S26HS256T / S26HS512T / S26HS01GT
S26HL256T / S26HL512T / S26HL01GT

256-Mb (32-MB), 512-Mb (64-MB), 1-Gb (128-MB) HS-T (1.8-V), HL-T (3.0-V) Semper™ Flash with HyperBus™ Interface

Device Overview

Architecture

- Cypress 45-nm MirrorBit® technology that stores two data bits in each memory array cell
- Sector Architecture options
  - Uniform: Address space consists of all 256 KB Sectors
  - Hybrid
    - Configuration 1: Address space consists of thirty-two 4 KB sectors grouped either on the top or the bottom while the remaining sectors are all 256 KB
    - Configuration 2: Address space consists of thirty-two 4 KB sectors equally split between top and bottom while the remaining sectors are all 256 KB
- Page Programming buffer of 256 or 512 bytes
- OTP Secure Silicon Region (SSR) of 1024 bytes (32 × 32 bytes)

Interface

- HyperBus™ Interface
  - JEDEC eXpanded SPI (JESD251) compliant
  - DDR option runs up to 400 MBps (200 MHz clock speed)
  - Supports Data Strobe (DS) to simplify the read data capture in high-speed systems
- Legacy (x1) SPI (1S-1S-1S)
  - JEDEC eXpanded SPI (JESD251) compliant
  - SDR option runs up to 21 MBps (166 MHz clock speed)
- Semper Flash with HyperBus Interface devices support default boot in Legacy SPI (x1) or HyperBus Interface (x8)

Highlights

- Safety Features
  - Functional Safety with the Industry's first ISO26262 ASIL B compliant and ASIL D ready NOR flash
  - EnduraFlex™ Architecture provides High-Endurance and Long Retention Partitions
  - Interface CRC detects errors on communication interface between host controller and Semper™ Flash device
  - Data Integrity CRC detects errors in memory array
  - SafeBoot reports device initialization failures, detects configuration corruption, and provides recovery options
  - Built-in Error Correcting Code (ECC) corrects Single-bit Error and detects Double-bit Error (SECDED) on memory array data
  - Sector Erase Status indicator for power loss during erase
- Protection Features
  - Advanced Sector Protection for individual memory array sector based protection
- AutoBoot enables immediate access to the memory array following power-on
- Hardware Reset through CS# Signaling method (JEDEC) AND individual RESET# pin

Identification

- Serial Flash Discoverable Parameters (SFDP) describing device functions and features
- Device Identification, Manufacturer Identification and Unique Identification

Data Integrity

- 256 Mb Devices
  - Minimum 640,000 Program-Erase Cycles for the Main array
- 512 Mb Devices
  - Minimum 1,280,000 Program-Erase Cycles for the Main array
- 1 Gb Devices
  - Minimum 2,560,000 Program-Erase Cycles for the Main array
- All Devices
  - Minimum 300,000 Program-Erase Cycles for the 4 KB Sectors
  - Minimum 25 Years Data Retention

Supply Voltage

- 1.7-V to 2.0-V (HS-T)
- 2.7-V to 3.6-V (HL-T)

Grade / Temperature Range

- Industrial (–40 °C to +85 °C)
- Industrial Plus (–40 °C to +105 °C)
- Automotive AEC-Q100 Grade 3 (–40 °C to +85 °C)
- Automotive AEC-Q100 Grade 2 (–40 °C to +105 °C)
- Automotive AEC-Q100 Grade 1 (–40 °C to +125 °C)

Packages

- 256 Mb and 512 Mb:
  - 24-ball BGA 6 × 8 mm
- 1 Gb:
  - 24-ball BGA 8 × 8 mm

Performance Summary

Maximum Read Rates

Transaction	Initial Access Latency (Cycles)	Clock Rate (MHz)	MBps
SPI Read	0	50	6.25
SPI Fast Read	10	166	20.75
HyperBus Read DDR (HS-T)	16	200	400
HyperBus Read DDR (HL-T)	14	166	332

Typical Program and Erase Rates

Operation	KBps
256B Page Programming (4 KB Sector / 256 KB Sector)	595 / 533
512B Page Programming (4 KB Sector / 256 KB Sector)	753 / 898
256 KB Sector Erase	331
4 KB Sector Erase	95

Typical Current Consumption

Operation	HL-T Current (mA)	HS-T Current (mA)
SDR Read 50 MHz	10	10
DDR Read (HyperBus)	75 (166 MHz)	156 (200 MHz)
Program	50	50
Erase	50	50
Standby (HS-T)	0.014	0.011
Deep Power Down (HS-T)	0.0022	0.0013

Data Integrity

Program / Erase (PE) Endurance - High Endurance (256 KB Sectors)

Sectors in Partition	Minimum PE Cycles	Minimum Retention Time	Unit
512 (Default for 1 Gb devices)	2,560,000	2	Years
508	2,540,000		
504	2,520,000		
...	...		
256 (Default for 512 Mb devices)	1,280,000		
252	1,260,000		
128 (Default for 256 Mb devices)	640,000		
...			
28	140,000		
24	120,000		
20	100,000		

Note Minimum Cycles is for entire High Endurance Partition.

Program / Erase Endurance - Long Retention Partition (256 KB Sectors)

Minimum PE Cycles	Minimum Retention Time	Unit
500	25	Years

Note Minimum Cycles is for each sector.

Program / Erase Endurance 4 KB Sector and Nonvolatile Register Array

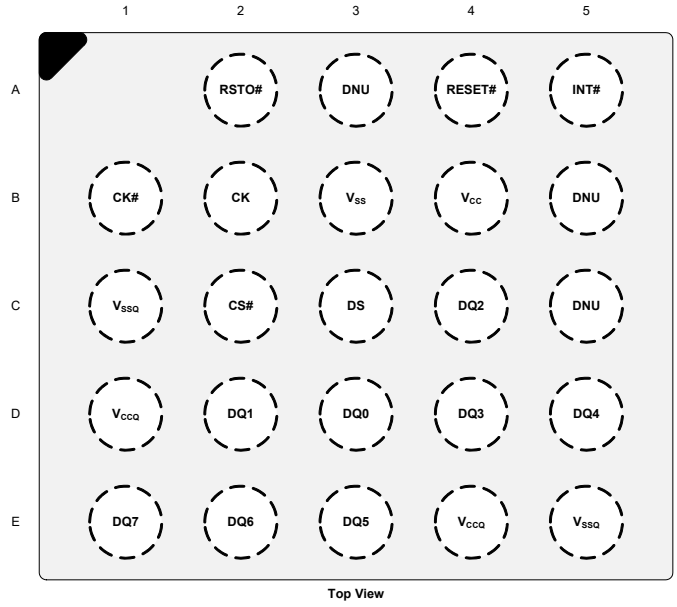
Flash Memory Type	Minimum Cycles	Unit	Minimum Retention Time	Unit
Program/Erase cycles per 4 KB sector	500	PE cycles	25	Years
	300,000 **Note** It is required to restrict the power loss events to 300 times per sector during program or erase operation to achieve the mentioned endurance cycles.		2	
Program/Erase cycles per Persistent Protection Bits (PPB) array or nonvolatile register array **Note** Each write transaction to a nonvolatile register causes a PE cycle on the entire nonvolatile register array.	500		25	

Contents

1. Pinout and Signal Description 5
2. Interface Overview 7
2.1 General Description 7
2.2 HyperBus Transaction Protocol 9
2.3 Legacy (x1) SPI Transaction Protocol 11
2.4 Embedded Operations 14
2.5 Register Naming Convention 15
2.6 HyperBus Transaction Naming Convention 15
2.7 Legacy (x1) SPI Transaction Naming Convention 16
3. Address Space Maps 17
3.1 Semper Flash Memory Array 17
3.2 Address Space Overlays (ASO) (HyperBus Only) 19
3.3 Device ID Address Space 20
3.4 HyperBus Unique ID 21
3.5 SFDP JEDEC JESD216 Serial Flash Discoverable Parameters (SFDP) Space 21
3.6 Secure SSR Address Space 34
3.7 Registers (Legacy (x1) SPI Only) 34
4. Features 36
4.1 Error Detection and Correction 36
4.2 EnduraFlex Architecture (Wear Leveling) 40
4.3 Interface CRC 44
4.4 Data Integrity CRC 46
4.5 Data Protection Schemes 47
4.6 SafeBoot 56
4.7 AutoBoot 61
4.8 Read Transactions 62
4.9 Write Transactions - Legacy (x1) SPI 70
4.10 Program 72
4.11 Erase 78
4.12 Suspend and Resume Embedded Operation 82
4.13 Error Types and Reporting - HyperBus 85
4.14 Error Typer and Reporting - Legacy (x1) SPI 86
4.15 Reset 87
4.16 Power Modes 90
4.17 Power Up and Power Down 91
5. Registers 93
5.1 Register Naming Convention 93
5.2 HyperBus Registers 94
5.3 Legacy (x1) SPI Registers 110
6. Transaction Table 126
6.1 HyperBus Transaction Table 126
6.2 Legacy (x1) SPI Transaction Table 140
7. Electrical Characteristics 143
7.1 Absolute Maximum Ratings[66] 143
7.2 Operating Range 143
7.3 Thermal Resistance 143
7.4 Capacitance Characteristics 143
7.5 Latchup Characteristics 143
7.6 DC Characteristics 144
7.7 AC Test Conditions 146
8. Timing Characteristics 147
8.1 Timing Waveforms 150
9. Package Diagrams 153
10. Ordering Information 155
10.1 Valid Combinations — Standard Grade 156
10.2 Valid Combinations — Automotive Grade / AEC-Q100 157
11. Document History Page 158
Sales, Solutions, and Legal Information 159
Worldwide Sales and Design Support 159
Products 159
PSoC® Solutions 159
Cypress Developer Community 159
Technical Support 159

1. Pinout and Signal Description

Figure 1. 24-Ball BGA Pinout Configuration[1]



Top View

Table 1. Signal Description

Symbol	Type	Mandatory / Optional	Description
CS#	Input	Mandatory	**Chip Select (CS#)**. All bus transactions are initiated with a HIGH to LOW transition on CS# and terminated with a LOW to HIGH transition on CS#. Driving CS# LOW enables the device, placing it in the Active mode. When CS# is driven HIGH, the device enters Standby mode, unless an internal embedded operation is in progress. All other input pins are ignored and the output pins are put in HIGH impedance state.
CK, CK#[2, 3]	Input	Mandatory	**Clock (CK, CK#)**. Clock provides the timing of the serial interface. Single ended and differential clock modes are offered. Transactions are latched either on the rising edge of CK signal (single ended) or on the crossing of the CK and CK# signals (differential). In Legacy (x1) SPI interface, command, address and data inputs are latched on rising edge of the clock, and data is output on the falling edge of the clock. In HyperBus (x8) interface, for single ended clock, command, address and data input are latched with respect to the rising and falling edge of the CK. In differential clock mode, command, address and data inputs are latched with respect to the crossing of CK and CK#. **Differential Clock.** CK and CK# are used. **Single Ended:** CK is used (CK# is not used and can be left floating).
DS	Output	Mandatory	**Read Data Strobe (DS)**. DS is used for data read operations only and indicates output data valid for HyperBus interface. During a read transaction while CS# is LOW, DS toggles to synchronize data output until CS# goes HIGH. Output data during read transactions are edge aligned with DS.
DQ[7:0]	Input/Output	Mandatory	**Serial Data (DQ[7:0])**. Bidirectional signals that transfer command, address and data information. **Legacy (x1) SPI Interface**: DQ[0] is an input (SI) and DQ[1] is an output (SO). **HyperBus (x8) Interface**: DQ[7:0] are input and output.
RESET#	Input (Weak Pull-up)	Optional	**Hardware Reset (RESET#)**. When Low, the device will self initialize and return to the array read state. DS and DQ[7:0] are placed into the High-Z state when RESET# is Low. RESET# includes a weak pull-up, meaning, if RESET# is left unconnected it will be pulled up to the High state.
INT#	Output (Open Drain)	Optional	**System Interrupt (INT#)**. 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.

Notes
1. Flash memory devices in BGA packages can be damaged if exposed to ultrasonic cleaning methods. The package, data integrity, or both may be compromised if the package body is exposed to temperatures above 150 °C for prolonged periods of time.
2. The clock is not required to be free running.
3. CK and CK# are not true differential signals. They are compliment signals. Care must be taken to ensure system level terminations are properly designed in.

Table 1. Signal Description (Continued)

Symbol	Type	Mandatory / Optional	Description
RSTO#	Output (Open Drain)	Optional	**Reset Output (RSTO#)**. 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 Standby state. Transactions are blocked when RSTO# is LOW. During this period, the device cannot be selected, will not accept any transactions, and does not drive outputs other than RSTO#.
V_{CC}	Power Supply	Mandatory	Core Power Supply
V_{CCQ}	Power Supply	Mandatory	Input / Output Power Supply
V_{SS}	Ground Supply	Mandatory	Core Ground
V_{SSQ}	Ground Supply	Mandatory	Input / Output Ground
DNU	–	–	**Do Not Use**.

2. Interface Overview

2.1 General Description

The Cypress Semper™ Flash family of products are high-speed CMOS, MirrorBit NOR flash devices that are compliant with the JEDEC JESD251 eXpanded SPI (xSPI) specification. Semper Flash is designed for Functional Safety with development according to ISO 26262 standard to achieve ASIL-B compliance and ASIL-D readiness.

Semper Flash with HyperBus Interface devices support both the HyperBus Interface as well as Legacy (x1) SPI. Both interfaces serially transfer transactions reducing the number of interface connection signals. SPI supports SDR whereas HyperBus supports DDR.

The HyperBus interface (DDR) transfers two data bytes per clock cycle on the data (DQ) signals. A read or program/write access 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 Semper Flash when receiving command/address/data information on the DQ signals. The Read Data Strobe (DS) is an output from Semper Flash that indicates when data is being transferred from the memory. DS 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 DS.

Read and program/write operations to Semper Flash 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 one or more16-bit values.

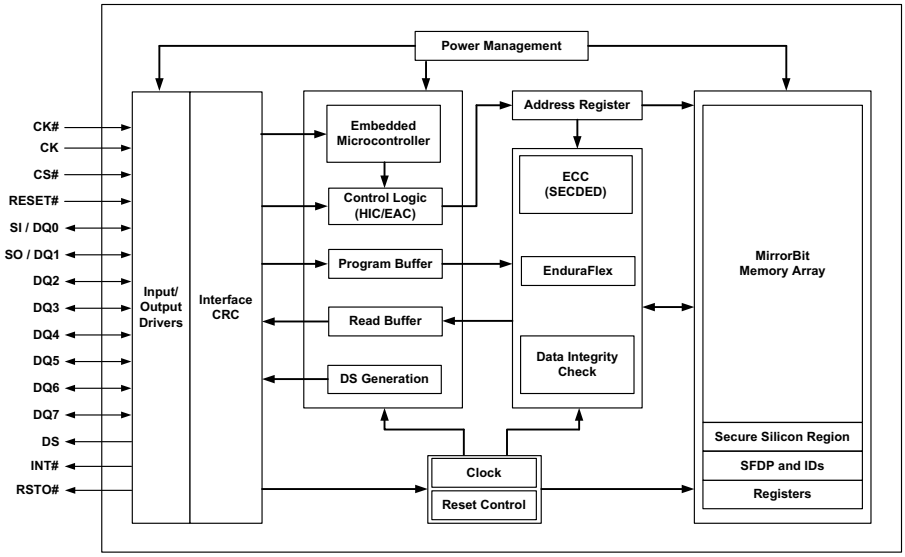
Each random read accesses a 32-Byte length and aligned set of data called a page. Each page consists of a pair of 16-Byte aligned groups of array data called half-pages. Half-pages are aligned on 16-Byte address boundaries. A read access requires two clock cycles to define the target half-page address and the burst type, then an additional initial latency. 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 400/333 MBps data rate *[1-Byte (8-bit data bus) * 2 (Data on both clock edges) * 200/166 MHz = 400/333 MBps]*

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 can change a memory bit from a 0 to a 1. An erase operation must be performed on a complete sector (4 KBs or 256 KBs).

Semper Flash provides a flexible sector architecture. The address space can be configured as either a uniform 256 KB sector array, or a hybrid configuration 1 array where thirty-two 4 KB sectors are either at the top or at the bottom while the remaining sectors are all 256 KB, or a hybrid configuration 2 array where the thirty-two 4 KB sectors are equally split between the top and the bottom while the remaining sectors are all 256 KB.

The Page Programming Buffer used during a single programming operation is configurable to either 256 bytes or 512 bytes. The 512 byte option provides the highest programming throughput.

Figure 2. Logic Block Diagram



The Semper Flash family consists of multiple densities with, 1.8V and 3.0V core and I/O voltage options.

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, program and write data transfers with the host system. 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, and notifies the EAC of power transition, and write transfers. The EAC interrogates the command memory, after a program or write transfer, for legal command sequences and performs the related Embedded Algorithms.

Changing the nonvolatile data in the memory array requires a sequence of operations that are part of 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. The EAC receives the command, performs all the necessary steps to complete the transaction, and provides status information during the progress of an EA.

In addition to the mandatory signals CS#, CK, SI/DQ0, SO/DQ1, DQ [7:2] and DS, the Semper Flash devices also include optional signals CK#, RESET#, INT# and 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 Data Strobe (DS) is synchronized with the output data during read transactions enabling host system to capture data at high clock frequency operation. The INT# is an open-drain output can provide an interrupt to the HyperFlash master to indicate when the HyperFlash transitions from busy to ready at the end of a program or erase operation or to indicate the detection of an ECC error during read. 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 Standby state

EnduraFlex™ Architecture provides system designers the ability to customize Semper Flash's endurance and retention for their specific application. The host defines partitions for high endurance or long retention, providing up to 1+ million cycles or 25 years of data retention.

Semper Flash devices support error detection and correction by generating an embedded Hamming error correction code during memory array programming. This ECC code is then used for single-bit and double-bit error detection and single-bit correction during read.

Semper Flash devices have built-in diagnostic features providing the host system with the device status.

- Program and Erase Operation: Reporting of program or erase success, failure and suspend statuses
- Error Detection and Correction: 1-bit and/or 2-bit error status with address trapping and error count
- Data Integrity Check: Error detection over memory array contents
- Interface CRC: Error detection over interface transactions
- SafeBoot: Reporting of proper flash device initialization and configuration corruption recovery
- Sector Erase Status: Reporting of erase success or failure status per sector
- Sector Erase Counter: Counts the number of erase cycles per sector

2.2 HyperBus Transaction Protocol

Transaction

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 Semper Flash 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, Semper Flash 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 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 devices.
- Whether a transaction will use a linear or wrapped burst sequence.
  - Linear and wrapped burst are fully supported for read transactions. Semper Flash also supports Hybrid burst which combines one wrapped burst followed by linear burst.
  - Semper Flash write transactions do not support burst sequence and ignore the burst type indication. Write command transactions transfer a single word per write. Only the Word Program command write data transfer may be done with a linear burst.
- 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 HyperBus master 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 DS edge (read data) and the second eight bits (7-0) being transferred on the LOW going CK or DS 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 DS transitions and Write data is center aligned with clock edges.

Table 2. Command / Address Bit Assignment

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
44-39 (1 Gb) 44-38 (512 Mb) 44-37 (256 Mb)	Reserved	Reserved for future address expansion. Reserved bits should be set to 0 by the host controller.
38-16 (1 Gb) 37-16 (512 Mb) 36-16 (256 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.

Transaction Capture

CK/CK# mark the transfer of each bit or group of bits between the host and memory. Command, address and write data bits transfer occurs on CK edges or CK/CK# crossing.

Note All attempts to read the flash memory array during a program or erase (embedded operations) are ignored. The embedded operation will continue to execute without any effect. A very limited set of commands are accepted during an embedded operation.

Read

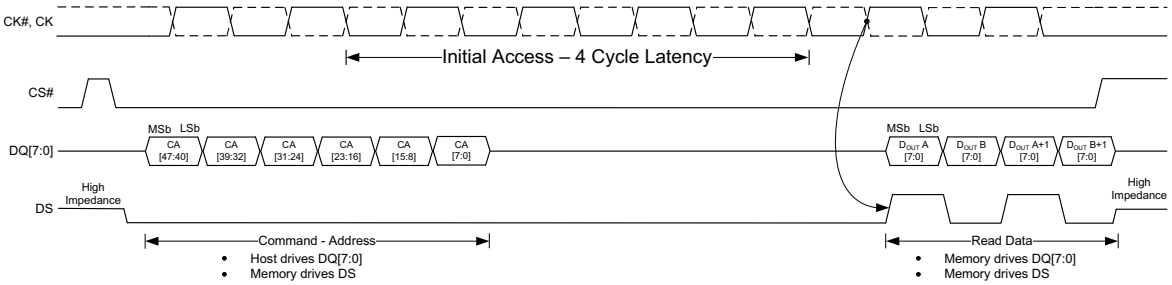
CA0 indicates that a read operation is to be performed and also indicates the burst type (wrapped or linear). Read operations begin the internal array access as soon as the half-page address has been presented in CA0 and CA1. CA2 identifies the target word address within the chosen half-page. The host 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 Data Strobe (DS) and begins outputting the target data. New data is output in an edge aligned fashion upon every transition of DS. 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 4.8.5 Burst Types on page 66. Wrapped reads can be performed from the main array, the SFDP Tables and the Secure Silicon Region (SSR). Read transfers can be ended at any time by bringing CS# High when CK = Low and CK# = High.

When a linear burst reaches the last address in the array, if the burst continues, the address counter will wrap around and roll back to address 000000h, allowing the read sequence to be continued indefinitely.The entire memory can therefore be read out with one single read instruction.

The 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 (see Table 57 on page 67).

Inter-page boundary latency can be avoided by choosing double initial latency CFR2x[7] option. This allows the device to fetch two consecutive pages during each access allowing latency free data access.

Figure 3. Read Operation (Single Initial Latency)[4, 5, 6, 7, 8]

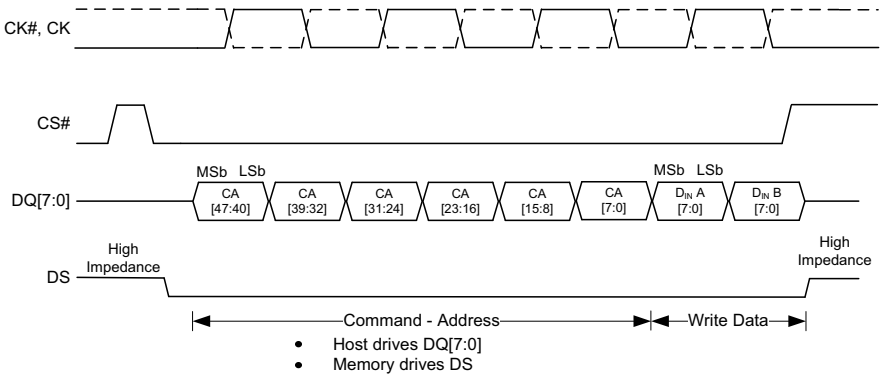


Write Operations

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 Semper Flash device only supports a single write transaction of 16-bit 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.

Notes
4. Transactions must be initiated with CK = LOW and CK# = HIGH. CS# must return HIGH before a new transaction is initiated.
5. Read access from the flash array starts once CA[23:16] is captured.
6. The read latency is defined by the Read Latency value in the Volatile Configuration Register (or the Nonvolatile Configuration Register).
7. In this example of a read operation, the Latency Count was set to four clocks.
8. Data out during a Register Read transaction 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.

Figure 4. Write Operation[9, 10, 11]



2.3 Legacy (x1) SPI Transaction Protocol

Transaction

- During the time that CS# is active (Low) the clock signal (CK) is toggled while command information is first transferred on the data input (SI/DQ0) signal followed by address and data from the host to the flash device. The clock continues to toggle during the transfer of read data from the flash device to the host (SO/DQ1) or write data from the host to the flash device (SI/DQ0). When the host has transferred the desired amount of data, the host drives the CS# inactive (High). The period during which CS# is active is called a transaction on the bus.
- While CS# is inactive, the CK is not required to toggle.
- The command transfer occurs at the beginning of every transaction. The address, latency cycles, and data transfer phases are optional and their presence depends on the protocol mode or command transferred.

Transaction Capture

- CK marks the transfer of each bit or group of bits between the host and memory. Command, address and write data bits transfer occurs on CK rising edge.

Note All attempts to read the flash memory array during a program or erase (embedded operations) are ignored. The embedded operation will continue to execute without any effect. A very limited set of commands are accepted during an embedded operation. These are discussed in Section 4.12 Suspend and Resume Embedded Operation on page 82.

Protocol

- The Legacy (x1) SPI mode is the default protocol following Power-On-Reset (POR) but, flash devices can be configured to boot-up in the HyperBus mode.
- Each transaction begins with an 8-bit (1-byte) command. The command selects the type of information transfer or device operation to be performed.
- This protocol uses SI/DQ[0] to transfer information from host to flash device and SO/DQ[1] to transfer information from flash device to host. On each DQ, information is placed on the DQ line in Most Significant bit (MSb) to Least Significant bit (LSb) order within each byte. Sequential address bytes are transferred in highest order to lowest order sequence. Sequential data bytes are transferred in lowest address to highest address order.
- DQ[7:2] are not used for data transfer period. Hence, the DQ[7:2] signals will be high impedance.

Notes
9. Transactions must be initiated with CK = LOW and CK# = HIGH. CS# must return HIGH before a new transaction is initiated.
10. DS will be driven Low as long as CS# is LOW but is not used during Write transactions.
11. Write operations are limited to a transaction of a single word (16-bit) or a linear write burst supported only when loading data during a Word Program command.

2.3.1 Transaction Details

Figure 5. SPI Transaction with Command Input

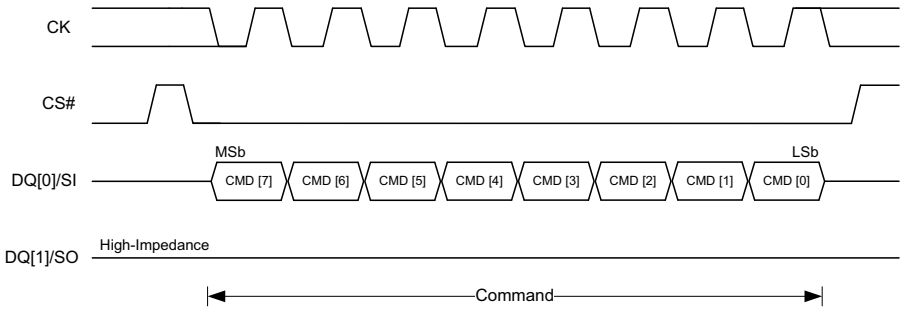


Figure 6. SPI Transaction with Command and Address Input

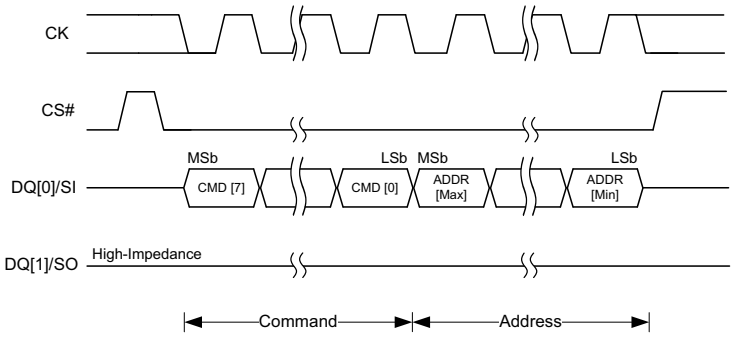


Figure 7. SPI Transaction with Command and Two Input Addresses

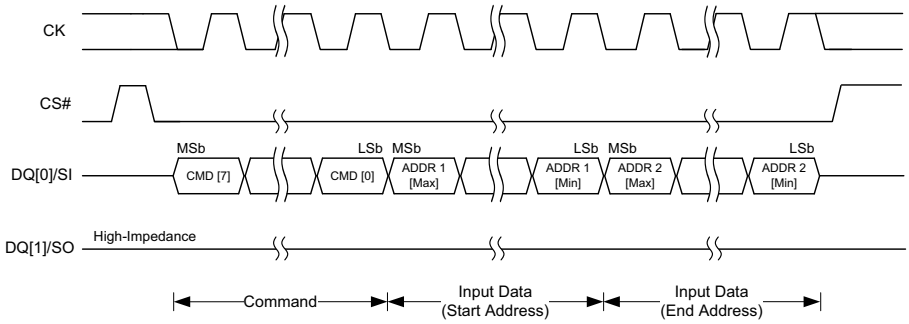


Figure 8. SPI Program Transaction with Command, Address, and Data Input

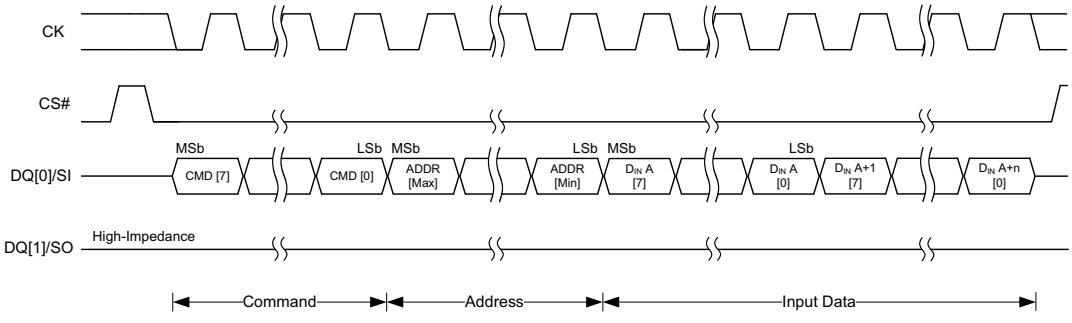


Figure 9. SPI Program Transaction with Command and Data Input

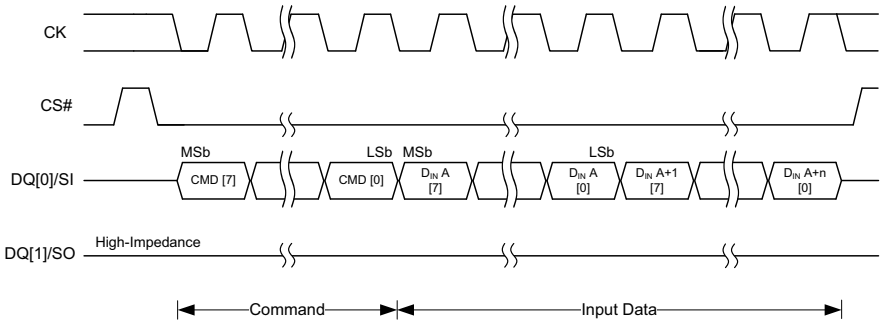


Figure 10. SPI Read Transaction with Command Input (Output Latency)

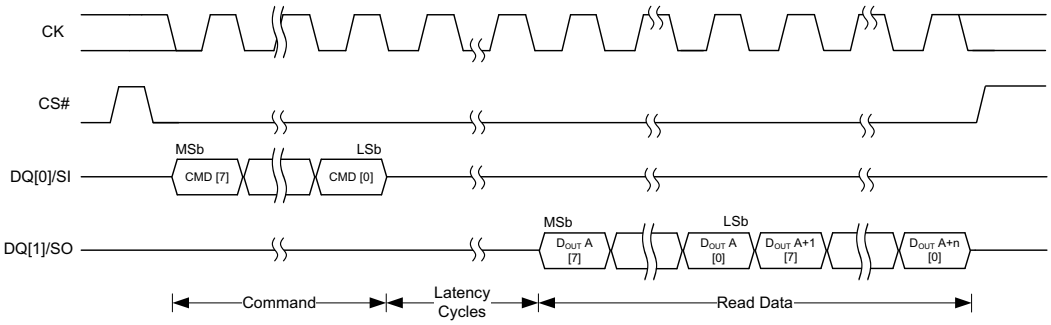


Figure 11. SPI Read Transaction with Command and Address Input (Output Latency)

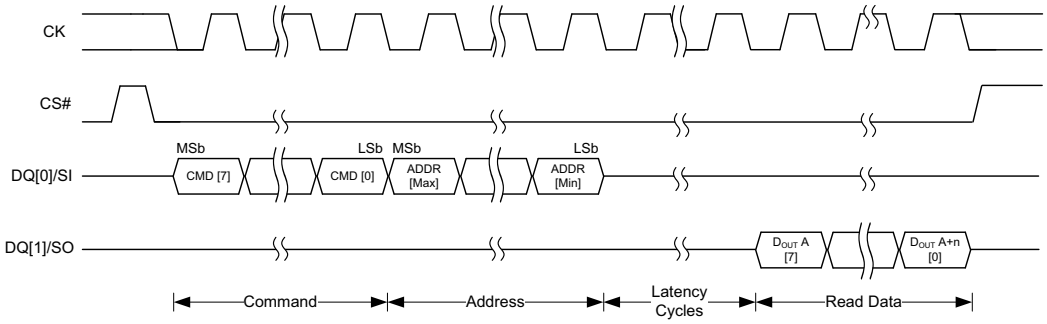


Figure 12. SPI Read Transaction with Command and Address Input (No Output Latency)

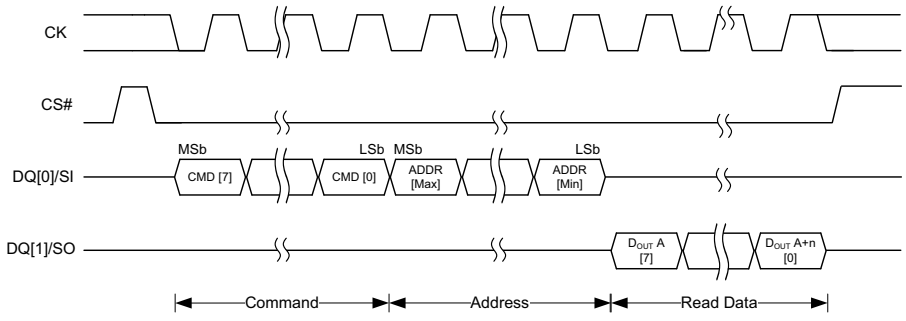
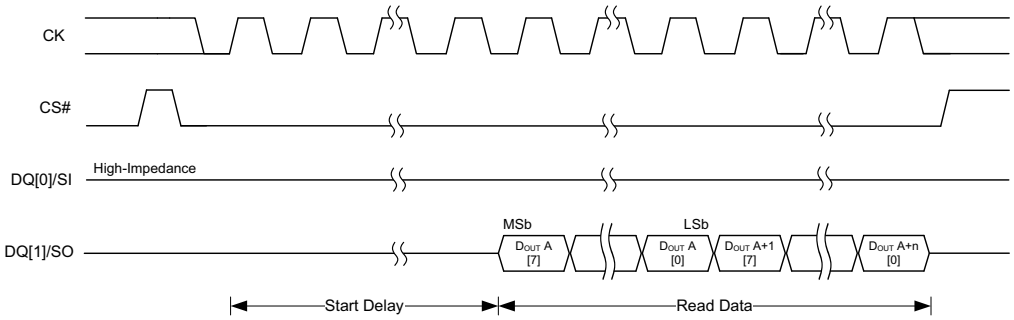


Figure 13. SPI Transaction with Output Data Sequence (AutoBoot)



2.4 Embedded Operations

2.4.1 Embedded Algorithm Controller (EAC)

The EAC takes transactions from the host system for programming and erasing the flash memory arrays and performs all the complex operations needed to change the nonvolatile 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 (DPD)
- Standby (Read mode)
- Address Space Switching
- Embedded Algorithms (EA)
- Advanced Sector Protection (ASP) Management

2.4.1.1 Deep Power Down (DPD)

In the DPD mode, current consumption is driven to the lowest level. The DPD mode must be entered while the device is in the Standby state while not in an ASO.

