# MBM29LV080A-70/90/12

Data Sheet

**SPANSION**™

July 2003

The following document specifies Spansion memory products that are now offered by both Advanced Micro Devices and Fujitsu. Although the document is marked with the name of the company that originally developed the specification, these products will be offered to customers of both AMD and Fujitsu.

## Continuity of Specifications

There is no change to this datasheet as a result of offering the device as a Spansion product. Any changes that have been made are the result of normal datasheet improvement and are noted in the document revision summary, where supported. Future routine revisions will occur when appropriate, and changes will be noted in a revision summary.

## Continuity of Ordering Part Numbers

AMD and Fujitsu continue to support existing part numbers beginning with "Am" and "MBM". To order these products, please use only the Ordering Part Numbers listed in this document.

## For More Information

Please contact your local AMD or Fujitsu sales office for additional information about Spansion memory solutions.

**AMD**

**FUJITSU**

# FLASH MEMORY
## CMOS

# 8M (1M × 8) BIT

# MBM29LV080A-70/90/12

## ■ DESCRIPTION

The MBM29LV080A is a 16 M-bit, 3.0 V-only Flash memory organized as 1 M bytes of 8 bits each. The 1 M bytes of data is divided into 32 sectors of 64 K bytes of flexible erase capability. The 8 bits of data will appear on $DQ_0$ to $DQ_7$. The MBM29LV080A is offered in a 40-pin TSOP (I) package. The device is 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 device can also be reprogrammed in standard EPROM programmers.
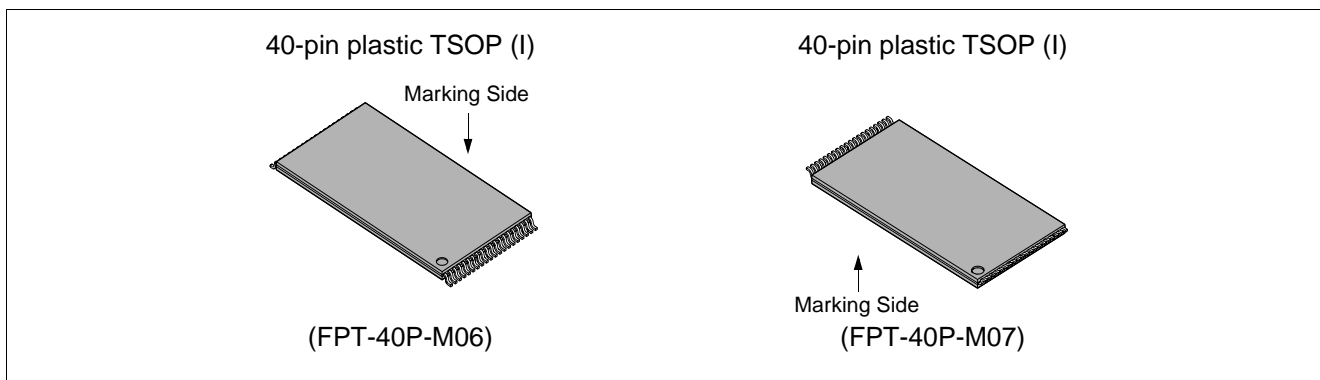
The standard MBM29LV080A offers access times of 70 ns and 120 ns, allowing operation of high-speed micro-processors without wait states. To eliminate bus contention the device has separate chip enable ($\overline{CE}$), write enable ($\overline{WE}$), and output enable ($\overline{OE}$) controls.

*(Continued)*

## ■ PRODUCT LINE UP

| | MBM29LV080A | | |
|---|---|---|---|
| | -70 | -90 | -12 |
| Power Supply Voltage $V_{CC}$ (V) | 3.3 V $^{+0.3\,V}_{-0.3\,V}$ | 3.0 V $^{+0.6\,V}_{-0.3\,V}$ | |
| Max. Address Access Time (ns) | 70 | 90 | 120 |
| Max. $\overline{CE}$ Access Time (ns) | 70 | 90 | 120 |
| Max. $\overline{OE}$ Access Time (ns) | 30 | 35 | 50 |

## ■ PACKAGES

40-pin plastic TSOP (I)

Marking Side

(FPT-40P-M06)

40-pin plastic TSOP (I)

Marking Side

(FPT-40P-M07)

*(Continued)*

The MBM29LV080A is pin and command set compatible with JEDEC standard E$^2$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 device is similar to reading from 5.0 V and 12.0 V Flash or EPROM devices.

The MBM29LV080A is 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 margins. 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 device automatically times the erase pulse widths and verifies proper cell margins.

Any individual sector is typically erased and verified in 1.0 second. (If already preprogrammed.)

The device also features a sector erase architecture. The sector mode allows each sector to be erased and reprogrammed without affecting other sectors. The MBM29LV080A is erased when shipped from the factory.

Fujitsu has implemented an Erase Suspend feature that enables the user to put erase on hold for any period of time to read data from or program data to a mom-busy sector. Thus, true background erase can be achieved.

The device features 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 $\overline{Data}$ Polling of DQ$_7$, by the Toggle Bit feature on DQ$_6$, or the RY/$\overline{BY}$ output pin. Once the end of a program or erase cycle has been completed, the device internally resets to the read mode.

The MBM29LV080A also has a hardware $\overline{RESET}$ pin. When this pin is driven low, execution of any Embedded Program Algorithm or Embedded Erase Algorithm is terminated. The internal state machine is then reset to the read mode. The $\overline{RESET}$ pin may be tied to the system reset circuitry. Therefore, if a system reset occurs during the Embedded Program Algorithm or Embedded Erase Algorithm, the device is automatically reset to the read mode and will have erroneous data stored in the address locations being programmed or erased. These locations need re-writing after the Reset. Resetting the device enables the system's microprocessor to read the boot-up firmware from the Flash memory.
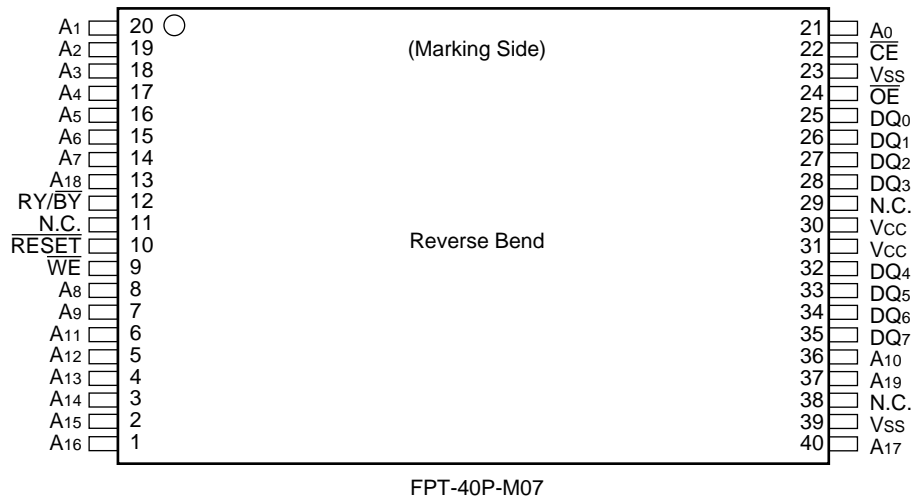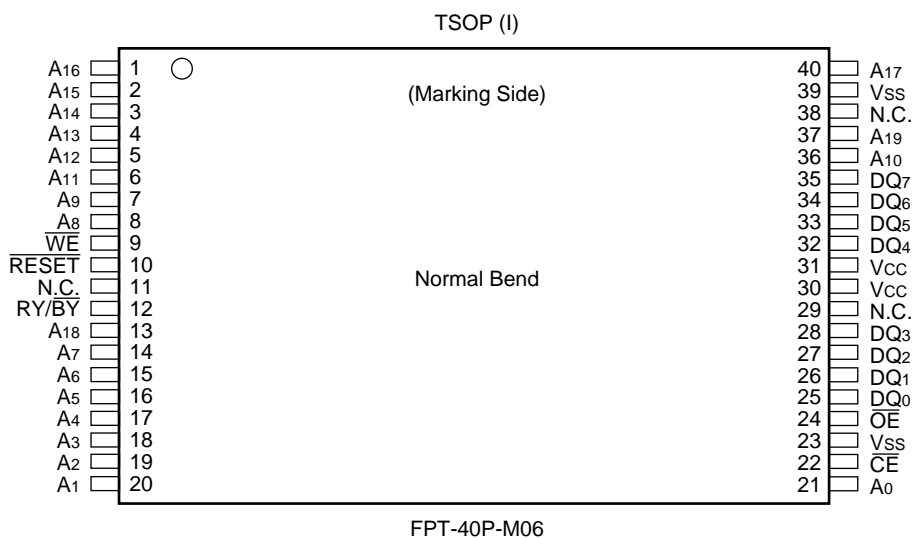
Fujitsu's Flash technology combines years of Flash memory manufacturing experience to produce the highest levels of quality, reliability, and cost effectiveness. The MBM29LV080A memory electrically erases all bits within a sector simultaneously via Fowler-Nordhiem tunneling. The bytes are programmed one byte at a time using the EPROM programming mechanism of hot electron injection.

## ■ FEATURES

- **Address specification is not necessary during command sequence**
- **Single 3.0 V read, program and erase**
  Minimizes system level power requirements
- **Compatible with JEDEC-standard commands**
  Uses same software commands as E$^2$PROMs
- **Compatible with JEDEC-standard world-wide pinouts**
  40-pin TSOP (I) (Package suffix: PTN-Normal Bend Type, PTR-Reversed Bend Type)
- **Minimum 100,000 program/erase cycles**
- **High performance**
  70 ns maximum access time
- **Sector erase architecture**
  16 sectors of 64 K bytes each
  Any combination of sectors can be concurrently erased. MBM29LV080A also supports full chip erase.
- **Embedded Erase™\* Algorithms**
  Automatically pre-programs and erases the chip or any sector
- **Embedded Program™\* Algorithms**
  Automatically programs and verifies data at specified address
- **$\overline{\text{Data}}$ polling and toggle bit feature for detection of program or erase cycle completion**
- **Ready/Busy output (RY/$\overline{\text{BY}}$)**
  Hardware method for detection of program or erase cycle completion
- **Automatic sleep mode**
  When addresses remain stable, automatically switches themselves to low power mode
- **Low V$_{CC}$ write inhibit $\leq$ 2.5 V**
- **Hardware $\overline{\text{RESET}}$ pin**
  Resets internal state machine to the read mode
- **Erase suspend/resume**
  Suspends the erase operation to allow a read data and/or program in another sector within the same device
- **Sector protection**
  Hardware method disables any combination of sectors from program or erase operations
- **Sector protection set function by extended sector protect command**
- **Temporary sector unprotection**
  Temporary sector unprotection via the $\overline{\text{RESET}}$ pin


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

3

## ■ PIN ASSIGNMENTS

TSOP (I)

```
        A16 ⊏  1   ○              40 ⊐ A17
        A15 ⊏  2     (Marking Side)  39 ⊐ VSS
        A14 ⊏  3              38 ⊐ N.C.
        A13 ⊏  4              37 ⊐ A19
        A12 ⊏  5              36 ⊐ A10
        A11 ⊏  6              35 ⊐ DQ7
         A9 ⊏  7              34 ⊐ DQ6
         A8 ⊏  8              33 ⊐ DQ5
         WE ⊏  9              32 ⊐ DQ4
      RESET ⊏ 10              31 ⊐ VCC
       N.C. ⊏ 11   Normal Bend  30 ⊐ VCC
      RY/BY ⊏ 12              29 ⊐ N.C.
        A18 ⊏ 13              28 ⊐ DQ3
         A7 ⊏ 14              27 ⊐ DQ2
         A6 ⊏ 15              26 ⊐ DQ1
         A5 ⊏ 16              25 ⊐ DQ0
         A4 ⊏ 17              24 ⊐ OE
         A3 ⊏ 18              23 ⊐ VSS
         A2 ⊏ 19              22 ⊐ CE
         A1 ⊏ 20              21 ⊐ A0
```

FPT-40P-M06

```
         A1 ⊏ 20   ○              21 ⊐ A0
         A2 ⊏ 19     (Marking Side)  22 ⊐ CE
         A3 ⊏ 18              23 ⊐ VSS
         A4 ⊏ 17              24 ⊐ OE
         A5 ⊏ 16              25 ⊐ DQ0
         A6 ⊏ 15              26 ⊐ DQ1
         A7 ⊏ 14              27 ⊐ DQ2
        A18 ⊏ 13              28 ⊐ DQ3
      RY/BY ⊏ 12              29 ⊐ N.C.
       N.C. ⊏ 11              30 ⊐ VCC
      RESET ⊏ 10   Reverse Bend  31 ⊐ VCC
         WE ⊏  9              32 ⊐ DQ4
         A8 ⊏  8              33 ⊐ DQ5
         A9 ⊏  7              34 ⊐ DQ6
        A11 ⊏  6              35 ⊐ DQ7
        A12 ⊏  5              36 ⊐ A10
        A13 ⊏  4              37 ⊐ A19
        A14 ⊏  3              38 ⊐ N.C.
        A15 ⊏  2              39 ⊐ VSS
        A16 ⊏  1              40 ⊐ A17
```

FPT-40P-M07

4

## ■ PIN DESCRIPTIONS

| Pin Name | Function |
|---|---|
| $A_0$ to $A_{19}$ | Address Inputs |
| $DQ_0$ to $DQ_7$ | Data Inputs/Outputs |
| $\overline{CE}$ | Chip Enable |
| $\overline{OE}$ | Output Enable |
| $\overline{WE}$ | Write Enable |
| RY/$\overline{BY}$ | Ready/Busy Output |
| $\overline{RESET}$ | Hardware Reset Pin/Temporary Sector Unprotection |
| $V_{SS}$ | Device Ground |
| $V_{CC}$ | Device Power Supply |
| N.C. | Pin Not Connected Internally |

