PRELIMINARY

2.4 VOLT ADVANCED+ BOOT BLOCK FLASH MEMORY

28F800C2, 28F160C2 (x16)

- **Flexible SmartVoltage Technology**
  - **2.4 V–3.0 V Read/Program/Erase**
  - **12 V for Fast Production Programming**
- **High Performance**
  - **2.4 V–3.0 V: 100 ns Max Access Time**
  - **2.7 V–3.0 V: 90 ns Max Access Time**
- **Optimized Architecture for Code Plus Data Storage**
  - **Eight 4- Kword Blocks, Top or Bottom Locations**
  - **Up to Sixty-Three 32-Kword Blocks**
  - **Fast Program Suspend Capability**
  - **Fast Erase Suspend Capability**
- **Flexible Block Locking**
  - **Lock/Unlock Any Block**
  - **Full Protection on Power-Up**
  - **WP# Pin for Hardware Block Protection**
  - **V_{PP} = GND Option**
  - **V_{CC} Lockout Voltage**
- **Low Power Consumption**
  - **8 mA Typical Read Power**
  - **10 µA Typical Standby Power with Automatic Power Savings Feature**
- **Extended Temperature Operation**
  - **–40 °C to +85 °C**
- **Improved 12 V Production Programming**
  - **Faster Production Programming**
  - **No Additional System Logic**
- **128-bit Protection Register**
  - **64-bit Unique Device Identifier**
  - **64-bit User Programmable OTP Cells**
- **Extended Cycling Capability**
  - **Minimum 100,000 Block Erase Cycles**
- **Supports Flash Data Integrator Software**
  - **Flash Memory Manager**
  - **System Interrupt Manager**
  - **Supports Parameter Storage, Streaming Data (e.g., voice)**
- **Automated Word/Byte Program and Block Erase**
  - **Command User Interface**
  - **Status Registers**
- **Cross-Compatible Command Support**
  - **Intel Basic Command Set**
  - **Common Flash Interface**
- **x16 I/O for Various Applications**
  - **48-Ball µBGA* Package**
  - **48-Lead TSOP Package**
- **0.25 µ ETOX™ VI Flash Technology**

The 0.25 µm 2.4 Volt Advanced+ Boot Block flash memory, manufactured on Intel's latest 0.25 µ technology, represents a feature-rich solution for low power applications. These flash memory devices incorporate low voltage capability (2.4 V read, program and erase) with high-speed, low-power operation. Flexible block locking allows any block to be independently locked or unlocked. A 128-bit protection register enhances customers' ability to develop secure systems. Add to this the Intel-developed Flash Data Integrator (FDI) software and you have a cost-effective, flexible, monolithic code plus data storage solution. 2.4 Volt Advanced+ Boot Block products will be available in 48-lead TSOP and 48-ball µBGA* packages. All devices have a 16-bit data bus. Additional information on this product family can be obtained by accessing Intel's Flash website: http://www.intel.com/design/flash.

June 1999

Order Number: 290647-002

Information in this document is provided in connection with Intel products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, and Intel disclaims any express or implied warranty, relating to sale and/or use of Intel products including liability or warranties relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright or other intellectual property right. Intel products are not intended for use in medical, life saving, or life sustaining applications.

Intel may make changes to specifications and product descriptions at any time, without notice.

The 28F800C2 and 28F160C2 may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an ordering number and are referenced in this document, or other Intel literature, may be obtained from:

Intel Corporation
P.O. Box 5937
Denver, CO 80217-9808

or call 1-800-548-4725
or visit Intel's website at http:\\www.intel.com

 CG-041493

*Other brands and names are the property of their respective owners.

CONTENTS

REVISION HISTORY

Date of Revision	Version	Description
11/17/98	-001	Original version
06/11/99	-002	Removed all references to x8 configurations Removed 32-Mbit offering Appendix C, *CFI Query Structure*, tables updated

1.0 INTRODUCTION

This document contains the specifications for the 2.4 Volt Advanced+ Boot Block flash memory family. These flash memories add features which can be used to enhance the security of systems: instant block locking and a protection register.

Throughout this document, the term "2.4 V" refers to the full voltage range 2.4 V–3.0 V (except where noted otherwise) and "V_{PP} = 12 V" refers to 12 V ±5%. Sections 1 and 2 provide an overview of the flash memory family including applications, pinouts, pin descriptions and memory organization. Section 3 describes the operation of these products. Finally, Section 4 contains the operating specifications.

1.1 2.4 Volt Advanced+ Boot Block Flash Memory Enhancements

The 2.4 Volt Advanced+ Boot Block flash memory features:

- Zero-latency, flexible block locking
- 128-bit Protection Register
- Simple system implementation for 12 V production programming with 2.4 V in-field programming
- Ultra-low power operation at 2.4 V
- Minimum 100,000 block erase cycles
- Common Flash Interface for software query of device specs and features

Table 1. 2.4 Volt Advanced+ Boot Block Feature Summary

Feature	8 Mbit(1), 16 Mbit	Reference
V_{CC} Operating Voltage	2.4 V – 3.0 V	Table 8
V_{PP} Voltage	Provides complete write protection with optional 12 V Fast Programming	Table 8
V_{CCQ} I/O Voltage	2.4 V– 3.0 V	
Bus Width	16-bit	Table 2
Speed (ns)	8/16 Mbit: 100, 120 @ 2.4 V and 90, 110 @ 2.7 V	Section 4.4
Blocking (top or bottom)	8 x 4-Kword parameter 8-Mb: 15 x 32-Kword main 16-Mb: 31 x 32-Kword main	Section 2.2 Appendix E
Operating Temperature	Extended: –40 °C to +85 °C	Table 8
Program/Erase Cycling	100,000 cycles	Table 8
Packages	48-Lead TSOP 48-Ball µBGA* CSP(1)	Figures 1 and 2
Block Locking	Flexible locking of any block with zero latency	Section 3.3
Protection Register	64-bit unique device number, 64-bit user programmable	Section 3.4

NOTE:

1. 8-Mbit density not available in µBGA* CSP.

1.2 Product Overview

Intel provides secure low voltage memory solutions with the Advanced Boot Block family of products. A new block locking feature allows instant locking/unlocking of any block with zero-latency. A 128-bit protection register allows unique flash device identification.

Discrete supply pins provide single voltage read, program, and erase capability at 2.4 V while also allowing 12 V V_{PP} for faster production programming. Improved 12 V, a new feature designed to reduce external logic, simplifies board designs when combining 12 V production programming with 2.4 V in-field programming.

The 2.4 Volt Advanced+ Boot Block flash memory products are available in x16 packages in the following densities: (see Section 6, *Ordering Information*)

- 8-Mbit (8,388,608 bit) flash memories organized as either 512 Kwords of 16 bits each.
- 16-Mbit (16,777,216 bit) flash memories organized as either 1024 Kwords of 16 bits each.

Eight 4-Kword parameter blocks are located at either the top (denoted by -T suffix) or the bottom (-B suffix) of the address map in order to accommodate different microprocessor protocols for kernel code location. The remaining memory is grouped into 64-Kbyte main blocks. (See Appendix E.)

All blocks can be locked or unlocked instantly to provide complete protection for code or data. (see Section 3.3 for details).

The Command User Interface (CUI) serves as the interface between the microprocessor or microcontroller and the internal operation of the flash memory. The internal Write State Machine (WSM) automatically executes the algorithms and timings necessary for program and erase operations, including verification, thereby unburdening the microprocessor or microcontroller. The status register indicates the status of the WSM by signifying block erase or word program completion and status.

Program and erase automation allows program and erase operations to be executed using an industry-standard two-write command sequence to the CUI. Program operations are performed in word or byte increments. Erase operations erase all locations within a block simultaneously. Both program and erase operations can be suspended by the system software in order to read from any other block. In addition, data can be programmed to another block during an erase suspend.

The 2.4 Volt Advanced+ Boot Block flash memories offer two low power savings features: Automatic Power Savings (APS) and standby mode. The device automatically enters APS mode following the completion of a read cycle. Standby mode is initiated when the system deselects the device by driving CE# inactive. Combined, these two power savings features significantly reduce power consumption.

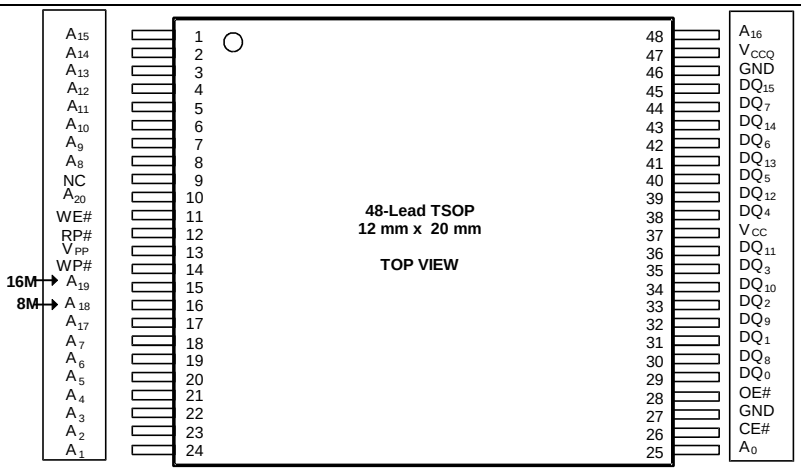
The device can be reset by lowering RP# to GND. This provides CPU-memory reset synchronization and additional protection against bus noise that may occur during system reset and power-up/down sequences (see Section 3.5 and 3.6).

Refer to the *DC Characteristics* Section 4.3 for complete current and voltage specifications. Refer to the *AC Characteristics* Sections 4.4 and 4.5, for read and write performance specifications. Program and erase times and shown in Section 4.6.

2.0 PRODUCT DESCRIPTION

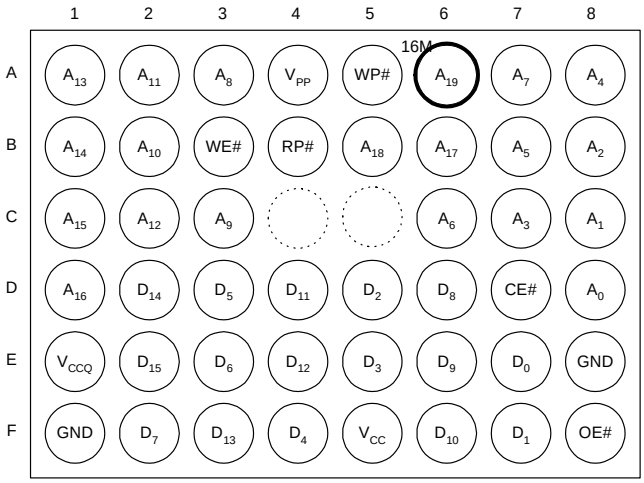
This section provides device pin descriptions and package pinouts for the 2.4 Volt Advanced+ Boot Block flash memory family which is available in 48-lead TSOP (x16), and 48-ball µBGA packages (Figures 1 and 2, respectively).

2.1 Package Pinouts



NOTE:
Lower densities will have NC on the upper address pins. For example, an 8-Mbit device will have NC on Pins 9 and 15.

Figure 1. 48-Lead TSOP Package for x16 Configurations



NOTES:
1. Shaded connections indicate the upgrade address connections. Lower density devices will not have the upper address solder balls. Routing is not recommended in this area. A_{19} is the upgrade address for the 16-Mbit device.
2. 8-Mbit not available on µBGA* CSP.

Figure 2. x16 48-Ball µBGA* Chip Size Package (Top View, Ball Down)

Table 2. 2.4 Volt Advanced+ Boot Block Pin Descriptions

Symbol	Type	Name and Function
A_0–A_{21}	INPUT	**ADDRESS INPUTS:** Memory addresses are internally latched during a program or erase cycle. 8-Mbit: A[0-18], 16-Mbit: A[0-19]
DQ_0–DQ_7	INPUT/OUTPUT	**DATA INPUTS/OUTPUTS:** Inputs array data on the second CE# and WE# cycle during a Program command. Inputs commands to the Command User Interface when CE# and WE# are active. Data is internally latched. Outputs array, configuration and status register data. The data pins float to tri-state when the chip is de-selected or the outputs are disabled.
DQ_8–DQ_{15}	INPUT/OUTPUT	**DATA INPUTS/OUTPUTS:** Inputs array data on the second CE# and WE# cycle during a Program command. Data is internally latched. Outputs array and configuration data. The data pins float to tri-state when the chip is de-selected.
CE#	INPUT	**CHIP ENABLE:** Activates the internal control logic, input buffers, decoders and sense amplifiers. CE# is active low. CE# high de-selects the memory device and reduces power consumption to standby levels.
OE#	INPUT	**OUTPUT ENABLE:** Enables the device's outputs through the data buffers during a read operation. OE# is active low.
WE#	INPUT	**WRITE ENABLE:** Controls writes to the command register and memory array. WE# is active low. Addresses and data are latched on the rising edge of the second WE# pulse.
RP#	INPUT	**RESET/DEEP POWER-DOWN:** Uses two voltage levels (V_{IL}, V_{IH}) to control reset/deep power-down mode. **When RP# is at logic low, the device is in reset/deep power-down mode**, which drives the outputs to High-Z, resets the Write State Machine, and minimizes current levels (I_{CCD}). **When RP# is at logic high, the device is in standard operation**. When RP# transitions from logic-low to logic-high, the device resets all blocks to locked and defaults to the read array mode.
WP#	INPUT	**WRITE PROTECT:** Controls the lock-down function of the flexible Locking feature **When WP# is a logic low, the lock-down mechanism is enabled** and blocks marked lock-down cannot be unlocked through software. **When WP# is logic high, the lock-down mechanism is disabled** and blocks previously locked-down are now locked and can be unlocked and locked through software. After WP# goes low, any blocks previously marked lock-down revert to that state. See Section 3.3 for details on block locking.
V_{CC}	SUPPLY	**DEVICE POWER SUPPLY:** [2.4 V–3.0 V] Supplies power for device operations.

