S26KL512S / S26KS512S
S26KL256S / S26KS256S
S26KL128S / S26KS128S

512 Mbit (64 Mbyte), 256 Mbit (32 Mbyte), 128 Mbit (16 Mbyte) 1.8V/3.0V HyperFlash™ Family

Distinctive Characteristics

HyperBus^{TM} Low Signal Count Interface

- 3.0V I/O, 11 bus signals
  - Single ended clock
- 1.8V I/O, 12 bus signals
  - Differential clock (CK, CK#)
- Chip Select (CS#)
- 8-bit data bus (DQ[7:0])
- Read-Write Data Strobe (RWDS)
  - HyperFlash memories use RWDS only as a Read Data Strobe

High Performance

- Up to 333 MB/s sustained read throughput
- Double-Data Rate (DDR) – two data transfers per clock
- 166-MHz clock rate (333 MB/s) at 1.8V V_{CC}
- 100-MHz clock rate (200 MB/s) at 3.0V V_{CC}
- 96-ns initial random read access time
  - Initial random access read latency: 5 to 16 clock cycles
- Sequential burst transactions
- Configurable Burst Characteristics
  - Wrapped burst lengths:
    - 16 bytes (8 clocks)
    - 32 bytes (16 clocks)
    - 64 bytes (32 clocks)
  - Linear burst
  - Hybrid option — one wrapped burst followed by linear burst
  - Wrapped or linear burst type selected in each transaction
  - Configurable output drive strength
- Low Power Modes
  - Active Clock Stop During Read: 12 mA, no wake-up required
  - Standby: 25 µA (typical), no wake-up required
  - Deep Power-Down: 8 µA (typical)
    - 300 µs wake-up required
- INT# output to generate external interrupt
  - Busy to Ready Transition
- RSTO# output to generate system level power-on reset
  - User configurable RSTO# Low period
- 512-byte Program Buffer
  - Programming in 16-byte half-page multiples, up to a maximum of 512-bytes
- Sector Erase
  - Uniform 256-kB sectors
  - Optional Eight 4-kB Parameter Sectors (32 kB total)
- Advanced Sector Protection
  - Volatile and non-volatile protection methods for each sector
- Separate 1024-byte one-time program array
- Operating Temperature
  - Industrial (–40°C to +85°C)
  - Industrial Plus (–40°C to +105°C)
  - Extended (–40°C to +125°C)
- ISO/TS16949 and AEC Q100 Certified
- Endurance
  - 100,000 cycle endurance for any sector (minimum)
- Retention
  - 20 year data retention (typical)
- Power-On Reset, Erase and Program Current
  - Max Peak ≤ 100 mA
- Packaging Options
  - 24-Ball FBGA

Performance Summary

Read Access Timings	
Maximum Clock Rate at 1.8V V_{CC}/$V_{CC}Q$	166 MHz
Maximum Clock Rate at 3.0V V_{CC}/$V_{CC}Q$	100 MHz
Maximum Access Time, (t_{ACC})	96 ns
Maximum CS# Access Time to First Word @ 166 MHz	118 ns

Typical Program / Erase Times	
Single Word Programming (2B = 16b)	500 µs (~4 kB/s)
Write Buffer Programming (512B = 4096b)	475 µs (~1 MB/s)
Sector Erase Time (256 kB = 2 Mb)	930 ms (~282 kB/s)

Typical Current Consumption	
Burst Read (Continuous Read at 166 MHz)	80 mA
Power-On Reset	80 mA
Sector Erase Current	60 mA
Write Buffer Programming Current	60 mA
Standby (CS# = High)	25 µA
Deep Power-Down (CS# = High, 85°C)	30 µA (512 Mb)
	4 µA (all other densities)

Contents

Software Interface

Hardware Interface

1. General Description

The Cypress HyperFlash family of products are high-speed CMOS, MirrorBit® NOR flash devices with the HyperBus low signal count DDR (Double Data Rate) interface, that achieves high speed read throughput. The DDR protocol transfers two data bytes per clock cycle on the data (DQ) signals. A read or write access for the HyperFlash consists of a series of 16-bit wide, one clock cycle data transfers at the internal HyperFlash core and two corresponding 8-bit wide, one-half-clock-cycle data transfers on the DQ signals.

Both data and command/address information are transferred in DDR fashion over the 8-bit data bus. The clock input signals are used for signal capture by the HyperFlash device when receiving command/address/data information on the DQ signals. The Read Write Data Strobe (RWDS) is an output from the HyperFlash device that indicates when data is being transferred from the memory to the host. RWDS is referenced to the rising and falling edges of CK during the data transfer portion of read operations. Command/ address/write-data values are center aligned with the clock edges and read-data values are edge aligned with the transitions of RWDS.

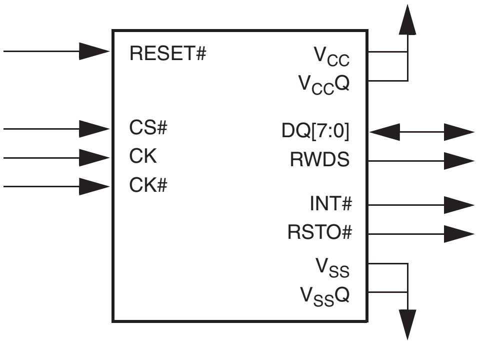
Read and write operations to the HyperFlash device are burst oriented. Read transactions can be specified to use either a wrapped or linear burst. During wrapped operation, accesses start at a selected location and continue for a configured number of locations in a group wrap sequence. During linear operation accesses start at a selected location and continue in a sequential manner until the read operation is terminated, when CS# returns High. Write transactions transfer a one or more16-bit values.

All inputs are LV-CMOS compatible. Devices have a V_{CC} and $V_{CC}Q$ supply of 1.8V or 3.0V (nominal). For additional information on HyperBus interface operation, please refer to the HyperBus Specification. This data sheet describes only the HyperFlash device dependent aspects of HyperBus interface operation.

2. Product Overview

The HyperFlash family consists of multiple densities, 1.8V or 3.0V core and I/O, non-volatile, synchronous flash memory devices. These devices have an 8-bit (1-byte) wide DDR data bus and use only word wide (16-bit data) address boundaries. Read operations provide 16 bits of data during each clock cycle (8 bits on each clock edge). Write operations take 16 bits of data from each clock cycle (8 bits on each clock edge).

Figure 2.1 HyperFlash Interface



HyperFlash memories use the HyperBus low signal count, high-speed interface and combine the best features of Execute-In-Place (XIP) and Data Storage flash memories. This family has the fast random access of XIP flash along with the high density and fast program speed of Data Storage flash.

Each random read accesses a 32-byte length and aligned set of data called a page. Each page consists of a pair of 16-byte aligned groups of array data called half-pages. Half-pages are aligned on 16-byte address boundaries. A read access requires two clock cycles to define the target half-page address and the burst type, then an additional initial latency of t_{ACC}. During the initial latency period the third clock cycle will specify the starting address within the target half-page. After the initial data value has been output, additional data can be read from the Page on subsequent clock cycles in either a wrapped or linear manner. When configured in linear burst mode, while a page is being burst out, the device will automatically fetch the next sequential page from the MirrorBit flash memory array. This simultaneous burst output while fetching from the array allows for a linear sequential burst operation that can provide a sustained output of 333 MB/s data rate (1-byte (8-bit data bus) * 2 (Data on both clock edges) * 166 MHz = 333 MB/s).

S26KL/S26KS Address Map

Type	Count	Addresses	Notes
Word Address within a half-page (16 byte)	8 (word addresses)	A2 – A0	16 bytes
Word Address within Write Buffer Line (512 byte)	256 (word addresses)	A7 – A0	512 bytes
Half-pages (16 bytes) within Erase Sector (256 kB)	8192 (half-pages)	A16 – A3	
Write Buffer Lines (512 bytes) within Erase Sector (256 kB)	512 (lines)	A16 – A8	
Total Number of Erase Sectors (256 kB)	256 (512 Mb) 128 (256 Mb) 64 (128 Mb)	Amax – A17	

The device control logic is subdivided into two parallel operating sections: the Host Interface Controller (HIC) and the Embedded Algorithm Controller (EAC). The HIC monitors signal levels on the device inputs and drives outputs as needed to complete read and write data transfers with the host system (HyperBus master). The HIC delivers data from the currently entered address map on read transfers; places write transfer address and data information into the EAC command memory; notifies the EAC of power transition, and write transfers. The EAC looks in the command memory, after a write transfer, for legal command sequences and performs the related Embedded Algorithms.

Changing the non-volatile data in the memory array requires a complex sequence of operations that are called Embedded Algorithms (EA). The algorithms are managed entirely by the internal EAC. The main algorithms perform programming and erase of the main flash array data. The host system writes command codes to the flash device address space. The EAC receives the command, performs all the necessary steps to complete the command, and provides status information during the progress of an EA.

The erased state of each memory bit is a logic 1. Programming changes a logic 1 (High) to a logic 0 (Low). Only an erase operation is able to change a 0 to a 1. An erase operation must be performed on an entire 256-kbyte (or 4-kbyte for parameter sectors) aligned group of data called a Sector. When shipped from Cypress all Sectors are erased.

Programming is done via a 512-byte Write Buffer. It is possible to write from one to 256 words, anywhere within the Write Buffer before starting a programming operation. Within the flash memory array, each 512-byte aligned group of data is called a Line. A programming operation transfers data from the volatile Write Buffer to a non-volatile memory array Line. The operation is called Write Buffer Programming.

The Write Buffer is filled with 1s after reset or the completion of any operation using the Write Buffer. Any locations not written to a 0 by a Write to Buffer command are by default still filled with 1s. Any 1s in the Write Buffer do not affect data in the memory array during a programming operation.

In addition to the mandatory signals dedicated to the HyperBus, the device also includes a Reset (RESET#) input, an Interrupt (INT#) output, and a Reset Output (RSTO#).

When RESET# transitions from Low to High the device returns to the default state that occurs after an internal Power-On Reset (POR).

The INT# output can provide an interrupt to the HyperBus master to indicate when the HyperFlash transitions from busy to ready at the end of a program or erase operation.

The RSTO# is an open-drain output used to indicate when a POR is occurring within the device and can be used as a system level reset signal. Upon completion of the internal POR the RSTO# signal will transition from Low to high impedance after a user defined timeout period has expired. Upon transition to the high impedance state the external pull-up resistance will pull RSTO# High and the device immediately is placed into the Idle state.

3. Signal Descriptions

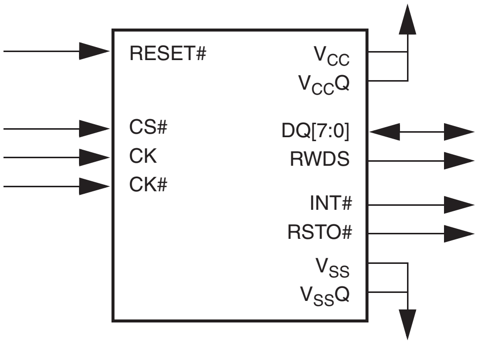


Figure 3.1 HypeFlash Interface

Signal Descriptions

Symbol	Type	Description
CS#	Input	**Chip Select.** HyperFlash bus transactions are initiated with a High to Low transition. HyperFlash bus transactions are terminated with a Low to High transition.
CK, CK#	Input	**Differential Clock.** Command / Address / Data information is input or output with respect to the crossing of the CK and CK# signals. CK# is only used on the 1.8V devices and may be left open or connected to CK on 3V devices.
RWDS	Output	**Read Write Data Strobe.** Output data during read transactions are edge aligned with RWDS.
DQ[7..0]	Input / Output	**Data Input / Output.** Command / Address / Data information is transferred on these DQs during read and write transactions.
INT#	Output (open drain)	**INT Output.** When Low, the device is indicating that an internal event has occurred. This signal is intended to be used as a system level interrupt for the device to indicate that an on-chip event has occurred. INT# is an open-drain output.
RESET#	Input	**Hardware Reset.** When Low, the device will self initialize and return to the array read state. RWDS and DQ[7:0] are placed into the High-Z state when RESET# is Low. RESET# includes a weak pull-up, if RESET# is left unconnected it will be pulled up to the High state.
RSTO#	Output (open drain)	**RSTO# Output.** RSTO# is an open-drain output used to indicate when a POR is occurring within the device and can be used as a system level reset signal. Upon completion of the internal POR the RSTO# signal will transition from Low to high impedance after a user defined timeout period has elapsed. Upon transition to the high impedance state the external pull-up resistance will pull RSTO# High and the device immediately is placed into the Idle state.
V_{CC}	Power Supply	**Core Power.**
$V_{CC}Q$	Power Supply	**Input / Output Power.**
V_{SS}	Power Supply	**Core Ground.**
$V_{SS}Q$	Power Supply	**Input / Output Ground.**

4. HyperBus Protocol

All bus transactions can be classified as either read or write. A bus transaction is started with CS# going Low with CK = Low and CK# = High. The transaction to be performed is presented to the HyperFlash device during the first three clock cycles in a DDR manner using all six clock edges. These first three clocks transfer three words of Command / Address (CA0, CA1, CA2) information to define the transaction characteristics:

- Read or write transaction.
- Whether the transaction will be to the memory array or to register space.
  - Although the HyperBus protocol provides for slave devices that have both memory and register address spaces, HyperFlash memories described in this specification do not differentiate between memory and registers as separate address spaces. There is a single address space selected by any transaction, independent of whether the transaction indicates the target location is in memory space or register space. Write transactions always place the transaction address and data into a a command register set (buffer). Read transactions return data from the memory array or from a register address space window that has been temporarily overlaid within the single address space by the execution of commands. The single address space with register space overlays methodology is backward compatible with legacy parallel NOR Flash memory program and erase software drivers.
- Whether a read transaction will use a linear or wrapped burst sequence.
  - Write command transactions transfer a single word per write. The Word Program write data transfer may be done with a linear burst up to 50 MHz. Write transactions do not support a wrapped burst sequence and ignore the burst type indication.
- The target half-page address (row and upper order column address).
- The target Word (within half-page) address (lower order column address).

Once the transaction has been defined, a number of idle clock cycles are used to satisfy any read latency requirements before data is transferred. Once the target data has been transferred the HyperBusmaster host completes the transaction by driving CS# High with CK = Low and CK# = High. Data is transferred as 16-bit values with the first eight bits (15-8) transferred on a High going CK (write data or CA bits) or RWDS edge (read data) and the second eight bits (7-0) being transferred on the Low going CK or RWDS edge. Data transfers during read or write operations can be ended at any time by bringing CS# High when CK = Low and CK# = High. Read data is edge aligned with RWDS transitions and Write data is center aligned with clock edges.

4.1 Command / Address Bit Assignments

Command / Address Bit Assignments

CA Bit#	Bit Name	Bit Function
47	R/W#	Identifies the transaction as a Read or Write. 1 = Read operation 0 = Write operation Target space is defined in CA46.
46	Target	Indicates whether the Read or Write operation accesses the memory or register spaces. 0 = memory space 1 = register space The register space is intended to be used by volatile memory and peripheral devices. The HyperFlash devices will not take advantage of this feature and this bit should be set to 0 during Read or Write transactions.
45	Burst Type	Indicates whether the burst will be linear or wrapped. 0 = Wrapped Burst 1 = Linear Burst
44-38 (512 Mb) 44-37 (256 Mb) 44-36 (128 Mb)	Reserved	Reserved for future address expansion. Reserved bits should be set to 0 by the host controller.
37-16 (512 Mb) 36-16 (256 Mb) 35-16 (128 Mb)	Row and Upper Column Address	Half page component of target address.
15-3	Reserved	Reserved for future column address expansion. Reserved bits should be set to 0 by the host controller.
2-0	Lower Column Address	Lower Column component of the target address: System word address bits A2-0 selecting the starting word within a half-page.

4.2 Read Operations

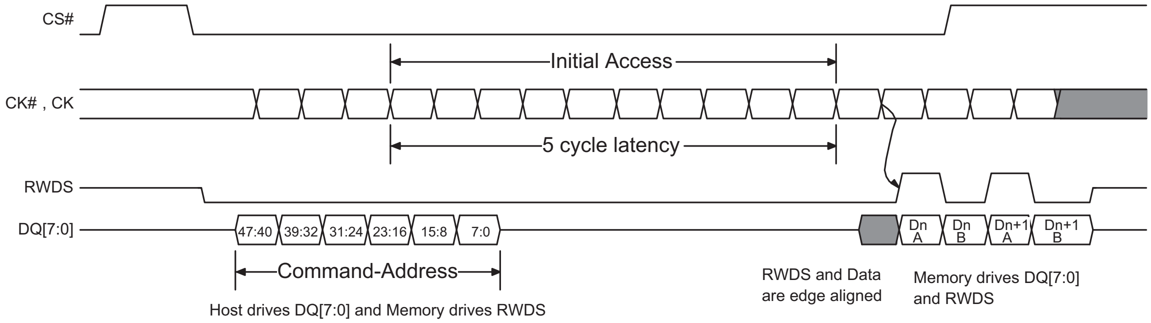
CA0 indicates that a read operation is to be performed and also indicates the burst type (wrapped or linear). Read operations begin the internal array access as soon as the half-page address has been presented in CA0 and CA1. CA2 identifies the target word address within the chosen half-page. The host then continues clocking for a number of cycles defined by the latency count setting in the Configuration Register. Once these latency clocks have been completed the memory starts to simultaneously transition the Read Write Data Strobe (RWDS) and begins outputting the target data. New data is output in an edge aligned fashion upon every transition of RWDS. Data will continue to be output as long as the host continues to transition the clock (CK and CK#). Wrapped bursts will continue to wrap within the burst length and linear burst will output data in a sequential manner across page boundaries. A Hybrid Burst provides one initial wrapped burst followed by linear burst, as described in Section 6.3.11, *Hybrid Burst* on page 48. Wrapped reads can be performed from the main array, the CFI Tables and the Secure Silicon Region. Read transfers can be ended at any time by bringing CS# High when CK = Low and CK# = High.

During linear read operations a number of additional latency cycles may be required when crossing the boundary between the target address page and the next sequential page (Figure on page 11, Figure on page 11). The number of latency cycles depends on the starting address within the target page and the number of initial latency cycles defined in the Configuration Register.

When a linear burst reaches the last address in the array the burst will wrap to address 0.

16-byte and 32-byte wrapped bursts do not cross page boundaries and do not incur inter-page boundary crossing latencies. For a 64-byte wrapped burst read, a latency may occur during the target address to next page boundary crossing, depending on the starting address (Table , *64-byte Wrapped Burst Address Sequence (Latency Code = 16)* on page 32).

Figure 4.1 Read Operation



Notes:
1. *Transactions must be initiated with CK = Low and CK# = High. CS# must return High before a new transaction is initiated.*
2. *Read access from the flash array starts once CA[23:16] is captured.*
3. *The read latency is defined by the Read Latency value in the Volatile Configuration Register (or the Non-Volatile Configuration Register).*
4. *In this example of a read operation the Latency Count was set to five clocks.*

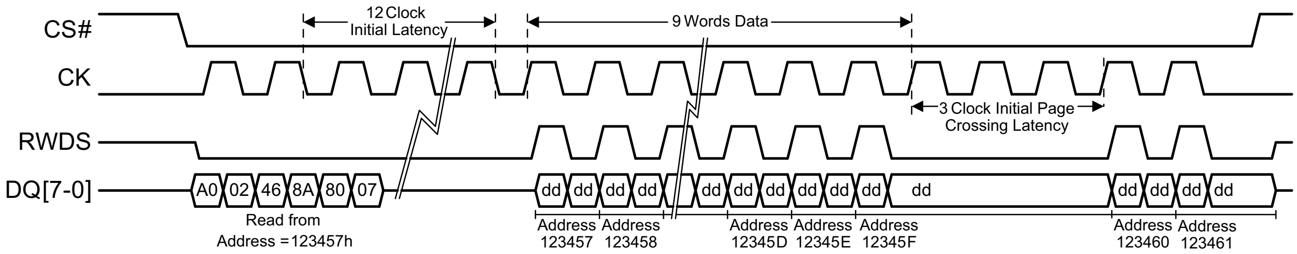
Maximum Operating Frequency for Latency Code Options

Latency Code	Latency Clocks	Maximum Operating Frequency (MHz)
0000	5	52
0001	6	62
0010	7	72
0011	8	83
0100	9	93
0101	10	104
0110	11	114
0111	12	125
1000	13	135
1001	14	145
1010	15	156
1011	16	166
1100	Reserved	NA
1101	Reserved	NA
1110	Reserved	NA
1111	Reserved	NA

Notes:

1. *Default NVCR latency setting when the device is shipped from the factory is 16 clocks.*
2. *The Latency Code is the value loaded into (Non) Volatile Configuration Register bits xVCR[7:4].*
3. *Maximum Operating Frequency assumed to be using a device with t_{ACC} = 96 ns.*

Figure 4.2 Read Transaction Crossing a Page Boundary



Notes:

1. *Read operation starting at device address 123457h.*
2. *Latency code loaded into the Configuration Register is 0111b which results in 12 latency clocks.*
3. *Page boundary crossing requires three clocks in this case. 12 clock initial latency minus 9 clocks (words) of initial data.*
4. *CK# is not shown but is the complement of the CK signal.*
5. *CA45 = 1 for a linear Read burst.*

First Page Boundary Crossing During Linear Read (Latency Count = 11 Clocks)

Target Address	Clock Cycle																						
	0	1	2	3	...	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
0	CA0	CA1	CA2	Bus Turnaround + Initial Latency			D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17
1							D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18
2							D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19
3							D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20
4							D4	D6	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21
5							D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22
6							D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	X	D16	D17	D18	D19	D20	D21	D22
7							D7	D8	D9	D10	D11	D12	D13	D14	D15	X	X	D16	D17	D18	D19	D20	D21	D22
8							D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	D24	D25
9							D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26
10							D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26	D27
11							D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26	D27	D28
12							D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26	D27	D28	D29
13							D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26	D27	D28	D29	D30
14							D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	X	D24	D25	D26	D27	D28	D29	D30
15							D15	D16	D17	D18	D19	D20	D21	D22	D23	X	X	D24	D25	D26	D27	D28	D29	D30
16							D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26	D27	D28	D29	D30	D31	D32	D33
	—	—	1	2	...	11	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	Latency Count																							

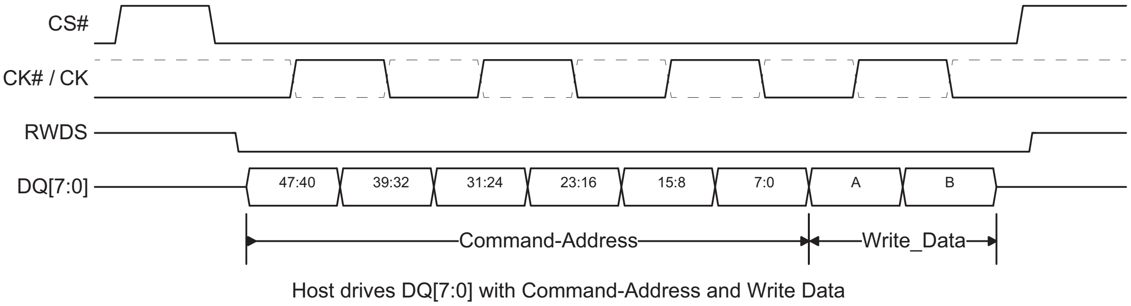
First Page Boundary Crossing During Linear Read (Latency Count = 16 Clocks)

Target Address	Clock Cycle After CS# Goes Low																						
	0	1	2	3	...	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
0	CA0	CA1	CA2	Bus Turnaround + Initial Latency			D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17
1							D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	X	D16	D17
2							D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	X	X	D16	D17
3							D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	X	X	X	D16	D17
4							D4	D6	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	X	X	X	X	D16	D17
5							D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	X	X	X	X	X	D16	D17
6							D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	X	X	X	X	X	X	D16	D17
7							D7	D8	D9	D10	D11	D12	D13	D14	D15	X	X	X	X	X	X	X	D16	D17
8							D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	D24	D25
9							D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	X	D24	D25
10							D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	X	X	D24	D25
11							D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	X	X	X	D24	D25
12							D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	X	X	X	X	D24	D25
13							D13	D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	X	X	X	X	X	D24	D25
14							D14	D15	D16	D17	D18	D19	D20	D21	D22	D23	X	X	X	X	X	X	D24	D25
15							D15	D16	D17	D18	D19	D20	D21	D22	D23	X	X	X	X	X	X	X	D24	D25
16							D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26	D27	D28	D29	D30	D31	D32	D33
	—	—	1	2	...	16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	Latency Count																							

4.3 Write Operations

A write operation starts with the first three clock cycles providing the CAx (Command / Address) information indicating the transaction characteristics. The Burst Type bit CA[45] is 'don't care' because the HyperFlash device only supports a single write transaction of 16b or a continuous linear write burst that is only supported when loading data during a Word Program command. Immediately following the CA information the host is able to transfer the write data on the DQ bus. The first byte (A) of data is presented on the rising edge of CK and the second byte (B) is presented on the falling edge of CK. Write data is center aligned with the CK/CK# inputs. Write transfers can be ended at any time by bringing CS# High when CK = Low and CK# = High.