## ■ BLOCK DIAGRAM
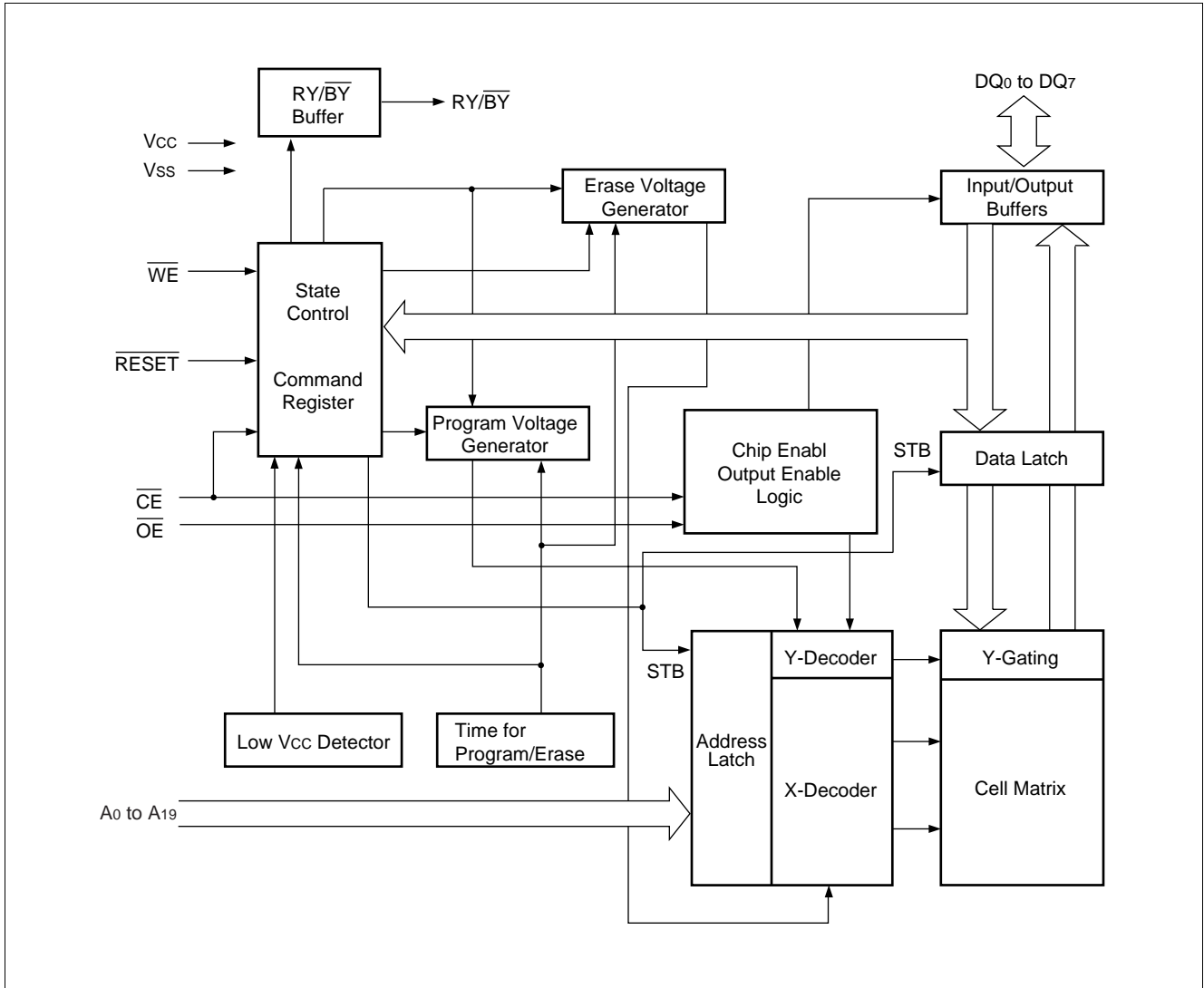


## ■ LOGIC SYMBOL
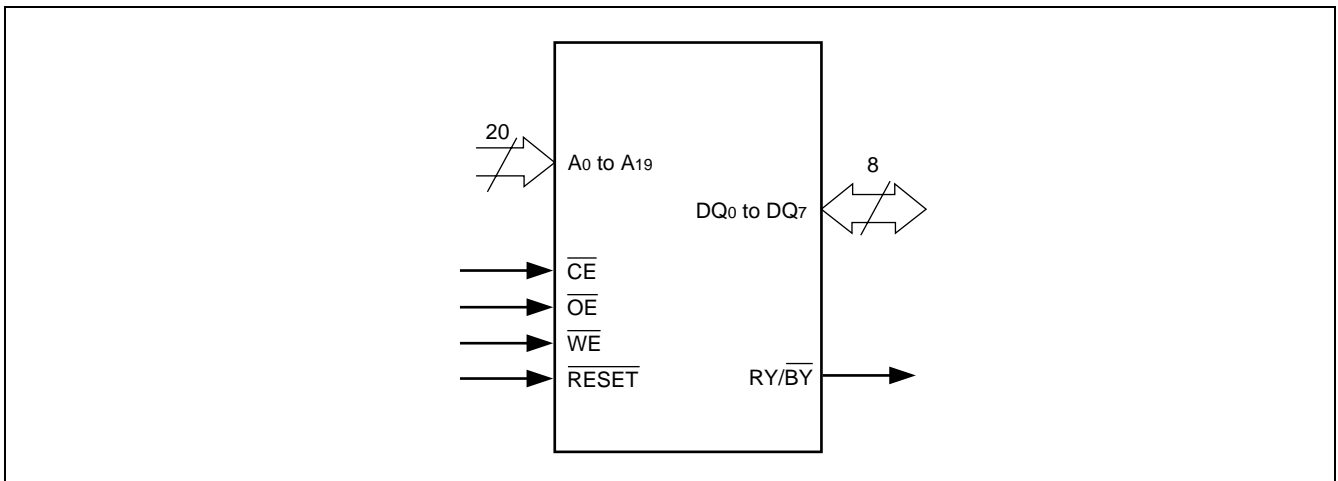
## ■ DEVICE BUS OPERATIONS

**Table 1 User Bus Operation**

| Operation | $\overline{CE}$ | $\overline{OE}$ | $\overline{WE}$ | $A_0$ | $A_1$ | $A_6$ | $A_9$ | $A_{10}$ | $DQ_0$ to $DQ_7$ | $\overline{RESET}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Auto-Select Manufacture Code *1 | L | L | H | L | L | L | $V_{ID}$ | L | Code | H |
| Auto-Select Device Code *1 | L | L | H | H | L | L | $V_{ID}$ | L | Code | H |
| Read*2 | L | L | H | $A_0$ | $A_1$ | $A_6$ | $A_9$ | $A_{10}$ | $D_{OUT}$ | H |
| Standby | H | X | X | X | X | X | X | X | High-Z | H |
| Output Disable | L | H | H | X | X | X | X | X | High-Z | H |
| Write (Program/Erase) | L | H | L | $A_0$ | $A_1$ | $A_6$ | $A_9$ | $A_{10}$ | $D_{IN}$ | H |
| Enable Sector Protection *3, *4 | L | $V_{ID}$ | ⎍ | L | H | L | $V_{ID}$ | X | X | H |
| Verify Sector Protection *3, *4 | L | L | H | L | H | L | $V_{ID}$ | L | Code | H |
| Temporary Sector Unprotection *5 | X | X | X | X | X | X | X | X | X | $V_{ID}$ |
| Reset (Hardware)/Standby | X | X | X | X | X | X | X | X | High-Z | L |

**Legend:** L = $V_{IL}$, H = $V_{IH}$, X = $V_{IL}$ or $V_{IH}$. ⎍ = pulse input. See "■ ELECTRICAL CHARACTERISTICS
1. DC Characteristics" for voltage levels.

*1 : Manufacturer and device codes may also be accessed via a command register write sequence. See Table 3.

*2 : $\overline{WE}$ can be $V_{IL}$ if $\overline{OE}$ is $V_{IL}$, $\overline{OE}$ at $V_{IH}$ initiates the write operations.

*3 : Refer to the section on Sector Protection.

*4 : $V_{CC}$ = 3.3 V ±10%

*5 : It is also used for the extended sector protection.

**Table 2 Standard Command Definitions**

| 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 | XXXh | F0h | — | — | — | — | — | — | — | — | — | — |
| Read/Reset* | 3 | XXXh | AAh | XXXh | 55h | XXXh | F0h | RA | RD | — | — | — | — |
| Autoselect | 3 | XXXh | AAh | XXXh | 55h | XXXh | 90h | — | — | — | — | — | — |
| Byte Program | 4 | XXXh | AAh | XXXh | 55h | XXXh | A0h | PA | PD | — | — | — | — |
| Chip Erase | 6 | XXXh | AAh | XXXh | 55h | XXXh | 80h | XXXh | AAh | XXXh | 55h | XXXh | 10h |
| Sector Erase | 6 | XXXh | AAh | XXXh | 55h | XXXh | 80h | XXXh | AAh | XXXh | 55h | SA | 30h |
| Sector Erase Suspend | 1 | XXXh | B0h | — | — | — | — | — | — | — | — | — | — |
| Sector Erase Resume | 1 | XXXh | 30h | — | — | — | — | — | — | — | — | — | — |

*: Both Read/Reset commands are functionally equivalent, resetting the device to the read mode.

Notes: • Address bit = X = "H" or "L".
    • Bus operations are defined in "Table 1 User Bus Operation".
    • RA = Address of the memory location to be read.
      PA = Address of the memory location to be programmed. Addresses are latched on the falling edge of the $\overline{WE}$ pulse.
      SA = Address of the sector to be erased. The combination of $A_{19}$, $A_{18}$, $A_{17}$, and $A_{16}$ will uniquely select any sector.
    • RD = Data read from location RA during read operation.
      PD = Data to be programmed at location PA. Data is latched on the rising edge of $\overline{WE}$.

**Table 3 Extended Command Definitions**

| Command Sequence | Bus Write Cycles Req'd | First Bus Write Cycle | | Second Bus Write Cycle | | Third Bus Write Cycle | | Fourth Bus Read Cycle | |
|---|---|---|---|---|---|---|---|---|---|
| | | Addr | Data | Addr | Data | Addr | Data | Addr | Data |
| Fast Mode Set | 3 | XXXh | AAh | XXXh | 55h | XXXh | 20h | — | — |
| Fast Program *1 | 2 | XXXh | A0h | PA | PD | — | — | — | — |
| Fast Mode Reset *1 | 2 | XXXh | 90h | XXXh | F0h *3 | — | — | — | — |
| Extended Sector Protection *2 | 4 | XXXh | 60h | SPA | 60h | SPA | 40h | SPA | SD |

SPA:Sector address to be protected. Set sector address (SA) and $(A_{10}, A_6, A_1, A_0)$ = (0, 0, 1, 0).
SD: Sector protection verify data. Output 01h at protected sector addresses and output 00h at unprotected sector addresses.

*1: This command is valid during Fast Mode.

*2: This command is valid while $\overline{RESET}$=$V_{ID}$.

*3: The data "00h" is also acceptable.

**Table 4.1 Sector Protection Verify Autoselect Code**

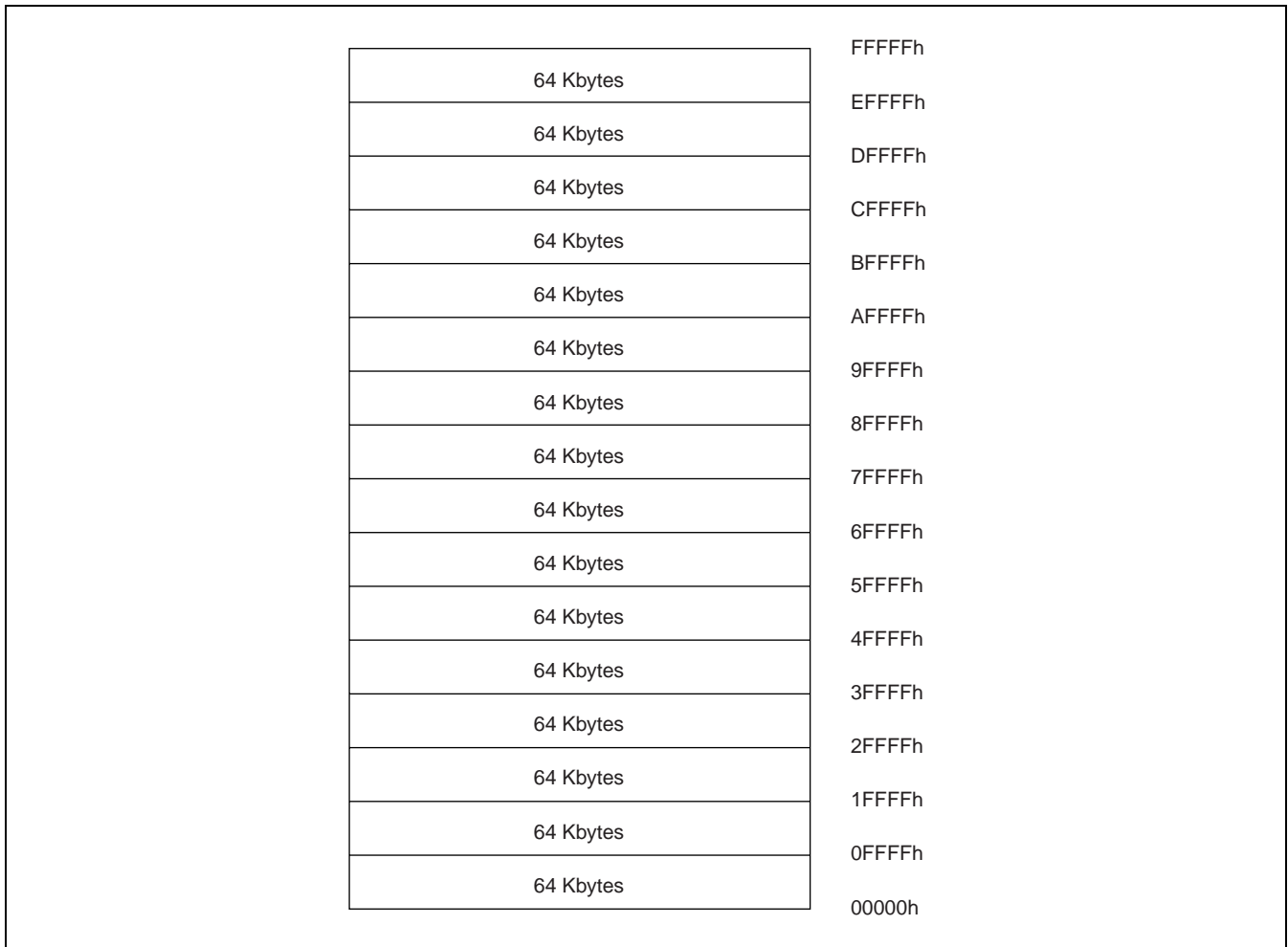| Type | $A_{16}$ to $A_{19}$ | $A_{10}$ | $A_6$ | $A_1$ | $A_0$ | Code (HEX) |
|---|---|---|---|---|---|---|
| Manufacture's Code | X | $V_{IL}$ | $V_{IL}$ | $V_{IL}$ | $V_{IL}$ | 04h |
| Device Code | X | $V_{IL}$ | $V_{IL}$ | $V_{IL}$ | $V_{IH}$ | 38h |
| Sector Protection | Sector Addresses | $V_{IL}$ | $V_{IL}$ | $V_{IH}$ | $V_{IL}$ | 01h* |

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

**Table 4.2 Expanded Autoselect Code**

| Type | Code | $DQ_7$ | $DQ_6$ | $DQ_5$ | $DQ_4$ | $DQ_3$ | $DQ_2$ | $DQ_1$ | $DQ_0$ |
|---|---|---|---|---|---|---|---|---|---|
| Manufacture's Code | 04h | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| Device Code | 38h | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| Sector Protection | 01h | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

■ **FLEXIBLE SECTOR-ERASE ARCHITECTURE**

- Sixteen 64 K byte sectors
- Loading the sector erase buffer may be done in any sequence and with any number of sectors (0 to 15).
- Individual-sector or multiple-sector erase capability
- Sector protection is user-definable.

| | |
|---|---|
| 64 Kbytes | FFFFFh |
| 64 Kbytes | EFFFFh |
| 64 Kbytes | DFFFFh |
| 64 Kbytes | CFFFFh |
| 64 Kbytes | BFFFFh |
| 64 Kbytes | AFFFFh |
| 64 Kbytes | 9FFFFh |
| 64 Kbytes | 8FFFFh |
| 64 Kbytes | 7FFFFh |
| 64 Kbytes | 6FFFFh |
| 64 Kbytes | 5FFFFh |
| 64 Kbytes | 4FFFFh |
| 64 Kbytes | 3FFFFh |
| 64 Kbytes | 2FFFFh |
| 64 Kbytes | 1FFFFh |
| 64 Kbytes | 0FFFFh |
| | 00000h |

**Table 5 Sector Address**

| Sector Address | A$_{19}$ | A$_{18}$ | A$_{17}$ | A$_{16}$ | Address Range |
|---|---|---|---|---|---|
| SA0 | 0 | 0 | 0 | 0 | 00000h to 0FFFFh |
| SA1 | 0 | 0 | 0 | 1 | 10000h to 1FFFFh |
| SA2 | 0 | 0 | 1 | 0 | 20000h to 2FFFFh |
| SA3 | 0 | 0 | 1 | 1 | 30000h to 3FFFFh |
| SA4 | 0 | 1 | 0 | 0 | 40000h to 4FFFFh |
| SA5 | 0 | 1 | 0 | 1 | 50000h to 5FFFFh |
| SA6 | 0 | 1 | 1 | 0 | 60000h to 6FFFFh |
| SA7 | 0 | 1 | 1 | 1 | 70000h to 7FFFFh |
| SA8 | 1 | 0 | 0 | 0 | 80000h to 8FFFFh |
| SA9 | 1 | 0 | 0 | 1 | 90000h to 9FFFFh |
| SA10 | 1 | 0 | 1 | 0 | A0000h to AFFFFh |
| SA11 | 1 | 0 | 1 | 1 | B0000h to BFFFFh |
| SA12 | 1 | 1 | 0 | 0 | C0000h to CFFFFh |
| SA13 | 1 | 1 | 0 | 1 | D0000h to DFFFFh |
| SA14 | 1 | 1 | 1 | 0 | E0000h to EFFFFh |
| SA15 | 1 | 1 | 1 | 1 | F0000h to FFFFFh |

## ■ FUNCTIONAL DESCRIPTION

### Read Mode

The MBM29LV080A has 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 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 $t_{ACC}$ - $t_{CE}$ time.) See "Figure 5.1 Read Operation Timing Diagram" for timing specifications.

### Standby Mode

There are two ways to implement the standby mode on the MBM29LV080A device, 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 μA 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 μA 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 MBM29LV080A data.

This mode can be used effectively with an application requested low power consumption such as handy terminals. To activate this mode, MBM29LV080A automatically switches itself to low power mode when MBM29LV080A 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 μA (CMOS level).

Standard address access timings provide new data when addresses are changed. While in sleep mode, output data is latched and always available to the system.

### Output Disable

With the $\overline{OE}$ input at a logic high level ($V_{IH}$), output from the device is 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 device and will identify its manufacturer and type. This mode is intended for use by programming equipment for the purpose of automatically matching the device to be programmed with its corresponding programming algorithm. This mode is functional over the entire temperature range of the device.

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 device outputs by toggling address $A_0$ from $V_{IL}$ to $V_{IH}$. All addresses are DON'T CARES except $A_0$, $A_1$, $A_6$, and $A_{10}$. (See "Table 4.1 Sector Protection Verify Autoselect Code".)

The manufacturer and device codes may also be read via the command register, for instances when the MBM29LV080A is erased or programmed in a system without access to high voltage on the $A_9$ pin. The command sequence is illustrated in Table 2. (Refer to Autoselect Command section.)

