# PUMA 84FV256006 - 90/12/15

**Issue 5.3 July 2001**

## Description

The PUMA 84 range of devices provide a high density, surface mount memory solution with density up to twice that of standard monolithic devices.

The PUMA 84 may accomodate various memory technologies including SRAM, FLASH and EEPROM. The devices are designed to offer a defined upgrade path and may be user configured as 8, 16 or 32 bits wide.

The PUMA 84FV256006 is a 8M x 32, 3.3V FLASH Module in a 84 'J' Leaded package which complies with the JEDEC 84 PLCC standard.

Access times of 90/120/150 ns are available.

The 3.3V device is available to commercial and industrial temperature grade.

## Features

- Access times of 90, 120 and 150ns.
- 3.3V ± 10% $V_{CC}$.
- Commercial and Industrial temperature grades
- JEDEC Standard 84 'J' Lead surface mount package.
- May be organised as 8M x 32, 16M x 16 and 32M x 8
- Flexible Sector (64 KByte) Architecture.
- Embedded Algorithms.
- Multiple Ground pins for maximum noise immunity.

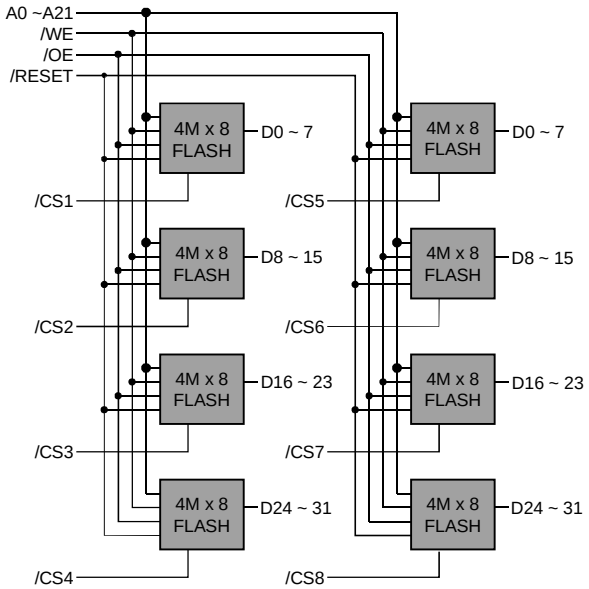
## Package Details

PUMA 84 - Plastic 84 'J' Leaded Package.

Max. Dimensions - 30.35 x 30.35 x 6.00 (nom)

All Dimensions in mm.

## Block Diagram




## Pin Definition

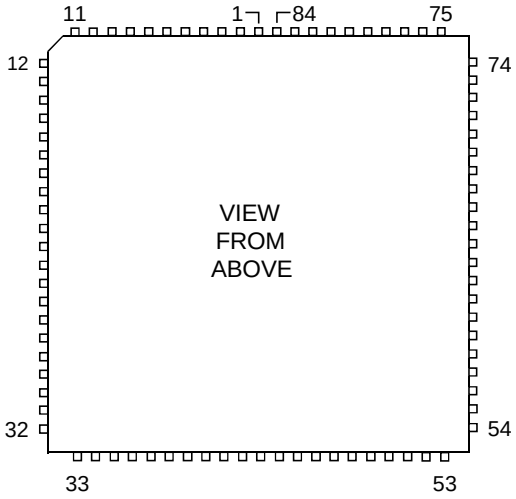
**See page 2.**

## Pin Functions

| Description | Signal |
|---|---|
| Address Input | A0~A21 |
| Data Input/Output | D0~D31 |
| Chip Select | /CS1~8 |
| Write Enable | /WE |
| Output Enable | /OE |
| Hardware Reset | /RESET |
| No Connect | NC |
| Power | $V_{CC}$ |
| Ground | GND |

**Elm Road, West Chirton Industrial Estate, North Shields, NE29 8SE, England.**
**TEL +44 (0191) 2930500. FAX +44 (0191) 2590997 E-mail: hmpsales@mosaicsemi.com**

## Pin Definition - PUMA84FV256006




| Pin | Signal | Pin | Signal |
|---|---|---|---|
| 1 | $V_{CC}$ | 43 | $V_{CC}$ |
| 2 | NC | 44 | A13 |
| 3 | /CS1 | 45 | A12 |
| 4 | /CS2 | 46 | A11 |
| 5 | /CS3 | 47 | A10 |
| 6 | /CS4 | 48 | A9 |
| 7 | A17 | 49 | A8 |
| 8 | A18 | 50 | A7 |
| 9 | D16 | 51 | D0 |
| 10 | A19 | 52 | NC |
| 11 | A20 | 53 | NC |
| 12 | A21 | 54 | NC |
| 13 | NC | 55 | NC |
| 14 | D17 | 56 | D1 |
| 15 | D18 | 57 | D2 |
| 16 | D19 | 58 | D3 |
| 17 | GND | 59 | GND |
| 18 | D20 | 60 | D4 |
| 19 | D21 | 61 | D5 |
| 20 | D22 | 62 | D6 |
| 21 | D23 | 63 | D7 |
| 22 | $V_{CC}$ | 64 | $V_{CC}$ |
| 23 | D24 | 65 | D8 |
| 24 | D25 | 66 | D9 |
| 25 | D26 | 67 | D10 |
| 26 | D27 | 68 | D11 |
| 27 | GND | 69 | GND |
| 28 | D28 | 70 | D12 |
| 29 | D29 | 71 | D13 |
| 30 | D30 | 72 | D14 |
| 31 | NC | 73 | NC |
| 32 | NC | 74 | NC |
| 33 | /RESET | 75 | D15 |
| 34 | NC | 76 | A14 |
| 35 | D31 | 77 | A15 |
| 36 | A6 | 78 | A16 |
| 37 | A5 | 79 | /WE |
| 38 | A4 | 80 | /OE |
| 39 | A3 | 81 | /CS5 |
| 40 | A2 | 82 | /CS6 |
| 41 | A1 | 83 | /CS7 |
| 42 | A0 | 84 | /CS8 |

## Absolute Maximum Ratings(1)

| Parameter | Symbol | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Supply Voltage | $V_{CC}$ | -2.0 | - | +7.0 | V |
| Voltage on $V_{CC}$ relative to GND | $V_T$ | -0.5 | - | +4.0 | V |
| Voltage on A9, /OE, /RESET relative to GND | $V_{A9}$ | -0.5 | - | +12.5 | V |
| All other pins relative to GND | $V_{AA}$ | -0.5 | - | $V_{CC}$+0.5V | V |
| Storage Temperature | $T_{STG}$ | -65 | - | +100 | $^{O}$C |

*Notes : (1) Minimum DC Voltage on input or I/O pins is -0.5V. During voltage transitions, inputs may overshoot GND to -2.0V for periods of up to 20ns . See Maximum DC Voltage on output and I/O pins is $V_{CC}$+0.5V. During Voltage transitions, outputs may overshoot to $V_{CC}$+2.0V for periods up to 20ns.*

*(2) Minimum DC input voltage on A9, /OE, /RESET pins is -0.5V. During voltage transitions, A9 ,/OE and /RESET pins may overshoot GND to -2.0V for periods of up to 20ns. Maximum DC Input voltage on A9, /OE and /RESET is 12.5V which may overshoot to 13.5V fo periods up to 20ns.*

*(3) No more than one output shorted at a time. Duration of a short circuit should not be greater than one second. Stresses Greater than those listed in this section 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 of this specification is not implied. Exposure of the device to absolute maximum rating conditions for extended periods may affect device reliability.*

## Recommended Operating Conditions

| Parameter | | Symbol | Min | Typ | Max | Unit | |
|---|---|---|---|---|---|---|---|
| Supply Voltage | | $V_{CC}$ | 3.0 | 3.3 | 3.6 | V | |
| Input High Voltage | | $V_{IH}$ | $V_{CC}$x 0.7 | - | $V_{CC}$+0.3 | V | |
| Input Low Voltage | | $V_{IL}$ | -0.5 | - | 0.8 | V | |
| Operating Temperature | (Commercial) | $T_A$ | 0 | - | 70 | $^{O}$C | |
| | (Industrial) | $T_{AI}$ | -40 | - | 85 | $^{O}$C | (I Suffix) |

## Capacitance

($V_{CC}$ = 3.3V, $T_A$ = 25$^{O}$C, F=1MHz.)

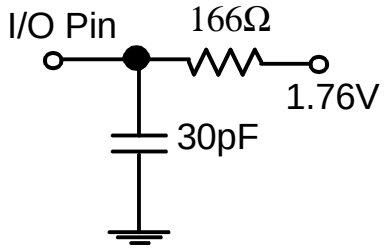
| Parameter | Symbol | Test Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Input Capacitance | $C_{IN}$ | $V_{IN}$=0V | - | 6 | 7.5 | pF |
| Output Capacitance | $C_{OUT}$ | $V_{OUT}$=0V | - | 8.5 | 12 | pF |

*Note : These Parameters are calculated not measured.*

## Test Conditions

- Input pulse levels : 0V to 3.0V
- Input rise and fall times : 20ns
- Input and Output timing reference levels : 1.5V
- Output Load : See Load Diagram.
- Module tested in 32 bit mode.
- $V_{CC}$ = 3.3V±10%

## Output Load



## DC Electrical Characteristics

($V_{CC}$=3.3V±10%, $T_A$=-40°C to 85°C)

| Parameter | | Symbol | Test Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|---|
| Input Load Current | | $I_{LI1}$ | $V_{IN}$ = GND to $V_{CC}$, $V_{CC}$ = $V_{CC}$ max | -8 | - | +8 | µA |
| A9 Input Load Current | | $I_{LI2}$ | $V_{CC}$ = $V_{CC}$ max ; A9 = 12.5 V | - | - | 280 | µA |
| Output Leakage Current | | $I_{LO}$ | $V_{OUT}$ = GND to $V_{CC}$ , $V_{CC}$ = $V_{CC}$ max | -8 | - | +8 | µA |
| $V_{CC}$ Active Read(1,2) | 32 Bit | $I_{CCR32}$ | /CS# = $V_{IL}$, /OE = $V_{IH}$, 5MHZ | - | - | 77 | mA |
| | 16 Bit | $I_{CCR16}$ | As Above | - | - | 39 | mA |
| | 8 Bit | $I_{CCR8}$ | As Above | - | - | 20 | mA |
| $V_{CC}$ Active Write(1,3,5) | 32 Bit | $I_{CCW32}$ | /CS# = $V_{IL}$, /OE = $V_{IH}$ | - | - | 144 | mA |
| | 16 Bit | $I_{CCW16}$ | As Above | - | - | 72 | mA |
| | 8 Bit | $I_{CCW8}$ | As Above | - | - | 36 | mA |
| Standby Supply Current(1) | TTL | $I_{SB}$ | $V_{CC}$=$V_{CC}$max,/CS=$V_{IH}$(1) /OE=$V_{IH}$ | - | - | 40 | µA |
| $V_{CC}$ Reset Current(1) | | $I_{CCRES}$ | /RESET = GND ± 0.3 V, | | | 40 | µA |
| Automatic Sleep Mode(1,4) | | $I_{CCASM}$ | $V_{IH}$ = $V_{CC}$ ± 0.3 V; $V_{IL}$ = GND ± 0.3 V | | | 40 | µA |
| Autoselect and Temporary Sector Unprotect Voltage | | $V_{ID}$ | $V_{CC}$ = 3.3 V | 11.5 | - | 12.5 | V |
| Output Voltage Low | | $V_{OL}$ | $I_{OL}$=12mA, $V_{CC}$ = $V_{CC}$ Min | - | - | 0.45 | V |
| Output Voltage High | | $V_{OH1}$ | $I_{OH}$=-2.5mA, $V_{CC}$ = $V_{CC}$ Min | 2.4 | - | - | V |
| Low $V_{CC}$ Lock-Out Voltage(5) | | $V_{LKO}$ | | 2.3 | - | 2.5 | V |

