



# Intel StrataFlash<sup>®</sup> Wireless Memory (L30)

28F640L30, 28F128L30, 28F256L30

Datasheet

## Product Features

- **High performance Read-While-Write/Erase**
  - 85 ns initial access
  - 52 MHz 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): 5  $\mu$ s/byte (Typ)
  - 1.8 V low-power buffered and non-buffered programming at 7  $\mu$ s/byte (Typ)
- **Architecture**
  - Asymmetrically-blocked architecture
  - Multiple 8-Mbit partitions: 64 Mb and 128 Mb devices
  - Multiple 16-Mbit partitions: 256-Mb devices
  - Four 16-Kword 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  $\mu$ A (Typ)
  - 4-Word synchronous read current: 17 mA (Typ) at 54 MHz
  - Automatic Power Savings (APS) mode
- **Software**
  - 20  $\mu$ s (Typ) program suspend
  - 20  $\mu$ s (Typ) erase suspend
  - Intel<sup>®</sup> 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:  $-25^{\circ}C$  to  $+85^{\circ}C$
  - Minimum 100,000 erase cycles per block
  - ETOX<sup>™</sup> VIII process technology (0.13  $\mu$ m)
- **Density and Packaging**
  - 64-, 128- and 256-Mbit density in VF BGA packages
  - 128/0, and 256/0 Density in Stacked-CSP
  - 16-bit wide data bus

The Intel StrataFlash<sup>®</sup> wireless memory (L30) product is the latest generation of Intel StrataFlash<sup>®</sup> memory devices featuring flexible, multiple-partition, dual operation. It provides high performance synchronous-burst read mode and asynchronous 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.

The L30 device is manufactured using Intel<sup>®</sup> 0.13  $\mu$ m ETOX<sup>™</sup> 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. Verify with your local Intel sales office that you have the latest datasheet before finalizing a design.

---



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 Intel StrataFlash® Wireless Memory (L30) 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 © 2004, Intel Corporation

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

# Contents

---

<b>1.0</b>	<b>Introduction</b> .....	7
	1.1 Nomenclature .....	7
	1.2 Acronyms .....	7
	1.3 Conventions .....	8
<b>2.0</b>	<b>Functional Overview</b> .....	9
<b>3.0</b>	<b>Package Information</b> .....	10
	3.1 VF BGA Packages .....	10
	3.2 SCSP Packages .....	12
	3.3 Intel® UT SCSP Packages .....	13
<b>4.0</b>	<b>Ballout and Signal Descriptions</b> .....	14
	4.1 Signal Ballout .....	14
	4.1.1 VF BGA Package Ballout .....	14
	4.2 SCSP Package Ballout .....	16
	4.3 Signal Descriptions .....	17
	4.3.1 VF BGA Package Signal Descriptions .....	17
	4.3.2 128/0 and 256/0 SCSP Package Signal Descriptions .....	19
	4.4 Memory Map .....	21
<b>5.0</b>	<b>Maximum Ratings and Operating Conditions</b> .....	23
	5.1 Absolute Maximum Ratings .....	23
	5.2 Operating Conditions .....	23
<b>6.0</b>	<b>Electrical Specifications</b> .....	24
	6.1 DC Current Characteristics .....	24
	6.2 DC Voltage Characteristics .....	25
<b>7.0</b>	<b>AC Characteristics</b> .....	26
	7.1 AC Test Conditions .....	26
	7.2 Capacitance .....	27
	7.3 AC Read Specifications for 64-Mbit and 128-Mbit Densities .....	28
	7.4 AC Read Specifications for 256-Mb Density .....	29
	7.5 AC Write Specifications .....	34
	7.6 Program and Erase Characteristics .....	37
<b>8.0</b>	<b>Power and Reset Specifications</b> .....	38
	8.1 Power Up and Down .....	38
	8.2 Reset Specifications .....	38
	8.3 Power Supply Decoupling .....	39
	8.4 Automatic Power Saving (APS) .....	39
<b>9.0</b>	<b>Device Operations</b> .....	40
	9.1 Bus Operations .....	40
	9.1.1 Reads .....	40
	9.1.2 Writes .....	40
	9.1.3 Output Disable .....	40

	9.1.4	Standby .....	41
	9.1.5	Reset .....	41
	9.2	Device Commands .....	41
	9.3	Command Definitions .....	43
<b>10.0</b>		<b>Read Operations .....</b>	<b>45</b>
	10.1	Asynchronous Page-Mode Read .....	45
	10.2	Synchronous Burst-Mode Read .....	45
	10.2.1	Burst Suspend .....	46
	10.3	Read Configuration Register (RCR) .....	46
	10.3.1	Read Mode .....	47
	10.3.2	Latency Count .....	48
	10.3.3	WAIT Polarity .....	50
	10.3.3.1	WAIT Signal Function .....	50
	10.3.4	Data Hold .....	51
	10.3.5	WAIT Delay .....	52
	10.3.6	Burst Sequence .....	52
	10.3.7	Clock Edge .....	53
	10.3.8	Burst Wrap .....	53
	10.3.9	Burst Length .....	53
<b>11.0</b>		<b>Programming Operations .....</b>	<b>54</b>
	11.1	Word Programming .....	54
	11.1.1	Factory Word Programming .....	55
	11.2	Buffered Programming .....	55
	11.3	Buffered Enhanced Factory Programming .....	56
	11.3.1	Buffered EFP Requirements and Considerations .....	56
	11.3.2	Buffered EFP Setup Phase .....	57
	11.3.3	Buffered EFP Program/Verify Phase .....	57
	11.3.4	Buffered EFP Exit Phase .....	58
	11.4	Program Suspend .....	58
	11.5	Program Resume .....	58
	11.6	Program Protection .....	59
<b>12.0</b>		<b>Erase Operations .....</b>	<b>60</b>
	12.1	Block Erase .....	60
	12.2	Erase Suspend .....	60
	12.3	Erase Resume .....	61
	12.4	Erase Protection .....	61
<b>13.0</b>		<b>Security Modes .....</b>	<b>62</b>
	13.1	Block Locking .....	62
	13.1.1	Lock Block .....	62
	13.1.2	Unlock Block .....	62
	13.1.3	Lock-Down Block .....	62
	13.1.4	Block Lock Status .....	63
	13.1.5	Block Locking During Suspend .....	63
	13.2	Protection Registers .....	64
	13.2.1	Reading the Protection Registers .....	65
	13.2.2	Programming the Protection Registers .....	66
	13.2.3	Locking the Protection Registers .....	66

<b>14.0</b>	<b>Dual-Operation Considerations</b> .....	67
14.1	Memory Partitioning .....	67
14.2	Read-While-Write Command Sequences .....	67
14.2.1	Simultaneous Operation Details.....	68
14.2.2	Synchronous and Asynchronous Read-While-Write Characteristics and Waveforms .....	68
14.2.2.1	Write operation to asynchronous read transition .....	68
14.2.2.2	Write to synchronous read operation transition .....	69
14.2.2.3	Write Operation with Clock Active .....	69
14.2.3	Read Operation During Buffered Programming Flowchart.....	69
14.3	Simultaneous Operation Restrictions .....	70
<b>15.0</b>	<b>Special Read States</b> .....	71
15.1	Read Status Register .....	71
15.1.1	Clear Status Register .....	72
15.2	Read Device Identifier .....	72
15.3	CFI Query .....	73
<b>Appendix A</b>	<b>Write State Machine (WSM)</b> .....	74
<b>Appendix B</b>	<b>Flowcharts</b> .....	81
<b>Appendix C</b>	<b>Common Flash Interface</b> .....	89
<b>Appendix D</b>	<b>Additional Information</b> .....	99
<b>Appendix E</b>	<b>Ordering Information for VF BGA Package</b> .....	100
<b>Appendix F</b>	<b>Ordering Information for SCSP Package</b> .....	101

## Revision History

---

Revision Date	Revision	Description
10/14/02	-001	Initial Release
02/08/03	-002	Revised 256Mb Partition Size Revised 256Mb Memory Map Changed WAIT function to de-assert during Asynchronous Operations (Asynchronous Reads and all Writes) Changed WAIT function to active during Synchronous Non-Array Read Updated all Waveforms to reflect new WAIT function Revised Section 8.2.2 Added Synchronous Read to Write transition Section Added new AC specs: R15, R16, R17, R111, R311, R312, W21, and W22 Various text edits
04/11/03	-003	Improved Bin 1 to 85ns from 90ns Improved Frequency to 52MHz from 50MHz Added SCSP for 128/0 and 256/0 Ball-out and Mechanical Drawing
08/04/03	-004	Changed I <sub>CCS</sub> and I <sub>CCR</sub> values Added 256Mb AC Speed Changed Program and Erase Spec Combined the Buffered Programming Flow Chart and Read While Buffered programming Flow Chart Revised Read While Buffered Programming Flow Chart Revised Appendix A Write State Machine Revised CFI Table 21 CFI Identification Various text edits
12/12/03	-005	New EMTS format.

## 1.0 Introduction

This document provides information about the Intel StrataFlash® wireless memory (L30) device and 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: 16-Kword, and 64-Kword.

**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<sub>CC</sub>:** 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 Functional Overview

This section provides an overview of the features and capabilities of the L30 device.

The 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,  $V_{CC}$  and  $V_{PP}$  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 deselected 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.

### 3.0 Package Information

#### 3.1 VF BGA Packages

Figure 1. 64- and 128-Mbit, 56-Ball VF BGA Package Drawing and Dimensions

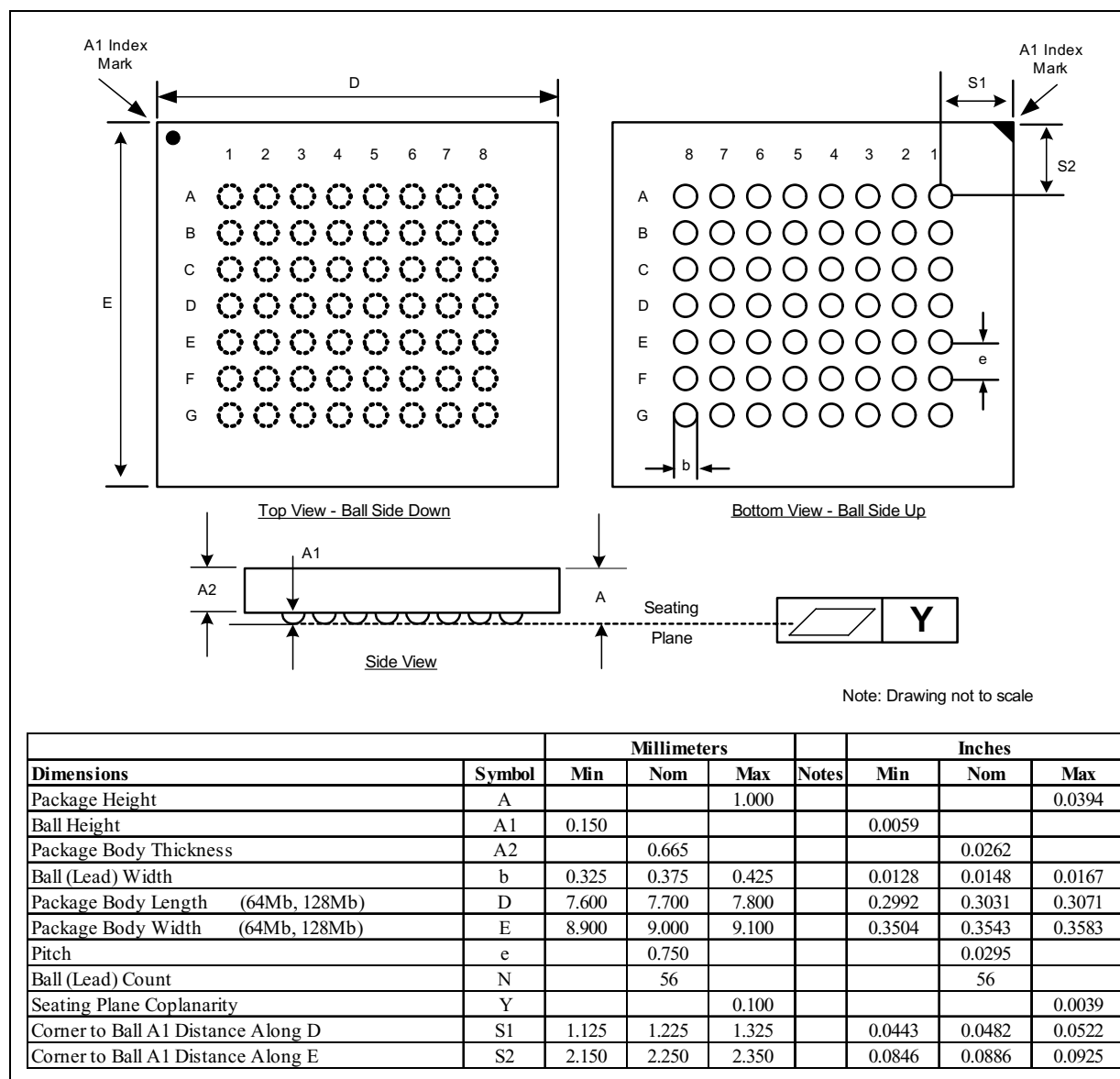
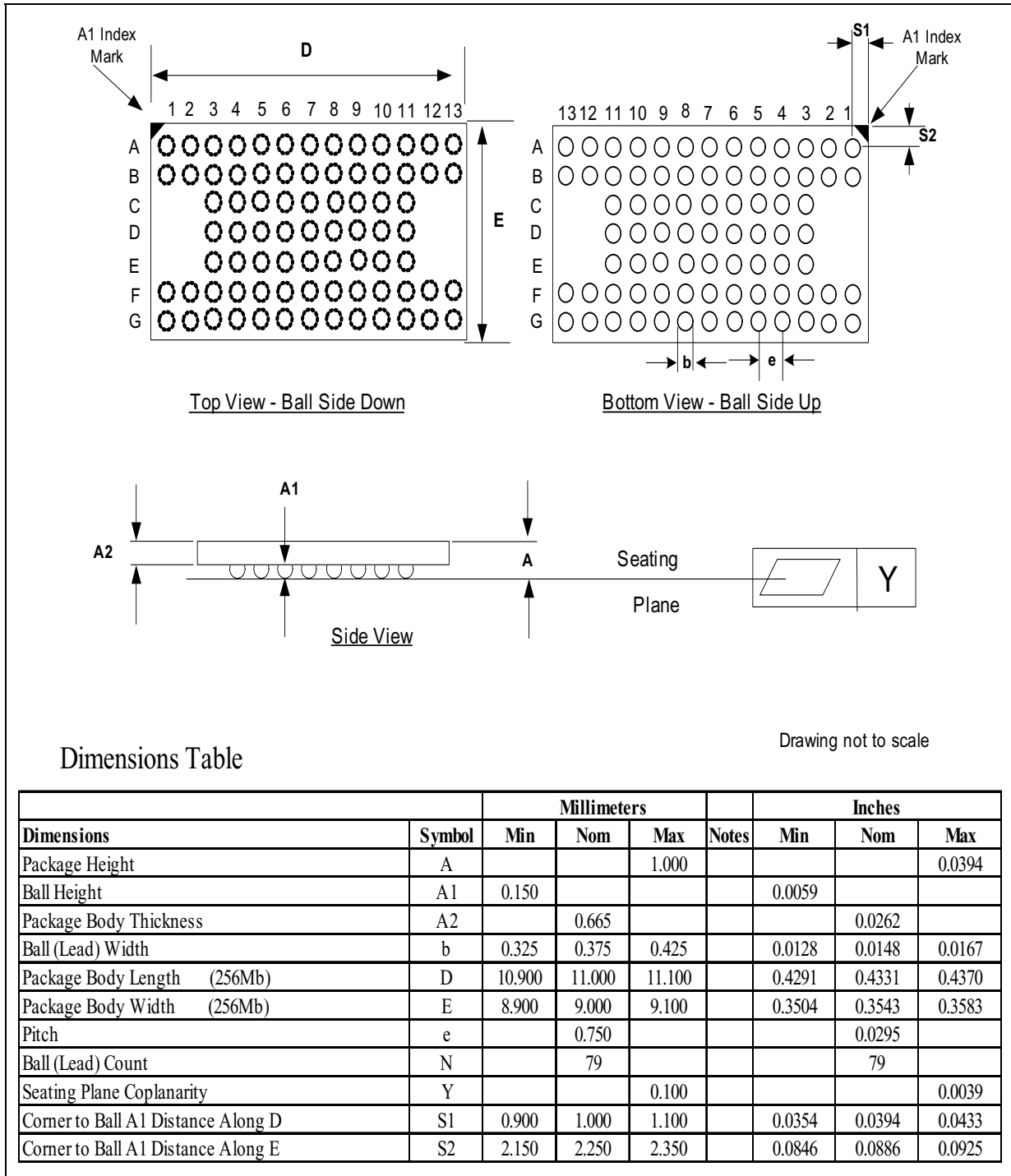
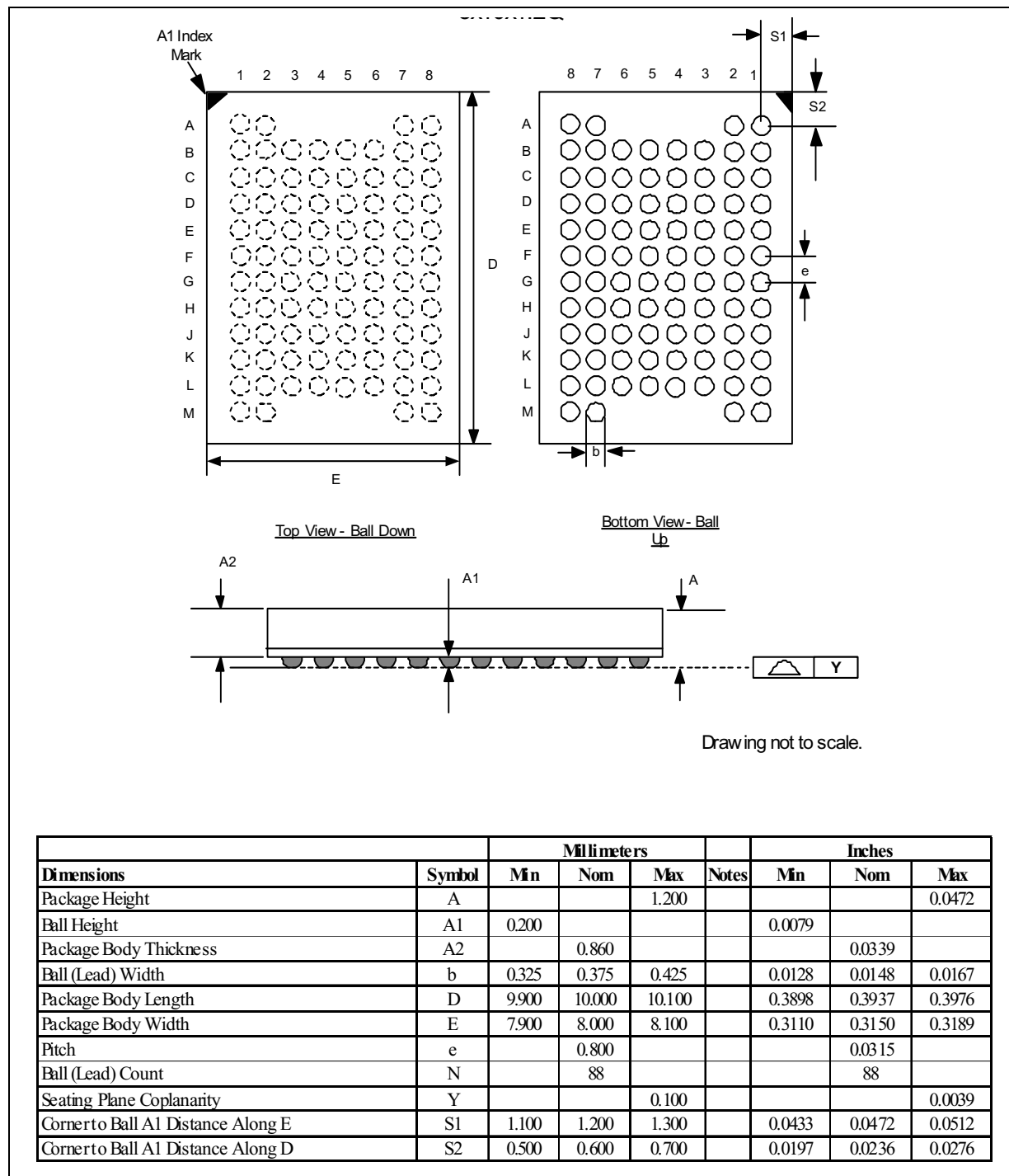


Figure 2. 256-Mbit, 79-Ball VF BGA Package Drawing and Dimensions



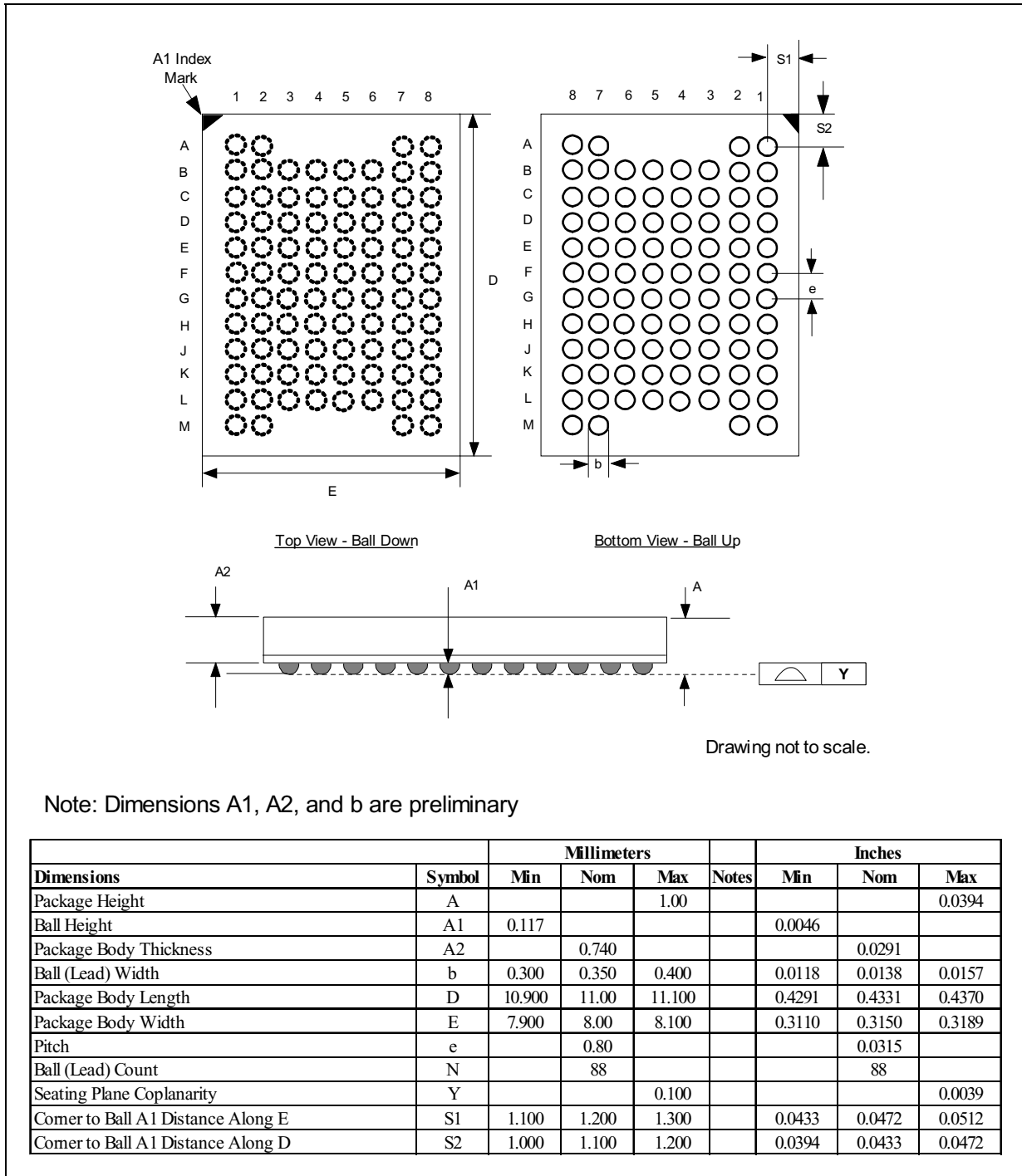
### 3.2 SCSP Packages

Figure 3. 128-Mbit, 88-ball (80-active ball) SCSP Drawing and Dimensions



### 3.3 Intel® UT SCSP Packages

Figure 4. 256-Mbit, 88-ball (80-active ball) Intel® UT-SCSP Drawing and Dimensions



