

# FLASH MEMORY

CMOS

## 16M (2M × 8/1M × 16) BIT Dual Operation

### MBM29DL16XTE/BE<sub>70/90</sub>

#### ■ FEATURES

- 0.23 μm Process Technology
- Simultaneous Read/Write operations (dual bank)

Multiple devices available with different bank sizes

(Refer to "MBM29DL16XTE/BE Device Bank Divisions Table" in ■GENERAL DESCRIPTION)

Host system can program or erase in one bank, then immediately and simultaneously read from the other bank

Zero latency between read and write operations

Read-while-erase

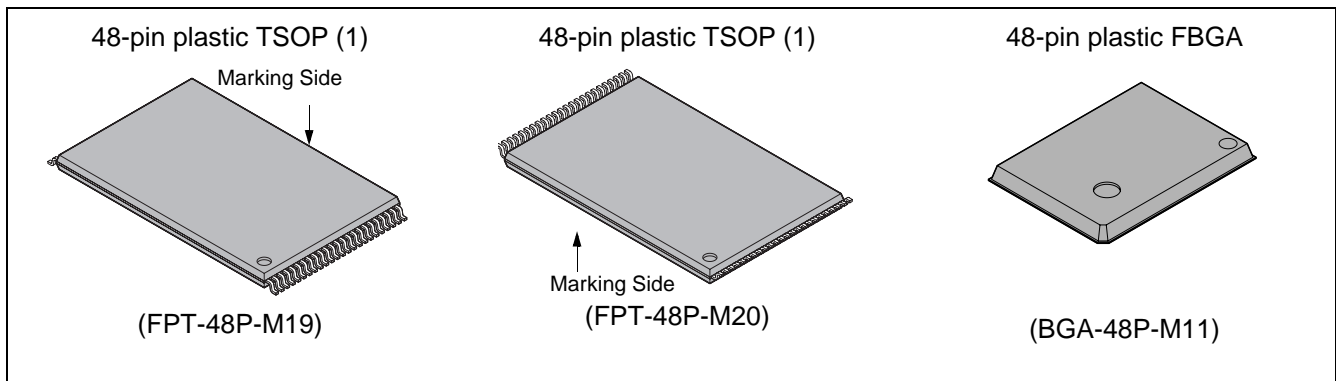
Read-while-program

(Continued)

#### ■ PRODUCT LINE UP

Part No.	MBM29DL16XTE/BE70	MBM29DL16XTE/BE90
Address Access Time (Max)	70 ns	90 ns
$\overline{CE}$ Access Time (Max)	70 ns	90 ns
$\overline{OE}$ Access Time (Max)	30 ns	35 ns
Power Supply Voltage	3.0 V <sup>+0.6V</sup> <sub>-0.3V</sub>	

#### ■ PACKAGES



# MBM29DL16XTE/BE70/90

(Continued)

- **Single 3.0 V read, program, and erase**  
Minimizes system level power requirements
- **Compatible with JEDEC-standard commands**  
Uses same software commands as E<sup>2</sup>PROMs
- **Compatible with JEDEC-standard world-wide pinouts**  
48-pin TSOP(1) (Package suffix: TN – Normal Bend Type, TR – Reversed Bend Type)  
48-pin FBGA (Package suffix: PBT)
- **Minimum 100,000 program/erase cycles**
- **High performance**  
70 ns maximum access time
- **Sector erase architecture**  
Eight 4K word and thirty one 32K word sectors in word mode  
Eight 8K byte and thirty one 64K byte sectors in byte mode  
Any combination of sectors can be concurrently erased. Also supports full chip erase.
- **Boot Code Sector Architecture**  
T = Top sector  
B = Bottom sector
- **HiddenROM region**  
64K byte of HiddenROM, accessible through a new “HiddenROM Enable” command sequence  
Factory serialized and protected to provide a secure electronic serial number (ESN)
- **WP/ACC input pin**  
At V<sub>IL</sub>, allows protection of boot sectors, regardless of sector group protection/unprotection status  
At V<sub>ACC</sub>, increases program performance
- **Embedded Erase™\* Algorithms**  
Automatically pre-programs and erases the chip or any sector
- **Embedded Program™\* Algorithms**  
Automatically writes and verifies data at specified address
- **Data Polling and Toggle Bit feature for detection of program or erase cycle completion**
- **Ready/Busy output (RY/BY)**  
Hardware method for detection of program or erase cycle completion
- **Automatic sleep mode**  
When addresses remain stable, automatically switch themselves to low power mode.
- **Low V<sub>CC</sub> write inhibit ≤ 2.5 V**
- **Program Suspend/Resume**  
Suspends the program operation to allow a read in another sector with in the same device
- **Erase Suspend/Resume**  
Suspends the erase operation to allow a read data and/or program in another sector within the same device
- **Sector group protection**  
Hardware method disables any combination of sector groups from program or erase operations
- **Sector Group Protection Set function by Extended sector group protection command**
- **Fast Programming Function by Extended Command**
- **Temporary sector group unprotection**  
Temporary sector group unprotection via the  $\overline{\text{RESET}}$  pin.
- **In accordance with CFI (Common Flash Memory Interface)**

\*: Embedded Erase™ and Embedded Program™ are trademarks of Advanced Micro Devices, Inc.

## ■ GENERAL DESCRIPTION

The MBM29DL16XTE/BE are a 16M-bit, 3.0 V-only Flash memory organized as 2M bytes of 8 bits each or 1M words of 16 bits each. The MBM29DL16XTE/BE are offered in a 48-pin TSOP(1) and 48-pin FBGA Package. These devices are designed to be programmed in-system with the standard system 3.0 V  $V_{CC}$  supply. 12.0 V  $V_{PP}$  and 5.0 V  $V_{CC}$  are not required for write or erase operations. The devices can also be reprogrammed in standard EPROM programmers.

MBM29DL16XTE/BE are organized into two banks, Bank 1 and Bank 2, which are considered to be two separate memory arrays for operations. It is the Fujitsu's standard 3 V only Flash memories with the additional capability of allowing a normal non-delayed read access from a non-busy bank of the array while an embedded write (either a program or an erase) operation is simultaneously taking place on the other bank.

In the MBM29DL16XTE/BE, a new design concept is implemented, so called "Sliding Bank Architecture". Under this concept, the MBM29DL16XTE/BE can be produced a series of devices with different Bank 1/Bank 2 size combinations; 0.5 Mb/15.5 Mb, 2 Mb/14 Mb, 4 Mb/12 Mb, 8 Mb/8 Mb.

The standard MBM29DL16XTE/BE offer access times 70 ns and 90 ns, allowing operation of high-speed microprocessors without wait states. To eliminate bus contention the devices have separate chip enable ( $\overline{CE}$ ), write enable ( $\overline{WE}$ ), and output enable ( $\overline{OE}$ ) controls.

The MBM29DL16XTE/BE are pin and command set compatible with JEDEC standard E<sup>2</sup>PROMs. Commands are written to the command register using standard microprocessor write timings. Register contents serve as input to an internal state-machine which controls the erase and programming circuitry. Write cycles also internally latch addresses and data needed for the programming and erase operations. Reading data out of the devices is similar to reading from 5.0 V and 12.0 V Flash or EPROM devices.

The MBM29DL16XTE/BE are programmed by executing the program command sequence. This will invoke the Embedded Program Algorithm which is an internal algorithm that automatically times the program pulse widths and verifies proper cell margin. Typically, each sector can be programmed and verified in about 0.5 seconds. Erase is accomplished by executing the erase command sequence. This will invoke the Embedded Erase Algorithm which is an internal algorithm that automatically preprograms the array if it is not already programmed before executing the erase operation. During erase, the devices automatically time the erase pulse widths and verify proper cell margin.

A sector is typically erased and verified in 1.0 second. (If already completely preprogrammed.)

The devices also feature a sector erase architecture. The sector mode allows each sector to be erased and reprogrammed without affecting other sectors. The MBM29DL16XTE/BE are erased when shipped from the factory.

The devices feature single 3.0 V power supply operation for both read and write functions. Internally generated and regulated voltages are provided for the program and erase operations. A low  $V_{CC}$  detector automatically inhibits write operations on the loss of power. The end of program or erase is detected by  $\overline{\text{Data}}$  Polling of  $DQ_7$ , by the Toggle Bit feature on  $DQ_6$ , or the  $\overline{RY/BY}$  output pin. Once the end of a program or erase cycle has been completed, the devices internally reset to the read mode.

Fujitsu's Flash technology combines years of EPROM and E<sup>2</sup>PROM experience to produce the highest levels of quality, reliability, and cost effectiveness. The MBM29DL16XTE/BE memories electrically erase the entire chip or all bits within a sector simultaneously via Fowler-Nordhiem tunneling. The bytes/words are programmed one byte/word at a time using the EPROM programming mechanism of hot electron injection.

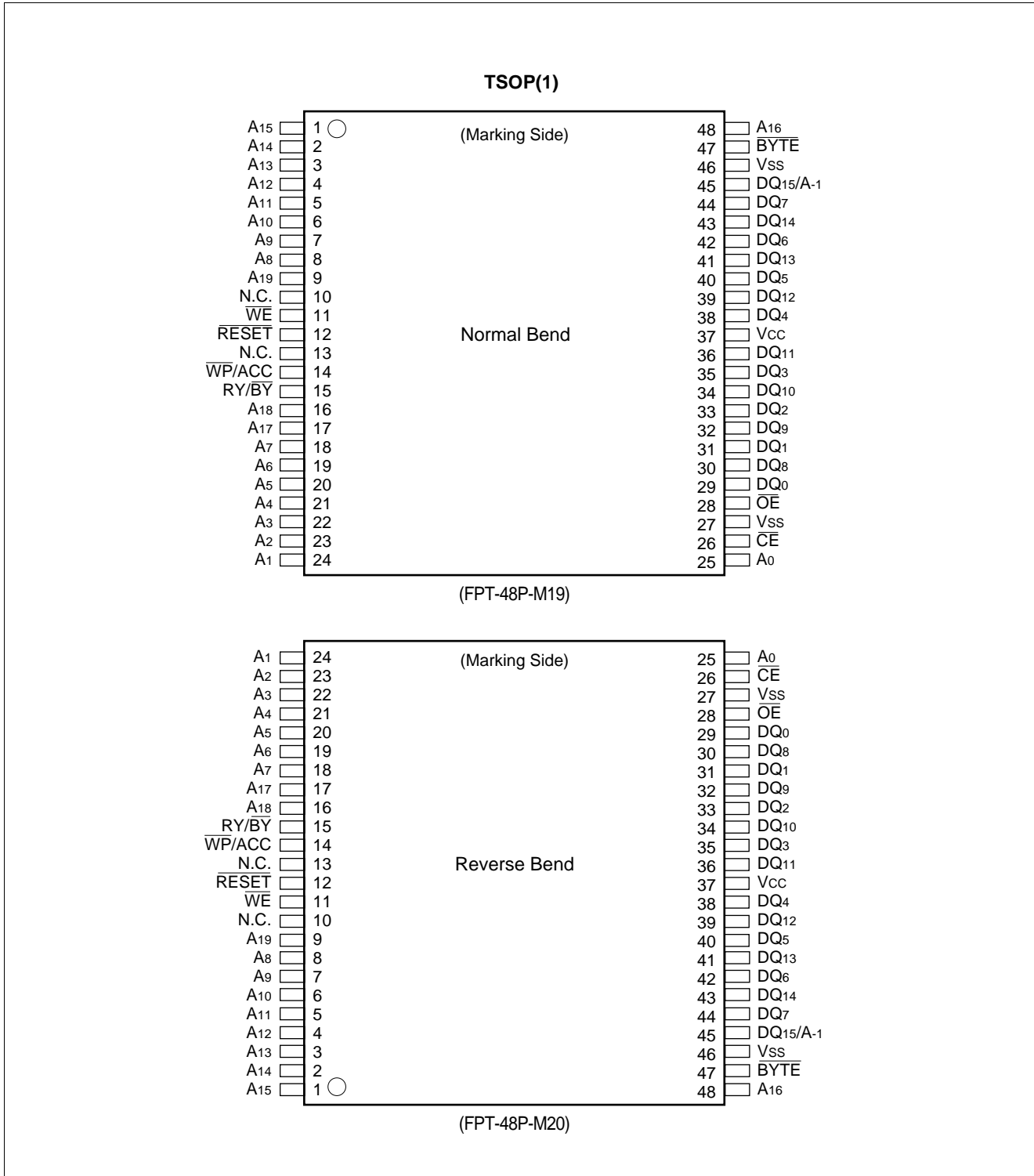
# MBM29DL16XTE/BE70/90

**MBM29DL16XTE/BE Device Bank Divisions Table**

Device Part Number	Organization	Bank 1		Bank 2	
		Megabits	Sector Sizes	Megabits	Sector Sizes
MBM29DL161TE/BE	× 8/× 16	0.5 Mbit	Eight 8K byte/4K word	15.5 Mbit	Thirty-one 64K byte/32K word
MBM29DL162TE/BE		2 Mbit	Eight 8K byte/4K word, three 64K byte/32K word	14 Mbit	Twenty-eight 64K byte/32K word
MBM29DL163TE/BE		4 Mbit	Eight 8K byte/4K word, seven 64K byte/32K word	12 Mbit	Twenty-four 64K byte/32K word
MBM29DL164TE/BE		8 Mbit	Eight 8K byte/4K word, fifteen 64K byte/32K word	8 Mbit	Sixteen 64K byte/32K word

# MBM29DL16XTE/BE70/90

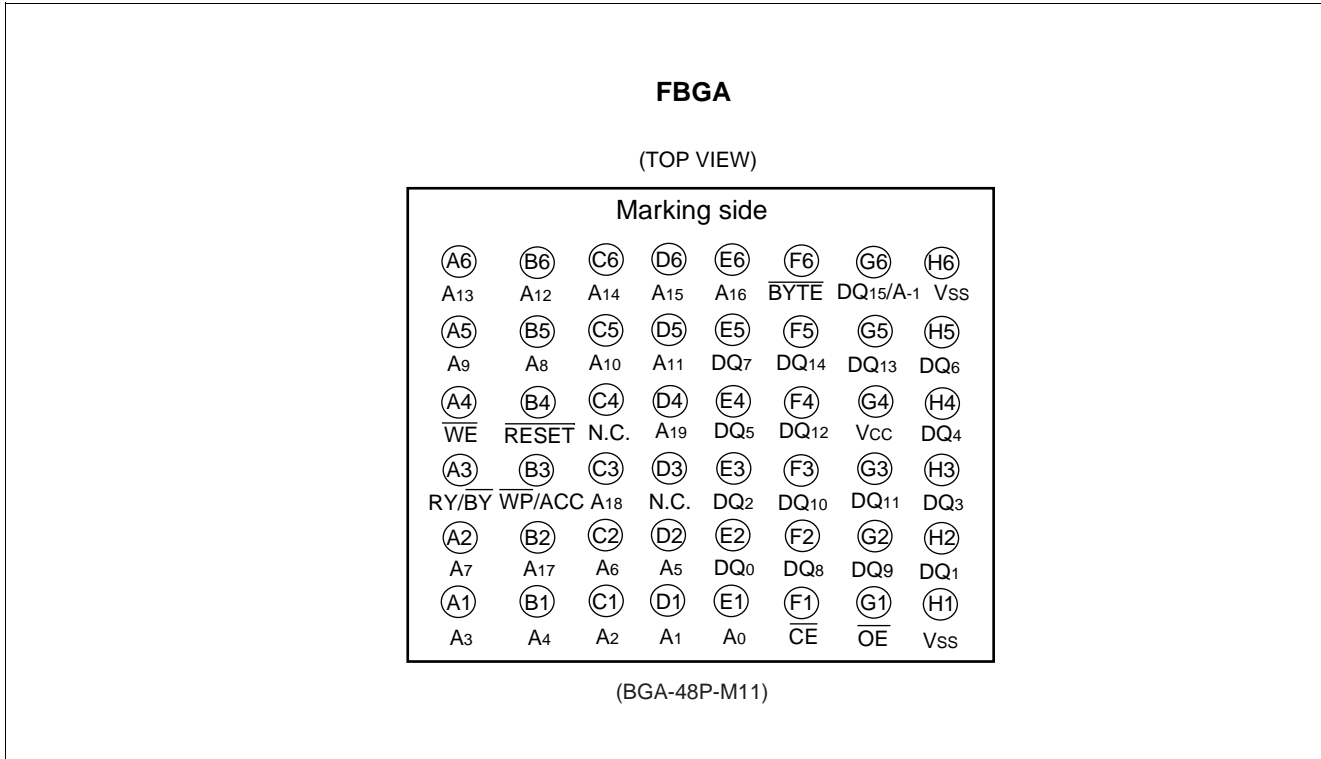
## ■ PIN ASSIGNMENTS



(Continued)

# MBM29DL16XTE/BE70/90

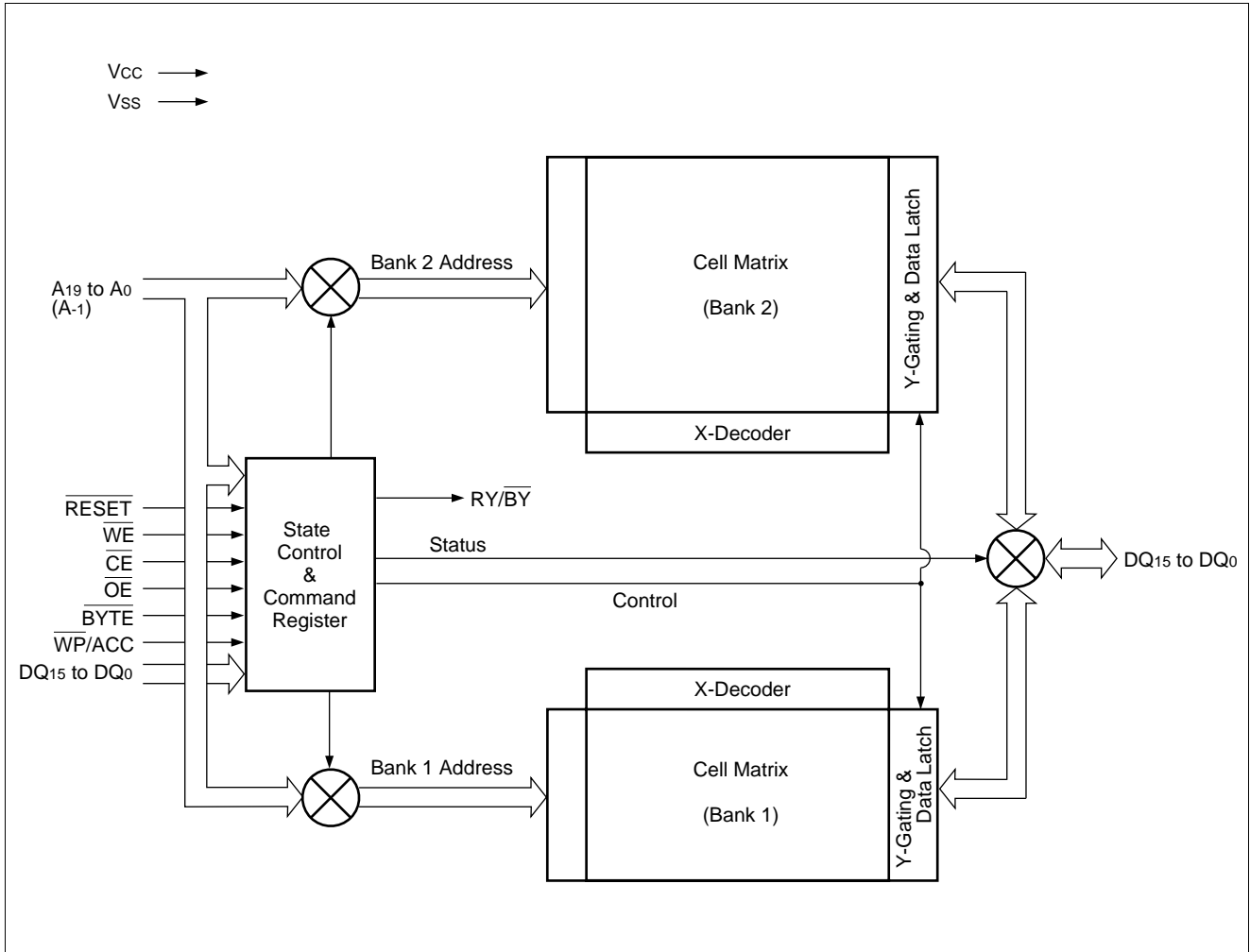
(Continued)



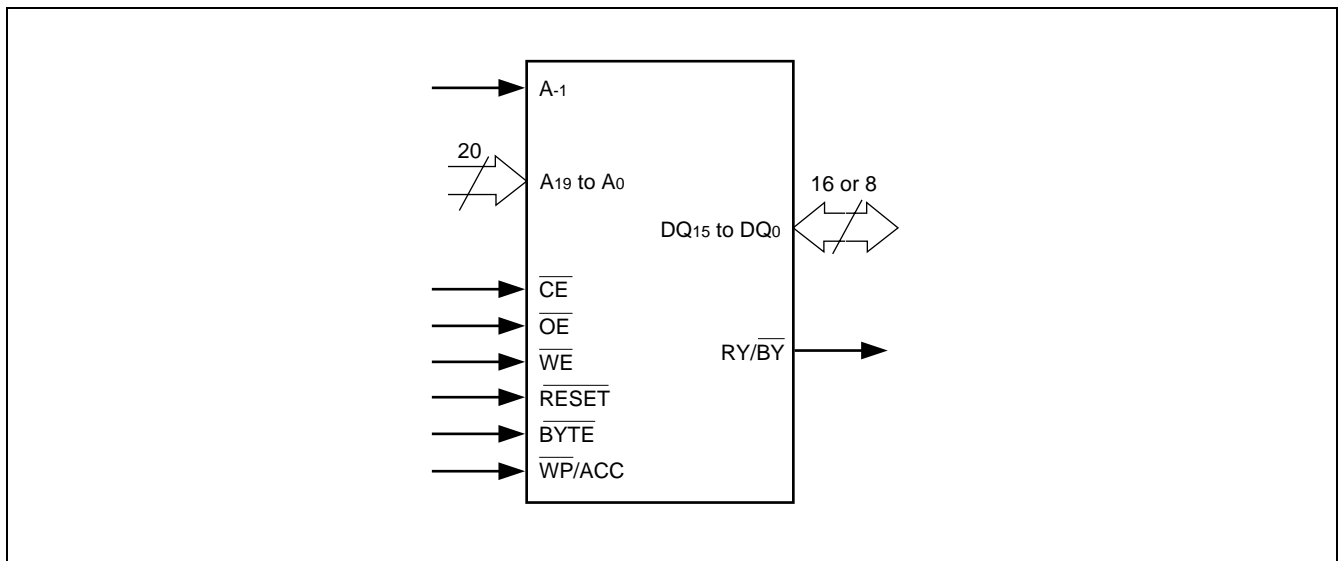
## ■ PIN DESCRIPTIONS

Pin Name	Function	Pin Name	Function
A <sub>19</sub> to A <sub>0</sub> , A-1	Address Input	RY/BY	Ready/Busy Output
DQ <sub>15</sub> to DQ <sub>0</sub>	Data Input/Output	BYTE	Selects 8-bit or 16-bit mode
CE	Chip Enable	WP/ACC	Hardware Write Protection/ Program Acceleration
OE	Output Enable	V <sub>ss</sub>	Device Ground
WE	Write Enable	V <sub>cc</sub>	Device Power Supply
RESET	Hardware Reset Pin/ Temporary Sector Group Unprotection	N.C.	No Internal Connection

## ■ BLOCK DIAGRAM



## ■ LOGIC SYMBOL



# MBM29DL16XTE/BE70/90

## ■ DEVICE BUS OPERATION

MBM29DL16XTE/BE User Bus Operations Table ( $\overline{\text{BYTE}} = V_{IH}$ )

Operation	$\overline{\text{CE}}$	$\overline{\text{OE}}$	$\overline{\text{WE}}$	A <sub>0</sub>	A <sub>1</sub>	A <sub>6</sub>	A <sub>9</sub>	DQ <sub>15</sub> to DQ <sub>0</sub>	$\overline{\text{RESET}}$	$\overline{\text{WP/ACC}}$
Auto-Select Manufacturer Code* <sup>1</sup>	L	L	H	L	L	L	V <sub>ID</sub>	Code	H	X
Auto-Select Device Code* <sup>1</sup>	L	L	H	H	L	L	V <sub>ID</sub>	Code	H	X
Read* <sup>3</sup>	L	L	H	A <sub>0</sub>	A <sub>1</sub>	A <sub>6</sub>	A <sub>9</sub>	D <sub>OUT</sub>	H	X
Standby	H	X	X	X	X	X	X	High-Z	H	X
Output Disable	L	H	H	X	X	X	X	High-Z	H	X
Write (Program/Erase)	L	H	L	A <sub>0</sub>	A <sub>1</sub>	A <sub>6</sub>	A <sub>9</sub>	D <sub>IN</sub>	H	X
Enable Sector Group Protection* <sup>2, *4</sup>	L	V <sub>ID</sub>	$\overline{\square}$	L	H	L	V <sub>ID</sub>	X	H	X
Verify Sector Group Protection* <sup>2, *4</sup>	L	L	H	L	H	L	V <sub>ID</sub>	Code	H	X
Temporary Sector Group Unprotection* <sup>5</sup>	X	X	X	X	X	X	X	X	V <sub>ID</sub>	X
Reset (Hardware) / Standby	X	X	X	X	X	X	X	High-Z	L	X
Boot Block Sector Write Protection	X	X	X	X	X	X	X	X	X	L

**Legend:** L = V<sub>IL</sub>, H = V<sub>IH</sub>, X = V<sub>IL</sub> or V<sub>IH</sub>,  $\overline{\square}$  = Pulse input. See ■DC CHARACTERISTICS for voltage levels.

\*1: Manufacturer and device codes are accessed via a command register write sequence. See “MBM29DL16XTE/BE Command Definitions Table”.

\*2: Refer to “Sector Group Protection” in ■FUNCTIONAL DESCRIPTION.

\*3:  $\overline{\text{WE}}$  can be V<sub>IL</sub> if  $\overline{\text{OE}}$  is V<sub>IL</sub>,  $\overline{\text{OE}}$  at V<sub>IH</sub> initiates the write operations.

\*4: V<sub>CC</sub> = +2.7 V to +3.6 V

\*5: Also used for the extended sector group protection.

MBM29DL16XTE/BE User Bus Operations Table ( $\overline{\text{BYTE}} = V_{IL}$ )

Operation	$\overline{\text{CE}}$	$\overline{\text{OE}}$	$\overline{\text{WE}}$	DQ <sub>15</sub> / A <sub>-1</sub>	A <sub>0</sub>	A <sub>1</sub>	A <sub>6</sub>	A <sub>9</sub>	DQ <sub>7</sub> to DQ <sub>0</sub>	$\overline{\text{RESET}}$	$\overline{\text{WP/ACC}}$
Auto-Select Manufacturer Code* <sup>1</sup>	L	L	H	L	L	L	L	V <sub>ID</sub>	Code	H	X
Auto-Select Device Code* <sup>1</sup>	L	L	H	L	H	L	L	V <sub>ID</sub>	Code	H	X
Read* <sup>3</sup>	L	L	H	A <sub>-1</sub>	A <sub>0</sub>	A <sub>1</sub>	A <sub>6</sub>	A <sub>9</sub>	D <sub>OUT</sub>	H	X
Standby	H	X	X	X	X	X	X	X	High-Z	H	X
Output Disable	L	H	H	X	X	X	X	X	High-Z	H	X
Write (Program/Erase)	L	H	L	A <sub>-1</sub>	A <sub>0</sub>	A <sub>1</sub>	A <sub>6</sub>	A <sub>9</sub>	D <sub>IN</sub>	H	X
Enable Sector Group Protection* <sup>2, *4</sup>	L	V <sub>ID</sub>	$\overline{\square}$	L	L	H	L	V <sub>ID</sub>	X	H	X
Verify Sector Group Protection* <sup>2, *4</sup>	L	L	H	L	L	H	L	V <sub>ID</sub>	Code	H	X
Temporary Sector Group Unprotection* <sup>5</sup>	X	X	X	X	X	X	X	X	X	V <sub>ID</sub>	X
Reset (Hardware) / Standby	X	X	X	X	X	X	X	X	High-Z	L	X
Boot Block Sector Write Protection	X	X	X	X	X	X	X	X	X	X	L

**Legend:** L = V<sub>IL</sub>, H = V<sub>IH</sub>, X = V<sub>IL</sub> or V<sub>IH</sub>,  $\overline{\square}$  = Pulse input. See ■DC CHARACTERISTICS for voltage levels.

\*1: Manufacturer and device codes are accessed via a command register write sequence. See “MBM29DL16XTE/BE Command Definitions Table”.

\*2: Refer to “Sector Group Protection” in ■FUNCTIONAL DESCRIPTION.

\*3:  $\overline{\text{WE}}$  can be V<sub>IL</sub> if  $\overline{\text{OE}}$  is V<sub>IL</sub>,  $\overline{\text{OE}}$  at V<sub>IH</sub> initiates the write operations.

\*4: V<sub>CC</sub> = +2.7 V to +3.6 V

\*5: Also used for the extended sector group protection.



# MBM29DL16XTE/BE70/90

## MBM29DL16XTE/BE Command Definitions Table

Command Sequence		Bus Write Cycles Req'd	First Bus Write Cycle		Second Bus Write Cycle		Third Bus Write Cycle		Fourth Bus Read/Write Cycle		Fifth Bus Write Cycle		Sixth Bus Write Cycle	
			Addr.	Data	Addr.	Data	Addr.	Data	Addr.	Data	Addr.	Data	Addr.	Data
Read/Reset*1	Word	1	XXXh	F0h	—	—	—	—	—	—	—	—	—	—
	Byte													
Read/Reset*1	Word	3	555h	AAh	2AAh	55h	555h	F0h	RA*7	RD*7	—	—	—	—
	Byte		AAAh		555h		AAAh							
Autoselect	Word	3	555h	AAh	2AAh	55h	(BA) 555h	90h	IA*7	ID*7	—	—	—	—
	Byte		AAAh		555h		(BA) AAAh							
Program	Word	4	555h	AAh	2AAh	55h	555h	A0h	PA	PD	—	—	—	—
	Byte		AAAh		555h		AAAh							
Program Suspend		1	BA	B0h	—	—	—	—	—	—	—	—	—	—
Program Resume		1	BA	30h	—	—	—	—	—	—	—	—	—	—
Chip Erase	Word	6	555h	AAh	2AAh	55h	555h	80h	555h	AAh	2AAh	55h	555h	10h
	Byte		AAAh		555h		AAAh		AAAh		555h		AAAh	
Sector Erase	Word	6	555h	AAh	2AAh	55h	555h	80h	555h	AAh	2AAh	55h	SA	30h
	Byte		AAAh		555h		AAAh		AAAh		555h			
Erase Suspend		1	BA	B0h	—	—	—	—	—	—	—	—	—	—
Erase Resume		1	BA	30h	—	—	—	—	—	—	—	—	—	—
Set to Fast Mode	Word	3	555h	AAh	2AAh	55h	555h	20h	—	—	—	—	—	—
	Byte		AAAh		555h		AAAh							
Fast Program*2	Word	2	XXXh	A0h	PA	PD	—	—	—	—	—	—	—	—
	Byte													
Reset from Fast Mode*2	Word	2	BA	90h	XXXh	*6 F0h	—	—	—	—	—	—	—	—
	Byte													
Extended Sector Group Protection*3	Word	3	XXXh	60h	SPA	60h	SPA	40h	SPA*7	SD*7	—	—	—	—
	Byte													
Query *4	Word	1	(BA) 55h	98h	—	—	—	—	—	—	—	—	—	—
	Byte		(BA) AAh											
HiddenROM Entry	Word	3	555h	AAh	2AAh	55h	555h	88h	—	—	—	—	—	—
	Byte		AAAh		555h		AAAh							
HiddenROM Program*5	Word	4	555h	AAh	2AAh	55h	555h	A0h	PA (HRA)	PD	—	—	—	—
	Byte		AAAh		555h		AAAh							
HiddenROM Erase*5	Word	6	555h	AAh	2AAh	55h	555h	80h	555h	AAh	2AAh	55h	HRA	30h
	Byte		AAAh		555h		AAAh		AAAh		555h			
HiddenROM Exit*5	Word	4	555h	AAh	2AAh	55h	(HRBA) 555h	90h	XXXh	00h	—	—	—	—
	Byte		AAAh		555h		(HRBA) AAAh							

# MBM29DL16XTE/BE70/90

- Notes:
- Address bits  $A_{19}$  to  $A_{11} = X = \text{“H”}$  or  $\text{“L”}$  for all address commands except or Program Address (PA), Sector Address (SA), and Bank Address (BA).
  - Bus operations are defined in “MBM29DL16XTE/BE User Bus Operations Tables ( $\overline{\text{BYTE}} = V_{IH}$  and  $\overline{\text{BYTE}} = V_{IL}$ )”.
  - RA: Address of the memory location to be read
  - IA : Autoselect read address that sets both the bank address specified at ( $A_{19}, A_{18}, A_{17}, A_{16}, A_{15}$ ) and all the other  $A_6, A_1, A_0, (A_{-1})$ .
  - PA: Address of the memory location to be programmed  
Addresses are latched on the falling edge of the write pulse.
  - SA: Address of the sector to be erased. The combination of  $A_{19}, A_{18}, A_{17}, A_{16}, A_{15}, A_{14}, A_{13}$ , and  $A_{12}$  will uniquely select any sector.
  - BA: Bank Address ( $A_{19}$  to  $A_{15}$ )
  - RD: Data read from location RA during read operation.
  - ID : Device code/manufacture code for the address located by IA.
  - PD: Data to be programmed at location PA. Data is latched on the rising edge of write pulse.
  - SPA: Sector group address to be protected. Set sector group address (SGA) and ( $A_6, A_1, A_0$ ) = (0, 1, 0).
  - SD: Sector group protection verify data. Output 01h at protected sector group addresses and output 00h at unprotected sector group addresses.
  - HRA: Address of the HiddenROM area  
29DL16XTE (Top Boot Type) Word Mode: 0F8000h to 0FFFFFFh  
Byte Mode: 1F0000h to 1FFFFFFh  
29DL16XBE (Bottom Boot Type) Word Mode: 000000h to 007FFFh  
Byte Mode: 000000h to 00FFFFh
  - HRBA: Bank Address of the HiddenROM area  
29DL16XTE (Top Boot Type) :  $A_{19} = A_{18} = A_{17} = A_{16} = A_{15} = V_{IH}$   
29DL16XBE (Bottom Boot Type) :  $A_{19} = A_{18} = A_{17} = A_{16} = A_{15} = V_{IL}$
  - The system should generate the following address patterns:  
Word Mode: 555h or 2AAh to addresses  $A_{10}$  to  $A_0$   
Byte Mode: AAh or 555h to addresses  $A_{10}$  to  $A_0$  and  $A_{-1}$
  - Both Read/Reset commands are functionally equivalent, resetting the device to the read mode.
  - Command combinations not described in Command Definitions table are illegal.

