# 3 Volt Intel® StrataFlash™ Memory

## *28F128J3A, 28F640J3A, 28F320J3A (x8/x16)*

**Preliminary Datasheet**

## Product Features

- High-Density Symmetrically-Blocked Architecture
  - —128 128-Kbyte Erase Blocks (128 M)
  - —64 128-Kbyte Erase Blocks (64 M)
  - —32 128-Kbyte Erase Blocks (32 M)
- High Performance Interface Asynchronous Page Mode Reads
  - —100/25 ns Read Access Time (32 M)
  - —120/25 ns Read Access Time (64 M)
  - —150/25 ns Read Access Time (128 M)
- 2.7 V–3.6 V $V_{CC}$ Operation
- 128-bit Protection Register
  - —64-bit Unique Device Identifier
  - —64-bit User Programmable OTP Cells
- Enhanced Data Protection Features
  Absolute Protection with $V_{PEN}$ = GND
  - —Flexible Block Locking
  - —Block Erase/Program Lockout during Power Transitions
- Packaging
  - —56-Lead TSOP Package
  - —64-Ball Intel® Easy BGA Package
- Cross-Compatible Command Support Intel Basic Command Set
  - —Common Flash Interface
  - —Scalable Command Set
- 32-Byte Write Buffer
  - —6 µs per Byte Effective Programming Time
- 12,800,000 Total Erase Cycles (128 M)
  6,400,000 Total Erase Cycles (64 M)
  3,200,000 Total Erase Cycles (32 M)
  - —100,000 Erase Cycles per Block
- Automation Suspend Options
  - —Block Erase Suspend to Read
  - —Block Erase Suspend to Program
  - —Program Suspend to Read
- 0.25 µ Intel® StrataFlash™ Memory Technology

Capitalizing on Intel's 0.25 µ generation two-bit-per-cell technology, second generation Intel® StrataFlash™ memory products provide 2X the bits in 1X the space, with new features for mainstream performance. Offered in 128-Mbit (16-Mbyte), 64-Mbit, and 32-Mbit densities, these devices bring reliable, two-bit-per-cell storage technology to the flash market segment.

Benefits include: more density in less space, high-speed interface, lowest cost-per-bit NOR devices, support for code and data storage, and easy migration to future devices.

Using the same NOR-based ETOX™ technology as Intel's one-bit-per-cell products, Intel StrataFlash memory devices take advantage of over one billion units of manufacturing experience since 1987. As a result, Intel StrataFlash components are ideal for code and data applications where high density and low cost are required. Examples include networking, telecommunications, digital set top boxes, audio recording, and digital imaging.

By applying FlashFile™ memory family pinouts, Intel StrataFlash memory components allow easy design migrations from existing Word-Wide FlashFile memory (28F160S3 and 28F320S3), and first generation Intel StrataFlash memory (28F640J5 and 28F320J5) devices.

Intel StrataFlash memory components deliver a new generation of forward-compatible software support. By using the Common Flash Interface (CFI) and the Scalable Command Set (SCS), customers can take advantage of density upgrades and optimized write capabilities of future Intel StrataFlash memory devices.

Manufactured on Intel® 0.25 micron ETOX™ VI process technology, Intel StrataFlash memory provides the highest levels of quality and reliability.

**Notice:** This document contains preliminary information on new products in production. The specifications are subject to change without notice. Verify with your local Intel sales office that you have the latest datasheet before finalizing a design.

Order Number: 290667-006
June 2000

# Contents

# Revision History

| Date of Revision | Version | Description |
|---|---|---|
| 07/07/99 | -001 | Original Version |
| 08/03/99 | -002 | $A_0$–$A_2$ indicated on block diagram |
| 09/07/99 | -003 | Changed Minimum Block Erase time,$I_{OL}$, $I_{OH}$, Page Mode and Byte Mode currents. Modified RP# on *AC Waveform for Write Operations* |
| 12/16/99 | -004 | Changed Block Erase time and $t_{AVWH}$<br>Removed all references to 5 V I/O operation<br>Corrected *Ordering Information*, Valid Combinations entries<br>Changed Min program time to 211 µs<br>Added DU to Lead Descriptions table<br>Changed Chip Scale Package to Ball Grid Array Package<br>Changed default read mode to page mode<br>Removed erase queuing from Figure 10, *Block Erase Flowchart* |
| 03/16/00 | -005 | Added Program Max time<br>Added Erase Max time<br>Added Max page mode read current<br>Moved tables to correspond with sections<br>Fixed typographical errors in ordering information and DC parameter table<br>Removed $V_{CCQ1}$ setting and changed $V_{CCQ2/3}$ to $V_{CCQ1/2}$<br>Added recommended resister value for STS pin<br>Change operation temperature range<br>Removed note that rp# could go to 14 V<br>Removed $V_{OL}$ of 0.45 V<br>Removed $V_{OH}$ of 2.4 V<br>Updated $I_{CCR}$ Typ values<br>Added Max lock-bit program and lock times<br>Added note on max measurements |
| 06/26/00 | -006 | Updated cover sheet statement of 700 million units to one billion.<br>Corrected Table 10 to show correct maximum program times.<br>Corrected error in Max block program time in section 6.7<br>Corrected typical erase time in section 6.7 |

# 1.0 Product Overview

The 0.25 µ 3 Volt Intel StrataFlash memory family contains high-density memories organized as 16 Mbytes or 8 Mwords (128-Mbit), 8 Mbytes or 4 Mwords (64-Mbit), and 4 Mbytes or 2 Mwords (32-Mbit). These devices can be accessed as 8- or 16-bit words. The 128-Mbit device is organized as one-hundred-twenty-eight 128-Kbyte (131,072 bytes) erase blocks. The 64-Mbit device is organized as sixty-four 128-Kbyte erase blocks while the 32-Mbits device contains thirty-two 128-Kbyte erase blocks. Blocks are selectively and individually lockable and unlockable in-system. A 128-bit protection register has multiple uses, including unique flash device identification.

The device's optimized architecture and interface dramatically increases read performance by supporting page-mode reads. This read mode is ideal for non-clock memory systems.

A Common Flash Interface (CFI) permits software algorithms to be used for entire families of devices. This allows device-independent, JEDEC ID-independent, and forward- and backward-compatible software support for the specified flash device families. Flash vendors can standardize their existing interfaces for long-term compatibility.

Scaleable Command Set (SCS) allows a single, simple software driver in all host systems to work with all SCS-compliant flash memory devices, independent of system-level packaging (e.g., memory card, SIMM, or direct-to-board placement). Additionally, SCS provides the highest system/device data transfer rates and minimizes device and system-level implementation costs.

A Command User Interface (CUI) serves as the interface between the system processor and internal operation of the device. A valid command sequence written to the CUI initiates device automation. An internal Write State Machine (WSM) automatically executes the algorithms and timings necessary for block erase, program, and lock-bit configuration operations.

A block erase operation erases one of the device's 128-Kbyte blocks typically within one second—independent of other blocks. Each block can be independently erased 100,000 times. Block erase suspend mode allows system software to suspend block erase to read or program data from any other block. Similarly, program suspend allows system software to suspend programming (byte/word program and write-to-buffer operations) to read data or execute code from any other block that is not being suspended.

Each device incorporates a Write Buffer of 32 bytes (16 words) to allow optimum programming performance. By using the Write Buffer, data is programmed in buffer increments. This feature can improve system program performance more than 20 times over non-Write Buffer writes.

Individual block locking uses block lock-bits to lock and unlock blocks. Block lock-bits gate block erase and program operations. Lock-bit configuration operations set and clear lock-bits (Set Block Lock-Bit and Clear Block Lock-Bits commands).

The status register indicates when the WSM's block erase, program, or lock-bit configuration operation is finished.

The STS (STATUS) output gives an additional indicator of WSM activity by providing both a hardware signal of status (versus software polling) and status masking (interrupt masking for background block erase, for example). Status indication using STS minimizes both CPU overhead and system power consumption. When configured in level mode (default mode), it acts as a RY/BY# pin. When low, STS indicates that the WSM is performing a block erase, program, or lock-bit configuration. STS-high indicates that the WSM is ready for a new command, block erase is

suspended (and programming is inactive), program is suspended, or the device is in reset/power-down mode. Additionally, the configuration command allows the STS pin to be configured to pulse on completion of programming and/or block erases.

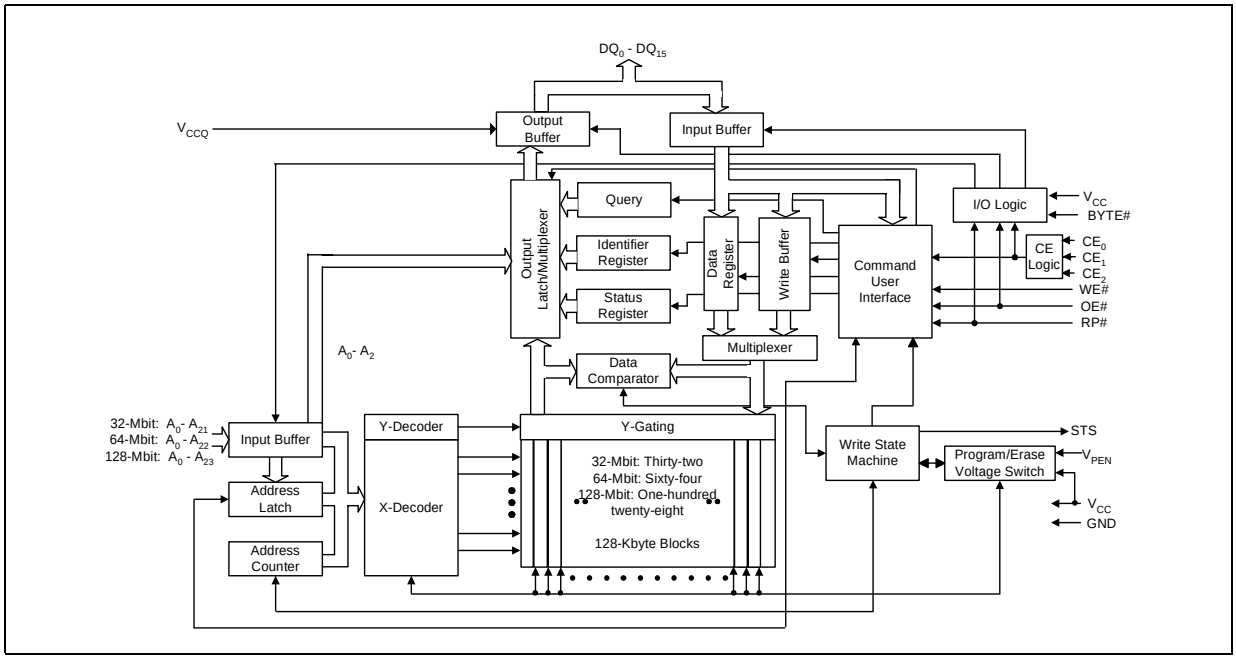
Three CE pins are used to enable and disable the device. A unique CE logic design (see Table 2, “Chip Enable Truth Table” on page 7) reduces decoder logic typically required for multi-chip designs. External logic is not required when designing a single chip, a dual chip, or a 4-chip miniature card or SIMM module.

The BYTE# pin allows either x8 or x16 read/writes to the device. BYTE# at logic low selects 8-bit mode; address $A_0$ selects between the low byte and high byte. BYTE# at logic high enables 16-bit operation; address $A_1$ becomes the lowest order address and address $A_0$ is not used (don’t care). A device block diagram is shown in Figure 1 on page 2.

When the device is disabled (see Table 2 on page 7) and the RP# pin is at $V_{CC}$, the standby mode is enabled. When the RP# pin is at GND, a further power-down mode is enabled which minimizes power consumption and provides write protection during reset. A reset time ($t_{PHQV}$) is required from RP# switching high until outputs are valid. Likewise, the device has a wake time ($t_{PHWL}$) from RP#-high until writes to the CUI are recognized. With RP# at GND, the WSM is reset and the status register is cleared.

3 Volt Intel StrataFlash memory devices are available in two package types. Both 56-lead TSOP (Thin Small Outline Package) and BGA (Ball Grid Array Package) support all offered densities. Figure 2 and Figure 3 show the pinouts.

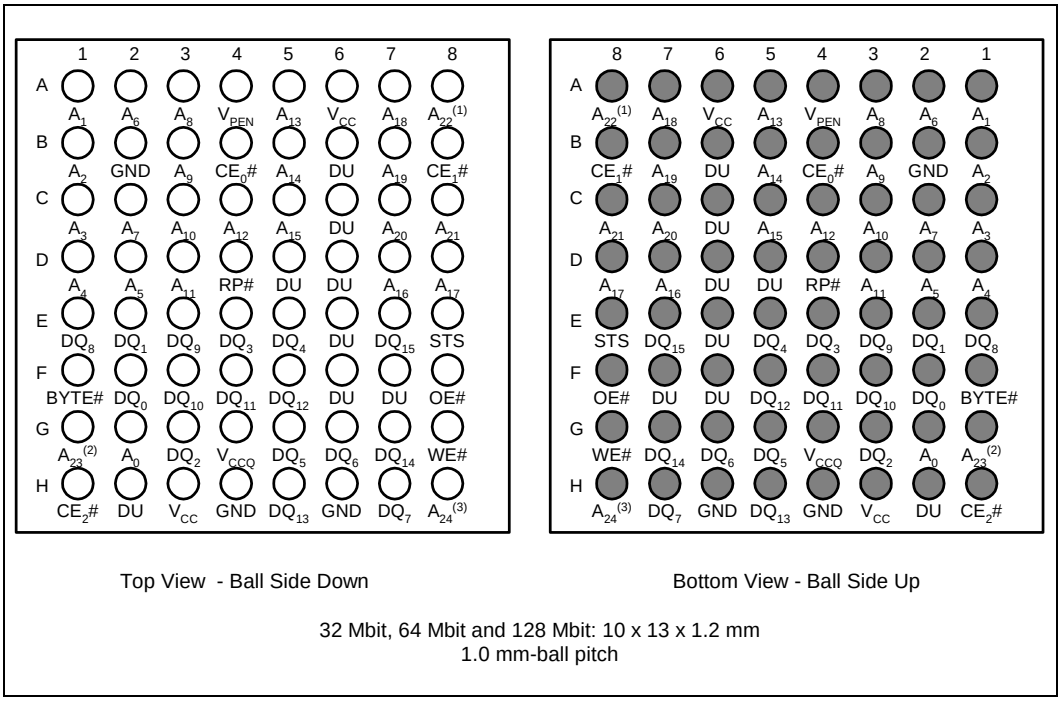
**Figure 1. 3 Volt Intel® StrataFlash™ Memory Block Diagram**



## Table 1. Lead Descriptions

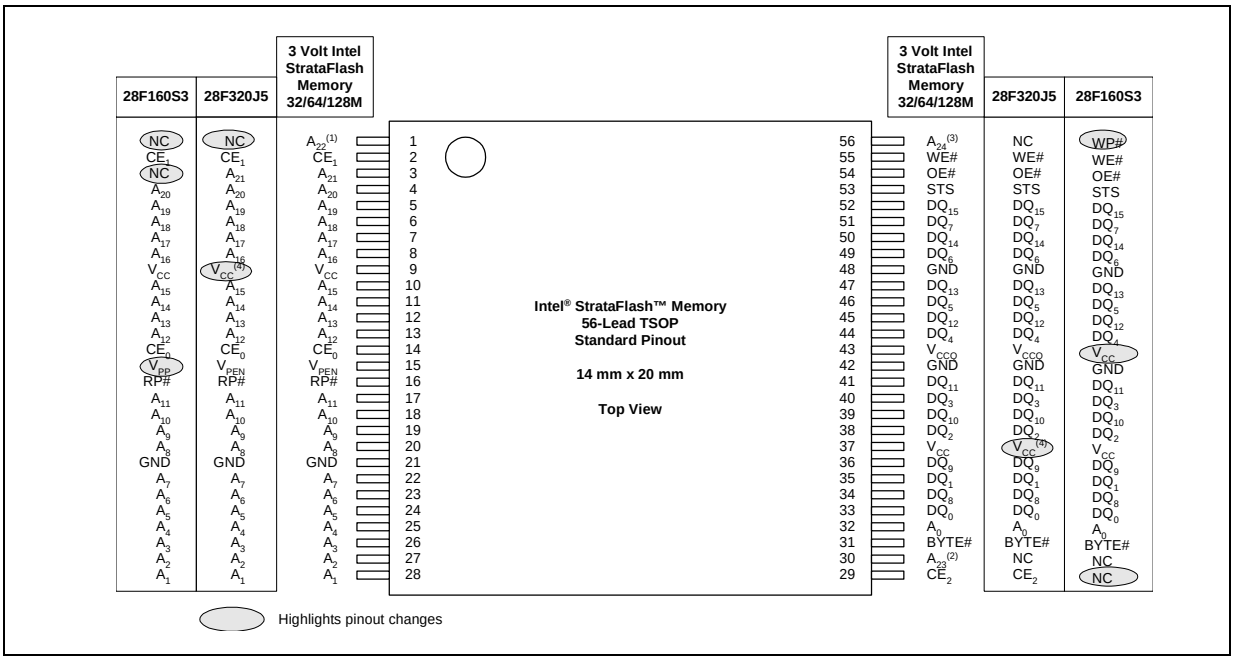
| Symbol | Type | Name and Function |
|---|---|---|
| $A_0$ | INPUT | **BYTE-SELECT ADDRESS:** Selects between high and low byte when the device is in x8 mode. This address is latched during a x8 program cycle. Not used in x16 mode (i.e., the $A_0$ input buffer is turned off when BYTE# is high). |
| $A_1$–$A_{23}$ | INPUT | **ADDRESS INPUTS:** Inputs for addresses during read and program operations. Addresses are internally latched during a program cycle.<br>32-Mbit: $A_0$–$A_{21}$<br>64-Mbit: $A_0$–$A_{22}$<br>128-Mbit: $A_0$–$A_{23}$ |
| $DQ_0$–$DQ_7$ | INPUT/ OUTPUT | **LOW-BYTE DATA BUS:** Inputs data during buffer writes and programming, and inputs commands during Command User Interface (CUI) writes. Outputs array, query, identifier, or status data in the appropriate read mode. Floated when the chip is de-selected or the outputs are disabled. Outputs $DQ_6$–$DQ_0$ are also floated when the Write State Machine (WSM) is busy. Check SR.7 (status register bit 7) to determine WSM status. |
| $DQ_8$–$DQ_{15}$ | INPUT/ OUTPUT | **HIGH-BYTE DATA BUS:** Inputs data during x16 buffer writes and programming operations. Outputs array, query, or identifier data in the appropriate read mode; not used for status register reads. Floated when the chip is de-selected, the outputs are disabled, or the WSM is busy. |
| $CE_0$, $CE_1$, $CE_2$ | INPUT | **CHIP ENABLES:** Activates the device's control logic, input buffers, decoders, and sense amplifiers. When the device is de-selected (see Table 2 on page 7), power reduces to standby levels.<br>All timing specifications are the same for these three signals. Device selection occurs with the first edge of $CE_0$, $CE_1$, or $CE_2$ that enables the device. Device deselection occurs with the first edge of $CE_0$, $CE_1$, or $CE_2$ that disables the device (see Table 2 on page 7). |
| RP# | INPUT | **RESET/ POWER-DOWN:** Resets internal automation and puts the device in power-down mode. RP#-high enables normal operation. Exit from reset sets the device to read array mode. When driven low, RP# inhibits write operations which provides data protection during power transitions. |
| OE# | INPUT | **OUTPUT ENABLE:** Activates the device's outputs through the data buffers during a read cycle. OE# is active low. |
| WE# | INPUT | **WRITE ENABLE:** Controls writes to the Command User Interface, the Write Buffer, and array blocks. WE# is active low. Addresses and data are latched on the rising edge of the WE# pulse. |
| STS | OPEN DRAIN OUTPUT | **STATUS:** Indicates the status of the internal state machine. When configured in level mode (default mode), it acts as a RY/BY# pin. When configured in one of its pulse modes, it can pulse to indicate program and/or erase completion. For alternate configurations of the STATUS pin, see the Configurations command. Tie STS to $V_{CCQ}$ with a pull-up resistor. |
| BYTE# | INPUT | **BYTE ENABLE:** BYTE# low places the device in x8 mode. All data is then input or output on $DQ_0$–$DQ_7$, while $DQ_8$–$DQ_{15}$ float. Address $A_0$ selects between the high and low byte. BYTE# high places the device in x16 mode, and turns off the $A_0$ input buffer. Address $A_1$ then becomes the lowest order address. |
| $V_{PEN}$ | INPUT | **ERASE / PROGRAM / BLOCK LOCK ENABLE:** For erasing array blocks, programming data, or configuring lock-bits.<br>With $V_{PEN} \le V_{PENLK}$, memory contents cannot be altered. |
| $V_{CC}$ | SUPPLY | **DEVICE POWER SUPPLY:** With $V_{CC} \le V_{LKO}$, all write attempts to the flash memory are inhibited. |
| $V_{CCQ}$ | OUTPUT BUFFER SUPPLY | **OUTPUT BUFFER POWER SUPPLY:** This voltage controls the device's output voltages. To obtain output voltages compatible with system data bus voltages, connect $V_{CCQ}$ to the system supply voltage. |
| GND | SUPPLY | **GROUND:** Do not float any ground pins. |
| NC | | **NO CONNECT:** Lead is not internally connected; it may be driven or floated. |
| DU | | DON'T USE: **Do not drive ball to $V_{IH}$ or $V_{IL}$, leave disconnected** |