***Notes:***

1. *Maximum $I_{CC}$ specifications are tested with $V_{CC}$ = $V_{CC}$ max.*
2. *The $I_{CC}$ current listed is typically is less than 2 mA/MHz, with /OE at $V_{IH}$ . Typical specifications are for $V_{CC}$ = 3.0 V.*
3. *$I_{CC}$ active while Embedded Erase or Embedded Program is in progress.*
4. *Automatic sleep mode enables the low power mode. Typical sleep mode current is 200 nA.*
5. *Not 100% tested.*

## Read Cycle

| | | 90 | | 120 | | 150 | | |
|---|---|---|---|---|---|---|---|---|
| Parameter | Symbol | Min | Max | Min | Max | Min | Max | Units |
| Read Cycle Time | $t_{RC}$ | 90 | - | 120 | - | 150 | - | ns |
| Address to Output Delay | $t_{ACC}$ | - | 90 | - | 120 | - | 150 | ns |
| Chip Select to Output | $t_{CE}$ | - | 90 | - | 120 | - | 150 | ns |
| Output Enable to Output | $t_{OE}$ | - | 40 | - | 50 | - | 55 | ns |
| Output Enable to Output High Z | $t_{DF}$ | - | 30 | - | 30 | - | 35 | ns |
| Output Hold From Address /CS or /OE whichever occurs first | $t_{OH}$ | 0 | - | 0 | - | 0 | - | ns |

## Erase/Program

| | | 90 | | | 120 | | | 150 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Parameter | Symbol | Min | Typ | Max | Min | Typ | Max | Min | Typ. | Max | Units |
| Write Cycle Time | $t_{WC}$ | 90 | - | - | 120 | - | - | 150 | - | - | ns |
| Address Setup Time | $t_{AS}$ | 0 | - | - | 0 | - | - | 0 | - | - | ns |
| Address Hold Time | $t_{AH}$ | 45 | - | - | 50 | - | - | 55 | - | - | ns |
| Data Setup Time | $t_{DS}$ | 45 | - | - | 50 | - | - | 55 | - | - | ns |
| Data Hold Time | $t_{DH}$ | 0 | - | - | 0 | - | - | 0 | - | - | ns |
| Read Recover before Write | $t_{GHWL}$ | 0 | - | - | 0 | - | - | 0 | - | - | ns |
| /CS Setup Time | $t_{CS}$ | 0 | - | - | 0 | - | - | 0 | - | - | ns |
| /CS Hold Time | $t_{CH}$ | 0 | - | - | 0 | - | - | 0 | - | - | ns |
| /WE Pulse Width | $t_{WP}$ | 45 | - | - | 50 | - | - | 55 | - | - | ns |
| /WE Pulse Width High | $t_{WPH}$ | 30 | - | - | 30 | - | - | 30 | - | - | ns |
| Programming Operation[2] | $t_{WHWH1}$ | - | 9 | - | - | 9 | - | - | 9 | - | µs |
| Sector Erase Operation[2] | $t_{WHWH2}$ | - | - | 0.7 | - | - | 0.7 | - | - | 0.7 | s |
| $V_{CC}$ Setup Time | $t_{VCS}$ | 50 | - | - | 50 | - | - | 50 | - | - | µs |

*Notes : (1) Not 100% tested.*
*(2) This does not include the preprogramming time.*

## Erase/Program Alternate /CS controlled Writes

| | | 90 | | | 120 | | | 150 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Parameter | Symbol | Min | Typ | Max | Min | Typ | Max | Min | Typ. | Max | Units |
| Write Cycle Time[1] | $t_{WC}$ | 90 | - | - | 120 | - | - | 150 | - | - | ns |
| Address Setup Time | $t_{AS}$ | 0 | - | - | 0 | - | - | 0 | - | - | ns |
| Address Hold Time | $t_{AH}$ | 45 | - | - | 50 | - | - | 55 | - | - | ns |
| Data Setup Time | $t_{DS}$ | 45 | - | - | 50 | - | - | 55 | - | - | ns |
| Data Hold Time | $t_{DH}$ | 0 | - | - | 0 | - | - | 0 | - | - | ns |
| Read Recover before Write | $t_{GHEL}$ | 0 | - | - | 0 | - | - | 0 | - | - | ns |
| /WE Setup Time | $t_{WS}$ | 0 | - | - | 0 | - | - | 0 | - | - | ns |
| /WE Hold Time | $t_{WH}$ | 0 | - | - | 0 | - | - | 0 | - | - | ns |
| /CS Pulse Width | $t_{CP}$ | 45 | - | - | 50 | - | - | 55 | - | - | ns |
| /CS Pulse Width High | $t_{CPH}$ | 30 | - | - | 30 | - | - | 30 | - | - | ns |
| Programming Operation[2] | $t_{WHWH1}$ | - | 9 | - | - | 9 | - | - | 9 | - | µs |
| Sector Erase Operation[2] | $t_{WHWH2}$ | - | 0.7 | - | - | 0.7 | - | - | 0.7 | - | s |

*Notes : (1) Not 100% tested.*
*(2) This does not include the preprogramming time.*

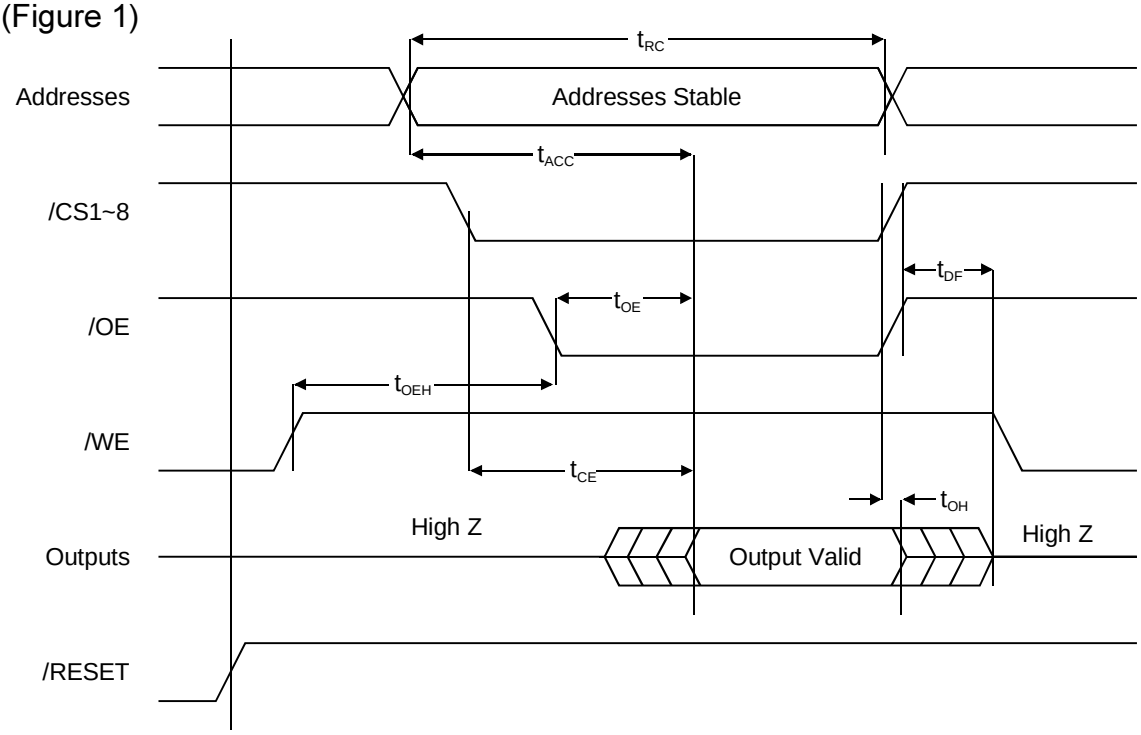
## Hardware Reset (/RESET)

| Parameter | Description | Test Setup | All Speed Options | Unit |
|---|---|---|---|---|
| $t_{READY}$ | /RESET Pin Low (NOT During Embedded Algorithms) to Read or Write* | Max | 500 | ns |
| $t_{RP}$ | /RESET Pulse Width | Min | 500 | ns |
| $t_{RH}$ | /RESET High Time before Read* | Min | 50 | ns |

**Note : Not 100% tested*

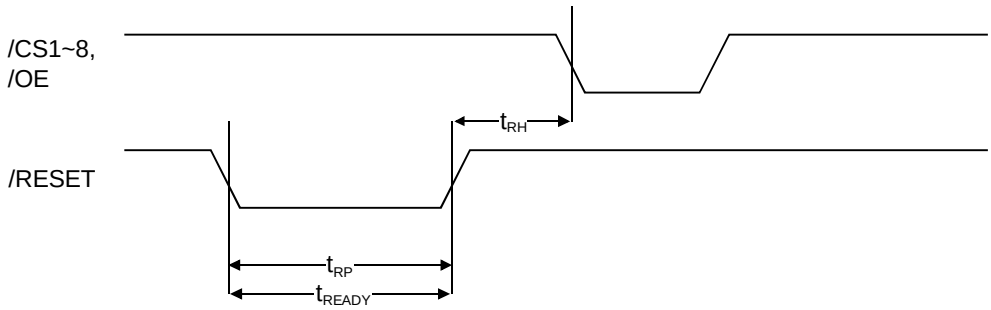
## Read Operations Timings

(Figure 1)



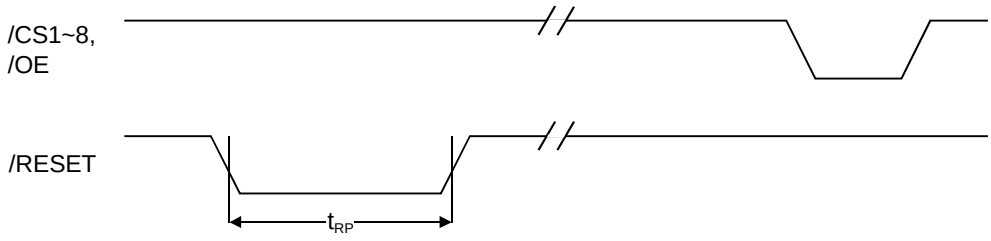

## RESET Timings

(Figure 2)