\*1: Both of these reset commands are equivalent.

\*2: This command is valid during Fast Mode.

\*3: This command is valid while  $\overline{\text{RESET}} = V_{ID}$  (except during HiddenROM MODE).

\*4: The valid addresses are  $A_6$  to  $A_0$ .

\*5: This command is valid during HiddenROM mode.

\*6: The data “00h” is also acceptable.

\*7: The fourth bus cycle is only for read.

# MBM29DL16XTE/BE70/90

**MBM29DL161TE/BE Sector Group Protection Verify Autoselect Codes Table**

Type		A <sub>19</sub> to A <sub>12</sub>	A <sub>6</sub>	A <sub>1</sub>	A <sub>0</sub>	A <sub>-1</sub> *1	Code (HEX)
Manufacturer's Code	Byte	BA*3	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IL</sub>	04h
	Word					X	0004h
Device Code	MBM29DL161TE	BA*3	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	36h
						Word	X
	MBM29DL161BE	BA*3	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	39h
						Word	X
Sector Group Protection	Byte	Sector Group Addresses	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	V <sub>IL</sub>	01h*2
	Word					X	0001h*2

\*1: A<sub>-1</sub> is for Byte mode. At Byte mode, DQ<sub>14</sub> to DQ<sub>8</sub> are High-Z and DQ<sub>15</sub> is A<sub>-1</sub>, the lowest address.

\*2: Outputs 01h at protected sector group addresses and outputs 00h at unprotected sector group addresses.

\*3: When V<sub>ID</sub> is applied to A<sub>9</sub>, both Bank1 and Bank2 are put into Autoselect mode, which makes simultaneous operation unable to be executed. Consequently, specifying the bank address is not required. However, the bank address needs to be indicated when Autoselect mode is read out at command mode, because then it enables to activate simultaneous operation.

**Extended Autoselect Code Table**

Type		Code	DQ <sub>15</sub>	DQ <sub>14</sub>	DQ <sub>13</sub>	DQ <sub>12</sub>	DQ <sub>11</sub>	DQ <sub>10</sub>	DQ <sub>9</sub>	DQ <sub>8</sub>	DQ <sub>7</sub>	DQ <sub>6</sub>	DQ <sub>5</sub>	DQ <sub>4</sub>	DQ <sub>3</sub>	DQ <sub>2</sub>	DQ <sub>1</sub>	DQ <sub>0</sub>
Manufacturer's Code	(B)*	04h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	0	0	0	1	0	0
	(W)	0004h	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Device Code	MBM29DL161TE	(B)*	36h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	1	1	0	1	1	0
		(W)	2236h	0	0	1	0	0	0	1	0	0	0	1	1	0	1	1
	MBM29DL161BE	(B)*	39h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	1	1	1	0	0	1
		(W)	2239h	0	0	1	0	0	0	1	0	0	0	1	1	1	0	0
Sector Group Protection	(B)*	01h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	0	0	0	0	0	1
	(W)	0001h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

(B) : Byte mode

(W) : Word mode

HI-Z : High-Z

\*: At Byte mode, DQ<sub>14</sub> to DQ<sub>8</sub> are High-Z and DQ<sub>15</sub> is A<sub>-1</sub>, the lowest address.

# MBM29DL16XTE/BE70/90

**MBM29DL162TE/BE Sector Group Protection Verify Autoselect Codes Table**

Type		A <sub>19</sub> to A <sub>12</sub>	A <sub>6</sub>	A <sub>1</sub>	A <sub>0</sub>	A <sub>-1</sub> *1	Code (HEX)
Manufacturer's Code	Byte	BA*3	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IL</sub>	04h
	Word					X	0004h
Device Code	MBM29DL162TE	BA*3	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	2Dh
						Word	X
	MBM29DL162BE	BA*3	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	2Eh
						Word	X
Sector Group Protection		Sector Group Addresses	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	V <sub>IL</sub>	01h*2
						X	0001h*2

\*1 : A<sub>-1</sub> is for Byte mode. At Byte mode, DQ<sub>14</sub> to DQ<sub>8</sub> are High-Z and DQ<sub>15</sub> is A<sub>-1</sub>, the lowest address.

\*2 : Outputs 01h at protected sector group addresses and outputs 00h at unprotected sector group addresses.

\*3 : When V<sub>ID</sub> is applied to A<sub>9</sub>, both Bank 1 and Bank 2 are put into Autoselect mode, which makes simultaneous operation unable to be executed. Consequently, specifying the bank address is not required. However, the bank address needs to be indicated when Autoselect mode is read out at command mode, because then it enables to activate simultaneous operation.

**Extended Autoselect Code Table**

Type		Code	DQ <sub>15</sub>	DQ <sub>14</sub>	DQ <sub>13</sub>	DQ <sub>12</sub>	DQ <sub>11</sub>	DQ <sub>10</sub>	DQ <sub>9</sub>	DQ <sub>8</sub>	DQ <sub>7</sub>	DQ <sub>6</sub>	DQ <sub>5</sub>	DQ <sub>4</sub>	DQ <sub>3</sub>	DQ <sub>2</sub>	DQ <sub>1</sub>	DQ <sub>0</sub>
Manufacturer's Code	(B)*	04h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	0	0	0	1	0	0
	(W)	0004h	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Device Code	MBM29DL162TE	(B)*	2Dh	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	1	0	1	1	0	1
		(W)	222Dh	0	0	1	0	0	0	1	0	0	0	1	0	1	1	0
	MBM29DL162BE	(B)*	2Eh	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	1	0	1	1	1	0
		(W)	222Eh	0	0	1	0	0	0	1	0	0	0	1	0	1	1	1
Sector Group Protection		(B)*	01h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	0	0	0	0	0	1
		(W)	0001h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(B) : Byte mode

(W) : Word mode

HI-Z : High-Z

\* : At Byte mode, DQ<sub>14</sub> to DQ<sub>8</sub> are High-Z and DQ<sub>15</sub> is A<sub>-1</sub>, the lowest address.

# MBM29DL16XTE/BE70/90

**MBM29DL163TE/BE Sector Group Protection Verify Autoselect Codes Table**

Type		A <sub>19</sub> to A <sub>12</sub>	A <sub>6</sub>	A <sub>1</sub>	A <sub>0</sub>	A <sub>-1</sub> * <sup>1</sup>	Code (HEX)
Manufacturer's Code	Byte	BA* <sup>3</sup>	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IL</sub>	04h
	Word					X	0004h
Device Code	MBM29DL163TE	BA* <sup>3</sup>	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	28h
						Word	X
	MBM29DL163BE	BA* <sup>3</sup>	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	2Bh
						Word	X
Sector Group Protection		Sector Group Addresses	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	V <sub>IL</sub>	01h* <sup>2</sup>
						Word	X

\*1 : A<sub>-1</sub> is for Byte mode. At Byte mode, DQ<sub>14</sub> to DQ<sub>8</sub> are High-Z and DQ<sub>15</sub> is A<sub>-1</sub>, the lowest address.

\*2 : Outputs 01h at protected sector group addresses and outputs 00h at unprotected sector group addresses.

\*3 : When V<sub>ID</sub> is applied to A<sub>9</sub>, both Bank 1 and Bank 2 are put into Autoselect mode, which makes simultaneous operation unable to be executed. Consequently, specifying the bank address is not required. However, the bank address needs to be indicated when Autoselect mode is read out at command mode, because then it enables to activate simultaneous operation.

**Extended Autoselect Code Table**

Type		Code	DQ <sub>15</sub>	DQ <sub>14</sub>	DQ <sub>13</sub>	DQ <sub>12</sub>	DQ <sub>11</sub>	DQ <sub>10</sub>	DQ <sub>9</sub>	DQ <sub>8</sub>	DQ <sub>7</sub>	DQ <sub>6</sub>	DQ <sub>5</sub>	DQ <sub>4</sub>	DQ <sub>3</sub>	DQ <sub>2</sub>	DQ <sub>1</sub>	DQ <sub>0</sub>	
Manufacturer's Code	(B)*	04h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	0	0	0	1	0	0	
	(W)	0004h	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
Device Code	MBM29DL163TE	(B)*	28h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	1	0	1	0	0	0	
		(W)	2228h	0	0	1	0	0	0	1	0	0	0	1	0	1	0	0	0
	MBM29DL163BE	(B)*	2Bh	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	1	0	1	0	1	1	1
		(W)	222Bh	0	0	1	0	0	0	1	0	0	0	1	0	1	0	1	1
Sector Group Protection		(B)*	01h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	0	0	0	0	0	1	
		(W)	0001h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	

(B) : Byte mode

(W) : Word mode

HI-Z : High-Z

\* : At Byte mode, DQ<sub>14</sub> to DQ<sub>8</sub> are High-Z and DQ<sub>15</sub> is A<sub>-1</sub>, the lowest address.

# MBM29DL16XTE/BE70/90

**MBM29DL164TE/BE Sector Group Protection Verify Autoselect Codes Table**

Type		A <sub>19</sub> to A <sub>12</sub>	A <sub>6</sub>	A <sub>1</sub>	A <sub>0</sub>	A <sub>-1</sub> * <sup>1</sup>	Code (HEX)
Manufacturer's Code	Byte	BA* <sup>3</sup>	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IL</sub>	04h
	Word					X	0004h
Device Code	MBM29DL164TE	BA* <sup>3</sup>	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	33h
						Word	X
	MBM29DL164BE	BA* <sup>3</sup>	V <sub>IL</sub>	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	35h
						Word	X
Sector Group Protection		Sector Group Addresses	V <sub>IL</sub>	V <sub>IH</sub>	V <sub>IL</sub>	V <sub>IL</sub>	01h* <sup>2</sup>
						Word	X

\*1 : A<sub>-1</sub> is for Byte mode. At Byte mode, DQ<sub>14</sub> to DQ<sub>8</sub> are High-Z and DQ<sub>15</sub> is A<sub>-1</sub>, the lowest address.

\*2 : Outputs 01h at protected sector group addresses and outputs 00h at unprotected sector group addresses.

\*3 : When V<sub>ID</sub> is applied to A<sub>9</sub>, both Bank 1 and Bank 2 are put into Autoselect mode, which makes simultaneous operation unable to be executed. Consequently, specifying the bank address is not required. However, the bank address needs to be indicated when Autoselect mode is read out at command mode, because then it enables to activate simultaneous operation.

**Expanded Autoselect Code Table**

Type		Code	DQ <sub>15</sub>	DQ <sub>14</sub>	DQ <sub>13</sub>	DQ <sub>12</sub>	DQ <sub>11</sub>	DQ <sub>10</sub>	DQ <sub>9</sub>	DQ <sub>8</sub>	DQ <sub>7</sub>	DQ <sub>6</sub>	DQ <sub>5</sub>	DQ <sub>4</sub>	DQ <sub>3</sub>	DQ <sub>2</sub>	DQ <sub>1</sub>	DQ <sub>0</sub>
Manufacturer's Code	(B)*	04h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	0	0	0	1	0	0
	(W)	0004h	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Device Code	MBM29DL164TE	(B)*	33h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	1	1	0	0	1	1
		(W)	2233h	0	0	1	0	0	0	1	0	0	0	1	1	0	0	1
	MBM29DL164BE	(B)*	35h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	1	1	0	1	0	1
		(W)	2235h	0	0	1	0	0	0	1	0	0	0	1	1	0	1	0
Sector Group Protection		(B)*	01h	A <sub>-1</sub>	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	HI-Z	0	0	0	0	0	0	0	1
		(W)	0001h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

(B) : Byte mode

(W) : Word mode

HI-Z : High-Z

\* : At Byte mode, DQ<sub>14</sub> to DQ<sub>8</sub> are High-Z and DQ<sub>15</sub> is A<sub>-1</sub>, the lowest address.

## ■ FLEXIBLE SECTOR-ERASE ARCHITECTURE

Sector Address Table (MBM29DL161TE)

Bank	Sector	Sector Address							Sector Size (Kbytes/ Kwords)	Address Range (×8)	Address Range (×16)	
		Bank Address					A <sub>14</sub>	A <sub>13</sub>				A <sub>12</sub>
		A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>						
Bank 2	SA0	0	0	0	0	0	X	X	X	64/32	000000h to 00FFFFh	000000h to 007FFFh
	SA1	0	0	0	0	1	X	X	X	64/32	010000h to 01FFFFh	008000h to 00FFFFh
	SA2	0	0	0	1	0	X	X	X	64/32	020000h to 02FFFFh	010000h to 017FFFh
	SA3	0	0	0	1	1	X	X	X	64/32	030000h to 03FFFFh	018000h to 01FFFFh
	SA4	0	0	1	0	0	X	X	X	64/32	040000h to 04FFFFh	020000h to 027FFFh
	SA5	0	0	1	0	1	X	X	X	64/32	050000h to 05FFFFh	028000h to 02FFFFh
	SA6	0	0	1	1	0	X	X	X	64/32	060000h to 06FFFFh	030000h to 037FFFh
	SA7	0	0	1	1	1	X	X	X	64/32	070000h to 07FFFFh	038000h to 03FFFFh
	SA8	0	1	0	0	0	X	X	X	64/32	080000h to 08FFFFh	040000h to 047FFFh
	SA9	0	1	0	0	1	X	X	X	64/32	090000h to 09FFFFh	048000h to 04FFFFh
	SA10	0	1	0	1	0	X	X	X	64/32	0A0000h to 0AFFFFh	050000h to 057FFFh
	SA11	0	1	0	1	1	X	X	X	64/32	0B0000h to 0BFFFFh	058000h to 05FFFFh
	SA12	0	1	1	0	0	X	X	X	64/32	0C0000h to 0CFFFFh	060000h to 067FFFh
	SA13	0	1	1	0	1	X	X	X	64/32	0D0000h to 0DFFFFh	068000h to 06FFFFh
	SA14	0	1	1	1	0	X	X	X	64/32	0E0000h to 0EFFFFh	070000h to 077FFFh
	SA15	0	1	1	1	1	X	X	X	64/32	0F0000h to 0FFFFFFh	078000h to 07FFFFh
	SA16	1	0	0	0	0	X	X	X	64/32	100000h to 10FFFFh	080000h to 087FFFh
	SA17	1	0	0	0	1	X	X	X	64/32	110000h to 11FFFFh	088000h to 08FFFFh
	SA18	1	0	0	1	0	X	X	X	64/32	120000h to 12FFFFh	090000h to 097FFFh
	SA19	1	0	0	1	1	X	X	X	64/32	130000h to 13FFFFh	098000h to 09FFFFh
	SA20	1	0	1	0	0	X	X	X	64/32	140000h to 14FFFFh	0A0000h to 0A7FFFh
	SA21	1	0	1	0	1	X	X	X	64/32	150000h to 15FFFFh	0A8000h to 0AFFFFh
	SA22	1	0	1	1	0	X	X	X	64/32	160000h to 16FFFFh	0B0000h to 0B7FFFh
	SA23	1	0	1	1	1	X	X	X	64/32	170000h to 17FFFFh	0B8000h to 0BFFFFh
	SA24	1	1	0	0	0	X	X	X	64/32	180000h to 18FFFFh	0C0000h to 0C7FFFh
	SA25	1	1	0	0	1	X	X	X	64/32	190000h to 19FFFFh	0C8000h to 0CFFFFh
	SA26	1	1	0	1	0	X	X	X	64/32	1A0000h to 1AFFFFh	0D0000h to 0D7FFFh
	SA27	1	1	0	1	1	X	X	X	64/32	1B0000h to 1BFFFFh	0D8000h to 0DFFFFh
	SA28	1	1	1	0	0	X	X	X	64/32	1C0000h to 1CFFFFh	0E0000h to 0E7FFFh
	SA29	1	1	1	0	1	X	X	X	64/32	1D0000h to 1DFFFFh	0E8000h to 0EFFFFh
SA30	1	1	1	1	0	X	X	X	64/32	1E0000h to 1EFFFFh	0F0000h to 0F7FFFh	
Bank 1	SA31	1	1	1	1	1	0	0	0	8/4	1F0000h to 1F1FFFh	0F8000h to 0F8FFFh
	SA32	1	1	1	1	1	0	0	1	8/4	1F2000h to 1F3FFFh	0F9000h to 0F9FFFh
	SA33	1	1	1	1	1	0	1	0	8/4	1F4000h to 1F5FFFh	0FA000h to 0FAFFFh
	SA34	1	1	1	1	1	0	1	1	8/4	1F6000h to 1F7FFFh	0FB000h to 0FBFFFh
	SA35	1	1	1	1	1	1	0	0	8/4	1F8000h to 1F9FFFh	0FC000h to 0FCFFFh
	SA36	1	1	1	1	1	1	0	1	8/4	1FA000h to 1FBFFFh	0FD000h to 0FDFFFh
	SA37	1	1	1	1	1	1	1	0	8/4	1FC000h to 1FDFFFh	0FE000h to 0FEFFFh
	SA38	1	1	1	1	1	1	1	1	8/4	1FE000h to 1FFFFFFh	0FF000h to 0FFFFFFh

Note: The address range is A<sub>19</sub>: A<sub>-1</sub> if in byte mode ( $\overline{\text{BYTE}} = V_{IL}$ ).  
 The address range is A<sub>19</sub>: A<sub>0</sub> if in word mode ( $\text{BYTE} = V_{IH}$ )

# MBM29DL16XTE/BE70/90

Sector Address Table (MBM29DL161BE)

Bank	Sector	Sector Address							Sector Size (Kbytes/ Kwords)	Address Range (×8)	Address Range (×16)	
		Bank Address					A <sub>14</sub>	A <sub>13</sub>				A <sub>12</sub>
		A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>						
Bank 2	SA38	1	1	1	1	1	X	X	X	64/32	1F0000h to 1FFFFFFh	0F8000h to 0FFFFFFh
	SA37	1	1	1	1	0	X	X	X	64/32	1E0000h to 1FFFFFFh	0F0000h to 0F7FFFh
	SA36	1	1	1	0	1	X	X	X	64/32	1D0000h to 1DFFFFh	0E8000h to 0EFFFFh
	SA35	1	1	1	0	0	X	X	X	64/32	1C0000h to 1CFFFFh	0E0000h to 0E7FFFh
	SA34	1	1	0	1	1	X	X	X	64/32	1B0000h to 1BFFFFh	0D8000h to 0DFFFFh
	SA33	1	1	0	1	0	X	X	X	64/32	1A0000h to 1AFFFFh	0D0000h to 0D7FFFh
	SA32	1	1	0	0	1	X	X	X	64/32	190000h to 19FFFFh	0C8000h to 0CFFFFh
	SA31	1	1	0	0	0	X	X	X	64/32	180000h to 18FFFFh	0C0000h to 0C7FFFh
	SA30	1	0	1	1	1	X	X	X	64/32	170000h to 17FFFFh	0B8000h to 0BFFFFh
	SA29	1	0	1	1	0	X	X	X	64/32	160000h to 16FFFFh	0B0000h to 0B7FFFh
	SA28	1	0	1	0	1	X	X	X	64/32	150000h to 15FFFFh	0A8000h to 0AFFFFh
	SA27	1	0	1	0	0	X	X	X	64/32	140000h to 14FFFFh	0A0000h to 0A7FFFh
	SA26	1	0	0	1	1	X	X	X	64/32	130000h to 13FFFFh	098000h to 09FFFFh
	SA25	1	0	0	1	0	X	X	X	64/32	120000h to 12FFFFh	090000h to 097FFFh
	SA24	1	0	0	0	X	X	X	X	64/32	110000h to 11FFFFh	088000h to 08FFFFh
	SA23	1	0	0	0	0	X	X	X	64/32	100000h to 10FFFFh	080000h to 087FFFh
	SA22	0	1	1	1	1	X	X	X	64/32	0F0000h to 0FFFFFFh	078000h to 07FFFFh
	SA21	0	1	1	1	0	X	X	X	64/32	0E0000h to 0EFFFFh	070000h to 077FFFh
	SA20	0	1	1	0	1	X	X	X	64/32	0D0000h to 0DFFFFh	068000h to 06FFFFh
	SA19	0	1	1	0	0	X	X	X	64/32	0C0000h to 0CFFFFh	060000h to 067FFFh
	SA18	0	1	0	1	1	X	X	X	64/32	0B0000h to 0BFFFFh	058000h to 05FFFFh
	SA17	0	1	0	1	0	X	X	X	64/32	0A0000h to 0AFFFFh	050000h to 057FFFh
	SA16	0	1	0	0	1	X	X	X	64/32	090000h to 09FFFFh	048000h to 04FFFFh
	SA15	0	1	0	0	0	X	X	X	64/32	080000h to 08FFFFh	040000h to 047FFFh
SA14	0	0	1	1	1	X	X	X	64/32	070000h to 07FFFFh	038000h to 03FFFFh	
SA13	0	0	1	1	0	X	X	X	64/32	060000h to 06FFFFh	030000h to 037FFFh	
SA12	0	0	1	0	1	X	X	X	64/32	050000h to 05FFFFh	028000h to 02FFFFh	
SA11	0	0	1	0	0	X	X	X	64/32	040000h to 04FFFFh	020000h to 027FFFh	
SA10	0	0	0	1	1	X	X	X	64/32	030000h to 03FFFFh	018000h to 01FFFFh	
SA9	0	0	0	1	0	X	X	X	64/32	020000h to 02FFFFh	010000h to 017FFFh	
SA8	0	0	0	0	1	X	X	X	64/32	010000h to 01FFFFh	008000h to 00FFFFh	
Bank 1	SA7	0	0	0	0	0	1	1	1	8/4	00E000h to 00FFFFh	007000h to 007FFFh
	SA6	0	0	0	0	0	1	1	0	8/4	00C000h to 00DFFFh	006000h to 006FFFh
	SA5	0	0	0	0	0	1	0	1	8/4	00A000h to 00BFFFh	005000h to 005FFFh
	SA4	0	0	0	0	0	1	0	0	8/4	008000h to 009FFFh	004000h to 004FFFh
	SA3	0	0	0	0	0	0	1	1	8/4	006000h to 007FFFh	003000h to 003FFFh
	SA2	0	0	0	0	0	0	1	0	8/4	004000h to 005FFFh	002000h to 002FFFh
	SA1	0	0	0	0	0	0	0	1	8/4	002000h to 003FFFh	001000h to 001FFFh
	SA0	0	0	0	0	0	0	0	0	8/4	000000h to 001FFFh	000000h to 000FFFh

Note: The address range is A<sub>19</sub>: A<sub>-1</sub> if in byte mode ( $\overline{\text{BYTE}} = V_{IL}$ ).  
 The address range is A<sub>19</sub>: A<sub>0</sub> if in word mode ( $\overline{\text{BYTE}} = V_{IH}$ ).



# MBM29DL16XTE/BE70/90

Sector Address Table (MBM29DL162TE)

Bank	Sector	Sector Address							Sector Size (Kbytes/ Kwords)	Address Range (×8)	Address Range (×16)	
		Bank Address					A <sub>14</sub>	A <sub>13</sub>				A <sub>12</sub>
		A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>						
Bank 2	SA0	0	0	0	0	0	X	X	X	64/32	000000h to 00FFFFh	000000h to 007FFFh
	SA1	0	0	0	0	1	X	X	X	64/32	010000h to 01FFFFh	008000h to 00FFFFh
	SA2	0	0	0	1	0	X	X	X	64/32	020000h to 02FFFFh	010000h to 017FFFh
	SA3	0	0	0	1	1	X	X	X	64/32	030000h to 03FFFFh	018000h to 01FFFFh
	SA4	0	0	1	0	0	X	X	X	64/32	040000h to 04FFFFh	020000h to 027FFFh
	SA5	0	0	1	0	1	X	X	X	64/32	050000h to 05FFFFh	028000h to 02FFFFh
	SA6	0	0	1	1	0	X	X	X	64/32	060000h to 06FFFFh	030000h to 037FFFh
	SA7	0	0	1	1	1	X	X	X	64/32	070000h to 07FFFFh	038000h to 03FFFFh
	SA8	0	1	0	0	0	X	X	X	64/32	080000h to 08FFFFh	040000h to 047FFFh
	SA9	0	1	0	0	1	X	X	X	64/32	090000h to 09FFFFh	048000h to 04FFFFh
	SA10	0	1	0	1	0	X	X	X	64/32	0A0000h to 0AFFFFh	050000h to 057FFFh
	SA11	0	1	0	1	1	X	X	X	64/32	0B0000h to 0BFFFFh	058000h to 05FFFFh
	SA12	0	1	1	0	0	X	X	X	64/32	0C0000h to 0CFFFFh	060000h to 067FFFh
	SA13	0	1	1	0	1	X	X	X	64/32	0D0000h to 0DFFFFh	068000h to 06FFFFh
	SA14	0	1	1	1	0	X	X	X	64/32	0E0000h to 0EFFFFh	070000h to 077FFFh
	SA15	0	1	1	1	1	X	X	X	64/32	0F0000h to 0FFFFFh	078000h to 07FFFFh
	SA16	1	0	0	0	0	X	X	X	64/32	100000h to 10FFFFh	080000h to 087FFFh
	SA17	1	0	0	0	1	X	X	X	64/32	110000h to 11FFFFh	088000h to 08FFFFh
	SA18	1	0	0	1	0	X	X	X	64/32	120000h to 12FFFFh	090000h to 097FFFh
	SA19	1	0	0	1	1	X	X	X	64/32	130000h to 13FFFFh	098000h to 09FFFFh
	SA20	1	0	1	0	0	X	X	X	64/32	140000h to 14FFFFh	0A0000h to 0A7FFFh
	SA21	1	0	1	0	1	X	X	X	64/32	150000h to 15FFFFh	0A8000h to 0AFFFFh
	SA22	1	0	1	1	0	X	X	X	64/32	160000h to 16FFFFh	0B0000h to 0B7FFFh
	SA23	1	0	1	1	1	X	X	X	64/32	170000h to 17FFFFh	0B8000h to 0BFFFFh
	SA24	1	1	0	0	0	X	X	X	64/32	180000h to 18FFFFh	0C0000h to 0C7FFFh
	SA25	1	1	0	0	1	X	X	X	64/32	190000h to 19FFFFh	0C8000h to 0CFFFFh
	SA26	1	1	0	1	0	X	X	X	64/32	1A0000h to 1AFFFFh	0D0000h to 0D7FFFh
SA27	1	1	0	1	1	X	X	X	64/32	1B0000h to 1BFFFFh	0D8000h to 0DFFFFh	
Bank 1	SA28	1	1	1	0	0	X	X	X	64/32	1C0000h to 1CFFFFh	0E0000h to 0E7FFFh
	SA29	1	1	1	0	1	X	X	X	64/32	1D0000h to 1DFFFFh	0E8000h to 0EFFFFh
	SA30	1	1	1	1	0	X	X	X	64/32	1E0000h to 1EFFFFh	0F0000h to 0F7FFFh
	SA31	1	1	1	1	1	0	0	0	8/4	1F0000h to 1F1FFFh	0F8000h to 0F8FFFh
	SA32	1	1	1	1	1	0	0	1	8/4	1F2000h to 1F3FFFh	0F9000h to 0F9FFFh
	SA33	1	1	1	1	1	0	1	0	8/4	1F4000h to 1F5FFFh	0FA000h to 0FAFFFh
	SA34	1	1	1	1	1	0	1	1	8/4	1F6000h to 1F7FFFh	0FB000h to 0FBFFFh
	SA35	1	1	1	1	1	1	0	0	8/4	1F8000h to 1F9FFFh	0FC000h to 0FCFFFh
	SA36	1	1	1	1	1	1	0	1	8/4	1FA000h to 1FBFFFh	0FD000h to 0FDFFFh
	SA37	1	1	1	1	1	1	1	0	8/4	1FC000h to 1FDFFFh	0FE000h to 0FEFFFh
	SA38	1	1	1	1	1	1	1	1	8/4	1FE000h to 1FFFFFh	0FF000h to 0FFFFFh

Note: The address range is A<sub>19</sub>: A<sub>-1</sub> if in byte mode ( $\overline{\text{BYTE}} = V_{IL}$ ).  
 The address range is A<sub>19</sub>: A<sub>0</sub> if in word mode ( $\overline{\text{BYTE}} = V_{IH}$ )

# MBM29DL16XTE/BE70/90

Sector Address Table (MBM29DL162BE)

Bank	Sector	Sector Address							Sector Size (Kbytes/ Kwords)	Address Range (×8)	Address Range (×16)	
		Bank Address					A <sub>14</sub>	A <sub>13</sub>				A <sub>12</sub>
		A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>						
Bank 2	SA38	1	1	1	1	1	X	X	X	64/32	1F0000h to 1FFFFFFh	0F8000h to 0FFFFFFh
	SA37	1	1	1	1	0	X	X	X	64/32	1E0000h to 1FFFFFFh	0F0000h to 0F7FFFh
	SA36	1	1	1	0	1	X	X	X	64/32	1D0000h to 1DFFFFh	0E8000h to 0EFFFFh
	SA35	1	1	1	0	0	X	X	X	64/32	1C0000h to 1CFFFFh	0E0000h to 0E7FFFh
	SA34	1	1	0	1	1	X	X	X	64/32	1B0000h to 1BFFFFh	0D8000h to 0DFFFFh
	SA33	1	1	0	1	0	X	X	X	64/32	1A0000h to 1AFFFFh	0D0000h to 0D7FFFh
	SA32	1	1	0	0	1	X	X	X	64/32	190000h to 19FFFFh	0C8000h to 0CFFFFh
	SA31	1	1	0	0	0	X	X	X	64/32	180000h to 18FFFFh	0C0000h to 0C7FFFh
	SA30	1	0	1	1	1	X	X	X	64/32	170000h to 17FFFFh	0B8000h to 0BFFFFh
	SA29	1	0	1	1	0	X	X	X	64/32	160000h to 16FFFFh	0B0000h to 0B7FFFh
	SA28	1	0	1	0	1	X	X	X	64/32	150000h to 15FFFFh	0A8000h to 0AFFFFh
	SA27	1	0	1	0	0	X	X	X	64/32	140000h to 14FFFFh	0A0000h to 0A7FFFh
	SA26	1	0	0	1	1	X	X	X	64/32	130000h to 13FFFFh	098000h to 09FFFFh
	SA25	1	0	0	1	0	X	X	X	64/32	120000h to 12FFFFh	090000h to 097FFFh
	SA24	1	0	0	0	X	X	X	X	64/32	110000h to 11FFFFh	088000h to 08FFFFh
	SA23	1	0	0	0	0	X	X	X	64/32	100000h to 10FFFFh	080000h to 087FFFh
	SA22	0	1	1	1	1	X	X	X	64/32	0F0000h to 0FFFFFFh	078000h to 07FFFFh
	SA21	0	1	1	1	0	X	X	X	64/32	0E0000h to 0EFFFFh	070000h to 077FFFh
	SA20	0	1	1	0	1	X	X	X	64/32	0D0000h to 0DFFFFh	068000h to 06FFFFh
	SA19	0	1	1	0	0	X	X	X	64/32	0C0000h to 0CFFFFh	060000h to 067FFFh
SA18	0	1	0	1	1	X	X	X	64/32	0B0000h to 0BFFFFh	058000h to 05FFFFh	
SA17	0	1	0	1	0	X	X	X	64/32	0A0000h to 0AFFFFh	050000h to 057FFFh	
SA16	0	1	0	0	1	X	X	X	64/32	090000h to 09FFFFh	048000h to 04FFFFh	
SA15	0	1	0	0	0	X	X	X	64/32	080000h to 08FFFFh	040000h to 047FFFh	
SA14	0	0	1	1	1	X	X	X	64/32	070000h to 07FFFFh	038000h to 03FFFFh	
SA13	0	0	1	1	0	X	X	X	64/32	060000h to 06FFFFh	030000h to 037FFFh	
SA12	0	0	1	0	1	X	X	X	64/32	050000h to 05FFFFh	028000h to 02FFFFh	
SA11	0	0	1	0	0	X	X	X	64/32	040000h to 04FFFFh	020000h to 027FFFh	
Bank 1	SA10	0	0	0	1	1	X	X	X	64/32	030000h to 03FFFFh	018000h to 01FFFFh
	SA9	0	0	0	1	0	X	X	X	64/32	020000h to 02FFFFh	010000h to 017FFFh
	SA8	0	0	0	0	1	X	X	X	64/32	010000h to 01FFFFh	008000h to 00FFFFh
	SA7	0	0	0	0	0	1	1	1	8/4	00E000h to 00FFFFh	007000h to 007FFFh
	SA6	0	0	0	0	0	1	1	0	8/4	00C000h to 00DFFFh	006000h to 006FFFh
	SA5	0	0	0	0	0	1	0	1	8/4	00A000h to 00BFFFh	005000h to 005FFFh
	SA4	0	0	0	0	0	1	0	0	8/4	008000h to 009FFFh	004000h to 004FFFh
	SA3	0	0	0	0	0	0	1	1	8/4	006000h to 007FFFh	003000h to 003FFFh
	SA2	0	0	0	0	0	0	1	0	8/4	004000h to 005FFFh	002000h to 002FFFh
	SA1	0	0	0	0	0	0	0	1	8/4	002000h to 003FFFh	001000h to 001FFFh
	SA0	0	0	0	0	0	0	0	0	8/4	000000h to 001FFFh	000000h to 000FFFh

Note: The address range is A<sub>19</sub>: A<sub>-1</sub> if in byte mode ( $\overline{\text{BYTE}} = V_{IL}$ ).  
 The address range is A<sub>19</sub>: A<sub>0</sub> if in word mode ( $\overline{\text{BYTE}} = V_{IH}$ ).

