



# FLASH MEMORY

## MT28F642D18 MT28F642D20

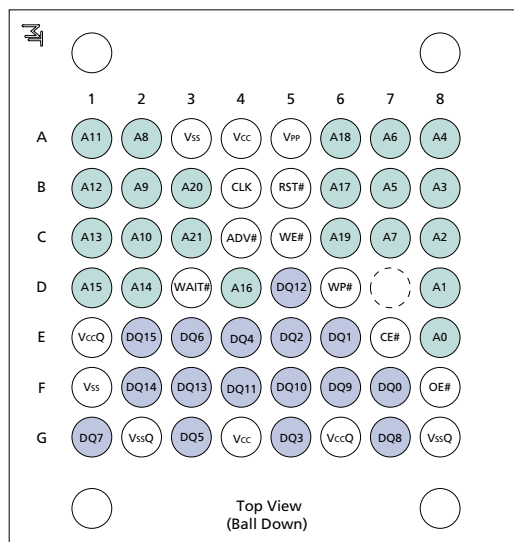
Low Voltage, Extended Temperature  
0.18µm Process Technology

### FEATURES

- Single device supports asynchronous, page, and burst operations
- Flexible dual-bank architecture
  - Support for true concurrent operation with zero latency
  - Read bank *a* during program bank *b* and vice versa
  - Read bank *a* during erase bank *b* and vice versa
- Basic configuration:
  - One hundred and thirty-five erasable blocks
  - Bank *a* (16Mb for data storage)
  - Bank *b* (48Mb for program storage)
- V<sub>CC</sub>, V<sub>CCQ</sub>, V<sub>PP</sub> voltages
  - 1.70V (MIN), 1.90V (MAX) V<sub>CC</sub>, V<sub>CCQ</sub> (MT28F642D18 only)
  - 1.80V (MIN), 2.20V (MAX) V<sub>CC</sub>, and 2.25V (MAX) V<sub>CCQ</sub> (MT28F642D20 only)
  - 1.80V (TYP) V<sub>PP</sub> (in-system PROGRAM/ERASE)
  - 12V ±5% (HV) V<sub>PP</sub> tolerant (factory programming compatibility)
- Random access time: 70ns @ 1.80V V<sub>CC</sub><sup>1</sup>
- Burst Mode read access
  - MAX clock rate: 54 MHz (<sup>t</sup>CLK = 18.5ns)
  - Burst latency: 70ns @ 1.80V V<sub>CC</sub> and 54 MHz
  - <sup>t</sup>CLK: 15ns @ 1.80V V<sub>CC</sub> and 54 MHz
- Page Mode read access<sup>1</sup>
  - Four-/eight-word page
  - Interpage read access: 70ns @ 1.80V
  - Intrapage read access: 30ns @ 1.80V
- Low power consumption (V<sub>CC</sub> = 2.20V)
  - Asynchronous Read < 15mA
  - Interpage Read < 15mA
  - Intrapage Read < 5mA
  - Continuous Burst Read < 10mA
  - WRITE < 55mA (MAX)
  - ERASE < 45mA (MAX)
  - Standby < 50µA (MAX)
  - Automatic power save (APS) feature
  - Deep power-down < 25µA (MAX)
- Enhanced write and erase suspend options
- Accelerated programming algorithm (APA) in-system and in-factory
- Dual 64-bit chip protection registers for security purposes

NOTE: 1. Data based on MT28F642D20 device.

### PIN ASSIGNMENT 59-Ball FBGA



**NOTE:** See page 7 for Ball Description Table.  
See page 50 for mechanical drawing.

- Cross-compatible command support
  - Extended command set
  - Common flash interface
- PROGRAM/ERASE cycle
  - 100,000 WRITE/ERASE cycles per block

### OPTIONS

### MARKING

- |                                |     |
|--------------------------------|-----|
| Timing                         |     |
| 80ns access                    | -80 |
| 70ns access                    | -70 |
| Frequency                      |     |
| 40 MHz                         | 4   |
| 54 MHz                         | 5   |
| Boot Block Configuration       |     |
| Top                            | T   |
| Bottom                         | B   |
| Package                        |     |
| 59-ball FBGA (8 x 7 ball grid) | FN  |
| Operating Temperature Range    |     |
| Extended (-40°C to +85°C)      | ET  |

Part Number Example:

**MT28F642D20FN-804 TET**



## GENERAL DESCRIPTION

The MT28F642D20 and MT28F642D18 are high-performance, high-density, nonvolatile memory solutions that can significantly improve system performance. This new architecture features a two-memory-bank configuration that supports dual-bank operation with no latency.

A high-performance bus interface allows a fast burst or page mode data transfer; a conventional asynchronous bus interface is provided as well.

The devices allow soft protection for blocks, as read-only, by configuring soft protection registers with dedicated command sequences. For security purposes, two 64-bit chip protection registers are provided.

The embedded WORD WRITE and BLOCK ERASE functions are fully automated by an on-chip write state machine (WSM). Two on-chip status registers, one for each of the two memory partitions, can be used to monitor the WSM status and to determine the progress of the program/erase task.

The erase/program suspend functionality allows compatibility with existing EEPROM emulation software packages.

These devices are manufactured using 0.18µm process technology.

Please refer to the Micron Web site ([www.micron.com/flash](http://www.micron.com/flash)) for the latest data sheet.

## ARCHITECTURE AND MEMORY ORGANIZATION

The Flash devices contain two separate banks of memory (bank *a* and bank *b*) for simultaneous READ and WRITE operations, which are available in the following bank segmentation configurations:

- Bank *a* comprises one-fourth of the memory and contains 8 x 4K-word parameter blocks and 31 x 32K-word blocks.
- Bank *b* represents three-fourths of the memory, is equally sectored, and contains 96 x 32K-word blocks.

Figures 2 and 3 show the bottom and top memory organizations.

## DEVICE MARKING

Due to the size of the package, Micron's standard part number is not printed on the top of each device. Instead, an abbreviated device mark comprised of a five-digit alphanumeric code is used. The abbreviated device marks are cross referenced to the Micron part numbers in Table 1.

**Table 1**  
**Cross Reference for Abbreviated Device Marks**

| PART NUMBER           | PRODUCT MARKING | SAMPLE MARKING | MECHANICAL SAMPLE MARKING |
|-----------------------|-----------------|----------------|---------------------------|
| MT28F642D20FN-705 TET | FW906           | FX906          | FY906                     |
| MT28F642D20FN-705 BET | FW905           | FX905          | FY905                     |
| MT28F642D20FN-804 TET | FW907           | FX907          | FY907                     |
| MT28F642D20FN-804 BET | FW908           | FX908          | FY908                     |
| MT28F642D18FN-705 TET | FW909           | FX909          | FY909                     |
| MT28F642D18FN-705 BET | FW910           | FX910          | FY910                     |
| MT28F642D18FN-804 TET | FW911           | FX911          | FY911                     |
| MT28F642D18FN-804 BET | FW912           | FX912          | FY912                     |

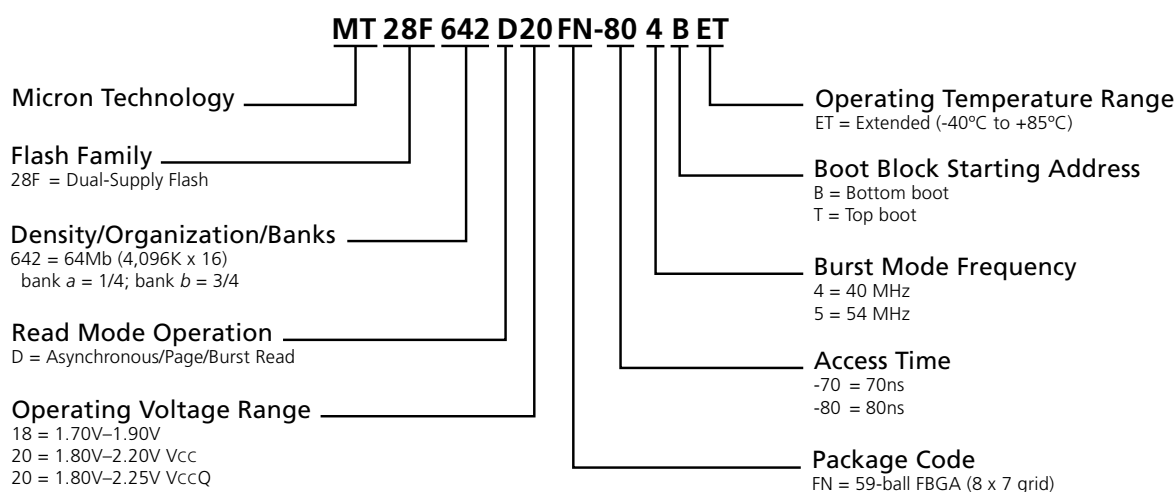


## PART NUMBERING INFORMATION

Micron's low-power devices are available with several different combinations of features (see Figure 1).

Valid combinations of features and their corresponding part numbers are listed in Table 2.

**Figure 1  
Part Number Chart**



**Table 2  
Valid Part Number Combinations<sup>1</sup>**

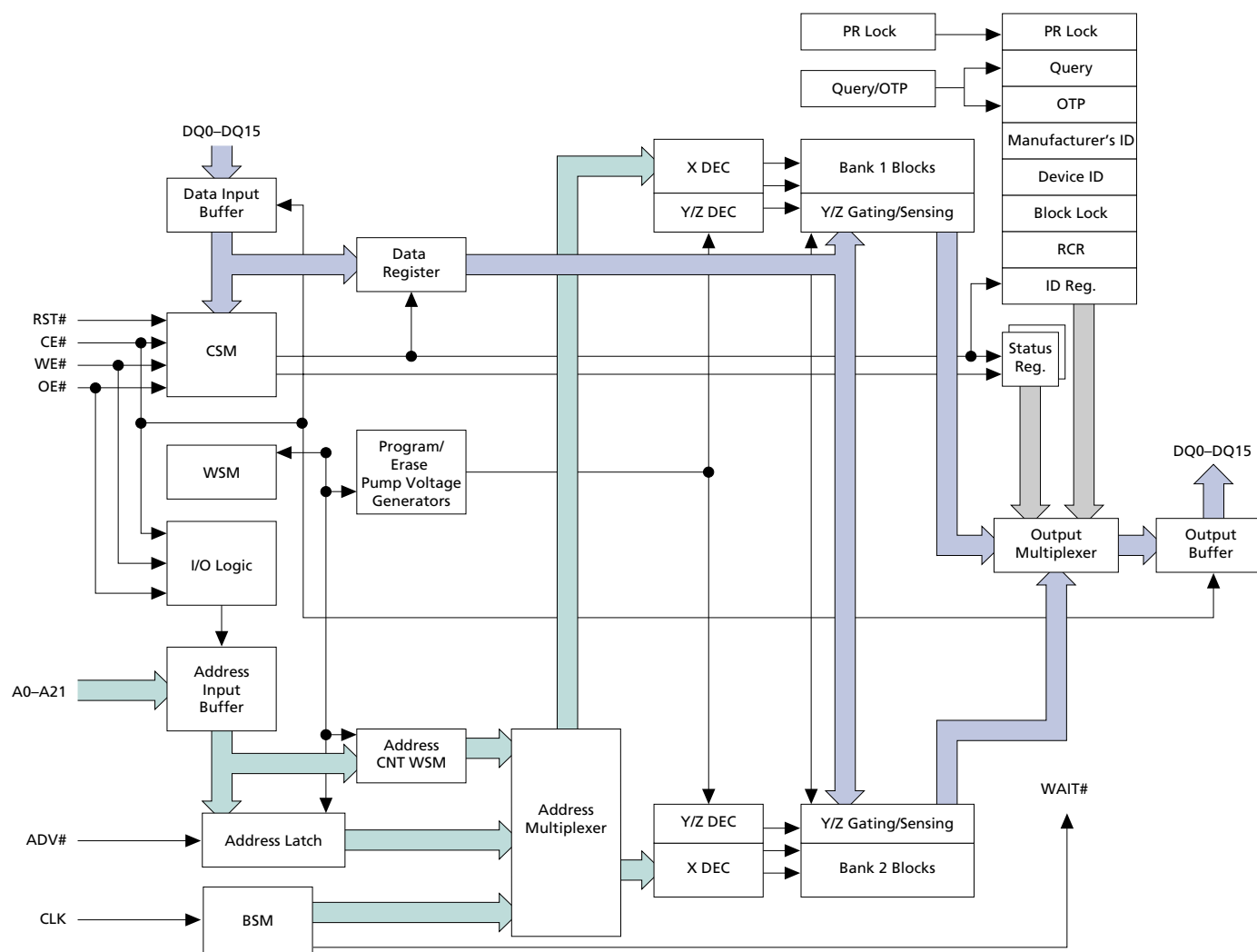
| PART NUMBER           | ACCESS TIME (ns) | BOOT BLOCK STARTING ADDRESS | BURST FREQUENCY (MHz) | OPERATING TEMPERATURE RANGE |
|-----------------------|------------------|-----------------------------|-----------------------|-----------------------------|
| MT28F642D20FN-705 TET | 70               | Top                         | 54                    | -40°C to +85°C              |
| MT28F642D20FN-705 BET | 70               | Bottom                      | 54                    | -40°C to +85°C              |
| MT28F642D20FN-804 TET | 80               | Top                         | 40                    | -40°C to +85°C              |
| MT28F642D20FN-804 BET | 80               | Bottom                      | 40                    | -40°C to +85°C              |
| MT28F642D18FN-705 TET | 70               | Top                         | 54                    | -40°C to +85°C              |
| MT28F642D18FN-705 BET | 70               | Bottom                      | 54                    | -40°C to +85°C              |
| MT28F642D18FN-804 TET | 80               | Top                         | 40                    | -40°C to +85°C              |
| MT28F642D18FN-804 BET | 80               | Bottom                      | 40                    | -40°C to +85°C              |

**NOTE:** 1. For part number combinations not listed in this table, please contact your Micron representative.



# 4 MEG x 16 ASYNC/PAGE/BURST FLASH MEMORY

## FUNCTIONAL BLOCK DIAGRAM





# 4 MEG x 16 ASYNCR/PAGE/BURST FLASH MEMORY

**Figure 2**  
**Bottom Boot Block Device**

| Bank b = 48Mb |                                     |                        |
|---------------|-------------------------------------|------------------------|
| Block         | Block Size<br>(K-bytes/<br>K-words) | Address Range<br>(x16) |
| 134           | 64/32                               | 3F8000h–3FFFFFh        |
| 133           | 64/32                               | 3F0000h–3F7FFFh        |
| 132           | 64/32                               | 3E8000h–3EFFFFh        |
| 131           | 64/32                               | 3E0000h–3E7FFFh        |
| 130           | 64/32                               | 3D8000h–3DFFFFh        |
| 129           | 64/32                               | 3D0000h–3D7FFFh        |
| 128           | 64/32                               | 3C8000h–3CFFFFh        |
| 127           | 64/32                               | 3C0000h–3C7FFFh        |
| 126           | 64/32                               | 3B8000h–3BFFFFh        |
| 125           | 64/32                               | 3B0000h–3B7FFFh        |
| 124           | 64/32                               | 3A8000h–3AFFFFh        |
| 123           | 64/32                               | 3A0000h–3A7FFFh        |
| 122           | 64/32                               | 398000h–39FFFFh        |
| 121           | 64/32                               | 390000h–397FFFh        |
| 120           | 64/32                               | 388000h–38FFFFh        |
| 119           | 64/32                               | 380000h–387FFFh        |
| 118           | 64/32                               | 378000h–37FFFFh        |
| 117           | 64/32                               | 370000h–377FFFh        |
| 116           | 64/32                               | 368000h–36FFFFh        |
| 115           | 64/32                               | 360000h–367FFFh        |
| 114           | 64/32                               | 358000h–35FFFFh        |
| 113           | 64/32                               | 350000h–357FFFh        |
| 112           | 64/32                               | 348000h–34FFFFh        |
| 111           | 64/32                               | 340000h–347FFFh        |
| 110           | 64/32                               | 338000h–33FFFFh        |
| 109           | 64/32                               | 330000h–337FFFh        |
| 108           | 64/32                               | 328000h–32FFFFh        |
| 107           | 64/32                               | 320000h–327FFFh        |
| 106           | 64/32                               | 318000h–31FFFFh        |
| 105           | 64/32                               | 310000h–317FFFh        |
| 104           | 64/32                               | 308000h–30FFFFh        |
| 103           | 64/32                               | 300000h–307FFFh        |
| 102           | 64/32                               | 2F8000h–2FFFFFh        |
| 101           | 64/32                               | 2F0000h–2F7FFFh        |
| 100           | 64/32                               | 2E8000h–2EFFFFh        |
| 99            | 64/32                               | 2E0000h–2E7FFFh        |
| 98            | 64/32                               | 2D8000h–2DFFFFh        |
| 97            | 64/32                               | 2D0000h–2D7FFFh        |
| 96            | 64/32                               | 2C8000h–2CFFFFh        |
| 95            | 64/32                               | 2C0000h–2C7FFFh        |
| 94            | 64/32                               | 2B8000h–2BFFFFh        |
| 93            | 64/32                               | 2B0000h–2B7FFFh        |
| 92            | 64/32                               | 2A8000h–2AFFFFh        |
| 91            | 64/32                               | 2A0000h–2A7FFFh        |
| 90            | 64/32                               | 298000h–29FFFFh        |
| 89            | 64/32                               | 290000h–297FFFh        |
| 88            | 64/32                               | 288000h–28FFFFh        |
| 87            | 64/32                               | 280000h–287FFFh        |

| Bank b = 48Mb |                                     |                        |
|---------------|-------------------------------------|------------------------|
| Block         | Block Size<br>(K-bytes/<br>K-words) | Address Range<br>(x16) |
| 86            | 64/32                               | 278000h–27FFFFh        |
| 85            | 64/32                               | 270000h–277FFFh        |
| 84            | 64/32                               | 268000h–26FFFFh        |
| 83            | 64/32                               | 260000h–267FFFh        |
| 82            | 64/32                               | 258000h–25FFFFh        |
| 81            | 64/32                               | 250000h–257FFFh        |
| 80            | 64/32                               | 248000h–24FFFFh        |
| 79            | 64/32                               | 240000h–247FFFh        |
| 78            | 64/32                               | 238000h–23FFFFh        |
| 77            | 64/32                               | 230000h–237FFFh        |
| 76            | 64/32                               | 228000h–22FFFFh        |
| 75            | 64/32                               | 220000h–227FFFh        |
| 74            | 64/32                               | 218000h–21FFFFh        |
| 73            | 64/32                               | 210000h–217FFFh        |
| 72            | 64/32                               | 208000h–20FFFFh        |
| 71            | 64/32                               | 200000h–207FFFh        |
| 70            | 64/32                               | 1F8000h–1FFFFFh        |
| 69            | 64/32                               | 1F0000h–1F7FFFh        |
| 68            | 64/32                               | 1E8000h–1EFFFFh        |
| 67            | 64/32                               | 1E0000h–1E7FFFh        |
| 66            | 64/32                               | 1D8000h–1DFFFFh        |
| 65            | 64/32                               | 1D0000h–1D7FFFh        |
| 64            | 64/32                               | 1C8000h–1CFFFFh        |
| 63            | 64/32                               | 1C0000h–1C7FFFh        |
| 62            | 64/32                               | 1B8000h–1BFFFFh        |
| 61            | 64/32                               | 1B0000h–1B7FFFh        |
| 60            | 64/32                               | 1A8000h–1AFFFFh        |
| 59            | 64/32                               | 1A0000h–1A7FFFh        |
| 58            | 64/32                               | 198000h–19FFFFh        |
| 57            | 64/32                               | 190000h–197FFFh        |
| 56            | 64/32                               | 188000h–18FFFFh        |
| 55            | 64/32                               | 180000h–187FFFh        |
| 54            | 64/32                               | 178000h–17FFFFh        |
| 53            | 64/32                               | 170000h–177FFFh        |
| 52            | 64/32                               | 168000h–16FFFFh        |
| 51            | 64/32                               | 160000h–167FFFh        |
| 50            | 64/32                               | 158000h–15FFFFh        |
| 49            | 64/32                               | 150000h–157FFFh        |
| 48            | 64/32                               | 148000h–14FFFFh        |
| 47            | 64/32                               | 140000h–147FFFh        |
| 46            | 64/32                               | 138000h–13FFFFh        |
| 45            | 64/32                               | 130000h–137FFFh        |
| 44            | 64/32                               | 128000h–12FFFFh        |
| 43            | 64/32                               | 120000h–127FFFh        |
| 42            | 64/32                               | 118000h–11FFFFh        |
| 41            | 64/32                               | 110000h–117FFFh        |
| 40            | 64/32                               | 108000h–10FFFFh        |
| 39            | 64/32                               | 100000h–107FFFh        |

| Bank a = 16Mb |                                     |                        |
|---------------|-------------------------------------|------------------------|
| Block         | Block Size<br>(K-bytes/<br>K-words) | Address Range<br>(x16) |
| 38            | 64/32                               | 0F8000h–0FFFFFh        |
| 37            | 64/32                               | 0F0000h–0F7FFFh        |
| 36            | 64/32                               | 0E8000h–0EFFFFh        |
| 35            | 64/32                               | 0E0000h–0E7FFFh        |
| 34            | 64/32                               | 0D8000h–0DFFFFh        |
| 33            | 64/32                               | 0D0000h–0D7FFFh        |
| 32            | 64/32                               | 0C8000h–0CFFFFh        |
| 31            | 64/32                               | 0C0000h–0C7FFFh        |
| 30            | 64/32                               | 0B8000h–0BFFFFh        |
| 29            | 64/32                               | 0B0000h–0B7FFFh        |
| 28            | 64/32                               | 0A8000h–0AFFFFh        |
| 27            | 64/32                               | 0A0000h–0A7FFFh        |
| 26            | 64/32                               | 098000h–09FFFFh        |
| 25            | 64/32                               | 090000h–097FFFh        |
| 24            | 64/32                               | 088000h–08FFFFh        |
| 23            | 64/32                               | 080000h–087FFFh        |
| 22            | 64/32                               | 078000h–07FFFFh        |
| 21            | 64/32                               | 070000h–077FFFh        |
| 20            | 64/32                               | 068000h–06FFFFh        |
| 19            | 64/32                               | 060000h–067FFFh        |
| 18            | 64/32                               | 058000h–05FFFFh        |
| 17            | 64/32                               | 050000h–057FFFh        |
| 16            | 64/32                               | 048000h–04FFFFh        |
| 15            | 64/32                               | 040000h–047FFFh        |
| 14            | 64/32                               | 038000h–03FFFFh        |
| 13            | 64/32                               | 030000h–037FFFh        |
| 12            | 64/32                               | 028000h–02FFFFh        |
| 11            | 64/32                               | 020000h–027FFFh        |
| 10            | 64/32                               | 018000h–01FFFFh        |
| 9             | 64/32                               | 010000h–017FFFh        |
| 8             | 64/32                               | 008000h–00FFFFh        |
| 7             | 8/4                                 | 007000h–007FFFh        |
| 6             | 8/4                                 | 006000h–006FFFh        |
| 5             | 8/4                                 | 005000h–005FFFh        |
| 4             | 8/4                                 | 004000h–004FFFh        |
| 3             | 8/4                                 | 003000h–003FFFh        |
| 2             | 8/4                                 | 002000h–002FFFh        |
| 1             | 8/4                                 | 001000h–001FFFh        |
| 0             | 8/4                                 | 000000h–00FFFh         |