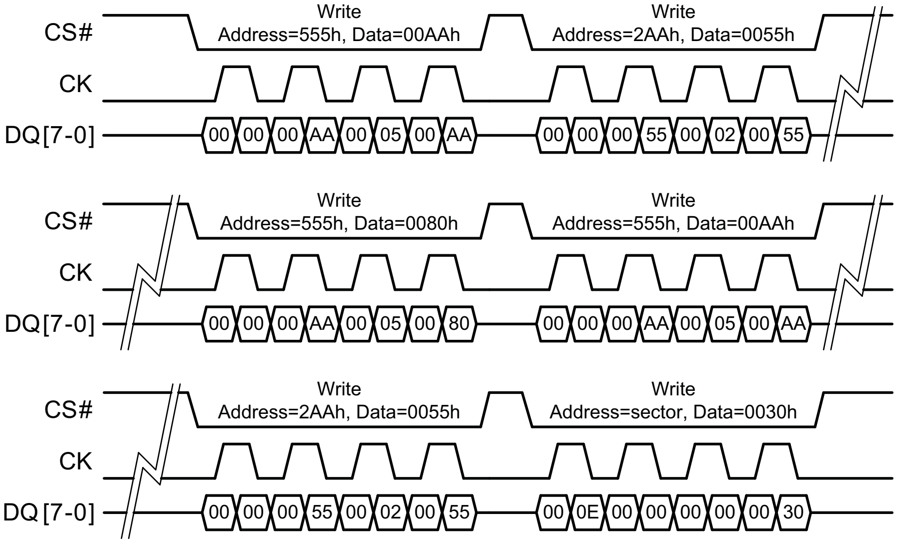
Figure 4.3 Write Operation



Notes:

1. *Transactions must be initiated with CK = Low and CK# = High. CS# must return High before a new transaction is initiated.*
2. *RWDS will be driven Low as long as CS# is Low.*
3. *Write operations are limited to a transaction of a single word (16b) or a linear write burst supported only when loading data during a Word Program command.*

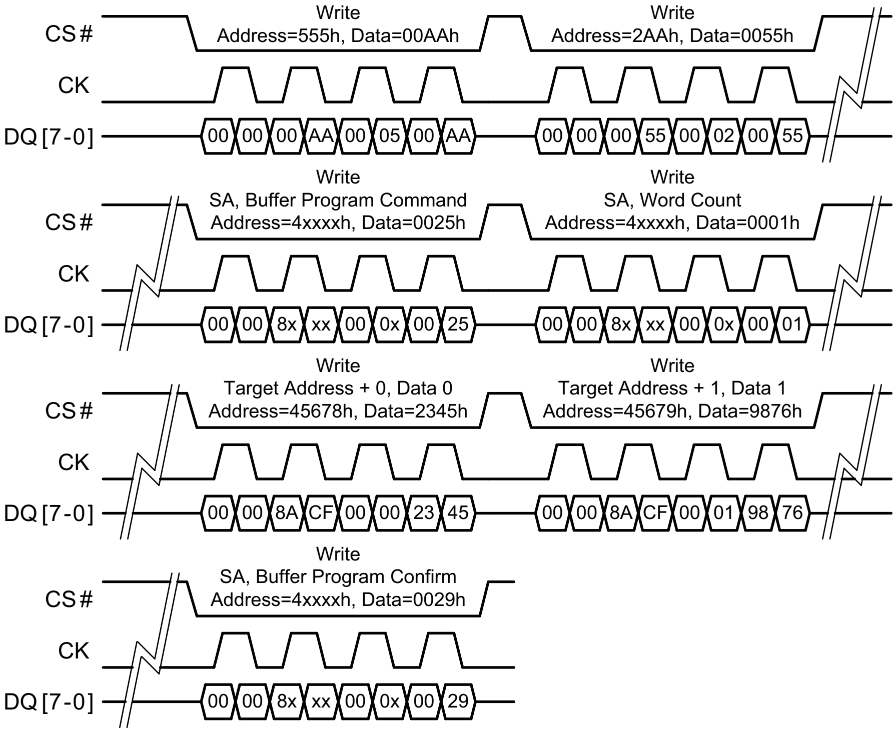
Figure 4.4 Write Transaction Usage Example: Erase Operation Command Sequence



Notes:

1. *See Figure 6.4, Sector Erase Operation on page 28 for the Erase Operation Command Sequence flowchart.*
2. *Erase operation to the sector starting at 0700000h.*
3. *CK# is not shown but is the complement of the CK signal.*
4. *RWDS is not shown and is not used during Write transactions.*

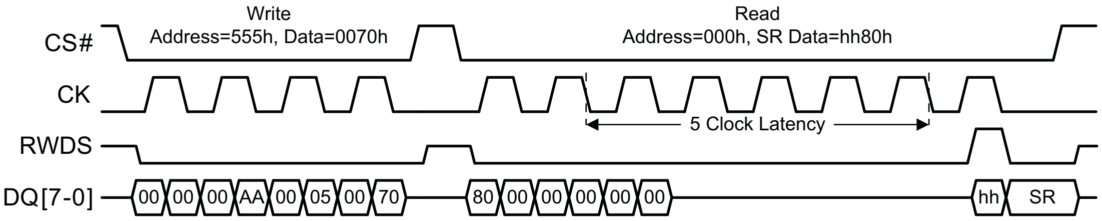
Figure 4.5 Write Transaction Usage Example: Write Buffer Program Command Sequence



Notes:

1. *See Figure 6.2, Write Buffer Programming Operation with Status Register on page 24 for the Write Buffer Program Operation Command Sequence flowchart.*
2. *Program 2345h into address 45678h, and 9876h into address 45679h.*
3. *CK# is not shown but is the complement of the CK signal.*
4. *RWDS is not shown and is not used during Write transactions.*

Figure 4.6 Status Read Transaction Example



Note:

1. *CK# is not shown but is the complement of the CK signal.*

Software Interface

5. Address Space Maps

Although the HyperBus protocol provides for slave devices that have both memory and register address spaces, HyperFlash memories described in this specification do not differentiate between memory and registers as separate address spaces. There is a single address space selected by any transaction, independent of whether the HyperBus transaction indicates the target location is in memory space or register space of the selected device.

Write transactions always place the transaction address and data into a a command register set (buffer).

Read transactions return data from the memory array or from a register address space window that has been temporarily overlaid within the single address space by the execution of commands. The single address range with register space overlays methodology is backward compatible with legacy parallel NOR Flash memory program and erase software drivers.

There are several separate address spaces that may appear within the address range of the flash memory device. One address space is visible (entered) at any given time.

- Flash Memory Array: the main non-volatile memory array used for storage of data that may be randomly accessed by read operations.
- ID/CFI: a flash memory array used for Cypress factory programmed device characteristics information. This area contains the Device Identification (ID) and Common Flash Interface (CFI) information tables.
- Secure Silicon Region (SSR): a 1024-byte one-time programmable non-volatile memory array used for Cypress factory programmed permanent data, and customer programmable permanent data.
- Persistent Protection Bits (PPB): a non-volatile memory array with one bit for each Sector. When programmed, each bit protects the related Sector from erasure and programming.
- PPB Lock Bit: a volatile register bit used to enable or disable programming and erase of the PPB bits.
- Password: an OTP non-volatile array used to store a 64-bit password used to enable changing the state of the PPB Lock Bit when using Password Mode Sector Protection.
- Dynamic Protection Bits (DYB): a volatile array with one bit for each Sector. When set, each bit protects the related Sector from erasure and programming.
- Status or Peripheral Registers: register access used to display Embedded Algorithm status and read or write other registers.

The Flash Memory Array is the primary and default address space but, it may be overlaid by one other address space, at any one time. Each alternate address space is called an Address Space Overlay (ASO).

Each ASO replaces (overlays) either the sector selected by the command that enters the ASO or the entire flash device address range, depending on the ASO entry command. If only one sector is overlaid by an ASO the remaining sectors of the memory array remain readable. Any address range not defined by a particular ASO address map, is reserved for future use. Unless otherwise stated all read accesses outside of an ASO address map returns non-valid (undefined) data. The locations will display actively driven data but the meaning of whatever 1s or 0s appear are not defined.

There are multiple address map modes that determine what appears in the flash device address space at any given time:

- Read Mode
- Status Register (SR) Mode
- Address Space Overlay (ASO) Mode
- Peripheral Register Mode

In Read Mode the entire Flash Memory Array may be directly read by the host system memory controller. The memory device Embedded Algorithm Controller (EAC), puts the device in Read Mode during Power-On, after a Hardware Reset, after a Command Reset, or after an Embedded Algorithm (EA) is suspended. Read accesses and commands are accepted in Read Mode. A subset of commands is accepted in Read Mode when an EA is suspended.

While in any mode, the Status Register read command may be issued to cause the Status Register ASO to appear at every word address in the device address space. In this Status Register ASO Mode, the device interface waits for a read access and, any write access is ignored. The next read access to the device accesses the content of the Status Register, exits the Status Register ASO, and returns to the previous (calling) mode in which the Status Register read command was received.

Similarly, commands that read and write other registers use Peripheral Register Mode, in which the register appears in a temporary ASO that is automatically exited after the read or write of the command selected register. The read or write occurs in the last cycle of the register access command sequence.

In EA Mode the EAC is performing an Embedded Algorithm, such as programming or erasing a non-volatile memory array. While in EA Mode, none of the Flash Memory Array is readable. While in EA Mode, only the Program / Erase Suspend command or the Status Register Read command will be accepted. All other commands are ignored. Thus, no other ASO may be entered from the EA Mode.

In ASO Mode, one of the remaining overlay address spaces is entered (overlaid on the Flash Memory Array address map). Only one ASO may be entered at any one time. Commands to the device affect the currently entered ASO. Only certain commands are valid for each ASO. These are listed in each ASO related section of Table , *Command Definitions* on page 51.

The following ASOs have non-volatile data that may be programmed to change 1s to 0s:

- Secure Silicon Region
- ASP Configuration Register (ASPR)
- Persistent Protection Bits (PPB)
- Password
- Only the PPB ASO has non-volatile data that may be erased to change 0s to 1s.

When a program or erase command is issued while one of the non-volatile ASOs is entered, the EA operates on the ASO. The ASO is not readable while the EA is active. When the EA is completed the ASO remains entered and is again readable. Suspend and Resume commands are ignored during an EA operating on any of these ASOs.

The Peripheral Register Mode is used to manage the Power-On Reset Timer, Interrupt Configuration Register, Interrupt Status Register, Volatile Configuration Register, and the Non-Volatile Configuration Register.

5.1 Flash Memory Array

The S26KL/S26KS family has a uniform sector architecture with a sector size of 256 kB. Table to Table show the sector architecture of the three devices.

A user configuration option is available to overlay either the first sector (SA00) or last sector (SAmax) with eight 4-kB Parameter-Sectors. The Parameter-Sector address map showing how the lowest or highest sector is partitioned is shown in Table and Table . The Parameter-Sectors can be erased and programmed in the normal manner using the standard erase and program command sequences targeting the appropriate Parameter-Sector addresses. Note that the smaller Parameter-Sectors need to include A[16:11] as part of the address identifying the target Parameter-Sector during erase and program command sequences. Configuring the first or last uniform sector to include the parameter sectors is accomplished with the Non-Volatile Configuration Register.

Note: The following tables have been condensed to show sector related information for an entire device on a single page. Sectors and their address ranges that are not explicitly listed (such as SA001 – SA510) have sector starting and ending addresses that form the same pattern as all other sectors of that size. For example, all 256-kB sectors have the pattern XX00000h – XX1FFFFh.

S26KL512S and S26KS512S Sector and Memory Address Map

Sector Size (kbyte)	Sector Count	Sector Range	Address Range (16-bit)	Notes
256	256	SA00	0000000h – 001FFFFh	Sector Starting Address
		:	:	–
		SA255	1FE0000h – 1FFFFFFh	Sector Ending Address

S26KL256S and S26KS256S Sector and Memory Address Map

Sector Size (kbyte)	Sector Count	Sector Range	Address Range (16-bit)	Notes
256	128	SA00	0000000h – 001FFFFh	Sector Starting Address
		:	:	–
		SA127	0FE0000h – 0FFFFFFh	Sector Ending Address

S26KL128S and S26KS128S Sector and Memory Address Map

Sector Size (kbyte)	Sector Count	Sector Range	Address Range (16-bit)	Notes
256	64	SA00	0000000h – 001FFFFh	Sector Starting Address
		:	:	–
		SA63	07E0000h – 07FFFFFh	Sector Ending Address

Main Array Sector 0 Overlaid with Eight 4-kB Parameter-Sectors

Main Array Sector Size	Parameter-Sector Number	Address Size	**Address Range** (16-bit)	Notes
256 kB	0	4 kB	0000000h – 00007FFh	Start of Parameter-Sector 0
	1	4 kB	0000800h – 0000FFFh	Parameter-Sector 1
	2	4 kB	0001000h – 00017FFh	Parameter-Sector 2
	3	4 kB	0001800h – 0001FFFh	Parameter-Sector 3
	4	4 kB	0002000h – 00027FFh	Parameter-Sector 4
	5	4 kB	0002800h – 0002FFFh	Parameter-Sector 5
	6	4 kB	0003000h – 00037FFh	Parameter-Sector 6
	7	4 kB	0003800h – 0003FFFh	End of Parameter-Sector 7
	Exposed Portion of Main Array Sector 0	224 kB	0004000h – 001FFFFh	Mapped to exposed portion of Main Array Sector 0

Last Sector Overlaid with Eight 4-kB Parameter-Sectors

Main Array Sector Size	Parameter-Sector Number	Address Size	**Address Range** (16-bit)	Notes
256 kB	Exposed portion of last sector in main array	224 kB	1FF0000h – 1FFBFFFh	Mapped to exposed portion of Main Array Sector (last)
	0	4 kB	1FFC000h – 1FFC7FFh	Start of Parameter-Sector 0
	1	4 kB	1FFC800h – 1FFCFFFh	Parameter-Sector 1
	2	4 kB	1FFD000h – 1FFD7FFh	Parameter-Sector 2
	3	4 kB	1FFD800h – 1FFDFFFh	Parameter-Sector 3
	4	4 kB	1FFE000h – 1FFE7FFh	Parameter-Sector 4
	5	4 kB	1FFE800h – 1FFEFFFh	Parameter-Sector 5
	6	4 kB	1FFF000h – 1FFF7FFh	Parameter-Sector 6
	7	4 kB	1FFF800h – 1FFFFFFh	End of Parameter-Sector 7

Notes:

1. *When mapping the Parameter-Sectors to the highest sector the upper order address bits depend on device density. Table uses the 512-Mbit device address mapping.*
2. *The sector number of the last sector is 255 for 512-Mb density, 127 for 256-Mb density, and 63 for 128-Mb density*
3. *When performing erase and program operations on a Parameter-Sector, the sector address that must be specified in the command sequence is A[max:11].*

5.2 Device ID and CFI (ID-CFI) ASO

There are two traditional methods for systems to identify the type of flash memory installed in the system. One has traditionally been called Autoselect and is now referred to as Device Identification (ID). The other method is called Common Flash Interface (CFI).

For ID, a command is used to enable an address space overlay where up to 16 word locations can be read to get JEDEC manufacturer identification (ID), device ID, and some configuration and protection status information from the flash memory. The system can use the manufacturer and device IDs to select the appropriate driver software to use with the flash device.

CFI also uses a command to enable an Address Space Overlay where an extendable table of standard information about how the flash memory is organized and operates can be read. With this method the driver software does not have to be written with the specifics of each possible memory device in mind. Instead the driver software is written in a more general way to handle many different devices but adjusts the driver behavior based on the information in the CFI table.

Traditionally these two address spaces have used separate commands and were separate overlays. However, the mapping of these two address spaces are non-overlapping and so can be combined in to a single address space and appear together in a single overlay. Either of the traditional commands used to access (enter) the Autoselect (ID) or CFI overlay will cause the now combined ID-CFI address map to appear.

The ID-CFI address map appears within, and overlays the Flash Memory Array data of the sector selected by the address used in the ID-CFI enter command. While the ID-CFI ASO is entered the content of all other sectors is undefined.

The ID-CFI address map starts at location zero of the selected sector. Locations above the maximum defined address of the ID-CFI ASO to the maximum address of the selected sector have undefined data. The ID-CFI enter commands use the same address and data values used on previous generation memories to access the JEDEC Manufacturer ID (Autoselect) and Common Flash Interface (CFI) information, respectively.

ID-CFI Address Map Overview

Word Address	Description	Read / Write
(SA) + 0000h to 000Fh	Device ID (traditional Autoselect values)	Read Only
(SA) + 0010h to 0079h	CFI data structure	Read Only
(SA) + 007Ah to 00FFh	Undefined	Read Only

For the complete address map see Table , *ID (Autoselect) Address Map* on page 57.

5.2.1 Device ID

The Joint Electron Device Engineering Council (JEDEC) standard JEP106T defines the manufacturer ID for a compliant memory. Common industry usage defined a method and format for reading the manufacturer ID and a device specific ID from a memory device. The manufacturer and device ID information is primarily intended for programming equipment to automatically match a device with the corresponding programming algorithm. Cypress has added additional fields within this 32-byte address space.

The original industry format was structured to work with any memory data bus width e.g. x8, x16, x32. The ID code values are traditionally byte wide but are located at bus width address boundaries such that incrementing the device address inputs will read successive byte, word, or double word locations with the ID codes always located in the least significant byte location of the data bus. Because the device data bus is word wide, each code byte is located in the lower half of each word location. The original industry format made the high order byte always zero. Cypress has modified the format to use both bytes in some words of the address space. For the detail description of the Device ID address map see Table , *ID (Autoselect) Address Map* on page 57.

5.2.2 Common Flash Memory Interface

The JEDEC Common Flash Interface (CFI) specification (JESD68.01) defines a standardized data structure that may be read from a flash memory device, which allows vendor-specified software algorithms to be used for entire families of devices. The data structure contains information for system configuration such as various electrical and timing parameters, and special functions supported by the device. Software support can then be device-independent, Device ID-independent, and forward-and-backward-compatible for entire flash device families.

The system can read CFI information at the addresses within the selected sector as shown in 9.3, *Device ID and Common Flash Interface (ID-CFI) ASO Map* on page 57.

Like the Device ID information, CFI information is structured to work with any memory data bus width e.g. x8, x16, x32. The code values are always byte wide but are located at data bus width address boundaries such that incrementing the device address reads successive byte, word, or double word locations with the codes always located in the least significant byte location of the data bus. Because the data bus is word wide, each code byte is located in the lower half of each word location and the high order byte is always zero.

For further information, refer to the *CFI Specification, Version 1.4* (or later), and the *JEDEC publications JEP137-A and JESD68.01*. Refer to JEDEC (http://www.jedec.org) for the standards and the CFI Specification at http://www.spansion.com/Support/TechnicalDocuments/Pages/ApplicationNotes.aspx or contact the local sales office.

6. Embedded Operations

6.1 Embedded Algorithm Controller (EAC)

The EAC takes commands from the host system for programming and erasing the flash memory arrays and performs all the complex operations needed to change the non-volatile memory state. This frees the host system from any need to manage the program and erase processes.

There are five EAC operation categories:

- Deep Power-Down
- Standby (Read Mode)
- Address Space Switching
- Embedded Algorithms (EA)
- Advanced Sector Protection (ASP) Management

6.1.1 Deep Power-Down

In the Deep Power-Down (DPD) Mode current consumption is driven to the lowest level. The DPD Mode must be entered while the device is in the Idle state while not in an ASO. DPD can also be entered when the device is in either the erase suspend or program suspend states.

6.1.2 EAC Standby

In the Standby Mode current consumption is greatly reduced. The EAC enters its Standby Mode when no command is being processed and no Embedded Algorithm is in progress. If the device is deselected
(CS# = High) during an Embedded Algorithm, the device still draws active current until the operation is completed (I_{CC3}). I_{CC4} in 11.2, *DC Characteristics (CMOS Compatible)* on page 63 represents the standby current specification when both the Host Interface and EAC are in their Standby state.

6.1.3 Address Space Switching

Writing specific address and data sequences (command sequences) switch the memory device address space from the Flash Memory Array to one of the Address Space Overlays (ASO).

Embedded Algorithms operate on the information visible in the currently active (entered) ASO. The system continues to have access to the ASO until the system issues an ASO Exit command, performs a Hardware Reset, or until power is removed from the device. An ASO Exit Command switches from an ASO back to the Flash Memory Array address space. The commands accepted when a particular ASO is entered are listed between the ASO Enter and Exit commands in the command definitions table. See 7.1, *Command Summary* on page 51 for address and data requirements for all command sequences.

6.1.4 Embedded Algorithms (EA)

Changing the non-volatile data in the memory array requires a complex sequence of operations that are called Embedded Algorithms (EA). The algorithms are managed entirely by the device’s internal Embedded Algorithm Controller (EAC). The main algorithms perform programming and erase of the Main Array Data and the ASOs. The host system writes command codes to the flash device address space. The EAC receives the commands, performs all the necessary steps to complete the command, and provides status information during the progress of an EA.

6.2 Program and Erase Summary

Flash data bits are erased in parallel in a large group called a sector. The erase operation places each data bit in the sector in the logical 1 state (High). Flash data bits may be individually programmed from the erased 1 state to the programmed logical 0 (low) state. A data bit of 0 cannot be programmed back to a 1. A succeeding read shows that the data is still 0. Only erase operations can convert a 0 to a 1. Programming the same word location more than once with different 0 bits will result in the logical AND of the previous data and the new data being programmed.

The duration of program and erase operations is shown in Section 8., *Embedded Algorithm Performance* on page 55.

Program and erase operations may be suspended.

- An erase operation may be suspended to allow either programming or reading of another sector (not in the erase sector) in the erase operation. No other erase operation can be started during an erase suspend.
- A program operation may be suspended to allow reading of another location (not in the Line being programmed).
- No other program or erase operation may be started during a suspended program operation; program or erase commands will be ignored during a suspended program operation.
- After an intervening program operation or read access is complete the suspended erase or program operation may be resumed.
- Program and Erase operations may be interrupted as often as necessary but in order for a program or erase operation to progress to completion there must be some periods of time between resume and the next suspend commands greater than or equal to t_{PRS} or t_{ERS} in *Embedded Algorithm Performance* on page 55.
- When an Embedded Algorithm (EA) is complete, the EAC returns to the operation state and address space from which the EA was started (Erase Suspend or EAC Standby).

The system can determine the status of a program or erase operation by reading the Status Register (*Status Register* on page 36).

Any commands written to the device during the Embedded Program Algorithm are ignored except the Program Suspend, and Status Read command.

Any commands written to the device during the Embedded Erase Algorithm are ignored except Erase Suspend and Status Read command.

A Hardware Reset immediately terminates any in progress program / erase operation and returns to Read Mode after t_{RPH} time. The terminated operation should be reinitiated once the device has returned to the Idle state, to ensure data integrity.

For performance and reliability reasons programming is internally done on 16-byte half-pages, using an aligned 16-byte address range.

I_{CC3} in *DC Characteristics (CMOS Compatible)* on page 63 represents the active current specification for a write (Embedded Algorithm) operation.

6.2.1 Program Granularity

The S26KL/S26KS supports two methods of programming, Word or Write Buffer Programming.

Word programming examines the data word supplied by the command and programs 0’s in the addressed memory array word to match the 0’s in the command data word.

Write Buffer Programming examines the write buffer and programs 0’s in the addressed memory array Line to match the 0’s in the write buffer. The write buffer does not need to be completely filled with data. It is allowed to program as little as a single bit, several bits, a single word, a few words, a half-page, multiple half-pages, or the entire buffer as one programming operation. Use of the write buffer method reduces host system overhead in writing program commands and reduces memory device internal overhead in programming operations to make Write Buffer Programming more efficient and thus faster than programming individual words with the Word Programming command.

Each half-page can be programmed by either method. Half-pages programmed by different methods may be mixed within a Line.

