1.8 Volt Intel StrataFlash® Wireless Memory with 3.0 Volt I/O (L30)

28F640L30, 28F128L30, 28F256L30

Datasheet

Product Features

- **High performance Read-While-Write/Erase**
  - 90 ns initial access
  - 50MHz with zero wait state, 17 ns clock-to-data output synchronous-burst mode
  - 25 ns asynchronous-page mode
  - 4-, 8-, 16-, and continuous-word burst mode
  - Burst suspend
  - Programmable WAIT configuration
  - Buffered Enhanced Factory Programming (Buffered EFP): 3.5 µs/byte (Typ)
  - 1.8 V low-power buffered and non-buffered programming @ 10 µs/byte (Typ)
- **Architecture**
  - Asymmetrically-blocked architecture
  - Multiple 8-Mbit partitions
  - Four 16K-Word parameter blocks: top or bottom configurations
  - 64K-Word main blocks
  - Dual-operation: Read-While-Write (RWW) or Read-While-Erase (RWE)
  - Status register for partition and device status
- **Power**
  - 1.7 V - 2.0 V V_{CC} operation
  - I/O voltage: 2.2 V - 3.3 V
  - Standby current: 30 µA (Typ)
  - 4-Word synchronous read current: 17 mA (Typ) @ 54 MHz
  - Automatic Power Savings (APS) mode
- **Software**
  - 20 µs (Typ) program suspend
  - 20 µs (Typ) erase suspend
  - Intel® Flash Data Integrator (FDI) optimized
  - Basic Command Set (BCS) and Extended Command Set (ECS) compatible
  - Common Flash Interface (CFI) capable
- **Security**
  - OTP space:
    - 64 unique device identifier bits
    - 64 user-programmable OTP bits
    - Additional 2048 user-programmable OTP bits
  - Absolute write protection: V_{PP} = GND
  - Power-transition erase/program lockout
  - Individual zero-latency block locking
  - Individual block lock-down
- **Quality and Reliability**
  - Expanded temperature: –25° C – +85° C
  - Minimum 100,000 erase cycles per block
  - ETOX™ VIII process technology (0.13 µm)
- **Density and Packaging**
  - 64-, 128- and 256-Mbit density in VF BGA packages
  - 16-bit wide data bus

The 1.8 Volt Intel StrataFlash® wireless memory with 3-Volt I/O product is the latest generation of Intel StrataFlash® memory devices featuring flexible, multiple-partition, dual operation. It provides high performance asynchronous read mode and synchronous-burst read mode using 1.8 Volt low-voltage, multi-level cell (MLC) technology.

The multiple-partition architecture enables background programming or erasing to occur in one partition while code execution or data reads take place in another partition. This dual-operation architecture also allows two processors to interleave code operations while program and erase operations take place in the background. 8-Mbit partitions allow system designers to choose the size of the code and data segments.

The 1.8 Volt Intel StrataFlash® wireless memory with 3-Volt I/O device is manufactured using Intel 0.13 µm ETOX™ VIII process technology. It is available in industry-standard chip scale packaging.

.

Notice: This document contains information on products in the design phase of development. The information here is subject to change without notice. Do not finalize a design with this information.

Order Number: 251903
October 2002

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. Intel products are not intended for use in medical, life saving, or life sustaining applications.

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

This document contains information on products in the design phase of development. The information here is subject to change without notice. Do not finalize a design with this information.

The 1.8 Volt Intel StrataFlash® Wireless Memory with 3.0 Volt I/O datasheet may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

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

Copies of documents which have an ordering number and are referenced in this document, or other Intel literature may be obtained by calling 1-800-548-4725 or by visiting Intel's website at http://www.intel.com.

Copyright © 2002, Intel Corporation

* Other names and brands may be claimed as the property of others.

Contents

1.0 Introduction 7

1.1 Nomenclature 7
1.2 Acronyms 7
1.3 Conventions 8

2.0 Device Description 9

2.1 Product Overview 9
2.2 Ballout Diagrams 10
2.3 Signal Descriptions 11
2.4 Memory Map 12

3.0 Device Operations 14

3.1 Bus Operations 14
3.1.1 Reads 14
3.1.2 Writes 14
3.1.3 Output Disable 14
3.1.4 Standby 15
3.1.5 Reset 15
3.2 Device Commands 15
3.3 Command Definitions 17

4.0 Read Operations 19

4.1 Asynchronous Page-Mode Read 19
4.2 Synchronous Burst-Mode Read 20
4.2.1 Burst Suspend 20
4.3 Read Configuration Register (RCR) 21
4.3.1 Read Mode 22
4.3.2 Latency Count 22
4.3.3 WAIT Polarity 23
4.3.3.1 WAIT Signal Function 24
4.3.4 Data Hold 24
4.3.5 WAIT Delay 25
4.3.6 Burst Sequence 25
4.3.7 Clock Edge 26
4.3.8 Burst Wrap 26
4.3.9 Burst Length 26

5.0 Programming Operations 27

5.1 Word Programming 27
5.1.1 Factory Word Programming 28
5.2 Buffered Programming 28
5.3 Buffered Enhanced Factory Programming 29
5.3.1 Buffered EFP Requirements and Considerations 29
5.3.2 Buffered EFP Setup Phase 30
5.3.3 Buffered EFP Program/Verify Phase 30
5.3.4 Buffered EFP Exit Phase 31
5.4 Program Suspend 31

5.5 Program Resume 31
5.6 Program Protection 32

6.0 Erase Operations 33

6.1 Block Erase 33
6.2 Erase Suspend 33
6.3 Erase Resume 34
6.4 Erase Protection 34

7.0 Security Modes 35

7.1 Block Locking 35
7.1.1 Lock Block 35
7.1.2 Unlock Block 35
7.1.3 Lock-Down Block 35
7.1.4 Block Lock Status 36
7.1.5 Block Locking During Suspend 36
7.2 Protection Registers 37
7.2.1 Reading the Protection Registers 38
7.2.2 Programming the Protection Registers 39
7.2.3 Locking the Protection Registers 39

8.0 Dual-Operation Considerations 40

8.1 Memory Partitioning 40
8.2 Read-While-Write Command Sequences 40
8.2.1 Simultaneous Operation Details 41
8.2.2 Synchronous and Asynchronous Read-While-Write Characteristics and Waveforms41
8.2.3 Read Operation During Buffered Programming Flowchart 42
8.3 Simultaneous Operation Restrictions 42

9.0 Special Read States 43

9.1 Read Status Register 43
9.1.1 Clear Status Register 44
9.2 Read Device Identifier 44
9.3 CFI Query 45

10.0 Power and Reset 46

10.1 Power-Up/Down Characteristics 46
10.2 Power Supply Decoupling 46
10.3 Automatic Power Saving (APS) 46
10.4 Reset Characteristics 46

11.0 Thermal and DC Characteristics 48

11.1 Absolute Maximum Ratings 48
11.2 Operating Conditions 48
11.3 DC Current Characteristics 49
11.4 DC Voltage Characteristics 50

12.0 AC Characteristics 51

12.1 AC Read Specifications (VCCQ = 2.2 V – 3.3 V) 51
12.2 AC Write Specifications 56

12.3 Program and Erase Characteristics 60
12.4 Reset Specifications 60
12.5 AC Test Conditions 61
12.6 Capacitance 62

Appendix A Write State Machine (WSM) 63

Appendix B Flowcharts 70

Appendix C Common Flash Interface 79

Appendix D Mechanical Information 89

Appendix E Additional Information 91

Appendix F Ordering Information 92

Revision History

Revision Date	Revision	Description
10/14/02	-001	Initial Release

1.0 Introduction

This document provides information about the 1.8 Volt Intel StrataFlash® wireless memory with 3-Volt I/O (L30) device. This document describes the L30 flash memory device features, operation, and specifications.

1.1 Nomenclature

1.8 V: V_{CC} voltage range of 1.7 V – 2.0 V (except where noted)

3.0 V Range: V_{CCQ} voltage range of 2.2 V – 3.3 V

VPP = 9.0 V: V_{PP} voltage range of 8.5 V – 9.5 V

Block: A group of bits, bytes or words within the flash memory array that erase simultaneously when the Erase command is issued to the device. The L30 flash memory device has two block sizes: 16K-Word, and 64K-Word.

Main block: An array block that is usually used to store code and/or data. Main blocks are larger than parameter blocks.

Parameter block: An array block that is usually used to store frequently changing data or small system parameters that traditionally would be stored in EEPROM.

Top parameter device: Previously referred to as a top-boot device, a device with its parameter partition located at the highest physical address of its memory map. Parameter blocks within a parameter partition are located at the highest physical address of the parameter partition.

Bottom parameter device: Previously referred to as a bottom-boot device, a device with its parameter partition located at the lowest physical address of its memory map. Parameter blocks within a parameter partition are located at the lowest physical address of the parameter partition.

Partition: A group of blocks that share common program/erase circuitry. Blocks within a partition also share a common status register. If any block within a partition is being programmed or erased, only status register data (rather than array data) is available when any address within that partition is read.

Main partition: A partition containing only main blocks.

Parameter partition: A partition containing parameter blocks and main blocks.

1.2 Acronyms

CUI: Command User Interface

MLC: Multi-Level Cell

OTP: One-Time Programmable

PLR: Protection Lock Register

PR: Protection Register

RCR: Read Configuration Register

RFU: Reserved for Future Use

SR: Status Register

WSM: Write State Machine

1.3 Conventions

VCC: signal or voltage connection
V_{CC}: signal or voltage level

0x: hexadecimal number prefix
0b: binary number prefix

SR[4]: Denotes an individual register bit.
A[15:0]: Denotes a group of similarly named signals, such as address or data bus.
A5: Denotes one element of a signal group membership, such as an address.

bit: binary unit
byte: eight bits
word: two bytes, or sixteen bits

Kbit: 1024 bits
KByte: 1024 bytes
KWord: 1024 words
Mbit: 1,048,576 bits
MByte: 1,048,576 bytes
MWord: 1,048,576 words

2.0 Device Description

This section provides an overview of the features and capabilities of the 1.8 Volt Intel StrataFlash® wireless memory with 3-Volt I/O (L30) device.

2.1 Product Overview

The 1.8 Volt Intel StrataFlash® wireless memory with 3-Volt I/O (L30) device provides read-while-write and read-while-erase capability with density upgrades through 256-Mbit. This family of devices provides high performance at low voltage on a 16-bit data bus. Individually erasable memory blocks are sized for optimum code and data storage.

Each device density contains one parameter partition and several main partitions. The flash memory array is grouped into multiple 8-Mbit partitions. By dividing the flash memory into partitions, program or erase operations can take place at the same time as read operations.

Although each partition has write, erase and burst read capabilities, simultaneous operation is limited to write or erase in one partition while other partitions are in read mode. The L30 flash memory device allows burst reads that cross partition boundaries. User application code is responsible for ensuring that burst reads don't cross into a partition that is programming or erasing.

Upon initial power up or return from reset, the device defaults to asynchronous page-mode read. Configuring the Read Configuration Register enables synchronous burst-mode reads. In synchronous burst mode, output data is synchronized with a user-supplied clock signal. A WAIT signal provides easy CPU-to-flash memory synchronization.

In addition to the enhanced architecture and interface, the device incorporates technology that enables fast factory program and erase operations. Designed for low-voltage systems, the L30 flash memory device supports read operations with V_{CC} at 1.8 V, and erase and program operations with V_{PP} at 1.8 V or 9.0 V. Buffered Enhanced Factory Programming (Buffered EFP) provides the fastest flash array programming performance with V_{PP} at 9.0 Volt, which increases factory throughput. With V_{PP} at 1.8 V, VCC and VPP can be tied together for a simple, ultra low power design. In addition to voltage flexibility, a dedicated V_{PP} connection provides complete data protection when V_{PP} is less than V_{PPLK}.

A Command User Interface (CUI) is the interface between the system processor and all internal operations of the device. An internal Write State Machine (WSM) automatically executes the algorithms and timings necessary for block erase and program. A Status Register indicates erase or program completion and any errors that may have occurred.

An industry-standard command sequence invokes program and erase automation. Each erase operation erases one block. The Erase Suspend feature allows system software to pause an erase cycle to read or program data in another block. Program Suspend allows system software to pause programming to read other locations. Data is programmed in word increments (x16).

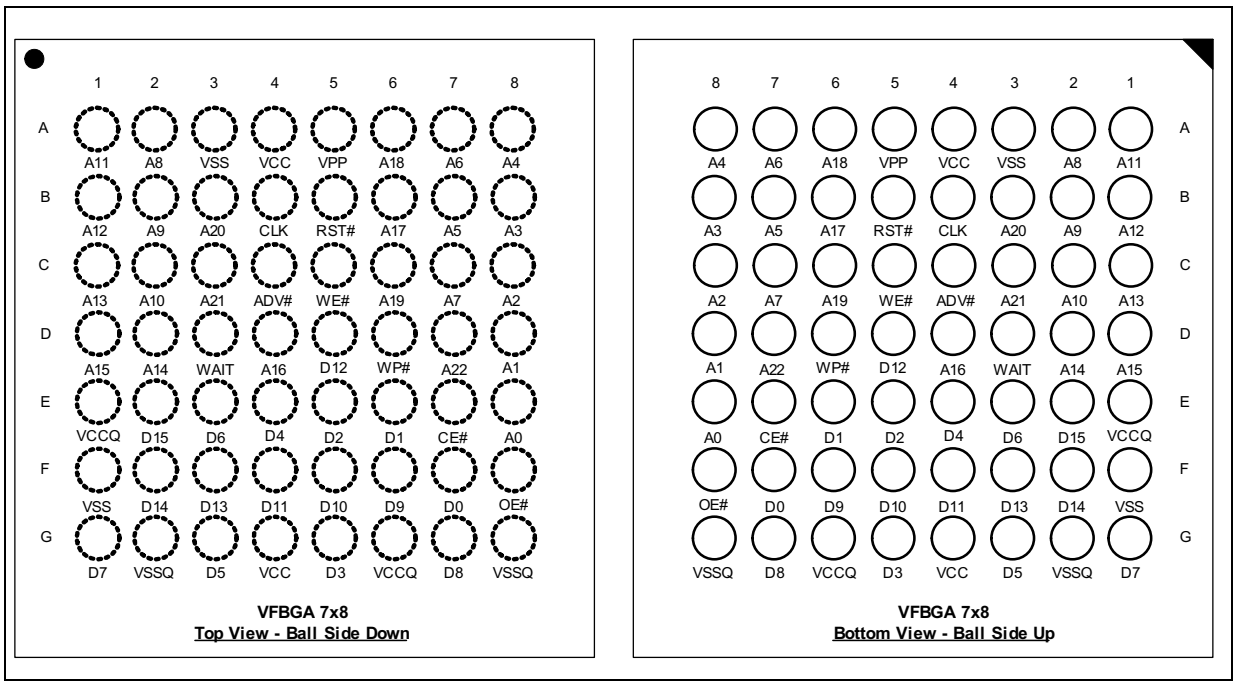
The L30 flash memory device offers power savings through Automatic Power Savings (APS) mode and standby mode. The device automatically enters APS following read-cycle completion. Standby is initiated when the system deselects the device by deasserting CE# or by asserting RST#. Combined, these features can significantly reduce power consumption.

The L30 flash memory device's protection register allows unique flash device identification that can be used to increase system security. Also, the individual Block Lock feature provides zero-latency block locking and unlocking.

2.2 Ballout Diagrams

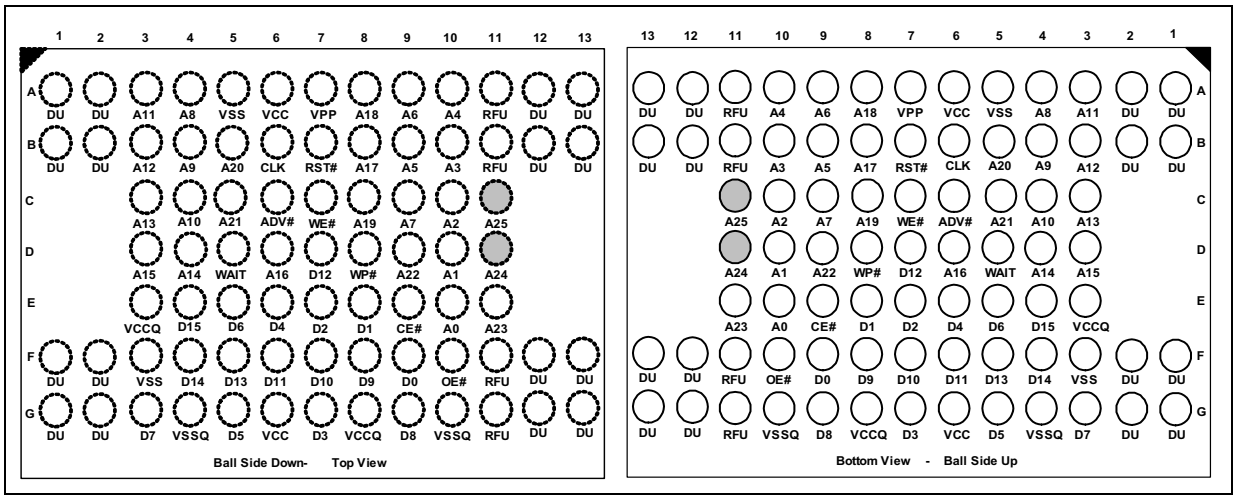
The L30 flash memory device is available in a VF BGA package with 0.75 mm ball-pitch. Figure 1 shows the ballout for the 64-Mbit and 128-Mbit devices in the 56-ball VF BGA package with a 7 x 8 active-ball matrix. Figure 2 shows the device ballout for the 256-Mbit device in the 63-ball VF BGA package with a 7 x 9 active-ball matrix. Both package densities are ideal for space-constrained board applications

Figure 1. 7x8 Active-Ball Matrix for 64-, and 128-Mbit Densities in VF BGA Packages



NOTE: On lower-density devices, upper-address balls can be treated as NC. (e.g., for 64-Mbit density, A22 will be NC)

Figure 2. 7x9 Active-Ball Matrix for 256-Mbit Density in VF BGA Package



NOTE: On lower density devices upper address balls can be treated as RFUs. (A24 is for 512Mb and A25 is for 1Gb densities.) All ball locations are populated.

2.3 Signal Descriptions

Table 1 describes the active signals used on the L30 flash memory device.

Table 1. Signal Descriptions

Symbol	Type	Name and Function
A[MAX:0]	In	**ADDRESS:** Device address inputs. 64-Mbit: A[21:0]; 128-Mbit: A[22:0]; 256-Mbit: A[23:0].
D[15:0]	In/Out	**DATA INPUT/OUTPUTS:** Inputs data and commands during write cycles; outputs data during memory, Status Register, Protection Register, and Read Configuration Register reads. Data balls float when the CE# or OE# are de-asserted. Data is internally latched during writes.
ADV#	In	**ADDRESS VALID:** Active-low input. During synchronous read operations, addresses are latched on the rising edge of ADV#, or on the next valid CLK edge with ADV# low, whichever occurs first.
CE#	In	**CHIP ENABLE:** Active-low input. CE#-low selects the device. CE#-high deselects the device, placing it in standby, with D[15:0] and WAIT in High-Z.
CLK	In	**CLOCK:** Synchronizes the device with the system's bus frequency in synchronous-read mode and increments the internal address generator. During synchronous read operations, addresses are latched on the rising edge of ADV#, or on the next valid CLK edge with ADV# low, whichever occurs first.
OE#	In	**OUTPUT ENABLE:** Active-low input. OE#-low enables the device's output data buffers during read cycles. OE#-high places the data outputs in High-Z.
RST#	In	**RESET:** Active-low input. 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.
WAIT	Out	**WAIT:** Wait is driven when CE# is asserted. RCR[10] [WP] determines the WAIT -asserted logic level. • In synchronous array read modes, WAIT indicates invalid data when asserted and valid data when de-asserted. • In synchronous non-array read modes, asynchronous page mode, and all write modes, WAIT is asserted.
WE#	In	**WRITE ENABLE:** Active-low input. WE# controls writes to the device. Address and data are latched on the rising edge of WE#.
WP#	In	**WRITE PROTECT:** Active-low input. WP#-low enables the lock-down mechanism. Blocks in lock-down cannot be unlocked with the Unlock command. WP#-high overrides the lock-down function enabling blocks to be erased or programmed using software commands.
VPP	Pwr	**ERASE/ PROGRAM POWER:** Valid V_{PP} voltages on this ball allow block erase and program functions. Flash memory array contents cannot be altered when $V_{PP} \le V_{PPLK}$. Block erase and program at invalid V_{PP} voltages should not be attempted.
VCC	Pwr	**DEVICE CORE POWER SUPPLY:** Core (logic) source voltage. Writes to the flash array are inhibited when $V_{CC} \le V_{LKO}$. Operations at invalid V_{CC} voltages should not be attempted.
VCCQ	Pwr	**OUTPUT POWER SUPPLY:** Output-driver source voltage. This ball can be tied directly to V_{CC} if operating within V_{CC} range.
VSS	Pwr	**GROUND:** Ground reference for device logic voltages. Connect to system ground.
VSSQ	Pwr	**GROUND:** Ground reference for device output voltages. Connect to system ground.
DU	-	**DON'T USE:** Do not use this ball. This ball should not be connected to any power supplies, signals or other balls, and must be left floating.
NC	-	**NO CONNECT:** No internal connection; can be driven or floated.
RFU	-	**RESERVED for FUTURE USE:** Reserved by Intel for future device functionality and enhancement.

2.4 Memory Map

The L30 flash memory array is divided into multiple 8-Mbit partitions. Each device density contains one parameter partition and several main partitions. The 8-Mbit top or bottom parameter partition contains four 16K-Word blocks and seven 64K-Word blocks. There are multiple 8-Mbit main partitions. The 8-Mbit main partitions each contains eight 64K-Word blocks.

The device multi-partition architecture is divided as follow:

- **The 64-Mbit device** contains eight partitions: one 8-Mbit parameter partition, **seven** 8-Mbit main partitions.
- **The 128-Mbit device** contains sixteen partitions: one 8-Mbit parameter partition, **fifteen** 8-Mbit main partitions.
- **The 256-Mbit device** contains thirty-two partitions: one 8-Mbit parameter partition, **thirty-one** 8-Mbit main partitions.

Table 2. Top Parameter Memory Map

		Size (KW)	Blk	64-Mbit
8-Mbit Parameter Partition	One Partition	16	66	3FC000-3FFFFF
		16	65	3F8000-3FBFFF
		16	64	3F4000-3F7FFF
		16	63	3F0000-3F3FFF
		64	62	3E0000-3EFFFF
		⋮	⋮	⋮
		64	56	380000-38FFFF
8-Mbit Main Partition	Seven Partitions	64	55	370000-37FFFF
		⋮	⋮	⋮
		64	0	000000-00FFFF

		Size (KW)	Blk	128-Mbit	Blk	256-Mbit
8-Mbit Parameter Partition	One Partition	16	130	7FC000-7FFFFF	258	FFC000-FFFFFF
		16	129	7F8000-7FBFFF	257	FF8000-FFBFFF
		16	128	7F4000-7F7FFF	256	FF4000-FF7FFF
		16	127	7F0000-7F3FFF	253	FF0000-FF3FFF
		64	126	7E0000-7EFFFF	254	FE0000-FEFFFF
		⋮	⋮	⋮	⋮	⋮
		64	120	780000-78FFFF	248	F80000-F8FFFF
8-Mbit Main Partitions	Fifteen Partitions	64	119	770000-77FFFF	247	F70000-F7FFFF
		⋮				
		64	0	000000-00FFFF	128	800000-80FFFF
	Sixteen Partitions	64			127	7F0000-7FFFFF
		⋮				
		64			0	000000-00FFFF

Table 3. Bottom Parameter Memory Map

		Size (KW)	Blk	64-Mbit	Blk	128-Mbit	Blk	256-Mbit
8-Mbit Main Partitions	Sixteen Partitions	64					258	FF0000-FFFFFF
		⋮					⋮	⋮
		64					131	800000-80FFFF
	Fifteen Partitions	64	66	3F0000-3FFFFF	130	7F0000-7FFFFF	130	7F0000-7FFFFF
		⋮	⋮	⋮	⋮	⋮	⋮	⋮
		64	11	080000-08FFFF	11	080000-08FFFF	11	080000-08FFFF
8-Mbit Parameter Partition	One Partition	64	10	070000-07FFFF	10	070000-07FFFF	10	070000-07FFFF
		⋮	⋮	⋮	⋮	⋮	⋮	⋮
		64	4	010000-01FFFF	4	010000-01FFFF	4	010000-01FFFF
		16	3	00C000-00FFFF	3	00C000-00FFFF	3	00C000-00FFFF
		16	2	008000-00BFFF	2	008000-00BFFF	2	008000-00BFFF
		16	1	004000-007FFF	1	004000-007FFF	1	004000-007FFF
		16	0	000000-003FFF	0	000000-003FFF	0	000000-003FFF

3.0 Device Operations

This section provides an overview of device operations. The system CPU provides control of all in-system read, write, and erase operations of the device via the system bus. The on-chip Write State Machine (WSM) manages all block-erase and word-program algorithms.

Device commands are written to the Command User Interface (CUI) to control all flash memory device operations. The CUI does not occupy an addressable memory location; it is the mechanism through which the flash device is controlled.

3.1 Bus Operations

CE#-low and RST# high enable device read operations. The device internally decodes upper address inputs to determine the accessed partition. ADV#-low opens the internal address latches. OE#-low activates the outputs and gates selected data onto the I/O bus.

In asynchronous mode, the address is latched when ADV# goes high or continuously flows through if ADV# is held low. In synchronous mode, the address is latched by the first of either the rising ADV# edge or the next valid CLK edge with ADV# low (WE# and RST# must be VIH; CE# must be VIL).

3.1.1 Reads

To perform a read operation, RST# and WE# must be deasserted while CE# and OE# are asserted. CE# is the device-select control. When asserted, it enables the flash memory device. OE# is the data-output control. When asserted, the addressed flash memory data is driven onto the I/O bus. See Section 4.0, “Read Operations” on page 19 for details on the available read modes, and see Section 9.0, “Special Read States” on page 43 for details regarding the available read states.

The Automatic Power Savings (APS) feature provides low power operation following reads during active mode. After data is read from the memory array and the address lines are quiescent, APS automatically places the device into standby. In APS, device current is reduced to I_{CCAPS} (see Section 11.3, “DC Current Characteristics” on page 49).

3.1.2 Writes

To perform a write operation, both CE# and WE# are asserted while RST# and OE# are deasserted. All device write operations are asynchronous, with CLK being ignored. During a write operation, address and data are latched on the rising edge of WE# or CE#, whichever occurs first. Table 4, “Command Bus Cycles” on page 16 shows the bus cycle sequence for each of the supported device commands, while Table 5, “Command Codes and Definitions” on page 17 describes each command. See Section 12.0, “AC Characteristics” on page 51 for signal-timing details.

Note: Write operations with invalid V_{CC} and/or V_{PP} voltages can produce spurious results and should not be attempted.

3.1.3 Output Disable

When OE# is deasserted, device outputs D[15:0] are disabled and placed in a high-impedance (High-Z) state.

3.1.4 Standby

When CE# is deasserted the device is deselected and placed in standby, substantially reducing power consumption. In standby, the data outputs are placed in High-Z, independent of the level placed on OE#. Standby current, I_{CCS}, is the average current measured over any 5 ms time interval, 5 µs after CE# is deasserted. During standby, average current is measured over the same time interval 5 µs after CE# is deasserted.

When the device is deselected (while CE# is deasserted) during a program or erase operation, it continues to consume active power until the program or erase operation is completed.

3.1.5 Reset

As with any automated device, it is important to assert RST# when the system is reset. When the system comes out of reset, the system processor attempts to read from the flash memory if it is the system boot device. If a CPU reset occurs with no flash memory reset, improper CPU initialization may occur because the flash memory may be providing status information rather than array data. Intel® flash memory devices allow proper CPU initialization following a system reset through the use of the RST# input. RST# should be controlled by the same low-true reset signal that resets the system CPU.

After initial power-up or reset, the device defaults to asynchronous Read Array, and the Status Register is set to 0x80. Asserting RST# de-energizes all internal circuits, and places the output drivers in High-Z. When RST# is asserted, the device shuts down the operation in progress, a process which takes a minimum amount of time to complete. When RST# has been deasserted, the device is reset to asynchronous Read Array state.

Note: If RST# is asserted during a program or erase operation, the operation is terminated and the memory contents at the aborted location (for a program) or block (for an erase) are no longer valid, because the data may have been only partially written or erased.

When returning from a reset (RST# deasserted), a minimum wait is required before the initial read access outputs valid data. Also, a minimum delay is required after a reset before a write cycle can be initiated. After this wake-up interval passes, normal operation is restored. See Section 12.0, “AC Characteristics” on page 51 for details about signal-timing.

3.2 Device Commands

Device operations are initiated by writing specific device commands to the Command User Interface (CUI). See Table 4, “Command Bus Cycles” on page 16.

Several commands are used to modify array data including Word Program and Block Erase commands. Writing either command to the CUI initiates a sequence of internally-timed functions that culminate in the completion of the requested task. However, the operation can be aborted by either asserting RST# or by issuing an appropriate suspend command.

