

# Parallel NOR Flash Embedded Memory

**M29W640GH, M29W640GL****M29W640GT, M29W640GB**

## Features

- Supply voltage
  - $V_{CC} = 2.7\text{--}3.6\text{V}$  (program, erase, read)
  - $V_{PP} = 12\text{V}$  for fast program (optional)
- Asynchronous random/page read
  - Page width: 4 words
  - Page access: 25ns
  - Random access: 60ns, 70ns, 90ns
- Fast program commands
  - 2-word/4-byte program (without  $V_{PP} = 12\text{V}$ )
  - 4-word/8-byte program (with  $V_{PP} = 12\text{V}$ )
  - 16-word/32-byte write buffer
- Programming time
  - 10 $\mu\text{s}$  per byte/word TYP
  - Chip program time: 10 s (4-word program)
  - Double word/quadruple byte program
- Memory organization
  - M29W640GH/L 128 main blocks, 64KB each
  - M29W640GT/B 127 main blocks, 64KB each and 8 boot blocks, 8KB each
- Program/erase controller
  - Embedded byte/word program algorithms
- Program/erase suspend and resume
  - Read from any block during a PROGRAM SUSPEND operation
  - Read or program another block during an ERASE SUSPEND operation
- Hardware block protection
  - $V_{PP}/WP\#$  pin for fast program and write protect
  - Temporary block unprotect mode
- Common Flash interface
  - 64-bit security code
- 128-word extended memory block
  - Extra block used as security block or to store additional information
- Low power consumption: Standby and automatic mode
- 100,000 PROGRAM/ERASE cycles per block
- Electronic signature
  - Manufacturer code: 0020h
- Device summary: part number and device code
  - M29W640GH: uniform, last block protected by  $V_{PP}/WP\#$
  - 227Eh + 220Ch + 2201h
  - M29W640GL: uniform, first block protected by  $V_{PP}/WP\#$
  - 227Eh + 220Ch + 2200h
  - M29W640GT: top boot block
  - 227Eh + 2210h + 2201h
  - M29W640GB: bottom boot block
  - 227Eh + 2210h + 2200h
- RoHS-compliant packages
  - 48-pin TSOP (NA) 12mm x 20mm
  - 56-pin TSOP (NB) 14mm x 20mm
  - 48-ball TFBGA (ZA) 6mm x 8mm
  - 64-ball FBGA (ZS) 11mm x 13mm
  - 64-ball TBGA (ZF) 10mm x 13mm
- Automotive certified parts available

## Part Numbering Information

Available with extended memory block prelocked by Micron. Devices are shipped from the factory with memory content bits erased to 1. For available options, such as packages or speed, 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.

**Table 1: Part Number Information**

| Part Number Category | Category Details                                                                                                                                                       |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Device Type          | M29 = Parallel Flash memory                                                                                                                                            |
| Operating Voltage    | W = 2.7 to 3.6V                                                                                                                                                        |
| Device Function      | 640G = 64Mb (x8/x16) boot block, uniform or boot block                                                                                                                 |
| Array Matrix         | T = Top boot                                                                                                                                                           |
|                      | B = Bottom boot                                                                                                                                                        |
|                      | L = First block protected by V <sub>pp</sub> /WP#                                                                                                                      |
|                      | H = Last block protected by V <sub>pp</sub> /WP#                                                                                                                       |
| Speed                | 60 = 60ns (in conjunction with temperature range 6; denotes industrial grade –40°C to 85°C parts)                                                                      |
|                      | 6A = 60ns (in conjunction with temperature range 6; denotes automotive grade –40°C to 85°C parts)                                                                      |
|                      | 7A = 70ns (in conjunction with temperature range 6; denotes automotive grade –40°C to 85°C parts)                                                                      |
|                      | 70 = 70ns (in conjunction with temperature range 6; denotes industrial grade –40°C to 85°C parts; temperature range = 3 denotes automotive grade –40°C to 125°C parts) |
|                      | 90 = 90ns (in conjunction with temperature range 6; denotes industrial grade –40°C to 85°C parts)                                                                      |
| Package              | NA = 48-pin TSOP, 12mm x 20mm                                                                                                                                          |
|                      | NB = 56-pin TSOP, 14mm x 20mm (available upon request)                                                                                                                 |
|                      | ZA = 48-ball TFBGA, 6mm x 8mm, 0.8mm pitch                                                                                                                             |
|                      | ZS = 64-ball FBGA, 11mm x 13mm, 1mm pitch                                                                                                                              |
|                      | ZF = 64-ball TBGA, 10mm x 13mm, 1mm pitch (available upon request)                                                                                                     |
| Temperature Range    | 6 = –40°C to 85°C                                                                                                                                                      |
|                      | 3 = –40°C to 125°C (Automotive)                                                                                                                                        |
| Shipping Options     | E = RoHS-compliant package, standard packing                                                                                                                           |
|                      | F = RoHS-compliant package, tape and reel packing                                                                                                                      |

## Contents

|                                                           |    |
|-----------------------------------------------------------|----|
| General Description .....                                 | 7  |
| Signal Assignments .....                                  | 10 |
| Signal Descriptions .....                                 | 14 |
| Memory Organization .....                                 | 17 |
| Memory Configuration .....                                | 17 |
| Uniform Block Memory Map, x16 – 64Mb Density .....        | 17 |
| Uniform Block Memory Map, x8 – 64Mb Density .....         | 17 |
| Bus Operations .....                                      | 18 |
| Read .....                                                | 18 |
| Write .....                                               | 18 |
| Standby and Automatic Standby .....                       | 18 |
| Output Disable .....                                      | 18 |
| Status Register .....                                     | 19 |
| Data Polling Bit (DQ7) .....                              | 19 |
| Toggle Bit (DQ6) .....                                    | 19 |
| Error Bit (DQ5) .....                                     | 19 |
| Erase Timer Bit (DQ3) .....                               | 20 |
| Alternative Toggle Bit (DQ2) .....                        | 20 |
| Write to Buffer and Program Abort Bit (DQ1) .....         | 20 |
| READ Operations .....                                     | 24 |
| READ/RESET Command .....                                  | 24 |
| READ CFI Command .....                                    | 24 |
| AUTO SELECT Operations .....                              | 25 |
| AUTO SELECT Command .....                                 | 25 |
| Command Interface .....                                   | 28 |
| READ/RESET Command .....                                  | 28 |
| AUTO SELECT Command .....                                 | 28 |
| READ CFI QUERY Command .....                              | 28 |
| PROGRAM Command .....                                     | 29 |
| PROGRAM SUSPEND Command .....                             | 29 |
| PROGRAM RESUME Command .....                              | 30 |
| Fast Program Commands .....                               | 30 |
| DOUBLE BYTE PROGRAM Command .....                         | 30 |
| QUADRUPLE BYTE PROGRAM Command .....                      | 30 |
| OCTUPLE BYTE PROGRAM Command .....                        | 30 |
| DOUBLE WORD PROGRAM Command .....                         | 31 |
| QUADRUPLE WORD PROGRAM Command .....                      | 31 |
| WRITE TO BUFFER AND PROGRAM Command .....                 | 32 |
| WRITE TO BUFFER AND PROGRAM CONFIRM Command .....         | 33 |
| WRITE TO BUFFER AND PROGRAM ABORT AND RESET Command ..... | 33 |
| UNLOCK BYPASS Command .....                               | 33 |
| UNLOCK BYPASS PROGRAM Command .....                       | 33 |
| UNLOCK BYPASS RESET Command .....                         | 34 |
| CHIP ERASE Command .....                                  | 34 |
| BLOCK ERASE Command .....                                 | 34 |
| ERASE SUSPEND Command .....                               | 35 |
| ERASE RESUME Command .....                                | 35 |
| ENTER EXTENDED BLOCK Command .....                        | 35 |
| EXIT EXTENDED BLOCK Command .....                         | 35 |
| BLOCK PROTECT and CHIP UNPROTECT Commands .....           | 36 |



## 64Mb: 3V Embedded Parallel NOR Flash Features

|                                                        |    |
|--------------------------------------------------------|----|
| BLOCK PROTECT Command .....                            | 36 |
| Programmer Technique .....                             | 36 |
| In-System Technique .....                              | 39 |
| Common Flash Interface .....                           | 46 |
| Absolute Ratings and Operating Conditions .....        | 51 |
| DC Characteristics .....                               | 53 |
| Read AC Characteristics .....                          | 54 |
| Write AC Characteristics .....                         | 57 |
| Toggle and Alternative Toggle AC Characteristics ..... | 61 |
| Program/Erase Characteristics .....                    | 63 |
| Reset Characteristics .....                            | 67 |
| Package Dimensions .....                               | 68 |
| Revision History .....                                 | 73 |
| Rev. C – 07/13 .....                                   | 73 |
| Rev. B – 06/13 .....                                   | 73 |
| Rev. A – 11/12 .....                                   | 73 |

## List of Figures

|                                                               |    |
|---------------------------------------------------------------|----|
| Figure 1: Logic Diagram .....                                 | 8  |
| Figure 2: 48-Pin TSOP .....                                   | 10 |
| Figure 3: 56-Pin TSOP .....                                   | 11 |
| Figure 4: 48-Ball TFBGA .....                                 | 12 |
| Figure 5: 64-Ball TFBGA .....                                 | 13 |
| Figure 6: Data Polling Flowchart .....                        | 22 |
| Figure 7: Data Toggle Flowchart .....                         | 23 |
| Figure 8: Programmer Equipment Block Protect Flowchart .....  | 37 |
| Figure 9: Programmer Equipment Chip Unprotect Flowchart ..... | 38 |
| Figure 10: In-System Equipment Block Protect Flowchart .....  | 40 |
| Figure 11: In-System Equipment Chip Protect Flowchart .....   | 41 |
| Figure 12: AC Measurement Load Circuit .....                  | 52 |
| Figure 13: AC Measurement I/O Waveform .....                  | 52 |
| Figure 14: Random AC Timing .....                             | 55 |
| Figure 15: Page Read AC Timing .....                          | 56 |
| Figure 16: WE#-Controlled AC Timing .....                     | 58 |
| Figure 17: CE#-Controlled AC Timing .....                     | 60 |
| Figure 18: Toggle/Alternative Toggle, CE# Controlled .....    | 61 |
| Figure 19: Toggle/Alternative Toggle, OE# Controlled .....    | 62 |
| Figure 20: WE# Controlled Program Waveform .....              | 64 |
| Figure 21: CE# Controlled Program Waveform .....              | 65 |
| Figure 22: Chip/Block Erase Waveform .....                    | 66 |
| Figure 23: Reset/Block Temporary Unprotect AC Waveforms ..... | 67 |
| Figure 24: Accelerated Programming Timing Waveform .....      | 67 |
| Figure 25: 48-Pin TSOP – 12mm x 20mm .....                    | 68 |
| Figure 26: 56-Pin TSOP – 14mm x 20mm .....                    | 69 |
| Figure 27: 48-Ball TFBGA – 6mm x 8mm .....                    | 70 |
| Figure 28: 64-Ball TBGA – 10mm x 13mm .....                   | 71 |
| Figure 29: 64-Ball FBGA – 11mm x 13mm .....                   | 72 |

## List of Tables

|                                                                      |    |
|----------------------------------------------------------------------|----|
| Table 1: Part Number Information .....                               | 2  |
| Table 2: Signal Names .....                                          | 8  |
| Table 3: Protection Granularity on the M29W640GH and M29W640GL ..... | 9  |
| Table 4: Protection Granularity on the M29W640GT .....               | 9  |
| Table 5: Protection Granularity on the M29W640GB .....               | 9  |
| Table 6: Signal Descriptions .....                                   | 14 |
| Table 7: Hardware Protection .....                                   | 16 |
| Table 8: x16 Uniform Blocks [127:0] .....                            | 17 |
| Table 9: x8 Uniform Blocks [127:0] .....                             | 17 |
| Table 10: Bus Operations .....                                       | 18 |
| Table 11: Status Register Bits .....                                 | 20 |
| Table 12: Read Electronic Signature .....                            | 25 |
| Table 13: Block Protection .....                                     | 26 |
| Table 14: Programmer Technique Bus Operations .....                  | 39 |
| Table 15: Commands – 16-Bit Mode (BYTE# = V <sub>IH</sub> ) .....    | 42 |
| Table 16: Commands – 8-Bit Mode (BYTE# = V <sub>IL</sub> ) .....     | 44 |
| Table 17: Query Structure Overview .....                             | 46 |
| Table 18: CFI Query Identification String .....                      | 46 |
| Table 19: CFI Query System Interface Information .....               | 47 |
| Table 20: Device Geometry Definition .....                           | 47 |
| Table 21: Primary Algorithm-Specific Extended Query Table .....      | 48 |
| Table 22: Security Code Area .....                                   | 50 |
| Table 23: Absolute Maximum/Minimum Ratings .....                     | 51 |
| Table 24: Operating Conditions .....                                 | 51 |
| Table 25: Input/Output Capacitance .....                             | 52 |
| Table 26: DC Current Characteristics .....                           | 53 |
| Table 27: DC Voltage Characteristics .....                           | 53 |
| Table 28: Read AC Characteristics .....                              | 54 |
| Table 29: WE#-Controlled Write AC Characteristics .....              | 57 |
| Table 30: CE#-Controlled Write AC Characteristics .....              | 59 |
| Table 31: Toggle and Alternative Toggle AC Characteristics .....     | 61 |
| Table 32: Program/Erase Times and Endurance Cycles .....             | 63 |
| Table 33: Reset/Block Temporary Unprotect AC Characteristics .....   | 67 |

## General Description

The M29W640G is a 64Mb (8Mb x8 or 4Mb x16) nonvolatile memory that can be read, erased, and reprogrammed. These operations can be performed using a single low voltage (2.7–3.6V) supply. Upon power-up, the device defaults to read mode.

The memory is divided into blocks that can be erased independently so that valid data can be preserved while old data is erased. 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.

The M29W640GH and M29W640GL memory array is organized into 128 uniform blocks of 64KB each (or 32Kwords each).

The M29W640GT and M29W640GB feature an asymmetric memory block, each having an array of 135 blocks divided into 8 parameter blocks of 8KB each (or 4 Kwords each), and 127 main blocks of 64KB each (or 32Kwords each). The M29W640GT has the parameter blocks at the top of the memory array whereas the M29W640GB locates the parameter blocks starting from the bottom.

Blocks are protected by groups to prevent accidental PROGRAM or ERASE commands from modifying the memory.

The M29W640G supports asynchronous random read and page read from all blocks of the array. Chip enable, output enable, and write enable signals control the bus operation. They enable simple connection to most microprocessors, often without additional logic.

The  $V_{pp}/WP\#$  signal is used to enable faster programming of the device. Protection from PROGRAM/ERASE commands can be obtained by holding  $V_{pp}/WP\#$  to  $V_{ss}$ :

- On the M29W640GH and M29W640GL, the last and first blocks are protected.
- On the M29W640GT and M29W640GB, the last two and first two boot blocks are protected.

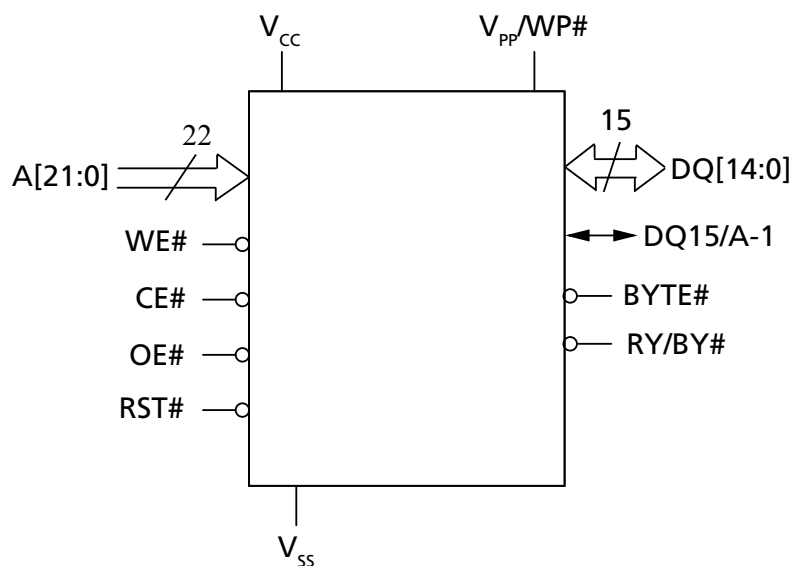
The M29W640G devices feature a full set of fast program commands to improve programming throughput:

- 2-byte PROGRAM (it is not necessary to raise  $V_{pp}/WP\#$  to 12V before issuing this command)
- 2 words/4-byte PROGRAM (it is not necessary to raise  $V_{pp}/WP\#$  to 12V before issuing this command)
- 4 word/8-byte PROGRAM ( $V_{pp}/WP\#$  must be raised to 12V before issuing this command)
- WRITE TO BUFFER and PROGRAM (enables program in one shot a buffer of 16 words/32 bytes)

The M29W640G has an extra block, the extended block, of 128 words in x16 mode or 256 bytes in x8 mode that can be accessed using a dedicated command. The extended block can be protected, and therefore, is useful for storing security information. However, protection is not reversible; once protected, the protection cannot be undone.

The memory is delivered with all bits erased (set to 1).

**Figure 1: Logic Diagram**



**Table 2: Signal Names**

| Name                 | Description                                                         | Direction      |
|----------------------|---------------------------------------------------------------------|----------------|
| A0-A21               | Address inputs                                                      | Inputs         |
| CE#                  | Chip enable                                                         | Input          |
| OE#                  | Output enable                                                       | Input          |
| WE#                  | Write enable                                                        | Input          |
| RP#                  | Reset/Block temporary unprotect                                     | Input          |
| RY/BY#               | Ready/Busy                                                          | Input          |
| BYTE#                | Byte/Word organization select                                       | Input          |
| DQ0-DQ7              | Data input/outputs                                                  | I/O            |
| DQ8-DQ14             | Data input/outputs                                                  | I/O            |
| DQ15A-1 (or DQ15)    | Data input/output or address input (or data I/O)                    | I/O            |
| V <sub>CC</sub>      | Supply voltage                                                      | Supply voltage |
| V <sub>PP</sub> /WP# | Supply voltage for FAST PROGRAM (optional) or WRITE PROTECT command | Supply voltage |
| V <sub>SS</sub>      | Ground                                                              | —              |
| NC                   | Not connected internally                                            | —              |

Note: 1. V<sub>PP</sub>/WP# may be left floating because it is internally connected to a pull-up resistor to enable PROGRAM/ERASE commands.

**Table 3: Protection Granularity on the M29W640GH and M29W640GL**

| Block      | KB/Kwords | Protection Block Group | (x8)                         | (x16)                        |
|------------|-----------|------------------------|------------------------------|------------------------------|
| 0 to 3     | 4 x 64/32 | Block level            | 000000h-03FFFFh <sup>1</sup> | 000000h-01FFFFh <sup>1</sup> |
| 4 to 7     | 4 x 64/32 | Protection group       | 040000h-07FFFFh              | 020000h-03FFFFh              |
| –          | –         | –                      | –                            | –                            |
| 120 to 123 | 4 x 64/32 | Protection group       | 780000h-7BFFFFh              | 3C0000h-3DFFFFh              |
| 124 to 127 | 4 x 64/32 | Block level            | 7C0000h-7FFFFFh              | 3E0000h-3FFFFFh              |

Note: 1. Used as the extended block addresses in extended block mode.

**Table 4: Protection Granularity on the M29W640GT**

| Block      | KB/Kwords            | Protection Block Group | (x8)                         | (x16)                        |
|------------|----------------------|------------------------|------------------------------|------------------------------|
| 0 to 3     | 4 x 64/32            | Protection group       | 000000h-03FFFFh <sup>1</sup> | 000000h-01FFFFh <sup>1</sup> |
| 4 to 7     | 4 x 64/32            | Protection group       | 040000h-07FFFFh              | 020000h-03FFFFh              |
| –          | –                    | –                      | –                            | –                            |
| 120 to 123 | 4 x 64/32            | Protection group       | 780000h-7BFFFFh              | 3C0000h-3DFFFFh              |
| 124 to 126 | 3 x 64/32            | Protection group       | 7C0000h-7EFFFFh              | 3E0000h-3F7FFFh              |
| 127 to 134 | 8 x 8/4 <sup>2</sup> | Block level            | 7F0000h-7FFFFFh              | 3F8000h-3FFFFFh              |

Notes: 1. Used as the extended block addresses in extended block mode.  
2. Boot blocks.

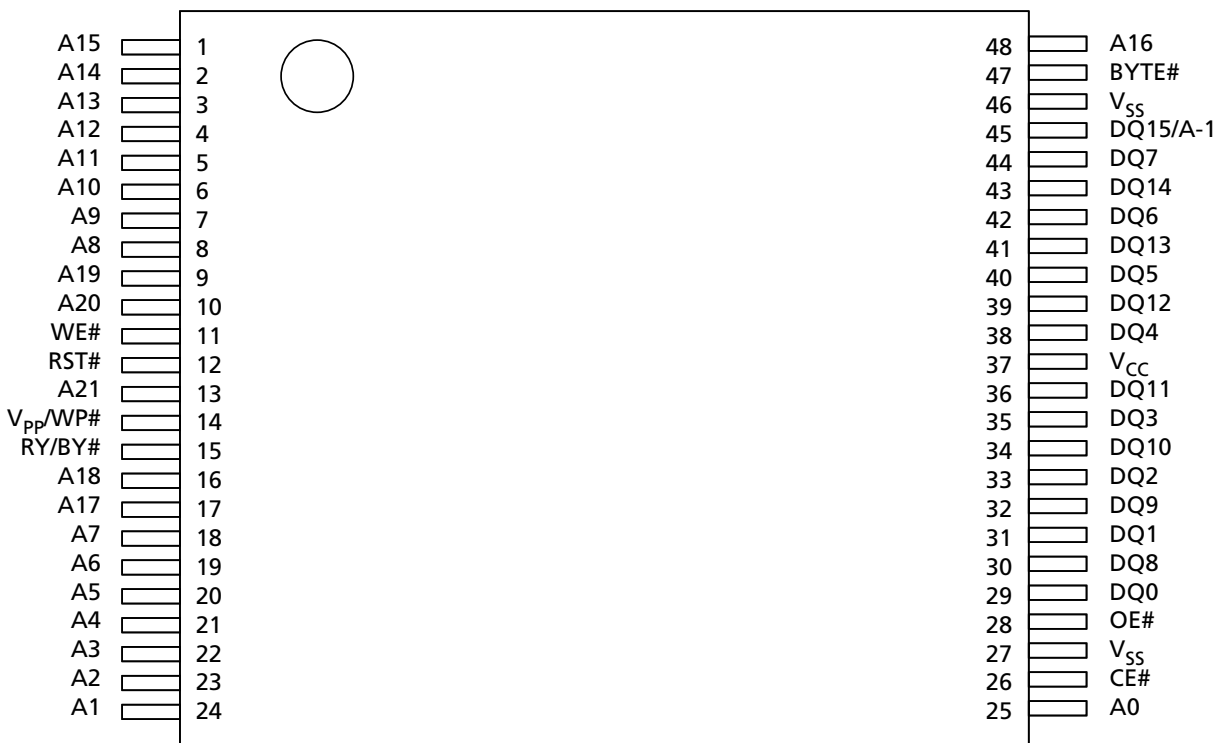
**Table 5: Protection Granularity on the M29W640GB**

| Block      | KB/Kwords           | Protection Block Group | (x8)                         | (x16)                        |
|------------|---------------------|------------------------|------------------------------|------------------------------|
| 0 to 7     | 8x 8/4 <sup>1</sup> | Block level            | 000000h-00FFFFh <sup>2</sup> | 000000h-007FFFh <sup>2</sup> |
| 8 to 10    | 3 x 64/32           | Protection group       | 010000h-03FFFFh              | 008000h-01FFFFh              |
| 11 to 14   | 4 x 64/32           | Protection group       | 040000h-07FFFFh              | 020000h-03FFFFh              |
| –          | –                   | –                      | –                            | –                            |
| 127 to 130 | 4 x 64/32           | Protection group       | 780000h-7BFFFFh              | 3C0000h-3DFFFFh              |
| 131 to 134 | 4 x 64/32           | Protection group       | 7C0000h-7FFFFFh              | 3E0000h-3FFFFFh              |

Notes: 1. Boot blocks.  
2. Used as the extended block addresses in extended block mode.

