1.8 Volt Intel® Dual-Plane Flash Memory

28F320D18 (x16)

Product Preview Datasheet

Product Features

- 32-Mbit density with 16-Bit Data Bus
- High Performance Reads
  - — 110/40 ns 4-Word Page Mode
  - — 40 MHz (110/20 ns) Zero Wait-State Synchronous Burst Mode
- Dual Partition Architecture
  - — 25%/75% Partition Sizes 32 Mb ⇒ 8 Mb + 24 Mb
  - — Program or Erase during Reads
  - — Status Register for Each Partition
- Low Power Operation
  - — 1.8 V Read and Write Operations
  - — V_{CCQ} for I/O Isolation and System Compatibility
  - — Automatic Power Savings Mode
- Enhanced Code + Data Storage
  - — Flash Data Integrator (FDI) Software Optimized
  - — 5 µs Typical Program/Erase Suspends
- 128-Bit Protection Register
  - — 64 Unique Device Identifier Bits
  - — 64 User-Programmable OTP Bits
- µBGA* CSP 60-Ball 7x8 Matrix (four support balls)
- Flexible Blocking Architecture
  - — Eight, 4-Kword Parameter Code/Data Blocks
  - — Sixty-three, 32-Kword Main Code/Data Blocks
- Enhanced Data Protection
  - — V_{PP} = GND ⇒ Absolute Write Protection
  - — Erase/Program Lockout during Power Transitions
  - — Individual Dynamic Zero-Latency Block Locking
  - — Individual Block Lock-Down
- Automated Program/Erase Algorithms
  - — 1.8 V Low-Power 22 µs/Word (Typ) Programming
  - — 12 V No Glue Logic 8 µs/Word (Typ) Production Programming and 1.1 sec Erase (Typ)
- Cross-Compatible Command Support
  - — Intel Basic Command Set
  - — Common Flash Interface (CFI)
- Extended Temperature –40° C to +85° C
- Minimum 100,000 Block Erase Cycles
- ETOX™ VI Flash Technology (0.25 µm)

The 1.8 Volt Intel® Dual-Plane Flash memory provides high performance asynchronous and synchronous burst reads. It is an ideal memory for low-voltage burst CPUs. Combining high read performance with flash memory's intrinsic nonvolatility, 1.8 Volt Dual-Plane Flash memory eliminates the traditional system-performance paradigm of shadowing redundant code memory from slow nonvolatile storage to faster execution memory. It reduces the total memory requirement that increase reliability and reduces overall system power consumption and cost.

The 1.8 Volt Dual-Plane Flash memory's two partitions allow background programming or erasing to occur in one partition while program-execution reads take place in the other partition. This allows for higher data write throughput compared to single partition architectures. The dual partition architecture also allows two processors to interleave code operations while program and erase operations take place in the background.

1.8 Volt Dual-Plane Flash memory is manufactured on Intel® 0.25 µm ETOX™ VI process technology. It is available in an industry-standard µBGA* CSP package which is ideal for board-constrained applications.

Order Number: 290672-002
October 1999

Contents

Revision History

Date of Revision	Version	Description
09/20/99	-001	Original version
10/12/99	-002	Corrected Figure 1, “60-Ball µBGA* Package Ballout” Corrected titles for Figure 24 —Figure 27

1.0 Introduction

This datasheet contains information about the 32-Mbit 1.8 Volt Intel® Dual-Plane Flash memory. Section 1.0 provides a flash memory overview. Sections 2.0 through 6.0 describe the memory functionality. Section 7.0 describes the design considerations for this device and Section 8.0 describes the electrical specifications for extended temperature product offerings.

1.1 Document Conventions

Throughout this document, references are made to bottom, top, parameter, and main partitions. To clarify these references, the following convention has been adopted:

- **Main partition**: contains only main blocks.
- **Parameter partition**: contains a mixture of main and parameter blocks.
- **Bottom partition:** the partition located at the lowest physical device address. This partition may be a main partition or a parameter partition.
- **Top partition**: the partition located at the highest physical device address. This partition may be a main partition or a parameter partition.
- **Bottom parameter device**: has the parameter partition at the bottom of the memory map with the parameter blocks at the bottom of that partition. This was formerly referred to as bottom-boot. Since many applications actually boot and execute code from the top (main) blocks and treat the bottom (parameter) blocks as data blocks, bottom-boot and top-boot have become misnomers, thus the nomenclature change.
- **Top parameter device**: has the parameter partition at the top of the memory map with its parameter blocks at the top. This was formerly referred to as a top-boot device.
- **Main block(s)**: 32-Kword block
- **Parameter block(s)**: 4-Kword block

1.2 Product Overview

The 1.8 Volt Dual-Plane Flash memory provides simultaneous read while write/erase capability. The memory provides high performance reads at low voltage with a 16-bit data bus. Individually erasable blocks are optimally sized for code and data storage. The eight 4-Kword parameter blocks are located in the parameter partition. The rest of the device is grouped into sixty-three 32-Kword main blocks within the main and parameter partitions.

By dividing the flash memory array into two isolated partitions, simultaneous operation capability permits program or block-erase operations during read operations. The main partition is ¾ of the memory and contains only main blocks. The parameter partition is ¼ of the total memory and contains parameter blocks and main blocks. Burst reads are limited to within a partition. Usage of simultaneous modes will be described further throughout this document.

The device’s optimized architecture and interface dramatically increases read performance beyond asynchronous reads. The device supports asynchronous word accesses, 4-word page mode and synchronous burst reads from main blocks. Parameter blocks support asynchronous word accesses, 4-word page mode and single synchronous reads only.

Upon initial power up or return from reset, the device defaults to a standard asynchronous page-mode read configuration. Writing to the read configuration register at any device address enables both partitions' synchronous burst reads. In synchronous burst mode, the CLK input increments an internal burst address generator, synchronizes flash memory with the host CPU, and outputs data every CLK cycle. A WAIT# output signal provides easy CPU-to-flash memory communication and synchronization.

In addition to the enhanced architecture and interface, 1.8 Volt Dual-Plane Flash memory incorporates technology that enables fast factory programming/erasing and low-power designs. Specifically designed for low-voltage systems, 1.8 Volt Dual-Plane Flash memory supports read operations at 1.8 V V_{CC} and block erase and program operations at 1.8 V or 12 V V_{PP}. The 12 V V_{PP} option renders the fastest program/erase performance that can increase factory throughput. With the 1.8 V V_{PP} option, V_{CC} and V_{PP} can be tied together for a simple, ultra low-power design. In addition to the voltage flexibility, the dedicated V_{PP} pin gives complete data protection when $V_{PP} \leq V_{PPLK}$.

The device's Command User Interface (CUI) is the system processor's interface to 1.8 Volt Dual-Plane Flash memory's internal operation. Writing a valid command sequence to the CUI initiates device Write State Machine (WSM) controlled automation that automatically executes the block-erase and program algorithms and timings. The status register indicates the WSM's state by indicating block erase or program completion and status.

An industry-standard command sequence invokes block-erase and program automation. Each block erase operation erases one block. Data is programmed in word increments. Erase suspend allows system software to pause a block erase so it can read or program data in another block in the same partition. Program suspend allows system software to suspend programming so it can read from another location in the same partition. It is also possible to nest suspends as follows: suspend erase in the first partition, start programming in the second partition, suspend programming in the second partition and then read from the second partition.

1.8 Volt Dual-Plane Flash memory offers two low-power savings features: Automatic Power Savings (APS) and standby mode. The device automatically enters APS mode following read cycle completion. Standby mode is initiated when the system deselects the device by driving CE# inactive. RST# also resets the device to read array mode, provides write protection, and clears the status register. Combined, these two features significantly reduce power consumption.

2.0 Product Description

2.1 Ballouts

The Intel 1.8 Volt Dual-Plane Flash memory is available in a 60-ball (7 x 8 matrix with four support balls) µBGA* CSP (Chip Scale Package) package with 0.75 mm ball pitch that is ideal for board-constrained applications.

Figure 1, "60-Ball µBGA* Package Ballout" on page 4 shows the component ballout.

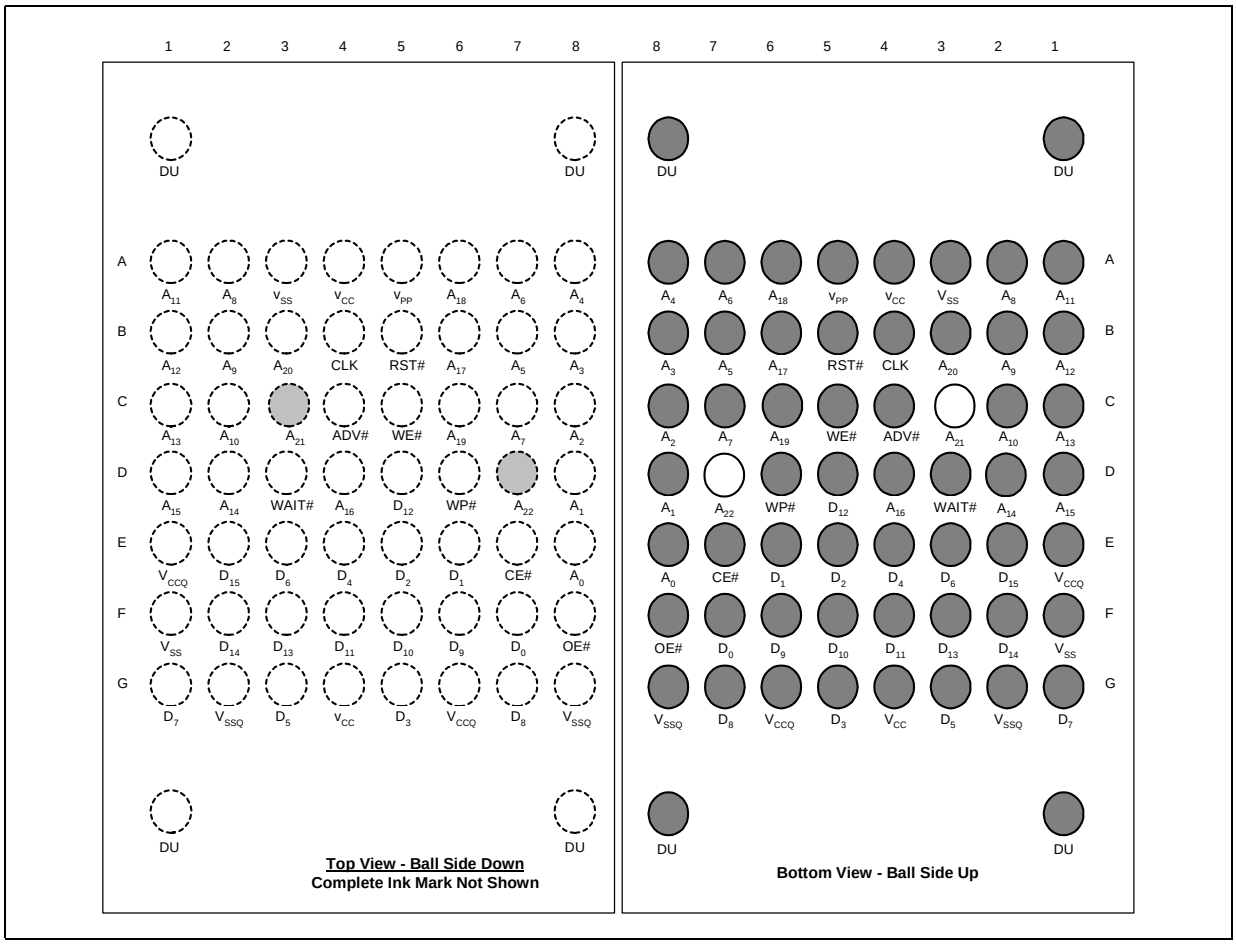
2.2 Ball Description

Figure 1, "Ball Descriptions" on page 3 describes ball usage.

Table 1. Ball Descriptions

Sym	Type	Name and Function
A_0–A_{20}	I	**ADDRESS INPUTS:** for memory addresses. 32-Mbit: $A_{0\text{-}20}$
DQ_0–DQ_{15}	I/O	**DATA INPUT/OUTPUTS:** Inputs data and commands during write cycles, outputs data during memory, status register, and configuration code reads. Data balls float when the chip or outputs are deselected. Data is internally latched during writes.
CLK	I	**CLOCK:** Synchronizes the memory to the system bus operating frequency in synchronous-read configuration. The first rising (or falling if RCR.6 is 0) CLK edge latches the address when ADV# is active or upon a rising ADV# edge. This is used only for synchronous operation.
ADV#	I	**ADDRESS VALID:** ADV# indicates valid address presence on address inputs. Addresses are latched on ADV#'s rising edge during read and write operations. This is used only for synchronous operation.
RST#	I	**RESET:** When low, RST# resets internal automation and inhibits write operations. This provides data protection during power transitions. RST#-high enables normal operation. Exit from reset places the device in asynchronous read array mode.
OE#	I	**OUTPUT ENABLE:** OE# gates the device's outputs during a read cycle.
WE#	I	**WRITE ENABLE:** WE# controls writes to the CUI and array. Addresses and data are latched on the WE# pulse's rising edge.
WP#	I	**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 3.2 for details on block locking.
WAIT#	O	**WAIT:** Feeds back data valid status in synchronous burst mode while OE# is asserted. When high during a burst sequence, data is valid. WAIT#-low indicates invalid data. WAIT# is pulled high by an internal register. Several component WAIT# signals can be tied together to drive one system WAIT signal. WAIT# is used only for synchronous operation. It also works during a 4, 8-word burst mode if the No-Wrap bit (RCR.3) is set to 1.
V_{PP}	Pwr	**BLOCK ERASE AND PROGRAM POWER:** A valid voltage on this pin allows block erase or data programming. Memory contents cannot be altered when $V_{PP} \le V_{PPLK}$. Block erase and program at invalid V_{PP} voltages should not be attempted. 11.4 V–12.6 V V_{PP} can be applied to main blocks for 1000 cycles maximum and to parameter blocks for 2500 cycles. Maximum V_{PP} can be connected to 12 V for 80 hours maximum total. Use of this pin at 12 V beyond these limits may reduce block cycling capability or cause permanent damage.
V_{CC}	Pwr	**DEVICE POWER SUPPLY (1.65 V–1.95 V):** Flash memory writes are inhibited at $V_{CC} \le V_{LKO}$. Device operations at invalid V_{CC} voltages should not be attempted.
V_{CCQ}	Pwr	**OUTPUT POWER SUPPLY (1.65 V–1.95 V):** Enables all outputs to be driven at 1.65 V to 1.95 V. This input may be tied directly to V_{CC}.
V_{SSQ}	Pwr	**I/O GROUND:** Do not float any ground pins.
V_{SS}	Pwr	**GROUND:** Do not float any ground pins.
NC		**NO CONNECT:** Lead is not internally connected; it may be driven or floated.
DU		**DON'T USE:** Do not use this pin. This pin should not be connected to any power supplies, signals or other pins.

Figure 1. 60-Ball µBGA* Package Ballout



NOTE: Flash upgrade address lines are shown for A_{21} (64-Mbit flash) and A_{22} (128-Mbit flash) for information purposes only since these devices are currently not available. Lower density devices will not have the upper address solder balls. Routing is not recommended in this area.

2.3 Memory Blocking Organization

The device is divided into two physical partitions. This allows it to perform simultaneous read-while-write and read-while-erase operations. The device's asymmetrically blocked architecture enables system code and data integration within a single flash device. Each block can be erased independently. See Figure 2, "32-Mbit Top Parameter Memory Map" on page 6 and and *Figure 3, "32-Mbit Bottom Parameter Memory Map" on page 7* for block address locations.

2.3.1 Dual Physical Partitions

The device has an 8-Mb partition (8 parameter blocks plus 15 main blocks) and a 24-Mb partition (48 main blocks).

Only one partition at a time is allowed to be in program or erase mode. It is also not possible to do burst reads that cross partition boundaries. Table 2 on page 9 summarizes simultaneous commands allowed with dual partitions. For a detailed description of commands allowed using dual partitions see Table 11 on page 38.

2.3.2 Parameter Blocks

The memory architecture includes parameter blocks that allow storage of frequently updated small parameters that would normally be stored in EEPROM. By using software techniques, the word-rewrite functionality of EEPROMs can be emulated. The device contains eight 4-Kword (4,096-words) parameter blocks within the parameter partition.

2.3.3 Main Blocks

The remainder of the array is divided into equal-size 32-Kword main blocks that can store code and/or data. See Figure 2, “32-Mbit Top Parameter Memory Map” on page 6 and Figure 3, “32-Mbit Bottom Parameter Memory Map” on page 7 .

Figure 2. 32-Mbit Top Parameter Memory Map

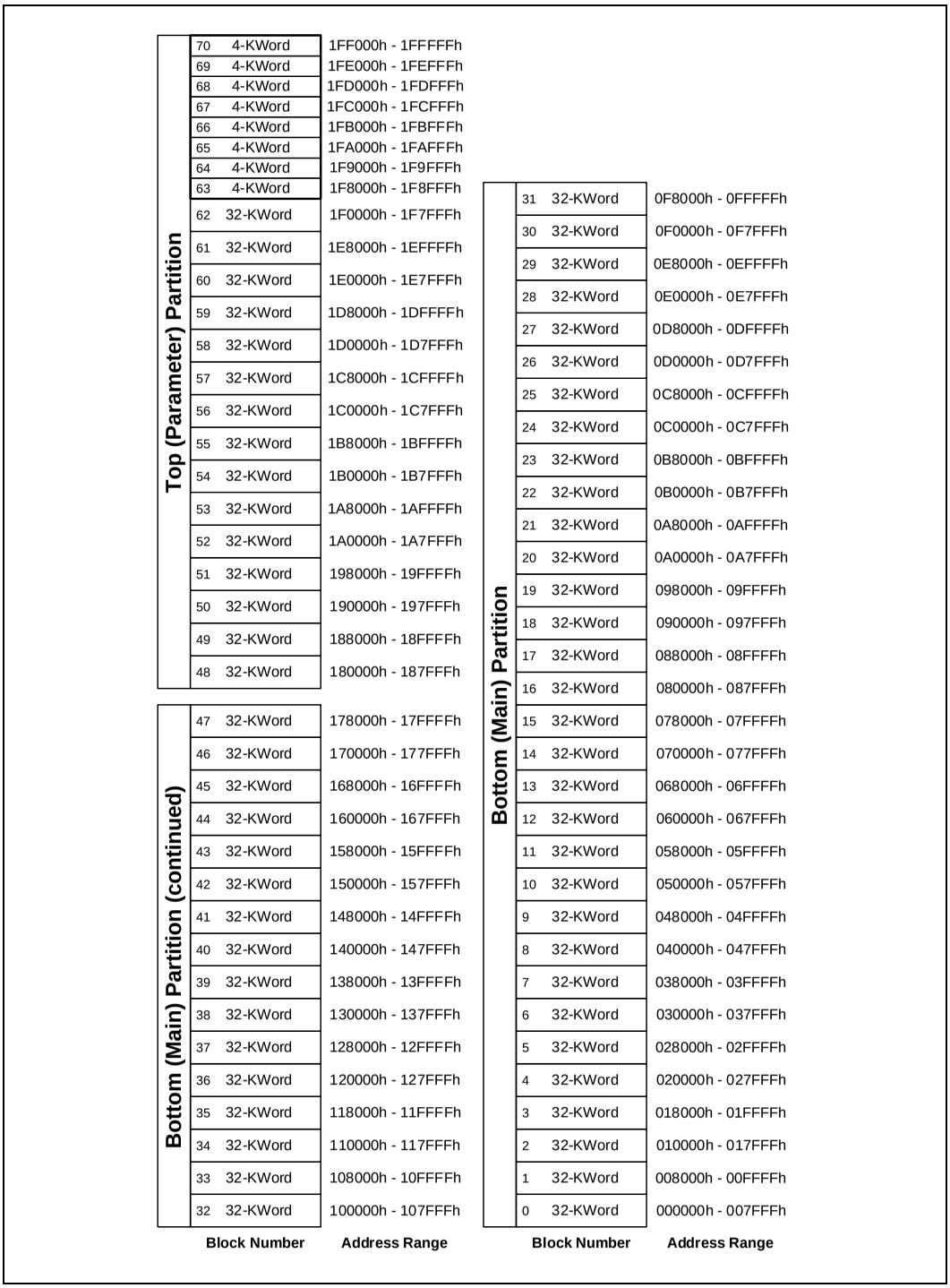
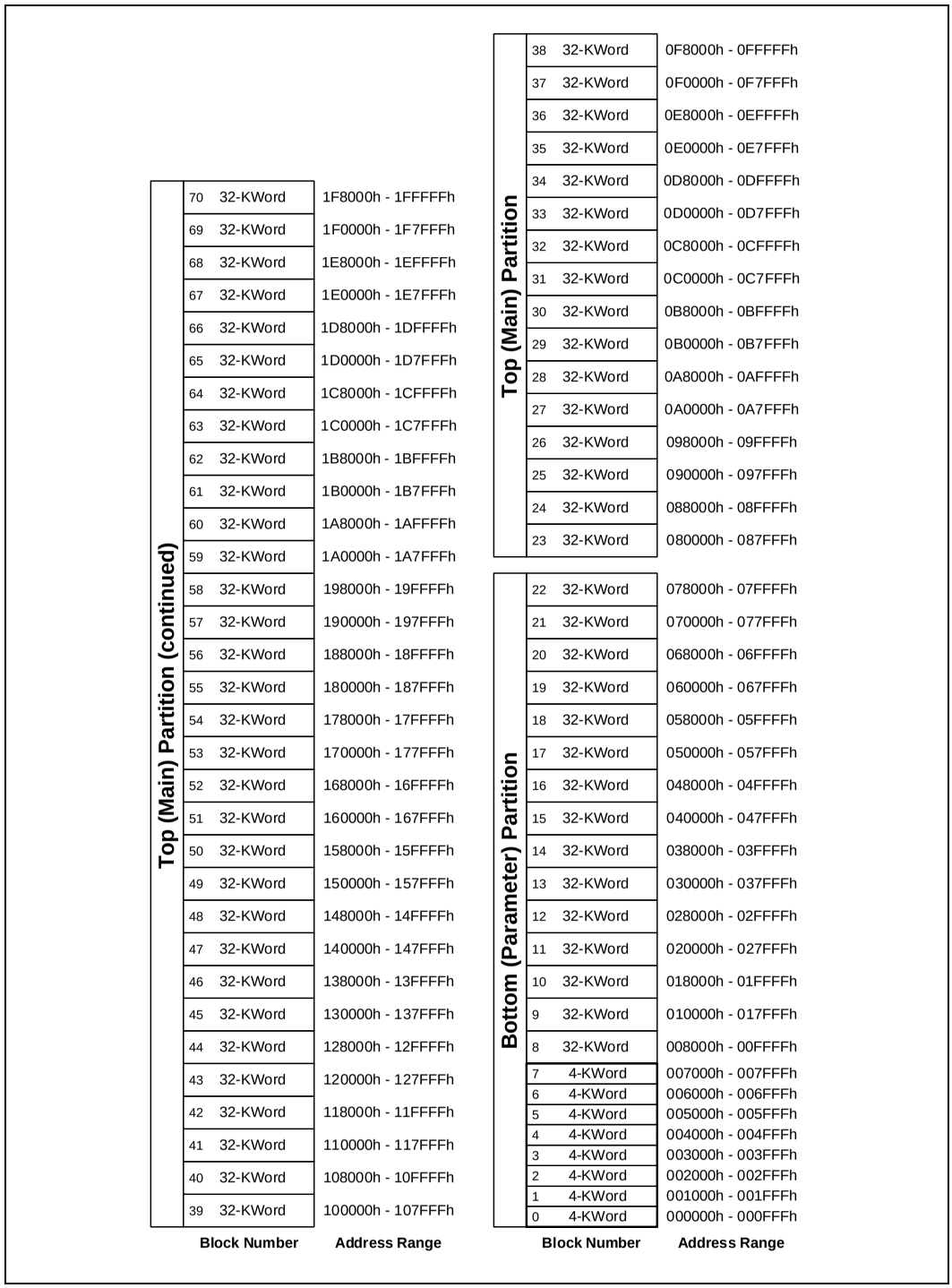


Figure 3. 32-Mbit Bottom Parameter Memory Map



3.0 Principles of Operation

The 1.8 Volt Dual-Plane Flash memory component includes an on-chip Write State Machine (WSM) to manage block erase and program. It allows for CMOS-level control inputs, fixed power supplies, and minimal processor overhead with RAM-like interface timings.