Table 4. Command Bus Cycles

Mode	Command	Bus Cycles	First Bus Cycle			Second Bus Cycle		
			Oper	Addr[1]	Data[2]	Oper	Addr[1]	Data[2]
Read	Read Array	1	Write	PnA	0xFF			
	Read Device Identifier	≥ 2	Write	PnA	0x90	Read	PBA+IA	ID
	CFI Query	≥ 2	Write	PnA	0x98	Read	PnA+QA	QD
	Read Status Register	2	Write	PnA	0x70	Read	PnA	SRD
	Clear Status Register	1	Write	X	0x50			
Program	Word Program	2	Write	WA	0x40/ 0x10	Write	WA	WD
	Buffered Program[3]	≥ 2	Write	WA	0xE8	Write	WA	N - 1
	Buffered Enhanced Factory Program (Buffered EFP)[4]	> 2	Write	WA	0x80	Write	WA	0xD0
Erase	Block Erase	2	Write	BA	0x20	Write	BA	0xD0
Suspend	Program/Erase Suspend	1	Write	X	0xB0			
	Program/Erase Resume	1	Write	X	0xD0			
Block Locking/ Unlocking	Lock Block	2	Write	BA	0x60	Write	BA	0x01
	Unlock Block	2	Write	BA	0x60	Write	BA	0xD0
	Lock-down Block	2	Write	BA	0x60	Write	BA	0x2F
Protection	Program Protection Register	2	Write	PRA	0xC0	Write	PRA	PD
	Program Lock Register	2	Write	LRA	0xC0	Write	LRA	LRD
Configuration	Program Read Configuration Register	2	Write	RCD	0x60	Write	RCD	0x03

NOTES:

1. First command cycle address should be the same as the operation's target address.
   PnA = Address within the partition.
   PBA = Partition base address.
   IA = Identification code address offset.
   QA = CFI Query address offset.
   BA = Address within the block.
   WA = Word address of memory location to be written.
   PRA = Protection Register address.
   LRA = Lock Register address.
   X = Any valid address within the device.
2. ID = Identifier data.
   QD = Query data on D[15:0].
   SRD = Status Register data.
   WD = Word data.
   N = Word count of data to be loaded into the write buffer.
   PD = Protection Register data.
   PD = Protection Register data.
   LRD = Lock Register data.
   RCD = Read Configuration Register data on A[15:0]. A[MAX:16] can select any partition.
3. The second cycle of the Buffered Program Command is the word count of the data to be loaded into the write buffer. This is followed by up to 32 words of data.Then the confirm command (0xD0) is issued, triggering the array programming operation.
4. The confirm command (0xD0) is followed by the buffer data.

3.3 Command Definitions

Valid device command codes and descriptions are shown in Table 5.

Table 5. Command Codes and Definitions (Sheet 1 of 2)

Mode	Code	Device Mode	Description
Read	0xFF	Read Array	Places the addressed partition in Read Array mode. Array data is output on D[15:0].
	0x70	Read Status Register	Places the addressed partition in Read Status Register mode. The partition enters this mode after a program or erase command is issued. Status Register data is output on D[7:0].
	0x90	Read Device ID or Configuration Register	Places the addressed partition in Read Device Identifier mode. Subsequent reads from addresses within the partition outputs manufacturer/device codes, Configuration Register data, Block Lock status, or Protection Register data on D[15:0].
	0x98	Read Query	Places the addressed partition in Read Query mode. Subsequent reads from the partition addresses output Common Flash Interface information on D[7:0].
	0x50	Clear Status Register	The WSM can only set Status Register error bits. The Clear Status Register command is used to clear the SR error bits.
Write	0x40	Word Program Setup	First cycle of a 2-cycle programming command; prepares the CUI for a write operation. On the next write cycle, the address and data are latched and the WSM executes the programming algorithm at the addressed location. During program operations, the partition responds only to Read Status Register and Program Suspend commands. CE# or OE# must be toggled to update the Status Register. The Read Array command must be issued to read array data after programming has finished.
	0x10	Alternate Word Program Setup	Equivalent to the Word Program Setup command, 0x40.
	0xE8	Buffered Program	This command loads a variable number of bytes up to the buffer size of 32 words onto the program buffer.
	0xD0	Buffered Program Confirm	The confirm command is Issued after the data streaming for writing into the buffer is done. This instructs the WSM to perform the Buffered Program algorithm, writing the data from the buffer to the flash memory array.
	0x80	Buffered Enhanced Factory Programming Setup	First cycle of a 2-cycle command; initiates Buffered Enhanced Factory Program mode (Buffered EFP). The CUI then waits for the Buffered EFP Confirm command, 0xD0, that initiates the Buffered EFP algorithm. All other commands are ignored when Buffered EFP mode begins.
	0xD0	Buffered EFP Confirm	If the previous command was Buffered EFP Setup (0x80), the CUI latches the address and data, and prepares the device for Buffered EFP mode.
Erase	0x20	Block Erase Setup	First cycle of a 2-cycle command; prepares the CUI for a block-erase operation. The WSM performs the erase algorithm on the block addressed by the Erase Confirm command. If the next command *is not* the Erase Confirm (0xD0) command, the CUI sets Status Register bits SR[4] and SR[5], and places the addressed partition in read status register mode.
	0xD0	Block Erase Confirm	If the first command was Block Erase Setup (0x20), the CUI latches the address and data, and the WSM erases the addressed block. During block-erase operations, the partition responds only to Read Status Register and Erase Suspend commands. CE# or OE# must be toggled to update the Status Register.
Suspend	0xB0	Program or Erase Suspend	This command issued to any device address initiates a suspend of the currently-executing program or block erase operation. The Status Register indicates successful suspend operation by setting either SR[2] (program suspended) or SR[6] (erase suspended), along with SR[7] (ready). The Write State Machine remains in the suspend mode regardless of control signal states (except for RST# asserted).
	0xD0	Suspend Resume	This command issued to any device address resumes the suspended program or block-erase operation.

Table 5. Command Codes and Definitions (Sheet 2 of 2)

Mode	Code	Device Mode	Description
Block Locking/ Unlocking	0x60	Lock Block Setup	First cycle of a 2-cycle command; prepares the CUI for block lock configuration changes. If the next command is not Block Lock (0x01), Block Unlock (0xD0), or Block Lock-Down (0x2F), the CUI sets Status Register bits SR[4] and SR[5], indicating a command sequence error.
	0x01	Lock Block	If the previous command was Block Lock Setup (0x60), the addressed block is locked.
	0xD0	Unlock Block	If the previous command was Block Lock Setup (0x60), the addressed block is unlocked. If the addressed block is in a lock-down state, the operation has no effect.
	0x2F	Lock-Down Block	If the previous command was Block Lock Setup (0x60), the addressed block is locked down.
Protection	0xC0	Program Protection Register Setup	First cycle of a 2-cycle command; prepares the device for a Protection Register or Lock Register program operation. The second cycle latches the register address and data, and starts the programming algorithm. CE# or OE# must be toggled to update the Status Register. The Read Array command must be issued to read array data after programming has finished.
Configu-ration	0x60	Read Configuration Register Setup	First cycle of a 2-cycle command; prepares the CUI for device read configuration. If the Set Read Configuration Register command (0x03) is not the next command, the CUI sets Status Register bits SR[4] and SR[5], indicating a command sequence error.
	0x03	Read Configuration Register	If the previous command was Read Configuration Register Setup (0x60), the CUI latches the address and writes A[15:0] to the Read Configuration Register. Following a Configure Read Configuration Register command, subsequent read operations access array data.

4.0 Read Operations

The device supports two read modes: asynchronous page mode and synchronous burst mode. Asynchronous page mode is the default read mode after device power-up or a reset. The Read Configuration Register must be configured to enable synchronous burst reads of the flash memory array (see Section 4.3, “Read Configuration Register (RCR)” on page 21).

Each partition of the device can be in any of four read states: Read Array, Read Identifier, Read Status or Read Query. Upon power-up, or after a reset, all partitions of the device default to Read Array. To change a partition’s read state, the appropriate read command must be written to the device (see Section 3.2, “Device Commands” on page 15). See Section 9.0, “Special Read States” on page 43 for details regarding Read Status, Read ID, and CFI Query modes.

If the Read Array command is written to a partition that is performing a program or erase operation, invalid data is read until the program or erase operation completes. Subsequent reads produce array data. If a Program Suspend or Erase Suspend command is issued during a program or erase operation, a subsequent Read Array command puts the addressed partition into Read Array. The Read Array command functions independent of V_{PP}.

The following sections describe read-mode operations in detail.

4.1 Asynchronous Page-Mode Read

Following a device power-up or reset, asynchronous page mode is the default read mode and all partitions are set to Read Array. However, to perform array reads after any other device operation (e.g. write operation), the Read Array command must be issued in order to read from the flash memory array. Asynchronous page-mode reads are permitted in all blocks.

Note: Asynchronous page-mode reads can only be performed when Read Configuration Register bit RCR[15] is set (see Section 4.3, “Read Configuration Register (RCR)” on page 21).

To perform an asynchronous page-mode read, an address is driven onto A[MAX:0], and CE# and ADV# are asserted. WE# and RST# must already have been deasserted. WAIT is asserted during asynchronous page mode. ADV# can be driven high to latch the address, or it can be held low throughout the read cycle. CLK is not used for asynchronous page-mode reads, and is ignored. If only asynchronous reads are to be performed, CLK should be tied to a valid V_{IH} level. Array data is driven onto D[15:0] after an initial access time t_{AVQV} delay. (see Section 12.0, “AC Characteristics” on page 51).

In asynchronous page mode, four data words are “sensed” simultaneously from the flash memory array and loaded into an internal page buffer. The buffer word corresponding to the initial address on A[MAX:0] is driven onto D[15:0] after the initial access delay. Address bits A[MAX:2] select the 4-word group. Address bits A[1:0] determine which word of the 4-word group is output from the data buffer at any given time.

4.2 Synchronous Burst-Mode Read

Read Configuration register bits CR[15:0] must be set before synchronous burst operation can be performed. (See Section 4.3, “Read Configuration Register (RCR)” on page 21 for details). Synchronous burst mode outputs 4, 8, 16, or a continuous number of contiguous words. To perform a synchronous burst- read, an initial address is driven onto A[MAX:0], and CE# and ADV# are asserted. WE# and RST# must already have been deasserted. ADV# is asserted, and then deasserted to latch the address. Alternately, ADV# can remain asserted throughout the burst access, in which case the address is latched on the next valid CLK edge while ADV# is asserted.

The first word is output from the data buffer on the next valid CLK edge after the initial access latency delay (see Section 4.3.2, “Latency Count” on page 22). Subsequent data is output on valid CLK edges following a minimum delay. Synchronous burst-mode reads can only step through the data buffer once, and can only do so in a sequential manner, starting from the address latched at the beginning of the burst cycle (see Section 12.0, “AC Characteristics” on page 51).

During synchronous read operations, WAIT is driven with respect to CE#. WAIT indicates invalid data when asserted, and valid data when de-asserted with respect to a valid clock edge. See Figure 18, “Burst Suspend Timing” on page 55 for additional details.

The device supports 4-word, 8-word, 16-word, and continuous-word bursts. Synchronous burst-mode reads can only be performed in a partition that is in Read Array. If an attempt is made to perform a synchronous burst-mode read from a partition that is not in Read Array (e.g. Read Status Register, Read Device ID, CFI Query) the flash device still drives data onto the data bus with respect to the clock, but not in accordance with the settings of the RCR. The addressed data is output on the next valid clock edge following the initial asynchronous access delay, and remains on the bus for the duration of the access cycle. WAIT is asserted during synchronous non-array read.

4.2.1 Burst Suspend

The Burst Suspend feature of the device can reduce or eliminate the initial access latency incurred when system software needs to suspend a burst sequence that is in progress in order to retrieve data from another device on the same system bus. The system processor can resume the burst sequence later. Burst suspend provides maximum benefit in non-cache systems.

Burst accesses can be suspended during the initial access latency (before data is received) or after the device has output data. When a burst access is suspended, internal array sensing continues and any previously latched internal data is retained. A burst sequence can be suspended and resumed without limit as long as device operation conditions are met.

Burst Suspend occurs when CE# is asserted, the current address has been latched (either ADV# rising edge or valid CLK edge), CLK is halted, and OE# is deasserted. CLK can be halted when it is at V_{IH} or V_{IL}.

To resume the burst access, OE# is reasserted and CLK is restarted. Subsequent CLK edges resume the burst sequence.

Within the device, CE# gates WAIT. Therefore, during Burst Suspend WAIT remains active and does not revert to a high-impedance state when OE# is deasserted. This can cause contention with another device attempting to control the system’s READY signal during a Burst Suspend. Systems using the Burst Suspend feature should not connect the device’s WAIT signal directly to the system’s READY signal. See Figure 18, “Burst Suspend Timing” on page 55.

4.3 Read Configuration Register (RCR)

The RCR is used to select the read mode (synchronous or asynchronous), and it defines the synchronous burst characteristics of the device. To modify RCR settings, use the Configure Read Configuration Register command (see Section 3.2, "Device Commands" on page 15).

RCR contents can be examined using the Read Device Identifier command, and then reading from <partition base address> + 0x05 (see Section 9.2, "Read Device Identifier" on page 44).

The RCR is shown in Table 6. The following sections describe each RCR bit.

Table 6. Read Configuration Register Description

Read Configuration Register (RCR)															
Read Mode	RES	Latency Count			WAIT Polarity	Data Hold	WAIT Delay	Burst Seq	CLK Edge	RES	RES	Burst Wrap	Burst Length		
RM	R	LC[2:0]			WP	DH	WD	BS	CE	R	R	BW	BL[2:0]		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit	Name	Description
15	Read Mode (RM)	0 = Synchronous burst-mode read 1 = Asynchronous page-mode read (default)
14	Reserved (R)	Reserved bits should be cleared (0)
13:11	Latency Count (LC[2:0])	010 =Code 2 011 =Code 3 100 =Code 4 101 =Code 5 110 =Code 6 111 =Code 7 (Other bit settings are reserved)
10	Wait Polarity (WP)	0 =WAIT signal is active low 1 =WAIT signal is active high (default)
9	Data Hold (DH)	0 =Data held for a 1-clock data cycle 1 =Data held for a 2-clock data cycle (default)
8	Wait Delay (WD)	0 =WAIT de-asserted with valid data 1 =WAIT de-asserted one data cycle before valid data (default)
7	Burst Sequence (BS)	0 =Reserved 1 =Linear (default)
6	Clock Edge (CE)	0 = Falling edge 1 = Rising edge (default)
5:4	Reserved (R)	Reserved bits should be cleared (0)
3	Burst Wrap (BW)	0 =Wrap; Burst accesses wrap within burst length set by BL[2:0] 1 =No Wrap; Burst accesses do not wrap within burst length (default)
2:0	Burst Length (BL[2:0])	001 =4-word burst 010 =8-word burst 011 =16-word burst 111 =Continuous-word burst (default) (Other bit settings are reserved)

NOTE: Latency Code 2, Data Hold for a 2-clock data cycle (DH = 1) Wait must be de-asserted with valid data (WD = 0). WD = 1 is not supported.

4.3.1 Read Mode

The Read Mode (RM) bit selects synchronous burst-mode or asynchronous page-mode operation for the device. When the RM bit is set, asynchronous page mode is selected (default). When RM is cleared, synchronous burst mode is selected.

4.3.2 Latency Count

The Latency Count bits, LC[2:0], tell the device how many clock cycles must elapse from the rising edge of ADV# (or from the first valid clock edge after ADV# is asserted) until the first data word is to be driven onto D[15:0]. The input clock frequency is used to determine this value. Figure 3 shows the data output latency for the different settings of LC[2:0].

During synchronous burst a Latency Count setting of Code 5 will cause 1 WAIT state (Code 6 will cause 2 WAIT states, and Code 7 will cause 3 WAIT states) after every four words, regardless of whether a 16-word boundary is crossed. If CR.[9] (Data Hold) bit is set (data hold of two clocks) this WAIT condition will not occur because enough clocks elapse during each burst cycle to eliminate subsequent WAIT states.

Refer to Table 7, “LC and Frequency Support for Bin 1 tAVQV/tCHQV (90ns / 17ns)” on page 23 and Table 8, “LC and Frequency Support for Bin 2 tAVQV/tCHQV (110ns / 20ns)” on page 23 for Latency Code Settings.

Figure 3. First-Access Latency Count

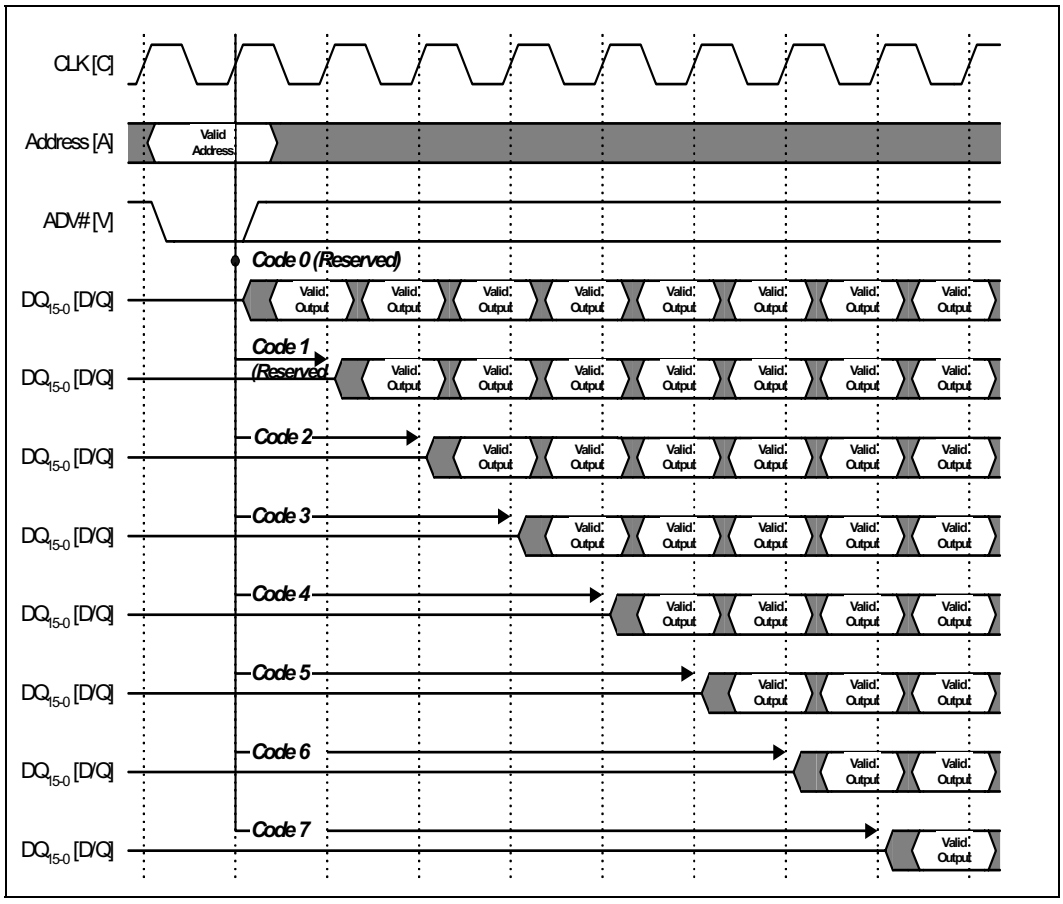


Table 7. LC and Frequency Support for Bin 1 t_{AVQV}/t_{CHQV} (90ns / 17ns)

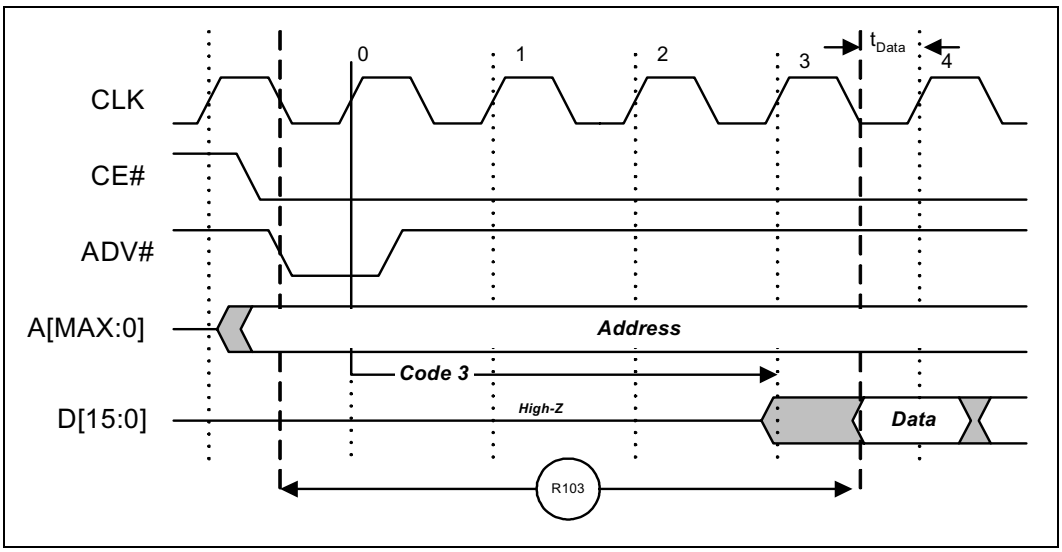
Latency Count Settings	Frequency Support (MHz)
2	≤ 27
3	≤ 40
4, 5, 6, or 7	≤ 50

Table 8. LC and Frequency Support for Bin 2 t_{AVQV}/t_{CHQV} (110ns / 20ns)

Latency Count Settings	Frequency Support (MHz)
2	≤ 22
3	≤ 33
4, 5, 6, or 7	≤40

See Figure 4, “Example Latency Count Setting using Code 3” on page 23.

Figure 4. Example Latency Count Setting using Code 3



4.3.3 WAIT Polarity

The WAIT Polarity bit (WP), RCR[10] determines the asserted level (V_{OH} or V_{OL}) of WAIT. When WP is set, WAIT is asserted-high (default). When WP is cleared, WAIT is asserted-low. WAIT changes state on valid clock edges during active bus cycles (CE# asserted, RST# deasserted).

4.3.3.1 WAIT Signal Function

The WAIT signal indicates data valid when the device is operating in synchronous mode (CR[15]=0), and when addressing a partition that is currently in read-array mode. The WAIT signal is only "deasserted" when data is valid on the bus.

When the device is operating in synchronous non-read-array mode, such as read status, read ID, or read query, WAIT is set to an "asserted" state as determined by CR[10].

When the device is operating in asynchronous page mode, asynchronous single word read mode, and all write operations, WAIT is set to an "asserted" state as determined by CR[10]. See Figure 13, "Asynchronous Single-Word Read (ADV# Latch)" on page 52, and Figure 14, "Asynchronous Page-Mode Read Timing" on page 53.

From a system perspective, the WAIT signal is in the asserted state (based on CR[10]) when the device is operating in synchronous non-read-array mode (such as Read ID, Read Query, or Read Status), or if the device is operating in asynchronous mode (CR[15]=1). In these cases, the system software should ignore (mask) the WAIT signal, because it does not convey any useful information about the validity of what is appearing on the data bus.

Table 9. WAIT Summary Table

CONDITION	WAIT
CE# = V_{IH} CE# = V_{IL}	High-Z Active
OE# = V_{IH} OE# = V_{IL}	High-Z Active
Synchronous Array Reads	Active
Synchronous Non-Array Reads	Asserted
All Asynchronous Reads and all Writes	Asserted

NOTE: Active: WAIT is asserted until data becomes valid, then de-asserts

4.3.4 Data Hold

For burst read operations, the Data Hold (DH) bit determines whether the data output remains valid on D[15:0] for one or two clock cycles. This period of time is called the "**data cycle**". When DH is set, output data is held for two clocks (default). When DH is cleared, output data is held for one clock (see Figure 5). The processor's data setup time and the flash memory's clock-to-data output delay should be considered when determining whether to hold output data for one or two clocks.

A method for determining the Data Hold configuration is shown below:

To set the device at one clock data hold for subsequent reads, the following condition must be satisfied:

$$\mathbf{t_{CHQV}\ (ns) + t_{DATA}\ (ns) \le One\ CLK\ Period\ (ns)}$$

t_{DATA} = Data set up to Clock (defined by CPU)

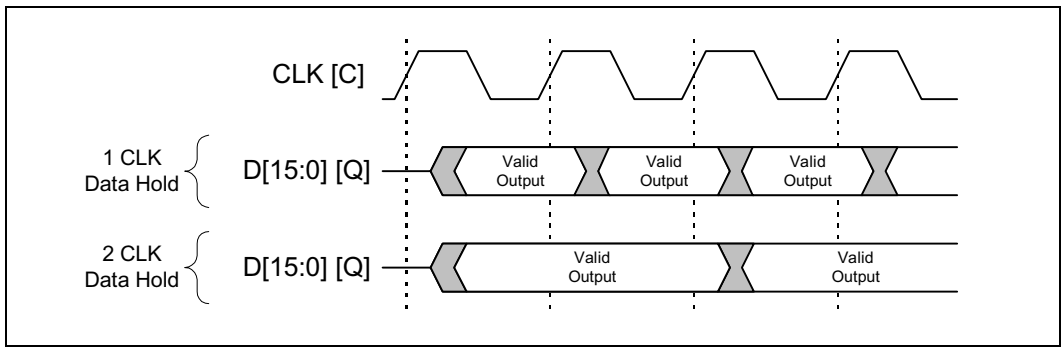
For example, with a clock frequency of 40 MHz, the clock period is 25 ns. Assuming t_{CHQV} = 20 ns and t_{DATA} = 4ns. Applying these values to the formula above:

$$20\ ns + 4\ ns \le 25\ ns$$

The equation is satisfied and data will be available at every clock period with data hold setting at one clock.

If t_{CHQV} (ns) + t_{DATA} (ns) > One CLK Period (ns), data hold setting of 2 clock periods must be used.

Figure 5. Data Hold Timing



4.3.5 WAIT Delay

The WAIT Delay (WD) bit controls the WAIT assertion-delay behavior during synchronous burst reads. WAIT can be asserted either during or one data cycle before valid data is output on DQ[15:0]. When WD is set, WAIT is de-asserted one data cycle *before* valid data (default). When WD is cleared, WAIT is de-asserted *during* valid data.

4.3.6 Burst Sequence

The Burst Sequence (BS) bit selects linear-burst sequence (default). Only linear-burst sequence is supported. Table 10 shows the synchronous burst sequence for all burst lengths, as well as the effect of the Burst Wrap (BW) setting.

Table 10. Burst Sequence Word Ordering (Sheet 1 of 2)

Start Addr. (DEC)	Burst Wrap (RCR[3])	Burst Addressing Sequence (DEC)			
		4-Word Burst (BL[2:0] = 0b001)	8-Word Burst (BL[2:0] = 0b010)	16-Word Burst (BL[2:0] = 0b011)	Continuous Burst (BL[2:0] = 0b111)
0	0	0-1-2-3	0-1-2-3-4-5-6-7	0-1-2-3-4…14-15	0-1-2-3-4-5-6-…
1	0	1-2-3-0	1-2-3-4-5-6-7-0	1-2-3-4-5…15-0	1-2-3-4-5-6-7-…
2	0	2-3-0-1	2-3-4-5-6-7-0-1	2-3-4-5-6…15-0-1	2-3-4-5-6-7-8-…
3	0	3-0-1-2	3-4-5-6-7-0-1-2	3-4-5-6-7…15-0-1-2	3-4-5-6-7-8-9-…
4	0		4-5-6-7-0-1-2-3	4-5-6-7-8…15-0-1-2-3	4-5-6-7-8-9-10…
5	0		5-6-7-0-1-2-3-4	5-6-7-8-9…15-0-1-2-3-4	5-6-7-8-9-10-11…
6	0		6-7-0-1-2-3-4-5	6-7-8-9-10…15-0-1-2-3-4-5	6-7-8-9-10-11-12-…
7	0		7-0-1-2-3-4-5-6	7-8-9-10…15-0-1-2-3-4-5-6	7-8-9-10-11-12-13…
⋮	⋮	⋮	⋮	⋮	⋮
14	0			14-15-0-1-2…12-13	14-15-16-17-18-19-20-…
15	0			15-0-1-2-3…13-14	15-16-17-18-19-20-21-…
⋮	⋮	⋮	⋮	⋮	⋮
0	1	0-1-2-3	0-1-2-3-4-5-6-7	0-1-2-3-4…14-15	0-1-2-3-4-5-6-…
1	1	1-2-3-4	1-2-3-4-5-6-7-8	1-2-3-4-5…15-16	1-2-3-4-5-6-7-…
2	1	2-3-4-5	2-3-4-5-6-7-8-9	2-3-4-5-6…16-17	2-3-4-5-6-7-8-…
3	1	3-4-5-6	3-4-5-6-7-8-9-10	3-4-5-6-7…17-18	3-4-5-6-7-8-9-…
4	1		4-5-6-7-8-9-10-11	4-5-6-7-8…18-19	4-5-6-7-8-9-10…
5	1		5-6-7-8-9-10-11-12	5-6-7-8-9…19-20	5-6-7-8-9-10-11…
6	1		6-7-8-9-10-11-12-13	6-7-8-9-10…20-21	6-7-8-9-10-11-12-…

Table 10. Burst Sequence Word Ordering (Sheet 2 of 2)

7	1		7-8-9-10-11-12-13-14	7-8-9-10-11…21-22	7-8-9-10-11-12-13…
⋮	⋮	⋮	⋮	⋮	⋮
14	1			14-15-16-17-18…28-29	14-15-16-17-18-19-20-…
15	1			15-16-17-18-19…29-30	15-16-17-18-19-20-21-…

4.3.7 Clock Edge

The Clock Edge (CE) bit selects either a rising (default) or falling clock edge for CLK. This clock edge is used at the start of a burst cycle, to output synchronous data, and to assert/deassert WAIT.