## Figure 2. 3 Volt Intel® StrataFlash™ Memory Easy BGA Package




**NOTES:**
1. Address $A_{22}$ is only valid on 64-Mbit densities and above, otherwise, it is a no connect (NC)
2. Address $A_{23}$ is only valid on 128-Mbit densities and above, otherwise, it is a no connect (NC)
3. Address $A_{24}$ is only valid on 256-Mbit densities and above, otherwise, it is a no connect (NC)
4. Don't Use (DU) pins refer to pins that should not be connected

## Figure 3. 3 Volt Intel® StrataFlash™ Memory 56-Lead TSOP (32/64/128 Mbit) Offers an Easy Migration from the 32-Mbit Intel StrataFlash Component (28F320J5) or the 16-Mbit FlashFile™ Component (28F160S3)




0667-03

**NOTES:**

1. $A_{22}$ exists on 64-, 128- and 256-Mbit densities. On 32-Mbit densities this pin is a no-connect (NC).
2. $A_{23}$ exists on 128-Mbit densities. On 32- and 64-Mbit densities this pin is a no-connect (NC).
3. $A_{24}$ exists on 256-Mbit densities. On 32-, 64- and 128-Mbit densities this pin is a no-connect (NC).
4. $V_{CC}$ = 5 V ± 10% for the 28F640J5/28F320J5.

# 2.0 Principles of Operation

The Intel StrataFlash memory devices include an on-chip WSM to manage block erase, program, and lock-bit configuration functions. It allows for 100% TTL-level control inputs, fixed power supplies during block erasure, program, lock-bit configuration, and minimal processor overhead with RAM-like interface timings.

After initial device power-up or return from reset/power-down mode (see Section 3.0, "Bus Operations" on page 7), the device defaults to read array mode. Manipulation of external memory control pins allows array read, standby, and output disable operations.

Read array, status register, query, and identifier codes can be accessed through the CUI (Command User Interface) independent of the $V_{PEN}$ voltage. $V_{PENH}$ on $V_{PEN}$ enables successful block erasure, programming, and lock-bit configuration. All functions associated with altering memory contents—block erase, program, lock-bit configuration—are accessed via the CUI and verified through the status register.

Commands are written using standard micro-processor write timings. The CUI contents serve as input to the WSM, which controls the block erase, program, and lock-bit configuration. The internal algorithms are regulated by the WSM, including pulse repetition, internal verification, and margining of data. Addresses and data are internally latched during program cycles.

Interface software that initiates and polls progress of block erase, program, and lock-bit configuration can be stored in any block. This code is copied to and executed from system RAM during flash memory updates. After successful completion, reads are again possible via the Read Array command. Block erase suspend allows system software to suspend a block erase to read or program data from/to any other block. Program suspend allows system software to suspend a program to read data from any other flash memory array location.

## 2.1 Data Protection

Depending on the application, the system designer may choose to make the $V_{PEN}$ switchable (available only when memory block erases, programs, or lock-bit configurations are required) or hardwired to $V_{PENH}$. The device accommodates either design practice and encourages optimization of the processor-memory interface.

When $V_{PEN} \leq V_{PENLK}$, memory contents cannot be altered. The CUI's two-step block erase, byte/word program, and lock-bit configuration command sequences provide protection from unwanted operations even when $V_{PENH}$ is applied to $V_{PEN}$. All program functions are disabled when $V_{CC}$ is below the write lockout voltage $V_{LKO}$ or when RP# is $V_{IL}$. The device's block locking capability provides additional protection from inadvertent code or data alteration by gating erase and program operations.

# 3.0 Bus Operations

The local CPU reads and writes flash memory in-system. All bus cycles to or from the flash memory conform to standard microprocessor bus cycles.

**Figure 4. Memory Map**

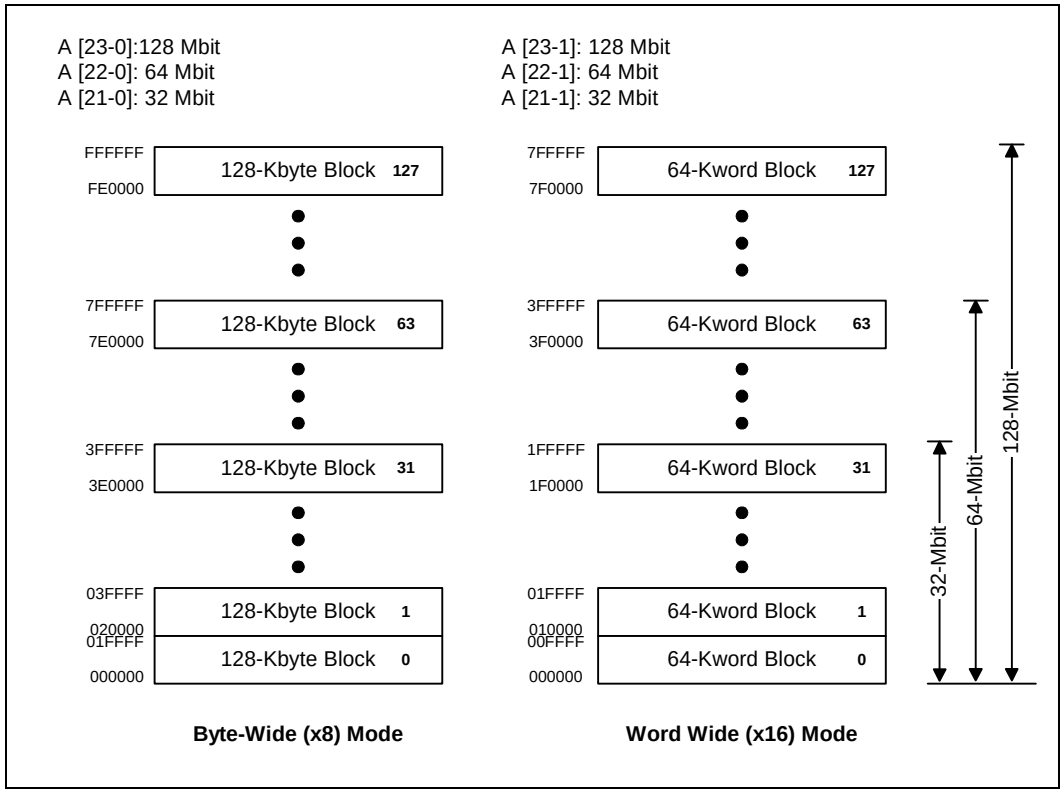



**Table 2. Chip Enable Truth Table**

| $CE_2$ | $CE_1$ | $CE_0$ | DEVICE |
|---|---|---|---|
| $V_{IL}$ | $V_{IL}$ | $V_{IL}$ | Enabled |
| $V_{IL}$ | $V_{IL}$ | $V_{IH}$ | Disabled |
| $V_{IL}$ | $V_{IH}$ | $V_{IL}$ | Disabled |
| $V_{IL}$ | $V_{IH}$ | $V_{IH}$ | Disabled |
| $V_{IH}$ | $V_{IL}$ | $V_{IL}$ | Enabled |
| $V_{IH}$ | $V_{IL}$ | $V_{IH}$ | Enabled |
| $V_{IH}$ | $V_{IH}$ | $V_{IL}$ | Enabled |
| $V_{IH}$ | $V_{IH}$ | $V_{IH}$ | Disabled |

**NOTE:** For single-chip applications, $CE_2$ and $CE_1$ can be strapped to GND.

## 3.1 Read

Information can be read from any block, query, identifier codes, or status register independent of the $V_{PEN}$ voltage.

Upon initial device power-up or after exit from reset/power-down mode, the device automatically resets to read array mode. Otherwise, write the appropriate read mode command (Read Array, Read Query, Read Identifier Codes, or Read Status Register) to the CUI. Six control pins dictate the data flow in and out of the component: $CE_0$, $CE_1$, $CE_2$, OE#, WE#, and RP#. The device must be enabled (see Table 2, “Chip Enable Truth Table” on page 7), and OE# must be driven active to obtain data at the outputs. $CE_0$, $CE_1$, and $CE_2$ are the device selection controls and, when enabled (see Table 2), select the memory device. OE# is the data output ($DQ_0$–$DQ_{15}$) control and, when active, drives the selected memory data onto the I/O bus. WE# must be at $V_{IH}$.

When reading information in read array mode, the device defaults to asynchronous page mode. This mode provides high data transfer rate for memory subsystems. In this state, data is internally read and stored in a high-speed page buffer. $A_{2:0}$ addresses data in the page buffer. The page size is four words or eight bytes. Asynchronous word/byte mode is supported with no additional commands required.

## 3.2 Output Disable

With OE# at a logic-high level ($V_{IH}$), the device outputs are disabled. Output pins $DQ_0$–$DQ_{15}$ are placed in a high-impedance state.

## 3.3 Standby

$CE_0$, $CE_1$, and $CE_2$ can disable the device (see Table 2) and place it in standby mode which substantially reduces device power consumption. $DQ_0$–$DQ_{15}$ outputs are placed in a high-impedance state independent of OE#. If deselected during block erase, program, or lock-bit configuration, the WSM continues functioning, and consuming active power until the operation completes.

## 3.4 Reset/Power-Down

RP# at $V_{IL}$ initiates the reset/power-down mode.

In read modes, RP#-low deselects the memory, places output drivers in a high-impedance state, and turns off numerous internal circuits. RP# must be held low for a minimum of $t_{PLPH}$. Time $t_{PHQV}$ is required after return from reset mode until initial memory access outputs are valid. After this wake-up interval, normal operation is restored. The CUI is reset to read array mode and status register is set to 80H.

During block erase, program, or lock-bit configuration modes, RP#-low will abort the operation. In default mode, STS transitions low and remains low for a maximum time of $t_{PLPH} + t_{PHRH}$ until the reset operation is complete. Memory contents being altered are no longer valid; the data may be partially corrupted after a program or partially altered after an erase or lock-bit configuration. Time $t_{PHWL}$ is required after RP# goes to logic-high ($V_{IH}$) before another command can be written.

As with any automated device, it is important to assert RP# during system reset. When the system comes out of reset, it expects to read from the flash memory. Automated flash memories provide status information when accessed during block erase, program, or lock-bit configuration modes. If a CPU reset occurs with no flash memory reset, proper initialization may not occur because the flash memory may be providing status information instead of array data. Intel® Flash memories allow proper initialization following a system reset through the use of the RP# input. In this application, RP# is controlled by the same RESET# signal that resets the system CPU.

## 3.5 Read Query

The read query operation outputs block status information, CFI (Common Flash Interface) ID string, system interface information, device geometry information, and Intel-specific extended query information.

## 3.6 Read Identifier Codes

The read identifier codes operation outputs the manufacturer code, device code and the block lock configuration codes for each block (see Figure 5 on page 10). Using the manufacturer and device codes, the system CPU can automatically match the device with its proper algorithms. The block lock configuration codes identify locked and unlocked blocks.

## 3.7 Write

Writing commands to the CUI enables reading of device data, query, identifier codes, inspection and clearing of the status register, and, when $V_{PEN} = V_{PENH}$, block erasure, program, and lock-bit configuration.

The Block Erase command requires appropriate command data and an address within the block to be erased. The Byte/Word Program command requires the command and address of the location to be written. Set Block Lock-Bit commands require the command and block within the device to be locked. The Clear Block Lock-Bits command requires the command and address within the device.

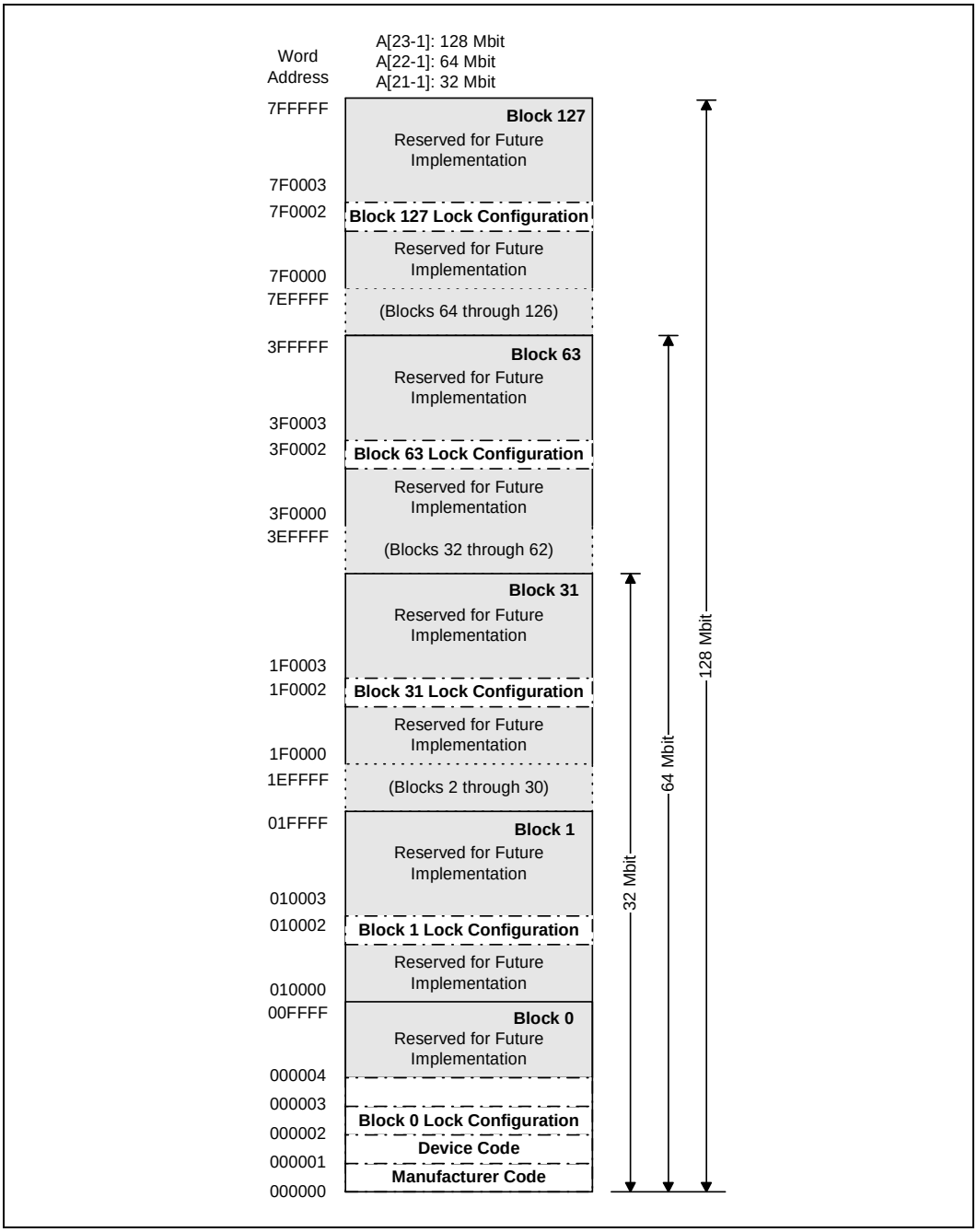
The CUI does not occupy an addressable memory location. It is written when the device is enabled and WE# is active. The address and data needed to execute a command are latched on the rising edge of WE# or the first edge of $CE_0$, $CE_1$, or $CE_2$ that disables the device (see Table 2). Standard microprocessor write timings are used.

# 4.0 Command Definitions

When the $V_{PEN}$ voltage ≤ $V_{PENLK}$, only read operations from the status register, query, identifier codes, or blocks are enabled. Placing $V_{PENH}$ on $V_{PEN}$ additionally enables block erase, program, and lock-bit configuration operations.

Device operations are selected by writing specific commands into the CUI. Table 4 defines these commands.

## Figure 5. Device Identifier Code Memory Map




**NOTE:** $A_0$ is not used in either x8 or x16 modes when obtaining these identifier codes. Data is always given on the low byte in x16 mode (upper byte contains 00h).