3.1 Bus Operations

The local CPU reads and writes flash memory in-system. All flash memory read and write cycles conform to standard microprocessor bus cycles.

3.1.1 Read

The flash memory's bottom partition, whether top- or bottom-parameter configuration, has three read modes available: read array, identifier/CFI codes, and status register. The top partition has only read array and status register read modes. Each partition can be in one of its read modes independent of the other partition's mode. However simultaneous read commands in both partitions are not allowed.

Page mode and synchronous burst mode for both partitions are enabled by writing the Set Read Configuration Register command to any device address. This sets the read configuration, burst order, burst length, and frequency configuration.

For all read operations, CE# must be driven active to enable the device. The device internally decodes upper address inputs to determine which partition is activated.

OE# controls data outputs (DQ_0–DQ_{15}) onto the I/O bus when active. WE# must be at V_{IH}.

3.1.2 Output Disable

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

3.1.3 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. 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.

3.1.4 Write

The Command User Interface (CUI) does not occupy an addressable memory location within its partition, but it must be accessed by the system processor at the correct partition address range. Programming/erasing may occur in only one partition at a time. The other partition must be in one of the read modes (see Table 2 on page 9).

Table 2. Simultaneous Commands Allowed with Dual Partitions (1,2)

If one partition is:	Then the commands allowed in the other partition are:						
	Read	Read Status	Read ID/CFI	Program	Erase	Program Suspend	Erase Suspend
Idle	✓	✓	✓	✓	✓	✓	✓
Reading							
Reading Status Register							
Reading ID/CFI							
Programming	✓	✓	✓				
Erasing	✓	✓	✓				
Program Suspended	✓	✓	✓				
Erase Suspended	✓	✓	✓	✓		✓	

NOTES:
1. For detailed description of the command allowed using dual partitions see Table 11 on page 38.

2. Dual Partition Restrictions:
   a. Status register reflects partition state, not WSM state — this allows a status register for each partition.
   b. Only one partition can be programmed or erased at a time — no command queuing.
   c. Commands must be written to an address within the block targeted by that command.
   d. It is not possible to do burst reads that cross partition boundaries.

3.1.5 Reset

The device enters a reset mode when RST# is driven low. In reset mode, internal circuitry is turned off and outputs are placed in a high-impedance state.

After return from reset, a time t_{PHQV} is required until outputs are valid, and a delay (t_{PHWL} or t_{PHEL}) is required before a write sequence can be initiated. After this wake-up interval, normal operation is restored. The device defaults to read array mode, the status register is set to 80H, and the read configuration register defaults to asynchronous reads.

If RST# is taken low during a block erase or program operation, the operation will be aborted and the memory contents at the aborted location are no longer valid. See Figure 30, “AC Waveform for Reset Operations” on page 64 for detailed information regarding reset timings.

3.1.6 Read Query

The read query mode is only available in the bottom partition and requires that the Read Query command be written to the bottom partition. The mode outputs Common Flash Interface (CFI) data when the device is read. The CFI data structure contains information such as block size, density, command set and electrical specifications. In this mode, read cycles retrieve CFI information. To return to read array mode, write the Read Array command (FFH).

3.2 Flexible Block Locking

For both configuration and status modes, 1.8 Volt Dual-Plane Flash memory will decode the block locking and status registers within each partition.

1.8 Volt Dual-Plane Flash memory offers an instant, individual block locking scheme that allows any block to be locked or unlocked with no latency, enabling instant code and data protection. 1.8 Volt Dual-Plane Flash memory also features a hardware lock-down on main blocks and parameter blocks. This enables critical code and data security while other blocks are programmed or erased.

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).

Each block can be set to Locked, Unlocked, and Lock-Down, as described in the following sections. A comprehensive state table for the locking functions is shown in Table 3, “Block Locking State Transitions” on page 11, and a flowchart for locking operations is shown in *Figure 11, “Locking Operations Flowchart” on page 29*. A block can be Locked, Unlocked and Locked-Down in one partition while programming or erasing the other partition.

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, and Z = bit DQ_0 of the Block Lock status. Table 3 defines all of these possible locking states.

Table 3. Block Locking State Transitions

Current State				Erase/Program Allowed?	Lock Command Input Result (Next State)		
WP#	DQ_1	DQ_0	Name		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].

3.2.1 Locking Operation

The following summarizes the locking operation.

- All blocks are locked on power-up. They can then 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. Commands can then 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.

3.2.2 Locked State

All blocks default to locked on power-up or reset (states [001] or [101]). A program or erase operation 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 command. An unlocked block can be locked by writing the Lock command sequence, 60H followed by 01H.

3.2.3 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 command. A locked block can be unlocked by writing the Unlock command sequence, 60H followed by D0H.

3.2.4 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 re-locked [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.

3.2.5 Reading a Block's Lock Status

The lock status of every block can be read in the device identifier read mode of the device. To enter this mode, write 90H to the device. Subsequent reads at Block Base 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 4. Block Lock Status

Item	Address	Data
Block Lock Configuration	Block Base Address +002	LOCK
• Block Is Unlocked		$DQ_0 = 0$
• Block Is Locked		$DQ_0 = 1$
• Block Is Locked-Down		$DQ_1 = 1$

3.2.6 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.

3.2.7 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.

3.2.8 $V_{PP} \leq V_{PPLK}$ for Complete Protection

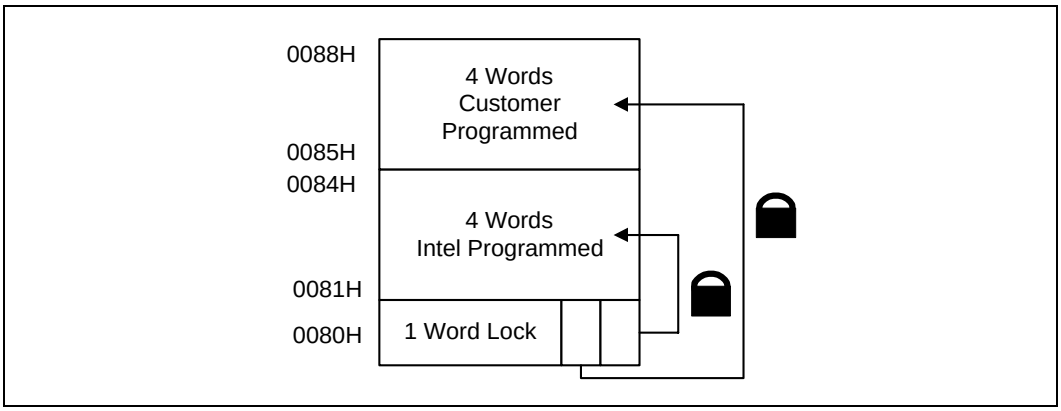
The V_{PP} programming voltage can be held low for complete write protection of all blocks in the flash device. When V_{PP} is below V_{PPLK}, any block erase or program operation will result in a error, prompting the corresponding status register bit (SR.3) to be set.

3.3 128-Bit Protection Register

1.8 Volt Dual-Plane Flash memory includes a 128-bit protection register than can be used to enhance the security of a system design. For example, the number contained in the protection register can be used to match 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-bit protection register is divided into two 64-bit segments (Figure 4, “Protection Register Memory Map” on page 14). The Intel segment is programmed at the Intel factory with a unique 64-bit number, which is not changeable. The customer segment is blank allowing customers to program as desired. Once the customer segment is programmed, it can be locked to prevent reprogramming.

Figure 4. Protection Register Memory Map



3.3.1 Reading the Protection Register

The protection register is read by using the Read Device Identifier command (90H). Once in this mode, read cycles from addresses shown in Appendix B retrieve the specified protection register information. To return to read array mode, use the Read Array command (FFH).

3.3.2 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. First write the Protection Program Setup command, C0H. The next write to the device will latch in address and data to program the specified location. The allowable addresses are shown in Appendix B. See Figure 12, “Protection Register Programming Flowchart” on page 30.

Any attempt to address Protection Program commands outside the defined protection register address space should not be performed. 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 = 1).

3.3.3 Locking the Protection Register

The customer-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.

Table 5. Command Definitions(1)

	Command	Bus Cycles	Notes	First Bus Cycle			Second Bus Cycle		
				Oper	Addr(2)	Data(3)	Oper	Addr(2	Data(3)
READ	Read Array/Reset	1		Write	PnA	FFH			
	Read Device Identification Codes	≥ 2	4,5	Write	CA	90H	Read	CA	CD
	Read Query	≥ 2	4,5	Write	QA	98H	Read	QA	QD
	Read Status Register	2		Write	BA	70H	Read	BA	SRD
	Clear Status Register	1		Write	BA	50H			
PROGRAM / ERASE	Block Erase	2	7	Write	BA	20H	Write	BA	D0H
	Program	2	6, 7	Write	WA	40H/ 10H	Write	WA	WD
	Program/Erase Suspend	1	6	Write	BA	B0H			
	Program/Erase Resume	1	6	Write	BA	D0H			
LOCK	Lock Block	2		Write	BA	60H	Write	BA	01H
	Unlock Block	2		Write	BA	60H	Write	BA	D0H
	Lock-down Block	2		Write	BA	60H	Write	BA	2FH
CONFIGURATION	Protection Program	2		Write	PA	C0H	Write	PA	PD
	Lock Protection Program	2		Write	LPA	C0H	Write	LPA	FFFDH
	Set Read Configuration Register	2	2, 5	Write	RCD	60H	Write	RCD	03H

NOTE:

1. Commands other than those shown above are reserved by Intel for future device implementations and should not be used.
2. First cycle command addresses should be the same as the operation's target address. Examples: the first-cycle address for the Read Device Identification Codes command should be the same as the Identification Code address (IA); the first cycle address for the Program command should be the same as the word address (WA) to be programmed; the first cycle address for the Erase/Program Suspend command should be the same as the address within the block to be suspended; etc.
   CA = Identification code address.
   BA = Address within the block.
   LPA = Lock Protection Address is obtained from the CFI (via the Read Query command). 1.8 V Dual-Plane Flash memory's LPA is at 0080h.
   PA = User programmable 4-word protection address in the device identification plane.
   PnA = Address within the partition.
   QA = Query code address.
   WA = Word address of memory location to be written.
3. SRD = Data read from the status register.
   WD = Data to be written at location WA is latched on the rising edge of WE# or CE# (whichever goes high first).
   CD = Identifier code data.
   PD =User programmable 4-word protection data.
   QD = Query code data.
   RCD = Read Configuration register code data presented on device addresses A_{15-0}. Upper address bits can select either partition. See Table 8 for read configuration register bits descriptions.
4. Following the Read Device Identification Codes or Read Query commands, read operations output manufacturer and device configuration or CFI query information and the read configuration register.
5. Read Device Identification and Read Query addresses must be within the bottom partition.
6. Following a block erase, program, and suspend operation, read operations access the status register.
7. The WSM recognizes either 40H or 10H program setup commands.

4.0 Command Definitions

Device operations are selected by writing specific commands into a partition's CUI. Since commands are partition-specific, it's important to write commands within the target partition's address range (see Table 5 on page 15).

4.1 Read Array Command

Upon initial device power-up or after reset, both partitions default to read array mode and to the asynchronous read configuration power-up state. The Read Array command places the addressed partition into read array mode. Once the WSM starts a block erase or program on a partition, it will not recognize the Read Array command until the WSM completes its operation or until the WSM is suspended by an Erase or Program Suspend command. However, a Read Array command in the other partition will be accepted.

4.2 Read Device Identification Command

The read device identification mode is initiated by writing the Read Device Identification command to the bottom partition. The top partition's mode is not affected by this operation. See Table 6, for device identifier code values.

Table 6. Identifier Codes

Code			Address	Data
Manufacturer Code			00000	0089
Device Code	32 Mbit	-T	00001	88D2
	32 Mbit	-B	00001	88D3
Block Lock Configuration • Block Is Unlocked • Block Is Locked • Block Is Locked-Down			Block Address +002	Lock DQ_0 = 0 DQ_0 = 1 DQ_1 = 1
Read Configuration Register (1)			00005	RCR(2)
Protection Register Lock			0080	PR-LK(3)
Protection Register			0081-0088	PR(4)

NOTE:
1. Sampled, not 100% tested.
2. RCR = Read Configuration Register
3. PR-LK = Protection Register Lock
4. PR = Protection Register

4.3 Read Query Command

The Read Query command is available only in the bottom partition and puts that partition into the read query mode. Partition reads will output Common Flash Interface (CFI) information.

4.4 Read Status Register Command

A partition's status register can be read at any time by writing the Read Status Register command to the partition's CUI. Subsequent single transfer read operations to that partition will output its status register data until another valid command is written. This operation does not affect the other partition's mode. See Table 7 for status register bit definitions.

Table 7. Status Register Definition

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

	NOTES:
SR.7 = WRITE STATE MACHINE STATUS (WSMS) 1 = Ready 0 = Busy	Check SR.7 to determine block erase or program completion. SR.6–0 are invalid while SR.7 = "0."
SR.6 = ERASE SUSPEND STATUS (ESS) 1 = Block Erase Suspended 0 = Block Erase in Progress/Completed	When an Erase Suspend command is issued, the WSM halts execution and sets both SR.7 and SR.6 to "1." SR.6 remains set until an Erase Resume command is written to the CUI.
SR.5 = ERASE STATUS (ES) 1 = Error in Block Erasure 0 = Successful Block Erase	If both SR.5 and SR.4 are "1"s after a block erase or lock block attempt, an improper command sequence was entered.
SR.4 = PROGRAM STATUS (PS) 1 = Error in Program 0 = Successful Program	
SR.3 = V_{PP} STATUS (VPPS) 1 = V_{PP} Low Detect, Operation Abort 0 = V_{PP} OK	SR.3 does not provide a continuous V_{PP} feedback. The WSM interrogates and indicates the V_{PP} level only after a block erase or program operation. SR.3 is not guaranteed to report accurate feedback when $V_{PP} \neq V_{PP1/2}$ or V_{PPLK}.
SR.2 = PROGRAM SUSPEND STATUS (PSS) 1 = Program Suspended 0 = Program in Progress/Completed	When a Program Suspend command is issued, the WSM halts execution and sets both SR.7 and SR.2 to "1." SR.2 remains set until a Program Resume command is written to the CUI.
SR.1 = DEVICE PROTECT STATUS (DPS) 1 = Block Erase or Program Attempted on a Locked Block, Operation Abort 0 = Unlocked	If a block erase or program operation is attempted to a locked block, SR.1 is set by the WSM and aborts the operation if WP# = V_{IL}.
SR.0 = RESERVED FOR FUTURE ENHANCEMENTS (R)	SR.0 is reserved for future use and should be masked out when polling the status register.

4.5 Clear Status Register Command

Status register bits SR.5, SR.4, SR.3, and SR.1 are set to "1"s by the WSM and can only be cleared by issuing the Clear Status Register command. These bits indicate various error conditions. By allowing system software to reset these bits, several operations may be performed (such as cumulatively erasing or writing several bytes in sequence). The status register may be polled to determine if a problem occurred during the sequence. The Clear Status Register command functions independently of the applied V_{PP} voltage. After executing this command, the device returns to read array mode. The Clear Status Register command clears only the status register of the addressed partition.

4.6 Block Erase Command

The two-cycle Block Erase command initiates one block erase at the addressed block within the selected partition. After writing the command, the device automatically outputs status register data when any address within the partition is read. The CPU can detect block erase completion by analyzing the partition's status register bit SR.7. The partition will remain in status register read mode until another command is written to its CUI. Only one partition can be in an erase mode at a time; the other partition must be in one of the read modes.

4.7 Program Command

A two-cycle command sequence written to the target partition initiates a program operation. Only one partition can be in program mode at a time; the other partition must be in one of the read modes.

Program setup (standard 40H or alternate 10H) is written, followed by a second write that specifies the address and data. The WSM then takes over, controlling the internal program algorithm. After the program sequence is written, the device automatically outputs status register data when read (see Figure 8, "Automated Program Flowchart" on page 26). The CPU can detect the completion of the program event by analyzing status register bit SR.7. When the program operation completes, check status register bit SR.4 for an error flag ("1"). If an error is detected, check status register bits SR.4, SR.3, and SR.1 to understand what caused the problem.

The status register of the partition being programmed can be examined by addressing any block address. After examining the status register, it should be cleared if an error was detected before issuing a new command. The partition remains in status register read mode until another command is written to the CUI.

4.8 Block Erase Suspend/Resume Command

The Block Erase Suspend command allows block erase interruption to read or program data in another block within the target partition. Once the block erase process starts, writing the Block Erase Suspend command requests that the WSM suspend the block erase operation after a certain latency period. The device continues to output status register data when read after the Block Erase Suspend command is issued. Status Register bits SR.7 and SR.6 indicate when the block erase operation has been suspended (both will be set to "1"). Specification t_{WHRH2} defines the block erase suspend latency.

At this point, a Read Array command can be written to read data from blocks other than that which is suspended. A Program command sequence can also be issued during erase suspend to program data in other blocks. Using the Program Suspend command (see Section 4.9), a program operation can be suspended during an erase suspend. The only other valid commands while block erase is suspended are Read Status Register, Block Erase Resume, Lock Block, Unlock Block, Lock Down Block and Set Read Configuration Register.

During a block erase suspend, the chip can go into a pseudo-standby mode by taking CE# to V_{IH}, which reduces active current draw. V_{PP} must remain at $V_{PP1/2}$ while block erase is suspended. WP# must also remain at V_{IL} or V_{IH}.

To resume the block erase operation, write the Block Erase Resume command to the CUI. This will automatically clear status register bits SR.6 and SR.7. After the Erase Resume command is written, the device automatically outputs status register data when read (seeFigure 9, “Block Erase Suspend/Resume Flowchart” on page 27). Block erase cannot resume until program operations initiated during block erase suspend have completed.

It is also possible to nest suspends as follows: Suspend erase in the first partition, start programming in the second partition, suspend programming in the second partition and then read from the second partition.

4.9 Program Suspend/Resume Command

The Program Suspend command allows program interruption to read data in other flash memory locations within the target partition.

Once the program process starts, writing the Program Suspend command requests that the WSM suspend the program operation after a certain latency period. The device continues to output status register data when read after issuing Program Suspend command. Status register bits SR.7 and SR.2 indicate when the program operation has been suspended (both will be set to “1”). Specification t_{WHRH1} defines the program suspend latency.

At this point, a Read Array command can be written to read data from locations other than that which is suspended. The only other valid commands while block erase is suspended are Read Status Register, Program Resume, Read Query and Read Device Identification.

During a program suspend, the chip can go into a pseudo-standby mode by taking CE# to V_{IH}, which reduces active current draw. V_{PP} and WP# must remain unchanged.

To resume the program, write the Program Resume command to the CUI. This will automatically clear status register bits SR.7 and SR.2. After the Program Resume command is written, the device automatically outputs status register data when read (see Figure 10, “Program Suspend/Resume Flowchart” on page 28).

It is also possible to nest suspends as follows: Suspend erase in the first partition, start programming in the second partition, suspend programming in the second partition and then read from the second partition.

Table 8. Read Configuration Register Definition

RM	R	FC2	FC1	FC0	R	DOC	WC
15	14	13	12	11	10	9	8
BS	CC	R	R	BW	BL2	BL1	BL0
7	6	5	4	3	2	1	0

	NOTES:
RCR.15 = READ MODE (RM) 0 = Synchronous Burst Reads Enabled 1 = Asynchronous Reads Enabled (Default)	Read mode configuration affects reads from main blocks. Parameter block, status register, and configuration reads support single read cycles.
RCR.14 = RESERVED FOR FUTURE ENHANCEMENTS (R)	This bit is reserved for future use. Set reserved bits to “0.”
RCR.13–11 = FREQUENCY CONFIGURATION (FC2-0) 000 = Code 0 reserved for future use 001 = Code 1 reserved for future use 010 = Code 2 011 = Code 3 100 = Code 4 101 = Code 5 reserved for future use 110 = Code 6 reserved for future use 111 = Code 7 reserved for future use (Default)	See Section 4.10.2 for information about the frequency configuration and its effect on the initial read. Undocumented combinations of bits RCR.14–11 are reserved by Intel Corporation for future implementations and should not be used.
RCR.10 = RESERVED FOR FUTURE ENHANCEMENTS (R)	This bit is reserved for future use. Set reserved bits to “0.”
RCR.9 = DATA OUTPUT CONFIGURATION (DOC) 0 = Hold Data for One Clock 1 = Reserved for future use (Default)	Undocumented combinations of bits RCR.10–9 are reserved by Intel Corporation for future implementations and should not be used.
RCR.8 = WAIT CONFIGURATION (WC) 0 = WAIT# Asserted During Delay 1 = WAIT# Asserted One Data Cycle Before Delay (Default)	
RCR.7 = BURST SEQUENCE (BS) 0 = Intel Burst Order 1 = Linear Burst Order (Default)	
RCR.6 = CLOCK CONFIGURATION (CC) 0 = Burst Starts and Data Output on Falling Clock Edge 1 = Burst Starts and Data Output on Rising Clock Edge (Default)	
RCR.5-4 = RESERVED FOR FUTURE ENHANCEMENTS (R)	These bits are reserved for future use. Set reserved bits to “0.”
RCR.3 = BURST WRAP (BW) 0 = Wrap bursts within burst length set by RCR.2-0 1 = Don’t wrap accesses within burst length set by RCR.2-0.(Default)	See Section 4.10.7 for information about the BURST WRAP configuration.
RCR.2–0 = BURST LENGTH (BL2–0) 001 = 4 Word Burst 010 = 8 Word Burst 011 = Reserved for future use 111 = Continuous (Linear) Burst (Default)	In the asynchronous page mode, the burst length always equals four words.

4.10 Set Read Configuration Command

The Set Read Configuration command writes data to the Read Configuration register (RCR).

This operation is initiated by a two-cycle command sequence. The RCR can be configured by writing the command at any device address. Read configuration setup is written followed by a second write that specifies the data to be written to the read configuration register. This data is placed on the address bus, $A_{15:0}$, and is latched on the rising edge of ADV#, CE#, or WE# (whichever occurs first). The read configuration data sets the device's read configuration, burst order, frequency configuration, and burst length. The command functions independently of the applied V_{PP} voltage. After executing this command, the device returns to read array mode.

Note: 1. The RCR can be read via the Read Device Identification command (90H). Address 00005 contains the RCR data. See Table 6, "Identifier Codes" on page 16.

2. All the bits in the RCR are set to "1" on device power-up or reset.

4.10.1 Device Read Configuration

Each partition supports a high performance synchronous burst mode read configuration. A read configuration register bit sets the read configuration. The RCR can be read via the Read Device Identification command (90H) at address 00005.

The main partition contains only main blocks and supports asynchronous, page mode, and synchronous read configurations. Its status register supports only single asynchronous and single synchronous reads.

The parameter partition's parameter blocks and status register support only single asynchronous and single synchronous read operations. Its main blocks support asynchronous, page mode, and synchronous read configurations.