Note that the device maintains its configuration during DPD. In other words, the device exits DPD in the same state as it went in. However, ECC related registers (ECC Status, Error Detection Counter, and Address Trap) will not maintain their state and will get reset upon DPD exit.

Note that a configuration option exists where the device can power-up in DPD. If selected, CS# must be held High during POR to enter DPD.

2.4.1.2 Standby

In the Standby State current consumption is greatly reduced. The EAC enters its Standby State 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_{CC4}).

2.4.1.3 Address Space Switching - HyperBus Only

Writing specific address and data sequences (transactions) 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 transaction accepted when a particular ASO is entered are listed between the ASO Enter and Exit transactions in the transaction table. See Section 6. Transaction Table on page 126 for address and data requirements for all transaction sequences.

Note that while in any mode, the Status Register read transaction may be issued to cause the Status Register ASO to appear at every word address in the device address space.

2.4.1.4 Embedded Algorithms (EA)

Changing the nonvolatile 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 in HyperBus. 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.

2.4.1.5 Advanced Sector Protection (ASP) Management

ASP provides protection methods to disable or enable programming or erase operations in any or all sectors. The EAC manages this protection to maintain data integrity.

2.5 Register Naming Convention

Figure 14. Register Naming Convention

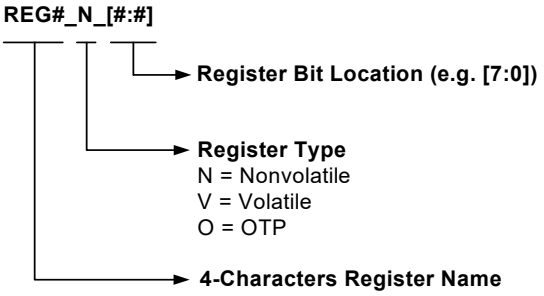
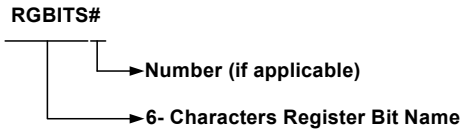
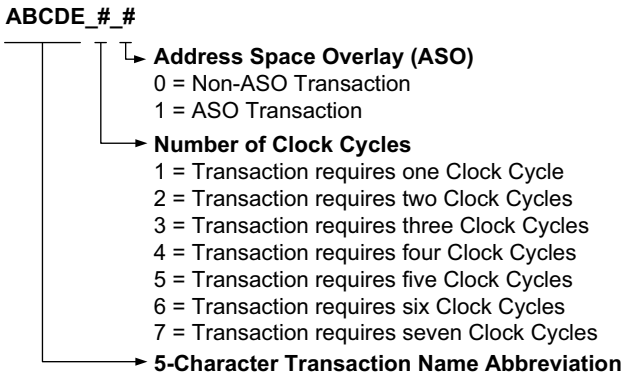


Figure 15. Register Bit Naming Convention



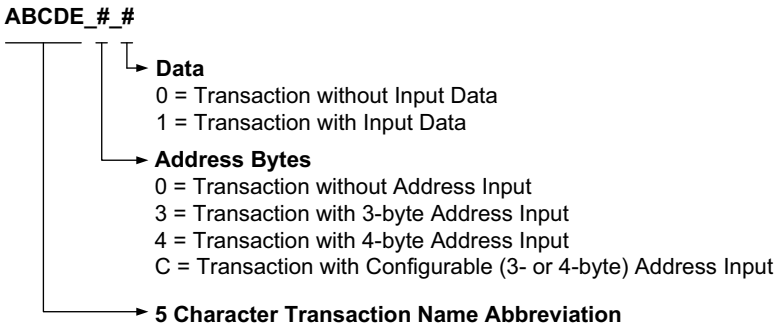
2.6 HyperBus Transaction Naming Convention

Figure 16. HyperBus Transaction Naming Convention



2.7 Legacy (x1) SPI Transaction Naming Convention

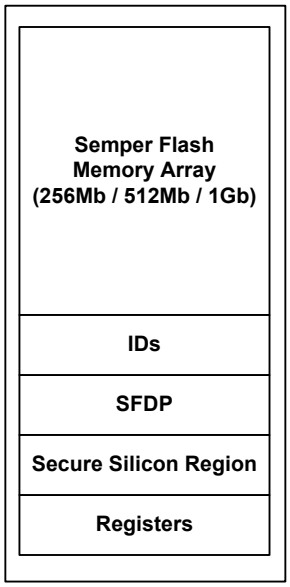
Figure 17. Legacy (x1) SPI Transaction Naming Convention



3. Address Space Maps

The address space for the Semper Flash family contains addressing for the main memory array, Address Space Overlays (ASO), Manufacturer ID, Device ID, Unique ID, Serial Flash Discoverable Parameters (SFDP), SSR, and Registers.

Figure 18. Semper Flash Address Space Map Overview



3.1 Semper Flash Memory Array

The main flash array is divided into units called physical sectors.

The HL-T/HS-T family sector architecture supports the following options:

- 256 Mb, 512 Mb, 1 Gb supports 256 KB Uniform sector options
- 256 Mb, 512 Mb, 1 Gb Hybrid sector options
  - Configuration 1 - Physical set of thirty-two 4 KB sectors and one 128 KB sector at the top or bottom of address space with all remaining sectors of 256 KB
  - Configuration 2 - Physical set of sixteen 4 KB sectors and one 192 KB sector at both the top and bottom of the address space with all remaining sectors of 256 KB

The combination of the sector architecture selection bits in Configuration registers support the different sector architecture options of the Semper Flash family. See Section 5. Registers on page 93 for more information.

Table 3. 256 KB Uniform Sector Address Map

Sector Size (KB)	S26HL01GT and S26HS01GT			S26HL512T and S26HS512T			S26HL256T and S26HS256T		
	Sector Count	Sector Range	Byte Address Range (Sector Starting Address - Sector Ending Address)	Sector Count	Sector Range	Byte Address Range (Sector Starting Address - Sector Ending Address)	Sector Count	Sector Range	Byte Address Range (Sector Starting Address - Sector Ending Address)
256	512	SA00	0000000h – 001FFFFh	256	SA00	0000000h – 001FFFFh	128	SA00	00000000h – 0001FFFFh
		:	:		:	:		:	:
		SA511	3FE0000h – 3FFFFFFh		SA255	1FE0000h – 1FFFFFFh		SA127	0FE0000h – 0FFFFFFh

Table 4. Top Hybrid Configuration 1 Thirty-two 4 KB Sectors and 256 KB Uniform Sectors Address Map

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
	.....			
	30	4 KB	000F000h – 000F7FFh	Parameter-Sector 30
	31	4 KB	000F800h – 000FFFFh	End of Parameter-Sector 31
	Exposed Portion of Main Array Sector 0	128 KB	0010000h – 001FFFFh	Mapped to exposed portion of Main Array Sector 0

Table 5. Bottom Hybrid Configuration 1 Thirty-two 4 KB Sectors and 256 KB Uniform Sectors Address Map

Main Array Sector Size	Parameter-Sector Number	Address Size	Address Range (16-bit)	Notes
256 KB	Exposed portion of last sector in main array	128 KB	xx00000h – xx0FFFFh	Mapped to exposed portion of Main Array Sector (last)
	0	4 KB	xx10000h – xx107FFh	Start of Parameter-Sector 0
	1	4 KB	xx10800h – xx10FFFh	Parameter-Sector 1
	.....			
	30	4 KB	xx0F000h – xx0F7FFh	Parameter-Sector 30
	31	4 KB	xx1F800h – xx1FFFFh	End of Parameter-Sector 31

Table 6. Hybrid Configuration 2 Bottom Sixteen and Top Sixteen 4 KB Sectors Address Map

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
	.....			
	15	4 KB	0007800h – 007FFFh	Parameter-Sector 15
	Exposed Portion of Main Array Sector 0	192 KB	008000h – 001FFFFh	Mapped to exposed portion of Main Array Sector 0
	.....			
	Exposed portion of last sector in main array	192 KB	xx00000h – xx17FFFh	Mapped to exposed portion of Main Array Sector (last)
	0	4 KB	xx18000h – xx187FFh	Start of Parameter-Sector 0
	1	4 KB	xx18800h – xx18FFFh	Parameter-Sector 1
	.....			
	14	4 KB	xx1F000h – xx1F7FFh	Parameter-Sector 14
	15	4 KB	xx1F800h – xx1FFFFh	End of Parameter-Sector 15

These are condensed tables that use a couple of sectors as references. There are address ranges that are not explicitly listed. All 4 KB sectors have the pattern xxxxx000h-xxxxx7FFh. All 256 KB sectors have the pattern xxx00000h-xxx1FFFFh.

3.2 Address Space Overlays (ASO) (HyperBus Only)

Although the HyperBus protocol provides for slave devices that have both memory and register address spaces, Semper Flash devices 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 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 nonvolatile memory array used for storage of data that may be randomly accessed by read operations.
- ID/SFDP (ID): a flash memory array used for Cypress factory programmed device characteristics information. This area contains the Device Identification (ID), Unique ID (UID) and Serial Flash Discoverable Parameter (SFDP) parameters/tables.
- SSR: a 1024B OTP nonvolatile memory array used for Cypress factory programmed permanent data, and customer programmable permanent data.
- Advanced Sector Protection Configuration Register (ASP): a register to configure the device's data protection schemes.
- Password (PSWD): an OTP nonvolatile 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.
- Persistent Protection Bits (PPB): a nonvolatile memory array with one bit for each Sector. When programmed, each bit protects the related Sector from erasure and programming.
- PPB Lock Bit (PPBL): a Volatile Register bit used to enable or disable programming and erase of the PPB bits.
- 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.
- ECC Status (ECCST): read the ECC Status, address of ECC error and total ECC error count.
- Data Integrity Check (DICRC): read the memory array data CRC check-value.
- Interface CRC (ICRC): read the interface CRC check-value.
- AutoBoot (ATB): a Nonvolatile Register to set the starting address and delay for AutoBoot feature.
- Sector Erase Count (SEC): a 32-bit value showing the number of times a sector has been erased.
- EnduraFlex Architecture's Pointers (EFP): a set of five Nonvolatile registers to configure four pointers to choose between either high endurance or long retention memory regions.
- 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 their meaning is 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 or after a Hardware 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 nonvolatile 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 128 on page 126.

The following ASOs have nonvolatile data that may be programmed to change 1s to 0s:

- SSR
- ASP Configuration Register (ASPR)
- Persistent Protection Bits (PPB)
- Password
- AutoBoot
- EnduraFlex Architecture’s Data Pointers
- Only the PPB and AutoBoot ASOs have nonvolatile data that may be erased to change 0s to 1s.

When a program or erase command is issued while one of the nonvolatile 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 POR Timer, Interrupt Configuration Register, Interrupt Status Register, Volatile Configuration Register, and the Nonvolatile Configuration Register.

3.3 Device ID Address Space

This particular region of the memory is assigned to manufacturer and device identification.

- The manufacturer identification is assigned by JEDEC.
- The device identification is assigned by Cypress.

Table 7. Device ID

Address (Word)	Data	Description
(SA) + 0800h	0034h	Cypress Manufacturer ID
(SA) + 0801h	007Bh = 1.8V 006Ah = 3.0V	Interface Voltage Type
(SA) + 0802h	0019h = 256 Mb 001Ah = 512 Mb 001Bh = 1024 Mb	Device Density
(SA) + 0803h	000Fh	Device ID Length
(SA) + 0804h	0090h	Device Sector Architecture (Shipped from the factory)
(SA) + 0805h	FFFFh	Device Family ID

Note
12. Legacy (x1) SPI uses Read Identification (RDIDN_0_0) transaction to read the Device ID. No address is required.

Table 7. Device ID (Continued)

Address (Word)	Data	Description
(SA) + 0806h	FFFFh	Reserved
(SA) + 0807h	FFFFh	
(SA) + 0808h	FFFFh	
(SA) + 0809h	FFFFh	Reserved
(SA) + 080Ah	FFFFh	
(SA) + 080Bh	FFFFh	
(SA) + 080Ch	FFFFh	
(SA) + 080Dh	FFFFh	
(SA) + 080Eh	FFFFh	
(SA) + 080Fh	FFFFh	

Note
12. Legacy (x1) SPI uses Read Identification (RDIDN_0_0) transaction to read the Device ID. No address is required.

3.4 HyperBus Unique ID

Table 8. HyperBus Unique ID

Address (Word)	Data
(SA) + 0200h	Device Dependent
(SA) + 0201h	
(SA) + 0202h	
(SA) + 0203h	

Notes
13. 0s shown. Actual Unique ID is device dependent.
14. Legacy (x1) SPI uses Read Unique ID (RDUID_0_0) transaction to read the Unique ID. No address is required.

3.5 SFDP JEDEC JESD216 Serial Flash Discoverable Parameters (SFDP) Space

Table 9. SFDP Overview Address Map

Byte Address		Description
0000h	0000h	Location zero within JEDEC JESD216C SFDP space - start of SFDP header
,,,		Remainder of SFDP header followed by undefined space
0100h	0080h	Start of SFDP parameter tables. The SFDP parameter table data starting at 0100h.
...		Remainder of SFDP parameter tables followed by either more parameters or undefined space

Note
15. Legacy (x1) SPI uses Read Serial Discoverable Parameters (RSFDP_3_0) transaction to read the Unique ID. Address is required.

The SFDP standard provides a consistent method of describing the functional and feature capabilities of this serial flash device in a standard set of internal parameter tables. These parameter tables can be interrogated by host system software to enable adjustments needed to accommodate divergent features. The SFDP address space has a header starting at address zero that identifies the SFDP data structure and provides a pointer to each parameter. The SFDP address space is programmed by Cypress and read-only for the host system.

HyperBus accesses the ID-SFDP ASO by issuing the ID-SFDP Entry command sequence during Read mode. SFDP is an Address Overlay and must be mapped to a Sector Address (SA).

Legacy (x1) SPI uses RSFDP_3_0 transaction to access SFDP and does not require an Address Overlay. Table 10 shows (SA) which must be omitted for Legacy (x1) SPI.

3.5.1 JEDEC SFDP Rev D Header Table

Table 10. JEDEC SFDP Rev D Header Table

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
00h	(SA) + 00h	SFDP Header	53h	This is the entry point for Read SFDP (5Ah) command i.e., location zero within SFDP space ASCII “S”
01h			46h	ASCII “F”
02h	(SA) + 01h		44h	ASCII “D”
03h			50h	ASCII “P”
04h	(SA) + 02h		08h	SFDP Minor Revision (08h = JEDEC JESD216 Revision D)
05h			01h	SFDP Major Revision (01h = JEDEC JESD216 Revision D) This is the original major revision. This major revision is compatible with all SFDP reading and parsing software.
06h	(SA) + 03h		03h	Number of Parameter Headers (zero based, 03h = 4 parameters)
07h			FFh (x1) FAh (x8)	Booting up in SPI 1S-1S-1S (FFh) Booting up in xSPI NOR Profile 2 HyperBus, (8D, 8D, 8D) operation, 5-byte addressing for SFDP command, 8 wait states (FAh)
08h	(SA) + 04h	1st Parameter Header	00h	Parameter ID LSB (00h = JEDEC SFDP Basic SPI Flash Parameter)
09h			00h	Parameter Minor Revision (00h = JEDEC JESD216 Revision D)
0Ah	(SA) + 05h		01h	Parameter Major Revision (01h = The original major revision - all SFDP software is compatible with this major revision.
0Bh			14h	Parameter Table Length (14h = 20 DWORDs are in the Parameter table)
0Ch	(SA) + 06h		00h (x1) 80h (x8)	Parameter Table Pointer Byte 0 (DWORD = 4-byte aligned) JEDEC Basic SPI Flash parameter x1 byte offset = 0100h, x8 word offset = 0080h
0Dh			01h (x1) 00h (x8)	Parameter Table Pointer Byte 1
0Eh	(SA) + 07h		00h	Parameter Table Pointer Byte 2
0Fh			FFh	Parameter ID MSB (FFh = JEDEC defined legacy Parameter ID)
10h	(SA) + 08h	2nd Parameter Header	81h	Parameter ID LSB (81h = JEDEC Sector Map)
11h			00h	Parameter Table Minor Revision (00h = JEDEC JESD216 Revision D)
12h	(SA) + 09h		01h	Parameter Table Major Revision (01h = JEDEC JESD216 Revision D)
13h			1Ah	Parameter Table Length (1Ah = 26 DWORDs are in the Parameter table)
14h	(SA) + 0Ah		50h (x1) A8h (x8)	Parameter Table Pointer Byte 0 (DWORD = 4 byte aligned) JEDEC Sector Map = x1 byte offset = 0150h, x8 word offset = 00A8h
15h			01h (x1) 00h (x8)	Parameter Table Pointer Byte 1
16h	(SA) + 0Bh		00h	Parameter Table Pointer Byte 2
17h			FFh	Parameter ID MSB (FFh = JEDEC defined Parameter)
18h	(SA) + 0Ch	3rd Parameter Header	06h	Parameter ID LSB (JEDEC xSPI (HyperBus) Profile 2.0)
19h			00h	Parameter Table Minor Revision (00h = JEDEC JESD216 Revision D)
1Ah	(SA) + 0Dh		01h	Parameter Table Major Revision (01h = JEDEC JESD216 Revision D)
1Bh			03h	Parameter Table Length (3h = 3 DWORDs are in the Parameter table)
1Ch	(SA) + 0Eh		B8h (x1) DCh (x8)	Parameter Table Pointer Byte 0 (DWORD = 4 byte aligned) JEDEC xSPI Profile 2.0 = x1 byte offset = 01B8h, x8 word offset = 00DCh
1Dh			01h (x1) 00h (x8)	Parameter Table Pointer Byte 1
1Eh	(SA) + 0Fh		00h	Parameter Table Pointer Byte 2
1Fh			FFh	Parameter ID MSB (FFh = JEDEC defined Parameter)

Table 10. JEDEC SFDP Rev D Header Table (Continued)

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
20h	(SA) + 10h	4th Parameter Header	0Ah	Parameter ID LSB (0Ah = Command Sequences to change to HyperBus (8D-8D-8D) mode)
21h			00h	Parameter Table Minor Revision (00h = JEDEC JESD216 Revision D)
22h	(SA) + 11h		01h	Parameter Table Major Revision (01h = JEDEC JESD216 Revision D)
23h			04h	Parameter Table Length (4h = 4DWORDs are in the Parameter table)
24h	(SA) + 12h		C4h (x1) E2h (x8)	Parameter Table Pointer Byte 0 (DWORD = 4 byte aligned) JEDEC Status, Control and Configuration Register Map x1 byte offset = 01C4h, x8 word offset = 00E2h
25h			01h (x1) 00h (x8)	Parameter Table Pointer Byte 1
26h	(SA) + 13h		00h	Parameter Table Pointer Byte 2
27h			FFh	Parameter ID MSB (FFh = JEDEC defined Parameter)

3.5.2 JEDEC SFDP Rev D Parameter Table

For the SFDP data structure, there are three independent parameter tables. Two of the tables have a fixed length and one table has a variable structure and length depending on the device density Ordering Part Number (OPN). The Parameter table is presented as single table in Table 11.

Table 11. JEDEC SFDP Rev D Parameter Table

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
100h	(SA) + 80h	JEDEC Basic Flash Parameter DWORD-1	E7h	Bits 7:5 = unused = 111b Bit 4 = 50h is Volatile Status Register write instruction and Status Register is default = 0b Bit 3 = Block Protect Bits are Nonvolatile / volatile Nonvolatile = 0b Bit 2 = Program Buffer > 64Bytes = 1b Bits 1:0 = Uniform 4 KB erase is unavailable = 11b
101h			21h	Bits 15:8 = Uniform 4 KB erase opcode = 21h
102h	(SA) + 81h		8Ah	Bit 23 = Unused = 1b Bit 22 = Supports Quad Out (1-1-4) Read = No = 0b Bit 21 = Supports Quad I/O (1-4-4) Read = No = 0b Bit 20 = Supports Dual I/O (1-2-2) Read = No = 0b Bit 19 = Supports DDR = Yes = 1b Bit 18:17 = Number of Address Bytes = 3- or 4-Bytes = 01b Bit 16 = Supports Dual Out (1-1-2) Read = No = 0b
103h			FFh	Bits 31:24 = Unused = FFh
104h	(SA) + 82h	JEDEC Basic Flash Parameter DWORD-2	FFh	Density in bits, zero based, 256 Mb = 0FFFFFFFh Density in bits, zero based, 512 Mb = 1FFFFFFFh Density in bits, zero based, 1 Gb = 3FFFFFFFh
105h			FFh	
106h	(SA) + 83h		FFh	
107h			0Fh for 256M 1Fh for 512M 3Fh for 1G	
108h	(SA) + 84h	JEDEC Basic Flash Parameter DWORD-3	00h	Not Supported
109h			00h	
10Ah	(SA) + 85h		00h	
10Bh			00h	
10Ch	(SA) + 86h	JEDEC Basic Flash Parameter DWORD-4	00h	Not Supported
10Dh			00h	
10Eh	(SA) + 87h		00h	
10Fh			00h	

Table 11. JEDEC SFDP Rev D Parameter Table (Continued)

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
110h	(SA) + 88h	JEDEC Basic Flash Parameter DWORD-5	EEh	Bits 7:5 RFU = 111b Bit 4 = Not Supported = 0b Bits 3:1 RFU = 111b Bit 0 = 2-2-2 Not Supported = 0b
111h			FFh	Bits 15:8 = RFU = FFh
112h	(SA) + 89h		FFh	Bits 23:16 = RFU = FFh
113h			FFh	Bits 31:24 = RFU = FFh
114h	(SA) + 8Ah	JEDEC Basic Flash Parameter DWORD-6	FFh	Bits 7:0 = RFU = FFh
115h			FFh	Bits 15:8 = RFU = FFh
116h	(SA) + 8Bh		00h	Not Supported
117h			00h	
118h	(SA) + 8Ch	JEDEC Basic Flash Parameter DWORD-7	FFh	Bits 7:0 = RFU = FFh
119h			FFh	Bits 15:8 = RFU = FFh
11Ah	(SA) + 8Dh		00h	Not Supported
11Bh			00h	
11Ch	(SA) + 8Eh	JEDEC Basic Flash Parameter DWORD-8	0Ch	Erase type 1 size 2^N Bytes = 2^12 Bytes = 4 KB (Initial Delivery State)
11Dh			21h	Erase type 1 Instruction
11Eh	(SA) + 8Fh		00h	Erase type 2 size 2^N Bytes = Not Supported
11Fh			FFh	Erase type 2 instruction = Not Supported
120h	(SA) + 90h	JEDEC Basic Flash Parameter DWORD-9	00h	Erase type 3 not supported
121h			FFh	Erase type 3 instruction
122h	(SA) + 91h		12h	4 Size, 256 KB erase instruction = Erase type size = 2^N (where N= 18) = 12h
123h			DCh	Erase type 4 instruction = DCh
124h	(SA) + 92h	JEDEC Basic Flash Parameter DWORD-10	23h	Bits 31:30 = Erase type 4 Erase, Typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 128 ms = 10b Bits 29:25 = Erase type 4 Erase, Typical time count = 00101b (typ erase time = count +1 * units = 6 * 128 ms = 768 ms) Bits 24:23 = Erase type 3 Erase, Typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 1S = 11b (RFU) Bits 22:18 = Erase type 3 Erase, Typical time count = 11111b (RFU) Bits 17:16 = Erase type 2 Erase, Typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 1S = 11b (RFU) Bits 15:11 = Erase type 2 Erase, Typical time count = 11111b (RFU) Bits 10:9 = Erase type 1 Erase, Typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 16ms = 01b Bits 8:4 = Erase type 1 Erase, Typical time count = 00010b (typ erase time = count +1 * units = 3 * 16 ms = 48 ms) Bits 3:0 = Count = (Max Erase time / (2 * Typical Erase time)) - 1 = 0011b
125h			FAh	
126h	(SA) + 93h		FFh	
127h			8Bh	
128h	(SA) + 94h	JEDEC Basic Flash Parameter DWORD-11	91h	Bits 31 = Reserved = 1b Bits 30:29 = Chip Erase Typical time units (00b: 16 ms, 01b: 256 ms, 10b: 4 s, 11b: 64 s) = 11b (256M, 512M, and 1G) Bits 28:24 = Chip Erase Typical time count = 00001b (256M), 00011b (512M), and 00110b (1G) Bits 23:19 = Byte Program Typical Time, additional byte = 11111b Bits 18:14 = Byte Program Typical Time, first byte = 11111b Bits 13 = Page Program Typical Time unit (0: 8 µs, 1: 64 µs) = 64 µs = 1b Bits 12:8 = Page Program Typical Time Count = 01000 (typ Program time = count +1 * units = 9 * 64 µs = 576 µs) Bits 7:4 = Page Size (512B) = 2^N bytes = 1001h Bits 3:0 = Count = [Max page program time / (2 * Typical page program time)] - 1 = 0001b
129h			E8h	
12Ah	(SA) + 95h		FFh	
12Bh			E1h for 256M E3h for 512M E6h for 1G	

Table 11. JEDEC SFDP Rev D Parameter Table (Continued)

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
12Ch	(SA) + 96h	JEDEC Basic Flash Parameter DWORD-12	ECh	Bit 31 = Suspend and Resume supported = 0b Bits 30:29 = Suspend in-progress erase max latency units (00b: 128ns, 01b: 1µs, 10b: 8µs, 11b: 64µs) = 64 µs = 11b Bits 28:24 = Suspend in-progress erase max latency count = 00000b, max erase suspend latency = count +1 * units = 1 * 64 µs = 64 µs Bits 23:20 = Erase resume to suspend interval count = 0001b, interval = count +1 * 64 µs = 2 * 64 µs = 128 µs Bits 19:18 = Suspend in-progress program max latency units (00b: 128ns, 01b: 1µs, 10b: 8µs, 11b: 64µs) = 64 µs = 11b Bits 17:13 = Suspend in-progress program max latency count = 00000b, max erase suspend latency = count +1 * units = 1 * 64 µs = 64 µs Bits 12:9 = Program resume to suspend interval count = 0001b, interval = count +1 * 64 µs = 2 * 64 µs = 128 µs Bit 8 = Reserved = 1b Bits 7:4 = Prohibited operations during erase suspend = xxx0b: May not initiate a new erase anywhere (erase nesting not permitted) + xx1xb: May not initiate a page program in the erase suspended sector size + x1xxb: May not initiate a read in the erase suspended sector size + 1xxxb: The erase and program restrictions in bits 5:4 are sufficient = 1110b Bits 3:0 = Prohibited Operations During Program Suspend = xxx0b: May not initiate a new erase anywhere (erase nesting not permitted) + xx0xb: May not initiate a new page program anywhere (program nesting not permitted) + x1xxb: May not initiate a read in the program suspended page size + 1xxxb: The erase and program restrictions in bits 1:0 are sufficient = 1100b
12Dh			03h	
12Eh	(SA) + 97h		1Ch	
12Fh			60h	
130h	(SA) + 98h	JEDEC Basic Flash Parameter DWORD-13	30h	Bits 31:24 = Erase Suspend Instruction = B0h Bits 23:16 = Erase Resume Instruction = 30h Bits 15:8 = Program Suspend Instruction = B0h Bits 7:0 = Program Resume Instruction = 30h
131h			B0h	
132h	(SA) + 99h		30h	
133h			B0h	
134h	(SA) + 9Ah	JEDEC Basic Flash Parameter DWORD-14	F7h	Bits 7:4 = RFU = Fh Bit 3:2 = Status Register Polling Device Busy = 01b: Legacy status polling supported = Use legacy polling by reading the Status Register with 05h instruction and checking WIP bit[0] (0 = ready; 1 = busy). Bits 1:0 = RFU = 11b
135h			66h	Bit 31 = DPD Supported = Supported = 0 Bits 30:23 = Enter DPD Instruction = B9h Bits 22:15 = Exit DPD Instruction not supported = 00h Bits 14:13 = Exit DPD to next operation delay units = (00b: 128ns, 01b: 1µs, 10b: 8µs, 11b: 64µs) = 64µs = 11b Bits 12:8 = Exit DPD to next operation delay count = 00110, Exit DPD to next operation delay = (count+1) * units = (6 + 1) * 64 µs = 448 µs
136h	(SA) + 9Bh		72h	
137h			01h	
138h	(SA) + 9Ch	JEDEC Basic Flash Parameter DWORD-15	00h	Bits 31:24 = Reserved = FFh Bit 23 = Hold or RESET Disable = not supported = 0b Bits 22:0 = Not supported = 000000h
139h			00h	
13Ah	(SA) + 9Dh		00h	
13Bh			FFh	
13Ch	(SA) + 9Eh	JEDEC Basic Flash Parameter DWORD-16	F9h	Bit 7 = Reserved = 1 Bits 6:0 = Volatile or Nonvolatile Register and Write Enable Instruction for Status Register 1 = xxx_xxx1b: Nonvolatile Status Register 1, powers-up to last written value, use instruction 06h to enable write. + xxx_1xxxb: Nonvolatile/Volatile Status Register 1 powers-up to last written value in the nonvolatile status register, use instruction 06h to enable write to nonvolatile status register. Volatile status register may be activated after power-up to override the nonvolatile status register, use instruction 50h to enable write and activate the volatile status register. + xx1_xxxxb: Status Register 1 contains a mix of volatile and nonvolatile bits. The 06h instruction is used to enable writing of the register. + x1x_xxxxb: Reserved + 1xx_xxxxb: Reserved = 1111001b
13Dh			10h	
13Eh	(SA) + 9Fh		00h	Bits 31:24 = Enter 4-byte Addressing + xx1x_xxxxb: Supports dedicated 4-Byte address instruction set. Refer to the vendor datasheet for the instruction set definition. + 1xxx_xxxxb: Reserved = 1010_0000b Bits 23:14 = Not supported = 000h Bits 13:8 = Soft Reset and Rescue Sequence Support + x1_xxxxb: issue reset enable instruction 66h, then issue reset instruction 99h. The reset enable, reset sequence may be issued on 1, 2, or 4 wires depending on the device operating mode. = 010000b
13Fh			A0h	

Table 11. JEDEC SFDP Rev D Parameter Table (Continued)

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
140h	(SA) + A0h	JEDEC Basic Flash Parameter DWORD-17	00h	Not Supported
141h			00h	
142h	(SA) + A1h		00h	
143h			00h	
144h	(SA) + A2h	JEDEC Basic Flash Parameter DWORD-18	00h	Bit 31 = 0b Bit 30:29 = 10b Bit 28 = 0b Bit 27:26 = Not supported = 00b Bits 25:24 = 00b Bit 23 = JEDEC SPI Protocol Reset Supported = 1b Bit 22:18 = 00001b xSPI Support for drive strength Bits 17:0 = Reserved = 00000h
145h			00h	
146h	(SA) + A3h		84h	
147h			40h	
148h	(SA) + A4h	JEDEC Basic Flash Parameter DWORD-19	00h	Not Supported
149h				
14Ah	(SA) + A5h			
14Bh				
14Ch	(SA) + A6h	JEDEC Basic Flash Parameter DWORD-20	FFh	Bits 31:28 = Maximum operation speed of device in 8D-8D-8D mode when utilizing Data Strobe = 1000b (200 MHz) / 0111b (166 MHz) Bits 27:24 = 8D-8D-8D mode without using Data Strobe is not characterized = 1110b Bits 23:20 = 1111b Bits 19:16 = 1111b Bit 15:0 = Not Supported = FFFFh
14Dh			FFh	
14Eh	(SA) + A7h		FFh	
14Fh			8Eh HS-T 7Eh HL-T	

3.5.3 Sector Map Parameter

Table 12 provides a means to identify how the device address map is configured and provides a sector map for each supported configuration. This is done by defining a sequence of commands to read out the relevant configuration register bits that affect the selection of an address map. When more than one configuration bit must be read, all the bits are concatenated into an index value that is used to select the current address map.The value of some configuration bits may make other configuration bit values not relevant (don't care), hence not all possible combinations of the index value define valid address maps. Only selected configuration bit combinations are supported by the SFDP Sector Map Parameter Table. Other combinations must not be used in configuring the sector address map when using this SFDP parameter table to determine the sector map. The following index value combinations are supported.