## 4.0 Ballout and Signal Descriptions

### 4.1 Signal Ballout

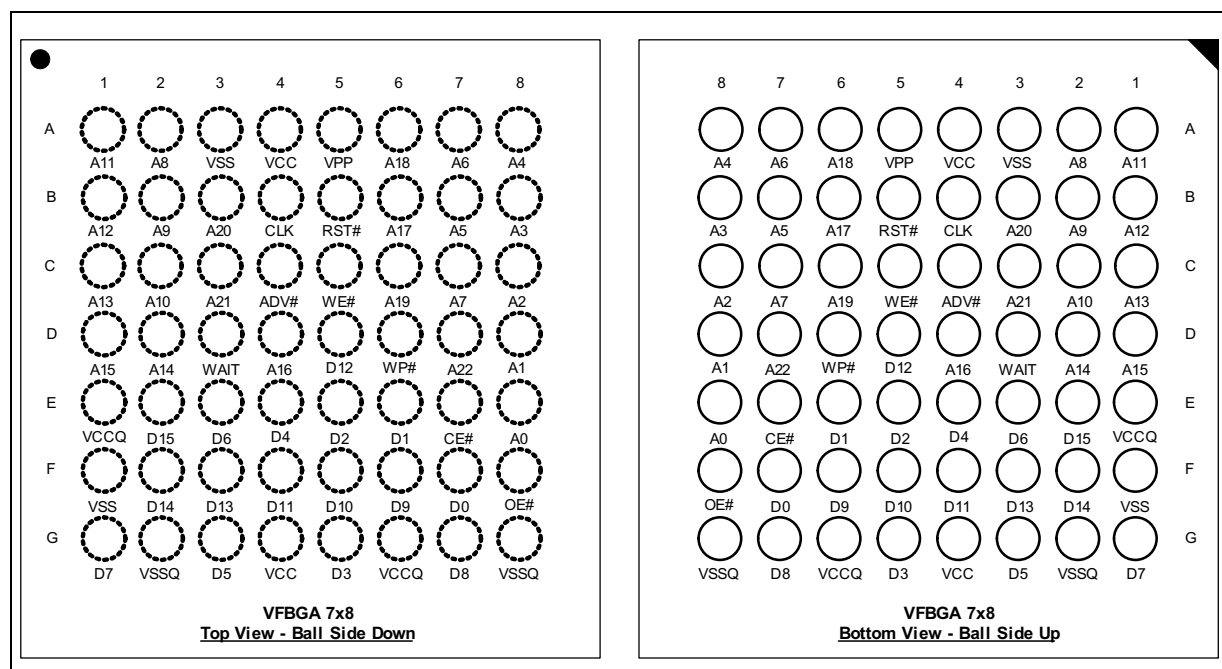
This section includes signal ballouts for the following packages:

- VF BGA Package Ballout
- SCSP Package Ballout

#### 4.1.1 VF BGA Package Ballout

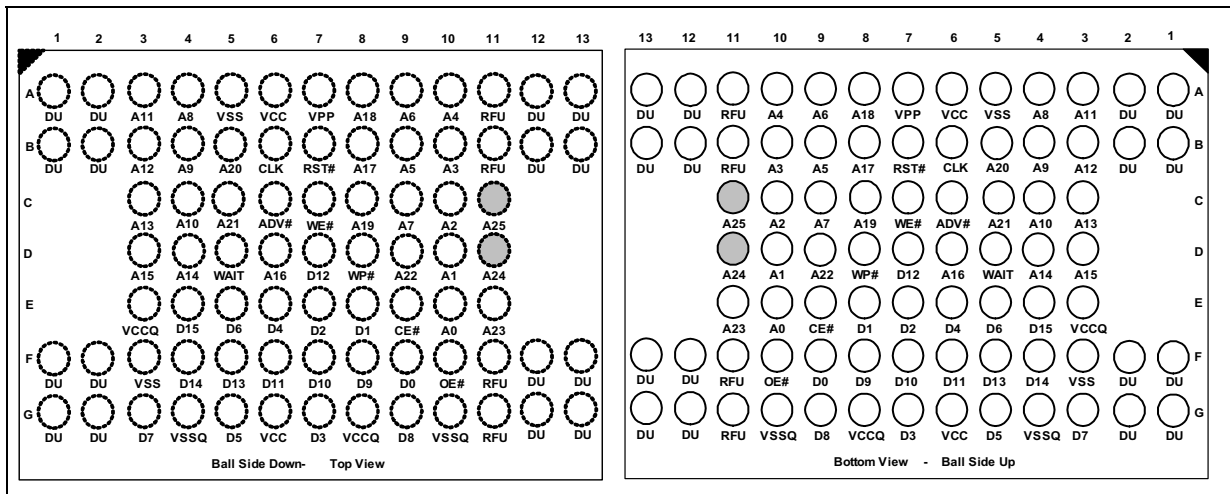
The L30 flash memory device is available in a VF BGA package with 0.75 mm ball-pitch. [Figure 5](#) shows the ballout for the 64-Mbit and 128-Mbit devices in the 56-ball VF BGA package with a 7 x 8 active-ball matrix. [Figure 6](#) shows the device ballout for the 256-Mbit device in the 63-ball VF BGA package with a 7 x 9 active-ball matrix. Both package densities are ideal for space-constrained board applications

**Figure 5. 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 6. 7x9 Active-Ball Matrix for 256-Mbit Density in VF BGA Package

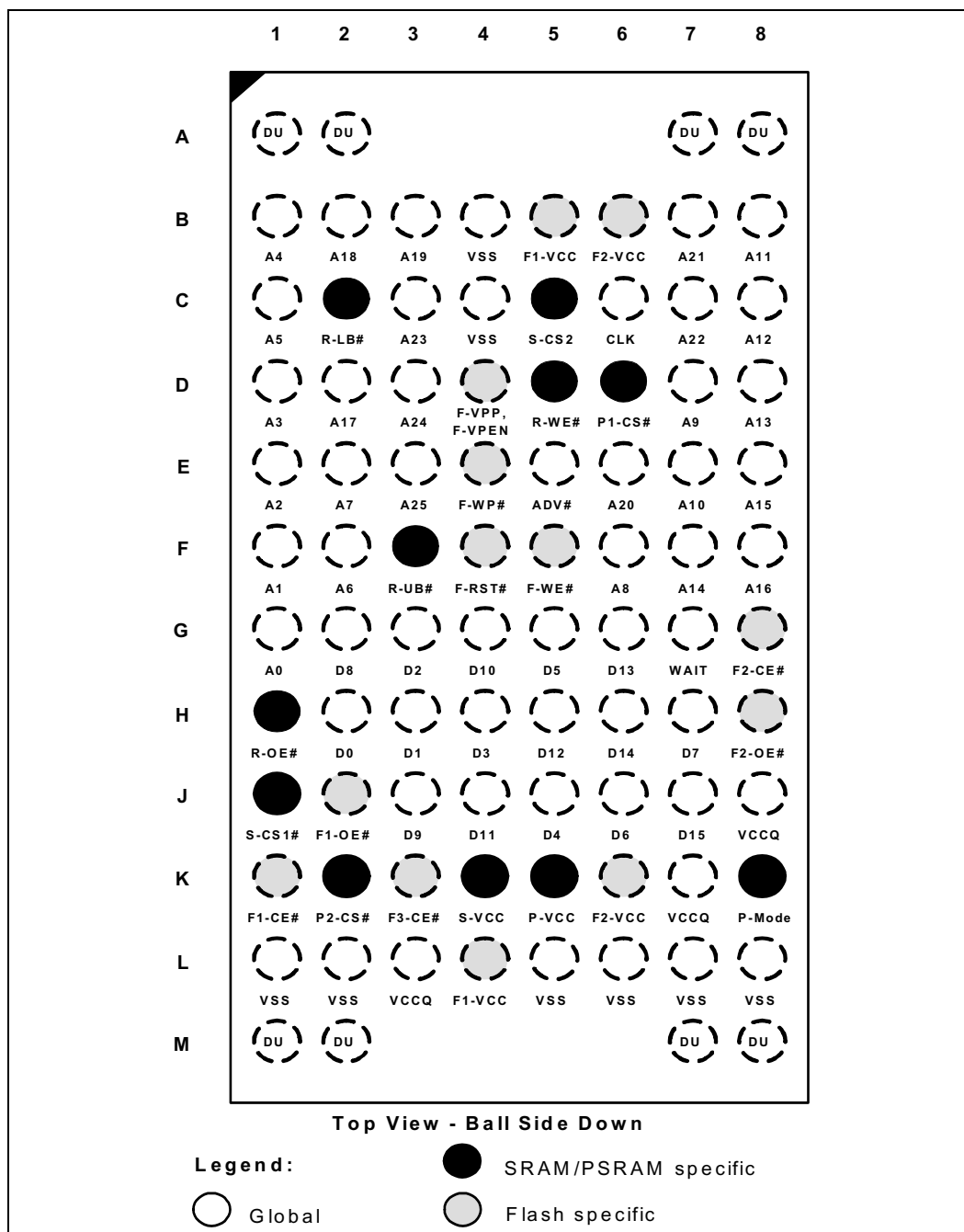


**NOTE:** On lower density devices upper address balls can be treated as RFUs. (A24 is for 512Mb and A25 is for 1Gb densities.) All ball locations are populated.

## 4.2 SCSP Package Ballout

The L30 wireless memory in Quad+ ballout device is available in an 88-ball (80-active ball) Stacked Chip Scale Package (SCSP) for the 128-Mbit device and in an 88-ball (80-active ball) Intel® Ultra-Thin Stacked Chip Scale Package (Intel® UT-SCSP) for the 256-Mbit device. Figure 7 shows the signal ballout. Refer to Section 3.0, “Package Information” on page 10.

Figure 7. 88-Ball (80-Active Ball) SCSP Package Ballout





### 4.3 Signal Descriptions

This section includes signal descriptions for the following packages:

- VF BGA Package Signal Descriptions
- SCSP Package Signal Descriptions

#### 4.3.1 VF BGA Package Signal Descriptions

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

**Table 1. Signal Descriptions (Sheet 1 of 2)**

Symbol	Type	Name and Function
A[MAX:0]	In	<b>ADDRESS:</b> Device address inputs. 64-Mbit: A[21:0]; 128-Mbit: A[22:0]; 256-Mbit: A[23:0].
D[15:0]	In/Out	<b>DATA INPUT/OUTPUTS:</b> 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 deasserted. Data is internally latched during writes.
ADV#	In	<b>ADDRESS VALID:</b> 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. In asynchronous mode, the address is latched when ADV# going high or continuously flows through if ADV# is held low.
CE#	In	<b>CHIP ENABLE:</b> 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	<b>CLOCK:</b> 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	<b>OUTPUT ENABLE:</b> Active-low input. OE#-low enables the device's output data buffers during read cycles. OE#-high places the data outputs in High-Z and WAIT in High-Z.
RST#	In	<b>RESET:</b> 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	<b>WAIT:</b> Indicates data valid in synchronous array or non-array burst reads. Configuration Register bit 10 (CR.10, WT) determines its polarity when asserted. With CE# and OE# at V <sub>IL</sub> , WAIT's active output is V <sub>OL</sub> or V <sub>OH</sub> when CE# and OE# are asserted. WAIT is high-Z if CE# or OE# is V <sub>IH</sub> . <ul style="list-style-type: none"> <li>• In synchronous array or non-array read modes, WAIT indicates invalid data when asserted and valid data when deasserted.</li> <li>• In asynchronous page mode, and all write modes, WAIT is deasserted.</li> </ul>
WE#	In	<b>WRITE ENABLE:</b> Active-low input. WE# controls writes to the device. Address and data are latched on the rising edge of WE#.
WP#	In	<b>WRITE PROTECT:</b> 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/I	<b>ERASE AND PROGRAM POWER:</b> A valid voltage on this pin allows erasing or programming. Memory contents cannot be altered when V <sub>PP</sub> ≤ V <sub>PPLK</sub> . Block erase and program at invalid V <sub>PP</sub> voltages should not be attempted. Set V <sub>PP</sub> = V <sub>CC</sub> for in-system program and erase operations. To accommodate resistor or diode drops from the system supply, the V <sub>IH</sub> level of V <sub>PP</sub> can be as low as V <sub>PPL</sub> min. V <sub>PP</sub> must remain above V <sub>PPL</sub> min to perform in-system flash modification. VPP may be 0 V during read operations. V <sub>PPH</sub> can be applied to main blocks for 1000 cycles maximum and to parameter blocks for 2500 cycles. VPP can be connected to 12 V for a cumulative total not to exceed 80 hours. Extended use of this pin at 12 V may reduce block cycling capability.

**Table 1. Signal Descriptions (Sheet 2 of 2)**

Symbol	Type	Name and Function
VCC	Pwr	<b>DEVICE CORE POWER SUPPLY:</b> Core (logic) source voltage. Writes to the flash array are inhibited when $V_{CC} \leq V_{LKO}$ . Operations at invalid $V_{CC}$ voltages should not be attempted.
VCCQ	Pwr	<b>OUTPUT POWER SUPPLY:</b> Output-driver source voltage.
VSS	Pwr	<b>GROUND:</b> Ground reference for device logic voltages. Connect to system ground.
VSSQ	Pwr	<b>GROUND:</b> Ground reference for device output voltages. Connect to system ground.
DU	-	<b>DON'T USE:</b> 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	-	<b>NO CONNECT:</b> No internal connection; can be driven or floated.
RFU	-	<b>RESERVED for FUTURE USE:</b> Reserved by Intel for future device functionality and enhancement.

### 4.3.2 128/0 and 256/0 SCSP Package Signal Descriptions

Table 2 describes the active signals used on the 128/0 and 256/0-Mbit SCSP.

**Table 2. Device Signal Descriptions for S-CSP (Sheet 1 of 2)**

Symbol	Type	Description
A[Max:0]	Input	<b>ADDRESS INPUTS:</b> Inputs for all die addresses during read and write operations. <ul style="list-style-type: none"> <li>128-Mbit Die: A[Max] = A22</li> <li>256-Mbit Die: A[Max] = A23</li> </ul>
D[15:0]	Input/Output	<b>DATA INPUTS/OUTPUTS:</b> Inputs data and commands during write cycles, outputs data during read cycles. Data signals float when the device or its outputs are deselected. Data is internally latched during writes.
CE#1 CE#2	Input	<b>FLASH CHIP ENABLE:</b> Low-true: CE#-low selects the associated flash memory die. When asserted, flash internal control logic, input buffers, decoders, and sense amplifiers are active. When deasserted, the associated flash die is deselected, power is reduced to standby levels, data and WAIT outputs are placed in high-Z state.  CE#1 selects flash die #1; CE#2 selects flash die #2. CE#2 is available on stacked combinations with two flash die and is RFU (Reserved For Future Use) on stacked combinations with only one flash die.
S-CS1# S-CS2	Input	<b>SRAM CHIP SELECTS:</b> When both SRAM chip selects are asserted, SRAM internal control logic, input buffers, decoders, and sense amplifiers are active. When either/both SRAM chip selects are deasserted (S-CS1# = VIH or S-CS2 = VIL), the SRAM is deselected and its power is reduced to standby levels.  <b>Treat this signal as NC (No Connect) for this device.</b>
P-CS#	Input	<b>PSRAM CHIP SELECT:</b> Low-true; When asserted, PSRAM internal control logic, input buffers, decoders, and sense amplifiers are active. When deasserted, the PSRAM is deselected and its power is reduced to standby levels.  <b>Treat this signal as NC (No Connect) for this device.</b>
OE#1 OE#2	Input	<b>FLASH OUTPUT ENABLE:</b> Low-true; OE#-low enables the flash output buffers. OE#-high disables the flash output buffers, and places the flash outputs in High-Z.  OE#1 controls the outputs of flash die #1; OE#2 controls the outputs of flash die #2. OE#2 is available on stacked combinations with two flash die and is RFU on stacked combinations with only one flash die.
R-OE#	Input	<b>RAM OUTPUT ENABLE:</b> Low-true; R-OE#-low enables the selected RAM output buffers. R-OE#-high disables the RAM output buffers, and places the selected RAM outputs in High-Z.  <b>Treat this signal as NC (No Connect) for this device.</b>
WE#	Input	<b>FLASH WRITE ENABLE:</b> Low-true; WE# controls writes to the selected flash die. Address and data are latched on the rising edge of WE#.
R-WE#	Input	<b>RAM WRITE ENABLE:</b> Low-true; R-WE# controls writes to the selected RAM die.  <b>Treat this signal as NC (No Connect) for this device.</b>
CLK	Input	<b>FLASH CLOCK:</b> 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.
WAIT	Output	<b>FLASH WAIT:</b> Indicates data valid in synchronous array or non-array burst reads. Configuration Register bit 10 (CR.10, WT) determines its polarity when asserted. With CE# and OE# at V <sub>IL</sub> , WAIT's active output is V <sub>OL</sub> or V <sub>OH</sub> when CE# and OE# are asserted. WAIT is high-Z if CE# or OE# is V <sub>IH</sub> . <ul style="list-style-type: none"> <li>In synchronous array or non-array read modes, WAIT indicates invalid data when asserted and valid data when deasserted.</li> <li>In asynchronous page mode, and all write modes, WAIT is deasserted.</li> </ul>

Table 2. Device Signal Descriptions for S-CSP (Sheet 2 of 2)

WP#	Input	<b>FLASH WRITE PROTECT:</b> Low-true; WP# enables/disables the lock-down protection mechanism of the selected flash die. WP#-low enables the lock-down mechanism - locked down blocks cannot be unlocked with software commands. WP#-high disables the lock-down mechanism, allowing locked down blocks to be unlocked with software commands.
ADV#	Input	<b>FLASH ADDRESS VALID:</b> 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. In asynchronous mode, the address is latched when ADV# going high or continuously flows through if ADV# is held low.
R-UB# R-LB#	Input	<b>RAM UPPER / LOWER BYTE ENABLES:</b> Low-true; During RAM reads, R-UB#-low enables the RAM high order bytes on D[15:8], and R-LB#-low enables the RAM low-order bytes on D[7:0]. <b>Treat this signal as NC (No Connect) for this device.</b>
RST#	Input	<b>FLASH RESET:</b> Low-true; RST#-low initializes flash internal circuitry and disables flash operations. RST#-high enables flash operation. Exit from reset places the flash in asynchronous read array mode.
P-Mode	Input	<b>PSRAM MODE:</b> Low-true; P-MODE is used to program the configuration register, and enter/exit low power mode. <b>Treat this signal as NC (No Connect) for this device.</b>
VPP, VPEM	Power	<b>FLASH PROGRAM / ERASE POWER:</b> A valid voltage on this pin allows erasing or programming. Memory contents cannot be altered when $V_{PP} \leq V_{PPLK}$ . Block erase and program at invalid $V_{PP}$ voltages should not be attempted. Set $V_{PP} = V_{CC}$ for in-system program and erase operations. To accommodate resistor or diode drops from the system supply, the $V_{IH}$ level of $V_{PP}$ can be as low as $V_{PP1 \text{ min}}$ . $V_{PP}$ must remain above $V_{PP1 \text{ min}}$ to perform in-system flash modification. $V_{PP}$ may be 0 V during read operations. $V_{PP2}$ can be applied to main blocks for 1000 cycles maximum and to parameter blocks for 2500 cycles. $V_{PP}$ can be connected to 12 V for a cumulative total not to exceed 80 hours. Extended use of this pin at 12 V may reduce block cycling capability <b>VPEM ((Erase/Program/Block Lock Enables) is not available for L18/L30 products.</b>
VCC1 VCC2	Power	<b>FLASH LOGIC POWER:</b> VCC1 supplies power to the core logic of flash die #1; VCC2 supplies power to the core logic of flash die #2. Write operations are inhibited when $V_{CC} < V_{LKO}$ . Device operations at invalid $V_{CC}$ voltages should not be attempted.
S-VCC	Power	<b>SRAM POWER SUPPLY:</b> Supplies power for SRAM operations. <b>Treat this signal as NC (No Connect) for this device.</b>
P-VCC	Power	<b>PSRAM POWER SUPPLY:</b> Supplies power for PSRAM operations. <b>Treat this signal as NC (No Connect) for this device.</b>
VCCQ	Power	<b>FLASH I/O POWER:</b> Supply power for the input and output buffers.
VSS	Power	<b>GROUND:</b> Connect to system ground. Do not float any VSS connection.
RFU		<b>RESERVED for FUTURE USE:</b> Reserve for future device functionality/enhancements. Contact Intel regarding their future use.
DU		<b>DON'T USE:</b> Do not connect to any other signal, or power supply; must be left floating.
NC		<b>NO CONNECT:</b> No internal connection; can be driven or floated.

## 4.4 Memory Map

For the 64-Mb device and the 128-Mb device, the 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 16-Kword blocks and seven 64-Kword blocks. There are multiple 8-Mbit main partitions. The 8-Mbit main partitions each contains eight 64-Kword 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.

For the 256-Mb device, the memory array is divided into multiple 16-Mbit partitions. Each device contains one parameter partition and fifteen main partitions. The 16-Mbit top or bottom parameter partition contains four 16-Kword blocks and fifteen 64-Kword blocks. There are fifteen 16-Mbit main partitions. The 16-Mbit main partitions each contains sixteen 64-Kword blocks. [Table 3](#) and [Table 4](#) show the top and bottom parameter memory maps.

**Table 3. Top Parameter Memory Map**

		Size (KW)	Blk	64-Mbit
8-Mbit Parameter Partition	One Partition	16	66	3FC000-3FFFFFF
		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
8-Mbit Parameter Partition	One Partition	16	130	7FC000-7FFFFFF
		16	129	7F8000-7FBFFF
		16	128	7F4000-7F7FFF
		16	127	7F0000-7F3FFF
		64	126	7E0000-7EFFFF
		...	...	...
		64	120	780000-78FFFF
8-Mbit Main Partitions	Fifteen Partitions	64	119	770000-77FFFF
		...	...	...
		64	0	000000-00FFFF

		Size (KW)	Blk	256-Mbit
16-Mbit Parameter Partition	One Partition	16	258	FFC000-FFFFFF
		16	257	FF8000-FFBFFF
		16	256	FF4000-FF7FFF
		16	255	FF0000-FF3FFF
		64	254	FE0000-FEFFFF
		...	...	...
		64	240	F00000-FFFFFF
16-Mbit Main Partitions	Seven Partitions	64	239	EF0000-EFFFFFF
		...	...	...
		64	128	800000-80FFFF
	Eight Partitions	64	127	7F0000-7FFFFFF
		64	0	000000-00FFFF

**Table 4. Bottom Parameter Memory Map**

Size (KW)	Blk	64-Mbit
-----------	-----	---------

8-Mbit Main Partitions	Seven Partitions	64	66	3F0000-3FFFFF
		⋮	⋮	⋮
		64	11	080000-08FFFF
8-Mbit Parameter Partition	One Partition	64	10	070000-07FFFF
		⋮	⋮	⋮
		64	4	010000-01FFFF
		16	3	00C000-00FFFF
		16	2	008000-00BFFF
		16	1	004000-007FFF
		16	0	000000-003FFF

Size (KW)	Blk	128-Mbit
-----------	-----	----------

8-Mbit Main Partitions	Fifteen Partitions	64	130	7F0000-7FFFFF
		⋮	⋮	⋮
		64	11	080000-08FFFF
8-Mbit Parameter Partition	One Partition	64	10	070000-07FFFF
		⋮	⋮	⋮
		64	4	010000-01FFFF
		16	3	00C000-00FFFF
		16	2	008000-00BFFF
		16	1	004000-007FFF
		16	0	000000-003FFF

Size (KW)	Blk	256-Mbit
-----------	-----	----------

16-Mbit Main Partitions	Eight Partitions	64	258	FF0000-FFFFFF	
		⋮	⋮	⋮	⋮
		64	131	100000-10FFFF	
	Seven Partitions	64	130	7F0000-7FFFFF	
		⋮	⋮	⋮	⋮
	64	19	100000-10FFFF		
16-Mbit Parameter Partition	One Partition	64	18	0F0000-0FFFFF	
		⋮	⋮	⋮	
		64	4	010000-01FFFF	
		16	3	00C000-00FFFF	
		16	2	008000-00BFFF	
		16	1	004000-007FFF	
16	0	000000-003FFF			

## 5.0 Maximum Ratings and Operating Conditions

### 5.1 Absolute Maximum Ratings

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

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 +10 V	1,2,3
VCC voltage	–0.2 V to +2.5 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.

### 5.2 Operating Conditions

Symbol	Parameter	Min	Max	Units	Notes	
$T_C$	Operating Temperature	–25	+85	°C	1	
$V_{CC}$	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			
$V_{PPH}$	Factory word programming $V_{PP}$	8.5	9.5		Hours	2
$t_{PPH}$	Maximum VPP Hours	$V_{PP} = V_{PPH}$		80		
Block Erase Cycles	Main and Parameter Blocks	$V_{PP} = V_{CC}$		100,000		
	Main Blocks	$V_{PP} = V_{PPH}$		1000		
	Parameter Blocks	$V_{PP} = V_{PPH}$		2500		

**NOTES:**

1.  $T_C$  = Case Temperature
2. In typical operation, the VPP program voltage is  $V_{PPL}$ . VPP can be connected to 8.50 V – 9.5 V for 1000 cycles on main blocks, and 2500 cycles on parameter blocks.