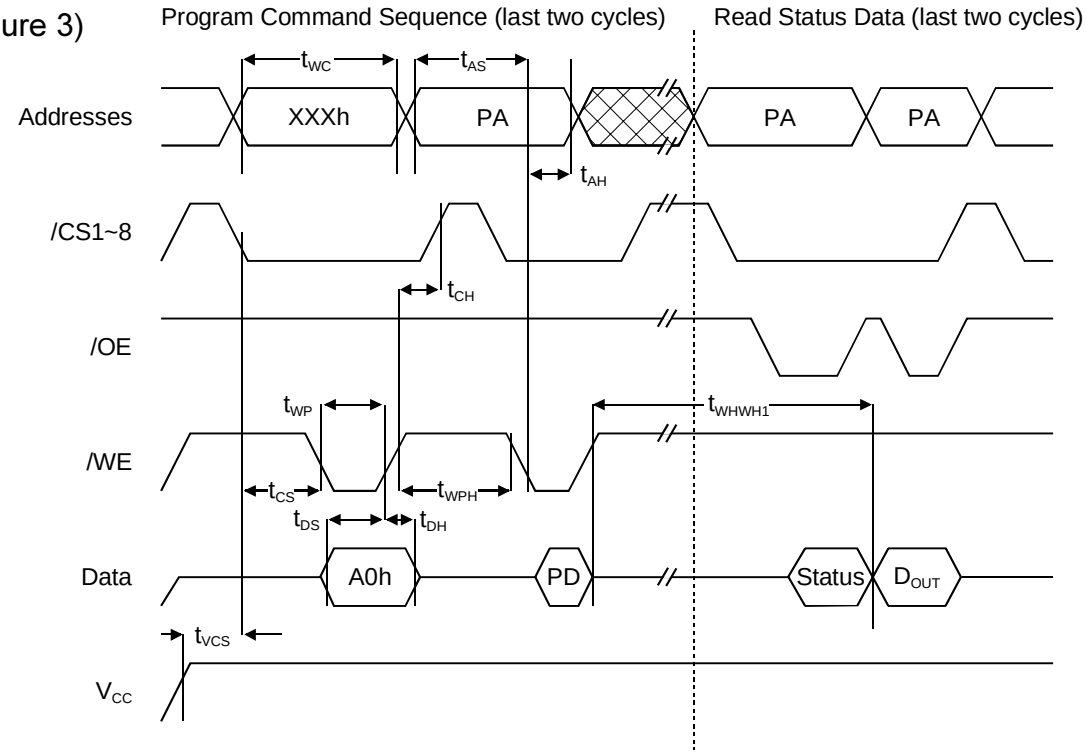

*Reset Timings NOT during Embedded Algorithms*

*Reset Timings during Embedded Algorithms*



## Program Operation Timings

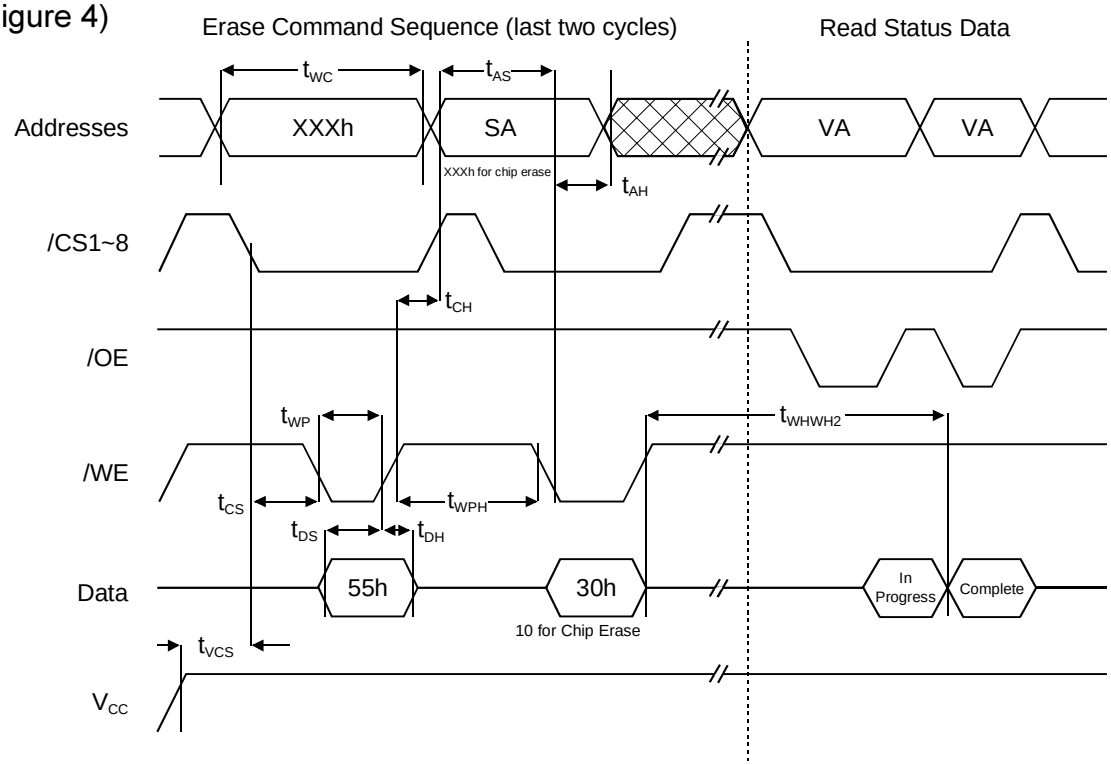
(Figure 3)




Note : PA = Program Address, PD = Program Data, DOUT is the true data at the program address.

## Chip/Sector Erase Operation Timings

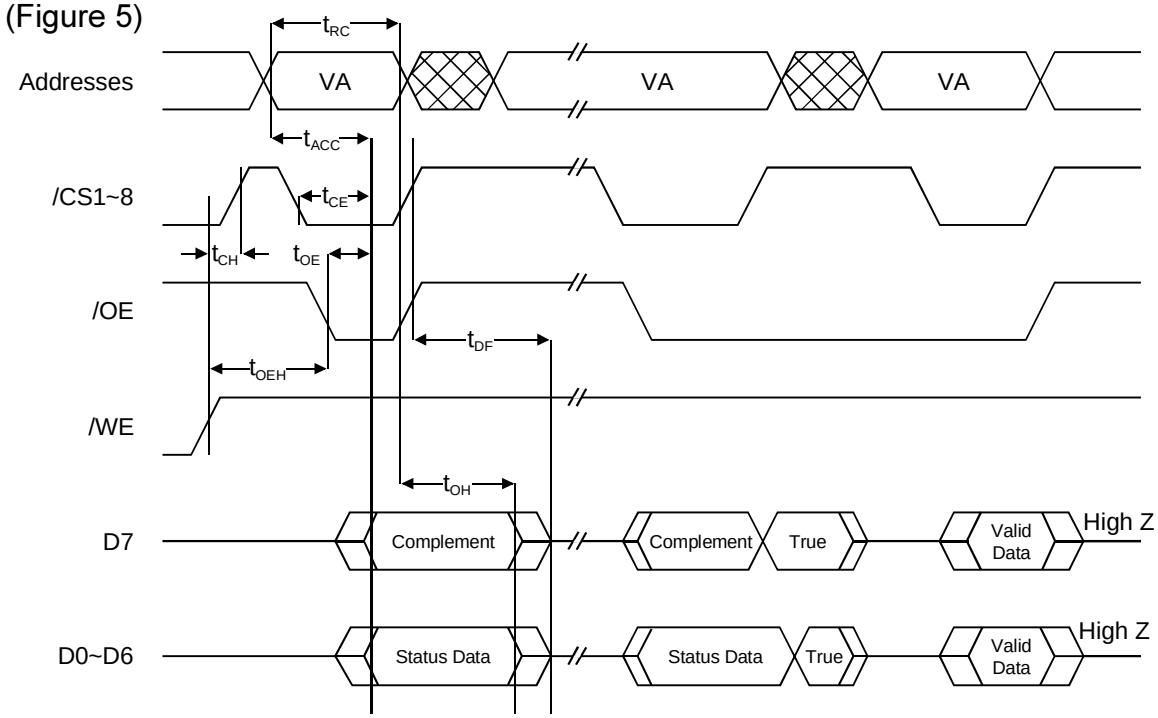
(Figure 4)




Note : SA=Sector Address. VA=Valid Address for reading status data.

## Data Polling Timings (During Embedded Algorithms)

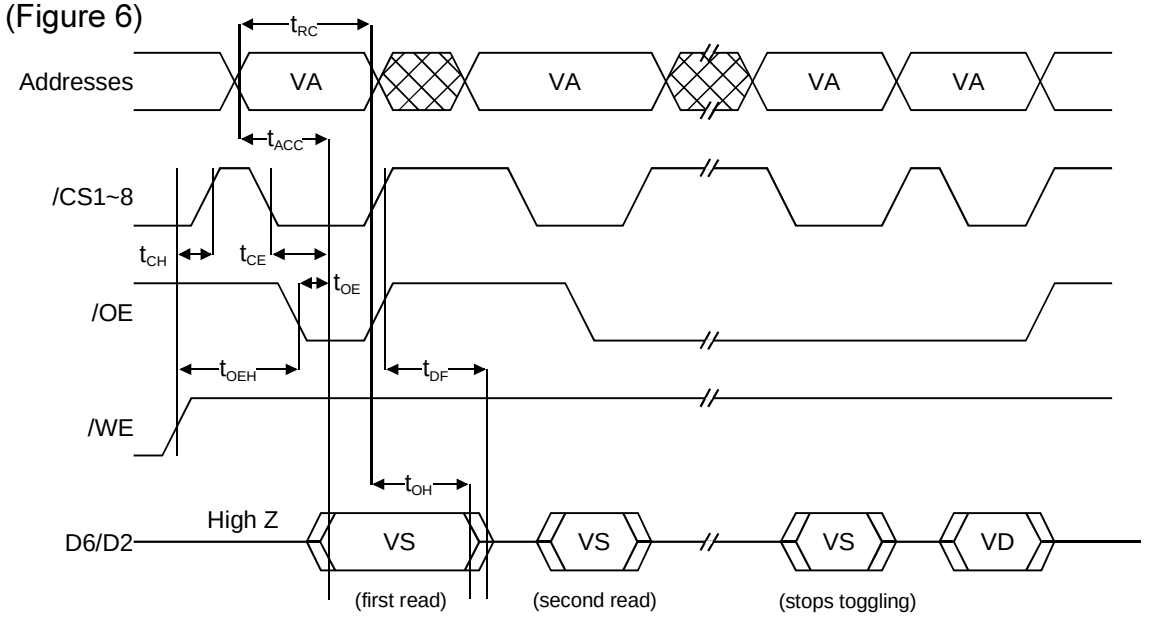
(Figure 5)




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

## Toggle Bit Timings (During Embedded Algorithms)

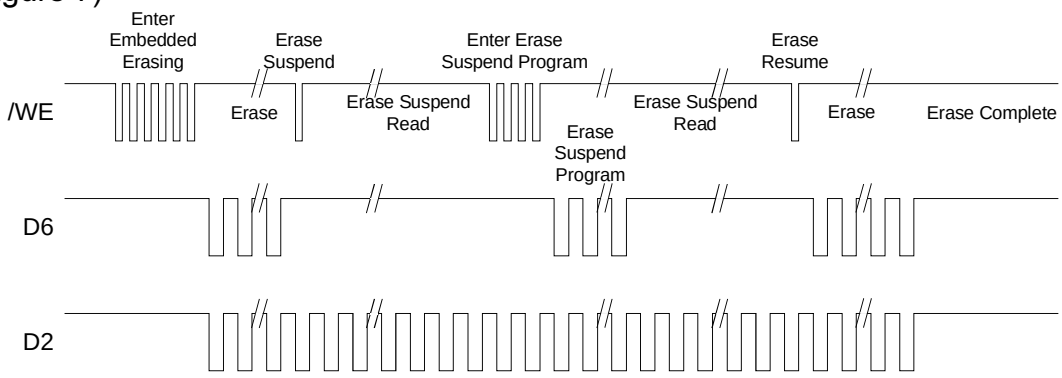
(Figure 6)




Note :
VA=Valid Address; not required for D6. Illustration shows first two status cycle after command sequence, last status read cycle, and array data read cycle.
VS=Valid Status, VD=Valid Data