Word Programming and Write Buffer Programming, more than once within a half-page, is supported for legacy software compatibility. However, for the best data integrity, it is recommended to not use Word Programming more than once within a half-page and, to not use Write Buffer Programming to program within a half-page, more than once. For applications requiring multiple programming operations within the same half-page, it is recommended to add system software Error Detection and Correction, to enhance the data integrity of half-pages that are programmed more than once.

Future silicon process generations of HyperFlash may no longer support multiple program operations, within the same half-page, without an erase operation on the sector containing the half-page. Planning for software migration to future generations: should adopt data structures and data management methods that can support only one programming operation, per half-page, per erase.

6.2.2 Incremental Programming

The same word location or half-page may be programmed more than once, by either the Word or Write Buffer Programming methods, to incrementally change 1’s to 0’s. However as noted in Section 6.2.1, *Program Granularity* on page 21 incremental programming affects data integrity. It is recommended to add system software Error Detection and Correction, to enhance the data integrity of half-pages that are programmed more than once.

6.2.3 Program Methods

6.2.3.1 Word Programming

Word programming is used to program a single word or a group of words anywhere in the flash memory arrays.

The minimum Word Programming command sequence requires four command write transactions. The program command sequence is initiated by issuing two unlock command write transactions (transactions one and two), followed by the program set-up command (transaction three). The program address and data are written next (transaction four), which in turn initiates the Embedded Programming algorithm. The system is not required to provide further controls or timing. The device automatically generates the program pulses and verifies the programmed cell margin internally. When the Embedded Programming algorithm is complete, the EAC then returns to its Standby Mode.

The four transaction Word Programming command sequence described earlier is used to program a single (16-bit) word. Multiple sequential words can be programmed with the Word Programming sequence by using the burst write capability. The unlock and program command sequence is identical to a single Word Programming sequence but during the data / address transaction multiple sequential data values are loaded during a single assertion of CS#. The data presented is programmed into sequential addresses starting with the target address identified in the Command-Address phase of the burst write transaction. A maximum of 256 words can be programmed as long as an aligned 256 word address boundary is not crossed. When using a burst write to load multiple words during a Word Program sequence the clock rate must be lowered as described in Figure 12.1, *Burst Write During Load of Multiple Words During a Word Program Command Timing Diagram* on page 69.

The system can determine the status of the program operation by reading the Status Register (*Status Register* on page 36).

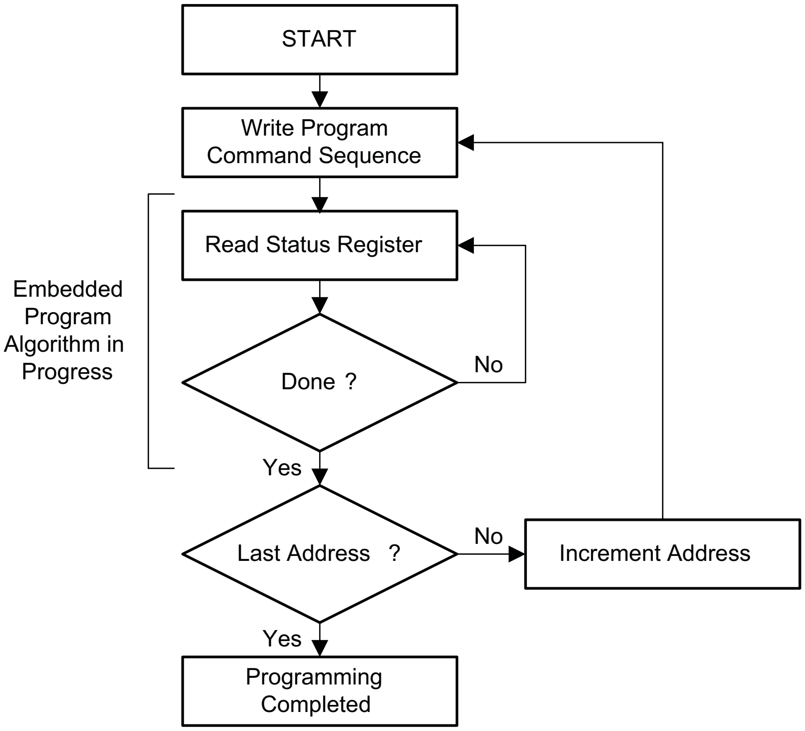
Any commands other than Program Suspend and Status Register Read written to the device during the Embedded Program algorithm are ignored.

Note that a Hardware Reset (RESET# = V_{IL}) or power loss immediately terminates the programming operation and returns the device to Read Mode after t_{RPH} time. The termination may leave the area being programmed in an intermediate state with invalid or unstable data values. Once the device has completed the Hardware Reset operation, the program command sequence may be reinitiated with the same data to complete the programming operation, to ensure the data is fully programmed. However, to ensure the best data integrity, the sector in which the program operation was terminated must be erased and re-programed.

The Word Programming command may also be used when the SSR ASO is entered.

A modified version of the Word Programming command, without unlock write cycles, is used for programming when entered into the ASP Configuration Register (ASPR), Password, and PPB ASOs. The same command is used to change volatile bits when entered in to the PPB Lock, and DYB ASOs. See Table on page 51 for Program Command sequences.

Figure 6.1 Word Program Operation



6.2.3.2 Write Buffer Programming

A write buffer is used to program data within a 512-byte address range aligned on a 512-byte boundary (Line). Thus, a full Write Buffer Programming operation must be aligned on a Line boundary. Programming operations of less than a full 512 bytes may start on any word boundary but may not cross a Line boundary. At the start of a Write Buffer Programming operation all bit locations in the buffer are all 1's (FFFFh words) thus any locations not loaded will retain the existing data. See *Product Overview* on page 4 for information on address map.

Write Buffer Programming allows up to 512 bytes to be programmed in one operation. It is possible to program from 1 bit up to 512 bytes in each Write Buffer Programming operation. It is strongly recommended that a multiple of 16-byte half-pages be written and each half-page written only once. For the very best performance, programming should be done in full Lines of 512 bytes aligned on 512-byte boundaries.

Write Buffer Programming is supported only in the Flash Memory Array or the SSR ASO.

The Write Buffer Programming operation is initiated by first writing two unlock cycles. This is followed by a third write cycle of the Write to Buffer command with the Sector Address (SA), in which programming is to occur. Next, the system writes the number of word locations minus one. This tells the device how many write buffer addresses will be loaded with data and therefore when to expect the Program Buffer to flash confirm command. The Sector Address provided in both the Write to Buffer command and the Write Word Count command must match. The Sector to be programmed must be unlocked (unprotected). If a programming operation is attempted to a locked sector, the operation will be aborted and the failure will be indicated in the Status Register (see Table , *Status Register* on page 37).

The system then writes the starting address and data word. This starting address is the first address and data pair to be programmed, and selects the starting word address within the write buffer Line. The Sector address must match the Write to Buffer command Sector Address or the operation will abort and return to the initiating state. All subsequent single word address and data pair write transactions must be in sequential order. All write buffer addresses must be within the same Line. If the system attempts to load data outside this range, the operation will abort and return to the initiating state.

The word counter decrements for each data word loaded. Note that while counting down the data writes, every write is considered to be data being loaded into the write buffer. No commands are possible during the write buffer loading period. The only way to stop loading the write buffer is to write with an address that is outside the Line of the programming operation. This invalid address will immediately abort the Write to Buffer command sequence and set the Write Buffer Abort Status Bit (SR[3]).

Once the specified number of write buffer locations has been loaded, the system must then write the Program Buffer to Flash command at the Sector Address. The device then goes busy. The Embedded Program algorithm automatically programs and verifies the data for the correct data pattern. The system is not required to provide any controls or timings during these operations. If an incorrect number of write buffer locations have been loaded the operation will abort and return to the initiating state. The abort occurs when anything other than the Program Buffer to Flash is written when that command is expected at the end of the word count.

The Write-Buffer Embedded Programming operation can be suspended using the Program Suspend command. When the Embedded Program algorithm is complete, the EAC then returns to the EAC Standby or Erase Suspend Standby state where the programming operation was started.

The system can determine the status of the program operation by using the Status Register (Table , *Status Register* on page 37). See Figure 6.2, *Write Buffer Programming Operation with Status Register* on page 24 for a diagram of the programming operation.

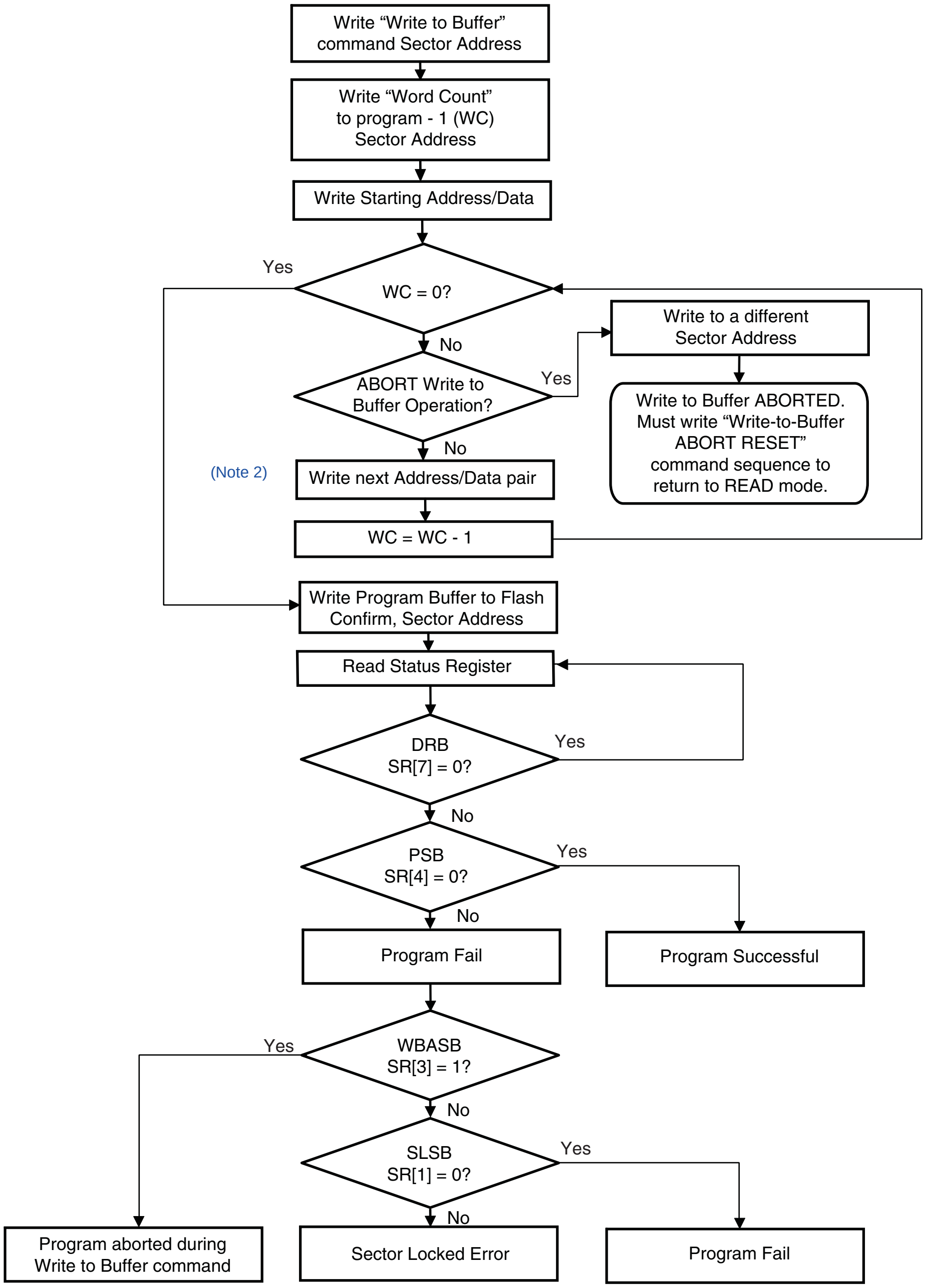
The Write Buffer Programming Sequence will be Aborted under the following conditions:

- Load a Word Count value greater than the buffer size (255).
- Write an address that is outside the Line provided in the Write to Buffer command.
- The Program Buffer to Flash command is not issued after the Write Word Count number of data words is loaded.

When any of the conditions that cause an abort of write buffer command occur the abort will happen immediately after the offending condition, and will indicate a Program Fail in the Status Register at bit location 4 (PSB = 1) due to Write Buffer Abort bit location 3 (WBASB = 1). The next successful program operation will clear the failure status or a Clear Status Register may be issued to clear the PSB status bit.

The Write Buffer Programming sequence can be terminated by the following: Hardware Reset or Power cycle. However, using either of these methods may leave the area being programmed in an intermediate state with invalid or unstable data values. In this case the same area will need to be reprogrammed with the same data or erased to ensure data values are properly programmed or erased. To ensure the best data integrity, the sector in which the program operation was terminated must be erased and re-programed.

Figure 6.2 Write Buffer Programming Operation with Status Register


Write "Write to Buffer" command Sector Address
Write "Word Count" to program - 1 (WC) Sector Address
Write Starting Address/Data
WC = 0?
Yes
No
Write to a different Sector Address
ABORT Write to Buffer Operation?
Yes
Write to Buffer ABORTED. Must write "Write-to-Buffer ABORT RESET" command sequence to return to READ mode.
No
(Note 2)
Write next Address/Data pair
WC = WC - 1
Write Program Buffer to Flash Confirm, Sector Address
Read Status Register
DRB SR[7] = 0?
Yes
No
PSB SR[4] = 0?
Yes
No
Program Fail
Program Successful
WBASB SR[3] = 1?
Yes
No
SLSB SR[1] = 0?
Yes
No
Program aborted during Write to Buffer command
Sector Locked Error
Program Fail

Notes:

1. *See* Table , *Command Definitions on page 51* for the command sequence as required for Write Buffer Programming.
2. *When the Sector Address is specified, any address in the selected sector is acceptable. However, when loading Write-Buffer address locations with data, all addresses MUST fall within the selected Line.*

Write Buffer Programming Command Sequence

Sequence	Address	Data	Comment
Issue Unlock Command 1	555	AA	
Issue Unlock Command 2	2AA	55	
Issue Write to Buffer Command at Sector Address	SA	0025h	
Issue Number of Locations at Sector Address	SA	WC	WC = number of words to program minus 1.
Example: WC of 0 = 1 word to program WC of 1 = 2 words to program			
Load Starting Address / Data pair	Starting Address	PD	Selects a Line and loads first Address / Data Pair.
Load next Address / Data pair	WBL	PD	All addresses **must** be within the selected Line boundaries, and have to be loaded in sequential order.
Load LAST Address / Data pair	WBL	PD	All addresses **must** be within the selected Line boundaries, and have to be loaded in sequential order.
Issue Write Buffer Program Confirm at Sector Address	SA	0029h	This command **must** follow the last write buffer location loaded, or the operation will ABORT.
Device goes busy			

Legend:
SA = Sector Address (Non-Sector Address bits are ‘don't care’. Any address within the Sector is sufficient.)
WBL = Write Buffer Location (MUST be within the boundaries of the Line specified by the Starting Address.)
WC = Word Count
PD = Program Data

6.2.4 Program Suspend / Program Resume Commands

The Program Suspend command allows the system to interrupt an embedded programming operation so that data can be read from any non-suspended Line. When the Program Suspend command is written during a programming process, the device halts the programming operation within t_{PSL} (program suspend latency) and updates the status bits. Addresses are ‘don't care’ when writing the Program Suspend command.

After the programming operation has been suspended, the system can read array data from any non-suspended Line. The Program Suspend command may also be issued during a programming operation while an erase is suspended. In this case, data may be read from any addresses not in Erase Suspend or Program Suspend.

After the Program Resume command is written, the device reverts to programming and the status bits are updated. The system can determine the status of the program operation by reading the Status Register. Refer to *Status Register* on page 36 for information on these status bits.

Accesses and commands that are valid during Program Suspend are:

- Read to any other non-erase-suspended sector
- Read to any other non-program-suspended Line
- Status Read command
- Exit ASO or Command Set Exit
- Program Resume command
- Load Interrupt Configuration Register
- Load Interrupt Status Register

The system must write the Program Resume command to exit the Program Suspend Mode and continue the programming operation. Further writes of the Program Resume command are ignored. Another Program Suspend command can be written after the device has resumed programming.

Program operations can be interrupted as often as necessary but in order for a program operation to progress to completion there must be some periods of time between resume and the next suspend command greater than or equal to t_{PRS} in *Embedded Algorithm Controller (EAC)* on page 19.

Program suspend and resume is not supported while entered in an ASO. While in program suspend entry into ASO is not supported.

6.2.5 Blank Check

The Blank Check command will confirm if the selected Flash Memory Array sector is fully erased. The Blank Check command does not allow for reads to the array during the Blank Check. Reads to the array while this command is executing will return unknown data.

To initiate a Blank Check on a sector, write 33h to address 555h in the sector, while the EAC is in the Standby state.

The Blank Check command may not be written while the device is actively programming or erasing or suspended.

Use the Status Register read to confirm if the device is still busy and when complete if the sector is blank or not. Bit 7 of the Status Register will show if the device is performing a Blank Check (similar to an erase operation). Bit 5 of the Status Register will be cleared to 0 if the sector is erased and set to 1 if not erased.

As soon as any bit is found to not be erased, the device will halt the operation and report the results.

Once the Blank Check is completed, the EAC will return to the Standby state.

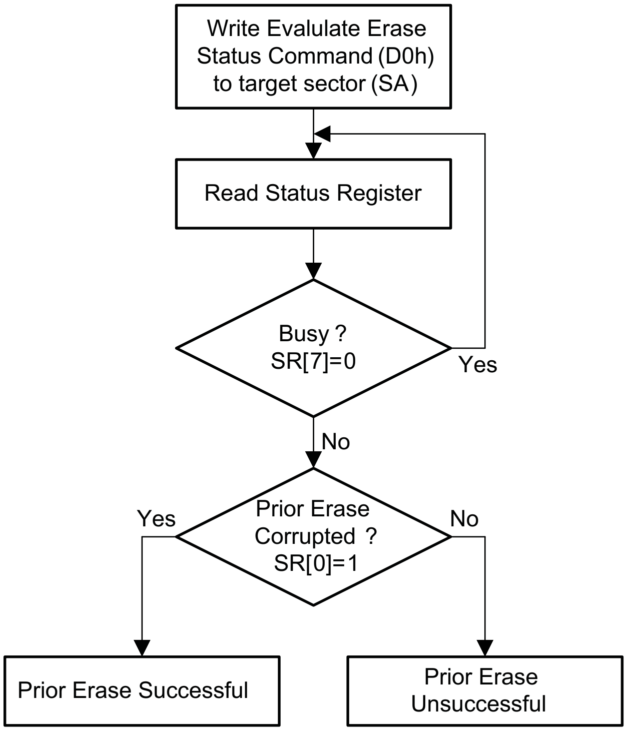
6.2.6 Evaluate Erase Status

The Evaluate Erase Status (EES) command verifies that the last erase operation on the addressed sector was completed successfully. If the selected sector was successfully erased the Sector Erase Status Bit in the Status Register (SR[0]) is set to 1. If the selected sector was not completely erased SR[0] is cleared to 0. See Figure 6.3, *Evaluate Erase Status Software Sequence* on page 26 for details.

The EES command can be used to detect erase operations that failed due to loss of power, reset, or failure during an erase operation.

The EES command requires t_{EES} to complete and update the Sector Erase Status Bit in the Status Register (SR[0]). The Device Ready Bit in the Status Register (SR[7]) may be read using the Read Status Register (70h) command to determine when the EES command has finished. Once the Device Ready Bit in the Status Register indicates that the device has returned to the Ready (1) state the Sector Erase Status Bit (SR[0]) indicates whether the target sector was successfully erased. If a sector is found not erased with SR[0] = 0, the sector must be erased again to ensure reliable storage of data in the sector.

Figure 6.3 Evaluate Erase Status Software Sequence


Write Evalulate Erase Status Command (D0h) to target sector (SA)
Read Status Register
Busy ? SR[7]=0
Yes
No
Yes
Prior Erase Corrupted ? SR[0]=1
No
Prior Erase Successful
Prior Erase Unsuccessful

6.2.7 Erase Methods

6.2.7.1 Chip Erase

The Chip Erase function erases the entire Flash Memory Array. The device does not require the system to preprogram prior to erase. The Embedded Erase Algorithm automatically programs and verifies the entire memory for an all 0 data pattern prior to electrical erase. After a successful Chip Erase, all locations within the device contain FFFFh. The system is not required to provide any controls or timings during these operations. The Chip Erase command sequence is initiated by writing two unlock cycles, followed by a set-up command. Two additional unlock write cycles are then followed by the Chip Erase command, which in turn invokes the Embedded Erase Algorithm.

When the Embedded Erase Algorithm is complete, the EAC returns to the Standby state. Note that while the Embedded Erase operation is in progress, the system cannot read valid data from the array. The system can determine the status of the erase operation by reading the Status Register. Refer to *Status Register* on page 36 for information on these status bits. Once the Chip Erase operation has begun, only a Status Read, Hardware Reset, or Power cycle are valid. All other commands are ignored. However, a Hardware Reset or Power Cycle immediately terminates the erase operation and returns to Read Mode after t_{RPH} time. If a chip erase operation is terminated, the Chip Erase command sequence must be reinitiated once the device has returned to the Idle state to ensure data integrity.

Sectors protected by the ASP DYB and PPB bits will not be erased. See *Software Interface Reference* on page 51. If a sector is protected during Chip Erase, Chip Erase will skip the protected sector and continue with next sector erase. The Status Register Erase Status Bit and Sector Lock Bit are not set to 1 by a failed erase on a protected sector.

6.2.7.2 Sector Erase

The Sector Erase function erases one sector in the memory array. The device does not require the system to preprogram prior to erase. The Embedded Erase Algorithm automatically programs and verifies the entire sector for an all 0 data pattern prior to electrical erase. After a successful sector erase, all locations within the erased sector contain FFFFh. The system is not required to provide any controls or timings during these operations. The Sector Erase command sequence is initiated by writing two unlock cycles, followed by a set-up command. Two additional unlock write cycles are then followed by the address of the sector to be erased, and the Sector Erase command.

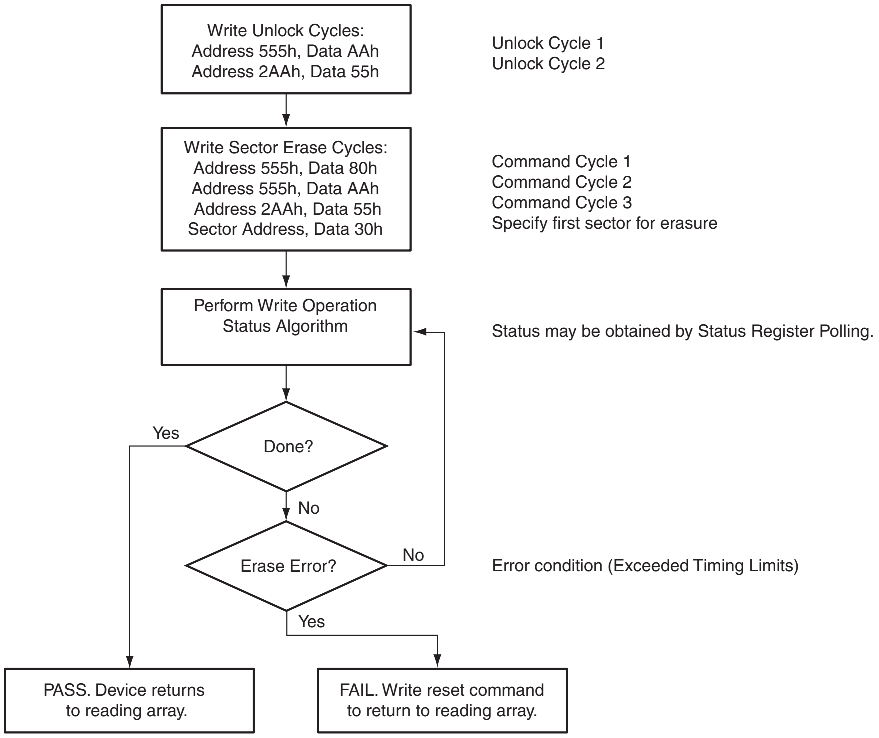
The system can determine the status of the erase operation by reading the Status Register. Refer to *Status Register* on page 36 for information on these status bits.

