

# Parallel NOR Flash Embedded Memory

## MT28EW256ABA

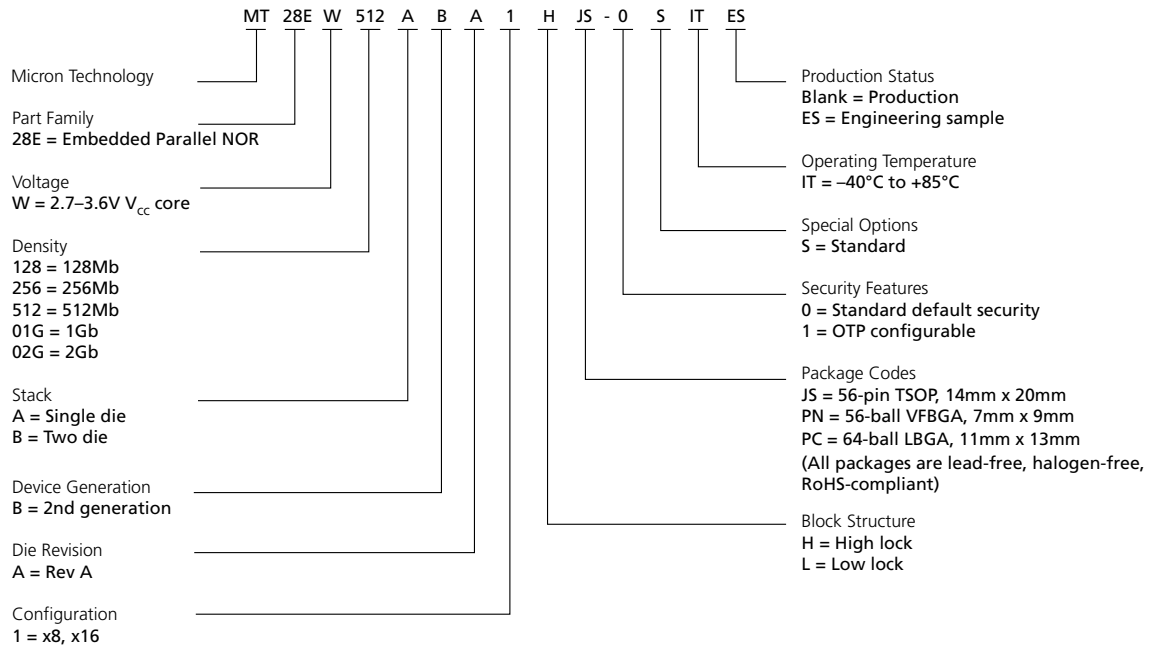
### Features

- Single-level cell (SLC) process technology
- Density: 256Mb
- Supply voltage
  - $V_{CC} = 2.7\text{--}3.6\text{V}$  (program, erase, read)
  - $V_{CCQ} = 1.65 - V_{CC}$  (I/O buffers)
- Asynchronous random/page read
  - Page size: 16 words or 32 bytes
  - Page access: 20ns
  - Random access: 70ns ( $V_{CC} = V_{CCQ} = 2.7\text{--}3.6\text{V}$ )
  - Random access: 75ns ( $V_{CCQ} = 1.65\text{--}V_{CC}$ )
- Buffer program (512-word program buffer)
  - 2.0 MB/s (TYP) when using full buffer program
  - 2.5 MB/s (TYP) when using accelerated buffer program ( $V_{HH}$ )
- Word/Byte program: 25us per word (TYP)
- Block erase (128KB): 0.2s (TYP)
- Memory organization
  - Uniform blocks: 128KB or 64KW each
  - x8/x16 data bus
- Program/erase suspend and resume capability
  - Read from another block during a PROGRAM SUSPEND operation
  - Read or program another block during an ERASE SUSPEND operation
- Unlock bypass, block erase, chip erase, and write to buffer capability
- BLANK CHECK operation to verify an erased block
- CYCLIC REDUNDANCY CHECK (CRC) operation to verify a program pattern
- $V_{PP}/WP\#$  protection
  - Protects first or last block regardless of block protection settings
- Software protection
  - Volatile protection
  - Nonvolatile protection
  - Password protection
- Extended memory block
  - 128-word (256-byte) block for permanent, secure identification
  - Programmed or locked at the factory or by the customer
- JESD47-compliant
  - 100,000 (minimum) ERASE cycles per block
  - Data retention: 20 years (TYP)
- Package
  - 56-pin TSOP, 14 x 20mm (JS)
  - 64-ball LBGA, 11 x 13mm (PC)
  - 56-ball VFBGA, 7 x 9mm (PN)
- RoHS-compliant, halogen-free packaging
- Operating temperature
  - Ambient:  $-40^{\circ}\text{C}$  to  $+85^{\circ}\text{C}$

### Part Numbering Information

For available options, such as packages or high/low protection, or for further information, contact your Micron sales representative. Part numbers can be verified at [www.micron.com](http://www.micron.com). Feature and specification comparison by device type is available at [www.micron.com/products](http://www.micron.com/products). Contact the factory for devices not found.

**Figure 1: Part Number Chart**





## Contents

|  |    |
|--|----|
| Important Notes and Warnings .....                       | 7  |
| General Description .....                                | 8  |
| Signal Assignments .....                                 | 9  |
| Signal Descriptions .....                                | 12 |
| Memory Organization .....                                | 14 |
| Memory Configuration .....                               | 14 |
| Memory Map .....   | 14 |
| Bus Operations .....                                     | 15 |
| Read .....   | 15 |
| Write .....  | 15 |
| Standby .....  | 16 |
| Output Disable .....                                     | 16 |
| Reset .....  | 16 |
| Registers .....  | 17 |
| Data Polling Register .....                              | 17 |
| Lock Register .....                                      | 22 |
| Standard Command Definitions – Address-Data Cycles ..... | 24 |
| READ and AUTO SELECT Operations .....                    | 27 |
| READ/RESET Command .....                                 | 27 |
| READ CFI Command .....                                   | 27 |
| AUTO SELECT Command .....                                | 27 |
| Read Electronic Signature .....                          | 28 |
| Cyclic Redundancy Check Operation .....                  | 29 |
| CYCLIC REDUNDANCY CHECK Command .....                    | 29 |
| Cyclic Redundancy Check Operation Command Sequence ..... | 29 |
| Bypass Operations .....                                  | 32 |
| UNLOCK BYPASS Command .....                              | 32 |
| UNLOCK BYPASS RESET Command .....                        | 32 |
| Program Operations .....                                 | 33 |
| PROGRAM Command .....                                    | 33 |
| UNLOCK BYPASS PROGRAM Command .....                      | 33 |
| WRITE TO BUFFER PROGRAM Command .....                    | 33 |
| UNLOCK BYPASS WRITE TO BUFFER PROGRAM Command .....      | 36 |
| WRITE TO BUFFER PROGRAM CONFIRM Command .....            | 36 |
| BUFFERED PROGRAM ABORT AND RESET Command .....           | 36 |
| PROGRAM SUSPEND Command .....                            | 36 |
| PROGRAM RESUME Command .....                             | 37 |
| ACCELERATED BUFFERED PROGRAM Operations .....            | 37 |
| Erase Operations .....                                   | 38 |
| CHIP ERASE Command .....                                 | 38 |
| UNLOCK BYPASS CHIP ERASE Command .....                   | 38 |
| BLOCK ERASE Command .....                                | 38 |
| UNLOCK BYPASS BLOCK ERASE Command .....                  | 39 |
| ERASE SUSPEND Command .....                              | 39 |
| ERASE RESUME Command .....                               | 40 |
| ACCELERATED CHIP ERASE Operations .....                  | 40 |
| BLANK CHECK Operation .....                              | 41 |
| BLANK CHECK Commands .....                               | 41 |
| Device Protection .....                                  | 42 |
| Hardware Protection .....                                | 42 |



|  |    |
|--|----|
| Software Protection .....  | 42 |
| Volatile Protection Mode .....                                   | 43 |
| Nonvolatile Protection Mode .....                                | 43 |
| Password Protection Mode .....                                   | 43 |
| Block Protection Command Definitions – Address-Data Cycles ..... | 46 |
| Protection Operations .....                                      | 49 |
| LOCK REGISTER Commands .....                                     | 49 |
| PASSWORD PROTECTION Commands .....                               | 49 |
| NONVOLATILE PROTECTION Commands .....                            | 49 |
| NONVOLATILE PROTECTION BIT LOCK BIT Commands .....               | 50 |
| VOLATILE PROTECTION Commands .....                               | 51 |
| EXTENDED MEMORY BLOCK Commands .....                             | 51 |
| EXIT PROTECTION Command .....                                    | 52 |
| Common Flash Interface .....                                     | 53 |
| Power-Up and Reset Characteristics .....                         | 57 |
| Absolute Ratings and Operating Conditions .....                  | 59 |
| DC Characteristics .....   | 61 |
| Read AC Characteristics .....                                    | 63 |
| Write AC Characteristics .....                                   | 67 |
| Data Polling/Toggle AC Characteristics .....                     | 74 |
| Program/Erase Characteristics .....                              | 76 |
| Package Dimensions .....   | 78 |
| Revision History .....   | 81 |
| Rev. G – 05/18 .....   | 81 |
| Rev. F – 11/16 .....   | 81 |
| Rev. E – 4/16 .....  | 81 |
| Rev. D – 5/15 .....  | 81 |
| Rev. C – 4/15 .....  | 81 |
| Rev. B – 1/15 .....  | 81 |
| Rev. A – 6/14 .....  | 81 |

## List of Figures

|   |    |
|---|----|
| Figure 1: Part Number Chart .....   | 2  |
| Figure 2: Logic Diagram .....   | 8  |
| Figure 3: 56-Pin TSOP (Top View) .....                                    | 9  |
| Figure 4: 64-Ball LBGA (Top View – Balls Down) .....                      | 10 |
| Figure 5: 56-Ball VFBGA (Top View – Balls Down) .....                     | 11 |
| Figure 6: Data Polling Flowchart .....                                    | 19 |
| Figure 7: Toggle Bit Flowchart .....                                      | 20 |
| Figure 8: Data Polling/Toggle Bit Flowchart .....                         | 21 |
| Figure 9: Lock Register Program Flowchart .....                           | 23 |
| Figure 10: Boundary Condition of Program Buffer Size .....                | 34 |
| Figure 11: WRITE TO BUFFER PROGRAM Flowchart .....                        | 35 |
| Figure 12: Software Protection Scheme .....                               | 44 |
| Figure 13: Set/Clear Nonvolatile Protection Bit Algorithm Flowchart ..... | 50 |
| Figure 14: Power-Up Timing .....  | 57 |
| Figure 15: Reset AC Timing – No PROGRAM/ERASE Operation in Progress ..... | 58 |
| Figure 16: Reset AC Timing During PROGRAM/ERASE Operation .....           | 58 |
| Figure 17: AC Measurement Load Circuit .....                              | 60 |
| Figure 18: AC Measurement I/O Waveform .....                              | 60 |
| Figure 19: Random Read AC Timing (8-Bit Mode) .....                       | 64 |
| Figure 20: Random Read AC Timing (16-Bit Mode) .....                      | 65 |
| Figure 21: BYTE# Transition Read AC Timing .....                          | 65 |
| Figure 22: Page Read AC Timing (16-Bit Mode) .....                        | 66 |
| Figure 23: WE#-Controlled Program AC Timing (8-Bit Mode) .....            | 68 |
| Figure 24: WE#-Controlled Program AC Timing (16-Bit Mode) .....           | 69 |
| Figure 25: CE#-Controlled Program AC Timing (8-Bit Mode) .....            | 71 |
| Figure 26: CE#-Controlled Program AC Timing (16-Bit Mode) .....           | 72 |
| Figure 27: Chip/Block Erase AC Timing (16-Bit Mode) .....                 | 73 |
| Figure 28: Accelerated Program AC Timing .....                            | 73 |
| Figure 29: Data Polling AC Timing .....                                   | 74 |
| Figure 30: Toggle/Alternative Toggle Bit Polling AC Timing .....          | 75 |
| Figure 31: 56-Pin TSOP – 14mm x 20mm (Package Code: JS) .....             | 78 |
| Figure 32: 64-Ball LBGA – 11mm x 13mm (Package Code: PC) .....            | 79 |
| Figure 33: 56-Ball VFBGA – 7mm x 9mm (Package Code: PN) .....             | 80 |



## List of Tables

|  |    |
|--|----|
| Table 1: Signal Descriptions .....   | 12 |
| Table 2: Blocks[2047:0] .....  | 14 |
| Table 3: Bus Operations .....  | 15 |
| Table 4: Data Polling Register Bit Definitions .....   | 17 |
| Table 5: Operations and Corresponding Bit Settings .....                                     | 18 |
| Table 6: Lock Register Bit Definitions .....   | 22 |
| Table 7: Standard Command Definitions – Address-Data Cycles, 8-Bit and 16-Bit .....          | 24 |
| Table 8: Block Protection .....  | 28 |
| Table 9: Read Electronic Signature – 256Mb .....   | 28 |
| Table 10: Command Sequence – Range of Blocks .....   | 29 |
| Table 11: Command Sequence – Entire Chip .....   | 31 |
| Table 12: ACCELERATED PROGRAM Requirements and Recommendations .....                         | 37 |
| Table 13: ACCELERATED CHIP ERASE Requirements and Recommendations .....                      | 41 |
| Table 14: V <sub>pp</sub> /WP# Functions .....   | 42 |
| Table 15: Block Protection Status .....  | 45 |
| Table 16: Block Protection Command Definitions – Address-Data Cycles, 8-Bit and 16-Bit ..... | 46 |
| Table 17: Extended Memory Block Address and Data .....                                       | 51 |
| Table 18: Query Structure Overview .....   | 53 |
| Table 19: CFI Query Identification String .....  | 53 |
| Table 20: CFI Query System Interface Information .....                                       | 54 |
| Table 21: Device Geometry Definition .....   | 54 |
| Table 22: Primary Algorithm-Specific Extended Query Table .....                              | 55 |
| Table 23: Power-Up Specifications .....  | 57 |
| Table 24: Reset AC Specifications .....  | 58 |
| Table 25: Absolute Maximum/Minimum Ratings .....   | 59 |
| Table 26: Operating Conditions .....   | 59 |
| Table 27: Input/Output Capacitance .....   | 60 |
| Table 28: DC Current Characteristics .....   | 61 |
| Table 29: DC Voltage Characteristics .....   | 62 |
| Table 30: Read AC Characteristics – V <sub>CC</sub> =V <sub>CCQ</sub> = 2.7-3.6V .....       | 63 |
| Table 31: Read AC Characteristics – V <sub>CCQ</sub> = 1.65V-V <sub>CC</sub> .....           | 63 |
| Table 32: WE#-Controlled Write AC Characteristics .....                                      | 67 |
| Table 33: CE#-Controlled Write AC Characteristics .....                                      | 70 |
| Table 34: Data Polling/Toggle AC Characteristics .....                                       | 74 |
| Table 35: Program/Erase Characteristics .....  | 76 |

## Important Notes and Warnings

Micron Technology, Inc. ("Micron") reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions. This document supersedes and replaces all information supplied prior to the publication hereof. You may not rely on any information set forth in this document if you obtain the product described herein from any unauthorized distributor or other source not authorized by Micron.

**Automotive Applications.** Products are not designed or intended for use in automotive applications unless specifically designated by Micron as automotive-grade by their respective data sheets. Distributor and customer/distributor shall assume the sole risk and liability for and shall indemnify and hold Micron harmless against all claims, costs, damages, and expenses and reasonable attorneys' fees arising out of, directly or indirectly, any claim of product liability, personal injury, death, or property damage resulting directly or indirectly from any use of non-automotive-grade products in automotive applications. Customer/distributor shall ensure that the terms and conditions of sale between customer/distributor and any customer of distributor/customer (1) state that Micron products are not designed or intended for use in automotive applications unless specifically designated by Micron as automotive-grade by their respective data sheets and (2) require such customer of distributor/customer to indemnify and hold Micron harmless against all claims, costs, damages, and expenses and reasonable attorneys' fees arising out of, directly or indirectly, any claim of product liability, personal injury, death, or property damage resulting from any use of non-automotive-grade products in automotive applications.

**Critical Applications.** Products are not authorized for use in applications in which failure of the Micron component could result, directly or indirectly in death, personal injury, or severe property or environmental damage ("Critical Applications"). Customer must protect against death, personal injury, and severe property and environmental damage by incorporating safety design measures into customer's applications to ensure that failure of the Micron component will not result in such harms. Should customer or distributor purchase, use, or sell any Micron component for any critical application, customer and distributor shall indemnify and hold harmless Micron and its subsidiaries, subcontractors, and affiliates and the directors, officers, and employees of each against all claims, costs, damages, and expenses and reasonable attorneys' fees arising out of, directly or indirectly, any claim of product liability, personal injury, or death arising in any way out of such critical application, whether or not Micron or its subsidiaries, subcontractors, or affiliates were negligent in the design, manufacture, or warning of the Micron product.

**Customer Responsibility.** Customers are responsible for the design, manufacture, and operation of their systems, applications, and products using Micron products. ALL SEMICONDUCTOR PRODUCTS HAVE INHERENT FAILURE RATES AND LIMITED USEFUL LIVES. IT IS THE CUSTOMER'S SOLE RESPONSIBILITY TO DETERMINE WHETHER THE MICRON PRODUCT IS SUITABLE AND FIT FOR THE CUSTOMER'S SYSTEM, APPLICATION, OR PRODUCT. Customers must ensure that adequate design, manufacturing, and operating safeguards are included in customer's applications and products to eliminate the risk that personal injury, death, or severe property or environmental damages will result from failure of any semiconductor component.

**Limited Warranty.** In no event shall Micron be liable for any indirect, incidental, punitive, special or consequential damages (including without limitation lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort, warranty, breach of contract or other legal theory, unless explicitly stated in a written agreement executed by Micron's duly authorized representative.

## General Description

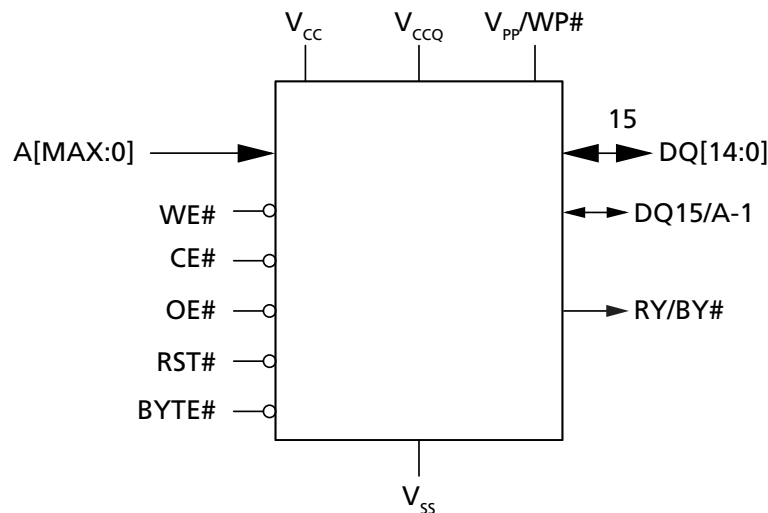
The device is an asynchronous, uniform block, parallel NOR Flash memory device. READ, ERASE, and PROGRAM operations are performed using a single low-voltage supply. Upon power-up, the device defaults to read array mode.

The main memory array is divided into uniform blocks that can be erased independently so that valid data can be preserved while old data is purged. PROGRAM and ERASE commands are written to the command interface of the memory. An on-chip program/erase controller simplifies the process of programming or erasing the memory by taking care of all special operations required to update the memory contents. The end of a PROGRAM or ERASE operation can be detected and any error condition can be identified. The command set required to control the device is consistent with JEDEC standards.

CE#, OE#, and WE# control the bus operation of the device and enable a simple connection to most microprocessors, often without additional logic.

The device supports asynchronous random read and page read from all blocks of the array. It also features an internal program buffer that improves throughput by programming 512 words via one command sequence. A 128-word extended memory block overlaps addresses with array block 0. Users can program this additional space and then protect it to permanently secure the contents. The device also features different levels of hardware and software protection to secure blocks from unwanted modification.

**Figure 2: Logic Diagram**





## Signal Assignments

Figure 3: 56-Pin TSOP (Top View)



- Notes:
1. A-1 is the least significant address bit in x8 mode.
  2. A23 is valid for 256Mb and above; otherwise, it is RFU.
  3. A24 is valid for 512Mb and above; otherwise, it is RFU.
  4. A25 is valid for 1Gb and above; otherwise, it is RFU.

**Figure 4: 64-Ball LBGGA (Top View – Balls Down)**



- Notes:
1. A-1 is the least significant address bit in x8 mode.
  2. A23 is valid for 256Mb and above; otherwise, it is RFU.
  3. A24 is valid for 512Mb and above; otherwise, it is RFU.
  4. A25 is valid for 1Gb and above; otherwise, it is RFU.

**Figure 5: 56-Ball VFBGA (Top View – Balls Down)**



- Notes:
1. A-1 is the least significant address bit in x8 mode.
  2. A23 is valid for 256Mb and above; otherwise, it is RFU.
  3. A24 is valid for 512Mb and above; otherwise, it is RFU.
  4. A25 is valid for 1Gb and above; otherwise, it is RFU.

## Signal Descriptions

The signal description table below is a comprehensive list of signals for this device family. All signals listed may not be supported on this device. See Signal Assignments for information specific to this device.

**Table 1: Signal Descriptions**

| Name                 | Type   | Description   |
|----------------------|--------|---|
| A[MAX:0]             | Input  | <b>Address:</b> Selects array cells to access during READ operations. Controls commands sent to the program/erase controller command interface during WRITE operations.   |
| CE#                  | Input  | <b>Chip enable:</b> Activates the device, enabling READ and WRITE operations. When CE# is HIGH, the device goes to standby and data outputs are High-Z.   |
| OE#                  | Input  | <b>Output enable:</b> Active LOW input. OE# LOW enables data output buffers during READ cycles. When OE# is HIGH, data outputs are High-Z.  |
| WE#                  | Input  | <b>Write enable:</b> Controls WRITE operations to the device. Address is latched on the falling edge of WE# and data is latched on the rising edge.   |
| V <sub>pp</sub> /WP# | Input  | <b>V<sub>pp</sub>/Write Protect:</b> Provides WRITE PROTECT and V <sub>HH</sub> functionality, which protects the lowest or highest block and enables the device to enter unlock bypass mode.   |
| BYTE#                | Input  | <b>Byte/word organization select:</b> Selects x8 or x16 bus mode. When BYTE# is LOW, the device is in x8 mode and when HIGH, the device is in x16 mode. Under byte configuration, BYTE# should not be toggled during any WRITE operation.<br>Caution: This pin cannot be floated.   |
| RST#                 | Input  | <b>Reset:</b> When held LOW for at least <sup>t</sup> PLPH, applies a hardware reset to the device control logic and places it in standby. After RST# goes HIGH, the device is ready for READ and WRITE operations; that is, after <sup>t</sup> PHEL or <sup>t</sup> PHWL, whichever occurs last.   |
| DQ[7:0]              | I/O    | <b>Data I/O:</b> During a READ operation, outputs data stored at the selected address. During a WRITE operation, represents the commands sent to the command interface.   |
| DQ[14:8]             | I/O    | <b>Data I/O:</b> During a READ operation when BYTE# is HIGH, outputs data stored at the selected address. When BYTE# is LOW, these pins are High-Z and not used. During a WRITE operation, these bits are not used. When reading the data polling register, these bits should be ignored.   |
| DQ15/A-1             | I/O    | <b>Data I/O or address input:</b> When device is in x16 bus mode, this pin behaves as data I/O, together with DQ[14:8]. When device is in x8 bus mode, this pin behaves as the least significant bit of the address.<br>Unless explicitly stated elsewhere, DQ15 = data I/O (x16 mode) and A-1 = address input (x8 mode).   |
| RY/BY#               | Output | <b>Ready busy:</b> Open-drain output used to identify when the device is performing a PROGRAM or ERASE operation. During a PROGRAM or ERASE operation, RY/BY# is LOW. During read, auto select, and erase suspend modes, RY/BY# is High-Z.<br>Enables RY/BY# pins from several devices to be connected to a single pull-up resistor which is connected to V <sub>CCQ</sub> . Therefore, RYBY# LOW indicates when one or more of the devices are busy. A 10K Ohm or bigger resistor is recommended as pull-up resistor to achieve 0.1V V <sub>OL</sub> . |
| V <sub>CC</sub>      | Supply | <b>Supply voltage:</b> Provides power supply for READ, PROGRAM, and ERASE operations. When V <sub>CC</sub> ≤ V <sub>LKO</sub> , the device is disabled, any PROGRAM or ERASE operation is aborted, and any altered content will be invalid.<br>Capacitors of 0.1μF and 0.01μF should be connected between V <sub>CC</sub> and V <sub>SS</sub> to decouple the current surges from the power supply. The PCB track widths must be sufficient to carry the currents required during PROGRAM and ERASE operations.   |

**Table 1: Signal Descriptions (Continued)**

| <b>Name</b>      | <b>Type</b> | <b>Description</b>  |
|------------------|-------------|---|
| V <sub>CCQ</sub> | Supply      | <b>I/O supply voltage:</b> Provides power supply to the I/O pins and enables all outputs to be powered independently from V <sub>CC</sub> . Capacitors of 0.1μF and 0.01μF should be connected between V <sub>CCQ</sub> and V <sub>SS</sub> to decouple the current surges from the power supply. |
| V <sub>SS</sub>  | Supply      | <b>Ground:</b> All V <sub>SS</sub> pins must be connected to system ground.   |
| RFU              | —           | <b>Reserved for future use:</b> Reserved by Micron for future device functionality and enhancement. Recommend that these be left floating. May be connected internally, but external connections will not affect operation.   |
| DNU              | —           | <b>Do not use:</b> Do not connect to any other signal or power supply; must be left floating.   |
| NC               | —           | <b>No connect:</b> No internal connection; can be driven or floated.  |



## Memory Organization

### Memory Configuration

The main memory array is divided into 128KB or 64KW uniform blocks.

