# Am45DL3208G

Data Sheet

September 2003

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

## Continuity of Specifications

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

## Continuity of Ordering Part Numbers

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

## For More Information

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

Publication Number **26460** Revision **B** Amendment **+1** Issue Date **March 12, 2004**

**THIS PAGE LEFT INTENTIONALLY BLANK.**

PRELIMINARY

# Am45DL3208G

## Stacked Multi-Chip Package (MCP) Flash Memory and SRAM

### 32 Megabit (4 M x 8-Bit/2 M x 16-Bit) CMOS 3.0 Volt-only, Simultaneous Operation Flash Memory and 8 Mbit (1 M x 8-Bit/512 K x 16-Bit) Pseudo Static RAM

## DISTINCTIVE CHARACTERISTICS

### MCP Features

- **Power supply voltage of 2.7 to 3.3 volt**
- **High performance**
  - — Access time as fast as 70 ns
- **Package**
  - — 73-Ball FBGA
- **Operating Temperature**
  - — –40°C to +85°C

### Flash Memory Features

#### ARCHITECTURAL ADVANTAGES

- **Simultaneous Read/Write operations**
  - — Data can be continuously read from one bank while executing erase/program functions in another bank.
  - — Zero latency between read and write operations
- **Flexible Bank™ architecture**
  - — Read may occur in any of the three banks not being written or erased.
  - — Four banks may be grouped by customer to achieve desired bank divisions.
- **Manufactured on 0.17 µm process technology**
- **SecSi™ (Secured Silicon) Sector: Extra 256 Byte sector**
  - — *Factory locked and identifiable:* 16 bytes available for secure, random factory Electronic Serial Number; verifiable as factory locked through autoselect function. ExpressFlash option allows entire sector to be available for factory-secured data
  - — *Customer lockable:* Sector is one-time programmable. Once sector is locked, data cannot be changed.
- **Zero Power Operation**
  - — Sophisticated power management circuits reduce power consumed during inactive periods to nearly zero.
- **Top or bottom boot sectors**
- **Compatible with JEDEC standards**
  - — Pinout and software compatible with single-power-supply flash standard

#### PERFORMANCE CHARACTERISTICS

- **High performance**
  - — Access time as fast as 70 ns
  - — Program time: 4 µs/word typical utilizing Accelerate function
- **Ultra low power consumption (typical values)**
  - — 2 mA active read current at 1 MHz
  - — 10 mA active read current at 5 MHz
  - — 200 nA in standby or automatic sleep mode
- **Minimum 1 million write cycles guaranteed per sector**
- **20 year data retention at 125°C**
  - — Reliable operation for the life of the system

#### SOFTWARE FEATURES

- **Data Management Software (DMS)**
  - — AMD-supplied software manages data programming, enabling EEPROM emulation
  - — Eases historical sector erase flash limitations
- **Supports Common Flash Memory Interface (CFI)**
- **Program/Erase Suspend/Erase Resume**
  - — Suspends program/erase operations to allow programming/erasing in same bank
- **Data# Polling and Toggle Bits**
  - — Provides a software method of detecting the status of program or erase cycles
- **Unlock Bypass Program command**
  - — Reduces overall programming time when issuing multiple program command sequences

#### HARDWARE FEATURES

- **Any combination of sectors can be erased**
- **Ready/Busy# output (RY/BY#)**
  - — Hardware method for detecting program or erase cycle completion
- **Hardware reset pin (RESET#)**
  - — Hardware method of resetting the internal state machine to the read mode
- **WP#/ACC input pin**
  - — Write protect (WP#) function protects sectors 0 and 1 (bottom boot) or 69 and 70 (top boot), regardless of sector protect status
  - — Acceleration (ACC) function accelerates program timing
- **Sector protection**
  - — Hardware method of locking a sector, either in-system or using programming equipment, to prevent any program or erase operation within that sector
  - — Temporary Sector Unprotect allows changing data in protected sectors in-system

### Pseudo SRAM Features

- **Power dissipation**
  - — Operating: 30 mA maximum
  - — Standby: 100 µA maximum
- **CE1s# and CE2s Chip Select**
- **Power down features using CE1s# and CE2s**
- **Data retention supply voltage: 2.7 to 3.3 volt**
- **Byte data control: LB#s (DQ7–DQ0), UB#s (DQ15–DQ8)**



Publication# **26460** Rev: **B** Amendment/**+1**
Issue Date: **March 12, 2004**

# GENERAL DESCRIPTION

## Am29DL320G Features

The Am29DL320G is a 32 megabit, 3.0 volt-only flash memory device, organized as 2,097,152 words of 16 bits each or 4,194,304 bytes of 8 bits each. Word mode data appears on DQ15–DQ0; byte mode data appears on DQ7–DQ0. The device is designed to be programmed in-system with the standard 3.0 volt $V_{CC}$ supply, and can also be programmed in standard EPROM programmers.

The device is available with an access time of 70 or 85 ns and is offered in a 73-ball FBGA package. Standard control pins—chip enable (CE#f), write enable (WE#), and output enable (OE#)—control normal read and write operations, and avoid bus contention issues.

The device requires only a **single 3.0 volt power supply** for both read and write functions. Internally generated and regulated voltages are provided for the program and erase operations.

## Simultaneous Read/Write Operations with Zero Latency

The Simultaneous Read/Write architecture provides **simultaneous operation** by dividing the memory space into **four banks,** two 4 Mb banks with small and large sectors, and two 12 Mb banks of large sectors only. Sector addresses are fixed, system software can be used to form user-defined bank groups.

During an Erase/Program operation, any of the three non-busy banks may be read from. Note that only two banks can operate simultaneously. The device can improve overall system performance by allowing a host system to program or erase in one bank, then immediately and simultaneously read from the other bank, with zero latency. This releases the system from waiting for the completion of program or erase operations.

The Am29DL320G can be organized as both a top and bottom boot sector configuration.

| Bank | Megabits | Sector Sizes |
|---|---|---|
| Bank 1 | 4 Mb | Eight 8 Kbyte/4 Kword,<br>Seven 64 Kbyte/32 Kword |
| Bank 2 | 12 Mb | Forty-eight 64 Kbyte/32 Kword |
| Bank 3 | 12 Mb | Forty-eight 64 Kbyte/32 Kword |
| Bank 4 | 4 Mb | Sixteen 64 Kbyte/32 Kword |

The **SecSi™ (Secured Silicon) Sector** is an extra 256 byte sector capable of being permanently locked by AMD or customers. The **SecSi Indicator Bit** (DQ7) is permanently set to a 1 if the part is **factory locked**, and set to a 0 if **customer lockable**. This way, customer lockable parts can never be used to replace a factory locked part.

Factory locked parts provide several options. The SecSi Sector may store a secure, random 16 byte ESN (Electronic Serial Number), customer code (programmed through AMD's ExpressFlash service), or both. Customer Lockable parts may utilize the SecSi Sector as a one-time programmable area.

**DMS (Data Management Software)** allows systems to easily take advantage of the advanced architecture of the simultaneous read/write product line by allowing removal of EEPROM devices. DMS will also allow the system software to be simplified, as it will perform all functions necessary to modify data in file structures, as opposed to single-byte modifications. To write or update a particular piece of data (a phone number or configuration data, for example), the user only needs to state which piece of data is to be updated, and where the updated data is located in the system. This is an advantage compared to systems where user-written software must keep track of the old data location, status, logical to physical translation of the data onto the Flash memory device (or memory devices), and more. Using DMS, user-written software does not need to interface with the Flash memory directly. Instead, the user's software accesses the Flash memory by calling one of only six functions. AMD provides this software to simplify system design and software integration efforts.

The device offers complete compatibility with the **JEDEC single-power-supply Flash command set standard**. Commands are written to the command register using standard microprocessor write timings. Reading data out of the device is similar to reading from other Flash or EPROM devices.

The host system can detect whether a program or erase operation is complete by using the device **status bits:** RY/BY# pin, DQ7 (Data# Polling) and DQ6/DQ2 (toggle bits). After a program or erase cycle has been completed, the device automatically returns to the read mode.

The **sector erase architecture** allows memory sectors to be erased and reprogrammed without affecting the data contents of other sectors. The device is fully erased when shipped from the factory.

**Hardware data protection** measures include a low $V_{CC}$ detector that automatically inhibits write operations during power transitions. The **hardware sector protection** feature disables both program and erase operations in any combination of the sectors of memory. This can be achieved in-system or via programming equipment.

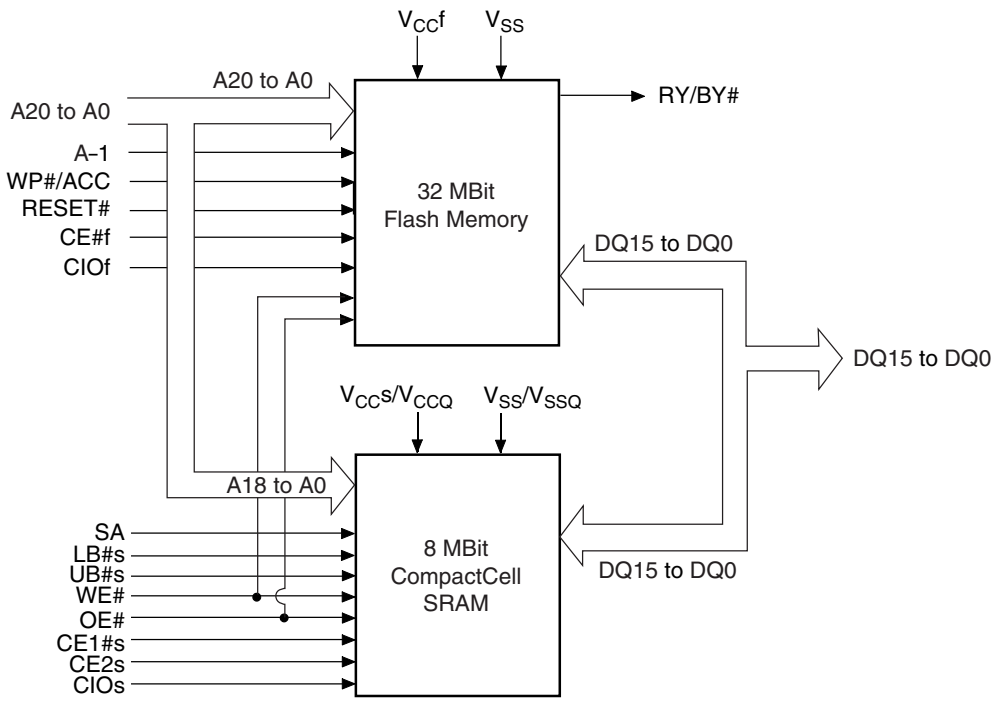
The device offers two power-saving features. When addresses have been stable for a specified amount of time, the device enters the **automatic sleep mode**. The system can also place the device into the **standby mode**. Power consumption is greatly reduced in both modes.

## TABLE OF CONTENTS

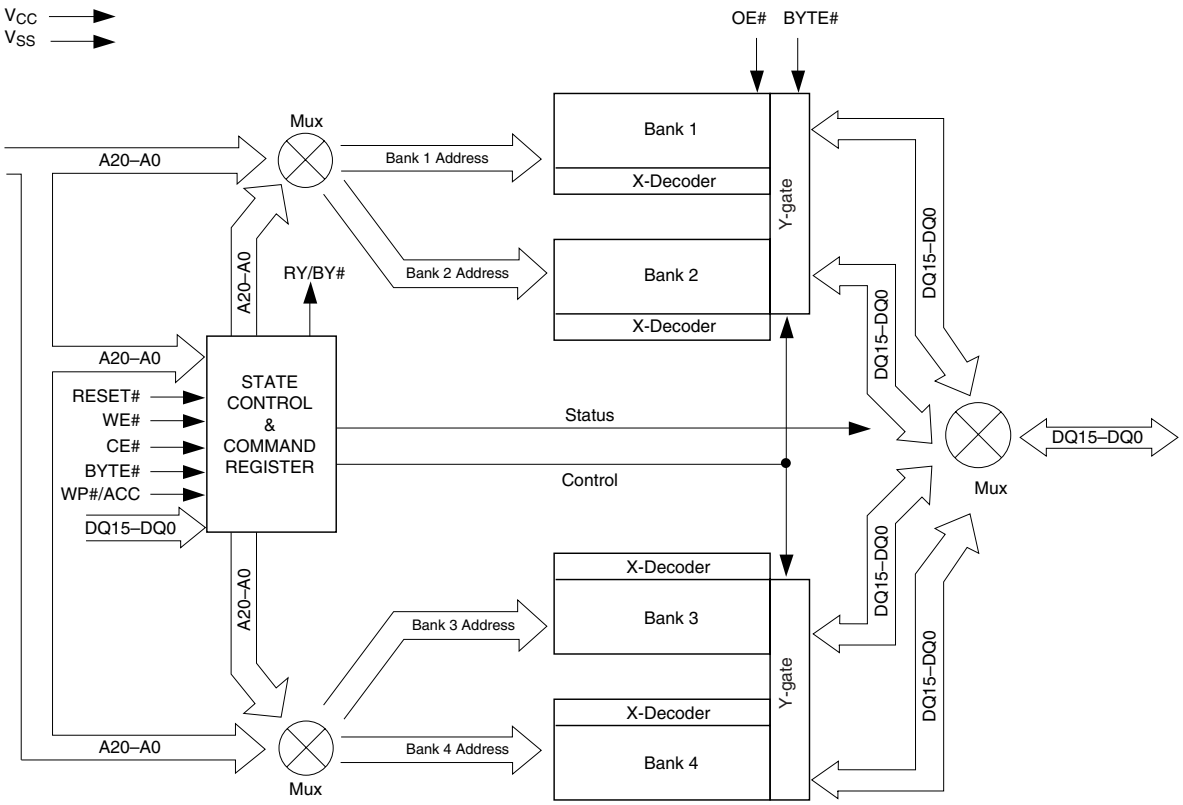
## PRODUCT SELECTOR GUIDE

| Part Number | | Am45DL3208G | | | |
|---|---|---|---|---|---|
| **Speed Options** | Standard Voltage Range: $V_{CC}$ = 2.7–3.3 V | **Flash Memory** | | **Pseudo SRAM** | |
| | | **70** | **85** | **70** | **85** |
| Max Access Time (ns) | | 70 | 85 | 70 | 85 |
| CE#f Access (ns) | | 70 | 85 | 70 | 85 |
| OE# Access (ns) | | 30 | 40 | 35 | 45 |

## MCP BLOCK DIAGRAM

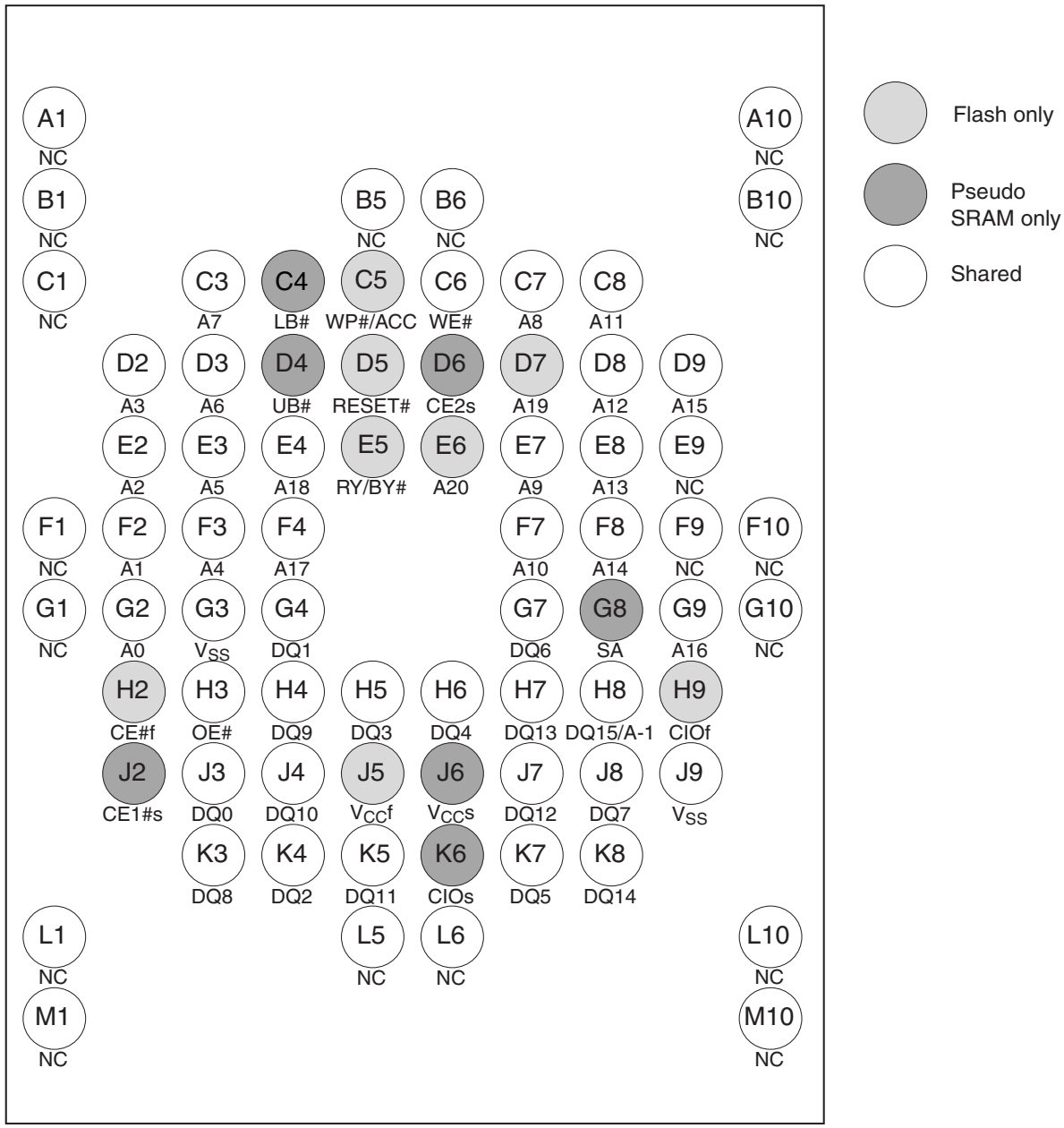


## FLASH MEMORY BLOCK DIAGRAM



## CONNECTION DIAGRAM

**73-Ball FBGA**
Top View




### Special Package Handling Instructions

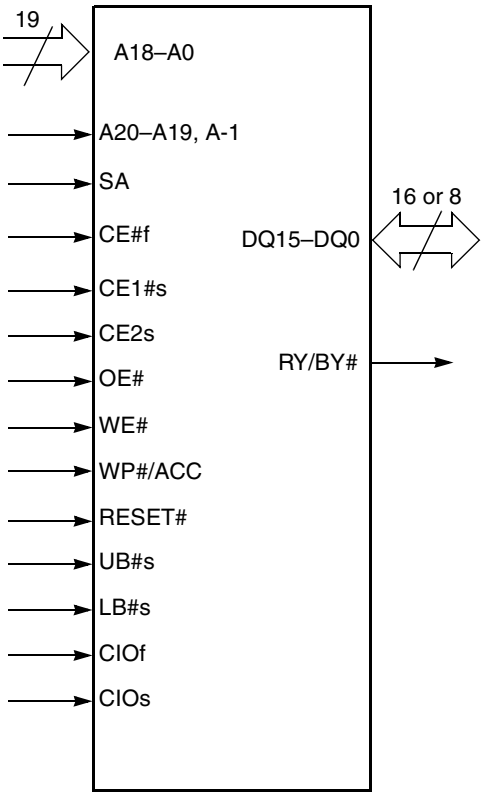
Special handling is required for Flash Memory products in molded packages (TSOP, BGA, PDIP, SSOP, PLCC).

The package and/or data integrity may be compromised if the package body is exposed to temperatures above 150°C for prolonged periods of time.

## PIN DESCRIPTION

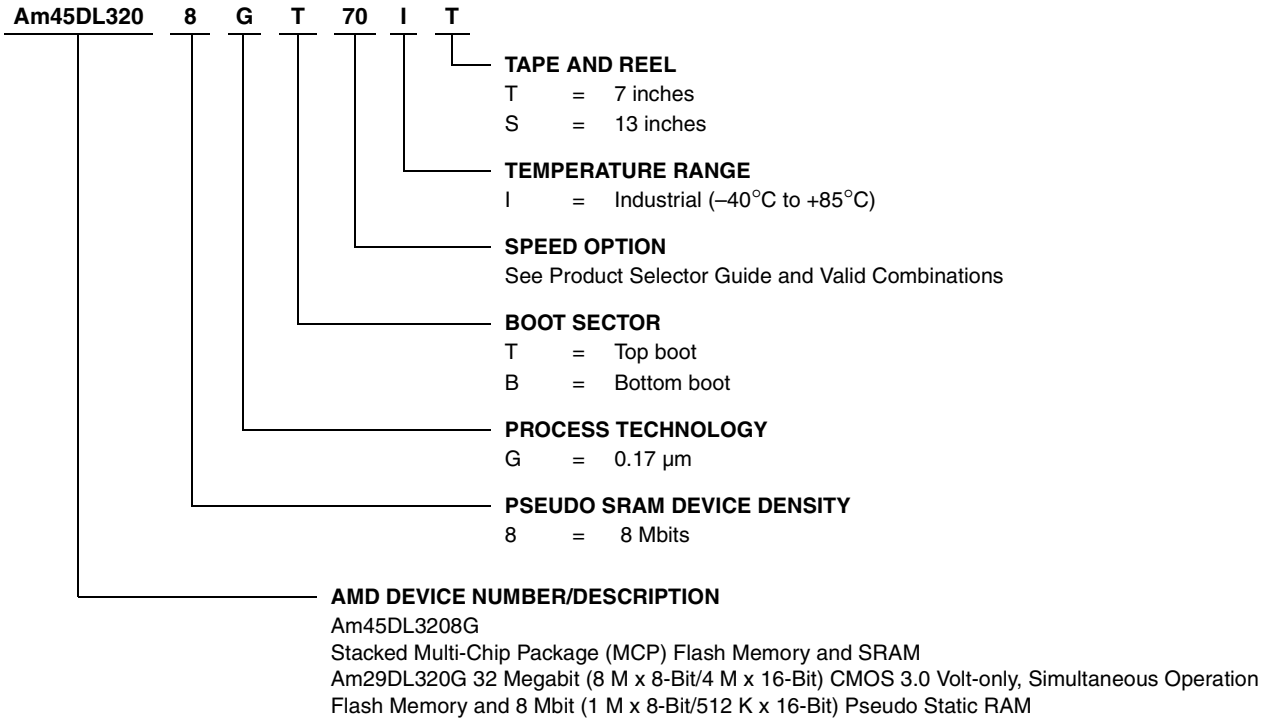
| | | |
|---|---|---|
| A18–A0 | = | 19 Address Inputs (Common) |
| A20–A19, A-1 | = | 3 Address Inputs (Flash) |
| SA | = | Lowest Order Address Pin (PSRAM) Byte mode |
| DQ15–DQ0 | = | 16 Data Inputs/Outputs (Common) |
| CE#f | = | Chip Enable (Flash) |
| CE#1s | = | Chip Enable 1 (PSRAM) |
| CE2s | = | Chip Enable 2 (PSRAM) |
| OE# | = | Output Enable (Common) |
| WE# | = | Write Enable (Common) |
| RY/BY# | = | Ready/Busy Output |
| UB#s | = | Upper Byte Control (PSRAM) |
| LB#s | = | Lower Byte Control (PSRAM) |
| CIOf | = | I/O Configuration (Flash) CIOf = $V_{IH}$ = Word mode (x16), CIOf = $V_{IL}$ = Byte mode (x8) |
| CIOs | = | I/O Configuration (PSRAM) CIOs = $V_{IH}$ = Word mode (x16), CIOs = $V_{IL}$ = Byte mode (x8) |
| RESET# | = | Hardware Reset Pin, Active Low |
| WP#/ACC | = | Hardware Write Protect/ Acceleration Pin (Flash) |
| $V_{CC}$f | = | Flash 3.0 volt-only single power supply (see Product Selector Guide for speed options and voltage supply tolerances) |
| $V_{CC}$s | = | PSRAM Power Supply |
| $V_{SS}$ | = | Device Ground (Common) |
| NC | = | Pin Not Connected Internally |

## LOGIC SYMBOL



## ORDERING INFORMATION

The order number (Valid Combination) is formed by the following:




**Valid Combinations**

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations and to check on newly released combinations.

| Valid Combinations | | |
|---|---|---|
| **Order Number** | | **Package Marking** |
| Am45DL3208GT70I<br>Am45DL3208GB70I | T, S | M450000008<br>M450000009 |
| Am45DL3208GT85I<br>Am45DL3208GB85I | T, S | M45000000A<br>M45000000B |

## MCP DEVICE BUS OPERATIONS

This section describes the requirements and use of the device bus operations, which are initiated through the internal command register. The command register itself does not occupy any addressable memory location. The register is a latch used to store the commands, along with the address and data information needed to execute the command. The contents of the register serve as inputs to the internal state machine. The state machine outputs dictate the function of the device. Tables 1-3 lists the device bus operations, the inputs and control levels they require, and the resulting output. The following subsections describe each of these operations in further detail.

