

16-bit Proprietary Microcontroller

CMOS

F²MC-16LX MB90540/545 Series

MB90543/F543/549/F549/V540

■ DESCRIPTION

The MB90540/545 series with FULL-CAN and FLASH ROM is specially designed for automotive and industrial applications. Its main features are two on board CAN Interfaces (one for MB90V545 series), which conform to V2.0 Part A and Part B, supporting very flexible message buffering. Thus, offering more functions than a normal full CAN approach. In the new 0.5 μ m Technology Fujitsu now also offer FLASH-ROM. An internal voltage booster substitutes the necessity of a second programming voltage.

An on board voltage regulator provides 3V to the internal MCU core. This constitutes a major advantage in terms of EMI and power consumption.

The internal PLL clock frequency multiplier, provides an internal 62.5 nsec instruction cycle time with an external 4 MHz clock.

Further more it features 4 channels Output Capture Units and 8 channels Input Capture Units with a 16-bit free running timer. Two UARTs constitute additional functionality for communication purposes.

The external bus interface allows full use to be made of the 16MByte address space.

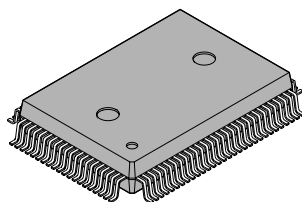
■ FEATURES

- 16-bit core CPU : 4MHz external clock (16 MHz internal, 62.5 nsec instr. cycle time)
- 32 kHz Subsystem Clock
- New 0.5 μ m CMOS Process Technology
- Internal voltage regulator supports 3V MCU core, offering low EMI and low power consumption figures
- FULL-CAN interfaces (MB90540 series : 2 interf., MB90545 series : 1 interf.); conform to Version 2.0 Part A and Part B, flexible message buffering (mailbox and FIFO buffering can be mixed)

(Continued)

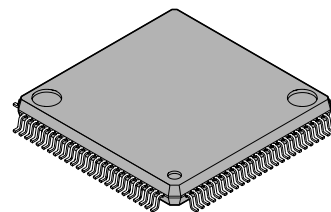
■ PACKAGE

100-pin Plastic QFP



(FPT-100P-M06)

100-pin Plastic LQFP



(FPT-100P-M05)

MB90540/545 Series

(Continued)

- Powerful interrupt functions (8 progr. priority levels; 8 external interrupts)
- EI²OS - Automatic transfer function indep.of CPU
- 18-bit Time-base counter
- Watchdog Timer
- 2 full duplex UARTs; UART0 supports 10.4 KBAud (USA standard), UART 1 also for serial transfer with clock (SCI) programmable
- Serial I/O: 1ch for synchronous data transfer
- A/D Converter: 8 ch. analog inputs (Resolution 10 bits or 8 bits)
- 16-bit reload timer * 2ch
- ICU (Input capture) 16bit * 8 ch
- OCU (Output capture) 16bit * 4ch
- 16-bit Programmable Pulse Generator 4ch
- External bus interface
- Optimized instruction set for controller applications (bit, byte, word and long-word data types; 23 different addressing modes; barrel shift; variety of pointers)
- 4-byte instruction execution queue
- signed multiply (16bit*16bit) and divide (32bit/16bit) instructions available
- Program Patch Function
- Fast Interrupt processing
- Low Power Consumption - 10 different power saving modes: (Sleep, Stop, CPU intermittent mode, Hardware standby,...)
- Package: 100-pin plastic QFP

Controller Area Network (CAN) - License of Robert Bosch GmbH

MB90540/545 Series

■ PRODUCT LINEUP

The following table provides a quick outlook of the MB90540/545 Series

Features	MB90V540	MB90F543/F549	MB90543/549
CPU	F ² MC-16LX CPU		
System clock	On-chip PLL clock multiplier (× 1, × 2, × 3, × 4, 1/2 when PLL stop) Minimum instruction execution time: 62.5 ns (4 MHz osc. PLL × 4)		
ROM	External	Boot-block Flash memory 128 K/256 Kbytes	Mask ROM 128 K/256 Kbytes
RAM	8 Kbytes	6 Kbytes	6 Kbytes
Technology	0.5 μm CMOS with on-chip voltage regulator for internal power supply	0.5 μm CMOS with on-chip voltage regulator for internal power supply + Flash memory On-chip charge pump for programming voltage	0.5 μm CMOS with on-chip voltage regulator for internal power supply
Operating voltage range	5 V±10 %		
Temperature range	– 40 to 85 °C		
Package	PGA-256	QFP100	
UART0	Full duplex double buffer Supports asynchronous/synchronous (with start/stop bit) transfer Baud rate: 4808/5208/9615/10417/19230/38460/62500/500000 bps (asynchronous) 500K/1M/2Mbps (synchronous) at System clock = 16 MHz		
UART1(SCI)	Full duplex double buffer Asynchronous (start-stop synchronized) and CLK-synchronous communication Baud rate: 1202/2404/4808/9615/31250 bps (asynchronous) 62.5K/12K/250K/500K/1 Mbps (synchronous) at 6,8,10,12,16 MHz		
Serial IO	Transfer can be started from MSB or LSB Supports internal clock synchronized transfer and external clock synchronized transfer Supports positive-edge and negative-edge clock synchronization Baud rate : 31.25K/62.5K/125K/500K/1Mbps at System clock = 16MHz		
A/D Converter	10-bit or 8-bit resolution 8 input channels Conversion time: 26.3 μs (per one channel)		
16-bit Reload Timer (2 channels)	Operation clock frequency: $f_{sys}/2^1$, $f_{sys}/2^3$, $f_{sys}/2^5$ (f_{sys} = System clock frequency) Supports External Event Count function		
16-bit IO Timer	Signals an interrupt when overflow Supports Timer Clear when a match with Output Compare(Channel 0) Operation clock freq.: $f_{sys}/2^2$, $f_{sys}/2^4$, $f_{sys}/2^6$, $f_{sys}/2^8$ (f_{sys} = System clock freq.)		
16-bit Output Compare (4 channels)	Signals an interrupt when a match with 16-bit IO Timer Four 16-bit compare registers A pair of compare registers can be used to generate an output signal		

(Continued)

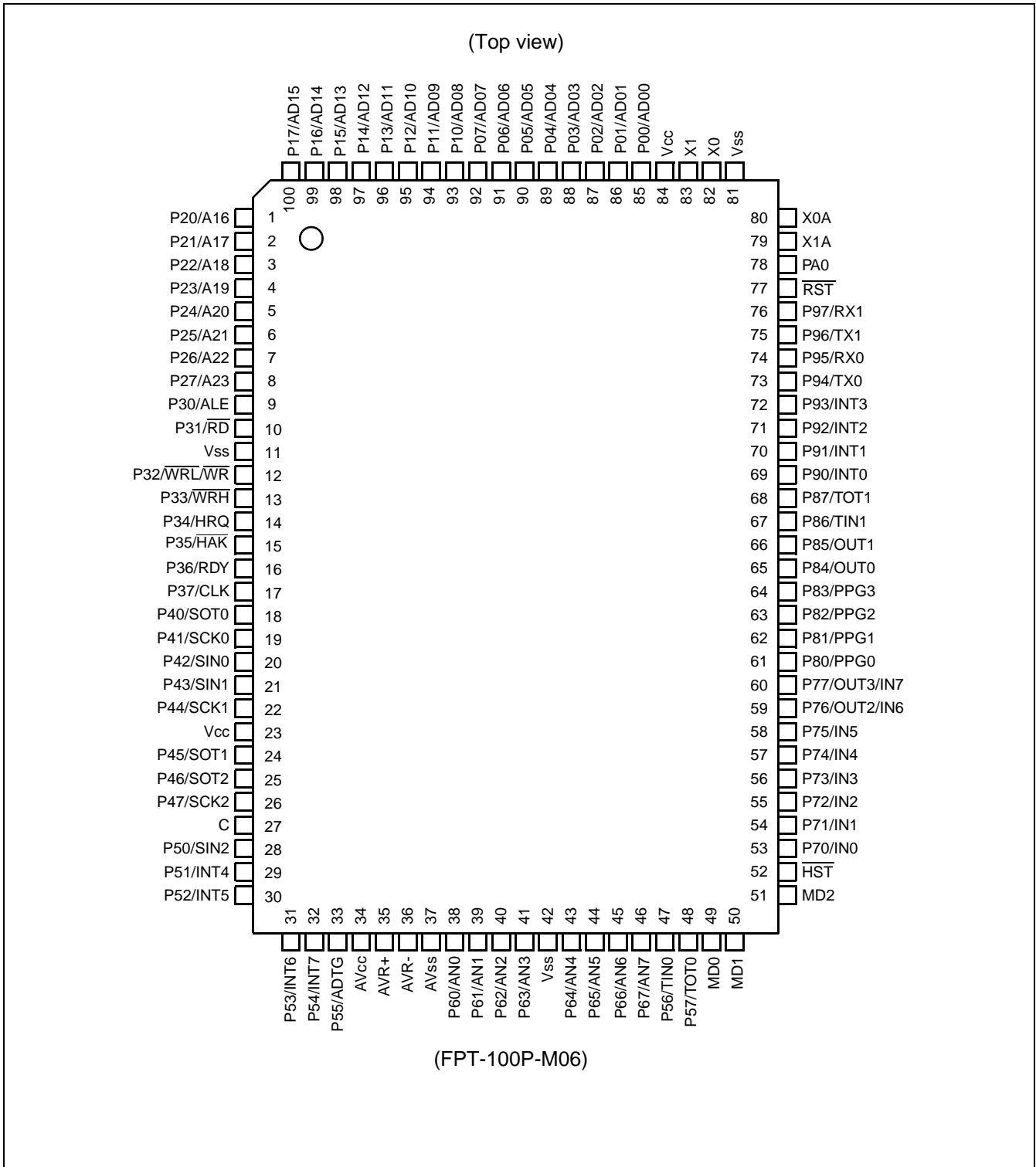
MB90540/545 Series

(Continued)

Features	MB90V540	MB90F543/F549	MB90543/549
16-bit Input Capture (8 channels)	Rising edge, falling edge or rising & falling edge sensitive Four 16-bit Capture registers Signals an interrupt upon external event		
8/16-bit Programmable Pulse Generator (4 channels)	Supports 8-bit and 16-bit operation modes Eight 8-bit reload counters Eight 8-bit reload registers for L pulse width Eight 8-bit reload registers for H pulse width A pair of 8-bit reload counters can be configured as one 16-bit reload counter or as 8-bit prescaler plus 8-bit reload counter 4 output pins Operation clock freq.: f_{sys} , $f_{sys}/2^1$, $f_{sys}/2^2$, $f_{sys}/2^3$, $f_{sys}/2^4$ or $128\mu s @ f_{osc}=4MHz$ (f_{sys} = System clock frequency, f_{osc} = Oscillation clock frequency)		
CAN Interface 540 series: 2 channels 545 series: 1 channel	Conforms to CAN Specification Version 2.0 Part A and B Automatic re-transmission in case of error Automatic transmission responding to Remote Frame Prioritized 16 message buffers for data and ID's Supports multiple messages Flexible configuration of acceptance filtering: Full bit compare / Full bit mask / Two partial bit masks Supports up to 1Mbps		
32 kHz Subclock	Sub-clock for low power operation		
External Interrupt (8 channels)	Can be programmed edge sensitive or level sensitive		
IO Ports	Virtually all external pins can be used as general purpose IO All push-pull outputs and schmitt trigger inputs Bit-wise programmable as input/output or peripheral signal		
Flash Memory	—	Supports automatic programming, Embedded Algorithm™ *1 Write/Erase/Erase-Suspend/Resume commands A flag indicating completion of the algorithm Number of erase cycles: 10,000 times Data retention time: 10 years Flash Writer from Minato Electronics Inc. Boot block configuration Erase can be performed on each block Block protection with external programming voltage Flash Security Feature: protects the content of the Flash memory	—

*1: Embedded Algorithm is a trade mark of Advanced Micro Devices Inc.

■ PIN ASSIGNMENT



MB90540/545 Series

■ PIN DESCRIPTION

No.	Pin name	Circuit type	Function
82 83	X0 X1	A (Oscillation)	High speed oscillator input pins
80 79	X0A X1A	A (Oscillation)	Low speed oscillator input pins
77	$\overline{\text{RST}}$	B	External reset request input
52	$\overline{\text{HST}}$	C	Hardware standby input
85 to 92	P00 to P07	I	General I/O port with programmable pullup. This function is enabled in the single-chip mode.
	AD00 to AD07		I/O pins for 8 lower bits of the external address/data bus. This function is enabled when the external bus is enabled.
93 to 100	P10 to P17	I	General I/O port with programmable pullup. This function is enabled in the single-chip mode.
	AD08 to AD15		I/O pins for 8 higher bits of the external address/data bus. This function is enabled when the external bus is enabled.
1 to 8	P20 to P27	H	General I/O port with programmable pullup. This function is enabled in the single-chip mode.
	A16 to A23		Output pins for A16 to A23 of the external address bus. This function is enabled when the external bus is enabled.
9	P30	I	General I/O port with programmable pullup. This function is enabled in the single-chip mode.
	ALE		Address latch enable output pin. This function is enabled when the external bus is enabled.
10	P31	I	General I/O port with programmable pullup. This function is enabled in the single-chip mode.
	$\overline{\text{RD}}$		Read strobe output pin for the data bus. This function is enabled when the external bus is enabled.
12	P32	I	General I/O port with programmable pullup. This function is enabled in the single-chip mode or when the $\overline{\text{WR}}/\overline{\text{WRL}}$ pin output is disabled.
	$\overline{\text{WRL}}$		Write strobe output pin for the data bus. This function is enabled when both the external bus and the $\overline{\text{WR}}/\overline{\text{WRL}}$ pin output are enabled. $\overline{\text{WRL}}$ is used to write-strobe 8 lower bits of the data bus in 16-bit access while $\overline{\text{WR}}$ is used to write-strobe 8 bits of the data bus in 8-bit access.
	$\overline{\text{WR}}$		
13	P33	I	General I/O port with programmable pullup. This function is enabled in the single-chip mode or external bus 8-bit mode or when $\overline{\text{WRH}}$ pin output is disabled.
	$\overline{\text{WRH}}$		Write strobe output pin for the 8 higher bits of the data bus. This function is enabled when the external bus is enabled, when the external bus 16-bit mode is selected, and when the $\overline{\text{WRH}}$ output pin is enabled.

MB90540/545 Series

No.	Pin name	Circuit type	Function
14	P34	I	General I/O port with programmable pullup. This function is enabled in the single-chip mode or when hold function is disabled.
	HRQ		Hold request input pin. This function is enabled when both the external bus and the hold function are enabled.
15	P35	I	General I/O port with programmable pullup. This function is enabled in the single-chip mode or when hold function is disabled.
	$\overline{\text{HAK}}$		Hold acknowledge output pin. This function is enabled when both the external bus and the hold function are enabled.
16	P36	I	General I/O port with programmable pullup. This function is enabled in the single-chip mode or when the external ready function is disabled.
	RDY		Ready input pin. This function is enabled when both the external bus and the external ready function are enabled.
17	P37	H	General I/O port with programmable pullup. This function is enabled in the single-chip mode or when the clock output is disabled.
	CLK		CLK output pin. This function is enabled when both the external bus and CLK output are enabled.
18	P40	G	General I/O port. This function is enabled when UART0 disables serial data output.
	SOT0		Serial data output pin for UART0. This function is enabled when UART0 enables serial data output.
19	P41	G	General I/O port. This function is enabled when UART0 disables clock output.
	SCK0		Clock I/O pin for UART0. This function is enabled when UART0 enables clock output.
20	P42	G	General I/O port. This function is always enabled.
	SIN0		Serial data input pin for UART0. While UART0 is operating for input, the input of the pin is used as required. Except when the function is intentionally used, output from the other functions must be stopped.
21	P43	G	General I/O port. This function is always enabled.
	SIN1		Serial data input pin for UART1. While UART1 is operating for input, the input of the pin is used as required. Except when the function is intentionally used, output from the other functions must be stopped.
22	P44	G	General I/O port. This function is enabled when UART1 disables clock output.
	SCK1		Clock pulse input/output pin for UART1. This function is enabled when UART1 enables clock output.
24	P45	G	General I/O port. This function is enabled when UART1 disables serial data output.
	SOT1		Serial data output pin for UART1. This function is enabled when UART1 enables serial data output.

MB90540/545 Series

No.	Pin name	Circuit type	Function
25	P46	G	General I/O port. This function is enabled when the Serial IO disables serial data output.
	SOT2		Serial data output pin for the Serial IO. This function is enabled when the Serial IO enables serial data output.
26	P47	G	General I/O port. This function is enabled when the Serial IO disables clock output.
	SCK2		Clock pulse input/output pin for the Serial IO. This function is enabled when the Serial IO enables clock output.
28	P50	D	General I/O port. This function is always enabled.
	SIN2		Serial data input pin for the Serial IO. While the Serial IO is operating for input, the input of the pin is used as required. Except when the function is intentionally used, output from the other functions must be stopped.
29 to 32	P51 to P54	D	General I/O port. This function is always enabled.
	INT4 to INT7		External interrupt request input pins for INT4 to INT7. While external interrupt is allowed, the input of the pin is used as required. Except when the function is intentionally used, output from the other functions must be stopped.
33	P55	D	General I/O port. This function is always enabled.
	ADTG		Trigger input pin for the A/D converter. While the A/D converter is operating for input, the input of the pin is used as required. Except when the function is intentionally used, output from the other functions must be stopped.
38 to 41	P60 to P63	E	General I/O port. The function is enabled when the analog input enable register specifies port.
	AN0 to AN3		Analog input pins for the A/D converter. This function is enabled when the analog input enable register specifies AD.
43 to 46	P64 to P67	E	General I/O port. The function is enabled when the analog input enable register specifies port.
	AN4 to AN7		Analog input pins for the A/D converter. This function is enabled when the analog input enable register specifies AD.
47	P56	D	General I/O port. This function is always enabled.
	TIN0		Event input pin for the reload timers 0. While the reload timer is operating for input, the input of the pin is used as required. Except when the function is intentionally used, output from the other functions must be stopped.
48	P57	D	General I/O port. This function is enabled when the reload timers 0 disables output.
	TOT0		Output pin for the reload timers 0. This function is enabled when the reload timers 0 enables output.