4.10.2 Frequency Configuration

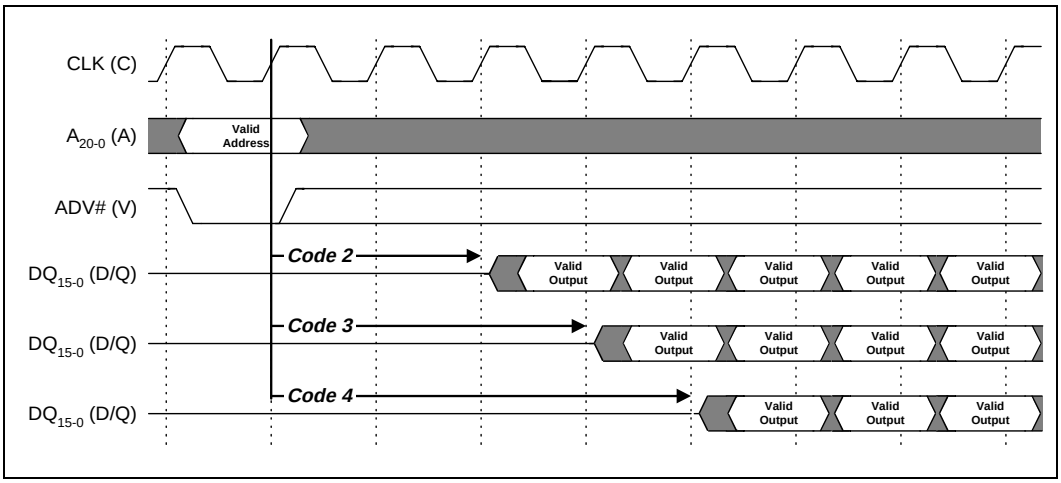
The frequency configuration informs the device of the number of clocks that must elapse after ADV# is driven active before data will be available. This value is determined by the input clock frequency. See Table 9 for the specific input CLK frequency configuration code.

Table 9. Frequency Configuration Settings

Frequency Configuration Code	Input CLK Frequency (V_{CC} = 1.65 V–1.95 V)	
	-110 ns	-120 ns
1	Reserved	Reserved
2	≤ 24 MHz	≤ 21 MHz
3	≤ 36 MHz	≤ 32 MHz
4	≤ 40 MHz	≤ 40 MHz

Figure 5, "Frequency Configuration" on page 22, illustrates data output latency from ADV# going active for different frequency configuration codes.

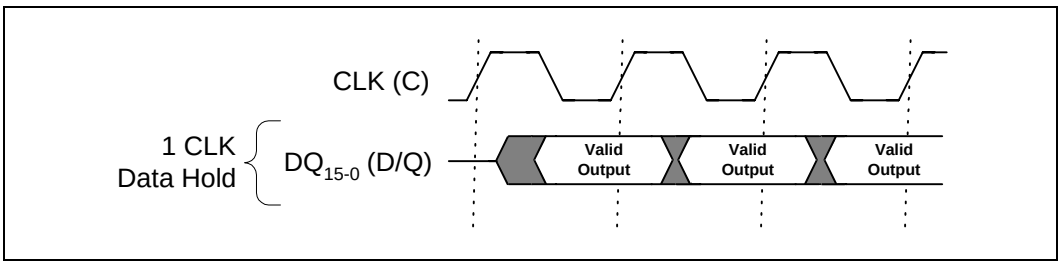
Figure 5. Frequency Configuration



4.10.3 Data Output Configuration

The output configuration determines the number of clocks that data will be held valid. The data hold time for the 1.8 V Dual-Plane Flash memory is one clock.

Figure 6. Output Configuration



4.10.4 WAIT# Configuration

The WAIT# configuration bit controls the behavior of the WAIT# output signal. This output signal can be set to be asserted during or one CLK cycle before an output delay when continuous burst length is enabled. Its setting will depend on the system and CPU characteristic. WAIT# can also be asserted in the 4- or 8-word burst length when RCR.3 = 1 (no-wrap mode) if the no-wrap read crosses the first 16-word boundary.

4.10.5 Burst Sequence

The burst sequence specifies the order in which data is addressed in synchronous burst mode. This order is programmable as either linear or Intel burst order. The continuous burst length only supports linear burst order. The order chosen will depend on the CPU characteristic. See Table 10, “Sequence and Burst Length” on page 23 for more details.

Table 10. Sequence and Burst Length

Starting Address	Wrap	No-Wrap(1)	Burst Addressing Sequence (Dec)				
			4-Word Burst Length (RCR.2-0 = 001)		8-Word Burst Length (RCR.2-0 = 010)		Continuous Burst (RCR.2-0 = 111)
(Dec)	RCR.3 =	RCR.3 =	Linear	Intel	Linear	Intel	Linear
0	0		0-1-2-3	0-1-2-3	0-1-2-3-4-5-6-7	0-1-2-3-4-5-6-7	0-1-2-3-4-5-6-...
1	0		1-2-3-0	1-0-3-2	1-2-3-4-5-6-7-0	1-0-3-2-5-4-7-6	1-2-3-4-5-6-7-...
2	0		2-3-0-1	2-3-0-1	2-3-4-5-6-7-0-1	2-3-0-1-6-7-4-5	2-3-4-5-6-7-8-...
3	0		3-0-1-2	3-2-1-0	3-4-5-6-7-0-1-2	3-2-1-0-7-6-5-4	3-4-5-6-7-8-9-...
4	0				4-5-6-7-0-1-2-3	4-5-6-7-0-1-2-3-	4-5-6-7-8-9-10...
5	0				5-6-7-0-1-2-3-4	5-4-7-6-1-0-3-2	5-6-7-8-9-10-11...
6	0				6-7-0-1-2-3-4-5	6-7-4-5-2-3-0-1	6-7-8-9-10-11-12-...
7	0				7-0-1-2-3-4-5-6	7-6-5-4-3-2-1-0	7-8-9-10-11-12-13...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
14	0						14-15-16-17-18-19-20-...
15	0						15-16-17-18-19-20-21-...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0		1	0-1-2-3	NA	0-1-2-3-4-5-6-7	NA	0-1-2-3-4-5-6-...
1		1	1-2-3-4	NA	1-2-3-4-5-6-7-8	NA	1-2-3-4-5-6-7-...
2		1	2-3-4-5	NA	2-3-4-5-6-7-8-9	NA	2-3-4-5-6-7-8-...
3		1	3-4-5-6	NA	3-4-5-6-7-8-9-10	NA	3-4-5-6-7-8-9-...
4		1			4-5-6-7-8-9-10-11	NA	4-5-6-7-8-9-10...
5		1			5-6-7-8-9-10-11-12	NA	5-6-7-8-9-10-11...
6		1			6-7-8-9-10-11-12-13	NA	6-7-8-9-10-11-12-...
7		1			7-8-9-10-11-12-13-14	NA	7-8-9-10-11-12-13...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
14		1					14-15-16-17-18-19-20-...
15		1					15-16-17-18-19-20-21-...

NOTE:
1. The burst wrap bit (RCR.3) determines whether 4- or 8-word burst-accesses wrap within the burst-length boundary or whether they cross word-length boundaries to perform linear accesses. In the no-wrap mode (RCR.3 = 1), the device operates similar to continuous linear burst mode but consumes less power during 4- and 8-word bursts.

4.10.6 Clock Configuration

The clock configuration bit configures the device to start a burst cycle, output data, and assert WAIT# on the rising or falling edge of the clock. This CLK flexibility enables interfacing the 1.8 Volt Dual-Plane Flash memory to a wide range of burst CPUs.

4.10.7 Burst Wrap

The burst wrap bit determines whether 4- or 8-word burst-accesses wrap within the burst-length boundary or whether they cross word-length boundaries to perform linear accesses. No-wrap mode (RCR.3 = 1) enables WAIT# to hold off the system processor, as it does in the continuous burst mode. In the no-wrap mode, the device operates similar to continuous linear burst mode but consumes less power during 4- and 8-word bursts.

For example, if RCR.3 = 0 (wrap mode) and RCR.2-0 = 001 (4-word burst length), then possible linear burst sequences are 0-1-2-3, 1-2-3-0, 2-3-0-1, and 3-0-1-2.

If RCR.3 = 1 (no-wrap mode) and RCR.2-0 = 001 (4-word burst length), then possible linear burst sequences are 0-1-2-3, 1-2-3-4, 2-3-4-5, and 3-4-5-6. RCR.3 = 1 not only enables limited non-aligned sequential burst, but also reduces power by minimizing the number of internal read operations.

The above 4-word burst sequences can also be achieved by setting RCR.2-0 bits for continuous linear burst mode (111). However, significantly more power may be consumed. The 1-2-3-4 sequence, for example, will consume power during the initial access, again during the internal pipeline lookup as the processor reads word 2, and possibly again, depending on system timing, near the end of the sequence as the device pipelines the next 4-word sequence. RCR.3 = 1 mode reduces this excess power consumption.

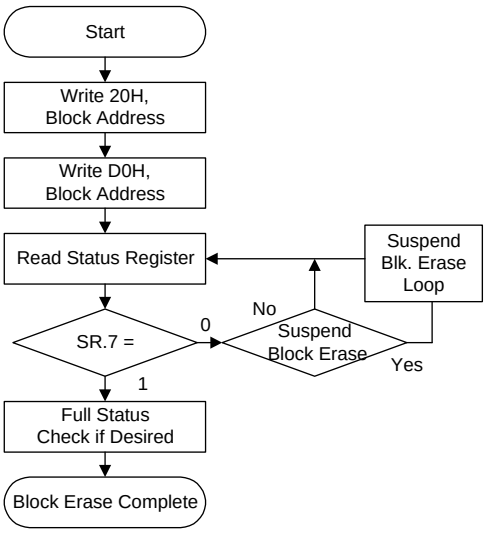
4.10.8 Burst Length

The burst length is the number of words that the device will output. The device supports burst lengths of four and eight words. It also supports a continuous burst mode. In continuous burst mode, the device will linearly output data until the internal burst counter reaches the end of the device’s burst-able address space or a partition boundary. Bits RCR.2–0 in the read configuration register set the burst length.

4.10.8.1 Continuous Burst Length

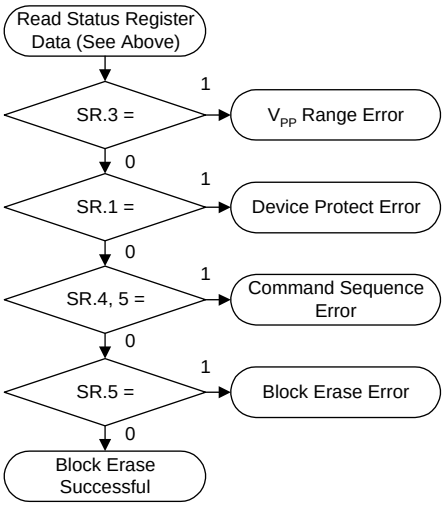
When operating in the continuous burst mode or 4-, 8-word burst with burst wrap bit set (RCR.3 = 1), the flash memory may incur an output delay when the burst sequence crosses the first sixteen word boundary. The starting address dictates whether or not a delay will occur. If the starting address is aligned to a four word boundary, the delay will not be seen. If the starting address is the end of a four word boundary, the output delay will be equal to the frequency configuration setting; this is the worst case delay. The delay will only take place once during a continuous burst access. If the burst sequence never crosses a sixteen word boundary, the delay will never happen. The flash memory uses the WAIT# output pin in the continuous burst configuration to inform the system if this output delay occurs.

Figure 7. Automated Block Erase 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 Addr = Within Block to Be Erased
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 operation to place device in read array mode.		

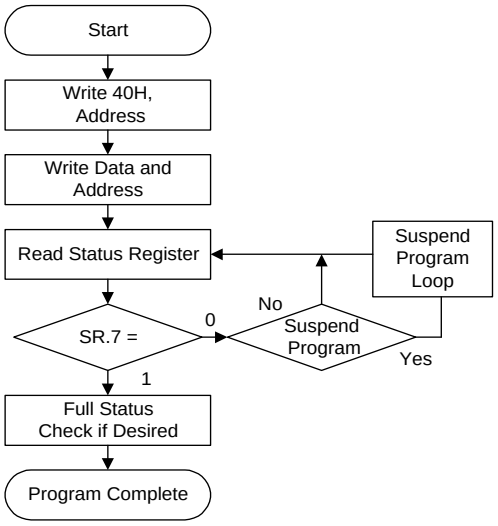
FULL STATUS CHECK PROCEDURE



Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Error Detect
Standby		Check SR.1 1 = Device Protect Detect
Standby		Check SR.4, 5 Both 1 = Command Sequence Error
Standby		Check SR.5 1 = Block Erase Error
SR.5, SR.4, SR.3 and SR.1 are only cleared by the Clear Staus Register command, in cases where multiple blocks are erased before full status is checked. If an error is detected, clear the status register before attempting retry or other error recovery.		

0672_07

Figure 8. Automated Program Flowchart



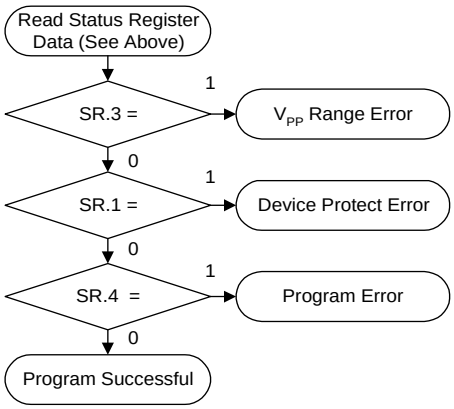
Bus Operation	Command	Comments
Write	Program Setup	Data = 40H Addr = Location to Be Written
Write	Data	Data = Data to Be Written Addr = Location to Be Written
Read		Status Register Data Addr = Location to Be Written
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy

Repeat for subsequent byte writes.

SR full status check can be done after each word write or after a sequence of program operations.

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

FULL STATUS CHECK PROCEDURE



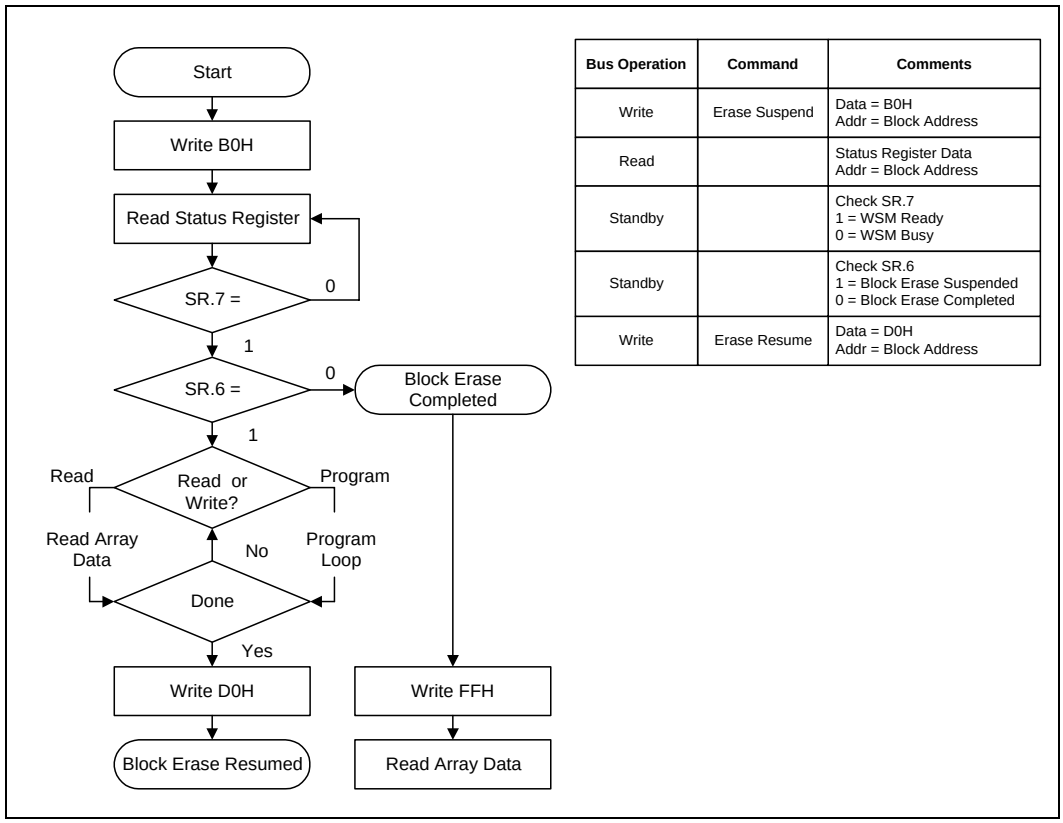
Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Error Detect
Standby		Check SR.1 1 = Device Protect Detect
Standby		Check SR.4 1 = Program Error

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

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

0672_08

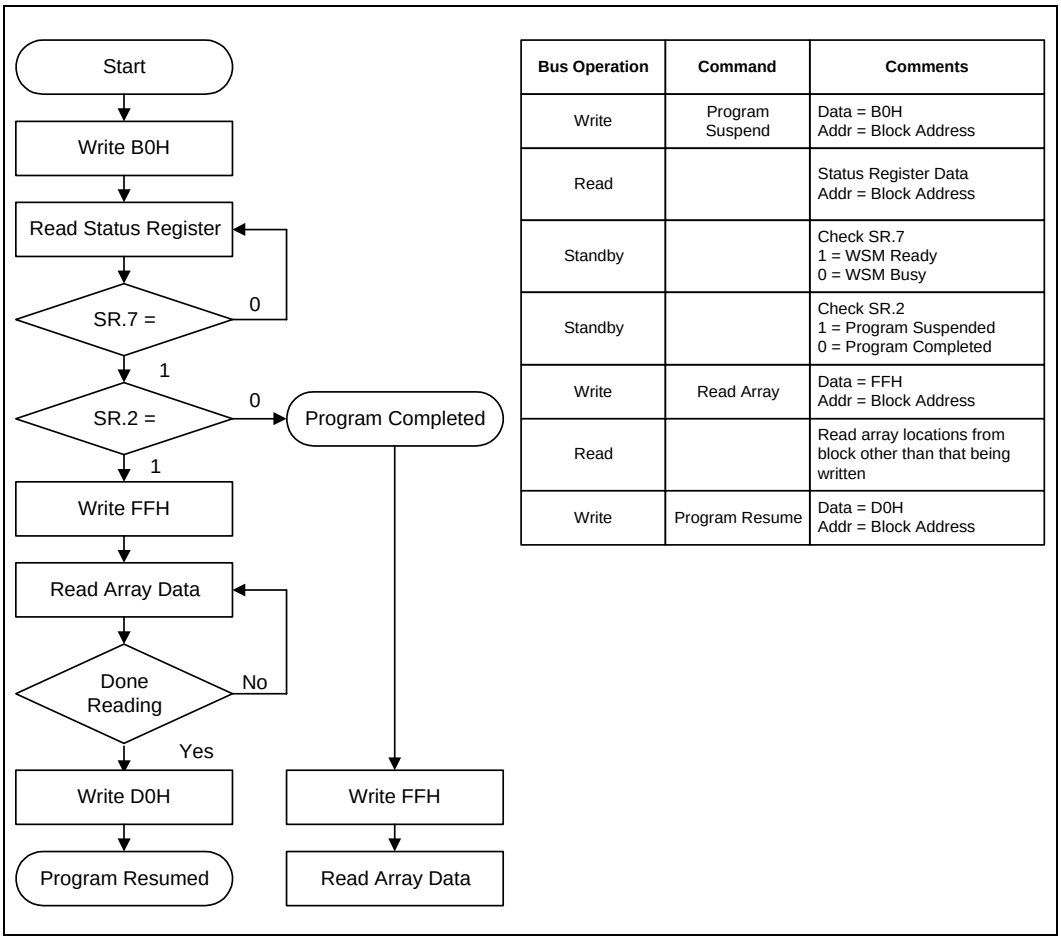
Figure 9. Block Erase Suspend/Resume Flowchart



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

0672_09

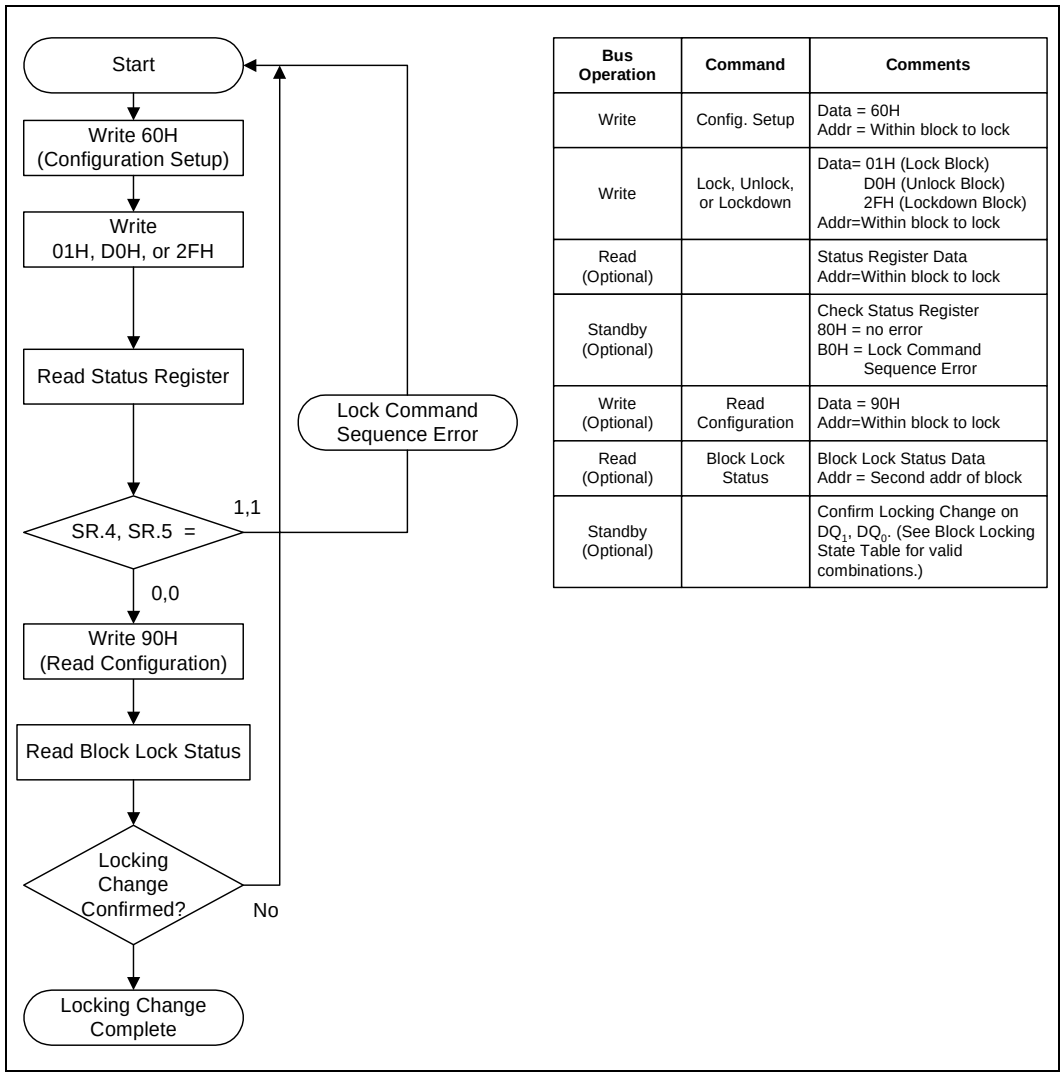
Figure 10. Program Suspend/Resume Flowchart



Bus Operation	Command	Comments
Write	Program Suspend	Data = B0H Addr = Block Address
Read		Status Register Data Addr = Block Address
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 = Block Address
Read		Read array locations from block other than that being written
Write	Program Resume	Data = D0H Addr = Block Address

0672_10

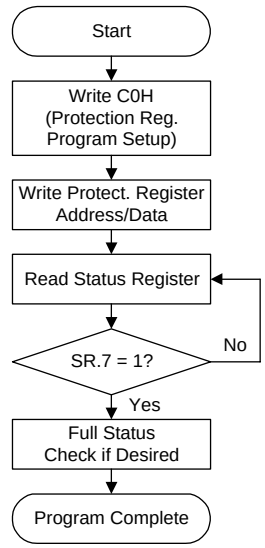
Figure 11. Locking Operations Flowchart



Bus Operation	Command	Comments
Write	Config. Setup	Data = 60H Addr = Within block to lock
Write	Lock, Unlock, or Lockdown	Data= 01H (Lock Block) D0H (Unlock Block) 2FH (Lockdown Block) Addr=Within block to lock
Read (Optional)		Status Register Data Addr=Within block to lock
Standby (Optional)		Check Status Register 80H = no error B0H = Lock Command Sequence Error
Write (Optional)	Read Configuration	Data = 90H Addr=Within block to lock
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.)