# 4 MEG x 16 ASYNCR/PAGE/BURST FLASH MEMORY

**Figure 3**  
**Top Boot Block Device**

| Bank a = 16Mb |                                     |                        |
|---------------|-------------------------------------|------------------------|
| Block         | Block Size<br>(K-bytes/<br>K-words) | Address Range<br>(x16) |
| 134           | 8/4                                 | 3FF000h–3FFFFFh        |
| 133           | 8/4                                 | 3FE000h–3FEFFFh        |
| 132           | 8/4                                 | 3FD000h–3FDFFFh        |
| 131           | 8/4                                 | 3FC000h–3FCFFFh        |
| 130           | 8/4                                 | 3FB000h–3FBFFFh        |
| 129           | 8/4                                 | 3FA000h–3FAFFFh        |
| 128           | 8/4                                 | 3F9000h–3F9FFFh        |
| 127           | 8/4                                 | 3F8000h–3F8FFFh        |
| 126           | 64/32                               | 3F0000h–3F7FFFh        |
| 125           | 64/32                               | 3E8000h–3E7FFFh        |
| 124           | 64/32                               | 3E0000h–3E7FFFh        |
| 123           | 64/32                               | 3D8000h–3D7FFFh        |
| 122           | 64/32                               | 3D0000h–3D7FFFh        |
| 121           | 64/32                               | 3C8000h–3C7FFFh        |
| 120           | 64/32                               | 3C0000h–3C7FFFh        |
| 119           | 64/32                               | 3B8000h–3B7FFFh        |
| 118           | 64/32                               | 3B0000h–3B7FFFh        |
| 117           | 64/32                               | 3A8000h–3A7FFFh        |
| 116           | 64/32                               | 3A0000h–3A7FFFh        |
| 115           | 64/32                               | 398000h–397FFFh        |
| 114           | 64/32                               | 390000h–397FFFh        |
| 113           | 64/32                               | 388000h–387FFFh        |
| 112           | 64/32                               | 380000h–387FFFh        |
| 111           | 64/32                               | 378000h–377FFFh        |
| 110           | 64/32                               | 370000h–377FFFh        |
| 109           | 64/32                               | 368000h–367FFFh        |
| 108           | 64/32                               | 360000h–367FFFh        |
| 107           | 64/32                               | 358000h–357FFFh        |
| 106           | 64/32                               | 350000h–357FFFh        |
| 105           | 64/32                               | 348000h–347FFFh        |
| 104           | 64/32                               | 340000h–347FFFh        |
| 103           | 64/32                               | 338000h–337FFFh        |
| 102           | 64/32                               | 330000h–337FFFh        |
| 101           | 64/32                               | 328000h–327FFFh        |
| 100           | 64/32                               | 320000h–327FFFh        |
| 99            | 64/32                               | 318000h–317FFFh        |
| 98            | 64/32                               | 310000h–317FFFh        |
| 97            | 64/32                               | 308000h–307FFFh        |
| 96            | 64/32                               | 300000h–307FFFh        |

| Bank b = 48Mb |                                     |                        |
|---------------|-------------------------------------|------------------------|
| Block         | Block Size<br>(K-bytes/<br>K-words) | Address Range<br>(x16) |
| 95            | 64/32                               | 2F8000h–2FFFFFh        |
| 94            | 64/32                               | 2F0000h–2F7FFFh        |
| 93            | 64/32                               | 2E8000h–2E7FFFh        |
| 92            | 64/32                               | 2E0000h–2E7FFFh        |
| 91            | 64/32                               | 2D8000h–2D7FFFh        |
| 90            | 64/32                               | 2D0000h–2D7FFFh        |
| 89            | 64/32                               | 2C8000h–2C7FFFh        |
| 88            | 64/32                               | 2C0000h–2C7FFFh        |
| 87            | 64/32                               | 2B8000h–2B7FFFh        |
| 86            | 64/32                               | 2B0000h–2B7FFFh        |
| 85            | 64/32                               | 2A8000h–2A7FFFh        |
| 84            | 64/32                               | 2A0000h–2A7FFFh        |
| 83            | 64/32                               | 298000h–297FFFh        |
| 82            | 64/32                               | 290000h–297FFFh        |
| 81            | 64/32                               | 288000h–287FFFh        |
| 80            | 64/32                               | 280000h–287FFFh        |
| 79            | 64/32                               | 278000h–277FFFh        |
| 78            | 64/32                               | 270000h–277FFFh        |
| 77            | 64/32                               | 268000h–267FFFh        |
| 76            | 64/32                               | 260000h–267FFFh        |
| 75            | 64/32                               | 258000h–257FFFh        |
| 74            | 64/32                               | 250000h–257FFFh        |
| 73            | 64/32                               | 248000h–247FFFh        |
| 72            | 64/32                               | 240000h–247FFFh        |
| 71            | 64/32                               | 238000h–237FFFh        |
| 70            | 64/32                               | 230000h–237FFFh        |
| 69            | 64/32                               | 228000h–227FFFh        |
| 68            | 64/32                               | 220000h–227FFFh        |
| 67            | 64/32                               | 218000h–217FFFh        |
| 66            | 64/32                               | 210000h–217FFFh        |
| 65            | 64/32                               | 208000h–207FFFh        |
| 64            | 64/32                               | 200000h–207FFFh        |
| 63            | 64/32                               | 1F8000h–1FFFFFh        |
| 62            | 64/32                               | 1F0000h–1F7FFFh        |
| 61            | 64/32                               | 1E8000h–1E7FFFh        |
| 60            | 64/32                               | 1E0000h–1E7FFFh        |
| 59            | 64/32                               | 1D8000h–1D7FFFh        |
| 58            | 64/32                               | 1D0000h–1D7FFFh        |
| 57            | 64/32                               | 1C8000h–1C7FFFh        |
| 56            | 64/32                               | 1C0000h–1C7FFFh        |
| 55            | 64/32                               | 1B8000h–1B7FFFh        |
| 54            | 64/32                               | 1B0000h–1B7FFFh        |
| 53            | 64/32                               | 1A8000h–1A7FFFh        |
| 52            | 64/32                               | 1A0000h–1A7FFFh        |
| 51            | 64/32                               | 198000h–197FFFh        |
| 50            | 64/32                               | 190000h–197FFFh        |
| 49            | 64/32                               | 188000h–187FFFh        |
| 48            | 64/32                               | 180000h–187FFFh        |

| Bank b = 48Mb |                                     |                        |
|---------------|-------------------------------------|------------------------|
| Block         | Block Size<br>(K-bytes/<br>K-words) | Address Range<br>(x16) |
| 47            | 64/32                               | 178000h–17FFFFh        |
| 46            | 64/32                               | 170000h–177FFFh        |
| 45            | 64/32                               | 168000h–167FFFh        |
| 44            | 64/32                               | 160000h–167FFFh        |
| 43            | 64/32                               | 158000h–157FFFh        |
| 42            | 64/32                               | 150000h–157FFFh        |
| 41            | 64/32                               | 148000h–147FFFh        |
| 40            | 64/32                               | 140000h–147FFFh        |
| 39            | 64/32                               | 138000h–137FFFh        |
| 38            | 64/32                               | 130000h–137FFFh        |
| 37            | 64/32                               | 128000h–127FFFh        |
| 36            | 64/32                               | 120000h–127FFFh        |
| 35            | 64/32                               | 118000h–117FFFh        |
| 34            | 64/32                               | 110000h–117FFFh        |
| 33            | 64/32                               | 108000h–107FFFh        |
| 32            | 64/32                               | 100000h–107FFFh        |
| 31            | 64/32                               | 0F8000h–0FFFFFh        |
| 30            | 64/32                               | 0F0000h–0F7FFFh        |
| 29            | 64/32                               | 0E8000h–0E7FFFh        |
| 28            | 64/32                               | 0E0000h–0E7FFFh        |
| 27            | 64/32                               | 0D8000h–0DFFFFh        |
| 26            | 64/32                               | 0D0000h–0D7FFFh        |
| 25            | 64/32                               | 0C8000h–0CFFFFh        |
| 24            | 64/32                               | 0C0000h–0C7FFFh        |
| 23            | 64/32                               | 0B8000h–0BFFFFh        |
| 22            | 64/32                               | 0B0000h–0B7FFFh        |
| 21            | 64/32                               | 0A8000h–0AFFFFh        |
| 20            | 64/32                               | 0A0000h–0A7FFFh        |
| 19            | 64/32                               | 098000h–097FFFh        |
| 18            | 64/32                               | 090000h–097FFFh        |
| 17            | 64/32                               | 088000h–08FFFFh        |
| 16            | 64/32                               | 080000h–087FFFh        |
| 15            | 64/32                               | 078000h–077FFFh        |
| 14            | 64/32                               | 070000h–077FFFh        |
| 13            | 64/32                               | 068000h–06FFFFh        |
| 12            | 64/32                               | 060000h–067FFFh        |
| 11            | 64/32                               | 058000h–05FFFFh        |
| 10            | 64/32                               | 050000h–057FFFh        |
| 9             | 64/32                               | 048000h–04FFFFh        |
| 8             | 64/32                               | 040000h–047FFFh        |
| 7             | 64/32                               | 038000h–03FFFFh        |
| 6             | 64/32                               | 030000h–037FFFh        |
| 5             | 64/32                               | 028000h–02FFFFh        |
| 4             | 64/32                               | 020000h–027FFFh        |
| 3             | 64/32                               | 018000h–01FFFFh        |
| 2             | 64/32                               | 010000h–017FFFh        |
| 1             | 64/32                               | 008000h–00FFFFh        |
| 0             | 64/32                               | 000000h–007FFFh        |


**BALL DESCRIPTIONS**

| 59-BALL FBGA<br>NUMBERS                                                                               | SYMBOL          | TYPE             | DESCRIPTION                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------------------------------------------------------------------------------------|-----------------|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| E8, D8, C8, B8,<br>A8, B7, A7, C7,<br>A2, B2, C2, A1,<br>B1, C1, D2, D1,<br>D4, B6, A6, C6,<br>B3, C3 | A0–A21          | Input            | Address Inputs: Inputs for the addresses during READ and WRITE operations. Addresses are internally latched during READ and WRITE cycles.                                                                                                                                                                                                                                                                                                                                                     |
| B4                                                                                                    | CLK             | Input            | Clock: Synchronizes the Flash memory to the system operating frequency during synchronous burst mode READ operations. When configured for synchronous burst mode READs, address is latched on the first rising (or falling, depending upon the read configuration register setting) CLK edge when ADV# is active or upon a rising ADV# edge, whichever occurs first. CLK is ignored during asynchronous access READ and WRITE operations and during READ PAGE ACCESS operations. <sup>1</sup> |
| C4                                                                                                    | ADV#            | Input            | Address Valid: Indicates that a valid address is present on the address inputs. Addresses are latched on the rising edge of ADV# during READ and WRITE operations. ADV# may be tied active during asynchronous READ and WRITE operations. <sup>1</sup>                                                                                                                                                                                                                                        |
| A5                                                                                                    | V <sub>PP</sub> | Input            | Program/Erase Enable: [0.9V–2.20V or 11.4V–12.6V] Operates as input at logic levels to control complete device protection. Provides factory programming compatibility when driven to 11.4V–12.6V.                                                                                                                                                                                                                                                                                             |
| E7                                                                                                    | CE#             | Input            | Chip Enable: Activates the device when LOW. When CE# is HIGH, the device is disabled and goes into standby power mode.                                                                                                                                                                                                                                                                                                                                                                        |
| F8                                                                                                    | OE#             | Input            | Output Enable: Enables the output buffers when LOW. When OE# is HIGH, the output buffers are disabled.                                                                                                                                                                                                                                                                                                                                                                                        |
| C5                                                                                                    | WE#             | Input            | Write Enable: Determines if a given cycle is a WRITE cycle. If WE# is LOW, the cycle is either a WRITE to the command state machine (CSM) or to the memory array.                                                                                                                                                                                                                                                                                                                             |
| B5                                                                                                    | RST#            | Input            | Reset: When RST# is a logic LOW, the device is in reset mode, which drives the outputs to High-Z and resets the write state machine (WSM). When RST# is at logic HIGH, the device is in standard operation. When RST# transitions from logic LOW to logic HIGH, the device resets all blocks to locked and defaults to the read array mode.                                                                                                                                                   |
| D6                                                                                                    | WP#             | Input            | Write Protect: Controls the lock down function of the flexible locking feature.                                                                                                                                                                                                                                                                                                                                                                                                               |
| F7, E6, E5, G5,<br>E4, G3, E3, G1,<br>G7, F6, F5, F4,<br>D5, F3, F2, E2                               | DQ0–DQ15        | Input/<br>Output | Data Inputs/Outputs: Inputs array data on the second CE# and WE# cycle during PROGRAM command. Inputs commands to the command user interface when CE# and WE# are active. DQ0–DQ15 output data when CE# and OE# are active.                                                                                                                                                                                                                                                                   |
| D3                                                                                                    | WAIT#           | Output           | Wait: Provides data valid feedback during continuous burst read access. The signal is gated by OE# and CE#. This signal is always kept at a valid logic level.                                                                                                                                                                                                                                                                                                                                |

**NOTE:** 1. The CLK and ADV# inputs can be tied to V<sub>SS</sub> if the device is always operating in asynchronous or page mode. The WAIT# signal can be ignored when operating in asynchronous or page mode, as it is always held at logic "1" or "0," depending on the RCR8 setting (see Table 9).

(continued on next page)


**BALL DESCRIPTIONS (continued)**

| 59-BALL FBGA<br>NUMBERS | SYMBOL           | TYPE   | DESCRIPTION                                                                                                         |
|-------------------------|------------------|--------|---------------------------------------------------------------------------------------------------------------------|
| A4, G4                  | V <sub>CC</sub>  | Supply | Device Power Supply: [1.70V–1.90V (MT28F642D18) or 1.80V–2.20V (MT28F642D20)] Supplies power for device operation.  |
| E1, G6                  | V <sub>CCQ</sub> | Supply | I/O Power Supply: [1.70V–1.90V (MT28F642D18) or 1.80V–2.25V (MT28F642D20)] Supplies power for input/output buffers. |
| G2, G8                  | V <sub>SSQ</sub> | Supply | I/O Ground. Do not float any ground ball.                                                                           |
| A3, F1                  | V <sub>SS</sub>  | Supply | Do not float any ground ball.                                                                                       |
| D7                      | –                | –      | Contact ball is not physically present.                                                                             |

## COMMAND STATE MACHINE (CSM)

Commands are issued to the command state machine (CSM) using standard microprocessor write timings. The CSM acts as an interface between external microprocessors and the internal write state machine (WSM). The available commands are listed in Table 3, their definitions are given in Table 4, and their descriptions in Table 5. Program and erase algorithms are automated by an on-chip WSM. Table 6 shows the CSM transition states. Once a valid PROGRAM/ERASE command is entered, the WSM executes the appropriate algorithm. The algorithm generates the necessary timing signals to control the device internally and accomplish the requested operation. A command is valid only if the exact sequence of WRITES is completed. After the WSM completes its task, the WSM status bit (SR7) is set to a logic HIGH level (1) (see Table 8), allowing the CSM to respond to the full command set again.

## OPERATIONS

Device operations are selected by entering a standard JEDEC 8-bit command code with conventional microprocessor timings into an on-chip CSM through I/Os DQ0–DQ7. The number of bus cycles required to activate a command is typically one or two. The first operation is always a WRITE. Control signals CE#, ADV#, and WE# must be at a logic LOW level ( $V_{IL}$ ), and OE# and RST# must be at logic HIGH ( $V_{IH}$ ). The second operation, when

needed, can be a WRITE or a READ depending upon the command. During a READ operation, control signals CE#, ADV#, and OE# must be at a logic LOW level ( $V_{IL}$ ), and WE# and RST# must be at logic HIGH ( $V_{IH}$ ).

Table 7 illustrates the bus operations for all the modes: write, read, reset, standby, and output disable.

When the device is powered up, internal reset circuitry initializes the chip to a read array mode of operation. Changing the mode of operation requires that a command code be entered into the CSM. For each of the memory partitions, an on-chip status register is available. These two registers enable the progress of various operations that take place on a memory bank to be monitored. Either of the two status registers is interrogated by entering a READ STATUS REGISTER command onto the CSM (cycle 1), specifying an address within the memory partition boundary, and reading the register data on I/Os DQ0–DQ7 (cycle 2). Status register bits SR0–SR7 correspond to DQ0–DQ7 (see Table 8).

## COMMAND DEFINITION

Once a specific command code has been entered, the WSM executes an internal algorithm, generating the necessary timing signals to program, erase, and verify data. See Table 4 for the CSM command definitions and data for each of the bus cycles.

**Table 3**  
**Command State Machine Codes For Device Mode Selection**

| COMMAND DQ0–DQ7 | CODE ON DEVICE MODE                     |
|-----------------|-----------------------------------------|
| 10h             | Accelerated programming algorithm (APA) |
| 20h             | Block erase setup                       |
| 40h             | Program setup                           |
| 50h             | Clear status register                   |
| 60h             | Protection configuration setup          |
| 60h             | Set read configuration register         |
| 70h             | Read status register                    |
| 90h             | Read protection configuration register  |
| 98h             | Read query                              |
| B0h             | Program/erase suspend                   |
| C0h             | Protection register program/lock        |
| D0h             | Program/erase resume – erase confirm    |
| D1h             | Check block erase confirm               |
| FFh             | Read array                              |



## STATUS REGISTER

The status register allows the user to determine whether the state of a PROGRAM/ERASE operation is pending or complete. The status register is monitored by toggling OE# and CE# and reading the resulting status code on I/Os DQ0–DQ7. The high-order I/Os (DQ8–DQ15) are set to 00h internally, so only the low-order I/Os (DQ0–DQ7) need to be interpreted. Address lines select the status register pertinent to the selected memory partition.

Register data is updated and latched on the falling edge of ADV# or the rising (falling) edge of CLK when ADV# is LOW during synchronous burst mode, or on the falling edge of OE# or CE#, whichever occurs last. Latching the data prevents errors from occurring if the register input changes during status register monitoring.

The status register provides a reading of the internal state of the WSM to the external microprocessor.

During periods when the WSM is active, the status register can be polled to determine the WSM status. Table 8 defines the status register bits.

After monitoring the status register during a PROGRAM/ERASE operation, the data appearing on DQ0–DQ7 remains as status register data until a new command is issued to the CSM. To return the device to other modes of operation, a new command must be issued to the CSM.

## COMMAND STATE MACHINE OPERATIONS

The CSM decodes instructions for the commands listed in Table 3. The 8-bit command code is input to the device on DQ0–DQ7 (see Table 3 for CSM codes and Table 4 for command definitions). During a PROGRAM or ERASE cycle, the CSM informs the WSM that a PROGRAM or ERASE cycle has been requested.

**Table 4**  
**Command Definitions**

| COMMAND                                 | FIRST BUS CYCLE |                      |      | SECOND BUS CYCLE |                      |      |
|-----------------------------------------|-----------------|----------------------|------|------------------|----------------------|------|
|                                         | OPERATION       | ADDRESS <sup>1</sup> | DATA | OPERATION        | ADDRESS <sup>1</sup> | DATA |
| READ ARRAY                              | WRITE           | WA                   | FFh  |                  |                      |      |
| READ PROTECTION CONFIGURATION REGISTER  | WRITE           | IA                   | 90h  | READ             | IA                   | ID   |
| READ STATUS REGISTER                    | WRITE           | BA                   | 70h  | READ             | X                    | SRD  |
| CLEAR STATUS REGISTER                   | WRITE           | BA                   | 50h  |                  |                      |      |
| READ QUERY                              | WRITE           | QA                   | 98h  | READ             | QA                   | QD   |
| BLOCK ERASE SETUP                       | WRITE           | BA                   | 20h  | WRITE            | BA                   | D0h  |
| PROGRAM SETUP                           | WRITE           | WA                   | 40h  | WRITE            | WA                   | WD   |
| ACCELERATED PROGRAMMING ALGORITHM (APA) | WRITE           | WA                   | 10h  | WRITE            | WA                   | WD   |
| PROGRAM/ERASE SUSPEND                   | WRITE           | BA                   | B0h  |                  |                      |      |
| PROGRAM/ERASE RESUME – ERASE CONFIRM    | WRITE           | BA                   | D0h  |                  |                      |      |
| LOCK BLOCK                              | WRITE           | BA                   | 60h  | WRITE            | BA                   | 01h  |
| UNLOCK BLOCK                            | WRITE           | BA                   | 60h  | WRITE            | BA                   | D0h  |
| LOCK DOWN BLOCK                         | WRITE           | BA                   | 60h  | WRITE            | BA                   | 2Fh  |
| CHECK BLOCK ERASE                       | WRITE           | BA                   | 20h  | WRITE            | BA                   | D1h  |
| PROTECTION REGISTER PROGRAM             | WRITE           | PA                   | C0h  | WRITE            | PA                   | PD   |
| PROTECTION REGISTER LOCK                | WRITE           | LPA                  | C0h  | WRITE            | LPA                  | FFDh |
| SET READ CONFIGURATION REGISTER         | WRITE           | RCD                  | 60h  | WRITE            | RCD                  | 03h  |

**NOTE:** 1. WA: Word address of memory location to be written, or read  
 IA: Identification code address  
 BA: Address within the block  
 ID: Identification code data  
 SRD: Data read from the status register  
 QA: Query code address  
 QD: Query code data

WD: Data to be written at the location WA  
 PA: Protection register address  
 PD: Data to be written at the location PA  
 LPA: Lock protection register address  
 RCD: Data to be written in the read configuration register  
 X: "Don't Care"



**Table 5  
Command Descriptions**

| CODE | DEVICE MODE                     | BUS CYCLE | DESCRIPTION                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|------|---------------------------------|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 10h  | APA                             | First     | Prepares the CSM for an ACCELERATED PROGRAM ALGORITHM (APA) command.                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| 20h  | Erase Setup                     | First     | Prepares the CSM for the ERASE command. If the next command is not a CHECK BLOCK ERASE or ERASE CONFIRM command, the command will be ignored, and the bank will go to the read status mode and wait for another command.                                                                                                                                                                                                                                                                                                                |
| 40h  | Program Setup                   | First     | A two-cycle command: The first cycle prepares for a PROGRAM operation, and the second cycle latches addresses and data and initiates the WSM to execute the program algorithm. The Flash outputs status register data on the rising edge of ADV#, or on the rising clock edge when ADV# is LOW during synchronous burst mode, or on the falling edge of OE# or CE#, whichever occurs first.                                                                                                                                             |
| 50h  | Clear Status Register           | First     | The WSM can set the block lock status (SR1), VPP status (SR3), program status (SR4), and erase status (SR5) bits in the status register to "1," but it cannot clear them to "0." Issuing this command clears those bits to "0."                                                                                                                                                                                                                                                                                                         |
| 60h  | Protection Configuration Setup  | First     | Prepares the CSM for changes to the block locking status. If the next command is not BLOCK UNLOCK, BLOCK LOCK, or BLOCK LOCK DOWN, the command will be ignored, and the device will go to the read status mode.                                                                                                                                                                                                                                                                                                                         |
|      | Set Read Configuration Register | First     | Puts the device into the set read configuration mode so that it will be possible to set the option bits related to burst read mode.                                                                                                                                                                                                                                                                                                                                                                                                     |
| 70h  | Read Status Register            | First     | Places the device into a read status register mode. Reading the device will output the contents of the status register for the addressed bank. The device will automatically enter this mode for the addressed bank after a PROGRAM or ERASE operation has been initiated.                                                                                                                                                                                                                                                              |
| 90h  | Read Protection Configuration   | First     | Puts the device into the read protection configuration mode so that reading the device will output the manufacturer/device codes, block lock status, protection register, or protection register lock status.                                                                                                                                                                                                                                                                                                                           |
| 98h  | Read Query                      | First     | Puts the device into the read query mode so that reading the device will output common flash interface information.                                                                                                                                                                                                                                                                                                                                                                                                                     |
| B0h  | Program Suspend                 | First     | Issuing this command will suspend the currently executing PROGRAM/ERASE/CHECK BLOCK ERASE operation. The status register will indicate when the operation has been successfully suspended by setting either the program suspend (SR2) or erase suspend (SR6) bit, and the WSM status bit (SR7) to a "1" (ready). The WSM will continue to idle in the suspend state, regardless of the state of all input control signals except RST#, which will immediately shut down the WSM and the remainder of the chip if RST# is driven to VIL. |
|      | Erase Suspend                   |           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|      | Check Block Erase Suspend       |           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |

(continued on next page)



**Table 5**  
**Command Descriptions (continued)**

| CODE | DEVICE MODE                            | BUS CYCLE | DESCRIPTION                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|------|----------------------------------------|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| C0h  | Program Device Protection Register     | First     | Writes a specific code into the device protection register.                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|      | Lock Device Protection Register        | First     | Locks the device protection register; data can no longer be changed.                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| D0h  | Erase Confirm                          | Second    | If the previous command was an ERASE SETUP command, then the CSM will close the address and data latches, and it will begin erasing the block indicated on the address bus. During programming/erase, the device will respond only to the READ STATUS REGISTER, PROGRAM SUSPEND, or ERASE SUSPEND command. It will output status register data on the rising edge of ADV#, or on the rising clock edge when ADV# is LOW during synchronous burst mode, or on the falling edge of OE# or CE#, whichever occurs last. |
|      | Program/Erase/Check Block Erase Resume | First     | If a PROGRAM, ERASE or CHECK BLOCK ERASE operation was previously suspended, this command will resume the operation.                                                                                                                                                                                                                                                                                                                                                                                                |
| FFh  | Read Array                             | First     | During read array mode, array data will be output on the data bus.                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| 01h  | Lock Block                             | Second    | If the previous command was PROTECTION CONFIGURATION SETUP, the CSM will latch the address and lock the block indicated on the address bus.                                                                                                                                                                                                                                                                                                                                                                         |
| 03h  | Read Configuration Register Data       | Second    | If the previous command was SET READ CONFIGURATION REGISTER, the configuration bits presented on the address bus will be stored into the read configuration register.                                                                                                                                                                                                                                                                                                                                               |
| 2Fh  | Lock Down                              | Second    | If the previous command was PROTECTION CONFIGURATION SETUP, the CSM will latch the address and lock down the block indicated on the address bus.                                                                                                                                                                                                                                                                                                                                                                    |
| D0h  | Unlock Block                           | Second    | If the previous command was PROTECTION CONFIGURATION SETUP, the CSM will latch the address and unlock the block indicated on the address bus. If the block had been previously set to lock down, this operation will have no effect.                                                                                                                                                                                                                                                                                |
| D1h  | Check Block Erase Confirm              | Second    | If the previous command was ERASE SETUP, the CSM will close the address latches and check to see that the block is completely erased.                                                                                                                                                                                                                                                                                                                                                                               |
| 00h  | Invalid/Reserved                       |           | Unassigned command that should not be used.                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |

During a PROGRAM cycle, the WSM controls the program sequences and the CSM responds to a PROGRAM SUSPEND command only.