MB90540/545 Series

No.	Pin name	Circuit type	Function
53 to 58	P70 to P75	D	General I/O ports. This function is always enabled.
	IN0 to IN5		Data sample input pins for input captures ICU0 to ICU5. While the ICU is for input, the input of the pin is used as required. Except when the function is intentionally used, output from the other functions must be stopped.
59 to 60	P76 to P77	D	General I/O ports. This function is enabled when the OCU disables waveform output.
	OUT2 to OUT3		Waveform output pins for output compares OCU2 and OCU3. This function is enabled when the OCU enables waveform output.
	IN6 to IN7		Data sample input pin for input captures ICU6 and ICU7. While the ICU is for input, the input of the pin is used as required. Except when the function is intentionally used, output from the other functions must be stopped.
61 to 64	P80 to P83	D	General I/O ports. This function is enabled when PPG disables waveform output.
	PPG0 to PPG3		Output pins for PPGs. This function is enabled when PPG enables waveform output.
65 to 66	P84 to P85	D	General I/O ports. This function is enabled when the OCU disables waveform output.
	OUT0 to OUT1		Waveform output pins for output compares OCU0 and OCU1. This function is enabled when the OCU enables waveform output.
67	P86	D	General I/O port. This function is always enabled.
	TIN1		Event input pin for the reload timers 1. While the reload timer is operating for input, the input of the pin is used as required. Except when the function is intentionally used, output from the other functions must be stopped.
68	P87	D	General I/O port. This function is enabled when the reload timers 0 disables output.
	TOT1		Output pin for the reload timers 1 This function is enabled when the reload timers 1 enables output.
69 to 72	P90 to P93	D	General I/O port. This function is always enabled.
	INT0 to INT3		External interrupt request input pins for INT0 to INT3. While external interrupt is allowed, the input of the pin is used as required. Except when the function is intentionally used, output from the other functions must be stopped.
73	P94	D	General I/O port. This function is enabled when CAN0 disables output.
	TX0		TX Output pin for CAN0. This function is enabled when CAN0 enables output.
74	P95	D	General I/O port. This function is always enabled.
	RX0		RX input pin for CAN0 Interface. When the CAN function is used, output from the other functions must be stopped.

MB90540/545 Series

No.	Pin name	Circuit type	Function
75	P96	D	General I/O port. This function is enabled when CAN1 disables output.
	TX1		TX Output pin for CAN1. This function is enabled when CAN1 enables output (only MB90540 series).
76	P97	D	General I/O port. This function is always enabled.
	RX1		RX input pin for CAN1 Interface. When the CAN function is used, output from the other functions must be stopped (only MB90540 series).
78	PA0	D	General I/O port. This function is always enabled.
34	AVCC	Power supply	Power supply for the A/D Converter. This power supply must be turned on or off while a voltage higher than or equal to AVcc is applied to Vcc.
37	AVSS	Power supply	Dedicated ground pin for the A/D Converter
35	AVR+	Power supply	Reference voltage input for the A/D Converter. This power supply must be turned on or off while a voltage higher than or equal to AVR+ is applied to AVcc.
36	AVR-	Power supply	Lower reference voltage input for the A/D Converter
49 50	MD0 MD1	C	Input pins for specifying the operating mode. The pins must be directly connected to Vcc or Vss.
51	MD2	F	Input pin for specifying the operating mode. The pin must be directly connected to Vcc or Vss.
27	C		This is the power supply stabilization capacitor pin. It should be connected externally to an 0.1 μ F ceramic capacitor.
23; 84	Vcc	Power supply	Power supply for digital circuits
11; 42 81	Vss	Power supply	Ground for digital circuits

■ I/O CIRCUIT TYPE

Circuit type	Diagram	Remarks
A		<ul style="list-style-type: none"> • Oscillation feedback resistor: 1 MΩ approx.
B		<ul style="list-style-type: none"> • Hysteresis input with pull-up Resistor: 50 kΩ approx.
C		<ul style="list-style-type: none"> • Hysteresis input
D		<ul style="list-style-type: none"> • CMOS output • Hysteresis input

MB90540/545 Series

Circuit type	Diagram	Remarks
E		<ul style="list-style-type: none"> • CMOS output • Hysteresis input • Analog input
F		<ul style="list-style-type: none"> • Hysteresis input • Pull-down Resistor: 50 kΩ approx. (except FLASH devices)
G		<ul style="list-style-type: none"> • CMOS output • Hysteresis input • TTL input (FLASH devices only)

MB90540/545 Series

Circuit type	Diagram	Remarks
H		<ul style="list-style-type: none"> • CMOS output • Hysteresis input • Programmable pullup resistor: 50 kΩ approx.
I		<ul style="list-style-type: none"> • CMOS output • Hysteresis input • TTL input (FLASH devices only) • Programmable pullup resistor: 50 kΩ approx.

MB90540/545 Series

■ HANDLING DEVICES

(1) Preventing latch-up

CMOS IC chips may suffer latch-up under the following conditions:

- A voltage higher than V_{cc} or lower than V_{ss} is applied to an input or output pin.
- A voltage higher than the rated voltage is applied between V_{cc} and V_{ss} .
- The AV_{cc} power supply is applied before the V_{cc} voltage.

Latch-up may increase the power supply current drastically, causing thermal damage to the device.

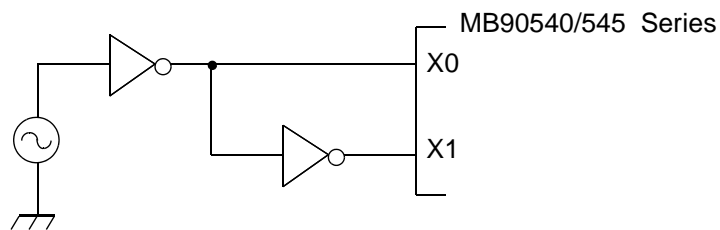
(2) Handling unused input pins

Do not leave unused input pins open, as doing so may cause misoperation of the device. Use a pull-up or pull-down resistor.

(3) Using external clock

To use external clock, drive the X0 and X1 pins in reverse phase.

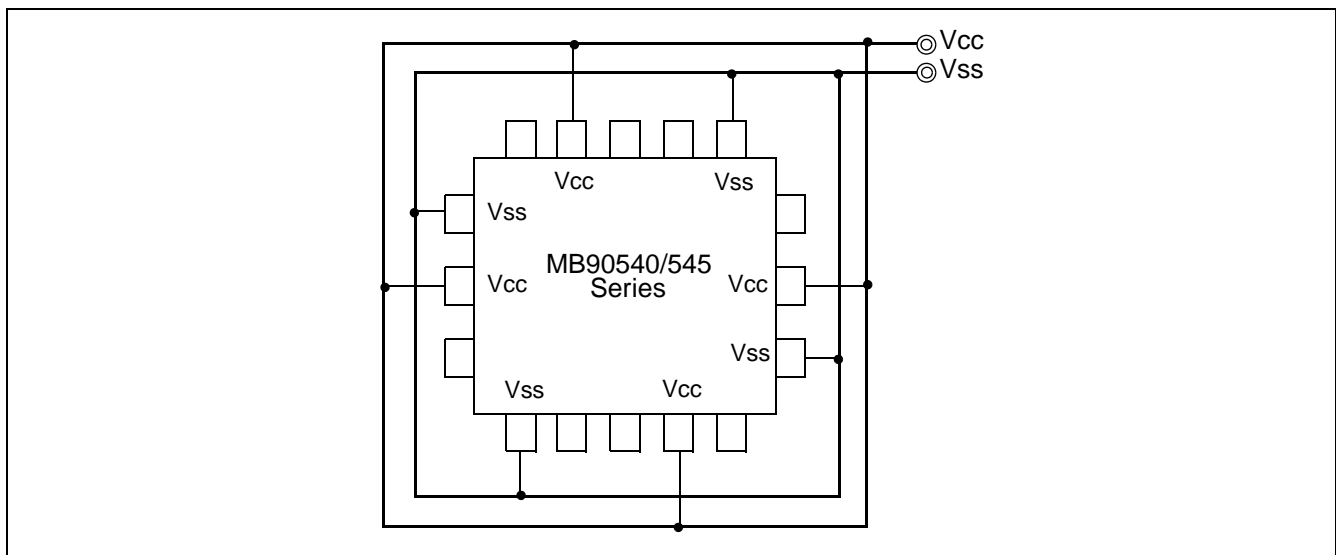
Below is a diagram of how to use external clock.



Using external clock

(4) Power supply pins (V_{cc}/V_{ss})

Ensure that all V_{cc} -level power supply pins are at the same potential. In addition, ensure the same for all V_{ss} -level power supply pins. (See the figure below.) If there are more than one V_{cc} or V_{ss} system, the device may operate incorrectly even within the guaranteed operating range.



(5) Pull-up/down resistors

The MB90540/545 Series does not support internal pull-up/down resistors(except Port0 - Port3:pull-up resistors). Use external components where needed.

(6) Crystal Oscillator Circuit

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

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

(7) Turning-on Sequence of Power Supply to A/D Converter and Analog Inputs

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

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

(8) Connection of Unused Pins of A/D Converter

Connect unused pins of A/D converter to $AV_{CC} = V_{CC}$, $AV_{SS} = AVR_{+} = V_{SS}$.

(9) N.C. Pin

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

(10) Notes on Energization

To prevent the internal regulator circuit from malfunctioning, set the voltage rise time during energization at 50 or more ms (0.2 V to 2.7 V).

(11) Initialization

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

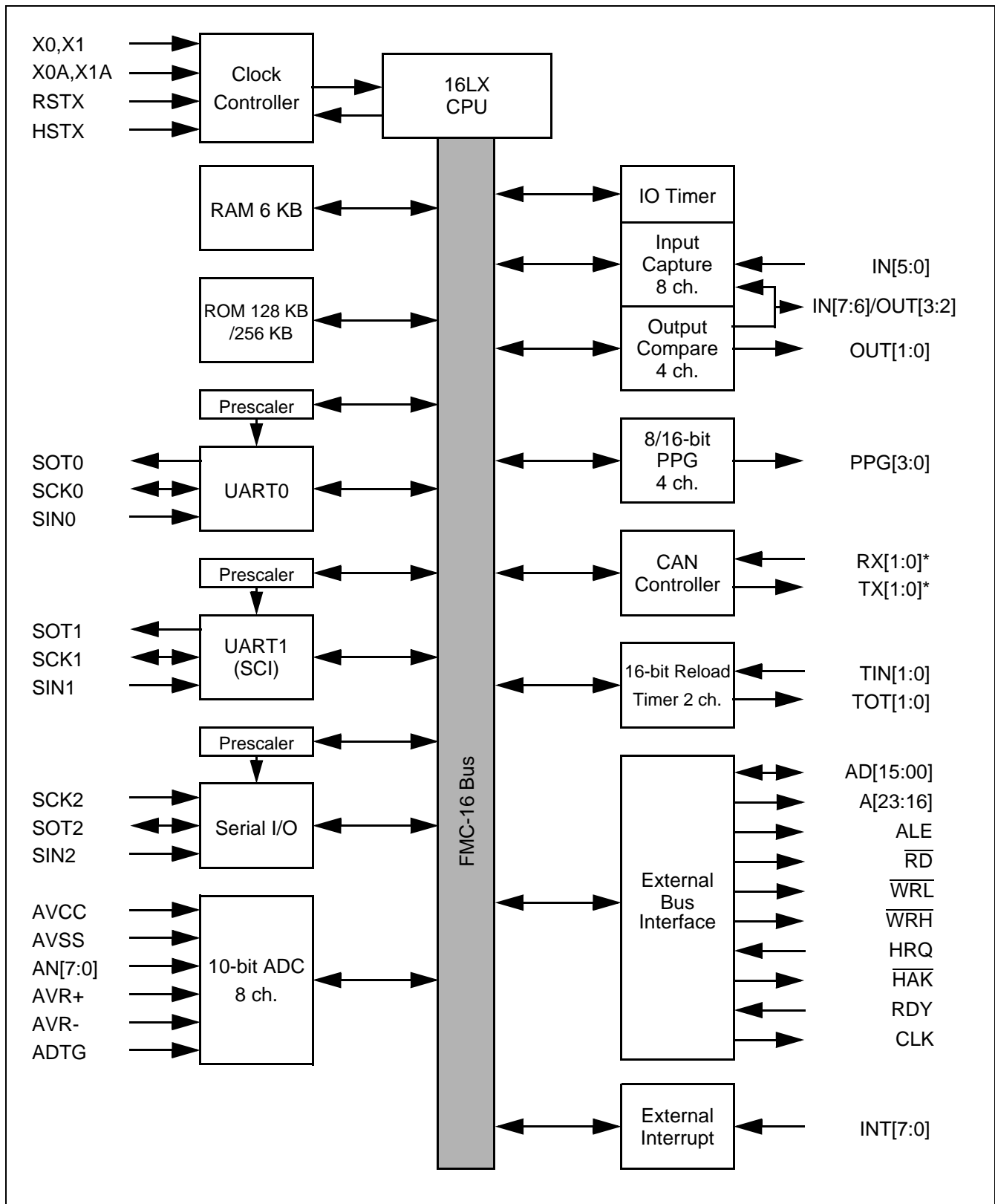
(12) Directions of “DIV A, Ri” and “DIVW A, RWi” instructions

In the Signed multiplication and division instructions (“DIV A, Ri” and “DIVW A, RWi”), the value of the corresponding bank register (DTB, ADB, USB, SSB) is set in “00h”.

If the values of the corresponding bank register (DTB,ADB,USB,SSB) are setting other than “00h”, the remainder by the execution result of the instruction is not stored in the register of the instruction operand.

MB90540/545 Series

■ BLOCK DIAGRAM



MB90540/545 Series

■ MEMORY SPACE

The memory space of the MB90540/545 Series is shown below

MB90V540		MB90543/F543		MB90549/F549	
FFFFFF _H	ROM (FF bank)	FFFFFF _H	ROM (FF bank)	FFFFFF _H	ROM (FF bank)
FF0000 _H		FF0000 _H		FF0000 _H	
FEFFFF _H	ROM (FE bank)	FEFFFF _H	ROM (FE bank)	FEFFFF _H	ROM (FE bank)
FE0000 _H		FE0000 _H		FE0000 _H	
FDFFFF _H	ROM (FD bank)		External	FDFFFF _H	ROM (FD bank)
FD0000 _H				FD0000 _H	
FCFFFF _H	ROM (FC bank)			FCFFFF _H	ROM (FC bank)
FC0000 _H				FC0000 _H	
	External				External
00FFFF _H	ROM (Image of FF bank)	00FFFF _H	ROM (Image of FF bank)	00FFFF _H	ROM (Image of FF bank)
004000 _H		004000 _H		004000 _H	
003FFF _H	Peripheral	003FFF _H	Peripheral	003FFF _H	Peripheral
003900 _H		003900 _H		003900 _H	
	External	002000 _H	External	002000 _H	External
0020FF _H					
001FF5 _H	ROM correction	0018FF _H		0018FF _H	
001FF0 _H			RAM 6K		RAM 6K
	RAM 8K				
000100 _H		000100 _H		000100 _H	
	External		External		External
0000BF _H	Peripheral	0000BF _H	Peripheral	0000BF _H	Peripheral
000000 _H		000000 _H		000000 _H	

Memory space map

The high-order portion of bank 00 gives the image of the FF bank ROM to make the small model of the C compiler effective. Since the low-order 16 bits are the same, the table in ROM can be referenced without using the far specification in the pointer declaration.

For example, an attempt to access 00C000_H accesses the value at FFC000_H in ROM.

The ROM area in bank FF exceeds 48 Kbytes, and its entire image cannot be shown in bank 00.

The image between FF4000_H and FFFFFFF_H is visible in bank 00, while the image between FF0000_H and FF3FFF_H is visible only in bank FF.