### Memory Map

**Table 2: Blocks[2047:0]**

| Block | Block Size | Address Range (x8) |           | Block Size | Address Range (x16) |           |
|-------|------------|--------------------|-----------|------------|---------------------|-----------|
|       |            | Start              | End       |            | Start               | End       |
| 2047  | 128KB      | FFE 0000h          | FFF FFFFh | 64KW       | 7FF 0000h           | 7FF FFFFh |
| ⋮     |            | ⋮                  | ⋮         |            | ⋮                   | ⋮         |
| 1023  |            | 7FE 0000h          | 7FF FFFFh |            | 3FF 0000h           | 3FF FFFFh |
| ⋮     |            | ⋮                  | ⋮         |            | ⋮                   | ⋮         |
| 511   |            | 3FE 0000h          | 3FF FFFFh |            | 1FF 0000h           | 1FF FFFFh |
| ⋮     |            | ⋮                  | ⋮         |            | ⋮                   | ⋮         |
| 255   |            | 1FE 0000h          | 1FF FFFFh |            | 0FF 0000h           | 0FF FFFFh |
| ⋮     |            | ⋮                  | ⋮         |            | ⋮                   | ⋮         |
| 127   |            | 0FE 0000h          | 0FF FFFFh |            | 07F 0000h           | 07F FFFFh |
| ⋮     |            | ⋮                  | ⋮         |            | ⋮                   | ⋮         |
| 63    |            | 07E 0000h          | 07F FFFFh |            | 03F 0000h           | 03F FFFFh |
| ⋮     |            | ⋮                  | ⋮         |            | ⋮                   | ⋮         |
| 0     |            | 000 0000h          | 001 FFFFh |            | 000 0000h           | 000 FFFFh |

Note: 1. 128Mb device = Blocks 0–127; 256Mb device = Blocks 0–255; 512Mb device = Blocks 0–511; 1Gb device = Blocks 0–1023; 2Gb device = Blocks 0–2047.

## Bus Operations

**Table 3: Bus Operations**

Notes 1 and 2 apply to entire table

| Operation         | CE# | OE# | WE# | RST# | V <sub>pp</sub> /WP# | 8-Bit Mode                     |          |                         | 16-Bit Mode        |                         |
|-------------------|-----|-----|-----|------|----------------------|--------------------------------|----------|-------------------------|--------------------|-------------------------|
|                   |     |     |     |      |                      | A[ <b>MAX:0</b> ],<br>DQ15/A-1 | DQ[14:8] | DQ[7:0]                 | A[ <b>MAX:0</b> ]  | DQ15/A-1,<br>DQ[14:0]   |
| READ              | L   | L   | H   | H    | X                    | Address                        | High-Z   | Data output             | Address            | Data output             |
| WRITE             | L   | H   | L   | H    | H <sup>3</sup>       | Command<br>address             | High-Z   | Data input <sup>4</sup> | Command<br>address | Data input <sup>4</sup> |
| STANDBY           | H   | X   | X   | H    | X                    | X                              | High-Z   | High-Z                  | X                  | High-Z                  |
| OUTPUT<br>DISABLE | L   | H   | H   | H    | X                    | X                              | High-Z   | High-Z                  | X                  | High-Z                  |
| RESET             | X   | X   | X   | L    | X                    | X                              | High-Z   | High-Z                  | X                  | High-Z                  |

- Notes:
1. Typical glitches of less than 3ns on CE#, OE#, and WE# are ignored by the device and do not affect bus operations.
  2. H = Logic level HIGH (V<sub>IH</sub>); L = Logic level LOW (V<sub>IL</sub>); X = HIGH or LOW.
  3. If WP# is LOW, then the highest or the lowest block remains protected, depending on line item.
  4. Data input is required when issuing a command sequence or when performing data polling or block protection.

### Read

Bus READ operations read from the memory cells, registers, extended memory block, or CFI space. To accelerate the READ operation, the memory array can be read in page mode where data is internally read and stored in a page buffer.

Page size is 16 words (32 bytes) and is addressed by address inputs A[3:0] in x16 bus mode and A[3:0] plus DQ15/A-1 in x8 bus mode. The extended memory blocks and CFI area support page read mode.

A valid bus READ operation involves setting the desired address on the address inputs, taking CE# and OE# LOW, and holding WE# HIGH. The data I/Os will output the value. If CE# goes HIGH and returns LOW for a subsequent access, a random read access is performed and t<sub>ACC</sub> or t<sub>CE</sub> is required. (See AC Characteristics for details about when the output becomes valid).

### Write

Bus WRITE operations write to the command interface. A valid bus WRITE operation begins by setting the desired address on the address inputs. The address inputs are latched by the command interface on the falling edge of CE# or WE#, whichever occurs last. The data I/Os are latched by the command interface on the rising edge of CE# or WE#, whichever occurs first. OE# must remain HIGH during the entire bus WRITE operation (See AC Characteristics for timing requirement details).

## Standby

Driving CE# HIGH in read mode causes the device to enter standby and data I/Os to be High-Z (See DC Characteristics).

During PROGRAM or ERASE operations, the device will continue to use the program/erase supply current ( $I_{CC3}$ ) until the operation completes. The device cannot be placed into standby mode during a PROGRAM/ERASE operation.

## Output Disable

Data I/Os are High-Z when OE# is HIGH.

## Reset

During reset mode the device is deselected and the outputs are High-Z. The device is in reset mode when RST# is LOW. The power consumption is reduced to the standby level, independently from CE#, OE#, or WE# inputs.

When RST# is HIGH, a time of  $t_{PHEL}$  is required before a READ operation can access the device, and a delay of  $t_{PHWL}$  is required before a write sequence can be initiated. After this wake-up interval, normal operation is restored, the device defaults to read array mode, and the data polling register is reset.

If RST# is driven LOW during a PROGRAM/ERASE operation or any other operation that requires writing to the device, the operation will abort within  $t_{PLRH}$ , and memory contents at the aborted block or address are no longer valid.



## Registers

### Data Polling Register

**Table 4: Data Polling Register Bit Definitions**

Note 1 applies to entire table

| Bit | Name                       | Settings   | Description   | Notes   |
|-----|----------------------------|--|---|---------|
| DQ7 | Data polling bit           | 0 or 1, depending on operations                    | Monitors whether the program/erase controller has successfully completed its operation, or has responded to an ERASE SUSPEND operation.   | 2, 4    |
| DQ6 | Toggle bit                 | Toggles: 0 to 1; 1 to 0; and so on                 | Monitors whether the program, erase, or blank check controller has successfully completed its operations, or has responded to an ERASE SUSPEND operation. During a PROGRAM/ERASE/BLANK CHECK operation, DQ6 toggles from 0 to 1, 1 to 0, and so on, with each successive READ operation from any address. | 3, 4, 5 |
| DQ5 | Error bit                  | 0 = Success<br>1 = Failure                         | Identifies errors detected by the program/erase controller. DQ5 is set to 1 when a PROGRAM, BLOCK ERASE, or CHIP ERASE operation fails to write the correct data to the memory, or when a BLANK CHECK or CRC operation fails.   | 4, 6    |
| DQ3 | Erase timer bit            | 0 = Erase not in progress<br>1 = Erase in progress | Identifies the start of program/erase controller operation during a BLOCK ERASE command. Before the program/erase controller starts, this bit set to 0, and additional blocks to be erased can be written to the command interface.   | 4       |
| DQ2 | Alternative toggle bit     | Toggles: 0 to 1; 1 to 0; and so on                 | During CHIP ERASE, BLOCK ERASE, and ERASE SUSPEND operations, DQ2 toggles from 0 to 1, 1 to 0, and so on, with each successive READ operation from addresses within the blocks being erased.  | 3, 4    |
| DQ1 | Buffered program abort bit | 1 = Abort  | Indicates a BUFFER PROGRAM, EFI BLANK CHECK, or CRC operation abort. The BUFFERED PROGRAM ABORT and RESET command must be issued to return the device to read mode (see WRITE TO BUFFER PROGRAM command).   | –       |

- Notes:
1. The data polling register can be read during PROGRAM, ERASE, or ERASE SUSPEND operations; the READ operation outputs data on DQ[7:0].
  2. For a PROGRAM operation in progress, DQ7 outputs the complement of the bit being programmed. For a READ operation from the address previously programmed successfully, DQ7 outputs existing DQ7 data. For a READ operation from addresses with blocks to be erased while an ERASE SUSPEND operation is in progress, DQ7 outputs 0; upon successful completion of the ERASE SUSPEND operation, DQ7 outputs 1. For an ERASE operation in progress, DQ7 outputs 0; upon ERASE operation's successful completion, DQ7 outputs 1. During a BUFFER PROGRAM operation, the data polling bit is valid only for the last word being programmed in the write buffer.
  3. After successful completion of a PROGRAM, ERASE, or BLANK CHECK operation, the device returns to read mode.
  4. During erase suspend mode, READ operations to addresses within blocks not being erased output memory array data as if in read mode. A protected block is treated the same as a block not being erased. See the Toggle Flowchart for more information.

5. During erase suspend mode, DQ6 toggles when addressing a cell within a block being erased. The toggling stops when the program/erase controller has suspended the ERASE operation. See the Toggle Flowchart for more information.
6. When DQ5 is set to 1, a READ/RESET (F0h) command must be issued before any subsequent command.

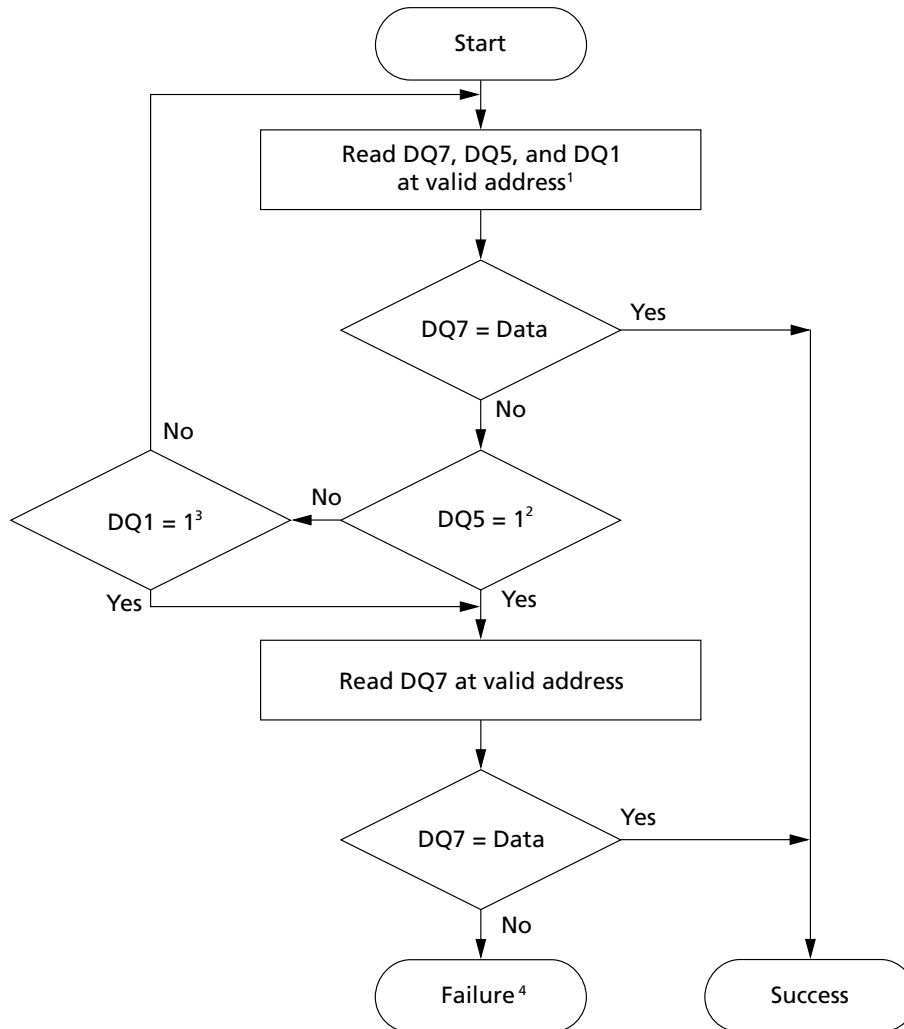
**Table 5: Operations and Corresponding Bit Settings**

Note 1 applies to entire table

| Operation                    | Address              | DQ7  | DQ6       | DQ5 | DQ3 | DQ2       | DQ1 | RY/BY# | Notes |
|------------------------------|----------------------|--|-----------|-----|-----|-----------|-----|--------|-------|
| PROGRAM                      | Any address          | DQ7#   | Toggle    | 0   | –   | –         | 0   | 0      | 2     |
| EFI BLANK CHECK              | Any address          | 1  | Toggle    | 0   | –   | –         | 0   | 0      | 3     |
| CRC range of blocks          | Any address          | 1  | Toggle    | 0   | –   | –         | 0   | 0      |       |
| CRC chip                     | Any address          | DQ7#   | Toggle    | 0   | –   | –         | 0   | 0      | 4     |
| CHIP ERASE                   | Any address          | 0  | Toggle    | 0   | 1   | Toggle    | –   | 0      | –     |
| BLOCK ERASE before time-out  | Erasing block        | 0  | Toggle    | 0   | 0   | Toggle    | –   | 0      | –     |
|                              | Non-erasing block    | 0  | Toggle    | 0   | 0   | No toggle | –   | 0      | –     |
| BLOCK ERASE                  | Erasing block        | 0  | Toggle    | 0   | 1   | Toggle    | –   | 0      | –     |
|                              | Non-erasing block    | 0  | Toggle    | 0   | 1   | No toggle | –   | 0      | –     |
| PROGRAM SUSPEND              | Programming block    | Invalid operation                            |           |     |     |           |     | High-Z | –     |
|                              | Nonprogramming block | Outputs memory array data as if in read mode |           |     |     |           |     | High-Z | –     |
| ERASE SUSPEND                | Erasing block        | 1  | No Toggle | 0   | –   | Toggle    | –   | High-Z | –     |
|                              | Non-erasing block    | Outputs memory array data as if in read mode |           |     |     |           |     | High-Z | –     |
| PROGRAM during ERASE SUSPEND | Erasing block        | DQ7#   | Toggle    | 0   | –   | Toggle    | –   | 0      | 2     |
|                              | Non-erasing block    | DQ7#   | Toggle    | 0   | –   | No Toggle | –   | 0      | 2     |
| BUFFERED PROGRAM ABORT       | Any address          | DQ7#   | Toggle    | 0   | –   | –         | 1   | High-Z | –     |
| PROGRAM Error                | Any address          | DQ7#   | Toggle    | 1   | –   | –         | –   | High-Z | 2     |
| ERASE Error                  | Any address          | 0  | Toggle    | 1   | 1   | Toggle    | –   | High-Z | –     |
| EFI BLANK CHECK Error        | Any address          | 0  | Toggle    | 1   | 1   | Toggle    | –   | High-Z | –     |
| CRC range of blocks error    | Any address          | 1  | Toggle    | 1   | –   | –         | –   | High-Z | –     |
| CRC chip error               | Any address          | DQ7#   | Toggle    | 1   | –   | –         | –   | High-Z | 4     |

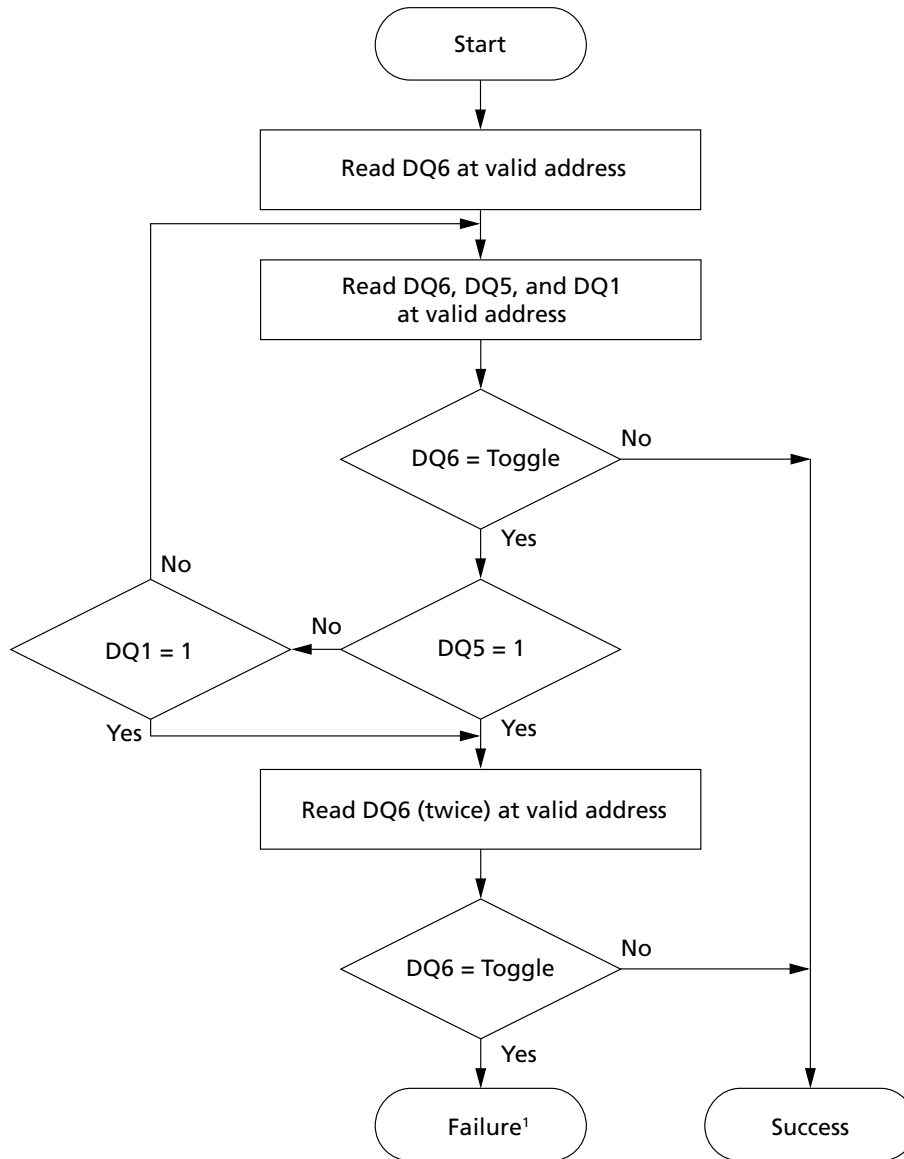
- Notes:
1. Unspecified data bits should be ignored.
  2. DQ7# for buffer program is related to the last address location loaded.
  3. EFI = enhanced Flash interface.
  4. DQ7# is the reverse DQ7 of the last word or byte loaded before CRC chip confirm command cycle.

Figure 6: Data Polling Flowchart



- Notes:
1. Valid address is the last address being programmed or an address within the block being erased.
  2. Failure results: DQ5 = 1 indicates an operation error. A READ/RESET (F0h) command must be issued before any subsequent command.
  3. Failure results: DQ1 = 1 indicates a WRITE TO BUFFER PROGRAM ABORT operation. A full three-cycle RESET (AAh/55h/F0h) command sequence must be used to reset the aborted device.
  4. The data polling process does not support the BLANK CHECK operation. The process represented in the Toggle Bit Flowchart figure can provide information on the BLANK CHECK operation.

Figure 7: Toggle Bit Flowchart



- Notes:
1. Failure results: DQ5 = 1 indicates an operation error; DQ1 = 1 indicates a WRITE TO BUFFER PROGRAM ABORT operation.
  2. The toggle bit process supports the BLANK CHECK operation.

Figure 8: Data Polling/Toggle Bit Flowchart



## Lock Register

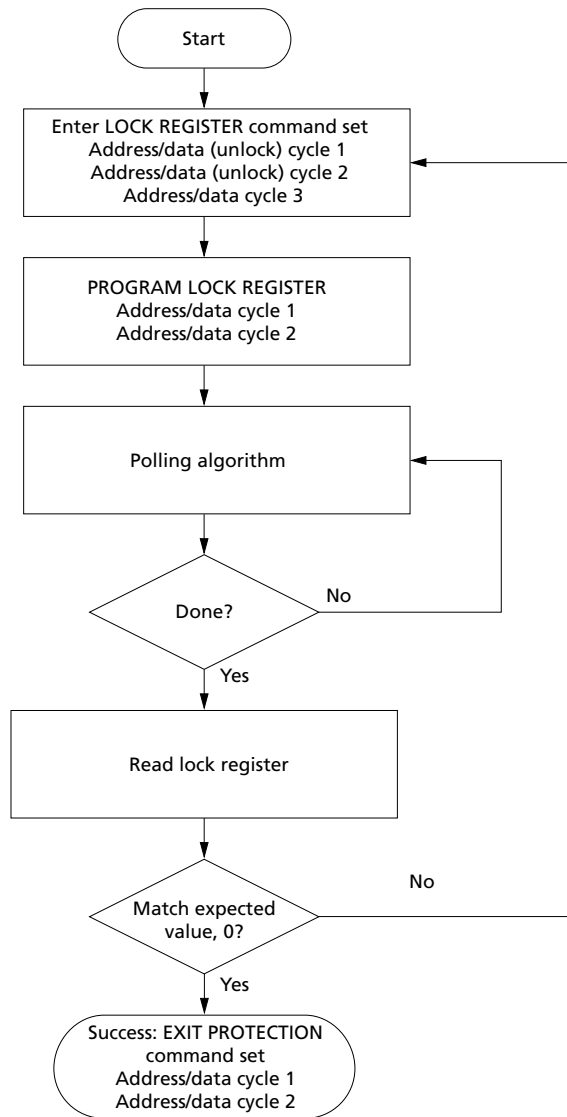
**Table 6: Lock Register Bit Definitions**

Note 1 applies to entire table

| Bit | Name                                 | Settings  | Description   | Notes |
|-----|--------------------------------------|---|---|-------|
| DQ2 | Password protection mode lock bit    | 0 = Password protection mode enabled<br>1 = Password protection mode disabled (Default)   | Places the device permanently in password protection mode.  | 2     |
| DQ1 | Nonvolatile protection mode lock bit | 0 = Nonvolatile protection mode enabled with password protection mode permanently disabled<br>1 = Nonvolatile protection mode enabled (Default) | Places the device in nonvolatile protection mode with password protection mode permanently disabled. When shipped from the factory, the device will operate in nonvolatile protection mode, and the memory blocks are unprotected.    | 2     |
| DQ0 | Extended memory block protection bit | 0 = Protected<br>1 = Unprotected (Default)  | If the device is shipped with the extended memory block unlocked, the block can be protected by setting this bit to 0. The extended memory block protection status can be read in auto select mode by issuing an AUTO SELECT command. | –     |

- Notes:
1. The lock register is a 16-bit, one-time programmable register. DQ[15:3] are reserved and are set to a default value of 1.
  2. The password protection mode lock bit and nonvolatile protection mode lock bit cannot both be programmed to 0. Any attempt to program one while the other is programmed causes the operation to abort, and the device returns to read mode. The device is shipped from the factory with the default setting.

