Numonyx™ StrataFlash® Wireless Memory (L18) with AD-Multiplexed I/O

Datasheet

Product Features

- High performance Read-While-Write/Erase
  - 85 ns initial access
  - 54MHz with zero wait state, 14 ns clock-to-data output synchronous-burst mode
  - 4-, 8-, 16-, and continuous-word burst mode
  - Burst suspend
  - Programmable WAIT configuration
  - Buffered Enhanced Factory Programming (Buffered EFP): 5 µs/byte (Typ)
  - 1.8 V low-power buffered and non-buffered programming @ 7 µs/byte (Typ)
- Architecture
  - Asymmetrically-blocked architecture
  - Multiple 8-Mbit partitions: 64Mb and 128Mb devices
  - Multiple 16-Mbit partitions: 256Mb devices
  - Four 16-KWord parameter blocks: top configuration
  - 64-KWord main blocks
  - Dual-operation: Read-While-Write (RWW) or Read-While-Erase (RWE)
  - Status register for partition and device status
- Density and Packaging
  - 64-, 128-, and 256 Mbit density in VF BGA package
  - 16-bit wide data bus
- Power
  - 1.7 V to 2.0 V V_{CC} operation
  - I/O voltage: 1.35 V – 2.0 V, 1.7 V– 2.0 V
  - Standby current: 25 µA (Typ) for 256-Mbit
  - 4-Word synchronous read current: 15 mA (Typ) @ 54 MHz
  - Automatic Power Savings (APS) mode
- 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
- Software
  - 20 µs (Typ) program suspend
  - 20 µs (Typ) erase suspend
  - Intel® Flash Data Integrator (FDI) optimized
  - Basic Command Set (BCS) and Extended Command Set (ECS) compatible
  - Common Flash Interface (CFI) capable
- Quality and Reliability
  - Expanded temperature: –25° C to +85° C
  - Minimum 100,000 erase cycles per block
  - Intel ETOX* VIII process technology (0.13 µm)

313295-04
November 2007

Contents

1.0 Introduction 7
1.1 Nomenclature 7
1.2 Acronyms 7
1.3 Conventions 8
2.0 Functional Overview 9
3.0 Package Information 10
4.0 Ballout and Signal Descriptions 12
4.1 Signal Descriptions 13
4.2 Memory Map 15
5.0 Maximum Ratings and Operating Conditions 17
5.1 Absolute Maximum Ratings 17
5.2 Operating Conditions 17
6.0 Electrical Specifications 18
6.1 DC Current Characteristics 18
6.2 DC Voltage Characteristics 19
7.0 AC Characteristics 20
7.1 AC Test Conditions 20
7.2 Capacitance 21
7.3 AC Read Specifications (VCCQ = 1.35 V – 2.0 V) 21
7.4 AC Read Specifications: 64- and 128-Mb Densities 22
7.5 AC Read Specifications for 256-Mb Density 23
7.6 AC Write Specifications 30
7.7 Program and Erase Characteristics 32
8.0 Power and Reset Specifications 33
8.1 Power Up and Down 33
8.2 Reset 33
8.3 Power Supply Decoupling 34
8.4 Automatic Power Saving (APS) 35
9.0 Device Operations 36
9.1 Bus Operations 36
9.1.1 Reads 36
9.1.2 Writes 36
9.1.3 Output Disable 36
9.1.4 Standby 37
9.1.5 Reset 37
9.2 Device Commands 37
9.3 Command Definitions 38
10.0 Read Operations 41
10.1 Asynchronous Read Mode 41
10.2 Synchronous Burst-Mode Read 41
10.2.1 Burst Suspend 42
10.3 Read Configuration Register (RCR) 42
10.3.1 Read Mode 43
10.3.2 Latency Count 43
10.3.3 WAIT Polarity 45
10.3.3.1 WAIT Signal Function 45
10.3.4 Data Hold 46

10.3.5 WAIT Delay 46
10.3.6 Burst Sequence 47
10.3.7 Clock Edge 47
10.3.8 Burst Wrap 48
10.3.9 Burst Length 48

11.0 Programming Operations 49
11.1 Word Programming 49
11.1.1 Factory Word Programming 50
11.2 Buffered Programming 50
11.3 Buffered Enhanced Factory Programming 51
11.3.1 Buffered EFP Requirements and Considerations 51
11.3.2 Buffered EFP Setup Phase 52
11.3.3 Buffered EFP Program/Verify Phase 52
11.3.4 Buffered EFP Exit Phase 53
11.4 Program Suspend 53
11.5 Program Resume 53
11.6 Program Protection 53

12.0 Erase Operations 55
12.1 Block Erase 55
12.2 Erase Suspend 55
12.3 Erase Resume 56
12.4 Erase Protection 56

13.0 Security Modes 57
13.1 Block Locking 57
13.1.1 Lock Block 57
13.1.2 Unlock Block 57
13.1.3 Lock-Down Block 57
13.1.4 Block Lock Status 58
13.1.5 Block Locking During Suspend 58
13.2 Protection Registers 59
13.2.1 Reading the Protection Registers 60
13.2.2 Programming the Protection Registers 61
13.2.3 Locking the Protection Registers 61

14.0 Dual-Operation Considerations 62
14.1 Memory Partitioning 62
14.2 Read-While-Write Command Sequences 62
14.2.1 Simultaneous Operation Details 63
14.2.2 Synchronous and Asynchronous Read-While-Write Characteristics and Waveforms 63
14.2.2.1 Write operation to asynchronous read transition 63
14.2.2.2 Write to synchronous read operation transition 63
14.2.2.3 Write Operation with Clock Active 64
14.2.3 Read Operation During Buffered Programming Flowchart 64
14.2.4 Simultaneous Operation Restrictions 64

15.0 Special Read States 66
15.1 Read Status Register 66
15.1.1 Clear Status Register 67
15.2 Read Device Identifier 67
15.3 CFI Query 68

A Write State Machine 69

B Flowcharts 76

B.1 Common Flash Interface (CFI) 83
B.2 Query Structure Output..... 84
B.3 Query Structure Overview 84
B.4 CFI Query Identification String 85
B.5 Device Geometry Definition 87
B.6 Intel-Specific Extended Query Table..... 88

C Ordering Information 96

Revision History

Date	Revision	Description
May 2006	001	Initial Release
July 2006	002	Removed Intel Confidential status.
August 2007	003	Updated ordering information
November 2007	04	Applied Numonyx branding.

1.0 Introduction

This document provides information about the Numonyx™ StrataFlash® Wireless Memory (L18) with AD-Multiplexed I/O device. This document describes device features, operation, and specifications.

The Numonyx™ StrataFlash® Wireless Memory (L18) with AD-Multiplexed I/O product is the latest generation of Intel StrataFlash® memory featuring flexible, multiple-partition, dual operation. It provides high performance asynchronous read mode and synchronous-burst read mode using 1.8 V low-voltage, multi-level cell (MLC) technology.

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

The Numonyx™ StrataFlash® Wireless Memory (L18) with AD-Multiplexed I/O device is manufactured using Intel 0.13 µm ETOX™ VIII process technology, available in industry-standard chip scale packaging.

1.1 Nomenclature

1.8 V	Vcc voltage range of 1.7 V – 2.0 V (except where noted)
1.8 V Extended Range	Vccq voltage range of 1.35 V – 2.0 V
VPP = 9.0 V	VPP 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 device has two block sizes: 16K-Word and 64K-Word.
Main block	An array block that is usually used to store code and/or data. Main blocks are larger than parameter blocks.
Parameter block	An array block that is usually used to store frequently changing data or small system parameters that traditionally would be stored in EEPROM.
Top parameter device	Previously referred to as a top-boot device, a device with its parameter partition located at the highest physical address of its memory map. Parameter blocks within a parameter partition are located at the highest physical address of the parameter partition.
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
SR	Status Register
WSM	Write State Machine

1.3 Conventions

VCC	Signal or voltage connection
V_{CC}	Signal or voltage level
0x	Hexadecimal number prefix
0b	Binary number prefix
SR[4]	Denotes an individual register bit
A[15:0]	Denotes a group of similarly named signals, such as address or data bus
A5	
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

The Numonyx™ StrataFlash® Wireless Memory (L18) with AD-Multiplexed I/O device provides read-while-write and read-while-erase capability with density upgrades through 256-Mbit. This device 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 for the 64-Mbit and 128-Mbit devices, and into multiple 16-Mbit partitions for the 256-Mbit device. 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 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 read mode. 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, it 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 VPP at 9.0 V**,** which increases factory throughput. With V_{PP} at 1.8 V, VCC and VPP can be tied together for a simple, ultra low power design. In addition to voltage flexibility, a dedicated V_{PP} connection provides complete data protection when V_{PP} is less than V_{PPLK}.

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

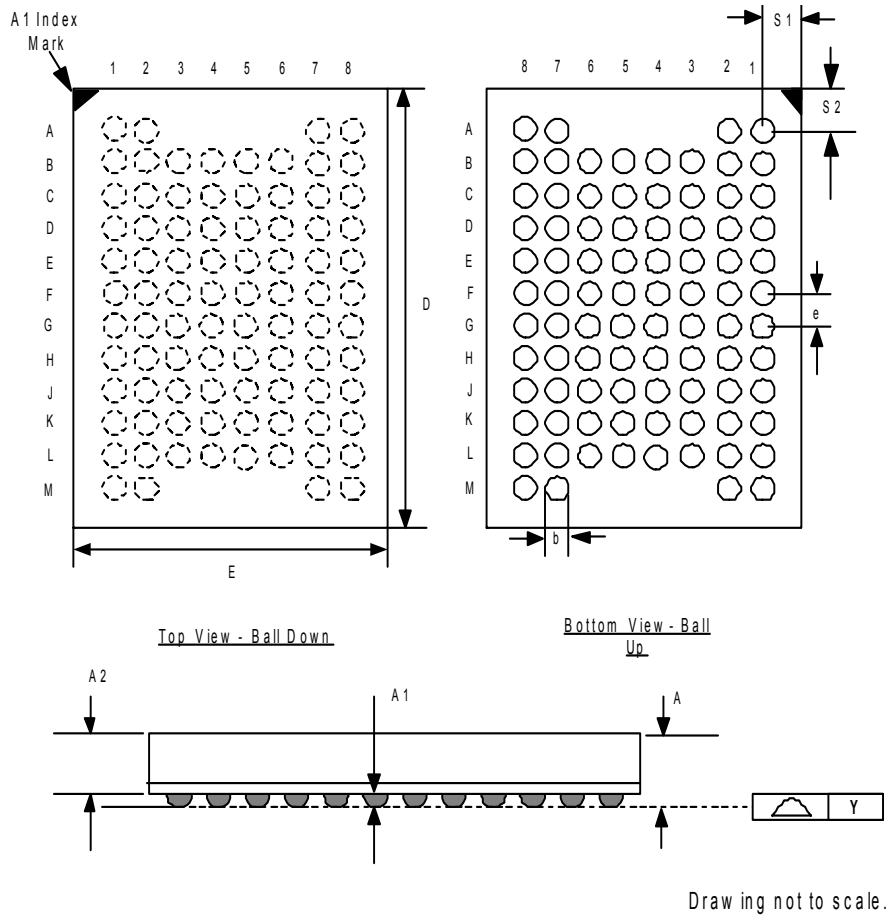
An industry-standard command sequence invokes program and erase automation. Each erase operation erases one block. The Erase Suspend feature allows system software to pause an erase cycle to read or program data in another block. Program Suspend allows system software to pause programming to read other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

The Numonyx™ StrataFlash® Wireless Memory (L18 AD-Mux) device offers power savings through Automatic Power Savings (APS) mode and standby mode. The device automatically enters APS following read-cycle completion. Standby is initiated when the system deselects the device by deasserting CE# or by asserting RST#. Combined, these features can significantly reduce power consumption.

The 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.

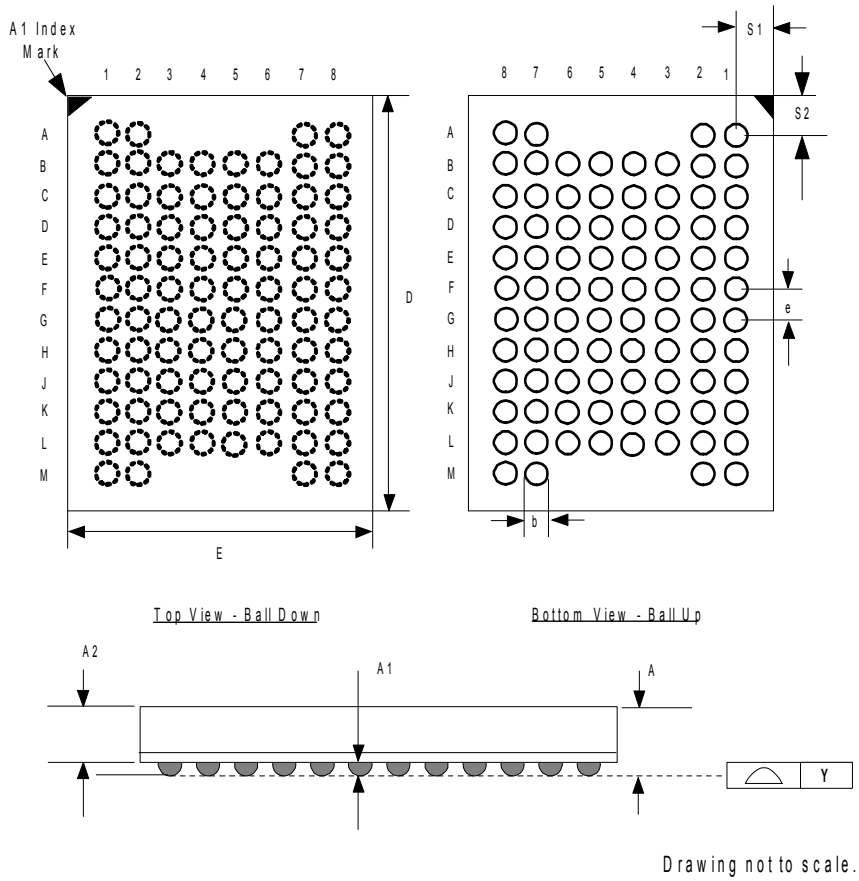
3.0 Package Information

Figure 1: 64- and 128-Mbit, 88-ball (80-active ball) (8x10x1.2 mm)



		Millimeters				Inches		
Dimensions	**Symbol**	**Min**	**Nom**	**Max**	**Notes**	**Min**	**Nom**	**Max**
Package Height	A			1.200				0.0472
Ball Height	A1	0.200				0.0079		
Package Body Thickness	A2		0.860				0.0339	
Ball (Lead) Width	b	0.325	0.375	0.425		0.0128	0.0148	0.0167
Package Body Length	D	9.900	10.000	10.100		0.3898	0.3937	0.3976
Package Body Width	E	7.900	8.000	8.100		0.3110	0.3150	0.3189
Pitch	e		0.800				0.0315	
Ball (Lead) Count	N		88				88	
Seating Plane Coplanarity	Y			0.100				0.0039
Corner to Ball A1 Distance Along E	S1	1.100	1.200	1.300		0.0433	0.0472	0.0512
Corner to Ball A1 Distance Along D	S2	0.500	0.600	0.700		0.0197	0.0236	0.0276

Figure 2: 256-Mbit, 88-ball (80-active ball) (8x11x1.0 mm)



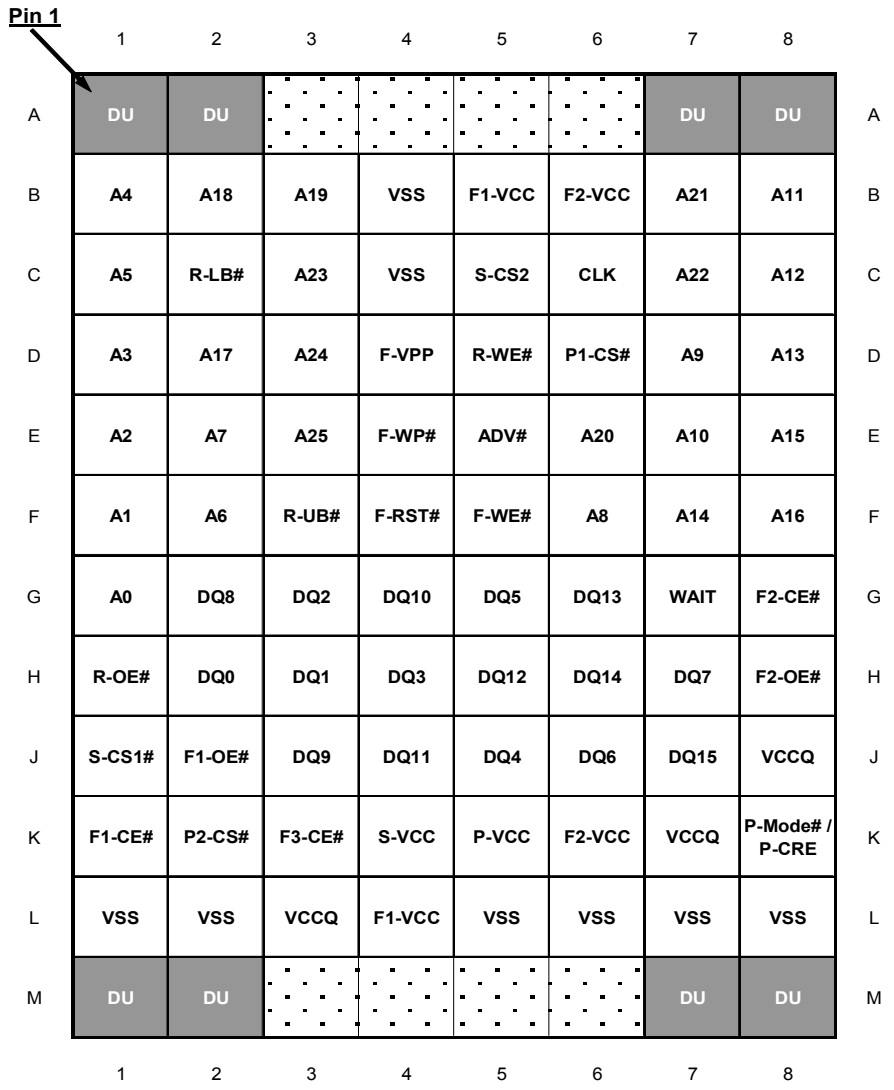
Drawing not to scale.

Note: Dimensions A1, A2, and b are preliminary

		Millimeters				Inches		
Dimensions	**Symbol**	**Min**	**Nom**	**Max**	**Notes**	**Min**	**Nom**	**Max**
Package Height	A			1.00				0.0394
Ball Height	A1	0.117				0.0046		
Package Body Thickness	A2		0.740				0.0291	
Ball (Lead) Width	b	0.300	0.350	0.400		0.0118	0.0138	0.0157
Package Body Length	D	10.900	11.00	11.100		0.4291	0.4331	0.4370
Package Body Width	E	7.900	8.00	8.100		0.3110	0.3150	0.3189
Pitch	e		0.80				0.0315	
Ball (Lead) Count	N		88				88	
Seating Plane Coplanarity	Y			0.100				0.0039
Corner to Ball A1 Distance Along E	S1	1.100	1.200	1.300		0.0433	0.0472	0.0512
Corner to Ball A1 Distance Along D	S2	1.000	1.100	1.200		0.0394	0.0433	0.0472

4.0 Ballout and Signal Descriptions

Table 1: QUAD+ Ballout



4.1 Signal Descriptions

Table 2: Signal Descriptions (Sheet 1 of 3)

Symbol	Type	Signal Descriptions	Notes
Address and Data Signals, AD-Mux			
A[MAX:16]	Input	**ADDRESS:** Global device signals. Shared address inputs for all memory die during Read and Write operations. • 256-Mbit: AMAX = A23 • 128-Mbit: AMAX = A22 • 64-Mbit: AMAX = A21 • A0 is the lowest-order word address. • Unused address inputs should be treated as RFU.	
DQ[15:0]	Input / Output	**ADDRESS-DATA MULTIPLEXED INPUTS/ OUTPUTS:** AD-Mux I/O flash signals. During AD-Mux Read cycles, DQ[15:0] are used to input the lower address followed by read-data output. During AD-Mux Write cycles, DQ[15:0] are used to input the lower address followed by commands or data. • DQ[15:0] are High-Z when the device is deselected or its output is disabled. • DQ[15:0] is only used with AD-Mux I/O flash device.	1
Control Signals			
ADV#	Input	**ADDRESS VALID:** Flash- and Synchronous PSRAM-specific signal; low-true input. • During a synchronous flash Read operation, the address is latched on the rising edge of ADV# or the first active CLK edge whichever occurs first. In an asynchronous flash Read operation, the address is latched on the rising edge of ADV# or continuously flows through while ADV# is low. • During synchronous PSRAM read and synchronous write modes, the address is either latched on the first rising clock edge after ADV# assertion or on the rising edge of ADV# whichever edge comes first. In asynchronous read and asynchronous write modes, ADV# can be used to latch the address, but can be held low for the entire operation as well. ***Note:*** During A/D-Mux I/O operation, ADV# must remain deasserted during the data phase.	
F[3:1]-CE#	Input	**FLASH CHIP ENABLE:** Flash-specific signal; low-true input. When low, F-CE# selects the associated flash memory die. When high, F-CE# deselects the associated flash die. Flash die power is reduced to standby levels, and its data and F-WAIT outputs are placed in a High-Z state. • F1-CE# is dedicated to flash die #1. • F[3:2]-CE# are dedicated to flash die #3 through #2, respectively, if present. Otherwise, any unused flash chip enable should be treated as RFU.	
CLK	Input	**CLOCK:** Flash- and Synchronous PSRAM-specific input signal. CLK synchronizes the flash and/or synchronous PSRAM with the system clock during synchronous operations.	
F[2:1]-OE#	Input	**FLASH OUTPUT ENABLE:** Flash-specific signal; low-true input. When low, F-OE# enables the output drivers of the selected flash die. When high, F-OE# disables the output drivers of the selected flash die and places the output drivers in High-Z. • F2-OE# common to all other flash dies, if present. Otherwise it is an RFU, however, it is highly recommended to always common F1-OE# and F2-OE# on the PCB.	
R-OE#	Input	**RAM OUTPUT ENABLE:** PSRAM- and SRAM-specific signal; low-true input. When low, R-OE# enables the output drivers of the selected memory die. When high, R-OE# disables the output drivers of the selected memory die and places the output drivers in High-Z. If device not present, treat as RFU.	2
F-RST#	Input	**FLASH RESET:** Flash-specific signal; low-true input. When low, F-RST# resets internal operations and inhibits writes. When high, F-RST# enables normal operation.	