# MBM29DL16XTE/BE70/90

Sector Address Table (MBM29DL163TE)

Bank	Sector	Sector Address							Sector Size (Kbytes/ Kwords)	Address Range (×8)	Address Range (×16)	
		Bank Address					A <sub>14</sub>	A <sub>13</sub>				A <sub>12</sub>
		A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>						
Bank 2	SA0	0	0	0	0	0	X	X	X	64/32	000000h to 00FFFFh	000000h to 007FFFh
	SA1	0	0	0	0	1	X	X	X	64/32	010000h to 01FFFFh	008000h to 00FFFFh
	SA2	0	0	0	1	0	X	X	X	64/32	020000h to 02FFFFh	010000h to 017FFFh
	SA3	0	0	0	1	1	X	X	X	64/32	030000h to 03FFFFh	018000h to 01FFFFh
	SA4	0	0	1	0	0	X	X	X	64/32	040000h to 04FFFFh	020000h to 027FFFh
	SA5	0	0	1	0	1	X	X	X	64/32	050000h to 05FFFFh	028000h to 02FFFFh
	SA6	0	0	1	1	0	X	X	X	64/32	060000h to 06FFFFh	030000h to 037FFFh
	SA7	0	0	1	1	1	X	X	X	64/32	070000h to 07FFFFh	038000h to 03FFFFh
	SA8	0	1	0	0	0	X	X	X	64/32	080000h to 08FFFFh	040000h to 047FFFh
	SA9	0	1	0	0	1	X	X	X	64/32	090000h to 09FFFFh	048000h to 04FFFFh
	SA10	0	1	0	1	0	X	X	X	64/32	0A0000h to 0AFFFFh	050000h to 057FFFh
	SA11	0	1	0	1	1	X	X	X	64/32	0B0000h to 0BFFFFh	058000h to 05FFFFh
	SA12	0	1	1	0	0	X	X	X	64/32	0C0000h to 0CFFFFh	060000h to 067FFFh
	SA13	0	1	1	0	1	X	X	X	64/32	0D0000h to 0DFFFFh	068000h to 06FFFFh
	SA14	0	1	1	1	0	X	X	X	64/32	0E0000h to 0EFFFFh	070000h to 077FFFh
	SA15	0	1	1	1	1	X	X	X	64/32	0F0000h to 0FFFFFh	078000h to 07FFFFh
	SA16	1	0	0	0	0	X	X	X	64/32	100000h to 10FFFFh	080000h to 087FFFh
	SA17	1	0	0	0	1	X	X	X	64/32	110000h to 11FFFFh	088000h to 08FFFFh
	SA18	1	0	0	1	0	X	X	X	64/32	120000h to 12FFFFh	090000h to 097FFFh
	SA19	1	0	0	1	1	X	X	X	64/32	130000h to 13FFFFh	098000h to 09FFFFh
	SA20	1	0	1	0	0	X	X	X	64/32	140000h to 14FFFFh	0A0000h to 0A7FFFh
	SA21	1	0	1	0	1	X	X	X	64/32	150000h to 15FFFFh	0A8000h to 0AFFFFh
	SA22	1	0	1	1	0	X	X	X	64/32	160000h to 16FFFFh	0B0000h to 0B7FFFh
SA23	1	0	1	1	1	X	X	X	64/32	170000h to 17FFFFh	0B8000h to 0BFFFFh	
Bank 1	SA24	1	1	0	0	0	X	X	X	64/32	180000h to 18FFFFh	0C0000h to 0C7FFFh
	SA25	1	1	0	0	1	X	X	X	64/32	190000h to 19FFFFh	0C8000h to 0CFFFFh
	SA26	1	1	0	1	0	X	X	X	64/32	1A0000h to 1AFFFFh	0D0000h to 0D7FFFh
	SA27	1	1	0	1	1	X	X	X	64/32	1B0000h to 1BFFFFh	0D8000h to 0DFFFFh
	SA28	1	1	1	0	0	X	X	X	64/32	1C0000h to 1CFFFFh	0E0000h to 0E7FFFh
	SA29	1	1	1	0	1	X	X	X	64/32	1D0000h to 1DFFFFh	0E8000h to 0EFFFFh
	SA30	1	1	1	1	0	X	X	X	64/32	1E0000h to 1EFFFFh	0F0000h to 0F7FFFh
	SA31	1	1	1	1	1	0	0	0	8/4	1F0000h to 1F1FFFh	0F8000h to 0F8FFFh
	SA32	1	1	1	1	1	0	0	1	8/4	1F2000h to 1F3FFFh	0F9000h to 0F9FFFh
	SA33	1	1	1	1	1	0	1	0	8/4	1F4000h to 1F5FFFh	0FA000h to 0FAFFFh
	SA34	1	1	1	1	1	0	1	1	8/4	1F6000h to 1F7FFFh	0FB000h to 0FBFFFh
	SA35	1	1	1	1	1	1	0	0	8/4	1F8000h to 1F9FFFh	0FC000h to 0FCFFFh
	SA36	1	1	1	1	1	1	0	1	8/4	1FA000h to 1FBFFFh	0FD000h to 0FDFFFh
	SA37	1	1	1	1	1	1	1	0	8/4	1FC000h to 1FDFFFh	0FE000h to 0FEFFFh
	SA38	1	1	1	1	1	1	1	1	8/4	1FE000h to 1FFFFFh	0FF000h to 0FFFFFh

Note: The address range is A<sub>19</sub>: A<sub>-1</sub> if in byte mode ( $\overline{\text{BYTE}} = V_{IL}$ ).  
 The address range is A<sub>19</sub>: A<sub>0</sub> if in word mode ( $\overline{\text{BYTE}} = V_{IH}$ )

# MBM29DL16XTE/BE70/90

Sector Address Table (MBM29DL163BE)

Bank	Sector	Sector Address							Sector Size (Kbytes/ Kwords)	Address Range (×8)	Address Range (×16)	
		Bank Address					A <sub>14</sub>	A <sub>13</sub>				A <sub>12</sub>
		A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>						
Bank 2	SA38	1	1	1	1	1	X	X	X	64/32	1F0000h to 1FFFFFFh	0F8000h to 0FFFFFFh
	SA37	1	1	1	1	0	X	X	X	64/32	1E0000h to 1FFFFFFh	0F0000h to 0F7FFFh
	SA36	1	1	1	0	1	X	X	X	64/32	1D0000h to 1DFFFFh	0E8000h to 0EFFFFh
	SA35	1	1	1	0	0	X	X	X	64/32	1C0000h to 1CFFFFh	0E0000h to 0E7FFFh
	SA34	1	1	0	1	1	X	X	X	64/32	1B0000h to 1BFFFFh	0D8000h to 0DFFFFh
	SA33	1	1	0	1	0	X	X	X	64/32	1A0000h to 1AFFFFh	0D0000h to 0D7FFFh
	SA32	1	1	0	0	1	X	X	X	64/32	190000h to 19FFFFh	0C8000h to 0CFFFFh
	SA31	1	1	0	0	0	X	X	X	64/32	180000h to 18FFFFh	0C0000h to 0C7FFFh
	SA30	1	0	1	1	1	X	X	X	64/32	170000h to 17FFFFh	0B8000h to 0BFFFFh
	SA29	1	0	1	1	0	X	X	X	64/32	160000h to 16FFFFh	0B0000h to 0B7FFFh
	SA28	1	0	1	0	1	X	X	X	64/32	150000h to 15FFFFh	0A8000h to 0AFFFFh
	SA27	1	0	1	0	0	X	X	X	64/32	140000h to 14FFFFh	0A0000h to 0A7FFFh
	SA26	1	0	0	1	1	X	X	X	64/32	130000h to 13FFFFh	098000h to 09FFFFh
	SA25	1	0	0	1	0	X	X	X	64/32	120000h to 12FFFFh	090000h to 097FFFh
	SA24	1	0	0	0	X	X	X	X	64/32	110000h to 11FFFFh	088000h to 08FFFFh
	SA23	1	0	0	0	0	X	X	X	64/32	100000h to 10FFFFh	080000h to 087FFFh
	SA22	0	1	1	1	1	X	X	X	64/32	0F0000h to 0FFFFFFh	078000h to 07FFFFh
	SA21	0	1	1	1	0	X	X	X	64/32	0E0000h to 0EFFFFh	070000h to 077FFFh
	SA20	0	1	1	0	1	X	X	X	64/32	0D0000h to 0DFFFFh	068000h to 06FFFFh
	SA19	0	1	1	0	0	X	X	X	64/32	0C0000h to 0CFFFFh	060000h to 067FFFh
SA18	0	1	0	1	1	X	X	X	64/32	0B0000h to 0BFFFFh	058000h to 05FFFFh	
SA17	0	1	0	1	0	X	X	X	64/32	0A0000h to 0AFFFFh	050000h to 057FFFh	
SA16	0	1	0	0	1	X	X	X	64/32	090000h to 09FFFFh	048000h to 04FFFFh	
SA15	0	1	0	0	0	X	X	X	64/32	080000h to 08FFFFh	040000h to 047FFFh	
Bank 1	SA14	0	0	1	1	1	X	X	X	64/32	070000h to 07FFFFh	038000h to 03FFFFh
	SA13	0	0	1	1	0	X	X	X	64/32	060000h to 06FFFFh	030000h to 037FFFh
	SA12	0	0	1	0	1	X	X	X	64/32	050000h to 05FFFFh	028000h to 02FFFFh
	SA11	0	0	1	0	0	X	X	X	64/32	040000h to 04FFFFh	020000h to 027FFFh
	SA10	0	0	0	1	1	X	X	X	64/32	030000h to 03FFFFh	018000h to 01FFFFh
	SA9	0	0	0	1	0	X	X	X	64/32	020000h to 02FFFFh	010000h to 017FFFh
	SA8	0	0	0	0	1	X	X	X	64/32	010000h to 01FFFFh	008000h to 00FFFFh
	SA7	0	0	0	0	0	1	1	1	8/4	00E000h to 00FFFFh	007000h to 007FFFh
	SA6	0	0	0	0	0	1	1	0	8/4	00C000h to 00DFFFh	006000h to 006FFFh
	SA5	0	0	0	0	0	1	0	1	8/4	00A000h to 00BFFFh	005000h to 005FFFh
	SA4	0	0	0	0	0	1	0	0	8/4	008000h to 009FFFh	004000h to 004FFFh
	SA3	0	0	0	0	0	0	1	1	8/4	006000h to 007FFFh	003000h to 003FFFh
	SA2	0	0	0	0	0	0	1	0	8/4	004000h to 005FFFh	002000h to 002FFFh
	SA1	0	0	0	0	0	0	0	1	8/4	002000h to 003FFFh	001000h to 001FFFh
SA0	0	0	0	0	0	0	0	0	8/4	000000h to 001FFFh	000000h to 000FFFh	

Note: The address range is A<sub>19</sub>: A<sub>-1</sub> if in byte mode ( $\overline{\text{BYTE}} = V_{IL}$ ).  
 The address range is A<sub>19</sub>: A<sub>0</sub> if in word mode ( $\text{BYTE} = V_{IH}$ ).

# MBM29DL16XTE/BE70/90

Sector Address Table (MBM29DL164TE)

Bank	Sector	Sector Address							Sector Size (Kbytes/ Kwords)	Address Range (×8)	Address Range (×16)	
		Bank Address					A <sub>14</sub>	A <sub>13</sub>				A <sub>12</sub>
		A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>						
Bank 2	SA0	0	0	0	0	0	X	X	X	64/32	000000h to 00FFFFh	000000h to 007FFFh
	SA1	0	0	0	0	1	X	X	X	64/32	010000h to 01FFFFh	008000h to 00FFFFh
	SA2	0	0	0	1	0	X	X	X	64/32	020000h to 02FFFFh	010000h to 017FFFh
	SA3	0	0	0	1	1	X	X	X	64/32	030000h to 03FFFFh	018000h to 01FFFFh
	SA4	0	0	1	0	0	X	X	X	64/32	040000h to 04FFFFh	020000h to 027FFFh
	SA5	0	0	1	0	1	X	X	X	64/32	050000h to 05FFFFh	028000h to 02FFFFh
	SA6	0	0	1	1	0	X	X	X	64/32	060000h to 06FFFFh	030000h to 037FFFh
	SA7	0	0	1	1	1	X	X	X	64/32	070000h to 07FFFFh	038000h to 03FFFFh
	SA8	0	1	0	0	0	X	X	X	64/32	080000h to 08FFFFh	040000h to 047FFFh
	SA9	0	1	0	0	1	X	X	X	64/32	090000h to 09FFFFh	048000h to 04FFFFh
	SA10	0	1	0	1	0	X	X	X	64/32	0A0000h to 0AFFFFh	050000h to 057FFFh
	SA11	0	1	0	1	1	X	X	X	64/32	0B0000h to 0BFFFFh	058000h to 05FFFFh
	SA12	0	1	1	0	0	X	X	X	64/32	0C0000h to 0CFFFFh	060000h to 067FFFh
	SA13	0	1	1	0	1	X	X	X	64/32	0D0000h to 0DFFFFh	068000h to 06FFFFh
	SA14	0	1	1	1	0	X	X	X	64/32	0E0000h to 0EFFFFh	070000h to 077FFFh
SA15	0	1	1	1	1	X	X	X	64/32	0F0000h to 0FFFFFh	078000h to 07FFFFh	
Bank 1	SA16	1	0	0	0	0	X	X	X	64/32	100000h to 10FFFFh	080000h to 087FFFh
	SA17	1	0	0	0	1	X	X	X	64/32	110000h to 11FFFFh	088000h to 08FFFFh
	SA18	1	0	0	1	0	X	X	X	64/32	120000h to 12FFFFh	090000h to 097FFFh
	SA19	1	0	0	1	1	X	X	X	64/32	130000h to 13FFFFh	098000h to 09FFFFh
	SA20	1	0	1	0	0	X	X	X	64/32	140000h to 14FFFFh	0A0000h to 0A7FFFh
	SA21	1	0	1	0	1	X	X	X	64/32	150000h to 15FFFFh	0A8000h to 0AFFFFh
	SA22	1	0	1	1	0	X	X	X	64/32	160000h to 16FFFFh	0B0000h to 0B7FFFh
	SA23	1	0	1	1	1	X	X	X	64/32	170000h to 17FFFFh	0B8000h to 0BFFFFh
	SA24	1	1	0	0	0	X	X	X	64/32	180000h to 18FFFFh	0C0000h to 0C7FFFh
	SA25	1	1	0	0	1	X	X	X	64/32	190000h to 19FFFFh	0C8000h to 0CFFFFh
	SA26	1	1	0	1	0	X	X	X	64/32	1A0000h to 1AFFFFh	0D0000h to 0D7FFFh
	SA27	1	1	0	1	1	X	X	X	64/32	1B0000h to 1BFFFFh	0D8000h to 0DFFFFh
	SA28	1	1	1	0	0	X	X	X	64/32	1C0000h to 1CFFFFh	0E0000h to 0E7FFFh
	SA29	1	1	1	0	1	X	X	X	64/32	1D0000h to 1DFFFFh	0E8000h to 0EFFFFh
	SA30	1	1	1	1	0	X	X	X	64/32	1E0000h to 1EFFFFh	0F0000h to 0F7FFFh
	SA31	1	1	1	1	1	0	0	0	8/4	1F0000h to 1F1FFFh	0F8000h to 0F8FFFh
	SA32	1	1	1	1	1	0	0	1	8/4	1F2000h to 1F3FFFh	0F9000h to 0F9FFFh
	SA33	1	1	1	1	1	0	1	0	8/4	1F4000h to 1F5FFFh	0FA000h to 0FAFFFh
	SA34	1	1	1	1	1	0	1	1	8/4	1F6000h to 1F7FFFh	0FB000h to 0FBFFFh
	SA35	1	1	1	1	1	1	0	0	8/4	1F8000h to 1F9FFFh	0FC000h to 0FCFFFh
	SA36	1	1	1	1	1	1	0	1	8/4	1FA000h to 1FBFFFh	0FD000h to 0FDFFFh
	SA37	1	1	1	1	1	1	1	0	8/4	1FC000h to 1FDFFFh	0FE000h to 0FEFFFh
	SA38	1	1	1	1	1	1	1	1	8/4	1FE000h to 1FFFFFh	0FF000h to 0FFFFFh

Note: The address range is A<sub>19</sub>: A<sub>-1</sub> if in byte mode ( $\overline{\text{BYTE}} = V_{IL}$ ).  
 The address range is A<sub>19</sub>: A<sub>0</sub> if in word mode ( $\overline{\text{BYTE}} = V_{IH}$ )

# MBM29DL16XTE/BE70/90

Sector Address Table (MBM29DL164BE)

Bank	Sector	Sector Address							Sector Size (Kbytes/ Kwords)	Address Range (×8)	Address Range (×16)	
		Bank Address					A <sub>14</sub>	A <sub>13</sub>				A <sub>12</sub>
		A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>						
Bank 2	SA38	1	1	1	1	1	X	X	X	64/32	1F0000h to 1FFFFFFh	0F8000h to 0FFFFFFh
	SA37	1	1	1	1	0	X	X	X	64/32	1E0000h to 1FFFFFFh	0F0000h to 0F7FFFh
	SA36	1	1	1	0	1	X	X	X	64/32	1D0000h to 1DFFFFh	0E8000h to 0EFFFFh
	SA35	1	1	1	0	0	X	X	X	64/32	1C0000h to 1CFFFFh	0E0000h to 0E7FFFh
	SA34	1	1	0	1	1	X	X	X	64/32	1B0000h to 1BFFFFh	0D8000h to 0DFFFFh
	SA33	1	1	0	1	0	X	X	X	64/32	1A0000h to 1AFFFFh	0D0000h to 0D7FFFh
	SA32	1	1	0	0	1	X	X	X	64/32	190000h to 19FFFFh	0C8000h to 0CFFFFh
	SA31	1	1	0	0	0	X	X	X	64/32	180000h to 18FFFFh	0C0000h to 0C7FFFh
	SA30	1	0	1	1	1	X	X	X	64/32	170000h to 17FFFFh	0B8000h to 0BFFFFh
	SA29	1	0	1	1	0	X	X	X	64/32	160000h to 16FFFFh	0B0000h to 0B7FFFh
	SA28	1	0	1	0	1	X	X	X	64/32	150000h to 15FFFFh	0A8000h to 0AFFFFh
	SA27	1	0	1	0	0	X	X	X	64/32	140000h to 14FFFFh	0A0000h to 0A7FFFh
	SA26	1	0	0	1	1	X	X	X	64/32	130000h to 13FFFFh	098000h to 09FFFFh
	SA25	1	0	0	1	0	X	X	X	64/32	120000h to 12FFFFh	090000h to 097FFFh
SA24	1	0	0	0	X	X	X	X	64/32	110000h to 11FFFFh	088000h to 08FFFFh	
SA23	1	0	0	0	0	X	X	X	64/32	100000h to 10FFFFh	080000h to 087FFFh	
Bank 1	SA22	0	1	1	1	1	X	X	X	64/32	0F0000h to 0FFFFFFh	078000h to 07FFFFh
	SA21	0	1	1	1	0	X	X	X	64/32	0E0000h to 0EFFFFh	070000h to 077FFFh
	SA20	0	1	1	0	1	X	X	X	64/32	0D0000h to 0DFFFFh	068000h to 06FFFFh
	SA19	0	1	1	0	0	X	X	X	64/32	0C0000h to 0CFFFFh	060000h to 067FFFh
	SA18	0	1	0	1	1	X	X	X	64/32	0B0000h to 0BFFFFh	058000h to 05FFFFh
	SA17	0	1	0	1	0	X	X	X	64/32	0A0000h to 0AFFFFh	050000h to 057FFFh
	SA16	0	1	0	0	1	X	X	X	64/32	090000h to 09FFFFh	048000h to 04FFFFh
	SA15	0	1	0	0	0	X	X	X	64/32	080000h to 08FFFFh	040000h to 047FFFh
	SA14	0	0	1	1	1	X	X	X	64/32	070000h to 07FFFFh	038000h to 03FFFFh
	SA13	0	0	1	1	0	X	X	X	64/32	060000h to 06FFFFh	030000h to 037FFFh
	SA12	0	0	1	0	1	X	X	X	64/32	050000h to 05FFFFh	028000h to 02FFFFh
	SA11	0	0	1	0	0	X	X	X	64/32	040000h to 04FFFFh	020000h to 027FFFh
	SA10	0	0	0	1	1	X	X	X	64/32	030000h to 03FFFFh	018000h to 01FFFFh
	SA9	0	0	0	1	0	X	X	X	64/32	020000h to 02FFFFh	010000h to 017FFFh
	SA8	0	0	0	0	1	X	X	X	64/32	010000h to 01FFFFh	008000h to 00FFFFh
	SA7	0	0	0	0	0	1	1	1	8/4	00E000h to 00FFFFh	007000h to 007FFFh
	SA6	0	0	0	0	0	1	1	0	8/4	00C000h to 00DFFFh	006000h to 006FFFh
SA5	0	0	0	0	0	1	0	1	8/4	00A000h to 00BFFFh	005000h to 005FFFh	
SA4	0	0	0	0	0	1	0	0	8/4	008000h to 009FFFh	004000h to 004FFFh	
SA3	0	0	0	0	0	0	1	1	8/4	006000h to 007FFFh	003000h to 003FFFh	
SA2	0	0	0	0	0	0	1	0	8/4	004000h to 005FFFh	002000h to 002FFFh	
SA1	0	0	0	0	0	0	0	1	8/4	002000h to 003FFFh	001000h to 001FFFh	
SA0	0	0	0	0	0	0	0	0	8/4	000000h to 001FFFh	000000h to 000FFFh	

Note: The address range is A<sub>19</sub>: A<sub>-1</sub> if in byte mode ( $\overline{\text{BYTE}} = V_{IL}$ ).  
 The address range is A<sub>19</sub>: A<sub>0</sub> if in word mode ( $\overline{\text{BYTE}} = V_{IH}$ ).

# MBM29DL16XTE/BE70/90

**Sector Group Addresses Table (MBM29DL16XTE)  
(Top Boot Block)**

Sector Group	A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>	A <sub>14</sub>	A <sub>13</sub>	A <sub>12</sub>	Sectors
SGA0	0	0	0	0	0	X	X	X	SA0
SGA1	0	0	0	0	1	X	X	X	SA1 to SA3
	0	0	0	1	0	X	X	X	
	0	0	0	1	1	X	X	X	
SGA2	0	0	1	X	X	X	X	X	SA4 to SA7
SGA3	0	1	0	X	X	X	X	X	SA8 to SA11
SGA4	0	1	1	X	X	X	X	X	SA12 to SA15
SGA5	1	0	0	X	X	X	X	X	SA16 to SA19
SGA6	1	0	1	X	X	X	X	X	SA20 to SA23
SGA7	1	1	0	X	X	X	X	X	SA24 to SA27
SGA8	1	1	1	0	0	X	X	X	SA28 to SA30
	1	1	1	0	1	X	X	X	
	1	1	1	1	0	X	X	X	
SGA9	1	1	1	1	1	0	0	0	SA31
SGA10	1	1	1	1	1	0	0	1	SA32
SGA11	1	1	1	1	1	0	1	0	SA33
SGA12	1	1	1	1	1	0	1	1	SA34
SGA13	1	1	1	1	1	1	0	0	SA35
SGA14	1	1	1	1	1	1	0	1	SA36
SGA15	1	1	1	1	1	1	1	0	SA37
SGA16	1	1	1	1	1	1	1	1	SA38

# MBM29DL16XTE/BE70/90

**Sector Group Addresses Table (MBM29DL16XBE)  
(Bottom Boot Block)**

Sector Group	A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>	A <sub>14</sub>	A <sub>13</sub>	A <sub>12</sub>	Sectors
SGA0	0	0	0	0	0	0	0	0	SA0
SGA1	0	0	0	0	0	0	0	1	SA1
SGA2	0	0	0	0	0	0	1	0	SA2
SGA3	0	0	0	0	0	0	1	1	SA3
SGA4	0	0	0	0	0	1	0	0	SA4
SGA5	0	0	0	0	0	1	0	1	SA5
SGA6	0	0	0	0	0	1	1	0	SA6
SGA7	0	0	0	0	0	1	1	1	SA7
SGA8	0	0	0	0	1	X	X	X	SA8 to SA10
	0	0	0	1	0	X	X	X	
	0	0	0	1	1	X	X	X	
SGA9	0	0	1	X	X	X	X	X	SA11 to SA14
SGA10	0	1	0	X	X	X	X	X	SA15 to SA18
SGA11	0	1	1	X	X	X	X	X	SA19 to SA22
SGA12	1	0	0	X	X	X	X	X	SA23 to SA26
SGA13	1	0	1	X	X	X	X	X	SA27 to SA30
SGA14	1	1	0	X	X	X	X	X	SA31 to SA34
SGA15	1	1	1	0	0	X	X	X	SA35 to SA37
	1	1	1	0	1	X	X	X	
	1	1	1	1	0	X	X	X	
SGA16	1	1	1	1	1	X	X	X	SA38



# MBM29DL16XTE/BE70/90

Common Flash Memory Interface Code Table

Description	A <sub>6</sub> to A <sub>0</sub>	DQ <sub>15</sub> to DQ <sub>0</sub>
Query-unique ASCII string "QRY"	10h 11h 12h	0051h 0052h 0059h
Primary OEM Command Set 02h: AMD/FJ standard type	13h 14h	0002h 0000h
Address for Primary Extended Table	15h 16h	0040h 0000h
Alternate OEM Command Set (00h = not applicable)	17h 18h	0000h 0000h
Address for Alternate OEM Extended Table	19h 1Ah	0000h 0000h
V <sub>CC</sub> Min (write/erase) DQ <sub>7</sub> to DQ <sub>4</sub> : 1 V, DQ <sub>3</sub> to DQ <sub>0</sub> : 100 mV	1Bh	0027h
V <sub>CC</sub> Max (write/erase) DQ <sub>7</sub> to DQ <sub>4</sub> : 1 V, DQ <sub>3</sub> to DQ <sub>0</sub> : 100 mV	1Ch	0036h
V <sub>PP</sub> Min voltage	1Dh	0000h
V <sub>PP</sub> Max voltage	1Eh	0000h
Typical timeout per single byte/word write 2 <sup>N</sup> μs	1Fh	0004h
Typical timeout for Min size buffer write 2 <sup>N</sup> μs	20h	0000h
Typical timeout per individual sector erase 2 <sup>N</sup> ms	21h	000Ah
Typical timeout for full chip erase 2 <sup>N</sup> ms	22h	0000h
Max timeout for byte/word write 2 <sup>N</sup> times typical	23h	0005h
Max timeout for buffer write 2 <sup>N</sup> times typical	24h	0000h
Max timeout per individual sector erase 2 <sup>N</sup> times typical	25h	0004h
Max timeout for full chip erase 2 <sup>N</sup> times typical	26h	0000h
Device Size = 2 <sup>N</sup> byte	27h	0015h
Flash Device Interface description 02h : ×8/×16	28h 29h	0002h 0000h
Max. number of bytes in multi-byte write = 2 <sup>N</sup>	2Ah 2Bh	0000h 0000h
Number of Erase Block Regions within device	2Ch	0002h
Erase Block Region 1 Information bit 15 to bit 0 : y = number of sectors bit 31 to bit 16 : z = size (z×256 bytes)	2Dh 2Eh 2Fh 30h	0007h 0000h 0020h 0000h

Description	A <sub>6</sub> to A <sub>0</sub>	DQ <sub>15</sub> to DQ <sub>0</sub>
Erase Block Region 2 Information bit 15 to bit 0 : y = number of sectors bit 31 to bit 16 : z = size (z×256 bytes)	31h 32h 33h 34h	001Eh 0000h 0000h 0001h
Query-unique ASCII string "PRI"	40h 41h 42h	0050h 0052h 0049h
Major version number, ASCII	43h	0031h
Minor version number, ASCII	44h	0032h
Address Sensitive Unlock 00h = Required	45h	0000h
Erase Suspend 02h = To Read & Write	46h	0002h
Sector Protection 00h = Not Supported X = Number of sectors in per group	47h	0001h
Sector Temporary Unprotection 01h = Supported	48h	0001h
Sector Protection Algorithm	49h	0004h
Number of Sector for Bank 2 00h = Not Supported 1Fh = MBM29DL161TE 1Ch = MBM29DL162TE 18h = MBM29DL163TE 10h = MBM29DL164TE 1Fh = MBM29DL161BE 1Ch = MBM29DL162BE 18h = MBM29DL163BE 10h = MBM29DL164BE	4Ah	00XXh
Burst Mode Type 00h = Not Supported	4Bh	0000h
Page Mode Type 00h = Not Supported	4Ch	0000h
V <sub>ACC</sub> (Acceleration) Supply Minimum DQ <sub>7</sub> to DQ <sub>4</sub> : 1 V, DQ <sub>3</sub> to DQ <sub>0</sub> : 100 mV	4Dh	0085h
V <sub>ACC</sub> (Acceleration) Supply Maximum DQ <sub>7</sub> to DQ <sub>4</sub> : 1 V, DQ <sub>3</sub> to DQ <sub>0</sub> : 100 mV	4Eh	0095h
Boot Type 02h = MBM29DL16XBE 03h = MBM29DL16XTE	4Fh	00XXh
Program Suspend 01h = Supported	50h	0001h

# MBM29DL16XTE/BE70/90

## ■ FUNCTIONAL DESCRIPTION

### • Simultaneous Operation

MBM29DL16XTE/BE have feature, which is capability of reading data from one bank of memory while a program or erase operation is in progress in the other bank of memory (simultaneous operation), in addition to the conventional features (read, program, erase, erase-suspend read, and erase-suspend program). The bank selection can be selected by bank address ( $A_{19}$  to  $A_{15}$ ) with zero latency.

The MBM29DL161TE/BE have two banks which contain  
Bank 1 (8KB × 8 sectors) and Bank 2 (64KB × 31 sectors).

The MBM29DL162TE/BE have two banks which contain  
Bank 1 (8KB × 8 sectors, 64KB × 3 sectors) and Bank 2 (64KB × 28 sectors).

The MBM29DL163TE/BE have two banks which contain  
Bank 1 (8KB × 8 sectors, 64KB × 7 sectors) and Bank 2 (64KB × 24 sectors).

The MBM29DL164TE/BE have two banks which contain  
Bank 1 (8KB × 8 sectors, 64KB × 15 sectors) and Bank 2 (64KB × 16 sectors).

The simultaneous operation can not execute multi-function mode in the same bank. “Simultaneous Operation Table” shows combination to be possible for simultaneous operation. (Refer to “(8) Bank-to-bank Read/Write Timing Diagram” in ■TIMING DIAGRAM.)

Simultaneous Operation Table

Case	Bank 1 Status	Bank 2 Status
1	Read mode	Read mode
2	Read mode	Autoselect mode
3	Read mode	Program mode
4	Read mode	Erase mode *
5	Autoselect mode	Read mode
6	Program mode	Read mode
7	Erase mode *	Read mode

\*: By writing erase suspend command on the bank address of sector being erased, the erase operation becomes suspended so that it enables reading from or programming the remaining sectors.

