SPANSION™ Flash Memory

Data Sheet

September 2003

This 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. 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.

FLASH MEMORY

CMOS

64M (4M × 16) BIT

MBM29LV652UE90

■ GENERAL DESCRIPTION

The MBM29LV652UE is a 64M-bit, 3.0 V-only Flash memory organized as 4M words of 16 bits each. The device is designed to MBM29LV652UE 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.

To eliminate bus contention the devices have separate chip enable ($\overline{CE}$), write enable ($\overline{WE}$), and output enable ($\overline{OE}$) controls.

The MBM29LV652UE is entirely command set compatible with JEDEC single-power-supply Flash standard. 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.
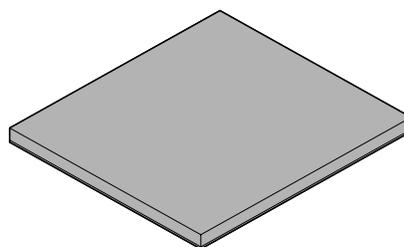
Typically, each sector can be programmed and verified in about 0.5 seconds.

■ PRODUCT LINE UP

Part No.	MBM29LV652UE90
Power Supply Voltage (V)	$V_{CC} = 3.3\ V\ ^{+0.3\ V}_{-0.3\ V}$
Max Address Access Time (ns)	90
Max $\overline{CE}$ Access Time (ns)	90
Max $\overline{OE}$ Access Time (ns)	35

■ PACKAGE

63-ball plastic FBGA

(BGA-63P-M02)

Fujitsu

(Continued)

A sector is typically erased and verified in 1.0 second. (If already completely 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 MBM29LV652UE is erased when shipped from the factory. 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. Once the end of a program or erase cycle has been completed, the devices internally reset to the read mode.
The devices electrically erase all bits within a sector simultaneously via Fowler-Nordhiem tunneling. The words are programmed one word at a time using the EPROM programming mechanism of hot electron injection.

■ FEATURES

- **0.23 μm Process Technology**
- **Single 3.0 V read, program and erase**
  Minimizes system level power requirements
- **Compatible with JEDEC-standards**
  Uses same software commands with single-power supply Flash
- **Address don't care during the command sequence**
- **Industry-standard pinouts**
  63-ball FBGA (Package suffix: PBT)
- **Minimum 100,000 program/erase cycles**
- **High performance**
  90 ns maximum access time
- **Flexible sector architecture**
  One hundred twenty-eight 32K word sectors
  Any combination of sectors can be concurrently erased. Also supports full chip erase
- **HiddenROM region**
  128 word of HiddenROM, accessible through a new "HiddenROM Enable" command sequence
  Factory serialized and protected to provide a secure electronic serial number (ESN)
- **Ready/Busy Output (RY/$\overline{\text{BY}}$)**
  Hardware method for detection of program or erase cycle completion
- **ACC input pin**
  At V_{ACC}, 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
- **$\overline{\text{Data}}$ Polling and Toggle Bit feature 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**
- **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 protect command**
- **Fast Programming Function by Extended Command**

(Continued)

(Continued)
- **Temporary sector group unprotection**
  Temporary sector group unprotection via the $\overline{\text{RESET}}$ pin
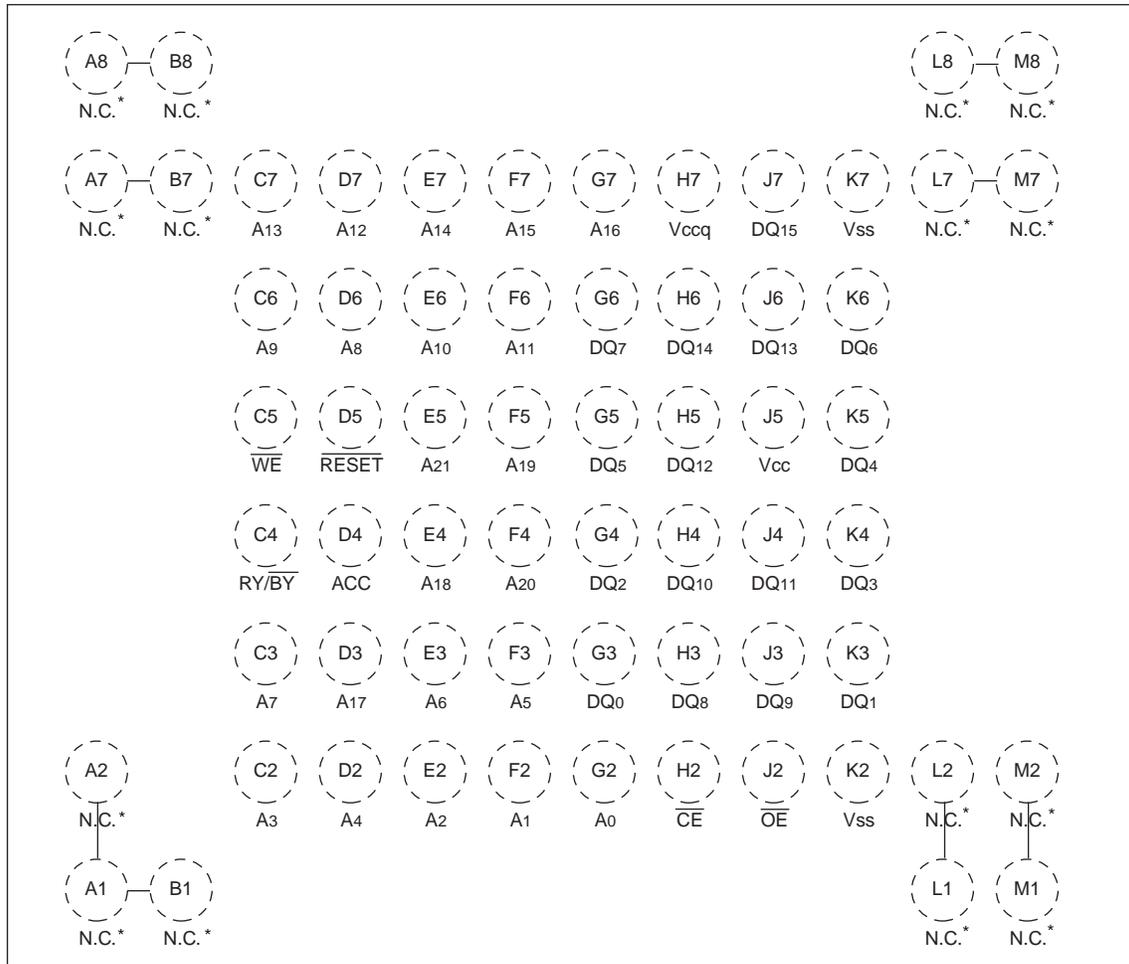 This feature allows code changes in previously locked sectors
- **In accordance with CFI (Common Flash Memory Interface)**

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

■ PIN ASSIGNMENT

FBGA
(TOP VIEW)
Marking Side

A8 N.C.*	B8 N.C.*							L8 N.C.*	M8 N.C.*
A7 N.C.*	B7 N.C.*	C7 A_{13}	D7 A_{12}	E7 A_{14}	F7 A_{15}	G7 A_{16}	H7 Vccq	J7 DQ_{15}	K7 Vss	L7 N.C.*	M7 N.C.*
		C6 A_9	D6 A_8	E6 A_{10}	F6 A_{11}	G6 DQ_7	H6 DQ_{14}	J6 DQ_{13}	K6 DQ_6
		C5 $\overline{WE}$	D5 $\overline{RESET}$	E5 A_{21}	F5 A_{19}	G5 DQ_5	H5 DQ_{12}	J5 Vcc	K5 DQ_4
		C4 $RY/\overline{BY}$	D4 ACC	E4 A_{18}	F4 A_{20}	G4 DQ_2	H4 DQ_{10}	J4 DQ_{11}	K4 DQ_3
		C3 A_7	D3 A_{17}	E3 A_6	F3 A_5	G3 DQ_0	H3 DQ_8	J3 DQ_9	K3 DQ_1
A2 N.C.*		C2 A_3	D2 A_4	E2 A_2	F2 A_1	G2 A_0	H2 $\overline{CE}$	J2 $\overline{OE}$	K2 Vss	L2 N.C.*	M2 N.C.*
A1 N.C.*	B1 N.C.*									L1 N.C.*	M1 N.C.*

(BGA-63P-M02)

* : Peripheral balls on each corner are shorted together via the substrate but not connected to the die.

■ PIN DESCRIPTION

Pin	Function
A_{21} to A_0	Address Inputs
DQ_{15} to DQ_0	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 Group Unprotection
ACC	Program Acceleration
$V_{CC}q$	Output Buffer Power
N.C.	No Internal Connection
V_{SS}	Device Ground
V_{CC}	Device Power Supply

■ BLOCK DIAGRAM
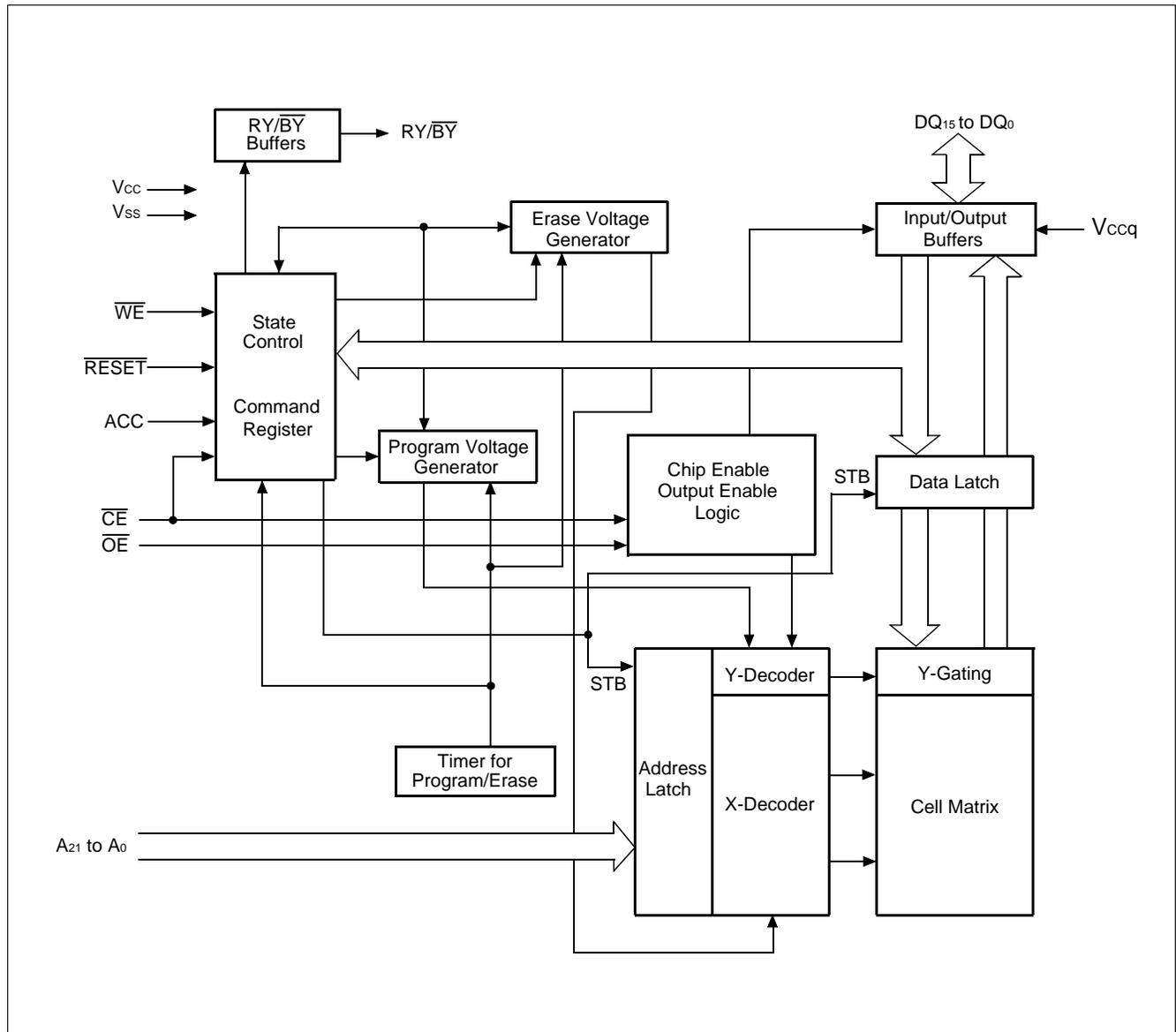
■ LOGIC SYMBOL
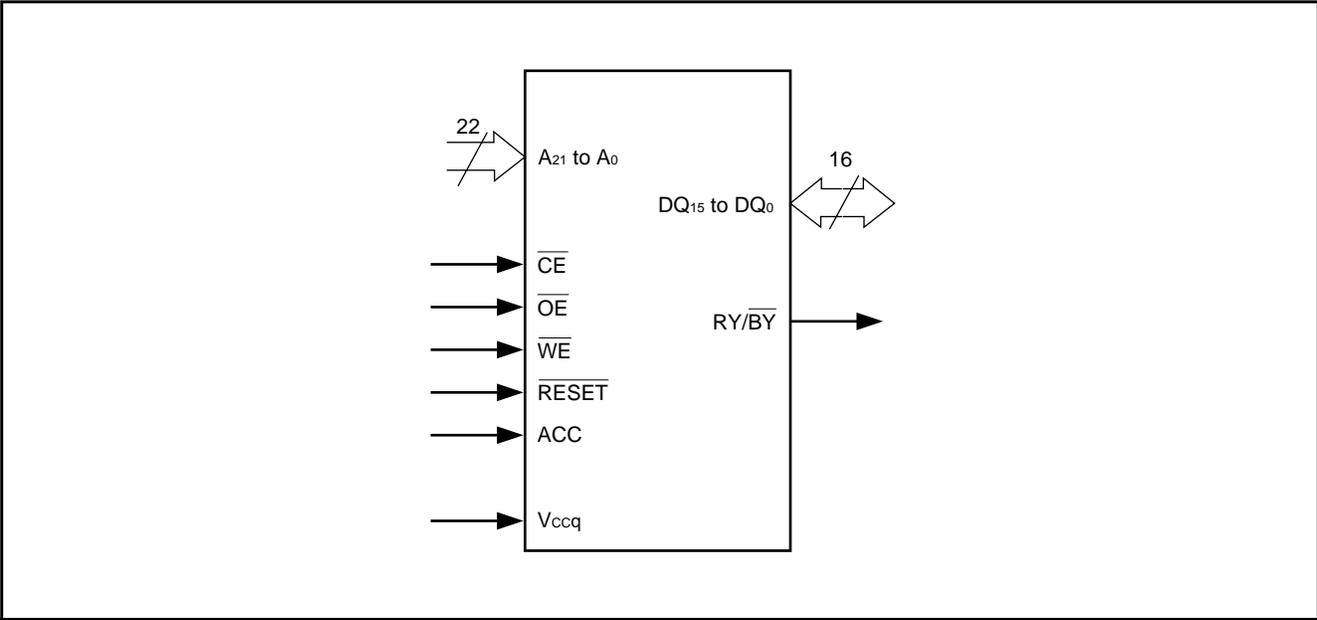
■ DEVICE BUS OPERATION

MBM29LV652UE User Bus Operations Table

Operation	$\overline{CE}$	$\overline{OE}$	$\overline{WE}$	A_0	A_1	A_6	A_9	DQ$_{15}$ to DQ$_0$	$\overline{RESET}$
Auto-Select Manufacture Code *1	L	L	H	L	L	L	V_{ID}	Code	H
Auto-Select Device Code *1	L	L	H	H	L	L	V_{ID}	Code	H
Read *3	L	L	H	A_0	A_1	A_6	A_9	D_{OUT}	H
Standby	H	X	X	X	X	X	X	High-Z	H
Output Disable	L	H	H	X	X	X	X	High-Z	H
Write (Program/Erase)	L	H	L	A_0	A_1	A_6	A_9	D_{IN}	H
Enable Sector Group Protection *2 *4	L	V_{ID}	⌐_⌐	L	H	L	V_{ID}	X	H
Verify Sector Group Protection *2 *4	L	L	H	L	H	L	V_{ID}	Code	H
Temporary Sector Group Unprotection *5	X	X	X	X	X	X	X	X	V_{ID}
Reset (Hardware)/Standby	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 "■ DC CHARACTERISTICS" for voltage levels.

*1: Manufacturer and device codes are accessed via a command register write sequence. See "MBM29LV652UE Command Definitions Table".

*2: Refer to the section on Sector Group Protection.