Table 2. 2.4 Volt Advanced+ Boot Block Pin Descriptions (Continued)

Symbol	Type	Name and Function
V_{CCQ}	INPUT	**I/O POWER SUPPLY:** Supplies power for input/output buffers. [2.4 V–3.0 V] This input should be tied directly to V_{CC}.
V_{PP}	INPUT/ SUPPLY	**PROGRAM/ERASE POWER SUPPLY:** [1.65 V–3.0 V or 11.4 V–12.6 V] Operates as a input at logic levels to control complete device protection. Supplies power for accelerated program and erase operations in 12 V ± 5% range. This pin cannot be left floating. **Lower $V_{PP} \leq V_{PPLK}$, to protect all contents** against Program and Erase commands. **Set $V_{PP} = V_{CC}$ for in-system read, program and erase operations**. In this configuration, V_{PP} can drop as low as 1.65 V to allow for resistor or diode drop from the system supply. Note that if V_{PP} is driven by a logic signal, $V_{IH} = 1.65$. That is, V_{PP} must remain above 1.65V to perform in-system flash modifications. **Raise V_{PP} to 12 V ± 5% for faster program and erase** in a production environment. Applying 12 V ± 5% to V_{PP} can only be done for a maximum of 1000 cycles on the main blocks and 2500 cycles on the parameter blocks. V_{PP} may be connected to 12 V for a total of 80 hours maximum. See Section 3.4 for details on V_{PP} voltage configurations.
GND	SUPPLY	**GROUND:** For all internal circuitry. All ground inputs **must** be connected.
NC		**NO CONNECT:** Pin may be driven or left floating.

2.2 Block Organization

The 2.4 Volt Advanced+ Boot Block is an asymmetrically-blocked architecture that enables system integration of code and data within a single flash device. Each block can be erased independently of the others up to 100,000 times. For the address locations of each block, see the memory maps in Appendix E.

2.2.1 PARAMETER BLOCKS

The 2.4 Volt Advanced+ Boot Block flash memory architecture includes parameter blocks to facilitate storage of frequently updated small parameters (i.e., data that would normally be stored in an EEPROM). Each device contains eight parameter blocks of 4-Kwords (4,096 words).

2.2.2 MAIN BLOCKS

After the parameter blocks, the remainder of the array is divided into 32-Kword (32,768 words) main blocks for data or code storage. Each 8-Mbit or 16-Mbit device contains 15 or 31 main blocks, respectively.

3.0 PRINCIPLES OF OPERATION

The 2.4 Volt Advanced+ Boot Block flash memory family utilizes a CUI and automated algorithms to simplify program and erase operations. The CUI allows for 100% CMOS-level control inputs and fixed power supplies during erasure and programming.

The internal WSM completely automates program and erase operations while the CUI signals the start of an operation and the status register reports status. The CUI handles the WE# interface to the data and address latches, as well as system status requests during WSM operation.

3.1 Bus Operation

The 2.4 Volt Advanced+ Boot Block flash memory devices read, program and erase in-system via the local CPU or microcontroller. All bus cycles to or from the flash memory conform to standard microcontroller bus cycles. Four control pins dictate the data flow in and out of the flash component: CE#, OE#, WE# and RP#. These bus operations are summarized in Table 3.

3.1.1 READ

The flash memory has four read modes available: read array, read configuration, read status and read query. These modes are accessible independent of the V_{PP} voltage. The appropriate read mode command must be issued to the CUI to enter the corresponding mode. Upon initial device power-up or after exit from reset, the device automatically defaults to read array mode.

CE# and OE# must be driven active to obtain data at the outputs. CE# is the device selection control; when active it enables the flash memory device. OE# is the data output control and it drives the selected memory data onto the I/O bus. For all read modes, WE# and RP# must be at V_{IH}. Figure 7 illustrates a read cycle.

3.1.2 OUTPUT DISABLE

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

3.1.3 STANDBY

Deselecting the device by bringing CE# to a logic-high level (V_{IH}) places the device in standby mode, which substantially reduces device power consumption without any latency for subsequent read accesses. In standby, outputs are placed in a high-impedance state independent of OE#. If deselected during program or erase operation, the device continues to consume active power until the program or erase operation is complete.

Table 3. Bus Operations(1)

Mode	Note	RP#	CE#	OE#	WE#	DQ_{0-7}	DQ_{8-15}
Read (Array, Status, Configuration, or Query)	2-4	V_{IH}	V_{IL}	V_{IL}	V_{IH}	D_{OUT}	D_{OUT}
Output Disable	2	V_{IH}	V_{IL}	V_{IH}	V_{IH}	High Z	High Z
Standby	2	V_{IH}	V_{IH}	X	X	High Z	High Z
Reset	2,7	V_{IL}	X	X	X	High Z	High Z
Write	2,5-7	V_{IH}	V_{IL}	V_{IH}	V_{IL}	D_{IN}	D_{IN}

NOTES:

1. 8-bit devices use only DQ [0:7], 16-bit devices use DQ [0:15]
2. X must be V_{IL}, V_{IH} for control pins and addresses.
3. See *DC Characteristics* for V_{PPLK}, V_{PP1}, V_{PP2}, V_{PP3}, voltages.
4. Manufacturer and device codes may also be accessed in read configuration mode (A_1–A_{20} = 0). See Table 4.
5. Refer to Table 5 for valid D_{IN} during a write operation.
6. To program or erase the lockable blocks, hold WP# at V_{IH}.
7. RP# must be at GND ± 0.2 V to meet the maximum deep power-down current specified.

3.1.4 RESET

From read mode, RP# at V_{IL} for time t_{PLPH} deselects the memory, places output drivers in a high-impedance state, and turns off all internal circuits. After return from reset, a time t_{PHQV} is required until the initial read access outputs are valid. A delay (t_{PHWL} or t_{PHEL}) is required after return from reset before a write can be initiated. After this wake-up interval, normal operation is restored. The CUI resets to read array mode, the status register is set to 80H, and all blocks are locked. This case is shown in Figure 9A.

If RP# is taken low for time t_{PLPH} during a program or erase operation, the operation will be aborted and the memory contents at the aborted location (for a program) or block (for an erase) are no longer valid, since the data may be partially erased or written. The abort process goes through the following sequence: When RP# goes low, the device shuts down the operation in progress, a process which takes time t_{PLRH} to complete. After this time t_{PLRH}, the part will either reset to read array mode (if RP# has gone high during t_{PLRH}, Figure 9B) or enter reset mode (if RP# is still logic low after t_{PLRH}, Figure 9C). In both cases, after returning from an aborted operation, the relevant time t_{PHQV} or t_{PHWL}/t_{PHEL} must be waited before a read or write operation is initiated, as discussed in the previous paragraph. However, in this case, these delays are referenced to the end of t_{PLRH} rather than when RP# goes high.

As with any automated device, it is important to assert RP# during system reset. When the system comes out of reset, processor expects to read from the flash memory. Automated flash memories provide status information when read during program or block erase operations. If a CPU reset occurs with no flash memory reset, proper CPU initialization may not occur because the flash memory may be providing status information instead of array data. Intel's flash memories allow proper CPU 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.

3.1.5 WRITE

A write takes place when both CE# and WE# are low and OE# is high. Commands are written to the Command User Interface (CUI) using standard microprocessor write timings to control flash operations. The CUI does not occupy an addressable memory location. The address and data buses are latched on the rising edge of the second WE# or CE# pulse, whichever occurs first. Figure 8 illustrates a program and erase operation. The available commands are shown in Table 6, and Appendix A provides detailed information on moving between the different modes of operation using CUI commands.

There are two commands that modify array data: Program (40H) and Erase (20H). Writing either of these commands to the internal Command User Interface (CUI) initiates a sequence of internally-timed functions that culminate in the completion of the requested task (unless that operation is aborted by either RP# being driven to V_{IL} for t_{PLRH} or an appropriate suspend command).

3.2 Modes of Operation

The flash memory has four read modes and two write modes. The read modes are read array, read configuration, read status, and read query. The write modes are program and erase. Three additional modes (erase suspend to program, erase suspend to read and program suspend to read) are available only during suspended operations. These modes are reached using the commands summarized in Tables 5 and 6. A comprehensive chart showing the state transitions is in Appendix A.

3.2.1 READ ARRAY

When RP# transitions from V_{IL} (reset) to V_{IH}, the device defaults to read array mode and will respond to the read control inputs (CE#, address inputs, and OE#) without any additional CUI commands.

When the device is in read array mode, four control signals control data output:

- WE# must be logic high (V_{IH})
- CE# must be logic low (V_{IL})
- OE# must be logic low (V_{IL})
- RP# must be logic high (V_{IH})

In addition, the address of the desired location must be applied to the address pins. If the device is not in read array mode, as would be the case after a program or erase operation, the Read Array command (FFH) must be written to the CUI before array reads can take place.

3.2.2 READ CONFIGURATION

The read configuration mode outputs the manufacturer/device identifier. The device is switched to this mode by writing the read configuration command (90H). Once in this mode, read cycles from addresses shown in Table 4 retrieve the specified information. To return to read array mode, write the Read Array command (FFH).

The Read Configuration mode outputs three types of information: the manufacturer/device identifier, the block locking status, and the protection register. The device is switched to this mode by writing the Read Configuration command (90H). Once in this mode, read cycles from addresses shown in Table 4 retrieve the specified information. To return to read array mode, write the Read Array command (FFH).

Table 4. Read Configuration Table

Item	Address	Data
Manufacturer Code (x16)	00000	0089
Device ID (See Appendix F)	00001	ID
Block Lock Configuration[2]	XX002[(1)]	LOCK
• Block Is Unlocked		$DQ_0 = 0$
• Block Is Locked		$DQ_0 = 1$
• Block Is Locked-Down		$DQ_1 = 1$
Protection Register Lock[3]	80	PR-LK
Protection Register (x16)	81-88	PR

NOTES:

1. "XX" specifies the block address of lock configuration being read.
2. See Section 3.3.4 for valid lock status outputs.
3. See Section 3.4 for protection register information.
4. Other locations within the configuration address space are reserved by Intel for future use.

3.2.3 READ STATUS REGISTER

The status register indicates the status of device operations, and the success/failure of that operation. The Read Status Register (70H) command causes subsequent reads to output data from the status register until another command is issued. To return to reading from the array, issue a Read Array (FFH) command.

The status register bits are output on DQ_0–DQ_7. The upper byte, DQ_8–DQ_{15}, outputs 00H during a Read Status Register command.

The contents of the status register are latched on the falling edge of OE# or CE#, whichever occurs last. This prevents possible bus errors which might occur if status register contents change while being read. CE# or OE# must be toggled with each subsequent status read, or the status register will not indicate completion of a program or erase operation.

When the WSM is active, SR.7 will indicate the status of the WSM; the remaining bits in the status register indicate whether the WSM was successful in performing the desired operation (see Table 7).

3.2.3.1 Clearing the Status Register

The WSM sets status bits 1 through 7 to "1," and clears bits 2, 6 and 7 to "0," but cannot clear status bits 1 or 3 through 5 to "0." Because bits 1, 3, 4 and 5 indicate various error conditions, these bits can only be cleared through the use of the Clear Status Register (50H) command. By allowing the system software to control the resetting of these bits, several operations may be performed (such as cumulatively programming several addresses or erasing multiple blocks in sequence) before reading the status register to determine if an error occurred during that series. Clear the status register before beginning another command or sequence. Note that the Read Array command must be issued before data can be read from the memory array. Resetting the device also clears the status register.

3.2.4 READ QUERY

The read query mode outputs Common Flash Interface (CFI) data when the device is read. This can be accessed by writing the Read Query Command (98H). The CFI data structure contains information such as block size, density, command set and electrical specifications. Once in this mode, read cycles from addresses shown in Appendix C retrieve the specified information. To return to read array mode, write the Read Array command (FFH).

3.2.5 PROGRAM MODE

Programming is executed using a two-write sequence. The Program Setup command (40H) is written to the CUI followed by a second write which specifies the address and data to be programmed. The WSM will execute a sequence of internally timed events to program desired bits of the addressed location, then verify the bits are sufficiently programmed. Programming the memory results in specific bits within an address location being changed to a "0." If the user attempts to program "1"s, the memory cell contents do not change and no error occurs.

The status register indicates programming status: while the program sequence executes, status bit 7 is "0." The status register can be polled by toggling either CE# or OE#. While programming, the only valid commands are Read Status Register, Program Suspend, and Program Resume.

When programming is complete, the program status bits should be checked. If the programming operation was unsuccessful, bit SR.4 of the status register is set to indicate a program failure. If SR.3 is set then V_{PP} was not within acceptable limits, and the WSM did not execute the program command. If SR.1 is set, a program operation was attempted on a locked block and the operation was aborted.

The status register should be cleared before attempting the next operation. Any CUI instruction can follow after programming is completed; however, to prevent inadvertent status register reads, be sure to reset the CUI to read array mode.

3.2.5.1 Suspending and Resuming Program

The Program Suspend command halts an in-progress program operation so that data can be read from other locations of memory. Once the programming process starts, writing the Program Suspend command to the CUI requests that the WSM suspend the program sequence (at predetermined points in the program algorithm). The device continues to output status register data after the Program Suspend command is written. Polling status register bits SR.7 and SR.2 will determine when the program operation has been suspended (both will be set to "1"). t_{WHRH1}/t_{EHRH1} specify the program suspend latency.

A Read Array command can now be written to the CUI to read data from blocks other than that which is suspended. The only other valid commands, while program is suspended, are Read Status Register, Read Configuration, Read Query, and Program Resume. After the Program Resume command is written to the flash memory, the WSM will continue with the programming process and status register bits SR.2 and SR.7 will automatically be cleared. The device automatically outputs status register data when read (see Figure 11 in Appendix B, *Program Suspend/Resume Flowchart*) after the Program Resume command is written. V_{PP} must remain at the same V_{PP} level used for program while in program suspend mode. RP# must also remain at V_{IH}.

3.2.6 ERASE MODE

To erase a block, write the Erase Set-up and Erase Confirm commands to the CUI, along with an address identifying the block to be erased. This address is latched internally when the Erase Confirm command is issued. Block erasure results in all bits within the block being set to "1." Only one block can be erased at a time. The WSM will execute a sequence of internally timed events to program all bits within the block to "0," erase all bits within the block to "1," then verify that all bits within the block are sufficiently erased. While the erase executes, status bit 7 is a "0."

When the status register indicates that erasure is complete, check the erase status bit to verify that the erase operation was successful. If the Erase operation was unsuccessful, SR.5 of the status register will be set to a "1," indicating an erase failure. If V_{PP} was not within acceptable limits after the Erase Confirm command was issued, the WSM will not execute the erase sequence; instead, SR.5 of the status register is set to indicate an erase error, and SR.3 is set to a "1" to identify that V_{PP} supply voltage was not within acceptable limits.