MB90540/545 Series

■ I/O MAP

Address	Register	Abbreviation	Access	Peripheral	Initial value
00 _H	Port 0 data register	PDR0	R/W	Port 0	XXXXXXXX _B
01 _H	Port 1 data register	PDR1	R/W	Port 1	XXXXXXXX _B
02 _H	Port 2 data register	PDR2	R/W	Port 2	XXXXXXXX _B
03 _H	Port 3 data register	PDR3	R/W	Port 3	XXXXXXXX _B
04 _H	Port 4 data register	PDR4	R/W	Port 4	XXXXXXXX _B
05 _H	Port 5 data register	PDR5	R/W	Port 5	XXXXXXXX _B
06 _H	Port 6 data register	PDR6	R/W	Port 6	XXXXXXXX _B
07 _H	Port 7 data register	PDR7	R/W	Port 7	XXXXXXXX _B
08 _H	Port 8 data register	PDR8	R/W	Port 8	XXXXXXXX _B
09 _H	Port 9 data register	PDR9	R/W	Port 9	XXXXXXXX _B
0A _H	Port A data register	PDRA	R/W	Port A	_____X _B
0B _H to 0F _H	Reserved				
10 _H	Port 0 direction register	DDR0	R/W	Port 0	0 0 0 0 0 0 0 0 _B
11 _H	Port 1 direction register	DDR1	R/W	Port 1	0 0 0 0 0 0 0 0 _B
12 _H	Port 2 direction register	DDR2	R/W	Port 2	0 0 0 0 0 0 0 0 _B
13 _H	Port 3 direction register	DDR3	R/W	Port 3	0 0 0 0 0 0 0 0 _B
14 _H	Port 4 direction register	DDR4	R/W	Port 4	0 0 0 0 0 0 0 0 _B
15 _H	Port 5 direction register	DDR5	R/W	Port 5	0 0 0 0 0 0 0 0 _B
16 _H	Port 6 direction register	DDR6	R/W	Port 6	0 0 0 0 0 0 0 0 _B
17 _H	Port 7 direction register	DDR7	R/W	Port 7	0 0 0 0 0 0 0 0 _B
18 _H	Port 8 direction register	DDR8	R/W	Port 8	0 0 0 0 0 0 0 0 _B
19 _H	Port 9 direction register	DDR9	R/W	Port 9	0 0 0 0 0 0 0 0 _B
1A _H	Port A direction register	DDRA	R/W	Port A	_____0 _B
1B _H	Analog Input Enable	ADER	R/W	Port 6, A/D	1 1 1 1 1 1 1 1 _B
1C _H	Port 0 Pullup control register	PUCR0	R/W	Port 0	0 0 0 0 0 0 0 0 _B
1D _H	Port 1 Pullup control register	PUCR1	R/W	Port 1	0 0 0 0 0 0 0 0 _B
1E _H	Port 2 Pullup control register	PUCR2	R/W	Port 2	0 0 0 0 0 0 0 0 _B
1F _H	Port 3 Pullup control register	PUCR3	R/W	Port 3	0 0 0 0 0 0 0 0 _B
20 _H	Serial Mode Control Register 0	UMC0	R/W	UART0	0 0 0 0 0 1 0 0 _B
21 _H	Status Register 0	USR0	R/W		0 0 0 1 0 0 0 0 _B
22 _H	Input/Output Data Register 0	UIDR0/ UODR0	R/W		XXXXXXXX _B
23 _H	Rate and Data Register 0	URD0	R/W		0 0 0 0 0 0 0 X _B

(Continued)

MB90540/545 Series

Address	Register	Abbreviation	Access	Peripheral	Initial value
24 _H	Serial Mode Register 1	SMR1	R/W	UART1	0 0 0 0 0 0 0 0 _B
25 _H	Serial Control Register 1	SCR1	R/W		0 0 0 0 0 1 0 0 _B
26 _H	Input/Output Data Register 1	SIDR1/ SODR1	R/W		XXXXXXXX _B
27 _H	Serial Status Register 1	SSR1	R/W		0 0 0 0 1 _ 0 0 _B
28 _H	UART1 Prescaler Control Register	U1CDCR	R/W		0 _ _ 1 1 1 1 _B
29 _H	Edge Selector	SES1	R/W		_ _ _ _ _ 0 _B
2A _H	Reserved				
2B _H	Serial IO Prescaler	SCDCR	R/W	Serial IO	0 _ _ 1 1 1 1 _B
2C _H	Serial Mode Control	SMCS	R/W		_ _ _ _ 0 0 0 0 _B
2D _H	Serial Mode Control	SMCS	R/W		0 0 0 0 0 0 1 0 _B
2E _H	Serial Data	SDR	R/W		XXXXXXXX _B
2F _H	Edge Selector	SES2	R/W		_ _ _ _ _ 0 _B
30 _H	External Interrupt Enable	ENIR	R/W	External Interrupt	0 0 0 0 0 0 0 0 _B
31 _H	External Interrupt Request	EIRR	R/W		XXXXXXXX _B
32 _H	External Interrupt Level	ELVR	R/W		0 0 0 0 0 0 0 0 _B
33 _H	External Interrupt Level	ELVR	R/W		0 0 0 0 0 0 0 0 _B
34 _H	A/D Control Status 0	ADCS0	R/W	A/D Converter	0 0 0 0 0 0 0 0 _B
35 _H	A/D Control Status 1	ADCS1	R/W		0 0 0 0 0 0 0 0 _B
36 _H	A/D Data 0	ADCR0	R		XXXXXXXX _B
37 _H	A/D Data 1	ADCR1	R/W		0 0 0 0 1 _ XX _B
38 _H	PPG0 operation mode control register	PPGC0	R/W	16-bit Programmable Pulse Generator 0/1	0 _ 0 0 0 _ _ 1 _B
39 _H	PPG1 operation mode control register	PPGC1	R/W		0 _ 0 0 0 0 0 1 _B
3A _H	PPG0 and PPG1 clock select register	PPG01	R/W		0 0 0 0 0 0 _ _ _B
3B _H	Reserved				
3C _H	PPG2 operation mode control register	PPGC2	R/W	16-bit Programmable Pulse Generator 2/3	0 _ 0 0 0 _ _ 1 _B
3D _H	PPG3 operation mode control register	PPGC3	R/W		0 _ 0 0 0 0 0 1 _B
3E _H	PPG2 and PPG3 clock select register	PPG23	R/W		0 0 0 0 0 0 _ _ _B
3F _H	Reserved				
40 _H	PPG4 operation mode control register	PPGC4	R/W	16-bit Programmable Pulse Generator 4/5	0 _ 0 0 0 _ _ 1 _B
41 _H	PPG5 operation mode control register	PPGC5	R/W		0 _ 0 0 0 0 0 1 _B
42 _H	PPG4 and PPG5 clock select register	PPG45	R/W		0 0 0 0 0 0 _ _ _B
43 _H	Reserved				
44 _H	PPG6 operation mode control register	PPGC6	R/W	16-bit Programmable Pulse Generator 6/7	0 _ 0 0 0 _ _ 1 _B
45 _H	PPG7 operation mode control register	PPGC7	R/W		0 _ 0 0 0 0 0 1 _B
46 _H	PPG6 and PPG7 clock select register	PPG67	R/W		0 0 0 0 0 0 _ _ _B

(Continued)

MB90540/545 Series

Address	Register	Abbreviation	Access	Peripheral	Initial value
47 _H to 4B _H	Reserved				
4C _H	Input Capture Control Status 0/1	ICS01	R/W	Input Capture 0/1	0 0 0 0 0 0 0 0 _B
4D _H	Input Capture Control Status 2/3	ICS23	R/W	Input Capture 2/3	0 0 0 0 0 0 0 0 _B
4E _H	Input Capture Control Status 4/5	ICS45	R/W	Input Capture 4/5	0 0 0 0 0 0 0 0 _B
4F _H	Input Capture Control Status 6/7	ICS67	R/W	Input Capture 6/7	0 0 0 0 0 0 0 0 _B
50 _H	Timer Control Status 0	TMCSR0	R/W	16-bit Reload Timer 0	0 0 0 0 0 0 0 0 _B
51 _H	Timer Control Status 0	TMCSR0	R/W		__ __ __ 0 0 0 0 _B
52 _H	Timer 0/Reload 0	TMR0/ TMRLR0	R/W		XXXXXXXX _B
53 _H	Timer 0/Reload 0	TMR0/ TMRLR0	R/W		XXXXXXXX _B
54 _H	Timer Control Status 1	TMCSR1	R/W	16-bit Reload Timer 1	0 0 0 0 0 0 0 0 _B
55 _H	Timer Control Status 1	TMCSR1	R/W		__ __ __ 0 0 0 0 _B
56 _H	Timer 1/Reload 1	TMR1/ TMRLR1	R/W		XXXXXXXX _B
57 _H	Timer 1/Reload 1	TMR1/ TMRLR1	R/W		XXXXXXXX _B
58 _H	Output Compare Control Status 0	OCS0	R/W	Output Compare 0/1	0 0 0 0 __ _ 0 0 _B
59 _H	Output Compare Control Status 1	OCS1	R/W		__ _ 0 0 0 0 0 0 _B
5A _H	Output Compare Control Status 2	OCS2	R/W	Output Compare 2/3	0 0 0 0 __ _ 0 0 _B
5B _H	Output Compare Control Status 3	OCS3	R/W		__ _ 0 0 0 0 0 0 _B
5C _H to 6B _H	Reserved				
6C _H	Timer Data	TCDDT	R/W	I/O Timer	0 0 0 0 0 0 0 0 _B
6D _H	Timer Data	TCDDT	R/W		0 0 0 0 0 0 0 0 _B
6E _H	Timer Control	TCCS	R/W		0 0 0 0 0 0 0 0 _B
6F _H	ROM Mirror	ROMM	R/W	ROM Mirror	__ __ __ __ __ 1 _B
70 _H to 7F _H	Reserved for CAN 0 Interface . Refer to "CAN Controller Hardware Manual"				
80 _H to 8F _H	Reserved for CAN 1 Interface . Refer to "CAN Controller Hardware Manual"				
90 _H to 9D _H	Reserved				
9E _H	ROM Correction Control Status	PACSR	R/W	ROM Correction	0 0 0 0 0 0 0 0 _B
9F _H	Delayed Interrupt/release	DIRR	R/W	Delayed Interrupt	__ __ __ __ __ 0 _B
A0 _H	Low-power Mode	LPMCR	R/W	Low Power Controller	0 0 0 1 1 0 0 0 _B
A1 _H	Clock Selector	CKSCR	R/W	Low Power Controller	1 1 1 1 1 1 0 0 _B
A2 _H to A4 _H	Reserved				

(Continued)

MB90540/545 Series

(Continued)

Address	Register	Abbreviation	Access	Peripheral	Initial value
A5 _H	Automatic ready function select reg.	ARSR	W	External Memory Access	0 0 1 1 _ _ 0 0 _B
A6 _H	External address output control reg.	HACR	W		0 0 0 0 0 0 0 0 _B
A7 _H	Bus control signal select register	ECSR	W		0 0 0 0 0 0 0 _ _B
A8 _H	Watchdog Control	WDTC	R/W	Watchdog Timer	XXXXX 1 1 1 _B
A9 _H	Time Base Timer Control	TBTC	R/W	Time Base Timer	1 - - 0 0 1 0 0 _B
AA _H	Watch timer control register	WTC	R/W	Watch Timer	1 X 0 0 0 0 0 0 _B
AB _H to AD _H	Reserved				
AE _H	Flash Control Status (Flash only, otherwise reserved)	FMCS	R/W	Flash Memory	0 0 0 X 0 _ _ 0 _B
AF _H	Reserved				
B0 _H	Interrupt control register 00	ICR00	R/W	Interrupt controller	0 0 0 0 0 1 1 1 _B
B1 _H	Interrupt control register 01	ICR01	R/W		0 0 0 0 0 1 1 1 _B
B2 _H	Interrupt control register 02	ICR02	R/W		0 0 0 0 0 1 1 1 _B
B3 _H	Interrupt control register 03	ICR03	R/W		0 0 0 0 0 1 1 1 _B
B4 _H	Interrupt control register 04	ICR04	R/W		0 0 0 0 0 1 1 1 _B
B5 _H	Interrupt control register 05	ICR05	R/W		0 0 0 0 0 1 1 1 _B
B6 _H	Interrupt control register 06	ICR06	R/W		0 0 0 0 0 1 1 1 _B
B7 _H	Interrupt control register 07	ICR07	R/W		0 0 0 0 0 1 1 1 _B
B8 _H	Interrupt control register 08	ICR08	R/W		0 0 0 0 0 1 1 1 _B
B9 _H	Interrupt control register 09	ICR09	R/W		0 0 0 0 0 1 1 1 _B
BA _H	Interrupt control register 10	ICR10	R/W		0 0 0 0 0 1 1 1 _B
BB _H	Interrupt control register 11	ICR11	R/W		0 0 0 0 0 1 1 1 _B
BC _H	Interrupt control register 12	ICR12	R/W		0 0 0 0 0 1 1 1 _B
BD _H	Interrupt control register 13	ICR13	R/W		0 0 0 0 0 1 1 1 _B
BE _H	Interrupt control register 14	ICR14	R/W		0 0 0 0 0 1 1 1 _B
BF _H	Interrupt control register 15	ICR15	R/W		0 0 0 0 0 1 1 1 _B
CO _H to FF _H	External				

Address	Register	Abbreviation	Access	Peripheral	Initial value
1FF0 _H	ROM Correction Address 0	PADR0	R/W	ROM Correction	XXXXXXXX _B
1FF1 _H	ROM Correction Address 1	PADR0	R/W		XXXXXXXX _B
1FF2 _H	ROM Correction Address 2	PADR0	R/W		XXXXXXXX _B
1FF3 _H	ROM Correction Address 3	PADR1	R/W		XXXXXXXX _B
1FF4 _H	ROM Correction Address 4	PADR1	R/W		XXXXXXXX _B
1FF5 _H	ROM Correction Address 5	PADR1	R/W		XXXXXXXX _B

MB90540/545 Series

Address	Register	Abbreviation	Access	Peripheral	Initial value
3900 _H	Reload L	PRLLO	R/W	16-bit Programmable Pulse Generator 0/1	XXXXXXXX _B
3901 _H	Reload H	PRLH0	R/W		XXXXXXXX _B
3902 _H	Reload L	PRL11	R/W		XXXXXXXX _B
3903 _H	Reload H	PRLH1	R/W		XXXXXXXX _B
3904 _H	Reload L	PRL22	R/W	16-bit Programmable Pulse Generator 2/3	XXXXXXXX _B
3905 _H	Reload H	PRLH2	R/W		XXXXXXXX _B
3906 _H	Reload L	PRL33	R/W		XXXXXXXX _B
3907 _H	Reload H	PRLH3	R/W		XXXXXXXX _B
3908 _H	Reload L	PRL44	R/W	16-bit Programmable Pulse Generator 4/5	XXXXXXXX _B
3909 _H	Reload H	PRLH4	R/W		XXXXXXXX _B
390A _H	Reload L	PRL55	R/W		XXXXXXXX _B
390B _H	Reload H	PRLH5	R/W		XXXXXXXX _B
390C _H	Reload L	PRL66	R/W	16-bit Programmable Pulse Generator 6/7	XXXXXXXX _B
390D _H	Reload H	PRLH6	R/W		XXXXXXXX _B
390E _H	Reload L	PRL77	R/W		XXXXXXXX _B
390F _H	Reload H	PRLH7	R/W		XXXXXXXX _B
3910 _H to 3917 _H	Reserved				
3918 _H	Input Capture 0	IPCP0	R	Input Capture 0/1	XXXXXXXX _B
3919 _H	Input Capture 0	IPCP0	R		XXXXXXXX _B
391A _H	Input Capture 1	IPCP1	R		XXXXXXXX _B
391B _H	Input Capture 1	IPCP1	R		XXXXXXXX _B
391C _H	Input Capture 2	IPCP2	R	Input Capture 2/3	XXXXXXXX _B
391D _H	Input Capture 2	IPCP2	R		XXXXXXXX _B
391E _H	Input Capture 3	IPCP3	R		XXXXXXXX _B
391F _H	Input Capture 3	IPCP3	R		XXXXXXXX _B
3920 _H	Input Capture 4	IPCP4	R	Input Capture 4/5	XXXXXXXX _B
3921 _H	Input Capture 4	IPCP4	R		XXXXXXXX _B
3922 _H	Input Capture 5	IPCP5	R		XXXXXXXX _B
3923 _H	Input Capture 5	IPCP5	R		XXXXXXXX _B
3924 _H	Input Capture 6	IPCP6	R	Input Capture 6/7	XXXXXXXX _B
3925 _H	Input Capture 6	IPCP6	R		XXXXXXXX _B
3926 _H	Input Capture 7	IPCP7	R		XXXXXXXX _B
3927 _H	Input Capture 7	IPCP7	R		XXXXXXXX _B

(Continued)

MB90540/545 Series

(Continued)

Address	Register	Abbreviation	Access	Peripheral	Initial value
3928 _H	Output Compare 0	OCCP0	R/W	Output Compare 0/1	XXXXXXXX _B
3929 _H	Output Compare 0	OCCP0	R/W		XXXXXXXX _B
392A _H	Output Compare 1	OCCP1	R/W		XXXXXXXX _B
392B _H	Output Compare 1	OCCP1	R/W		XXXXXXXX _B
392C _H	Output Compare 2	OCCP2	R/W	Output Compare 2/3	XXXXXXXX _B
392D _H	Output Compare 2	OCCP2	R/W		XXXXXXXX _B
392E _H	Output Compare 3	OCCP3	R/W		XXXXXXXX _B
392F _H	Output Compare 3	OCCP3	R/W		XXXXXXXX _B
3930 _H to 39FF _H	Reserved				
3A00 _H to 3AFF _H	Reserved for CAN 0 Interface. Refer to "CAN Controller Hardware Manual"				
3B00 _H to 3BFF _H	Reserved for CAN 0 Interface. Refer to "CAN Controller Hardware Manual"				
3C00 _H to 3CFF _H	Reserved for CAN 1 Interface. Refer to "CAN Controller Hardware Manual"				
3D00 _H to 3DFF _H	Reserved for CAN 1 Interface. Refer to "CAN Controller Hardware Manual"				
3E00 _H to 3FFF _H	Reserved				

Note Initial value of "_" represents unused bit, "X" represents unknown value.
 Addresses in the range 0000_H to 00FF_H, which are not listed in the table, are reserved for the primary functions of the MCU. A read access to these reserved addresses results reading "X" and any write access should not be performed.