4.3.8 Burst Wrap

The Burst Wrap (BW) bit determines whether 4-word, 8-word, or 16-word burst length accesses wrap within the selected word-length boundaries or cross word-length boundaries. When BW is set, burst wrapping does not occur (default). When BW is cleared, burst wrapping occurs.

When performing synchronous burst reads with BW set (no wrap), an output delay may occur when the burst sequence crosses its first device-row (16-word) boundary. If the burst sequence's start address is 4-word aligned, then no delay occurs. If the start address is at the end of a 4-word boundary, the worst case output delay is one clock cycle less than the first access Latency Count. This delay can take place only once, and doesn't occur if the burst sequence does not cross a device-row boundary. WAIT informs the system of this delay when it occurs.

4.3.9 Burst Length

The Burst Length bit (BL[2:0]) selects the linear burst length for all synchronous burst reads of the flash memory array. The burst lengths are 4-word, 8-word, 16-word, and continuous word.

Continuous-burst accesses are linear only, and do not wrap within any word length boundaries (see Table 10, "Burst Sequence Word Ordering" on page 25). When a burst cycle begins, the device outputs synchronous burst data until it reaches the end of the "burstable" address space.

5.0 Programming Operations

The device supports three programming methods: Word Programming (40h/10h), Buffered Programming (E8h, D0h), and Buffered Enhanced Factory Programming (Buffered EFP) (80h, D0h). See Section 3.0, “Device Operations” on page 14 for details on the various programming commands issued to the device.

Successful programming requires the addressed block to be unlocked. If the block is locked down, WP# must be deasserted and the block unlocked before attempting to program the block. Attempting to program a locked block causes a program error (SR[4] and SR[1] set) and termination of the operation. See Section 7.0, “Security Modes” on page 35 for details on locking and unlocking blocks.

The following sections describe device programming in detail.

5.1 Word Programming

Word programming operations are initiated by writing the Word Program Setup command to the device (see Section 3.0, “Device Operations” on page 14). This is followed by a second write to the device with the address and data to be programmed. The partition accessed during both write cycles outputs Status Register data when read. The partition accessed during the second cycle (the data cycle) of the program command sequence is the location where the data is written. See Figure 29, “Word Program Flowchart” on page 70.

Programming can occur in only one partition at a time; all other partitions must be in a read state or in erase suspend. V_{PP} must be above V_{PPLK}, and within the specified V_{PPL} min/max values (nominally 1.8 V).

During programming, the Write State Machine (WSM) executes a sequence of internally-timed events that program the desired data bits at the addressed location, and verifies that the bits are sufficiently programmed. Programming the flash memory array changes “ones” to “zeros.” Memory array bits that are zeros can be changed to ones only by erasing the block (see Section 6.0, “Erase Operations” on page 33).

The Status Register can be examined for programming progress and errors by reading any address within the partition that is being programmed. The partition remains in the Read Status Register state until another command is written to that partition. Issuing the Read Status Register command to another partition address sets that partition to the Read Status Register state, allowing programming progress to be monitored at that partition’s address.

Status Register bit SR[7] indicates the programming status while the sequence executes. Commands that can be issued to the programming partition during programming are Program Suspend, Read Status Register, Read Device Identifier, CFI Query, and Read Array (this returns unknown data). CE# or OE# must be toggled to update Status Register contents.

When programming has finished, Status Register bit SR[4] (when set) indicates a programming failure. If SR[3] is set, the WSM could not perform the word programming operation because V_{PP} was outside of its acceptable limits. If SR[1] is set, the word programming operation attempted to program a locked block, causing the operation to abort.

Before issuing a new command, the Status Register contents should be examined and then cleared using the Clear Status Register command. Any valid command can follow, when word programming has completed.

5.1.1 Factory Word Programming

Factory word programming is similar to word programming in that it uses the same commands and programming algorithms. However, factory word programming enhances the programming performance with $V_{PP} = V_{PPH}$. This can enable faster programming times during OEM manufacturing processes. Factory word programming is not intended for extended use. See Section 11.2, "Operating Conditions" on page 48 for limitations when $V_{PP} = V_{PPH}$.

Note: When $V_{PP} = V_{PPL}$, the device draws programming current from the V_{CC} supply. If V_{PP} is driven by a logic signal, V_{PPL} must remain above V_{PPL} MIN to program the device. When $V_{PP} = V_{PPH}$, the device draws programming current from the V_{PP} supply. Figure 6, "Example VPP Supply Connections" on page 32 shows examples of device power supply configurations.

5.2 Buffered Programming

The device features a 32-word buffer to enable optimum programming performance. For Buffered Programming, data is first written to an on-chip write buffer. Then the buffer data is programmed into the flash memory array in buffer-size increments. This can improve system programming performance significantly over non-buffered programming.

When the Buffered Programming Setup command is issued (see Section 3.2, "Device Commands" on page 15), Status Register information is updated and reflects the availability of the buffer. SR[7] indicates buffer availability: if set, the buffer is available; if cleared, the buffer is not available. To retry, issue the Buffered Programming Setup command again, and re-check SR[7]. When SR[7] is set, the buffer is ready for loading. (see Figure 31, "Buffered Program Flowchart" on page 72).

On the next write, a word count is written to the device at the buffer address. This tells the device how many data words will be written to the buffer, up to the maximum size of the buffer.

On the next write, a device start address is given along with the first data to be written to the flash memory array. Subsequent writes provide additional device addresses and data. All data addresses must lie within the start address plus the word count. Optimum programming performance and lower power usage are obtained by aligning the starting address at the beginning of a 32-word boundary (A[4:0] = 0x00). A misaligned starting address doubles the total program time.

After the last data is written to the buffer, the Buffered Programming Confirm command is issued. The WSM begins to program buffer contents to the flash memory array. If a command other than the Buffered Programming Confirm command is written to the device, a command sequence error occurs and Status Register bits SR[7,5,4] are set. If an error occurs while writing to the array, the device stops programming, and Status Register bits SR[7,4] are set, indicating a programming failure.

Reading from another partition is allowed while data is being programmed into the array from the write buffer (see Section 8.0, "Dual-Operation Considerations" on page 40).

Additional buffer writes can be initiated by issuing another Buffered Programming Setup command and repeating the buffered program sequence. Buffered programming may be performed with $V_{PP} = V_{PPL}$ or V_{PPH} (see Section 11.2, "Operating Conditions" on page 48 for limitations when operating the device with $V_{PP} = V_{PPH}$).

When Status Register bits SR[5,4] are set, the device does not accept Buffered Program commands. If an attempt is made to program past an erase-block boundary using the Buffered Program command, the device aborts the operation. This generates a command sequence error, and Status Register bits SR[5,4] are set.

If Buffered programming is attempted while V_{PP} is below V_{PPLK}, Status Register bits SR[4,3] are set. If any errors are detected that have set Status Register bits, the Status Register should be cleared using the Clear Status Register command.

5.3 Buffered Enhanced Factory Programming

Buffered Enhanced Factory Programing (Buffered EFP) speeds up Multi-Level Cell (MLC) flash programming for today's beat-rate-sensitive manufacturing environments. The enhanced programming algorithm used in Buffered EFP eliminates traditional programming elements that drive up overhead in device programmer systems.

Buffered EFP consists of three phases: Setup, Program/Verify, and Exit (see Figure 32, “Buffered EFP Flowchart” on page 73). It uses a write buffer to spread MLC program performance across 32 data words. Verification occurs in the same phase as programming to accurately program the flash memory cell to the correct bit state.

A single two-cycle command sequence programs the entire block of data. This enhancement eliminates three write cycles per buffer: two commands and the word count for each set of 32 data words. Host programmer bus cycles fill the device’s write buffer followed by a status check. SR[0] indicates when data from the buffer has been programmed into sequential flash memory array locations.

Following the buffer-to-flash array programming sequence, the Write State Machine (WSM) increments internal addressing to automatically select the next 32-word array boundary. This aspect of Buffered EFP saves host programming equipment the address-bus setup overhead.

With adequate continuity testing, programming equipment can rely on the WSM’s internal verification to ensure that the device has programmed properly. This eliminates the external post-program verification and its associated overhead.

5.3.1 Buffered EFP Requirements and Considerations

Buffered EFP requirements:

- Ambient temperature: $T_A = 25°C, \pm 5°C$
- V_{CC} within specified operating range.
- VPP driven to V_{PPH}.
- Target block unlocked before issuing the Buffered EFP Setup and Confirm commands.
- The first-word address (WA0) for the block to be programmed must be held constant from the setup phase through all data streaming into the target block, until transition to the exit phase is desired.
- WA0 must align with the start of an array buffer boundary[1].

Buffered EFP considerations:

- For optimum performance, cycling must be limited below 100 erase cycles per block[2].
- Buffered EFP programs one block at a time; all buffer data must fall within a single block[3].
- Buffered EFP cannot be suspended.
- Programming to the flash memory array can occur only when the buffer is full[4].

- Read operation while performing Buffered EFP is not supported.

NOTES:
1. Word buffer boundaries in the array are determined by A[4:0] (0x00 through 0x1F). The alignment start point is A[4:0] = 0x00.
2. Some degradation in performance may occur if this limit is exceeded, but the internal algorithm continues to work properly.
3. If the internal address counter increments beyond the block's maximum address, addressing wraps around to the beginning of the block.
4. If the number of words is less than 32, remaining locations must be filled with 0xFFFF.

5.3.2 Buffered EFP Setup Phase

After receiving the Buffered EFP Setup and Confirm command sequence, Status Register bit SR[7] (Ready) is cleared, indicating that the WSM is busy with Buffered EFP algorithm startup. A delay before checking SR[7] is required to allow the WSM enough time to perform all of its setups and checks (Block-Lock status, V_{PP} level, etc.). If an error is detected, SR[4] is set and Buffered EFP operation terminates. If the block was found to be locked, SR[1] is also set. SR[3] is set if the error occurred due to an incorrect V_{PP} level.

Note: Reading from the device after the Buffered EFP Setup and Confirm command sequence outputs Status Register data. Do not issue the Read Status Register command; it will be interpreted as data to be loaded into the buffer.

5.3.3 Buffered EFP Program/Verify Phase

After the Buffered EFP Setup Phase has completed, the host programming system must check SR[7,0] to determine the availability of the write buffer for data streaming. SR[7] cleared indicates the device is busy and the Buffered EFP program/verify phase is activated. SR[0] indicates the write buffer is available.

Two basic sequences repeat in this phase: loading of the write buffer, followed by buffer data programming to the array. For Buffered EFP, the count value for buffer loading is always the maximum buffer size of 32 words. During the buffer-loading sequence, data is stored to sequential buffer locations starting at address 0x00. Programming of the buffer contents to the flash memory array starts as soon as the buffer is full. If the number of words is less than 32, the remaining buffer locations must be filled with 0xFFFF.

Caution: *The buffer must be completely filled for programming to occur. Supplying an address outside of the current block's range during a buffer-fill sequence causes the algorithm to exit immediately. Any data previously loaded into the buffer during the fill cycle is not programmed into the array.*

The starting address for data entry must be buffer size aligned, if not the Buffered EFP algorithm will be aborted and the program fail (SR[4]) flag will be set.

Data words from the write buffer are directed to sequential memory locations in the flash memory array; programming continues from where the previous buffer sequence ended. The host programming system must poll SR[0] to determine when the buffer program sequence completes. SR[0] cleared indicates that all buffer data has been transferred to the flash array; SR[0] set indicates that the buffer is not available yet for the next fill cycle. The host system may check full status for errors at any time, but it is only necessary on a block basis after Buffered EFP exit.

The host programming system continues the Buffered EFP algorithm by providing the next group of data words to be written to the buffer. Alternatively, it can terminate this phase by changing the block address to one outside of the current block’s range.

The Program/Verify phase concludes when the programmer writes to a different block address; data supplied must be 0xFFFF. Upon Program/Verify phase completion, the device enters the Buffered EFP Exit phase.

5.3.4 Buffered EFP Exit Phase

When SR[7] is set, the device has returned to normal operating conditions. A full status check should be performed on the partition being programmed at this time to ensure the entire block programmed successfully. When exiting the Buffered EFP algorithm with a block address change, the read mode of both the programmed and the addressed partition will not change. After Buffered EFP exit, any valid command can be issued to the device.

5.4 Program Suspend

Issuing the Program Suspend command while programming suspends the programming operation. This allows data to be accessed from memory locations other than the one being programmed. The Program Suspend command can be issued to any device address; the corresponding partition is not affected. A program operation can be suspended to perform reads only. Additionally, a program operation that is running during an erase suspend can be suspended to perform a read operation (see Figure 30, “Program Suspend/Resume Flowchart” on page 71).

When a programming operation is executing, issuing the Program Suspend command requests the WSM to suspend the programming algorithm at predetermined points. The partition that is suspended continues to output Status Register data after the Program Suspend command is issued. Programming is suspended when Status Register bits SR[7,2] are set. Suspend latency is specified in Section 12.3, “Program and Erase Characteristics” on page 60.

To read data from blocks within the suspended partition, the Read Array command must be issued to that partition. Read Array, Read Status Register, Read Device Identifier, CFI Query, program RCR, and Program Resume are valid commands during a program suspend.

A program operation does not need to be suspended in order to read data from a block in another partition that is not programming. If the other partition is already in a Read Array, Read Device Identifier, or CFI Query state, issuing a valid address returns corresponding read data. If the other partition is not in a read mode, one of the read commands must be issued to the partition before data can be read.

During a program suspend, deasserting CE# places the device in standby, reducing active current. V_{PP} must remain at its programming level, and WP# must remain unchanged while in program suspend. If RST# is asserted, the device is reset.

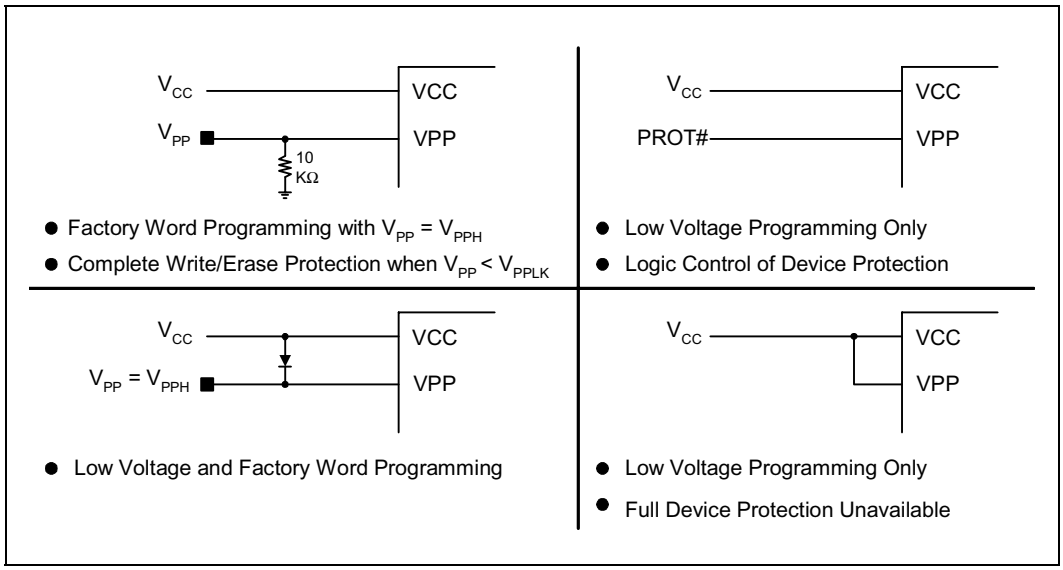
5.5 Program Resume

The Resume command instructs the device to continue programming, and automatically clears Status Register bits SR[7,2]. This command can be written to any partition. When read at the partition that’s programming, the device outputs data corresponding to the partition’s last state. If error bits are set, the Status Register should be cleared before issuing the next instruction. RST# must remain deasserted (see Figure 30, “Program Suspend/Resume Flowchart” on page 71).

5.6 Program Protection

When $V_{PP} = V_{IL}$, absolute hardware write protection is provided for all device blocks. If V_{PP} is below V_{PPLK}, programming operations halt and SR[3] is set indicating a V_{PP}-level error. Block lock registers are not affected by the voltage level on V_{PP}; they may still be programmed and read, even if V_{PP} is less than V_{PPLK}.

Figure 6. Example VPP Supply Connections



6.0 Erase Operations

Flash erasing is performed on a block basis. An entire block is erased each time an erase command sequence is issued, and only one block is erased at a time. When a block is erased, all bits within that block read as logical ones. The following sections describe block erase operations in detail.

6.1 Block Erase

Block erase operations are initiated by writing the Block Erase Setup command to the address of the block to be erased (see Section 3.2, "Device Commands" on page 15). Next, the Block Erase Confirm command is written to the address of the block to be erased. Erasing can occur in only one partition at a time; all other partitions must be in a read state. If the device is placed in standby (CE# deasserted) during an erase operation, the device completes the erase operation before entering standby.V_{PP} must be above V_{PPLK} and the block must be unlocked (see Figure 33, "Block Erase Flowchart" on page 74).

During a block erase, the Write State Machine (WSM) executes a sequence of internally-timed events that conditions, erases, and verifies all bits within the block. Erasing the flash memory array changes "zeros" to "ones." Memory array bits that are ones can be changed to zeros only by programming the block (see Section 5.0, "Programming Operations" on page 27).

The Status Register can be examined for block erase progress and errors by reading any address within the partition that is being erased. The partition remains in the Read Status Register state until another command is written to that partition. Issuing the Read Status Register command to another partition address sets that partition to the Read Status Register state, allowing erase progress to be monitored at that partition's address. SR[0] indicates whether the addressed partition or another partition is erasing. The partition's Status Register bit SR[7] is set upon erase completion.

Status Register bit SR[7] indicates block erase status while the sequence executes. When the erase operation has finished, Status Register bit SR[5] indicates an erase failure if set. SR[3] set would indicate that the WSM could not perform the erase operation because V_{PP} was outside of its acceptable limits. SR[1] set indicates that the erase operation attempted to erase a locked block, causing the operation to abort. CE# or OE# must be toggled to update Status Register contents.

Before issuing a new command, the Status Register contents should be examined and then cleared using the Clear Status Register command. Any valid command can follow once the block erase operation has completed.

6.2 Erase Suspend

Issuing the Erase Suspend command while erasing suspends the block erase operation. This allows data to be accessed from memory locations other than the one being erased. The Erase Suspend command can be issued to any device address; the corresponding partition is not affected. A block erase operation can be suspended to perform a word or write-buffer program operation, or a read operation within any block except the block that is erase suspended (see Figure 30, "Program Suspend/Resume Flowchart" on page 71).

When a block erase operation is executing, issuing the Erase Suspend command requests the WSM to suspend the erase algorithm at predetermined points. The partition that is suspended continues to output Status Register data after the Erase Suspend command is issued. Block erase is suspended when Status Register bits SR[7,6] are set. Suspend latency is specified in Section 12.3, "Program and Erase Characteristics" on page 60.

To read data from blocks within the suspended partition (other than an erase-suspended block), the Read Array command must be issued to that partition first. During Erase Suspend, a Program command can be issued to any block other than the erase-suspended block. Block erase cannot resume until program operations initiated during erase suspend complete. Read Array, Read Status Register, Read Device Identifier, CFI Query, and Erase Resume are valid commands during Erase Suspend. Additionally, Clear Status Register, Program, Program Suspend, Block Lock, Block Unlock, and Block Lock-Down are valid commands during Erase Suspend.

To read data from a block in a partition that is not erasing, the erase operation does not need to be suspended. If the other partition is already in Read Array, Read Device Identifier, or CFI Query, issuing a valid address returns corresponding data. If the other partition is not in a read state, one of the read commands must be issued to the partition before data can be read.

During an erase suspend, deasserting CE# places the device in standby, reducing active current. V_{PP} must remain at a valid level, and WP# must remain unchanged while in erase suspend. If RST# is asserted, the device is reset.

6.3 Erase Resume

The Erase Resume command instructs the device to continue erasing, and automatically clears status register bits SR[7,6]. This command can be written to any partition. When read at the partition that's erasing, the device outputs data corresponding to the partition's last state. If status register error bits are set, the Status Register should be cleared before issuing the next instruction. RST# must remain deasserted (see Figure 30, "Program Suspend/Resume Flowchart" on page 71).

6.4 Erase Protection

When $V_{PP} = V_{IL}$, absolute hardware erase protection is provided for all device blocks. If V_{PP} is below V_{PPLK}, erase operations halt and SR[3] is set indicating a V_{PP}-level error.

7.0 Security Modes

The device features security modes used to protect the information stored in the flash memory array. The following sections describe each security mode in detail.

7.1 Block Locking

Individual instant block locking is used to protect user code and/or data within the flash memory array. All blocks power up in a locked state to protect array data from being altered during power transitions. Any block can be locked or unlocked with no latency. Locked blocks cannot be programmed or erased; they can only be read.

Software-controlled security is implemented using the Block Lock and Block Unlock commands. Hardware-controlled security can be implemented using the Block Lock-Down command along with asserting WP#. Also, V_{PP} data security can be used to inhibit program and erase operations (see Section 5.6, “Program Protection” on page 32 and Section 6.4, “Erase Protection” on page 34).

7.1.1 Lock Block

To lock a block, issue the Lock Block Setup command. The next command must be the Lock Block command issued to the desired block’s address (see Section 3.2, “Device Commands” on page 15 and Figure 35, “Block Lock Operations Flowchart” on page 76). If the Set Read Configuration Register command is issued after the Block Lock Setup command, the device configures the RCR instead.

Block lock and unlock operations are not affected by the voltage level on V_{PP}. The block lock bits may be modified and/or read even if V_{PP} is below V_{PPLK}.

7.1.2 Unlock Block

The Unlock Block command is used to unlock blocks (see Section 3.2, “Device Commands” on page 15). Unlocked blocks can be read, programmed, and erased. Unlocked blocks return to a locked state when the device is reset or powered down. If a block is in a lock-down state, WP# must be deasserted before it can be unlocked (see Figure 7, “Block Locking State Diagram” on page 36).

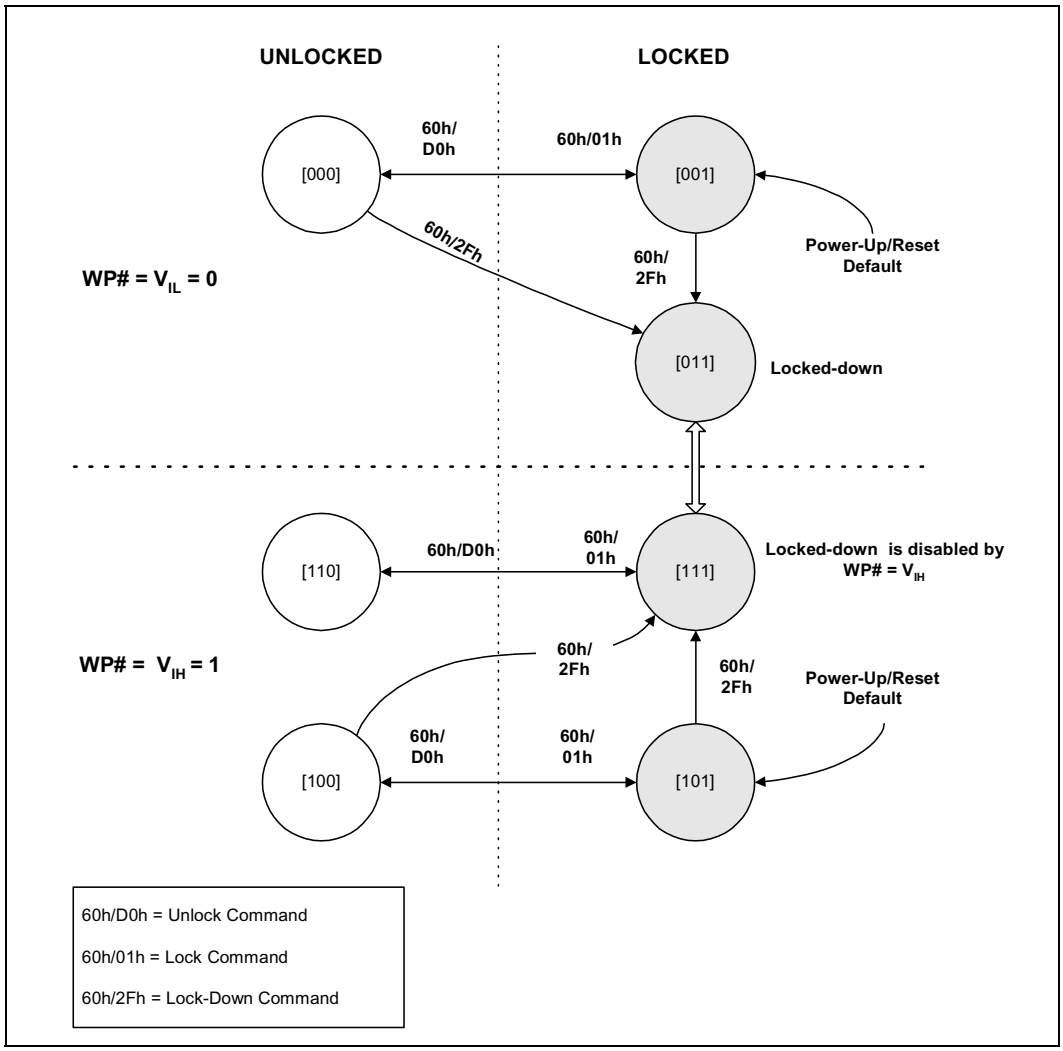
7.1.3 Lock-Down Block

A locked or unlocked block can be locked-down by writing the Lock-Down Block command sequence (see Section 3.2, “Device Commands” on page 15). Blocks in a lock-down state cannot be programmed or erased; they can only be read. However, unlike locked blocks, their locked state cannot be changed by software commands alone. A locked-down block can only be unlocked by issuing the Unlock Block command with WP# deasserted. To return an unlocked block to locked-down state, a Lock command must be issued prior to changing WP# to V_{IL}. Locked-down blocks revert to the locked state upon reset or power up the device (see Figure 7, “Block Locking State Diagram” on page 36).

7.1.4 Block Lock Status

The Read Device Identifier command is used to determine a block's lock status (see Section 9.2, "Read Device Identifier" on page 44). Data bits D[1:0] display the addressed block's lock status; D0 is the addressed block's lock bit, while D1 is the addressed block's lock-down bit.

Figure 7. Block Locking State Diagram



7.1.5 Block Locking During Suspend

Block lock and unlock changes can be performed during an erase suspend. To change block locking during an erase operation, first issue the Erase Suspend command. Monitor the Status Register until SR[7] and SR[6] are set, indicating the device is suspended and ready to accept another command.

Next, write the desired lock command sequence to a block, which changes the lock state of that block. After completing block lock or unlock operations, resume the erase operation using the Erase Resume command.

Note: A Lock Block Setup command followed by any command other than Lock Block, Unlock Block, or Lock-Down Block produces a command sequence error and set Status Register bits SR[4] and SR[5]. If a command sequence error occurs during an erase suspend, SR[4] and SR[5] remains set, even after the erase operation is resumed. Unless the Status Register is cleared using the Clear Status Register command before resuming the erase operation, possible erase errors may be masked by the command sequence error.

If a block is locked or locked-down during an erase suspend of the *same* block, the lock status bits change immediately. However, the erase operation completes when it is resumed. Block lock operations cannot occur during a program suspend. See Appendix A, "Write State Machine (WSM)" on page 63, which shows valid commands during an erase suspend.