**Table 1. Device Bus Operations—Flash Word Mode, CIOf = $V_{IH}$; P SRAM Word Mode, CIOs = $V_{CC}$**

| Operation (Notes 1, 2) | CE#f | CE1#s | CE2s | OE# | WE# | SA | Addr. | LB#s | UB#s | RESET# | WP#/ACC (Note 4) | DQ7–DQ0 | DQ15–DQ8 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Read from Flash | L | H | X | L | H | X | $A_{IN}$ | X | X | H | L/H | $D_{OUT}$ | $D_{OUT}$ |
| | | X | L | | | | | | | | | | |
| Write to Flash | L | H | X | H | L | X | $A_{IN}$ | X | X | H | (Note 4) | $D_{IN}$ | $D_{IN}$ |
| | | X | L | | | | | | | | | | |
| Standby | $V_{CC} \pm 0.3$ V | H | X | X | X | X | X | X | X | $V_{CC} \pm 0.3$ V | H | High-Z | High-Z |
| | | X | L | | | | | | | | | | |
| Output Disable | L | L | H | H | H | X | X | L | X | H | L/H | High-Z | High-Z |
| | | | | H | H | X | X | X | L | | | | |
| Flash Hardware Reset | X | H | X | X | X | X | X | X | X | L | L/H | High-Z | High-Z |
| | | X | L | | | | | | | | | | |
| Sector Protect (Note 5) | L | H | X | H | L | X | SADD, A6 = L, A1 = H, A0 = L | X | X | $V_{ID}$ | L/H | $D_{IN}$ | X |
| | | X | L | | | | | | | | | | |
| Sector Unprotect (Note 5) | L | H | X | H | L | X | SADD, A6 = H, A1 = H, A0 = L | X | X | $V_{ID}$ | (Note 6) | $D_{IN}$ | X |
| | | X | L | | | | | | | | | | |
| Temporary Sector Unprotect | X | H | X | X | X | X | X | X | X | $V_{ID}$ | (Note 6) | $D_{IN}$ | High-Z |
| | | X | L | | | | | | | | | | |
| Read from PSRAM | H | L | H | L | H | X | $A_{IN}$ | L | L | H | X | $D_{OUT}$ | $D_{OUT}$ |
| | | | | | | | | H | L | | | High-Z | $D_{OUT}$ |
| | | | | | | | | L | H | | | $D_{OUT}$ | High-Z |
| Write to PSRAM | H | L | H | X | L | X | $A_{IN}$ | L | L | H | X | $D_{IN}$ | $D_{IN}$ |
| | | | | | | | | H | L | | | High-Z | $D_{IN}$ |
| | | | | | | | | L | H | | | $D_{IN}$ | High-Z |

***Legend:*** *L = Logic Low = $V_{IL}$, H = Logic High = $V_{IH}$, $V_{ID}$ = 11.5–12.5 V, $V_{HH}$ = 9.0 ± 0.5 V, X = Don't Care, SA = PSRAM Address Input, Byte Mode, SADD = Flash Sector Address, $A_{IN}$ = Address In, $D_{IN}$ = Data In, $D_{OUT}$ = Data Out*

***Notes:***

1. *Other operations except for those indicated in this column are inhibited.*
2. *Do not apply CE#f = $V_{IL}$, CE1#s = $V_{IL}$ and CE2s = $V_{IH}$ at the same time.*
3. *Don't care or open LB#s or UB#s.*
4. *If WP#/ACC = $V_{IL}$ , the boot sectors will be protected. If WP#/ACC = $V_{IH}$ the boot sectors protection will be removed. If WP#/ACC = $V_{ACC}$ (9V), the program time will be reduced by 40%.*
5. *The sector protect and sector unprotect functions may also be implemented via programming equipment. See the "Sector/Sector Block Protection and Unprotection" section.*
6. *If WP#/ACC = $V_{IL}$, the two outermost boot sectors remain protected. If WP#/ACC = $V_{IH}$, the two outermost boot sector protection depends on whether they were last protected or unprotected using the method described in "Sector/Sector Block Protection and Unprotection". If WP#/ACC = $V_{HH,}$ all sectors will be unprotected.*

**Table 2. Device Bus Operations—Flash Word Mode, CIOf = $V_{IH}$; PSRAM Byte Mode, CIOs = $V_{SS}$**

| Operation (Notes 1, 2) | CE#f | CE1#s | CE2s | OE# | WE# | SA | Addr. | LB#s (Note 3) | UB#s (Note 3) | RESET# | WP#/ACC (Note 4) | DQ7–DQ0 | DQ15–DQ8 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Read from Flash | L | H | X | L | H | X | $A_{IN}$ | X | X | H | L/H | $D_{OUT}$ | $D_{OUT}$ |
| | | X | L | | | | | | | | | | |
| Write to Flash | L | H | X | H | L | X | $A_{IN}$ | X | X | H | (Note 3) | $D_{IN}$ | $D_{IN}$ |
| | | X | L | | | | | | | | | | |
| Standby | $V_{CC} \pm 0.3$ V | H | X | X | X | X | X | X | X | $V_{CC} \pm 0.3$ V | H | High-Z | High-Z |
| | | X | L | | | | | | | | | | |
| Output Disable | L | L | H | H | H | SA | X | X | X | H | L/H | High-Z | High-Z |
| Flash Hardware Reset | X | H | X | X | X | X | X | X | X | L | L/H | High-Z | High-Z |
| | | X | L | | | | | | | | | | |
| Sector Protect (Note 5) | L | H | X | H | L | X | SADD, A6 = L, A1 = H, A0 = L | X | X | $V_{ID}$ | L/H | $D_{IN}$ | X |
| | | X | L | | | | | | | | | | |
| Sector Unprotect (Note 5) | L | H | X | H | L | X | SADD, A6 = H, A1 = H, A0 = L | X | X | $V_{ID}$ | (Note 6) | $D_{IN}$ | X |
| | | X | L | | | | | | | | | | |
| Temporary Sector Unprotect | X | H | X | X | X | X | $A_{IN}$ | X | X | $V_{ID}$ | (Note 6) | $D_{IN}$ | High-Z |
| | | X | L | | | | | | | | | | |
| Read from PSRAM | H | L | H | L | H | SA | $A_{IN}$ | X | X | H | X | $D_{OUT}$ | High-Z |
| Write to PSRAM | H | L | H | X | L | SA | $A_{IN}$ | X | X | H | X | $D_{IN}$ | High-Z |

***Legend:*** *L = Logic Low = $V_{IL}$, H = Logic High = $V_{IH}$, $V_{ID}$ = 11.5–12.5 V, $V_{HH}$ = 9.0 ± 0.5 V, X = Don't Care, SA = PSRAM Address Input, Byte Mode, SADD = Flash Sector Address, $A_{IN}$ = Address In, $D_{IN}$ = Data In, $D_{OUT}$ = Data Out*

***Notes:***

1. *Other operations except for those indicated in this column are inhibited.*
2. *Do not apply CE#f = $V_{IL}$, CE1#s = $V_{IL}$ and CE2s = $V_{IH}$ at the same time.*
3. *Don't care or open LB#s or UB#s.*
4. *If WP#/ACC = $V_{IL}$ , the boot sectors will be protected. If WP#/ACC = $V_{IH}$ the boot sectors protection will be removed. If WP#/ACC = $V_{ACC}$ (9V), the program time will be reduced by 40%.*
5. *The sector protect and sector unprotect functions may also be implemented via programming equipment. See the "Sector/Sector Block Protection and Unprotection" section.*
6. *If WP#/ACC = $V_{IL}$, the two outermost boot sectors remain protected. If WP#/ACC = $V_{IH}$, the two outermost boot sector protection depends on whether they were last protected or unprotected using the method described in "Sector/Sector Block Protection and Unprotection". If WP#/ACC = $V_{HH,}$ all sectors will be unprotected.*

**Table 3. Device Bus Operations—Flash Byte Mode, CIOf = $V_{SS}$; PSRAM Word Mode, CIOs = $V_{CC}$**

| Operation (Notes 1, 2) | CE#f | CE1#s | CE2s | OE# | WE# | SA | Addr. | LB#s (Note 3) | UB#s (Note 3) | RESET# | WP#/ACC (Note 4) | DQ7–DQ0 | DQ15–DQ8 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Read from Flash | L | H | X | L | H | X | $A_{IN}$ | X | X | H | L/H | $D_{OUT}$ | High-Z |
| | | X | L | | | | | | | | | | |
| Write to Flash | L | H | X | H | L | X | $A_{IN}$ | X | X | H | (Note 3) | $D_{IN}$ | High-Z |
| | | X | L | | | | | | | | | | |
| Standby | $V_{CC} \pm 0.3$ V | H | X | X | X | X | X | X | X | $V_{CC} \pm 0.3$ V | H | High-Z | High-Z |
| | | X | L | | | | | | | | | | |
| Output Disable | L | L | H | H | H | X | X | L | X | H | L/H | High-Z | High-Z |
| | | | | | | | | X | L | | | | |
| Flash Hardware Reset | X | H | X | X | X | X | X | X | X | L | L/H | High-Z | High-Z |
| | | X | L | | | | | | | | | | |
| Sector Protect (Note 5) | L | H | X | H | L | X | SADD, A6 = L, A1 = H, A0 = L | X | X | $V_{ID}$ | L/H | $D_{IN}$ | X |
| | | X | L | | | | | | | | | | |
| Sector Unprotect (Note 5) | L | H | X | H | L | X | SADD, A6 = L, A1 = H, A0 = L | X | X | $V_{ID}$ | (Note 6) | $D_{IN}$ | X |
| | | X | L | | | | | | | | | | |
| Temporary Sector Unprotect | X | H | x | X | X | X | $A_{IN}$ | X | X | $V_{ID}$ | (Note 6) | $D_{IN}$ | High-Z |
| | | X | L | | | | | | | | | | |
| Read from PSRAM | H | L | H | L | H | X | $A_{IN}$ | L | L | H | X | $D_{OUT}$ | $D_{OUT}$ |
| | | | | | | | | H | L | | | High-Z | $D_{OUT}$ |
| | | | | | | | | L | H | | | $D_{OUT}$ | High-Z |
| Write to PSRAM | H | L | H | X | L | X | $A_{IN}$ | L | L | H | X | $D_{IN}$ | $D_{IN}$ |
| | | | | | | | | H | L | | | High-Z | $D_{IN}$ |
| | | | | | | | | L | H | | | $D_{IN}$ | High-Z |

***Legend:*** *L = Logic Low = $V_{IL}$, H = Logic High = $V_{IH}$, $V_{ID}$ = 11.5–12.5 V, $V_{HH}$ = 9.0 ± 0.5 V, X = Don't Care, SA = PSRAM Address Input, Byte Mode, SADD = Flash Sector Address, $A_{IN}$ = Address In (for Flash Byte Mode, DQ15 = A-1), $D_{IN}$ = Data In, $D_{OUT}$ = Data Out*

***Notes:***

1. *Other operations except for those indicated in this column are inhibited.*
2. *Do not apply CE#f = $V_{IL}$, CE1#s = $V_{IL}$ and CE2s = $V_{IH}$ at the same time.*
3. *Don't care or open LB#s or UB#s.*
4. *If WP#/ACC = $V_{IL}$ , the boot sectors will be protected. If WP#/ACC = $V_{IH}$ the boot sectors protection will be removed. If WP#/ACC = $V_{ACC}$ (9V), the program time will be reduced by 40%.*
5. *The sector protect and sector unprotect functions may also be implemented via programming equipment. See the "Sector/Sector Block Protection and Unprotection" section.*
6. *If WP#/ACC = $V_{IL}$, the two outermost boot sectors remain protected. If WP#/ACC = $V_{IH}$, the two outermost boot sector protection depends on whether they were last protected or unprotected using the method described in "Sector/Sector Block Protection and Unprotection". If WP#/ACC = $V_{HH}$, all sectors will be unprotected.*

**Table 4. Device Bus Operations—Flash Byte Mode, CIOf = $V_{IL}$; PSRAM Byte Mode, CIOs = $V_{SS}$**

| Operation (Notes 1, 2) | CE#f | CE1#s | CE2s | OE# | WE# | SA | Addr. | LB#s (Note 3) | UB#s (Note 3) | RESET# | WP#/ACC (Note 4) | DQ7–DQ0 | DQ15–DQ8 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Read from Flash | L | H | X | L | H | X | $A_{IN}$ | X | X | H | L/H | $D_{OUT}$ | High-Z |
| | | X | L | | | | | | | | | | |
| Write to Flash | L | H | X | H | L | X | $A_{IN}$ | X | X | H | (Note 3) | $D_{IN}$ | High-Z |
| | | X | L | | | | | | | | | | |
| Standby | $V_{CC} \pm 0.3$ V | H | X | X | X | X | X | X | X | $V_{CC} \pm 0.3$ V | H | High-Z | High-Z |
| | | X | L | | | | | | | | | | |
| Output Disable | H | L | H | H | H | SA | X | X | X | H | L/H | High-Z | High-Z |
| Flash Hardware Reset | X | H | X | X | X | X | X | X | X | L | L/H | High-Z | High-Z |
| | | X | L | | | | | | | | | | |
| Sector Protect (Note 5) | L | H | X | H | L | X | SADD, A6 = L, A1 = H, A0 = L | X | X | $V_{ID}$ | L/H | $D_{IN}$ | X |
| | | X | L | | | | | | | | | | |
| Sector Unprotect (Note 5) | L | H | X | H | L | X | SADD, A6 = L, A1 = H, A0 = L | X | X | $V_{ID}$ | (Note 6) | $D_{IN}$ | X |
| | | X | L | | | | | | | | | | |
| Temporary Sector Unprotect | X | H | X | X | X | X | $A_{IN}$ | X | X | $V_{ID}$ | (Note 6) | $D_{IN}$ | High-Z |
| | | X | L | | | | | | | | | | |
| Read from SRAM | H | L | H | L | H | SA | $A_{IN}$ | X | X | H | X | $D_{OUT}$ | High-Z |
| Write to SRAM | H | L | H | X | L | SA | $A_{IN}$ | X | X | H | X | $D_{IN}$ | High-Z |

***Legend:*** *L = Logic Low = $V_{IL}$, H = Logic High = $V_{IH}$, $V_{ID}$ = 11.5–12.5 V, $V_{HH}$ = 9.0 ± 0.5 V, X = Don't Care, SA = PSRAM Address Input, Byte Mode, SADD = Flash Sector Address, $A_{IN}$ = Address In (for Flash Byte Mode, DQ15 = A-1), $D_{IN}$ = Data In, $D_{OUT}$ = Data Out*

***Notes:***

1. *Other operations except for those indicated in this column are inhibited.*
2. *Do not apply CE#f = $V_{IL}$, CE1#s = $V_{IL}$ and CE2s = $V_{IH}$ at the same time.*
3. *Don't care or open LB#s or UB#s.*
4. *If WP#/ACC = $V_{IL}$ , the boot sectors will be protected. If WP#/ACC = $V_{IH}$ the boot sectors protection will be removed. If WP#/ACC = $V_{ACC}$ (9V), the program time will be reduced by 40%.*
5. *The sector protect and sector unprotect functions may also be implemented via programming equipment. See the "Sector/Sector Block Protection and Unprotection".*
6. *If WP#/ACC = $V_{IL}$, the two outermost boot sectors remain protected. If WP#/ACC = $V_{IH}$, the two outermost boot sector protection depends on whether they were last protected or unprotected using the method described in "Sector/Sector Block Protection and Unprotection". If WP#/ACC = $V_{HH}$, all sectors will be unprotected.*

## FLASH DEVICE BUS OPERATIONS

### Word/Byte Configuration

The CIOf pin controls whether the device data I/O pins operate in the byte or word configuration. If the CIOf pin is set at logic '1', the device is in word configuration, DQ15–DQ0 are active and controlled by CE#f and OE#.

If the CIOf pin is set at logic '0', the device is in byte configuration, and only data I/O pins DQ7–DQ0 are active and controlled by CE#f and OE#. The data I/O pins DQ14–DQ8 are tri-stated, and the DQ15 pin is used as an input for the LSB (A-1) address function.

### Requirements for Reading Array Data

To read array data from the outputs, the system must drive the CE#f and OE# pins to $V_{IL}$. CE#f is the power control and selects the device. OE# is the output control and gates array data to the output pins. WE# should remain at $V_{IH}$. The CIOf pin determines

whether the device outputs array data in words or bytes.

The internal state machine is set for reading array data upon device power-up, or after a hardware reset. This ensures that no spurious alteration of the memory content occurs during the power transition. No command is necessary in this mode to obtain array data. Standard microprocessor read cycles that assert valid addresses on the device address inputs produce valid data on the device data outputs. Each bank remains enabled for read access until the command register contents are altered.

Refer to the AC Read-Only Operations table for timing specifications and to Figure 15 for the timing diagram. $I_{CC1}$ in the DC Characteristics table represents the active current specification for reading array data.

## Writing Commands/Command Sequences

To write a command or command sequence (which includes programming data to the device and erasing sectors of memory), the system must drive WE# and CE#f to $V_{IL}$, and OE# to $V_{IH}$.

For program operations, the CIOf pin determines whether the device accepts program data in bytes or words. Refer to “Word/Byte Configuration” for more information.

The device features an **Unlock Bypass** mode to facilitate faster programming. Once a bank enters the Unlock Bypass mode, only two write cycles are required to program a word or byte, instead of four. The “Byte/Word Program Command Sequence” section has details on programming data to the device using both standard and Unlock Bypass command sequences.

An erase operation can erase one sector, multiple sectors, or the entire device. Tables 5 and 7 indicate the address space that each sector occupies. Similarly, a “sector address” is the address bits required to uniquely select a sector. The “Flash Command Definitions” section has details on erasing a sector or the entire chip, or suspending/resuming the erase operation.

The device address space is divided into four banks. A “bank address” is the address bits required to uniquely select a bank.

$I_{CC2}$ in the DC Characteristics table represents the active current specification for the write mode. The Flash AC Characteristics section contains timing specification tables and timing diagrams for write operations.

### Accelerated Program Operation

The device offers accelerated program operations through the ACC function. This is one of two functions provided by the WP#/ACC pin. This function is primarily intended to allow faster manufacturing throughput at the factory.

If the system asserts $V_{HH}$ on this pin, the device automatically enters the aforementioned Unlock Bypass mode, temporarily unprotects any protected sectors, and uses the higher voltage on the pin to reduce the time required for program operations. The system would use a two-cycle program command sequence as required by the Unlock Bypass mode. Removing $V_{HH}$ from the WP#/ACC pin returns the device to normal operation. *Note that $V_{HH}$ must not be asserted on WP#/ACC for operations other than accelerated programming, or device damage may result. In addition, the WP#/ACC pin must not be left floating or unconnected; inconsistent behavior of the device may result.* See “Write Protect (WP#)” on page 20 for related information.

### Autoselect Functions

If the system writes the autoselect command sequence, the device enters the autoselect mode. The system can then read autoselect codes from the internal register (which is separate from the memory array) on DQ15–DQ0. Standard read cycle timings apply in this mode. Refer to the Sector/Sector Block Protection and Unprotection and Autoselect Command Sequence sections for more information.

## Simultaneous Read/Write Operations with Zero Latency

This device is capable of reading data from one bank of memory while programming or erasing in the other bank of memory. An erase operation may also be suspended to read from or program to another location within the same bank (except the sector being erased). Figure 22 shows how read and write cycles may be initiated for simultaneous operation with zero latency. $I_{CC6}$f and $I_{CC7}$f in the  table represent the current specifications for read-while-program and read-while-erase, respectively.

## Standby Mode

When the system is not reading or writing to the device, it can place the device in the standby mode. In this mode, current consumption is greatly reduced, and the outputs are placed in the high impedance state, independent of the OE# input.

The device enters the CMOS standby mode when the CE#f and RESET# pins are both held at $V_{CC}$ ± 0.3 V. (Note that this is a more restricted voltage range than $V_{IH}$.) If CE#f and RESET# are held at $V_{IH}$, but not within $V_{CC}$ ± 0.3 V, the device will be in the standby mode, but the standby current will be greater. The device requires standard access time ($t_{CE}$) for read access when the device is in either of these standby modes, before it is ready to read data.

If the device is deselected during erasure or programming, the device draws active current until the operation is completed.

$I_{CC3}$f in the table represents the standby current specification.

## Automatic Sleep Mode

The automatic sleep mode minimizes Flash device energy consumption. The device automatically enables this mode when addresses remain stable for $t_{ACC}$ + 30 ns. The automatic sleep mode is independent of the CE#f, WE#, and OE# control signals. Standard address access timings provide new data when addresses are changed. While in sleep mode, output data is latched and always available to the system. $I_{CC5}$f in the table represents the automatic sleep mode current specification.

## RESET#: Hardware Reset Pin

The RESET# pin provides a hardware method of resetting the device to reading array data. When the RESET# pin is driven low for at least a period of $t_{RP}$, the device immediately terminates any operation in progress, tristates all output pins, and ignores all read/write commands for the duration of the RESET# pulse. The device also resets the internal state machine to reading array data. The operation that was interrupted should be reinitiated once the device is ready to accept another command sequence, to ensure data integrity.

Current is reduced for the duration of the RESET# pulse. When RESET# is held at $V_{SS}$±0.3 V, the device draws CMOS standby current ($I_{CC4}$f). If RESET# is held at $V_{IL}$ but not within $V_{SS}$±0.3 V, the standby current will be greater.

The RESET# pin may be tied to the system reset circuitry. A system reset would thus also reset the Flash memory, enabling the system to read the boot-up firmware from the Flash memory.

If RESET# is asserted during a program or erase operation, the RY/BY# pin remains a “0” (busy) until the internal reset operation is complete, which requires a time of $t_{READY}$ (during Embedded Algorithms). The system can thus monitor RY/BY# to determine whether the reset operation is complete. If RESET# is asserted when a program or erase operation is not executing (RY/BY# pin is “1”), the reset operation is completed within a time of $t_{READY}$ (not during Embedded Algorithms). The system can read data $t_{RH}$ after the RESET# pin returns to $V_{IH}$.

Refer to the AC Characteristics tables for RESET# parameters and to Figure 16 for the timing diagram.

## Output Disable Mode

When the OE# input is at $V_{IH}$, output from the device is disabled. The output pins are placed in the high impedance state.

**Table 5. Top Boot Sector Addresses**

| Bank | Sector | Sector Address A20–A12 | Sector Size (Kbytes/Kwords) | (x8) Address Range | (x16) Address Range |
|---|---|---|---|---|---|
| Bank 4 | SA0 | 000000xxx | 64/32 | 000000h–00FFFFh | 000000h–07FFFh |
| | SA1 | 000001xxx | 64/32 | 010000h–01FFFFh | 008000h–0FFFFh |
| | SA2 | 000010xxx | 64/32 | 020000h–02FFFFh | 010000h–17FFFh |
| | SA3 | 000011xxx | 64/32 | 030000h–03FFFFh | 018000h–01FFFFh |
| | SA4 | 000100xxx | 64/32 | 040000h–04FFFFh | 020000h–027FFFh |
| | SA5 | 000101xxx | 64/32 | 050000h–05FFFFh | 028000h–02FFFFh |
| | SA6 | 000110xxx | 64/32 | 060000h–06FFFFh | 030000h–037FFFh |
| | SA7 | 000111xxx | 64/32 | 070000h–07FFFFh | 038000h–03FFFFh |
| Bank 3 | SA8 | 001000xxx | 64/32 | 080000h–08FFFFh | 040000h–047FFFh |
| | SA9 | 001001xxx | 64/32 | 090000h–09FFFFh | 048000h–04FFFFh |
| | SA10 | 001010xxx | 64/32 | 0A0000h–0AFFFFh | 050000h–057FFFh |
| | SA11 | 001011xxx | 64/32 | 0B0000h–0BFFFFh | 058000h–05FFFFh |
| | SA12 | 001100xxx | 64/32 | 0C0000h–0CFFFFh | 060000h–067FFFh |
| | SA13 | 001101xxx | 64/32 | 0D0000h–0DFFFFh | 068000h–06FFFFh |
| | SA14 | 001110xxx | 64/32 | 0E0000h–0EFFFFh | 070000h–077FFFh |
| | SA15 | 001111xxx | 64/32 | 0F0000h–0FFFFFh | 078000h–07FFFFh |
| | SA16 | 010000xxx | 64/32 | 100000h–10FFFFh | 080000h–087FFFh |
| | SA17 | 010001xxx | 64/32 | 110000h–11FFFFh | 088000h–08FFFFh |
| | SA18 | 010010xxx | 64/32 | 120000h–12FFFFh | 090000h–097FFFh |

**Table 5. Top Boot Sector Addresses (Continued)**

| Bank | Sector | Sector Address A20–A12 | Sector Size (Kbytes/Kwords) | (x8) Address Range | (x16) Address Range |
|---|---|---|---|---|---|
| Bank 3 (continued) | SA19 | 010011xxx | 64/32 | 130000h–13FFFFh | 098000h–09FFFFh |
| | SA20 | 010100xxx | 64/32 | 140000h–14FFFFh | 0A0000h–0A7FFFh |
| | SA21 | 010101xxx | 64/32 | 150000h–15FFFFh | 0A8000h–0AFFFFh |
| | SA22 | 010110xxx | 64/32 | 160000h–16FFFFh | 0B0000h–0B7FFFh |
| | SA23 | 010111xxx | 64/32 | 170000h–17FFFFh | 0B8000h–0BFFFFh |
| | SA24 | 011000xxx | 64/32 | 180000h–18FFFFh | 0C0000h–0C7FFFh |
| | SA25 | 011001xxx | 64/32 | 190000h–19FFFFh | 0C8000h–0CFFFFh |
| | SA26 | 011010xxx | 64/32 | 1A0000h–1AFFFFh | 0D0000h–0D7FFFh |
| | SA27 | 011011xxx | 64/32 | 1B0000h–1BFFFFh | 0D8000h–0DFFFFh |
| | SA28 | 011100xxx | 64/32 | 1C0000h–1CFFFFh | 0E0000h–0E7FFFh |
| | SA29 | 011101xxx | 64/32 | 1D0000h–1DFFFFh | 0E8000h–0EFFFFh |
| | SA30 | 011110xxx | 64/32 | 1E0000h–1EFFFFh | 0F0000h–0F7FFFh |
| | SA31 | 011111xxx | 64/32 | 1F0000h–1FFFFFh | 0F8000h–0FFFFFh |
| Bank 2 | SA32 | 100000xxx | 64/32 | 200000h–20FFFFh | 100000h–107FFFh |
| | SA33 | 100001xxx | 64/32 | 210000h–21FFFFh | 108000h–10FFFFh |
| | SA34 | 100010xxx | 64/32 | 220000h–22FFFFh | 110000h–117FFFh |
| | SA35 | 100011xxx | 64/32 | 230000h–23FFFFh | 118000h–11FFFFh |
| | SA36 | 100100xxx | 64/32 | 240000h–24FFFFh | 120000h–127FFFh |
| | SA37 | 100101xxx | 64/32 | 250000h–25FFFFh | 128000h–12FFFFh |
| | SA38 | 100110xxx | 64/32 | 260000h–26FFFFh | 130000h–137FFFh |
| | SA39 | 100111xxx | 64/32 | 270000h–27FFFFh | 138000h–13FFFFh |
| | SA40 | 101000xxx | 64/32 | 280000h–28FFFFh | 140000h–147FFFh |
| | SA41 | 101001xxx | 64/32 | 290000h–29FFFFh | 148000h–14FFFFh |
| | SA42 | 101010xxx | 64/32 | 2A0000h–2AFFFFh | 150000h–157FFFh |
| | SA43 | 101011xxx | 64/32 | 2B0000h–2BFFFFh | 158000h–15FFFFh |
| | SA44 | 101100xxx | 64/32 | 2C0000h–2CFFFFh | 160000h–167FFFh |
| | SA45 | 101101xxx | 64/32 | 2D0000h–2DFFFFh | 168000h–16FFFFh |
| | SA46 | 101110xxx | 64/32 | 2E0000h–2EFFFFh | 170000h–177FFFh |
| | SA47 | 101111xxx | 64/32 | 2F0000h–2FFFFFh | 178000h–17FFFFh |
| | SA48 | 110000xxx | 64/32 | 300000h–30FFFFh | 180000h–187FFFh |
| | SA49 | 110001xxx | 64/32 | 310000h–31FFFFh | 188000h–18FFFFh |
| | SA50 | 110010xxx | 64/32 | 320000h–32FFFFh | 190000h–197FFFh |
| | SA51 | 110011xxx | 64/32 | 330000h–33FFFFh | 198000h–19FFFFh |
| | SA52 | 110100xxx | 64/32 | 340000h–34FFFFh | 1A0000h–1A7FFFh |
| | SA53 | 110101xxx | 64/32 | 350000h–35FFFFh | 1A8000h–1AFFFFh |
| | SA54 | 110110xxx | 64/32 | 360000h–36FFFFh | 1B0000h–1B7FFFh |
| | SA55 | 110111xxx | 64/32 | 370000h–37FFFFh | 1B8000h–1BFFFFh |
| Bank 1 | SA56 | 111000xxx | 64/32 | 380000h–38FFFFh | 1C0000h–1C7FFFh |
| | SA57 | 111001xxx | 64/32 | 390000h–39FFFFh | 1C8000h–1CFFFFh |
| | SA58 | 111010xxx | 64/32 | 3A0000h–3AFFFFh | 1D0000h–1D7FFFh |
| | SA59 | 111011xxx | 64/32 | 3B0000h–3BFFFFh | 1D8000h–1DFFFFh |
| | SA60 | 111100xxx | 64/32 | 3C0000h–3CFFFFh | 1E0000h–1E7FFFh |
| | SA61 | 111101xxx | 64/32 | 3D0000h–3DFFFFh | 1E8000h–1EFFFFh |
| | SA62 | 111110xxx | 64/32 | 3E0000h–3EFFFFh | 1F0000h–1F7FFFh |
| | SA63 | 111111000 | 8/4 | 3F0000h–3F1FFFh | 1F8000h–1F8FFFh |
| | SA64 | 111111001 | 8/4 | 3F2000h–3F3FFFh | 1F9000h–1F9FFFh |
| | SA65 | 111111010 | 8/4 | 3F4000h–3F5FFFh | 1FA000h–1FAFFFh |
| | SA66 | 111111011 | 8/4 | 3F6000h–3F7FFFh | 1FB000h–1FBFFFh |
| | SA67 | 111111100 | 8/4 | 3F8000h–3F9FFFh | 1FC000h–1FCFFFh |
| | SA68 | 111111101 | 8/4 | 3FA000h–3FBFFFh | 1FD000h–1FDFFFh |
| | SA69 | 111111110 | 8/4 | 3FC000h–3FDFFFh | 1FE000h–1FEFFFh |
| | SA70 | 111111111 | 8/4 | 3FE000h–3FFFFFh | 1FF000h–1FFFFFh |

***Note:*** *The address range is A20:A-1 in byte mode (BYTE#=$V_{IL}$) or A20:A0 in word mode (BYTE#=$V_{IH}$). The bank address bits are A20–A18.*

### Table 6. Top Boot SecSi™ Sector Addresses

| Device | Sector Address A20–A12 | Sector Size (Bytes/Words) | (x8) Address Range | (x16) Address Range |
|---|---|---|---|---|
| Am29DL320GT | 111111xxx | 256/128 | 3FE000h–3FE0FFh | 1F0000h–1FF07Fh |