During an ERASE cycle, the CSM responds to an ERASE SUSPEND command only. When the WSM has completed its task, the WSM status bit (SR7) is set to a logic HIGH level and the CSM responds to the full command set. The CSM stays in the current command state until the microprocessor issues another command.

The WSM successfully initiates an ERASE or PROGRAM operation only when  $V_{PP}$  is within its correct voltage range.

## **CLEAR STATUS REGISTER**

The internal circuitry can set, but not clear, the block lock status bit (SR1), the  $V_{PP}$  status bit (SR3), the program status bit (SR4), and the erase status bit (SR5) of the status register. The CLEAR STATUS REGISTER command (50h) allows the external microprocessor to clear these status bits and synchronize to the internal operations. When the status bits are cleared, the device returns to the read array mode.

## **READ OPERATIONS**

The following READ operations are available: READ ARRAY, READ PROTECTION CONFIGURATION REGISTER, READ QUERY and READ STATUS REGISTER.

### **READ ARRAY**

The array is read by entering the command code FFh on DQ0–DQ7. Control signals CE#, ADV#, and OE# must be at a logic LOW level ( $V_{IL}$ ) and WE# and RST# must be at a logic HIGH level ( $V_{IH}$ ) to read data from the array. Data is available on DQ0–DQ15. Any valid address within any of the blocks selects that address and allows data to be read from that address. Upon initial power-up or device reset, the device defaults to the read array mode.

## **READ PROTECTION CONFIGURATION DATA**

The read protection configuration mode outputs five types of information: the manufacturer/device identifier, the block locking status, the read configuration register, the protection register, and PR lock status. Two bus cycles are required for this operation: the chip identification data is read by entering the command code 90h on DQ0–DQ7 and the identification code address on the address lines. Control signals CE#, ADV#, and OE# must be at a logic LOW level ( $V_{IL}$ ), and WE# and RST# must be at a logic HIGH level ( $V_{IH}$ ) to read data from the protection configuration register. Data is available on DQ0–DQ15. After data is read from the protection configuration register, the READ ARRAY command, FFh, must be issued to the bank containing address 00h prior to issuing other commands. See Table 13 for further details.

### **READ QUERY**

The read query mode outputs common flash interface (CFI) data when the device is read (see Table 15). Two bus cycles are required for this operation. It is possible to access the query by writing the read query command code 98h on DQ0–DQ7 to the bank containing address 0h. Control signals CE#, ADV#, and OE# must be at a logic LOW level ( $V_{IL}$ ) and WE# and RST# must be at a logic HIGH level ( $V_{IH}$ ) to read data from the query. The CFI data structure contains information such as block size, density, command set, and electrical specifications. To return to read array mode, write the read array command code FFh on DQ0–DQ7.

### **READ STATUS REGISTER**

The status register is read by entering the command code 70h on DQ0–DQ7. Two bus cycles are required for this operation: one to enter the command code and a second to read the status register. The addresses for both cycles must be in the same partition. In a READ cycle, the address is latched on the rising edge of the ADV# signal. Register data is updated and latched on the falling edge of ADV# or the rising (falling) CLK when ADV# is LOW during burst mode, or on the falling edge of OE# or CE#, whichever occurs last.

**Table 6**  
**Command State Machine Transition Table**

| Command input to the present partition (and next state of the present partition) |                       |                  |               |                                             |                |                    |                           |                 |                            |                                         |                 |                       |                | Present state of the present partition |                |       |                     | Present state of the other partition |      |      |  |  |  |
|----------------------------------------------------------------------------------|-----------------------|------------------|---------------|---------------------------------------------|----------------|--------------------|---------------------------|-----------------|----------------------------|-----------------------------------------|-----------------|-----------------------|----------------|----------------------------------------|----------------|-------|---------------------|--------------------------------------|------|------|--|--|--|
| 03h Write RCR confirm                                                            | 2Fh Lock down confirm | 01h Lock confirm | C0h OTP setup | 60h Lock/Unlock /Lock down, write RCR setup | 98h Read query | 90h Read device ID | 50h Clear status register | 70h Read status | 80h Program /Erase suspend | D0h BE confirm, P/E resume, ULB confirm | 20h Erase setup | 10h/40h Program setup | FFh Read array | SR7                                    | Data when read | State | Mode                |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  |               | Lock/RCR setup                              | Read query     | Read ID            | Read array                | Read status     | Read array                 |                                         |                 |                       | 1              | Array                                  | Array          | Read  | 1                   | Setup                                |      |      |  |  |  |
| Read array                                                                       |                       |                  | OTP setup     |                                             |                |                    |                           |                 | Read array                 |                                         | Erase setup     | Program setup         |                |                                        |                |       | Read array          |                                      | 2    | Busy |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 3                          | Idle                                    |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 4                          | Erase suspend                           |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 5                          | Prog. suspend                           |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  |               | Lock/RCR setup                              | Read query     | Read ID            | Read array                | Read status     | Read array                 |                                         |                 |                       | 1              | CFI                                    | Query          |       | 6                   | Setup                                |      |      |  |  |  |
| Read array                                                                       |                       |                  | OTP setup     |                                             |                |                    |                           |                 | Read array                 |                                         | Erase setup     | Program setup         |                |                                        |                |       | Read array          |                                      | 7    | Busy |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 8                          | Idle                                    |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 9                          | Erase suspend                           |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 10                         | Prog. suspend                           |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  |               | Lock/RCR setup                              | Read query     | Read ID            | Read array                | Read status     | Read array                 |                                         |                 |                       | 1              | ID                                     | Device ID      |       | 11                  | Setup                                |      |      |  |  |  |
| Read array                                                                       |                       |                  | OTP setup     |                                             |                |                    |                           |                 | Read array                 |                                         | Erase setup     | Program setup         |                |                                        |                |       | Read array          |                                      | 12   | Busy |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 13                         | Idle                                    |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 14                         | Erase suspend                           |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 15                         | Prog. suspend                           |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  |               | Lock/RCR setup                              | Read query     | Read ID            | Read array                | Read status     | Read array                 |                                         |                 |                       | 1              | Status                                 | Status         |       | 16                  | Setup                                |      |      |  |  |  |
| Read array                                                                       |                       |                  | OTP setup     |                                             |                |                    |                           |                 | Read array                 |                                         | Erase setup     | Program setup         |                |                                        |                |       | Read array          |                                      | 17   | Busy |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 18                         | Idle                                    |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 19                         | Erase suspend                           |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Read array                                                                       |                       |                  | Read array    |                                             |                |                    |                           |                 | 20                         | Prog. suspend                           |                 |                       |                |                                        |                |       |                     |                                      |      |      |  |  |  |
| Protection register busy                                                         |                       |                  |               |                                             |                |                    |                           |                 |                            |                                         |                 |                       |                | 1                                      | Status         | Setup | Protection register | 21                                   | Idle |      |  |  |  |
| Protection register busy                                                         |                       |                  |               |                                             |                |                    |                           |                 |                            |                                         |                 |                       |                | 0                                      | Status         | Busy  |                     | 22                                   | Idle |      |  |  |  |
| Read array                                                                       |                       |                  |               | Lock/RCR setup                              | Read query     | Read ID            | Read array                | Read status     | Read array                 |                                         |                 |                       | 1              | Status                                 | Done           | 23    |                     | Setup                                |      |      |  |  |  |
| Read array                                                                       |                       |                  | OTP setup     | Lock/RCR setup                              | Read query     | Read ID            | Read array                | Read status     | Read array                 |                                         | Erase setup     | Program setup         | Read array     | 1                                      | Status         | Done  |                     | 24                                   | Busy |      |  |  |  |
| Read array                                                                       |                       |                  |               |                                             |                |                    |                           |                 | Read array                 |                                         | 25              |                       |                |                                        |                |       |                     | Idle                                 |      |      |  |  |  |
| Read array                                                                       |                       |                  |               |                                             |                |                    |                           |                 | Read array                 |                                         | 26              |                       |                |                                        |                |       |                     | Erase suspend                        |      |      |  |  |  |
| Read array                                                                       |                       |                  |               |                                             |                |                    |                           |                 | Read array                 |                                         | 27              | Prog. suspend         |                |                                        |                |       |                     |                                      |      |      |  |  |  |

(continued on the next page)



**Table 6**  
**Command State Machine Transition Table (continued)**

| Command input to the present partition (and next state of the present partition) |                       |                  |                |                                             |                            |                         |                            |                             |                            |                                         |                            |                       |                | Present state of the present partition |                |             |               | Present state of the other partition |               |               |  |  |  |  |  |  |  |  |  |  |  |
|----------------------------------------------------------------------------------|-----------------------|------------------|----------------|---------------------------------------------|----------------------------|-------------------------|----------------------------|-----------------------------|----------------------------|-----------------------------------------|----------------------------|-----------------------|----------------|----------------------------------------|----------------|-------------|---------------|--------------------------------------|---------------|---------------|--|--|--|--|--|--|--|--|--|--|--|
| 03h Write RCR confirm                                                            | 2Fh Lock down confirm | 01h Lock confirm | C0H OTP setup  | 60h Lock/Unlock /Lock down, write RCR setup | 98h Read query             | 90h Read device ID      | 50h Clear status register  | 70h Read status             | 80h Program /Erase suspend | D0h BE confirm, P/E resume, ULB confirm | 20h Erase setup            | 10h/40h Program setup | FFh Read array | SR7                                    | Data when read | Status      | Mode          |                                      |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Set RCR                                                                          | LB/ULB                |                  | Lock/RCR Error |                                             |                            |                         |                            |                             | Lock/ RCR Error            | LB/ULB                                  |                            | Lock / RCR Error      |                |                                        | 1              | Status      | Setup         | Lock/ RCR                            | 28            | Any state     |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                | Lock/RCR setup                              | Read query                 | Read ID                 | Read array                 | Read status                 | Read array                 |                                         |                            |                       | 1              | Status                                 | Error          | 29          | Setup         |                                      |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  | OTP setup      |                                             |                            |                         |                            |                             | Read array                 |                                         | Erase setup                | Program setup         |                |                                        |                | Read array  | 30            |                                      | Busy          |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         | 31                         |                       |                |                                        |                |             | Idle          |                                      |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         |                            |                       |                |                                        |                |             | 32            |                                      | Erase suspend |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                | Lock/RCR setup                              | Read query                 | Read ID                 | Read array                 | Read status                 | Read array                 |                                         |                            |                       | 1              | Status                                 | Lock/ Unlock   | 33          | Prog. suspend |                                      |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  | OTP setup      |                                             |                            |                         |                            |                             | Read array                 |                                         | Erase setup                | Program setup         |                |                                        |                | Read array  | 34            |                                      | Setup         |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         | 35                         |                       |                |                                        |                |             | Busy          |                                      |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         |                            |                       |                |                                        |                |             | 36            |                                      | Idle          |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                | Lock/RCR setup                              | Read query                 | Read ID                 | Read array                 | Read status                 | Read array                 |                                         |                            |                       | 1              | Array                                  | Set RCR        | 37          | Erase suspend |                                      |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  | OTP setup      |                                             |                            |                         |                            |                             | Read array                 |                                         | Erase setup                | Program setup         |                |                                        |                | Read array  | 38            | Prog. suspend                        |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         |                            |                       |                |                                        |                |             | 39            | Setup                                |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         |                            |                       |                |                                        |                |             | 40            | Busy                                 |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                | Lock/RCR setup                              | Read query                 | Read ID                 | Read array                 | Read status                 | Read array                 |                                         |                            |                       | 1              | Array                                  | Set RCR        | 41          | Idle          |                                      |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  | OTP setup      |                                             |                            |                         |                            |                             | Read array                 |                                         | Erase setup                | Program setup         |                |                                        |                | Read array  | 42            | Erase suspend                        |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         |                            |                       |                |                                        |                |             | 43            | Prog. suspend                        |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         |                            |                       |                |                                        |                |             |               |                                      |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Program Busy                                                                     |                       |                  |                |                                             |                            |                         |                            |                             |                            |                                         |                            |                       |                | 1                                      | Status         | Setup       | Program       | 44                                   | Any state     |               |  |  |  |  |  |  |  |  |  |  |  |
| Program Busy                                                                     |                       |                  |                |                                             |                            |                         |                            |                             | PS read status             | Program busy                            |                            |                       | 0              | Status                                 | Busy           | 45          |               | Idle                                 |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                | Lock/RCR setup                              | Read query                 | Read ID                 | Read array                 | Read status                 | Read array                 |                                         |                            |                       | 1              | Status                                 | Done           | 46          |               | Setup                                |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  | OTP setup      |                                             |                            |                         |                            |                             | Read array                 |                                         | Erase setup                | Program setup         |                |                                        |                | Read array  |               | 47                                   | Busy          |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         | 48                         |                       |                |                                        |                |             |               | Idle                                 |               |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         |                            |                       |                |                                        |                |             |               | 49                                   | Erase suspend |               |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                | Lock/RCR setup                              | Program suspend read query | Program suspend read ID | Program suspend read array | Program suspend read status | Program suspend read array | Program busy                            | Program suspend read array |                       |                | 1                                      | Status         | Read status |               | Program suspend                      | 50            | Prog. suspend |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  | OTP setup      |                                             |                            |                         |                            |                             |                            |                                         | Read array                 |                       |                |                                        |                |             |               |                                      | 51            | Setup         |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             |                            |                                         | Read array                 |                       |                |                                        |                |             |               |                                      | 52            | Idle          |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  | OTP setup      |                                             |                            |                         |                            |                             |                            |                                         | Read array                 |                       |                |                                        |                |             |               |                                      | 53            | Erase suspend |  |  |  |  |  |  |  |  |  |  |  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             |                            |                                         | Read array                 |                       |                |                                        |                |             |               |                                      |               |               |  |  |  |  |  |  |  |  |  |  |  |

(continued on the next page)



**Table 6**  
**Command State Machine Transition Table (continued)**

| Command input to the present partition (and next state of the present partition) |                       |                  |                |                                             |                            |                         |                            |                             |                            |                                         |                            |                       |                | Present state of the present partition |                |            |                 | Present state of the other partition |               |       |
|----------------------------------------------------------------------------------|-----------------------|------------------|----------------|---------------------------------------------|----------------------------|-------------------------|----------------------------|-----------------------------|----------------------------|-----------------------------------------|----------------------------|-----------------------|----------------|----------------------------------------|----------------|------------|-----------------|--------------------------------------|---------------|-------|
| 03h Write RCR confirm                                                            | 2Fh Lock down confirm | 01h Lock confirm | C0h OTP setup  | 60h Lock/Unlock /Lock down, write RCR setup | 98h Read query             | 90h Read device ID      | 50h Clear status register  | 70h Read status             | 80h Program /Erase suspend | D0h BE confirm, P/E resume, ULB confirm | 20h Erase setup            | 10h/40h Program setup | FFh Read array | SR7                                    | Data when read | State      | Mode            |                                      |               |       |
| Program suspend read array                                                       |                       |                  |                | Lock/RCR setup                              | Program suspend read query | Program suspend read ID | Program suspend read array | Program suspend read status | Program suspend read array | Program busy                            | Program suspend read array |                       |                | 1                                      | Array          | Read array | Program suspend | 54                                   | Setup         |       |
|                                                                                  |                       |                  |                |                                             |                            |                         |                            |                             |                            |                                         |                            |                       |                |                                        |                |            |                 | 55                                   | Idle          |       |
| Program suspend read array                                                       |                       |                  |                | Lock/RCR setup                              | Program suspend read query | Program suspend read ID | Program suspend read array | Program suspend read status | Program suspend read array | Program busy                            | Program suspend read array |                       |                | 1                                      | ID             | Read ID    |                 | 56                                   | Erase suspend |       |
|                                                                                  |                       |                  |                |                                             |                            |                         |                            |                             |                            |                                         |                            |                       |                |                                        |                |            |                 | 57                                   | Setup         |       |
| Program suspend read array                                                       |                       |                  |                | Lock/RCR setup                              | Program suspend read query | Program suspend read ID | Program suspend read array | Program suspend read status | Program suspend read array | Program busy                            | Program suspend read array |                       |                | 1                                      | CFI            | Read Query |                 | 58                                   | Idle          |       |
|                                                                                  |                       |                  |                |                                             |                            |                         |                            |                             |                            |                                         |                            |                       |                |                                        |                |            |                 | 59                                   | Erase suspend |       |
|                                                                                  |                       |                  |                |                                             |                            |                         |                            |                             |                            |                                         |                            |                       |                |                                        |                |            |                 | 60                                   | Setup         |       |
|                                                                                  |                       |                  |                |                                             |                            |                         |                            |                             |                            |                                         |                            |                       |                |                                        |                |            |                 | 61                                   | Idle          |       |
|                                                                                  |                       |                  |                |                                             |                            |                         |                            |                             |                            |                                         |                            |                       |                |                                        |                |            | 62              | Erase suspend                        |               |       |
| Set RCR                                                                          | LB/ULB                |                  | Lock/RCR Error |                                             |                            |                         |                            |                             | Lock/RCR Error             | LB/ULB                                  | Lock/RCR Error             |                       |                | 1                                      | Status         | Setup      | Erase           | 63                                   | Idle          |       |
| Read array                                                                       |                       |                  |                | Lock/RCR setup                              | Read query                 | Read ID                 | Read array                 | Read status                 | Read array                 |                                         |                            |                       |                | 1                                      | Status         | Error      |                 | 64                                   | Setup         |       |
| Read array                                                                       |                       |                  | OTP setup      |                                             |                            |                         |                            |                             | Read array                 |                                         | Erase setup                | Program setup         |                |                                        |                |            |                 | Read array                           | 65            | Busy  |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         |                            |                       |                |                                        |                |            |                 | 66                                   | Idle          |       |
|                                                                                  |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         |                            |                       |                |                                        |                |            |                 | 67                                   | Erase suspend |       |
| Read array                                                                       |                       |                  |                | Lock/RCR setup                              | Read query                 | Read ID                 | Read array                 | Read status                 | Read array                 |                                         |                            |                       |                | 1                                      | Status         | Done       |                 | 68                                   | Prog. suspend |       |
| Read array                                                                       |                       |                  | OTP setup      |                                             |                            |                         |                            |                             | Read array                 |                                         | Erase setup                | Program setup         |                |                                        |                |            |                 | Read array                           | 69            | Setup |
| Read array                                                                       |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         |                            |                       |                |                                        |                |            |                 | 70                                   | Busy          |       |
|                                                                                  |                       |                  |                |                                             |                            |                         |                            |                             | Read array                 |                                         |                            |                       |                |                                        |                |            |                 | 71                                   | Idle          |       |
|                                                                                  |                       |                  |                | Lock/RCR setup                              | Read query                 | Read ID                 | Read array                 | Read status                 | Read array                 |                                         | Erase setup                | Program setup         | Read array     | 1                                      | Status         | Done       |                 | 72                                   | Erase suspend |       |
| Read array                                                                       |                       |                  | Read array     |                                             |                            |                         |                            |                             | 73                         | Prog. suspend                           |                            |                       |                |                                        |                |            |                 |                                      |               |       |
| Block erase busy                                                                 |                       |                  |                |                                             |                            |                         |                            |                             | ES read status             | Erase busy                              |                            |                       |                |                                        |                |            |                 | 0                                    | Status        | Busy  |

(continued on the next page)



**Table 6**  
**Command State Machine Transition Table (continued)**

| Command input to the present partition (and next state of the present partition) |                       |                          |                       |                                             |                           |                          |                           |                          |                            |                                         |                 |                       |                | Present state of the present partition |                |       |      | Present state of the other partition |  |
|----------------------------------------------------------------------------------|-----------------------|--------------------------|-----------------------|---------------------------------------------|---------------------------|--------------------------|---------------------------|--------------------------|----------------------------|-----------------------------------------|-----------------|-----------------------|----------------|----------------------------------------|----------------|-------|------|--------------------------------------|--|
| 03h Write RCR confirm                                                            | 2Fh Lock down confirm | 01h Lock confirm         | C0h OTP setup         | 60h Lock/Unlock /Lock down, write RCR setup | 98h Read query            | 90h Read device ID       | 50h Clear status register | 70h Read status          | 80h Program /Erase suspend | D0h BE confirm, P/E resume, ULB confirm | 20h Erase setup | 10h/40h Program setup | FFh Read array | SR7                                    | Data when read | State | Mode |                                      |  |
| Erase suspend read array                                                         | Lock/RCR setup        | Erase suspend read query | Erase suspend read ID | Erase suspend read array                    | Erase suspend read status | ES read array            | Erase busy                | Erase suspend read array |                            | 1                                       | Array           | Read array            | Erase suspend  | 75                                     | Setup          |       |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | Erase suspend read array |                           |                          | 76                         |                                         |                 |                       |                | Busy                                   |                |       |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | ES read array            | Erase busy                | ES read array            | Prog. setup                |                                         |                 |                       |                | ES read array                          | 77             | Idle  |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | ES read array            |                           |                          | 78                         |                                         |                 |                       |                | Prog. suspend                          |                |       |      |                                      |  |
| Erase suspend read array                                                         | Lock/RCR setup        | Erase suspend read query | Erase suspend read ID | Erase suspend read array                    | Erase suspend read status | ES read array            | Erase busy                | Erase suspend read array |                            | 1                                       | ID              | Read ID               |                | 79                                     | Setup          |       |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | Erase suspend read array |                           |                          | 80                         |                                         |                 |                       |                | Busy                                   |                |       |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | ES read array            | Erase busy                | ES read array            | Prog. setup                |                                         |                 |                       |                | ES read array                          | 81             | Idle  |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | ES read array            |                           |                          | 82                         |                                         |                 |                       |                | Prog. suspend                          |                |       |      |                                      |  |
| Erase suspend read array                                                         | Lock/RCR setup        | Erase suspend read query | Erase suspend read ID | Erase suspend read array                    | Erase suspend read status | ES read array            | Erase busy                | Erase suspend read array |                            | 1                                       | CFI             | Read query            |                | 83                                     | Setup          |       |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | Erase suspend read array |                           |                          | 84                         |                                         |                 |                       |                | Busy                                   |                |       |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | ES read array            | Erase busy                | ES read array            | Prog. setup                |                                         |                 |                       |                | ES read array                          | 85             | Idle  |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | ES read array            |                           |                          | 86                         |                                         |                 |                       |                | Prog. suspend                          |                |       |      |                                      |  |
| Erase suspend read array                                                         | Lock/RCR setup        | Erase suspend read query | Erase suspend read ID | Erase suspend read array                    | Erase suspend read status | ES read array            | Erase busy                | Erase suspend read array |                            | 1                                       | Status          | Error                 | 87             | Setup                                  |                |       |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | Erase suspend read array |                           |                          | 88                         |                                         |                 |                       | Busy           |                                        |                |       |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | ES read array            | Erase busy                | ES read array            | Prog. setup                |                                         |                 |                       | ES read array  | 89                                     | Idle           |       |      |                                      |  |
|                                                                                  |                       |                          |                       |                                             |                           | ES read array            |                           |                          | 90                         |                                         |                 |                       | Prog. suspend  |                                        |                |       |      |                                      |  |