### Table 3. Bus Operations

| Mode | Notes | RP# | $CE_{0,1,2}$[1] | OE#[2] | WE#[2] | Address | $V_{PEN}$ | DQ[3] | STS (default mode) |
|---|---|---|---|---|---|---|---|---|---|
| Read Array | 4,5,6 | $V_{IH}$ | Enabled | $V_{IL}$ | $V_{IH}$ | X | X | $D_{OUT}$ | High Z[7] |
| Output Disable | | $V_{IH}$ | Enabled | $V_{IH}$ | $V_{IH}$ | X | X | High Z | X |
| Standby | | $V_{IH}$ | Disabled | X | X | X | X | High Z | X |
| Reset/Power-Down Mode | | $V_{IL}$ | X | X | X | X | X | High Z | High Z[7] |
| Read Identifier Codes | | $V_{IH}$ | Enabled | $V_{IL}$ | $V_{IH}$ | See Figure 5 | X | Note 8 | High Z[7] |
| Read Query | | $V_{IH}$ | Enabled | $V_{IL}$ | $V_{IH}$ | See Table 7 | X | Note 9 | High Z[7] |
| Read Status (WSM off) | | $V_{IH}$ | Enabled | $V_{IL}$ | $V_{IH}$ | X | X | $D_{OUT}$ | |
| Read Status (WSM on) | | $V_{IH}$ | Enabled | $V_{IL}$ | $V_{IH}$ | X | X | $DQ_7 = D_{OUT}$<br>$DQ_{15–8}$ = High Z<br>$DQ_{6–0}$ = High Z | |
| Write | 6,10,11 | $V_{IH}$ | Enabled | $V_{IH}$ | $V_{IL}$ | X | $V_{PENH}$ | $D_{IN}$ | X |

**NOTES:**
1. See Table 2 for valid CE configurations.
2. OE# and WE# should never be enabled simultaneously.
3. DQ refers to $DQ_0$–$DQ_7$ if BYTE# is low and $DQ_0$–$DQ_{15}$ if BYTE# is high.
4. Refer to *DC Characteristics*. When $V_{PEN} \leq V_{PENLK}$, memory contents can be read, but not altered.
5. X can be $V_{IL}$ or $V_{IH}$ for control and address pins, and $V_{PENLK}$ or $V_{PENH}$ for $V_{PEN}$. See *DC Characteristics* for $V_{PENLK}$ and $V_{PENH}$ voltages.
6. In default mode, STS is $V_{OL}$ when the WSM is executing internal block erase, program, or lock-bit configuration algorithms. It is $V_{OH}$ when the WSM is not busy, in block erase suspend mode (with programming inactive), program suspend mode, or reset/power-down mode.
7. High Z will be $V_{OH}$ with an external pull-up resistor.
8. See Section 3.6 for read identifier code data.
9. See Section 4.2 for read query data.
10. Command writes involving block erase, program, or lock-bit configuration are reliably executed when $V_{PEN} = V_{PENH}$ and $V_{CC}$ is within specification.
11. Refer to Table 4 for valid $D_{IN}$ during a write operation.

## Table 4. Intel® StrataFlash™ Memory Command Set Definitions(1)

| Command | Scaleable or Basic Command Set(2) | Bus Cycles Req'd. | Notes | First Bus Cycle | | | Second Bus Cycle | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | Oper(3) | Addr(4) | Data(5,6) | Oper(3) | Addr(4) | Data(5,6) |
| Read Array | SCS/BCS | 1 | | Write | X | FFH | | | |
| Read Identifier Codes | SCS/BCS | ≥ 2 | 7 | Write | X | 90H | Read | IA | ID |
| Read Query | SCS | ≥ 2 | | Write | X | 98H | Read | QA | QD |
| Read Status Register | SCS/BCS | 2 | 8 | Write | X | 70H | Read | X | SRD |
| Clear Status Register | SCS/BCS | 1 | | Write | X | 50H | | | |
| Write to Buffer | SCS/BCS | > 2 | 9, 10, 11 | Write | BA | E8H | Write | BA | N |
| Word/Byte Program | SCS/BCS | 2 | 12,13 | Write | X | 40H or 10H | Write | PA | PD |
| Block Erase | SCS/BCS | 2 | 11,12 | Write | BA | 20H | Write | BA | D0H |
| Block Erase, Program Suspend | SCS/BCS | 1 | 12,14 | Write | X | B0H | | | |
| Block Erase, Program Resume | SCS/BCS | 1 | 12 | Write | X | D0H | | | |
| Configuration | SCS | 2 | | Write | X | B8H | Write | X | CC |
| Set Read Configuration | | 2 | | Write | X | 60H | Write | RCD | 03H |
| Set Block Lock-Bit | SCS | 2 | | Write | X | 60H | Write | BA | 01H |
| Clear Block Lock-Bits | SCS | 2 | 15 | Write | X | 60H | Write | X | D0H |
| Protection Program | | 2 | | Write | X | C0H | Write | PA | PD |

**NOTES:**

1. Commands other than those shown above are reserved by Intel for future device implementations and should not be used.
2. The Basic Command Set (BCS) is the same as the 28F008SA Command Set or Intel Standard Command Set. The Scaleable Command Set (SCS) is also referred to as the Intel Extended Command Set.
3. Bus operations are defined in Table 3.
4. X = Any valid address within the device.
   BA = Address within the block.
   IA = Identifier Code Address: see Figure 5 and Table 15.
   QA = Query database Address.
   PA = Address of memory location to be programmed.
   RCD = Data to be written to the read configuration register. This data is presented to the device on $A_{16-1}$; all other address inputs are ignored.
5. ID = Data read from Identifier Codes.
   QD = Data read from Query database.
   SRD = Data read from status register. See Table 16 for a description of the status register bits.
   PD = Data to be programmed at location PA. Data is latched on the rising edge of WE#.
   CC = Configuration Code.
6. The upper byte of the data bus ($DQ_8$–$DQ_{15}$) during command writes is a "Don't Care" in x16 operation.
7. Following the Read Identifier Codes command, read operations access manufacturer, device and block lock codes. See Section 4.3 for read identifier code data.
8. If the WSM is running, only $DQ_7$ is valid; $DQ_{15}$–$DQ_8$ and $DQ_6$–$DQ_0$ float, which places them in a high-impedance state.
9. After the Write to Buffer command is issued check the XSR to make sure a buffer is available for writing.

10. The number of bytes/words to be written to the Write Buffer = N + 1, where N = byte/word count argument. Count ranges on this device for byte mode are N = 00H to N = 1FH and for word mode are N = 0000H to N = 000FH. The third and consecutive bus cycles, as determined by N, are for writing data into the Write Buffer. The Confirm command (D0H) is expected after exactly N + 1 write cycles; any other command at that point in the sequence aborts the write to buffer operation. Please see Figure 7, "Write to Buffer Flowchart" on page 30 for additional information.
11. The write to buffer or erase operation does not begin until a Confirm command (D0h) is issued.
12. Attempts to issue a block erase or program to a locked block.
13. Either 40H or 10H are recognized by the WSM as the byte/word program setup.
14. Program suspends can be issued after either the Write-to-Buffer or Word-/Byte-Program operation is initiated.
15. The clear block lock-bits operation simultaneously clears all block lock-bits.

## 4.1 Read Array Command

Upon initial device power-up and after exit from reset/power-down mode, the device defaults to read array mode. The read configuration register defaults to asynchronous read page mode. The Read Array command also causes the device to enter read array mode. The device remains enabled for reads until another command is written. Once the internal WSM has started a block erase, program, or lock-bit configuration, the device will not recognize the Read Array command until the WSM completes its operation unless the WSM is suspended via an Erase or Program Suspend command. The Read Array command functions independently of the $V_{PEN}$ voltage.

## 4.2 Read Query Mode Command

This section defines the data structure or "database" returned by the Common Flash Interface (CFI) Query command. System software should parse this structure to gain critical information such as block size, density, x8/x16, and electrical specifications. Once this information has been obtained, the software will know which command sets to use to enable flash writes, block erases, and otherwise control the flash component. The Query is part of an overall specification for multiple command set and control interface descriptions called Common Flash Interface, or CFI.

### 4.2.1 Query Structure Output

The Query "database" allows system software to gain information for controlling the flash component. This section describes the device's CFI-compliant interface that allows the host system to access Query data.

Query data are always presented on the lowest-order data outputs ($DQ_{0–7}$) only. The numerical offset value is the address relative to the maximum bus width supported by the device. On this family of devices, the Query table device starting address is a 10h, which is a word address for x16 devices.

For a word-wide (x16) device, the first two bytes of the Query structure, "Q" and "R" in ASCII, appear on the low byte at word addresses 10h and 11h. This CFI-compliant device outputs 00H data on upper bytes. Thus, the device outputs ASCII "Q" in the low byte ($DQ_{0–7}$) and 00h in the high byte ($DQ_{8–15}$).

At Query addresses containing two or more bytes of information, the least significant data byte is presented at the lower address, and the most significant data byte is presented at the higher address.

In all of the following tables, addresses and data are represented in hexadecimal notation, so the "h" suffix has been dropped. In addition, since the upper byte of word-wide devices is always "00h," the leading "00" has been dropped from the table notation and only the lower byte value is shown. Any x16 device outputs can be assumed to have 00h on the upper byte in this mode.

**Table 5. Summary of Query Structure Output as a Function of Device and Mode**

| Device Type/ Mode | Query start location in maximum device bus width addresses | Query data with maximum device bus width addressing | | | Query data with byte addressing | | |
|---|---|---|---|---|---|---|---|
| | | Hex Offset | Hex Code | ASCII Value | Hex Offset | Hex Code | ASCII Value |
| x16 device<br>x16 mode | 10h | 10:<br>11:<br>12: | 0051<br>0052<br>0059 | "Q"<br>"R"<br>"Y" | 20:<br>21:<br>22: | 51<br>00<br>52 | "Q"<br>"Null"<br>"R" |
| x16 device<br>x8 mode | N/A(1) | N/A(1) | | | 20:<br>21:<br>22: | 51<br>51<br>52 | "Q"<br>"Q"<br>"R" |

**NOTE:**
1. The system must drive the lowest order addresses to access all the device's array data when the device is configured in x8 mode. Therefore, word addressing, where these lower addresses are not toggled by the system, is "Not Applicable" for x8-configured devices.

**Table 6. Example of Query Structure Output of a x16- and x8-Capable Device**

| Word Addressing | | | Byte Addressing | | |
|---|---|---|---|---|---|
| Offset | Hex Code | Value | Offset | Hex Code | Value |
| $A_{15}$–$A_0$ | D15–$D_0$ | | $A_7$–$A_0$ | $D_7$–$D_0$ | |
| 0010h | 0051 | "Q" | 20h | 51 | "Q" |
| 0011h | 0052 | "R" | 21h | 51 | "Q" |
| 0012h | 0059 | "Y" | 22h | 52 | "R" |
| 0013h | $P\_ID_{LO}$ | PrVendor | 23h | 52 | "R" |
| 0014h | $P\_ID_{HI}$ | ID # | 24h | 59 | "Y" |
| 0015h | $P_{LO}$ | PrVendor | 25h | 59 | "Y" |
| 0016h | $P_{HI}$ | TblAdr | 26h | $P\_ID_{LO}$ | PrVendor |
| 0017h | $A\_ID_{LO}$ | AltVendor | 27h | $P\_ID_{LO}$ | ID # |
| 0018h | $A\_ID_{HI}$ | ID # | 28h | $P\_ID_{HI}$ | ID # |
| ... | ... | ... | ... | ... | ... |

## 4.2.2 Query Structure Overview

The Query command causes the flash component to display the Common Flash Interface (CFI) Query structure or "database." The structure sub-sections and address locations are summarized below. See *AP-646 Common Flash Interface (CFI) and Command Sets* (order number 292204) for a full description of CFI.

The following sections describe the Query structure sub-sections in detail.

**Table 7. Query Structure(1)**

| Offset | Sub-Section Name | Description |
|---|---|---|
| 00h | | Manufacturer Code |
| 01h | | Device Code |
| (BA+2)h(2) | Block Status Register | Block-Specific Information |
| 04-0Fh | Reserved | *Reserved for Vendor-Specific Information* |
| 10h | CFI Query Identification String | *Reserved for Vendor-Specific Information* |
| 1Bh | System Interface Information | Command Set ID and Vendor Data Offset |
| 27h | Device Geometry Definition | Flash Device Layout |
| *P*(3) | Primary Intel-Specific Extended Query Table | Vendor-Defined Additional Information Specific to the Primary Vendor Algorithm |

**NOTES:**
1. Refer to the Query Structure Output section and offset 28h for the detailed definition of offset address as a function of device bus width and mode.
2. BA = Block Address beginning location (i.e., 02000h is block 2's beginning location when the block size is 128 Kbyte).
3. Offset 15 defines "P" which points to the *Primary Intel-Specific Extended Query* Table.

## 4.2.3 Block Status Register

The block status register indicates whether an erase operation completed successfully or whether a given block is locked or can be accessed for flash program/erase operations.

Block Erase Status (BSR.1) allows system software to determine the success of the last block erase operation. BSR.1 can be used just after power-up to verify that the $V_{CC}$ supply was not accidentally removed during an erase operation. This bit is only reset by issuing another erase operation to the block. The block status register is accessed from word address 02h within each block.

**Table 8. Block Status Register**

| Offset | Length | Description | Address | Value |
|---|---|---|---|---|
| (BA+2)h(1) | 1 | Block Lock Status Register | BA+2: | --00 or --01 |
| | | BSR.0 Block Lock Status<br>0 = Unlocked<br>1 = Locked | BA+2: | (bit 0): 0 or 1 |
| | | BSR.1 Block Erase Status<br>0 = Last erase operation completed successfully<br>1 = Last erase operation did not complete successfully | BA+2: | (bit 1): 0 or 1 |
| | | BSR 2–7: *Reserved for Future Use* | BA+2: | (bit 2–7): 0 |

**NOTE:**
1. BA = The beginning location of a Block Address (i.e., 008000h is block 1's (64-KB block) beginning location in word mode).

## 4.2.4 CFI Query Identification String

The CFI Query Identification String provides verification that the component supports the Common Flash Interface specification. It also indicates the specification version and supported vendor-specified command set(s).

### Table 9. CFI Identification

| Offset | Length | Description | Add. | Hex Code | Value |
|---|---|---|---|---|---|
| 10h | 3 | Query-unique ASCII string “QRY” | 10<br>11:<br>12: | --51<br>--52<br>--59 | “Q”<br>“R”<br>“Y” |
| 13h | 2 | Primary vendor command set and control interface ID code.<br>16-bit ID code for vendor-specified algorithms | 13:<br>14: | --01<br>--00 | |
| 15h | 2 | Extended Query Table primary algorithm address | 15:<br>16: | --31<br>--00 | |
| 17h | 2 | Alternate vendor command set and control interface ID code.<br>0000h means no second vendor-specified algorithm exists | 17:<br>18: | --00<br>--00 | |
| 19h | 2 | Secondary algorithm Extended Query Table address.<br>0000h means none exists | 19:<br>1A: | --00<br>--00 | |

## 4.2.5 System Interface Information

The following device information can optimize system interface software.

### Table 10. System Interface Information

| Offset | Length | Description | Add. | Hex Code | Value |
|---|---|---|---|---|---|
| 1Bh | 1 | $V_{CC}$ logic supply minimum program/erase voltage<br>bits 0–3 BCD 100 mV<br>bits 4–7 BCD volts | 1B: | --27 | 2.7 V |
| 1Ch | 1 | $V_{CC}$ logic supply maximum program/erase voltage<br>bits 0–3 BCD 100 mV<br>bits 4–7 BCD volts | 1C: | --36 | 3.6 V |
| 1Dh | 1 | $V_{PP}$ [programming] supply minimum program/erase voltage<br>bits 0–3 BCD 100 mV<br>bits 4–7 HEX volts | 1D: | --00 | 0.0 V |
| 1Eh | 1 | $V_{PP}$ [programming] supply maximum program/erase voltage<br>bits 0–3 BCD 100 mV<br>bits 4–7 HEX volts | 1E: | --00 | 0.0 V |
| 1Fh | 1 | “n” such that typical single word program time-out = $2^n$ µs | 1F: | --07 | 128 µs |
| 20h | 1 | “n” such that typical max. buffer write time-out = $2^n$ µs | 20: | --07 | 128 µs |
| 21h | 1 | “n” such that typical block erase time-out = $2^n$ ms | 21: | --0A | 1 s |
| 22h | 1 | “n” such that typical full chip erase time-out = $2^n$ ms | 22: | --00 | NA |
| 23h | 1 | “n” such that maximum word program time-out = $2^n$ times typical | 23: | --04 | 2 ms |
| 24h | 1 | “n” such that maximum buffer write time-out = $2^n$ times typical | 24: | --04 | 2 ms |
| 25h | 1 | “n” such that maximum block erase time-out = $2^n$ times typical | 25: | --04 | 16 s |
| 26h | 1 | “n” such that maximum chip erase time-out = $2^n$ times typical | 26: | --00 | NA |

## 4.2.6 Device Geometry Definition

This field provides critical details of the flash device geometry.

**Table 11. Device Geometry Definition**

| Offset | Length | Description | Code See Table Below | | |
|---|---|---|---|---|---|
| 27h | 1 | "n" such that device size = $2^n$ in number of bytes | 27: | | |
| 28h | 2 | Flash device interface: x8 async x16 async x8/x16 async<br>28:00,29:00 28:01,29:00 28:02,29:00 | 28:<br>29: | --02<br>--00 | x8/ x16 |
| 2Ah | 2 | "n" such that maximum number of bytes in write buffer = $2^n$ | 2A:<br>2B: | --05<br>--00 | 32 |
| 2Ch | 1 | Number of erase block regions within device:<br>1. x = 0 means no erase blocking; the device erases in "bulk"<br>2. x specifies the number of device or partition regions with one or more contiguous same-size erase blocks<br>3. Symmetrically blocked partitions have one blocking region<br>4. Partition size = (total blocks) x (individual block size) | 2C: | --01 | 1 |
| 2Dh | 4 | Erase Block Region 1 Information<br>bits 0–15 = y, y+1 = number of identical-size erase blocks<br>bits 16–31 = z, region erase block(s) size are z x 256 bytes | 2D:<br>2E:<br>2F:<br>30: | | |

**Device Geometry Definition**

| Address | 32 Mbit | 64 Mbit | 128 Mbit |
|---|---|---|---|
| 27: | --16 | --17 | --18 |
| 28: | --02 | --02 | --02 |
| 29: | --00 | --00 | --00 |
| 2A: | --05 | --05 | --05 |
| 2B: | --00 | --00 | --00 |
| 2C: | --01 | --01 | --01 |
| 2D: | --1F | --3F | --7F |
| 2E: | --00 | --00 | --00 |
| 2F: | --00 | --00 | --00 |
| 30: | --02 | --02 | --02 |

## 4.2.7 Primary-Vendor Specific Extended Query Table

Certain flash features and commands are optional. The *Primary Vendor-Specific Extended Query* table specifies this and other similar information.