Once the sector erase operation has begun, the Status Register Read and Erase Suspend commands are valid. All other commands are ignored by the Embedded Algorithm Controller. However, note that a Hardware Reset immediately terminates the erase operation and returns to Read Mode after t_{RPH} time. If a sector erase operation is terminated, the Sector Erase command sequence must be reinitiated once the device has reset operation to ensure data integrity.

See *Embedded Algorithm Controller (EAC)* on page 19 for parameters and timing diagrams.

Sectors protected by the ASP DYB and PPB bits will not be erased. See *Software Interface Reference* on page 51. If an erase operation is attempted to a locked sector the operation will be aborted and the failure will be indicated in the Status Register (see Table , *Status Register* on page 37).

Figure 6.4 Sector Erase Operation



6.2.8 Erase Suspend / Erase Resume

The Erase Suspend command allows the system to interrupt a sector erase operation and then read data from, or program data to, the Flash Memory Array. This command is valid only during sector erase or program operation. The Erase Suspend command is ignored if written during the chip erase operation.

When the Erase Suspend command is written during the sector erase operation, the device requires a maximum of t_{ESL} (erase suspend latency) to suspend the erase operation and update the status bits.

After the erase operation has been suspended, the part enters the Erase-Suspend Mode. The system can read data from or program data to the Flash Memory Array. Reading at any address within erase-suspended sectors produces undetermined data. The system can determine if a sector is actively erasing or is erase-suspended by reading the Status Register. Refer to *Status Register* on page 36 for information on these status bits.

After an erase-suspended program operation is complete, the EAC returns to the Erase-Suspend state. The system can determine the status of the program operation by reading the Status Register, just as in the standard program operation.

If a program failure occurs during erase suspend the Status Register Clear or Software Reset commands will return the device to the erase suspended state.   Erase will need to be resumed and completed before again trying to program the memory array.

Accesses and commands that are valid during Erase Suspend are:

- Read to any other non-suspended sector
- Program to any other non-suspended sector
- Status Read command
- Exit ASO or Command Set Exit
- Erase Resume command
- SSR Entry
- SSR Read
- SSR Program

To resume the sector erase operation, the system must write the Erase Resume command. The device will revert to erasing and the status bits will be updated. Further writes of the Resume command are ignored. Another Erase Suspend command can be written after the chip has resumed erasing.

Note that the DYB ASO can not be entered while the device is in the Erase Suspend state.

6.2.9 Non-Volatile Configuration Register and Volatile Configuration Register

The Non-Volatile Configuration Register (NVCR) and the Volatile Configuration Register (VCR) are used to define the operating conditions for the HyperFlash bus. Configurable characteristics include:

1. Wrapped Burst Length (16-byte, 32-byte, or 64-byte wrapped burst).
   a. 16-byte and 32-byte wrapped bursts behave in the legacy manner, 64-byte wrapped burst behave as shown in Table , *64-byte Wrapped Burst Address Sequence (Latency Code = 16)* on page 32.
2. Read Latency (5 to 16 clocks to allow for initial read latency).
3. Output Driver Drive Strength.
4. Whether the 4-kB Parameter-Sectors are used and how they are mapped into the address map.
5. SSR Freeze bit to lock the Secure Silicon Region.
6. xVCR Freeze bit to lock the Volatile Configuration Register and the Non-Volatile Configuration Register.

The contents of the VCR and NVCR can be loaded and read back as described in Table , *Command Definitions* on page 51. The HyperFlash device uses the contents of the NVCR to define bus characteristics upon Power-Up or after a Hardware Reset. If the host system loads the VCR, the bus characteristics will be defined by the contents of the VCR (Figure 6.5). The NVCR is intended to hold a default setting to allow alignment with the host controller settings during boot operation. The VCR will often be updated with optimized settings during the boot process. The source for bus characteristics will shift from the NVCR (after Power-Up or Hard

Reset) to the VCR once the VCR is loaded. Once the VCR is loaded only a Power-Up or Hard Reset will return bus characteristics back to the NVCR settings. When unlocked the VCR can be altered at any time while the device is idle.

The number of times the NVCR can be erased and reprogrammed is defined by the n_NVCR spec (see Table , *Program / Erase Endurance* on page 56). To assure consistent bus configuration during and after NVCR programming the VCR should be used to define bus operating characteristics when programming the NVCR.

VCR and NVCR Configuration Register Bit Assignments

xVCR Bit	Function	Settings (Binary)
xVCR.15	Reserved	1 - Reserved (default)
xVCR14 – xVCR12	Drive Strength	See Table , *Drive Strength Code* on page 30
xVCR.11	xVCR Freeze	0 - VCR or NVCR Locked (No Programming or Erasing of NVCR, no changes to VCR) 1 - VCR and NVCR Unlocked (factory default)
xVCR.10	SSR Freeze	0 - Secure Silicon Region Locked (Programming not allowed) 1 - Secure Silicon Region Unlocked (factory default)
xVCR.9 – xVCR.8	Parameter-Sector Mapping	00 - Parameter-Sectors and Read Password Sectors mapped into lowest addresses 01 - Parameter-Sectors and Read Password Sectors mapped into highest addresses 10 - Uniform Sectors with Read Password Sector mapped into lowest addresses. (factory default) 11 - Uniform Sectors with Read Password Sector mapped into highest addresses
xVCR.7 – xVCR.4	Read Latency	0000 - 5 Clock Latency 0001 - 6 Clock Latency 0010 - 7 Clock Latency 0011 - 8 Clock Latency 0100 - 9 Clock Latency ... 1011 - 16 Clock Latency (factory default) See Table , *Maximum Operating Frequency for Latency Code Options* on page 10
xVCR.3	Reserved	1 - Reserved (default)
xVCR.2	Reserved	0 - Reserved (default)
xVCR.1 – xVCR.0	Burst Length	00 - Reserved 01 - 64 bytes 10 - 16 bytes 11 - 32 bytes (factory default)

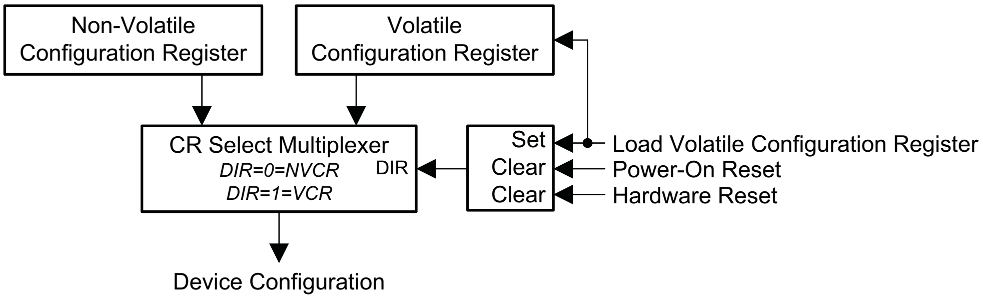
Note:
1. *The placement of the Configuration Register bits are the same in both the Non-Volatile and Volatile Configuration Registers.*

Drive Strength Code

xVCR14 – xVCR12	Typical Impedance 1.8V $V_{CC}Q$	Typical Impedance 3V $V_{CC}Q$	Units
000 (default)	27	20	Ohms
001	117	71	
010	68	40	
011	45	27	
100	34	20	
101	27	16	
110	24	14	
111	20	12	

Note:
1. *Typical Impedance measured at nominal $V_{CC}Q$, 25°C.*

Figure 6.5 Configuration Control



Notes:

1. *A Software Reset will not change the state of the CR Select Multiplexer.*
2. *Programming or erasure of the NVCR does not impact the contents of the VCR that has been previously loaded.*
3. *If the VCR has not been loaded, programming of the NVCR will be result in the VCR being loaded with the new NVCR value.*

VCR and NVCR Freeze Bits Immediately After Power-Up or Hardware Reset

NVCR.11 Bit	VCR.11 Bit	NVCR	VCR
1	1	Programmable / Erasable	Settable / Clearable
1	0	Temporarily Locked	Temporarily Locked
0	1	Programmable / Erasable	Settable / Clearable (1), (3)
0	0	Permanently Locked	Permanently Locked (4)

Notes:

1. *Programming / Erasing the NVCR will not impact behavior until after the next POR or Hardware Reset.*
2. *Loading the VCR will impact behavior immediately.*
3. *This state occurs after NVCR[11] = VCR[11] = 1 and the NVCR[11] bit is programmed. The state will only exist until the next POR or Hardware Reset. Thereafter NVCR[11] = VCR[11] = 0.*
4. *This state occurs after POR or Hardware Reset when NVCR[11] was previously programmed.*

.

Example Burst Sequences

VCR / NVCR [1:0]	CA[45]	Wrap Boundary (Bytes)	Start Address (Hex)	Address Sequence (Hex) (Words)
XX	1	Linear	XXXXXX03	03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, ...
10	0	16	XXXXXX02	02, 03, 04, 05, 06, 07, 00, 01, ...
10	0	16	XXXXXX0C	0C, 0D, 0E, 0F, 08, 09, 0A, 0B, ...
11	0	32	XXXXXX0A	0A, 0B, 0C, 0D, 0E, 0F, 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, ...
11	0	32	XXXXXX1E	1E, 1F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, ...
01	0	64	XXXXXX03	03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F, 00, 01, 02, ...
01	0	64	XXXXXX2E	2E, 2F, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 3A, 3B, 3C, 3D, 3E, 3F, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D, , ...

64-byte Wrapped Burst Address Sequence (Latency Code = 16)

	Clock Cycle																																												
Target Address	**0**	**1**	**2**	**3**	**…**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**	**31**	**32**	**33**	**34**	**35**	**36**	**37**	**38**	**39**	**40**	**41**	**42**	**43**	**44**	**45**	**46**	**47**	**48**	**49**	**50**	**51**	**52**	**53**	**54**	**55**	**56**
0	CA0	CA1	CA2	Bus Turnaround + Initial Latency			0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	–	–	–	–	–	–	–
1							1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	X	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	–	–	–	–	–	–
2							2	3	4	5	6	7	8	9	10	11	12	13	14	15	X	X	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	–	–	–	–	–
3							3	4	5	6	7	8	9	10	11	12	13	14	15	X	X	X	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	–	–	–	–
4							4	5	6	7	8	9	10	11	12	13	14	15	X	X	X	X	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	–	–	–
5							5	6	7	8	9	10	11	12	13	14	15	X	X	X	X	X	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	–	–
6							6	7	8	9	10	11	12	13	14	15	X	X	X	X	X	X	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	–
7							7	8	9	10	11	12	13	14	15	X	X	X	X	X	X	X	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6
8							8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	–	–	–	–	–	–	–
9							9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	X	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	–	–	–	–	–	–
10							10	11	12	13	14	15	16	17	18	19	20	21	22	23	X	X	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	–	–	–	–	–
11							11	12	13	14	15	16	17	18	19	20	21	22	23	X	X	X	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	–	–	–	–
12							12	13	14	15	16	17	18	19	20	21	22	23	X	X	X	X	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	–	–	–
13							13	14	15	16	17	18	19	20	21	22	23	X	X	X	X	X	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	–	–
14							14	15	16	17	18	19	20	21	22	23	X	X	X	X	X	X	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	–
15							15	16	17	18	19	20	21	22	23	X	X	X	X	X	X	X	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
23							16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	–	–	–	–	–	–	–
17							17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	–	–	–	–	–	–
18							18	19	20	21	22	23	24	25	26	27	28	29	30	31	X	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	–	–	–	–	–
19							19	20	21	22	23	24	25	26	27	28	29	30	31	X	X	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	–	–	–	–
20							20	21	22	23	24	25	26	27	28	29	30	31	X	X	X	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	–	–	–
21							21	22	23	24	25	26	27	28	29	30	31	X	X	X	X	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	–	–
22							22	23	24	25	26	27	28	29	30	31	X	X	X	X	X	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	–
23							23	24	25	26	27	28	29	30	31	X	X	X	X	X	X	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
24							24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	–	–	–	–	–	–	–
25							25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	X	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	–	–	–	–	–	–
26							26	27	28	29	30	31	0	1	2	3	4	5	6	7	X	X	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	–	–	–	–	–
27							27	28	29	30	31	0	1	2	3	4	5	6	7	X	X	X	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	–	–	–	–
28							28	29	30	31	0	1	2	3	4	5	6	7	X	X	X	X	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	–	–	–
28							29	30	31	0	1	2	3	4	5	6	7	X	X	X	X	X	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	–	–
30							30	31	0	1	2	3	4	5	6	7	X	X	X	X	X	X	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	–
31							31	0	1	2	3	4	5	6	7	X	X	X	X	X	X	X	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Latency Count	–	–	1	2	…	16	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–

Legend:

X = marks idle periods on the bus when RWDS does not toggle. – = indicates that the 64-byte wrapped burst has completed.

64-byte Wrapped Burst Address Sequence (Latency Code = 12)

Target Address	Clock Cycle																																												
	0	1	2	3	...	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52
0	CA0	CA1	CA2	Bus Turnaround + Initial Latency			0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	–	–	–	–	–	–	–
1							1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	–	–	–	–	–	–	–
2							2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	–	–	–	–	–	–	–
3							3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	–	–	–	–	–	–	–
4							4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	–	–	–	–	–	–	–
5							5	6	7	8	9	10	11	12	13	14	15	X	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	–	–	–	–	–	–
6							6	7	8	9	10	11	12	13	14	15	X	X	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	–	–	–	–	–
7							7	8	9	10	11	12	13	14	15	X	X	X	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	–	–	–	–
8							8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	–	–	–	–	–	–	–
9							9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	–	–	–	–	–	–	–
10							10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	–	–	–	–	–	–	–
11							11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	–	–	–	–	–	–	–
12							12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	–	–	–	–	–	–	–
13							13	14	15	16	17	18	19	20	21	22	23	X	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	–	–	–	–	–	–
14							14	15	16	17	18	19	20	21	22	23	X	X	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	–	–	–	–	–
15							15	16	17	18	19	20	21	22	23	X	X	X	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	–	–	–	–
23							16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	–	–	–	–	–	–	–
17							17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	–	–	–	–	–	–	–
18							18	19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	–	–	–	–	–	–	–
19							19	20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	–	–	–	–	–	–	–
20							20	21	22	23	24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	–	–	–	–	–	–	–
21							21	22	23	24	25	26	27	28	29	30	31	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	–	–	–	–	–	–
22							22	23	24	25	26	27	28	29	30	31	X	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	–	–	–	–	–
23							23	24	25	26	27	28	29	30	31	X	X	X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	–	–	–	–
24							24	25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	–	–	–	–	–	–	–
25							25	26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	–	–	–	–	–	–	–
26							26	27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	–	–	–	–	–	–	–
27							27	28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	–	–	–	–	–	–	–
28							28	29	30	31	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	–	–	–	–	–	–	–
28							29	30	31	0	1	2	3	4	5	6	7	X	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	–	–	–	–	–	–
30							30	31	0	1	2	3	4	5	6	7	X	X	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	–	–	–	–	–
31							31	0	1	2	3	4	5	6	7	X	X	X	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	–	–	–	–
Latency Count	–	–	1	2	...	12	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–

Legend:
X = marks idle periods on the bus when RWDS does not toggle. – = indicates that the 64-byte wrapped burst has completed.

6.2.10 ASO Entry and Exit

6.2.10.1 ID-CFI ASO

The system can access the ID-CFI ASO by issuing the ID-CFI Entry command sequence during Read Mode. This entry command uses the Sector Address (SA) in the command to determine which sector will be overlaid. See the detail descriptions in Table , *Command Definitions* on page 51, Table , *ID-CFI Address Map Overview* on page 17, Section 5.2.1, *Device ID* on page 17, and Section 5.2.2, *Common Flash Memory Interface* on page 18.

The ID-CFI ASO allows the following activities:

- Read ID-CFI ASO, using the same SA as used in the Entry command.
- ASO Exit.

The following is a C source code example of using the CFI Entry and Exit functions. Refer to the *Low Level Driver User Guide* for general information on Cypress flash memory software development guidelines.

```
/* Example: CFI Entry command */
*( (UINT16 *)base_addr + 0x55 ) = 0x0098; /* write CFI entry command */

/* Example: CFI Exit command */
*( (UINT16 *)base_addr + 0x000 ) = 0x00F0; /* write cfi exit command */
```

6.2.10.2 Status Register ASO

When the Status Register read command is issued, the current status is captured by the register and the ASO is entered. The first read access in the Status Register ASO exits the ASO and returns to the address space map in use when the Status Register read command was issued. No other command should be sent before reading the status to exit the Status Register ASO. The contents of the Status Register is only output as the first data value of a burst read, indeterminate data will be output during subsequent clock cycles.

6.2.10.3 Secure Silicon Region ASO

The system can access the Secure Silicon Region by issuing the Secure Silicon Region Entry command sequence during Read Mode. This entry command uses the Sector Address (SA) in the command to determine which sector will be overlaid.

The Secure Silicon Region ASO allows the following activities:

- Read Secure Silicon Region, using the same SA as used in the entry command. Reads within the overlaid SA but outside of the SSR will return indeterminate data.
- Reads to a SA outside of the Secure Silicon Region will retrieve array data. A read from the array will not cause an exit from the SSR ASO.
- Program the customer Secure Silicon Region using the Word or Write Buffer Programming commands.
- ASO Exit using legacy Secure Silicon Exit command for backward software compatibility.
- ASO Exit using the common exit command for all ASO – alternative for a consistent exit method.

6.2.10.4 ASP Configuration Register (ASPR) ASO

The system can access the ASP Configuration Register by issuing the ASP Configuration Register entry command sequence during Read Mode. This entry command does not use a sector address from the entry command. The ASP Configuration Register appears at word location 0 in the device address space. All other locations in the device address space are undefined.

The ASP Configuration Register ASO allows the following activities:

- Read ASP Configuration Register, using device address location 0.
- Program the customer ASP Configuration Register using a modified Word Programming command.
- ASO Exit using legacy Command Set Exit command for backward software compatibility.
- ASO Exit using the common exit command for all ASO – alternative for a consistent exit method.

6.2.10.5 Password ASO

The system can access the Password ASO by issuing the Password entry command sequence during Read Mode. This entry command does not use a sector address from the entry command. The Password appears at word locations 0 to 3 in the device address space. All other locations in the device address space are undefined.

The Password ASO allows the following activities:

- Read Password, using device address location 0 to 3.
- Program the Password using a modified Word Programming command.
- Unlock the PPB Lock Bit with the Password Unlock command.
- ASO Exit using legacy Command Set Exit command for backward software compatibility.
- ASO Exit using the common exit command for all ASO – alternative for a consistent exit method.

6.2.10.6 PPB ASO

The system can access the PPB ASO by issuing the PPB entry command sequence during Read Mode. This entry command does not use a sector address from the entry command. The PPB bit for a sector appears in bit 0 of all word locations in the sector.

The PPB ASO allows the following activities:

- Read PPB protection status of a sector in bit 0 of any word in the sector.
- Program the PPB bit using a modified Word Programming command.
- Erase all PPB bits with the PPB erase command.
- ASO Exit using legacy Command Set Exit command for backward software compatibility.
- ASO Exit using the common exit command for all ASO – alternative for a consistent exit method.

6.2.10.7 PPB Lock ASO

The system can access the PPB Lock ASO by issuing the PPB Lock entry command sequence during Read Mode. This entry command does not use a sector address from the entry command. The global PPB Lock bit appears in bit 0 of all word locations in the device.

The PPB Lock ASO allows the following activities:

- Read PPB Lock protection status in bit 0 of any word in the device address space.
- Clear the PPB Lock bit using a modified Word Programming command.
- ASO Exit using legacy Command Set Exit command for backward software compatibility.
- ASO Exit using the common exit command for all ASO – alternative for a consistent exit method.

6.2.10.8 Dynamic Protection Bits (DYB) ASO

The system can access the DYB ASO by issuing the DYB entry command sequence during Read Mode. This entry command does not use a sector address from the entry command. The DYB bit for a sector appears in bit 0 of all word locations in the sector.

The DYB ASO allows the following activities:

- Read DYB protection status of a sector in bit 0 of any word in the sector.
- Set the DYB bit using a modified Word Programming command.
- Clear the DYB bit using a modified Word Programming command.
- ASO Exit using legacy Command Set Exit command for backward software compatibility.
- ASO Exit using the common exit command for all ASO – alternative for a consistent exit method.

6.2.10.9 Software (Command) Reset / ASO exit

Software Reset is part of the command set (See Table , *Command Definitions* on page 51) that also returns the EAC to Standby state and must be used for the following conditions:

- Exit ID/CFI Mode
- Clear timeout bit (DQ5) for data polling when timeout occurs

Software Reset does not affect EA Mode. Reset commands are ignored once programming or erasure has begun, until the operation is complete. Software Reset does not affect outputs; it serves primarily to return to Read Mode from an ASO Mode or from a failed program or erase operation.

Software Reset may cause a return to Read Mode from undefined states that might result from invalid command sequences. However, a Hardware Reset may be required to return to normal operation from some undefined states.

There is no Software Reset latency requirement. The reset command is executed during the t_{WPH} period.

6.2.11 Status Register

The status of program and erase operations is provided by a single 16-bit Status Register. The Status Register Read command is issued followed by one read access of the Status Register information. The contents of the Status Register is aliased (overlaid) in all locations of the device address space. The overlay is in effect for one read access, specifically the next read access that follows the Status Register Read command. After the one Status Register access, the Status Register ASO is exited.

The Status Register contains bits related to the results – success or failure – of the most recently completed Embedded Algorithms (EA):

- Erase Status (bit 5),
- Program Status (bit 4),
- Write Buffer Abort (bit 3),
- Sector Locked Status (bit 1),
- Sector Erase Status Bit (bit 0).

and, bits related to the current state of any in process EA:

- Device Busy (bit 7),
- Erase Suspended (bit 6),
- Program Suspended (bit 2),

The current state bits indicate whether an EA is in process, suspended, or completed.

The upper 7 bits (bits 15:9) are reserved. These have an undefined High or Low value that can change from one status read to another. These bits should be treated as ‘don't care’ and ignored by any software reading status.

The Clear Status Register command and the Software Reset command will clear to 0 the results related bits of the Status Register (bits 5, 4, 3, 1, and 0) but will not affect the current state bits.

Status Register

Bit #	15:9	8	7	6	5	4	3	2	1	0
Bit Description	Reserved	Reserved	Device Ready Bit	Erase Suspend Status Bit	Erase Status Bit	Program Status Bit	Write Buffer Abort Status Bit	Program Suspend Status Bit	Sector Lock Status Bit	Sector Erase Status Bit
Bit Name			DRB	ESSB	ESB	PSB	WBASB	PSSB	SLSB	ESTAT
Reset Status	X	0	1	0	0	0	0	0	0	0
Busy Status	Invalid	Invalid	0	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
Ready Status	X		1	0 = No Erase in Suspension 1 = Erase in Suspension	0 = Erase Successful 1 = Erase Fail	0 = Program Successful 1 = Program Fail	0 = Program Not Aborted 1 = Program Aborted during Write to Buffer Command	0 = No Program in Suspension 1 = Program in Suspension	0 = Sector Not Locked during Operation 1 = Sector Locked Error	0 = Sector Erase Status Command Result = previous erase did not complete successfully 1 = Sector Erase Status Command Result = previous erase completed successfully

Notes:

1. *Bits 15 thru 9 are reserved for future use and may display as 0 or 1. These bits should be ignored (masked) when checking status.*
2. *Bit 7 is 1 when there is no Embedded Algorithm in progress in the device.*
3. *Bit 8 and bits 6 thru 0 are valid only if Bit 7 is 1.*
4. *All bits are put in their reset status by cold reset or warm reset.*
5. *Bits 5, 4, 3, 1, and 0 are cleared to 0 by the Clear Status Register command or Software Reset command.*
6. *Upon issuing the Erase Suspend command, the user must continue to read status until DRB becomes 1.*
7. *ESSB is cleared to 0 by the Erase Resume command.*
8. *ESB reflects success or failure of the most recent erase operation.*
9. *PSB reflects success or failure of the most recent program operation.*
10. *During Erase Suspend, programming to the suspended sector, will cause program failure and set the Program status bit to 1.*
11. *During Erase Suspend, an erase operation will cause an erase failure and set the Erase status bit to 1.*
12. *During Program Suspend, a programming operation will cause a program failure and set the Program status bit to 1.*
13. *During Program Suspend, an erase operation will cause an erase failure and set the Erase status bit to 1.*
14. *Upon issuing the Program Suspend command, the user must continue to read status until DRB becomes 1.*
15. *PSSB is cleared to 0 by the Program Resume command.*
16. *SLSB indicates that a program or erase operation failed because the target memory region was locked.*
17. *SLSB reflects the status of the most recent program or erase operation.*

6.2.12 Error Types and Clearing Procedures

There are three types of errors reported by the embedded operation status methods. Depending on the error type, the status reported and procedure for clearing the error status is different. The following is the clearing of error status:

- If an ASO was entered before the error the device remains entered in the ASO awaiting ASO read or a command write.
- If an erase was suspended before the error the device returns to the erase suspended state awaiting flash array read or a command write.
- Otherwise, the device will be in Standby state awaiting flash array read or a command write.