**Figure 9: Lock Register Program Flowchart**



- Notes:
1. Each lock register bit can be programmed only once.
  2. See the Block Protection Command Definitions table for address-data cycle details.
  3. DQ5 and DQ1 are ignored in this algorithm flow.



## Standard Command Definitions – Address-Data Cycles

**Table 7: Standard Command Definitions – Address-Data Cycles, 8-Bit and 16-Bit**

Note 1 applies to entire table

| Command and Code/Subcode                    | Bus Size | Address and Data Cycles |    |     |    |     |    |        |        |     |    |     |         | Notes |
|---|----------|-------------------------|----|-----|----|-----|----|--------|--------|-----|----|-----|---------|-------|
|   |          | 1st                     |    | 2nd |    | 3rd |    | 4th    |        | 5th |    | 6th |         |       |
|   |          | A                       | D  | A   | D  | A   | D  | A      | D      | A   | D  | A   | D       |       |
| <b>READ and AUTO SELECT Operations</b>      |          |                         |    |     |    |     |    |        |        |     |    |     |         |       |
| READ/RESET (F0h)                            | x8       | X                       | F0 |     |    |     |    |        |        |     |    | 2   |         |       |
|   |          | AAA                     | AA | 555 | 55 | X   | F0 |        |        |     |    |     |         |       |
|   | x16      | X                       | F0 |     |    |     |    |        |        |     |    |     |         |       |
|   |          | 555                     | AA | 2AA | 55 | X   | F0 |        |        |     |    |     |         |       |
| READ CFI (98h)                              | x8       | AAA                     | 98 |     |    |     |    |        |        |     |    |     |         |       |
|   | x16      | 555                     |    |     |    |     |    |        |        |     |    |     |         |       |
| EXIT READ CFI (F0h)                         | x8       | X                       | F0 |     |    |     |    |        |        |     |    |     |         |       |
|   | x16      |                         |    |     |    |     |    |        |        |     |    |     |         |       |
| AUTO SELECT (90h)                           | x8       | AAA                     | AA | 555 | 55 | AAA | 90 | Note 3 | Note 3 |     |    |     | 4, 5    |       |
|   | x16      | 555                     |    | 2AA |    | 555 |    |        |        |     |    |     |         |       |
| EXIT AUTO SELECT (F0h)                      | x8       | X                       | F0 |     |    |     |    |        |        |     |    |     |         |       |
|   | x16      |                         |    |     |    |     |    |        |        |     |    |     |         |       |
| <b>BYPASS Operations</b>                    |          |                         |    |     |    |     |    |        |        |     |    |     |         |       |
| UNLOCK BYPASS (20h)                         | x8       | AAA                     | AA | 555 | 55 | AAA | 20 |        |        |     |    |     |         |       |
|   | x16      | 555                     |    | 2AA |    | 555 |    |        |        |     |    |     |         |       |
| UNLOCK BYPASS RESET (90h/00h)               | x8       | X                       | 90 | X   | 00 |     |    |        |        |     |    |     |         |       |
|   | x16      |                         |    |     |    |     |    |        |        |     |    |     |         |       |
| <b>PROGRAM Operations</b>                   |          |                         |    |     |    |     |    |        |        |     |    |     |         |       |
| PROGRAM (A0h)                               | x8       | AAA                     | AA | 555 | 55 | AAA | A0 | PA     | PD     |     |    |     |         |       |
|   | x16      | 555                     |    | 2AA |    | 555 |    |        |        |     |    |     |         |       |
| UNLOCK BYPASS PROGRAM (A0h)                 | x8       | X                       | A0 | PA  | PD |     |    |        |        |     | 6  |     |         |       |
|   | x16      |                         |    |     |    |     |    |        |        |     |    |     |         |       |
| WRITE TO BUFFER PROGRAM (25h)               | x8       | AAA                     | AA | 555 | 55 | BAd | 25 | BAd    | N      | PA  | PD |     | 7, 8, 9 |       |
|   | x16      | 555                     |    | 2AA |    |     |    |        |        |     |    |     |         |       |
| UNLOCK BYPASS WRITE TO BUFFER PROGRAM (25h) | x8       | BAd                     | 25 | BAd | N  | PA  | PD |        |        |     |    |     | 6       |       |
|   | x16      |                         |    |     |    |     |    |        |        |     |    |     |         |       |
| WRITE TO BUFFER PROGRAM CONFIRM (29h)       | x8       | BAd                     | 29 |     |    |     |    |        |        |     |    | 7   |         |       |
|   | x16      |                         |    |     |    |     |    |        |        |     |    |     |         |       |
| BUFFERED PROGRAM ABORT and RESET (F0h)      | x8       | AAA                     | AA | 555 | 55 | AAA | F0 |        |        |     |    |     |         |       |
|   | x16      | 555                     |    | 2AA |    | 555 |    |        |        |     |    |     |         |       |
| PROGRAM SUSPEND (B0h)                       | x8       | X                       | B0 |     |    |     |    |        |        |     |    |     |         |       |
|   | x16      |                         |    |     |    |     |    |        |        |     |    |     |         |       |





**Table 7: Standard Command Definitions – Address-Data Cycles, 8-Bit and 16-Bit (Continued)**

Note 1 applies to entire table

| Command and Code/Subcode                                     | Bus Size | Address and Data Cycles |    |     |    |          |    |          |    |          |    |          |    | Notes |
|--|----------|-------------------------|----|-----|----|----------|----|----------|----|----------|----|----------|----|-------|
|  |          | 1st                     |    | 2nd |    | 3rd      |    | 4th      |    | 5th      |    | 6th      |    |       |
|  |          | A                       | D  | A   | D  | A        | D  | A        | D  | A        | D  | A        | D  |       |
| PROGRAM RESUME (30h)   | x8       | X                       | 30 |     |    |          |    |          |    |          |    |          |    |       |
|  | x16      |                         |    |     |    |          |    |          |    |          |    |          |    |       |
| <b>ERASE Operations</b>                                      |          |                         |    |     |    |          |    |          |    |          |    |          |    |       |
| CHIP ERASE (80/10h)  | x8       | AAA                     | AA | 555 | 55 | AAA      | 80 | AAA      | AA | 555      | 55 | AAA      | 10 |       |
|  | x16      | 555                     |    | 2AA |    | 555      |    | 555      |    | 2AA      |    | 555      |    |       |
| UNLOCK BYPASS CHIP ERASE (80/10h)                            | x8       | X                       | 80 | X   | 10 |          |    |          |    |          |    |          |    | 6     |
|  | x16      |                         |    |     |    |          |    |          |    |          |    |          |    |       |
| BLOCK ERASE (80/30h)   | x8       | AAA                     | AA | 555 | 55 | AAA      | 80 | AAA      | AA | 555      | 55 | BAd      | 30 | 10    |
|  | x16      | 555                     |    | 2AA |    | 555      |    | 555      |    | 2AA      |    |          |    |       |
| UNLOCK BYPASS BLOCK ERASE (80/30h)                           | x8       | X                       | 80 | BAd | 30 |          |    |          |    |          |    |          |    | 6     |
|  | x16      |                         |    |     |    |          |    |          |    |          |    |          |    |       |
| ERASE SUSPEND (80h)  | x8       | X                       | B0 |     |    |          |    |          |    |          |    |          |    |       |
|  | x16      |                         |    |     |    |          |    |          |    |          |    |          |    |       |
| ERASE RESUME (30h)   | x8       | X                       | 30 |     |    |          |    |          |    |          |    |          |    |       |
|  | x16      |                         |    |     |    |          |    |          |    |          |    |          |    |       |
| <b>Enhanced Flash Interface (EFI) BLANK CHECK Operations</b> |          |                         |    |     |    |          |    |          |    |          |    |          |    |       |
| EFI BLANK CHECK SETUP (EB/76h)                               | x8       | AAA                     | AA | 555 | 55 | BAd + 00 | EB | BAd + 00 | 76 | BAd + 00 | 00 | BAd + 00 | 00 |       |
|  | x16      | 555                     |    | 2AA |    |          |    |          |    |          |    |          |    |       |
| EFI BLANK CHECK CONFIRM and READ (29h)                       | x8       | BAd + 00                | 29 |     |    |          |    |          |    |          |    |          |    |       |
|  | x16      |                         |    |     |    |          |    |          |    |          |    |          |    |       |

- Notes:
1. A = Address; D = Data; X = "Don't Care"; BAd = Any address in the block; N = Number of bytes (x8) or words (x16) to be programmed; PA = Program address; PD = Program data; Gray shading = Not applicable. All values in the table are hexadecimal. Some commands require both a command code and subcode.
  2. A full three-cycle RESET command sequence must be used to reset the device in the event of a buffered program abort error (DQ1 = 1).
  3. These cells represent READ cycles (versus WRITE cycles for the others).
  4. AUTO SELECT enables the device to read the manufacturer code, device code, block protection status, and extended memory block protection indicator.
  5. AUTO SELECT addresses and data are specified in the Electronic Signature table and the Extended Memory Block Protection table.
  6. For any UNLOCK BYPASS ERASE/PROGRAM command, the first two UNLOCK cycles are unnecessary.
  7. BAd must be the same as the address loaded during the WRITE TO BUFFER PROGRAM 3rd and 4th cycles.
  8. WRITE TO BUFFER PROGRAM operation: maximum cycles = 261 (x8) and 517 (x16). UNLOCK BYPASS WRITE TO BUFFER PROGRAM operation: maximum cycles = 259 (x8), 515

- (x16). WRITE TO BUFFER PROGRAM operation:  $N + 1$  = bytes (x8) or words (x16) to be programmed; maximum buffer size = 256 bytes (x8) and 512 words (x16).
9. For x8, A[ $MAX:7$ ] address pins should remain unchanged while A[6:0] and A-1 pins are used to select a byte within the  $N + 1$  byte page. For x16, A[ $MAX:9$ ] address pins should remain unchanged while A[8:0] pins are used to select a word within the  $N+1$  word page.
  10. BLOCK ERASE address cycles can extend beyond six address-data cycles, depending on the number of blocks to erase.

## **READ and AUTO SELECT Operations**

### **READ/RESET Command**

The READ/RESET (F0h) command returns the device to read mode and resets the errors in the data polling register. One or three bus WRITE operations can be used to issue the READ/RESET command. Note: A full three-cycle RESET command sequence must be used to reset the device in the event of a buffered program abort error (DQ1 = 1).

Once a PROGRAM, ERASE, or SUSPEND operation begins, RESET commands are ignored until the operation is complete. Read/reset serves primarily to return the device to read mode from a failed PROGRAM or ERASE operation. Read/reset may cause a return to read mode from undefined states that might result from invalid command sequences. A hardware reset may be required to return to normal operation from some undefined states.

To exit the unlock bypass mode, the system must issue a two-cycle UNLOCK BYPASS RESET command sequence. A READ/RESET command will not exit unlock bypass mode.

### **READ CFI Command**

The READ CFI (98h) command puts the device in read CFI mode and is only valid when the device is in read array or auto select mode. One bus WRITE cycle is required to issue the command.

Once in read CFI mode, bus READ operations will output data from the CFI memory area (Refer to the Common Flash Interface for details).

Read CFI mode is exited by performing a reset. The device returns to read mode unless it entered read CFI mode after an ERASE SUSPEND or PROGRAM SUSPEND command, in which case it returns to erase or program suspend mode.

### **AUTO SELECT Command**

At power-up or after a hardware reset, the device is in read mode. It can then be put in auto select mode by issuing an AUTO SELECT (90h) command. Auto select mode enables the following device information to be read:

- Electronic signature, which includes manufacturer and device code information as shown in the Electronic Signature table.
- Block protection, which includes the block protection status and extended memory block protection indicator, as shown in the Block Protection table.

Electronic signature or block protection information is read by executing a READ operation with control signals and addresses set, as shown in the Read Electronic Signature table or the Block Protection table, respectively. In addition, this device information can be read or set by issuing an AUTO SELECT command.

Auto select mode can be used by the programming equipment to automatically match a device with the application code to be programmed.

Three consecutive bus WRITE operations are required to issue an AUTO SELECT command. The device remains in auto select mode until a READ/RESET or READ CFI command is issued.



## 256Mb: x8/x16, 3V, MT28EW Embedded Parallel NOR READ and AUTO SELECT Operations

The device cannot enter auto select mode when a PROGRAM or ERASE operation is in progress (RY/BY# LOW). However, auto select mode can be entered if the PROGRAM or ERASE operation has been suspended by issuing a PROGRAM SUSPEND or ERASE SUSPEND command.

Auto select mode is exited by performing a reset. The device returns to read mode unless it entered auto select mode after an ERASE SUSPEND or PROGRAM SUSPEND command, in which case it returns to erase or program suspend mode.

**Table 8: Block Protection**

Note 1 applies to entire table

| Read Cycle   | CE# | OE# | WE# | Address Input         |         |    |    | Data Input/Output |          |                  |                       |                  |                    |
|--|-----|-----|-----|-----------------------|---------|----|----|-------------------|----------|------------------|-----------------------|------------------|--------------------|
|  |     |     |     | 8-Bit/16-Bit          |         |    |    | 8-Bit Only        |          | 8-Bit Only       |                       | 16-Bit Only      |                    |
|  |     |     |     | A[MAX:16]             | A[15:2] | A1 | A0 | DQ15/A-1          | DQ[14:8] | DQ[7:0]          | DQ15/A-1,<br>DQ[14:0] |                  |                    |
| <b>128-bit (0x0-0x7) Factory-Programmable Extended memory protection Indicator (bit DQ7)</b> |     |     |     |                       |         |    |    |                   |          |                  |                       |                  |                    |
| Low lock   | L   | L   | H   | L                     | L       | H  | H  | X                 | X        | 09h <sup>2</sup> | 0009h <sup>2</sup>    | 89h <sup>3</sup> | 0089h <sup>3</sup> |
| High lock  | L   | L   | H   | L                     | L       | H  | H  | X                 | X        | 19h <sup>2</sup> | 0019h <sup>2</sup>    | 99h <sup>3</sup> | 0099h <sup>3</sup> |
| <b>Block protection status</b>   |     |     |     |                       |         |    |    |                   |          |                  |                       |                  |                    |
| Protected  | L   | L   | H   | Block base<br>address | L       | H  | L  | X                 | X        | 01h              | 0001h                 |                  |                    |
| Unprotected  | L   | L   | H   |                       | L       | H  | L  | X                 | X        | 00h              | 0000h                 |                  |                    |

- Notes: 1. H = Logic level HIGH ( $V_{IH}$ ); L = Logic level LOW ( $V_{IL}$ ); X = HIGH or LOW.  
2. Customer-lockable (default).  
3. Micron prelocked.

## Read Electronic Signature

**Table 9: Read Electronic Signature – 256Mb**

Note 1 applies to entire table

| READ Cycle        | CE# | OE# | WE# | Address Input |    |    |    |    | Data Input/Output |          |            |                       |             |
|-------------------|-----|-----|-----|---------------|----|----|----|----|-------------------|----------|------------|-----------------------|-------------|
|                   |     |     |     | 8-Bit/16-Bit  |    |    |    |    | 8-Bit Only        |          | 8-Bit Only |                       | 16-Bit Only |
|                   |     |     |     | A[MAX:4]      | A3 | A2 | A1 | A0 | DQ15/A-1          | DQ[14:8] | DQ[7:0]    | DQ15/A-1,<br>DQ[14:0] |             |
| Manufacturer code | L   | L   | H   | L             | L  | L  | L  | L  | X                 | X        | 89h        | 0089h                 |             |
| Device code 1     | L   | L   | H   | L             | L  | L  | L  | H  | X                 | X        | 7Eh        | 227Eh                 |             |
| Device code 2     | L   | L   | H   | L             | H  | H  | H  | L  | X                 | X        | 22h        | 2222h                 |             |
| Device code 3     | L   | L   | H   | L             | H  | H  | H  | H  | X                 | X        | 01h        | 2201h                 |             |

- Note: 1. H = Logic level HIGH ( $V_{IH}$ ); L = Logic level LOW ( $V_{IL}$ ); X = HIGH or LOW.

## Cyclic Redundancy Check Operation

### CYCLIC REDUNDANCY CHECK Command

The CYCLIC REDUNDANCY CHECK (CRC) command is a nonsecure hash function designed to detect accidental changes to raw data. Typically, it is used in digital networks and storage devices such as hard disk drives. A CRC-enabled device calculates a short, fixed-length binary sequence known as the CRC code (or CRC). The device CRC operation will generate the CRC result of the whole device or of an address range specified by the operation. Then the CRC result is compared with the expected CRC data provided in the sequence. Finally, the device indicates a pass or fail through the data polling register. If the CRC fails, corrective action is possible, such as re-verifying with a normal READ mode or rewriting the array data.

CRC is a higher performance alternative to reading data directly to verify recently programmed data, or as a way to periodically check the data integrity of a large block of data against a stored CRC reference over the life of the product.

CRC helps improve test efficiency for programmer or burn-in stress tests. No system hardware changes are required to enable CRC.

The CRC-64 operation follows the ECMA standard; the generating polynomial is:

$$G(x) = x^{64} + x^{62} + x^{57} + x^{55} + x^{54} + x^{53} + x^{52} + x^{47} + x^{46} + x^{45} + x^{40} + x^{39} + x^{38} + x^{37} + x^{35} + x^{33} + x^{32} + x^{31} + x^{29} + x^{27} + x^{24} + x^{23} + x^{22} + x^{21} + x^{19} + x^{17} + x^{13} + x^{12} + x^{10} + x^9 + x^7 + x^4 + x + 1$$

**Note:** The data stream sequence is from LSB to MSB and the default initial CRC value is all zeros.

The CRC command sequences are shown in the tables below, for an entire die or for a selected range, respectively.

### Cyclic Redundancy Check Operation Command Sequence

**Table 10: Command Sequence – Range of Blocks**

Note 1 and 2 apply to entire table.

| Word Mode |          | Byte Mode             |                       | Description                         | Notes |
|-----------|----------|-----------------------|-----------------------|-------------------------------------|-------|
| A[MAX:0]  | DQ[15:0] | A[MAX:0],<br>DQ15/A-1 | DQ[7:0]               |                                     |       |
| 0000555   | 00AAh    | 0000AAA               | AAh                   | UI unlock cycle 1                   |       |
| 00002AA   | 0055h    | 0000555               | 55h                   | UI unlock cycle 2                   |       |
| 0000000   | 00EBh    | 0000000               | EBh                   | Extended function interface command |       |
| 0000000   | 0027h    | 0000000               | 27h                   | CRC sub-op code                     |       |
| 0000000   | 000Ah    | 0000000               | 15h                   | N-1 data count                      |       |
| 0000000   | FFFEh    | 0000000               | FEh                   | CRC operation option data           |       |
|           |          | 0000001               | FFh                   |                                     |       |
| 0000001   | Data     | 0000002               | Low byte of the data  | 1st word of 64-bit expected CRC     |       |
|           |          | 0000003               | High byte of the data |                                     |       |

**Table 10: Command Sequence – Range of Blocks (Continued)**

Note 1 and 2 apply to entire table.

| Word Mode |                                  | Byte Mode             |                                  | Description  | Notes |
|-----------|----------------------------------|-----------------------|----------------------------------|--|-------|
| A[MAX:0]  | DQ[15:0]                         | A[MAX:0],<br>DQ15/A-1 | DQ[7:0]                          |  |       |
| 0000002   | Data                             | 0000004               | Low byte of the data             | 2nd word of 64-bit expected CRC                      |       |
|           |                                  | 0000005               | High byte of the data            |  |       |
| 0000003   | Data                             | 0000006               | Low byte of the data             | 3rd word of 64-bit expected CRC                      |       |
|           |                                  | 0000007               | High byte of the data            |  |       |
| 0000004   | Data                             | 0000008               | Low byte of the data             | 4th word of 64-bit expected CRC                      |       |
|           |                                  | 0000009               | High byte of the data            |  |       |
| 0000005   | A <sub>14</sub> -A <sub>1</sub>  | 000000A               | A <sub>6</sub> -A <sub>1</sub>   | Byte address to start                                | 3     |
|           |                                  | 0000011               | A <sub>14</sub> -A <sub>7</sub>  |  |       |
| 0000006   | A <sub>30</sub> -A <sub>15</sub> | 000000C               | A <sub>22</sub> -A <sub>15</sub> | Byte address to start                                | 3     |
|           |                                  | 000000D               | A <sub>30</sub> -A <sub>23</sub> |  |       |
| 0000007   | Reserved                         | 000000E               | Reserved                         | Default as 0000h                                     |       |
|           |                                  | 000000F               | Reserved                         |  |       |
| 0000008   | A <sub>14</sub> -A <sub>1</sub>  | 0000010               | A <sub>6</sub> -A <sub>1</sub>   | Byte address to stop                                 | 3     |
|           |                                  | 0000011               | A <sub>14</sub> -A <sub>7</sub>  |  |       |
| 0000009   | A <sub>30</sub> -A <sub>15</sub> | 0000012               | A <sub>22</sub> -A <sub>15</sub> | Byte address to stop                                 | 3     |
|           |                                  | 0000013               | A <sub>30</sub> -A <sub>23</sub> |  |       |
| 000000A   | Reserved                         | 0000014               | Reserved                         | Default as 0000h                                     |       |
|           |                                  | 0000015               | Reserved                         |  |       |
| 0000000   | 0029h                            | 0000000               | 29h                              | Confirm command                                      |       |
| 0000000   | Read                             | 0000000               | Read                             | Continue data polling to wait for device to be ready |       |

- Notes:
1. If the CRC check fails, a check error is generated by setting DQ5 = 1.
  2. This is a byte-aligned operation, whether BYTE# is HIGH or LOW.
  3. The stop address must be bigger than the start address; otherwise, the algorithm will take no action.

**Table 11: Command Sequence – Entire Chip**

| Word Mode         |                   | Byte Mode                      |                       | Description  |
|-------------------|-------------------|--------------------------------|-----------------------|--|
| A[ <b>MAX:0</b> ] | DQ[ <b>15:0</b> ] | A[ <b>MAX:0</b> ],<br>DQ15/A-1 | DQ[ <b>7:0</b> ]      |  |
| 0000555           | 00AAh             | 0000AAA                        | AAh                   | UI unlock cycle 1                                    |
| 00002AA           | 0055h             | 0000555                        | 55h                   | UI unlock cycle 2                                    |
| 0000000           | 00EBh             | 0000000                        | EBh                   | Extended function interface command                  |
| 0000000           | 0027h             | 0000000                        | 27h                   | CRC sub-op code                                      |
| 0000000           | 0004h             | 0000000                        | 09h                   | N-1 data count                                       |
| 0000000           | FFFFh             | 0000000                        | FFh                   | CRC operation option data                            |
|                   |                   | 0000001                        | FFh                   |  |
| 0000001           | Data              | 0000002                        | Low byte of the data  | 1st word of 64-bit expected CRC                      |
|                   |                   | 0000003                        | High byte of the data |  |
| 0000002           | Data              | 0000004                        | Low byte of the data  | 2nd word of 64-bit expected CRC                      |
|                   |                   | 0000005                        | High byte of the data |  |
| 0000003           | Data              | 0000006                        | Low byte of the data  | 3rd word of 64-bit expected CRC                      |
|                   |                   | 0000007                        | High byte of the data |  |
| 0000004           | Data              | 0000008                        | Low byte of the data  | 4th word of 64-bit expected CRC                      |
|                   |                   | 0000009                        | High byte of the data |  |
| 0000000           | 0029h             | 0000000                        | 0029h                 | Confirm command                                      |
| 0000000           | Read              | 0000000                        | Read                  | Continue data polling to wait for device to be ready |

Note: 1. Applies to entire table: If the CRC check fails, a check error is generated by setting DQ5 = 1.