**Table 7**  
**Bus Operations**

| MODE                                                                     | RST#            | CE#             | ADV#            | OE#             | WE#             | ADDRESS | DQ0-DQ15 |
|--------------------------------------------------------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------|----------|
| Read (array, status registers, device identification register, or query) | V <sub>IH</sub> | V <sub>IL</sub> | V <sub>IL</sub> | V <sub>IL</sub> | V <sub>IH</sub> | X       | Dout     |
| Standby                                                                  | V <sub>IH</sub> | V <sub>IH</sub> | X               | X               | X               | X       | High-Z   |
| Output Disable                                                           | V <sub>IH</sub> | V <sub>IL</sub> | X               | V <sub>IH</sub> | V <sub>IH</sub> | X       | High-Z   |
| Reset                                                                    | V <sub>IL</sub> | X               | X               | X               | X               | X       | High-Z   |
| Write                                                                    | V <sub>IH</sub> | V <sub>IL</sub> | V <sub>IL</sub> | V <sub>IH</sub> | V <sub>IL</sub> | X       | Din      |

**Table 8**  
**Status Register Bit Definitions**

| WSMS | ESS | ES | PS | VPPS | PSS | BLS | R |
|------|-----|----|----|------|-----|-----|---|
| 7    | 6   | 5  | 4  | 3    | 2   | 1   | 0 |

| STATUS BIT # | STATUS REGISTER BIT                                                                                                              | DESCRIPTION                                                                                                                                                                                                                                                                                                                       |
|--------------|----------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SR7          | WRITE STATE MACHINE STATUS (WSMS)<br>1 = Ready<br>0 = Busy                                                                       | Check WSM bit to determine word program or block erase completion before checking program or erase status bits.                                                                                                                                                                                                                   |
| SR6          | ERASE SUSPEND STATUS (ESS)<br>1 = BLOCK ERASE Suspended<br>0 = BLOCK ERASE in Progress/Completed                                 | When ERASE SUSPEND is issued, WSM halts execution and sets both WSMS and ESS bits to "1." ESS bit remains set to "1" until an ERASE RESUME command is issued.                                                                                                                                                                     |
| SR5          | ERASE/CHECK BLOCK ERASE STATUS (ES)<br>1 = Error in BLOCK ERASE/<br>CHECK BLOCK ERASE<br>0 = Successful BLOCK ERASE              | When this bit is set to "1" and ERASE CONFIRM is issued, WSM has applied the maximum number of erase pulses to the block and is still unable to verify successful block erasure. When this bit is set to "1" and CHECK BLOCK ERASE CONFIRM is issued, WSM has checked the block for its erase state, and the block is not erased. |
| SR4          | PROGRAM STATUS (PS)<br>1 = Error in PROGRAM<br>0 = Successful PROGRAM                                                            | When this bit is set to "1," WSM has attempted but failed to program a word.                                                                                                                                                                                                                                                      |
| SR3          | VPP STATUS (VPPS)<br>1 = VPP Low Detect, Operation Abort<br>0 = VPP = OK                                                         | The VPP status bit does not provide continuous indication of the VPP level. The WSM interrogates the VPP level only after the program or erase command sequences have been entered and informs the system if VPP < 0.9V. The VPP level is also checked before the PROGRAM/ERASE is verified by the WSM.                           |
| SR2          | PROGRAM SUSPEND STATUS (PSS)<br>1 = PROGRAM Suspended<br>0 = PROGRAM in Progress/Completed                                       | When PROGRAM SUSPEND is issued, WSM halts execution and sets both WSM and PSS bits to "1." PSS bit remains set to "1" until a PROGRAM RESUME command is issued.                                                                                                                                                                   |
| SR1          | BLOCK LOCK STATUS (BLS)<br>1 = PROGRAM/ERASE Attempted on a Locked Block; Operation Aborted<br>0 = No Operation to Locked Blocks | If a PROGRAM or ERASE operation is attempted to one of the locked blocks, this is set by the WSM. The operation specified is aborted and the device is returned to read status mode.                                                                                                                                              |
| SR0          | RESERVED FOR FUTURE ENHANCEMENT                                                                                                  | This bit is reserved for future use.                                                                                                                                                                                                                                                                                              |



## PROGRAMMING OPERATIONS

There are two CSM commands for programming: PROGRAM SETUP and ACCELERATED PROGRAMMING ALGORITHM (see Table 3).

### PROGRAM SETUP COMMAND

After the 40h command code is entered on DQ0-DQ7, the WSM takes over and correctly sequences the device to complete the PROGRAM operation. The WRITE operation may be monitored through the status register (see the Status Register section). During this time, the CSM will only respond to a PROGRAM SUSPEND command until the PROGRAM operation has been completed, after which time, all commands to the CSM become valid again. The PROGRAM operation can be suspended by issuing a PROGRAM SUSPEND command (B0h). Once the WSM reaches the suspend state, it allows the CSM to respond only to READ ARRAY, READ STATUS REGISTER, READ PROTECTION CONFIGURATION, READ QUERY, PROGRAM SETUP, or PROGRAM RESUME. During the PROGRAM SUSPEND operation, array data should be read from an address other than the one being programmed. To resume the PROGRAM operation, a PROGRAM RESUME command (D0h) must be issued to cause the CSM to clear the suspend state previously set (see Figure 4 for programming operation and Figure 5 for program suspend and program resume).

Taking RP# to V<sub>IL</sub> during programming aborts the PROGRAM operation.

### ACCELERATED PROGRAMMING ALGORITHM COMMAND

The accelerated programming algorithm (APA) is intended for in-system and in-factory use. Its 32 single-word internal buffer enables fast data stream programming.

The APA is activated when a 10h command is written. Upon activation, the word address and the data sequences must be provided to the WSM, without polling SR7. The same starting address must be provided for each data word. After all 32 sequences are issued, the status register reports a busy condition. Figure 6 shows the APA flowchart.

If the data stream is shorter than 32 words, use FFFFh to complete the data stream. Also, ensure the starting address is aligned with a 32-word boundary.

The APA is fully concurrent. For example, it can be interrupted and resumed during programming. When the APA is active, only a read access in the other bank is allowed.

For in-factory programming, the APA, along with an optimized set of programming parameters, minimizes chip programming time when  $11.4V \leq V_{PP} \leq 12.6V$ .

For in-system programming, when  $0.9V \leq V_{PP} \leq 2.2V$ , the APA and the 32 single-word buffer significantly improve both the system throughput and the average programming time when compared with standard programming practices. The accelerated programming functionality executes and verifies the APA without microprocessor intervention. This relieves the microprocessor from constantly monitoring the progress of the programming and erase activity, freeing up valuable memory bus bandwidth. This increases the system throughput.

## ERASE OPERATIONS

An ERASE operation must be used to initialize all bits in an array block to "1s." After BLOCK ERASE CONFIRM is issued, the CSM responds only to an ERASE SUSPEND command until the WSM completes its task.

Block erasure inside the memory array sets all bits within the address block to logic 1s. Erase is accomplished only by blocks; data at single address locations within the array cannot be erased individually. The block to be erased is selected by using any valid address within that block. Block erasure is initiated by a command sequence to the CSM: BLOCK ERASE SETUP (20h) followed by BLOCK ERASE CONFIRM (D0h) (see Figure 7). A two-command erase sequence protects against accidental erasure of memory contents.

When the BLOCK ERASE CONFIRM command is complete, the WSM automatically executes a sequence of events to complete the block erasure. During this sequence, the block is programmed with logic 0s, data is verified, all bits in the block are erased, and finally verification is performed to ensure that all bits are correctly erased. Monitoring of the ERASE operation is possible through the status register (see the Status Register section).

During the execution of an ERASE operation, the ERASE SUSPEND command (B0h) can be entered to direct the WSM to suspend the ERASE operation. Once the WSM has reached the suspend state, it allows the CSM to respond only to the READ ARRAY, READ STATUS REGISTER, READ QUERY, READ CHIP PROTECTION CONFIGURATION, PROGRAM SETUP, PROGRAM RESUME, ERASE RESUME, and LOCK SETUP (see the Block Locking section). During the ERASE SUSPEND operation, array data must be read from a block other than the one being erased. To resume the ERASE

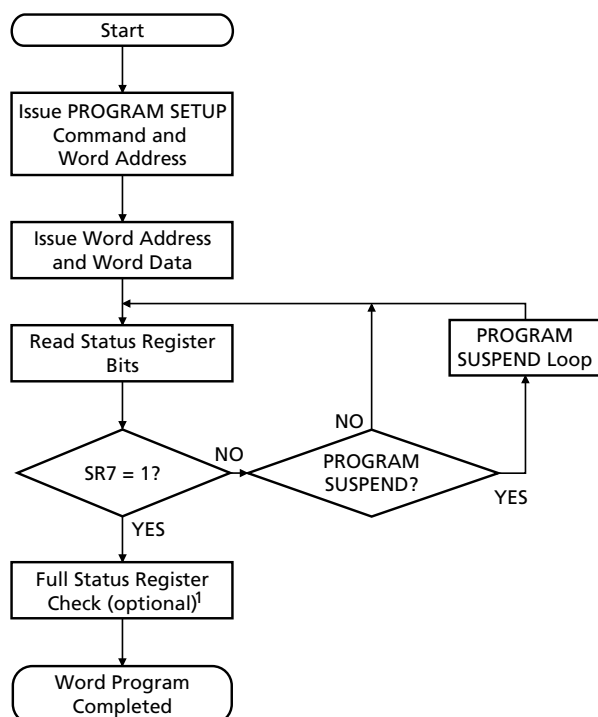


operation, an ERASE RESUME command (D0h) must be issued to cause the CSM to clear the suspend state previously set (see Figure 8). It is also possible that an ERASE in any bank can be suspended and a WRITE to another block in the same bank can be initiated. After the completion of a WRITE, an ERASE can be resumed by writing an ERASE RESUME command.

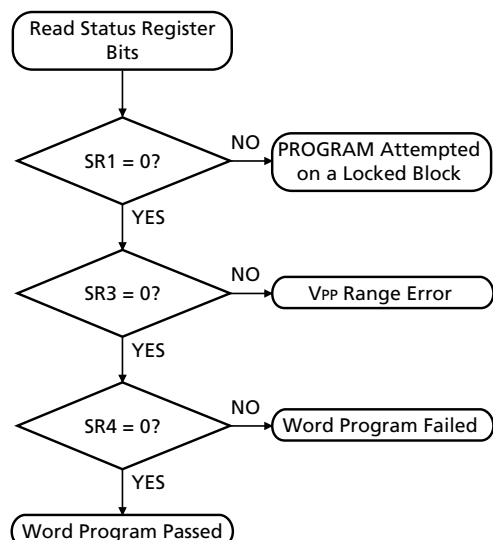
After an ERASE command completion, it is possible to check if the block has been erased successfully, using the CHECK BLOCK ERASE command. Two bus cycles are required for this operation: one to set up the CHECK BLOCK ERASE and the second one to start the execution of the command. If, after the operation, the SR5 bit is set to "0," the operation has been completed successfully. If it is set to "1," there has been an error during the BLOCK ERASE operation.



**Figure 4**  
**Automated Word Programming**  
**Flowchart**



**FULL STATUS REGISTER CHECK FLOW**



| BUS OPERATION | COMMAND             | COMMENTS                                                                |
|---------------|---------------------|-------------------------------------------------------------------------|
| WRITE         | WRITE PROGRAM SETUP | Data = 40h or 10h<br>Addr = Address of word to be programmed            |
| WRITE         | WRITE DATA          | Data = Word to be programmed<br>Addr = Address of word to be programmed |
| READ          |                     | Status register data<br>Toggle OE# or CE# to update status register     |
| Standby       |                     | Check SR7<br>1 = Ready, 0 = Busy                                        |

Repeat for subsequent words.

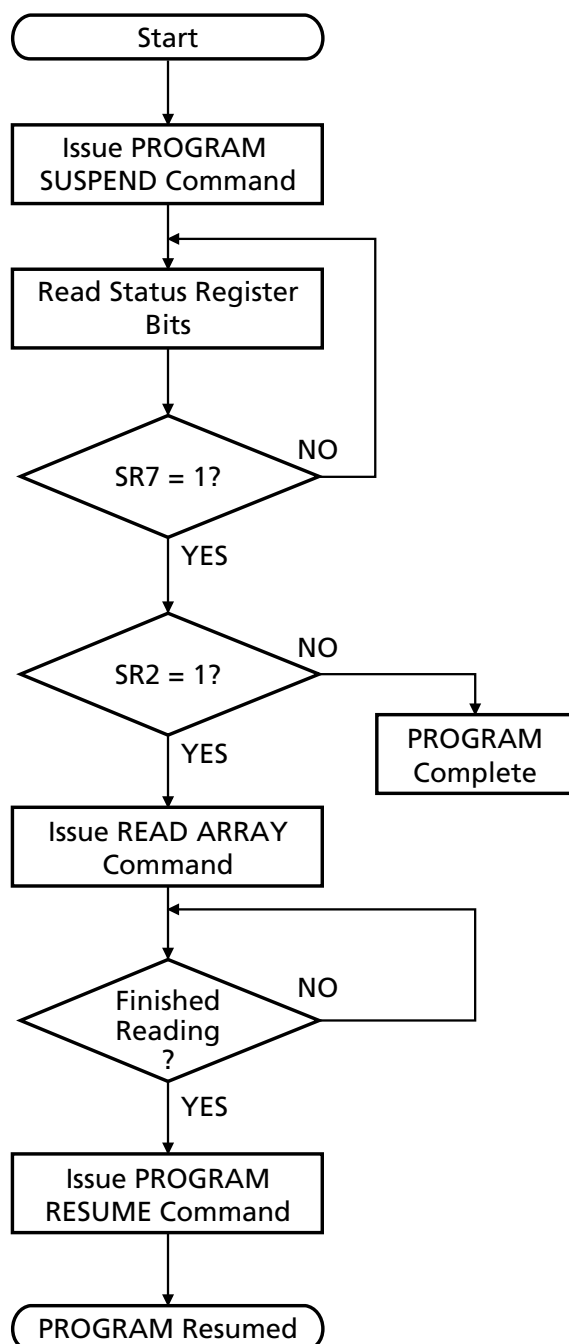
Write FFh after the last word programming operation to reset the device to read array mode.

| BUS OPERATION | COMMAND | COMMENTS                                                 |
|---------------|---------|----------------------------------------------------------|
| Standby       |         | Check SR1<br>1 = Detect locked block                     |
| Standby       |         | Check SR3 <sup>2</sup><br>1 = Detect V <sub>PP</sub> LOW |
| Standby       |         | Check SR4 <sup>3</sup><br>1 = Word program error         |

- NOTE:**
1. Full status register check can be done after each word or after a sequence of words.
  2. SR3 must be cleared before attempting additional PROGRAM/ERASE operations.
  3. SR4 is cleared only by the CLEAR STATUS REGISTER command, but it does not prevent additional program operation attempts.



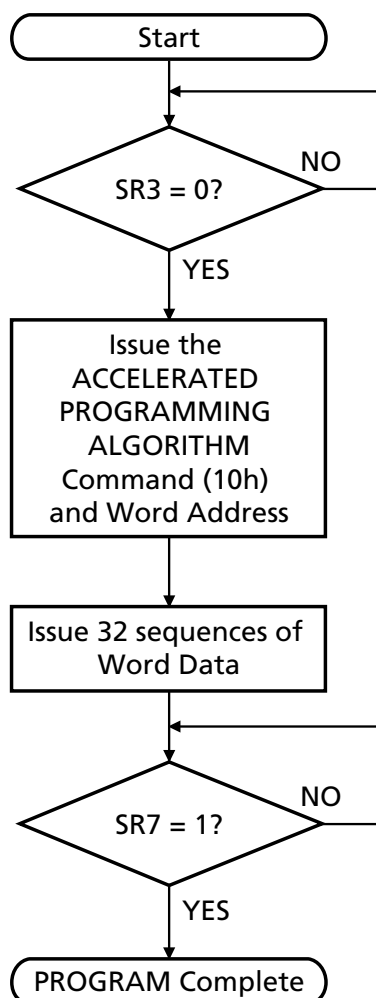
**Figure 5**  
**PROGRAM SUSPEND/  
PROGRAM RESUME Flowchart**



| BUS OPERATION | COMMAND         | COMMENTS                                                            |
|---------------|-----------------|---------------------------------------------------------------------|
| WRITE         | PROGRAM SUSPEND | Data = B0h                                                          |
| READ          |                 | Status register data<br>Toggle OE# or CE# to update status register |
| Standby       |                 | Check SR7<br>1 = Ready                                              |
| Standby       |                 | Check SR2<br>1 = Suspended                                          |
| WRITE         | READ ARRAY      | Data = FFh                                                          |
| READ          |                 | Read data from block other than that being programmed               |
| WRITE         | PROGRAM RESUME  | Data = D0h                                                          |



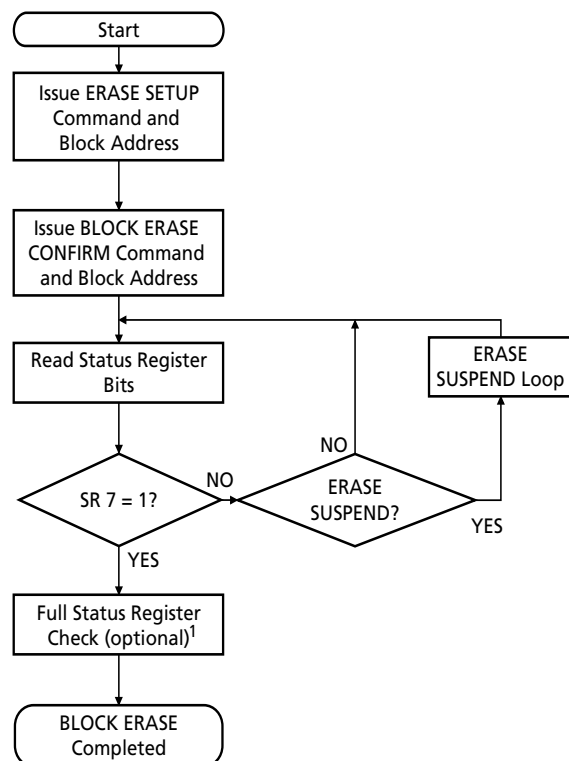
**Figure 6**  
**Accelerated Program**  
**Algorithm Flowchart**



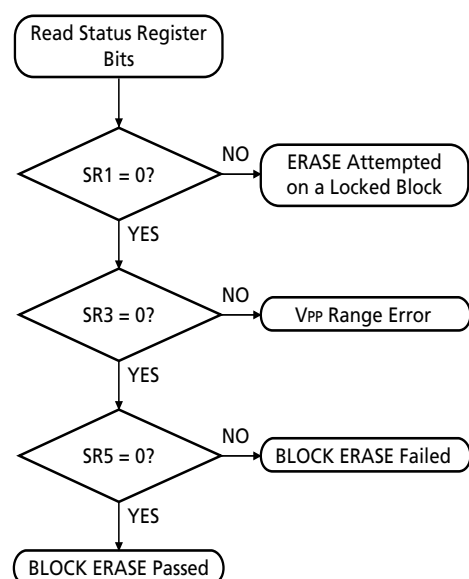
| BUS OPERATION | COMMAND                                   | COMMENTS                                                            |
|---------------|-------------------------------------------|---------------------------------------------------------------------|
| WRITE         | WRITE ACCELERATED PROGRAM ALGORITHM SETUP | Data = 10h<br>Addr = Start address                                  |
| WRITE         | WRITE DATA                                | Data = Word to be programmed<br>Addr = Start address                |
| READ          |                                           | Status register data<br>Toggle OE# or CE# to update status register |
| Standby       |                                           | Check SR7<br>1 = Ready, 0 = Busy                                    |



**Figure 7  
BLOCK ERASE Flowchart**



**FULL STATUS REGISTER CHECK FLOW**



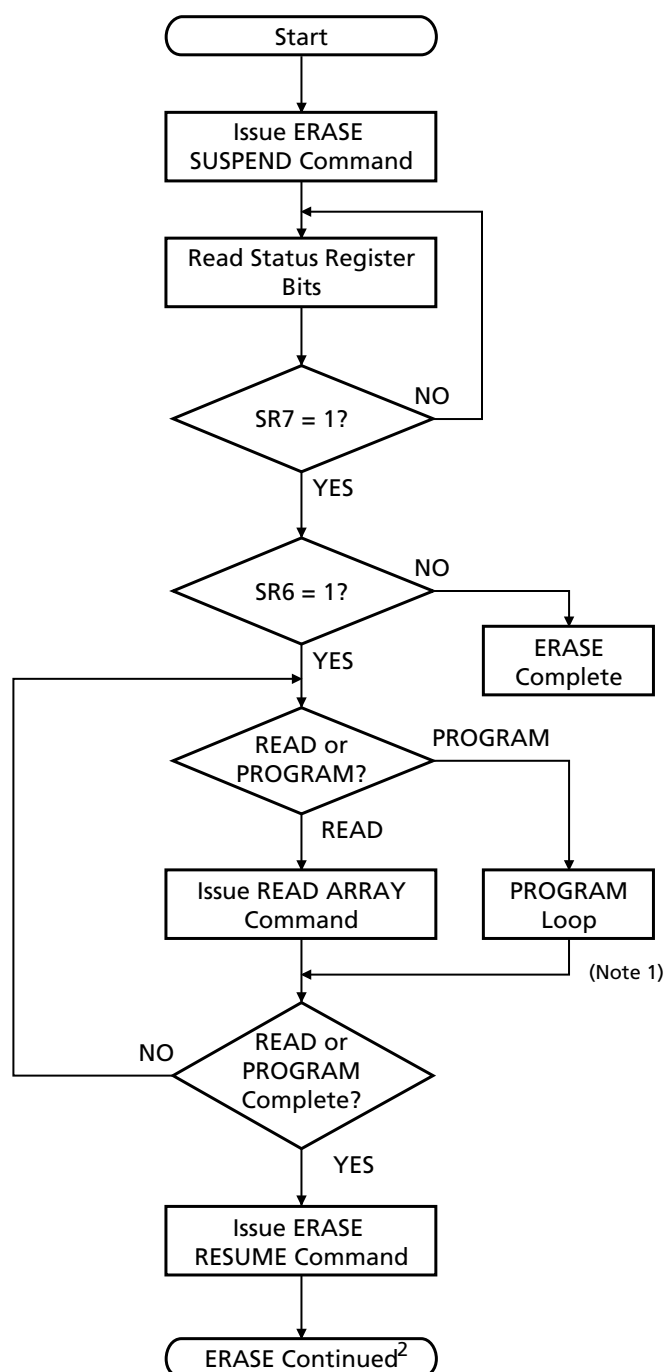
| BUS OPERATION                                                                                                           | COMMAND           | COMMENTS                                                            |
|-------------------------------------------------------------------------------------------------------------------------|-------------------|---------------------------------------------------------------------|
| WRITE                                                                                                                   | WRITE ERASE SETUP | Data = 20h<br>Block Addr = Address within block to be erased        |
| WRITE                                                                                                                   | ERASE             | Data = D0h<br>Block Addr = Address within block to be erased        |
| READ                                                                                                                    |                   | Status register data<br>Toggle OE# or CE# to update status register |
| Standby                                                                                                                 |                   | Check SR7<br>1 = Ready, 0 = Busy                                    |
| Repeat for subsequent blocks.<br>Write FFh after the last BLOCK ERASE operation to reset the device to read array mode. |                   |                                                                     |

| BUS OPERATION | COMMAND | COMMENTS                                                   |
|---------------|---------|------------------------------------------------------------|
| Standby       |         | Check SR1<br>1 = Detect locked block                       |
| Standby       |         | Check SR3 <sup>2</sup><br>1 = Detect V <sub>PP</sub> block |
| Standby       |         | Check SR5 <sup>3</sup><br>1 = BLOCK ERASE error            |

- NOTE:**
1. Full status register check can be done after each block or after a sequence of blocks.
  2. SR3 must be cleared before attempting additional PROGRAM/ERASE operations.
  3. SR5 is cleared only by the CLEAR STATUS REGISTER command in cases where multiple blocks are erased before full status is checked.