6.2.12.1 Embedded Operation Error (and Invalid Password)

If an error occurs during an embedded operation (program, erase, blank check, or password unlock) the Embedded Algorithm Controller remains active. The Status Register shows ready (SR[7] = 1) with valid status bits indicating the reason for the error. The Embedded Algorithm Controller remains active until the error status is detected by the host system status monitoring and the error status is cleared.

During embedded algorithm error status the Status Register will show the following:

- SR[7] = 1; Valid status displayed
- SR[6] = X; May or may not be erase suspended during the EA error
- SR[5] = 1 on erase or blank check error, else = 0
- SR[4] = 1 on program error or invalid password, else = 0
- SR[3] = X; Treat as ‘don’t care’ (masked)
- SR[2] = 0; No Program in suspension
- SR[1] = 0
- SR[0] = X; Treat as ‘don’t care’ (masked)

When the embedded algorithm error status is detected, it is necessary to clear the error status in order to return to normal operation, ready for a new read or command write. The error status can be cleared by writing:

- Reset command
- Status Register Clear command

Commands that are accepted during embedded algorithm error status are:

- Status Register Read
- Reset command
- Status Register Clear command

6.2.12.2 Protection Error

If an embedded algorithm attempts to change data within a protected area (program, or erase of a protected sector or OTP area) the device (EAC) goes busy for a period of 20 to 100 µs then returns to normal operation. Protection mechanisms include DYB, PPB, and Locks. During the busy period the Status Register shows not ready with invalid status bits (SR[7] = 0). If a programming or erase operation is attempted to a locked region the operation will be aborted and the failure will be indicated in the Status Register (see Table , *Status Register* on page 37).

Commands that are accepted during the protection error status busy period are:

- Status Register Read

When the busy period ends the device returns to normal operation, and the Status Register shows ready with valid status bits. The device is ready for flash array read or write of a new command.

After the protection error status busy period the Status Register will show the following:

- SR[7] = 1; Valid status displayed
- SR[6] = X; May or may not be erase suspended after the protection error busy period
- SR[5] = 1 on erase error, else = 0
- SR[4] = 1 on program or password unlock error, else = 0
- SR[3] = X; Treat as ‘don’t care’ (masked)
- SR[2] = 0; No Program in suspension
- SR[1] = 1; Error due to attempting to change a protected location
- SR[0] = X; Treat as ‘don’t care’ (masked)

Commands that are accepted after the protection error status busy period are:

- Any command

For cases where the Program Status Bit is set a further program operation will immediately clear SR[4]. For cases where the Erase Status Bit is set a further erase operation will immediately clear SR[6].

6.2.12.3 Write Buffer Abort

If an error occurs during a Write to Buffer command the device (EAC) remains busy. The Status Register shows ready with valid status bits. The device remains busy until the error status is detected by the host system status monitoring and the error status is cleared.

During embedded algorithm error status the Status Register will show the following:

- SR[7] = 1; Valid status displayed
- SR[6] = X; May or may not be erase suspended during the WBA error status
- SR[5] = 0; Erase successful
- SR[4] = 1; Programming related error, else = 0
- SR[3] = 1; Write buffer abort
- SR[2] = 0; No Program in suspension
- SR[1] = 0; Sector not locked during operation
- SR[0] = X; Treat as ‘don’t care’ (masked)

When the WBA error status is detected, it is necessary to clear the error status in order to return to normal operation, ready for a new read or command write. The error status can be cleared by writing:

- Write Buffer Abort Reset command
  - Clears the status register and returns to normal operation
- Status Register Clear command

Commands that are accepted during embedded algorithm error status are:

- Status Register Read
  - Reads the status register and returns to WBA busy state
- Write Buffer Abort Reset command
- Status Register Clear command

During an embedded algorithm, read transactions not associated with a Status Register Read will toggle RWDS and return indeterminate data.

6.3 Data Protection

6.3.1 Secure Silicon Region

Each device has a 1024-byte one-time programmable Secure Silicon Region (SSR) address space that is separate from the Flash Memory Array. The SSR area is divided into 32, individually lockable, 32-byte aligned and length regions.

In the 32-byte region starting at address zero:

- The 16 lowest address bytes are programmed by Cypress with a 128-bit random number. Only Cypress is able to program these bytes. Attempting to program 0s into these locations will fail and generate a Program Status Error (SR[4] = 1).
- The next 4 higher address bytes (SSR Lock Bytes) are used to provide one bit per SSR region to permanently protect each region from programming. The bytes are erased when shipped from Cypress. After an SSR region is programmed, it can be locked to prevent further programming, by programming the related protection bit in the SSR Lock Bytes.
- The next higher 12 bytes of the lowest address region are Reserved for Future Use (RFU). The bits in these RFU bytes may be programmed by the host system but it must be understood that a future device may use those bits for protection of a larger SSR space. The bytes are erased when shipped from Cypress.

The remaining regions are erased when shipped from Cypress, and are available for programming of additional permanent data.

Refer to Figure 6.6, *SSR Address Space* on page 42 for a pictorial representation of the SSR memory space.

The SSR memory space is intended for increased system security. SSR values, such as the random number programmed by Cypress, can be used to ‘mate’ a flash component with the system CPU / ASIC to prevent device substitution.

The configuration register SSR Freeze (xVCR10) bit protects the entire SSR memory space from programming when cleared (or programmed for NVCR) to 0. This allows trusted boot code to control programming of SSR regions then set the Freeze bit to prevent further SSR memory space programming during the remainder of normal power-on system operation.

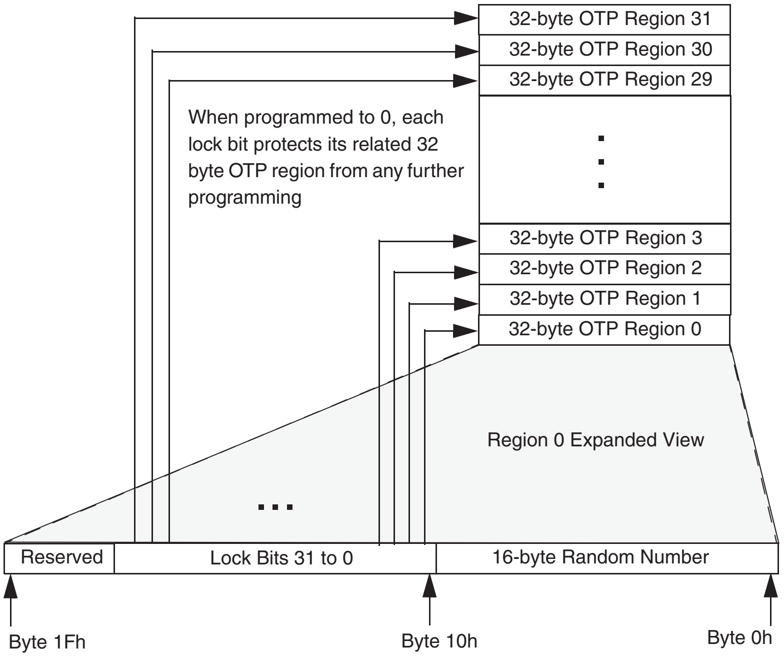
6.3.1.1 Reading the Secure Silicon Region Memory Space

Reading the SSR Region is performed once the SSR ASO is entered using the SSR entry sequence. The SSR is mapped to a specific sector identified during the SSR Entry command sequence. SSR Read operations within the sector identified during the SSR Entry command sequence but outside the valid 8-kB SSR address range will yield indeterminate data. Reads to sectors not overlaid by the SSR ASO will retrieve array data. A SSR Exit sequence will return the device to the array read ASO.

6.3.1.2 Programming Secure Silicon Region Memory Space

Programming the SSR memory is performed once the SSR ASO is entered using the SSR Entry sequence. The protocol of the SSR programming command is the same as normal array programming. The SSR programing sequences can be issued multiple times to any given SSR address, but this address space can never be erased. The valid address range for SSR Program is depicted in Figure 6.6, *SSR Address Space* on page 42. SSR Program operations outside the valid SSR address range will ignore address A9 and higher and will alias into the valid SSR address range. SSR Program operations while Freeze = 0 will fail with no indication of the failure. The SSR address space is not protected by the selection of an ASP Protection Mode. The Freeze SSR bit (xVCR.10) may be used to protect the SSR address space. A SSR Exit sequence will return the device to the Read Mode.

Figure 6.6 SSR Address Space



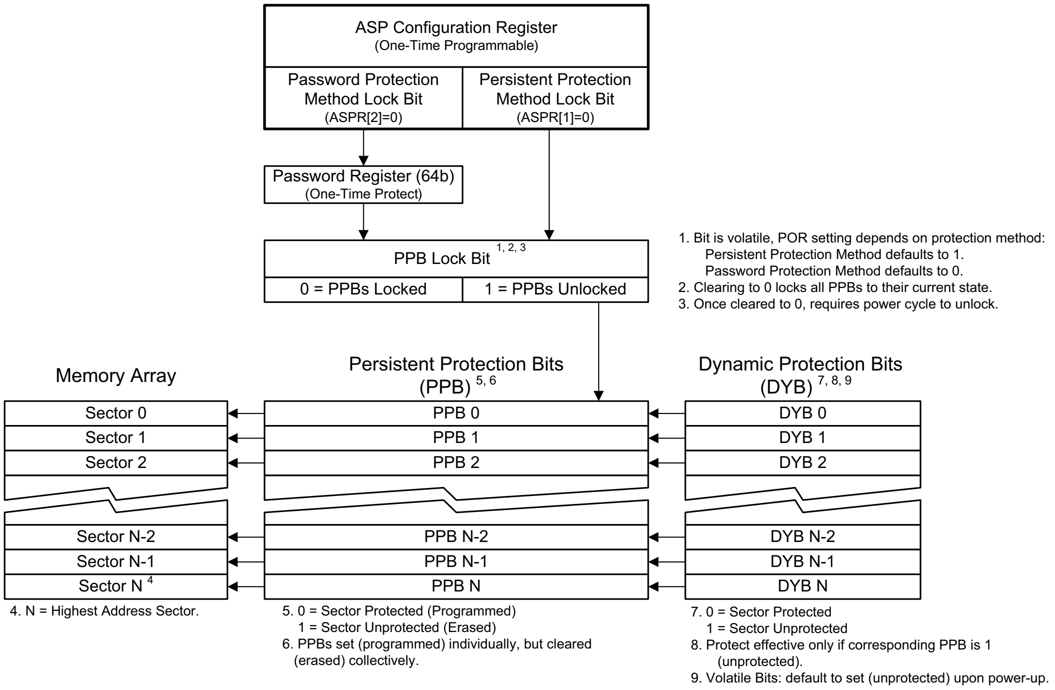
SSR Address Map

Region	Byte Address Range (Hex)	Contents	Initial Delivery State (Hex)
Region 0	0000	Least Significant Byte of Cypress Programmed Random Number	Cypress Programmed Random Number
	...	...	
	000F	Most Significant Byte of Cypress Programmed Random Number	
	0010 - 0013	Region Locking Bits Byte 10 [bit 0] locks region 0 from programming when = 0 ... Byte 13 [bit 7] locks region 31from programming when = 0	All Bytes = FF
	0014 - 001F	Reserved for Future Use (RFU)	All Bytes = FF
Region 1	0020 - 003F	Available for User Programming	All Bytes = FF
Region 2	0040 - 005F	Available for User Programming	All Bytes = FF
...	...	Available for User Programming	All Bytes = FF
Region 31	03E0 - 03FF	Available for User Programming	All Bytes = FF

6.3.2 Advanced Sector Protection (ASP)

Advanced Sector Protection (ASP) is a set of independent hardware and software methods used to disable or enable programming or erase operations, individually, in any or all sectors. This section describes the various methods of protecting data stored in the memory array. An overview of these methods is shown in Figure 6.7.

Figure 6.7 Advanced Sector Protection Overview



1. Bit is volatile, POR setting depends on protection method:
   Persistent Protection Method defaults to 1.
   Password Protection Method defaults to 0.
2. Clearing to 0 locks all PPBs to their current state.
3. Once cleared to 0, requires power cycle to unlock.

4. N = Highest Address Sector.

5. 0 = Sector Protected (Programmed)
   1 = Sector Unprotected (Erased)
6. PPBs set (programmed) individually, but cleared (erased) collectively.

7. 0 = Sector Protected
   1 = Sector Unprotected
8. Protect effective only if corresponding PPB is 1 (unprotected).
9. Volatile Bits: default to set (unprotected) upon power-up.

Every Flash Memory Array sector has a non-volatile (PPB) and a volatile (DYB) protection bit associated with it. When either bit is 0, the associated sector is protected from program and erase operations.

The PPB bits are protected from program and erase when the PPB Lock bit is 0. There are two methods for managing the state of the PPB Lock Bit, Persistent Protection and Password Protection.

The Persistent Protection method sets the PPB Lock Bit to 1 during POR or Hardware Reset so that the PPB bits are unprotected by a device reset. Software Reset does not affect the PPB Lock Bit. There is a command to clear the PPB Lock Bit to 0 to protect the PPB. There is no command in the Persistent Protection method to set the PPB Lock Bit therefore the PPB Lock Bit will remain at 0 until the next Power-Off or Hardware Reset. The Persistent Protection method allows boot code the option of changing sector protection by programming or erasing the PPB, then protecting the PPB from further change for the remainder of normal system operation by clearing (to 0) the PPB Lock Bit. This is sometimes called Boot-Code Controlled Sector Protection.

The Password method clears the PPB Lock Bit to 0 during POR or Hardware Reset to protect the PPB. A 64-bit password may be permanently programmed and hidden for the Password method. A command can be used to provide a password for comparison with the hidden password. If the password matches the PPB Lock Bit is set to 1 to unprotect the PPB. A command can be used to clear the PPB Lock Bit to 0. This method requires use of a password to control PPB Protection.

The selection of the PPB Lock management method is made by programming OTP bits in the ASP Configuration Register so as to permanently select the method used.

The PPB bits are erased so that all Flash Memory Array sectors are unprotected when shipped from Cypress.

6.3.3 PPB Lock

The Persistent Protection Lock Bit is a volatile bit for protecting all PPB bits. When cleared to 0, it locks all PPBs and when set to 1, it allows the PPBs to be changed. There is only one PPB Lock Bit per device.

The PPB Lock command is used to clear the bit to 0. The PPB Lock Bit must be cleared to 0 only after all the PPBs are configured to the desired settings.

In Persistent Protection Mode, the PPB Lock Bit is set to 1 during POR or a Hardware Reset. When cleared with the PPB Lock Bit Clear sequence, no software command sequence can set the PPB Lock Bit, only another Hardware Reset or Power-Up can set the PPB Lock Bit.

In the Password Protection Mode, the PPB Lock Bit is cleared to 0 during POR or a Hardware Reset. The PPB Lock Bit can only set to 1 by the Password Unlock command sequence. The PPB Lock Bit can be cleared back to 0 with the PPB Lock Bit Clear sequence.

6.3.4 Persistent Protection Bits (PPB)

The Persistent Protection Bits (PPB) are located in a separate non-volatile flash array. One of the PPB bits is assigned to each sector. When a PPB is programmed to 0 its related sector is protected from program and erase operations. The PPB are programmed individually but must be erased as a group, similar to the way individual words may be programmed in the main array but an entire sector must be erased at the same time. Preprogramming and verification prior to erasure are handled by the EAC.

Programming a PPB bit requires the typical word programming time. During a PPB bit programming operation or PPB bit erasing, the Status Register can be accessed to determine when the operation has completed. Erasing all the PPBs requires typical sector erase time.

If the PPB Lock Bit is 0, the PPB Program or erase command does not execute and times-out without programming or erasing the PPB. If an program or erase operation is attempted to the PPB bits when the PPB Lock Bit is 0 the operation will be aborted and the failure will be indicated in the Status Register (see Table , *Status Register* on page 37).

The protection state of a PPB for a given sector can be verified by writing a PPB Status Read command when entered in the PPB ASO.

6.3.5 Dynamic Protection Bits (DYB)

Dynamic Protection Bits are volatile and unique for each sector and can be individually modified. DYBs only control protection for sectors that have their PPBs cleared. By issuing the DYB Set or Clear command sequences, the DYB are set to 0 or cleared to 1, thus placing each sector in the protected or unprotected state respectively. This feature allows software to easily protect sectors against inadvertent changes, yet does not prevent the easy removal of protection when changes are needed.

The DYB can be set to 0 or cleared to 1 as often as needed.

6.3.6 Sector Protection States Summary

Each sector can be in one of the following protection states:

- Unlocked – The sector is unprotected and protection can be changed by a simple command. The protection state defaults to unprotected after a power cycle or Hardware Reset.
- Dynamically Locked – A sector is protected and protection can be changed by a simple command. The protection state is not saved across a power cycle or Hardware Reset.
- Persistently Locked – A sector is protected and protection can only be changed if the PPB Lock Bit is set to 1. The protection state is non-volatile and saved across a power cycle or Hardware Reset. Changing the protection state requires programming or erase of the PPB bits.

Sector Protection States

Protection Bit Values			Sector State
PPB Lock Bit	PPB	DYB	
1	1	1	Unprotected – PPB and DYB are changeable
1	1	0	Protected – PPB and DYB are changeable
1	0	1	Protected – PPB and DYB are changeable
1	0	0	Protected – PPB and DYB are changeable
0	1	1	Unprotected – PPB not changeable, DYB is changeable
0	1	0	Protected – PPB not changeable, DYB is changeable
0	0	1	Protected – PPB not changeable, DYB is changeable
0	0	0	Protected – PPB not changeable, DYB is changeable

6.3.7 ASP Configuration Register

The ASP Configuration Register (ASPR) holds the non-volatile OTP bits for controlling security management.

ASP Configuration Register

Bit	Default Value	Name
15, 14, 13, 12, 10, 9	1	Reserved
11	1	Hybrid Burst Type Enable (not available on the 512-Mb device) ASPR[11] = 0: Hybrid – One Wrapped burst sequence followed by linear burst ASPR[11] = 1: Legacy – Wrapped burst sequence only
8	0	Reserved
7	X	Reserved
6	1	Reserved
5	1	Reserved
4	1	Reserved
3	1	Reserved
2-1	1	Persistent / Password Protection Mode Lock Bits ASPR[2:1] = 00: Not Allowed ASPR[2:1] = 01: Password Mode Permanently Enabled, ASPR Frozen ASPR[2:1] = 10: Persistent Mode Permanently Enabled, ASPR Frozen ASPR[2:1] = 11: Persistent Mode Temporarily Enabled (default from factory)
0	1	Reserved

As shipped from the factory, all devices default to the Persistent Protection method, with all sectors unprotected, when power is applied. The device programmer or host system can then choose which sector protection method to use. Programming either of the following two, one-time programmable, non-volatile bits, locks the part permanently in that mode:

- Persistent Protection Mode Lock Bit (ASPR[1])
- Password Protection Mode Lock Bit (ASPR[2])

If both lock bits (ASPR[2] and ASPR[1]) are selected to be programmed at the same time, the operation will abort and Status Register bits SR[4] and SR[1] will be set to indicate the failure. Once the Password Mode Lock Bit is programmed, the Persistent Mode Lock Bit is permanently disabled and no changes to the protection scheme are allowed. Similarly, if the Persistent Mode Lock Bit is programmed, the Password Mode is permanently disabled. Programming attempts to the ASPR after either ASPR[2] or ASPR[1] have been programmed will be aborted and Status Register bits SR[4] and SR[1] will be set to indicate the failure.

If the Password Mode is chosen, the password must be programmed prior to setting the corresponding Lock Register Bit. The four-word password must be programmed in order in a 0-1-2-3 sequence, other programming sequences will result in undefined behavior. After the Password Protection Mode Lock Bit is programmed, a power cycle, Hardware Reset, or PPB Lock Bit Set command is required to set the PPB Lock Bit to 0 to protect the PPB array.

The programming time of the ASP Configuration Register is the same as the typical word programming time. During a ASP Configuration Register programming EA. The system can also determine the status of the ASPR programming by reading the Status Register. See *Status Register* on page 36 for information on these status bits.

6.3.8 Persistent Protection Mode

The Persistent Protection method sets the PPB Lock Bit to 1 during POR or Hardware Reset so that the PPB bits are unprotected by a device reset. There is a command to clear the PPB Lock Bit to 0 to protect the PPB. There is no command in the Persistent Protection method to set the PPB Lock Bit to 1 therefore the PPB Lock Bit will remain at 0 until the next power-off or Hardware Reset.

6.3.9 Password Protection Mode

Password Protection Mode allows an even higher level of security than the Persistent Sector Protection Mode, by requiring a 64-bit password for setting the PPB Lock Bit. In addition to this password requirement, after Power-Up or Hardware Reset, the PPB Lock Bit is cleared to 0 to ensure protection at Power-Up. Successful execution of the Password Unlock command by entering the entire password sets the PPB Lock Bit to 1, allowing for sector PPB modifications.

Password Protection Notes:

- The Password Program Command is only capable of programming 0’s.
- The password is all 1’s when shipped from Cypress. It is located in its own memory space and is accessible through the use of the Password Program and Password Read commands.
- All 64-bit password combinations are valid as a password.
- Once the Password is programmed and verified, the Password Protection Mode Locking Bit must be programmed (to 0) in order to prevent reading the password.
- The Password Protection Mode Lock Bit, once programmed (to 0), prevents reading the 64-bit password on the data bus and further password programming. All further program and read commands to the password region are disabled and these commands are ignored. Attempted programming of a protected Password will set the Sector Lock Status Bit (SR[1]) and the Program Status Bit (SR[4]). If a further programming operation is attempted on either the Password or the Password Protection Mode Lock Bit the operation will be aborted and the failure will be indicated in the Status Register (see Table , *Status Register* on page 37). There is no means to verify what the password is after the Password Protection Mode Lock Bit is programmed. Password verification is only allowed before selecting the Password Protection Mode.
- The Password Mode Lock Bit is not erasable.
- For the unlocking function to occur, the password portion can be entered in any order as long as the entire 64-bit password is entered. If the password unlock command provided password does not match the hidden internal password, the unlock operation fails in the same manner as a programming operation on a protected sector. The Status Register will return to the ready state with the Program Status Bit set to 1 indicating a failed programming operation due to a locked sector. In this case it is a failure to change the state of the PPB Lock Bit because it is still protected by the lack of a valid password.
- The device requires approximately t_{PSWD} = 100 µs for setting the PPB Lock Bit after the valid 64-bit password is given to the device.
- The Password Unlock command cannot be accepted any faster than once every t_{PSWD} (see Table , *Embedded Algorithm Characteristics* on page 55). This makes it take an unreasonably long time (58 million years) for a hacker to run through

all the 64-bit combinations in an attempt to correctly match a password. The EA status checking methods may be used to determine when the EAC is ready to accept a new password command.

- If the password is lost after setting the Password Mode Lock Bit, there is no way to clear the PPB Lock Bit.

6.3.10 Read Password Protection Mode

The Read Password Mode can replace the default *Password Protection Mode*. The Read Password Mode is enabled to replace the default PPB Password Protection Mode when the user programs ASPR[5] = 0. The Read Password Mode is not active until the password is programmed and ASPR[2] is programmed to 0.

The Read Password Protection Mode enables protecting the Flash Memory Array from read, program and erase. Only the lowest or highest (256-kB) sector address range, selected by the Non-Volatile Configuration Register bits xVCR[9:8], remains readable until a successful Password Unlock command is completed. Note that reads from the read-protected portion of the array will alias back to the readable sector.

In this mode the PPB Lock Bit is used to control the high order bits of address. When the PPB Lock Bit is 1, the address bits operate normally. When the PPB Lock Bit is 0, the address bits that select a main array sector address range are forced either to 0s (xVCR[9:8] = 00 or 10) or to 1s (xVCR[9:8] = 01 or 11) to select the lowest or highest address Flash Memory Array address range per the table below. When xVCR[9:8] = 00 or 10, the bottom (zero address) 256 kB of the array is readable. When xVCR[9:8] = 01 or 10, the top (maximum address) 256 kB of the array is readable.