## Bypass Operations

### UNLOCK BYPASS Command

The UNLOCK BYPASS (20h) command is used to place the device in unlock bypass mode. Three bus WRITE operations are required to issue the UNLOCK BYPASS command.

When the device enters unlock bypass mode, the two initial UNLOCK cycles required for a standard PROGRAM or ERASE operation are not needed, thus enabling faster total program or erase time.

The UNLOCK BYPASS command is used in conjunction with UNLOCK BYPASS PROGRAM or UNLOCK BYPASS ERASE commands to program or erase the device faster than with standard PROGRAM or ERASE commands. Using these commands can save considerable time when the cycle time to the device is long. When in unlock bypass mode, only the following commands are valid:

- The UNLOCK BYPASS PROGRAM command can be issued to program addresses within the device.
- The UNLOCK BYPASS BLOCK ERASE command can then be issued to erase one or more memory blocks.
- The UNLOCK BYPASS CHIP ERASE command can be issued to erase the whole memory array.
- The UNLOCK BYPASS WRITE TO BUFFER PROGRAM and UNLOCK BYPASS ENHANCED WRITE TO BUFFER PROGRAM commands can be issued to speed up the programming operation.
- The UNLOCK BYPASS RESET command can be issued to return the device to read mode.

In unlock bypass mode, the device can be read as if in read mode.

In addition to the UNLOCK BYPASS command, when  $V_{PP}/WP\#$  is raised to  $V_{HH}$ , the device automatically enters unlock bypass mode. When  $V_{PP}/WP\#$  returns to  $V_{IH}$  or  $V_{IL}$ , the device is no longer in unlock bypass mode, and normal operation resumes. The transitions from  $V_{IH}$  to  $V_{HH}$  and from  $V_{HH}$  to  $V_{IH}$  must be slower than  $t_{VHVPP}$ . (See the Accelerated Program, Data Polling/ Toggle AC Characteristics.)

**Note:** Micron recommends entering and exiting unlock bypass mode using the ENTER UNLOCK BYPASS and UNLOCK BYPASS RESET commands rather than raising  $V_{PP}/WP\#$  to  $V_{HH}$ .  $V_{PP}/WP\#$  should never be raised to  $V_{PPH}$  from any mode except read mode; otherwise, the device may be left in an indeterminate state.  $V_{PP}/WP\#$  should not remain at  $V_{HH}$  for than 80 hours cumulative.

### UNLOCK BYPASS RESET Command

The UNLOCK BYPASS RESET (90/00h) command is used to return to read/reset mode from unlock bypass mode. Two bus WRITE operations are required to issue the UNLOCK BYPASS RESET command. The READ/RESET command does not exit from unlock bypass mode.



## Program Operations

### PROGRAM Command

The PROGRAM (A0h) command can be used to program a value to one address in the memory array. The command requires four bus WRITE operations, and the final WRITE operation latches the address and data in the internal state machine and starts the program/erase controller. After programming has started, bus READ operations output the data polling register content.

Programming can be suspended and then resumed by issuing a PROGRAM SUSPEND command and a PROGRAM RESUME command, respectively.

If the address falls in a protected block, the PROGRAM command is ignored, and the data remains unchanged. The data polling register is not read, and no error condition is given.

After the PROGRAM operation has completed, the device returns to read mode, unless an error has occurred. When an error occurs, bus READ operations to the device continue to output the data polling register. A READ/RESET command must be issued to reset the error condition and return the device to read mode.

The PROGRAM command cannot change a bit set to 0 back to 1, and an attempt to do so is masked during a PROGRAM operation. Instead, an ERASE command must be used to set all bits in one memory block or in the entire memory from 0 to 1.

The PROGRAM operation is aborted by performing a hardware reset or by powering down the device. In this case, data integrity cannot be ensured, and it is recommended that the words or bytes that were aborted be reprogrammed.

### UNLOCK BYPASS PROGRAM Command

When the device is in unlock bypass mode, the UNLOCK BYPASS PROGRAM (A0h) command can be used to program one address in the memory array. The command requires two bus WRITE operations instead of four required by a standard PROGRAM command; the final WRITE operation latches the address and data and starts the program/erase controller (The standard PROGRAM command requires four bus WRITE operations). The PROGRAM operation using the UNLOCK BYPASS PROGRAM command behaves identically to the PROGRAM operation using the PROGRAM command. The operation cannot be aborted. A bus READ operation to the memory outputs the data polling register.

### WRITE TO BUFFER PROGRAM Command

The WRITE TO BUFFER PROGRAM (25h) command makes use of the program buffer to speed up programming and dramatically reduces system programming time compared to the standard non-buffered PROGRAM command. This product supports a 512-word (x16) or 256-byte (x8) maximum program buffer.

When issuing a WRITE TO BUFFER PROGRAM command,  $V_{pp}/WP\#$  can be held HIGH or raised to  $V_{HH}$ . Also, it can be held LOW if the block is not the lowest or highest block, depending on the part number.

The following successive steps are required to issue the WRITE TO BUFFER PROGRAM command:

First, two UNLOCK cycles are issued. Next, a third bus WRITE cycle sets up the WRITE TO BUFFER PROGRAM command. The set-up code can be addressed to any location within the targeted block. Then, a fourth bus WRITE cycle sets up the number of words/bytes to be programmed. Value  $n$  is written to the same block address, where  $n + 1$  is the number of words/bytes to be programmed. Value  $n + 1$  must not exceed the size of the program buffer, or the operation will abort. A fifth cycle loads the first address and data to be programmed. Last,  $n$  bus WRITE cycles load the address and data for each word/byte into the program buffer. Addresses must lie within the range from *the start address + 1* to *the start address + (n - 1)*.

Optimum programming performance and lower power usage are achieved by aligning the starting address at the beginning of a 512-word boundary ( $A[8:0] = 0x000h$ ). Any buffer size smaller than 512 words is allowed within a 512-word boundary, while all addresses used in the operation must lie within the 512-word boundary. In addition, any crossing boundary buffer program will result in a program abort. For a x8 application, maximum buffer size is 256 bytes; for a x16 application, the maximum buffer size is 1024 bytes.

To program the content of the program buffer, this command must be followed by a WRITE TO BUFFER PROGRAM CONFIRM command.

If an address is written several times during a WRITE TO BUFFER PROGRAM operation, the address/data counter will be decremented at each data load operation, and the data will be programmed to the last word loaded into the buffer.

Invalid address combinations or the incorrect sequence of bus WRITE cycles will abort the WRITE TO BUFFER PROGRAM command.

The data polling register bits DQ1, DQ5, DQ6, DQ7 can be used to monitor the device status during a WRITE TO BUFFER PROGRAM operation.

The WRITE TO BUFFER PROGRAM command should not be used to change a bit set to 0 back to 1, and an attempt to do so is masked during the operation. Rather than the WRITE TO BUFFER PROGRAM command, the ERASE command should be used to set memory bits from 0 to 1.

**Figure 10: Boundary Condition of Program Buffer Size**

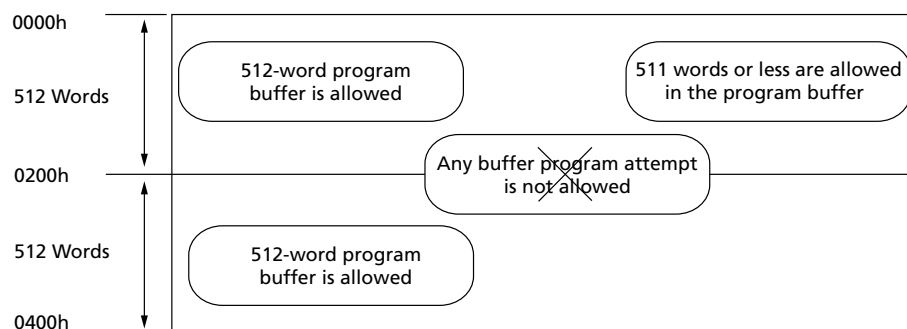


Figure 11: WRITE TO BUFFER PROGRAM Flowchart



- Notes:
1.  $n + 1$  is the number of addresses to be programmed.
  2. The BUFFERED PROGRAM ABORT AND RESET command (3 cycles reset) must be issued to return the device to read mode.
  3. When the block address is specified, any address in the selected block address space is acceptable. However, when loading program buffer address with data, all addresses must fall within the selected program buffer page.

## **UNLOCK BYPASS WRITE TO BUFFER PROGRAM Command**

When the device is in unlock bypass mode, the UNLOCK BYPASS WRITE TO BUFFER (25h) command can be used to program the device in fast program mode. The command requires two bus WRITE operations fewer than the standard WRITE TO BUFFER PROGRAM command.

The UNLOCK BYPASS WRITE TO BUFFER PROGRAM command behaves the same way as the WRITE TO BUFFER PROGRAM command: the operation cannot be aborted, and a bus READ operation to the memory outputs the data polling register.

The WRITE TO BUFFER PROGRAM CONFIRM command is used to confirm an UNLOCK BYPASS WRITE TO BUFFER PROGRAM command and to program the  $n + 1$  words/bytes loaded in the program buffer by this command.

## **WRITE TO BUFFER PROGRAM CONFIRM Command**

The WRITE TO BUFFER PROGRAM CONFIRM (29h) command is used to confirm a WRITE TO BUFFER PROGRAM command and to program the  $n + 1$  words/bytes loaded in the program buffer by this command.

## **BUFFERED PROGRAM ABORT AND RESET Command**

A BUFFERED PROGRAM ABORT AND RESET (F0h) command must be issued to reset the device to read mode when the BUFFER PROGRAM operation is aborted. The buffer programming sequence can be aborted in the following ways:

- Load a value that is greater than the page buffer size during the number of locations to program in the WRITE TO BUFFER PROGRAM command.
- Write to an address in a different block than the one specified during the WRITE BUFFER LOAD command.
- Write an address/data pair to a different write buffer page than the one selected by the starting address during the program buffer data loading stage of the operation.
- Write data other than the CONFIRM command after the specified number of data load cycles.

The abort condition is indicated by  $DQ1 = 1$ ,  $DQ7 = DQ7\#$  (for the last address location loaded),  $DQ6 = \text{toggle}$ , and  $DQ5 = 0$  (all of which are data polling register bits). A BUFFERED PROGRAM ABORT and RESET command sequence must be written to reset the device for the next operation.

**Note:** The full three-cycle BUFFERED PROGRAM ABORT and RESET command sequence is required when using buffer programming features in unlock bypass mode.

## **PROGRAM SUSPEND Command**

The PROGRAM SUSPEND (B0h) command can be used to interrupt a program operation so that data can be read from another block. When the PROGRAM SUSPEND command is issued during a program operation, the device suspends the operation within the program suspend latency time and updates the data polling register bits.

After the program operation has been suspended, data can be read from any address. However, data is invalid when read from an address where a program operation has been suspended.

The PROGRAM SUSPEND command may also be issued during a PROGRAM operation while an erase is suspended. In this case, data may be read from any address not in erase suspend or program suspend mode. To read from the extended memory block area (one-time programmable area), the ENTER/EXIT EXTENDED MEMORY BLOCK command sequences must be issued.

The system may also issue the AUTO SELECT command sequence when the device is in program suspend mode. The system can read as many auto select codes as required. When the device exits auto select mode, the device reverts to program suspend mode and is ready for another valid operation.

The PROGRAM SUSPEND operation is aborted by performing a device reset or power-down. In this case, data integrity cannot be ensured, and it is recommended that the words or bytes that were aborted be reprogrammed.

### PROGRAM RESUME Command

The PROGRAM RESUME (30h) command must be issued to exit a program suspend mode and resume a PROGRAM operation. The controller can use DQ7 or DQ6 data polling bits to determine the status of the PROGRAM operation. After a PROGRAM RESUME command is issued, subsequent PROGRAM RESUME commands are ignored. Another PROGRAM SUSPEND command can be issued after the device has resumed programming.

## ACCELERATED BUFFERED PROGRAM Operations

ACCELERATED BUFFER PROGRAM operations provides faster performance than standard program command sequences. Operations are enabled through  $V_{PP}/WP\#$  under the  $V_{HH}$  voltage supply.

When the system asserts  $V_{HH}$  on input, the device automatically enters the UNLOCK BYPASS mode, which enables the system to use the UNLOCK BYPASS WRITE TO BUFFER PROGRAM (25h) command sequence.

Removing  $V_{HH}$  from the  $V_{PP}$  upon completion of the embedded program operation returns the device to normal operation.

**Table 12: ACCELERATED PROGRAM Requirements and Recommendations**

| Device State                      | Requirements/Recommendations   |
|-----------------------------------|--|
| Device blocks                     | <b>Requirement:</b> Must be unprotected prior to raising $V_{PP}/WP\#$ to $V_{HH}$   |
| $V_{HH}$ applied to $V_{PP}/WP\#$ | <b>Requirement:</b> Maximum cumulative period of 80 hours.   |
| $V_{PP}/WP\#$                     | <b>Requirement:</b> Must not be at $V_{HH}$ for operations except ACCELERATED BUFFERED PROGRAM and CHIP ERASE; otherwise device can be damaged |
|                                   | <b>Recommendation:</b> Keep stable to $V_{HH}$ during ACCELERATED BUFFERED PROGRAM operation   |
| Power-up                          | <b>Recommendation:</b> Apply $V_{HH}$ on $V_{PP}/WP\#$ after $V_{CC}/V_{CCQ}$ is stable on.  |
| Power-down                        | <b>Recommendation:</b> Adjust $V_{PP}/WP\#$ from $V_{HH}$ to $V_{IH}/V_{IL}$ before $V_{CC}/V_{CCQ}$ goes LOW.                                 |

## Erase Operations

### CHIP ERASE Command

The CHIP ERASE (80/10h) command erases the entire chip. Six bus WRITE operations are required to issue the command and start the program/erase controller.

Protected blocks are not erased. If all blocks are protected, the data remains unchanged. No error is reported when protected blocks are not erased.

During the CHIP ERASE operation, the device ignores all other commands, including ERASE SUSPEND. It is not possible to abort the operation. All bus READ operations during CHIP ERASE output the data polling register on the data I/Os. See the Data Polling Register section for more details.

After the CHIP ERASE operation completes, the device returns to read mode, unless an error has occurred. If an error occurs, the device will continue to output the data polling register.

When the operation fails, a READ/RESET command must be issued to reset the error condition and return to read mode. The status of the array must be confirmed through the BLANK CHECK operation and the BLOCK ERASE command re-issued to the failed block.

The CHIP ERASE command sets all of the bits in unprotected blocks of the device to 1. All previous data is lost.

The operation is aborted by performing a reset or by powering down the device. In this case, data integrity cannot be ensured, and it is recommended that the entire chip be erased again.

### UNLOCK BYPASS CHIP ERASE Command

When the device is in unlock bypass mode, the UNLOCK BYPASS CHIP ERASE (80/10h) command can be used to erase all memory blocks at one time. The command requires only two bus WRITE operations instead of six using the standard CHIP ERASE command. The final bus WRITE operation starts the program/erase controller.

The UNLOCK BYPASS CHIP ERASE command behaves the same way as the CHIP ERASE command: the operation cannot be aborted, and a bus READ operation to the memory outputs the data polling register.

### BLOCK ERASE Command

The BLOCK ERASE (80/30h) command erases a list of one or more blocks. It sets all bits in the selected, unprotected blocks to 1. All previous, selected, unprotected blocks data in the selected blocks is lost.

Six bus WRITE operations are required to select the first block in the list. Each additional block in the list can be selected by repeating the sixth bus WRITE operation using the address of the additional block. After the command sequence is written, a block erase timeout occurs.

During the period specified by the block erase timeout parameter, additional block addresses and BLOCK ERASE commands can be written. Any command except BLOCK ERASE or ERASE SUSPEND during this timeout period resets that block to the read

mode. The system can monitor DQ3 to determine if the block erase timer has timed out.

After the program/erase controller has started, it is not possible to select any more blocks. Each additional block must therefore be selected within the timeout period of the last block. The timeout timer restarts when an additional block is selected. After the sixth bus WRITE operation, a bus READ operation outputs the data polling register. See the WE#-Controlled Program waveforms for details on how to identify if the program/erase controller has started the BLOCK ERASE operation.

After the BLOCK ERASE operation completes, the device returns to read mode, unless an error has occurred. If an error occurs, bus READ operations will continue to output the data polling register. A READ/RESET command must be issued to reset the error condition and return to read mode.

If any selected blocks are protected, they are ignored, and all the other selected blocks are erased. If all selected blocks are protected, the data remains unchanged. No error condition is given when protected blocks are not erased.

During the BLOCK ERASE operation, the device ignores all commands except the ERASE SUSPEND command and the READ/RESET command, which is accepted only during the timeout period. The operation is aborted by performing a hardware reset or powering down the device. In this case, data integrity cannot be ensured, and it is recommended that the aborted blocks be erased again.

## **UNLOCK BYPASS BLOCK ERASE Command**

When the device is in unlock bypass mode, the UNLOCK BYPASS BLOCK ERASE (80/30h) command can be used to erase one or more memory blocks at a time. The command requires two bus WRITE operations instead of six using the standard BLOCK ERASE command. The final bus WRITE operation latches the address of the block and starts the program/erase controller.

To erase multiple blocks (after the first two bus WRITE operations have selected the first block in the list), each additional block in the list can be selected by repeating the second bus WRITE operation using the address of the additional block.

Any command except BLOCK ERASE or ERASE SUSPEND during a timeout period resets that block to the read mode. The system can monitor DQ3 to determine if the block erase timer has timed out.

The UNLOCK BYPASS BLOCK ERASE command behaves the same way as the BLOCK ERASE command: the operation cannot be aborted, and a bus READ operation to the memory outputs the data polling register. See the BLOCK ERASE Command section for details.

## **ERASE SUSPEND Command**

The ERASE SUSPEND (B0h) command temporarily suspends a BLOCK ERASE operation. One bus WRITE operation is required to issue the command. The block address is "Don't Care."

The program/erase controller suspends the ERASE operation within the erase suspend latency time of the ERASE SUSPEND command being issued. However, when the ERASE SUSPEND command is written during the block erase timeout, the device immediately terminates the timeout period and suspends the ERASE operation. After the

program/erase controller has stopped, the device operates in read mode, and the erase is suspended.

During an ERASE SUSPEND operation, it is possible to execute these operations in arrays that are not suspended:

- READ (main memory array)
- PROGRAM
- WRITE TO BUFFER PROGRAM
- AUTO SELECT
- READ CFI
- UNLOCK BYPASS
- Extended memory block commands
- READ/RESET

Reading from a suspended block will output the data polling register. If an attempt is made to program in a protected or suspended block, the PROGRAM command is ignored and the data remains unchanged; also, the data polling register is not read and no error condition is given.

Before the RESUME command is initiated, the READ/RESET command must be issued to exit AUTO SELECT and READ CFI operations. In addition, the EXIT UNLOCK BYPASS and EXIT EXTENDED MEMORY BLOCK commands must be issued to exit unlock bypass and the extended memory block modes.

An ERASE SUSPEND command is ignored if it is written during a CHIP ERASE operation.

If the ERASE SUSPEND operation is aborted by performing a device hardware reset or power-down, data integrity cannot be ensured, and it is recommended that the suspended blocks be erased again.

## ERASE RESUME Command

The ERASE RESUME (30h) command restarts the program/erase controller after an ERASE SUSPEND operation.

The device must be in read array mode before the RESUME command will be accepted. An erase can be suspended and resumed more than once.

## ACCELERATED CHIP ERASE Operations

The ACCELERATED CHIP ERASE operation provides faster performance than the standard CHIP ERASE command sequence. Operations are enabled through  $V_{PP}/WP\#$  under the  $V_{HH}$  voltage supply.

When the system asserts  $V_{HH}$  on input, the device automatically enters the UNLOCK BYPASS mode, which enables the system to use the UNLOCK BYPASS CHIP ERASE (80/30h) command sequence.

When a block is protected, the CHIP ERASE command skips the protected block and continues with next block erase. The command algorithm skips a block that failed to erase and continues with the remaining blocks. The fail flag will be set for the operation.

Removing  $V_{HH}$  from the  $V_{PP}/WP\#$  upon completion of the embedded erase operation returns the device to normal operation. When an error occurs or when the operation



fails, the array status should be confirmed through the BLANK CHECK operation and the BLOCK ERASE command re-issued to the failed block.

**Table 13: ACCELERATED CHIP ERASE Requirements and Recommendations**

| Device Component/State            | Requirements/Recommendations   |
|-----------------------------------|--|
| $V_{pp}/WP\#$                     | <b>Requirement:</b> Must not be at $V_{HH}$ for operations except ACCELERATED PROGRAM and CHIP ERASE; otherwise device can be damaged. |
| $V_{HH}$ applied to $V_{pp}/WP\#$ | <b>Requirement:</b> Maximum cumulative period of 80 hours.   |
| Power-up                          | <b>Recommendation:</b> Apply $V_{HH}$ on $V_{pp}/WP\#$ after $V_{CC}/V_{CCQ}$ is stable on.  |
| Power-down                        | <b>Recommendation:</b> Adjust $V_{pp}/WP\#$ from $V_{HH}$ to $V_{IH}/V_{IL}$ before $V_{CC}/V_{CCQ}$ goes LOW.                         |

## BLANK CHECK Operation

### BLANK CHECK Commands

Two commands are required to execute a BLANK CHECK operation: BLANK CHECK SETUP (EB/76h) and BLANK CHECK CONFIRM AND READ (29h).

The BLANK CHECK operation determines whether a specified block is blank (that is, completely erased). It can also be used to determine whether a previous ERASE operation was successful, including ERASE operations that might have been interrupted by power loss.

The BLANK CHECK operation checks for cells that are programmed or over-erased. If it finds any, it returns a failure status, indicating that the block is not blank. If it returns a passing status, the block is guaranteed blank (all 1s) and is ready to program.

Before executing, the ERASE operation initiates an embedded BLANK CHECK operation, and if the target block is blank, the ERASE operation is skipped, benefitting overall cycle performance; otherwise, the ERASE operation continues.

The BLANK CHECK operation can occur in only one block at a time, and during its execution, reading the data polling register is the only other operation allowed. Reading from any address in the device enables reading the data polling register to monitor blank check progress or errors. Operations such as READ (array data), PROGRAM, ERASE, and any suspended operation are not allowed.

After the BLANK CHECK operation has completed, the device returns to read mode unless an error has occurred. When an error occurs, the device continues to output data polling register data. A READ/RESET command must be issued to reset the error condition and return the device to read mode.

## Device Protection

### Hardware Protection

The  $V_{PP}/WP\#$  function provides a hardware method of protecting either the highest or lowest block. When  $V_{PP}/WP\#$  is LOW, PROGRAM and ERASE operations on either of these block options is ignored to provide protection. When  $V_{PP}/WP\#$  is HIGH, the device reverts to the previous protection status for the highest or lowest block. PROGRAM and ERASE operations can modify the data in either of these block options unless block protection is enabled.

**Note:** Micron highly recommends driving  $V_{PP}/WP\#$  HIGH or LOW. If a system needs to float the  $V_{PP}/WP\#$  pin, without a pull-up/pull-down resistor and no capacitor, then an internal pull-up resistor is enabled.

**Table 14:  $V_{PP}/WP\#$  Functions**

| $V_{PP}/WP\#$ Settings | Function  |
|------------------------|---|
| $V_{IL}$               | Highest or lowest block is protected.   |
| $V_{IH}$               | Highest or lowest block is unprotected unless software protection is activated. |

### Software Protection

The following software protection modes are available:

- Volatile protection
- Nonvolatile protection
- Password protection

The device is shipped with all blocks unprotected. On first use, the device defaults to the nonvolatile protection mode but can be activated in either the nonvolatile protection or password protection mode.

The desired protection mode is activated by setting either the nonvolatile protection mode lock bit or the password protection mode lock bit of the lock register (see the Lock Register section). Both bits are one-time-programmable and nonvolatile; therefore, after the protection mode has been activated, it cannot be changed, and the device is set permanently to operate in the selected protection mode. It is recommended that the desired software protection mode be activated when first programming the device.

For the highest or lowest block, a higher level of block protection can be achieved by locking the block using nonvolatile protection mode and holding  $V_{PP}/WP\#$  LOW.