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

*4: V_{CC} = 3.3 V ± 10%

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

MBM29LV652UE 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	1	XXXh	F0h	—	—	—	—	—	—	—	—	—	—
Read/Reset*1	3	XXXh	AAh	XXXh	55h	XXXh	F0h	RA	RD	—	—	—	—
Autoselect	3	XXXh	AAh	XXXh	55h	XXXh	90h	IA	ID	—	—	—	—
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
Erase Suspend	1	XXXh	B0h	—	—	—	—	—	—	—	—	—	—
Erase Resume	1	XXXh	30h	—	—	—	—	—	—	—	—	—	—
Set to Fast Mode	3	XXXh	AAh	XXXh	55h	XXXh	20h	—	—	—	—	—	—
Fast Program *2	2	XXXh	A0h	PA	PD	—	—	—	—	—	—	—	—
Reset from Fast Mode *2	2	XXXh	90h	XXXh	F0h*6	—	—	—	—	—	—	—	—
Extended Sector Group Protection *3	4	XXXh	60h	SPA	60h	SPA	40h	SPA	SD	—	—	—	—
Query *4	1	XXh	98h	—	—	—	—	—	—	—	—	—	—
HiddenROM Entry	3	XXXh	AAh	XXXh	55h	XXXh	88h	—	—	—	—	—	—
HiddenROM Program *5	4	XXXh	AAh	XXXh	55h	XXXh	A0h	PA	PD	—	—	—	—
HiddenROM Exit *5	4	XXXh	AAh	XXXh	55h	XXXh	90h	XXXh	00h	—	—	—	—

*1 : Both of these reset commands are equivalent.

*2 : This command is valid during FAST Mode.

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

*4 : The valid address are A_6 to A_0.

*5 : This command is valid during HiddenROM mode.

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

Notes : • Address bits = X = "H" or "L" for all address commands except or Program Address (PA) and Sector Address (SA).
  • Bus operations are defined in "MBM29LV652UE User Bus Operations Table".
  • 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 write pulse.
    SA   = Address of the sector to be erased. The combination of A_{21}, A_{20}, A_{19}, A_{18}, A_{17}, A_{16} and A_{15} 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 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.
  • Both Read/Reset commands are functionally equivalent, resetting the device to the read mode.
  • HRBA = Bank Address of the HiddenROM area ($A_{21} = A_{20} = A_{19} = V_{IL}$)
  • The system should be genarated the following address patterns :
        Word mode : 555h or 2AAh to addresses A_{10} to A_0
        Byte mode : AAAh or 555h to addresses A_{10} to A_0, and A_{-1}
  • The command combinations not described in command Definitions table are illegal.

MBM29LV652UE Sector Group Protection Verify Autoselect Codes Table

Type	A_{21} to A_{17}	A_6	A_1	A_0	Code (HEX)
Manufacturer's Code	X	V_{IL}	V_{IL}	V_{IL}	04h
Device Code	X	V_{IL}	V_{IL}	V_{IH}	22D7h
Sector Group Protection	Sector Group Addresses	V_{IL}	V_{IH}	V_{IL}	01h *

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

Extended Autoselect Code Table

Type	Code	DQ_{15}	DQ_{14}	DQ_{13}	DQ_{12}	DQ_{11}	DQ_{10}	DQ_9	DQ_8	DQ_7	DQ_6	DQ_5	DQ_4	DQ_3	DQ_2	DQ_1	DQ_0
Manufacturer's Code	04h	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Device Code	22D7h	0	0	1	0	0	0	1	0	1	1	0	1	0	1	1	1
Sector Group Protection	01h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

■ FLEXIBLE SECTOR-ERASE ARCHITECTURE

Sector Address Table

Sector Address	A_{21}	A_{20}	A_{19}	A_{18}	A_{17}	A_{16}	A_{15}	Sector Size	Address Range
SA0	0	0	0	0	0	0	0	32K words	000000h to 007FFFh
SA1	0	0	0	0	0	0	1	32K words	008000h to 00FFFFh
SA2	0	0	0	0	0	1	0	32K words	010000h to 017FFFh
SA3	0	0	0	0	0	1	1	32K words	018000h to 01FFFFh
SA4	0	0	0	0	1	0	0	32K words	020000h to 027FFFh
SA5	0	0	0	0	1	0	1	32K words	028000h to 02FFFFh
SA6	0	0	0	0	1	1	0	32K words	030000h to 037FFFh
SA7	0	0	0	0	1	1	1	32K words	038000h to 03FFFFh
SA8	0	0	0	1	0	0	0	32K words	040000h to 047FFFh
SA9	0	0	0	1	0	0	1	32K words	048000h to 04FFFFh
SA10	0	0	0	1	0	1	0	32K words	050000h to 057FFFh
SA11	0	0	0	1	0	1	1	32K words	058000h to 05FFFFh
SA12	0	0	0	1	1	0	0	32K words	060000h to 067FFFh
SA13	0	0	0	1	1	0	1	32K words	068000h to 06FFFFh
SA14	0	0	0	1	1	1	0	32K words	070000h to 077FFFh
SA15	0	0	0	1	1	1	1	32K words	078000h to 07FFFFh
SA16	0	0	1	0	0	0	0	32K words	080000h to 087FFFh
SA17	0	0	1	0	0	0	1	32K words	088000h to 08FFFFh
SA18	0	0	1	0	0	1	0	32K words	090000h to 097FFFh
SA19	0	0	1	0	0	1	1	32K words	098000h to 09FFFFh
SA20	0	0	1	0	1	0	0	32K words	0A0000h to 0A7FFFh
SA21	0	0	1	0	1	0	1	32K words	0A8000h to 0AFFFFh
SA22	0	0	1	0	1	1	0	32K words	0B0000h to 0B7FFFh
SA23	0	0	1	0	1	1	1	32K words	0B8000h to 0BFFFFh
SA24	0	0	1	1	0	0	0	32K words	0C0000h to 0C7FFFh
SA25	0	0	1	1	0	0	1	32K words	0C8000h to 0CFFFFh
SA26	0	0	1	1	0	1	0	32K words	0D0000h to 0D7FFFh
SA27	0	0	1	1	0	1	1	32K words	0D8000h to 0DFFFFh
SA28	0	0	1	1	1	0	0	32K words	0E0000h to 0E7FFFh
SA29	0	0	1	1	1	0	1	32K words	0E8000h to 0EFFFFh
SA30	0	0	1	1	1	1	0	32K words	0F0000h to 0F7FFFh
SA31	0	0	1	1	1	1	1	32K words	0F8000h to 0FFFFFh

(Continued)

Sector Address	A_{21}	A_{20}	A_{19}	A_{18}	A_{17}	A_{16}	A_{15}	Sector Size	Address Range
SA32	0	1	0	0	0	0	0	32K words	100000h to 107FFFh
SA33	0	1	0	0	0	0	1	32K words	108000h to 10FFFFh
SA34	0	1	0	0	0	1	0	32K words	110000h to 117FFFh
SA35	0	1	0	0	0	1	1	32K words	118000h to 11FFFFh
SA36	0	1	0	0	1	0	0	32K words	120000h to 127FFFh
SA37	0	1	0	0	1	0	1	32K words	128000h to 12FFFFh
SA38	0	1	0	0	1	1	0	32K words	130000h to 137FFFh
SA39	0	1	0	0	1	1	1	32K words	138000h to 13FFFFh
SA40	0	1	0	1	0	0	0	32K words	140000h to 147FFFh
SA41	0	1	0	1	0	0	1	32K words	148000h to 14FFFFh
SA42	0	1	0	1	0	1	0	32K words	150000h to 157FFFh
SA43	0	1	0	1	0	1	1	32K words	158000h to 15FFFFh
SA44	0	1	0	1	1	0	0	32K words	160000h to 167FFFh
SA45	0	1	0	1	1	0	1	32K words	168000h to 16FFFFh
SA46	0	1	0	1	1	1	0	32K words	170000h to 177FFFh
SA47	0	1	0	1	1	1	1	32K words	178000h to 17FFFFh
SA48	0	1	1	0	0	0	0	32K words	180000h to 187FFFh
SA49	0	1	1	0	0	0	1	32K words	188000h to 18FFFFh
SA50	0	1	1	0	0	1	0	32K words	190000h to 197FFFh
SA51	0	1	1	0	0	1	1	32K words	198000h to 19FFFFh
SA52	0	1	1	0	1	0	0	32K words	1A0000h to 1A7FFFh
SA53	0	1	1	0	1	0	1	32K words	1A8000h to 1AFFFFh
SA54	0	1	1	0	1	1	0	32K words	1B0000h to 1B7FFFh
SA55	0	1	1	0	1	1	1	32K words	1B8000h to 1BFFFFh
SA56	0	1	1	1	0	0	0	32K words	1C0000h to 1C7FFFh
SA57	0	1	1	1	0	0	1	32K words	1C8000h to 1CFFFFh
SA58	0	1	1	1	0	1	0	32K words	1D0000h to 1D7FFFh
SA59	0	1	1	1	0	1	1	32K words	1D8000h to 1DFFFFh
SA60	0	1	1	1	1	0	0	32K words	1E0000h to 1E7FFFh
SA61	0	1	1	1	1	0	1	32K words	1E8000h to 1EFFFFh
SA62	0	1	1	1	1	1	0	32K words	1F0000h to 1F7FFFh
SA63	0	1	1	1	1	1	1	32K words	1F8000h to 1FFFFFh

(Continued)

Sector Address	A_{21}	A_{20}	A_{19}	A_{18}	A_{17}	A_{16}	A_{15}	Sector Size	Address Range
SA64	1	0	0	0	0	0	0	32K words	200000h to 207FFFh
SA65	1	0	0	0	0	0	1	32K words	208000h to 20FFFFh
SA66	1	0	0	0	0	1	0	32K words	210000h to 217FFFh
SA67	1	0	0	0	0	1	1	32K words	218000h to 21FFFFh
SA68	1	0	0	0	1	0	0	32K words	220000h to 227FFFh
SA69	1	0	0	0	1	0	1	32K words	228000h to 22FFFFh
SA70	1	0	0	0	1	1	0	32K words	230000h to 237FFFh
SA71	1	0	0	0	1	1	1	32K words	238000h to 23FFFFh
SA72	1	0	0	1	0	0	0	32K words	240000h to 247FFFh
SA73	1	0	0	1	0	0	1	32K words	248000h to 24FFFFh
SA74	1	0	0	1	0	1	0	32K words	250000h to 257FFFh
SA75	1	0	0	1	0	1	1	32K words	258000h to 25FFFFh
SA76	1	0	0	1	1	0	0	32K words	260000h to 267FFFh
SA77	1	0	0	1	1	0	1	32K words	268000h to 26FFFFh
SA78	1	0	0	1	1	1	0	32K words	270000h to 277FFFh
SA79	1	0	0	1	1	1	1	32K words	278000h to 27FFFFh
SA80	1	0	1	0	0	0	0	32K words	280000h to 287FFFh
SA81	1	0	1	0	0	0	1	32K words	288000h to 28FFFFh
SA82	1	0	1	0	0	1	0	32K words	290000h to 297FFFh
SA83	1	0	1	0	0	1	1	32K words	298000h to 29FFFFh
SA84	1	0	1	0	1	0	0	32K words	2A0000h to 2A7FFFh
SA85	1	0	1	0	1	0	1	32K words	2A8000h to 2AFFFFh
SA86	1	0	1	0	1	1	0	32K words	2B0000h to 2B7FFFh
SA87	1	0	1	0	1	1	1	32K words	2B8000h to 2BFFFFh
SA88	1	0	1	1	0	0	0	32K words	2C0000h to 2C7FFFh
SA89	1	0	1	1	0	0	1	32K words	2C8000h to 2CFFFFh
SA90	1	0	1	1	0	1	0	32K words	2D0000h to 2D7FFFh
SA91	1	0	1	1	0	1	1	32K words	2D8000h to 2DFFFFh
SA92	1	0	1	1	1	0	0	32K words	2E0000h to 2E7FFFh
SA93	1	0	1	1	1	0	1	32K words	2E8000h to 2EFFFFh
SA94	1	0	1	1	1	1	0	32K words	2F0000h to 2F7FFFh
SA95	1	0	1	1	1	1	1	32K words	2F8000h to 2FFFFFh

(Continued)

(Continued)

Sector Address	A_{21}	A_{20}	A_{19}	A_{18}	A_{17}	A_{16}	A_{15}	Sector Size	Address Range
SA96	1	1	0	0	0	0	0	32K words	300000h to 307FFFh
SA97	1	1	0	0	0	0	1	32K words	308000h to 30FFFFh
SA98	1	1	0	0	0	1	0	32K words	310000h to 317FFFh
SA99	1	1	0	0	0	1	1	32K words	318000h to 31FFFFh
SA100	1	1	0	0	1	0	0	32K words	320000h to 327FFFh
SA101	1	1	0	0	1	0	1	32K words	328000h to 32FFFFh
SA102	1	1	0	0	1	1	0	32K words	330000h to 337FFFh
SA103	1	1	0	0	1	1	1	32K words	338000h to 33FFFFh
SA104	1	1	0	1	0	0	0	32K words	340000h to 347FFFh
SA105	1	1	0	1	0	0	1	32K words	348000h to 34FFFFh
SA106	1	1	0	1	0	1	0	32K words	350000h to 357FFFh
SA107	1	1	0	1	0	1	1	32K words	358000h to 35FFFFh
SA108	1	1	0	1	1	0	0	32K words	360000h to 367FFFh
SA109	1	1	0	1	1	0	1	32K words	368000h to 36FFFFh
SA110	1	1	0	1	1	1	0	32K words	370000h to 377FFFh
SA111	1	1	0	1	1	1	1	32K words	378000h to 37FFFFh
SA112	1	1	1	0	0	0	0	32K words	380000h to 387FFFh
SA113	1	1	1	0	0	0	1	32K words	388000h to 38FFFFh
SA114	1	1	1	0	0	1	0	32K words	390000h to 397FFFh
SA115	1	1	1	0	0	1	1	32K words	398000h to 39FFFFh
SA116	1	1	1	0	1	0	0	32K words	3A0000h to 3A7FFFh
SA117	1	1	1	0	1	0	1	32K words	3A8000h to 3AFFFFh
SA118	1	1	1	0	1	1	0	32K words	3B0000h to 3B7FFFh
SA119	1	1	1	0	1	1	1	32K words	3B8000h to 3BFFFFh
SA120	1	1	1	1	0	0	0	32K words	3C0000h to 3C7FFFh
SA121	1	1	1	1	0	0	1	32K words	3C8000h to 3CFFFFh
SA122	1	1	1	1	0	1	0	32K words	3D0000h to 3D7FFFh
SA123	1	1	1	1	0	1	1	32K words	3D8000h to 3DFFFFh
SA124	1	1	1	1	1	0	0	32K words	3E0000h to 3E7FFFh
SA125	1	1	1	1	1	0	1	32K words	3E8000h to 3EFFFFh
SA126	1	1	1	1	1	1	0	32K words	3F0000h to 3F7FFFh
SA127	1	1	1	1	1	1	1	32K words	3F8000h to 3FFFFFh

Sector Group Address Table

Sector Group Address	A21	A20	A19	A18	A17	Sector Group Size	Sectors
SGA0	0	0	0	0	0	128K words	SA0 to SA3
SGA1	0	0	0	0	1	128K words	SA4 to SA7
SGA2	0	0	0	1	0	128K words	SA8 to SA11
SGA3	0	0	0	1	1	128K words	SA12 to SA15
SGA4	0	0	1	0	0	128K words	SA16 to SA19
SGA5	0	0	1	0	1	128K words	SA20 to SA23
SGA6	0	0	1	1	0	128K words	SA24 to SA27
SGA7	0	0	1	1	1	128K words	SA28 to SA31
SGA8	0	1	0	0	0	128K words	SA32 to SA35
SGA9	0	1	0	0	1	128K words	SA36 to SA39
SGA10	0	1	0	1	0	128K words	SA40 to SA43
SGA11	0	1	0	1	1	128K words	SA44 to SA47
SGA12	0	1	1	0	0	128K words	SA48 to SA51
SGA13	0	1	1	0	1	128K words	SA52 to SA55
SGA14	0	1	1	1	0	128K words	SA56 to SA59
SGA15	0	1	1	1	1	128K words	SA60 to SA63
SGA16	1	0	0	0	0	128K words	SA64 to SA67
SGA17	1	0	0	0	1	128K words	SA68 to SA71
SGA18	1	0	0	1	0	128K words	SA72 to SA75
SGA19	1	0	0	1	1	128K words	SA76 to SA79
SGA20	1	0	1	0	0	128K words	SA80 to SA83
SGA21	1	0	1	0	1	128K words	SA84 to SA87
SGA22	1	0	1	1	0	128K words	SA88 to SA91
SGA23	1	0	1	1	1	128K words	SA92 to SA95
SGA24	1	1	0	0	0	128K words	SA96 to SA99
SGA25	1	1	0	0	1	128K words	SA100 to SA103
SGA26	1	1	0	1	0	128K words	SA104 to SA107
SGA27	1	1	0	1	1	128K words	SA108 to SA111
SGA28	1	1	1	0	0	128K words	SA112 to SA115
SGA29	1	1	1	0	1	128K words	SA116 to SA119
SGA30	1	1	1	1	0	128K words	SA120 to SA123
SGA31	1	1	1	1	1	128K words	SA124 to SA127