ASP Configuration Register Selection of Persistent and Password Protection Modes

ASP Bit	Default Value	Name
2	1	Persistent / Password Protection Mode Lock Bits ASPR[2:1] = 00: Not Allowed ASPR[2:1] = 01: Password Mode Permanently Enabled ASPR[2:1] = 10: Persistent Mode Permanently Enabled ASPR[2:1] = 11: Persistent Mode Temporarily Enabled (default from factory)
1	1	

xVCR Mapping of Boot Block Address Range

xVCR Bit	Default Value	Name
xVCR.9 – xVCR.8	11	00 - Map Parameter-Sectors and Read Password Sectors mapped into lowest addresses. 01 - Map Parameter-Sectors and Read Password Sectors mapped into highest addresses. 10 - Uniform Sectors with Read Password Sector mapped into lowest addresses. 11 - Uniform Sectors with Read Password Sector mapped into highest addresses.

The PPB bits are protected from program and erase when the PPB Lock Bit is 0 and may be programmed or erased when the PPB Lock Bit is 1.

The PPB Lock Bit is set to 0 by POR or Hardware Reset, same as in PPB Password Protection Mode.

Read Password Protection Notes:

- When the Read Password OPN option is ordered, the user can program the ASPR[5] bit to 0 and use Read Password, or not, as desired.
- The command sequence for programming, reading, and locking of the Password for Read Password Method is the same as the default for the PPB Password Method.
- When the Read Password Mode and Password Protection Mode are enabled (i.e. ASPR[2] and ASPR[5] are programmed to 0), then all addresses are redirected to the Boot Sector until the password unlocking sequence is properly entered, with the correct password. At which time, the Read Password Mode is disabled and all addressing will select the proper location.
- If a system Hardware Reset occurs, then the Read Password Mode is re-enabled.
- ASPR[5] is used to select between Read Password versus PPB Password options. If ASPR[5] = 0 then the device is ready for Read Password. However, Read Password is not enabled until ASPR[2] = 0. At which point, all addresses select only within the top or bottom sectors, until the device is unlocked with the proper unlocking sequence and

Password. When ASPR[2] = 1 the addresses select normally. This allows users to program in code, test it, provide a password, and then lock it by programming ASPR[2] = 0.

- The Read Password command sequence return undefined results if sent when Read Password Protection is in use. The PPB Lock Bit may only be returned to 0 by a Hardware Reset, POR or the PPB Lock Bit Clear command sequence.
- Only the ID Read command, Password Unlock command and array reads are valid during Read Password Mode while the PPB Lock Bit = 0. Other commands are disabled until the password is supplied to enable reading of the entire device and normal command operation.
- When Read Password Protection Mode is active (ASPR[5] = 0, ASPR[2] = 0, PPB Lock Bit = 0), reading of the main array is allowed but forced to have only the boot sector visible via the forcing of memory sector address to 0 or 1s. Reading the DYB, or PPB address space returns undefined data.
- Programming memory spaces or writing registers is not allowed when Read Password Protection Mode is active. RESET operates normally, and bus protocol may be modified by resetting mode bits.

6.3.11 Hybrid Burst

An additional type of burst that combines one wrapped burst followed by linear burst, is supported by the 256 Mbit and lower density members of the HyperFlash family. The 512 Mbit density HyperFlash does not support this type of burst.

The beginning of a hybrid burst will wrap once within the target address wrapped burst length group, before switching to linear burst of data beyond the end of the initial wrapped burst length group. Hybrid burst is supported for 16-byte and 32-byte but not 64-byte wrapped burst length groups.

ASP Configuration Register Bit Assignment for Hybrid Burst Type Enable

Bit	Default Value	Name
11	1	Hybrid Burst Type Enable ASPR[11] = 0: Hybrid - One Wrapped burst sequence followed by linear burst ASPR[11] = 1: Legacy - Wrapped burst sequence only

Example burst sequences for 32-byte, and 16-byte Hybrid burst reads:

1. 32-byte example (wrap within 32-byte boundary before transitioning to linear burst)
   a. 06-07-08-09-0A-0B-0C-0D-0E-0F-00-01-02-03-04-05-10-11
   b. 0E-0F-00-01-02-03-04-05-06-07-08-09-0A-0B-0C-0D-10-11
2. 16-byte example (wrap within 16-byte boundary before transitioning to linear burst)
   a. 06-07-00-01-02-03-04-05-08-09
   b. 03-04-05-06-07-00-01-02-08-09

6.3.12 INT# Output

The INT# pin is an open-drain output used to indicate to the host system that an event has occurred within the flash device. The user can select to transition the INT# output pin to the active (Low) state when:

- Transitioning from the Busy to the Ready state

Operation is controlled with the Interrupt Configuration Register (ICR) where the INT# output (normally High) is enabled. The Interrupt Configuration Register determines when an internal event is enabled to trigger a High to Low transition on the INT# output pin. The Interrupt Status Register indicates what enabled internal event(s) have occurred since the last time the ISR has been cleared. If enabled, the INT# output pin will then transition from High to Low upon the occurrence of an enabled event. Once the host recognizes that INT# has transitioned to the Low state the Interrupt Status Register can be read to determine which internal event was responsible. The INT# output can be forced to transition back to the high impedance state (returned High by an external pull-up resistance) using three methods:

- Disable the INT# output by loading a 1 into bit 15 of the Interrupt Configuration Register. The Interrupt Status Register will be cleared upon loading ICR[15] with a 1.

- Disable the event channel responsible for causing the output to transition Low by loading a 1 into the appropriate event enable bit in the Interrupt Configuration Register. The associated bit in the Interrupt Status Register will be cleared upon loading a 1 into the corresponding bit in the ICR.
- Reset the appropriate bit (by writing a 1) in the Interrupt Status Register bit that indicates which internal event occurred to cause the output to go Low. All Interrupt Status Register bits that are Low and are also enabled in the Interrupt Configuration Register must be reset before the INT# output will return High.

The INT# output will also be returned to the default (disabled, High-Z) state with a Hardware Reset (RESET# = Low) or a Power-On Reset. Hardware Reset and Power-On Reset disable all interrupts by setting the Interrupt Configuration Register back to the default (all interrupts disabled) state.

Interrupt Configuration Register

Bits	Function	Type	POR Default State	RESET# Default State	Description
15	INT# Output Enable	Volatile, Read / Write	1	1	1 = INT# output disabled (High or Open-Drain) 0 = INT# output enabled, internal events will cause a High to Low transition
14	Reserved		1	1	Reserved
13 to 5	Reserved		1	1	Reserved for Future Use
4	READY		1	1	1 = Ready/Busy transitions will not transition the INT# output 0 = A Busy to Ready transition will cause a High to Low transition on the INT# output
3	Reserved		1	1	Reserved for future use
2	Reserved		1	1	Reserved for future use
1	Reserved		1	1	Reserved for future use
0	Reserved		1	1	Reserved for future use

Note:
1. *Both POR and Hardware Reset results in all interrupt channels being disabled.*

Interrupt Status Register

Bits	Function	Type	POR Default State	RESET# Default State	Description
15 to 5	Reserved	Volatile, Read / Write	1	1	Reserved for Future Use
4	READY		1	1	1 = A Busy to Ready transition has not occurred 0 = A Busy to Ready transition has occurred
3	Reserved		1	1	Reserved for future use
2 (3), (4)	POR Detect		0	1	1 = POR has not occurred 0 = POR has occurred
1	Reserved		1	1	Reserved for future use
0	Reserved		1	1	Reserved for future use

Notes:
1. *Hardware Reset results in all ISR bits being set to 1.*
2. *POR results in the ISR POR Detect bit (ISR[2]) being cleared to 0, all other bits to be set to 1.*
3. *ISR[2] is cleared (to 0) during POR and is only set (to 1) with a Hardware Reset (RESET# = 0) or with a write to the ISR.*
4. *The INT# output state is not affected by the value of ISR[2].*
5. *Writing to the ISR can only flip bits from the 0 to the 1 state. Only an interrupt occurrence flips an ISR bit from the 1 to the 0 state.*

6.4 Volatile and Non-Volatile Configuration Register Summary

6.4.1 Non-Volatile Configuration Registers

Non-Volatile Configuration Registers

Symbol	Name	Width (bits)	NV Type	Default Value	Reference
NVCR	Non-Volatile Configuration Register	16	P/E	8EBBh	6.2.9, *Non-Volatile Configuration Register and Volatile Configuration Register* on page 29
	Password Protection Register	64	OTP	FFFF FFFF FFFF FFFFh	6.3.9, *Password Protection Mode* on page 46
PPB	Persistent Protection Bits	1-bit per sector	P/E	1	6.3.4, *Persistent Protection Bits (PPB)* on page 44
ASPR	ASP Configuration Register	16	OTP	FEFFh	6.3.7, *ASP Configuration Register* on page 45
PORTime	Power-On Reset Time	16	OTP	FFFFh	Section 11.3.1, *Power-On (Cold) Reset (POR)* on page 66

6.4.2 Volatile Configuration Registers

Volatile Configuration Registers

Symbol	Name	Width (bits)	Default Value	Reference
VCR	Volatile Configuration Register	16	NVCR	6.2.9, *Non-Volatile Configuration Register and Volatile Configuration Register* on page 29
DYB	Dynamic Protection Bits	1-bit per sector	1	6.3.5, *Dynamic Protection Bits (DYB)* on page 44
	PPB Lock Bit	1	ASPR[2]	6.3.3, *PPB Lock* on page 44
ICR	Interrupt Configuration Register	16	FFFFh	6.3.12, *INT# Output* on page 48

6.4.3 Volatile Results and Status Registers

Volatile Results and Status Registers

Name	Width (bits)	Default Value	Reference
Sector Lock Status	4-bit per sector	NA	See (Note 17) for Table , *Command Definitions* on page 51
Status Register	16	xx80h	Table , *Status Register* on page 37
Interrupt Status Register	16	FFFBh	Table , *Interrupt Status Register* on page 49

7. Software Interface Reference

7.1 Command Summary

Command Definitions

Command Sequence	Cycles	Bus Cycles (Notes 1 - 4)													
		First		Second		Third		Fourth		Fifth		Sixth		Seventh	
		Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Read (5)	1	RA	RD												
Reset / ASO Exit (6), (14)	1	XXX	F0												
Status Register Read (16)	2	555	70	XXX	RD										
Status Register Clear	1	555	71												
Enter Deep Power-Down	3	555	AA	2AA	55	XXX	B9								
Program Power-On Reset Timer Register	4	555	AA	2AA	55	555	34	XXX	PORTime						
Read Power-On Reset Timer Register	4	555	AA	2AA	55	555	3C	XXX	RD PORTime						
Load Interrupt Configuration Register	4	555	AA	2AA	55	555	36	XXX	ICR						
Read Interrupt Configuration Register	4	555	AA	2AA	55	555	C4	XXX	RD ICR						
Load Interrupt Status Register	4	555	AA	2AA	55	555	37	XXX	ISR						
Read Interrupt Status Register	4	555	AA	2AA	55	555	C5	XXX	RD ISR						
Load Volatile Configuration Register	4	555	AA	2AA	55	555	38	XXX	VCR						
Read Volatile Configuration Register	4	555	AA	2AA	55	555	C7	XXX	RD VCR						
Program Non-Volatile Configuration Register	4	555	AA	2AA	55	555	39	XXX	NVCR						
Erase Non-Volatile Configuration Register	3	555	AA	2AA	55	555	C8								
Read Non-Volatile Configuration Register	4	555	AA	2AA	55	555	C6	XXX	RD NVCR						
Word Program	4	555	AA	2AA	55	555	A0	PA	PD						
Write to Buffer	6	555	AA	2AA	55	SA	25	SA	WC	WBL	PD	WBL	PD		
Program Buffer to Flash (confirm)	1	SA	29												
Write-to-Buffer-Abort Reset (10)	3	555	AA	2AA	55	555	F0								
Chip Erase	6	555	AA	2AA	55	555	80	555	AA	2AA	55	555	10		
Sector Erase	6	555	AA	2AA	55	555	80	555	AA	2AA	55	SA	30		
Blank Check	1	(SA) 555	33												
Evaluate Erase Status	1	(SA) 555	D0												
Erase Suspend	1	XXX	B0												
Erase Resume	1	XXX	30												
Program Suspend	1	XXX	51												
Program Resume	1	XXX	50												

Command Definitions (Continued)

	Command Sequence	Cycles	Bus Cycles (Notes 1 - 4)													
			First		Second		Third		Fourth		Fifth		Sixth		Seventh	
			Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
ID-CFI (Autoselect) ASO (23)	ID (Autoselect) Entry	3	555	AA	2AA	55	(SA) 555	90								
	CFI Enter (Note 7)	3	(SA) 55	98												
	ID-CFI Read	1	(SA) RA	RD												
	Reset / ASO Exit (6), (14)	1	XXX	F0 or FF												
	Secure Silicon Region Command Definitions															
Secure Silicon Region (SSR) ASO	SSR Entry	3	555	AA	2AA	55	(SA) 555	88								
	Read (5)	1	RA	(SA) RD												
	Word Program	4	555	AA	2AA	55	555	A0	PA	PD						
	Write to Buffer	6	555	AA	2AA	55	SA	25	SA	WC	WBL	PD	WBL	PD		
	Program Buffer to Flash (confirm)	1	SA	29												
	Write-to-Buffer-Abort Reset (10)	3	555	AA	2AA	55	555	F0								
	SSR Exit (10)	4	555	AA	2AA	55	555	90	XX	00h						
	Reset / ASO Exit (6), (14)	1	XXX	F0												
ASP Config. Register (ASPR) ASO	ASP Register Entry	3	555	AA	2AA	55	555	40								
	Program	2	XXX	A0	XXX	PD										
	ASPR Read (16)	1	0	RD												
	ASPR ASO Exit (10)	2	XXX	90	XXX	0										
	Reset / ASO Exit (6), (14)	1	XXX	F0												
	Password Protection Command Set Definitions															
Password ASO	Password ASO Entry	3	555	AA	2AA	55	555	60								
	Program (12)	2	XXX	A0	PWA x	PWDx										
	Read	4	0	PWD 0	1	PWD 1	2	PWD 2	3	PWD 3						
	Unlock	7	0	25	0	3	0	PWD 0	1	PWD 1	2	PWD 2	3	PWD 3	0	29
	Command Set Exit (11), (14)	2	XXX	90	XXX	0										
	Reset / ASO Exit (6), (14)	1	XXX	F0												

Command Definitions (Continued)

	Command Sequence	Cycles	Bus Cycles (Notes 1 - 4)													
			First		Second		Third		Fourth		Fifth		Sixth		Seventh	
			Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Non-Volatile Sector Protection Command Set Definitions																
PPB (Non-Volatile Sector Protection)	PPB Entry	3	555	AA	2AA	55	555	C0								
	PPB Program (15)	2	XXX	A0	SA	0										
	All PPB Erase (15)	2	XXX	80	0	30										
	PPB Read (15), (16)	1	SA	RD (0)												
	SA Protection Status (16), (17)	2	XXX	60	SA	RD										
	Command Set Exit (11), (14)	2	XXX	90	XXX	0										
	Reset / ASO Exit (6), (14)	1	XXX	F0												
Global Non-Volatile Sector Protection Freeze Command Set Definitions																
PPB Lock Bit	PPB Lock Entry	3	555	AA	2AA	55	555	50								
	PPB Lock Bit Clear	2	XXX	A0	XXX	0										
	PPB Lock Status Read (16)	1	XXX	RD (0)												
	Command Set Exit (11), (14)	2	XXX	90	XXX	0										
	Reset / ASO Exit (14)	1	XXX	F0												
Volatile Sector Protection Command Set Definitions																
DYB (Volatile Sector Protection) ASO	DYB ASO Entry	3	555	AA	2AA	55	555	E0								
	DYB Set (15)	2	XXX	A0	SA	0										
	DYB Clear (15)	2	XXX	A0	SA	1										
	DYB Status Read (16)	1	SA	RD (0)												
	SA Protection Status (15), (16), (17)	2	XXX	60	SA	RD										
	Command Set Exit (11), (14)	2	XXX	90	XXX	0										
	Reset / ASO Exit (14)	1	XXX	F0												

Command Definitions Legend:

X = Don't care.

RA = Address of the memory to be read.

RD = Data read from location RA during read operation.

PA = Address of the memory location to be programmed.

PD = Data to be programmed at location PA.

SA = Address of the sector selected. Address bits A_{MAX}-A17 for 256-kB sectors and A_{MAX}-A11 for 4-kB parameter sectors uniquely select any sector.

WBL = Write Buffer Location. The address must be within the same Line.

WC = Word Count is the number of write buffer locations to load minus 1.

PWAx = Password address for word0 = 00h, word1 = 01h, word2 = 02h, and word3 = 03h.

PWDx = Password data word0, word1, word2, and word3.

Notes:

1. *All values are in hexadecimal. All addresses reference 16-bit words.*
2. *Except for the following, all bus cycles are write cycle: read cycle during Read, ID/CFI Read (Manufacturing ID / Device ID), Indicator Bits, Secure Silicon Region Read, SSR Lock Read, and 2nd cycle of Status Register Read.*
3. *Data bits DQ15-DQ8 are 'don't care' in command sequences, except for RD, PD, WC and PWD.*
4. *Address bits A_{MAX}-A11 are 'don't care' for unlock and command cycles, unless SA or PA required. (A_{MAX} is the Highest Address pin.)*
5. *No unlock or command cycles required when reading array data.*
6. *The Reset command is required to return to reading array data when device is in the ID-CFI (Autoselect) Mode, or if DQ5 goes High (while the device is providing status data).*

7. Command is valid when device is ready to read array data or when device is in ID-CFI (Autoselect) Mode.

8. The system can read and program / program suspend in non-erasing sectors, or enter the ID-CFI ASO, when in the Erase Suspend Mode. The Erase Suspend command is valid only during a sector erase operation.

9. The Erase Resume / Program Resume command is valid only during the Erase Suspend / Program Suspend Modes.

10. Issue this command sequence to return to READ Mode after detecting device is in a Write-to-Buffer-Abort state. Note that the full command sequence is required if resetting out of ABORT.

11. The Exit command returns the device to reading the array.

12. For PWDx, only one portion of the password can be programmed per each ‘A0’ command. Portions of the password must be programmed in sequential order (PWD0-PWD3).

13. All ASP Register bits are one-time programmable. The program state = 0 and the erase state = 1. Both the Persistent Protection Mode Lock Bit and the Password Protection Mode Lock Bit cannot be programmed at the same time or the ASPR Register Bits Program operation aborts and returns the device to Read Mode. ASPR Register bits that are reserved for future use are undefined and may be 0’s or 1's.

14. If any of the Entry commands was issued, an Exit command must be issued to reset the device into Read Mode.

15. Bit 0 = 0 indicates the Protected State, Bit 0 = 1 indicates the Unprotected State. Bits 1 through 15 are all 1s. The sector address for DYB set, DYB clear, or PPB Program command may be any location within the sector; the lower order bits of the sector address are ‘don't care’.

16. Data out during a Status Register Read transaction, DYB Read, PPB Read, SA Protection Read, Password Read, POR Timer Read, ICR Read, ISR Read, VCR Read, NVCR Read, FIDR Read, ASPR Read, PPBL Read Register Read transactions is only valid during the first word output by the device. Subsequent data values output if CK/CK# continue to toggle while CS# remains Low are undefined.

17. Data out during a SA Protection Status Read indicates whether the indicated sector is protected in bits 0-3.

18. Bit 0 – Indicates whether the indicated sector is protected (0 = protected, 1 = unprotected).

19. Bit 1 – Protected using the sector’s DYB bit (0 = protected, 1 = unprotected).

20. Bit 2 – Protected using the sector’s PPB bit (0 = protected, 1 = unprotected).

21. Bits 4 through 15 are all 1s.

22. The smaller parameter-sectors need to include A[16:11] as part of the address identifying the target parameter-sector during erase and program command sequences.

23. Both the ID (Autoselect) Entry and the CFI Entry allows access to the same ID/CFI data set. All data contained in within the ID/CFI data set is available after using either the ID or CFI Entry sequences.

24. ‘Non-512 Mb’ devices only.

8. Embedded Algorithm Performance

Embedded Algorithm Characteristics

Parameter	Min	Typ (1)	Max (2)	Unit	Comments
Sector Erase Time 256-Kbyte	–	930	2900	ms	Includes pre-programming prior to erasure (4)
Parameter Sector Erase Time 4-Kbyte	–	240	725	ms	
Chip Erase Time (128 Mb)		55	115	s	
Chip Erase Time (256 Mb)	–	110	231	s	
Chip Erase Time (512 Mb)	–	220	462	s	
Single Word Programming Time	–	500	1260	µs	Word Programming Command Sequence
Half-page (16-byte) Buffered Programming Time	–	270	1000	µs	Buffered Programming Command Sequence
Buffer Programming Time (full 512-byte)	–	475	2000	µs	
Erase Suspend / Erase Resume (t_{ESL})	–		50	µs	
Program Suspend / Program Resume (t_{PSL})	–		50	µs	
Erase Resume to next Erase Suspend (t_{ERS})	–	100		µs	Minimum of 60 ns but ≥ typical periods are needed for Erase to progress to completion
Program Resume to next Program Suspend (t_{PRS})	–	100		µs	Minimum of 60 ns but ≥ typical periods are needed for Program to progress to completion
Blank Check (256-kB Sector)	–	15	17	ms	
NOP (Number of Program-operations, per Line)	–		256		Industrial Temperature
	–		32		Industrial Plus Temperature Only a single program operation on each 8-word (16-byte) half-page
Evaluate Erase Status Time (t_{EES})	–	70	100	µs	
Password Comparison Time (t_{PSWD})	80	100	120	µs	

Notes:

1. *Typical program and erase times assume the following conditions: 25°C, (1.8V or 3.0V) V_{CC}, 10,000 cycle, and a checkerboard data pattern.*
2. *Under worst case conditions of 90°C, V_{CC} = (1.70V or 2.7V), 100,000 cycles, and a random data pattern.*
3. *Effective write buffer specification is based upon a 512-byte write buffer operation.*
4. *In the pre-programming step of the Embedded Erase algorithm, all words are programmed to 0000h before Sector and Chip erasure.*
5. *System-level overhead is the time required to execute the bus-cycle sequence for the program command. See Table , Command Definitions on page 51 for further information on command definitions.*

9. Data Integrity

9.1 Endurance

Program / Erase Endurance

Non-Volatile Unit	Temperature Range	Minimum	Unit
Any Sector	Industrial	100K	Program-Erase cycles
	Industrial Plus	100K	
	Extended	10K	
Configuration Register	Industrial	100K	
	Industrial Plus	100K	
	Extended	10K	

Note:

1. Cycling data collection was limited to 100K cycles.

9.2 Data Retention

Data Retention

Parameter	Typical	Unit
Data Retention Time after 1K cycles or less with one programming operation, per half-page, per erase	20	Years

9.3 Device ID and Common Flash Interface (ID-CFI) ASO Map

The Device ID portion of the ASO (word locations 0h to 0Fh) provides manufacturer ID, device ID and basic feature set information for the device.

For additional information see *ID-CFI ASO* on page 34.