## 6.0 Electrical Specifications

### 6.1 DC Current Characteristics

Sym	Parameter		V <sub>CC</sub>		Unit	Test Conditions	Notes	
			1.7 V – 2.0 V					
			V <sub>CCQ</sub>	2.2 V – 3.3 V				
		Typ	Max					
I <sub>LI</sub>	Input Load Current				±2	μA	V <sub>CC</sub> = V <sub>CC</sub> Max V <sub>CCQ</sub> = V <sub>CCQ</sub> Max V <sub>IN</sub> = V <sub>CCQ</sub> or GND	1
I <sub>LO</sub>	Output Leakage Current	D[15:0], WAIT			±10	μA	V <sub>CC</sub> = V <sub>CC</sub> Max V <sub>CCQ</sub> = V <sub>CCQ</sub> Max V <sub>IN</sub> = V <sub>CCQ</sub> or GND	
I <sub>CCS</sub> I <sub>CCD</sub>	V <sub>CC</sub> Standby, Power Down		64 Mbit	20	35	μA	V <sub>CC</sub> = V <sub>CC</sub> Max V <sub>CCQ</sub> = V <sub>CCQ</sub> Max CE# = V <sub>CCQ</sub> RST# = V <sub>CCQ</sub> (for I <sub>CCS</sub> ) RST# = GND (for I <sub>CCD</sub> ) WP# = V <sub>IH</sub>	1,2
		128 Mbit	30	75				
		256 Mbit	55	115				
I <sub>CCAPS</sub>	APS		64 Mbit	20	35	μA	V <sub>CC</sub> = V <sub>CC</sub> Max V <sub>CCQ</sub> = V <sub>CCQ</sub> Max CE# = V <sub>SSQ</sub> RST# = V <sub>CCQ</sub> All inputs are at rail to rail (V <sub>CCQ</sub> or V <sub>SSQ</sub> ).	
		128 Mbit	30	75				
		256 Mbit	55	115				
I <sub>CCR</sub>	Average V <sub>CC</sub> Read Current	Asynchronous Single-Word f = 5MHz (1 CLK)	14	16	mA		V <sub>CC</sub> = V <sub>CC</sub> MAX CE# = V <sub>IL</sub> OE# = V <sub>IH</sub> Inputs: V <sub>IL</sub> or V <sub>IH</sub>	1
		Page-Mode Read f = 13 MHz (5 CLK)	9	10	mA	4-Word Read		
		Synchronous Burst Read f = 40MHz	13	17	mA	Burst length=4		
			15	19	mA	Burst length=8		
			17	21	mA	Burst length=16		
			21	26	mA	Burst length = Continuous		
		Synchronous Burst Read f = 54MHz	16	19	mA	Burst length=4		
			19	23	mA	Burst length=8		
			22	26	mA	Burst length=16		
			23	28	mA	Burst Length = Continuous		
I <sub>CCW</sub> , I <sub>CCE</sub>	V <sub>CC</sub> Program Current,		36	51	mA	V <sub>PP</sub> = V <sub>PP</sub> L, program/erase in progress	1,3,4, 7	
	V <sub>CC</sub> Erase Current		26	33	mA	V <sub>PP</sub> = V <sub>PP</sub> H, program/erase in progress	1,3,5, 7	
I <sub>CCWS</sub> , I <sub>CCES</sub>	V <sub>CC</sub> Program Suspend Current,		64 Mbit	20	35	μA	CE# = V <sub>CCQ</sub> ; suspend in progress	1,6,3
	V <sub>CC</sub> Erase Suspend Current		128 Mbit	30	75			
			256 Mbit	55	115			
I <sub>PPS</sub> , I <sub>PPWS</sub> , I <sub>PPES</sub>	V <sub>PP</sub> Standby Current, V <sub>PP</sub> Program Suspend Current, V <sub>PP</sub> Erase Suspend Current		0.2	5	μA	V <sub>PP</sub> = V <sub>PP</sub> L, suspend in progress	1,3	



Sym	Parameter	V <sub>CC</sub>		Unit	Test Conditions	Notes
		V <sub>CC</sub>	1.7 V – 2.0 V			
		V <sub>CCQ</sub>	2.2 V – 3.3 V			
		Typ	Max			
I <sub>PPR</sub>	V <sub>PP</sub> Read		2	15	μA	V <sub>PP</sub> ≤ V <sub>CC</sub>
I <sub>PPW</sub>	V <sub>PP</sub> Program Current		0.05	0.10	mA	V <sub>PP</sub> = V <sub>PP<sub>L</sub></sub> , program in progress
			8	22		V <sub>PP</sub> = V <sub>PP<sub>H</sub></sub> , program in progress
I <sub>PPE</sub>	V <sub>PP</sub> Erase Current		0.05	0.10	mA	V <sub>PP</sub> = V <sub>PP<sub>L</sub></sub> , erase in progress
			8	22		V <sub>PP</sub> = V <sub>PP<sub>H</sub></sub> , erase in progress

**NOTES:**

- All currents are RMS unless noted. Typical values at typical V<sub>CC</sub>, T<sub>C</sub> = +25°C.
- I<sub>CCS</sub> is the average current measured over any 5 ms time interval 5 μs after CE# is deasserted.
- Sampled, not 100% tested.
- V<sub>CC</sub> read + program current is the sum of V<sub>CC</sub> read and V<sub>CC</sub> program currents.
- V<sub>CC</sub> read + erase current is the sum of V<sub>CC</sub> read and V<sub>CC</sub> erase currents.
- I<sub>CCES</sub> is specified with the device deselected. If device is read while in erase suspend, current is I<sub>CCES</sub> plus I<sub>CCR</sub>.
- I<sub>CCW</sub>, I<sub>CCE</sub> measured over typical or max times specified in [Section 7.6, "Program and Erase Characteristics"](#) on page 37

## 6.2 DC Voltage Characteristics

Sym	Parameter	V <sub>CC</sub>		Unit	Test Condition	Notes
		V <sub>CC</sub>	1.7 V – 2.0 V			
		V <sub>CCQ</sub>	2.2 V – 3.3 V			
		Min	Max			
V <sub>IL</sub>	Input Low Voltage		0	0.4	V	1
V <sub>IH</sub>	Input High Voltage		V <sub>CCQ</sub> - 0.4	V <sub>CCQ</sub>	V	
V <sub>OL</sub>	Output Low Voltage			0.1	V	V <sub>CC</sub> = V <sub>CCMIN</sub> V <sub>CCQ</sub> = V <sub>CCQMIN</sub> I <sub>OL</sub> = 100 μA
V <sub>OH</sub>	Output High Voltage		V <sub>CCQ</sub> - 0.1		V	V <sub>CC</sub> = V <sub>CCMIN</sub> V <sub>CCQ</sub> = V <sub>CCQMIN</sub> I <sub>OH</sub> = -100 μA
V <sub>PP<sub>LK</sub></sub>	V <sub>PP</sub> Lock-Out Voltage			0.4	V	2
V <sub>LKO</sub>	V <sub>CC</sub> Lock Voltage		1.0		V	
V <sub>LKOQ</sub>	V <sub>CCQ</sub> Lock Voltage		0.9		V	

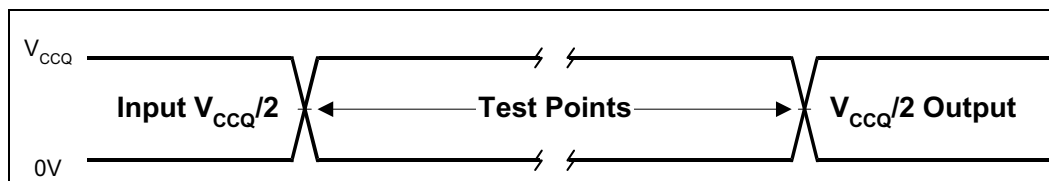
**NOTES:**

- V<sub>IL</sub> can undershoot to -0.4V and V<sub>IH</sub> can overshoot to V<sub>CCQ</sub>+0.4V for durations of 20 ns or less.
- V<sub>PP</sub> < V<sub>PP<sub>LK</sub></sub> inhibits erase and program operations. Do not use V<sub>PP<sub>L</sub></sub> and V<sub>PP<sub>H</sub></sub> outside their valid ranges.

## 7.0 AC Characteristics

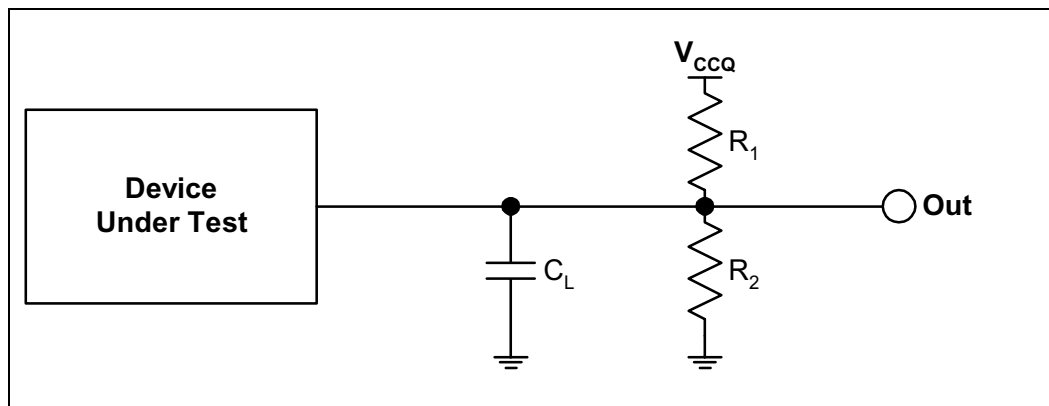
### 7.1 AC Test Conditions

Figure 8. 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  $V_{CC} = V_{CCMin}$ .

Figure 9. Transient Equivalent Testing Load Circuit



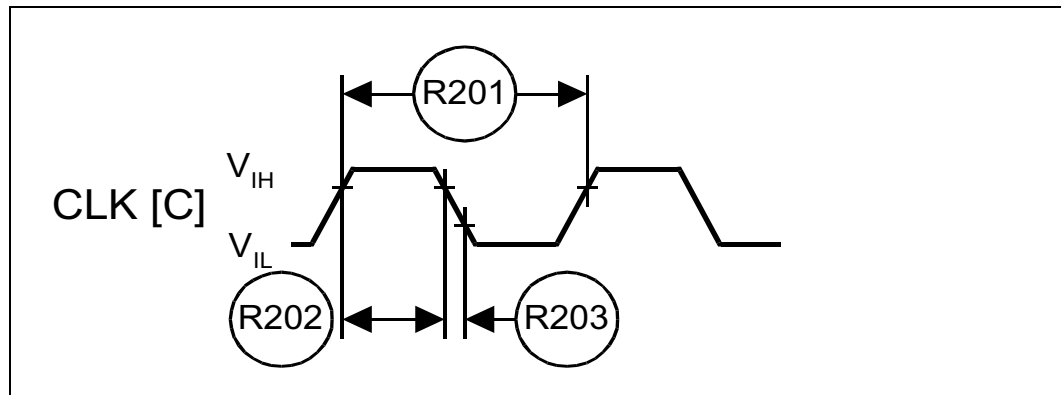
**NOTES:**

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 5. Test configuration component value for worst case speed conditions

Test Configuration	$C_L$ (pF)	$R_1$ ( $\Omega$ )	$R_2$ ( $\Omega$ )
2.0 V Standard Test	30	22K	22K

Figure 10. Clock Input AC Waveform



## 7.2 Capacitance

Table 6. Capacitance

Symbol	Parameter <sup>1</sup>	Typ	Max	Unit	Condition	Note
$C_{IN}$	Input Capacitance	6	8	pF	$V_{IN} = 0.0\text{ V}$	1,2,3
$C_{OUT}$	Output Capacitance	8	12	pF	$V_{OUT} = 0.0\text{ V}$	1,2
$C_{CE\#}$	CE# Input Capacitance	10	12	pF	$V_{IN} = 0.0\text{ V}$	

**NOTES:**

1.  $T_C = +25^\circ\text{C}$ ,  $f = 1\text{ MHz}$ .
2. Sampled, not 100% tested.
3.  $C_{IN}(\text{Max}) = 10\text{ pF}$  for 256Mbit Density

## 7.3 AC Read Specifications for 64-Mbit and 128-Mbit Densities

Num	Symbol	Parameter	Speed	-85		Units	Notes
				Min	Max		
<b>Asynchronous Specifications</b>							
R1	$t_{AVAV}$	Read cycle time		85		ns	6
R2	$t_{AVQV}$	Address to output valid			85	ns	
R3	$t_{ELQV}$	CE# low to output valid			85	ns	
R4	$t_{GLQV}$	OE# low to output valid			25	ns	1,2
R5	$t_{PHQV}$	RST# high to output valid			150	ns	1
R6	$t_{ELQX}$	CE# low to output in low-Z		0		ns	1,3
R7	$t_{GLQX}$	OE# low to output in low-Z		0		ns	1,2,3
R8	$t_{EHQZ}$	CE# high to output in high-Z			24	ns	1,3
R9	$t_{GHQZ}$	OE# high to output in high-Z			24	ns	
R10	$t_{OH}$	Output hold from first occurring address, CE#, or OE# change		0		ns	
R11	$t_{EHEL}$	CE# pulse width high		20		ns	1
R12	$t_{ELTV}$	CE# low to WAIT valid			16	ns	1
R13	$t_{EHTZ}$	CE# high to WAIT high Z			17	ns	1,3
R15	$t_{GLTV}$	OE# low to WAIT valid			17	ns	1
R16	$t_{GLTX}$	OE# low to WAIT in low-Z		0		ns	1,3
R17	$t_{GHTZ}$	OE# high to WAIT in high-Z			20	ns	1,3
<b>Latching Specifications</b>							
R101	$t_{AVVH}$	Address setup to ADV# high		10		ns	1
R102	$t_{ELVH}$	CE# low to ADV# high		10		ns	
R103	$t_{VLQV}$	ADV# low to output valid			85	ns	
R104	$t_{VLVH}$	ADV# pulse width low		10		ns	
R105	$t_{VHVL}$	ADV# pulse width high		10		ns	
R106	$t_{VHAX}$	Address hold from ADV# high		9		ns	1,4
R108	$t_{APA}$	Page address access			25	ns	1
R111	$t_{phvh}$	RST# high to ADV# high		30		ns	1
<b>Clock Specifications</b>							
R200	$f_{CLK}$	CLK frequency			52	MHz	1,3
R201	$t_{CLK}$	CLK period		19.2		ns	
R202	$t_{CH/CL}$	CLK high/low time		9		ns	
R203	$t_{FCLK/RCLK}$	CLK fall/rise time			3	ns	
<b>Synchronous Specifications</b>							
R301	$t_{AVCH/L}$	Address setup to CLK		9		ns	1
R302	$t_{VLCH/L}$	ADV# low setup to CLK		9		ns	
R303	$t_{ELCH/L}$	CE# low setup to CLK		9		ns	
R304	$t_{CHQV} / t_{CLQV}$	CLK to output valid			17	ns	
R305	$t_{CHQX}$	Output hold from CLK		3		ns	1,5
R306	$t_{CHAX}$	Address hold from CLK		10		ns	1,4,5
R307	$t_{CHTV}$	CLK to WAIT valid			20	ns	1,5
R311	$t_{CHVL}$	CLK Valid to ADV# Setup		0		ns	1
R312	$t_{CHTX}$	WAIT Hold from CLK		3		ns	1,5

### NOTES:

- See Figure 8, "AC Input/Output Reference Waveform" on page 26 for timing measurements and maximum allowable input slew rate.
- OE# may be delayed by up to  $t_{ELQV} - t_{GLQV}$  after CE#'s falling edge without impact to  $t_{ELQV}$ .
- Sampled, not 100% tested.
- Address hold in synchronous burst mode is  $t_{CHAX}$  or  $t_{VHAX}$ , whichever timing specification is satisfied first.
- Applies only to subsequent synchronous reads.

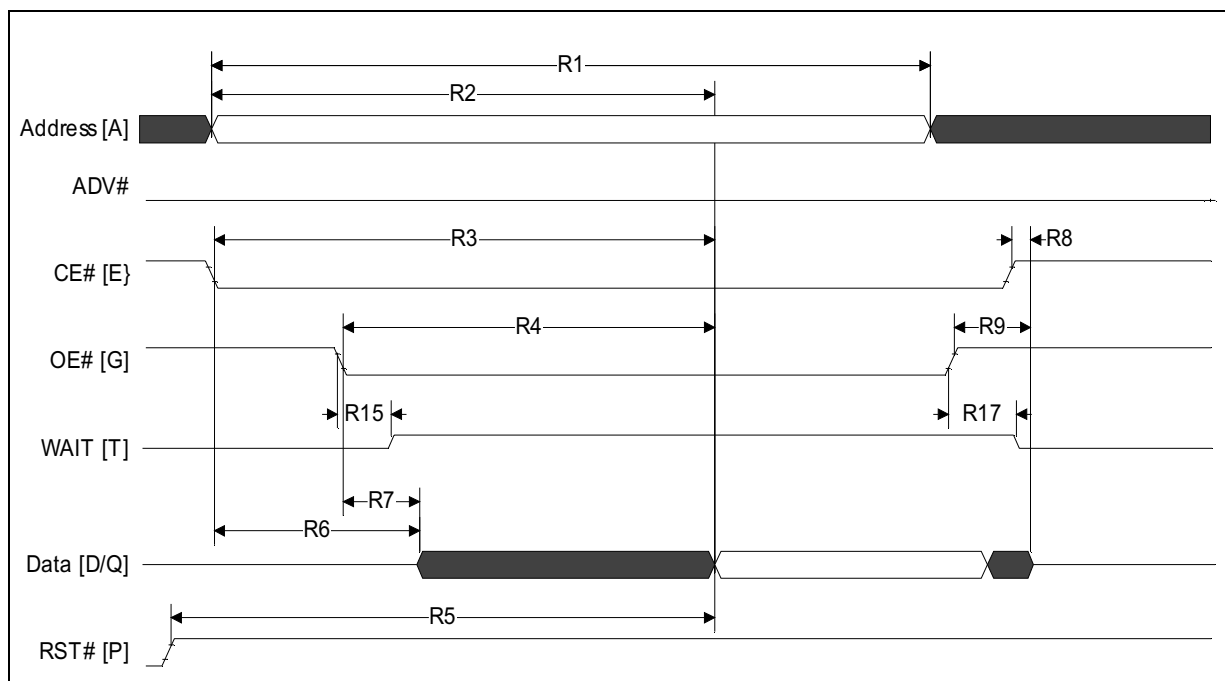
## 7.4 AC Read Specifications for 256-Mb Density

Num	Symbol	Parameter	Speed	-85		Units	Notes
				Min	Max		
<b>Asynchronous Specifications</b>							
R1	$t_{AVAV}$	Read cycle time	$V_{CC} = 1.8 V - 2.0$	85		ns	6
			$V_{CC} = 1.7 V - 2.0$	88			
R2	$t_{AVQV}$	Address to output valid	$V_{CC} = 1.8 V - 2.0$		85	ns	
			$V_{CC} = 1.7 V - 2.0$		88		
R3	$t_{ELQV}$	CE# low to output valid	$V_{CC} = 1.8 V - 2.0$		85	ns	
			$V_{CC} = 1.7 V - 2.0$		88		
R4	$t_{GLQV}$	OE# low to output valid			25	ns	1,2
R5	$t_{PHQV}$	RST# high to output valid			150	ns	1
R6	$t_{ELQX}$	CE# low to output in low-Z		0		ns	1,3
R7	$t_{GLQX}$	OE# low to output in low-Z		0		ns	1,2,3
R8	$t_{EHQZ}$	CE# high to output in high-Z			24	ns	1,3
R9	$t_{GHQZ}$	OE# high to output in high-Z			24	ns	
R10	$t_{OH}$	Output hold from first occurring address, CE#, or OE# change		0		ns	
R11	$t_{EHEL}$	CE# pulse width high		20		ns	1
R12	$t_{ELTV}$	CE# low to WAIT valid			16	ns	1
R13	$t_{EHTZ}$	CE# high to WAIT high Z			17	ns	1,3
R15	$t_{GLTV}$	OE# low to WAIT valid			17	ns	1
R16	$t_{GLTX}$	OE# low to WAIT in low-Z		0		ns	1,3
R17	$t_{GHTZ}$	OE# high to WAIT in high-Z			20	ns	1,3
<b>Latching Specifications</b>							
R101	$t_{AVVH}$	Address setup to ADV# high		10		ns	1
R102	$t_{ELVH}$	CE# low to ADV# high		10			
R103	$t_{VLQV}$	ADV# low to output valid	$V_{CC} = 1.8 V - 2.0$		85	ns	
			$V_{CC} = 1.7 V - 2.0$		88		
R104	$t_{VLVH}$	ADV# pulse width low		10		ns	
R105	$t_{VHVL}$	ADV# pulse width high		10		ns	
R106	$t_{VHAX}$	Address hold from ADV# high		9		ns	1,4
R108	$t_{APA}$	Page address access			25	ns	1
R111	$t_{phvh}$	RST# high to ADV# high		30		ns	1
<b>Clock Specifications</b>							
R200	$f_{CLK}$	CLK frequency			52	MHz	1,3
R201	$t_{CLK}$	CLK period		19.2		ns	
R202	$t_{CH/CL}$	CLK high/low time		9		ns	
R203	$t_{FCLK/RCLK}$	CLK fall/rise time			3	ns	
<b>Synchronous Specifications</b>							
R301	$t_{AVCH/L}$	Address setup to CLK		9		ns	1
R302	$t_{VLCH/L}$	ADV# low setup to CLK		9			
R303	$t_{ELCH/L}$	CE# low setup to CLK		9			
R304	$t_{CHQV} / t_{CLQV}$	CLK to output valid			17		
R305	$t_{CHQX}$	Output hold from CLK		3		ns	1,5
R306	$t_{CHAX}$	Address hold from CLK		10		ns	1,4,5
R307	$t_{CHTV}$	CLK to WAIT valid			17	ns	1,5
R311	$t_{CHVL}$	CLK Valid to ADV# Setup		0		ns	1
R312	$t_{CHTX}$	WAIT Hold from CLK		3		ns	1,5

**NOTES:**

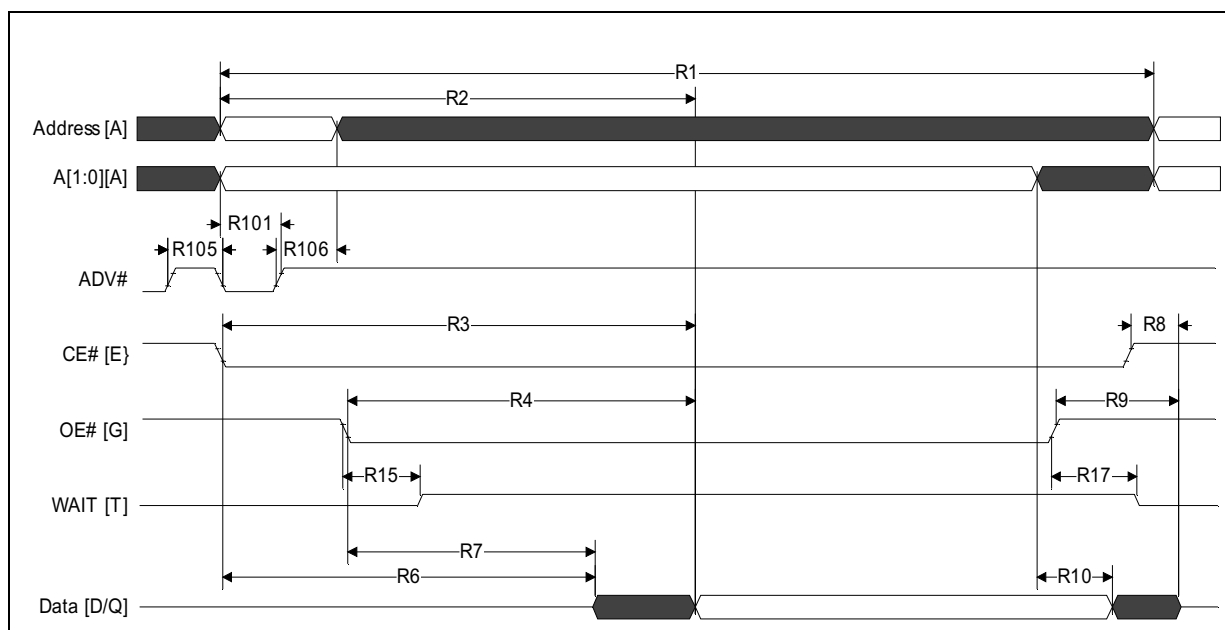
1. See Figure 8, "AC Input/Output Reference Waveform" on page 26 for timing measurements and max allowable input slew rate.
2. OE# may be delayed by up to  $t_{ELQV} - t_{GLQV}$  after CE#'s falling edge without impact to  $t_{ELQV}$ .
3. Sampled, not 100% tested.
4. Address hold in synchronous burst mode is  $t_{CHAX}$  or  $t_{VHAX}$ , whichever timing specification is satisfied first.
5. Applies only to subsequent synchronous reads.

Figure 11. Asynchronous Single-Word Read (ADV# Low)



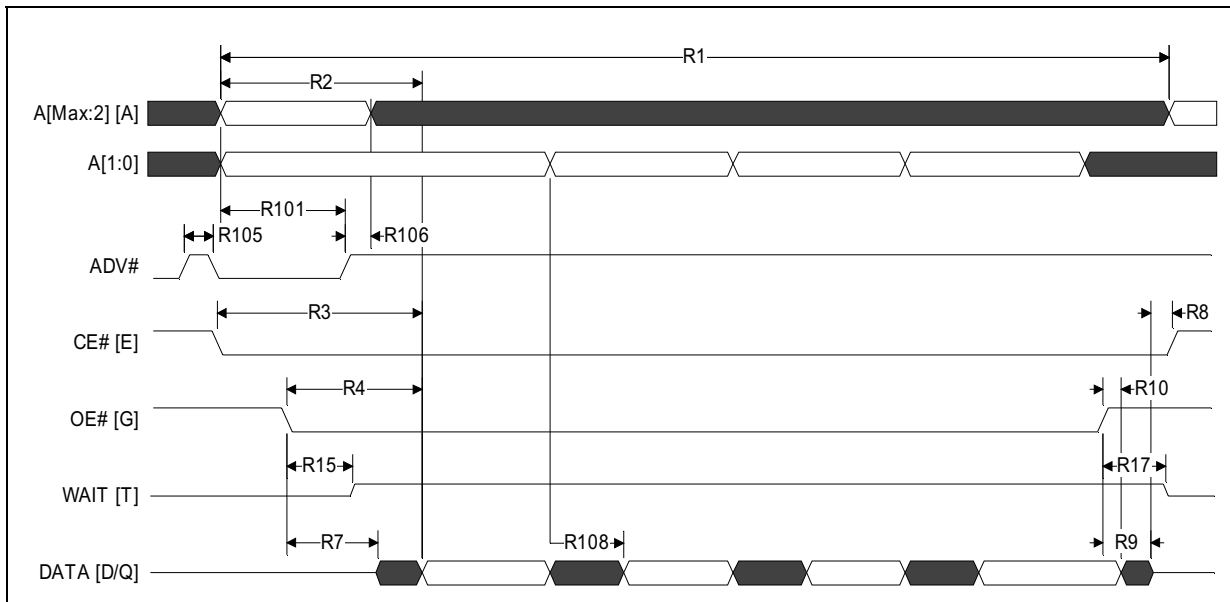
NOTE: WAIT shown deasserted during asynchronous read mode (CR[10]=0 Wait asserted low).