### Table 7. Bottom Boot Sector Addresses

| | Sector | Sector Address A20–A12 | Sector Size (Kbytes/Kwords) | (x8) Address Range | (x16) Address Range |
|---|---|---|---|---|---|
| Bank 1 | SA0 | 000000000 | 8/4 | 000000h–001FFFh | 000000h–000FFFh |
| | SA1 | 000000001 | 8/4 | 002000h–003FFFh | 001000h–001FFFh |
| | SA2 | 000000010 | 8/4 | 004000h–005FFFh | 002000h–002FFFh |
| | SA3 | 000000011 | 8/4 | 006000h–007FFFh | 003000h–003FFFh |
| | SA4 | 000000100 | 8/4 | 008000h–009FFFh | 004000h–004FFFh |
| | SA5 | 000000101 | 8/4 | 00A000h–00BFFFh | 005000h–005FFFh |
| | SA6 | 000000110 | 8/4 | 00C000h–00DFFFh | 006000h–006FFFh |
| | SA7 | 000000111 | 8/4 | 00E000h–00FFFFh | 007000h–007FFFh |
| | SA8 | 000001xxx | 64/32 | 010000h–01FFFFh | 008000h–00FFFFh |
| | SA9 | 000010xxx | 64/32 | 020000h–02FFFFh | 010000h–017FFFh |
| | SA10 | 000011xxx | 64/32 | 030000h–03FFFFh | 018000h–01FFFFh |
| | SA11 | 000100xxx | 64/32 | 040000h–04FFFFh | 020000h–027FFFh |
| | SA12 | 000101xxx | 64/32 | 050000h–05FFFFh | 028000h–02FFFFh |
| | SA13 | 000110xxx | 64/32 | 060000h–06FFFFh | 030000h–037FFFh |
| | SA14 | 000111xxx | 64/32 | 070000h–07FFFFh | 038000h–03FFFFh |
| Bank 2 | SA15 | 001000xxx | 64/32 | 080000h–08FFFFh | 040000h–047FFFh |
| | SA16 | 001001xxx | 64/32 | 090000h–09FFFFh | 048000h–04FFFFh |
| | SA17 | 001010xxx | 64/32 | 0A0000h–0AFFFFh | 050000h–057FFFh |
| | SA18 | 001011xxx | 64/32 | 0B0000h–0BFFFFh | 058000h–05FFFFh |
| | SA19 | 001100xxx | 64/32 | 0C0000h–0CFFFFh | 060000h–067FFFh |
| | SA20 | 001101xxx | 64/32 | 0D0000h–0DFFFFh | 068000h–06FFFFh |
| | SA21 | 001110xxx | 64/32 | 0E0000h–0EFFFFh | 070000h–077FFFh |
| | SA22 | 001111xxx | 64/32 | 0F0000h–0FFFFFh | 078000h–07FFFFh |
| | SA23 | 010000xxx | 64/32 | 100000h–10FFFFh | 080000h–087FFFh |
| | SA24 | 010001xxx | 64/32 | 110000h–11FFFFh | 088000h–08FFFFh |
| | SA25 | 010010xxx | 64/32 | 120000h–12FFFFh | 090000h–097FFFh |
| | SA26 | 010011xxx | 64/32 | 130000h–13FFFFh | 098000h–09FFFFh |
| | SA27 | 010100xxx | 64/32 | 140000h–14FFFFh | 0A0000h–0A7FFFh |
| | SA28 | 010101xxx | 64/32 | 150000h–15FFFFh | 0A8000h–0AFFFFh |
| | SA29 | 010110xxx | 64/32 | 160000h–16FFFFh | 0B0000h–0B7FFFh |
| | SA30 | 010111xxx | 64/32 | 170000h–17FFFFh | 0B8000h–0BFFFFh |
| | SA31 | 011000xxx | 64/32 | 180000h–18FFFFh | 0C0000h–0C7FFFh |
| | SA32 | 011001xxx | 64/32 | 190000h–19FFFFh | 0C8000h–0CFFFFh |
| | SA33 | 011010xxx | 64/32 | 1A0000h–1AFFFFh | 0D0000h–0D7FFFh |
| | SA34 | 011011xxx | 64/32 | 1B0000h–1BFFFFh | 0D8000h–0DFFFFh |
| | SA35 | 011100xxx | 64/32 | 1C0000h–1CFFFFh | 0E0000h–0E7FFFh |
| | SA36 | 011101xxx | 64/32 | 1D0000h–1DFFFFh | 0E8000h–0EFFFFh |
| | SA37 | 011110xxx | 64/32 | 1E0000h–1EFFFFh | 0F0000h–0F7FFFh |
| | SA38 | 011111xxx | 64/32 | 1F0000h–1FFFFFh | 0F8000h–0FFFFFh |

**Table 7. Bottom Boot Sector Addresses (Continued)**

| | Sector | Sector Address A20–A12 | Sector Size (Kbytes/Kwords) | (x8) Address Range | (x16) Address Range |
|---|---|---|---|---|---|
| Bank 3 | SA39 | 100000xxx | 64/32 | 200000h–20FFFFh | 100000h–107FFFh |
| | SA40 | 100001xxx | 64/32 | 210000h–21FFFFh | 108000h–10FFFFh |
| | SA41 | 100010xxx | 64/32 | 220000h–22FFFFh | 110000h–117FFFh |
| | SA42 | 100011xxx | 64/32 | 230000h–23FFFFh | 118000h–11FFFFh |
| | SA43 | 100100xxx | 64/32 | 240000h–24FFFFh | 120000h–127FFFh |
| | SA44 | 100101xxx | 64/32 | 250000h–25FFFFh | 128000h–12FFFFh |
| | SA45 | 100110xxx | 64/32 | 260000h–26FFFFh | 130000h–137FFFh |
| | SA46 | 100111xxx | 64/32 | 270000h–27FFFFh | 138000h–13FFFFh |
| | SA47 | 101000xxx | 64/32 | 280000h–28FFFFh | 140000h–147FFFh |
| | SA48 | 101001xxx | 64/32 | 290000h–29FFFFh | 148000h–14FFFFh |
| | SA49 | 101010xxx | 64/32 | 2A0000h–2AFFFFh | 150000h–157FFFh |
| | SA50 | 101011xxx | 64/32 | 2B0000h–2BFFFFh | 158000h–15FFFFh |
| | SA51 | 101100xxx | 64/32 | 2C0000h–2CFFFFh | 160000h–167FFFh |
| | SA52 | 101101xxx | 64/32 | 2D0000h–2DFFFFh | 168000h–16FFFFh |
| | SA53 | 101110xxx | 64/32 | 2E0000h–2EFFFFh | 170000h–177FFFh |
| | SA54 | 101111xxx | 64/32 | 2F0000h–2FFFFFh | 178000h–17FFFFh |
| | SA55 | 111000xxx | 64/32 | 300000h–30FFFFh | 180000h–187FFFh |
| | SA56 | 110001xxx | 64/32 | 310000h–31FFFFh | 188000h–18FFFFh |
| | SA57 | 110010xxx | 64/32 | 320000h–32FFFFh | 190000h–197FFFh |
| | SA58 | 110011xxx | 64/32 | 330000h–33FFFFh | 198000h–19FFFFh |
| | SA59 | 110100xxx | 64/32 | 340000h–34FFFFh | 1A0000h–1A7FFFh |
| | SA60 | 110101xxx | 64/32 | 350000h–35FFFFh | 1A8000h–1AFFFFh |
| | SA61 | 110110xxx | 64/32 | 360000h–36FFFFh | 1B0000h–1B7FFFh |
| | SA62 | 110111xxx | 64/32 | 370000h–37FFFFh | 1B8000h–1BFFFFh |
| Bank 4 | SA63 | 111000xxx | 64/32 | 380000h–38FFFFh | 1C0000h–1C7FFFh |
| | SA64 | 111001xxx | 64/32 | 390000h–39FFFFh | 1C8000h–1CFFFFh |
| | SA65 | 111010xxx | 64/32 | 3A0000h–3AFFFFh | 1D0000h–1D7FFFh |
| | SA66 | 111011xxx | 64/32 | 3B0000h–3BFFFFh | 1D8000h–1DFFFFh |
| | SA67 | 111100xxx | 64/32 | 3C0000h–3CFFFFh | 1E0000h–1E7FFFh |
| | SA68 | 111101xxx | 64/32 | 3D0000h–3DFFFFh | 1E8000h–1EFFFFh |
| | SA69 | 111110xxx | 64/32 | 3E0000h–3EFFFFh | 1F0000h–1F7FFFh |
| | SA70 | 111111xxx | 64/32 | 3F0000h–3FFFFFh | 1F8000h–1FFFFFh |

***Note:*** *The address range is A20:A-1 in byte mode (BYTE#=$V_{IL}$) or A20:A0 in word mode (BYTE#=$V_{IH}$). The bank address bits are A20–A18.*

**Table 8. Bottom Boot SecSi™ Sector Addresses**

| Device | Sector Address A20–A12 | Sector Size (Bytes/Words) | (x8) Address Range | (x16) Address Range |
|---|---|---|---|---|
| Am29DL320GB | 000000xxx | 256/128 | 000000h–0000FFh | 00000h–00007Fh |

## Sector/Sector Block Protection and Unprotection

(Note: For the following discussion, the term “sector” applies to both sectors and sector blocks. A sector block consists of two or more adjacent sectors that are protected or unprotected at the same time (see Tables 9 and 10).

The hardware sector protection feature disables both program and erase operations in any sector. The hardware sector unprotection feature re-enables both program and erase operations in previously protected sectors. Sector protection/unprotection can be implemented via two methods.

**Table 9. Top Boot Sector/Sector Block Addresses for Protection/Unprotection**

| Sector | A20–A12 | Sector/ Sector Block Size |
|---|---|---|
| SA0 | 000000XXX | 64 Kbytes |
| SA1–SA3 | 000001XXX, 000010XXX 000011XXX | 192 (3x64) Kbytes |
| SA4–SA7 | 0001XXXXX | 256 (4x64) Kbytes |
| SA8–SA11 | 0010XXXXX | 256 (4x64) Kbytes |
| SA12–SA15 | 0011XXXXX | 256 (4x64) Kbytes |
| SA16–SA19 | 0100XXXXX | 256 (4x64) Kbytes |
| SA20–SA23 | 0101XXXXX | 256 (4x64) Kbytes |
| SA24–SA27 | 0110XXXXX | 256 (4x64) Kbytes |
| SA28–SA31 | 0111XXXXX | 256 (4x64) Kbytes |
| SA32–SA35 | 1000XXXXX | 256 (4x64) Kbytes |
| SA36–SA39 | 1001XXXXX | 256 (4x64) Kbytes |
| SA40–SA43 | 1010XXXXX | 256 (4x64) Kbytes |
| SA44–SA47 | 1011XXXXX | 256 (4x64) Kbytes |
| SA48–SA51 | 1100XXXXX | 256 (4x64) Kbytes |
| SA52–SA55 | 1101XXXXX | 256 (4x64) Kbytes |
| SA56–SA59 | 1110XXXXX | 256 (4x64) Kbytes |
| SA60–SA62 | 111100XXX, 111101XXX, 111110XXX | 192 (4x64) Kbytes |
| SA63 | 111111000 | 8 Kbytes |
| SA64 | 111111001 | 8 Kbytes |
| SA65 | 111111010 | 8 Kbytes |
| SA66 | 111111011 | 8 Kbytes |
| SA67 | 111111100 | 8 Kbytes |
| SA68 | 111111101 | 8 Kbytes |
| SA69 | 111111110 | 8 Kbytes |
| SA70 | 111111111 | 8 Kbytes |

**Table 10. Bottom Boot Sector/Sector Block Addresses for Protection/Unprotection**

| Sector | A20–A12 | Sector/Sector Block Size |
|---|---|---|
| SA70 | 111111XXX | 64 Kbytes |
| SA69-SA67 | 111110XXX, 111101XXX, 111100XXX | 192 (3x64) Kbytes |
| SA66-SA63 | 1110XXXXX | 256 (4x64) Kbytes |
| SA62-SA59 | 1101XXXXX | 256 (4x64) Kbytes |
| SA58-SA55 | 1100XXXXX | 256 (4x64) Kbytes |
| SA54-SA51 | 1011XXXXX | 256 (4x64) Kbytes |
| SA50-SA47 | 1010XXXXX | 256 (4x64) Kbytes |
| SA46-SA43 | 1001XXXXX | 256 (4x64) Kbytes |
| SA42-SA39 | 1000XXXXX | 256 (4x64) Kbytes |
| SA38-SA35 | 0111XXXXX | 256 (4x64) Kbytes |
| SA34-SA31 | 0110XXXXX | 256 (4x64) Kbytes |
| SA30-SA27 | 0101XXXXX | 256 (4x64) Kbytes |
| SA26-SA23 | 0100XXXXX | 256 (4x64) Kbytes |
| SA22–SA19 | 0011XXXXX | 256 (4x64) Kbytes |
| SA18-SA15 | 0010XXXXX | 256 (4x64) Kbytes |
| SA14-SA11 | 0001XXXXX | 256 (4x64) Kbytes |
| SA10-SA8 | 000011XXX, 000010XXX, 000001XXX | 192 (3x64) Kbytes |
| SA7 | 000000111 | 8 Kbytes |
| SA6 | 000000110 | 8 Kbytes |
| SA5 | 000000101 | 8 Kbytes |
| SA4 | 000000100 | 8 Kbytes |
| SA3 | 000000011 | 8 Kbytes |
| SA2 | 000000010 | 8 Kbytes |
| SA1 | 000000001 | 8 Kbytes |
| SA0 | 000000000 | 8 Kbytes |

The hardware sector protection feature disables both program and erase operations in any sector. The hardware sector unprotection feature re-enables both program and erase operations in previously protected sectors. Note that the sector unprotect algorithm unprotects all sectors in parallel. All previously protected sectors must be individually re-protected. To change data in protected sectors efficiently, the temporary sector un protect function is available. See “Temporary Sector Unprotect”.

Sector Protection/Unprotection requires $V_{ID}$ on the RESET# pin only, and can be implemented either in-system or via programming equipment. Figure 2 shows the algorithms and Figure 27 shows the timing diagram. This method uses standard microprocessor bus cycle timing. For sector unprotect, all unprotected sectors must first be protected prior to the first sector unprotect write cycle. Note that the sector unprotect algorithm unprotects all sectors in parallel. All previ-

ously protected sectors must be individually re-protected. To change data in protected sectors efficiently, the temporary sector unprotect function is available. See "Temporary Sector Unprotect".

The device is shipped with all sectors unprotected. AMD offers the option of programming and protecting sectors at its factory prior to shipping the device through AMD's ExpressFlash™ Service. Contact an AMD representative for details.

It is possible to determine whether a sector is protected or unprotected. See the Sector/Sector Block Protection and Unprotection section for details.

## Write Protect (WP#)

The Write Protect function provides a hardware method of protecting certain boot sectors without using $V_{ID}$. This function is one of two provided by the WP#/ACC pin.

If the system asserts $V_{IL}$ on the WP#/ACC pin, the device disables program and erase functions in the two "outermost" 8 Kbyte boot sectors independently of whether those sectors were protected or unprotected using the method described in "Sector/Sector Block Protection and Unprotection". The two outermost 8 Kbyte boot sectors are the two sectors containing the lowest addresses in a top-boot-configured device, or the two sectors containing the highest addresses in a top-boot-configured device.

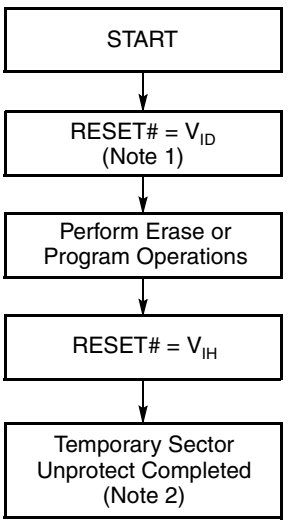
If the system asserts $V_{IH}$ on the WP#/ACC pin, the device reverts to whether the two outermost 8 Kbyte boot sectors were last set to be protected or unprotected. That is, sector protection or unprotection for these two sectors depends on whether they were last protected or unprotected using the method described in "Sector/Sector Block Protection and Unprotection".

Note that the WP#/ACC pin must not be left floating or unconnected; inconsistent behavior of the device may result.

## Temporary Sector Unprotect

(Note: For the following discussion, the term "sector" applies to both sectors and sector blocks. A sector block consists of two or more adjacent sectors that are protected or unprotected at the same time (see Table 9).

This feature allows temporary unprotection of previously protected sectors to change data in-system. The Sector Unprotect mode is activated by setting the RESET# pin to $V_{ID}$. During this mode, formerly protected sectors can be programmed or erased by selecting the sector addresses. Once $V_{ID}$ is removed from the RESET# pin, all the previously protected sectors are protected again. Figure 1 shows the algorithm, and Figure 26 shows the timing diagrams, for this feature. If the WP#/ACC pin is at $V_{IL}$, sectors 0, 1, 69, and 70 will remain protected during the Temporary sector Unprotect mode.




***Notes:***

1. *All protected sectors unprotected (If WP#/ACC = $V_{IL}$, sectors 0 and 1 (bottom boot) or 69 and 70 (top boot) will remain protected).*
2. *All previously protected sectors are protected once again.*

**Figure 1. Temporary Sector Unprotect Operation**

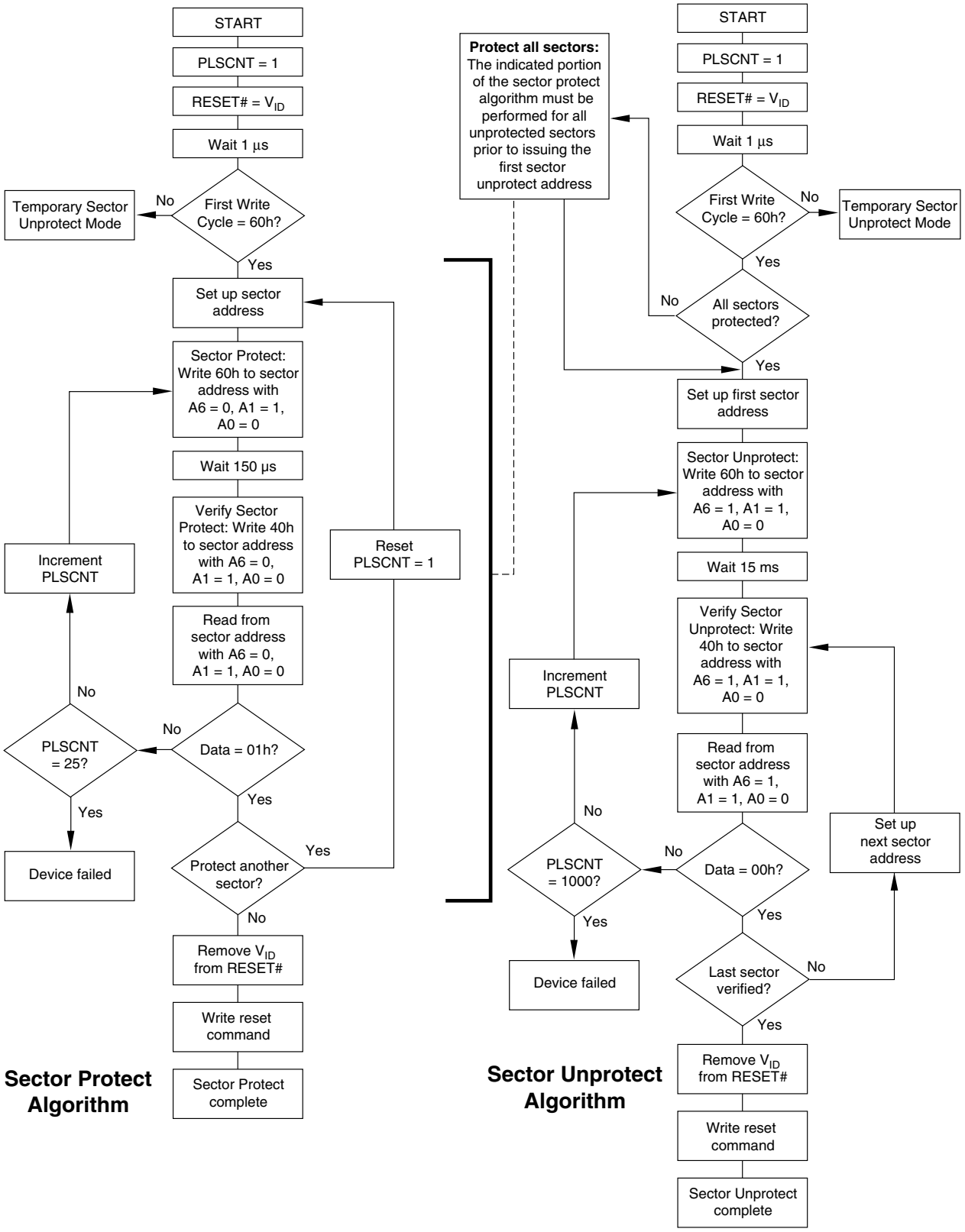



**Figure 2. In-System Sector Protect/Unprotect Algorithms**

## SecSi™ (Secured Silicon) Sector Flash Memory Region

The SecSi (Secured Silicon) Sector feature provides a Flash memory region that enables permanent part identification through an Electronic Serial Number (ESN). The SecSi Sector is 256 bytes in length, and uses a SecSi Sector Indicator Bit (DQ7) to indicate whether or not the SecSi Sector is locked when shipped from the factory. This bit is permanently set at the factory and cannot be changed, which prevents cloning of a factory locked part. This ensures the security of the ESN once the product is shipped to the field.

AMD offers the device with the SecSi Sector either factory locked or customer lockable. The factory-locked version is always protected when shipped from the factory, and has the SecSi (Secured Silicon) Sector Indicator Bit permanently set to a "1." The customer-lockable version is shipped with the SecSi Sector unprotected, allowing customers to utilize the that sector in any manner they choose. The customer-lockable version has the SecSi (Secured Silicon) Sector Indicator Bit permanently set to a "0." Thus, the SecSi Sector Indicator Bit prevents customer-lockable devices from being used to replace devices that are factory locked.

The system accesses the SecSi Sector Secure through a command sequence (see "Enter SecSi™ Sector/Exit SecSi Sector Command Sequence"). After the system has written the Enter SecSi Sector command sequence, it may read the SecSi Sector by using the addresses normally occupied by the boot sectors. This mode of operation continues until the system issues the Exit SecSi Sector command sequence, or until power is removed from the device. On power-up, or following a hardware reset, the device reverts to sending commands to the first 256 bytes of Sector 0.

### Factory Locked: SecSi Sector Programmed and Protected At the Factory

In a factory locked device, the SecSi Sector is protected when the device is shipped from the factory. The SecSi Sector cannot be modified in any way. The device is preprogrammed with both a random number and a secure ESN. The 8-word random number will at addresses 000000h–000007h in word mode (or 000000h–00000Fh in byte mode). The secure ESN will be programmed in the next 8 words at addresses 000008h–00000Fh (or 000010h–000020h in byte mode). The device is available preprogrammed with one of the following:

- A random, secure ESN only
- Customer code through the ExpressFlash service
- Both a random, secure ESN and customer code through the ExpressFlash service.

Customers may opt to have their code programmed by AMD through the AMD ExpressFlash service. AMD programs the customer's code, with or without the random ESN. The devices are then shipped from AMD's factory with the SecSi Sector permanently locked. Contact an AMD representative for details on using AMD's ExpressFlash service.

### Customer Lockable: SecSi Sector NOT Programmed or Protected At the Factory

If the security feature is not required, the SecSi Sector can be treated as an additional Flash memory space. The SecSi Sector can be read any number of times, but can be programmed and locked only once. Note that the accelerated programming (ACC) and unlock bypass functions are not available when programming the SecSi Sector.

The SecSi Sector area can be protected using one of the following procedures:

- Write the three-cycle Enter SecSi Sector Region command sequence, and then follow the in-system sector protect algorithm as shown in Figure 2, except that *RESET# may be at either $V_{IH}$ or $V_{ID}$*. This allows in-system protection of the SecSi Sector Region without raising any device pin to a high voltage. Note that this method is only applicable to the SecSi Sector.
- To verify the protect/unprotect status of the SecSi Sector, follow the algorithm shown in Figure 3.

Once the SecSi Sector is locked and verified, the system must write the Exit SecSi Sector Region command sequence to return to reading and writing the remainder of the array.

The SecSi Sector lock must be used with caution since, once locked, there is no procedure available for unlocking the SecSi Sector area and none of the bits in the SecSi Sector memory space can be modified in any way.

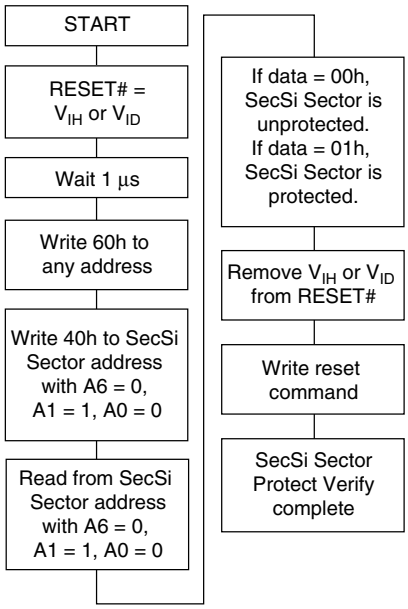



**Figure 3. SecSi Sector Protect Verify**

## Hardware Data Protection

The command sequence requirement of unlock cycles for programming or erasing provides data protection against inadvertent writes (refer to Tables 15 and 16 for command definitions). In addition, the following hardware data protection measures prevent accidental erasure or programming, which might otherwise be caused by spurious system level signals during $V_{CC}$ power-up and power-down transitions, or from system noise.

### Low $V_{CC}$ Write Inhibit

When $V_{CC}$ is less than $V_{LKO}$, the device does not accept any write cycles. This protects data during $V_{CC}$ power-up and power-down. The command register and all internal program/erase circuits are disabled, and the device resets to the read mode. Subsequent writes are ignored until $V_{CC}$ is greater than $V_{LKO}$. The system must provide the proper signals to the control pins to prevent unintentional writes when $V_{CC}$ is greater than $V_{LKO}$.

### Write Pulse “Glitch” Protection

Noise pulses of less than 5 ns (typical) on OE#, CE#f or WE# do not initiate a write cycle.

### Logical Inhibit

Write cycles are inhibited by holding any one of OE# = $V_{IL}$, CE#f = $V_{IH}$ or WE# = $V_{IH}$. To initiate a write cycle, CE#f and WE# must be a logical zero while OE# is a logical one.

### Power-Up Write Inhibit

If WE# = CE#f = $V_{IL}$ and OE# = $V_{IH}$ during power up, the device does not accept commands on the rising edge of WE#. The internal state machine is automatically reset to the read mode on power-up.

## COMMON FLASH MEMORY INTERFACE (CFI)

The Common Flash Interface (CFI) specification outlines device and host system software interrogation handshake, which allows specific vendor-specified software algorithms to be used for entire families of devices. Software support can then be device-independent, JEDEC ID-independent, and forward- and backward-compatible for the specified flash device families. Flash vendors can standardize their existing interfaces for long-term compatibility.

This device enters the CFI Query mode when the system writes the CFI Query command, 98h, to address 55h in word mode (or address AAh in byte mode), any time the device is ready to read array data. The system can read CFI information at the addresses given in Tables 11–14. To terminate reading CFI data, the system must write the reset command.The CFI Query mode is not accessible when the device is executing an Embedded Program or embedded Erase algorithm.

The system can also write the CFI query command when the device is in the autoselect mode. The device enters the CFI query mode, and the system can read CFI data at the addresses given in Tables 11–14. The system must write the reset command to return the device to reading array data.

For further information, please refer to the CFI Specification and CFI Publication 100, available via the World Wide Web at http://www.amd.com/flash/cfi. Alternatively, contact an AMD representative for copies of these documents.