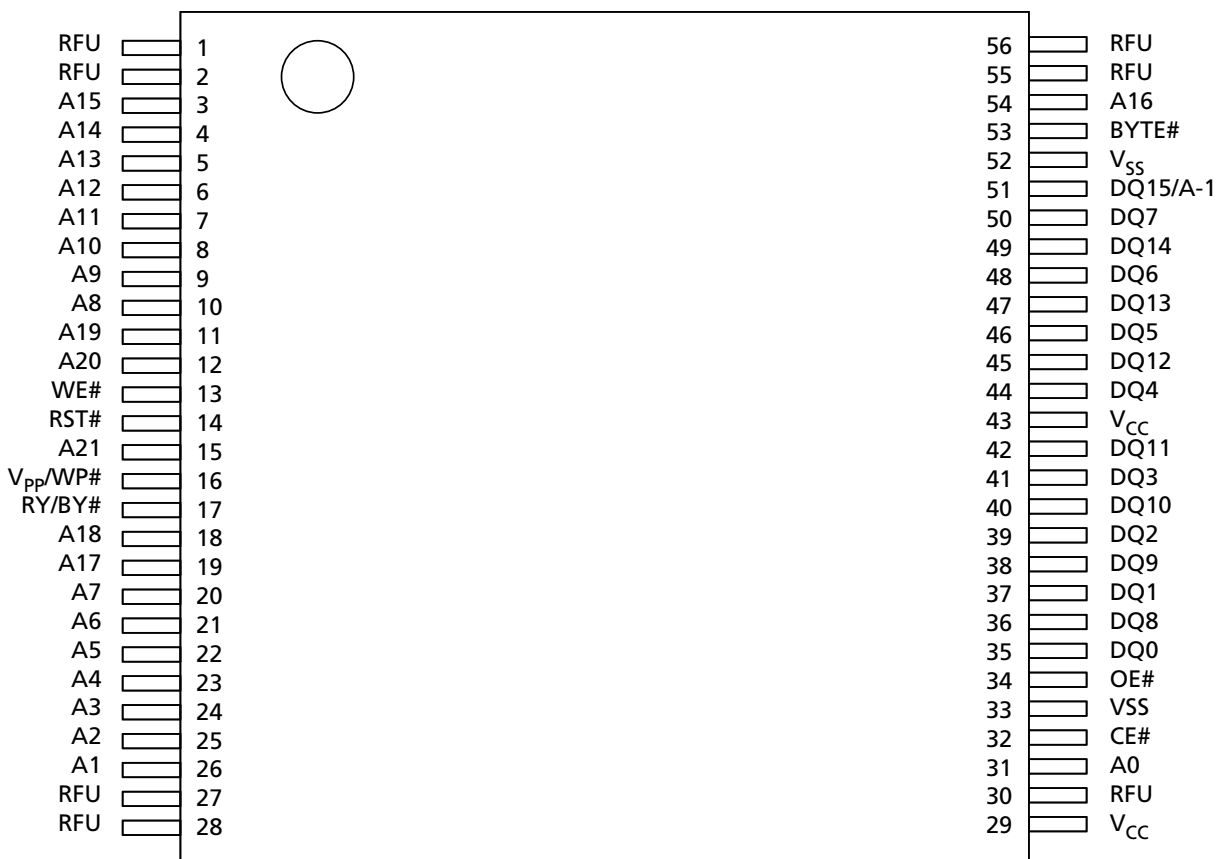
## Signal Assignments

Figure 2: 48-Pin TSOP



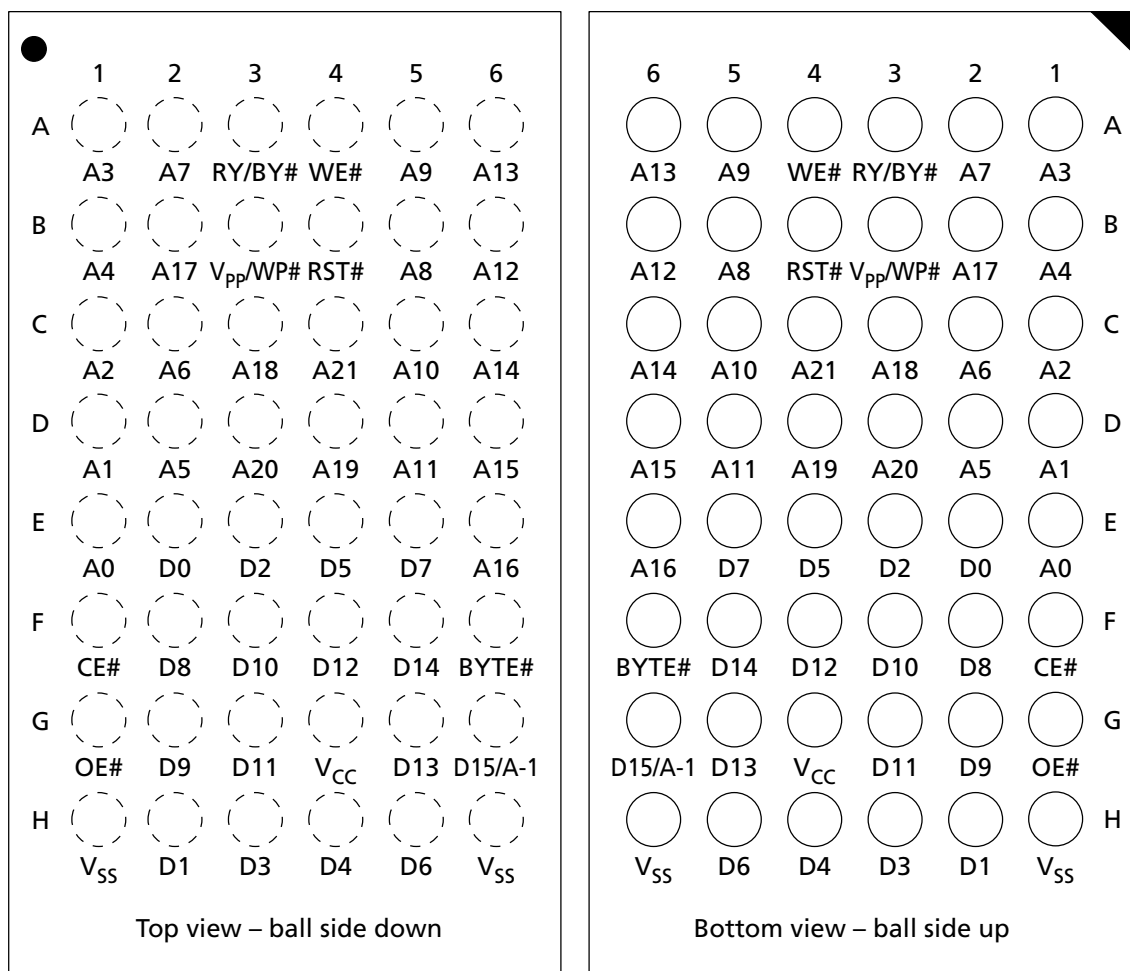
Note: 1. RFU = reserved for future use.

**Figure 3: 56-Pin TSOP**



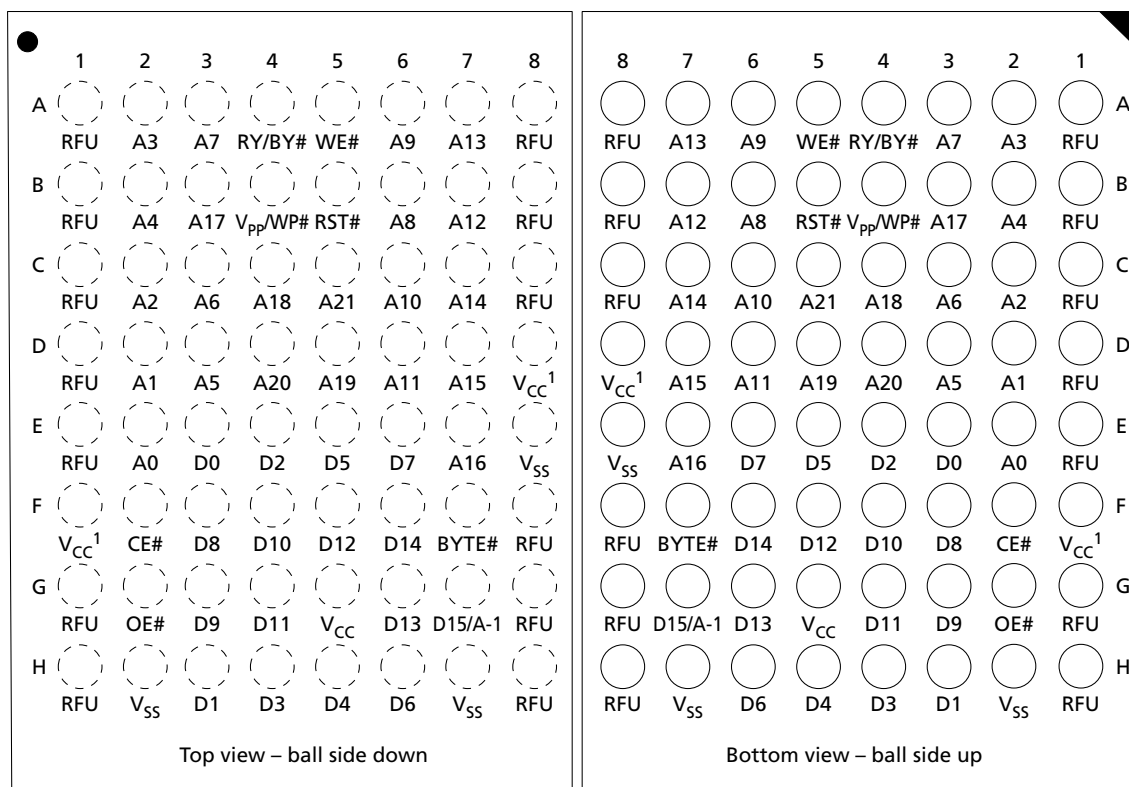
Note: 1. RFU = reserved for future use.

**Figure 4: 48-Ball TFBGA**



Note: 1. RFU = reserved for future use.

**Figure 5: 64-Ball TFBGA**



- Notes:
1. RFU = reserved for future use.
  2. Pads D8 and F1 are connected on the M29W640GT and M29W640GB devices.

## Signal Descriptions

The following table 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 6: Signal Descriptions**

| Name                 | Type  | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|----------------------|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| A[MAX:0]             | Input | <b>Address:</b> Select the cells in the memory array to access during bus READ operations. During bus WRITE operations they control the commands sent to the command interface of the program/erase controller.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| CE#                  | Input | <b>Chip enable:</b> Activates the memory, allowing bus READ and bus WRITE operations to be performed. When CE# is HIGH, all other pins are ignored.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| OE#                  | Input | <b>Output enable:</b> Controls the bus READ operation of the memory.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| WE#                  | Input | <b>Write enable:</b> Controls the bus WRITE operation of the memory's command interface.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| V <sub>pp</sub> /WP# | Input | <p><b>V<sub>pp</sub>/WP#:</b> Provides two functions: V<sub>pp</sub> enables the memory to use an external high-voltage power supply to reduce the time required for UNLOCK BYPASS PROGRAM operations. WP# performs hardware protection by protection the last block at the end of the addressable area (M29W640GH) or the first block at the beginning of the addressable area (M29W640GL). It protects the last two blocks at the end of the addressable area (M29W640GT) and the first two boot blocks at the beginning of the addressable area (M29W640GB).</p> <p>V<sub>pp</sub>/WP# may be left floating or unconnected (see DC Characteristics). When V<sub>pp</sub>/WP# is LOW, the last or first block in the M29W640GH and M29W640GL, respectively, and the last or first two blocks in the M29W640GT and M29W640GB, respectively, are protected. PROGRAM and ERASE operations in this block are ignored while V<sub>pp</sub>/WP# is LOW, even when RST# is at V<sub>ID</sub>.</p> <p>When V<sub>pp</sub>/WP# is HIGH, V<sub>IH</sub>, the device reverts to the previous protection status of the outermost blocks. PROGRAM and ERASE operations can now modify the data in the outermost blocks unless the block is protected using block protection.</p> <p>Applying 12V to V<sub>pp</sub>/WP# will temporarily unprotect any block previously protected (including the outermost blocks) using a high-voltage block protection technique (in-system or programmer technique). (See Hardware Protection for details. When V<sub>pp</sub>/WP# is raised to V<sub>pp</sub>, the device automatically enters the unlock bypass mode. When V<sub>pp</sub>/WP# returns to V<sub>IH</sub> or V<sub>IL</sub>, normal operation resumes. During UNLOCK BYPASS PROGRAM operations, the device draws I<sub>pp</sub> from the pin to supply the programming circuits. (See UNLOCK BYPASS Command.) The transitions from V<sub>IH</sub> to V<sub>pp</sub> and from V<sub>pp</sub> to V<sub>IH</sub> must be slower than t<sub>VHVPP</sub> (See the Accelerated Program Timing waveforms).</p> <p>Never raise V<sub>pp</sub>/WP# to V<sub>pp</sub> from any mode except read mode; otherwise, the device may be left in an indeterminate state.</p> <p>A 0.1μF capacitor should be connected between V<sub>pp</sub>/WP# and the V<sub>SS</sub> ground pin to decouple the current surges from the power supply. The PCB track widths must be sufficient to carry the currents required during an UNLOCK BYPASS PROGRAM operation, I<sub>pp</sub>.</p> |

**Table 6: Signal Descriptions (Continued)**

| Name     | Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DQ15/A-1 | I/O    | <p><b>Data I/O or address input:</b> When HIGH, behaves as a data I/O pin (as DQ8–DQ14). When LOW, behaves as an address pin; DQ15A–1 LOW will select the LSB of the addressed word; DQ15A–1 HIGH will select the MSB.</p> <p>Throughout the text, consider references to the data I/O to include this pin when BYTE# is HIGH and references to the address inputs to include this pin when BYTE# is LOW, except when stated explicitly otherwise.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| RST#     | Input  | <p><b>Reset/Block temporary unprotect:</b> Applies a hardware reset to the memory or temporarily unprotect all blocks that have been protected.</p> <p>Note that if <math>V_{PP}/WP</math> is at <math>V_{IL}</math>, then the last and the first block in the M29W640GH and M29W640GL, respectively, and the last two and first two blocks in the M29W640GT and M29W640GB, respectively, will remain protected, even if RST# is at <math>V_{ID}</math>.</p> <p>A hardware reset is achieved by holding RST# LOW for at least <math>t_{PLPX}</math>. After RST# goes HIGH, the memory will be ready for bus READ and bus WRITE operations after <math>t_{PHEL}</math> or <math>t_{RHEL}</math>, whichever occurs last. (See Reset Characteristics for more details.)</p> <p>Holding RST# at <math>V_{ID}</math> will temporarily unprotect the protected blocks in the memory. PROGRAM and ERASE operations on all blocks will be possible. The transition from <math>V_{IH}</math> to <math>V_{ID}</math> must be slower than <math>t_{PHPHH}</math>.</p> |
| DQ[14:8] | I/O    | <p><b>Data I/O:</b> Outputs the data stored at the selected address during a bus READ operation when BYTE# is HIGH. When BYTE# is LOW, these pins are not used and are High-Z. During bus WRITE operations, the command register does not use these bits. When reading the status register these bits should be ignored.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| DQ[7:0]  | I/O    | <p><b>Data I/O:</b> Outputs the data stored at the selected address during a bus READ operation. During bus WRITE operations, they represent the commands sent to the command interface of the program/erase controller.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| RY/BY#   | Output | <p><b>Ready busy:</b> Open-drain output that identifies when the device is performing a PROGRAM or ERASE operation. During PROGRAM or ERASE operations, RY/BY# is LOW, and is High-Z during read mode, auto select mode, and erase suspend mode.</p> <p>After a hardware reset, bus READ and WRITE operations cannot begin until RY/BY# becomes High-Z. (See Reset Characteristics for more details.)</p> <p>The use of an open-drain output enables RY/BY# pins from several devices to be connected to a single pull-up resistor. A LOW will then indicate that one, or more, of the devices is busy.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| BYTE#    | Input  | <p><b>BYTE#/Word organization select:</b> Switches between the x8 and x16 bus modes of the device. When LOW, the device is in x8 mode; when HIGH, it is in x16 mode.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

**Table 6: Signal Descriptions (Continued)**

| Name            | Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-----------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| V <sub>CC</sub> | Supply | <p><b>Supply voltage:</b> Provides the power supply for all operations (READ, PROGRAM, and ERASE).</p> <p>The command interface is disabled when the V<sub>CC</sub> supply voltage is less than the lockout voltage, V<sub>LKO</sub>. This prevents bus WRITE operations from accidentally damaging the data during power-up, power-down, and power surges. If the program/erase controller is programming or erasing during this time, then the operation aborts and the memory contents being altered will be invalid.</p> <p>A 0.1 µF capacitor should be connected between the V<sub>CC</sub> supply voltage pin and the V<sub>SS</sub> ground pin 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, I<sub>CC3</sub>.</p> |
| V <sub>SS</sub> | Supply | <p><b>Ground:</b> Reference for all voltage measurements. The device features two V<sub>SS</sub> pins which must be both connected to the system ground.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| RFU             | —      | <b>Reserved for future use:</b> RFUs should be not connected.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

**Table 7: Hardware Protection**

| V <sub>pp</sub> /WP#               | RST#                               |                         | Function                                                                                                                                                                           |
|------------------------------------|------------------------------------|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| V <sub>IL</sub>                    | V <sub>IH</sub>                    | M29W640GT and M29W640GB | Last 2 blocks at the end of the addressable area (M29W640GT) and first 2 blocks at the beginning of the addressable area (M29W640GB) protected from program/erase operations       |
|                                    |                                    | M29W640GH and M29W640GL | Last block at the end of the addressable area (M29W640GH) and first block at the beginning of the addressable area (M29W640GL) protected from program/erase operations             |
|                                    | V <sub>ID</sub>                    | M29W640GT and M29W640GB | All blocks temporarily unprotected except the last 2 blocks at the end of the addressable area (M29W640GT) and first 2 blocks at the beginning of the addressable area (M29W640GB) |
|                                    |                                    | M29W640GH and M29W640GL | All blocks temporarily unprotected except the last block at the end of the addressable area (M29W640GH) and first block at the beginning of the addressable area (M29W640GL)       |
| V <sub>IH</sub> or V <sub>ID</sub> | V <sub>ID</sub>                    |                         | All blocks temporarily unprotected                                                                                                                                                 |
| V <sub>PP</sub>                    | V <sub>IH</sub> or V <sub>ID</sub> |                         | All blocks temporarily unprotected                                                                                                                                                 |

## Memory Organization

### Memory Configuration

The memory array for M29W640GH and M29W640GL devices is organized into 128 uniform blocks of 64KB each for x8 and 32KW each for x16.

### Uniform Block Memory Map, x16 – 64Mb Density

**Table 8: x16 Uniform Blocks [127:0]**

| Block | Block Size | Address Range |           | Notes |
|-------|------------|---------------|-----------|-------|
|       |            | Start         | End       |       |
| 127   | 32KW       | 003F 8000     | 003F FFFF |       |
| 126   | 32KW       | 003F 0000     | 003F 7FFF |       |
| 125   | 32KW       | 003E 8000     | 003E FFFF |       |
| 124   | 32KW       | 003E 0000     | 003E 7FFF |       |
| ⋮     | ⋮          | ⋮             | ⋮         |       |
| 3     | 32KW       | 0001 8000     | 0001 FFFF | 1     |
| 2     | 32KW       | 0001 0000     | 0001 7FFF |       |
| 1     | 32KW       | 0000 8000     | 0000 FFFF |       |
| 0     | 32KW       | 0000 0000     | 0000 7FFF |       |

Note: 1. Used as the extended block addresses when the device is in extended block mode.

### Uniform Block Memory Map, x8 – 64Mb Density

**Table 9: x8 Uniform Blocks [127:0]**

| Block | Block Size | Address Range |           | Notes |
|-------|------------|---------------|-----------|-------|
|       |            | Start         | End       |       |
| 127   | 64KB       | 007F 0000     | 007F FFFF |       |
| 126   | 64KB       | 007E 0000     | 007E FFFF |       |
| 125   | 64KB       | 007D 0000     | 007D FFFF |       |
| 124   | 64KB       | 007C 0000     | 007C FFFF |       |
| ⋮     | ⋮          | ⋮             | ⋮         |       |
| 3     | 64KB       | 0003 0000     | 0003 FFFF | 1     |
| 2     | 64KB       | 0002 0000     | 0002 FFFF |       |
| 1     | 64KB       | 0001 0000     | 0001 FFFF |       |
| 0     | 64KB       | 0000 0000     | 0000 FFFF |       |

Note: 1. Used as the extended block addresses when the device is in extended block mode.

## Bus Operations

**Table 10: Bus Operations**

Notes 1 and 2 apply to entire table

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

- Notes:
1. Typical glitches of less than 5ns on CE# and WE# are ignored by the device and do not affect bus operations.
  2. H = Logic level HIGH ( $V_{IH}$ ); L = Logic level LOW ( $V_{IL}$ ); X = HIGH or LOW.
  3. If WP# = LOW, the highest/lowest block remains protected, depending on the line item.
  4. Data input is required when issuing a command sequence or performing data polling or block protection.

### Read

Bus READ operations read from the memory cells, registers, or CFI space. A valid READ operation requires setting the appropriate address on the address inputs, taking CE# and OE# LOW and holding WE# HIGH. Data I/O signals output the value.

### Write

Bus WRITE operations write to the command interface. A valid WRITE operation requires setting the appropriate address on the address inputs. These are latched by the command interface on the falling edge of CE# or WE#, whichever occurs last. Values on data I/O signals are latched by the command interface on the rising edge of CE# or WE#, whichever occurs first. OE# must remain HIGH during the entire operation.