After an erase operation, clear the status register (50H) before attempting the next operation. Any CUI instruction can follow after erasure is completed; however, to prevent inadvertent status register reads, it is advisable to place the flash in read array mode after the erase is complete.

3.2.6.1 Suspending and Resuming Erase

Since an erase operation requires on the order of seconds to complete, an Erase Suspend command is provided to allow erase-sequence interruption in order to read data from or program data to another block in memory. Once the erase sequence is started, writing the Erase Suspend command to the CUI suspends the erase sequence at a predetermined point in the erase algorithm. The status register will indicate if/when the erase operation has been suspended. Erase suspend latency is specified by t_{WHRH2}/t_{EHRH2}.

A Read Array/Program command can now be written to the CUI to read/program data from/to blocks other than that which is suspended. This nested Program command can subsequently be suspended to read yet another location. The only valid commands while erase is suspended are Read Status Register, Read Configuration, Read Query, Program Setup, Program Resume, Erase Resume, Lock Block, Unlock Block and Lock-Down Block. During erase suspend mode, the chip can be placed in a pseudo-standby mode by taking CE# to V_{IH}. This reduces active current consumption.

Erase Resume continues the erase sequence when CE# = V_{IL}. As with the end of a standard erase operation, the status register must be read and cleared before the next instruction is issued.

Table 5. Command Bus Definitions

Command	Notes	First Bus Cycle			Second Bus Cycle		
		Oper	Addr	Data	Oper	Addr	Data
Read Array	4	Write	X	FFH			
Read Configuration	2, 4	Write	X	90H	Read	IA	ID
Read Query	2, 4	Write	X	98H	Read	QA	QD
Read Status Register	4	Write	X	70H	Read	X	SRD
Clear Status Register	4	Write	X	50H			
Program	3,4	Write	X	40H/10H	Write	PA	PD
Block Erase/Confirm	4	Write	X	20H	Write	BA	D0H
Program/Erase Suspend	4	Write	X	B0H			
Program/Erase Resume	4	Write	X	D0H			
Lock Block	4	Write	X	60H	Write	BA	01H
Unlock Block	4	Write	X	60H	Write	BA	D0H
Lock-Down Block	4	Write	X	60H	Write	BA	2FH
Protection Program	4	Write	X	C0H	Write	PA	PD

X = Don't Care PA = Prog Addr BA = Block Addr IA = Identifier Addr. QA = Query Addr.
SRD = Status Reg. Data PD = Prog Data ID = Identifier Data QD = Query Data

NOTES:

1. Bus operations are defined in Table 3.
2. Following the Read Configuration or Read Query commands, read operations output device configuration or CFI query information, respectively. See Section 3.2.2 and 3.2.4.
3. Either 40H or 10H command is valid, but the Intel standard is 40H.
4. When writing commands, the upper data bus [DQ_8–DQ_{15}] should be either V_{IL} or V_{IH}, to minimize current draw.

Table 6. Command Codes and Descriptions

Code	Device Mode	Description
FF	Read Array	Places device in read array mode, such that array data will be output on the data pins.
40	Program Set-Up	This is a two-cycle command. The first cycle prepares the CUI for a program operation. The second cycle latches addresses and data information and initiates the WSM to execute the Program algorithm. The flash outputs status register data when CE# or OE# is toggled. A Read Array command is required after programming to read array data. See Section 3.2.5.
20	Erase Set-Up	Prepares the CUI for the Erase Confirm command. If the next command is not an Erase Confirm command, then the CUI will (a) set both SR.4 and SR.5 of the status register to a "1," (b) place the device into the read status register mode, and (c) wait for another command. See Section 3.2.6.
D0	Erase Confirm	If the previous command was an Erase Set-Up command, then the CUI will close the address and data latches, and begin erasing the block indicated on the address pins. During program/erase, the device will respond only to the Read Status Register, Program Suspend and Erase Suspend commands and will output status register data when CE# or OE# is toggled.
	Program/Erase Resume	If a program or erase operation was previously suspended, this command will resume that operation.
	Unlock Block	If the previous command was Configuration Set-Up, the CUI will latch the address and unlock the block indicated on the address pins. If the block had been previously set to Lock-Down, this operation will have no effect. (Sect. 3.3)
B0	Program Suspend Erase Suspend	Issuing this command will begin to suspend the currently executing program/erase operation. The status register will indicate when the operation has been successfully suspended by setting either the program suspend (SR.2) or erase suspend (SR.6) and the WSM status bit (SR.7) to a "1" (ready). The WSM will continue to idle in the SUSPEND state, regardless of the state of all input control pins except RP#, which will immediately shut down the WSM and the remainder of the chip if RP# is driven to V_{IL}. See Sections 3.2.5.1 and 3.2.6.1.
70	Read Status Register	This command places the device into read status register mode. Reading the device will output the contents of the status register, regardless of the address presented to the device. The device automatically enters this mode after a program or erase operation has been initiated. See Section 3.2.3.
50	Clear Status Register	The WSM can set the block lock status (SR.1) , V_{PP} Status (SR.3), program status (SR.4), and erase status (SR.5) bits in the status register to "1," but it cannot clear them to "0." Issuing this command clears those bits to "0."
90	Read Configuration	Puts the device into the read configuration mode, so that reading the device will output the manufacturer/device codes or block lock status. Section 3.2.2.
60	Configuration Set-Up	Prepares the CUI for changes to the device configuration, such as block locking changes. If the next command is not Block Unlock, Block Lock, or Block Lock-Down, then the CUI will set both the program and erase status register bits to indicate a command sequence error. See Section 3.3.
01	Lock-Block	If the previous command was Configuration Set-Up, the CUI will latch the address and lock the block indicated on the address pins. (Section 3.3)

Table 6. Command Codes and Descriptions (Continued)

Code	Device Mode	Description
2F	Lock-Down	If the previous command was a Configuration Set-Up command, the CUI will latch the address and lock-down the block indicated on the address pins. (Section 3.3)
98	Read Query	Puts the device into the read query mode, so that reading the device will output Common Flash Interface information. See Section 3.2.4 and Appendix C.
C0	Protection Program Setup	This is a two-cycle command. The first cycle prepares the CUI for an program operation to the protection register. The second cycle latches addresses and data information and initiates the WSM to execute the Protection Program algorithm to the protection register. The flash outputs status register data when CE# or OE# is toggled. A Read Array command is required after programming to read array data. See Section 3.4.
10	Alt. Prog Set-Up	Operates the same as Program Set-up command. (See 40H/Program Set-Up)
00	Invalid/ Reserved	Unassigned commands that should not be used. Intel reserves the right to redefine these codes for future functions.

NOTE:

See Appendix A for mode transition information.

Table 7. Status Register Bit Definition

WSMS	ESS	ES	PS	VPPS	PSS	BLS	R
7	6	5	4	3	2	1	0

	NOTES:
SR.7 WRITE STATE MACHINE STATUS 1 = Ready (WSMS) 0 = Busy	Check Write State Machine bit first to determine Word Program or Block Erase completion, before checking Program or Erase Status bits.
SR.6 = ERASE-SUSPEND STATUS (ESS) 1 = Erase Suspended 0 = Erase In Progress/Completed	When Erase Suspend is issued, WSM halts execution and sets both WSMS and ESS bits to "1." ESS bit remains set to "1" until an Erase Resume command is issued.
SR.5 = ERASE STATUS (ES) 1 = Error In Block Erase 0 = Successful Block Erase	When this bit is set to "1," WSM has applied the max. number of erase pulses to the block and is still unable to verify successful block erasure.
SR.4 = PROGRAM STATUS (PS) 1 = Error in Programming 0 = Successful Programming	When this bit is set to "1," WSM has attempted but failed to program a word/byte.
SR.3 = V_{PP} STATUS (VPPS) 1 = V_{PP} Low Detect, Operation Abort 0 = V_{PP} OK	The V_{PP} status bit does not provide continuous indication of V_{PP} level. The WSM interrogates V_{PP} level only after the Program or Erase command sequences have been entered, and informs the system if V_{PP} has not been switched on. The V_{PP} is also checked before the operation is verified by the WSM. The V_{PP} status bit is not guaranteed to report accurate feedback between V_{PPLK} and V_{PP1}Min.
SR.2 = PROGRAM SUSPEND STATUS (PSS) 1 = Program Suspended 0 = Program in Progress/Completed	When Program Suspend is issued, WSM halts execution and sets both WSMS and PSS bits to "1." PSS bit remains set to "1" until a Program Resume command is issued.
SR.1 = BLOCK LOCK STATUS 1 = Prog/Erase attempted on a locked block; Operation aborted. 0 = No operation to locked blocks	If a program or erase operation is attempted to one of the locked blocks, this bit is set by the WSM. The operation specified is aborted and the device is returned to read status mode.
SR.0 = RESERVED FOR FUTURE ENHANCEMENTS (R)	This bit is reserved for future use and should be masked out when polling the status register.

NOTES:

1. A Command Sequence Error is indicated when both SR.4 , SR.5 and SR.7 are set.

3.3 Flexible Block Locking

The Intel® 2.4 Volt Advanced+ Boot Block products offer an instant, individual block locking scheme that allows any block to be locked or unlocked with no latency, enabling instant code and data protection.

This locking scheme offers two levels of protection. The first level allows software-only control of block locking (useful for data blocks that change frequently), while the second level requires hardware interaction before locking can be changed (useful for code blocks that change infrequently).

The following sections will discuss the operation of the locking system. The term “state [XYZ]” will be used to specify locking states; e.g., “state [001],” where X = value of WP#, Y = bit DQ_1 of the Block Lock status register, and Z = bit DQ_0 of the Block Lock status register. Table 9 defines all of these possible locking states.

3.3.1 LOCKING OPERATION

The following concisely summarizes the locking functionality.

- All blocks power-up locked, then can be unlocked or locked with the Unlock and Lock commands.
- The Lock-Down command locks a block and prevents it from being unlocked when WP# = 0.
  - — When WP# = 1, Lock-Down is overridden and commands can unlock/lock locked-down blocks.
  - — When WP# returns to 0, locked-down blocks return to Lock-Down.
  - — Lock-Down is cleared only when the device is reset or powered-down.

The locking status of each block can set to Locked, Unlocked, and Lock-Down, each of which will be described in the following sections. A comprehensive state table for the locking functions is shown in Table 9, and a flowchart for locking operations is shown in Figure 14.

3.3.2 LOCKED STATE

The default status of all blocks upon power-up or reset is locked (states [001] or [101]). Locked blocks are fully protected from alteration. Any program or erase operations attempted on a locked block will return an error on bit SR.1 of the status register. The status of a locked block can be changed to Unlocked or Lock-Down using the appropriate software commands. An Unlocked block can be locked by writing the Lock command sequence, 60H followed by 01H.

3.3.3 UNLOCKED STATE

Unlocked blocks (states [000], [100], [110]) can be programmed or erased. All unlocked blocks return to the Locked state when the device is reset or powered down. The status of an unlocked block can be changed to Locked or Locked-Down using the appropriate software commands. A Locked block can be unlocked by writing the Unlock command sequence, 60H followed by D0H.

3.3.4 LOCK-DOWN STATE

Blocks that are Locked-Down (state [011]) are protected from program and erase operations (just like Locked blocks), but their protection status cannot be changed using software commands alone. A Locked or Unlocked block can be Locked-down by writing the Lock-Down command sequence, 60H followed by 2FH. Locked-Down blocks revert to the Locked state when the device is reset or powered down.

The Lock-Down function is dependent on the WP# input pin. When WP# = 0, blocks in Lock-Down [011] are protected from program, erase, and lock status changes. When WP# = 1, the Lock-Down function is disabled ([111]) and locked-down blocks can be individually unlocked by software command to the [110] state, where they can be erased and programmed. These blocks can then be relocked [111] and unlocked [110] as desired while WP# remains high. When WP# goes low, blocks that were previously locked-down return to the Lock-Down state [011] regardless of any changes made while WP# was high. Device reset or power-down resets all blocks, including those in Lock-Down, to Locked state.

3.3.5 READING A BLOCK'S LOCK STATUS

The lock status of every block can be read in the configuration read mode of the device. To enter this mode, write 90H to the device. Subsequent reads at Block Address + 00002 will output the lock status of that block. The lock status is represented by the lowest two output pins, DQ_0 and DQ_1. DQ_0 indicates the Block Lock/Unlock status and is set by the Lock command and cleared by the Unlock command. It is also automatically set when entering Lock-Down. DQ_1 indicates Lock-Down status and is set by the Lock-Down command. It cannot be cleared by software, only by device reset or power-down.

Table 8. Block Lock Status

Item	Address	Data
Block Lock Configuration	XX002	LOCK
• Block Is Unlocked		$DQ_0 = 0$
• Block Is Locked		$DQ_0 = 1$
• Block Is Locked-Down		$DQ_1 = 1$

3.3.6 LOCKING OPERATIONS DURING ERASE SUSPEND

Changes to block lock status can be performed during an erase suspend by using the standard locking command sequences to unlock, lock, or lock-down a block. This is useful in the case when another block needs to be updated while an erase operation is in progress.

To change block locking during an erase operation, first write the erase suspend command (B0H), then check the status register until it indicates that the erase operation has been suspended. Next write the desired lock command sequence to a block and the lock status will be changed. After completing any desired lock, read, or program operations, resume the erase operation with the Erase Resume command (D0H).

If a block is locked or locked-down during a suspended erase of the same block, the locking status bits will be changed immediately, but when the erase is resumed, the erase operation will complete.

Locking operations cannot be performed during a program suspend. Refer to Appendix A for detailed information on which commands are valid during erase suspend.

3.3.7 STATUS REGISTER ERROR CHECKING

Using nested locking or program command sequences during erase suspend can introduce ambiguity into status register results.

Since locking changes are performed using a two cycle command sequence, e.g., 60H followed by 01H to lock a block, following the Configuration Setup command (60H) with an invalid command will produce a lock command error (SR.4 and SR.5 will be set to 1) in the status register. If a lock command error occurs during an erase suspend, SR.4 and SR.5 will be set to 1, and will remain at 1 after the erase is resumed. When erase is complete, any possible error during the erase cannot be detected via the status register because of the previous locking command error.

A similar situation happens if an error occurs during a program operation error nested within an erase suspend.