MB90540/545 Series

■ CAN CONTROLLER

The MB90540 series contains two CAN controller (CAN0 and CAN1), the MB90545 series contains only one (CAN0). The Evaluation Chip MB90V540 also has two CAN controller.

The CAN controller has the following features:

- Conforms to CAN Specification Version 2.0 Part A and B
 - Supports transmission/reception in standard frame and extended frame formats
- Supports transmitting of data frames by receiving remote frames
- 16 transmitting/receiving message buffers
 - 29-bit ID and 8-byte data
 - Multi-level message buffer configuration
- Provides full-bit comparison, full-bit mask, acceptance register 0/acceptance register 1 for each message buffer as 1D acceptance mask
 - Two acceptance mask registers in either standard frame format or extended frame formats
- Bit rate programmable from 10 Kbits/s to 1 Mbits/s (when input clock is at 16 MHz)

List of Control Registers

Address		Register	Abbreviation	Access	Initial Value
CAN0	CAN1				
000070 _H	000080 _H	Message buffer valid register	BVALR	R/W	00000000 00000000 _B
000071 _H	000081 _H				
000072 _H	000082 _H	Transmit request register	TREQR	R/W	00000000 00000000 _B
000073 _H	000083 _H				
000074 _H	000084 _H	Transmit cancel register	TCANR	W	00000000 00000000 _B
000075 _H	000085 _H				
000076 _H	000086 _H	Transmit complete register	TCR	R/W	00000000 00000000 _B
000077 _H	000087 _H				
000078 _H	000088 _H	Receive complete register	RCR	R/W	00000000 00000000 _B
000079 _H	000089 _H				
00007A _H	00008A _H	Remote request receiving register	RRTRR	R/W	00000000 00000000 _B
00007B _H	00008B _H				
00007C _H	00008C _H	Receive overrun register	ROVRR	R/W	00000000 00000000 _B
00007D _H	00008D _H				
00007E _H	00008E _H	Receive interrupt enable register	RIER	R/W	00000000 00000000 _B
00007F _H	00008F _H				

MB90540/545 Series

List of Control Registers

Address		Register	Abbreviation	Access	Initial Value
CAN0	CAN1				
003B00 _H	003D00 _H	Control status register	CSR	R/W, R	00---000 0----0-1 _B
003B01 _H	003D01 _H				
003B02 _H	003D02 _H	Last event indicator register	LEIR	R/W	----- 000-0000 _B
003B03 _H	003D03 _H				
003B04 _H	003D04 _H	Receive/transmit error counter	RTEC	R	00000000 00000000 _B
003B05 _H	003D05 _H				
003B06 _H	003D06 _H	Bit timing register	BTR	R/W	-11111111 11111111 _B
003B07 _H	003D07 _H				
003B08 _H	003D08 _H	IDE register	IDER	R/W	XXXXXXXX XXXXXXXX _B
003B09 _H	003D09 _H				
003B0A _H	003D0A _H	Transmit RTR register	TRTRR	R/W	00000000 00000000 _B
003B0B _H	003D0B _H				
003B0C _H	003D0C _H	Remote frame receive waiting register	RFWTR	R/W	XXXXXXXX XXXXXXXX _B
003B0D _H	003D0D _H				
003B0E _H	003D0E _H	Transmit interrupt enable register	TIER	R/W	00000000 00000000 _B
003B0F _H	003D0F _H				
003B10 _H	003D10 _H	Acceptance mask select register	AMSR	R/W	XXXXXXXX XXXXXXXX _B
003B11 _H	003D11 _H				XXXXXXXX XXXXXXXX _B
003B12 _H	003D12 _H				XXXXXXXX XXXXXXXX _B
003B13 _H	003D13 _H	Acceptance mask register 0	AMR0	R/W	XXXXXXXX XXXXXXXX _B
003B14 _H	003D14 _H				XXXXXXXX XXXXXXXX _B
003B15 _H	003D15 _H				XXXXX--- XXXXXXXX _B
003B16 _H	003D16 _H				XXXXXXXX XXXXXXXX _B
003B17 _H	003D17 _H	Acceptance mask register 1	AMR1	R/W	XXXXXXXX XXXXXXXX _B
003B18 _H	003D18 _H				XXXXXXXX XXXXXXXX _B
003B19 _H	003D19 _H				XXXXX--- XXXXXXXX _B
003B1A _H	003D1A _H				XXXXXXXX XXXXXXXX _B
003B1B _H	003D1B _H				

MB90540/545 Series

List of Message Buffers (ID Registers) (1)

Address		Register	Abbreviation	Access	Initial Value
CAN0	CAN1				
003A00H to 003A1FH	003C00H to 003C1FH	General-purpose RAM	—	R/W	XXXXXXXX _B to XXXXXXXX _B
003A20H	003C20H	ID register 0	IDR0	R/W	XXXXXXXX XXXXXXXX _B
003A21H	003C21H				XXXXX--- XXXXXXXX _B
003A22H	003C22H				
003A23H	003C23H				
003A24H	003C24H	ID register 1	IDR1	R/W	XXXXXXXX XXXXXXXX _B
003A25H	003C25H				XXXXX--- XXXXXXXX _B
003A26H	003C26H				
003A27H	003C27H				
003A28H	003C28H	ID register 2	IDR2	R/W	XXXXXXXX XXXXXXXX _B
003A29H	003C29H				XXXXX--- XXXXXXXX _B
003A2AH	003C2AH				
003A2BH	003C2BH				
003A2CH	003C2CH	ID register 3	IDR3	R/W	XXXXXXXX XXXXXXXX _B
003A2DH	003C2DH				XXXXX--- XXXXXXXX _B
003A2EH	003C2EH				
003A2FH	003C2FH				
003A30H	003C30H	ID register 4	IDR4	R/W	XXXXXXXX XXXXXXXX _B
003A31H	003C31H				XXXXX--- XXXXXXXX _B
003A32H	003C32H				
003A33H	003C33H				
003A34H	003C34H	ID register 5	IDR5	R/W	XXXXXXXX XXXXXXXX _B
003A35H	003C35H				XXXXX--- XXXXXXXX _B
003A36H	003C36H				
003A37H	003C37H				
003A38H	003C38H	ID register 6	IDR6	R/W	XXXXXXXX XXXXXXXX _B
003A39H	003C39H				XXXXX--- XXXXXXXX _B
003A3AH	003C3AH				
003A3BH	003C3BH				

MB90540/545 Series

List of Message Buffers (ID Registers) (2)

Address		Register	Abbreviation	Access	Initial Value
CAN0	CAN1				
003A3C _H	003C3C _H	ID register 7	IDR7	R/W	XXXXXXXX XXXXXXXX _B
003A3D _H	003C3D _H				XXXXX--- XXXXXXXX _B
003A3E _H	003C3E _H				
003A3F _H	003C3F _H				
003A40 _H	003C40 _H	ID register 8	IDR8	R/W	XXXXXXXX XXXXXXXX _B
003A41 _H	003C41 _H				XXXXX--- XXXXXXXX _B
003A42 _H	003C42 _H				
003A43 _H	003C43 _H				
003A44 _H	003C44 _H	ID register 9	IDR9	R/W	XXXXXXXX XXXXXXXX _B
003A45 _H	003C45 _H				XXXXX--- XXXXXXXX _B
003A46 _H	003C46 _H				
003A47 _H	003C47 _H				
003A48 _H	003C48 _H	ID register 10	IDR10	R/W	XXXXXXXX XXXXXXXX _B
003A49 _H	003C49 _H				XXXXX--- XXXXXXXX _B
003A4A _H	003C4A _H				
003A4B _H	003C4B _H				
003A4C _H	003C4C _H	ID register 11	IDR11	R/W	XXXXXXXX XXXXXXXX _B
003A4D _H	003C4D _H				XXXXX--- XXXXXXXX _B
003A4E _H	003C4E _H				
003A4F _H	003C4F _H				
003A50 _H	003C50 _H	ID register 12	IDR12	R/W	XXXXXXXX XXXXXXXX _B
003A51 _H	003C51 _H				XXXXX--- XXXXXXXX _B
003A52 _H	003C52 _H				
003A53 _H	003C53 _H				
003A54 _H	003C54 _H	ID register 13	IDR13	R/W	XXXXXXXX XXXXXXXX _B
003A55 _H	003C55 _H				XXXXX--- XXXXXXXX _B
003A56 _H	003C56 _H				
003A57 _H	003C57 _H				
003A58 _H	003C58 _H	ID register 14	IDR14	R/W	XXXXXXXX XXXXXXXX _B
003A59 _H	003C59 _H				XXXXX--- XXXXXXXX _B
003A5A _H	003C5A _H				
003A5B _H	003C5B _H				
003A5C _H	003C5C _H	ID register 15	IDR15	R/W	XXXXXXXX XXXXXXXX _B
003A5D _H	003C5D _H				XXXXX--- XXXXXXXX _B
003A5E _H	003C5E _H				
003A5F _H	003C5F _H				

MB90540/545 Series

List of Message Buffers (DLC Registers and Data Registers) (1)

Address		Register	Abbreviation	Access	Initial Value
CAN0	CAN1				
003A60 _H	003C60 _H	DLC register 0	DLCR0	R/W	----XXXX _B
003A61 _H	003C61 _H				
003A62 _H	003C62 _H	DLC register 1	DLCR1	R/W	----XXXX _B
003A63 _H	003C63 _H				
003A64 _H	003C64 _H	DLC register 2	DLCR2	R/W	----XXXX _B
003A65 _H	003C65 _H				
003A66 _H	003C66 _H	DLC register 3	DLCR3	R/W	----XXXX _B
003A67 _H	003C67 _H				
003A68 _H	003C68 _H	DLC register 4	DLCR4	R/W	----XXXX _B
003A69 _H	003C69 _H				
003A6A _H	003C6A _H	DLC register 5	DLCR5	R/W	----XXXX _B
003A6B _H	003C6B _H				
003A6C _H	003C6C _H	DLC register 6	DLCR6	R/W	----XXXX _B
003A6D _H	003C6D _H				
003A6E _H	003C6E _H	DLC register 7	DLCR7	R/W	----XXXX _B
003A6F _H	003C6F _H				

MB90540/545 Series

List of Message Buffers (DLC Registers and Data Registers) (2)

Address		Register	Abbreviation	Access	Initial Value
CAN0	CAN1				
003A70 _H	003C70 _H	DLC register 8	DLCR8	R/W	----XXXX
003A71 _H	003C71 _H				
003A72 _H	003C72 _H	DLC register 9	DLCR9	R/W	----XXXX _B
003A73 _H	003C73 _H				
003A74 _H	003C74 _H	DLC register 10	DLCR10	R/W	----XXXX _B
003A75 _H	003C75 _H				
003A76 _H	003C76 _H	DLC register 11	DLCR11	R/W	----XXXX _B
003A77 _H	003C77 _H				
003A78 _H	003C78 _H	DLC register 12	DLCR12	R/W	----XXXX _B
003A79 _H	003C79 _H				
003A7A _H	003C7A _H	DLC register 13	DLCR13	R/W	----XXXX _B
003A7B _H	003C7B _H				
003A7C _H	003C7C _H	DLC register 14	DLCR14	R/W	----XXXX _B
003A7D _H	003C7D _H				
003A7E _H	003C7E _H	DLC register 15	DLCR15	R/W	----XXXX _B
003A7F _H	003C7F _H				
003A80 _H to 003A87 _H	003C80 _H to 003C87 _H	Data register 0 (8 bytes)	DTR0	R/W	XXXXXXXX _B to XXXXXXXX _B
003A88 _H to 003A8F _H	003C88 _H to 003C8F _H	Data register 1 (8 bytes)	DTR1	R/W	XXXXXXXX _B to XXXXXXXX _B
003A90 _H to 003A97 _H	003C90 _H to 003C97 _H	Data register 2 (8 bytes)	DTR2	R/W	XXXXXXXX _B to XXXXXXXX _B
003A98 _H to 003A9F _H	003C98 _H to 003C9F _H	Data register 3 (8 bytes)	DTR3	R/W	XXXXXXXX _B to XXXXXXXX _B
003AA0 _H to 003AA7 _H	003CA0 _H to 003CA7 _H	Data register 4 (8 bytes)	DTR4	R/W	XXXXXXXX _B to XXXXXXXX _B
003AA8 _H to 003AAF _H	003CA8 _H to 003CAF _H	Data register 5 (8 bytes)	DTR5	R/W	XXXXXXXX _B to XXXXXXXX _B
003AB0 _H to 003AB7 _H	003CB0 _H to 003CB7 _H	Data register 6 (8 bytes)	DTR6	R/W	XXXXXXXX _B to XXXXXXXX _B

MB90540/545 Series

List of Message Buffers (DLC Registers and Data Registers) (3)

Address		Register	Abbreviation	Access	Initial Value
CAN0	CAN1				
003AB8H to 003ABFH	003CB8H to 003CBFH	Data register 7 (8 bytes)	DTR7	R/W	XXXXXXXX _B to XXXXXXXX _B
003AC0H to 003AC7H	003CC0H to 003CC7H	Data register 8 (8 bytes)	DTR8	R/W	XXXXXXXX _B to XXXXXXXX _B
003AC8H to 003ACFH	003CC8H to 003CCFH	Data register 9 (8 bytes)	DTR9	R/W	XXXXXXXX _B to XXXXXXXX _B
003AD0H to 003AD7H	003CD0H to 003CD7H	Data register 10 (8 bytes)	DTR10	R/W	XXXXXXXX _B to XXXXXXXX _B
003AD8H to 003ADFH	003CD8H to 003CDFH	Data register 11 (8 bytes)	DTR11	R/W	XXXXXXXX _B to XXXXXXXX _B
003AE0H to 003AE7H	003CE0H to 003CE7H	Data register 12 (8 bytes)	DTR12	R/W	XXXXXXXX _B to XXXXXXXX _B
003AE8H to 003AEFH	003CE8H to 003CEFH	Data register 13 (8 bytes)	DTR13	R/W	XXXXXXXX _B to XXXXXXXX _B
003AF0H to 003AF7H	003CF0H to 003CF7H	Data register 14 (8 bytes)	DTR14	R/W	XXXXXXXX _B to XXXXXXXX _B
003AF8H to 003AFFH	003CF8H to 003CFFH	Data register 15 (8 bytes)	DTR15	R/W	XXXXXXXX _B to XXXXXXXX _B

■ INTERRUPT MAP

Interrupt cause	POS clear	Interrupt vector		Interrupt control register	
		Number	Address	Number	Address
Reset	N/A	#08	FFFFDC _H	—	—
INT9 instruction	N/A	#09	FFFFD8 _H	—	—
Exception	N/A	#10	FFFFD4 _H	—	—
CAN 0 RX	N/A	#11	FFFFD0 _H	ICR00	0000B0 _H
CAN 0 TX/NS	N/A	#12	FFFFCC _H		
CAN 1 RX	N/A	#13	FFFFC8 _H	ICR01	0000B1 _H
CAN 1 TX/NS	N/A	#14	FFFFC4 _H		
External Interrupt INT0/INT1	*1	#15	FFFFC0 _H	ICR02	0000B2 _H
Time Base Timer	N/A	#16	FFFFBC _H		
16-bit Reload Timer 0	*1	#17	FFFFB8 _H	ICR03	0000B3 _H
A/D Converter	*1	#18	FFFFB4 _H		
I/O Timer	N/A	#19	FFFFB0 _H	ICR04	0000B4 _H
External Interrupt INT2/INT3	*1	#20	FFFFAC _H		
Serial I/O	*1	#21	FFFFA8 _H	ICR05	0000B5 _H
PPG 0/1	N/A	#22	FFFFA4 _H		
Input Capture 0	*1	#23	FFFFA0 _H	ICR06	0000B6 _H
External Interrupt INT4/INT5	*1	#24	FFFF9C _H		
Input Capture 1	*1	#25	FFFF98 _H	ICR07	0000B7 _H
PPG 2/3	N/A	#26	FFFF94 _H		
External Interrupt INT6/INT7	*1	#27	FFFF90 _H	ICR08	0000B8 _H
Watch Timer	N/A	#28	FFFF8C _H		
PPG 4/5	N/A	#29	FFFF88 _H	ICR09	0000B9 _H
Input Capture 2/3	*1	#30	FFFF84 _H		
PPG 6/7	N/A	#31	FFFF80 _H	ICR10	0000BA _H
Output Compare 0	*1	#32	FFFF7C _H		
Output Compare 1	*1	#33	FFFF78 _H	ICR11	0000BB _H
Input Capture 4/5	*1	#34	FFFF74 _H		
Output Compare 2/3 - Input Capture 6/7	*1	#35	FFFF70 _H	ICR12	0000BC _H
16-bit Reload Timer 1	*1	#36	FFFF6C _H		
UART 0 RX	*2	#37	FFFF68 _H	ICR13	0000BD _H
UART 0 TX	*1	#38	FFFF64 _H		
UART 1 RX	*2	#39	FFFF60 _H	ICR14	0000BE _H
UART 1 TX	*1	#40	FFFF5C _H		
Flash Memory	N/A	#41	FFFF58 _H	ICR15	0000BF _H
Delayed interrupt	N/A	#42	FFFF54 _H		

MB90540/545 Series

*1: The interrupt request flag is cleared by the I²OS interrupt clear signal.

