instruction is summed with the value in the "cycles" column.
Operation	Indicates operation of instruction.
LH	Indicates special operations involving the bits 15 through 08 of the accumulator. Z: Transfers "0". X: Extends before transferring. —: Transfers nothing.
AH	Indicates special operations involving the high-order 16 bits in the accumulator. *: Transfers from AL to AH. —: No transfer. Z: Transfers 00 <sub>H</sub> to AH. X: Transfers 00 <sub>H</sub> or FF <sub>H</sub> to AH by extending AL.
I	Indicates the status of each of the following flags: I (interrupt enable), S (stack), T (sticky bit), N (negative), Z (zero), V (overflow), and C (carry). *: Changes due to execution of instruction. —: No change. S: Set by execution of instruction. R: Reset by execution of instruction.
S	
T	
N	
Z	
V	
C	
RMW	Indicates whether the instruction is a read-modify-write instruction (a single instruction that reads data from memory, etc., processes the data, and then writes the result to memory). *: Instruction is a read-modify-write instruction —: Instruction is not a read-modify-write instruction Note: Cannot be used for addresses that have different meanings depending on whether they are read or written.

# MB90F243H

**Table 2 Explanation of Symbols in Table of Instructions**

Symbol	Explanation
A	32-bit accumulator The number of bits used varies according to the instruction. Byte: Low order 8 bits of AL Word: 16 bits of AL Long: 32 bits of AL, AH
AH	High-order 16 bits of A
AL	Low-order 16 bits of A
SP	Stack pointer (USP or SSP)
PC	Program counter
SPCU	Stack pointer upper limit register
SPCL	Stack pointer lower limit register
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	Direct addressing
addr24	Physical direct addressing
addr24 0 to 15	Bits 0 to 15 of addr24
addr24 16 to 23	Bits 16 to 23 of addr24
io	I/O area (000000H to 0000FFH)

(Continued)

# MB90F243H

(Continued)

Symbol	Explanation
#imm4	4-bit immediate data
#imm8	8-bit immediate data
#imm16	16-bit immediate data
#imm32	32-bit immediate data
ext (imm8)	16-bit data signed and extended from 8-bit immediate data
disp8	8-bit displacement
disp16	16-bit displacement
bp	Bit offset value
vct4	Vector number (0 to 15)
vct8	Vector number (0 to 255)
( )b	Bit address
rel	Branch specification relative to PC
ear	Effective addressing (codes 00 to 07)
eam	Effective addressing (codes 08 to 1F)
rlst	Register list

# MB90F243H

**Table 3 Effective Address Fields**

Code	Notation	Address format	Number of bytes in address extension*
00 01 02 03 04 05 06 07	R0 RW0 RL0 R1 RW1 (RL0) R2 RW2 RL1 R3 RW3 (RL1) R4 RW4 RL2 R5 RW5 (RL2) R6 RW6 RL3 R7 RW7 (RL3)	Register direct "ea" corresponds to byte, word, and long-word types, starting from the left	—
08 09 0A 0B	@RW0 @RW1 @RW2 @RW3	Register indirect	0
0C 0D 0E 0F	@RW0 + @RW1 + @RW2 + @RW3 +	Register indirect with post-increment	0
10 11 12 13 14 15 16 17	@RW0 + disp8 @RW1 + disp8 @RW2 + disp8 @RW3 + disp8 @RW4 + disp8 @RW5 + disp8 @RW6 + disp8 @RW7 + disp8	Register indirect with 8-bit displacement	1
18 19 1A 1B	@RW0 + disp16 @RW1 + disp16 @RW2 + disp16 @RW3 + disp16	Register indirect with 16-bit displacement	2
1C 1D 1E 1F	@RW0 + RW7 @RW1 + RW7 @PC + dip16 addr16	Register indirect with index Register indirect with index PC indirect with 16-bit displacement Direct address	0 0 2 2

\* : The number of bytes for address extension is indicated by the "+" symbol in the "#" (number of bytes) column in the Table of Instructions.

# MB90F243H

**Table 4 Number of Execution Cycles for Each Form of Addressing**

Code	Operand	(a)*
		Number of execution cycles for each form of addressing
00 to 07	Ri RWi RLi	Listed in Table of Instructions
08 to 0B	@RWj	1
0C to 0F	@RWj +	4
10 to 17	@RWi + disp8	1
18 to 1B	@RWj + disp16	1
1C	@RW0 + RW7	2
1D	@RW1 + RW7	2
1E	@PC + dip16	2
1F	@addr16	1

\* :“(a)” is used in the “cycles” (number of cycles) column and column B (correction value) in the Table of Instructions.

**Table 5 Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles**

Operand	(b)*	(c)*	(d)*
	byte	word	long
Internal register	+ 0	+ 0	+ 0
Internal RAM even address	+ 0	+ 0	+ 0
Internal RAM odd address	+ 0	+ 1	+ 2
Even address not in internal RAM	+ 1	+ 1	+ 2
Odd address not in internal RAM	+ 1	+ 3	+ 6
External data bus (8 bits)	+ 1	+ 3	+ 6

\* :“(b)”, “(c)”, and “(d)” are used in the “cycles” (number of cycles) column and column B (correction value) in the Table of Instructions.