**Table 11. CFI Query Identification String**

| Addresses (Word Mode) | Addresses (Byte Mode) | Data | Description |
|---|---|---|---|
| 10h<br>11h<br>12h | 20h<br>22h<br>24h | 0051h<br>0052h<br>0059h | Query Unique ASCII string “QRY” |
| 13h<br>14h | 26h<br>28h | 0002h<br>0000h | Primary OEM Command Set |
| 15h<br>16h | 2Ah<br>2Ch | 0040h<br>0000h | Address for Primary Extended Table |
| 17h<br>18h | 2Eh<br>30h | 0000h<br>0000h | Alternate OEM Command Set (00h = none exists) |
| 19h<br>1Ah | 32h<br>34h | 0000h<br>0000h | Address for Alternate OEM Extended Table (00h = none exists) |

**Table 12. System Interface String**

| Addresses (Word Mode) | Addresses (Byte Mode) | Data | Description |
|---|---|---|---|
| 1Bh | 36h | 0027h | $V_{CC}$ Min. (write/erase)<br>D7–D4: volt, D3–D0: 100 millivolt |
| 1Ch | 38h | 0036h | $V_{CC}$ Max. (write/erase)<br>D7–D4: volt, D3–D0: 100 millivolt |
| 1Dh | 3Ah | 0000h | $V_{PP}$ Min. voltage (00h = no $V_{PP}$ pin present) |
| 1Eh | 3Ch | 0000h | $V_{PP}$ Max. voltage (00h = no $V_{PP}$ pin present) |
| 1Fh | 3Eh | 0004h | Typical timeout per single byte/word write $2^N$ µs |
| 20h | 40h | 0000h | Typical timeout for Min. size buffer write $2^N$ µs (00h = not supported) |
| 21h | 42h | 000Ah | Typical timeout per individual block erase $2^N$ ms |
| 22h | 44h | 0000h | Typical timeout for full chip erase $2^N$ ms (00h = not supported) |
| 23h | 46h | 0005h | Max. timeout for byte/word write $2^N$ times typical |
| 24h | 48h | 0000h | Max. timeout for buffer write $2^N$ times typical |
| 25h | 4Ah | 0004h | Max. timeout per individual block erase $2^N$ times typical |
| 26h | 4Ch | 0000h | Max. timeout for full chip erase $2^N$ times typical (00h = not supported) |

**Table 13. Device Geometry Definition**

| Addresses (Word Mode) | Addresses (Byte Mode) | Data | Description |
|---|---|---|---|
| 27h | 4Eh | 0016h | Device Size = $2^N$ byte |
| 28h<br>29h | 50h<br>52h | 0002h<br>0000h | Flash Device Interface description (refer to CFI publication 100) |
| 2Ah<br>2Bh | 54h<br>56h | 0000h<br>0000h | Max. number of byte in multi-byte write = $2^N$<br>(00h = not supported) |
| 2Ch | 58h | 0003h | Number of Erase Block Regions within device |
| 2Dh<br>2Eh<br>2Fh<br>30h | 5Ah<br>5Ch<br>5Eh<br>60h | 0007h<br>0000h<br>0020h<br>0000h | Erase Block Region 1 Information<br>(refer to the CFI specification or CFI publication 100) |
| 31h<br>32h<br>33h<br>34h | 62h<br>64h<br>66h<br>68h | 003Eh<br>0000h<br>0000h<br>0001h | Erase Block Region 2 Information<br>(refer to the CFI specification or CFI publication 100) |
| 35h<br>36h<br>37h<br>38h | 6Ah<br>6Ch<br>6Eh<br>70h | 0000h<br>0000h<br>0000h<br>0000h | Erase Block Region 3 Information<br>(refer to the CFI specification or CFI publication 100) |
| 39h<br>3Ah<br>3Bh<br>3Ch | 72h<br>74h<br>76h<br>78h | 0000h<br>0000h<br>0000h<br>0000h | Erase Block Region 4 Information<br>(refer to the CFI specification or CFI publication 100) |

**Table 14. Primary Vendor-Specific Extended Query**

| Addresses (Word Mode) | Addresses (Byte Mode) | Data | Description |
|---|---|---|---|
| 40h<br>41h<br>42h | 80h<br>82h<br>84h | 0050h<br>0052h<br>0049h | Query-unique ASCII string “PRI” |
| 43h | 86h | 0031h | Major version number, ASCII (reflects modifications to the silicon) |
| 44h | 88h | 0033h | Minor version number, ASCII (reflects modifications to the CFI table) |
| 45h | 8Ah | 0004h | Address Sensitive Unlock (Bits 1-0)<br>0 = Required, 1 = Not Required<br>Silicon Revision Number (Bits 7-2) |
| 46h | 8Ch | 0002h | Erase Suspend<br>0 = Not Supported, 1 = To Read Only, 2 = To Read & Write |
| 47h | 8Eh | 0001h | Sector Protect<br>0 = Not Supported, X = Number of sectors in per group |
| 48h | 90h | 0001h | Sector Temporary Unprotect<br>00 = Not Supported, 01 = Supported |
| 49h | 92h | 0004h | Sector Protect/Unprotect scheme<br>01 =29F040 mode, 02 = 29F016 mode, 03 = 29F400, 04 = 29LV800 mode |
| 4Ah | 94h | 0038h | Simultaneous Operation<br>00 = Not Supported, X = Number of Sectors (excluding Bank 1) |
| 4Bh | 96h | 0000h | Burst Mode Type<br>00 = Not Supported, 01 = Supported |
| 4Ch | 98h | 0000h | Page Mode Type<br>00 = Not Supported, 01 = 4 Word Page, 02 = 8 Word Page |
| 4Dh | 9Ah | 0085h | ACC (Acceleration) Supply Minimum<br>00h = Not Supported, D7-D4: Volt, D3-D0: 100 mV |
| 4Eh | 9Ch | 0095h | ACC (Acceleration) Supply Maximum<br>00h = Not Supported, D7-D4: Volt, D3-D0: 100 mV |
| 4Fh | 9Eh | 000xh | Top/Bottom Boot Sector Flag<br>02h = Bottom Boot Device, 03h = Top Boot Device |

## FLASH COMMAND DEFINITIONS

Writing specific address and data commands or sequences into the command register initiates device operations. Tables 15 and 16 define the valid register command sequences. *Writing incorrect address and data values or writing them in the improper sequence may place the device in an unknown state.* A reset command is then required to return the device to reading array data.

All addresses are latched on the falling edge of WE# or CE#f, whichever happens later. All data is latched on the rising edge of WE# or CE#f, whichever happens first. Refer to the AC Characteristics section for timing diagrams.

### Reading Array Data

The device is automatically set to reading array data after device power-up. No commands are required to retrieve data. Each bank is ready to read array data after completing an Embedded Program or Embedded Erase algorithm.

After the device accepts an Erase Suspend command, the corresponding bank enters the erase-suspend-read mode, after which the system can read data from any non-erase-suspended sector within the same bank. The system can read array data using the standard read timing, except that if it reads at an address within erase-suspended sectors, the device outputs status data. After completing a programming operation in the Erase Suspend mode, the system may once again read array data with the same exception. See the Erase Suspend/Erase Resume Commands section for more information.

The system *must* issue the reset command to return a bank to the read (or erase-suspend-read) mode if DQ5 goes high during an active program or erase operation, or if the bank is in the autoselect mode. See the next section, Reset Command, for more information.

See also Requirements for Reading Array Data in the section for more information. The Read-Only Operations table provides the read parameters, and Figure 15 shows the timing diagram.

### Reset Command

Writing the reset command resets the banks to the read or erase-suspend-read mode. Address bits are don't cares for this command.

The reset command may be written between the sequence cycles in an erase command sequence before erasing begins. This resets the bank to which the system was writing to the read mode. Once erasure begins, however, the device ignores reset commands until the operation is complete.

The reset command may be written between the sequence cycles in a program command sequence before programming begins. This resets the bank to which the system was writing to the read mode. If the program command sequence is written to a bank that is in the Erase Suspend mode, writing the reset command returns that bank to the erase-suspend-read mode. Once programming begins, however, the device ignores reset commands until the operation is complete.

The reset command may be written between the sequence cycles in an autoselect command sequence. Once in the autoselect mode, the reset command must be written to return to the read mode. If a bank entered the autoselect mode while in the Erase Suspend mode, writing the reset command returns that bank to the erase-suspend-read mode.

If DQ5 goes high during a program or erase operation, writing the reset command returns the banks to the read mode (or erase-suspend-read mode if that bank was in Erase Suspend).

### Autoselect Command Sequence

The autoselect command sequence allows the host system to access the manufacturer and device codes, and determine whether or not a sector is protected. The autoselect command sequence may be written to an address within a bank that is either in the read or erase-suspend-read mode. The autoselect command may not be written while the device is actively programming or erasing in the other bank.

The autoselect command sequence is initiated by first writing two unlock cycles. This is followed by a third write cycle that contains the bank address and the autoselect command. The bank then enters the autoselect mode. The system may read any number of autoselect codes without reinitiating the command sequence.

Tables 15 and 16 show the address and data requirements. To determine sector protection information, the system must write to the appropriate bank address (BA) and sector address (SADD). Tables 5 and 7 show the address range and bank number associated with each sector.

The system must write the reset command to return to the read mode (or erase-suspend-read mode if the bank was previously in Erase Suspend).

### Enter SecSi™ Sector/Exit SecSi Sector Command Sequence

The SecSi Sector region provides a secured data area containing a random, sixteen-byte electronic serial number (ESN). The system can access the SecSi

Sector region by issuing the three-cycle Enter SecSi Sector command sequence. The device continues to access the SecSi Sector region until the system issues the four-cycle Exit SecSi Sector command sequence. The Exit SecSi Sector command sequence returns the device to normal operation. The SecSi Sector is not accessible when the device is executing an Embedded Program or embedded Erase algorithm. Tables 15 and 16 show the address and data requirements for both command sequences. See also "SecSi™ (Secured Silicon) Sector Flash Memory Region" for further information. *Note that the ACC function and unlock bypass modes are not available when the SecSi Sector is enabled.*

## Byte/Word Program Command Sequence

The system may program the device by word or byte, depending on the state of the CIOf pin. Programming is a four-bus-cycle operation. The program command sequence is initiated by writing two unlock write cycles, followed by the program set-up command. The program address and data are written next, which in turn initiate the Embedded Program algorithm. The system is *not* required to provide further controls or timings. The device automatically provides internally generated program pulses and verifies the programmed cell margin. Tables 15 and 16 show the address and data requirements for the byte program command sequence.

When the Embedded Program algorithm is complete, that bank then returns to the read mode and addresses are no longer latched. The system can determine the status of the program operation by using DQ7, DQ6, or RY/BY#. Refer to the Flash Write Operation Status section for information on these status bits.

Any commands written to the device during the Embedded Program Algorithm are ignored. Note that a **hardware reset** immediately terminates the program operation. The program command sequence should be reinitiated once that bank has returned to the read mode, to ensure data integrity. *Note that the SecSi Sector, autoselect, and CFI functions are unavailable when a program operation is in progress.*

Programming is allowed in any sequence and across sector boundaries. **A bit cannot be programmed from "0" back to a "1."** Attempting to do so may cause that bank to set DQ5 = 1, or cause the DQ7 and DQ6 status bits to indicate the operation was successful. However, a succeeding read will show that the data is still "0." Only erase operations can convert a "0" to a "1."

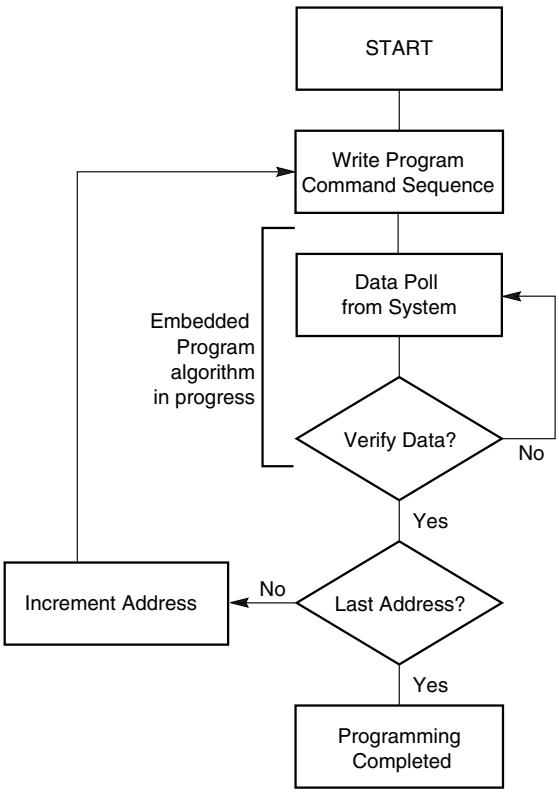
### Unlock Bypass Command Sequence

The unlock bypass feature allows the system to program bytes or words to a bank faster than using the standard program command sequence. The unlock bypass command sequence is initiated by first writing two unlock cycles. This is followed by a third write cycle containing the unlock bypass command, 20h. That bank then enters the unlock bypass mode. A two-cycle unlock bypass program command sequence is all that is required to program in this mode. The first cycle in this sequence contains the unlock bypass program command, A0h; the second cycle contains the program address and data. Additional data is programmed in the same manner. This mode dispenses with the initial two unlock cycles required in the standard program command sequence, resulting in faster total programming time. Tables 15 and 16 show the requirements for the command sequence.

During the unlock bypass mode, only the Unlock Bypass Program and Unlock Bypass Reset commands are valid. To exit the unlock bypass mode, the system must issue the two-cycle unlock bypass reset command sequence. The first cycle must contain the bank address and the data 90h. The second cycle need only contain the data 00h. The bank then returns to the read mode.

The device offers accelerated program operations through the WP#/ACC pin. When the system asserts $V_{HH}$ on the WP#/ACC pin, the device automatically enters the Unlock Bypass mode. The system may then write the two-cycle Unlock Bypass program command sequence. The device uses the higher voltage on the WP#/ACC pin to accelerate the operation. *Note that the WP#/ACC pin must not be at $V_{HH}$ any operation other than accelerated programming, or device damage may result. In addition, the WP#/ACC pin must not be left floating or unconnected; inconsistent behavior of the device may result.*

Figure 4 illustrates the algorithm for the program operation. Refer to the Flash Erase and Program Operations table in the AC Characteristics section for parameters, and Figure 19 for timing diagrams.




***Note:*** *See Tables 15 and 16 for program command sequence.*

**Figure 4. Program Operation**

## Chip Erase Command Sequence

Chip erase is a six bus cycle operation. The chip erase command sequence is initiated by writing two unlock cycles, followed by a set-up command. Two additional unlock write cycles are then followed by the chip erase command, which in turn invokes the Embedded Erase algorithm. The device does *not* require the system to preprogram prior to erase. The Embedded Erase algorithm automatically preprograms and verifies the entire memory for an all zero data pattern prior to electrical erase. The system is not required to provide any controls or timings during these operations. Tables 15 and 16 show the address and data requirements for the chip erase command sequence.

When the Embedded Erase algorithm is complete, that bank returns to the read mode and addresses are no longer latched. The system can determine the status of the erase operation by using DQ7, DQ6, DQ2, or RY/BY#. Refer to the Flash Write Operation Status section for information on these status bits.

Any commands written during the chip erase operation are ignored. However, note that a **hardware reset** immediately terminates the erase operation. If that occurs, the chip erase command sequence should be reinitiated once that bank has returned to reading array data, to ensure data integrity. *Note that the SecSi Sector, autoselect, and CFI functions are unavailable when an erase operation in is progress.*

Figure 5 illustrates the algorithm for the erase operation. Refer to the Flash Erase and Program Operations tables in the AC Characteristics section for parameters, and Figure 21 section for timing diagrams.

## Sector Erase Command Sequence

Sector erase is a six bus cycle operation. The sector erase command sequence is initiated by writing two unlock cycles, followed by a set-up command. Two additional unlock cycles are written, and are then followed by the address of the sector to be erased, and the sector erase command. Tables 15 and 16 show the address and data requirements for the sector erase command sequence.

The device does *not* require the system to preprogram prior to erase. The Embedded Erase algorithm automatically programs and verifies the entire memory for an all zero data pattern prior to electrical erase. The system is not required to provide any controls or timings during these operations.

After the command sequence is written, a sector erase time-out of 80 µs occurs. During the time-out period, additional sector addresses and sector erase commands may be written. Loading the sector erase buffer may be done in any sequence, and the number of sectors may be from one sector to all sectors. The time between these additional cycles must be less than 80 µs, otherwise erasure may begin. Any sector erase address and command following the exceeded time-out may or may not be accepted. It is recommended that processor interrupts be disabled during this time to ensure all commands are accepted. The interrupts can be re-enabled after the last Sector Erase command is written. **Any command other than Sector Erase or Erase Suspend during the time-out period resets that bank to the read mode.** The system must rewrite the command sequence and any additional addresses and commands. *Note that the SecSi Sector, autoselect, and CFI functions are unavailable when an erase operation in is progress.*

The system can monitor DQ3 to determine if the sector erase timer has timed out (See the section on DQ3: Sector Erase Timer.). The time-out begins from the rising edge of the final WE# pulse in the command sequence.

When the Embedded Erase algorithm is complete, the bank returns to reading array data and addresses are

no longer latched. Note that while the Embedded Erase operation is in progress, the system can read data from the non-erasing bank. The system can determine the status of the erase operation by reading DQ7, DQ6, DQ2, or RY/BY# in the erasing bank. Refer to the Flash Write Operation Status section for information on these status bits.

Once the sector erase operation has begun, only the Erase Suspend command is valid. All other commands are ignored. However, note that a **hardware reset** immediately terminates the erase operation. If that occurs, the sector erase command sequence should be reinitiated once that bank has returned to reading array data, to ensure data integrity. *Note that the SecSi Sector, autoselect, and CFI functions are unavailable when an erase operation in is progress.*

Figure 5 illustrates the algorithm for the erase operation. Refer to the Flash Erase and Program Operations tables in the AC Characteristics section for parameters, and Figure 21 section for timing diagrams.

## Erase Suspend/Erase Resume Commands

The Erase Suspend command, B0h, allows the system to interrupt a sector erase operation and then read data from, or program data to, any sector not selected for erasure. The bank address is required when writing this command. This command is valid only during the sector erase operation, including the 80 µs time-out period during the sector erase command sequence. The Erase Suspend command is ignored if written during the chip erase operation or Embedded Program algorithm.

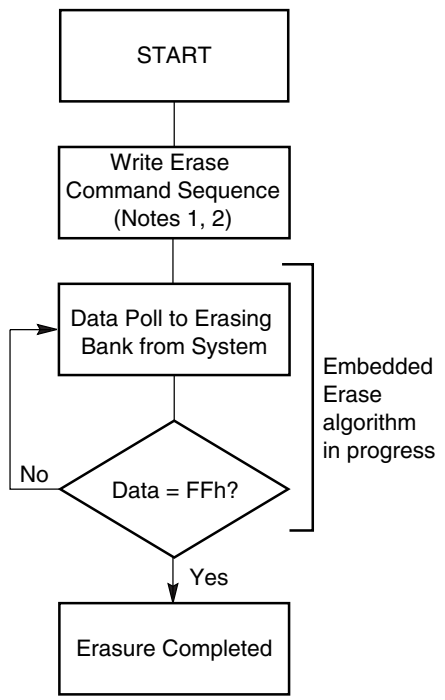
When the Erase Suspend command is written during the sector erase operation, the device requires a maximum of 20 µs to suspend the erase operation. However, when the Erase Suspend command is written during the sector erase time-out, the device immediately terminates the time-out period and suspends the erase operation. Addresses are “don’t-cares” when writing the Erase suspend command.

After the erase operation has been suspended, the bank enters the erase-suspend-read mode. The system can read data from or program data to any sector not selected for erasure. (The device “erase suspends” all sectors selected for erasure.) Reading at any address within erase-suspended sectors produces status information on DQ7–DQ0. The system can use DQ7, or DQ6 and DQ2 together, to determine if a sector is actively erasing or is erase-suspended. Refer to the Flash Write Operation Status section for information on these status bits.

After an erase-suspended program operation is complete, the bank returns to the erase-suspend-read mode. The system can determine the status of the program operation using the DQ7 or DQ6 status bits, just as in the standard Byte Program operation. Refer to the Flash Write Operation Status section for more information.

In the erase-suspend-read mode, the system can also issue the autoselect command sequence. The device allows reading autoselect codes even at addresses within erasing sectors, since the codes are not stored in the memory array. When the device exits the autoselect mode, the device reverts to the Erase Suspend mode, and is ready for another valid operation. Refer to the Sector/Sector Block Protection and Unprotection and Autoselect Command Sequence sections for details.

To resume the sector erase operation, the system must write the Erase Resume command (address bits are don’t care). The bank address of the erase-suspended bank is required when writing this command. Further writes of the Resume command are ignored. Another Erase Suspend command can be written after the chip has resumed erasing.




***Notes:***

1. *See Tables 15 and 16 for erase command sequence.*
2. *See the section on DQ3 for information on the sector erase timer.*

**Figure 5. Erase Operation**

## Table 15. Command Definitions (Flash Word Mode)

| Command Sequence (Note 1) | | Cycles | Bus Cycles (Notes 2–5) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | First | | Second | | Third | | Fourth | | Fifth | | Sixth | |
| | | | Addr | Data | Addr | Data | Addr | Data | Addr | Data | Addr | Data | Addr | Data |
| Read (Note 6) | | 1 | RA | RD | | | | | | | | | | |
| Reset (Note 7) | | 1 | XXX | F0 | | | | | | | | | | |
| Autoselect (Note 8) | Manufacturer ID | 4 | 555 | AA | 2AA | 55 | (BA)555 | 90 | (BA)X00 | 0001 | | | | |
| | Device ID (Note 9) | 4 | 555 | AA | 2AA | 55 | (BA)555 | 90 | (BA)X01 | 7E | (BA) 0E | 0A | (BA) 0F | 0000/ 0001 |
| | SecSi Sector Factory Protect (Note 10) | 4 | 555 | AA | 2AA | 55 | (BA)555 | 90 | (BA)X03 | 0082/0002 | | | | |
| | Sector Protect Verify (Note 11) | 4 | 555 | AA | 2AA | 55 | (BA)555 | 90 | (SADD) X02 | 0000/0001 | | | | |
| Enter SecSi Sector Region | | 3 | 555 | AA | 2AA | 55 | 555 | 88 | | | | | | |
| Exit SecSi Sector Region | | 4 | 555 | AA | 2AA | 55 | 555 | 90 | XXX | 00 | | | | |
| Program | | 4 | 555 | AA | 2AA | 55 | 555 | A0 | PA | PD | | | | |
| Unlock Bypass | | 3 | 555 | AA | 2AA | 55 | 555 | 20 | | | | | | |
| Unlock Bypass Program (Note 12) | | 2 | XXX | A0 | PA | PD | | | | | | | | |
| Unlock Bypass Reset (Note 13) | | 2 | XXX | 90 | XXX | 00 | | | | | | | | |
| Chip Erase | | 6 | 555 | AA | 2AA | 55 | 555 | 80 | 555 | AA | 2AA | 55 | 555 | 10 |
| Sector Erase | | 6 | 555 | AA | 2AA | 55 | 555 | 80 | 555 | AA | 2AA | 55 | SADD | 30 |
| Erase Suspend (Note 14) | | 1 | BA | B0 | | | | | | | | | | |
| Erase Resume (Note 15) | | 1 | BA | 30 | | | | | | | | | | |
| CFI Query (Note 16) | | 1 | 55 | 98 | | | | | | | | | | |

***Legend:***

*X = Don't care*

*RA = Address of the memory location to be read.*

*RD = Data read from location RA during read operation.*

*PA = Address of the memory location to be programmed. Addresses latch on the falling edge of the WE# or CE#f pulse, whichever happens later.*

*PD = Data to be programmed at location PA. Data latches on the rising edge of WE# or CE#f pulse, whichever happens first.*

*SADD = Address of the sector to be verified (in autoselect mode) or erased. Address bits A20–A12 uniquely select any sector.*

*BA = Address of the bank that is being switched to autoselect mode, is in bypass mode, or is being erased.*

***Notes:***

1. *See Table 1 for description of bus operations.*
2. *All values are in hexadecimal.*
3. *Except for the read cycle and the fourth cycle of the autoselect command sequence, all bus cycles are write cycles.*
4. *Data bits DQ15–DQ8 are don't care in command sequences, except for RD and PD.*
5. *Unless otherwise noted, address bits A20–A12 are don't cares.*
6. *No unlock or command cycles required when bank is in read mode.*
7. *The Reset command is required to return to reading array data (or to the erase-suspend-read mode if previously in Erase Suspend) when a bank is in the autoselect mode, or if DQ5 goes high (while the bank is providing status information).*
8. *The fourth cycle of the autoselect command sequence is a read cycle. The system must provide the bank address to obtain the manufacturer ID, device ID, or SecSi Sector factory protect information. See the Autoselect Command Sequence section for more information.*
9. *The device ID must be read across three cycles. The device ID is 01h for top boot and 00h for bottom boot.*
10. *The data is 82h for factory locked and 02h for not factory locked.*
11. *The data is 00h for an unprotected sector/sector block and 01h for a protected sector/sector block.*
12. *The Unlock Bypass command is required prior to the Unlock Bypass Program command.*
13. *The Unlock Bypass Reset command is required to return to reading array data when the bank is in the unlock bypass mode.*
14. *The system may read and program in non-erasing sectors, or enter the autoselect mode, when in the Erase Suspend mode. The Erase Suspend command is valid only during a sector erase operation, and requires the bank address.*
15. *The Erase Resume command is valid only during the Erase Suspend mode, and requires the bank address.*
16. *Command is valid when device is ready to read array data or when device is in autoselect mode.*

## Table 16. Command Definitions (Flash Byte Mode)

| Command Sequence (Note 1) | | Cycles | Bus Cycles (Notes 2–5) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | First | | Second | | Third | | Fourth | | Fifth | | Sixth | |
| | | | Addr | Data | Addr | Data | Addr | Data | Addr | Data | Addr | Data | Addr | Data |
| Read (Note 6) | | 1 | RA | RD | | | | | | | | | | |
| Reset (Note 7) | | 1 | XXX | F0 | | | | | | | | | | |
| Autoselect (Note 8) | Manufacturer ID | 4 | AAA | AA | 555 | 55 | (BA) AAA | 90 | (BA) 00 | 01 | | | | |
| | Device ID (Note 9) | 6 | AAA | AA | 555 | 55 | (BA) AAA | 90 | (BA) 02 | 7E | (BA) 1C | 0A | (BA) 1E | 00/01 |
| | SecSi™ Sector Factory Protect (Note 10) | 4 | AAA | AA | 555 | 55 | (BA) AAA | 90 | (BA) X06 | 82/02 | | | | |
| | Sector Protect Verify (Note 11) | 4 | AAA | AA | 555 | 55 | (BA) AAA | 90 | (SADD) X04 | 00 | | | | |
| | | | | | | | | | | 01 | | | | |
| Enter SecSi Sector Region | | 3 | AAA | AA | 555 | 55 | AAA | 88 | | | | | | |
| Exit SecSi Sector Region | | 4 | AAA | AA | 555 | 55 | AAA | 90 | XXX | 00 | | | | |
| Program | | 4 | AAA | AA | 555 | 55 | AAA | A0 | PA | PD | | | | |
| Unlock Bypass | | 3 | AAA | AA | 555 | 55 | AAA | 20 | | | | | | |
| Unlock Bypass Program (Note 12) | | 2 | XXX | A0 | PA | PD | | | | | | | | |
| Unlock Bypass Reset (Note 13) | | 2 | XXX | 90 | XXX | 00 | | | | | | | | |
| Chip Erase | | 6 | AAA | AA | 555 | 55 | AAA | 80 | AAA | AA | 555 | 55 | AAA | 10 |
| Sector Erase | | 6 | AAA | AA | 555 | 55 | AAA | 80 | AAA | AA | 555 | 55 | SADD | 30 |
| Erase Suspend (Note 14) | | 1 | BA | B0 | | | | | | | | | | |
| Erase Resume (Note 15) | | 1 | BA | 30 | | | | | | | | | | |
| CFI Query (Note 16) | | 1 | 55 | 98 | | | | | | | | | | |

***Legend:***

*X = Don't care*

*RA = Address of the memory location to be read.*

*RD = Data read from location RA during read operation.*

*PA = Address of the memory location to be programmed. Addresses latch on the falling edge of the WE# or CE#f pulse, whichever happens later.*

*PD = Data to be programmed at location PA. Data latches on the rising edge of WE# or CE#f pulse, whichever happens first.*

*SADD = Address of the sector to be verified (in autoselect mode) or erased. Address bits A20–A12 uniquely select any sector.*

*BA = Address of the bank that is being switched to autoselect mode, is in bypass mode, or is being erased.*

***Notes:***

1. *See Table 1 for description of bus operations.*
2. *All values are in hexadecimal.*
3. *Except for the read cycle and the fourth cycle of the autoselect command sequence, all bus cycles are write cycles.*
4. *Data bits DQ15–DQ8 are don't care in command sequences, except for RD and PD.*
5. *Unless otherwise noted, address bits A20–A12 are don't cares.*
6. *No unlock or command cycles required when bank is in read mode.*
7. *The Reset command is required to return to reading array data (or to the erase-suspend-read mode if previously in Erase Suspend) when a bank is in the autoselect mode, or if DQ5 goes high (while the bank is providing status information).*
8. *The fourth cycle of the autoselect command sequence is a read cycle. The system must provide the bank address to obtain the manufacturer ID, device ID, or SecSi Sector factory protect information. Data bits DQ15–DQ8 are don't care. See the Autoselect Command Sequence section for more information.*
9. *The device ID must be read across three cycles. The device ID is 00h for top boot and 01h for bottom boot.*
10. *The data is 80h for factory locked and 00h for not factory locked.*
11. *The data is 00h for an unprotected sector/sector block and 01h for a protected sector/sector block.*
12. *The Unlock Bypass command is required prior to the Unlock Bypass Program command.*
13. *The Unlock Bypass Reset command is required to return to reading array data when the bank is in the unlock bypass mode.*
14. *The system may read and program in non-erasing sectors, or enter the autoselect mode, when in the Erase Suspend mode. The Erase Suspend command is valid only during a sector erase operation, and requires the bank address.*
15. *The Erase Resume command is valid only during the Erase Suspend mode, and requires the bank address.*
16. *Command is valid when device is ready to read array data or when device is in autoselect mode.*

## FLASH WRITE OPERATION STATUS

The device provides several bits to determine the status of a program or erase operation: DQ2, DQ3, DQ5, DQ6, and DQ7. Table 17 and the following subsections describe the function of these bits. DQ7 and DQ6 each offer a method for determining whether a program or erase operation is complete or in progress. The device also provides a hardware-based output signal, RY/BY#, to determine whether an Embedded Program or Erase operation is in progress or has been completed.