## D2 vs D6

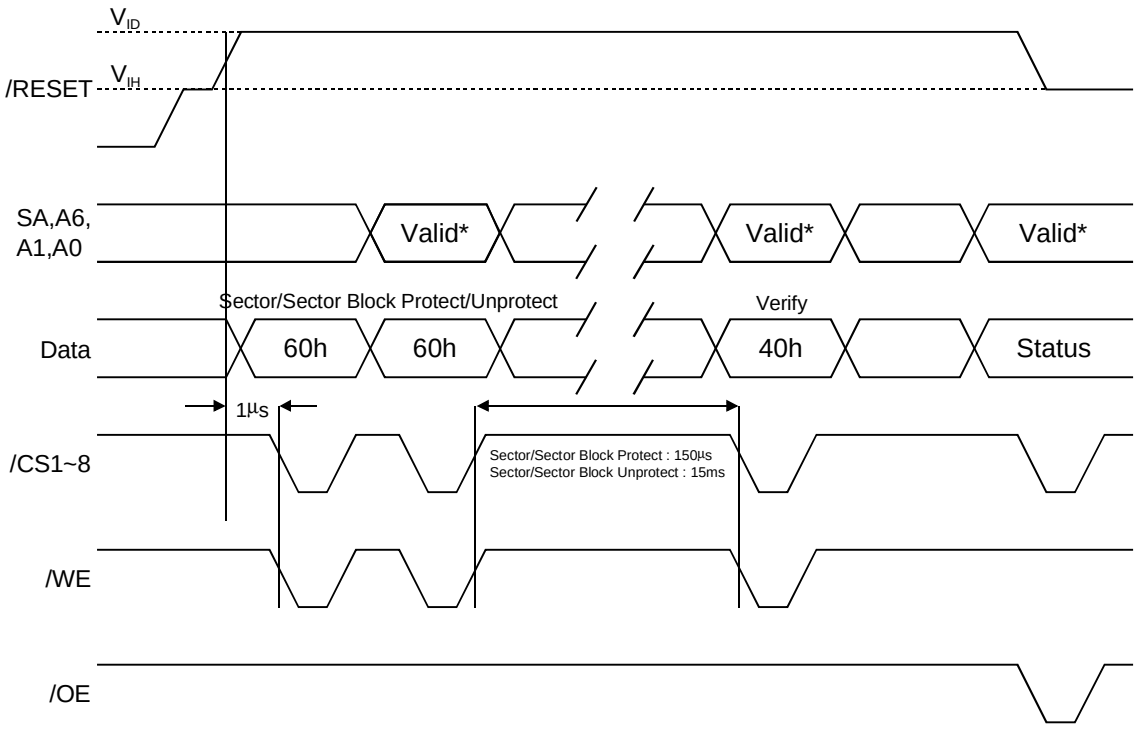
(Figure 7)




**Note :** The system can use /OE or /CS1~8 to toggle D2/D6. D2 toggles only when read at an address within an erase-suspended sector.

## Sector Protect/Unprotect Timing Diagram

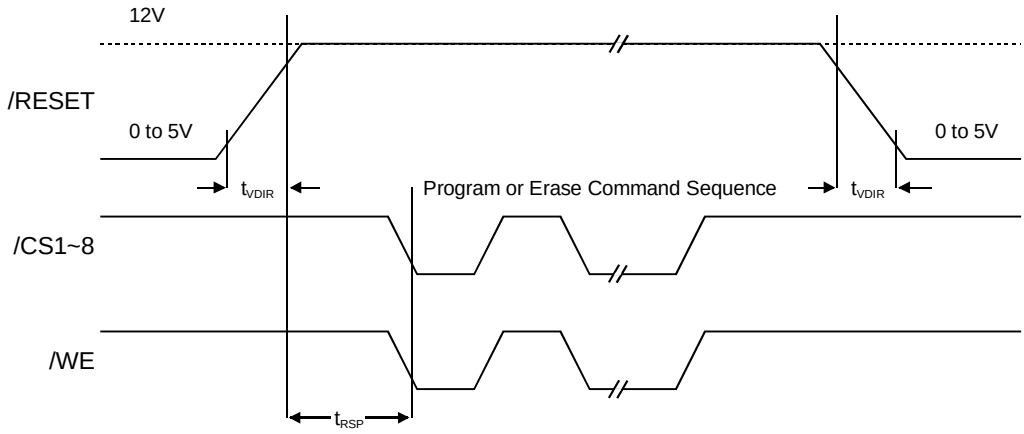
(Figure 8)




**Note :** For Sector Protect, A6=0, A1=1, A0=0. For Sector Unprotect, A6=1, A1=1, A0=0.

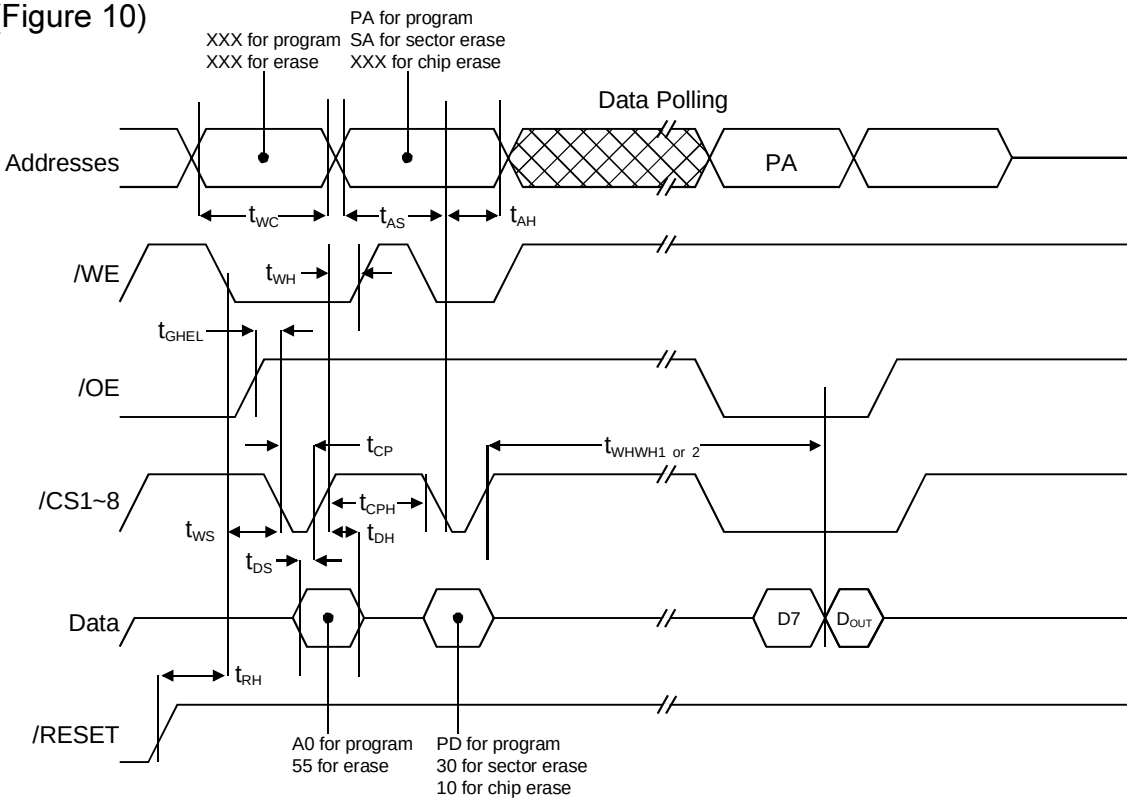
## Temporary Sector/Sector Block Unprotect Timing Diagram

(Figure 9)




## Alternate /CS Controlled Write Operation Timings

(Figure 10)

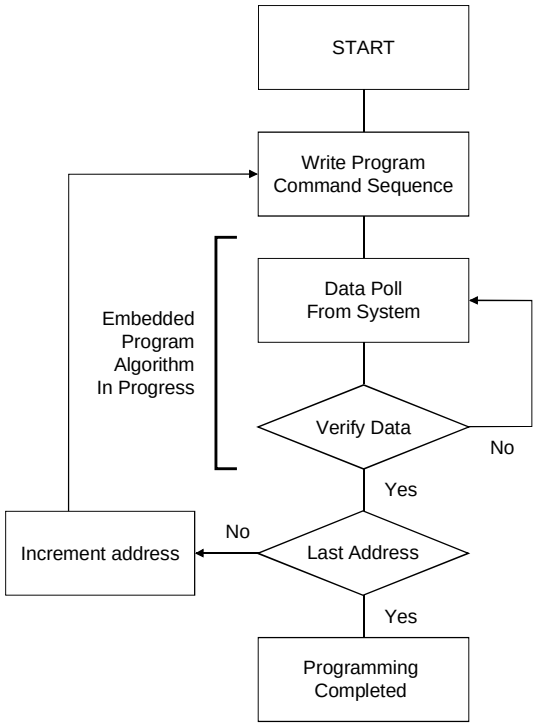



Notes:

1. PA =Program Address, PD=Program Data, SA=Sector Address, D7=Compliment of Data Input, $D_{OUT}$=Array Data.
2. Figure indicates the last two bus cycles of the command sequence.

## Program Operation

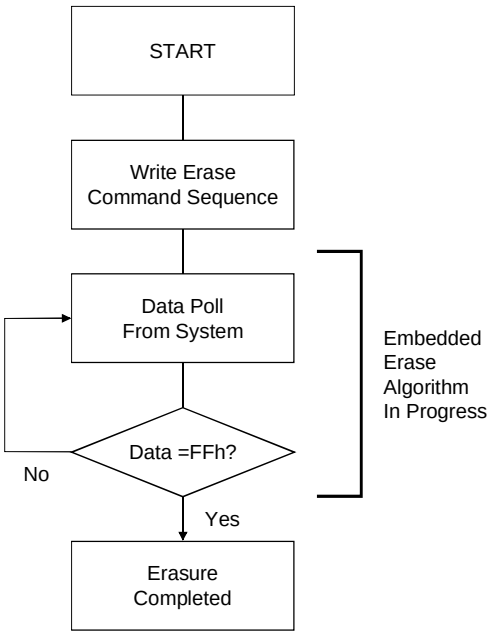
(Figure 11)




Note :
See the appropriate Command definitions table for program command sequence

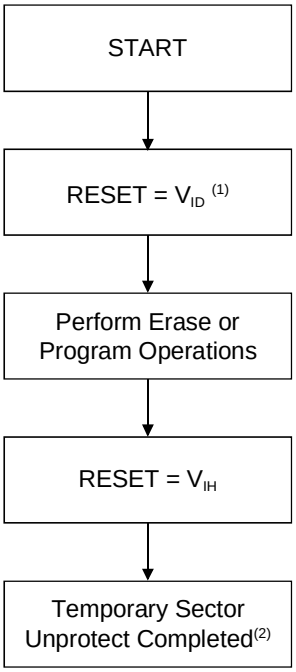
## Erase Operation

(Figure 12)




Note :
1. See the appropriate Command definitions table for erase command sequence.
2. See 'D3 : Sector Erase Timer' For more information.