Table 9. Block Locking State Transitions

Current State				Erase/Prog	Lock Command Input Result [Next State]		
WP#	DQ_1	DQ_0	Name	Allowed?	Lock	Unlock	Lock-Down
0	0	0	"Unlocked"	Yes	Goes To [001]	No Change	Goes To [011]
0	0	1	"Locked" (Default)	No	No Change	Goes To [000]	Goes To [011]
0	1	1	"Locked-Down"	No	No Change	No Change	No Change
1	0	0	"Unlocked"	Yes	Goes To [101]	No Change	Goes To [111]
1	0	1	"Locked"	No	No Change	Goes To [100]	Goes To [111]
1	1	0	Lock-Down Disabled	Yes	Goes To [111]	No Change	Goes To [111]
1	1	1	Lock-Down Disabled	No	No Change	Goes To [110]	No Change

NOTES:

1. In this table, the notation [XYZ] denotes the locking state of a block, where X= WP#, Y = DQ_1, and Z = DQ_0. The current locking state of a block is defined by the state of WP# and the two bits of the block lock status (DQ_0, DQ_1). DQ_0 indicates if a block is locked (1) or unlocked (0). DQ_1 indicates if a block has been locked-down (1) or not (0).
2. At power-up or device reset, all blocks default to Locked state [001] (if WP#= 0). Holding WP# = 0 is the recommended default.
3. The "Erase/Program Allowed?" column shows whether erase and program operations are enabled (Yes) or disabled (No) in that block's current locking state.
4. The "Lock Command Input Result [Next State]" column shows the result of writing the three locking commands (Lock, Unlock, Lock-Down) in the current locking state. For example, "Goes To [001]" would mean that writing the command to a block in the current locking state would change it to [001].

3.4 128-Bit Protection Register

The Advanced+ Boot Block architecture includes a 128-bit protection register than 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. Additional application information can be found in Intel application note *AP-657 Designing with the Advanced+ Boot Block Flash Memory Architecture*.

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 designs to program as desired. Once the customer segment is programmed, it can be locked to prevent reprogramming.

3.4.1 READING THE PROTECTION REGISTER

The protection register is read in the configuration read mode. The device is switched to this mode by writing the Read Configuration command (90H). Once in this mode, read cycles from addresses shown in Appendix G retrieve the specified information. To return to read array mode, write the Read Array command (FFH).

3.4.2 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 Appendix G. See Figure 15 for the *Protection Register Programming Flowchart*.

Attempts to address Protection Program commands outside the defined protection register address space should not be attempted. This space is reserved for future use. Attempting to program to a previously 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).

3.4.3 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. This bit 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.

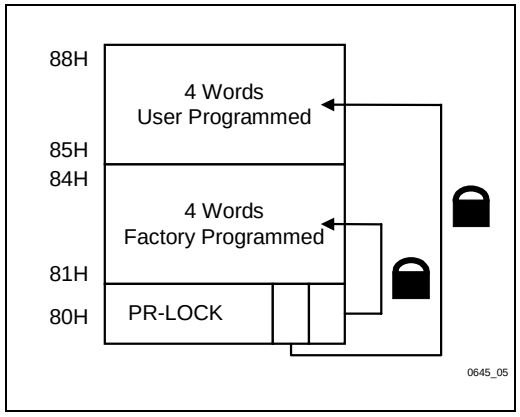


Figure 3. Protection Register Memory Map

3.5 V_{PP} Program and Erase Voltages

Intel’s 2.4 Volt Advanced+ Boot Block products provide in-system programming and erase in the 1.65 V–3.0 V range. For fast production programming, it also includes a low-cost, backward-compatible 12 V programming feature.

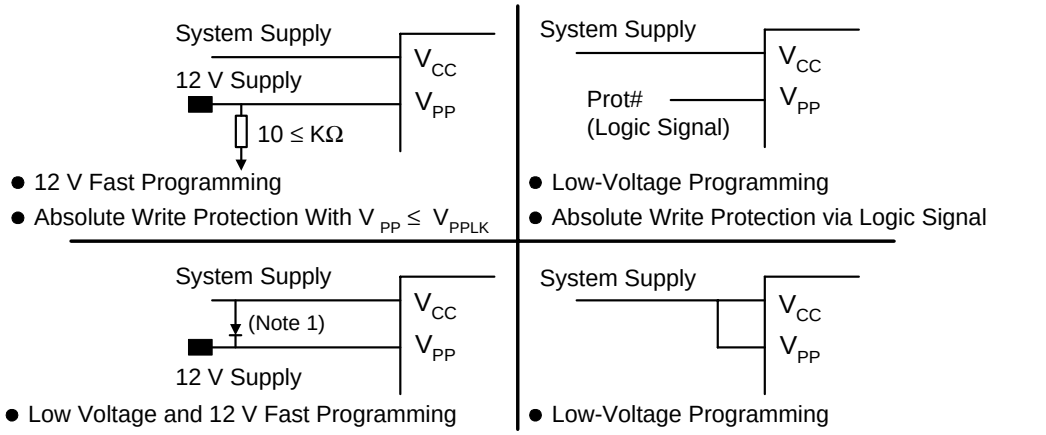
3.5.1 IMPROVED 12 V OPERATION FOR PRODUCTION PROGRAMMING

When V_{PP} is between 1.65 V and 3.0 V, all program and erase current is drawn through the V_{CC} pin. Note that if V_{PP} is driven by a logic signal, V_{IH} min = 1.65 V. That is, V_{PP} must remain above 1.65 V to perform in-system flash modifications. When V_{PP} is connected to a 12 V power supply, the device draws program and erase current directly from the V_{PP} pin. This eliminates the need for an external switching transistor to control the voltage V_{PP}. Figure 4 shows examples of how the flash power supplies can be configured for various usage models.

The 12 V V_{PP} mode enhances programming performance during the short period of time typically found in manufacturing processes; however, it is not intended for extended use. 12 V may be applied to V_{PP} during program and erase operations for a maximum of 1000 cycles on the main blocks and 2500 cycles on the parameter blocks. V_{PP} may be connected to 12 V for a total of 80 hours maximum. Stressing the device beyond these limits may cause permanent damage.

3.5.2 $V_{PP} \leq V_{PPLK}$ FOR COMPLETE PROTECTION

In addition to the flexible block locking, the V_{PP} programming voltage can be held low for absolute hardware write protection of all blocks in the flash device. When V_{PP} is below V_{PPLK}, any program or erase operation will result in a error, prompting the corresponding status register bit (SR.3) to be set.



NOTE:

1. A resistor can be used if the V_{CC} supply can sink adequate current based on resistor value. See *AP-657 Designing with the Advanced+ Boot Block Flash Memory Architecture* for details.

Figure 4. Example Power Supply Configurations

3.6 Power Consumption

Intel's flash devices have a tiered approach to power savings that can significantly reduce overall system power consumption. The Automatic Power Savings (APS) feature reduces power consumption when the device is selected but idle. If the CE# is deasserted, the flash enters its standby mode, where current consumption is even lower. The combination of these features can minimize memory power consumption, and therefore, overall system power consumption.

3.6.1 ACTIVE POWER (Program/Erase/Read)

With CE# at a logic-low level and RP# at a logic-high level, the device is in the active mode. Refer to the DC Characteristic tables for I_{CC} current values. Active power is the largest contributor to overall system power consumption. Minimizing the active current could have a profound effect on system power consumption, especially for battery-operated devices.

3.6.2 AUTOMATIC POWER SAVINGS (APS)

Automatic Power Savings provides low-power operation during read mode. After data is read from the memory array and the address lines are quiescent, APS circuitry places the device in a mode where typical current is comparable to I_{CCS}. The flash stays in this static state with outputs valid until a new location is read.

3.6.3 STANDBY POWER

With CE# at a logic-high level (V_{IH}) and device in read mode, the flash memory is in standby mode, which disables much of the device's circuitry and substantially reduces power consumption. Outputs are placed in a high-impedance state independent of the status of the OE# signal. If CE# transitions to a logic-high level during erase or program operations, the device will continue to perform the operation and consume corresponding active power until the operation is completed.

System engineers should analyze the breakdown of standby time versus active time and quantify the respective power consumption in each mode for their specific application. This will provide a more accurate measure of application-specific power and energy requirements.

3.6.4 DEEP POWER-DOWN MODE

The deep power-down mode is activated when RP# = V_{IL} (GND ± 0.2 V). During read modes, RP# going low de-selects the memory and places the outputs in a high impedance state. Recovery from deep power-down requires a minimum time of t_{PHQV} for read operations and t_{PHWL}/t_{PHEL} for write operations.

During program or erase modes, RP# transitioning low will abort the in-progress operation. The memory contents of the address being programmed or the block being erased are no longer valid as the data integrity has been compromised by the abort. During deep power-down, all internal circuits are switched to a low power savings mode (RP# transitioning to V_{IL} or turning off power to the device clears the status register).

3.7 Power-Up/Down Operation

The device is protected against accidental block erasure or programming during power transitions. Power supply sequencing is not required, since the device is indifferent as to which power supply, V_{PP} or V_{CC}, powers-up first.

3.7.1 RP# CONNECTED TO SYSTEM RESET

The use of RP# during system reset is important with automated program/erase devices since the system expects to read from the flash memory when it comes out of reset. If a CPU reset occurs without a flash memory reset, proper CPU initialization will not occur because the flash memory may be providing status information instead of array data. Intel recommends connecting RP# to the system CPU RESET# signal to allow proper CPU/flash initialization following system reset.

System designers must guard against spurious writes when V_{CC} voltages are above V_{LKO}. Since both WE# and CE# must be low for a command write, driving either signal to V_{IH} will inhibit writes to the device. The CUI architecture provides additional protection since alteration of memory contents can only occur after successful completion of the two-step command sequences. The device is also disabled until RP# is brought to V_{IH}, regardless of the state of its control inputs. By holding the device in reset (RP# connected to system PowerGood) during power-up/down, invalid bus conditions during power-up can be masked, providing yet another level of memory protection.

3.7.2 V_{CC}, V_{PP} AND RP# TRANSITIONS

The CUI latches commands as issued by system software and is not altered by V_{PP} or CE# transitions or WSM actions. Its default state upon power-up, after exit from reset mode or after V_{CC} transitions above V_{LKO} (Lockout voltage), is read array mode.

After any program or block erase operation is complete (even after V_{PP} transitions down to V_{PPLK}), the CUI must be reset to read array mode via the Read Array command if access to the flash memory array is desired.

3.8 Power Supply Decoupling

Flash memory's power switching characteristics require careful device decoupling. System designers should consider three supply current issues:

1. Standby current levels (I_{CCS})
2. Read current levels (I_{CCR})
3. Transient peaks produced by falling and rising edges of CE#.

Transient current magnitudes depend on the device outputs' capacitive and inductive loading. Two-line control and proper decoupling capacitor selection will suppress these transient voltage peaks. Each flash device should have a 0.1 µF ceramic capacitor connected between each V_{CC} and GND, and between its V_{PP} and GND. These high-frequency, inherently low-inductance capacitors should be placed as close as possible to the package leads.

4.0 ELECTRICAL SPECIFICATIONS

NOTICE: This datasheet contains preliminary information on new products in production. Do not finalize a design with this information. Revised information will be published when the product is available. Verify with your local Intel Sales office that you have the latest datasheet before finalizing a design.

4.1 Absolute Maximum Ratings*

Extended Operating Temperature

- During Read ..... –40 °C to +85 °C
- During Block Erase and Program ..... –40 °C to +85 °C
- Temperature Under Bias..... –40 °C to +85 °C

Storage Temperature ..... –65 °C to +125 °C

Voltage on Any Pin (except V_{CC} and V_{PP}) with Respect to GND .....–0.5 V to +3.7 V[1]

V_{PP} Voltage (for Block Erase and Program) with Respect to GND .....–0.5 V to +13.5 V[1,2,4]

V_{CC} and V_{CCQ} Supply Voltage with Respect to GND .....–0.2 V to +3.0 V[1]

Output Short Circuit Current..... 100 mA[3]

** 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 effect device reliability.*

NOTES:

1. Minimum DC voltage is –0.5 V on input/output pins. During transitions, this level may undershoot to –2.0 V for periods < 20 ns. Maximum DC voltage on input/output pins is V_{CC} + 0.5 V which, during transitions, may overshoot to V_{CC} + 2.0 V for periods < 20 ns.
2. Maximum DC voltage on V_{PP} may overshoot to +14.0 V for periods < 20 ns.
3. Output shorted for no more than one second. No more than one output shorted at a time.
4. V_{PP} voltage is normally 1.65 V–3.0 V. Connection to supply of 11.4 V–12.6 V can only be done for 1000 cycles on the main blocks and 2500 cycles on the parameter blocks during program/erase. V_{PP} may be connected to 12 V for a total of 80 hours maximum. See Section 3.5 for details.

4.2 Operating Conditions

Table 10. Temperature and Voltage Operating Conditions

Symbol	Parameter	Notes	Min	Max	Units
T_A	Operating Temperature		–40	+85	°C
V_{CC1}	V_{CC} Supply Voltage	1	2.4	3.0	Volts
V_{CC2}		1	2.7	3.0	
V_{CCQ1}	I/O Supply Voltage	1	2.4	3.0	Volts
V_{PP1}	Supply Voltage	1	1.65	3.0	Volts
V_{PP2}		1, 2	11.4	12.6	Volts
Cycling	Block Erase Cycling	2	100,000		Cycles

NOTES:

1. V_{CC} and V_{CCQ} must share the same supply when they are in the V_{CC1} range.
2. Applying V_{PP} = 11.4 V–12.6 V during a program/erase can only be done for a maximum of 1000 cycles on the main blocks and 2500 cycles on the parameter blocks. V_{PP} may be connected to 12 V for a total of 80 hours maximum. See Section 3.5 for details.