7.2 Protection Registers

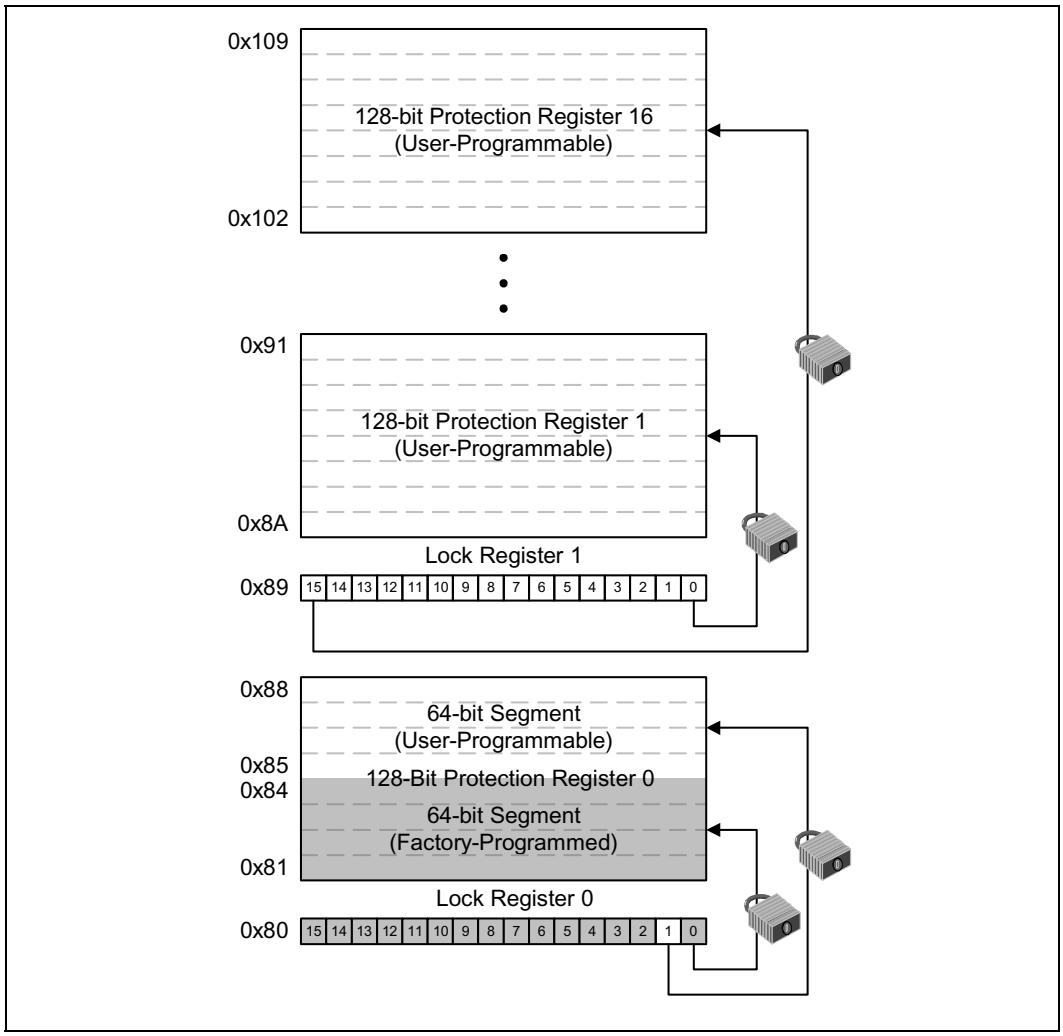
The device contains 17 Protection Registers (PRs) that can be used to implement system security measures and/or device identification. Each Protection Register can be individually locked.

The first 128-bit Protection Register is comprised of two 64-bit (8-word) segments. The lower 64-bit segment is pre-programmed at the factory with a unique 64-bit number. The other 64-bit segment, as well as the other sixteen 128-bit Protection Registers, are blank. Users can program these registers as needed. When programmed, users can then lock the Protection Register(s) to prevent additional bit programming (see Figure 8, "Protection Register Map" on page 38).

The user-programmable Protection Registers contain one-time programmable (OTP) bits; when programmed, register bits cannot be erased. Each Protection Register can be accessed multiple times to program individual bits, as long as the register remains unlocked.

Each Protection Register has an associated Lock Register bit. When a Lock Register bit is programmed, the associated Protection Register can only be read; it can no longer be programmed. Additionally, because the Lock Register bits themselves are OTP, when programmed, Lock Register bits cannot be erased. Therefore, when a Protection Register is locked, it cannot be unlocked

Figure 8. Protection Register Map



7.2.1 Reading the Protection Registers

The Protection Registers can be read from within any partition's address space. To read the Protection Register, first issue the Read Device Identifier command at any partitions' address to place that partition in the Read Device Identifier state (see Section 3.2, "Device Commands" on page 15). Next, perform a read operation at that partition's base address plus the address offset corresponding to the register to be read. Table 13, "Device Identifier Information" on page 45 shows the address offsets of the Protection Registers and Lock Registers. Register data is read 16 bits at a time.

Note: If a program or erase operation occurs within the device while it is reading a Protection Register, certain restrictions may apply. See Table 11, "Simultaneous Operation Restrictions" on page 42 for details.

7.2.2 Programming the Protection Registers

To program any of the Protection Registers, first issue the Program Protection Register command at the parameter partition's base address plus the offset to the desired Protection Register (see Section 3.2, "Device Commands" on page 15). Next, write the desired Protection Register data to the same Protection Register address (see Figure 8, "Protection Register Map" on page 38).

The device programs the 64-bit and 128-bit user-programmable Protection Register data 16 bits at a time (see Figure 36, "Protection Register Programming Flowchart" on page 77). Issuing the Program Protection Register command outside of the Protection Register's address space causes a program error (SR[4] set). Attempting to program a locked Protection Register causes a program error (SR[4] set) and a lock error (SR[1] set).

Note: If a program or erase operation occurs when programming a Protection Register, certain restrictions may apply. See Table 11, "Simultaneous Operation Restrictions" on page 42 for details.

7.2.3 Locking the Protection Registers

Each Protection Register can be locked by programming its respective lock bit in the Lock Register. To lock a Protection Register, program the corresponding bit in the Lock Register by issuing the Program Lock Register command, followed by the desired Lock Register data (see Section 3.2, "Device Commands" on page 15). The physical addresses of the Lock Registers are 0x80 for register 0 and 0x89 for register 1. These addresses are used when programming the lock registers (see Table 13, "Device Identifier Information" on page 45).

Bit 0 of Lock Register 0 is already programmed at the factory, locking the lower, pre-programmed 64-bit region of the first 128-bit Protection Register containing the unique identification number of the device. Bit 1 of Lock Register 0 can be programmed by the user to lock the user-programmable, 64-bit region of the first 128-bit Protection Register. The other bits in Lock Register 0 are not used.

Lock Register 1 controls the locking of the upper sixteen 128-bit Protection Registers. Each of the 16 bits of Lock Register 1 correspond to each of the upper sixteen 128-bit Protection Registers. Programming a bit in Lock Register 1 locks the corresponding 128-bit Protection Register.

Caution: After being locked, the Protection Registers cannot be unlocked.

8.0 Dual-Operation Considerations

The multi-partition architecture of the device allows background programming (or erasing) to occur in one partition while data reads (or code execution) take place in another partition.

8.1 Memory Partitioning

The L30 flash memory array is divided into multiple 8-Mbit partitions, which allows simultaneous read-while-write operations. Simultaneous program and erase is not allowed. Only one partition at a time can be in program or erase mode.

The flash device supports read-while-write operations with *bus cycle granularity* and not command granularity. In other words, it is *not* assumed that both bus cycles of a two cycle command (an erase command for example) will always occur as back to back bus cycles to the flash device. In practice, code fetches (reads) may be interspersed between write cycles to the flash device, and they will likely be directed to a different partition than the one being written. This is especially true when a processor is executing code from one partition that instructs the processor to program or erase in another partition.

8.2 Read-While-Write Command Sequences

When issuing commands to the device, a read operation can occur between 2-cycle Write command's (Figure 9, and Figure 10). However, a write operation issued between a 2-cycle commands write sequence causes a command sequence error. (See Figure 11)

When reading from the same partition after issuing a Setup command, Status Register data is returned, regardless of the read mode of the partition prior to issuing the Setup command.

.

Figure 9. Operating Mode with Correct Command Sequence Example

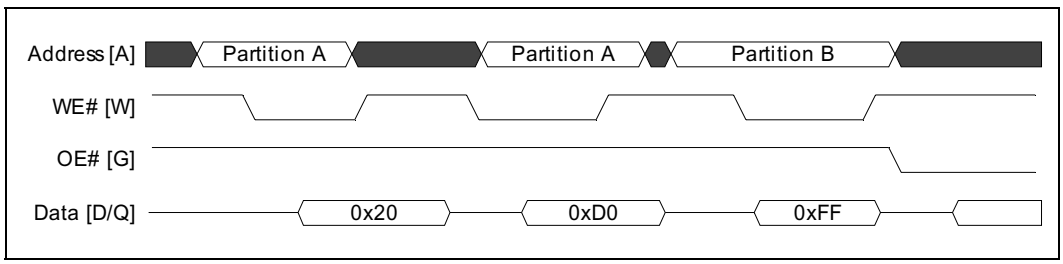


Figure 10. Operating Mode with Correct Command Sequence Example

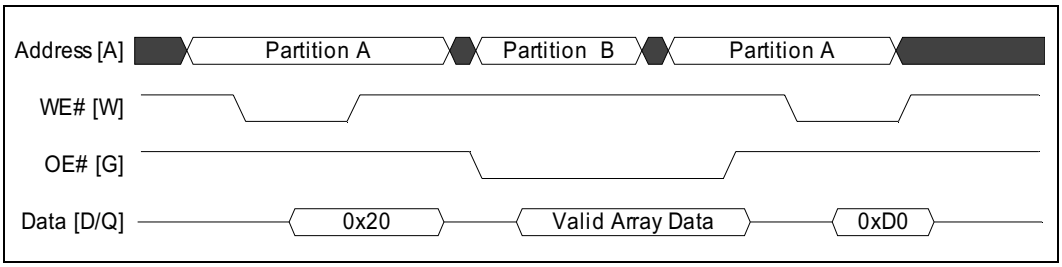
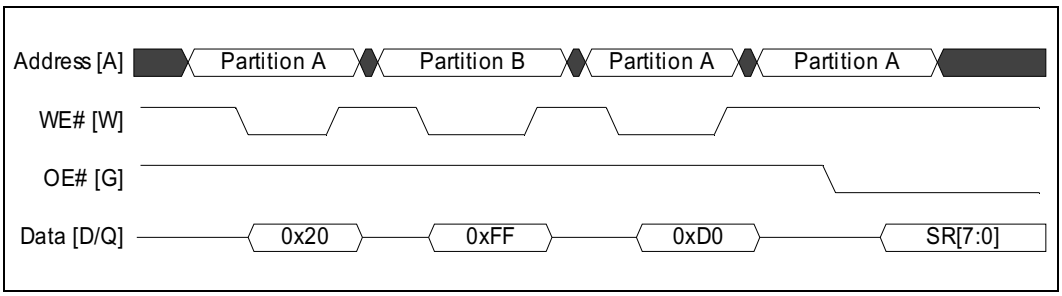


Figure 11. Operating Mode with Illegal Command Sequence Example



8.2.1 Simultaneous Operation Details

The L30 flash memory device supports simultaneous read from one partition while programming or erasing in any other partition. Certain features like the Protection Registers and Query data have special requirements with respect to simultaneous operation capability. These will be detailed in the following sections.

8.2.2 Synchronous and Asynchronous Read-While-Write Characteristics and Waveforms

W18 - t_{WHAV}

The AC parameter W18 (t_{WHAV}-WE# High to Address Valid) is required when transitioning from a write cycle (WE# going high) to perform an asynchronous read (only address valid is required).

W19 and W20 - t_{WHCV} and t_{WHVH}

The AC parameters W19 or W20 (t_{WHCV}-WE# High to Clock Valid, and t_{WHVH} - WE# High to ADV# High) is required when transitioning from a write cycle (WE# going high) to perform a synchronous burst read. A delay from WE# going high to a valid clock edge or ADV# going high to latch a new address must be met.

See Figure 21, “Write to Asynchronous Read Timing” on page 58, Figure 22, “Synchronous Read to Write Timing” on page 58 and Figure 23, “Write to Synchronous Read Timing” on page 59 for representation of these timings.

8.2.3 Read Operation During Buffered Programming Flowchart

The multi-partition architecture of the device allows background programming (or erasing) to occur in one partition while data reads (or code execution) take place in another partition.

To perform a read while buffered programming operation, first issue a Buffered Program set up command in a partition. When a read operation occurs in the same partition after issuing a setup command, Status Register data will be returned, regardless of the read mode of the partition prior to issuing the setup command. Toggle CE# or OE# to update Status Register data.

To read data from a block in other partition and the other partition already in read array mode, a new block address must be issued. However, if the other partition is ***not*** already in read array mode, issuing a read array command will cause the buffered program operation to abort and a command sequence error would be posted in the Status Register. See Figure 37, "Read While Buffered Programming Flowchart" on page 78 for more details.

Note: Simultaneous read-while-Buffered EFP is not supported.

8.3 Simultaneous Operation Restrictions

Since the L30 flash memory device supports simultaneous read from one partition while programming or erasing in another partition, certain features like the Protection Registers and CFI Query data have special requirements with respect to simultaneous operation capability. (Table 11 provides details on restrictions during simultaneous operations.)

Table 11. Simultaneous Operation Restrictions

Protection Register or CFI data	Parameter Partition Array Data	Other Partitions	Notes
Read	(See Notes)	Write/Erase	While programming or erasing in a main partition, the Protection Register or CFI data may be read from any other partition. Reading the parameter partition array data is not allowed if the Protection Register or Query data is being read from addresses within the parameter partition.
(See Notes)	Read	Write/Erase	While programming or erasing in a main partition, read operations are allowed in the parameter partition. Accessing the Protection Registers or CFI data from parameter partition addresses is not allowed when reading array data from the parameter partition.
Read	Read	Write/Erase	While programming or erasing in a main partition, read operations are allowed in the parameter partition. Accessing the Protection Registers or CFI data in a partition that is *different* from the one being programed/erased, and also *different* from the parameter partition is allowed.
Write	No Access Allowed	Read	While programming the Protection Register, reads are only allowed in the other main partitions. Access to array data in the parameter partition is not allowed. Programming of the Protection Register can only occur in the parameter partition, which means this partition is in Read Status.
No Access Allowed	Write/Erase	Read	While programming or erasing the parameter partition, reads of the Protection Registers or CFI data are not allowed in *any* partition. Reads in partitions other than the main partitions are supported.

9.0 Special Read States

The following sections describe non-array read states. Non-array reads can be performed in asynchronous read or synchronous burst mode. However, a non-array read operation occurs as asynchronous single-word mode or single synchronous mode. When non-array reads are performed in page mode or burst mode, only the first data is valid. All subsequent data are undefined.

Each partition can be in one of its read states independent of other partitions' modes. See Figure 12, "Asynchronous Single-Word Read (ADV# Low)" on page 52 and Figure 15, "Synchronous Single-Word Array-Read Timing" on page 53 for details.

9.1 Read Status Register

The status of any partition is determined by reading the Status Register from the address of that particular partition. To read the Status Register, issue the Read Status Register command within the desired partition's address range. Status Register information is available at the partition address to which the Read Status Register, Word Program, or Block Erase command was issued. Status Register data is automatically made available following a Word Program, Block Erase, or Block Lock command sequence. Reads from a partition after any of these command sequences outputs that partition's status until another valid command is written to that partition (e.g. Read Array command).

Status Register data is output on D[7:0], while 0x00 is output on D[15:8]. The falling edge of OE#, ADV#, or CE# (whichever occurs first) updates and latches the Status Register contents. Status Register read operations do not affect the read state of the other partitions.

The Device Write Status bit (SR[7]) provides overall status of the device. The Partition Status bit (SR[0]) indicates whether the addressed partition or some other partition is actively programming or erasing. Status register bits SR[6:1] present status and error information about the program, erase, suspend, V_{PP}, and block-locked operations.

Table 12. Status Register Description (Sheet 1 of 2)

Status Register (SR)							Default Value = 0x80
Device Write Status	Erase Suspend Status	Erase Status	Program Status	V_{PP} Status	Program Suspend Status	Block-Locked Status	Partition Status
DWS	ESS	ES	PS	VPPS	PSS	BLS	PWS
7	6	5	4	3	2	1	0

Bit	Name	Description
7	Device Write Status (DWS)	0 = Device is busy; program or erase cycle in progress; SR[0] valid. 1 = Device is ready; SR[6:1] are valid.
6	Erase Suspend Status (ESS)	0 = Erase suspend not in effect. 1 = Erase suspend in effect.
5	Erase Status (ES)	0 = Erase successful. 1 = Erase fail or program sequence error when set with SR[4,7].
4	Program Status (PS)	0 = Program successful. 1 = Program fail or program sequence error when set with SR[5,7]
3	V_{PP} Status (VPPS)	0 = VPP within acceptable limits during program or erase operation. 1 = VPP < VPPLK during program or erase operation.

Table 12. Status Register Description (Sheet 2 of 2)

Status Register (SR)		Default Value = 0x80
2	Program Suspend Status (PSS)	0 = Program suspend not in effect. 1 = Program suspend in effect.
1	Block-Locked Status (BLS)	0 = Block not locked during program or erase. 1 = Block locked during program or erase; operation aborted.
0	Partition Write Status (PWS)	DWS PWS 0 0 = Program or erase operation in addressed partition. 0 1 = Program or erase operation in other partition. 1 0 = No active program or erase operations. 1 1 = Reserved. (Non-buffered EFP operation. For Buffered EFP operation, see Section 5.3, "Buffered Enhanced Factory Programming" on page 29).

Always clear the Status Register prior to resuming erase operations. Avoids Status Register ambiguity when issuing commands during Erase Suspend. If a command sequence error occurs during an erase-suspend state, the Status Register contains the command sequence error status (SR[7,5,4] set). When the erase operation resumes and finishes, possible errors during the erase operation cannot be detected via the Status Register because it contains the previous error status.

9.1.1 Clear Status Register

The Clear Status Register command clears the status register, leaving all partition read states unchanged. It functions independent of V_{PP}. The Write State Machine (WSM) sets and clears SR[7,6,2,0], but it sets bits SR[5:3,1] without clearing them. The Status Register should be cleared before starting a command sequence to avoid any ambiguity. A device reset also clears the Status Register.

9.2 Read Device Identifier

The Read Device Identifier command instructs the addressed partition to output manufacturer code, device identifier code, block-lock status, protection register data, or configuration register data when that partition's addresses are read (see Section 3.2, "Device Commands" on page 15 for details on issuing the Read Device Identifier command). Table 13, "Device Identifier Information" on page 45 and Table 14, "Device ID codes" on page 45 show the address offsets and data values for this device.

Issuing a Read Device Identifier command to a partition that is programming or erasing places that partition in the Read Identifier state while the partition continues to program or erase in the background.

Table 13. Device Identifier Information

Item	Address[1,2]	Data
Manufacturer Code	PBA + 0x00	0089h
Device ID Code	PBA + 0x01	ID (see Table 14)
Block Lock Configuration: • Block Is Unlocked • Block Is Locked • Block Is not Locked-Down • Block Is Locked-Down	BBA + 0x02	Lock Bit: DQ_0 = 0b0 DQ_0 = 0b1 DQ_1 = 0b0 DQ_1 = 0b1
Configuration Register	PBA + 0x05	Configuration Register Data
Lock Register 0	PBA + 0x80	PR-LK0
64-bit Factory-Programmed Protection Register	PBA + 0x81–0x84	Factory Protection Register Data
64-bit User-Programmable Protection Register	PBA + 0x85–0x88	User Protection Register Data
Lock Register 1	PBA + 0x89	Protection Register Data
128-bit User-Programmable Protection Registers	PBA + 0x8A–0x109	PR-LK1

NOTES:
1. PBA = Partition Base Address.
2. BBA = Block Base Address.

Table 14. Device ID codes

ID Code Type	Device Density	Device Identifier Codes	
		–T (Top Parameter)	–B (Bottom Parameter)
Device Code	64 Mbit	8811	8814
	128 Mbit	8812	8815
	256 Mbit	8813	8816

9.3 CFI Query

The CFI Query command instructs the device to output Common Flash Interface (CFI) data when partition addresses are read. See Section 3.2, “Device Commands” on page 15 for details on issuing the CFI Query command. Appendix C, “Common Flash Interface” on page 79 shows CFI information and address offsets within the CFI database.

Issuing the CFI Query command to a partition that is programming or erasing places that partition’s outputs in the CFI Query state, while the partition continues to program or erase in the background.

The CFI Query command is subject to read restrictions dependent on parameter partition availability, as described in Table 11.

10.0 Power and Reset

10.1 Power-Up/Down Characteristics

Power supply sequencing is not required if VCC, VCCQ, and VPP are connected together; If VCCQ and/or VPP are not connected to the VCC supply, then V_{CC} should attain V_{CCMIN} before applying V_{CCQ} and V_{PP}. Device inputs should not be driven before supply voltage equals V_{CCMIN}.

Power supply transitions should only occur when RST# is low. This protects the device from accidental programming or erasure during power transitions.

10.2 Power Supply Decoupling

Flash memory devices require careful power supply de-coupling. Three basic power supply current considerations are: 1) standby current levels; 2) active current levels; and 3) transient peaks produced when CE# and OE# are asserted and deasserted.

When the device is accessed, many internal conditions change. Circuits within the device enable charge-pumps, and internal logic states change at high speed. All of these internal activities produce transient signals. Transient current magnitudes depend on the device outputs' capacitive and inductive loading. Two-line control and correct de-coupling capacitor selection suppress transient voltage peaks.

Because Intel® Multi-Level Cell (MLC) flash memory devices draw their power from VCC, VPP, and VCCQ, each power connection should have a 0.1 µF ceramic capacitor connected to a corresponding ground connection (e.g.VCCQ to VSSQ). High-frequency, inherently low-inductance capacitors should be placed as close as possible to package leads.

Additionally, for every eight devices used in the system, a 4.7 µF electrolytic capacitor should be placed between power and ground close to the devices. The bulk capacitor is meant to overcome voltage droop caused by PCB trace inductance.

10.3 Automatic Power Saving (APS)

Automatic Power Saving (APS) provides low power operation during a read's active state. I_{CCAPS} is the average current measured over any 5 ms time interval, 5 µs after CE# is deasserted. During APS, average current is measured over the same time interval 5 µs after the following events happen: (1) there is no internal read, program or erase operations cease; (2) CE# is asserted; (3) the address lines are quiescent and at V_{SSQ} or V_{CCQ}. OE# may also be driven during APS.

10.4 Reset Characteristics

Asserting RST# during a system reset is important with automated program/erase devices because systems typically expect to read from flash memory when coming out of reset. If a CPU reset occurs without a flash memory reset, proper CPU initialization may not occur. This is because the flash memory may be providing status information, instead of array data as expected. Connect RST# to the same active-low reset signal used for CPU initialization.

Also, because the device is disabled when RST# is asserted, it ignores its control inputs during power-up/down. Invalid bus conditions are masked, providing a level of memory protection.

System designers should guard against spurious writes when V_{CC} voltages are above V_{LKO}. Because both WE# and CE# must be asserted for a write operation, deasserting either signal inhibits writes to the device.

The Command User Interface (CUI) architecture provides additional protection because alteration of memory contents can only occur after successful completion of a two-step command sequence (see Section 3.2, “Device Commands” on page 15).

11.0 Thermal and DC Characteristics

11.1 Absolute Maximum Ratings

Warning: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only.

Parameter	Maximum Rating	Notes
Temperature under bias	–25 °C to +85 °C	
Storage temperature	–65 °C to +125 °C	
Voltage on any signal (except VCC, VPP)	–0.5 V to +3.8 V	1
VPP voltage	–0.2 V to +14 V	1,2,3
VCC voltage	–0.2 V to +2.45 V	1
VCCQ voltage	–0.2 V to +3.8 V	1
Output short circuit current	100 mA	4

NOTES:
1. Voltages shown are specified with respect to V_{SS}. Minimum DC voltage is –0.5 V on input/output signals and –0.2 V on V_{CC}, V_{CCQ}, and V_{PP}. During transitions, this level may undershoot to –2.0 V for periods <20 ns. Maximum DC voltage on V_{CC} is V_{CC} +0.5 V, which, during transitions, may overshoot to V_{CC} +2.0 V for periods <20 ns. Maximum DC voltage on input/output signals and V_{CCQ} is V_{CCQ} +0.5 V, which, during transitions, may overshoot to V_{CCQ} +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. Program/erase voltage is typically 1.7 V–2.0 V. 9.0 V can be applied for 80 hours maximum total, to any blocks for 1000 cycles maximum. 9.0 V program/erase voltage may reduce block cycling capability.
4. Output shorted for no more than one second. No more than one output shorted at a time.

11.2 Operating Conditions

Warning: Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.

Symbol	Parameter		Min	Max	Units	Notes
T_A	Operating Temperature		–25	+85	°C	
V_{CC}	VCC Supply Voltage		1.7	2.0	V	
V_{CCQ}	I/O Supply Voltage		2.2	3.3		
V_{PPL}	V_{PP} Voltage Supply (Logic Level)		0.9	2.0		1
V_{PPH}	Factory word programming V_{PP}		8.5	9.5		
t_{PPH}	Maximum VPP Hours	$V_{PP} = V_{PPH}$		80	Hours	
Block Erase Cycles	Main and Parameter Blocks	$V_{PP} = V_{CC}$	100,000		Cycles	
	Main Blocks	$V_{PP} = V_{PPH}$		1000		
	Parameter Blocks	$V_{PP} = V_{PPH}$		2500		

NOTES:
1. In typical operation, the VPP program voltage is VPPL. VPP can be connected to 8.50 V – 9.5 V for 1000 cycles on main blocks, and 2500 cycles on parameter blocks.

11.3 DC Current Characteristics

Sym	Parameter		V_{CCQ}	2.2 V – 3.3 V		Unit	Test Conditions	Notes
				Typ	Max			
I_{LI}	Input Load Current				±2	µA	$V_{CC} = V_{CC}$ Max $V_{CCQ} = V_{CCQ}$ Max $V_{IN} = V_{CCQ}$ or GND	1
I_{LO}	Output Leakage Current	D[15:0], WAIT			±10	µA	$V_{CC} = V_{CC}$ Max $V_{CCQ} = V_{CCQ}$ Max $V_{IN} = V_{CCQ}$ or GND	
I_{CCS} I_{CCD}	V_{CC} Standby, Power Down		64 Mbit	20	35	µA	$V_{CC} = V_{CC}$Max $V_{CCQ} = V_{CCQ}$Max CE# = V_{CCQ} RST# = V_{CCQ} (for I_{CCS}) RST# = GND (for I_{CCD}) WP# = V_{IH}	1,2
			128 Mbit	30	55			
			256 Mbit	55	95			
I_{CCAPS}	APS		64 Mbit	20	35	µA	$V_{CC} = V_{CC}$ Max $V_{CCQ} = V_{CCQ}$ Max CE# = V_{SSQ} RST# = V_{CCQ} All inputs are at rail to rail (V_{CCQ} or V_{SSQ}).	
			128 Mbit	30	55			
			256 Mbit	55	95			
I_{CCR}	Average V_{CC} Read Current	Asynchronous Single-Word f = 5MHz (1 CLK)		14	16	mA		1
		Page-Mode Read f = 13 MHz (5 CLK)		9	10	mA	4-Word Read	
		Synchronous Burst Read f = 40MHz		16	19	mA	Burst length=4	
				20	24	mA	Burst length=8	
				23	27	mA	Burst length=16	
				30	35	mA	Burst length = Continuous	
		Synchronous Burst Read f = 54MHz		18	21	mA	Burst length=4	
				24	28	mA	Burst length=8	
				28	33	mA	Burst length=16	
				30	35	mA	Burst Length = Continuous	
							$V_{CC} = V_{CC}$MAX CE# = V_{IL} OE# = V_{IH} Inputs: V_{IL} or V_{IH}	
I_{CCW}, I_{CCE}	V_{CC} Program Current, V_{CC} Erase Current			36	51	mA	$V_{PP} = V_{PPL}$, program/erase in progress	1,3,4,7
				26	33	mA	$V_{PP} = V_{PPH}$, program/erase in progress	1,3,5,7
I_{CCWS}, I_{CCES}	V_{CC} Program Suspend Current, V_{CC} Erase Suspend Current		64 Mbit	20	35	µA	CE# = V_{CCQ}; suspend in progress	1,6,3
			128 Mbit	30	55			
			256 Mbit	55	95			
I_{PPS}, I_{PPWS}, I_{PPES}	V_{PP} Standby Current, V_{PP} Program Suspend Current, V_{PP} Erase Suspend Current			0.2	5	µA	$V_{PP} = V_{PPL}$, suspend in progress	1,3

Sym	Parameter	V_{CCQ}	2.2 V – 3.3 V		Unit	Test Conditions	Notes
			Typ	Max			
I_{PPR}	V_{PP} Read		2	15	µA	$V_{PP} \le V_{CC}$	1,3
I_{PPW}	V_{PP} Program Current		0.05	0.10	mA	$V_{PP} = V_{PPL}$, program in progress	
			8	22		$V_{PP} = V_{PPH}$, program in progress	
I_{PPE}	V_{PP} Erase Current		0.05	0.10	mA	$V_{PP} = V_{PPL}$, erase in progress	
			8	22		$V_{PP} = V_{PPH}$, erase in progress	

NOTES:
1. All currents are RMS unless noted. Typical values at typical V_{CC}, T_A = +25°C.
2. I_{CCS} is the average current measured over any 5 ms time interval 5 µs after CE# is deasserted.
3. Sampled, not 100% tested.
4. V_{CC} read + program current is the sum of V_{CC} read and V_{CC} program currents.
5. V_{CC} read + erase current is the sum of V_{CC} read and V_{CC} erase currents.
6. I_{CCES} is specified with the device deselected. If device is read while in erase suspend, current is I_{CCES} plus I_{CCR}.
7. I_{CCW}, I_{CCE} measured over typical or max times specified in Section 12.3, “Program and Erase Characteristics” on page 60

11.4 DC Voltage Characteristics

Sym	Parameter	V_{CCQ}	2.2 V – 3.3 V		Unit	Test Condition	Notes
			Min	Max			
V_{IL}	Input Low Voltage		0	0.4	V		1
V_{IH}	Input High Voltage		V_{CCQ} –0.4	V_{CCQ}	V		
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			0.4	V		2
V_{LKO}	V_{CC} Lock Voltage		1.0		V		
V_{LKOQ}	V_{CCQ} Lock Voltage		0.9		V		

NOTES:
1. V_{IL} can undershoot to –0.4V and V_{IH} can overshoot to V_{CCQ}+0.4V for durations of 20 ns or less.
2. $V_{PP} < V_{PPLK}$ inhibits erase and program operations. Do not use V_{PPL} and V_{PPH} outside their valid ranges.