Common Flash Memory Interface Code Table

Description	A_6 to A_0	DQ_{15} to DQ_0	Description	A_6 to A_0	DQ_{15} to DQ_0
Query-unique ASCII string "QRY"	10h 11h 12h	0051h 0052h 0059h	Erase Block Region 2 Information bit 15 to bit 0 : y = Sector number bit 31 to bit 16 : z = Size (zx 256 Byte)	31h 32h 33h 34h	0000h 0000h 0000h 0000h
Primary OEM Command Set 02h: AMD/FJ standard type	13h 14h	0002h 0000h	Query-unique ASCII string "PRI"	40h 41h 42h	0050h 0052h 0049h
Address for Primary Extended Table	15h 16h	0040h 0000h	Major version number, ASCII	43h	0031h
Alternate OEM Command Set (00h = not applicable)	17h 18h	0000h 0000h	Minor version number, ASCII	44h	0031h
Address for Alternate OEM Extended Table	19h 1Ah	0000h 0000h	Address Sensitive Unlock 00h = Required	45h	0001h
V_{CC} Min (write/erase) DQ_7-DQ_4: 1 V, DQ_3-DQ_0: 100 mV	1Bh	0027h	Erase Suspend 02h = To Read & Write	46h	0002h
V_{CC} Max (write/erase) DQ_7-DQ_4: 1 V, DQ_3-DQ_0: 100 mV	1Ch	0036h	Sector Protection 00h = Not Supported X = Number of sectors in per group	47h	0004h
V_{PP} Min voltage	1Dh	0000h	Sector Temporary Unprotection 01h = Supported	48h	0001h
V_{PP} Max voltage	1Eh	0000h	Sector Protection Algorithm	49h	0004h
Typical timeout per single byte/word write 2^N μs	1Fh	0004h	Number of Sector for Bank 2 00h = Not Supported	4Ah	0000h
Typical timeout for Min size buffer write 2^N μs	20h	0000h	Burst Mode Type 00h = Not Supported	4Bh	0000h
Typical timeout per individual sector erase 2^N ms	21h	000Ah	Page Mode Type 00h = Not Supported	4Ch	0000h
Typical timeout for full chip erase 2^N ms	22h	0000h	ACC (Acceleration) Supply Minimum 00h = Not Supported, DQ_7-DQ_4: 1 V, DQ_3-DQ_0: 100 mV	4Dh	00B5h
Max timeout for byte/word write 2^N times typical	23h	0005h	ACC (Acceleration) Supply Maximum 00h = Not Supported, DQ_7-DQ_4: 1 V, DQ_3-DQ_0: 100 mV	4Eh	00C5h
Max timeout for buffer write 2^N times typical	24h	0000h			
Max timeout per individual sector erase 2^N times typical	25h	0004h			
Max timeout for full chip erase 2^N times typical	26h	0000h			
Device Size = 2^N byte	27h	0017h			
Flash Device Interface description	28h 29h	0001h 0000h			
Max number of byte in multi-byte write = 2^N	2Ah 2Bh	0000h 0000h			
Number of Erase Block Regions within device	2Ch	0001h			
Erase Block Region 1 Information bit 15 to bit 0 : y = Sector number bit 31 to bit 16 : z = Size (zx 256 Byte)	2Dh 2Eh 2Fh 30h	007Fh 0000h 0000h 0001h			

■ FUNCTIONAL DESCRIPTION

Read Mode

The MBM29LV652UE 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_{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" or "L".

Standby Mode

There are two ways to implement the standby mode on the MBM29LV652UE 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 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 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 MBM29LV652UE data. This mode can be used effectively with an application requesting low power consumption such as handy terminals.

To activate this mode, MBM29LV652UE automatically switch themselves to low power mode when MBM29LV652UE addresses remain stable during access fine 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).

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 MBM29LV652UE 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. The Autoselect command may also be used to check the status of write-protected sectors (see "MBM29LV652UE Sector Group Protection Verify Autoselect Codes Table" and "Extended Autoselect Code Table" in "■ DEVICE BUS OPERATION"). 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_0, A_1, and A_6. (See "MBM29LV652UE User Bus Operations Table" in "■ DEVICE BUS OPERATION".)

The manufacturer and device codes may also be read via the command register, for instances when the MBM29LV652UE is erased or programmed in a system without access to high voltage on the A_9 pin. The

command sequence is illustrated in "MBM29LV652UE Command Definitions Table" in "■ DEVICE BUS OPER-ATION". (Refer to Autoselect Command section.)

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 (MBM29LV652UE = 22D7h).These two words are given in the tables "MBM29LV652UE Sector Group Protection Verify Autoselect Codes Table" and "Extended Autoselect Code Table" in "■ DEVICE BUS OPERATION". 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 "MBM29LV652UE Sector Group Protection Verify Autoselect Codes Table" and "Extended Autoselect Code Table" in "■ DEVICE BUS OPERATION")

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 Waveforms for specific timing parameters.

Sector Group Protection

The MBM29LV652UE features hardware sector group protection. This feature will disable both program and erase operations in any combination of thirty two sector groups of memory. (See "Sector Group Address Table" 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_{ID} on address pin A_9 and control pin $\overline{OE}$, (suggest V_{ID} = 11.5 V), $\overline{CE} = V_{IL}$ and $A_0 = A_6 = V_{IL}$, $A_1 = V_{IH}$. The sector group addresses (A_{21}, A_{20}, A_{19}, A_{18}, and A_{17}) should be set to the sector to be protected. "Sector Address Table" in "■ FLEXIBLE SECTOR-ERASE ARCHITECTURE" defines the sector address for each of the one hundred twenty-eight (128) individual sectors, and "Sector Group Address Table" in "FLEXIBLE SECTOR-ERASE ARCHITECTURE" defines the sector group address for each of the thirty-two (32) 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 " (11) Sector Group Protection Timing Diagram" in "■ TIMING DIAGRAM" and " (15) 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_{ID} on address pin A_9 with $\overline{CE}$ and $\overline{OE}$ at V_{IL} and $\overline{WE}$ at V_{IH}. Scanning the sector group addresses (A_{21}, A_{20}, A_{19}, A_{18}, and A_{17}) while (A_6, A_1, A_0) = (0, 1, 0) will produce a logical "1" code at device output DQ_0 for a protected sector. Otherwise the device will produce "0" for unprotected sector. In this mode, the lower order addresses, except for A_0, A_1, and A_6 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 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_{21}, A_{20}, A_{19}, A_{18}, and A_{17}) are the desired sector group address will produce a logical "1" at DQ_0 for a protected sector group. See "MBM29LV652UE Sector Group Protection Verify Autoselect Codes Table" and "Extended Autoselect Code Table" in "■ DEVICE BUS OPERATION".

Temporary Sector Group Unprotection

This feature allows temporary unprotection of previously protected sector groups of the MBM29LV652UE devices in order to change data. The Sector Group Unprotection mode is activated by setting the $\overline{RESET}$ pin to high voltage (V_{ID}). During this mode, formerly protected sector groups can be programmed or erased by selecting

the sector group addresses. Once the V_{ID} is taken away from the $\overline{RESET}$ pin, all the previously protected sector groups will be protected again. Refer to " (12) Temporary Sector Group Unprotection Timing Diagram" in "■ TIMING DIAGRAM" and " (6) Temporary Sector Group Unprotection Algorithm" in "■ FLOW CHART".

This temporary sector group unprotect mode is disabled whenever the chip is in the HiddenROM mode. This area can not be programmed within this mode. Moreover once this area is programmed, it is always protected no matter in which mode.

$\overline{RESET}$
Hardware Reset Pin

The MBM29LV652UE devices 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 "t_{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_{READY}" after the $\overline{RESET}$ pin is driven low. Furthermore, once the $\overline{RESET}$ pin goes high, the devices require an additional "t_{RH}" before it will allow read access. When the $\overline{RESET}$ 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.

Accelerated Program Operation

MBM29LV652UE offers accelerated program operation which enables the programming in high speed. If the system asserts V_{ACC} to the ACC pin, the device automatically enters the acceleration mode and the time required for program operation will reduce to about 50%. This function is primarily intended to allow high speed program, so caution is needed as the sector group will temporarily be unprotected.

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

Removing V_{ACC} from the ACC pin returns the device to normal operation. Do not remove V_{ACC} from the ACC pin while programming. See " (14) Accelerated Program Timing Diagram" in "■ TIMING DIAGRAM". Erase opeation during Accelerated Program Operations is strictly prohibited.

■ COMMAND DEFINITIONS

Device operations are selected by writing specific address and data sequences into the command register. Writing incorrect data values or writing them in the improper sequence will reset the devices to the read mode. "MBM29LV652UE 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_7 to DQ_0 and DQ_{15} to DQ_8 bits are ignored.

Read/Reset Command

In order to return from Autoselect mode or Exceeded Timing Limits ($DQ_5 = 1$) 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 the AC Read Characteristics and Waveforms for the specific timing parameters.

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_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 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 address and the Autoselect command. Then the manufacture and device codes can be read from the address, and an actual data of memory cell can be read from the another address.

Following the command write, a read cycle from address XX00h retrieves the manufacture code of 04h. A read cycle from address XX01h returns the device code (MBM29LV652UE = 22D7h).

All manufacturer and device codes will exhibit odd parity with DQ_7 defined as the parity bit. Sector state (protection or unprotection) will be informed by address XX02h. Scanning the sector group addresses (A_{21}, A_{20}, A_{19}, A_{18}, and A_{17}) while (A_6, A_1, A_0) = (0, 1, 0) will produce a logical "1" at device output DQ_0 for a protected sector group. The programming verification should be performed by verify sector group protection on the protected sector. (See "MBM29LV652UE User Bus Operations Table" in "■ DEVICE BUS OPERATION".)

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.

Word Programming

The devices are programmed on a 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), and 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.

21

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.

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_7 ($\overline{Data}$ Polling), and DQ_6 (Toggle Bit). The chip erase begins on the rising edge of the last $\overline{CE}$ or $\overline{WE}$, whichever happens first in the command sequence and terminates when the data on DQ_7 is "1" (See Write Operation Status section.) at which time the device returns to read the mode.

Chip Erase Time; Sector Erase Time $\times$ 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 $\overline{CE}$ or $\overline{WE}$ whichever happens later, while the command (Data $=$ 30h) is latched on the rising edge of $\overline{CE}$ or $\overline{WE}$ which happens first. After time-out of "t_{TOW}" 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 "MBM29LV652UE 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_{TOW}" 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_{TOW}" from the rising edge of last $\overline{CE}$ or $\overline{WE}$ whichever happens first will initiate the execution of the Sector Erase command(s). If another falling edge of $\overline{CE}$ or $\overline{WE}$, whichever happens first occurs within the "t_{TOW}" 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 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 127).

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 DQ$_7$ ($\overline{\text{Data}}$ Polling), and DQ$_6$ (Toggle Bit) or RY/$\overline{\text{BY}}$.

The sector erase begins after the "t$_{TOW}$" time out from the rising edge of $\overline{\text{CE}}$ or $\overline{\text{WE}}$ whichever happens first for the last sector erase command pulse and terminates when the data on DQ$_7$ is "1" (See Write Operation Status section.) 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

" (2) Embedded Erase^{TM} Algorithm" in "■ FLOW CHART" illustrates the Embedded Erase^{TM} 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. Writting 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 addresses are "Don't Care" when writting 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$_{SPD}$" to suspend the erase operation. When the devices have entered the erase-suspended mode, the RY/$\overline{\text{BY}}$ 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 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, 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 RY/$\overline{\text{BY}}$ output pin, $\overline{\text{Data}}$ polling of $\overline{\text{DQ}}_7$ 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.

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

MBM29LV652UE has Fast Mode function. This mode dispenses with the initial two unclock 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. (Refer to " (7) Extended Sector Group Protection Algorithm" 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) Extended Sector Group Protection Algorithm" in "■ FLOW CHART".)

(3) Extended Sector Group Protection

In addition to normal sector group protection, the MBM29LV652UE has Extended Sector Group Protection as extended function. This function enable 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 only $\overline{RESET}$ pin requires V_{ID} for sector group protection in this mode. The extended sector group protection requires V_{ID} on $\overline{RESET}$ pin. 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_{21}, A_{20}, A_{19}, A_{18}, and A_{17}) 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 μs. To verify programming of the protection circuitry, the sector group addresses pins (A_{21}, A_{20}, A_{19}, A_{18}, and A_{17}) 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 " (13) Extended Sector Group Protection Timing Diagram" in "■ TIMING DIAGRAM" and " (8) Embedded Program™ Algorithm for Fast Mode" 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. 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 the CFI code table. 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 programmed, 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 128 words in length. After the system has written the Enter HiddenROM command sequence, it may read the HiddenROM region by using device addresses A_6 to A_0 (A_{14} to A_7 are "00", A_{21} to A_{15} are don't care). That is, the device sends only program command that would normally be sent to the address 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 address.

HiddenROM Entry Command

MBM29LV652UE has 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 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 128words in length. Write the HiddenROM entry command sequence to enter the HiddenROM area. It is called as HiddenROM mode when the HiddenROM area appears. After the system has written the Enter HiddenROM command sequence, it may read the HiddenROM region by using device addresses A_6 to A_0 (A_{14} to A_7 are "00", A_{21} to A_{15} are don't care).

Read/program of the HiddenROM area is possible during HiddenROM mode. Write the HiddenROM reset command sequence to exit the HiddenROM mode.

HiddenROM Program Command

To program the data to the HiddenROM area, write the HiddenROM program command sequence during HiddenROM mode. This command is 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_7 data poling, DQ_6 toggle bit and $RY/\overline{BY}$ pin.

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 to select (A_6, A_1, A_0) = (0,1,0) and HiddenROM area and write the sector group protect command(60h) during the HiddenROM mode. Same command sequence could be used because other then the HiddenROM mode and that is does not apply high voltage to $\overline{RESET}$ pin, it is same as the extension sector group protect in the past. Please refer "Function Explanation **Extended Command** (3) Extentended Sector Group Protection" for details of extention sector group protect setting.

The other is to apply high voltage (V_{ID}) to A_9 and $\overline{OE}$, specify the sector address to select (A_6, A_1, A_0) = (0,1,0) and HiddenROM area, 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 to select 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 agian. Same command sequence could be used for the above method because other then the HiddenROM mode, it is same as the sector group protect in the past. Please refer "Function Explanation Secor Group Protection" for details of sector group protect setting

Once it is protected, protection can not be cancelled, so please pay closest attention.

Write Operation Status

Detailed in "Hardware Sequence Flags", all the status flags that can be used to check the status of the device for current mode operation. During sector erase, the part provides the status flags automatically to the I/O ports. The 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.

Once erase suspend is entered, address sensitivity still applies. If the address of a non-erasing sector (that is, one available for read) is provided, then stored data can be read from the device. If the address of an erasing sector (that is, one unavailable for read) is applied, the device will output its status bits.

Hardware Sequence Flags Table

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 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)	$\overline{DQ_7}$	Toggle	0	0	1*
Exceeded Time Limits	Embedded Program Algorithm		$\overline{DQ_7}$	Toggle	1	0	1
	Embedded Erase Algorithm		0	Toggle	1	1	N/A
	Erase Suspended Mode	Erase Suspend Program (Non-Erase Suspended Sector)	$\overline{DQ_7}$	Toggle	1	0	N/A

*: Successive reads from the erasing or erase-suspend sector will cause DQ$_2$ to toggle. Reading from non-erase suspend sector address will indicate logic "1" at the DQ$_2$ bit.

Notes: • DQ$_0$ and DQ$_1$ are reserve pins for future use.
      • DQ$_4$ is Fujitsu internal use only.

DQ$_7$
$\overline{Data}$ Polling

The MBM29LV652UE devices feature $\overline{Data}$ 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$_7$. 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 $\overline{Data}$ Polling (DQ$_7$) is shown in " (3) $\overline{Data}$ Polling Algorithm" in "■ FLOW CHART".

For programming, the $\overline{Data}$ Polling is valid after the rising edge of fourth write pulse in the four write pulse sequence.

For chip erase and sector erase, the $\overline{Data}$ Polling is valid after the rising edge of the sixth write pulse in the six write pulse sequence. $\overline{Data}$ 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 MBM29LV652UE data pins (DQ$_7$) may change asynchronously while the output enable ($\overline{OE}$) is asserted low. This means that the devices are 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$_6$ to DQ$_0$ may be still invalid. The valid data on DQ$_7$ to DQ$_0$ will be read on the successive read attempts.

The $\overline{Data}$ 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) $\overline{Data}$ Polling during Embedded Algorithm Operation Timing Diagram" in "■ TIMING DIAGRAM" for the $\overline{Data}$ Polling timing specifications and diagram.