0672_11

Figure 12. Protection Register Programming 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

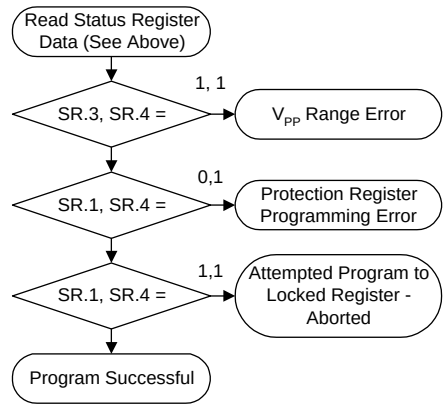
Protection Program operations can only be addressed within the protection register address space. Addresses outside the defined space will return an error.

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: Operation 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.

0672_12

5.0 Data Protection

The 1.8 Volt Dual-Plane Flash memory architecture features dynamic hardware block-locking so critical code can be kept secure while non-locked blocks are programmed or erased.

5.1 $V_{PP} \leq V_{PPLK}$ for Complete Protection

The V_{PP} programming voltage can be held low for complete write protection of all flash device blocks. When V_{PP} is below V_{PPLK}, block erase or program operations result in an error in the corresponding partition's status register; bit (SR.3) is set.

5.2 WP# = V_{IL} for Block Lock Down

Locked down blocks are securely or permanently locked down when WP# = V_{IL}; any block erase or program operation to a locked-down block will result in an error, which will be reflected in the status register.

6.0 Program and Erase Voltages

Intel 1.8 Volt Dual-Plane Flash memory provides in-system programming and erase in the 1.8 V range. For fast production programming, 1.8 Volt Dual-Plane Flash memory includes a low-cost, backward-compatible high-performance Improved-12 V programming feature.

When V_{PP} is between V_{PP1} min and V_{PP1} max, all program and erase current is drawn through the V_{CC} pin. Note that if V_{PP} is driven by a logic signal, V_{PP} must remain above V_{PP1} min 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 V_{PP} voltage.

Figure 13, "Example Power Supply Configurations" on page 34, shows examples of how the flash power supplies can be configured for various usage models.

6.1 Improved-12 V Programming Operation for Fast Manufacturing

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.

7.0 Design Considerations

This section will describe how to use and design with the 1.8 Volt Dual-Plane Flash memory. It will focus on the dual partition architecture as well as the integrated features of the device.

Today's high-performance CPUs and ASICs designed for portable and handheld applications place relentless demands on memory for increased data transfer speeds, as well as very low power operation. This requires a new memory approach to help bridge the performance gap between the processor and memory. 1.8 Volt Dual-Plane Flash memory satisfies both of these requirements by operating at 1.8 volts and also providing hardware simultaneous read-while-program/erase capabilities through its dual partition architecture. It also supports two high-performance interfaces (asynchronous page mode and synchronous burst mode at 40 MHz max) with zero wait states.

This section will cover these new features and how to implement them in designs using 1.8 Volt Dual-Plane Flash memory. The following is a list of the key topics that will be covered:

- Flash Hardware Design Considerations.
- Flash Software Design Considerations.
- System Design Considerations.
- Design Tools and Software

For detailed device specifications and more information, refer to Section 10.0 for a full list of companion documents.

7.1 Flash Hardware Design Considerations

7.1.1 Flash Power Consumption

While in operation, the flash device consumes active power. Intel® Flash devices have power saving features, Automatic Power Savings (APS) and standby modes that reduce overall memory and system power consumption.

7.1.1.1 Active Power

With CE# at a logic-low level and RST# at a logic-high level, the device is in active mode. Only one partition at a time is active if both partitions are in read mode. However, both partitions can be active simultaneously if one is in read mode and the other is performing background program or erase. The active "read" partition is selected when CE# is low and a valid partition address is present. See Table 2 on page 9, for simultaneous commands allowed with dual partitions.

7.1.1.2 Using No-Wrap Mode

The burst wrap bit (RCR.3) of the Read Configuration Register determines whether 4- or 8-word burst-accesses wrap within the burst-length boundary or whether they cross word-length boundaries to perform linear accesses. No-wrap mode (RCR.3 = 1) enables WAIT# to hold off the system processor, as it does in the continuous burst mode. In the no-wrap mode, the device operates similar to continuous linear burst mode but consumes less power during 4- and 8-word bursts. Set RCR.3 = 1 for lower power operation and non-wrapped linear bursts.

For example, if RCR.3 = 0 (wrap mode) and RCR.2-0 = 001 (4-word burst length), then possible linear burst sequences are 0-1-2-3, 1-2-3-0, 2-3-0-1, and 3-0-1-2.

If RCR.3 = 1 (no-wrap mode) and RCR.2-0 = 001 (4-word burst length), then possible linear burst sequences are 0-1-2-3, 1-2-3-4, 2-3-4-5, and 3-4-5-6. RCR.3 = 1 not only enables limited non-aligned sequential burst, but also reduces power by minimizing the number of internal read operations.

The above 4-word burst sequences can also be achieved by setting RCR.2-0 bits for continuous linear burst mode (111). However, significantly more power may be consumed. The 1-2-3-4 sequence, for example, will consume power during the initial access, again during the internal pipeline lookup as the processor reads word 2, and possibly again, depending on system timing, near the end of the sequence as the device pipelines the next 4-word sequence. RCR.3 = 1 (no-wrap mode) mode reduces this excess power consumption.

7.1.1.3 Automatic Power Savings

Automatic Power Savings (APS) provides low-power operation during active mode, allowing the flash to put itself into a low current state when not being accessed. After data is read from the memory array, the device's power consumption enters the APS mode where typical I_{CC} current is comparable to I_{CCS}. The flash memory stays in this static state with outputs valid until a new location is read.

7.1.1.4 Standby Power

With CE# at a logic-high level (V_{IH}) and both partitions are in read mode, the flash memory is in standby mode. Outputs (DQ_0–DQ_{15}) are placed in high-impedance state independent of the OE# signal's state. If CE# transitions to a logic-high during erase or program operations, the device continues the operation, consuming corresponding active power until the operation completes.

7.1.1.5 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 does not care which power supply, V_{PP}, V_{CC}, or V_{CCQ}, powers up first.

7.1.1.5.1 RST# Connection

The use of RST# 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 RST# to the system reset signal to allow proper CPU/flash initialization following system reset.

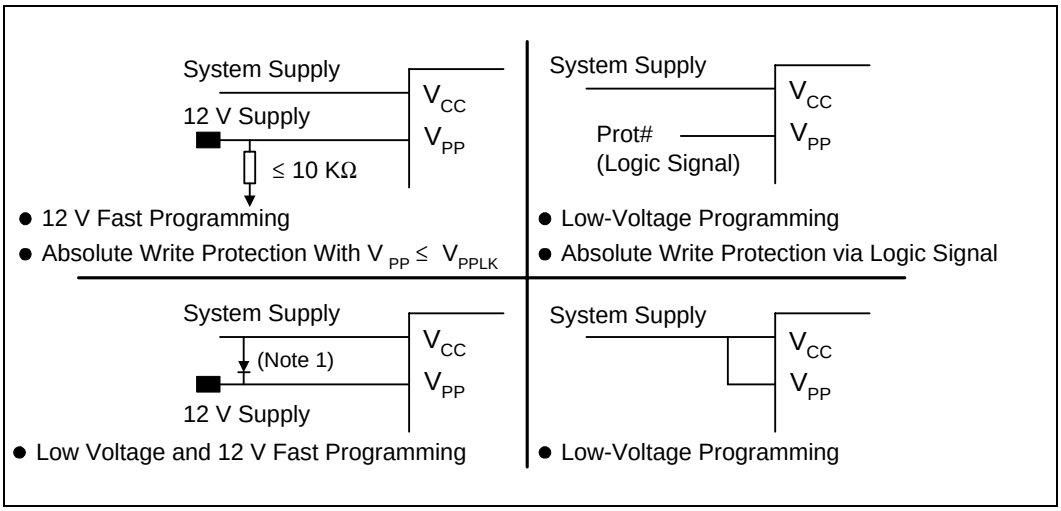
System designers must guard against spurious writes when V_{CC} voltages are above V_{LKO} and V_{PP} is active. Since both WE# and CE# must be low for a command write, driving either signal to V_{IH} will inhibit writes to the device. The CUI architecture provides additional protection since alteration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

7.1.1.5.2 V_{CC}, V_{PP}, RST# 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 deep power-down mode or after V_{CC} transitions above V_{LKO} (lockout voltage), is read array mode.

After any block erase or program 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.

Figure 13. Example Power Supply Configurations



PSU_CONF T

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

7.1.1.6 Power Supply Decoupling

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

- Standby current levels (I_{CCS})
- Active current levels (I_{CCR})
- 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 de-coupling capacitor selection will suppress these transient voltage peaks. Each flash device should have a 0.1 µF ceramic capacitor connected between each V_{CC}, V_{CCQ} and V_{SSQ}, and between its V_{PP} and V_{SS}. These high-frequency, inherently low-inductance capacitors should be placed as close as possible to the package leads.

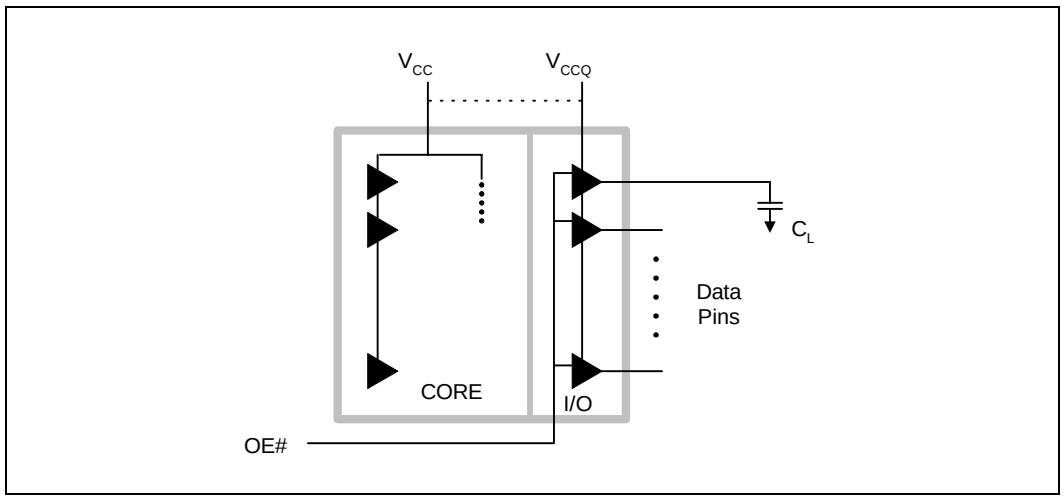
7.1.1.6.1 Circuit Board V_{PP} Trace

Designing for in-system writes to the flash memory requires special consideration of the V_{PP} power supply trace by the printed circuit board designer. The V_{PP} pin supplies the flash memory cells current for programming and erasing. V_{PP} trace widths and layout should be similar to that of V_{CC}. Adequate V_{PP} supply traces, and decoupling capacitors placed adjacent to the component, will decrease spikes and overshoots.

7.1.2 Flash Core and I/O Voltage

1.8 Volt Dual-Plane Flash memory matches a true 1.8 V EIA/JEDEC Standard from 1.65 V to 1.95 V. It can read and program down to 1.65 V. The flash device is separated into two sections, the core and the I/O (Figure 14). There are two separate power pins, V_{CC} and V_{CCQ} which provide power to the device's core, and to the I/O respectively. The separate V_{CCQ} pin can help provide noise isolation from the V_{CC} power supply when connected to a separate 1.8 Volt supply. V_{CC} must always be at the same or higher voltage than the voltage applied to V_{CCQ}, they can be connected together.

Figure 14. Flash Core and I/O Voltage Separation



The total power consumption of the device is the sum of the power consumed by the core and the power consumed by the I/Os. The total power used by the I/O pins is a function of the I/O voltage, the operating frequency, and the capacitance of the pins as shown in the following equation. C_L is the load capacitance and f is the I/O switching frequency.

$$P_{READ\_I/O} = 0.5 * C_L * f * (V_{CCQ})^2 * (\text{number of I/O pins})$$

More information on I/O power consumption can be found in applications note *AP-641 Achieving Low Power with Advanced Boot Block Flash Memory.*

For fast production programming, 1.8 Volt Dual-Plane Flash memory includes a 12 V programming feature. With 12 V connected to V_{PP}, programming time is significantly reduced, which is important for fast factory programming. When used in mobile applications where a second 12 V supply is unavailable, V_{PP} program voltage must be 1.65 V—1.95 V during program and erase cycles. Connecting V_{PP} to a 12 V supply (11.4 V—12.6 V) should only be done for a maximum of 1000 cycles on the main blocks and 2500 cycles on the parameter blocks, and should not be connected for more than 80 hours.

7.2 Flash Software Design Considerations

7.2.1 Conventions and Definitions

Throughout this section references are made to words and phrases which are explained below.

- **Plane vs. Partition**: Both of these words refer to memory areas within the flash device. A memory plane is a memory area with its own address range. The main array, status register, ID/protection register and query are different read planes. These planes may be accessed by putting the device into the desired mode via commands to the Command User Interface. The main array plane is split into two physical partitions, with a continuous address range throughout the two partitions.
- **Writing a command vs. Writing to the array**: A write is any memory cycle where WE# is asserted. It is used to get a command or data into the device. A write command is issued to change the device's mode. V_{PP} does not need to be valid during write command operations. Writing to the array, or programming, refers to storing memory into the array plane. This is done with a two-cycle write command, and V_{PP} has to be valid during the program operation.
- **Current Partition**: This is the partition to which commands are currently being written. For example, if data is being programmed into the bottom partition, this becomes the current partition and the top partition is referred to as the other partition. If after that, a program command is issued to the top partition, it then becomes the current partition and the bottom partition is now the other partition.
- **Other Partition**: The partition, which is either idle or busy, to which commands are not currently being written.
- **Current State**: The state that the Command User Interface is currently in. It can be either the current partition or the other partition.
- **Next State**: This will be the state of the flash component after it has received the command to go into this state.
- **Setup**: This refers to the current state of the either partition. Setup refers to erase, program, protection register, block lock/unlock/lock-down and Set Read Configuration Register setup.
- **Busy**: The other partition is in a busy state when it is in erase, program or protection register program mode.
- **Idle**: The other partition is in idle mode when it is not in setup, busy, or erase/program suspend mode.
- **LB** = Lock Block
- **ULB** = Unlock Block
- **OTP** = One-Time Programmable Protection Register
- **RCR** = Read Configuration Register
- **ES** = Erase Suspend
- **PS** = Program Suspend
- **BE** = Block Erase

7.2.2 Using Dual Partitions

This section describes four examples of synthesizing the flash component's next state, knowing the current state of each partition and the input. These examples will use Table 11 on page 38. In these examples, the partitions will be referred to as top partition and bottom partition. Also, each example assumes that both partitions start in read array mode. Table 11 will help with software design showing the allowable commands that one partition can accept based on the mode of the other partition.

Sheets 1 and 2 of Table 11 should be read by placing them adjacent to each other. Sheets 3 and 4 are continuations of Sheets 1 and 2, as shown by the row numbers.

Table 11. Write State Machine - Next State Table (Sheet 1 of 4)

Row Number	Current state of the Other Partition	Current State of the Current Partition: Mode	State	Data when read	SR.7	Command Input to the Current Partition (and Next State of the Current Partition): Read Array (FFH)	Program Setup (10H/40H)	Erase Setup (20H)	BE Confirm, P/E Resume, ULB Confirm (D0H)	Program/Erase Suspend (B0H)	Row Number
1	Setup	Read	Array	Array	1	Read Array					1
2	Busy										2
3	Idle					Read Array	Program Setup	Erase Setup	Read Array		3
4	Erase Suspend							Read Array			4
5	Prog. Suspend					Read Array					5
6	Setup		Query	CFI	1	Read Array					6
7	Busy										7
8	Idle					Read Array	Program Setup	Erase Setup	Read Array		8
9	Erase Suspend							Read Array			9
10	Prog. Suspend					Read Array					10
11	Setup		Device Identification	ID	1	Read Array					11
12	Busy										12
13	Idle					Read Array	Program Setup	Erase Setup	Read Array		13
14	Erase Suspend							Read Array			14
15	Prog. Suspend					Read Array					15
16	Setup		Status	Status	1	Read Array					16
17	Busy										17
18	Idle					Read Array	Program Setup	Erase Setup	Read Array		18
19	Erase Suspend							Read Array			19
20	Prog. Suspend					Read Array					20
21	Any State	Lock/RCR	Setup	Status	1	Lock/RCR Error			LB/ULB	L/RCR Error	21
22	Setup		Error	Status	1	Read Array					22
23	Busy										23
24	Idle					Read Array	Program Setup	Erase Setup	Read Array		24
25	Erase Suspend							Read Array			25
26	Prog. Suspend					Read Array					26
27	Setup		Lock/Unlock Block	Status	1	Read Array					27
28	Busy										28
29	Idle					Read Array	Program Setup	Erase Setup	Read Array		29
30	Erase Suspend							Read Array			30
31	Prog. Suspend					Read Array					31
32	Setup		Set RCR	Array	1	Read Array					32
33	Busy										33
34	Idle					Read Array	Program Setup	Erase Setup	Read Array		34
35	Erase Suspend							Read Array			35
36	Prog. Suspend					Read Array					36
37	Idle	Protection Register	Setup	Status	1	Protection Register (Busy)					37
38	Idle		Busy	Status	0	Protection Register (Busy)					38
39	Setup		Done	Status	1	Read Array					39
40	Busy										40
41	Idle					Read Array	Program Setup	Erase Setup	Read Array		41
42	Erase Suspend							Read Array			42
43	Prog. Suspend					Read Array					43

Table 11. Write State Machine - Next State Table (Sheet 2 of 4)

Row Number	Command Input to the Current Partition (and Next State of the Current Partition)									Row Number
	Read Status (70H)	Clear Status Register (50H)	Read Device ID (90H)	Read Query (98H)	Lock/Unlock, Lock-Down, Write RCR Setup (60H)	OTP Setup (C0H)	Lock Block Confirm (01H)	Lock-Down Block Confirm (2FH)	Write RCR Confirm (03H)	
1	Read Status	Read Array	Read Device ID	Read Query	Lock/RCR Setup	Read Array				1
2										2
3						OTP Setup	Read Array			3
4						Read Array				4
5										5
6	Read Status	Read Array	Read Device ID	Read Query	Lock/RCR Setup	Read Array				6
7										7
8						OTP Setup	Read Array			8
9						Read Array				9
10										10
11	Read Status	Read Array	Read Device ID	Read Query	Lock/RCR Setup	Read Array				11
12										12
13						OTP Setup	Read Array			13
14						Read Array				14
15										15
16	Read Status	Read Array	Read Device ID	Read Query	Lock/RCR Setup	Read Array				16
17										17
18						OTP Setup	Read Array			18
19						Read Array				19
20										20
21	Lock/RCR Error						LB/ULB	LB/ULB	Set RCR	21
22	Read Status	Read Array	Read Device ID	Read Query	Lock/RCR Setup	Read Array				22
23										23
24						OTP Setup	Read Array			24
25						Read Array				25
26										26
27	Read Status	Read Array	Read Device ID	Read Query	Lock/RCR Setup	Read Array				27
28										28
29						OTP Setup	Read Array			29
30						Read Array				30
31										31
32	Read Status	Read Array	Read Device ID	Read Query	Lock/RCR Setup	Read Array				32
33										33
34						OTP Setup	Read Array			34
35						Read Array				35
36										36
37	Protection Register (Busy)									37
38	Protection Register (Busy)									38
39	Read Status	Read Array	Read Device ID	Read Query	Lock/RCR Setup	Read Array				39
40										40
41						OTP Setup	Read Array			41
42						Read Array				42
43										43

Table 11. Write State Machine - Next State Table (Sheet 3 of 4)

Row Number	Current state of the Other Partition	Current State of the Current Partition: Mode	State	Data when read	SR.7	Command Input to the Current Partition (and Next State of the Current Partition): Read Array (FFH)	Program Setup (10H/40H)	Erase Setup (20H)	BE Confirm, P/E Resume, ULB Confirm (D0H)	Program/Erase Suspend (B0H)	Row Number
44	Any State	Program	Setup	Status	1	Program (Busy)					44
45	Idle	Program	Busy	Status	0	Program (Busy)				PS Read Status	45
46	Setup	Program	Done	Status	1	Read Array					46
47	Busy	Program	Done	Status	1	Read Array					47
48	Idle	Program	Done	Status	1	Read Array	Program Setup	Erase Setup	Read Array		48
49	Erase Suspend	Program	Done	Status	1	Read Array					49
50	Prog. Suspend	Program	Done	Status	1	Read Array					50
51	Setup	Program Suspend	Read Status	Status	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	51
52	Idle	Program Suspend	Read Status	Status	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	52
53	Erase Suspend	Program Suspend	Read Status	Status	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	53
54	Setup	Program Suspend	Read Array	Array	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	54
55	Idle	Program Suspend	Read Array	Array	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	55
56	Erase Suspend	Program Suspend	Read Array	Array	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	56
57	Setup	Program Suspend	Read Device ID	ID	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	57
58	Idle	Program Suspend	Read Device ID	ID	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	58
59	Erase Suspend	Program Suspend	Read Device ID	ID	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	59
60	Setup	Program Suspend	Read Query	CFI	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	60
61	Idle	Program Suspend	Read Query	CFI	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	61
62	Erase Suspend	Program Suspend	Read Query	CFI	1	Program Suspend Read Array			Program (Busy)	Program Suspend Read Array	62
63	Idle	Erase	Setup	Status	1	Erase Error			Erase (Busy)	Erase Error	63
64	Setup	Erase	Error	Status	1	Read Array					64
65	Busy	Erase	Error	Status	1	Read Array					65
66	Idle	Erase	Error	Status	1	Read Array	Program Setup	Erase Setup	Read Array		66
67	Erase Suspend	Erase	Error	Status	1	Read Array					67
68	Prog. Suspend	Erase	Error	Status	1	Read Array					68
69	Setup	Erase	Done	Status	1	Read Array					69
70	Busy	Erase	Done	Status	1	Read Array					70
71	Idle	Erase	Done	Status	1	Read Array	Program Setup	Erase Setup	Read Array		71
72	Erase Suspend	Erase	Done	Status	1	Read Array					72
73	Prog. Suspend	Erase	Done	Status	1	Read Array					73
74	Idle	Erase	Busy	Status	0	Erase (Busy)				ES Read Status	74
75	Setup	Erase Suspend	Read Status	Status	1	Erase Suspend Read Array			Erase (Busy)	ES Read Array	75
76	Busy	Erase Suspend	Read Status	Status	1	Erase Suspend Read Array					76
77	Idle	Erase Suspend	Read Status	Status	1	ES Read Array	Program Setup	ES Read Array	Erase (Busy)	ES Read Array	77
78	Prog. Suspend	Erase Suspend	Read Status	Status	1	Erase Suspend Read Array					78
79	Setup	Erase Suspend	Read Array	Array	1	Erase Suspend Read Array			Erase (Busy)	ES Read Array	79
80	Busy	Erase Suspend	Read Array	Array	1	Erase Suspend Read Array					80
81	Idle	Erase Suspend	Read Array	Array	1	ES Read Array	Program Setup	ES Read Array	Erase (Busy)	ES Read Array	81
82	Prog. Suspend	Erase Suspend	Read Array	Array	1	Erase Suspend Read Array					82
83	Setup	Erase Suspend	Read Device ID	ID	1	Erase Suspend Read Array			Erase (Busy)	ES Read Array	83
84	Busy	Erase Suspend	Read Device ID	ID	1	Erase Suspend Read Array					84
85	Idle	Erase Suspend	Read Device ID	ID	1	ES Read Array	Program Setup	ES Read Array	Erase (Busy)	ES Read Array	85
86	Prog. Suspend	Erase Suspend	Read Device ID	ID	1	Erase Suspend Read Array					86
87	Setup	Erase Suspend	Read Query	CFI	1	Erase Suspend Read Array			Erase (Busy)	ES Read Array	87
88	Busy	Erase Suspend	Read Query	CFI	1	Erase Suspend Read Array					88
89	Idle	Erase Suspend	Read Query	CFI	1	ES Read Array	Program Setup	ES Read Array	Erase (Busy)	ES Read Array	89
90	Prog. Suspend	Erase Suspend	Read Query	CFI	1	Erase Suspend Read Array					90