## MB90F243H

Table 6 Transfer Instructions (Byte) [50 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOV A, dir	2	2	(b)	byte (A) ← (dir)	Z	*	—	—	—	*	*	—	—	—
MOV A, addr16	3	2	(b)	byte (A) ← (addr16)	Z	*	—	—	—	*	*	—	—	—
MOV A, Ri	1	1	0	byte (A) ← (Ri)	Z	*	—	—	—	*	*	—	—	—
MOV A, ear	2	1	0	byte (A) ← (ear)	Z	*	—	—	—	*	*	—	—	—
MOV A, eam	2+	2+ (a)	(b)	byte (A) ← (eam)	Z	*	—	—	—	*	*	—	—	—
MOV A, io	2	2	(b)	byte (A) ← (io)	Z	*	—	—	—	*	*	—	—	—
MOV A, #imm8	2	2	0	byte (A) ← imm8	Z	*	—	—	—	*	*	—	—	—
MOV A, @A	2	2	(b)	byte (A) ← ((A))	Z	—	—	—	—	*	*	—	—	—
MOV A, @RLi+disp8	3	6	(b)	byte (A) ← ((RLi))+disp8)	Z	*	—	—	—	*	*	—	—	—
MOV A, @SP+disp8	3	3	(b)	byte (A) ← ((SP)+disp8)	Z	*	—	—	—	*	*	—	—	—
MOVP A, addr24	5	3	(b)	byte (A) ← (addr24)	Z	*	—	—	—	*	*	—	—	—
MOVP A, @A	2	2	(b)	byte (A) ← ((A))	Z	—	—	—	—	*	*	—	—	—
MOVN A, #imm4	1	1	0	byte (A) ← imm4	Z	*	—	—	—	R	*	—	—	—
MOVX A, dir	2	2	(b)	byte (A) ← (dir)	X	*	—	—	—	*	*	—	—	—
MOVX A, addr16	3	2	(b)	byte (A) ← (addr16)	X	*	—	—	—	*	*	—	—	—
MOVX A, Ri	2	1	0	byte (A) ← (Ri)	X	*	—	—	—	*	*	—	—	—
MOVX A, ear	2	1	0	byte (A) ← (ear)	X	*	—	—	—	*	*	—	—	—
MOVX A, eam	2+	2+ (a)	(b)	byte (A) ← (eam)	X	*	—	—	—	*	*	—	—	—
MOVX A, io	2	2	(b)	byte (A) ← (io)	X	*	—	—	—	*	*	—	—	—
MOVX A, #imm8	2	2	0	byte (A) ← imm8	X	*	—	—	—	*	*	—	—	—
MOVX A, @A	2	2	(b)	byte (A) ← ((A))	X	—	—	—	—	*	*	—	—	—
MOVX A, @RWi+disp8	2	3	(b)	byte (A) ← ((RWi))+disp8)	X	*	—	—	—	*	*	—	—	—
MOVX A, @RLi+disp8	3	6	(b)	byte (A) ← ((RLi))+disp8)	X	*	—	—	—	*	*	—	—	—
MOVX A, @SP+disp8	3	3	(b)	byte (A) ← ((SP)+disp8)	X	*	—	—	—	*	*	—	—	—
MOVPX A, addr24	5	3	(b)	byte (A) ← (addr24)	X	*	—	—	—	*	*	—	—	—
MOVPX A, @A	2	2	(b)	byte (A) ← ((A))	X	—	—	—	—	*	*	—	—	—
MOV dir, A	2	2	(b)	byte (dir) ← (A)	—	—	—	—	—	*	*	—	—	—
MOV addr16, A	3	2	(b)	byte (addr16) ← (A)	—	—	—	—	—	*	*	—	—	—
MOV Ri, A	1	1	0	byte (Ri) ← (A)	—	—	—	—	—	*	*	—	—	—
MOV ear, A	2	2	0	byte (ear) ← (A)	—	—	—	—	—	*	*	—	—	—
MOV eam, A	2+	2+ (a)	(b)	byte (eam) ← (A)	—	—	—	—	—	*	*	—	—	—
MOV io, A	2	2	(b)	byte (io) ← (A)	—	—	—	—	—	*	*	—	—	—
MOV @RLi+disp8, A	3	6	(b)	byte ((RLi) +disp8) ← (A)	—	—	—	—	—	*	*	—	—	—
MOV @SP+disp8, A	3	3	(b)	byte ((SP)+disp8) ← (A)	—	—	—	—	—	*	*	—	—	—
MOVP addr24, A	5	3	(b)	byte (addr24) ← (A)	—	—	—	—	—	*	*	—	—	—
MOV Ri, ear	2	2	0	byte (Ri) ← (ear)	—	—	—	—	—	*	*	—	—	—
MOV Ri, eam	2+	3+ (a)	(b)	byte (Ri) ← (eam)	—	—	—	—	—	*	*	—	—	—
MOVP @A, Ri	2	3	(b)	byte ((A)) ← (Ri)	—	—	—	—	—	*	*	—	—	—
MOV ear, Ri	2	3	0	byte (ear) ← (Ri)	—	—	—	—	—	*	*	—	—	—
MOV eam, Ri	2+	3+ (a)	(b)	byte (eam) ← (Ri)	—	—	—	—	—	*	*	—	—	—
MOV Ri, #imm8	2	2	0	byte (Ri) ← imm8	—	—	—	—	—	*	*	—	—	—
MOV io, #imm8	3	3	(b)	byte (io) ← imm8	—	—	—	—	—	—	—	—	—	—
MOV dir, #imm8	3	3	(b)	byte (dir) ← imm8	—	—	—	—	—	—	—	—	—	—
MOV ear, #imm8	3	2	0	byte (ear) ← imm8	—	—	—	—	—	*	*	—	—	—
MOV eam, #imm8	3+	2+ (a)	(b)	byte (eam) ← imm8	—	—	—	—	—	—	—	—	—	—
MOV @AL, AH	2	2	(b)	byte ((A)) ← (AH)	—	—	—	—	—	*	*	—	—	—

(Continued)

# MB90F243H

(Continued)

Mnemonic		#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
XCH	A, ear	2	3	0	byte (A) ↔ (ear)	Z	-	-	-	-	-	-	-	-	-
XCH	A, eam	2+	3+ (a)	2× (b)	byte (A) ↔ (eam)	Z	-	-	-	-	-	-	-	-	-
XCH	Ri, ear	2	4	0	byte (Ri) ↔ (ear)	-	-	-	-	-	-	-	-	-	-
XCH	Ri, eam	2+	5+ (a)	2× (b)	byte (Ri) ↔ (eam)	-	-	-	-	-	-	-	-	-	-