### • Read Mode

The MBM29DL16XTE/BE have two control functions which must be satisfied in order to obtain data at the outputs.  $\overline{CE}$  is the power control and should be used for a device selection.  $\overline{OE}$  is the output control and should be used to gate data to the output pins if a device is selected.

Address access time ( $t_{ACC}$ ) is equal to the delay from stable addresses to valid output data. The chip enable access time ( $t_{CE}$ ) is the delay from stable addresses and stable  $\overline{CE}$  to valid data at the output pins. The output enable access time ( $t_{OE}$ ) is the delay from the falling edge of  $\overline{OE}$  to valid data at the output pins. (Assuming the addresses have been stable for at least  $t_{ACC-t_{OE}}$  time.) When reading out a data without changing addresses after power-up, it is necessary to input hardware reset or to change  $\overline{CE}$  pin from “H” to “L”.

## • Standby Mode

There are two ways to implement the standby mode on the MBM29DL16XTE/BE devices, one using both the  $\overline{CE}$  and  $\overline{RESET}$  pins; the other via the  $\overline{RESET}$  pin only.

When using both pins, a CMOS standby mode is achieved with  $\overline{CE}$  and  $\overline{RESET}$  inputs both held at  $V_{CC} \pm 0.3$  V. Under this condition the current consumed is less than 5  $\mu$ A max. During Embedded Algorithm operation,  $V_{CC}$  active current ( $I_{CC2}$ ) is required even  $\overline{CE} = "H"$ . The device can be read with standard access time ( $t_{CE}$ ) from either of these standby modes.

When using the  $\overline{RESET}$  pin only, a CMOS standby mode is achieved with  $\overline{RESET}$  input held at  $V_{SS} \pm 0.3$  V ( $\overline{CE} = "H"$  or  $"L"$ ). Under this condition the current is consumed is less than 5  $\mu$ A max. Once the  $\overline{RESET}$  pin is taken high, the device requires  $t_{RH}$  of wake up time before outputs are valid for read access.

In the standby mode the outputs are in the high impedance state, independent of the  $\overline{OE}$  input.

## • Automatic Sleep Mode

There is a function called automatic sleep mode to restrain power consumption during read-out of MBM29DL16XTE/BE data. This mode can be used effectively with an application requested low power consumption such as handy terminals.

To activate this mode, MBM29DL16XTE/BE automatically switch themselves to low power mode when MBM29DL16XTE/BE addresses remain stably during access time of 150 ns. It is not necessary to control  $\overline{CE}$ ,  $\overline{WE}$ , and  $\overline{OE}$  on the mode. Under the mode, the current consumed is typically 1  $\mu$ A (CMOS Level).

During simultaneous operation,  $V_{CC}$  active current ( $I_{CC2}$ ) is required.

Since the data are latched during this mode, the data are read-out continuously. If the addresses are changed, the mode is canceled automatically and MBM29DL16XTE/BE read-out the data for changed addresses.

## • Output Disable

With the  $\overline{OE}$  input at a logic high level ( $V_{IH}$ ), output from the devices are disabled. This will cause the output pins to be in a high impedance state.

## • Autoselect

The autoselect mode allows the reading out of a binary code from the devices and will identify its manufacturer and type. This mode is intended for use by programming equipment for the purpose of automatically matching the devices to be programmed with its corresponding programming algorithm. This mode is functional over the entire temperature range of the devices.

To activate this mode, the programming equipment must force  $V_{ID}$  (11.5 V to 12.5 V) on address pin  $A_9$ . Two identifier bytes may then be sequenced from the devices outputs by toggling address  $A_0$  from  $V_{IL}$  to  $V_{IH}$ . All addresses are DON'T CARES except  $A_6$ ,  $A_1$   $A_0$ , and  $(A_{-1})$ . (See "MBM29DL16XTE/BE User Bus Operations Tables ( $\overline{BYTE} = V_{IH}$  and  $\overline{BYTE} = V_{IL}$ )" in ■DEVICE BUS OPERATION.)

The manufacturer and device codes may also be read via the command register, for instances when the MBM29DL16XTE/BE are erased or programmed in a system without access to high voltage on the  $A_9$  pin. The command sequence is illustrated in "MBM29DL16XTE/BE Command Definitions Table" in ■DEVICE BUS OPERATION. (Refer to "Autoselect Command" in ■COMMAND DEFINITIONS.)

Word 0 ( $A_0 = V_{IL}$ ) represents the manufacturer's code (Fujitsu = 04h) and word 1 ( $A_0 = V_{IH}$ ) represents the device identifier code (MBM29DL161TE = 36h and MBM29DL161BE = 39h for  $\times 8$  mode; MBM29DL161TE = 2236h and MBM29DL161BE = 2239h for  $\times 16$  mode), (MBM29DL162TE = 2Dh and MBM29DL162BE = 2Eh for  $\times 8$  mode; MBM29DL162TE = 222Dh and MBM29DL162BE = 222Eh for  $\times 16$  mode), (MBM29DL163TE = 28h and MBM29DL163BE = 2Bh for  $\times 8$  mode; MBM29DL163TE = 2228h and MBM29DL163BE = 222Bh for  $\times 16$  mode),

# MBM29DL16XTE/BE70/90

(MBM29DL164TE = 33h and MBM29DL164BE = 35h for ×8 mode; MBM29DL164TE = 2233h and MBM29DL164BE = 2235h for ×16 mode). These two bytes/words are given in “MBM29DL16XTE/BE Sector Group Protection Verify Autoselect Codes Tables” and “Expanded Autoselect Code Tables” in ■DEVICE BUS OPERATION. All identifiers for manufactures and device will exhibit odd parity with DQ<sub>7</sub> defined as the parity bit. In order to read the proper device codes when executing the autoselect, A<sub>1</sub> must be V<sub>IL</sub>. (See “MBM29DL16XTE/BE Sector Group Protection Verify Autoselect Codes Tables” and “Expanded Autoselect Code Tables” in ■DEVICE BUS OPERATION.)

In case of applying V<sub>ID</sub> on A<sub>9</sub>, since both Bank 1 and Bank 2 enter Autoselect mode, the simultaneous operation cannot be executed.

## • Write

Device erasure and programming are accomplished via the command register. The contents of the register serve as inputs to the internal state machine. The state machine outputs dictate the function of the device.

The command register itself does not occupy any addressable memory location. The register is a latch used to store the commands, along with the address and data information needed to execute the command. The command register is written by bringing  $\overline{WE}$  to V<sub>IL</sub>, while  $\overline{CE}$  is at V<sub>IL</sub> and  $\overline{OE}$  is at V<sub>IH</sub>. Addresses are latched on the falling edge of  $\overline{WE}$  or  $\overline{CE}$ , whichever happens later; while data is latched on the rising edge of  $\overline{WE}$  or  $\overline{CE}$ , whichever happens first. Standard microprocessor write timings are used.

Refer to “AC Characteristics” in ■ELECTRICAL CHARACTERISTICS and ■TIMING DIAGRAM.

## • Sector Group Protection

The MBM29DL16XTE/BE feature hardware sector group protection. This feature will disable both program and erase operations in any combination of 17 sector groups of memory. (See “Sector Group Addresses Tables (MBM29DL16XTE/BE)” in ■FLEXIBLE SECTOR-ERASE ARCHITECTURE). The sector group protection feature is enabled using programming equipment at the user’s site. The device is shipped with all sector groups unprotected.

To activate this mode, the programming equipment must force V<sub>ID</sub> on address pin A<sub>9</sub> and control pin  $\overline{OE}$ , (suggest V<sub>ID</sub> = 11.5 V),  $\overline{CE}$  = V<sub>IL</sub> and A<sub>0</sub> = A<sub>6</sub> = V<sub>IL</sub>, A<sub>1</sub> = V<sub>IH</sub>. The sector group addresses (A<sub>19</sub>, A<sub>18</sub>, A<sub>17</sub>, A<sub>16</sub>, A<sub>15</sub>, A<sub>14</sub>, A<sub>13</sub>, and A<sub>12</sub>) should be set to the sector to be protected. “Sector Address Tables (MBM29DL161TE/BE, MBM29DL162TE/BE, MBM29DL163TE/BE, MBM29DL164TE/BE)” in ■FLEXIBLE SECTOR-ERASE ARCHITECTURE define the sector address for each of the thirty nine (39) individual sectors, and “Sector Group Addresses Tables (MBM29DL16XTE/BE)” in ■FLEXIBLE SECTOR-ERASE ARCHITECTURE define the sector group address for each of the seventeen (17) individual group sectors. Programming of the protection circuitry begins on the falling edge of the  $\overline{WE}$  pulse and is terminated with the rising edge of the same. Sector group addresses must be held constant during the  $\overline{WE}$  pulse. See “(15) AC Waveforms for Sector Group Protection” in ■TIMING DIAGRAM and “(5) Sector Group Protection Algorithm” in ■FLOW CHART for sector group protection waveforms and algorithm.

To verify programming of the protection circuitry, the programming equipment must force V<sub>ID</sub> on address pin A<sub>9</sub> with  $\overline{CE}$  and  $\overline{OE}$  at V<sub>IL</sub> and  $\overline{WE}$  at V<sub>IH</sub>. Scanning the sector group addresses (A<sub>19</sub>, A<sub>18</sub>, A<sub>17</sub>, A<sub>16</sub>, A<sub>15</sub>, A<sub>14</sub>, A<sub>13</sub>, and A<sub>12</sub>) while (A<sub>6</sub>, A<sub>1</sub>, A<sub>0</sub>) = (0, 1, 0) will produce a logical “1” code at device output DQ<sub>0</sub> for a protected sector. Otherwise the device will produce “0” for unprotected sector. In this mode, the lower order addresses, except for A<sub>6</sub>, A<sub>1</sub>, and A<sub>0</sub> are DON’T CARES. Address locations with A<sub>1</sub> = V<sub>IL</sub> are reserved for Autoselect manufacturer and device codes. A<sub>-1</sub> requires to apply to V<sub>IL</sub> on byte mode.

It is also possible to determine if a sector group is protected in the system by writing an Autoselect command. Performing a read operation at the address location XX02h, where the higher order addresses (A<sub>19</sub>, A<sub>18</sub>, A<sub>17</sub>, A<sub>16</sub>, A<sub>15</sub>, A<sub>14</sub>, A<sub>13</sub>, and A<sub>12</sub>) are the desired sector group address will produce a logical “1” at DQ<sub>0</sub> for a protected sector

group. See “MBM29DL16XTE/BE Sector Group Protection Verify Autoselect Codes Tables” and “Expanded Autoselect Code Tables” in ■DEVICE BUS OPERATION for Autoselect codes.

## • Temporary Sector Group Unprotection

This feature allows temporary unprotection of previously protected sector groups of the MBM29DL16XTE/BE devices in order to change data. The Sector Group Unprotection mode is activated by setting the  $\overline{\text{RESET}}$  pin to high voltage ( $V_{\text{ID}}$ ). During this mode, formerly protected sector groups can be programmed or erased by selecting the sector group addresses. Once the  $V_{\text{ID}}$  is taken away from the  $\overline{\text{RESET}}$  pin, all the previously protected sector groups will be protected again. Refer to “(16) Temporary Sector Group Unprotection Timing Diagram” in ■TIMING DIAGRAM and “(6) Temporary Sector Group Protection Algorithm” in ■FLOW CHART.

## • RESET

### Hardware Reset

The MBM29DL16XTE/BE devices may be reset by driving the  $\overline{\text{RESET}}$  pin to  $V_{\text{IL}}$ . The  $\overline{\text{RESET}}$  pin has a pulse requirement and has to be kept low ( $V_{\text{IL}}$ ) for at least “ $t_{\text{RP}}$ ” in order to properly reset the internal state machine. Any operation in the process of being executed will be terminated and the internal state machine will be reset to the read mode “ $t_{\text{READY}}$ ” after the  $\overline{\text{RESET}}$  pin is driven low. Furthermore, once the  $\overline{\text{RESET}}$  pin goes high, the devices require an additional “ $t_{\text{RH}}$ ” before it will allow read access. When the  $\overline{\text{RESET}}$  pin is low, the devices will be in the standby mode for the duration of the pulse and all the data output pins will be tri-stated. If a hardware reset occurs during a program or erase operation, the data at that particular location will be corrupted. Please note that the  $\text{RY}/\overline{\text{BY}}$  output signal should be ignored during the  $\overline{\text{RESET}}$  pulse. See “(11)  $\overline{\text{RESET}}$ ,  $\text{RY}/\overline{\text{BY}}$  Timing Diagram” in ■TIMING DIAGRAM for the timing diagram. Refer to Temporary Sector Group Unprotection for additional functionality.

## • Byte/Word Configuration

The  $\overline{\text{BYTE}}$  pin selects the byte (8-bit) mode or word (16-bit) mode for the MBM29DL16XTE/BE devices. When this pin is driven high, the devices operate in the word (16-bit) mode. The data is read and programmed at  $\text{DQ}_{15}$  to  $\text{DQ}_0$ . When this pin is driven low, the devices operate in byte (8-bit) mode. Under this mode, the  $\text{DQ}_{15}/\text{A}_{-1}$  pin becomes the lowest address bit and  $\text{DQ}_{14}$  to  $\text{DQ}_8$  bits are tri-stated. Refer to “(12) Timing Diagram for Word Mode Configuration”, “(13) Timing Diagram for Byte Mode Configuration” and “(14)  $\overline{\text{BYTE}}$  Timing Diagram for Write Operations” in ■TIMING DIAGRAM.

## • Boot Block Sector Protection

The Write Protect function provides a hardware method of protecting certain boot sectors without using  $V_{\text{ID}}$ . This function is one of two provided by the  $\overline{\text{WP}}/\text{ACC}$  pin.

If the system asserts  $V_{\text{IL}}$  on the  $\overline{\text{WP}}/\text{ACC}$  pin, the device disables program and erase functions in the two “outermost” 8K byte boot sectors (MBM29DL16XTE: SA37 and SA38, MBM29DL16XBE: SA0 and SA1) independently of whether those sectors were protected or unprotected using the method described in “Sector Group Protection”. The two outermost 8K byte boot sectors are the two sectors containing the lowest addresses in a bottom-boot-configured device, or the two sectors containing the highest addresses in a top-boot-configured device.

If the system asserts  $V_{\text{IH}}$  on the  $\overline{\text{WP}}/\text{ACC}$  pin, the device reverts to whether the two outermost 8K byte boot sectors were last set to be protected or unprotected. That is, sector group protection or unprotection for these two sectors depends on whether they were last protected or unprotected using the method described in “Sector Group Protection”.

- **Accelerated Program Operation**

MBM29DL16XTE/BE offer accelerated program operation which enables the programming in high speed. If the system asserts  $V_{ACC}$  to the  $\overline{WP}/ACC$  pin, the device automatically enters the acceleration mode and the time required for program operation will reduce to about 60%. This function is primarily intended to allow high speed program, so caution is needed as the sector group will temporarily be unprotected.

The system would use a fast program command sequence when programming during acceleration mode. Set command to fast mode and reset command from fast mode are not necessary. When the device enters the acceleration mode, the device automatically set to fast mode. Therefore, the present sequence could be used for programming and detection of completion during acceleration mode.

Removing  $V_{ACC}$  from the  $\overline{WP}/ACC$  pin returns the device to normal operation. Do not remove  $V_{ACC}$  from  $\overline{WP}/ACC$  pin while programming. See "(18) Accelerated Program Timing Diagram" in ■TIMING DIAGRAM.

Erase operation at Acceleration mode is strictly prohibited.

## ■ COMMAND DEFINITIONS

Device operations are selected by writing specific address and data sequences into the command register. Writing incorrect address and data values or writing them in the improper sequence will reset the devices to the read mode. Some commands are required Bank Address (BA) input. When command sequences are inputted to bank being read, the commands have priority than reading. “MBM29DL16XTE/BE Command Definitions Table” in ■DEVICE BUS OPERATION defines the valid register command sequences. Note that the Erase Suspend (B0h) and Erase Resume (30h) commands are valid only while the Sector Erase operation is in progress. Moreover both Read/Reset commands are functionally equivalent, resetting the device to the read mode. Please note that commands are always written at DQ<sub>7</sub> to DQ<sub>0</sub> and DQ<sub>15</sub> to DQ<sub>8</sub> bits are ignored.

### • Read/Reset Command

In order to return from Autoselect mode or Exceeded Timing Limits (DQ<sub>5</sub> = 1) to Read/Reset mode, the Read/Reset operation is initiated by writing the Read/Reset command sequence into the command register. Microprocessor read cycles retrieve array data from the memory. The devices remain enabled for reads until the command register contents are altered.

The devices will automatically power-up in the Read/Reset state. In this case, a command sequence is not required to read data. Standard microprocessor read cycles will retrieve array data. This default value ensures that no spurious alteration of the memory content occurs during the power transition. Refer to “2. AC Characteristics • Read Only Operations Characteristics” in ■ELECTRICAL CHARACTERISTICS and ■TIMING DIAGRAM.

### • Autoselect Command

Flash memories are intended for use in applications where the local CPU alters memory contents. As such, manufacture and device codes must be accessible while the devices reside in the target system. PROM programmers typically access the signature codes by raising A<sub>9</sub> to a high voltage. However, multiplexing high voltage onto the address lines is not generally desired system design practice.

The device contains an Autoselect command operation to supplement traditional PROM programming methodology. The operation is initiated by writing the Autoselect command sequence into the command register.

The Autoselect command sequence is initiated by first writing two unlock cycles. This is followed by a third write cycle that contains the bank address (BA) and the Autoselect command. Then the manufacture and device codes can be read from the bank, and an actual data of memory cell can be read from the another bank.

Following the command write, a read cycle from address (BA)00h retrieves the manufacture code of 04h. A read cycle from address (BA)01h for ×16((BA)02h for ×8) returns the device code (MBM29DL161TE = 36h and MBM29DL161BE = 39h for ×8 mode; MBM29DL161TE = 2236h and MBM29DL161BE = 2239h for ×16 mode), (MBM29DL162TE = 2Dh and MBM29DL162BE = 2Eh for ×8 mode; MBM29DL162TE = 222Dh and MBM29DL162BE = 222Eh for ×16 mode), (MBM29DL163TE = 28h and MBM29DL163BE = 2Bh for ×8 mode; MBM29DL163TE = 2228h and MBM29DL163BE = 222Bh for ×16 mode), (MBM29DL164TE = 33h and MBM29DL164BE = 35h for ×8 mode; MBM29DL164TE = 2233h and MBM29DL164BE = 2235h for ×16 mode). (See “MBM29DL16XTE/BE Sector Group Protection Verify Autoselect Codes Tables” and “Expanded Autoselect Code Tables” in ■DEVICE BUS OPERATION.)

All manufacturer and device codes will exhibit odd parity with DQ<sub>7</sub> defined as the parity bit. Sector state (protection or unprotection) will be informed by address (BA)02h for ×16 ((BA)04h for ×8). Scanning the sector group addresses (A<sub>19</sub>, A<sub>18</sub>, A<sub>17</sub>, A<sub>16</sub>, A<sub>15</sub>, A<sub>14</sub>, A<sub>13</sub>, and A<sub>12</sub>) while (A<sub>6</sub>, A<sub>1</sub>, A<sub>0</sub>) = (0, 1, 0) will produce a logical “1” at device output DQ<sub>0</sub> for a protected sector group. The programming verification should be performed by verify sector group protection on the protected sector. (See “MBM29DL16XTE/BE User Bus Operations Tables (BYT<sub>E</sub> = V<sub>IH</sub> and BYT<sub>E</sub> = V<sub>IL</sub>)” in ■DEVICE BUS OPERATION.)

The manufacture and device codes can be allowed reading from selected bank. To read the manufacture and device codes and sector group protection status from non-selected bank, it is necessary to write Read/Reset command sequence into the register and then Autoselect command should be written into the bank to be read.

If the software (program code) for Autoselect command is stored into the Flash memory, the device and manufacture codes should be read from the other bank where is not contain the software.

To terminate the operation, it is necessary to write the Read/Reset command sequence into the register, and also to write the Autoselect command during the operation, execute it after writing Read/Reset command sequence.

## • Byte/Word Programming

The devices are programmed on a byte-by-byte (or word-by-word) basis. Programming is a four bus cycle operation. There are two “unlock” write cycles. These are followed by the program set-up command and data write cycles. Addresses are latched on the falling edge of  $\overline{CE}$  or  $\overline{WE}$ , whichever happens later and the data is latched on the rising edge of  $\overline{CE}$  or  $\overline{WE}$ , whichever happens first. The rising edge of  $\overline{CE}$  or  $\overline{WE}$  (whichever happens first) begins programming. Upon executing the Embedded Program Algorithm command sequence, the system is not required to provide further controls or timings. The device will automatically provide adequate internally generated program pulses and verify the programmed cell margin.

The system can determine the status of the program operation by using  $DQ_7$  ( $\overline{Data}$  Polling),  $DQ_6$  (Toggle Bit), or  $RY/\overline{BY}$ . The  $\overline{Data}$  Polling and Toggle Bit must be performed at the memory location which is being programmed.

The automatic programming operation is completed when the data on  $DQ_7$  is equivalent to data written to this bit at which time the devices return to the read mode and addresses are no longer latched. (See “Hardware Sequence Flags Table”.) Therefore, the devices require that a valid address to the devices be supplied by the system at this particular instance of time. Hence,  $\overline{Data}$  Polling must be performed at the memory location which is being programmed.

Any commands written to the chip during this period will be ignored. If hardware reset occurs during the programming operation, it is impossible to guarantee the data are being written.

Programming is allowed in any sequence and across sector boundaries. Beware that a data “0” cannot be programmed back to a “1”. Attempting to do so may either hang up the device or result in an apparent success according to the data polling algorithm but a read from Read/Reset mode will show that the data is still “0”. Only erase operations can convert “0”s to “1”s.

“(1) Embedded Program™ Algorithm” in ■FLOW CHART illustrates the Embedded Program™ Algorithm using typical command strings and bus operations.

## • Program Suspend/Resume

The Program Suspend command allows the system to interrupt a program operation so that data can be read from any address. Writing the Program Suspend command (B0h) during the Embedded Program operation immediately suspends the programming. The Program Suspend command may also be issued during a programming operation while an erase is suspend. The bank addresses of sector being programmed should be set when writing the Program Suspend command.

When the Program Suspend command is written during a programming process, the device halts the program operation within 1  $\mu$ s and updates the status bits.

After the program operation has been suspended, the system can read data from any address. The data at program-suspend address is not valid. Normal read timing and command definitions apply.



After the Program Resume command (30h) is written, the device reverts to programming. The bank addresses of sector being suspended should be set when writing the Program Resume command. The system can determine the status of the program operation using the DQ<sub>7</sub> or DQ<sub>6</sub> status bits, just as in the standard program operation. See “Write Operation Status” for more information.

The system may also write the autoselect command sequence when the device is in the Program Suspend mode.

The device allows reading autoselect codes at the addresses within programming sectors, since the codes are not stored in the memory. When the device exits the autoselect mode, the device reverts to the Program Suspend mode, and is ready for another valid operation. See “Autoselect Command” for more information.

The system must write the Program Resume command (address bits are “Bank Address”) to exit the Program Suspend mode and continue the programming operation. Further writes of the Resume command are ignored. Another Program Suspend command can be written after the device has resumed programming.

## • Chip Erase

Chip erase is a six bus cycle operation. There are two “unlock” write cycles. These are followed by writing the “set-up” command. Two more “unlock” write cycles are then followed by the chip erase command.

Chip erase does not require the user to program the device prior to erase. Upon executing the Embedded Erase Algorithm command sequence the devices will automatically program and verify the entire memory for an all zero data pattern prior to electrical erase (Preprogram function). The system is not required to provide any controls or timings during these operations.

The system can determine the status of the erase operation by using DQ<sub>7</sub> ( $\overline{\text{Data}}$  Polling), DQ<sub>6</sub> (Toggle Bit), or RY/ $\overline{\text{BY}}$ . The chip erase begins on the rising edge of the last  $\overline{\text{CE}}$  or  $\overline{\text{WE}}$ , whichever happens first in the command sequence and terminates when the data on DQ<sub>7</sub> is “1” (See “Write Operation Status”.) at which time the device returns to read mode.

Chip Erase Time; Sector Erase Time × All sectors + Chip Program Time (Preprogramming)

“(2) Embedded Erase™ Algorithm” in ■FLOW CHART illustrates the Embedded Erase™ Algorithm using typical command strings and bus operations.

## • Sector Erase

Sector erase is a six bus cycle operation. There are two “unlock” write cycles. These are followed by writing the “set-up” command. Two more “unlock” write cycles are then followed by the Sector Erase command. The sector address (any address location within the desired sector) is latched on the falling edge of  $\overline{\text{CE}}$  or  $\overline{\text{WE}}$  whichever happens later, while the command (Data = 30h) is latched on the rising edge of  $\overline{\text{CE}}$  or  $\overline{\text{WE}}$  which happens first. After time-out of “t<sub>row</sub>” from the rising edge of the last sector erase command, the sector erase operation will begin.

Multiple sectors may be erased concurrently by writing the six bus cycle operations on “MBM29DL16XTE/BE Command Definitions Table” in ■DEVICE BUS OPERATION. This sequence is followed with writes of the Sector Erase command to addresses in other sectors desired to be concurrently erased. The time between writes must be less than “t<sub>row</sub>” otherwise that command will not be accepted and erasure will start. It is recommended that processor interrupts be disabled during this time to guarantee this condition. The interrupts can be re-enabled after the last Sector Erase command is written. A time-out of “t<sub>row</sub>” from the rising edge of last  $\overline{\text{CE}}$  or  $\overline{\text{WE}}$  whichever happens first will initiate the execution of the Sector Erase command(s). If another falling edge of  $\overline{\text{CE}}$  or  $\overline{\text{WE}}$ , whichever happens first occurs within the “t<sub>row</sub>” time-out window the timer is reset. (Monitor DQ<sub>3</sub> to determine if the sector erase timer window is still open, see section DQ<sub>3</sub>, Sector Erase Timer.) Any command other than Sector Erase or Erase Suspend during this time-out period will reset the devices to the read mode, ignoring the previous command string. Resetting the devices once execution has begun will corrupt the data in the sector.

In that case, restart the erase on those sectors and allow them to complete. (Refer to the Write Operation Status section for Sector Erase Timer operation.) Loading the sector erase buffer may be done in any sequence and with any number of sectors (0 to 38).

Sector erase does not require the user to program the devices prior to erase. The devices automatically program all memory locations in the sector(s) to be erased prior to electrical erase (Preprogram function). When erasing a sector or sectors the remaining unselected sectors are not affected. The system is not required to provide any controls or timings during these operations.

The system can determine the status of the erase operation by using  $\overline{DQ_7}$  (Data Polling),  $DQ_6$  (Toggle Bit), or  $\overline{RY/\overline{BY}}$ .

The sector erase begins after the “ $t_{\text{row}}$ ” time out from the rising edge of  $\overline{CE}$  or  $\overline{WE}$  whichever happens first for the last sector erase command pulse and terminates when the data on  $DQ_7$  is “1” (See “Write Operation Status”). at which time the devices return to the read mode.  $\overline{\text{Data}}$  polling and Toggle Bit must be performed at an address within any of the sectors being erased.

Multiple Sector Erase Time; [Sector Erase Time + Sector Program Time (Preprogramming)] × Number of Sector Erase

In case of multiple sector erase across bank boundaries, a read from bank (read-while-erase) can not performe.

“(2) Embedded Erase™ Algorithm” in ■FLOW CHART illustrates the Embedded Erase™ Algorithm using typical command strings and bus operations.

## • Erase Suspend/Resume

The Erase Suspend command allows the user to interrupt a Sector Erase operation and then perform data reads from or programs to a sector not being erased. This command is applicable ONLY during the Sector Erase operation which includes the time-out period for sector erase. The Erase Suspend command will be ignored if written during the Chip Erase operation or Embedded Program Algorithm. Writing the Erase Suspend command (B0h) during the Sector Erase time-out results in immediate termination of the time-out period and suspension of the erase operation.

Writing the Erase Resume command (30h) resumes the erase operation. The bank addresses of sector being erasing or suspending should be set when writing the Erase Suspend or Erase Resume command.

When the Erase Suspend command is written during the Sector Erase operation, the device will take a maximum of “ $t_{\text{SPD}}$ ” to suspend the erase operation. When the devices have entered the erase-suspended mode, the  $\overline{RY/\overline{BY}}$  output pin will be at High-Z and the  $DQ_7$  bit will be at logic “1”, and  $DQ_6$  will stop toggling. The user must use the address of the erasing sector for reading  $DQ_6$  and  $DQ_7$  to determine if the erase operation has been suspended. Further writes of the Erase Suspend command are ignored.

When the erase operation has been suspended, the devices default to the erase-suspend-read mode. Reading data in this mode is the same as reading from the standard read mode except that the data must be read from sectors that have not been erase-suspended. Successively reading from the erase-suspended sector while the device is in the erase-suspend-read mode will cause  $DQ_2$  to toggle. (See “ $DQ_2$ ”.)

After entering the erase-suspend-read mode, the user can program the device by writing the appropriate command sequence for Program. This program mode is known as the erase-suspend-program mode. Again, programming in this mode is the same as programming in the regular Program mode except that the data must be programmed to sectors that are not erase-suspended. Successively reading from the erase-suspended sector while the devices are in the erase-suspend-program mode will cause  $DQ_2$  to toggle. The end of the erase-suspended Program operation is detected by the  $\overline{RY/\overline{BY}}$  output pin,  $\overline{\text{Data}}$  polling of  $DQ_7$  or by the Toggle Bit I ( $DQ_6$ ) which is the same as the regular Program operation. Note that  $DQ_7$  must be read from the Program address while  $DQ_6$  can be read from any address within bank being erase-suspended.

To resume the operation of Sector Erase, the Resume command (30h) should be written to the bank being erase suspended. Any further writes of the Resume command at this point will be ignored. Another Erase Suspend command can be written after the chip has resumed erasing.

## • Extended Command

### (1) Fast Mode

MBM29DL16XTE/BE have Fast Mode function. This mode dispenses with the initial two unlock cycles required in the standard program command sequence by writing Fast Mode command into the command register. In this mode, the required bus cycle for programming is two cycles instead of four bus cycles in standard program command. (Do not write erase command in this mode.) The read operation is also executed after exiting this mode. To exit this mode, it is necessary to write Fast Mode Reset command into the command register. The first cycle must contain the bank address. (Refer to “(7) Embedded Program™ Algorithm for Fast Mode” in ■FLOW CHART.) The  $V_{CC}$  active current is required even  $\overline{CE} = V_{IH}$  during Fast Mode.

### (2) Fast Programming

During Fast Mode, the programming can be executed with two bus cycles operation. The Embedded Program Algorithm is executed by writing program set-up command (A0h) and data write cycles (PA/PD). (Refer to “(7) Embedded Program™ Algorithm for Fast Mode” in ■FLOW CHART.)

### (3) Extended Sector Group Protection

In addition to normal sector group protection, the MBM29DL16XTE/BE have Extended Sector Group Protection as extended function. This function enables to protect sector group by forcing  $V_{ID}$  on  $\overline{RESET}$  pin and write a command sequence. Unlike conventional procedure, it is not necessary to force  $V_{ID}$  and control timing for control pins. The extended sector group protection requires  $V_{ID}$  on  $\overline{RESET}$  pin only. With this condition, the operation is initiated by writing the set-up command (60h) into the command register. Then, the sector group addresses pins ( $A_{20}, A_{19}, A_{18}, A_{17}, A_{16}, A_{15}, A_{14}, A_{13}$  and  $A_{12}$ ) and ( $A_6, A_1, A_0$ ) = (0, 1, 0) should be set to the sector group to be protected (recommend to set  $V_{IL}$  for the other addresses pins), and write extended sector group protection command (60h). A sector group is typically protected in 250  $\mu s$ . To verify programming of the protection circuitry, the sector group addresses pins ( $A_{20}, A_{19}, A_{18}, A_{17}, A_{16}, A_{15}, A_{14}, A_{13}$  and  $A_{12}$ ) and ( $A_6, A_1, A_0$ ) = (0, 1, 0) should be set and write a command (40h). Following the command write, a logical “1” at device output  $DQ_0$  will produce for protected sector in the read operation. If the output data is logical “0”, please repeat to write extended sector group protection command (60h) again. To terminate the operation, it is necessary to set  $\overline{RESET}$  pin to  $V_{IH}$ . (Refer to “(17) Extended Sector Group Protection Timing Diagram” in ■TIMING DIAGRAM and “(8) Extended Sector Group Protection Algorithm” in ■FLOW CHART.)

### (4) CFI (Common Flash Memory Interface)

The CFI (Common Flash Memory Interface) specification outlines device and host system software interrogation handshake which allows specific vendor-specified software algorithms to be used for entire families of devices. This allows device-independent, JEDEC ID-independent, and forward-and backward-compatible software support for the specified flash device families. Refer to CFI specification in detail.