3.5.3.1 SPI x1 Configuration Register Index Value

- CFR3N[3] - 0 = Hybrid Architecture, 1 = Uniform Architecture
- CFR1N[2] - 0 = 4-KB parameter sectors at bottom, 1 = 4-KB sectors at top
- CFR1N[6] - 0 = 4-KB parameter grouped together, 1 = 4-KB sectors split between bottom and top

Table 12. SPI x1 Sector Map Parameter

CFR3N[3]	CFR1N[2]	CFR1N[6]	Index Value	Description
0	0	0	00h	4-KB sectors at bottom with remainder 256-KB sectors
0	1	0	03h	4-KB sectors at top with remainder 256-KB sectors
0	X	1	01h	4-KB sectors split between top and bottom with remainder 256-KB sectors
1	X	X	04h	Uniform 256-KB sectors

3.5.3.2 HyperBus x8 Configuration Register Index Value

- CFR1N[9:8] - 00 = 4-KB parameter sectors at bottom, 01 = 4-KB parameter sectors at top, 1X = Uniform 256-KB sectors
- CFR2N[2] - 0 = 1 = 4-KB parameter sectors split between bottom and top, when uniform sectors are not selected.

Table 13. HyperBus x8 Sector Map Parameter

CFR1N[9:8]	CFR2N[2]	Index Value	Description
00	0	00h	4-KB sectors at bottom with remainder 256-KB sectors
01	0	03h	4-KB sectors at top with remainder 256-KB sectors
XX	1	01h	4-KB sectors split between top and bottom with remainder 256-KB sectors
10 or 11	X	04h	Uniform 256-KB sectors

Table 14. JEDEC SFDP Rev D, Sector Map Parameter Table

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
150h	(SA) + A8h	JEDEC Sector Map Parameter DWORD-1 Config. Detect-1	FCh	Config. Detect -1 x1 Uniform 256 KB Sectors or Hybrid Sectors Bits 31:24 = Read data mask = 0000_1000b: Select bit 3 of the data byte for UNHYSA value 0 = Hybrid map with 4 KB parameter sectors 1 = Uniform map Bits 23:22 = Configuration detection command address length = 11b: Variable length Bits 21:20 = RFU = 11b Bits 19:16 = Configuration detection command latency = 1111b: variable latency Bits 15:8 = Configuration detection instruction = 65h: Read any register Bits 7:2 = RFU = 111111b Bit 1 = Command Descriptor = 0 Bit 0 = Not the end descriptor = 0
151h			65h	
152h	(SA) + A9h		FFh	
153h			08h	
154h	(SA) + AAh	JEDEC Sector Map Parameter DWORD-2 Config. Detect-1	04h	Bits 31:0 = Address Value Configuration Register 3 (bit 3) = 00800004h
155h			00h	
156h	(SA) + ABh		80h	
157h			00h	
158h	(SA) + ACh	JEDEC Sector Map Parameter DWORD-3 Config. Detect-2	FCh	Config. Detect-2 x1 4 KB Hybrid Sectors Split between Top and Bottom Bits 31:24 = Read data mask = 0100_0000b: Select bit 6of the data byte for SP4 KBS value 0 = 4 KB parameter sectors are grouped together 1 = 4 KB parameter sectors are split between High and Low Addresses Bits 23:22 = Configuration detection command address length = 11b: Variable length Bits 21:20 = RFU = 11b Bits 19:16 = Configuration detection command latency = 1111b: variable latency Bits 15:8 = Configuration detection instruction = 65h: Read any register Bits 7:2 = RFU = 111111b Bit 1 = Command Descriptor = 0 Bit 0 = Not the end descriptor = 0
159h			65h	
15Ah	(SA) + ADh		FFh	
15Bh			40h	
15Ch	(SA) + AEh	JEDEC Sector Map Parameter DWORD-4 Config. Detect-2	02h	Bits 31:0 = Address Value Configuration Register 1 (bit 6) = 00800002h
15Dh			00h	
15Eh	(SA) + AFh		80h	
15Fh			00h	
160h	(SA) + B0h	JEDEC Sector Map Parameter DWORD-5 Config. Detect-3	FCh	Config Detect-3 x1 4 KB Hybrid Sectors on Top or Bottom Bits 31:24 = Read data mask = 0000_0100b: Select bit 2 of the data byte for TB4 KBS value 0 = 4 KB parameter sectors at bottom 1 = 4 KB parameter sectors at top Bits 23:22 = Configuration detection command address length = 11b: Variable length Bits 21:20 = RFU = 11b Bits 19:16 = Configuration detection command latency = 1111b: variable latency Bits 15:8 = Configuration detection instruction = 65h: Read any register Bits 7:2 = RFU = 111111b Bit 1 = Command Descriptor = 0 Bit 0 = Not the end descriptor = 0
161h			65h	
162h	(SA) + B1h		FFh	
163h			04h	
164h	(SA) + B2h	JEDEC Sector Map Parameter DWORD-6 Config. Detect-3	02h	Bits 31:0 = Address Value Configuration Register 1 (bit 2)= 00800002h
165h			00h	
166h	(SA) + B3h		80h	
167h			00h	
168h	(SA) + B4h	JEDEC Sector Map Parameter DWORD-7 Config. Detect-4	FCh	Config. Detect-2 x8 4 KB Hybrid Sectors Split between Top and Bottom Bits 31:24 = Read 1st byte data mask = 0000_0100b: Select bit 2 of the data byte for SP4KBS value 0 = 4 KB parameter sectors are grouped together 1 = 4 KB parameter sectors are split between High and Low Addresses Bits 23:22 = Configuration detection command address length = 11b: Variable length Bits 21:20 = RFU = 11b Bits 19:16 = Configuration detection command latency = 1111b: variable latency Bits 15:8 = Configuration detection instruction = C9h: Read Configuration register 1 Bits 7:2 = RFU = 111111b Bit 1 = Command Descriptor = 0 Bit 0 = Not the end descriptor = 0
169h			C9h	
16Ah	(SA) + B5h		FFh	
16Bh			04h	

Table 14. JEDEC SFDP Rev D, Sector Map Parameter Table (Continued)

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
16Ch	(SA) + B6h	JEDEC Sector Map Parameter DWORD-8 Config. Detect-4	00h	Bits 31:0 = Address Value Configuration Register 1 (bit 2) = 00000000h
16Dh			00h	
16Eh	(SA) + B7h		00h	
16Fh			00h	
170h	(SA) + B8h	JEDEC Sector Map Parameter DWORD-9 Config. Detect-5	FDh	Config. Detect -1 x8 Uniform 256 KB Sectors or Hybrid at Top or Bottom Sectors Bits 31:24 = Read 2nd byte data mask = 0000_0011b: Select bit 9 and 8 of the top data byte for TBKBS value 00 = Hybrid map with Bottom 4 KB parameter sectors,1 01 = Hybrid map with top Parameter sectors 1x = Uniform sectors Bits 23:22 = Configuration detection command address length = 11b: Variable length Bits 21:20 = RFU = 11b Bits 19:16 = Configuration detection command latency = 1111b: variable latency Bits 15:8 = Configuration detection instruction = C7h: Read Configuration Register 2 Bits 7:2 = RFU = 111111b Bit 1 = Command Descriptor = 0 Bit 0 = Is the end descriptor = 1
171h			C7h	
172h	(SA) + B9h		FFh	
173h			03h	
174h	(SA) + BAh	JEDEC Sector Map Parameter DWORD-10 Config. Detect-5	00h	Bits 31:0 = Address Value Configuration Register 2 (bit 9 & 8) = 00000000h
175h			00h	
176h	(SA) + BBh		00h	
177h			00h	
178h	(SA) + BCh	JEDEC Sector Map Parameter DWORD-11 Config-0 Header	FEh	Configuration Index 00h 4 KB sectors at bottom with remainder 256 KB Bits 31:24 = RFU = FFh Bits 23:16 = Region count (DWORDs -1) = 02h: Three regions Bits 15:8 = Configuration ID = 00h, 4 KB sectors bottom with remainder 256 KB Bits 7:2 = RFU = 111111b Bit 1 = Map Descriptor = 1 Bit 0 = Not the end descriptor = 0
179h			00h	
17Ah	(SA) + BDh		02h	
17Bh			FFh	
17Ch	(SA) + BEh	JEDEC Sector Map Parameter DWORD-12 Config-0 Region-0	F1h	Region 0 of 4 KB sectors Bits 31:8 = Region size (thirty-two 4 KB) = 0001F3h: Region size as count-1 of 256 Byte units = 32 x 4 KB sectors = 128 KB Count = 128 KB/256 = 500, value = count -1 = 500-1 = 499 = 1F3h Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 0b ---Erase Type 4 is not defined Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is 256 KB erase and is not supported in the 4 KB sector region Bit 1 = Erase Type 2 support = 0b ---Erase Type 2 is 64 KB erase and is not supported Bit 0 = Erase Type 1 support = 1b ---Erase Type 1 is 4 KB erase and is supported in the 4 KB sector region
17Dh			F3h	
17Eh	(SA) + BFh		01h	
17Fh			00h	
180h	(SA) + C0h	JEDEC Sector Map Parameter DWORD-13 Config-0 Region-1	F8h	Region 1 of 128 KB sector Bits 31:8 = Region size = 0001F3h: Region size as count-1 of 256 Byte units = 1 x 128 KB sectors = 128 KB Count = 128 KB/256 = 500, value = count -1 = 500-1 = 499 = 1F3h Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 1b ---Erase Type 4 is 256 KB erase and is supported in the 128 KB sector region Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is not defined Bit 1 = Erase Type 2 support = 0b ---Erase Type 2 is not defined Bit 0 = Erase Type 1 support = 0b --- Erase Type 1 is 4 KB erase and is not supported in the 4 KB sector region
181h			F3h	
182h	(SA) + C1h		01h	
183h			00h	

Table 14. JEDEC SFDP Rev D, Sector Map Parameter Table (Continued)

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
184h	(SA) + C2h	JEDEC Sector Map Parameter DWORD-14 Config-0 Region-2	F8h	Region 2 Uniform 256 KB sectors Bits 31:8 = 512 Mb device Region size = 03E417h: Region size as count-1 of 256 Byte units = 255 x 256 KB sectors = 65,280 KB Count = 65,280 KB/256 = 255,000 value = count -1 = 255,000-1 = 254,999= 3E417h Bits 31:8 = 1 Gb device Region size = 01FEFFh: Region size as count-1 of 256 Byte units = 511 x 256 KB sectors = 130,816 KB Count = 130,816 KB/256 = 511,000, value = count -1 = 511,000-1 = 510,999 = 7CC17h Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 1b ---Erase Type 4 is 256 KB erase and is supported in the 256 KB sector region Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is not defined Bit 1 = Erase Type 2 support = 0b ---Erase Type 2 is not defined Bit 0 = Erase Type 1 support = 0b --- Erase Type 1 is 4 KB erase and is not supported in the 256 KB sector region
185h			17h	
186h	(SA) + C3h		E4h(512 Mb) CCh (1 Gb)	
187h			03h (512 Mb) 07h (1 Gb)	
188h	(SA) + C4h	JEDEC Sector Map Parameter DWORD-15 Config-3 Header	FEh	Configuration Index 03h 4 KB sectors at Top with remainder 256 KB Bits 31:24 = RFU = FFh Bits 23:16 = Region count (DWORDs -1) = 02h: Three regions Bits 15:8 = Configuration ID = 03h: 4 KB sectors at top with remainder 256 KB sectors Bits 7:2 = RFU = 111111b Bit 1 = Map Descriptor = 1 Bit 0 = Not the end descriptor = 0
189h			03h	
18Ah	(SA) + C5h		02h	
18Bh			FFh	
18Ch	(SA) + C6h	JEDEC Sector Map Parameter DWORD-16 Config-3 Region-0	F8h	Region 0 Uniform 256 KB sectors Bits 31:8 = 512 Mb device Region size = 03E417h: Region size as count-1 of 256 Byte units = 255 x 256 KB sectors = 65,280 KB Count = 65,280 KB/256 = 255,000 value = count -1 = 255,000-1 = 254,999 = 3E417h Bits 31:8 = 1 Gb device Region size = 01FEFFh: Region size as count-1 of 256 Byte units = 511 x 256KB sectors = 130,816KB Count = 130,816KB/256 = 511,000, value = count -1 = 511,000-1 = 510,999 = 7CC17h Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 1b ---Erase Type 4 is 256 KB erase and is supported in the 256 KB sector region Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is not defined Bit 1 = Erase Type 2 support = 0b ---Erase Type 2 is not defined Bit 0 = Erase Type 1 support = 0b ---Erase Type 1 is 4 KB erase and is not supported in the 256 KB sector region
18Dh			17h	
18Eh	(SA) + C7h		E4h(512 Mb) CCh (1 Gb)	
18Fh			03h (512 Mb) 07h (1 Gb)	
190h	(SA) + C8h	JEDEC Sector Map Parameter DWORD-17 Config-3 Region-1	F8h	Region 1 of 128 KB sector Bits 31:8 = Region size = 0001F3h: Region size as count-1 of 256 Byte units = 1 x 128 KB sectors = 128 KB Count = 128 KB/256 = 500, value = count -1 = 500-1 = 499 = 1F3h Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 1b ---Erase Type 4 is 256 KB erase and is supported in the 128 KB sector region Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is not defined Bit 1 = Erase Type 2 support = 0b ---Erase Type 2 is not defined Bit 0 = Erase Type 1 support = 0b ---Erase Type 1 is 4 KB erase and is not supported in the 4 KB sector region
191h			F3h	
192h	(SA) + C9h		01h	
193h			00h	
194h	(SA) + CAh	JEDEC Sector Map Parameter DWORD-18 Config-3 Region-2	F1h	Region 2 of 4 KB sectors Bits 31:8 = Region size (thirty-two 4 KB) = 0001F3h: Region size as count-1 of 256 Byte units = 32 x 4 KB sectors = 128 KB Count = 128 KB/256 = 500, value = count -1 = 500-1 = 499 = 1F3h Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 0b ---Erase Type 4 is not defined Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is 256 KB erase and is not supported in the 4 KB sector region Bit 1 = Erase Type 2 support = 0b ---Erase Type 2 is 64 KB erase and is not supported Bit 0 = Erase Type 1 support = 1b ---Erase Type 1 is 4 KB erase and is supported in the 4 KB sector region
195h			F3h	
196h	(SA) + CBh		01h	
197h			00h	

Table 14. JEDEC SFDP Rev D, Sector Map Parameter Table (Continued)

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
198h	(SA) + CCh	JEDEC Sector Map Parameter DWORD-19 Config-1 Header	FEh	Configuration Index 01h 4 KB sectors split between Bottom and Top with remainder 256 KB Bits 31:24 = RFU = FFh Bits 23:16 = Region count (DWORDs -1) = 04h: Five regions Bits 15:8 = Configuration ID = 01h: 4 KB sectors split between bottom and top with remainder 256 KB sectors Bits 7:2 = RFU = 111111b Bit 1 = Map Descriptor = 1 Bit 0 = Not the end descriptor = 0
199h			01h	
19Ah	(SA) + CDh		04h	
19Bh			FFh	
19Ch	(SA) + CEh	JEDEC Sector Map Parameter DWORD-20 Config-1 Region-0	F1h	Region 0 of 4 KB sectors Bits 31:8 = Region size (thirty-two 4 KB) = 0001F3h: Region size as count-1 of 256 Byte units = 32 x 4 KB sectors = 128 KB Count = 128 KB/256 = 500, value = count -1 = 500-1 = 499 = 1F3h Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 0b ---Erase Type 4 is not defined Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is 256 KB erase and is not supported in the 4 KB sector region Bit 1 = Erase Type 2 support = 0b ---Erase Type 2 is 64 KB erase and is not supported Bit 0 = Erase Type 1 support = 1b ---Erase Type 1 is 4 KB erase and is supported in the 4 KB sector region
19Dh			F3h	
19Eh	(SA) + CFh		01h	
19Fh			00h	
1A0h	(SA) + D0h	JEDEC Sector Map Parameter DWORD-21 Config-1 Region-1	F8h	Region 1 of 192KB sector Bits 31:8 = Region size = 0002EDh: Region size as count-1 of 256 Byte units = 1 x 192 KB sectors = 192 KB Count = 192 KB/256 = 750, value = count -1 = 750-1 = 749 = 2EDh Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 1b ---Erase Type 4 is 256 KB erase and is supported in the 192 KB sector region Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is not defined Bit 1 = Erase Type 2 support = 0b ---Erase Type 2 is not defined Bit 0 = Erase Type 1 support = 0b --- Erase Type 1 is 4 KB erase and is not supported in the 4 KB sector region
1A1h			EDh	
1A2h	(SA) + D1h		02h	
1A3h			00h	
1A4h	(SA) + D2h	JEDEC Sector Map Parameter DWORD-22 Config-1 Region-2	F8h	Region 2 Uniform 256 KB sectors Bits 31:8 = 512 Mb device Region size = 03E02Fh: Region size as count-1 of 256 Byte units = 254 x 256 KB sectors = 65,024 KB Count = 65,024 KB/256 = 254,000 value = count -1 = 254,000-1 = 253,999 = 3E02Fh Bits 31:8 = 1 Gb device Region size = 07C82Fh: Region size as count-1 of 256 Byte units = 510 x 256 KB sectors = 130,560 KB Count = 130,560 KB/256 = 510,000, value = count -1 = 510,000-1 = 509,999 = 7C82Fh Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 1b ---Erase Type 4 is 256 KB erase and is supported in the 256 KB sector region Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is not defined Bit 1 = Erase Type 2 support = 0b ---Erase Type 2 is not defined Bit 0 = Erase Type 1 support = 0b ---Erase Type 1 is 4 KB erase and is not supported in the 256 KB sector region
1A5h			2Fh	
1A6h	(SA) + D3h		E0h (512 Mb) C8h (1 Gb)	
1A7h			03h (512 Mb) 07h (1 Gb)	
1A8h	(SA) + D4h	JEDEC Sector Map Parameter DWORD-23 Config-1 Region-3	F8h	Region 3of 192 KB sector Bits 31:8 = Region size = 0002EDh: Region size as count-1 of 256 Byte units = 1 x 192 KB sectors = 192 KB Count = 192 KB/256 = 750, value = count -1 = 750-1 = 749 = 2EDh Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 1b ---Erase Type 4 is 256 KB erase and is supported in the 192 KB sector region Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is not defined Bit 1 = Erase Type 2 support = 0b ---Erase Type 22 is not defined Bit 0 = Erase Type 1 support = 0b ---Erase Type 1 is 4 KB erase and is not supported in the 4 KB sector region
1A9h			EDh	
1AAh	(SA) + D5h		02h	
1ABh			00h	
1ACh	(SA) + D6h	JEDEC Sector Map Parameter DWORD-24 Config-1 Region-5	F1h	Region 5 of 4 KB sectors Bits 31:8 = Region size (thirty-two 4 KB) = 0001F3h: Region size as count-1 of 256 Byte units = 32 x 4 KB sectors = 128 KB Count = 128 KB/256 = 500, value = count -1 = 500-1 = 499 = 1F3h Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 0b ---Erase Type 4 is not defined Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is 256 KB erase and is not supported in the 4 KB sector region Bit 1 = Erase Type 2 support = 0b ---Erase Type 2 is 64 KB erase and is not supported Bit 0 = Erase Type 1 support = 1b ---Erase Type 1 is 4 KB erase and is supported in the 4 KB sector region
1ADh			F3h	
1AEh	(SA) + D7h		01h	
1AFh			00h	

Table 14. JEDEC SFDP Rev D, Sector Map Parameter Table (Continued)

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
1B0h	(SA) + D8h	JEDEC Sector Map Parameter DWORD-25 Config-4 Header	FFh	Configuration Index 04h Uniform 256 KB sectors Bits 31:24 = RFU = FFh Bits 23:16 = Region count (DWORDs -1) = 00h: One region Bits 15:8 = Configuration ID = 04h: Uniform 256 KB sectors Bits 7:2 = RFU = 111111b Bit 1 = Map Descriptor = 1 Bit 1 = Not the end descriptor = 0
1B1h			04h	
1B2h	(SA) + D9h		00h	
1B3h			FFh	
1B4h	(SA) + DAh	JEDEC Sector Map Parameter DWORD-26 Config-4 Region-0	F8h	Region 0 Uniform 256 KB sectors Bits 31:8 = 512 Mb device Region size = 03E7FFh: Region size as count-1 of 256 Byte units = 256 x 256 KB sectors = 65,536 KB Count = 65,280 KB/256 = 256,000 value = count -1 = 256,000-1 = 255,999= 3E7FFh Bits 31:8 = 1 Gb device Region size = 07CFFFh: Region size as count-1 of 256 Byte units = 512 x 256 KB sectors = 131,072 KB Count = 131,072 KB/256 = 512,000, value = count -1 = 512,000-1 = 511,999 = 7CFFFh Bits 7:4 = RFU = Fh Erase Type not supported = 0 / supported = 1 Bit 3 = Erase Type 4 support = 1b ---Erase Type 4 is 256 KB erase and is supported in the 256 KB sector region Bit 2 = Erase Type 3 support = 0b ---Erase Type 3 is not defined Bit 1 = Erase Type 2 support = 0b ---Erase Type 2 is not defined Bit 0 = Erase Type 1 support = 0b ---Erase Type 1 is 4 KB erase and is not supported in the 256 KB sector region
1B5h			FFh	
1B6h	(SA) + DBh		E7h(512 Mb) CFh (1 Gb)	
1B7h			03h (512 Mb) 07h (1 Gb)	

3.5.4 xSPI Profile 2

Table 15. xSPI Profile 2 Parameters

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
1B8h	(SA) + DCh	JEDEC xSPI (Profile 2.0) DWORD 1	E0h	Bit 7 = Program Suspend Supported = 1b Bit 6 = Program Resume Supported = 1b Bit 5 = Enter SPI (1S-1S-1S) Supported = 1b Bit 4:0 = Reserved (00000b)
1B9h			FFh	Bit 15 = Deep Power Down Supported = 1b Bit 14 = Word Program Supported = 1b Bit 13 = Sector Erase Supported = 1b Bit 12 = Chip Erase Supported = 1b Bit 11 = Erase Suspend Supported = 1b Bit 10 = Erase Resume Supported = 1b Bit 9 = Write to Buffer Supported = 1b Bit 8 = Program Write to Buffer Supported = 1b
1BAh	(SA) + DDh		FFh	Bit 23 = Memory Write Linear Supported = 1b Bit 22 = WREN1 Supported = 1b Bit 21 = WREN2 Supported = 1b Bit 20 = SREN Supported = 1b Bit 19 = Status Register Read Supported = 1b Bit 18 = Status Register Clear Supported = 1b Bit 17 = Configuration Register Read Supported = 1b Bit 16 = Configuration Register Load Supported = 1b
1BBh			FFh	Bit 31 = xSPI Profile 2 Supported = 1b Bit 30 = Register Read Wrap Supported = 1b Bit 29 = Register Read Linear Supported = 1b Bit 28 = Memory Read Wrap Supported = 1b Bit 27 = Memory Read Linear Supported = 1b Bit 26 = Register Write Wrap Supported = 1b Bit 25 = Register Write Linear Supported = 1b Bit 24 = Memory Write Wrap Supported = 1b
1BCh	(SA) + DEh	JEDEC xSPI (Profile 2.0) DWORD 2	2Ch	Bits 15:12 = Reserved = 0000b Bits 11:7 = 200 MHz 16 Dummy Cycles = 10000b Bits 6:2 = 200MHz Configuration Bit Pattern = 01011b Bits 1:0 = Reserved = 00b
1BDh			08h	
1BEh	(SA) + DFh		00h	Bits 31:16 = Reserved = 00000000000000000b
1BFh			00h	

Table 15. xSPI Profile 2 Parameters (Continued)

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
1C0h	(SA) + E0h	JEDEC xSPI (Profile 2.0) DWORD 3	0Ch	Bits 31:27 = 166MHz 14 Dummy Cycles = 01110b Bits 26:22 = 166MHz Configuration Bit Pattern = 01001b Bits 21:17 = 133MHz 12 Dummy Cycles = 01100b Bits 16:12 = 133MHz Configuration Bit Pattern = 00111b Bits 11:7 = 100MHz 8 Dummy Cycles = 01000b Bits 6:2 = 100MHz Configuration Bit Pattern = 00011b Bits 1:0 = Reserved = 00b
1C1h			74h	
1C2h	(SA) + E1h		58h	
1C3h			72h	

Table 16. Command Sequences to Change to Profile 2.0 Mode Parameters

Byte Address (x1 SPI)	Word Address (x8 HB)	SFDP DWORD Name	Data	Description
1C4h	(SA) + E2h	Command Sequences Profile Change DWORD 1	00h	Bits 31:24 = 01h Bits 23:16 = 06h Bits 15:8 = 00h Bits 7:0 = 00h
1C5h			00h	
1C6h	(SA) + E3h		06h	
1C7h			01h	
1C8h	(SA) + E4h	Command Sequences Profile Change DWORD 2	00h	Bits 31:24 = 00h Bits 23:16 = 00h Bits 15:8 = 00h Bits 7:0 = 00h
1C9h			00h	
1CAh	(SA) + E5h		00h	
1CBh			00h	
1CCh	(SA) + E6h	Command Sequences Profile Change DWORD 3	80h	Bits 31:24 = 06h Bits 23:16 = 71h Bits 15:8 = 00h Bits 7:0 = 80h
1CDh			00h	
1CEh	(SA) + E7h		71h	
1CFh			06h	
1D0h	(SA) + E8h	Command Sequences Profile Change DWORD 4	00h	Bits 31:24 = 00h Bits 23:16 = 04h Bits 15:8 = 02h Bits 7:0 = 00h
1D1h			02h	
1D2h	(SA) + E9h		04h	
1D3h			00h	

3.6 Secure SSR Address Space

Each HL-T/HS-T family memory device has a 1024-byte SSR which is OTP address space. This address space is separate from the main flash 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 bytes contain a 128-bit random number. The random number cannot be written to, erased or programmed and any attempts will return an PRGERR flag.
- The next four bytes are used to provide one bit per secure region (32 bits in total) to permanently protect once set to “0” from writing, erasing or programming.
- All other bytes are reserved.

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

Table 17. SSR Address Map

Region	Byte Address Range	Contents	Initial Delivery State
Region 0	000h	LSB of Cypress Programmed Random Number	Cypress Programmed Random Number
	...	...	
	00Fh	MSB of Cypress Programmed Random Number	
	010h to 013h	Region Locking Bits Byte 10h [bit 0] locks region 0 from programming when = 0 ... Byte 13h [bit 7] locks region 31 from programming when = 0	All Bytes = FFh
	014h to 01Fh	Reserved for Future Use (RFU)	All Bytes = FFh
Region 1	020h to 03Fh	Available for User Programming	
Region 2	040h to 05Fh		
...	...		
Region 31	3E0h to 3FFh		

3.7 Registers (Legacy (x1) SPI Only)

Registers are small groups of memory cells used to configure how the HL-T/HS-T family memory device operates, or to report the status of device operations. The registers are accessed by specific commands and addresses. Table 18 shows the address map for every available register in Legacy (x1) SPI in this flash memory device.

Table 18. Register Address Map

Function	Register Type	Register Name	Volatile Component Address (hex)	Nonvolatile Component Address (hex)
Device Status	Status Register 1	STR1N[7:0], STR1V[7:0]	0x00800000	0x00000000
	Status Register 2	STR2V[7:0]	0x00800001	N/A
Device Configuration	Configuration Register 1	CFR1N[7:0], CFR1V[7:0]	0x00800002	0x00000002
	Configuration Register 2	CFR2N[7:0], CFR2V[7:0]	0x00800003	0x00000003
	Configuration Register 3	CFR3N[7:0], CFR3V[7:0]	0x00800004	0x00000004
	Configuration Register 4	CFR4N[7:0], CFR4V[7:0]	0x00800005	0x00000005
Interface CRC	Interface CRC Enable Register	ICEV[7:0]	0x00800008	N/A
EnduraFlex Architecture	EnduraFlex Architecture Selection Register 0 [7:0]	EFX0O[7:0]	N/A	0x00000050
	EnduraFlex Architecture Selection Register 0 [15:8]	EFX0O[15:8]		0x00000051
	EnduraFlex Architecture Selection Register 1 [7:0]	EFX1O[7:0]		0x00000052
	EnduraFlex Architecture Selection Register 1 [15:8]	EFX1O[15:8]		0x00000053
	EnduraFlex Architecture Selection Register 2 [7:0]	EFX2O[7:0]		0x00000054
	EnduraFlex Architecture Selection Register 2 [15:8]	EFX2O[15:8]		0x00000055
	EnduraFlex Architecture Selection Register 3 [7:0]	EFX3O[7:0]		0x00000056
	EnduraFlex Architecture Selection Register 3 [15:8]	EFX3O[15:8]		0x00000057
	EnduraFlex Architecture Selection Register 4 [7:0]	EFX4O[7:0]		0x00000058
	EnduraFlex Architecture Selection Register 4 [15:8]	EFX4O[15:8]		0x00000059

Table 18. Register Address Map (Continued)

Function	Register Type	Register Name	Volatile Component Address (hex)	Nonvolatile Component Address (hex)
Interrupt Pin	Interrupt Configuration Register	INCV[7:0]	0x00800068	N/A
	Interrupt Status Register	INSV[7:0]	0x00800067	
Error Correction	ECC Status Register	ESCV[7:0]	0x00800089	
	ECC Error Detection Count Register [7:0]	ECTV[7:0]	0x0080008A	
	ECC Error Detection Count Register [15:8]	ECTV[15:8]	0x0080008B	
	ECC Address Trap Register [7:0]	EATV[7:0]	0x0080008E	
	ECC Address Trap Register [15:8]	EATV[15:8]	0x0080008F	
	ECC Address Trap Register [23:16]	EATV[23:16]	0x00800040	
	ECC Address Trap Register [31:24]	EATV[31:24]	0x00800041	
AutoBoot	AutoBoot Register [7:0]	ATBN[7:0]	N/A	0x00000042
	AutoBoot Register [15:8]	ATBN[15:8]		0x00000043
	AutoBoot Register [23:16]	ATBN[23:16]		0x00000044
	AutoBoot Register [31:24]	ATBN[31:24]		0x00000045
Erase Count	Sector Erase Count Register [7:0]	SECV[7:0]	0x00800091	N/A
	Sector Erase Count Register [15:8]	SECV[15:8]	0x00800092	
	Sector Erase Count Register [23:16]	SECV[23:16]	0x00800093	
Data Integrity Check	Data Integrity Check CRC Register [7:0]	DCRV[7:0]	0x00800095	
	Data Integrity Check CRC Register [15:8]	DCRV[15:8]	0x00800096	
	Data Integrity Check CRC Register [23:16]	DCRV[23:16]	0x00800097	
	Data Integrity Check CRC Register [31:24]	DCRV[31:24]	0x00800098	
Protection and Security	Advanced Sector Protection Register [7:0]	ASPO[7:0]	N/A	0x00000030
	Advanced Sector Protection Register [15:8]	ASPO[15:8]		0x00000031
	ASP PPB Lock Register (Persistent Protection Block)	PPLV[7:0]	0x0080009B	N/A
	ASP Password Register [7:0]	PWDO[7:0]	N/A	0x00000020
	ASP Password Register [15:8]	PWDO[15:8]		0x00000021
	ASP Password Register [23:16]	PWDO[23:16]		0x00000022
	ASP Password Register [31:24]	PWDO[31:24]		0x00000023
	ASP Password Register [39:32]	PWDO[39:32]		0x00000024
	ASP Password Register [47:40]	PWDO[47:40]		0x00000025
	ASP Password Register [55:48]	PWDO[55:48]		0x00000026
	ASP Password Register [63:56]	PWDO[63:56]		0x00000027

4. Features

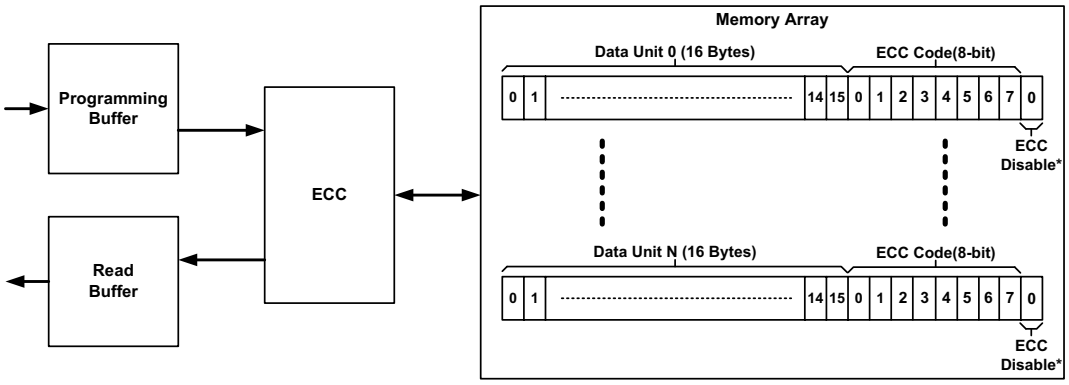
4.1 Error Detection and Correction

Semper Flash family devices support error detection and correction by generating an embedded Hamming error correction code during memory array programming. This ECC code is then used for error detection and correction during read operations. The ECC is based on a 16-byte data unit. When the 16-byte data unit is loaded into the Program Buffer and is transferred to the 128-bits flash memory array Line for programming (after an erase), an 8-bit Error Correction Code (ECC) for each data unit is also programmed into a portion of the memory array that is not visible to the host system software. This ECC information is then checked during each Flash array read operation. Any 1-bit error within the data unit will be corrected by the ECC logic. The 16-byte data unit is the smallest program granularity on which ECC is enabled.