DQ$_6$
Toggle Bit I

The MBM29LV652UE 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 ($\overline{\text{OE}}$ 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 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 μs 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 μs 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$_6$ to toggle. In addition, an Erase Suspend/Resume command will cause the DQ$_6$ to toggle.

See Figure " (7) Toggle Bit I during Embedded Algorithm Operation Timing Diagram" in "■ TIMING DIAGRAM" for the Toggle Bit I timing specifications and diagram.

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. $\overline{\text{Data}}$ 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 "MBM29LV652UE User Bus Operations Table" in "■ DEVICE BUS OPERATION".

The DQ$_5$ 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$_7$ bit and DQ$_6$ never stops toggling. Once the devices have exceeded timing limits, the DQ$_5$ 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$_3$
Sector Erase Timer

After the completion of the initial sector erase command sequence the sector erase time-out will begin. DQ$_3$ 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$_3$ 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 $\overline{\text{Data}}$ 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 "Hardware Sequence Flags Table".

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. The behavior of these two status bits, along with that of DQ_7, is summarized as follows:

For example, DQ_2 and DQ_6 can be used together to determine if the erase-suspend-read mode is in progress. (DQ_2 toggles while DQ_6 does not.) See also "Toggle Bit Status Table" and " (8) DQ_2 vs. DQ_6" in "■ TIMING DIAGRAM".

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

Reading Toggle Bits DQ_6 /DQ_2

Whenever the system initially begins reading toggle bit status, it must read DQ_7 to DQ_0 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_7 to DQ_0 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_5 is high (see the section on DQ_5) . 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_5 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_5 has not gone high. The system may continue to monitor the toggle bit and DQ_5 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 (See "Toggle bit algorithm" in "■ FLOW CHART").

Toggle Bit Status Table

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

*1 : Successive reads from the erasing or erase-suspend sector will cause DQ_2 to toggle.

*2 : Reading from non-erase suspend sector address will indicate logic "1" at the DQ_2 bit.

RY/$\overline{BY}$
Ready/Busy

The MBM29LV652UE 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 MBM29LV652UE is 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 " (9) RY/$\overline{BY}$ Timing Diagram during Program/Erase Operation Timing Diagram" and " (10) $\overline{RESET}$, 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_{CC}; multiples of devices can be connected via several RY/$\overline{BY}$ pins in parallel.

Data Protection

The MBM29LV652UE 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 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 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 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, $\overline{CE}$ and $\overline{WE}$ must be a logical zero while $\overline{OE}$ is a logical one.

Power-up Write Inhibit

Power-up of the 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 read mode on power-up.

Sector Protection

Device user is able to protect each sector group individually to store and protect data. Protection circuit voids both write and erase commands that are addressed to protected sectors. Any commands to write or erase addressed to protected sector are ignored (See "Sector Group Protection" in "■ FUNCTIONAL DESCRIPTION".

■ ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Rating		Unit
		Min	Max	
Ambient Temperature with Power Applied	T_A	− 40	+ 85	°C
Storage Temperature	Tstg	− 55	+ 125	°C
Voltage with respect to Ground All Pins except A_9, $\overline{OE}$, ACC and $\overline{RESET}$ *1, *2	V_{IN}, V_{OUT}	− 0.5	V_{CC} + 0.5	V
Power Supply Voltage *1	V_{CC}	− 0.5	+ 4.0	V
A_9, $\overline{OE}$, ACC, and $\overline{RESET}$ *1, *3	V_{IN}	− 0.5	+ 13.0	V
Power Supply Voltage*1	$V_{CC}q$	− 0.2	+ 7.0	V

*1 : Voltage is defined on the basis of V_{SS} = 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_{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.

*3 : Minimum DC input voltage on A_9, $\overline{OE}$, ACC and $\overline{RESET}$ pins is − 0.5 V. During voltage transitions, A_9, $\overline{OE}$, ACC, 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, $\overline{OE}$, ACC, 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 CONDITIONS

Parameter	Symbol	Value		Unit
		Min	Max	
Ambient Temperature	T_A	− 40	+ 85	°C
Power Supply Voltage (V_{CC})*	V_{CC}	+ 3.0	+ 3.6	V
Power Supply Voltage ($V_{CC}q$)	$V_{CC}q$	+ 2.7	+ 3.6	V

* : Voltage is defined on the basis of V_{SS} = GND = 0 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/MAXIMUM UNDERSHOOT
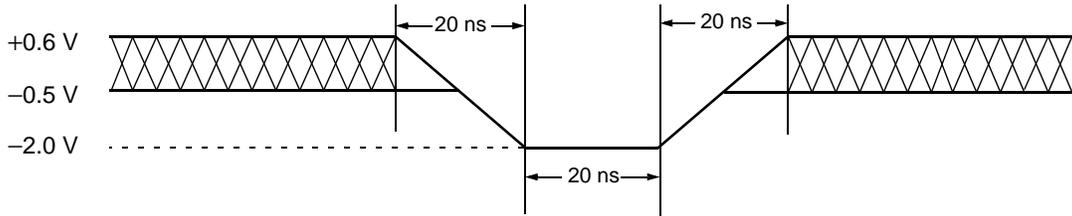
Maximum Undershoot Waveform
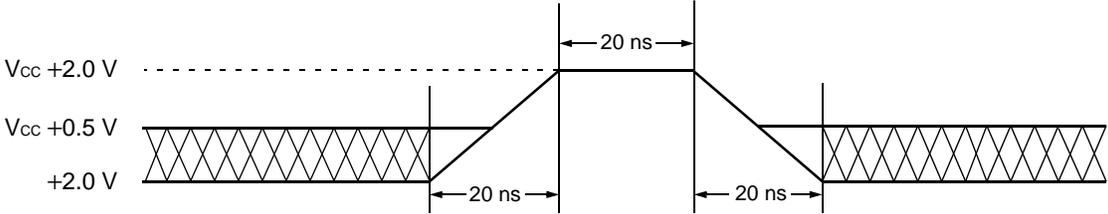
Maximum Overshoot Waveform 1

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

Maximum Overshoot Waveform 2

■ DC CHARACTERISTICS

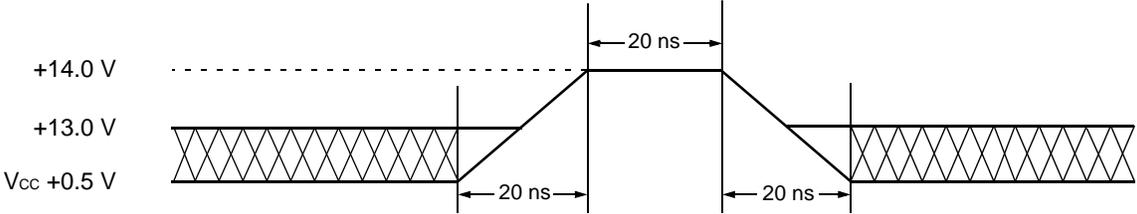| Parameter	Symbol	Test Conditions	Min	Max	Unit
Input Leakage Current	I_{LI}	$V_{IN} = V_{SS}$ to V_{CC}, $V_{CC} = V_{CC}$ Max, $V_{CCq} = V_{CCq}$ Max	− 1.0	+ 1.0	μA
Output Leakage Current	I_{LO}	$V_{OUT} = V_{SS}$ to V_{CC}, $V_{CC} = V_{CC}$ Max, $V_{CCq} = V_{CCq}$ Max	− 1.0	+ 1.0	μA
A_9, $\overline{OE}$, $\overline{RESET}$ 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}	$\overline{CE} = V_{IL}$, $\overline{OE} = V_{IH}$, $V_{CC} = V_{CC}$ Max, $V_{CCq} = V_{CCq}$ Max, f = 5 MHz	—	16	mA
		$\overline{CE} = V_{IL}$, $\overline{OE} = V_{IH}$, $V_{CC} = V_{CC}$ Max, $V_{CCq} = V_{CCq}$ Max, f = 1 MHz	—	7	mA
V_{CC} Active Current*2	I_{CC2}	$\overline{CE} = V_{IL}$, $\overline{OE} = V_{IH}$, $V_{CC} = V_{CC}$ Max, $V_{CCq} = V_{CCq}$ Max	—	40	mA
V_{CC} Current (Standby)	I_{CC3}	$V_{CC} = V_{CC}$ Max, $V_{CCq} = V_{CCq}$ Max, $\overline{CE} = V_{CC} \pm 0.3$ V, $\overline{RESET} = V_{CC} \pm 0.3$ V	—	5	μA
V_{CC} Current (Standby, $\overline{RESET}$)	I_{CC4}	$V_{CC} = V_{CC}$ Max, $V_{CCq} = V_{CCq}$ Max, $\overline{RESET} = V_{SS} \pm 0.3$ V	—	5	μA
V_{CC} Current (Automatic Sleep Mode)*5	I_{CC5}	$V_{CC} = V_{CC}$ Max, $V_{CCq} = V_{CCq}$ 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	—	5	μA
ACC Accelerated Program Current	I_{ACC}	$V_{CC} = V_{CC}$ Max, $ACC = V_{ACC}$ Max	—	20	mA
Input Low Level Voltage	V_{IL}	—	− 0.5	0.6	V
Input High Level Voltage	V_{IH}	—	2.0	$V_{CC} + 0.5$	V
Voltage for Program Acceleration	V_{ACC}	—	11.5	12.5	V
Voltage for Autoselect, Sector Protection (A_9, $\overline{OE}$, $\overline{RESET}$)*3, *4	V_{ID}	—	11.5	12.5	V
Output Low Voltage	V_{OL}	$I_{OL} = 4.0$ mA, $V_{CC} = V_{CC}$ Min, $V_{CCq} = V_{CCq}$ Min	—	0.45	V
Output High Voltage	V_{OH1}	$I_{OH} = − 2.0$ mA, $V_{CC} = V_{CC}$ Min, $V_{CCq} = V_{CCq}$ Min	2.4	—	V
	V_{OH2}	$I_{OH} = − 100$ μA, V_{CC} Min, $V_{CCq} = V_{CCq}$ Min	$V_{CCq} − 0.4$	—	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 : This timing is only for Sector Group Protection operation and Auto select mode.

*4 : Applicable for only V_{CC} applying.
　　Automatic sleep mode enables the low power mode when address remain stable for 150 ns.

*5 : Embedded Algorithm (program or erase) is in progress (@5 MHz) .

■ AC CHARACTERISTICS

• Read Only Operations Characteristics

Parameter	Symbols		Test Setup	Value (Note)		Unit
	JEDEC	Standard		Min	Max	
Read Cycle Time	t_{AVAV}	t_{RC}	—	90	—	ns
Address to Output Delay	t_{AVQV}	t_{ACC}	$\overline{CE} = V_{IL}$, $\overline{OE} = V_{IL}$	—	90	ns
Chip Enable to Output Delay	t_{ELQV}	t_{CE}	$\overline{OE} = V_{IL}$	—	90	ns
Output Enable to Output Delay	t_{GLQV}	t_{OE}	—	—	35	ns
Chip Enable to Output High-Z	t_{EHQZ}	t_{DF}	—	—	30	ns
Output Enable to Output High-Z	t_{GHQZ}	t_{DF}	—	—	30	ns
Output Hold Time From Address, $\overline{CE}$ or $\overline{OE}$, Whichever Occurs First	t_{AXQX}	t_{OH}	—	0	—	ns
$\overline{RESET}$ Pin Low to Read Mode	—	t_{READY}	—	—	20	μs

Note: Test Conditions:
      Output Load: 1 TTL gate and 30 pF (MBM29LV652UE90)
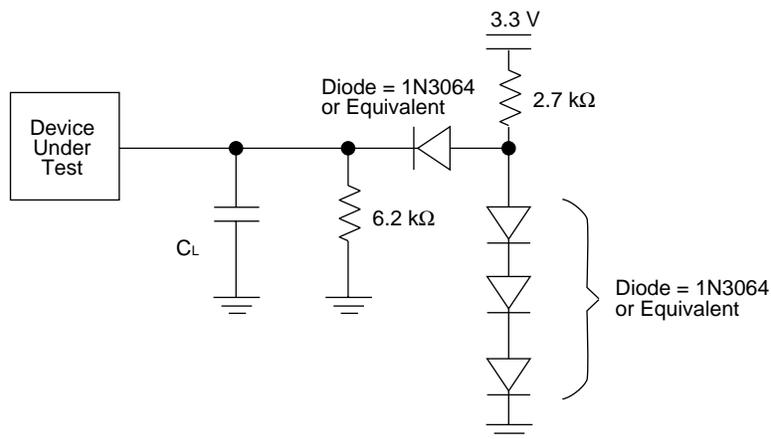   Input rise and fall times: 5 ns
      Input pulse levels: 0.0 V or 3.0 V
      Timing measurement reference level
             Input   : 1.5 V
             Output : 1.5 V

Note : C_L = 30 pF including jig capacitance (MBM29LV652UE90)

Test Conditions

33

• **Write (Erase/Program) Operations**

Parameter		Symbol		Value			Unit
		JEDEC	Standard	Min	Typ	Max	
Write Cycle Time		t_{AVAV}	t_{WC}	90	—	—	ns
Address Setup Time		t_{AVWL}	t_{AS}	0	—	—	ns
Address Hold Time		t_{WLAX}	t_{AH}	45	—	—	ns
Data Setup Time		t_{DVWH}	t_{DS}	35	—	—	ns
Data Hold Time		t_{WHDX}	t_{DH}	0	—	—	ns
Output Enable Setup Time		—	t_{OES}	0	—	—	ns
Output Enable Hold Time	Read	—	t_{OEH}	0	—	—	ns
	Toggle and $\overline{Data}$ Polling			10	—	—	ns
Read Recover Time Before Write		t_{GHWL}	t_{GHWL}	0	—	—	ns
Read Recover Time Before Write		t_{GHEL}	t_{GHEL}	0	—	—	ns
$\overline{CE}$ Setup Time		t_{ELWL}	t_{CS}	0	—	—	ns
$\overline{WE}$ Setup Time		t_{WLEL}	t_{WS}	0	—	—	ns
$\overline{CE}$ Hold Time		t_{WHEH}	t_{CH}	0	—	—	ns
$\overline{WE}$ Hold Time		t_{EHWH}	t_{WH}	0	—	—	ns
Write Pulse Width		t_{WLWH}	t_{WP}	35	—	—	ns
$\overline{CE}$ Pulse Width		t_{ELEH}	t_{CP}	35	—	—	ns
Write Pulse Width High		t_{WHWL}	t_{WPH}	30	—	—	ns
$\overline{CE}$ Pulse Width High		t_{EHEL}	t_{CPH}	30	—	—	ns
Word Programming Operation		t_{WHWH1}	t_{WHWH1}	—	16	—	µs
Sector Erase Operation*[1]		t_{WHWH2}	t_{WHWH2}	—	1	—	s
V_{CC} Setup Time		—	t_{VCS}	50	—	—	µs
Rise Time to V_{ID}*[2]		—	t_{VIDR}	500	—	—	ns
Rise Time to V_{ACC}*[3]		—	t_{VACCR}	500	—	—	ns
Voltage Transition Time*[2]		—	t_{VLHT}	4	—	—	µs
Write Pulse Width*[2]		—	t_{WPP}	100	—	—	µs
$\overline{OE}$ Setup Time to $\overline{WE}$ Active*[2]		—	t_{OESP}	4	—	—	µs
$\overline{CE}$ Setup Time to $\overline{WE}$ Active*[2]		—	t_{CSP}	4	—	—	µs
Recover Time From RY/$\overline{BY}$		—	t_{RB}	0	—	—	ns
$\overline{RESET}$ Pulse Width		—	t_{RP}	500	—	—	ns

(Continued)

(Continued)

Parameter	Symbol		Value			Unit
	JEDEC	Standard	Min	Typ	Max	
$\overline{\text{RESET}}$ Hold Time Before Read	—	t_{RH}	200	—	—	ns
Program/Erase Valid to RY/$\overline{\text{BY}}$ Delay	—	t_{BUSY}	—	—	90	ns
Delay Time from Embedded Output Enable	—	t_{EOE}	—	—	90	ns
Erase Suspend Transition Time	—	t_{SPD}	—	—	20	µs
Erase Time-out Time	—	t_{TOW}	50	—	—	µs

*1 : This does not include the preprogramming time.

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

*3 : This timing is for Accelerated Program operation.