Table 11. Write State Machine -Next State Table (Sheet 4 of 4)

Row Number	Command Input to the Current Partition (and Next State of the Current Partition)								
	Read Status (70H)	Clear Status Register (50H)	Read Device ID (90H)	Read Query (98H)	Lock/Unlock, Lock-Down, Write RCR Setup (60H)	OTP Setup (C0H)	Lock Block Confirm (01H)	Lock-Down Block Confirm (2FH)	Write RCR Confirm (03H)
44	Program (Busy)								
45	Program (Busy)								
46	Read Status	Read Array	Read Device ID	Read Query	Lock/RCR Setup	Read Array			
47									
48						OTP Setup	Read Array		
49						Read Array			
50									
51	Program Suspend Read Status	Program Suspend Read Array	Program Suspend Read Device ID	Program Suspend Read Query	Program Suspend Read Array				
52									
53									
54	Program Suspend Read Status	Program Suspend Read Array	Program Suspend Read Device ID	Program Suspend Read Query	Program Suspend Read Array				
55									
56									
57	Program Suspend Read Status	Program Suspend Read Array	Program Suspend Read Device ID	Program Suspend Read Query	Program Suspend Read Array				
58									
59									
60	Program Suspend Read Status	Program Suspend Read Array	Program Suspend Read Device ID	Program Suspend Read Query	Program Suspend Read Array				
61									
62									
63	Erase Error								
64	Read Status	Read Array	Read Device ID	Read Query	Lock/RCR Setup	Read Array			
65									
66						OTP Setup	Read Array		
67						Read Array			
68									
69	Read Status	Read Array	Read Device ID	Read Query	Lock/RCR Setup	Read Array			
70									
71						OTP Setup	Read Array		
72						Read Array			
73									
74	Block Erase (Busy)								
75	Erase Suspend Read Status	Erase Suspend Read Array	Erase Suspend Read Device ID	Erase Suspend Read Query	Lock/RCR Setup	Erase Suspend Read Array			
76									
77									
78									
79	Erase Suspend Read Status	Erase Suspend Read Array	Erase Suspend Read Device ID	Erase Suspend Read Query	Lock/RCR Setup	Erase Suspend Read Array			
80									
81									
82									
83	Erase Suspend Read Status	Erase Suspend Read Array	Erase Suspend Read Device ID	Erase Suspend Read Query	Lock/RCR Setup	Erase Suspend Read Array			
84									
85									
86									
87	Erase Suspend Read Status	Erase Suspend Read Array	Erase Suspend Read Device ID	Erase Suspend Read Query	Lock/RCR Setup	Erase Suspend Read Array			
88									
89									
90									

7.2.2.1 Basic Status Register Read

The first example shows how to read the status register of the top partition. The current state of the partition is read array mode, and the bottom partition is idle. This is shown on the state table as row 3. When a Read Status Register command (70H) is issued, the next state becomes Read Status (Table 11, row 18). Subsequent reads from this partition will output status register data. Throughout these operations, the bottom partition stays idle.

7.2.2.2 Erase Suspend to Read

The next example will show how to suspend an erase operation in the top partition to read information from the same partition. The current state of the top partition is read array mode, and the bottom partition is idle (row 3). When an Erase Setup command (20H) is issued, the partition is put into an Erase Setup state (row 63). In order to start the erase, an Erase Confirm command is given which puts the partition into an Erase (busy) state (row 74). Before the erase has completed, information needs to be read from a different block within the same partition. To do this, an Erase Suspend (B0H) command is issued. This is the only command that this partition will accept; all other commands will be ignored. The partition then goes into an Erase Suspend Read Status state (row 77). The status register can be read to determine when the erase has been successfully suspended. At this point a Read Array command can be given which puts the partition into erase suspend read array mode (row 81). Array data can now be read. The partition will stay in this mode, and the erase will stay suspended until an Erase Resume command (D0H) is issued which returns the partition back to an Erase (busy) state. When the erase has completed, the partition will go into the Erase (done) state (row 72), and the partition is ready to accept another command.

7.2.2.3 Read While Erase/Program

This example will describe reading from the bottom partition while the top partition is in erase mode. The top and bottom partitions are both initially in read array mode (row 3). The top partition is issued an Erase Setup (20H) command putting the partition into an Erase Setup state (row 63). An Erase Confirm command is then given, putting the top partition into an Erase (busy) state. Information from the bottom partition then needs to be read. The state table is now used to show the state of the bottom partition, which has become the current partition. Its state is shown in row 2. The partition is already in read array mode, so the Read Array command does not have to be issued. If, however, the device is in a different read mode, such as read status mode, then a Read Array command will have to be issued. The block in the top partition continues to be erased throughout the read cycle. When the erase has completed, the current state of the top partition is shown by row 71, and the partition is ready to accept a new command.

7.2.2.4 Read While Program-Suspend During Erase-Suspend

This example will outline reading from the bottom partition while the top partition is in erase suspend mode and the bottom partition is in program mode. Both top and bottom partitions are initially in read array mode. An Erase Setup command is issued to the top partition, putting it into Erase Setup, row 63. An Erase Confirm command is then issued to the top partition which starts erasing the block (row 74). At this point data needs to be programmed into the bottom partition. The erase in the top partition is suspended by issuing the Erase Suspend command. The state table is now used to show the state of the bottom partition. Its current state is shown in row 4; it is in read array mode while the other partition is in Erase Suspend mode. A Program Setup command is issued to the bottom partition. The next command to the partition programs the device, and puts the partition into the Program (busy) state, row 45. At this point, data from the bottom partition needs to be read. Its program cycle is suspended with a Program Suspend command. Its current state is shown in row 53. Issuing a Read Array command to the bottom partition will put it into the

Program Suspend Read Array state, row 56. Array data can now be read. When array data has been read, the program can be resumed by writing the Program Resume command. The partition goes back to program mode, then completes programming the device. The bottom partition is then idle, and the top partition is still in erase suspend mode. The state table now shows the current state of the top partition in row 77. An Erase Resume command resumes the erase. When the erase has completed, both partitions are idle, and can accept new commands.

7.2.3 Addresses during Writes to Flash

In previous Intel Flash products such as the Fast Boot Block and Intel® StrataFlash™ memory families, the address while writing a command was a don't care. In 1.8 Volt Dual-Plane Flash memory this address should be the address location to which the command is referring (see Table 5 on page 15). For example, the first address in a block erase command should be an address within that block; the first address in a program command should be the address of the word to be programmed.

7.3 System Design Considerations

7.3.1 CPU Compatibility

1.8 Volt Dual-Plane Flash memory supports two high-performance read modes:

- Asynchronous page mode
- Synchronous burst mode

These two read modes allow the processor, if capable, to achieve much higher bandwidth than was previously possible using single read accesses. The asynchronous page mode is ideal for non-clocked memory systems and is compatible with standard page-mode ROM. If the memory subsystem has access to an external processor referenced clock, the synchronous burst mode can be used for increased read performance, provided the clock frequency is below 40 MHz.

If the system CPU or ASIC does not support burst or page-mode reads, single synchronous and asynchronous reads are possible.

Whether the flash component is in synchronous or asynchronous mode depends on the setting in the Configuration Register. Setting bit 15 to 0 enables synchronous burst reads, and setting the bit to 1 enables asynchronous reads.

Upon reset, the device defaults to asynchronous page mode, and is put into read array mode. This corresponds to the state of most processors upon startup. It is important to reset the flash memory device when the processor is reset. This is because when the processor returns from reset it will request memory from the flash array. If the flash has not been reset, it is possible for it to be in read status or read ID mode, which would then return unwanted data to the processor.

7.3.2 Flash Integrated Features

The key to enabling compatibility between 1.8 Volt Dual-Plane Flash memory and today's burst-capable microprocessors are 1.8 Volt Dual-Plane Flash memory's integrated features. These features, listed and explained below, help simplify and eliminate excess system interface logic.

- Address Latch

- Read Configuration Register
- Status Register
- WAIT# Output

7.3.2.1 Address Latch

The address latch latches the address during read and write cycles. The internal address latch is controlled by ADV#. When ADV# is low, the latch is open. The latch closes when ADV# is driven high or upon the first rising (or falling) edge of CLK when ADV# is low. This stores the current address on the bus into the flash memory device and lets the address bus change without affecting the flash. This pin works the same in write operations; the address to be written to gets latched on the rising ADV# edge. Since writes are asynchronous, CLK is ignored and the address is not latched on the clock edge. During asynchronous reads the address latch does not need to be used, but addresses must then stay stable during the entire read operation. If ADV# is not used, addresses are latched on the rising edge of CE# during reads, and on the rising edge of CE# or WE# during writes, whichever goes high first.

7.3.2.2 Read Configuration Register

The read configuration register is a 16-bit register which sets the device's read configuration, burst order, frequency configuration and burst length. This register is stored in volatile memory within the memory device, and is initialized upon return from reset. With the Read Configuration Register, features of the flash memory device can be easily changed. Previous flash memory devices such as Advanced Boot Block and Intel StrataFlash memory families did not contain this register; rather features in these devices were set in hardware and were unchangeable. Being able to change these features allows a single flash memory component to have several different hardware features, configurable by the user. This allows this flash chip to work with a wide array of processors, regardless of their hardware requirements.

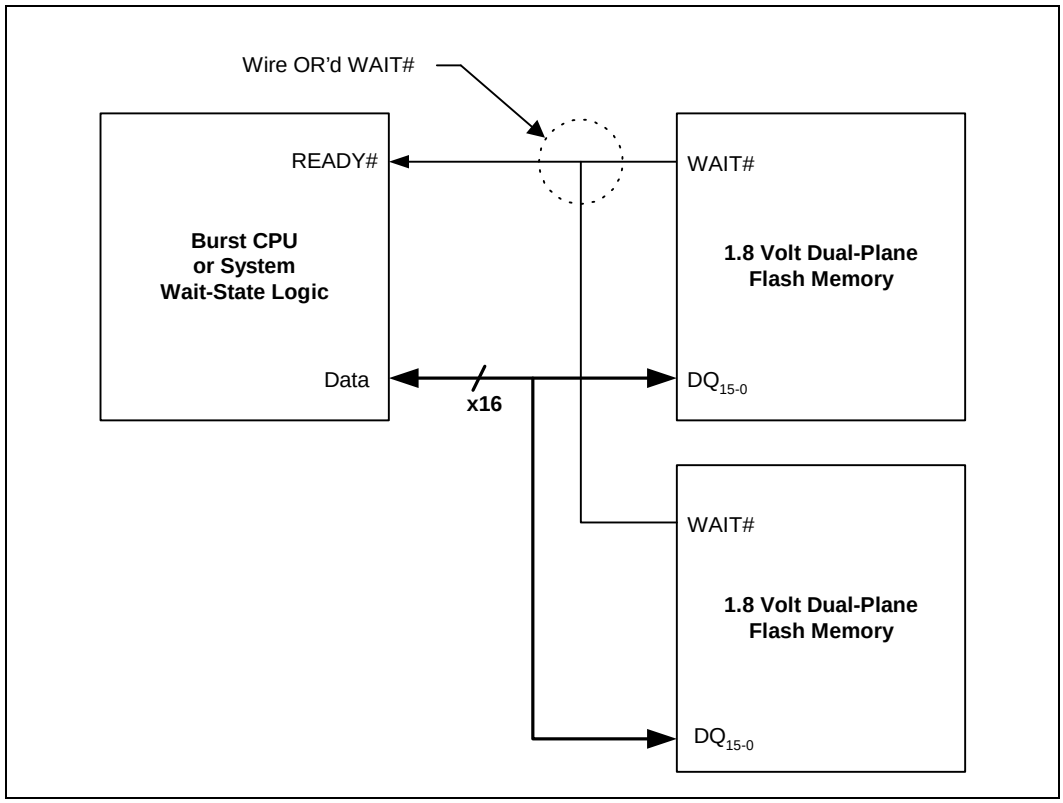
7.3.2.3 Status Register

1.8 Volt Dual-Plane Flash memory contains two status registers, one for each partition. Each one is an eight-bit register which contains the current information about the write state machine, the logic which controls programming and erasing the device's memory blocks. This register will report if a program or erase command had completed successfully, and if not, a reason for the error. Also this register will report when a program or erase has been suspended, so that the processor can then issue a program, erase, or read command. This register cannot be written to, only cleared, by issuing the Clear Status Register command, or by resetting the device.

7.3.2.4 WAIT# Output

1.8 Volt Dual-Plane Flash memory supports four-word, eight-word, and continuous burst lengths. In continuous burst length, or in 4- or 8-word burst accesses with RCR.3 = 1, an output ball, WAIT# is provided to simplify CPU to memory communication. The WAIT# informs the system when data is valid. At a logic 1, there is valid data on the bus, at a logic 0, the data on the bus is invalid. Figure 15, "WAIT# Pin Connection Using Multiple Flash Memory Components" on page 45, shows how the WAIT# signal can be OR'd for interface to a CPU, to use multiple flash components.

Figure 15. WAIT# Pin Connection Using Multiple Flash Memory Components



7.3.3 Using Asynchronous Page Mode

Upon power-up or return from reset, the device defaults to asynchronous page mode, with a page size of four words. This read mode is only supported from the main blocks, and is not supported from other locations within the device, such as the parameter blocks, the device identification codes, query information, and status register.

In asynchronous page mode, CLK is ignored and ADV# must be held low throughout the page access. With ADV# low, the internal address latch is open, allowing new page accesses. The initial valid address will store four words of data in the internal page buffer. Each word is then output onto the data lines by toggling address lines A_{1-0}.

If an application only uses the asynchronous page mode capability, CLK and ADV# can be tied to V_{SS}, as shown in Figure 20, "Different Clock Options" on page 48. This shows an ideal, glueless interface. If the processor does not provide any or all of these signals, some glue logic may be required. More information on signal generation is covered later in this section. Grounding CLK and ADV# will minimize the power consumed by these two pins and will simplify the interface, making it compatible with standard flash memory and industry standard page mode ROMs. With the ADV# signal tied low, the addresses cannot be latched into the device. This means that addresses must stay valid throughout the entire read or write cycle, until CE# or either WE# or OE# go high. Figure 17, "Asynchronous Page Mode Read Waveform" on page 46, shows an asynchronous read timing diagram with ADV# held low. Note that address lines A_{1-0} are toggled to clock out the data.

Figure 16. Asynchronous Page Mode Block Diagram

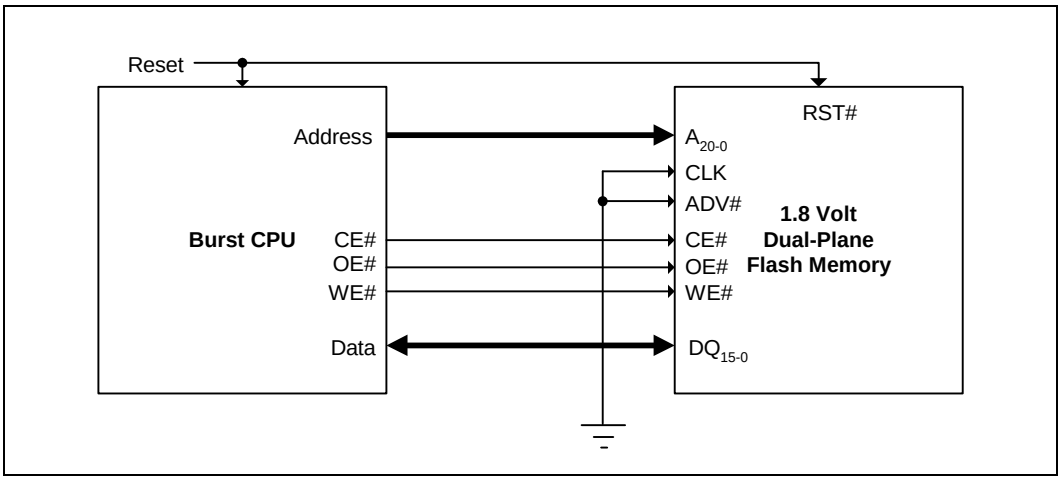
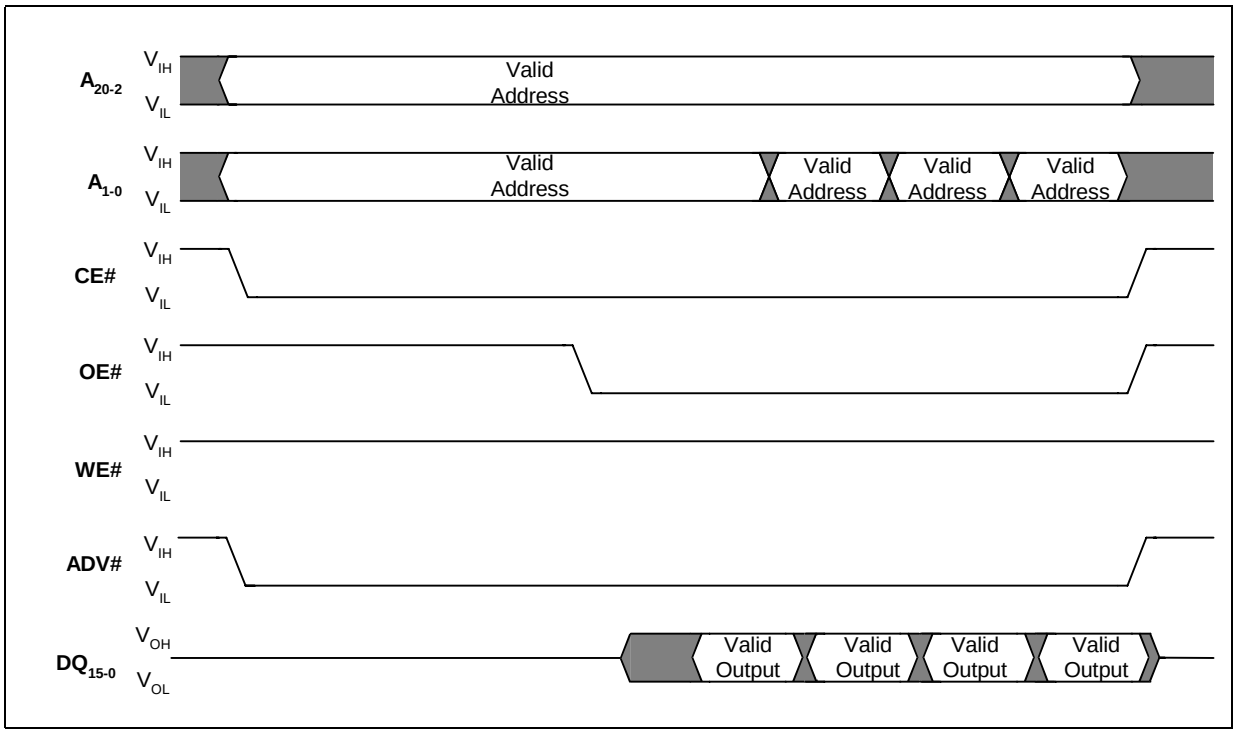


Figure 17. Asynchronous Page Mode Read Waveform



7.3.4 Using Synchronous Burst Mode

Synchronous burst mode provides a performance increase over asynchronous page-mode reads. It supports effective zero wait-state performance up to 40 MHz. This read mode is only supported from main blocks, and is not supported from other locations within the device, such as parameter blocks, the device identification mode, Query information, and Status Register. However, read operations from these locations while in synchronous burst mode transpire as single synchronous

reads. Burst reads are limited to within a partition, it is not possible to do a burst read across the partition boundary. A block diagram showing signal connections is shown in Figure 18. This is an ideal interface, and some glue logic may be required if the processor does not provide any or all of these signals. Figure 19 shows a synchronous burst mode read timing diagram. Note that only one address is needed from the processor to generate four valid data outputs.

Figure 18. Synchronous Burst Read Interface Block Diagram

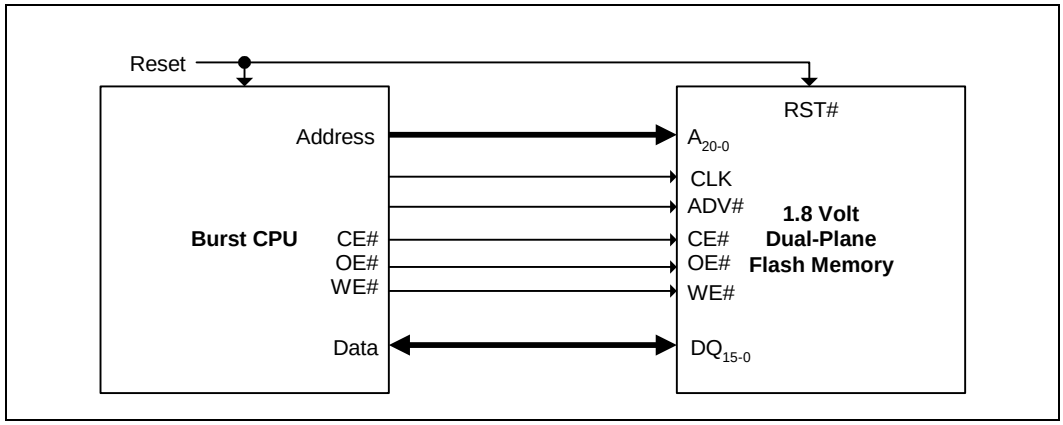
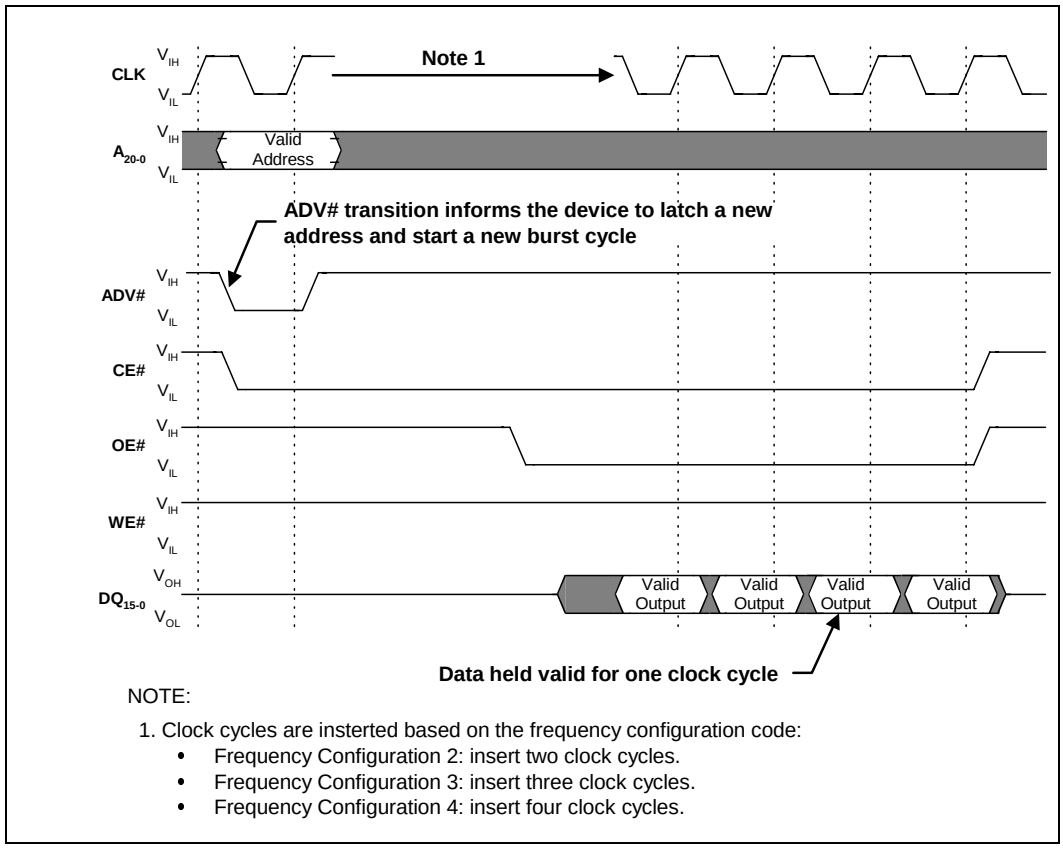


Figure 19. Synchronous Burst Mode Read Waveform



NOTE:

1. Clock cycles are insterted based on the frequency configuration code:
   - Frequency Configuration 2: insert two clock cycles.
   - Frequency Configuration 3: insert three clock cycles.
   - Frequency Configuration 4: insert four clock cycles.