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



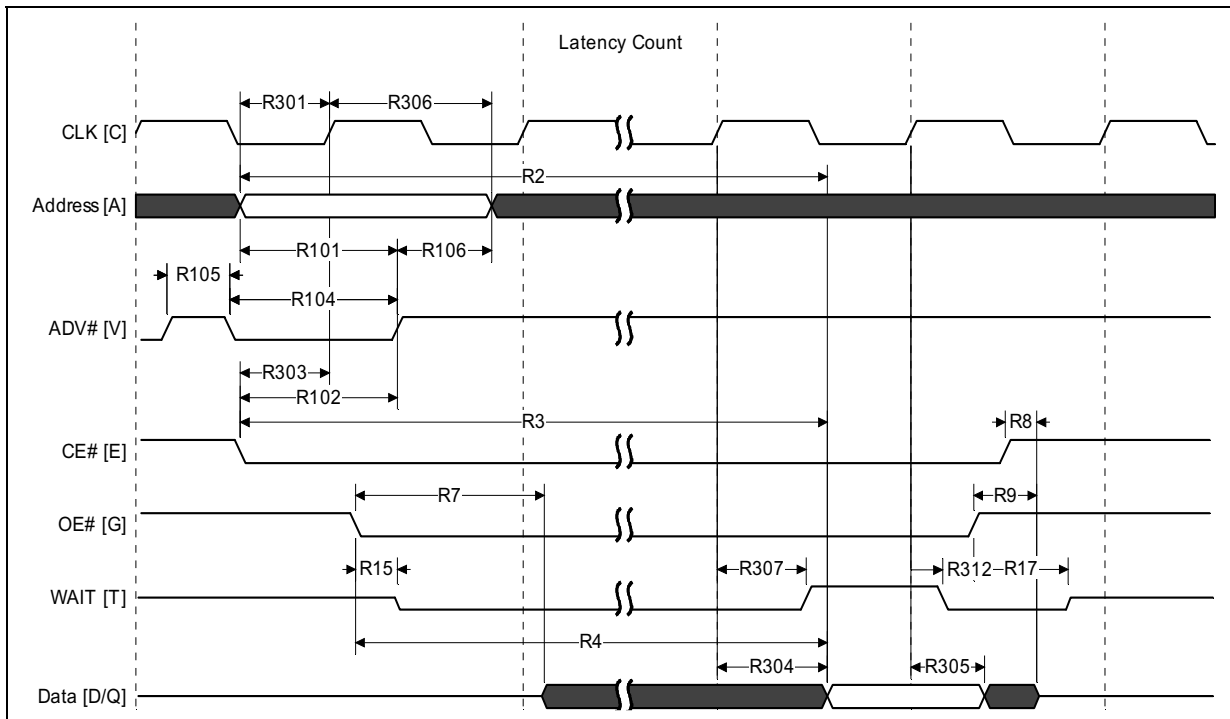
NOTE: WAIT shown deasserted during asynchronous read mode (CR[10]=0 Wait asserted low).

**Figure 13. Asynchronous Page-Mode Read Timing**



**NOTE:** WAIT shown deasserted during asynchronous read mode (CR[10]=0 Wait asserted low)

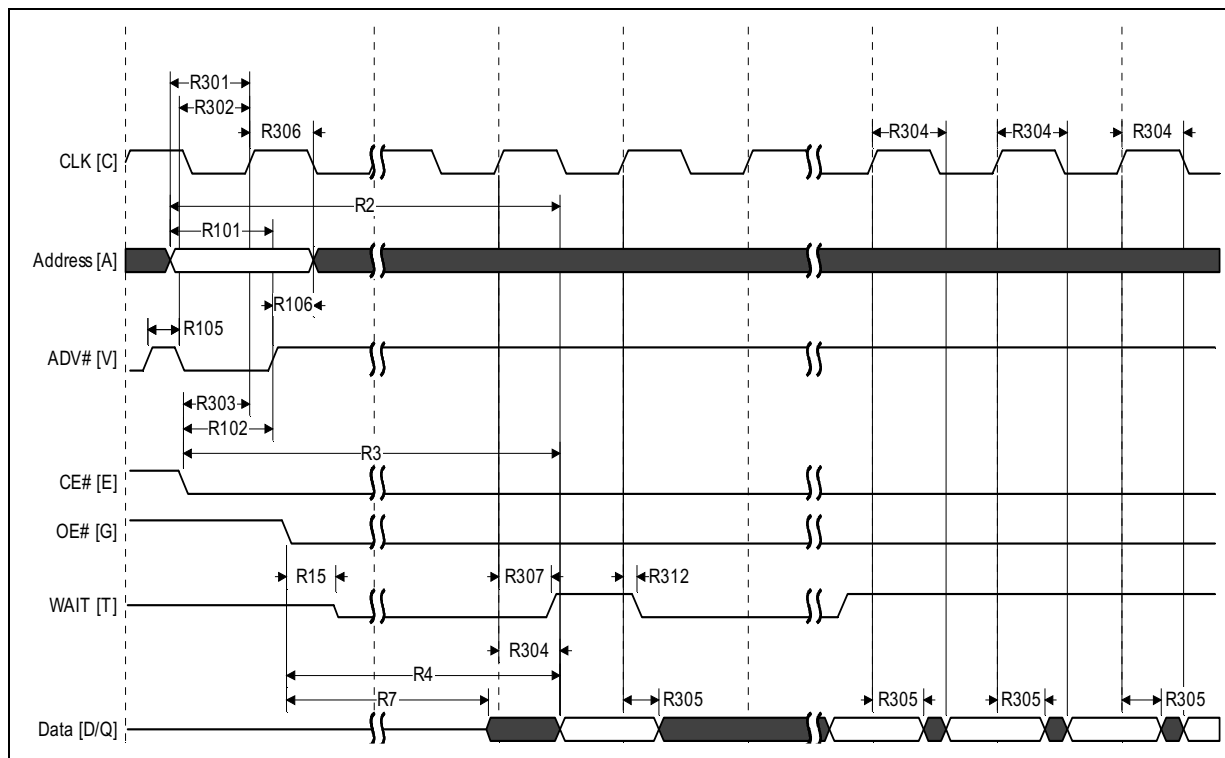
**Figure 14. Synchronous Single-Word Array or Non-array Read Timing**



**NOTES:**

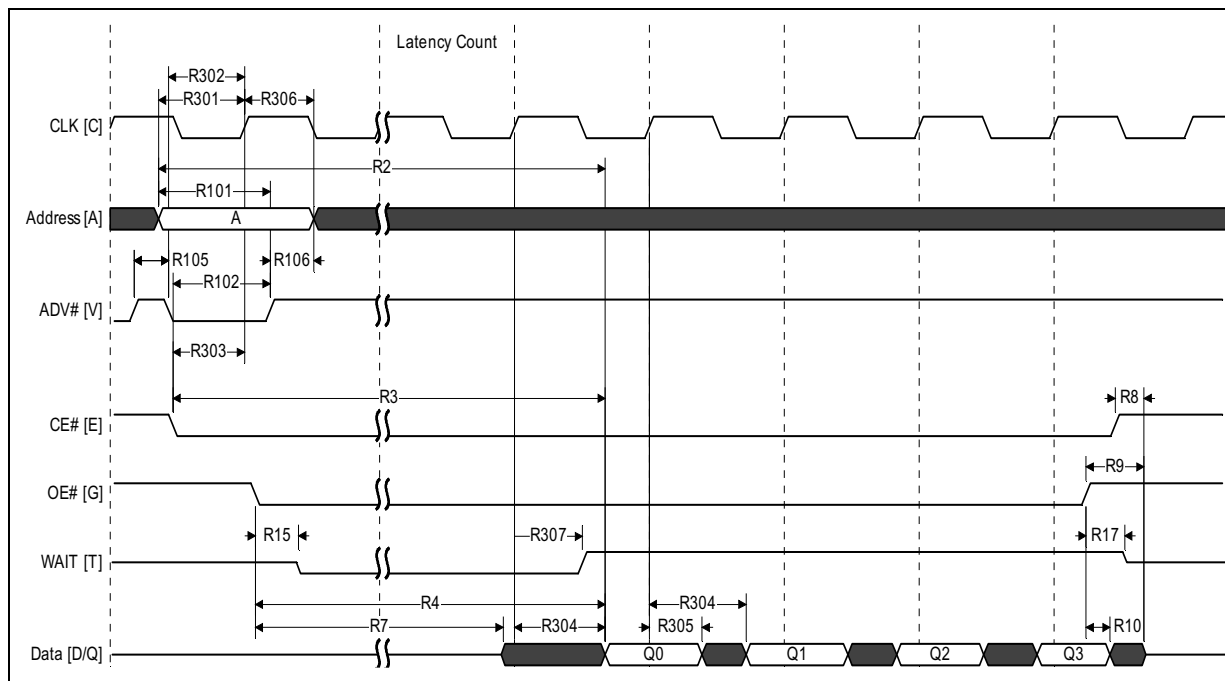
1. WAIT is driven per OE# assertion during synchronous array or non-array read, and can be configured to assert either during or one data cycle before valid data.
2. This diagram illustrates the case in which an n-word burst is initiated to the flash memory array and it is terminated by CE# deassertion after the first word in the burst.

Figure 15. Continuous Burst Read, showing an Output Delay Timing



**NOTE:** 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.

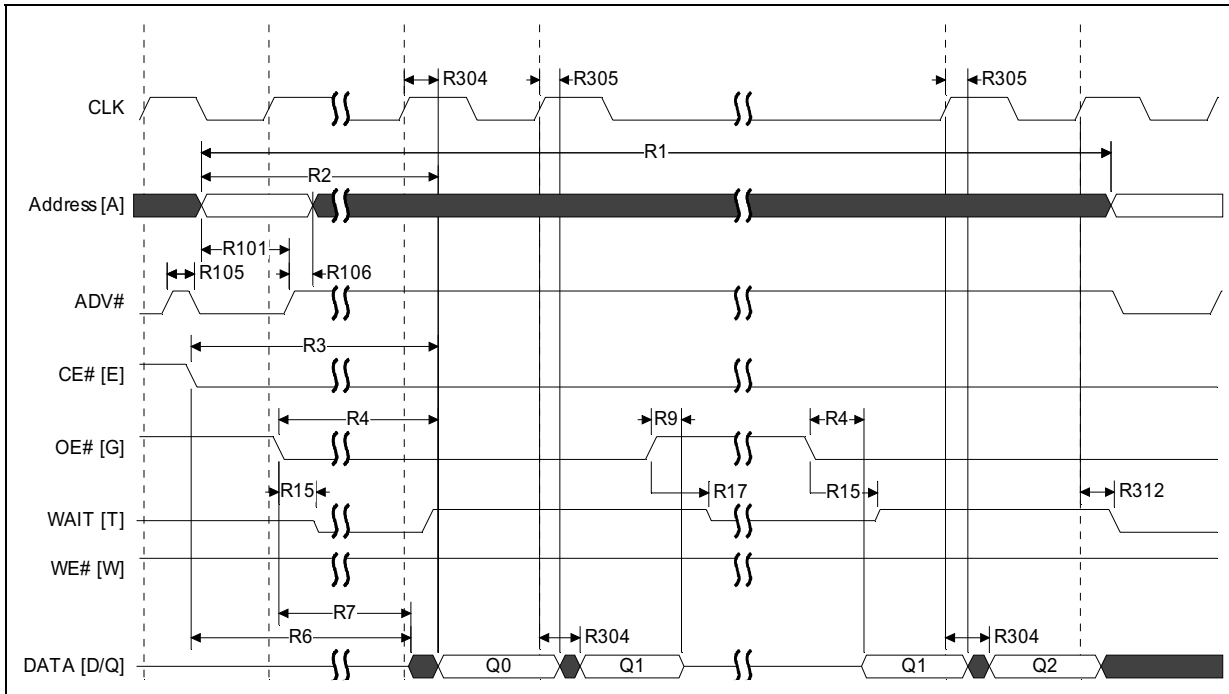
Figure 16. Synchronous Burst-Mode Four-Word Read Timing





**NOTE:** WAIT is driven per OE# assertion during synchronous array or non-array read. WAIT asserted during initial latency and deasserted during valid data (CR.10 = 0 Wait asserted low).

**Figure 17. Burst Suspend Timing**



- NOTES:**
1. CLK can be stopped in either high or low state.
  2. WAIT is driven per OE# assertion during synchronous array or non-array read. WAIT asserted during initial latency and deasserted during valid data (CR.10 = 0 Wait asserted low).

## 7.5 AC Write Specifications

Nbr.	Symbol	Parameter <sup>(1, 2)</sup>	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 <sub>PP</sub> setup to WE# high	200		ns	1,2,3,7
W11	$t_{QVVL}$	V <sub>PP</sub> 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
<b>Write to Asynchronous Read Specifications</b>						
W18	$t_{WHAV}$	WE# high to Address valid	0		ns	1,2,3,6
<b>Write to Synchronous Read Specifications</b>						
W19	$t_{WHCHL}$	WE# high to Clock valid	19		ns	1,2,3,6,10
W20	$t_{WHVH}$	WE# high to ADV# high	19		ns	
<b>Write Specifications with Clock Active</b>						
W21	$t_{VHWL}$	ADV# high to WE# low		20	ns	1,2,3,11
W22	$t_{CHWL}$	Clock high to WE# low		20	ns	

### NOTES:

- Write timing characteristics during erase suspend are the same as write-only operations.
- A write operation can be terminated with either CE# or WE#.
- Sampled, not 100% tested.
- 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,  $t_{WLWH} = t_{ELEH} = t_{WLEH} = t_{ELWH}$ .
- 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,  $t_{WHWL} = t_{EHEL} = t_{WHEL} = t_{EHWL}$ .
- $t_{WHVH}$  or  $t_{WHCHL}$  must be met when transitioning from a write cycle to a synchronous burst read.
- V<sub>PP</sub> and WP# should be at a valid level until erase or program success is determined.
- This specification is only applicable when transitioning from a write cycle to an asynchronous read. See spec W19 and W20 for synchronous read.
- When doing a Read Status operation following any command that alters the Status Register, W14 is 20 ns.
- Add 10ns if the write operations results in a RCR or block lock status change, for the subsequent read operation to reflect this change.
- These specs are required only when the device is in a synchronous mode and clock is active during address setup phase.

Figure 18. Write to Write Timing

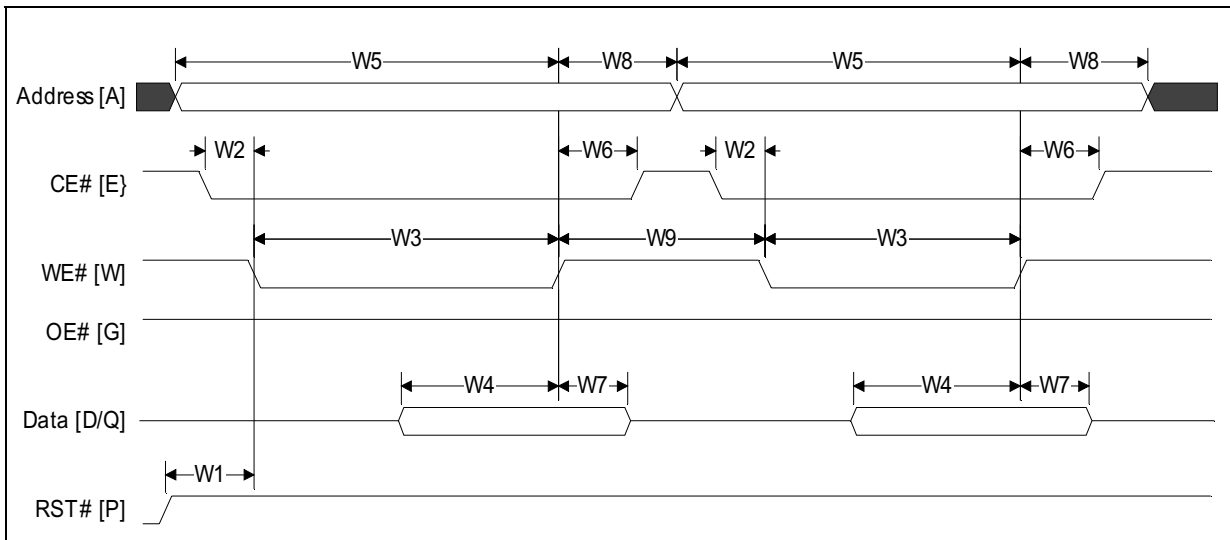
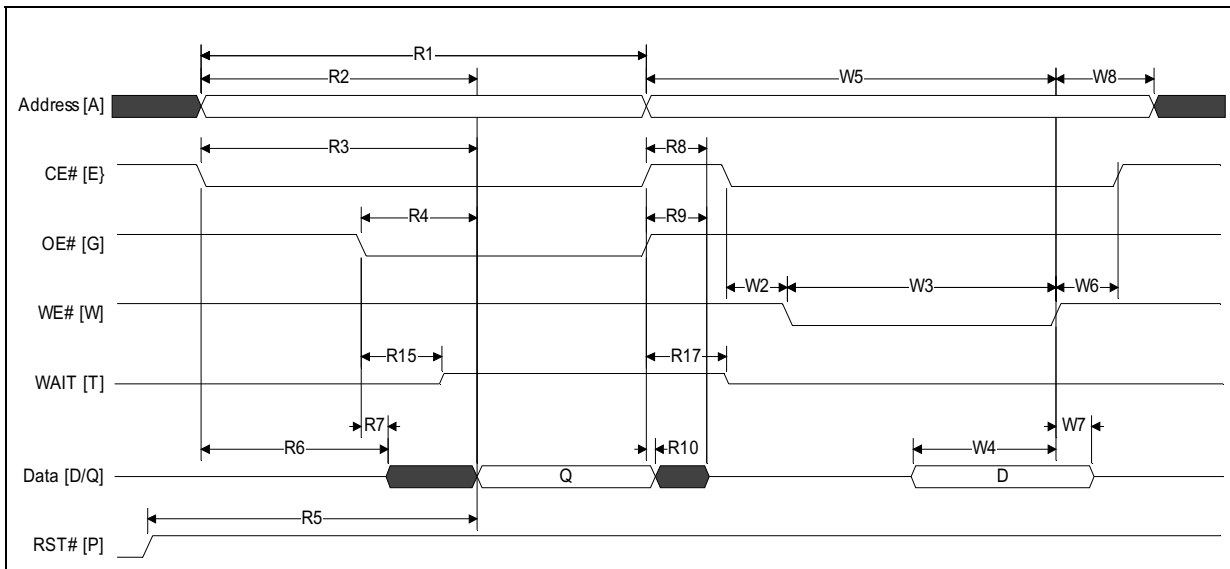


Figure 19. Asynchronous Read to Write Timing



**NOTE:** Wait deasserted during asynchronous read and during write. WAIT High-Z during write per OE# deasserted.

Figure 20. Write to Asynchronous Read Timing

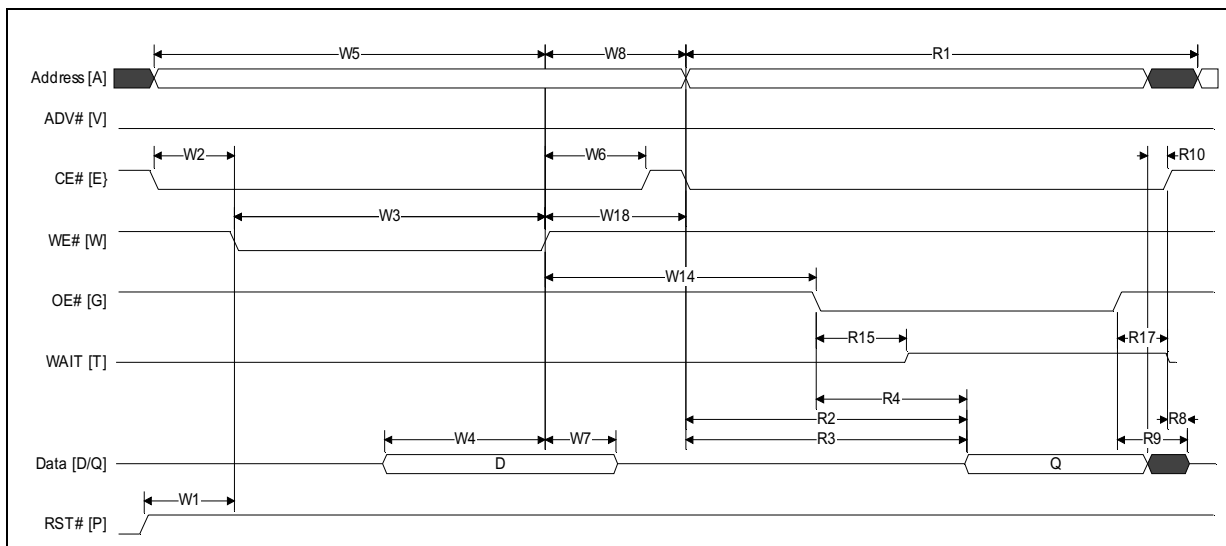
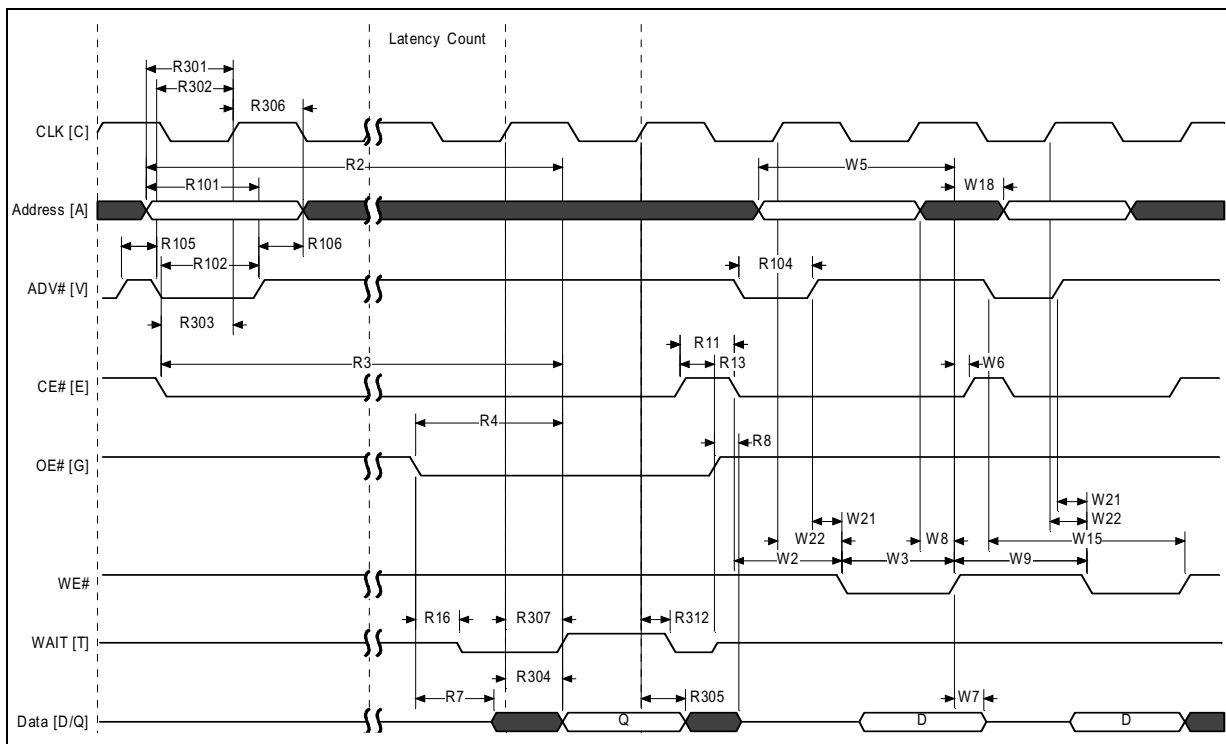
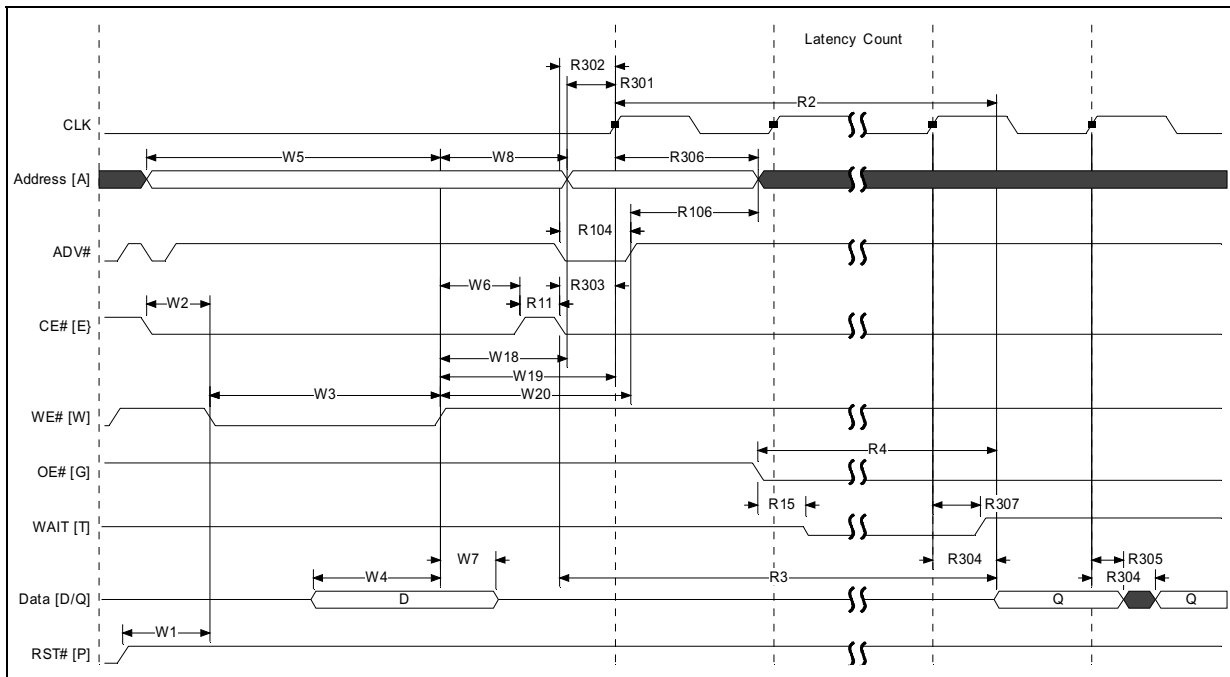


Figure 21. Synchronous Read to Write Timing



**NOTE:** WAIT shown deasserted and High-Z per OE# de-assertion during write operation (CR[10]=0 Wait asserted low). Clock is ignored during write operation.