**Table 12. Primary Vendor-Specific Extended Query**

| Offset[1] P = 31h | Length | Description (Optional Flash Features and Commands) | Add. | Hex Code | Value |
|---|---|---|---|---|---|
| (P+0)h | 3 | Primary extended query table | 31: | --50 | "P" |
| (P+1)h | | Unique ASCII string "PRI" | 32: | --52 | "R" |
| (P+2)h | | | 33: | --49 | "I" |
| (P+3)h | 1 | Major version number, ASCII | 34: | --31 | "1" |
| (P+4)h | 1 | Minor version number, ASCII | 35: | --31 | "1" |
| (P+5)h<br>(P+6)h<br>(P+7)h<br>(P+8)h | 4 | Optional feature and command support (1=yes, 0=no)<br>*bits 9–31 are reserved; undefined bits are "0." If bit 31 is "1" then another 31 bit field of optional features follows at the end of the bit-30 field.* | 36:<br>37:<br>38:<br>39: | --0A<br>--00<br>--00<br>--00 | |
| | | bit 0 Chip erase supported | bit 0 = 0 | | No |
| | | bit 1 Suspend erase supported | bit 1 = 1 | | Yes |
| | | bit 2 Suspend program supported | bit 2 = 1 | | Yes |
| | | bit 3 Legacy lock/unlock supported | bit 3 = 1[1] | | Yes[1] |
| | | bit 4 Queued erase supported | bit 4 = 0 | | No |
| | | bit 5 Instant Individual block locking supported | bit 5 = 0 | | No |
| | | bit 6 Protection bits supported | bit 6 = 1 | | Yes |
| | | bit 7 Page-mode read supported | bit 7 = 1 | | Yes |
| | | bit 8 Synchronous read supported | bit 8 = 0 | | No |
| (P+9)h | 1 | Supported functions after suspend: read Array, Status, Query<br>Other supported operations are:<br>*bits 1–7 reserved; undefined bits are "0"* | 3A: | --01 | |
| | | bit 0 Program supported after erase suspend | bit 0 = 1 | | Yes |
| (P+A)h<br>(P+B)h | 2 | Block status register mask<br>*bits 2–15 are Reserved; undefined bits are "0"* | 3B:<br>3C: | --01<br>--00 | |
| | | bit 0 Block Lock-Bit Status register active | bit 0 = 1 | | Yes |
| | | bit 1 Block Lock-Down Bit Status active | bit 1 = 0 | | No |
| (P+C)h | 1 | $V_{CC}$ logic supply highest performance program/erase voltage<br>bits 0–3 BCD value in 100 mV<br>bits 4–7 BCD value in volts | 3D: | --33 | 3.3 V |
| (P+D)h | 1 | $V_{PP}$ optimum program/erase supply voltage<br>bits 0–3 BCD value in 100 mV<br>bits 4–7 HEX value in volts | 3E: | --00 | 0.0 V |

**NOTE:**
1. Future devices may not support the described "Legacy Lock/Unlock" function. Thus bit 3 would have a value of "0."

**Table 13. Protection Register Information**

| Offset[1] P = 31h | Length | Description (Optional Flash Features and Commands) | Add. | Hex Code | Value |
|---|---|---|---|---|---|
| (P+E)h | 1 | Number of Protection register fields in JEDEC ID space. "00h," indicates that 256 protection bytes are available | 3F: | --01 | 01 |
| (P+F)h<br>(P+10)h<br>(P+11)h<br>(P+12)h | 4 | Protection Field 1: Protection Description<br>This field describes user-available One Time Programmable (OTP) protection register bytes. Some are pre-programmed with device-unique serial numbers. Others are user-programmable. Bits 0-15 point to the protection register lock byte, the section's first byte. The following bytes are factory pre-programmed and user-programmable.<br>bits 0-7 = Lock/bytes JEDEC-plane physical low address<br>bits 8-15 = Lock/bytes JEDEC-plane physical high address<br>bits 16-23 = "n" such that $2^n$ = factory pre-programmed bytes<br>bits 24-31 = "n" such that $2^n$ = user-programmable bytes | 40: | --00 | 00h |

**NOTE:**
1. The variable P is a pointer which is defined at CFI offset 15h.

**Table 14. Burst Read Information**

| Offset[1] P = 31h | Length | Description (Optional Flash Features and Commands) | Add. | Hex Code | Value |
|---|---|---|---|---|---|
| (P+13)h | 1 | Page Mode Read capability<br>bits 0–7 = "n" such that $2^n$ HEX value represents the number of read-page bytes. See offset 28h for device word width to determine page-mode data output width. 00h indicates no read page buffer. | 44: | --03 | 8 byte |
| (P+14)h | 1 | Number of synchronous mode read configuration fields that follow. 00h indicates no burst capability. | 45: | --00 | 0 |
| (P+15)h | | *Reserved for future use* | 46: | | |

**NOTE:**
1. The variable P is a pointer which is defined at CFI offset 15h.

## 4.3 Read Identifier Codes Command

The identifier code operation is initiated by writing the Read Identifier Codes command. Following the command write, read cycles from addresses shown in Figure 5 on page 10 retrieve the manufacturer, device and block lock configuration codes (see Table 15 for identifier code values). Page-mode reads are not supported in this read mode. To terminate the operation, write another valid command. Like the Read Array command, the Read Identifier Codes command functions independently of the $V_{PEN}$ voltage. This command is valid only when the WSM is off or the device is suspended. Following the Read Identifier Codes command, the following information can be read:

**Table 15. Identifier Codes**

| Code | | Address[1] | Data |
|---|---|---|---|
| Manufacture Code | | 00000 | (00) 89 |
| Device Code | 32-Mbit | 00001 | (00) 16 |
| | 64-Mbit | 00001 | (00) 17 |
| | 128-Mbit | 00001 | (00) 18 |
| Block Lock Configuration | | x0002[2] | |
| • Block Is Unlocked | | | $DQ_0$ = 0 |
| • Block Is Locked | | | $DQ_0$ = 1 |
| • Reserved for Future Use | | | $DQ_{1–7}$ |

**NOTES:**
1. $A_0$ is not used in either x8 or x16 modes when obtaining the identifier codes. The lowest order address line is $A_1$. Data is always presented on the low byte in x16 mode (upper byte contains 00h).
2. X selects the specific block's lock configuration code. See Figure 5 for the device identifier code memory map.

## 4.4 Read Status Register Command

The status register may be read to determine when a block erase, program, or lock-bit configuration is complete and whether the operation completed successfully. It may be read at any time by writing the Read Status Register command. After writing this command, all subsequent read operations output data from the status register until another valid command is written. Page-mode reads are not supported in this read mode. The status register contents are latched on the falling edge of OE# or the first edge of $CE_0$, $CE_1$, or $CE_2$ that enables the device (see Table 2, "Chip Enable Truth Table" on page 7). OE# must toggle to $V_{IH}$ or the device must be disabled (see Table 2) before further reads to update the status register latch. The Read Status Register command functions independently of the $V_{PEN}$ voltage.

During a program, block erase, set lock-bit, or clear lock-bit command sequence, only SR.7 is valid until the Write State Machine completes or suspends the operation. Device I/O pins $DQ_0$–$DQ_6$ and $DQ_8$–$DQ_{15}$ are placed in a high-impedance state. When the operation completes or suspends (check status register bit 7), all contents of the status register are valid when read.

### Table 16. Status Register Definitions

| WSMS | ESS | ECLBS | PSLBS | VPENS | R | DPS | R |
|---|---|---|---|---|---|---|---|
| bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | bit2 | bit 1 | bit 0 |
| High Z When Busy? | Status Register Bits | | | | Notes | | |
| No | SR.7 = WRITE STATE MACHINE STATUS<br>1 = Ready<br>0 = Busy | | | | Check STS or SR.7 to determine block erase, program, or lock-bit configuration completion. SR.6–SR.0 are not driven while SR.7 = “0.” | | |
| Yes | SR.6 = ERASE SUSPEND STATUS<br>1 = Block Erase Suspended<br>0 = Block Erase in Progress/Completed | | | | | | |
| Yes | SR.5 = ERASE AND CLEAR LOCK-BITSSTATUS<br>1 = Error in Block Erasure or Clear Lock-Bits<br>0 = Successful Block Erase or Clear Lock-Bits | | | | If both SR.5 and SR.4 are “1”s after a block erase or lock-bit configuration attempt, an improper command sequence was entered. | | |
| Yes | SR.4 = PROGRAM AND SET LOCK-BIT STATUS<br>1 = Error in Setting Lock-Bit<br>0 = Successful Set Block Lock Bit | | | | | | |
| Yes | SR.3 = PROGRAMMING VOLTAGE STATUS<br>1 = Low Programming Voltage Detected, Operation Aborted<br>0 = Programming Voltage OK | | | | SR.3 does not provide a continuous programming voltage level indication. The WSM interrogates and indicates the programming voltage level only after Block Erase, Program, Set Block Lock-Bit, or Clear Block Lock-Bits command sequences. | | |
| Yes | SR.2 = PROGRAM SUSPEND STATUS<br>1 = Program suspended<br>0 = Program in progress/completed | | | | | | |
| Yes | SR.1 = DEVICE PROTECT STATUS<br>1 = Block Lock-Bit Detected, Operation Abort<br>0 = Unlock | | | | SR.1 does not provide a continuous indication of block lock-bit values. The WSM interrogates the block lock-bits only after Block Erase, Program, or Lock-Bit configuration command sequences. It informs the system, depending on the attempted operation, if the block lock-bit is set. Read the block lock configuration codes using the Read Identifier Codes command to determine block lock-bit status. | | |
| Yes | SR.0 = RESERVED FOR FUTURE ENHANCEMENTS | | | | SR.0 is reserved for future use and should be masked when polling the status register. | | |

### Table 17. eXtended Status Register Definitions

| WBS | Reserved | |
|---|---|---|
| bit 7 | bits 6—0 | |
| High Z When Busy? | Status Register Bits | Notes |
| No | XSR.7 = WRITE BUFFER STATUS<br>1 = Write buffer available<br>0 = Write buffer not available | After a Buffer-Write command, XSR.7 = 1 indicates that a Write Buffer is available. |
| Yes | XSR.6–XSR.0 = RESERVED FOR FUTURE ENHANCEMENTS | SR.6–SR.0 are reserved for future use and should be masked when polling the status register. |

## 4.5 Clear Status Register Command

Status register bits SR.5, SR.4, SR.3, and SR.1 are set to "1"s by the WSM and can only be reset by the Clear Status Register command. These bits indicate various failure conditions (see Table 16). By allowing system software to reset these bits, several operations (such as cumulatively erasing or locking multiple blocks or writing several bytes in sequence) may be performed. The status register may be polled to determine if an error occurred during the sequence.

To clear the status register, the Clear Status Register command (50H) is written. It functions independently of the applied $V_{PEN}$ voltage. The Clear Status Register command is only valid when the WSM is off or the device is suspended.

## 4.6 Block Erase Command

Erase is executed one block at a time and initiated by a two-cycle command. A block erase setup is first written, followed by an block erase confirm. This command sequence requires an appropriate address within the block to be erased (erase changes all block data to FFH). Block preconditioning, erase, and verify are handled internally by the WSM (invisible to the system). After the two-cycle block erase sequence is written, the device automatically outputs status register data when read (see Figure 10, "Block Erase Flowchart" on page 33). The CPU can detect block erase completion by analyzing the output of the STS pin or status register bit SR.7. Toggle OE#, $CE_0$, $CE_1$, or $CE_2$ to update the status register.

When the block erase is complete, status register bit SR.5 should be checked. If a block erase error is detected, the status register should be cleared before system software attempts corrective actions. The CUI remains in read status register mode until a new command is issued.

This two-step command sequence of set-up followed by execution ensures that block contents are not accidentally erased. An invalid Block Erase command sequence will result in both status register bits SR.4 and SR.5 being set to "1." Also, reliable block erasure can only occur when $V_{CC}$ is valid and $V_{PEN} = V_{PENH}$. If block erase is attempted while $V_{PEN} \leq V_{PENLK}$, SR.3 and SR.5 will be set to "1." Successful block erase requires that the corresponding block lock-bit be cleared. If block erase is attempted when the corresponding block lock-bit is set, SR.1 and SR.5 will be set to "1."

## 4.7 Block Erase Suspend Command

The Block Erase Suspend command allows block-erase interruption to read or program data in another block of memory. Once the block erase process starts, writing the Block Erase Suspend command requests that the WSM suspend the block erase sequence at a predetermined point in the algorithm. The device outputs status register data when read after the Block Erase Suspend command is written. Polling status register bit SR.7 then SR.6 can determine when the block erase operation has been suspended (both will be set to "1"). In default mode, STS will also transition to $V_{OH}$. Specification $t_{WHRH}$ defines the block erase suspend latency.

At this point, a Read Array command can be written to read data from blocks other than that which is suspended. A program command sequence can also be issued during erase suspend to program data in other blocks. During a program operation with block erase suspended, status register bit SR.7 will return to "0" and STS output (in default mode) will transition to $V_{OL}$. However, SR.6 will remain "1" to indicate block erase suspend status. Using the Program Suspend command, a program operation can also be suspended. Resuming a suspended programming operation by

issuing the Program Resume command allows continuing of the suspended programming operation. To resume the suspended erase, the user must wait for the programming operation to complete before issuing the Block Erase Resume command.

The only other valid commands while block erase is suspended are Read Query, Read Status Register, Clear Status Register, Configure, and Block Erase Resume. After a Block Erase Resume command is written to the flash memory, the WSM will continue the block erase process. Status register bits SR.6 and SR.7 will automatically clear and STS (in default mode) will return to $V_{OL}$. After the Erase Resume command is written, the device automatically outputs status register data when read (see Figure 11, "Block Erase Suspend/Resume Flowchart" on page 34). $V_{PEN}$ must remain at $V_{PENH}$ (the same $V_{PEN}$ level used for block erase) while block erase is suspended. Block erase cannot resume until program operations initiated during block erase suspend have completed.

## 4.8 Write to Buffer Command

To program the flash device, a Write to Buffer command sequence is initiated. A variable number of bytes, up to the buffer size, can be loaded into the buffer and written to the flash device. First, the Write to Buffer Setup command is issued along with the Block Address (see Figure 7, "Write to Buffer Flowchart" on page 30). At this point, the eXtended Status Register (XSR, see Table 17) information is loaded and XSR.7 reverts to "buffer available" status. If XSR.7 = 0, the write buffer is not available. To retry, continue monitoring XSR.7 by issuing the Write to Buffer setup command with the Block Address until XSR.7 = 1. When XSR.7 transitions to a "1," the buffer is ready for loading.

Now a word/byte count is given to the part with the Block Address. On the next write, a device start address is given along with the write buffer data. Subsequent writes provide additional device addresses and data, depending on the count. All subsequent addresses must lie within the start address plus the count.

Internally, this device programs many flash cells in parallel. Because of this parallel programming, maximum programming performance and lower power are obtained by aligning the start address at the beginning of a write buffer boundary (i.e., $A_4$–$A_0$ of the start address = 0).

After the final buffer data is given, a Write Confirm command is issued. This initiates the WSM (Write State Machine) to begin copying the buffer data to the flash array. If a command other than Write Confirm is written to the device, an "Invalid Command/Sequence" error will be generated and Status Register bits SR.5 and SR.4 will be set to a "1." For additional buffer writes, issue another Write to Buffer Setup command and check XSR.7.

If an error occurs while writing, the device will stop writing, and status register bit SR.4 will be set to a "1" to indicate a program failure. The internal WSM verify only detects errors for "1"s that do not successfully program to "0"s. If a program error is detected, the status register should be cleared. Any time SR.4 and/or SR.5 is set (e.g., a media failure occurs during a program or an erase), the device will not accept any more Write to Buffer commands. Additionally, if the user attempts to program past an erase block boundary with a Write to Buffer command, the device will abort the write to buffer operation. This will generate an "Invalid Command/Sequence" error and status register bits SR.5 and SR.4 will be set to a "1."

Reliable buffered writes can only occur when $V_{PEN} = V_{PENH}$. If a buffered write is attempted while $V_{PEN} \leq V_{PENLK}$, status register bits SR.4 and SR.3 will be set to "1." Buffered write attempts with invalid $V_{CC}$ and $V_{PEN}$ voltages produce spurious results and should not be attempted. Finally, successful programming requires that the corresponding block lock-bit be reset. If a buffered write is attempted when the corresponding block lock-bit is set, SR.1 and SR.4 will be set to "1."

## 4.9 Byte/Word Program Commands

Byte/Word program is executed by a two-cycle command sequence. Byte/Word program setup (standard 40H or alternate 10H) is written followed by a second write that specifies the address and data (latched on the rising edge of WE#). The WSM then takes over, controlling the program and program verify algorithms internally. After the program sequence is written, the device automatically outputs status register data when read (see Figure 8, "Byte/Word Program Flowchart" on page 31). The CPU can detect the completion of the program event by analyzing the STS pin or status register bit SR.7.

When program is complete, status register bit SR.4 should be checked. If a program error is detected, the status register should be cleared. The internal WSM verify only detects errors for "1"s that do not successfully program to "0"s. The CUI remains in read status register mode until it receives another command.

Reliable byte/word programs can only occur when $V_{CC}$ and $V_{PEN}$ are valid. If a byte/word program is attempted while $V_{PEN} \leq V_{PENLK}$, status register bits SR.4 and SR.3 will be set to "1." Successful byte/word programs require that the corresponding block lock-bit be cleared. If a byte/word program is attempted when the corresponding block lock-bit is set, SR.1 and SR.4 will be set to "1."

## 4.10 Program Suspend Command

The Program Suspend command allows program interruption to read data in other flash memory locations. Once the programming process starts (either by initiating a write to buffer or byte/word program operation), writing the Program Suspend command requests that the WSM suspend the program sequence at a predetermined point in the algorithm. The device continues to output status register data when read after the Program Suspend command is written. Polling status register bits SR.7 can determine when the programming operation has been suspended. When SR.7 = 1, SR.2 should also be set to "1", indicating that the device is in the program suspend mode. STS in level RY/BY# mode will also transition to $V_{OH}$. Specification $t_{WHRH1}$ defines the program suspend latency.

At this point, a Read Array command can be written to read data from locations other than that which is suspended. The only other valid commands while programming is suspended are Read Query, Read Status Register, Clear Status Register, Configure, and Program Resume. After a Program Resume command is written, the WSM will continue the programming process. Status register bits SR.2 and SR.7 will automatically clear and STS in RY/BY# mode will return to $V_{OL}$. After the Program Resume command is written, the device automatically outputs status register data when read. $V_{PEN}$ must remain at $V_{PENH}$ and $V_{CC}$ must remain at valid $V_{CC}$ levels (the same $V_{PEN}$ and $V_{CC}$ levels used for programming) while in program suspend mode. Refer to Figure 9, "Program Suspend/Resume Flowchart" on page 32.

## 4.11 Set Read Configuration Command

This command is not support on this product. This device will default to the asynchronous page mode. If this command is given to the device it will not effect the operation of the device.

### 4.11.1 Read Configuration

The device will support both asynchronous page mode and standard word/byte reads. No configuration is required.

Status register and identifier only support standard word/byte single read operations.

**Table 18. Read Configuration Register Definition**

| RM | R | R | R | R | R | R | R |
|---|---|---|---|---|---|---|---|
| 16 ($A_{16}$) | 15 | 14 | 13 | 12 | 11 | 10 | 9 |
| R | R | R | R | R | R | R | R |
| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |

| | Notes |
|---|---|
| RCR.16 = READ MODE (RM)<br>0 = Standard Word/Byte Reads Enabled (Default)<br>1 = Page-Mode Reads Enabled | Read mode configuration effects reads from the flash array. Status register, query, and identifier reads support standard word/byte read cycles. |
| RCR.15–1 = RESERVED FOR FUTURE ENHANCEMENTS (R) | These bits are reserved for future use. Set these bits to “0.” |

## 4.12 Configuration Command

The Status (STS) pin can be configured to different states using the Configuration command. Once the STS pin has been configured, it remains in that configuration until another configuration command is issued or RP# is asserted low. Initially, the STS pin defaults to RY/BY# operation where RY/BY# low indicates that the state machine is busy. RY/BY# high indicates that the state machine is ready for a new operation or suspended. Table 19, “Configuration Coding Definitions” on page 26 displays the possible STS configurations.