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

# MB90F243H

**Table 7 Transfer Instructions (Word) [40 Instructions]**

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOVW A, dir	2	2	(c)	word (A) ← (dir)	-	*	-	-	-	*	*	-	-	-
MOVW A, addr16	3	2	(c)	word (A) ← (addr16)	-	*	-	-	-	*	*	-	-	-
MOVW A, SP	1	2	0	word (A) ← (SP)	-	*	-	-	-	*	*	-	-	-
MOVW A, RWi	1	1	0	word (A) ← (RWi)	-	*	-	-	-	*	*	-	-	-
MOVW A, ear	2	1	0	word (A) ← (ear)	-	*	-	-	-	*	*	-	-	-
MOVW A, eam	2+	2+ (a)	(c)	word (A) ← (eam)	-	*	-	-	-	*	*	-	-	-
MOVW A, io	2	2	(c)	word (A) ← (io)	-	*	-	-	-	*	*	-	-	-
MOVW A, @A	2	2	(c)	word (A) ← ((A))	-	-	-	-	-	*	*	-	-	-
MOVW A, #imm16	3	2	0	word (A) ← imm16	-	*	-	-	-	*	*	-	-	-
MOVW A, @RWi+disp8	2	3	(c)	word (A) ← ((RWi) +disp8)	-	*	-	-	-	*	*	-	-	-
MOVW A, @RLi+disp8	3	6	(c)	word (A) ← ((RLi) +disp8)	-	*	-	-	-	*	*	-	-	-
MOVW A, @SP+disp8	3	3	(c)	word (A) ← ((SP) +disp8)	-	*	-	-	-	*	*	-	-	-
MOVPWA, addr24	5	3	(c)	word (A) ← (addr24)	-	*	-	-	-	*	*	-	-	-
MOVPWA, @A	2	2	(c)	word (A) ← ((A))	-	-	-	-	-	*	*	-	-	-
MOVW dir, A	2	2	(c)	word (dir) ← (A)	-	-	-	-	-	*	*	-	-	-
MOVW addr16, A	3	2	(c)	word (addr16) ← (A)	-	-	-	-	-	*	*	-	-	-
MOVW SP, #imm16	4	2	0	word (SP) ← imm16	-	-	-	-	-	*	*	-	-	-
MOVW SP, A	1	2	0	word (SP) ← (A)	-	-	-	-	-	*	*	-	-	-
MOVW RWi, A	1	1	0	word (RWi) ← (A)	-	-	-	-	-	*	*	-	-	-
MOVW ear, A	2	2	0	word (ear) ← (A)	-	-	-	-	-	*	*	-	-	-
MOVW eam, A	2+	2+ (a)	(c)	word (eam) ← (A)	-	-	-	-	-	*	*	-	-	-
MOVW io, A	2	2	(c)	word (io) ← (A)	-	-	-	-	-	*	*	-	-	-
MOVW @RWi+disp8, A	2	3	(c)	word ((RWi) +disp8) ← (A)	-	-	-	-	-	*	*	-	-	-
MOVW @RLi+disp8, A	3	6	(c)	word ((RLi) +disp8) ← (A)	-	-	-	-	-	*	*	-	-	-
MOVW @SP+disp8, A	3	3	(c)	word ((SP) +disp8) ← (A)	-	-	-	-	-	*	*	-	-	-
MOVPWaddr24, A	5	3	(c)	word (addr24) ← (A)	-	-	-	-	-	*	*	-	-	-
MOVPW@A, RWi	2	3	(c)	word ((A)) ← (RWi)	-	-	-	-	-	*	*	-	-	-
MOVW RWi, ear	2	2	0	word (RWi) ← (ear)	-	-	-	-	-	*	*	-	-	-
MOVW RWi, eam	2+	3+ (a)	(c)	word (RWi) ← (eam)	-	-	-	-	-	*	*	-	-	-
MOVW ear, RWi	2	3	0	word (ear) ← (RWi)	-	-	-	-	-	*	*	-	-	-
MOVW eam, RWi	2+	3+ (a)	(c)	word (eam) ← (RWi)	-	-	-	-	-	*	*	-	-	-
MOVW RWi, #imm16	3	2	0	word (RWi) ← imm16	-	-	-	-	-	*	*	-	-	-
MOVW io, #imm16	4	3	(c)	word (io) ← imm16	-	-	-	-	-	-	-	-	-	-
MOVW ear, #imm16	4	2	0	word (ear) ← imm16	-	-	-	-	-	*	*	-	-	-
MOVW eam, #imm16	4+	2+ (a)	(c)	word (eam) ← imm16	-	-	-	-	-	-	-	-	-	-
MOVW @AL, AH	2	2	(c)	word ((A)) ← (AH)	-	-	-	-	-	*	*	-	-	-
XCHW A, ear	2	3	0	word (A) ↔ (ear)	-	-	-	-	-	-	-	-	-	-
XCHW A, eam	2+	3+ (a)	2× (c)	word (A) ↔ (eam)	-	-	-	-	-	-	-	-	-	-
XCHW RWi, ear	2	4	0	word (RWi) ↔ (ear)	-	-	-	-	-	-	-	-	-	-
XCHW RWi, eam	2+	5+ (a)	2× (c)	word (RWi) ↔ (eam)	-	-	-	-	-	-	-	-	-	-

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



## MB90F243H

Table 8 Transfer Instructions (Long Word) [11 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOVL A, ear	2	1	0	long (A) ← (ear)	—	—	—	—	—	*	*	—	—	—
MOVL A, eam	2+	3+ (a)	(d)	long (A) ← (eam)	—	—	—	—	—	*	*	—	—	—
MOVL A, # imm32	5	3	0	long (A) ← imm32	—	—	—	—	—	*	*	—	—	—
MOVL A, @SP + disp8	3	4	(d)	long (A) ← ((SP) + disp8)	—	—	—	—	—	*	*	—	—	—
MOVPL A, addr24	5	4	(d)	long (A) ← (addr24)	—	—	—	—	—	*	*	—	—	—
MOVPL A, @A	2	3	(d)	long (A) ← ((A))	—	—	—	—	—	*	*	—	—	—
MOVPL@A, RLi	2	5	(d)	long ((A)) ← (RLi)	—	—	—	—	—	*	*	—	—	—
MOVL @SP + disp8, A	3	4	(d)	long ((SP) + disp8) ← (A)	—	—	—	—	—	*	*	—	—	—
MOVPL addr24, A	5	4	(d)	long (addr24) ← (A)	—	—	—	—	—	*	*	—	—	—
MOVL ear, A	2	2	0	long (ear) ← (A)	—	—	—	—	—	*	*	—	—	—
MOVL eam, A	2+	3+ (a)	(d)	long (eam) ← (A)	—	—	—	—	—	*	*	—	—	—

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

# MB90F243H

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

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

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

# MB90F243H

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

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

For an explanation of “(a)”, “(b)”, “(c)” and “(d)”, refer to Table 4, “Number of Execution Cycles for Each Form 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	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
CMP A	1	2	0	byte (AH) – (AL)	–	–	–	–	–	*	*	*	*	–
CMP A, ear	2	2	0	byte (A) – (ear)	–	–	–	–	–	*	*	*	*	–
CMP A, eam	2+	2+ (a)	(b)	byte (A) – (eam)	–	–	–	–	–	*	*	*	*	–
CMP A, #imm8	2	2	0	byte (A) – imm8	–	–	–	–	–	*	*	*	*	–
CMPW A	1	2	0	word (AH) – (AL)	–	–	–	–	–	*	*	*	*	–
CMPW A, ear	2	2	0	word (A) – (ear)	–	–	–	–	–	*	*	*	*	–
CMPW A, eam	2+	2+ (a)	(c)	word (A) – (eam)	–	–	–	–	–	*	*	*	*	–
CMPW A, #imm16	3	2	0	word (A) – imm16	–	–	–	–	–	*	*	*	*	–
CMPL A, ear	2	3	0	long (A) – (ear)	–	–	–	–	–	*	*	*	*	–
CMPL A, eam	2+	4+ (a)	(d)	long (A) – (eam)	–	–	–	–	–	*	*	*	*	–
CMPL A, #imm32	5	3	0	long (A) – imm32	–	–	–	–	–	*	*	*	*	–

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

# MB90F243H

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

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

For an explanation of “(b)” and “(c)”, refer to Table 5, “Correction Values for Number of Cycle Used to Calculate Number of Actual Cycles.”