Table 2: Signal Descriptions (Sheet 2 of 3)

Symbol	Type	Signal Descriptions	Notes
WAIT	Output	**WAIT:** Flash -and Synchronous PSRAM-specific signal; configurable true-level output. When asserted, WAIT indicates invalid output data. When deasserted, WAIT indicates valid output data. • WAIT is driven whenever the flash or the synchronous PSRAM is selected and its output enable is low. • WAIT is High-Z whenever flash or the synchronous PSRAM is deselected, or its output enable is high.	
F-WE#	Input	**FLASH WRITE ENABLE:** Flash-specific signal; low-true input. When low, F-WE# enables Write operations for the enabled flash die. Address and data are latched on the rising edge of F-WE#.	
R-WE#	Input	**RAM WRITE ENABLE:** PSRAM- and SRAM-specific signal; low-true input. When low, R-WE# enables Write operations for the selected memory die. Data is latched on the rising edge of R-WE#. If device not present, treat as RFU.	2
F-WP#	Input	**FLASH WRITE PROTECT:** Flash-specific signals; low-true inputs. When low, F-WP# enables the Lock-Down mechanism. When high, F-WP# overrides the Lock-Down function, enabling locked-down blocks to be unlocked with the Unlock command. • F-WP1# is dedicated to flash die #1. • F-WP2# is common to all other flash dies, if present. Otherwise it is an RFU.	
P-CRE	Input	**PSRAM CONTROL REGISTER ENABLE:** Synchronous PSRAM-specific signal; high-true input. When high, P-CRE enables access to the Refresh Control Register (P-RCR) or Bus Control Register (P-BCR). When low, P-CRE enables normal Read or Write operations. If PSRAM not present, treat as RFU.	3
P-MODE#	Input	**PSRAM MODE#:** Asynchronous only PSRAM-specific signal; low-true input. When low, P-MODE# enables access to the configuration register, and to enter or exit Low-Power mode. When high, P-MODE# enables normal Read or Write operations. If PSRAM not present, treat as RFU.	3
P[2:1]-CS#	Input	**PSRAM CHIP SELECT:** PSRAM-specific signal; low-true input. When low, P-CS# selects the associated PSRAM memory die. When high, P-CS# deselects the associated PSRAM die. PSRAM die power is reduced to standby levels, and its data and WAIT outputs are placed in a High-Z state. • P1-CS# is dedicated to PSRAM die #1. If PSRAM not present, treat as RFU. • P2-CS# is dedicated to PSRAM die #2. If PSRAM not present, treat as RFU.	
S-CS1# S-CS2	Input	**SRAM CHIP SELECTS:** SRAM-specific signals; S-CS1# low-true input, S-CS2 high-true input. When both S-CS1# and S-CS2 are asserted, the SRAM die is selected. When either S-CS1# or S-CS2 is deasserted, the SRAM die is deselected. • S-CS1# and S-CS2 are dedicated to SRAM when present. If SRAM not present, treat as RFU.	2
R-UB# R-LB#	Input	**RAM UPPER/LOWER BYTE ENABLES:** PSRAM- and SRAM-specific signals; low-true inputs. When low, R-UB# enables DQ[15:8] and R-LB# enables DQ[7:0] during PSRAM or SRAM Read and Write cycles. When high, R-UB# masks DQ[15:8] and R-LB# masks DQ[7:0]. If device not present, treat as RFU./	2
Power Signals			
F-VPP	Power	**FLASH PROGRAM/ERASE VOLTAGE:** Flash specific. F-VPP supplies program or erase power to the flash die.	
F[2:1]-VCC	Power	**FLASH CORE POWER SUPPLY:** Flash specific. F[2:1]-VCC supplies the core power to the flash die. F2-VCC is recommended to be tied to F1-VCC, else it is an RFU.	
VCCQ	Power	**I/O POWER SUPPLY:** Global device I/O power. VCCQ supplies the device input/output driver voltage.	
P-VCC	Power	**PSRAM CORE POWER SUPPLY:** PSRAM specific. P-VCC supplies the core power to the PSRAM die. If PSRAM not present, treat as RFU.	2
S-VCC	Power	**SRAM POWER SUPPLY:** SRAM specific. S-VCC supplies the core power to the SRAM die. If SRAM not present, treat as RFU.	2

Table 2: Signal Descriptions (Sheet 3 of 3)

Symbol	Type	Signal Descriptions	Notes
VSS	Ground	**DEVICE GROUND:** Global ground reference for all signals and power supplies. Connect all VSS balls to system ground. Do not float any VSS connections.	
DU	—	**DO NOT USE:** This ball should not be connected to any power supplies, signals, or other balls. This ball can be left floating.	
RFU	—	RESERVED for FUTURE USE: Reserved by Intel for future device functionality and enhancement. This ball must be left floating.	

Notes:
1. Only used when AD-Mux I/O flash is present.
2. Only available on stacked device combinations with PSRAM, and/or SRAM die. Otherwise treated as RFU.
3. P-CRE and P-MODE# share the same package ball at location K8. Only one signal function is available, depending on the stacked device combination.

4.2 Memory Map

The 64Mb and 128Mb memory array is divided into multiple 8-Mbit partitions. Each device density contains one parameter partition and several main partitions. The 8-Mbit top parameter partition contains four 16K-Word blocks and seven 64K-Word blocks. There are multiple 8-Mbit main partitions. The 8-Mbit main partitions each contains eight 64K-Word blocks.

The device multi-partition architecture is divided as follow:

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

The 256Mb memory array is divided into multiple 16-Mbit partitions. Each device contains one parameter partition and fifteen main partitions. The 16-Mbit top parameter partition contains four 16K-Word blocks and fifteen 64K-Word blocks. There are fifteen 16-Mbit main partitions. The 16-Mbit main partitions each contains sixteen 64K-Word blocks.

Table 3: Top Parameter Memory Map, 128-Mbit (Sheet 1 of 2)

		Size (KW)	Blk	64-Mbit
			43	
8-Mbit Parameter Partition	One Partition	16	66	3FC000-3FFFFF
		16	65	3F8000-3FBFFF
		16	64	3F4000-3F7FFF
		16	63	3F0000-3F3FFF
		64	62	3E0000-3EFFFF
		⋮	⋮	⋮
		64	56	380000-38FFFF

		Size (KW)	Blk	128-Mbit
8-Mbit Parameter Partition	One Partition	16	130	7FC000-7FFFFF
		16	129	7F8000-7FBFFF
		16	128	7F4000-7F7FFF
		16	127	7F0000-7F3FFF
		64	126	7E0000-7EFFFF
		⋮	⋮	⋮
		64	120	780000-78FFFF

Table 3: Top Parameter Memory Map, 128-Mbit (Sheet 2 of 2)

		Size (KW)	Blk	64-Mbit
			43	
8-Mbit Main Partition	Seven Partitions	64	55	370000-37FFFF
		⋮	⋮	⋮
		64	0	000000-00FFFF

		Size (KW)	Blk	128-Mbit
8-Mbit Main Partitions	Fifteen Partitions	64	119	770000-77FFFF
		⋮		
		64	0	000000-00FFFF
	Sixteen Partitions	64		
		⋮		
		64		

Table 4: Top Parameter Memory Map, 256-Mbit

		Size (KW)	Blk	256-Mbit
16-Mbit Parameter Partition	One Partition	16	258	FFC000-FFFFFF
		16	257	FF8000-FFBFFF
		16	256	FF4000-FF7FFF
		16	253	FF0000-FF3FFF
		64	254	FE0000-FEFFFF
		⋮	⋮	⋮
		64	240	F00000-FFFFFF
16-Mbit Main Partitions	Seven Partitions	64	239	EF0000-EFFFFF
		⋮		
		64	128	800000-80FFFF
	Eight Partitions	64	127	7F0000-7FFFFF
		⋮		
		64	0	000000-00FFFF

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.

Table 5: Absolute Maximum Ratings Table

Parameter	Maximum Rating	Notes
Temperature under bias	–25 °C to +85 °C	
Storage temperature	–65 °C to +125 °C	
Voltage on any signal (except V_{CC}, VPP)	–0.5 V to +2.5 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 +2.5 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 +10.0 V for periods <20 ns.
3. Program/erase voltage is typically 1.7 V to 2.0 V. 9 V can be applied for 80 hours maximum total, to any blocks for 1000 cycles maximum. 9 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

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

Table 6: Operating Conditions Table

Symbol	Parameter		Min	Max	Units	Notes
T_C	Operating Temperature		–25	+85	°C	
V_{CC}	V_{CC} Supply Voltage		1.7	2.0	V	
V_{CCQ}	I/O Supply Voltage	1.8 V Range	1.7	2.0	V	
		1.8 V Extended Range	1.35	2.0	V	
V_{PPL}	V_{PP} Voltage Supply (Logic Level)		0.9	2.0	V	
V_{PPH}	Factory programming V_{PP}		8.5	9.5		
t_{PPH}	Maximum V_{PP} Hours	$V_{PP} = V_{PPH}$		80	Hours	1
Block Erase Cycles	Main and Parameter Blocks	$V_{PP} = V_{CC}$	100,000		Cycles	
	Main Blocks	$V_{PP} = V_{PPH}$		1000		
	Parameter Blocks	$V_{PP} = V_{PPH}$		2500		

Notes:
1. T_C = Case Temperature
2. In typical operation, the VPP program voltage is V_{PPL}. VPP can be connected to 8.5 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

Table 7: DC Current Characteristics (Sheet 1 of 2)

Sym	Parameter		V_{CC}	1.7 V – 2.0 V		Unit	Test Conditions	Notes
			V_{CCQ}	1.7 V – 2.0 V 1.35 V – 2.0 V				
				Typ	Max			
I_{LI}	Input Load Current				±1	µA	V_{CC} = V_{CC} MAX V_{CCQ} = V_{CCQ} Max V_{IN} = V_{CCQ} or GND	1
I_{LO}	Output Leakage Current	AD[15:0], WAIT			±1	µA	V_{CC} = V_{CC} MAX V_{CCQ} = V_{CCQ} Max V_{IN} = V_{CCQ} or GND	
I_{CCS} I_{CCD}	V_{CC} Standby, Power Down	64 Mbit		15	30	µA	V_{CC} = V_{CC}Max V_{CCQ} = V_{CCQ}Max CE# = V_{CCQ} RST# = V_{CCQ} (for I_{CCS}) RST# = GND (for I_{CCD}) WP# = V_{IH}	1,2
		128 Mbit		20	70			
		256 Mbit		25	110			
I_{CCAPS}	APS	64 Mbit		15	30	µA	V_{CC} = V_{CC} MAX V_{CCQ} = V_{CCQ} Max CE# = V_{SSQ} RST# = V_{CCQ} All inputs are at rail to rail (V_{CCQ} or V_{SSQ}).	1
		128 Mbit		20	70			
		256 Mbit		25	110			
I_{CCR}	Average V_{CC} Read Current	Asynchronous Single-Word f = 5MHz (1 CLK)		13	15	mA		
		Synchronous Burst Read f = 40MHz, LC = 3		12	16	mA	Burst length=4	
				14	18	mA	Burst length=8	
				16	20	mA	Burst length=16	
				20	25	mA	Burst length = Continuous	
		Synchronous Burst Read f = 54MHz, LC = 4		15	18	mA	Burst length=4	
				18	22	mA	Burst length=8	
				21	25	mA	Burst length=16	
				22	27	mA	Burst Length = Continuous	
							V_{CC} = V_{CC}MAX CE# = V_{IL} OE# = V_{IH} Inputs: V_{IL} or V_{IH}	
I_{CCW}, I_{CCE}	V_{CC} Program Current, V_{CC} Erase Current			35	50	mA	V_{PP} = V_{PPL}, program/erase in progress	1,3,4,7
				25	32	mA	V_{PP} = V_{PPH}, program/erase in progress	1,3,5,7
I_{CCWS}, I_{CCES}	V_{CC} Program Suspend Current, V_{CC} Erase Suspend Current	64 Mbit		15	30	µA	CE# = V_{CCQ}; suspend in progress	1,6,3
		128 Mbit		20	70			
		256 Mbit		25	110			
I_{PPS}, I_{PPWS}, I_{PPES}	V_{PP} Standby Current, V_{PP} Program Suspend Current, V_{PP} Erase Suspend Current			0.2	5	µA	V_{PP} = V_{PPL}, suspend in progress	1,3

Table 7: DC Current Characteristics (Sheet 2 of 2)

Sym	Parameter	V_{CC}	1.7 V – 2.0 V		Unit	Test Conditions	Notes
		V_{CCQ}	1.7 V – 2.0 V 1.35 V – 2.0 V				
			Typ	Max			
I_{PPR}	V_{PP} Read		2	15	µA	$V_{PP} \le V_{CC}$	1,3
I_{PPW}	V_{PP} Program Current		0.05	0.10	mA	$V_{PP} = V_{PPL}$, program in progress	
			8	22		$V_{PP} = V_{PPH}$, program in progress	
I_{PPE}	V_{PP} Erase Current		0.05	0.10	mA	$V_{PP} = V_{PPL}$, erase in progress	
			8	22		$V_{PP} = V_{PPH}$, erase in progress	

Notes:
1. All currents are RMS unless noted. Typical values at typical V_{CC}, T_C = +25°C.
2. I_{CCS} is the average current measured over any 5 ms time interval 5 µs after CE# is deasserted.
3. Sampled, not 100% tested.
4. V_{CC} read + program current is the sum of V_{CC} read and V_{CC} program currents.
5. V_{CC} read + erase current is the sum of V_{CC} read and V_{CC} erase currents.
6. I_{CCES} is specified with the device deselected. If device is read while in erase suspend, current is I_{CCES} plus I_{CCR}
7. I_{CCW}, I_{CCE} measured over typical or max times specified in Section 7.7, "Program and Erase Characteristics" on page 32.

6.2 DC Voltage Characteristics

Table 8: DC Voltage Characteristics

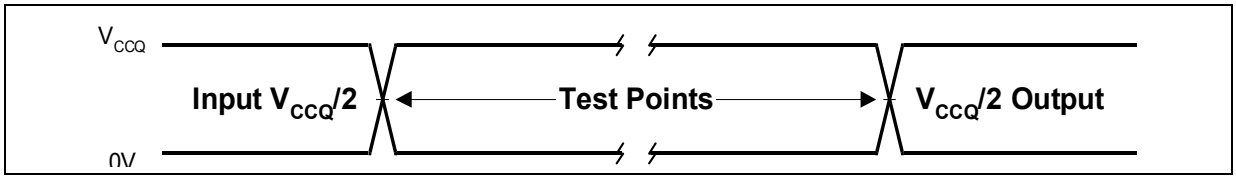
Sym	Parameter	V_{CCQ}	1.35 V – 2.0 V		1.7 V – 2.0 V		Unit	Test Condition	Notes
			Min	Max	Min	Max			
V_{IL}	Input Low Voltage		0	0.2	0	0.4	V		1
V_{IH}	Input High Voltage		V_{CCQ} –0.2	V_{CCQ}	V_{CCQ} –0.4	V_{CCQ}	V		
V_{OL}	Output Low Voltage			0.1		0.1	V	$V_{CC} = V_{CC}MIN$ $V_{CCQ} = V_{CCQ}MIN$ I_{OL} = 100 µA	
V_{OH}	Output High Voltage		V_{CCQ} –0.1		V_{CCQ} –0.1		V	$V_{CC} = V_{CC}MIN$ $V_{CCQ} = V_{CCQ}MIN$ I_{OH} = –100 µA	
V_{PPLK}	V_{PP} Lock-Out Voltage			0.4		0.4	V		2
V_{LKO}	V_{CC} Lock Voltage		1.0		1.0		V		
V_{LKOQ}	V_{CCQ} Lock Voltage		0.9		0.9		V		

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

7.0 AC Characteristics

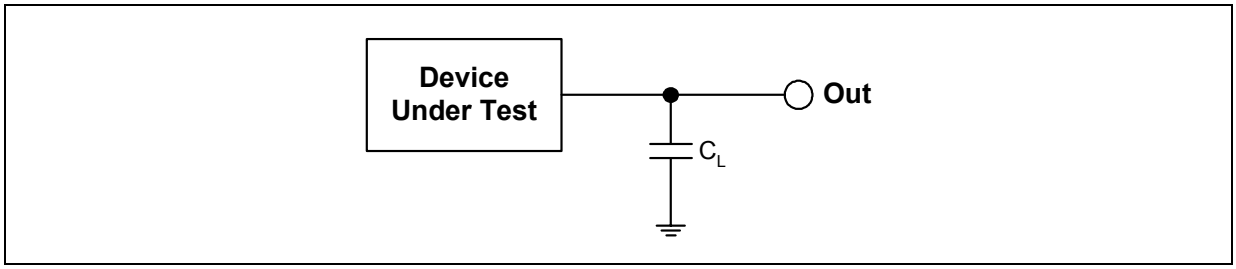
7.1 AC Test Conditions

Figure 3: AC Input/Output Reference Waveform



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

Figure 4: Transient Equivalent Testing Load Circuit

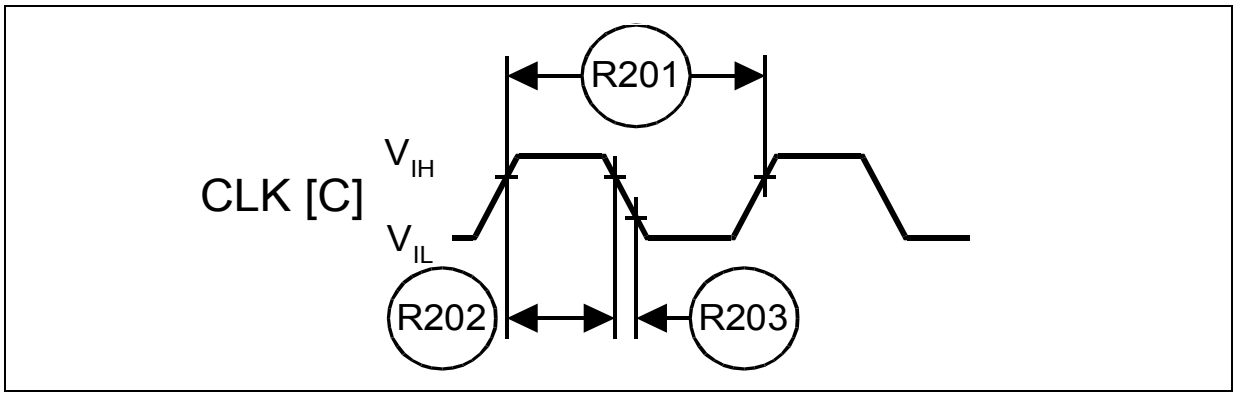


Notes:
1. See the following table for component values.
2. Test configuration component value for worst case speed conditions.
3. C_L includes jig capacitance.

Table 9: Test Configuration

Test Configuration	C_L (pF)
V_{CCQ} Min Standard Test	30

Figure 5: Clock Input AC Waveform



7.2 Capacitance

Table 10: Capacitance

Symbol	Parameter	Signals	Min	Typ	Max	Unit	Condition	Note
C_{IN}	Input Capacitance	Address, CE#, WE#, OE#, RST#, CLK, ADV#, WP#	2	6	7	pF	Typ temp= 25 °C, Max temp = 85 °C, V_{CC}=V_{CCQ}=(0-1.95) V, Silicon die	1,2
C_{OUT}	Output Capacitance	Data, WAIT	2	4	5	pF		

Notes:
1. Sampled, not 100% tested.
2. Silicon die capacitance only, add 1 pF for discrete packages.

7.3 AC Read Specifications (V_{CCQ} = 1.35 V – 2.0 V)

Table 11: AC Read Specifications (V_{CCQ} = 1.35 V – 2.0 V) (Sheet 1 of 2)

Num	Symbol	Parameter	All Densities Speed	-90		Units	Notes
				Min	Max		
Asynchronous Specifications							
R1	t_{AVAV}	Read cycle time		90		ns	1,6
R2	t_{AVQV}	Address to output valid			90	ns	
R3	t_{ELQV}	CE# low to output valid			90	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			20	ns	1,3
R9	t_{GHQZ}	OE# high to output in high-Z			20	ns	
R10	t_{OH}	Output hold from first occurring address, CE#, or OE# change		0		ns	
R11	t_{EHEL}	CE# pulse width high		17		ns	1
R12	t_{ELTV}	CE# low to WAIT valid			17	ns	1
R13	t_{EHTZ}	CE# high to WAIT high Z			17	ns	1,3
R14	t_{GHTV}	OE# high to WAIT Valid			17	ns	
R15	t_{GLTV}	OE# low to WAIT Valid			17	ns	
Latching Specifications							
R101	t_{AVVH}	Address setup to ADV# high		7		ns	1
R102	t_{ELVH}	CE# low to ADV# high		10		ns	
R103	t_{VLQV}	ADV# low to output valid			90	ns	
R104	t_{VLVH}	ADV# pulse width low		7		ns	
R105	t_{VHVL}	ADV# pulse width high		7		ns	
R106	t_{VHAX}	Address hold from ADV# high		7		ns	1,4
R107	t_{VHGL}	ADV# high to OE# low		7		ns	1
R111	t_{PHVH}	RST# high to ADV# high		30		ns	1
Clock Specifications							

Table 11: AC Read Specifications (V_{CCQ} = 1.35 V – 2.0 V) (Sheet 2 of 2)

Num	Symbol	Parameter	All Densities Speed	-90		Units	Notes
				Min	Max		
R200	f_{CLK}	CLK frequency			47	MHz	1,3
R201	t_{CLK}	CLK period		21.3		ns	
R202	$t_{CH/CL}$	CLK high/low time		4.5		ns	
R203	$t_{FCLK/RCLK}$	CLK fall/rise time			3	ns	
Synchronous Specifications							
R301	$t_{AVCH/L}$	Address setup to CLK		7		ns	1
R302	$t_{VLCH/L}$	ADV# low setup to CLK		7		ns	
R303	$t_{ELCH/L}$	CE# low setup to CLK		7		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		7		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

7.4 AC Read Specifications: 64- and 128-Mb Densities

Table 12: AC Read Specifications (V_{CCQ} = 1.7 V – 2.0 V) (Sheet 1 of 2)

Num	Symbol	Parameter	All Densities Speed	-85		Units	Notes
				Min	Max		
Asynchronous Specifications							
R1	t_{AVAV}	Read cycle time		85		ns	1,6
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			20	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			17	ns	1,3
R9	t_{GHQZ}	OE# high to output in high-Z			17	ns	
R10	t_{OH}	Output hold from first occurring address, CE#, or OE# change		0		ns	
R11	t_{EHEL}	CE# pulse width high		14		ns	1
R12	t_{ELTV}	CE# low to WAIT valid			14	ns	
R13	t_{EHTZ}	CE# high to WAIT high Z			14	ns	1,3
R14	t_{GHTV}	OE# high to WAIT Valid			14	ns	1
R15	t_{GLTV}	OE# low to WAIT Valid			14	ns	
Latching Specifications							

Table 12: AC Read Specifications (V_{CCQ} = 1.7 V – 2.0 V) (Sheet 2 of 2)

Num	Symbol	Parameter	All Densities Speed	-85		Units	Notes
				Min	Max		
R101	t_{AVVH}	Address setup to ADV# high		7		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		7		ns	
R105	t_{VHVL}	ADV# pulse width high		7		ns	
R106	t_{VHAX}	Address hold from ADV# high		7		ns	1,4
R107	t_{VHGL}	ADV# high to OE# low		7		ns	1
R111	t_{phvh}	RST# high to ADV# high		30		ns	1
Clock Specifications							
R200	f_{CLK}	CLK frequency			54	MHz	1,3
R201	t_{CLK}	CLK period		18.5		ns	
R202	$t_{CH/CL}$	CLK high/low time		3.5		ns	
R203	$t_{FCLK/RCLK}$	CLK fall/rise time			3	ns	
Synchronous Specifications							
R301	$t_{AVCH/L}$	Address setup to CLK		7		ns	1
R302	$t_{VLCH/L}$	ADV# low setup to CLK		7		ns	
R303	$t_{ELCH/L}$	CE# low setup to CLK		7		ns	
R304	$t_{CHQV\ /\ tCLQV}$	CLK to output valid			14	ns	
R305	t_{CHQX}	Output hold from CLK		3		ns	1,5
R306	t_{CHAX}	Address hold from CLK		7		ns	1,4,5
R307	t_{CHTV}	CLK to WAIT valid			14	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 3, "AC Input/Output Reference Waveform" on page 20 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.
6. The specifications in Table 11 will **only** be used by customers (1) who desire a 1.35 to 2.0 V_{CCQ} operating range OR (2) who desire to transition their host controller from a 1.7 V to 2.0 V V_{CCQ} voltage now to a lower range in the future.