Byte 0 ($A_0 = V_{IL}$) represents the manufacture's code (Fujitsu = 04h) and byte 1 ($A_0 = V_{IH}$) represents the device identifier code MBM29LV080A = 38h. All identifiers for manufactures and device will exhibit odd parity with $DQ_7$ defined as the parity bit. In order to read the proper device codes when executing the autoselect, $A_1$ must be $V_{IL}$. (See "Table 4.1 Sector Protection Verify Autoselect Code" and "Table 4.2 Expanded Autoselect Code".)

In order to determine which sectors are write protected, $A_1$ must be at $V_{IH}$ while running through the sector addresses; if the selected sector is protected, a logical '1' will be output on $DQ_0$ ($DQ_0$ = 1).

## 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 $V_{IL}$, while $\overline{CE}$ is at $V_{IL}$ and $\overline{OE}$ is at $V_{IH}$. 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 Write Characteristics and the Erase/Programming Timing Diagram for specific timing parameters.

## Sector Protection

The MBM29LV080A features hardware sector protection. This feature will disable both program and erase operations in any number of sectors (0 through 15). The sector protection feature is enabled using programming equipment at the user's site. The device is shipped with all sectors unprotected.
To activate this mode, the programming equipment must force $V_{ID}$ on address pin $A_9$ and control pin $\overline{OE}$, (suggest $V_{ID}$ = 11.5 V), $\overline{CE}$ = $V_{IL}$, $A_0$ = $A_6$ = $V_{IL}$, and $A_1$ = $V_{IH}$. The sector addresses ($A_{19}$, $A_{18}$, $A_{17}$,and $A_{16}$) should be set to the sector to be protected. Table 5 define the sector address for each of the sixteen (16) individual 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 addresses must be held constant during the $\overline{WE}$ pulse. See figures 13 and 21 for sector protection waveforms and algorithm.
To verify programming of the protection circuitry, the programming equipment must force $V_{ID}$ on address pin $A_9$ with $\overline{CE}$ and $\overline{OE}$ at $V_{IL}$ and $\overline{WE}$ at $V_{IH}$. Scanning the sector addresses ($A_{19}$, $A_{18}$, $A_{17}$,and $A_{16}$) while ($A_{10}$, $A_6$, $A_1$, $A_0$) = (0, 0, 1, 0) will produce a logical "1" code at device output $DQ_0$ for a protected sector. Otherwise the devices will read 00h for unprotected sector. In this mode, the lower order addresses, except for $A_0$, $A_1$, $A_6$, and $A_{10}$ are DON'T CARES. Address locations with $A_1$ = $V_{IL}$ are reserved for Autoselect manufacturer and device codes.
It is also possible to determine if a sector is protected in the system by writing an Autoselect command. Performing a read operation at the address location XX02h, where the higher order addresses ($A_{19}$, $A_{18}$, $A_{17}$,and $A_{16}$). are the sector address will produce a logical "1" at $DQ_0$ for a protected sector. See "Table 4.1 Sector Protection Verify Autoselect Code" and "Table 4.2 Expanded Autoselect Code" for Autoselect codes.

## Temporary Sector Unprotection

This feature allows temporary unprotection of previously protected sectors of the MBM29LV080A device in order to change data. The Sector Unprotection mode is activated by setting the $\overline{RESET}$ pin to high voltage (12 V). During this mode, formerly protected sectors can be programmed or erased by selecting the sector addresses. Once the 12 V is taken away from the $\overline{RESET}$ pin, all the previously protected sectors will be protected again. See "Figure 15 Temporary Sector Unprotection Timing Diagram" and "Figure 22 Temporary Sector Unprotection Algorithm".

## 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 read mode. "Table 3 Extended Command Definitions" 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 $DQ_0$ to $DQ_7$ bits are ignored.

## Read/Reset Command

The read or 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 device remains 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 the AC Read Character-istics and the specific timing parameters.(See "Figure 5.1 Read Operation Timing Diagram" and "Figure 5.2 Hardware Reset/Read Operation 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 device resides in the target system. PROM pro-grammers typically access the signature codes by raising $A_9$ 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 method-ology. The operation is initiated by writing the Autoselect command sequence into the command register. Fol-lowing the command write, a read cycle from address XX00h retrieves the manufacture code of 04h. A read cycle from address X001h returns the device code (MBM29LV080A = 38h). (See "Table 4.1 Sector Protection Verify Autoselect Code" and "Table 4.2 Expanded Autoselect Code".)

All manufacturer and device codes will exhibit odd parity with the MSB ($DQ_7$) defined as the parity bit.

Sector state (protection or unprotection) will be informed address X0002h.

Scanning the sector addresses ($A_{19}$, $A_{18}$, $A_{17}$, $A_{16}$) while ($A_{10}$, $A_6$, $A_1$, $A_0$) = (0, 0, 1, 0) will produce a logical "1" at device output $DQ_0$ for a protected sector. The programming verification should be perform margin mode on the protected sector. (See "■ DEVICE BUS OPERATIONS".)

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 se-quence.

## Byte Programming

The device is programmed on a byte-by-byte 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 automatic programming operation is completed when the data on $DQ_7$ is equivalent to data written to this bit at which time the device returns to the read mode and addresses are no longer latched. (See "Table 7 Toggle Bit Status" and "Table 6 Hardware Sequence Flags".) Therefore, the device requires that a valid address to the device 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 pro-gramming 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 (exceed timing limits), 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.

"Figure 17 Embedded Program™ Algorithm" illustrates the Embedded Program™ Algorithm using typical com-mand strings and bus operations.

## 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 device will automatically program and verify the entire memory for an all zero data pattern prior to electrical erase. The system is not required to provide any controls or timings during these operations.

The automatic erase begins on the rising edge of the last $\overline{WE}$ pulse in the command sequence and terminates when the data on $DQ_7$ is "1" (See "Write Operation Status".) at which time the device returns to read the mode.

"Figure 18 Embedded Erase™ Algorithm" 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 $\overline{WE}$, while the command (Data = 30h) is latched on the rising edge of $\overline{WE}$. After time-out of 50 $\mu$s 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 "Table 3 Extended Command Definitions". 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 50 μs, 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 50 $\mu$s from the rising edge of the last $\overline{WE}$ will initiate the execution of the Sector Erase command(s). If another falling edge of the $\overline{WE}$ occurs within the 50 μs time-out window the timer is reset. (Monitor $DQ_3$ to determine if the sector erase timer window is still open, see section $DQ_3$, 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 that sector. In that case, restart the erase on those sectors and allow them to complete. (Refer to the "Write Operation Status" for Sector Erase Timer operation.) Loading the sector erase buffer may be done in any sequence and with any number of sectors (0 to 15).

Sector erase does not require the user to program the devices prior to erase. The device automatically programs all memory locations in the sector(s) to be erased prior to electrical erase. 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 automatic sector erase begins after the 50 μs time out from the rising edge of the $\overline{WE}$ pulse for the last sector erase command pulse and terminates when the data on $DQ_7$ is "1" (See the section on "Write Operation Status") at which time the device returns to the read mode. Data polling must be performed at an address within any of the sectors being erased. Multiple Sector Erase Time; [Sector Program Time (Preprogramming) + Sector Erase Time] $\times$ Number of Sector Erase.
"Figure 18 Embedded Erase™ Algorithm" 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 program 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 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 resumes the erase operation. The addresses are "DON'T CARES" when writing the Erase Suspend or Erase Resume commands.
When the Erase Suspend command is written during the Sector Erase operation, the device will take a maximum of 20 $\mu$s to suspend the erase operation. When the devices have entered the erase-suspended mode, the RY/$\overline{BY}$ output pin 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 device defaults 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 the section on "$DQ_2$".)
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, pro-

15

gramming 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 RY/$\overline{BY}$ output pin, $\overline{Data}$ polling of $DQ_7$, or the Toggle Bit ($DQ_6$) 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.

To resume the operation of Sector Erase, the Resume command (30h) should be written. 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

MBM29LV080A has 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 normal 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. (Refer to "Figure 23 Extended Sector Protection Algorithm".) 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 Setup command (A0h) and data write cycles (PA/PD). (Refer to "Figure 23 Extended Sector Protection Algorithm".)

(3) Extended Sector Protection

In addition to normal sector protection, the MBM29LV080A has Extended Sector Protection as extended function. This function enables to protect sector 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 only $\overline{RESET}$ pin requires $V_{ID}$ for sector protection in this mode. The extended sector protect requires $V_{ID}$ on $\overline{RESET}$ pin. With this condition, the operation is initiated by writing the Setup command (60h) into the command register. Then, the sector addresses pins ($A_{19}$, $A_{18}$, $A_{17}$, and $A_{16}$) and ($A_{10}$, $A_6$, $A_1$, $A_0$) = (0, 0, 1, 0) should be set to the sector to be protected (recommend to set $V_{IL}$ for the other addresses pins), and write extended sector protect command (60h). A sector is typically protected in 250 μs. To verify programming of the protection circuitry, the sector addresses pins ($A_{19}$, $A_{18}$, $A_{17}$, and $A_{16}$) and ($A_{10}$, $A_6$, $A_1$, $A_0$) = (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 protect command (60h) again. To terminate the operation, it is necessary to set $\overline{RESET}$ pin to $V_{IH}$.

**Write Operation Status**

**Table 6 Hardware Sequence Flags**

| Status | | | $DQ_7$ | $DQ_6$ | $DQ_5$ | $DQ_3$ | $DQ_2$ |
|---|---|---|---|---|---|---|---|
| In Progress | Embedded Program Algorithm | | $\overline{DQ_7}$ | Toggle | 0 | 0 | 1 |
| | Embedded/Erase Algorithm | | 0 | Toggle | 0 | 1 | Toggle |
| | Erase Suspend 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) | $\overline{DQ_7}$ | Toggle*1 | 0 | 0 | 1*2 |
| Exceeded Time Limits | Embedded Program Algorithm | | $\overline{DQ_7}$ | Toggle | 1 | 0 | 1 |
| | Embedded/Erase Algorithm | | 0 | Toggle | 1 | 1 | N/A |
| | Erase Suspend Program (Non-Erase Suspended Sector) | | $\overline{DQ_7}$ | Toggle | 1 | 0 | N/A |

*1: Performing successive read operations from any address will cause $DQ_6$ to toggle.

*2: Reading the byte address being programmed while in the erase-suspend program mode will indicate logic "1" at the $DQ_2$ bit. However, successive reads from the erase-suspended sector will cause $DQ_2$ to toggle.

Notes : • $DQ_0$ and $DQ_1$ are reserve pins for future use.
  • $DQ_4$ is Fujitsu internal use only.

**$DQ_7$**

**$\overline{\text{Data}}$ Polling**

The MBM29LV080A device features $\overline{\text{Data}}$ 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 device will produce the complement of the data last written to $DQ_7$. Upon completion of the Embedded Program Algorithm, an attempt to read the device will produce the true data last written to $DQ_7$. During the Embedded Erase Algorithm, an attempt to read the device will produce a "0" at the $DQ_7$ output. Upon completion of the Embedded Erase Algorithm an attempt to read the device will produce a "1" at the $DQ_7$ output. The flowchart for $\overline{\text{Data}}$ Polling ($DQ_7$) is shown in "Figure 19 $\overline{\text{Data}}$ Polling Algorithm".

For chip erase and sector erase, the $\overline{\text{Data}}$ Polling is valid after the rising edge of the sixth $\overline{\text{WE}}$ pulse in the six write pulse sequence. $\overline{\text{Data}}$ 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. Once the Embedded Algorithm operation is close to being completed, the MBM29LV080A data pins ($DQ_7$) may change asynchronously while the output enable ($\overline{\text{OE}}$) is asserted low. This means that the device is driving status information on $DQ_7$ 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 $DQ_7$ output, it may read the status or valid data. Even if the device has completed the Embedded Algorithm operation and $DQ_7$ has a valid data, the data outputs on $DQ_0$ to $DQ_6$ may be still invalid. The valid data on $DQ_0$ to $DQ_7$ 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 "Table 7 Toggle Bit Status".)

See "Figure 9 $\overline{\text{Data}}$ Polling during Embedded Algorithm Operation Timing Diagram" for the $\overline{\text{Data}}$ Polling timing specifications and diagrams.

**DQ$_6$**

**Toggle Bit I**

The MBM29LV080A also features 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 ($\overline{OE}$ toggling) data from the devices will result in DQ$_6$ toggling between one and zero. Once the Embedded Program or Erase Algorithm cycle is completed, DQ$_6$ 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 $\overline{WE}$ 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 $\overline{WE}$ 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 2 µs and then stop toggling without the data having changed. In erase, the device 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 50 µs and then drop back into read mode, having changed none of the data.

Either $\overline{CE}$ or $\overline{OE}$ toggling will cause the DQ$_6$ to toggle. In addition, an Erase Suspend/Resume command will cause the DQ$_6$ to toggle.

See "Figure 10 Toggle Bit 1 during Embedded Algorithm Operation Timing Diagram" and "Figure 20 Toggle Bit Algorithm" for the Toggle Bit I timing specifications and diagrams.

**DQ$_5$**

**Exceeded Timing Limits**

DQ$_5$ will indicate if the program or erase time has exceeded the specified limits (internal pulse count). Under these conditions DQ$_5$ will produce a "1". This is a failure condition which indicates that the program or erase cycle was not successfully completed. $\overline{Data}$ Polling DQ$_7$, DQ$_6$ is the only operating function of the device under this condition. The $\overline{CE}$ circuit will partially power down the device under these conditions (to approximately 2 mA). The $\overline{OE}$ and $\overline{WE}$ pins will control the output disable functions as described in "■ DEVICE BUS OPERATIONS".

The DQ$_5$ failure condition may also appear if a user tries to program a non blank location without erasing. In this case the device locks out and never completes the Embedded Algorithm operation. Hence, the system never reads a valid data on DQ$_7$ bit and DQ$_6$ never stops toggling. Once the device has exceeded timing limits, the DQ$_5$ bit will indicate a "1" . Please note that this is not a device failure condition since the device was incorrectly used. If this occurs, reset the device with command sequence.

**DQ$_3$**

**Sector Erase Timer**

After completion of the initial sector erase command sequence, the sector erase time-out begins. DQ$_3$ will remain low until the time-out is complete. $\overline{Data}$ Polling and Toggle Bit are valid after the initial Sector Erase command sequence.

If $\overline{Data}$ Polling or the Toggle Bit I indicates that a valid erase command has been written, DQ$_3$ may be used to determine if the sector erase timer window is still open. If DQ$_3$ 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 $\overline{Data}$ Polling or Toggle Bit I. If DQ$_3$ 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$_3$ prior to and following each subsequent sector erase command. If DQ$_3$ were high on the second status check, the command may not have been accepted.

See "Table 6 Hardware Sequence Flags".

**DQ$_2$**

**Toggle Bit II**

This Toggle Bit II, along with DQ$_6$, 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$_2$ 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 $DQ_2$ 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 $DQ_2$ bit.

$DQ_6$ is different from $DQ_2$ in that $DQ_6$ toggles only when the standard program or Erase, or Erase Suspend Program operation is in progress.

For example, $DQ_2$ and $DQ_6$ can be used together to determine the erase-suspend-read mode. ($DQ_2$ toggles while $DQ_6$ does not.) See also the below Table 7 and "Figure 16 $DQ_2$ vs. $DQ_6$".

Furthermore, $DQ_2$ can also be used to determine which sector is being erased. When the devices are in the erase mode, $DQ_2$ toggles if this bit is read from the erasing sector.

**Table 7 Toggle Bit Status**

| Mode | $DQ_7$ | $DQ_6$ | $DQ_2$ |
|---|---|---|---|
| Program | $\overline{DQ_7}$ | Toggle | 1 |
| Erase | 0 | Toggle | Toggle |
| Erase-Suspend Read (Erase-Suspended Sector)[1] | 1 | 1 | Toggle |
| Erase-Suspend Program | $\overline{DQ_7}$ | Toggle[1] | 1[2] |

[1]: Performing successive read operations from any address will cause $DQ_6$ to toggle.

[2]: Reading the address being programmed while in the erase-suspend program mode will indicate logic "1" at the $DQ_2$ bit. However, successive reads from the erase-suspended sector will cause $DQ_2$ to toggle.

## RY/$\overline{BY}$

### Ready/Busy Pin

The MBM29LV080A provides 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 device is busy with either a program or erase operation. If the output is high, the device is ready to accept any read/write or erase operation. When the RY/$\overline{BY}$ pin is low, the device will not accept any additional program or erase commands with the exception of the Erase Suspend command. If the MBM29LV080A is placed in an Erase Suspend mode, the RY/$\overline{BY}$ output will be high, by means of connecting with a pull up resistor to $V_{CC}$.

During programming, the RY/$\overline{BY}$ pin is driven low after the rising edge of the fourth $\overline{WE}$ pulse. During an erase operation, the RY/$\overline{BY}$ pin is driven low after the rising edge of the sixth $\overline{WE}$ pulse. The RY/$\overline{BY}$ pin will indicate a busy condition during the $\overline{RESET}$ pulse. See "Figure 11 RY/$\overline{BY}$ Timing Diagram during Program/Erase Operations" and "Figure 12 $\overline{RESET}$, RY/$\overline{BY}$ 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, RY/$\overline{BY}$ pins can be tied together in parallel with a pull-up resistor to $V_{CC}$.