- \*1: 3 when dividing into zero, 6 when an overflow occurs, and 14 normally.
- \*2: 3 when dividing into zero, 5 when an overflow occurs, and 13 normally.
- \*3: 5 + (a) when dividing into zero, 7 + (a) when an overflow occurs, and 17 + (a) normally.
- \*4: 3 when dividing into zero, 5 when an overflow occurs, and 21 normally.
- \*5: 4 + (a) when dividing into zero, 7 + (a) when an overflow occurs, and 25 + (a) normally.
- \*6: (b) when dividing into zero or when an overflow occurs, and 2 × (b) normally.
- \*7: (c) when dividing into zero or when an overflow occurs, and 2 × (c) normally.
- \*8: 3 when byte (AH) is zero, and 7 when byte (AH) is not 0.
- \*9: 3 when byte (ear) is zero, and 7 when byte (ear) is not 0.
- \*10: 4 + (a) when byte (eam) is zero, and 8 + (a) when byte (eam) is not 0.
- \*11: 3 when word (AH) is zero, and 11 when word (AH) is not 0.
- \*12: 3 when word (ear) is zero, and 11 when word (ear) is not 0.
- \*13: 4 + (a) when word (eam) is zero, and 12 + (a) when word (eam) is not 0.

## MB90F243H

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

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

For an explanation of “(b)” and “(c)”, refer to Table 5, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

- \*1: 3 when dividing into zero, 8 or 18 when an overflow occurs, and 18 normally.
- \*2: 3 when dividing into zero, 10 or 21 when an overflow occurs, and 22 normally.
- \*3: 4 + (a) when dividing into zero, 11 + (a) or 22 + (a) when an overflow occurs, and 23 + (a) normally.
- \*4: When the dividend is positive: 4 when dividing into zero, 10 or 29 when an overflow occurs, and 30 normally.  
When the dividend is negative: 4 when dividing into zero, 11 or 30 when an overflow occurs, and 31 normally.
- \*5: When the dividend is positive: 4 + (a) when dividing into zero, 11 + (a) or 30 + (a) when an overflow occurs, and 31 + (a) normally.  
When the dividend is negative: 4 + (a) when dividing into zero, 12 + (a) or 31 + (a) when an overflow occurs, and 32 + (a) normally.
- \*6: (b) when dividing into zero or when an overflow occurs, and 2 × (b) normally.
- \*7: (c) when dividing into zero or when an overflow occurs, and 2 × (c) normally.
- \*8: 3 when byte (AH) is zero, 12 when the result is positive, and 13 when the result is negative.
- \*9: 3 when byte (ear) is zero, 12 when the result is positive, and 13 when the result is negative.
- \*10: 4 + (a) when byte (eam) is zero, 13 + (a) when the result is positive, and 14 + (a) when the result is negative.
- \*11: 3 when word (AH) is zero, 12 when the result is positive, and 13 when the result is negative.
- \*12: 3 when word (ear) is zero, 16 when the result is positive, and 19 when the result is negative.
- \*13: 4 + (a) when word (eam) is zero, 17 + (a) when the result is positive, and 20 + (a) when the result is negative.

Note: Which of the two values given for the number of execution cycles applies when an overflow error occurs in a DIV or DIVW instruction depends on whether the overflow was detected before or after the operation.

# MB90F243H

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

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

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

# MB90F243H

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

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
ANDL A, ear	2	5	0	long (A) ← (A) and (ear)	-	-	-	-	-	*	*	R	-	-
ANDL A, eam	2+	6+ (a)	(d)	long (A) ← (A) and (eam)	-	-	-	-	-	*	*	R	-	-
ORL A, ear	2	5	0	long (A) ← (A) or (ear)	-	-	-	-	-	*	*	R	-	-
ORL A, eam	2+	6+ (a)	(d)	long (A) ← (A) or (eam)	-	-	-	-	-	*	*	R	-	-
XORL A, ear	2	5	0	long (A) ← (A) xor (ear)	-	-	-	-	-	*	*	R	-	-
XORL A, eam	2+	6+ (a)	(d)	long (A) ← (A) xor (eam)	-	-	-	-	-	*	*	R	-	-

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

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

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

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

**Table 17 Absolute Value Instructions (Byte/Word/Long Word) [3 Instructions]**

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
ABS A	2	2	0	byte (A) ← absolute value (A)	Z	-	-	-	-	*	*	*	-	-
ABSW A	2	2	0	word (A) ← absolute value (A)	-	-	-	-	-	*	*	*	-	-
ABSL A	2	4	0	long (A) ← absolute value (A)	-	-	-	-	-	*	*	*	-	-

**Table 18 Normalize Instructions (Long Word) [1 Instruction]**

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
NRML A, R0	2	*	0	long (A) ← Shifts to the position at which “1” was set first byte (R0) ← current shift count	-	-	-	-	*	-	-	-	-	-

\* : 5 when the contents of the accumulator are all zeroes, 5 + (R0) in all other cases.