7.5 AC Read Specifications for 256-Mb Density

Table 13: AC Read Specifications (Sheet 1 of 3)

Num	Symbol	Parameter	All Densities Speed	-90		Units	Notes
				Min	Max		
Asynchronous Specifications							

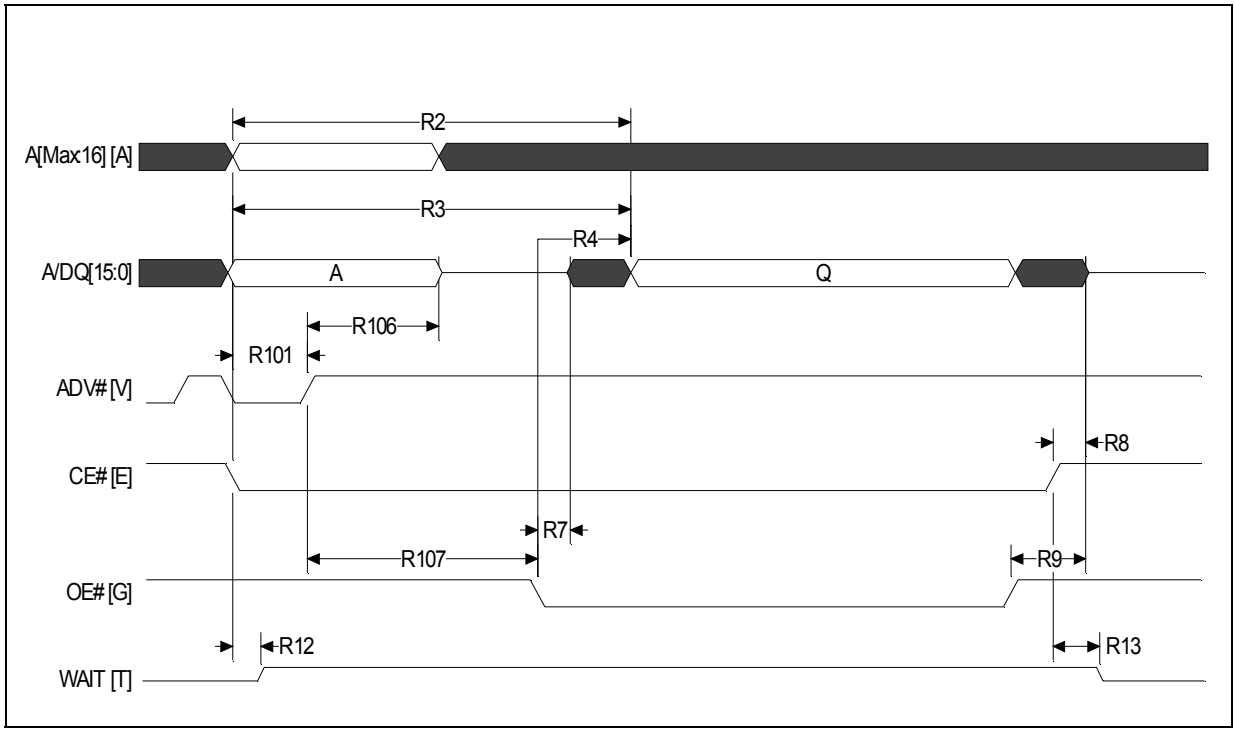
Table 13: AC Read Specifications (Sheet 2 of 3)

Num	Symbol	Parameter	All Densities Speed	-90		Units	Notes
				Min	Max		
R1	t_{AVAV}	Read cycle time	$V_{CC} = V_{CCQ}$ = 1.8 V – 2.0	85		ns	1,6
			$V_{CC} = V_{CCQ}$ = 1.7 V – 2.0	88			
R2	t_{AVQV}	Address to output valid	$V_{CC} = V_{CCQ}$ = 1.8 V – 2.0		85	ns	
			$V_{CC} = V_{CCQ}$ = 1.7 V – 2.0		88		
R3	t_{ELQV}	CE# low to output valid	$V_{CC} = V_{CCQ}$ = 1.8 V – 2.0		85	ns	
			$V_{CC} = V_{CCQ}$ = 1.7 V – 2.0		88		
R4	t_{GLQV}	OE# low to output valid			20	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			17	ns	1,3
R9	t_{GHQZ}	OE# high to output in high-Z			17	ns	
R10	t_{OH}	Output hold from first occurring address, CE#, or OE# change		0		ns	
R11	t_{EHEL}	CE# pulse width high		14		ns	1
R12	t_{ELTV}	CE# low to WAIT valid			14	ns	
R13	t_{EHTZ}	CE# high to WAIT high Z			14	ns	1,3
R14	t_{GHTV}	OE# high to WAIT Valid			14	ns	1
R15	t_{GLTV}	OE# low to WAIT Valid			14	ns	
Latching Specifications							
R101	t_{AVVH}	Address setup to ADV# high		7		ns	1
R102	t_{ELVH}	CE# low to ADV# high		10		ns	
R103	t_{VLQV}	ADV# low to output valid	$V_{CC} = V_{CCQ}$ = 1.8 V – 2.0		85	ns	
			$V_{CC} = V_{CCQ}$ = 1.7 V – 2.0		88		
R104	t_{VLVH}	ADV# pulse width low		7		ns	
R105	t_{VHVL}	ADV# pulse width high		7		ns	
R106	t_{VHAX}	Address hold from ADV# high		7		ns	1,4
R107	t_{VHGL}	ADV# high to OE# low		7		ns	1
R111	t_{phvh}	RST# high to ADV# high		30		ns	1
Clock Specifications							
R200	f_{CLK}	CLK frequency			54	MHz	1,3
R201	t_{CLK}	CLK period		18.5		ns	
R202	$t_{CH/CL}$	CLK high/low time		3.5		ns	
R203	$t_{FCLK/RCLK}$	CLK fall/rise time			3	ns	
Synchronous Specifications							
R301	$t_{AVCH/L}$	Address setup to CLK		7		ns	1
R302	$t_{VLCH/L}$	ADV# low setup to CLK		7		ns	
R303	$t_{ELCH/L}$	CE# low setup to CLK		7		ns	
R304	$t_{CHQV\ /\ tCLQV}$	CLK to output valid			14	ns	

Table 13: AC Read Specifications (Sheet 3 of 3)

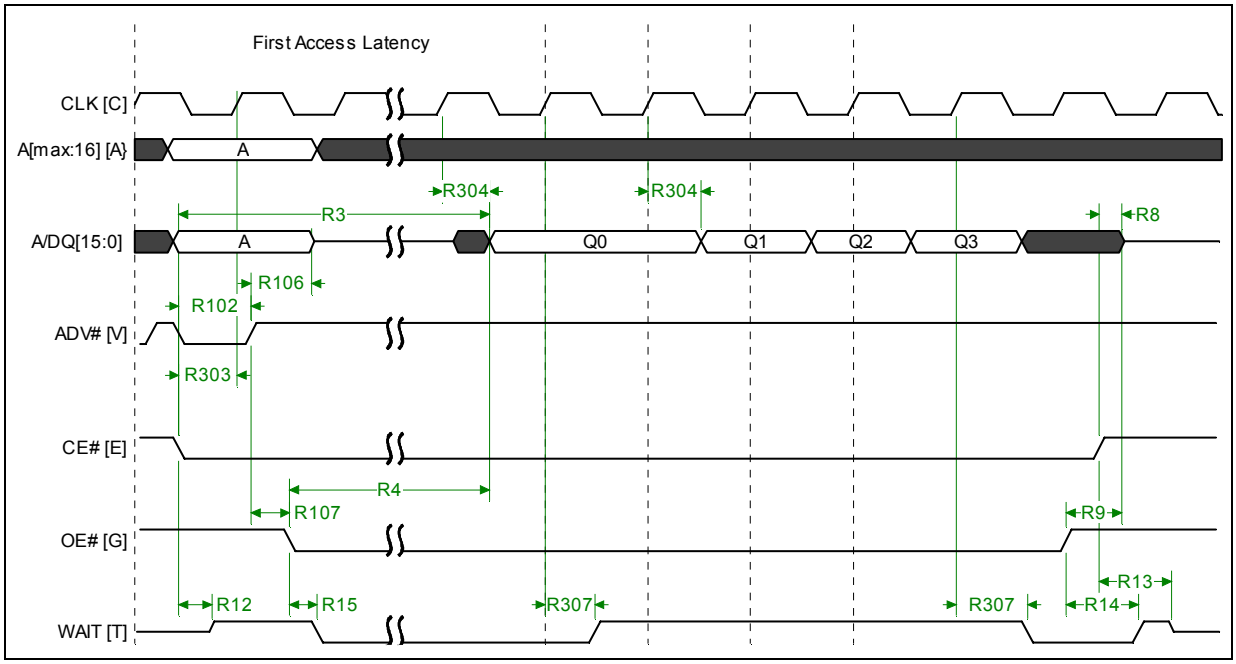
Num	Symbol	Parameter	All Densities Speed	-90		Units	Notes
				Min	Max		
R305	t_{CHQX}	Output hold from CLK		3		ns	1,5
R306	t_{CHAX}	Address hold from CLK		7		ns	1,4,5
R307	t_{CHTV}	CLK to WAIT valid			14	ns	1,5
R311	t_{CHVL}	CLK Valid to ADV# Setup		0		ns	1
R312	t_{CHTX}	WAIT Hold from CLK		3		ns	1,5

Figure 6: Asynchronous Single-Word Read Timing



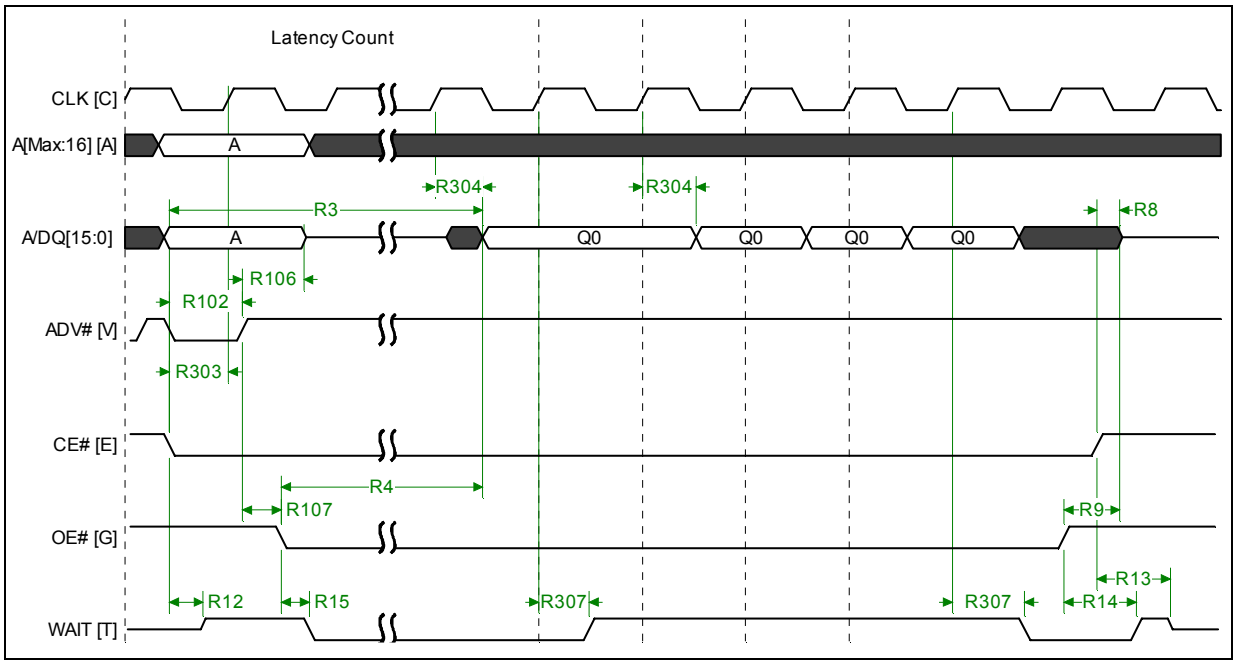
Note: WAIT is deasserted (CR [10] = 0) during asynchronous read mode.

Figure 7: Synchronous Array Read with Flow-through Feature Timing



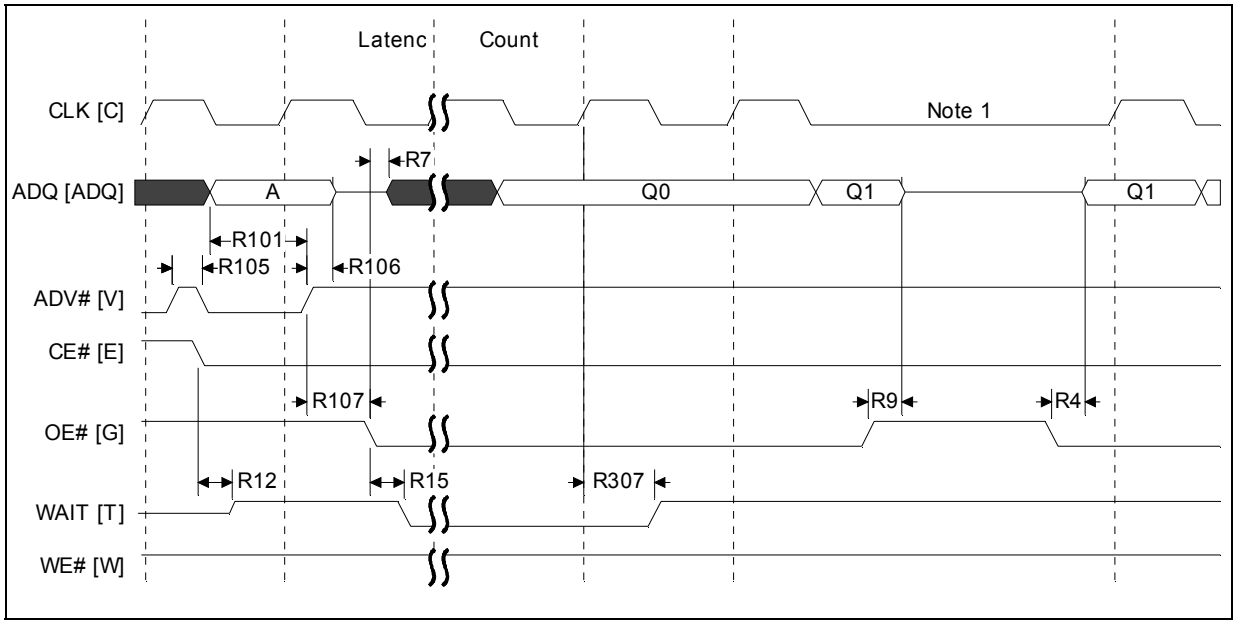
Notes:
1. WAIT active low (asserted) during initial access and deasserted during valid read array data
2. WAIT deasserted during OE# = Vih.
3. Flow through feature as shown during the first data word.

Figure 8: Synchronous Non-Array Read with Flow-through Feature Timing



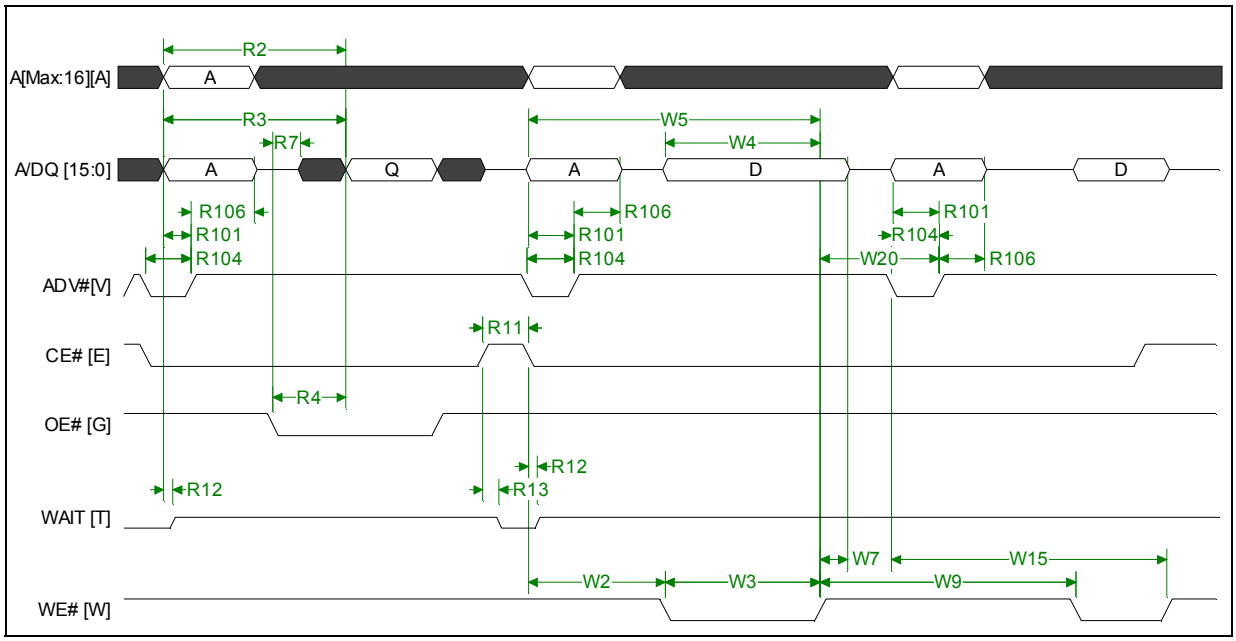
Notes:
1. WAIT active low (asserted) during initial access and deasserted during valid read non-array data
2. WAIT deasserted during OE# = Vih
3. Flow through feature as shown during the first data word.

Figure 9: Burst Suspend Timing



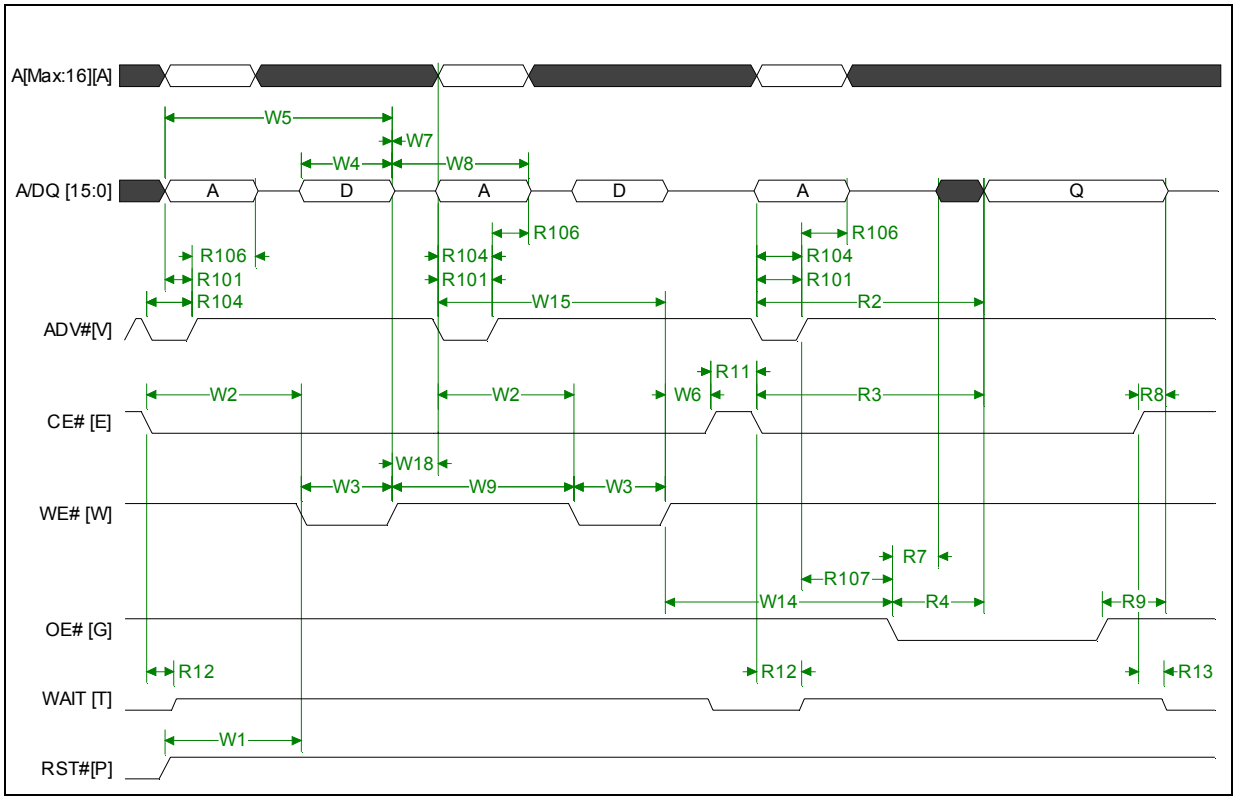
Notes:
1. During burst suspend Clock signal can be held high or low
2. WAIT asserted low (CR[10] = 0).