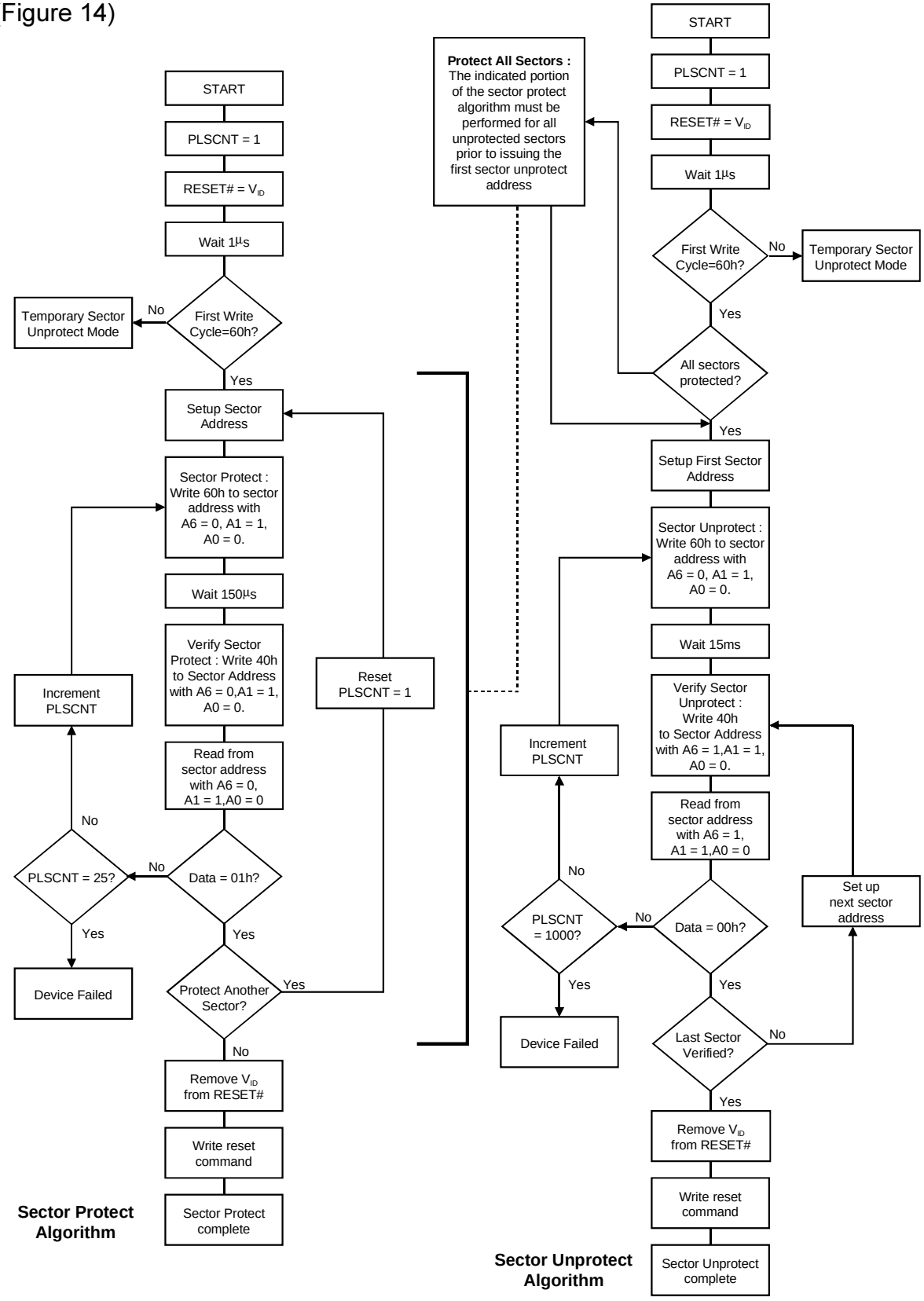
## Temporary Sector Unprotect Operation (Figure 13)




**Notes :**
1. All protected sectors unprotected.
2. All previously protected sectors are protected once again.

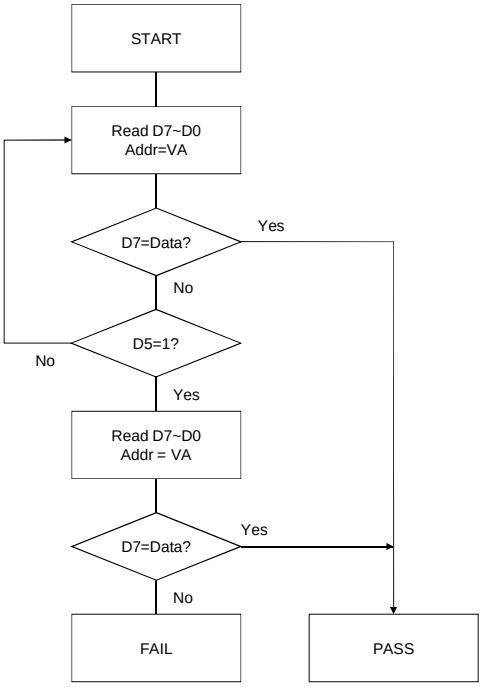
## In-System Sector Protect/Unprotect Algorithms

(Figure 14)



## Data Polling Algorithm

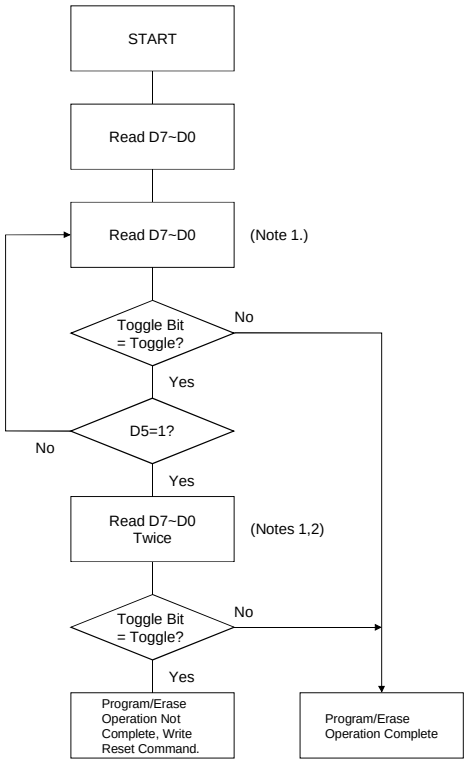
(Figure 15)




Note :
1. VA = Valid Address for programming. During a sector erase operartion, a valid adddress is an address within any sector selected for erasure. During chip erase, a valid address is any non-protected sector address.
2. D7 should be rechecked even if D5 = '1' becease D7 may change simultaneously with D5.

## Toggle Bit Algorithm

(Figure 16)




Note :
1. Read Toggle bit twice to determine whether or not it is toggling. See text.
2. Recheck toggle bit because it may stop toggling as D5 changes to '1' . See text.

This section describes the requirements and use ofthe 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 composed of latches that 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. Table 1 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.

## Requirements for Reading Array Data

To read array data from the outputs, the system must drive the /CS# and /OE pins to $V_{IL}$ . /CS# 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 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. The device remains enabled for read access until the command register contents are altered.

See “Reading Array Data” for more information. Refer to the AC Read Operations table for timing specifications and to Figure 1 for the timing diagram. $I_{CCR32}/I_{CCR16}/I_{CCR8}$ 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 /CS# to $V_{IL}$ , and /OE to $V_{IH}$ .

The device features an **Unlock Bypass** mode to facilitate faster programming. Once the device enters theUnlock Bypass mode, only two write cycles are required to program a byte, instead of four. The “Byte 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. Table indicates the address space that each sector occupies. A “sector address” consists of the address bits required to uniquely select a sector. The “Writing specific address and data commands or sequences into the command register initiates device operations. Table 9 defines the valid register command sequences. Writing incorrect address and data values or writing them in the improper sequence resets the device to reading array data.” section has details on erasing a sector or the entire chip, or suspending/resuming the erase operation.

After 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 D7–D0. Standard read cycle timings apply in this mode. Refer to the Autoselect Mode and Autoselect Command Sequence sections for more information. $I_{CCW32}/I_{CCW16}/I_{CCW8}$ in the DC Characteristics table represents the active current specification for the write mode. The "AC Characteristics" section contains timing specification tables and timing diagrams for write operations.

## Program and Erase Operation Status

During an erase or program operation, the system may check the status of the operation by reading the status bits on D7–D0. Standard read cycle timings and $I_{CCR32}/I_{CCR16}/I_{CCR8}$ read specifications apply. Refer to "Write Operation Status" for more information, and to "AC Characteristics" for timing diagrams.

## Standby Mode

When the system is not reading or writing to the device, it can place the device in the standby mode. Inthis 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 /CS# 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 /CS# 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.

The device also enters the standby mode when the /RESET pin is driven low. Refer to the next section, "/RESET: Hardware Reset Pin".

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

$I_{SB}$ in the DC Characteristics 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 i ndependent of the /CS#, /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. $I_{CCRES}$ in the DC Characteristics 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 GND ± 0.3V, the device draws CMOS standby current ($I_{CCRES}$). If /RESET is held at $V_{IL}$ but not within GND ± 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.

Refer to the AC Characteristics tables for /RESET parameters and to Figure 2 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.

## Autoselect Mode

The autoselect mode provides manufacturer and device identification, and sector protection verification, through identifier codes output on D7–D0. This mode is primarily intended for programming equipment to automatically match a device to be programmed with its corresponding programming algorithm. However, the autoselect codes can also be accessed in-system through the command register.

When using programming equipment, the autoselect mode requires $V_{ID}$ (11.5 V to 12.5 V) on address pin A9. Address pins A6, A1, and A0 must be as shown in Table 3. In addition, when verifying sector protection, the sector address must appear on the appropriate highest order address bits (see Table ). Table 3 shows the remaining address bits that are don't care. When all necessary bits have been set as required, the programming equipment may then read the corresponding identifier code on D7-D0.

To access the autoselect codes in-system, the host system can issue the autoselect command via the command register, as shown in Table 9. This method does not require $V_{ID}$ . See "Writing specific address and data commands or sequences into the command register initiates device operations. Table 9 defines the valid register command sequences. Writing incorrect address and data values or writing them in the improper sequence resets the device to reading array data." for details on using the autoselect mode.

## 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 Table 4).

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.

The primary method requires $V_{ID}$ on the /RESET pin only, and can be implemented either in-system or via programming equipment. Figure 14 shows the algorithms and Figure 8 shows the timing diagram. This method uses standard microprocessor bus cycleing. For sector unprotect, all unprotected sectors must first be protected prior to the first sector unprotect write cycle.

The alternate method intended only for programming equipment requires $V_{ID}$ on address pin A9 and /OE. The device is shipped with all sectors unprotected. It is possible to determine whether a sector is protected or unprotected. See "Autoselect Mode" for details.

## Temporary Sector/Sector Block 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 4).

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 13 shows the algorithm, and Figure 9 shows the timing diagrams, for this feature.

## Hardware Data Protection

The command sequence requirement of unlock cycles for programming or erasing provides data protection against inadvertent writes (refer to Table 9 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. 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 5ns (typical) on /OE, /CS# or /WE do not initiate a write cycle.

## Logical Inhibit

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

## Power-Up Write Inhibit

If /WE = /CS# = $V_{IL}$ and /OE = $V_{IH}$ during power up, the device does not accept commands on the rising pro-edge of /WE. The internal state machine is automatically reset to reading array data on power-up.

## COMMON FLASH MEMORY INTERFACE (CFI)

The Common Flash Interface (CFI) specification out-lines 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, any time the device is ready to read array data. The system can read CFI information at the addresses given in Tables 5–8. To terminate reading CFI data, the system must write the reset command.

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 5–8. The system must write the reset command to return the device to the autoselect mode.

Writing specific address and data commands or sequences into the command register initiates device operations. Table 9 defines the valid register command sequences. Writing **incorrect address and data values** or writing them in the **improper sequence** resets the device to reading array data. All addresses are latched on the falling edge of /WE# or /CS#, whichever happens later. All data is latched on the rising edge of /WE or /CS#, whichever happens first. Refer to the appropriate timing diagrams in the "AC Characteristics" section.

## Reading Array Data

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

After the device accepts an Erase Suspend command, the device enters the Erase Suspend mode. The system can read array data using the standard read timings, 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 "Erase Suspend/Erase Resume Commands" for more information on this mode.