# MB90F243H

**Table 19 Shift Instructions (Byte/Word/Long Word) [27 Instructions]**

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
RORC A	2	2	0	byte (A) ← Right rotation with carry	—	—	—	—	—	*	*	—	*	—
ROLC A	2	2	0	byte (A) ← Left rotation with carry	—	—	—	—	—	*	*	—	*	—
RORC ear	2	2	0	byte (ear) ← Right rotation with carry	—	—	—	—	—	*	*	—	*	*
RORC eam	2+	3+ (a)	2× (b)	byte (eam) ← Right rotation with carry	—	—	—	—	—	*	*	—	*	*
ROLC ear	2	2	0	byte (ear) ← Left rotation with carry	—	—	—	—	—	*	*	—	*	*
ROLC eam	2+	3+ (a)	2× (b)	byte (eam) ← Left rotation with carry	—	—	—	—	—	*	*	—	*	*
ASR A, R0	2	*1	0	byte (A) ← Arithmetic right barrel shift (A, R0)	—	—	—	—	*	*	*	—	*	—
LSR A, R0	2	*1	0	byte (A) ← Logical right barrel shift (A, R0)	—	—	—	—	*	*	*	—	*	—
LSL A, R0	2	*1	0	byte (A) ← Logical left barrel shift (A, R0)	—	—	—	—	—	*	*	—	*	—
ASR A, #imm8	3	*3	0	byte (A) ← Arithmetic right barrel shift (A, imm8)	—	—	—	—	*	*	*	—	*	—
LSR A, #imm8	3	*3	0	byte (A) ← Logical right barrel shift (A, imm8)	—	—	—	—	*	*	*	—	*	—
LSL A, #imm8	3	*3	0	byte (A) ← Logical left barrel shift (A, imm8)	—	—	—	—	—	*	*	—	*	—
ASRW A	1	2	0	word (A) ← Arithmetic right shift (A, 1 bit)	—	—	—	—	*	*	*	—	*	—
LSRW A/SHRW A	1	2	0	word (A) ← Logical right shift (A, 1 bit)	—	—	—	—	*	R	*	—	*	—
LSLW A/SHLW A	1	2	0	word (A) ← Logical left shift (A, 1 bit)	—	—	—	—	—	*	*	—	*	—
ASRW A, R0	2	*1	0	word (A) ← Arithmetic right barrel shift (A, R0)	—	—	—	—	*	*	*	—	*	—
LSRW A, R0	2	*1	0	word (A) ← Logical right barrel shift (A, R0)	—	—	—	—	*	*	*	—	*	—
LSLW A, R0	2	*1	0	word (A) ← Logical left barrel shift (A, R0)	—	—	—	—	—	*	*	—	*	—
ASRW A, #imm8	3	*3	0	word (A) ← Arithmetic right barrel shift (A, imm8)	—	—	—	—	*	*	*	—	*	—
LSRW A, #imm8	3	*3	0	word (A) ← Logical right barrel shift (A, imm8)	—	—	—	—	*	*	*	—	*	—
LSLW A, #imm8	3	*3	0	word (A) ← Logical left barrel shift (A, imm8)	—	—	—	—	—	*	*	—	*	—
ASRL A, R0	2	*2	0	long (A) ← Arithmetic right shift (A, R0)	—	—	—	—	*	*	*	—	*	—
LSRL A, R0	2	*2	0	long (A) ← Logical right barrel shift (A, R0)	—	—	—	—	*	*	*	—	*	—
LSLL A, R0	2	*2	0	long (A) ← Logical left barrel shift (A, R0)	—	—	—	—	—	*	*	—	*	—
ASRL A, #imm8	3	*4	0	long (A) ← Arithmetic right shift (A, imm8)	—	—	—	—	*	*	*	—	*	—
LSRL A, #imm8	3	*4	0	long (A) ← Logical right barrel shift (A, imm8)	—	—	—	—	*	*	*	—	*	—
LSLL A, #imm8	3	*4	0	long (A) ← Logical left barrel shift (A, imm8)	—	—	—	—	—	*	*	—	*	—

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

\*1: 3 when R0 is 0, 3 + (R0) in all other cases.

\*2: 3 when R0 is 0, 4 + (R0) in all other cases.

\*3: 3 when imm8 is 0, 3 + (imm8) in all other cases.

\*4: 3 when imm8 is 0, 4 + (imm8) in all other cases.



## MB90F243H

Table 20 Branch 1 Instructions [31 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
BZ/BEQ rel	2	*1	0	Branch when (Z) = 1	-	-	-	-	-	-	-	-	-	-
BNZ/BNE rel	2	*1	0	Branch when (Z) = 0	-	-	-	-	-	-	-	-	-	-
BC/BLO rel	2	*1	0	Branch when (C) = 1	-	-	-	-	-	-	-	-	-	-
BNC/BHS rel	2	*1	0	Branch when (C) = 0	-	-	-	-	-	-	-	-	-	-
BN rel	2	*1	0	Branch when (N) = 1	-	-	-	-	-	-	-	-	-	-
BP rel	2	*1	0	Branch when (N) = 0	-	-	-	-	-	-	-	-	-	-
BV rel	2	*1	0	Branch when (V) = 1	-	-	-	-	-	-	-	-	-	-
BNV rel	2	*1	0	Branch when (V) = 0	-	-	-	-	-	-	-	-	-	-
BT rel	2	*1	0	Branch when (T) = 1	-	-	-	-	-	-	-	-	-	-
BNT rel	2	*1	0	Branch when (T) = 0	-	-	-	-	-	-	-	-	-	-
BLT rel	2	*1	0	Branch when (V) xor (N) = 1	-	-	-	-	-	-	-	-	-	-
BGE rel	2	*1	0	Branch when (V) xor (N) = 0	-	-	-	-	-	-	-	-	-	-
BLE rel	2	*1	0	((V) xor (N)) or (Z) = 1	-	-	-	-	-	-	-	-	-	-
BGT rel	2	*1	0	((V) xor (N)) or (Z) = 0	-	-	-	-	-	-	-	-	-	-
BLS rel	2	*1	0	Branch when (C) or (Z) = 1	-	-	-	-	-	-	-	-	-	-
BHI rel	2	*1	0	Branch when (C) or (Z) = 0	-	-	-	-	-	-	-	-	-	-
BRA rel	2	*1	0	Branch unconditionally	-	-	-	-	-	-	-	-	-	-
JMP @A	1	2	0	word (PC) ← (A)	-	-	-	-	-	-	-	-	-	-
JMP addr16	3	2	0	word (PC) ← addr16	-	-	-	-	-	-	-	-	-	-
JMP @ear	2	3	0	word (PC) ← (ear)	-	-	-	-	-	-	-	-	-	-
JMP @eam	2+	4+ (a)	(c)	word (PC) ← (eam)	-	-	-	-	-	-	-	-	-	-
JMPP @ear *3	2	3	0	word (PC) ← (ear), (PCB) ← (ear+2)	-	-	-	-	-	-	-	-	-	-
JMPP @eam *3	2+	4+ (a)	(d)	word (PC) ← (eam), (PCB) ← (eam+2)	-	-	-	-	-	-	-	-	-	-
JMPP addr24	4	3	0	word (PC) ← ad24 0 to 15 (PCB) ← ad24 16 to 23	-	-	-	-	-	-	-	-	-	-
CALL @ear *4	2	4	(c)	word (PC) ← (ear)	-	-	-	-	-	-	-	-	-	-
CALL @eam *4	2+	5+ (a)	2× (c)	word (PC) ← (eam)	-	-	-	-	-	-	-	-	-	-
CALL addr16 *5	3	5	(c)	word (PC) ← addr16	-	-	-	-	-	-	-	-	-	-
CALLV #vct4 *5	1	5	2× (c)	Vector call instruction	-	-	-	-	-	-	-	-	-	-
CALLP @ear *6	2	7	2× (c)	word (PC) ← (ear) 0 to 15, (PCB) ← (ear) 16 to 23	-	-	-	-	-	-	-	-	-	-
CALLP @eam *6	2+	8+ (a)	*2	word (PC) ← (eam) 0 to 15, (PCB) ← (eam) 16 to 23	-	-	-	-	-	-	-	-	-	-
CALLP addr24 *7	4	7	2× (c)	word (PC) ← addr 0 to 15, (PCB) ← addr 16 to 23	-	-	-	-	-	-	-	-	-	-