### Standby and Automatic Standby

When the device is in read mode, driving CE# HIGH places the device in standby mode and drives data I/Os to High-Z. Supply current is reduced to standby ( $I_{CC2}$ ), by holding CE# within  $V_{CC} \pm 0.2V$ .

During PROGRAM or ERASE operations, the device continues to use the program/erase supply current ( $I_{CC3}$ ) until the operation completes.

Automatic standby enables low power consumption during read mode. When CMOS levels ( $V_{CC} \pm 0.2V$ ) drive the bus, and following a READ operation and a period of inactivity specified in DC Characteristics, the memory enters automatic standby as internal supply current is reduced to  $I_{CC2}$ . Data I/O signals still output data if a READ operation is in progress.

### Output Disable

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

## Status Register

Bus READ operations from any address, always read the status register during PROGRAM and ERASE operations. It is also read during erase suspend when an address within a block being erased is accessed.

### Data Polling Bit (DQ7)

The data polling bit can be used to identify whether the program/erase controller has successfully completed its operation or if it has responded to an erase suspend. The data polling bit is output on DQ7 when the status register is read.

During PROGRAM operations, DQ7 outputs the complement of the bit being programmed to DQ7. After successful completion of the PROGRAM operation, the memory returns to read mode and bus READ operations from the address just programmed output DQ7, not its complement.

During ERASE operations DQ7 outputs 0, the complement of the erased state of DQ7. After successful completion of the ERASE operation the memory returns to read mode.

In erase suspend mode, DQ7 will output a 1 during a bus READ operation within a block being erased. DQ7 will change from a 0 to a 1 when the program/erase controller has suspended the ERASE operation. The Data Polling Flowchart gives an example of how to use DQ7. A valid address is the address being programmed or an address within the block being erased.

### Toggle Bit (DQ6)

The toggle bit can be used to identify whether the program/erase controller has successfully completed its operation or if it has responded to an erase suspend. The toggle bit is output on DQ6 when the status register is read.

During PROGRAM and ERASE operations, DQ6 changes from 0 to 1 to 0, and so forth, with successive bus READ operations at any address. After successful completion of the operation, the memory returns to read mode.

During erase suspend mode, DQ6 will output when addressing a cell within a block being erased. DQ6 will stop toggling when the program/erase controller has suspended the ERASE operation.

The Data Toggle Flowchart gives an example of how to use DQ6 and the toggle and alternative toggle waveforms describe toggle bit timing.

### Error Bit (DQ5)

The error bit can be used to identify 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. If DQ5 is set, a READ/RESET command must be issued before other commands are issued. The error bit is output on DQ5 when the status register is read.

Note that the PROGRAM command cannot change a bit set to 0 back to 1 and attempting to do so will set DQ5 to 1. A bus READ operation to that address will show the bit remains 0. One of the ERASE commands must be used to set all the bits in a block or in the whole memory from 0 to 1.

## Erase Timer Bit (DQ3)

The erase timer bit can be used to identify the start of program/erase controller operation during a BLOCK ERASE command. After the program/erase controller starts erasing, DQ3 is set to 1. Before the program/erase controller starts, DQ3 is set to 0 and additional blocks to be erased may be written to the command interface. The erase timer bit is output on DQ3 when the status register is read.

## Alternative Toggle Bit (DQ2)

The alternative toggle bit can be used to monitor the program/erase controller during ERASE operations. It is output on DQ2 when the status register is read.

During CHIP ERASE and BLOCK ERASE operations, DQ2 changes from 0 to 1 to 0, and so forth, with successive bus READ operations from addresses within the blocks being erased. A protected block is treated the same as a block not being erased. After the operation completes, the memory returns to read mode.

During erase suspend, DQ2 changes from 0 to 1 to 0, and so forth, with successive bus READ operations from addresses within the blocks being erased. Bus READ operations to addresses within blocks not being erased will output the memory cell data as if in read mode.

After an ERASE operation that causes DQ5 to be set, DQ2 can be used to identify which block or blocks have caused the error. DQ2 changes from 0 to 1 to 0, and so forth, with successive bus READ operations from addresses within blocks that have not erased correctly. DQ2 does not change if the addressed block has erased correctly.

## Write to Buffer and Program Abort Bit (DQ1)

DQ1 is set to 1 when a WRITE TO BUFFER AND PROGRAM operation aborts. Otherwise, DQ1 is set to 0. The WRITE TO BUFFER AND PROGRAM ABORT AND RESET command must be issued to return the device to read mode (see Command Interface for more information).

**Table 11: Status Register Bits**

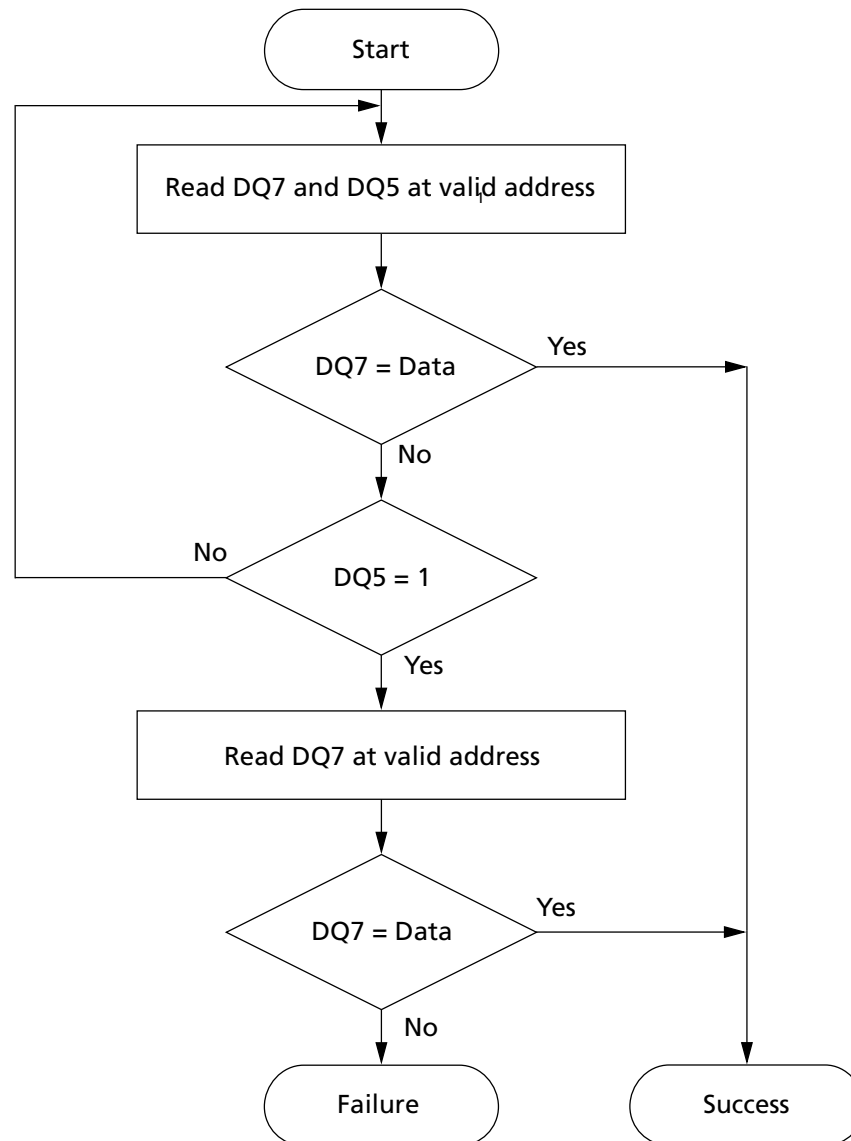
| Operation                         | Address           | DQ7  | DQ6    | DQ5 | DQ3 | DQ2       | DQ1 | RY/BY# |
|-----------------------------------|-------------------|------|--------|-----|-----|-----------|-----|--------|
| PROGRAM                           | Any address       | DQ7# | Toggle | 0   | –   | –         | 0   | 0      |
| PROGRAM DURING ERASE SUSPEND      | Any address       | DQ7# | Toggle | 0   | –   | –         | –   | 0      |
| WRITE TO BUFFER AND PROGRAM ABORT | Any address       | DQ7# | Toggle | 0   | –   | –         | 1   | 0      |
| WRITE TO BUFFER AND PROGRAM       | Any address       | DQ7# | Toggle | 0   | –   | –         | 0   | 0      |
| PROGRAM ERROR                     | Any address       | DQ7# | Toggle | 1   | –   | –         | –   | High-Z |
| CHIP ERASE                        | Any address       | 0    | Toggle | 0   | 1   | Toggle    | –   | 0      |
| BLOCK ERASE BEFORE TIMEOUT        | Erasing block     | 0    | Toggle | 0   | 0   | Toggle    | –   | 0      |
|                                   | Non-erasing block | 0    | Toggle | 0   | 0   | No Toggle | –   | 0      |

**Table 11: Status Register Bits (Continued)**

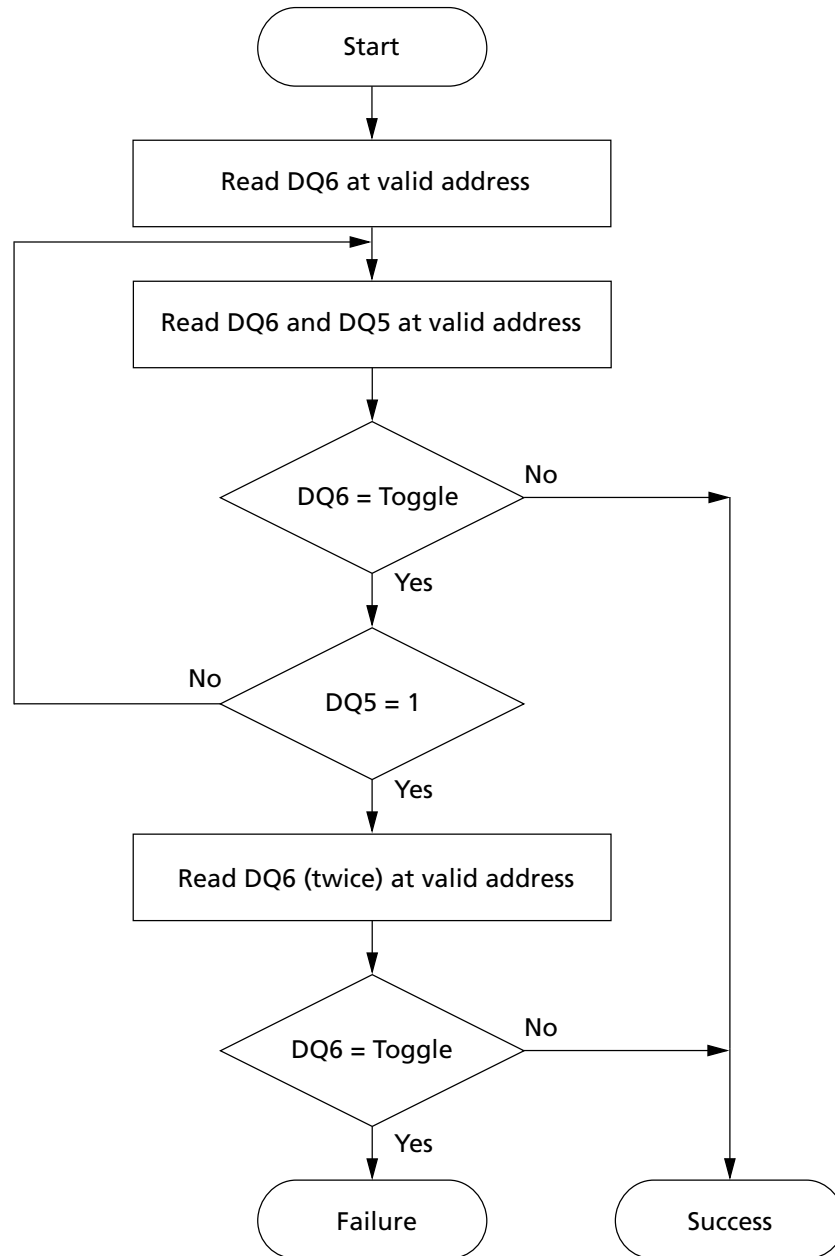
| Operation     | Address              | DQ7                 | DQ6       | DQ5 | DQ3 | DQ2       | DQ1 | RY/BY# |
|---------------|----------------------|---------------------|-----------|-----|-----|-----------|-----|--------|
| BLOCK ERASE   | Erasing block        | 0                   | Toggle    | 0   | 1   | Toggle    | –   | 0      |
|               | Non-erasing block    | 0                   | Toggle    | 0   | 1   | No Toggle | –   | 0      |
| ERASE SUSPEND | Erasing block        | 1                   | No Toggle | 0   | –   | Toggle    | –   | High-Z |
|               | Non-erasing block    | Data read as normal |           |     |     |           | –   | High-Z |
| ERASE ERROR   | Good block address   | 0                   | Toggle    | 1   | 1   | No Toggle | –   | High-Z |
|               | Faulty block address | 0                   | Toggle    | 1   | 1   | Toggle    | –   | High-Z |

Note: 1. Unspecified data bits should be ignored.

**Figure 6: Data Polling Flowchart**



**Figure 7: Data Toggle Flowchart**



## **READ Operations**

### **READ/RESET Command**

The READ/RESET (F0h) command returns the device to read mode and resets the errors in the status register. One or three bus WRITE operations can be used to issue the READ/RESET command.

To return the device to read mode, this command can be issued between bus WRITE cycles before the start of a PROGRAM or ERASE operation. If the READ/RESET command is issued during the timeout of a BLOCK ERASE operation, the device requires up to 10 $\mu$ s to abort, during which time no valid data can be read.

### **READ CFI Command**

The READ CFI (98h) command puts the device in read CFI mode and is valid only 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. A READ/RESET command must be issued to return the device to the previous mode (read array or auto select). A second READ/RESET command is required to put the device in read array mode from auto select mode.

## AUTO SELECT Operations

### 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 or by applying  $V_{ID}$  to A9. Auto select mode enables the following device information to be read:

- Electronic signature, which includes manufacturer and device code information.
- Block protection, which includes the block protection status and extended memory block protection indicator.

Electronic signature or block protection information is read by executing a READ operation with control signals and addresses set.

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.

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.

To enter auto select mode by applying  $V_{ID}$  to A9 (see the following tables).

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 12: Read Electronic Signature**

Note 1 applies to entire table

Note 1 Applies to entire table

| Signal                          | READ Cycle        |                 |                 |                 |                 |                 | Notes |
|---------------------------------|-------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-------|
|                                 | Manufacturer Code | Device Code 1   | Device Code 2   |                 | Device Code 3   |                 |       |
|                                 |                   |                 | GH/GL           | GT/GB           | GH/GT           | GL/GB           |       |
| CE#                             | L                 | L               | L               | L               | L               | L               |       |
| OE#                             | L                 | L               | L               | L               | L               | L               |       |
| WE#                             | H                 | H               | H               | H               | H               | H               |       |
| Address Input, 8-Bit and 16-Bit |                   |                 |                 |                 |                 |                 |       |
| A[MAX:10]                       | X                 | X               | X               | X               | X               | X               |       |
| A9                              | V <sub>ID</sub>   | V <sub>ID</sub> | V <sub>ID</sub> | V <sub>ID</sub> | V <sub>ID</sub> | V <sub>ID</sub> | 2     |
| A8                              | X                 | X               |                 | X               |                 | X               |       |
| A[7:5]                          | L                 | L               |                 | L               |                 | L               |       |
| A4                              | X                 | X               |                 | X               |                 | X               |       |
| A[3:1]                          | L                 | L               |                 | H               |                 | H               |       |
| A0                              | L                 | H               |                 | L               |                 | H               |       |
| Address Input, 8-Bit Only       |                   |                 |                 |                 |                 |                 |       |

**Table 12: Read Electronic Signature (Continued)**

Note 1 applies to entire table

| Signal                   | READ Cycle        |               |               |       |               |       | Notes |
|--------------------------|-------------------|---------------|---------------|-------|---------------|-------|-------|
|                          | Manufacturer Code | Device Code 1 | Device Code 2 |       | Device Code 3 |       |       |
|                          |                   |               | GH/GL         | GT/GB | GH/GT         | GL/GB |       |
| DQ[15]/A-1               | X                 | X             |               | X     |               | X     |       |
| Data I/O, 8-Bit Only     |                   |               |               |       |               |       |       |
| DQ[14:8]                 | X                 | X             |               | X     |               | X     |       |
| DQ[7:0]                  | 20h               | 7Eh           | 0Ch           | 10h   | 01h           | 00h   |       |
| Data I/O, 16-Bit Only    |                   |               |               |       |               |       |       |
| DQ[15]/A-1, and DQ[14:0] | 0020h             | 227Eh         | 220Ch         | 2210h | 2201h         | 2200h |       |

- Notes: 1. H = Logic level HIGH ( $V_{IH}$ ); L = Logic level LOW ( $V_{IL}$ ); X = HIGH or LOW.  
2. When using the AUTO SELECT command to enter auto select mode, applying  $V_{ID}$  to A9 is not required. A9 can be either  $V_{IL}$  or  $V_{IH}$ .

**Table 13: Block Protection**

Note 1 applies to entire table

| Operation                    | CE#             | OE#             | WE # | Address Input |     |           |     |           |                 |         |    |         |         |    | Data I/O |                                                                                                        |
|------------------------------|-----------------|-----------------|------|---------------|-----|-----------|-----|-----------|-----------------|---------|----|---------|---------|----|----------|--------------------------------------------------------------------------------------------------------|
|                              |                 |                 |      | A[MA X]       | A15 | A[14:1 3] | A12 | A[11:1 0] | A9              | A[8:7 ] | A6 | A[5:4 ] | A[3: 2] | A1 | A0       | DQ[15]/A-1, DQ[14:0]                                                                                   |
| BLOCK PROTECT (Group)        | L               | V <sub>ID</sub> | LP   | Block address |     |           |     | X         | V <sub>ID</sub> | X       |    |         |         |    |          | X                                                                                                      |
| CHIP UNPROTECT               | V <sub>ID</sub> | V <sub>ID</sub> | LP   | X             | H   | X         | H   | X         | V <sub>ID</sub> | X       |    |         |         |    |          | X                                                                                                      |
| VERIFY BLOCK PROTECTION      | L               | L               | H    | Block address |     |           |     | X         | V <sub>ID</sub> | X       | L  | X       | L       | H  | L        | Pass = xx01h<br>Retry = xx00h                                                                          |
| VERIFY BLOCK UNPROTECT       | L               | L               | H    | Block address |     |           |     | X         | V <sub>ID</sub> | X       | H  | X       | L       | H  | L        | Retry = xx01h<br>Pass = xx00h                                                                          |
| READ BLOCK PROTECTION STATUS | L               | L               | H    | Block address |     |           |     | X         | V <sub>ID</sub> | X       | L  | X       | L       | H  | L        | Protected (x8) = 01h<br>Unprotected (x8) = 00h<br>Protected (x16) = 0001h<br>Unprotected (x16) = 0000h |
| GL, GT, GB                   |                 |                 |      |               |     |           |     |           |                 |         |    |         |         |    |          |                                                                                                        |

**Table 13: Block Protection (Continued)**

Note 1 applies to entire table

Notes apply to both tables

| Operation                                               | CE# | OE# | WE# | Address Input    |     |              |     |              |                 |            |    |            |            |    |    | Data I/O                                                   |
|---------------------------------------------------------|-----|-----|-----|------------------|-----|--------------|-----|--------------|-----------------|------------|----|------------|------------|----|----|------------------------------------------------------------|
|                                                         |     |     |     | A[MA<br>X]       | A15 | A[14:1<br>3] | A12 | A[11:1<br>0] | A9              | A[8:7<br>] | A6 | A[5:4<br>] | A[3:<br>2] | A1 | A0 | DQ[15]/A-1,<br>DQ[14:0]                                    |
| READ EX-<br>TENDED<br>MEMORY<br>BLOCK<br>verify<br>CODE | L   | L   | H   | Block<br>address |     |              |     | X            | V <sub>ID</sub> | X          | L  | X          | L          | H  | H  | Factory-<br>locked = 98h<br>Customer-<br>lockable =<br>18h |
| GH                                                      |     |     |     |                  |     |              |     |              |                 |            |    |            |            |    |    |                                                            |
| READ EX-<br>TENDED<br>memory<br>BLOCK<br>verify<br>CODE | L   | L   | H   | Block<br>address |     |              |     | X            | V <sub>ID</sub> | X          | L  | X          | L          | H  | H  | Factory-<br>locked = 81h<br>Customer-<br>lockable =<br>01h |

Note: 1. H = Logic level HIGH (V<sub>IH</sub>); L = Logic level LOW (V<sub>IL</sub>); X = HIGH or LOW.

## Command Interface

All bus WRITE operations to the memory are interpreted by the command interface. Commands consist of one or more sequential bus WRITE operations. Failure to observe a valid sequence of bus WRITE operations will result in the memory returning to read mode. The long command sequences are imposed to maximize data security.

The address used for the commands changes depending on whether the memory is in 16-bit or 8-bit mode. See the x8 and x16 command tables, depending on the configuration that is being used, for a summary of the commands.

### READ/RESET Command

The READ/RESET command returns the memory to its read mode. It also resets the errors in the status register. Either one or three bus WRITE operations can be used to issue the READ/RESET command.

The READ/RESET command can be issued, between bus WRITE cycles before the start of a PROGRAM or ERASE operation, to return the device to read mode. If the READ/RESET command is issued during the timeout of a BLOCK ERASE operation, then the device will take up to 10 $\mu$ s to abort. During the abort period, no valid data can be read from the device. The READ/RESET command will not abort an ERASE operation when issued while in erase suspend.

### AUTO SELECT Command

The AUTO SELECT command is used to read the manufacturer code, the device code, the block protection status, and the extended memory block verify code. Three consecutive bus WRITE operations are required to issue the AUTO SELECT command. After the AUTO SELECT command is issued, the memory remains in auto select mode until a READ/RESET command is issued. READ CFI QUERY and READ/RESET commands are accepted in auto select mode, all other commands are ignored.

In auto select mode, the manufacturer code and the device code can be read by using a bus READ operation with addresses and control signals set, as shown Bus Operations, except for A9 (which is "Don't Care").

The block protection status of each block can be read using a bus READ operation with addresses and control signals set, as shown in Bus Operations, except for A9 (which is "Don't Care"). If the addressed block is protected, then 01h is output on DQ0–DQ7; otherwise, 00h is output (in 8-bit mode).

The protection status of the extended memory block, or extended memory block verify code, can be read using a bus READ operation with addresses and control signals, except for A9 (which is "Don't Care"). If the extended block is "factory-locked," then 80h is output on DQ0–DQ7; otherwise, 00h is output (8-bit mode).

### READ CFI QUERY Command

The READ CFI QUERY command is used to read data from the CFI. This command is valid when the device is in the read array mode, or when the device is in auto select mode.