*2: The interrupt request flag is cleared by the I²OS interrupt clear signal. A stop request is available.

N/A: The interrupt request flag is not cleared by the I²OS interrupt clear signal.

Note: For a peripheral module with two interrupt causes for a single interrupt number, both interrupt request flags are cleared by the I²OS interrupt clear signal.

Note: At the end of I²OS, the I²OS clear signal will be asserted for all the interrupt flags assigned to the same interrupt number. If one interrupt flag starts the I²OS and in the meantime another interrupt flag is set by hardware event, the later event is lost because the flag is cleared by the I²OS clear signal caused by the first event. So it is recommended not to use the I²OS for this interrupt number.

Note: If I²OS is enabled, I²OS is initiated when one of the two interrupt signals in the same interrupt control register (ICR) is asserted. This means that different interrupt sources share the same I²OS Descriptor which should be unique for each interrupt source. For this reason, when one interrupt source uses the I²OS, the other interrupt should be disabled.

■ ELECTRICAL CHARACTERISTICS

1. Absolute Maximum Ratings

($V_{SS} = AV_{SS} = 0V$)

Parameter	Symbol	Value		Units	Remarks
		Min.	Max.		
Power supply voltage	V_{CC}	$V_{SS} - 0.3$	$V_{SS} + 6.0$	V	
	AV_{CC}	$V_{SS} - 0.3$	$V_{SS} + 6.0$	V	$V_{CC} = AV_{CC}$ *1
	AVR_{\pm}	$V_{SS} - 0.3$	$V_{SS} + 6.0$	V	$AV_{CC} \geq AVR_{\pm}$, $AVR_{+} \geq AVR_{-}$
Input voltage	V_I	$V_{SS} - 0.3$	$V_{SS} + 6.0$	V	*2
Output voltage	V_O	$V_{SS} - 0.3$	$V_{SS} + 6.0$	V	*2
Clamp Current	I_{CLAMP}	-2.0	2.0	mA	
"L" level max. output current	I_{OL}	—	15	mA	
"L" level avg. output current	I_{OLAV}	—	4	mA	Average value over a period of 100ms
"L" level max. overall output current	ΣI_{OL}	—	100	mA	
"L" level avg. overall output current	ΣI_{OLAV}	—	50	mA	Average value over a period of 100ms
"H" level max. output current	I_{OH}	—	-15	mA	
"H" level avg. output current	I_{OHAV}	—	-4	mA	Average value over a period of 100ms
"H" level max. overall output current	ΣI_{OH}	—	-100	mA	
"H" level avg. overall output current	ΣI_{OHAV}	—	-50	mA	Average value over a period of 100ms
Power consumption	P_D	—	500	mW	MB90F543/F549
		—	400	mW	MB90543/549
Operating temperature	T_A	-40	+85	°C	
Storage temperature	T_{STG}	-55	+150	°C	

*1: Set AV_{CC} and V_{CC} to the same voltage. Make sure that AV_{CC} does not exceed V_{CC} and that the voltage at the analog inputs does not exceed AV_{CC} when the power is switched on.

*2: V_I and V_O should not exceed $V_{CC} + 0.3V$. V_I should not exceed the specified ratings. However if the maximum current to/from a input is limited by some means with external components, the I_I rating supercedes the V_I rating.

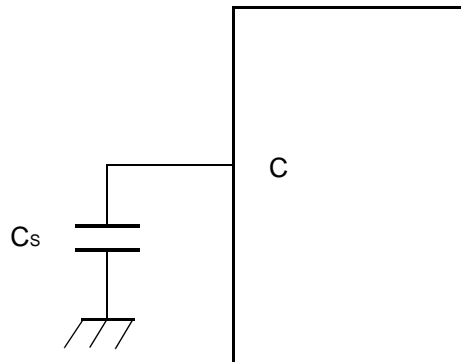
MB90540/545 Series

2. Recommended Conditions

($V_{SS} = AV_{SS} = 0V$)

Parameter	Symbol	Value			Units	Remarks
		Min.	Typ.	Max.		
Power supply voltage	V_{CC}	4.5	5.0	5.5	V	
Input H voltage	V_{IHS}	$0.8 V_{CC}$		$V_{CC} + 0.3$	V	CMOS hysteresis input pin
	V_{IHM}	$V_{CC} - 0.3$		$V_{CC} + 0.3$	V	MD input pin
Input L voltage	V_{ILS}	$V_{SS} - 0.3$		$0.2 V_{CC}$	V	CMOS hysteresis input pin
	V_{ILM}	$V_{SS} - 0.3$		$V_{SS} + 0.3$	V	MD input pin
Smooth capacitor	C_S	0.022	0.1	1.0	μF	Use a ceramic capacitor or capacitor of better AC characteristics. Capacitor at the V_{CC} should be greater than this capacitor.
Operating temperature	T_A	-40		+85	$^{\circ}C$	

• C Pin Connection Diagram



MB90540/545 Series

3. DC Characteristics

($V_{CC} = 5.0\text{ V} \pm 10\%$, $V_{SS} = AV_{SS} = 0\text{V}$, $T_A = -40\text{ }^\circ\text{C}$ to $+85\text{ }^\circ\text{C}$)

Parameter	Symbol	Pin	Condition	Value			Units	Remarks
				Min.	Typ.	Max.		
Output H voltage	V_{OH}	All output pins	$V_{CC} = 4.5\text{V}$, $I_{OH} = -4.0\text{mA}$	$V_{CC} - 0.5$	—	—	V	
Output L voltage	V_{OL}	All output pins	$V_{CC} = 4.5\text{V}$, $I_{OL} = 4.0\text{mA}$	—	—	0.4	V	
Input leak current	I_{IL}		$V_{CC} = 5.5\text{V}$, $V_{SS} < V_I < V_{CC}$	-5	—	5	μA	
Power supply current*	I_{CC}	V_{CC}	$V_{CC} = 5.0\text{ V} \pm 10\%$, Internal frequency: 16 MHz, At normal operating	—	TBD	TBD	mA	MB90543/549
				—	45	60	mA	MB90F543/F549
	I_{CCS}		$V_{CC} = 5.0\text{V} \pm 10\%$, Internal frequency: 16 MHz, At sleep	—	TBD	TBD	mA	MB90543/549
				—	13	22	mA	MB90F543/F549
	I_{CCL}		$V_{CC} = 5.0\text{V}$, Internal frequency: 8 kHz, At sub operation	—	TBD	TBD	mA	MB90543/549
				—	0.2	1	mA	MB90F543/F549
	I_{CCLS}		$V_{CC} = 5.0\text{V}$, Internal frequency: 8 kHz, At sub sleep	—	TBD	TBD	μA	MB90543/549
				—	10	50	μA	MB90F543/F549
	I_{CCT}		$V_{CC} = 5.0\text{V}$, Internal frequency: 8 kHz, At watch mode	—	TBD	TBD	μA	MB90543/549
				—	10	50	μA	MB90F543/F549
I_{CCH1}	$V_{CC} = 5.0\text{ V} \pm 10\%$, At stop, $T_A = 25^\circ\text{C}$	—	TBD	TBD	μA	MB90543/549		
		—	5	20	μA	MB90F543/F549		
I_{CCH2}	$V_{CC} = 5.0\text{ V} \pm 10\%$, At hardware standby mode, $T_A = 25^\circ\text{C}$	—	TBD	TBD	μA	MB90543/549		
		—	50	100	μA	MB90F543/F549		
Input capacity	C_{IN}	Other than AV_{CC} , AV_{SS} , $AVR+$, $AVR-$, C, V_{CC} , V_{SS}	—	10	80	pF		

*: Current values are tentative. They are subject to change without notice according to improvements in the characteristics. The power supply current testing conditions are when using the external clock.

MB90540/545 Series

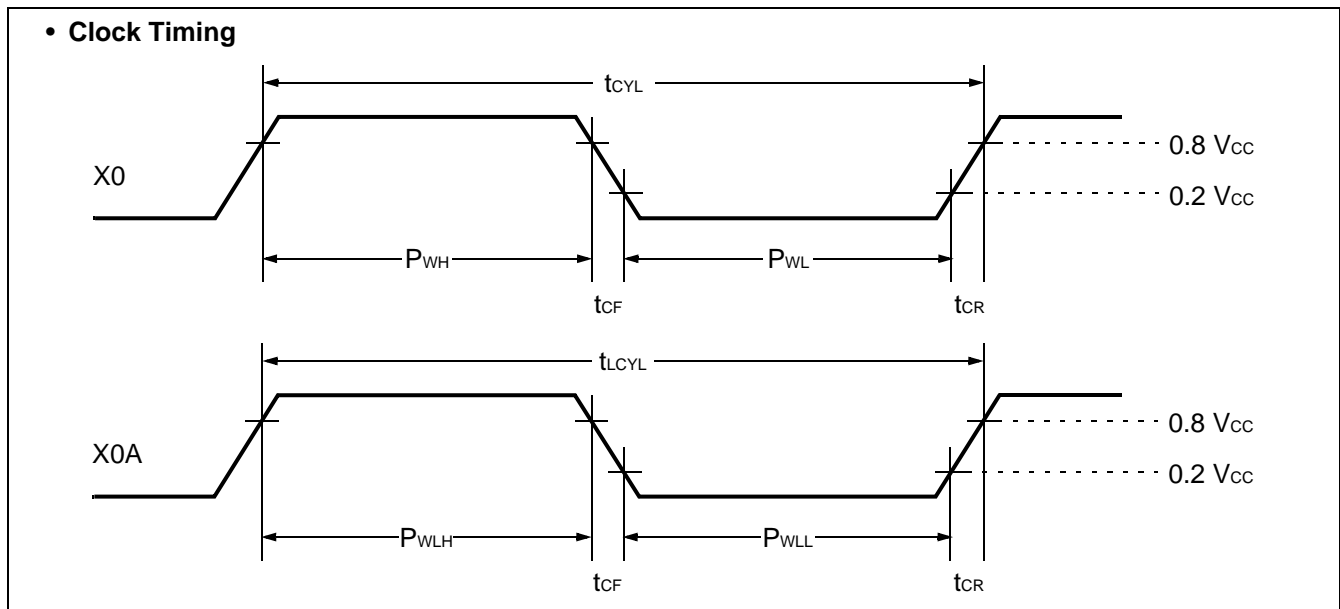
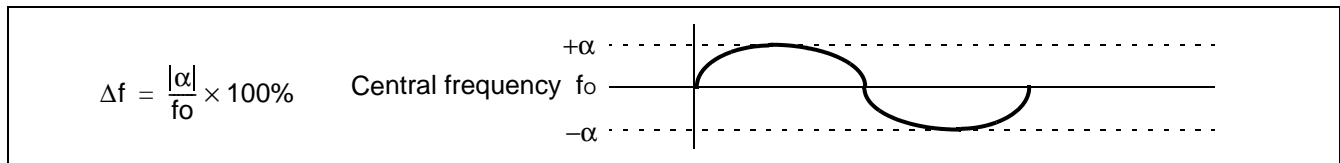
4. AC Characteristics

(1) Clock Timing

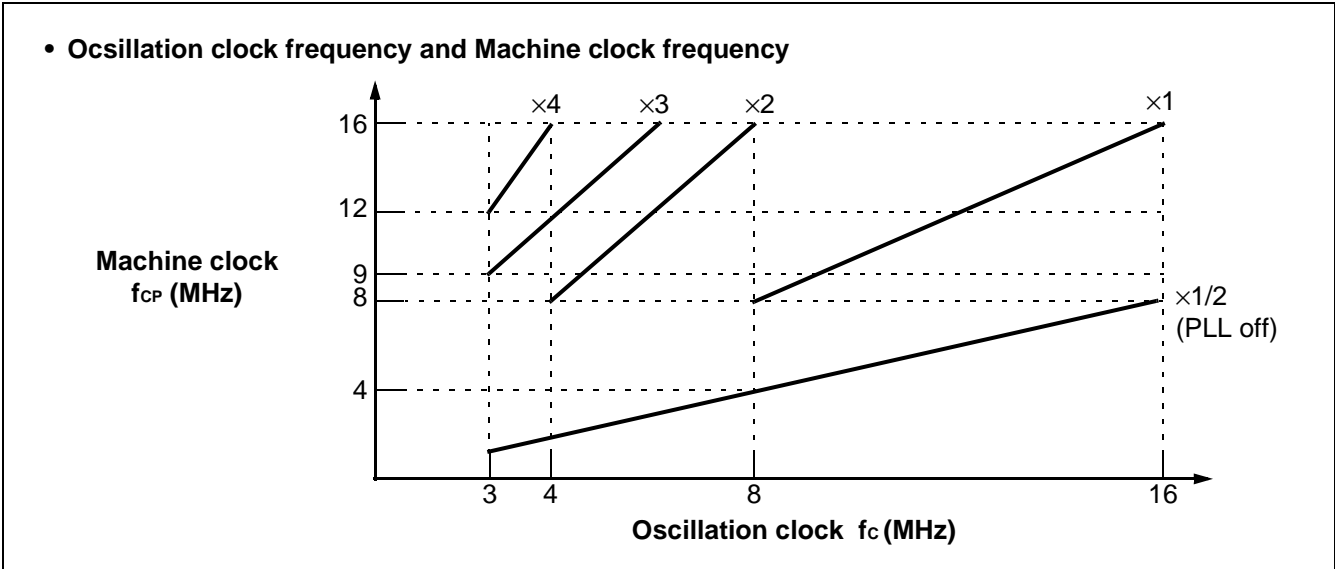
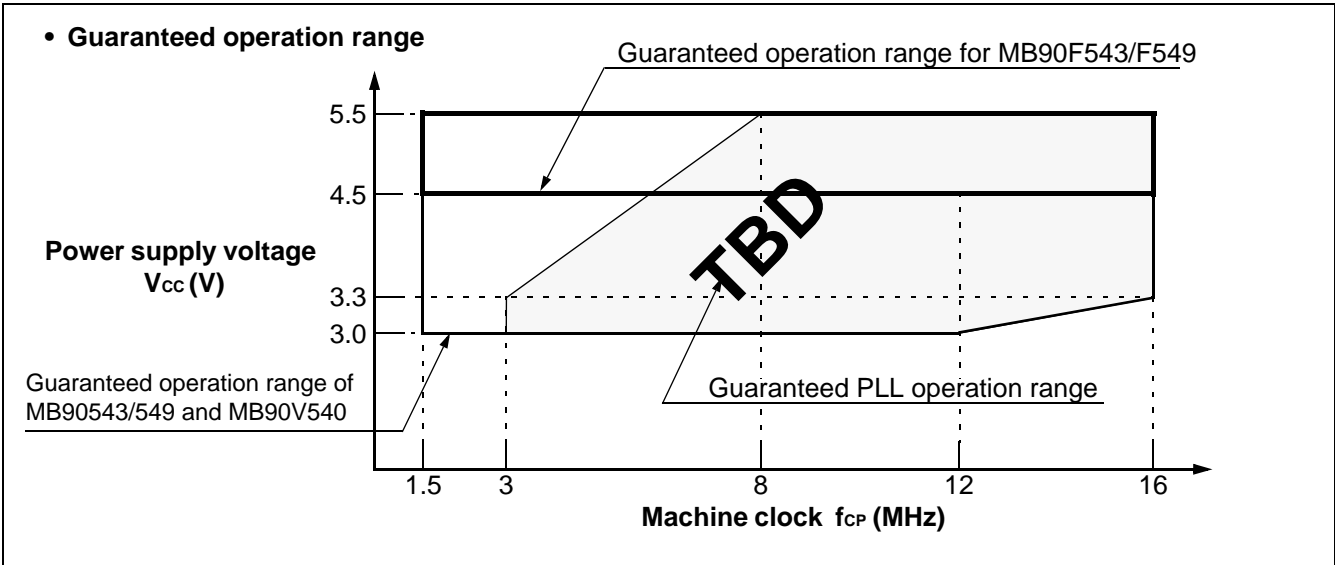
($V_{CC} = 5.0\text{ V} \pm 10\%$, $V_{SS} = AV_{SS} = 0\text{ V}$, $T_A = -40\text{ }^\circ\text{C}$ to $+85\text{ }^\circ\text{C}$)

Parameter	Symbol	Pin	Value			Units	Remarks
			Min.	Typ.	Max.		
Oscillation frequency	f_c	X0, X1	3	—	16	MHz	
	f_{CL}	X0A, X1A	—	32.768	—	kHz	
Oscillation cycle time	t_{CYL}	X0, X1	62.5	—	333	ns	
	t_{LCYL}	X0A, X1A	—	30.5	—	μs	
Frequency deviation with PLL *	Δf	—	—	—	5	%	
Input clock pulse width	P_{WH}, P_{WL}	X0	10	—	—	ns	Duty ratio is about 30 to 70%.
	P_{WLH}, P_{WLL}	X0A	—	15.2	—	μs	
Input clock rise and fall time	t_{CR}, t_{CF}	X0	—	—	5	ns	When using external clock
Machine clock frequency	f_{CP}	—	1.5	—	16	MHz	When using main clock
	f_{LCP}	—	—	8.192	—	kHz	When using sub-clock
Machine clock cycle time	t_{CP}	—	62.5	—	666	ns	When using main clock
	t_{LCP}	—	—	122.1	—	μs	When using sub-clock

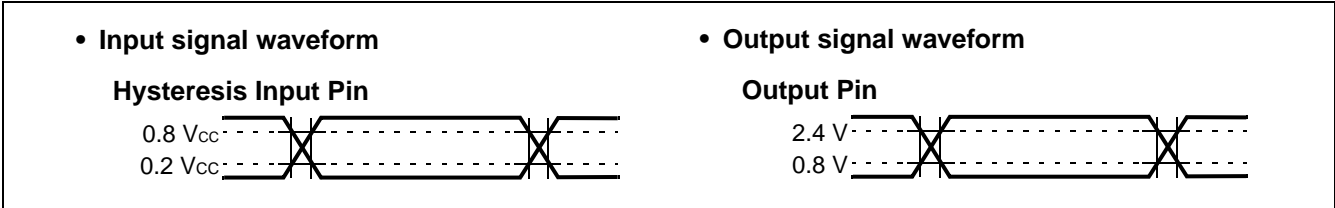
* : Frequency deviation indicates the maximum frequency difference from the target frequency when using a multiplied clock.