For an explanation of "(a)", "(c)" and "(d)", refer to Table 4, "Number of Execution Cycles for Each Form of Addressing," and Table 5, "Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles."

\*1: 3 when branching, 2 when not branching.

\*2:  $3 \times (c) + (b)$

\*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) branch address.

\*7: Save (long word) to stack.

# MB90F243H

**Table 21 Branch 2 Instructions [20 Instructions]**

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
CBNE A, #imm8, rel	3	*1	0	Branch when byte (A) $\neq$ imm8	-	-	-	-	-	*	*	*	*	-
CWBNE A, #imm16, rel	4	*1	0	Branch when byte (A) $\neq$ imm16	-	-	-	-	-	*	*	*	*	-
CBNE ear, #imm8, rel	4	*1	0	Branch when byte (ear) $\neq$ imm8	-	-	-	-	-	*	*	*	*	-
CBNE eam, #imm8, rel	4+	*1	(b)	Branch when byte (eam) $\neq$ imm8	-	-	-	-	-	*	*	*	*	-
CWBNE ear, #imm16, rel	5	*3	0	Branch when word (ear) $\neq$ imm16	-	-	-	-	-	*	*	*	*	-
CWBNE eam, #imm16, rel	5+	*2	(c)	Branch when word (eam) $\neq$ imm16	-	-	-	-	-	*	*	*	*	-
DBNZ ear, rel	3	*4	0	Branch when byte (ear) = (ear) - 1, and (ear) $\neq$ 0	-	-	-	-	-	*	*	*	-	-
DBNZ eam, rel	3+	*2	2 $\times$ (b)	Branch when byte (ear) = (eam) - 1, and (eam) $\neq$ 0	-	-	-	-	-	*	*	*	-	*
DWBNZ ear, rel	3	*4	0	Branch when word (ear) = (ear) - 1, and (ear) $\neq$ 0	-	-	-	-	-	*	*	*	-	-
DWBNZ eam, rel	3+		2 $\times$ (c)	Branch when word (eam) = (eam) - 1, and (eam) $\neq$ 0	-	-	-	-	-	*	*	*	-	*
		14												
		12												
INT #vct8	2	13	8 $\times$ (c)	Software interrupt	-	-	R	S	-	-	-	-	-	-
INT addr16	3	14	6 $\times$ (c)	Software interrupt	-	-	R	S	-	-	-	-	-	-
INTP addr24	4	9	6 $\times$ (c)	Software interrupt	-	-	R	S	-	-	-	-	-	-
INT9	1	11	8 $\times$ (c)	Software interrupt	-	-	R	S	-	-	-	-	-	-
RETI	1		6 $\times$ (c)	Return from interrupt	-	-	*	*	*	*	*	*	*	-
RETIQ *6	2	6	*5	Return from interrupt	-	-	*	*	*	*	*	*	*	-
LINK #imm8	2		(c)	At constant entry, save old frame pointer to stack, set new frame pointer, and allocate local pointer area	-	-	-	-	-	-	-	-	-	-
		5												
UNLINK	1		(c)	At constant entry, retrieve old frame pointer from stack.	-	-	-	-	-	-	-	-	-	-
		4												
		5												
RET *7	1		(c)	Return from subroutine	-	-	-	-	-	-	-	-	-	-
RETP *8	1		(d)	Return from subroutine	-	-	-	-	-	-	-	-	-	-

For an explanation of “(b)”, “(c)” and “(d)”, refer to Table 5, “Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles.”

\*1: 4 when branching, 3 when not branching

\*2: 5 when branching, 4 when not branching

\*3: 5 + (a) when branching, 4 + (a) when not branching

\*4: 6 + (a) when branching, 5 + (a) when not branching

\*5: 3  $\times$  (b) + 2  $\times$  (c) when an interrupt request is generated, 6  $\times$  (c) when returning from the interrupt.

\*6: High-speed interrupt return instruction. When an interrupt request is detected during this instruction, the instruction branches to the interrupt vector without performing stack operations when the interrupt is generated.

\*7: Return from stack (word)

\*8: Return from stack (long word)