One bus WRITE cycle is required to issue the READ CFI QUERY command. After the command is issued, subsequent bus READ operations read from the CFI.

The READ/RESET command must be issued to return the device to the previous mode (the read array mode or auto select mode). A second READ/RESET command would be needed if the device is to be put in the read array mode from auto selected mode.

## **PROGRAM Command**

The PROGRAM command can be used to program a value to one address in the memory array at a time. The command requires four bus WRITE operations; the final WRITE operation latches the address and data, and starts the program/erase controller.

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, then the PROGRAM command is ignored, the data remains unchanged. The status register is never read and no error condition is given.

During a PROGRAM operation, the memory will ignore all commands. It is not possible to issue any command to abort or pause the operation. Bus READ operations during the PROGRAM operation will output the status register on the data I/Os. (See Status Register for more details.)

After the PROGRAM operation has completed, the memory will return to the read mode, unless an error has occurred. When an error occurs, the memory will continue to output the status register. A READ/RESET command must be issued to reset the error condition and return to read mode.

Note that the PROGRAM command cannot change a bit set to 0 back to 1. One of the ERASE commands must be used to set all the bits in a block or in the whole memory from 0 to 1. (Refer to Program/Erase Characteristics.)

## **PROGRAM SUSPEND Command**

The PROGRAM SUSPEND command allows the system to interrupt a PROGRAM operation so that data can be read from any block. When the PROGRAM SUSPEND command is issued during a PROGRAM operation, the device suspends the PROGRAM operation within the program suspend latency time and updates the status register bits (see Program/Erase Characteristics).

After the PROGRAM operation has been suspended, the system can read array data from any address. However, data read from program-suspended addresses is not valid.

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 addresses not in ERASE SUSPEND or PROGRAM SUSPEND. If a read is needed from the extended block area (one-time program area), the user must use the proper command sequences to enter and exit this region.

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

## **PROGRAM RESUME Command**

After the PROGRAM RESUME command is issued, the device reverts to programming. The controller can determine the status of the PROGRAM operation using the DQ7 or DQ6 status bits, just as in the standard PROGRAM operation.

The system must write the PROGRAM RESUME command, to exit the program suspend mode and to continue the programming operation.

Further issuing of the RESUME command is ignored. Another PROGRAM SUSPEND command can be written after the device has resumed programming.

## **Fast Program Commands**

There are five fast program commands available to improve the programming throughput, by writing several adjacent words or bytes in parallel:

- QUADRUPLE and OCTUPLE BYTE PROGRAM, available for x8 operations
- DOUBLE and QUADRUPLE WORD PROGRAM, available for x16 operations
- WRITE TO BUFFER AND PROGRAM

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

## **DOUBLE BYTE PROGRAM Command**

The DOUBLE BYTE PROGRAM command is used to write a page of two adjacent bytes in parallel. The two bytes must differ only in DQ15A-1. Three bus WRITE cycles are necessary to issue the DOUBLE BYTE PROGRAM command:

The first bus cycle sets up the DOUBLE BYTE PROGRAM command; the second bus cycle latches the address and the data of the first byte to be written; and the third bus cycle latches the address and the data of the second byte to be written.

It is not necessary to raise  $V_{pp}/WP\#$  to 12V before issuing this command.

## **QUADRUPLE BYTE PROGRAM Command**

The QUADRUPLE BYTE PROGRAM command is used to write a page of four adjacent bytes in parallel. The four bytes must differ only for addresses A0, DQ15A-1. Five bus write cycles are necessary to issue the QUADRUPLE BYTE PROGRAM command:

The first bus cycle sets up the QUADRUPLE BYTE PROGRAM command; the second bus cycle latches the address and the data of the first byte to be written; the third bus cycle latches the address and the data of the second byte to be written; the fourth bus cycle latches the address and the data of the third byte to be written; and the fifth bus cycle latches the address and the data of the fourth byte to be written and starts the program/erase controller.

It is not necessary to raise  $V_{pp}/WP\#$  to 12V before issuing this command.

## **OCTUPLE BYTE PROGRAM Command**

This is used to write eight adjacent bytes, in x8 mode, simultaneously. The addresses of the eight bytes must differ only in A1, A0 and DQ15A-1.

12V must be applied to  $V_{PP}/WP\#$  prior to issuing an OCTUPLE BYTE PROGRAM command. Care must be taken because applying a 12V voltage to  $V_{PP}/WP\#$ , because it will temporarily unprotect any protected block.

Nine bus WRITE cycles are necessary to issue the command:

The first bus cycle sets up the command; the second bus cycle latches the address and the data of the first byte to be written; the third bus cycle latches the address and the data of the second byte to be written; the fourth bus cycle latches the address and the data of the third byte to be written; the fifth bus cycle latches the address and the data of the fourth byte to be written; the sixth bus cycle latches the address and the data of the fifth byte to be written; the seventh bus cycle latches the address and the data of the sixth byte to be written; the eighth bus cycle latches the address and the data of the seventh byte to be written; and the ninth bus cycle latches the address and the data of the eighth byte to be written and starts the program/erase controller.

## **DOUBLE WORD PROGRAM Command**

The DOUBLE WORD PROGRAM command is used to write a page of two adjacent words in parallel. The two words must differ only for the address A0.

Three bus WRITE cycles are necessary to issue the DOUBLE WORD PROGRAM command:

The first bus cycle sets up the DOUBLE WORD PROGRAM command; the second bus cycle latches the address and the data of the first word to be written; and the third bus cycle latches the address and the data of the second word to be written and starts the program/erase controller.

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

Note that the fast program commands cannot change a bit set to 0 back to 1. One of the ERASE commands must be used to set all the bits in a block or in the whole memory from 0 to 1.

Typical program times are given in Program/Erase Characteristics.

### **Note:**

It is not necessary to raise  $V_{PP}/WP\#$  to 12V before issuing this command.

## **QUADRUPLE WORD PROGRAM Command**

This is used to write a page of four adjacent words (or 8 adjacent bytes), in x16 mode, simultaneously. The addresses of the four words must differ only in A1 and A0.

12V must be applied to  $V_{PP}/WP\#$  prior to issuing a QUADRUPLE BYTE PROGRAM command. Care must be taken because applying a 12V voltage to  $V_{PP}/WP\#$ , because it will temporarily unprotect any protected block.

Five bus WRITE cycles are necessary to issue the command:

The first bus cycle sets up the command; the second bus cycle latches the address and the data of the first word to be written; the third bus cycle latches the address and the data of the second word to be written; the fourth bus cycle latches the address and the

data of the third word to be written; and the fifth bus cycle latches the address and the data of the fourth word to be written and starts the program/erase controller.

## **WRITE TO BUFFER AND PROGRAM Command**

The WRITE TO BUFFER AND PROGRAM command makes use of the device's 32-byte write buffer to speed up programming. 16 words/32 bytes can be loaded into the write buffer. Each write buffer has the same A4–A22 addresses. The WRITE TO BUFFER AND PROGRAM command dramatically reduces system programming time compared to the standard nonbuffered PROGRAM command.

When issuing a WRITE TO BUFFER AND PROGRAM command,  $V_{PP}/WP\#$  can be either held HIGH or raised to  $V_{PPH}$ .

Five successive steps are required to issue the WRITE TO BUFFER AND PROGRAM command: The WRITE TO BUFFER AND PROGRAM command starts with two UNLOCK cycles. The third bus WRITE cycle sets up the WRITE TO BUFFER AND PROGRAM command. The setup code can be addressed to any location within the targeted block. The fourth bus WRITE cycle sets up the number of words to be programmed. Value  $n$  is written to the same block address, where  $n + 1$  is the number of words to be programmed. The value of  $n + 1$  must not exceed the size of the write buffer or the operation will abort. The fifth cycle loads the first address and data to be programmed. The value of  $n$  bus WRITE cycles is used to load the address and data for each word into the write buffer. Addresses must lie within the range from the start address+1 to the start address +  $n - 1$ . Optimum performance is obtained when the start address corresponds to a 64-byte boundary. If the start address is not aligned to a 64-byte boundary, the total programming time is doubled.

All the addresses used in the WRITE TO BUFFER AND PROGRAM operation must lie within the same page. If an address is written several times during a WRITE TO BUFFER AND 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 failing to follow the correct sequence of bus WRITE cycles will abort the WRITE TO BUFFER AND PROGRAM operation.

DQ1, DQ5, DQ6, DQ7 can be used to monitor the device status during a WRITE TO BUFFER AND PROGRAM operation. It is possible to detect PROGRAM operation fails when changing programmed data from 0 to 1; that is, when reprogramming data in a portion of memory already programmed. The resulting data will be the logical OR between the previous value and the current value.

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

A WRITE TO BUFFER AND PROGRAM ABORT AND RESET command must be issued to abort the WRITE TO BUFFER AND PROGRAM operation and reset the device in read mode.

The write 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 step
- Write to an address in a block different 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 write 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 = DATA# (for the last address location loaded), DQ6 = toggle, and DQ5 = 0. A WRITE TO BUFFER ABORT AND RESET command sequence must be written to reset the device for the next operation. Note that the full three-cycle WRITE TO BUFFER ABORT AND RESET command sequence is required when using write-buffer-programming features in unlock bypass mode.

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

The WRITE TO BUFFER AND PROGRAM CONFIRM command is used to confirm a WRITE TO BUFFER AND PROGRAM command and to program the  $n + 1$  words loaded in the write buffer by this command.

## **WRITE TO BUFFER AND PROGRAM ABORT AND RESET Command**

The WRITE TO BUFFER AND PROGRAM ABORT AND RESET command is used to reset the device after a WRITE TO BUFFER AND PROGRAM command has been aborted.

## **UNLOCK BYPASS Command**

The UNLOCK BYPASS command is used in conjunction with the UNLOCK BYPASS PROGRAM command to program the memory faster than with the standard PROGRAM commands. When the cycle time to the device is long, considerable time saving can be made by using these commands. Three bus WRITE operations are required to issue the UNLOCK BYPASS command.

After the UNLOCK BYPASS command has been issued, the memory will only accept the UNLOCK BYPASS PROGRAM command and the UNLOCK BYPASS RESET command. The memory can be read as if in read mode.

When  $V_{pp}$  is applied to  $V_{pp}/WP\#$ , the memory automatically enters the unlock bypass mode and the UNLOCK BYPASS PROGRAM command can be issued immediately.

## **UNLOCK BYPASS PROGRAM Command**

The UNLOCK BYPASS command is used in conjunction with the UNLOCK BYPASS PROGRAM command to program the memory. When the cycle time to the device is long, considerable time saving can be made by using these commands. Three bus WRITE operations are required to issue the UNLOCK BYPASS command.

After the UNLOCK BYPASS command has been issued, the memory will only accept the UNLOCK BYPASS PROGRAM command and the UNLOCK BYPASS RESET command. The memory can be read as if in read mode.

The memory offers accelerated PROGRAM operations through  $V_{pp}/WP\#$ . When the system asserts  $V_{pp}$  on  $V_{pp}/WP\#$ , the memory automatically enters the unlock bypass mode. The system may then write the two-cycle UNLOCK BYPASS PROGRAM command sequence. The memory uses the higher voltage on  $V_{pp}/WP\#$  to accelerate the UNLOCK BYPASS PROGRAM operation.

Never raise  $V_{pp}/WP\#$  to  $V_{pp}$  from any mode except read mode; otherwise, the memory may be left in an indeterminate state.

## **UNLOCK BYPASS RESET Command**

The UNLOCK BYPASS RESET command can be used to return to read/reset mode from unlock bypass mode. Two bus WRITE operations are required to issue the UNLOCK BYPASS RESET command. A READ/RESET command does not exit from unlock bypass mode.

## **CHIP ERASE Command**

The CHIP ERASE command can be used to erase the entire chip. Six bus WRITE operations are required to issue the CHIP ERASE command and start the program/erase controller.

If any blocks are protected, then these are ignored and all the other blocks are erased. If all of the blocks are protected the CHIP ERASE operation appears to start but will terminate within about 100 $\mu$ s, leaving the data unchanged. No error condition is given when protected blocks are ignored.

During the ERASE operation the memory will ignore all commands, including the ERASE SUSPEND command. It is not possible to issue any command to abort the operation. All bus READ operations during the CHIP ERASE operation will output the status register on the data I/Os.

After the CHIP ERASE operation has completed, the memory will return to the read mode, unless an error has occurred. When an error occurs the memory will continue to output the status register. A READ/RESET command must be issued to reset the error condition and return to read mode.

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

## **BLOCK ERASE Command**

The BLOCK ERASE command can be used to erase a list of one or more blocks. 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. The BLOCK ERASE operation starts the program/erase controller about 50 $\mu$ s after the last bus WRITE operation. After the program/erase controller starts, it is not possible to select any more blocks. Each additional block must therefore be selected within 50 $\mu$ s of the last block. The 50 $\mu$ s timer restarts when an additional block is selected. The status register can be read after the sixth bus WRITE operation. ( See the status register section for details on how to identify whether the program/erase controller has started the BLOCK ERASE operation.)

If any selected blocks are protected, then these are ignored and all the other selected blocks are erased. If all of the selected blocks are protected, the BLOCK ERASE operation appears to start but will terminate within about 100 $\mu$ s, leaving the data unchanged. No error condition is given when protected blocks are ignored.

During the BLOCK ERASE operation, the memory will ignore all commands except the ERASE SUSPEND command. (Typical block erase times are given in the Program/Erase Characteristics.) All bus READ operations during the BLOCK ERASE operation will output the status register on the data I/Os.

After the BLOCK ERASE operation has completed, the memory will return to the read mode, unless an error has occurred. When an error occurs, the memory will continue to

output the status register. A READ/RESET command must be issued to reset the error condition and return to read mode.

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

## **ERASE SUSPEND Command**

The ERASE SUSPEND command may be used to temporarily suspend a BLOCK ERASE operation and return the memory to read mode. The command requires one bus WRITE operation.

The program/erase controller will suspend within the erase suspend latency time of the ERASE SUSPEND command being issued. After the program/erase controller has stopped, the memory will be set to read mode and the erase will be suspended. If the ERASE SUSPEND command is issued during the period when the memory is waiting for an additional block (before the program/erase controller starts), then the erase is suspended immediately and will start immediately when the ERASE RESUME command is issued. It is not possible to select any further blocks to erase after the ERASE RESUME.

During ERASE SUSPEND, it is possible to read and program cells in blocks that are not being erased; both READ and PROGRAM operations behave as normal on these blocks. If any attempt is made to program in a protected block or in the suspended block, then the PROGRAM command is ignored and the data remains unchanged. The status register is not read and no error condition is given. Reading from blocks that are being erased will output the status register.

It is also possible to issue the AUTO SELECT, READ CFI QUERY, and UNLOCK BYPASS commands during an ERASE SUSPEND. The READ/RESET command must be issued to return the device to read array mode before the RESUME command will be accepted.

## **ERASE RESUME Command**

The ERASE RESUME command must be used to restart the program/erase controller after an erase suspend. 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.

## **ENTER EXTENDED BLOCK Command**

The device has an extra 256-byte block (extended block) that can only be accessed using the ENTER EXTENDED BLOCK command. Three bus WRITE cycles are required to issue the ENTER EXTENDED BLOCK command. After the command has been issued, the device enters extended block mode where all bus READ or WRITE operations to the boot block addresses access the extended block. The extended block (with the same address as the boot blocks) cannot be erased, and can be treated as OTP memory. In extended block mode, the boot blocks are not accessible.

The extended block can be protected; however, once protected, the protection cannot be undone.

## **EXIT EXTENDED BLOCK Command**

The EXIT EXTENDED BLOCK command is used to exit from the extended block mode and return the device to read mode. Four bus WRITE operations are required to issue the command.

## **BLOCK PROTECT and CHIP UNPROTECT Commands**

Groups of blocks can be protected against accidental program or erase. (See Memory Organization for the protection groups.) The whole chip can be unprotected to allow the data inside the blocks to be changed.

### **BLOCK PROTECT Command**

Block protection can be used to prevent any operation from modifying the data stored in the Flash. Each block can be protected individually. Once protected, PROGRAM and ERASE operations on the block fail to change the data.

There are three techniques that can be used to control block protection. These are programmer technique, in-system technique, and temporary unprotect. Temporary unprotect is controlled by RST#.

Unlike the command interface of the program/erase controller, the techniques for protecting and unprotecting blocks change between different Flash memory suppliers. Care should be taken when changing drivers for one part to work on another.

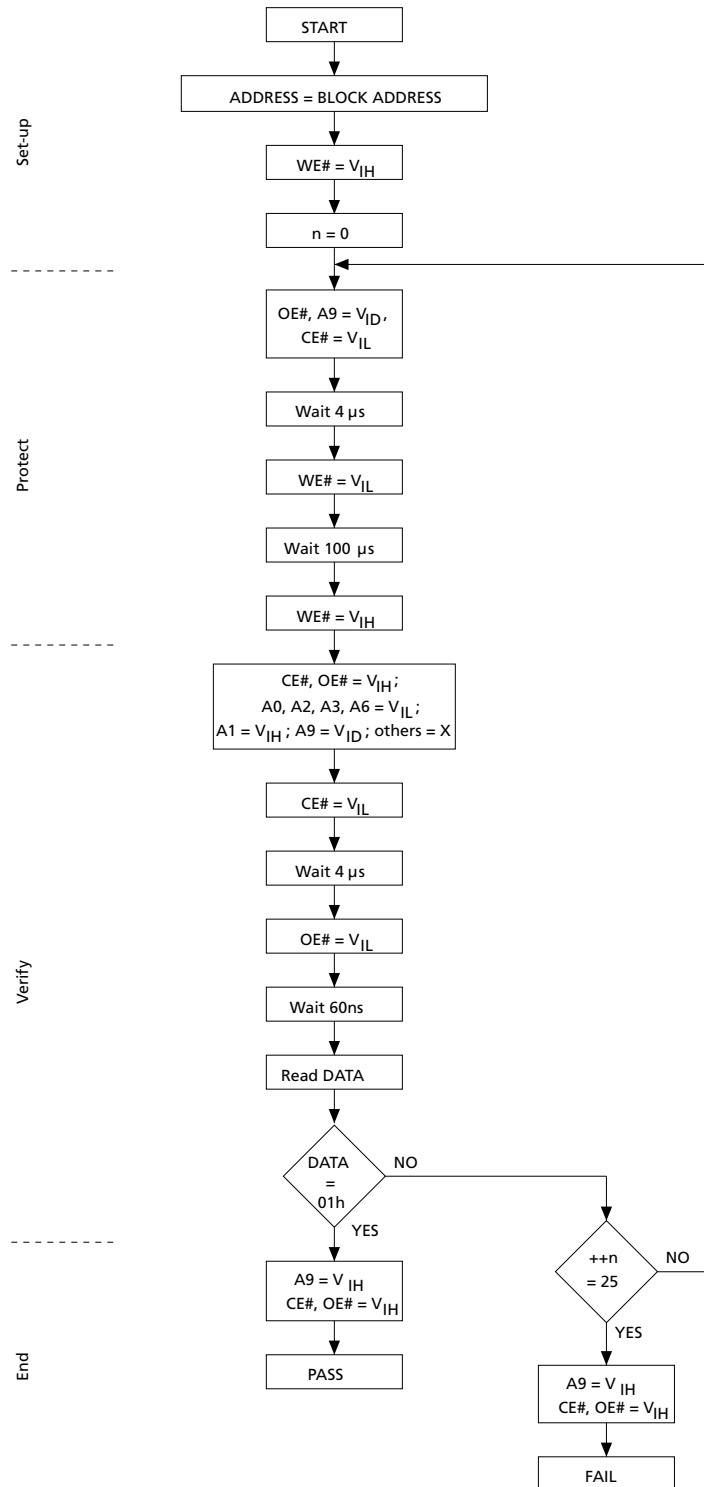
### **Programmer Technique**

The programmer technique uses high voltage levels ( $V^{ID}$ ) on some of the bus pins. These cannot be achieved using a standard microprocessor bus; therefore, the technique is recommended only for use in programming equipment.

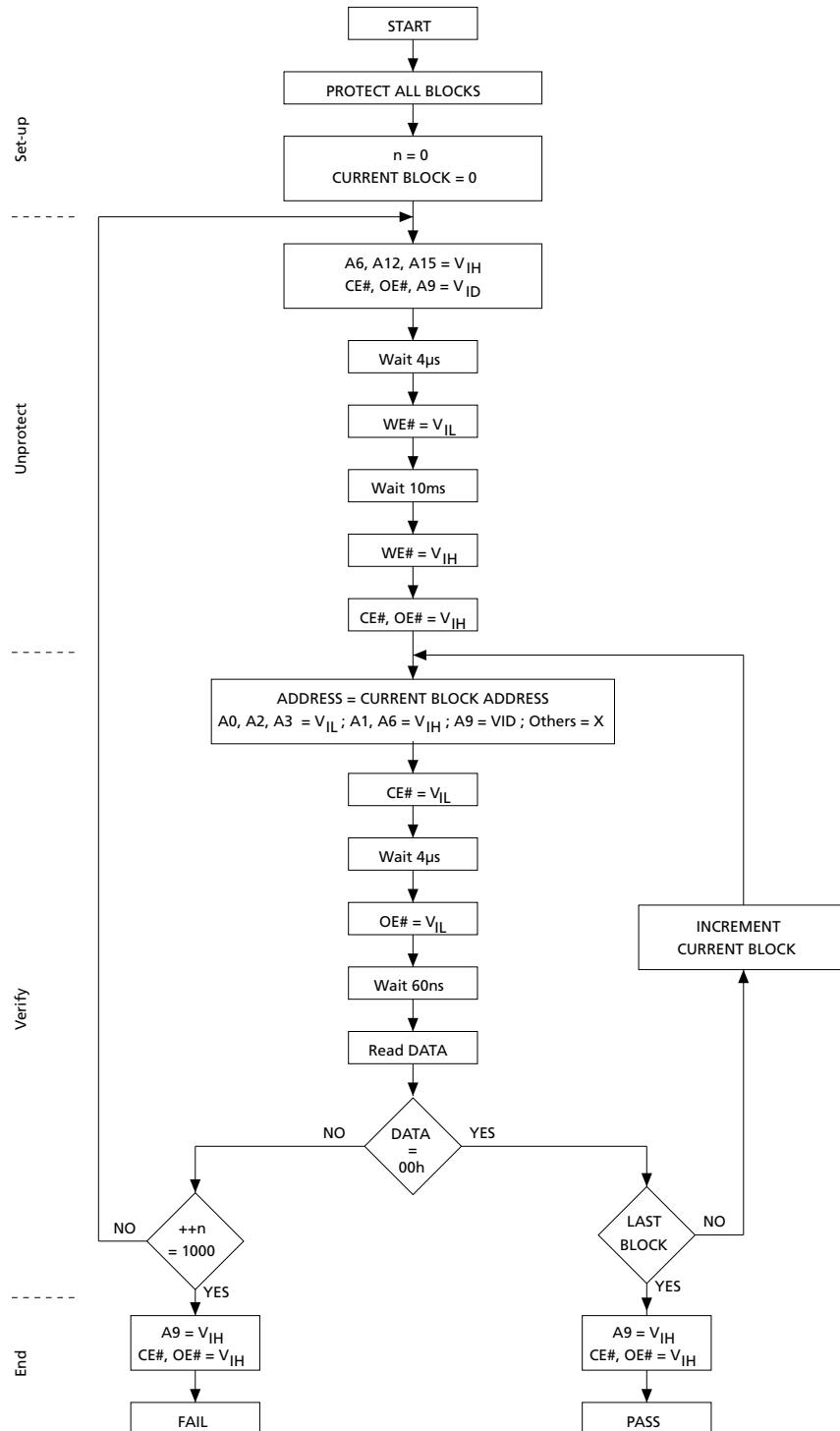
To protect a block, follow the steps in the following figure. To unprotect the whole chip, it is necessary to protect all of the blocks first, then all blocks can be unprotected at the same time. (See the Programmer Equipment Chip Protect Flowchart.)