When any amount of data is first programmed within a 16-byte data unit, the ECC value is set for the entire data unit. If additional data is subsequently programmed into the same data unit, without an erase, then the ECC for that data unit is disabled and the 1-bit ECC disable bit is set. A sector erase is needed to again enable ECC on that data unit.

These are automatic operations transparent to the user. The transparency of the ECC feature enhances data reliability for typical programming operations which write data once to each data unit while also facilitating software compatibility with previous generations of products by still allowing for single-byte programming and bit-walking (in this case, ECC will be disabled) in which the same data unit is programmed more than once.

Figure 19. 16-Byte ECC Data Unit Example



* 9th ECC Code bit in case of 2-bit Error Detection Enabled

Semper NOR Flash supports 2-bit error detection as the default ECC configuration. In this configuration, any 1-bit error in a data unit is corrected and any 2-bit error is detected and reported. The 16-byte unit data requires a 9-bit Error Correction Code for 2-bit error detection. When 2-bit error detection is enabled, byte-programming, bit-walking, or multiple program operations to the same data unit (without an erase) are not allowed and will result in a Program Error. Changing the ECC mode from 1-bit error detection to 2-bit error detection, or from 2-bit error detection to 1-bit error detection will invalidate all the ECC syndromes related to data in the memory array. When changing the ECC mode, the host must first erase all sectors in the device. If the ECC mode is changed without erasing programmed data, subsequent read operations will result in undefined behavior.

4.1.1 ECC Error Reporting

There are five methods for reporting to the host system when ECC errors are detected.

- ECC Data Unit Status provides the status of 1-bit or 2-bit errors in data units.
- The Address Trap Register captures the address location of the first ECC error encountered after POR or reset during memory array read.
- An ECC Error Detection counter keeps a tally of the number of 1-bit or 2-bit errors that have occurred in data units during reads.
- The Interrupt (INT#) output can be enabled to indicate when either a 1-bit or 2-bit error is detected as data is read.
- In HyperBus interface, a mode may be enabled to cause the Data Strobe (DS) to stop toggling (stall) when reading a Half-page containing a 2-bit error. The stall condition can be detected by the Semper Flash master as a bus error when DS does not transition for more than 32 clock cycles. The Data Strobe Stall control bit in CFRIx[2] can be used to enable DS stall. If enabled (CFRIx[2] = 0), upon 2-bit error, the DS will be driven Low after 2 clock cycles. DS will remain in the Low state as long as CS# remains asserted, normal DS functionality resumes as soon as CS# returns High. If the DS Stall Control bit is in the disabled state (CFRIx[2] = 1), DS behavior is not impacted.

4.1.1.1 ECC Data Unit Status (EDUS)

- The status of ECC in each data unit is provided by the 16-bit ECC Data Unit Status for HyperBus and 8-bit ECC Data Unit Status for Legacy (x1) SPI.
- The contents of the ECC Data Unit status indicate, for the selected data unit, whether there is a 1-bit error corrected, 2-bit error detected or the ECC is disabled for that data unit.

Table 19. ECC Data Unit Status

Bits	Field Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EDUS[15:4]	RESRVD	Reserved For Future Use	V => R	0000	These bits are Reserved for future use. **Note** For Legacy (x1) SPI Only, ECC status is only 8 bits.
EDUS[3]	ECC2BD	ECC Error 2-bit Error Detection Flag	V => R	0	This bit indicates whether a two bit error is detected in the data unit, if two bit ECC error detection is enabled CFR4V[3] = 1. Single byte programming and bit walking in which the same data unit is programmed more than once is not allowed within the same data unit, because ECC cannot be turned off and ECCOFF bit will always be “0” and will result in a Program Error. When CFR4V[3] = 0 and 2-bit error detection is disabled, ECC2BD bit will alway be “0”. Selection Options: 1 = Two Bit Error detected 0 = No error
EDUS[2]	RESRVD	Reserved For Future Use	V => R	0	This bit is Reserved for future use.
EDUS[1]	ECC1BC	ECC Error 1-bit Error Detection and Correction Flag	V => R	0	This bit indicates whether an error was corrected in the data unit. Selection Options: 1 = Single Bit Error corrected in the addressed data unit 0 = No single bit error was corrected in the addressed data unit
EDUS[0]	ECCOFF	Data Unit ECC Off/On Flag	V => R	0	This bit indicates whether the ECC syndrome is OFF in the data unit. Selection Options: 1 = ECC is OFF in the selected data unit 0 = ECC is ON in the selected data unit

4.1.1.2 ECC Error Address Trap (EATV)

- A 32-bit register is provided to capture the ECC data unit address where an ECC error is first encountered during a read of the flash array. Only the address of the first enabled error type (“2-bit only” or “1-bit or 2-bit” as selected in CFR2x[6]) encountered after POR, hardware reset, or the ECC Clear transaction is captured. The EATV Register is only updated during Read transactions.

  The EATV Register contains the address that was accessed when the error was detected. The failing bits may not be located at the exact address indicated in the register, but will be located within the aligned 16-byte ECC data unit where the error was detected. If errors are found in multiple ECC data units during a single read operation, only the address of the first failing ECC unit address is captured in the EATV Register.

  When 2-bit error detection is not enabled and the same ECC unit is programmed more than once, ECC error detection for that ECC unit is disabled, therefore no error is recognized to trap the address.

- The Address Trap Register can be read using HyperBus or Legacy (x1) SPI Address Trap Register read transaction.

- Clear ECC Status Register transaction, POR, or CS# Signaling/Hardware/Software reset clears the Address Trap Register.

Table 20. Error Upper / Lower Address Trap Register Bit Assignments

Density	Error Lower Address Register	Error Upper Address Register		
	All	256 Mb	512 Mb	1 Gb
EATV[15]	A15	0	0	0
EATV[14]	A14	0	0	0
EATV[13]	A13	0	0	0
EATV[12]	A12	0	0	0
EATV[11]	A11	0	0	0
EATV10]	A10	0	0	0
EATV[9]	A9	0	0	A25
EATV[8]	A8	0	A24	A24
EATV[7]	A7	A23	A23	A23
EATV[6]	A6	A22	A22	A22
EATV[5]	A5	A21	A21	A21
EATV[4]	A4	A20	A20	A20
EATV[3]	A3	A19	A19	A19
EATV[2]	A2	A18	A18	A18
EATV[1]	A1	A17	A17	A17
EATV[0]	A0	A16	A16	A16

4.1.1.3 ECC Error Detection Counter (ECTV)

- A 16-bit register is provided to count the number of 1-bit or 2-bit errors that occur as data is read from the flash memory array. Only errors recognized in the main array will cause the Error Detection Counter to increment. ECTV Register is only updated during Read transaction. Read ECC Status transaction does not affect the ECTV Register.

  The 16-bit Error Detection Counter will not increment beyond FFFFh. However, the ECC continues to work.

  Note that during continuous read operations, when a 1-bit or a 2-bit error is detected, the clock may continue toggling and the memory device will continue incrementing the data address and placing new data on the DQ signals; any additional data units with errors that are encountered will be counted until CS# is brought back HIGH.

  During a read transaction only one error is counted for each data unit found with an error. Each read transaction will cause a new read of the target data unit. If multiple read transactions access the same data unit containing an error, the error counter will increment each time that data unit is read.

  When 2-bit error detection is not enabled and the same data unit is programmed more than once, ECC error detection for that data unit is disabled so, no error can be recognized or counted.

- The ECC Error Detection Counter Register can be read using HyperBus or Legacy (x1) SPI ECC Error Detection Counter read transaction.

- ECTV Register is set to 0 on POR, CS# Signaling/Hardware/Software Reset or with Clear ECC Status Register transaction.

Table 21. Error Detection Counter (Volatile)

Bit Position	EDC Result
EDC[15]	R15
EDC[14]	R14
EDC[13]	R13
EDC[12]	R12
EDC[11]	R11
EDC[10]	R10
EDC[9]	R9
EDC[8]	R8
EDC[7]	R7
EDC[6]	R6
EDC[5]	R5
EDC[4]	R4
EDC[3]	R3
EDC[2]	R2
EDC[1]	R1
EDC[0]	R0

4.1.1.4 INT# Output - HyperBus only

- Semper Flash supports INT# output pin to indicate to the host system that an event has occurred within the flash device. The user can configure the INT# output pin to transition to the active (Low) state when:
  - 2-bit ECC error is detected
  - 1-bit ECC error is detected
  - Transitioning from the Busy to the Ready state

  The INT# pin is only available in BGA package. Operation is controlled with the Interrupt Configuration Register (INCV) 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 (INSV) indicates what enabled internal event(s) have occurred since the last time the INSV was 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 INSV Register can be read to determine which internal event was responsible.

  INT# output status during POR, Hardware Reset, Software Reset, DPD Exit, or CS# Signaling Reset is not valid.
- The INCV and INSV can be accessed through Program/Read INCV/INSV HyperBus transactions.
- The INT# output can be forced to transition back to the High state (returned High by an external pull-up resistance) using the following methods:
  - Disable the INT# output by loading a ‘1’ into bit 15 of the Interrupt Configuration Register.
  - Reset the appropriate bit (by writing a ‘1’) in the INSV bit that indicates which internal event occurred to cause the output to go Low. All INSV bits that are Low and are also enabled in the INSV must be reset before the INT# output will return HIGH.
  - If the internal event is ECC, then issuing Clear ECC Errors transaction while in the ECC ASO will return INT# output HIGH. Clear ECC Errors clears INSV ECC related bits (INSV[1:0]) but does not disable INCV.
  - The INT# output will also be returned to the default (disabled, High-Z) state with CS# Signaling Reset, Hardware Reset (RESET# = Low) or a POR. Hardware Reset and POR disable all interrupts by setting the Interrupt Configuration Register back to the default (all interrupts disabled) state.

4.1.2 HyperBus ECC Related Registers and Transactions

Table 22. HyperBus ECC Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
Configuration Register - 2 (CFR2N, CFR2V)	Read Configuration Register 2 (RDVCR2_4_0, RDNCR2_4_0)
ECC Address Trap Register (EATV)	Read Address Trap Register (RDADTU_2_1, RDADTL_2_1) - ECC ASO
ECC Error Detection Counter Register (ECTV)	Read ECC Count Value Register (RDCONT_2_1) - ECC ASO
INT# Pin Configuration Register (INCV)	Read Interrupt Configuration Register (RDVINC_4_0)
INT# Pin Status Register (INSV)	Read Interrupt Status Register (RDVINS_4_0)
	Program Interrupt Configuration Register (PGVINC_4_0)
	Program Interrupt Status Register (PGVINC_4_0)

4.1.3 Legacy (x1) SPI ECC Related Registers and Transactions

Table 23. Legacy (x1) SPI ECC Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
Configuration Register - 2 (CFR2N, CFR2V)	Read Any Register (RDARG_4_0)
ECC Address Trap Register (EATV)	Write Any Register (WRARG_C_1)
ECC Error Detection Counter Register (ECTV)	Read ECC Status (RDECC_4_0)

4.2 EnduraFlex Architecture (Wear Leveling)

EnduraFlex Architecture allows partitioning of the main memory array into regions which can be configured as either high endurance or long retention. EnduraFlex implements wear leveling in high endurance regions where program/erase cycles are spread evenly across all the sectors which are part of the wear leveling pool. This greatly improves the reliability of the device by avoiding premature wear-out of an individual sector.

Architecturally, EnduraFlex's wear leveling algorithm is based on a mapping of logical sectors to physical sectors. During the lifetime of the part, this mapping is changed to maintain a uniform distribution of program/erase cycles over all physical sectors. The logical to physical mapping information is stored in a dedicated flash array which is updated when sectors are swapped. Sector swaps occur when an erase transaction is given.

EnduraFlex's high endurance region requires a minimum set of 20 sectors. To provide flexibility between configuring long retention, high endurance, or both regions, a four pointer architecture is provided. The factory default setting designates all sectors as high endurance as part of the wear leveling pool with all pointers disabled. The four pointers can be used to form a maximum of five regions which can each be configured as long retention or high endurance.

Figure 20 provides an overview of the EnduraFlex Architecture. It shows the five possible regions based on different sector architecture.

Note 4 KB sectors are not part of the EnduraFlex Architecture.

Figure 20. EnduraFlex™ Architecture Overview

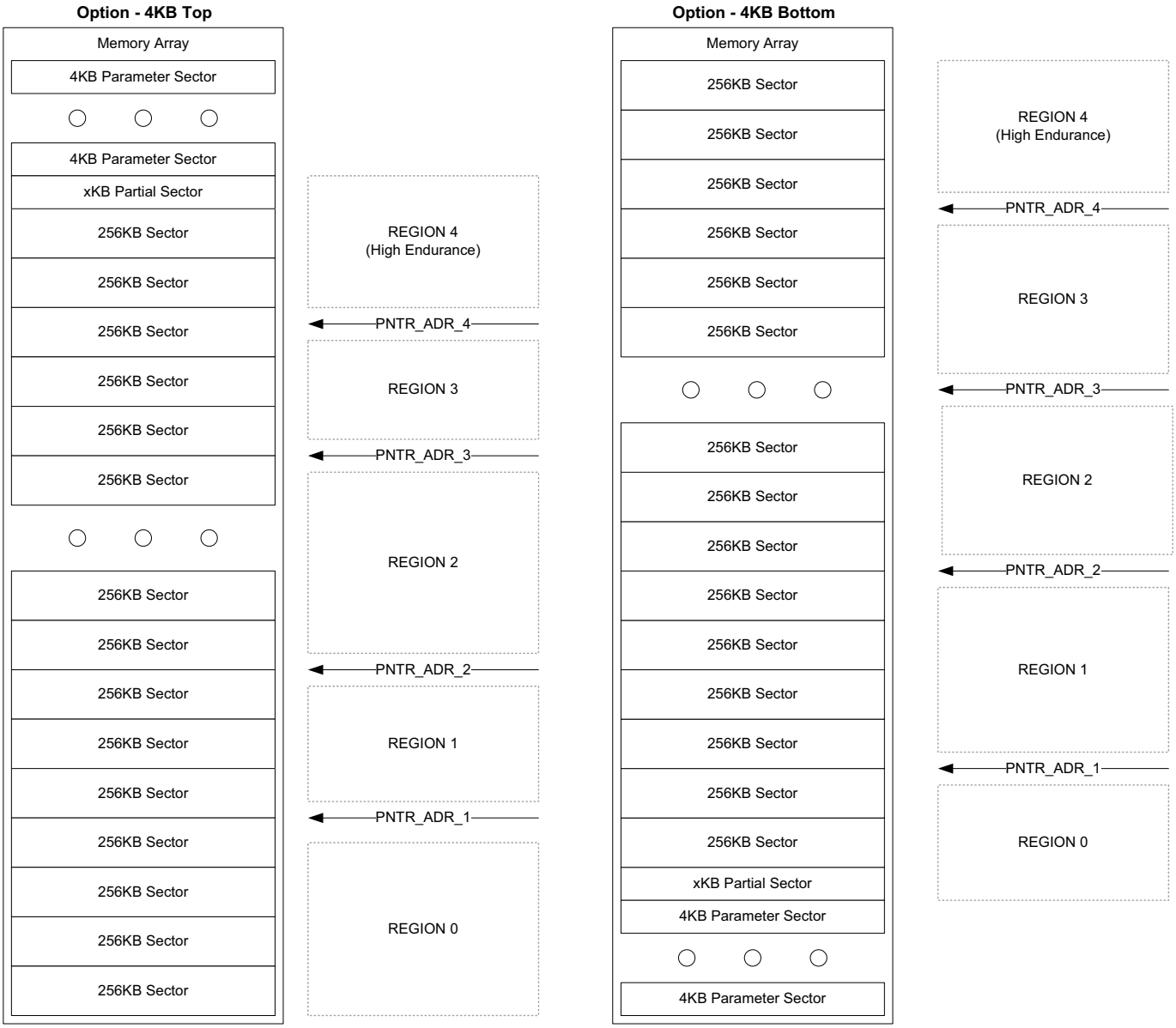


Figure 20. EnduraFlex™ Architecture Overview (Continued)

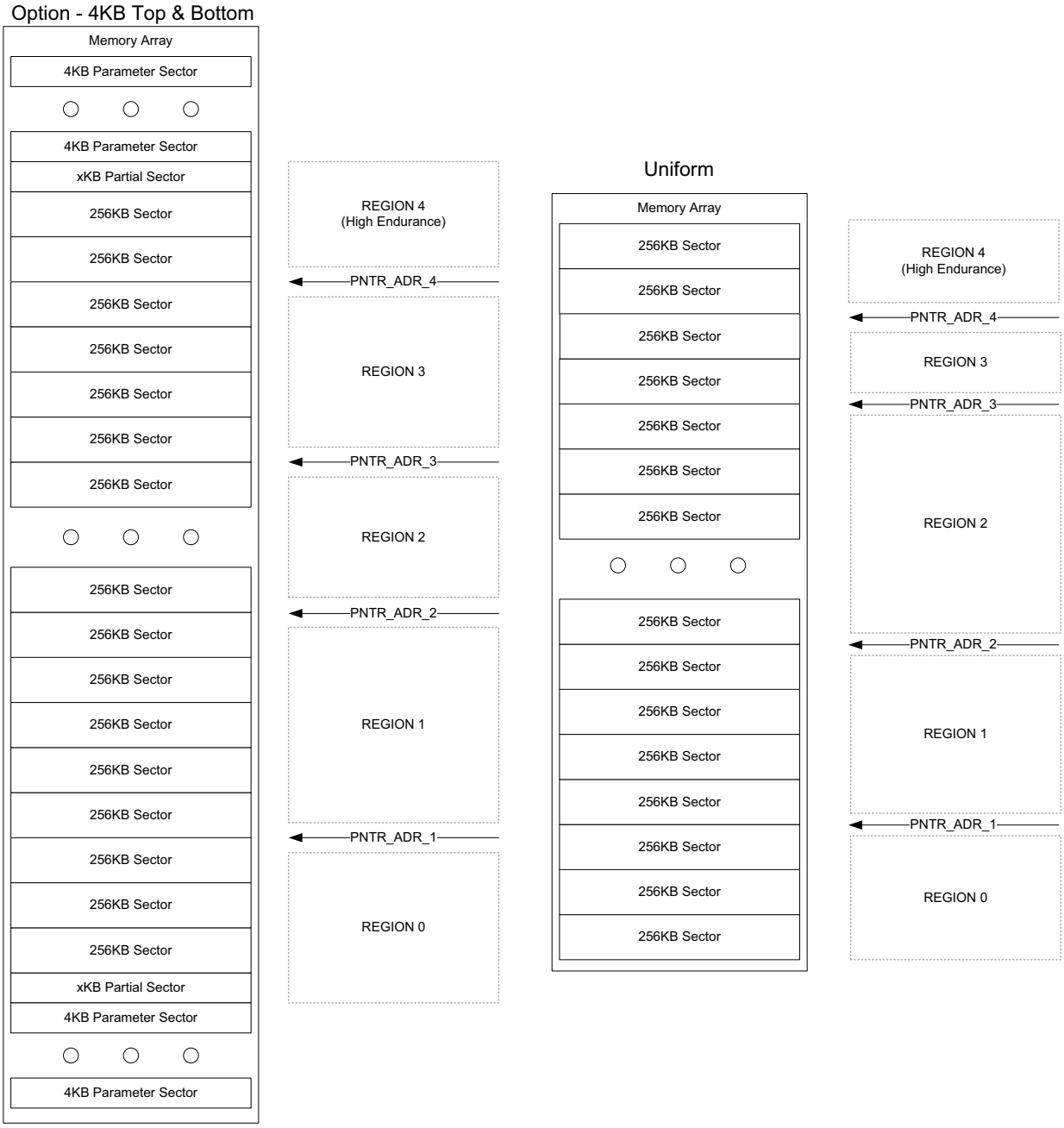


Table 24. Region Definitions[16, 17, 18, 19]

Region	Lower Limit	Upper Limit
0	Sector 0	Address Pointer 1
1	Address Pointer 1	Address Pointer 2
2	Address Pointer 2	Address Pointer 3
3	Address Pointer 3	Address Pointer 4
4	Address Pointer 4	Highest Sector

Notes
16. The pointer addresses must obey the following rules:
    Pointer#4 address > Pointer#3 address
    Pointer#3 address > Pointer#2 address
    Pointer#2 address > Pointer#1 address
17. 4 KB sectors are excluded.
18. It is required that the high data endurance and long data retention regions are configured at the time the device is first powered-up by the customer. Once configured, they can never be changed again.
19. The minimum size of any high endurance region is 20 sectors.

4.2.1 Configuration 1: Maximum Endurance - Single High Endurance Region

Maximum endurance is achieved when all 256 KB sectors are designated as high endurance. All sectors must be designated as high endurance using the EnduraFlex pointer architecture. Maximum endurance pointer configuration is shown in Table 25.

Table 25. EnduraFlex Pointer Values for Maximum Endurance Configuration[20]

Pointer #	Pointer Address EPTADn[8:0]	Region Type ERGNTn	Pointer Enable# EPTEBn	Global Region Selection GBLSEL	Wear Leveling Enable WRLVEN
0	N/A	N/A	N/A	1'b1	1'b1
1	9'b111111111	1'b1	1'b1	N/A	N/A
2	9'b111111111				
3	9'b111111111				
4	9'b111111111				

Note
20. This is also the default configuration of the device.

4.2.2 Configuration 2: Two Region Selection - One Long Retention Region and One High Endurance Region

Sectors for long retention or high endurance must be delineated using the EnduraFlex pointer architecture. Region 0 is defined as long retention and consists of 16 sectors. Region 1 is defined as high endurance and has 240 sectors. The pointer setup for two region configuration is shown in Table 26.

Table 26. EnduraFlex Pointer Values for Two Region Configuration[21]

Pointer #	Pointer Address EPTADn[8:0]	Region Type ERGNTn	Pointer Enable# EPTEBn	Global Region Selection GBLSEL	Wear Leveling Enable WRLVEN
0	N/A	N/A	N/A	1'b0	1'b1
1	9'b000000111	1'b0	1'b1	N/A	N/A

Note
21. The number of pointers defined is based on the number of regions configured.

4.2.3 HyperBus EnduraFlex Related Registers and Transaction

Table 27. HyperBus EnduraFlex Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
EnduraFlex Architecture Selection Registers (EFX4O,EFX3O,EFX2O,EFX1O,EFX0O)	Read EnduraFlex Registers (RDOENX_1_1) - EnduraFlex ASO
	Program EnduraFlex Registers (PGOENX_2_1) - EnduraFlex ASO

4.2.4 Legacy (x1) SPI EnduraFlex Related Registers and Transaction

Table 28. Legacy (x1) SPI EnduraFlex Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
EnduraFlex Architecture Selection Registers (EFX4O,EFX3O,EFX2O,EFX1O,EFX0O)	Read Any Register (RDARG_4_0)
	Write Any Register (WRARG_C_1)

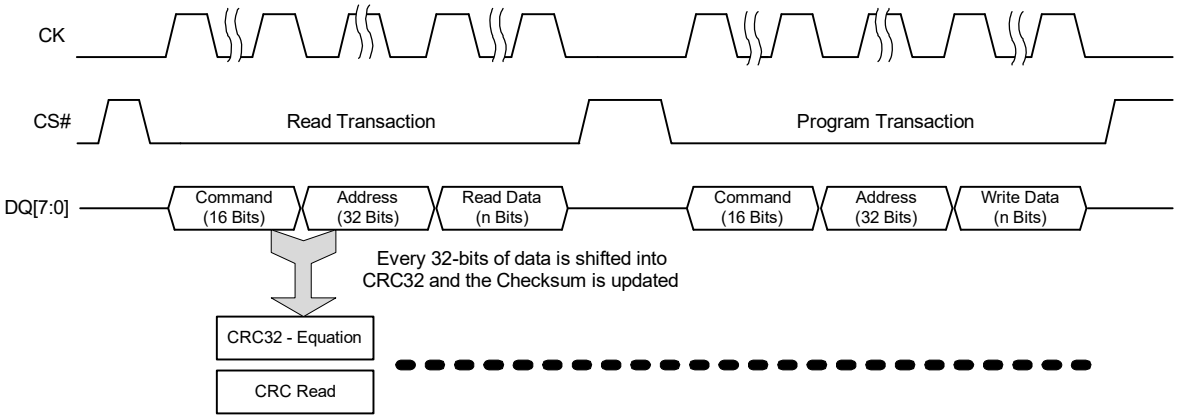
4.3 Interface CRC

Interface CRC performs a hardware accelerated CRC calculation on the communication between a host and the device, ensuring the integrity of information transferred. A CRC is an error-detecting code commonly used in devices to detect accidental changes to raw data.

The Interface CRC method in HL-T/HS-T family devices relies entirely on the host to verify the CRC check-value and take appropriate actions. The device calculates the CRC check-value which the host reads using the Read Interface CRC transaction (RDCRC_4_0, RICRC_1_1). The check-value calculated includes all transaction contents while CS# is LOW, namely command, address and data. This CRC checksum can be generated across either a single transaction or a set of transactions. The only limitation is that the data size over which the slave is calculating the CRC checksum must be less than 2^{32} bits.

The host must also calculate the CRC check-value over the same transaction sequence. When ready, the host can read the device's calculated CRC check-value and compare it with its own. If there is a mismatch, the host can choose to repeat the complete transaction sequence.

Figure 21. CRC Calculation Overview



Note At the end of the CRC read transaction, the device resets the CRC check-value and reinitializes the CRC polynomial.

CRC32 Polynomial: $X^{32} + X^{28} + X^{27} + X^{26} + X^{25} + X^{23} + X^{22} + X^{20} + X^{19} + X^{18} + X^{14} + X^{13} + X^{11} + X^{10} + X^{9} + X^{8} + X^{6} + 1$

Note The CRC polynomial between the host and the device must be identical.

Note The Interface CRC check-value will reset to 0xFFFFFFFFh under the following conditions:

- POR
- Hardware Reset
- Software Reset
- CS# Signaling Reset
- A read of the Interface CRC check-value
- Exit from Deep Power Down

Note If a transaction is aborted before the command is legally received, i.e. the transfer length is cut short by CS# deasserting early - the transferred data will still be clocked into the CRC check-value, but it is no longer guaranteed. When using Interface CRC, only valid, non-aborted transactions must be used.

Note It is required to read Interface CRC value before any Volatile Status Register read and Clear Interface CRC value after any volatile Status Register read(s).

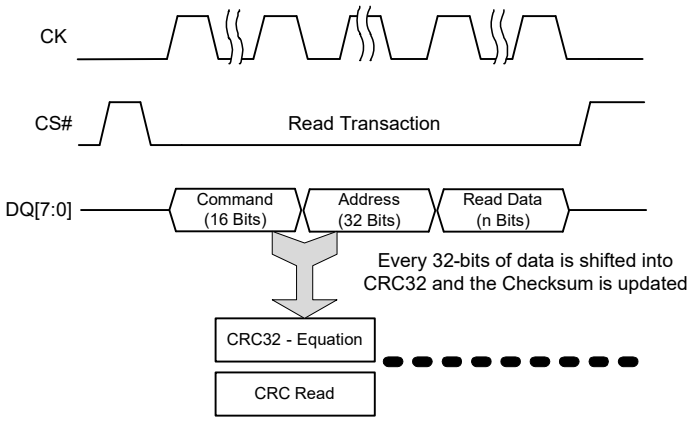
Note When Interface CRC is disabled, the interface CRC Register value becomes indeterminate. It is recommended to read the interface CRC Register before disabling the interface CRC feature, and again after enabling the interface CRC feature to re-initialize the CRC calculation.

4.3.1 Read

The read operation is performed when the host specifies the READ transaction while CS# is LOW. The device then provides the data from the memory based on the address. Any number of bytes can be read (burst reads) to consecutive addresses without issuing a new READ transaction.

For transaction protection, the device performs the CRC over the entire transaction sequence (CS# LOW state) using the CRC32 polynomial. Once the CS# is brought HIGH, the CRC calculation is stopped and the check-value latched into the CRC Register. If multiple READ transactions are executed by the host, the device continues updating the CRC check-value between every CS# LOW cycle.

Figure 22. Read CRC Protection



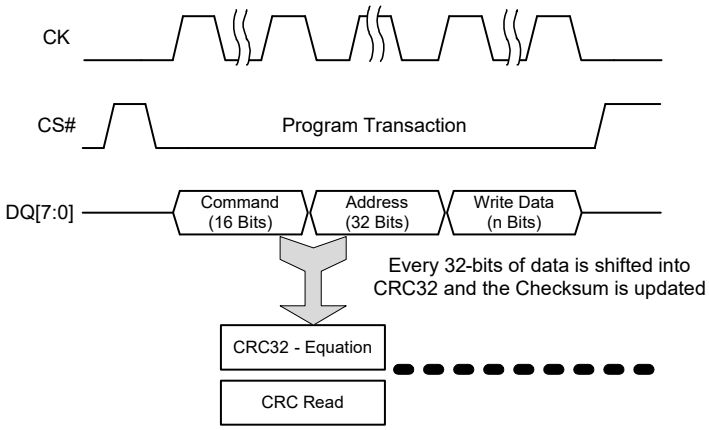
Note Back to back Interface CRC read transaction will not show the CRC checksum value being reset. At the end of each read interface CRC Register transaction, the interface CRC Register will get reset and updates itself with new CRC checksum value after getting a transaction with valid input data for at least three clock cycles.

4.3.2 Program / Erase

The program operation is performed when the host specifies a program transaction while CS# is LOW. Up to 256 bytes / 512 bytes can be written (burst writes) to consecutive addresses without issuing new program transaction. The erase operation is performed when the host specifies an erase transaction while CS# is LOW. Either a single sector or the complete device can be erased.

For transaction protection, the slave device will perform the CRC over the entire instruction sequence (CS# LOW state) using the proposed CRC32 polynomial. Once the CS# is brought HIGH to complete the Program / Erase transaction, the CRC calculation will be stopped and the checksum latched into the CRC Register. If multiple Program / Erase transactions are executed by the host, the slave will continue updating the CRC checksum between every CS# LOW cycle.

Figure 23. Program CRC Protection



The host device will read the CRC checksum from the slave device using the Read Interface CRC transaction. The slave device will include the RDCRC_4_0 transaction as part of the CRC checksum and then place the checksum data on the data bus. If the host device upon receiving the slave's CRC checksum finds a mismatch with its own calculated CRC checksum, it can re-issue the Program / Erase transaction to the slave device. For Flash, multiple Program / Erase to the same location due to CRC checksum errors will affect data endurance.

4.3.3 HyperBus Interface CRC Related Registers and Transaction

Table 29. HyperBus Interface CRC Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
N/A	Read Interface CRC Register (RDICRC_1_1) - Read Interface ASO

4.3.4 Legacy (x1) SPI Interface CRC Related Registers and Transaction

Table 30. Legacy (x1) SPI Interface CRC Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
Interface CRC Enable Register (ICEV)	N/A

4.4 Data Integrity CRC

HL-T/HS-T family devices have a group of transactions to perform a hardware accelerated CRC calculation over a user defined address range in the memory array. The calculation is another type of embedded operation similar to programming or erase, in which the device is busy while the calculation is in progress. The CRC operation uses the same CRC32 polynomial as Interface CRC to determine the CRC check-value.

CRC32 Polynomial: $X^{32} + X^{28} + X^{27} + X^{26} + X^{25} + X^{23} + X^{22} + X^{20} + X^{19} + X^{18} + X^{14} + X^{13} + X^{11} + X^{10} + X^{9} + X^{8} + X^{6} + 1$

The check-value generation sequence is started by entering the Data Integrity Check (DICHK_4_1, LDENAD_1_1) transaction. The transaction includes/requires loading the beginning address into the CRC Start Address Register. The transaction also includes/requires loading the ending address into the CRC End Address Register. Bringing CS# HIGH starts the CRC calculation. The CRC process calculates the check-value on the data contained at the starting address through the ending address. Note that in HyperBus interface, loading the ending address initiates the Data Integrity CRC calculation.

During the calculation period, the device goes into the Busy state (RDYBSY = 1). Once the check-value calculation is completed, the device returns to the Ready state (RDYBSY = 0) and the calculated check-value is available to be read. The check-value is stored in the Data Integrity CRC Register (DCRV[31:0]) and can be read (RDARG_4_0, RDDICL_2_1, RDDICU_2_1).

The check-value calculation can only be initiated when the device is in Standby State; and once started it can be suspended with the CRC Suspend transaction (SPEPD_0_0, SP_DIC_1_1) to read data from the memory array. During the Suspended state, the CRC Suspend Status Bit in the Status Register will be set (DICRCS = 1). Once suspended, the host can read the Status Register, read data from the array and can resume the CRC calculation by using the CRC Resume transaction (RSEPS_0_0, RS_DIC_1_1).