■ ERASE AND PROGRAMMING PERFORMANCE

Parameter	Limit			Unit	Comments
	Min	Typ	Max		
Sector Erase Time	—	1	10	s	Excludes programming time prior to erasure
Programming Time	—	16	360	µs	Excludes system-level overhead
Chip Programming Time	—	—	200	s	Excludes system-level overhead
Erase/Program Cycle	100,000	—	—	cycle	—

■ PIN CAPACITANCE

Parameter	Symbol	Test Setup	Typ	Max	Unit
Input Capacitance	C_{IN}	$V_{IN} = 0$	6.0	7.5	pF
Output Capacitance	C_{OUT}	$V_{OUT} = 0$	8.5	12.0	pF
Control Pin Capacitance	C_{IN2}	$V_{IN} = 0$	8.0	10.0	pF
ACC Pin Capacitance	C_{IN3}	$V_{IN} = 0$	15.0	20.0	pF

Note : Test conditions $T_A = + 25 \,^{\circ}C$, f = 1.0 MHz

■ 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
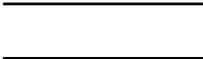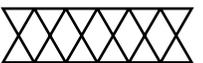
(1) Read Operation Timing Diagram
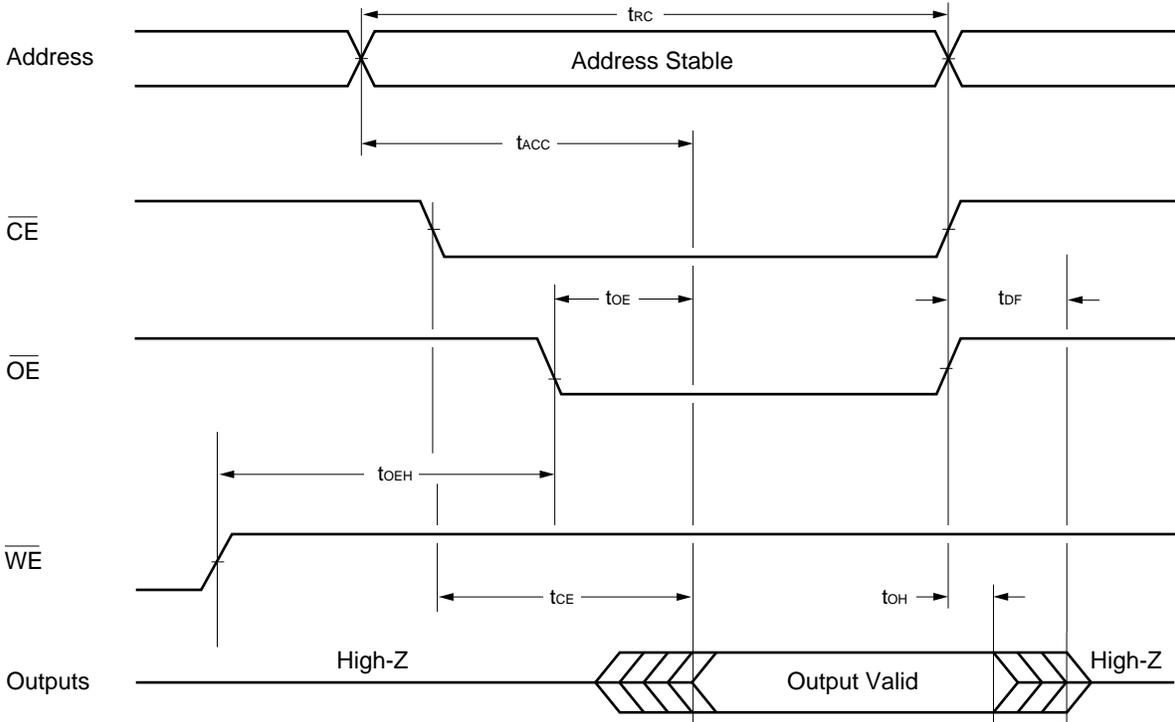
(2) Hardware Reset/Read Operation Timing Diagram
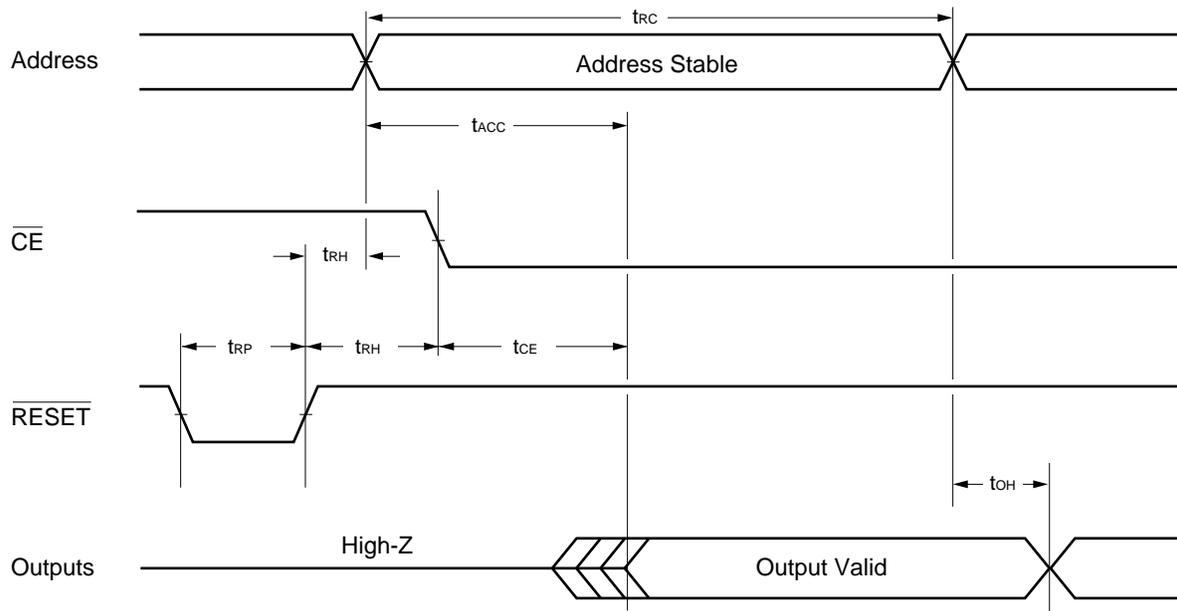
(3) Alternate $\overline{WE}$ Controlled Program Operation Timing Diagram
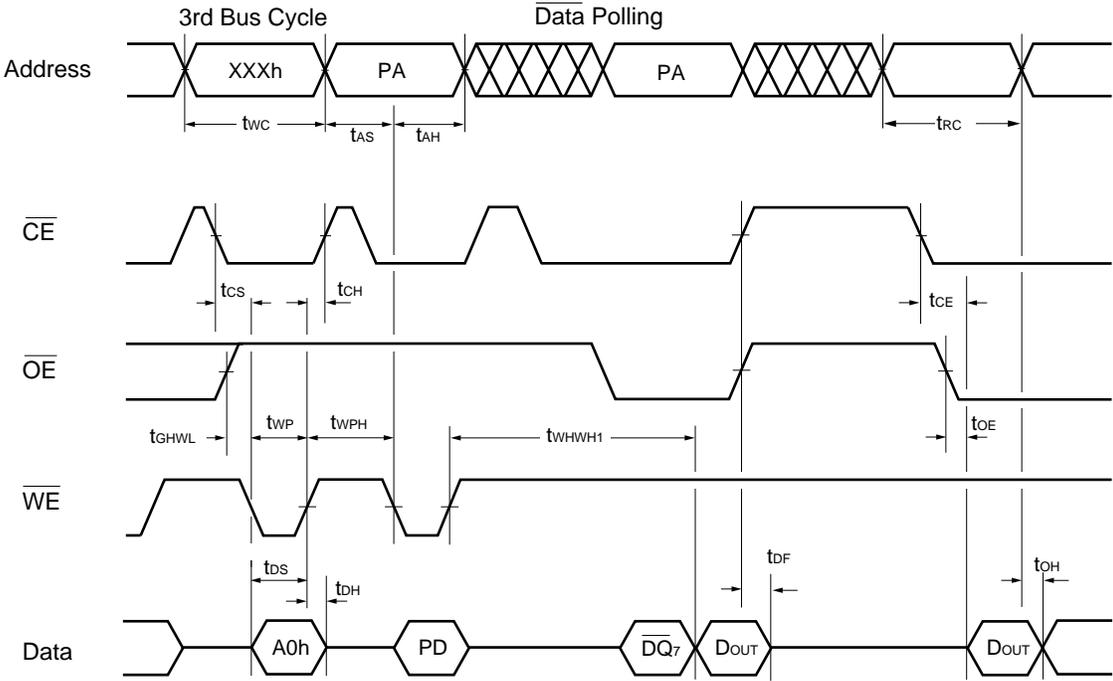
Notes :　• PA is address of the memory location to be programmed.
　　　　• PD is data to be programmed at byte address.
　　　　• $\overline{DQ_7}$ is the output of the complement of the data written to the device.
　　　　• D_{OUT} is the output of the data written to the device.
　　　　• Figure indicates the last two bus cycles out of four bus cycle sequence.

(4) Alternate $\overline{CE}$ Controlled Program Operation Timing Diagram
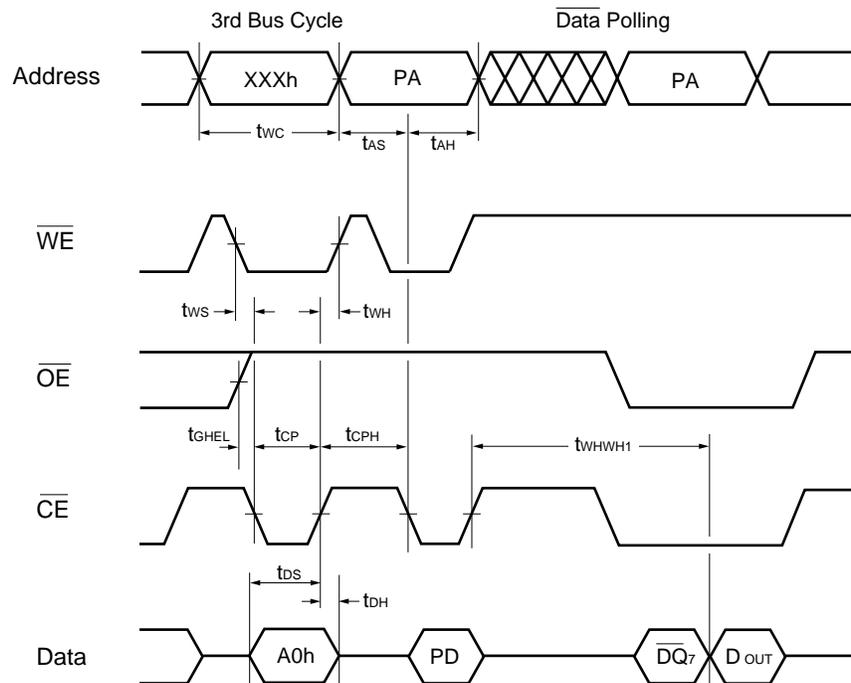
3rd Bus Cycle · $\overline{Data}$ Polling

Address

XXXh · PA · PA

t_{WC} · t_{AS} · t_{AH}

$\overline{WE}$

t_{WS} · t_{WH}

$\overline{OE}$

t_{GHEL} · t_{CP} · t_{CPH} · t_{WHWH1}

$\overline{CE}$

t_{DS} · t_{DH}

Data

A0h · PD · $\overline{DQ_7}$ · D $_{OUT}$

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.

(5) Chip/Sector Erase Operation Timing Diagram
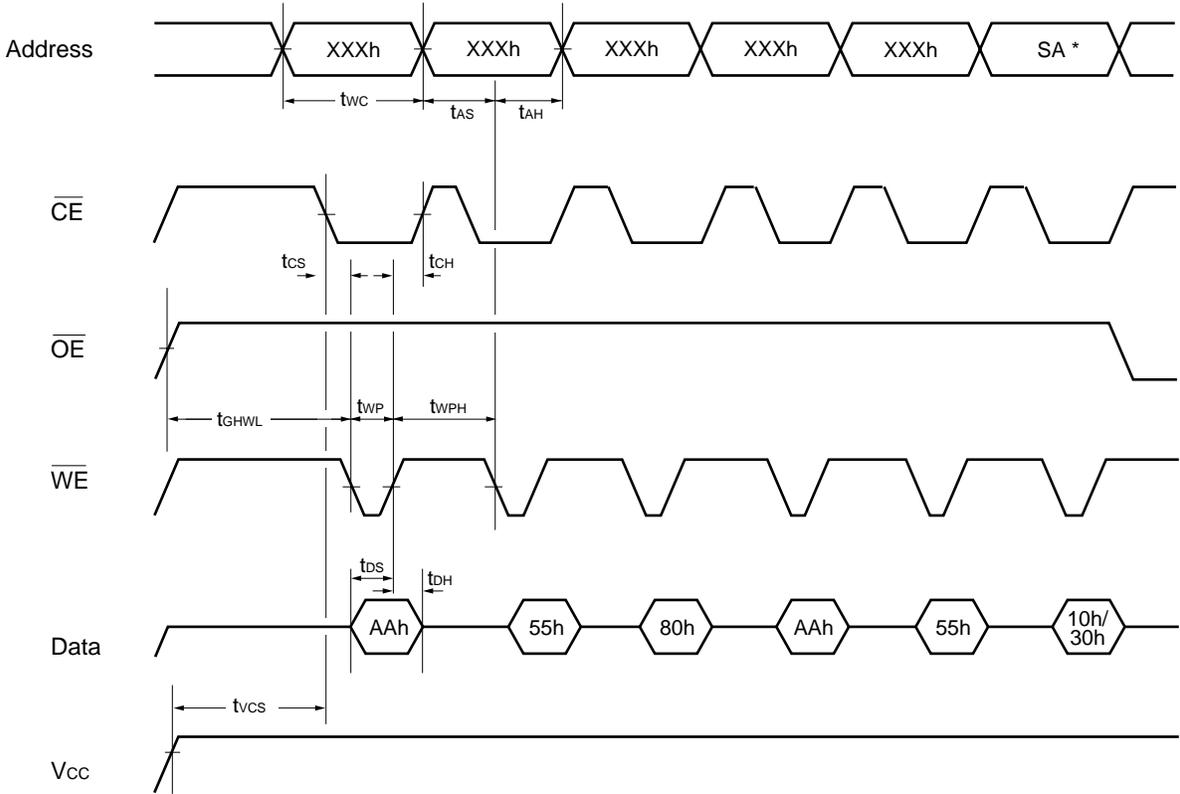
* : SA is the sector address for Sector Erase. Addresses = XXXh for Chip Erase.

(6) $\overline{\text{Data}}$ Polling during Embedded Algorithm Operation Timing Diagram

CE

OE

WE

t_{CH} t_{OE} t_{DF}

t_{OEH}

t_{CE}

*

DQ7 | Data | $\overline{DQ_7}$ | DQ7 = Valid Data | High-Z

$t_{WHWH1 \text{ or } 2}$

DQ6 to DQ0 | Data | DQ6 to DQ0 = Output Flag | DQ6 to DQ0 Valid Data | High-Z

t_{BUSY} t_{EOE}

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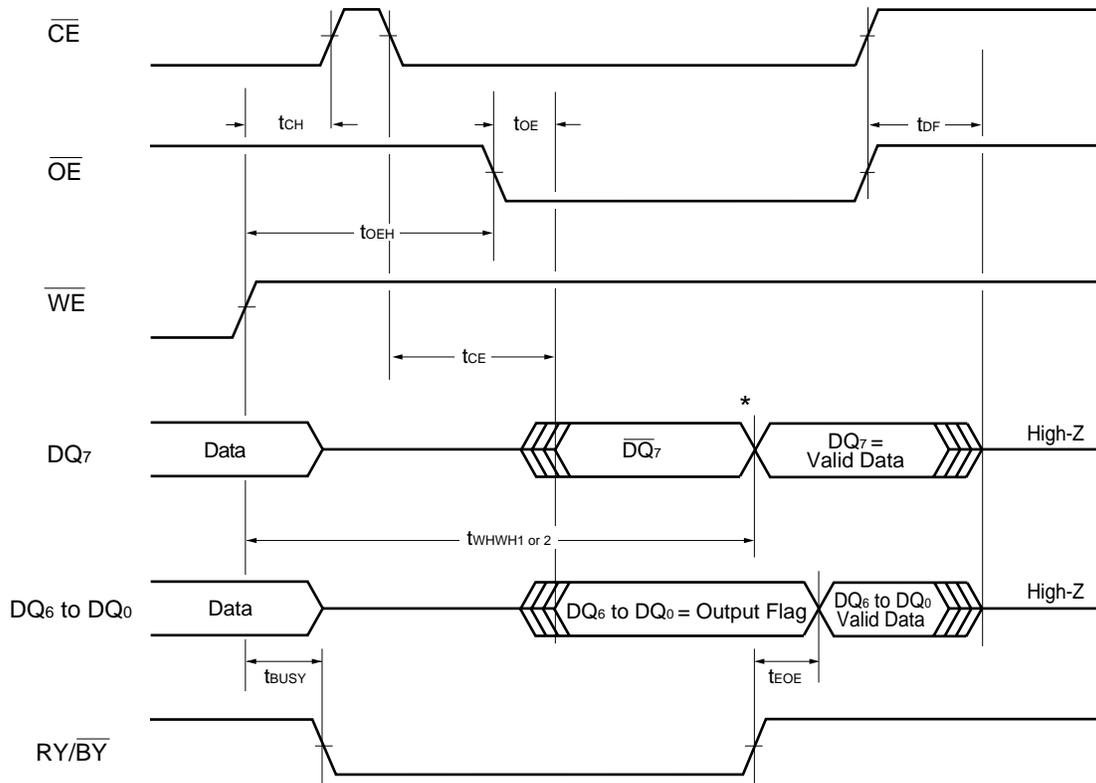
* : DQ7 = Valid Data (The device has completed the Embedded operation.)