Figure 10: Asynchronous Read to Write Timing



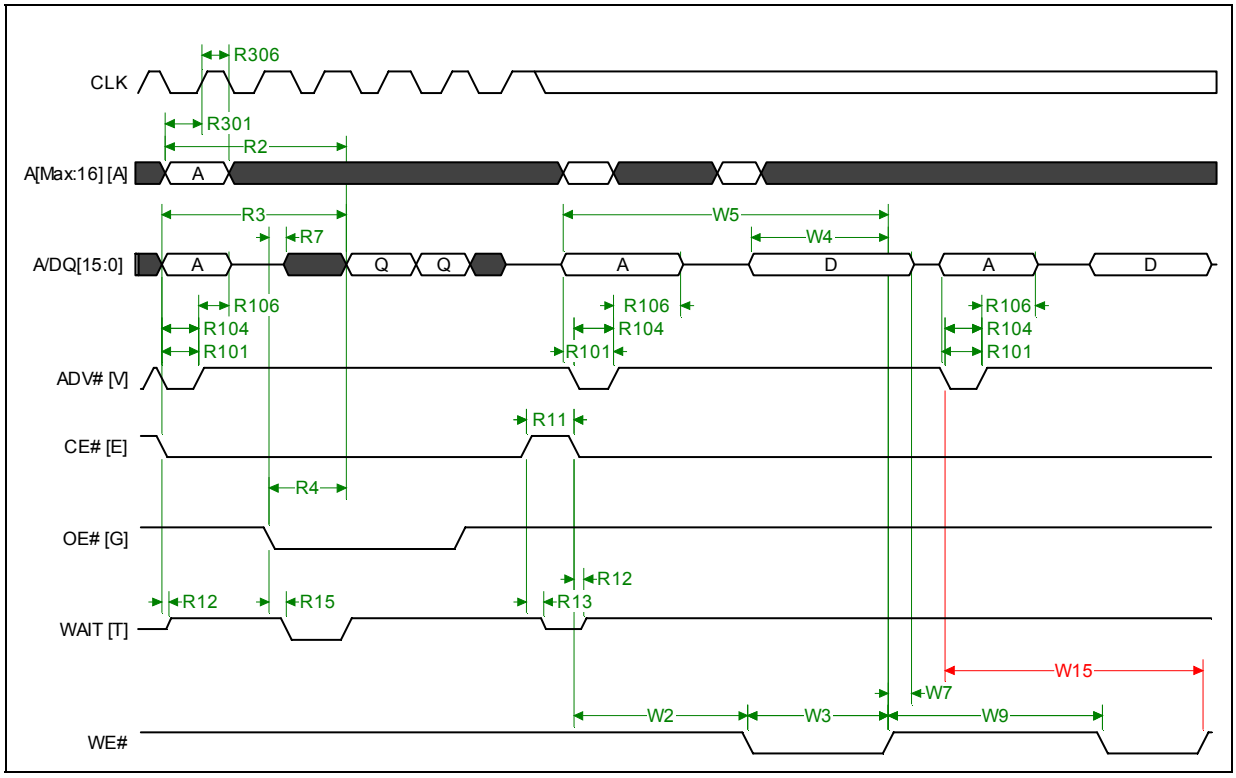
Note: WAIT deasserted (CR[10] = 0) during asynchronous operations.

Figure 11: Write to Asynchronous Read Timing



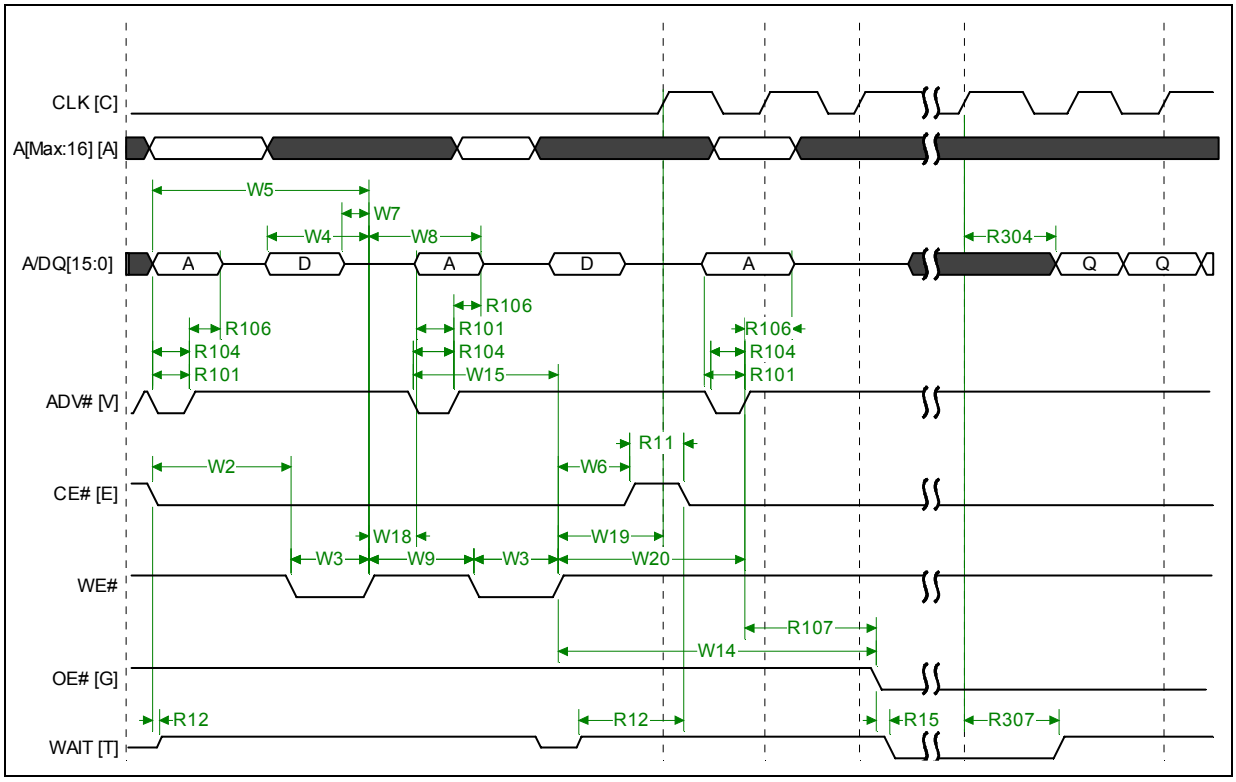
Note: WAIT deasserted (CR[10] = 0) during asynchronous operations.

Figure 12: Synchronous Read to Write Timing



Note: WAIT shown deasserted (CR[10] = 0) during write operation.

Figure 13: Write to Synchronous Read



Note: WAIT shown deasserted (CR[10] = 0) during write operation.

7.6 AC Write Specifications

Table 14: AC Write Specifications (Sheet 1 of 2)

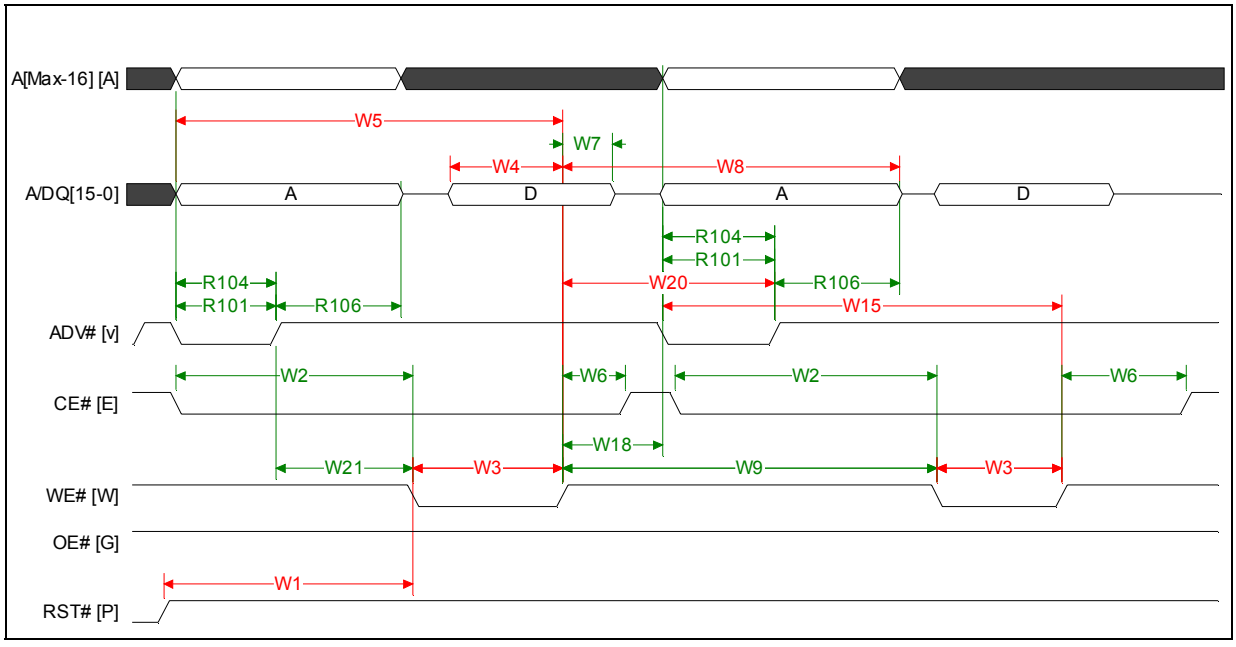
Num	Symbol	Parameter	Min	Max	Units	Notes
W1	t_{PHWL}	RST# high recovery to WE# low	150		ns	1,2,3
W2	t_{ELWL}	CE# setup to WE# low	0		ns	1,2,3
W3	t_{WLWH}	WE# write pulse width low	50		ns	1,2,4
W4	t_{DVWH}	Data setup to WE# high	50		ns	1,2
W5	t_{AVWH}	Address setup to WE# high	50		ns	
W6	t_{WHEH}	CE# hold from WE# high	0		ns	
W7	t_{WHDX}	Data hold from WE# high	0		ns	
W8	t_{WHAX}	Address hold from WE# high	0		ns	
W9	t_{WHWL}	WE# pulse width high	20		ns	1,2,5
W10	t_{VPWH}	V_{PP} setup to WE# high	200		ns	1,2,3,7
W11	t_{QVVL}	V_{PP} hold from Status read	0		ns	
W12	t_{QVBL}	WP# hold from Status read	0		ns	1,2,3,7
W13	t_{BHWH}	WP# setup to WE# high	200		ns	
W14	t_{WHGL}	WE# high to OE# low	0		ns	1,2,9

Table 14: AC Write Specifications (Sheet 2 of 2)

Num	Symbol	Parameter	Min	Max	Units	Notes
W15	t_{VLWH}	ADV# low to WE# high	69		ns	1,2
W16	t_{WHQV}	WE# high to read valid	t_{AVQV}+35		ns	1,2,3,6, 10
Write to Asynchronous Read Specification						
W18	t_{WHAV}	WE# high to Address valid	0		ns	1,2,3,6, 10
Write to Synchronous Read Specification						
W19	$t_{WHCH/L}$	WE# high to Clock valid	19		ns	1,2,3,6, 10
W20	t_{WHVH}	WE# high to ADV# high	19		ns	
Write Specifications with Clock Active						
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:
1. Write timing characteristics during erase suspend are the same as write-only operations.
2. A write operation can be terminated with either CE# or WE#.
3. Sampled, not 100% tested.
4. Write pulse width low (t_{WLWH} or t_{ELEH}) is defined from CE# or WE# low (whichever occurs last) to CE# or WE# high (whichever occurs first). Hence, $t_{WLWH} = t_{ELEH} = t_{WLEH} = t_{ELWH}$.
5. Write pulse width high (t_{WHWL} or t_{EHEL}) is defined from CE# or WE# high (whichever occurs first) to CE# or WE# low (whichever occurs last). Hence, $t_{WHWL} = t_{EHEL} = t_{WHEL} = t_{EHW}$).
6. t_{WHVH} or $t_{WHCH/L}$ must be met when transitioning from a write cycle to a synchronous burst read.
7. V_{PP} should be at a valid level until erase or program success is determined.
8. This specification is only applicable when transitioning from a write cycle to an asynchronous read. See specs W19 and W20 for synchronous read.
9. When doing a Read Status operation following any command that alters the Status Register, W14 is 20 ns.
10. Add 10ns if the write operation results in a RCR or block lock status change, for the subsequent read operation to reflect this change.
11. These specs are required only when the device is in a synchronous mode and clock is active during address setup phase.

Figure 14: Write Timing



7.7 Program and Erase Characteristics

Table 15: Program and Erase Characteristics

Num	Symbol	Parameter		V_{PPL}			V_{PPH}			Units	Notes
				Min	Typ	Max	Min	Typ	Max		
Word Programming											
W200	$t_{PROG/W}$	Program Time	Single word		90	180		85	170	µs	1
			Single cell		30	60		30	60		
Buffered Programming											
W200	$t_{PROG/W}$	Program Time	Single word		90	180		85	170	µs	1,2
W201	$t_{PROG/PB}$		One Buffer (32 words)		440	880		340	680		
Buffered EFP											
W400	$t_{BEFP/W}$	Program	Single word	N/A	N/A	N/A	N/A	10	N/A	µs	1,2
W452	$t_{BEFP/SETUP}$		Buffered EFP Setup	N/A	N/A	N/A	5	N/A	N/A	µs	1,2
Erasing and Suspending											
W500	$t_{ERS/PB}$	Erase Time	16-KWord Parameter		0.4	2.5		0.4	2.5	s	1
W501	$t_{ERS/MB}$		64-KWord Main		1.2	4		1.0	4		
W600	$t_{SUSP/P}$	Suspend Latency	Program suspend		20	25		20	25	µs	
W601	$t_{SUSP/E}$		Erase suspend		20	25		20	25		

Notes:
1. Typical values measured at T_C = +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 V_{CC} should attain V_{CCMIN} before applying VCCQ and VPP. Device inputs should not be driven before supply voltage equals V_{CCMIN}.

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

8.2 Reset

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

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

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

The Command User Interface (CUI) architecture provides additional protection because alteration of memory contents can only occur after successful completion of a two-step command sequence (see Section 9.2, “Device Commands” on page 37).

Figure 15: Reset Operation Waveforms

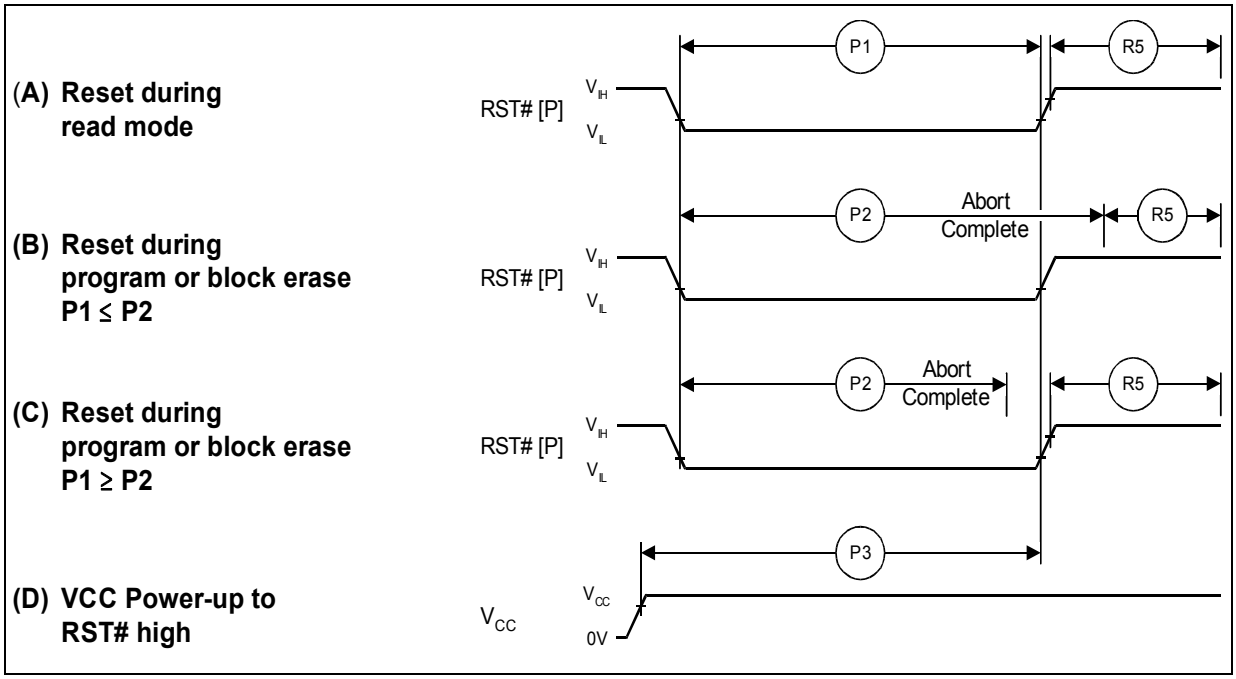


Table 16: Reset Specifications

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

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

8.3 Power Supply Decoupling

Flash memory devices require careful power supply decoupling. Three basic power supply current considerations are as follows:

1. Standby current levels
2. Active current levels
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 decoupling capacitor selection suppress transient voltage peaks.

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

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

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 µs after CE# is deasserted. During APS, average current is measured over the same time interval 5 µs after the following events happen: (1) there is no internal read, program or erase operations cease; (2) CE# is asserted; (3) the address lines are quiescent and at V_{SSQ} or V_{CCQ}. OE# may also be driven during APS.

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. 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 41 for details on the available read modes, and see Section 15.0, "Special Read States" on page 66 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 18).

9.1.2 Writes

To perform a write operation, both CE# and WE# are asserted while RST# and OE# are deasserted. All device write operations are asynchronous, with CLK being ignored. During a write operation, address and data are latched on the rising edge of WE# or CE#, whichever occurs first. Table 17, "Command Bus Cycles" on page 38 shows the bus cycle sequence for each of the supported device commands, while Table 18, "Command Codes and Definitions" on page 39 describes each command. See Section 7.0, "AC Characteristics" on page 20 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 AD[15:0] are disabled and placed in a high-impedance (High-Z) state.

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 µs after CE# is deasserted. During standby, average current is measured over the same time interval 5 µs after CE# is deasserted.

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

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. Intel® flash memories 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 20 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 17, “Command Bus Cycles” on page 38.

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 17: Command Bus Cycles

Mode	Command	Bus Cycles	First Bus Cycle			Second Bus Cycle			Notes
			Oper	Addr	Data	Oper	Addr	Data	
Read	Read Array	1	Write	PnA	0xFF				1,2
	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	> 2	Write	WA	0xE8	Write	WA	N - 1	1,2,3
	Buffered Enhanced Factory Program (Buffered EFP)	> 2	Write	WA	0x80	Write	WA	0xD0	1,2,4
Erase	Block Erase	2	Write	BA	0x20	Write	BA	0xD0	1,2
Suspend	Program/Erase Suspend	1	Write	X	0xB0				
	Program/Erase Resume	1	Write	X	0xD0				
Block Locking/ Unlocking	Lock Block	2	Write	BA	0x60	Write	BA	0x01	
	Unlock Block	2	Write	BA	0x60	Write	BA	0xD0	
	Lock-down Block	2	Write	BA	0x60	Write	BA	0x2F	
Protection	Program Protection Register	2	Write	PRA	0xC0	Write	PRA	PD	
	Program Lock Register	2	Write	LRA	0xC0	Write	LRA	LRD	
Configuration	Program Read Configuration Register	2	Write	RCD	0x60	Write	RCD	0x03	

Notes:

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

9.3 Command Definitions

Table 18 shows valid device command codes and descriptions.

Table 18: 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 AD[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 AD[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 AD[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 AD[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. 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 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 Setup	First cycle of a 2-cycle command; prepares the device to receive a variable number of bytes up to the write buffer size of 32 words. The second cycle contains the number of bytes to be transferred.
	0xD0	Buffered Program Confirm	Issued after writing all data to the write buffer; instructs the WSM to perform its Buffered Programming algorithm, writing the data from the write buffer to the flash memory array.
	0x80	Buffered Enhanced Factory Programming Setup	First cycle of a 2-cycle command; initiates Buffered Enhanced Factory Program mode (Buffered EFP). The CUI then waits for the Buffered EFP Confirm command, 0xD0, that initiates the Buffered EFP algorithm. All other commands are ignored when Buffered EFP mode begins.
	0xD0	Buffered EFP Confirm	If the previous command was Buffered EFP Setup (0x80), the CUI latches the address and data, and prepares the device for Buffered EFP mode.
Erase	0x20	Block Erase Setup	First cycle of a 2-cycle command; prepares the CUI for a block-erase operation. The WSM performs the erase algorithm on the block addressed by the Erase Confirm command. If the next command *is not* the Erase Confirm (0xD0) command, the CUI sets Status Register bits SR[4] and SR[5], and places the addressed partition in read status register mode.
	0xD0	Block Erase Confirm	If the first command was Block Erase Setup (0x20), the CUI latches the address and data, and the WSM erases the addressed block. During block-erase operations, the partition responds only to Read Status Register and Erase Suspend commands.
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 18: Command Codes and Definitions (Sheet 2 of 2)

Mode	Code	Device Mode	Description
Block Locking/ Unlocking	0x60	Lock Block Setup	First cycle of a 2-cycle command; prepares the CUI for block lock configuration changes. If the next command is not Block Lock (0x01), Block Unlock (0xD0), or Block Lock-Down (0x2F), the CUI sets Status Register bits SR[4] and SR[5], indicating a command sequence error.
	0x01	Lock Block	If the previous command was Block Lock Setup (0x60), the addressed block is locked.
	0xD0	Unlock Block	If the previous command was Block Lock Setup (0x60), the addressed block is unlocked. If the addressed block is in a lock-down state, the operation has no effect.
	0x2F	Lock-Down Block	If the previous command was Block Lock Setup (0x60), the addressed block is locked down.
Protection	0xC0	Program Protection Register Setup	First cycle of a 2-cycle command; prepares the device for a Protection Register or Lock Register program operation. The second cycle latches the register address and data, and starts the programming algorithm. The Read Array command must be issued to read array data after programming has finished.
Configuration	0x60	Configure Read Configuration Register Setup	First cycle of a 2-cycle command; prepares the CUI for a Read Configuration Register program operation. If the Configure 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	Configure Read Configuration Register	If the previous command was Configure 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 read mode and synchronous burst mode. Asynchronous array read mode is the default read mode after device power-up or a reset. The Read Configuration Register (RCR) must be set before synchronous burst operation can be performed (see Section 10.3, "Read Configuration Register (RCR)" on page 42).

Each partition of the device can be in any of four read states: Read Array, Read Identifier, Read Status or Read Query. 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 37). See Section 15.0, "Special Read States" on page 66 for details regarding Read Status, Read ID, and CFI Query modes.

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

The following sections describe read-mode operations in detail.

10.1 Asynchronous Read Mode

Following a device power-up or reset, asynchronous array read 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 operations, reads Status, Query, ID, etc.), the Read Array command must be issued in order to read from the flash memory array. Each asynchronous read retrieves one data word. Asynchronous reads are permitted in all blocks.

Note: Asynchronous reads can only be performed when Read Configuration Register bit RCR[15] is set (see Section 10.3, "Read Configuration Register (RCR)" on page 42).

To perform an asynchronous array or non-array read, an address is driven onto A[MAX:16] and AD[15:0], and ADV# and CE# are asserted. WE# and RST# must already have been deasserted. WAIT is deasserted during asynchronous read mode. ADV# is driven high to latch the address information. CLK is not used during asynchronous reads, and is ignored. A valid data is driven onto AD[15:0] after an initial access delay (see Section 7.0, "AC Characteristics" on page 20).