Different interface considerations need to be made when booting from 1.8 Volt Dual-Plane Flash memory depending on whether or not the processor supports burst read operations at boot-up.

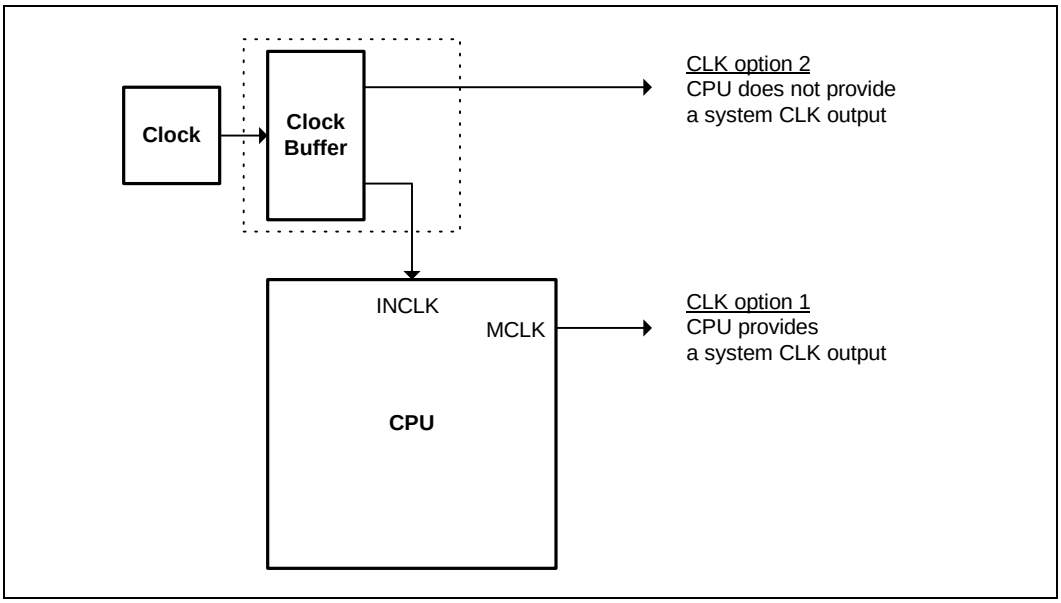
- Case 1, the processor does not support burst read mode at boot-up, but rather boots up in asynchronous page mode. This is the initial state of the flash memory, so no special design considerations need to be made. After booting up, the processor can, if possible, configure the flash memory for synchronous burst mode.
- Case 2, the processor does support burst mode at boot-up. After return from reset, the flash memory defaults to asynchronous read mode, which is inherently slower than synchronous burst mode. External interface logic will be needed to inform the processor of this, and to insert wait states to match the flash memory's timing with the processor's timing. This logic is only necessary until the processor has a chance to switch the memory device to synchronous burst mode, at which time the external logic must be notified of this change. This can be accomplished via a write-able register within the system wait-state logic or via a general purpose I/O (GPIO) pin. The GPIO pin may operate as an input into the system logic.

7.3.5 Signal Generation

Other than address and data pins, 1.8 Volt Dual-Plane Flash memory has several control pins as well. This section will cover these pins and how to generate these signals.

- ADV# can be derived from the processor's transaction start signal. If the processor does not have this type of signal, other standard CPU control signals can be used to control ADV#. The key characteristic of this signal is that it must toggle to inform the device to latch a new address. If this signal is not used, in asynchronous page mode only, then CE# must toggle to inform the flash memory of a new address.
- CLK can be derived from the processor's memory clock output. If the processor does not supply this control signal to the memory subsystem, the signal can be received from the clock signal generator through a clock buffer. This buffer minimizes clock load and skew. The clock signal must have a period of at least 25 ns. Figure 20 illustrates different clock options.

Figure 20. Different Clock Options



- WP#: on the Fast Boot Block and other Intel Flash memory families, this pin was the only way of locking and unlocking lockable blocks. On 1.8 Volt Dual-Plane Flash memory, locking and unlocking lockable blocks is possible through both hardware and software. Initially, upon reset, all blocks are locked and cannot be programmed or erased regardless of the value of WP#. In order to write to or erase a block, it must first be unlocked. This is done through software. An unlocked block can be programmed or erased regardless of the value of WP#. Only when a block is marked “lock-down” does the WP# pin have an effect on memory. In order to program/erase a locked-down block, the WP# pin must be high, and the block must then be unlocked. The block may then be programmed or erased as long as WP# is high. When WP# goes low, the block reverts to lock-down and can no longer be programmed or erased. The only way to get the block out of lock-down mode is to reset the device. If WP# is not used, it should be tied high. This will insure that blocks can be locked and unlocked through software, even after setting the lock-down bit. With WP# tied low, blocks can still be locked and unlocked through software, but if a block is locked down, it will remain in a lock-down state, and cannot be programmed or erased until the flash memory is reset.
- WE# / OE#: Processors that have separate pins to signal reads and writes can, in most cases, connect directly to these pins on the flash memory component. Processors that have a single pin which determines a read or a write can use this signal directly as either WE# or OE#, depending on what the low value means on that pin. The other input signal will then need to be generated via external logic to ensure that it goes low and high at the right times.
- RST# on the flash can be connected to the reset signal to the processor provided that the time from deserting reset to the processor’s first memory request is longer than the time required of the flash. The maximum delay from deasserting reset to valid data for 1.8 Volt Dual-Plane Flash memory is 150 ns. If the processor takes less time than that and requires memory before the flash component is ready, the data will be invalid. If the processor takes longer than the flash to reset, there is no problem. If this pin is kept low on power-up it will prevent possible spurious writes. If V_{PP} ramps up before V_{CC} or V_{CC} drops before V_{PP}, random noise on the data pins can possibly enter a program command (40H) with CE# and WE# low and OE# high. With RST# low it will prevent this spurious write.
- Most processors will expect data during read cycles much sooner than the flash memory component can provide it. For this reason the processor needs to be able to pause and wait for the flash memory. This can be done by programming the processor to generate a set number of wait states, if possible. If the processor is unable to internally generate wait-states, an input pin to the processor tells it when to pause and wait for valid data from the flash memory. This pin can then also be used as an input from the flash memory’s WAIT# signal during continuous reads or during 4- or 8-word reads in non-wrap mode (RCR.3 = 1)

7.3.6 Using WAIT# in Burst Mode

The 1.8 Volt Dual-Plane Flash memory supports 4-word, 8-word, and continuous burst lengths. In continuous burst length, or in 4-word or 8-word burst lengths with no-wrap (RCR.3 = 1), an output pin, WAIT#, is provided to simplify CPU to memory communication. The WAIT# informs the system to when data is valid.

- WAIT# = Logic ’1’ means Valid Data
- WAIT# = Logic ’0’ means Invalid Data

When operating in the continuous burst mode, or during 4 or 8-word reads in non-wrap mode (RCR.3 = 1), the flash memory may incur an output delay when the burst sequence crosses the first 16-word boundary. The starting address dictates whether or not a delay will occur. If the starting address is aligned to a four-word boundary, the delay will not be seen. If the starting address is the end of a four-word boundary, the output delay will be equal to the frequency configuration setting;

this is the worst case delay. The delay will only take place once during a continuous burst access, and if the burst sequence never crosses a 16-word boundary, the delay will never happen. When the output delay is encountered, the WAIT# pin will be asserted. This signal should be fed into the systems wait-state control logic or directly to the CPU.

The WAIT# output pin is gated by OE# and CE#. If either OE# or CE# go inactive, the WAIT# output buffer turns off. An internal pull-up resistor holds WAIT# at a logic '1' state. The resistor value is approximately 1 MΩ. This output configuration allows multiple banks of flash enable wire ORing, as shown in Figure 15, "WAIT# Pin Connection Using Multiple Flash Memory Components" on page 45. WAIT# can also be configured for a couple different characteristics to help simplify system usage. It can be configured for assertion during the delay or one data cycle before the delay.

7.3.7 Write Operations

Write operations are used to switch the memory device between modes, to initiate a program or erase, to lock or unlock blocks, and to write memory to the device to be stored. Commands that switch modes of the device or suspend/resume a program or erase take one write cycle. Commands which initiate a program or erase or lock/unlock blocks take two write cycles. A program command is required before each data word to be programmed into the flash device, even if multiple data words are programmed back to back.

Write operations transpire as asynchronous operations, similar to other Intel Flash memory families, such as Advanced or Advanced+ Boot Block and Fast Boot Block memory. The flash memory latches the address during writes the same way as during reads. The data, as in all Intel Flash memory components, is latched on the rising edge of CE# or WE#, whichever goes high first. For a write cycle, WE# and CE# are interchangeable.

After completing a program or erase or program/erase suspend command, the flash device automatically goes into read status mode. Any reads to the flash at this point will return status register data. This data is not updated to the output pins continually, rather CE# and/or OE# need to be toggled for updated status register data. After a Set Read Configuration Register command, the flash device goes into read array mode.

7.4 Design Tools and Software

7.4.1 Design Tools

Several tools are available which will simplify designing in 1.8 Volt Dual-Plane Flash memory components into a system. They include VHDL and Verilog bus functional models, Timing Designer* files, and IBIS files. Some of these tools can be found on Intel's website, otherwise they can be obtained by contacting an Intel field representative. Visit http://developer.intel.com/design/flash/swtools/ for more details.

7.4.2 Flash Data Integrator (FDI)

Intel® Flash Data Integrator software is a code plus data storage manager for use in real-time embedded applications. This software enables code execution and data storage in a single flash device. It handles parameter, data-stream, and packet storage, as well as sophisticated file-system features like wear-leveling, power-loss recovery and block reclaims. By consolidating code and data storage in a single flash device, FDI reduces component count, allowing decreases in board

size, power consumption and cost. FDI is designed to fully support the special features of 1.8 Volt Dual-Plane Flash memory. More information on Flash Data Integrator (FDI) software is available on Intel's website at: http://developer.intel.com/design/flcomp/.

8.0 Electrical Specifications

8.1 Absolute Maximum Ratings

Parameter	Maximum Rating
Temperature under Bias	–40°C to +85°C
Storage Temperature	–65°C to +125°C
Voltage On Any Pin (except V_{PP})	–0.5 V to +2.45 V(1)
V_{PP} Voltage	–2.0 V to +14.0 V(1,2,4)
V_{CC} and V_{CCQ} Voltage	–0.2V to +2.45V
Output Short Circuit Current	100 mA(3)

NOTES:
1. All specified voltages are with respect to V_{SS}. Minimum DC voltage is –0.5 V on input/output pins and –0.2 V on V_{CC} and V_{PP} pins. During transitions, this level may undershoot to –2.0 V for periods <20 ns. Maximum DC voltage on input/output pins and V_{CC} 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} Program voltage is normally 1.65 V–1.95 V. Connection to a 11.4 V–12.6 V supply can be done for a maximum of 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.

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

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

8.2 Extended Temperature Operation

Symbol	Parameter	Notes	Min	Nominal	Max	Unit
T_A	Operating temperature		-40	+25	+85	°C
V_{CC}	V_{CC} supply voltage	1	1.65	1.8	1.95	V
V_{CCQ}	I/O supply voltage	1	1.65	1.8	1.95	V
V_{PP1}	V_{PP} supply voltage when used as a logic control	1	0.9	1.8	1.95	V
V_{PP2}	V_{PP} supply voltage	1, 2	11.4	12	12.6	V
Cycling	Main block erase cycling; V_{PP} = 1.8 V		100,000			Cycles
	Parameter block erase cycling; V_{PP} = 1.8 V		100,000			Cycles
	Main block erase cycling; V_{PP} = 12 V, 80 hrs.				1000	Cycles
	Parameter block erase cycling; V_{PP} = 12 V, 80 hrs.				2500	Cycles
	Maximum V_{PP} hours at 12 V				80	Hours

NOTES:
1. See DC Characteristics tables for voltage range-specific specifications.
2. Applying V_{PP} =11.4 V– 12.6 V during a program or erase can be done for a maximum of 1000 cycles on the main blocks and 2500 cycles on the parameter blocks. A permanent connection to V_{PP} = 11.4 V– 12.6 V is not allowed and can cause damage to the device.

8.3 Capacitance

T_A = +25°C, f = 1 MHz

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

NOTE: 1. Sampled, not 100% tested.

8.4 DC Characteristics (1)

Sym	Parameter(1)		Note	Min	Typ	Max	Unit	Test Condition	
I_{LI}	Input Load Current					±1	µA	$V_{CC} = V_{CC}$ Max, $V_{CCQ} = V_{CCQ}$ Max, $V_{IN} = V_{CCQ}$ or V_{SSQ}	
I_{LO}	Output Leakage Current		2			±1	µA	$V_{CC} = V_{CC}$ Max, $V_{CCQ} = V_{CCQ}$ Max, $V_{IN} = V_{CCQ}$ or V_{SSQ}	
I_{CCS}	V_{CC} Standby Current				13	20	µA	$V_{CC} = V_{CC}$ Max, CE# = RST# = V_{CCQ}, WP# = V_{CCQ} or V_{SSQ}, $A_{18} = A_{19} = V_{CC}$ ± 0.2V	
I_{CCR}	Average V_{CC} Read Current	Page Mode	3		12	18	mA	4 Word Read	f = 13 MHz $V_{CC} = V_{CC}$ Max
		Synchronous CLK = 13 MHz	3, 4		9	13	mA	Burst length = 4	CE#=V_{IL}, OE#=V_{IH}, Inputs = V_{IH} or V_{IL}
					11	17	mA	Burst length = 8	
					42	59	mA	Burst length = Continuous	
		Synchronous CLK = 40 MHz	3, 4		20	29	mA	Burst length = 4	f = 40 MHz $V_{CC} = V_{CC}$ Max
					28	40	mA	Burst length = 8	CE#=V_{IL}, OE#=V_{IH}, Inputs = V_{IH} or V_{IL}
					42	59	mA	Burst length = Continuous	
I_{CCW} + I_{PPW}	V_{CC} Program Current		5, 6		30	55	mA	$V_{PP} = V_{PP1}$, program in progress	
					18	35	mA	$V_{PP} = V_{PP2}$, program in progress	
I_{CCE}+ I_{PPE}	V_{CC} Block Erase Current		5, 7		30	45	mA	$V_{PP} = V_{PP1}$, block erase in progress	
					16	35	mA	$V_{PP} = V_{PP2}$, block erase in progress	
I_{CCWS}	V_{CC} Program Suspend Current		5, 8		13	20	µA	CE# = V_{IH}, program suspend in progress	
I_{CCES}	V_{CC} Erase Suspend Current		5, 8		13	20	µA	CE# = V_{IH}, block erase suspend in progress	
I_{PPS}	V_{PP} Standby Current				0.5	1	µA	$V_{PP} \le V_{CC}$	
I_{PPR}	V_{PP} Read Current				0.5	1	µA	$V_{PP} \le V_{CC}$	
I_{PPWS}	V_{PP} Program Suspend Current		5		0.5	1	µA	$V_{PP} = V_{PP1/2}$, program suspended	
I_{PPES}	V_{PP} Erase Suspend Current		5		0.5	1	µA	$V_{PP} = V_{PP1/2}$, erase suspended	
V_{IL}	Input Low Voltage			-0.4		0.4	V		
V_{IH}	Input High Voltage			V_{CCQ} - 0.4		V_{CCQ} +0.4	V		

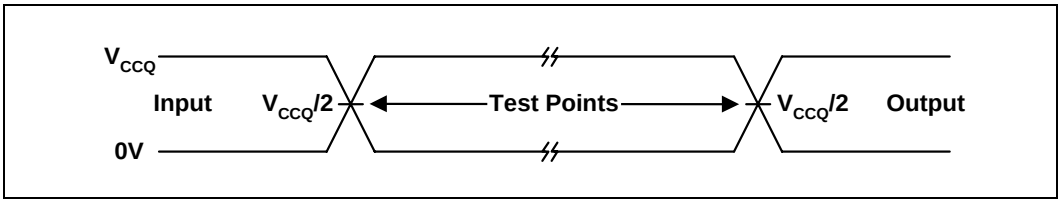
DC Characteristics, continued

Sym	Parameter[1]	Note	Min	Typ	Max	Unit	Test Condition
V_{OL}	Output Low Voltage				0.1	V	$V_{CC} = V_{CC}$ Min, $V_{CCQ} = V_{CCQ}$ Min, I_{OL} = 100 µA
V_{OH}	Output High Voltage		V_{CCQ} - 0.1			V	$V_{CC} = V_{CC}$ Min, $V_{CCQ} = V_{CCQ}$ Min, I_{OH} = –100 µA
V_{PPLK}	V_{PP} Lock-Out Voltage	9			0.4	V	
V_{LKO}	V_{CC} Lock Voltage		1.0			V	

NOTE:
1. All currents are RMS unless noted. Typical values at typical V_{CC}, T_A = +25°C.
2. WAIT# I_{LO} = 2 µA max.
3. Automatic Power Savings (APS) reduces I_{CCR} to approximately standby levels in static operation.
4. The burst wrap bit (RCR.3) determines whether 4- or 8-word burst-accesses wrap within the burst-length boundary or whether they cross word-length boundaries to perform linear accesses. In the no-wrap mode (RCR.3 = 1), the device operates similar to continuous linear burst mode but consumes less power during 4- and 8-word bursts.
5. Sampled, not 100% tested.
6. V_{CC} read + program current is the summation of V_{CC} Read and V_{CC} program currents.
7. V_{CC} read + program current is the summation of V_{CC} Read and V_{CC} block erase currents.
8. I_{CCES} is specified with device deselected. If device is read while in erase suspend, current draw is sum of I_{CCES} and I_{CCR}.
9. Erase and program operations are inhibited when $V_{PP} \leq V_{PPLK}$ and not guaranteed outside valid V_{PP1} and V_{PP2} ranges.

8.5 AC I/O Test Conditions

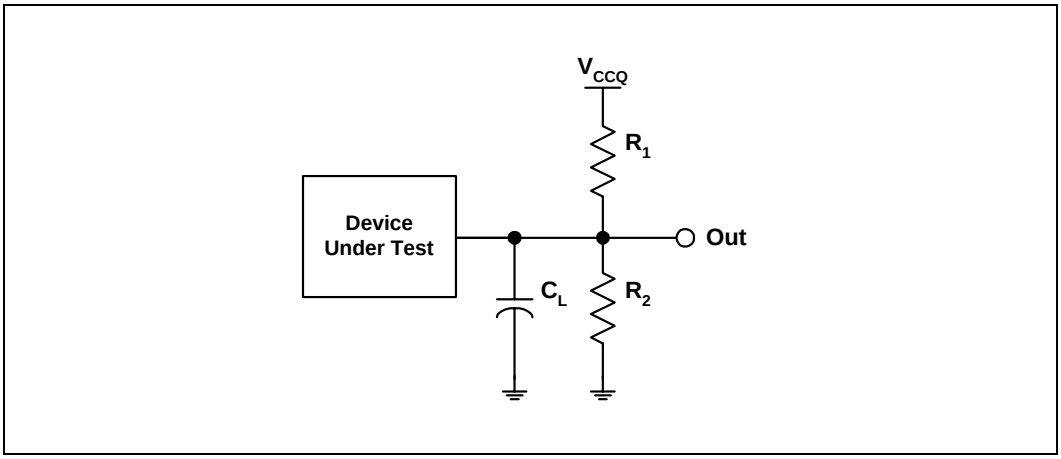
Figure 21. AC Input/Output Reference Waveform



0672_21

NOTE: AC test inputs are driven at 1.65 V for a Logic "1" and 0.0 V for a Logic "0." Input timing begins, and output timing ends, at $V_{CCQ}/2$. Input rise and fall times (10% to 90%) < 5 ns. Worst case speed conditions are when V_{CC} = 1.65 V.

Figure 22. Transient Equivalent Testing Load Circuit



0672_22

NOTE: See table for component values.

Test configuration component value for worst case speed conditions

Test Configuration	C_L (pF)	R_1 (Ω)	R_2 (Ω)
1.8 V Standard Test	50	16.7K	16.7K

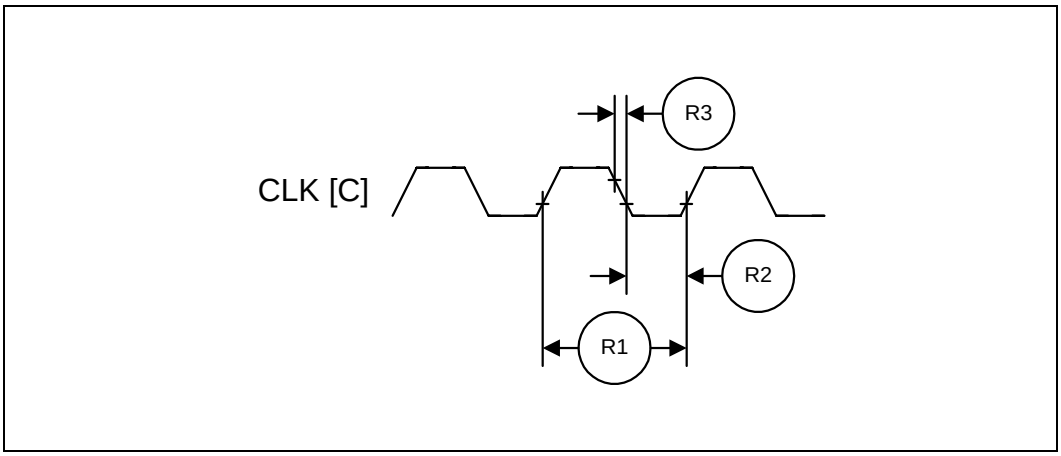
NOTE: C_L includes jig capacitance.