(7) Toggle Bit I during Embedded Algorithm Operation Timing Diagram

CE

t_{OEH}

WE

t_{OES}

OE

t_{DH} *

DQ6 | Data (DQ7 to DQ0) | DQ6 = Toggle | DQ6 = Toggle | DQ6 = Stop Toggling | DQ7 to DQ0 Data Valid

t_{OE}
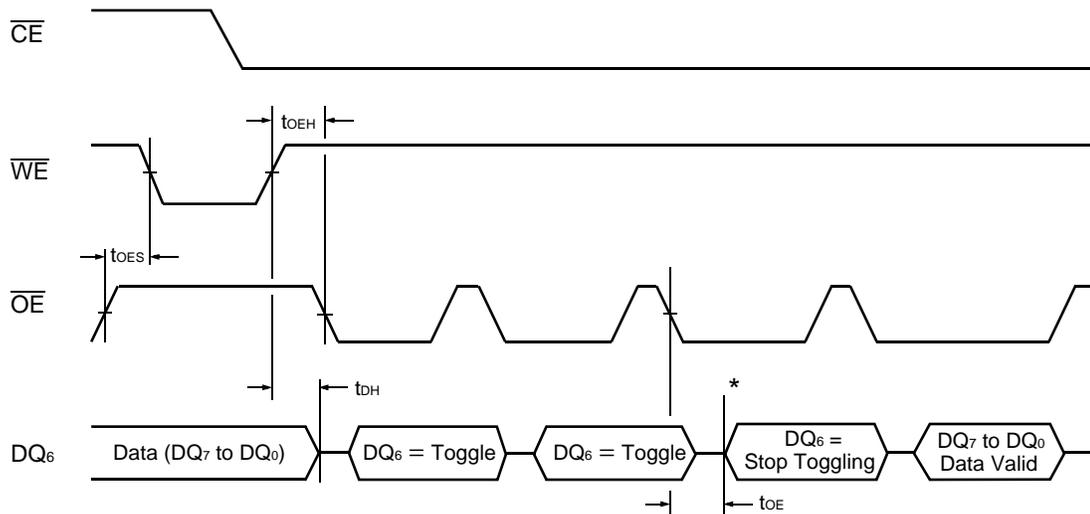
* : DQ6 = Stops toggling. (The device has completed the Embedded operation.)

42

(8) DQ₂ vs. DQ₆

Enter
Embedded
Erasing

Erase
Suspend

Enter Erase
Suspend Program

Erase
Resume

WE   Erase   Erase Suspend   Erase   Erase Suspend   Erase   Erase
                              Read   Suspend   Read   Complete
                                                     Program

DQ₆

DQ₂

Toggle
DQ₂ and DQ₆
with OE or CE
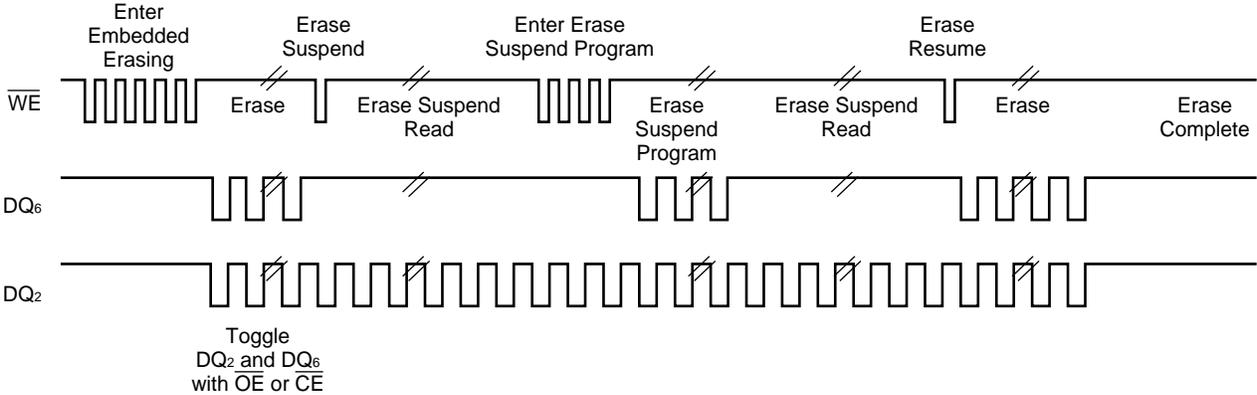
Note : DQ₂ is read from the erase-suspended sector.

(9) RY/BY Timing Diagram during Program/Erase Operation Timing Diagram
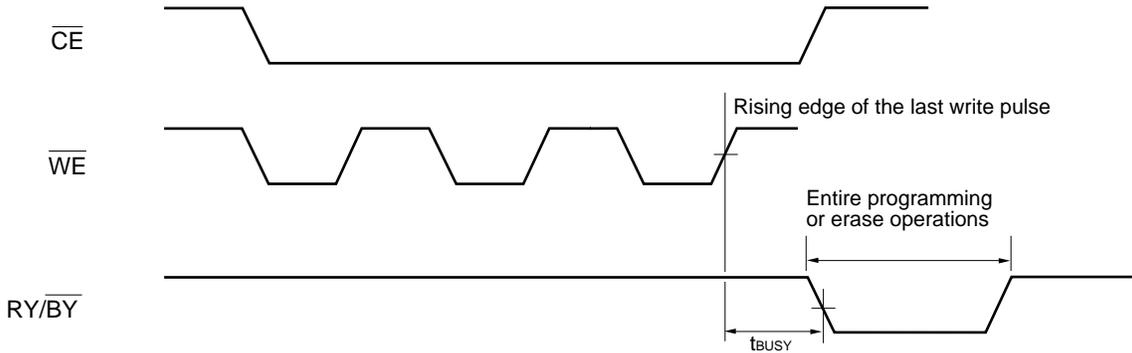
CE

WE

Rising edge of the last write pulse

Entire programming
or erase operations

RY/BY

t_BUSY

(10) RESET,RY/BY Timing Diagram

WE

RESET

t_RP

t_RB

RY/BY

t_READY

43
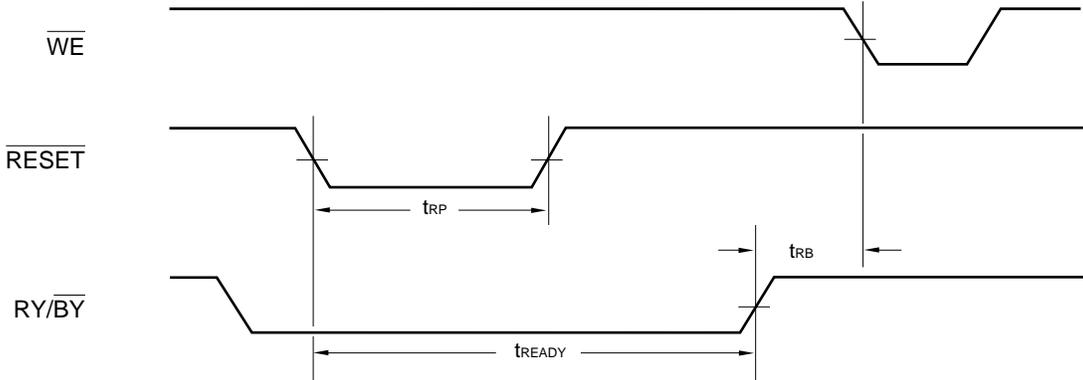
(11) Sector Group Protection Timing Diagram
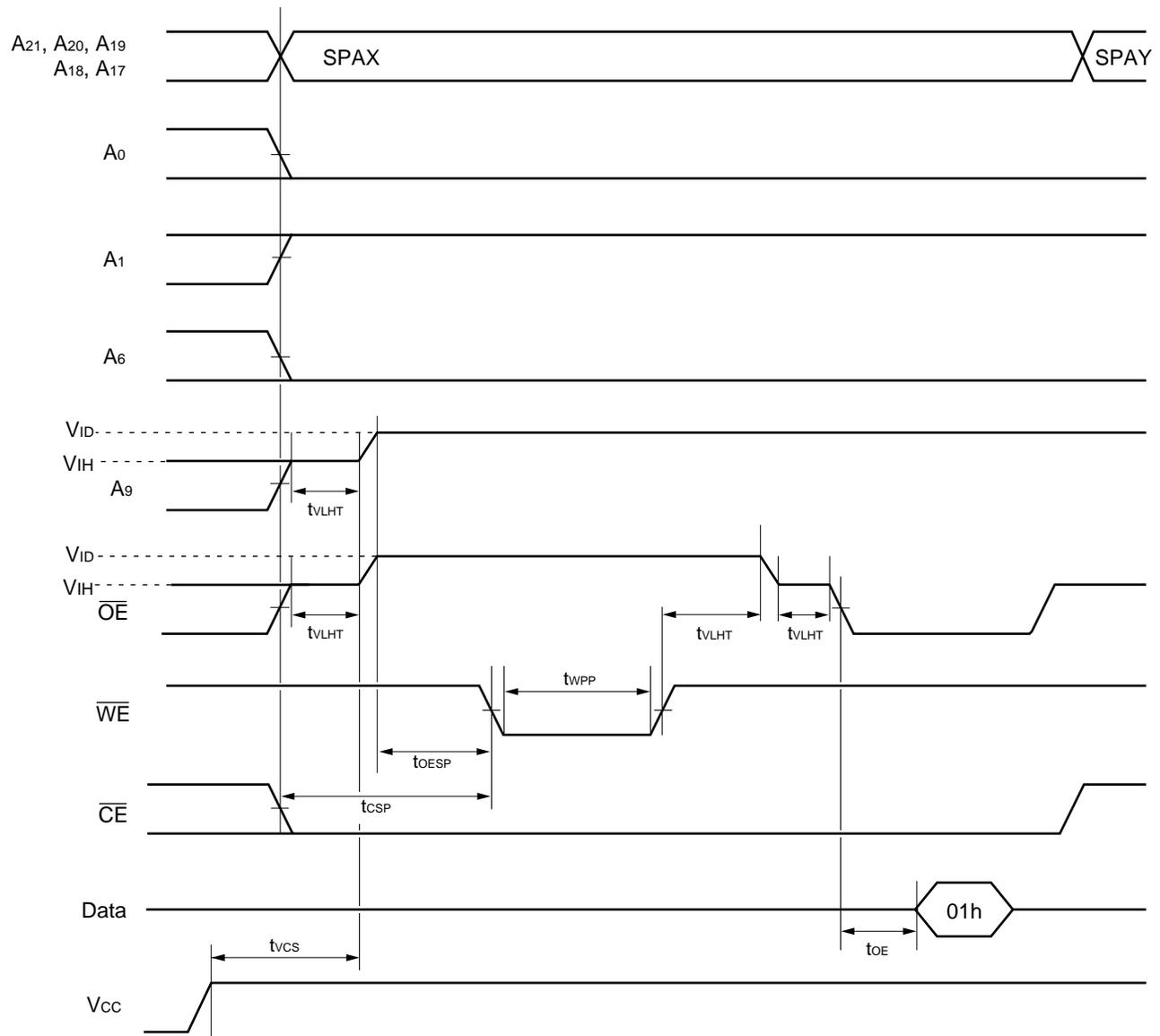
SPAX = Sector Group Address to be protected
SPAY = Sector Group Address to be protected

(12) Temporary Sector Group Unprotection Timing Diagram
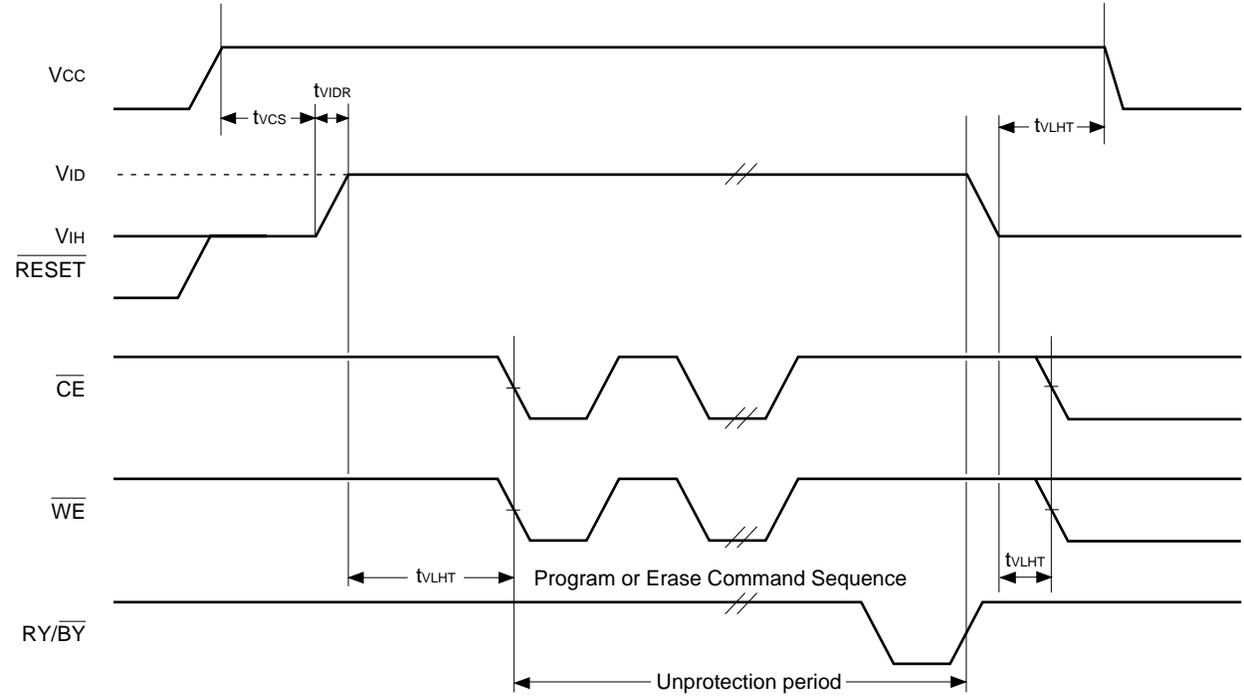
(13) Extended Sector Group Protection Timing Diagram
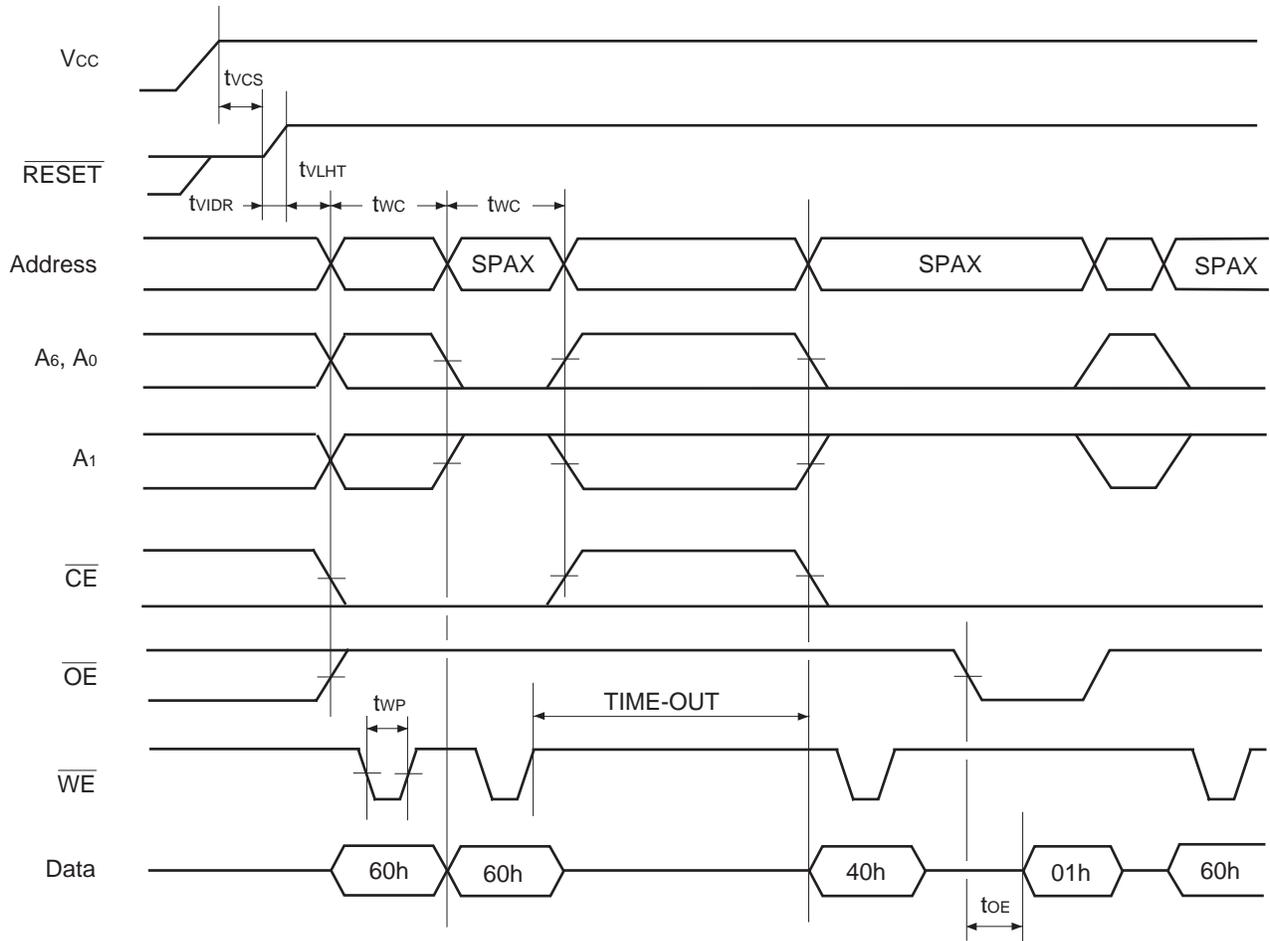
SPAX   : Sector Group Address to be protected
SPAY   : Next Sector Group Address to be protected
TIME-OUT : Time-Out window = 250 μs (Min)