MB90540/545 Series



AC characteristics are set to the measured reference voltage values below.

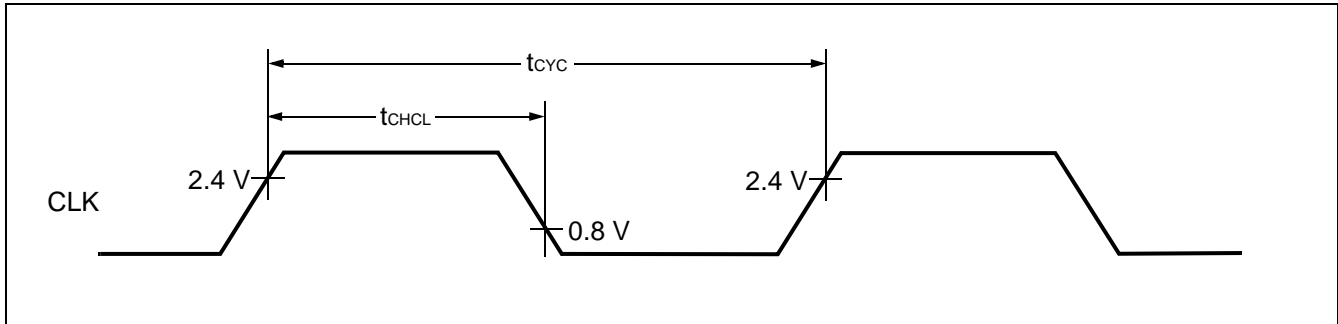


MB90540/545 Series

(2) Clock Output Timing

($V_{CC} = 5.0\text{ V} \pm 10\%$, $V_{SS} = AV_{SS} = 0.0\text{ V}$, $T_A = -40\text{ }^\circ\text{C}$ to $+85\text{ }^\circ\text{C}$)

Parameter	Symbol	Pin	Condition	Value		Units	Remarks
				Min.	Max.		
Cycle time	t_{CYC}	CLK	$V_{CC} = 5\text{ V} \pm 10\%$	62.5	—	ns	
CLK $\uparrow \Rightarrow$ CLK \downarrow	t_{CHCL}			20	—	ns	



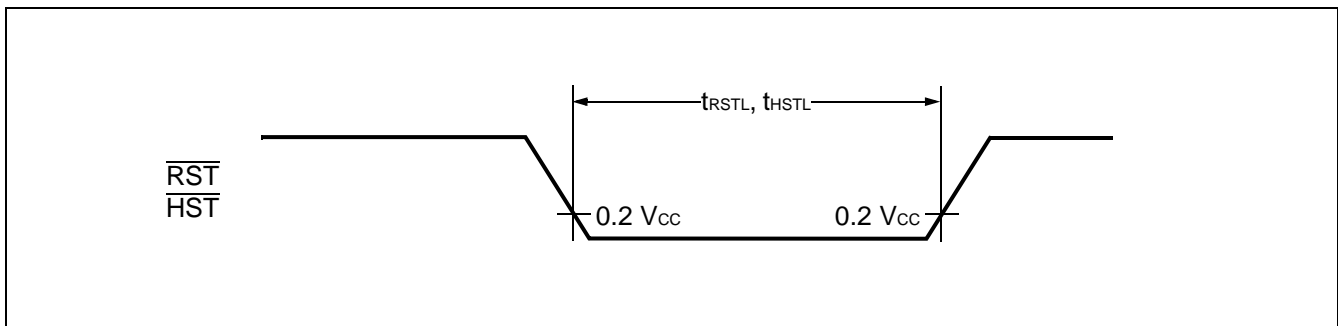
(3) Reset and Hardware Standby Input

($V_{CC} = 5.0\text{ V} \pm 10\%$, $V_{SS} = AV_{SS} = 0.0\text{ V}$, $T_A = -40\text{ }^\circ\text{C}$ to $+85\text{ }^\circ\text{C}$)

Parameter	Symbol	Pin	Value		Units	Remarks
			Min.	Max.		
Reset input time	t_{RSTL}	$\overline{\text{RST}}$	$16 t_{CP}$	—	ns	
Hardware standby input time	t_{HSTL}	$\overline{\text{HST}}$	$16 t_{CP}$	—	ns	

" t_{CP} " represents one cycle time of the machine clock.

Any reset can not fully initialize the Flash Memory if it is performing the automatic algorithm.



MB90540/545 Series

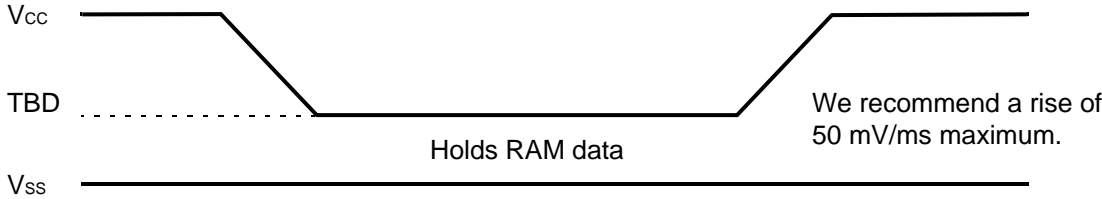
(4) Power On Reset

($V_{CC} = 5.0\text{ V} \pm 10\%$, $V_{SS} = AV_{SS} = 0.0\text{ V}$, $T_A = -40\text{ }^\circ\text{C}$ to $+85\text{ }^\circ\text{C}$)

Parameter	Symbol	Pin	Condition	Value		Units	Remarks
				Min.	Max.		
Power on rise time	t_R	V_{CC}	—	0.05	30	ms	
Power off time	t_{OFF}	V_{CC}		50	—	ms	Due to repetitive operation



If you change the power supply voltage too rapidly, a power on reset may occur. We recommend that you startup smoothly by restraining voltages when changing the power supply voltage during operation, as shown in the figure below. Perform while not using the PLL clock. However, if voltage drops are within 1 mV/sec, you can operate while using the PLL clock.



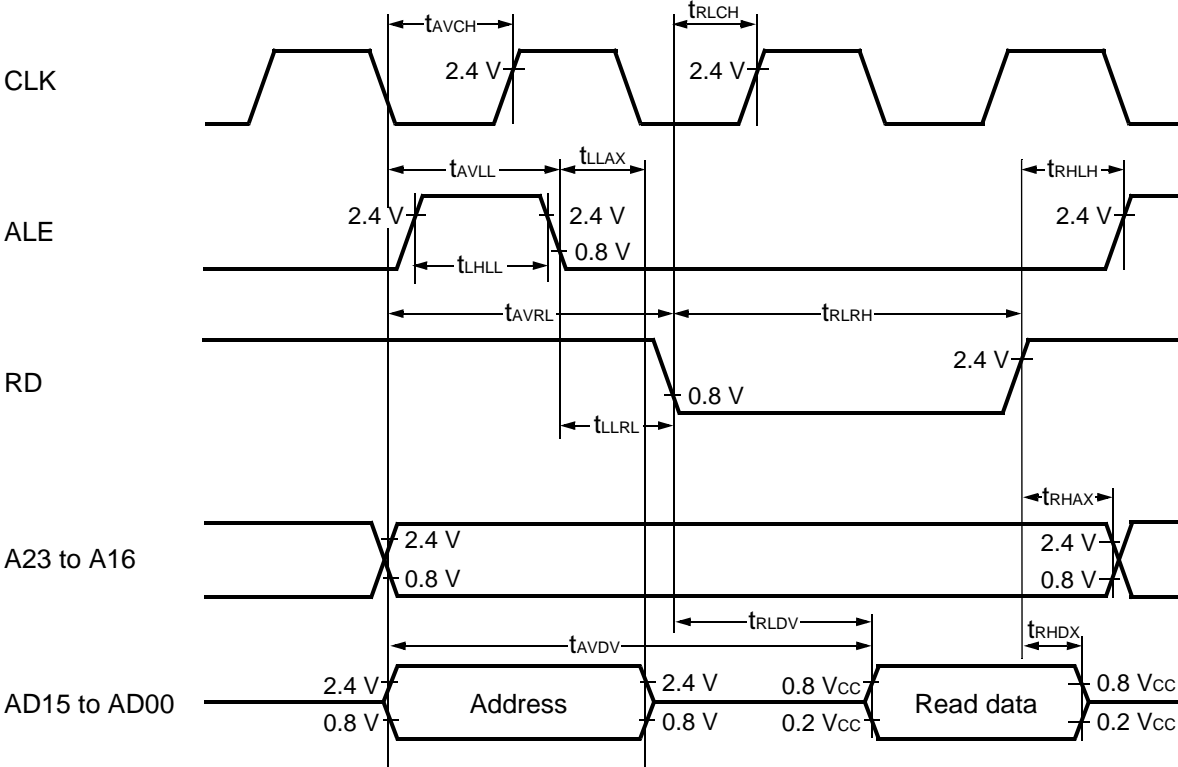
MB90540/545 Series

(5) Bus Timing (Read)

($V_{CC} = 4.5\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_A = -40\text{ }^{\circ}\text{C to }+85\text{ }^{\circ}\text{C}$)

Parameter	Symbol	Pin	Condition	Value		Units	Remarks
				Min.	Max.		
ALE pulse width	t_{LHLL}	ALE	—	$t_{CP}/2 - 20$		ns	
Valid address \Rightarrow ALE \downarrow time	t_{AVLL}	ALE, A23 to A16, AD15 to AD00		$t_{CP}/2 - 20$	—	ns	
ALE $\downarrow \Rightarrow$ Address valid time	t_{LLAX}	ALE, AD15 to AD00		$t_{CP}/2 - 15$	—	ns	
Valid address $\Rightarrow \overline{RD} \downarrow$ time	t_{AVRL}	A23 to A16, AD15 to AD00, \overline{RD}		$t_{CP} - 15$	—	ns	
Valid address \Rightarrow Valid data input	t_{AVDV}	A23 to A16, AD15 to AD00		—	$5 t_{CP}/2 - 60$	ns	
\overline{RD} pulse width	t_{RLRH}	\overline{RD}		$3 t_{CP}/2 - 20$	—	ns	
$\overline{RD} \downarrow \Rightarrow$ Valid data input	t_{RLDV}	\overline{RD} , AD15 to AD00		—	$3 t_{CP}/2 - 60$	ns	
$\overline{RD} \uparrow \Rightarrow$ Data hold time	t_{RHDX}	\overline{RD} , AD15 to AD00		0	—	ns	
$\overline{RD} \downarrow \Rightarrow$ ALE \uparrow time	t_{RHLH}	\overline{RD} , ALE		$t_{CP}/2 - 15$	—	ns	
$\overline{RD} \uparrow \Rightarrow$ Address valid time	t_{RHAX}	\overline{RD} , A23 to A16		$t_{CP}/2 - 10$	—	ns	
Valid address \Rightarrow CLK \uparrow time	t_{AVCH}	A23 to A16, AD15 to AD00, CLK		$t_{CP}/2 - 20$	—	ns	
$\overline{RD} \downarrow \Rightarrow$ CLK \uparrow time	t_{RLCH}	\overline{RD} , CLK		$t_{CP}/2 - 20$	—	ns	
ALE $\downarrow \Rightarrow \overline{RD} \downarrow$ time	t_{LLRL}	ALE, \overline{RD}		$t_{CP}/2 - 15$	—	ns	

• Bus Timing (Read)

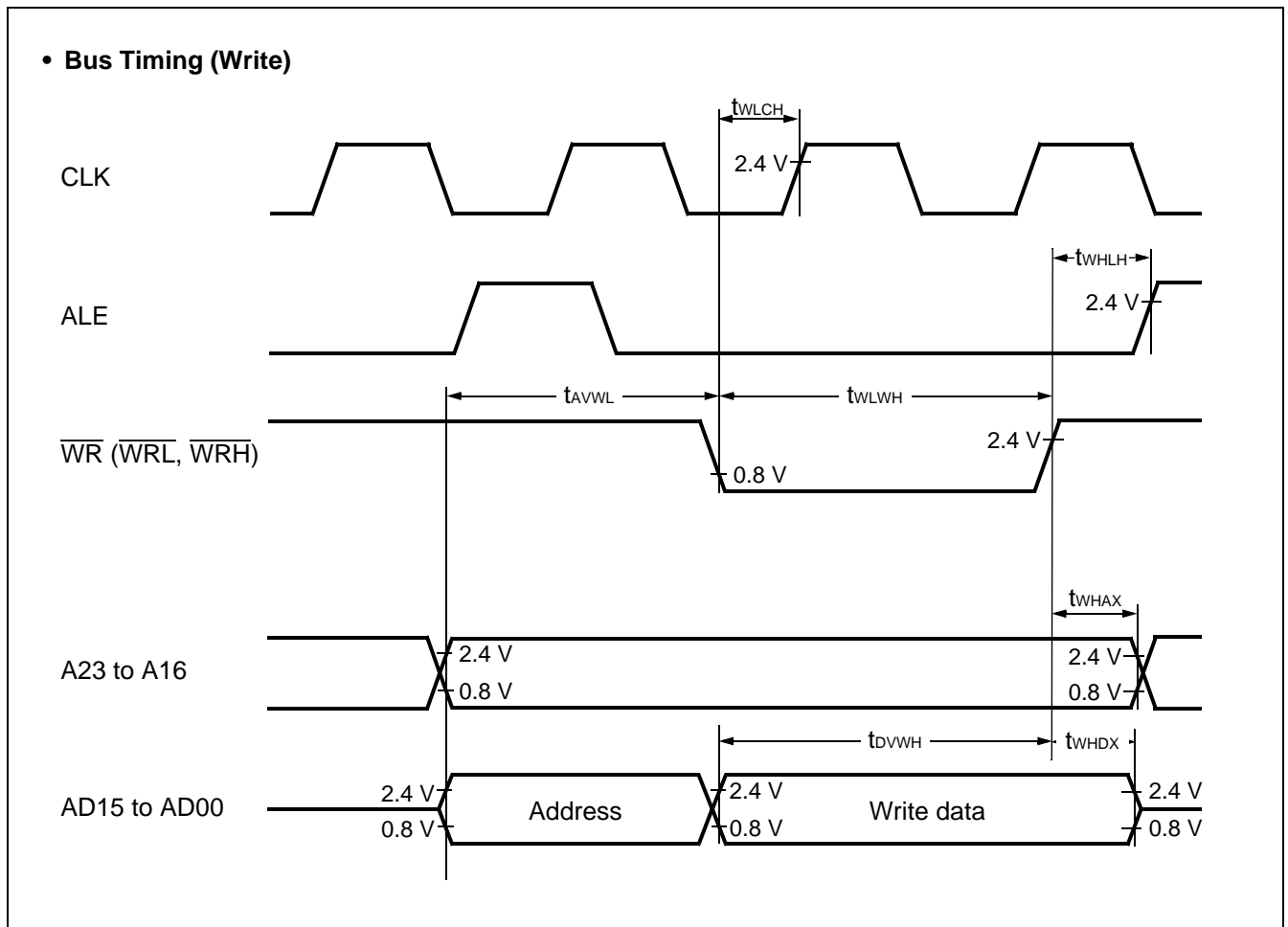


MB90540/545 Series

(6) Bus Timing (Write)

($V_{CC} = 4.5\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_A = -40\text{ }^\circ\text{C to }+85\text{ }^\circ\text{C}$)

Parameter	Symbol	Pin	Condition	Value		Units	Remarks
				Min.	Max.		
Valid address $\Rightarrow \overline{WR} \downarrow$ time	t_{AVWL}	A23 to A16, AD15 to AD00, \overline{WR}	—	$t_{CP} - 15$	—	ns	
\overline{WR} pulse width	t_{WLWH}	WR		$3 t_{CP}/2 - 20$	—	ns	
Valid data output $\Rightarrow \overline{WR} \uparrow$ time	t_{DVWH}	AD15 to AD00, \overline{WR}		$3 t_{CP}/2 - 20$	—	ns	
$\overline{WR} \uparrow \Rightarrow$ Data hold time	t_{WHDX}	AD15 to AD00, \overline{WR}		20	—	ns	
$\overline{WR} \uparrow \Rightarrow$ Address valid time	t_{WHAX}	A23 to A16, \overline{WR}		$t_{CP}/2 - 10$	—	ns	
$\overline{WR} \uparrow \Rightarrow$ ALE \uparrow time	t_{WHLH}	\overline{WR} , ALE		$t_{CP}/2 - 15$	—	ns	
$\overline{WR} \downarrow \Rightarrow$ CLK \uparrow time	t_{WLCH}	\overline{WR} , CLK		$t_{CP}/2 - 20$	—	ns	