The operation is initiated by writing the query command (98h) into the command register. The bank address should be set when writing this command. Then the device information can be read from the bank, and an actual data of memory cell be read from the another bank. Following the command write, a read cycle from specific address retrieves device information. Please note that output data of upper byte ( $DQ_{15}$  to  $DQ_8$ ) is “0” in word mode (16 bit) read. Refer to “Common Flash Memory Interface Code Table” in ■FLEXIBLE SECTOR-ERASE ARCHITECTURE. To terminate operation, it is necessary to write the read/reset command sequence into the register. (See “Common Flash Memory Interface Code Table” in ■FLEXIBLE SECTOR-ERASE ARCHITECTURE.)

## • HiddenROM Region

The HiddenROM feature provides a Flash memory region that the system may access through a new command sequence. This is primarily intended for customers who wish to use an Electronic Serial Number (ESN) in the device with the ESN protected against modification. Once the HiddenROM region is protected, any further modification of that region is impossible. This ensures the security of the ESN once the product is shipped to the field.

The HiddenROM region is 64 Kbytes in length and is stored at the same address of the 8 KB ×8 sectors. The MBM29DL16XTE occupies the address of the byte mode 1F0000h to 1FFFFFFh (word mode 0F8000h to 0FFFFFFh) and the MBM29DL16XBE type occupies the address of the byte mode 000000h to 00FFFFFFh (word mode 000000h to 007FFFh). After the system has written the Enter HiddenROM command sequence, the system may read the HiddenROM region by using the addresses normally occupied by the boot sectors. That is, the device sends all commands that would normally be sent to the boot sectors to the HiddenROM region. This mode of operation continues until the system issues the Exit HiddenROM command sequence, or until power is removed from the device. On power-up, or following a hardware reset, the device reverts to sending commands to the boot sectors.

## • HiddenROM Entry Command

MBM29DL16XTE/BE have a HiddenROM area with One Time Protect function. This area is to enter the security code and to unable the change of the code once set. Program/erase is possible in this area until it is protected. However, once it is protected, it is impossible to unprotect, so please use this with caution.

HiddenROM area is 64 Kbyte and in the same address area of 8 KB sectors. The address of top boot is 1F0000h to 1FFFFFFh at byte mode (0F8000h to 0FFFFFFh at word mode) and the bottom boot is 000000h to 00FFFFFFh at byte mode (000000h to 007FFFh at word mode). These areas are normally the boot block area (8 KB × 8 sectors). Therefore, write the HiddenROM entry command sequence to enter the HiddenROM area. It is called as HiddenROM mode when the HiddenROM area appears.

Sector other than the boot block area could be read during HiddenROM mode. Read/program/erase of the HiddenROM area is possible during HiddenROM mode. Write the HiddenROM reset command sequence to exit the HiddenROM mode. The bank address of the HiddenROM should be set on the third cycle of this reset command sequence.

In case of MBM29DL161TE/BE, whose Bank 1 size is 0.5 Mbit, the simultaneous operation cannot execute multi-function mode between the HiddenROM area and Bank 2 Region.

## • HiddenROM Program Command

To program the data to the HiddenROM area, write the HiddenROM program command sequence during HiddenROM mode. This command is the same as the program command in the past except to write the command during HiddenROM mode. Therefore the detection of completion method is the same as in the past, using the DQ<sub>7</sub> data poling, DQ<sub>6</sub> toggle bit and RY/ $\overline{\text{BY}}$  pin. Need to pay attention to the address to be programmed. If the address other than the HiddenROM area is selected to program, the data of the address will be changed.

## • HiddenROM Erase Command

To erase the HiddenROM area, write the HiddenROM erase command sequence during HiddenROM mode. This command is the same as the sector erase command in the past except to write the command during HiddenROM mode. Therefore the detection of completion method is the same as in the past, using the DQ<sub>7</sub> data poling, DQ<sub>6</sub> toggle bit and RY/ $\overline{\text{BY}}$  pin. Need to pay attention to the sector address to be erased. If the sector address other than the HiddenROM area is selected, the data of the sector will be changed.

## • HiddenROM Protect Command

There are two methods to protect the HiddenROM area. One is to write the sector group protect setup command (60h), set the sector address in the HiddenROM area and  $(A_6, A_1, A_0) = (0, 1, 0)$ , and write the sector group protect command (60h) during the HiddenROM mode. The same command sequence could be used because except that it is in the HiddenROM mode and that it does not apply high voltage to  $\overline{\text{RESET}}$  pin, it is the same as the extension sector group protect in the past. Please refer to "Extended Command (3) Extended Sector Group Protection" for details of extension sector group protect setting.

The other is to apply high voltage ( $V_{ID}$ ) to  $A_9$  and  $\overline{\text{OE}}$ , set the sector address in the HiddenROM area and  $(A_6, A_1, A_0) = (0, 1, 0)$ , and apply the write pulse during the HiddenROM mode. To verify the protect circuit, apply high voltage ( $V_{ID}$ ) to  $A_9$ , specify  $(A_6, A_1, A_0) = (0, 1, 0)$  and the sector address in the HiddenROM area, and read. When "1" appears to  $DQ_0$ , the protect setting is completed. "0" will appear to  $DQ_0$  if it is not protected. Please apply write pulse again. The same command sequence could be used for the above method because other than the HiddenROM mode, it is the same as the sector group protect in the past. Please refer to "Sector Group Protection" in ■FUNCTIONAL DESCRIPTION for details of sector group protect setting

Other sector group will be effected if the address other than the HiddenROM area is selected for the sector group address, so please be careful. Once it is protected, protection can not be cancelled, so please pay closest attention.

## • Write Operation Status

Detailed in "Hardware Sequence Flags Table" are all the status flags that can determine the status of the bank for the current mode operation. The read operation from the bank where is not operate Embedded Algorithm returns a data of memory cell. These bits offer a method for determining whether a Embedded Algorithm is completed properly. Information on  $DQ_2$  is address sensitive. This means that if an address from an erasing sector is consecutively read, then the  $DQ_2$  bit will toggle. However,  $DQ_2$  will not toggle if an address from a non-erasing sector is consecutively read. This allows the user to determine which sectors are erasing and which are not.

The status flag is not output from bank (non-busy bank) not executing Embedded Algorithm. For example, there is bank (busy bank) which is now executing Embedded Algorithm. When the read sequence is [1] <busy bank>, [2] <non-busy bank>, [3] <busy bank>, the  $DQ_6$  is toggling in the case of [1] and [3]. In case of [2], the data of memory cell is outputted. In the erase-suspend read mode with the same read sequence,  $DQ_6$  will not be toggled in the [1] and [3].

In the erase suspend read mode,  $DQ_2$  is toggled in the [1] and [3]. In case of [2], the data of memory cell is outputted.

Hardware Sequence Flags Table

Status		DQ <sub>7</sub>	DQ <sub>6</sub>	DQ <sub>5</sub>	DQ <sub>3</sub>	DQ <sub>2</sub>	
In Progress	Embedded Program Algorithm		$\overline{\text{DQ}}_7$	Toggle	0	0	1
	Embedded Erase Algorithm		0	Toggle	0	1	Toggle*1
	Erase Suspended Mode	Erase Suspend Read (Erase Suspended Sector)	1	1	0	0	Toggle
		Erase Suspend Read (Non-Erase Suspended Sector)	Data	Data	Data	Data	Data
		Erase Suspend Program (Non-Erase Suspended Sector)	$\overline{\text{DQ}}_7$	Toggle	0	0	1 *2
	Program Suspended Mode	Program Suspend Read (Program Suspended Sector)	Data	Data	Data	Data	Data
Program Suspend Read (Non-Program Suspended Sector)		Data	Data	Data	Data	Data	
Exceeded Time Limits	Embedded Program Algorithm		$\overline{\text{DQ}}_7$	Toggle	1	0	1
	Embedded Erase Algorithm		0	Toggle	1	1	N/A
	Erase Suspended Mode	Erase Suspend Program (Non-Erase Suspended Sector)	$\overline{\text{DQ}}_7$	Toggle	1	0	N/A

\*1 : Successive reads from the erasing or erase-suspend sector cause DQ<sub>2</sub> to toggle.

\*2 : Reading from non-erase suspend sector address indicates logic "1" at the DQ<sub>2</sub> bit.

• **DQ<sub>7</sub>**

Data Polling

The MBM29DL16XTE/BE devices feature Data Polling as a method to indicate to the host that the Embedded Algorithms are in progress or completed. During the Embedded Program Algorithm an attempt to read the devices will produce the complement of the data last written to DQ<sub>7</sub>. Upon completion of the Embedded Program Algorithm, an attempt to read the device will produce the true data last written to DQ<sub>7</sub>. During the Embedded Erase Algorithm, an attempt to read the device will produce a "0" at the DQ<sub>7</sub> output. Upon completion of the Embedded Erase Algorithm an attempt to read the device will produce a "1" at the DQ<sub>7</sub> output. The flowchart for Data Polling (DQ<sub>7</sub>) is shown in "(3) Data Polling Algorithm" in ■FLOW CHART.

For programming, the Data Polling is valid after the rising edge of fourth write pulse in the four write pulse sequence.

For chip erase and sector erase, the Data Polling is valid after the rising edge of the sixth write pulse in the six write pulse sequence. Data Polling must be performed at sector address within any of the sectors being erased and not a protected sector. Otherwise, the status may not be valid.

If a program address falls within a protected sector, Data Polling on DQ<sub>7</sub> is active for approximately 1 μs, then that bank returns to the read mode. After an erase command sequence is written, if all sectors selected for erasing are protected, Data Polling on DQ<sub>7</sub> is active for approximately 400 μs, then the bank returns to read mode.

Once the Embedded Algorithm operation is close to being completed, the MBM29DL16XTE/BE data pins (DQ<sub>7</sub>) may change asynchronously while the output enable ( $\overline{\text{OE}}$ ) is asserted low. This means that the devices are

driving status information on DQ<sub>7</sub> at one instant of time and then that byte's valid data at the next instant of time. Depending on when the system samples the DQ<sub>7</sub> output, it may read the status or valid data. Even if the device has completed the Embedded Algorithm operation and DQ<sub>7</sub> has a valid data, the data outputs on DQ<sub>6</sub> to DQ<sub>0</sub> may be still invalid. The valid data on DQ<sub>7</sub> to DQ<sub>0</sub> will be read on the successive read attempts.

The  $\overline{\text{Data}}$  Polling feature is only active during the Embedded Programming Algorithm, Embedded Erase Algorithm or sector erase time-out. (See "Hardware Sequence Flags Table".)

See "(6) AC Waveforms for  $\overline{\text{Data}}$  Polling during Embedded Algorithm Operations" in ■TIMING DIAGRAM for the  $\overline{\text{Data}}$  Polling timing specifications and diagrams.

## • DQ<sub>6</sub>

### Toggle Bit I

The MBM29DL16XTE/BE also feature the "Toggle Bit I" as a method to indicate to the host system that the Embedded Algorithms are in progress or completed.

During an Embedded Program or Erase Algorithm cycle, successive attempts to read ( $\overline{\text{CE}}$  or  $\overline{\text{OE}}$  toggling) data from the devices will result in DQ<sub>6</sub> toggling between "1" and "0". Once the Embedded Program or Erase Algorithm cycle is completed, DQ<sub>6</sub> will stop toggling and valid data will be read on the next successive attempts. During programming, the Toggle Bit I is valid after the rising edge of the fourth write pulse in the four write pulse sequence. For chip erase and sector erase, the Toggle Bit I is valid after the rising edge of the sixth write pulse in the six write pulse sequence. The Toggle Bit I is active during the sector time out.

In programming, if the sector being written to is protected, the toggle bit will toggle for about 1  $\mu\text{s}$  and then stop toggling without the data having changed. In erase, the devices will erase all the selected sectors except for the ones that are protected. If all selected sectors are protected, the chip will toggle the toggle bit for about 400  $\mu\text{s}$  and then drop back into read mode, having changed none of the data.

Either  $\overline{\text{CE}}$  or  $\overline{\text{OE}}$  toggling will cause the DQ<sub>6</sub> to toggle. In addition, an Erase Suspend/Resume command will cause the DQ<sub>6</sub> to toggle.

The system can use DQ<sub>6</sub> to determine whether a sector is actively erasing or is erase-suspended. When a bank is actively erasing (that is, the Embedded Erase Algorithm is in progress), DQ<sub>6</sub> toggles. When a bank enters the Erase Suspend mode, DQ<sub>6</sub> stops toggling. Successive read cycles during the erase-suspend-program cause DQ<sub>6</sub> to toggle.

To operate toggle bit function properly,  $\overline{\text{CE}}$  or  $\overline{\text{OE}}$  must be high when bank address is changed.

See "(7) AC Waveforms for Toggle Bit I during Embedded Algorithm Operations" in ■TIMING DIAGRAM for the Toggle Bit I timing specifications and diagrams.

## • DQ<sub>5</sub>

### Exceeded Timing Limits

DQ<sub>5</sub> will indicate if the program or erase time has exceeded the specified limits (internal pulse count). Under these conditions DQ<sub>5</sub> will produce a "1". This is a failure condition which indicates that the program or erase cycle was not successfully completed.  $\overline{\text{Data}}$  Polling is the only operating function of the devices under this condition. The  $\overline{\text{CE}}$  circuit will partially power down the device under these conditions (to approximately 2 mA). The  $\overline{\text{OE}}$  and  $\overline{\text{WE}}$  pins will control the output disable functions as described in "MBM29DL16XTE/BE User Bus Operations Tables ( $\overline{\text{BYTE}} = V_{\text{IH}}$  and  $\overline{\text{BYTE}} = V_{\text{IL}}$ )" in ■DEVICE BUS OPERATION.

The DQ<sub>5</sub> failure condition may also appear if a user tries to program a non blank location without erasing. In this case the devices lock out and never complete the Embedded Algorithm operation. Hence, the system never reads a valid data on DQ<sub>7</sub> bit and DQ<sub>6</sub> never stops toggling. Once the devices have exceeded timing limits, the

DQ<sub>5</sub> bit will indicate a “1.” Please note that this is not a device failure condition since the devices were incorrectly used. If this occurs, reset the device with command sequence.

- **DQ<sub>3</sub>**

## Sector Erase Timer

After the completion of the initial sector erase command sequence the sector erase time-out will begin. DQ<sub>3</sub> will remain low until the time-out is complete.  $\overline{\text{Data}}$  Polling and Toggle Bit are valid after the initial sector erase command sequence.

If  $\overline{\text{Data}}$  Polling or the Toggle Bit I indicates the device has been written with a valid erase command, DQ<sub>3</sub> may be used to determine if the sector erase timer window is still open. If DQ<sub>3</sub> is high (“1”) the internally controlled erase cycle has begun; attempts to write subsequent commands to the device will be ignored until the erase operation is completed as indicated by  $\overline{\text{Data}}$  Polling or Toggle Bit I. If DQ<sub>3</sub> is low (“0”), the device will accept additional sector erase commands. To insure the command has been accepted, the system software should check the status of DQ<sub>3</sub> prior to and following each subsequent Sector Erase command. If DQ<sub>3</sub> were high on the second status check, the command may not have been accepted.

See “Hardware Sequence Flags Table”.

- **DQ<sub>2</sub>**

## Toggle Bit II

This toggle bit II, along with DQ<sub>6</sub>, can be used to determine whether the devices are in the Embedded Erase Algorithm or in Erase Suspend.

Successive reads from the erasing sector will cause DQ<sub>2</sub> to toggle during the Embedded Erase Algorithm. If the devices are in the erase-suspended-read mode, successive reads from the erase-suspended sector will cause DQ<sub>2</sub> to toggle. When the devices are in the erase-suspended-program mode, successive reads from the byte address of the non-erase suspended sector will indicate a logic “1” at the DQ<sub>2</sub> bit.

DQ<sub>6</sub> is different from DQ<sub>2</sub> in that DQ<sub>6</sub> toggles only when the standard program or Erase, or Erase Suspend Program operation is in progress. The behavior of these two status bits, along with that of DQ<sub>7</sub>, is summarized as follows:

For example, DQ<sub>2</sub> and DQ<sub>6</sub> can be used together to determine if the erase-suspend-read mode is in progress. (DQ<sub>2</sub> toggles while DQ<sub>6</sub> does not.) See also “Toggle Bit Status Table” and “(9) DQ<sub>2</sub> vs. DQ<sub>6</sub>” in ■TIMING DIAGRAM.

Furthermore, DQ<sub>2</sub> can also be used to determine which sector is being erased. When the device is in the erase mode, DQ<sub>2</sub> toggles if this bit is read from an erasing sector.

To operate toggle bit function properly,  $\overline{\text{CE}}$  or  $\overline{\text{OE}}$  must be high when bank address is changed.

- **Reading Toggle Bits DQ<sub>6</sub>/DQ<sub>2</sub>**

Whenever the system initially begins reading toggle bit status, it must read DQ<sub>7</sub> to DQ<sub>0</sub> at least twice in a row to determine whether a toggle bit is toggling. Typically a system would note and store the value of the toggle bit after the first read. After the second read, the system would compare the new value of the toggle bit with the first. If the toggle bit is not toggling, the device has completed the program or erase operation. The system can read array data on DQ<sub>7</sub> to DQ<sub>0</sub> on the following read cycle.

However, if, after the initial two read cycles, the system determines that the toggle bit is still toggling, the system also should note whether the value of DQ<sub>5</sub> is high (see the section on “DQ<sub>5</sub>”). If it is, the system should then determine again whether the toggle bit is toggling, since the toggle bit may have stopped toggling just as DQ<sub>5</sub> went high. If the toggle bit is no longer toggling, the device has successfully completed the program or erase



operation. If it is still toggling, the device did not complete the operation successfully, and the system must write the reset command to return to reading array data.

The remaining scenario is that the system initially determines that the toggle bit is toggling and DQ<sub>5</sub> has not gone high. The system may continue to monitor the toggle bit and DQ<sub>5</sub> through successive read cycles, determining the status as described in the previous paragraph. Alternatively, it may choose to perform other system tasks. In this case, the system must start at the beginning of the algorithm when it returns to determine the status of the operation. (Refer to “(4) Toggle Bit Algorithm” in ■FLOW CHART.)

**Toggle Bit Status Table**

Mode	DQ <sub>7</sub>	DQ <sub>6</sub>	DQ <sub>2</sub>
Program	$\overline{DQ_7}$	Toggle	1
Erase	0	Toggle	Toggle* <sup>1</sup>
Erase-Suspend Read (Erase-Suspended Sector)	1	1	Toggle
Erase-Suspend Program	$\overline{DQ_7}$	Toggle	1* <sup>2</sup>

\*1 : Successive reads from the erasing or erase-suspend sector cause DQ<sub>2</sub> to toggle.

\*2 : Reading from the non-erase suspend sector address indicates logic “1” at the DQ<sub>2</sub> bit.

## • RY/ $\overline{BY}$

### Ready/Busy

The MBM29DL16XTE/BE provide a RY/ $\overline{BY}$  open-drain output pin as a way to indicate to the host system that the Embedded Algorithms are either in progress or has been completed. If the output is low, the devices are busy with either a program or erase operation. If the output is high, the devices are ready to accept any read/write or erase operation. When the RY/ $\overline{BY}$  pin is low, the devices will not accept any additional program or erase commands. If the MBM29DL16XTE/BE are placed in an Erase Suspend mode, the RY/ $\overline{BY}$  output will be high.

During programming, the RY/ $\overline{BY}$  pin is driven low after the rising edge of the fourth write pulse. During an erase operation, the RY/ $\overline{BY}$  pin is driven low after the rising edge of the sixth write pulse. The RY/ $\overline{BY}$  pin will indicate a busy condition during the  $\overline{RESET}$  pulse. Refer to “(10) RY/ $\overline{BY}$  Timing Diagram during Program/Erase Operations” and “(11)  $\overline{RESET}$ , RY/ $\overline{BY}$  Timing Diagram” in ■TIMING DIAGRAM for a detailed timing diagram. The RY/ $\overline{BY}$  pin is pulled high in standby mode.

Since this is an open-drain output, the pull-up resistor needs to be connected to V<sub>CC</sub> ; multiples of devices may be connected to the host system via more than one RY/ $\overline{BY}$  pin in parallel.

## • Data Protection

The MBM29DL16XTE/BE are designed to offer protection against accidental erasure or programming caused by spurious system level signals that may exist during power transitions. During power up the devices automatically reset the internal state machine in the Read mode. Also, with its control register architecture, alteration of the memory contents only occurs after successful completion of specific multi-bus cycle command sequences.

The devices also incorporate several features to prevent inadvertent write cycles resulting from V<sub>CC</sub> power-up and power-down transitions or system noise.

- **Low  $V_{CC}$  Write Inhibit**

To avoid initiation of a write cycle during  $V_{CC}$  power-up and power-down, a write cycle is locked out for  $V_{CC}$  less than  $V_{LKO}$  (Min). If  $V_{CC} < V_{LKO}$ , the command register is disabled and all internal program/erase circuits are disabled. Under this condition the device will reset to the read mode. Subsequent writes will be ignored until the  $V_{CC}$  level is greater than  $V_{LKO}$ . It is the users responsibility to ensure that the control pins are logically correct to prevent unintentional writes when  $V_{CC}$  is above  $V_{LKO}$  (Min).

If Embedded Erase Algorithm is interrupted, there is possibility that the erasing sector(s) cannot be used.

- **Write Pulse “Glitch” Protection**

Noise pulses of less than 3 ns (typical) on  $\overline{OE}$ ,  $\overline{CE}$ , or  $\overline{WE}$  will not initiate a write cycle.

- **Logical Inhibit**

Writing is inhibited by holding any one of  $\overline{OE} = V_{IL}$ ,  $\overline{CE} = V_{IH}$ , or  $\overline{WE} = V_{IH}$ . To initiate a write cycle  $\overline{CE}$  and  $\overline{WE}$  must be a logic “0” while  $\overline{OE}$  is a logic “1”.

- **Power-Up Write Inhibit**

Power-up of the devices with  $\overline{WE} = \overline{CE} = V_{IL}$  and  $\overline{OE} = V_{IH}$  will not accept commands on the rising edge of  $\overline{WE}$ . The internal state machine is automatically reset to the read mode on power-up.

- **Sector Group Protection**

Device user is able to protect each sector group individually to store and protect data. Protection circuit voids both program and erase commands that are addressed to protected sectors.

Any command to program or erase addressed to protected sector are ignored (see “Sector Group Protection” in ■ FUNCTIONAL DESCRIPTION).

## ■ ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Rating		Unit
		Min	Max	
Storage Temperature	T <sub>stg</sub>	-55	+125	°C
Ambient Temperature with Power Applied	T <sub>A</sub>	-40	+85	°C
Voltage with Respect to Ground All Pins except A <sub>9</sub> , $\overline{OE}$ , $\overline{RESET}$ *1, *2	V <sub>IN</sub> , V <sub>OUT</sub>	-0.5	V <sub>CC</sub> +0.5	V
Power Supply Voltage *1	V <sub>CC</sub>	-0.5	+4.0	V
A <sub>9</sub> , $\overline{OE}$ , and $\overline{RESET}$ *1, *3	V <sub>IN</sub>	-0.5	+13.0	V
$\overline{WP}/ACC$ *1, *4	V <sub>ACC</sub>	-0.5	+10.5	V

\*1 : Voltage is defined on the basis of V<sub>SS</sub> = GND = 0 V.

\*2 : Minimum DC voltage on input or I/O pins is -0.5 V. During voltage transitions, input or I/O pins may undershoot V<sub>SS</sub> to -2.0 V for periods of up to 20 ns. Maximum DC voltage on input or I/O pins is V<sub>CC</sub> +0.5 V. During voltage transitions, input or I/O pins may overshoot to V<sub>CC</sub> +2.0 V for periods of up to 20 ns.

\*3 : Minimum DC input voltage on A<sub>9</sub>,  $\overline{OE}$  and  $\overline{RESET}$  pins is -0.5 V. During voltage transitions, A<sub>9</sub>,  $\overline{OE}$  and  $\overline{RESET}$  pins may undershoot V<sub>SS</sub> to -2.0 V for periods of up to 20 ns. Voltage difference between input and supply voltage (V<sub>IN</sub>-V<sub>CC</sub>) does not exceed 9.0 V. Maximum DC input voltage on A<sub>9</sub>,  $\overline{OE}$  and  $\overline{RESET}$  pins is +13.0 V which may overshoot to +14.0 V for periods of up to 20 ns.

\*4 : Minimum DC input voltage on  $\overline{WP}/ACC$  pin is -0.5 V. During voltage transitions,  $\overline{WP}/ACC$  pin may undershoot V<sub>SS</sub> to -2.0 V for periods of up to 20 ns. Maximum DC input voltage on  $\overline{WP}/ACC$  pin is +10.5 V which may overshoot to +12.0 V for periods of up to 20 ns when V<sub>CC</sub> is applied.

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

## ■ RECOMMENDED OPERATING CONDITIONS

Parameter	Symbol	Conditions	Value		Unit
			Min	Max	
Ambient Temperature	T <sub>A</sub>	MBM29DL16XTE/BE70/90	-40	+85	°C
Power Supply Voltage*	V <sub>CC</sub>	MBM29DL16XTE/BE70/90	+2.7	+3.6	V

\* : Voltage is defined on the basis of V<sub>SS</sub> = GND = 0 V.

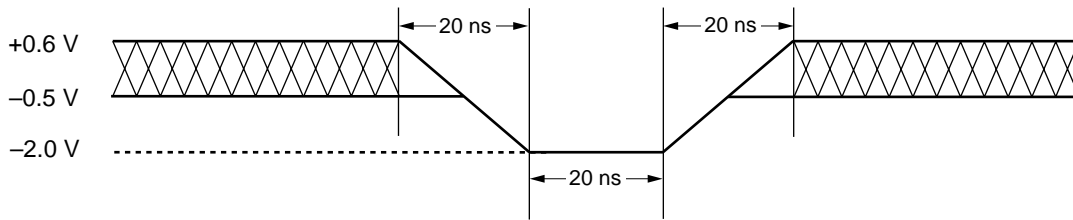
WARNING: The recommended operating conditions are required in order to ensure the normal operation of the semiconductor device. All of the device's electrical characteristics are warranted when the device is operated within these ranges.

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

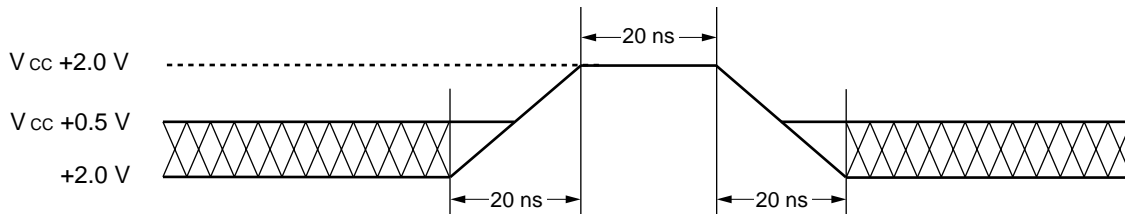
No warranty is made with respect to uses, operating conditions, or combinations not represented on the data sheet. Users considering application outside the listed conditions are advised to contact their FUJITSU representatives beforehand.

# MBM29DL16XTE/BE70/90

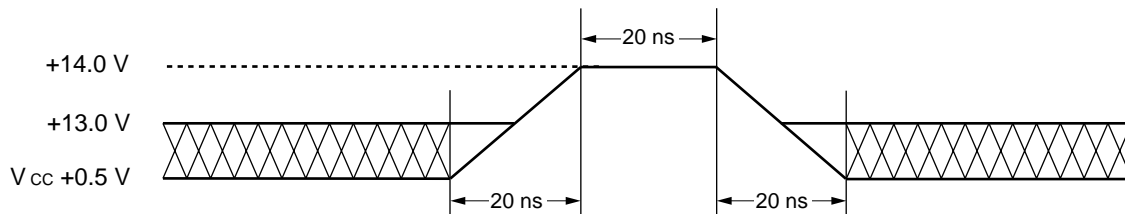
## ■ MAXIMUM OVERSHOOT/MAXIMUM UNDERSHOOT



Maximum Undershoot Waveform



Maximum Overshoot Waveform 1



Note: This waveform is applied for  $A_9$ ,  $\overline{OE}$ , and  $\overline{RESET}$ .

Maximum Overshoot Waveform 2

## ■ ELECTRICAL CHARACTERISTICS

### 1. DC Characteristics

Parameter	Symbol	Conditions	Value			Unit	
			Min	Typ	Max		
Input Leakage Current	$I_{LI}$	$V_{IN} = V_{SS}$ to $V_{CC}$ , $V_{CC} = V_{CC}$ Max	-1.0	—	+1.0	$\mu A$	
Output Leakage Current	$I_{LO}$	$V_{OUT} = V_{SS}$ to $V_{CC}$ , $V_{CC} = V_{CC}$ Max	-1.0	—	+1.0	$\mu A$	
$A_9$ , $\overline{OE}$ , $\overline{RESET}$ Inputs Leakage Current	$I_{LIT}$	$V_{CC} = V_{CC}$ Max, $A_9$ , $\overline{OE}$ , $\overline{RESET} = 12.5$ V	—	—	+35	$\mu A$	
$\overline{WP}/ACC$ Accelerated Program Current	$I_{LIA}$	$V_{CC} = V_{CC}$ Max, $\overline{WP}/ACC = V_{ACC}$ Max	—	—	20	mA	
$V_{CC}$ Active Current *1	$I_{CC1}$	$\overline{CE} = V_{IL}$ , $\overline{OE} = V_{IH}$ , $f = 5$ MHz	Byte	—	—	13	mA
			Word	—	—	15	
		$\overline{CE} = V_{IL}$ , $\overline{OE} = V_{IH}$ , $f = 1$ MHz	Byte	—	—	7	mA
			Word	—	—	7	
$V_{CC}$ Active Current *2	$I_{CC2}$	$\overline{CE} = V_{IL}$ , $\overline{OE} = V_{IH}$	—	—	35	mA	
$V_{CC}$ Current (Standby)	$I_{CC3}$	$V_{CC} = V_{CC}$ Max, $\overline{CE} = V_{CC} \pm 0.3$ V, $\overline{RESET} = V_{CC} \pm 0.3$ V, $\overline{WP}/ACC = V_{CC} \pm 0.3$ V	—	1	5	$\mu A$	
$V_{CC}$ Current (Standby, Reset)	$I_{CC4}$	$V_{CC} = V_{CC}$ Max, $\overline{RESET} = V_{SS} \pm 0.3$ V	—	1	5	$\mu A$	
$V_{CC}$ Current (Automatic Sleep Mode) *5	$I_{CC5}$	$V_{CC} = V_{CC}$ Max, $\overline{CE} = V_{SS} \pm 0.3$ V, $\overline{RESET} = V_{CC} \pm 0.3$ V, $V_{IN} = V_{CC} \pm 0.3$ V or $V_{SS} \pm 0.3$ V	—	1	5	$\mu A$	
$V_{CC}$ Active Current *6 (Read-While-Program)	$I_{CC6}$	$\overline{CE} = V_{IL}$ , $\overline{OE} = V_{IH}$	Byte	—	—	48	mA
			Word	—	—	50	
$V_{CC}$ Active Current *6 (Read-While-Erase)	$I_{CC7}$	$\overline{CE} = V_{IL}$ , $\overline{OE} = V_{IH}$	Byte	—	—	48	mA
			Word	—	—	50	
$V_{CC}$ Active Current (Erase-Suspend-Program)	$I_{CC8}$	$\overline{CE} = V_{IL}$ , $\overline{OE} = V_{IH}$	—	—	35	mA	
Input Low Voltage	$V_{IL}$	—	-0.5	—	+0.6	V	
Input High Voltage	$V_{IH}$	—	2.0	—	$V_{CC}+0.3$	V	
Voltage for Autoselect and Sector Group Protection ( $A_9$ , $\overline{OE}$ , $\overline{RESET}$ ) *3, *4	$V_{ID}$	—	11.5	12	12.5	V	
Voltage for $\overline{WP}/ACC$ Sector Group Protection/Unprotection and Program Acceleration *4	$V_{ACC}$	—	8.5	9.0	9.5	V	
Output Low Voltage	$V_{OL}$	$I_{OL} = 4.0$ mA, $V_{CC} = V_{CC}$ Min	—	—	0.45	V	
Output High Voltage	$V_{OH1}$	$I_{OH} = -2.0$ mA, $V_{CC} = V_{CC}$ Min	2.4	—	—	V	
	$V_{OH2}$	$I_{OH} = -100$ $\mu A$	$V_{CC} - 0.4$	—	—	V	
Low $V_{CC}$ Lock-Out Voltage	$V_{LKO}$	—	2.3	2.4	2.5	V	

(Continued)