### DQ7: Data# Polling

The Data# Polling bit, DQ7, indicates to the host system whether an Embedded Program or Erase algorithm is in progress or completed, or whether a bank is in Erase Suspend. Data# Polling is valid after the rising edge of the final WE# pulse in the command sequence.

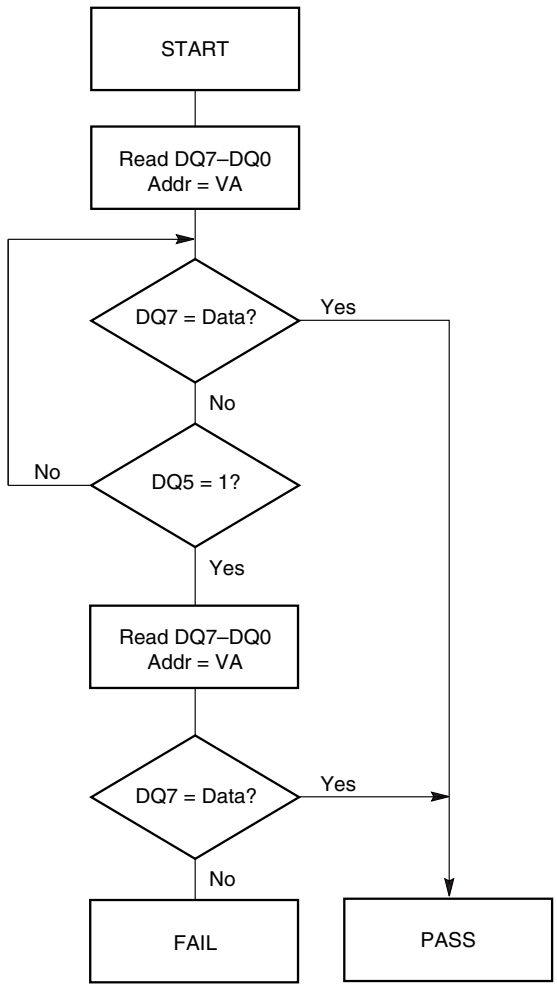
During the Embedded Program algorithm, the device outputs on DQ7 the complement of the datum programmed to DQ7. This DQ7 status also applies to programming during Erase Suspend. When the Embedded Program algorithm is complete, the device outputs the datum programmed to DQ7. The system must provide the program address to read valid status information on DQ7. If a program address falls within a protected sector, Data# Polling on DQ7 is active for approximately 1 µs, then that bank returns to the read mode.

During the Embedded Erase algorithm, Data# Polling produces a “0” on DQ7. When the Embedded Erase algorithm is complete, or if the bank enters the Erase Suspend mode, Data# Polling produces a “1” on DQ7. The system must provide an address within any of the sectors selected for erasure to read valid status information on DQ7.

After an erase command sequence is written, if all sectors selected for erasing are protected, Data# Polling on DQ7 is active for approximately 100 µs, then the bank returns to the read mode. If not all selected sectors are protected, the Embedded Erase algorithm erases the unprotected sectors, and ignores the selected sectors that are protected. However, if the system reads DQ7 at an address within a protected sector, the status may not be valid.

When the system detects DQ7 has changed from the complement to true data, it can read valid data at DQ15–DQ0 (or DQ7–DQ0 for byte mode) on the *following* read cycles. Just prior to the completion of an Embedded Program or Erase operation, DQ7 may change asynchronously with DQ15–DQ8 (DQ7–DQ0 in byte mode) while Output Enable (OE#) is asserted low. That is, the device may change from providing status information to valid data on DQ7. Depending on when the system samples the DQ7 output, it may read the status or valid data. Even if the device has completed the program or erase operation and DQ7 has valid data, the data outputs on DQ15–DQ0 may be still invalid. Valid data on DQ15–DQ0 (or DQ7–DQ0 for byte mode) will appear on successive read cycles.

Table 17 shows the outputs for Data# Polling on DQ7. Figure 6 shows the Data# Polling algorithm. Figure 23 in the AC Characteristics section shows the Data# Polling timing diagram.




***Notes:***

1. *VA = Valid address for programming. During a sector erase operation, a valid address is any sector address within the sector being erased. During chip erase, a valid address is any non-protected sector address.*
2. *DQ7 should be rechecked even if DQ5 = “1” because DQ7 may change simultaneously with DQ5.*

**Figure 6. Data# Polling Algorithm**

## RY/BY#: Ready/Busy#

The RY/BY# is a dedicated, open-drain output pin which indicates whether an Embedded Algorithm is in progress or complete. The RY/BY# status is valid after the rising edge of the final WE# pulse in the command sequence. Since RY/BY# is an open-drain output, several RY/BY# pins can be tied together in parallel with a pull-up resistor to $V_{CC}$.

If the output is low (Busy), the device is actively erasing or programming. (This includes programming in the Erase Suspend mode.) If the output is high (Ready), the device is in the read mode, the standby mode, or one of the banks is in the erase-suspend-read mode.

Table 17 shows the outputs for RY/BY#.

## DQ6: Toggle Bit I

Toggle Bit I on DQ6 indicates whether an Embedded Program or Erase algorithm is in progress or complete, or whether the device has entered the Erase Suspend mode. Toggle Bit I may be read at any address, and is valid after the rising edge of the final WE# pulse in the command sequence (prior to the program or erase operation), and during the sector erase time-out.

During an Embedded Program or Erase algorithm operation, successive read cycles to any address cause DQ6 to toggle. The system may use either OE# or CE#f to control the read cycles. When the operation is complete, DQ6 stops toggling.

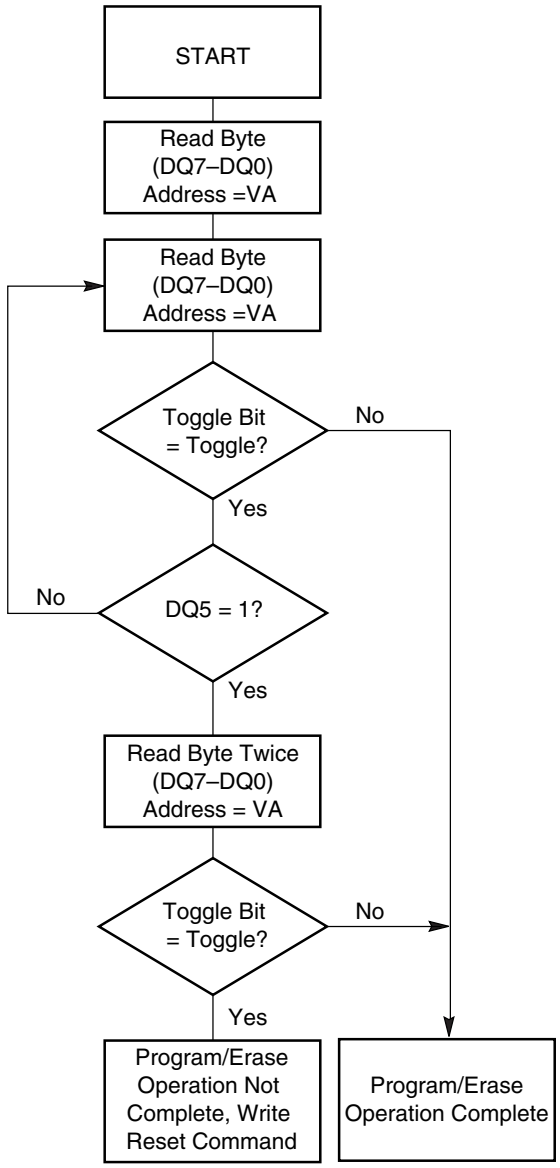
After an erase command sequence is written, if all sectors selected for erasing are protected, DQ6 toggles for approximately 100 µs, then returns to reading array data. If not all selected sectors are protected, the Embedded Erase algorithm erases the unprotected sectors, and ignores the selected sectors that are protected.

The system can use DQ6 and DQ2 together to determine whether a sector is actively erasing or is erase-suspended. When the device is actively erasing (that is, the Embedded Erase algorithm is in progress), DQ6 toggles. When the device enters the Erase Suspend mode, DQ6 stops toggling. However, the system must also use DQ2 to determine which sectors are erasing or erase-suspended. Alternatively, the system can use DQ7 (see the subsection on DQ7: Data# Polling).

If a program address falls within a protected sector, DQ6 toggles for approximately 1 µs after the program command sequence is written, then returns to reading array data.

DQ6 also toggles during the erase-suspend-program mode, and stops toggling once the Embedded Program algorithm is complete.

Table 17 shows the outputs for Toggle Bit I on DQ6. Figure 7 shows the toggle bit algorithm. Figure 24 in the “Flash AC Characteristics” section shows the toggle bit timing diagrams. Figure 25 shows the differences between DQ2 and DQ6 in graphical form. See also the subsection on DQ2: Toggle Bit II.




***Note:*** *The system should recheck the toggle bit even if DQ5 = “1” because the toggle bit may stop toggling as DQ5 changes to “1.” See the subsections on DQ6 and DQ2 for more information.*

**Figure 7. Toggle Bit Algorithm**

## DQ2: Toggle Bit II

The “Toggle Bit II” on DQ2, when used with DQ6, indicates whether a particular sector is actively erasing (that is, the Embedded Erase algorithm is in progress), or whether that sector is erase-suspended. Toggle Bit II is valid after the rising edge of the final WE# pulse in the command sequence.

DQ2 toggles when the system reads at addresses within those sectors that have been selected for erasure. (The system may use either OE# or CE#f to control the read cycles.) But DQ2 cannot distinguish whether the sector is actively erasing or is erase-suspended. DQ6, by comparison, indicates whether the device is actively erasing, or is in Erase Suspend, but cannot distinguish which sectors are selected for erasure. Thus, both status bits are required for sector and mode information. Refer to Table 17 to compare outputs for DQ2 and DQ6.

Figure 7 shows the toggle bit algorithm in flowchart form, and the section “DQ2: Toggle Bit II” explains the algorithm. See also the DQ6: Toggle Bit I subsection. Figure 24 shows the toggle bit timing diagram. Figure 25 shows the differences between DQ2 and DQ6 in graphical form.

## Reading Toggle Bits DQ6/DQ2

Refer to Figure 7 for the following discussion. Whenever the system initially begins reading toggle bit status, it must read DQ15–DQ0 (or DQ7–DQ0 for byte mode) at least twice in a row to determine whether a toggle bit is toggling. Typically, the system would note and store the value of the toggle bit after the first read. After the second read, the system would compare the new value of the toggle bit with the first. If the toggle bit is not toggling, the device has completed the program or erase operation. The system can read array data on DQ15–DQ0 (or DQ7–DQ0 for byte mode) on the following read cycle.

However, if after the initial two read cycles, the system determines that the toggle bit is still toggling, the system also should note whether the value of DQ5 is high (see the section on DQ5). If it is, the system should then determine again whether the toggle bit is toggling, since the toggle bit may have stopped toggling just as DQ5 went high. If the toggle bit is no longer toggling, the device has successfully completed the program or erase operation. If it is still toggling, the device did not completed the operation successfully, and the system must write the reset command to return to reading array data.

The remaining scenario is that the system initially determines that the toggle bit is toggling and DQ5 has not gone high. The system may continue to monitor the toggle bit and DQ5 through successive read cycles, determining the status as described in the previous paragraph. Alternatively, it may choose to perform other system tasks. In this case, the system must start at the beginning of the algorithm when it returns to determine the status of the operation (top of Figure 7).

## DQ5: Exceeded Timing Limits

DQ5 indicates whether the program or erase time has exceeded a specified internal pulse count limit. Under these conditions DQ5 produces a “1,” indicating that the program or erase cycle was not successfully completed.

The device may output a “1” on DQ5 if the system tries to program a “1” to a location that was previously programmed to “0.” **Only an erase operation can change a “0” back to a “1.”** Under this condition, the device halts the operation, and when the timing limit has been exceeded, DQ5 produces a “1.”

Under both these conditions, the system must write the reset command to return to the read mode (or to the erase-suspend-read mode if a bank was previously in the erase-suspend-program mode).

## DQ3: Sector Erase Timer

After writing a sector erase command sequence, the system may read DQ3 to determine whether or not erasure has begun. (The sector erase timer does not apply to the chip erase command.) If additional sectors are selected for erasure, the entire time-out also applies after each additional sector erase command. When the time-out period is complete, DQ3 switches from a “0” to a “1.” If the time between additional sector erase commands from the system can be assumed to be less than 50 µs, the system need not monitor DQ3. See also the Sector Erase Command Sequence section.

After the sector erase command is written, the system should read the status of DQ7 (Data# Polling) or DQ6 (Toggle Bit I) to ensure that the device has accepted the command sequence, and then read DQ3. If DQ3 is “1,” the Embedded Erase algorithm has begun; all further commands (except Erase Suspend) are ignored until the erase operation is complete. If DQ3 is “0,” the device will accept additional sector erase commands. To ensure the command has been accepted, the system software should check the status of DQ3 prior to and following each subsequent sector erase command. If DQ3 is high on the second status check, the last command might not have been accepted.

Table 17 shows the status of DQ3 relative to the other status bits.

**Table 17. Write Operation Status**

| Status | | | DQ7 (Note 2) | DQ6 | DQ5 (Note 1) | DQ3 | DQ2 (Note 2) | RY/BY# |
|---|---|---|---|---|---|---|---|---|
| Standard Mode | Embedded Program Algorithm | | DQ7# | Toggle | 0 | N/A | No toggle | 0 |
| | Embedded Erase Algorithm | | 0 | Toggle | 0 | 1 | Toggle | 0 |
| Erase Suspend Mode | Erase-Suspend-Read | Erase Suspended Sector | 1 | No toggle | 0 | N/A | Toggle | 1 |
| | | Non-Erase Suspended Sector | Data | Data | Data | Data | Data | 1 |
| | Erase-Suspend-Program | | DQ7# | Toggle | 0 | N/A | N/A | 0 |

***Notes:***

1. *DQ5 switches to '1' when an Embedded Program or Embedded Erase operation has exceeded the maximum timing limits. Refer to the section on DQ5 for more information.*
2. *DQ7 and DQ2 require a valid address when reading status information. Refer to the appropriate subsection for further details.*
3. *When reading write operation status bits, the system must always provide the bank address where the Embedded Algorithm is in progress. The device outputs array data if the system addresses a non-busy bank.*

## ABSOLUTE MAXIMUM RATINGS

Storage Temperature
Plastic Packages . . . . . –55°C to +125°C

Ambient Temperature
with Power Applied. . . . . –65°C to +85°C

Voltage with Respect to Ground

$V_{CC}$f/$V_{CC}$s (Note 1) . . . . . –0.5 V to +4.0 V

RESET# (Note 2) . . . . . –0.5 V to +12.5 V

WP#/ACC . . . . . –0.5 V to +10.5 V

All other pins (Note 1) . . . . . –0.5 V to $V_{CC}$ +0.5 V

Output Short Circuit Current (Note 3) . . . . . 200 mA

***Notes:***

1. *Minimum DC voltage on input or I/O pins is –0.5 V. During voltage transitions, input or I/O pins may overshoot $V_{SS}$ to –2.0 V for periods of up to 20 ns. Maximum DC voltage on input or I/O pins is $V_{CC}$ +0.5 V. See Figure 8. During voltage transitions, input or I/O pins may overshoot to $V_{CC}$ +2.0 V for periods up to 20 ns. See Figure 9.*
2. *Minimum DC input voltage on pins RESET#, and WP#/ACC is –0.5 V. During voltage transitions, WP#/ACC, and RESET# may overshoot $V_{SS}$ to –2.0 V for periods of up to 20 ns. See Figure 8. Maximum DC input voltage on pin RESET# is +12.5 V which may overshoot to +14.0 V for periods up to 20 ns. Maximum DC input voltage on WP#/ACC is +9.5 V which may overshoot to +12.0 V for periods up to 20 ns.*
3. *No more than one output may be shorted to ground at a time. Duration of the short circuit should not be greater than one second.*

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

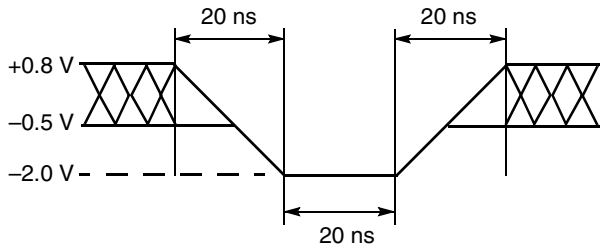



**Figure 8. Maximum Negative Overshoot Waveform**

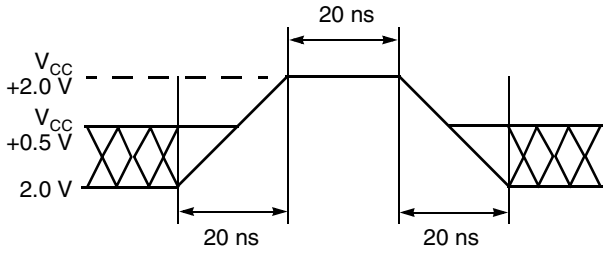



**Figure 9. Maximum Positive Overshoot Waveform**

## OPERATING RANGES

### Industrial (I) Devices

Ambient Temperature ($T_A$) . . . . . –40°C to +85°C

### $V_{CC}$f/$V_{CC}$s Supply Voltages

$V_{CC}$f/$V_{CC}$s for standard voltage range . . 2.7 V to 3.3 V

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

## FLASH DC CHARACTERISTICS

### CMOS Compatible

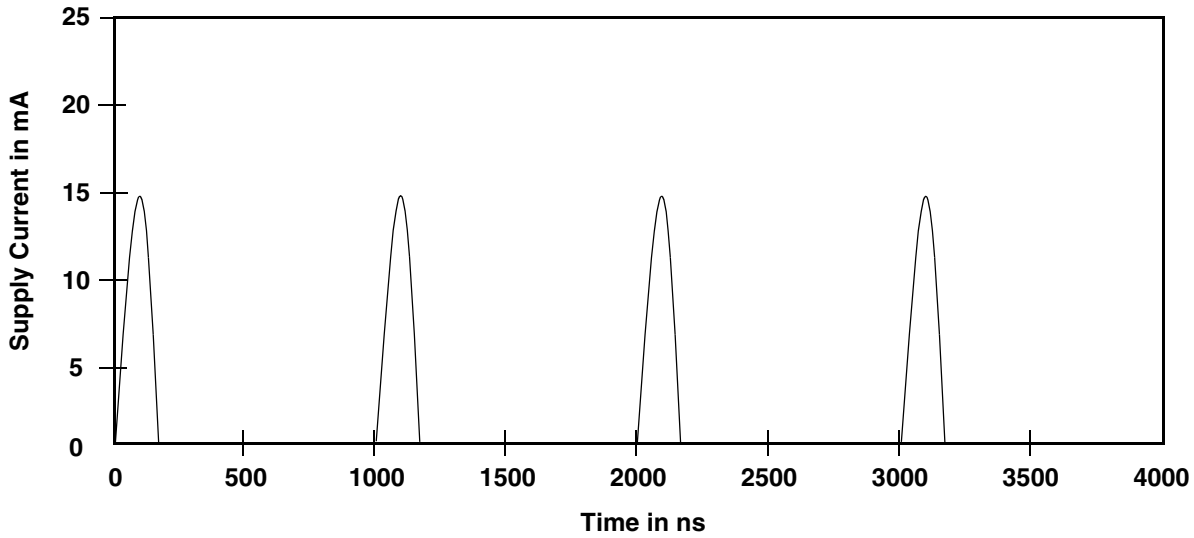
| Parameter Symbol | Parameter Description | Test Conditions | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|---|
| $I_{LI}$ | Input Load Current | $V_{IN} = V_{SS}$ to $V_{CC}$, $V_{CC} = V_{CC\ max}$ | | | | ±1.0 | µA |
| $I_{LR}$ | Reset Leakage Current | $V_{CC} = V_{CC\ max}$; RESET# = 12.5 V | | | | 35 | µA |
| $I_{LIT}$ | RESET# Input Load Current | $V_{CC} = V_{CC\ max}$; RESET# = 12.5 V | | | | 35 | µA |
| $I_{LO}$ | Output Leakage Current | $V_{OUT} = V_{SS}$ to $V_{CC}$, $V_{CC} = V_{CC\ max}$ | | | | ±1.0 | µA |
| $I_{LIA}$ | ACC Input Leakage Current | $V_{CC} = V_{CC\ max}$, WP#/ACC = $V_{ACC\ max}$ | | | | 35 | µA |
| $I_{CC1}f$ | Flash $V_{CC}$ Active Read Current (Notes 1, 2) | CE#f = $V_{IL}$, OE# = $V_{IH}$, Byte Mode | 5 MHz | | 10 | 16 | mA |
| | | | 1 MHz | | 2 | 4 | |
| | | CE#f = $V_{IL}$, OE# = $V_{IH}$, Word Mode | 5 MHz | | 10 | 16 | |
| | | | 1 MHz | | 2 | 4 | |
| $I_{CC2}f$ | Flash $V_{CC}$ Active Write Current (Notes 2, 3) | CE#f = $V_{IL}$, OE# = $V_{IH}$, WE# = $V_{IL}$ | | | 15 | 30 | mA |
| $I_{CC3}f$ | Flash $V_{CC}$ Standby Current (Note 2) | $V_{CC}f = V_{CC\ max}$, CE#f, RESET#, WP#/ACC = $V_{CC}f \pm 0.3$ V | | | 0.2 | 5 | µA |
| $I_{CC4}f$ | Flash $V_{CC}$ Reset Current (Note 2) | $V_{CC}f = V_{CC\ max}$, RESET# = $V_{SS} \pm 0.3$ V, WP#/ACC = $V_{CC}f \pm 0.3$ V | | | 0.2 | 5 | µA |
| $I_{CC5}f$ | Flash $V_{CC}$ Current Automatic Sleep Mode (Notes 2, 4) | $V_{CC}f = V_{CC\ max}$, $V_{IH} = V_{CC} \pm 0.3$ V; $V_{IL} = V_{SS} \pm 0.3$ V | | | 0.2 | 5 | µA |
| $I_{CC6}f$ | Flash $V_{CC}$ Active Read-While-Program Current (Notes 1, 2) | CE#f = $V_{IL}$, OE# = $V_{IH}$ | Byte | | 21 | 45 | mA |
| | | | Word | | 21 | 45 | |
| $I_{CC7}f$ | Flash $V_{CC}$ Active Read-While-Erase Current (Notes 1, 2) | CE#f = $V_{IL}$, OE# = $V_{IH}$ | Byte | | 21 | 45 | mA |
| | | | Word | | 21 | 45 | |
| $I_{CC8}f$ | Flash $V_{CC}$ Active Program-While-Erase-Suspended Current (Notes 2, 5) | CE#f = $V_{IL}$, OE#f = $V_{IH}$ | | | 17 | 35 | mA |
| $V_{IL}$ | Input Low Voltage | | | –0.2 | | 0.8 | V |
| $V_{IH}$ | Input High Voltage | | | 2.4 | | $V_{CC}$ + 0.2 | V |
| $V_{HH}$ | Voltage for WP#/ACC Program Acceleration and Sector Protection/Unprotection | | | 8.5 | | 9.5 | V |
| $V_{ID}$ | Voltage for Sector Protection, Autoselect and Temporary Sector Unprotect | | | 11.5 | | 12.5 | V |
| $V_{OL}$ | Output Low Voltage | $I_{OL}$ = 4.0 mA, $V_{CC}f = V_{CC}s = V_{CC\ min}$ | | | | 0.45 | V |
| $V_{OH1}$ | Output High Voltage | $I_{OH}$ = –2.0 mA, $V_{CC}f = V_{CC}s = V_{CC\ min}$ | | 0.85 x $V_{CC}$ | | | V |
| $V_{OH2}$ | | $I_{OH}$ = –100 µA, $V_{CC} = V_{CC\ min}$ | | $V_{CC}$–0.4 | | | |
| $V_{LKO}$ | Flash Low $V_{CC}$ Lock-Out Voltage (Note 5) | | | 2.3 | | 2.5 | V |

***Notes:***

1. *The $I_{CC}$ current listed is typically less than 2 mA/MHz, with OE# at $V_{IH}$.*
2. *Maximum $I_{CC}$ specifications are tested with $V_{CC} = V_{CC}$max.*
3. *$I_{CC}$ active while Embedded Erase or Embedded Program is in progress.*
4. *Automatic sleep mode enables the low power mode when addresses remain stable for $t_{ACC}$ + 30 ns. Typical sleep mode current is 200 nA.*
5. *Not 100% tested.*