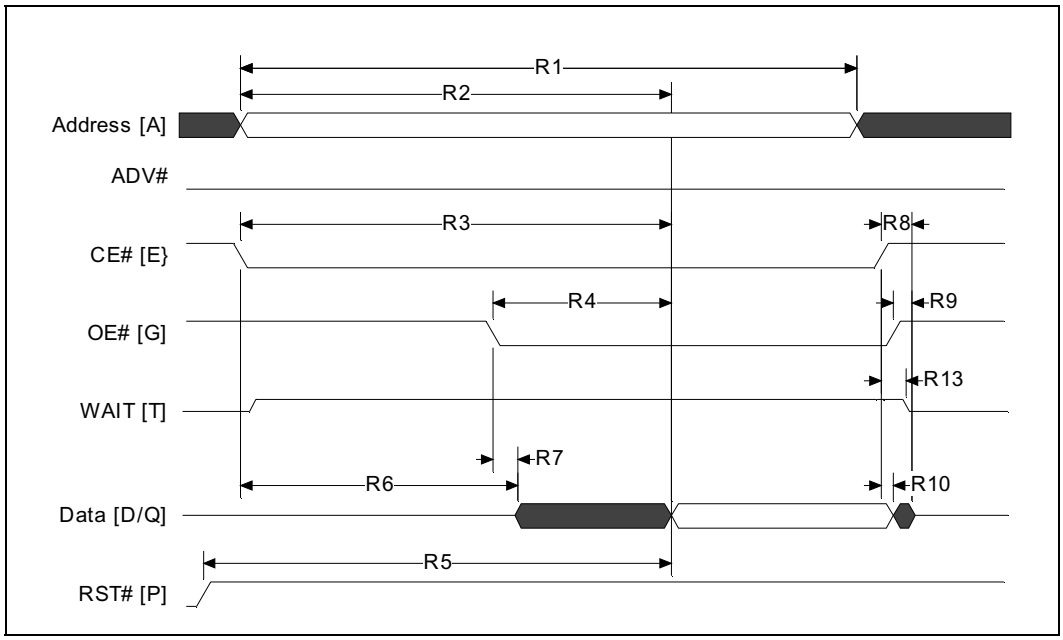
12.0 AC Characteristics

12.1 AC Read Specifications (V_{CCQ} = 2.2 V – 3.3 V)

Num	Symbol	Parameter	Speed	–90		–110		Units	Notes
				Min	Max	Min	Max		
Asynchronous Specifications									
R1	t_{AVAV}	Read cycle time		90		110		ns	6
R2	t_{AVQV}	Address to output valid			90		110	ns	
R3	t_{ELQV}	CE# low to output valid			90		110	ns	
R4	t_{GLQV}	OE# low to output valid			25		30	ns	1,2
R5	t_{PHQV}	RST# high to output valid			150		150	ns	1
R6	t_{ELQX}	CE# low to output in low-Z		0		0		ns	1,3
R7	t_{GLQX}	OE# low to output in low-Z		0		0		ns	1,2,3
R8	t_{EHQZ}	CE# high to output in high-Z			24		24	ns	1,3
R9	t_{GHQZ}	OE# high to output in high-Z			24		24	ns	
R10	t_{OH}	Output hold from first occurring address, CE#, or OE# change		0		0		ns	
R11	t_{EHEL}	CE# pulse width high		20		20		ns	1
R12	t_{ELTV}	CE# low to WAIT valid			16		20	ns	1
R13	t_{EHTZ}	CE# high to WAIT high Z			20		20	ns	1,3
R16	t_{GLTX}	OE# low to WAIT in low-Z		0		0		ns	1,3
R17	t_{GHTZ}	OE# high to WAIT in high-Z			24		24	ns	1,3
Latching Specifications									
R101	t_{AVVH}	Address setup to ADV# high		10		12		ns	1
R102	t_{ELVH}	CE# low to ADV# high		10		12		ns	
R103	t_{VLQV}	ADV# low to output valid			90		110	ns	
R104	t_{VLVH}	ADV# pulse width low		10		12		ns	
R105	t_{VHVL}	ADV# pulse width high		10		12		ns	
R106	t_{VHAX}	Address hold from ADV# high		9		10		ns	1,4
R108	t_{APA}	Page address access			25		25	ns	1
Clock Specifications									
R200	f_{CLK}	CLK frequency			50		40	MHz	1,3
R201	t_{CLK}	CLK period		20		25		ns	
R202	$t_{CH/CL}$	CLK high/low time		9		9		ns	
R203	$t_{FCLK/RCLK}$	CLK fall/rise time			3		3	ns	
Synchronous Specifications									
R301	$t_{AVCH/L}$	Address setup to CLK		9		9		ns	1
R302	$t_{VLCH/L}$	ADV# low setup to CLK		9		9		ns	
R303	$t_{ELCH/L}$	CE# low setup to CLK		9		9		ns	
R304	t_{CHQV} / t_{CLQV}	CLK to output valid			17		20	ns	
R305	t_{CHQX}	Output hold from CLK		3		3		ns	1,5
R306	t_{CHAX}	Address hold from CLK		10		10		ns	1,4,5
R307	t_{CHTV}	CLK to WAIT valid			20		22	ns	1,5

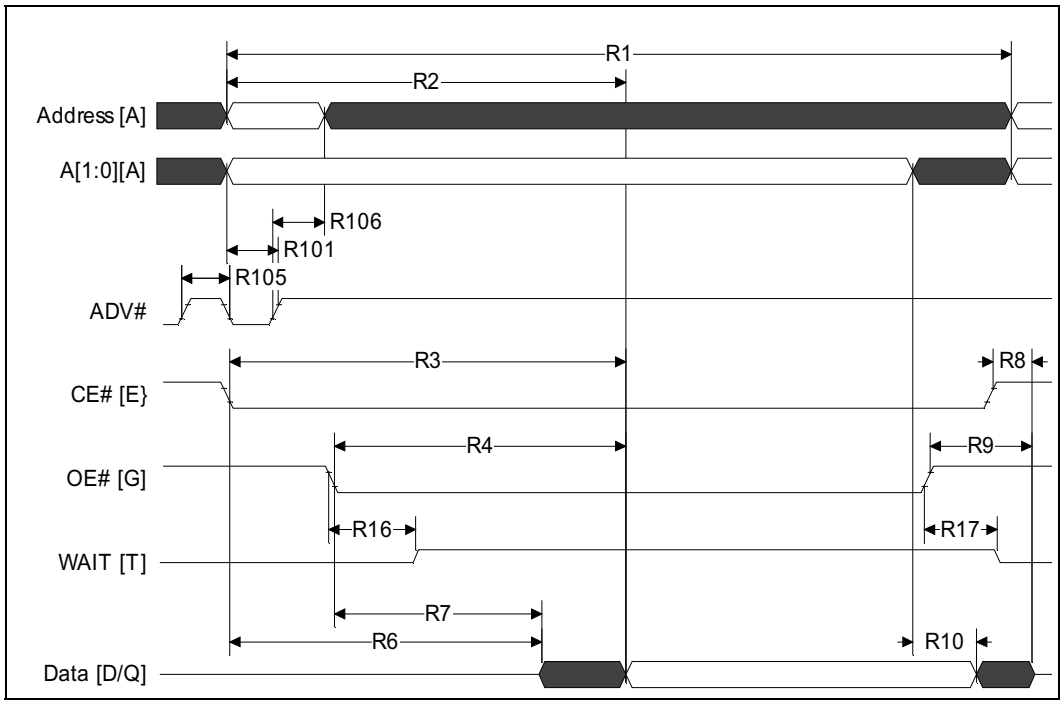
NOTES:
1. See Figure 25, “AC Input/Output Reference Waveform” on page 61 for timing measurements and maximum allowable input slew rate.
2. OE# may be delayed by up to $t_{ELQV} - t_{GLQV}$ after CE#’s falling edge without impact to t_{ELQV}.
3. Sampled, not 100% tested.
4. Address hold in synchronous burst mode is t_{CHAX} or t_{VHAX}, whichever timing specification is satisfied first.
5. Applies only to subsequent synchronous reads.

Figure 12. Asynchronous Single-Word Read (ADV# Low)



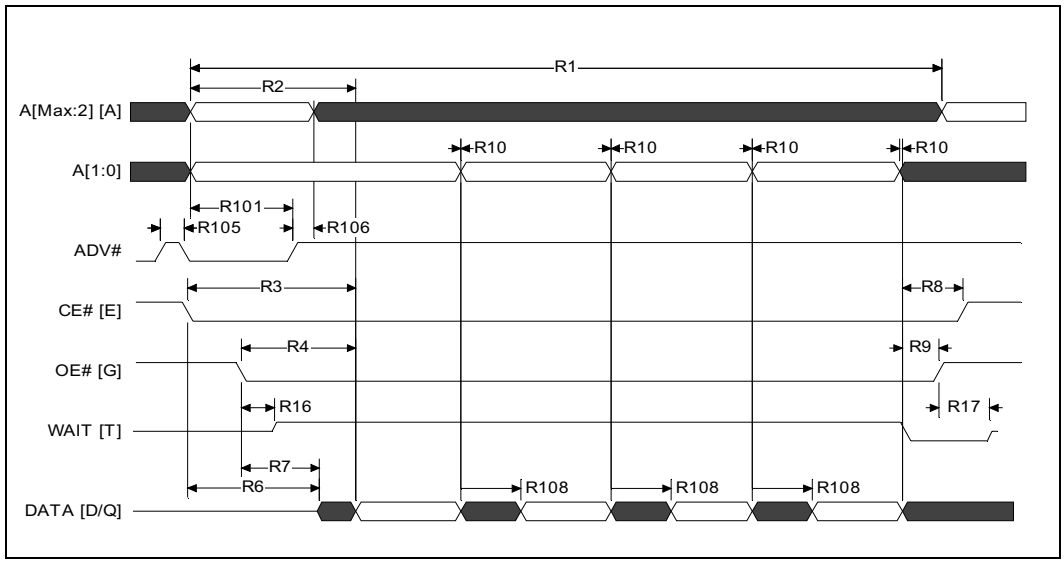
NOTE: WAIT asserted (CR[10]=1) during asynchronous read mode.

Figure 13. Asynchronous Single-Word Read (ADV# Latch)



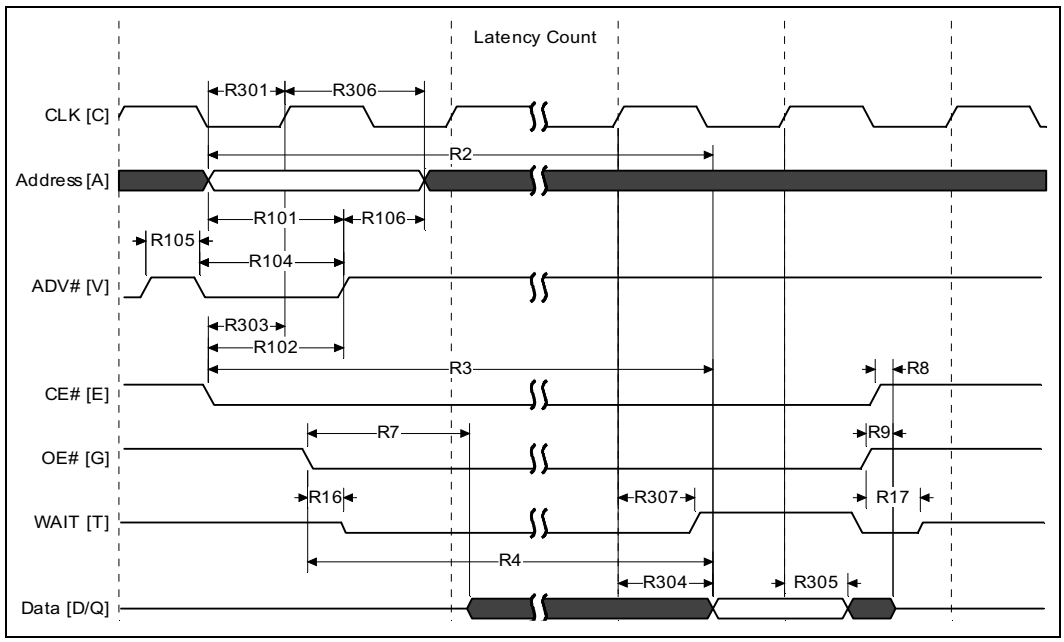
NOTE: WAIT asserted (CR[10]=1) during asynchronous read mode.

Figure 14. Asynchronous Page-Mode Read Timing



NOTE: WAIT asserted (CR[10]=1) during asynchronous read mode.

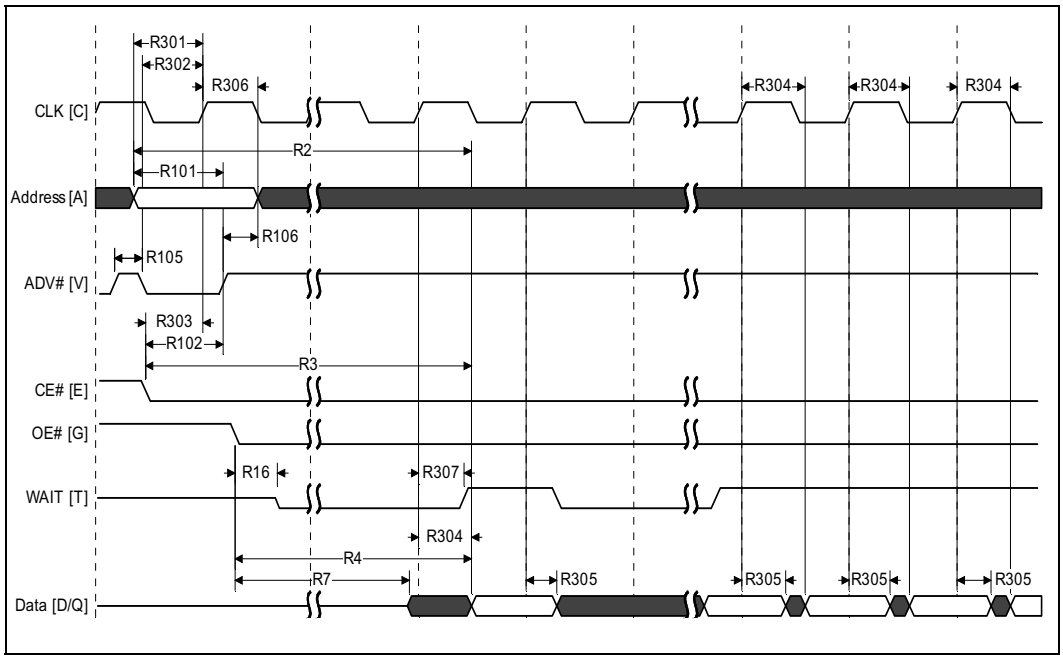
Figure 15. Synchronous Single-Word Array-Read Timing



NOTES:

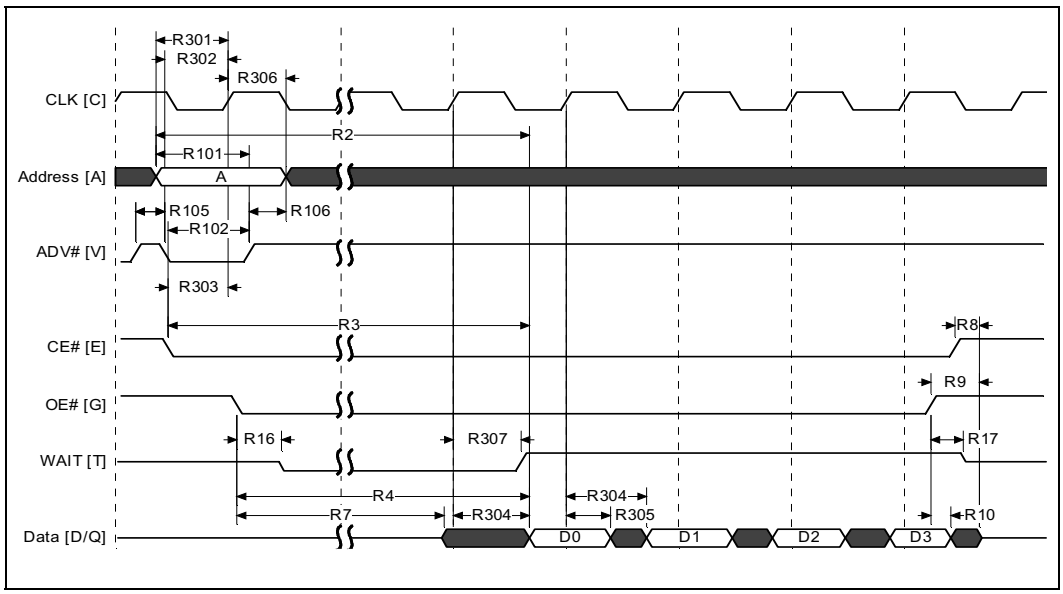
1. WAIT is driven per OE# asserted during synchronous array read, and can be configured to assert either during or one data cycle before valid data.
2. This diagram illustrates the case in which an n-word burst is initiated to the flash memory array and it is terminated by CE# deassertion after the first word in the burst. If this had been an access to Status, ID, or Query space, the active-low WAIT signal would have remained asserted (low) as long as CE# was asserted (low).

Figure 16. Continuous Burst Read, showing an Output Delay Timing



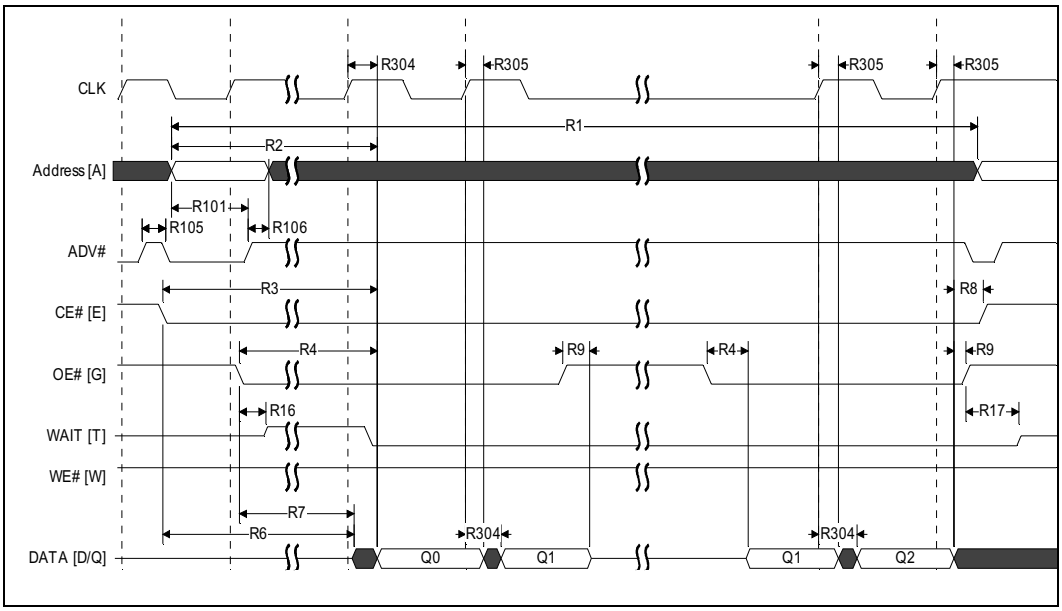
NOTES:
1. At the end of Word Line; the delay incurred when a burst access crosses a 16-word boundary and the starting address is not 4-word boundary aligned.
2. WAIT asserted (CR[10]=0) during synchronous array read, and can be configured to assert either during or one data cycle before valid data.

Figure 17. Synchronous Burst-Mode Four-Word Read Timing



NOTE: WAIT asserted (CR[10]=0) during synchronous array read, and can be configured to assert either during or one data cycle before valid data.

Figure 18. Burst Suspend Timing



NOTE: CLK can be stopped in either high or low state.

12.2 AC Write Specifications

Nbr.	Symbol	Parameter [1, 2]	Min	Max	Units	Notes
W1	t_{PHWL}	RST# high recovery to WE# low	150		ns	1,2,3
W2	t_{ELWL}	CE# setup to WE# low	0		ns	1,2,3
W3	t_{WLWH}	WE# write pulse width low	50		ns	1,2,4
W4	t_{DVWH}	Data setup to WE# high	50		ns	1,2
W5	t_{AVWH}	Address setup to WE# high	50		ns	
W6	t_{WHEH}	CE# hold from WE# high	0		ns	
W7	t_{WHDX}	Data hold from WE# high	0		ns	
W8	t_{WHAX}	Address hold from WE# high	0		ns	
W9	t_{WHWL}	WE# pulse width high	20		ns	1,2,5
W10	t_{VPWH}	V_{PP} setup to WE# high	200		ns	1,2,3,7
W11	t_{QVVL}	V_{PP} hold from Status read	0		ns	
W12	t_{QVBL}	WP# hold from Status read	0		ns	1,2,3,7
W13	t_{BHWH}	WP# setup to WE# high	200		ns	
W14	t_{WHGL}	WE# high to OE# low	0		ns	1,2,9
W16	t_{WHQV}	WE# high to read valid	t_{AVQV}+35		ns	1,2,3,6,10
Write to Asynchronous Read Specifications						
W18	t_{WHAV}	WE# high to Address valid	0		ns	1,2,3,6
Write to Synchronous Read Specifications						
W19	$t_{WHCH/L}$	WE# high to Clock valid	19		ns	1,2,3,6,10
W20	t_{WHVH}	WE# high to ADV# high	19		ns	

NOTES:
1. Write timing characteristics during erase suspend are the same as write-only operations.
2. A write operation can be terminated with either CE# or WE#.
3. Sampled, not 100% tested.
4. Write pulse width low (t_{WLWH} or t_{ELEH}) is defined from CE# or WE# low (whichever occurs last) to CE# or WE# high (whichever occurs first). Hence, $t_{WLWH} = t_{ELEH} = t_{WLEH} = t_{ELWH}$.
5. Write pulse width high (t_{WHWL} or t_{EHEL}) is defined from CE# or WE# high (whichever occurs first) to CE# or WE# low (whichever occurs last). Hence, $t_{WHWL} = t_{EHEL} = t_{WHEL} = t_{EHWL}$).
6. t_{WHVH} or $t_{WHCH/L}$ must be met when transitioning from a write cycle to a synchronous burst read.
7. V_{PP} and WP# should be at a valid level until erase or program success is determined.
8. This specification is only applicable when transitioning from a write cycle to an asynchronous read. See spec W19 and W20 for synchronous read.
9. When doing a Read Status operation following a program or erase write cycle, W14 is 20ns.
10. Add 10ns if the write operations results in a RCR or block lock status change, for the subsequent read operation to reflect this change.

Figure 19. Write to Write Timing

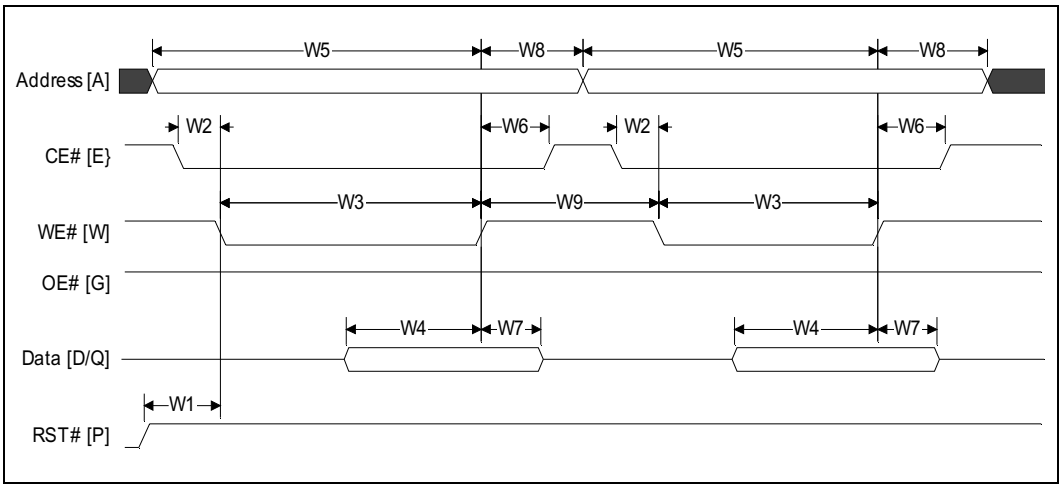


Figure 20. Asynchronous Read to Write Timing

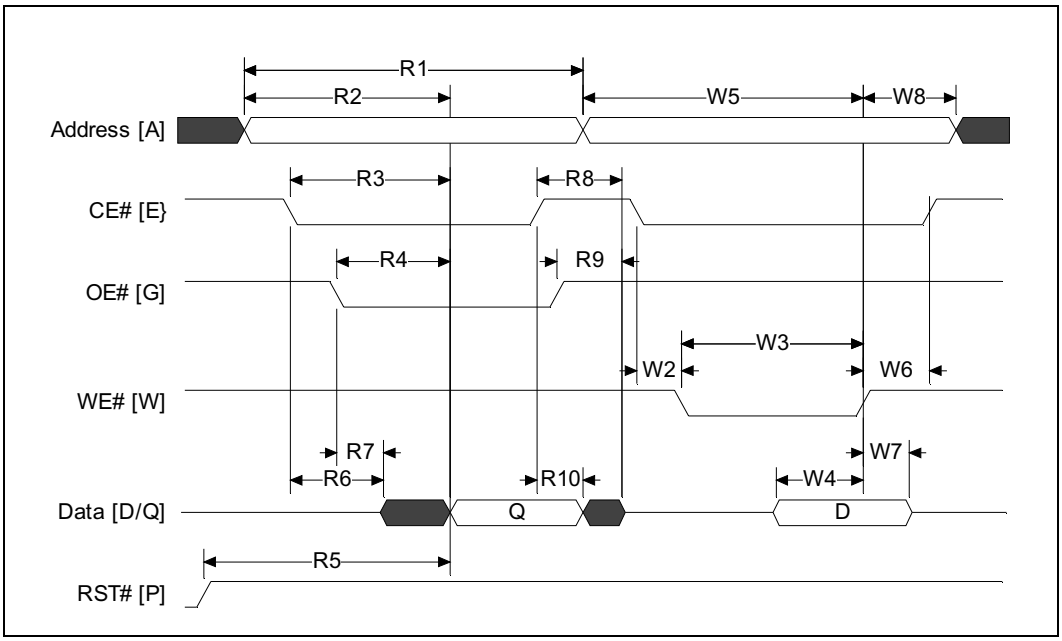


Figure 21. Write to Asynchronous Read Timing

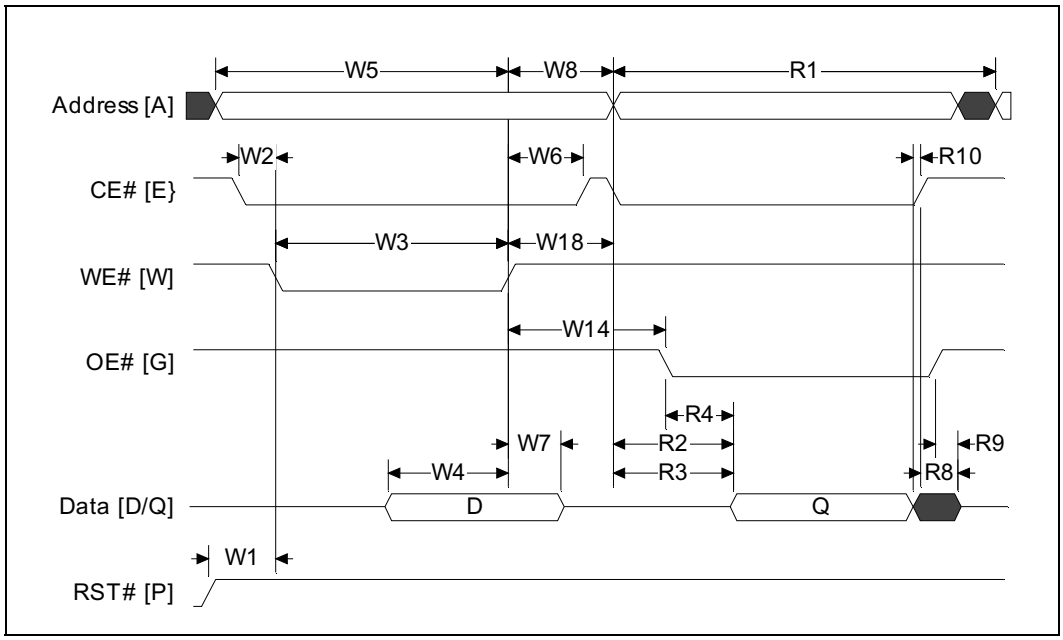
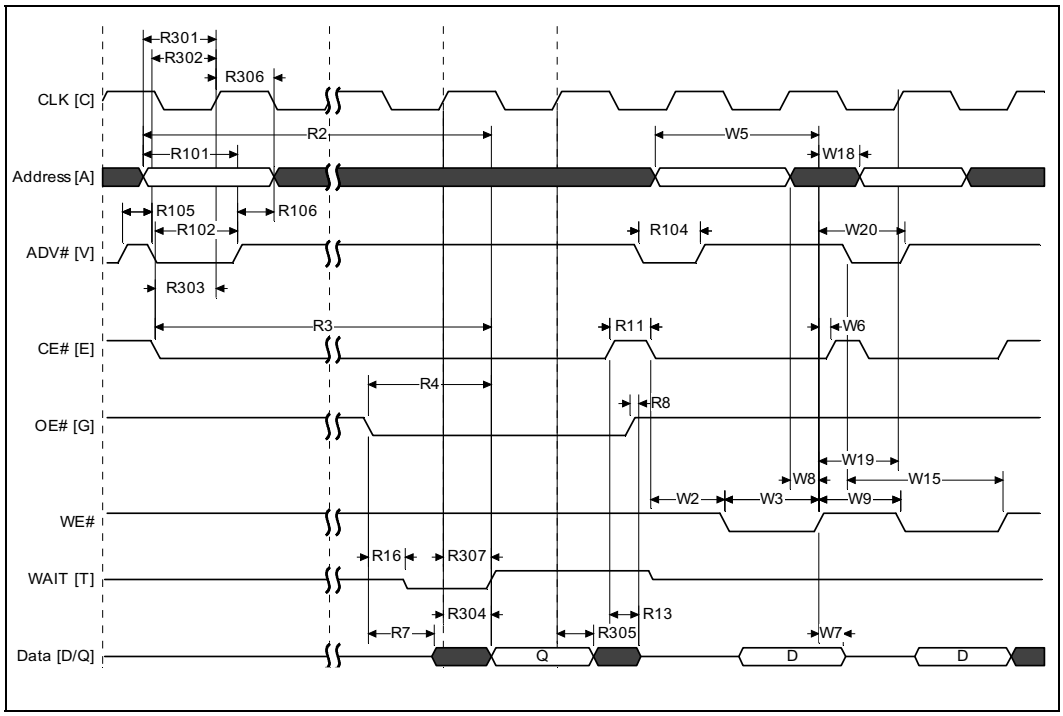
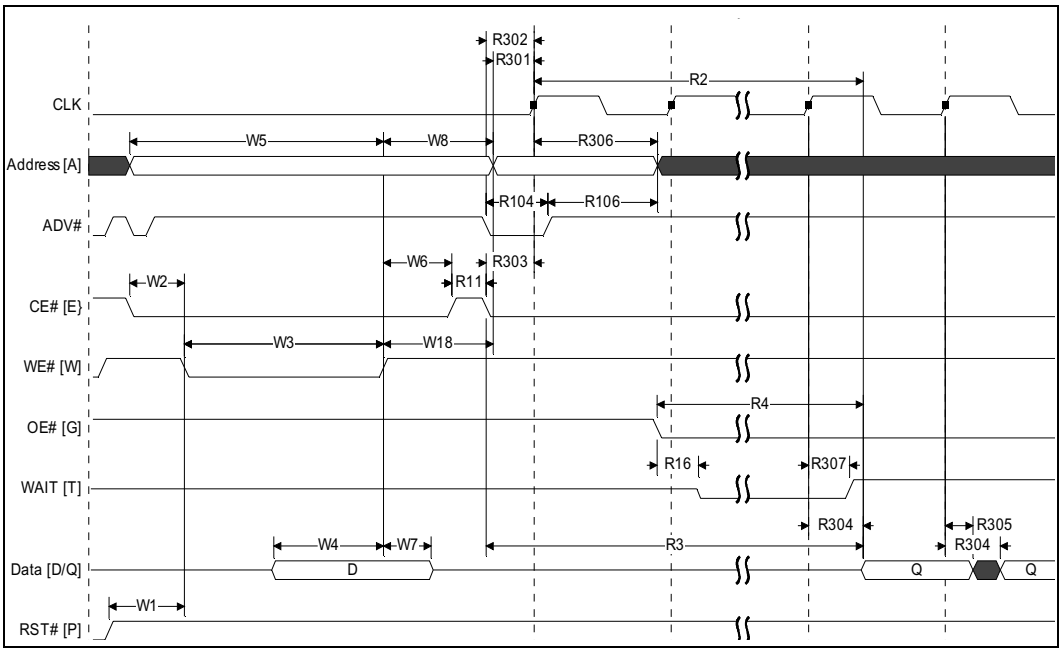


Figure 22. Synchronous Read to Write Timing



NOTE: WAIT shown asserted (CR[10]=0) during write operation.