10.2 Synchronous Burst-Mode Read

Synchronous burst mode outputs 4-, 8-, 16-, or a continuous number of contiguous words after the device latches one address. Read Configuration register bits CR[15:0] must be set before synchronous burst operation can be performed. (See Section 10.3, "Read Configuration Register (RCR)" on page 42 for details). To perform a synchronous burst read, an initial address is driven onto A[MAX:16] and AD[15:0], and ADV# and CE# are asserted. WE# and RST# must already have been deasserted.

During synchronous array and non-array read modes, the first valid data is driven onto AD[15:0], with respect to a valid clock edge after the asynchronous access time (tAVQV) has been met, regardless of the latency setting. As shown in Figure 17, "Example Latency Count Setting with Flow- through feature" on page 45, with a latency setting of 4 clocks, data is driven after the third clock since the tAVQV requirement has been met. That data continues to be available on the data bus until the first access latency period is over, as described in Section 10.3.2, "Latency Count" on page 43. This

"flow through" behavior *only* applies to the first access of any synchronous read bus cycle. All subsequent data is driven on valid clock edges following the first access latency; 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, after OE# is driven low WAIT indicates invalid data on subsequent clock edges when asserted, and valid data when de-asserted with respect to a valid clock edge. See Figure 7, "Synchronous Array Read with Flow-through Feature Timing" on page 26 and Figure 9, "Burst Suspend Timing" on page 27 for additional details. Synchronous burst reads are permitted in all blocks.

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}.

To resume the burst access, OE# is reasserted and CLK is restarted. Subsequent CLK edges resume the burst sequence. See Figure 9, "Burst Suspend Timing" on page 27.

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 37).

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 67).

The RCR is shown in Table 19. The following sections describe each RCR bit

Table 19: Read Configuration Register Description (Sheet 1 of 2)

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 read (default)
14	Reserved (R)	Reserved bits should be cleared (0)

Table 19: Read Configuration Register Description (Sheet 2 of 2)

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 de-asserted with valid data 1 =WAIT de-asserted one data cycle before valid data (default)
7	Burst Sequence (BS)	0 =Reserved 1 =Linear (default)
6	Clock Edge (CE)	0 = Falling edge 1 = Rising edge (default)
5:4	Reserved (R)	Reserved bits should be cleared (0)
3	Burst Wrap (BW)	0 =Wrap; Burst accesses wrap within burst length set by BL[2:0] 1 =No Wrap; Burst accesses do not wrap within burst length (default)
2:0	Burst Length (BL[2:0])	001 =4-word burst 010 =8-word burst 011 =16-word burst 111 =Continuous-word burst (default) (Other bit settings are reserved)

10.3.1 Read Mode

The Read Mode (RM) bit selects synchronous burst-mode or asynchronous read-mode operation for the device. When the RM bit is set, asynchronous read 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 before the WAIT signal indicates valid data is present on the device data signals AD[15:0]. The input clock frequency determines this value. Figure 16 shows the data output latency from ADV#-asserted for different settings of LC[2:0]. The Latency Count does not affect when data becomes available on the data signals. Valid data is driven onto the data bus, with respect to a valid clock edge, as soon as possible after the asynchronous access time is satisfied (or another word after it is sensed). In this way, the data "flows-through" on the first access, with respect to an active clock edge. The data continues to be available on the data bus until the latency period is over. The flow-through behavior only applies to the first access of any bus cycle. All subsequent data is driven on valid clock edges following the first access latency period.

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

Figure 16: First-Access Latency Count

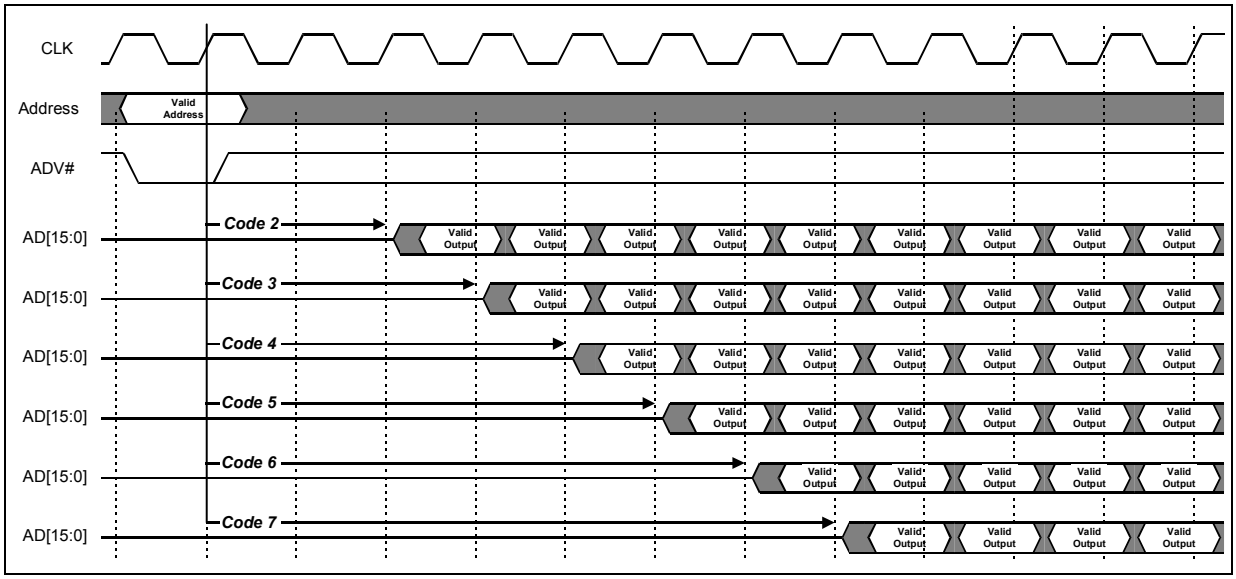


Table 20: LC and Frequency Support (t_{AVQV}/t_{CHQV} = 85 ns / 14 ns)

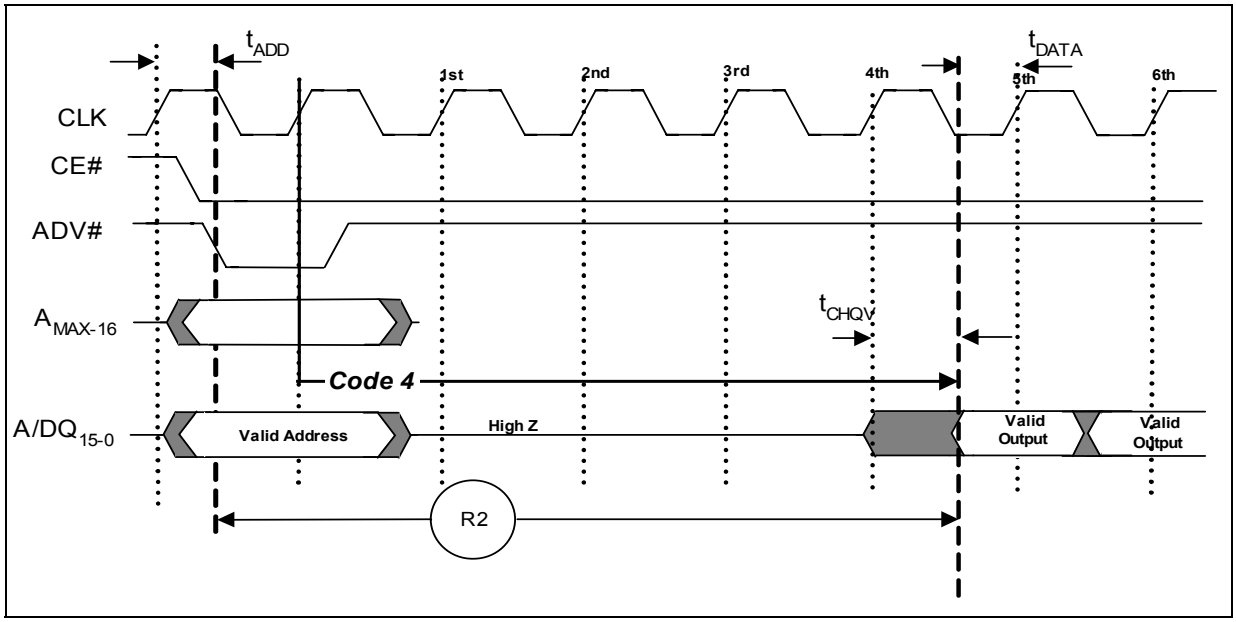
V_{CCQ} = 1.7 V to 2.0 V	
Latency Count Settings	**Frequency Support (MHz)**
2	£ 28
3	£ 40
4, 5, 6 or 7	£ 54

Table 21: LC and Frequency Support (t_{AVQV}/t_{CHQV} = 90 ns / 17 ns)

V_{CCQ} = 1.35 V to 2.0 V	
Latency Count Settings	**Frequency Support (MHz)**
2	£ 27
3, 4, 5, 6 or 7	£ 40

See Figure 17, "Example Latency Count Setting with Flow- through feature" on page 45.

Figure 17: Example Latency Count Setting with Flow- through feature



Note: The waveform above illustrates the Latency Count of 4 with Flow-through feature. The Flow-through feature will be shown only when the initial access time is one clock less than the LC setting.

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 and 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.

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 single word read mode, WAIT is set to an "de-asserted" state as determined by CR[10]. See Table 22, "WAIT Summary Table" on page 45, and Figure 6, "Asynchronous Single-Word Read Timing" on page 25.

Table 22: WAIT Summary Table

CONDITION	WAIT
CE# = V_{IH} CE# = V_{IL}	High-Z Driven
OE# = V_{IH} OE# = V_{IL}	De-asserted Active
Synchronous Array and Non-array Reads	Active
All Asynchronous Read and all Write	De-asserted

Note: Active: WAIT is asserted until data becomes valid, then de-asserts. WAIT is asserted during the initial access (latency) and at the end of the burst cycle with OE# low.

10.3.4 Data Hold

For burst read operations, the Data Hold (DH) bit determines whether the data output remains valid on AD[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 18). 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:

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

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

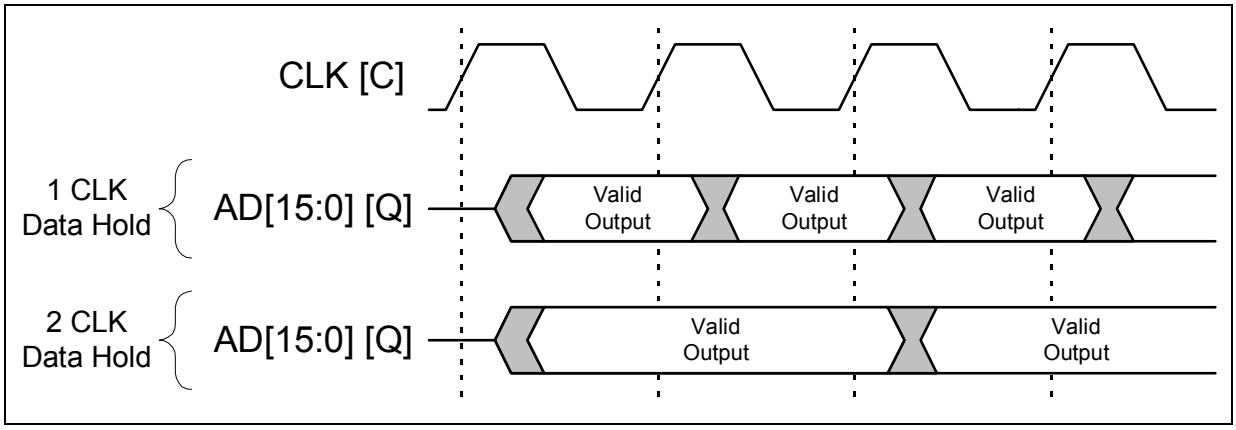
For example, with a clock frequency of 54 MHz, the clock period is 18.5 ns. Assuming t_{CHQV} = 14ns and t_{DATA} = 4ns. Applying these values to the formula above:

14 ns + 4 ns ≤ 18.5 ns

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

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

Figure 18: 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 AD[15:0]. When WD is set, WAIT is de-asserted one data cycle *before* valid data (default). When WD is cleared, WAIT is de-asserted *during* valid data.

10.3.6 Burst Sequence

The Burst Sequence (BS) bit selects linear-burst sequence (default). Only linear-burst sequence is supported. Table 23 shows the synchronous burst sequence for all burst lengths, as well as the effect of the Burst Wrap (BW) setting.

Table 23: 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 over to the next burst-length segment. 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 23, "Burst Sequence Word Ordering" on page 47). 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, buffered programming, and Buffered Enhanced Factory Programming (Buffered EFP). See Section 9.0, "Device Operations" on page 36 for details on the various programming commands issued to the device.

Successful programming requires the addressed block to be unlocked. If the block is locked down, WP# must be deasserted and the block unlocked before attempting to program the block. Attempting to program a locked block causes a program error (SR[4] and SR[1] set) and termination of the operation. See Section 13.0, "Security Modes" on page 57 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 36). 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 31, "Word Program Flowchart" on page 76.

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 55).

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). 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.

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 17 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 19, "Example VPP Supply Connections" on page 54 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 37), Status Register information is updated and reflects the availability of the write buffer. SR[7] indicates buffer availability: if set, the buffer is available; if cleared, the write 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 33, "Buffer Program Flowchart" on page 78).

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 is 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 Section 14.0, "Dual-Operation Considerations" on page 62).

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

Buffered EFP consists of three phases: Setup, Program/Verify, and Exit (see Figure 34, "Buffered EFP Flowchart" on page 79). 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 command sequence is used to program a 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_C = 25°C, ±5°C
- V_{CC} within specified operating range.
- VPP driven to V_{PPH}.
- Target block unlocked before issuing the Buffered EFP Setup and Confirm commands.
- The first-word address (WA0) for the block to be programmed must be held constant from the setup phase through all data streaming into the target block, until transition to the exit phase is desired.
- WA0 must align with the start of an array buffer boundary[1].

Buffered EFP considerations:

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

- Read operation while performing Buffered EFP is not supported.

Notes:

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

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.

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 at this time to ensure the entire block programmed successfully. 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 32, "Program Suspend/Resume Flowchart" on page 77).

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 7.7, "Program and Erase Characteristics" on page 32.

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

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

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

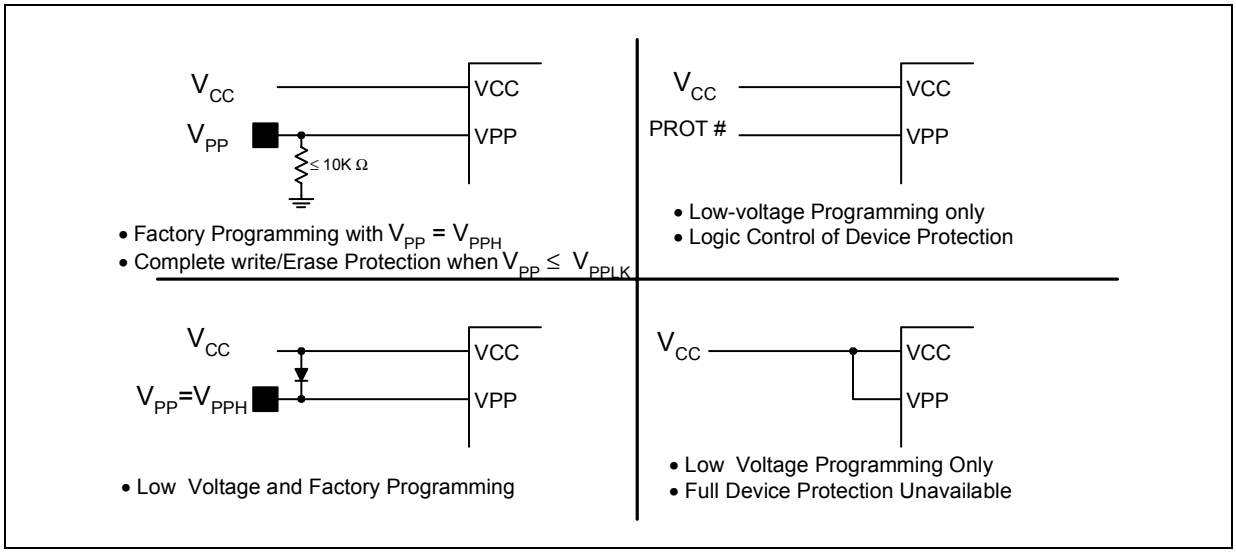
11.5 Program Resume

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 32, "Program Suspend/Resume Flowchart" on page 77).

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 19: 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 37). 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 35, "Block Erase Flowchart" on page 80).

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 49).

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

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

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

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 write-buffer program operation, or a read operation within any block except the block that is erase suspended (see Figure 32, "Program Suspend/Resume Flowchart" on page 77).

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 7.7, "Program and Erase Characteristics" on page 32.

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 32, "Program Suspend/Resume Flowchart" on page 77).

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_{PP} data security can be used to inhibit program and erase operations (see Section 11.6, "Program Protection" on page 53 and Section 12.4, "Erase Protection" on page 56).

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 37 and Figure 37, "Block Lock Operations Flowchart" on page 82). If the Set Read Configuration Register command is issued after the Block Lock Setup command, the device configures the RCR instead.

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

Table 24: Block Locking

V_{PP}	WP#	RST#	Block Write Protection	Block Lock Bits
X	X	V_{IL}	All blocks write/erase protected	Block lock bits may not be changed
$\leq V_{PPLK}$	V_{IL}	V_{IH}	**All** blocks write/erase protected	Lock-Down block states may **not** be changed
$\leq V_{PPLK}$	V_{IH}	V_{IH}	**All** blocks write/erase protected	All Lock-Down block states **may be** changed
$> V_{PPLK}$	V_{IL}	V_{IH}	All Lock-Down and Locked blocks write/ erase protected	Lock-Down block states may **not** be changed
$> V_{PPLK}$	V_{IH}	V_{IH}	All Lock-Down and Locked blocks write/ erase protected	All Lock-Down block states **may be** changed

13.1.2 Unlock Block

The Unlock Block command is used to unlock blocks (see Section 9.2, "Device Commands" on page 37). 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 20, "Block Locking State Diagram" on page 58).

13.1.3 Lock-Down Block

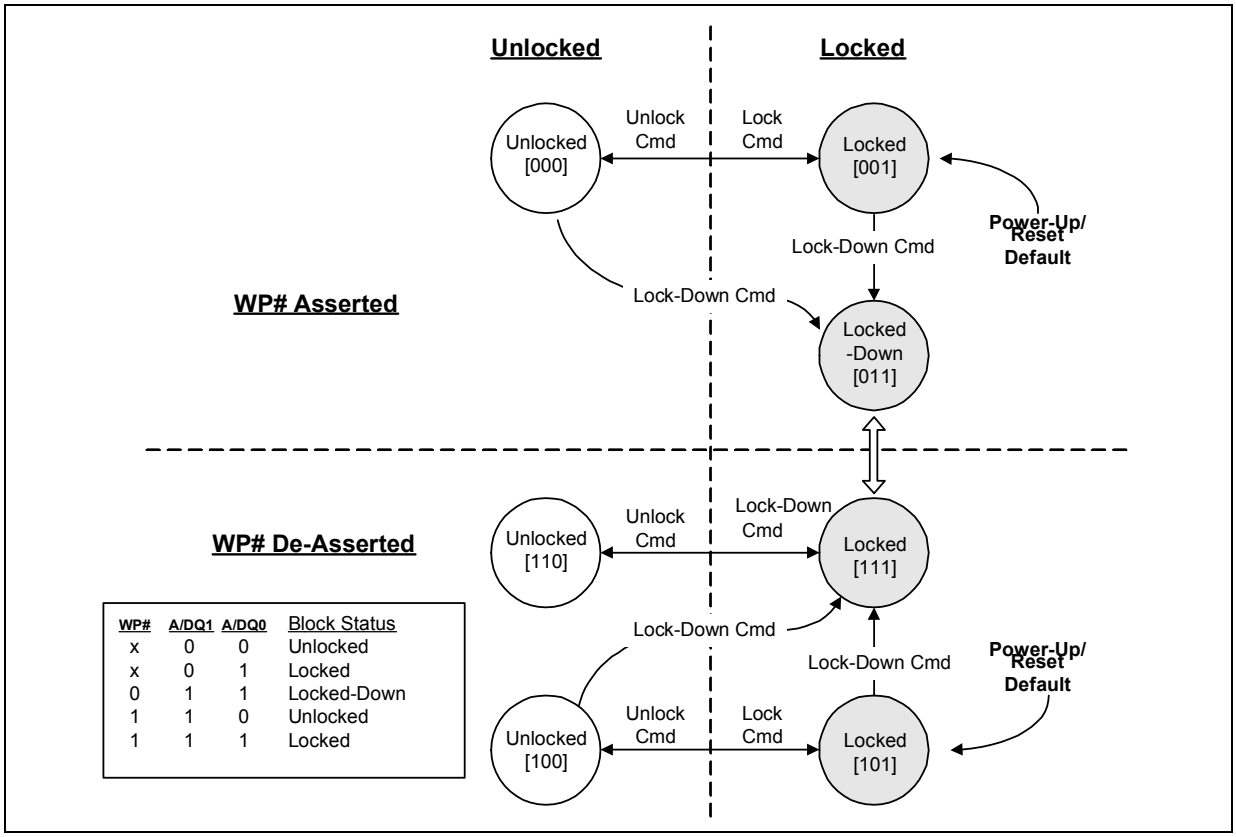
A locked or unlocked block can be locked-down by writing the Lock-Down Block command sequence (see Section 9.2, "Device Commands" on page 37). Blocks in a lock-down state cannot be programmed or erased; they can only be read. However, unlike locked blocks, their locked state cannot be changed by software commands

alone. A locked-down block can only be unlocked by issuing the Unlock Block command with WP# deasserted. To return an unlocked block to locked-down state, a Lock command (60h/01h) must be issued prior to changing WP# to V_{IL}. A locked or unlocked block can be locked-down by writing the Lock-Down Block command sequence. Locked-down blocks revert to the locked state upon reset or power up the device (see Figure 20, "Block Locking State Diagram" on page 58).

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 67). Data bits AD[1:0] display the addressed block's lock status; AD0 is the addressed block's lock bit, while AD1 is the addressed block's lock-down bit.

Figure 20: 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" on page 69, which shows valid commands during an erase suspend.