(14) Accelerated Program Timing Diagram
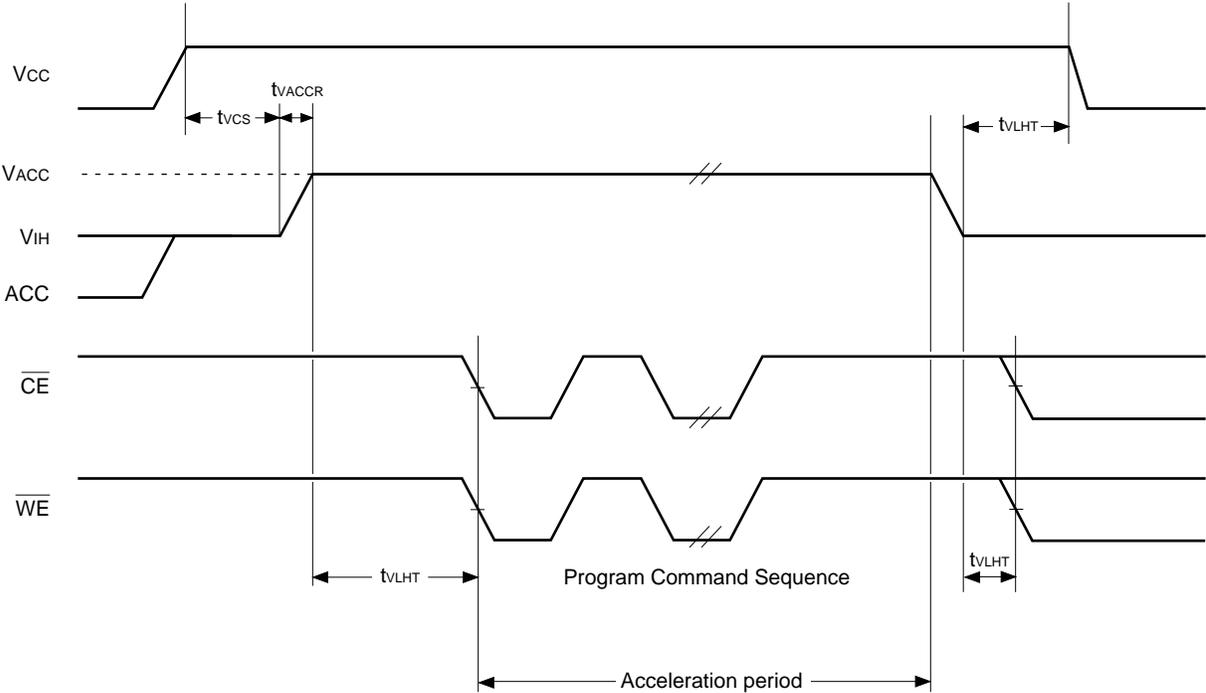
■ FLOW CHART

EMBEDDED ALGORITHMS

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

Program Command Sequence (Address/Command):

```
            ┌──────────────┐
            │   XXXh/AAh   │
            └──────────────┘
                   │
                   ▼
            ┌──────────────┐
            │   XXXh/55h   │
            └──────────────┘
                   │
                   ▼
            ┌──────────────┐
            │   XXXh/A0h   │
            └──────────────┘
                   │
                   ▼
    ┌──────────────────────────────┐
    │ Program Address/Program Data │
    └──────────────────────────────┘
```
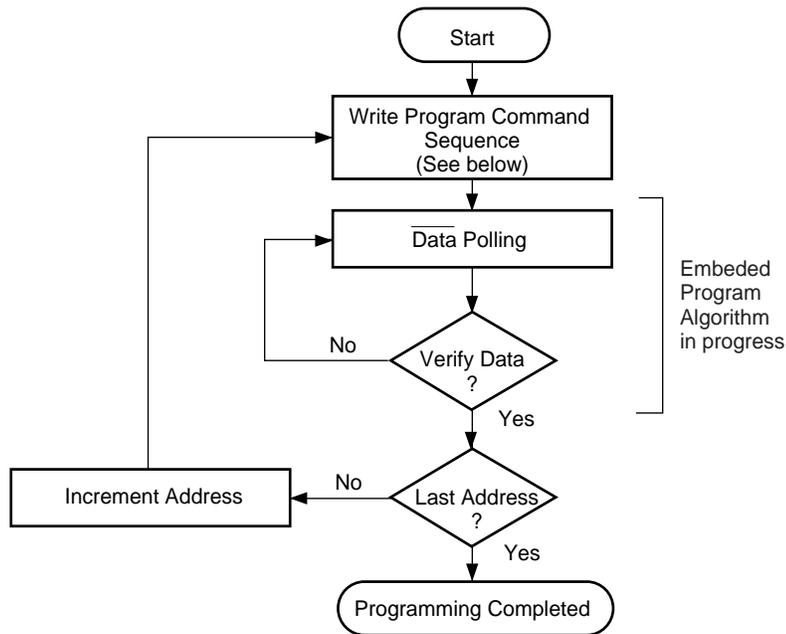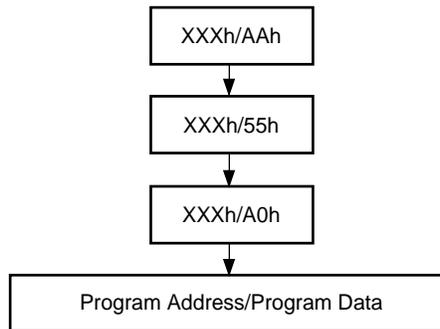
(1) Embedded Program™ Algorithm

EMBEDDED ALGORITHMS

```
                    ┌─────────────┐
                    │    Start    │
                    └─────────────┘
                           │
                           ▼
              ┌──────────────────────────┐
              │  Write Erase Command      │
              │  Sequence                 │
              │  (See below)              │
              └──────────────────────────┘
                           │
                           ▼                    ┐
              ┌──────────────────────────┐      │
         ┌───▶│  ‾‾‾‾                     │      │
         │    │  Data Polling             │      │  Embeded
         │    └──────────────────────────┘      │  Program
         │               │                      │  Algorithm
         │               ▼                      │  in progress
      No │          ◇─────────────◇            │
         └──────────│  Data = FFh   │           │
                    │      ?        │           ┘
                     ◇─────────────◇
                           │ Yes
                           ▼
              ┌──────────────────────────┐
              │   Erasure Completed       │
              └──────────────────────────┘
```

Chip Erase Command Sequence (Address/Command):

```
┌──────────────┐
│  XXXh/AAh    │
└──────────────┘
        │
        ▼
┌──────────────┐
│  XXXh/55h    │
└──────────────┘
        │
        ▼
┌──────────────┐
│  XXXh/80h    │
└──────────────┘
        │
        ▼
┌──────────────┐
│  XXXh/AAh    │
└──────────────┘
        │
        ▼
┌──────────────┐
│  XXXh/55h    │
└──────────────┘
        │
        ▼
┌──────────────┐
│  XXXh/10h    │
└──────────────┘
```
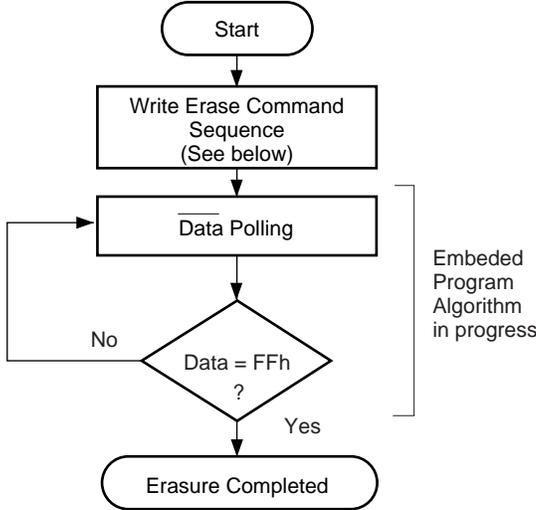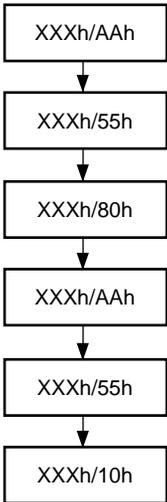
Individual Sector/Multiple Sector Erase Command Sequence (Address/Command):

```
┌──────────────┐
│  XXXh/AAh    │
└──────────────┘
        │
        ▼
┌──────────────┐
│  XXXh/55h    │
└──────────────┘
        │
        ▼
┌──────────────┐
│  XXXh/80h    │
└──────────────┘
        │
        ▼
┌──────────────┐
│  XXXh/AAh    │
└──────────────┘
        │
        ▼
┌──────────────┐
│  XXXh/55h    │
└──────────────┘
        │
        ▼
┌────────────────────┐
│  Sector Address/30h │
└────────────────────┘
        │
        ▼
┌────────────────────┐
│  Sector Address/30h │     Additional sector
└────────────────────┘     erase commands
        ⋮                   are optional.
┌────────────────────┐
│  Sector Address/30h │
└────────────────────┘
```

(2) Embedded Erase™ Algorithm

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
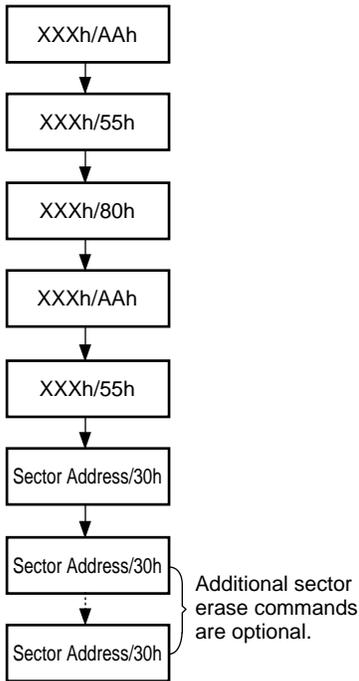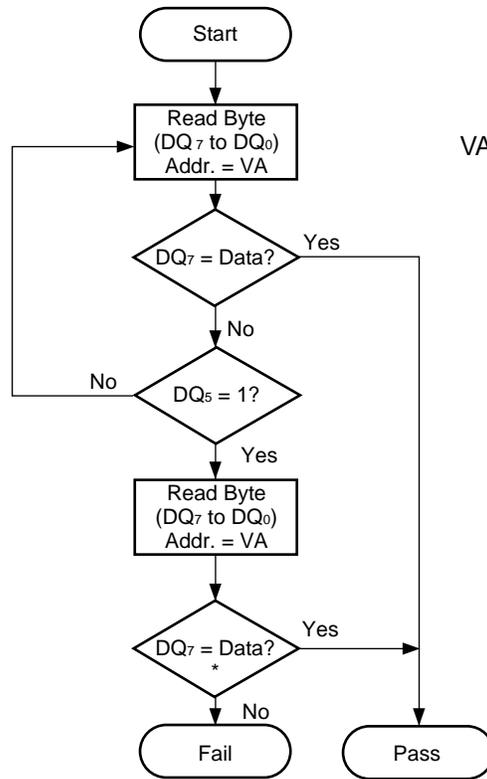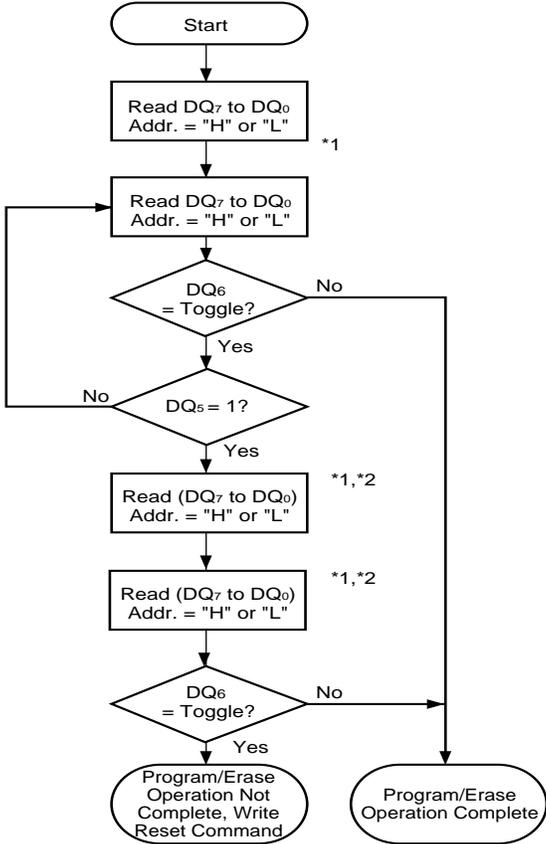
VA = Byte address for programming
= Any of the sector addresses within the sector being erased during sector erase or multiple sector erases operation
= Any of the sector addresses within the sector not being protected during chip erase
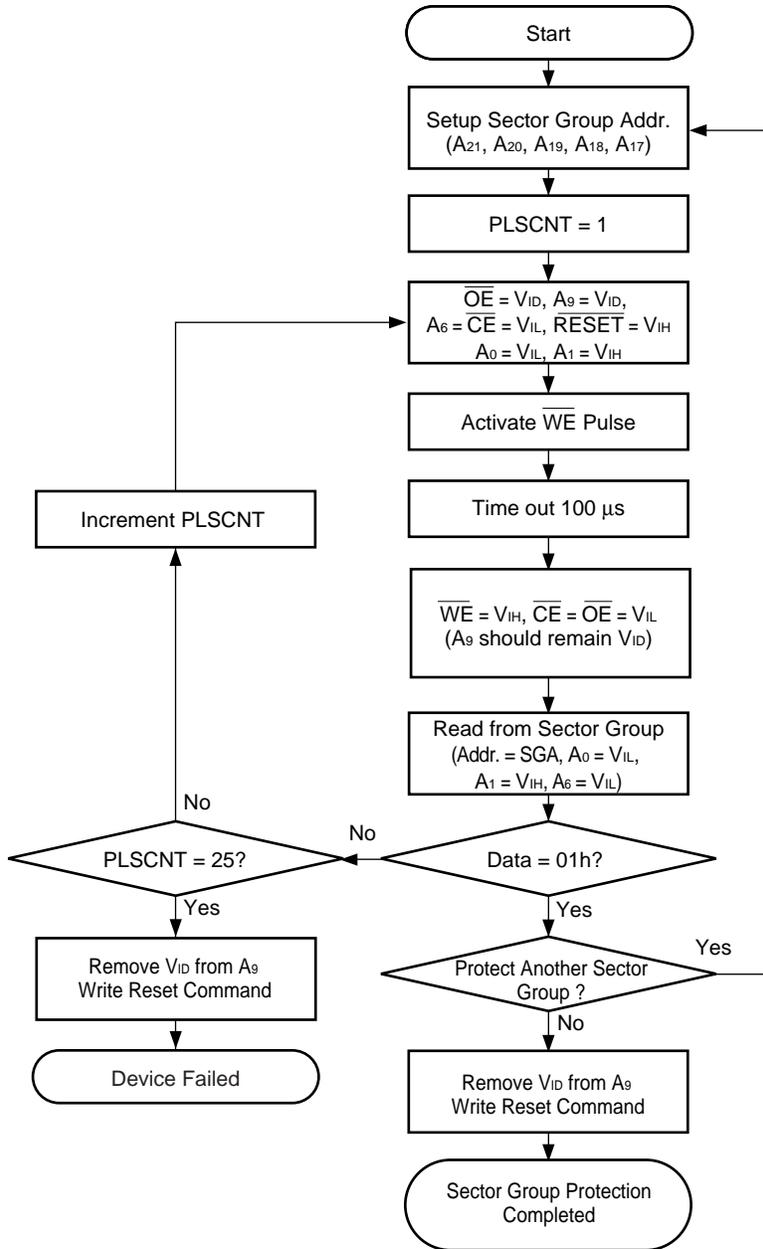
* : DQ$_7$ is rechecked even if DQ$_5$ = "1" because DQ$_7$ may change simultaneously with DQ$_5$.