The system *must* issue the reset command to re-enable the device for reading array data if D5 goes high, or while in the autoselect mode. See the "Reset Command" section, next.

See also "Requirements for Reading Array Data" in the "Device Bus Operations" section for more information. The Read Operations table provides the read parameters, and Figure 1 shows the timing diagram.

## Reset Command

Writing the reset command to the device resets the device to reading array data. Address bits are don't care for this command.

The reset command may be written between the sequence cycles in an erase command sequence before erasing begins. This resets the device to reading array data. 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 device to reading array data (also applies to programming in Erase Suspend 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 reading array data (also applies to autoselect during Erase Suspend).

If D5 goes high during a program or erase operation, writing the reset command returns the device to reading array data (also applies during Erase Suspend).

## Autoselect Command Sequence

The autoselect command sequence allows the host system to access the manufacturer and devices codes, and determine whether or not a sector is protected. Table 9 shows the address and data requirements. This method is an alternative to that shown in Table 3, which is intended for PROM programmers and requires $V_{ID}$ on address bit A9.

The autoselect command sequence is initiated by writing two unlock cycles, followed by the autoselect command. The device then enters the autoselect mode, and the system may read at any address any number of times, without initiating another command sequence. A read cycle at address XX00h retrieves the manufacturer code. A read cycle at address XX01h returns the device code. A read cycle containing a sector address (SA) and the address 02h returns 01h if that sector is protected, or 00h if it is unprotected. Refer to Table for valid sector addresses.

The system must write the reset command to exit the autoselect mode and return to reading array data.

## Byte Program Command Sequence

The device programs one byte of data for each program operation. The command sequence requires four bus cycles, and 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 generates the program pulses and verifies the programmed cell margin. Table 9 shows the address and data requirements for the byte program command sequence.

When the Embedded Program algorithm is complete, the device then returns to reading array data and addresses are no longer latched. The system can determine the status of the program operation by using D7 or D6. See “Write Operation Status” 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 programming operation. The Byte Program command sequence should be reinitiated once the device has reset to reading array data, to ensure data integrity. Programming is allowed in any sequence and across sector boundaries. **A bit cannot be programmed from a “0” back to a “1”.** Attempting to do so may halt the operation and set D5 to “1,” or cause the Data Polling algorithm 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 to the device 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. The device 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. Table 9 shows the requirements for the command sequence.

During the unlock bypass mode, only the Unlock By-pass 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 data 90h; the second cycle the data 00h. Addresses are don't cares for both cycles. The device then returns to reading array data.

## 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. Table 9 shows the address and data requirements for the chip erase command sequence.

Any commands written to the chip during the Embedded Erase algorithm are ignored. Note that a **hardware reset** during the chip erase operation immediately terminates the operation. The Chip Erase command sequence should be reinitiated once the device has returned to reading array data, to ensure data integrity.

The system can determine the status of the erase operation by using D7, D6 or D2. See "Write Operation Status" for information on these status bits. When the Embedded Erase algorithm is complete, the device returns to reading array data and addresses are no longer latched.

Figure 12 illustrates the algorithm for the erase operation. See the Erase Program Operations tables in "AC Characteristics" for parameters, and to Figure 4 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 write cycles are then followed by the address of the sector to be erased, and the sector erase command. Table 9 shows the address and data requirements for the sector erase command sequence.

The device does *not* require the system to preprogram the memory prior to erase. The Embedded Erase algorithm automatically programs and verifies the sector 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 50 µs begins. 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 50µs, otherwise the last address and command might not be accepted, and erasure may begin. 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. If the time between additional sector erase commands can be assumed to be less than 50 µs, the system need not monitor D3. **Any command other than Sector Erase or Erase Suspend during the time-out period resets the device to reading array data.** The system must rewrite the command sequence and any additional sector addresses and commands.

The system can monitor D3 to determine if the sector erase timer has timed out. (See the “D3: Sector Erase Timer” section.) The time-out begins from the rising edge of the final /WE pulse in the command sequence.

Once the sector erase operation has begun, only the Erase Suspend command is valid. All other commands are ignored. Note that a **hardware reset** during the sector erase operation immediately terminates the operation. The Sector Erase command sequence should be reinitiated once the device has returned to reading array data, to ensure data integrity.

When the Embedded Erase algorithm is complete, the device returns to reading array data and addresses are no longer latched. The system can determine the status of the erase operation by using D7, D6 or D2. (Refer to “Write Operation Status” for information on these status bits.)

Figure 12 illustrates the algorithm for the erase operation. Refer to the Erase/Program Operations tables in the “AC Characteristics” section for parameters, and to Figure 4 for timing diagrams.

## Erase Suspend/Erase Resume Commands

The Erase Suspend command allows the system to interrupt a sector erase operation and then read data from, or program data to, any sector not selected for erasure. This command is valid only during the sector erase operation, including the time-out period 50 µs during the sector erase command sequence. The Erase Suspend command is ignored if written during the chip erase operation or Embedded Program algorithm. Writing the Erase Suspend command during the Sector Erase time-out immediately terminates the time-out period and suspends the erase operation. Addresses are “don’t-cares” when writing the Erase Suspend command.

When the Erase Suspend command is written during a 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.

After the erase operation has been suspended, the system can read array data from or program data to any sector not selected for erasure. (The device “erase suspends” all sectors selected for erasure.) Normal read and write timings and command definitions apply. Reading at any address within erase-suspended sectors produces status data on D7–D0. The system can use D7, or D6 and D2 together, to determine if a sect or is actively erasing or is erase-suspended. See “Write Operation Status” for information on these status bits.

After an erase-suspended program operation is complete, the system can once again read array data within non-suspended sectors. The system can deter mine the status of the program operation using the D7 or D6 status bits, just as in the standard program operation. See “Write Operation Status” for more information.

The system may also write the autoselect command sequence when the device is in the Erase Suspend mode. 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. See “Autoselect Command Sequence” for more information.

The system must write the Erase Resume command (address bits are “don’t care”) to exit the erase suspend mode and continue the sector erase operation. Further writes of the Resume command are ignored. Another Erase Suspend command can be written after the device has resumed erasing.

The device provides several bits to determine the status of a write operation: D2, D3, D5, D6 and D7. Table 10 and the following subsections describe the functions of these bits. D7 and D6 each offer a method for determining whether a program or erase operation is complete or in progress. These three bits are discussed first.

## D7: Data Polling

The Data Polling bit, D7, indicates to the host system whether an Embedded Algorithm is in progress or completed, or whether the device is in Erase Suspend. Data Polling is valid after the rising edge of the final /WE pulse in the program or erase command sequence.

During the Embedded Program algorithm, the device outputs on D7 the complement of the datum programmed to D7. This D7 status also applies to programming during Erase Suspend. When the Embedded Program algorithm is complete, the device outputs the datum programmed to D7. The system must provide the program address to read valid status information on D7. If a program address falls within a protected sector, Data Polling on D7 is active for approximately 1 µs, then the device returns to reading array data.

During the Embedded Erase algorithm, Data Polling produces a “0” on D7. When the Embedded Erase algorithm is complete, or if the device enters the Erase Suspend mode, Data Polling produces a “1” on D7. This is analogous to the complement/true datum output described for the Embedded Program algorithm: the erase function changes all the bits in a sector to “1”; prior to this, the device outputs the “complement,” or “0.” The system must provide an address within any of the sectors selected for erasure to read valid status information on D7.

After an erase command sequence is written, if all sectors selected for erasing are protected, Data Polling on D7 is active for approximately 100µs, then the device 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.

When the system detects D7 has changed from the complement to true data, it can read valid data at D7–D0 on the *following* read cycles. This is because D7 may change asynchronously with D0–D6 while Output Enable (/OE) is asserted low. Figure 5, Data Polling Timings (During Embedded Algorithms), in the “AC Characteristics” section illustrates this.Table 10 shows the outputs for Data Polling on D7. Figure 15 shows the Data Polling algorithm.

## D6: Toggle Bit I

Toggle Bit I on D6 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 D6 to toggle (The system may use either /OE or/CS to control the read cycles). When the operation is complete, D6 stops toggling.

After an erase command sequence is written, if all sectors selected for erasing are protected, D6 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 D6 and D2 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), D6 toggles. When the device enters the Erase Suspend mode, D6 stops toggling. However, the system must also use D2 to determine which sectors are erasing or erase-suspended. Alternatively, the system can use D7 (see the subsection on D7: Data Polling).

If a program address falls within a protected sector, D6 toggles for approximately 1 µs after the program command sequence is written, then returns to reading array data.

D6 also toggles during the erase-suspend-program mode, and stops toggling once the Embedded Program algorithm is complete.

Table 10 shows the outputs for Toggle Bit I on D6. Figure 16 shows the toggle bit algorithm in flowchart form, and the section “Reading Toggle Bits D6/D2” explains the algorithm. Figure 6 in the “AC Characteristics” section shows the toggle bit timing diagrams. Figure 7 shows the differences between D2 and D6 in graphical form. See also the subsection on D2: Toggle Bit II.

## D2: Toggle Bit II

The “Toggle Bit II” on D2, when used with D6, 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.

D2 toggles when the system reads at addresses within those sectors that have been selected for erasure. (The system may use either /OE or /CS# to control the read cycles.) But D2 cannot distinguish whether the sector is actively erasing or is erase-suspended. D6, 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 10 to compare outputs for D2 and D6.

Figure 16 shows the toggle bit algorithm in flowchart form, and the section “Reading Toggle Bits D6/D2” explains the algorithm. See also the D6: Toggle Bit I subsection. Figure 6 shows the toggle bit timing diagram. Figure 7 shows the differences between D2 and D6 in graphical form.

## Reading Toggle Bits D6/D2

Refer to Figure 16 for the following discussion. Whenever the system initially begins reading toggle bit status, it must read D7–D0 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 D7–D0 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 D5 is high (see the section on D5). 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 D5 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 D5 has not gone high. The system may continue to monitor the toggle bit and D5 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 16).

Table 10 shows the outputs for Toggle Bit I on D6. Figure 16 shows the toggle bit algorithm. Figure 6 in the “AC Characteristics” section shows the toggle bit timing diagrams. Figure 7 shows the differences between D2 and D6 in graphical form. See also the subsection on D2: Toggle Bit II.

## D5: Exceeded Timing Limits

D5 indicates whether the program or erase time has exceeded a specified internal pulse count limit. Under these conditions D5 produces a “1.” This is a failure condition that indicates the program or erase cycle was not successfully completed.

The D5 failure condition may appear if the system tries to program a “1” to a location that is 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 operation has exceeded the timing limits, D5 produces a “1” .

Under both these conditions, the system must issue the reset command to return the device to reading array data.

## D3: Sector Erase Timer

After writing a sector erase command sequence, the system may read D3 to determine whether or not an erase operation 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 is complete, D3 switches from “0” to “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 D3. See also the “Sector Erase Command Sequence” section.

After the sector erase command sequence is written, the system should read the status on D7 (Data Polling) or D6 (Toggle Bit I) to ensure the device has accepted the command sequence, and then read D3. If D3 is “1”, the internally controlled erase cycle has begun; all further commands (other than Erase Suspend) are ignored until the erase operation is complete.If D3 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 D3 prior to and following each subsequent sector erase command. If D3 is high on the second status check, the last command might not have been accepted. Table 10 shows the outputs for D3.