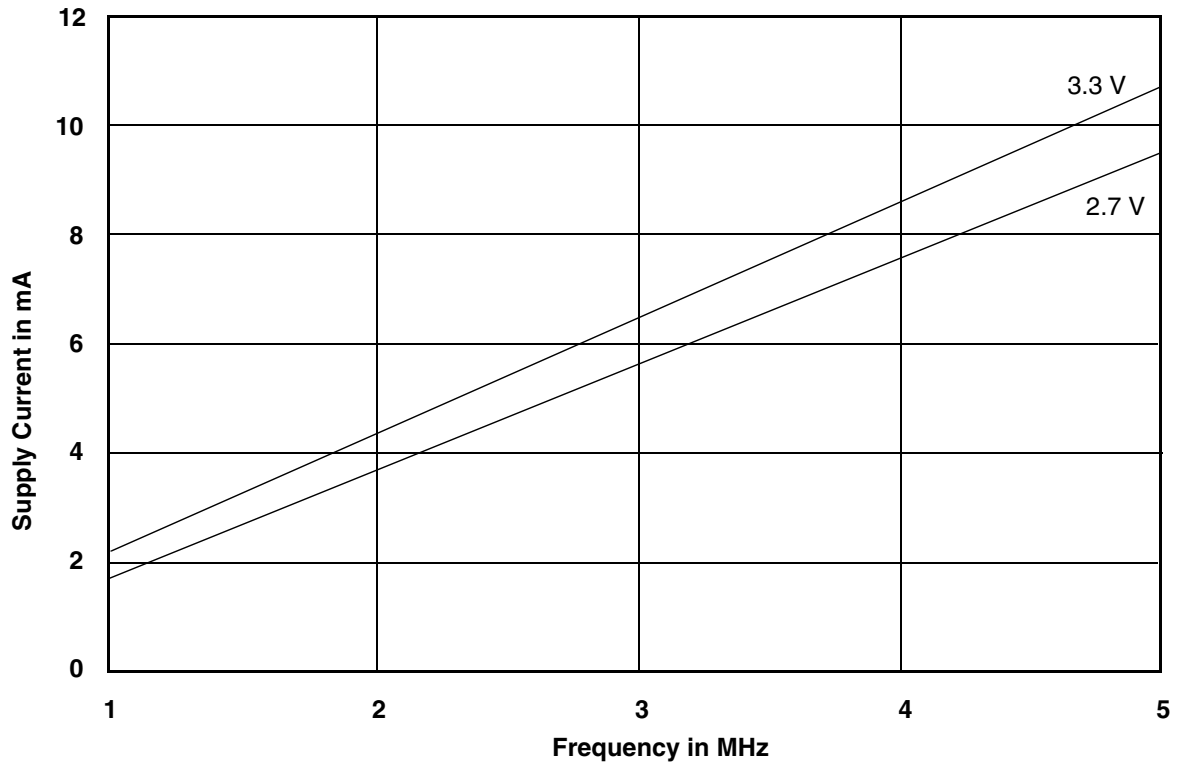
## FLASH DC CHARACTERISTICS

### Zero-Power Flash




***Note:*** *Addresses are switching at 1 MHz*

**Figure 10. $I_{CC1}$ Current vs. Time (Showing Active and Automatic Sleep Currents)**




***Note:*** *T = 25 °C*

**Figure 11. Typical $I_{CC1}$ vs. Frequency**

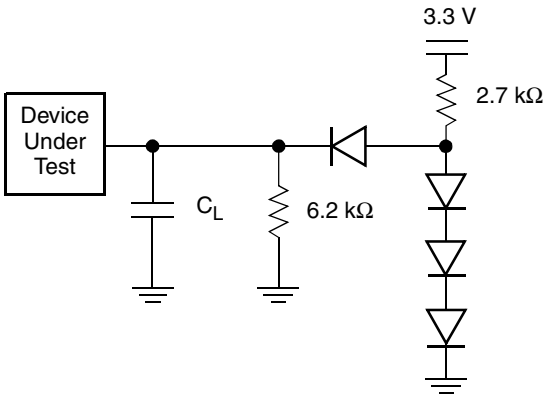
## Pseudo SRAM DC AND OPERATING CHARACTERISTICS (NOTE 1)

| Parameter Symbol | Parameter Description | Test Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $I_{LI}$ | Input Leakage Current | $V_{IN} = V_{SS}$ to $V_{CC}$ | –1.0 | | 1.0 | µA |
| $I_{LO}$ | Output Leakage Current | CE1#s = $V_{IH}$, CE2s = $V_{IL}$ or OE# = $V_{IH}$ or WE# = $V_{IL}$, $V_{IO}$= $V_{SS}$ to $V_{CC}$ | –1.0 | | 1.0 | µA |
| $I_{CC}$ | Operating Power Supply Current | $I_{IO}$ = 0 mA, CE1#s = $V_{IL}$, CE2s = $V_{IH}$, $V_{IN}$ = $V_{IH}$ or $V_{IL}$ | | | 2 | mA |
| $I_{CC1}$s | Average Operating Current | Cycle time = 1 µs, 100% duty, $I_{IO}$ = 0 mA, CE1#s ≤ 0.2 V, CE2 ≥ $V_{CC}$ – 0.2 V, $V_{IN}$ ≤ 0.2 V or $V_{IN}$ ≥ $V_{CC}$ – 0.2 V | | | 3 | mA |
| $I_{CC2}$s | Average Operating Current | Cycle time = Min., $I_{IO}$ = 0 mA, 100% duty, CE1#s = $V_{IL}$, CE2s = $V_{IH}$, $V_{IN}$ = $V_{IL}$ = or $V_{IH}$ | | | 30 | mA |
| $V_{IL}$ | Input Low Voltage | | –0.2 (Note 3) | | 0.4 | V |
| $V_{IH}$ | Input High Voltage | | 2.2 | | $V_{CC}$+0.2 (Note 2) | V |
| $V_{OL}$ | Output Low Voltage | $I_{OL}$ = 2.0 mA | | | 0.4 | V |
| $V_{OH}$ | Output High Voltage | $I_{OH}$ = –1.0 mA | 2.2 | | | V |
| $I_{SB}$ | Standby Current (TTL) | CE1#s = $V_{IH}$, CE2 = $V_{IL}$, Other inputs = $V_{IH}$ or $V_{IL}$ | | | 0.3 | mA |
| $I_{SB1}$ | Standby Current (CMOS) | CE1#s ≥ $V_{CC}$ – 0.2 V, CE2 ≥ $V_{CC}$ – 0.2 V (CE1#s controlled) or CE2 ≤ 0.2 V (CE2s controlled), CIOs = $V_{SS}$ or $V_{CC}$, Other input = 0 ~ $V_{CC}$ | | | 100 | µA |

***Notes:***

1. *$T_A$= –40° to 85°C, otherwise specified.*
2. *Overshoot: $V_{CC}$+1.0V if pulse width ≤ 20 ns.*
3. *Undershoot: –1.0V if pulse width ≤ 20 ns.*
4. *Overshoot and undershoot are sampled, not 100% tested.*
5. *Stable power supply required 200 µs before device operation.*

## TEST CONDITIONS




***Note:*** *Diodes are IN3064 or equivalent*

**Figure 12. Test Setup**

**Table 18. Test Specifications**

| Test Condition | 70, 85 | Unit |
|---|---|---|
| Output Load | 1 TTL gate | |
| Output Load Capacitance, $C_L$ (including jig capacitance) | 30 | pF |
| Input Rise and Fall Times | 5 | ns |
| Input Pulse Levels | 0.0–3.0 | V |
| Input timing measurement reference levels | 1.5 | V |
| Output timing measurement reference levels | 1.5 | V |

## KEY TO SWITCHING WAVEFORMS

| WAVEFORM | INPUTS | OUTPUTS |
|---|---|---|
| | Steady | |
| | Changing from H to L | |
| | Changing from L to H | |
| | Don't Care, Any Change Permitted | Changing, State Unknown |
| | Does Not Apply | Center Line is High Impedance State (High Z) |

KS000010-PAL

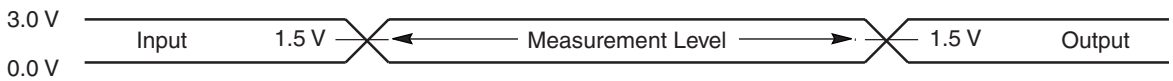



**Figure 13. Input Waveforms and Measurement Levels**

## AC CHARACTERISTICS

### Pseudo SRAM CE#s Timing

| Parameter | | Description | Test Setup | | All Speeds | Unit |
|---|---|---|---|---|---|---|
| JEDEC | Std | | | | | |
| — | $t_{CCR}$ | CE#s Recover Time | — | Min | 0 | ns |

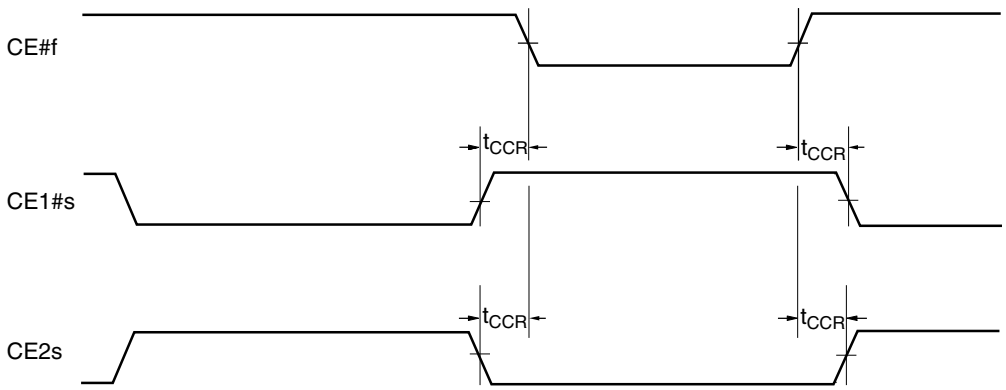



**Figure 14. Timing Diagram for Alternating Between Pseudo SRAM and Flash**

## FLASH AC CHARACTERISTICS

### Read-Only Operations

| Parameter | | Description | | Test Setup | | Speed | | Unit |
|---|---|---|---|---|---|---|---|---|
| JEDEC | Std. | | | | | 70 | 85 | |
| $t_{AVAV}$ | $t_{RC}$ | Read Cycle Time (Note 1) | | | Min | 70 | 85 | ns |
| $t_{AVQV}$ | $t_{ACC}$ | Address to Output Delay | | CE#f, OE# = $V_{IL}$ | Max | 70 | 85 | ns |
| $t_{ELQV}$ | $t_{CE}$ | Chip Enable to Output Delay | | OE# = $V_{IL}$ | Max | 70 | 85 | ns |
| $t_{GLQV}$ | $t_{OE}$ | Output Enable to Output Delay | | | Max | 30 | 40 | ns |
| $t_{EHQZ}$ | $t_{DF}$ | Chip Enable to Output High Z (Notes 1, 3) | | | Max | 16 | 16 | ns |
| $t_{GHQZ}$ | $t_{DF}$ | Output Enable to Output High Z (Notes 1, 3) | | | Max | 16 | | ns |
| $t_{AXQX}$ | $t_{OH}$ | Output Hold Time From Addresses, CE#f or OE#, Whichever Occurs First | | | Min | 0 | | ns |
| | $t_{OEH}$ | Output Enable Hold Time (Note 1) | Read | | Min | 0 | | ns |
| | | | Toggle and Data# Polling | | Min | 10 | | ns |

***Notes:***

1. *Not 100% tested.*
2. *See Figure 12 and Table 18 for test specifications*
3. *Measurements performed by placing a 50Ω termination on the data pin with a bias of $V_{CC}/2$. The time from OE# high to the data bus driven to $V_{CC}/2$ is taken as $t_{DF}$*

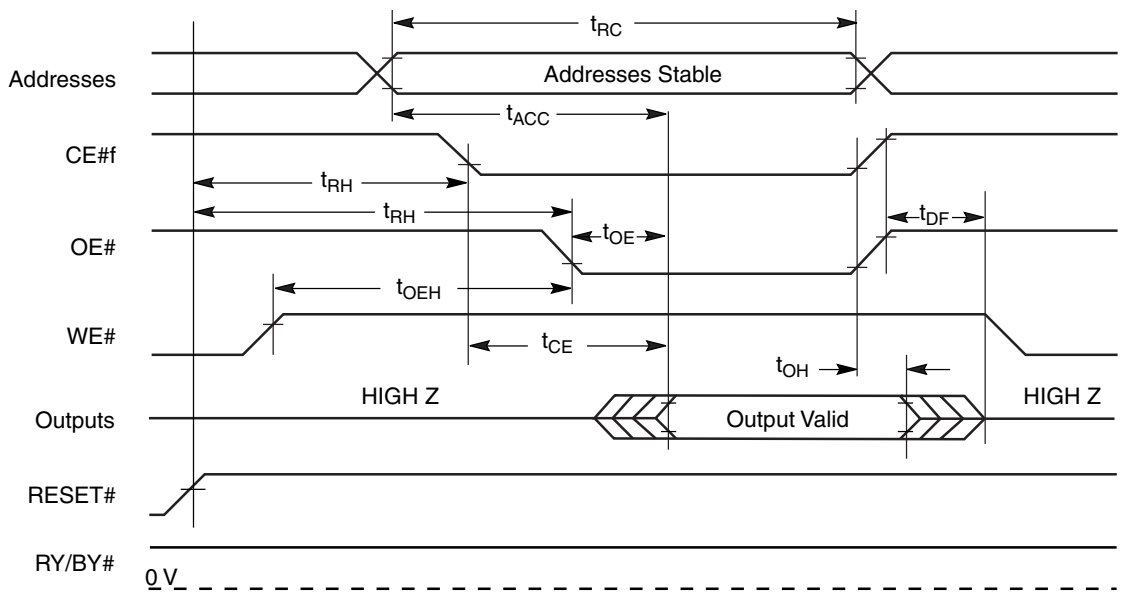



**Figure 15. Read Operation Timings**

## FLASH AC CHARACTERISTICS

### Hardware Reset (RESET#)

| Parameter | | Description | | All Speed Options | Unit |
|---|---|---|---|---|---|
| JEDEC | Std | | | | |
| | $t_{Ready}$ | RESET# Pin Low (During Embedded Algorithms) to Read Mode (See Note) | Max | 20 | µs |
| | $t_{Ready}$ | RESET# Pin Low (NOT During Embedded Algorithms) to Read Mode (See Note) | Max | 500 | ns |
| | $t_{RP}$ | RESET# Pulse Width | Min | 500 | ns |
| | $t_{RH}$ | Reset High Time Before Read (See Note) | Min | 50 | ns |
| | $t_{RPD}$ | RESET# Low to Standby Mode | Min | 20 | µs |
| | $t_{RB}$ | RY/BY# Recovery Time | Min | 0 | ns |

***Note:*** *Not 100% tested.*

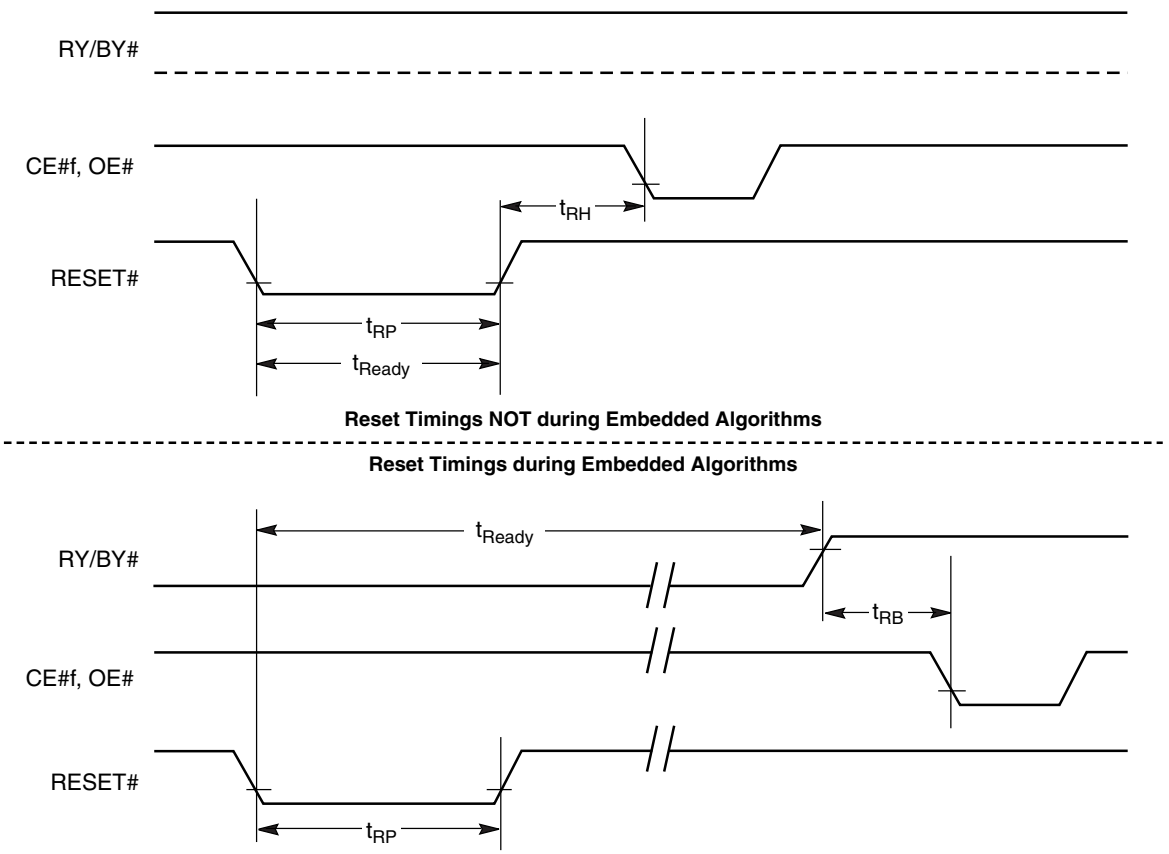



**Figure 16. Reset Timings**

## FLASH AC CHARACTERISTICS

### Word/Byte Configuration (CIOf)

| Parameter | | Description | | Speed | | Unit |
|---|---|---|---|---|---|---|
| JEDEC | Std | | | 70 | 85 | |
| | $t_{ELFL}/t_{ELFH}$ | CE#f to CIOf Switching Low or High | Max | 5 | | ns |
| | $t_{FLQZ}$ | CIOf Switching Low to Output HIGH Z | Max | 16 | | ns |
| | $t_{FHQV}$ | CIOf Switching High to Output Active | Min | 70 | 85 | ns |

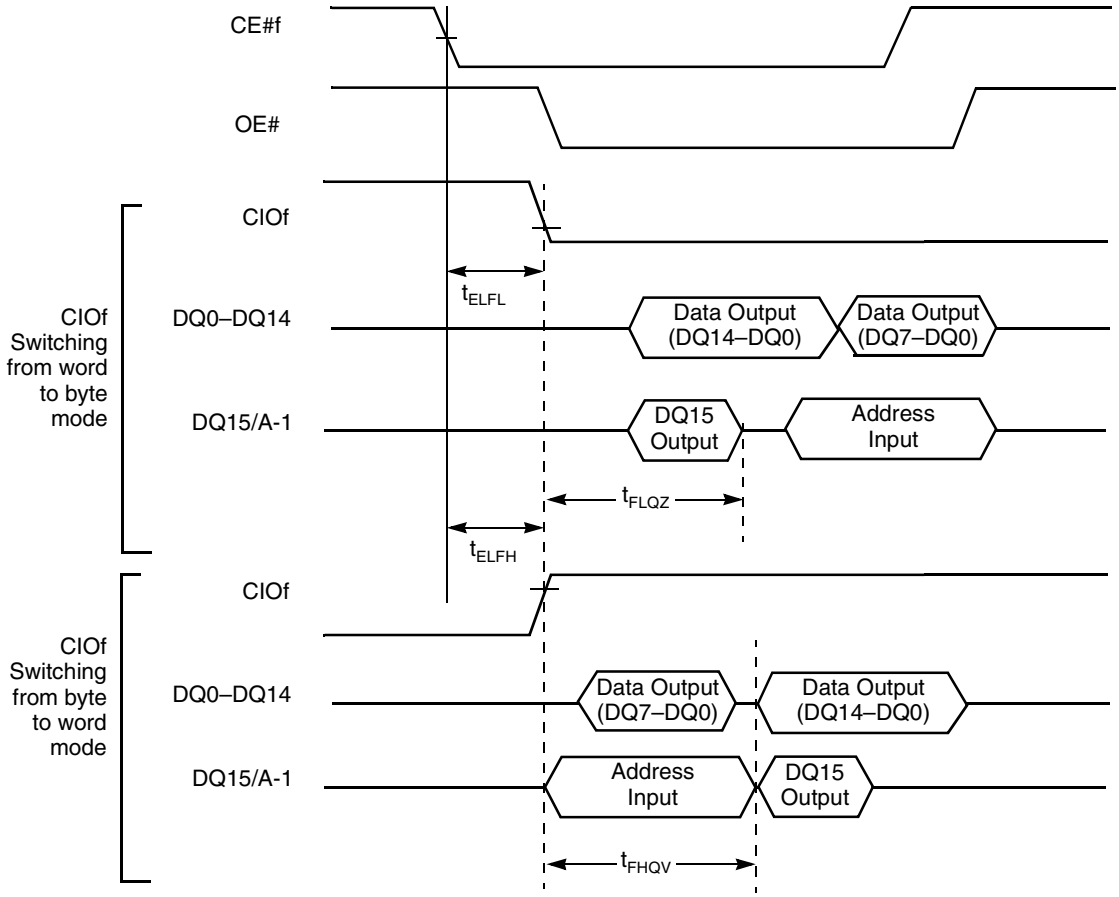
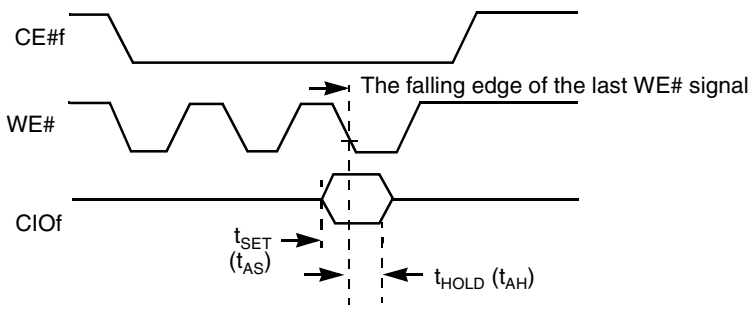



**Figure 17. CIOf Timings for Read Operations**




***Note:*** *Refer to the Erase/Program Operations table for $t_{AS}$ and $t_{AH}$ specifications.*

**Figure 18. CIOf Timings for Write Operations**

## FLASH AC CHARACTERISTICS

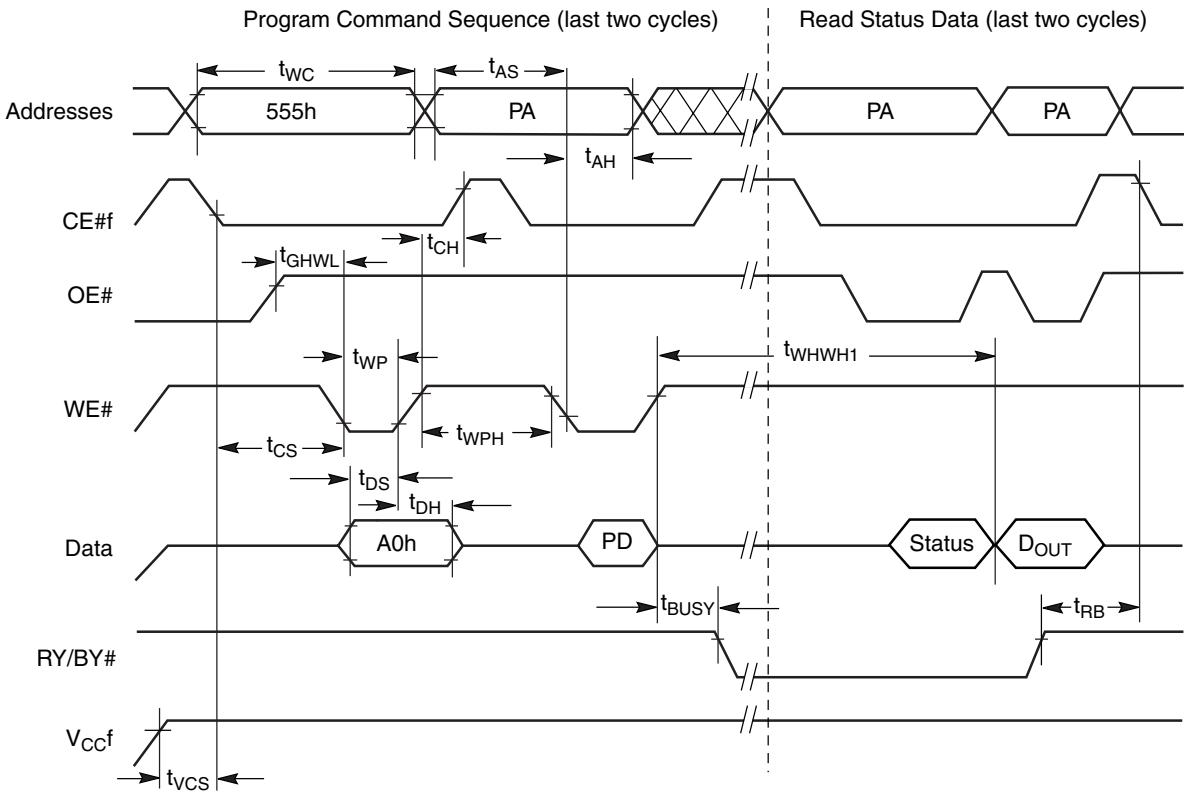
### Flash Erase and Program Operations

| Parameter | | Description | | | Speed Options | | Unit |
|---|---|---|---|---|---|---|---|
| JEDEC | Std | | | | 70 | 85 | |
| $t_{AVAV}$ | $t_{WC}$ | Write Cycle Time (Note 1) | | Min | 70 | 85 | ns |
| $t_{AVWL}$ | $t_{AS}$ | Address Setup Time (WE# to Address) | | Min | 0 | | ns |
| | $t_{ASO}$ | Address Setup Time to OE# or CE#f Low During Toggle Bit Polling | | Min | 15 | | ns |
| $t_{WLAX}$ | $t_{AH}$ | Address Hold Time (WE# to Address) | | Min | 45 | | ns |
| | $t_{AHT}$ | Address Hold Time From CE#f or OE# High During Toggle Bit Polling | | Min | 0 | | ns |
| $t_{DVWH}$ | $t_{DS}$ | Data Setup Time | | Min | 35 | 45 | ns |
| $t_{WHDX}$ | $t_{DH}$ | Data Hold Time | | Min | 0 | | ns |
| | $t_{OEH}$ | OE# Hold Time | Read | Min | 0 | | ns |
| | | | Toggle and Data# Polling | Min | 10 | | ns |
| | $t_{OEPH}$ | Output Enable High During Toggle Bit Polling | | Min | 20 | | ns |
| $t_{GHEL}$ | $t_{GHEL}$ | Read Recovery Time Before Write (OE# High to CE#f Low) | | Min | 0 | | ns |
| $t_{GHWL}$ | $t_{GHWL}$ | Read Recovery Time Before Write (OE# High to WE# Low) | | Min | 0 | | ns |
| $t_{WLEL}$ | $t_{WS}$ | WE# Setup Time (CE#f to WE#) | | Min | 0 | | ns |
| $t_{ELWL}$ | $t_{CS}$ | CE#f Setup Time (WE# to CE#f) | | Min | 0 | | ns |
| $t_{EHWH}$ | $t_{WH}$ | WE# Hold Time (CE#f to WE#) | | Min | 0 | | ns |
| $t_{WHEH}$ | $t_{CH}$ | CE#f Hold Time (CE#f to WE#) | | Min | 0 | | ns |
| $t_{WLWH}$ | $t_{WP}$ | Write Pulse Width | | Min | 30 | 35 | ns |
| $t_{ELEH}$ | $t_{CP}$ | CE#f Pulse Width | | Min | 30 | 35 | ns |
| $t_{WHDL}$ | $t_{WPH}$ | Write Pulse Width High | | Min | 30 | | ns |
| | $t_{SR/W}$ | Latency Between Read and Write Operations | | Min | 0 | | ns |
| $t_{WHWH1}$ | $t_{WHWH1}$ | Programming Operation (Note 2) | Byte | Typ | 5 | | µs |
| | | | Word | Typ | 7 | | |
| $t_{WHWH1}$ | $t_{WHWH1}$ | Accelerated Programming Operation, Word or Byte (Note 2) | | Typ | 4 | | µs |
| $t_{WHWH2}$ | $t_{WHWH2}$ | Sector Erase Operation (Note 2) | | Typ | 0.4 | | sec |
| | $t_{VCS}$ | $V_{CC}$f Setup Time (Note 1) | | Min | 50 | | µs |
| | $t_{RB}$ | Write Recovery Time From RY/BY# | | Min | 0 | | ns |
| | $t_{BUSY}$ | Program/Erase Valid To RY/BY# Delay | | Max | 90 | | ns |

***Notes:***

1. *Not 100% tested.*
2. *See the "Flash Erase And Programming Performance" section for more information.*

## FLASH AC CHARACTERISTICS




***Notes:***

1. *PA = program address, PD = program data, $D_{OUT}$ is the true data at the program address.*
2. *Illustration shows device in word mode.*

**Figure 19. Program Operation Timings**

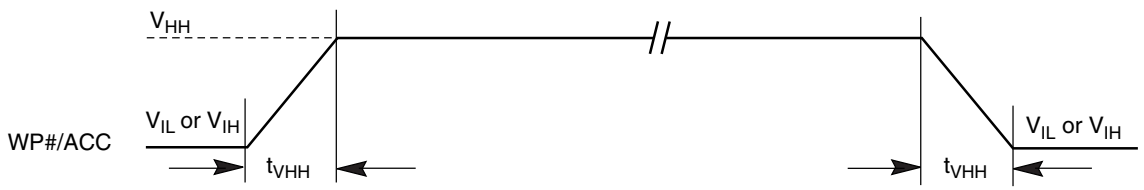
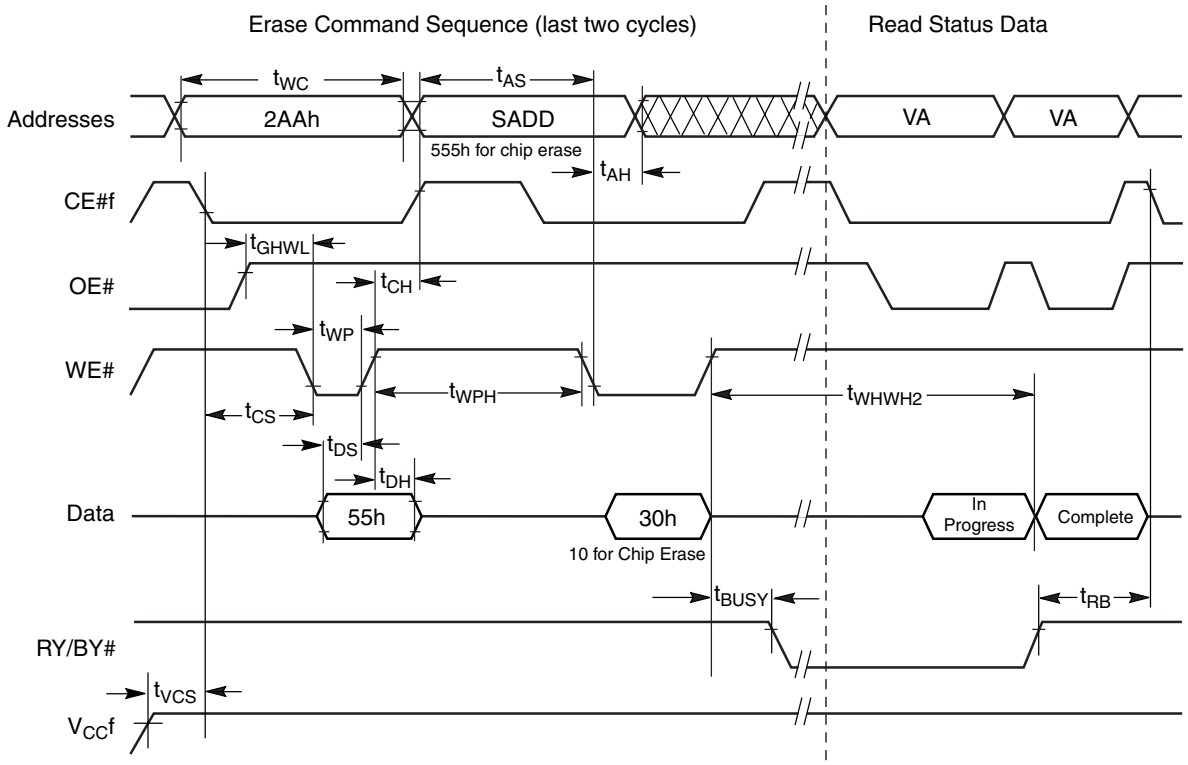