The Ending Address (ENDADD) must be at least two addresses higher than the Starting Address (STRADD). If ENDADD < STRADD + 3 the check-value calculation will abort and the device will return to the Ready state (RDYBSY = 0). In Legacy (x1) SPI, Data Integrity CRC abort status bit will be set (DICRCA = 1) to indicate the aborted condition. The DICRCA bit can be cleared, once set, by Software Reset or a valid subsequent CRC command execution. In HyperBus, Write Buffer Abort bit will be set (WRBFAB = 1) to indicate the aborted condition. A Status Register Clear will clear the WRBFAB bit and allow a subsequent valid CRC command or CRC ASO exit sequence. If ENDADD < STRADD + 3, the check-value will hold indeterminate data.

Note Any invalid transaction during CRC check-value calculation can corrupt the check-value data.

4.4.1 HyperBus Data Integrity Check Related Registers and Transactions

Table 31. HyperBus Data Integrity CRC Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
Status Register (STRV)	Read Status Register (RDVSTR_2_0) Clear Status Register Failure Flags (CLVSTR_1_0)
Data Integrity CRC Check-Value Register (DCRV)	Load Start Address (LDSTAD_1_1) - Data Integrity Check ASO Load End Address (LDENAD_1_1) - Data Integrity Check ASO Read Data Integrity CRC Register Upper Word (RDDICU_2_1) - Data Integrity Check ASO Read Data Integrity CRC Register Lower Word (RDDICL_2_1) - Data Integrity Check ASO

4.4.2 Legacy (x1) SPI Data Integrity Check Related Registers and Transactions

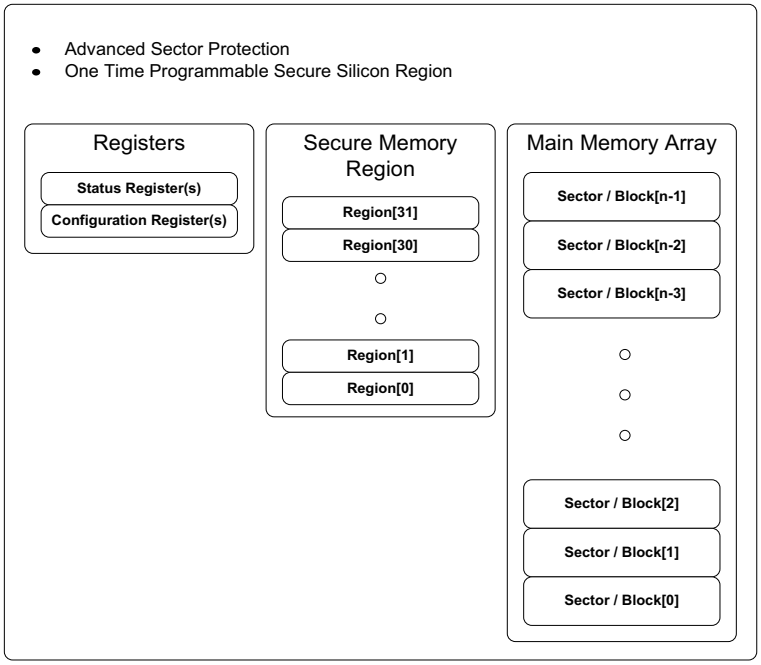
Table 32. Legacy (x1) SPI Data Integrity CRC Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
Status Register 1 (STR1N, STR1V)	Data Integrity Check (DICHK_4_1)
Status Register 2 (STR2V)	Suspend Erase/Program/Data Integrity Check (SPEPD_0_0)
Data Integrity CRC Check-Value Register (DCRV)	Resume Erase/Program/ Data Integrity Check (RSEPD_0_0)

4.5 Data Protection Schemes

Data protection is required to safeguard against unintended changes to stored data and device configuration. This includes inadvertent erasing or programming the memory array as well as writing to the Configuration registers, which can alter the functionality of the device. Three types of protection schemes are discussed which range from protecting either a single or a group of sectors or the complete memory array. Figure 24 shows an overview of different protection schemes along with applicable data regions.

Figure 24. Data Protection and Security (Write/Program/Erase) Schemes



4.5.1 Advanced Sector Protection (ASP)

The Advanced Sector Protection (ASP) scheme allows each memory array sector to be independently controlled for protection against erasing or programming, either by volatile or nonvolatile locking features. The nonvolatile locking configuration can also be locked, and password-protected.

The main memory array sectors are protected against erase and program by volatile (Dynamic Bits - DYB) and nonvolatile (Persistent Protection Bits - PPB) protection bit pairs. Each DYB/PPB bit pair can be individually set to '0' protecting the related sector or cleared to '1' unprotecting the related sector. DYB protection bits can be set and cleared as often as needed whereas PPB bits being nonvolatile must adhere to their respective technology based endurance requirements. Figure 25 shows an overview of ASP.

Figure 25. Advanced Sector Protection (Nonvolatile)

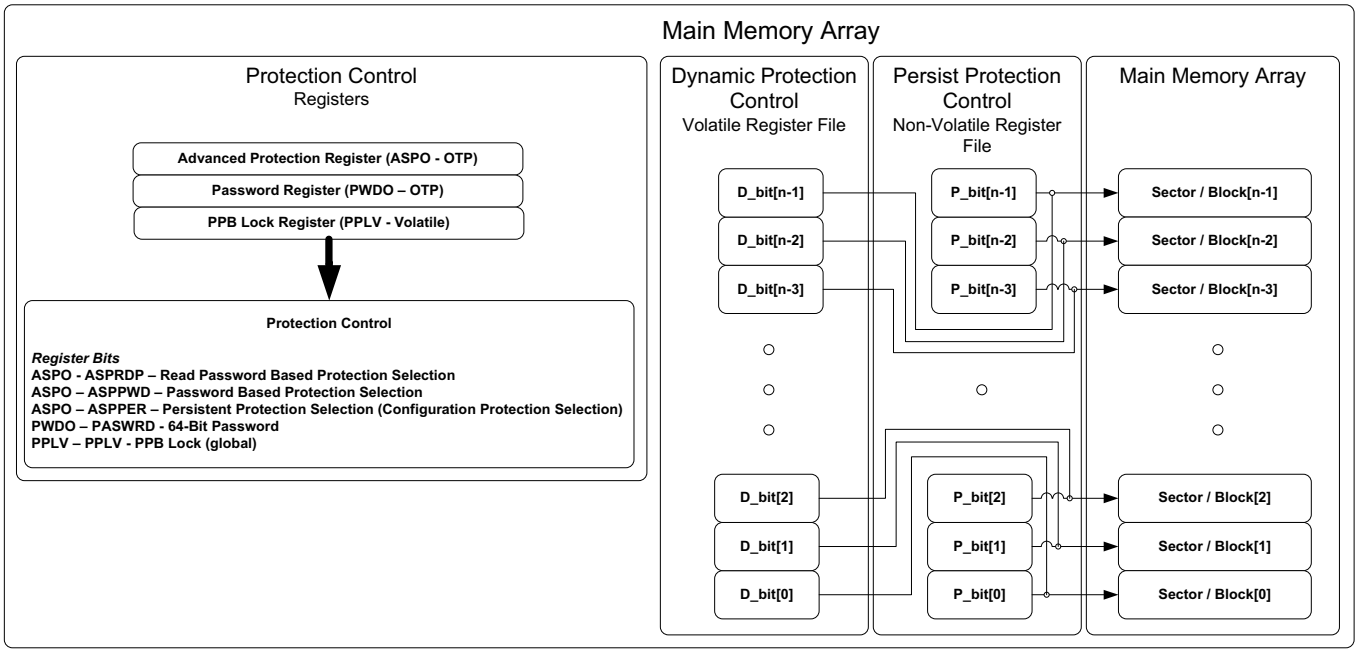
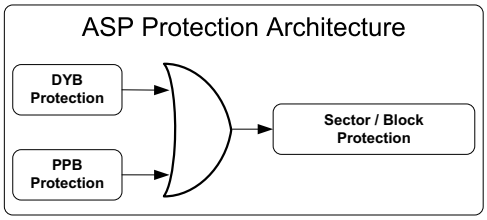


Figure 26. DYB and PPB Protection Control



ASP provides a rich set of configuration options producing multiple data protection schemes which can be employed based on design or system needs. These configuration options are discussed in Section 4.5.1.1 ASP Protection Summary on page 49 through Section 4.5.1.9 Legacy (x1) SPI ASP Related Registers and Transactions on page 54.

4.5.1.1 ASP Protection Summary

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 mode 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 Protection mode 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.

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 nonvolatile and saved across a power cycle or Hardware Reset. Changing the protection state requires programming or erase of the PPB bits.

Table 33. 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	
1	0	0	
0	1	1	Unprotected – PPB not changeable, DYB is changeable
0	1	0	Protected – PPB not changeable, DYB is changeable
0	0	1	
0	0	0	

4.5.1.2 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

4.5.1.3 DYB (Volatile) Sector Protection

Dynamic Protection Bits (DYB) 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 Write transaction, 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. As mentioned above, the DYBs can be set to 0 or cleared to 1 as often as needed.

4.5.1.4 PPB (Nonvolatile) Sector Protection

The Persistent Protection Bits (PPB) are located in a separate nonvolatile 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. 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.

4.5.1.5 Persistent Protection Mode

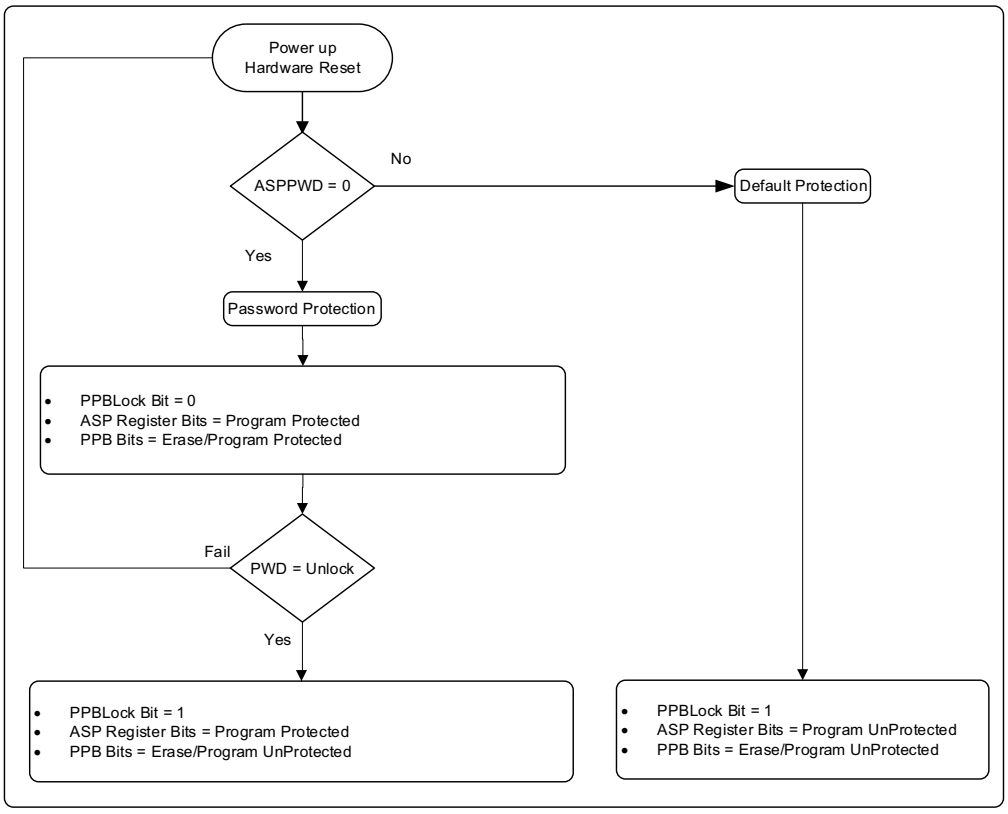
The Persistent Protection method sets the PPB Lock bit to 1 during POR or Hardware Reset so that the PPB bits are unprotected by a device reset. There is a command to clear the PPB Lock bit to 0 to protect the PPB. There is no command in the Persistent Protection method to set the PPB Lock bit to 1 therefore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

4.5.1.6 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.

The Password Protection scheme flowchart is shown in Figure 27.

Figure 27. Password Protection Scheme Flowchart



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') 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. In Legacy (x1) SPI, attempted programming of a protected Password will set the Program Status bit. In HyperBus, attempted programming of a protected Password will set the Sector Lock Status bit and the Program Status bit. 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. 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 needs to be entered in exact order. 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 t_{PSWD} for setting the PPB Lock bit after the valid 64-bit password is given to the device.
- The Password Unlock command is used to unlock the password. If the supplied password does not match the hidden internal password, the device gets locked out and does not accept any further instructions. Either a hardware reset or a CS# Signaling Reset or a power cycle are required to bring the device back to standby mode to try reentering the password again.
- If the password is lost after setting the Password Mode Lock bit, there is no way to clear the PPB Lock bit.

4.5.1.7 Read Password Protection Mode

The Read Password mode can replace the default Section 4.5.1.6 Password Protection Mode on page 50. 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 Nonvolatile Configuration Register bits xVCR1[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 or to 1s to select the lowest or highest address Flash Memory Array address range. In SPI, TBPROT (CFR1x[5] is used for this address range selection whereas in HyperBus, TB4KBS[1:0] (CFR1x[9:8] bits are used.

Note that AutoBoot is disabled when the Read Password protection is enabled. The ATBR[0] (AutoBoot Enable) bit is ignored in Read Password mode.

Table 34. ASP Configuration Register Selection of Persistent and Password Protection Modes

ASPO 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 / Password Modes Disabled (factory default)
1	1	

Table 35. HyperBus CFR1x Mapping of Boot Block Address Range

CFR1x Bit	Default Value	Name
CFR1x[9:8]	11	00 = Map Parameter-Sectors and Read Password Sector mapped into lowest addresses. 01 = Map Parameter-Sectors and Read Password Sector 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.

Table 36. Legacy (x1) SPI CFR1x Mapping of Boot Block Address Range

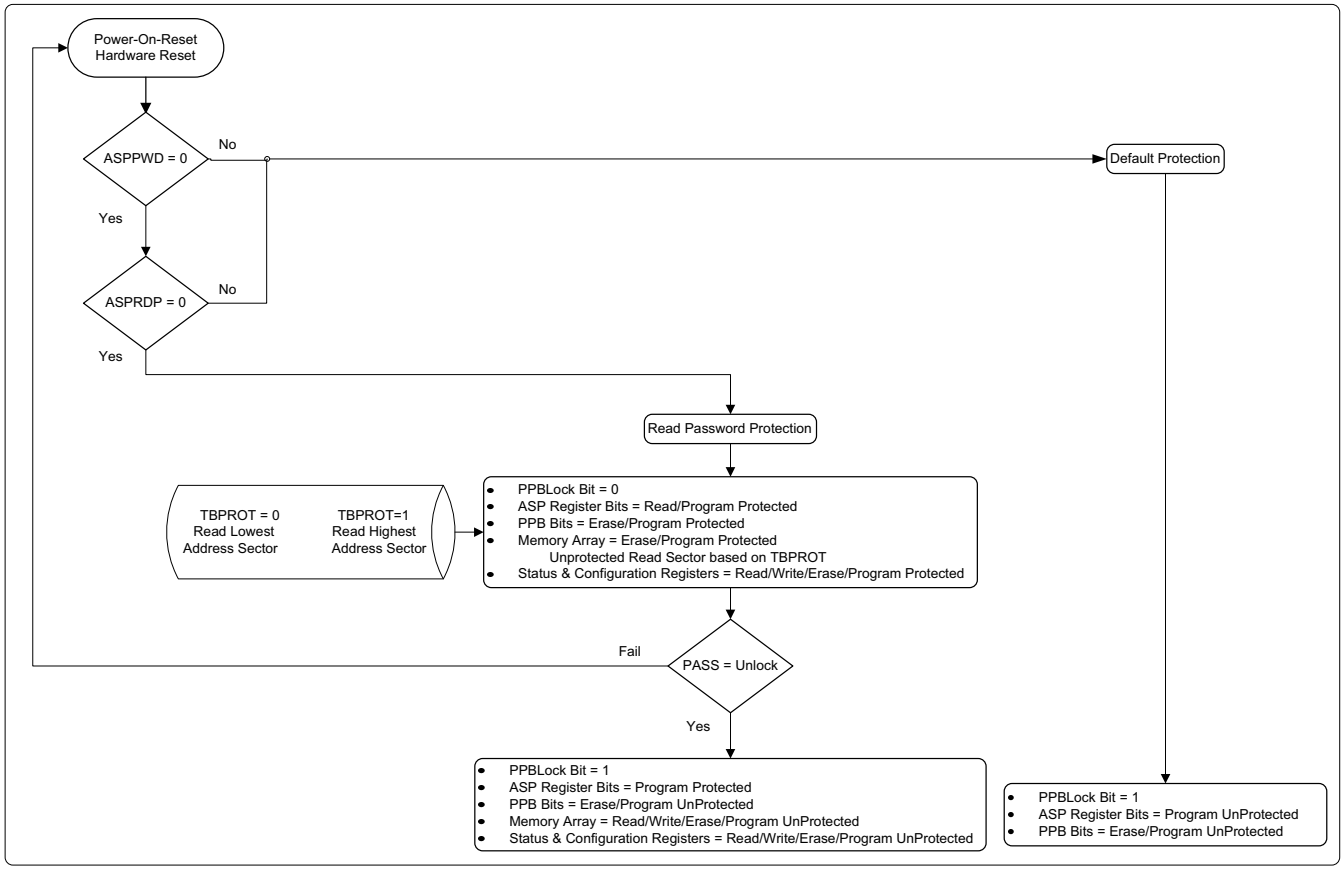
CFR1x Bit	Default Value	Name
CFR1x[5]	0	0 = Map Parameter-Sectors and Read Password Sector mapped into highest (top) addresses. 1 = Map Parameter-Sectors and Read Password Sector mapped into lowest (bottom) addresses.

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.

The Read Password Protection scheme flowchart is shown in Figure 28.

Figure 28. Read Password Protection Scheme Flowchart



Read Password Protection Notes

- Read Password is enabled by programming the ASPR[5] bit to 0.
- 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.

4.5.1.8 HyperBus ASP Related Registers and Transactions

Table 37. HyperBus ASP Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
Advanced Sector Protection Register (ASPO)	Program Advanced Sector Protection Register (PGOASP_2_1) - Advanced Sector Protection ASO Read Advanced Sector Protection Register (RDOASP_1_1) - Advanced Sector Protection ASO
Configuration Register 1 (CFR1N, CFR1V)	Read Volatile Configuration Register 1 (RDVCR1_4_0) Read Nonvolatile Configuration Register 1 (RDNCR1_4_0)
	Program Nonvolatile Persistent Protection Bits (PGNPPB_2_1) - PPB ASO
	Erase Nonvolatile Persistent Protection Bits (ERNPPB_2_1) - PPB ASO
	Read Nonvolatile Persistent Protection Bits (RDNPPB_1_1) - PPB ASO
	Clear Volatile Persistent Protection Lock (CLVPPL_2_1) - PPB Lock ASO
	Read Volatile Persistent Protection Lock (RDVPPL_1_1) - PPB Lock ASO
	Set Volatile Dynamic Protection Bits (STVDYB_2_1) - DYB ASO
	Clear Volatile Dynamic Protection Bit (CLVDYB_2_1) - DYB ASO
	Read Volatile Dynamic Protection Bit (RDVDYB_1_1) - DYB ASO
	Sector Protection Status (PRTSTS_2_1) - PPB/DYB ASO

4.5.1.9 Legacy (x1) SPI ASP Related Registers and Transactions

Table 38. Legacy (x1) SPI ASP Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
Advanced Sector Protection Register (ASPO)	Read Dynamic Protection Bit (RDDYB_4_0)
Configuration Register 1 (CFR1N, CFR1V)	Write Dynamic Protection Bit (WRDYB_4_1)
	Read Persistent Protection Bit (RDPPB_4_0)
	Program Persistent Protection Bit (PRPPB_4_0)
	Erase Persistent Protection Bit (ERPPB_0_0)
	Write PPB Protection Lock Bit (WRPLB_0_0)
	Read Password Protection Mode Lock Bit (RDPLB_0_0)
	Password Unlock (PWDUL_0_1)
	Write Enable (WRENB_0_0)
	Read Any Register (RDARG_4_0)
	Write Any Register (WRARG_C_1)

4.5.2 Secure Silicon Region (SSR)

Each device has a 1024B OTP 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.
- The next four 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 12B 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.

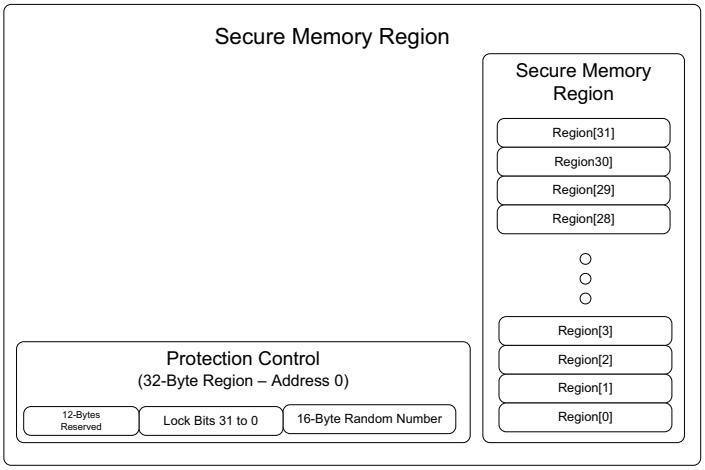
The remaining regions are erased when shipped from Cypress, and are available for programming of additional permanent data.

See Figure 29 on page 54 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.

In HyperBus, the Configuration Register Temporary Locking selection of SSR (CFR1x[10]) bit protects the entire SSR memory space from programming when cleared. This allows trusted boot code to control programming of SSR regions then set the Locking bit to prevent further SSR memory space programming during the remainder of normal power-on system operation.

Figure 29. OTP Protection (Nonvolatile)



4.5.2.1 Reading the SSR Memory Space

In HyperBus interface, 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 1KB 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.

In Legacy (x1) SPI interface, reading the SSR Region is performed using Read SSR transaction (RDSSR_4_0).

4.5.2.2 Programming SSR Memory Space

In HyperBus interface, 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 Table 39. SSR Program operations outside the valid SSR address range will ignore address A9 and higher and will alias into the valid SSR address range. It is required to align start address to 32 bits while programming SSR space, which means the A0 bit of address should be 0'b and host should deassert CS# to align with 32 bits. SSR Program operations while Temporary Locking selection of SSR (CFR1x[10]) = 0 will fail with no indication of the failure. The SSR address space is not protected by the selection of an ASP Protection mode. A SSR Exit sequence will return the device to the Read mode.

In Legacy (x1) SPI interface, programming the SSR Region is performed using Read SSR transaction (PRSSR_4_0). It is required to align start address to 32 bits while programming SSR space, which means the address bits A1 and A0 should be 0'b and host should deassert CS# to align with 32 bits.

Table 39. SSR Address Map

Region	Byte Address Range (Hex)	Contents	Initial Delivery State (Hex)
Region 0	0000h	Least Significant Byte of Cypress Programmed Random Number	Cypress Programmed Random Number
	...	...	
	000Fh	Most Significant Byte of Cypress Programmed Random Number	
	0010h - 0013h	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 = FFh
	0014h - 001Fh	Reserved for Future Use (RFU)	
Region 1	0020h - 003Fh	Available for User Programming	
Region 2	0040h - 005Fh		
...	...		
Region 31	03E0h - 03FFh		

4.5.2.3 HyperBus SSR Related Registers and Transactions

Table 40. HyperBus SSR Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
N/A	Read Secure Silicon Region (RD_SSR_1_1) - SSR ASO
	Program Secure Silicon Region Word (PG_SSR_4_1) - SSR ASO
	Load Secure Silicon Region Buffer (LDBSSR_5_1) - SSR ASO
	Program Secure Silicon Region Buffer Confirm (PGCSSR_1_1) - SSR ASO
	Reset Write to Buffer Abort (RSWSSR_3_1) - SSR ASO

4.5.2.4 Legacy (x1) SPI SSR Related Registers and Transactions

Table 41. Legacy (x1) SPI SSR Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
N/A	Program Secure Silicon Region (PRSSR_4_1)
	Read Secure Silicon Region (RDSSR_4_0)

4.6 SafeBoot

Semper Flash memory devices contain an embedded microcontroller which is used to initialize the device, manage embedded operations, and perform other advanced functionality. An initialization failure of this embedded microcontroller or corruption of the Nonvolatile Configuration registers can render the flash device unusable. Barring a catastrophic event, such as permanent corruption of the embedded microcontroller firmware, it is possible to recover the device.

The SafeBoot feature allows Status Register polling to detect an embedded microcontroller initialization failure or Configuration Register corruption through error signatures.

4.6.1 Microcontroller Initialization Failure Detection (x1 Boot Option)

If the microcontroller embedded in the flash device fails to initialize, a hardware reset can recover the device, unless it is a catastrophic failure. This hardware reset must be initiated by the Host controller. Upon detecting a failed microcontroller initialization, the flash device automatically reverts to Legacy (x1) SPI Interface and provides a failure signature in its Status Register.

Table 42 shows the device's Status Register bits upon detecting an initialization failure.

Table 42. Status Register 1 Power-On Detection Signature

Bit	Field Name	Function	Detection Signature
STR1V[7]	RESVRD	Reserved for Future Use	0
STR1V[6]	PRGERR	Programming Error Status Flag	1
STR1V[5]	ERSERR	Erasing Error Status Flag	1
STR1V[4]	RESVRD	Reserved for Future Use	0
STR1V[3]			0
STR1V[2]			0
STR1V[1]	WRPGEN	Write/Program Enable Status Flag	0
STR1V[0]	RDYBSY	Device Ready/Busy Status Flag	1

Table 43. Interface Configuration upon Detecting Power-On Failure[22]

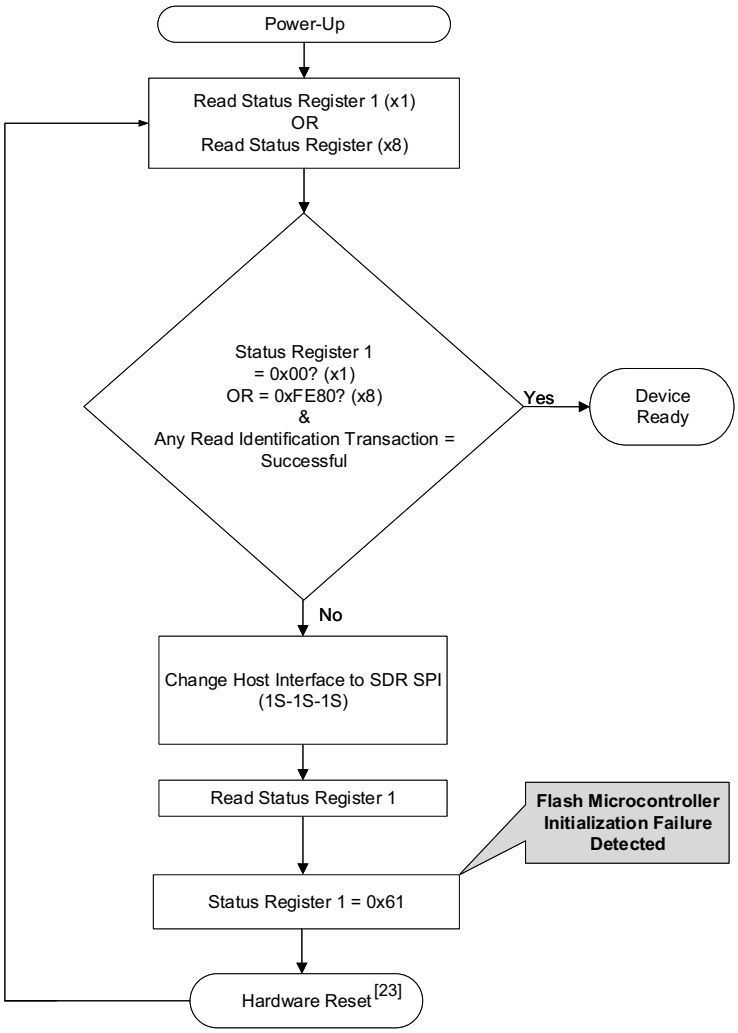
Interface	Transactions Supported	Register Type	Address (# of Bytes)	Frequency of Operation	Register Read Latency (# of Clock Cycles)	Output Impedance
SPI (1S-1S-1S)	Read Status Register 1 (RDSR1_0_0) Read Any Register (RDARG_4_0)	Status Register (Volatile Only)	4	Maximum (allowed for RDSR1_0_0, RDARG_4_0)	2	45Ω

Note
22. For reading the Status Register, providing the Nonvolatile Status Register address to RDARG_4_0 will produce indeterminate results.

4.6.1.1 Host Polling Behavior

The host will need to go through a Status Register polling sequence to determine if an initialization failure has occurred in the device. The flowchart for the sequence is shown in Figure 30.

Figure 30. Host Polling Sequence for Microcontroller Initialization Failure Detection



4.6.1.2 Microcontroller Initialization Failure Detection Related Registers and Transactions

Table 44. Microcontroller Initialization Failure Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)	Related HyperBus Transactions (see Table 128 on page 126)
Status Register 1 Volatile (STR1V)	Read Any Register (RDARG_4_0)	N/A
	Read Status Register -1 (RDSR1_0_0)	

Note
23. If you have V_{cc} within specifications and a hardware reset does not resolve the issue, replace the flash device.

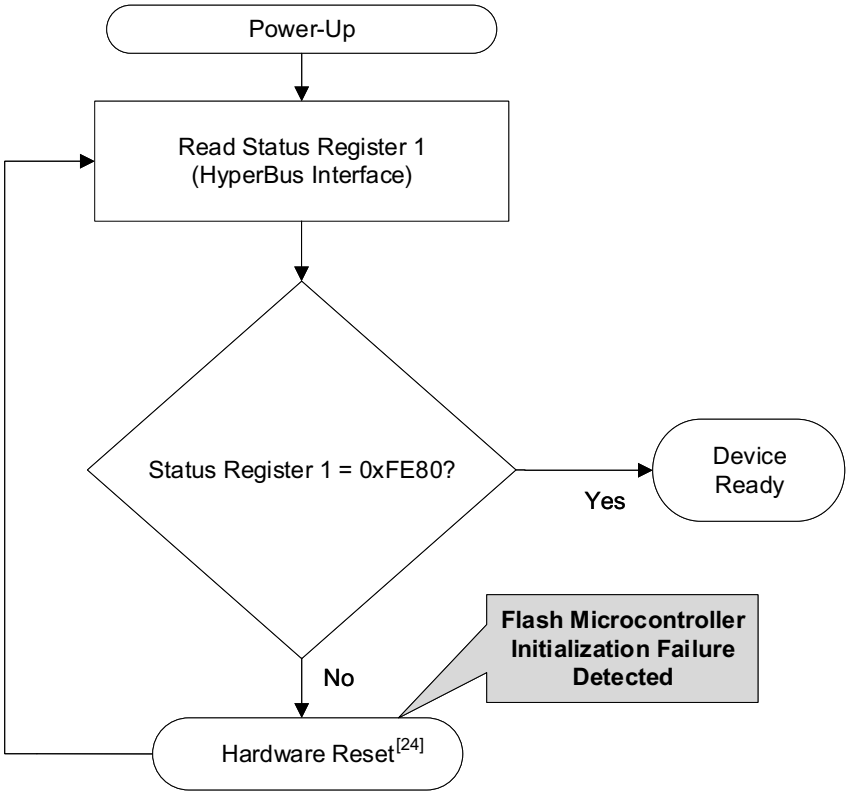
4.6.2 Microcontroller Initialization Failure Detection (x8 Boot Option)

Upon detecting a failed microcontroller initialization, the flash device will stay in HyperBus interface and won't accept any transaction from the host controller. Also RSTO# will not transition from LOW to HIGH due to failed microcontroller initialization.

4.6.2.1 Host Polling Behavior

The host will need to go through a Status Register polling sequence OR look for a RSTO# signal transitioning from LOW to HIGH to determine if an initialization failure has occurred in the device. The flowchart for the sequence is shown in Figure 31. Upon detecting Microcontroller Initialization Failure Detection, the device supports output impedance of 30Ω.

Figure 31. Host Polling Sequence for Microcontroller Initialization Failure Detection (x8 Boot Option)



Note
24. If you have V_{cc} within specifications and a hardware reset does not resolve the issue, replace the flash device.

4.6.3 Configuration Corruption Detection (HyperBus Interface)

If during device's configuration update with HyperBus interface, such as erasing and writing to a Nonvolatile Register, a power loss occurs, or a hardware reset is initiated, the Erase or Write Register transaction will get interrupted. The device will return to Standby mode, but the Nonvolatile Register data is most likely corrupted since the embedded erase or write operation was prematurely terminated.

Semper Flash with HyperBus interface will flag the PRGERR (STRV[4]) bit in case of a power loss occurs or a hardware reset is initiated during erasing the Nonvolatile Configuration Register. Therefore, it is recommended to poll the Status Register before writing the Nonvolatile Configuration registers. Upon detecting corruption in the configuration, the device supports the read latency of 20 cycles.

Before initiating the normal operation, it is recommended to verify the device configuration after Nonvolatile Register write operation. If the register value verification does not meet to the required configuration then it is required to rewrite the configuration again in HyperBus interface.