## Table 1 - Device Bus Operations

| Operation | /CS# | /OE | /WE | /RESET | Addresses | D0~D7 |
|---|---|---|---|---|---|---|
| Read | L | L | H | H | $A_{IN}$ | $D_{OUT}$ |
| Write | L | H | L | H | $A_{IN}$ | $D_{IN}$ |
| Standby | $V_{CC}$±0.3V | X | X | $V_{CC}$±0.3V | X | HIGH-Z |
| Output Disable | L | H | H | H | X | HIGH-Z |
| Hardware Reset | X | X | X | L | X | HIGH-Z |
| Sector/Sector Block Protect(1) | L | H | L | $V_{ID}$ | SA, A6=L, A1=H, A0=L | $D_{IN}$, $D_{OUT}$ |
| Sector/Sector Block Unprotect(1) | L | H | L | $V_{ID}$ | SA, A6=H, A1=H, A0=L | $D_{IN}$, $D_{OUT}$ |
| Temporary Sector/Sector Block Unprotect | X | X | X | $V_{ID}$ | $A_{IN}$ | $D_{IN}$ |

***Legend:***

*L = Logic Low = $V_{IL}$ , H = Logic High = $V_{IH}$ , $V_{ID}$ = 12.0 ± 0.5 V, X = Don't Care, $A_{IN}$ = Address In, $D_{IN}$ = Data In, $D_{OUT}$ = Data Out, SA = Sector Addresses.*

***Notes:***

1. *The sector protect and sector unprotect functions may also be implemented via programming equipment. See the "Sector/Sector Block Protection and Unprotection" section.*

## Table 2 - Sector Address Table

| Sector | A21 | A20 | A19 | A18 | A17 | A16 | Address Range |
|---|---|---|---|---|---|---|---|
| SA0 | 0 | 0 | 0 | 0 | 0 | 0 | 000000~00FFFF |
| SA1 | 0 | 0 | 0 | 0 | 0 | 1 | 010000~01FFFF |
| SA2 | 0 | 0 | 0 | 0 | 1 | 0 | 020000~02FFFF |
| SA3 | 0 | 0 | 0 | 0 | 1 | 1 | 030000~03FFFF |
| SA4 | 0 | 0 | 0 | 1 | 0 | 0 | 040000~04FFFF |
| SA5 | 0 | 0 | 0 | 1 | 0 | 1 | 050000~05FFFF |
| SA6 | 0 | 0 | 0 | 1 | 1 | 0 | 060000~06FFFF |
| SA7 | 0 | 0 | 0 | 1 | 1 | 1 | 070000~07FFFF |
| SA8 | 0 | 0 | 1 | 0 | 0 | 0 | 080000~08FFFF |
| SA9 | 0 | 0 | 1 | 0 | 0 | 1 | 090000~09FFFF |
| SA10 | 0 | 0 | 1 | 0 | 1 | 0 | 0A0000~0AFFFF |
| SA11 | 0 | 0 | 1 | 0 | 1 | 1 | 0B0000~0BFFFF |
| SA12 | 0 | 0 | 1 | 1 | 0 | 0 | 0C0000~0CFFFF |
| SA13 | 0 | 0 | 1 | 1 | 0 | 1 | 0D0000~0DFFFF |
| SA14 | 0 | 0 | 1 | 1 | 1 | 0 | 0E0000~0EFFFF |
| SA15 | 0 | 0 | 1 | 1 | 1 | 1 | 0F0000~0FFFFF |
| SA16 | 0 | 1 | 0 | 0 | 0 | 0 | 100000~10FFFF |
| SA17 | 0 | 1 | 0 | 0 | 0 | 1 | 110000~11FFFF |
| SA18 | 0 | 1 | 0 | 0 | 1 | 0 | 120000~12FFFF |
| SA19 | 0 | 1 | 0 | 0 | 1 | 1 | 130000~13FFFF |
| SA20 | 0 | 1 | 0 | 1 | 0 | 0 | 140000~14FFFF |
| SA21 | 0 | 1 | 0 | 1 | 0 | 1 | 150000~15FFFF |
| SA22 | 0 | 1 | 0 | 1 | 1 | 0 | 160000~16FFFF |
| SA23 | 0 | 1 | 0 | 1 | 1 | 1 | 170000~17FFFF |
| SA24 | 0 | 1 | 1 | 0 | 0 | 0 | 180000~18FFFF |
| SA25 | 0 | 1 | 1 | 0 | 0 | 1 | 190000~19FFFF |
| SA26 | 0 | 1 | 1 | 0 | 1 | 0 | 1A0000~1AFFFF |
| SA27 | 0 | 1 | 1 | 0 | 1 | 1 | 1B0000~1BFFFF |
| SA28 | 0 | 1 | 1 | 1 | 0 | 0 | 1C0000~1CFFFF |
| SA29 | 0 | 1 | 1 | 1 | 0 | 1 | 1D0000~1DFFFF |
| SA30 | 0 | 1 | 1 | 1 | 1 | 0 | 1E0000~1EFFFF |
| SA31 | 0 | 1 | 1 | 1 | 1 | 1 | 1F0000~1FFFFF |

*Continued overleaf.*

## Table 2 - Sector Address Table (continued)

| Sector | A21 | A20 | A19 | A18 | A17 | A16 | Address Range |
|---|---|---|---|---|---|---|---|
| SA32 | 1 | 0 | 0 | 0 | 0 | 0 | 200000~20FFFF |
| SA33 | 1 | 0 | 0 | 0 | 0 | 1 | 210000~21FFFF |
| SA34 | 1 | 0 | 0 | 0 | 1 | 0 | 220000~02FFFF |
| SA35 | 1 | 0 | 0 | 0 | 1 | 1 | 230000~23FFFF |
| SA36 | 1 | 0 | 0 | 1 | 0 | 0 | 240000~24FFFF |
| SA37 | 1 | 0 | 0 | 1 | 0 | 1 | 250000~25FFFF |
| SA38 | 1 | 0 | 0 | 1 | 1 | 0 | 260000~26FFFF |
| SA39 | 1 | 0 | 0 | 1 | 1 | 1 | 070000~27FFFF |
| SA40 | 1 | 0 | 1 | 0 | 0 | 0 | 280000~28FFFF |
| SA41 | 1 | 0 | 1 | 0 | 0 | 1 | 290000~29FFFF |
| SA42 | 1 | 0 | 1 | 0 | 1 | 0 | 2A0000~2AFFFF |
| SA43 | 1 | 0 | 1 | 0 | 1 | 1 | 2B0000~2BFFFF |
| SA44 | 1 | 0 | 1 | 1 | 0 | 0 | 2C0000~2CFFFF |
| SA45 | 1 | 0 | 1 | 1 | 0 | 1 | 2D0000~2DFFFF |
| SA46 | 1 | 0 | 1 | 1 | 1 | 0 | 2E0000~2EFFFF |
| SA47 | 1 | 0 | 1 | 1 | 1 | 1 | 2F0000~2FFFFF |
| SA48 | 1 | 1 | 0 | 0 | 0 | 0 | 300000~30FFFF |
| SA49 | 1 | 1 | 0 | 0 | 0 | 1 | 310000~31FFFF |
| SA50 | 1 | 1 | 0 | 0 | 1 | 0 | 320000~32FFFF |
| SA51 | 1 | 1 | 0 | 0 | 1 | 1 | 330000~33FFFF |
| SA52 | 1 | 1 | 0 | 1 | 0 | 0 | 340000~34FFFF |
| SA53 | 1 | 1 | 0 | 1 | 0 | 1 | 350000~35FFFF |
| SA54 | 1 | 1 | 0 | 1 | 1 | 0 | 360000~36FFFF |
| SA55 | 1 | 1 | 0 | 1 | 1 | 1 | 370000~37FFFF |
| SA56 | 1 | 1 | 1 | 0 | 0 | 0 | 380000~38FFFF |
| SA57 | 1 | 1 | 1 | 0 | 0 | 1 | 390000~39FFFF |
| SA58 | 1 | 1 | 1 | 0 | 1 | 0 | 3A0000~3AFFFF |
| SA59 | 1 | 1 | 1 | 0 | 1 | 1 | 3B0000~3BFFFF |
| SA60 | 1 | 1 | 1 | 1 | 0 | 0 | 3C0000~3CFFFF |
| SA61 | 1 | 1 | 1 | 1 | 0 | 1 | 3D0000~3DFFFF |
| SA62 | 1 | 1 | 1 | 1 | 1 | 0 | 3E0000~3EFFFF |
| SA63 | 1 | 1 | 1 | 1 | 1 | 1 | 3F0000~3FFFFF |

## Table 3 - Autoselect Codes (High Voltage Method)

| Description | /CS# | /OE | /WE | A20 to A16 | A15 to A10 | A9 | A8 to A7 | A6 | A5 to A2 | A1 | A0 | D7 to D0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sector Protection Verification | L | L | H | SA | X | $V_{ID}$ | X | L | X | H | L | 01h (protected) |
| | | | | | | | | | | | | 00h (unprotected) |

*L = Logic Low = $V_{IL}$ , H = Logic High = $V_{IH}$ , SA = Sector Address, X = Don't care.*

## Table 4 - Sector Block Addresses for Protection/Unprotection.

| Sector/Sector Block | A21~A16 | Sector/Sector Block Size |
|---|---|---|
| SA0 | 000000 | 64 Kbytes |
| SA1~SA3 | 000001,000010,000011 | 192 (3 x 64) Kbytes |
| SA4~SA7 | 000100, 000101, 000110, 000111 | 256 (4 x 64) Kbytes |
| SA8~SA11 | 001000, 001001, 001010, 001011 | 256 (4 x 64) Kbytes |
| SA12~SA15 | 001100, 001101, 001110, 001111 | 256 (4 x 64) Kbytes |
| SA16~SA19 | 010000, 010001, 010010, 010011 | 256 (4 x 64) Kbytes |
| SA20~SA23 | 010100, 010101, 010110, 010111 | 256 (4 x 64) Kbytes |
| SA24~SA27 | 011000, 011001, 011010, 011011 | 256 (4 x 64) Kbytes |
| SA28~SA31 | 011100, 011101, 011110, 011111 | 256 (4 x 64) Kbytes |
| SA32~SA35 | 100000, 100001, 100010, 100011 | 256 (4 x 64) Kbytes |
| SA36~SA39 | 100100, 100101, 100110, 100111 | 256 (4 x 64) Kbytes |
| SA40~SA43 | 101000, 101001, 101010, 101011 | 256 (4 x 64) Kbytes |
| SA44~SA47 | 101100, 101101, 101110, 101111 | 256 (4 x 64) Kbytes |
| SA48~SA51 | 110000, 110001, 110010, 110011 | 256 (4 x 64) Kbytes |
| SA52~SA55 | 110100, 110101, 110110, 110111 | 256 (4 x 64) Kbytes |
| SA56~SA59 | 111000, 11101, 111010, 111011 | 256 (4 x 64) Kbytes |
| SA60~SA62 | 111100, 111101, 111110 | 192 (3 x 64) Kbytes |
| SA63 | 111111 | 64 Kbytes |

## Table 5 - CFI Query Identification String

| Addresses | Data | Description |
|---|---|---|
| 10h<br>11h<br>12h | 51h<br>52h<br>59h | Query Unique ASCII string “QRY” |
| 13h<br>14h | 02h<br>00h | Primary OEM Command Set |
| 15h<br>16h | 40h<br>00h | Address For Primary Extended Table |
| 17h<br>18h | 00h<br>00h | Alternate OEM Command Set (00h = None Exists) |
| 19h<br>1Ah | 00h<br>00h | Address for Alternate OEM Extended Table (00h = None Exists) |