**Figure 22. Write to Synchronous Read Timing**


**NOTE:** WAIT shown deasserted and High-Z per OE# de-assertion during write operation (CR[10]=0 Wait asserted low).

## 7.6 Program and Erase Characteristics

Nbr.	Symbol	Parameter	V <sub>PPL</sub>			V <sub>PPH</sub>			Units	Notes	
			Min	Typ	Max	Min	Typ	Max			
<b>Conventional Word Programming</b>											
W200	t <sub>PROG/W</sub>	Program Time	Single word		90	180	85	170	μs	1	
			Single cell		30	60	30	60			
<b>Buffered Programming</b>											
W200	t <sub>PROG/W</sub>	Program Time	Single word		90	180	85	170	μs	1	
W251	t <sub>BUFF</sub>		One Buffer (32 words)		440	880	340	680			
<b>Buffered Enhanced Factory Programming</b>											
W451	t <sub>BEFP/W</sub>	Program	Single word		N/A	N/A	N/A	10	N/A	μs	1,2
W452	t <sub>BEFP/Setup</sub>		Buffered EFP Setup		N/A	N/A	N/A	5	N/A		
<b>Erasing and Suspending</b>											
W500	t <sub>ERS/PB</sub>	Erase Time	16-KWord Parameter		0.4	2.5	0.4	2.5	s	1	
W501	t <sub>ERS/MB</sub>		64-KWord Main		1.2	4	1.0	4			
W600	t <sub>SUSP/P</sub>	Suspend	Program suspend		20	25	20	25	μs		
W601	t <sub>SUSP/E</sub>	Latency	Erase suspend		20	25	20	25			

**NOTES:**

1. Typical values measured at T<sub>C</sub> = +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.

## 8.0 Power and Reset Specifications

### 8.1 Power Up and Down

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 VCC should attain VCCMIN before applying VCCQ and VPP. Device inputs should not be driven before supply voltage equals VCCMIN.

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

### 8.2 Reset Specifications

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.

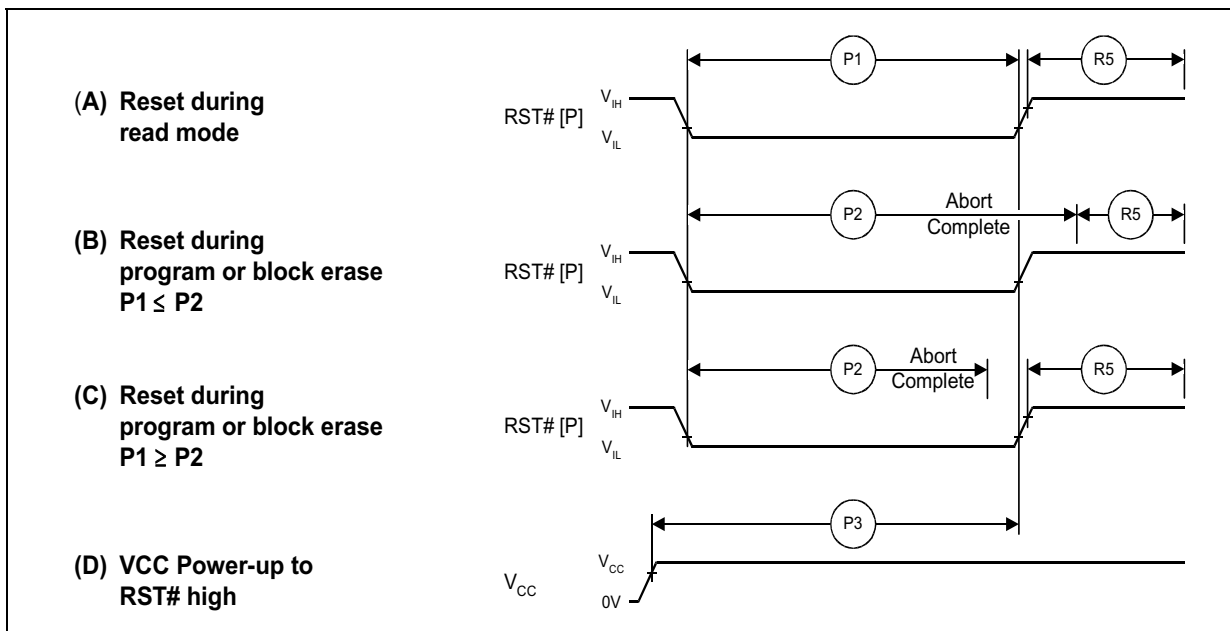
System designers should guard against spurious writes when VCC voltages are above VLKO. Because both WE# and CE# must be asserted for a write operation, deasserting either signal inhibits writes to the device.

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 9.2, “Device Commands” on page 41).

Nbr.	Symbol	Parameter	Min	Max	Unit	Notes
P1	t <sub>PLPH</sub>	RST# pulse width low	100		ns	1,2,3,4
P2	t <sub>PLRH</sub>	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 <sub>VCCPH</sub>	VCC 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<sub>PLPH</sub> is <t<sub>PLPH</sub> 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 VCC supply, device will not be ready until t<sub>VCCPH</sub> after VCC >= VCC min.
6. If RST# is tied to any supply/signal with VCCQ voltage levels, the RST# input voltage must not exceed VCC until VCC >= VCC(min).
7. Reset completes within t<sub>PLPH</sub> if RST# is asserted while no erase or program operation is executing.

**Figure 23. Reset Operation Waveforms**


### 8.3 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  $\mu\text{F}$  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  $\mu\text{F}$  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.

### 8.4 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  $\mu\text{s}$  after CE# is deasserted. During APS, average current is measured over the same time interval 5  $\mu\text{s}$  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.

## 9.0 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.

### 9.1 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).

#### 9.1.1 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 10.0, “Read Operations” on page 45](#) for details on the available read modes, and see [Section 15.0, “Special Read States” on page 71](#) 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 6.1, “DC Current Characteristics” on page 24](#)).

#### 9.1.2 Writes

To perform a write operation, both CE# and WE# are asserted while RST# and OE# are deasserted. During a write operation, address and data are latched on the rising edge of WE# or CE#, whichever occurs first. [Table 7, “Command Bus Cycles” on page 42](#) shows the bus cycle sequence for each of the supported device commands, while [Table 8, “Command Codes and Definitions” on page 43](#) describes each command. See [Section 7.0, “AC Characteristics” on page 26](#) 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.

#### 9.1.3 Output Disable

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



### 9.1.4 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  $\mu$ s after CE# is deasserted. During standby, average current is measured over the same time interval 5  $\mu$ s 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.

### 9.1.5 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. Flash memory devices from Intel 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 7.0, “AC Characteristics” on page 26](#) for details about signal-timing.

## 9.2 Device Commands

Device operations are initiated by writing specific device commands to the Command User Interface (CUI). See [Table 7, “Command Bus Cycles” on page 42](#).

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 7. Command Bus Cycles

Mode	Command	Bus Cycles	First Bus Cycle			Second Bus Cycle		
			Oper	Addr <sup>1</sup>	Data <sup>2</sup>	Oper	Addr <sup>1</sup>	Data <sup>2</sup>
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 <sup>3</sup>	> 2	Write	WA	0xE8	Write	WA	N - 1
	Buffered Enhanced Factory Program (Buffered EFP) <sup>4</sup>	> 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:

- 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.
- 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.
- 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.
- The confirm command (0xD0) is followed by the buffer data.

## 9.3 Command Definitions

Valid device command codes and descriptions are shown in [Table 8](#).

**Table 8. 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 in asynchronous read. CE# or ADV# must be toggled to update the Status Register Data for synchronous Non-array read. 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 <i>not</i> 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 in asynchronous read. CE# or ADV# must be toggled to update the Status Register Data for synchronous Non-array read.
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 8. Command Codes and Definitions (Sheet 2 of 2)

Mode	Code	Device Mode	Description
<b>Block Locking/Unlocking</b>	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.
<b>Protection</b>	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
<b>Configuration</b>	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.

## 10.0 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 10.3, “Read Configuration Register \(RCR\)”](#) on page 46).

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 9.2, “Device Commands”](#) on page 41). See [Section 15.0, “Special Read States”](#) on page 71 for details regarding Read Status, Read ID, and CFI Query modes.

The following sections describe read-mode operations in detail.

### 10.1 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.

**Note:** Asynchronous page-mode reads can only be performed when Read Configuration Register bit RCR[15] is set (see [Section 10.3, “Read Configuration Register \(RCR\)”](#) on page 46).

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 deasserted during asynchronous page mode. ADV# can be driven high to latch the address, or it must 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<sub>IH</sub> level, WAIT signal can be floated and ADV# must be tied to ground. Array data is driven onto D[15:0] after an initial access time t<sub>AVQV</sub> delay. (see [Section 7.0, “AC Characteristics”](#) on page 26).

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 page. Address bits A[1:0] determine which word of the 4-word page is output from the data buffer at any given time.

### 10.2 Synchronous Burst-Mode Read

Read Configuration register bits CR[15:0] must be set before synchronous burst operation can be performed. Synchronous burst mode can be performed for both array and non-array reads such as Read ID, Read Status or Read Query. (See [Section 10.3, “Read Configuration Register \(RCR\)”](#) on page 46 for details). Synchronous burst mode outputs 4-, 8-, 16-, or continuous-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.

During synchronous array and non-array read modes, the first word is output from the data buffer on the next valid CLK edge after the initial access latency delay (see [Section 10.3.2, “Latency Count” on page 48](#)). Subsequent data is output on valid CLK edges following a minimum delay. However, for a synchronous non-array read, the same word of data will be output on successive clock edges until the burst length requirements are satisfied.

During synchronous read operations, WAIT is driven with respect to OE# assertion. WAIT indicates invalid data when asserted, and valid data when deasserted with respect to a valid clock edge. See [Figure 14](#) through [Figure 16](#) for additional details.

### 10.2.1 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  $V_{IL}$ . WAIT is in High-Z during OE# de-assertion.

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

Within the device, CE# and OE# gate WAIT. Therefore, during Burst Suspend WAIT is placed in high-impedance state when OE# is deasserted and resumed active when OE# is re-asserted. See [Figure 17, “Burst Suspend Timing” on page 33](#).

## 10.3 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 9.2, “Device Commands” on page 41](#)).

RCR contents can be examined using the Read Device Identifier command, and then reading from <partition base address> + 0x05 (see [Section 15.2, “Read Device Identifier” on page 72](#)).

The RCR is shown in Table 9. The following sections describe each RCR bit.

**Table 9. 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 (default) (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 deasserted with valid data 1 =WAIT deasserted 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 deasserted with valid data (WD = 0). Latency Code 2, Data Hold for a 2-cock data cycle (DH=1) Wait deasserted one data cycle before valid data (WD = 1) combination is not supported.

### 10.3.1 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.

### 10.3.2 Latency Count

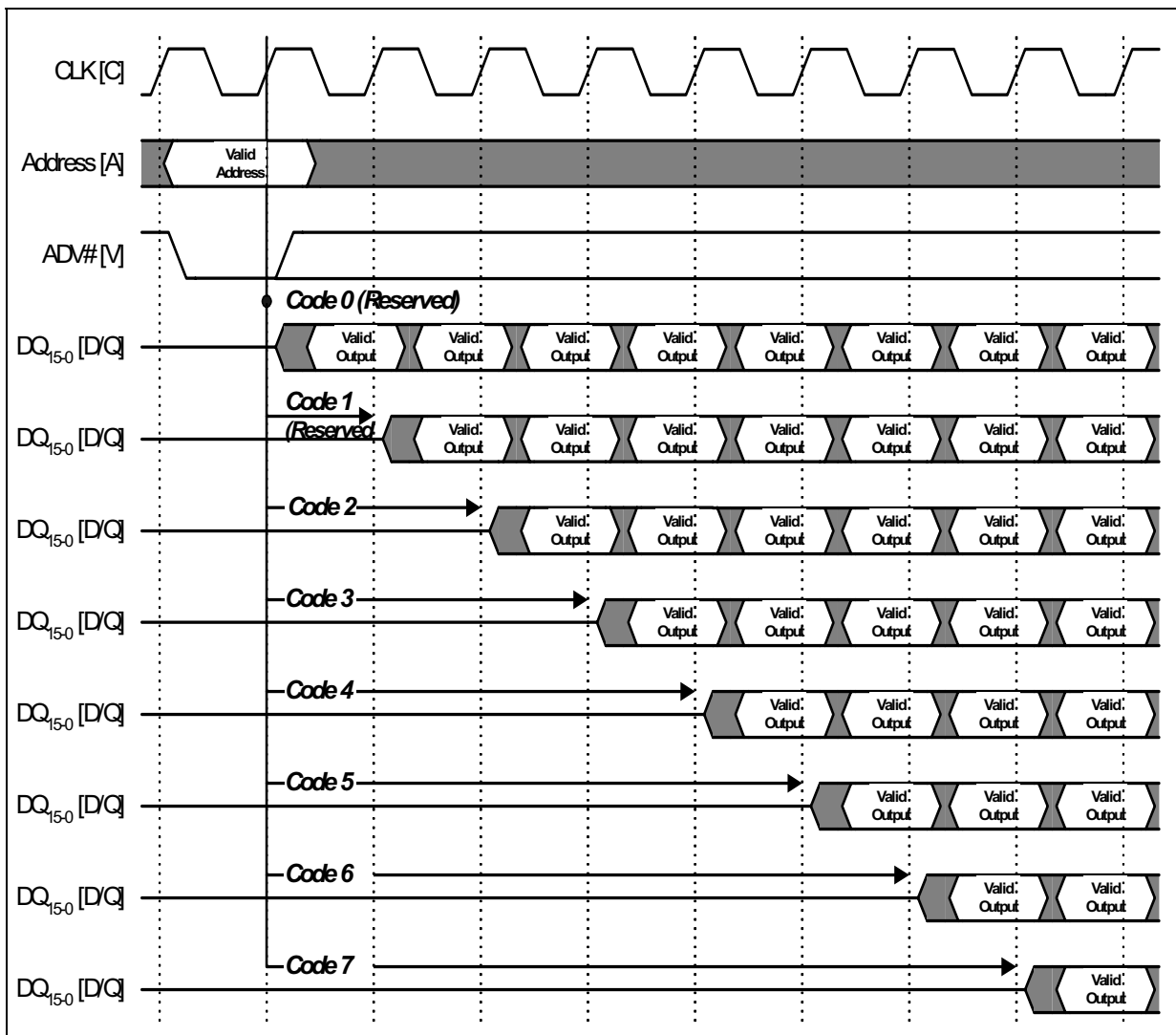
The Latency Count bits, LC[2:0], tell the device how many clock cycles must elapse from the rising edge of ADV# (or from the first valid clock edge after ADV# is asserted) until the first data word is to be driven onto D[15:0]. The input clock frequency is used to determine this value.

[Figure 24](#) shows the data output latency for the different settings of LC[2:0].

Synchronous burst with a Latency Count setting of Code 4 will result in zero WAIT state; however, 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 10](#), “LC and Frequency Support for Bin 1 tAVQV/tCHQV (85ns / 17ns)” on page 49 and [Table 11](#), “LC and Frequency Support for Bin 2 tAVQV/tCHQV (110ns / 20ns)” on page 50 for Latency Code Settings.



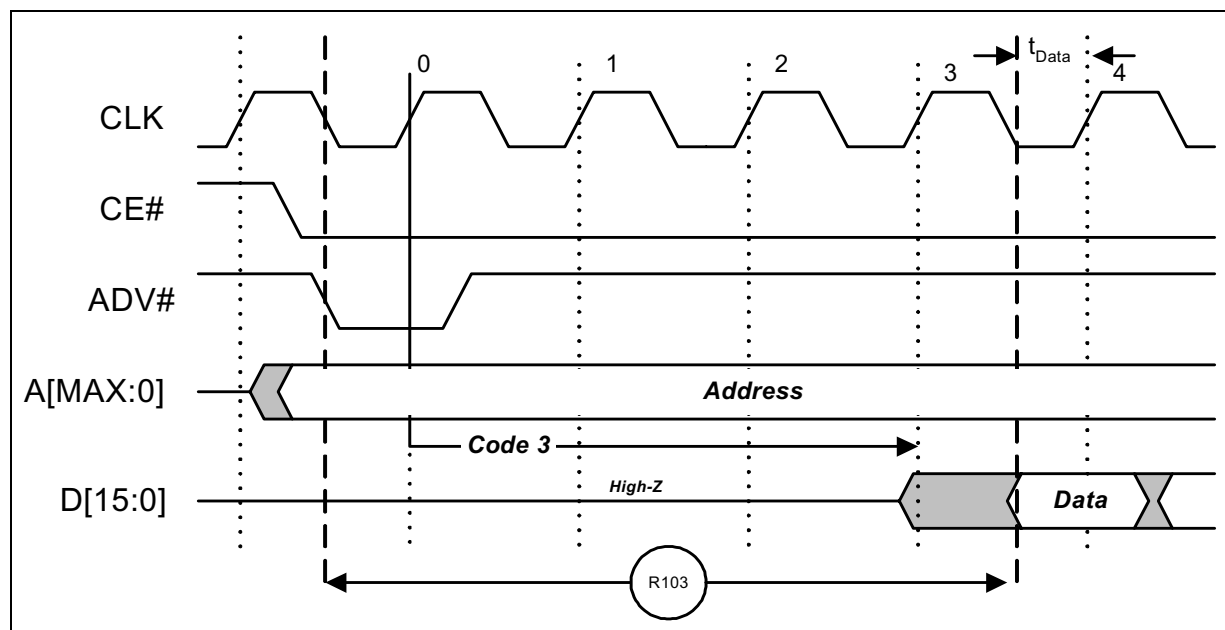
**Figure 24. First-Access Latency Count**

**Table 10. LC and Frequency Support for Bin 1  $t_{AVQV}/t_{CHQV}$  (85ns / 17ns)**

Latency Count Settings	Frequency Support (MHz)
2	$\leq 27$
3	$\leq 40$
4, 5, 6, or 7	$\leq 52$

**Table 11. LC and Frequency Support for Bin 2  $t_{AVQV}/t_{CHQV}$  (110ns / 20ns)**

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

See Figure 25, “Example Latency Count Setting using Code 3.

**Figure 25. Example Latency Count Setting using Code 3**

### 10.3.3 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, OE# asserted, RST# deasserted).

#### 10.3.3.1 WAIT Signal Function

The WAIT signal indicates data valid when the device is operating in synchronous mode (CR[15]=0). The WAIT signal is only “deasserted” when data is valid on the bus.

When the device is operating in synchronous non-array read mode, such as read status, read ID, or read query. The WAIT signal is also “deasserted” when data is valid on the bus.

WAIT behavior during synchronous non-array reads at the end of word line works correctly only on the first data access.

When the device is operating in asynchronous page mode, asynchronous single word read mode, and all write operations, WAIT is set to a deasserted state as determined by CR[10]. See [Figure 12, “Asynchronous Single-Word Read \(ADV# Latch\)”](#) on page 30, and [Figure 13, “Asynchronous Page-Mode Read Timing”](#) on page 31.

**Table 12. WAIT Functionality Table**

Condition	WAIT	Notes
CE# = '1', OE# = 'X' CE# = 'X', OE# = '1'	High-Z	1
OE# = 'X', OE# = '0'	Active	1
Synchronous Array Reads	Active	1
Synchronous Non-Array Reads	Active	1
All Asynchronous Reads	deasserted	1
All Writes	High-Z	1,2

**NOTES:**

1. **Active:** WAIT is asserted until data becomes valid, then de-asserts
2. When OE# = V<sub>IH</sub> during writes, WAIT = High-Z

### 10.3.4 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 26](#)). 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:

$$t_{\text{CHQV}} (\text{ns}) + t_{\text{DATA}} (\text{ns}) \leq \text{One CLK Period (ns)}$$

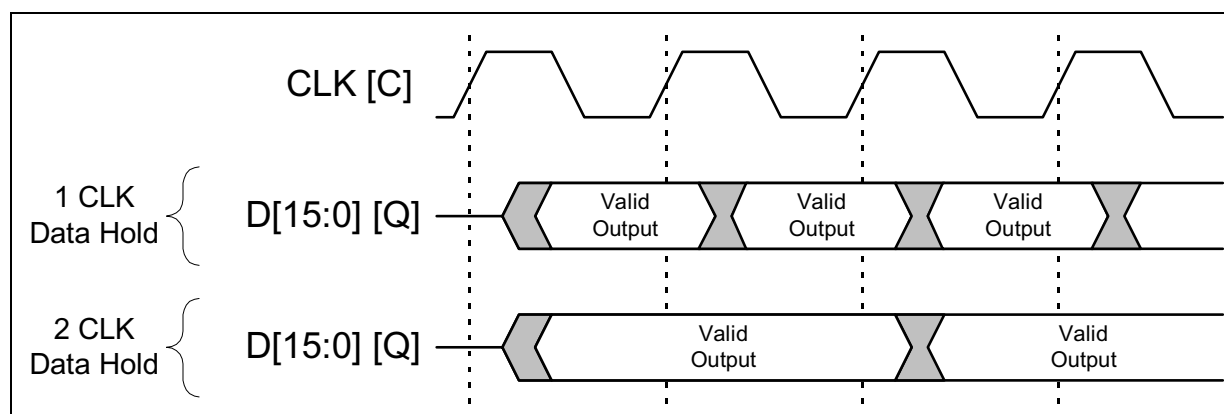
$t_{\text{DATA}}$  = Data set up to Clock (defined by CPU)

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

$$20 \text{ ns} + 4 \text{ ns} \leq 25 \text{ ns}$$

The equation is satisfied and data will be available at every clock period with data hold setting at one clock. If  $t_{\text{CHQV}} (\text{ns}) + t_{\text{DATA}} (\text{ns}) > \text{One CLK Period (ns)}$ , data hold setting of 2 clock periods must be used.

Figure 26. Data Hold Timing



### 10.3.5 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 deasserted one data cycle *before* valid data (default). When WD is cleared, WAIT is deasserted *during* valid data.

### 10.3.6 Burst Sequence

The Burst Sequence (BS) bit selects linear-burst sequence (default). Only linear-burst sequence is supported. Table 13 shows the synchronous burst sequence for all burst lengths, as well as the effect of the Burst Wrap (BW) setting.

Table 13. Burst Sequence Word Ordering

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-...
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-...

### 10.3.7 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.

### 10.3.8 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.

### 10.3.9 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 13, "Burst Sequence Word Ordering" on page 52](#)). When a burst cycle begins, the device outputs synchronous burst data until it reaches the end of the "burstable" address space.

## 11.0 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 9.0, “Device Operations” on page 40](#) 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 must be 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 13.0, “Security Modes” on page 62](#) for details on locking and unlocking blocks.

The following sections describe device programming in detail.

### 11.1 Word Programming

Word programming operations are initiated by writing the Word Program Setup command to the device (see [Section 9.0, “Device Operations” on page 40](#)). 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 39, “Word Program Flowchart” on page 81](#).

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 12.0, “Erase Operations” on page 60](#)).

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).

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.

### 11.1.1 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 5.2, “Operating Conditions” on page 23](#) 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 27, “Example VPP Supply Connections” on page 59](#) shows examples of device power supply configurations.

## 11.2 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 9.2, “Device Commands” on page 41](#)), 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 41, “Buffer Program Flowchart” on page 83](#)).

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 ( $A[4:0] = 0x00$ ). Crossing a 32-word boundary during programming will double the total programming time.

After the last data is written to the buffer, the Buffered Programming Confirm command must be issued to the original block address. 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 [Figure 41, “Buffer Program Flowchart” on page 83](#)).