ID (Autoselect) Address Map

Word Address	Data	Description
(SA) + 0000h	0001h	Manufacture ID
(SA) + 0001h	007Eh	Device ID
(SA) + 0002h	Reserved	RFU
(SA) + 0003h	Reserved	
(SA) + 0004h	Reserved	
(SA) + 0005h	Reserved	
(SA) + 0006h	Reserved	
(SA) + 0007h	Reserved	
(SA) + 0008h	Reserved	
(SA) + 0009h	Reserved	
(SA) + 000Ah	Reserved	
(SA) + 000Bh	Reserved	
(SA) + 000Ch	0005h	Lower Software Bits Bit 0 - Status Register Support 1 = Status Register Supported 0 = Status Register Not Supported Bit 1 - DQ Polling Support 1 = DQ Bits Polling Supported 0 = DQ Bits Polling Not Supported Bit 3-2 - Command Set Support 11 = Reserved 10 = Reserved 01 = HyperFlash Command Set 00 = Classic Command Set Bits 4-F – Reserved = 0
(SA) + 000Dh	Reserved	Upper Software Bits
(SA) + 000Eh	0070h = 512 Mb @ 1.8V 006Fh = 512 Mb @ 3.0V 0072h = 256 Mb @ 1.8V 0071h = 256 Mb @ 3.0V 0074h = 128 Mb @ 1.8V 0073h = 128 Mb @ 3.0V	Device ID
(SA) + 000Fh	0000h	Device ID

CFI Query Identification String

Word Address	Data	Description
(SA) + 0010h (SA) + 0011h (SA) + 0012h	0051h 0052h 0059h	Query Unique ASCII string “QRY”
(SA) + 0013h (SA) + 0014h	0002h 0000h	Primary OEM Command Set
(SA) + 0015h (SA) + 0016h	0040h 0000h	Address for Primary Extended Table

CFI Query Identification String (Continued)

Word Address	Data	Description
(SA) + 0017h (SA) + 0018h	0000h 0000h	Alternate OEM Command Set (00h = none exists)
(SA) + 0019h (SA) + 001Ah	0000h 0000h	Address for Alternate OEM Extended Table (00h = none exists)

CFI System Interface String

Word Address	Data	Description
(SA) + 001Bh	0017h for V_{CC} = 1.8V 0027h for V_{CC} = 3.0V	V_{CC} Min. (erase / program) (D7-D4: volts, D3-D0: 100 millivolts)
(SA) + 001Ch	0019h for V_{CC} = 1.8V 0036h for V_{CC} = 3.0V	V_{CC} Max. (erase / program) (D7-D4: volts, D3-D0: 100 millivolts)
(SA) + 001Dh	0000h	V_{PP} Min. voltage (00h = no V_{PP} pin present)
(SA) + 001Eh	0000h	V_{PP} Max. voltage (00h = no V_{PP} pin present)
(SA) + 001Fh	0009h	Typical timeout per single word write 2^N µs
(SA) + 0020h	0009h	Typical timeout for max multi-byte program, 2^N µs (00h = not supported)
(SA) + 0021h	000Ah	Typical timeout per individual block erase 2^N ms
(SA) + 0022h	0012h (512 Mb) 0011h (256 Mb) 0010h (128 Mb)	Typical timeout for full chip erase 2^N ms (00h = not supported)
(SA) + 0023h	0002h	Max. timeout for single word write 2^N times typical
(SA) + 0024h	0002h	Max. timeout for buffer write 2^N times typical
(SA) + 0025h	0002h	Max. timeout per individual block erase 2^N times typical
(SA) + 0026h	0002h	Max. timeout for full chip erase 2^N times typical (00h = not supported)

CFI Device Geometry Definition

Word Address	Data	Description
(SA) + 0027h	001Ah (512 Mb) 0019h (256 Mb) 0018h (128 Mb)	Device Size = 2^N byte;
(SA) + 0028h	0000h	Flash Device Interface Description 0 = x8-only, 1 = x16-only, 2 = x8 / x16 capable
(SA) + 0029h	0000h	
(SA) + 002Ah	0009h	Max. number of byte in multi-byte write = 2^N (00 = not supported)
(SA) + 002Bh	0000h	
(SA) + 002Ch	0001h	Number of Erase Block Regions within device 1 = Uniform Device, 2 = Boot Device
(SA) + 002Dh	See Description	Erase Block Region 1 Information (refer to JEDEC JESD68-01 or JEP137 specifications) 00FFh, 0000h, 0000h, 0004h = 512 Mb (256 x 2-Mb blocks) 007Fh, 0000h, 0000h, 0004h = 256 Mb (128 x 2-Mb blocks) 003Fh, 0000h, 0000h, 0004h = 128 Mb (64 x 2-Mb blocks)
(SA) + 002Eh		
(SA) + 002Fh		
(SA) + 0030h		

CFI Device Geometry Definition (Continued)

Word Address	Data	Description
(SA) + 0031h	0000h	Erase Block Region 2 Information (refer to JEDEC JESD68-01 or JEP137 specifications)
(SA) + 0032h	0000h	
(SA) + 0033h	0000h	
(SA) + 0034h	0000h	
(SA) + 0035h	0000h	Erase Block Region 3 Information (refer to JEDEC JESD68-01 or JEP137 specifications)
(SA) + 0036h	0000h	
(SA) + 0037h	0000h	
(SA) + 0038h	0000h	
(SA) + 0039h	0000h	Erase Block Region 4 Information (refer to JEDEC JESD68-01 or JEP137 specifications)
(SA) + 003Ah	0000h	
(SA) + 003Bh	0000h	
(SA) + 003Ch	0000h	

CFI Primary Vendor-Specific Extended Query

Word Address	Data	Description
(SA) + 0040h	0050h	Query-unique ASCII string “PRI”
(SA) + 0041h	0052h	
(SA) + 0042h	0049h	
(SA) + 0043h	0031h	Major version number, ASCII
(SA) + 0044h	0035h	Minor version number, ASCII
(SA) + 0045h	001Ch	Address Sensitive Unlock (Bits 1-0) 00b = Required, 01b = Not Required Process Technology (Bits 5-2) 0000b = 0.23 µm Floating Gate 0001b = 0.17 µm Floating Gate 0010b = 0.23 µm MirrorBit 0011b = 0.13 µm Floating Gate 0100b = 0.11 µm MirrorBit 0101b = 0.09 µm Floating Gate 0110b = 0.09 µm MirrorBit 0111b = 0.065 µm MirrorBit Eclipse 1000b = 0.065 µm MirrorBit 1001b = 0.045 µm MirrorBit
(SA) + 0046h	0002h	Erase Suspend 0 = Not Supported 1 = Read Only 2 = Read and Write
(SA) + 0047h	0001h	Sector Protect 00 = Not Supported X = Number of sectors in smallest group
(SA) + 0048h	0000h	Temporary Sector Unprotect 00 = Not Supported 01 = Supported
(SA) + 0049h	0008h	Sector Protect / Unprotect Scheme 04 = High Voltage Method 05 = Software Command Locking Method 08 = Advanced Sector Protection Method
(SA) + 004Ah	0000h	Simultaneous Operation 00 = Not Supported X = Number of banks

CFI Primary Vendor-Specific Extended Query (Continued)

Word Address	Data	Description
(SA) + 004Bh	0001h	Burst Mode Type 00 = Not Supported 01 = Supported
(SA) + 004Ch	0000h	Page Read Mode Type 00 = Not Supported 01 = 4 Word Page 02 = 8 Word Page 03 =16 Word Page
(SA) + 004Dh	0000h	ACC (Acceleration) Supply Minimum 00 = Not Supported D7-D4: Volt D3-D0: 100 mV
(SA) + 004Eh	0000h	ACC (Acceleration) Supply Maximum 00 = Not Supported D7-D4: Volt D3-D0: 100 mV
(SA) + 004Fh	0000h	WP# Protection 00h = Flash device without WP Protect (No Boot) 01h = Eight 8-kB Sectors at TOP and Bottom with WP (Dual Boot) 02h = Bottom Boot Device with WP Protect (Bottom Boot) 03h = Top Boot Device with WP Protect (Top Boot) 04h = Uniform, Bottom WP Protect (Uniform Bottom Boot) 05h = Uniform, Top WP Protect (Uniform Top Boot) 06h = WP Protect for all sectors 07h = Uniform, Top or Bottom WP Protect
(SA) + 0050h	0001h	Program Suspend 00 = Not Supported 01 = Supported
(SA) +0051h	0000h	Unlock Bypass 00 = Not Supported 01 = Supported
(SA) + 0052h	000Ah	Secure Silicon Sector (Customer OTP Area = 1024B) Size 2^N (bytes)
(SA) + 0053h	008Dh	Software Features Bit 0: Status Register polling (1 = Supported, 0 = Not Supported) Bit 1: DQ polling (1 = Supported, 0 = Not Supported) Bit 2: New Program Suspend / Resume commands (1 = Supported, 0 = Not Supported) Bit 3: Word Programming (1 = Supported, 0 = Not Supported) Bit 4: Bit-Field Programming (1 = Supported, 0 = Not Supported) Bit 5: Autodetect Programming (1 = Supported, 0 = Not Supported) Bit 6: RFU Bit 7: Multiple Writes per Line (1 = Supported, 0 = Not Supported)
(SA) + 0054h	0005h	Page Size = 2^N bytes
(SA) + 0055h	0006h	Erase Suspend Timeout Maximum < 2^N (µs)
(SA) + 0056h	0006h	Program Suspend Timeout Maximum < 2^N (µs)
(SA) + 0057h to (SA) + 0077h	FFFFh	Reserved for Future Use
(SA) + 0078h	0006h	Embedded Hardware Reset Timeout Maximum < 2^N (µs) Reset with Reset Pin
(SA) + 0079h	0009h	Non-Embedded Hardware Reset Timeout Maximum < 2^N (µs) Power-On Reset

Hardware Interface

For the general description of the HyperBus hardware interface of HyperFlash memories refer to the HyperBus Specification. The following section describes HyperFlash device dependent aspects of hardware interface.

10. Interface States

10.1 Hardware Reset

- Terminates any operation in progress
- DQ[7:0] are placed into the High-Z state when RESET# is Low
- Exits any ASO
- Tristates all outputs
- Resets the Status Register
- Resets the EAC to Standby state
- CS# is ignored for the duration of the reset operation (t_{RPH})
- To meet the Reset current specification (I_{CC5}) CS# must be held High

To ensure data integrity any non-volatile operation that was interrupted should be reinitiated once the device completes the Hardware Reset process.

10.2 Power-Off with Hardware Data Protection

The memory is considered to be powered off when the core power supply (V_{CC}) drops below the lock-out voltage (V_{LKO}). When V_{CC} is below V_{LKO}, the entire memory array is protected against a program or erase operation. This ensures that no spurious alteration of the memory content can occur during power transition. During a power supply transition down to Power-Off, $V_{CC}Q$ should remain less than or equal to V_{CC}.

If V_{CC} goes below V_{RST} (Min) then returns above V_{RST} (Min) to V_{CC} minimum, the Power-On Reset interface state is entered and the EAC starts the Cold Reset Embedded Algorithm.

10.3 Power Conservation Modes

10.3.1 Active Clock Stop

The Active Clock Stop Mode reduces device interface energy consumption to the I_{CC6} level during the data transfer portion of a read operation. The device automatically enables this mode when CK/CK# remain stable for t_{ACC} + 30 ns while data is being output during a read. While in Active Clock Stop Mode, output data is latched and always driven onto the data bus. I_{CC6} in 11.2, *DC Characteristics (CMOS Compatible)* on page 63 represents the Active Clock Stop Mode current specification.

Active Clock Stop Mode helps reduce current consumption when the host system clock has stopped to pause the read data transfer. Even though CS# may be Low throughout these extended data transfer cycles, the memory device host interface will go into the Active Clock Stop current at t_{ACC} + 30 ns. This allows the device to transition into a lower current mode if the read data transfer is stalled. Active read current will resume once the data transfer is restarted with a toggling clock.

However, the EAC operates independent of the Active Clock Stop Mode of the host interface and will continue to draw current during an active Embedded Algorithm. Only when both the host interface and EAC are in their Standby states is the Active Clock Stop current level achieved.

10.3.2 Deep Power-Down

In the Deep Power-Down (DPD) Mode current consumption is driven to the lowest level. The DPD Mode must be entered while the device is in the Idle state while not in an ASO. DPD can also be entered when the device is in either the erase suspend or program suspend states. DPD is entered using the DPD Entry command sequence (See Table , *Command Definitions* on page 51).

Exiting the DPD Mode is accomplished with the assertion of CS# during any read or write transaction (CS# Low for t_{DPDCSL}). During the t_{DPDOUT} period the device will ignore command sequences (read and write transactions will not be processed) and RWDS will not toggle during an attempted read transaction. A 'dummy' write transaction is the preferred method for exiting DPD mode.

Driving the RESET# input Low (for the minimum t_{RP} time) will also cause the device to exit the DPD Mode. The device will take t_{DPDOUT} to return to the Idle state.

Upon exit from DPD mode the device will have the same default settings that exist after Power-On Reset.

DPD Mode Entry and Exit Timing

Symbol	Parameter	Min	Max	Unit
t_{DPDIN}	Deep Power-Down CR[15]=0 register write to DPD power level	10	—	µs
t_{DPDCSL}	Length of CS# Low period to cause an exit from Deep Power-Down	25	—	ns
t_{DPDOUT}	Deep Power-Down to Idle wakeup time	—	300	µs

Figure 10.1 Deep Power Down Entry Timing

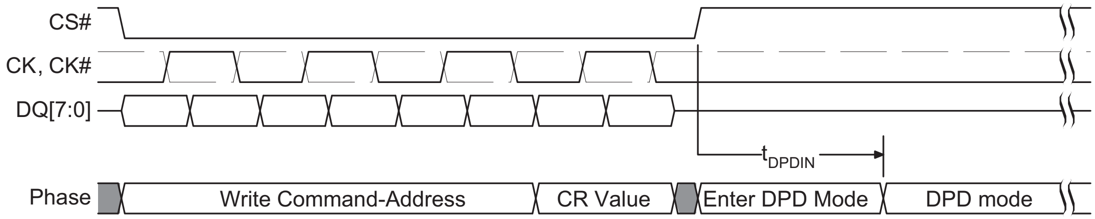


Figure 10.2 Deep Power Down CS# Exit Timing

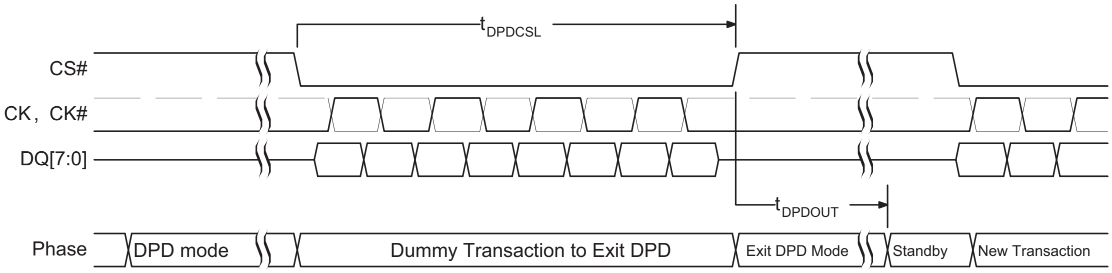
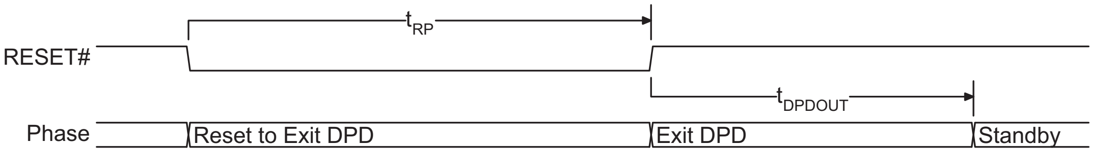


Figure 10.3 Deep Power Down RESET# Exit Timing



11. Electrical Specifications

The following section describes HyperFlash device dependent aspects of electrical specifications.

11.1 Absolute Maximum Ratings

Ambient Temperature with Power Applied–65°C to +125°C

11.2 DC Characteristics (CMOS Compatible)

DC Characteristics (CMOS Compatible)

Parameter	Description	Test Conditions	Min	Typ (9)	Max	Unit
I_{CC1}	V_{CC} Active Read Current (core current only, IO switching current is not included)	CS# = V_{IL}, @166 MHz, V_{CC} = 1.95V		130	180	mA
		CS# = V_{IL}, @100 MHz, V_{CC} = 3.6V		80	100	mA
I_{IO1}	V_{CC}Q Active Read Current of IOs	CS# = V_{IL}, @166 MHz, V_{CC}Q = 1.95V, C_{LOAD} = 20 pf		80	100	mA
		CS# = V_{IL}, @100 MHz, V_{CC}Q = 3.6V, C_{LOAD} = 20 pf		80	100	mA
I_{CC3P}	V_{CC} Active Program Current (1), (2)	V_{CC} = V_{CC} max		60	100	mA
I_{CC3E}	V_{CC} Active Erase Current (1), (2)	V_{CC} = V_{CC} max		60	100	mA
I_{CC4I}	V_{CC} Standby Current for Industrial Temp. (–40°C to +85°C)	CS#, RESET# = V_{CC}, V_{CC} = V_{CC} max		25	100	µA
I_{CC4IC}	V_{CC} Standby Current for Industrial Plus (Automotive - In Cabin) Temp. (–40°C to +105°C)	CS#, RESET# = V_{CC}, V_{CC} = V_{CC} max		25	300	µA
I_{CC4E}	V_{CC} Standby Current for Extended Temp. (–40°C to +125°C)	CS#, RESET# = V_{CC}, V_{CC} = V_{CC} max		25	300	µA
I_{CC5}	V_{CC} Reset Current (5)	CS# = $V_{IH,}$ RESET# = V_{SS}, V_{CC} = V_{CC} max		10	20	mA
I_{CC6}	Active Clock Stop Mode (3)	V_{IH}= V_{CC}, V_{IL} = V_{SS}, V_{CC} = 1.95V		6	12	mA
		V_{IH}= V_{CC}, V_{IL} = V_{SS}, V_{CC} = 3.6V		6	12	mA
I_{CC7}	V_{CC} Current during Power-Up (4)	CS# = X, V_{CC} = V_{CC} max,		80	100	mA
I_{DPD}	Deep Power-Down Current 512 Mb @ 25 °C	CS#, RESET#, V_{CC} = V_{CC} max		8	18	µA
	Deep Power-Down Current 512 Mb @ 85 °C			30	50	µA
	Deep Power-Down Current 512 Mb @ 105 °C			95	150	µA
	Deep Power-Down Current 512 Mb @ 125 °C			150	250	µA
	Deep Power-Down Current (all other densities) @ 25 °C			3	6	µA
	Deep Power-Down Current (all other densities) @ 85 °C			4	10	µA
	Deep Power-Down Current (all other densities) @ 105 °C			5	15	µA
	Deep Power-Down Current 256 Mb @ 125 °C			15	25	µA
	Deep Power-Down Current 128 Mb @ 125 °C			10	15	µA
V_{OL}	Output Low Voltage	I_{OL} = 100 µA for DQ7-DQ0 I_{OL} = 2 mA for INT# and RSTO#			0.15 x V_{CC}Q	V

Notes:

1. *I_{CC} active while Embedded Algorithm is in progress.*
2. *Not 100% tested.*
3. *Active Clock Stop Mode enables the lower power mode when the CK/CK# signals remain stable for t_{ACC} + 30 ns.*
4. *V_{CC}Q = 1.70V to 1.95V or 2.7V to 3.6V.*
5. *V_{CC} = V_{CC}Q = 1.8V or V_{CC} = V_{CC}Q = 3.0V.*
6. *During Power-Up there are spikes of current demand, the system needs to be able to supply this current to insure the part initializes correctly.*
7. *If an embedded operation is in progress at the start of reset, the current consumption will remain at the embedded operation specification until the embedded operation is stopped by the reset. If no embedded operation is in progress when reset is started, or following the stopping of an embedded operation, I_{CC7} will be drawn during the remainder of t_{RPH}. After the end of t_{RPH} the device will go to Standby Mode until the next read or write.*
8. *The recommended pull-up resistor for the INT# and RSTO# outputs is 5k to 10k Ohms.*
9. *Typical I_{CC} values are measured at t_{AI} = 25°C and V_{CC} = V_{CC}Q = 1.8V or 3.0V (not applicable to I_{DPD} for 85°C and 105°C).*

11.2.1 Capacitance Characteristics

1.8V Capacitive Characteristics

Description	Parameter	Min	Max	Unit
Input Capacitance (CK, CK#, CS#)	CI	1.5	3.0	pF
Delta Input Capacitance (CK, CK#)	CID		0.25	pF
Output Capacitance (RWDS)	CO	1.5	4.5	pF
I/O Pin Capacitance (DQx)	CIO	1.5	4.5	pF
I/O Pin Capacitance Delta (DQx)	CIOD	–	0.5	pF
INT#, RSTO# Pin Capacitance	COP	–	8.0	pF

Notes:

1. *These values are guaranteed by design and are tested on a sample basis only.*
2. *Pin capacitance is measured according to JEP147 procedure for measuring capacitance using a vector network analyzer. $V_{CC,}$ V_{CC}Q are applied and all other pins (except the pin under test) floating. DQs should be in the High Impedance state.*
3. *The capacitance values for the CK, CK#, RWDS and DQx pins must have similar capacitance values to allow for signal propagation time matching in the system. The capacitance value for CS# is not as critical because there are no critical timings between CS# going active (Low) and data being presented on the DQs bus.*

3.0V Capacitive Characteristics

Description	Parameter	Min	Max	Unit
Input Capacitance (CK, CS#)	CI	1.5	3.0	pF
Output Capacitance (RWDS)	CO	1.5	6.5	pF
I/O Pin Capacitance (DQx)	CIO	1.5	6.5	pF
I/O Pin Capacitance Delta (DQx)	CIOD	–	0.5	pF
INT#, RSTO# Pin Capacitance	COP	–	8.0	pF

Notes:

1. *These values are guaranteed by design and are tested on a sample basis only.*
2. *Pin capacitance is measured according to JEP147 procedure for measuring capacitance using a vector network analyzer. $V_{CC,}$ V_{CC}Q are applied and all other pins (except the pin under test) floating. DQs should be in the High Impedance state.*
3. *The capacitance values for the CK, RWDS and DQx pins must have similar capacitance values to allow for signal propagation time matching in the system. The capacitance value for CS# is not as critical because there are no critical timings between CS# going active (Low) and data being presented on the DQs bus.*

11.3 Power-Up and Power-Down

The memory is considered to be powered-off when the core power supply (V_{CC}) drops below the V_{CC} lock-out voltage (V_{LKO}). When V_{CC} is below V_{LKO}, the entire memory array is protected against a program or erase operation. This ensures that no spurious alteration of the memory content can occur during power transition. During a power supply transition down to the V_{SS} level, $V_{CC}Q$ should remain less than or equal to V_{CC}.

If V_{CC} goes below V_{CC} Reset (V_{RST}) then returns above V_{RST} to V_{CC} minimum, the Power-On Reset interface state is entered and the EAC starts the Cold Reset Embedded Algorithm.

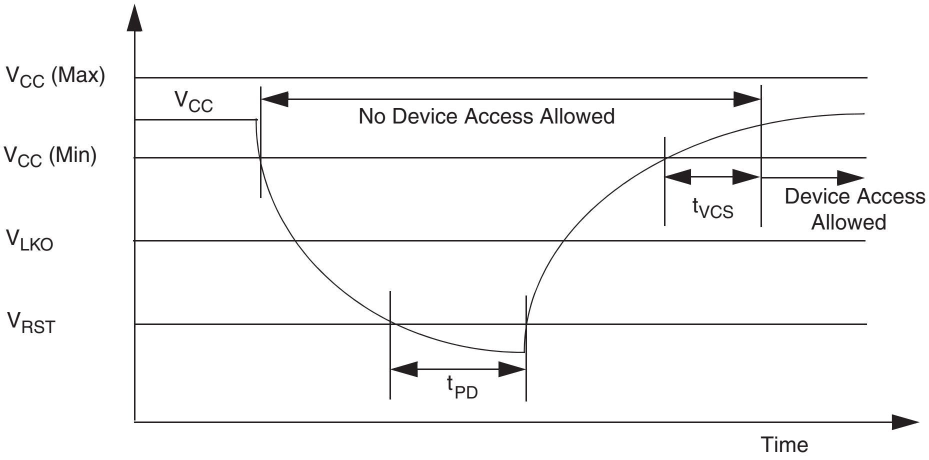
V_{CC} must always be greater than or equal to $V_{CC}Q$ ($V_{CC} \geq V_{CC}Q$).

The device ignores all inputs until a time delay of t_{VCS} has elapsed after the moment that V_{CC} and $V_{CC}Q$ both rise above, and stay above, the minimum V_{CC} thresholds. During t_{VCS} the device is performing Power-On Reset operations.

During Power-Down or voltage drops below V_{LKO}, the V_{CC} voltages must drop below V_{RST} for a period of t_{PD} for the part to initialize correctly when V_{CC} and $V_{CC}Q$ again rise to their operating ranges. See Figure 11.1, *Power-Down or Voltage Drop* on page 65. If during a voltage drop the V_{CC} stays above V_{LKO} the part will stay initialized and will work correctly when V_{CC} is again above V_{CC} minimum. If the part locks up from improper initialization, a Software Reset can be used to initialize the part correctly.

Normal precautions must be taken for supply decoupling to stabilize the V_{CC} and $V_{CC}Q$ power supplies. Each device in a system should have the V_{CC} and $V_{CC}Q$ power supplies decoupled by a suitable capacitor close to the package connections (this capacitor is generally on the order of 0.1 µF).