13.2 Protection Registers

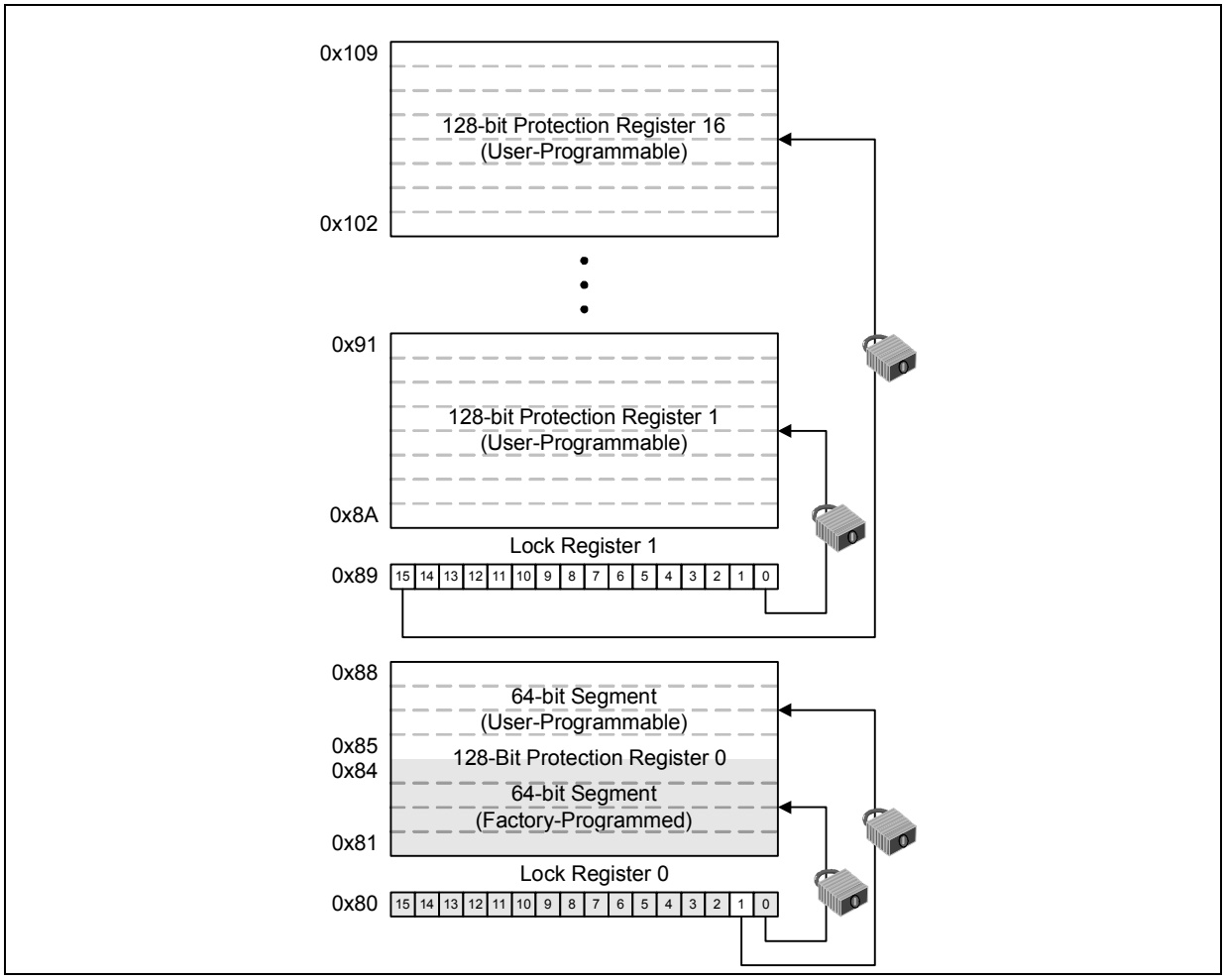
The device contains 17 Protection Registers (PRs) that can be used to implement system security measures and/or device identification. Each Protection Register can be individually locked.

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 21, "Protection Register Map" on page 60).

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 21: 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 37). Next, perform a read operation at that partition's base address plus the address offset corresponding to the register to be read. Table 27, "Device Identifier Information" on page 67 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 25, "Simultaneous Operation Restrictions" on page 65 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 37). Next, write the desired Protection Register data to the same Protection Register address (see Figure 21, "Protection Register Map" on page 60).

The device programs the 64-bit and 128-bit user-programmable Protection Register data 16 bits at a time (see Figure 38, "Protection Register Programming Flowchart" on page 83). 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 25, "Simultaneous Operation Restrictions" on page 65 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 37). 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 27, "Device Identifier Information" on page 67).

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 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 22, and Figure 23). However, a write operation issued between a 2-cycle commands write sequence causes a command sequence error. (See Figure 24)

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 22: Operating Mode with Correct Command Sequence Example

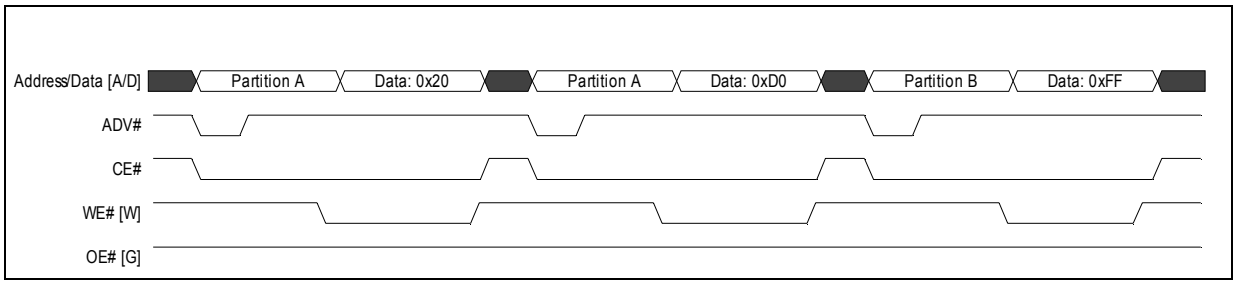


Figure 23: Operating Mode with Correct Command Sequence Example

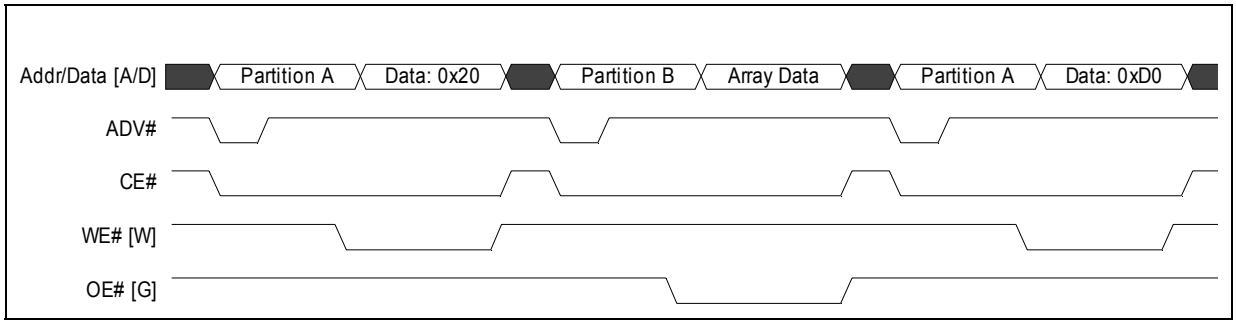
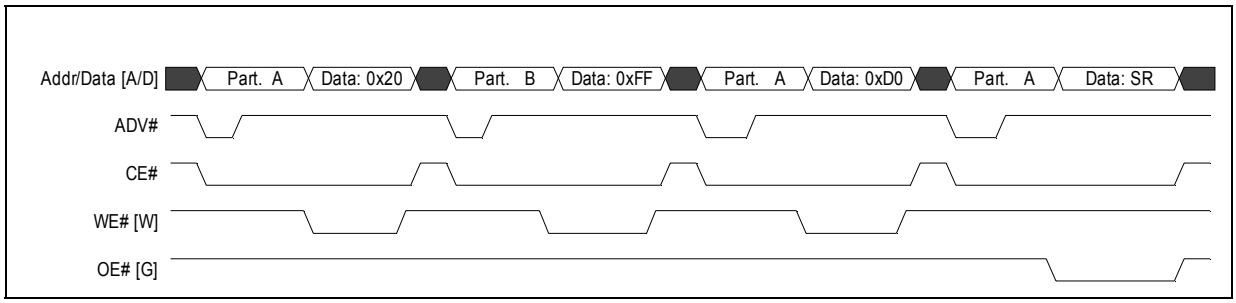


Figure 24: Operating Mode with Illegal Command Sequence Example



14.2.1 Simultaneous Operation Details

The L18 device supports simultaneous read from one partition while programming or erasing in any other partition. Certain features like the Protection Registers and CFI Query data have special requirements with respect to simultaneous operation capability. (Table 25 provides details on restrictions during simultaneous operations.)

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 11, "Write to Asynchronous Read Timing" on page 28, Figure 12, "Synchronous Read to Write Timing" on page 29, and Figure 13, "Write to Synchronous Read" on page 30, 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 33, "Buffer Program Flowchart" on page 78 for more details.

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

14.2.4 Simultaneous Operation Restrictions

The Protection Registers share some of the same internal flash resources as the parameter partition. Therefore, simultaneous read-while-write is only allowed between the protection register and main partitions. Table 25 describes the operation allowed using read-while-write/erase with the protection register.

Table 25: Simultaneous Operation Restrictions

Protection Register or CFI data	Parameter Partition Array Data	Other Partitions	Notes
Read	(See Notes)	Write/Erase	While programming or erasing in a main partition, the Protection Register or CFI data may be read from any other partition. Reading the parameter partition array data is not allowed if the Protection Register or Query data is being read from addresses within the parameter partition.
(See Notes)	Read	Write/Erase	While programming or erasing in a main partition, read operations are allowed in the parameter partition. Accessing the Protection Registers or CFI data from parameter partition addresses is not allowed when reading array data from the parameter partition.
Read	Read	Write/Erase	While programming or erasing in a main partition, read operations are allowed in the parameter partition. Accessing the Protection Registers or CFI data in a partition that is *different* from the one being programed/erased, and also *different* from the parameter partition is allowed.
Write	No Access Allowed	Read	While programming the Protection Register, reads are only allowed in the other main partitions. Access to array data in the parameter partition is not allowed. Programming of the Protection Register can only occur in the parameter partition, which means this partition is in Read Status.
No Access Allowed	Write/Erase	Read	While programming or erasing the parameter partition, reads of the Protection Registers or CFI data are not allowed in *any* partition. Reads in partitions other than the 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 is exactly the same as an array read. However, when using synchronous burst mode for non-array reads, the same word of data requested will be output on successive clock edges until the burst length requirements are satisfied. The "flow-through" feature also applies to synchronous non-array reads. Refer to Section 10.3.1, "Read Mode" on page 43 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).

Status Register data is output on AD[7:0], while 0x00 is output on AD[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. Status Register read operations do not affect the read state of the other partitions.

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

Table 26: Status Register Description (Sheet 1 of 2)

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

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

Table 26: Status Register Description (Sheet 2 of 2)

Status Register (SR)		**Default Value = 0x80**
2	Program Suspend Status (PSS)	0 = Program suspend not in effect. 1 = Program suspend in effect.
1	Block-Locked Status (BLS)	0 = Block not locked during program or erase. 1 = Block locked during program or erase; operation aborted.
0	Partition Write Status (PWS)	DWS PWS 0 0 = Program or erase operation in addressed partition. 0 1 = Program or erase operation in other partition. 1 0 = No active program or erase operations. 1 1 = Reserved. (Non-buffered EFP operation. For Buffered EFP operation, see Section 11.3, "Buffered Enhanced Factory Programming" on page 51).

Always clear the Status Register prior to resuming erase operations. Avoid 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_{PP}. The Write State Machine (WSM) sets and clears SR[7,6,2,0], but it sets bits SR[5:3,1] without clearing them. The Status Register should be cleared before starting a command sequence to avoid any ambiguity. A device reset also clears the Status Register.

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 37 for details on issuing the Read Device Identifier command). Table 27, "Device Identifier Information" on page 67 and Table 28, "Device ID codes" on page 68 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 27: Device Identifier Information (Sheet 1 of 2)

Item	**Address(1,2)**	**Data**
Manufacturer Code	PBA + 0x00	0089h
Device ID Code	PBA + 0x01	ID (see Table 28)
Block Lock Configuration: • Block Is Unlocked • Block Is Locked • Block Is not Locked-Down • Block Is Locked-Down	BBA + 0x02	Lock Bit: AD0 = 0b0 AD0 = 0b1 AD1 = 0b0 AD1 = 0b1
Configuration Register	PBA + 0x05	Configuration Register Data

Table 27: Device Identifier Information (Sheet 2 of 2)

Item	Address(1,2)	Data
Lock Register 0	PBA + 0x80	Protection Register Lock Bits
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 Lock Bits
128-bit User-Programmable Protection Registers	PBA + 0x8A–0x109	User Protection Register Data

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

Table 28: Device ID codes

ID Code Type	Device Density	Device Identifier Codes	
		–T (Top Parameter)	–B (Bottom Parameter)
Device Code	64 Mbit	8808	8834
	128 Mbit	8809	8835
	256 Mbit	880A	8836

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 37 for details on issuing the CFI Query command. Appendix B, "Common Flash Interface (CFI)" on page 83 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 25.

Appendix A Write State Machine

Figure 25 through Figure 30 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 25: Write State Machine—Next State Table (Sheet 1 of 6)

		Command Input to Chip and resulting *Chip* Next State										
Current Chip State (7)		Read Array (2) (FFH)	Word Program (3,4) (10H/40H)	Buffered Program (BP) (E8H)	Erase Setup (3,4) (20H)	Buffered Enhanced Factory Pgm Setup (3, 4) (80H)	BE Confirm, P/E Resume, ULB, Confirm (8) (D0H)	BP / Prg / Erase Suspend (B0H)	Read Status (70H)	Clear Status Register (5) (50H)	Read ID/Query (90H, 98H)	Lock, Unlock, Lock-down, CR setup (4) (60H)
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											
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 26: Write State Machine—Next State Table (Sheet 2 of 6)

Current Chip State (7)		Command Input to Chip and resulting *Chip* Next State										
		Read Array (2) (FFH)	Word Program (3,4) (10H/40H)	Buffered Program (BP) (E8H)	Erase Setup (3,4) (20H)	Buffered Enhanced Factory Pgm Setup (3, 4) (80H)	BE Confirm, P/E Resume, ULB, Confirm (8) (D0H)	BP / Prg / Erase Suspend (B0H)	Read Status (70H)	Clear Status Register (5) (50H)	Read ID/Query (90H, 98H)	Lock, Unlock, Lock-down, CR setup (4) (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. (7)										

Figure 27: Write State Machine—Next State Table (Sheet 3 of 6)

Current Chip State (7)		Command Input to Chip and resulting *Chip* Next State						
		OTP Setup (4) (C0H)	Lock Block Confirm (8) (01H)	Lock-Down Block Confirm (8) (2FH)	Write RCR Confirm (8) (03H)	Block Address (?WA0) 9 (XXXXH)	Illegal Cmds or BEFP Data (1) (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)		
OTP	Setup	OTP Busy						Ready
	Busy	Word Program Busy						N/A
Word Program	Setup	Word Program Busy						Ready
	Busy	Word Program Suspend						N/A
	Suspend	BP Load 1						
BP	Setup	BP Load 2				Ready (BP Load 2	BP Load 2	
	BP Load 1	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 Load 2	Ready (Error)				Ready (Error) (Proceed if unlocked or lock error)	Ready (Error)	
	BP Confirm	BP Busy						Ready
	BP Busy	BP Suspend						N/A
	BP Suspend	Ready (Error)						
Erase	Setup	Erase Busy						Ready
	Busy	Erase Suspend						N/A
	Suspend							

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

Current Chip State (7)		Command Input to Chip and resulting *Chip* Next State						
		OTP Setup (4) (C0H)	Lock Block Confirm (8) (01H)	Lock-Down Block Confirm (8) (2FH)	Write RCR Confirm (8) (03H)	Block Address (?WA0) 9 (XXXXH)	Illegal Cmds or BEFP Data (1) (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						N/A
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)		
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 29: Write State Machine—Next State Table (Sheet 5 of 6)

Output Next State Table

	Command Input to Chip and resulting *Output* Mux Next State										
Current chip state	Read Array [2] (FFH)	Word Program Setup (3,4) (10H/40H)	BP Setup (E8H)	Erase Setup [3,4] (20H)	Buffered Enhanced Factory Pgm Setup [3, 4] (30H)	BE Confirm, P/E Resume, ULB Confirm [8] (D0H)	Program/ Erase Suspend (B0H)	Read Status (70H)	Clear Status Register [5] (50H)	Read ID/Query (90H, 98H)	Lock, Unlock, Lock-down, CR setup [4] (60H)
BEFP Setup, BEFP Pgm & Verify Busy, Erase Setup, OTP Setup, BP: Setup, Load 1, Load 2, Confirm, Word Pgm Setup, Word Pgm Setup in Erase Susp, BP Setup, Load1, Load 2, Confirm in Erase Suspend	Status Read										
Lock/CR Setup, Lock/CR Setup in Erase Susp	Status Read										
OTP Busy	Read Array	Status Read				Output does not change.		Status Read	Output mux does not change.	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										ID Read	

Figure 30: Write State Machine—Next State Table (Sheet 6 of 6)

Output Next State Table

	Command Input to Chip and resulting *Output* Mux Next State						
Current chip state	OTP Setup [4] (C0H)	Lock Block Confirm [8] (01H)	Lock-Down Block Confirm [8] (2FH)	Write CR Confirm [8] (03H)	Block Address (?WA0) (FFFFH)	Illegal Cmds or BEFP Data [1] (all other codes)	WSM Operation Completes
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							

Notes:

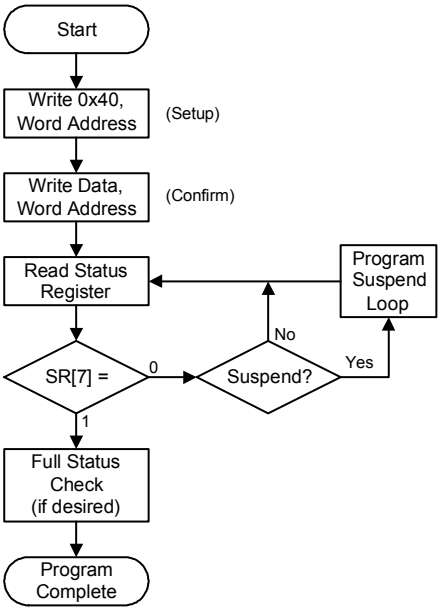
1. "Illegal commands" include commands outside of the allowed command set (allowed commands: 40H [pgm], 20H [erase], etc.)
2. If a "Read Array" is attempted from a busy partition, the result will be invalid data. The ID and Query data are located at different locations in the address map.
3. 1st and 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

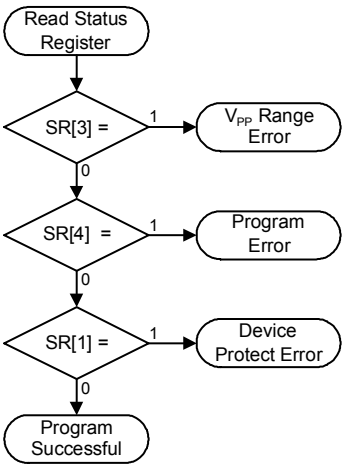
Figure 31: Word Program Flowchart

WORD PROGRAM PROCEDURE



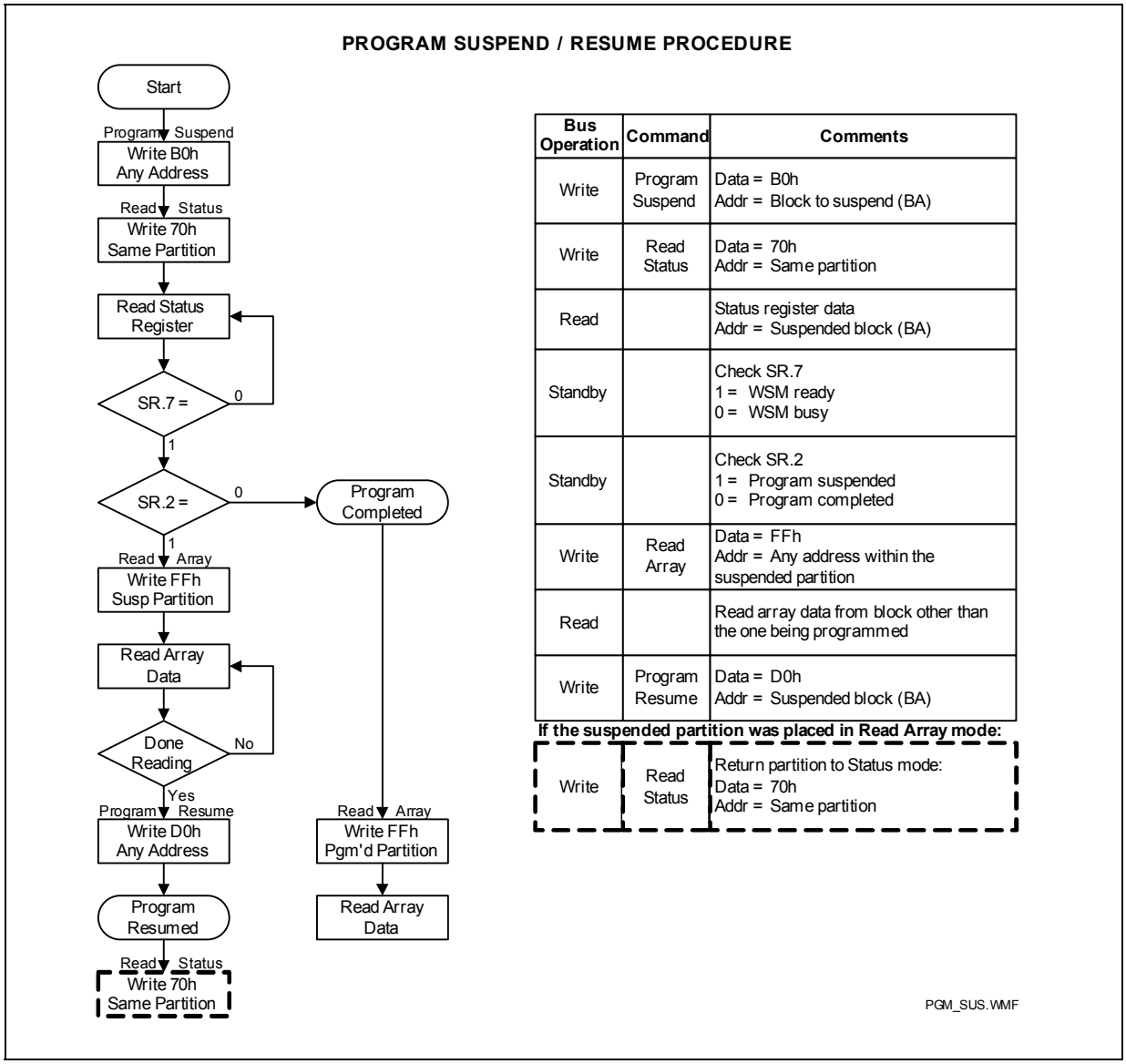
Bus Operation	Command	Comments
Write	Program Setup	Data = 0x40 Addr = Location to program
Write	Data	Data = Data to program Addr = Location to program
Read	None	Status register data
Idle	None	Check SR[7] 1 = WSM Ready 0 = WSM Busy
Repeat for subsequent Word Program operations. Full Status Register check can be done after each program, or after a sequence of program operations. Write 0xFF after the last operation to set to the Read Array state.		