4.6.4 Configuration Corruption Detection - (x1 Legacy SPI)

If during device's configuration update with legacy SPI (1S-1S-1S) protocol, such as writing to a Nonvolatile Register, a power loss occurs or a hardware reset is initiated, the write register transaction will get interrupted. The device will return to Standby mode but the Nonvolatile Register data is most likely corrupted since the embedded write operation was prematurely terminated. During the next power-up, the configuration corruption is detected with appropriate failure flags displayed in the Status Register allowing rewriting the configuration again. The device will maintain the configured protection scheme.

Table 45 shows the device's Status Register bits upon detecting a configuration corruption.

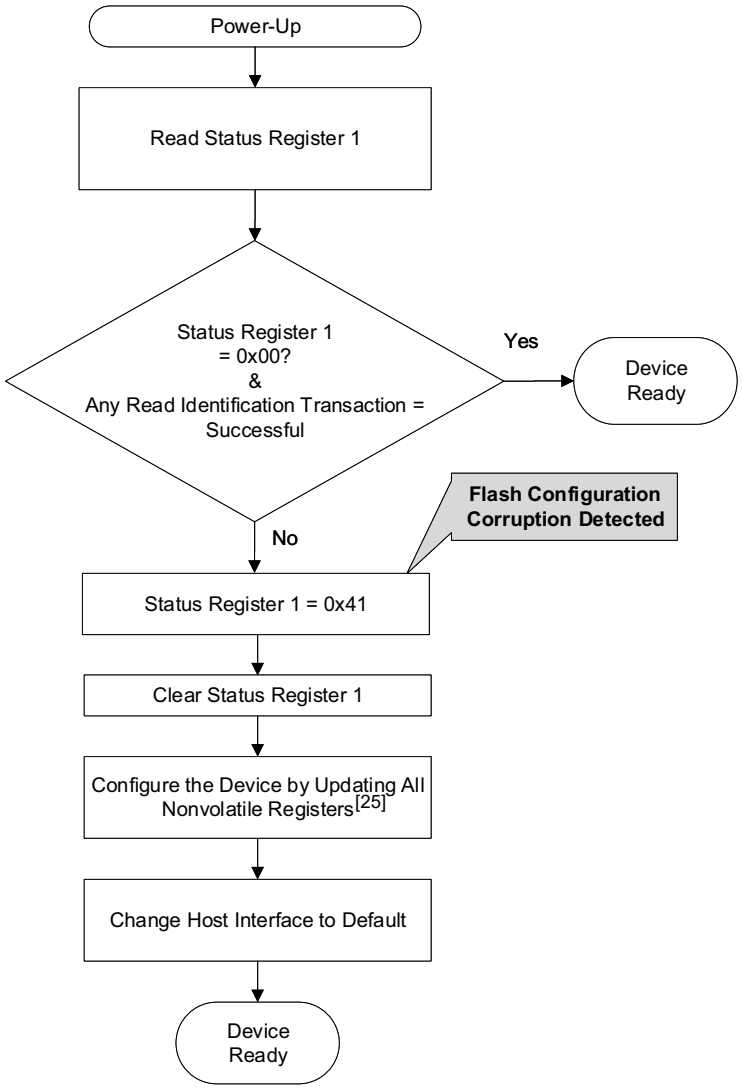
Table 45. Status Register 1 Configuration Corruption Detection Signature

Bit	Field Name	Function	Detection Signature
STR1V[7]	RESVRD	Reserved for Future Use	0
STR1V[6]	PRGERR	Programming Error Status Flag	1
STR1V[5]	ERSERR	Erasing Error Status Flag	0
STR1V[4]	RESVRD	Reserved for Future Use	0
STR1V[3]			0
STR1V[2]			0
STR1V[1]	WRPGEN	Write/Program Enable Status Flag	0
STR1V[0]	RDYBSY	Device Ready/Busy Status Flag	1

Table 46. Interface Configuration upon Detecting Configuration Corruption

Interface	Transactions Supported	Address (# of Bytes)	Frequency of Operation	Output Impedance
Legacy (x1) SPI	All SPI Transactions	4	Maximum	45Ω

Figure 32. Host Polling Sequence for Configuration Corruption Detection (x1 Legacy SPI)



4.6.4.1 Configuration Corruption Detection Related Registers

Table 47. Configuration Corruption Detection Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
Status Register 1 Volatile (STR1V)	All 1S-1S-1S Transactions

Note

25. As soon as first Write Any Register transaction updates the Nonvolatile Status register or Configuration register, all remaining nonvolatile status and configuration registers go back to the predefined state (STR1N = 0x00, CFR1N = 0x00, CFR2N = 0x00, CFR3N = 0x00, CFR4N = 0x00). It is recommended to initiate SafeBoot recovery operation by configuring the Address byte length and latency followed by rest configurations.

4.7 AutoBoot

AutoBoot allows the host to read data from HL-T/HS-T family of devices after power up or after a hardware reset without having to send any read transactions (including the address). Based on the device configuration, data is output on the interface I/Os once CS# is brought Low and CK is toggled.

The starting address for the read data is specified in the AutoBoot Register (ATBN[31:9] - STADR[22:0]). This starting address can be at any page boundary location in the memory (512 byte page boundary). Also identified in the AutoBoot Register is a starting delay which is represented as the number of clock cycles (ATBN[8:1] - STDLY[7:0]). This delay is instituted before the data is read out. The delay can be programmed to meet the host's requirements but a minimum amount is required to meet the memory access times based on the frequency for operation. It is highly recommended to check the Status Register 1 value after successful or unsuccessful AutoBoot execution to verify the configuration corruption (SafeBoot).

Note Wrap function must be disabled for AutoBoot.

Note AutoBoot is disabled when the Read Password feature is enabled, as part of the Advanced Sector Protection. It is recommended to disable AutoBoot (ATBN[0] - ATBTEN) when Read Password feature is enabled.

Note Autoboot with Interface CRC enabled requires reading out at least 4 words of data.

Note It is highly recommended to assign first AutoBoot address in the Long Retention region.

4.7.1 HyperBus AutoBoot Related Registers and Transactions

Table 48. HyperBus AutoBoot Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
AutoBoot Register (ATBN)	Program AutoBoot Register (PGNATB_2_1) - AutoBoot ASO
	Read AutoBoot Register (RDATBN_1_0) - AutoBoot ASO
	AutoBoot HyperBus Transaction (see Figure 27 on page 50)

4.7.2 Legacy (x1) SPI AutoBoot Related Registers and Transactions

Table 49. Legacy (x1) SPI AutoBoot Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
AutoBoot Register (ATBN)	Read Any Register (RDARG_4_0)
	Write Any Register (WRARG_C_1)
	AutoBoot Transaction (see Figure 13 on page 14)

4.8 Read Transactions

HL-T/HS-T supports different read transactions, namely: Read Memory array, Read Device Identification, Read Register, Read Secure Silicon, Read Protection DYB and Read Protection PPB bits. These read transactions can use any of following two interfaces/ protocols:

- Legacy (x1) SPI interface with SDR
- HyperBus interface with DDR

These read transactions use the following features:

- The read transactions require latency cycles following the command/address bytes to allow time to access the memory array (except RDAY1_4_0 and RDAY1_C_0 of Legacy (x1) SPI protocol).
- Data Strobe (DS), available only in HyperBus, is an output clock which is edge aligned with read data enabling the memory controller to capture data (see Section 4.8.4 Data Strobe (DS) - HyperBus on page 65).
- The read transaction has the option of wrapped, hybrid, or linear bursts.

4.8.1 Read Identification Transactions

There are three unique identification transactions, each supporting the two protocols (Legacy (x1) SPI) and (HyperBus).

4.8.1.1 Read Device Identification Transaction

The Read Device Identification (RDIDN_0_0, RDIDSF_1_1) transaction provides read access to manufacturer identification and device identification. The Legacy (x1) SPI mode has no address cycles. In SPI mode, the transaction uses latency cycles set by (CFR3V[1:0]) to enable maximum clock frequency of 166 MHz under. Similarly in HyperBus mode, latency cycles set by (CFR1V[7:4]) to enable maximum clock frequency of 166 MHz under HL-T and 200 MHz under HS-T. The HyperBus mode supports DS for capturing data.

4.8.1.2 Read SFDP Transaction

The Read Serial Flash Discoverable Parameters (RSFDP_3_0, RDIDSF_1_1) transaction provides access to the JEDEC Serial Flash Discovery Parameters (SFDP). In SPI mode, the transaction uses a 3-byte address. If a non-zero address is set, the selected location in the SFDP space is the starting point of the data read. This enables random access to any parameter in the SFDP space. Read SFDP Transaction is not supported in Read Password mode before the password is provided. In SPI mode, the maximum clock frequency for the Read SFDP transaction is 156 MHz whereas in HyperBus mode, 166 MHz under HL-T, and 200 MHz under HS-T. The HyperBus mode supports DS for capturing data.

4.8.1.3 Read Unique Identification Transaction

Read Unique Identification (RDUID_0_0, RDIDSF_1_1) transaction is similar to Read Device Identification transaction, but accesses a different 64-bit number, which is unique to each device. The Unique ID is factory programmed.

4.8.1.4 HyperBus Read Identification Related Register and Transaction

Table 50. HyperBus Read Identification Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
HyperBus Configuration Register 1 (CFR1N, CFR1V)	Read ID/Unique ID/SFDP (RDIDSF_1_1) - ID/Unique ID/SFDP ASO

4.8.1.5 Legacy (x1) SPI Read Identification Related Register and Transaction

Table 51. Legacy (x1) SPI Read Identification Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
SPI Configuration Register 3 (CFR3N, CFR3V)	Read Identification (RDIDN_0_0)

4.8.2 Read Memory Array Transactions

Memory array data can be read from the memory starting at any byte boundary. Data bytes are sequentially read from incrementally higher byte addresses until the host ends the data transfer by driving CS# input HIGH. If the byte address reaches the maximum address of the memory array, the read will continue at address zero of the array.

4.8.2.1 Read Transaction - HyperBus

The HyperBus Read transaction provides the fastest data throughput using DDR protocol. This protocol supports DS for capturing data. The option of linear burst length as well as wrapped burst length is available. This transaction uses latency cycles set by (CFR1V[7:4]) to enable maximum clock frequency of 166 MHz under HL-T, and 200 MHz under HS-T.

4.8.2.2 Read and Fast Read Transactions - Legacy (x1) SPI

The SPI Read and Fast Read transactions are supported for Host systems that require backward compatibility to legacy SPI. This protocol does not support the DS for capture of data. The options of linear and wrapped read length is available. The Read transaction supports a maximum clock frequency of 50 MHz and requires no latency cycles. The Fast Read Transaction uses latency cycles set by (CFR2V[3:0]) to enable maximum clock frequency of 166 MHz.

4.8.2.3 HyperBus Read Memory Array Related Registers and Transactions

Table 52. HyperBus Read Memory Array Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
HyperBus Configuration Register 1 (CFR2N, CFR2V)	Read (RDMARY_1_0)

4.8.2.4 Legacy (x1) SPI Read Memory Array Related Registers and Transactions

Table 53. Legacy (x1) SPI Read Memory Array Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
SPI Configuration Register 2 (CFR2N, CFR2V)	Read (RDAY1_4_0, RDAY1_C_0)
SPI Configuration Register 4 (CFR4N, CFR4V)	Read Fast (RDAY2_C_0)

4.8.3 Read Registers Transactions

There are multiple registers for reporting embedded operation status as well as controlling device configuration options. Registers contain both volatile and nonvolatile bits. There are two transaction types to read the Registers, namely generic and dedicated. Generic (Read Any Register) transaction provides a way to read all device registers: Nonvolatile and Volatile by address selection. Dedicated Register Read transactions which is defined per register and only reads the contents of that register. Legacy (x1) SPI supports both transaction types whereas HyperBus supports dedicated transactions only.

4.8.3.1 Read Configuration Register Transaction - HyperBus

The Read Configuration Register (RDVCR1_4_0, RDVCR2_4_0, RDNCR1_4_0, RDNCR2_4_0) transactions read both the device Configuration registers (Volatile and Nonvolatile). This is followed by a number of latency cycles set by (CFR1V[7:4]) to enable maximum clock frequency of 166 MHz under HL-T HyperBus mode, and 200 MHz under HS-T. The protocol supports DS for capturing data.

4.8.3.2 Read Any Register - Legacy (x1) SPI

The Read Any Register (RDARG_4_0) transaction is the best way to read all device registers, both Nonvolatile and Volatile. The transaction includes the address of the register to be read. This is followed by a number of latency cycles set by (CFR2V[3:0]) for reading Nonvolatile registers and CFR3V[7:6] for reading Volatile registers. Then, the selected register contents are returned. If the read access is continued, the same addressed register contents are returned until the transaction is terminated; only one byte register location is read by each RDARG_4_0 transaction. For registers with more that one byte of data, the RDARG_4_0 transaction must be repeated with the address of the second byte (and so on) to read each byte of data.

The maximum clock frequency for the RDARG_4_0 transaction is 166 MHz under SPI mode.

The RDARG_4_0 transaction can be used during embedded operations to read Status Register 1 (STR1V). It is not used for reading registers such as ASP PPB Access Register (PPAV) and ASP Dynamic Block Access Register (DYAV). There are separate commands required to select and read the location in the array accessed. The RDARG_4_0 transaction will read invalid data from the PASS Register locations if the ASP Password protection mode is selected by programming ASPR[2:0]. Reading undefined locations provides undefined data.

4.8.3.3 Read Status Registers Transaction

The Read Status Register (RDSR1_0_0, RDSR2_0_0, RDVSTR_2_0) transactions allow the registers' volatile contents to be read. The SPI mode has no address cycles. The SPI transaction uses latency cycles set by (CFR3V[7:6]) for reading Volatile registers to enable maximum clock frequency of 166 MHz. The HyperBus transaction uses latency cycles set by (CFR1V[7:4]) for reading Volatile registers to enable maximum clock frequency of 166 MHz under HL-T, and 200 MHz under HS-T. The HyperBus mode supports DS for capturing data.

The Status Register (SPI - volatile only) contents can be read at any time, even while a program, erase, or write operation is in progress.

In SPI mode, it is possible to continuously read Status Register by providing multiples of eight clock cycles. The status is updated for each eight cycle read.

4.8.3.4 Read Dynamic Protection Bit (DYB) Access Register Transaction

The Read DYB Access Register (RDDYB_4_0, RDVDYB_1_1) transaction reads the contents of the DYB Access Register. The SPI transaction uses latency cycles set by (CFR3V[7:6]) for reading Volatile registers to enable maximum clock frequency of 166 MHz. The HyperBus transaction uses latency cycles set by (CFR1V[7:4]) for reading Volatile registers to enable maximum clock frequency of 166 MHz under HL-T, and 200 MHz under HS-T. The HyperBus mode supports the DS for capturing data. It is not possible to read DYB Access Register continuously for the entire array. Each location must be read with a separate Read DYB transaction.

4.8.3.5 Read Persistent Protection Bit (PPB) Access Register Transaction

The Read PPB Access Register (RDPBB_4_0, RDNPPB_1_1) transaction reads the contents of the PPB Access Register. The SPI transaction uses latency cycles set by (CFR3V[7:6]) for reading Volatile registers to enable maximum clock frequency of 166 MHz. The HyperBus transaction uses latency cycles set by (CFR1V[7:4]) for reading Volatile registers to enable maximum clock frequency of 166 MHz under HL-T, and 200 MHz under HS-T. The HyperBus mode supports the DS for capturing data. It is not possible to read PPB Access Register continuously for all the sectors in the array. Each location must be read with a separate Read PPB transaction.

4.8.3.6 Read PPB Lock Transaction

The Read PPB Lock Register (RDPLB_0_0, RDVPPL_1_1) transaction allows reading the global Persistent Protection Lock bit. The SPI mode has no address cycles. The SPI transaction uses latency cycles set by (CFR3V[7:6]) for reading Volatile registers to enable maximum clock frequency of 166 MHz. The HyperBus transaction uses latency cycles set by (CFR1V[7:4]) for reading Volatile registers to enable maximum clock frequency of 166 MHz under HL-T, and 200 MHz under HS-T. The HyperBus mode supports DS for capturing data.

4.8.3.7 Read ECC Data Unit Status

The Read ECC Data Unit Status (RDECC_4_0, RDECST_1_1) transaction is used to determine the ECC status of the addressed unit data. In this transaction, the LSb of the address must be aligned to an ECC data unit. The SPI transaction uses latency cycles set by (CFR3V[7:6]) for reading Volatile registers to enable maximum clock frequency of 166 MHz. The HyperBus transaction uses latency cycles set by (CFR1V[7:4]) for reading Volatile registers to enable maximum clock frequency of 166 MHz under HL-T, and 200 MHz under HS-T. The HyperBus mode supports DS for capturing data.

In SPI mode, the device outputs the byte contents of the ECC Status for the selected ECC unit. In HyperBus mode, the device outputs the word contents of the ECC status for the selected ECC unit. To read the next ECC unit status, another Read ECC Data Unit Status transaction must be initiated to the next ECC data unit aligned address - incremented by 16 bytes.

4.8.3.8 Read Power-On-Reset (POR) Timer Transaction - HyperBus

The Read Power-On-Reset Timer (RDNPOR_4_0) transaction is used to read the contents of the POR Timer Register. This transaction uses latency cycles set by (CFR1V[7:4]) to enable maximum clock frequency of 166 MHz under HL-T, and 200 MHz. under HS-T. The HyperBus mode supports DS for capturing data. The POR Timer Register controls the functionality of the RSTO# pin.

4.8.3.9 HyperBus Read Register Related Registers and Transactions

Table 54. HyperBus Read Register Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
HyperBus Configuration Register 1 (CFR1N, CFR1V)	Read ID/Unique ID/SFDP (RDIDSF_1_1) - Device ID/ Unique ID/SFDP ASO
	Read Status Register (RDVSTR_2_0)
	Read Volatile Configuration Register 1 (RDVCR1_4_0) Read Volatile Configuration Register 2 (RDVCR2_4_0)
	Read Nonvolatile Configuration Register 1 (RDNCR1_4_0) Read Nonvolatile Configuration Register 2 (RDNCR2_4_0)
	Read Nonvolatile Persistent Protection Bits (RDNPPB_1_1)
	Read Volatile Persistent Protection Lock (RDVPPL_1_1)
	Read Volatile Dynamic Protection Bits (RDVDYB_1_1)
	Read Error Correction (ECC) Status (RDECST_1_1)
	Read Nonvolatile POR Timer Register (PRNPOR_4_0)

4.8.3.10 Legacy (x1) SPI Read Register Related Registers and Transactions

Table 55. Legacy (x1) SPI Read Register Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 132 on page 143)
SPI Configuration Register 2 (CFR2N, CFR2V)	Read Device ID (RDIDN_0_0)
SPI Configuration Register 3 (CFR3N, CFR3V)	Read SFDP (RSFDP_3_0)

4.8.4 Data Strobe (DS) - HyperBus

To allow for higher data rates, data strobe signal (DS) is added to DDR devices. The data strobe is non-free-running signal driven by the device, which is also driving the data signals. At the board level, the strobe signal has identical loading to data signals and needs to be routed similarly. During the period of data transfer in a read transaction, the Data Strobe (DS) signal is driven by the device and transitions edge aligned with the IO signal data transitions. DS is used as an additional output signal with the same timing characteristics as other data outputs but with the guarantee of transitioning with every data bit transferred. The DS signal transitions can be received and internally phase shifted by the master to be used as an internal read data clock to capture each data bit transferred.

DS behavior is described as follows:

- DS will start transiting upon valid data
- DS will continue to toggle as long as CS# is low and CK/CK# is toggling. Exceptions to this rule are:
  - Inter-page boundary latency
  - DS Stall upon detecting an ECC error when reading a Half-page of data.

4.8.5 Burst Types

The device supports three burst types during Read transactions:

- Linear Burst
- Wrapped Burst
- Hybrid Burst - HyperBus

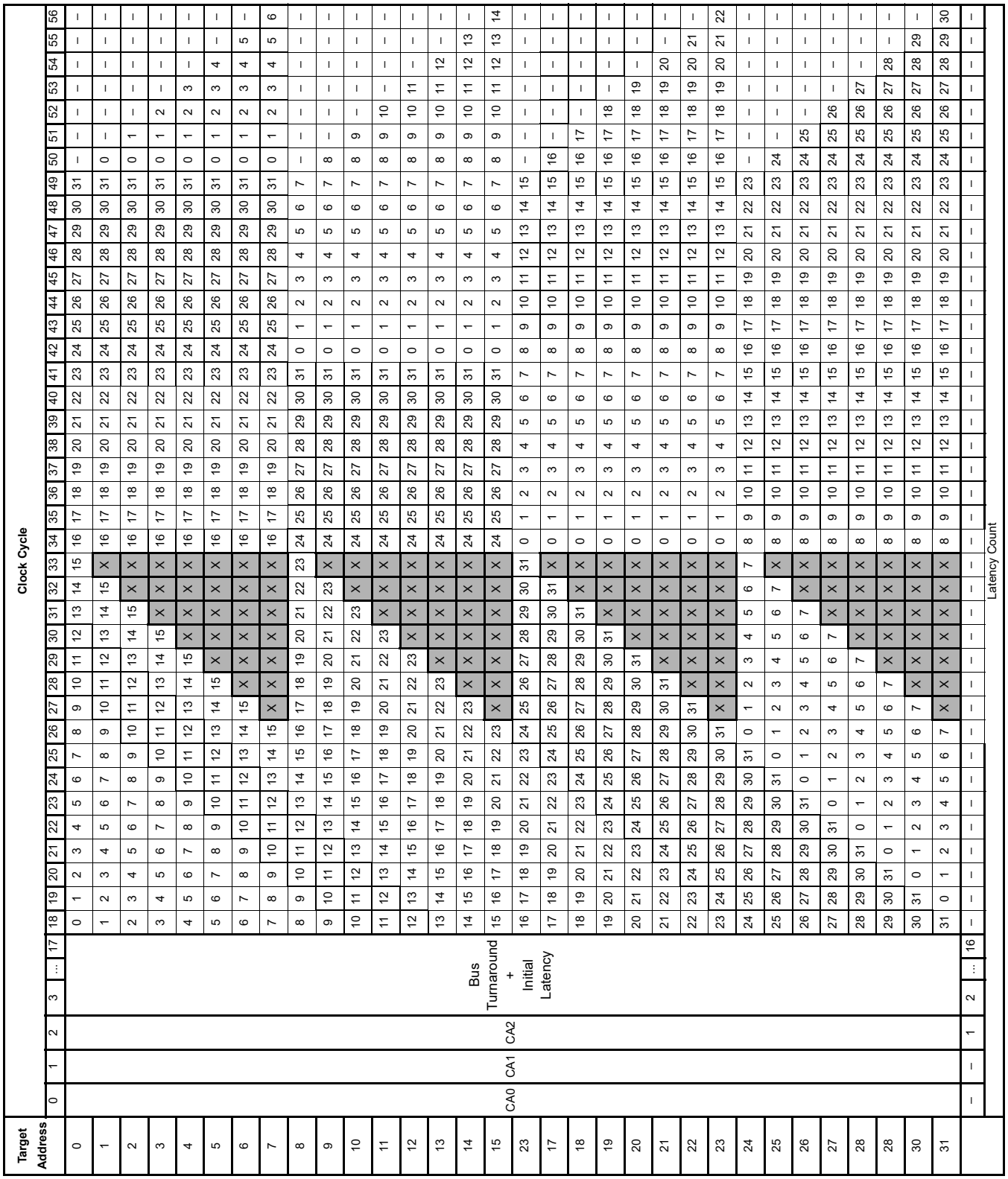
Linear burst start at a selected location and output data in a sequential manner (can read the whole memory array).

Wrapped burst accesses start at a selected location and continue for a configured number of locations in a group wrap sequence.

Table 56. Example Burst Sequences

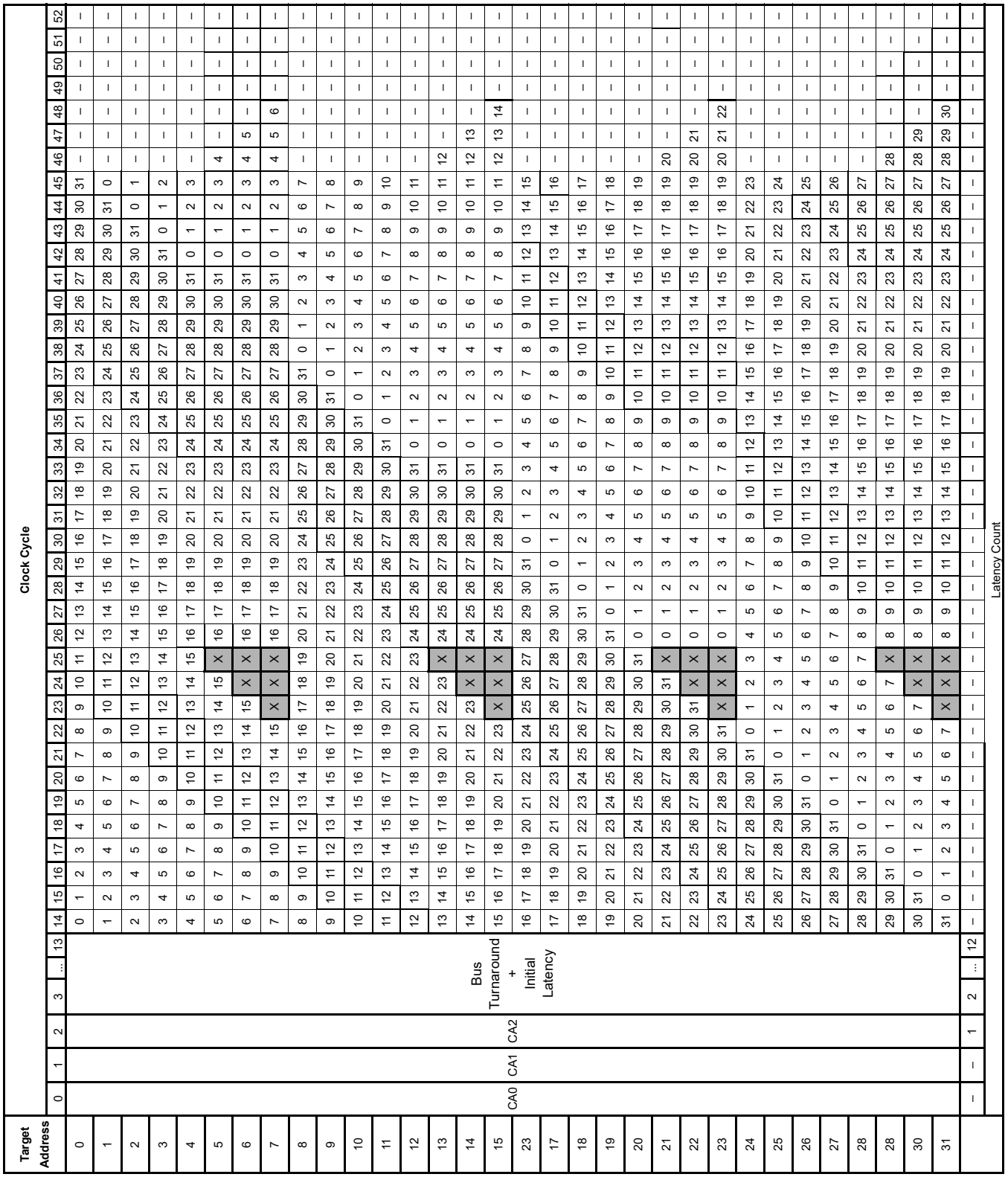
CFR1x [1:0]	CA[45]	Wrap Boundary (Bytes)	Start Address (Hex)	Address Sequence (Hex) (Words)
XX	1	Linear	XXXXXX03	03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, ...
10	0	16	XXXXXX02	02, 03, 04, 05, 06, 07, 00, 01, ...
10	0	16	XXXXXX0C	0C, 0D, 0E, 0F, 08, 09, 0A, 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, ...

Table 57. Example 1: 64-Byte Wrapped Burst Address Sequence (Latency Code = 16)



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

Table 58. Example 2: 64-Byte Wrapped Burst Address Sequence (Latency Code = 12)



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

Table 59. Example 3: 64-Byte Wrapped Burst Address Sequence (Latency Code = 20)

| Target Address | Clock Cycle After CS# Goes Low | | |

To calculate latency when crossing a page boundary, use the following formula:

```
if ((PS - LTCY) < ADDR & (SP -1))
{
      ((ADDR & (SP -1)) - PS + LTCY)
}
else
{0}
```

Where:

PS = Page size = 16 words

SP = Sub-page size = 8 words

LTCY = Latency

ADDR = Target address

Hybrid burst combines one wrapped burst followed by linear 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.

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

4.9 Write Transactions - Legacy (x1) SPI

There are write transactions for writing to the Registers.

4.9.1 Write Enable Transaction

The Write Enable (WRENB_0_0) transaction sets the Write Program Enable Status (WRPGEN) bit of the Status Register 1 (STR1V[1]) to 1. The WRPGEN bit must be set to 1 by issuing the Write Enable (WRENB_0_0) Transaction to enable write, program, and erase transactions.

4.9.2 Write Disable Transaction

The Write Disable (WRDIS_0_0) transaction clears the Write Program Enable Status (WRPGEN) bit of the Status Register 1 (STR1V[1]) to 0.

The WRPGEN bit can be cleared to 0 by issuing the Write Disable (WRDIS_0_0) transaction to disable commands that requires WRPGEN be set to 1 for execution. The WRDIS_0_0 transaction can be used by the user to protect memory areas against inadvertent write, program, or erase operations that can corrupt the contents of the memory. The WRDIS_0_0 transaction is ignored during an embedded operation while RDYBSY bit = 1 (STR1V[0]).

4.9.3 Clear Program and Erase Failure Flags Transaction

The Clear Program and Erase Failure Flags (CLPEF_0_0) transaction resets bit STR1V[5] (Erase Error Flag) and bit STR1V[6] (Program Error Flag) to 0. The Clear Status Register transaction will be accepted even when the device remains busy with RDYBSY set to 1, as the device does remain busy when either error bit is set. The WRPGEN bit will be unchanged after this command is executed.

4.9.4 Clear ECC Status Register Transaction

The Clear ECC Status Register (CLECC_0_0) transaction resets bit ECSV[4] (2-bit ECC Detection), bit ECSV[3] (1-bit ECC Correction), INSV[1:0] ECC detection status bits, Address Trap Register EATV[31:0], and ECC Detection Counter ECTV[15:0]. It is not necessary to set the WRPGEN bit before a Clear Status Register transaction is executed. The Clear Status Register transaction will be accepted even when the device remains busy with WRPGEN set to 1, as the device does remain busy when either error bit is set. The WRPGEN bit will be unchanged after this command is executed.

4.9.5 Write Any Register Transaction

The Write Any Register (WRARG_C_1) transaction provides a way to write any device register, Nonvolatile or Volatile. The transaction includes the address of the register to be written, followed by one byte of data to write in the addressed register.

Before the WRARG_C_1 transaction can be accepted by the device, a Write Enable (WRENB_0_0) transaction must be issued and decoded, which sets the Write/Program Enable bit (WRPGEN) in the Status Register to enable any write operations. The RDYDSY bit in STR1V[0] can be checked to determine when the operation is completed. The PRGERR and ERSERR bits in STR1V[6:5] can be checked to determine if any error occurred during the operation.

Some registers have a mixture of bit types and individual rules controlling which bits can be modified. Some bits are read only, some are OTP, and some are designated Reserved (DNU).

Read only bits are never modified and the related bits in the WRARG_C_1 transaction data byte are ignored without setting a program or erase error indication (PRGERR or ERSERR in STR1V[6:5]). Hence, the value of these bits in the WRARG_C_1 data byte do not matter.

OTP bits can only be programmed to the level opposite of their default state. Writing of OTP bits back to their default state is ignored and no error is set.

Nonvolatile bits which are changed by the WRARG_C_1 data require Nonvolatile Register write time (t_W) to be updated. The update process involves an erase and a program operation on the Nonvolatile Register bits. If either the erase or program portion of the update fails, the related error bit and RDYBSY bit in STR1V will be set to 1.

Status Register 1 can be repeatedly read (polled) to monitor the RDYBSY bit (STR1V[0]) and the error bits (STR1V[6,5]) to determine when the register write is completed or failed. If there is a write failure, the CLPEF_0_0 transaction is used to clear the error status and enable the device to return to standby state.

The ASP PPB Lock Register (PPLV) Register cannot be written by the WRARG_C_1 transaction. Only the Write PPB Lock Bit (WRPLB_0_0) transaction can write the PPLV Register.

The Data Integrity Check Register cannot be written by the WRARG_C_1 transaction. The Data Integrity Check Register is loaded by running the Data Integrity Check transaction (DICHK_4_1).

4.9.6 Write PPB Lock Bit

The Write PPB Lock Bit (WRPLB_0_0) transaction clears the PPB Lock Register PPLV[0] to zero. The PPBLCK bit is used to protect the PPB bits. When PPLV[0] = 0, the PPB Program/Erase transaction will be aborted. In Read Password Protection mode, PPBLCK bit is also used to control the high order bits of the address by forcing the address range to be limited to one sector where boot code is stored, until the read password is supplied.

Before the WRPLB_0_0 transaction can be accepted by the device, a Write Enable (WRENB_0_0) transaction must be issued and decoded by the device, which sets the Write/Program Enable (WRPGEN) in the Status Register 1 to enable any write operations.

While the operation is in progress, the Status Register can still be read to check the value of the RDYBSY bit. The WRPGEN bit is a 1 during the self-timed operation, and is a 0 when it is completed. When the Write PPB Lock transaction is completed, the RDYBSY bit is set to a 0.