4.3 Capacitance

$T_A = 25$ °C, $f = 1$ MHz

Sym	Parameter	Notes	Typ	Max	Units	Conditions
C_{IN}	Input Capacitance	1	6	8	pF	$V_{IN} = 0$ V
C_{OUT}	Output Capacitance	1	10	12	pF	$V_{OUT} = 0$ V

NOTE:

1. Sampled, not 100% tested.

4.4 DC Characteristics

Sym	Parameter	V_{CC}	2.4 V–3.0 V		Unit	Test Conditions
		V_{CCQ}	2.4 V–3.0 V			
		Note	Typ	Max		
I_{LI}	Input Load Current	1,7		± 1	µA	$V_{CC} = V_{CC}$Max $V_{CCQ} = V_{CCQ}$Max $V_{IN} = V_{CCQ}$ or GND
I_{LO}	Output Leakage Current	1,7	0.2	± 10	µA	$V_{CC} = V_{CC}$Max $V_{CCQ} = V_{CCQ}$Max $V_{IN} = V_{CCQ}$ or GND
I_{CCS}	V_{CC} Standby Current	1	10	25	µA	$V_{CC} = V_{CC}$Max CE# = RP# = V_{CCQ} WP# = V_{CCQ} or GND
I_{CCD}	V_{CC} Deep Power-Down Current	1,7	7	20	µA	$V_{CC} = V_{CC}$Max $V_{CCQ} = V_{CCQ}$Max $V_{IN} = V_{CCQ}$ or GND RP# = GND ± 0.2 V
I_{CCR}	V_{CC} Read Current	1,5,7	8	12	mA	$V_{CC} = V_{CC}$Max $V_{CCQ} = V_{CCQ}$Max OE# = V_{IH}, CE# = V_{IL} f = 5 MHz, I_{OUT} = 0 mA Inputs = V_{IL} or V_{IH}
I_{CCW}	V_{CC} Program Current	1,4	18	55	mA	$V_{PP} = V_{PP1}$ Program in Progress
			8	15	mA	$V_{PP} = V_{PP2}$ (12 V) Program in Progress
I_{CCE}	V_{CC} Erase Current	1,4	16	45	mA	$V_{PP} = V_{PP1}$ Erase in Progress
			8	15	mA	$V_{PP} = V_{PP2}$ (12 V) Erase in Progress

4.4 DC Characteristics (Continued)

Sym	Parameter	V_{CC}	2.4 V–3.0 V		Unit	Test Conditions
		V_{CCQ}	2.4 V–3.0 V			
		Note	Typ	Max		
I_{CCES}	V_{CC} Erase Suspend Current	1,2,4	10	25	µA	CE# = V_{IH}, Erase Suspend in Progress
I_{CCWS}	V_{CC} Program Suspend Current	1,2,4	10	25	µA	CE# = V_{IH}, Program Suspend in Progress
I_{PPD}	V_{PP} Deep Power-Down Current	1	0.2	5	µA	RP# = GND ± 0.2 V $V_{PP} \leq V_{CC}$
I_{PPS}	V_{PP} Standby Current	1	0.2	5	µA	$V_{PP} \leq V_{CC}$
I_{PPR}	V_{PP} Read Current	1	2	±15	µA	$V_{PP} \leq V_{CC}$
		1,4	50	200	µA	$V_{PP} > V_{CC}$
I_{PPW}	V_{PP} Program Current	1,4	0.05	0.1	mA	$V_{PP} = V_{PP1}$ Program in Progress
			8	22	mA	$V_{PP} = V_{PP2}$ (12 V) Program in Progress
I_{PPE}	V_{PP} Erase Current	1,4	0.05	0.1	mA	$V_{PP} = V_{PP1}$ Program in Progress
			8	22	mA	$V_{PP} = V_{PP2}$ (12 V) Program in Progress
I_{PPES}	V_{PP} Erase Suspend Current	1,4	0.2	5	µA	$V_{PP} = V_{PP1}$ Erase Suspend in Progress
			50	200	µA	$V_{PP} = V_{PP2}$ (12 V) Erase Suspend in Progress
I_{PPWS}	V_{PP} Program Suspend Current	1,4	0.2	5	µA	$V_{PP} = V_{PP1}$ Program Suspend in Progress
			50	200	µA	$V_{PP} = V_{PP2}$ (12 V) Program Suspend in Progress

4.4 DC Characteristics (Continued)

Sym	Parameter	V_{CC} / V_{CCQ} Note	2.4 V–3.0 V Min	2.4 V–3.0 V Max	Unit	Test Conditions
V_{IL}	Input Low Voltage		-0.4	V_{CC} *0.22 V	V	
V_{IH}	Input High Voltage		2.0	V_{CCQ} +0.3 V	V	
V_{OL}	Output Low Voltage	7	-0.10	0.10	V	$V_{CC} = V_{CC}Min$ $V_{CCQ} = V_{CCQ}Min$ I_{OL} = 100 µA
V_{OH}	Output High Voltage	7	V_{CCQ} - 0.1 V		V	$V_{CC} = V_{CC}Min$ $V_{CCQ} = V_{CCQ}Min$ I_{OH} = –100 µA
V_{PPLK}	V_{PP} Lock-Out Voltage	3		1.0	V	Complete Write Protection
V_{PP1}	V_{PP} during Program / Erase Operations	3	1.65	3.0	V	
V_{PP2}		3,6	11.4	12.6		
V_{LKO}	V_{CC} Prog/Erase Lock Voltage		1.5		V	
V_{LKO2}	V_{CCQ} Prog/Erase Lock Voltage		1.2		V	

NOTES:

1. All currents are in RMS unless otherwise noted. Typical values at nominal V_{CC}, T_A = +25 °C.
2. I_{CCES} and I_{CCWS} are specified with device de-selected. If device is read while in erase suspend, current draw is sum of I_{CCES} and I_{CCR}. If the device is read while in program suspend, current draw is the sum of I_{CCWS} and I_{CCR}.
3. Erase and Program are inhibited when $V_{PP} < V_{PPLK}$ and not guaranteed outside the valid V_{PP} ranges of V_{PP1} and V_{PP2}.
4. Sampled, not 100% tested.
5. Automatic Power Savings (APS) reduces I_{CCR} to approximately standby levels in static operation (CMOS inputs).
6. Applying V_{PP} = 11.4 V–12.6 V during program/erase can only be done for a maximum of 1000 cycles on the main blocks and 2500 cycles on the parameter blocks. V_{PP} may be connected to 12 V for a total of 80 hours maximum. See Section 3.4 for details.
7. The test conditions $V_{CC}Max$, $V_{CCQ}Max$, $V_{CC}Min$, and $V_{CCQ}Min$ refer to the maximum or minimum V_{CC} or V_{CCQ} voltage listed at the top of each column.

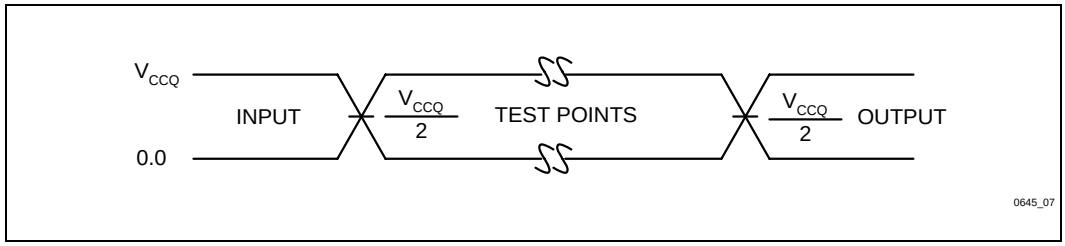


Figure 5. Input/Output Reference Waveform

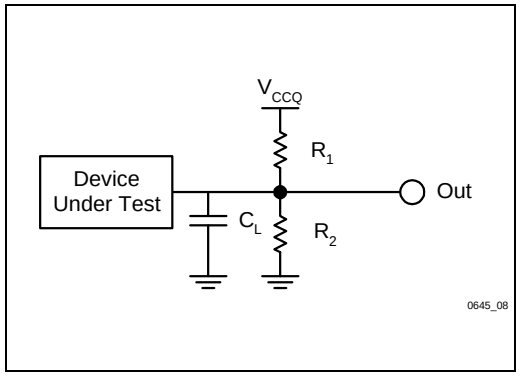


Figure 6. Test Configuration

Test Configuration Component Values Table

Test Configuration	C_L (pF)	R_1 (Ω)	R_2 (Ω)
2.4 V–3.0 V Standard Test	50	22K	22K

NOTE:
C_L includes jig capacitance.

4.5 AC Characteristics—Read Operations[1,4]—Extended Temperature

			Density	8/16 Mbit								
			Product	–100				–120				
			V_{CC}	2.7 V–3.0 V		2.4 V–3.0 V		2.7 V–3.0 V		2.4 V–3.0 V		
#	Sym	Parameter	Note	Min	Max	Min	Max	Min	Max	Min	Max	Unit
R1	t_{AVAV}	Read Cycle Time		90		100		110		120		ns
R2	t_{AVQV}	Address to Output Delay			90		100		110		120	ns
R3	t_{ELQV}	CE# to Output Delay	2		90		100		110		120	ns
R4	t_{GLQV}	OE# to Output Delay	2		30		30		30		30	ns
R5	t_{PHQV}	RP# to Output Delay			150		150		150		150	ns
R6	t_{ELQX}	CE# to Output in Low Z	3	0		0		0		0		ns
R7	t_{GLQX}	OE# to Output in Low Z	3	0		0		0		0		ns
R8	t_{EHQZ}	CE# to Output in High Z	3		25		25		25		25	ns
R9	t_{GHQZ}	OE# to Output in High Z	3		20		20		20		20	ns
R10	t_{OH}	Output Hold from Address, CE#, or OE# Change, Whichever Occurs First	3	0		0		0		0		ns

NOTES:

1. See Figure 7: *AC Waveform: Read Operations*.
2. OE# may be delayed up to t_{ELQV}–t_{GLQV} after the falling edge of CE# without impact on t_{ELQV}.
3. Sampled, but not 100% tested.
4. See Figure 5: *Input/Output Reference Waveform* for timing measurements and maximum allowable input slew rate.

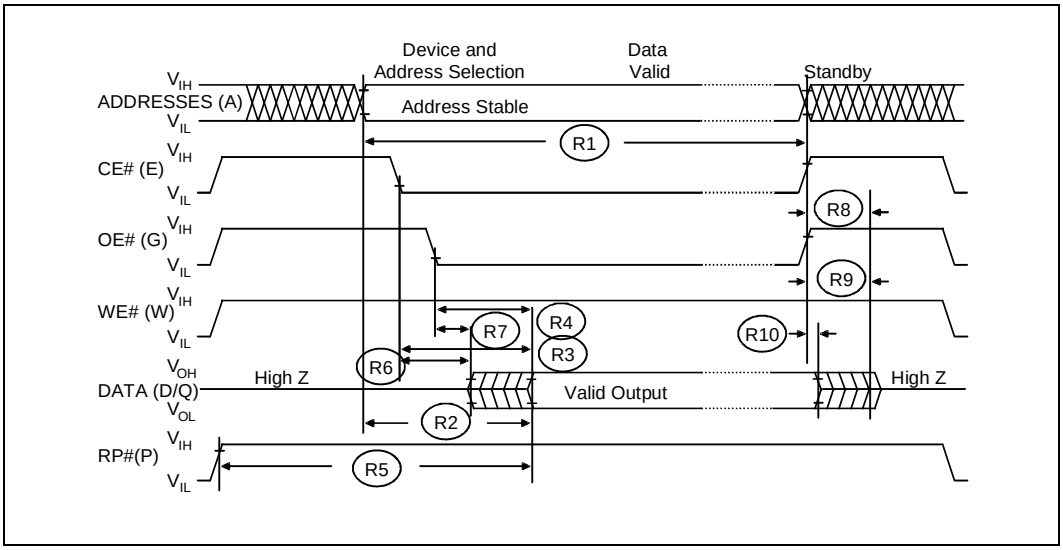


Figure 7. AC Waveform: Read Operations

4.6 AC Characteristics—Write Operations[1,5,6]—Extended Temperature

				Density	8/16 Mbit				
				Product	–100		–120		
				2.7 V – 3.0 V	90		110		
				2.4 V – 3.0 V		100		120	
#	Sym	Parameter	Note		Min	Min	Min	Min	Unit
W1	t_{PHWL}/ t_{PHEL}	RP# High Recovery to WE# (CE#) Going Low			150	150	150	150	ns
W2	t_{ELWL}/ t_{WLEL}	CE# (WE#) Setup to WE# (CE#) Going Low			0	0	0	0	ns
W3	t_{WLWH}/ t_{ELEH}	WE# (CE#) Pulse Width	4		60	70	70	70	ns
W4	t_{DVWH}/ t_{DVEH}	Data Setup to WE# (CE#) Going High	2		50	60	60	60	ns
W5	t_{AVWH}/ t_{AVEH}	Address Setup to WE# (CE#) Going High	2		60	70	70	70	ns
W6	t_{EHWH}/ t_{WHEH}	CE# (WE#) Hold Time from WE# (CE#) High			0	0	0	0	ns
W7	t_{WHDX}/ t_{EHDX}	Data Hold Time from WE# (CE#) High	2		0	0	0	0	ns
W8	t_{WHAX}/ t_{EHAX}	Address Hold Time from WE# (CE#) High	2		0	0	0	0	ns
W9	t_{WHWL}/ t_{EHEL}	WE# (CE#) Pulse Width High	4		30	30	30	30	ns
W10	t_{VPWH}/ t_{VPEH}	V_{PP} Setup to WE# (CE#) Going High	3		200	200	200	200	ns
W11	t_{QVVL}	V_{PP} Hold from Valid SRD	3		0	0	0	0	ns
W12	t_{BHWH} / t_{BHEH}	WP# Setup to WE# (CE#) Going High	3		0	0	0	0	ns
W13	t_{QVBL}	WP# Hold from Valid SRD	3		0	0	0	0	ns

NOTES:

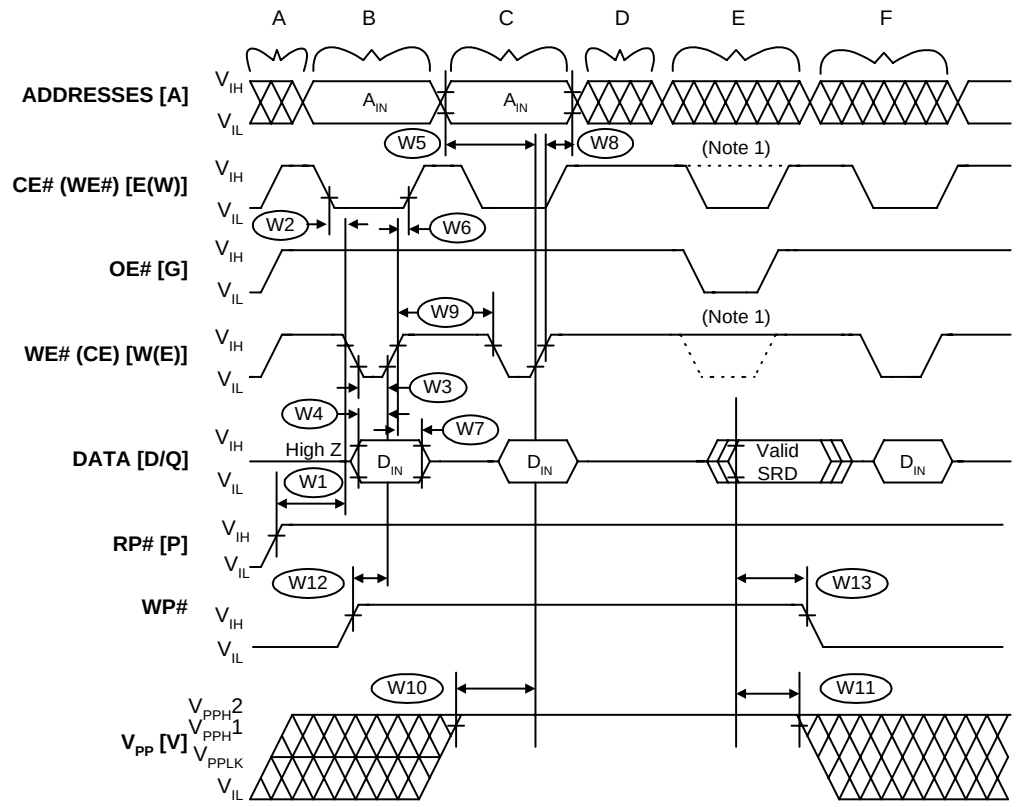
1. Write timing characteristics during erase suspend are the same as during write-only operations.
2. Refer to Table 5 for valid A_{IN} or D_{IN}.
3. Sampled, but not 100% tested.
4. Write pulse width (t_{WP}) is defined from CE# or WE# going low (whichever goes low last) to CE# or WE# going high (whichever goes high first). Hence, $t_{WP} = t_{WLWH} = t_{ELEH} = t_{WLEH} = t_{ELWH}$. Similarly, Write pulse width high (t_{WPH}) is defined from CE# or WE# going high (whichever goes high first) to CE# or WE# going low (whichever goes low first). Hence, $t_{WPH} = t_{WHWL} = t_{EHEL} = t_{WHEL} = t_{EHWL}$.
5. See Figure 5: *Input/Output Reference Waveform* for timing measurements and maximum allowable input slew rate.
6. See Figure 8: *AC Waveform: Program and Erase Operations.*

4.7 Erase and Program Timings(1)

Symbol	Parameter	V_{PP} Note	1.65 V–3.0 V Typ(1)	1.65 V–3.0 V Max	11.4 V–12.6 V Typ(1)	11.4 V–12.6 V Max	Unit
t_{BWPB}	4-KW Parameter Block Word Program Time	2, 3	0.10	0.30	0.03	0.12	s
t_{BWMB}	32-KW Main Block Word Program Time	2, 3	0.8	2.4	0.24	1	s
t_{WHQV1} / t_{EHQV1}	Word Program Time	2, 3	22	200	8	185	µs
t_{WHQV2} / t_{EHQV2}	4-KW Parameter Block Erase Time	2, 3	0.5	4	0.4	4	s
t_{WHQV3} / t_{EHQV3}	32-KW Main Block Erase Time	2, 3	1	5	0.6	5	s
t_{WHRH1} / t_{EHRH1}	Program Suspend Latency	3	5	10	5	10	µs
t_{WHRH2} / t_{EHRH2}	Erase Suspend Latency	3	5	20	5	20	µs

NOTES:

1. Typical values measured at T_A = +25 °C and nominal voltages.
2. Excludes external system-level overhead.
3. Sampled, but not 100% tested.



NOTES:

1. CE# must be toggled low when reading Status Register Data. WE# must be inactive (high) when reading Status Register Data.

A. V_{CC} Power-Up and Standby.
B. Write Program or Erase Setup Command.
C. Write Valid Address and Data (for Program) or Erase Confirm Command.
D. Automated Program or Erase Delay.
E. Read Status Register Data (SRD): reflects completed program/erase operation.
F. Write Read Array Command.

Figure 8. AC Waveform: Program and Erase Operations

4.8 Reset Operations

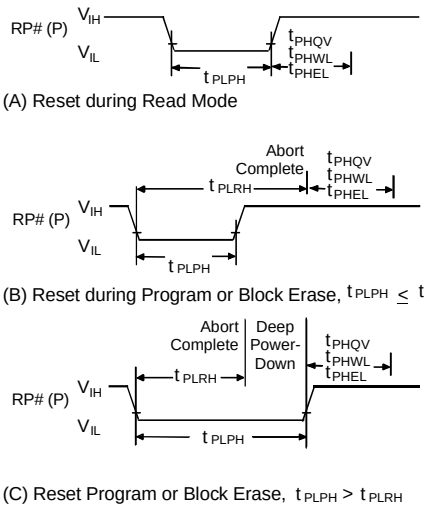


Figure 9. AC Waveform: Reset Operation

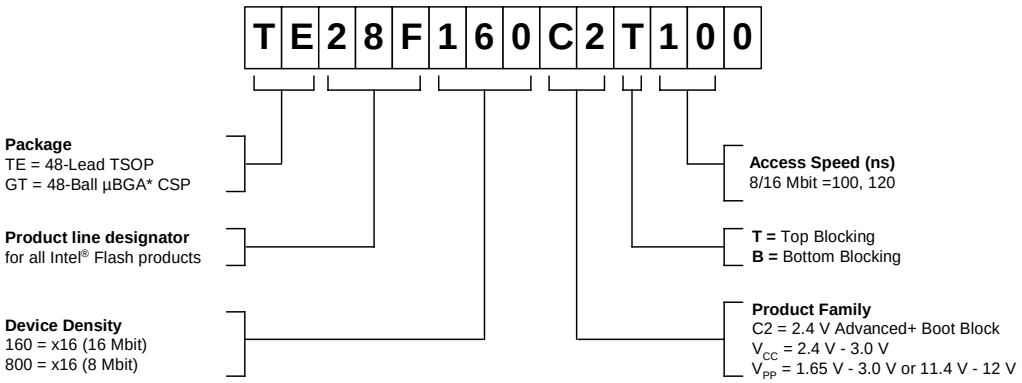
Table 11. Reset Specifications(1)

Symbol	Parameter	Notes	V_{CC} 2.4 V–3.0 V		Unit
			Min	Max	
t_{PLPH}	RP# Low to Reset during Read (If RP# is tied to V_{CC}, this specification is not applicable)	2,4	100		ns
t_{PLRH1}	RP# Low to Reset during Block Erase	3,4		22	µs
t_{PLRH2}	RP# Low to Reset during Program	3,4		12	µs

NOTES:

1. See Section 3.1.4 for a full description of these conditions.
2. If t_{PLPH} is < 100 ns the device may still reset but this is not guaranteed.
3. If RP# is asserted while a block erase or word program operation is not executing, the reset will complete within 100 ns.
4. Sampled, but not 100% tested.

5.0 ORDERING INFORMATION



VALID COMBINATIONS (All Extended Temperature)

Extended	**16M**	TE28F160C2TA100	GT28F160C2TA100
		TE28F160C2BA100	GT28F160C2BA100
		TE28F160C2TA120	GT28F160C2TA120
		TE28F160C2BA120	GT28F160C2BA120
Extended	**8M**	TE28F800C2TA100	
		TE28F800C2BA100	
		TE28F800C2TA120	
		TE28F800C2BA120	

NOTE:

1. The second line of the 48-ball µBGA package top side mark specifies assembly codes. For samples only, the first character signifies either “E” for engineering samples or “S” for silicon daisy chain samples. All other assembly codes without an “E” or “S” as the first character are production units.

6.0 ADDITIONAL INFORMATION(1,2)

Order Number	Document/Tool
298006	*2.4 Volt Advanced+ Boot Block Flash Memory Specification Update*
210830	*Flash Memory Databook*
297645	*3 Volt Advanced+ Boot Block Flash Memory; 28F800C3, 28F160C3, 28F320C3* datasheet
292216	*AP-658 Designing for Upgrade to the Advanced+ Boot Block Flash Memory*
292215	*AP-657 Designing with the Advanced+ Boot Block Flash Memory Architecture*
Contact your Intel Representative	*Flash Data Integrator (FDI) Software Developer's Kit*
297874	*FDI Interactive: Play with Intel's Flash Data Integrator on Your PC*

NOTES:

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 or http://developer.intel.com for technical documentation and tools.

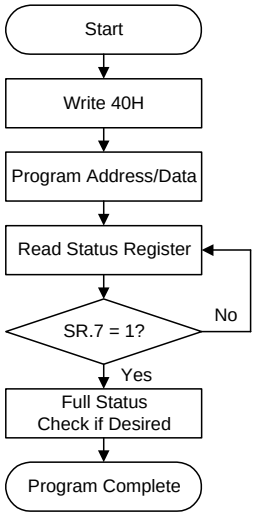
APPENDIX A
WSM CURRENT/NEXT STATES

Current State	SR.7	Data When Read	Command Input (and Next State)							
			Read Array (FFH)	Program Setup (10/40H)	Erase Setup (20H)	Erase Confirm (D0H)	Prog/Ers Suspend (B0H)	Prog/Ers Resume (D0)	Read Status (70H)	Clear Status (50H)
Read Array	"1"	Array	Read Array	Program Setup	Erase Setup	Read Array			Read Status	Read Array
Read Status	"1"	Status	Read Array	Program Setup	Erase Setup	Read Array			Read Status	Read Array
Read Config.	"1"	Config	Read Array	Program Setup	Erase Setup	Read Array			Read Status	Read Array
Read Query	"1"	CFI	Read Array	Program Setup	Erase Setup	Read Array			Read Status	Read Array
Lock Setup	"1"	Status	Lock Command Error			Lock (Done)	Lock Cmd. Error	Lock (Done)	Lock Cmd. Error	
Lock Cmd. Error	"1"	Status	Read Array	Program Setup	Erase Setup	Read Array			Read Status	Read Array
Lock Oper. (Done)	"1"	Status	Read Array	Program Setup	Erase Setup	Read Array			Read Status	Read Array
Prot. Prog. Setup	"1"	Status	Protection Register Program							
Prot. Prog. (Not Done)	"0"	Status	Protection Register Program (Not Done)							
Prot. Prog. (Done)	"1"	Status	Read Array	Program Setup	Erase Setup	Read Array			Read Status	Read Array
Prog. Setup	"1"	Status	Program							
Program (Not Done)	"0"	Status	Program (Not Done)				Prog. Sus. Status	Program (Not Done)		
Prog. Susp. Status	"1"	Status	Prog. Sus. Read Array	Program Suspend Read Array		Program (Not Done)	Prog. Sus. Rd. Array	Program (Not Done)	Prog. Sus. Status	Prog. Sus. Rd. Array
Prog. Susp. Read Array	"1"	Array	Prog. Sus. Read Array	Program Suspend Read Array		Program (Not Done)	Prog. Sus. Rd. Array	Program (Not Done)	Prog. Sus. Status	Prog. Sus. Rd. Array
Prog. Susp. Read Config	"1"	Config	Prog. Sus. Read Array	Program Suspend Read Array		Program (Not Done)	Prog. Sus. Rd. Array	Program (Not Done)	Prog. Sus. Status	Prog. Sus. Rd. Array
Prog. Susp. Read Query	"1"	CFI	Prog. Sus. Read Array	Program Suspend Read Array		Program (Not Done)	Prog. Sus. Rd. Array	Program (Not Done)	Prog. Sus. Status	Prog. Sus. Rd. Array
Program (Done)	"1"	Status	Read Array	Program Setup	Erase Setup	Read Array			Read Status	Read Array
Erase Setup	"1"	Status	Erase Command Error			Erase (Not Done)	Erase Cmd. Error	Erase (Not Done)	Erase Command Error	
Erase Cmd. Error	"1"	Status	Read Array	Program Setup	Erase Setup	Read Array			Read Status	Read Array
Erase (Not Done)	"0"	Status	Erase (Not Done)				Erase Sus. Status	Erase (Not Done)		
Ers. Susp. Status	"1"	Status	Erase Sus. Read Array	Program Setup	Ers. Sus. Rd. Array	Erase	Ers. Sus. Rd. Array	Erase	Erase Sus. Status	Ers. Sus. Rd. Array
Erase Susp. Array	"1"	Array	Erase Sus. Read Array	Program Setup	Ers. Sus. Rd. Array	Erase	Ers. Sus. Rd. Array	Erase	Erase Sus. Status	Ers. Sus. Rd. Array
Ers. Susp. Read Config	"1"	Config	Erase Sus. Read Array	Program Setup	Ers. Sus. Rd. Array	Erase	Ers. Sus. Rd. Array	Erase	Erase Sus. Status	Ers. Sus. Rd. Array
Ers. Susp. Read Query	"1"	CFI	Erase Sus. Read Array	Program Setup	Ers. Sus. Rd. Array	Erase	Ers. Sus. Rd. Array	Erase	Erase Sus. Status	Ers. Sus. Rd. Array
Erase (Done)	"1"	Status	Read Array	Program Setup	Erase Setup	Read Array			Read Status	Read Array

APPENDIX A
WSM CURRENT/NEXT STATES (Continued)