**Figure 20. Accelerated Program Timing Diagram**

## FLASH AC CHARACTERISTICS




***Notes:***

1. *SADD = sector address (for Sector Erase), VA = Valid Address for reading status data (see "Flash Write Operation Status".*
2. *These waveforms are for the word mode.*

**Figure 21. Chip/Sector Erase Operation Timings**

## FLASH AC CHARACTERISTICS

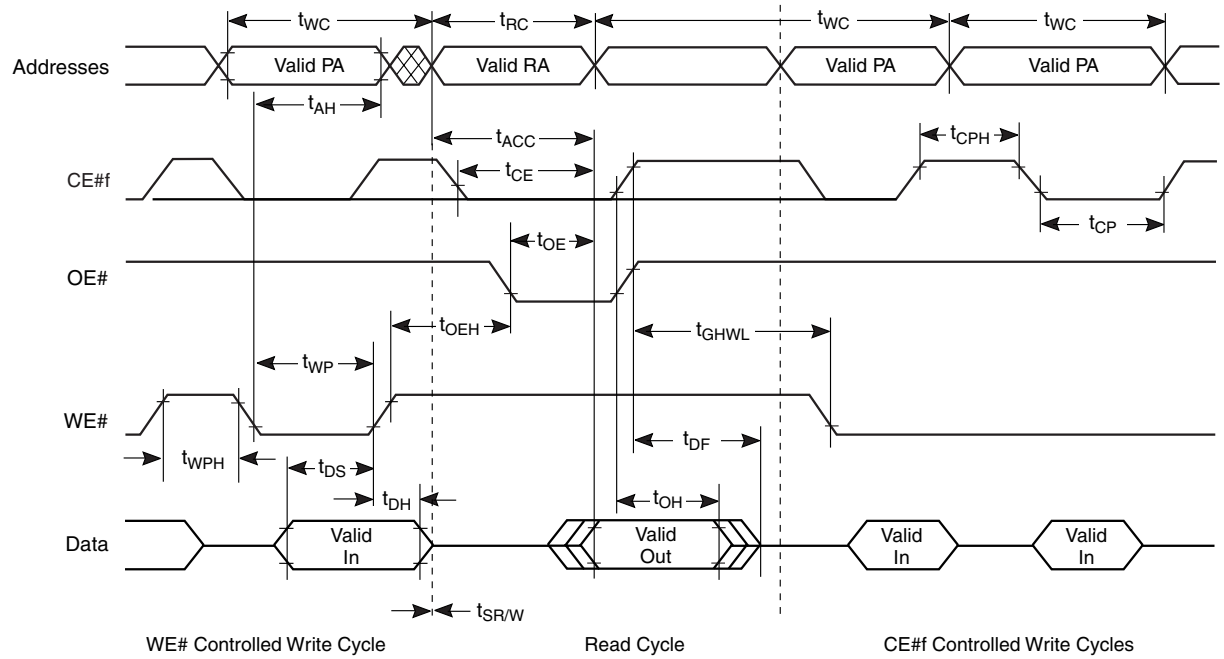
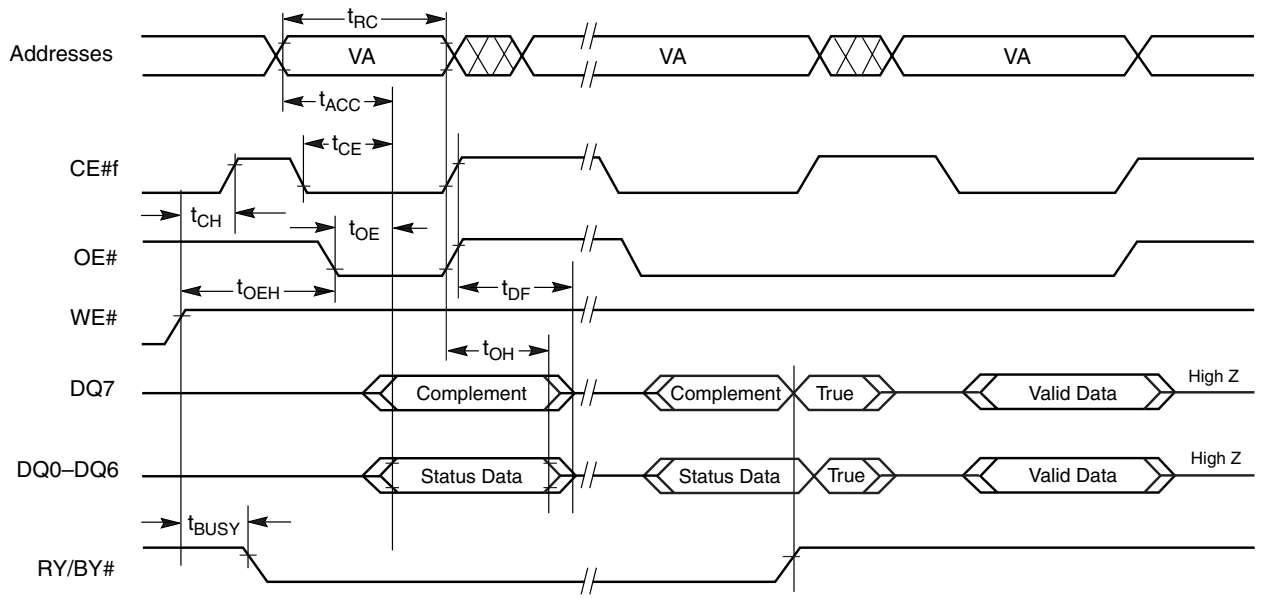


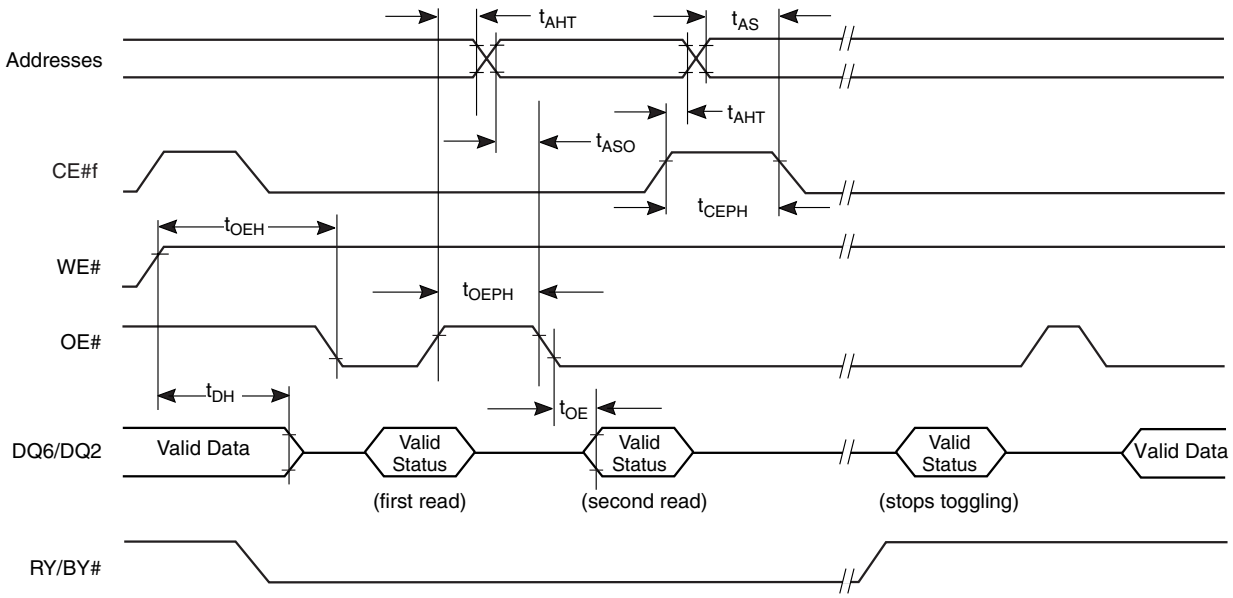

**Figure 22. Back-to-back Read/Write Cycle Timings**




***Note:*** *VA = Valid address. Illustration shows first status cycle after command sequence, last status read cycle, and array data read cycle.*

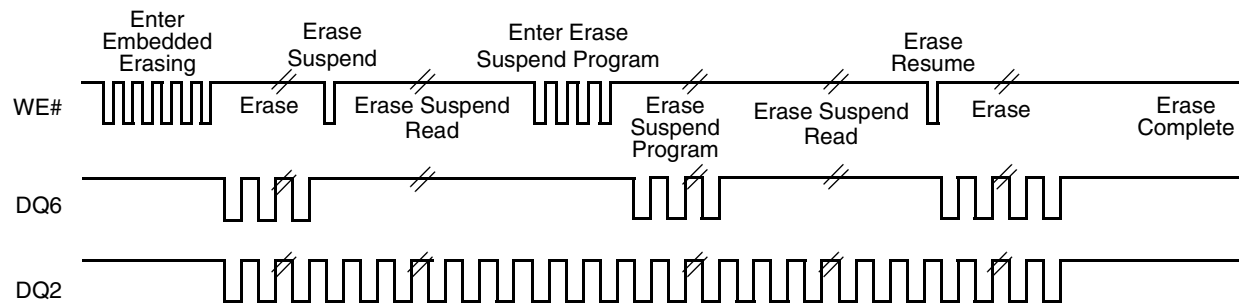
**Figure 23. Data# Polling Timings (During Embedded Algorithms)**

## FLASH AC CHARACTERISTICS




***Note:*** *VA = Valid address; not required for DQ6. Illustration shows first two status cycle after command sequence, last status read cycle, and array data read cycle.*

**Figure 24. Toggle Bit Timings (During Embedded Algorithms)**




***Note:*** *DQ2 toggles only when read at an address within an erase-suspended sector. The system may use OE# or CE#f to toggle DQ2 and DQ6.*

**Figure 25. DQ2 vs. DQ6**

## FLASH AC CHARACTERISTICS

### Temporary Sector Unprotect

| Parameter | | | | | |
|---|---|---|---|---|---|
| JEDEC | Std | Description | | All Speed Options | Unit |
| | $t_{VIDR}$ | $V_{ID}$ Rise and Fall Time (See Note) | Min | 500 | ns |
| | $t_{VHH}$ | $V_{HH}$ Rise and Fall Time (See Note) | Min | 250 | ns |
| | $t_{RSP}$ | RESET# Setup Time for Temporary Sector Unprotect | Min | 4 | µs |
| | $t_{RRB}$ | RESET# Hold Time from RY/BY# High for Temporary Sector Unprotect | Min | 4 | µs |

***Note:*** *Not 100% tested.*

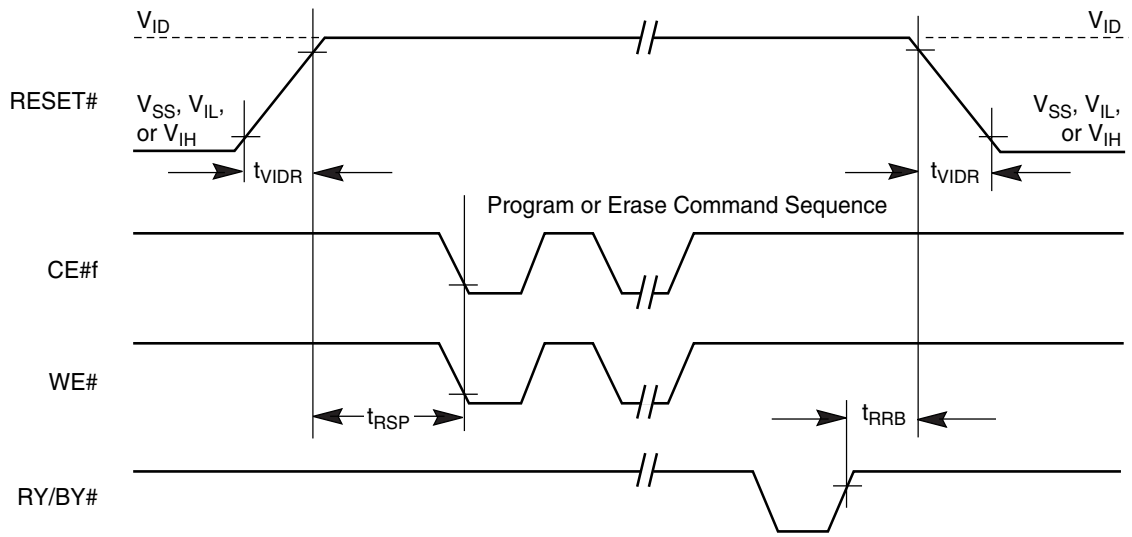
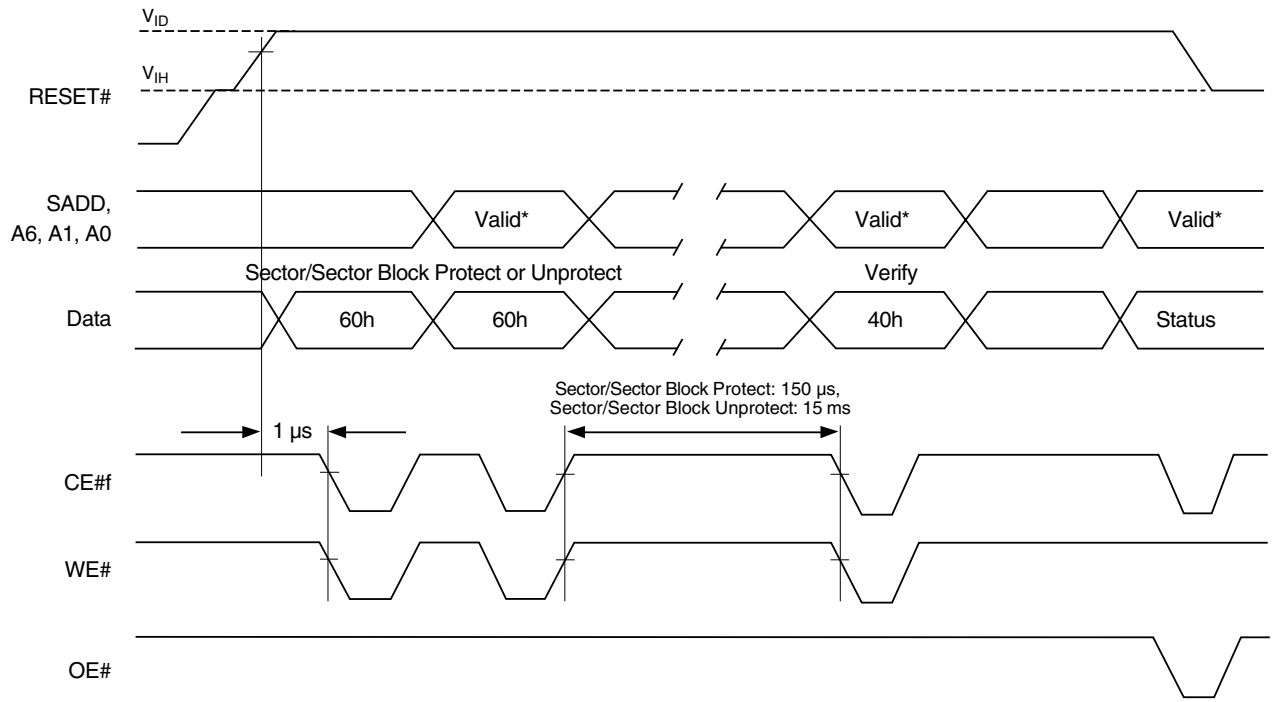



**Figure 26. Temporary Sector Unprotect Timing Diagram**

## FLASH AC CHARACTERISTICS




** For sector protect, A6 = 0, A1 = 1, A0 = 0. For sector unprotect, A6 = 1, A1 = 1, A0 = 0, SADD = Sector Address.*

**Figure 27. Sector/Sector Block Protect and Unprotect Timing Diagram**

## FLASH AC CHARACTERISTICS

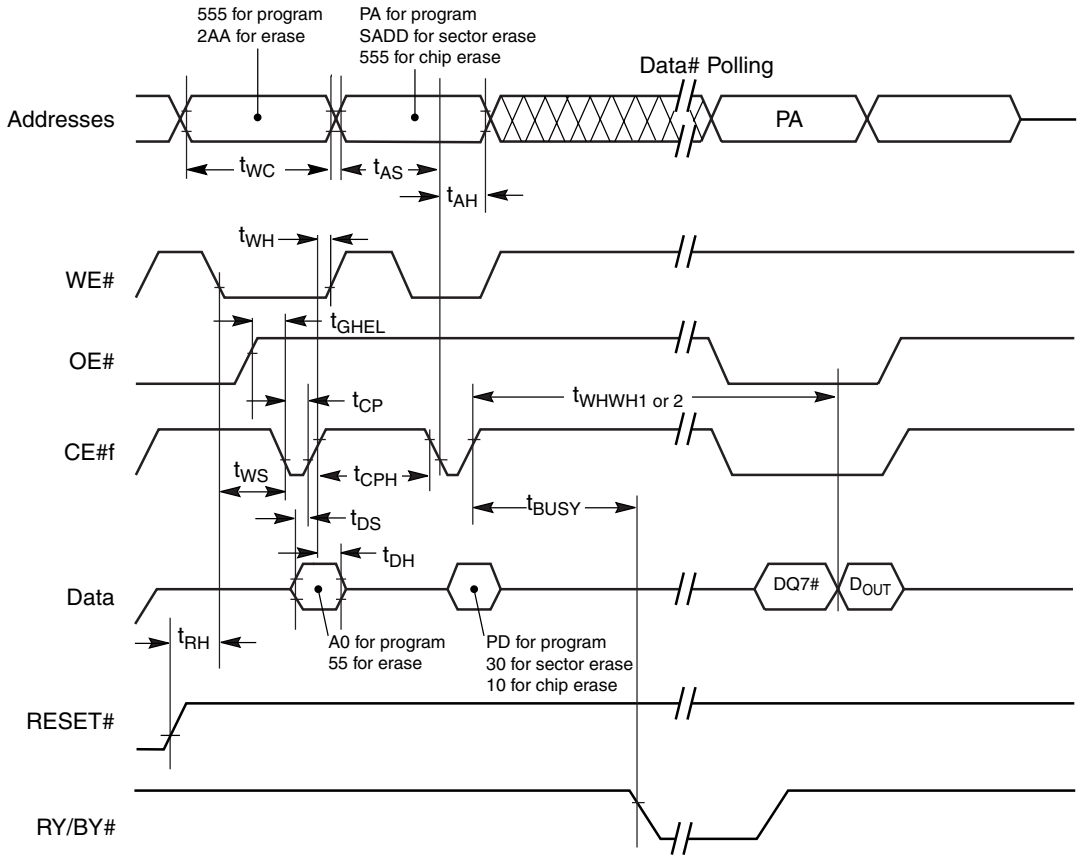
### Alternate CE#f Controlled Erase and Program Operations

| Parameter | | Description | | | Speed | | Unit |
|---|---|---|---|---|---|---|---|
| JEDEC | Std | | | | 70 | 85 | |
| $t_{AVAV}$ | $t_{WC}$ | Write Cycle Time (Note 1) | | Min | 70 | 85 | ns |
| $t_{AVWL}$ | $t_{AS}$ | Address Setup Time | | Min | 0 | | ns |
| $t_{ELAX}$ | $t_{AH}$ | Address Hold Time | | Min | 40 | 45 | ns |
| $t_{DVEH}$ | $t_{DS}$ | Data Setup Time | | Min | 40 | 45 | ns |
| $t_{EHDX}$ | $t_{DH}$ | Data Hold Time | | Min | 0 | | ns |
| $t_{GHEL}$ | $t_{GHEL}$ | Read Recovery Time Before Write (OE# High to WE# Low) | | Min | 0 | | ns |
| $t_{WLEL}$ | $t_{WS}$ | WE# Setup Time | | Min | 0 | | ns |
| $t_{EHWH}$ | $t_{WH}$ | WE# Hold Time | | Min | 0 | | ns |
| $t_{ELEH}$ | $t_{CP}$ | CE#f Pulse Width | | Min | 40 | 45 | ns |
| $t_{EHEL}$ | $t_{CPH}$ | CE#f Pulse Width High | | Min | 30 | | ns |
| $t_{WHWH1}$ | $t_{WHWH1}$ | Programming Operation (Note 2) | Byte | Typ | 5 | | µs |
| | | | Word | Typ | 7 | | |
| $t_{WHWH1}$ | $t_{WHWH1}$ | Accelerated Programming Operation, Word or Byte (Note 2) | | Typ | 4 | | µs |
| $t_{WHWH2}$ | $t_{WHWH2}$ | Sector Erase Operation (Note 2) | | Typ | 0.4 | | sec |

***Notes:***

1. *Not 100% tested.*
2. *See the "Flash Erase And Programming Performance" section for more information.*

## FLASH AC CHARACTERISTICS




***Notes:***

1. *Figure indicates last two bus cycles of a program or erase operation.*
2. *PA = program address, SADD = sector address, PD = program data.*
3. *DQ7# is the complement of the data written to the device. $D_{OUT}$ is the data written to the device.*
4. *Waveforms are for the word mode.*

**Figure 28. Flash Alternate CE#f Controlled Write (Erase/Program) Operation Timings**

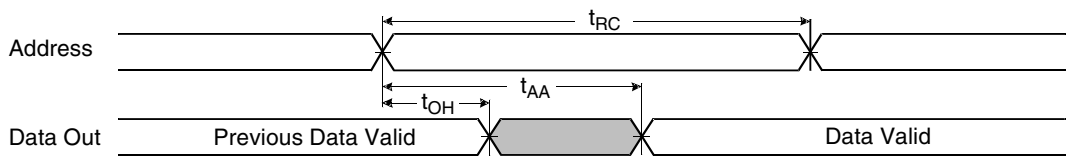
## Pseudo SRAM AC CHARACTERISTICS

### Power Up Time

When powering up the SRAM, maintain $V_{CC}$s for 100 µs minimum with CE#1s at $V_{IH}$.

### Read Cycle

| Parameter Symbol | Description | | Speed | | Unit |
|---|---|---|---|---|---|
| | | | 70 | 85 | |
| $t_{RC}$ | Read Cycle Time | Min | 70 | 85 | ns |
| $t_{AA}$ | Address Access Time | Max | 70 | 85 | ns |
| $t_{CO1}$, $t_{CO2}$ | Chip Enable to Output | Max | 70 | 85 | ns |
| $t_{OE}$ | Output Enable Access Time | Max | 35 | 40 | ns |
| $t_{BA}$ | LB#s, UB#s to Access Time | Max | 70 | 85 | ns |
| $t_{LZ1}$, $t_{LZ2}$ | Chip Enable (CE1#s Low and CE2s High) to Low-Z Output | Min | 10 | | ns |
| $t_{BLZ}$ | UB#, LB# Enable to Low-Z Output | Min | 10 | | ns |
| $t_{OLZ}$ | Output Enable to Low-Z Output | Min | 5 | | ns |
| $t_{HZ1}$, $t_{HZ2}$ | Chip Disable to High-Z Output | Max | 25 | | ns |
| $t_{BHZ}$ | UB#s, LB#s Disable to High-Z Output | Max | 25 | | ns |
| $t_{OHZ}$ | Output Disable to High-Z Output | Max | 25 | | ns |
| $t_{OH}$ | Output Data Hold from Address Change | Min | 10 | | ns |



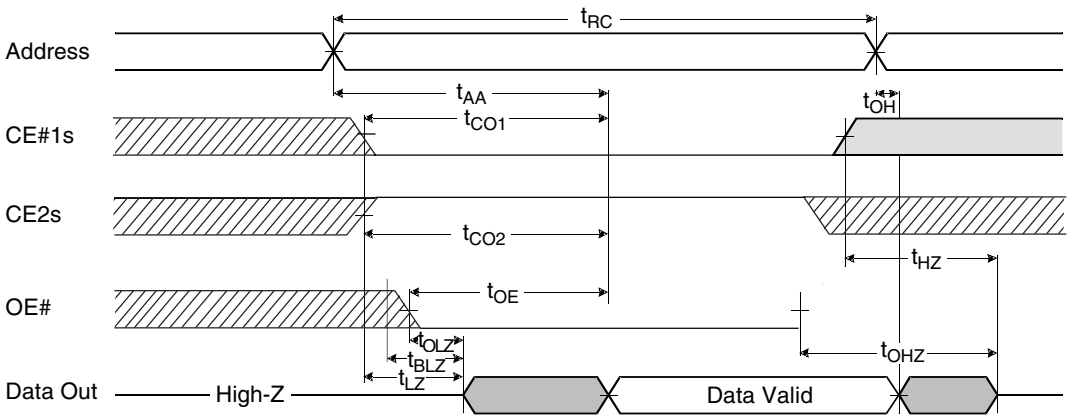

***Notes:***

1. *CE1#s = OE# = $V_{IL}$, CE2s = WE# = $V_{IH}$, UB#s and/or LB#s = $V_{IL}$*
2. *Do not access device with cycle timing shorter than $t_{RC}$ for continuous periods < 10 µs.*

**Figure 29. Pseudo SRAM Read Cycle—Address Controlled**

## Pseudo SRAM AC CHARACTERISTICS

### Read Cycle




***Notes:***

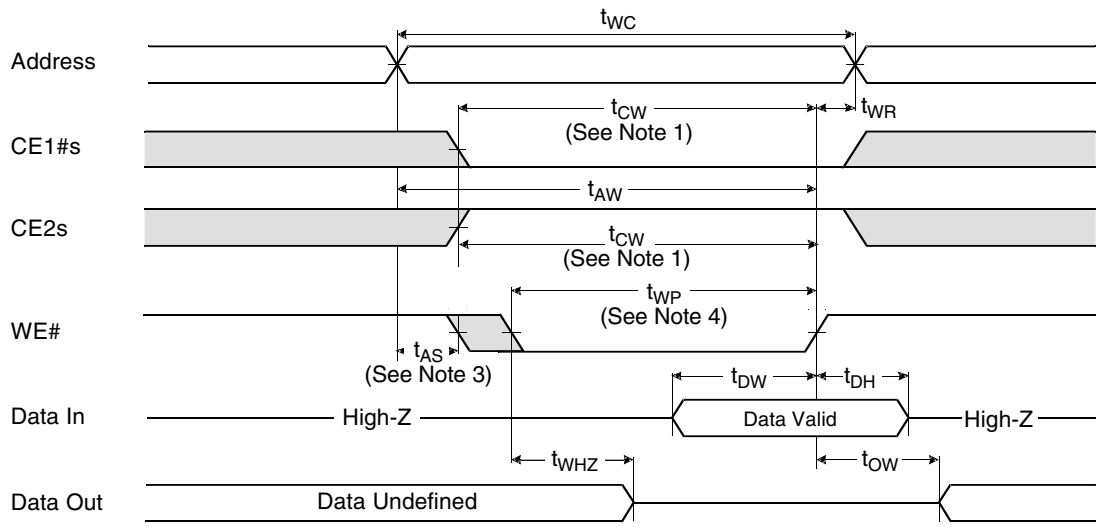
1. *WE# = $V_{IH}$, if CIOs is low, ignore UB#s/LB#s timing.*
2. *$t_{HZ}$ and $t_{OHZ}$ are defined as the time at which the outputs achieve the open circuit conditions and are not referenced to output voltage levels.*
3. *At any given temperature and voltage condition, $t_{HZ}$ (Max.) is less than $t_{LZ}$ (Min.) both for a given device and from device to device interconnection.*
4. *Do not access device with cycle timing shorter than $t_{RC}$ for continuous periods < 10 µs.*