To reconfigure the Status (STS) pin to other modes, the Configuration command is given followed by the desired configuration code. The three alternate configurations are all pulse mode for use as a system interrupt as described below. For these configurations, bit 0 controls Erase Complete interrupt pulse, and bit 1 controls Program Complete interrupt pulse. Supplying the 00h configuration code with the Configuration command resets the STS pin to the default RY/BY# level mode. The possible configurations and their usage are described in Table 19, “Configuration Coding Definitions” on page 26. The Configuration command may only be given when the device is not busy or suspended. Check SR.7 for device status. An invalid configuration code will result in both status register bits SR.4 and SR.5 being set to “1.” When configured in one of the pulse modes, the STS pin pulses low with a typical pulse width of 250 ns.

### Table 19. Configuration Coding Definitions

| Reserved | | Pulse on Program Complete[1] | Pulse on Erase Compete[1] |
|---|---|---|---|
| bits 7—2 | | bit 1 | bit 0 |
| $DQ_7$–$DQ_2$ = Reserved<br>$DQ_1$–$DQ_0$ = STS Pin Configuration Codes<br>00 = default, level mode RY/BY# (device ready) indication<br>01 = pulse on Erase complete<br>10 = pulse on Program complete<br>11 = pulse on Erase or Program Complete<br>Configuration Codes 01b, 10b, and 11b are all pulse mode such that the STS pin pulses low then high when the operation indicated by the given configuration is completed.<br>Configuration Command Sequences for STS pin configuration (masking bits $DQ_7$–$DQ_2$ to 00h) are as follows:<br>Default RY/BY# level mode: B8h, 00h<br>ER INT (Erase Interrupt): B8h, 01h<br>Pulse-on-Erase Complete<br>PR INT (Program Interrupt): B8h, 02h<br>Pulse-on-Program Complete<br>ER/PR INT (Erase or Program Interrupt): B8h, 03h<br>Pulse-on-Erase or Program Complete | $DQ_7$–$DQ_2$ are reserved for future use.<br>default ($DQ_1$–$DQ_0$ = 00) RY/BY#, level mode<br>— used to control HOLD to a memory controller to prevent accessing a flash memory subsystem while any flash device's WSM is busy.<br>configuration 01 ER INT, pulse mode<br>— used to generate a system interrupt pulse when any flash device in an array has completed a Block Erase. Helpful for reformatting blocks after file system free space reclamation or “cleanup”<br>configuration 10 PR INT, pulse mode<br>— used to generate a system interrupt pulse when any flash device in an array has complete a Program operation. Provides highest performance for servicing continuous buffer write operations.<br>configuration 11 ER/PR INT, pulse mode<br>— used to generate system interrupts to trigger servicing of flash arrays when either erase or program operations are completed when a common interrupt service routine is desired. | | |

**NOTE:** 1. When the device is configured in one of the pulse modes, the STS pin pulses low with a typical pulse width of 250 ns.

## 4.13 Set Block Lock-Bit Commands

A flexible block locking and unlocking scheme is enabled via block lock-bits. The block lock-bits gate program and erase operations. Individual block lock-bits can be set using the Set Block Lock-Bit command. This command is invalid while the WSM is running or the device is suspended.

Set block lock-bit commands are executed by a two-cycle sequence. The set block setup along with appropriate block address is followed by either the set block lock-bit confirm (and an address within the block to be locked). The WSM then controls the set lock-bit algorithm. After the sequence is written, the device automatically outputs status register data when read (see Figure 12 on page 35). The CPU can detect the completion of the set lock-bit event by analyzing the STS pin output or status register bit SR.7.

When the set lock-bit operation is complete, status register bit SR.4 should be checked. If an error is detected, the status register should be cleared. The CUI will remain in read status register mode until a new command is issued.

This two-step sequence of set-up followed by execution ensures that lock-bits are not accidentally set. An invalid Set Block Lock-Bit command will result in status register bits SR.4 and SR.5 being set to “1.” Also, reliable operations occur only when $V_{CC}$ and $V_{PEN}$ are valid. With $V_{PEN} \leq V_{PENLK}$, lock-bit contents are protected against alteration.

## 4.14 Clear Block Lock-Bits Command

All set block lock-bits are cleared in parallel via the Clear Block Lock-Bits command. Block lock-bits can be cleared using only the Clear Block Lock-Bits command. This command is invalid while the WSM is running or the device is suspended.

Clear block lock-bits command is executed by a two-cycle sequence. A clear block lock-bits setup is first written. The device automatically outputs status register data when read (see Figure 13 on page 36). The CPU can detect completion of the clear block lock-bits event by analyzing the STS pin output or status register bit SR.7.

When the operation is complete, status register bit SR.5 should be checked. If a clear block lock-bit error is detected, the status register should be cleared. The CUI will remain in read status register mode until another command is issued.

This two-step sequence of set-up followed by execution ensures that block lock-bits are not accidentally cleared. An invalid Clear Block Lock-Bits command sequence will result in status register bits SR.4 and SR.5 being set to "1." Also, a reliable clear block lock-bits operation can only occur when $V_{CC}$ and $V_{PEN}$ are valid. If a clear block lock-bits operation is attempted while $V_{PEN} \leq V_{PENLK}$, SR.3 and SR.5 will be set to "1."

If a clear block lock-bits operation is aborted due to $V_{PEN}$ or $V_{CC}$ transitioning out of valid range, block lock-bit values are left in an undetermined state. A repeat of clear block lock-bits is required to initialize block lock-bit contents to known values.

## 4.15 Protection Register Program Command

The 3 Volt Intel StrataFlash memory includes a 128-bit protection register that can be used to increase the security of a system design. For example, the number contained in the protection register can be used to "mate" the flash component with other system components such as the CPU or ASIC, preventing device substitution.

The 128-bits of the protection register are divided into two 64-bit segments. One of the segments is programmed at the Intel factory with a unique 64-bit number, which is unchangeable. The other segment is left blank for customer designers to program as desired. Once the customer segment is programmed, it can be locked to prevent reprogramming.

### 4.15.1 Reading the Protection Register

The protection register is read in the identification read mode. The device is switched to this mode by writing the Read Identifier command (90H). Once in this mode, read cycles from addresses shown in Table 20 or Table 21 retrieve the specified information. To return to read array mode, write the Read Array command (FFH).

### 4.15.2 Programming the Protection Register

The protection register bits are programmed using the two-cycle Protection Program command. The 64-bit number is programmed 16 bits at a time for word-wide parts and eight bits at a time for byte-wide parts. First write the Protection Program Setup command, C0H. The next write to the

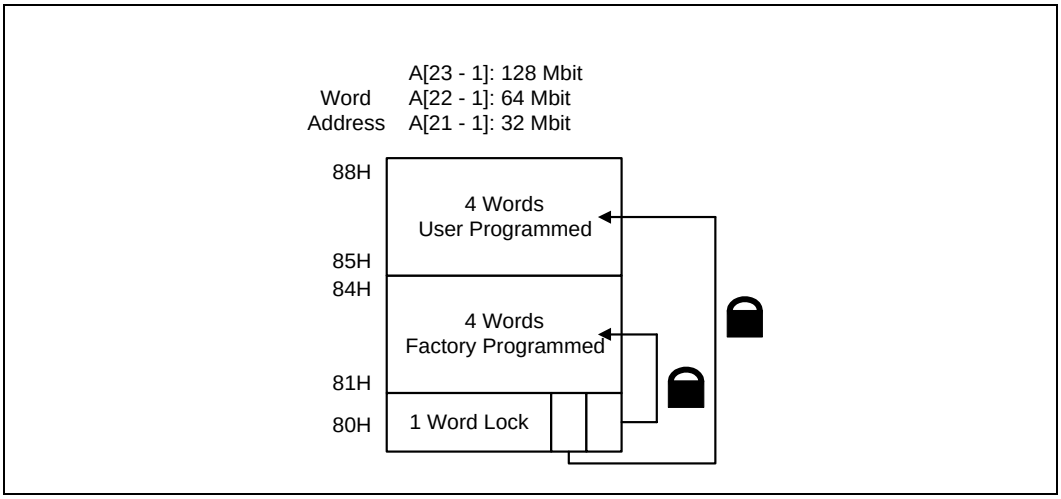
device will latch in address and data and program the specified location. The allowable addresses are shown in Table 20 or Table 21. See Figure 14, “Protection Register Programming Flowchart” on page 37

Any attempt to address Protection Program commands outside the defined protection register address space will result in a status register error (program error bit SR.4 will be set to 1). Attempting to program a locked protection register segment will result in a status register error (program error bit SR.4 and lock error bit SR.1 will be set to 1).

### 4.15.3 Locking the Protection Register

The user-programmable segment of the protection register is lockable by programming Bit 1 of the PR-LOCK location to 0. Bit 0 of this location is programmed to 0 at the Intel factory to protect the unique device number. Bit 1 is set using the Protection Program command to program “FFFD” to the PR-LOCK location. After these bits have been programmed, no further changes can be made to the values stored in the protection register. Protection Program commands to a locked section will result in a status register error (program error bit SR.4 and Lock Error bit SR.1 will be set to 1). Protection register lockout state is not reversible.

**Figure 6. Protection Register Memory Map**




**NOTE:** $A_0$ is not used in x16 mode when accessing the protection register map (See Table 20 for x16 addressing). For x8 mode $A_0$ is used (See Table 21 for x8 addressing).

### Table 20. Word-Wide Protection Register Addressing

| Word | Use | A8 | A7 | A6 | A5 | A4 | A3 | A2 | A1 |
|---|---|---|---|---|---|---|---|---|---|
| LOCK | Both | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | Factory | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | Factory | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 2 | Factory | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 3 | Factory | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 4 | User | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 5 | User | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 6 | User | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 7 | User | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

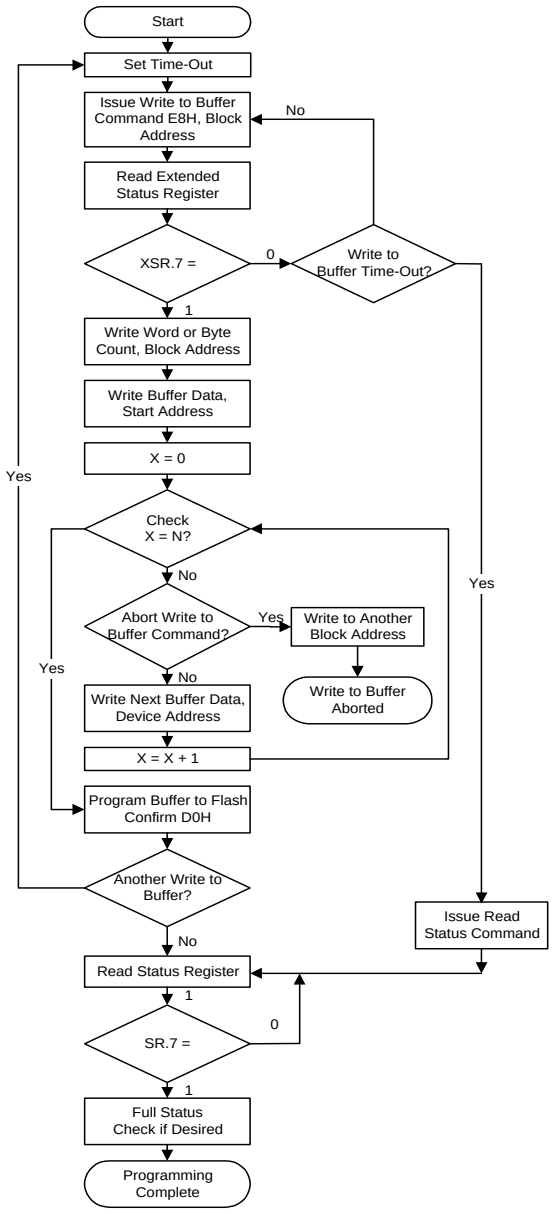
**NOTE:** 1. All address lines not specified in the above table must be 0 when accessing the Protection Register, i.e., $A_{23}$–$A_9$ = 0.

### Table 21. Byte-Wide Protection Register Addressing

| Byte | Use | A8 | A7 | A6 | A5 | A4 | A3 | A2 | A1 |
|---|---|---|---|---|---|---|---|---|---|
| LOCK | Both | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| LOCK | Both | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | Factory | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | Factory | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 2 | Factory | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 3 | Factory | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 4 | Factory | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 5 | Factory | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 6 | Factory | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 7 | Factory | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 8 | User | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 9 | User | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| A | User | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| B | User | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| C | User | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| D | User | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| E | User | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| F | User | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

**NOTE:** 1. All address lines not specified in the above table must be 0 when accessing the Protection Register, i.e., $A_{23}$–$A_9$ = 0.

## Figure 7. Write to Buffer Flowchart



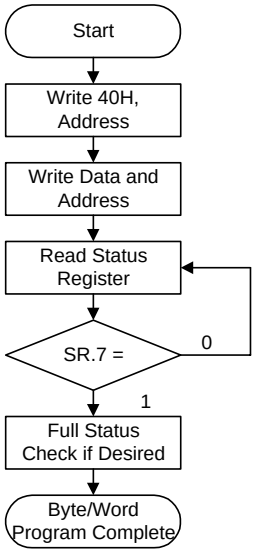

| Bus Operation | Command | Comments |
|---|---|---|
| Write | Write to Buffer | Data = E8H<br>Block Address |
| Read | | XSR. 7 = Valid<br>Addr = Block Address |
| Standby | | Check XSR. 7<br>1 = Write Buffer Available<br>0 = Write Buffer Not Available |
| Write<br>(Note 1, 2) | | Data = N = Word/Byte Count<br>N = 0 Corresponds to Count = 1<br>Addr = Block Address |
| Write<br>(Note 3, 4) | | Data = Write Buffer Data<br>Addr = Device Start Address |
| Write<br>(Note 5, 6) | | Data = Write Buffer Data<br>Addr = Device Address |
| Write | Program Buffer to Flash Confirm | Data = D0H<br>Addr = Block Address |
| Read<br>(Note 7) | | Status Register Data with the Device Enabled, OE# Low Updates SR<br>Addr = Block Address |
| Standby | | Check SR.7<br>1 = WSM Ready<br>0 = WSM Busy |

1. Byte or word count values on $DQ_0$ - $DQ_7$ are loaded into the count register. Count ranges on this device for byte mode are N = 00H to 1FH and for word mode are N = 0000H to 000FH.
2. The device now outputs the status register when read (XSR is no longer available).
3. Write Buffer contents will be programmed at the device start address or destination flash address.
4. Align the start address on a Write Buffer boundary for maximum programming performance (i.e., $A_4$ - $A_0$ of the start address = 0).
5. The device aborts the Write to Buffer command if the current address is outside of the original block address.
6. The status register indicates an "improper command sequence" if the Write to Buffer command is aborted. Follow this with a Clear Status Register command.
7. Toggling OE# (low to high to low) updates the status register. This can be done in place of issuing the Read Status Register command.

Full status check can be done after all erase and write sequences complete. Write FFH after the last operation to reset the device to read array mode.

0606_07A

## Figure 8. Byte/Word Program Flowchart



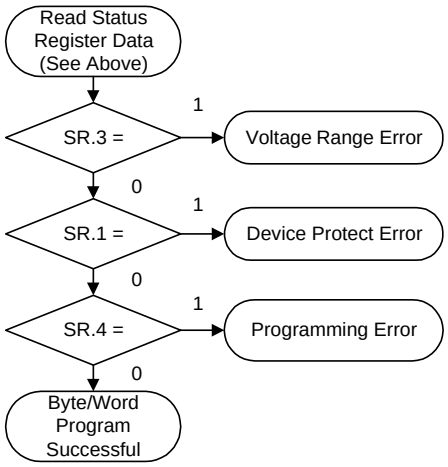

| Bus Operation | Command | Comments |
|---|---|---|
| Write | Setup Byte/ Word Program | Data = 40H<br>Addr = Location to Be Programmed |
| Write | Byte/Word Program | Data = Data to Be Programmed<br>Addr = Location to Be Programmed |
| Read (Note 1) | | Status Register Data |
| Standby | | Check SR.7<br>1 = WSM Ready<br>0 = WSM Busy |

1. Toggling OE# (low to high to low) updates the status register. This can be done in place of issuing the Read Status Register command. Repeat for subsequent programming operations.

SR full status check can be done after each program operation, or after a sequence of programming operations.

Write FFH after the last program operation to place device in read array mode.

**FULL STATUS CHECK PROCEDURE**



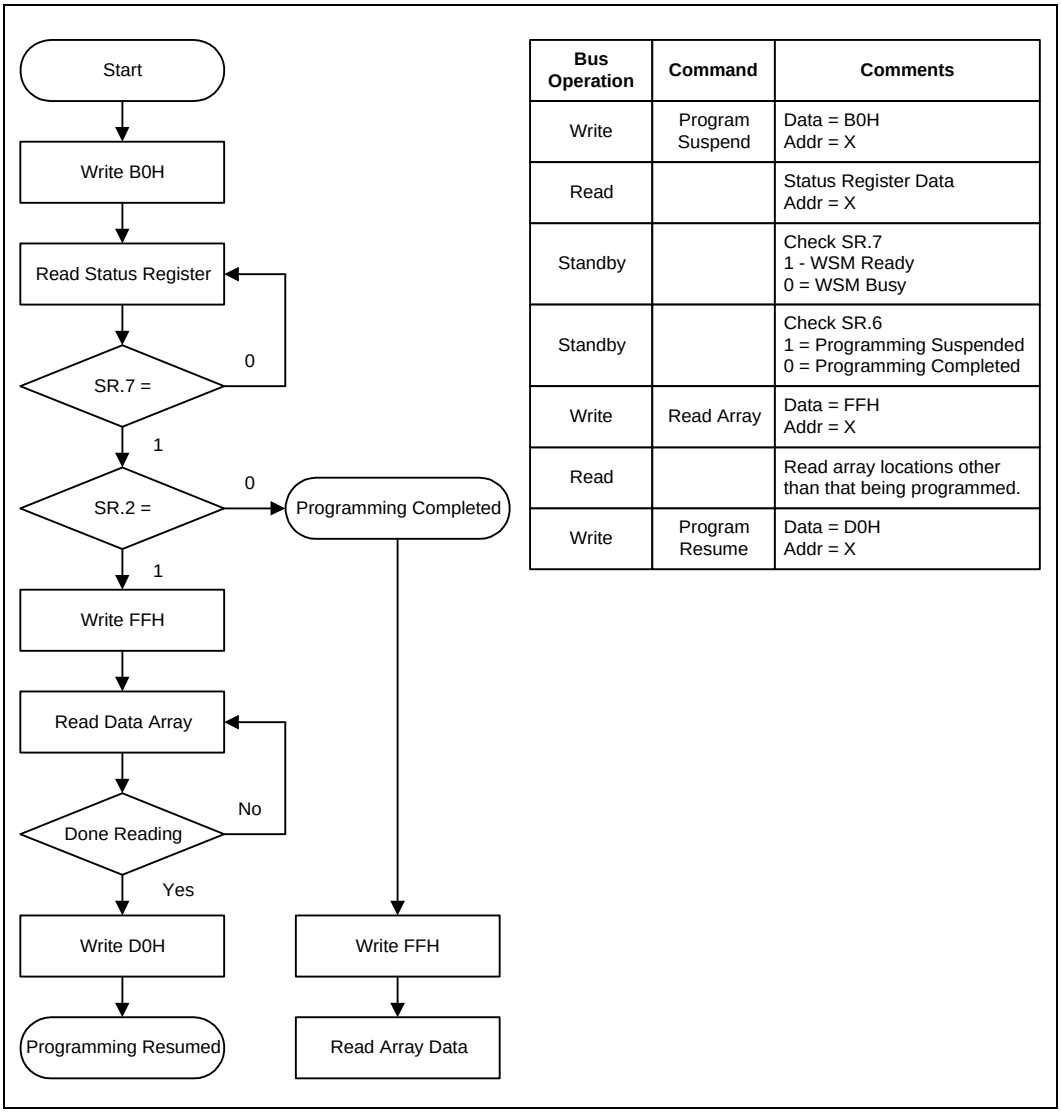

| Bus Operation | Command | Comments |
|---|---|---|
| Standby | | Check SR.3<br>1 = Programming to Voltage Error Detect |
| Standby | | Check SR.1<br>1 = Device Protect Detect<br>RP# = $V_{IH}$, Block Lock-Bit Is Set<br>Only required for systems implemeting lock-bit configuration. |
| Standby | | Check SR.4<br>1 = Programming Error |

Toggling OE# (low to high to low) updates the status register. This can be done in place of issuing the Read Status Register command. Repeat for subsequent programming operations.