Figure 23. Write to Synchronous Read Timing



NOTE: WAIT shown asserted (CR[10]=0) during write operation.

12.3 Program and Erase Characteristics

Nbr.	Symbol	Parameter		V_{PPL}			V_{PPH}			Units	Notes
				Min	Typ	Max	Min	Typ	Max		
Conventional Word Programming											
W200	$t_{PROG/W}$	Program Time	Single word		150	TBD		150	TBD	µs	1
			Single cell		30	TBD		30	TBD		
Buffered Programming											
W200	$t_{PROG/W}$	Program Time	Single word		150	TBD		150	TBD	µs	1
W201	$t_{PROG/PB}$		One Buffer (32 words)		640	TBD		288	864		
Buffered Enhanced Factory Programming											
W451	$t_{BEFP/W}$	Program	Single word	N/A	N/A	N/A	N/A	7	21	µs	1,2
W452	$t_{BEFP/Setup}$		Buffered EFP Setup	N/A	N/A	N/A	5	N/A	N/A		1
Erasing and Suspending											
W500	$t_{ERS/PB}$	Erase Time	16-KWord Parameter		0.4	2.5		0.4	2.5	s	1
W501	$t_{ERS/MB}$		64-KWord Main		0.8	4		0.7	4		
W600	$t_{SUSP/P}$	Suspend Latency	Program suspend		20	25		20	25	µs	
W601	$t_{SUSP/E}$		Erase suspend		20	25		20	25		

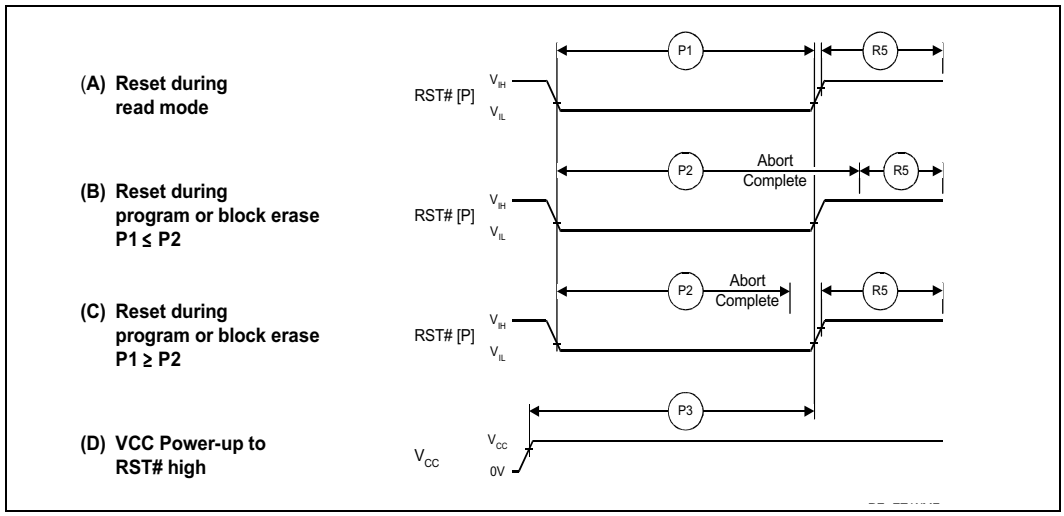
NOTES:
1. Typical values measured at T_A = +25 °C and nominal voltages. Performance numbers are valid for all speed versions. Excludes system overhead. Sampled, but not 100% tested.
2. Averaged over entire device.

12.4 Reset Specifications

Nbr.	Symbol	Parameter	Min	Max	Unit	Notes
P1	t_{PLPH}	RST# pulse width low	100		ns	1,2,3,4
P2	t_{PLRH}	RST# low to device reset during erase		25	µs	1,3,4,7
		RST# low to device reset during program		25		1,3,4,7
P3	t_{VCCPH}	V_{CC} Power valid to RST# de-assertion (high)	60			1,4,5,6

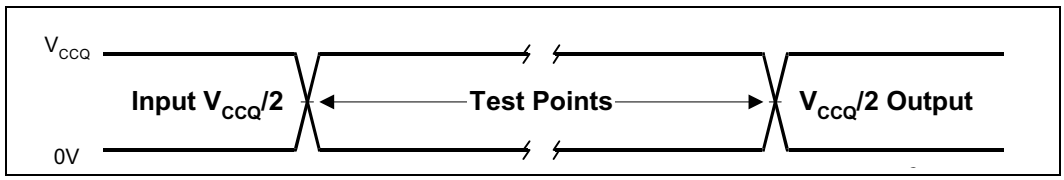
NOTES:
1. These specifications are valid for all device versions (packages and speeds).
2. The device may reset if t_{PLPH} is <t_{PLPH} MIN, but this is not guaranteed.
3. Not applicable if RST# is tied to Vcc.
4. Sampled, but not 100% tested.
5. If RST# is tied to the V_{CC} supply, device will not be ready until t_{VCCPH} after V_{CC} >= V_{CC} min.
6. If RST# is tied to any supply/signal with V_{CCQ} voltage levels, the RST# input voltage must not exceed V_{CC} until V_{CC} >= V_{CC}(min).
7. Reset completes within t_{PLPH} if RST# is asserted while no erase or program operation is executing.

Figure 24. Reset Operation Waveforms



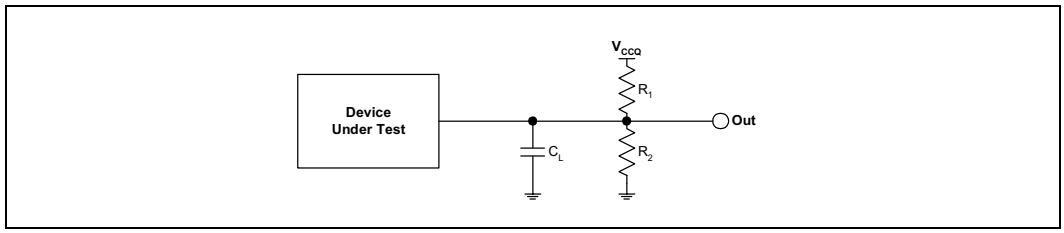
12.5 AC Test Conditions

Figure 25. AC Input/Output Reference Waveform



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

Figure 26. Transient Equivalent Testing Load Circuit

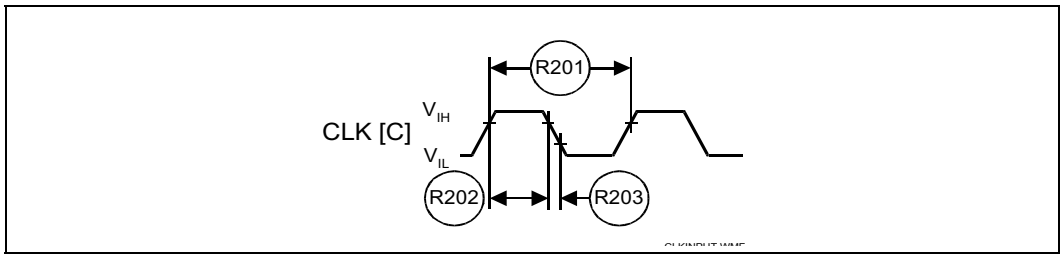


NOTES:
1. See the following table for component values.
2. Test configuration component value for worst case speed conditions.
3. C_L includes jig capacitance

Table 15. Test configuration component value for worst case speed conditions

Test Configuration	C_L (pF)	R_1 (Ω)	R_2 (Ω)
2.0 V Standard Test	30	22K	22K

Figure 27. Clock Input AC Waveform



12.6 Capacitance

Table 16. Capacitance

Symbol	Parameter[1]	Typ	Max	Unit	Condition
C_{IN}	Input Capacitance	6	8	pF	V_{IN} = 0.0 V
C_{OUT}	Output Capacitance	8	12	pF	V_{OUT} = 0.0 V
$C_{CE\#}$	CE# Input Capacitance	10	12	pF	V_{IN} = 0.0 V

NOTES:
1. TA = +25°C, f = 1 MHz.
2. Sampled, not 100% tested.

Appendix A Write State Machine (WSM)

Figure 28 shows the command state transitions (Next State Table) based on incoming commands. Only one partition can be actively programming or erasing at a time. Each partition stays in its last read state (Read Array, Read Device ID, CFI Query or Read Status Register) until a new command changes it. The next WSM state does not depend on the partition's output state.

Figure 28. Write State Machine — Next State Table (Sheet 1 of 6)

		Command Input to Chip and resulting *Chip* Next State										
Current Chip State (8)		Read Array (3)	Word Program (4,5)	Write to Buffered Program (BP)	Erase Setup (4,5)	Buffered Enhanced Factory Pgm Setup (4)	BE Confirm, P/E Resume, ULB, Confirm (9)	BP / Prg / Erase Suspend	Read Status	Clear Status Register (6)	Read ID/Query	Lock, Unlock, Lock-down, CR setup (5)
		(FFH)	(10H/40H)	(E8H)	(20H)	(80H)	(D0H)	(B0H)	(70H)	(50H)	(90H, 98H)	(60H)
Ready		Ready	Program Setup	BP Setup	Erase Setup	BEFP Setup	Ready					Lock/CR Setup
Lock/CR Setup		Ready (Lock Error [Botch])					Ready (Unlock Block)	Ready (Lock Error [Botch])				
OTP	Setup	OTP Busy										
	Busy											
Word Program	Setup	Word Program Busy										
	Busy	Program Busy						Word Program Suspend	Word Program Busy			
	Suspend	Word Program Suspend					Word Program Busy	Word Program Suspend				
BP	Setup	BP Load 1 {Give word count load [N-1]}; If N=0 (word count =1) go to BP Confirm; Else (N not = 0) go to BP Load 2										
	BP Load 1	BP Load 2 (Give data load)										
	BP Load 2	BP Confirm when count=0, ELSE BP load 2 (note: BP will Botch at this point if any block address is different from the first address)										
	BP Confirm	Ready (Error [Botch])					BP Busy	Ready (Error [Botch])				
	BP Busy	BP Busy						BP Suspend	BP Busy			
	BP Suspend	BP Suspend					BP Busy	BP Suspend				
Erase	Setup	Ready (Error [Botch])					Erase Busy	Ready (Error [Botch])				
	Busy	Erase Busy						Erase Suspend	Erase Busy			
	Suspend	Erase Suspend	Word Program Setup in Erase Suspend	BP Setup in Erase Suspend	Erase Suspend		Erase Busy	Erase Suspend				Lock/CR Setup in Erase Suspend
Word Program in Erase Suspend	Setup	Word Program Busy in Erase Suspend										
	Busy	Word Program Busy in Erase Suspend						Word Program Suspend in Erase Suspend	Word Program Busy in Erase Suspend Busy			
	Suspend	Word Program Suspend in Erase Suspend					Word Program Busy in Erase Suspend	Word Program Suspend in Erase Suspend				

Figure 28. Write State Machine — Next State Table (Sheet 2 of 6)

BP in Erase Suspend	Setup	BP Load 1 in Erase Suspend {Give word count load [N-1]}; If N=0 (word count =1) go to BP Confirm; Else (N not = 0) go to BP Load 2			
	BP Load 1	BP Load 2 in Erase Suspend (Give data load)			
	BP Load 2	BP Confirm in Erase Suspend when count=0, ELSE BP load 2 (note: BP will Botch at this point if any block address is different from the first address)			
	BP Confirm	Erase Suspend (Error [Botch BP])	BP Busy in Erase Suspend	Ready (Error [Botch BP] in Erase Suspend)	
	BP Busy	BP Busy in Erase Suspend		BP Suspend in Erase Suspend	BP Busy in Erase Suspend
	BP Suspend	BP Suspend in Erase Suspend	BP Busy in Erase Suspend	BP Suspend in Erase Suspend	
Lock/CR Setup in Erase Suspend		Erase Suspend (Lock Error [Botch])	Erase Suspend (Unlock Block)	Erase Suspend (Lock Error [Botch])	
Buffered Enhanced Factory Program Mode	Setup	Ready (Error [Botch])	BEFP Loading Data (X=32)	Ready (Error [Botch])	
	BEFP Busy	BEFP Program and Verify Busy (if Block Address given matches address given on BEFP Setup command). Commands treated as data. (7)			

Figure 28. Write State Machine — Next State Table (Sheet 3 of 6)

Tyax *Output* Next State Table

Current chip state	Command Input to Chip and resulting *Output* Mux Next State										
	Read Array (3)	Word Program Setup (4,5)	BP Setup	Erase Setup (4,5)	Buffered Enhanced Factory Pgm Setup (4)	BE Confirm, P/E Resume, ULB Confirm (9)	Program/ Erase Suspend	Read Status	Clear Status Register (6)	Read ID/Query	Lock, Unlock, Lock-down, CR setup (5)
	(FFH)	(10H/40H)	(E8H)	(20H)	(30H)	(D0H)	(B0H)	(70H)	(50H)	(90H, 98H)	(60H)
BEFP Setup, BEFP Pgm & Verify Busy, Erase Setup, OTP Setup, BP: Setup, Load 1, Load 2, Confirm, Word Pgm Setup, Word Pgm Setup in Erase Susp, BP Setup, Load1, Load 2, Confirm in Erase Suspend	Status Read										
Lock/CR Setup, Lock/CR Setup in Erase Susp	Status Read										
OTP Busy	Read Array	Status Read				Output mux does not change.	Status Read	Output mux does not change.	Status Read	Status Read	Status Read
Ready, Erase Suspend, BP Suspend										ID Read	
BP Busy, Word Program Busy, Erase Busy, BP Busy, BP Busy in Erase Suspend, Word Pgm Suspend, Word Pgm Busy in Erase Suspend, Pgm Suspend In Erase Suspend											

Figure 28. Write State Machine — Next State Table (Sheet 4 of 6)

Command Input to Chip and resulting *Chip* Next State						
OTP Setup (5)	Lock Block Confirm (9)	Lock-Down Block Confirm (9)	Write CR Confirm (9)	Block Address (WA0)	Illegal Cmds or BEFP Data (2)	WSM Operation Completes
(C0H)	(01H)	(2FH)	(03H)	(XXXXH)	(all other codes)	
OTP Setup	Ready					N/A
Ready (Lock Error [Botch])	Ready (Lock Block)	Ready (Lock Down Blk)	Ready (Set CR)	Ready (Lock Error [Botch])		
OTP Busy						Ready
Word Program Busy						N/A
Word Program Busy						Ready
Word Program Suspend						N/A
BP Load 1 {Give word count load [N-1]}; If N=0 (word count =1) go to BP Confirm; Else (N not = 0) go to BP Load 2						
BP Load 2 (Give data load)				Exit		
BP Confirm when count=0, ELSE BP load 2 (note: BP will Botch at this point if any block address is different from the first address)						
Ready (Error [Botch])						
BP Busy						Ready
BP Suspend						N/A
Ready (Error [Botch])						
Erase Busy						Ready
Erase Suspend						N/A
Word Program Busy in Erase Suspend						
Word Program Busy in Erase Suspend Busy						Erase Suspend
Word Program Suspend in Erase Suspend						N/A

Figure 28. Write State Machine — Next State Table (Sheet 5 of 6)

BP Load 1 in Erase Suspend {Give word count load [N-1]}; If N=0 (word count =1) go to BP Confirm; Else N ? 0 go to BP Load 2					N/A
BP Load 2 in Erase Suspend (Give data load)				Exit	
BP Confirm in Erase Suspend when count=0, ELSE BP load 2 (note: BP will Botch at this point if any block address is different from the first address)					
Ready (Error [Botch BP] in Erase Suspend)					
BP Busy in Erase Suspend					Erase Suspend
BP Suspend in Erase Suspend					N/A
Erase Suspend (Lock Error [Botch])	Erase Suspend (Lock Block)	Erase Suspend (Lock Down Block)	Erase Suspend (Set CR)	Erase Suspend (Lock Error [Botch])	
Ready (Error [Botch])					
BEFP Program and Verify Busy (if Block Address given matches address given on BEFP Setup command). Commands treated as data. (7)				Ready	BEFP Busy	Ready

Figure 28. Write State Machine — Next State Table (Sheet 6 of 6)

Tyax *Output* Next State Table

Command Input to Chip and resulting *Output* Mux Next State						
OTP Setup (5)	Lock Block Confirm (9)	Lock-Down Block Confirm (9)	Write CR Confirm (9)	Block Address (WA0)	Illegal Cmds or BEFP Data (2)	WSM Operation Completes
(C0H)	(01H)	(2FH)	(03H)	(FFFFH)	(all other codes)	
Status Read						Output mux does not change.
Status Read			Array Read	Status Read		
Status Read	Output mux does not change.			Array Read	Output mux does not change.	

NOTES:

1. The "Partition Data When Read" field shows what the user will read from the flash chip after issuing the appropriate command given the Partition Address is not changed from the address given during the command. "Read-while-write" functionality gives more flexibility in data output from the device. The data read from the chip depends on the Partition Address applied to the device; Each partition is placed into one of 3 possible output states during commands: Read Array, Read Status or Read ID/CFI, depending on the command given to the chip; This partition's output state is retained until a new command is given to the chip at that Partition Address; For example, this allows the user to set partition #1's output state to Read Array, and partition #4's output state to Read Status; Every time the partition address is changed to partition #4 (without issuing a new command), the Status will be read from the chip.
2. "Illegal commands" include commands outside of the allowed command set (allowed commands: 40H [pgm], 20H [erase], etc.)
3. If a "Read Array" is attempted from a busy partition, the result will be "garbage" data (we should not tell the user that it will actually be Status Register data). The key point is that the output mux for that partition will be pointing to the "array", but garbage data will be output. When the user returns to this partition address some time in the future, the output mux will be in the "Read Array" state from its last visit. "Read ID" and "Read Query" commands do the exact same thing in the device. The ID and Query data are located at different locations in the address map.
4. 1st and 2nd cycles of "2 cycles write commands" must be given to the same partition address, or unexpected results will occur.
5. The 2nd cycle of the following 2 cycle commands will be ignored by the user interface: Program Setup, Erase Setup, OTP Setup and Lock/Unlock/Lock-down/CR setup when issued in an "illegal condition". Illegal conditions are such as "pgm setup while busy", "erase setup while busy", etc.
6. The Clear Status command only clears the error bits in the status register if the device is not in the following modes: WSM running (Pgm Busy, Erase Busy, Pgm Busy In Erase Suspend, OTP Busy, BEFP modes).
7. BEFP writes are only allowed when the status register bit #0 = 0, or else the data is ignored.
8. The "current state" is that of the "chip" and not of the "partition"; Each partition "remembers" which output (Array, ID/CFI or Status) it was last pointed to on the last instruction to the "chip", but the next state of the chip does not depend on where the partition's output mux is presently pointing to.
9. Confirm commands (Lock Block, Unlock Block, Lock-Down Block, Configuration Register) perform the operation and then move to the Ready State.

Appendix B Flowcharts

Figure 29. Word Program Flowchart

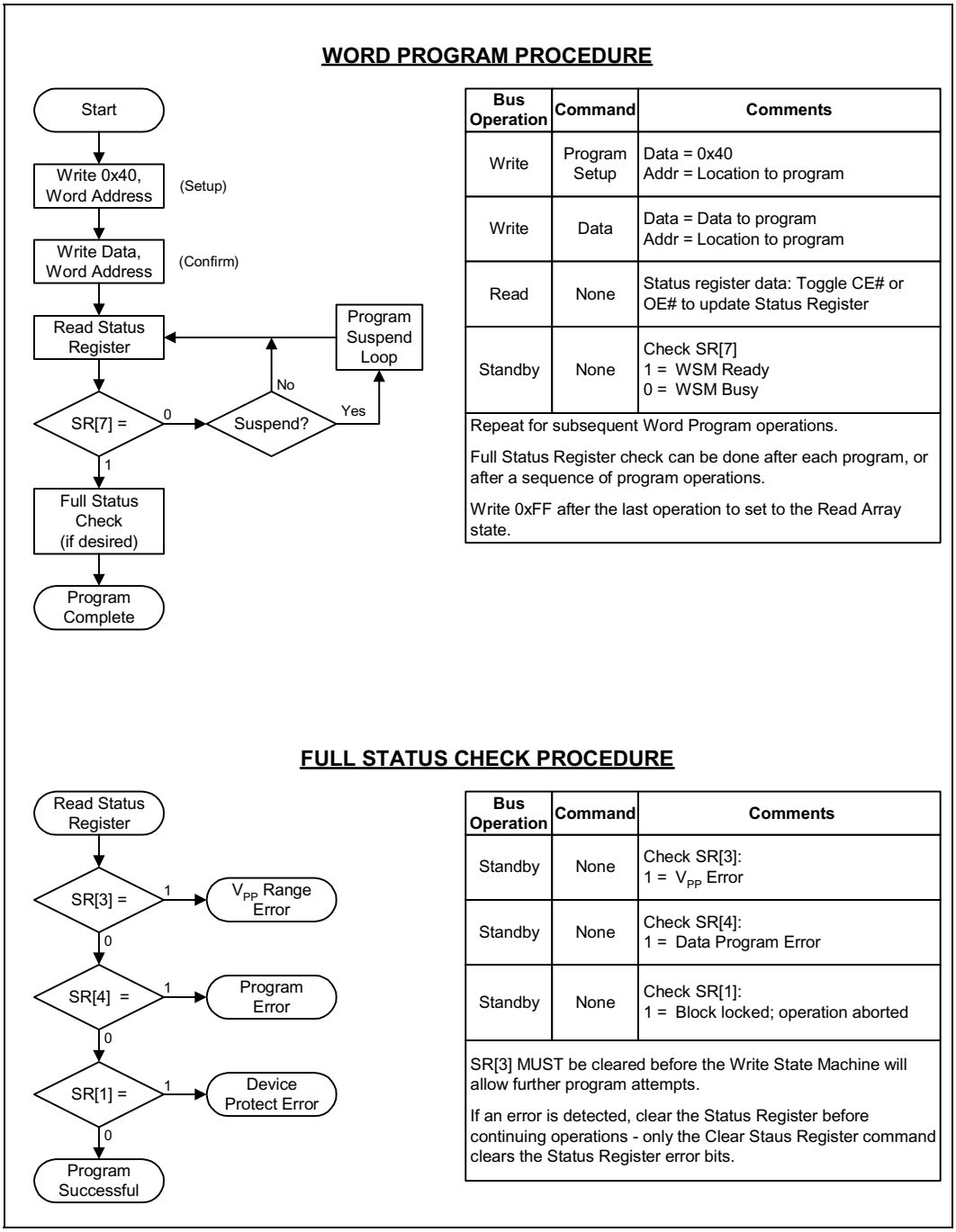
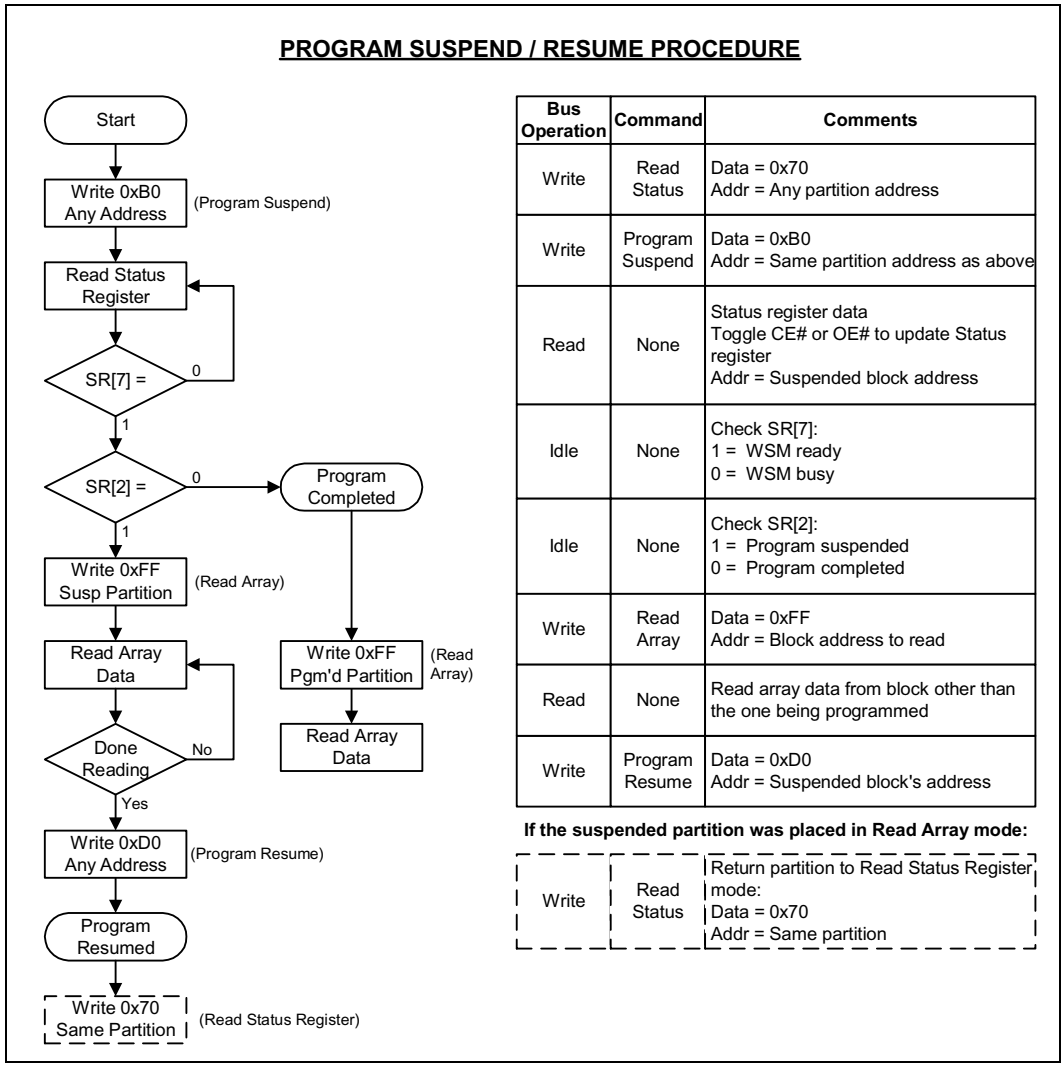