## Table 6 - System Interface String

| Addresses | Data | Description |
|---|---|---|
| 1Bh | 27h | $V_{CC}$ Min. (write/erase) D7–D4: volt, D3–D0: 100 millivolt |
| 1Ch | 36h | $V_{CC}$ Max. (write/erase) D7–D4: volt, D3–D0: 100 millivolt |
| 1Dh | 00h | $V_{PP}$ Min. voltage (00h = no $V_{PP}$ pin present) |
| 1Eh | 00h | $V_{PP}$ Max. voltage (00h = no $V_{PP}$ pin present) |
| 1Fh | 04h | Typical timeout per single byte/word write $2^N$ µs |
| 20h | 00h | Typical timeout for Min. size buffer write $2^N$ µs (00h = not supported) |
| 21h | 0Ah | Typical timeout per individual block erase 2 N ms |
| 22h | 00h | Typical timeout for full chip erase $2^N$ ms (00h = not supported) |
| 23h | 05h | Max. timeout for byte/word write $2^N$ times typical |
| 24h | 00h | Max. timeout for buffer write $2^N$ times typical |
| 25h | 04h | Max. timeout per individual block erase $2^N$ times typical |
| 26h | 00h | Max. timeout for full chip erase $2^N$ times typical (00h = not supported) |

## Table 7 - Device Geometry Definition

| Addresses | Data | Description |
|---|---|---|
| 27h | 16h | Device Size = $2^N$ byte |
| 28h | 00h | FLASH Device Interface Description (refer to CFI Publication 100) |
| 29h | 00h | |
| 2Ah | 00h | Max. Number of Byte in multi-byte write = $2^N$ (00h = Not supported) |
| 2Bh | 00h | |
| 2Ch | 01h | Number of Erase Block Regions within Device. |
| 2Dh | 3Fh | Erase Block Region 1 Information (Refer to CFI specification or CFI publication 100) |
| 2Eh | 00h | |
| 2Fh | 00h | |
| 30h | 01h | |
| 31h | 00h | Erase Block Region 2 Information |
| 32h | 00h | |
| 33h | 00h | |
| 34h | 00h | |
| 35h | 00h | Erase Block Region 3 Information |
| 36h | 00h | |
| 37h | 00h | |
| 38h | 00h | |
| 39h | 00h | Erase Block Region 4 Information |
| 3Ah | 00h | |
| 3Bh | 00h | |
| 3Ch | 00h | |

## Table 8 - Primary Vendor Specific Extended Query

| Addresses | Data | Description |
|---|---|---|
| 40h<br>41h<br>42h | 50h<br>52h<br>49h | Query-unique ASCII string “PRI” |
| 43h | 31h | Major version number, ASCII |
| 44h | 30h | Minor version number, ASCII |
| 45h | 01h | **Address Sensitive Unlock** : 0 = Required, 1 = Not Required |
| 46h | 02h | **Erase Suspend**<br>0 = Not Supported, 1 = To Read Only, 2 = To Read & Write |
| 47h | 01h | **Sector Protect** : 0 = Not Supported, X = Number of Sectors in Group. |
| 48h | 04h | **Sector Temporary Unprotect** : 04 = Supported |
| 49h | 04h | **Sector Protect/Unprotect Scheme**<br>01 = 29F040 mode, 02 = 29F016 mode,<br>03 = 29F400 mode, 04 = 29LV800A mode. |
| 4Ah | 20h | **Simultaneous Operation** : 20 = Not Supported |
| 4Bh | 00h | **Burst Mode Type** : 00 = Not Supported, 01 = Supported |
| 4Ch | 00h | **Page Mode Type** :<br>00 = Not Supported, 01 = 4 Word Page, 02 = 8 Word Page |

## Table 9 - Command Definitions

| Command Sequence[1] | Cycles. | Bus Cycles [2-4] | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | First | | Second | | Third | | Fourth | | Fifth | | Sixth | |
| | | Addr | Data | Addr | Data | Addr | Data | Addr | Data | Addr | Data | Addr | Data |
| Read[5] | 1 | RA | RD | | | | | | | | | | |
| Reset[6] | 1 | XXX | F0 | | | | | | | | | | |
| Auto~select[7] SPV[9] | 4 | XXX | AA | XXX | 55 | 0XXXXX or 2XXXXX | 90 | (SA) X02 | 00 | | | | |
| | | XXX | | XXX | | | | | 01 | | | | |
| Byte Program | 4 | XXX | AA | XXX | 55 | XXX | A0 | PA | PD | | | | |
| Unlock Bypass | 3 | XXX | AA | XXX | 55 | XXX | 20 | | | | | | |
| Unlock Bypass Program[10] | 2 | XXX | A0 | PA | PD | | | | | | | | |
| Unlock Bypass Reset[11] | 2 | XXX | 90 | XXX | 00 | | | | | | | | |
| Chip Erase | 6 | XXX | AA | XXX | 55 | XXX | 80 | XXX | AA | XXX | 55 | XXX | 10 |
| Sector Erase | 6 | XXX | AA | XXX | 55 | XXX | 80 | XXX | AA | XXX | 55 | SA | 30 |
| Erase Suspend[12] | 1 | XXX | B0 | | | | | | | | | | |
| Erase Resume[13] | 1 | XXX | 30 | | | | | | | | | | |
| CFI Query [14] | 1 | XXX | 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 are latched on the falling edge of the /WE or /CS# pulse.*
***PD*** = *Data to be programmed at location PA. Data is latched on the rising edge of /WE or /CS# pulse.*
***SA*** = *Address of the sector to be erased or verified. Address bits A21–A16 uniquely select any sector.*
***SPV*** = *Sector Protect Verify.*

***Notes:***

1. *See Table 1 for descriptions of bus operations.*
2. *All values are in hexadecimal.*
3. *Except when reading array or autoselect data, all bus cycles are write operations.*
4. *Address bits are don't care for unlock and command cycles, except when PA or SA is required.*
5. *No unlock or command cycles required when device is in read mode.*
6. *The Reset command is required to return to the read mode when the device is in the autoselect mode or if D5 goes high.*
7. *The fourth cycle of the autoselect command sequence is a read cycle.*
8. *In the third and fourth cycles of the command sequence, set A21 to 0.*
9. *In the third cycle of the command sequence, address bit A21 must be set to 0 if verifying sectors 0–31, or to 1 if verifying sectors 32–64. The data in the fourth cycle is 00h for an unprotected sector/sector block and 01h for a protected sector/sector block.*
10. *The Unlock Bypass command is required prior to the Unlock Bypass Program command.*
11. *The Unlock Bypass Reset command is required to return to reading array data when the device is in the Unlock Bypass mode.*
12. *The system may read and program functions 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.*
13. *The Erase Resume command is valid only during the Erase Suspend mode.*
14. *Command is valid when device is ready to read array data or when device is in autoselect mode.*

## Table 10 - Write Operation Status

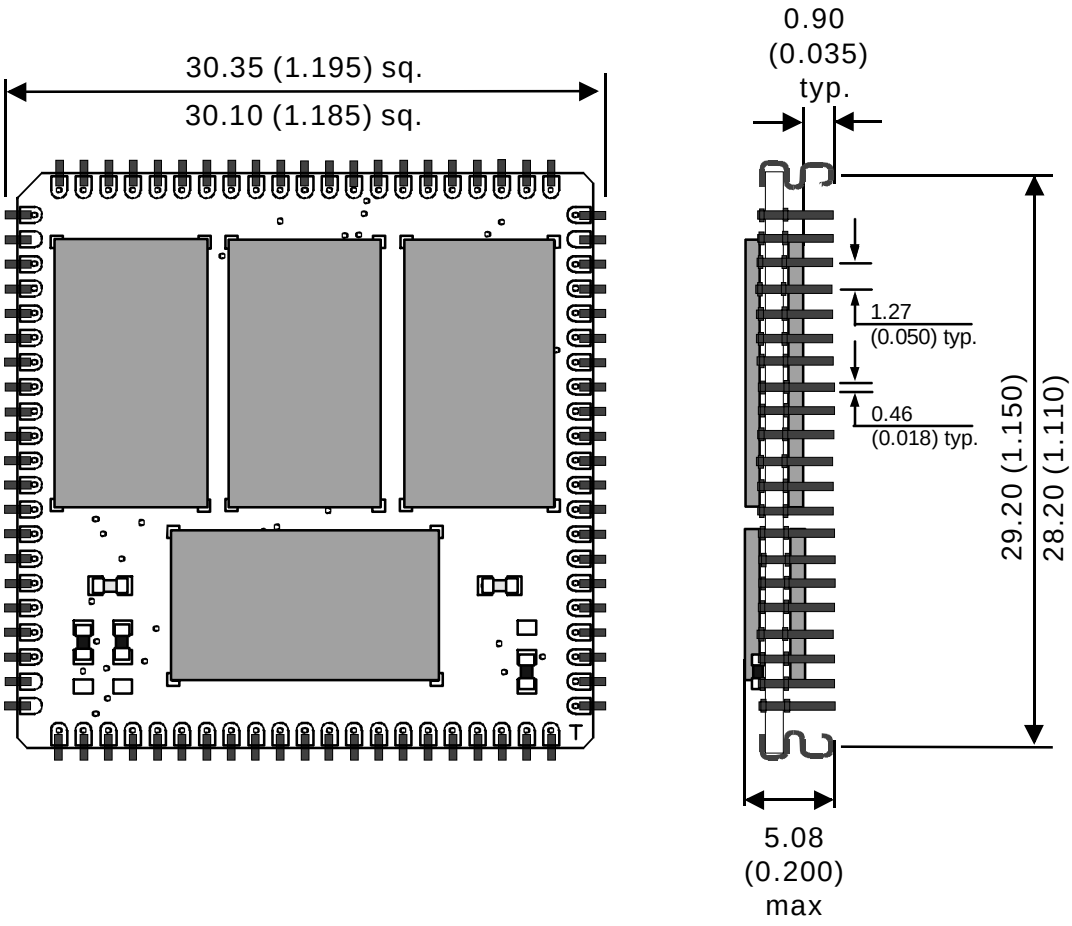
| Operation | | D7[2] | D6 | D5[1] | D3 | D2[2] |
|---|---|---|---|---|---|---|
| Standard Mode | Embedded Program Algorithm | D7 | Toggle | 0 | N/A | No Toggle |
| | Embedded Erase Algorithm | 0 | Toggle | 0 | 1 | Toggle |
| Erase Suspend Mode | Reading Within Erase Suspended Sector | 1 | No Toggle | 0 | N/A | Toggle |
| | Reading Within Non-Erase Suspended Sector | Data | Data | Data | Data | Data |
| | Erase-Suspend-Program | D7 | Toggle | 0 | N/A | N/A |

***Notes:***

1. *D5 switches to '1' when an Embedded Program or Embedded Erase operation has exceeded the maximum timing limits. See "D5: Exceeded Timing Limits" for more information.*
2. *D7 and D2 require a valid address when reading status information. Refer to the appropriate subsection for further details.*

## Package Details

PUMA 84 - Plastic 84 'J' Leaded Package.



## Ordering Information

**PUMA 84FV256006I - 90**

| | | |
|---|---|---|
| Speed | 90 = | 90ns |
| | 12 = | 120ns |
| | 15 = | 150ns |
| Temperature Range | Blank = | Commercial |
| | I = | Industrial |
| Memory Organisation | 256006 = | configurable as 8M x 32, 16M x 16 or 32M x 8 |
| Power Consumption | V = | 3.3V $\pm$ 10% $V_{CC}$ |
| Technology | F = | FLASH |
| Package | PUMA 84 = | Plastic 84 'J' Leaded Package |

Note :
Although this data is believed to be accurate the information contained herein is not intended to and does not create any warranty of merchantibility or fitness for a particular purpose.
Our products are subject to a constant process of development. Data may be changed without notice.
Products are not authorised for use as critical components in life support devices without the express written approval of a company director.