**Figure 8**  
**ERASE SUSPEND/ERASE RESUME**  
**Flowchart**

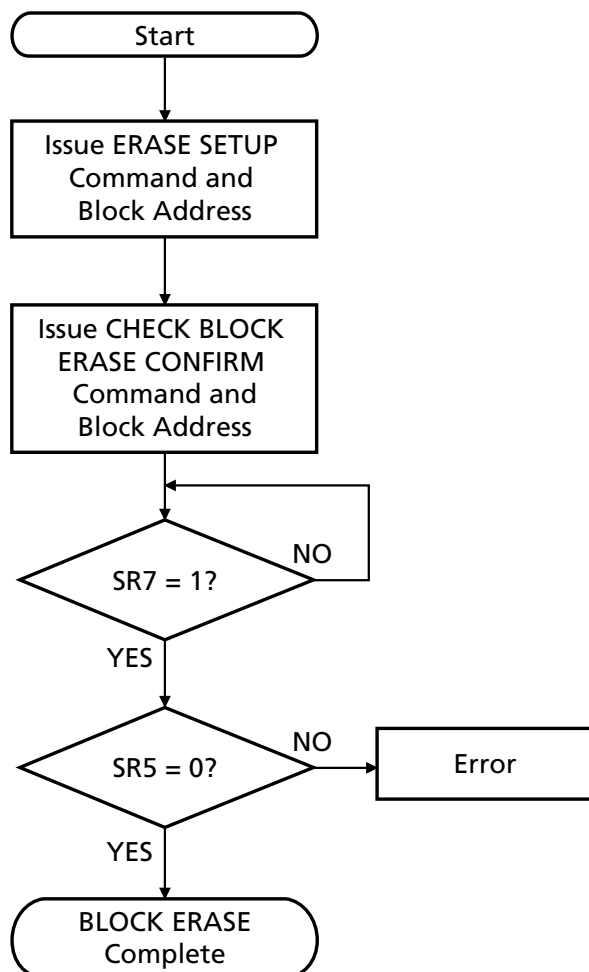


| BUS OPERATION | COMMAND       | COMMENTS                                                            |
|---------------|---------------|---------------------------------------------------------------------|
| WRITE         | ERASE SUSPEND | Data = B0h                                                          |
| READ          |               | Status register data<br>Toggle OE# or CE# to update status register |
| Standby       |               | Check SR7<br>1 = Ready                                              |
| Standby       |               | Check SR6<br>1 = Suspended                                          |
| WRITE         | READ ARRAY    | Data = FFh                                                          |
| READ          |               | Read data from block other than that being erased.                  |
| WRITE         | ERASE RESUME  | Data = D0h                                                          |

**NOTE:** 1. See Word Programming Flowchart for complete programming procedure.  
2. See BLOCK ERASE Flowchart for complete erasure procedure.



**Figure 9**  
**CHECK BLOCK ERASE Flowchart**



| BUS OPERATION | COMMAND                   | COMMENTS                                                            |
|---------------|---------------------------|---------------------------------------------------------------------|
| WRITE         | ERASE SETUP               | Data = 20h<br>Block Addr = Address within block to be checked       |
| WRITE         | CHECK BLOCK ERASE CONFIRM | Data = D1h<br>Block Addr = Address within block to be checked       |
| READ          |                           | Status register data<br>Toggle OE# or CE# to update status register |
| Standby       |                           | Check SR7 and SR5                                                   |



## READ-WHILE-WRITE/ERASE CONCURRENCY

It is possible for the device to read from one bank while erasing/writing to another bank. Once a bank enters the WRITE/ERASE operation, the other bank automatically enters read array mode. For example, during a READ CONCURRENCY operation, if a PROGRAM/ERASE command is issued in bank *a*, then bank *a* changes to the read status mode and bank *b* defaults to the read array mode. The device will read from bank *b* if the latched address resides in bank *b* (see Figure 10). Similarly, if a PROGRAM/ERASE command is issued in bank *b*, then bank *b* changes to read status mode and bank *a* defaults to read array mode. When returning to bank *a*, the device will read PROGRAM/ERASE status if the latched address resides in bank *a*.

A correct bank address must be specified to read the status register after returning from a concurrent read in the other bank.

When reading the CFI or the chip protection register, concurrent operation is not allowed on the top boot device. Concurrent READ of the CFI or the chip protection register is only allowed when a PROGRAM or ERASE operation is performed on bank *b* on the bottom boot device. For a bottom boot device, reading of the CFI table or the chip protection register is only allowed if bank *b* is in read array mode. For a top boot device, reading of the CFI table or the chip protection register is only allowed if bank *a* is in read array mode.

## READ CONFIGURATION REGISTER (RCR) MODE

The SET READ CONFIGURATION REGISTER command is a WRITE operation to the read configuration register (RCR). It is a two-cycle command sequence. Read configuration setup is written, followed by a second write that specifies the data to be written to the read configuration register. The data is placed on the address bus A0–A15, and it is latched on the rising edge of ADV#, CE#, or WE#, whichever occurs first. The read configuration provides the read mode (burst, synchronous, or asynchronous), burst order, latency counter, and burst length. After executing this command, the device returns to read array mode.

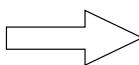
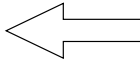
## READ CONFIGURATION

The device supports three read configurations: asynchronous, synchronous burst mode, and page mode. The bit RCR15 (see Table 9) in the read configuration register sets the read configuration. Asynchronous random mode is the default read mode.

At power-up, the RCR is set to BFCFh.

Status registers and the device identification register support asynchronous and single synchronous READ operations only.

**Figure 10**  
**READ-While-WRITE Concurrency**

| Bank a                                                                                                                                                                                                                                          | Bank b                                                                                                                                                                                                                                          |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 - Erasing/writing to bank a<br>2 - Erasing in bank a can be suspended, and a WRITE to another block in bank a can be initiated.<br>3 - After the WRITE in that block is complete, an ERASE can be resumed by writing an ERASE RESUME command. | <br>1 - Reading from bank b                                                                                                                                  |
| 1 - Reading bank a<br>                                                                                                                                      | 1 - Erasing/writing to bank b<br>2 - Erasing in bank b can be suspended, and a WRITE to another block in bank b can be initiated.<br>3 - After the WRITE in that block is complete, an ERASE can be resumed by writing an ERASE RESUME command. |



**Table 9**  
**Read Configuration Register**

| RM | R  | LC2 | LC1 | LC0 | WSP | HDO | WC |
|----|----|-----|-----|-----|-----|-----|----|
| 15 | 14 | 13  | 12  | 11  | 10  | 9   | 8  |

| BS | CC | R | DPD | BW | BL2 | BL1 | BL0 |
|----|----|---|-----|----|-----|-----|-----|
| 7  | 6  | 5 | 4   | 3  | 2   | 1   | 0   |

| BIT # | DESCRIPTION                | FUNCTION                                                                                                                                                                                                                                                |
|-------|----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15    | Read Mode (RM)             | 0 = Synchronous Burst Access Mode<br>1 = Asynchronous/Page Access Mode (default)                                                                                                                                                                        |
| 14    | Reserved                   | Default = 0                                                                                                                                                                                                                                             |
| 13–11 | Latency Counter (LC)       | Sets the number of clock cycles before valid data out:<br>000 = Code 0 - reserved<br>001 = Code 1 - reserved<br>010 = Code 2 - reserved<br>011 = Code 3<br>100 = Code 4<br>101 = Code 5<br>110 = Code 6 - reserved<br>111 = Code 7 - reserved (default) |
| 10    | Wait Signal Polarity (WSP) | 0 = WAIT signal is active LOW<br>1 = WAIT signal is active HIGH (default)                                                                                                                                                                               |
| 9     | Hold Data Out (HDO)        | Sets the data output configuration:<br>0 = Hold data for one clock<br>1 = Hold data for two clocks (default)                                                                                                                                            |
| 8     | Wait Configuration (WC)    | Controls the behavior of the WAIT# output signal:<br>0 = WAIT# asserted during delay<br>1 = WAIT# asserted one data cycle before delay (default)                                                                                                        |
| 7     | Burst Sequence (BS)        | Specifies the order in which data is addressed in synchronous burst mode:<br>0 = Interleaved<br>1 = Linear (default)                                                                                                                                    |
| 6     | Clock Configuration (CC)   | Defines the clock edge on which the BURST operation starts and data is referenced:<br>0 = Falling edge<br>1 = Rising edge (default)                                                                                                                     |
| 5     | Reserved                   | Default = 0                                                                                                                                                                                                                                             |
| 4     | Deep Power-Down (DPD)      | 0 = No DPD mode (default)<br>1 = DPD mode                                                                                                                                                                                                               |
| 3     | Burst Wrap (BW)            | 0 = Burst wraps within the burst length<br>1 = No burst wrap (default)                                                                                                                                                                                  |
| 2–0   | Burst Length (BL)          | Sets the number of words the device will output in burst mode:<br>001 = 4 words<br>010 = 8 words<br>111 = Continuous burst (default)                                                                                                                    |

### LATENCY COUNTER

The latency counter provides the number of clocks that must elapse after ADV# is set active before data will be available. This value depends on the input clock frequency. See Table 10 for the specific input clock frequency configuration code. Also, see Figure 11 for the timing diagrams pertinent to codes 2, 3, and 4.

Table 10 illustrates the data output latency from ADV#, active asynchronous access, and system strobe for different latency counter codes.

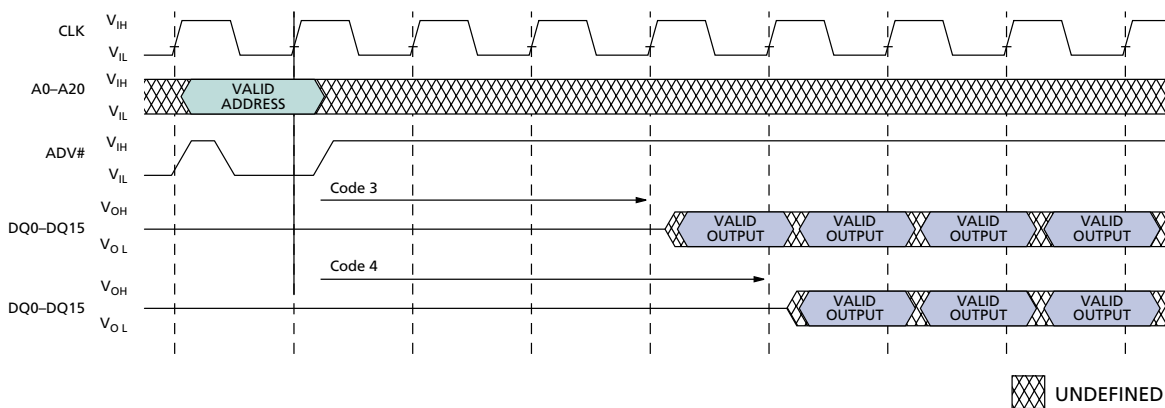
### HOLD DATA OUTPUT CONFIGURATION

The hold data output configuration specifies for how many clocks data will be held valid. (See Figure 12.)

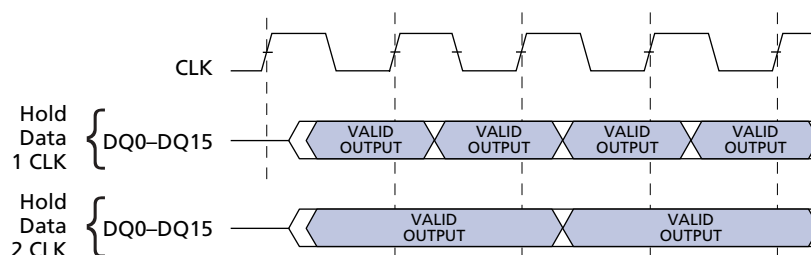
### WAIT# CONFIGURATION

The wait configuration bit, RCR8, sets the behavior of the WAIT# output signal. The WAIT# signal can be active during output delay or one data cycle before delay, when continuous burst length is enabled. WAIT# = 1 indicates valid data when RCR10 = 0. WAIT# = 0 indicates invalid data when RCR10 = 0. The setting of WAIT# before or during the delay (RCR8) will depend on the system and CPU characteristic. If RCR3 = 1 (no wrap mode), then WAIT# can also be enabled in a four- or eight-word burst if the no-wrap burst crosses the first eight-word boundary.

**Figure 11  
Latency Counter**



**Figure 12  
Hold Data Output Configuration**





### BURST SEQUENCE

The burst sequence specifies the address order of the data in synchronous burst mode. It can be programmed as either linear or interleaved burst order. Continuous burst length only supports linear burst order. See Table 11 for more details.

### CLOCK CONFIGURATION

The clock configuration configures the starting burst cycle, output data, and WAIT# signal to be asserted on the rising or falling edge of the clock.

**Table 10**  
**Clock Frequency vs. First Access Latency<sup>1</sup>**

| FREQUENCY (MHz) | PERIOD (ns) | LATENCY COUNTER CONFIGURATION | CLK CYCLES FOR FIRST DATA | SYSTEM STROBE (ns) |
|-----------------|-------------|-------------------------------|---------------------------|--------------------|
| 40              | 25          | 3                             | 4                         | 100                |
| 54              | 18.5        | 4                             | 5                         | 92.5               |

**NOTE:** 1. Data is valid only if <sup>t</sup>AKS is applied.

**Table 11**  
**Sequence and Burst Length**

| STARTING ADDRESS (DEC) | WRAP | NO WRAP | 4-WORD BURST LENGTH |             | 8-WORD BURST LENGTH  |                 | CONTINUOUS BURST         |
|------------------------|------|---------|---------------------|-------------|----------------------|-----------------|--------------------------|
|                        | RCR3 | RCR3    | LINEAR              | INTERLEAVED | LINEAR               | INTERLEAVED     | LINEAR                   |
| 0                      | 0    |         | 0-1-2-3             | 0-1-2-3     | 0-1-2-3-4-5-6-7      | 0-1-2-3-4-5-6-7 | 0-1-2-3-4-5-6-...        |
| 1                      | 0    |         | 1-2-3-0             | 1-0-3-2     | 1-2-3-4-5-6-7-0      | 1-0-3-2-5-4-7-6 | 1-2-3-4-5-6-7-...        |
| 2                      | 0    |         | 2-3-0-1             | 2-3-0-1     | 2-3-4-5-6-7-0-1      | 2-3-0-1-6-7-4-5 | 2-3-4-5-6-7-8-...        |
| 3                      | 0    |         | 3-0-1-2             | 3-2-1-0     | 3-4-5-6-7-0-1-2      | 3-2-1-0-7-6-5-4 | 3-4-5-6-7-8-9-...        |
| 4                      | 0    |         |                     |             | 4-5-6-7-0-1-2-3      | 4-5-6-7-0-1-2-3 | 4-5-6-7-8-9-10-...       |
| 5                      | 0    |         |                     |             | 5-6-7-0-1-2-3-4      | 5-4-7-6-1-0-3-2 | 5-6-7-8-9-10-11-...      |
| 6                      | 0    |         |                     |             | 6-7-0-1-2-3-4-5      | 6-7-4-5-2-3-0-1 | 6-7-8-9-10-11-12-...     |
| 7                      | 0    |         |                     |             | 7-0-1-2-3-4-5-6      | 7-6-5-4-3-2-1-0 | 6-7-8-9-10-11-12-13-...  |
| ...                    | ...  | ...     | ...                 | ...         | ...                  | ...             | ...                      |
| 14                     | 0    |         |                     |             |                      |                 | 14-15-16-17-18-19-20-..  |
| 15                     | 0    |         |                     |             |                      |                 | 15-16-17-18-19-20-21-..  |
| ...                    | ...  | ...     | ...                 | ...         | ...                  | ...             | ...                      |
| 0                      |      | 1       | 0-1-2-3             | NA          | 0-1-2-3-4-5-6-7      | NA              | 0-1-2-3-4-5-6-...        |
| 1                      |      | 1       | 1-2-3-4             | NA          | 1-2-3-4-5-6-7-8      | NA              | 1-2-3-4-5-6-7-...        |
| 2                      |      | 1       | 2-3-4-5             | NA          | 2-3-4-5-6-7-8-9      | NA              | 2-3-4-5-6-7-8-...        |
| 3                      |      | 1       | 3-4-5-6             | NA          | 3-4-5-6-7-8-9-10     | NA              | 3-4-5-6-7-8-9-...        |
| 4                      |      | 1       |                     |             | 4-5-6-7-8-9-10-11    | NA              | 4-5-6-7-8-9-10-...       |
| 5                      |      | 1       |                     |             | 5-6-7-8-9-10-11-12   | NA              | 5-6-7-8-9-10-11-...      |
| 6                      |      | 1       |                     |             | 6-7-8-9-10-11-12-13  | NA              | 6-7-8-9-10-11-12-...     |
| 7                      |      | 1       | ...                 |             | 7-8-9-10-11-12-13-14 | NA              | 7-8-9-10-11-12-13-...    |
| ...                    | ...  | ...     | ...                 | ...         | ...                  | ...             | ...                      |
| 14                     |      | 1       | ...                 |             |                      |                 | 14-15-16-17-18-19-20-... |
| 15                     |      | 1       |                     |             |                      |                 | 15-16-17-18-19-20-21-... |



### BURST WRAP

The burst wrap option, RCR3, signals if a four- or an eight-word linear burst access wraps within the burst length or whether it crosses the eight-word boundary. In wrap mode (RCR3 = 0) the four- or eight-word access will wrap within the four or eight words, respectively. In no-wrap mode (RCR3 = 1), the device operates similarly to a continuous burst. For example, in a four-word burst, no-wrap mode, the possible linear burst sequences that do not assert WAIT# are:

|         |             |
|---------|-------------|
| 0-1-2-3 | 8-9-10-11   |
| 1-2-3-4 | 9-10-11-12  |
| 2-3-4-5 | 10-11-12-13 |
| 3-4-5-6 | 11-12-13-14 |
| 4-5-6-7 | 12-13-14-15 |

The worst-case delay is seen at the end of the eight-word boundary: 7-8-9-10 and 15-16-17-18. In a four-word burst, wrap mode, no WAIT# is asserted and the possible wrap sequences are:

|         |                |
|---------|----------------|
| 0-1-2-3 | 5-6-7-4        |
| 1-2-3-0 | 6-7-4-5        |
| 2-3-0-1 | 7-4-5-6        |
| 3-0-1-2 | 8-9-10-11      |
| 4-5-6-7 | 9-10-11-8 etc. |

Refer to Table 11 for a list of acceptable sequences. When the continuous burst option is selected, the internal address wraps to 000000h if the device is read past the last address.

### BURST LENGTH

The burst length defines the number of words the device outputs. The device supports a burst length of four or eight words. The device can also be set in continuous burst mode. In this mode the device linearly outputs data until the internal burst counter reaches the end of the burstable address space. RCR2 sets the burst length.

### CONTINUOUS BURST LENGTH

During continuous burst mode operation, the Flash memory may have an output delay when the burst sequence crosses the first eight-word boundary. Also, in four- or eight-word bursts with the burst wrap set to no wrap (RCR3 = 1), the Flash memory may have an output delay when the burst sequence crosses the first eight-word boundary. The starting address dictates whether or not a delay occurs. If the starting address is aligned with an eight-word boundary, the delay is not seen. For a four-word burst, if the starting address is aligned with a four-word boundary, a delay is not seen. If the starting address is at the end of an eight-word boundary, the output delay is the maximum delay, equal to the latency counter setting.

The delay happens only once during a continuous burst access. If the burst never crosses an eight-word boundary, the WAIT# is not asserted. The activation of WAIT# informs the system if this output delay occurs.

**Table 12**  
**Block Locking State Transition**

| WP# | DQ1 | DQ0 | NAME               | ERASE/PROG ALLOWED | LOCK      | UNLOCK    | LOCK DOWN |
|-----|-----|-----|--------------------|--------------------|-----------|-----------|-----------|
| 0   | 0   | 0   | Unlocked           | Yes                | To [001]  | No Change | To [011]  |
| 0   | 0   | 1   | Locked (Default)   | No                 | No Change | To [000]  | To [011]  |
| 0   | 1   | 1   | Lock Down          | No                 | No Change | No Change | No Change |
| 1   | 0   | 0   | Unlocked           | Yes                | To [101]  | No Change | To [111]  |
| 1   | 0   | 1   | Locked             | No                 | No Change | To [100]  | To [111]  |
| 1   | 1   | 0   | Lock Down Disabled | Yes                | To [111]  | No Change | To [111]  |
| 1   | 1   | 1   | Lock Down Disabled | No                 | No Change | To [110]  | No Change |

## WAIT# SIGNAL IN BURST MODE

In the continuous burst mode or in the four- or eight-word burst mode with no wrap ( $RCR3 = 1$ ), the output WAIT# informs the system when data is valid. When WAIT# is asserted during delay ( $RCR8 = 0$ ), WAIT# = 1 indicates valid data and WAIT# = 0 indicates invalid data. If  $RCR8 = 0$ , WAIT# is asserted on the same cycle on which the delay occurs. If  $RCR8 = 1$ , WAIT# is asserted one cycle before the delay occurs.

## BLOCK LOCKING

The Flash devices provide a flexible locking scheme that allows each block to be individually locked or unlocked with no latency.

The devices offer two-level protection for the blocks. The first level allows software-only control of block locking (for data that needs to be changed frequently), while the second level requires hardware interaction before locking can be changed (code which does not require frequent updates).

Control signals WP#, DQ1, and DQ0 define the state of a block; for example, state [001] means WP# = 0, DQ1 = 0 and DQ0 = 1.

Table 12 defines all of the possible locking states.

**NOTE:** All blocks are software-locked upon completion of a power-up sequence.

## LOCKED STATE

After a power-up sequence completion, or after a reset sequence, all blocks are locked (states [001] or [101]). This means full protection from alteration. Any PROGRAM or ERASE operations attempted on a locked block will return an error on bit SR1 of the status register. The status of a locked block can be changed to unlocked or lock down using the appropriate software commands. Writing the lock command sequence, 60h followed by 01h, can lock an unlocked block.