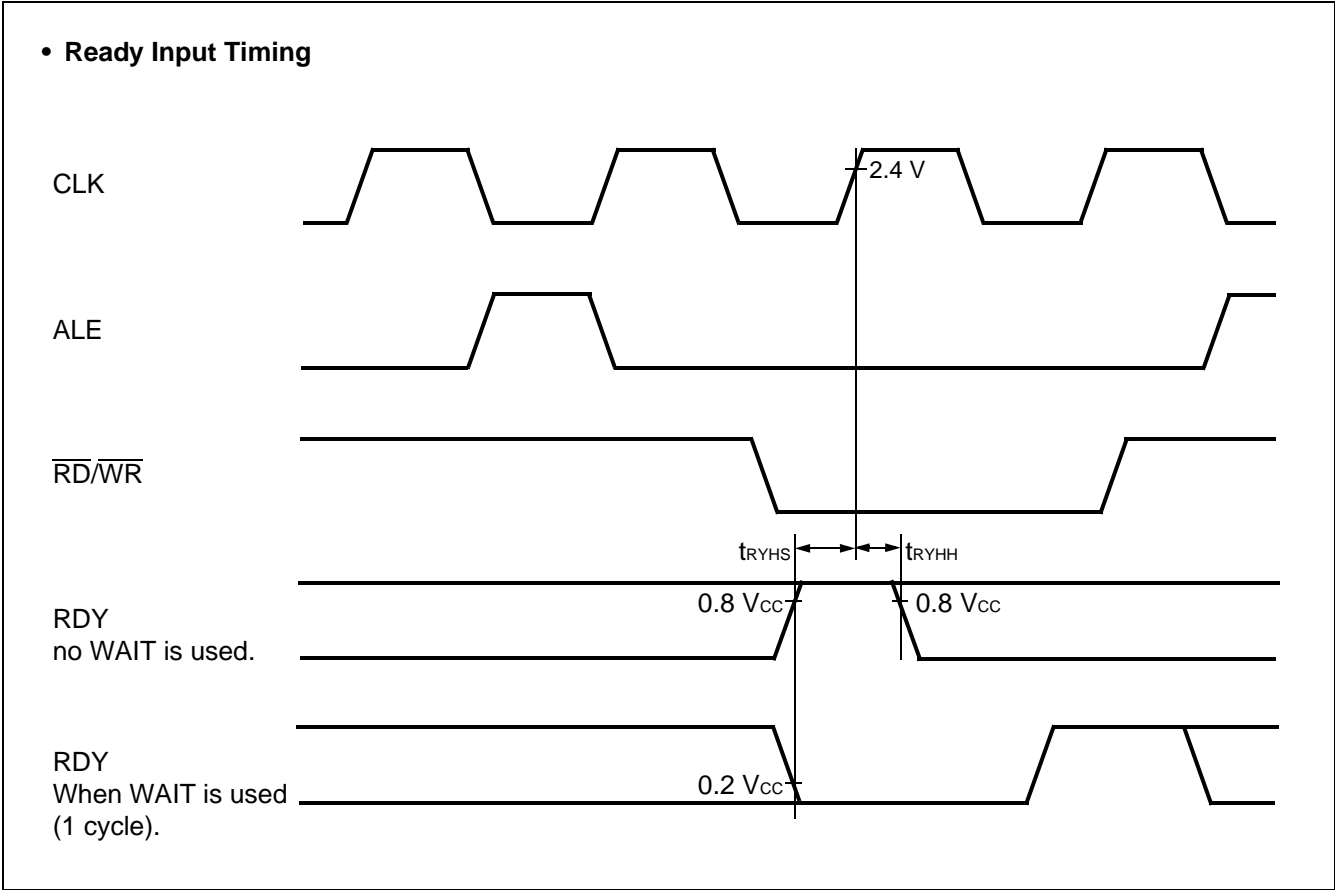


(7) Ready Input Timing

($V_{CC} = 4.5\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_A = -40\text{ }^\circ\text{C to }+85\text{ }^\circ\text{C}$)

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

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



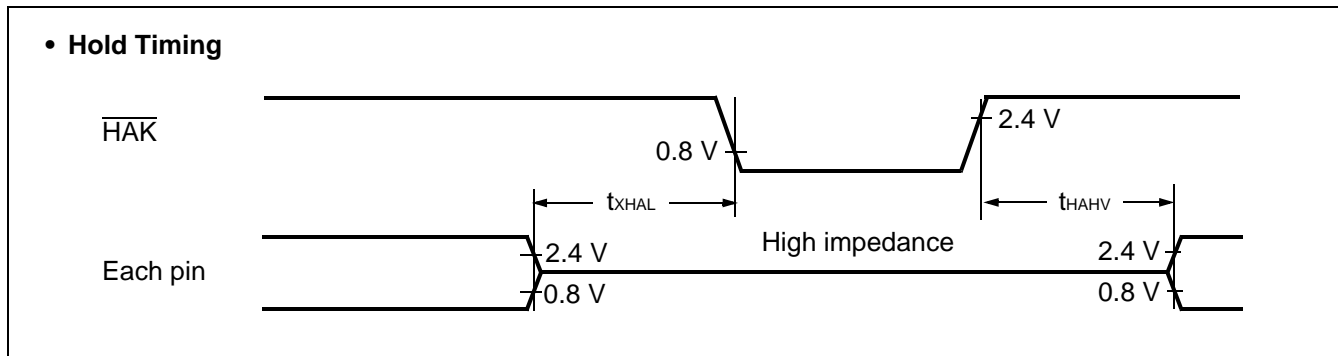
MB90540/545 Series

(8) Hold Timing

($V_{CC} = 4.5\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_A = -40\text{ }^\circ\text{C to }+85\text{ }^\circ\text{C}$)

Parameter	Symbol	Pin	Condition	Value		Units	Remarks
				Min.	Max.		
Pin floating $\Rightarrow \overline{\text{HAK}} \downarrow$ time	t_{XHAL}	$\overline{\text{HAK}}$	—	30	t_{CP}	ns	
$\overline{\text{HAK}} \uparrow$ time \Rightarrow Pin valid time	t_{HAHV}	$\overline{\text{HAK}}$		t_{CP}	$2 t_{CP}$	ns	

Note: There is more than 1 cycle from when HRQ reads in until the $\overline{\text{HAK}}$ is changed.



(9) UART0/1, Serial I/O Timing

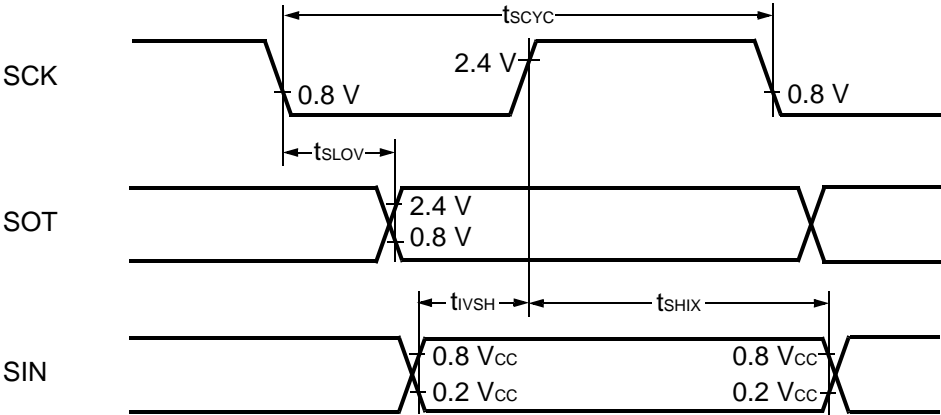
($V_{CC} = 4.5\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_A = -40\text{ }^\circ\text{C to }+85\text{ }^\circ\text{C}$)

Parameter	Symbol	Pin Symbol	Condition	Value		Units	Remarks
				Min.	Max.		
Serial clock cycle time	t_{SCYC}	SCK0 to SCK2	Internal clock operation output pins are $C_L = 80\text{ pF} + 1\text{ TTL}$.	$8 t_{CP}$	—	ns	
SCK $\downarrow \Rightarrow$ SOT delay time	t_{SLOV}	SCK0 to SCK2, SOT0 to SOT2		-80	80	ns	
Valid SIN \Rightarrow SCK \uparrow	t_{VSH}	SCK0 to SCK2, SIN0 to SIN2		100	—	ns	
SCK $\uparrow \Rightarrow$ Valid SIN hold time	t_{SHIX}	SCK0 to SCK2, SIN0 to SIN2		60	—	ns	
Serial clock "H" pulse width	t_{SHSL}	SCK0 to SCK2	External clock operation output pins are $C_L = 80\text{ pF} + 1\text{ TTL}$.	$4 t_{CP}$	—	ns	
Serial clock "L" pulse width	t_{LSLH}	SCK0 to SCK2		$4 t_{CP}$	—	ns	
SCK $\downarrow \Rightarrow$ SOT delay time	t_{SLOV}	SCK0 to SCK2, SOT0 to SOT2		—	150	ns	
Valid SIN \Rightarrow SCK \uparrow	t_{VSH}	SCK0 to SCK2, SIN0 to SIN2		60	—	ns	
SCK $\uparrow \Rightarrow$ Valid SIN hold time	t_{SHIX}	SCK0 to SCK2, SIN0 to SIN2		60	—	ns	

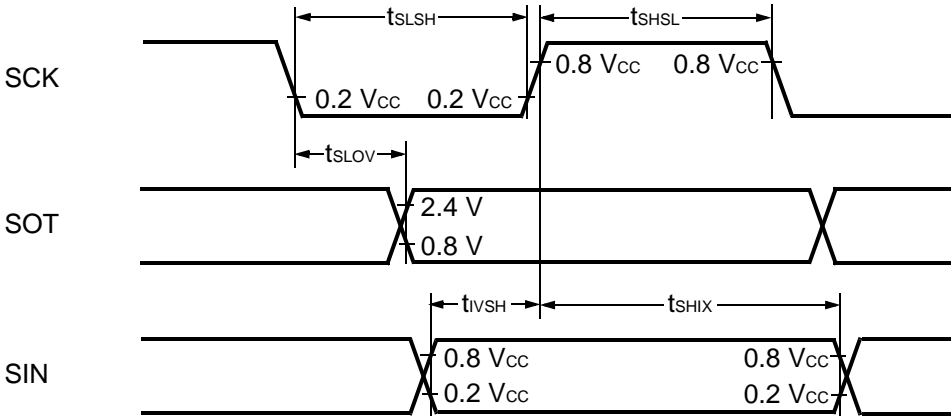
Note:

1. AC characteristic in CLK synchronized mode.
2. C_L is load capacity value of pins when testing.
3. t_{CP} is the machine cycle (Unit: ns).

• Internal Shift Clock Mode



• External Shift Clock Mode



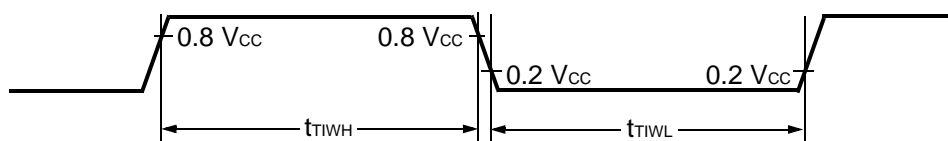
MB90540/545 Series

(10) Timer Related Resource Input Timing

($V_{CC} = 4.5\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_A = -40\text{ }^\circ\text{C to }+85\text{ }^\circ\text{C}$)

Parameter	Symbol	Pin	Condition	Value		Units	Remarks
				Min.	Max.		
Input pulse width	t_{TIWH}	TIN0, TIN1	—	4 t_{CP}	—	ns	
	t_{TIWL}	IN0 to IN7					

• Timer Input Timing

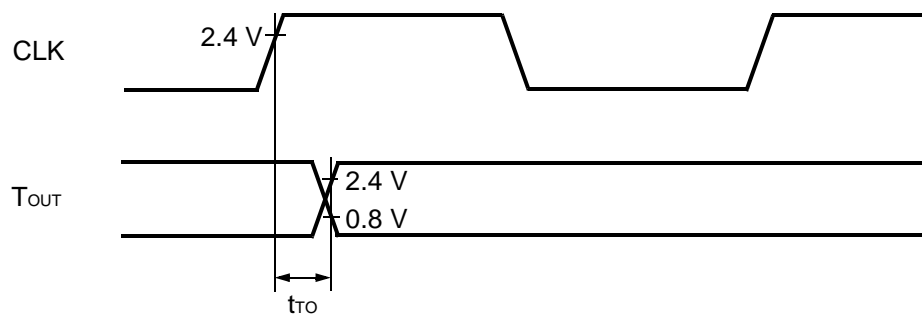


(11) Timer Related Resource Output Timing

($V_{CC} = 4.5\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_A = -40\text{ }^\circ\text{C to }+85\text{ }^\circ\text{C}$)

Parameter	Symbol	Pin	Condition	Value		Units	Remarks
				Min.	Max.		
$\text{CLK} \uparrow \Rightarrow T_{OUT}$ change time	t_{TO}	TOT0 to TOT1, PPG0 to PPG3	—	30	—	ns	

• Timer Output Timing



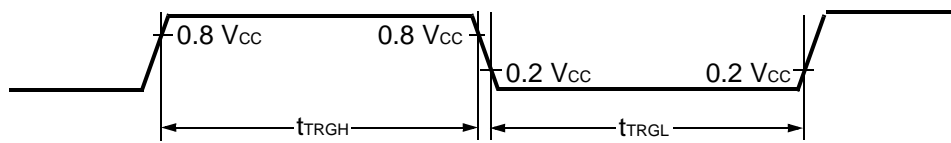
MB90540/545 Series

(12) Trigger Input Timing

($V_{CC} = 4.5$ to 5.5 V, $V_{SS} = 0$ V, $T_A = -40$ °C to $+85$ °C)

Parameter	Symbol	Pin	Condition	Value		Units	Remarks
				Min.	Max.		
Input pulse width	t_{TRGH} t_{TRGL}	INT0 to INT7, ADTG	—	5 t_{CP}	—	ns	

• Trigger Input Timing



MB90540/545 Series

5. A/D Converter

($V_{CC} = AV_{CC} = 5.0\text{ V} \pm 10\%$, $V_{SS} = AV_{SS} = 0\text{ V}$, $3.0\text{ V} \leq AVR_+ - AVR_-$, $T_A = -40\text{ }^\circ\text{C}$ to $+85\text{ }^\circ\text{C}$)

Parameter	Symbol	Pin	Rated Value			Units	Remarks
			Min.	Typ.	Max.		
Resolution	—	—	—	—	10	bit	
Conversion error	—	—	—	—	± 5.0	LSB	
Nonlinearity error	—	—	—	—	± 2.5	LSB	
Differential nonlinearity error	—	—	—	—	± 1.9	LSB	
Zero reading voltage	V_{OT}	AN0 to AN7	$AVR_- - 3.5$	$AVR_- + 0.5$	$AVR_- + 4.5$	mV	
Full scale reading voltage	V_{FST}	AN0 to AN7	$AVR_+ - 6.5$	$AVR_+ - 1.5$	$AVR_+ + 1.5$	mV	
Conversion time	—	—	—	$352t_{CP}$	—	ns	
Sampling time	—	—	—	$64t_{CP}$	—	ns	
Analog port input current	I_{AIN}	AN0 to AN7	-10	—	10	μA	
Analog input voltage range	V_{AIN}	AN0 to AN7	AVR_-	—	AVR_+	V	
Reference voltage range	—	AVR_+	$AVR_- + 2.7$	—	AV_{CC}	V	
	—	AVR_-	0	—	$AVR_+ - 2.7$	V	
Power supply current	I_A	AV_{CC}	—	5	—	mA	
	I_{AH}	AV_{CC}	—	—	5	μA	*1
Reference voltage current	I_R	AVR_+	200	400	600	μA	
	I_{RH}	AVR_+	—	—	5	μA	*1
Offset between input channels	—	AN0 to AN7	—	—	4	LSB	

*1: When not operating A/D converter, this is the current ($V_{CC} = AV_{CC} = AVR_+ = 5.0\text{ V}$) when the CPU is stopped.