## $\overline{RESET}$

### Hardware Reset Pin

The MBM29LV080A device may be reset by driving the $\overline{RESET}$ pin to $V_{IL}$. The $\overline{RESET}$ pin has a pulse requirement and has to be kept low ($V_{IL}$) for at least 500 ns 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 $t_{READY}$ after the $\overline{RESET}$ pin is driven low. Furthermore, once the $\overline{RESET}$ pin goes into high, the devices require an additional $t_{RH}$ before it will allow read access. When the $\overline{RESET}$ pin is low, the device 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 RY/$\overline{BY}$ output signal should be ignored during the $\overline{RESET}$ pulse. See "Figure 12 $\overline{RESET}$, RY/$\overline{BY}$ Timing Diagram" for the timing diagram. Refer to "Figure 15 Temporary Sector Unprotection Timing Diagram" for additional functionality.

If hardware reset occurs during Embedded Erase Algorithm, there is the possibility that the erasing sector(s) cannot be used.

## Data Protection

The MBM29LV080A is 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 device automatically resets 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 form $V_{CC}$ 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 2.3 V (typically 2.4 V). If $V_{CC} < V_{LKO}$, the command register is disabled and all internal program/erase circuits are disabled. Under this condition the device will be 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 2.3 V.
If Embedded Erase Algorithm is interrupted, there is the 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 logical zero while $\overline{OE}$ is a logical one.

## Power-up Write Inhibit

Power-up of the device 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.

## ■ ABSOLUTE MAXIMUM RATINGS

| Parameter | Symbol | Conditions | Rating | | Unit |
|---|---|---|---|---|---|
| | | | Min. | Max. | |
| Storage Temperature | Tstg | — | −55 | +125 | °C |
| Ambient Temperature with Power Applied | $T_A$ | — | −40 | +85 | °C |
| Voltage with respect to Ground All pins except A$_9$, $\overline{OE}$, $\overline{RESET}$ *1 | $V_{IN}$, $V_{OUT}$ | — | −0.5 | $V_{CC}$+0.5 | V |
| Power Supply Voltage *1 | $V_{CC}$ | — | −0.5 | +5.5 | V |
| A$_9$, $\overline{OE}$, and $\overline{RESET}$ *2 | $V_{IN}$ | — | −0.5 | +13.0 | V |

*1 : Minimum DC voltage on input or I/O pins is −0.5 V. During voltage transitions, input or I/O pins may undershoot $V_{SS}$ to −2.0 V for periods of up to 20 ns. Maximum DC voltage on input or I/O pins is $V_{CC}$ + 0.5 V. During voltage transitions, input or I/O pins may overshoot to $V_{CC}$ + 2.0 V for periods of up to 20 ns.

*2 : Minimum DC input voltage on A$_9$, $\overline{OE}$ and $\overline{RESET}$ pins is -0.5 V. During voltage transitions, A$_9$, $\overline{OE}$ and $\overline{RESET}$ pins may undershoot $V_{SS}$ to −2.0 V for periods of up to 20 ns. Voltage difference between input and supply voltage ($V_{IN}$ − $V_{CC}$) does not exceed 9.0 V. Maximum DC input voltage on A$_9$, $\overline{OE}$ and $\overline{RESET}$ pins is +13.0 V which may overshoot to 14.0 V for periods of up to 20 ns.

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 RANGES

| Parameter | Symbol | Conditions | Value | | Unit |
|---|---|---|---|---|---|
| | | | Min. | Max. | |
| Ambient Temperature | $T_A$ | — | −40 | +85 | °C |
| Power Supply Voltage | $V_{CC}$ | MBM29LV080A-70 | +3.0 | +3.6 | V |
| | | MBM29LV080A-90/12 | +2.7 | +3.6 | V |

Note : Operating ranges define those limits between which the functionality of the device is guaranteed.

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.

## ■ MAXIMUM OVERSHOOT / UNDERSHOOT



+0.6 V
−0.5 V
−2.0 V

20 ns
20 ns
20 ns
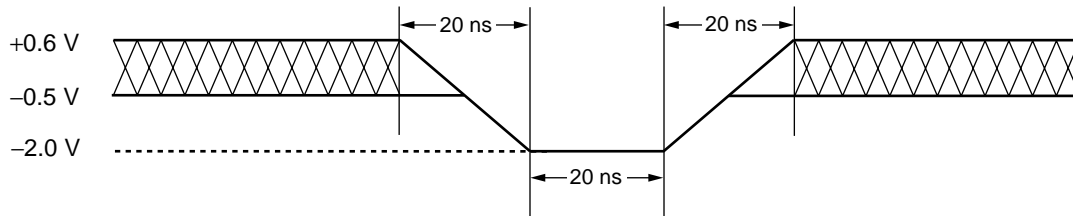
**Figure 1 Maximum Undershoot Waveform**



$V_{CC}$ +2.0 V
$V_{CC}$ +0.5 V
+2.0 V

20 ns
20 ns
20 ns
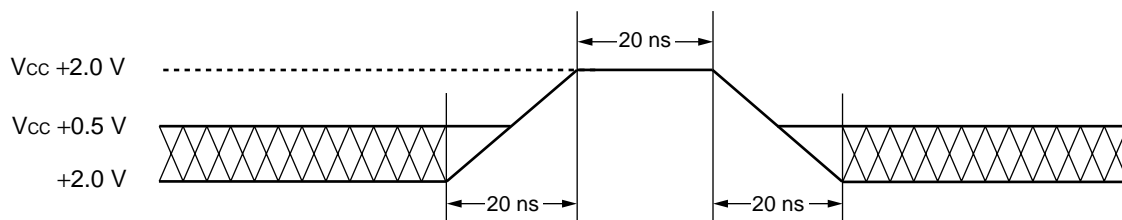
**Figure 2 Maximum Overshoot Waveform 1**



+14.0 V
+13.0 V
$V_{CC}$ +0.5 V

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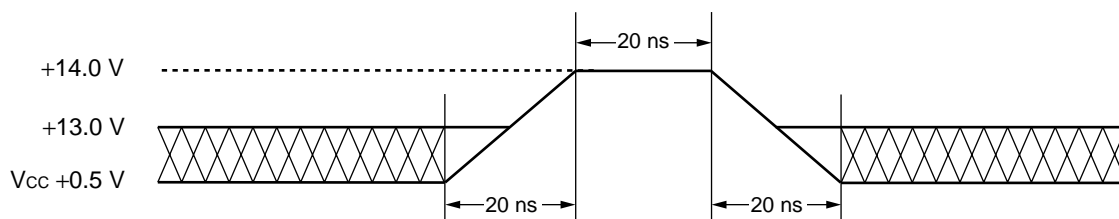
Note : This waveform is applied for $A_9$, $\overline{OE}$ and $\overline{RESET}$.

**Figure 3 Maximum Overshoot Waveform 2**

## ■ ELECTRICAL CHARACTERISTICS

### 1. DC Characteristics

| Parameter | Symbol | Condition | Value Min. | Value Max. | Unit |
|---|---|---|---|---|---|
| Input Leakage Current | $I_{LI}$ | $V_{IN}$ = $V_{SS}$ to $V_{CC}$, $V_{CC}$ = $V_{CC}$ Max. | −1.0 | +1.0 | µA |
| Output Leakage Current | $I_{LO}$ | $V_{OUT}$ = $V_{SS}$ to $V_{CC}$, $V_{CC}$ = $V_{CC}$ Max. | −1.0 | +1.0 | µA |
| $A_9$, $\overline{OE}$, $\overline{RESET}$ Inputs Leakage Current | $I_{LIT}$ | $V_{CC}$ = $V_{CC}$ Max., $A_9$, $\overline{OE}$, $\overline{RESET}$ = 12.5 V | — | 35 | µA |
| $V_{CC}$ Active Current*1 | $I_{CC1}$ | $\overline{CE}$ = $V_{IL}$, $\overline{OE}$ = $V_{IH}$, f = 10 MHz | — | 22 | mA |
| | | $\overline{CE}$ = $V_{IL}$, $\overline{OE}$ = $V_{IH}$, f = 5 MHz | — | 12 | mA |
| $V_{CC}$ 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 | — | 5 | µA |
| $V_{CC}$ Current during Reset (Standby, $\overline{RESET}$) | $I_{CC4}$ | $V_{CC}$ = $V_{CC}$ Max., $\overline{RESET}$ = $V_{SS} \pm 0.3$ V | — | 5 | µA |
| $V_{CC}$ Current (Automatic Sleep Mode) *3 | $I_{CC5}$ | $V_{CC}$ = $V_{CC}$ Max., $\overline{RESET}$ = $V_{CC} \pm 0.3$ V, $\overline{CE}$ = $V_{SS} \pm 0.3$ V, $V_{IN}$ = $V_{CC} \pm 0.3$ V or $V_{SS} \pm 0.3$ V | — | 5 | µA |
| Input Low Level | $V_{IL}$ | — | −0.5 | 0.6 | V |
| Input High Level | $V_{IH}$ | — | 2.0 | $V_{CC}$ + 0.3 | V |
| Voltage for Autoselect, Sector Protection and Temporary Sector Unprotection ($A_9$, $\overline{OE}$, $\overline{RESET}$) *4, *5 | $V_{ID}$ | — | 11.5 | 12.5 | V |
| Output Low Voltage Level | $V_{OL}$ | $I_{OL}$ = 4.0 mA, $V_{CC}$ = $V_{CC}$ Min. | — | 0.45 | V |
| Output High Voltage Level | $V_{OH1}$ | $I_{OH}$ = −2.0 mA, $V_{CC}$ = $V_{CC}$ Min. | 2.4 | — | V |
| | $V_{OH2}$ | $I_{OH}$ = −100 µA | $V_{CC}$ − 0.4 | — | V |
| Low $V_{CC}$ Lock-Out Voltage | $V_{LKO}$ | — | 2.3 | 2.5 | V |

*1 : The $I_{CC}$ current listed includes both the DC operating current and the frequency dependent component.

*2 : $I_{CC}$ active while Embedded Erase or Embedded Program is in progress.

*3 : Automatic sleep mode enables the low power mode when address remains stable for 150 ns.

*4 : This voltage is for Sector Protection operation.

*5 : ($V_{ID}$ - $V_{CC}$) do not exceed 9 V.

## 2. AC Characteristics

### • Read Only Operations Characteristics

| Parameter | Symbol | | Test Setup | Value | | | | | | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 70 | | 90 | | 12 | | |
| | JEDEC | Standard | | Min. | Max. | Min. | Max. | Min. | Max. | |
| Read Cycle Time | t$_{AVAV}$ | t$_{RC}$ | — | 70 | — | 90 | — | 120 | — | ns |
| Address to Output Delay | t$_{AVQV}$ | t$_{ACC}$ | $\overline{CE}$ = V$_{IL}$ $\overline{OE}$ = V$_{IL}$ | — | 70 | — | 90 | — | 120 | ns |
| Chip Enable to Output Delay | t$_{ELQV}$ | t$_{CE}$ | $\overline{OE}$ = V$_{IL}$ | — | 70 | — | 90 | — | 120 | ns |
| Output Enable to Output Delay | t$_{GLQV}$ | t$_{OE}$ | — | — | 30 | — | 35 | — | 50 | ns |
| Chip Enable to Output High-Z | t$_{EHQZ}$ | t$_{DF}$ | — | — | 25 | — | 30 | — | 30 | ns |
| Output Enable to Output High-Z | t$_{GHQZ}$ | t$_{DF}$ | — | — | 25 | — | 30 | — | 30 | ns |
| Output Hold Time From Address, $\overline{CE}$ or $\overline{OE}$, Whichever Occurs First | t$_{AXQX}$ | t$_{OH}$ | — | 0 | — | 0 | — | 0 | — | ns |
| $\overline{RESET}$ Pin Low to Read Mode | — | t$_{READY}$ | — | — | 20 | — | 20 | — | 20 | µs |

Note: Test Conditions: Output Load: 1 TTL gate and 30 pF (MBM29LV080A-70)
                                                1 TTL gate and 100 pF (MBM29LV080A-90/12)
           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

Notes: C$_L$ = 30 pF including jig capacitance (MBM29LV080A-70)
           C$_L$ = 100 pF including jig capacitance (MBM29LV080A-90/12)

**Figure 4 Test Conditions**
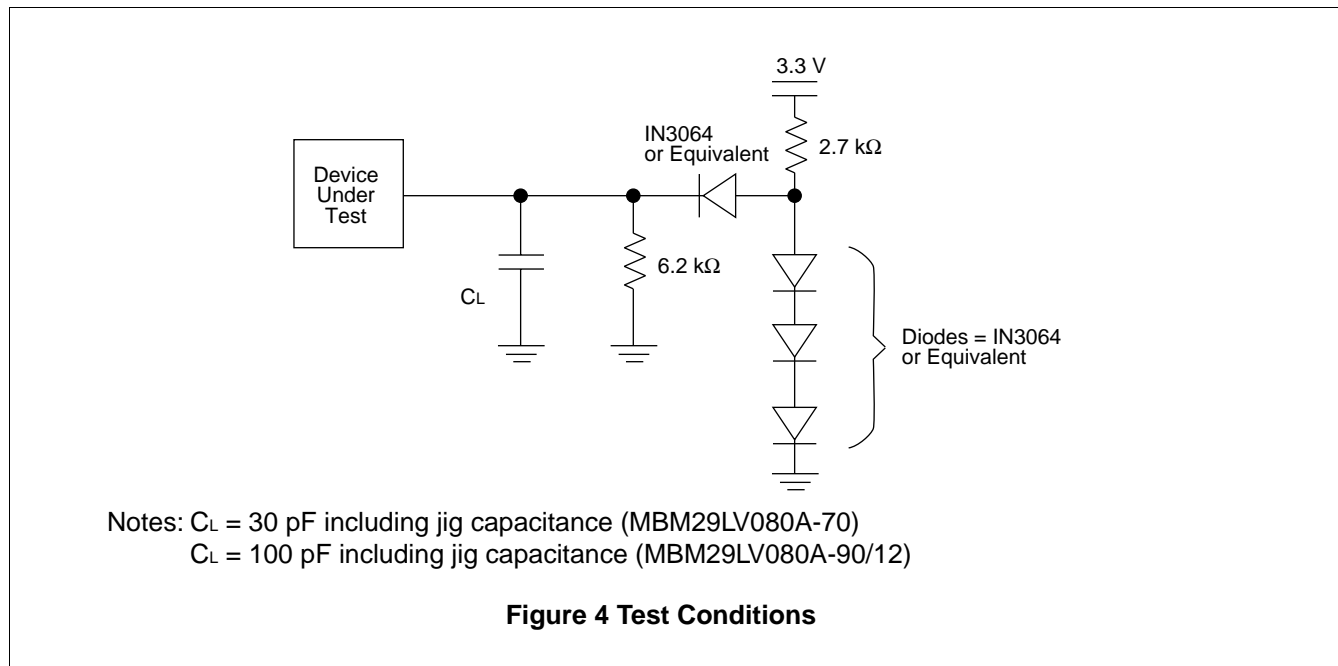
- **Write (Erase/Program) Operations Characteristics**