Blocks with volatile protection and nonvolatile protection can coexist within the memory array. If the user attempts to program or erase a protected block, the device ignores the command and returns to read mode.

The block protection status can be read by performing a read electronic signature or by issuing an AUTO SELECT command (see the Block Protection table).

Refer to the Block Protection Status table and the Software Protection Scheme figure for details on the block protection scheme. Refer to the Protection Operations section for a description of the command sets.

## **Volatile Protection Mode**

Volatile protection enables the software application to protect blocks against inadvertent change and can be disabled when changes are needed. Volatile protection bits are unique for each block and can be individually modified. Volatile protection bits control the protection scheme only for unprotected blocks whose nonvolatile protection bits are cleared to 1. Issuing a PROGRAM VOLATILE PROTECTION BIT or CLEAR VOLATILE PROTECTION BIT command sets to 0 or clears to 1 the volatile protection bits and places the associated blocks in the protected (0) or unprotected (1) state, respectively. The volatile protection bit can be set or cleared as often as needed.

When the device is first shipped, or after a power-up or hardware reset, the volatile protection bits default to 1 (unprotected).

## **Nonvolatile Protection Mode**

A nonvolatile protection bit is assigned to each block. Each of these bits can be set for protection individually by issuing a PROGRAM NONVOLATILE PROTECTION BIT command. Also, each device has one global volatile bit called the nonvolatile protection bit lock bit; it can be set to protect all nonvolatile protection bits at once. This global bit must be set to 0 only after all nonvolatile protection bits are configured to the desired settings. When set to 0, the nonvolatile protection bit lock bit prevents changes to the state of the nonvolatile protection bits. When cleared to 1, the nonvolatile protection bits can be set and cleared using the PROGRAM NONVOLATILE PROTECTION BIT and CLEAR ALL NONVOLATILE PROTECTION BITS commands, respectively.

No software command unlocks the nonvolatile protection bit lock bit unless the device is in password protection mode; in nonvolatile protection mode, the nonvolatile protection bit lock bit can be cleared only by taking the device through a hardware reset or power-up.

Nonvolatile protection bits cannot be cleared individually; they must be cleared all at once using a CLEAR ALL NONVOLATILE PROTECTION BITS command. They will remain set through a hardware reset or a power-down/power-up sequence.

If one of the nonvolatile protection bits needs to be cleared (unprotected), additional steps are required: First, the nonvolatile protection bit lock bit must be cleared to 1, using either a power-cycle or hardware reset. Then, the nonvolatile protection bits can be changed to reflect the desired settings. Finally, the nonvolatile protection bit lock bit must be set to 0 to lock the nonvolatile protection bits. The device now will operate normally.

To achieve the best protection, the PROGRAM NONVOLATILE PROTECTION LOCK BIT command should be executed early in the boot code, and the boot code should be protected by holding  $V_{pp}/WP\#$  LOW.

Nonvolatile protection bits and volatile protection bits have the same function when  $V_{pp}/WP\#$  is HIGH or when  $V_{pp}/WP\#$  is at the voltage for program acceleration ( $V_{HH}$ ).

## **Password Protection Mode**

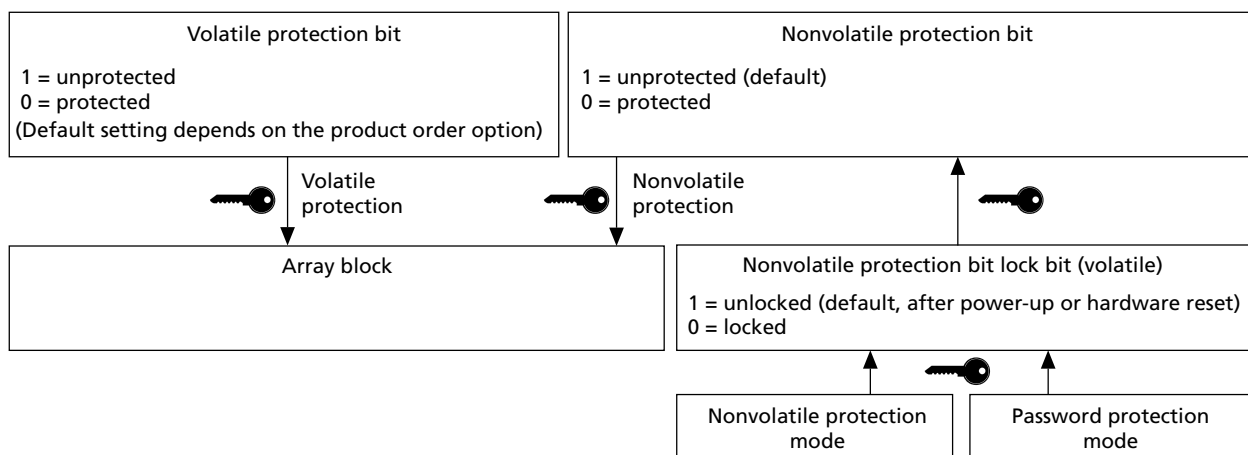
The password protection mode provides a higher level of security than the nonvolatile protection mode by requiring a 64-bit password to unlock the nonvolatile protection bit lock bit. In addition to this password requirement, the nonvolatile protection bit lock bit is set to 0 after power-up and reset to maintain the device in password protection mode.

Executing the UNLOCK PASSWORD command by entering the correct password clears the nonvolatile protection bit lock bit, enabling the block nonvolatile protection bits to be modified. If the password provided is incorrect, the nonvolatile protection bit lock bit remains locked, and the state of the nonvolatile protection bits cannot be modified.

To place the device in password protection mode, the following two steps are required: First, before activating the password protection mode, a 64-bit password must be set and the setting verified. Password verification is allowed only before the password protection mode is activated. Next, password protection mode is activated by programming the password protection mode lock bit to 0. This operation is irreversible. After the bit is programmed, it cannot be erased, the device remains permanently in password protection mode, and the 64-bit password can be neither retrieved nor reprogrammed. In addition, all commands to the address where the password is stored are disabled.

**Note:** There is no means to verify the password after password protection mode is enabled. If the password is lost after enabling the password protection mode, there is no way to clear the nonvolatile protection bit lock bit.

**Figure 12: Software Protection Scheme**



- Notes:
1. Volatile protection bits are programmed and cleared individually. Nonvolatile protection bits are programmed individually and cleared collectively.
  2. Once programmed to 0, the nonvolatile protection bit lock bit can be reset to 1 only by taking the device through a power-up or hardware reset.

**Table 15: Block Protection Status**

| Nonvolatile Protection Bit Lock Bit <sup>1</sup> | Nonvolatile Protection Bit <sup>2</sup> | Volatile Protection Bit <sup>3</sup> | Block Protection Status <sup>4</sup> | Block Protection Status   |
|--|---|--------------------------------------|--------------------------------------|---|
| 1  | 1                                       | 1                                    | 00h                                  | Block unprotected; nonvolatile protection bit changeable.   |
| 1  | 1                                       | 0                                    | 01h                                  | Block protected by volatile protection bit; nonvolatile protection bit changeable.                                  |
| 1  | 0                                       | 1                                    | 01h                                  | Block protected by nonvolatile protection bit; nonvolatile protection bit changeable.                               |
| 1  | 0                                       | 0                                    | 01h                                  | Block protected by nonvolatile protection bit and volatile protection bit; nonvolatile protection bit changeable.   |
| 0  | 1                                       | 1                                    | 00h                                  | Block unprotected; nonvolatile protection bit unchangeable.   |
| 0  | 1                                       | 0                                    | 01h                                  | Block protected by volatile protection bit; nonvolatile protection bit unchangeable.                                |
| 0  | 0                                       | 1                                    | 01h                                  | Block protected by nonvolatile protection bit; nonvolatile protection bit unchangeable.                             |
| 0  | 0                                       | 0                                    | 01h                                  | Block protected by nonvolatile protection bit and volatile protection bit; nonvolatile protection bit unchangeable. |

- Notes:
1. Nonvolatile protection bit lock bit: when cleared to 1, all nonvolatile protection bits are unlocked; when set to 0, all nonvolatile protection bits are locked.
  2. Block nonvolatile protection bit: when cleared to 1, the block is unprotected; when set to 0, the block is protected.
  3. Block volatile protection bit: when cleared to 1, the block is unprotected; when set to 0, the block is protected.
  4. Block protection status is checked under AUTO SELECT mode.



## Block Protection Command Definitions – Address-Data Cycles

**Table 16: Block Protection Command Definitions – Address-Data Cycles, 8-Bit and 16-Bit**

Notes 1 and 2 apply to entire table

| Command and Code/Subcode                        | Bus Size | Address and Data Cycles |            |      |      |     |      |     |      |     |          | Notes |            |
|---|----------|-------------------------|------------|------|------|-----|------|-----|------|-----|----------|-------|------------|
|   |          | 1st                     |            | 2nd  |      | 3rd |      | 4th |      | ... | nth      |       |            |
|   |          | A                       | D          | A    | D    | A   | D    | A   | D    |     | A        |       | D          |
| <b>LOCK REGISTER Commands</b>                   |          |                         |            |      |      |     |      |     |      |     |          |       |            |
| ENTER LOCK REGISTER COMMAND SET (40h)           | x8       | AAA                     | AA         | 555  | 55   | AAA | 40   |     |      |     |          |       | 3          |
|   | x16      | 555                     | AA         | 2AA  | 55   | 555 |      |     |      |     |          |       |            |
| PROGRAM LOCK REGISTER (A0h)                     | x8       | X                       | A0         | X    | Data |     |      |     |      |     | 5        |       |            |
|   | x16      |                         |            |      |      |     |      |     |      |     |          |       |            |
| READ LOCK REGISTER                              | x8       | X                       | Data       |      |      |     |      |     |      |     | 4, 5, 6  |       |            |
|   | x16      |                         |            |      |      |     |      |     |      |     |          |       |            |
| EXIT LOCK REGISTER (90h/00h)                    | x8       | X                       | 90         | X    | 00   |     |      |     |      |     | 3        |       |            |
|   | x16      |                         |            |      |      |     |      |     |      |     |          |       |            |
| <b>PASSWORD PROTECTION Commands</b>             |          |                         |            |      |      |     |      |     |      |     |          |       |            |
| ENTER PASSWORD PROTECTION COMMAND SET (60h)     | x8       | AAA                     | AA         | 555  | 55   | AAA | 60   |     |      |     |          |       | 3          |
|   | x16      | 555                     | AA         | 2AA  | 55   | 555 |      |     |      |     |          |       |            |
| PROGRAM PASSWORD (A0h)                          | x8       | X                       | A0         | PWAn | PWDn |     |      |     |      |     | 7        |       |            |
|   | x16      |                         |            |      |      |     |      |     |      |     |          |       |            |
| READ PASSWORD                                   | x8       | 00                      | PWD0       | 01   | PWD1 | 02  | PWD2 | 03  | PWD3 | ... | 07       | PWD7  | 4, 6, 8, 9 |
|   | x16      | 00                      | PWD0       | 01   | PWD1 | 02  | PWD2 | 03  | PWD3 |     |          |       |            |
| UNLOCK PASSWORD (25h/03h)                       | x8       | 00                      | 25         | 00   | 03   | 00  | PWD0 | 01  | PWD1 | ... | 00       | 29    | 8, 10      |
|   | x16      |                         |            |      |      |     |      |     |      |     |          |       |            |
| EXIT PASSWORD PROTECTION (90h/00h)              | x8       | X                       | 90         | X    | 00   |     |      |     |      |     | 3        |       |            |
|   | x16      |                         |            |      |      |     |      |     |      |     |          |       |            |
| <b>NONVOLATILE PROTECTION Commands</b>          |          |                         |            |      |      |     |      |     |      |     |          |       |            |
| ENTER NONVOLATILE PROTECTION COMMAND SET (C0h)  | x8       | AAA                     | AA         | 555  | 55   | AAA | C0   |     |      |     |          |       | 3          |
|   | x16      | 555                     | AA         | 2AA  | 55   | 555 |      |     |      |     |          |       |            |
| PROGRAM NONVOLATILE PROTECTION BIT (A0h)        | x8       | X                       | A0         | BAd  | 00   |     |      |     |      |     | 11       |       |            |
|   | x16      |                         |            |      |      |     |      |     |      |     |          |       |            |
| READ NONVOLATILE PROTECTION BIT STATUS          | x8       | BAd                     | READ (DQ0) |      |      |     |      |     |      |     | 4, 6, 11 |       |            |
|   | x16      |                         |            |      |      |     |      |     |      |     |          |       |            |
| CLEAR ALL NONVOLATILE PROTECTION BITS (80h/30h) | x8       | X                       | 80         | 00   | 30   |     |      |     |      |     | 12       |       |            |
|   | x16      |                         |            |      |      |     |      |     |      |     |          |       |            |



## 256Mb: x8/x16, 3V, MT28EW Embedded Parallel NOR Block Protection Command Definitions – Address-Data Cycles

**Table 16: Block Protection Command Definitions – Address-Data Cycles, 8-Bit and 16-Bit (Continued)**

Notes 1 and 2 apply to entire table

| Command and Code/Subcode                                    | Bus Size | Address and Data Cycles |            |     |    |     |    |              |      |     |          | Notes |   |
|---|----------|-------------------------|------------|-----|----|-----|----|--------------|------|-----|----------|-------|---|
|   |          | 1st                     |            | 2nd |    | 3rd |    | 4th          |      | ... | nth      |       |   |
|   |          | A                       | D          | A   | D  | A   | D  | A            | D    |     | A        |       | D |
| EXIT NONVOLATILE PROTECTION (90h/00h)                       | x8       | X                       | 90         | X   | 00 |     |    |              |      |     | 3        |       |   |
|   | x16      |                         |            |     |    |     |    |              |      |     |          |       |   |
| <b>NONVOLATILE PROTECTION BIT LOCK BIT Commands</b>         |          |                         |            |     |    |     |    |              |      |     |          |       |   |
| ENTER NONVOLATILE PROTECTION BIT LOCK BIT COMMAND SET (50h) | x8       | AAA                     | AA         | 555 | 55 | AAA | 50 |              |      |     | 3        |       |   |
|   | x16      | 555                     | AA         | 2AA | 55 | 555 |    |              |      |     |          |       |   |
| PROGRAM NONVOLATILE PROTECTION BIT LOCK BIT (A0h)           | x8       | X                       | A0         | X   | 00 |     |    |              |      |     | 11       |       |   |
|   | x16      |                         |            |     |    |     |    |              |      |     |          |       |   |
| READ NONVOLATILE PROTECTION BIT LOCK BIT STATUS             | x8       | X                       | READ (DQ0) |     |    |     |    |              |      |     | 4, 6, 11 |       |   |
|   | x16      |                         |            |     |    |     |    |              |      |     |          |       |   |
| EXIT NONVOLATILE PROTECTION BIT LOCK BIT (90h/00h)          | x8       | X                       | 90         | X   | 00 |     |    |              |      |     | 3        |       |   |
|   | x16      |                         |            |     |    |     |    |              |      |     |          |       |   |
| <b>VOLATILE PROTECTION Commands</b>                         |          |                         |            |     |    |     |    |              |      |     |          |       |   |
| ENTER VOLATILE PROTECTION COMMAND SET (E0h)                 | x8       | AAA                     | AA         | 555 | 55 | AAA | E0 |              |      |     | 3        |       |   |
|   | x16      | 555                     | AA         | 2AA | 55 | 555 |    |              |      |     |          |       |   |
| PROGRAM VOLATILE PROTECTION BIT (A0h)                       | x8       | X                       | A0         | BAd | 00 |     |    |              |      |     | 11       |       |   |
|   | x16      |                         |            |     |    |     |    |              |      |     |          |       |   |
| READ VOLATILE PROTECTION BIT STATUS                         | x8       | BAd                     | READ (DQ0) |     |    |     |    |              |      |     | 4, 6     |       |   |
|   | x16      |                         |            |     |    |     |    |              |      |     |          |       |   |
| CLEAR VOLATILE PROTECTION BIT (A0h)                         | x8       | X                       | A0         | BAd | 01 |     |    |              |      |     | 11       |       |   |
|   | x16      |                         |            |     |    |     |    |              |      |     |          |       |   |
| EXIT VOLATILE PROTECTION (90h/00h)                          | x8       | X                       | 90         | X   | 00 |     |    |              |      |     | 3        |       |   |
|   | x16      |                         |            |     |    |     |    |              |      |     |          |       |   |
| <b>EXTENDED MEMORY BLOCK Operations</b>                     |          |                         |            |     |    |     |    |              |      |     |          |       |   |
| ENTER EXTENDED MEMORY BLOCK (88h)                           | x8       | AAA                     | AA         | 555 | 55 | AAA | 88 |              |      |     |          |       |   |
|   | x16      | 555                     |            | 2AA |    | 555 |    |              |      |     |          |       |   |
| PROGRAM EXTENDED MEMORY BLOCK (A0h)                         | x8       | AAA                     | AA         | 555 | 55 | AAA | A0 | Word address | data |     |          |       |   |
|   | x16      | 555                     |            | 2AA |    | 555 |    |              |      |     |          |       |   |
| READ EXTENDED MEMORY BLOCK                                  | x8       | Word address            | data       |     |    |     |    |              |      |     |          |       |   |
|   | x16      |                         |            |     |    |     |    |              |      |     |          |       |   |



## 256Mb: x8/x16, 3V, MT28EW Embedded Parallel NOR Block Protection Command Definitions – Address-Data Cycles

**Table 16: Block Protection Command Definitions – Address-Data Cycles, 8-Bit and 16-Bit (Continued)**

Notes 1 and 2 apply to entire table

| Command and Code/Subcode             | Bus Size | Address and Data Cycles |    |     |    |     |    |     |    |     |     | Notes |   |
|--------------------------------------|----------|-------------------------|----|-----|----|-----|----|-----|----|-----|-----|-------|---|
|                                      |          | 1st                     |    | 2nd |    | 3rd |    | 4th |    | ... | nth |       |   |
|                                      |          | A                       | D  | A   | D  | A   | D  | A   | D  |     | A   |       | D |
| EXIT EXTENDED MEMORY BLOCK (90h/00h) | x8       | AAA                     | AA | 555 | 55 | 555 | 90 | X   | 00 |     |     |       |   |
|                                      | x16      | 555                     |    | 2AA |    |     |    |     |    |     |     |       |   |

- Notes:
- Key: A = Address and D = Data; X = "Don't Care;" BA# = Any address in the block; PWDn = Password bytes, n = 0 to 7 (x8)/words 0 to 3 (x16); PWA# = Password address, n = 0 to 7 (x8)/0 to 3 (x16); PWDn = Password words, n = 0 to 3 (x16); PWA# = Password address, n = 0 to 3(x16); Gray = Not applicable. All values in the table are hexadecimal.
  - DQ[15:8] are "Don't Care" during UNLOCK and COMMAND cycles. A[16:MAX] are "Don't Care" during UNLOCK and COMMAND cycles, unless an address is required.
  - The ENTER command sequence must be issued prior to any operation. It disables READ and WRITE operations from and to block 0. READ and WRITE operations from and to any other block are allowed. Also, when an ENTER COMMAND SET command is issued, an EXIT COMMAND SET command must be issued to return the device to READ mode.
  - READ REGISTER/PASSWORD commands have no command code; CE# and OE# are driven LOW and data is read according to a specified address.
  - Data = Lock register content.
  - All address cycles shown for this command are READ cycles.
  - Only one portion of the password can be programmed or read by each PROGRAM PASSWORD command.
  - Each portion of the password can be entered or read in any order as long as the entire 64-bit password is entered or read.
  - For the x8 READ PASSWORD command, the nth (and final) address cycle equals the 8th address cycle. From the 5th to the 8th address cycle, the values for each address and data pair continue the pattern shown in the table as follows: for x8, address and data = 04 and PWD4; 05 and PWD5; 06 and PWD6; 07 and PWD7.
  - For the x8 UNLOCK PASSWORD command, the nth (and final) address cycle equals the 11th address cycle. From the 5th to the 10th address cycle, the values for each address and data pair continue the pattern shown in the table as follows: address and data = 02 and PWD2; 03 and PWD3; 04 and PWD4; 05 and PWD5; 06 and PWD6; 07 and PWD7.  
  
For the x16 UNLOCK PASSWORD command, the nth (and final) address cycle equals the 7th address cycle. For the 5th and 6th address cycles, the values for the address and data pair continue the pattern shown in the table as follows: address and data = 02 and PWD2; 03 and PWD3.
  - Both nonvolatile and volatile protection bit settings are as follows: Protected state = 00; Unprotected state = 01.
  - The CLEAR ALL NONVOLATILE PROTECTION BITS command programs all nonvolatile protection bits before erasure. This prevents over-erasure of previously cleared nonvolatile protection bits.



## Protection Operations

Blocks can be protected individually against accidental PROGRAM or ERASE operations on both 8-bit and 16-bit configurations. The block protection scheme is shown in the Software Protection Scheme figure. Memory block and extended memory block protection is configured through the lock register.

### LOCK REGISTER Commands

After the ENTER LOCK REGISTER COMMAND SET (40h) command has been issued, all bus READ or PROGRAM operations can be issued to the lock register.

The PROGRAM LOCK REGISTER (A0h) command allows the lock register to be configured. The programmed data can then be checked with a READ LOCK REGISTER command by driving CE# and OE# LOW with the appropriate address data on the address bus.

### PASSWORD PROTECTION Commands

After the ENTER PASSWORD PROTECTION COMMAND SET (60h) command has been issued, the commands related to password protection mode can be issued to the device.

The PROGRAM PASSWORD (A0h) command is used to program the 64-bit password used in the password protection mode. To program the 64-bit password, the complete command sequence must be entered eight times at eight consecutive addresses selected by A[1:0] plus DQ15/A-1 in 8-bit mode, or four times at four consecutive addresses selected by A[1:0] in 16-bit mode. By default, all password bits are set to 1. The password can be checked by issuing a READ PASSWORD command.

**Note:** A password must be programmed per Flash memory die to enable password protection.

The READ PASSWORD command is used to verify the password used in password protection mode. To verify the 64-bit password, the complete command sequence must be entered eight times at eight consecutive addresses selected by A[1:0] plus DQ15/A-1 in 8-bit mode, or four times at four consecutive addresses selected by A[1:0] in 16-bit mode. If the password mode lock bit is programmed and the user attempts to read the password, the device will output 00h onto the I/O data bus.

The UNLOCK PASSWORD (25/03h) command is used to clear the nonvolatile protection bit lock bit, allowing the nonvolatile protection bits to be modified. The UNLOCK PASSWORD command must be issued, along with the correct password, and requires a 6 $\mu$ s delay between successive UNLOCK PASSWORD commands in order to prevent hackers from cracking the password by trying all possible 64-bit combinations. If this delay does not occur, the latest command will be ignored. Approximately 6 $\mu$ s is required for unlocking the device after the valid 64-bit password has been provided.

### NONVOLATILE PROTECTION Commands

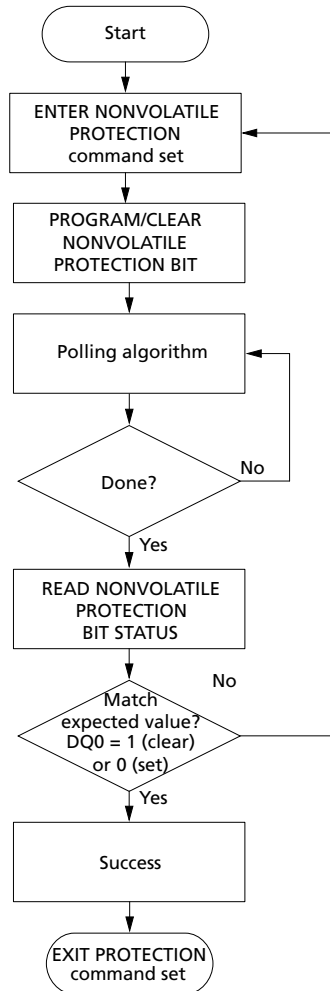
After the ENTER NONVOLATILE PROTECTION COMMAND SET (C0h) command has been issued, the commands related to nonvolatile protection mode can be issued to the device.

A block can be protected from program or erase by issuing a PROGRAM NONVOLATILE PROTECTION BIT (A0h) command, along with the block address. This command sets the nonvolatile protection bit to 0 for a given block.