Figure 30. Program Suspend/Resume Flowchart

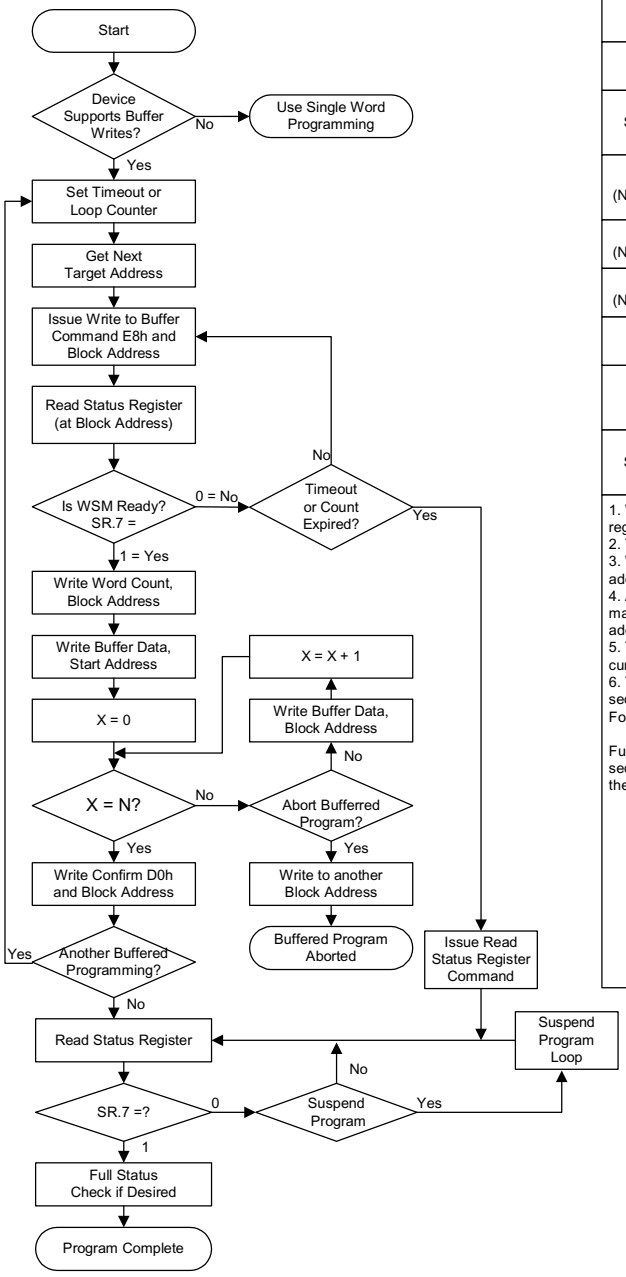


Bus Operation	Command	Comments
Write	Read Status	Data = 0x70 Addr = Any partition address
Write	Program Suspend	Data = 0xB0 Addr = Same partition address as above
Read	None	Status register data Toggle CE# or OE# to update Status register Addr = Suspended block address
Idle	None	Check SR[7]: 1 = WSM ready 0 = WSM busy
Idle	None	Check SR[2]: 1 = Program suspended 0 = Program completed
Write	Read Array	Data = 0xFF Addr = Block address to read
Read	None	Read array data from block other than the one being programmed
Write	Program Resume	Data = 0xD0 Addr = Suspended block's address

If the suspended partition was placed in Read Array mode:

Write	Read Status	Return partition to Read Status Register mode: Data = 0x70 Addr = Same partition

Figure 31. Buffered Program Flowchart

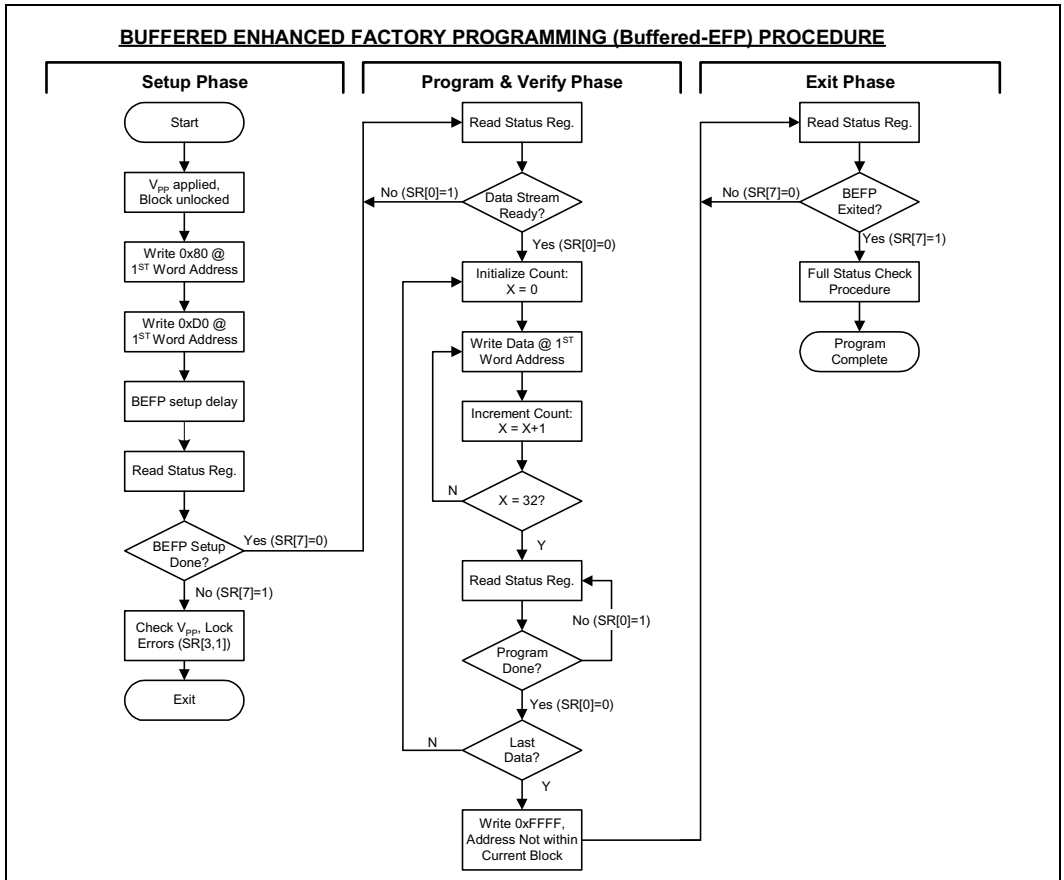


Bus Operation	Command	Comments
Write	Write to Buffer	Data = E8H Addr = Block Address
Read		SR.7 = Valid Addr = Block Address
Standby		Check SR.7 1 = Device WSM is Busy 0 = Device WSM is Ready
Write (Notes 1, 2)		Data = N-1 = Word Count N = 0 corresponds to count = 1 Addr = Block Address
Write (Notes 3, 4)		Data = Write Buffer Data Addr = Start Address
Write (Notes 5, 6)		Data = Write Buffer Data Addr = Block Address
Write	Program Confirm	Data = D0H Addr = Block Address
Read		Status register Data CE# and OE# low updates SR Addr = Block Address
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy

1. Word count values on DQ_0-DQ_7 are loaded into the Count register. Count ranges for this device are N = 0000h to 0001Fh.
2. The device outputs the status register when read.
3. Write Buffer contents will be programmed at the device start address or destination flash address.
4. Align the start address on a Write Buffer boundary for maximum programming performance (i.e., A_4–A_0 of the start address = 0).
5. The device aborts the Buffered Program command if the current address is outside the original block address.
6. The Status register indicates an "improper command sequence" if the Buffered Program command is aborted. Follow this with a Clear Status Register command.

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

Figure 32. Buffered EFP Flowchart



BEFP Setup

Bus State	Operation	Comments
Write	Unlock Block	V_{PPH} applied to VPP
Write (Note 1)	BEFP Setup	Data = 0x80 @ 1ST Word Address
Write	BEFP Confirm	Data = 0x80 @ 1ST Word Address
Read (Note 2)	Status Register	Data = Status Reg. Data Address = 1ST Word Addr
Standby	BEFP Setup Done?	Check SR[7]: 0 = BEFP Ready 1 = BEFP Not Ready
Standby	Error Condition Check	If SR[7] is set, check: SR[3] set = V_{PP} Error SR[1] set = Locked Block

BEFP Program & Verify

Bus State	Operation	Comments
Read	Status Register	Data = Status Register Data Address = 1ST Word Addr.
Standby	Data Stream Ready?	Check SR[0]: 0 = Ready for Data 1 = Not Ready for Data
Standby	Initialize Count	X = 0
Write (Note 3)	Load Buffer	Data = Data to Program Address = 1ST Word Addr.
Standby	Increment Count	X = X+1
Standby	Buffer Full?	X = 32? Yes = Read SR[0] No = Load Next Data Word
Read	Status Register	Data = Status Reg. data Address = 1ST Word Addr.
Standby	Program Done?	Check SR[0]: 0 = Program Done 1 = Program in Progress
Standby	Last Data?	No = Fill buffer again Yes = Exit
Write	Exit Program & Verify Phase	Data = 0xFFFF @ address not in current block

BEFP Exit

Bus State	Operation	Comments
Read	Status Register	Data = Status Reg. Data Address = 1ST Word Addr
Standby	Check Exit Status	Check SR[7]: 0 = Exit Not Completed 1 = Exit Completed

Repeat for subsequent blocks;

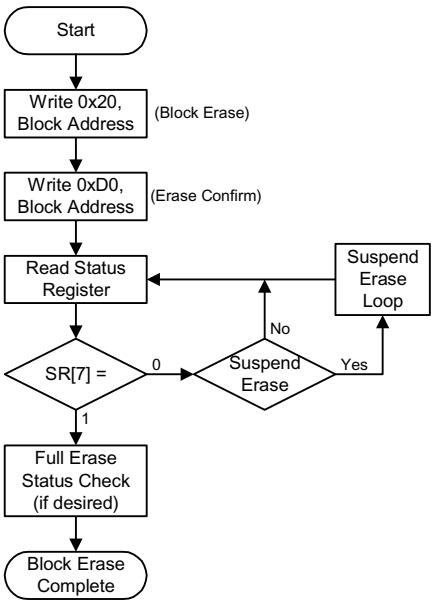
After BEFP exit, a full Status Register check can determine if any program error occurred;

See full Status Register check procedure in the Word Program flowchart.

Write 0xFF to enter Read Array state.

NOTES:
1. First-word address to be programmed within the target block must be aligned on a write-buffer boundary.
2. Status Register is updated when OE# is toggled.
3. Write-buffer contents are programmed sequentially to the flash array starting at the first word address; WSM internally increments addressing.

Figure 33. Block Erase Flowchart



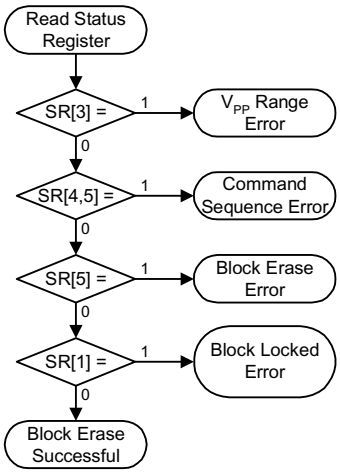
BLOCK ERASE PROCEDURE

Bus Operation	Command	Comments
Write	Block Erase Setup	Data = 0x20 Addr = Block to be erased (BA)
Write	Erase Confirm	Data = 0xD0 Addr = Block to be erased (BA)
Read	None	Status Register data. Toggle CE# or OE# to update Status register data
Idle	None	Check SR[7]: 1 = WSM ready 0 = WSM busy

Repeat for subsequent block erasures.

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

Write 0xFF after the last operation to enter read array mode.



FULL ERASE STATUS CHECK PROCEDURE

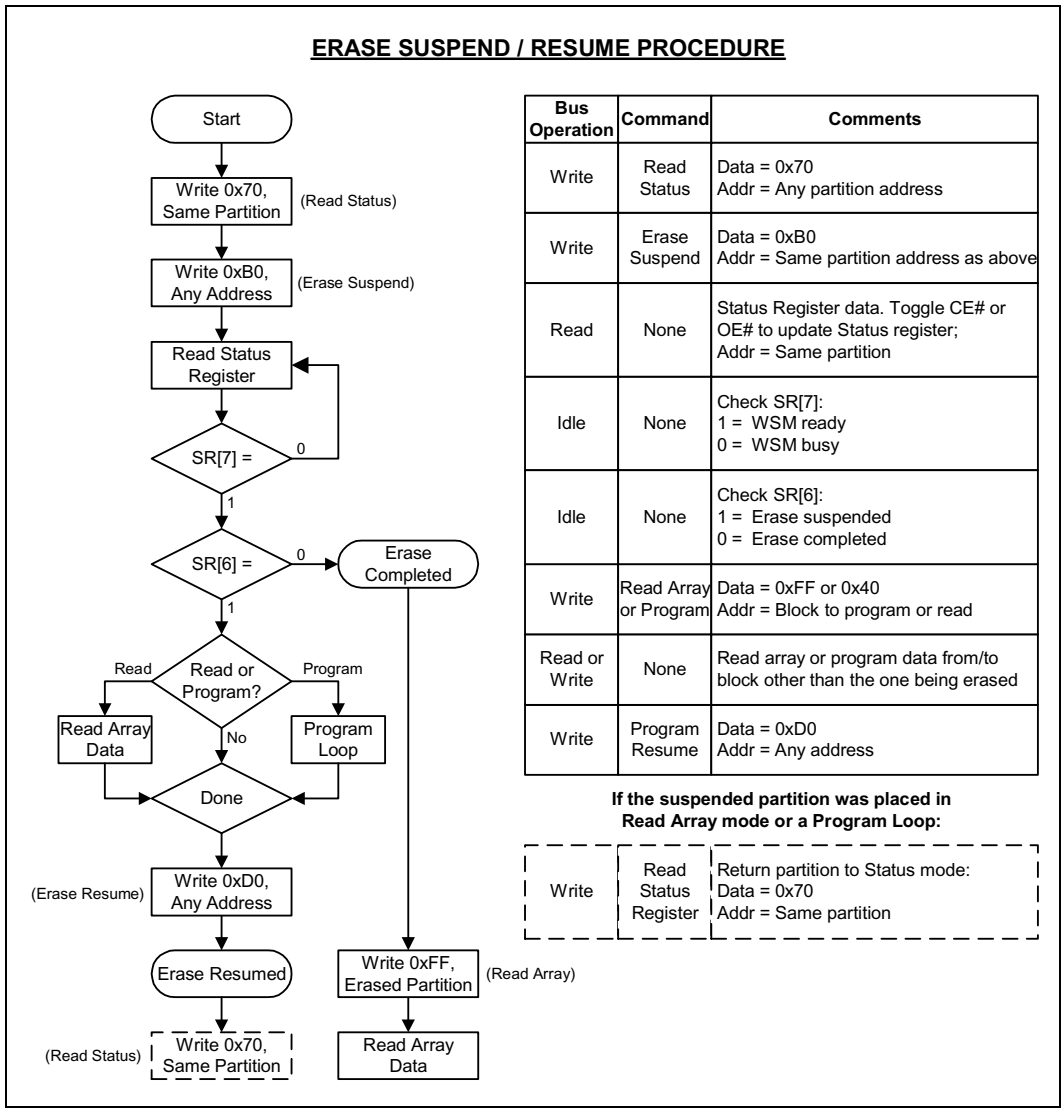
Bus Operation	Command	Comments
Idle	None	Check SR[3]: 1 = V_{PP} Range Error
Idle	None	Check SR[4,5]: Both 1 = Command Sequence Error
Idle	None	Check SR[5]: 1 = Block Erase Error
Idle	None	Check SR[1]: 1 = Attempted erase of locked block; erase aborted.

SR[1,3] must be cleared before the Write State Machine will allow further erase attempts.

Only the Clear Status Register command clears SR[1, 3, 4, 5].

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

Figure 34. Erase Suspend/Resume Flowchart

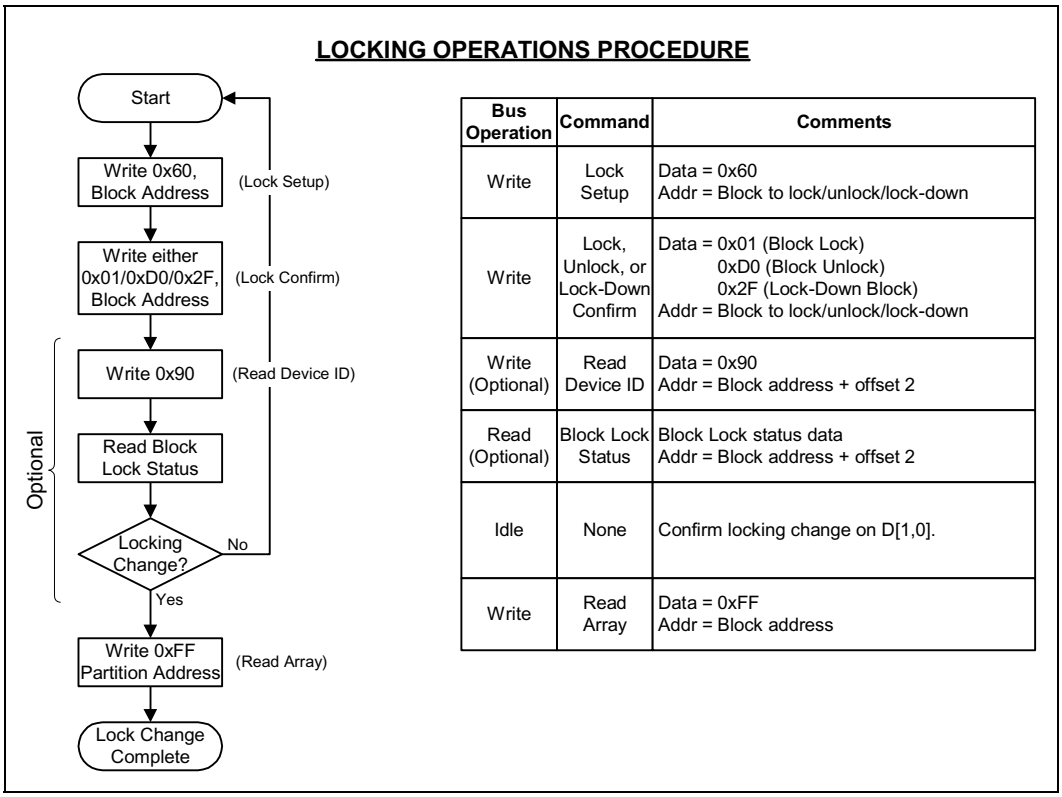


Bus Operation	Command	Comments
Write	Read Status	Data = 0x70 Addr = Any partition address
Write	Erase Suspend	Data = 0xB0 Addr = Same partition address as above
Read	None	Status Register data. Toggle CE# or OE# to update Status register; Addr = Same partition
Idle	None	Check SR[7]: 1 = WSM ready 0 = WSM busy
Idle	None	Check SR[6]: 1 = Erase suspended 0 = Erase completed
Write	Read Array or Program	Data = 0xFF or 0x40 Addr = Block to program or read
Read or Write	None	Read array or program data from/to block other than the one being erased
Write	Program Resume	Data = 0xD0 Addr = Any address

If the suspended partition was placed in Read Array mode or a Program Loop:

Write	Read Status Register	Return partition to Status mode: Data = 0x70 Addr = Same partition

Figure 35. Block Lock Operations Flowchart



Bus Operation	Command	Comments
Write	Lock Setup	Data = 0x60 Addr = Block to lock/unlock/lock-down
Write	Lock, Unlock, or Lock-Down Confirm	Data = 0x01 (Block Lock) 0xD0 (Block Unlock) 0x2F (Lock-Down Block) Addr = Block to lock/unlock/lock-down
Write (Optional)	Read Device ID	Data = 0x90 Addr = Block address + offset 2
Read (Optional)	Block Lock Status	Block Lock status data Addr = Block address + offset 2
Idle	None	Confirm locking change on D[1,0].
Write	Read Array	Data = 0xFF Addr = Block address

Figure 36. Protection Register Programming Flowchart

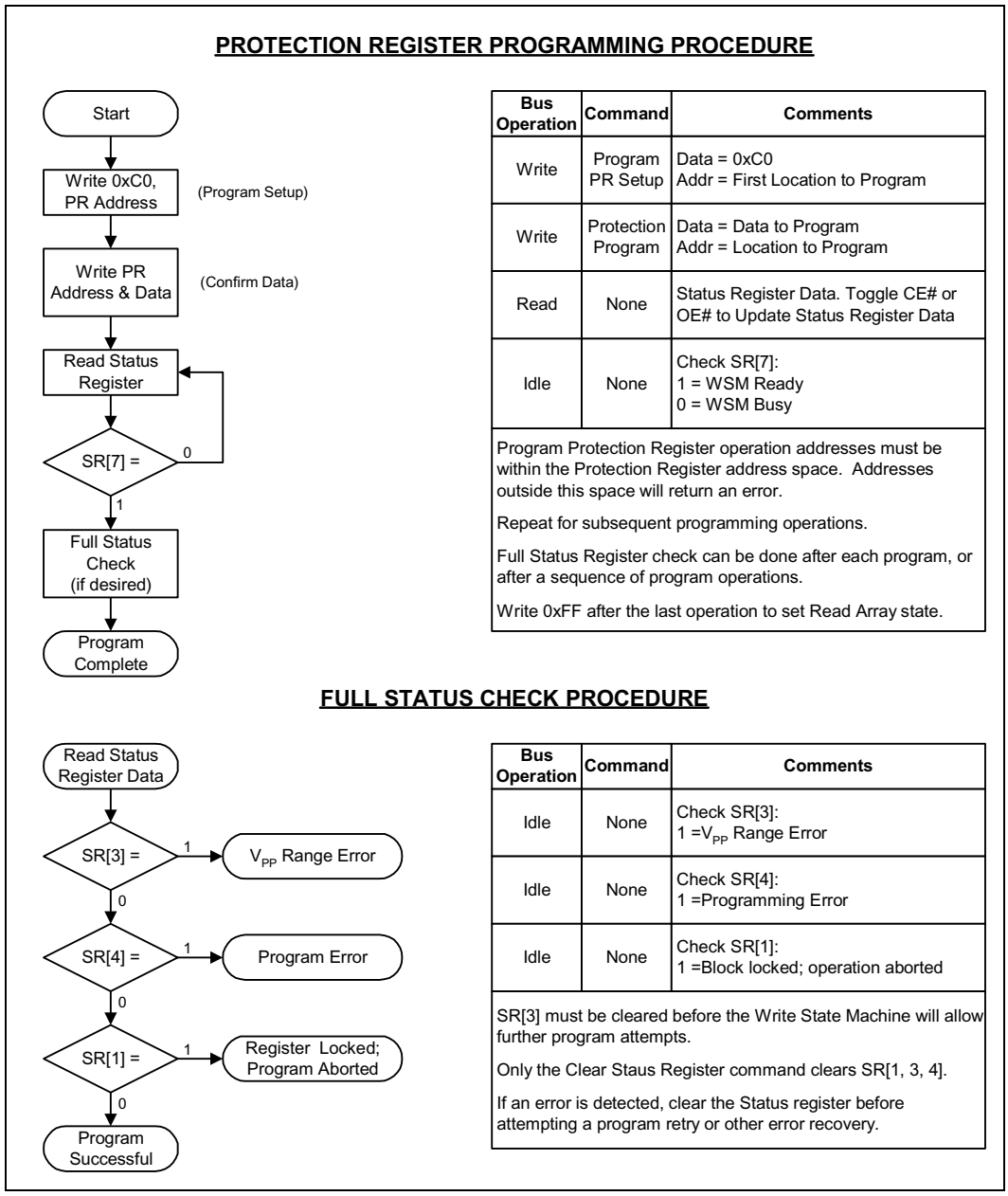
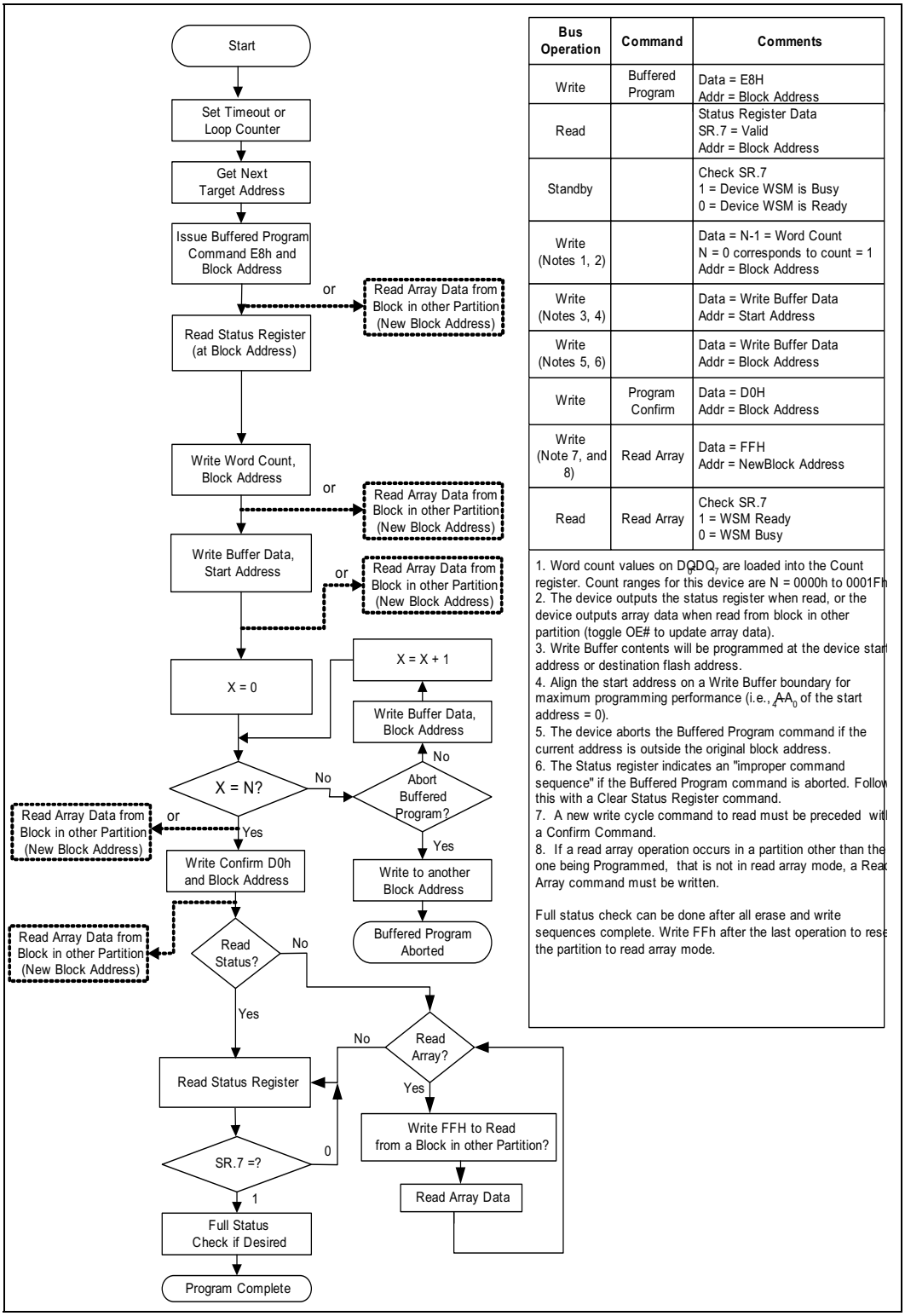


Figure 37. Read While Buffered Programming Flowchart



Bus Operation	Command	Comments
Write	Buffered Program	Data = E8H Addr = Block Address
Read		Status Register Data SR.7 = Valid Addr = Block Address
Standby		Check SR.7 1 = Device WSM is Busy 0 = Device WSM is Ready
Write (Notes 1, 2)		Data = N-1 = Word Count N = 0 corresponds to count = 1 Addr = Block Address
Write (Notes 3, 4)		Data = Write Buffer Data Addr = Start Address
Write (Notes 5, 6)		Data = Write Buffer Data Addr = Block Address
Write	Program Confirm	Data = D0H Addr = Block Address
Write (Note 7, and 8)	Read Array	Data = FFH Addr = NewBlock Address
Read	Read Array	Check SR.7 1 = WSM Ready 0 = WSM Busy

1. Word count values on DQ_0-DQ_7 are loaded into the Count register. Count ranges for this device are N = 0000h to 0001Fh
2. The device outputs the status register when read, or the device outputs array data when read from block in other partition (toggle OE# to update array data).
3. Write Buffer contents will be programmed at the device start address or destination flash address.
4. Align the start address on a Write Buffer boundary for maximum programming performance (i.e., A_4-A_0 of the start address = 0).
5. The device aborts the Buffered Program command if the current address is outside the original block address.
6. The Status register indicates an "improper command sequence" if the Buffered Program command is aborted. Follow this with a Clear Status Register command.
7. A new write cycle command to read must be preceded with a Confirm Command.
8. If a read array operation occurs in a partition other than the one being Programmed, that is not in read array mode, a Read Array command must be written.