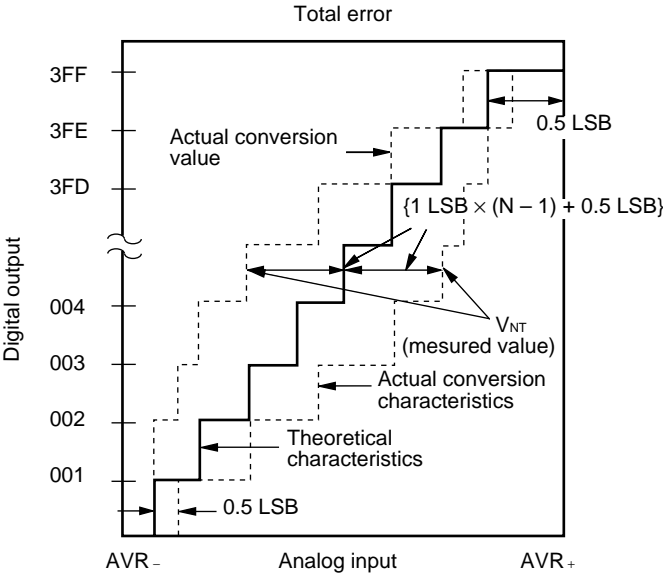
6. A/D Converter Glossary

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

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

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

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



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

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

$$V_{0T} (\text{Theoretical value}) = AVR_- + 0.5 \text{ LSB [V]}$$

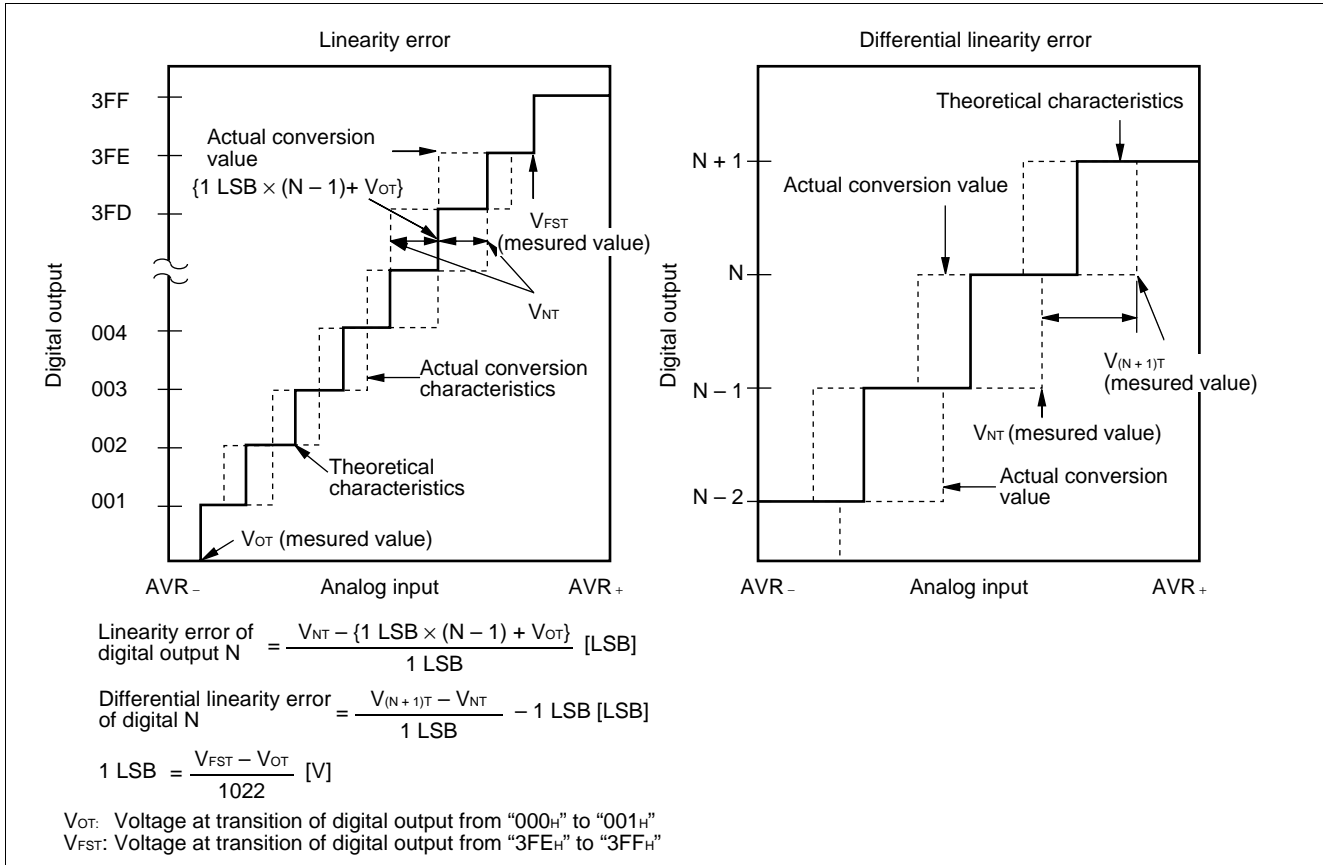
V_{NT} : Voltage at a transition of digital output from (N - 1) to N

$$V_{FST} (\text{Theoretical value}) = AVR_+ - 1.5 \text{ LSB [V]}$$

(Continued)

MB90540/545 Series

(Continued)



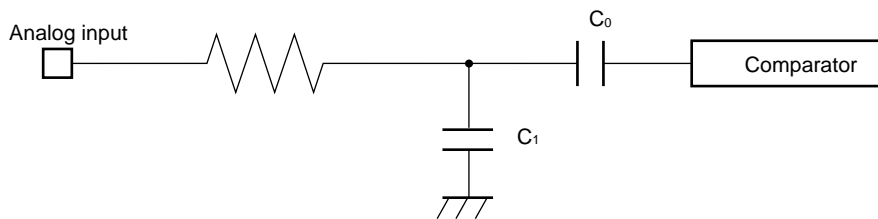
7. Notes on Using A/D Converter

Select the output impedance value for the external circuit of analog input according to the following conditions. Output impedance values of the external circuit of 15 kΩ or lower are recommended.

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

When the output impedance of the external circuit is too high, the sampling period for analog voltages may not be sufficient (sampling period = 4.00 μs @ machine clock of 16 MHz).

• Equipment of analog input circuit model



Note: Listed values must be considered as standards.

• Error

The smaller the $|AVR_+ - AVR_-|$, the greater the error would become relatively.

■ INSTRUCTIONS (340 INSTRUCTIONS)

Table 1 Explanation of Items in Tables of Instructions

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

• Number of execution cycles

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

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

MB90540/545 Series

Table 2 Explanation of Symbols in Tables of Instructions

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

Table 3 Effective Address Fields

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

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

MB90540/545 Series

Table 4 Number of Execution Cycles for Each Type of Addressing

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

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

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

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

Notes: • “(b)”, “(c)”, and “(d)” are used in the “~” (number of states) column and column B (correction value) in the tables of instructions.

- When the external data bus is used, it is necessary to add in the number of wait cycles used for ready input and automatic ready.

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

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

Notes: • When the external data bus is used, it is necessary to add in the number of wait cycles used for ready input and automatic ready.

- Because instruction execution is not slowed down by all program fetches in actuality, these correction values should be used for “worst case” calculations.

Table 7 Transfer Instructions (Byte) [41 Instructions]

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

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

MB90540/545 Series

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

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

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

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

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

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

MB90540/545 Series

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

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

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

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

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

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

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

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

*1: 3 when the result is zero, 7 when an overflow occurs, and 15 normally.

*2: 4 when the result is zero, 8 when an overflow occurs, and 16 normally.

*3: 6 + (a) when the result is zero, 9 + (a) when an overflow occurs, and 19 + (a) normally.

*4: 4 when the result is zero, 7 when an overflow occurs, and 22 normally.

*5: 6 + (a) when the result is zero, 8 + (a) when an overflow occurs, and 26 + (a) normally.

*6: (b) when the result is zero or when an overflow occurs, and 2 × (b) normally.

*7: (c) when the result is zero or when an overflow occurs, and 2 × (c) normally.

*8: 3 when byte (AH) is zero, and 7 when byte (AH) is not zero.

*9: 4 when byte (ear) is zero, and 8 when byte (ear) is not zero.

*10: 5 + (a) when byte (eam) is zero, and 9 + (a) when byte (eam) is not 0.

*11: 3 when word (AH) is zero, and 11 when word (AH) is not zero.

*12: 4 when word (ear) is zero, and 12 when word (ear) is not zero.

*13: 5 + (a) when word (eam) is zero, and 13 + (a) when word (eam) is not zero.

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

MB90540/545 Series

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

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

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

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

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

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

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

MB90540/545 Series

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

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

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

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

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

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

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

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

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

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

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

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

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

MB90540/545 Series

Table 19 Branch 1 Instructions [31 Instructions]

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

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

*2: (b) + 3 × (c)

*3: Read (word) branch address.

*4: W: Save (word) to stack; R: read (word) branch address.

*5: Save (word) to stack.

*6: W: Save (long word) to W stack; R: read (long word) R branch address.

*7: Save (long word) to stack.

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

Table 20 Branch 2 Instructions [19 Instructions]

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

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

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

MB90540/545 Series

Table 21 Other Control Instructions (Byte/Word/Long Word) [36 Instructions]

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

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

*2: $7 + 3 \times (\text{pop count}) + 2 \times (\text{last register number to be popped})$, 7 when rlst = 0 (no transfer register)

*3: $29 + (\text{push count}) - 3 \times (\text{last register number to be pushed})$, 8 when rlst = 0 (no transfer register)

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

*5: Pop count or push count.

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

Table 22 Bit Manipulation Instructions [21 Instructions]

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

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

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

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

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

MB90540/545 Series

Table 24 String Instructions [10 Instructions]

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

m: RW0 value (counter value)

n: Loop count

*1: 5 when RW0 is 0, $4 + 7 \times (RW0)$ for count out, and $7 \times n + 5$ when match occurs

*2: 5 when RW0 is 0, $4 + 8 \times (RW0)$ in any other case

*3: $(b) \times (RW0) + (b) \times (RW0)$ when accessing different areas for the source and destination, calculate (b) separately for each.

*4: $(b) \times n$

*5: $2 \times (RW0)$

*6: $(c) \times (RW0) + (c) \times (RW0)$ when accessing different areas for the source and destination, calculate (c) separately for each.

*7: $(c) \times n$

*8: $2 \times (RW0)$

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

MB90540/545 Series

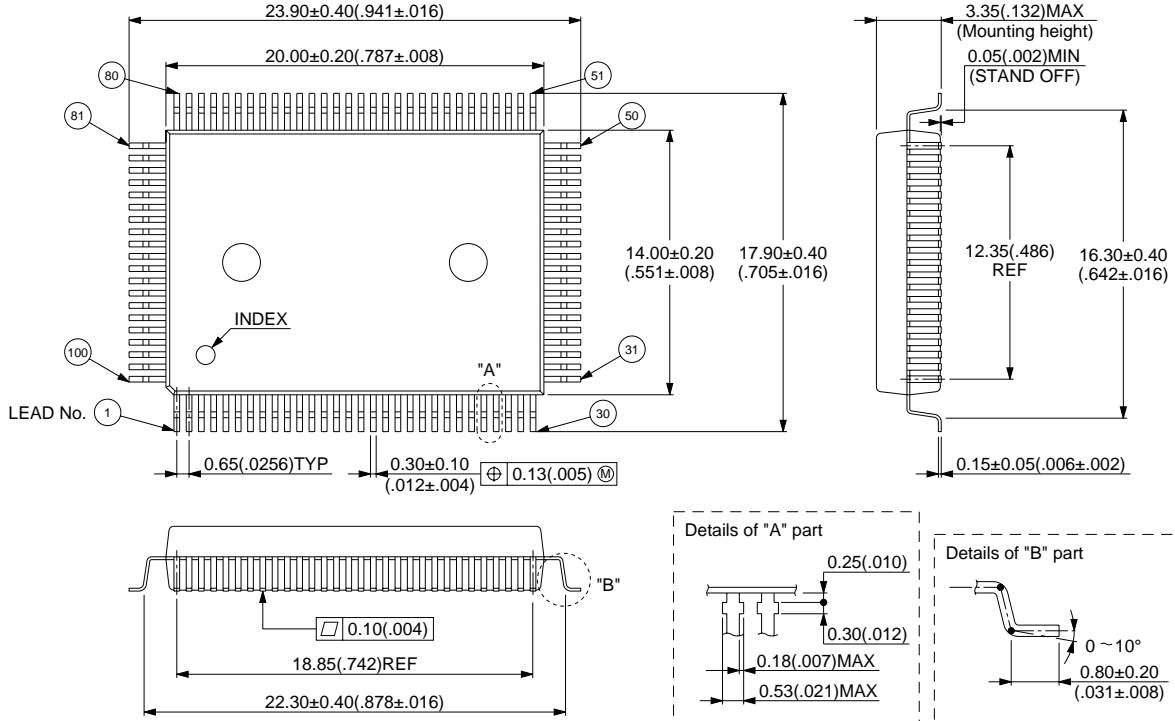
■ ORDERING INFORMATION

Part number	Package	Remarks
MB90543PF MB90F543PF MB90548PF MB90F548PF	100-pin Plastic QFP (FPT-100P-M06)	
MB90543PFF MB90F543PFF MB90548PFF MB90F548PFF	100-pin Plastic LQFP (FPT-100P-M05)	
MB90V540CR	256-pin Ceramic PGA (PGA-256C-A01)	For evaluation

MB90540/545 Series

■ PACKAGE DIMENSIONS

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

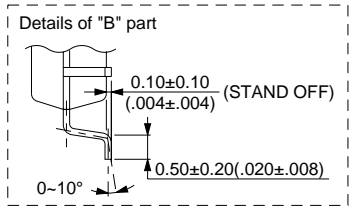
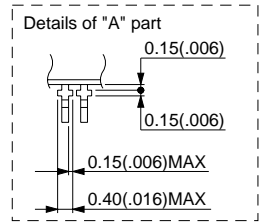
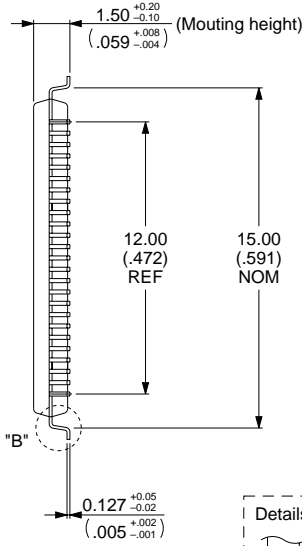
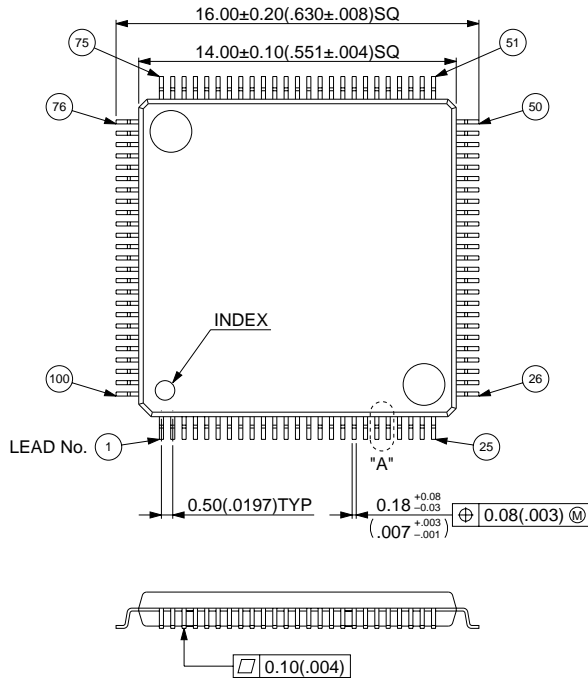


© 1994 FUJITSU LIMITED F100008-3C-2

Dimensions in mm (inches)

MB90540/545 Series

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

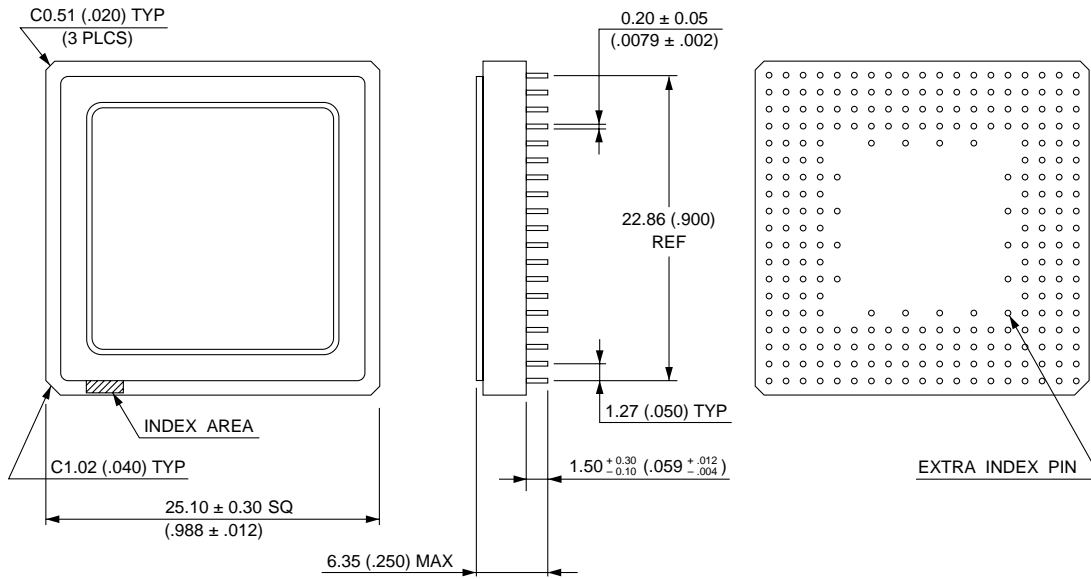


© 1995 FUJITSU LIMITED F100007S-2C-3

Dimensions in mm (inches)

MB90540/545 Series

250-pin Ceramic PGA
(PGA-256C-A01)



© 1994 FUJITSU LIMITED R256001SC-5-3

Dimensions in mm (inches)

MB90540/545 Series

FUJITSU LIMITED

For further information please contact:

Japan

FUJITSU LIMITED
Corporate Global Business Support Division
Electronic Devices
KAWASAKI PLANT, 4-1-1, Kamikodanaka
Nakahara-ku, Kawasaki-shi
Kanagawa 211-8588, Japan
Tel: 81(44) 754-3763
Fax: 81(44) 754-3329

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

North and South America

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

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

<http://www.fujitsumicro.com/>

Europe

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

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

F9909

© FUJITSU LIMITED Printed in Japan

All Rights Reserved.

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

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

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

CAUTION:

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

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

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