# MBM29DL16XTE/BE70/90

*(Continued)*

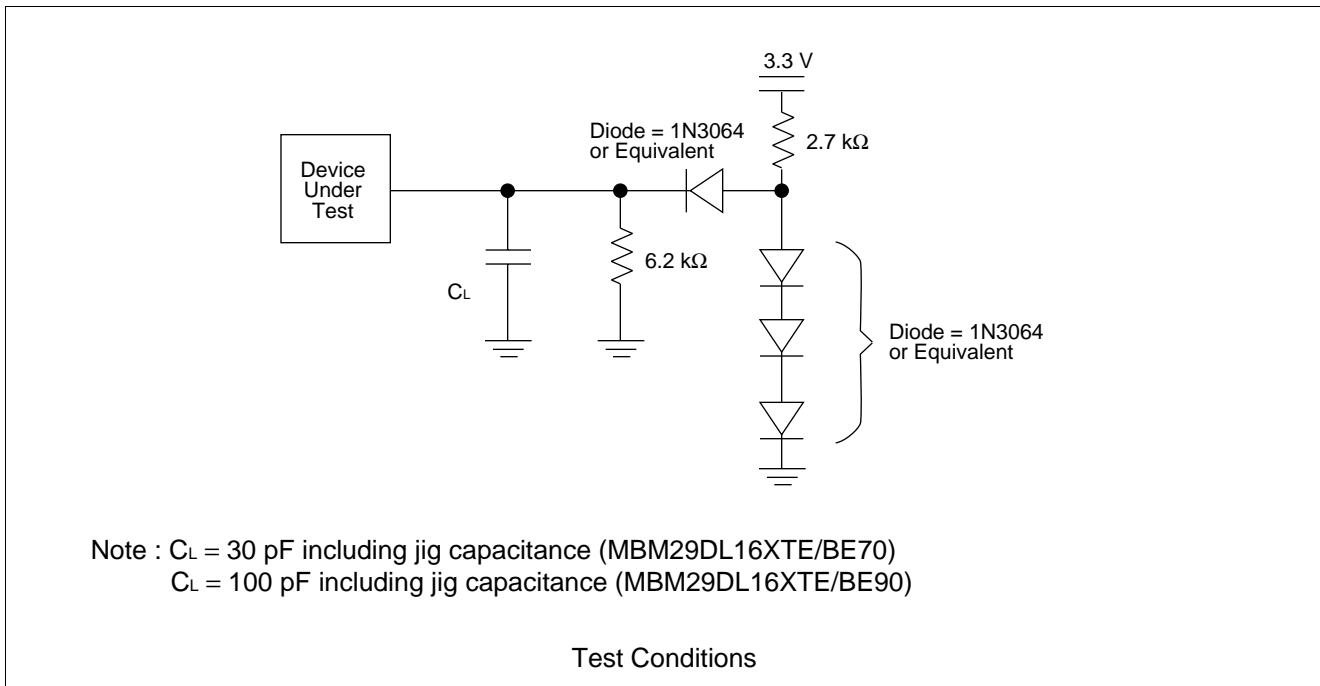
- \*1 : The  $I_{CC}$  current listed includes both the DC operating current and the frequency dependent component.
- \*2 :  $I_{CC}$  active while Embedded Algorithm (program or erase) is in progress.
- \*3 : This timing is only for Sector Group Protection operation and Autoselect mode.
- \*4 : Applicable for only  $V_{CC}$ .
- \*5 : Automatic sleep mode enables the low power mode when address remains stable for 150 ns.
- \*6 : Embedded Algorithm (program or erase) is in progress. (@5 MHz)

## 2. AC Characteristics

### • Read Only Operations Characteristics

Parameter	Symbol		Condi- tions	70		90		Unit
	JEDEC	Standard		Min	Max	Min	Max	
Read Cycle Time	t <sub>AVAV</sub>	t <sub>RC</sub>	—	70	—	90	—	ns
Address to Output Delay	t <sub>AVQV</sub>	t <sub>ACC</sub>	$\overline{CE} = V_{IL}$ $\overline{OE} = V_{IL}$	—	70	—	90	ns
Chip Enable to Output Delay	t <sub>ELQV</sub>	t <sub>CE</sub>	$\overline{OE} = V_{IL}$	—	70	—	90	ns
Output Enable to Output Delay	t <sub>GLQV</sub>	t <sub>OE</sub>	—	—	30	—	35	ns
Chip Enable to Output High-Z	t <sub>EHQZ</sub>	t <sub>DF</sub>	—	—	25	—	30	ns
Output Enable to Output High-Z	t <sub>GHQZ</sub>	t <sub>DF</sub>	—	—	25	—	30	ns
Output Hold Time From Addresses, $\overline{CE}$ or $\overline{OE}$ , Whichever Occurs First	t <sub>AXQX</sub>	t <sub>OH</sub>	—	0	—	0	—	ns
$\overline{RESET}$ Pin Low to Read Mode	—	t <sub>READY</sub>	—	—	20	—	20	μs
$\overline{CE}$ to $\overline{BYTE}$ Switching Low or High	—	t <sub>ELFL</sub> t <sub>ELFH</sub>	—	—	5	—	5	ns

Note: Test Conditions:  
 Output Load: 1 TTL gate and 30 pF (MBM29DL16XTE/BE70)  
                   1 TTL gate and 100 pF (MBM29DL16XTE/BE90)  
 Input rise and fall times: 5 ns  
 Input pulse levels: 0.0 V or 3.0 V  
 Timing measurement reference level  
     Input: 1.5 V  
     Output: 1.5 V



Note : C<sub>L</sub> = 30 pF including jig capacitance (MBM29DL16XTE/BE70)  
 C<sub>L</sub> = 100 pF including jig capacitance (MBM29DL16XTE/BE90)

# MBM29DL16XTE/BE70/90

## • Write/Erase/Program Operations

Parameter		Symbol		70			90			Unit
		JEDEC	Standard	Min	Typ	Max	Min	Typ	Max	
Write Cycle Time		t <sub>AVAV</sub>	t <sub>WC</sub>	70	—	—	90	—	—	ns
Address Setup Time		t <sub>AVWL</sub>	t <sub>AS</sub>	0	—	—	0	—	—	ns
Address Setup Time to $\overline{OE}$ Low During Toggle Bit Polling		—	t <sub>ASO</sub>	12	—	—	15	—	—	ns
Address Hold Time		t <sub>WLAX</sub>	t <sub>AH</sub>	45	—	—	45	—	—	ns
Address Hold Time from $\overline{CE}$ or $\overline{OE}$ High During Toggle Bit Polling		—	t <sub>AHT</sub>	0	—	—	0	—	—	ns
Data Setup Time		t <sub>DVWH</sub>	t <sub>DS</sub>	30	—	—	35	—	—	ns
Data Hold Time		t <sub>WHDX</sub>	t <sub>DH</sub>	0	—	—	0	—	—	ns
Output Enable Hold Time	Read	—	t <sub>OEH</sub>	0	—	—	0	—	—	ns
	Toggle and $\overline{Data}$ Polling			10	—	—	10	—	—	ns
$\overline{CE}$ High During Toggle Bit Polling		—	t <sub>CEPH</sub>	20	—	—	20	—	—	ns
$\overline{OE}$ High During Toggle Bit Polling		—	t <sub>OEPH</sub>	20	—	—	20	—	—	ns
Read Recover Time Before Write		t <sub>GHWL</sub>	t <sub>GHWL</sub>	0	—	—	0	—	—	ns
Read Recover Time Before Write		t <sub>GHEL</sub>	t <sub>GHEL</sub>	0	—	—	0	—	—	ns
$\overline{CE}$ Setup Time		t <sub>ELWL</sub>	t <sub>CS</sub>	0	—	—	0	—	—	ns
$\overline{WE}$ Setup Time		t <sub>WLEL</sub>	t <sub>WS</sub>	0	—	—	0	—	—	ns
$\overline{CE}$ Hold Time		t <sub>WHEH</sub>	t <sub>CH</sub>	0	—	—	0	—	—	ns
$\overline{WE}$ Hold Time		t <sub>EHWH</sub>	t <sub>WH</sub>	0	—	—	0	—	—	ns
Write Pulse Width		t <sub>WLWH</sub>	t <sub>WP</sub>	35	—	—	35	—	—	ns
$\overline{CE}$ Pulse Width		t <sub>ELEH</sub>	t <sub>CP</sub>	35	—	—	35	—	—	ns
Write Pulse Width High		t <sub>WHWL</sub>	t <sub>WPH</sub>	25	—	—	30	—	—	ns
$\overline{CE}$ Pulse Width High		t <sub>EHEL</sub>	t <sub>CPH</sub>	25	—	—	30	—	—	ns
Programming Operation	Byte	t <sub>WHWH1</sub>	t <sub>WHWH1</sub>	—	8	—	—	8	—	μs
	Word			—	16	—	—	16	—	μs
Sector Erase Operation*1		t <sub>WHWH2</sub>	t <sub>WHWH2</sub>	—	1	—	—	1	—	s
V <sub>CC</sub> Setup Time		—	t <sub>VCS</sub>	50	—	—	50	—	—	μs
Rise Time to V <sub>ID</sub> *2		—	t <sub>VIDR</sub>	500	—	—	500	—	—	ns
Rise Time to V <sub>ACC</sub> *3		—	t <sub>VACCR</sub>	500	—	—	500	—	—	ns
Voltage Transition Time*2		—	t <sub>VLHT</sub>	4	—	—	4	—	—	μs
Write Pulse Width*2		—	t <sub>WPP</sub>	100	—	—	100	—	—	μs
$\overline{OE}$ Setup Time to $\overline{WE}$ Active*2		—	t <sub>OESP</sub>	4	—	—	4	—	—	μs
$\overline{CE}$ Setup Time to $\overline{WE}$ Active *2		—	t <sub>CSP</sub>	4	—	—	4	—	—	μs
Recover Time From RY/BY		—	t <sub>RB</sub>	0	—	—	0	—	—	ns

(Continued)



(Continued)

Parameter	Symbol		70			90			Unit
	JEDEC	Standard	Min	Typ	Max	Min	Typ	Max	
$\overline{\text{RESET}}$ Pulse Width	—	t <sub>RP</sub>	500	—	—	500	—	—	ns
$\overline{\text{RESET}}$ High Level Period Before Read	—	t <sub>RH</sub>	200	—	—	200	—	—	ns
$\overline{\text{BYTE}}$ Switching Low to Output High-Z	—	t <sub>FLOZ</sub>	—	—	25	—	—	30	ns
$\overline{\text{BYTE}}$ Switching High to Output Active	—	t <sub>FHQV</sub>	—	—	70	—	—	90	ns
Program/Erase Valid to RY/ $\overline{\text{BY}}$ Delay	—	t <sub>BUSY</sub>	—	—	90	—	—	90	ns
Delay Time from Embedded Output Enable	—	t <sub>EOE</sub>	—	—	70	—	—	90	ns
Erase Time-out Time	—	t <sub>TOW</sub>	—	—	50	—	—	50	μs
Erase Suspend Transition Time	—	t <sub>SPD</sub>	—	—	20	—	—	20	μs

\*1 : This does not include preprogramming time.

\*2 : This timing is for Sector Group Protection operation.

\*3 : This timing is limited for Accelerated Protection operation.

## ■ ERASE AND PROGRAMMING PERFORMANCE

Parameter	Value			Unit	Comments
	Min	Typ	Max		
Sector Erase Time	—	1	10	s	Excludes programming time prior to erasure
Word Programming Time	—	16	360	μs	Excludes system-level overhead
Byte Programming Time	—	8	300	μs	
Chip Programming Time	—	—	50	s	Excludes system-level overhead
Program/Erase Cycle	100,000	—	—	cycle	

# MBM29DL16XTE/BE70/90

## ■ PIN CAPACITANCE

### 1. TSOP(1) pin capacitance

Parameter	Symbol	Condition	Value			Unit
			Min	Typ	Max	
Input Capacitance	C <sub>IN</sub>	V <sub>IN</sub> = 0	—	6.0	7.5	pF
Output Capacitance	C <sub>OUT</sub>	V <sub>OUT</sub> = 0	—	8.5	12.0	pF
Control Pin Capacitance	C <sub>IN2</sub>	V <sub>IN</sub> = 0	—	8.0	10.0	pF
$\overline{WP}/ACC$ Pin Capacitance	C <sub>IN3</sub>	V <sub>IN</sub> = 0	—	17.0	18.0	pF

Notes : • Test conditions T<sub>A</sub> = +25°C, f = 1.0 MHz  
• DQ<sub>15/A-1</sub> pin capacitance is stipulated by output capacitance.




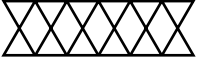
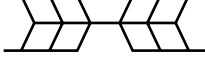
### 2. FBGA pin capacitance

Parameter	Symbol	Condition	Value			Unit
			Min	Typ	Max	
Input Capacitance	C <sub>IN</sub>	V <sub>IN</sub> = 0	—	7.0	9.0	pF
Output Capacitance	C <sub>OUT</sub>	V <sub>OUT</sub> = 0	—	9.5	13.0	pF
Control Pin Capacitance	C <sub>IN2</sub>	V <sub>IN</sub> = 0	—	9.0	11.0	pF
$\overline{WP}/ACC$ Pin Capacitance	C <sub>IN3</sub>	V <sub>IN</sub> = 0	—	17.0	18.0	pF

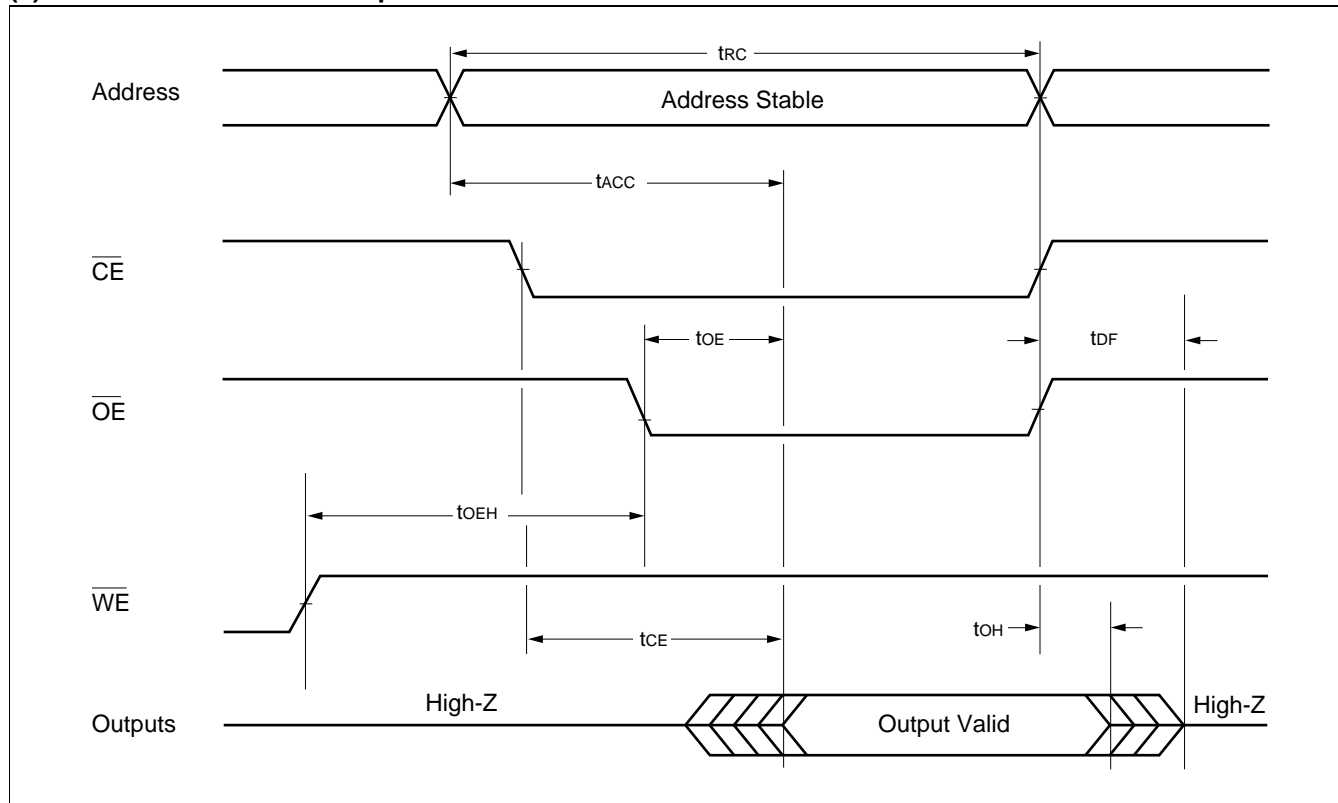
Notes : • Test conditions T<sub>A</sub> = +25°C, f = 1.0 MHz  
• DQ<sub>15/A-1</sub> pin capacitance is stipulated by output capacitance.

## ■ TIMING DIAGRAM

### • Key to Switching Waveforms

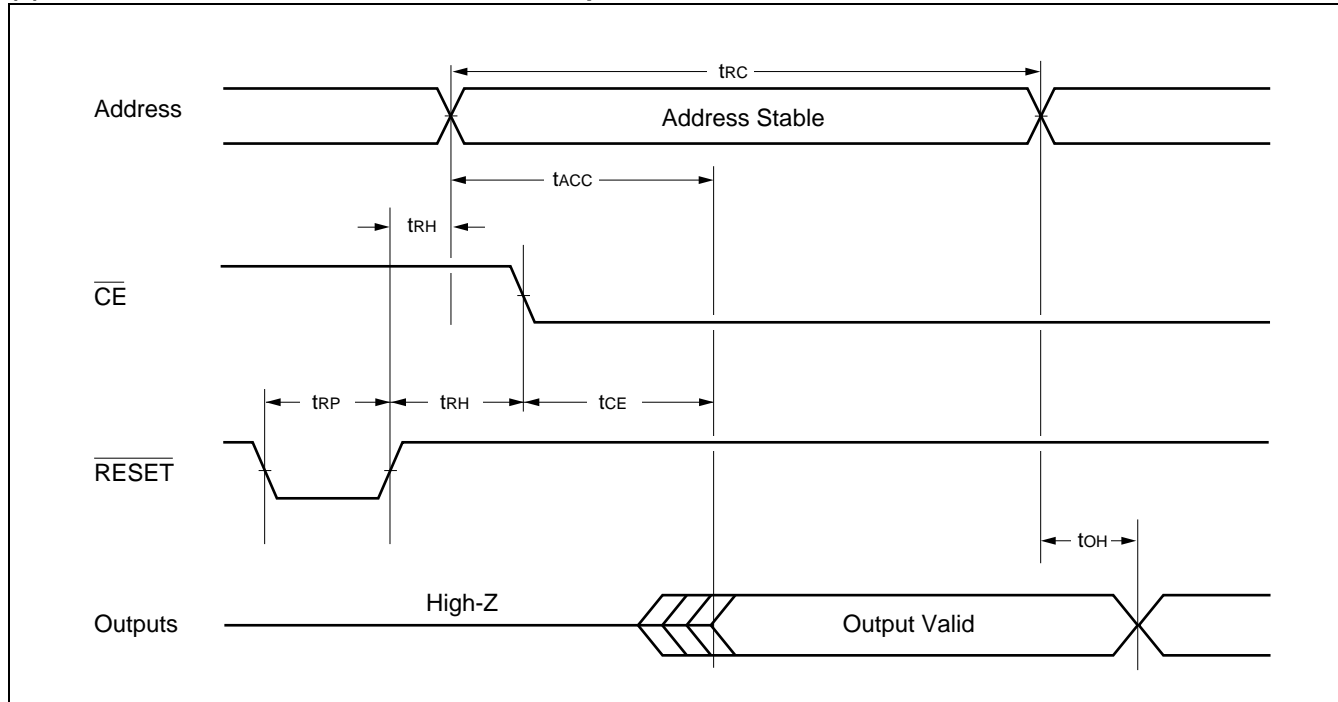
WAVEFORM	INPUTS	OUTPUTS
	Must Be Steady	Will Be Steady
	May Change from H to L	Will Be Changing from H to L
	May Change from L to H	Will Be Changing from L to H
	"H" or "L" Any Change Permitted	Changing State Unknown
	Does Not Apply	Center Line is High-Impedance "Off" State

### (1) AC Waveforms for Read Operations

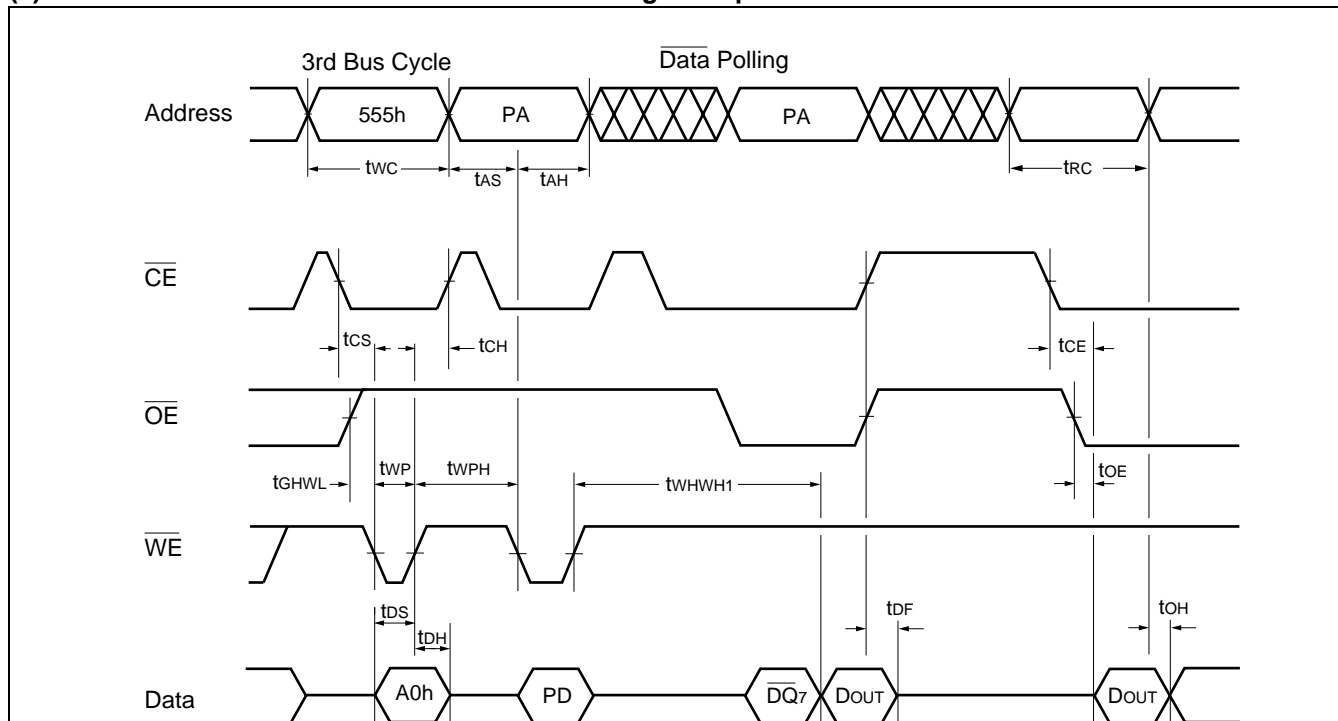


# MBM29DL16XTE/BE70/90

## (2) AC Waveforms for Hardware Reset/Read Operations

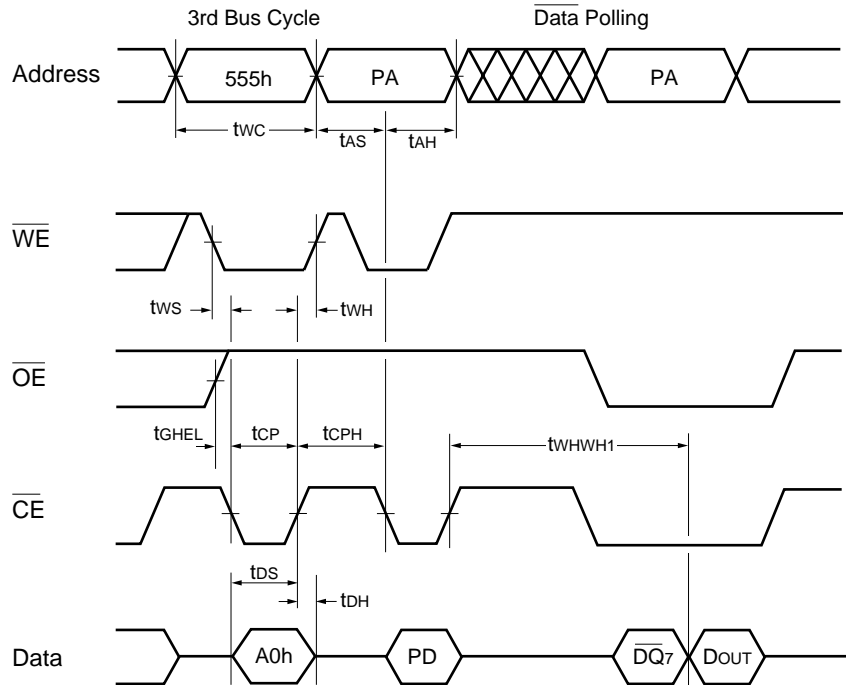


## (3) AC Waveforms for Alternate $\overline{WE}$ Controlled Program Operations



- Notes :
- PA is address of the memory location to be programmed.
  - PD is data to be programmed at byte address.
  - $\overline{DQ}_7$  is the output of the complement of the data written to the device.
  - $D_{OUT}$  is the output of the data written to the device.
  - Figure indicates the last two bus cycles out of four bus cycle sequence.
  - These waveforms are for the  $\times 16$  mode. These address differ from  $\times 8$  mode.

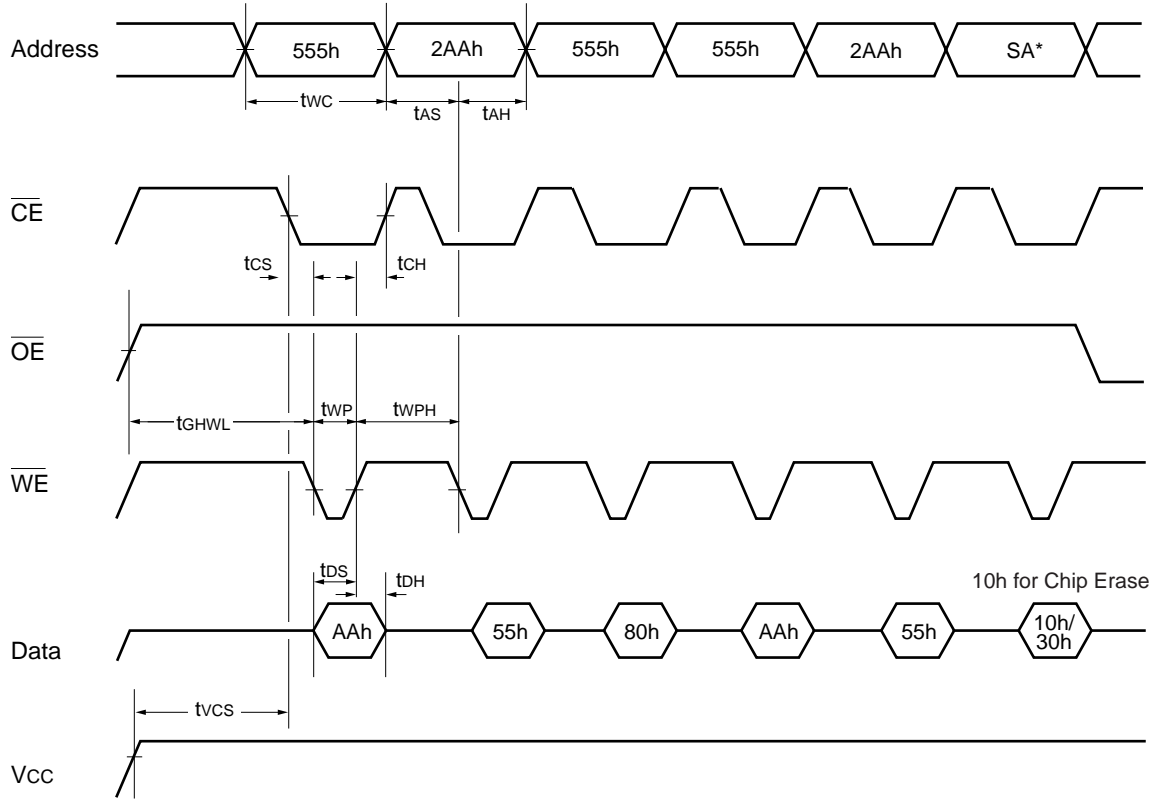
## (4) AC Waveforms for Alternate $\overline{\text{CE}}$ Controlled Program Operations



- Notes :
- PA is address of the memory location to be programmed.
  - PD is data to be programmed at byte address.
  - $\overline{\text{DQ}}_7$  is the output of the complement of the data written to the device.
  - $\text{D}_{\text{OUT}}$  is the output of the data written to the device.
  - Figure indicates the last two bus cycles out of four bus cycle sequence.
  - These waveforms are for the  $\times 16$  mode. These address differ from  $\times 8$  mode.

# MBM29DL16XTE/BE70/90

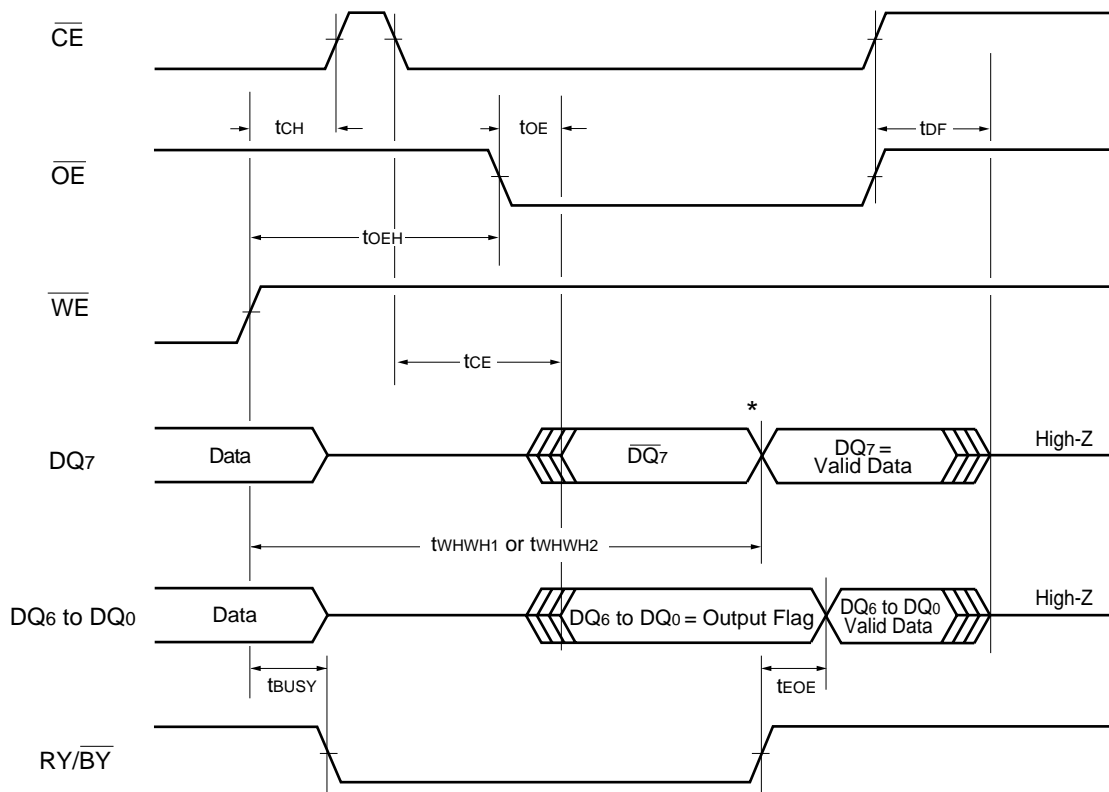
## (5) AC Waveforms for Chip/Sector Erase Operations



\* : SA is the sector address for Sector Erase. Addresses = 555h (Word) for Chip Erase.

Note : These waveforms are for the  $\times 16$  mode. The addresses differ from  $\times 8$  mode.

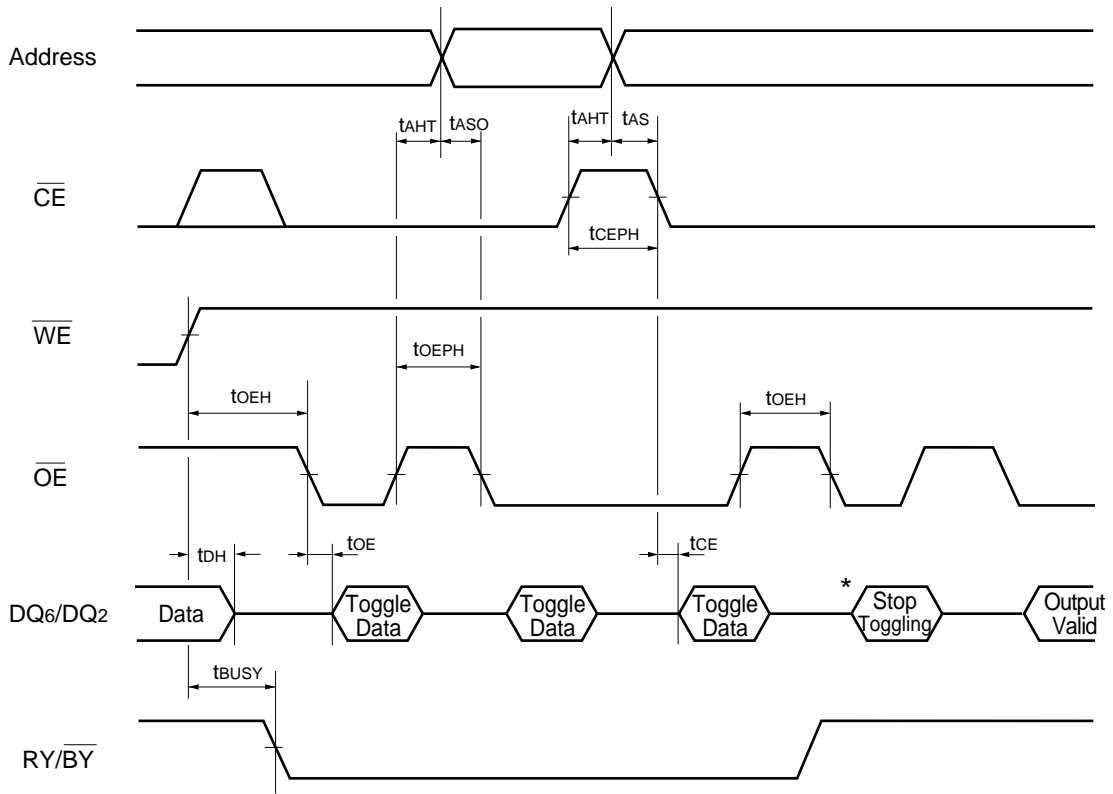
## (6) AC Waveforms for $\overline{\text{Data}}$ Polling during Embedded Algorithm Operations



\* :  $\text{DQ7} = \text{Valid Data}$  (The device has completed the Embedded operation).