The timing on these flowcharts is critical. Care should be taken to ensure that, where a pause is specified, it is followed as closely as possible. Do not abort the procedure before reaching the end. Chip unprotect can take several seconds and a user message should be provided to show that the operation is progressing.

**Figure 8: Programmer Equipment Block Protect Flowchart**



**Figure 9: Programmer Equipment Chip Unprotect Flowchart**



**Table 14: Programmer Technique Bus Operations**

Notes 1 and 2 apply to entire table

| Operation               | CE#             | OE#             | WE#             | Address Inputs                                                                                                                                 | Data I/Os                     |
|-------------------------|-----------------|-----------------|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------|
|                         |                 |                 |                 | A[MAX:0]                                                                                                                                       | DQ15/A-1, DQ[14:0]            |
| BLOCK PROTECT           | L               | V <sub>ID</sub> | L pulse         | A9 = V <sub>ID</sub><br>A[21:12] block addresses<br>others = X                                                                                 | X                             |
| CHIP UNPROTECT          | V <sub>ID</sub> | V <sub>ID</sub> | L pulse         | A9 = V <sub>ID</sub><br>A12 = V <sub>IH</sub><br>A15 = V <sub>IH</sub><br>others = X                                                           | X                             |
| BLOCK PROTECTION VERIFY | L               | L               | V <sub>IH</sub> | A0, A2, A3 = V <sub>IL</sub><br>A1 = V <sub>IH</sub><br>A6 = V <sub>IL</sub><br>A9 = V <sub>ID</sub><br>A[21:12] block addresses<br>others = X | Pass = XX01h<br>Retry = XX00h |
| BLOCK UNPROTECT VERIFY  | L               | L               | V <sub>IH</sub> | A0, A2, A3 = V <sub>IL</sub><br>A1 = V <sub>IH</sub><br>A6 = V <sub>IH</sub><br>A9 = V <sub>ID</sub><br>A[21:12] block addresses<br>others = X | Retry = XX01h<br>Pass = XX00h |

- Notes:
1. Typical glitches of less than 5ns on CE# 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.

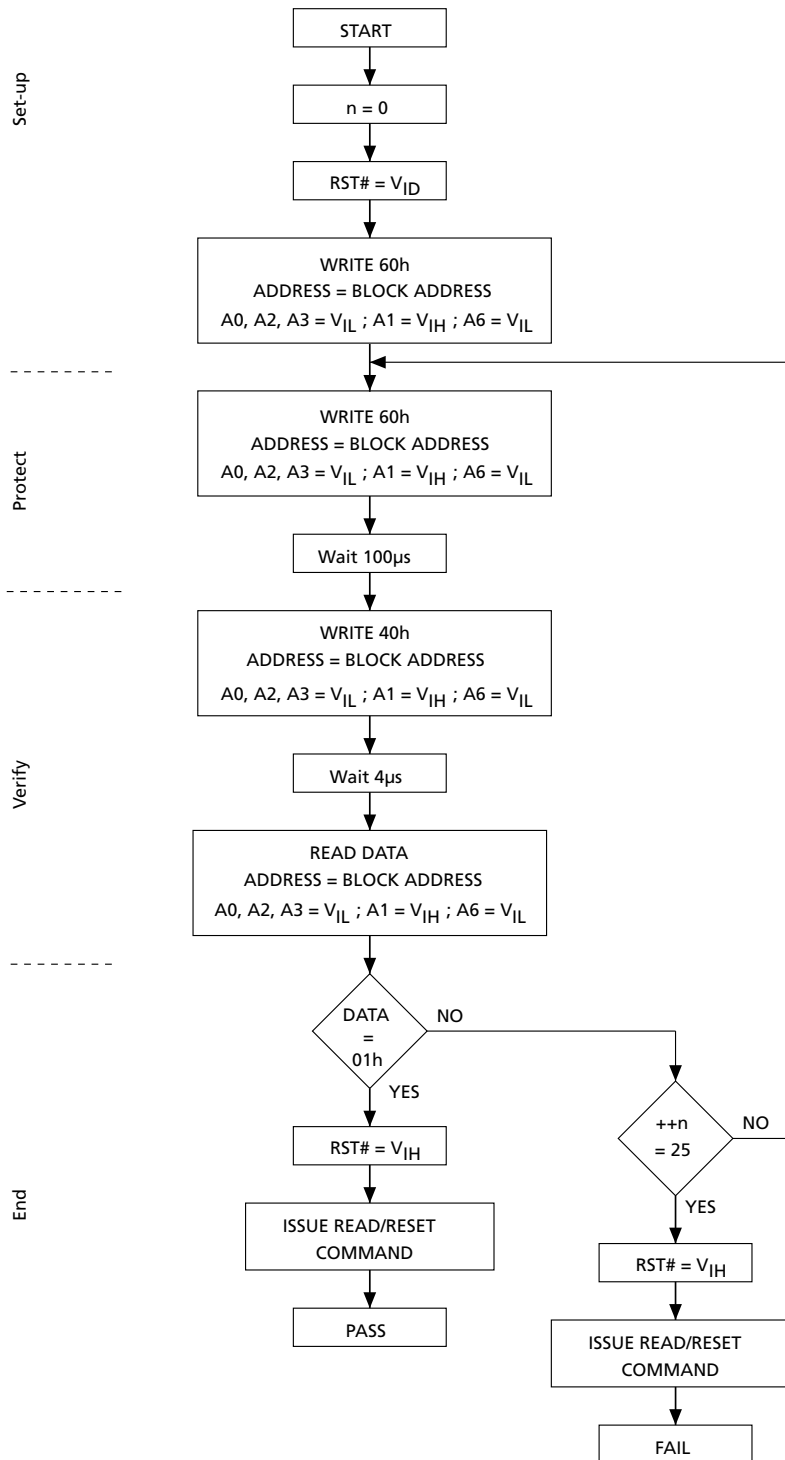
## In-System Technique

The in-system technique requires a high-voltage level on RST#. This can be achieved without violating the maximum ratings of the components on the microprocessor bus; therefore, this technique is suitable for use after the Flash has been fitted to the system.

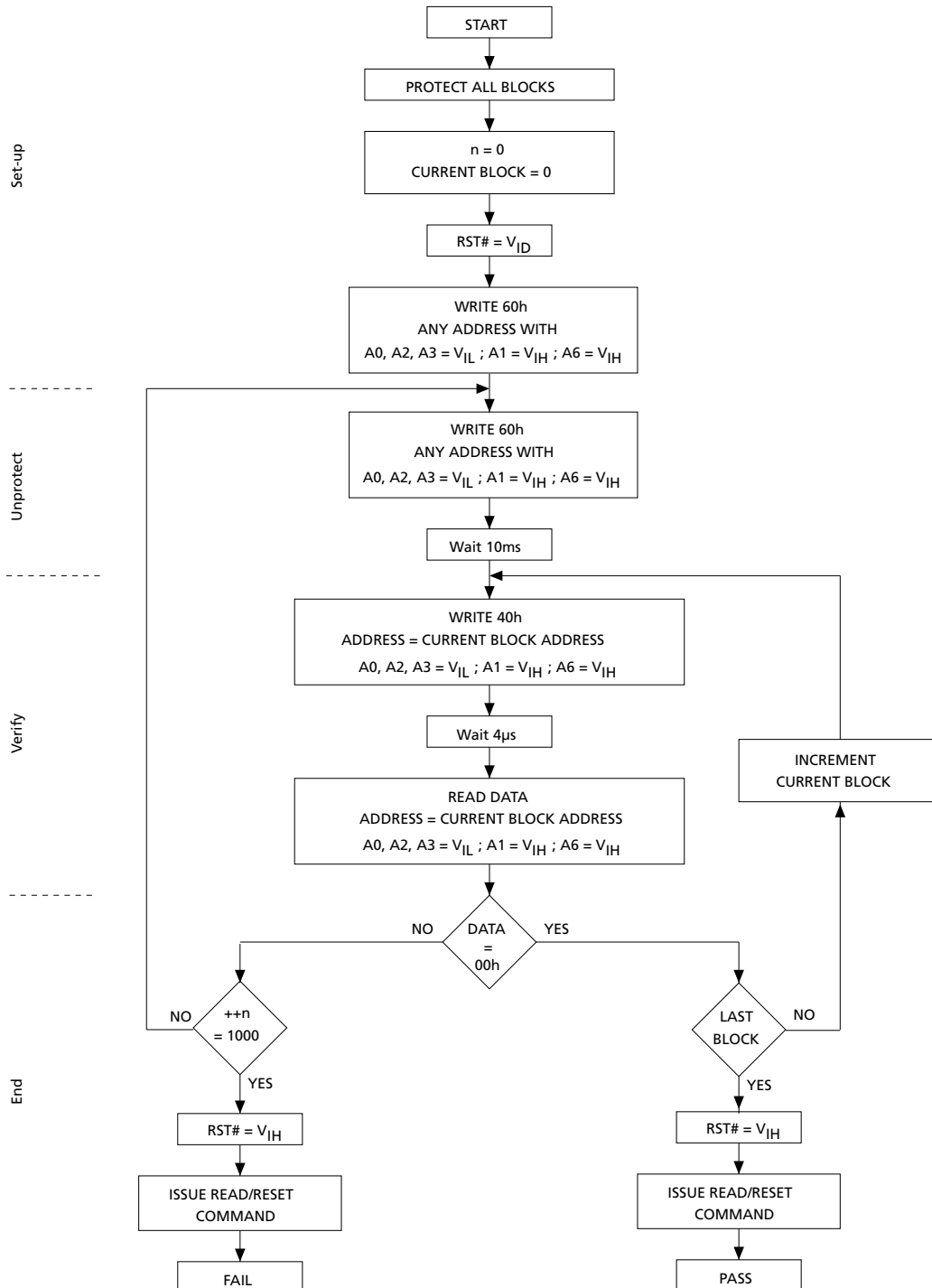
To protect a block, follow the steps in the following figure. To unprotect the whole chip, it is necessary to protect all of the blocks first, then all the blocks can be unprotected at the same time. (See the In-System Equipment Chip Unprotect Flowchart.)

The timing on these flowcharts is critical. Care should be taken to ensure that, where a pause is specified, it is followed as closely as possible. Do not allow the microprocessor to service interrupts that will upset the timing and do not abort the procedure before reaching the end. Chip unprotect can take several seconds and a user message should be provided to show that the operation is progressing.

**Figure 10: In-System Equipment Block Protect Flowchart**



**Figure 11: In-System Equipment Chip Protect Flowchart**



**Table 15: Commands – 16-Bit Mode (BYTE# = V<sub>IH</sub>)**

| Command                                     | Length | Bus WRITE Operations  |      |      |      |      |      |      |                      |                       |      |                        |      |
|---------------------------------------------|--------|-----------------------|------|------|------|------|------|------|----------------------|-----------------------|------|------------------------|------|
|                                             |        | 1st                   |      | 2nd  |      | 3rd  |      | 4th  |                      | 5th                   |      | 6th                    |      |
|                                             |        | Addr                  | Data | Addr | Data | Addr | Data | Addr | Data                 | Addr                  | Data | Addr                   | Data |
| READ/RESET                                  | 1      | X                     | F0   |      |      |      |      |      |                      |                       |      |                        |      |
|                                             | 3      | 555                   | AA   | 2AA  | 55   | X    | F0   |      |                      |                       |      |                        |      |
| AUTO SELECT                                 | 3      | 555                   | AA   | 2AA  | 55   | 555  | 90   |      |                      |                       |      |                        |      |
| READ CFI QUERY                              | 1      | 55                    | 98   |      |      |      |      |      |                      |                       |      |                        |      |
| PROGRAM                                     | 4      | 555                   | AA   | 2AA  | 55   | 555  | A0   | PA   | PD                   |                       |      |                        |      |
| DOUBLE WORD PROGRAM                         | 3      | 555                   | 50   | PA0  | PD0  | PA1  | PD1  |      |                      |                       |      |                        |      |
| QUADRUPLE WORD PROGRAM                      | 5      | 555                   | 56   | PA0  | PD0  | PA1  | PD1  | PA2  | PD2                  | PA3                   | PD3  |                        |      |
| UNLOCK BY-PASS                              | 3      | 555                   | AA   | 2AA  | 55   | 555  | 20   |      |                      |                       |      |                        |      |
| UNLOCK BY-PASS PROGRAM                      | 2      | X                     | A0   | PA   | PD   |      |      |      |                      |                       |      |                        |      |
| UNLOCK BY-PASS RESET                        | 2      | X                     | 90   | X    | 00   |      |      |      |                      |                       |      |                        |      |
| WRITE TO BUFFER AND PROGRAM                 | N + 5  | 555                   | AA   | 2AA  | 55   | BA   | 25   | BA   | N <sup>on page</sup> | PA <sup>on page</sup> | PD   | WBL <sup>on page</sup> | PD   |
| WRITE TO BUFFER AND PROGRAM ABORT AND RESET | 3      | 555                   | AA   | 2AA  | 55   | 555  | F0   |      |                      |                       |      |                        |      |
| WRITE TO BUFFER AND PROGRAM CONFIRM         | 1      | BA <sup>on page</sup> | 29   |      |      |      |      |      |                      |                       |      |                        |      |
| CHIP ERASE                                  | 6      | 555                   | AA   | 2AA  | 55   | 555  | 80   | 555  | AA                   | 2AA                   | 55   | 555                    | 10   |
| BLOCK ERASE                                 | 6+     | 555                   | AA   | 2AA  | 55   | 555  | 80   | 555  | AA                   | 2AA                   | 55   | BA                     | 30   |
| PROGRAM/ERASE SUSPEND                       | 1      | X                     | B0   |      |      |      |      |      |                      |                       |      |                        |      |
| PROGRAM/ERASE RESUME                        | 1      | X                     | 30   |      |      |      |      |      |                      |                       |      |                        |      |
| ENTER EXTENDED BLOCK                        | 3      | 555                   | AA   | 2AA  | 55   | 555  | 88   |      |                      |                       |      |                        |      |

**Table 15: Commands – 16-Bit Mode (BYTE# = V<sub>IH</sub>) (Continued)**

| Command             | Length | Bus WRITE Operations |      |      |      |      |      |      |      |      |      |      |      |
|---------------------|--------|----------------------|------|------|------|------|------|------|------|------|------|------|------|
|                     |        | 1st                  |      | 2nd  |      | 3rd  |      | 4th  |      | 5th  |      | 6th  |      |
|                     |        | Addr                 | Data | Addr | Data | Addr | Data | Addr | Data | Addr | Data | Addr | Data |
| EXIT EXTENDED BLOCK | 4      | 555                  | AA   | 2AA  | 55   | 555  | 90   | X    | 00   |      |      |      |      |

- Notes:
1. X = " Don't Care;" PA = Program address; PD = Program data; BA = Any address in the block. All values in the table are in hexadecimal. The command interface only uses A-1; A0-A10 and DQ0-DQ7 to verify the commands; A11-A20, DQ8-DQ14 and DQ15 are "Don't Care." DQ15A-1 is A-1 when BYTE# is V<sub>IL</sub> or DQ15 when BYTE# is V<sub>IH</sub>.
  2. The maximum number of cycles in the command sequence is 36. N + 1 is the number of words to be programmed during the WRITE TO BUFFER AND PROGRAM operation.
  3. Each buffer has the same A4-A22 addresses. A0-A3 are used to select a word within the N + 1 word page.
  4. The 6th cycle has to be issued N time. WBL scans the word inside the page.
  5. BA must be identical to the address loaded during the WRITE TO BUFFER AND PROGRAM 3rd and 4th cycles.

**Table 16: Commands – 8-Bit Mode (BYTE# = V<sub>IL</sub>)**

| Command                                     | Length | Bus WRITE Operations |      |      |      |      |      |      |                |                 |      |                  |                  |
|---------------------------------------------|--------|----------------------|------|------|------|------|------|------|----------------|-----------------|------|------------------|------------------|
|                                             |        | 1st                  |      | 2nd  |      | 3rd  |      | 4th  |                | 5th             |      | 6th              |                  |
|                                             |        | Addr                 | Data | Addr | Data | Addr | Data | Addr | Data           | Addr            | Data | Addr             | Data             |
| READ/RESET                                  | 1      | X                    | F0   |      |      |      |      |      |                |                 |      |                  |                  |
|                                             | 3      | AAA                  | AA   | 555  | 55   | X    | F0   |      |                |                 |      |                  |                  |
| AUTO SELECT                                 | 3      | AAA                  | AA   | 555  | 55   | AAA  | 90   |      |                |                 |      |                  |                  |
| READ CFI QUERY                              | 1      | AA                   | 98   |      |      |      |      |      |                |                 |      |                  |                  |
| PROGRAM                                     | 4      | AAA                  | AA   | 555  | 55   | AAA  | A0   | PA   | PD             |                 |      |                  |                  |
| DOUBLE BYTE PROGRAM                         | 3      | AAA                  | 50   | PA0  | PD0  | PA1  | PD1  |      |                |                 |      |                  |                  |
| QUADRUPLE BYTE PROGRAM                      | 5      | AAA                  | 56   | PA0  | PD0  | PA1  | PD1  | PA2  | PD2            | PA3             | PD3  |                  |                  |
| OCTUPLE BYTE PROGRAM                        | 9      | AAA                  | 8B   | PA0  | PD0  | PA1  | PD1  | PA2  | PD2            | PA3             | PD3  | PA4              | PD4 <sup>2</sup> |
| WRITE TO BUFFER AND PROGRAM                 | N + 5  | AAA                  | AA   | 555  | 55   | BA   | 25   | BA   | N <sup>2</sup> | PA <sup>4</sup> | PD   | WBL <sup>5</sup> | PD               |
| WRITE TO BUFFER AND PROGRAM ABORT AND RESET | 3      | AAA                  | AA   | 555  | 55   | AAA  | F0   |      |                |                 |      |                  |                  |
| WRITE TO BUFFER AND PROGRAM CONFIRM         | 1      | BA <sup>6</sup>      | 29   |      |      |      |      |      |                |                 |      |                  |                  |
| UNLOCK BY-PASS                              | 3      | AAA                  | AA   | 555  | 55   | AAA  | 20   |      |                |                 |      |                  |                  |
| UNLOCK BY-PASS PROGRAM                      | 2      | X                    | A0   | PA   | PD   |      |      |      |                |                 |      |                  |                  |
| UNLOCK BY-PASS RESET                        | 2      | X                    | 90   | X    | 00   |      |      |      |                |                 |      |                  |                  |
| CHIP ERASE                                  | 6      | AAA                  | AA   | 555  | 55   | AAA  | 80   | AAA  | AA             | 555             | 55   | AAA              | 10               |
| BLOCK ERASE                                 | 6+     | AAA                  | AA   | 555  | 55   | AAA  | 80   | AAA  | AA             | 555             | 55   | BA               | 30               |
| PROGRAM/ERASE SUSPEND                       | 1      | X                    | B0   |      |      |      |      |      |                |                 |      |                  |                  |
| PROGRAM/ERASE RESUME                        | 1      | X                    | 30   |      |      |      |      |      |                |                 |      |                  |                  |

**Table 16: Commands – 8-Bit Mode (BYTE# = V<sub>IL</sub>) (Continued)**

| Command              | Length | Bus WRITE Operations |      |      |      |      |      |      |      |      |      |      |      |
|----------------------|--------|----------------------|------|------|------|------|------|------|------|------|------|------|------|
|                      |        | 1st                  |      | 2nd  |      | 3rd  |      | 4th  |      | 5th  |      | 6th  |      |
|                      |        | Addr                 | Data | Addr | Data | Addr | Data | Addr | Data | Addr | Data | Addr | Data |
| ENTER EXTENDED BLOCK | 3      | AAA                  | AA   | 555  | 55   | AAA  | 88   |      |      |      |      |      |      |
| EXIT EXTENDED BLOCK  | 4      | AAA                  | AA   | 555  | 55   | AAA  | 90   | X    | 00   |      |      |      |      |

- Notes:
1. X = " Don't Care;" PA = Program address; PD = Program data; BA = Any address in the block. All values in the table are in hexadecimal. The command interface only uses A-1, A0-A10, and DQ0-DQ7 to verify the commands; A11-A20, DQ8-DQ14, and DQ15 are "Don't Care." DQ15A-1 is A-1 when BYTE# is V<sub>IL</sub> or DQ15 when BYTE# is V<sub>IH</sub>.
  2. The following is content for address: Data cycles 7 through 10: PA5-PD5, PA6-PD6, PA7-PD7, PA8-PD8.
  3. The maximum number of cycles in the command sequence is 68. N + 1 is the number of words to be programmed during the WRITE TO BUFFER AND PROGRAM operation.
  4. Each buffer has the same A4-A22 addresses. A0-A3 and DQ15A-1 are used to select a word within the N + 1 word page.
  5. The 6th cycle has to be issued N time. WBL scans the word inside the page.
  6. BA must be identical to the address loaded during the WRITE TO BUFFER AND PROGRAM 3rd and 4th cycles.