8.6 AC Read Characteristics

#	Sym	Product Parameter()	Notes	-110 Min	-110 Max	-120 Min	-120 Max	Unit
Clock Specifications								
R1	t_{CLK}	CLK period		25		25		ns
R2	t_{CH} (t_{CL})	CLK high (Low) time		7.5		7.5		ns
R3	t_{CHCL} (t_{CLCH})	CLK fall (Rise) time			5		5	ns
Synchronous Specifications								
R4	t_{AVCH}	Address valid setup to CLK		9		9		ns
R5	t_{VLCH}	ADV# low setup to CLK		9		9		ns
R6	t_{ELCH}	CE# low setup to CLK		9		9		ns
R7	t_{CHQV}	CLK to output delay			20		25	ns
R8	t_{CHQX}	Output hold from CLK	2	5		5		ns
R9	t_{CHAX}	Address hold from CLK	3	10		10		ns
R10	t_{CHTL} (t_{CHTH})	CLK to WAIT# delay			20		25	ns
R24	t_{EHEL}	CE# high between subsequent synchronous reads	4	20		20		ns
Asynchronous Specifications								
R11	t_{AVVH}	Address setup to ADV# high		10		10		ns
R12	t_{ELVH}	CE# low to ADV# going high		10		10		ns
R13	t_{AVQV}	Address to output delay	5		110		120	ns
R14	t_{ELQV}	CE# low to output delay			110		120	ns
R15	t_{VLQV}	ADV# low to output delay			110		120	ns
R16	t_{VLVH}	ADV# pulse width low		10		10		ns
R17	t_{VHVL}	ADV# pulse width high		10		10		ns
R18	t_{VHAX}	Address hold from ADV# high		9		9		ns
R19	t_{APA}	Page address access time			40		45	ns
R20	t_{GLQV}	OE# low to output delay			45		45	ns
R21	t_{PHQV}	RST# high to output delay			150		150	ns
R22	t_{EHQZ} t_{GHQZ}	CE# or OE# high to output in high z, whichever occurs first	3		25		35	ns
R23	t_{OH}	Output hold from first occurring address, CE#, or OE# change	3	0		0		ns

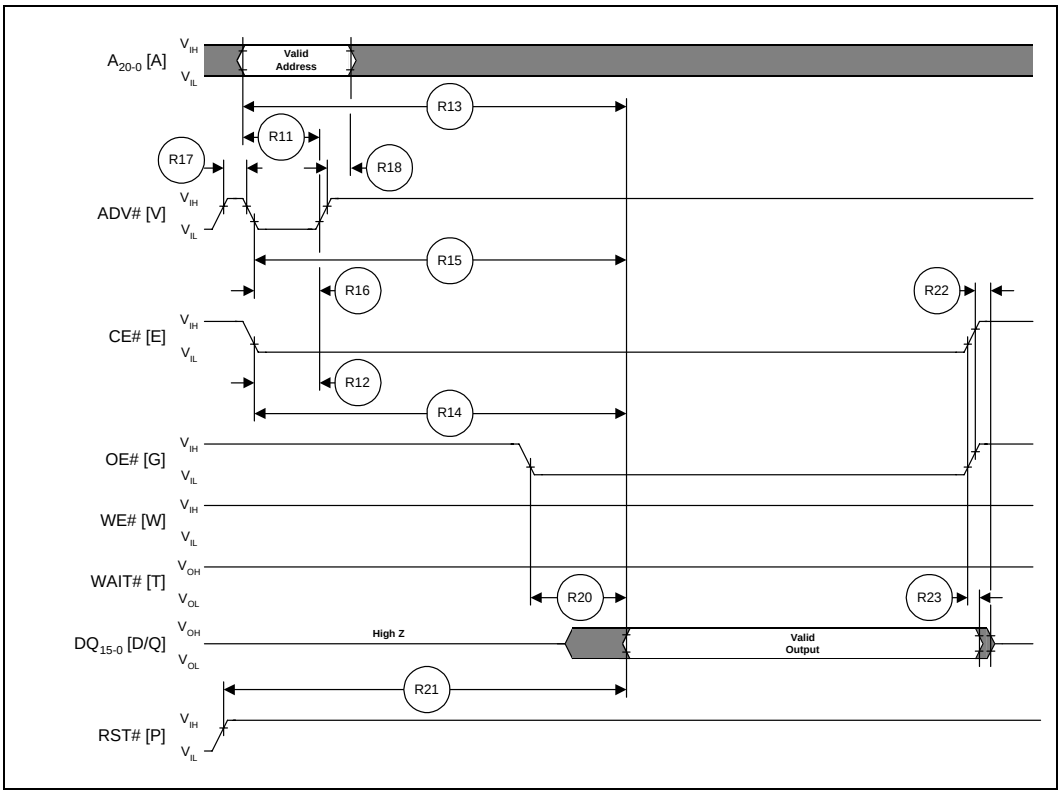
NOTES:
1. See AC input/output reference waveform for timing measurements and maximum allowable input slew rate.
2. Tested at worst case processor conditions.
3. Sampled, not 100% tested.
4. Applies only to subsequent synchronous reads.
5. OE# may be delayed up to t_{ELQV}–t_{GLQV} after the falling edge of CE# without impact to t_{ELQV}.

Figure 23. AC Waveform for CLK Input



0672_23

Figure 24. AC Waveform for Single Asynchronous Read Operations from Parameter Blocks, Status Register, Identifier Codes



0672_24

Figure 25. AC Waveform for Asynchronous Page-Mode Read Operations from Main Blocks

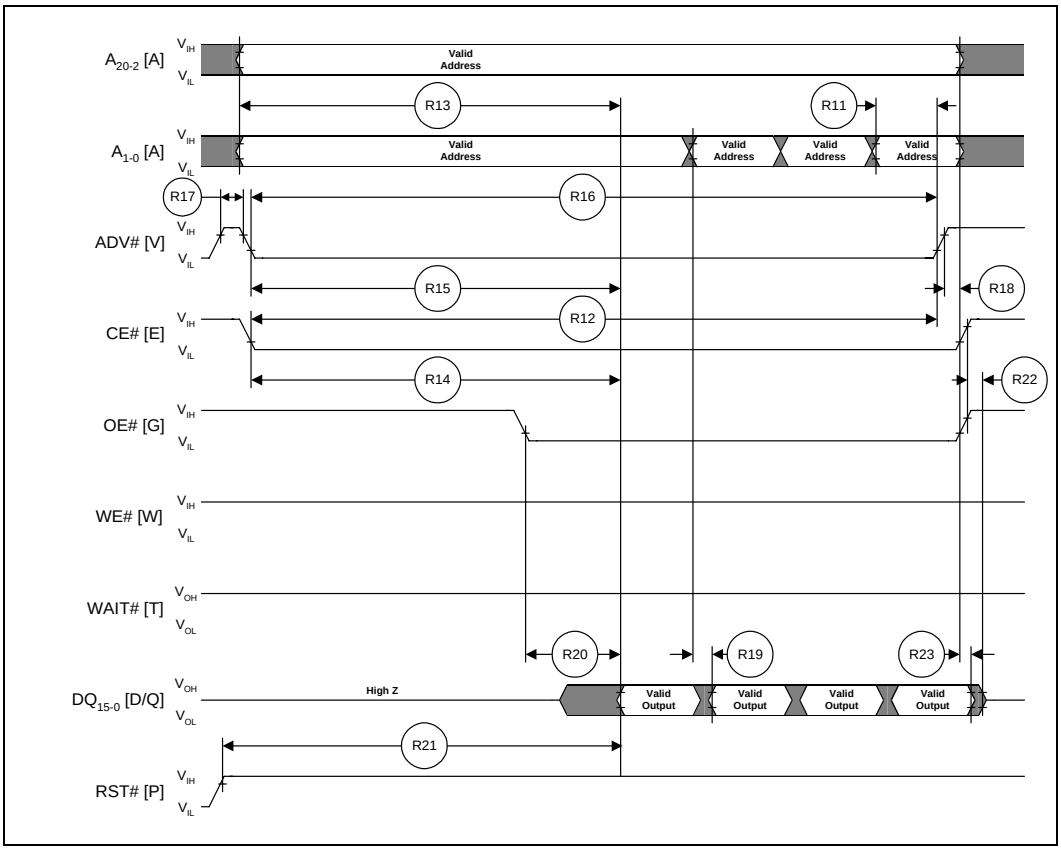
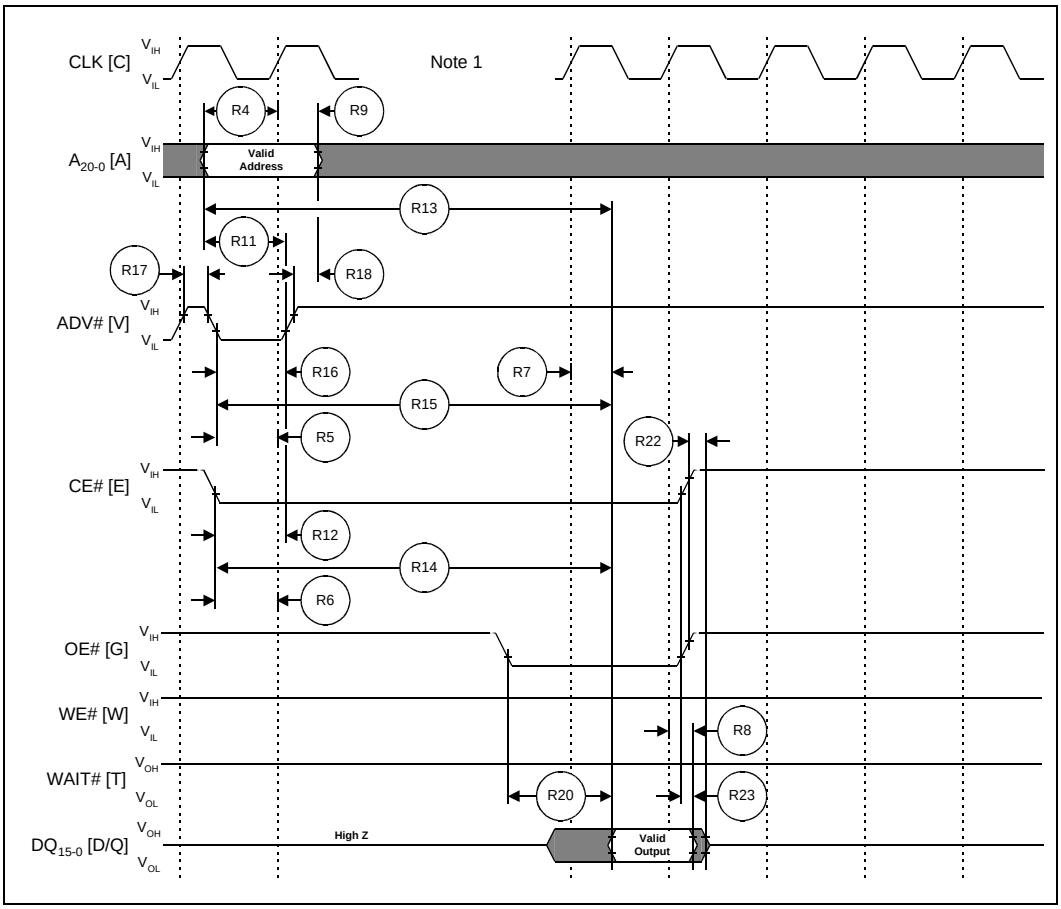


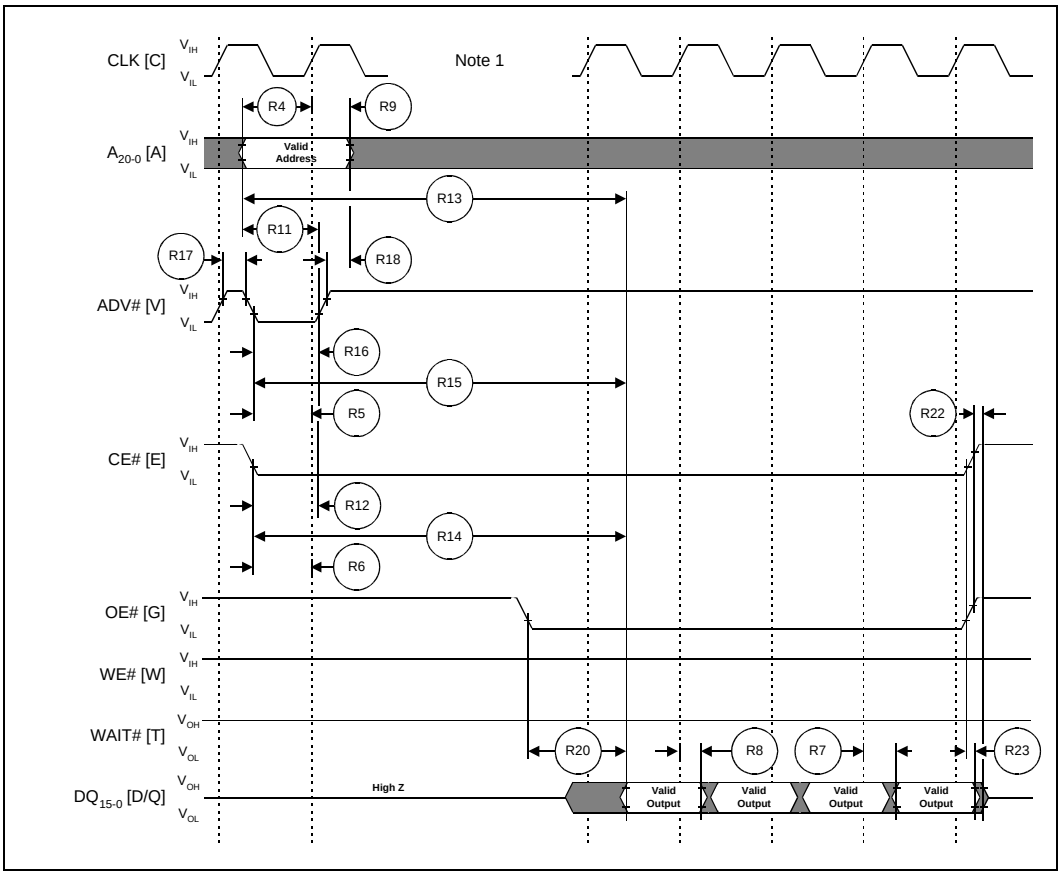
Figure 26. AC Waveform for Single Synchronous Read Operations from Parameter Blocks, Status Register, Identifier Codes



NOTE:

1. Depending upon the frequency configuration code value in the read configuration register, insert clock cycles:
   - Frequency Configuration 2 insert two clock cycles
   - Frequency Configuration 3 insert three clock cycles
   - Frequency Configuration 4 insert four clock cycles
2. See 4.10.2 for further information about the frequency configuration and its effect on the initial read.

Figure 27. AC Waveform for Synchronous Burst Read Operations, Four-Word Burst Length from Main Blocks

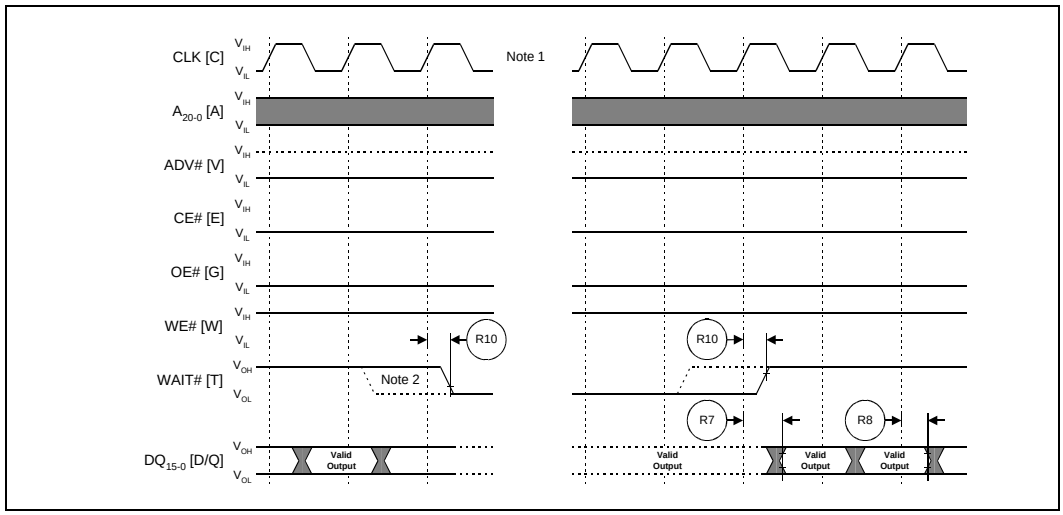


NOTE:

1. Depending upon the frequency configuration code value in the read configuration register, insert clock cycles:
   - Frequency Configuration 2 insert two clock cycles
   - Frequency Configuration 3 insert three clock cycles
   - Frequency Configuration 4 insert four clock cycles

   See 4.10.2 for further information about the frequency configuration and its effect on the initial read.

Figure 28. AC Waveform for Continuous Burst Read Showing an Output Delay with Data Output Configuration Set to One Clock



0672_28

NOTES:

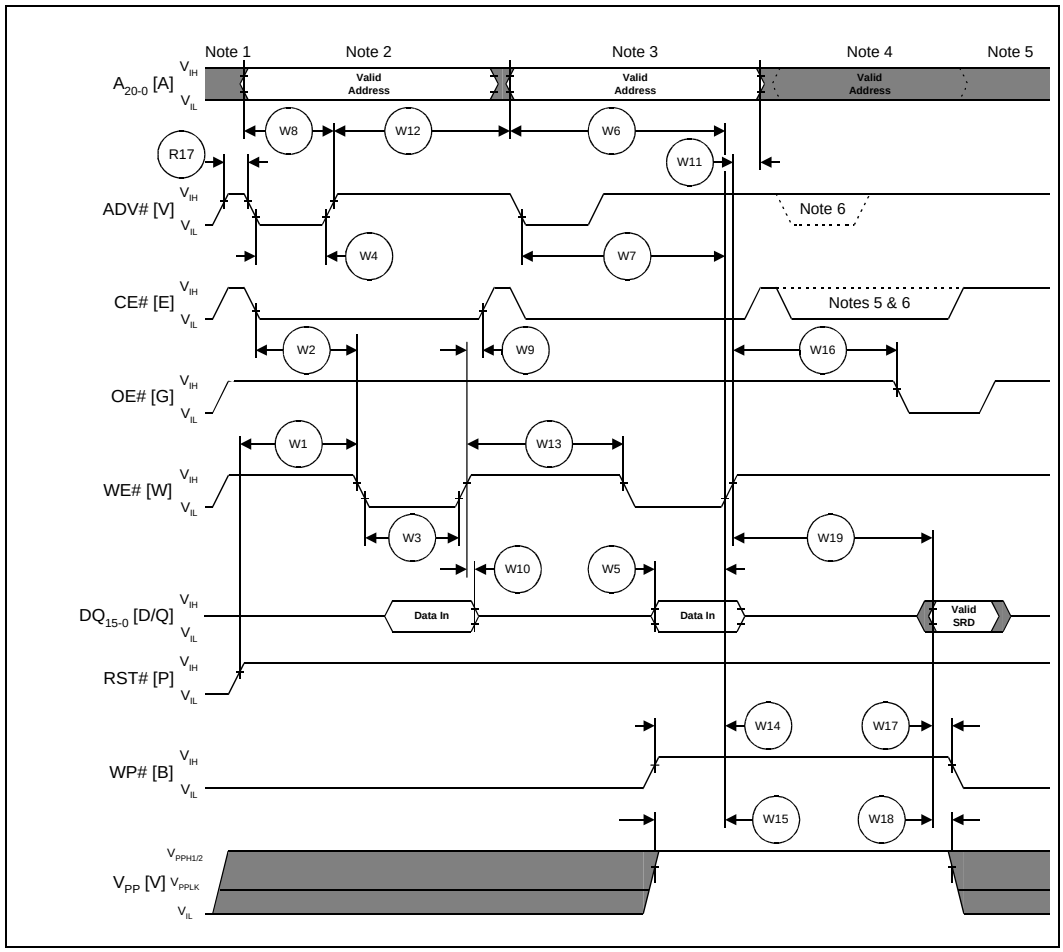
1. This delay occurs only in certain burst configurations. See 4.10.4 for further information about WAIT# behavior.
2. WAIT# configuration allows assertion one CLK cycle before or during output. See 4.10.4 for further information.

8.7 AC Write Characteristics

#	Sym	Parameter[1,2]	Notes	Min	Max	Unit
W1	t_{PHWL} (t_{PHEL})	RST# High Recovery to WE# (CE#) Going Low	3	150		ns
W2	t_{ELWL} (t_{WLEL})	CE# (WE#) Setup to WE# (CE#) Going Low	6	0		ns
W3	t_{WLWH}	Write Pulse Width	6	70		ns
W4	t_{VLVH}	ADV# Pulse Width		10		ns
W5	t_{DVWH} (t_{DVEH})	Data Setup to WE# (CE#) Going High	4	70		ns
W6	t_{AVWH} (t_{AVEH})	Address Setup to WE# (CE#) Going High	4	70		ns
W7	t_{VLWH} (t_{VLEH})	ADV# Setup to WE# (CE#) Going High		83		ns
W8	t_{AVVH}	Address Setup to ADV# Going High		10		ns
W9	t_{WHEH} (t_{EHWH})	CE# (WE#) Hold from WE# (CE#) High		0		ns
W10	t_{WHDX} (t_{EHDX})	Data Hold from WE# (CE#) High		0		ns
W11	t_{WHAX} (t_{EHAX})	Address Hold from WE# (CE#) High		0		ns
W12	t_{VHAX}	Address Hold from ADV# Going High		9		ns
W13	t_{WHWL} (t_{WHWL})	Write Pulse Width High	7	30		ns
W14	t_{BHWH} (t_{BHEH})	WP# Setup to WE# (CE#) Going High	3	200		ns
W15	t_{VVWH} (t_{QVEH})	V_{PP} Setup to WE# (CE#) Going High	3	200		ns
W16	t_{WHGL} (t_{EHGL})	Write Recovery before Read		0		ns
W17	t_{QVBL}	WP# Hold from Valid SRD	3, 5	0		ns
W18	t_{QVVL}	V_{PP} Hold from Valid SRD	3, 5	0		ns
W19	t_{WHQV}	WE# high to data valid	3, 8		t_{AVQV} + 50	ns

NOTES:
1. Read timing characteristics during block erase and program operations are the same as during read-only operations.
2. A write operation can be initiated and terminated with either CE# or WE#.
3. Sampled, not 100% tested.
4. Refer to Table 5 on page 15 for valid A_{IN} and D_{IN} for block erase or program.
5. V_{PP} should be held at $V_{PP1/2}$ until block erase or program success is determined.
6. 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, $t_{WP} = t_{WLWH} = t_{ELEH} = t_{WLEH} = t_{ELWH}$.
7. 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 last). Hence, $t_{WPH} = t_{WHWL} = t_{EHEL} = t_{WHEL} = t_{EHWL}$.
8. t_{WHQV} after Read Query, Device Identifier or Protection Register command = t_{AVQV} + 100 ns.

Figure 29. AC Waveform for Write Operations



0672_29

NOTES:

1. V_{CC} power-up and standby.
2. Write block erase or program setup.
3. Write block erase confirm or valid address and data.
4. Automated erase or program delay.
5. Read status register data.
6. For read operations, OE# and CE# must be driven active, and WE# de-asserted.