The status of a nonvolatile protection bit for a given block or group of blocks can be read by issuing a READ NONVOLATILE MODIFY PROTECTION BIT command, along with the block address.

The nonvolatile protection bits are erased simultaneously by issuing a CLEAR ALL NONVOLATILE PROTECTION BITS (80/30h) command. No specific block address is required. If the nonvolatile protection bit lock bit is set to 0, the command fails.

**Figure 13: Set/Clear Nonvolatile Protection Bit Algorithm Flowchart**



- Notes:
1. See the Block Protection Command Definitions table for address-data cycle details.
  2. DQ5 and DQ1 are ignored in this algorithm flow.

### NONVOLATILE PROTECTION BIT LOCK BIT Commands

After the ENTER NONVOLATILE PROTECTION BIT LOCK BIT COMMAND SET (50h) command has been issued, the commands that allow the nonvolatile protection bit lock bit to be set can be issued to the device.

The PROGRAM NONVOLATILE PROTECTION BIT LOCK BIT (A0h) command is used to set the nonvolatile protection bit lock bit to 0, thus locking the nonvolatile protection bits and preventing them from being modified.

The READ NONVOLATILE PROTECTION BIT LOCK BIT STATUS command is used to read the status of the nonvolatile protection bit lock bit.

## VOLATILE PROTECTION Commands

After the ENTER VOLATILE PROTECTION COMMAND SET (E0h) command has been issued, commands related to the volatile protection mode can be issued to the device.

The PROGRAM VOLATILE PROTECTION BIT (A0h) command individually sets a volatile protection bit to 0 for a given block. If the nonvolatile protection bit for the same block is set, the block is locked regardless of the value of the volatile protection bit (see the Block Protection Status table).

The status of a volatile protection bit for a given block can be read by issuing a READ VOLATILE PROTECTION BIT STATUS command along with the block address.

The CLEAR VOLATILE PROTECTION BIT (A0h) command individually clears (sets to 1) the volatile protection bit for a given block. If the nonvolatile protection bit for the same block is set, the block is locked regardless of the value of the volatile protection bit (see the Block Protection Status table).

## EXTENDED MEMORY BLOCK Commands

The device has one extra 128-word extended memory block that can be accessed only by the ENTER EXTENDED MEMORY BLOCK (88h) command. The extended memory block is 128 words (x16) or 256 bytes (x8). It is used as a security block to provide a permanent 128-bit secure ID number or to store additional information. The device can be shipped with the extended memory block prelocked permanently by Micron, including the 128-bit security identification number. Or, the device can be shipped with the extended memory block unlocked, enabling customers to permanently program and lock it (default). (See Lock Register, the AUTO SELECT command, and the Block Protection table).

**Table 17: Extended Memory Block Address and Data**

| Address         |                 | Data                      |                                  |                        |
|-----------------|-----------------|---------------------------|----------------------------------|------------------------|
| x8              | x16             | Micron prelocked          | Customer Lockable                |                        |
| 000000h–00000Fh | 000000h–000007h | Secure ID number          | Determined by customer (default) | Secure ID number       |
| 000010h–0000FFh | 000008h–00007Fh | Protected and unavailable |                                  | Determined by customer |

After the ENTER EXTENDED MEMORY BLOCK command has been issued, the device enters the extended memory block mode. All bus READ or PROGRAM operations are conducted on the extended memory block, and the extended memory block is addressed using the addresses occupied by block 0 in the other operating modes (see the Memory Map table).

In extended memory block mode, ERASE, CHIP ERASE, ERASE SUSPEND, and ERASE RESUME commands are not allowed. The extended memory block cannot be erased, and each bit of the extended memory block can only be programmed once.

The extended memory block is protected from further modification by programming lock register bit 0. Once invoked, this protection cannot be undone.

The device remains in extended memory block mode until the EXIT EXTENDED MEMORY BLOCK (90/00h) command is issued, which returns the device to read mode, or until power is removed from the device. After a power-up sequence or hardware reset, the device will revert to reading memory blocks in the main array.

### EXIT PROTECTION Command

The EXIT PROTECTION COMMAND SET (90/00h) command is used to exit the lock register, password protection, nonvolatile protection, volatile protection, and nonvolatile protection bit lock bit command set modes and return the device to read mode.

## Common Flash Interface

The common Flash interface (CFI) is a JEDEC-approved, standardized data structure that can be read from the Flash memory device. It allows a system's software to query the device to determine various electrical and timing parameters, density information, and functions supported by the memory. The system can interface easily with the device, enabling the software to upgrade itself when necessary.

When the READ CFI command is issued, the device enters CFI query mode and the data structure is read from memory. The following tables show the addresses (A[7:0], A-1) used to retrieve the data. The query data is always presented on the lowest order data outputs (DQ[7:0]), and the other data outputs (DQ[15:8]) are set to 0.

**Table 18: Query Structure Overview**

Note 1 applies to the entire table

| Address |     | Subsection Name                                 | Description   |
|---------|-----|---|---|
| x16     | x8  |   |   |
| 10h     | 20h | CFI query identification string                 | Command set ID and algorithm data offset                            |
| 1Bh     | 36h | System interface information                    | Device timing and voltage information                               |
| 27h     | 4Eh | Device geometry definition                      | Flash device layout   |
| 40h     | 80h | Primary algorithm-specific extended query table | Additional information specific to the primary algorithm (optional) |

Note: 1. Query data are always presented on the lowest order data outputs (DQ[7:0]). DQ[15:8] are set to 0.

**Table 19: CFI Query Identification String**

Note 1 applies to the entire table

| Address |     | Data  | Description  | Value   |
|---------|-----|-------|--|---------|
| x16     | x8  |       |  |         |
| 10h     | 20h | 0051h | Query unique ASCII string "QRY"  | "Q"     |
| 11h     | 22h | 0052h |  | "R"     |
| 12h     | 24h | 0059h |  | "Y"     |
| 13h     | 26h | 0002h | Primary algorithm command set and control interface ID code 16-bit ID code defining a specific algorithm     | –       |
| 14h     | 28h | 0000h |  |         |
| 15h     | 2Ah | 0040h | Address for primary algorithm extended query table (see the Primary Algorithm-Specific Extended Query Table) | P = 40h |
| 16h     | 2Ch | 0000h |  |         |
| 17h     | 2Eh | 0000h | Alternate vendor command set and control interface ID code second vendor-specified algorithm supported       | –       |
| 18h     | 30h | 0000h |  |         |
| 19h     | 32h | 0000h | Address for alternate algorithm extended query table   | –       |
| 1Ah     | 34h | 0000h |  |         |

Note: 1. Query data are always presented on the lowest order data outputs (DQ[7:0]). DQ[15:8] are set to 0.

**Table 20: CFI Query System Interface Information**

Note 1 applies to the entire table

| Address |     | Data  | Description  | Value  |
|---------|-----|-------|--|--------|
| x16     | x8  |       |  |        |
| 1Bh     | 36h | 0027h | V <sub>CC</sub> logic supply minimum program/erase voltage<br>Bits[7:4] BCD value in volts<br>Bits[3:0] BCD value in 100mV         | 2.7V   |
| 1Ch     | 38h | 0036h | V <sub>CC</sub> logic supply maximum program/erase voltage<br>Bits[7:4] BCD value in volts<br>Bits[3:0] BCD value in 100mV         | 3.6V   |
| 1Dh     | 3Ah | 0085h | V <sub>HH</sub> (programming) supply minimum program/erase voltage<br>Bits[7:4] hex value in volts<br>Bits[3:0] BCD value in 100mV | 8.5V   |
| 1Eh     | 3Ch | 0095h | V <sub>HH</sub> (programming) supply maximum program/erase voltage<br>Bits[7:4] hex value in volts<br>Bits[3:0] BCD value in 10mV  | 9.5V   |
| 1Fh     | 3Eh | 0005h | Typical timeout for single byte/word program = 2 <sup>n</sup> μs   | 32μs   |
| 20h     | 40h | 0009h | Typical timeout for maximum size buffer program = 2 <sup>n</sup> μs  | 512μs  |
| 21h     | 42h | 0008h | Typical timeout per individual block erase = 2 <sup>n</sup> ms   | 256ms  |
| 22h     | 44h | 0010h | Typical timeout for full chip erase = 2 <sup>n</sup> ms  | 66s    |
| 23h     | 46h | 0003h | Maximum timeout for byte/word program = 2 <sup>n</sup> times typical   | 256μs  |
| 24h     | 48h | 0002h | Maximum timeout for buffer program = 2 <sup>n</sup> times typical  | 2048μs |
| 25h     | 4Ah | 0003h | Maximum timeout per individual block erase = 2 <sup>n</sup> times typical  | 2s     |
| 26h     | 4Ch | 0003h | Maximum timeout for chip erase = 2 <sup>n</sup> times typical  | 528s   |

Note: 1. The values in this table are valid for both packages.

**Table 21: Device Geometry Definition**

| Address |     | Data  | Description  | Value                   |
|---------|-----|-------|--|-------------------------|
| x16     | x8  |       |  |                         |
| 27h     | 4Eh | 0019h | Device size = 2 <sup>n</sup> in number of bytes  | 32MB                    |
| 28h     | 50h | 0002h | Flash device interface code description  | x8, x16<br>asynchronous |
| 29h     | 52h | 0000h |  |                         |
| 2Ah     | 54h | 000xh | Maximum number of bytes in multi-byte program or page = 2 <sup>n</sup><br>X16 mode: 000Ah<br>X8 mode: 08h              | 1024 (x16)<br>256 (x8)  |
| 2Bh     | 56h | 0000h |  |                         |
| 2Ch     | 58h | 0001h | Number of erase block regions. It specifies the number of regions containing contiguous erase blocks of the same size. | 1                       |
| 2Dh     | 5Ah | 00FFh | Erase block region 1 information   | 256                     |
| 2Eh     | 5Ch | 0000h | Number of identical-size erase blocks = 00FFh + 1  |                         |
| 2Fh     | 5Eh | 0000h | Erase block region 1 information   | 128KB                   |
| 30h     | 60h | 0002h | Block size in region 1 = 0200h × 256 bytes   |                         |

**Table 21: Device Geometry Definition (Continued)**

| Address |     | Data  | Description                      | Value |
|---------|-----|-------|----------------------------------|-------|
| x16     | x8  |       |                                  |       |
| 31h     | 62h | 0000h | Erase block region 2 information | 0     |
| 32h     | 64h | 0000h |                                  |       |
| 33h     | 66h | 0000h |                                  |       |
| 34h     | 68h | 0000h |                                  |       |
| 35h     | 6Ah | 0000h | Erase block region 3 information | 0     |
| 36h     | 6Ch | 0000h |                                  |       |
| 37h     | 6Eh | 0000h |                                  |       |
| 38h     | 70h | 0000h |                                  |       |
| 39h     | 72h | 0000h | Erase block region 4 information | 0     |
| 3Ah     | 74h | 0000h |                                  |       |
| 3Bh     | 76h | 0000h |                                  |       |
| 3Ch     | 78h | 0000h |                                  |       |

**Table 22: Primary Algorithm-Specific Extended Query Table**

Note 1 applies to the entire table

| Address |     | Data  | Description  | Value         |
|---------|-----|-------|--|---------------|
| x16     | x8  |       |  |               |
| 40h     | 80h | 0050h | Primary algorithm extended query table unique ASCII string "PRI"   | "P"           |
| 41h     | 82h | 0052h |  | "R"           |
| 42h     | 84h | 0049h |  | "I"           |
| 43h     | 86h | 0031h | Major version number, ASCII  | "1"           |
| 44h     | 88h | 0033h | Minor version number, ASCII  | "3"           |
| 45h     | 8Ah | 001Ch | Address sensitive unlock (bits[1:0]):<br>00 = Required<br>01 = Not required<br>Process technology (bits [7:2])<br>0111b: 2nd Generation<br>0110b: 1st Generation | Required      |
| 46h     | 8Ch | 0002h | Erase suspend:<br>00 = Not supported<br>01 = Read only<br>02 = Read and write  | 2             |
| 47h     | 8Eh | 0001h | Block protection:<br>00 = Not supported<br>x = Number of blocks per group  | 1             |
| 48h     | 90h | 0000h | Temporary block unprotect scheme:<br>00 = Not supported<br>01 = Supported  | Not supported |
| 49h     | 92h | 0008h | Protect/unprotect scheme:<br>08 = Advanced sector protection method  | 8             |

**Table 22: Primary Algorithm-Specific Extended Query Table (Continued)**

Note 1 applies to the entire table

| Address |     | Data  | Description  | Value  |
|---------|-----|-------|--|--|
| x16     | x8  |       |  |  |
| 4Ah     | 94h | 0000h | Simultaneous operations:<br>Not supported  | –  |
| 4Bh     | 96h | 0000h | Burst mode:<br>00 = Not supported<br>01 = Supported  | Not supported  |
| 4Ch     | 98h | 0003h | Page mode:<br>00 = Not supported<br>01 = 4-word page<br>02 = 8-word page<br>03 = 16-word page  | 16-word page   |
| 4Dh     | 9Ah | 0085h | V <sub>HH</sub> supply minimum program/erase voltage:<br>Bits[7:4] hex value in volts<br>Bits[3:0] BCD value in 100mV                    | 8.5V   |
| 4Eh     | 9Ch | 0095h | V <sub>HH</sub> supply maximum program/erase voltage:<br>Bits[7:4] hex value in volts<br>Bits[3:0] BCD value in 100mV                    | 9.5V   |
| 4Fh     | 9Eh | 00xxh | WP# protection:<br>xx = 04h: Uniform device, HW protection for lowest block<br>xx = 05h: Uniform device, HW protection for highest block | Uniform +<br>V <sub>pp</sub> /WP# protect-<br>ing highest or<br>lowest block |
| 50h     | A0h | 0001h | Program suspend:<br>00 = Not supported<br>01 = Supported   | Supported  |

Note: 1. The values in this table are valid for both packages.



## Power-Up and Reset Characteristics

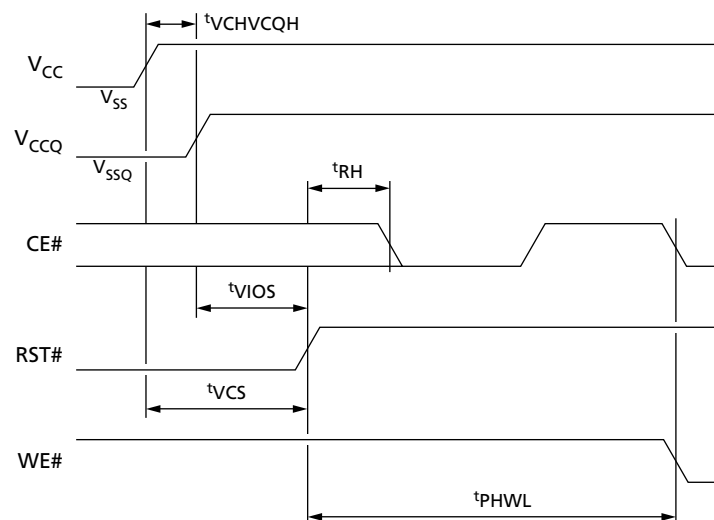
**Table 23: Power-Up Specifications**

Note 1 applies to entire table.

| Parameter                                     | Symbol            |                      | Min | Unit | Notes |
|---|-------------------|----------------------|-----|------|-------|
|   | Legacy            | JEDEC                |     |      |       |
| V <sub>CC</sub> HIGH to V <sub>CCQ</sub> HIGH | –                 | t <sub>VCHVCQH</sub> | 0   | μs   | 2     |
| V <sub>CC</sub> HIGH to rising edge of RST#   | t <sub>VCS</sub>  | t <sub>VCHPH</sub>   | 300 | μs   | 3, 4  |
| V <sub>CCQ</sub> HIGH to rising edge of RST#  | t <sub>VIOS</sub> | t <sub>VCQHPH</sub>  | 0   | μs   | 3, 4  |
| RST# HIGH to chip enable LOW                  | t <sub>RH</sub>   | t <sub>PHEL</sub>    | 50  | ns   |       |
| RST# HIGH to write enable LOW                 | –                 | t <sub>PHWL</sub>    | 150 | ns   |       |

- Notes:
1. Sampled only; not 100% tested.
  2. V<sub>CC</sub> should attain V<sub>CC,min</sub> from V<sub>SS</sub> simultaneously with or prior to applying V<sub>CCQ</sub> during power up. V<sub>CC</sub> should attain V<sub>SS</sub> during power down.
  3. If RST# is not stable for t<sub>VCS</sub> or t<sub>VIOS</sub>, the device will not allow any READ or WRITE operations, and a hardware reset is required.
  4. Power supply transitions should only occur when RST# is LOW.

**Figure 14: Power-Up Timing**

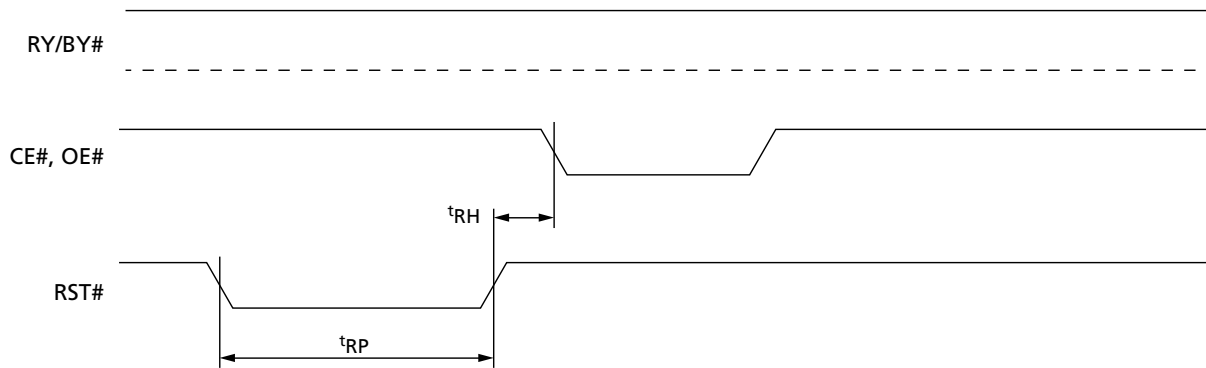


**Table 24: Reset AC Specifications**

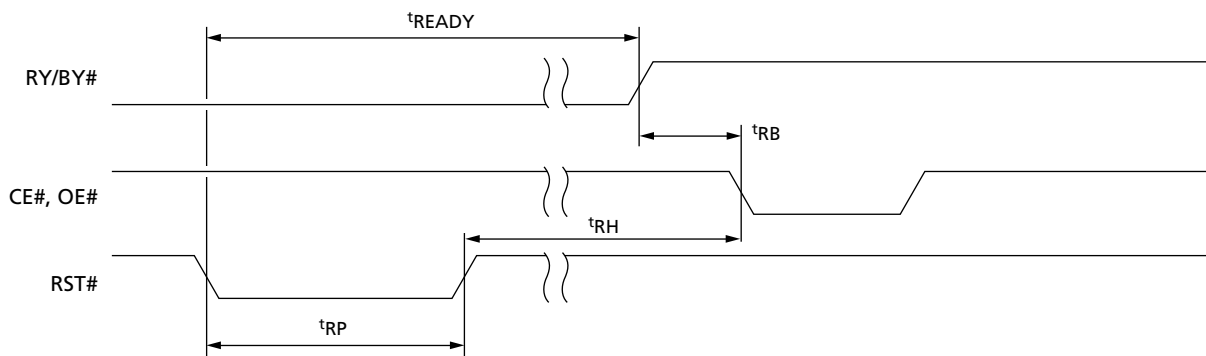
| Condition/Parameter                              | Symbol             |                                    | Min | Max | Unit          | Notes |
|--|--------------------|------------------------------------|-----|-----|---------------|-------|
|  | Legacy             | JEDEC                              |     |     |               |       |
| RST# LOW to read mode during program or erase    | $t_{\text{READY}}$ | $t_{\text{PLRH}}$                  | –   | 25  | $\mu\text{s}$ | 1     |
| RST# pulse width                                 | $t_{\text{RP}}$    | $t_{\text{PLPH}}$                  | 100 | –   | ns            |       |
| RST# HIGH to CE# LOW, OE# LOW                    | $t_{\text{RH}}$    | $t_{\text{PHEL}}, t_{\text{PHGL}}$ | 50  | –   | ns            | 1     |
| RST# LOW to standby mode during read mode        | $t_{\text{RPD}}$   | –                                  | 0   | –   | $\mu\text{s}$ |       |
| RST# LOW to standby mode during program or erase |                    |                                    | 0   | –   | $\mu\text{s}$ |       |
| RY/BY# HIGH to CE# LOW, OE# LOW                  | $t_{\text{RB}}$    | $t_{\text{RHEL}}, t_{\text{RHGL}}$ | 0   | –   | ns            | 1     |

Note: 1. Sampled only; not 100% tested.

**Figure 15: Reset AC Timing – No PROGRAM/ERASE Operation in Progress**



**Figure 16: Reset AC Timing During PROGRAM/ERASE Operation**



## Absolute Ratings and Operating Conditions

Stresses greater than those listed 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 outside those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may adversely affect reliability.

**Table 25: Absolute Maximum/Minimum Ratings**

| Parameter                   | Symbol     | Min  | Max           | Unit | Notes |
|-----------------------------|------------|------|---------------|------|-------|
| Temperature under bias      | $T_{BIAS}$ | -50  | 125           | °C   |       |
| Storage temperature         | $T_{STG}$  | -65  | 150           | °C   |       |
| Supply voltage              | $V_{CC}$   | -0.6 | $V_{CC} + 2$  | V    | 1, 2  |
| Input/output supply voltage | $V_{CCQ}$  | -0.6 | $V_{CCQ} + 2$ | V    | 1, 2  |
| Program/erase voltage       | $V_{PP}$   | -0.6 | 9.5           | V    | 3     |

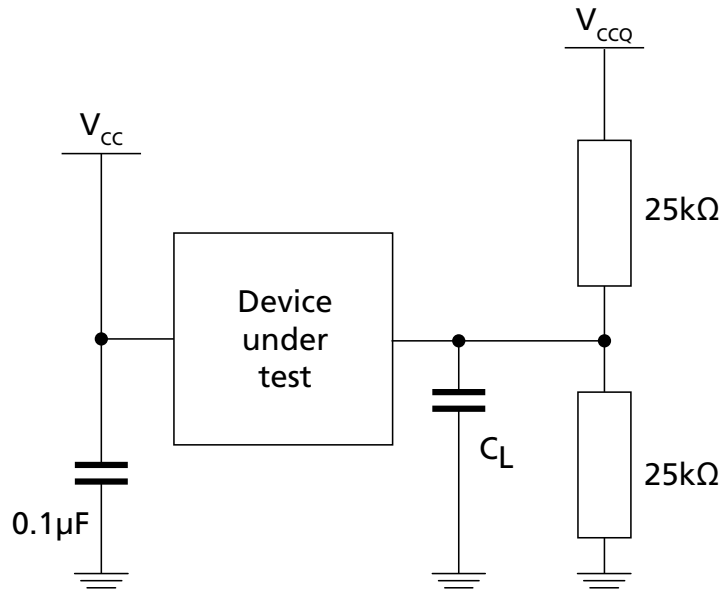
- Notes:
1. During signal transitions, minimum voltage may undershoot to  $-2V$  for periods less than 20ns.
  2. During signal transitions, maximum voltage may overshoot to  $V_{CC} + 2V$  for periods less than 20ns.
  3.  $V_{PP}$  must not remain at 9.5V for more than 80 hours cumulative.

**Table 26: Operating Conditions**

| Parameter   | Symbol    | Min            | Max | Unit | Notes |
|---|-----------|----------------|-----|------|-------|
| Supply voltage  | $V_{CC}$  | 2.7            | 3.6 | V    |       |
| Input/output supply voltage ( $V_{CCQ} \leq V_{CC}$ ) | $V_{CCQ}$ | 1.65           | 3.6 | V    |       |
| Accelerated buffered program/chip erase voltage       | $V_{HH}$  | 8.5            | 9.5 | V    |       |
| Ambient operating temperature                         | $T_A$     | -40            | 85  | °C   |       |
| Load capacitance                                      | $C_L$     | 30             |     | pF   |       |
| Input rise and fall times ( $V_{IL}$ to $V_{IH}$ )    | –         | 0.3            | 2.5 | ns   | 1, 2  |
| Input pulse voltages                                  | –         | 0 to $V_{CCQ}$ |     | V    |       |
| Input and output timing reference voltages            | –         | $V_{CCQ}/2$    |     | V    |       |
| Address to address skew                               | –         | –              | 3   | ns   |       |

- Notes:
1. If the rise/fall time is slower than 2.5ns, all timing specs must be derated by 0.5ns for every nanosecond push-out in rise/fall time. (Example: for a 10ns rise/fall time, all timing specs must be derated by  $(10 - 2.5) \times (0.5ns) = 3.75ns$ .)
  2. Applies to Address, CE#, OE#, and WE# signals.