SR.4, SR.3 and SR.1 are only cleared by the Clear Status Register command in cases where multiple locations are programmed before full status is checked.

If an error is detected, clear the status register before attempting retry or other error recovery.

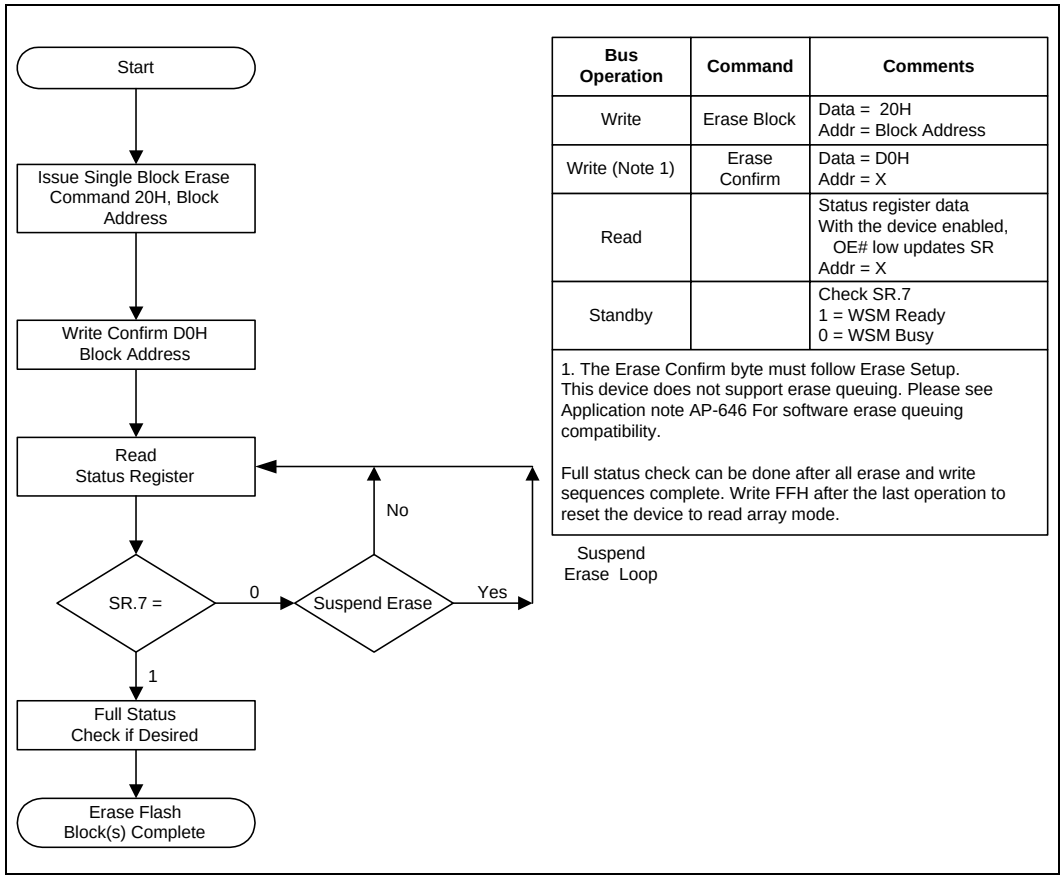
## Figure 9. Program Suspend/Resume Flowchart




| Bus Operation | Command | Comments |
|---|---|---|
| Write | Program Suspend | Data = B0H<br>Addr = X |
| Read | | Status Register Data<br>Addr = X |
| Standby | | Check SR.7<br>1 - WSM Ready<br>0 = WSM Busy |
| Standby | | Check SR.6<br>1 = Programming Suspended<br>0 = Programming Completed |
| Write | Read Array | Data = FFH<br>Addr = X |
| Read | | Read array locations other than that being programmed. |
| Write | Program Resume | Data = D0H<br>Addr = X |

0606_08

## Figure 10. Block Erase Flowchart




| Bus Operation | Command | Comments |
|---|---|---|
| Write | Erase Block | Data = 20H<br>Addr = Block Address |
| Write (Note 1) | Erase Confirm | Data = D0H<br>Addr = X |
| Read | | Status register data<br>With the device enabled,<br>OE# low updates SR<br>Addr = X |
| Standby | | Check SR.7<br>1 = WSM Ready<br>0 = WSM Busy |

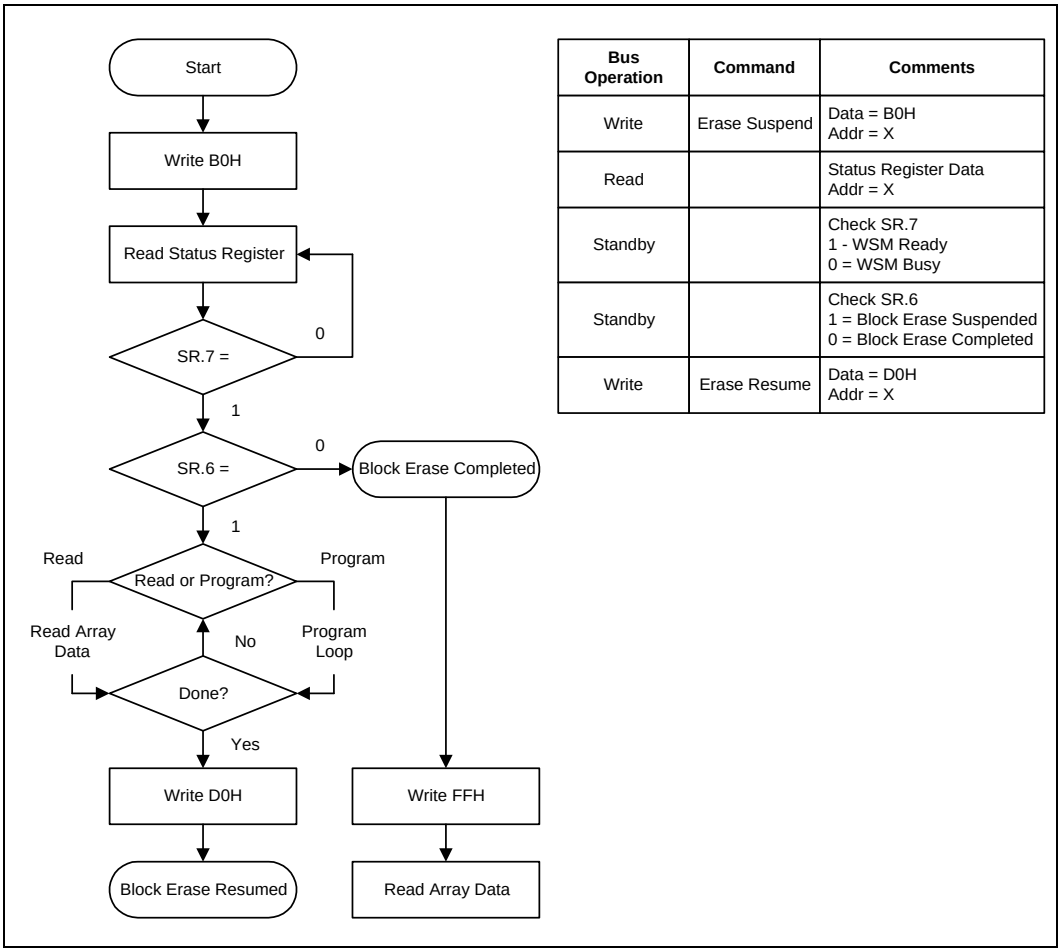
1. The Erase Confirm byte must follow Erase Setup. This device does not support erase queuing. Please see Application note AP-646 For software erase queuing compatibility.

Full status check can be done after all erase and write sequences complete. Write FFH after the last operation to reset the device to read array mode.

Suspend Erase Loop

0606_09

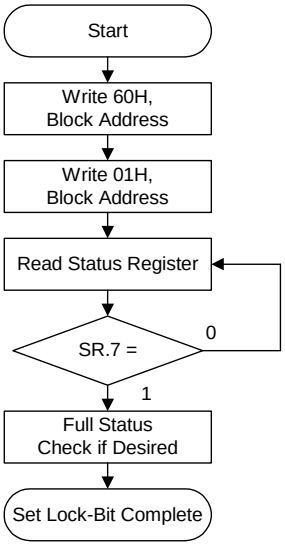
## Figure 11. Block Erase Suspend/Resume Flowchart




| Bus Operation | Command | Comments |
|---|---|---|
| Write | Erase Suspend | Data = B0H<br>Addr = X |
| Read | | Status Register Data<br>Addr = X |
| Standby | | Check SR.7<br>1 - WSM Ready<br>0 = WSM Busy |
| Standby | | Check SR.6<br>1 = Block Erase Suspended<br>0 = Block Erase Completed |
| Write | Erase Resume | Data = D0H<br>Addr = X |

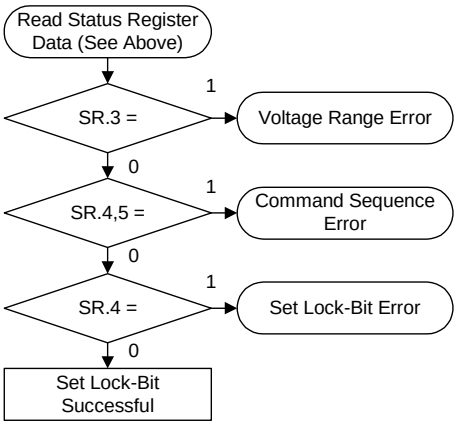
0606_10

## Figure 12. Set Block Lock-Bit Flowchart




| Bus Operation | Command | Comments |
|---|---|---|
| Write | Set Block Lock-Bit Setup | Data = 60H<br>Addr =Block Address |
| Write | Set Block Lock-Bit Confirm | Data = 01H<br>Addr = Block Address |
| Read | | Status Register Data |
| Standby | | Check SR.7<br>1 = WSM Ready<br>0 = WSM Busy |
| Repeat for subsequent lock-bit operations.<br><br>Full status check can be done after each lock-bit set operation or after a sequence of lock-bit set operations.<br><br>Write FFH after the last lock-bit set operation to place device in read array mode. | | |

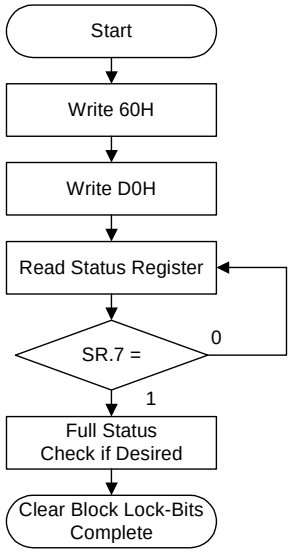
**FULL STATUS CHECK PROCEDURE**




| Bus Operation | Command | Comments |
|---|---|---|
| Standby | | Check SR.3<br>1 = Programming Voltage Error Detect |
| Standby | | Check SR.4, 5<br>Both 1 = Command Sequence Error |
| Standby | | Check SR.4<br>1 = Set Lock-Bit Error |
| SR.5, SR.4 and SR.3 are only cleared by the Clear Status Register command, in cases where multiple lock-bits are set before full status is checked.<br><br>If an error is detected, clear the status register before attempting retry or other error recovery. | | |

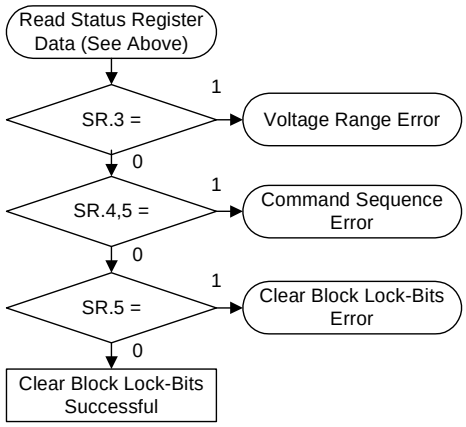
0606_11b

## Figure 13. Clear Lock-Bit Flowchart




| Bus Operation | Command | Comments |
|---|---|---|
| Write | Clear Block Lock-Bits Setup | Data = 60H<br>Addr = X |
| Write | Clear Block or Lock-Bits Confirm | Data = D0H<br>Addr = X |
| Read | | Status Register Data |
| Standby | | Check SR.7<br>1 = WSM Ready<br>0 = WSM Busy |
| Write FFH after the clear lock-bits operation to place device in read array mode. | | |

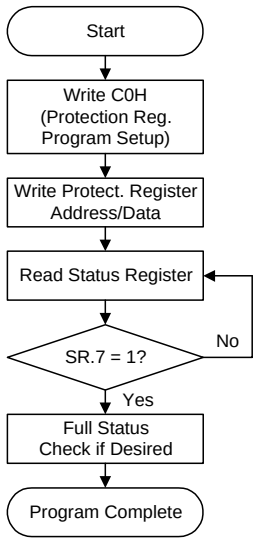
**FULL STATUS CHECK PROCEDURE**




| Bus Operation | Command | Comments |
|---|---|---|
| Standby | | Check SR.3<br>1 = Programming Voltage Error Detect |
| Standby | | Check SR.4, 5<br>Both 1 = Command Sequence Error |
| Standby | | Check SR.5<br>1 = Clear Block Lock-Bits Error |
| SR.5, SR.4, and SR.3 are only cleared by the Clear Status Register command.<br><br>If an error is detected, clear the status register before attempting retry or other error recovery. | | |

0606_12b

## Figure 14. Protection Register Programming Flowchart




| Bus Operation | Command | Comments |
|---|---|---|
| Write | Protection Program Setup | Data = C0H |
| Write | Protection Program | Data = Data to Program<br>Addr = Location to Program |
| Read | | Status Register Data Toggle CE# or OE# to Update Status Register Data |
| Standby | | Check SR.7<br>1 = WSM Ready<br>0 = WSM Busy |

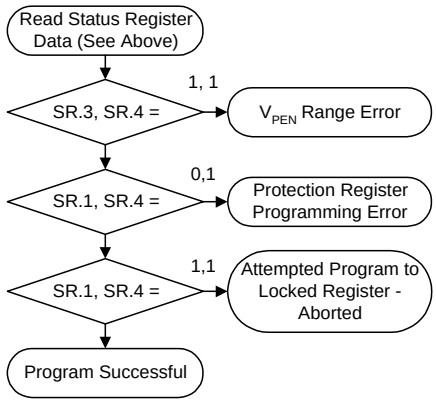
Protection Program operations can only be addressed within the protection register address space. Addresses outside the defined space will return an error.

Repeat for subsequent programming operations.

SR Full Status Check can be done after each program or after a sequence of program operations.

Write FFH after the last program operation to reset device to read array mode.

**FULL STATUS CHECK PROCEDURE**




| Bus Operation | Command | Comments |
|---|---|---|
| Standby | | SR.1 SR.3 SR.4<br>0 1 1 $V_{PEN}$ Low |
| Standby | | 0 0 1 Prot. Reg. Prog. Error |
| Standby | | 1 0 1 Register Locked: Aborted |

SR.3 MUST be cleared, if set during a program attempt, before further attempts are allowed by the Write State Machine.

SR.1, SR.3 and SR.4 are only cleared by the Clear Staus Register Command, in cases of multiple protection register program operations before full status is checked.

If an error is detected, clear the status register before attempting retry or other error recovery.

# 5.0 Design Considerations

## 5.1 Three-Line Output Control

The device will often be used in large memory arrays. Intel provides five control inputs ($CE_0$, $CE_1$, $CE_2$, OE#, and RP#) to accommodate multiple memory connections. This control provides for:

a. Lowest possible memory power dissipation.

b. Complete assurance that data bus contention will not occur.

To use these control inputs efficiently, an address decoder should enable the device (see Table 2) while OE# should be connected to all memory devices and the system's READ# control line. This assures that only selected memory devices have active outputs while de-selected memory devices are in standby mode. RP# should be connected to the system POWERGOOD signal to prevent unintended writes during system power transitions. POWERGOOD should also toggle during system reset.

## 5.2 STS and Block Erase, Program, and Lock-Bit Configuration Polling

STS is an open drain output that should be connected to $V_{CCQ}$ by a pull-up resistor to provide a hardware method of detecting block erase, program, and lock-bit configuration completion. It is recommended that a 2.5k resister be used between STS# and $V_{CCQ}$. In default mode, it transitions low after block erase, program, or lock-bit configuration commands and returns to High Z when the WSM has finished executing the internal algorithm. For alternate configurations of the STS pin, see the Configuration command.

STS can be connected to an interrupt input of the system CPU or controller. It is active at all times. STS, in default mode, is also High Z when the device is in block erase suspend (with programming inactive), program suspend, or in reset/power-down mode.

## 5.3 Power Supply Decoupling

Flash memory power switching characteristics require careful device decoupling. System designers are interested in three supply current issues; standby current levels, active current levels and transient peaks produced by falling and rising edges of $CE_0$, $CE_1$, $CE_2$, and OE#. Transient current magnitudes depend on the device outputs' capacitive and inductive loading. Two-line control and proper decoupling capacitor selection will suppress transient voltage peaks. Since Intel StrataFlash memory devices draw their power from three $V_{CC}$ pins (these devices do not include a $V_{PP}$ pin), it is recommended that systems without separate power and ground planes attach a 0.1 µF ceramic capacitor between each of the device's three $V_{CC}$ pins (this includes $V_{CCQ}$) and ground. These high-frequency, low-inductance capacitors should be placed as close as possible to package leads on each Intel StrataFlash memory device. Each device should have a 0.1 µF ceramic capacitor connected between its $V_{CC}$ and GND. These high-frequency, low inductance capacitors should be placed as close as possible to package leads. Additionally, for every eight devices, a 4.7 µF electrolytic capacitor should be placed between $V_{CC}$ and GND at the array's power supply connection. The bulk capacitor will overcome voltage slumps caused by PC board trace inductance.