**Figure 30. Pseudo SRAM Read Cycle**

## Pseudo SRAM AC CHARACTERISTICS

### Write Cycle

| Parameter Symbol | Description | | Speed | | Unit |
|---|---|---|---|---|---|
| | | | 70 | 85 | |
| $t_{WC}$ | Write Cycle Time | Min | 70 | 85 | ns |
| $t_{Cw}$ | Chip Enable to End of Write | Min | 60 | 70 | ns |
| $t_{AS}$ | Address Setup Time | Min | 0 | | ns |
| $t_{AW}$ | Address Valid to End of Write | Min | 60 | 70 | ns |
| $t_{BW}$ | UB#s, LB#s to End of Write | Min | 60 | 70 | ns |
| $t_{WP}$ | Write Pulse Time | Min | 50 | 60 | ns |
| $t_{WR}$ | Write Recovery Time | Min | 0 | | ns |
| $t_{WHZ}$ | Write to Output High-Z | Min | 0 | | ns |
| | | Max | 20 | 25 | |
| $t_{DW}$ | Data to Write Time Overlap | Min | 40 | 45 | ns |
| $t_{DH}$ | Data Hold from Write Time | Min | 0 | | ns |
| $t_{OW}$ | End Write to Output Low-Z | min | 5 | | ns |

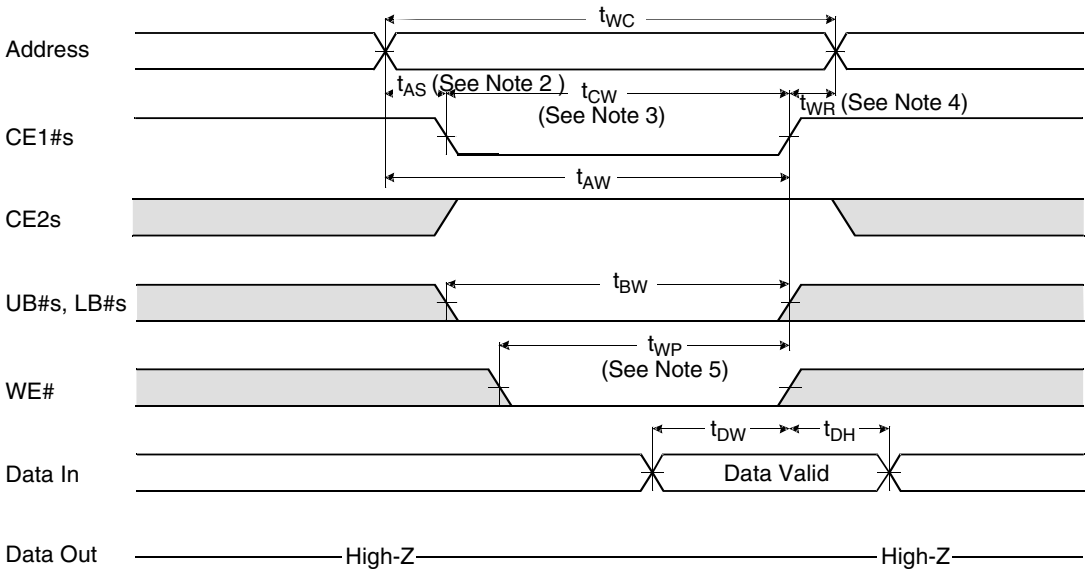



***Notes:***

1. *WE# controlled, if CIOs is low, ignore UB#s and LB#s timing.*
2. *$t_{CW}$ is measured from CE1#s going low to the end of write.*
3. *$t_{WR}$ is measured from the end of write to the address change. $t_{WR}$ applied in case a write ends as CE1#s or WE# going high.*
4. *$t_{AS}$ is measured from the address valid to the beginning of write.*
5. *A write occurs during the overlap ($t_{WP}$) of low CE#1 and low WE#. A write begins when CE1#s goes low and WE# goes low when asserting UB#s or LB#s for a single byte operation or simultaneously asserting UB#s and LB#s for a double byte operation. A write ends at the earliest transition when CE1#s goes high and WE# goes high. The $t_{WP}$ is measured from the beginning of write to the end of write.*

**Figure 31. Pseudo SRAM Write Cycle—WE# Control**

## Pseudo SRAM AC CHARACTERISTICS

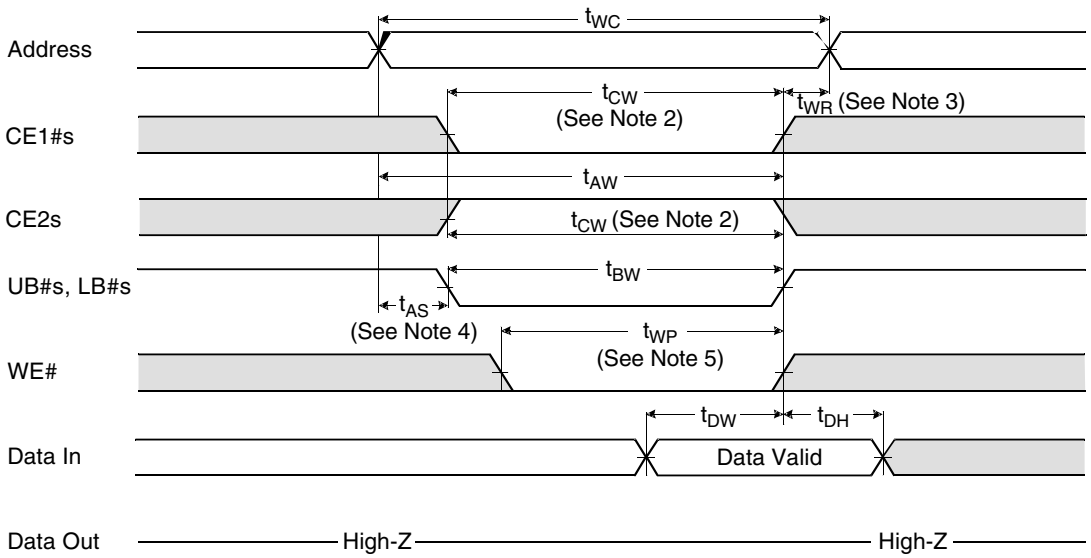



***Notes:***

1. *CE1#s controlled, if CIOs is low, ignore UB#s and LB#s timing.*
2. *$t_{CW}$ is measured from CE1#s going low to the end of write.*
3. *$t_{WR}$ is measured from the end of write to the address change. $t_{WR}$ applied in case a write ends as CE1#s or WE# going high.*
4. *$t_{AS}$ is measured from the address valid to the beginning of write.*
5. *A write occurs during the overlap ($t_{WP}$) of low CE1#s and low WE#. A write begins when CE1#s goes low and WE# goes low when asserting UB#s or LB#s for a single byte operation or simultaneously asserting UB#s and LB#s for a double byte operation. A write ends at the earliest transition when CE1#s goes high and WE# goes high. The $t_{WP}$ is measured from the beginning of write to the end of write.*

**Figure 32. Pseudo SRAM Write Cycle—CE1#s Control**

## Pseudo SRAM AC CHARACTERISTICS




***Notes:***

1. *UB#s and LB#s controlled, CIOs must be high.*
2. *$t_{CW}$ is measured from CE1#s going low to the end of write.*
3. *$t_{WR}$ is measured from the end of write to the address change. $t_{WR}$ applied in case a write ends as CE1#s or WE# going high.*
4. *$t_{AS}$ is measured from the address valid to the beginning of write.*
5. *A write occurs during the overlap ($t_{WP}$) of low CE#1s and low WE#. A write begins when CE1#s goes low and WE# goes low when asserting UB#s or LB#s for a single byte operation or simultaneously asserting UB#s and LB#s for a double byte operation. A write ends at the earliest transition when CE1#s goes high and WE# goes high. The $t_{WP}$ is measured from the beginning of write to the end of write.*

**Figure 33. Pseudo SRAM Write Cycle—UB#s and LB#s Control**

## FLASH ERASE AND PROGRAMMING PERFORMANCE

| Parameter | | Typ (Note 1) | Max (Note 2) | Unit | Comments |
|---|---|---|---|---|---|
| Sector Erase Time | | 0.4 | 5 | sec | Excludes 00h programming prior to erasure (Note 4) |
| Chip Erase Time | | 28 | | sec | |
| Byte Program Time | | 5 | 150 | µs | Excludes system level overhead (Note 5) |
| Accelerated Byte/Word Program Time | | 4 | 120 | µs | |
| Word Program Time | | 7 | 210 | µs | |
| Chip Program Time (Note 3) | Byte Mode | 21 | 63 | sec | |
| | Word Mode | 14 | 42 | | |

***Notes:***

1. *Typical program and erase times assume the following conditions: 25°C, 3.0 V $V_{CC}$, 1,000,000 cycles. Additionally, programming typicals assume checkerboard pattern.*
2. *Under worst case conditions of 90°C, $V_{CC}$ = 2.7 V, 1,000,000 cycles.*
3. *The typical chip programming time is considerably less than the maximum chip programming time listed, since most bytes program faster than the maximum program times listed.*
4. *In the pre-programming step of the Embedded Erase algorithm, all bytes are programmed to 00h before erasure.*
5. *System-level overhead is the time required to execute the two- or four-bus-cycle sequence for the program command. See Tables 15 and 16 for further information on command definitions.*
6. *The device has a minimum erase and program cycle endurance of 1,000,000 cycles.*

## LATCHUP CHARACTERISTICS

| Description | Min | Max |
|---|---|---|
| Input voltage with respect to $V_{SS}$ on all pins except I/O pins (including A9, OE#, and RESET#) | –1.0 V | 12.5 V |
| Input voltage with respect to $V_{SS}$ on all I/O pins | –1.0 V | $V_{CC}$ + 1.0 V |
| $V_{CC}$ Current | –100 mA | +100 mA |

***Note:*** *Includes all pins except $V_{CC}$. Test conditions: $V_{CC}$ = 3.0 V, one pin at a time.*

## PACKAGE PIN CAPACITANCE

| Parameter Symbol | Parameter Description | Test Setup | Typ | Max | Unit |
|---|---|---|---|---|---|
| $C_{IN}$ | Input Capacitance | $V_{IN}$ = 0 | 11 | 14 | pF |
| $C_{OUT}$ | Output Capacitance | $V_{OUT}$ = 0 | 12 | 16 | pF |
| $C_{IN2}$ | Control Pin Capacitance | $V_{IN}$ = 0 | 14 | 16 | pF |
| $C_{IN3}$ | WP#/ACC Pin Capacitance | $V_{IN}$ = 0 | 17 | 20 | pF |

***Notes:***

1. *Sampled, not 100% tested.*
2. *Test conditions $T_A$ = 25°C, f = 1.0 MHz.*

## FLASH DATA RETENTION

| Parameter Description | Test Conditions | Min | Unit |
|---|---|---|---|
| Minimum Pattern Data Retention Time | 150°C | 10 | Years |
| | 125°C | 20 | Years |

## SRAM DATA RETENTION

| Parameter Symbol | Parameter Description | Test Setup | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{DR}$ | $V_{CC}$ for Data Retention | CS1#s ≥ $V_{CC}$ – 0.2 V (Note 1) | 2.7 | | 3.3 | V |
| $I_{DR}$ | Data Retention Current | $V_{CC}$ = 3.0 V, CE1#s ≥ $V_{CC}$ – 0.2 V (Note 1) | | 1.0 (Note 2) | 100 | µA |
| $t_{SDR}$ | Data Retention Set-Up Time | See data retention waveforms | 0 | | | ns |
| $t_{RDR}$ | Recovery Time | | $t_{RC}$ | | | ns |

***Notes:***

1. *CE1#s ≥ $V_{CC}$ – 0.2 V, CE2s ≥ $V_{CC}$ – 0.2 V (CE1#s controlled) or CE2s ≤ 0.2 V (CE2s controlled), CIOs = $V_{SS}$ or $V_{CC}$.*
2. *Typical values are not 100% tested.*

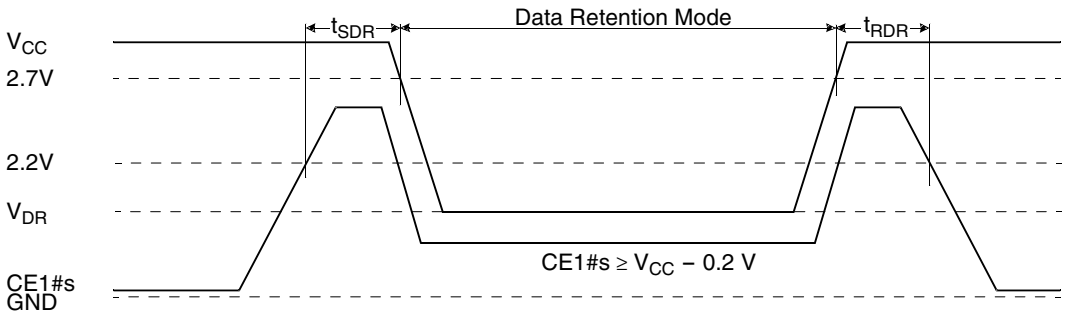



**Figure 34. CE1#s Controlled Data Retention Mode**

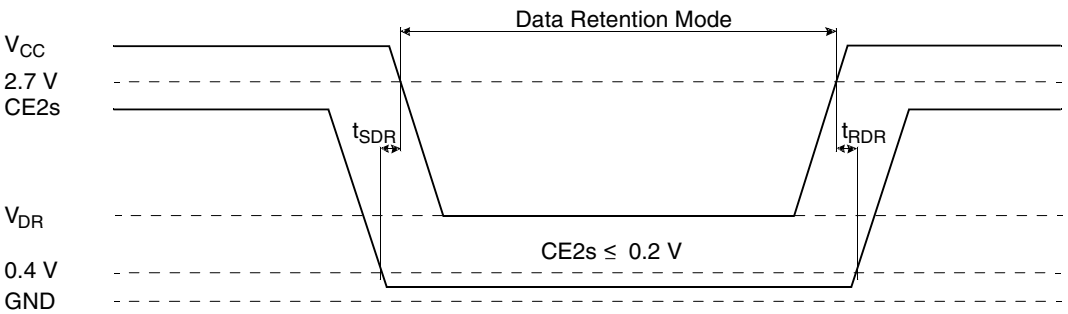
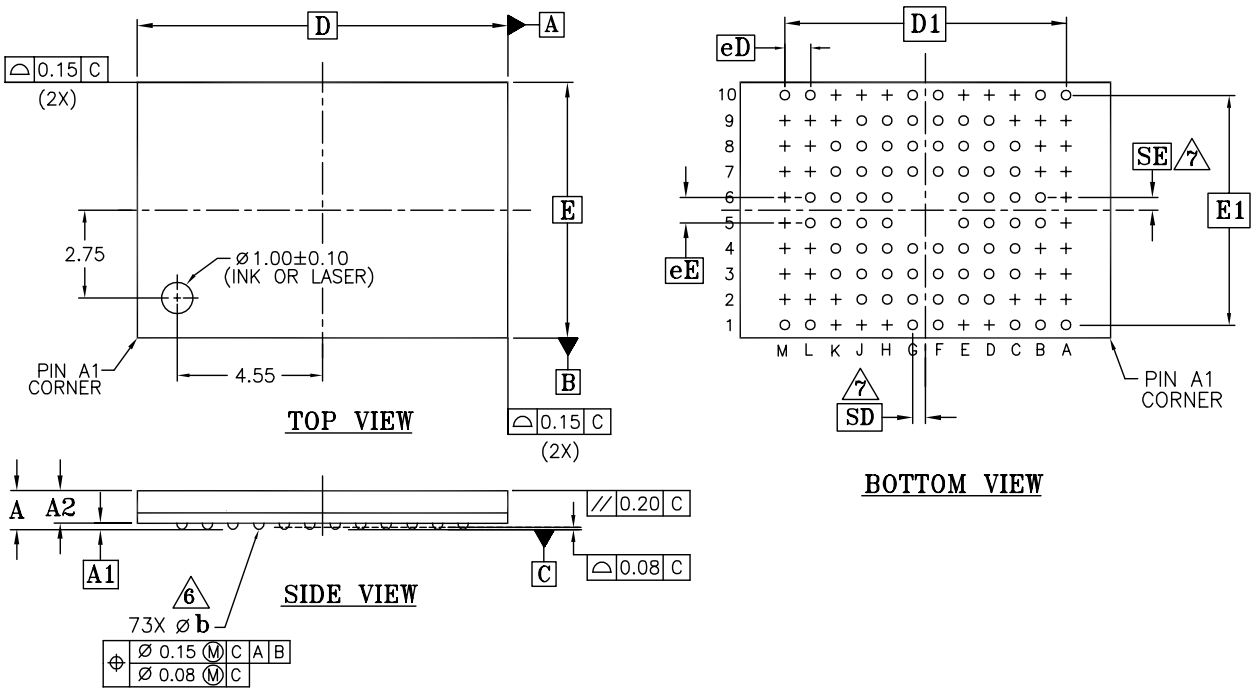



**Figure 35. CE2s Controlled Data Retention Mode**

## PHYSICAL DIMENSIONS

### FLB073—73-Ball Fine-Pitch Grid Array 8 x 11.6 mm




| PACKAGE | FLB 073 | | | NOTE |
|---|---|---|---|---|
| JEDEC | N/A | | | |
| | 8.00mm X 11.60mm PACKAGE | | | |
| SYMBOL | MIN. | NOM. | MAX. | |
| A | – | – | 1.40 | PROFILE |
| A1 | 0.20 | – | 0.30 | BALL HEIGHT |
| A2 | 0.95 | – | 1.09 | BODY THICKNESS |
| D | 11.60 BSC | | | BODY SIZE |
| E | 8.00 BSC | | | BODY SIZE |
| D1 | 8.80 BSC | | | MATRIX FOOTPRINT |
| E1 | 7.20 BSC | | | MATRIX FOOTPRINT |
| MD | 12 | | | MATRIX SIZE D DIRECTION |
| ME | 10 | | | MATRIX SIZE E DIRECTION |
| n | 73 | | | BALL COUNT |
| ø b | 0.25 | 0.30 | 0.35 | BALL DIAMETER |
| eE | 0.80 BSC | | | BALL PITCH |
| eD | 0.80 BSC | | | BALL PITCH |
| SD/SE | 0.40 BSC | | | SOLDER BALL PLACEMENT |
| | A2,A3,A4,A5,A6,A7,A8,A9,B2,B3,B4,B7,B8,B9 C10,D1,D10,E1,E10,F5,F6,G5,G6,H1,H10 J1,J10,K1,K10,L2,L3,L4,L7,L8,L9, M2,M3,M4,M5,M6,M7,M8,M9 | | | DEPOPULATED SOLDER BALLS |

NOTES:

1. DIMENSIONING AND TOLERANCING METHODS PER ASME Y14.5M–1994.
2. ALL DIMENSIONS ARE IN MILLIMETERS.
3. BALL POSITION DESIGNATION PER JESD 95–1, SPP–010.
4. e REPRESENTS THE SOLDER BALL GRID PITCH.
5. SYMBOL "MD" IS THE BALL MATRIX SIZE IN THE "D" DIRECTION.
   SYMBOL "ME" IS THE BALL MATRIX SIZE IN THE "E" DIRECTION.
   n IS THE NUMBER OF POPULATED SOLDER BALL POSITIONS FOR MATRIX SIZE MD X ME.

6 DIMENSION "b" IS MEASURED AT THE MAXIMUM BALL DIAMETER IN A PLANE PARALLEL TO DATUM C.

7 SD AND SE ARE MEASURED WITH RESPECT TO DATUMS A AND B AND DEFINE THE POSITION OF THE CENTER SOLDER BALL IN THE OUTER ROW.
WHEN THERE IS AN ODD NUMBER OF SOLDER BALLS IN THE OUTER ROW SD OR SE = 0.000.
WHEN THERE IS AN EVEN NUMBER OF SOLDER BALLS IN THE OUTER ROW, SD OR SE = e/2

8. "+" INDICATES THE THEORETICAL CENTER OF DEPOPULATED BALLS.
9. A1 CORNER TO BE IDENTIFIED BY CHAMFER, LASER OR INK MARK, METALLIZED MARK INDENTATION OR OTHER MEANS.

# REVISION SUMMARY

## Revision A (June 4, 2002)

Initial release.

## Revision A+1 (June 18, 2002)

### Pseudo SRAM DC and Operating Characteristics

Added $V_{IH}$ and $V_{IL}$ specifications and added notes to table.

## Revision A+2 (July 9, 2002)

### Command Definitions (Flash Word Mode)

Changed 80h to 82h for factory locked and 00h to 002h for not factory locked.

## Revision B (February 28, 2003)

### Customer Lockable: SecSi Sector NOT Programmed or Protected at the factory.

Added second bullet, SecSi sector-protect verify text and figure 3.

### SecSi Sector Flash Memory Region, and Enter SecSi Sector/Exit SecSi Sector Command Sequence

Noted that the ACC function and unlock bypass modes are not available when the SecSi sector is enabled.

### Byte/Word Program Command Sequence, Sector Erase Command Sequence, and Chip Erase Command Sequence

Noted that the SecSi Sector, autoselect, and CFI functions are unavailable when a program or erase operation is in progress.

### Common Flash Memory Interface (CFI)

Changed CFI website address.

## Revision B+1 (March 12, 2004)

### Table 15. Command Definitions (Flash Word Mode)

In Note 9, changed device ID for top boot to 01h and for bottom boot to 00h.

## Sales Offices and Representatives

### North America

ALABAMA . . . . . (256)830-9192
ARIZONA . . . . . (602)242-4400
CALIFORNIA,
  Irvine . . . . . (949)450-7500
  Sunnyvale . . . . . (408)732-2400
COLORADO . . . . . (303)741-2900
CONNECTICUT . . . . . (203)264-7800
FLORIDA,
  Clearwater . . . . . (727)793-0055
  Miami (Lakes) . . . . . (305)820-1113
GEORGIA . . . . . (770)814-0224
ILLINOIS,
  Chicago . . . . . (630)773-4422
MASSACHUSETTS . . . . . (781)213-6400
MICHIGAN . . . . . (248)471-6294
MINNESOTA . . . . . (612)745-0005
NEW JERSEY,
  Chatham . . . . . (973)701-1777
NEW YORK . . . . . (716)425-8050
NORTH CAROLINA . . . . . (919)840-8080
OREGON . . . . . (503)245-0080
PENNSYLVANIA . . . . . (215)340-1187
SOUTH DAKOTA . . . . . (605)692-5777
TEXAS,
  Austin . . . . . (512)346-7830
  Dallas . . . . . (972)985-1344
  Houston . . . . . (281)376-8084
VIRGINIA . . . . . (703)736-9568

### International

AUSTRALIA, North Ryde . . . . . TEL(61)2-88-777-222
BELGIUM, Antwerpen . . . . . TEL(32)3-248-43-00
BRAZIL, San Paulo . . . . . TEL(55)11-5501-2105
CHINA,
  Beijing . . . . . TEL(86)10-6510-2188
  Shanghai . . . . . TEL(86)21-635-00838
  Shenzhen . . . . . TEL(86)755-246-1550
FINLAND, Helsinki . . . . . TEL(358)881-3117
FRANCE, Paris . . . . . TEL(33)-1-49751010
GERMANY,
  Bad Homburg . . . . . TEL(49)-6172-92670
  Munich . . . . . TEL(49)-89-450530
HONG KONG, Causeway Bay . . . . . TEL(85)2-2956-0388
ITALY, Milan . . . . . TEL(39)-02-381961
INDIA, New Delhi . . . . . TEL(91)11-623-8620
JAPAN,
  Osaka . . . . . TEL(81)6-6243-3250
  Tokyo . . . . . TEL(81)3-3346-7600
KOREA, Seoul . . . . . TEL(82)2-3468-2600
RUSSIA, Moscow . . . . . TEL(7)-095-795-06-22
SWEDEN, Stockholm . . . . . TEL(46)8-562-540-00
TAIWAN, Taipei . . . . . TEL(886)2-8773-1555
UNITED KINGDOM,
  Frimley . . . . . TEL(44)1276-803100
  Haydock . . . . . TEL(44)1942-272888



### Representatives in U.S. and Canada

ARIZONA,
  Tempe - Centaur . . . . . (480)839-2320
CALIFORNIA,
  Calabasas - Centaur . . . . . (818)878-5800
  Irvine - Centaur . . . . . (949)261-2123
  San Diego - Centaur. . . . . (858)278-4950
  Santa Clara - Fourfront. . . . . (408)350-4800
CANADA,
  Burnaby, B.C. - Davetek Marketing. . . . . (604)430-3680
  Calgary, Alberta - Davetek Marketing. . . . . (403)283-3577
  Kanata, Ontario - J-Squared Tech. . . . . (613)592-9540
  Mississauga, Ontario - J-Squared Tech. . . . . (905)672-2030
  St Laurent, Quebec - J-Squared Tech. . . . . (514)747-1211
COLORADO,
  Golden - Compass Marketing . . . . . (303)277-0456
FLORIDA,
  Melbourne - Marathon Technical Sales . . . . . (321)728-7706
  Ft. Lauderdale - Marathon Technical Sales . . . . . (954)527-4949
  Orlando - Marathon Technical Sales . . . . . (407)872-5775
  St. Petersburg - Marathon Technical Sales . . . . . (727)894-3603
GEORGIA,
  Duluth - Quantum Marketing . . . . . (678)584-1128
ILLINOIS,
  Skokie - Industrial Reps, Inc. . . . . (847)967-8430
INDIANA,
  Kokomo - SAI . . . . . (765)457-7241
IOWA,
  Cedar Rapids - Lorenz Sales . . . . . (319)294-1000
KANSAS,
  Lenexa - Lorenz Sales . . . . . (913)469-1312
MASSACHUSETTS,
  Burlington - Synergy Associates . . . . . (781)238-0870
MICHIGAN,
  Brighton - SAI . . . . . (810)227-0007
MINNESOTA,
  St. Paul - Cahill, Schmitz & Cahill, Inc. . . . . (651)699-0200
MISSOURI,
  St. Louis - Lorenz Sales . . . . . (314)997-4558
NEW JERSEY,
  Mt. Laurel - SJ Associates . . . . . (856)866-1234
NEW YORK,
  Buffalo - Nycom, Inc. . . . . (716)741-7116
  East Syracuse - Nycom, Inc. . . . . (315)437-8343
  Pittsford - Nycom, Inc. . . . . (716)586-3660
  Rockville Centre - SJ Associates . . . . . (516)536-4242
NORTH CAROLINA,
  Raleigh - Quantum Marketing . . . . . (919)846-5728
OHIO,
  Middleburg Hts - Dolfuss Root & Co. . . . . (440)816-1660
  Powell - Dolfuss Root & Co. . . . . (614)781-0725
  Vandalia - Dolfuss Root & Co. . . . . (937)898-9610
  Westerville - Dolfuss Root & Co. . . . . (614)523-1990
OREGON,
  Lake Oswego - I Squared, Inc. . . . . (503)670-0557
UTAH,
  Murray - Front Range Marketing . . . . . (801)288-2500
VIRGINIA,
  Glen Burnie - Coherent Solution, Inc. . . . . (410)761-2255
WASHINGTON,
  Kirkland - I Squared, Inc. . . . . (425)822-9220
WISCONSIN,
  Pewaukee - Industrial Representatives . . . . . (262)574-9393

### Representatives in Latin America

ARGENTINA,
  Capital Federal Argentina/WW Rep. . . . . 54-11)4373-0655
CHILE,
  Santiago - LatinRep/WWRep. . . . . (+562)264-0993
COLUMBIA,
  Bogota - Dimser. . . . . (571)410-4182
MEXICO,
  Guadalajara - LatinRep/WW Rep. . . . . (523)817-3900
  Mexico City - LatinRep/WW Rep. . . . . (525)752-2727
  Monterrey - LatinRep/WW Rep. . . . . (528)369-6828
PUERTO RICO,
  Boqueron - Infitronics. . . . . (787)851-6000

AMD

One AMD Place, P.O. Box 3453, Sunnyvale, CA 94088-3453 408-732-2400
TWX 910-339-9280 TELEX 34-6306 800-538-8450 http://www.amd.com


01/03
Printed in USA