**Figure 17: AC Measurement Load Circuit**



Note: 1.  $C_L$  includes jig capacitance.

**Figure 18: AC Measurement I/O Waveform**



**Table 27: Input/Output Capacitance**

| Parameter          | Symbol    | Test Condition | Min | Max | Unit |
|--------------------|-----------|----------------|-----|-----|------|
| Input capacitance  | $C_{IN}$  | $V_{IN} = 0V$  | 3   | 11  | pF   |
| Output capacitance | $C_{OUT}$ | $V_{OUT} = 0V$ | 3   | 7   | pF   |

## DC Characteristics

**Table 28: DC Current Characteristics**

| Parameter                                  | Symbol                    | Conditions  | Min                                | Typ       | Max     | Unit        | Notes       |   |
|--|---------------------------|---|------------------------------------|-----------|---------|-------------|-------------|---|
| Input load current                         | $I_{LI}$                  | $0V \leq V_{IN} \leq V_{CC}$  | –                                  | –         | $\pm 1$ | $\mu A$     | 1           |   |
| Output leakage current                     | $I_{LO}$                  | $0V \leq V_{OUT} \leq V_{CC}$                                       | –                                  | –         | $\pm 1$ | $\mu A$     |             |   |
| $V_{CC}$ read current                      | Random read               | $I_{CC1}$<br>$CE\# = V_{IL}, OE\# = V_{IH},$<br>$f = 5 \text{ MHz}$ | –                                  | 26        | 31      | $\text{mA}$ |             |   |
|  | Page read                 |   | –                                  | 12        | 16      | $\text{mA}$ |             |   |
| $V_{CC}$ standby current (256Mb)           | $I_{CC2}$                 | $CE\# = V_{CCQ} \pm 0.2V,$<br>$RST\# = V_{CCQ} \pm 0.2V$            | –                                  | 65        | 135     | $\mu A$     |             |   |
| $V_{CC}$ program/erase/blank check current |                           | Program/erase controller active                                     | $V_{pp}/WP\# = V_{IL}$ or $V_{IH}$ | –         | 35      | 50          | $\text{mA}$ | 2 |
|  |                           |   | $V_{pp}/WP\# = V_{HH}$             | –         | 35      | 50          | $\text{mA}$ |   |
| $V_{pp}$ current                           | Read                      | $I_{PP1}$   | $V_{pp}/WP\# \leq V_{CC}$          | –         | 2       | 15          | $\mu A$     |   |
|  | Standby                   |   |                                    | $I_{PP2}$ | –       | 0.2         | 5           |   |
|  | PROGRAM operation ongoing | $I_{PP3}$   | $V_{pp}/WP\# = V_{HH}$             | –         | 5       | 10          | $\text{mA}$ |   |
|  |                           |   | $V_{pp}/WP\# = V_{CC}$             | –         | 0.05    | 0.10        | $\text{mA}$ |   |
|  | ERASE operation ongoing   | $I_{PP4}$   | $V_{pp}/WP\# = V_{HH}$             | –         | 5       | 10          | $\text{mA}$ |   |
| $V_{pp}/WP\# = V_{CC}$                     |                           |   | –                                  | 0.05      | 0.10    | $\text{mA}$ |             |   |

- Notes: 1. The maximum input load current is  $\pm 5\mu A$  on the  $V_{pp}/WP\#$  pin.  
2. Sampled only; not 100% tested.

**Table 29: DC Voltage Characteristics**

| Parameter                                      | Symbol    | Conditions  | Min           | Typ | Max             | Unit | Notes |
|--|-----------|---|---------------|-----|-----------------|------|-------|
| Input LOW voltage                              | $V_{IL}$  | $V_{CC} \geq 2.7V$  | -0.5          | -   | 0.8             | V    |       |
| Input HIGH voltage                             | $V_{IH}$  | $V_{CC} \geq 2.7V$  | $0.7V_{CCQ}$  | -   | $V_{CCQ} + 0.4$ | V    |       |
| Output LOW voltage                             | $V_{OL}$  | $I_{OL} = 100\mu A,$<br>$V_{CC} = V_{CC,min},$<br>$V_{CCQ} = V_{CCQ,min}$ | -             | -   | $0.15V_{CCQ}$   | V    |       |
| Output HIGH voltage                            | $V_{OH}$  | $I_{OH} = 100\mu A,$<br>$V_{CC} = V_{CC,min},$<br>$V_{CCQ} = V_{CCQ,min}$ | $0.85V_{CCQ}$ | -   | -               | V    |       |
| Voltage for $V_{pp}/WP\#$ program acceleration | $V_{pp}$  | -   | 8.5           | -   | 9.5             | V    | 1     |
| Program/erase lockout supply voltage           | $V_{LKO}$ | -   | 2.0           | -   | -               | V    | 2, 3  |

- Notes:
1.  $V_{pp}$  must not remain at 9.5V for more than 80 hours cumulative.
  2. Sampled only; not 100% tested.
  3. WRITE operations are not valid when  $V_{CC}$  supply drops below  $V_{LKO}$ .



## Read AC Characteristics

**Table 30: Read AC Characteristics –  $V_{CC} = V_{CCQ} = 2.7\text{-}3.6\text{V}$**

| Parameter  | Symbol     |  | Condition                            | Min | Max | Unit          | Notes |
|--|------------|--|--------------------------------------|-----|-----|---------------|-------|
|  | Legacy     | JEDEC                                      |                                      |     |     |               |       |
| Address valid to next address valid                            | $t_{RC}$   | $t_{AVAV}$                                 | $CE\# = V_{IL}$ ,<br>$OE\# = V_{IL}$ | 70  | –   | ns            |       |
| Address valid to output valid                                  | $t_{ACC}$  | $t_{AVQV}$                                 | $CE\# = V_{IL}$ ,<br>$OE\# = V_{IL}$ | –   | 70  | ns            |       |
| Address valid to output valid (page)                           | $t_{PAGE}$ | $t_{AVQV1}$                                | $CE\# = V_{IL}$ ,<br>$OE\# = V_{IL}$ | –   | 20  | ns            |       |
| CE# LOW to output valid  | $t_{CE}$   | $t_{ELQV}$                                 | $OE\# = V_{IL}$                      | –   | 70  | ns            |       |
| OE# LOW to output valid  | $t_{OE}$   | $t_{GLQV}$                                 | $CE\# = V_{IL}$                      | –   | 25  | ns            |       |
| CE# HIGH to output High-Z                                      | $t_{HZ}$   | $t_{EHQZ}$                                 | $OE\# = V_{IL}$                      | –   | 20  | ns            | 1     |
| OE# HIGH to output High-Z                                      | $t_{DF}$   | $t_{GHQZ}$                                 | $CE\# = V_{IL}$                      | –   | 15  | ns            | 1     |
| CE# HIGH, OE# HIGH, or address transition to output transition | $t_{OH}$   | $t_{EHQX}$ ,<br>$t_{GHQX}$ ,<br>$t_{AXQX}$ | –                                    | 0   | –   | ns            |       |
| CE# LOW to BYTE# LOW   | $t_{ELFL}$ | $t_{ELBL}$                                 | –                                    | –   | 10  | ns            |       |
| CE# LOW to BYTE# HIGH  | $t_{ELFH}$ | $t_{ELBH}$                                 | –                                    | –   | 10  | ns            |       |
| BYTE# LOW to output valid                                      | $t_{FLQV}$ | $t_{BLQV}$                                 | –                                    | –   | 1   | $\mu\text{s}$ |       |
| BYTE# HIGH to output valid                                     | $t_{FHQV}$ | $t_{BHQV}$                                 | –                                    | –   | 1   | $\mu\text{s}$ |       |

Note: 1. Sampled only; not 100% tested.

**Table 31: Read AC Characteristics –  $V_{CCQ} = 1.65\text{V}\text{-}V_{CC}$**

| Parameter  | Symbol     |  | Condition                            | Min | Max | Unit | Notes |
|--|------------|--|--------------------------------------|-----|-----|------|-------|
|  | Legacy     | JEDEC                                      |                                      |     |     |      |       |
| Address valid to next address valid                            | $t_{RC}$   | $t_{AVAV}$                                 | $CE\# = V_{IL}$ ,<br>$OE\# = V_{IL}$ | 75  | –   | ns   |       |
| Address valid to output valid                                  | $t_{ACC}$  | $t_{AVQV}$                                 | $CE\# = V_{IL}$ ,<br>$OE\# = V_{IL}$ | –   | 75  | ns   |       |
| Address valid to output valid (page)                           | $t_{PAGE}$ | $t_{AVQV1}$                                | $CE\# = V_{IL}$ ,<br>$OE\# = V_{IL}$ | –   | 20  | ns   |       |
| CE# LOW to output valid  | $t_{CE}$   | $t_{ELQV}$                                 | $OE\# = V_{IL}$                      | –   | 75  | ns   |       |
| OE# LOW to output valid  | $t_{OE}$   | $t_{GLQV}$                                 | $CE\# = V_{IL}$                      | –   | 25  | ns   |       |
| CE# HIGH to output High-Z                                      | $t_{HZ}$   | $t_{EHQZ}$                                 | $OE\# = V_{IL}$                      | –   | 20  | ns   | 1     |
| OE# HIGH to output High-Z                                      | $t_{DF}$   | $t_{GHQZ}$                                 | $CE\# = V_{IL}$                      | –   | 15  | ns   | 1     |
| CE# HIGH, OE# HIGH, or address transition to output transition | $t_{OH}$   | $t_{EHQX}$ ,<br>$t_{GHQX}$ ,<br>$t_{AXQX}$ | –                                    | 0   | –   | ns   |       |
| CE# LOW to BYTE# LOW   | $t_{ELFL}$ | $t_{ELBL}$                                 | –                                    | –   | 10  | ns   |       |
| CE# LOW to BYTE# HIGH  | $t_{ELFH}$ | $t_{ELBH}$                                 | –                                    | –   | 10  | ns   |       |

**Table 31: Read AC Characteristics –  $V_{CCQ} = 1.65V - V_{CC}$  (Continued)**

| Parameter                  | Symbol     |            | Condition | Min | Max | Unit    | Notes |
|----------------------------|------------|------------|-----------|-----|-----|---------|-------|
|                            | Legacy     | JEDEC      |           |     |     |         |       |
| BYTE# LOW to output valid  | $t_{FLQV}$ | $t_{BLQV}$ | –         | –   | 1   | $\mu s$ |       |
| BYTE# HIGH to output valid | $t_{FHQV}$ | $t_{BHQV}$ | –         | –   | 1   | $\mu s$ |       |

Note: 1. Sampled only; not 100% tested.

**Figure 19: Random Read AC Timing (8-Bit Mode)**

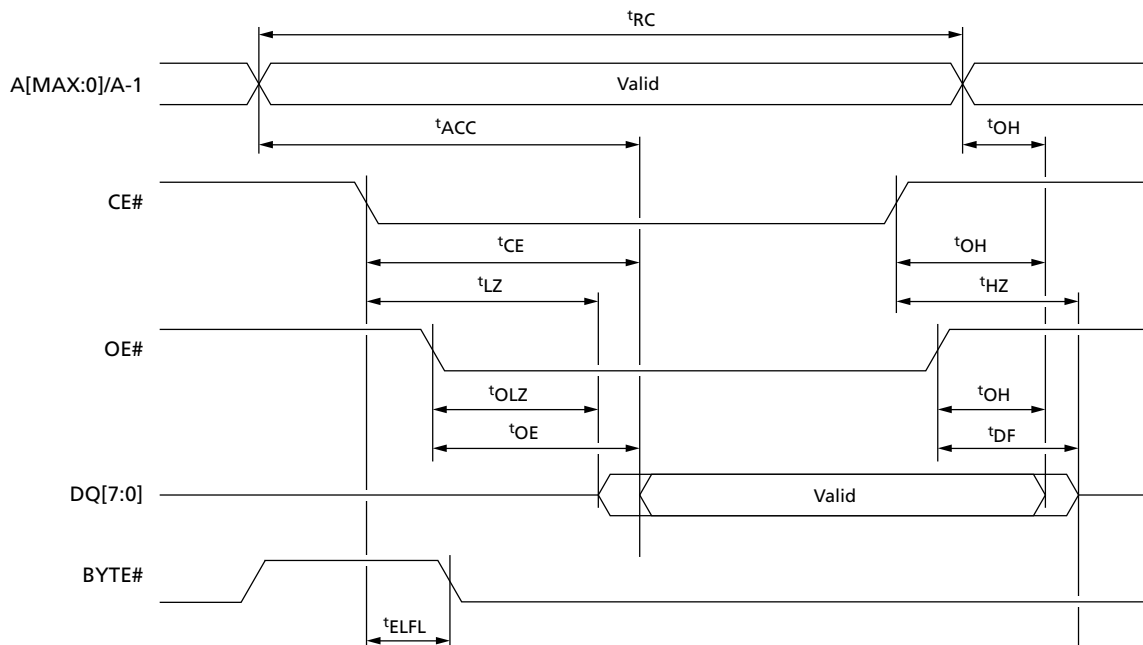




Figure 20: Random Read AC Timing (16-Bit Mode)

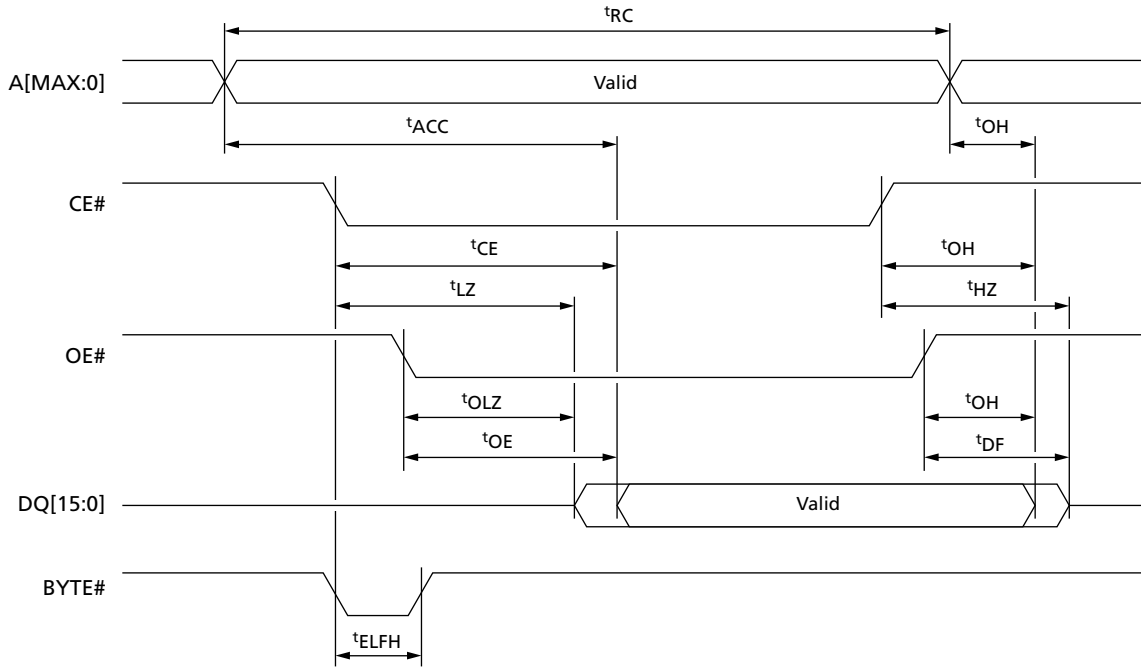
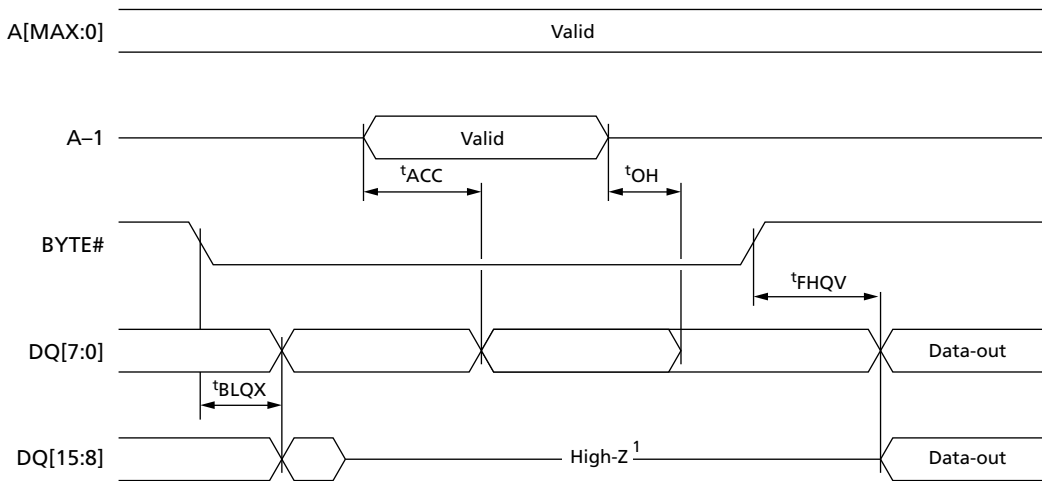


Figure 21: BYTE# Transition Read AC Timing



Note: 1. DQ15 transitions to be A-1 when BYTE# is LOW.

**Figure 22: Page Read AC Timing (16-Bit Mode)**



Note: 1. Page size is 16 words (32 bytes) and is addressed by address inputs A[3:0] in x16 bus mode and A[3:0] plus DQ15/A-1 in x8 bus mode.

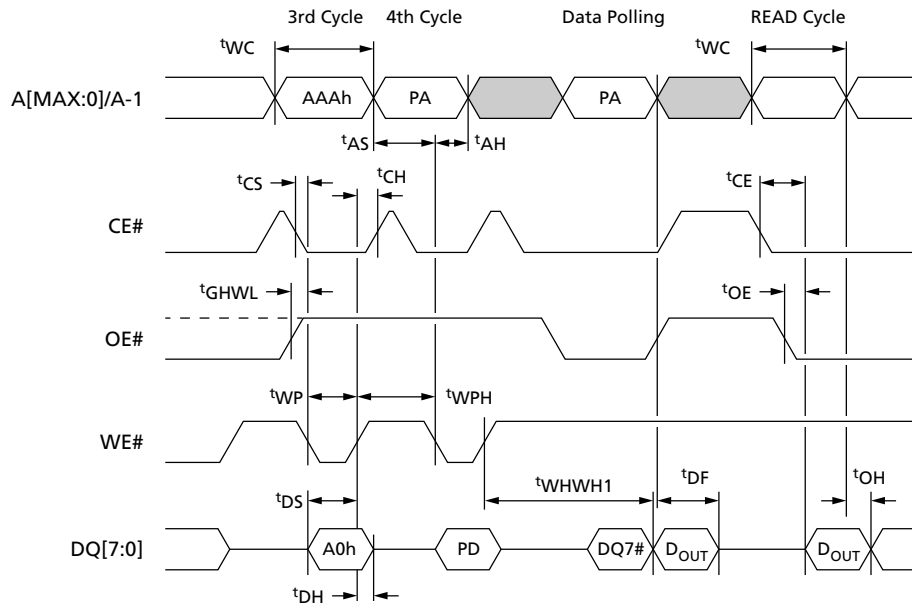
## Write AC Characteristics

Table 32: WE#-Controlled Write AC Characteristics

| Parameter   | Symbol            |                    | Min                       | Typ | Max | Unit | Notes |
|---|-------------------|--------------------|---------------------------|-----|-----|------|-------|
|   | Legacy            | JEDEC              |                           |     |     |      |       |
| WRITE cycle time  | t <sup>WC</sup>   | –                  | 60                        | –   | –   | ns   |       |
| CE# LOW to WE# LOW  | t <sup>CS</sup>   | t <sup>ELWL</sup>  | 0                         | –   | –   | ns   |       |
| WE# LOW to WE# HIGH                                       | t <sup>WP</sup>   | t <sup>WLWH</sup>  | 35                        | –   | –   | ns   |       |
| Input valid to WE# HIGH                                   | t <sup>DS</sup>   | t <sup>DVWH</sup>  | 30                        | –   | –   | ns   | 1     |
| WE# HIGH to input transition                              | t <sup>DH</sup>   | t <sup>WHDX</sup>  | 0                         | –   | –   | ns   |       |
| WE# HIGH to CE# HIGH                                      | t <sup>CH</sup>   | t <sup>WHEH</sup>  | 0                         | –   | –   | ns   |       |
| WE# HIGH to WE# LOW                                       | t <sup>WPH</sup>  | t <sup>WHWL</sup>  | 20                        | –   | –   | ns   |       |
| Address valid to WE# LOW                                  | t <sup>AS</sup>   | t <sup>AVWL</sup>  | 0                         | –   | –   | ns   |       |
| WE# LOW to address transition                             | t <sup>AH</sup>   | t <sup>WLAX</sup>  | 45                        | –   | –   | ns   |       |
| OE# HIGH to WE# LOW                                       | –                 | t <sup>GHWL</sup>  | 0                         | –   | –   | ns   |       |
| WE# HIGH to OE# LOW                                       | t <sup>OEH</sup>  | t <sup>WHGL</sup>  | 0                         | –   | –   | ns   |       |
| Program/erase valid to RY/BY# LOW                         | t <sup>BUSY</sup> | t <sup>WHRL</sup>  | –                         | –   | 90  | ns   | 2     |
| WE# HIGH to OE# valid                                     | –                 | t <sup>WHQV</sup>  | t <sup>AVQV</sup><br>+ 30 | –   | –   | ns   |       |
| V <sub>HH</sub> rise or fall time on V <sub>pp</sub> /WP# | –                 | t <sup>VHVPP</sup> | 250                       | –   | –   | ns   |       |

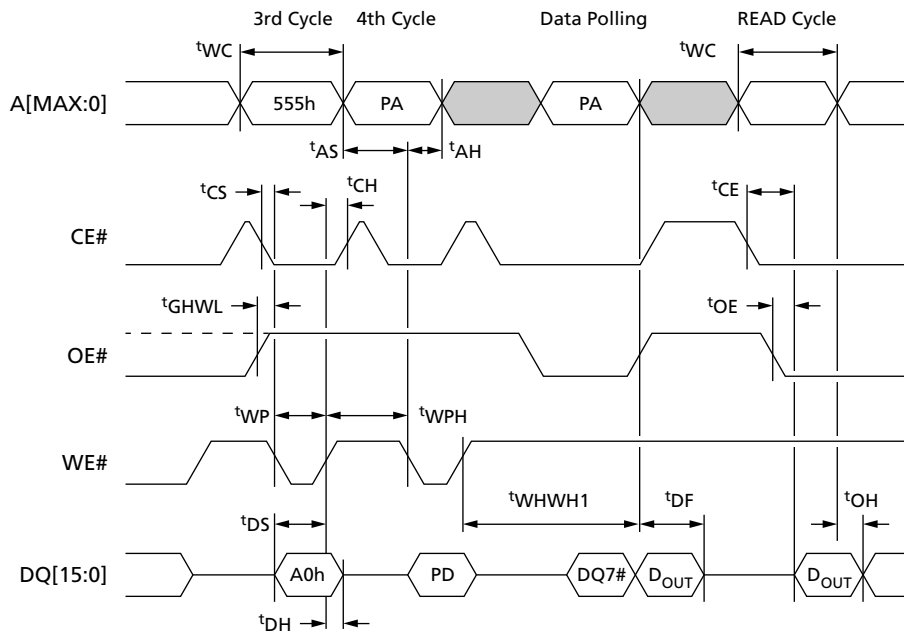
- Notes:
1. The user's write timing must comply with this specification. Any violation of this write timing specification may result in permanent damage to the NOR Flash device.
  2. Sampled only; not 100% tested.