## UNLOCKED STATE

Unlocked blocks (states [000], [100], [110]) can be programmed or erased. All unlocked blocks return to the locked state when the device is reset or powered down. An unlocked block can be locked or locked down using the appropriate software command sequence, 60h followed by D0h (see Table 4).

## LOCKED DOWN STATE

Blocks that are locked down (state [011]) are protected from PROGRAM and ERASE operations, but their protection status cannot be changed using software commands alone. A locked or unlocked block can be locked down by writing the lock down command sequence, 60h followed by 2Fh. Locked down blocks re-

vert to the locked state when the device is reset or powered down.

The LOCK DOWN function is dependent on the WP# input. When WP# = 0, blocks in lock down [011] are protected from program, erase, and lock status changes. When WP# = 1, the lock down function is disabled ([111]), and locked down blocks can be individually unlocked by a software command to the [110] state, where they can be erased and programmed. These blocks can then be relocked [111] and unlocked [110] as desired while WP# remains HIGH. When WP# goes LOW, blocks that were previously locked down return to the locked down state [011] regardless of any changes made while WP# was HIGH. Device reset or power-down resets all locks, including those in lock down, to locked state (see Table 13).

## READING A BLOCK'S LOCK STATUS

The lock status of every block can be read in the read device identification mode. To enter this mode, write 90h to the device. Subsequent READs at block address +00002 will output the lock status of that block. The lowest two outputs, DQ0 and DQ1, represent the lock status. DQ0 indicates the block lock/unlock status and is set by the LOCK command and cleared by the UNLOCK command. It is also automatically set when entering lock down. DQ1 indicates lock down status and is set by the LOCK DOWN command. It can only be cleared by reset or power-down, not by software. Table 12 shows the locking state transition scheme.

## LOCKING OPERATIONS DURING ERASE SUSPEND

Changes to block lock status can be performed during an ERASE SUSPEND by using the standard locking command sequences to unlock, lock, or lock down. This is useful in the case when another block needs to be updated while an ERASE operation is in progress.

To change block locking during an ERASE operation, first write the ERASE SUSPEND command (B0h), then check the status register until it indicates that the ERASE operation has been suspended. Next, write the desired lock command sequence to block lock, and the lock status will be changed. After completing any desired LOCK, READ, or PROGRAM operation, resume the ERASE operation with the ERASE RESUME command (D0h).

If a block is locked or locked down during an ERASE SUSPEND operation on the same block, the locking status bits will be changed immediately. Then, when the ERASE is resumed, the ERASE operation will complete.

A locking operation cannot be performed during a PROGRAM SUSPEND.

### STATUS REGISTER ERROR CHECKING

Using nested locking or program command sequences during erase suspend can introduce ambiguity into status register results.

Following protection configuration setup (60h), an invalid command will produce a lock command error (SR4 and SR5 will be set to "1") in the status register. If a lock command error occurs during an ERASE SUSPEND, SR4 and SR5 will be set to "1" and will remain at "1" after the ERASE SUSPEND is resumed. When the ERASE is complete, any possible error during the ERASE cannot be detected via the status register because of the previous locking command error.

A similar situation happens if an error occurs during a program operation error nested within an ERASE SUSPEND.

### CHIP PROTECTION REGISTER

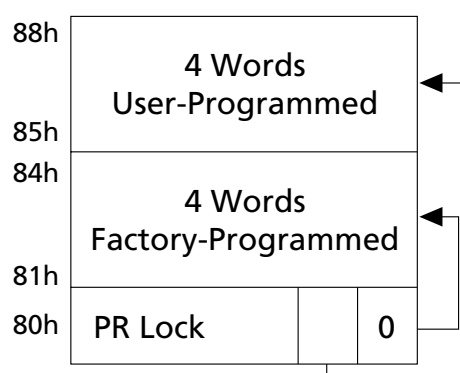
A 128-bit chip protection register can be used to fulfill the security considerations in the system (preventing device substitution).

The 128-bit security area is divided into two 64-bit segments. The first 64 bits are programmed at the manufacturing site with a unique 64-bit unchangeable number. The other segment is left blank for customers to program as desired. (See Figure 13).

### READING THE CHIP PROTECTION REGISTER

The chip protection register is read in the device identification mode. To enter this mode, load the 90h command to the bank containing address 00h. Once in this mode, READ cycles from addresses shown in Table 13 retrieve the specified information. To return to the read array mode, write the READ ARRAY command (FFh).

**Figure 13  
Protection Register Memory Map**



**Table 13  
Chip Configuration Addressing<sup>1</sup>**

| ITEM                          | ADDRESS <sup>2</sup> | DATA         |
|-------------------------------|----------------------|--------------|
| Manufacturer Code (x16)       | 00000h               | 002Ch        |
| Device Code                   | 00001h               |              |
| • Top boot configuration      |                      | 44B6         |
| • Bottom boot configuration   |                      | 44B7         |
| Block Lock Configuration      | XX002h               |              |
| • Block is unlocked           |                      | Lock         |
| • Block is locked             |                      | DQ0 = 0      |
| • Block is locked down        |                      | DQ0 = 1      |
| Read Configuration Register   | 00005h               | DQ1 = 1      |
| Chip Protection Register Lock | 80h                  | RCR          |
| Chip Protection Register 1    | 81h–84h              | PR Lock      |
| Chip Protection Register 2    | 85h–88h              | Factory Data |
|                               |                      | User Data    |

**NOTE:** 1. Other locations within the configuration address space are reserved by Micron for future use.  
2. "XX" specifies the block address of lock configuration.

## PROGRAMMING THE CHIP PROTECTION REGISTER

The first 64 bits (PR1) of the chip protection register (addresses 81h–84h) are programmed with a unique identifier at the factory. DQ0 of the PR lock register (address 80h) is programmed to a “0” state, locking the first 64 bits and preventing any further programming.

The second 64 bits (PR2) is a user area (addresses 85h–88h), where the user can program any information into this area as long as DQ1 of the PR lock register remains unprogrammed. After DQ1 of the PR lock register is programmed, no further programming is allowed on PR2. The programming sequence is similar to array programming except that the PROTECTION REGISTER PROGRAMMING SETUP command (C0h) is issued instead of an ARRAY PROGRAMMING SETUP command (40h), followed by the data to be programmed at addresses 85h–88h.

To program the PR lock bit for PR2 (to prevent further programming), use the above sequence on address 80h, with data of FFFDh (DQ1 = 0).

## ASYNCHRONOUS READ MODE

The asynchronous read mode is the default read configuration state. To use the device in an asynchronous-only application, ADV# and CLK must be tied to Vss and WAIT# should be floated.

Toggling the address lines from A0 to A21, the access is purely random (<sup>t</sup>AA).

The ADV# signal must be toggled to latch the address, and the CE# signal and the OE# signal must go LOW. In this case the data is placed on the data bus and the processor is ready to receive the data.

## SYNCHRONOUS BURST READ MODE

The burst read mode is used to achieve a faster data rate than is possible with asynchronous read mode. The rising edge of the clock (CLK) is used to latch the address with CE# and ADV# LOW (see timing diagram: Single Synchronous READ Operation). The burst read configuration is set in the read configuration register, where frequency, data output, WAIT# signal, burst se-

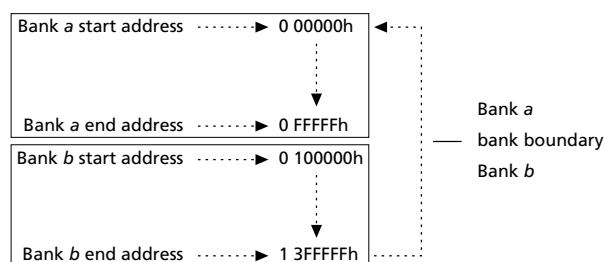
quence, clock, and burst length are configured setting the related bits.

All blocks in both banks can be burst read.

The BURST READ works across the bank boundary in the following way:

1. In a READ operation there is no bank boundary as far as burst access is concerned. If, for example, a burst starts in bank *a*, the application can keep clocking until the bank boundary is reached and then read from bank *b*. If the application keeps clocking beyond the last location of bank *b*, the internal counter restarts from bank *a* first address. (See Figure 14.)
2. If one bank is in program or erase mode and the application starts a burst access in that bank, then the status register data is returned. The internal address counter is incremented at every clock pulse.
3. If a burst access is started in one bank and the bank boundary is crossed, and the other bank is in program or erase mode, then the status register data is returned as the first location of the bank. If the application keeps clocking, the internal address counter gets incremented at every clock cycle. If bank end is crossed, then data from the other bank is returned as shown in Figure 14.

**Figure 14  
Bank Boundary Wrapping  
(Bottom Boot Example)**





## ASYNCHRONOUS PAGE READ MODE

After power-up or reset, the device operates in page mode over the whole memory array. The page size can be customized at the factory to four or eight words as required; but if no specification is made, the normal size is eight words. The initial portion of the page mode cycle is the same as the asynchronous access cycle. Holding CE# LOW and toggling addresses A0–A2 allows random access of other words in the page.

## V<sub>PP</sub>/V<sub>CC</sub> PROGRAM AND ERASE VOLTAGES

The Flash devices provide in-system programming and erase with V<sub>PP</sub> in the 0.9V–2.20V range. The 12V V<sub>PP</sub> mode programming is offered for compatibility with existing programming equipment, and it allows the APA execution as well.

The device can withstand 100,000 WRITE/ERASE operations, irrespective of the external V<sub>PP</sub> applied because the memory cells are always programmed using the internal power sources. This provides an optimal voltage profile in order to minimize the programming stress.

In addition to the flexible block locking, the V<sub>PP</sub> programming voltage can be held LOW for absolute hardware write protection of all blocks in the Flash device. When V<sub>PP</sub> is below V<sub>PPLK</sub>, any PROGRAM or ERASE operation will result in an error, prompting the corresponding status register bit (SR3) to be set.

During WRITE and ERASE operations, the WSM monitors the V<sub>PP</sub> voltage level. WRITE/ERASE operations are allowed only when V<sub>PP</sub> is within the ranges specified in Table 14.

When V<sub>CC</sub> is below V<sub>LKO</sub>, any WRITE/ERASE operation will be disabled.

**Table 14**  
**V<sub>PP</sub> Range (V)**

|            | MIN  | MAX  |
|------------|------|------|
| In System  | 0.9  | 2.2  |
| In Factory | 11.4 | 12.6 |

## STANDBY MODE

I<sub>CC</sub> supply current is reduced by applying a logic HIGH level on CE# and RST# to enter the standby mode. In the standby mode, the outputs are High-Z. Applying a CMOS logic HIGH level on CE# and RST# reduces the current to I<sub>CC4</sub> (MAX). If the device is deselected during an ERASE operation or during program-

ming, the device continues to draw current until the operation is complete.

## AUTOMATIC POWER SAVE (APS) MODE

Substantial power savings are realized during periods when the array is not being read and the device is in the active mode. During this time the device switches to the automatic power save mode. When the device switches to this mode, I<sub>CC</sub> is reduced to a level comparable to I<sub>CC4</sub>. Further power savings can be realized by applying a logic HIGH level on CE# to place the device in standby mode. The low level of power is maintained until another operation is initiated. In this mode, the I/Os retain the data from the last memory address read until a new address is read. This mode is entered automatically if no address or control signals toggle.

## DEVICE RESET

To correctly reset the Flash devices, the RST# signal must be asserted (RST# = V<sub>IL</sub>) for a minimum of <sup>t</sup>RP. After reset, the device can be accessed for a READ operation with a delayed access time of <sup>t</sup>RWH from the rising edge of RST#. RST# should be tied to the system reset to ensure that correct system initialization occurs. Please refer to the timing diagram for further details.

## DEEP POWER-DOWN

When RCR4 = 1, deep power-down can be enabled. In this configuration, applying a logic LOW to RST# reduces the current to I<sub>CC10</sub> and resets all the internal registers, with the exception of the RCR and the individual block protection status. To exit this mode, a wait time of 100μs (<sup>t</sup>RWHDP) must elapse after a logic HIGH is applied to RST#.

During the wait time, the device performs a full power-up sequence, and the power consumption may exceed the standby current limits.

## POWER-UP SEQUENCE

The following power-up sequence is recommended to initialize internal chip operations:

- At power-up, RST# should be kept at V<sub>IL</sub> for 2μs after V<sub>CC</sub> reaches V<sub>CC</sub> (MIN).
- V<sub>CCQ</sub> should not come up before V<sub>CC</sub>.
- V<sub>PP</sub> should be kept at V<sub>IL</sub> to maximize data integrity.

When the power-up sequence is completed, RST# should be brought to V<sub>IH</sub>. To ensure a proper power-up, the rise time of RST# (10%–90%) should be < 10μs.


**ABSOLUTE MAXIMUM RATINGS\***

Voltage to Any Ball Except V<sub>CC</sub> and V<sub>PP</sub>  
 with Respect to V<sub>SS</sub> ..... -0.5V to +2.45V  
 V<sub>PP</sub> Voltage (for BLOCK ERASE and PROGRAM  
 with Respect to V<sub>SS</sub>) ..... -0.5V to +13.5V\*\*  
 V<sub>CC</sub> and V<sub>CCQ</sub> Supply Voltage  
 with Respect to V<sub>SS</sub> ..... -0.3V to +2.45V  
 Output Short Circuit Current ..... 100mA  
 Operating Temperature Range ..... -40°C to +85°C  
 Storage Temperature Range ..... -55°C to +125°C  
 Soldering Cycle ..... 260°C for 10s

\*Stresses greater than those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

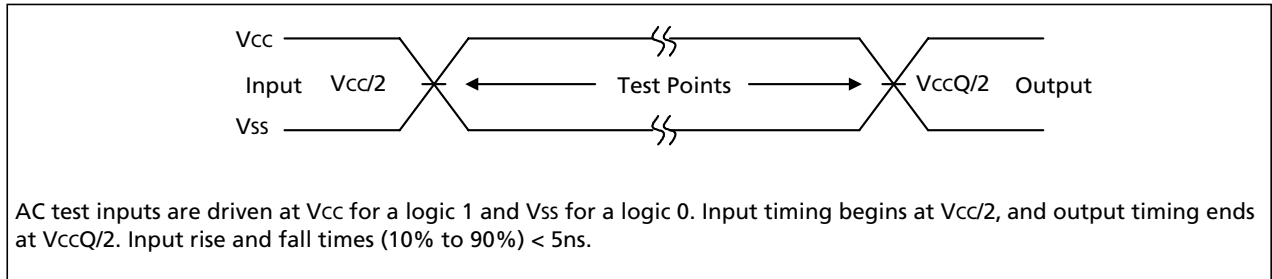
\*\*Maximum DC voltage on V<sub>PP</sub> may overshoot to +13.5V for periods < 20ns.

**RECOMMENDED OPERATING CONDITIONS**

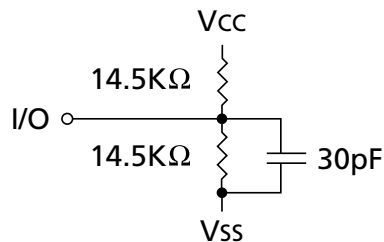
| PARAMETER                                           | SYMBOL                             | MIN              | MAX     | UNITS | NOTES    |
|-----------------------------------------------------|------------------------------------|------------------|---------|-------|----------|
| Operating temperature                               | t <sub>A</sub>                     | -40              | +85     | °C    |          |
| V <sub>CC</sub> supply voltage (MT28F642D18)        | V <sub>CC</sub>                    | 1.70             | 1.90    | V     |          |
| V <sub>CC</sub> supply voltage (MT28F642D20)        | V <sub>CC</sub>                    | 1.80             | 2.20    | V     |          |
| I/O supply voltage (V <sub>CC</sub> = 1.70–1.90V)   | V <sub>CCQ</sub>                   | 1.70             | 1.90    | V     |          |
| I/O supply voltage (V <sub>CC</sub> = 1.80–2.25V)   | V <sub>CCQ</sub>                   | 1.80             | 2.25    | V     |          |
| V <sub>PP</sub> voltage, when used as logic control | V <sub>PP1</sub>                   | 0.9              | 2.20    | V     |          |
| V <sub>PP</sub> in-factory programming voltage      | V <sub>PP2</sub>                   | 11.4             | 12.6    | V     |          |
| Block erase cycling                                 | V <sub>PP</sub> = V <sub>PP1</sub> | V <sub>PP1</sub> | 100,000 | –     | Cycles   |
| Block erase cycling                                 | V <sub>PP</sub> = V <sub>PP2</sub> | V <sub>PP2</sub> | –       | 100   | Cycles 1 |

**NOTE:** 1. V<sub>PP</sub> = V<sub>PP2</sub> is a maximum of 10 cumulative hours.

**Figure 15**  
**AC Input/Output Reference Waveform**



**Figure 16**  
**Output Load Circuit**



## CAPACITANCE

( $T_A = +25^\circ\text{C}$ ;  $f = 1 \text{ MHz}$ )

| PARAMETER/CONDITION <sup>1</sup> | SYMBOL           | TYP | MAX | UNITS |
|----------------------------------|------------------|-----|-----|-------|
| Input Capacitance                | C                | 7   | 12  | pF    |
| Output Capacitance               | C <sub>OUT</sub> | 9   | 12  | pF    |


**DC CHARACTERISTICS<sup>1</sup>**

| PARAMETER                                                                                                                                               | SYMBOL            | MT28F642D20<br>MT28F642D18 |        |                  | UNITS    | NOTES |
|---------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|----------------------------|--------|------------------|----------|-------|
|                                                                                                                                                         |                   | MIN                        | TYP    | MAX              |          |       |
| Input Low Voltage                                                                                                                                       | V <sub>IL</sub>   | 0                          | –      | 0.4              | V        | 2     |
| Input High Voltage                                                                                                                                      | V <sub>IH</sub>   | V <sub>CCQ</sub> - 0.4V    | –      | V <sub>CCQ</sub> | V        | 2     |
| Output Low Voltage<br>I <sub>OL</sub> = 100μA                                                                                                           | V <sub>OL</sub>   | –                          | –      | 0.1              | V        |       |
| Output High Voltage<br>I <sub>OH</sub> = -100μA                                                                                                         | V <sub>OH</sub>   | V <sub>CCQ</sub> - 0.1V    | –      | –                | V        |       |
| V <sub>PP</sub> Lockout Voltage                                                                                                                         | V <sub>PPLK</sub> | –                          | –      | 0.4              | V        |       |
| V <sub>PP</sub> During PROGRAM/ERASE Operations                                                                                                         | V <sub>PP1</sub>  | 0.9                        | –      | 2.2              | V        |       |
|                                                                                                                                                         | V <sub>PP2</sub>  | 11.4                       | –      | 12.6             | V        |       |
| V <sub>CC</sub> Program/Erase Lock Voltage                                                                                                              | V <sub>LKO</sub>  | 1                          | –      | –                | V        |       |
| Input Leakage Current                                                                                                                                   | I <sub>L</sub>    | –                          | –      | 1                | μA       |       |
| Output Leakage Current                                                                                                                                  | I <sub>OZ</sub>   | 0.2                        | –      | 1                | μA       |       |
| V <sub>CC</sub> Read Current<br>Asynchronous Random Read, 70ns cycle                                                                                    | I <sub>CC1</sub>  | –                          | –      | 15               | mA       | 3, 4  |
| V <sub>CC</sub> Page Mode Read Current, 70ns/30ns cycle                                                                                                 | I <sub>CC2</sub>  | –                          | –      | 5                | mA       | 3, 4  |
| V <sub>CC</sub> Burst Mode Read Current, 18.5ns cycle                                                                                                   | I <sub>CC3</sub>  | –                          | –      | 10               | mA       | 4     |
| V <sub>CC</sub> Standby Current                                                                                                                         | I <sub>CC4</sub>  | –                          | 25     | 50               | μA       |       |
| V <sub>CC</sub> Program Current                                                                                                                         | I <sub>CC5</sub>  | –                          | –      | 55               | mA       |       |
| V <sub>CC</sub> Erase Current                                                                                                                           | I <sub>CC6</sub>  | –                          | –      | 65               | mA       |       |
| V <sub>CC</sub> Erase Suspend Current                                                                                                                   | I <sub>CC7</sub>  | –                          | 25     | 50               | μA       | 5     |
| V <sub>CC</sub> Program Suspend Current                                                                                                                 | I <sub>CC8</sub>  | –                          | 25     | 50               | μA       | 5     |
| Read-While-Write Current                                                                                                                                | I <sub>CC9</sub>  | –                          | –      | 80               | mA       |       |
| Deep Power-Down Current                                                                                                                                 | I <sub>CC10</sub> | –                          | 15     | 25               | μA       |       |
| V <sub>PP</sub> Current<br>(Read, Standby, Erase Suspend,<br>Program Suspend)<br>V <sub>PP</sub> ≤ V <sub>CC</sub><br>V <sub>PP</sub> ≥ V <sub>CC</sub> | I <sub>PP1</sub>  | –<br>–                     | –<br>– | 1<br>200         | μA<br>μA |       |

- NOTE:**
1. All currents are in RMS unless otherwise noted.
  2. V<sub>IL</sub> may decrease to -0.4V, and V<sub>IH</sub> may increase to V<sub>CCQ</sub> + 0.3V for durations not to exceed 20ns.
  3. APS mode reduces I<sub>CC</sub> to approximately I<sub>CC4</sub> levels.
  4. Test conditions: V<sub>CC</sub> = V<sub>CC</sub> (MAX), CE# = V<sub>IL</sub>, OE# = V<sub>IH</sub>. All other inputs = V<sub>IH</sub> or V<sub>IL</sub>.
  5. I<sub>CC7</sub> and I<sub>CC8</sub> values are valid when the device is deselected. Any READ operation performed while in suspend mode will have an additional current draw of suspend current (I<sub>CC7</sub> or I<sub>CC8</sub>).


**ASYNCHRONOUS READ CYCLE TIMING REQUIREMENTS<sup>1</sup>**

| PARAMETER                                   | SYMBOL      | -70 |     | -80 |     | UNITS   | NOTES |
|---------------------------------------------|-------------|-----|-----|-----|-----|---------|-------|
|                                             |             | MIN | MAX | MIN | MAX |         |       |
| Address setup to ADV# HIGH                  | $t_{AVS}$   | 10  |     | 10  |     |         |       |
| CE# LOW to ADV# HIGH                        | $t_{CVS}$   | 10  |     | 10  |     |         |       |
| READ cycle time                             | $t_{RC}$    |     | 70  |     | 80  |         |       |
| Address to output delay                     | $t_{AA}$    |     | 70  |     | 80  | ns      |       |
| CE# LOW to output delay                     | $t_{ACE}$   |     | 70  |     | 80  | ns      |       |
| ADV# LOW to output delay                    | $t_{AADV}$  |     | 70  |     | 80  |         |       |
| ADV# pulse width LOW                        | $t_{VP}$    | 10  |     | 10  |     |         |       |
| ADV# pulse width HIGH                       | $t_{VPH}$   | 10  |     | 10  |     |         |       |
| Address hold from ADV# HIGH                 | $t_{AVH}$   | 3   |     | 3   |     |         |       |
| Page address access                         | $t_{APA}$   |     | 30  |     | 30  | ns      |       |
| OE# LOW to output delay                     | $t_{AOE}$   |     | 25  |     | 25  | ns      |       |
| RST# HIGH to output delay                   | $t_{RWH}$   |     | –   |     | –   | ns      | 2     |
| CE# or OE# HIGH to output High-Z            | $t_{OD}$    |     | 15  |     | 25  | ns      |       |
| Output hold from address, CE# or OE# change | $t_{OH}$    | 0   |     | 0   |     | ns      |       |
| RST# deep power-down                        | $t_{RWHDP}$ |     | 100 |     | 100 | $\mu$ s |       |

**BURST READ CYCLE TIMING REQUIREMENTS<sup>1</sup>**

| PARAMETER                                     | SYMBOL     | -705 |     | -804 |     | UNITS |
|-----------------------------------------------|------------|------|-----|------|-----|-------|
|                                               |            | MIN  | MAX | MIN  | MAX |       |
| CLK period                                    | $t_{CLK}$  | 18.5 |     | 25   |     | ns    |
| CLK HIGH (LOW) time                           | $t_{KP}$   | 5    |     | 7.5  |     | ns    |
| CLK fall (rise) time                          | $t_{KHKL}$ |      | 3   |      | 5   | ns    |
| Address valid set up to clock                 | $t_{AKS}$  | 7    |     | 7    |     | ns    |
| ADV# LOW set to CLK                           | $t_{VKS}$  | 7    |     | 7    |     | ns    |
| CE# LOW set to CLK                            | $t_{CKS}$  | 9    |     | 13   |     | ns    |
| CLK to output delay                           | $t_{ACLK}$ |      | 15  |      | 20  | ns    |
| Output hold from CLK                          | $t_{KOH}$  | 3.5  |     | 5    |     | ns    |
| Address hold from CLK                         | $t_{AKH}$  | 10   |     | 10   |     | ns    |
| CLK to WAIT# delay                            | $t_{KHTL}$ |      | 15  |      | 20  | ns    |
| CE# HIGH between subsequent synchronous READs | $t_{CBPH}$ | 20   |     | 20   |     | ns    |

**NOTE:** 1. See Figures 17 and 18 for timing requirements and load configuration.  
 2. For the MT28F642D18,  $t_{RWH}$  = 250ns (MAX); for the MT28F642D20,  $t_{RWH}$  = 200ns (MAX).