| Parameter | | Symbol | | MBM29LV080A | | | | | | | | | Unit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | **70** | | | **90** | | | **12** | | | |
| | | JEDEC | Standard | Min. | Typ. | Max. | Min. | Typ. | Max. | Min. | Typ. | Max. | |
| Write Cycle Time | | $t_{AVAV}$ | $t_{WC}$ | 70 | — | — | 90 | — | — | 120 | — | — | ns |
| Address Setup Time | | $t_{AVWL}$ | $t_{AS}$ | 0 | — | — | 0 | — | — | 0 | — | — | ns |
| Address Hold Time | | $t_{WLAX}$ | $t_{AH}$ | 45 | — | — | 45 | — | — | 50 | — | — | ns |
| Data Setup Time | | $t_{DVWH}$ | $t_{DS}$ | 35 | — | — | 45 | — | — | 50 | — | — | ns |
| Data Hold Time | | $t_{WHDX}$ | $t_{DH}$ | 0 | — | — | 0 | — | — | 0 | — | — | ns |
| Output Enable Setup Time | | — | $t_{OES}$ | 0 | — | — | 0 | — | — | 0 | — | — | ns |
| Output Enable Hold Time | Read | — | $t_{OEH}$ | 0 | — | — | 0 | — | — | 0 | — | — | ns |
| | Toggle and $\overline{Data}$ Polling | | | 10 | — | — | 10 | — | — | 10 | — | — | ns |
| Read Recover Time Before Write ($\overline{OE}$ High to $\overline{WE}$ Low) | | $t_{GHWL}$ | $t_{GHWL}$ | 0 | — | — | 0 | — | — | 0 | — | — | ns |
| Read Recover Time Before Write ($\overline{OE}$ High to $\overline{CE}$ Low) | | $t_{GHEL}$ | $t_{GHEL}$ | 0 | — | — | 0 | — | — | 0 | — | — | ns |
| $\overline{CE}$ Setup Time | | $t_{ELWL}$ | $t_{CS}$ | 0 | — | — | 0 | — | — | 0 | — | — | ns |
| $\overline{WE}$ Setup Time | | $t_{WLEL}$ | $t_{WS}$ | 0 | — | — | 0 | — | — | 0 | — | — | ns |
| $\overline{CE}$ Hold Time | | $t_{WHEH}$ | $t_{CH}$ | 0 | — | — | 0 | — | — | 0 | — | — | ns |
| $\overline{WE}$ Hold Time | | $t_{EHWH}$ | $t_{WH}$ | 0 | — | — | 0 | — | — | 0 | — | — | ns |
| Write Pulse Width | | $t_{WLWH}$ | $t_{WP}$ | 35 | — | — | 45 | — | — | 50 | — | — | ns |
| $\overline{CE}$ Pulse Width | | $t_{ELEH}$ | $t_{CP}$ | 35 | — | — | 45 | — | — | 50 | — | — | ns |
| Write Pulse Width High | | $t_{WHWL}$ | $t_{WPH}$ | 25 | — | — | 25 | — | — | 30 | — | — | ns |
| $\overline{CE}$ Pulse Width High | | $t_{EHEL}$ | $t_{CPH}$ | 25 | — | — | 25 | — | — | 30 | — | — | ns |
| Programming Operation | | $t_{WHWH1}$ | $t_{WHWH1}$ | — | 8 | — | — | 8 | — | — | 8 | — | µs |
| Sector Erase Operation [1] | | $t_{WHWH2}$ | $t_{WHWH2}$ | — | 1 | — | — | 1 | — | — | 1 | — | s |
| Delay Time from Embedded Output Enable | | — | $t_{EOE}$ | — | — | 30 | — | — | 35 | — | — | 50 | ns |
| $V_{CC}$ Setup Time | | — | $t_{VCS}$ | 50 | — | — | 50 | — | — | 50 | — | — | µs |
| Voltage Transition Time [2] | | — | $t_{VLHT}$ | 4 | — | — | 4 | — | — | 4 | — | — | µs |
| Write Pulse Width [2] | | — | $t_{WPP}$ | 100 | — | — | 100 | — | — | 100 | — | — | µs |
| $\overline{OE}$ Setup Time to $\overline{WE}$ Active [2] | | — | $t_{OESP}$ | 4 | — | — | 4 | — | — | 4 | — | — | µs |
| $\overline{CE}$ Setup Time to $\overline{WE}$ Active [2] | | — | $t_{CSP}$ | 4 | — | — | 4 | — | — | 4 | — | — | µs |
| Recover Time From RY/$\overline{BY}$ | | — | $t_{RB}$ | 0 | — | — | 0 | — | — | 0 | — | — | ns |
| $\overline{RESET}$ Hold Time Before Read | | — | $t_{RH}$ | 200 | — | — | 200 | — | — | 200 | — | — | ns |

*(Continued)*

*(Continued)*

| Parameter | Symbol | | MBM29LV080A | | | | | | | | | Unit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | 70 | | | 90 | | | 12 | | | |
| | JEDEC | Standard | Min. | Typ. | Max. | Min. | Typ. | Max. | Min. | Typ. | Max. | |
| Program/Erase Valid to RY/$\overline{BY}$ Delay | — | t$_{BUSY}$ | — | — | 90 | — | — | 90 | — | — | 90 | ns |
| Rise Time to V$_{ID}$ *2 | — | t$_{VIDR}$ | 500 | — | — | 500 | — | — | 500 | — | — | ns |
| $\overline{RESET}$ Pulse Width | — | t$_{RP}$ | 500 | — | — | 500 | — | — | 500 | — | — | ns |

*1 : This does not include the preprogramming time.

*2 : This timing is for Sector Protection operation.

# ■ TIMING DIAGRAM

## • Key to Switching Waveforms

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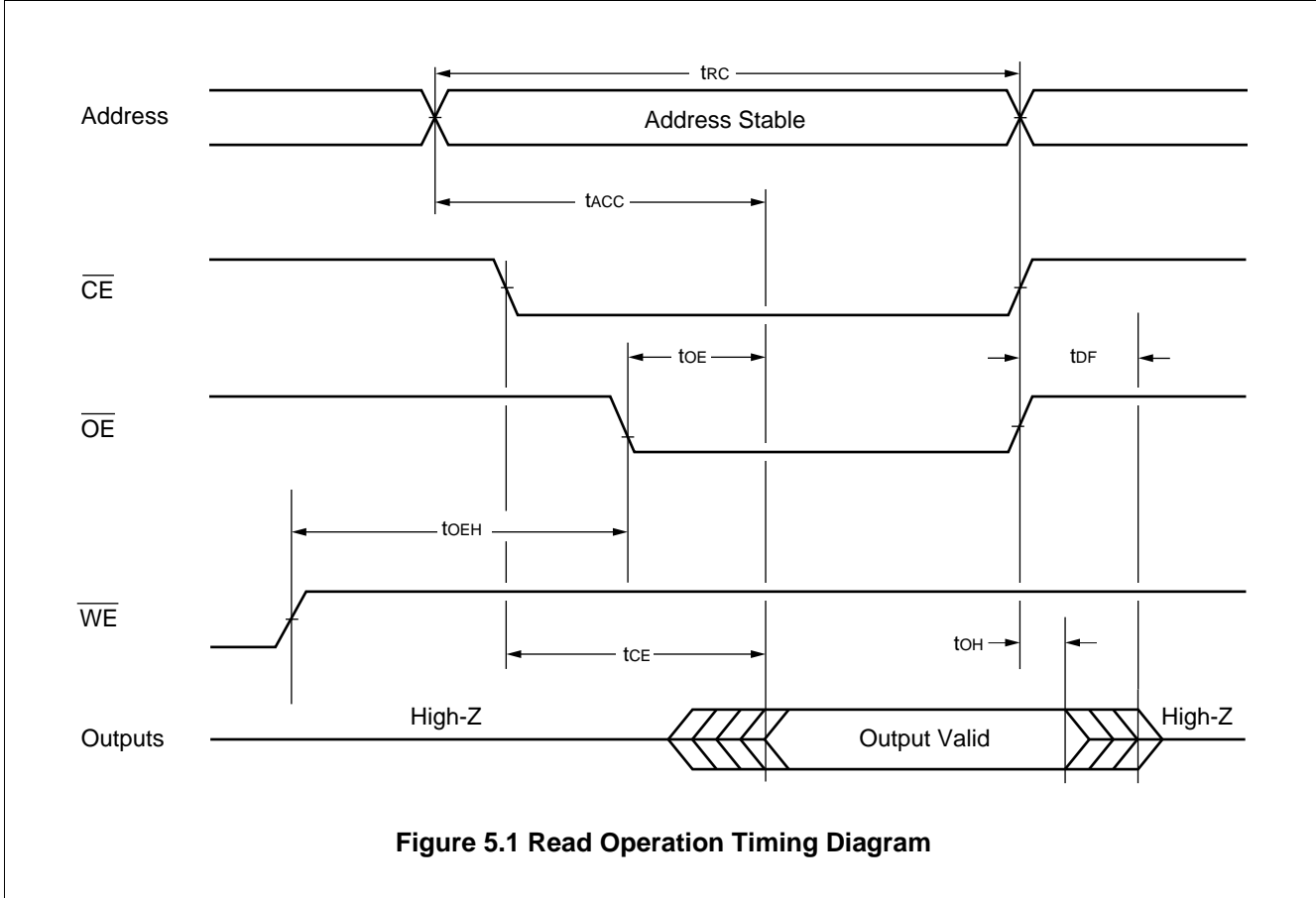


**Figure 5.1 Read Operation Timing Diagram**
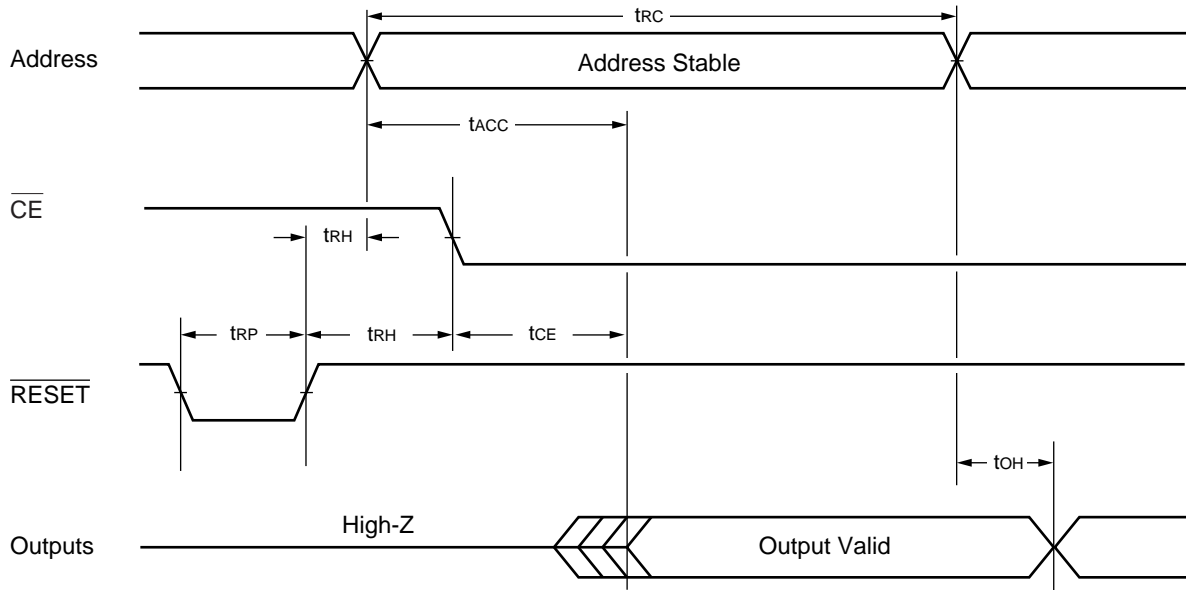
**Figure 5.2 Hardware Reset/Read Operation Timing Diagram**
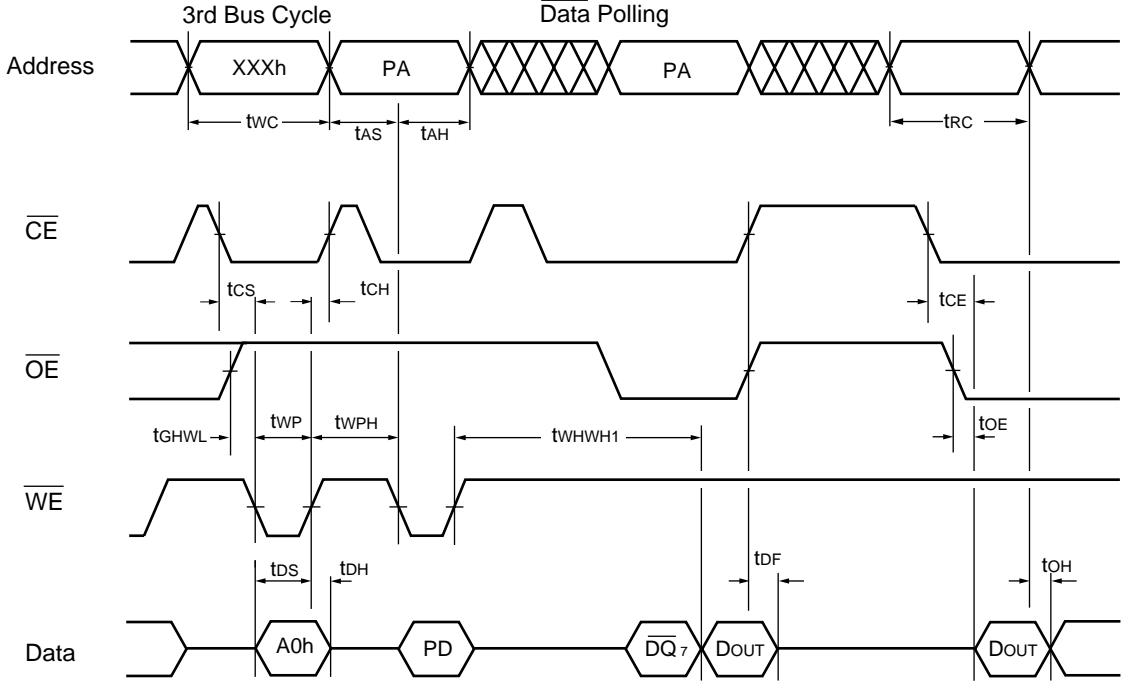
Notes: • PA is an address of the memory location to be programmed.
 • PD is data to be programmed at the byte address.
 • $\overline{DQ}_7$ is the output of the complement of the data written to the device.
 • D$_{OUT}$ is the output of the data written to the device.
 • Figure indicates last two bus cycles out of four bus cycle sequence.

**Figure 6 Alternate $\overline{WE}$ Controlled Program Operation Timing Diagram**
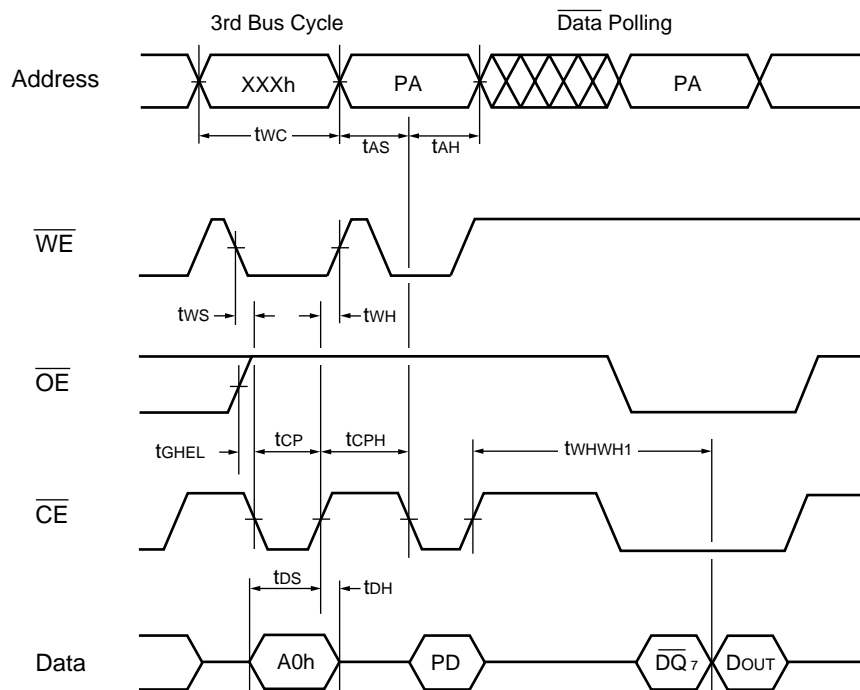
Notes: • PA is an address of the memory location to be programmed.
 • PD is data to be programmed at the byte address.
 • $\overline{DQ}_7$ is the output of the complement of the data written to the device.
 • $D_{OUT}$ is the output of the data written to the device.
 • Figure indicates last two bus cycles out of four bus cycle sequence.