Figure 11.1 Power-Down or Voltage Drop



1.8V Power-Up / Power-Down Voltage and Timing

Symbol	Parameter	Min	Max	Unit
V_{CC}	V_{CC} Power Supply	1.7	1.95	V
V_{LKO}	V_{CC} Cut-0ff below which re-initialization is required	1.5	–	V
V_{RST}	V_{CC} Low Voltage needed to ensure initialization will occur	0.5	–	V
t_{VCS}	V_{CC} and $V_{CC}Q \geq$ minimum to first access RESET# Low to High transition to first access (V_{CC} and $V_{CC}Q \geq$ minimum)	–	300	µs
t_{PD}	Duration of $V_{CC} \leq V_{RST}$	10	–	µs

Note:

1. V_{CC} ramp rate can be non-linear.

3.0V Power-Up / Power-Down Voltage and Timing

Symbol	Parameter	Min	Max	Unit
V_{CC}	V_{CC} Power Supply	2.7	3.6	V
V_{LKO}	V_{CC} Cut-0ff below which re-initialization is required	2.4	–	V
V_{RST}	V_{CC} Low Voltage needed to ensure initialization will occur	0.7	–	V
t_{VCS}	V_{CC} and $V_{CC}Q \geq$ minimum to first access RESET# Low to High transition to first access (V_{CC} and $V_{CC}Q \geq$ minimum)	–	300	µs
t_{PD}	Duration of $V_{CC} \leq V_{RST}$	10	–	µs

Note:
1. *V_{CC} ramp rate can be non-linear.*

11.3.1 Power-On (Cold) Reset (POR)

When power is first applied, with supply voltage below V_{LKO} then rising to reach operating range minimum, internal device configuration and Cold Reset activities are initiated. RESET# and CS# are ignored during the duration of the POR operation (t_{VCS}) and any user extended period of RSTO# Low. Command sequences are blocked while the device is in the POR state or RSTO# is Low. During this period, the device can not be selected, will not accept commands, and does not drive outputs other than RSTO#. RESET# Low during this POR period is optional. If RESET# is driven Low during POR it must satisfy the Hardware Reset parameters t_{RP} and t_{RPH} in which case the POR operations will be completed at the end of t_{VCS} and t_{RPH}. If RESET# is Low during t_{VCS} it may remain Low at the end of t_{VCS} to hold the device in the Hardware Reset state. If RESET# is High at the end of t_{VCS} the device will go to the Standby state. CS# must go to V_{IH} before the end of RSTO# Low.

During Cold Reset the device will draw I_{CC7} current. If CS# is Low during t_{VCS} the device may draw higher than typical POR current during t_{VCS} but will not exceed the maximum, and the level of CS# will not affect the Cold Reset EA.

If POR has not been properly completed by the end of t_{VCS}, a later transition to the Hardware Reset state will cause a transition to the Power-On Reset interface state and initiate the Cold Reset Embedded Algorithm. This ensures the device can complete a Cold Reset even if some aspect of the system Power-On voltage ramp-up causes the POR to not initiate or complete correctly.

RSTO# is an open-drain output used to indicate when a POR is occurring within the device and can be used as a system level reset signal. Upon completion of the internal POR the RSTO# signal will transition from Low to high impedance after a user defined timeout period has elapsed. Upon transition to the high impedance state the external pull-up resistance will pull RSTO# High and the device immediately is placed into the Idle state. While RSTO# is low, no commands are accepted by the device.

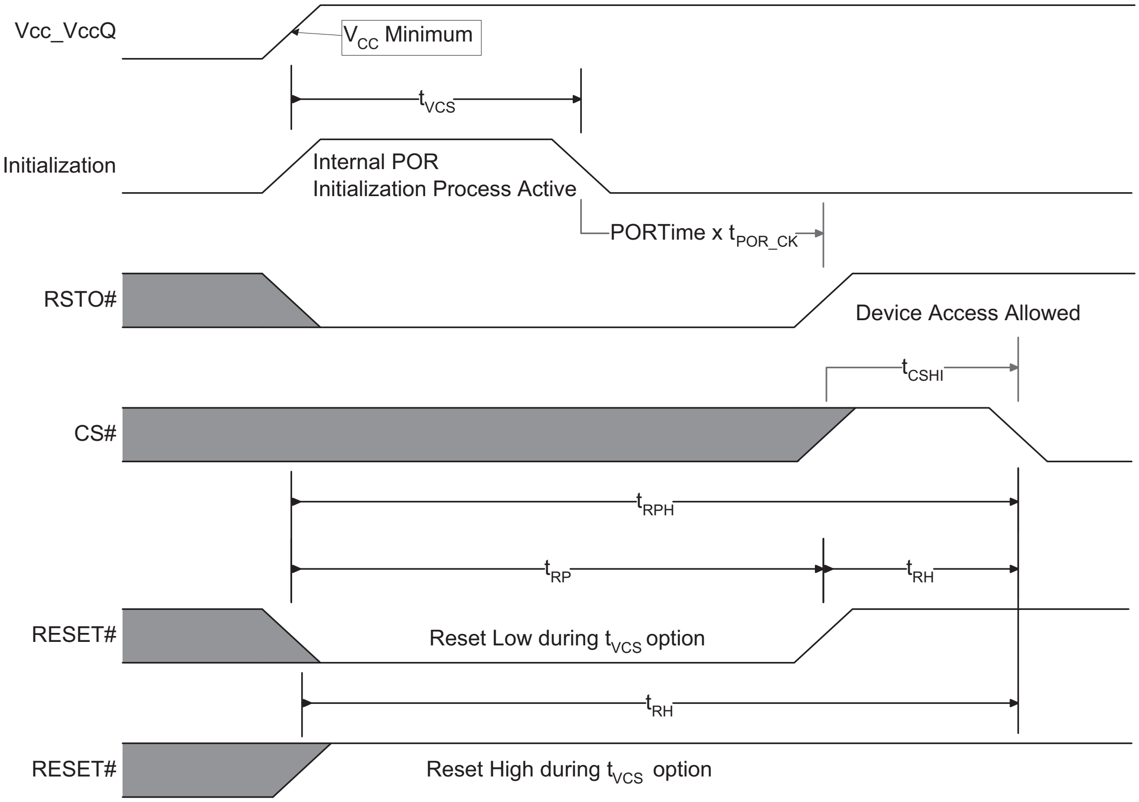
If the user wants to extend the RSTO# period beyond the POR (t_{VCS}) period the non-volatile PORTime Register must be programmed. The default value for this register (FFFFh), provides zero added time. The RSTO# signal will return to high impedance at the end of t_{VCS}. A value programmed into the 16-bit PORTime Register is multiplied by t_{POR_CK} (see Table , *User POR Extension Clock Timings* on page 66) to define the length of extension to the RSTO# pulse beyond t_{VCS}. The length of the programmed extension to the RSTO# assertion is the value programmed into the PORTime Register plus one clock cycle. The PORTime Register is OTP and, once programmed, will fail subsequent programming attempts.

User POR Extension Clock Timings

Parameter	Symbol	Min	Max	Unit
POR Extension Clock Period	t_{POR_CK}	25	42	µs

Note that both the RSTO# and INT# outputs are undefined while V_{CC} is below V_{CC}(min). By the time that V_{CC}(min) is reached, the INT# output will be in the high impedance state. The RSTO# output will transition from the Low to high impedance state after t_{VCS}, plus any additional user defined POR extension time, after V_{CC}(min) has been reached.

Figure 11.2 Power On Reset Signal Diagram



Notes:

1. *$V_{CC}Q$ must be the same voltage as V_{CC}.*
2. *PORTime is a customer programmed configuration register intended to allow RSTO# assertion beyond t_{VCS}. PORTime is described in Table , Non-Volatile Configuration Registers on page 50.*
3. *3. t_{POR_CK} is the internal (on-chip) clock period used to generate the extension to RSTO#. t_{POR_CK} is described in Table , User POR Extension Clock Timings on page 66.*

11.3.2 Hardware (Warm) Reset

The RESET# input provides a hardware method of resetting the device to the Standby state. While RESET# is Low command sequences and read operations are not allowed. Command sequences are blocked while the device is in the reset state.

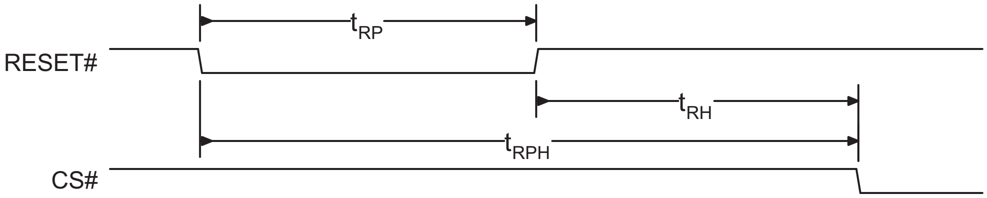
During Hardware Reset the device will draw I_{CC5} current. When RESET# continues to be held at V_{SS}, the device draws CMOS standby current (I_{CC4}). If RESET# is held at V_{IL}, but not at V_{SS}, the standby current is greater.

A Hardware Reset will cause the bus configuration to be defined by the Non-Volatile Configuration Register (NVCR). See Figure 11.3, *Hardware Reset Timing Diagram* on page 68.

After the device has completed POR and entered the Standby state, any later transition to the Hardware Reset state will initiate the Warm Reset Embedded Algorithm. A Warm Reset is much shorter than a Cold Reset, taking tens of µs (t_{RPH}) to complete. During the Warm Reset EA, any in progress Embedded Algorithm is stopped and the EAC is returned to its POR state without reloading EAC algorithms from non-volatile memory. After the Warm Reset EA is completed, the interface will remain in the Hardware Reset state if RESET# remains Low. When RESET# returns High the interface will transition to the Standby state. If RESET# is High at the end of the Warm Reset EA, the interface will directly transition to the Standby state.

If POR has not been properly completed by the end of t_{VCS}, a later transition to the Hardware Reset state will cause a transition to the Power-On Reset interface state and initiate the Cold Reset Embedded Algorithm. This ensures the device can complete a Cold Reset even if some aspect of the system Power-On voltage ramp-up causes the POR to not initiate or complete correctly.

Figure 11.3 Hardware Reset Timing Diagram



Power-On and Reset Parameters

Parameter	Description	Limit	Time	Unit
t_{VCS}	V_{CC} Setup Time to first access (1)	Min	300	µs
t_{RPH}	RESET# Low to CS# Low	Min	30	µs
t_{RP}	RESET# Pulse Width	Min	200	ns
t_{RH}	Time between RESET# (High) and CS# (Low)	Min	150	ns
t_{PD}	Duration of $V_{CC} \leq V_{RST}$	Min	1	µs
t_{CSHI}	Chip Select High Between Operations	Min	6.0	ns

Notes:

1. *Bus transactions (read and write) are not allowed during the Power-Up Reset time (t_{VCS}).*
2. *Timing measured from V_{CC} reaching V_{CC} min to V_{IH} on Reset and V_{IL} on CS#.*
3. *RESET# Low is optional during POR. If RESET is asserted during POR, the later of t_{RPH} and t_{VCS} will determine when CS# may go Low. If RESET# remains Low after t_{VCS} is satisfied, t_{RPH} is measured from the end of t_{VCS}. RESET must also be High t_{RH} before CS# goes Low.*
4. *V_{CC} ramp rate can be non-linear.*
5. *Sum of t_{RP} + t_{RH} must be equal to or greater than t_{RPH}.*

Hardware Reset can also be used to exit from DPD mode. Driving the RESET# input Low (for the minimum t_{RP} time) will also cause the device to exit the DPD Mode. The device will take t_{DPDOUT} to return to the Idle state. Upon exit from DPD the device will have the same default settings that exist after Power-On Reset. See Section 10.3.2, *Deep Power-Down* on page 62.

12. Timing Specifications

For the description of HyperBus interface common timing specifications refer to the HyperBus Specification. The following section describes HyperFlash device dependent aspects of timing specifications.

12.1 AC Characteristics

HyperFlash Specific 1.8V Read Timings

Parameter	Symbol	166 MHz		Unit
		Min	Max	
Read Initial Access Time	t_{ACC}	—	96	ns
Chip Select Active to RWDS Valid (Low)	t_{DSV}	—	8	ns

Note:
1. Sampled, not 100% tested.

HyperFlash Specific 3.0V Read Timings

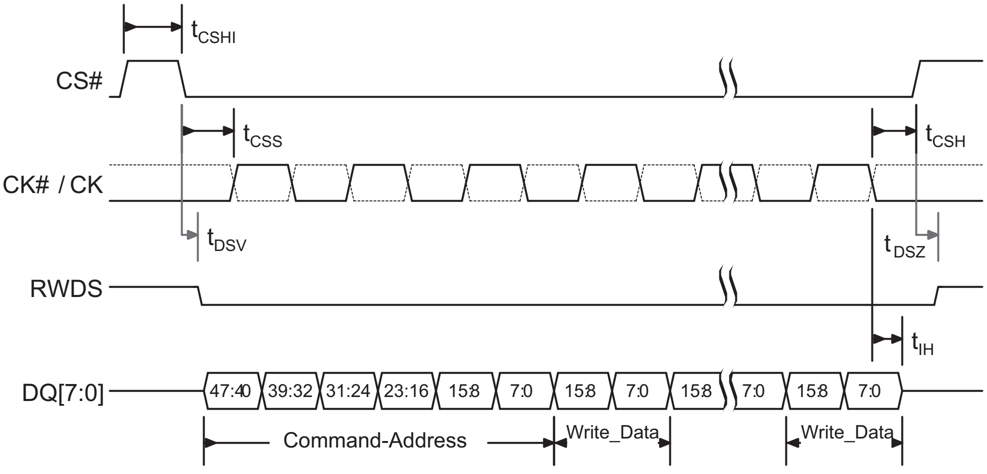
Parameter	Symbol	100 MHz		Unit
		Min	Max	
Read Initial Access Time	t_{ACC}	—	96	ns

Note:
1. Sampled, not 100% tested.

There are no HyperFlash specific write timings.

12.2 Word Programming with Multiple Word Burst Data Load

Figure 12.1 Burst Write During Load of Multiple Words During a Word Program Command Timing Diagram



Host drives DQ[7:0] with Command-Address and Write Data

Notes:
1. *Transactions must be initiated with CK = Low and CK# = High. CS# must return High before a new transaction is initiated.*
2. *HyperFlash memory drives RWDS Low during write while CS# is Low.*
3. *Burst Write operations are not allowed while in an ASO state.*
4. *Burst Write operations are only allowed while loading multiple words during a Word Program command.*
5. *Burst write operations are linear only, no wrapped burst write capability is supported.*
6. *CK# is only used on the 1.8V device. The 3V device uses a single ended clock input.*

Burst Write During Load of Multiple Words During a Word Program Command Timings

Parameter	Symbol	50 MHz (2)		Unit
		Min	Max	
Operating Frequency for Burst Write			50	MHz
Chip Select Setup to next CK Rising Edge	t_{CSS}	3	–	ns
Chip Select Active to RWDS Valid (Low)	t_{DSV}	–	8	ns
Input Setup	t_{IS}	1.0	–	ns
Input Hold	t_{IH}	1.0	–	ns
Chip Select Hold After CK Falling Edge	t_{CSH}	0	–	ns
Chip Select Inactive to RWDS High-Z	t_{DSZ}	–	6	ns
Chip Select High Between Operations	t_{CSHI}	10.0	–	ns

Notes:

1. *Sampled, not 100% tested.*
2. *50 MHz timings are only required when using a burst write during a Word Program command.*

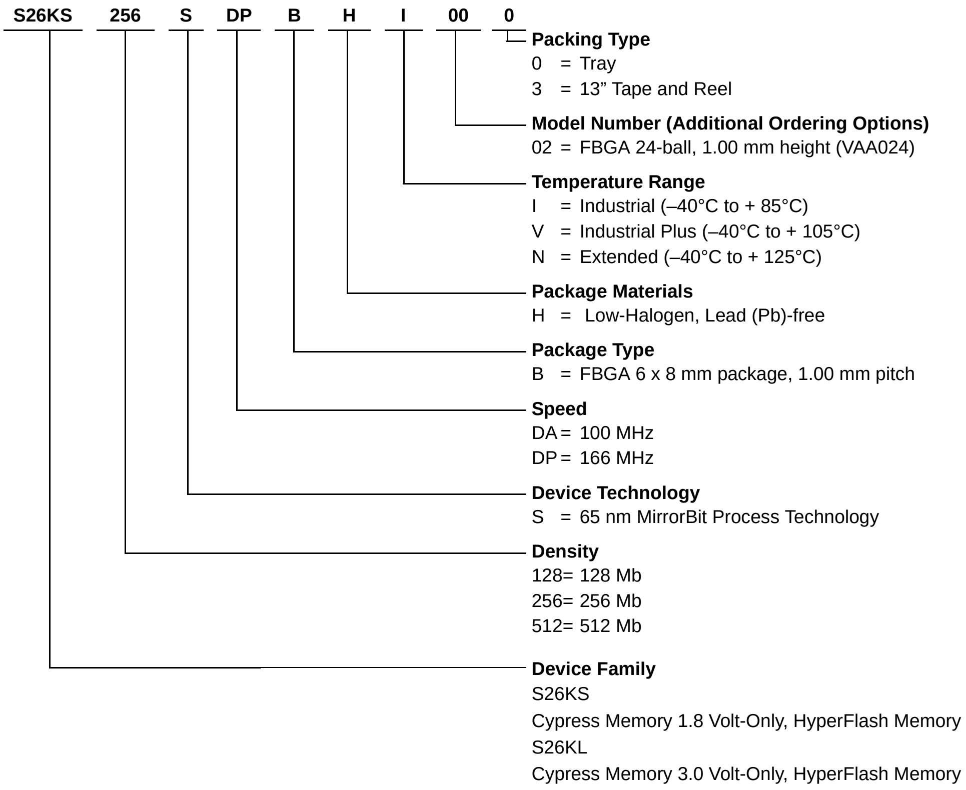
13. Physical Interface

See the HyperBus Specification for footprint and the 6 × 8 × 1 mm (VAA024) physical package diagram.

14. Ordering Information

14.1 Ordering Part Numbers

The ordering part number is formed by a valid combination of the following:



S26KS 256 S DP B H I 00 0

Packing Type
0 = Tray
3 = 13” Tape and Reel

Model Number (Additional Ordering Options)
02 = FBGA 24-ball, 1.00 mm height (VAA024)

Temperature Range
I = Industrial (–40°C to + 85°C)
V = Industrial Plus (–40°C to + 105°C)
N = Extended (–40°C to + 125°C)

Package Materials
H = Low-Halogen, Lead (Pb)-free

Package Type
B = FBGA 6 x 8 mm package, 1.00 mm pitch

Speed
DA = 100 MHz
DP = 166 MHz

Device Technology
S = 65 nm MirrorBit Process Technology

Density
128= 128 Mb
256= 256 Mb
512= 512 Mb

Device Family
S26KS
Cypress Memory 1.8 Volt-Only, HyperFlash Memory
S26KL
Cypress Memory 3.0 Volt-Only, HyperFlash Memory

14.2 Valid Combinations

The Valid Combinations table lists configurations planned to be available in volume. The table will be updated as new combinations are released. Consult your local sales representative to confirm availability of specific combinations and to check on newly released combinations.

Valid Combinations

Device Number	Speed Option	Package and Material	Temperature Range	Model Number	Packing Type	Ordering Part Number (x = Packing Type)	Package Marking
S26KL512S	DA	BH	I, V, N	02	0, 3	S26KL512SDABHI02x	6KL512SDAHI02
						S26KL512SDABHV02x	6KL512SDAHV02
						S26KL512SDABHN02x	6KL512SDAHN02
S26KL256S	DA	BH	I, V, N	02	0, 3	S26KL256SDABHI02x	6KL256SDAHI02
						S26KL256SDABHV02x	6KL256SDAHV02
						S26KL256SDABHN02x	6KL256SDAHN02
S26KL128S	DA	BH	I, V, N	02	0, 3	S26KL128SDABHI02x	6KL128SDAHI02
						S26KL128SDABHV02x	6KL128SDAHV02
						S26KL128SDABHN02x	6KL128SDAHN02
S26KS512S	DP	BH	I, V, N	02	0, 3	S26KS512SDPBHI02x	6KS512SDPHI02
						S26KS512SDPBHV02x	6KS512SDPHV02
						S26KS512SDPBHN02x	6KS512SDPHN02
S26KS256S	DP	BH	I, V, N	02	0, 3	S26KS256SDPBHI02x	6KS256SDPHI02
						S26KS256SDPBHV02x	6KS256SDPHV02
						S26KS256SDPBHN02x	6KS256SDPHN02
S26KS128S	DP	BH	I, V, N	02	0, 3	S26KS128SDPBHI02x	6KS128SDPHI02
						S26KS128SDPBHV02x	6KS128SDPHV02
						S26KS128SDPBHN02x	6KS128SDPHN02

Note:

1. FBGA package marking omits the leading S2, the package type character, and packing type character from the ordering part number.

15. Document History Page

Document Title: S26KL512S / S26KS512S / S26KL256S / S26KS256S / S26KL128S / S26KS128S, 512 Mbit (64 Mbyte), 256 Mbit (32 Mbyte), 128 Mbit (16 Mbyte) 1.8V/3.0V HyperFlash™ Family
Document Number: 001-99198

Rev.	ECN No.	Orig. of Change	Submission Date	Description of Change
**	–	RYSU	04/23/2015	Initial release
*A	–	RYSU	06/12/2015	Global: Changed ‘continuous’ to ‘linear’ Distinctive Characteristics: Removed ‘Feature Variations by Density’ table Signal Descriptions: HyperFlash Interface figure: updated HyperBus Protocol: Updated section Embedded Operations: Deep Power-Down: updated section Data Integrity: Program / Erase Endurance table: added Extended temperature, updated Minimum and Typical values Data Retention table: Updated Parameter; added Note Hardware Interface: Updated section Electrical Specifications: Updated section Timing Specifications: Updated section Physical Interface: Updated section Physical Diagrams: FBGA 25-Ball, 6 x 8 x 1 mm, 5x5 Array Package Outline Drawing: updated figure Ordering Part Numbers: Updated Valid Combinations: Added ‘Package Marking’
*B	–	RYSU	07/16/2015	Distinctive Characteristics: Updated Endurance and Retention Embedded Algorithm Performance: Embedded Algorithm Characteristics table: removed Note 6 Data Integrity: Program / Erase Endurance table: updated Minimum values, removed Typical values Data Retention table: updated Parameter, removed Note Physical Interface: Updated section; Removed connection diagram and physical diagrams Ordering Information: Updated section Valid Combinations: Updated table
*C	4854296	MAMC	07/24/2015	Updated to Cypress template
*D	4936066	MAMC	09/29/2015	Changed status from Advance to Final. Template updates: Removed cover page and Spansion Revision History.
*E	5035685	RYSU / CRLE	12/14/2015	Updated *Electrical Specifications*: Updated *DC Characteristics (CMOS Compatible)*: Changed maximum value of I_{DPD} parameter corresponding to “512 Mb @ 25 °C” from 15 µA to 18 µA. Added values of I_{DPD} parameter corresponding to “512 Mb @ 125 °C”, “256 Mb @ 125 °C” and “128 Mb @ 125 °C”. Updated *Ordering Information*: Updated *Valid Combinations*: Added Extended Temperature part numbers.

Sales, Solutions, and Legal Information

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Automotive	cypress.com/go/automotive
Clocks & Buffers	cypress.com/go/clocks
Interface	cypress.com/go/interface
Lighting & Power Control	cypress.com/go/powerpsoc
Memory	cypress.com/go/memory
PSoC	cypress.com/go/psoc
Touch Sensing	cypress.com/go/touch
USB Controllers	cypress.com/go/USB
Wireless/RF	cypress.com/go/wireless

PSoC® Solutions

psoc.cypress.com/solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP

Cypress Developer Community

Community | Forums | Blogs | Video | Training

Technical Support

cypress.com/go/support

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

Cypress Semiconductor:

S26KS512SDABHV030 S26KS128SDPBHI020 S26KL256SDABHI020 S26KL128SDABHV020
S26KL256SDABHV020 S26KL256SDABHV030 S26KL128SDABHV030 S26KS128SDPBHV020
S26KL128SDABHI030 S26KL128SDABHI020 S26KL256SDABHI030 S26KS512SDGBHV030 S26KS512SDGBHI030
S26KL512SDABHV020 S26KL512SDABHI020 S26KS512SDPBHV020 S26KS512SDPBHI020
S26KL512SDABHI030 S26KS256SDPBHI020 S26KS256SDPBHV020 S26KS256SDGBHV030
S26KS128SDGBHI030 S26KS128SDGBHV030 S26KS256SDGBHI030 S26KL512SDABHV030