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

Appendix C Common Flash Interface

Common Flash Interface (CFI) is part of an overall specification for multiple command-set and control-interface descriptions. This appendix describes the database structure containing the data returned by a read operation after issuing the CFI Query command (see Section 3.2, "Device Commands" on page 15). System software can parse this database structure to obtain information about the flash device, such as block size, density, bus width, and electrical specifications. The system software will then know which command set(s) to use to properly perform flash writes, block erases, reads and otherwise control the flash device.

C.1 Query Structure Output

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

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

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

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 17. Summary of Query Structure Output as a Function of Device and Mode

Device	Hex Offset	Hex Code	ASCII Value
Device Addresses	00010: 00011: 00012:	51 52 59	"Q" "R" "Y"

Table 18. Example of Query Structure Output of x16- Devices

Word Addressing:			Byte Addressing:		
Offset	Hex Code	Value	Offset	Hex Code	Value
A_X–A_0	D_{15}–D_0		A_X–A_0	D_7–D_0	
00010h	0051	"Q"	00010h	51	"Q"
00011h	0052	"R"	00011h	52	"R"
00012h	0059	"Y"	00012h	59	"Y"
00013h	P_ID_{LO}	PrVendor	00013h	P_ID_{LO}	PrVendor
00014h	P_ID_{HI}	ID #	00014h	P_ID_{LO}	ID #
00015h	P_{LO}	PrVendor	00015h	P_ID_{HI}	ID #
00016h	P_{HI}	TblAdr	00016h	...	...
00017h	A_ID_{LO}	AltVendor	00017h		
00018h	A_ID_{HI}	ID #	00018h		
...	...	...	...		

C.2 Query Structure Overview

The Query command causes the flash component to display the Common Flash Interface (CFI) Query structure or “database.” The structure sub-sections and address locations are summarized below.

Table 19. Query Structure

Offset	Sub-Section Name	Description(1)
00001-Fh	Reserved	Reserved for vendor-specific information
00010h	CFI query identification string	Command set ID and vendor data offset
0001Bh	System interface information	Device timing & voltage information
00027h	Device geometry definition	Flash device layout
P(3)	Primary Intel-specific Extended Query Table	Vendor-defined additional information specific

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 16K-word).
3. Offset 15 defines “P” which points to the Primary Intel-specific Extended Query Table.

C.3 CFI Query Identification String

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

Table 20. CFI Identification

Offset	Length	Description	Add.	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:	--0A --01	
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	

Table 21. System Interface Information

Offset	Length	Description	Add.	Hex Code	Value
1Bh	1	V_{CC} logic supply minimum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 BCD volts	1B:	--17	1.7V
1Ch	1	V_{CC} logic supply maximum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 BCD volts	1C:	--20	2.0V
1Dh	1	V_{PP} [programming] supply minimum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 HEX volts	1D:	--85	8.5V
1Eh	1	V_{PP} [programming] supply maximum program/erase voltage bits 0–3 BCD 100 mV bits 4–7 HEX volts	1E:	--95	9.5V
1Fh	1	“n” such that typical single word program time-out = 2^n µ-sec	1F:	--08	256µs
20h	1	“n” such that typical max. buffer write time-out = 2^n µ-sec	20:	--09	512µs
21h	1	“n” such that typical block erase time-out = 2^n m-sec	21:	--0A	1s
22h	1	“n” such that typical full chip erase time-out = 2^n m-sec	22:	--00	NA
23h	1	“n” such that maximum word program time-out = 2^n times typical	23:	--01	512µs
24h	1	“n” such that maximum buffer write time-out = 2^n times typical	24:	--01	1024µs
25h	1	“n” such that maximum block erase time-out = 2^n times typical	25:	--02	4s
26h	1	“n” such that maximum chip erase time-out = 2^n times typical	26:	--00	NA

C.4 Device Geometry Definition

Table 22. Device Geometry Definition

Offset	Length	Description	Code		
27h	1	"n" such that device size = 2^n in number of bytes	27:	See table below	
28h	2	Flash device interface code assignment: "n" such that n+1 specifies the bit field that represents the flash device width capabilities as described in the table: 7 6 5 4 3 2 1 0 — — — — x64 x32 x16 x8 15 14 13 12 11 10 9 8 — — — — — — — —	28: 29:	--01 --00	x16
2Ah	2	"n" such that maximum number of bytes in write buffer = 2^n	2A: 2B:	--06 --00	64
2Ch	1	Number of erase block regions (x) within device: 1. x = 0 means no erase blocking; the device erases in bulk 2. x specifies the number of device regions with one or more contiguous same-size erase blocks. 3. Symmetrically blocked partitions have one blocking region	2C:	See table below	
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 table below	
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 table below	
35h	4	Reserved for future erase block region information	35: 36: 37: 38:	See table below	

Address	64 Mbit		128 Mbit		256 Mbit	
	–B	–T	–B	–T	–B	–T
27:	--17	--17	--18	--18	--19	--19
28:	--01	--01	--01	--01	--01	--01
29:	--00	--00	--00	--00	--00	--00
2A:	--06	--06	--06	--06	--06	--06
2B:	--00	--00	--00	--00	--00	--00
2C:	--03	--03	--03	--03	--03	--03
2D:	--03	--17	--03	--30	--03	--6F
2E:	--00	--00	--00	--00	--00	--00
2F:	--80	--00	--80	--00	--80	--00
30:	--00	--04	--00	--04	--00	--04
31:	--0E	--0E	--1E	--1E	--1E	--1E
32:	--00	--00	--00	--00	--00	--00
33:	--00	--00	--00	--00	--00	--00
34:	--02	--02	--02	--02	--02	--02
35:	--17	--03	--30	--03	--6F	--03
36:	--00	--00	--00	--00	--00	--00
37:	--00	--80	--00	--80	--00	--80
38:	--04	--00	--04	--00	--04	--00

C.5 Intel-Specific Extended Query Table

Table 23. Primary Vendor-Specific Extended Query

Offset[1] P= 10Ah	Length	Description (Optional flash features and commands)	Add.	Hex Code	Value
(P+0)h	3	Primary extended query table	10A	--50	"P"
(P+1)h		Unique ASCII string "PRI"	10B:	--52	"R"
(P+2)h			10C:	--49	"I"
(P+3)h	1	Major version number, ASCII	10D:	--31	"1"
(P+4)h	1	Minor version number, ASCII	10E:	--33	"3"
(P+5)h	4	Optional feature and command support (1=yes, 0=no)	10F:	--E6	
(P+6)h		*bits 10–31 are reserved; undefined bits are "0." If bit 31 is*	110:	--03	
(P+7)h		*"1" then another 31 bit field of Optional features follows at*	111:	--00	
(P+8)h		*the end of the bit–30 field.*	112:	--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 Pagemode 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	113:	--01	
		Other supported operations are:			
		bits 1–7 reserved; undefined bits are "0"			
		bit 0 Program supported after erase suspend	bit 0	= 1	Yes
(P+A)h	2	Block status register mask	114:	--03	
(P+B)h		*bits 2–15 are Reserved; undefined bits are "0"*	115:	--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	116:	--18	1.8V
		bits 0–3 BCD value in 100 mV			
		bits 4–7 BCD value in volts			
(P+D)h	1	V_{PP} optimum program/erase supply voltage	117:	--90	9.0V
		bits 0–3 BCD value in 100 mV			
		bits 4–7 HEX value in volts			

Table 24. Protection Register Information

Offset[1] P= 10Ah	Length	Description (Optional flash features and commands)	Add.	Hex Code	Value
(P+E)h	1	Number of Protection register fields in JEDEC ID space. "00h," indicates that 256 protection fields are available	118:	--02	2
(P+F)h	4	Protection Field 1: Protection Description	119:	--80	80h
(P+10)h		This field describes user-available One Time Programmable	11A:	--00	00h
(P+11)h		(OTP) Protection register bytes. Some are pre-programmed	11B:	--03	8 byte
(P+12)h		with device-unique serial numbers. Others are user	11C:	--03	8 byte
		programmable. Bits 0–15 point to the Protection register Lock byte, the section's first byte. The following bytes are factory pre-programmed and user-programmable. bits 0–7 = Lock/bytes Jedec-plane physical low address bits 8–15 = Lock/bytes Jedec-plane physical high address bits 16–23 = "n" such that 2^n = factory pre-programmed bytes bits 24–31 = "n" such that 2^n = user programmable bytes			
(P+13)h	10	Protection Field 2: Protection Description	11D:	--89	89h
(P+14)h		Bits 0–31 point to the Protection register physical Lock-word	11E:	--00	00h
(P+15)h		address in the Jedec-plane.	11F:	--00	00h
(P+16)h		Following bytes are factory or user-programmable.	120:	--00	00h
(P+17)h		bits 32–39 = "n" ∴n = factory pgm'd groups (low byte)	121:	--00	0
(P+18)h		bits 40–47 = "n" ∴n = factory pgm'd groups (high byte)	122:	--00	0
(P+19)h		bits 48–55 = "n" \ 2n = factory programmable bytes/group	123:	--00	0
(P+1A)h		bits 56–63 = "n" ∴n = user pgm'd groups (low byte)	124:	--04	16
(P+1B)h		bits 64–71 = "n" ∴n = user pgm'd groups (high byte)	125:	--00	0
(P+1C)h		bits 72–79 = "n" ∴2^n = user programmable bytes/group	126:	--04	16

Table 25. Burst Read Information

Offset[1] P= 10Ah	Length	Description (Optional flash features and commands)	Add.	Hex Code	Value
(P+1D)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.	127:	--00	0 byte
(P+1E)h	1	Number of synchronous mode read configuration fields that follow. 00h indicates no burst capability.	128:	--04	4
(P+1F)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.	129:	--01	4
(P+20)h	1	Synchronous mode read capability configuration 2	12A:	--02	8
(P+21)h	1	Synchronous mode read capability configuration 3	12B:	--03	16
(P+22)h	1	Synchronous mode read capability configuration 4	12C:	--07	Cont

Table 26. Partition and Erase-block Region Information

Offset[1] P= 10Ah		Description	See table below		
				Address	
Bottom	Top	(Optional flash features and commands)	Len	Bot	Top
(P+23)h	(P+23)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	12D:	12D:

Partition Region 1 Information

Offset[1] P= 10Ah		Description	See table below		
				Address	
Bottom	Top	(Optional flash features and commands)	Len	Bot	Top
(P+24)h (P+25)h	(P+24)h (P+25)h	Number of identical partitions within the partition region	2	12E: 12F:	12E: 12F:
(P+26)h	(P+26)h	Number of program or erase operations allowed in a partition bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	130:	130:
(P+27)h	(P+27)h	Simultaneous program or erase operations allowed in other partitions while a partition in this region is in Program mode bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	131:	131:
(P+28)h	(P+28)h	Simultaneous program or erase operations allowed in other partitions while a partition in this region is in Erase mode bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	132:	132:
(P+29)h	(P+29)h	Types of erase block regions in this Partition Region. x = 0 = no erase blocking; the Partition Region erases in bulk x = number of erase block regions w/ contiguous same-size erase blocks. Symmetrically blocked partitions have one blocking region. Partition size = (Type 1 blocks)x(Type 1 block sizes) + (Type 2 blocks)x(Type 2 block sizes) +…+ (Type n blocks)x(Type n block sizes)	1	133:	133:
(P+2A)h (P+2B)h (P+2C)h (P+2D)h	(P+2A)h (P+2B)h (P+2C)h (P+2D)h	Partition Region 1 Erase Block Type 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	134: 135: 136: 137:	134: 135: 136: 137:
(P+2E)h (P+2F)h	(P+2E)h (P+2F)h	Partition 1 (Erase Block Type 1) Minimum block erase cycles x 1000	2	138: 139:	138: 139:
(P+30)h	(P+30)h	Partition 1 (erase block Type 1) bits per cell; internal ECC bits 0–3 = bits per cell in erase region bit 4 = reserved for “internal ECC used” (1=yes, 0=no) bits 5–7 = reserve for future use	1	13A:	13A:
(P+31)h	(P+31)h	Partition 1 (erase block Type 1) page mode and synchronous mode capabilities defined in Table 10. 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	13B:	13B:
(P+32)h (P+33)h (P+34)h (P+35)h		Partition Region 1 Erase Block Type 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	13C: 13D: 13E: 13F:	
(P+36)h (P+37)h		Partition 1 (Erase block Type 2) Minimum block erase cycles x 1000	2	140: 141:	
(P+38)h		Partition 1 (Erase block Type 2) 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 = reserve for future use	1	142:	
(P+39)h		Partition 1 (Erase block Type 2) pagemode and synchronous mode capabilities defined in Table 10 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	143:	

Partition Region 2 Information

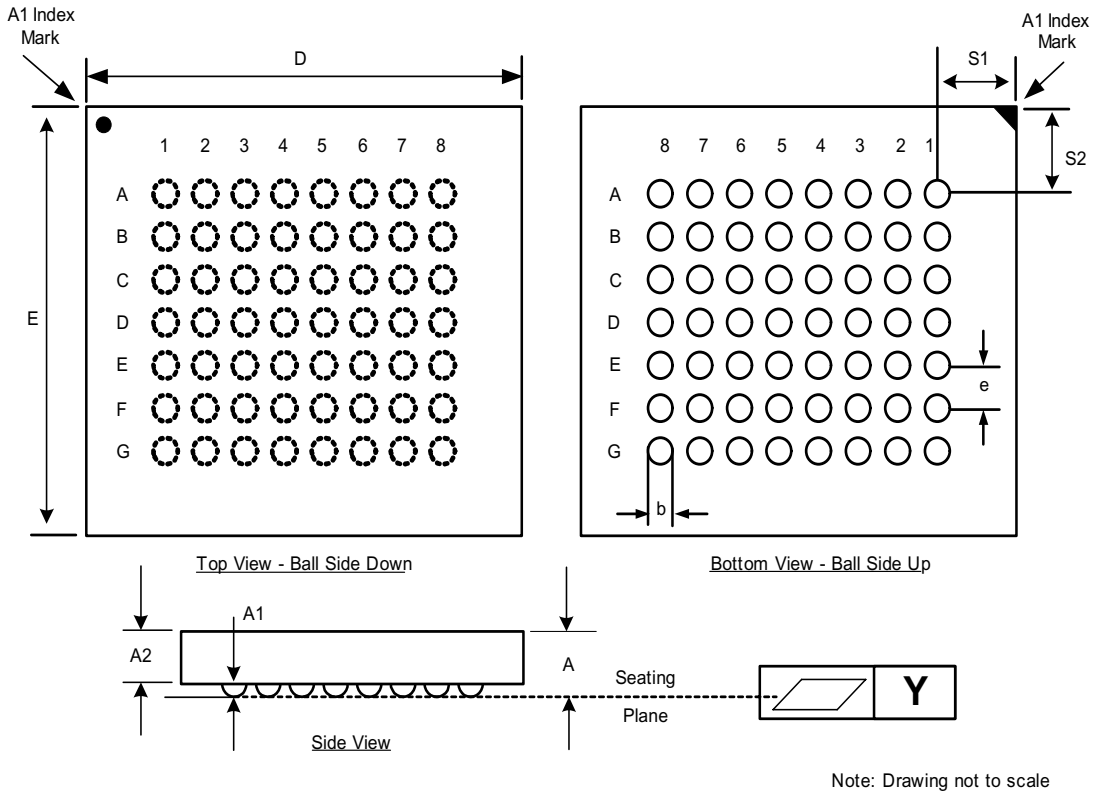
Offset[1] P= 10Ah		Description	See table below		
				Address	
Bottom	Top	(Optional flash features and commands)	Len	Bot	Top
(P+3A)h (P+3B)h	(P+32)h (P+33)h	Number of identical partitions within the partition region	2	144: 145:	13C: 13D:
(P+3C)h	(P+34)h	Number of program or erase operations allowed in a partition bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	146:	13E:
(P+3D)h	(P+35)h	Simultaneous program or erase operations allowed in other partitions while a partition in this region is in Program mode bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	147:	13F:
(P+3E)h	(P+36)h	Simultaneous program or erase operations allowed in other partitions while a partition in this region is in Erase mode bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	148:	140:
(P+3F)h	(P+37)h	Types of erase block regions in this Partition Region. x = 0 = no erase blocking; the Partition Region erases in bulk x = number of erase block regions w/ contiguous same-size erase blocks. Symmetrically blocked partitions have one blocking region. Partition size = (Type 1 blocks)x(Type 1 block sizes) + (Type 2 blocks)x(Type 2 block sizes) +…+ (Type n blocks)x(Type n block sizes)	1	149:	141:
(P+40)h (P+41)h (P+42)h (P+43)h	(P+38)h (P+39)h (P+3A)h (P+3B)h	Partition Region 2 Erase Block Type 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	14A: 14B: 14C: 14D:	142: 143: 144: 145:
(P+44)h (P+45)h	(P+3C)h (P+3D)h	Partition 2 (Erase block Type 1) Minimum block erase cycles x 1000	2	14E: 14F:	146: 147:
(P+46)h	(P+3E)h	Partition 2 (Erase block Type 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 = reserve for future use	1	150:	148:
(P+47)h	(P+3F)h	Partition 2 (erase block Type 1) pagemode and synchronous mode capabilities as defined in Table 10. 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	151:	149:
	(P+40)h (P+41)h (P+42)h (P+43)h	Partition Region 2 Erase Block Type 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	4		14A: 14B: 14C: 14D:
	(P+44)h (P+45)h	Partition 2 (Erase block Type 2) Minimum block erase cycles x 1000	2		14E: 14F:
	(P+46)h	Partition 2 (Erase block Type 2) 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 = reserve for future use	1		150:
	(P+47)h	Partition 2 (erase block Type 2) pagemode and synchronous mode capabilities as defined in Table 10. 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		151:

Partition and Erase-block Region Information

Address	64 Mbit		128 Mbit		256 Mbit	
	–B	–T	–B	–T	–B	–T
12D:	--03	--03	--03	--03	--03	--03
12E:	--01	--03	--01	--06	--01	--0E
12F:	--00	--00	--00	--00	--00	--00
130:	--11	--11	--11	--11	--11	--11
131:	--00	--00	--00	--00	--00	--00
132:	--00	--00	--00	--00	--00	--00
133:	--02	--01	--02	--01	--02	--01
134:	--03	--07	--03	--07	--03	--07
135:	--00	--00	--00	--00	--00	--00
136:	--80	--00	--80	--00	--80	--00
137:	--00	--04	--00	--04	--00	--04
138:	--64	--64	--64	--64	--64	--64
139:	--00	--00	--00	--00	--00	--00
13A:	--01	--01	--01	--01	--01	--01
13B:	--03	--03	--03	--03	--03	--03
13C:	--06	--01	--06	--03	--06	--03
13D:	--00	--00	--00	--00	--00	--00
13E:	--00	--11	--00	--11	--00	--11
13F:	--02	--00	--02	--00	--02	--00
140:	--64	--00	--64	--00	--64	--00
141:	--00	--01	--00	--01	--00	--01
142:	--01	--07	--01	--07	--01	--07
143:	--03	--00	--03	--00	--03	--00
144:	--01	--00	--03	--00	--03	--00
145:	--00	--02	--00	--02	--00	--02
146:	--11	--64	--11	--64	--11	--64
147:	--00	--00	--00	--00	--00	--00
148:	--00	--01	--00	--01	--00	--01
149:	--01	--03	--01	--03	--01	--03
14A:	--07	--01	--07	--01	--07	--01
14B:	--00	--00	--00	--00	--00	--00
14C:	--00	--11	--00	--11	--00	--11
14D:	--02	--00	--02	--00	--02	--00
14E:	--64	--00	--64	--00	--64	--00
14F:	--00	--02	--00	--02	--00	--02
150:	--01	--06	--01	--06	--01	--06
151:	--03	--00	--03	--00	--03	--00
152:	--03	--00	--06	--00	--0E	--00
153:	--00	--02	--00	--02	--00	--02
154:	--11	--64	--11	--64	--11	--64
155:	--00	--00	--00	--00	--00	--00
156:	--00	--01	--00	--01	--00	--01
157:	--01	--03	--01	--03	--01	--03
158:	--07	--03	--07	--03	--07	--03
159:	--00	--00	--00	--00	--00	--00
15A:	--00	--80	--00	--80	--00	--80
15B:	--04	--00	--04	--00	--04	--00
15C:	--64	--64	--64	--64	--64	--64
15D:	--00	--00	--00	--00	--00	--00
15E:	--01	--01	--01	--01	--01	--01
15F:	--03	--03	--03	--03	--03	--03

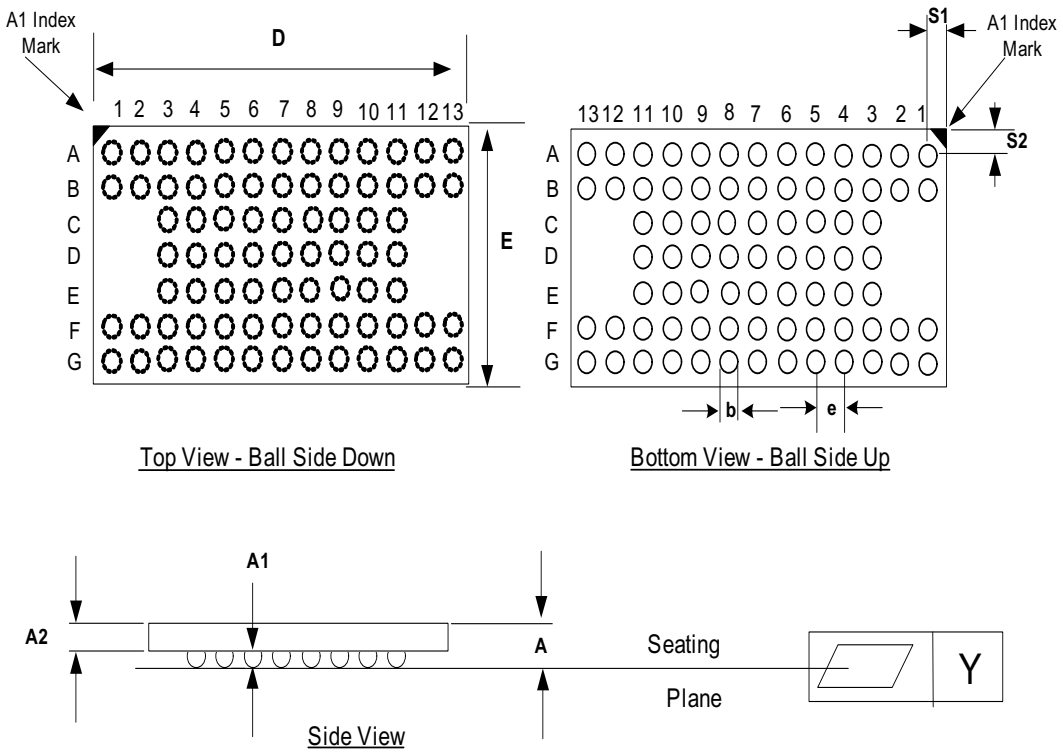
Appendix D Mechanical Information

Figure 38. 64- and 128-Mbit; 56-Ball VF BGA Package Drawing and Dimensions



		Millimeters				Inches		
Dimensions	**Symbol**	**Min**	**Nom**	**Max**	**Notes**	**Min**	**Nom**	**Max**
Package Height	A			1.000				0.0394
Ball Height	A1	0.150				0.0059		
Package Body Thickness	A2		0.665				0.0262	
Ball (Lead) Width	b	0.325	0.375	0.425		0.0128	0.0148	0.0167
Package Body Length (64Mb, 128Mb)	D	7.600	7.700	7.800		0.2992	0.3031	0.3071
Package Body Width (64Mb, 128Mb)	E	8.900	9.000	9.100		0.3504	0.3543	0.3583
Pitch	e		0.750				0.0295	
Ball (Lead) Count	N		56				56	
Seating Plane Coplanarity	Y			0.100				0.0039
Corner to Ball A1 Distance Along D	S1	1.125	1.225	1.325		0.0443	0.0482	0.0522
Corner to Ball A1 Distance Along E	S2	2.150	2.250	2.350		0.0846	0.0886	0.0925

Figure 39. 256-Mbit; 79-Ball VF BGA Package Drawing and Dimensions



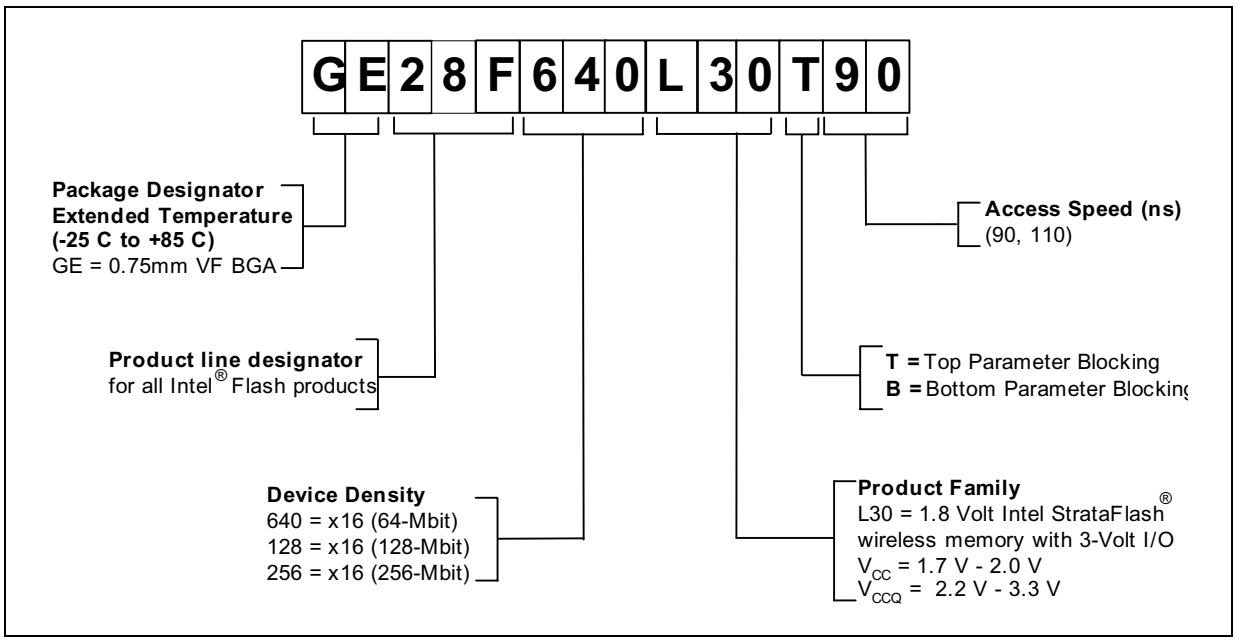
Drawing not to scale

Dimensions Table

		Millimeters				Inches		
Dimensions	**Symbol**	**Min**	**Nom**	**Max**	**Notes**	**Min**	**Nom**	**Max**
Package Height	A			1.000				0.0394
Ball Height	A1	0.150				0.0059		
Package Body Thickness	A2		0.665				0.0262	
Ball (Lead) Width	b	0.325	0.375	0.425		0.0128	0.0148	0.0167
Package Body Length (256Mb)	D	10.900	11.000	11.100		0.4291	0.4331	0.4370
Package Body Width (256Mb)	E	8.900	9.000	9.100		0.3504	0.3543	0.3583
Pitch	e		0.750				0.0295	
Ball (Lead) Count	N		79				79	
Seating Plane Coplanarity	Y			0.100				0.0039
Corner to Ball A1 Distance Along D	S1	0.900	1.000	1.100		0.0354	0.0394	0.0433
Corner to Ball A1 Distance Along E	S2	2.150	2.250	2.350		0.0846	0.0886	0.0925

Appendix E Additional Information

Order/Document Number	Document/Tool
251902	1.8 Volt Intel StrataFlash® Wireless Memory Datasheet
290701	1.8 Volt Intel® Wireless Flash Memory Datasheet
290702	1.8 Volt Intel® Wireless Flash Memory with 3-Volt I/O Datasheet
290737	3 Volt Synchronous Intel StrataFlash® Memory Datasheet
251908	Migration Guide for 1.8 Volt Intel® Wireless Flash Memory (W18/W30) to 1.8 Volt Intel StrataFlash® Wireless Memory (L18/L30), Application Note 753
251909	Migration Guide for 3 Volt Synchronous Intel StrataFlash® Memory (K3/K18) to 1.8 Volt Intel StrataFlash® Wireless Memory (L18/L30), Application Note 754
298161	Intel® Flash Memory Chip Scale Package User's Guide
297833	Intel® Flash Data Integrator (FDI) User's Guide
298136	Intel® Persistent Storage Manager User Guide

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 for technical documentation and tools.
3. For the most current information on Intel StrataFlash memory, visit our website at http://developer.intel.com/design/flash/isf.
4. This document is not available through Intel Literature. Please visit our website at http://developer.intel.com/design/flcomp/packdata/298049.htm.

Appendix F Ordering Information