4.9.7 Legacy (x1) SPI Write Transactions Related Registers and Transactions

Table 60. Legacy (x1) SPI Write Transactions Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
Status Register 1 (STR1N, STR1V)	Write Enable (WRENB_0_0)
Address Trap Register (EATV)	Write Any Register (WRARG_C_1)
ECC Detection Counter (ECTV)	Write PPB Lock Bit (WRPLB_0_0)

4.10 Program

There are program transactions for programming data into the device. These program transactions can use any of these two protocols:

- HyperBus interface with DDR - Program Registers, the Memory Array, SSR, Persistent Protection Bits, Dynamic Protection Bits, Clear ECC Status and Clear PPB Lock bit.
- Legacy (x1) SPI interface with SDR - Program the Memory Array, SSR, and Persistent Protection Bits.

4.10.1 HyperBus

4.10.1.1 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 State.

The four transaction Word Programming command sequence described earlier is used to program a single (16-bit) word (two bytes). 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 (512-Byte) can be programmed as long as an aligned 256 word (512-Byte) address boundary is not crossed.

The system can determine the status of the program operation by reading the Status Register (see Section 4.13 Error Types and Reporting - HyperBus on page 85).

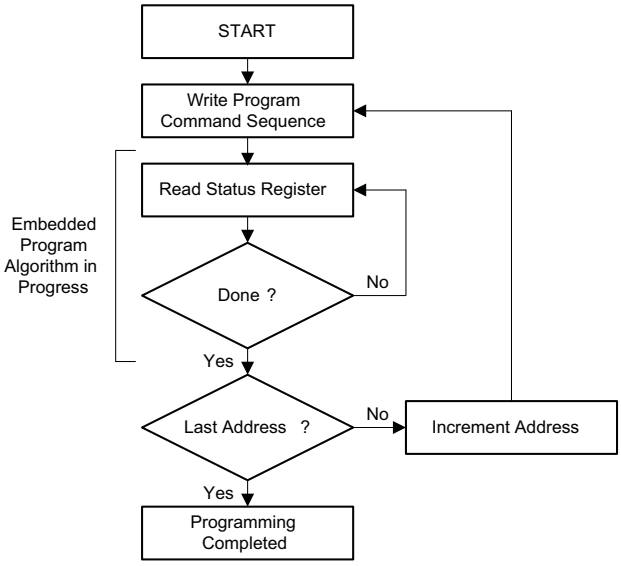
Any commands other than Program Suspend and Status Register Read written to the device during the Embedded Program algorithm are ignored.

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 Advanced Sector Protection, Password, and PPB ASOs. The same command is used to change volatile bits when entered into the PPB Lock, and DYB ASOs.

Figure 33. Word Program Operation



4.10.1.2 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-Byte 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 Section 3. Address Space Maps on page 17 for information on address map.

Write Buffer Programming allows up to 512-Byte to be programmed in one operation. It is possible to program from 1 bit up to 512-Byte 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-Byte 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.

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 (WRBFAB - STRV[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 as well when anything other than the Program Buffer to Flash is written when that command is expected at the end of the word count number of data words.

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. See Figure 34 on page 74 for a diagram of the programming operation.

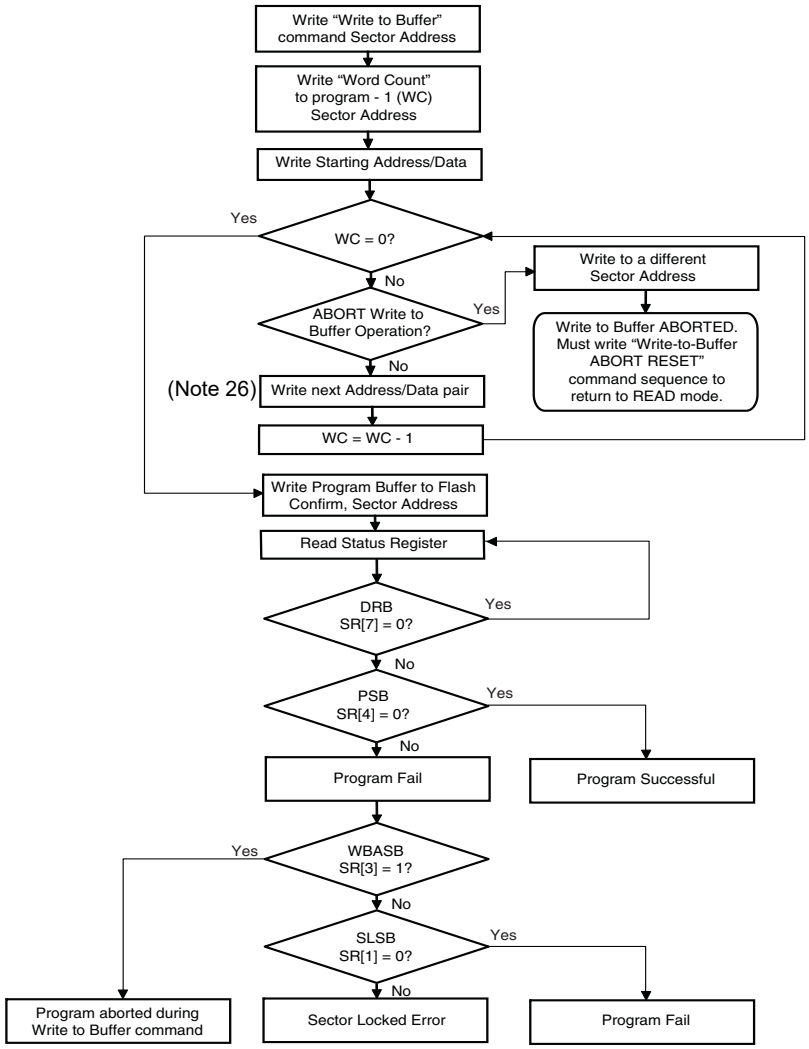
The Write Buffer Programming Sequence will be Aborted under the following conditions:

- 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 (PRGERR = 1 - STRV[4]) due to Write Buffer Abort bit location 3 (WRBFAB = 1 - STRV[3]). Write Buffer Abort Reset command or Clear Status Register command will clear the failure status.

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 34. Write Buffer Programming Operation with Status Register



Note
26. See Table 128 on page 126 for the command sequence as required for Write Buffer Programming.
27. 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.

4.10.1.3 Program Granularity

The S26HS-T/S26HL-T 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. Pages programmed by different methods may be mixed within a Line for the Industrial Temperature version (–40 °C to +85 °C) only. For the Industrial Plus version (–40 °C to +105 °C) and Automotive AEC-Q100 Grade 1 (–40 °C to +125 °C), Grade 2 (–40 °C to +105 °C) and Grade 3 (–40 °C to +85 °C) versions, the device will only support one programming operation on each page between erase operations. Moreover, Single Word Programming command is also not supported.

Word Programming and Write Buffer Programming, more than once within a half-page, is supported for legacy software compatibility. However, using Word Programming or Write Buffer Programming more than once within a half-page without an erase will disable the device's ECC functionality for that half-page. 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.

Note that if 2-bit ECC is enabled, multiple Word Programming or Write Buffer Programming within the same page will result in a Program Error.

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.

4.10.1.4 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 4.10.1.3 Program Granularity on page 75, incremental programming affects ECC syndrome bits and causes the device to disable ECC for that half-page.

Note that if 2-bit ECC is enabled, incremental Word Programming or Write Buffer Programming within the same page will result in a Program Error.

4.10.1.5 Program Register Transactions

The Program Register (PRNPOR_4_0, PGVINC_4_0, PGVINS_4_0, PGVCR1_4_0, PGVCR2_4_0, PGNCR1_4_0, PGNCR2_4_0, PGOASP_2_1, PGNPWD_2_1, PGNATB_2_1, PGOENX_2_1) transactions provide a way to program any device register, Nonvolatile or Volatile. The transactions include unlock cycles followed by one word of data to write.

4.10.1.6 Program SSR Transaction

The Program Secure Silicon transaction (PG_SSR_4_1) programs data in the SSR, which is in a different address space from the main array data and is OTP. The SSR is 1024 bytes, so the address bits from A31 to A10 must be zero for this transaction.

The PRGERR bit in STRV[4] may be checked to determine if any error occurred during the operation.

4.10.1.7 Set Dynamic Protection Bits (DYB)

The Set Dynamic Protect Bits (STVDYB_2_1) transaction sets a bit in the DYB Register to protect the addressed sector from being programed or erased.

4.10.1.8 Program Persistent Protection Bits (PPB)

The Program Persistent Protect Bits (PGNPPB_2_1) transaction programs a bit in the PPB Register to protect the addressed sector from being programed or erased.

The PRGERR bit in STRV[4] may be checked to determine if any error occurred during the operation. Program PPB bit transaction will abort when trying to program the PPB bits protected by PPBLCK (PPLV[0]) bit.

4.10.1.9 Clear ECC Status Register Transaction

The Clear ECC Status Register (CLRECC_1_1) transaction resets bit ECSV[4] (2-bit ECC Detection), bit ECSV[3] (1-bit ECC Correction), INSV[1:0] ECC detection status bits.

4.10.1.10 Clear PPB Lock Bit

The Clear PPB Lock Bit (CLVPPL_2_1) transaction clears the PPB Lock Register PPLV[0] to 0. The PPBLCK bit is used to protect the PPB bits. When PPLV[0] = 0, the PPB Program/Erase transaction will be aborted.

4.10.2 Legacy (x1) SPI

In SPI, before any program transaction can be accepted by the device, a Write Enable (WRENB_0_0) transaction must be issued which sets the Write/Program Enable (WRPGEN) bit in Status Register 1 to enable program operations. When a program transaction is completed, the WRPGEN bit is reset to a '0'.

While the program transaction is in progress, the Status Register may be read to check the value of the Device Ready/Busy (RDYBSY) bit. The RDYBSY bit is a '1' during the self-timed program transaction, and is a '0' when it is completed.

The PGMERR bit may be checked to determine if any error occurred during the program transaction.

A program transaction applied to a sector that has been Write Protected through any of the protection schemes, will not be executed and will set the PGMERR status fail bit.

4.10.2.1 Program Granularity

The HL-T/HS-T family supports multi-pass programming (bit walking) where programming a "0" over a "1" without performing the sector erase operation. Bit-walking is allowed for the non-AEC-Q100 industrial temperature range (–40°C to +85°C) of this device. It is required to perform only one programming operation (single-pass programming) on each ECC data unit between erase operations for the higher temperature range (–40°C to +105°C) and (–40°C to +125°C) devices and all AEC-Q100 devices.

Multi-pass programming without an erase operation will disable the device's ECC functionality for that data unit. Note that if 2-bit ECC is enabled, multi-pass Programming within the same sector will result in a Program Error.

4.10.2.2 Page Programming

Page Programming is done by loading a Page Buffer with data to be programmed and issuing a programming command to move data from the buffer to the memory array. This sets an upper limit on the amount of data that can be programmed with a single programming transaction. Page Programming allows up to a page size (either 256- or 512-bytes) to be programmed in one operation. The page size is determined by the Configuration Register 3 bit CFR3V[4]. The page is aligned on the page size address boundary. It is possible to program from one bit up to a page size in each Page Programming operation. It is recommended that a multiple of 16-byte length and aligned Program Blocks be written. This ensures that ECC is not disabled. For the very best Page Program throughput, programming should be done in full pages of 512 bytes aligned on 512-byte boundaries with each Page being programmed only once.

4.10.2.3 Program Page Transaction

The Program Page transaction (PRPGE_4_1) programs data into the memory array. If data more than a page size (256B or 512B) is sent to the device, then the space between the starting address and the page aligned end boundary, the data loading sequence will wrap from the last byte in the page to the zero byte location of the same page and begin overwriting any data previously loaded in the page. If less than a page of data is sent to the device, then the sent data bytes will be programmed in sequence, starting at the provided address within the page, without having any effect on the other bytes of the same page. The programming process is managed by the device internal control logic. The PRGERR bit indicates if an error has occurred in the programming transaction that prevents successful completion of programming. This includes attempted programming of a protected area (see Section 6. Transaction Table).

4.10.2.4 Program SSR Transaction

The Program Secure Silicon transaction (PRSSR_4_1) programs data in the SSR, which is in a different address space from the main array data and is OTP. The SSR is 1024 bytes, so the address bits from A31 to A10 must be zero for this transaction.

The PRGERR bit in STR1V[6] may be checked to determine if any error occurred during the operation.

To program the OTP array in bit granularity, the rest of the bits within a data byte can be set to 1.

Each SSR memory space can be programmed one or more times, provided that the region is not locked. Attempting to program zeros in a region that is locked will fail with the PRGERR bit in STR1V[6] set to 1. Programming once, even in a protected area does not cause an error and does not set PRGERR bit. Subsequent programming can be performed only on the unprogrammed bits (that is, 1 data). Programming more than once within an ECC unit will disable ECC on that data unit.

4.10.2.5 Program Persistent Protection Bit (PPB)

The Program Persistent Protect Bit (PRPPB_4_0) transaction programs a bit in the PPB Register to protect the sector of the provided address from being programed or erased.

The PRGERR bit in STR1V[6] may be checked to determine if any error occurred during the operation. Program PPB bit transaction will abort when trying to program the PPB bits protected by ASPPPB (ASPO[3]), ASPPRM (ASPO[0]) and PPBLCK (PPLV[0]) bit.

4.10.3 HyperBus Program Related Registers and Transactions

Table 61. HyperBus Program Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
HyperBus Status Register (STR1V)	Program Word (PGWORD_4_0)
HyperBus Configuration Register 1 (CFR1N, CRF1V)	PRNPOR_4_0 (PRNPOR_4_0) Program Volatile Interrupt Configuration Register (PGVINC_4_0) Program Volatile Interrupt Status Register (PGVINS_4_0) Program Volatile Configuration Register 1 (PGVCR1_4_0) Program Volatile Configuration Register 2 (PGVCR2_4_0) Program Nonvolatile Configuration Register 1 (PGNCR1_4_0) Program Nonvolatile Configuration Register 2 (PGNCR2_4_0) Program One-Time-Programmable Advanced Sector Protection Register (PGOASP_2_1) Program Nonvolatile Password (PGNPWD_2_1) Program Nonvolatile AutoBoot Register (RDATBN_1_0) Program One-Time-Programmable EnduraFlex Registers [4:0] (PGOENX_2_1)
HyperBus Advance Sector Protect Register (ASPO)	Program Secure Silicon Region Word (PG_SSR_4_1)
HyperBus ASP PPB Lock (PPLV)	Program Nonvolatile Persistent Protection Bits (PGNPPB_2_1) Set Volatile Dynamic Protection Bits (STVDYB_2_1)
	Clear ECC Error Status Failure Flags (CLRECC_1_1) Clear Status Register Failure Flags (CLVSTR_1_0)

4.10.4 Legacy (x1) SPI Program Related Registers and Transactions

Table 62. Legacy (x1) SPI Program Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
SPI Status Register 1 (STR1N, STR1V)	Write Enable (WRENB_0_0)
SPI Advance Sector Protect Register (ASPO)	Program Secure Silicon (PRSSR_4_1)
SPI ASP PPB Lock (PPLV)	Program Persistent Protection Bit (PRPPB_4_0)

4.11 Erase

There are erase transactions for erasing data bits to 1 (all bytes are FFh). These program transactions can use any of these two protocols:

- Legacy (x1) SPI Interface with SDR - Erase the Memory Array and Persistent Protection Bits.
- HyperBus Interface with DDR - Erase Registers, the Memory Array and Persistent Protection Bits.

4.11.1 HyperBus

4.11.1.1 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. See Section 4.13 Error Types and Reporting - HyperBus on page 85 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 Standby state to ensure data integrity.

Sectors protected by the ASP DYB and PPB bits will not be erased. 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.

Note that Chip Erase cannot be suspended.

4.11.2 Sector Erase

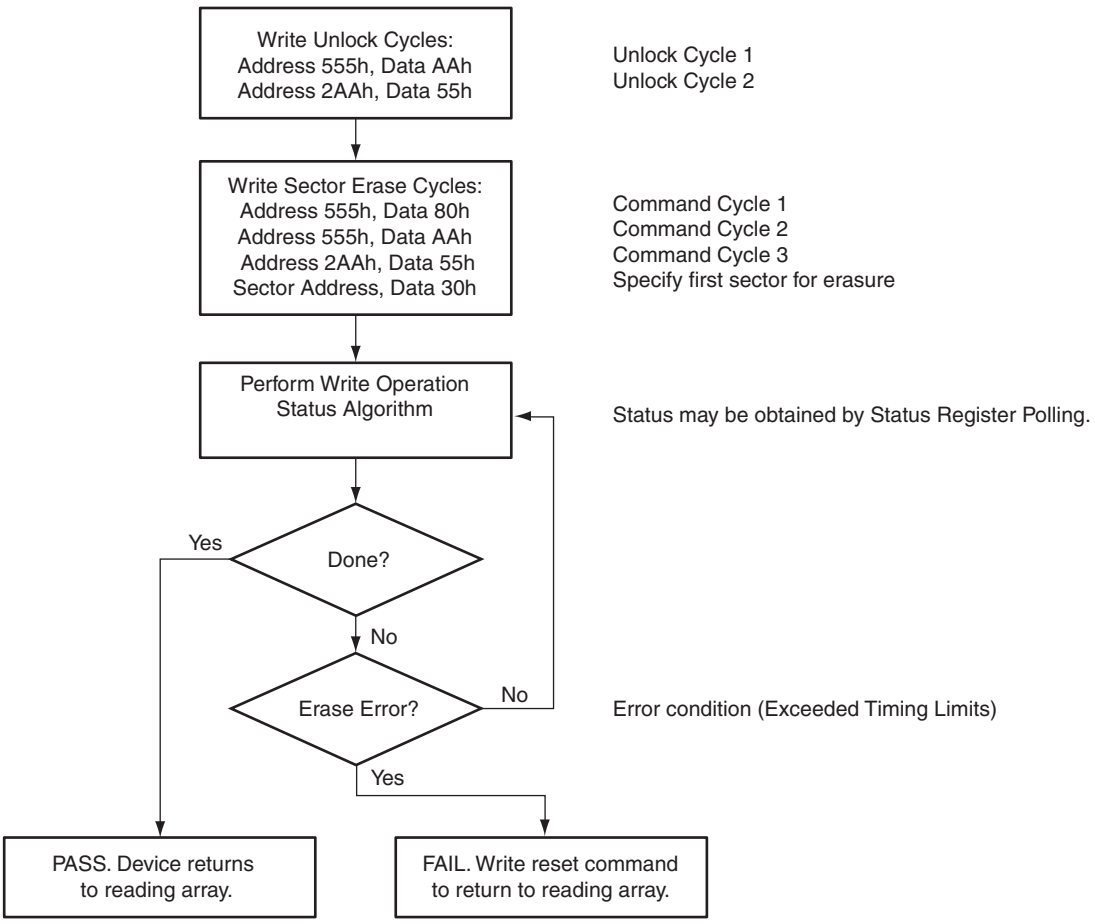
The Sector Erase function erases one sector in the memory array. The device does not require the system to preprogram prior to erase. The Embedded Erase Algorithm automatically programs and verifies the entire sector for an all 0 data pattern prior to electrical erase. After a successful sector erase, 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.

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 Section 2.4.1 Embedded Algorithm Controller (EAC) on page 14.

Sectors protected by the ASP DYB and PPB bits will not be erased. 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.

Figure 35. Sector Erase Operation



4.11.2.1 Erase Register Transactions

The Erase Register (ERNC12_3_0) transactions provide a way to erase any Nonvolatile Device Register. The transactions include unlock cycles.

The ERSERR bit in STRV[5] may be checked to determine if any error occurred during the operation.

4.11.2.2 Erase Persistent Protection Bits (PPB)

The Erase Persistent Protect Bits (ERNPPB_2_1) transaction erases all PPB bits.

The ERSERR bit in STRV[5] may be checked to determine if any error occurred during the operation. Erase PPB bit transaction will abort when trying to erase the PPB bits protected by PPBLCK (PPLV[0]) bit.

4.11.3 Erase Status and Count

4.11.3.1 Evaluate Erase Status Transaction

The Evaluate Erase Status (EVERST_1_0) transaction verifies that the last erase operation on the addressed sector was completed successfully. If the selected sector was successfully erased, then the erase status bit (SESTAT - STRV[0]) is set to 1. If the selected sector was not completely erased STRV[0] is 0.

The Evaluate Erase Status transaction can be used to detect when erase operations that have failed due to loss of power, reset, or failure during the erase operation. The transaction requires t_{EES} to complete and update the erase status in STRV. The RDYBSY bit (STRV[7]) can be read to determine when the Evaluate Erase Status transaction is completed. If a sector is found not erased with STRV[0] = 0, the sector must be erased again to ensure reliable storage of data in the sector.

4.11.3.2 Read Sector Erase Count Register Transaction

The Read Sector Erase Count (RDSECV_1_0) transaction outputs the number of erase cycles for the addressed sector. The erase cycle count is stored in the Sector Erase Count (SECV[22:0]) Register. Load Sector Address (LDSRAD_2_1) initiates loading the Sector Erase Count Register.

The transaction requires t_{SEC} to complete and update the SECV[22:0] Register. The RDYBSY bit may be read to determine when the Sector Erase Count Transaction finished. The SECV[23] bit is used to determine if the reported sector erase count is corrupted and was reset.

4.11.3.3 Blank Check Transaction

The Blank Check (BLKCHK_1_0) transaction 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.

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.

4.11.4 Legacy (x1) SPI

Before any erase transaction can be accepted by the device, a Write Enable (WRENB_0_0) transaction must be issued and decoded by the device. Erase transactions can only be executed by the device if the Write/Program Enable bit (WRPGEN) in the Status Register is set to ‘1’ to enable erase operations. When an erase transaction is completed, the WRPGEN bit is reset to a ‘0’.

While the erase transaction is in progress, the Status Register 1 may be read to check the value of the Device Ready/Busy (RDYBSY) bit. The RDYBSY bit is a ‘1’ during the self-timed erase transaction, and is a ‘0’ when it is completed.

The ERSERR bit in STR1V[5] can be checked to determine if any error occurred during the erase transaction.

An erase transaction applied to a sector that has been Write Protected through the Block Protection bits or ASP, will not be executed and will set the ERSERR status fail bit.

Erase transactions will be initiated when CS# is driven into the logic High state.

4.11.5 Erase 4 KB Sector Transaction

The Erase 4 KB Sector (ER004_4_0) transaction sets all the bits of a 4 KB sector to 1 (all bytes are FFh).

This transaction is ignored when the device is configured for uniform sectors only (CFR3V[3] = 1). If the Erase 4 KB sector transaction is issued to a non-4 KB sector address, the device will abort the operation and will not set the ERSERR status fail bit.

4.11.6 Erase 256 KB Sector Transaction

The Erase 256 KB Sector (ER256_4_0) transaction sets all bits in the addressed sector to 1 (all bytes are FFh).

A device configuration option (CFR3V[3]) determines if the Hybrid Sector Architecture is in use. When CFR3V[3] = 0, 4 KB sectors overlay a portion of the highest or lowest address 128 KB or 64 KB of the device address space. If a sector erase command is applied to a 256 KB sector that is overlaid by 4 KB sectors, the overlaid 4 KB sectors are not affected by the erase. Only the visible (non-overlaid) portion of the 128 KB or 192 KB sector is erased. When CFR3V[3] = 1, there are no 4 KB sectors in the device address space and the Sector Erase command always operates on fully visible 256 KB sectors.

When BLKCHK is enabled an erase transaction first evaluates the erase status of the sector. If the sector is found to erased, the erase operation is aborted. The erase operation is only executed if programmed bits are found in the sector. Disabling BLKCHK executes an erase operation unconditionally.

4.11.7 Erase Chip Transaction

The Erase Chip (ERCHP_0_0) transaction sets all bits to 1 (all bytes are FFh) inside the entire flash memory array.

An Erase Chip transaction can be executed only when the Block Protection (BP2, BP1, BP0) bits are set to 0's. If the BP bits are not zero, the transaction is not executed and ERSERR status fail bit is not set. The transaction will skip any sectors protected by the Advance Sector Protection DYB or PPB and the ERSERR status fail bit will not be set.

4.11.8 Erase Persistent Protection Bit (PPB) Transaction

The Erase PPB transaction sets all PPB bits to 1. This transaction will abort if PPB bits are protected by ASPPPB (ASPO[3]), ASPPRM (ASPO[0]) and PPBLCK (PPLV[0]) bit.

4.11.9 Erase Status and Count

4.11.9.1 Evaluate Erase Status Transaction

The Evaluate Erase Status (EVERS_4_0) transaction verifies that the last erase operation on the addressed sector was completed successfully. If the selected sector was successfully erased, then the erase status bit (STR2V[2]) is set to 1. If the selected sector was not completely erased STR2V[2] is 0. The Write/Program Enable transaction (to set the WRPGEN bit) is not required before this transaction. However, the RDYBSY bit is set by the device itself and cleared at the end of the operation, as visible in STR1V[0] when reading status.

The Evaluate Erase Status transaction can be used to detect when erase operations that have failed due to loss of power, reset, or failure during the erase operation. The transaction requires t_{EES} to complete and update the erase status in STR2V. The RDYBSY bit (STR1V[0]) can be read to determine when the Evaluate Erase Status transaction is completed. If a sector is found not erased with STR2V[2] = 0, the sector must be erased again to ensure reliable storage of data in the sector.

4.11.9.2 Sector Erase Count Transaction

The Sector Erase Count (SEERC_4_0) transaction outputs the number of erase cycles for the addressed sector. The erase cycle count is stored in Sector Erase Count (SECV[22:0]) Register, and can be read by using the Read Any Register transaction (RDARG_4_0). The RDYBSY bit is set by the device itself and cleared at the end of the operation, as visible in STR1V[0] when reading status.

The transaction requires t_{SEC} to complete and update the SECV[22:0] Register. The RDYBSY bit (STR1V[0]) may be read, to determine when the Sector Erase Count Transaction is finished. The SECV[23] bit is used to determine if the reported sector erase count is corrupted and was reset.

4.11.10 HyperBus Erase Related Registers and Transaction

Table 63. HyperBus Erase Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
HyperBus Status Register (STRV)	Erase Sector (ERSCTR_6_0)
HyperBus Configuration Register 1 (CFR1N, CFR1V)	Erase Chip (ERCHIP_6_0)
SPI ASP PPB Lock (PPLV) HyperBus ASP PPB Lock (PPLV)	Erase Nonvolatile Persistent Protection Bits (ERNPPB_2_1)
HyperBus Sector Erase Count Register (SECV)	Evaluate Erase Status (EVERST_1_0)
	Sector Erase Count (RDSECV_1_0)

4.11.11 Legacy (x1) SPI Erase Related Registers and Transaction

Table 64. Legacy (x1) SPI Erase Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
SPI Status Register 1 (STR1N, STR1V)	Write Enable (WRENB_0_0)
SPI Status Register 2 (STR2V)	Erase 4 KB Sector (ER004_4_0)
SPI ASP PPB Lock (PPLV)	Erase Chip (ERCHP_0_0)
SPI Sector Erase Count Register (SECV)	Sector Erase Count (SEERC_4_0)

4.12 Suspend and Resume Embedded Operation

HL-T/HS-T device can interrupt and suspend the running embedded operations such as Erase, Program, or Data Integrity Check. It can also resume the suspended operation once the host finishes the intermediate operation and sends the respective resume transaction to the device.

4.12.1 Erase, Program, or Data Integrity Check Suspend

The Suspend transaction allows the system to interrupt a program, erase or data integrity check operation and then read from any other non erase-suspended sector, non-program-suspended-page or the array. The Device Ready/Busy Status Flag (RDYBSY) in Status Register (SPI - STR1V[0], HyperBus STRV[7]) must be checked to know when the program, erase or data integrity check operation has stopped.

4.12.1.1 Program Suspend

- Program Suspend is valid only during a programming operation.
- The Program Operation Suspend Status flag (SPI - PROGMS STR2V[0], HyperBus - PROGMS STRV[2]) can be used to determine if a programming operation has been suspended or was completed at the time RDYSY changes to 0.
- A program operation can be suspended to allow a read operation.
- Reading at any address within a program-suspended page produces undetermined data.

4.12.1.2 Erase Suspend

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 operation. The Erase Suspend command is ignored if written during the chip erase operation.

- Erase Suspend is valid only during a sector erase operation.
- The Erase operation Suspend status flag (SPI - ERASES STR2V[1], HyperBus ERASES STRV[6]) can be used to determine if an erase operation has been suspended or was completed at the time RDYBSY changes to 0.
- A Bulk Erase operation cannot be suspended.
- An Erase operation can be suspended to allow a program operation or a read operation. After an erase-suspended program operation is complete, the device returns to the erase-suspend mode.
- A new erase operation is not allowed with an already suspended erase, program, or data integrity check operation. An erase command is ignored in this situation.
- If a program failure occurs during erase suspend the Status Register Clear or Software Reset transactions will return the device to the erase suspended state.
- Reading at any address within an erase-suspended sector produces undetermined data.

4.12.1.3 Data Integrity Check Suspend - Legacy SPI Only

- Data Integrity Check Suspend is valid only during a Data Integrity Check Calculation operation.
- The Memory Array Data Integrity Check CRC Suspend Status Flag (SPI - DICRCS STR2V[4], HyperBus - DICRCS STRV[8]) can be used to determine if a data integrity check operation has been suspended or was completed at the time RDYBSY changes to 0.
- A data integrity check operation can be suspended to allow a read operation.
- The time required for the suspend operation to complete is t_{PEDS}.

The system can determine the status of the program operation by reading the RDYBSY bit in the Status Register 1, just as in the standard program operation.

Table 65 lists the transactions allowed during the suspend operation.

Table 65. HyperBus Transactions Allowed During Suspend

Transaction Name	Allowed During Erase Suspend	Allowed During Program Suspend	Allowed During Data Integrity Check Suspend
Read (RDMARY_1_0) - Non Suspended Sector	Yes	Yes	Yes
Read Status Register (RDVSTR_2_0)			
Software Reset / ASO Exit (SRASOE_1_0)			
Resume Program (RSPROG_1_0)	No		No
Program Volatile Interrupt Configuration Register (PGVINC_4_0)	Yes		Yes
Read Volatile Interrupt Configuration Register (RDVINC_4_0)			
Program Volatile Interrupt Status Register (PGVINS_4_0)			
Read Volatile Interrupt Status Register (RDVINS_4_0)		No	No
Program Word (PGWORD_4_0) - Non Suspended Sector			
Program Write Buffer (LDBUFR_6_0, PGBFCM_1_0, RSTWBA_3_0)			
Read Volatile Configuration Register 1 & 2 (RDVCR1_4_0, RDVCR2_4_0)		Yes	Yes
Read Nonvolatile Configuration Register 1 & 2 (RDNCR1_4_0, RDNCR2_4_0)		No	No
Resume Erase (RSERSE_1_0)			
Device ID/Unique ID/SFDP ASO (IDSFE1_3_1, IDSFE2_1_1, RDIDSF_1_1, ASOEXT_1_1)		Yes	Yes
Secure Silicon Region ASO (SSRENT_3_1, RD_SSR_1_1, PG_SSR_4_1, LDBSSR_5_1, PGCSSR_1_1, RSWSSR_3_1, ASOEXT_1_1)			

Table 66. Legacy (x1) SPI Transactions Allowed During Suspend

Transaction Name	Allowed During Erase Suspend	Allowed During Program Suspend	Allowed During Data Integrity Check Suspend
Write Disable (WRDIS_0_0)	Yes	No	No
Read Status Register 1 (RDSR1_0_0, RDSR1_4_0)		Yes	Yes
Write Enable (WRENB_0_0)		No	No
Read Status Register 2 (RDSR2_0_0, RDSR2_4_0)		Yes	Yes
Program Page (PRPGE_4_1)		No	No
Read ECC Status (RDECC_4_0)		Yes	Yes
Clear ECC Status Register (CLECC_0_0)			
Read PPB Lock Bit (RDPLB_0_0, RDPLB_4_0)		Yes	Yes
Resume Program / Erase / Data Integrity Check (RSEPD_0_0)			
Program SSR (PRSSR_4_1)		No	No
Read SSR (RDSSR_4_0)		Yes	Yes
Read Unique ID (RDUID_0_0, RDUID_4_0)			
Read SFDP (RSFDP_3_0, RSFDP_4_0)			
Read Interface CRC Register (RDCRC_4_0)		Yes	Yes
Read Any Register (RDARG_4_0)			
Software Reset Enable (SRSTE_0_0)			

Table 66. Legacy (x1) SPI Transactions Allowed During Suspend (Continued)

Transaction Name	Allowed During Erase Suspend	Allowed During Program Suspend	Allowed During Data Integrity Check Suspend
Clear Program and Erase Failure Flags (CLPEF_0_0)	Yes	Yes	Yes
Software Reset (SFRST_0_0)			
Read Identification Register (RDIDIN_0_0, RDIDIN_4_0) (manufacturer and device identification)		Yes	Yes
Suspend Program / Erase / Data Integrity Check (SPEPD_0_0)		No	No
Read DYB (RDDYB_4_0)		Yes	Yes
Read PPB (RDPPB_4_0)			

4.12.2 Erase, Program, or Data Integrity Check Resume

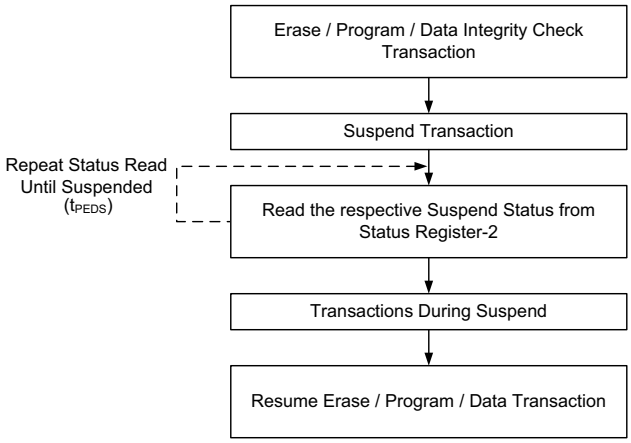
An Erase, Program, or Data Integrity Check Resume transaction must be written to resume a suspended operation. After program or read operations are completed during a Program, Erase or Data Integrity Check suspend, the Resume transaction is sent to resume the suspended operation.