## 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-1, A[7:0]) 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 17: 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) |
| 61h     | C2h | Security code area                              | 64-bit unique device number                                         |

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

**Table 18: 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     | AMD compatible |
| 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 19: 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 | 00B5h | V <sub>PPH</sub> (programming) supply minimum program/erase voltage<br>Bits[7:4] hex value in volts<br>Bits[3:0] BCD value in 100mV | 11.5V |
| 1Eh     | 3Ch | 00C5h | V <sub>PPH</sub> (programming) supply maximum program/erase voltage<br>Bits[7:4] hex value in volts<br>Bits[3:0] BCD value in 100mV | 12.5V |
| 1Fh     | 3Eh | 0004h | Typical timeout for single byte/word program = 2 <sup>n</sup> μs                                                                    | 16μs  |
| 20h     | 40h | 0004h | Typical timeout for maximum size buffer program = 2 <sup>n</sup> μs                                                                 | 16μs  |
| 21h     | 42h | 000Ah | Typical timeout per individual block erase = 2 <sup>n</sup> ms                                                                      | 1s    |
| 22h     | 44h | 0000h | Typical timeout for full chip erase = 2 <sup>n</sup> ms                                                                             | –     |
| 23h     | 46h | 0004h | Maximum timeout for byte/word program = 2 <sup>n</sup> times typical                                                                | 256μs |
| 24h     | 48h | 0004h | Maximum timeout for buffer program = 2 <sup>n</sup> times typical                                                                   | 200μs |
| 25h     | 4Ah | 0003h | Maximum timeout per individual block erase = 2 <sup>n</sup> times typical                                                           | 8s    |
| 26h     | 4Ch | 0000h | Maximum timeout for chip erase = 2 <sup>n</sup> times typical                                                                       | –     |

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

**Table 20: Device Geometry Definition**

| Address |     | Data  | Description                                                                                                                                                 | Value                   |
|---------|-----|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| x16     | x8  |       |                                                                                                                                                             |                         |
| 27h     | 4Eh | 0017h | Device size = 2 <sup>n</sup> in number of bytes                                                                                                             | 8MB                     |
| 28h     | 50h | 0002h | Flash device interface code description                                                                                                                     | x8, x16<br>asynchronous |
| 29h     | 52h | 0000h |                                                                                                                                                             |                         |
| 2Ah     | 54h | 0005h | Maximum number of bytes in multi-byte program or page = 2 <sup>n</sup>                                                                                      | 32                      |
| 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. M29W640GH and M29W640GL devices only | 1                       |
|         |     | 0002h | Number of erase block regions. It specifies the number of regions containing contiguous erase blocks of the same size. M29W640GT and M29W640GB devices only | 2                       |
| 2Dh     | 5Ah | 0007h | Region 1 information<br>Number of identical-size erase blocks = 007Fh + 1<br>M29W640GH and M29W640GL devices only                                           | 128                     |
| 2Eh     | 5Ch | 0000h |                                                                                                                                                             |                         |

**Table 20: Device Geometry Definition (Continued)**

| Address                  |                          | Data                             | Description                                                                                                                                                                             | Value |
|--------------------------|--------------------------|----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------|
| x16                      | x8                       |                                  |                                                                                                                                                                                         |       |
| 2Fh<br>30h               | 5Eh<br>60h               | 0000h<br>0000h                   | Region 1 information<br>Block size in region 1 = 0100h × 256 bytes<br>M29W640GH and M29W640GL devices only                                                                              | 64KB  |
| 2Dh<br>2Eh               | 5Ah<br>5Ch               | 0007h<br>0000h                   | Region 1 information<br>Number of identical-size erase blocks = 0007h + 1<br>M29W640GT and M29W640GB devices only                                                                       | 8     |
| 2Fh<br>30h               | 5Eh<br>60h               | 0020h<br>0000h                   | Region 1 information<br>Block size in region 1 = 0200h × 256 bytes<br>M29W640GT and M29W640GB devices only                                                                              | 8KB   |
| 31h<br>32h               | 62h<br>64h               | 007Eh<br>0000h                   | Region 2 information<br>Number of identical-size erase blocks = 007Eh + 1<br>M29W640GT and M29W640GB devices only                                                                       | 127   |
| 33h<br>34h               | 66h<br>68h               | 0000h<br>0001h                   | Region 2 information<br>Block size in region 2 = 0100h × 256 bytes<br>M29W640GT and M29W640GB devices only                                                                              | 64KB  |
| 35h<br>36h<br>37h<br>38h | 6Ah<br>6Ch<br>6Eh<br>70h | 0000h<br>0000h<br>0000h<br>0000h | Region 3 information<br>Number of identical-size erase blocks = 0000h + 1<br>Region 3 information<br>Block size in region 3 = 0000h × 256 bytes<br>M29W640GT and M29W640GB devices only | 0     |
| 39h<br>3Ah<br>3Bh<br>3Ch | 72h<br>74h<br>76h<br>78h | 0000h<br>0000h<br>0000h<br>0000h | Region 4 information<br>Number of identical-size erase blocks = 0000h + 1<br>Region 4 information<br>Block size in region 4 = 0000h × 256 bytes<br>M29W640GT and M29W640GB devices only | 0     |

Note: 1. For bottom boot devices, erase block region 1 is located from address 000000h to 007FFFh and erase block region 2 from address 008000h to 3FFFFFFh. For top boot devices, erase block region 1 is located from address 000000h to 3F7FFFh and erase block region 2 from address 3F8000h to 3FFFFFFh.

**Table 21: 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"   |

**Table 21: Primary Algorithm-Specific Extended Query Table (Continued)**

Note 1 applies to the entire table

| Address |     | Data  | Description                                                                                                                                                                                                                                          | Value                                                                        |
|---------|-----|-------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| x16     | x8  |       |                                                                                                                                                                                                                                                      |                                                                              |
| 45h     | 8Ah | 0000h | Address sensitive unlock (bits[1:0]):<br>00 = Required<br>01 = Not required<br>Silicon revision number (bits[7:2])                                                                                                                                   | 00                                                                           |
| 46h     | 8Ch | 0002h | Erase suspend:<br>00 = Not supported<br>01 = Read only<br>02 = Read and write                                                                                                                                                                        | 2                                                                            |
| 47h     | 8Eh | 0004h | Block protection:<br>00 = Not supported<br>x = Number of blocks per group                                                                                                                                                                            | 4                                                                            |
| 48h     | 90h | 0001h | Temporary block unprotect:<br>00 = Not supported<br>01 = Supported                                                                                                                                                                                   | 01                                                                           |
| 49h     | 92h | 0004h | Block protect/unprotect                                                                                                                                                                                                                              | 04                                                                           |
| 4Ah     | 94h | 0000h | Simultaneous operations:<br>Not supported                                                                                                                                                                                                            | –                                                                            |
| 4Bh     | 96h | 0000h | Burst mode:<br>00 = Not supported<br>01 = Supported                                                                                                                                                                                                  | 00                                                                           |
| 4Ch     | 98h | 0001h | Page mode:<br>00 = Not supported<br>01 = 4-word page; 02 = 8-word page                                                                                                                                                                               | 01                                                                           |
| 4Dh     | 9Ah | 00B5h | V <sub>PPH</sub> supply minimum program/erase voltage:<br>Bits[7:4] hex value in volts<br>Bits[3:0] BCD value in 100mV                                                                                                                               | 11.5V                                                                        |
| 4Eh     | 9Ch | 00C5h | V <sub>PPH</sub> supply maximum program/erase voltage:<br>Bits[7:4] hex value in volts<br>Bits[3:0] BCD value in 100mV                                                                                                                               | 12.5V                                                                        |
| 4Fh     | 9Eh | 00xxh | Top/bottom boot block flag:<br>xx = 02h: M29W640GB bottom boot device<br>xx = 03h M29W640GT top boot device<br>xx = 04h M29W640GL first block protected by V <sub>PP</sub> /WP#<br>xx = 05h: M29W640GH, last block protected by V <sub>PP</sub> /WP# | 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                                                                                                                                                                                             | 01                                                                           |

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

**Table 22: Security Code Area**

| Address |          | Data | Description                 |
|---------|----------|------|-----------------------------|
| x16     | x8       |      |                             |
| 61h     | C3h, C2h | XXXX | 64-bit unique device number |
| 62h     | C5h, C4h | XXXX |                             |
| 63h     | C7h, C6h | XXXX |                             |
| 64h     | C9h, C8h | XXXX |                             |

## 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 23: 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   |       |
| Input/output voltage   | $V_{IO}$   | -0.6 | $V_{CC} + 0.6$ | V    | 1, 2  |
| Supply voltage         | $V_{CC}$   | -0.6 | 4              | V    |       |
| Program voltage        | $V_{PP}$   | -0.6 | 13.5           | V    | 3     |
| Identification voltage | $V_{ID}$   | -0.6 | 13.5           | V    |       |

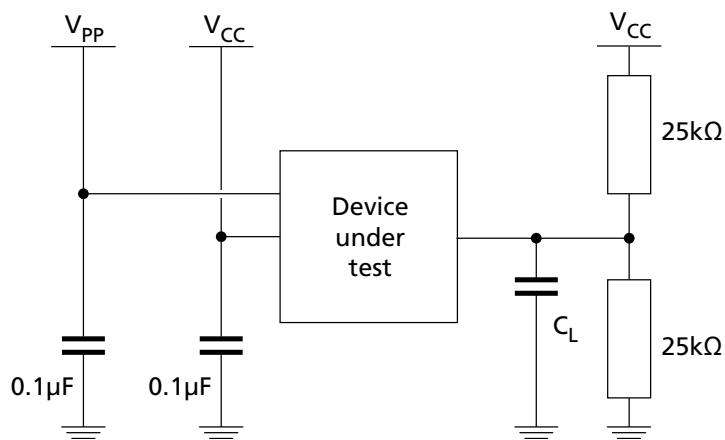
- Notes:
1. During signal transitions, minimum voltage may undershoot to -2V for periods less than 20ns.
  2. During signal transitions, maximum voltage may overshoot to  $V_{CC} + 2V$  for periods less than 20ns.
  3.  $V_{PP}$  must not remain at 12V for more than a total of 80 hours.

**Table 24: Operating Conditions**

| Parameter                                  | Symbol   | M29W640GT/B and<br>M29W640GH/L |        | Unit | Notes |
|--------------------------------------------|----------|--------------------------------|--------|------|-------|
|                                            |          | Min                            | Max    |      |       |
| Supply voltage                             | $V_{CC}$ | 2.7                            | 3.6    | V    |       |
| Ambient operating temperature              | $T_A$    | -40                            | 85/125 | °C   | 1     |
| Load capacitance                           | $C_L$    | 30                             |        | pF   |       |
| Input rise and fall times                  | —        | —                              | 10     | ns   |       |
| Input pulse voltages                       | —        | 0 to $V_{CC}$                  |        | V    |       |
| Input and output timing reference voltages | —        | $V_{CC}/2$                     |        | V    |       |

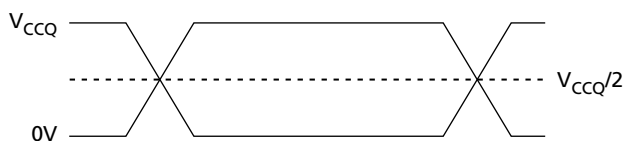
Note: 1. 85°C = industrial part; 125°C = autograde part.

**Figure 12: AC Measurement Load Circuit**



Note: 1.  $C_L$  includes jig capacitance.

**Figure 13: AC Measurement I/O Waveform**



**Table 25: Input/Output Capacitance**

| Parameter          | Symbol    | Test Condition | Min | Max | Unit |
|--------------------|-----------|----------------|-----|-----|------|
| Input capacitance  | $C_{IN}$  | $V_{IN} = 0V$  | –   | 6   | pF   |
| Output capacitance | $C_{OUT}$ | $V_{OUT} = 0V$ | –   | 12  | pF   |

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

## DC Characteristics

**Table 26: DC Current Characteristics**

| Parameter                                      | Symbol    | Conditions                                                                                           | Typ | Max     | Unit    | Notes |
|------------------------------------------------|-----------|------------------------------------------------------------------------------------------------------|-----|---------|---------|-------|
| Input leakage current                          | $I_{LI}$  | $0V \leq V_{IN} \leq V_{CC}$                                                                         | –   | $\pm 1$ | $\mu A$ | 1     |
| Output leakage current                         | $I_{LO}$  | $0V \leq V_{OUT} \leq V_{CC}$                                                                        | –   | $\pm 1$ | $\mu A$ |       |
| Read current                                   | $I_{CC1}$ | $CE\# = V_{IL}$ , $OE\# = V_{IH}$ ,<br>$f = 6 \text{ MHz}$                                           |     | 10      | mA      |       |
| Standby current                                | $I_{CC2}$ | $CE\# = V_{CC} \pm 0.2V$<br>$RST\# = V_{CC} \pm 0.2V$                                                |     | 100     | $\mu A$ | 2     |
| Program/Erase current                          | $I_{CC3}$ | Program/Erase controller ac-<br>tive: $V_{pp}/WP\# = V_{IL}$ or $V_{IH}$ ;<br>$V_{pp}/WP\# = V_{pp}$ | –   | 20      | mA      | 3     |
| Current for $V_{pp}/WP\#$ program acceleration | $I_{pp}$  | $V_{CC} = 2.7V \pm 10\%$                                                                             | –   | 15      | mA      |       |

- Notes:
1. The maximum input leakage current is  $\pm 5\mu A$  on  $V_{pp}/WP\#$ .
  2. When the bus is inactive for 300ns or more, the memory enters automatic standby.
  3. Sampled only; not 100% tested.

**Table 27: DC Voltage Characteristics**

| Parameter                                      | Symbol    | Conditions               | Min            | Max            | Unit | Notes |
|------------------------------------------------|-----------|--------------------------|----------------|----------------|------|-------|
| Input LOW voltage                              | $V_{IL}$  | –                        | –0.5           | 0.8            | V    |       |
| Input HIGH voltage                             | $V_{IH}$  | –                        | $0.7 V_{CC}$   | $V_{CC} + 0.3$ | V    |       |
| Voltage for $V_{pp}/WP\#$ program acceleration | $V_{pp}$  | $V_{CC} = 2.7V \pm 10\%$ | 11.5           | 12.5           | V    |       |
| Output LOW voltage                             | $V_{OL}$  | $I_{OL} = 1.8mA$         | –              | 0.45           | V    |       |
| Output HIGH voltage                            | $V_{OH}$  | $I_{OH} = -100\mu A$     | $V_{CC} - 0.4$ | –              | V    |       |
| Identification voltage                         | $V_{ID}$  | –                        | 11.5           | 12.5           | V    |       |
| Program/erase lockout supply voltage           | $V_{LKO}$ | –                        | 1.8            | 2.3            | V    | 1     |

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

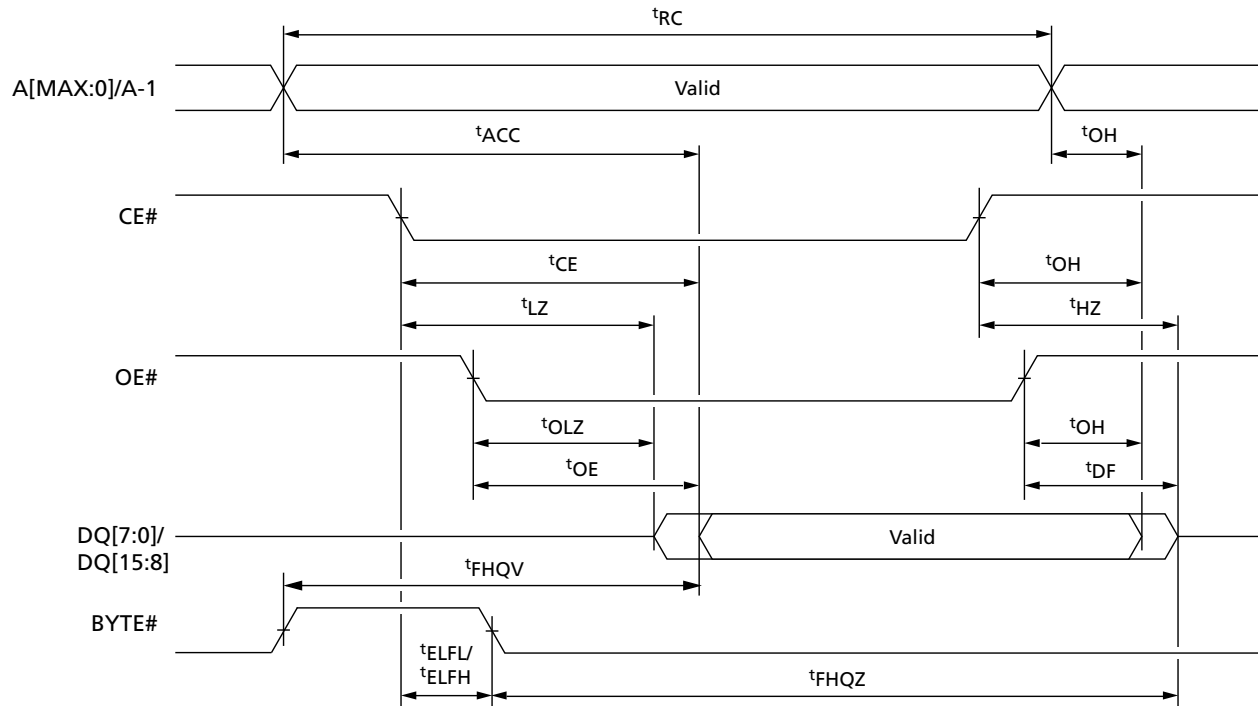
## Read AC Characteristics

**Table 28: Read AC Characteristics**

| Parameter                                            | Symbol     |                                            | Condition                            | Min/Max | 60ns | 70ns | 90ns | Unit | Notes |
|------------------------------------------------------|------------|--------------------------------------------|--------------------------------------|---------|------|------|------|------|-------|
|                                                      | Legacy     | JEDEC                                      |                                      |         |      |      |      |      |       |
| Address valid to next address valid                  | $t_{RC}$   | $t_{AVAV}$                                 | $CE\# = V_{IL}$ ,<br>$OE\# = V_{IL}$ | Min     | 60   | 70   | 90   | ns   |       |
| Address valid to output valid                        | $t_{ACC}$  | $t_{AVQV}$                                 | $CE\# = V_{IL}$ ,<br>$OE\# = V_{IL}$ | Max     | 60   | 70   | 90   | ns   |       |
| Address valid to output valid (page)                 | $t_{PAGE}$ | $t_{AVQV1}$                                | $CE\# = V_{IL}$ ,<br>$OE\# = V_{IL}$ | Max     | 25   | 30   | 30   | ns   |       |
| CE# LOW to output transition                         | $t_{LZ}$   | $t_{ELQX}$                                 | $OE\# = V_{IL}$                      | Min     | 0    | 0    | 0    | ns   | 1     |
| CE# LOW to output valid                              | $t_{CE}$   | $t_{ELQV}$                                 | $OE\# = V_{IL}$                      | Max     | 60   | 70   | 90   | ns   |       |
| OE# LOW to output transition                         | $t_{OLZ}$  | $t_{GLQX}$                                 | $CE\# = V_{IL}$                      | Min     | 0    | 0    | 0    | ns   | 1     |
| OE# LOW to output valid                              | $t_{OE}$   | $t_{GLQV}$                                 | $CE\# = V_{IL}$                      | Max     | 25   | 30   | 30   | ns   |       |
| CE# HIGH to output High-Z                            | $t_{HZ}$   | $t_{EHQZ}$                                 | $OE\# = V_{IL}$                      | Max     | 25   | 30   | 30   | ns   | 1     |
| OE# HIGH to output High-Z                            | $t_{DF}$   | $t_{GHQZ}$                                 | $CE\# = V_{IL}$                      | Max     | 25   | 30   | 30   | ns   | 1     |
| CE#, OE#, or address transition to output transition | $t_{OH}$   | $t_{EHQX}$ ,<br>$t_{GHQX}$ ,<br>$t_{AXQX}$ | –                                    | Min     | 0    | 0    | 0    | ns   |       |
| CE# to BYTE# LOW                                     | $t_{ELFL}$ | $t_{ELBL}$                                 | –                                    | Max     | 5    | 5    | 5    | ns   |       |
| CE# to BYTE# HIGH                                    | $t_{ELFH}$ | $t_{ELBH}$                                 | –                                    | Max     | 5    | 5    | 5    | ns   |       |
| BYTE# LOW to output High-Z                           | $t_{FLQZ}$ | $t_{BLQZ}$                                 | –                                    | Max     | 25   | 25   | 25   | ns   |       |
| BYTE# HIGH to output valid                           | $t_{FHQV}$ | $t_{BHQV}$                                 | –                                    | Max     | 25   | 30   | 30   | ns   |       |

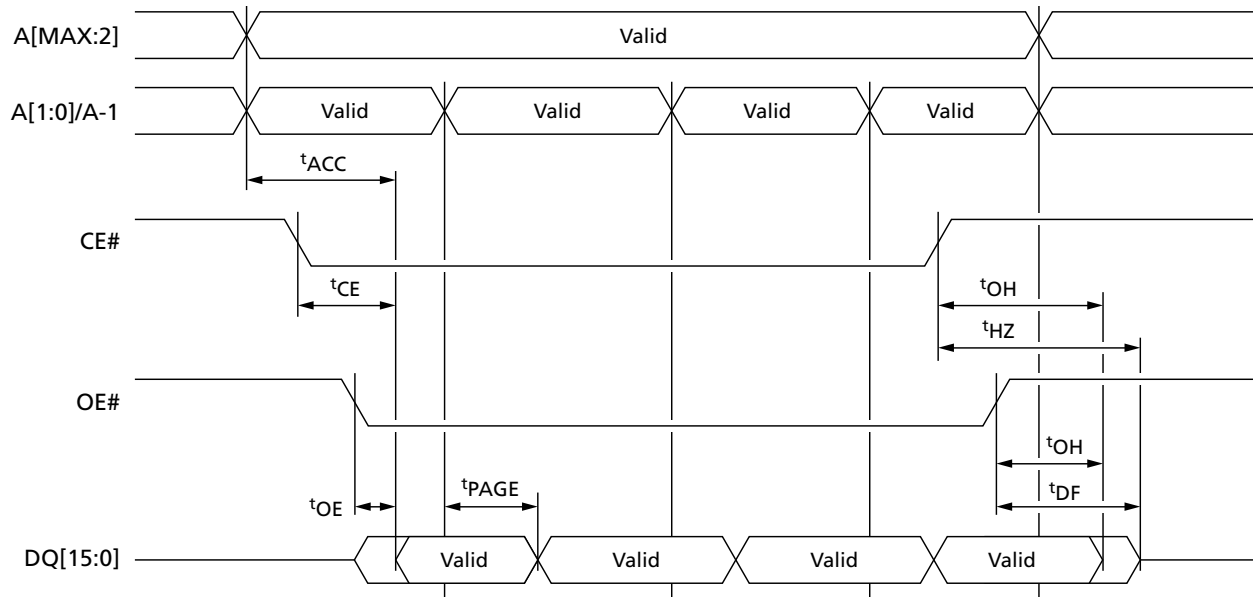
Note: 1. Sampled only; not 100% tested.

**Figure 14: Random AC Timing**



- Notes:
1. Data are output on DQ0-DQ15. DQ8-DQ14 are High-Z in 8-bit mode.
  2. Addresses differ in x8 mode.
  3. BYTE# =  $V_{IL}$  in x8 mode.

**Figure 15: Page Read AC Timing**



- Notes:
1. Data are output on DQ0-DQ15. DQ8-DQ14 are High-Z in 8-bit mode.
  2. Addresses differ in x8 mode.