FULL STATUS CHECK PROCEDURE



Bus Operation	Command	Comments
Idle	None	Check SR[3]: 1 = V_{PP} Error
Idle	None	Check SR[4]: 1 = Data Program Error
Idle	None	Check SR[1]: 1 = Block locked; operation aborted
If an error is detected, clear the Status Register before continuing operations - only the Clear Staus Register command clears the Status Register error bits.		

Figure 32: Program Suspend/Resume Flowchart

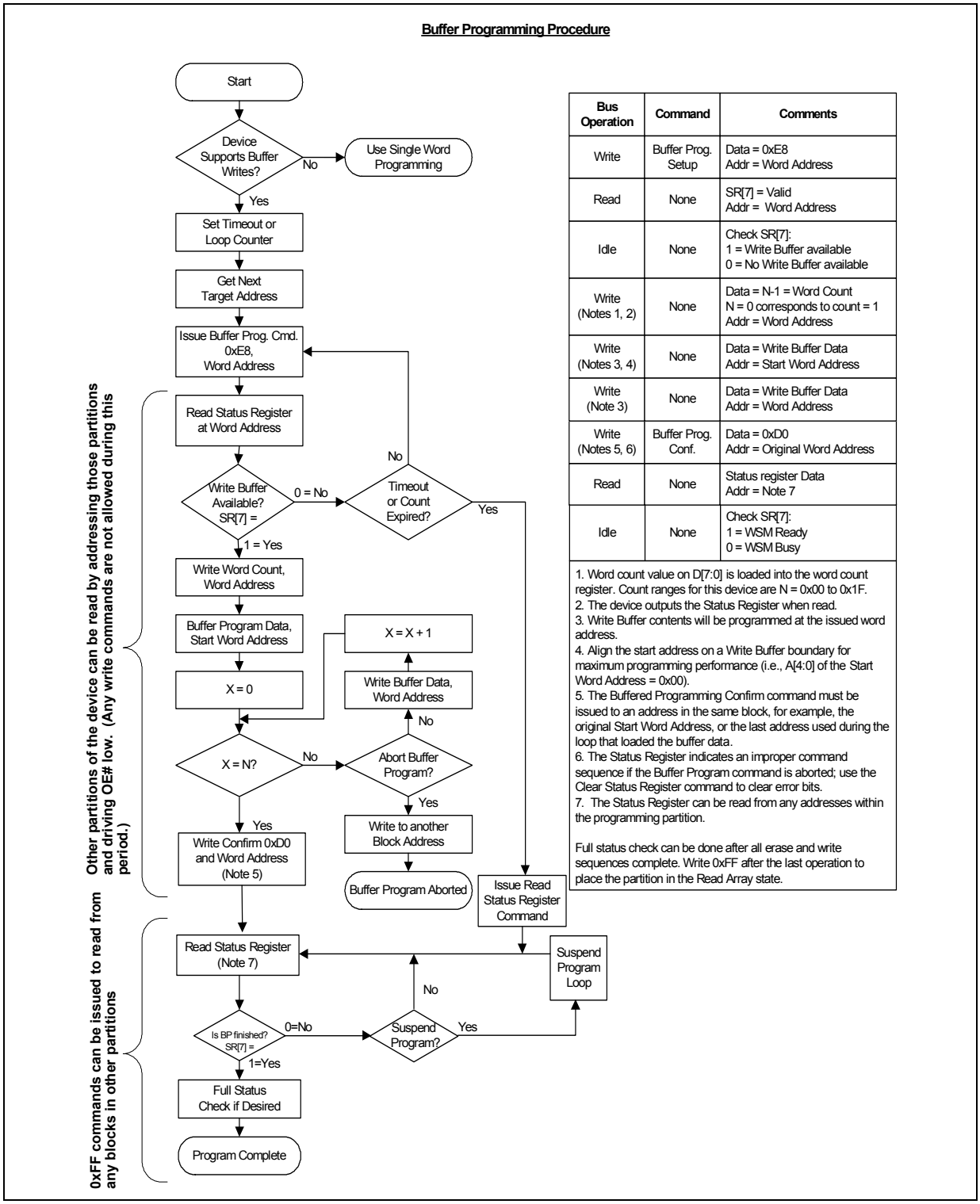


Bus Operation	Command	Comments
Write	Program Suspend	Data = B0h Addr = Block to suspend (BA)
Write	Read Status	Data = 70h Addr = Same partition
Read		Status register data Addr = Suspended block (BA)
Standby		Check SR.7 1 = WSM ready 0 = WSM busy
Standby		Check SR.2 1 = Program suspended 0 = Program completed
Write	Read Array	Data = FFh Addr = Any address within the suspended partition
Read		Read array data from block other than the one being programmed
Write	Program Resume	Data = D0h Addr = Suspended block (BA)

If the suspended partition was placed in Read Array mode:

Write	Read Status	Return partition to Status mode: Data = 70h Addr = Same partition

Figure 33: Buffer Program Flowchart

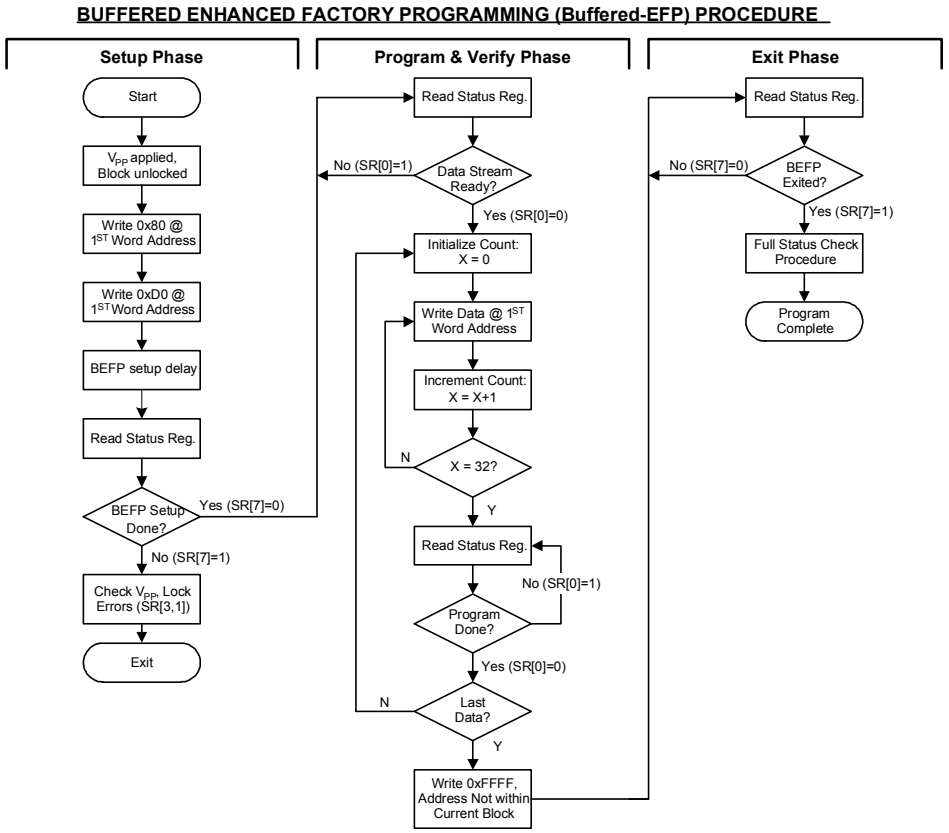


Bus Operation	Command	Comments
Write	Buffer Prog. Setup	Data = 0xE8 Addr = Word Address
Read	None	SR[7] = Valid Addr = Word Address
Idle	None	Check SR[7]: 1 = Write Buffer available 0 = No Write Buffer available
Write (Notes 1, 2)	None	Data = N-1 = Word Count N = 0 corresponds to count = 1 Addr = Word Address
Write (Notes 3, 4)	None	Data = Write Buffer Data Addr = Start Word Address
Write (Note 3)	None	Data = Write Buffer Data Addr = Word Address
Write (Notes 5, 6)	Buffer Prog. Conf.	Data = 0xD0 Addr = Original Word Address
Read	None	Status register Data Addr = Note 7
Idle	None	Check SR[7]: 1 = WSM Ready 0 = WSM Busy

1. Word count value on D[7:0] is loaded into the word count register. Count ranges for this device are N = 0x00 to 0x1F.
2. The device outputs the Status Register when read.
3. Write Buffer contents will be programmed at the issued word address.
4. Align the start address on a Write Buffer boundary for maximum programming performance (i.e., A[4:0] of the Start Word Address = 0x00).
5. The Buffered Programming Confirm command must be issued to an address in the same block, for example, the original Start Word Address, or the last address used during the loop that loaded the buffer data.
6. The Status Register indicates an improper command sequence if the Buffer Program command is aborted; use the Clear Status Register command to clear error bits.
7. The Status Register can be read from any addresses within the programming partition.

Full status check can be done after all erase and write sequences complete. Write 0xFF after the last operation to place the partition in the Read Array state.

Figure 34: Buffered EFP Flowchart



BEFP Setup

Bus State	Operation	Comments
Write	Unlock Block	V_{PPH} applied to VPP
Write (Note 1)	BEFP Setup	Data = 0x80 @ 1ST Word Address
Write	BEFP Confirm	Data = 0xD0 @ 1ST Word Address
Read	Status Register	Data = Status Reg. Data Address = 1ST Word Addr
Standby	BEFP Setup Done?	Check SR[7]: 0 = BEFP Ready 1 = BEFP Not Ready
Standby	Error Condition Check	If SR[7] is set, check: SR[3] set = V_{PP} Error SR[1] set = Locked Block

BEFP Program & Verify

Bus State	Operation	Comments
Read	Status Register	Data = Status Register Data Address = 1ST Word Addr.
Standby	Data Stream Ready?	Check SR[0]: 0 = Ready for Data 1 = Not Ready for Data
Standby	Initialize Count	X = 0
Write (Note 2)	Load Buffer	Data = Data to Program Address = 1ST Word Addr.
Standby	Increment Count	X = X+1
Standby	Buffer Full?	X = 32? Yes = Read SR[0] No = Load Next Data Word
Read	Status Register	Data = Status Reg. data Address = 1ST Word Addr.
Standby	Program Done?	Check SR[0]: 0 = Program Done 1 = Program in Progress
Standby	Last Data?	No = Fill buffer again Yes = Exit
Write	Exit Prog & Verify Phase	Data = 0xFFFF @ address not in current block

BEFP Exit

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

Repeat for subsequent blocks;

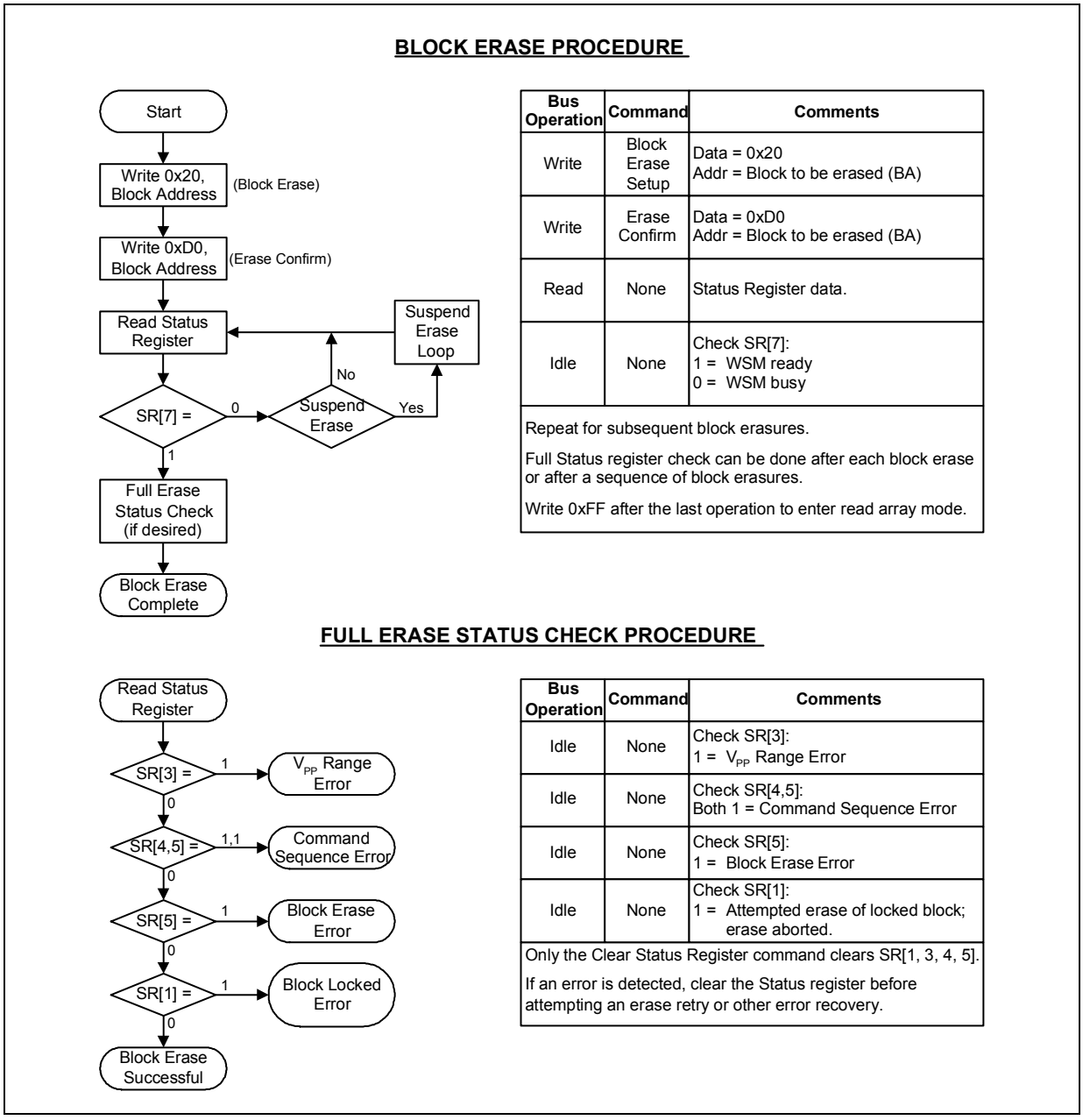
After BEFP exit, a full Status Register check can determine if any program error occurred;

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

Write 0xFF to enter Read Array state.

NOTES:
1. First-word address to be programmed within the target block must be aligned on a write-buffer boundary.
2. Write-buffer contents are programmed sequentially to the flash array starting at the first word address. WSM internally increments addressing.

Figure 35: Block Erase Flowchart



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

Repeat for subsequent block erasures.

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

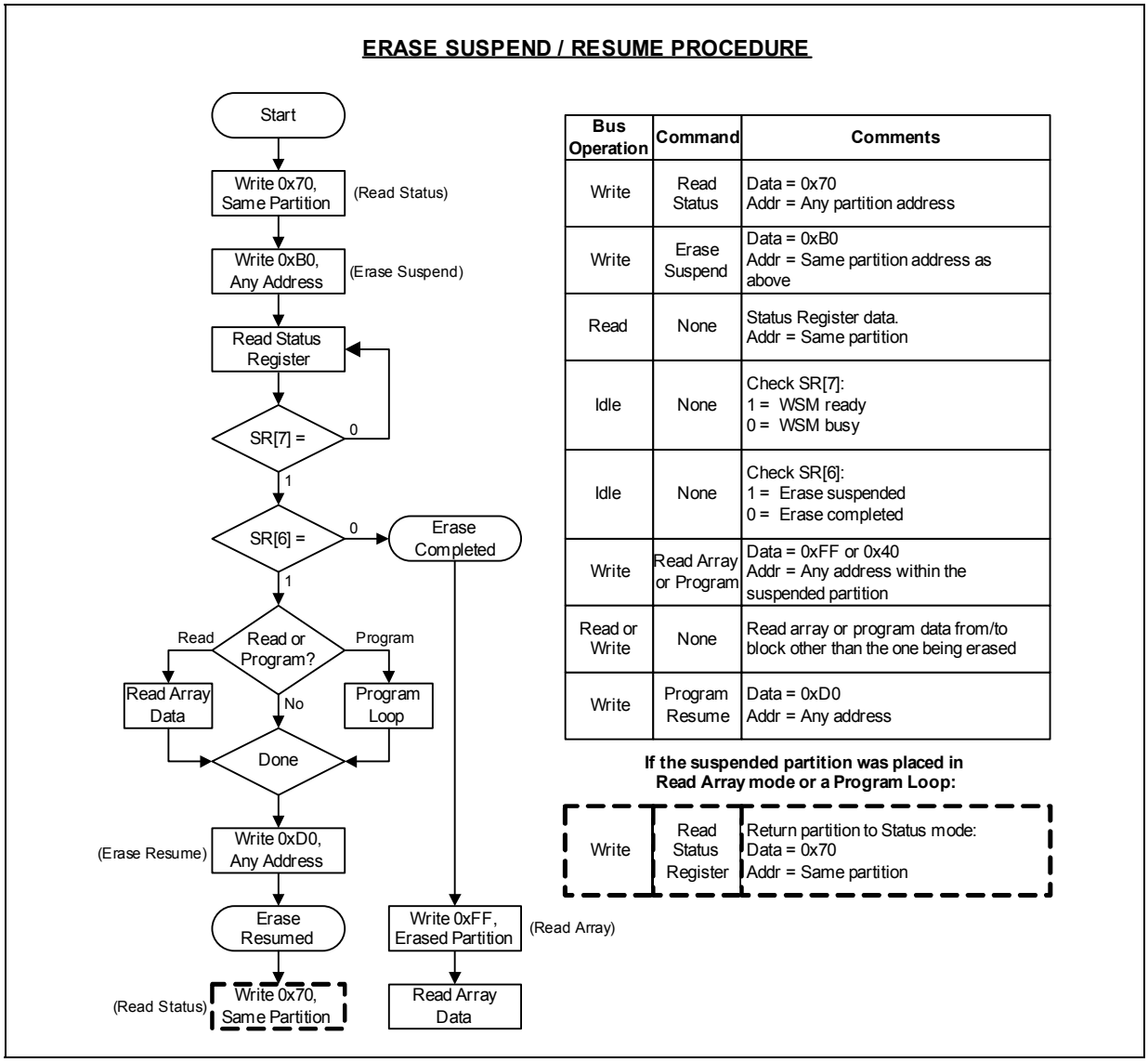
Write 0xFF after the last operation to enter read array mode.

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

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

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

Figure 36: Erase Suspend/Resume Flowchart

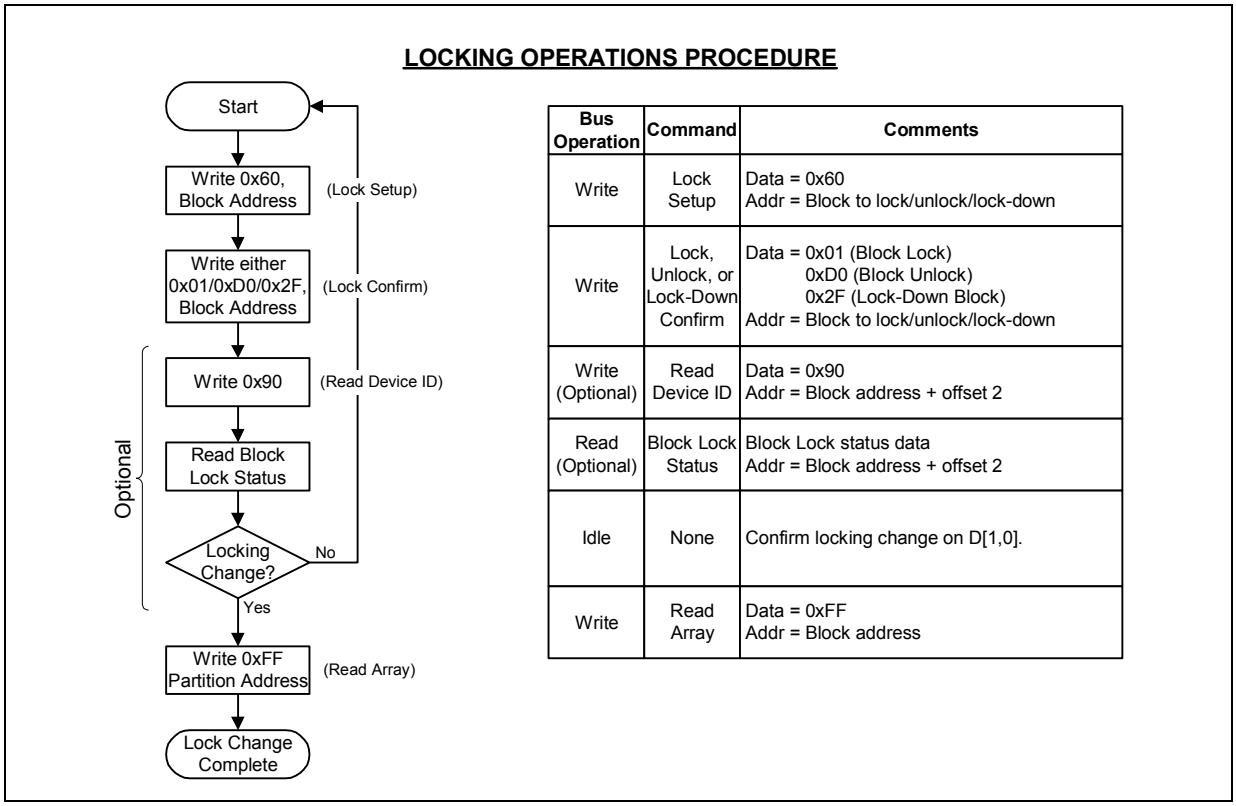


Bus Operation	Command	Comments
Write	Read Status	Data = 0x70 Addr = Any partition address
Write	Erase Suspend	Data = 0xB0 Addr = Same partition address as above
Read	None	Status Register data. Addr = Same partition
Idle	None	Check SR[7]: 1 = WSM ready 0 = WSM busy
Idle	None	Check SR[6]: 1 = Erase suspended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

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

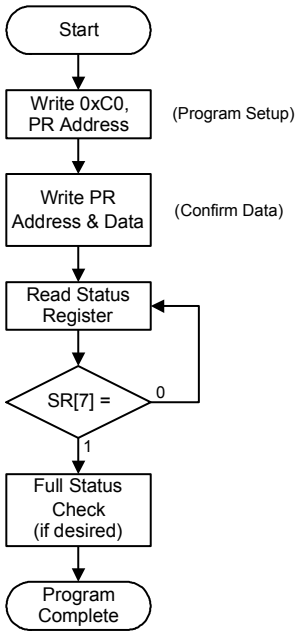
Write	Read Status Register	Return partition to Status mode: Data = 0x70 Addr = Same partition

Figure 37: Block Lock Operations Flowchart

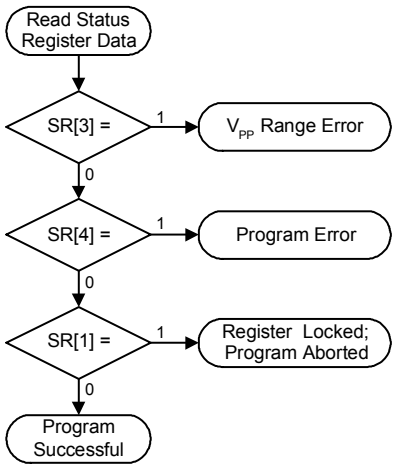


Bus Operation	Command	Comments
Write	Lock Setup	Data = 0x60 Addr = Block to lock/unlock/lock-down
Write	Lock, Unlock, or Lock-Down Confirm	Data = 0x01 (Block Lock) 0xD0 (Block Unlock) 0x2F (Lock-Down Block) Addr = Block to lock/unlock/lock-down
Write (Optional)	Read Device ID	Data = 0x90 Addr = Block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

Figure 38: Protection Register Programming Flowchart



Bus Operation	Command	Comments
Write	Program PR Setup	Data = 0xC0 Addr = First Location to Program
Write	Protection Program	Data = Data to Program Addr = Location to Program
Read	None	Status Register Data.
Idle	None	Check SR[7]: 1 = WSM Ready 0 = WSM Busy
Program Protection Register operation addresses must be within the Protection Register address space. Addresses outside this space will return an error. Repeat for subsequent programming operations. Full Status Register check can be done after each program, or after a sequence of program operations. Write 0xFF after the last operation to set Read Array state.		