# MBM29DL16XTE/BE70/90

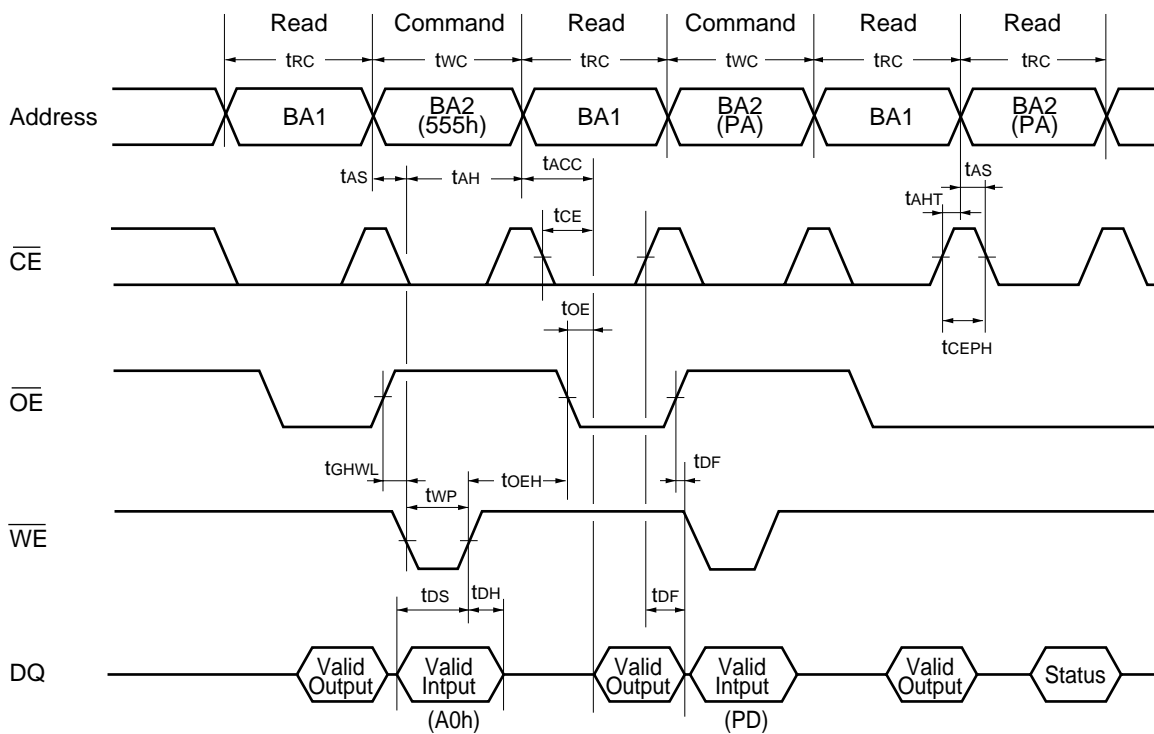
## (7) AC Waveforms for Toggle Bit I during Embedded Algorithm Operations



\* : DQ<sub>6</sub> stops toggling (The device has completed the Embedded operation).

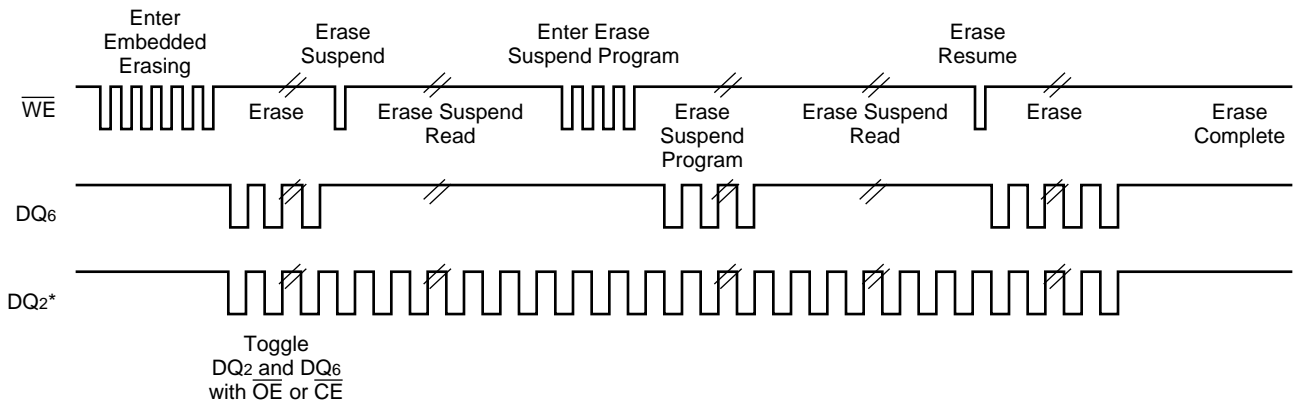


## (8) Bank-to-bank Read/Write Timing Diagram



Note: This is the example of Read for Bank 1 and Embedded Algorithm (program) for Bank 2.  
 BA1: Address corresponding to Bank 1.  
 BA2: Address corresponding to Bank 2.

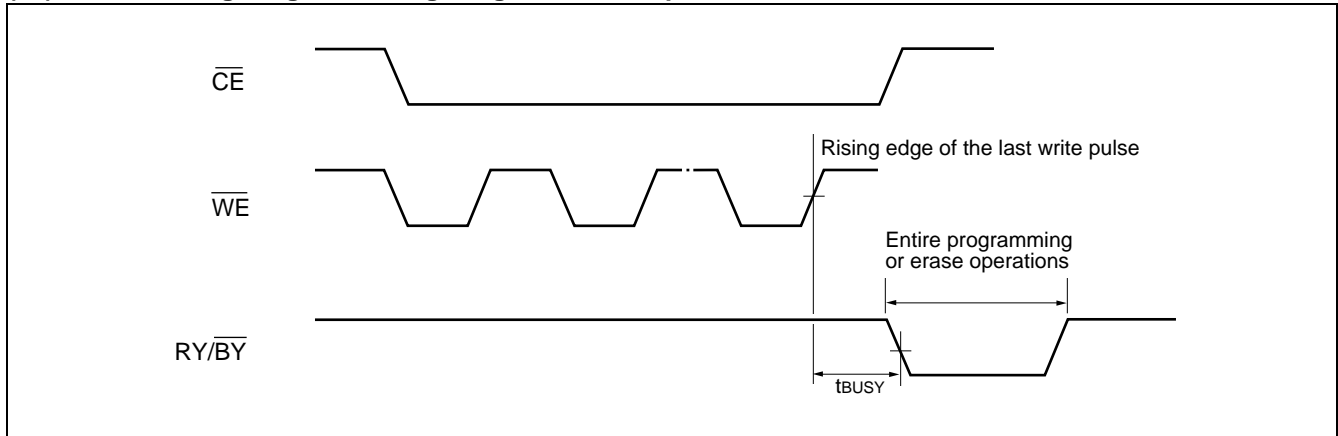
## (9) DQ<sub>2</sub> vs. DQ<sub>6</sub>



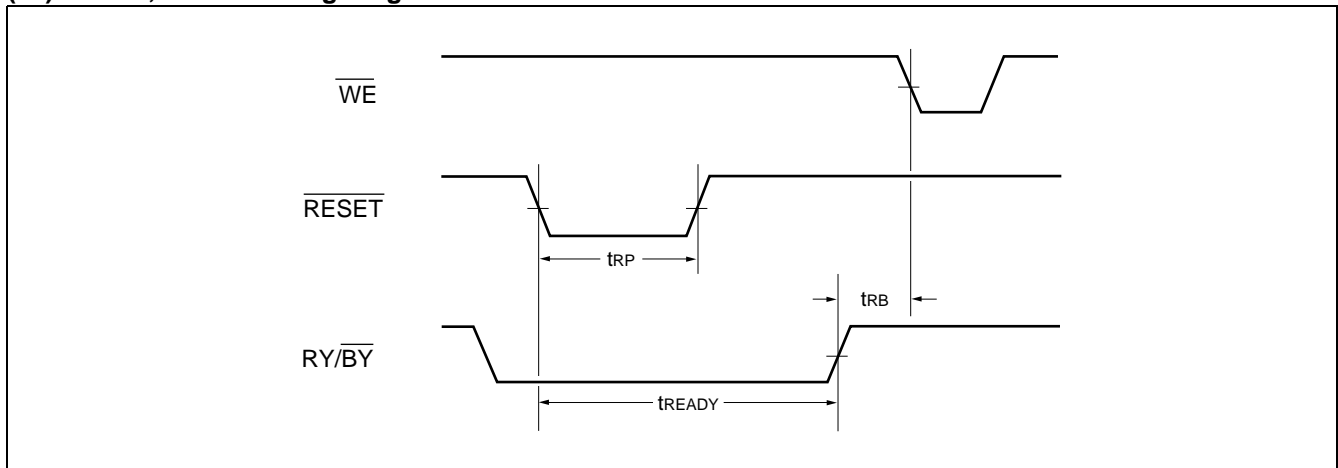
\* : DQ<sub>2</sub> is read from the erase-suspended sector.

# MBM29DL16XTE/BE70/90

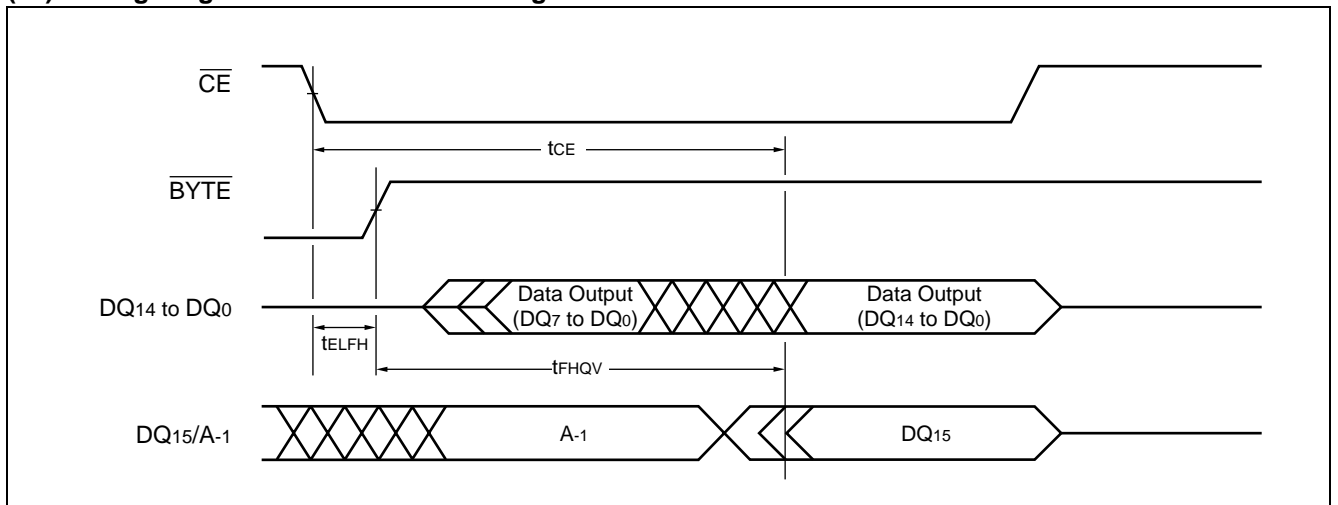
(10) RY/ $\overline{\text{BY}}$  Timing Diagram during Program/Erase Operations



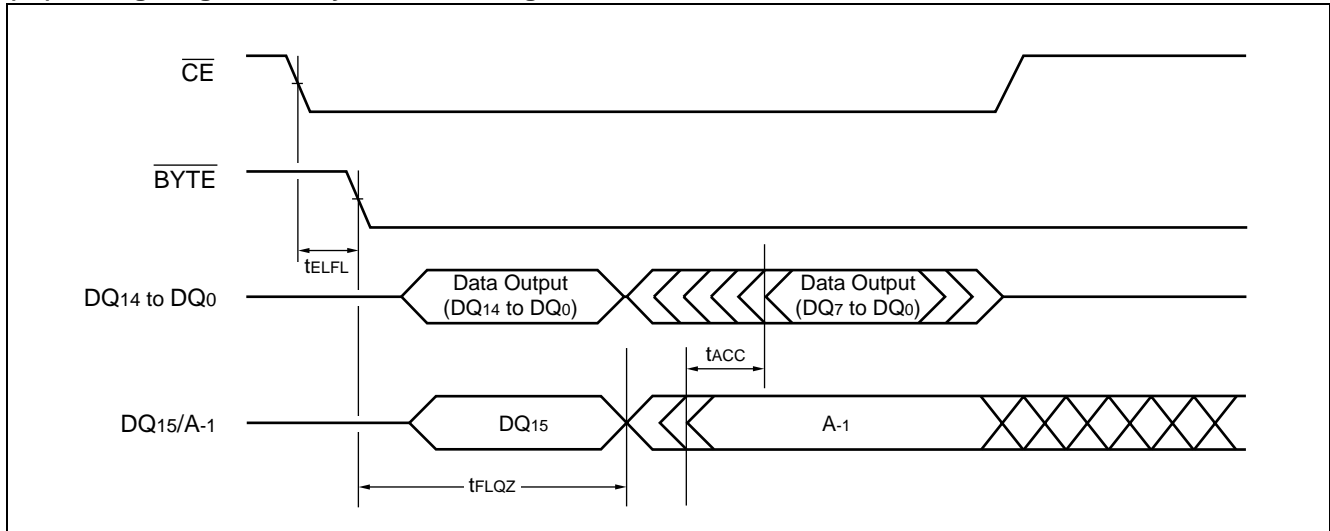
(11)  $\overline{\text{RESET}}$ , RY/ $\overline{\text{BY}}$  Timing Diagram



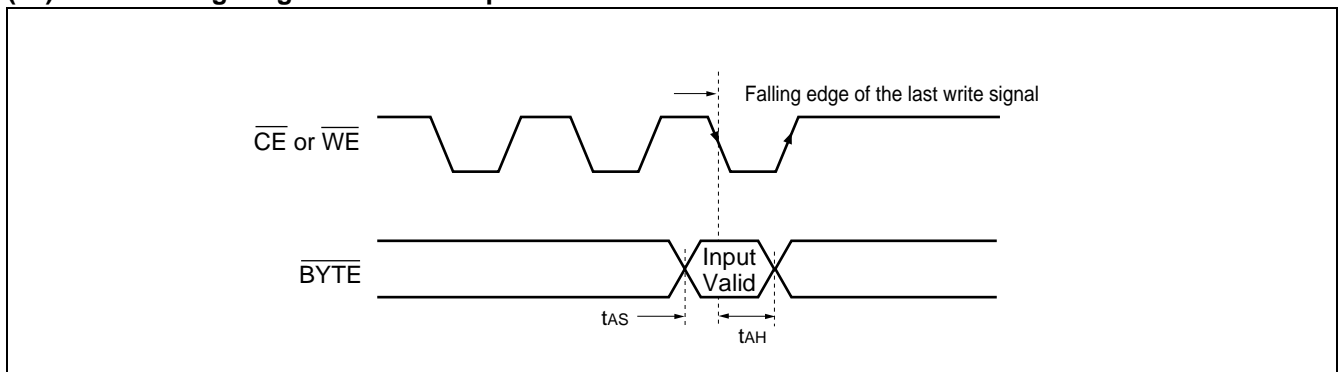
(12) Timing Diagram for Word Mode Configuration



**(13) Timing Diagram for Byte Mode Configuration**

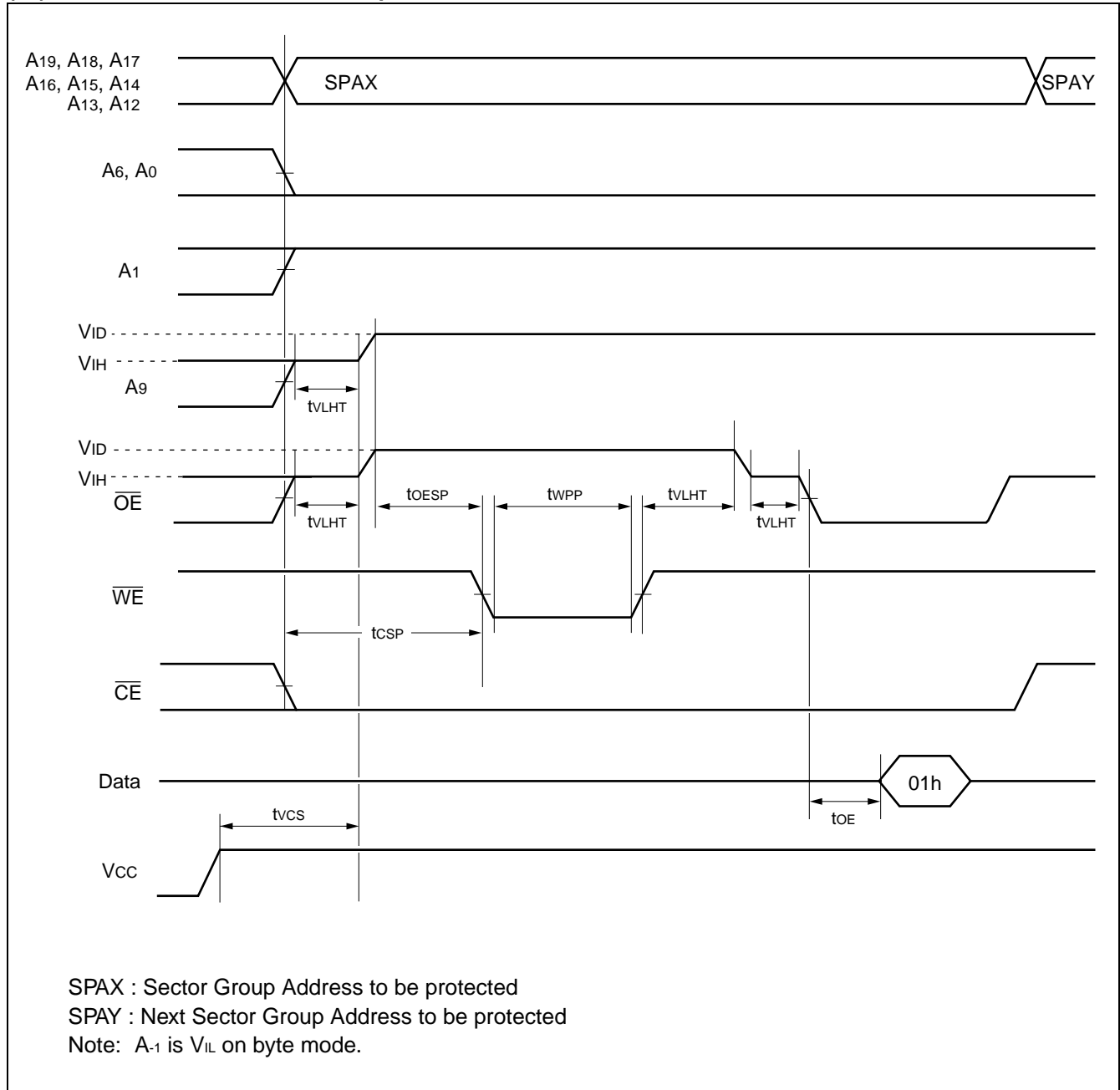


**(14)  $\overline{BYTE}$  Timing Diagram for Write Operations**

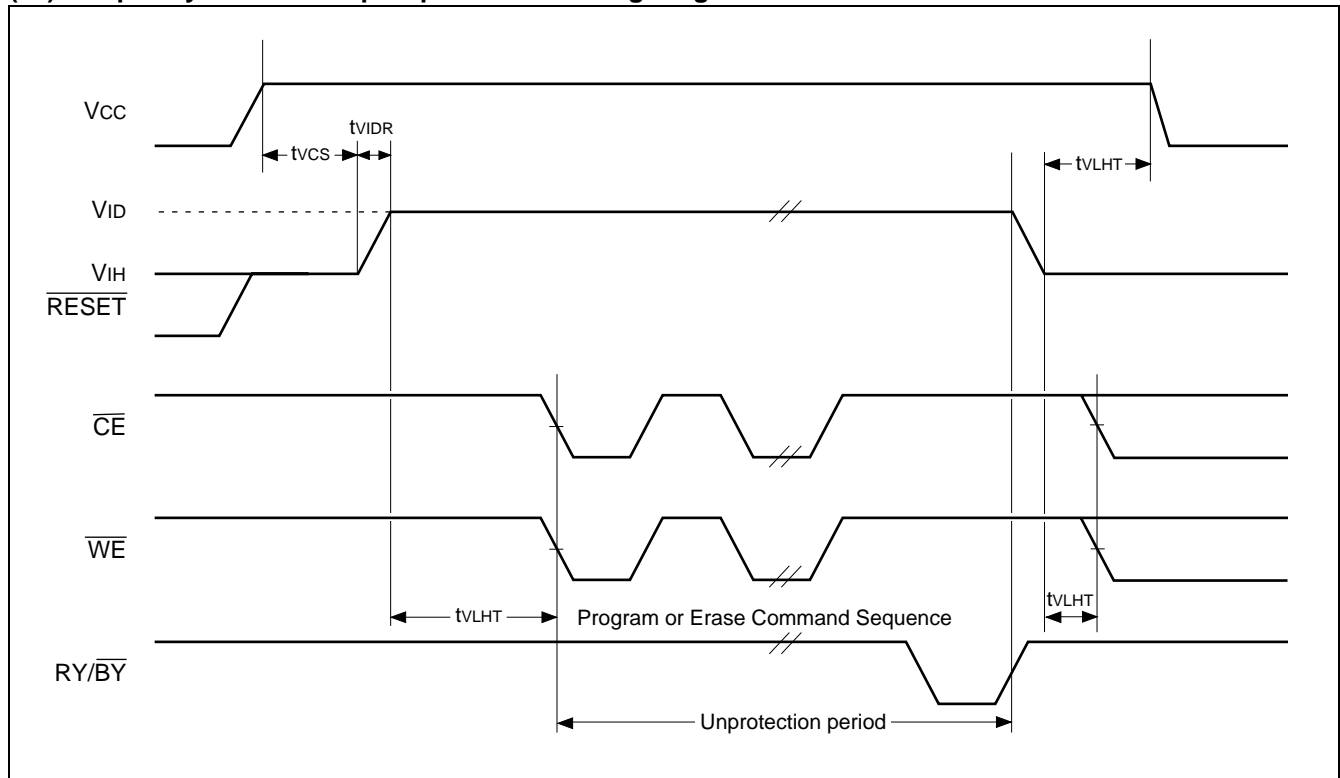


# MBM29DL16XTE/BE70/90

## (15) AC Waveforms for Sector Group Protection

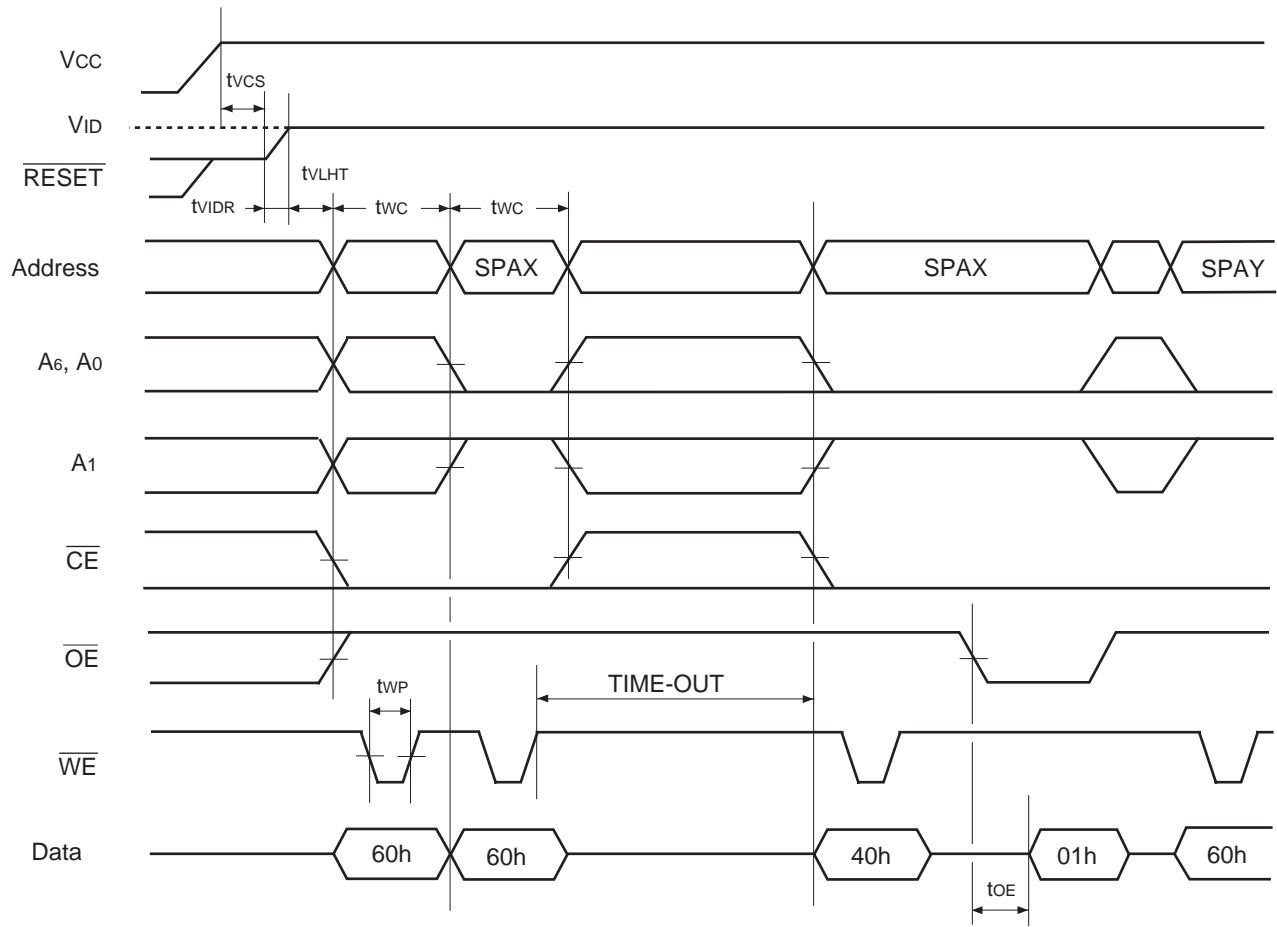


(16) Temporary Sector Group Unprotection Timing Diagram



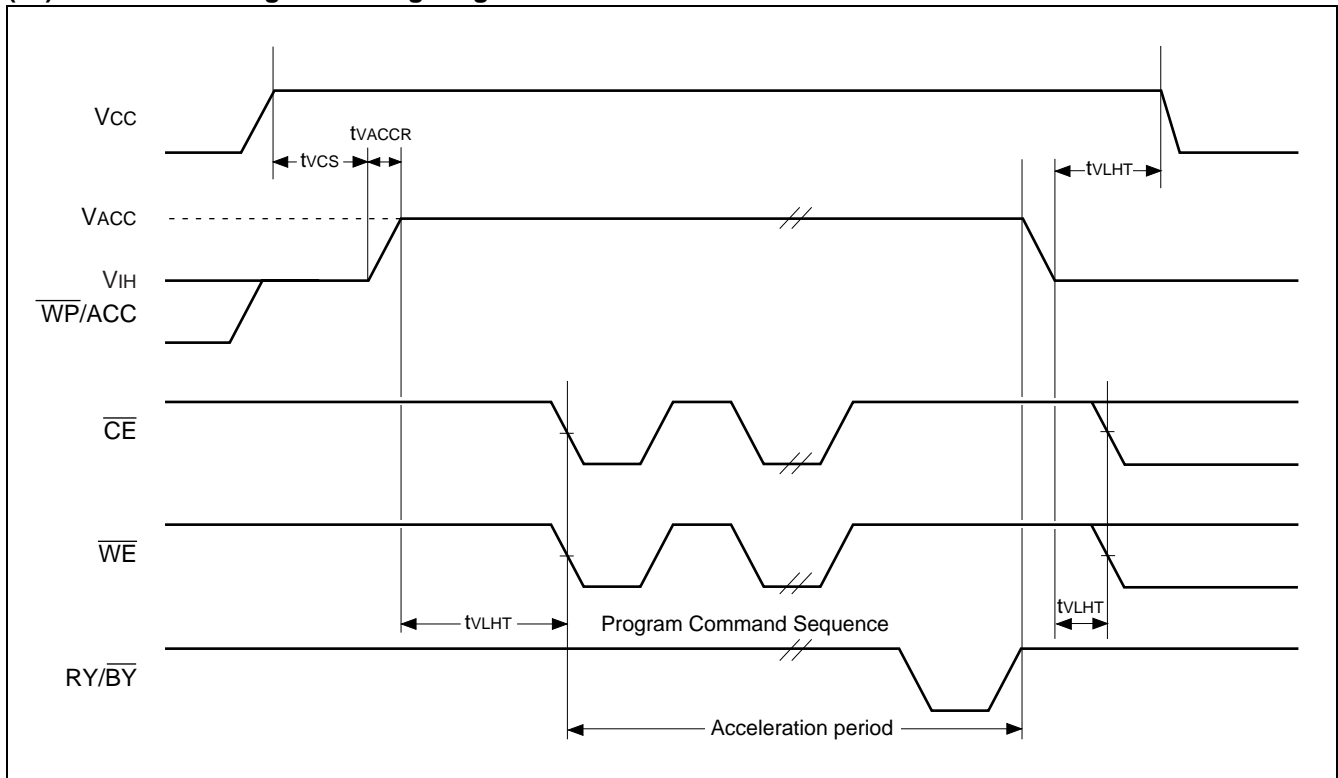
# MBM29DL16XTE/BE70/90

(17) Extended Sector Group Protection Timing Diagram



SPAX : Sector Group Address to be protected  
 SPAY : Next Sector Group Address to be protected  
 TIME-OUT : Time-Out window = 250  $\mu$ s (Min)  
 Note : A-1 is  $V_{IL}$  on byte mode.

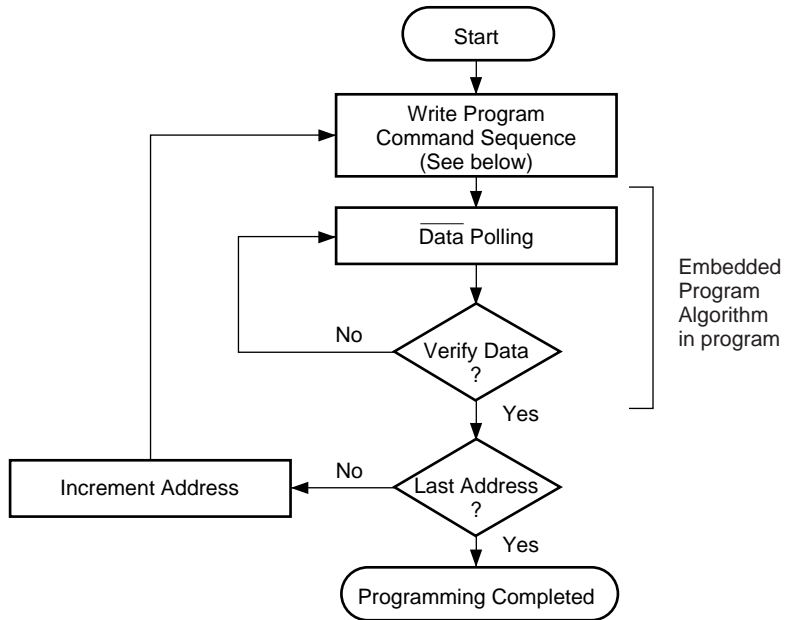
**(18) Accelerated Program Timing Diagram**



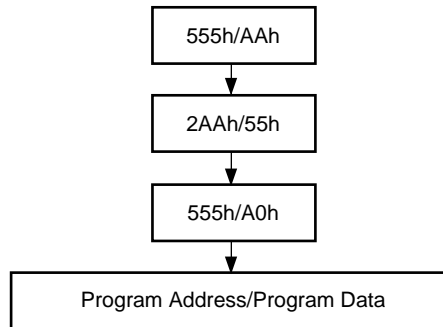
## ■ FLOW CHART

### (1) Embedded Program™ Algorithm

#### EMBEDDED ALGORITHMS



#### Program Command Sequence (Address/Command):

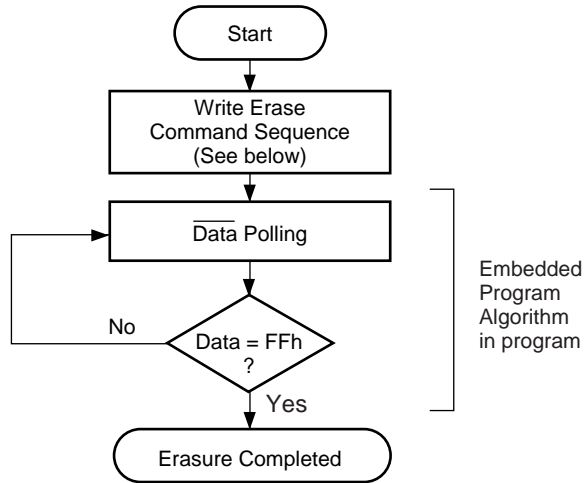


- Notes :
- The sequence is applied for  $\times 16$  mode.
  - The addresses differ from  $\times 8$  mode.