When Buffered Programming has completed, 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 5.2, “Operating Conditions” on page 23](#) for limitations when operating the device with  $V_{PP} = V_{PPH}$ ).

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.

## 11.3 Buffered Enhanced Factory Programming

Buffered Enhanced Factory Programming (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 42, “Buffered EFP Flowchart” on page 84](#)). 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.

### 11.3.1 Buffered EFP Requirements and Considerations

Buffered EFP requirements:

- Ambient temperature:  $T_A = 25^{\circ}\text{C}, \pm 5^{\circ}\text{C}$
- $V_{CC}$  within specified operating range.
- $V_{PP}$  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<sup>1</sup>.

Buffered EFP considerations:

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

### 11.3.2 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.

### 11.3.3 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. After the buffer fill cycle, no write cycles should be issued to the device until SR.0 = 0 and the device is ready for the next buffer fill.

**Note:** Any spurious writes are ignored after a buffer fill operation and when internal program is proceeding.

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.

### 11.3.4 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.

## 11.4 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 40, “Program Suspend/Resume Flowchart” on page 82](#)).

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 7.6, “Program and Erase Characteristics” on page 37](#).

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, 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<sub>PP</sub> must remain at its programming level, and WP# must remain unchanged while in program suspend. If RST# is asserted, the device is reset.

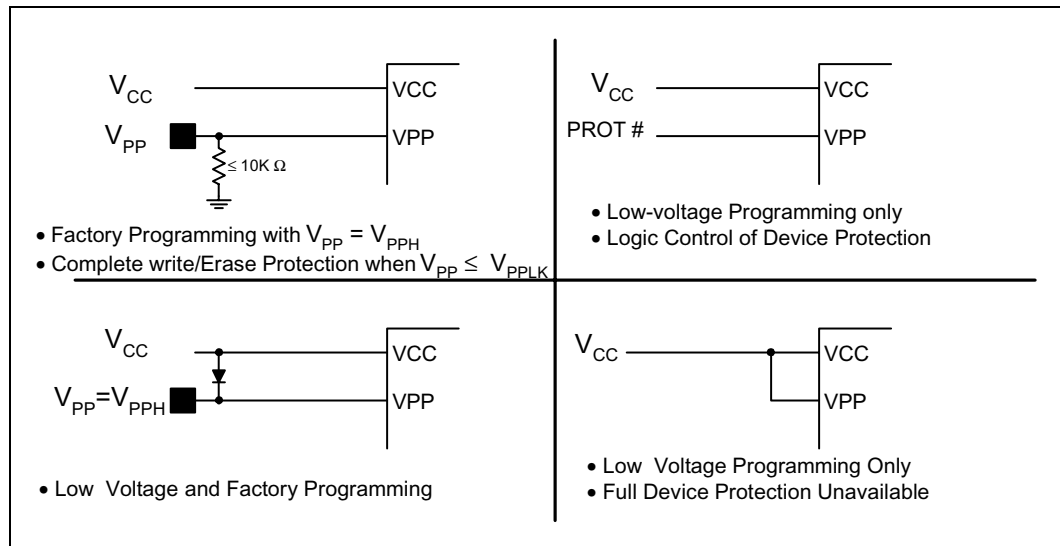
## 11.5 Program Resume

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 40, “Program Suspend/Resume Flowchart” on page 82](#)).

## 11.6 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 27. Example VPP Supply Connections**



## 12.0 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.

### 12.1 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 9.2, “Device Commands” on page 41](#)). 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 43, “Block Erase Flowchart” on page 85](#)).

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 11.0, “Programming Operations” on page 54](#)).

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.

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.

### 12.2 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 buffer program operation, or a read operation within any block except the block that is erase suspended (see [Figure 40, “Program Suspend/Resume Flowchart” on page 82](#)).

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 7.6, “Program and Erase Characteristics” on page 37](#).

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.

## 12.3 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 40, “Program Suspend/Resume Flowchart” on page 82](#)).

## 12.4 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.

## 13.0 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.

### 13.1 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<sub>PP</sub> data security can be used to inhibit program and erase operations (see Section 11.6, “Program Protection” on page 59 and Section 12.4, “Erase Protection” on page 61).

#### 13.1.1 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 9.2, “Device Commands” on page 41 and Figure 45, “Block Lock Operations Flowchart” on page 87). 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<sub>PP</sub>. The block lock bits may be modified and/or read even if V<sub>PP</sub> is below V<sub>PPLK</sub>.

#### 13.1.2 Unlock Block

The Unlock Block command is used to unlock blocks (see Section 9.2, “Device Commands” on page 41). 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 28, “Block Locking State Diagram” on page 63).

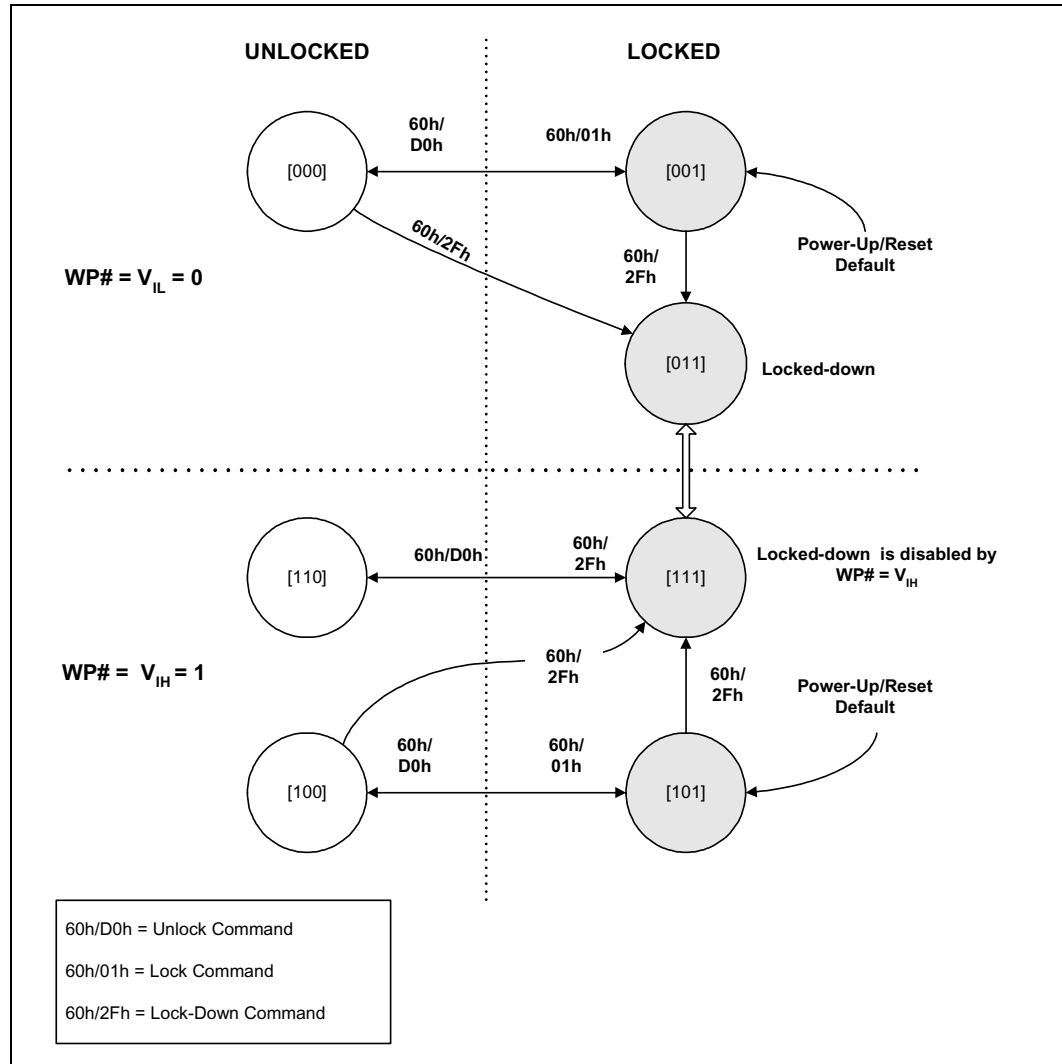
#### 13.1.3 Lock-Down Block

A locked or unlocked block can be locked-down by writing the Lock-Down Block command sequence (see Section 9.2, “Device Commands” on page 41). 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-Down command must be issued prior to changing WP# to V<sub>IL</sub>. Locked-down blocks revert to the locked state upon reset or power up the device (see Figure 28, “Block Locking State Diagram” on page 63).

### 13.1.4 Block Lock Status

The Read Device Identifier command is used to determine a block’s lock status (see Section 15.2, “Read Device Identifier” on page 72). 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 28. Block Locking State Diagram



### 13.1.5 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 74](#), which shows valid commands during an erase suspend.

## 13.2 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.

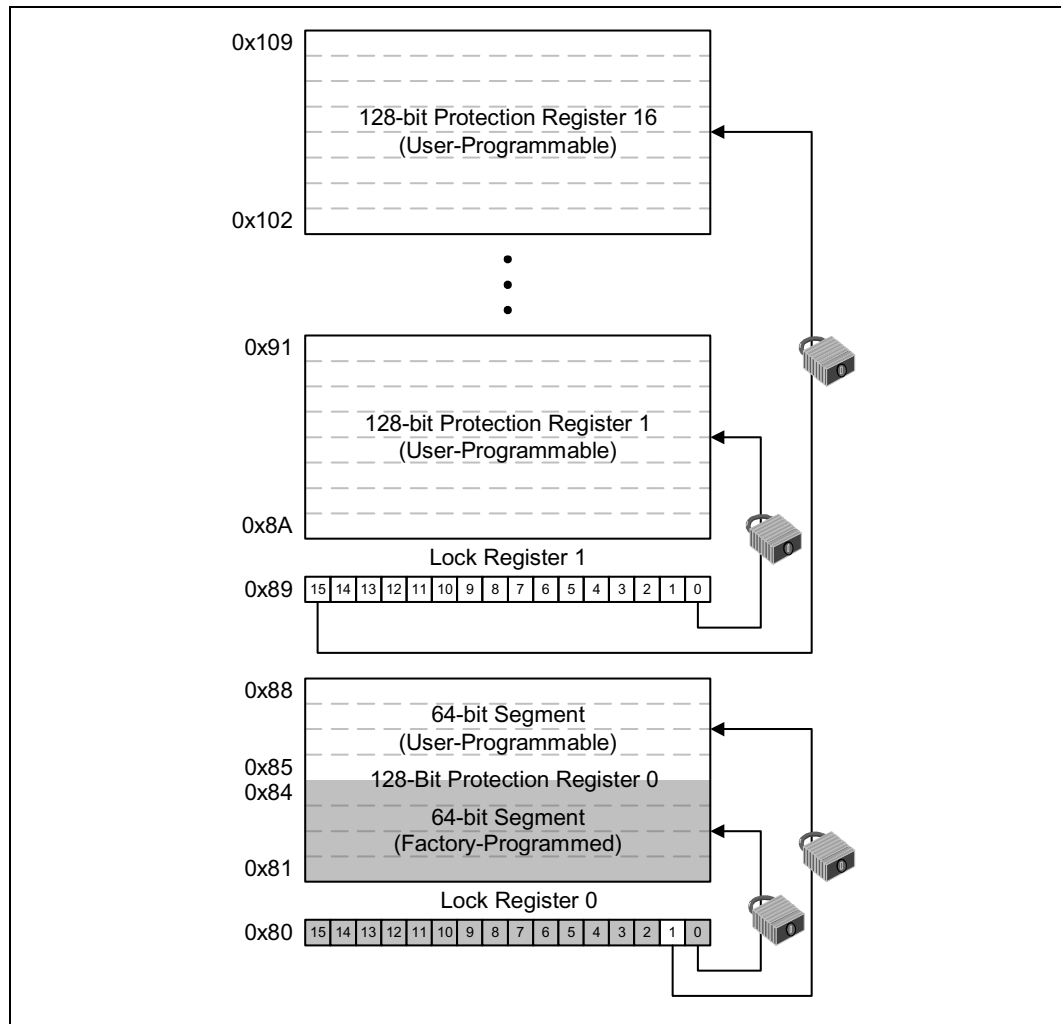
The first 128-bit Protection Register is comprised of two 64-bit (8-word) segments. The lower 64-bit segment is pre-programmed at the factory with a unique 64-bit number. The other 64-bit segment, as well as the other sixteen 128-bit Protection Registers, are blank. Users can program these registers as needed. When programmed, users can then lock the Protection Register(s) to prevent additional bit programming (see [Figure 29, “Protection Register Map” on page 65](#)).

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 29. Protection Register Map



### 13.2.1 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 9.2, “Device Commands” on page 41](#)). Next, perform a read operation at that partition’s base address plus the address offset corresponding to the register to be read. [Table 16, “Device Identifier Information” on page 73](#) 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 14, “Simultaneous Operation Restrictions” on page 70](#) for details.

### 13.2.2 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 9.2, "Device Commands" on page 41](#)). Next, write the desired Protection Register data to the same Protection Register address (see [Figure 29, "Protection Register Map" on page 65](#)).

The device programs the 64-bit and 128-bit user-programmable Protection Register data 16 bits at a time (see [Figure 46, "Protection Register Programming Flowchart" on page 88](#)). 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 14, "Simultaneous Operation Restrictions" on page 70](#) for details.

### 13.2.3 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 9.2, "Device Commands" on page 41](#)). 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 16, "Device Identifier Information" on page 73](#)).

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.

## 14.0 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.

### 14.1 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.

### 14.2 Read-While-Write Command Sequences

When issuing commands to the device, a read operation can occur between 2-cycle Write command's (Figure 30, and Figure 31). However, a write operation issued between a 2-cycle commands write sequence causes a command sequence error. (See Figure 32)

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 30. Operating Mode with Correct Command Sequence Example**

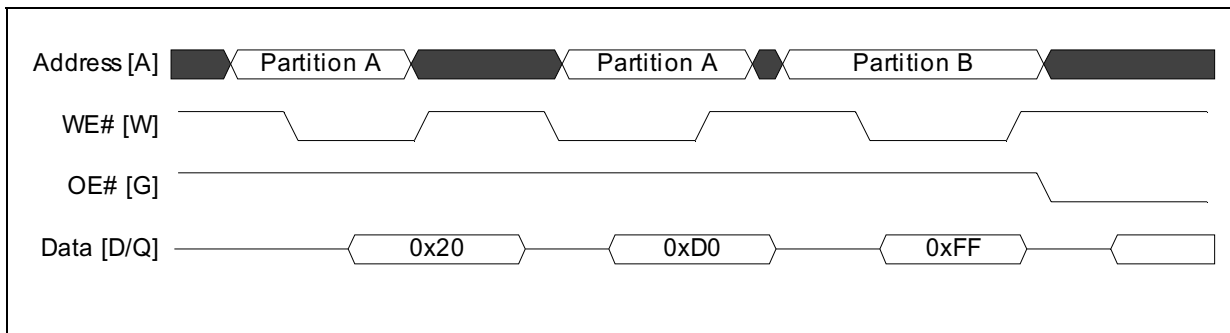


Figure 31. Operating Mode with Correct Command Sequence Example

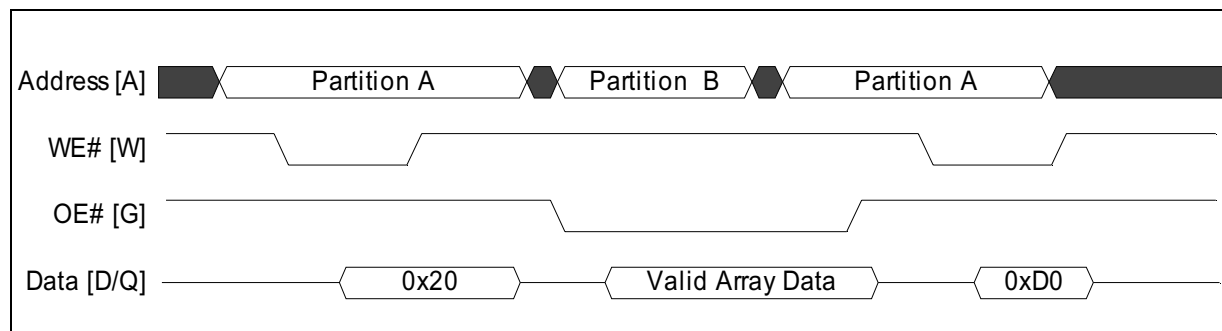
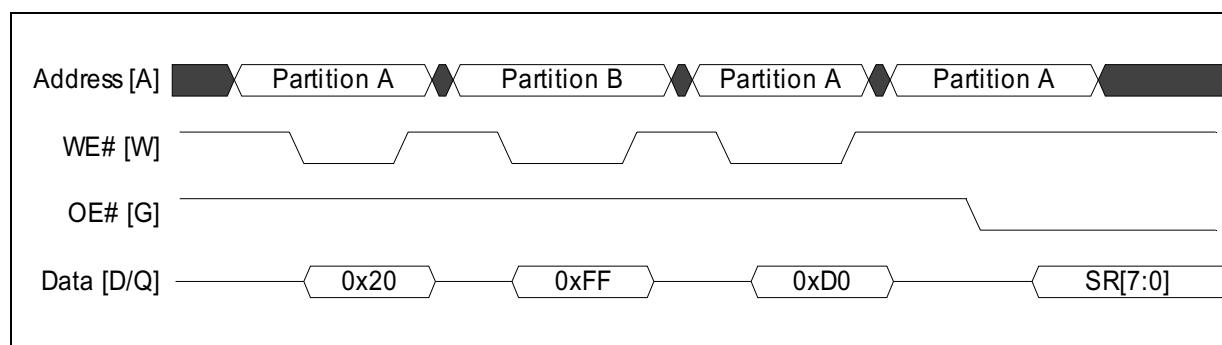


Figure 32. Operating Mode with Illegal Command Sequence Example



### 14.2.1 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.

### 14.2.2 Synchronous and Asynchronous Read-While-Write Characteristics and Waveforms

This section describes the transitions of write operation to asynchronous read, write to synchronous read, and write operation with clock active.

#### 14.2.2.1 Write operation to asynchronous read transition

##### 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).

### 14.2.2.2 Write to synchronous read operation transition

#### 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.

### 14.2.2.3 Write Operation with Clock Active

#### W21 - $t_{VHWL}$ W22 - $t_{CHWL}$

The AC parameters W21 ( $t_{VHWL}$ - ADV# High to WE# Low) and W22 ( $t_{CHWL}$  -Clock high to WE# low) are required when the device is in a synchronous mode and clock is active. A write bus cycle consists of two parts:

- the host provides an address to the flash device; and
- the host then provides data to the flash device.

The flash device in turn binds the received data with the received address. When operating synchronously (RCR.15 = 0), the address of a write cycle may be provided to the flash by the first active clock edge with ADV# low, or rising edge of ADV# as long as the applicable cycle separation conditions are met between each cycle.

If neither a clock edge nor a rising ADV# edge is used to provide a new address at the beginning of a write cycle (the clock is stopped and ADV# is low), the address may also be provided to the flash device by holding the address bus stable for the required amount of time (W5,  $t_{AVWH}$ ) before the rising WE# edge.

Alternatively, the host may choose not to provide an address to the flash device during subsequent write cycles (if ADV# is high and only CE# or WE# is toggled to separate the prior cycle from the current write cycle). In this case, the flash device will use the most recently provided address from the host.

Refer to [Figure 20, “Write to Asynchronous Read Timing” on page 36](#), [Figure 21, “Synchronous Read to Write Timing” on page 36](#), and [Figure 22, “Write to Synchronous Read Timing” on page 37](#), for representation of these timings.

### 14.2.3 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.

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 41, “Buffer Program Flowchart” on page 83](#) for more details.

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

## 14.3 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 14](#) provides details on restrictions during simultaneous operations.)

**Table 14. 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 <i>different</i> from the one being programmed/erased, and also <i>different</i> 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 <i>any</i> partition. Reads in partitions other than the parameter partition are supported.

## 15.0 Special Read States

The following sections describe non-array read states. Non-array reads can be performed in asynchronous read or synchronous burst mode. A non-array read operation occurs as asynchronous single-word mode. When non-array reads are performed in asynchronous page mode only the first data is valid and all subsequent data are undefined. When a non-array read operation occurs as synchronous burst mode, the same word of data requested will be output on successive clock edges until the burst length requirements are satisfied.

Each partition can be in one of its read states independent of other partitions' modes. See [Figure 11](#), "Asynchronous Single-Word Read (ADV# Low)" on page 30, [Figure 12](#), "Asynchronous Single-Word Read (ADV# Latch)" on page 30, and [Figure 14](#), "Synchronous Single-Word Array or Non-array Read Timing" on page 31 for details.

### 15.1 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).

The Status Register is read using single asynchronous-mode or synchronous burst mode reads. Status Register data is output on D[7:0], while 0x00 is output on D[15:8]. In asynchronous mode the falling edge of OE#, or CE# (whichever occurs first) updates and latches the Status Register contents. However, reading the Status Register in synchronous burst mode, CE# or ADV# must be toggled to update status data. The 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<sub>PP</sub>, and block-locked operations.

**Table 15. Status Register Description (Sheet 1 of 2)**

Status Register (SR)								Default Value = 0x80
Device Write Status	Erase Suspend Status	Erase Status	Program Status	V <sub>PP</sub> 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.					

Table 15. Status Register Description (Sheet 2 of 2)

Status Register (SR)		Default Value = 0x80
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 <sub>PP</sub> Status (VPPS)	0 = VPP within acceptable limits during program or erase operation. 1 = VPP < VPPLK during program or erase operation.
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)	<p><b>DWS PWS</b></p> <p>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.</p> <p>(Non-buffered EFP operation. For Buffered EFP operation, see <a href="#">Section 11.3, "Buffered Enhanced Factory Programming" on page 56</a>).</p>

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.

### 15.1.1 Clear Status Register

The Clear Status Register command clears the status register, leaving all partition read states unchanged. It functions independent of V<sub>PP</sub>. 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.

## 15.2 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 9.2, "Device Commands" on page 41](#) for details on issuing the Read Device Identifier command). [Table 16, "Device Identifier Information" on page 73](#) and [Table 17, "Device ID codes" on page 73](#) 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 16. Device Identifier Information**

Item	Address <sup>(1,2)</sup>	Data
Manufacturer Code	PBA + 0x00	0089h
Device ID Code	PBA + 0x01	ID (see <a href="#">Table 17</a> )
Block Lock Configuration: <ul style="list-style-type: none"> <li>• Block Is Unlocked</li> <li>• Block Is Locked</li> <li>• Block Is not Locked-Down</li> <li>• Block Is Locked-Down</li> </ul>	BBA + 0x02	Lock Bit:  DQ <sub>0</sub> = 0b0 DQ <sub>0</sub> = 0b1 DQ <sub>1</sub> = 0b0 DQ <sub>1</sub> = 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 17. 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

## 15.3 CFI Query

The CFI Query command instructs the device to output Common Flash Interface (CFI) data when partition addresses are read. See [Section 9.2, “Device Commands” on page 41](#) for details on issuing the CFI Query command. [Appendix C, “Common Flash Interface” on page 89](#) 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 14, “Simultaneous Operation Restrictions” on page 70](#).