Bus Operation	Command	Comments
Idle	None	Check SR[3]: 1 =V_{PP} Range Error
Idle	None	Check SR[4]: 1 =Programming Error
Idle	None	Check SR[1]: 1 =Block locked; operation aborted
Only the Clear Staus Register command clears SR[1, 3, 4]. If an error is detected, clear the Status register before attempting a program retry or other error recovery.		

B.1 Common Flash Interface (CFI)

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 37). 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.

B.2 Query Structure Output

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

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

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

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

Table 29: 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 30: Example of Query Structure Output of x16- Devices

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

B.3 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 31: Query Structure

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

Notes:
1. Refer to the Query Structure Output section and offset 28h for the detailed definition of offset address as a function of device bus width and mode.
2. BA = Block Address beginning location (i.e., 08000h is block 1's beginning location when the block size is 16K-word).
3. Offset 15 defines "P" which points to the Primary Intel-specific Extended Query Table.

B.4 CFI Query Identification String

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

Table 32: 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 33: System Interface Information

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

B.5 Device Geometry Definition

Table 34: Device Geometry Definition

Offset	Length	Description	Code		
27h	1	"n" such that device size = 2^n in number of bytes	27:	See table below	
28h	2	Flash device interface code assignment: "n" such that n+1 specifies the bit field that represents the flash device width capabilities as described in the table: 7 6 5 4 3 2 1 0 — — — — x64 x32 x16 x8 15 14 13 12 11 10 9 8 — — — — — — — —	28: 29:	--01 --00	x16
2Ah	2	"n" such that maximum number of bytes in write buffer = 2^n	2A: 2B:	--06 --00	64
2Ch	1	Number of erase block regions (x) within device: 1. x = 0 means no erase blocking; the device erases in bulk 2. x specifies the number of device regions with one or more contiguous same-size erase blocks. 3. Symmetrically blocked partitions have one blocking region	2C:	See table below	
2Dh	4	Erase Block Region 1 Information bits 0–15 = y, y+1 = number of identical-size erase blocks bits 16–31 = z, region erase block(s) size are z x 256 bytes	2D: 2E: 2F: 30:	See table below	
31h	4	Erase Block Region 2 Information bits 0–15 = y, y+1 = number of identical-size erase blocks bits 16–31 = z, region erase block(s) size are z x 256 bytes	31: 32: 33: 34:	See table below	
35h	4	Reserved for future erase block region information	35: 36: 37: 38:	See table below	

Address	64 Mbit		128 Mbit		256 Mbit	
	–B	–T	–B	–T	–B	–T
27:	--17	--17	--18	--18	--19	--19
28:	--01	--01	--01	--01	--01	--01
29:	--00	--00	--00	--00	--00	--00
2A:	--06	--06	--06	--06	--06	--06
2B:	--00	--00	--00	--00	--00	--00
2C:	--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

B.6 Intel-Specific Extended Query Table

Table 35: Primary Vendor-Specific Extended Query

Offset[1] P= 10Ah	Length	Description (Optional flash features and commands)	Add.	Hex Code	Value
(P+0)h	3	Primary extended query table	10A	--50	"P"
(P+1)h		Unique ASCII string "PRI"	10B:	--52	"R"
(P+2)h			10C:	--49	"I"
(P+3)h	1	Major version number, ASCII	10D:	--31	"1"
(P+4)h	1	Minor version number, ASCII	10E:	--33	"3"
(P+5)h	4	Optional feature and command support (1=yes, 0=no)	10F:	--E6	
(P+6)h		*bits 10–31 are reserved; undefined bits are "0." If bit 31 is*	110:	--03	
(P+7)h		*"1" then another 31 bit field of Optional features follows at*	111:	--00	
(P+8)h		*the end of the bit–30 field.*	112:	--00	
		bit 0 Chip erase supported	bit 0	= 0	No
		bit 1 Suspend erase supported	bit 1	= 1	Yes
		bit 2 Suspend program supported	bit 2	= 1	Yes
		bit 3 Legacy lock/unlock supported	bit 3	= 0	No
		bit 4 Queued erase supported	bit 4	= 0	No
		bit 5 Instant individual block locking supported	bit 5	= 1	Yes
		bit 6 Protection bits supported	bit 6	= 1	Yes
		bit 7 Pagemode read supported	bit 7	= 1	Yes
		bit 8 Synchronous read supported	bit 8	= 1	Yes
		bit 9 Simultaneous operations supported	bit 9	= 1	Yes
		bit 10 Reserved	bit 10	= 0	No
		bit 30 CFI Link(s) to follow	bit 30	= 0	No
		bit 31 Another "Optional Features" field to follow	bit 31	= 0	No
(P+9)h	1	Supported functions after suspend: read Array, Status, Query	113:	--01	
		Other supported operations are:			
		bits 1–7 reserved; undefined bits are "0"			
		bit 0 Program supported after erase suspend	bit 0	= 1	Yes
(P+A)h	2	Block status register mask	114:	--03	
(P+B)h		*bits 2–15 are Reserved; undefined bits are "0"*	115:	--00	
		bit 0 Block Lock-Bit Status register active	bit 0	= 1	Yes
		bit 1 Block Lock-Down Bit Status active	bit 1	= 1	Yes
(P+C)h	1	V_{CC} logic supply highest performance program/erase voltage	116:	--18	1.8V
		bits 0–3 BCD value in 100 mV			
		bits 4–7 BCD value in volts			
(P+D)h	1	V_{PP} optimum program/erase supply voltage	117:	--90	9.0V
		bits 0–3 BCD value in 100 mV			
		bits 4–7 HEX value in volts			

Table 36: Protection Register Information

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

Table 37: Burst Read Information

Offset[1] P= 10Ah	Length	Description (Optional flash features and commands)	Add.	Hex Code	Value
(P+1D)h	1	Page Mode Read capability bits 0–7 = "n" such that 2^n HEX value represents the number of read-page bytes. See offset 28h for device word width to determine page-mode data output width. 00h indicates no read page buffer.	127:	--00	0 byte
(P+1E)h	1	Number of synchronous mode read configuration fields that follow. 00h indicates no burst capability.	128:	--04	4
(P+1F)h	1	Synchronous mode read capability configuration 1 Bits 3–7 = Reserved bits 0–2 "n" such that 2^{n+1} HEX value represents the maximum number of continuous synchronous reads when the device is configured for its maximum word width. A value of 07h indicates that the device is capable of continuous linear bursts that will output data until the internal burst counter reaches the end of the device's burstable address space. This field's 3-bit value can be written directly to the Read Configuration Register bits 0–2 if the device is configured for its maximum word width. See offset 28h for word width to determine the burst data output width.	129:	--01	4
(P+20)h	1	Synchronous mode read capability configuration 2	12A:	--02	8
(P+21)h	1	Synchronous mode read capability configuration 3	12B:	--03	16
(P+22)h	1	Synchronous mode read capability configuration 4	12C:	--07	Cont

Table 38: Partition and Erase-block Region Information

Offset[1] P= 10Ah		Description	See table below		
				Address	
Bottom	Top	(Optional flash features and commands)	Len	Bot	Top
(P+23)h	(P+23)h	Number of device hardware-partition regions within the device. x = 0: a single hardware partition device (no fields follow). x specifies the number of device partition regions containing one or more contiguous erase block regions.	1	12D:	12D:

Table 39: Partition Region 1 Information

Offset[1] P= 10Ah		Description	See table below		
				Address	
Bottom	Top	(Optional flash features and commands)	Len	Bot	Top
(P+24)h (P+25)h	(P+24)h (P+25)h	Number of identical partitions within the partition region	2	12E: 12F:	12E: 12F:
(P+26)h	(P+26)h	Number of program or erase operations allowed in a partition bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	130:	130:
(P+27)h	(P+27)h	Simultaneous program or erase operations allowed in other partitions while a partition in this region is in Program mode bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	131:	131:
(P+28)h	(P+28)h	Simultaneous program or erase operations allowed in other partitions while a partition in this region is in Erase mode bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	132:	132:
(P+29)h	(P+29)h	Types of erase block regions in this Partition Region. x = 0 = no erase blocking; the Partition Region erases in bulk x = number of erase block regions w/ contiguous same-size erase blocks. Symmetrically blocked partitions have one blocking region. Partition size = (Type 1 blocks)x(Type 1 block sizes) + (Type 2 blocks)x(Type 2 block sizes) +…+ (Type n blocks)x(Type n block sizes)	1	133:	133:
(P+2A)h (P+2B)h (P+2C)h (P+2D)h	(P+2A)h (P+2B)h (P+2C)h (P+2D)h	Partition Region 1 Erase Block Type 1 Information bits 0–15 = y, y+1 = # identical-size erase blks in a partition 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 = # identical-size erase blks in a partition 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 40: Partition Region 2 Information

Offset[1] P= 10Ah		Description	See table below		
				Address	
Bottom	Top	(Optional flash features and commands)	Len	Bot	Top
(P+3A)h (P+3B)h	(P+32)h (P+33)h	Number of identical partitions within the partition region	2	144: 145:	13C: 13D:
(P+3C)h	(P+34)h	Number of program or erase operations allowed in a partition bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	146:	13E:
(P+3D)h	(P+35)h	Simultaneous program or erase operations allowed in other partitions while a partition in this region is in Program mode bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	147:	13F:
(P+3E)h	(P+36)h	Simultaneous program or erase operations allowed in other partitions while a partition in this region is in Erase mode bits 0–3 = number of simultaneous Program operations bits 4–7 = number of simultaneous Erase operations	1	148:	140:
(P+3F)h	(P+37)h	Types of erase block regions in this Partition Region. x = 0 = no erase blocking; the Partition Region erases in bulk x = number of erase block regions w/ contiguous same-size erase blocks. Symmetrically blocked partitions have one blocking region. Partition size = (Type 1 blocks)x(Type 1 block sizes) + (Type 2 blocks)x(Type 2 block sizes) +…+ (Type n blocks)x(Type n block sizes)	1	149:	141:
(P+40)h (P+41)h (P+42)h (P+43)h	(P+38)h (P+39)h (P+3A)h (P+3B)h	Partition Region 2 Erase Block Type 1 Information bits 0–15 = y, y+1 = # identical-size erase blks in a partition 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 = # identical-size erase blks in a partition 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 41: Partition and Erase Block 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:	--02	--02	--02	--02	--02	--02
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:	--02	--00	--02	--00	--02	--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	--02	--01	--02	--01	--02
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:	--02	--02	--02	--02	--02	--02

Table 42: Electrical Traceability

76h	76h	Bits 0 – 2: Stepping (See Table 3 and 4) Bit 3: Production Bit (See Table 3 and 4) Bits 4 – 7: Memory clock rate 0000 = 40MHz 0001 = 54MHz 0010 = 66MHz 1XXX = DDR Bits 8 – 9: Process Bit 10: Mass memory 0 = No 1 = Yes Bits 11 - 12: Ram Type 00 = No 01 = SRAM 10 = PSRAM 11 = DRAM Bits 13 – 15: RAM density 000 = no SRAM 001 = 4 Mb 010 = 8 Mb 011 = 16 Mb 100 = 32 Mb…	1	76h	76h
(P+48)h	(P+48)h	Reserved for future use	Resv'd	152:	152:

Table 43: CFI Revision History for Engineering Sample at Address 76h

Density	CFI Field Data	xxxxh	Revision	Comments
OMPU 64 Mbit - Bin 1	000 00 0 00 0001 **1100**	001Ch	Revision 2	A2 Silicon (128Mb A2 Silicon)
OMPU 64 Mbit - Bin 1	000 00 0 00 0001 **1011**	001Bh	Revision 3	A3 Silicon (128Mb A3 Silicon)
64 Mbit - Bin 1	000 00 0 00 0001 **1110**	001Eh	Revision 0	A0 Silicon
64 Mbit - Bin 1	000 00 0 00 0001 **1101**	001Dh	Revision 1	A1 Silicon
128 Mbit - Bin 1	000 00 0 00 0001 **1110**	001Eh	Revision 0	A0 Silicon
128 Mbit - Bin 1	000 00 0 00 0001 **1101**	001Dh	Revision 1	A1 Silicon
128 Mbit - Bin 1	000 00 0 00 0001 **1100**	001Ch	Revision 2	A2 Silicon
128 Mbit - Bin 1	000 00 0 00 0001 **1011**	001Bh	Revision 3	A3 Silicon
128 Mbit - Bin 2	000 00 0 00 0000 **1110**	000Eh	Revision 0	A0 Silicon
128 Mbit - Bin 2	000 00 0 00 0000 **1101**	000Dh	Revision 1	A1 Silicon
128 Mbit - Bin 2	000 00 0 00 0000 **1100**	000Ch	Revision 2	A2 Silicon
128 Mbit - Bin 2	000 00 0 00 0000 **1011**	000Bh	Revision 3	A3 Silicon
256 Mbit - Bin 1	000 00 0 00 0001 **1110**	001Eh	Revision 0	A0 Silicon
256 Mbit - Bin 1	000 00 0 00 0001 **1101**	001Dh	Revision 1	A1 Silicon
256 Mbit - Bin 2	000 00 0 00 0000 **1110**	000Eh	Revision 0	A0 Silicon
256 Mbit - Bin 2	000 00 0 00 0000 **1101**	000Dh	Revision 1	A1 Silicon

Table 44: CFI Revision History for Production Materials at Address 76h

Density	CFI Field Data	xxxxh	Revision	Comments
OMPU 64 Mbit - Bin 1	000 00 0 00 0001 **0**001	0011h	Revision 1	Production Silicon (128Mb Prod Silicon Rev)
OMPU 64 Mbit - Bin 1	000 00 0 00 0001 **0**010	0012h	Revision 2	Rev 2 If Errata
64 Mbit - Bin 1	000 00 0 00 0001 **0**001	0011h	Revision 1	Production Silicon
64 Mbit - Bin 1	000 00 0 00 0001 **0**010	0012h	Revision 2	Rev 2 If Errata
64 Mbit - Bin 1	000 00 0 00 0001 **0**011	0013h	Revision 3	Rev 3 If Errata
128 Mbit - Bin 1	000 00 0 00 0001 **0**001	0011h	Revision 1	Production Silicon
128 Mbit - Bin 1	000 00 0 00 0001 **0**010	0012h	Revision 2	Rev 2 If Errata
128 Mbit - Bin 1	000 00 0 00 0001 **0**011	0013h	Revision 3	Rev 3 If Errata
128 Mbit - Bin 2	000 00 0 00 0000 **0**001	0001h	Revision 1	Production Silicon
128 Mbit - Bin 2	000 00 0 00 0000 **0**010	0002h	Revision 2	Rev 2 If Errata
128 Mbit - Bin 2	000 00 0 00 0000 **0**011	0003h	Revision 3	Rev 3 If Errata
256 Mbit - Bin 1	000 00 0 00 0001 **0**001	0011h	Revision 1	Production Silicon
256 Mbit - Bin 1	000 00 0 00 0001 **0**010	0012h	Revision 2	Rev 2 If Errata
256 Mbit - Bin 1	000 00 0 00 0001 **0**011	0013h	Revision 3	Rev 3 If Errata
256 Mbit - Bin 2	000 00 0 00 0000 **0**001	0001h	Revision 1	Production Silicon
256 Mbit - Bin 2	000 00 0 00 0000 **0**010	0002h	Revision 2	Rev 2 If Errata
256 Mbit - Bin 2	000 00 0 00 0000 **0**011	0003h	Revision 3	Rev 3 If Errata

Appendix C Ordering Information

To order samples, obtain datasheets or inquire about any stack combination, please contact your local Intel representative.

Table 45: 38F Type Stacked Components

PF	38F	5070	M0	Y	0	B	0
Package Designator	Product Line Designator	Product Die/ Density Configuration	NOR Flash Product Family	Voltage/NOR Flash CE# Configuration	Parameter / Mux Configuration	Ballout Identifier	Device Details
PF = SCSP, RoHS RD = SCSP, Leaded	Stacked NOR Flash + RAM	Char 1 = Flash die #1 Char 2 = Flash die #2 Char 3 = RAM die #1 Char 4 = RAM die #2 (See Table 47, "38F / 48F Density Decoder" on page 97 for details)	First character applies to Flash die #1 Second character applies to Flash die #2 (See Table 48, "NOR Flash Family Decoder" on page 97 for details)	V = 1.8 V Core and I/O; Separate Chip Enable per die (See Table 49, "Voltage / NOR Flash CE# Configuration Decoder" on page 97 for details)	0 = No parameter blocks; Non-Mux I/O interface (See Table 50, "Parameter / Mux Configuration Decoder" on page 98 for details)	B = x16D Ballout (See Table 51 "Ballout Decoder" on page 98 for details)	0 = Original released version of this product

Table 46: 48F Type Stacked Components

PC	48F	4400	P0	V	B	0	0
Package Designator	Product Line Designator	Product Die/ Density Configuration	NOR Flash Product Family	Voltage/NOR Flash CE# Configuration	Parameter / Mux Configuration	Ballout Identifier	Device Details
PC = Easy BGA, RoHS RC = Easy BGA, Leaded JS = TSOP, RoHS TE = TSOP, Leaded PF = SCSP, RoHS RD = SCSP, Leaded	Stacked NOR Flash only	Char 1 = Flash die #1 Char 2 = Flash die #2 Char 3 = Flash die #3 Char 4 = Flash die #4 (See Table 47, "38F / 48F Density Decoder" on page 97 for details)	First character applies to Flash dies #1 and #2 Second character applies to Flash dies #3 and #4 (See Table 48, "NOR Flash Family Decoder" on page 97 for details)	V = 1.8 V Core and 3 V I/O; Virtual Chip Enable (See Table 49, "Voltage / NOR Flash CE# Configuration Decoder" on page 97 for details)	B = Bottom parameter; Non-Mux I/O interface (See Table 50, "Parameter / Mux Configuration Decoder" on page 98 for details)	0 = Discrete Ballout (See Table 51 "Ballout Decoder" on page 98 for details)	0 = Original released version of this product

Table 47: 38F / 48F Density Decoder

Code	Flash Density	RAM Density
0	No Die	No Die
1	32-Mbit	4-Mbit
2	64-Mbit	8-Mbit
3	128-Mbit	16-Mbit
4	256-Mbit	32-Mbit
5	512-Mbit	64-Mbit
6	1-Gbit	128-Mbit
7	2-Gbit	256-Mbit
8	4-Gbit	512-Mbit
9	8-Gbit	1-Gbit
A	16-Gbit	2-Gbit
B	32-Gbit	4-Gbit
C	64-Gbit	8-Gbit
D	128-Gbit	16-Gbit
E	256-Gbit	32-Gbit
F	512-Gbit	64-Gbit

Table 48: NOR Flash Family Decoder

Code	Family	Marketing Name
C	C3	Intel Advanced+ Boot Block Flash Memory
J	J3v.D	Intel Embedded Flash Memory
L	L18 / L30	Intel StrataFlash® Wireless Memory
M	M18	Intel StrataFlash® Cellular Memory
P	P30 / P33	Intel StrataFalsh® Embedded Memory
W	W18 / W30	Intel Wireless Flash Memory
0(zero)	-	No Die

Table 49: Voltage / NOR Flash CE# Configuration Decoder (Sheet 1 of 2)

Code	I/O Voltage (Volt)	Core Voltage (Volt)	CE# Configuration
Z	3.0	1.8	Seperate Chip Enable per die
Y	1.8	1.8	Seperate Chip Enable per die
X	3.0	3.0	Seperate Chip Enable per die
V	3.0	1.8	Virtual Chip Enable
U	1.8	1.8	Virtual Chip Enable
T	3.0	3.0	Virtual Chip Enable

Table 49: Voltage / NOR Flash CE# Configuration Decoder (Sheet 2 of 2)

Code	I/O Voltage (Volt)	Core Voltage (Volt)	CE# Configuration
R	3.0	1.8	Virtual Address
Q	1.8	1.8	Virtual Address
P	3.0	3.0	Virtual Address

Table 50: Parameter / Mux Configuration Decoder

Code, Mux Identification	Number of Flash Die	Bus Width	Flash Die 1	Flash Die 2	Flash Die 3	Flash Die 4
0 = Non Mux 1 = AD Mux[1] 2= AAD Mux 3 =Full" AD Mux[2]	Any	NA	Notation used for stacks that contain no parameter blocks			
B = Non Mux C = AD Mux F = "Full" Ad Mux	1	X16	Bottom	-	-	-
	2		Bottom	Top	-	-
	3		Bottom	Bottom	Top	-
	4		Bottom	Top	Bottom	Top
	2	X32	Bottom	Bottom	-	-
	4		Bottom	Bottom	Top	Top
T = Non Mux U = AD Mux W = "Full" Ad Mux	1	X16	Top	-	-	-
	2		Top	Bottom	-	-
	3		Top	Top	Bottom	-
	4		Top	Bottom	Top	Bottom
	2	X32	Top	Top	-	-
	4		Top	Top	Bottom	Bottom

1. Only Flash is Muxed and RAM is non-Muxed
2. Both Flash and RAM are AD-Muxed

Table 51: Ballout Decoder

Code	Ballout Definition
0 (Zero)	SDiscrete ballout (Easay BGA and TSOP)
B	x16D ballout, 105 ball (x16 NOR + NAND + DRAM Share Bus)
C	x16C ballout, 107 ball (x16 NOR + NAND + PSRAM Share Bus)
Q	QUAD/+ ballout, 88 ball (x16 NOR + PSRAM Share Bus)
U	x32SH ballout, 106 ball (x32 NOR only Share Bus)
V	x16SB ballout, 165 ball (x16 NOR / NAND + x16 DRAM Split Bus
W	x48D ballout, 165 ball (x16/x32 NOR + NAND + DRAM Split Bus