## MB90F243H

Table 22 Other Control Instructions (Byte/Word/Long Word) [36 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
PUSHW A	1	3	(c)	word (SP) ← (SP) -2, ((SP)) ← (A)	-	-	-	-	-	-	-	-	-	-
PUSHW AH	1	3	(c)	word (SP) ← (SP) -2, ((SP)) ← (AH)	-	-	-	-	-	-	-	-	-	-
PUSHW PS	1	3	(c)	word (SP) ← (SP) -2, ((SP)) ← (PS)	-	-	-	-	-	-	-	-	-	-
PUSHW rlst	2	*3	*4	(SP) ← (SP) -2n, ((SP)) ← (rlst)	-	-	-	-	-	-	-	-	-	-
POPW A	1	3	(c)	word (A) ← ((SP)), (SP) ← (SP) +2	-	*	-	-	-	-	-	-	-	-
POPW AH	1	3	(c)	word (AH) ← ((SP)), (SP) ← (SP) +2	-	-	-	-	-	-	-	-	-	-
POPW PS	1	3	(c)	word (PS) ← ((SP)), (SP) ← (SP) +2	-	-	*	*	*	*	*	*	*	-
POPW rlst	2	*2	*4	(rlst) ← ((SP)), (SP) ← (SP)	-	-	-	-	-	-	-	-	-	-
JCTX @A	1	9	6× (c)	Context switch instruction	-	-	*	*	*	*	*	*	*	-
AND CCR, #imm8	2	3	0	byte (CCR) ← (CCR) and imm8	-	-	*	*	*	*	*	*	*	-
OR CCR, #imm8	2	3	0	byte (CCR) ← (CCR) or imm8	-	-	*	*	*	*	*	*	*	-
MOV RP, #imm8	2	2	0	byte (RP) ← imm8	-	-	-	-	-	-	-	-	-	-
MOV ILM, #imm8	2	2	0	byte (ILM) ← imm8	-	-	-	-	-	-	-	-	-	-
MOVEA RWi, ear	2	3	0	word (RWi) ← ear	-	-	-	-	-	-	-	-	-	-
MOVEA RWi, eam	2+	2+ (a)	0	word (RWi) ← eam	-	-	-	-	-	-	-	-	-	-
MOVEA A, ear	2	2	0	word (A) ← ear	-	*	-	-	-	-	-	-	-	-
MOVEA A, eam	2+	1+ (a)	0	word (A) ← eam	-	*	-	-	-	-	-	-	-	-
ADDSP #imm8	2	3	0	word (SP) ← ext (imm8)	-	-	-	-	-	-	-	-	-	-
ADDSP #imm16	3	3	0	word (SP) ← imm16	-	-	-	-	-	-	-	-	-	-
MOV A, brgl	2	*1	0	byte (A) ← (brgl)	Z	*	-	-	-	*	*	-	-	-
MOV brg2, A	2	1	0	byte (brg2) ← (A)	-	-	-	-	-	*	*	-	-	-
MOV brg2, #imm8	3	2	0	byte (brg2) ← imm8	-	-	-	-	-	*	*	-	-	-
NOP	1	1	0	No operation	-	-	-	-	-	-	-	-	-	-
ADB	1	1	0	Prefix code for AD space access	-	-	-	-	-	-	-	-	-	-
DTB	1	1	0	Prefix code for DT space access	-	-	-	-	-	-	-	-	-	-
PCB	1	1	0	Prefix code for PC space access	-	-	-	-	-	-	-	-	-	-
SPB	1	1	0	Prefix code for SP space access	-	-	-	-	-	-	-	-	-	-
NCC	1	1	0	Prefix code for no flag change	-	-	-	-	-	-	-	-	-	-
CMR	1	1	0	Prefix code for the common register bank	-	-	-	-	-	-	-	-	-	-
MOVW SPCU, #imm16	4	2	0	word (SPCU) ← (imm16)	-	-	-	-	-	-	-	-	-	-
MOVW SPCL, #imm16	4	2	0	word (SPCL) ← (imm16)	-	-	-	-	-	-	-	-	-	-
SETSPC	2	2	0	Stack check operation enable	-	-	-	-	-	-	-	-	-	-
CLRSPC	2	2	0	Stack check operation disable	-	-	-	-	-	-	-	-	-	-
BTSCN A	2	*5	0	byte (A) ← position of "1" bit in word (A)	Z	-	-	-	-	-	*	-	-	-
BTSCNSA	2	*6	0	byte (A) ← position of "1" bit in word (A) × 2	Z	-	-	-	-	-	*	-	-	-
BTSCNDA	2	*7	0	byte (A) ← position of "1" bit in word (A) × 4	Z	-	-	-	-	-	*	-	-	-

For an explanation of "(a)" and "(c)", refer to Tables 4 and 5.

\*1: PCB, ADB, SSB, USB, and SPB: 1 cycle

DTB: 2 cycles

DPR: 3 cycles

\*2: 3 + 4 × (pop count)

\*3: 3 + 4 × (push count)

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

\*5: 3 when AL is 0, 5 when AL is not 0.

\*6: 4 when AL is 0, 6 when AL is not 0.

\*7: 5 when AL is 0, 7 when AL is not 0.

# MB90F243H

**Table 23 Bit Manipulation Instructions [21 Instructions]**

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

For an explanation of "(b)", refer to Table 5, "Correction Values for Number of Cycles Used to Calculate Number of Actual Cycles."

\*1: 5 when branching, 4 when not branching

\*2: 7 when condition is satisfied, 6 when not satisfied

\*3: Undefined count

\*4: Until condition is satisfied

# MB90F243H

**Table 24 Accumulator Manipulation Instructions (Byte/Word) [6 Instructions]**

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
SWAP	1	3	0	byte (A) 0 to 7 $\leftrightarrow$ (A) 8 to 15	-	-	-	-	-	-	-	-	-	-
SWAPW	1	2	0	word (AH) $\leftrightarrow$ (AL)	-	*	-	-	-	-	-	-	-	-
EXT	1	1	0	Byte code extension	X	-	-	-	-	*	*	-	-	-
EXTW	1	2	0	Word code extension	-	X	-	-	-	*	*	-	-	-
ZEXT	1	1	0	Byte zero extension	Z	-	-	-	-	R	*	-	-	-
ZEXTW	1	2	0	Word zero extension	-	Z	-	-	-	R	*	-	-	-

**Table 25 String Instructions [10 Instructions]**

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

m: RW0 value (counter value)

\*1: 3 when RW0 is 0,  $2 + 6 \times (\text{RW0})$  for count out, and  $6n + 4$  when match occurs