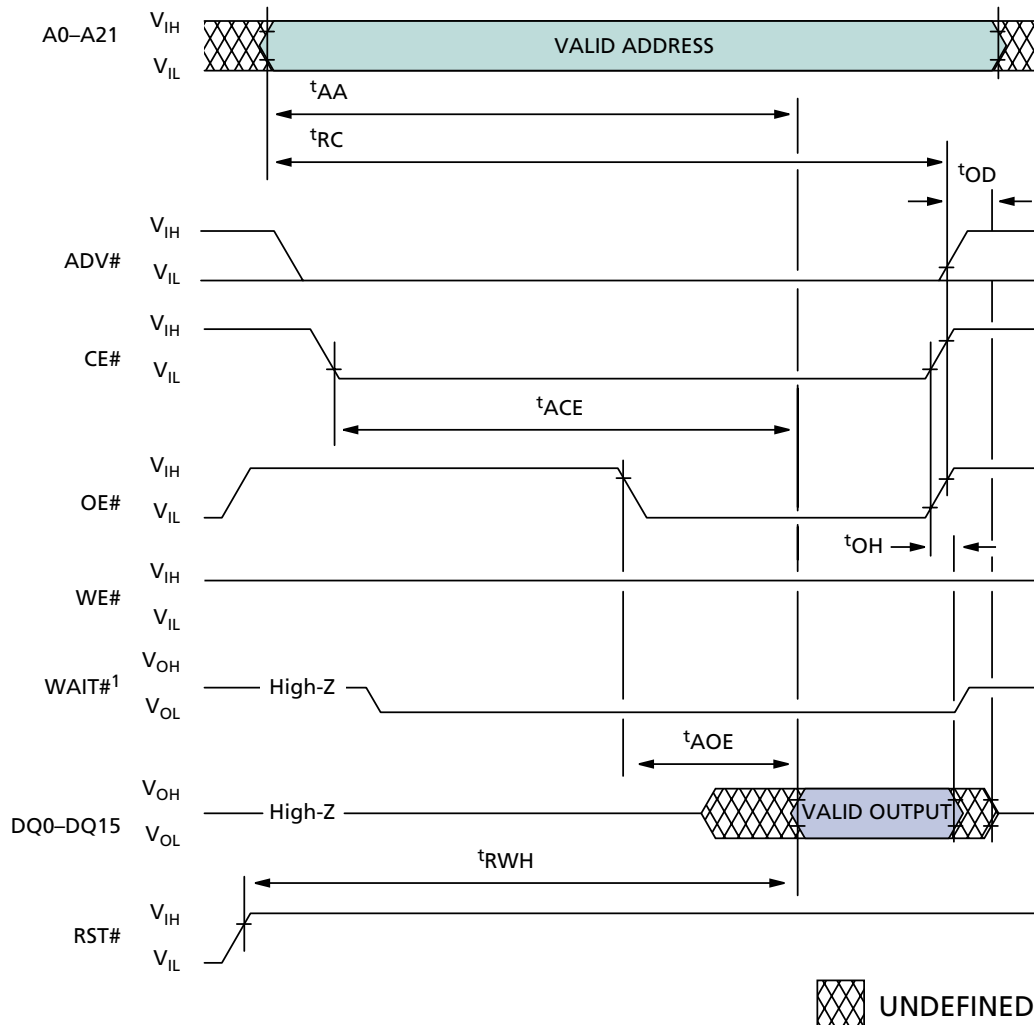


## WRITE CYCLE TIMING REQUIREMENTS

| PARAMETER                               | SYMBOL            | -70/-80 |                      | UNITS |
|-----------------------------------------|-------------------|---------|----------------------|-------|
|                                         |                   | MIN     | MAX                  |       |
| HIGH recovery to WE# going LOW          | t <sub>RS</sub>   | 150     |                      | ns    |
| CE# setup to WE# going LOW              | t <sub>CS</sub>   | 0       |                      | ns    |
| Write pulse width                       | t <sub>WP</sub>   | 70      |                      | ns    |
| ADV# pulse width                        | t <sub>VP</sub>   | 10      |                      | ns    |
| Data setup to WE# going HIGH            | t <sub>DS</sub>   | 70      |                      | ns    |
| Address setup to WE# going HIGH         | t <sub>AS</sub>   | 50      |                      | ns    |
| ADV# setup to WE# going HIGH            | t <sub>VS</sub>   | 70      |                      | ns    |
| Address setup to ADV# going HIGH        | t <sub>AVS</sub>  | 10      |                      | ns    |
| CE# hold from WE# HIGH                  | t <sub>CH</sub>   | 0       |                      | ns    |
| Data hold from WE# HIGH                 | t <sub>DH</sub>   | 0       |                      | ns    |
| Address hold from WE# HIGH              | t <sub>AH</sub>   | 0       |                      | ns    |
| Address hold to ADV# going HIGH         | t <sub>AVH</sub>  | 3       |                      | ns    |
| Write pulse width High                  | t <sub>WPH</sub>  | 30      |                      | ns    |
| RST# pulse width                        | t <sub>RP</sub>   | 100     |                      | ns    |
| WP# setup to WE# going HIGH             | t <sub>RHS</sub>  | 0       |                      | ns    |
| V <sub>PP</sub> setup to WE# going HIGH | t <sub>VPS</sub>  | 200     |                      | ns    |
| Write recovery before Read              | t <sub>WOS</sub>  | 50      |                      | ns    |
| WP# hold from valid SRD                 | t <sub>RHH</sub>  | 0       |                      | ns    |
| V <sub>PP</sub> hold from valid SRD     | t <sub>VPPH</sub> | 0       |                      | ns    |
| WE# HIGH to data valid                  | t <sub>WB</sub>   |         | t <sub>AA</sub> + 50 | ns    |

## ERASE AND PROGRAM TIMING REQUIREMENTS

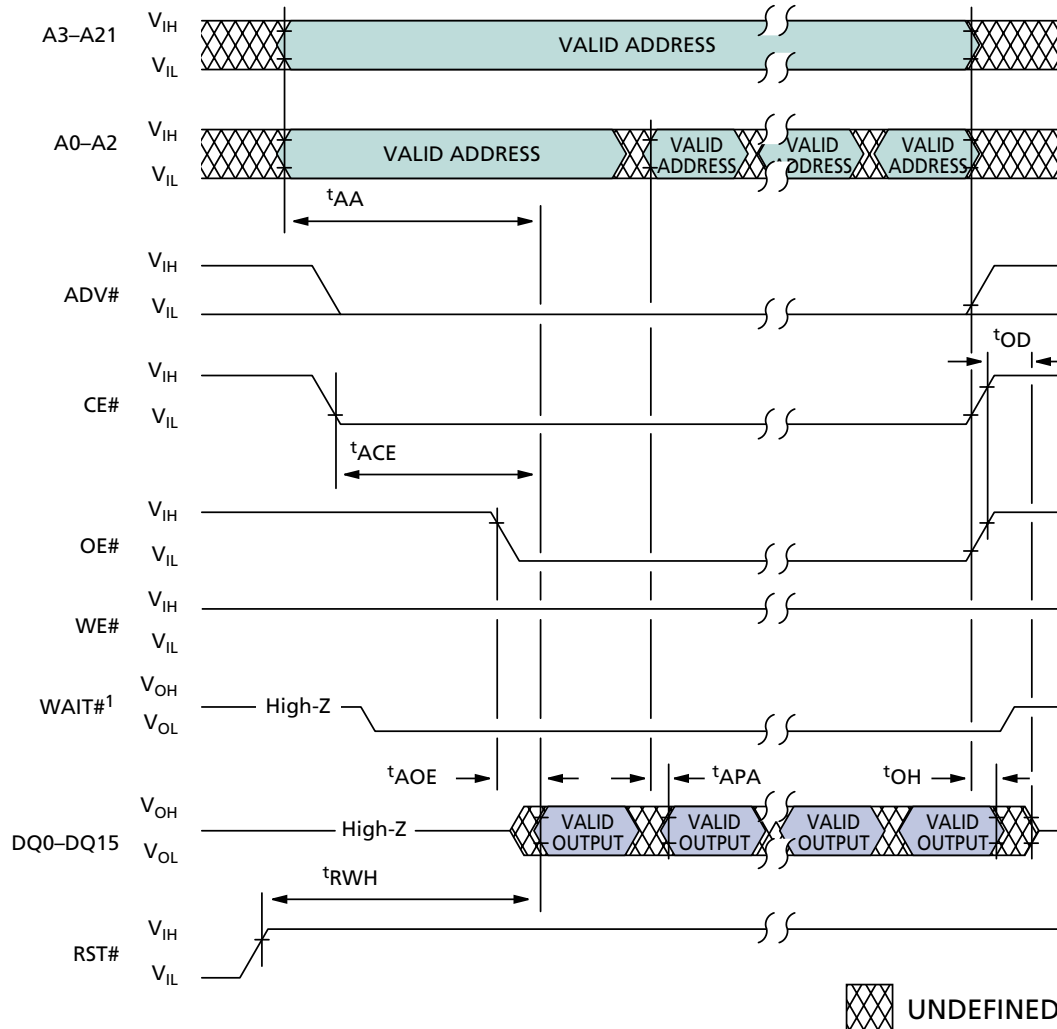
| PARAMETER                          | -70/-80 |        | UNITS |
|------------------------------------|---------|--------|-------|
|                                    | TYP     | MAX    |       |
| 4 KW parameter block program time  | 40      | 800    | ms    |
| 32 KW parameter block program time | 320     | 6,400  | ms    |
| Word program time                  | 8       | 10,000 | μs    |
| 4 KW parameter block erase time    | 0.3     | 6      | s     |
| 32 KW parameter block erase time   | 0.5     | 6      | s     |
| Program suspend latency            | 5       | 10     | μs    |
| Erase suspend latency              | 5       | 20     | μs    |
| Chip programming time (APA)        |         | 20     | s     |

**SINGLE ASYNCHRONOUS READ OPERATION**

**READ TIMING PARAMETERS**

| SYMBOL    | -70 |     | -80 |     | UNITS |
|-----------|-----|-----|-----|-----|-------|
|           | MIN | MAX | MIN | MAX |       |
| $t_{AA}$  |     | 70  |     | 80  | ns    |
| $t_{ACE}$ |     | 70  |     | 80  | ns    |
| $t_{AOE}$ |     | 25  |     | 25  | ns    |
| $t_{RC}$  |     | 70  |     | 80  | ns    |

| SYMBOL      | -70 |     | -80 |     | UNITS |
|-------------|-----|-----|-----|-----|-------|
|             | MIN | MAX | MIN | MAX |       |
| $t_{RWH}^2$ |     | —   |     | —   | ns    |
| $t_{OD}$    |     | 15  |     | 25  | ns    |
| $t_{OH}$    | 0   |     | 0   |     | ns    |

**NOTE:** 1. WAIT# is shown active LOW.  
2. For the MT28F642D18,  $t_{RWH} = 250\text{ns}$  (MAX); for the MT28F642D20,  $t_{RWH} = 200\text{ns}$  (MAX).

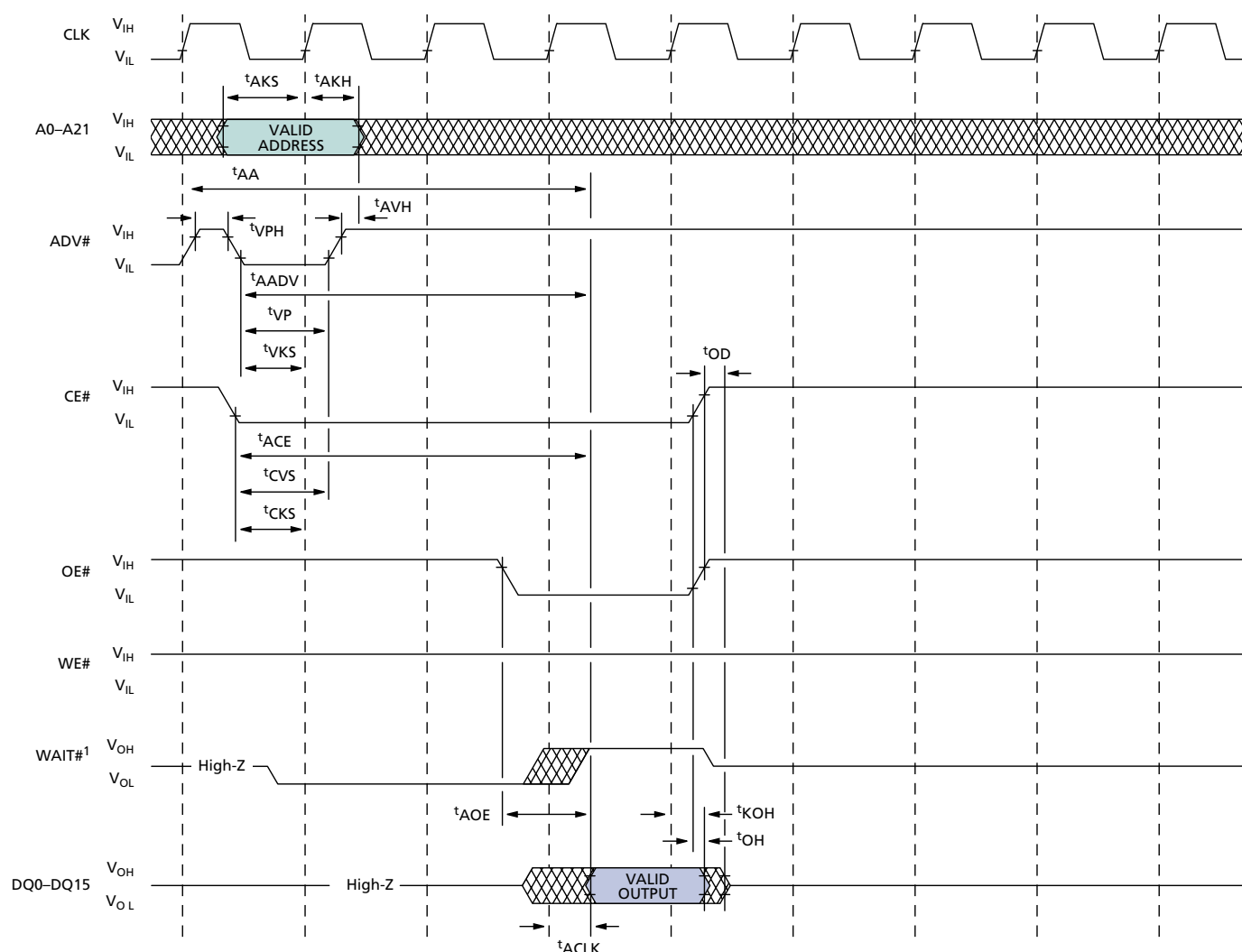
**ASYNCHRONOUS PAGE MODE READ OPERATION**

**READ TIMING PARAMETERS**

| SYMBOL    | -70 |     | -80 |     | UNITS |
|-----------|-----|-----|-----|-----|-------|
|           | MIN | MAX | MIN | MAX |       |
| $t_{AA}$  |     | 70  |     | 80  | ns    |
| $t_{ACE}$ |     | 70  |     | 80  | ns    |
| $t_{AOE}$ |     | 25  |     | 25  | ns    |
| $t_{RC}$  |     | 70  |     | 80  | ns    |

| SYMBOL      | -70 |     | -80 |     | UNITS |
|-------------|-----|-----|-----|-----|-------|
|             | MIN | MAX | MIN | MAX |       |
| $t_{RWH}^1$ |     | —   |     | —   | ns    |
| $t_{OD}$    |     | 15  |     | 25  | ns    |
| $t_{OH}$    | 0   |     | 0   |     | ns    |

**NOTE:** 1. WAIT# is shown active LOW.

2. For the MT28F642D18,  $t_{RWH} = 250\text{ns}$  (MAX); for the MT28F642D20,  $t_{RWH} = 200\text{ns}$  (MAX).


**SINGLE SYNCHRONOUS READ OPERATION**


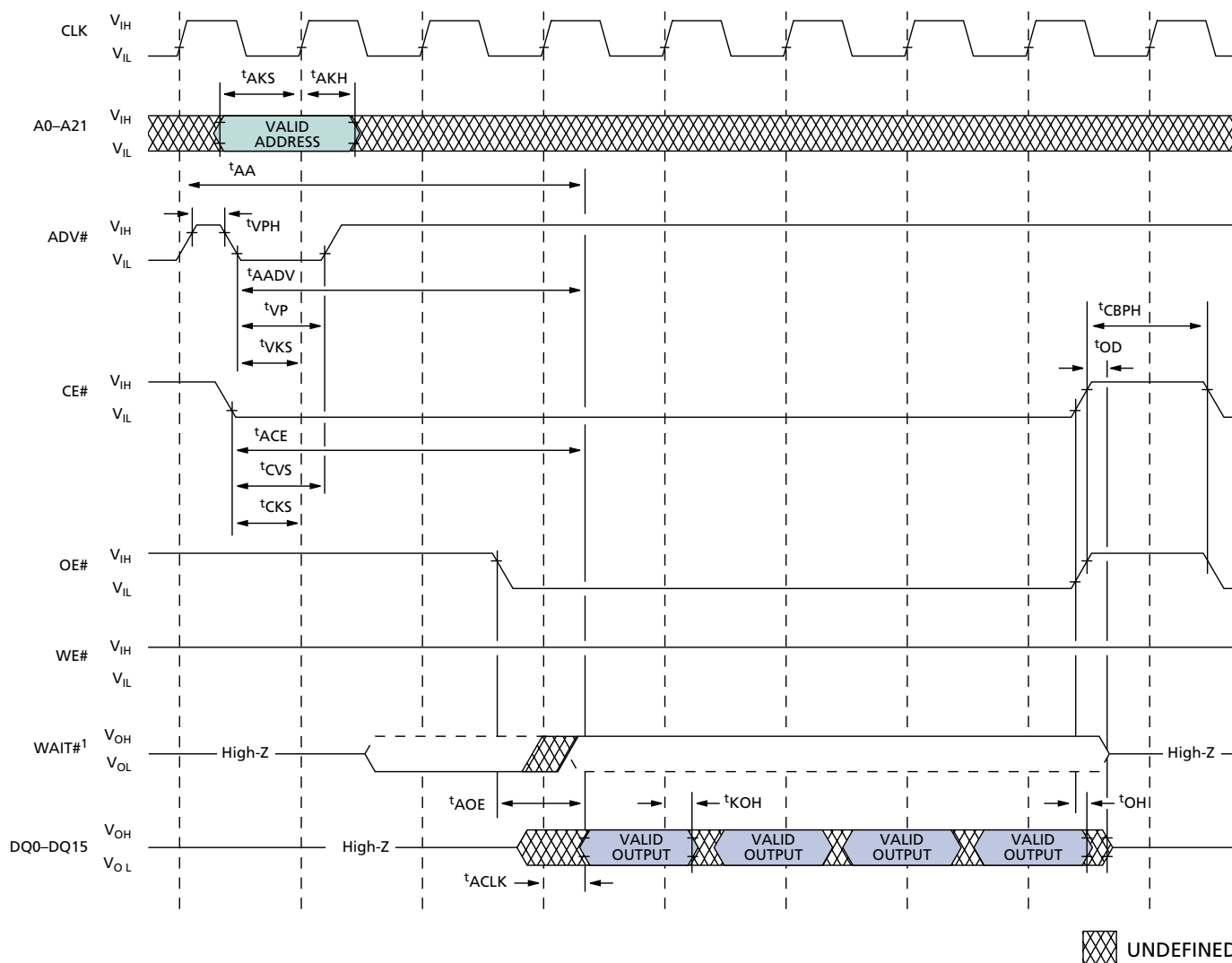
UNDEFINED

**READ TIMING PARAMETERS**

| SYMBOL    | -70 |     | -80 |     | UNITS |
|-----------|-----|-----|-----|-----|-------|
|           | MIN | MAX | MIN | MAX |       |
| $t_{AKS}$ | 7   |     | 7   |     | ns    |
| $t_{VKS}$ | 7   |     | 7   |     | ns    |
| $t_{CKS}$ | 9   |     | 13  |     | ns    |
| $t_{KOH}$ | 3.5 |     | 5   |     | ns    |
| $t_{AKH}$ | 10  |     | 10  |     | ns    |
| $t_{CVS}$ | 10  |     | 10  |     |       |
| $t_{AA}$  |     | 70  |     | 80  | ns    |
| $t_{ACE}$ |     | 70  |     | 80  | ns    |

| SYMBOL     | -70 |     | -80 |     | UNITS |
|------------|-----|-----|-----|-----|-------|
|            | MIN | MAX | MIN | MAX |       |
| $t_{AADV}$ |     | 70  |     | 80  |       |
| $t_{VP}$   | 10  |     | 10  |     |       |
| $t_{VPH}$  | 10  |     | 10  |     |       |
| $t_{AVH}$  | 3   |     | 3   |     |       |
| $t_{AOE}$  |     | 25  |     | 25  | ns    |
| $t_{OD}$   |     | 15  |     | 25  | ns    |
| $t_{OH}$   | 0   |     | 0   |     | ns    |