## 5.4 Input Signal Transitions - Reducing Overshoots and Undershoots When Using Buffers or Transceivers

As faster, high-drive devices such as transceivers or buffers drive input signals to flash memory devices, overshoots and undershoots can sometimes cause input signals to exceed flash memory specifications. (See “Absolute Maximum Ratings” on page 40.) Many buffer/transceiver vendors now carry bus-interface devices with internal output-damping resistors or reduced-drive outputs. Internal output-damping resistors diminish the nominal output drive currents, while still leaving sufficient drive capability for most applications. These internal output-damping resistors help reduce unnecessary overshoots and undershoots. Transceivers or buffers with balanced- or light-drive outputs also reduce overshoots and undershoots by diminishing output-drive currents. When considering a buffer/transceiver interface design to flash, devices with internal output-damping resistors or reduced-drive outputs should be used to minimize overshoots and undershoots. For additional information, please refer to the AP-647 *5 Volt Intel StrataFlash™ Memory Design Guide*.

## 5.5 $V_{CC}$, $V_{PEN}$, RP# Transitions

Block erase, program, and lock-bit configuration are not guaranteed if $V_{PEN}$ or $V_{CC}$ falls outside of the specified operating ranges, or RP# ≠ $V_{IH}$. If RP# transitions to $V_{IL}$ during block erase, program, or lock-bit configuration, STS (in default mode) will remain low for a maximum time of $t_{PLPH} + t_{PHRH}$ until the reset operation is complete. Then, the operation will abort and the device will enter reset/power-down mode. The aborted operation may leave data partially corrupted after programming, or partially altered after an erase or lock-bit configuration. Therefore, block erase and lock-bit configuration commands must be repeated after normal operation is restored. Device power-off or RP# = $V_{IL}$ clears the status register.

The CUI latches commands issued by system software and is not altered by $V_{PEN}$, $CE_0$, $CE_1$, or $CE_2$ transitions, or WSM actions. Its state is read array mode upon power-up, after exit from reset/power-down mode, or after $V_{CC}$ transitions below $V_{LKO}$. $V_{CC}$ must be kept at or above $V_{PEN}$ during $V_{CC}$ transitions.

After block erase, program, or lock-bit configuration, even after $V_{PEN}$ transitions down to $V_{PENLK}$, the CUI must be placed in read array mode via the Read Array command if subsequent access to the memory array is desired. $V_{PEN}$ must be kept at or below $V_{CC}$ during $V_{PEN}$ transitions.

## 5.6 Power-Up/Down Protection

The device is designed to offer protection against accidental block erasure, programming, or lock-bit configuration during power transitions. Internal circuitry resets the CUI to read array mode at power-up.

A system designer must guard against spurious writes for $V_{CC}$ voltages above $V_{LKO}$ when $V_{PEN}$ is active. Since WE# must be low and the device enabled (see Table 2) for a command write, driving WE# to $V_{IH}$ or disabling the device will inhibit writes. The CUI’s two-step command sequence architecture provides added protection against data alteration.

Keeping $V_{PEN}$ below $V_{PENLK}$ prevents inadvertent data alteration. In-system block lock and unlock capability protects the device against inadvertent programming. The device is disabled while RP# = $V_{IL}$ regardless of its control inputs.

## 5.7 Power Dissipation

When designing portable systems, designers must consider battery power consumption not only during device operation, but also for data retention during system idle time. Flash memory's nonvolatility increases usable battery life because data is retained when system power is removed.

# 6.0 Electrical Specifications

## 6.1 Absolute Maximum Ratings

| Parameter | Maximum Rating |
|---|---|
| Temperature under Bias Expanded | –25 °C to +85 °C |
| Storage Temperature | –65 °C to +125 °C |
| Voltage On Any Pin | –2.0 V to +5.0 V(1) |
| Output Short Circuit Current | 100 mA(2) |

**NOTES:**
1. All specified voltages are with respect to GND. Minimum DC voltage is –0.5 V on input/output pins and –0.2 V on $V_{CC}$ and $V_{PEN}$ pins. During transitions, this level may undershoot to –2.0 V for periods <20 ns. Maximum DC voltage on input/output pins, $V_{CC}$, and $V_{PEN}$ is $V_{CC}$ +0.5 V which, during transitions, may overshoot to $V_{CC}$ +2.0 V for periods <20 ns.
2. Output shorted for no more than one second. No more than one output shorted at a time.

NOTICE: This datasheet contains preliminary information on new products in production. The specifications are subject to change without notice. Verify with your local Intel Sales office that you have the latest datasheet before finalizing a design.

***Warning:*** Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.

## 6.2 Operating Conditions

**Table 22. Temperature and $V_{CC}$ Operating Conditions**

| Symbol | Parameter | Notes | Min | Max | Unit | Test Condition |
|---|---|---|---|---|---|---|
| $T_A$ | Operating Temperature | | –25 | +85 | °C | Ambient Temperature |
| $V_{CC1}$ | $V_{CC1}$ Supply Voltage (2.7 V–3.6 V) | | 2.70 | 3.60 | V | |
| $V_{CC2}$ | $V_{CC2}$ Supply Voltage (3.0 V–3.6 V) | | 3.00 | 3.60 | V | |
| $V_{CCQ1}$ | $V_{CCQ1}$ Supply Voltage (2.7 V–3.6 V) | | 2.70 | 3.60 | V | |
| $V_{CCQ2}$ | $V_{CCQ2}$ Supply Voltage (3.0 V–3.6 V) | | 3.00 | 3.60 | V | |

## 6.3 Capacitance

$T_A$ = +25 °C, f = 1 MHz

| Symbol | Parameter[1] | Typ | Max | Unit | Condition |
|---|---|---|---|---|---|
| $C_{IN}$ | Input Capacitance | 6 | 8 | pF | $V_{IN}$ = 0.0 V |
| $C_{OUT}$ | Output Capacitance | 8 | 12 | pF | $V_{OUT}$ = 0.0 V |

**NOTES:**
1. Sampled, not 100% tested.

## 6.4 DC Characteristics

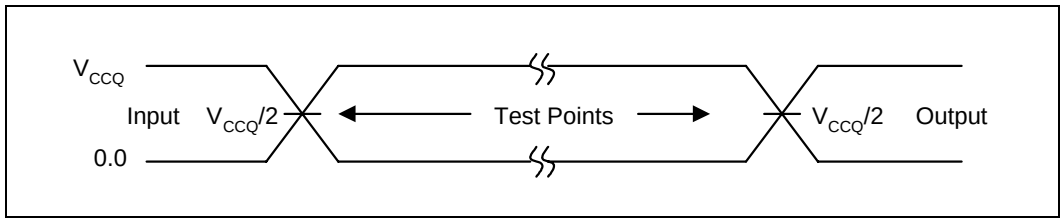
| Symbol | Parameter | Notes | Typ | Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|
| $I_{LI}$ | Input and $V_{PEN}$ Load Current | 1 | | ±1 | µA | $V_{CC} = V_{CC}$ Max; $V_{CCQ} = V_{CCQ}$ Max<br>$V_{IN} = V_{CCQ}$ or GND |
| $I_{LO}$ | Output Leakage Current | 1 | | ±10 | µA | $V_{CC} = V_{CC}$ Max; $V_{CCQ} = V_{CCQ}$ Max<br>$V_{IN} = V_{CCQ}$ or GND |
| $I_{LO}$ | Output Leakage Current | 1 | | ±10 | µA | $V_{CC} = V_{CC}$ Max; $V_{CCQ} = V_{CCQ}$ Max<br>$V_{IN} = V_{CCQ}$ or GND |
| $I_{CCS}$ | $V_{CC}$ Standby Current | 1,2,3,4 | 50 | 120 | µA | CMOS Inputs, $V_{CC} = V_{CC}$ Max,<br>Device is enabled (see Table 2, "Chip Enable Truth Table" on page 7),<br>RP# = $V_{CCQ}$ ± 0.2 V |
| | | | 0.71 | 2 | mA | TTL Inputs, $V_{CC} = V_{CC}$ Max,<br>Device is enabled (see Table 2), RP# = $V_{IH}$ |
| $I_{CCD}$ | $V_{CC}$ Power-Down Current | 4 | 50 | 120 | µA | RP# = GND ± 0.2 V, $I_{OUT}$ (STS) = 0 mA |
| $I_{CCR}$ | $V_{CC}$ Page Mode Read Current | 1,3,4 | 15 | 20 | mA | CMOS Inputs, $V_{CC} = V_{CC}$ Max, $V_{CCQ} = V_{CCQ}$ Max using standard 4 word page mode reads.<br>Device is enabled (see Table 2)<br>f = 5 MHz, $I_{OUT}$ = 0 mA |
| | | | 24 | 29 | mA | CMOS Inputs,$V_{CC} = V_{CC}$ Max, $V_{CCQ} = V_{CCQ}$ Max using standard 4 word page mode reads.<br>Device is enabled (see Table 2)<br>f = 33 MHz, $I_{OUT}$ = 0 mA |
| $I_{CCR}$ | $V_{CC}$ Byte Mode Read Current | 1,3,4 | 40 | 50 | mA | CMOS Inputs, $V_{CC} = V_{CC}$ Max, $V_{CCQ} = V_{CCQ}$ Max using standard word/byte single reads<br>Device is enabled (see Table 2)<br>f = 5 MHz, $I_{OUT}$ = 0 mA |
| $I_{CCW}$ | $V_{CC}$ Program or Set Lock-Bit Current | 1,4,5 | 35 | 60 | mA | CMOS Inputs, $V_{PEN} = V_{CC}$ |
| | | | 40 | 70 | mA | TTL Inputs, $V_{PEN} = V_{CC}$ |
| $I_{CCE}$ | $V_{CC}$ Block Erase or Clear Block Lock-Bits Current | 1,4,5 | 35 | 70 | mA | CMOS Inputs, $V_{PEN} = V_{CC}$ |
| | | | 40 | 80 | mA | TTL Inputs, $V_{PEN} = V_{CC}$ |
| $I_{CCWS}$<br>$I_{CCES}$ | $V_{CC}$ Program Suspend or Block Erase Suspend Current | 1,4,6 | | 10 | mA | Device is disabled (see Table 2) |

## DC Characteristics, Continued

| Symbol | Parameter | Notes | Min | Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input Low Voltage | 5 | –0.5 | 0.8 | V | |
| $V_{IH}$ | Input High Voltage | 5 | 2.0 | $V_{CCQ}$ + 0.5 | V | |
| $V_{OL}$ | Output Low Voltage | 2,5 | | 0.4 | V | $V_{CCQ} = V_{CCQ2/3}$ Min<br>$I_{OL}$ = 2 mA |
| | | | | 0.2 | V | $V_{CCQ} = V_{CCQ2/3}$ Min<br>$I_{OL}$ = 100 µA |
| $V_{OH}$ | Output High Voltage | 2,5 | 0.85 × $V_{CCQ}$ | | V | $V_{CCQ} = V_{CCQ}$ Min<br>$I_{OH}$ = –2.5 mA |
| | | | $V_{CCQ}$ – 0.2 | | V | $V_{CCQ} = V_{CCQ}$ Min<br>$I_{OH}$ = –100 µA |
| $V_{PENLK}$ | $V_{PEN}$ Lockout during Program, Erase and Lock-Bit Operations | 5,7,8 | | 0.8 | V | |
| $V_{PENH}$ | $V_{PEN}$ during Block Erase, Program, or Lock-Bit Operations | 7,8 | 2.7 | 3.6 | V | |
| $V_{LKO}$ | $V_{CC}$ Lockout Voltage | 9 | 2.0 | | V | |

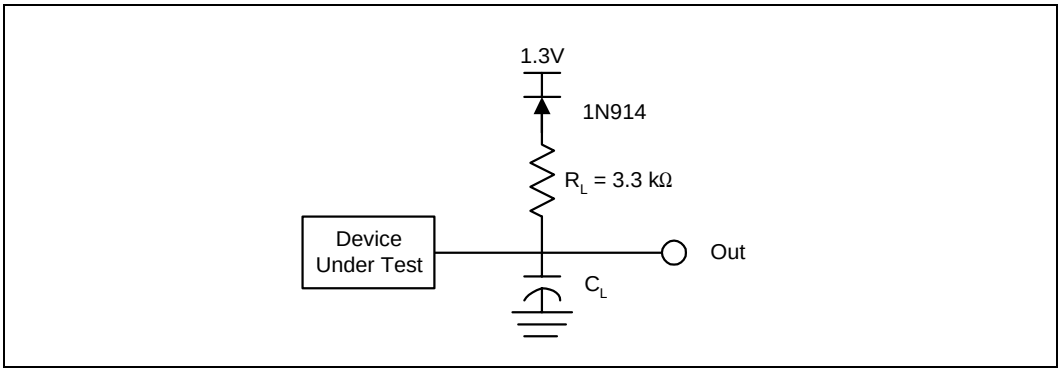
**NOTES:**
1. All currents are in RMS unless otherwise noted. These currents are valid for all product versions (packages and speeds). Contact Intel's Application Support Hotline or your local sales office for information about typical specifications.
2. Includes STS.
3. CMOS inputs are either $V_{CC}$ ± 0.2 V or GND ± 0.2 V. TTL inputs are either $V_{IL}$ or $V_{IH}$.
4. Current values are specified over the temperature range (0 °C to 70 °C) and may increase slightly at –25 °C.
5. Sampled, not 100% tested.
6. $I_{CCWS}$ and $I_{CCES}$ are specified with the device de-selected. If the device is read or written while in erase suspend mode, the device's current draw is $I_{CCR}$ or $I_{CCW}$.
7. Block erases, programming, and lock-bit configurations are inhibited when $V_{PEN} \le V_{PENLK}$, and not guaranteed in the range between $V_{PENLK}$ (max) and $V_{PENH}$ (min), and above $V_{PENH}$ (max).
8. Typically, $V_{PEN}$ is connected to $V_{CC}$ (2.7 V–3.6 V).
9. Block erases, programming, and lock-bit configurations are inhibited when $V_{CC} < V_{LKO}$, and not guaranteed in the range between $V_{LKO}$ (min) and $V_{CC}$ (min), and above $V_{CC}$ (max).

### Figure 15. Transient Input/Output Reference Waveform for $V_{CCQ}$ = 3.0 V–3.6 V or $V_{CCQ}$ = 2.7 V–3.6 V




**NOTE:** AC test inputs are driven at $V_{CCQ}$ for a Logic "1" and 0.0 V for a Logic "0." Input timing begins, and output timing ends, at $V_{CCQ}/2$ V (50% of $V_{CCQ}$). Input rise and fall times (10% to 90%) < 5 ns.

**Figure 16. Transient Equivalent Testing Load Circuit**




**NOTE:** $C_L$ Includes Jig Capacitance

| Test Configuration | $C_L$ (pF) |
|---|---|
| $V_{CCQ} = V_{CC}$ = 3.0 V–3.6 V | 30 |
| $V_{CCQ} = V_{CC}$ = 2.7 V–3.6 V | 30 |

## 6.5 AC Characteristics— Read-Only Operations[1,2]

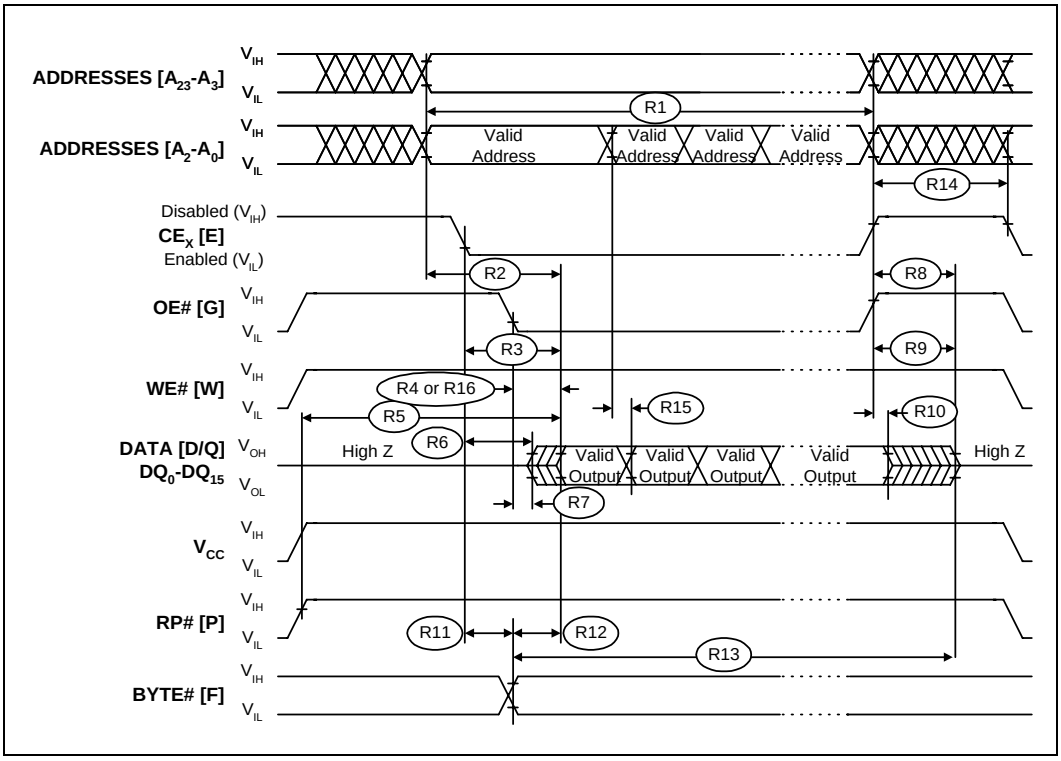
| Versions (All units in ns unless otherwise noted) | | | $V_{CC}$ | | 3.0 V–3.6 V [3] | | 2.7 V–3.6 V [3] | |
|---|---|---|---|---|---|---|---|---|
| | | | $V_{CCQ}$ | | 3.0 V–3.6 V [3] | | 2.7 V–3.6 V [3] | |
| # | Sym | Parameter | | Notes | Min | Max | Min | Max |
| R1 | $t_{AVAV}$ | Read/Write Cycle Time | 32 Mbit | | 100 | | 100 | |
| | | | 64 Mbit | | 120 | | 120 | |
| | | | 128 Mbit | | 150 | | 150 | |
| R2 | $t_{AVQV}$ | Address to Output Delay | 32 Mbit | | | 100 | | 100 |
| | | | 64 Mbit | | | 120 | | 120 |
| | | | 128 Mbit | | | 150 | | 150 |
| R3 | $t_{ELQV}$ | CEx to Output Delay | 32 Mbit | 2 | | 100 | | 100 |
| | | | 64 Mbit | 2 | | 120 | | 120 |
| | | | 128 Mbit | 2 | | 150 | | 150 |
| R4 | $t_{GLQV}$ | OE# to Non-Array Output Delay | | 2, 4 | | 50 | | 50 |
| R5 | $t_{PHQV}$ | RP# High to Output Delay | 32 Mbit | | | 150 | | 150 |
| | | | 64 Mbit | | | 180 | | 180 |
| | | | 128 Mbit | | | 210 | | 210 |
| R6 | $t_{ELQX}$ | CEx to Output in Low Z | | 5 | 0 | | 0 | |
| R7 | $t_{GLQX}$ | OE# to Output in Low Z | | 5 | 0 | | 0 | |
| R8 | $t_{EHQZ}$ | CEx High to Output in High Z | | 5 | | 55 | | 55 |
| R9 | $t_{GHQZ}$ | OE# High to Output in High Z | | 5 | | 15 | | 15 |
| R10 | $t_{OH}$ | Output Hold from Address, CEx, or OE# Change, Whichever Occurs First | | 5 | 0 | | 0 | |
| R11 | $t_{ELFL}/t_{ELFH}$ | CEx Low to BYTE# High or Low | | 5 | | 10 | | 10 |
| R12 | $t_{FLQV}/t_{FHQV}$ | BYTE# to Output Delay | | | | 1000 | | 1000 |
| R13 | $t_{FLQZ}$ | BYTE# to Output in High Z | | 5 | | 1000 | | 1000 |
| R14 | $t_{EHEL}$ | CEx High to CEx Low | | 5 | 0 | | 0 | |
| R15 | $t_{APA}$ | Page Address Access Time | | 5, 6 | | 25 | | 30 |
| R16 | $t_{GLQV}$ | OE# to Array Output Delay | | 4 | | 25 | | 30 |

**NOTES:**

$CE_X$ low is defined as the first edge of $CE_0$, $CE_1$, or $CE_2$ that enables the device. $CE_X$ high is defined at the first edge of $CE_0$, $CE_1$, or $CE_2$ that disables the device (see Table 2).