**Figure 7 Alternate $\overline{CE}$ Controlled Program Operation Timing Diagram**

Address: XXXh XXXh XXXh XXXh XXXh SA*

tWC tAS tAH

$\overline{CE}$

tCS tCH

$\overline{OE}$

tGHWL tWP tWPH

$\overline{WE}$

tDS tDH                    30h for Sector Erase

Data: AAh 55h 80h AAh 55h 10h

tVCS

Vcc
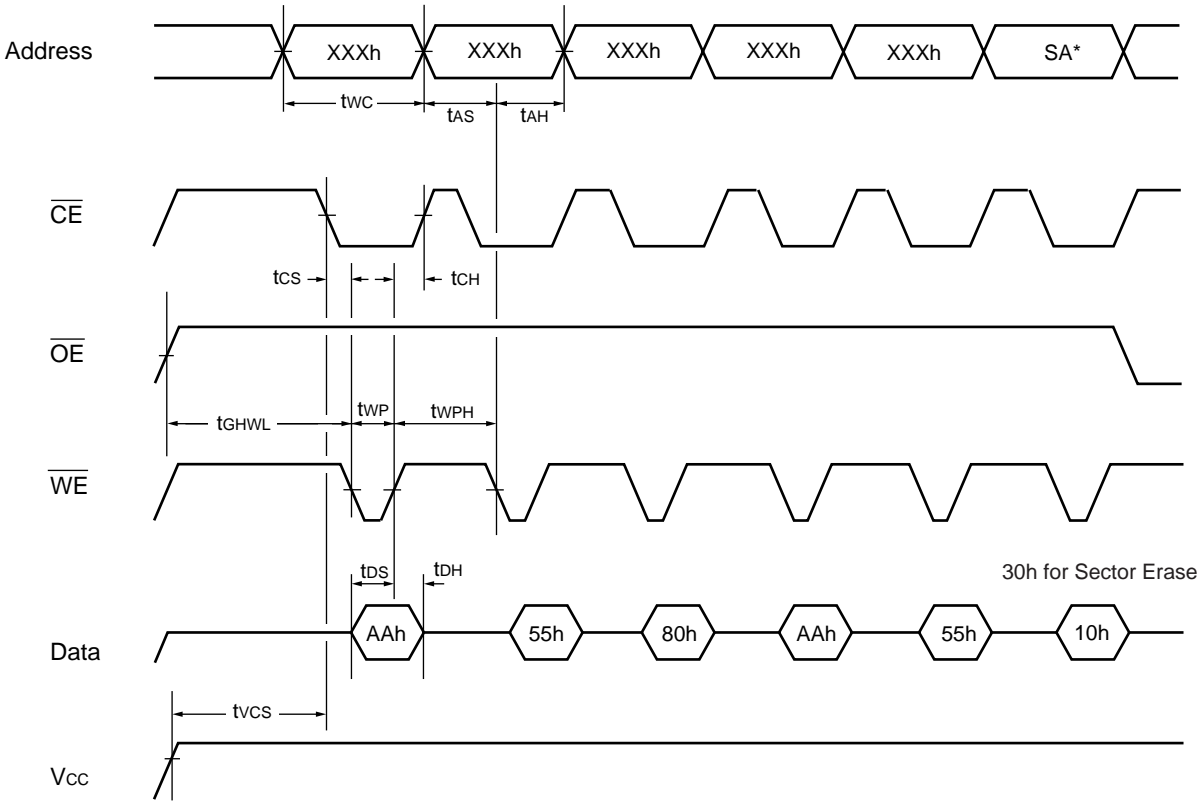
*: SA is the sector address for Sector Erase. Addresses = XXXh for Chip Erase.

**Figure 8 Chip/Sector Erase Operation Timing Diagram**
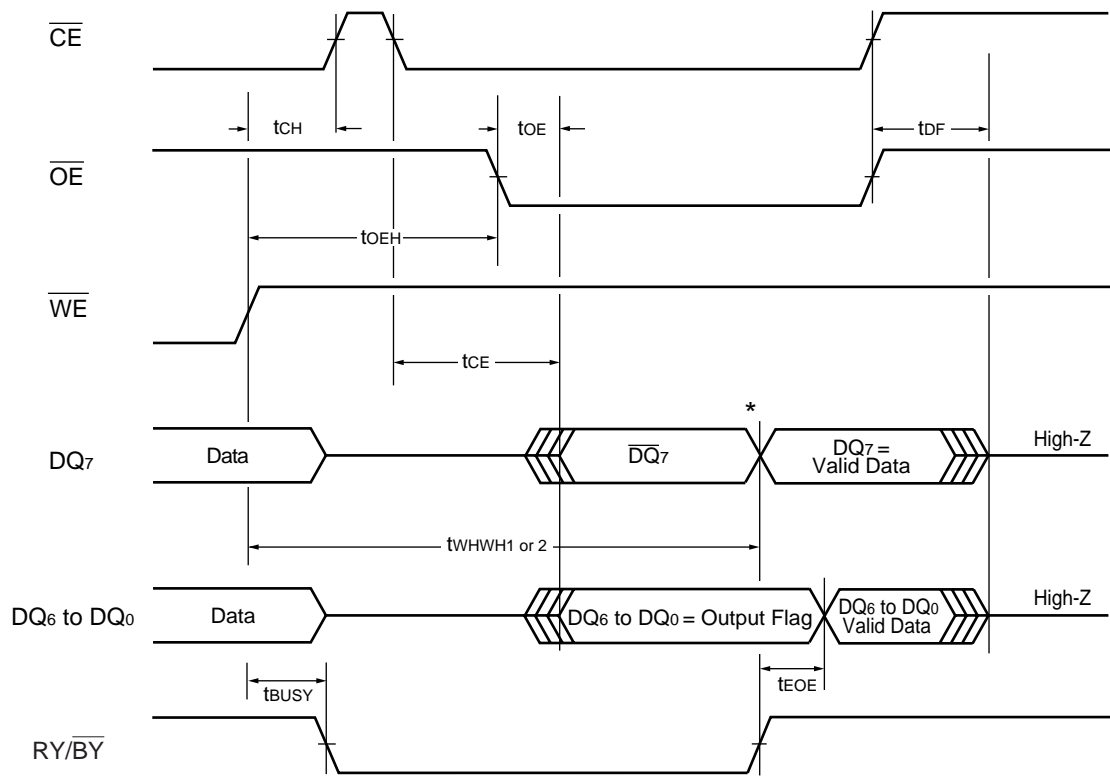
\*: $DQ_7$ = Valid Data (The device has completed the Embedded operation.)

**Figure 9 $\overline{\text{Data}}$ Polling during Embedded Algorithm Operation Timing Diagram**
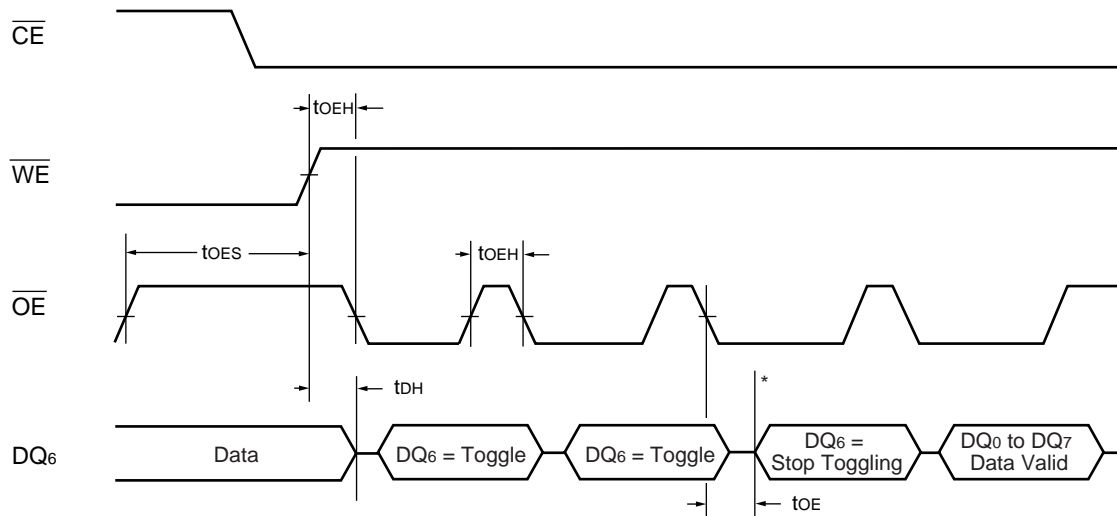


\*: $DQ_6$ stops toggling. (The device has completed the Embedded operation.)

**Figure 10 Toggle Bit I during Embedded Algorithm Operation Timing Diagram**
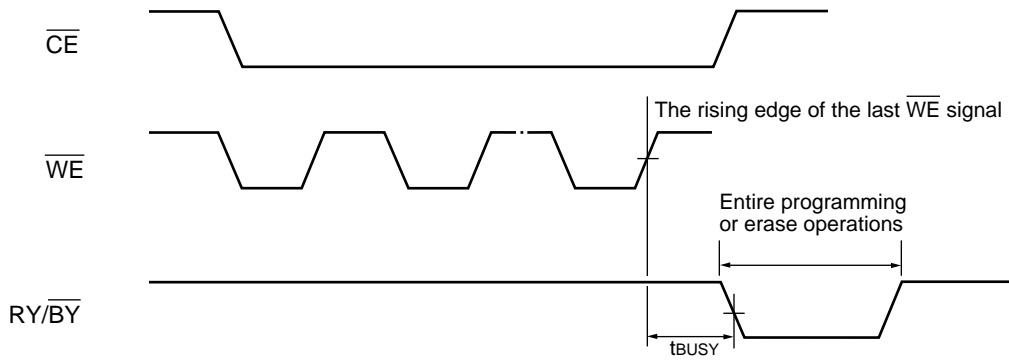
**Figure 11 RY/BY Timing Diagram during Program/Erase Operations**
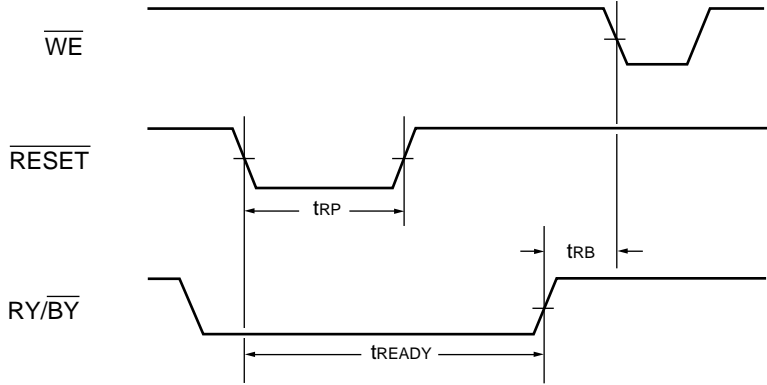


**Figure 12 RESET, RY/BY Timing Diagram**
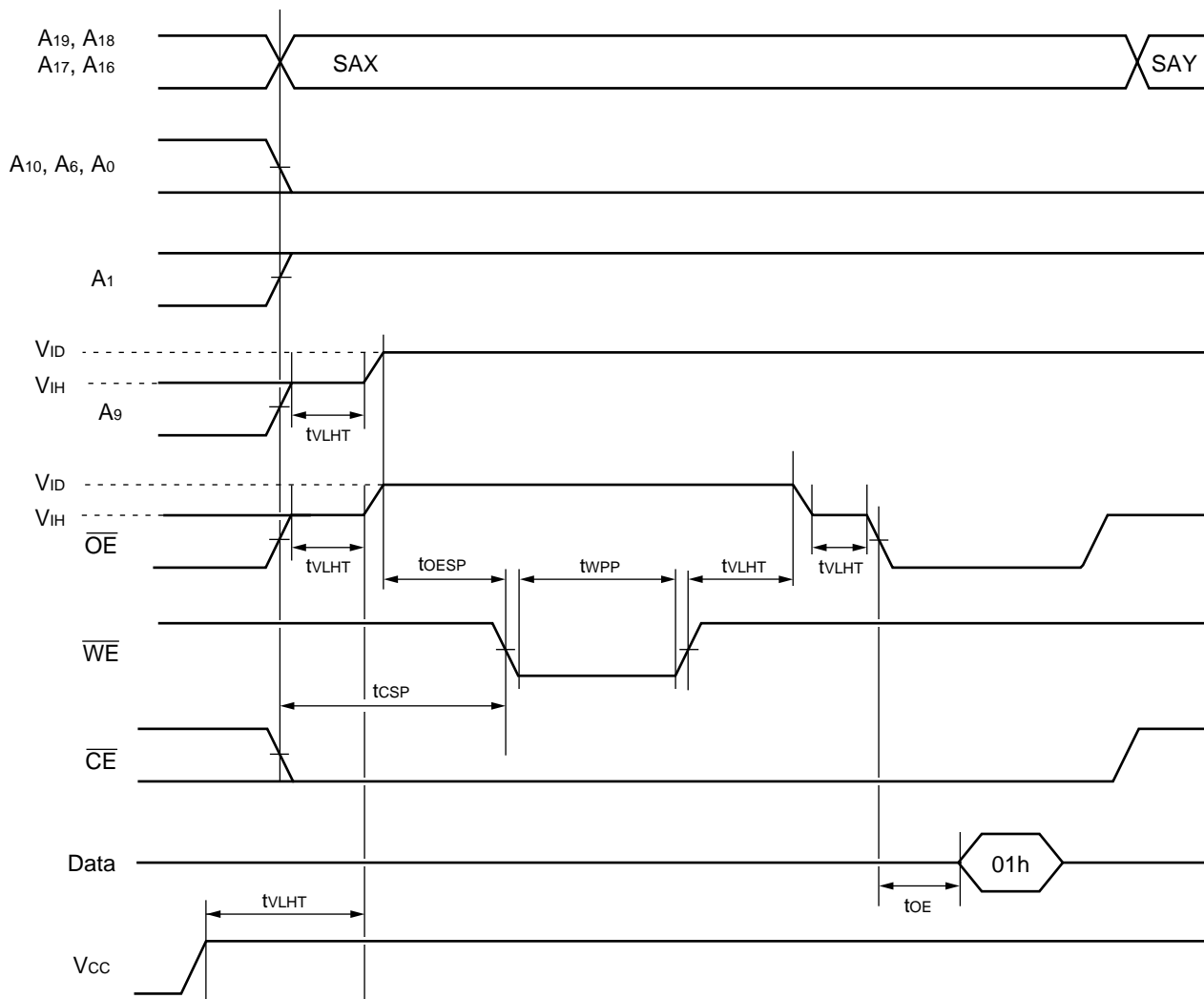
SAX: Sector Address to be protected
SAY: Next Sector Address to be protected

**Figure 13 Sector Protection Timing Diagram**
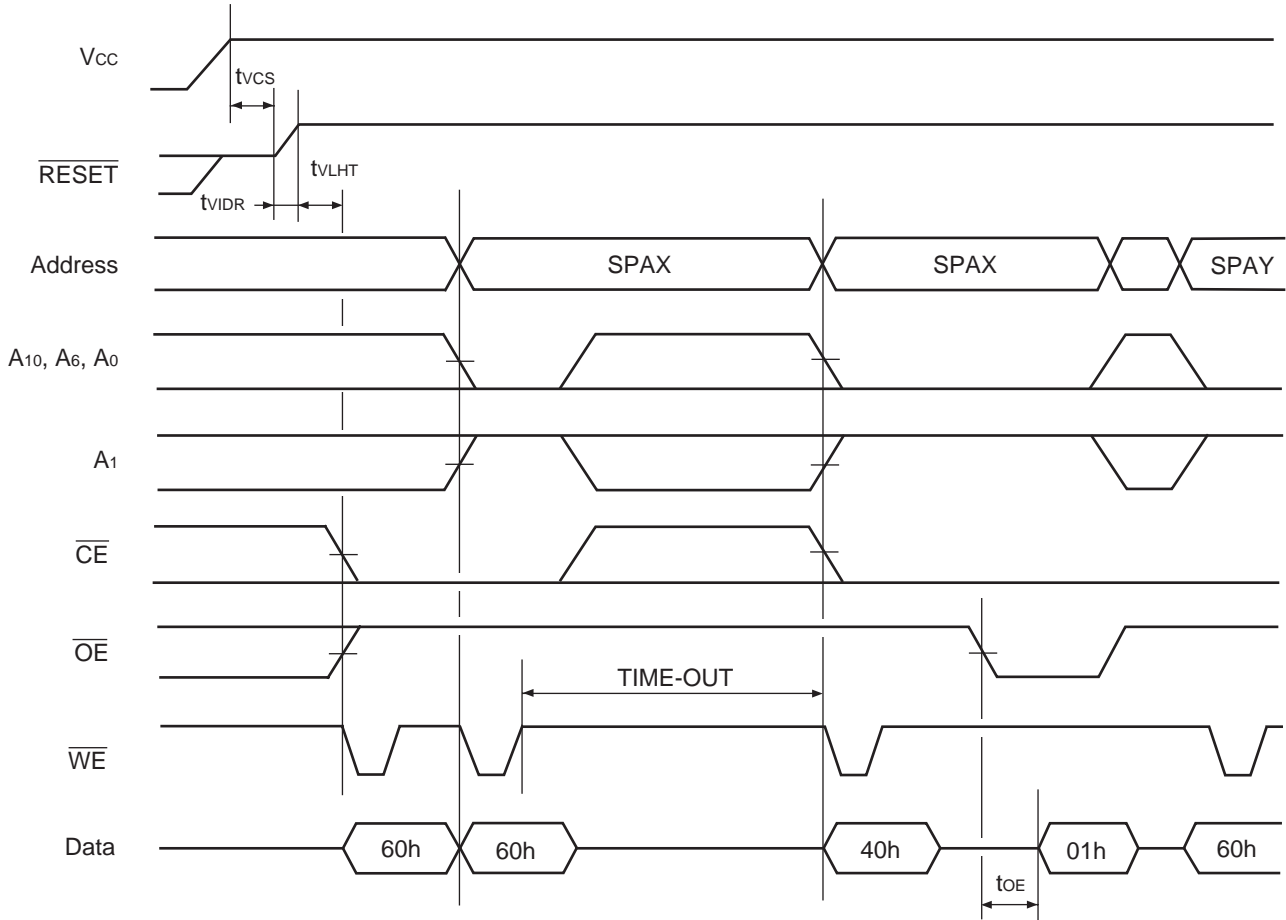
SPAX: Sector Address to be protected
SPAY: Next Sector Address to be protected
TIME-OUT: Time-Out window = 250 μs (Min.)

**Figure 14 Extended Sector Protection Timing Diagram**

**Figure 15 Temporary Sector Unprotection Timing Diagram**



Note: $DQ_2$ is read from the erase-suspended sector.