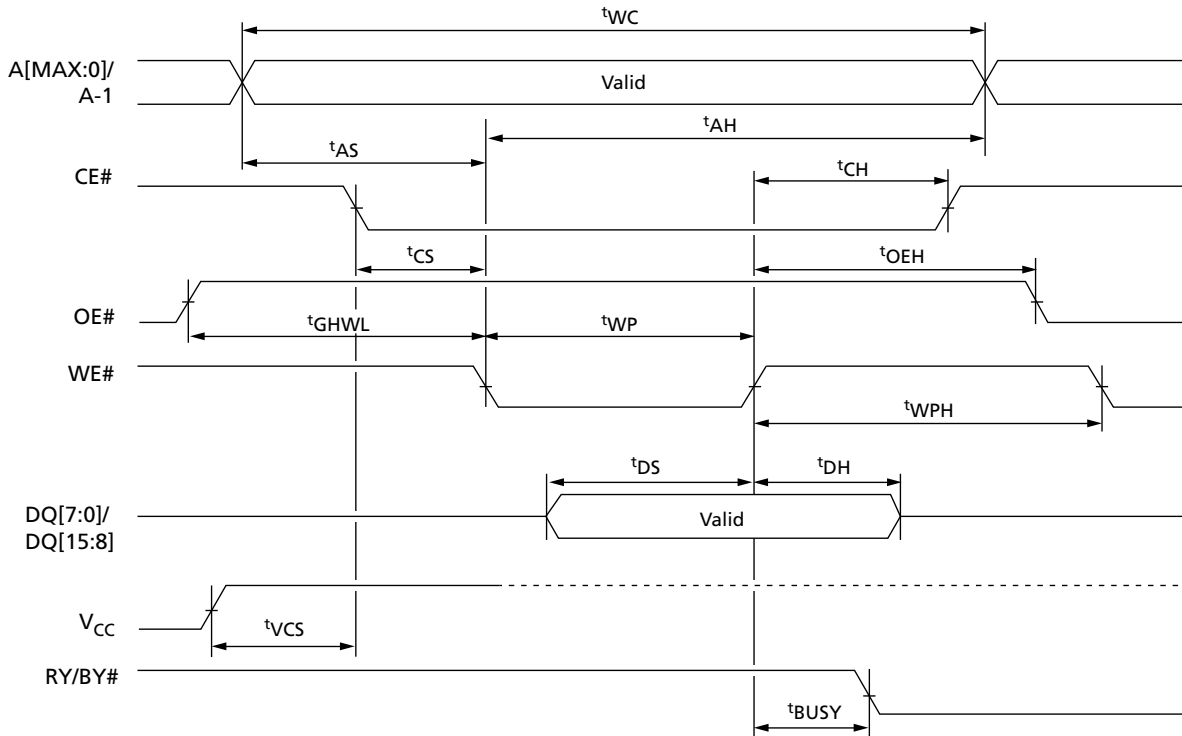
## Write AC Characteristics

**Table 29: WE#-Controlled Write AC Characteristics**

| Parameter                           | Symbol     |             | Min/Max | 60ns | 70ns | 90ns | Unit | Notes |
|-------------------------------------|------------|-------------|---------|------|------|------|------|-------|
|                                     | Legacy     | JEDEC       |         |      |      |      |      |       |
| Address valid to next address valid | $t_{WC}$   | $t_{AVAV}$  | Min     | 60   | 70   | 90   | ns   |       |
| CE# LOW to WE# LOW                  | $t_{CS}$   | $t_{ELWL}$  | Min     | 0    | 0    | 0    | ns   |       |
| WE# LOW to WE# HIGH                 | $t_{WP}$   | $t_{WLWH}$  | Min     | 35   | 35   | 35   | ns   |       |
| Input valid to WE# HIGH             | $t_{DS}$   | $t_{DVWH}$  | Min     | 30   | 30   | 30   | ns   |       |
| WE# HIGH to input transition        | $t_{DH}$   | $t_{WHDX}$  | Min     | 0    | 0    | 0    | ns   |       |
| WE# HIGH to CE# HIGH                | $t_{CH}$   | $t_{WHEH}$  | Min     | 0    | 0    | 0    | ns   |       |
| WE# HIGH to WE# LOW                 | $t_{WPH}$  | $t_{WHWL}$  | Min     | 25   | 25   | 25   | ns   |       |
| Address valid to WE# LOW            | $t_{AS}$   | $t_{AVWL}$  | Min     | 0    | 0    | 0    | ns   |       |
| WE# LOW to address transition       | $t_{AH}$   | $t_{WLAX}$  | Min     | 45   | 45   | 45   | ns   |       |
| OE# HIGH to WE# LOW                 | –          | $t_{GHWL}$  | Min     | 0    | 0    | 0    | ns   |       |
| WE# HIGH to OE# LOW                 | $t_{OEHL}$ | $t_{WHGL1}$ | Min     | 0    | 0    | 0    | ns   |       |
| Program/erase valid to RY# LOW      | $t_{BUSY}$ | $t_{WHRL1}$ | Max     | 0    | 0    | 0    | ns   | 1     |
| V <sub>CC</sub> HIGH to CE# LOW     | $t_{VCS}$  | $t_{VCHEL}$ | Min     | 50   | 50   | 50   | μs   |       |

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

**Figure 16: WE#-Controlled AC Timing**



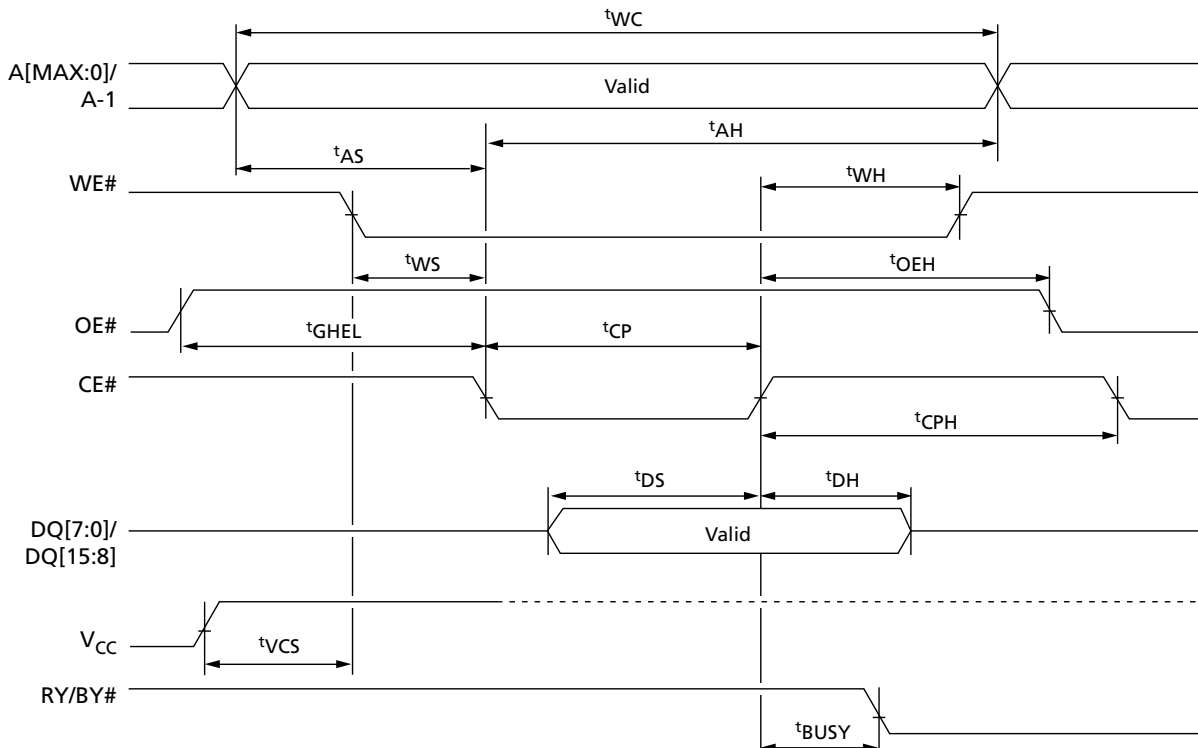
Note: 1. Addresses differ in x8 mode.

**Table 30: CE#-Controlled Write AC Characteristics**

| Parameter                           | Symbol     |             | Min/Max | 60ns | 70ns | 90ns | Unit | Notes |
|-------------------------------------|------------|-------------|---------|------|------|------|------|-------|
|                                     | Legacy     | JEDEC       |         |      |      |      |      |       |
| Address valid to next address valid | $t_{WC}$   | $t_{AVAV}$  | Min     | 60   | 70   | 90   | ns   |       |
| WE# LOW to CE# LOW                  | $t_{WS}$   | $t_{WLEL}$  | Min     | 0    | 0    | 0    | ns   |       |
| CE# LOW to CE# HIGH                 | $t_{CP}$   | $t_{ELEH}$  | Min     | 35   | 35   | 35   | ns   |       |
| Input valid to CE# HIGH             | $t_{DS}$   | $t_{DVEH}$  | Min     | 30   | 30   | 30   | ns   |       |
| CE# HIGH to input transition        | $t_{DH}$   | $t_{EHDX}$  | Min     | 0    | 0    | 0    | ns   |       |
| CE# HIGH to WE# HIGH                | $t_{WH}$   | $t_{EHWLH}$ | Min     | 0    | 0    | 0    | ns   |       |
| CE# HIGH to CE# LOW                 | $t_{CPH}$  | $t_{EHEL1}$ | Min     | 25   | 25   | 25   | ns   |       |
| Address valid to CE# LOW            | $t_{AS}$   | $t_{AVEL}$  | Min     | 0    | 0    | 0    | ns   |       |
| CE# LOW to address transition       | $t_{AH}$   | $t_{ELAX}$  | Min     | 45   | 45   | 45   | ns   |       |
| OE# HIGH to CE# LOW                 | –          | $t_{GHEL}$  | Min     | 0    | 0    | 0    | ns   |       |
| CE# HIGH to OE# LOW                 | $t_{OEHL}$ | $t_{EHGL1}$ | Min     | 0    | 0    | 0    | ns   |       |
| Program/Erase valid to RY/BY# LOW   | $t_{BUSY}$ | $t_{EHRL}$  | Max     | 0    | 0    | 0    | ns   | 1     |
| V <sub>CC</sub> HIGH to WE# LOW     | $t_{VCS}$  | $t_{VCHWL}$ | Min     | 50   | 50   | 50   | ns   |       |

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

**Figure 17: CE#-Controlled AC Timing**



Note: 1. Addresses differ in x8 mode.

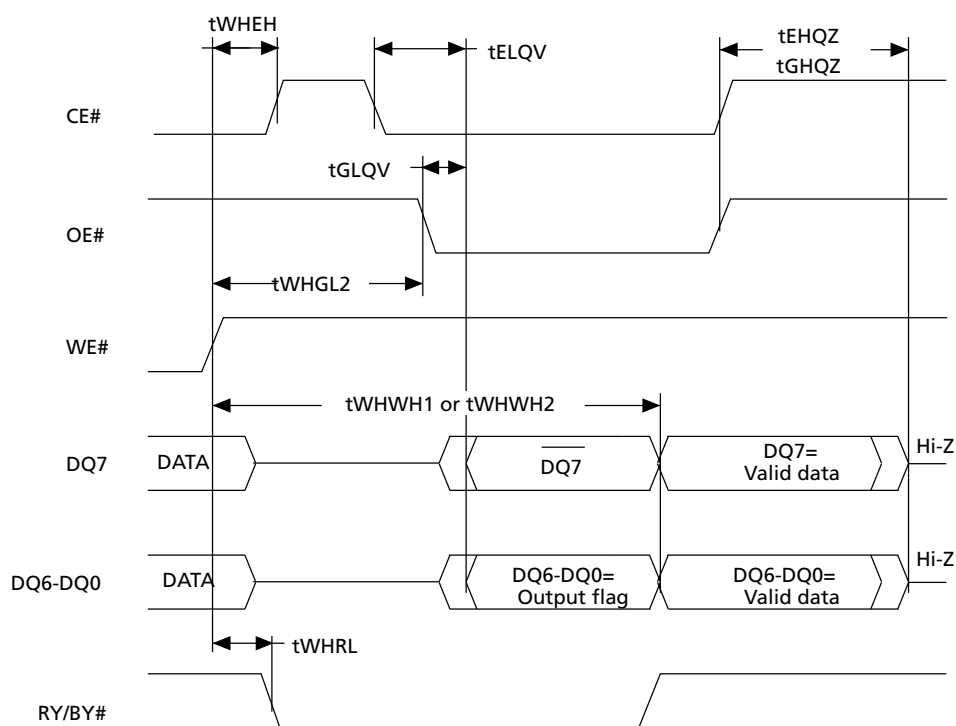
## Toggle and Alternative Toggle AC Characteristics

**Table 31: Toggle and Alternative Toggle AC Characteristics**

| Parameter                                               | Symbol     |             | Min/Max | 60ns | 70ns | 90ns | Unit |
|---------------------------------------------------------|------------|-------------|---------|------|------|------|------|
|                                                         | Legacy     | JEDEC       |         |      |      |      |      |
| Address setup time to OE# LOW during toggle bit polling | $t_{ASO}$  | $t_{AXGL}$  | Min     | 10   | 10   | 10   | ns   |
| Address hold time from OE# during toggle bit polling    | $t_{AHT}$  | $t_{GHAX}$  | Min     | 10   | 10   | 10   | ns   |
|                                                         |            | $t_{EHAX}$  | Min     | 10   | 10   | 10   | ns   |
| CE# HIGH during toggle bit polling                      | $t_{CEPH}$ | $t_{EHEL2}$ | Min     | 10   | 10   | 10   | ns   |
| Output hold time during data and toggle bit polling     | $t_{OEH}$  | $t_{WHGL2}$ | Min     | 20   | 20   | 20   | ns   |
|                                                         |            | $t_{GHGL2}$ | Min     | 20   | 20   | 20   | ns   |

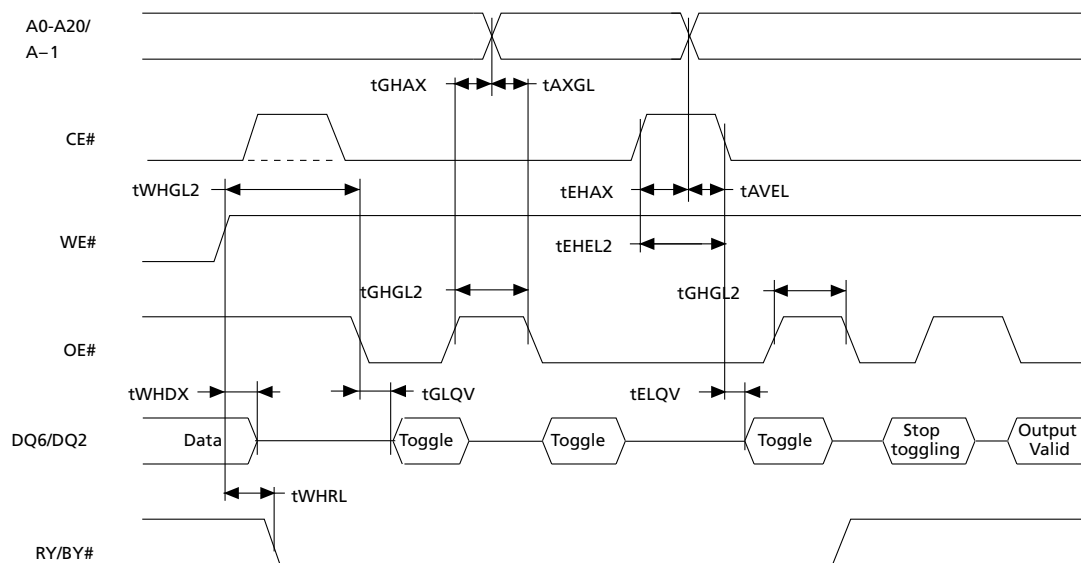
Note: 1. Data for  $t_{ELQV}$  and  $t_{GLQV}$  are in the Read AC Characteristics table.

**Figure 18: Toggle/Alternative Toggle, CE# Controlled**



- Notes:
1. DQ7 returns valid data bit when the ongoing PROGRAM or ERASE command is completed.
  2. See the AC Characteristics for specifications.

**Figure 19: Toggle/Alternative Toggle, OE# Controlled**



- Notes:
1. DQ6 stops toggling when the ongoing PROGRAM or ERASE command is completed. DQ2 stops toggling when the ongoing CHIP ERASE or BLOCK ERASE command is completed.
  2. Addresses differ in x8 mode.
  3. See the AC Characteristics for specifications.

## Program/Erase Characteristics

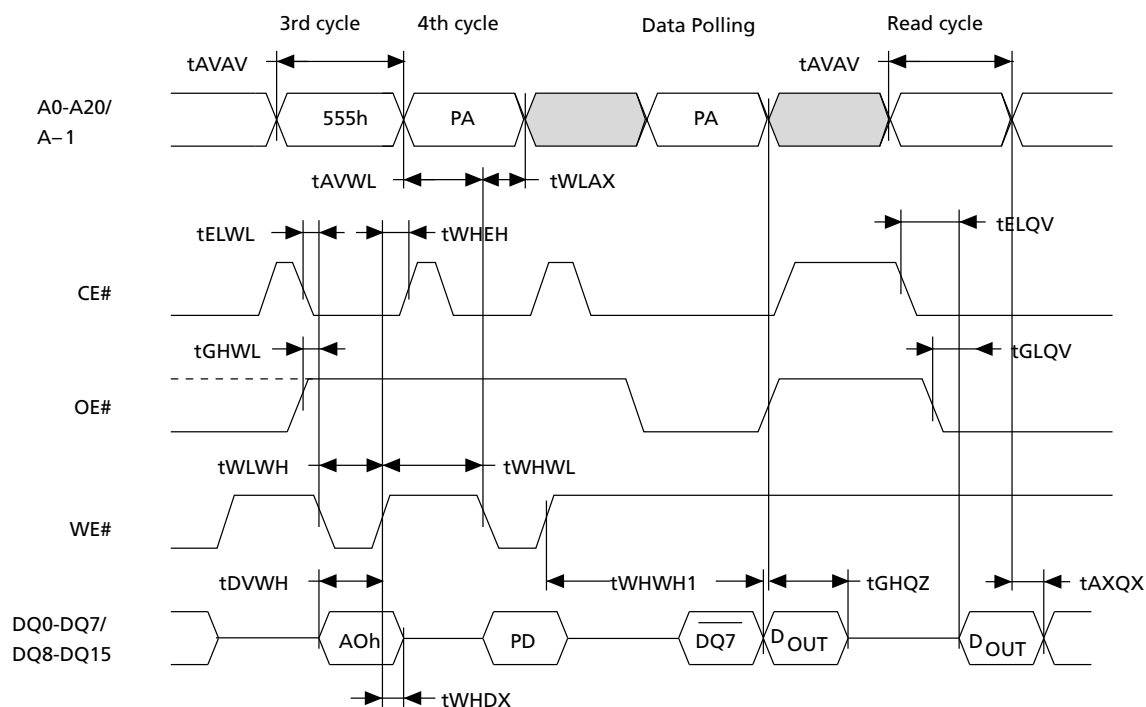
**Table 32: Program/Erase Times and Endurance Cycles**

Notes 1 and 2 apply to the entire table

| Parameter                                                                            | Symbol      | Min     | Typ | Max | Unit    | Notes |
|--------------------------------------------------------------------------------------|-------------|---------|-----|-----|---------|-------|
| Chip erase                                                                           |             | –       | 80  | 400 | s       | 3     |
| Block erase (64KB)                                                                   | $t_{WHWH2}$ | –       | 0.5 | –   | s       | 4, 5  |
| Erase suspend latency time                                                           |             | –       | –   | 50  | $\mu$ s | 6     |
| Program (byte or word)                                                               |             | –       | 10  | 200 | $\mu$ s | 3     |
| Program (double byte)                                                                |             |         | 10  | 200 | $\mu$ s | 3     |
| Program (double word/quadruple byte)                                                 |             |         | 10  | 200 | $\mu$ s | 3     |
| Program (quadruple word/octuple byte)                                                |             |         | 10  | 200 | $\mu$ s | 3     |
| Program (single byte and word)                                                       | $t_{WHWH1}$ | –       | 10  |     | $\mu$ s | 3, 7  |
| Program (32-byte 16-word using Write to Buffer and Program)                          |             |         | 180 |     | $\mu$ s |       |
| Program (32-byte 16-word using Write to Buffer and Program, $V_{PP}/V_{P\#} = 12V$ ) |             |         | 45  |     | $\mu$ s |       |
| Chip program (byte by byte)                                                          |             | –       | 80  | 400 | s       | 3     |
| Chip program (word by word)                                                          |             |         | 40  | 200 | s       | 3     |
| Chip program (double word/quadruple byte)                                            |             |         | 20  | 100 | s       | 3     |
| Chip program (quadruple word/octuple byte)                                           |             | –       | 10  | 50  | s       | 3     |
| Program suspend latency time                                                         |             | –       | –   | 4   | $\mu$ s |       |
| PROGRAM/ERASE cycles (per block)                                                     |             | 100,000 | –   | –   | cycles  |       |
| Data retention                                                                       |             | 20      | –   | –   | years   |       |

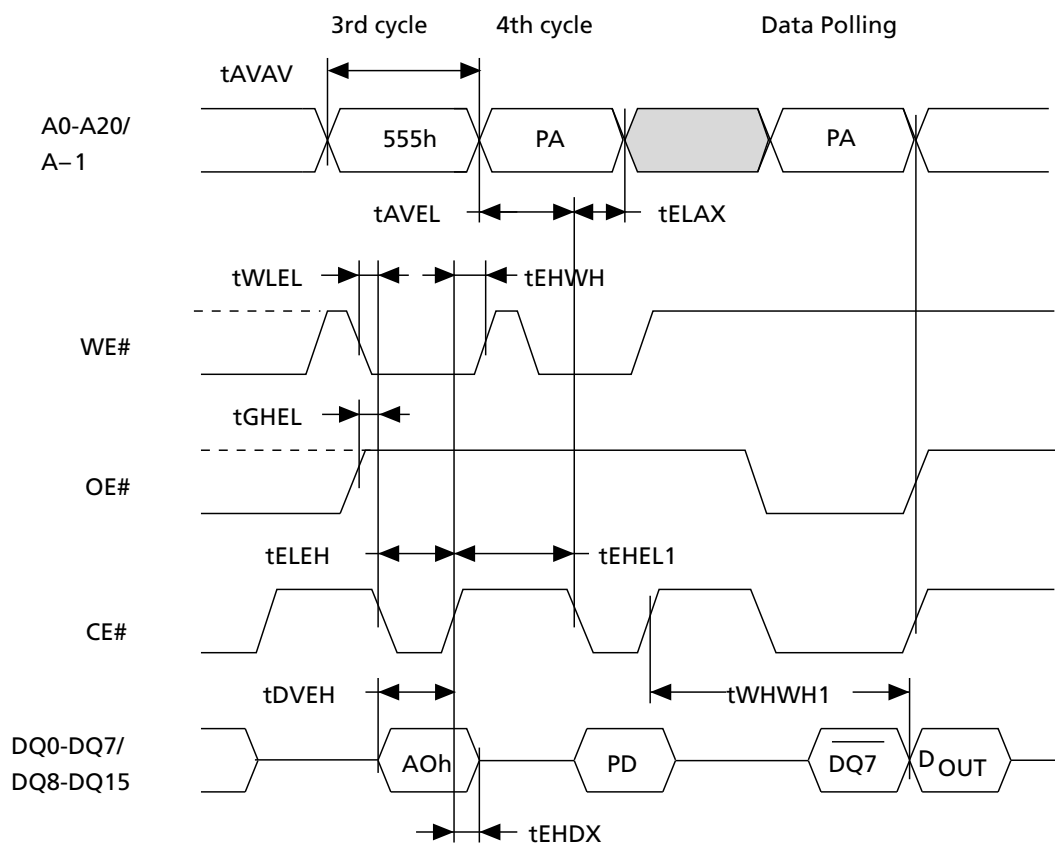
- Notes:
1. Typical values measured at room temperature and nominal voltages and for not cycled devices.
  2. Sampled, but not 100% tested.
  3. Maximum value measured at worst case conditions for both temperature and  $V_{CC}$  after 100,000 PROGRAM/ERASE cycles.
  4. Time does not include pre-programming time.
  5. Block erase polling cycle time.
  6. Maximum value measured at worst case conditions for both temperature and  $V_{CC}$ .
  7. Programming polling cycle time.