8.8 Reset Operations

Figure 30. AC Waveform for Reset Operations

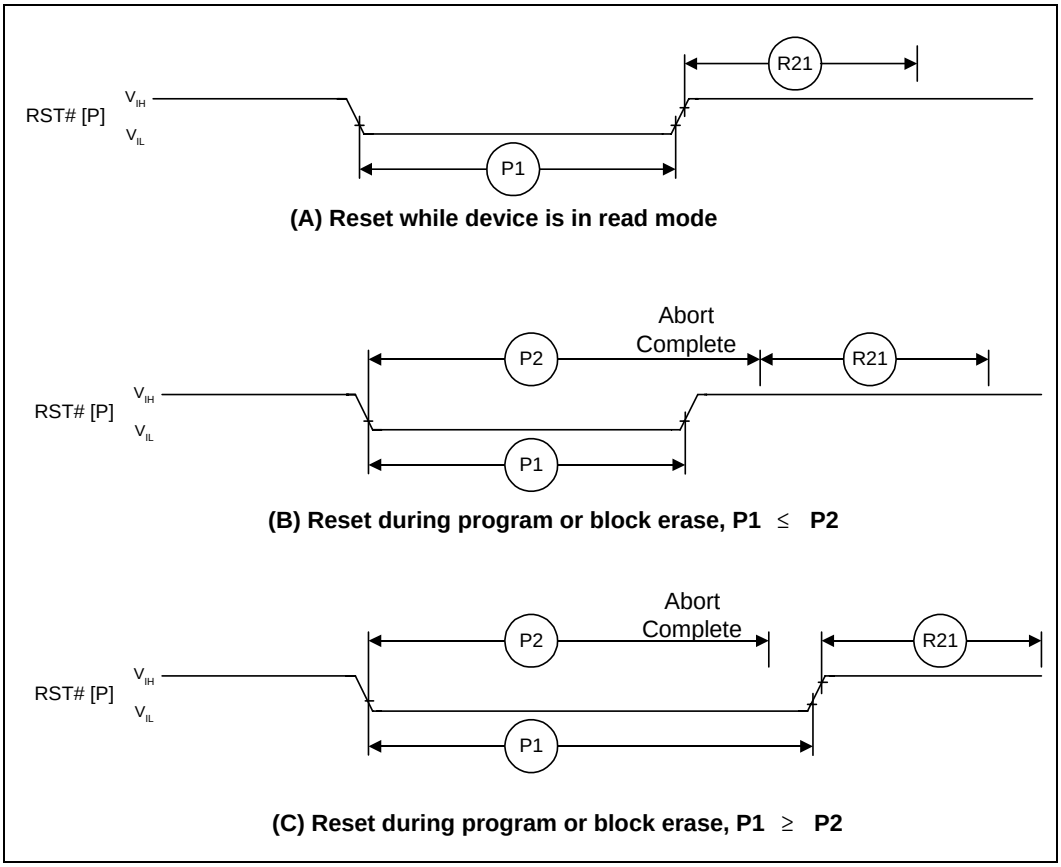


Table 12. Reset Specifications[1]

#	Symbol	Parameter	Notes	Min	Max	Unit
P1	t_{PLPH}	RST# low to reset during read (If RST# is tied to V_{CC}, this specification is not applicable)	2, 4	100		ns
P2	t_{PLRH}	RST# Low to reset during block erase	3, 4		22	µs
		RST# Low to reset during program	3, 4		12	µs

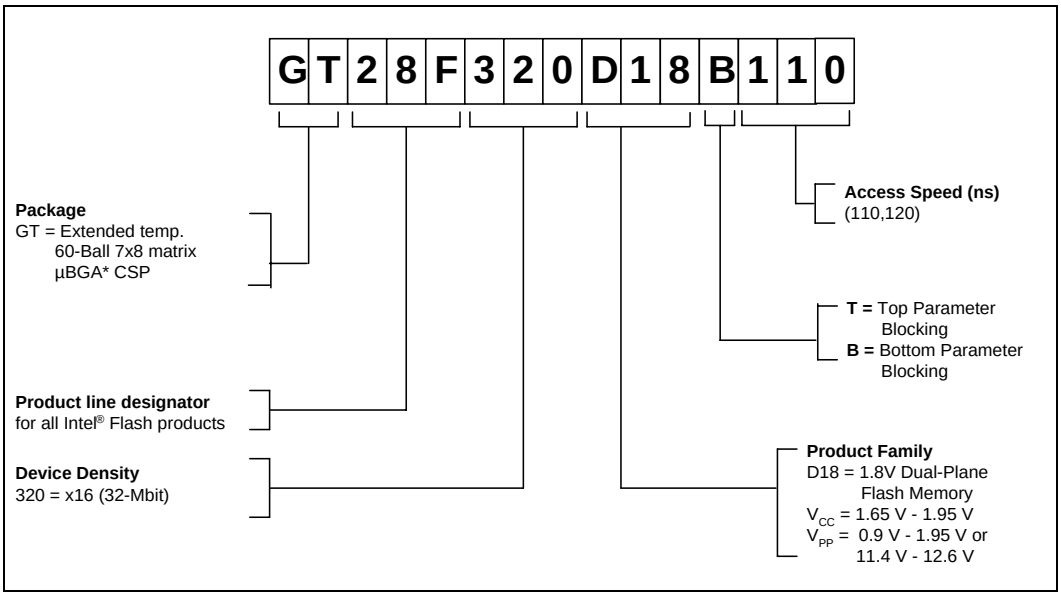
NOTES:
1. These specifications are valid for all product versions (packages and speeds).
2. t_{PLPH} is < 100 ns the device may still reset but this is not guaranteed.
3. If RST# is asserted while a block erase or word program operation is not executing, the reset will complete within 100 ns.
4. Sampled, but not 100% tested.

8.9 Block Erase and Program Performance

#	Symbol	Parameter[1]	V_{PP}	V_{PP1} (in system)		V_{PP2} (in manufacturing)		Unit
			Note	Typ[1]	Max[2]	Typ[1]	Max[2]	
W0	t_{BWPB}	4-KW Parameter Block Program Time	2	0.1	0.3	0.03	0.1	s
	t_{BWMB}	32-KW Main Block Program Time	2	0.8	2.4	0.24	0.80	s
	t_{WHQV1} / t_{EHQV1}	Word Program Time	2	22	200	8	185	µs
	t_{WHQV2} / t_{EHQV2}	4-KW Parameter BlockErase Time	2	1	4	0.8	4	s
	t_{WHQV3} / t_{EHQV3}	32-KW Main Block Erase Time	2	1.5	5	1.1	5	s
	t_{WHRH1} / t_{EHRH1}	Program Suspend Latency		5	10	5	10	µs
	t_{WHRH2} / t_{EHRH2}	Erase Suspend Latency		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.

9.0 Ordering Information



Valid Combinations (All Extended Temperature)

60-Ball 7x8 matrix µBGA CSP(1)	
GT28F320D18B110	GT28F320D18B120

NOTE:
1. The 60-Ball (7x8 matrix with 4 support balls) µBGA package top side mark reads F320D18. All product shipping boxes or trays provide the correct information regarding bus architecture.

10.0 Additional Information(1,2)

Order Number	Document/Tool
210830	*Flash Memory Databook*
292215	*AP-657 Designing with the Advanced+ Boot Block Flash Memory Architecture*
292204	*AP-646 Common Flash Interface (CFI) and Command Sets*
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*
Note 2	*µBGA* Package Mechanical and Shipping Media Specification*
297846	*Comprehensive User's Guide for µBGA* Packages*

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://intel.com/design/flash for technical documentation and tools.

APPENDIX A: Common Flash Interface

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.

A.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 device, 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 13. Summary of Query Structure Output As a Function of Device and Mode

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

Table 14. Example of Query Structure Output

Word Addressing			Byte Addressing		
Offset	**Hex Code**	**Value**	**Offset**	**Hex Code**	**Value**
A_{max}–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		
...	...	...	...		

A.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 15. 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

NOTES:
1. Refer to the *Query Structure Output* section and offset 28h for the detailed definition of offset address as a function of device bus width and mode.
2. BA = Block Address beginning location (i.e., 08000h is block 1’s beginning location when the block size is 32 Kword).
3. Offset 15 defines “P” which points to the *Primary Intel-Specific Extended Query* Table.

A.3 Block Lock Status

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 16. Block Lock Status Register

Offset	Length	Description	Address	Value
(BA+2)h(1)	1	Block Lock Status Register	BA+2:	--00 or --01
		BSR.0 Block Lock Status 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

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

A.4 CFI Query Identification String

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

Table 17. CFI Identification

Offset	Length	Description	Address	Hex Code	Value
10h	3	Query-unique ASCII string "QRY"	10: 11: 12:	--51 --52 --59	Q R Y
13h	2	Primary vendor command set and control interface ID code 16-bit ID code for vendor-specified algorithms	13: 14:	--03 --00	
15h	2	Extended Query Table primary algorithm address	15: 16:	--39 --00	
17h	2	Alternate vendor command set and control interface ID code 0000h means no second vendor-specified algorithm exists	17: 18:	--00 --00	
19h	2	Secondary algorithm Extended Query Table address 0000h means none exists	19: 1A:	--00 --00	

A.5 System Interface Information

Table 18. System Interface Information

Offset	Length	Description	Address	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:	--17	1.7 V
1Ch	1	V_{CC} logic supply maximum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 BCD volts	1C:	--19	1.9 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	n/a
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	n/a
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	n/a
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	n/a

A.6 Device Geometry Information

This field provides critical details of the flash device geometry.

Table 19. Device Geometry Information

Offset	Length	Description	Code		
27h	1	“n” such that device size = 2^n in number of bytes	27:	See *Device Geometry Definition* Table	
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:	--03	3

Table 19. Device Geometry Information

Offset	Length	Description	Code	
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:	See *Device Geometry Definition* Table
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:	See *Device Geometry Definition* Table
35h	4	Erase Block Region 3 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	35: 36: 37: 38:	See *Device Geometry Definition* Table

Device Geometry Definition

Address	32 Mbit		64 Mbit (info only)	
	–B	–T	–B	–T
27:	--16	--16	--17	--17
28:	--01	--01	--01	--01
29:	--00	--00	--00	--00
2A:	--00	--00	--00	--00
2B:	--00	--00	--00	--00
2C:	--03	--03	--03	--03
2D:	--07	--2F	--07	--5F
2E:	--00	--00	--00	--00
2F:	--20	--00	--20	--00
30:	--00	--01	--00	--01
31:	--0E	--0E	--1E	--1E
32:	--00	--00	--00	--00
33:	--00	--00	--00	--00
34:	--01	--01	--01	--01
35:	--2F	--07	--5F	--07
36:	--00	--00	--00	--00
37:	--00	--20	--00	--20
38:	--01	--00	--01	--00

A.6 Intel-Specific Extended Query Table

Table 20. Primary-Vendor Specific Extended Query

Offset[1] P = 39H	Length	Description (Optional Flash Features and Commands)	Address	Hex Code	Value
(P+0)h	3	Primary extended query table	39:	--50	"P"
(P+1)h		Unique ASCII string "PRI"	3A:	--52	"R"
(P+2)h			3B:	--49	"I"
(P+3)h	1	Major version number, ASCII	3C:	--31	"1"
(P+4)h	1	Minor version number, ASCII	3D:	--33	"3"
(P+5)h	4	Optional feature and command support (1=yes, 0=no)	3E:	--E6	
(P+6)h		bits 10–31 are reserved; undefined bits are "0." If bit 31 is "1"	3F:	--03	
(P+7)h		then another 31 bit field of optional features follows at the end of	40:	--00	
(P+8)h		the bit–30 field.	41:	--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 = 1	Yes	
		bit 8 Synchronous read supported	bit 8 = 1	Yes	
		bit 9 Simultaneous operations supported	bit 9 = 1	Yes	
(P+9)h	1	Supported functions after suspend: Read Array, Status, Query Other supported operations are: bits 1–7 reserved; undefined bits are "0"	42:	--01	
		bit 0 Program supported after erase suspend	bit 0 = 1	Yes	
(P+A)h	2	Block status register mask	43:	--03	
(P+B)h		bits 2–15 are *Reserved*; undefined bits are "0"	44:	--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	45:	--18	1.8 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	46:	--C0	12.0 V

Table 21. Protection Register Information

Offset(1) P = 39H	Length	Description (Optional Flash Features and Commands)	Address	Hex Code	Value
(P+E)h	1	Number of Protection register fields in JEDEC ID space “00h,” indicates that 256 protection bytes are available	47:	--01	01
(P+F)h	4	Protection Field 1: Protection Description	48:	--80	80h
(P+10)h		This field describes user-available One Time Programmable (OTP) protection register bytes. Some are	49:	--00	00h
(P+11)h		pre-programmed with device-unique serial numbers. Others are user programmable. Bits 0–15 point to the protection register	4A:	--03	8 byte
(P+12)h		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 2^n = factory pre-programmed bytes bits 24–31 = “n” such that 2^n = user programmable bytes	4B:	--03	8 byte

Table 22. Burst Read Information

Offset(1) P = 39H	Length	Description (Optional Flash Features and Commands)	Address	Hex Code	Value
(P+13)h	1	Page-mode read capability bits 0–7 = “n” such that 2^n HEX value represents the number of read-page bytes. See offset 28h for device word width to determine page-mode data output width. 00h indicates no read page buffer.	4C:	--03	8 byte
(P+14)h	1	Number of synchronous mode read configuration fields that follow. 00h indicates no burst capability.	4D:	--03	3
(P+15)h	1	Synchronous mode read capability configuration 1 bits 3–7 = *Reserved* bits 0–2 “n” such that 2^{n+1} HEX value represents the maximum number of continuous synchronous reads when the device is configured for its maximum word width. A value of 07h indicates that the device is capable of continuous linear bursts that will output data until the internal burst counter reaches the end of the device’s burstable address space. This field’s 3-bit value can be written directly to the read configuration register bits 0–2 if the device is configured for its maximum word width. See offset 28h for word width to determine the burst data output width.	4E:	--01	4
(P+16)h	1	Synchronous mode read capability configuration 2	4F:	--02	8
(P+17)h	1	Synchronous mode read capability configuration 3	50:	--07	Cont

Table 23. Partition and Erase-Block Region Information

Bottom Offset(1) P = 39H	Top Offset(1) P = 39H	Description (Optional Flash Features and Commands)	See Table 24		
			Length	Address	
				Bottom	Top
(P+18)h	(P+18)h	Number of device hardware-partition regions within the device x = 0: a single hardware partition device (no fields follow) x specifies the number of device partition regions containing one or more contiguous erase block regions.	1	51:	51:

Partition Region 1 Information

Bottom Offset(1) P = 39H	Top Offset(1) P = 39H	Description (Optional Flash Features and Commands)	See Table 24		
			Length	Address	
				Bottom	Top
(P+19)h (P+1A)h	(P+19)h (P+1A)h	Number of identical partitions within the partition region	2	52: 53:	52: 53:
(P+1B)h	(P+1B)h	Simultaneous program and erase operations allowed in other partitions while this partition is in read mode bits 0–3 = number of simultaneous program operations bits 4–7 = number of simultaneous erase operations	1	54:	54:
(P+1C)h	(P+1C)h	Simultaneous program and erase operations allowed in other partitions while this partition is in program mode bits 0–3 = number of simultaneous program operations bits 4–7 = number of simultaneous erase operations	1	55:	55:
(P+1D)h	(P+1D)h	Simultaneous program and erase operations allowed in other partitions while this partition is in erase mode bits 0–3 = number of simultaneous program operations bits 4–7 = number of simultaneous erase operations	1	56:	56:
(P+1E)h	(P+1E)h	Partitions' erase block regions in this Partition Region. 1. x = 0 = no erase blocking; the Partition Region erases in “bulk” 2. x specifies the number of erase block regions containing 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)	1	57:	57:
(P+1F)h (P+20)h (P+21)h (P+22)h	(P+1F)h (P+20)h (P+21)h (P+22)h	Partition Region 1 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	4	58: 59: 5A: 5B:	58: 59: 5A: 5B:
(P+23)h (P+24)h	(P+23)h (P+24)h	Partition 1 (Erase Region 1) Minimum block erase cycles x 1000	2	5C: 5D:	5C: 5D:
(P+25)h	(P+25)h	Partition 1 (erase region 1) bits per cell; internal error correction bits 0–3 = bits per cell in erase region bit 4 = reserved for “internal ECC used” (1=yes, 0=no) bits 5–7 = reserved for future use	1	5E:	5E:
(P+26)h	(P+26)h	Partition 1 (Erase Region 1) page mode and synchronous mode capabilities as defined in Table 19. bit 0 = page mode host reads permitted (1=yes, 0=no) bit 1 = synchronous host reads permitted (1=yes, 0=no) bit 2 = synchronous host writes permitted (1=yes, 0=no) bits 3–7 = reserved for future use	1	5F:	5F:
(P+27)h (P+28)h (P+29)h (P+2A)h		Partition Region 1 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 (bottom parameter device only)	4	60: 61: 62: 63:	

Partition Region 1 Information

Bottom Offset(1) P = 39H	Top Offset(1) P = 39H	Description (Optional Flash Features and Commands)	See Table 24		
			Length	Address	
				Bottom	Top
(P+2B)h (P+2C)h		Partition 1 (Erase Region 2) minimum block erase cycles x 1000 (bottom parameter device only)	2	64: 65:	
(P+2D)h		Partition 1 (Erase Region 2) bits per cell bottom parameter device only) bits 0–3 = bits per cell in erase region bit 4 = reserved for “internal ECC used” (1=yes, 0=no) bits 5–7 = reserved for future use	1	66:	
(P+2E)h		Partition 1 (Erase Region 2) page mode and synchronous mode capabilities defined in Table 19 (bottom parameter device only) bit 0 = page mode host reads permitted (1=yes, 0=no) bit 1 = synchronous host reads permitted (1=yes, 0=no) bit 2 = synchronous host writes permitted (1=yes, 0=no) bits 3–7 = reserved for future use	1	67:	

Partition Region 2 Information

Bottom Offset(1) P = 39H	Top Offset(1) P = 39H	Description (Optional Flash Features and Commands)	See Table 24		
			Length	Address	
				Bottom	Top
(P+2F)h (P+30)h	(P+27)h (P+28)h	Number of identical partitions within the partition region	2	68: 69:	60: 61:
(P+31)h	(P+29)h	Simultaneous program and erase operations allowed in other partitions while this partition is in read mode bits 0–3 = number of simultaneous program operation bits 4–7 = number of simultaneous erase operations	1	6A:	62:
(P+32)h	(P+2A)h	Simultaneous program and erase operations allowed in other partitions while this partition is in program mode bits 0–3 = number of simultaneous program operation bits 4–7 = number of simultaneous erase operations	1	6B:	63:
(P+33)h	(P+2B)h	Simultaneous program and erase operations allowed in other partitions while this partition is in erase mode bits 0–3 = number of simultaneous program operations bits 4–7 = number of simultaneous erase operations	1	6C:	64:
(P+2F)h	(P+27)h	Number of identical partitions within the partition region	2	68:	60:
(P+34)h	(P+2C)h	Partitions' erase block regions in this Partition Region. 1. x = 0 = no erase blocking; the Partition Region erases in “bulk” 2. x specifies the number of erase block regions containing 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)	1	6D:	65:
(P+35)h (P+36)h (P+37)h (P+38)h	(P+2D)h (P+2E)h (P+2F)h (P+30)h	Partition Region 2 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	4	6E: 6F: 70: 71:	66: 67: 68: 69:
(P+39)h (P+3A)h	(P+31)h (P+32)h	Partition 2 (Erase Region 1) minimum block erase cycles x 1000	2	72: 73:	6A: 6B:

Partition Region 2 Information

Bottom Offset(1) P = 39H	Top Offset(1) P = 39H	Description (Optional Flash Features and Commands)	See Table 24		
			Length	Address	
				Bottom	Top
(P+3B)h	(P+33)h	Partition 2 (Erase Region 1) bits per cell bits 0–3 = bits per cell in erase region bit 4 = reserved for “internal ECC used” (1=yes, 0=no) bits 5–7 = reserved for future use	1	74:	6C:
(P+3C)h	(P+34)h	Partition 2 (Erase Region 1) page mode and synchronous mode capabilities as defined in Table 19. bit 0 = page mode host reads permitted (1=yes, 0=no) bit 1 = synchronous host reads permitted (1=yes, 0=no) bit 2 = synchronous host writes permitted (1=yes, 0=no) bits 3–7 = reserved for future use	1	75:	6D:
	(P+35)h (P+36)h (P+37)h (P+38)h	Partition Region 2 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 (top parameter device only)	4		6E: 6F: 70: 71:
	(P+39)h (P+3A)h	Partition 2 (Erase Region 2) minimum block erase cycles x 1000 (top parameter device only)	2		72: 73:
	(P+3B)h	Partition 2 (Erase Region 2) bits per cell (top parameter only) bits 0–3 = bits per cell in erase region bit 4 = reserved for “internal ECC used” (1=yes, 0=no) bits 5–7 = reserved for future use	1		74:
	(P+3C)h	Partition 2 (Erase Region 2) page mode and synchronous mode capabilities as defined in Table 19. (top parameter only bit 0 = page mode host reads permitted (1=yes, 0=no) bit 1 = synchronous host reads permitted (1=yes, 0=no) bit 2 = synchronous host writes permitted (1=yes, 0=no) bits 3–7 = reserved for future use	1		75:
(P+3D)h	(P+3D)h	Features Space definitions (*reserved for future use*)	TBD	76:	76:
(P+3E)h	(P+3E)h	Reserved for future use	Rsv'd	77:	77:

NOTES:
1. The variable P is a pointer which is defined at CFI offset 15h.
2. For a 1-Mb 1.8 Volt Dual-Plane Flash memory, z_1 = 0100h = 256 ⇒ 256 * 256 = 64K, y_1 = 17h = 23d ⇒ y1+1 = 24 ⇒ 24 * 64K = 1½MB ⇒ Partition 2’s offset is 0018 0000h bytes (000C 0000h words).

Table 24. Partition and Erase-Block Region Information

Address	32 Mbit		64 Mbit (Info only)	
	–B	–T	–B	–T
51:	--02	--02	--02	--02
52:	--01	--01	--01	--01
53:	--00	--00	--00	--00
54:	--01	--01	--01	--01
55:	--00	--00	--00	--00
56:	--00	--00	--00	--00
58:	--07	--2F	--07	--5F
59:	--00	--00	--00	--00
5A:	--20	--00	--20	--00
5B:	--00	--01	--00	--01
5C:	--64	--64	--64	--64
5D:	--00	--00	--00	--00
5E:	--01	--01	--01	--01
5F:	--00	--03	--00	--03
60:	--0E	--01	--1E	--01
61:	--00	--00	--00	--00
62:	--00	--01	--00	--01
63:	--01	--00	--01	--00
64:	--64	--00	--64	--00
65:	--00	--02	--00	--02
66:	--01	--0E	--01	--1E
67:	--03	--00	--03	--00
68:	--01	--00	--01	--00
69:	--00	--01	--00	--01
6A:	--01	--64	--01	--64
6B:	--00	--00	--00	--00
6C:	--00	--01	--00	--01
6D:	--01	--03	--01	--03
6E:	--2F	--07	--5F	--07
6F:	--00	--00	--00	--00
70:	--00	--20	--00	--20
71:	--01	--00	--01	--00
72:	--64	--64	--64	--64
73:	--00	--00	--00	--00
74:	--01	--01	--01	--01
75:	--03	--00	--03	--00

APPENDIX B: Protection Register Addressing

Word	Use	ID Offset	A7	A6	A5	A4	A3	A2	A1	A0
LOCK	Both	0080h	1	0	0	0	0	0	0	0
0	Factory	0081h	1	0	0	0	0	0	0	1
1	Factory	0082h	1	0	0	0	0	0	1	0
2	Factory	0083h	1	0	0	0	0	0	1	1
3	Factory	0084h	1	0	0	0	0	1	0	0
4	User	0085h	1	0	0	0	0	1	0	1
5	User	0086h	1	0	0	0	0	1	1	0
6	User	0087h	1	0	0	0	0	1	1	1
7	User	0088h	1	0	0	0	1	0	0	0

NOTE: Upper addresses [A20:A8] should be set to zero.