**Figure 16 $DQ_2$ vs. $DQ_6$**

36
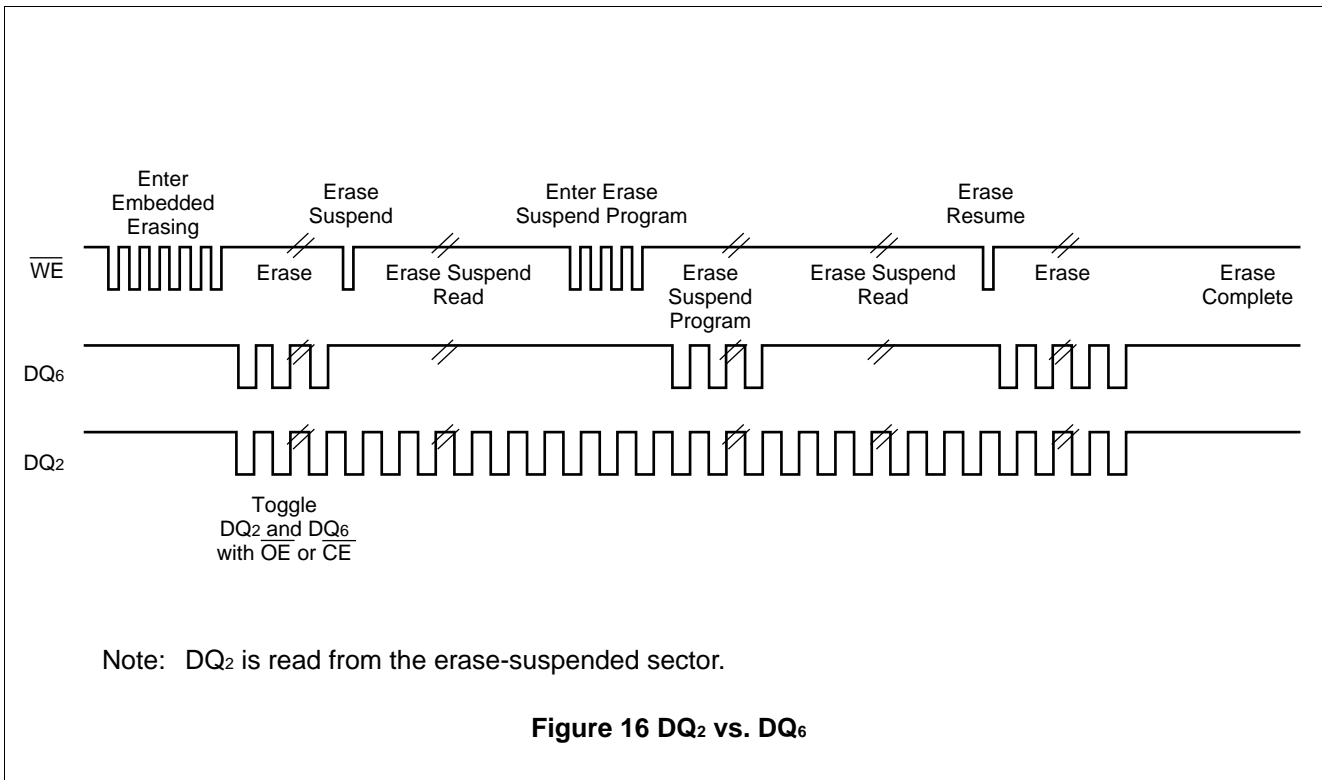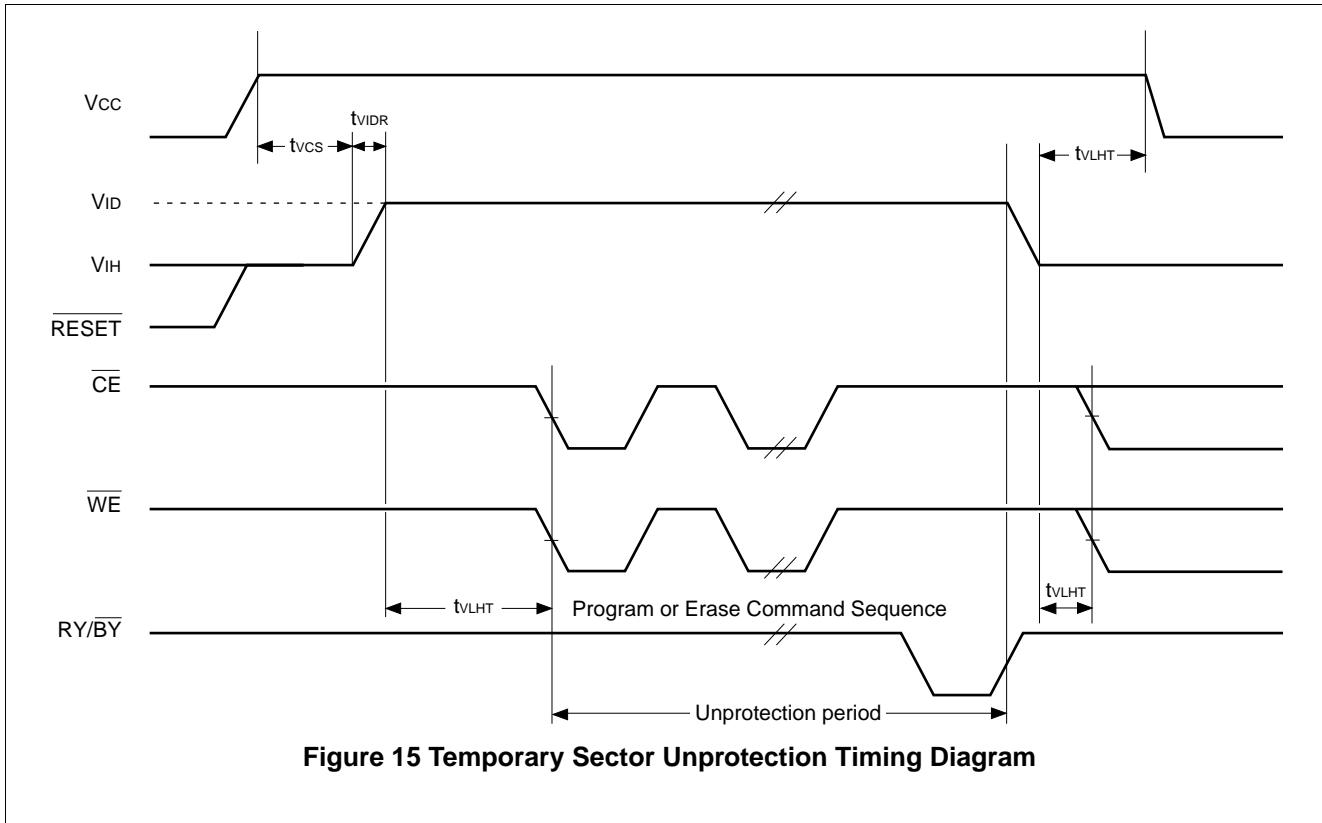
## ■ FLOW CHART

**EMBEDDED ALGORITHM**

```
                        ┌──────────┐
                        │  Start   │
                        └────┬─────┘
                             │
              ┌──────────────▼──────────────┐
      ┌──────▶│   Write Program Command     │
      │       │        Sequence             │
      │       │       (See below)           │
      │       └──────────────┬──────────────┘
      │                      │
      │         ┌────────────▼──────────────┐        ┐
      │    ┌───▶│       Data Polling        │        │ Embeded
      │    │    └────────────┬──────────────┘        │ Program
      │    │                 │                        │ Algorithm
      │    │            ╱────▼────╲                   │ in progress
      │    │   No      ╱ Verify Data ╲                │
      │    └──────────◀   ?           ╲               │
      │               ╲             ╱                 ┘
      │                ╲───┬────╱
      │                    │ Yes
      │               ╱────▼────╲
  ┌───┴──────────┐  No ╱ Last Address ╲
  │  Increment   │◀────   ?           ╲
  │   Address    │     ╲             ╱
  └──────────────┘      ╲───┬────╱
                            │ Yes
                  ┌─────────▼──────────┐
                  │ Programming Completed │
                  └────────────────────┘
```
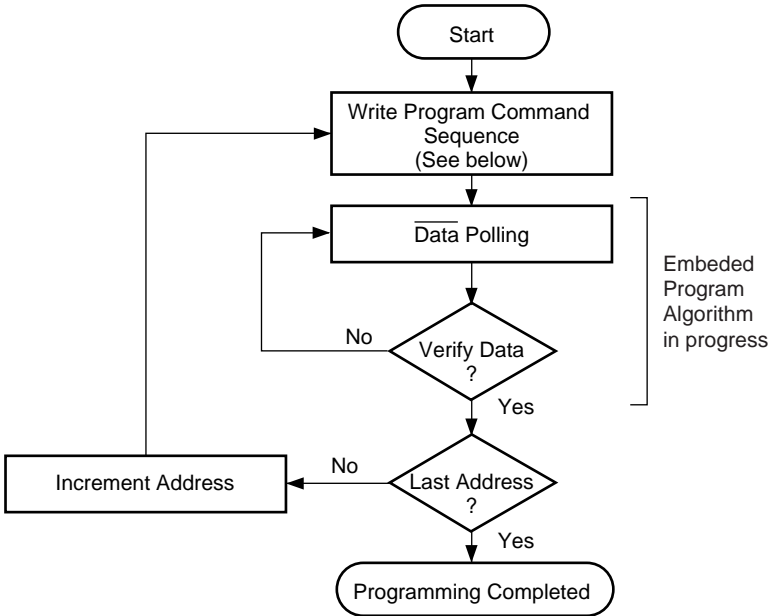
**Program Command Sequence (Address/Command):**

```
          ┌──────────────┐
          │  XXXh/AAh    │
          └──────┬───────┘
                 │
          ┌──────▼───────┐
          │  XXXh/55h    │
          └──────┬───────┘
                 │
          ┌──────▼───────┐
          │  XXXh/A0h    │
          └──────┬───────┘
                 │
    ┌────────────▼─────────────┐
    │ Program Address/Program Data │
    └──────────────────────────┘
```
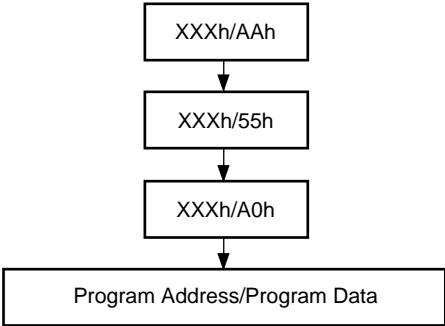
**Figure 17 Embedded Program™ Algorithm**

**EMBEDDED ALGORITHM**
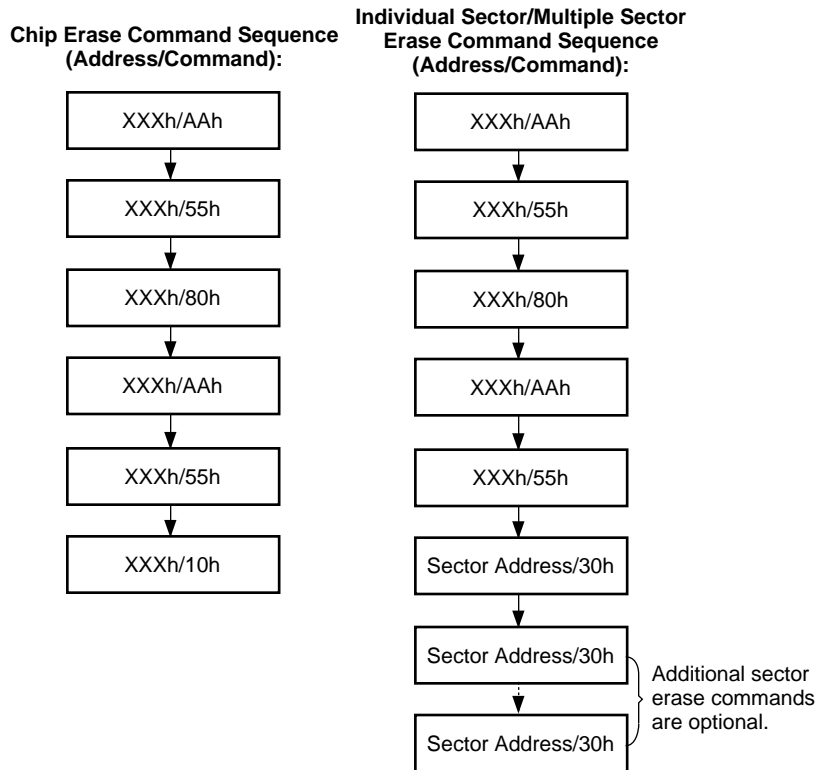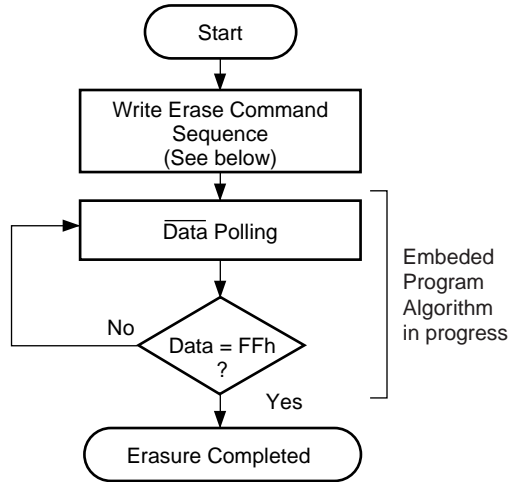


**Figure 18 Embedded Erase™ Algorithm**
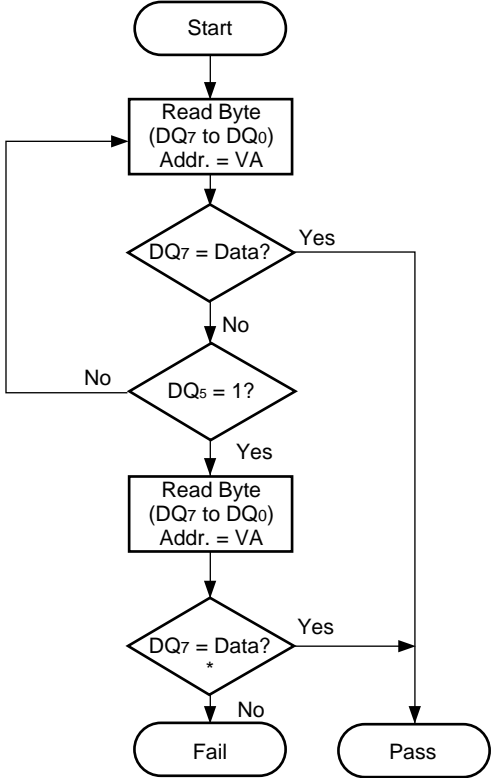
VA = Address for programming
  = Any of the sector addresses within the sector being erased during sector erase or multiple erases operation.
  = Any of the sector addresses within the sector not being protected during sector erase or multiple sector erases operation.

Start

Read Byte
$(DQ_7$ to $DQ_0)$
Addr. = VA

$DQ_7$ = Data?  → Yes

No

$DQ_5$ = 1?  No

Yes

Read Byte
$(DQ_7$ to $DQ_0)$
Addr. = VA

$DQ_7$ = Data?
*  → Yes

No

Fail          Pass

*:  $DQ_7$ is rechecked even if $DQ_5$ = "1" because $DQ_7$ may change simultaneously with $DQ_5$.

**Figure 19 $\overline{\text{Data}}$ Polling Algorithm**

```
                        Start

                   Read DQ7 to DQ0
                                        *1

                   Read DQ7 to DQ0

                    Toggle Bit          No
                    = Toggle
                       ?
                       Yes
         No
                    DQ5 = 1?

                       Yes     *1, *2
                   Read DQ7 to DQ0
                       Twice

                    Toggle Bit          No
                    = Toggle
                       ?
                       Yes

        Program/Erase           Program/Erase
        Operation Not           Operation Complete.
        Complete. Write
        Reset Command
```
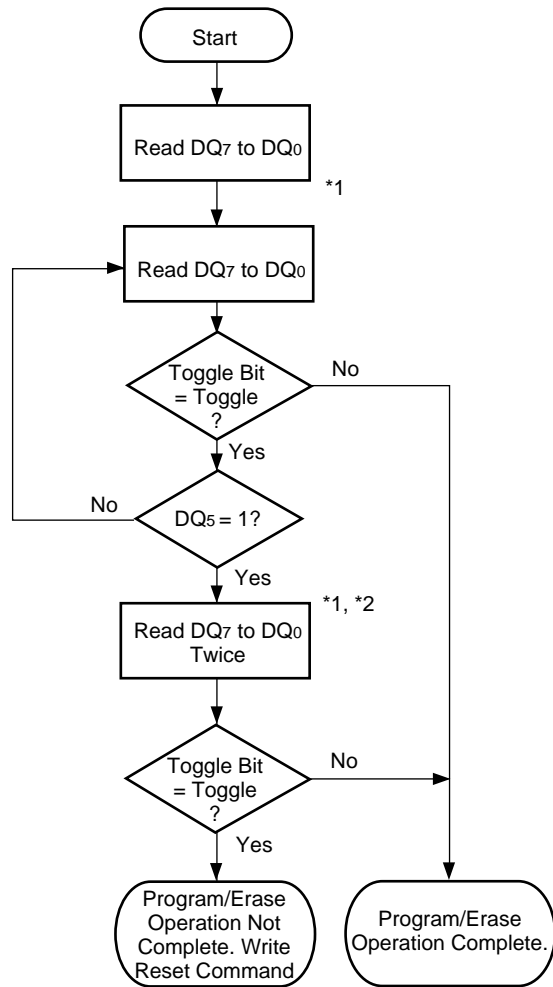
*1: Reset toggle bit twice to determine whether or not it is toggle.