## Appendix A Write State Machine (WSM)

Figure 33 through Figure 38 show 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 33. Write State Machine—Next State Table (Sheet 1 of 6)

Command Input to Chip and resulting <i>Chip</i> Next State												
Current Chip State <sup>(7)</sup>	Read Array <sup>(2)</sup> (FFH)	Word Program <sup>(3,4)</sup> (10H/40H) Program Setup	Buffered Program (BP) (E8H) BP Setup	Erase Setup <sup>(3,4)</sup> (20H) Erase Setup	Buffered Enhanced Factory Pgm Setup <sup>(3,4)</sup> (80H) BEFP Setup	BE Confirm, P/E Resume, ULB, Confirm <sup>(8)</sup> (D0H)	BP / Prg / Erase Suspend (B0H)	Read Status (70H)	Clear Status Register <sup>(5)</sup> (50H)	Read ID/Query (90H, 98H)	Lock, Unlock, Lock-down, CR setup <sup>(4)</sup> (60H) Lock/CR Setup	
Ready	Ready	Program Setup	BP Setup	Erase Setup	BEFP Setup	Ready					Lock/CR Setup	
Lock/CR Setup	Ready (Lock Error)					Ready (Unlock Block)	Ready (Lock Error)					
OTP	Setup	OTP Busy										
	Busy	OTP 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										
	BP Load 1	BP Load 2										
	BP Load 2	BP Confirm if Data load into Program Buffer is complete; Else BP Load 2										
	BP Confirm	Ready (Error)					BP Busy	Ready (Error)				
	BP Busy	BP Busy					BP Suspend	BP Busy				
	BP Suspend	BP Suspend					BP Busy	BP Suspend				
Erase	Setup	Ready (Error)					Erase Busy	Ready (Error)				
	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

Figure 34. Write State Machine—Next State Table (Sheet 2 of 6)

		Command Input to Chip and resulting <i>Chip</i> Next State										
Current Chip State <sup>(7)</sup>		Read Array <sup>(2)</sup>	Word Program <sup>(3,4)</sup>	Buffered Program (BP)	Erase Setup <sup>(3,4)</sup>	Buffered Enhanced Factory Pgm Setup <sup>(3,4)</sup>	BE Confirm, P/E Resume, ULB, Confirm <sup>(8)</sup>	BP / Prg / Erase Suspend	Read Status	Clear Status Register <sup>(5)</sup>	Read ID/Query	Lock, Unlock, Lock-down, CR setup <sup>(4)</sup>
		(FFH)	(10H/40H)	(E8H)	(20H)	(80H)	(D0H)	(B0H)	(70H)	(50H)	(90H, 98H)	(60H)
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			
BP in Erase Suspend	Setup	BP Load 1										
	BP Load 1	BP Load 2										
	BP Load 2	BP Confirm if Data load into Program Buffer is complete; Else BP Load 2										
	BP Confirm	Erase Suspend (Error)						BP Busy in Erase Suspend	Ready (Error 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)						Erase Suspend (Unlock Block)	Erase Suspend (Lock Error [Botch])			
Buffered Enhanced Factory Program Mode	Setup	Ready (Error)						BEFP Loading Data (X=32)	Ready (Error)			
	BEFP Busy	BEFP Program and Verify Busy (if Block Address given matches address given on BEFP Setup command). Commands treated as data. <sup>(7)</sup>										

Figure 35. Write State Machine—Next State Table (Sheet 3 of 6)

Current Chip State <sup>(7)</sup>		Command Input to Chip and resulting <i>Chip</i> Next State							
		OTP Setup <sup>(4)</sup>  (C0H)	Lock Block Confirm <sup>(6)</sup>  (01H)	Lock-Down Block Confirm <sup>(6)</sup>  (2FH)	Write RCR Confirm <sup>(6)</sup>  (03H)	Block Address (?WA0) <sup>9</sup>  (XXXXH)	Illegal Cmds or BEFP Data <sup>(1)</sup>  (all other codes)	WSM Operation Completes	
Ready		OTP Setup	Ready					N/A	
Lock/CR Setup		Ready (Lock Error)	Ready (Lock Block)	Ready (Lock Down Blk)	Ready (Set CR)	Ready (Lock Error)		N/A	
OTP	Setup	OTP Busy						Ready	
	Busy	Word Program Busy						N/A	
Word Program	Setup	Word Program Busy						Ready	
	Busy	Word Program Busy						Ready	
	Suspend	Word Program Suspend						N/A	
BP	Setup	BP Load 1							N/A
	BP Load 1	BP Load 2			Ready (BP Load 2)	BP Load 2		N/A	
	BP Load 2	BP Confirm if Data load into Program Buffer is complete; ELSE BP load 2			Ready	BP Confirm if Data load into Program Buffer is complete; ELSE BP Load 2		N/A	
	BP Confirm	Ready (Error)			Ready (Error) (Proceed if unlocked or lock error)	Ready (Error)		N/A	
	BP Busy	BP Busy						Ready	
	BP Suspend	BP Suspend						N/A	
Erase	Setup	Ready (Error)						N/A	
	Busy	Erase Busy						Ready	
	Suspend	Erase Suspend						N/A	

Figure 36. Write State Machine—Next State Table (Sheet 4 of 6)

Current Chip State <sup>(7)</sup>		Command Input to Chip and resulting <i>Chip</i> Next State						
		OTP Setup <sup>(4)</sup> (C0H)	Lock Block Confirm <sup>(8)</sup> (01H)	Lock-Down Block Confirm <sup>(8)</sup> (2FH)	Write RCR Confirm <sup>(8)</sup> (03H)	Block Address (?WAO) <sup>9</sup> (XXXXH)	Illegal Cmds or BEFP Data <sup>(1)</sup> (all other codes)	WSM Operation Completes
Word Program in Erase Suspend	Setup	Word Program Busy in Erase Suspend						NA
	Busy	Word Program Busy in Erase Suspend Busy						Erase Suspend
	Suspend	Word Program Suspend in Erase Suspend						N/A
BP in Erase Suspend	Setup	BP Load 1						N/A
	BP Load 1	BP Load 2			Ready (BP Load 2)	BP Load 2		
	BP Load 2	BP Confirm if Data load into Program Buffer is complete; Else BP Load 2			Ready	BP Confirm if Data load into Program Buffer is complete; Else BP Load 2		
	BP Confirm	Ready (Error in Erase Suspend)			Ready (Error) (Proceed if unlocked or lock error)	Ready (Error)		
	BP Busy	BP Busy in Erase Suspend						Erase Suspend
	BP Suspend	BP Suspend in Erase Suspend						
Lock/CR Setup in Erase Suspend	Erase Suspend (Lock Error)	Erase Suspend (Lock Block)	Erase Suspend (Lock Down Block)	Erase Suspend (Set CR)	Erase Suspend (Lock Error)		N/A	
Buffered Enhanced Factory Program Mode	Setup	Ready (Error)			Ready (BEFP Loading Data)	Ready (Error)		
	BEFP Busy	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 37. Write State Machine—Next State Table (Sheet 5 of 6)

Output Next State Table											
Command Input to Chip and resulting <i>Output</i> Mux Next State											
Current chip state	Read Array <sup>(2)</sup>	Word Program Setup (3,4)	BP Setup	Erase Setup <sup>(3,4)</sup>	Buffered Enhanced Factory Pgm Setup <sup>(3,4)</sup>	BE Confirm, P/E Resume, ULB Confirm <sup>(8)</sup>	Program/ Erase Suspend	Read Status	Clear Status Register <sup>(5)</sup>	Read ID/Query	Lock, Unlock, Lock-down, CR setup <sup>(4)</sup>
	(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	Status Read	Output does not change.	Status Read	Output mux does not change.	ID Read	Status Read	Status Read		
Ready, Erase Suspend, BP Suspend, 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 38. Write State Machine—Next State Table (Sheet 6 of 6)

<i>Output</i> Next State Table						
Current chip state	Command Input to Chip and resulting <i>Output</i> Mux Next State					
	OTP Setup <sup>(4)</sup>  (C0H)	Lock Block Confirm <sup>(8)</sup>  (01H)	Lock-Down Block Confirm <sup>(8)</sup>  (2FH)	Write CR Confirm <sup>(8)</sup>  (03H)	Block Address (?WA0)  (FFFFH)	Illegal Cmds or BEFP Data <sup>(1)</sup>  (all other codes)
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					Output does not change.
Lock/CR Setup, Lock/CR Setup in Erase Susp	Status Read		Array Read	Status Read		
OTP Busy	Status Read	Output does not change.		Array Read	Output does not change.	
Ready, Erase Suspend, BP Suspend 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	Status Read	Output does not change.		Array Read	Output does not change.	

**NOTES:**

1. "Illegal commands" include commands outside of the allowed command set (allowed commands: 40H [pgm], 20H [erase], etc.)
2. If a "Read Array" is attempted from a busy partition, the result will be invalid data. The ID and Query data are located at different locations in the address map.
3. 1st and 2nd cycles of "2 cycles write commands" must be given to the same partition address, or unexpected results will occur.
4. To protect memory contents against erroneous command sequences, there are specific instances in a multi-cycle command sequence in which the second cycle will be ignored. For example, when the device is program suspended and an erase setup command (0x20) is given followed by a confirm/resume command

- (0xD0), the second command will be ignored because it is unclear whether the user intends to erase the block or resume the program operation.
5. 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).
  6. BEFP writes are only allowed when the status register bit #0 = 0, or else the data is ignored.
  7. 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.
  8. Confirm commands (Lock Block, Unlock Block, Lock-Down Block, Configuration Register) perform the operation and then move to the Ready State.
  9. WA0 refers to the block address latched during the first write cycle of the current operation.