(3) $\overline{\text{Data}}$ Polling Algorithm

Start

Read DQ7 to DQ0
Addr. = "H" or "L"   *1

Read DQ7 to DQ0
Addr. = "H" or "L"

DQ6
= Toggle?   No

Yes

No   DQ5 = 1?

Yes

Read (DQ7 to DQ0)
Addr. = "H" or "L"   *1,*2

Read (DQ7 to DQ0)
Addr. = "H" or "L"   *1,*2

DQ6
= Toggle?   No

Yes

Program/Erase
Operation Not
Complete, Write
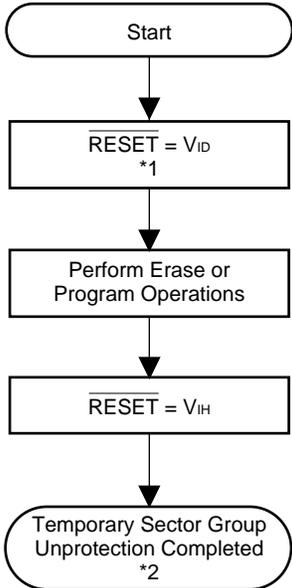Reset Command

Program/Erase
Operation Complete

*1 : Reset toggle bit twice to determine whether it is toggling.
*2 : Recheck toggle bit since it may stop toggle as DQ5 changes to "1".
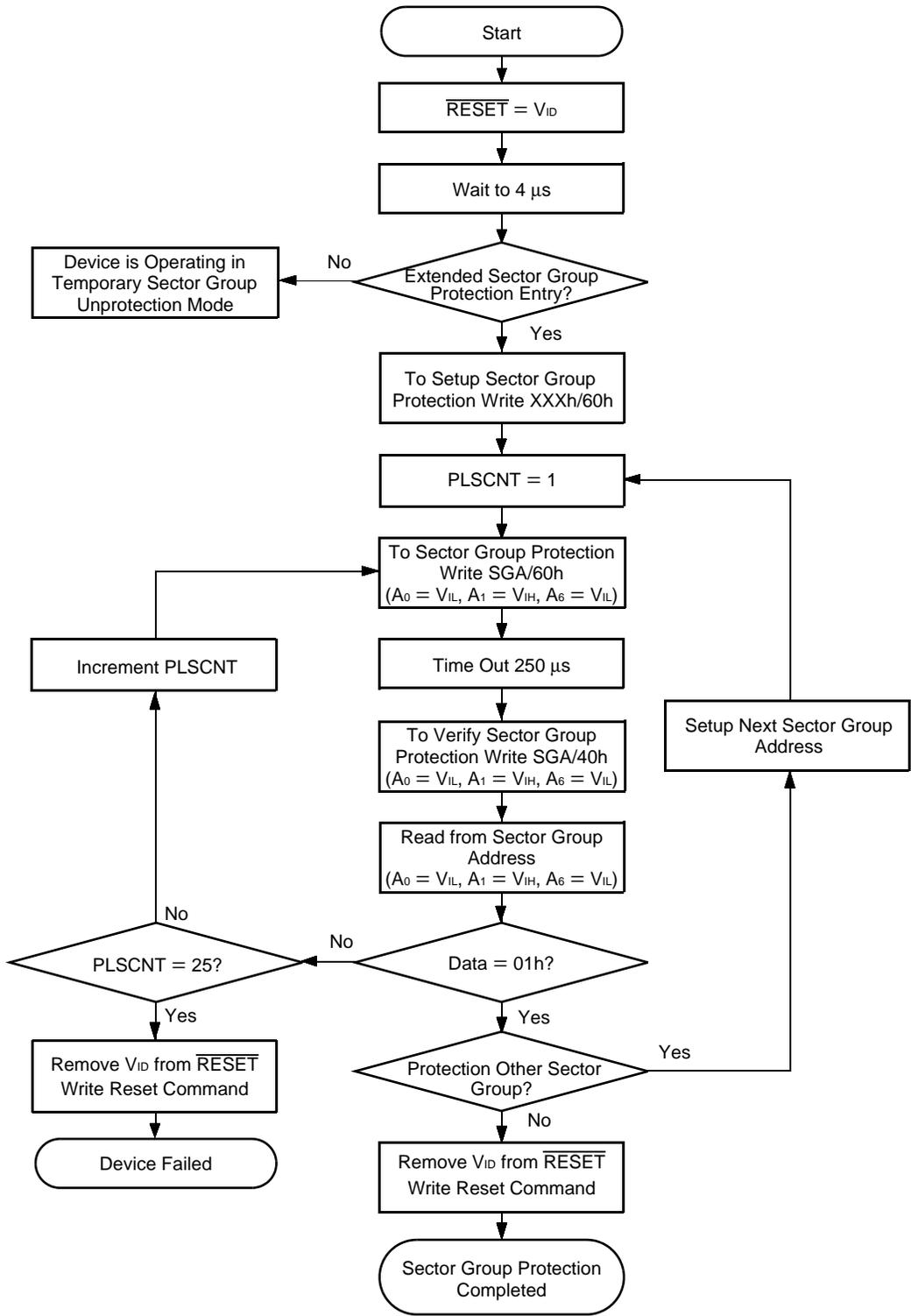
(4) Toggle Bit Algorithm

Start

Setup Sector Group Addr.
(A$_{21}$, A$_{20}$, A$_{19}$, A$_{18}$, A$_{17}$)

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 Group
(Addr. = SGA, A$_0$ = V$_{IL}$,
A$_1$ = V$_{IH}$, A$_6$ = V$_{IL}$)

Increment PLSCNT

No

No

PLSCNT = 25?   Data = 01h?

Yes   Yes

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

Protect Another Sector
Group ?

Yes

No

Device Failed

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

Sector Group Protection
Completed

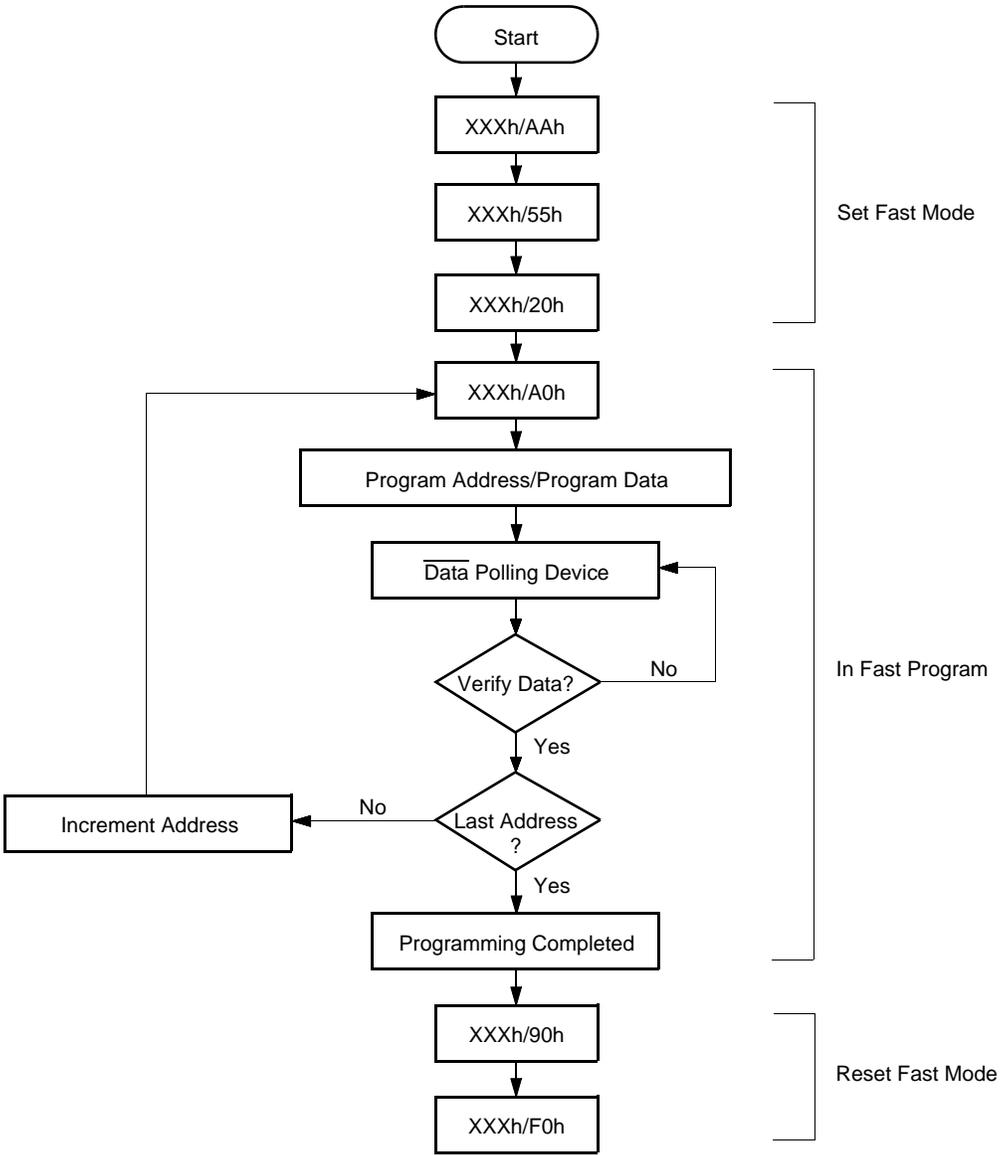(5) Sector Group Protection Algorithm

52

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

(6) Temporary Sector Group Unprotection Algorithm

MBM29LV652UE90

Start

$\overline{RESET} = V_{ID}$

Wait to 4 µs

Extended Sector Group Protection Entry? —No→ Device is Operating in Temporary Sector Group Unprotection Mode

Yes

To Setup Sector Group Protection Write XXXh/60h

PLSCNT = 1

To Sector Group Protection Write SGA/60h
$(A_0 = V_{IL}, A_1 = V_{IH}, A_6 = V_{IL})$

Time Out 250 µs

Increment PLSCNT

To Verify Sector Group Protection Write SGA/40h
$(A_0 = V_{IL}, A_1 = V_{IH}, A_6 = V_{IL})$

Setup Next Sector Group Address

Read from Sector Group Address
$(A_0 = V_{IL}, A_1 = V_{IH}, A_6 = V_{IL})$

No ← PLSCNT = 25? ← No ← Data = 01h?

Yes

Remove V_{ID} from $\overline{RESET}$ Write Reset Command

Device Failed

Protection Other Sector Group? —Yes→

No

Remove V_{ID} from $\overline{RESET}$ Write Reset Command

Sector Group Protection Completed

(7) Extended Sector Group Protection Algorithm

FAST MODE ALGORITHM

```
                        ┌─────────┐
                        │  Start  │
                        └────┬────┘
                             │
                      ┌──────▼──────┐
                      │  XXXh/AAh   │  ─────┐
                      └──────┬──────┘       │
                             │              │
                      ┌──────▼──────┐       │
                      │  XXXh/55h   │       ├── Set Fast Mode
                      └──────┬──────┘       │
                             │              │
                      ┌──────▼──────┐       │
                      │  XXXh/20h   │  ─────┘
                      └──────┬──────┘
                             │
                      ┌──────▼──────┐  ◄────────┐
                 ┌───►│  XXXh/A0h   │           │
                 │    └──────┬──────┘           │
                 │           │                  │
                 │  ┌────────▼─────────────┐    │
                 │  │Program Address/      │    │
                 │  │Program Data          │    │
                 │  └────────┬─────────────┘    │
                 │           │                  │
                 │  ┌────────▼─────────────┐    │
                 │  │ Data Polling Device  │◄─┐ │
                 │  └────────┬─────────────┘  │ │
                 │           │              No│ ├── In Fast Program
                 │        ╱──▼──╲             │ │
                 │       ╱ Verify ╲───────────┘ │
                 │       ╲ Data?  ╱             │
                 │        ╲──┬──╱               │
                 │        Yes│                  │
                 │        ╱──▼──╲               │
  ┌──────────────▼───┐ No ╱ Last  ╲            │
  │Increment Address │◄───╲Address?╱           │
  └──────────────────┘    ╲──┬──╱              │
                          Yes│                 │
                   ┌─────────▼──────────┐      │
                   │Programming Completed│  ────┘
                   └─────────┬──────────┘
                             │
                      ┌──────▼──────┐  ─────┐
                      │  XXXh/90h   │       │
                      └──────┬──────┘       ├── Reset Fast Mode
                             │              │
                      ┌──────▼──────┐       │
                      │  XXXh/F0h   │  ─────┘
                      └─────────────┘
```

(8) Embedded Program™ Algorithm for Fast Mode

■ ORDERING INFORMATION

Part No.	Package	Access Time (ns)	Remarks
MBM29LV652UE90PBT	63-pin plastic FBGA (BGA-63P-M02)	90	

MBM29LV652U   E   90   PBT

PACKAGE TYPE
PBT = 63-ball Fine pitch Ball Gind Array Package
  (FBGA) Standard Pinout

SPEED OPTION
See Product Selector Guide

DEVICE REVISION

DEVICE NUMBER/DESCRIPTION
MBM29LV652U
64 Mega-bit (4M × 16-Bit) CMOS Flash Memory
3.0 V-only Read, Program, and Erase

■ PACKAGE DIMENSION

63-ball plastic FBGA
(BGA-63P-M02)

11.00±0.10(.433±.004)

1.05 $^{+0.15}_{-0.10}$
$\left(.041 ^{+.006}_{-.004}\right)$
(Mounting height)

0.38±0.10
(.015±.004)
(Stand off)

10.00±0.10
(.394±.004)

(8.80(.346))

(7.20(.283))

(5.60(.220))

0.80(.031)TYP

(4.00(.157))

(5.60(.220))

8
7
6
5
4
3
2
1

M L K J H G F E D C B A

INDEX AREA

INDEX BALL

63-ø0.45±0.05
(63-ø0.18±.002)

⊕ 0.08(.003) Ⓜ

⌒ 0.10(.004)
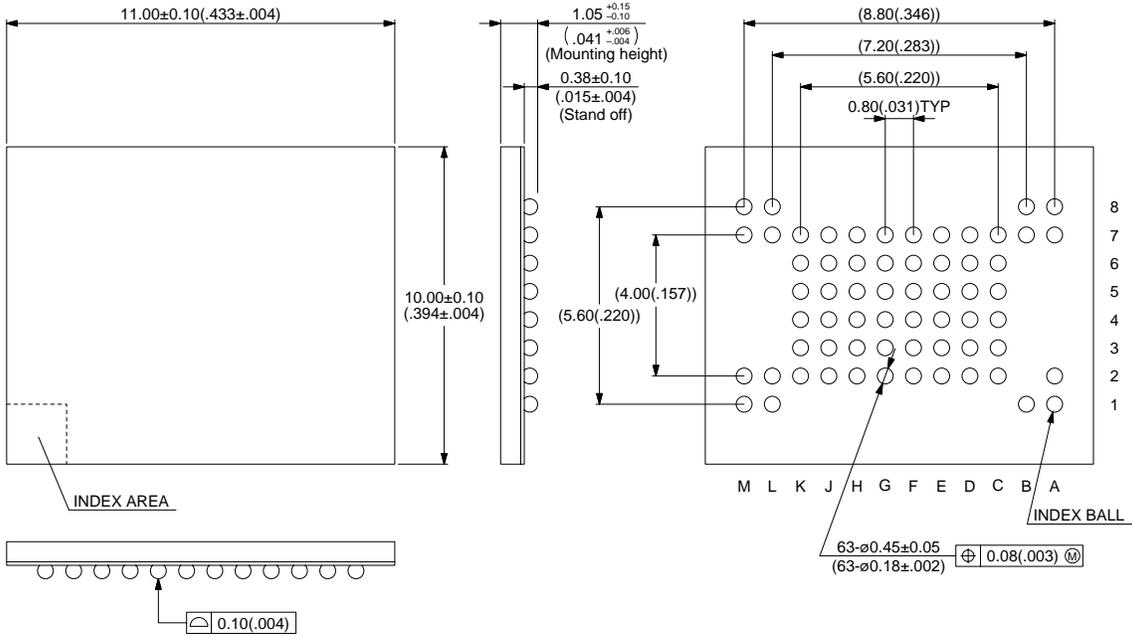
Ⓒ 2001 FUJITSU LIMITED B63002S-c-3-2

Dimensions in mm (inches)
Note : The values in parentheses are reference values.

FUJITSU LIMITED