**NOTE:** 1. WAIT# is shown active LOW.

**4-WORD SYNCHRONOUS BURST OPERATION**

**READ TIMING PARAMETERS**

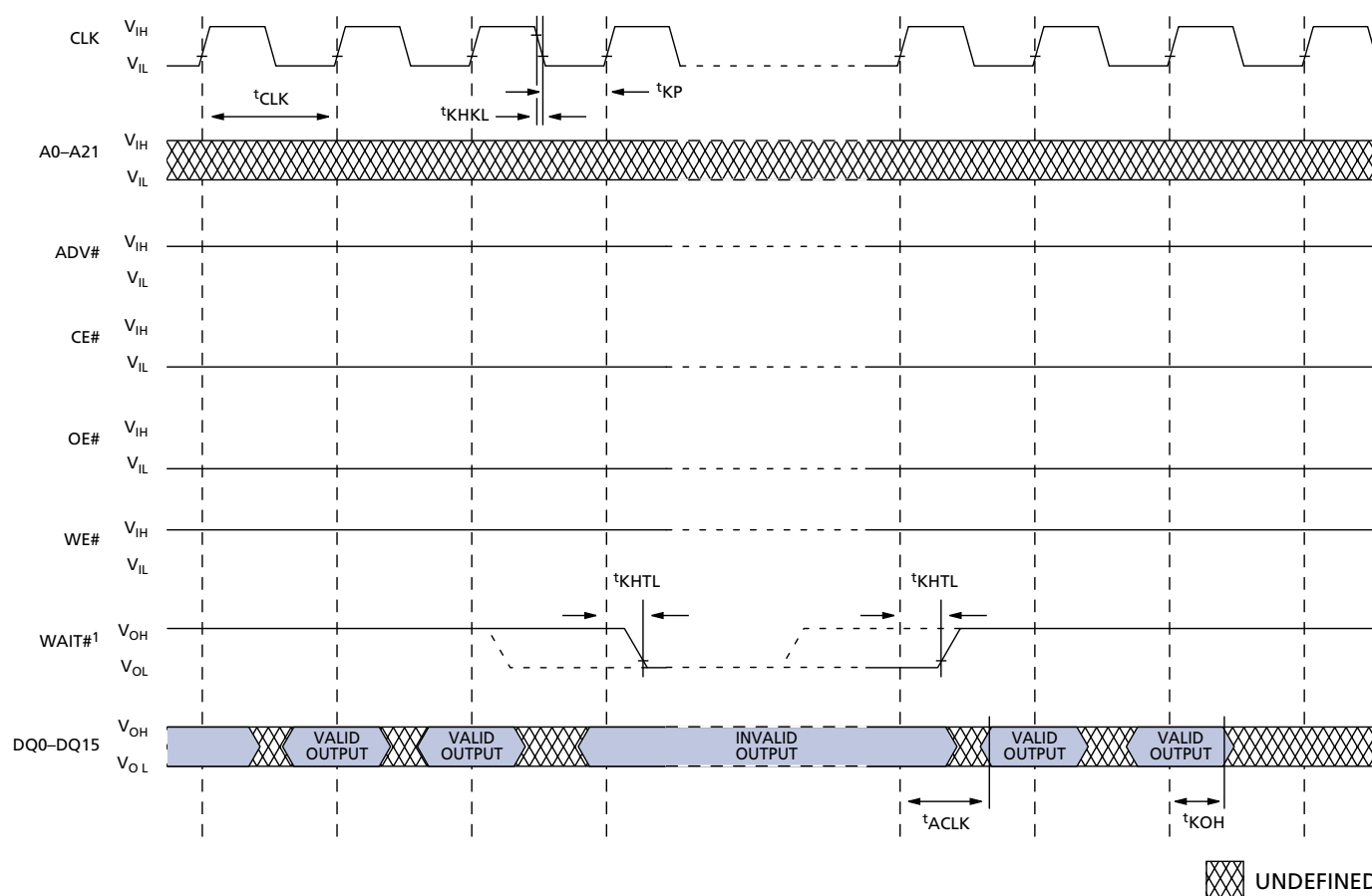
| SYMBOL    | -70 |     | -80 |     | UNITS |
|-----------|-----|-----|-----|-----|-------|
|           | MIN | MAX | MIN | MAX |       |
| $t_{AKS}$ | 7   |     | 7   |     | ns    |
| $t_{VKS}$ | 7   |     | 7   |     | ns    |
| $t_{CKS}$ | 9   |     | 13  |     | ns    |
| $t_{KOH}$ | 3.5 |     | 5   |     | ns    |
| $t_{AKH}$ | 10  |     | 10  |     | ns    |
| $t_{CVS}$ | 10  |     | 10  |     |       |
| $t_{AA}$  |     | 70  |     | 80  | ns    |
| $t_{ACE}$ |     | 70  |     | 80  | ns    |

| SYMBOL     | -70 |     | -80 |     | UNITS |
|------------|-----|-----|-----|-----|-------|
|            | MIN | MAX | MIN | MAX |       |
| $t_{AADV}$ |     | 70  |     | 80  |       |
| $t_{VP}$   | 10  |     | 10  |     |       |
| $t_{VPH}$  | 10  |     | 10  |     |       |
| $t_{AVH}$  | 3   |     | 3   |     |       |
| $t_{AOE}$  |     | 25  |     | 25  | ns    |
| $t_{OD}$   |     | 15  |     | 25  | ns    |
| $t_{OH}$   | 0   |     | 0   |     | ns    |

**NOTE:** 1. WAIT# is shown active LOW.



**CONTINUOUS BURST READ**  
**SHOWING AN OUTPUT DELAY WITH RCR8 = 0(1)**

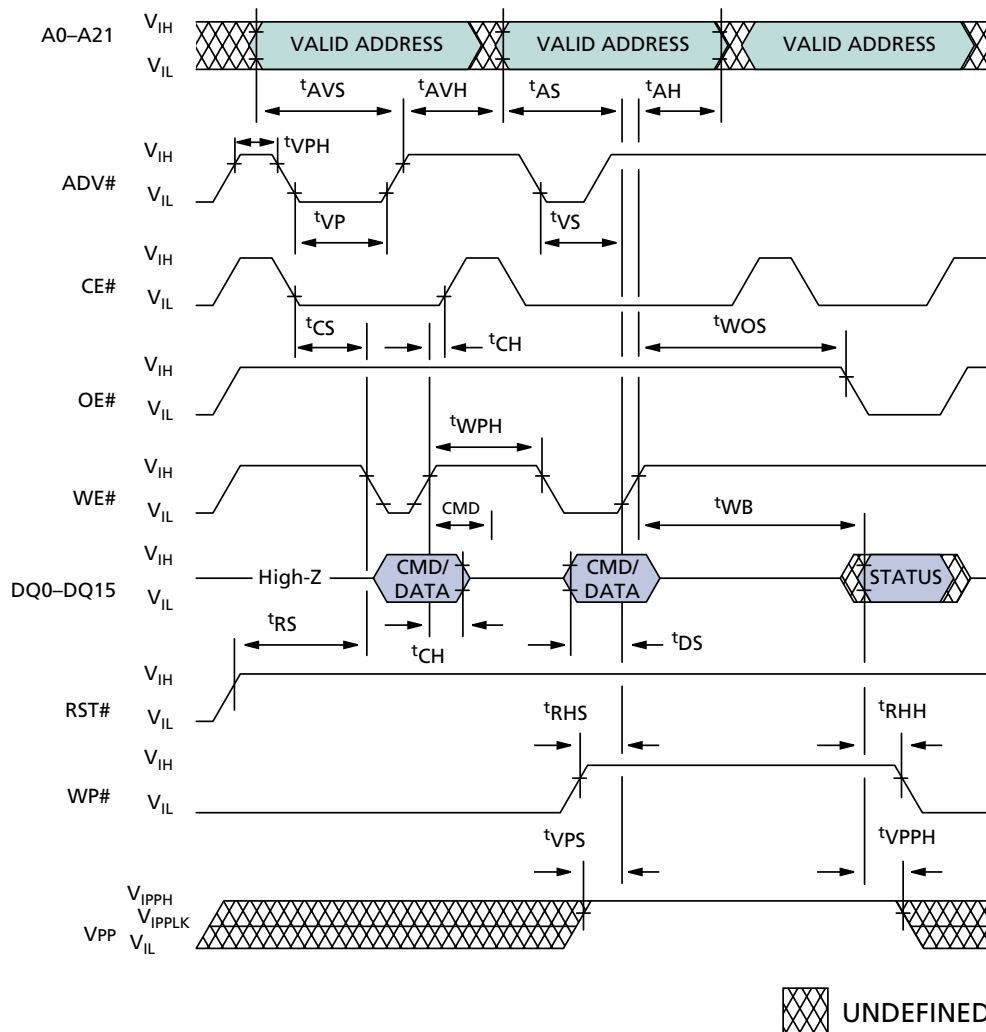


**READ TIMING PARAMETERS**

| SYMBOL    | -70  |     | -80 |     | UNITS |
|-----------|------|-----|-----|-----|-------|
|           | MIN  | MAX | MIN | MAX |       |
| $t_{CLK}$ | 18.5 |     | 25  |     | ns    |
| $t_{KP}$  | 5    |     | 7.5 |     | ns    |
| $t_{KHL}$ |      | 3   |     | 5   | ns    |

| SYMBOL     | -70 |     | -80 |     | UNITS |
|------------|-----|-----|-----|-----|-------|
|            | MIN | MAX | MIN | MAX |       |
| $t_{ACLK}$ |     | 15  |     | 20  | ns    |
| $t_{KOH}$  | 3.5 |     | 5   |     | ns    |
| $t_{KHTL}$ |     | 15  |     | 20  | ns    |

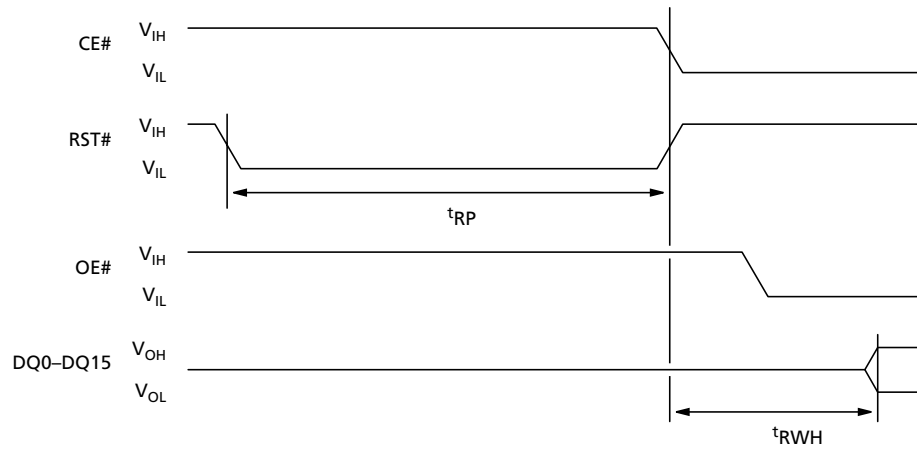
**NOTE:** 1. WAIT# is shown active LOW.

**TWO-CYCLE PROGRAMMING/ERASE OPERATION**

**WRITE TIMING PARAMETERS**

| SYMBOL           | -70/-80 |     | UNITS |
|------------------|---------|-----|-------|
|                  | MIN     | MAX |       |
| t <sub>RS</sub>  | 150     |     | ns    |
| t <sub>CS</sub>  | 0       |     | ns    |
| t <sub>WP</sub>  | 70      |     | ns    |
| t <sub>VP</sub>  | 10      |     | ns    |
| t <sub>DS</sub>  | 70      |     | ns    |
| t <sub>AS</sub>  | 50      |     | ns    |
| t <sub>VS</sub>  | 70      |     | ns    |
| t <sub>AVS</sub> | 10      |     | ns    |
| t <sub>CH</sub>  | 0       |     | ns    |
| t <sub>DH</sub>  | 0       |     | ns    |

| SYMBOL            | -70/-80 |                      | UNITS |
|-------------------|---------|----------------------|-------|
|                   | MIN     | MAX                  |       |
| t <sub>AH</sub>   | 0       |                      | ns    |
| t <sub>AVH</sub>  | 3       |                      | ns    |
| t <sub>WPH</sub>  | 30      |                      | ns    |
| t <sub>RP</sub>   | 100     |                      | ns    |
| t <sub>RHS</sub>  | 0       |                      | ns    |
| t <sub>VPS</sub>  | 200     |                      | ns    |
| t <sub>WOS</sub>  | 50      |                      | ns    |
| t <sub>RHH</sub>  | 0       |                      | ns    |
| t <sub>VPPH</sub> | 0       |                      | ns    |
| t <sub>WB</sub>   |         | t <sub>AA</sub> + 50 | ns    |

### RESET OPERATION



### READ AND WRITE TIMING PARAMETERS

| SYMBOL      | -70 |     | -80 |     | UNITS |
|-------------|-----|-----|-----|-----|-------|
|             | MIN | MAX | MIN | MAX |       |
| $t_{RWH}^1$ |     | —   |     | —   | ns    |
| $t_{RP}$    | 100 |     | 100 |     | ns    |

**NOTE:** 1. For the MT28F642D18,  $t_{RWH} = 250\text{ns}$  (MAX); for the MT28F642D20,  $t_{RWH} = 200\text{ns}$  (MAX).

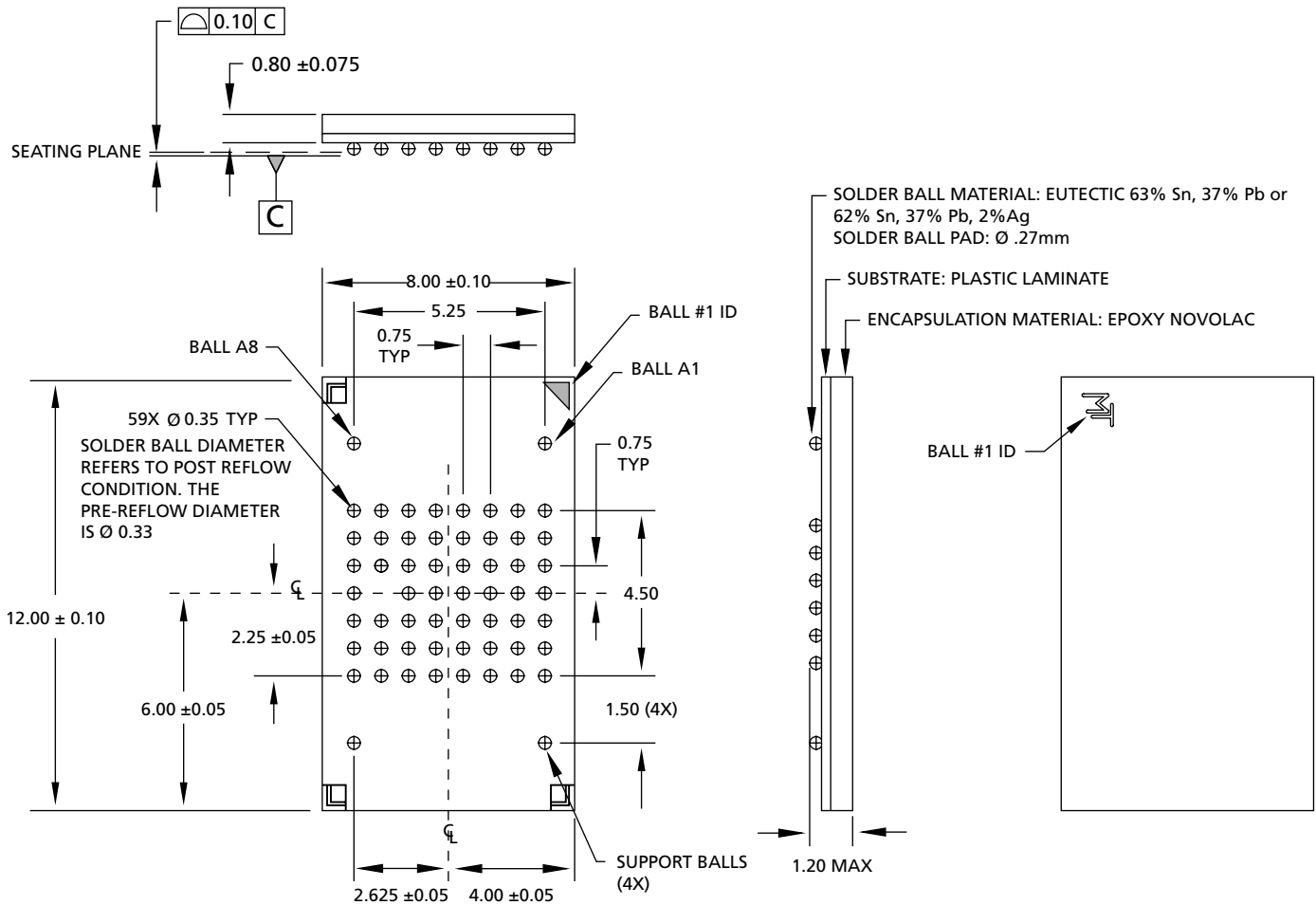

**Table 15**  
**CFI**

| OFFSET  | DATA       | DESCRIPTION                                                                                                     |
|---------|------------|-----------------------------------------------------------------------------------------------------------------|
| 0       | 2Ch        | Manufacturer code                                                                                               |
| 1       | B6h        | Top boot block device code                                                                                      |
|         | B7h        | Bottom boot block device code                                                                                   |
| 02 – 0F | reserved   | Reserved                                                                                                        |
| 10,11   | 0051, 0052 | "QR"                                                                                                            |
| 12      | 0059       | "Y"                                                                                                             |
| 13, 14  | 0003, 0000 | Primary OEM command set                                                                                         |
| 15, 16  | 0039, 0000 | Address for primary extended table                                                                              |
| 17, 18  | 0000, 0000 | Alternate OEM command set                                                                                       |
| 19, 1A  | 0000, 0000 | Address for OEM extended table                                                                                  |
| 1B      | 0017       | V <sub>DD</sub> MIN for Erase/Write; Bit7–Bit4 Volts in BCD, Bit3–Bit0 100mV in BCD                             |
| 1C      | 0022       | V <sub>DD</sub> MAX for Erase/Write; Bit7–Bit4 Volts in BCD, Bit3–Bit0 100mV in BCD                             |
| 1D      | 00B4       | V <sub>PP</sub> MIN for Erase/Write; Bit7–Bit4 Volts in Hex, Bit3–Bit0 100mV in BCD, 0000 = V <sub>PP</sub> pin |
| 1E      | 00C6       | V <sub>PP</sub> MAX for Erase/Write; Bit7–Bit4 Volts in Hex, Bit3–Bit0 100mV in BCD, 0000 = V <sub>PP</sub> pin |
| 1F      | 0003       | Typical timeout for single byte/word program, 2 <sup>n</sup> μs, 0000 = not supported                           |
| 20      | 0000       | Typical timeout for maximum size multiple byte/word program, 2 <sup>n</sup> μs, 0000 = not supported            |
| 21      | 0009       | Typical timeout for individual block erase, 2 <sup>n</sup> ms, 0000 = not supported                             |
| 22      | 0000       | Typical timeout for full chip erase, 2 <sup>n</sup> ms, 0000 = not supported                                    |
| 23      | 000C       | Maximum timeout for single byte/word program, 2 <sup>n</sup> μs, 0000 = not supported                           |
| 24      | 0000       | Maximum timeout for maximum size multiple byte/word program, 2 <sup>n</sup> μs, 0000 = not supported            |
| 25      | 0003       | Maximum timeout for individual block erase, 2 <sup>n</sup> ms, 0000 = not supported                             |
| 26      | 0000       | Maximum timeout for full chip erase, 2 <sup>n</sup> ms, 0000 = not supported                                    |
| 27      | 0017       | Device size, 2 <sup>n</sup> bytes                                                                               |
| 28      | 0001       | Bus interface, x8 = 0, x16 = 1, x8/x16 = 2                                                                      |
| 29      | 0000       | Flash device interface description, 0000 = async                                                                |
| 2A, 2B  | 0000, 0000 | Maximum number of bytes in multibyte program or page, 2 <sup>n</sup>                                            |
| 2C      | 0003       | Number of erase block regions within device (4K words and 32K words)                                            |
| 2D, 2E  | 005F–0000  | Top boot block device erase block region information 1, 96 blocks ...                                           |
|         | 0007–0000  | Bottom boot block device erase block region information 1, 8 blocks ...                                         |
| 2F, 30  | 0000–0001  | Top boot block device .....of 64KB                                                                              |
|         | 0020–0000  | Bottom boot block device .....of 8KB                                                                            |
| 31, 32  | 001E–0000  | 31 blocks of                                                                                                    |
| 33, 34  | 0000–0001  | .....64KB                                                                                                       |
| 35, 36  | 0007–0000  | Top boot block device ...8 blocks of                                                                            |
|         | 005F–0000  | Bottom boot block device ...96 blocks of                                                                        |

(continued on next page)


**Table 15 (continued)  
CFI**

| OFFSET               | DATA                         | DESCRIPTION                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|----------------------|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 37, 38               | 0020–0000                    | Top boot block device .....of 8KB                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|                      | 0000–0001                    | Bottom boot block device .....of 64KB                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 39, 3A               | 0050, 0052                   | "PR"                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| 3B                   | 0049                         | "I"                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 3C                   | 0030                         | Major version number, ASCII                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 3D                   | 0031                         | Minor version number, ASCII                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 3E<br>3F<br>40<br>41 | 00E6<br>0003<br>0000<br>0000 | Optional Feature and Command Support<br>Bit 0 Chip erase supported = no = 0<br>Bit 1 Suspend erase supported = yes = 1<br>Bit 2 Suspend program supported = yes = 1<br>Bit 3 Chip lock/unlock supported = no = 0<br>Bit 4 Queued erase supported = no = 0<br>Bit 5 Instant individual block locking supported = yes = 1<br>Bit 6 Protection bits supported = yes = 1<br>Bit 7 Page mode read supported = yes = 1<br>Bit 8 Synchronous read supported = yes = 1<br>Bit 9 Simultaneous operation supported = yes = 1 |
| 42                   | 0001                         | Program supported after erase suspend = yes                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 43, 44               | 0003, 0000                   | Bit 0 block lock status active = yes, Bit 1 block lock down active = yes                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| 45                   | 0018                         | V <sub>CC</sub> supply optimum, 00 = not supported, D7–D4 BCD V, D3–D0 100mV                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 46                   | 00C0                         | V <sub>PP</sub> supply optimum, 00 = not supported, D7–D4 Hex V, D3–D0 100mV                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 47                   | 0001                         | Number of protection register files in JEDEC ID space                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 48, 49               | 0080, 0000                   | Lock bytes LOW address, lock bytes HIGH address                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| 4A, 4B               | 0003, 0003                   | 2 <sup>n</sup> factory programmed bytes, 2 <sup>n</sup> user programmable bytes                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| 4C                   | 0003                         | Background Operation<br>0000 = Not used<br>0001 = 4% block split<br>0002 = 12% block split<br>0003 = 25% block split<br>0004 = 50% block split                                                                                                                                                                                                                                                                                                                                                                     |
| 4D                   | 0072                         | Burst Mode Type<br>0000 = No burst mode<br>00x1 = 4 words MAX<br>00x2 = 8 words MAX<br>00x3 = 16 words MAX<br>001x = Linear burst, and/or<br>002x = Interleaved burst, and/or<br>003x = Continuous burst                                                                                                                                                                                                                                                                                                           |
| 4E                   | 0002                         | Page Mode Type<br>0000 = No page mode<br>0001 = 4-word page<br>0002 = 8-word page<br>0003 = 16-word page<br>0004 = 32-word page                                                                                                                                                                                                                                                                                                                                                                                    |
| 4F                   | 0000                         | Not used                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |

**59- BALL FBGA**


- NOTE:**
1. All dimensions in millimeters.
  2. Package width and length do not include mold protrusion; allowable mold protrusion is 0.25mm per side.

**DATA SHEET DESIGNATION**

Advance: This data sheet contains initial descriptions of products still under development.



8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900

E-mail: [prodmktg@micron.com](mailto:prodmktg@micron.com), Internet: <http://www.micron.com>, Customer Comment Line: 800-932-4992

Micron and the M logo are registered trademarks and the Micron logo is a trademark of Micron Technology, Inc.



## REVISION HISTORY

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |      |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| Rev. 3, ADVANCE.....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | 8/02 |
| <ul style="list-style-type: none"> <li>Clarified device specific V<sub>CC</sub>, V<sub>CCQ</sub> and V<sub>PP</sub> voltages.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |      |
| Rev. 2, ADVANCE.....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | 7/02 |
| <ul style="list-style-type: none"> <li>Changed low power consumption voltage from 1.90V to 2.20V</li> <li>Corrected top boot block device address range for blocks 123 and 125</li> <li>Corrected Output Disable row in Bus Operations table</li> <li>Updated Status Register section</li> <li>Updated command descriptions</li> <li>Updated flowcharts</li> <li>Updated Read-While-Write/EraseConcurrency section</li> <li>Updated Read Configuration Register table</li> <li>Corrected addresses in Bank Boundary Wrapping figure</li> <li>Updated timing diagrams</li> <li>Corrected CFI table</li> </ul> |      |
| Original document, ADVANCE, Rev. 1 .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | 3/02 |