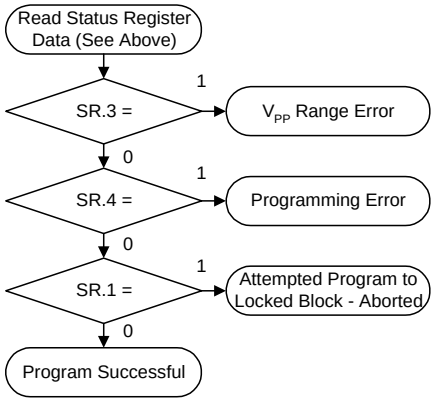
Current State	Command Input (and Next State)						
	Read Config (90H)	**Read Query (98H)**	**Lock Setup (60H)**	**Prot. Prog. Setup (C0H)**	**Lock Confirm (01H)**	**Lock Down Confirm (2FH)**	**Unlock Confirm (D0H)**
Read Array	Read Config.	Read Query	Lock Setup	Prot. Prog. Setup	Read Array		
Read Status	Read Config.	Read Query	Lock Setup	Prot. Prog. Setup	Read Array		
Read Config.	Read Config.	Read Query	Lock Setup	Prot. Prog. Setup	Read Array		
Read Query	Read Config.	Read Query	Lock Setup	Prot. Prog. Setup	Read Array		
Lock Setup	Locking Command Error				Lock Operation (Done)		
Lock Cmd. Error	Read Config.	Read Query	Lock Setup	Prot. Prog. Setup	Read Array		
Lock Operation (Done)	Read Config.	Read Query	Lock Setup	Prot. Prog. Setup	Read Array		
Prot. Prog. Setup	Protection Register Program						
Prot. Prog. (Not Done)	Protection Register Program (Not Done)						
Prot. Prog. (Done)	Read Config.	Read Query	Lock Setup	Prot. Prog. Setup	Read Array		
Prog. Setup	Program						
Program (Not Done)	Program (Not Done)						
Prog. Susp. Status	Prog. Susp. Read Config.	Prog. Susp. Read Query	Program Suspend Read Array				Program (Not Done)
Prog. Susp. Read Array	Prog. Susp. Read Config.	Prog. Susp. Read Query	Program Suspend Read Array				Program (Not Done)
Prog. Susp. Read Config.	Prog. Susp. Read Config.	Prog. Susp. Read Query	Program Suspend Read Array				Program (Not Done)
Prog. Susp. Read Query.	Prog. Susp. Read Config.	Prog. Susp. Read Query	Program Suspend Read Array				Program (Not Done)
Program (Done)	Read Config.	Read Query	Lock Setup	Prot. Prog. Setup	Read Array		
Erase Setup	Erase Command Error						Erase (Not Done)
Erase Cmd. Error	Read Config.	Read Query	Lock Setup	Prot. Prog. Setup	Read Array		
Erase (Not Done)	Erase (Not Done)						
Erase Suspend Status	Erase Suspend Read Config.	Erase Suspend Read Query	Lock Setup	Erase Suspend Read Array			Erase (Not Done)
Erase Suspend Array	Erase Suspend Read Config.	Erase Suspend Read Query	Lock Setup	Erase Suspend Read Array			Erase (Not Done)
Eras Sus. Read Config	Erase Suspend Read Config.	Erase Suspend Read Query	Lock Setup	Erase Suspend Read Array			Erase (Not Done)
Eras Sus. Read Query	Erase Suspend Read Config.	Erase Suspend Read Query	Lock Setup	Erase Suspend Read Array			Erase (Not Done)
Ers.(Done)	Read Config.	Read Query	Lock Setup	Prot. Prog. Setup	Read Array		

APPENDIX B
PROGRAM/ERASE FLOWCHARTS



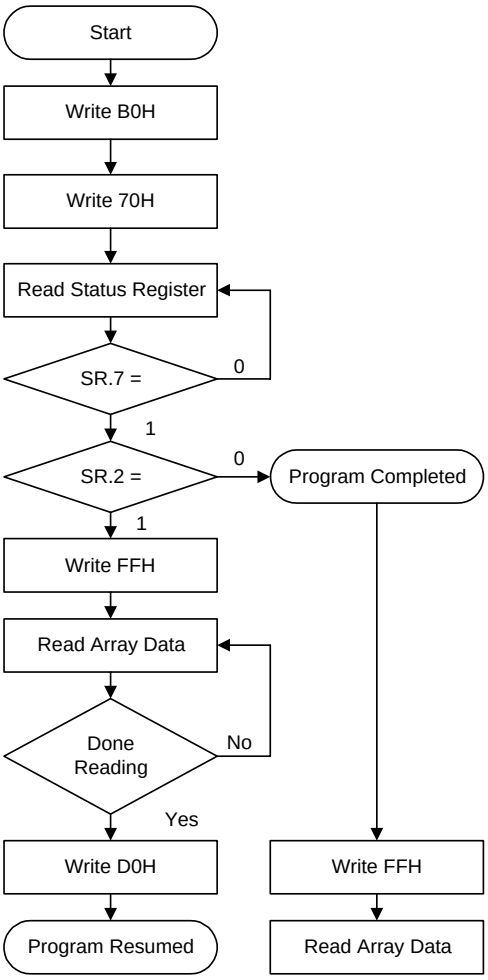
Bus Operation	Command	Comments
Write	Program Setup	Data = 40H
Write	Program	Data = Data to Program Addr = Location to Program
Read		Status Register Data Toggle CE# or OE# to Update Status Register Data
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy
Repeat for subsequent programming operations. SR Full 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



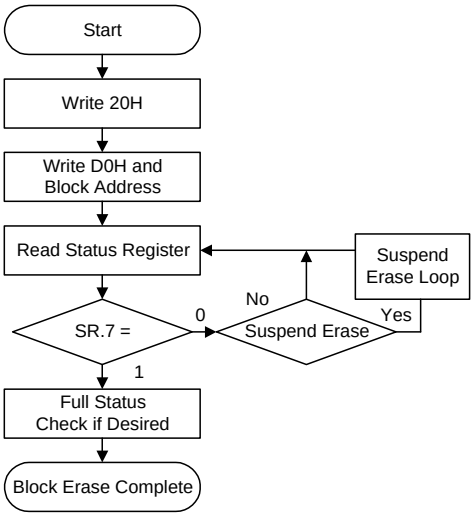
Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Low Detect
Standby		Check SR.4 1 = V_{PP} Program Error
Standby		Check SR.1 1 = Attempted Program to Locked Block - Program Aborted
SR.3 MUST be cleared, if set during a program attempt, before further attempts are allowed by the Write State Machine. SR.1, SR.3 and SR.4 are only cleared by the Clear Staus Register Command, in cases where multiple bytes are programmed before full status is checked. If an error is detected, clear the status register before attempting retry or other error recovery.		

Figure 10. Automated Word Programming Flowchart



Bus Operation	Command	Comments
Write	Program Suspend	Data = B0H Addr = X
Write	Read Status	Data=70H Addr=X
Read		Status Register Data Toggle CE# or OE# to Update Status Register Data Addr = X
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy
Standby		Check SR.2 1 = Program Suspended 0 = Program Completed
Write	Read Array	Data = FFH Addr = X
Read		Read array data from block other than the one being programmed.
Write	Program Resume	Data = D0H Addr = X

Figure 11. Program Suspend/Resume Flowchart



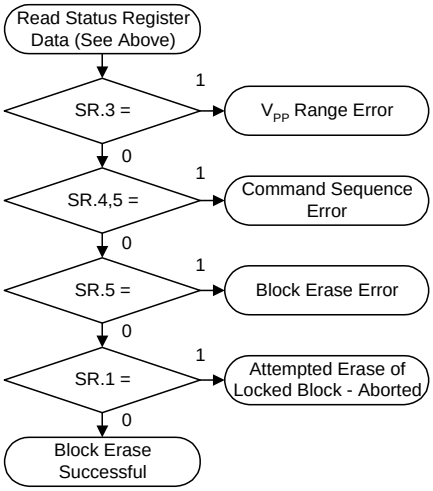
Bus Operation	Command	Comments
Write	Erase Setup	Data = 20H Addr = Within Block to Be Erased
Write	Erase Confirm	Data = D0H Addr = Within Block to Be Erased
Read		Status Register Data Toggle CE# or OE# to Update Status Register Data
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy

Repeat for subsequent block erasures.

Full Status Check can be done after each block erase or after a sequence of block erasures.

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

FULL STATUS CHECK PROCEDURE



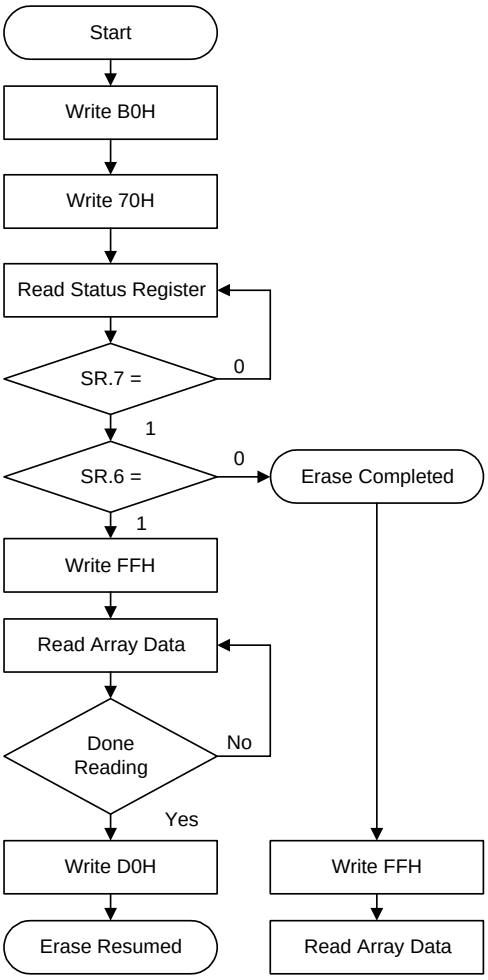
Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Low Detect
Standby		Check SR.4,5 Both 1 = Command Sequence Error
Standby		Check SR.5 1 = Block Erase Error
Standby		Check SR.1 1 = Attempted Erase of Locked Block - Erase Aborted

SR. 1 and 3 MUST be cleared, if set during an erase attempt, before further attempts are allowed by the Write State Machine.

SR.1, 3, 4, 5 are only cleared by the Clear Staus Register Command, in cases where multiple bytes are erased before full status is checked.

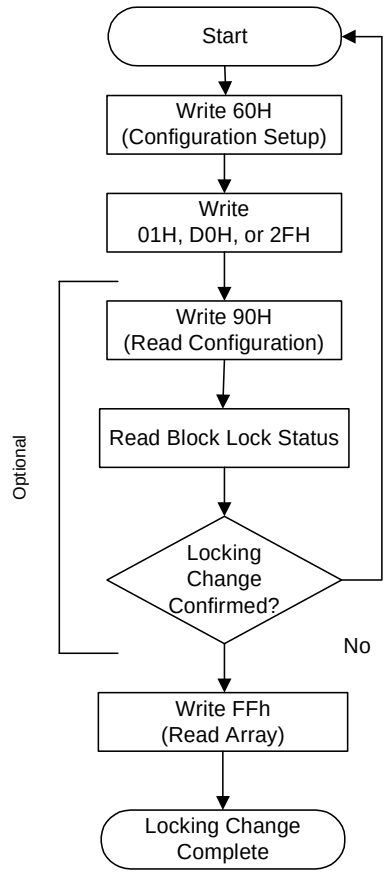
If an error is detected, clear the status register before attempting retry or other error recovery.

Figure 12. Automated Block Erase Flowchart



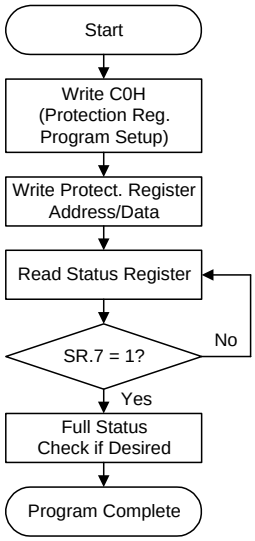
Bus Operation	Command	Comments
Write	Erase Suspend	Data = B0H Addr = X
Write	Read Status	Data=70H Addr=X
Read		Status Register Data Toggle CE# or OE# to Update Status Register Data Addr = X
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy
Standby		Check SR.6 1 = Erase Suspended 0 = Erase Completed
Write	Read Array	Data = FFH Addr = X
Read		Read array data from block other than the one being erased.
Write	Erase Resume	Data = D0H Addr = X

Figure 13. Erase Suspend/Resume Flowchart



Bus Operation	Command	Comments
Write	Config. Setup	Data = 60H Addr = X
Write	Lock, Unlock, or Lockdown	Data= 01H (Lock Block) D0H (Unlock Block) 2FH (Lockdown Block) Addr=Within block to lock
Write (Optional)	Read Configuration	Data = 90H Addr = X
Read (Optional)	Block Lock Status	Block Lock Status Data Addr = Second addr of block
Standby (Optional)		Confirm Locking Change on DQ_1, DQ_0. (See Block Locking State Table for valid combinations.)

Figure 14. Locking Operations Flowchart



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

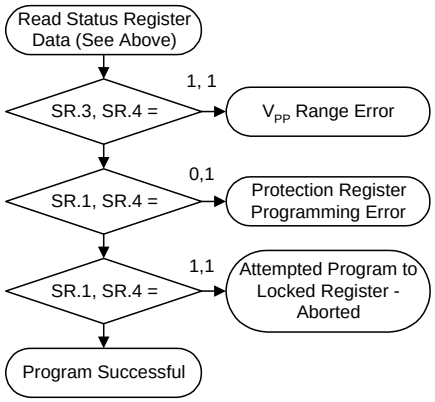
Protection Program operations can only be addressed within the protection register address space. Addresses outside the defined space will return an error.

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 0 1 1 V_{PP} 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.

Figure 15. Protection Register Programming Flowchart

APPENDIX C
COMMON FLASH INTERFACE QUERY STRUCTURE

This appendix 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.

C.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 C1. Summary of Query Structure Output As a Function of Device and Mode

Device	Hex Offset	Code	ASCII Value
Device Addresses	10:	51	“Q”
	11:	52	“R”
	12:	59	“Y”

Table C2. Example of Query Structure Output of x16 and x8 Devices

Word Addressing			Byte Addressing		
Offset	**Hex Code**	**Value**	**Offset**	**Hex Code**	**Value**
A_{15}–A_0	**D_{15}–D_0**		**A_7–A_0**	**D_7–D_0**	
0010h	0051	"Q"	10h	51	"Q"
0011h	0052	"R"	11h	52	"R"
0012h	0059	"Y"	12h	59	"Y"
0013h	P_ID_{LO}	PrVendor	13h	P_ID_{LO}	PrVendor
0014h	P_ID_{HI}	ID #	14h	P_ID_{LO}	ID #
0015h	P_{LO}	PrVendor	15h	P_ID_{HI}	ID #
0016h	P_{HI}	TblAdr	16h	...	...
0017h	A_ID_{LO}	AltVendor	17h		
0018h	A_ID_{HI}	ID #	18h		
...	...	...	...		

C.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.