*2: Recheck toggle bit because it may stop toggle as $DQ_5$ changes to "1".

**Figure 20 Toggle Bit Algorithm**

Start

Setup Sector Addr.
(A$_{19}$, A$_{18}$, A$_{17}$, A$_{16}$)

PLSCNT = 1

$\overline{OE}$ = V$_{ID}$, A$_9$ = V$_{ID}$,
A$_6$ = $\overline{CE}$ = V$_{IL}$, $\overline{RESET}$ = V$_{IH}$
A$_0$ = V$_{IL}$, A$_1$ = V$_{IH}$

Activate $\overline{WE}$ Pulse

Time out 100 µs

$\overline{WE}$ = V$_{IH}$, $\overline{CE}$ = $\overline{OE}$ = V$_{IL}$
(A$_9$ should remain V$_{ID}$)

Read from Sector
(A$_1$ = V$_{IH}$, A$_0$ = V$_{IL}$,
Addr. = SA, A$_6$ = V$_{IL}$)

Increment PLSCNT

No

PLSCNT = 25?

No

Data = 01h?

Yes

Yes

Protect Another Sector?

Yes

No

Remove V$_{ID}$ from A$_9$
Write Reset Command

Device Failed

Remove V$_{ID}$ from A$_9$
Write Reset Command

Sector Protection
Completed

**Figure 21 Sector Protection Algorithm**

Start

RESET = $V_{ID}$
*1

Perform Erase or
Program Operations

RESET = $V_{IH}$

Temporary Sector
Unprotection Completed
*2
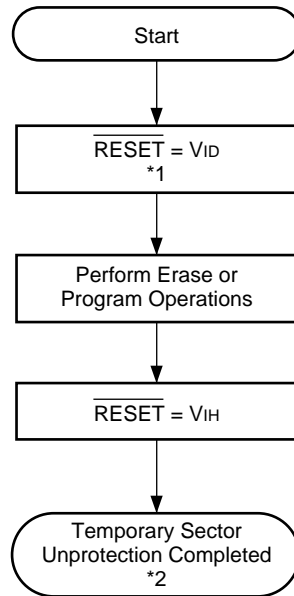
*1: All protected sectors are unprotected.

*2: All previously protected sectors are protected once again.

**Figure 22 Temporary Sector Unprotection Algorithm**

Start

$\overline{\text{RESET}}$ = V$_{ID}$

Wait to 4 μs

Extended Sector Protection Entry?

No → Device is Operating in Temporary Secto Unprotection Mode

Yes

To Setup Sector Protection Write XXXh/60h

PLSCNT = 1

To Sector Protection Write 60h to Sector Address (A$_0$ = V$_{IL}$, A$_1$ = V$_{IH}$, A$_6$ = V$_{IL}$)

Increment PLSCNT

Time out 250 μs

To Verify Sector Protection Write 40h to Sector Address (A$_0$ = V$_{IL}$, A$_1$ = V$_{IH}$, A$_6$ = V$_{IL}$)

Setup Next Sector Address

Read from Sector Address (A$_0$ = V$_{IL}$, A$_1$ = V$_{IH}$, A$_6$ = V$_{IL}$)

PLSCNT = 25?

No ← Data = 01h?

No

Yes

Yes → Protect Other Sector ?

Remove V$_{ID}$ from $\overline{\text{RESET}}$ Write Reset Command

No

Device Failed

Remove V$_{ID}$ from $\overline{\text{RESET}}$ Write Reset Command

Sector Protection Completed

**Figure 23 Extended Sector Protection Algorithm**

**FAST MODE ALGORITHM**

```
                    ┌──────────┐
                    │  Start   │
                    └────┬─────┘
                         ▼
                  ┌─────────────┐              ┐
                  │  XXXh/AAh   │              │
                  └──────┬──────┘              │
                         ▼                     │
                  ┌─────────────┐              │
                  │  XXXh/55h   │               Set Fast Mode
                  └──────┬──────┘              │
                         ▼                     │
                  ┌─────────────┐              │
                  │  XXXh/20h   │              ┘
                  └──────┬──────┘
                         ▼
                  ┌─────────────┐              ┐
         ┌───────▶│  XXXh/A0h   │              │
         │        └──────┬──────┘              │
         │               ▼                     │
         │   ┌─────────────────────────┐       │
         │   │ Program Address/Program Data │   │
         │   └───────────┬─────────────┘       │
         │               ▼                     │
         │        ┌─────────────┐              │
         │        │ Data Polling │◀──┐          │
         │        └──────┬──────┘    │          │
         │               ▼           │          │
         │          ╱Verify╲    No   │           In Fast Program
         │         ╱ Data?  ╲────────┘          │
         │          ╲      ╱                    │
         │           ╲    ╱                     │
         │            Yes                       │
         │             ▼                        │
   ┌──────────┐   No ╱Last    ╲                 │
   │Increment │◀────╱ Address  ╲                │
   │ Address  │     ╲    ?     ╱                │
   └──────────┘      ╲        ╱                 │
                       Yes                      │
                        ▼                       │
               ┌──────────────────┐             │
               │Programming Completed│          ┘
               └────────┬─────────┘
                        ▼
                 ┌─────────────┐               ┐
                 │  XXXh/90h   │               │
                 └──────┬──────┘               │
                        ▼                       Reset Fast Mode
                 ┌─────────────┐               │
                 │  XXXh/F0h   │               ┘
                 └─────────────┘
```
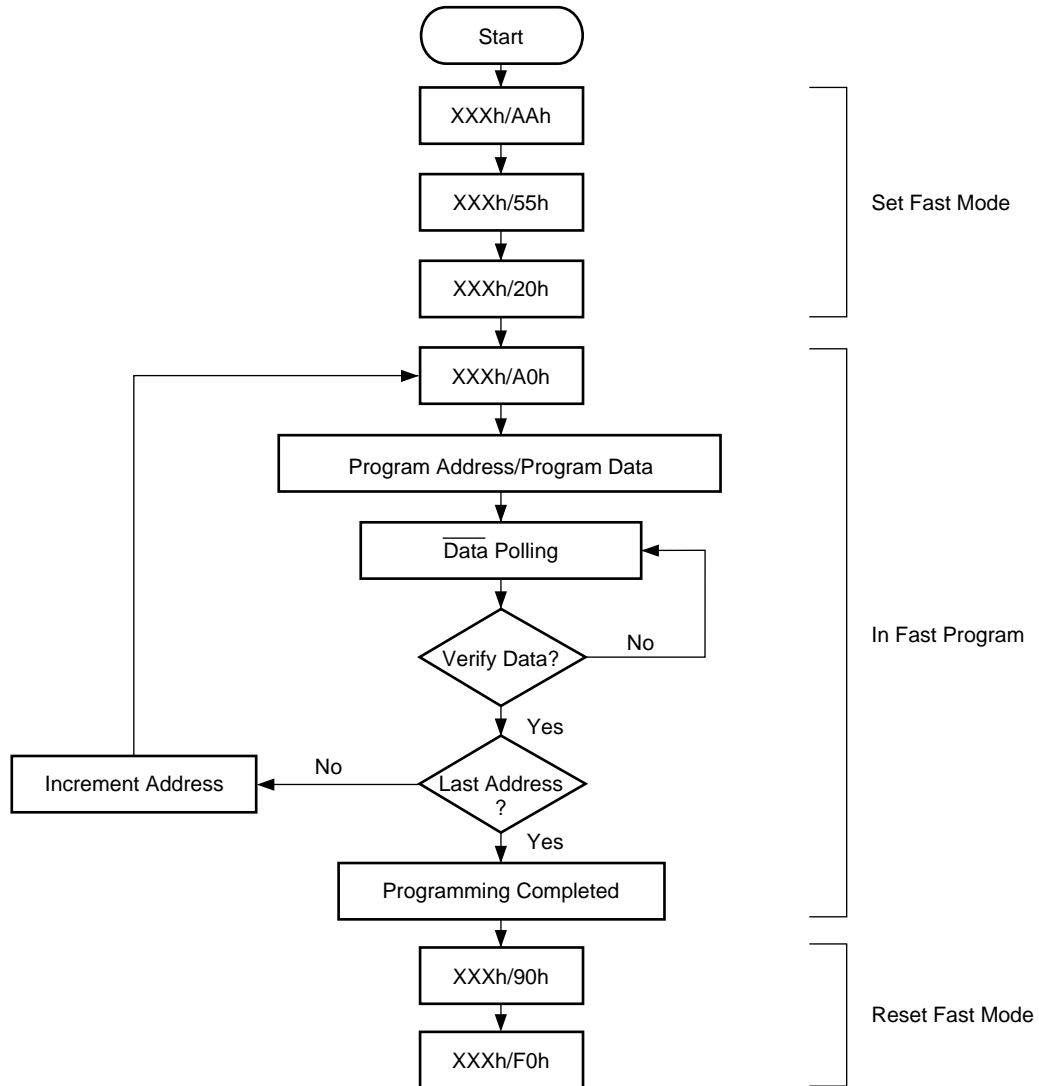
**Figure 24 Embedded Program™ Algorithm for Fast Mode**

■ **ERASE AND PROGRAMMING PERFORMANCE**

| Parameter | Limit | | | Unit | Comments |
|---|---|---|---|---|---|
| | Min. | Typ. | Max. | | |
| Sector Erase Time | — | 1 | 10 | s | Excludes programming time prior to erasure |
| Byte Programming Time | — | 8 | 300 | μs | Excludes system-level overhead |
| Chip Programming Time | — | 8.4 | 25 | s | Excludes system-level overhead |
| Erase/Program Cycle | 100,000 | — | — | cycle | — |

■ **TSOP (I) PIN CAPACITANCE**

| Parameter | Symbol | Test Setup | Value | | Unit |
|---|---|---|---|---|---|
| | | | Typ. | Max. | |
| Input Capacitance | $C_{IN}$ | $V_{IN} = 0$ | 7 | 10 | pF |
| Output Capacitance | $C_{OUT}$ | $V_{OUT} = 0$ | 8 | 10 | pF |
| Control Pin Capacitance | $C_{IN2}$ | $V_{IN} = 0$ | 10 | 12.5 | pF |

Note: Test conditions $T_A = 25°C$, f = 1.0 MHz

## ■ ORDERING INFORMATION

### Standard Products

Fujitsu standard products are available in several packages. The order number is formed by a combination of:

MBM29LV080    A    -70   PTN

PACKAGE TYPE
PTN = 40-pin Thin Small Outline Package
  (TSOP (I)) Normal Bend
PTR = 40-pin Thin Small Outline Package
  (TSOP (I)) Reverse Bend

SPEED OPTION
See Product Selector Guide

DEVICE REVISION

DEVICE NUMBER/DESCRIPTION
MBM29LV080
8Mega-bit (1M × 8-Bit) CMOS Flash Memory
3.0 V-only Read, Program, and Erase
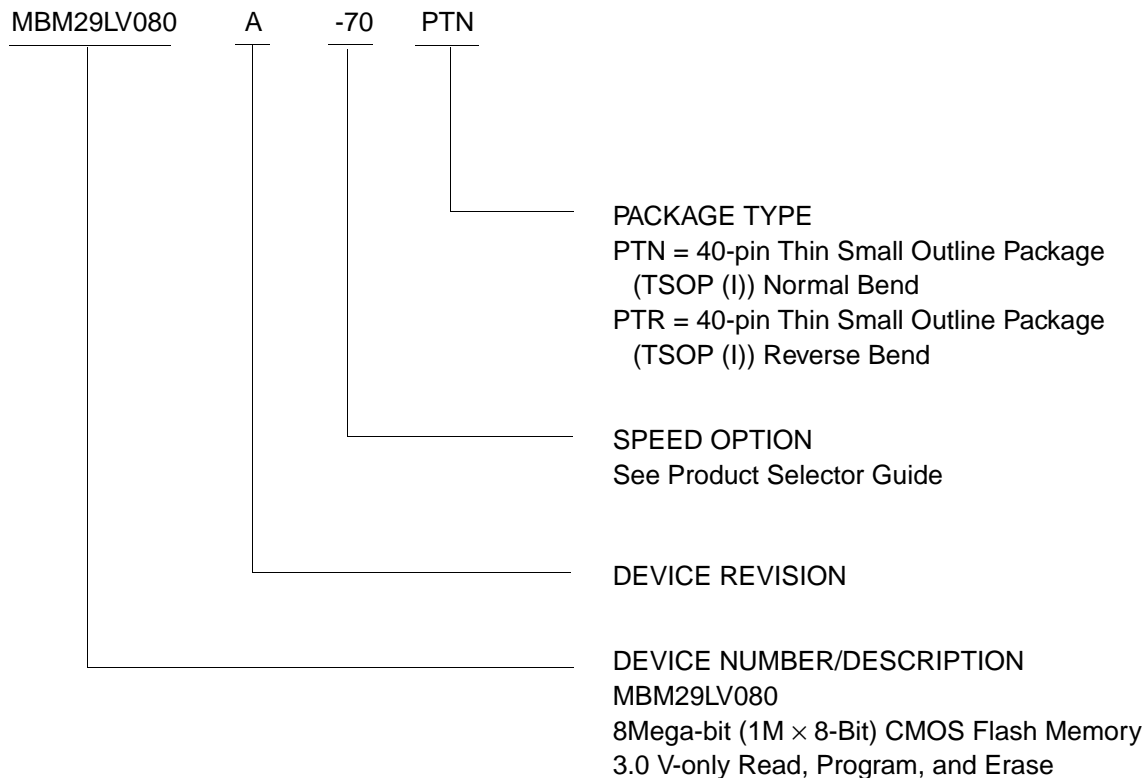
| Part number | Package | Remarks |
|---|---|---|
| MBM29LV080A-70PTV<br>MBM29LV080A-90PTV<br>MBM29LV080A-12PTV | 40-pin plastic TSOP(I)<br>(FPT-40P-M06)<br>Normal Bend | |
| MBM29LV080A-70PTR<br>MBM29LV080A-90PTR<br>MBM29LV080A-12PTR | 40-pin plastic TSOP(I)<br>(FPT-40P-M07)<br>Reverse Bend | |

## ■ PACKAGE DIMENSIONS

### 40-pin plastic TSOP(I) (FPT-40P-M06)

LEAD No. ①　　　　　　　　　　　　　　⑳

INDEX

"A"

⑳　　　　　　　　　　　　　　㉑

Details of "A" part

0.15(.006) MAX
0.35(.014) MAX

0.15(.006)　　0.25(.010)

0.15±0.05
(.006±.002)

20.00±0.20
(.787±.008)

18.40±0.20
(.724±.008)

0.10(.004)

19.00±0.20
(.748±.008)

0.50±0.10
(.020±.004)

10.00±0.20
(.394±.008)

0.05(.002)MIN
(STAND OFF)

$1.10 \, ^{+0.10}_{-0.05} \, (.043 \, ^{+.004}_{-.002})$
(Mounting height)

0.50(.0197)
TYP

9.50(.374)
REF.

0.20±0.10
(.008±.004)

0.10(.004) Ⓜ

© 2000 FUJITSU LIMITED F40007S-1C-2

Dimensions in mm (inches).

### 40-pin plastic TSOP(I) (FPT-40P-M07)

LEAD No. ①　　　　　　　　　　　　　　⑳

INDEX

"A"

⑳　　　　　　　　　　　　　　㉑

Details of "A" part

0.15(.006) MAX
0.35(.014) MAX

0.15(.006)　　0.25(.010)

19.00±0.20
(.748±.008)

0.50±0.10
(.020±.004)

0.15±0.05
(.006±.002)

0.10(.004)

18.40±0.20
(.724±.008)

20.00±0.20
(.787±.008)

0.20±0.10
(.008±.004)

0.10(.004) Ⓜ

9.50(.374)
REF.

0.05(.002)MIN
(STAND OFF)

0.50(.0197)
TYP

10.00±0.20
(.394±.008)

$1.10 \, ^{+0.10}_{-0.05} \, (.043 \, ^{+.004}_{-.002})$
(Mounting height)

© 2000 FUJITSU LIMITED F40008S-1C-2

Dimensions in mm (inches).

# FUJITSU LIMITED