Figure 23: WE#-Controlled Program AC Timing (8-Bit Mode)



- Notes:
1. Only the third and fourth cycles of the PROGRAM command are represented. The PROGRAM command is followed by checking of the data polling register bit and by a READ operation that outputs the data ( $D_{OUT}$ ) programmed by the previous PROGRAM command.
  2. PA is the address of the memory location to be programmed. PD is the data to be programmed.
  3. DQ7 is the complement of the data bit being programmed to DQ7 (See Data Polling Bit [DQ7]).
  4. See the following tables for timing details: Read AC Characteristics, WE#-Controlled Write AC Characteristics, and CE#-Controlled Write AC Characteristics.
  5. For  $t_{WHWH1}$  timing details, see the Program/Erase Characteristics table.

Figure 24: WE#-Controlled Program AC Timing (16-Bit Mode)



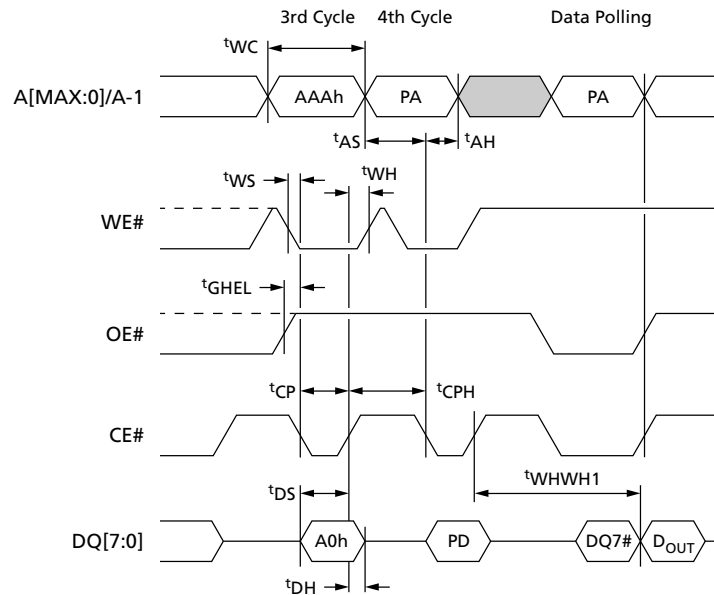
- Notes:
1. Only the third and fourth cycles of the PROGRAM command are represented. The PROGRAM command is followed by checking of the data polling register bit and by a READ operation that outputs the data ( $D_{OUT}$ ) programmed by the previous PROGRAM command.
  2. PA is the address of the memory location to be programmed. PD is the data to be programmed.
  3. DQ7 is the complement of the data bit being programmed to DQ7 (See Data Polling Bit [DQ7]).
  4. See the following tables for timing details: Read AC Characteristics, WE#-Controlled Write AC Characteristics, and CE#-Controlled Write AC Characteristics.
  5. For  $t_{WHWH1}$  timing details, see the Program/Erase Characteristics table.

**Table 33: CE#-Controlled Write AC Characteristics**

| Parameter   | Symbol            |                    | Min                       | Typ | Max | Unit | Notes |
|---|-------------------|--------------------|---------------------------|-----|-----|------|-------|
|   | Legacy            | JEDEC              |                           |     |     |      |       |
| WRITE cycle time  | t <sup>WC</sup>   | –                  | 60                        | –   | –   | ns   |       |
| WE# LOW to CE# LOW  | t <sup>WS</sup>   | t <sup>WLEL</sup>  | 0                         | –   | –   | ns   |       |
| CE# LOW to CE# HIGH                                       | t <sup>CP</sup>   | t <sup>ELEH</sup>  | 35                        | –   | –   | ns   |       |
| Input valid to CE# HIGH                                   | t <sup>DS</sup>   | t <sup>DVEH</sup>  | 30                        | –   | –   | ns   | 1     |
| CE# HIGH to input transition                              | t <sup>DH</sup>   | t <sup>EHDX</sup>  | 0                         | –   | –   | ns   |       |
| CE# HIGH to WE# HIGH                                      | t <sup>WH</sup>   | t <sup>EHWH</sup>  | 0                         | –   | –   | ns   |       |
| CE# HIGH to CE# LOW                                       | t <sup>CPH</sup>  | t <sup>EHEL</sup>  | 20                        | –   | –   | ns   |       |
| Address valid to CE# LOW                                  | t <sup>AS</sup>   | t <sup>AVEL</sup>  | 0                         | –   | –   | ns   |       |
| CE# LOW to address transition                             | t <sup>AH</sup>   | t <sup>ELAX</sup>  | 45                        | –   | –   | ns   |       |
| OE# HIGH to CE# LOW                                       | –                 | t <sup>GHEL</sup>  | 0                         | –   | –   | ns   |       |
| V <sub>HH</sub> rise or fall time on V <sub>pp</sub> /WP# | –                 | t <sup>VHVPP</sup> | 250                       | –   | –   | ns   |       |
| Program/erase valid to RY/BY# LOW                         | t <sup>BUSY</sup> | t <sup>WHRL</sup>  | –                         | –   | 90  | ns   | 2     |
| WE# HIGH to OE# valid                                     | –                 | t <sup>WHQV</sup>  | t <sup>AVQV</sup> +<br>30 | –   | –   | ns   |       |

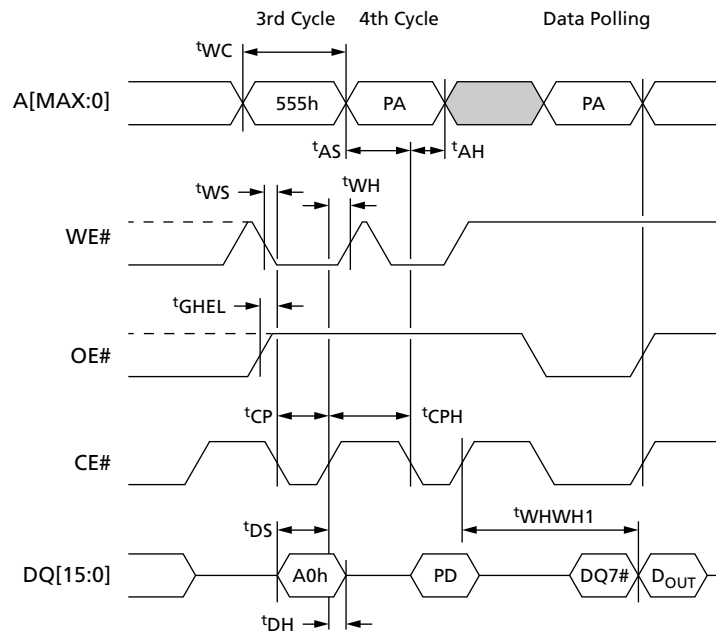
- Notes:
1. The user's write timing must comply with this specification. Any violation of this write timing specification may result in permanent damage to the NOR Flash device.
  2. Sampled only; not 100% tested.

Figure 25: CE#-Controlled Program AC Timing (8-Bit Mode)



- Notes:
1. Only the third and fourth cycles of the PROGRAM command are represented. The PROGRAM command is followed by checking of the data polling register bit.
  2. PA is the address of the memory location to be programmed. PD is the data to be programmed.
  3. DQ7 is the complement of the data bit being programmed to DQ7 (See Data Polling Bit [DQ7]).
  4. See the following tables for timing details: Read AC Characteristics, WE#-Controlled Write AC Characteristics, and CE#-Controlled Write AC Characteristics.
  5. For  $t_{WHWH1}$  timing details, see the Program/Erase Characteristics table.

Figure 26: CE#-Controlled Program AC Timing (16-Bit Mode)



- Notes:
1. Only the third and fourth cycles of the PROGRAM command are represented. The PROGRAM command is followed by checking of the data polling register bit.
  2. PA is the address of the memory location to be programmed. PD is the data to be programmed.
  3. DQ7 is the complement of the data bit being programmed to DQ7 (See Data Polling Bit [DQ7]).
  4. See the following tables for timing details: Read AC Characteristics, WE#-Controlled Write AC Characteristics, and CE#-Controlled Write AC Characteristics.
  5. For  $t_{WHWH1}$  timing details, see the Program/Erase Characteristics table.



Figure 27: Chip/Block Erase AC Timing (16-Bit Mode)



- Notes:
1. For a CHIP ERASE command, the address is 555h, and the data is 10h; for a BLOCK ERASE command, the address is BAd, and the data is 30h.
  2. BAd is the block address.
  3. See the following tables for timing details: Read AC Characteristics, WE#-Controlled Write AC Characteristics, and CE#-Controlled Write AC Characteristics.
  4. For  $t_{WHWH1}$  timing details, see the Program/Erase Characteristics table.

Figure 28: Accelerated Program AC Timing



## Data Polling/Toggle AC Characteristics

**Table 34: Data Polling/Toggle AC Characteristics**

Note 1 applies to entire table

| Parameter                                     | Symbol    |                      | Min | Max | Unit |
|---|-----------|----------------------|-----|-----|------|
|   | Legacy    | JEDEC                |     |     |      |
| Address setup time to CE# or OE# LOW          | $t_{ASO}$ | $t_{AXGL}$           | 15  | –   | ns   |
| Address hold time from OE# or CE# HIGH        | $t_{AHT}$ | $t_{GHAX}, t_{EHAX}$ | 0   | –   | ns   |
| CE# HIGH time                                 | $t_{EPH}$ | $t_{EHEL2}$          | 20  | –   | ns   |
| OE# HIGH time                                 | $t_{OPH}$ | $t_{GHGL2}$          | 20  | –   | ns   |
| WE# HIGH to OE# LOW (toggle and data polling) | $t_{OEH}$ | $t_{WHGL2}$          | 10  | –   | ns   |

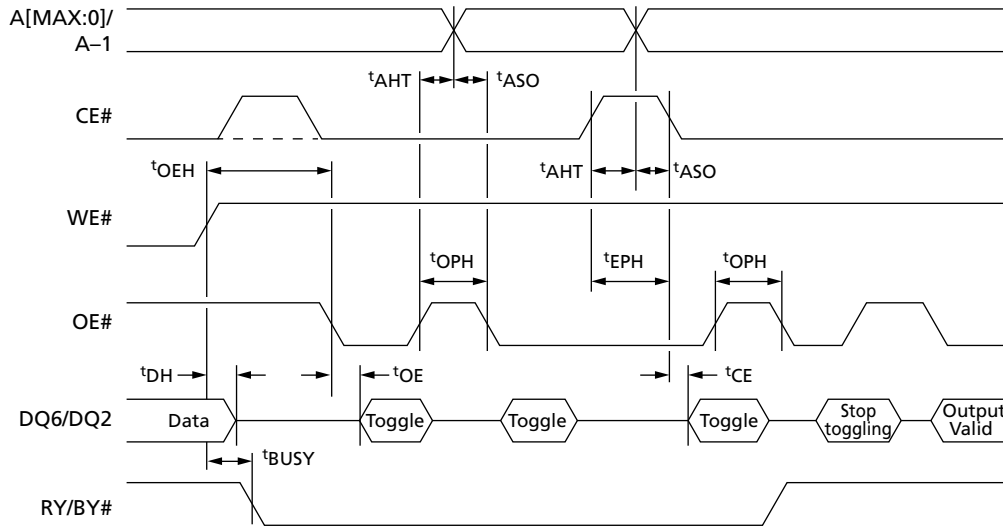
Note: 1. Sampled only; not 100% tested.

**Figure 29: Data Polling AC Timing**



- Notes:
1. DQ7 returns a valid data bit when the PROGRAM or ERASE command has completed.
  2. See the following tables for timing details: Read AC Characteristics and Data Polling/Toggle AC Characteristics.

**Figure 30: Toggle/Alternative Toggle Bit Polling AC Timing**



- Notes:
1. DQ6 stops toggling when the PROGRAM or ERASE command has completed. DQ2 stops toggling when the CHIP ERASE or BLOCK ERASE command has completed.
  2. See the following tables for timing details: Read AC Characteristics and Data Polling/Toggle AC Characteristics.



## Program/Erase Characteristics

**Table 35: Program/Erase Characteristics**

Notes 1 and 2 apply to entire table

| Parameter  | Buffer Size | Byte | Word | Min | Typ  | Max   | Unit | Notes |
|--|-------------|------|------|-----|------|-------|------|-------|
| <b>Erase</b>   |             |      |      |     |      |       |      |       |
| Block erase (128KB)  | –           | –    | –    | –   | 0.2  | 1.1   | s    | –     |
| Chip erase (256Mb)   | –           | –    | –    | –   | 52   | –     | s    | –     |
| Erase suspend latency time                                       | –           | –    | –    | –   | –    | 20    | µs   | –     |
| Block erase timeout  | –           | –    | –    | –   | –    | 50    | µs   | –     |
| Erase or erase resume to suspend                                 | –           | –    | –    | –   | 100  | –     | µs   | 3, 4  |
| Accelerated chip erase   | –           | –    | –    | –   | 47   | –     | s    | –     |
| <b>Program</b>   |             |      |      |     |      |       |      |       |
| Single-byte/single-word program                                  | –           | –    | –    | –   | 25   | 200   | µs   | –     |
| <b>Buffer Program (Byte mode)</b>                                |             |      |      |     |      |       |      |       |
| Byte write to buffer program ( <sup>t</sup> WHWH1)               | 64          | 64   | –    | –   | 92   | 460   | µs   | –     |
|  | 128         | 128  | –    | –   | 117  | 600   | µs   | –     |
|  | 256         | 256  | –    | –   | 171  | 900   | µs   | –     |
| Effective write to buffer program per byte ( <sup>t</sup> WHWH1) | 64          | 1    | –    | –   | 1.44 | 7.19  | µs   | –     |
|  | 128         | 1    | –    | –   | 0.91 | 4.69  | µs   | –     |
|  | 256         | 1    | –    | –   | 0.67 | 3.52  | µs   | –     |
| <b>Buffer Program (Word mode)</b>                                |             |      |      |     |      |       |      |       |
| Word write to buffer program ( <sup>t</sup> WHWH1)               | 32          | –    | 32   | –   | 92   | 460   | µs   | –     |
|  | 64          | –    | 64   | –   | 117  | 600   | µs   | –     |
|  | 128         | –    | 128  | –   | 171  | 900   | µs   | –     |
|  | 256         | –    | 256  | –   | 285  | 1500  | µs   | –     |
|  | 512         | –    | 512  | –   | 512  | 2000  | µs   | –     |
| Effective write to buffer program per word ( <sup>t</sup> WHWH1) | 32          | –    | 1    | –   | 2.88 | 14.38 | µs   | –     |
|  | 64          | –    | 1    | –   | 1.83 | 9.38  | µs   | –     |
|  | 128         | –    | 1    | –   | 1.34 | 7.03  | µs   | –     |
|  | 256         | –    | 1    | –   | 1.11 | 5.86  | µs   | –     |
|  | 512         | –    | 1    | –   | 1.0  | 3.9   | µs   | –     |
| Accelerated full buffer program time                             | –           | –    | –    | –   | 410  | –     | µs   | –     |
| Program suspend latency time                                     | –           | –    | –    | –   | –    | 15    | µs   | –     |
| <b>Nonvolatile protection</b>                                    |             |      |      |     |      |       |      |       |
| Set nonvolatile protection bit time                              | –           | –    | –    | –   | 25   | 200   | µs   | –     |
| Clear nonvolatile protection bit time                            | –           | –    | –    | –   | 80   | 1100  | ms   | –     |
| <b>Blank Check, CRC, and Program/Erase Endurance</b>             |             |      |      |     |      |       |      |       |
| Blank check: main block  | –           | –    | –    | –   | 3.2  | –     | ms   | –     |
| CRC check time: main block                                       | –           | –    | –    | –   | 5    | –     | ms   | –     |



**Table 35: Program/Erase Characteristics (Continued)**

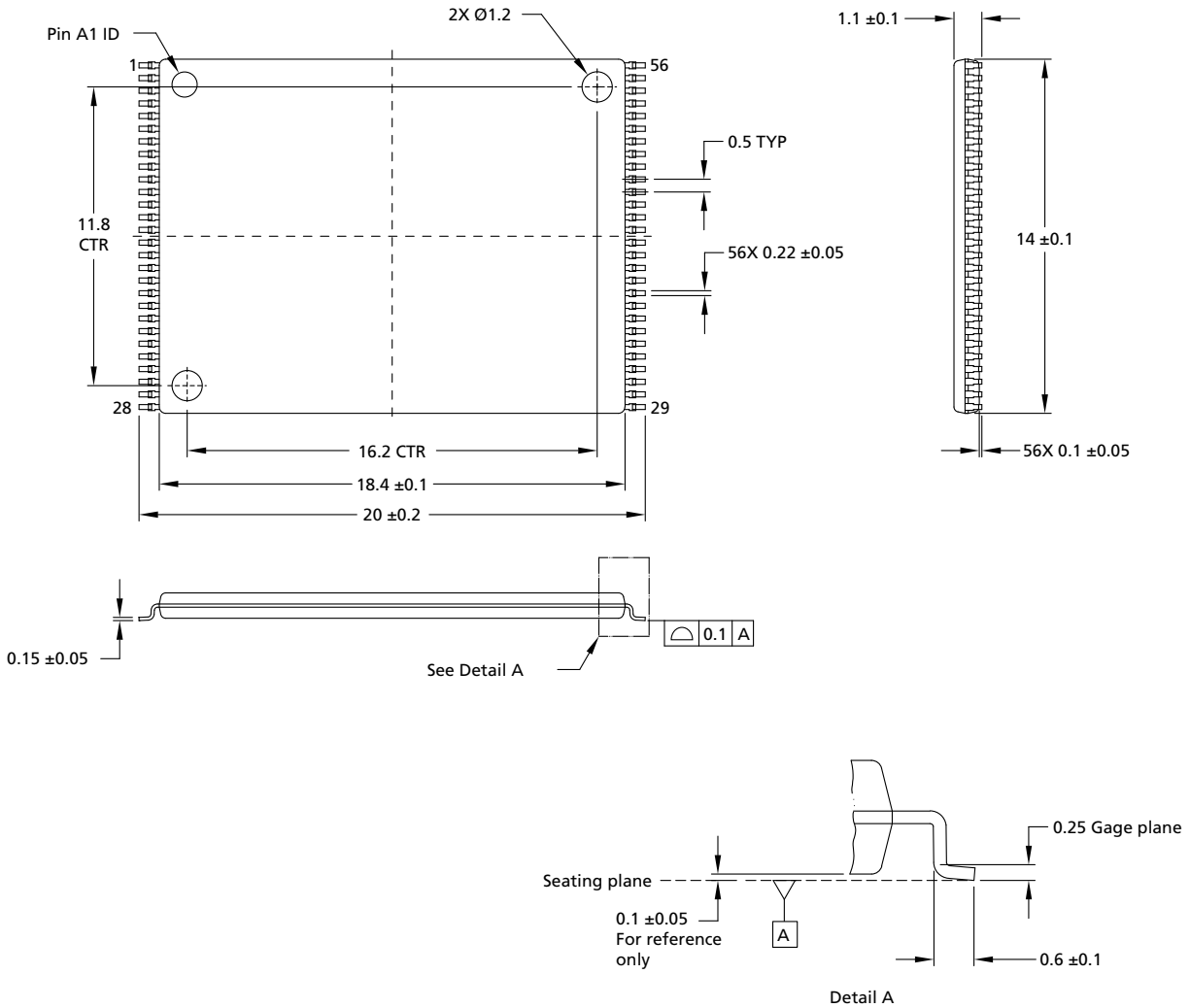
Notes 1 and 2 apply to entire table

| Parameter                         | Buffer Size | Byte | Word | Min     | Typ | Max | Unit   | Notes |
|-----------------------------------|-------------|------|------|---------|-----|-----|--------|-------|
| CRC check time: full chip (256Mb) | –           | –    | –    | –       | 2.5 | –   | s      | –     |
| PROGRAM/ERASE cycles (per block)  | –           | –    | –    | 100,000 | –   | –   | cycles | –     |

- Notes:
1. Typical values measured at room temperature and nominal voltages( $V_{cc}=3V$ ).
  2. Typical and maximum values are sampled, but not 100% tested.
  3. Erase to suspend is the time between an initial BLOCK ERASE or ERASE RESUME command and a subsequent ERASE SUSPEND command.
  4. This typical value allows an ERASE operation to progress to completion—it is important to note that the algorithm might never finish if the ERASE operation is *always* suspended less than this specification.

## Package Dimensions

Figure 31: 56-Pin TSOP – 14mm x 20mm (Package Code: JS)



- Notes:
1. All dimensions are in millimeters.
  2. Pin A1 ID diameter is 1mm.
  3. New package assembly site has effected an ASE process change (original ASE process is Amkor). The package shows two eject pins on the package mark: one in the corner by pin 56 and one in the corner by pin 28, each with diameter 2mm x 1.2mm.
  4. Package width and length include mold flash.

Figure 32: 64-Ball LBGGA – 11mm x 13mm (Package Code: PC)



Note: 1. All dimensions are in millimeters.

Figure 33: 56-Ball VFBGA – 7mm x 9mm (Package Code: PN)



Note: 1. All dimensions are in millimeters.



## Revision History

### Rev. G –05/18

- Added Important Notes and Warnings section for further clarification aligning to industry standards

### Rev. F – 11/16

- Updated 56-pin dimension drawing
- Added Note 4 to 56-pin dimension drawing

### Rev. E – 4/16

- Added 56-ball VFBGA 7mm x 9mm

### Rev. D – 5/15

- Updated document status to Production

### Rev. C – 4/15

- Updated block protection bit specification under Registers
- Clarified descriptions for <sup>t</sup>OH, <sup>t</sup>ELFL, <sup>t</sup>ELFH under Read AC Characteristics

### Rev. B – 1/15

- Change status to Preliminary version.

### Rev. A – 6/14

- Initial release

8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-4000  
www.micron.com/products/support Sales inquiries: 800-932-4992  
Micron and the Micron logo are trademarks of Micron Technology, Inc.  
All other trademarks are the property of their respective owners.

This data sheet contains minimum and maximum limits specified over the power supply and temperature range set forth herein. Although considered final, these specifications are subject to change, as further product development and data characterization sometimes occur.

## X-ON Electronics

Largest Supplier of Electrical and Electronic Components

*Click to view similar products for [NOR Flash](#) category:*

*Click to view products by [Micron](#) manufacturer:*

Other Similar products are found below :

[615309A](#) [MBM29F800BA-90PF-SFLE1](#) [8 611 200 906](#) [9990933135](#) [AT25DF021A-MHN-Y](#) [AT25DF256-SSHN-T](#) [EAN62691701](#)  
[MX29F200CTMI-70G](#) [PC28F320J3F75A](#) [8 905 959 252](#) [S29AL008J55BFIR20](#) [S29AL016J70FFI022](#) [S29AS016J70BHIF40](#) [S99-50389 P](#)  
[MBM29F800BA-70PFTN-SFLE1](#) [AT25DF011-MAHN-T](#) [AT25DF011-SSHN-T](#) [AT25DF011-XMHN-T](#) [AT25DF041B-MHN-Y](#)  
[AT45DB161E-CCUD-T](#) [RP-SDCCTH0](#) [S29PL127J70BAI020](#) [S99-50052](#) [MX25L6473EM2I-10G](#) [W29GL256SL9T](#) [W29GL128CH9C](#)  
[W29GL128CH9B](#) [MX25L3233FMI-08G](#) [S99-50243 P](#) [S29GL512T12TFN010](#) [S29GL512T10DHI020](#) [S26KS128SDGBHI030](#)  
[S26KL256SDABHB030](#) [S26KL128SDABHB020](#) [PC28F640J3F75B](#) [W29GL256SH9C](#) [S99-50239](#) [S29GL032N11FFIS12](#)  
[S26KS512SDABHB030](#) [S26KL256SDABHA020](#) [S25FS128SAGMFV100](#) [S25FS064SDSNFN030](#) [AM29F200BT-70EF\(SPANSION\)](#)  
[AM29F800BB-120SI\(SPANSION\)](#) [LE25S40MB-AH](#) [AT25DF512C-XMHN-T](#) [S25FL256LAGMFI001](#) [S26KL128SDABHN020](#)  
[520366231286](#) [AT25SL128A-UUE-T](#)