Table C3. 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	Command Set ID and Vendor Data Offset
1Bh	System Interface Information	Device Timing and Voltage Information
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 = The beginning location of a Block Address (e.g., 08000h is the beginning location of block 1 when the block size is 32 Kword).
3. Offset 15 defines "P" which points to the Primary Intel-specific Extended Query Table.

C.3 BLOCK LOCK 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 C4. Block Status Register

Offset	Length	Description	Add.	Value
(BA+2)h[1]	1	Block Lock Status Register	BA+2:	--00 or --01
		BSR.0 Block Lock Status 0 = Unlocked 1 = Locked	BA+2:	(bit 0): 0 or 1
		BSR.1 Block Lock-Down Status 0 = Not locked down 1 = Locked down	BA+2:	(bit 1): 0 or 1
		BSR 2–7: *Reserved for future use*	BA+2:	(bit 2–7): 0

NOTES:

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

C.4 CFI QUERY IDENTIFICATION STRING

The 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 C5. CFI Identification

Offset	Length	Description	Add.
10h	3	Query-unique ASCII string "QRY"	10 11: 12:
13h	2	Primary vendor command set and control interface ID code. 16-bit ID code for vendor-specified algorithms	13: 14:
15h	2	Extended Query Table primary algorithm address	15: 16:
17h	2	Alternate vendor command set and control interface ID code. 0000h means no second vendor-specified algorithm exists	17: 18:
19h	2	Secondary algorithm Extended Query Table address. 0000h means none exists	19: 1A:

C.5 SYSTEM INTERFACE INFORMATION

Table C6. System Interface Information

Offset	Length	Description	Add.	Hex Code	Value
1Bh	1	V_{CC} logic supply minimum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 BCD volts	1B:	--24	2.4 V
1Ch	1	V_{CC} logic supply maximum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 BCD volts	1C:	--30	3.0 V
1Dh	1	V_{PP} [programming] supply minimum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 HEX volts	1D:	--B4	11.4 V
1Eh	1	V_{PP} [programming] supply maximum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 HEX volts	1E:	--C6	12.6 V
1Fh	1	"n" such that typical single word program time-out = 2^n µs	1F:	--05	32 µs
20h	1	"n" such that typical max. buffer write time-out = 2^n µs	20:	--00	NA
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	512 µs
24h	1	"n" such that maximum buffer write time-out = 2^n times typical	24:	--00	NA
25h	1	"n" such that maximum block erase time-out = 2^n times typical	25:	--03	8 s
26h	1	"n" such that maximum chip erase time-out = 2^n times typical	26:	--00	NA

C.6 DEVICE GEOMETRY DEFINITION

Table C7. 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 28:00,29:00 28:01,29:00 28:02,29:00	28: 29:	--01 --00	x16
2Ah	2	"n" such that maximum number of bytes in write buffer = 2^n	2A: 2B:	--00 --00	0
2Ch	1	Number of erase block regions within device: 1. x = 0 means no erase blocking; the device erases in "bulk" 2. x specifies the number of device or partition regions with one or more contiguous same-size erase blocks. 3. Symmetrically blocked partitions have one blocking region 4. Partition size = (total blocks) x (individual block size)	2C:	--02	2
2Dh	4	Erase Block Region 1 Information bits 0–15 = y, y+1 = number of identical-size erase blocks bits 16–31 = z, region erase block(s) size are z x 256 bytes	2D: 2E: 2F: 30:		
31h	4	Erase Block Region 2 Information bits 0–15 = y, y+1 = number of identical-size erase blocks bits 16–31 = z, region erase block(s) size are z x 256 bytes	31: 32: 33: 34:		

Device Geometry Definition

Address	8 Mbit		16 Mbit	
	–B	–T	–B	–T
27:	--14	--14	--15	--15
28:	--01	--01	--01	--01
29:	--00	--00	--00	--00
2A:	--00	--00	--00	--00
2B:	--00	--00	--00	--00
2C:	--02	--02	--02	--02
2D:	--07	--0E	--07	--1E
2E:	--00	--00	--00	--00
2F:	--20	--00	--20	--00
30:	--00	--01	--00	--01
31:	--0E	--07	--1E	--07
32:	--00	--00	--00	--00
33:	--00	--20	--00	--20
34:	--01	--00	--01	--00

C.7 INTEL-SPECIFIC EXTENDED QUERY TABLE

Table C8. Primary-Vendor Specific Extended Query

Offset[1] P = 35h	Length	Description (Optional flash features and commands)	Add.	Hex Code	Value
(P+0)h	3	Primary extended query table	35:	--50	"P"
(P+1)h		Unique ASCII string "PRI"	36:	--52	"R"
(P+2)h			37:	--49	"I"
(P+3)h	1	Major version number, ASCII	38:	--31	"1"
(P+4)h	1	Minor version number, ASCII	39:	--30	"0"
(P+5)h	4	Optional feature and command support (1=yes, 0=no)	3A:	--66	
(P+6)h		*bits 9–31 are reserved; undefined bits are "0." If bit 31 is*	3B:	--00	
(P+7)h		*"1" then another 31 bit field of optional features follows at*	3C:	--00	
(P+8)h		*the end of the bit-30 field.*	3D:	--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 = 0		No
		bit 4 Queued erase supported	bit 4 = 0		No
		bit 5 Instant individual block locking supported	bit 5 = 1		Yes
		bit 6 Protection bits supported	bit 6 = 1		Yes
		bit 7 Page mode read supported	bit 7 = 0		No
		bit 8 Synchronous read supported	bit 8 = 0		No
(P+9)h	1	Supported functions after suspend: Read Array, Status, Query Other supported operations are: *bits 1–7 reserved; undefined bits are "0"*	3E:	--01	
		bit 0 Program supported after erase suspend	bit 0 = 1		Yes
(P+A)h	2	Block status register mask	3F:	--03	
(P+B)h		*bits 2–15 are Reserved; undefined bits are "0"*	40:	--00	
		bit 0 Block Lock-Bit Status register active	bit 0 = 1		Yes
		bit 1 Block Lock-Down Bit Status active	bit 1 = 1		Yes
(P+C)h	1	V_{CC} logic supply highest performance program/erase voltage bits 0–3 BCD value in 100 mV bits 4–7 BCD value in volts	41:	--30	3.0 V
(P+D)h	1	V_{PP} optimum program/erase supply voltage bits 0–3 BCD value in 100 mV bits 4–7 HEX value in volts	42:	--C0	12.0 V

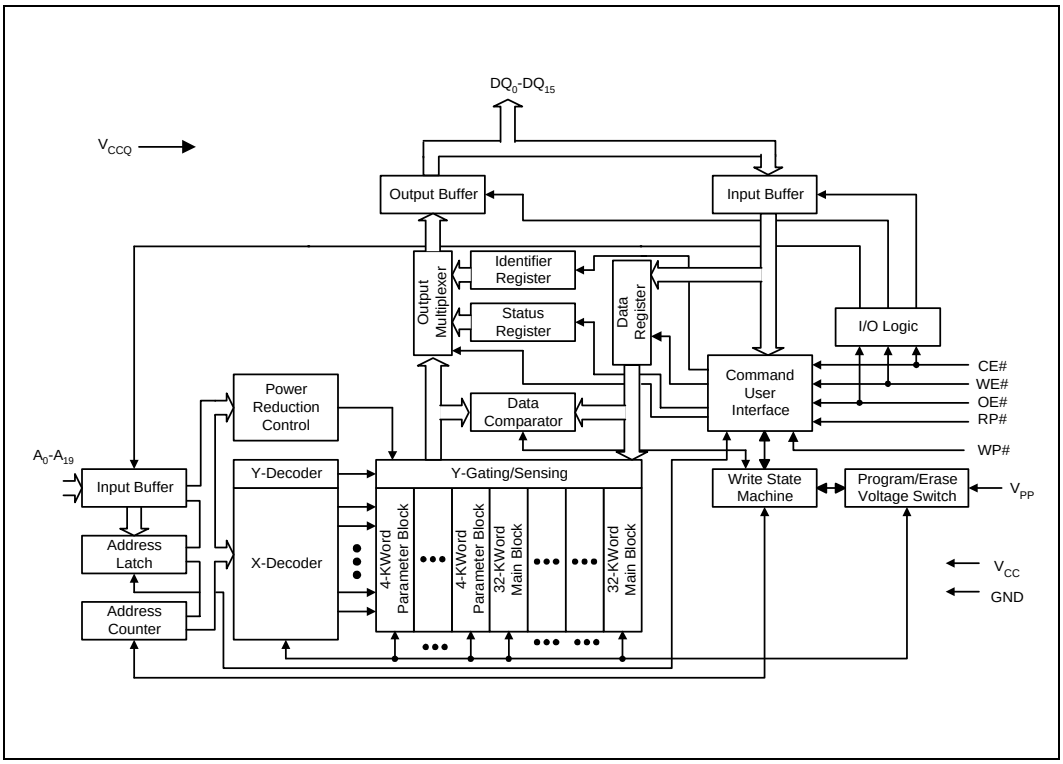
Table C9. Protection Register Information

Offset[1] P = 35h	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	43:	--01	01
(P+F)h	4	Protection Field 1: Protection Description	44:	--80	80h
(P+10)h		This field describes user-available One Time Programmable	45:	--00	00h
(P+11)h		(OTP) Protection register bytes. Some are pre-programmed	46:	--03	8 byte
(P+12)h		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. bits 0–7 = Lock/bytes Jedec-plane physical low address bits 8–15 = Lock/bytes Jedec-plane physical high address bits 16–23 = "n" such that 2n = factory pre-programmed bytes bits 24–31 = "n" such that 2n = user programmable bytes	47:	--03	8 byte
(P+13)h		Reserved for future use	48:		

NOTES:

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

APPENDIX D
ARCHITECTURE BLOCK DIAGRAM



APPENDIX E
WORD-WIDE MEMORY MAP DIAGRAMS

8-Mbit and 16-Mbit Word-Wide Memory Addressing

Top Boot			Bottom Boot		
Size (KW)	**8M**	**16M**	**Size (KW)**	**8M**	**16M**
4	7F000-7FFFF	FF000-FFFFF	32		
4	7E000-7EFFF	FE000-FEFFF	32		
4	7D000-7DFFF	FD000-FDFFF	32		
4	7C000-7CFFF	FC000-FCFFF	32		
4	7B000-7BFFF	FB000-FBFFF	32		
4	7A000-7AFFF	FA000-FAFFF	32		
4	79000-79FFF	F9000-F9FFF	32		
4	78000-78FFF	F8000-F8FFF	32		
32	70000-77FFF	F0000-F7FFF	32		
32	68000-6FFFF	E8000-EFFFF	32		
32	60000-67FFF	E0000-E7FFF	32		
32	58000-5FFFF	D8000-DFFFF	32		
32	50000-57FFF	D0000-D7FFF	32		
32	48000-4FFFF	C8000-CFFFF	32		
32	40000-47FFF	C0000-C7FFF	32		
32	38000-3FFFF	B8000-BFFFF	32		
32	30000-37FFF	B0000-B7FFF	32		
32	28000-2FFFF	A8000-AFFFF	32		
32	20000-27FFF	A0000-A7FFF	32		
32	18000-1FFFF	98000-9FFFF	32		
32	10000-17FFF	90000-97FFF	32		
32	08000-0FFFF	88000-8FFFF	32		
32	00000-07FFF	80000-87FFF	32		
32		78000-7FFFF	32		
32		70000-77FFF	32		
32		68000-6FFFF	32		
32		60000-67FFF	32		
32		58000-5FFFF	32		
32		50000-57FFF	32		
32		48000-4FFFF	32		
32		40000-47FFF	32		
32		38000-3FFFF	32		
This column continues on next page			This column continues on next page		

8-Mbit and 16-Mbit Word-Wide Memory Addressing (Continued)

Top Boot			Bottom Boot		
Size (KW)	**8M**	**16M**	**Size (KW)**	**8M**	**16M**
32		30000-37FFF	32		F8000-FFFFF
32		28000-2FFFF	32		F0000-F7FFF
32		20000-27FFF	32		E8000-EFFFF
32		18000-1FFFF	32		E0000-E7FFF
32		10000-17FFF	32		D8000-DFFFF
32		08000-0FFFF	32		D0000-D7FFF
32		00000-07FFF	32		C8000-CFFFF
32			32		C0000-C7FFF
32			32		B8000-BFFFF
32			32		B0000-B7FFF
32			32		A8000-AFFFF
32			32		A0000-A7FFF
32			32		98000-9FFFF
32			32		90000-97FFF
32			32		88000-8FFFF
32			32		80000-87FFF
32			32	78000-7FFFF	78000-7FFFF
32			32	70000-77FFF	70000-77FFF
32			32	68000-6FFFF	68000-6FFFF
32			32	60000-67FFF	60000-67FFF
32			32	58000-5FFFF	58000-5FFFF
32			32	50000-57FFF	50000-57FFF
32			32	48000-4FFFF	48000-4FFFF
32			32	40000-47FFF	40000-47FFF
32			32	38000-3FFFF	38000-3FFFF
32			32	30000-37FFF	30000-37FFF
32			32	28000-2FFFF	28000-2FFFF
32			32	20000-27FFF	20000-27FFF
32			32	18000-1FFFF	18000-1FFFF
32			32	10000-17FFF	10000-17FFF
32			32	08000-0FFFF	08000-0FFFF
32			4	07000-07FFF	07000-07FFF
32			4	06000-06FFF	06000-06FFF
32			4	05000-05FFF	05000-05FFF
32			4	04000-04FFF	04000-04FFF
32			4	03000-03FFF	03000-03FFF
32			4	02000-02FFF	02000-02FFF
32			4	01000-01FFF	01000-01FFF
32			4	00000-00FFF	00000-00FFF

APPENDIX F
DEVICE ID TABLE

Read Configuration Addresses and Data

Item		Address	Data
Manufacturer Code	x16	00000	0089
Device Code			
8-Mbit x 16-T	x16	00001	88C0
8-Mbit x 16-B	x16	00001	88C1
16-Mbit x 16-T	x16	00001	88C2
16-Mbit x 16-B	x16	00001	88C3

NOTE: Other locations within the configuration address space are reserved by Intel for future use.

APPENDIX G
PROTECTION REGISTER ADDRESSING

Word-Wide Protection Register Addressing

Word	Use	A7	A6	A5	A4	A3	A2	A1	A0
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_{21}–A_8 = 0.