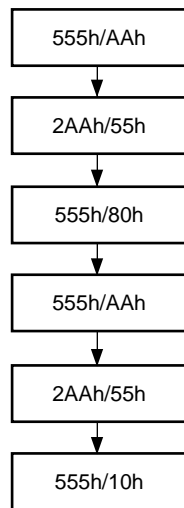


## (2) Embedded Erase™ Algorithm

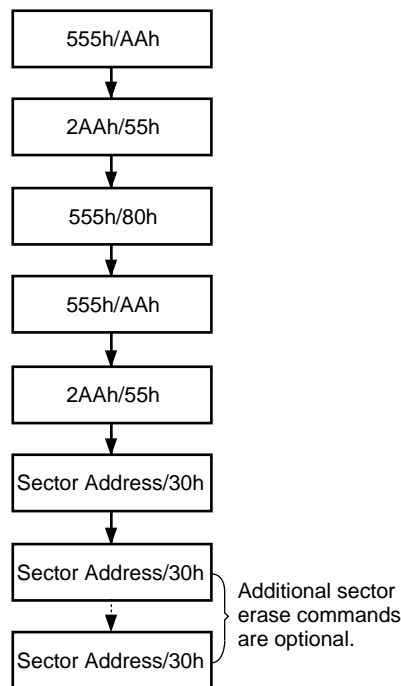
### EMBEDDED ALGORITHMS



#### Chip Erase Command Sequence (Address/Command):

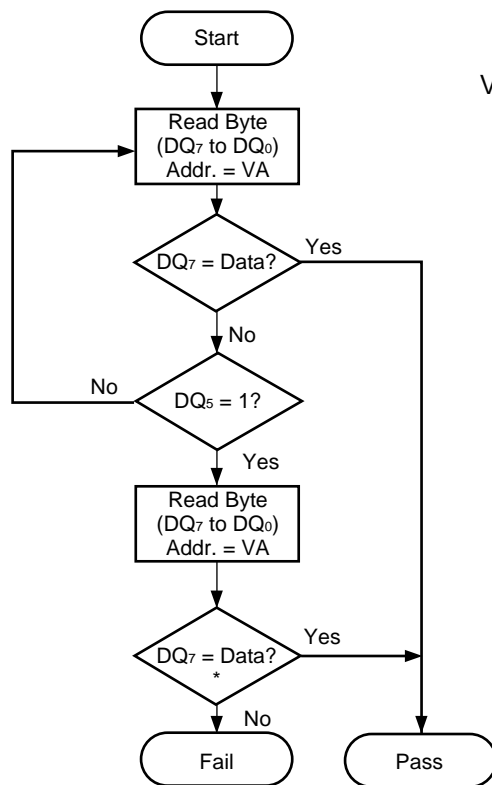


#### Individual Sector/Multiple Sector Erase Command Sequence (Address/Command):



- Notes :
- The sequence is applied for × 16 mode.
  - The addresses differ from × 8 mode.

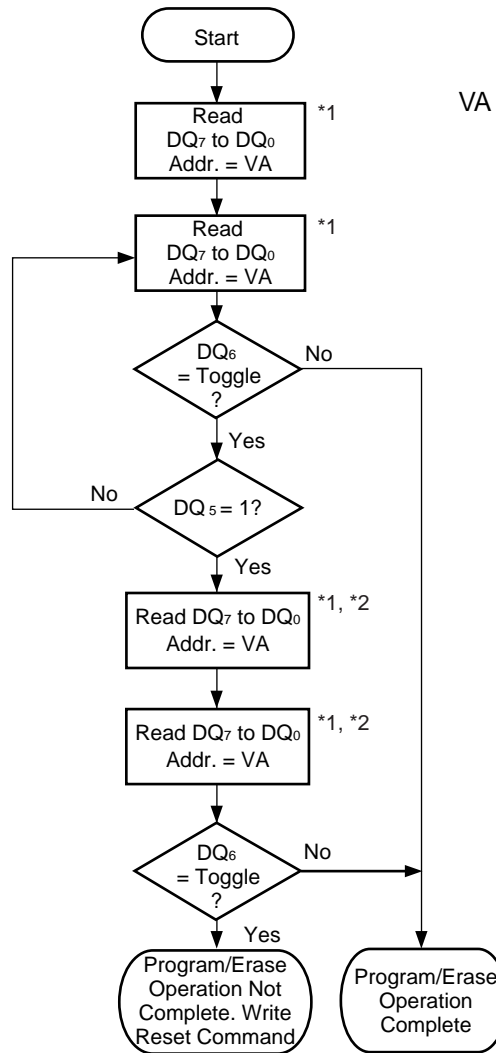
## (3) Data Polling Algorithm



VA = Address for programming  
= Any of the sector addresses within the sector being erased during sector erase or multiple sector erases operation.  
= Any of the sector addresses within the sector not being protected during chip erase operation.

\* : DQ<sub>7</sub> is rechecked even if DQ<sub>5</sub> = "1" because DQ<sub>7</sub> may change simultaneously with DQ<sub>5</sub>.

## (4) Toggle Bit Algorithm

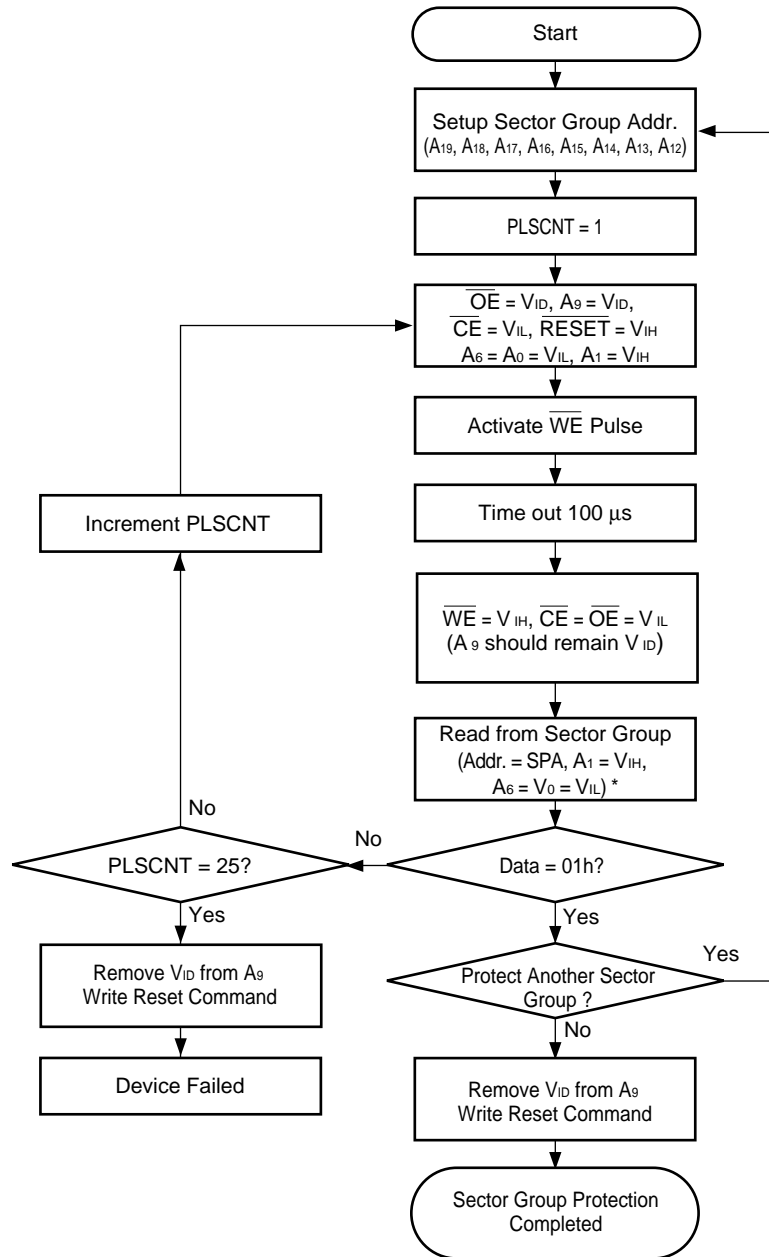


VA = Bank address being executed Embedded Algorithm.

\*1 : Read toggle bit twice to determine whether it is toggling.

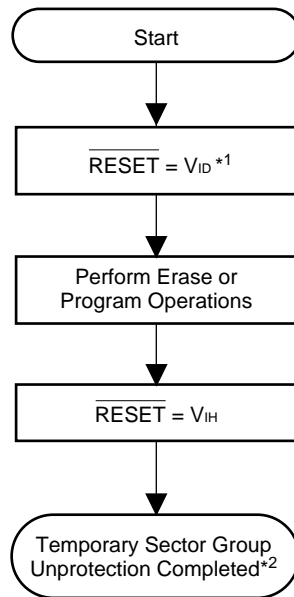
\*2 : Recheck toggle bit because it may stop toggling as DQ<sub>5</sub> changing to "1".

## (5) Sector Group Protection Algorithm



\* : A-1 is V<sub>IL</sub> on byte mode.

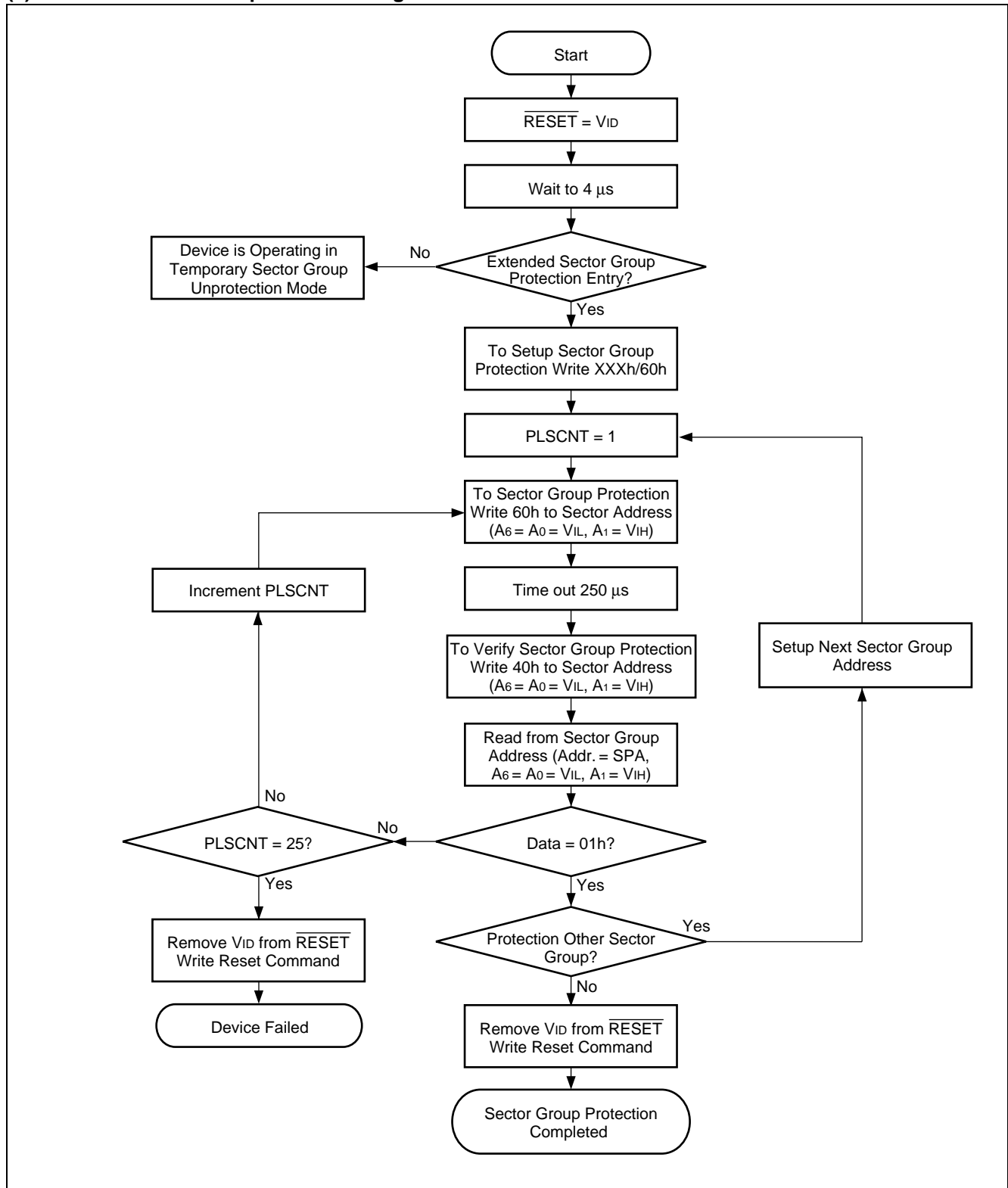
## (6) Temporary Sector Group Unprotection Algorithm



\*1 : All protected sector groups are unprotected.

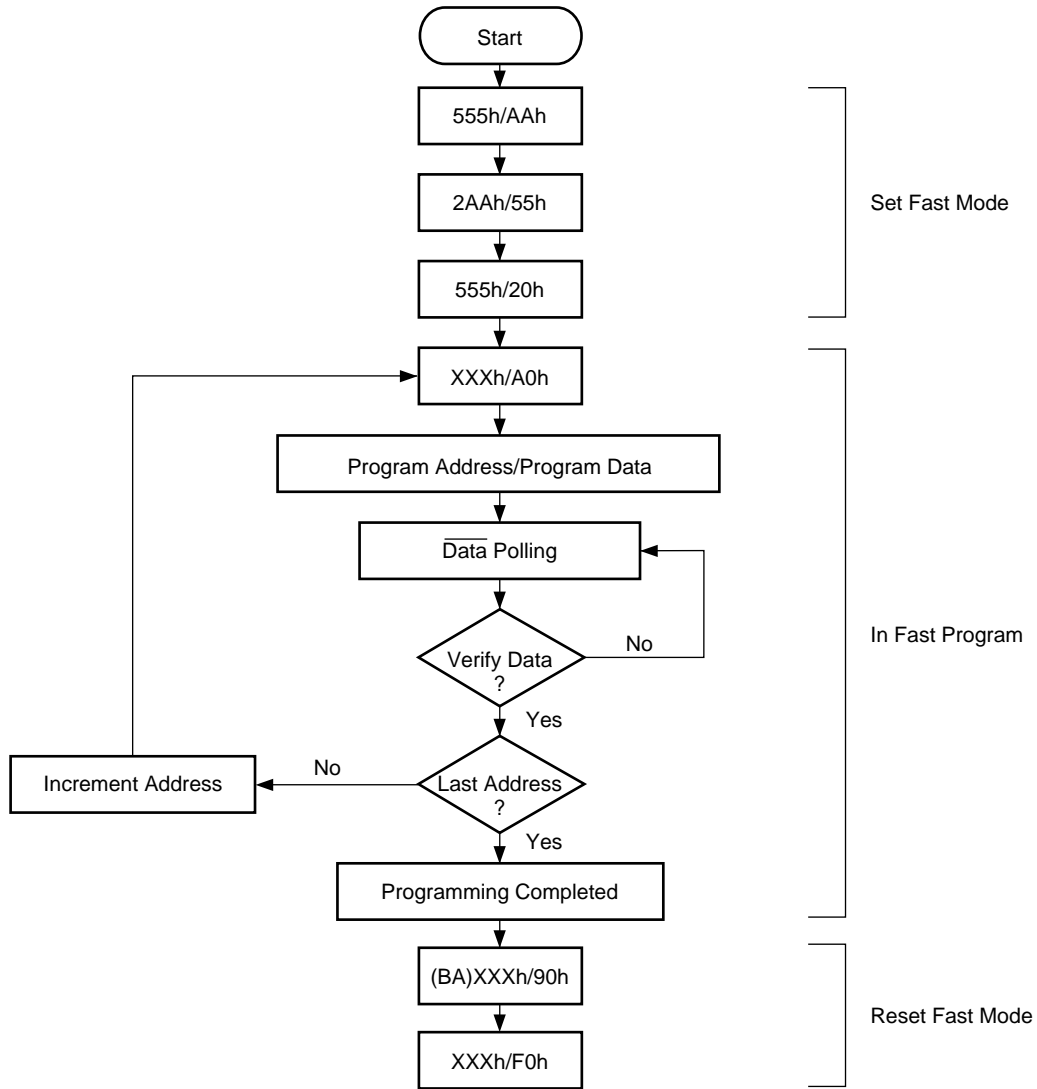
\*2 : All previously protected sector groups are protected once again.

## (7) Extended Sector Group Protection Algorithm



## (8) Embedded Program™ Algorithm for Fast Mode

### FAST MODE ALGORITHM



- Notes:
- The sequence is applied for  $\times 16$  mode.
  - The addresses differ from  $\times 8$  mode.

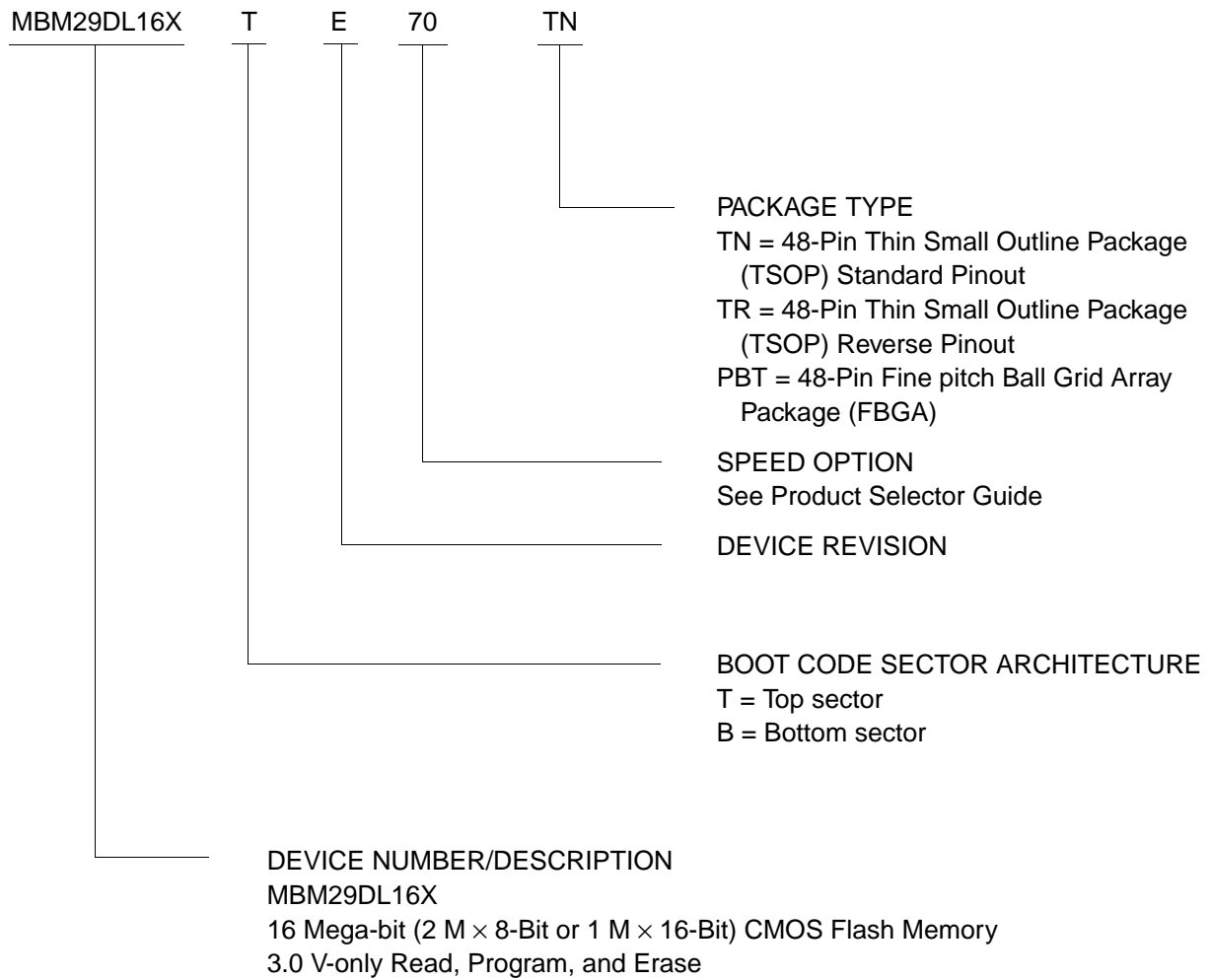
# MBM29DL16XTE/BE70/90

## ■ ORDERING INFORMATION

Part No.	Package	Access Tome	Remarks	
MBM29DL161TE-70TN	48-pin plastic TSOP (1) (FPT-48P-M19) Normal Bend	70	Top Sector	
MBM29DL161TE-90TN		90		
MBM29DL162TE-70TN		70		
MBM29DL162TE-90TN		90		
MBM29DL163TE-70TN		70		
MBM29DL163TE-90TN		90		
MBM29DL164TE-70TN		70		
MBM29DL164TE-90TN		90		
MBM29DL161TE-70TR		48-pin plastic TSOP (1) (FPT-48P-M20) Reverse Bend		70
MBM29DL161TE-90TR				90
MBM29DL162TE-70TR	70			
MBM29DL162TE-90TR	90			
MBM29DL163TE-70TR	70			
MBM29DL163TE-90TR	90			
MBM29DL164TE-70TR	70			
MBM29DL164TE-90TR	90			
MBM29DL161TE-70PBT	48-pin plastic FBGA (BGA-48P-M11)	70		
MBM29DL161TE-90PBT		90		
MBM29DL162TE-70PBT		70		
MBM29DL162TE-90PBT		90		
MBM29DL163TE-70PBT		70		
MBM29DL163TE-90PBT		90		
MBM29DL164TE-70PBT	70			
MBM29DL164TE-90PBT	90			
MBM29DL161BE-70TN	48-pin plastic TSOP (1) (FPT-48P-M19) Normal Bend	70	Bottom Sector	
MBM29DL161BE-90TN		90		
MBM29DL162BE-70TN		70		
MBM29DL162BE-90TN		90		
MBM29DL163BE-70TN		70		
MBM29DL163BE-90TN		90		
MBM29DL164BE-70TN	70			
MBM29DL164BE-90TN	90			
MBM29DL161BE-70TR	48-pin plastic TSOP (1) (FPT-48P-M20) Reverse Bend	70		
MBM29DL161BE-90TR		90		
MBM29DL162BE-70TR		70		
MBM29DL162BE-90TR		90		
MBM29DL163BE-70TR		70		
MBM29DL163BE-90TR		90		
MBM29DL164BE-70TR	70			
MBM29DL164BE-90TR	90			
MBM29DL161BE-70PBT	48-pin plastic FBGA (BGA-48P-M11)	70		
MBM29DL161BE-90PBT		90		
MBM29DL162BE-70PBT		70		
MBM29DL162BE-90PBT		90		
MBM29DL163BE-70PBT		70		
MBM29DL163BE-90PBT		90		
MBM29DL164BE-70PBT	70			
MBM29DL164BE-90PBT	90			



# MBM29DL16XTE/BE70/90



# MBM29DL16XTE/BE70/90

## PACKAGE DIMENSIONS

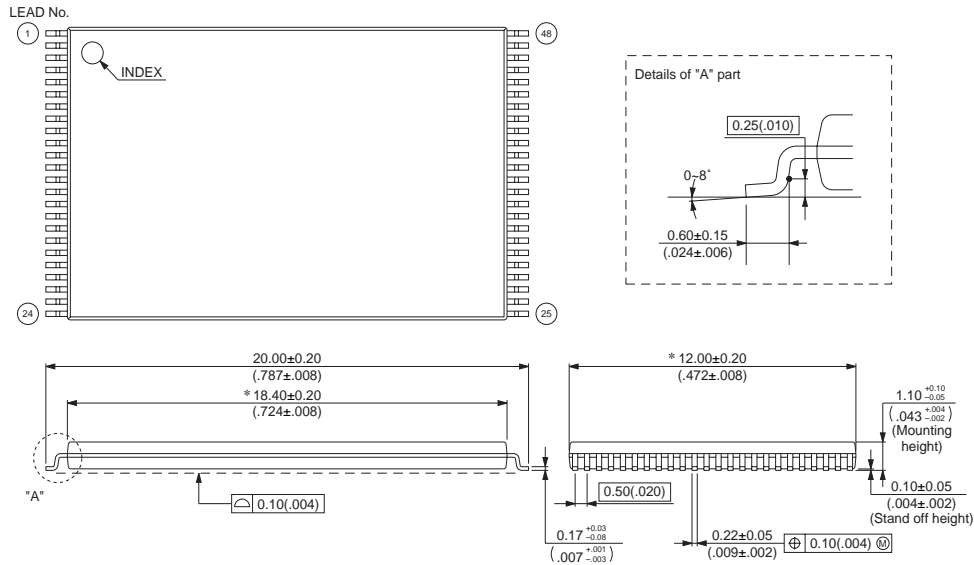
48-pin plastic TSOP(1)  
(FPT-48P-M19)

Note 1) \* : Values do not include resin protrusion.

Resin protrusion and gate protrusion are +0.15 (.006) Max (each side).

Note 2) Pins width and pins thickness include plating thickness.

Note 3) Pins width do not include tie bar cutting remainder.



© 2003 FUJITSU LIMITED F48029S-c-6-7

Dimensions in mm (inches).

Note : The values in parentheses are reference values.

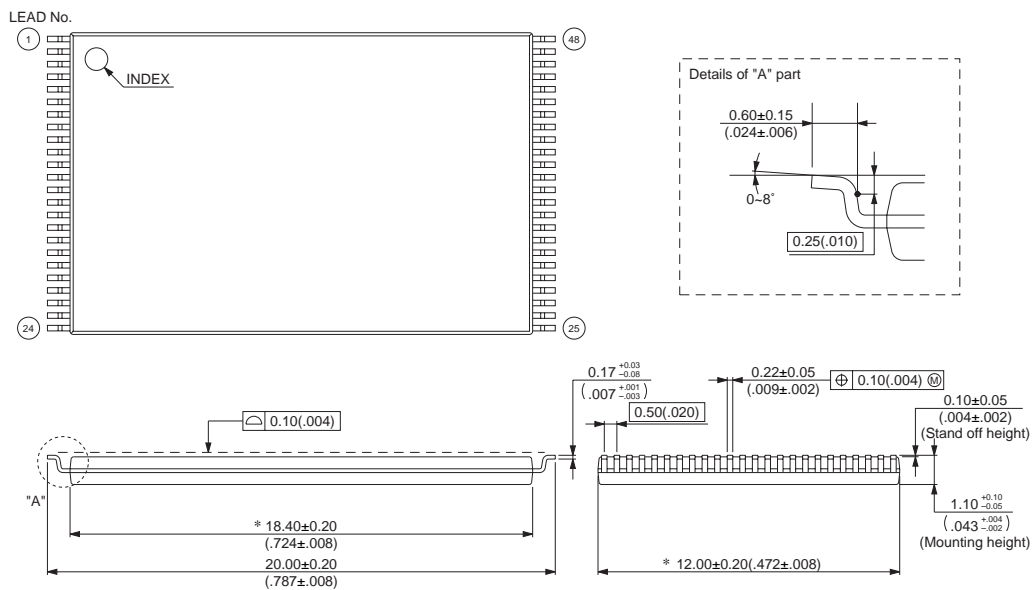
48-pin plastic TSOP(1)  
(FPT-48P-M20)

Note 1) \* : Values do not include resin protrusion.

Resin protrusion and gate protrusion are +0.15 (.006) Max (each side).

Note 2) Pins width and pins thickness include plating thickness.

Note 3) Pins width do not include tie bar cutting remainder.



© 2003 FUJITSU LIMITED F48030S-c-6-7

Dimensions in mm (inches).

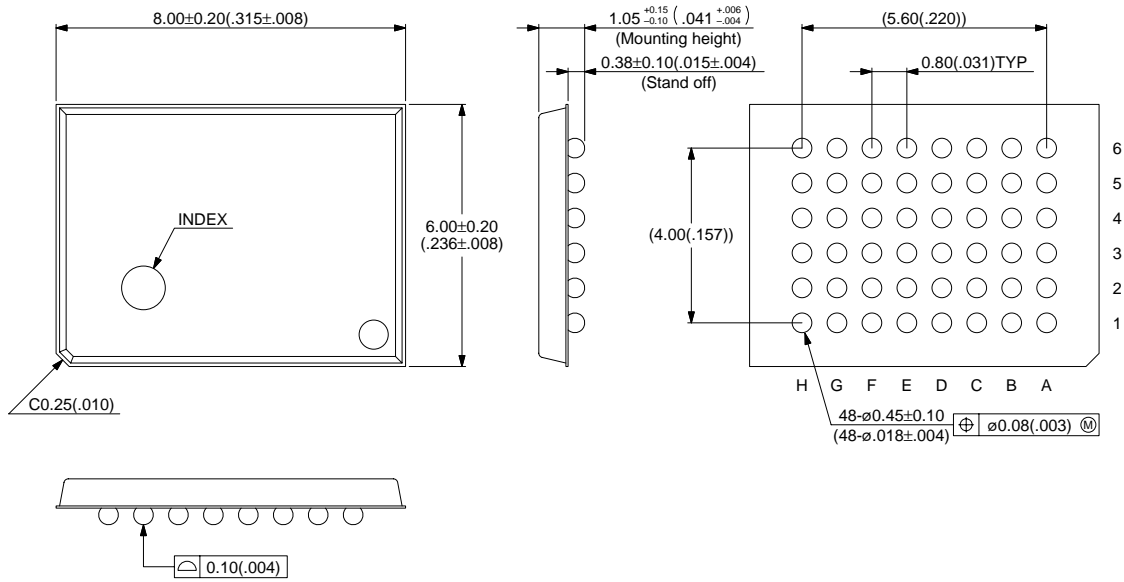
Note : The values in parentheses are reference values.

(Continued)

# MBM29DL16XTE/BE70/90

(Continued)

## 48-pin plastic FBGA (BGA-48P-M11)



© 2001 FUJITSU LIMITED B48011S-c-5-3

Dimensions in mm (inches).

Note : The values in parentheses are reference values.

## FUJITSU LIMITED

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, such as descriptions of function and application circuit examples, in this document are presented solely for the purpose of reference to show examples of operations and uses of Fujitsu semiconductor device; Fujitsu does not warrant proper operation of the device with respect to use based on such information. When you develop equipment incorporating the device based on such information, you must assume any responsibility arising out of such use of the information. Fujitsu assumes no liability for any damages whatsoever arising out of the use of the information.

Any information in this document, including descriptions of function and schematic diagrams, shall not be construed as license of the use or exercise of any intellectual property right, such as patent right or copyright, or any other right of Fujitsu or any third party or does Fujitsu warrant non-infringement of any third-party's intellectual property right or other right by using such information. Fujitsu assumes no liability for any infringement of the intellectual property rights or other rights of third parties which would result from the use of information contained herein.

The products described in this document are designed, developed and manufactured as contemplated for general use, including without limitation, ordinary industrial use, general office use, personal use, and household use, but are not designed, developed and manufactured as contemplated (1) for use accompanying fatal risks or dangers that, unless extremely high safety is secured, could have a serious effect to the public, and could lead directly to death, personal injury, severe physical damage or other loss (i.e., nuclear reaction control in nuclear facility, aircraft flight control, air traffic control, mass transport control, medical life support system, missile launch control in weapon system), or (2) for use requiring extremely high reliability (i.e., submersible repeater and artificial satellite).

Please note that Fujitsu will not be liable against you and/or any third party for any claims or damages arising in connection with above-mentioned uses of the products.

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

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