**Figure 20: WE# Controlled Program Waveform**



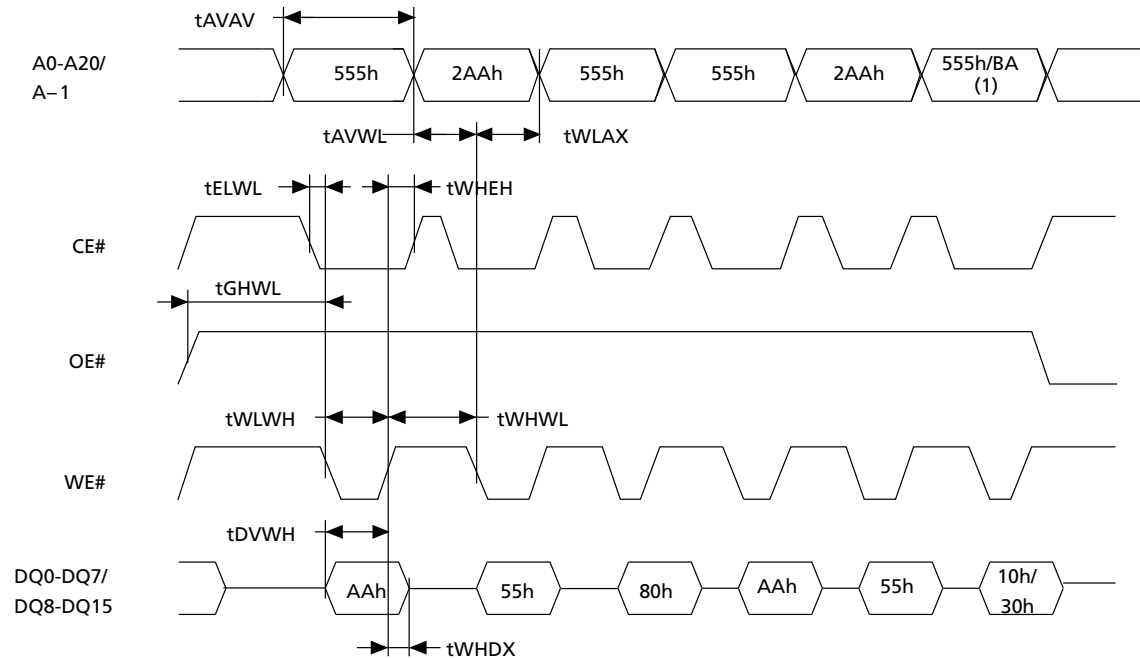
- Notes:
1. Only the third and fourth cycles of the Program command are represented. The Program command is followed by the check of status register data polling bit and by a read operation that outputs the data, DOUT, programmed by the previous Program command.
  2. PA is address of the memory location to be programmed. PD is the data to be programmed.
  3. DQ7# is the complement of the data bit being programmed to DQ7.
  4. Addresses differ in x8 mode.

**Figure 21: CE# Controlled Program Waveform**



- Notes:
1. Only the third and fourth cycles of the Program command are represented. The Program command is followed by the check of status register data polling bit.
  2. PA is 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.
  4. Addresses differ in x8 mode.

**Figure 22: Chip/Block Erase Waveform**



- Notes:
1. For a Chip Erase command, addresses and data are 555h and 10h, respectively, while they are BA and 30h for a Block Erase command.
  2. BA is the block address.
  3. Addresses differ in x8 mode.

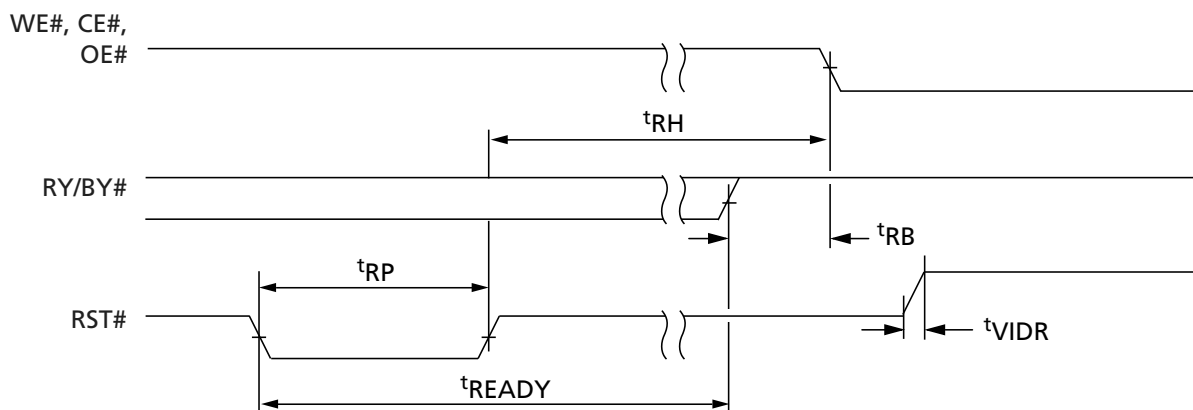
## Reset Characteristics

**Table 33: Reset/Block Temporary Unprotect AC Characteristics**

| Condition/Parameter                      | Symbol      |                                        | Min/Max | 60, 70, 90ns | Unit    | Notes |
|------------------------------------------|-------------|----------------------------------------|---------|--------------|---------|-------|
|                                          | Legacy      | JEDEC                                  |         |              |         |       |
| RST# HIGH to WE# LOW, CE# LOW, OE# LOW   | $t_{RH}$    | $t_{PHWL}$<br>$t_{PHEL}$<br>$t_{PHGL}$ | Min     | 50           | ns      | 1     |
| RY/BY# HIGH to WE# LOW, CE# LOW, OE# LOW | $t_{RB}$    | $t_{RHWL}$<br>$t_{RHEL}$<br>$t_{RHGL}$ | Min     | 0            | ns      | 1     |
| RST# pulse width                         | $t_{RP}$    | $t_{PLPX}$                             | Min     | 500          | ns      |       |
| RST# LOW to read mode                    | $t_{READY}$ | $t_{PLYH}$                             | Max     | 50           | $\mu s$ | 1     |
| RST# rise time to $V_{ID}$               | $t_{VIDR}$  | $t_{PHPHH}$                            | Min     | 500          | ns      | 1, 2  |
| $V_{pp}$ rise and fall time              | –           | $t_{VHVPP}$                            | Min     | 500          | ns      | 1     |

Notes: 1. Sampled only; not 100% tested.  
2. For fast program operations using  $V_{pp}/WP\#$  at 12V.

**Figure 23: Reset/Block Temporary Unprotect AC Waveforms**

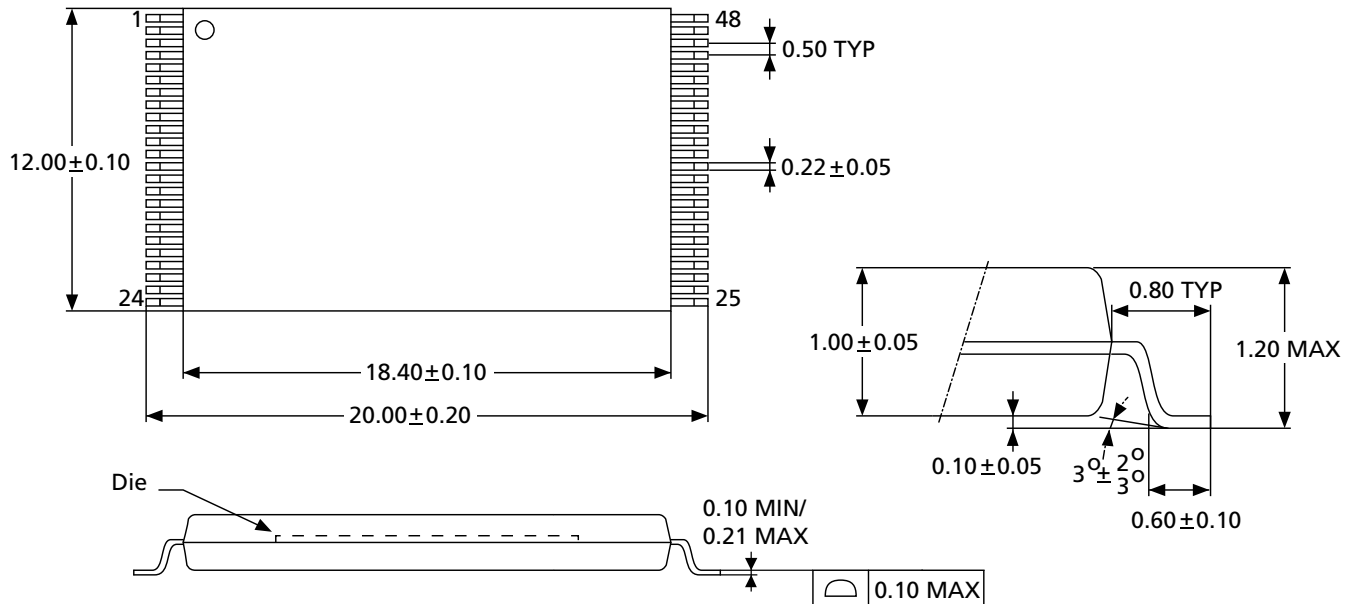


**Figure 24: Accelerated Programming Timing Waveform**



## Package Dimensions

**Figure 25: 48-Pin TSOP – 12mm x 20mm**

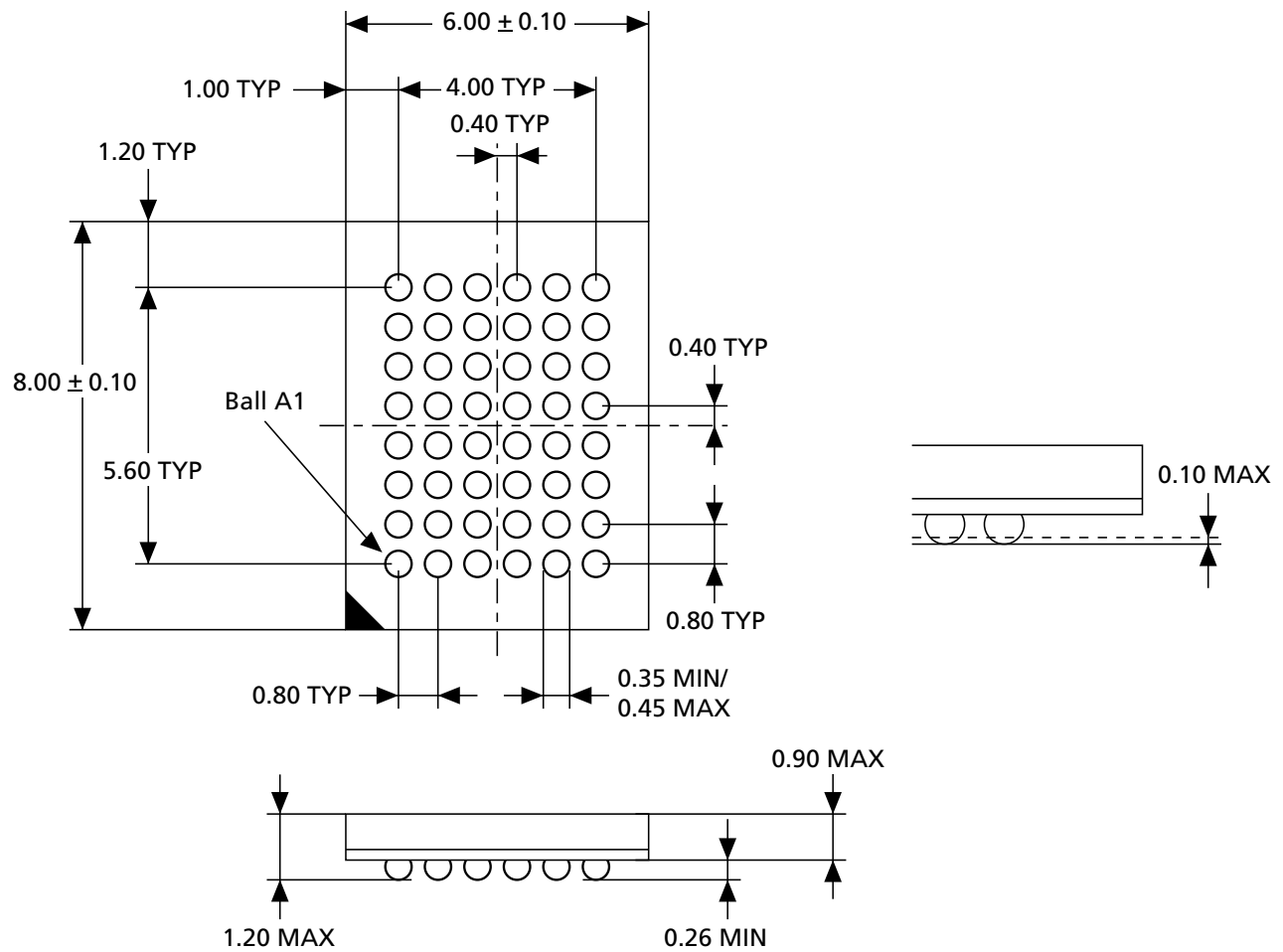


Note: 1. All dimensions are in millimeters.

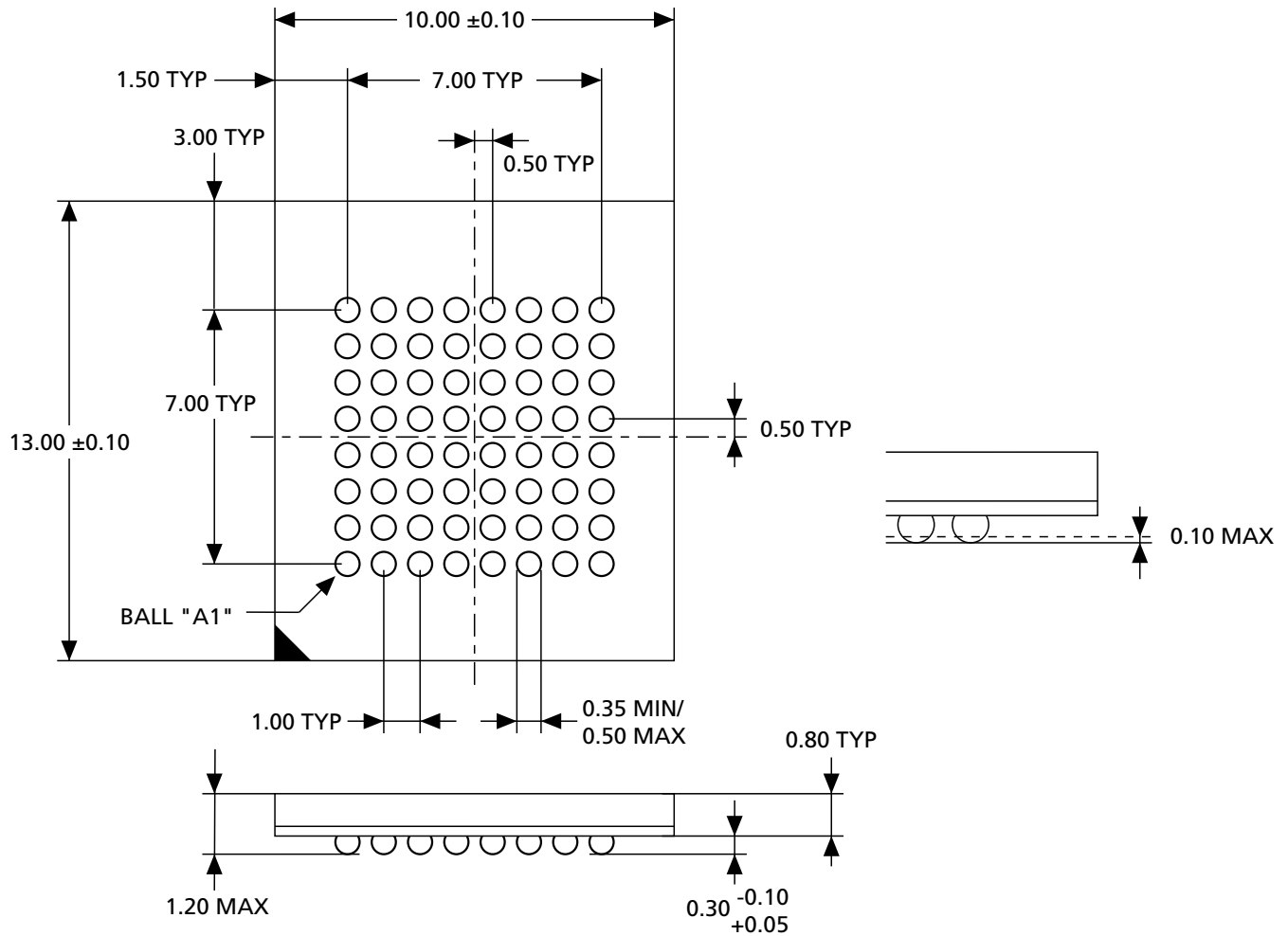
Technical drawing of a rectangular frame assembly. The main view shows a rectangular frame with dimensions: overall width  $20.00 \pm 0.20$ , overall height  $14.00 \pm 0.10$ , and an inner width of  $18.40 \pm 0.10$ . The left vertical member is labeled "Pin #1" and the right vertical member is labeled "N". A detail view on the right shows a cross-section of the vertical member with dimensions:  $0.50$  TYP (typical) for the top flange and  $0.22 \pm 0.05$  for the web. A detail view at the bottom left shows a cross-section of the horizontal member with dimensions:  $0.10$  MIN /  $0.21$  MAX for the flange and  $1.20$  MAX for the web. A detail view at the bottom right shows a cross-section of the corner joint with dimensions:  $1.00 \pm 0.05$  for the vertical member,  $0.10 \pm 0.05$  for the horizontal member, and a fillet radius of  $3^\circ/5^\circ$ . A dimension of  $0.10$  is shown for the corner joint. A note "See Detail A" points to the corner joint. A dimension of  $0.10$  is shown for the corner joint.

- Notes:
1. All dimensions are in millimeters.
  2. For the lead width value of  $0.22 \pm 0.05$ , there is also a legacy value of  $0.15 \pm 0.05$ .

**Figure 27: 48-Ball TFBGA – 6mm x 8mm**

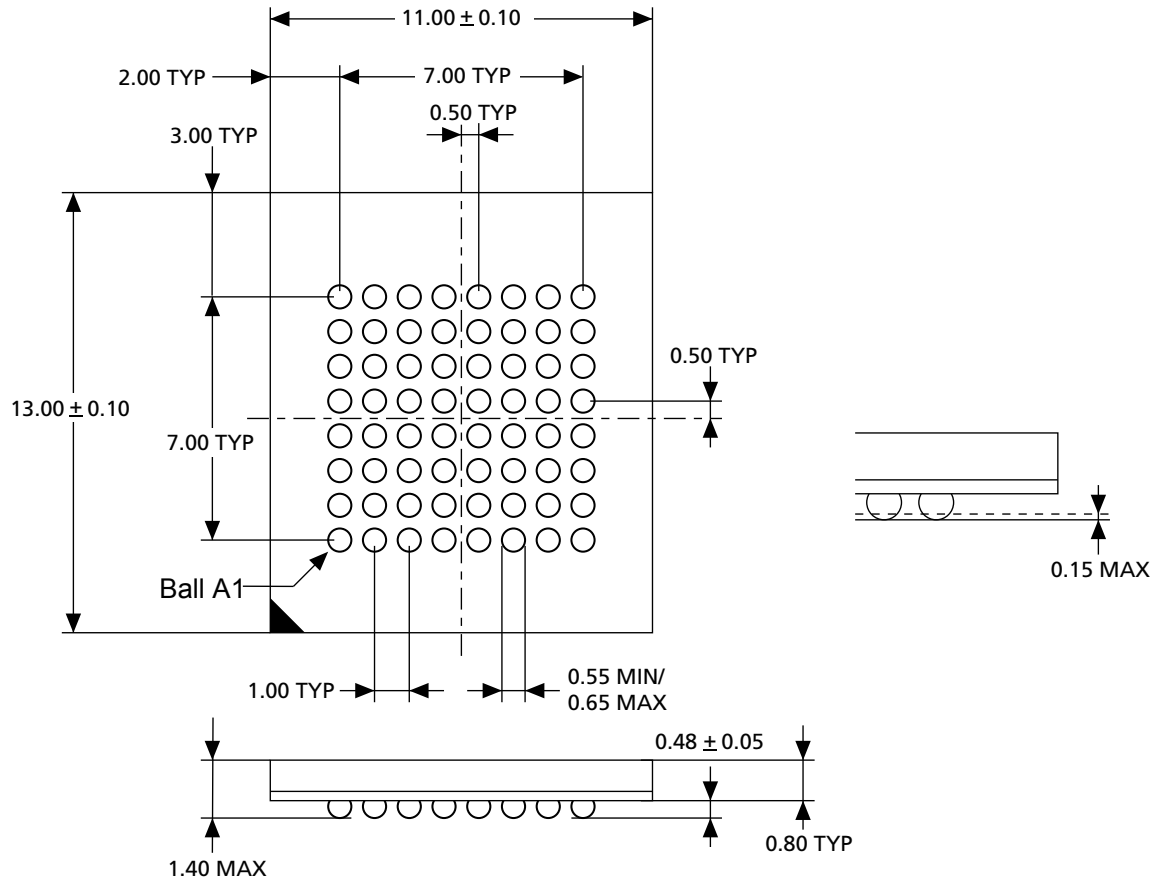


**Figure 28: 64-Ball TBGA – 10mm x 13mm**



Note: 1. All dimensions are in millimeters.

**Figure 29: 64-Ball FBGA – 11mm x 13mm**



Note: 1. All dimensions are in millimeters.

## **Revision History**

### **Rev. C – 07/13**

- Updated Command Interface and Command tables

### **Rev. B – 06/13**

- Removed Part Numbers by Array Matrix table in Features

### **Rev. A – 11/12**

- Initial Micron brand release

8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900  
[www.micron.com/productsupport](http://www.micron.com/productsupport) Customer Comment Line: 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.