\*2: 4 when RW0 is 0,  $2 + 6 \times (\text{RW0})$  in any other case

\*3:  $(b) \times (\text{RW0})$

\*4:  $(b) \times n$

\*5:  $(b) \times (\text{RW0})$

\*6:  $(c) \times (\text{RW0})$

\*7:  $(c) \times n$

\*8:  $(c) \times (\text{RW0})$

# MB90F243H

**Table 26 Multiple Data Transfer Instructions [18 Instructions]**

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOVM @A, @RLi, #imm8	3	*1	*3	Multiple data transfer byte ((A) ← ((RLi))	—	—	—	—	—	—	—	—	—	—
MOVM @A, eam, #imm8	3+	*2	*3	Multiple data transfer byte ((A) ← (eam)	—	—	—	—	—	—	—	—	—	—
MOVM addr16, @RLi, #imm8	5	*1	*3	Multiple data transfer byte (addr16) ← ((RLi))	—	—	—	—	—	—	—	—	—	—
MOVM addr16, eam, #imm8	5+	*2	*3	Multiple data transfer byte (addr16) ← (eam)	—	—	—	—	—	—	—	—	—	—
MOV MW @A, @RLi, #imm8	3	*1	*4	Multiple data transfer word ((A) ← ((RLi))	—	—	—	—	—	—	—	—	—	—
MOV MW @A, eam, #imm8	3+	*2	*4	Multiple data transfer word ((A) ← (eam)	—	—	—	—	—	—	—	—	—	—
MOV MW addr16, @RLi, #imm8	5	*1	*4	Multiple data transfer word (addr16) ← ((RLi))	—	—	—	—	—	—	—	—	—	—
MOV MW addr16, eam, #imm8	5+	*2	*4	Multiple data transfer word (addr16) ← (eam)	—	—	—	—	—	—	—	—	—	—
MOVM @RLi, @A, #imm8	3	*1	*3	Multiple data transfer byte ((RLi) ← ((A))	—	—	—	—	—	—	—	—	—	—
MOVM eam, @A, #imm8	3+	*2	*3	Multiple data transfer byte (eam) ← ((A))	—	—	—	—	—	—	—	—	—	—
MOVM @RLi, addr16, #imm8	5	*1	*3	Multiple data transfer byte ((RLi) ← (addr16)	—	—	—	—	—	—	—	—	—	—
MOVM eam, addr16, #imm8	5+	*2	*3	Multiple data transfer byte (eam) ← (addr16)	—	—	—	—	—	—	—	—	—	—
MOV MW @RLi, @A, #imm8	3	*1	*4	Multiple data transfer word ((RLi) ← ((A))	—	—	—	—	—	—	—	—	—	—
MOV MW eam, @A, #imm8	3+	*2	*4	Multiple data transfer word (eam) ← ((A))	—	—	—	—	—	—	—	—	—	—
MOV MW @RLi, addr16, #imm8	5	*1	*4	Multiple data transfer word ((RLi) ← (addr16)	—	—	—	—	—	—	—	—	—	—
MOV MW eam, addr16, #imm8	5+	*2	*4	Multiple data transfer word (eam) ← (addr16)	—	—	—	—	—	—	—	—	—	—
MOVM bnk : addr16, *5	7	*1	*3	Multiple data transfer	—	—	—	—	—	—	—	—	—	—
bnk : addr16, #imm8				byte (bnk:addr16) ← (bnk:addr16)										
MOV MW bnk : addr16, *5	7	*1	*4	Multiple data transfer	—	—	—	—	—	—	—	—	—	—
bnk : addr16, #imm8				word (bnk:addr16) ← (bnk:addr16)										

\*1:  $5 + \text{imm8} \times 5$ , 256 times when imm8 is zero.

\*2:  $5 + \text{imm8} \times 5 + (a)$ , 256 times when imm8 is zero.

\*3: Number of transfers  $\times (b) \times 2$

\*4: Number of transfers  $\times (c) \times 2$

\*5: The bank register specified by "bnk" is the same as for the MOVS instruction.

# MB90F243H

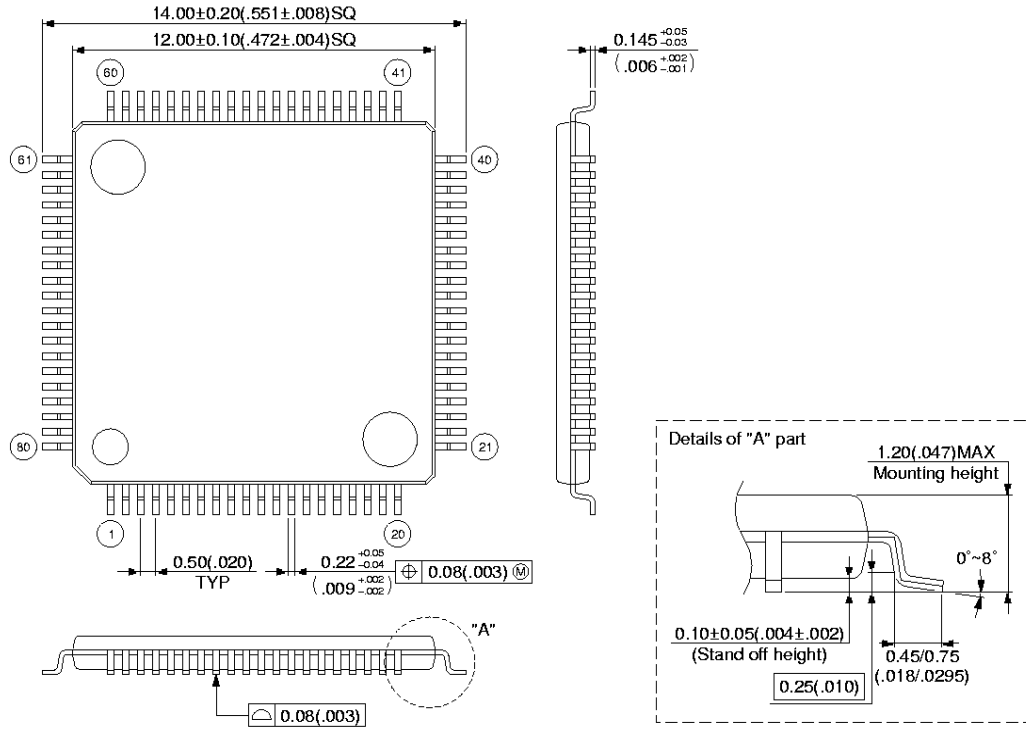
## ■ ORDERING INFORMATION

Part number	Package	Remarks
MB90F243HPFT-G-BND MB90F243HPFT-ES-BND	80-pin Plastic TQFP (FPT-80P-M15)	

# MB90F243H

## ■ PACKAGE DIMENSIONS

80-pin Plastic TQFP  
(FPT-80P-M15)



© 1997 FUJITSU LIMITED F80028S-1C-1

Dimensions in mm (inches)



# FUJITSU LIMITED

*For further information please contact:*

## Japan

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

<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 MIKROELEKTRONIK GmbH  
Am Siebenstein 6-10  
D-63303 Dreieich-Buchsschlag  
Germany  
Tel: (06103) 690-0  
Fax: (06103) 690-122

<http://www.fujitsu-edc.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/>

F9803

© FUJITSU LIMITED Printed in Japan

All Rights Reserved.

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

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

FUJITSU semiconductor devices are intended for use in standard applications (computers, office automation and other office equipment, industrial, communications, and measurement equipment, personal or household devices, etc.).

### CAUTION:

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

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

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