1. See AC Input/Output Reference Waveforms for the maximum allowable input slew rate.
2. OE# may be delayed up to $t_{ELQV}$-$t_{GLQV}$ after the first edge of $CE_0$, $CE_1$, or $CE_2$ that enables the device (see Table 2) without impact on $t_{ELQV}$.
3. See Figures 14–16, Transient Input/Output Reference Waveform for $V_{CCQ}$ = 3.0 V –3.6 V or $V_{CCQ}$ = 2.7 V – 3.6 V, and Transient Equivalent Testing Load Circuit for testing characteristics.
4. When reading the flash array a faster $t_{GLQV}$ (R16) applies. Non-array reads refer to status register reads, query reads, or device identifier reads.
5. Sampled, not 100% tested.
6. For devices configured to standard word/byte read mode, R15 ($t_{APA}$) will equal R2 ($t_{AVQV}$).

## Figure 17. AC Waveform for Both Page-Mode and Standard Word/Byte Read Operations




**NOTE:** $CE_X$ low is defined as the first edge of $CE_0$, $CE_1$, or $CE_2$ that enables the device. $CE_X$ high is defined at the first edge of $CE_0$, $CE_1$, or $CE_2$ that disables the device (see Table 2).
For standard word/byte read operations, R15 ($t_{APA}$) will equal R2 ($t_{AVQV}$).
When reading the flash array a faster $t_{GLQV}$ (R16) applies. Non-array reads refer to status register reads, query reads, or device identifier reads.

## 6.6 AC Characteristics— Write Operations[1,2]

| Versions | | | | Valid for All Speeds | | Unit |
|---|---|---|---|---|---|---|
| # | Symbol | Parameter | Notes | Min | Max | |
| W1 | $t_{PHWL}$ ($t_{PHEL}$) | RP# High Recovery to WE# ($CE_X$) Going Low | 3 | 1 | | µs |
| W2 | $t_{ELWL}$ ($t_{WLEL}$) | $CE_X$ (WE#) Low to WE# ($CE_X$) Going Low | 4 | 0 | | ns |
| W3 | $t_{WP}$ | Write Pulse Width | 4 | 70 | | ns |
| W4 | $t_{DVWH}$ ($t_{DVEH}$) | Data Setup to WE# ($CE_X$) Going High | 5 | 50 | | ns |
| W5 | $t_{AVWH}$ ($t_{AVEH}$) | Address Setup to WE# ($CE_X$) Going High | 5 | 55 | | ns |
| W6 | $t_{WHEH}$ ($t_{EHWH}$) | $CE_X$ (WE#) Hold from WE# ($CE_X$) High | | 10 | | ns |
| W7 | $t_{WHDX}$ ($t_{EHDX}$) | Data Hold from WE# ($CE_X$) High | | 0 | | ns |
| W8 | $t_{WHAX}$ ($t_{EHAX}$) | Address Hold from WE# ($CE_X$) High | | 0 | | ns |
| W9 | $t_{WPH}$ | Write Pulse Width High | 6 | 30 | | ns |
| W11 | $t_{VPWH}$ ($t_{VPEH}$) | $V_{PEN}$ Setup to WE# ($CE_X$) Going High | 3 | 0 | | ns |
| W12 | $t_{WHGL}$ ($t_{EHGL}$) | Write Recovery before Read | 7 | 35 | | ns |
| W13 | $t_{WHRL}$ ($t_{EHRL}$) | WE# ($CE_X$) High to STS Going Low | 8 | | 90 | ns |
| W15 | $t_{QVVL}$ | $V_{PEN}$ Hold from Valid SRD, STS Going High | 3,8,9 | 0 | | ns |

**NOTES:**

$CE_X$ low is defined as the first edge of $CE_0$, $CE_1$, or $CE_2$ that enables the device. $CE_X$ high is defined at the first edge of $CE_0$, $CE_1$, or $CE_2$ that disables the device (see Table 2).

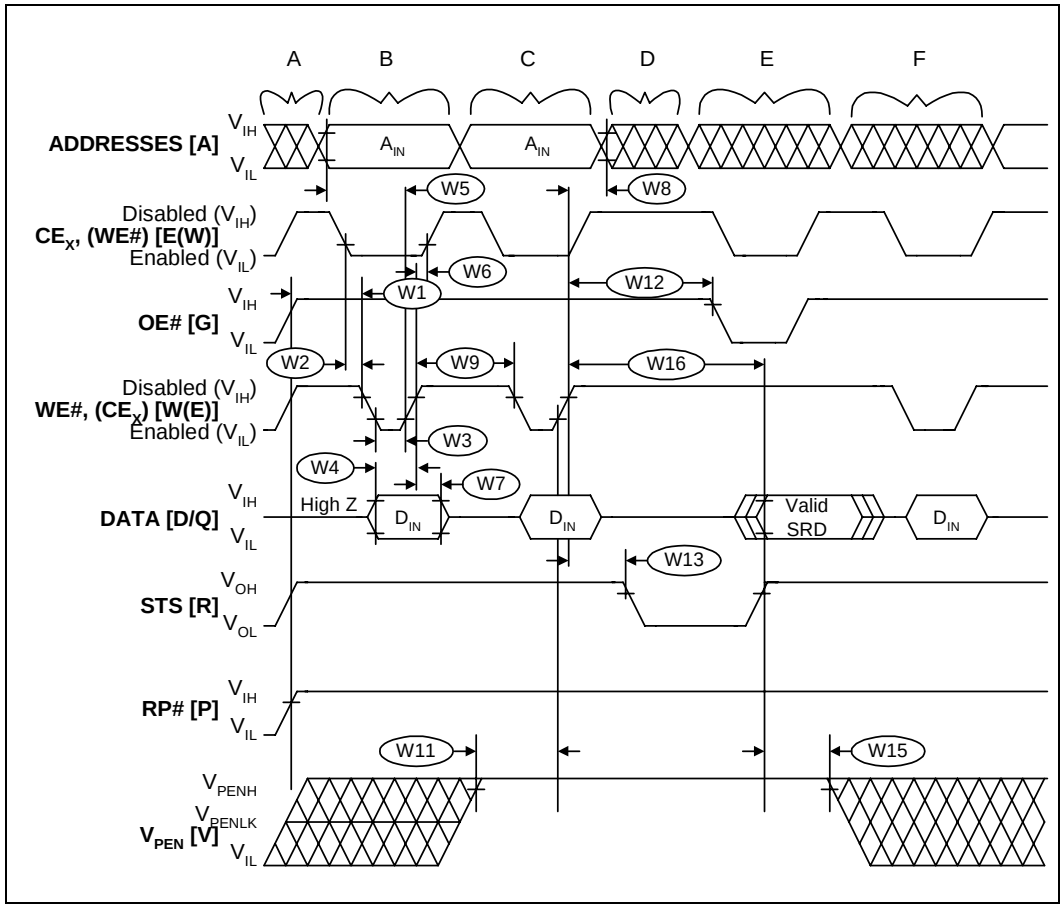
1. Read timing characteristics during block erase, program, and lock-bit configuration operations are the same as during read-only operations. Refer to *AC Characteristics–Read-Only Operations*.
2. A write operation can be initiated and terminated with either $CE_X$ or WE#.
3. Sampled, not 100% tested.
4. Write pulse width ($t_{WP}$) is defined from $CE_X$ or WE# going low (whichever goes low first) to $CE_X$ or WE# going high (whichever goes high first). Hence, $t_{WP} = t_{WLWH} = t_{ELEH} = t_{WLEH} = t_{ELWH}$. If $CE_X$ is driven low 10 ns before WE# going low, WE# pulse width requirement decreases to $t_{WP}$ - 10 ns.
5. Refer to Table 4 for valid $A_{IN}$ and $D_{IN}$ for block erase, program, or lock-bit configuration.
6. Write pulse width high ($t_{WPH}$) is defined from $CE_X$ or WE# going high (whichever goes high first) to $CE_X$ or WE# going low (whichever goes low first). Hence, $t_{WPH} = t_{WHWL} = t_{EHEL} = t_{WHEL} = t_{EHWL}$.
7. For array access, $t_{AVQV}$ is required in addition to $t_{WHGL}$ for any accesses after a write.
8. STS timings are based on STS configured in its RY/BY# default mode.
9. $V_{PEN}$ should be held at $V_{PENH}$ until determination of block erase, program, or lock-bit configuration success (SR.1/3/4/5 = 0).

## 6.7 Block Erase, Program, and Lock-Bit Configuration Performance(1,2,3)

| # | Sym | Parameter | Notes | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| W16 | | Write Buffer Byte Program Time (Time to Program 32 bytes/16 words) | 4,5,6,7 | 218 | 654 | µs |
| W16 | $t_{WHQV3}$ $t_{EHQV3}$ | Byte Program Time (Using Word/Byte Program Command) | 4 | 210 | 630 | µs |
| | | Block Program Time (Using Write to Buffer Command) | 4 | 0.8 | 2.4 | sec |
| W16 | $t_{WHQV4}$ $t_{EHQV4}$ | Block Erase Time | 4 | 1.0 | 5.0 | sec |
| W16 | $t_{WHQV5}$ $t_{EHQV5}$ | Set Lock-Bit Time | 4 | 64 | 75 | µs |
| W16 | $t_{WHQV6}$ $t_{EHQV6}$ | Clear Block Lock-Bits Time | 4 | 0.5 | 0.70 | sec |
| W16 | $t_{WHRH1}$ $t_{EHRH1}$ | Program Suspend Latency Time to Read | | 25 | 30 | µs |
| W16 | $t_{WHRH}$ $t_{EHRH}$ | Erase Suspend Latency Time to Read | | 26 | 35 | µs |

**NOTES:**
1. Typical values measured at $T_A$ = +25 °C and nominal voltages. Assumes corresponding lock-bits are not set. Subject to change based on device characterization.
2. These performance numbers are valid for all speed versions.
3. Sampled but not 100% tested.
4. Excludes system-level overhead.
5. These values are valid when the buffer is full, and the start address is aligned on a 32-byte boundary.
6. Effective per-byte program time ($t_{WHQV1}$, $t_{EHQV1}$) is 6.8 µs/byte (typical)
7. Effective per-word program time ($t_{WHQV2}$, $t_{EHQV2}$) is 13.6 µs/word (typical)
8. Max values are measured at worst case temperature and $V_{CC}$ corner after 100k cycles

## Figure 18. AC Waveform for Write Operations




0606_17

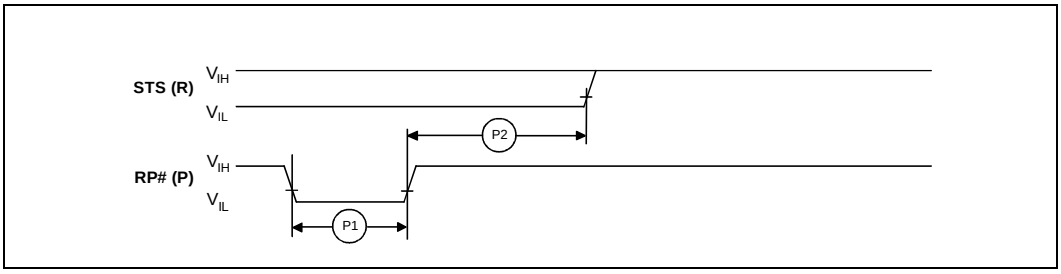
**NOTES:**

$CE_X$ low is defined as the first edge of $CE_0$, $CE_1$, or $CE_2$ that enables the device. $CE_X$ high is defined at the first edge of $CE_0$, $CE_1$, or $CE_2$ that disables the device (see Table 2).

STS is shown in its default mode (RY/BY#).

a. $V_{CC}$ power-up and standby.
b. Write block erase, write buffer, or program setup.
c. Write block erase or write buffer confirm, or valid address and data.
d. Automated erase delay.
e. Read status register or query data.
f. Write Read Array command.

## Figure 19. AC Waveform for Reset Operation



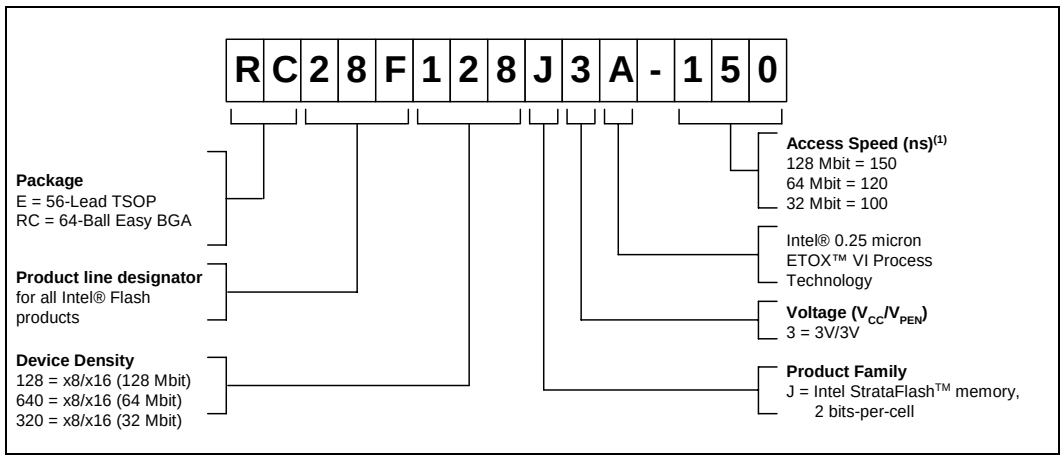

**NOTE:** STS is shown in its default mode (RY/BY#).

## Reset Specifications[1]

| # | Sym | Parameter | Notes | Min | Max | Unit |
|---|---|---|---|---|---|---|
| P1 | $t_{PLPH}$ | RP# Pulse Low Time (If RP# is tied to $V_{CC}$, this specification is not applicable) | 2 | 35 | | µs |
| P2 | $t_{PHRH}$ | RP# High to Reset during Block Erase, Program, or Lock-Bit Configuration | 3 | | 100 | ns |

**NOTES:**
1. These specifications are valid for all product versions (packages and speeds).
2. If RP# is asserted while a block erase, program, or lock-bit configuration operation is not executing then the minimum required RP# Pulse Low Time is 100 ns.
3. A reset time, $t_{PHQV}$, is required from the latter of STS (in RY/BY# mode) or RP# going high until outputs are valid.

# 7.0 Ordering Information




**NOTE:**
1. These speeds are for either the standard asynchronous read access times or for the first access of a page-mode read sequence.

### VALID COMBINATIONS

| 56-Lead TSOP | 64-Ball Easy BGA |
|---|---|
| E28F128J3A-150 | RC28F128J3A-150 |
| E28F640J3A-120 | RC28F640J3A-120 |
| E28F320J3A-100 | RC28F320J3A-100 |

# 8.0 Additional Information

| Order Number | Document/Tool |
|---|---|
| 298130 | *3 Volt Intel® StrataFlash™ Memory 28F128J3A, 28F640J3A, 320J3A Specification Update* |
| 290668 | *Intel® Persistent Storage Manager* datasheet |
| 292237 | *AP-689 Using Intel® Persistent Storage Manager* |
| Note 3 | *AP-707 3 Volt Intel® StrataFlash™ Memory CPU Interface Design Guide* |
| 290606 | *5 Volt Intel® StrataFlash™ MemoryI28F320J5 and 28F640J5* datasheet |
| 290608 | *3 Volt FlashFile™ Memory; 28F160S3 and 28F320S3* datasheet |
| 290609 | *5 Volt FlashFile™ Memory; 28F160S5 and 28F320S5* datasheet |
| 290429 | *5 Volt FlashFile™ Memory; 28F008SA* datasheet |
| 290598 | *3 Volt FlashFile™ Memory; 28F004S3, 28F008S3, 28F016S3* datasheet |
| 290597 | *5 Volt FlashFile™ Memory; 28F004S5, 28F008S5, 28F016S5* datasheet |
| 297859 | *AP-677 Intel® StrataFlash™ Memory Technology* |
| 292222 | *AP-664 Designing Intel® StrataFlash™ Memory into Intel® Architecture* |
| 292221 | *AP-663 Using the Intel® StrataFlash™ Memory Write Buffer* |
| 292218 | *AP-660 Migration Guide to 3 Volt Intel® StrataFlash™ Memory* |
| 292205 | *AP-647 5 Volt Intel® StrataFlash™ Memory Design Guide* |
| 292204 | *AP-646 Common Flash Interface (CFI) and Command Sets* |
| 292202 | *AP-644 Migration Guide to 5 Volt Intel® StrataFlash™ Memory* |
| 298161 | *Intel® Flash Memory Chip Scale Package User's Guide* |
| Note 4 | *Preliminary Mechanical Specification for Easy BGA Package* |

**NOTE:**
1. Please call the Intel Literature Center at (800) 548-4725 to request Intel documentation. International customers should contact their local Intel or distribution sales office.
2. Visit Intel's World Wide Web home page at http://www.intel.com for technical documentation and tools.
3. For the most current information on Intel StrataFlash memory, visit our website at http://developer.intel.com/design/flash/isf.
4. This document is available on the web at http://developer.intel.com/design/flcomp/packdata/298049.htm.