## Appendix B Flowcharts

Figure 39. Word Program Flowchart

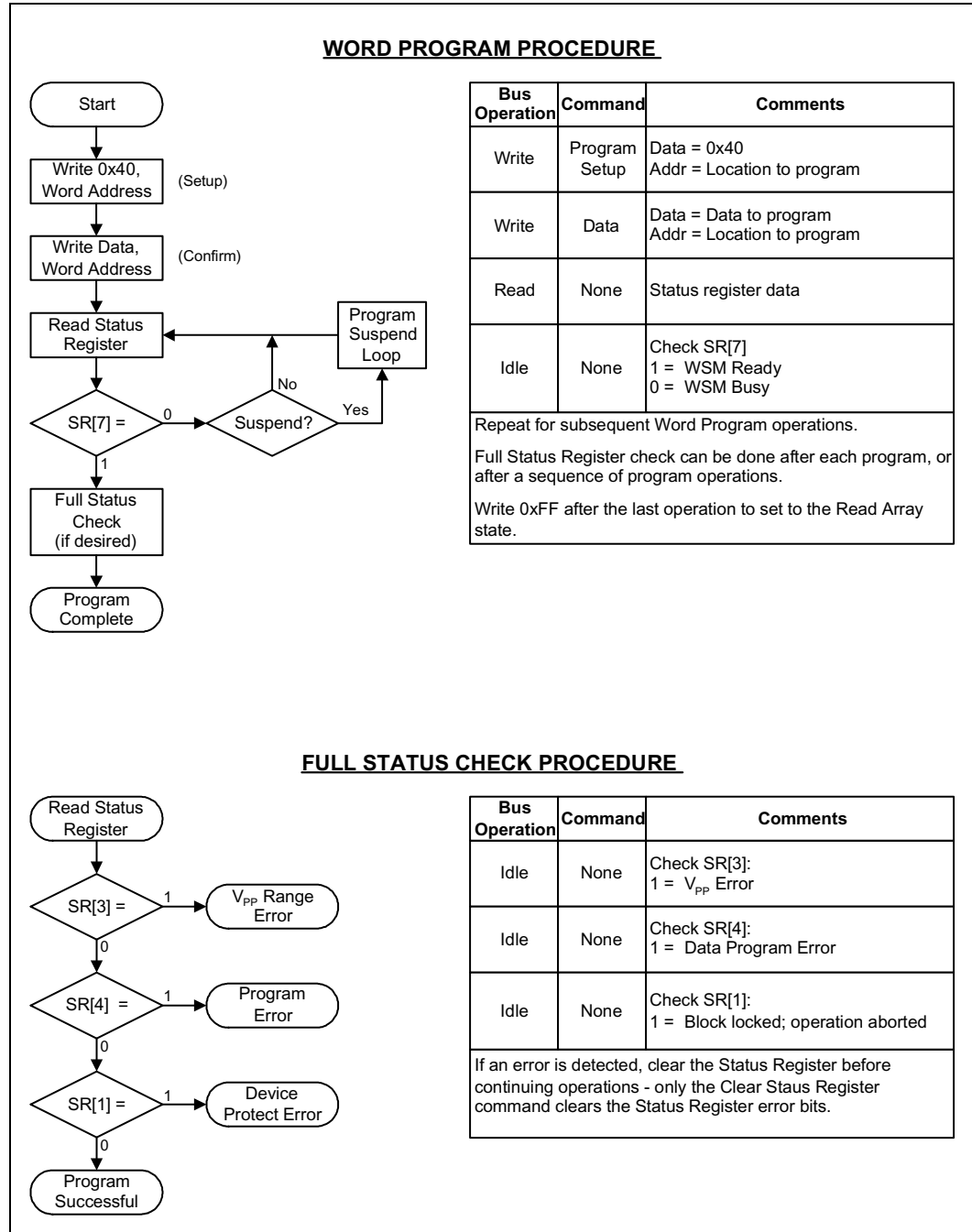


Figure 40. Program Suspend/Resume Flowchart

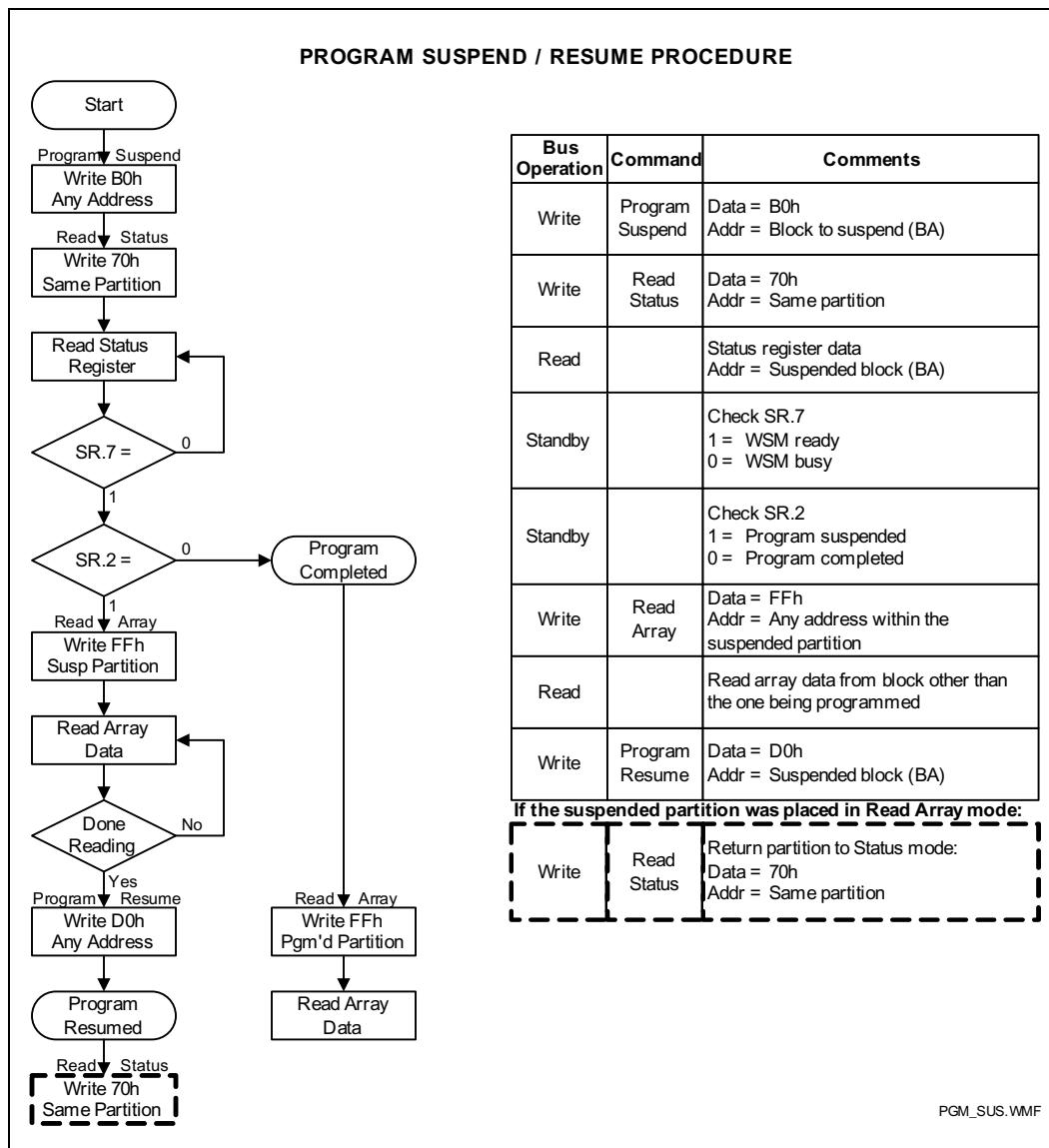


Figure 41. Buffer Program Flowchart

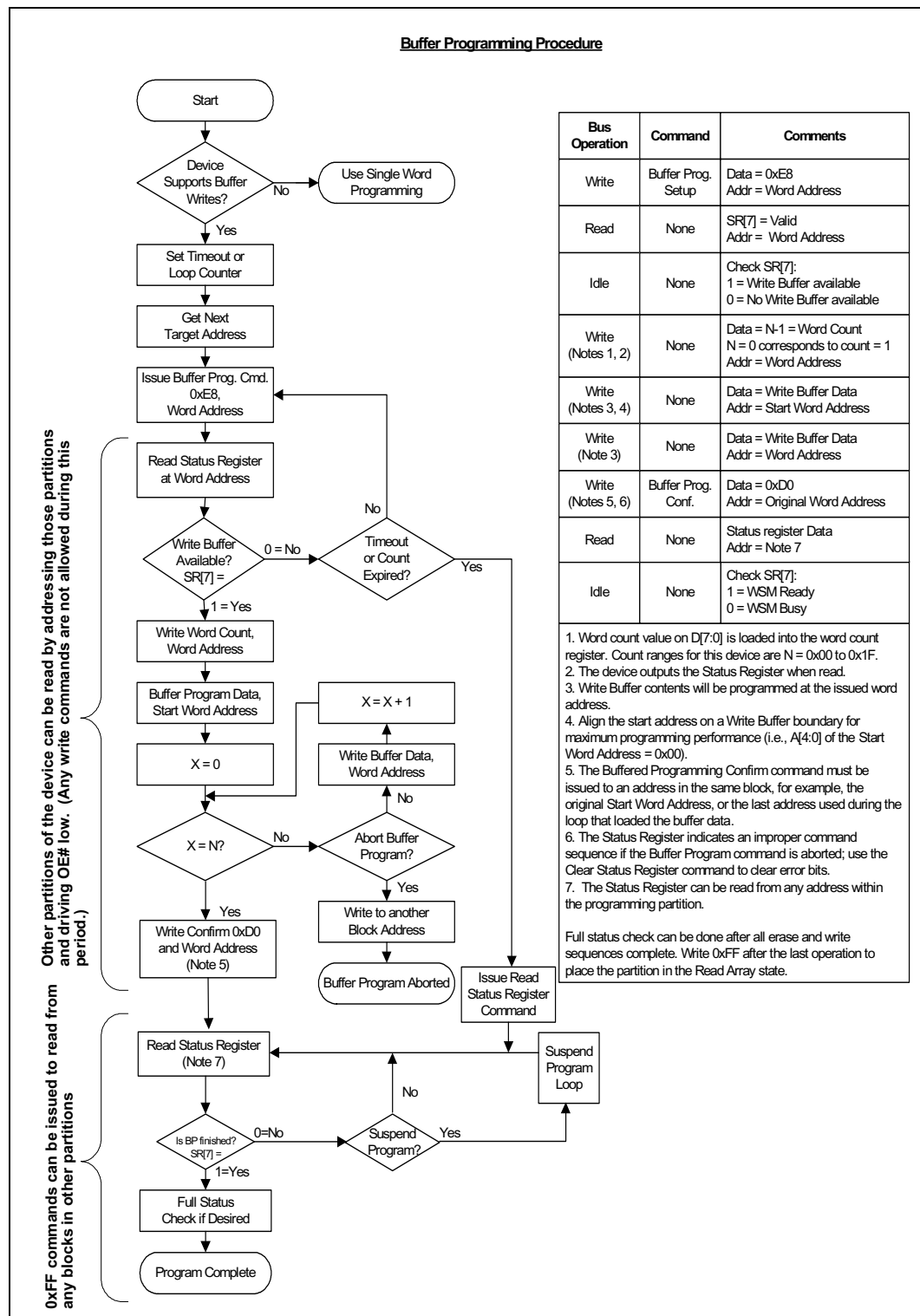


Figure 42. Buffered EFP Flowchart

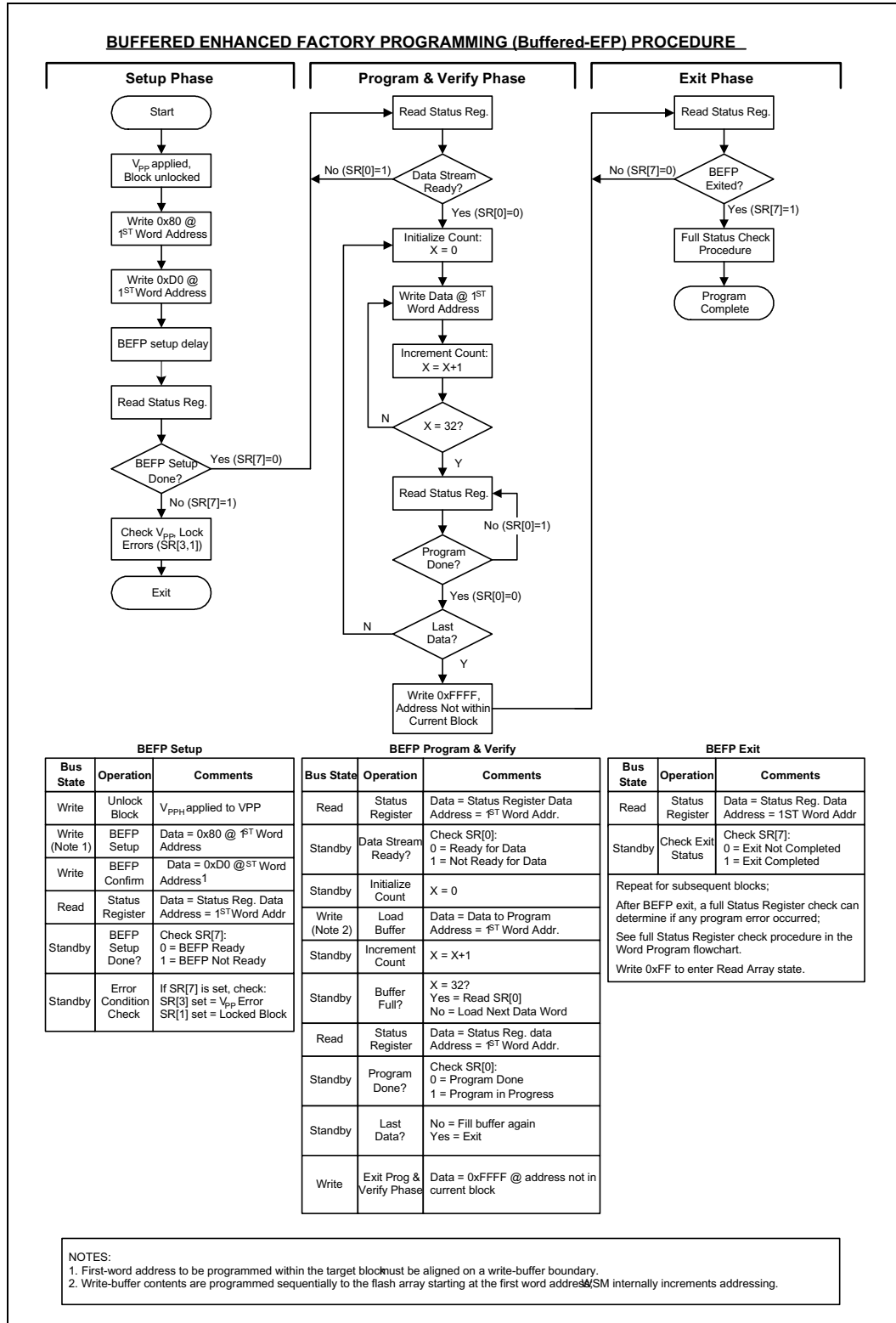


Figure 43. Block Erase Flowchart

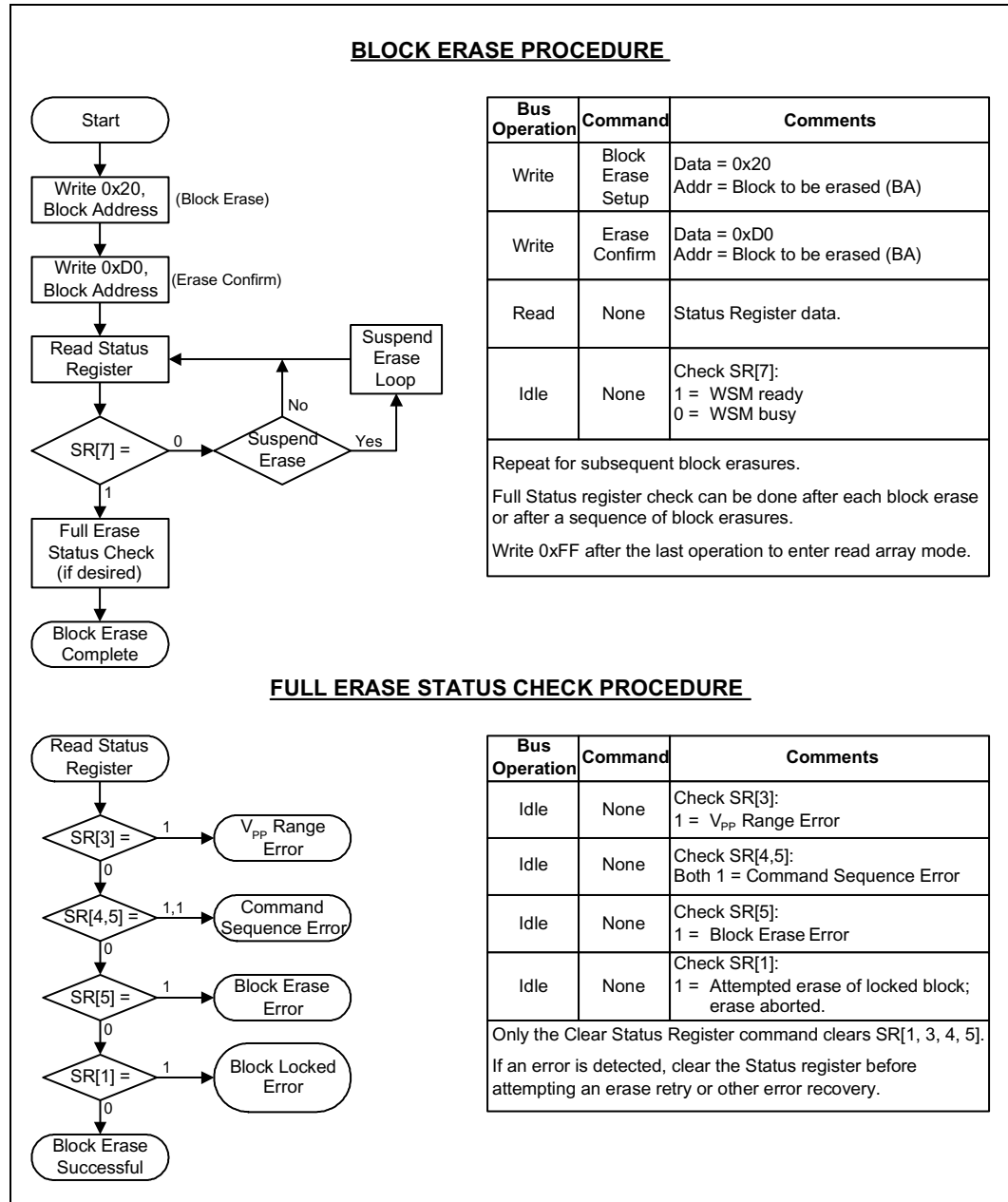


Figure 44. Erase Suspend/Resume Flowchart

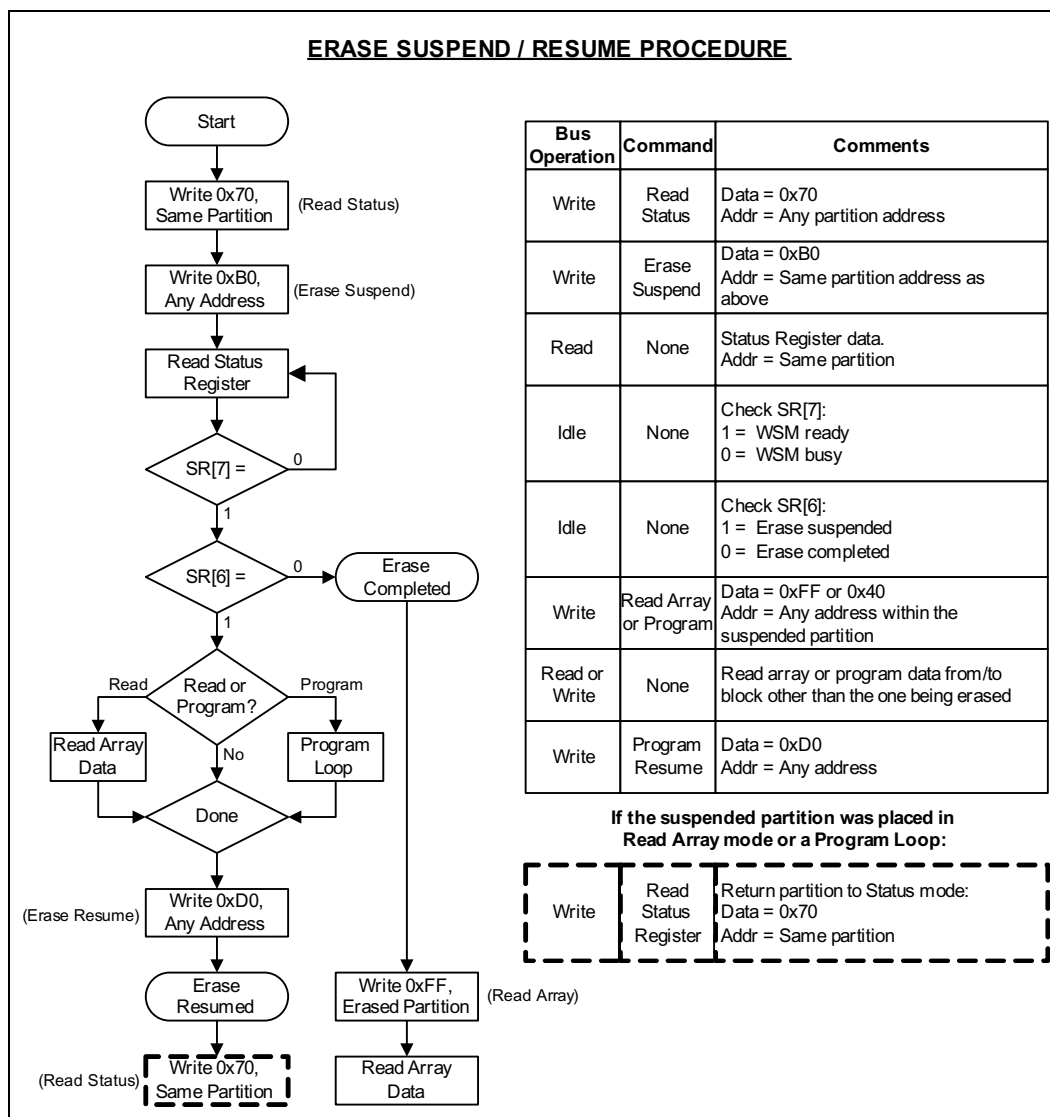


Figure 45. Block Lock Operations Flowchart

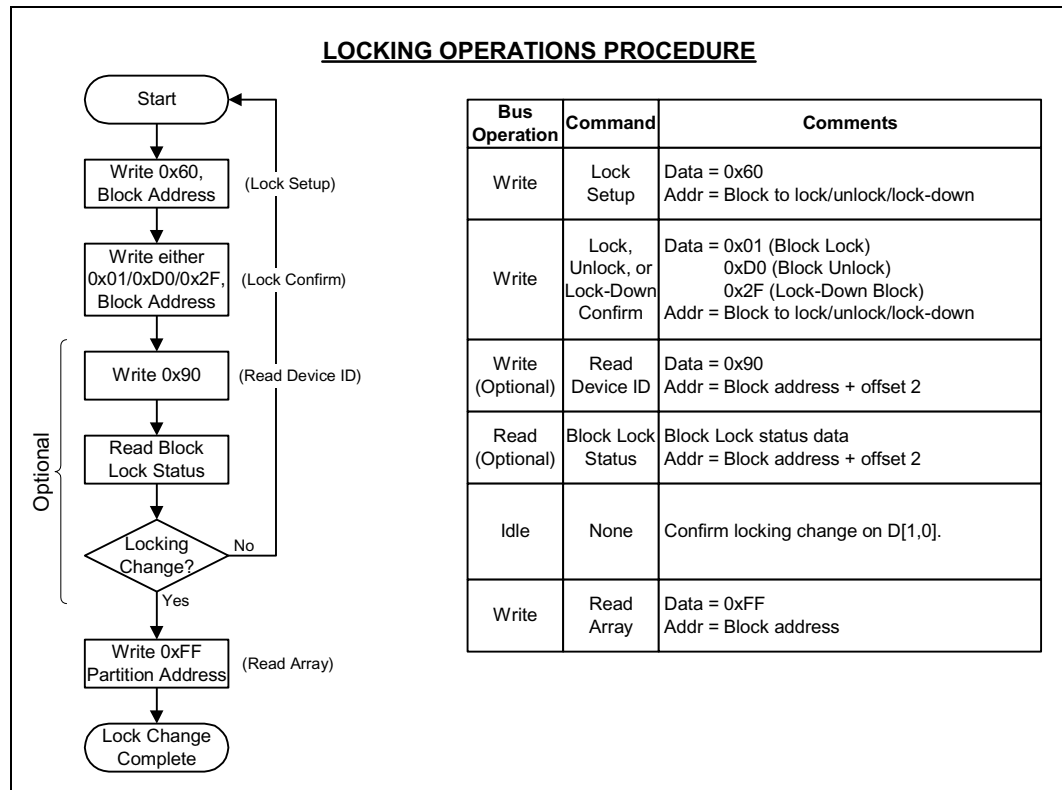
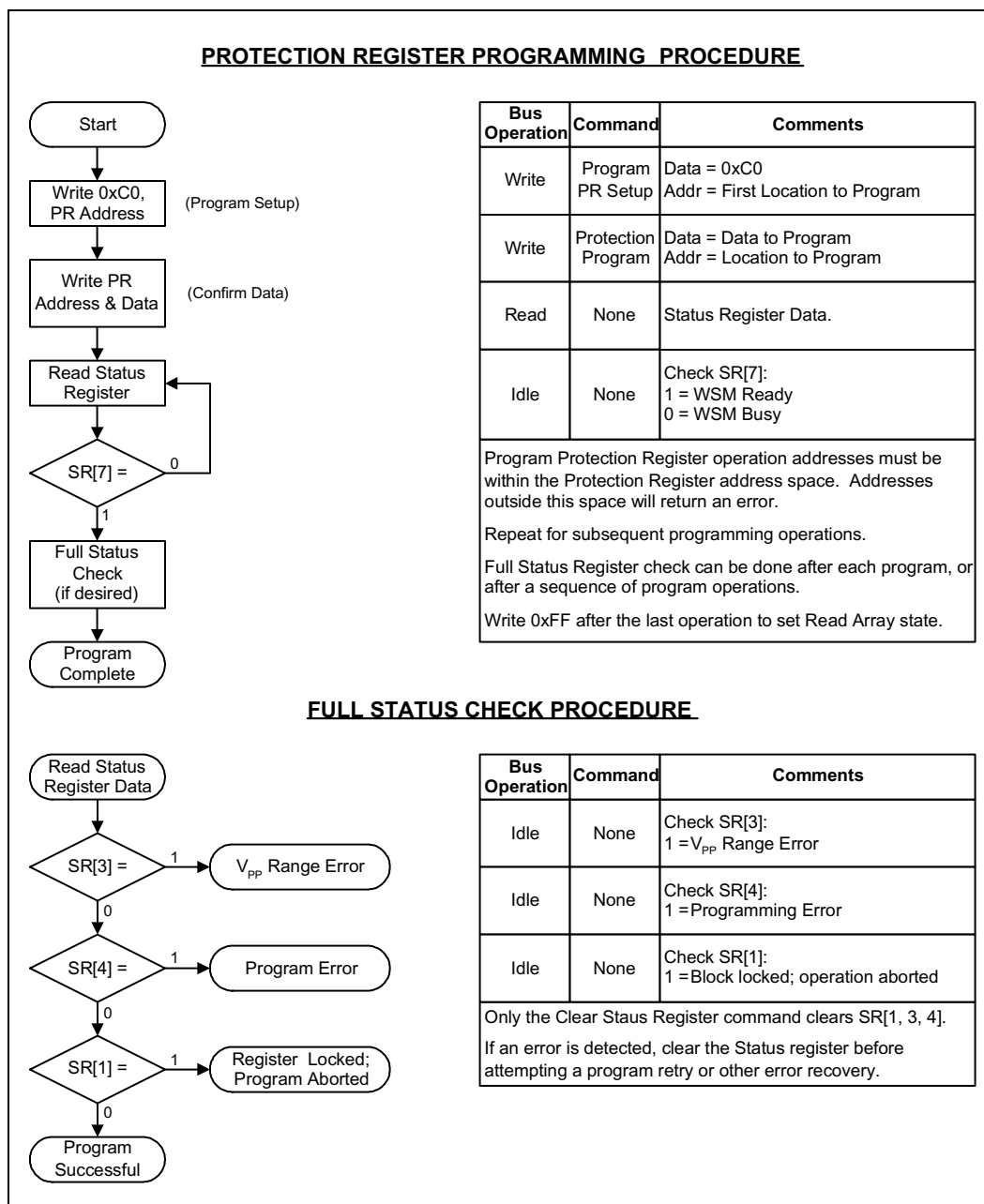


Figure 46. Protection Register Programming Flowchart





## Appendix C Common Flash Interface

The 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 9.2, “Device Commands” on page 41](#)). 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<sub>7-0</sub>) 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<sub>7-0</sub>) and 00h in the high byte (DQ<sub>15-8</sub>).

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

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

**Table 19. Example of Query Structure Output of x16- Devices**

Word Addressing:			Byte Addressing:		
Offset	Hex Code	Value	Offset	Hex Code	Value
A <sub>X</sub> -A <sub>0</sub>	D <sub>15</sub> -D <sub>0</sub>		A <sub>X</sub> -A <sub>0</sub>	D <sub>7</sub> -D <sub>0</sub>	
00010h	0051	"Q"	00010h	51	"Q"
00011h	0052	"R"	00011h	52	"R"
00012h	0059	"Y"	00012h	59	"Y"
00013h	P_ID <sub>LO</sub>	PrVendor	00013h	P_ID <sub>LO</sub>	PrVendor
00014h	P_ID <sub>HI</sub>	ID #	00014h	P_ID <sub>LO</sub>	ID #
00015h	P <sub>LO</sub>	PrVendor	00015h	P_ID <sub>HI</sub>	ID #
00016h	P <sub>HI</sub>	TblAdr	00016h	...	...
00017h	A_ID <sub>LO</sub>	AltVendor	00017h		
00018h	A_ID <sub>HI</sub>	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 20. Query Structure**

Offset	Sub-Section Name	Description <sup>(1)</sup>
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 <sup>(3)</sup>	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 16-Kword).
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 21. 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:	--01 --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 22. System Interface Information**

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

## C.4 Device Geometry Definition

Table 23. Device Geometry Definition

Offset	Length	Description	Code																																		
27h	1	"n" such that device size = 2 <sup>n</sup> 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:	28:	--01	x16																																
		<table border="1" style="width: 100%; text-align: center;"> <tr> <td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td> </tr> <tr> <td>—</td><td>—</td><td>—</td><td>—</td><td>x64</td><td>x32</td><td>x16</td><td>x8</td> </tr> <tr> <td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td> </tr> <tr> <td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td> </tr> </table>				7	6	5	4	3	2	1	0	—	—	—	—	x64	x32	x16	x8	15	14	13	12	11	10	9	8	—	—	—	—	—	—	—	—
		7				6	5	4	3	2	1	0																									
		—				—	—	—	x64	x32	x16	x8																									
15	14	13	12	11	10	9	8																														
—	—	—	—	—	—	—	—																														
29:	--00																																				
2Ah	2	"n" such that maximum number of bytes in write buffer = 2 <sup>n</sup>	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:	--02	--02	--02	--02	--02	--02
2D:	--03	--3E	--03	--7E	--03	--FE
2E:	--00	--00	--00	--00	--00	--00
2F:	--80	--00	--80	--00	--80	--00
30:	--00	--02	--00	--02	--00	--02
31:	--3E	--03	--7E	--03	--FE	--03
32:	--00	--00	--00	--00	--00	--00
33:	--00	--80	--00	--80	--00	--80
34:	--02	--00	--02	--00	--02	--00
35:	--00	--00	--00	--00	--00	--00
36:	--00	--00	--00	--00	--00	--00
37:	--00	--00	--00	--00	--00	--00
38:	--00	--00	--00	--00	--00	--00

## C.5 Intel-Specific Extended Query Table

Table 24. Primary Vendor-Specific Extended Query

Offset <sup>(1)</sup> P = 10Ah	Length	Description (Optional flash features and commands)	Add.	Hex Code	Value
(P+0)h (P+1)h (P+2)h	3	Primary extended query table Unique ASCII string "PRI"	10A: 10B: 10C:	--50 --52 --49	"P" "R" "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 (P+6)h (P+7)h (P+8)h	4	Optional feature and command support (1=yes, 0=no) <i>bits 10–31 are reserved; undefined bits are "0." If bit 31 is "1" then another 31 bit field of Optional features follows at the end of the bit–30 field.</i> bit 0 Chip erase supported bit 1 Suspend erase supported bit 2 Suspend program supported bit 3 Legacy lock/unlock supported bit 4 Queued erase supported bit 5 Instant individual block locking supported bit 6 Protection bits supported bit 7 Pagemode read supported bit 8 Synchronous read supported bit 9 Simultaneous operations supported	10F: 110: 111: 112:	--E6 --03 --00 --00	    No Yes Yes No No Yes Yes Yes Yes Yes
(P+9)h	1	Supported functions after suspend: read Array, Status, Query Other supported operations are: bits 1–7 reserved; undefined bits are "0" bit 0 Program supported after erase suspend	113:	--01	   Yes
(P+A)h (P+B)h	2	Block status register mask <i>bits 2–15 are Reserved; undefined bits are "0"</i> bit 0 Block Lock-Bit Status register active bit 1 Block Lock-Down Bit Status active	114: 115:	--03 --00	  Yes Yes
(P+C)h	1	V <sub>CC</sub> logic supply highest performance program/erase voltage bits 0–3 BCD value in 100 mV bits 4–7 BCD value in volts	116:	--18	1.8V
(P+D)h	1	V <sub>PP</sub> optimum program/erase supply voltage bits 0–3 BCD value in 100 mV bits 4–7 HEX value in volts	117:	--90	9.0V

Table 25. Protection Register Information

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

Table 26. Burst Read Information

Offset <sup>(1)</sup> 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 <sup>n</sup> 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:	--03	8 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 <sup>n+1</sup> 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 27. Partition and Erase-block Region Information**

Offset <sup>(1)</sup> P= 10Ah		Description  (Optional flash features and commands)	See table below		
Bottom	Top		Len	Address	
				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:

Table 28. Partition Region 1 Information

Offset <sup>(1)</sup> P = 10Ah		Description  (Optional flash features and commands)	See table below		
Bottom	Top		Len	Address	
				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:	



Table 29. Partition Region 2 Information

Offset <sup>(1)</sup> P = 10Ah		Description  (Optional flash features and commands)	See table below		
Bottom	Top		Len	Address	
				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:

Table 30. Partition and Erase Block Region Information

Address	64 Mbit		128 Mbit		256 Mbit	
	-B	-T	-B	-T	-B	-T
12D:	--02	--02	--02	--02	--02	--02
12E:	--01	--07	--01	--0F	--01	--0F
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	--0F
135:	--00	--00	--00	--00	--00	--00
136:	--80	--00	--80	--00	--80	--00
137:	--00	--02	--00	--02	--00	--02
138:	--64	--64	--64	--64	--64	--64
139:	--00	--00	--00	--00	--00	--00
13A:	--02	--02	--02	--02	--02	--02
13B:	--03	--03	--03	--03	--03	--03
13C:	--06	--01	--06	--01	--0E	--01
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	--02	--00	--02	--00	--02
142:	--02	--06	--02	--06	--02	--0E
143:	--03	--00	--03	--00	--03	--00
144:	--07	--00	--0F	--00	--0F	--00
145:	--00	--02	--00	--02	--00	--02
146:	--11	--64	--11	--64	--11	--64
147:	--00	--00	--00	--00	--00	--00
148:	--00	--02	--00	--02	--00	--02
149:	--01	--03	--01	--03	--01	--03
14A:	--07	--03	--07	--03	--0F	--03
14B:	--00	--00	--00	--00	--00	--00
14C:	--00	--80	--00	--80	--00	--80
14D:	--02	--00	--02	--00	--02	--00
14E:	--64	--64	--64	--64	--64	--64
14F:	--00	--00	--00	--00	--00	--00
150:	--02	--02	--02	--02	--02	--02
151:	--03	--03	--03	--03	--03	--03

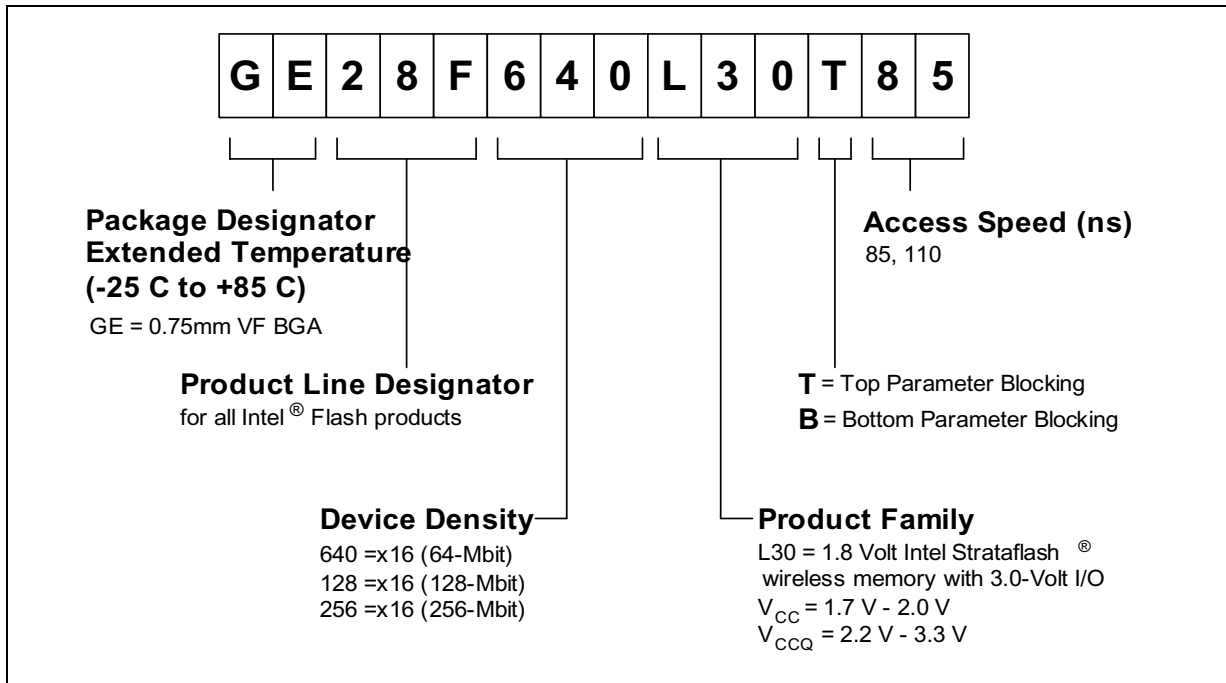
## Appendix D Additional Information

Order/Document Number	Document/Tool
251902	Intel StrataFlash® Wireless Memory (L18)
290701	Intel® Wireless Flash Memory (W18) Datasheet
290702	Intel® Wireless Flash Memory (W30)
290737	Intel StrataFlash® Synchronous Memory (K3/K18) 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>.

## Appendix E Ordering Information for VF BGA Package



## Appendix F Ordering Information for SCSP Package

Figure 47 shows the decoder for the 1.8 Volt Intel StrataFlash® wireless memory in Quad+ ballout products.

Figure 47. Decoder for L30 in Quad+ Ballout

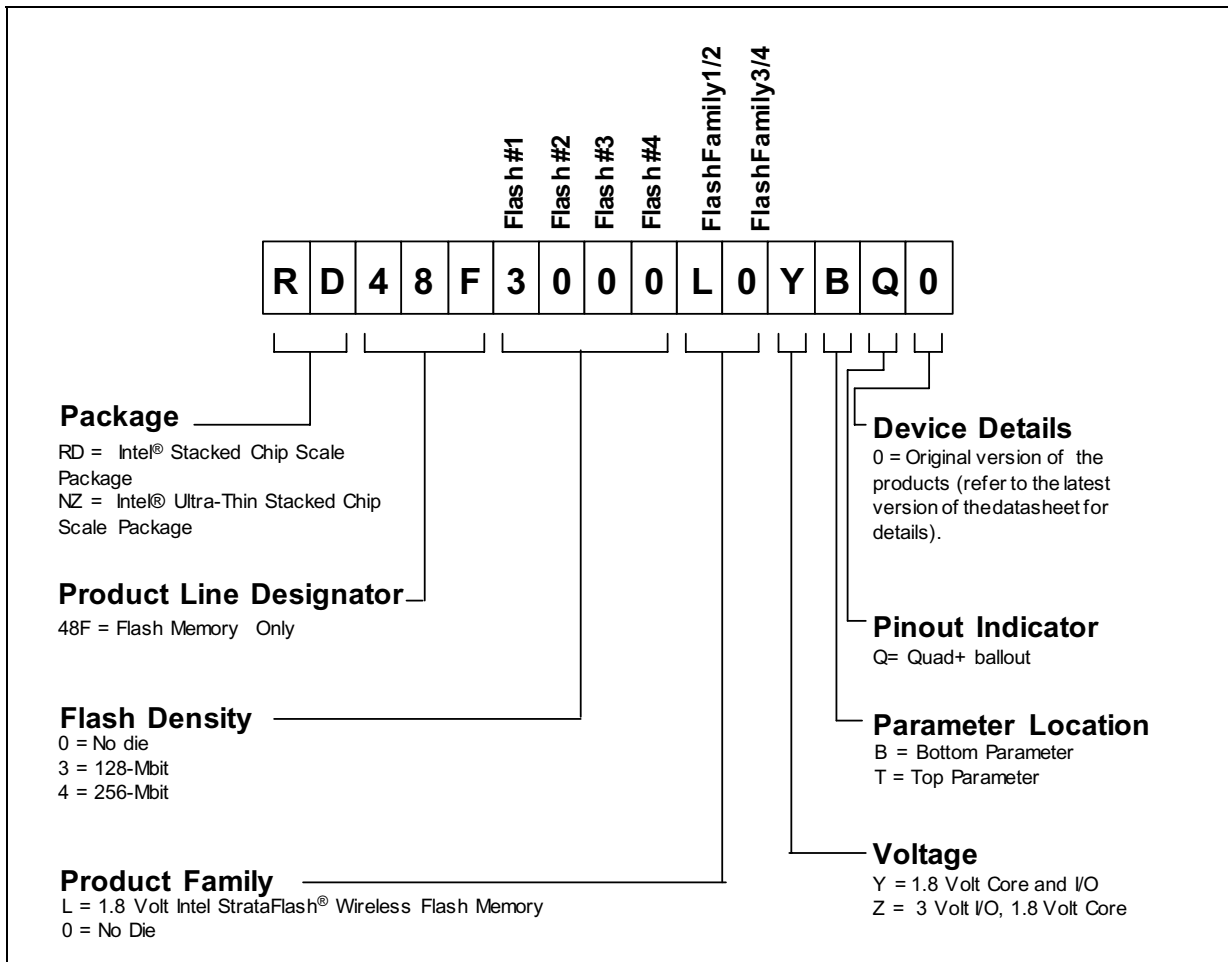


Table 31. Valid Combinations for SCSP Package

I/O	128-Mbit	256-Mbit
3.0 V I/O	RD48F3000L0ZTQ0	NZ48F4000L0ZTQ0
	RD48F3000L0ZBQ0	NZ48F4000L0ZBQ0