After an Erase, Program, or Data Integrity Check Resume transaction is issued, the RDYBSY bit in Status Register will be set to a 1 and the programming operation will resume if one is suspended. If no program operation is suspended, the suspended erase operation will resume. If there is no suspended program, erase or data integrity check operation, the resume transaction is ignored.

Program, Erase, or Data Integrity Check operations may be interrupted as often as necessary. For example, a program suspend transaction could immediately follow a program resume transaction, but for a program or erase operation to progress to completion there must be some period of time between resume and the next suspend transaction greater than or equal to t_{PEDRS}.

Figure 36 shows the flow of suspend and resume operation.

Figure 36. Suspend and Resume Sequence



4.12.3 HyperBus Suspend and Resume Related Registers and Transactions

Table 67. HyperBus Suspend and Resume Related Registers and Transactions

Related Registers (see Section 5.2 HyperBus Registers on page 94)	Related HyperBus Transactions (see Table 128 on page 126)
HyperBus Status Register (STRV)	Suspend Erase (SPERSE_1_0)
	Resume Erase (RSERSE_1_0)
	Suspend Program (SPPROG_1_0)
	Resume Program (RSPROG_1_0)
	Read Status Register (RDVSTR_2_0)

4.12.4 Legacy (x1) SPI Suspend and Resume Related Registers and Transactions

Table 68. Legacy (x1) SPI Suspend and Resume Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
SPI Status Register 1 (STR1N, STR1V)	Suspend Erase / Program / Data Integrity Check (SPEPD_0_0)
SPI Status Register 2 (STR2V)	Resume Erase / Program / Data Integrity Check (RSEPD_0_0)
	Read Any Register (RDARG_4_0)
	Read Status Register -1 (RDSR1_0_0)
	Read Status Register -2 (RDSR2_0_0)

4.13 Error Types and Reporting - HyperBus

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.

4.13.1 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, 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.

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].

4.13.2 Write Buffer Abort (Program Abort / DICRC 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 DS and return indeterminate data.

4.14 Error Typer and Reporting - Legacy (x1) SPI

There are two 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.

Table 69 and Table 70 provide the details for clearing the error status.

Table 69. Program Error (PRGERR) Summary

Error Flag	Symbol	Conditions
Program Error	PRGERR	Bits cannot be programmed '1' to '0'
		Trying to program in a protected region
		If ASP0[2] or ASP0[1] is 0, any Nonvolatile Register write attempting to change the value of CFR1N[6:2]/CFR1V[6:2]
		After the Password Protection Mode is selected and ASP Password Register update transaction executed
		SafeBoot Failure
		Configuration Failure

Table 70. Erase Error (ERSERR) Summary

Error Flag	Symbol	Conditions
Erase Error	ERSERR	Sector Device Erase - All bits cannot be erased to '1's
		Trying to erase a protected region
		Register Erase - All bits cannot be erased to '1's during Erase portion of Register Write
		SafeBoot Failure

4.15 Reset

HL-T/HS-T devices support four types of reset mechanisms.

- Hardware Reset (Using RESET# input pin)
- Power-On Reset (POR)
- CS# Signaling Reset
- Software Reset - SPI Only

4.15.1 Hardware Reset (Using RESET# Input Pin)

The RESET# input initiates the reset operation with a transition from logic High to logic Low for > t_{RP}, and causes the device to perform the full reset process that is performed during POR. The hardware reset process requires a period of t_{RH} to complete. See Table 136 on page 147 for timing specifications.

Figure 37. Hardware Reset using RESET# Input (Reset Pulse = t_{RP}(Min))

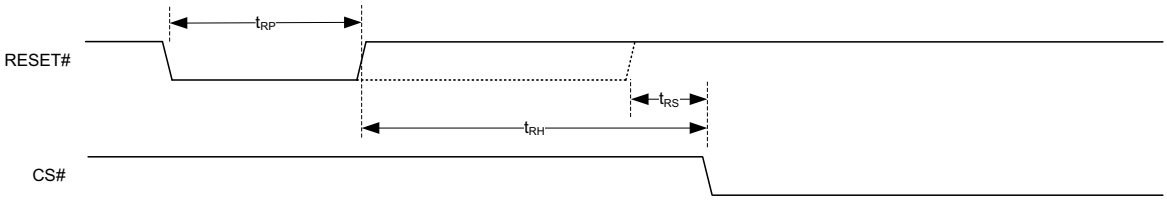
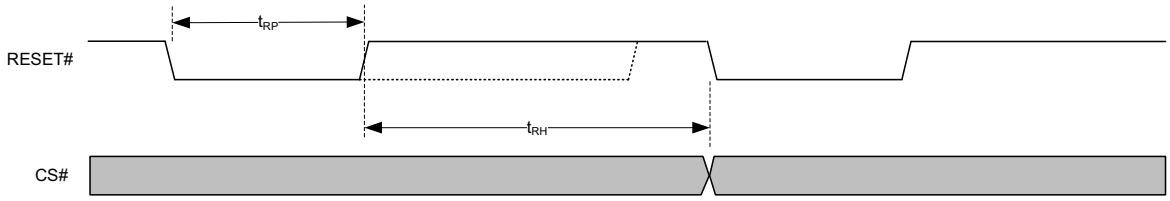


Figure 38. Hardware Reset using RESET# Input (Reset Pulse > (t_{RP} + t_{RH}))



Figure 39. Hardware Reset using RESET# Input (Back to back Hardware Reset)



4.15.2 Power-On Reset (POR)

The device executes a POR process until a time delay of t_{PU} has elapsed after the moment that V_{CC} rises above the minimum V_{CC} threshold (see Figure 40 and Figure 41). The device must not be selected during power-up (t_{PU}). Therefore, CS# must rise with V_{CC}. No commands may be sent to the device until the end of t_{PU}. See Table 136 for timing specifications.

RESET# is ignored during POR. If RESET# is LOW during POR and remains LOW through and beyond the end of t_{PU}, CS# must remain HIGH until t_{RS} after RESET# returns HIGH.

Figure 40. Reset LOW at the End of POR

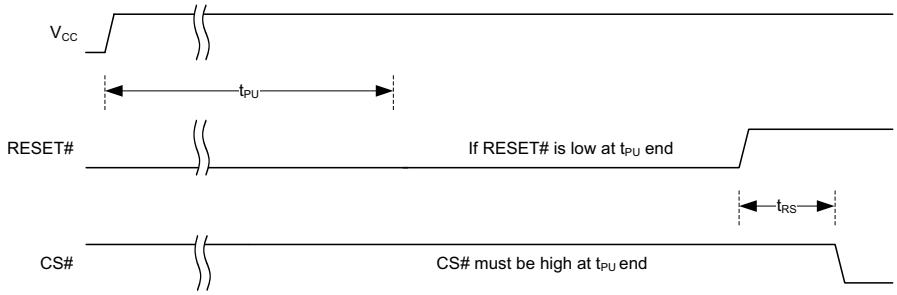
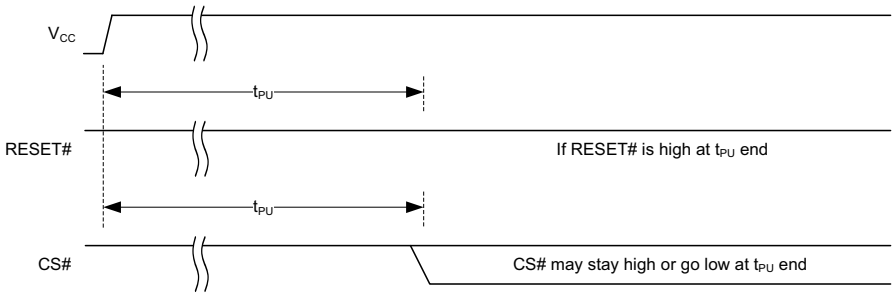


Figure 41. Reset HIGH at the End of POR



4.15.3 CS# Signaling Reset

The CS# Signaling Reset requires CS# and DQ0 signals. This reset method defines a signaling protocol, using existing signals, to initiate an SPI Flash hardware reset, independent of the device operating mode or number of package pins.

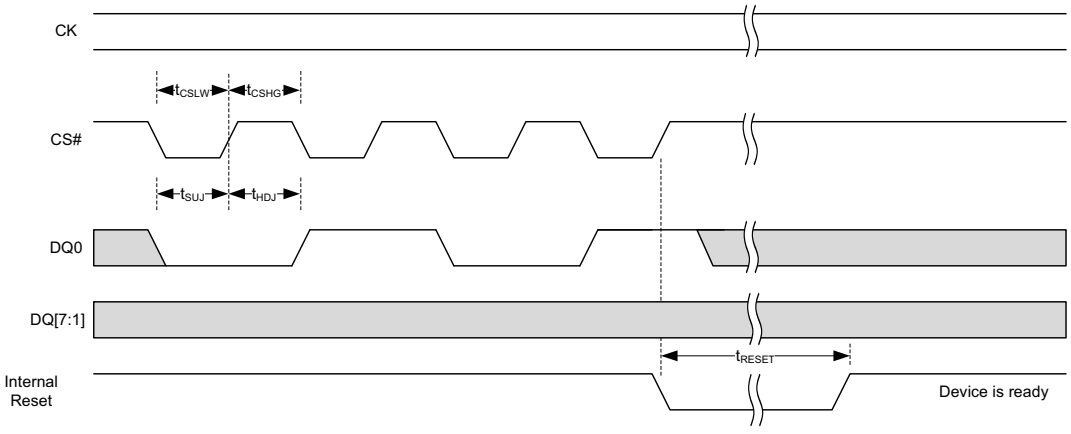
The Signaling Protocol is shown in Figure 42. See Table 136 on page 147 for timing specifications. The CS# signaling reset steps are as follows:

- CS# is driven active LOW.
- CK remains stable in either HIGH or LOW state.
- CS# and DQ0 are both driven LOW.
- CS# is driven HIGH (inactive).
- Repeat the above four steps, each time alternating the state of DQ0 for a total of four times.
- Reset occurs after the fourth CS# cycle completes and it goes HIGH (inactive).

After the fourth CS# pulse, the slave triggers its internal reset, the device terminates any operation in progress, makes all outputs high impedance, and ignores all read/write transactions for the duration of t_{RESET}. Then the device will be in standby state.

This reset sequence is not intended to be used at normal power on, but to be used only when the device is not responding to the system. This reset sequence will be operational from any state that the device may be in. Hence, CS# signaling reset is useful for packages that don't support a RESET# pin to provide behavior identical to Hardware Reset.

Figure 42. CS# Signaling Reset Protocol



4.15.4 Software Reset - SPI Only

Software controlled Reset transaction restores the device to its initial power up state, by reloading Volatile registers from nonvolatile default values except the protection registers. A reset transaction (SFRST_0_0) is executed when CS# is brought HIGH at the end of the transaction and requires tSR time to execute. See Table 136 on page 147 for timing specifications.

The Reset Enable (SRSTE_0_0) transaction is required immediately before a Reset transaction (SFRST_0_0) such that a software reset is a sequence of the two transactions. Any transaction other than SFRST_0_0 following the SRSTE_0_0 transaction will clear the reset enable condition and prevent a later SFRST_0_0 transaction from being recognized.

The Reset (SFRST_0_0) transaction immediately following a SRSTE_0_0 transaction, initiates the software reset process. During software reset, only RDSR1_4_0 and RDARG_4_0 of Status Register 1 are supported operations as long as the volatile and nonvolatile configuration states of the device are the same. if the configuration state is changing during software reset, reading Status Register 1 should only be done after the software reset time has elapsed.

The software reset is independent of the state of RESET#. If RESET# is HIGH or Unconnected, and the software reset transactions are issued, the device will perform software reset.

4.15.4.1 Software Reset Related Registers and Transactions

Table 71. Software Reset Related Registers and Transactions

Related Registers (see Section 5.3 Legacy (x1) SPI Registers on page 110)	Related SPI Transactions (see Table 130 on page 140)
N/A	Software Reset Enable (SRSTE_0_0)
	Software Reset (SFRST_0_0)

4.15.5 Reset Behavior

Table 72. Reset Behavior

Transaction / Register Name	POR	Hardware Reset and CS# Signaling Reset	Software Reset (x1 Mode Only)
Summary	• Device Reset • Status Bits Reset • All Volatile Registers Reset • Configuration Reload to Default • Volatile Protection Reset to Default • Nonvolatile Protection unchanged • Reset all Embedded operations	• Device Reset • Status Bits Reset • All Volatile Registers Reset • Configuration Reload to Default • Volatile Protection Reset to Default • Nonvolatile Protection unchanged • Reset all Embedded operations	• Device Reset • Status Bits Reset • Configuration Reload to Default • Volatile Protection Reset to Default • Nonvolatile Protection unchanged • Reset all Embedded operations
Interface Requirements	• All Inputs - Ignored • All Outputs - Tristated	• All Inputs - Ignored • All Outputs - Tristated	Transactions (SRSTE_0_0, SFRST_0_0)
Status Registers	Load from Nonvolatile Registers	Load from Nonvolatile Registers	Load from Nonvolatile Registers
Configuration Registers	Load from Nonvolatile Registers	Load from Nonvolatile Registers	Load from Nonvolatile Registers
Protection Registers	PPB Lock Register - Load based on ASPO[2:1]	PPB Lock Register - Load based on ASPO[2:1]	PPB Lock Register - No Change
	DYB Access Register - Load based on ASPO[4]	DYB Access Register - Load based on ASPO[4]	DYB Access Register - No Change
	Password Register - Load based on ASPO[2] and ASPO[0]	Password Register - Load based on ASPO[2] and ASPO[0]	Password Register - No Change
ECC Status Register	Load 0x00	Load 0x00	Load 0x00
AutoBoot Register	Load from Nonvolatile registers	Load from Nonvolatile registers	No Change
Data Integrity Check Register	Load 0x00	Load 0x00	Load 0x00
Interface CRC Register			
ECC Error Count Register			
Address Trap Register			
EnduraFlex Register	Load from Nonvolatile Registers	Load from Nonvolatile Registers	No Change
I/O Mode			
Memory/Register Erase in Progress	Not Applicable	Abort Erase	Abort Erase
Memory/Register Program in Progress		Abort Program	Abort Program
Memory/Register Read in Progress		Abort Read	Not Applicable

4.16 Power Modes

4.16.1 Active Power and Standby Power Modes

The device is enabled and in the Active Power mode when Chip Select (CS#) is LOW. When CS# is HIGH, the device is disabled, but may still be in an Active Power mode until all program, erase, and write operations have completed. The device then goes into the Standby Power mode, and power consumption drops to I_{SB}. See Table 134 on page 145 for parameter specifications.

4.16.2 Deep Power Down (DPD) Mode

Although the standby current during normal operation is relatively low, standby current can be further reduced with the DPD mode. The lower power consumption makes the DPD mode especially useful for battery powered applications.

4.16.2.1 Enter DPD

The device can enter DPD mode in two ways:

1. Enter DPD Mode using Transaction
2. Enter DPD Mode upon Power-up or Reset

Enter DPD Mode using the Enter Deep Power Down Mode Transaction

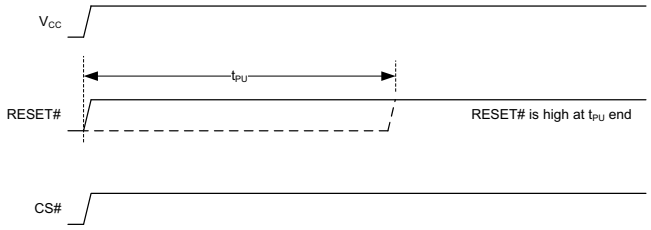
The DPD mode is enabled by sending the Enter Deep Power Down Mode Transaction (ENDPD_0_0, ENTDPD_3_0). After CS# is driven High, the power-down state will be entered within the time duration of t_{ENTDPD} (see Table 136 on page 147 for timing specifications) and power consumption drops to I_{DPD}.

DPD can only be entered from an idle state. The DPD transaction is accepted only while the device is not performing an embedded algorithm as indicated by the Ready/Busy Status Flag (RDYBSY). It is not allowed to send any transaction to device during t_{ENTDPD} time.

Enter DPD Mode upon Power-up or Reset

If the DPDPOR configuration bit is enabled, the device will be in DPD mode after the completion of Power-up, Hardware Reset or CS# Signaling Reset. During POR or Reset the CS# should follow the voltage applied on VCC to enter DPD mode as shown in Figure 43. It is not allowed to send any transaction to device during t_{ENTDPD} time.

Figure 43. Enter DPD Mode Upon Power-Up or Reset



4.16.2.2 Exit DPD

Device leaves DPD mode in one of the following ways:

Exit DPD Mode upon Hardware Reset

When the device is in DPD and DPDPOR = 0, a Hardware reset will return the device to Standby mode.

Exit DPD Mode upon CS# Pulse

When the device is in DPD or DPDPOR = 1, a CS# pulse of width t_{CSDPD} will bring the device out of DPD mode. The CS# should be driven HIGH after the pulse. HIGH to LOW transition on CS# is required to start a transaction cycle after the DPD exit. It takes t_{EXTDPD} to come out of DPD mode. The device will not respond until after t_{EXTDPD}.

Figure 44. Exit DPD Mode



The device maintains its configuration during DPD, meaning the device exits DPD in the same state as it entered. Registers such as the ECC Status, ECC Error Detection Counter, Address Trap, and Interrupt Status Registers will be cleared.

4.16.2.3 HyperBus DPD Related Registers and Transactions

Table 73. HyperBus DPD Related Registers and Transactions

Related Registers (see Section 5. Registers on page 93)	Related HyperBus Transactions (see Table 128 on page 126)
Configuration Register 1(CFR1N, CFR1V)	Enter Deep Power Down Mode (ENTDPD_3_0)

4.16.2.4 Legacy (x1) SPI DPD Related Registers and Transactions

Table 74. Legacy (x1) SPI DPD Related Registers and Transactions

Related Registers (see Section 5. Registers on page 93)	Related SPI Transactions (see Table 130 on page 140)
Configuration Register 4 (CFR4N, CFR4V)	Enter Deep Power Down Mode (ENDPD_0_0)

4.17 Power Up and Power Down

The device must not be selected at power up or power down until V_{CC} reaches the correct value as follows:

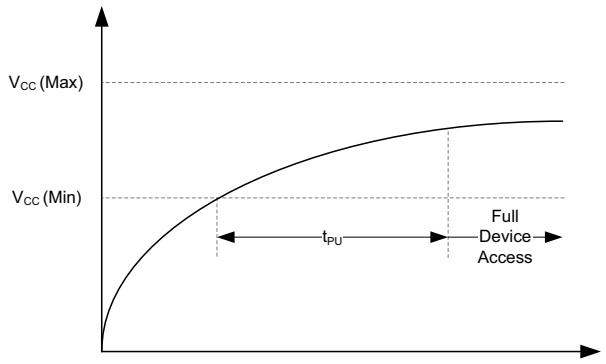
- V_{CC} (min) at power up, and then for a further delay of t_{PU}
- V_{SS} at power down

4.17.1 Power Up

The device ignores all transactions until a time delay of t_{PU} has elapsed after the moment that V_{CC} rises above the minimum V_{CC} threshold (see Figure 45). However, correct operation of the device is not guaranteed if V_{CC} returns below V_{CC} (min) during t_{PU}. No command should be sent to the device until the end of t_{PU}.

The device draws I_{POR} current during t_{PU}. After power up (t_{PU}), there is the option to be in the DPD mode or Standby mode. The DPDPOR bit in Configuration Register controls if the device will be in DPD or Standby mode after the completion of POR. If the DPDPOR bit is enabled, the device is in DPD mode after power up. A Hardware reset (RESET#) is required to return the device to Standby mode after POR.

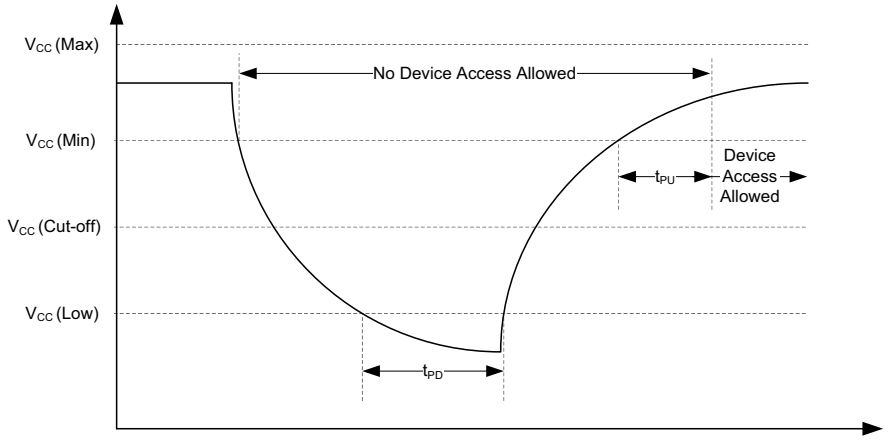
Figure 45. Power Up



4.17.2 Power Down

During power down or voltage drops below V_{CC}(cut-off), the voltage must drop below V_{CC}(Low) for a period of t_{PD} for the part to initialize correctly on power up (see Figure 46). If during a voltage drop the V_{CC} stays above V_{CC}(cut-off) the part will stay initialized and will work correctly when V_{CC} is again above V_{CC}(min). In the event POR did not complete correctly after power up, the assertion of the RESET# signal will restart the POR process.

Figure 46. Power Down and Voltage Drop

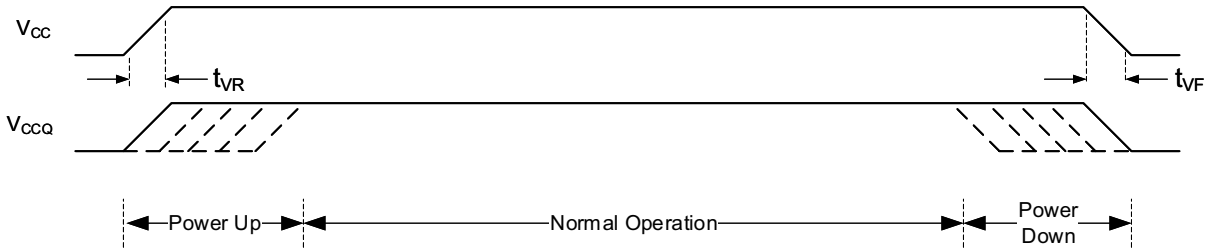


4.17.3 Power Up and Power Down Sequence

The following power sequence needs to be followed for the guaranteed reliable operation of HL-T/HS-T devices.

- Apply V_{CC} before V_{CCQ} during power up sequence. V_{CC} and V_{CCQ} can be applied simultaneously during power up, as long as V_{CCQ} does not exceed V_{CC}.
- During the power down mode, reduce the V_{CCQ} before V_{CC}. V_{CC} and V_{CCQ} can be reduced simultaneously during power down, as long as V_{CCQ} does not exceed V_{CC}.
- It is recommended to keep $V_{CCQ} \leq V_{CC}$.

Figure 47. Power Up and Power Down Sequence



5. Registers

Registers are small groups of storage cells used to configure as well as report the status of the device operations. HL-T/HS-T family of devices use separate nonvolatile and volatile storage groups to implement the different register bit types for legacy compatibility as well as new functionality. Each register is organized as a group of volatile bits with associated nonvolatile bits (if permanence is required). During power-up, hardware reset or software reset, the data in the nonvolatile bits of the register is transferred to the volatile bits to provide the default state of the volatile bits. When writing new data to nonvolatile bits of the register, the volatile bits are also updated with the new data. However, when writing new data to the Volatile Register bits the nonvolatile bits retain the old data. The register structure is shown in Figure 48.

Figure 48. Register Structure

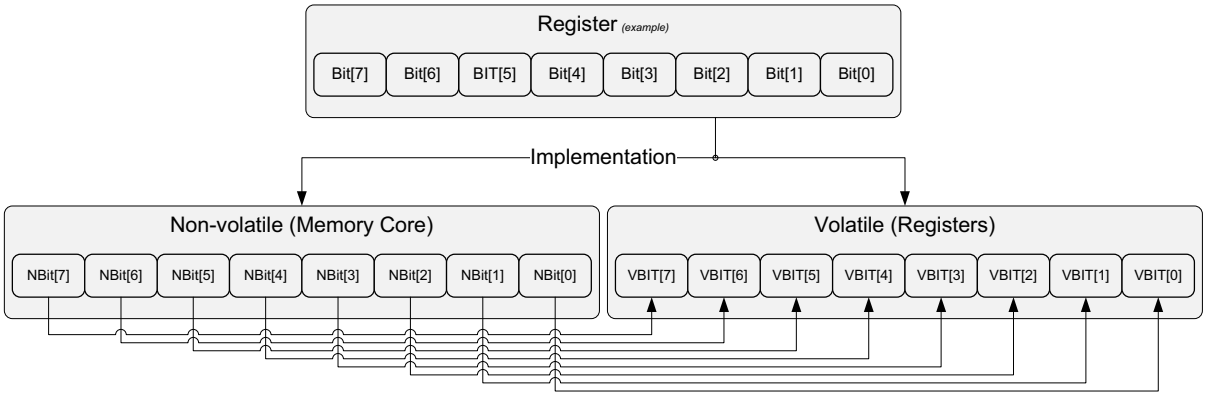
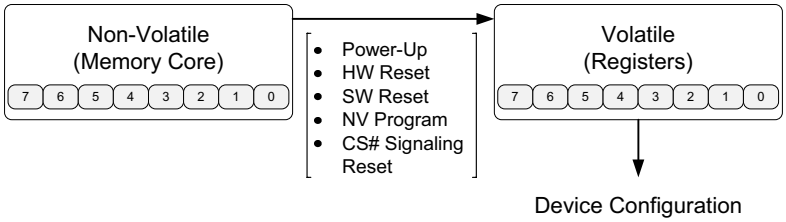


Figure 49. Data Movement Within Register Components



Note that Semper Flash with HyperBus interface uses a separate register set for each interface type: HyperBus and Legacy (x1) SPI. Both register sets must be configured for their respective interface operation.

5.1 Register Naming Convention

Table 75. Register Bit Description Convention

Bit Number	Name	Function	Read/Write	Factory Default (binary)	Description
REGNAME#T[x] T = N, V, O Descending Order	–	–	Possible Options: N/A - Not Applicable R - Readable Only R/W - Readable and Writable R/1 - Readable and One Time Programmable	Possible Options: 0 1	Format: Description of the Configuration bit 0 = Option ‘0’ selection of the Bit 1 = Option '1' selection of the Bit Dependency: Is this Bit part of a function which requires multiple bits for implementation?

5.2 HyperBus Registers

5.2.1 Configuration Register 1 (CFR1x)

Configuration Register 1 controls device functionality.

Table 76. Configuration Register 1

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
CFR1N[15] CFR1V[15]	DPDPOR	Deep Power Down power saving mode entry selection upon Power-On-Reset	N => R/W V => R/W	1	Description: The DPDPOR bit selects if the device will be in either the Deep Power Down (DPD) mode or the Standby mode after the completion of Power-On-Reset (POR). If enabled, DPDPOR configures the device to start in DPD mode to reduce current consumption until the device is needed. If the device is in DPD, a pulse on CS# or a Hardware reset will return the device to Standby mode. Selection Options: 0 = Power Down Power mode is entered upon the completion of Power-On-Reset 1 = Standby mode is entered upon the completion of Power-On-Reset Deep Dependency: N/A
CFR1N[14:12] CFR1V[14:12]	IOIMPD[2:0]	I/O Driver Output Impedance Selection	N => R/W V => R/W	101	Description: The IOIMPD[2:0] bits select the IO driver output impedance (drive strength). The output impedance configuration bits adjust the drive strength during normal device operation to meet system signal integrity requirements. Selection Options: 000 = 45Ω 001 = 120Ω 010 = 90Ω 011 = 60Ω 100 = 45Ω 101 = 30Ω (Factory Default) 110 = 20Ω 111 = 15Ω Dependency: N/A
CFR1N[11] CFR1V[11]	TLCFRP	Temporary Locking selection of Configuration Registers	N => R/W V => R/W	1	Description: The TLCFRP bit temporarily protects the Configuration Registers. Upon power-up or a hardware reset, TLCFRP is set to its default state. When selected, it protects the Configuration registers from programming. **Note** It is recommended to program and verify other Configuration Register bits before setting up TLCFRP = 0. Selection Options: 0 = Configuration Registers are protected 1 = Configuration Registers are not protected Dependency: N/A
CFR1N[10] CFR1V[10]	TLSSRP	Temporary Locking selection of Secure Silicon Region	N => R/W V => R/W	1	Description: The TLSSRP bit temporarily protects the SSR. Upon power-up or a hardware reset, TLSSRP is set to its default state. When selected, it protects the SSR from programming. Selection Options: 0 = Secure Silicon Region is protected 1 = Secure Silicon Region is not protected Dependency: N/A

Table 76. Configuration Register 1 (Continued)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
CFR1N[9:8] CFR1V[9:8]	TB4KBS[1:0]	Top or Bottom Address Range Selection for 4 KB Sector Block	N => R/W V => R/W	10	Description: The TB4KBP bits define the logical address location of the 4 KB sector block. The 4 KB sector block replaces the fitting portion of the highest or lowest address sector. Selection Options: 00 = 4 KB Sector Block and Read Password Sector is mapped into the bottom of the memory address space 01 = 4 KB Sector Block and Read Password Sector is mapped into the top of the memory address space 10 = Uniform Sector architecture and Read Password Sector is mapped into the bottom of the memory address space 11 = Uniform Sector architecture and Read Password Sector is mapped into the top of the memory address space Dependency: N/A
CFR1N[7:4] CFR1V[7:4]	MEMLAT[3:0]	Memory Array Read Latency selection - Dummy cycles required for initial data access	N => R/W V => R/W	1011	Description: The MEMLAT[3:0] bits control the read latency (dummy cycles) delay in all variable latency memory array and Nonvolatile Register read transactions. MEMLAT selection allows the user to adjust the read latency during normal operation based on different operating frequencies. The device will support 20 cycles read latency upon erasing the configuration register - 1. Selection Options: 0000 = 5 Latency Cycles Selection based on transaction opcodes ….. 1100 = 20 Latency Cycles Selection based on transaction opcodes Dependency: N/A
CFR5V[3]	RESVRD	Reserved for Future Use	N => R/W V => R/W	1	These bits are Reserved for future use. This bit must always be written/loaded to its default state.
CFR1N[2] CFR1V[2]	DTSTSL	Data Strobe Stall during Device Error Selection (DS)	N => R/W V => R/W	1	Description: The DTSTSL bit selects whether Data Strobe (DS) stalls on device errors such as 2-bit ECC detection. Selection Options: 0 = DS stalls on device errors 1 = DS does not stall on device errors Dependency: N/A
CFR1N[1:0] CFR1V[1:0]	RBSTWL[1:0]	Read Burst Wrap Length selection	N => R/W V => R/W	11	Description: The RBSTWL[1:0] bits select the read burst wrap length and alignment during normal operation. It selects the fixed length/aligned group of 16, 32, or 64 bytes. Selection Options: 00 = Reserved 01 = 64 Bytes Wrap length 10 = 16 Bytes Wrap length 11 = 32 Bytes Wrap length Dependency: N/A

Table 77. HyperBus Maximum Operating Frequency for MEMLAT[3:0] Latency Code Options

Latency Code	Latency Clocks Initial Latency Single/Double	Single Initial Latency Maximum Frequency (MHz)	Double Initial Latency Maximum Frequency (MHz)
0000	5/10	57	107
0001	6/12	71	121
0010	7/14	85	135
0011	8/16	100	150
0100	9/18	107	164
0101	10/20	114	178
0110	11/22	128	192
0111	12/24	142	200
1000	13/26	157	200
1001	14/28	171	200
1010	15/30	185	200
1011	16/32	200	200
1100	20/40	200	200
1101	Reserved	–	–
1110	Reserved	–	–
1111	Reserved	–	–

5.2.2 Configuration Register 2 (CFR2x)

Configuration Register 2 controls device functionality.

Table 78. Configuration Register 2

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
CFR2V[15:8]	RESVRD	Reserved for Future Use	N => R/W V => R/W	11111111	These bits are Reserved for future use. This bit must always be written/loaded to its default state.
CFR2N[7] CFR2V[7]	PGLTNS	Memory Read Page Boundary Latency selection	N => R/W V => R/W	1	Description: The PGLTNS bit selects whether single or double initial latency is encountered during read instructions. Double initial latency ensures there is no inter-page boundary latency. Selection Options: 0 = Double initial latency - zero latency at page boundaries 1 = Single initial latency - latency at page boundaries Dependency: N/A
CFR2N[6] CFR2V[6]	ATP12S	Address Trap Register 1-bit or 1-bit/2-bit ECC Error selection	N => R/W V => R/W	1	Description: The ATP12S bit selects between 1-bit ECC error detection/correction or both 1-bit ECC error detection/correction and 2-bit ECC error detection for Address Trap Register. Selection Options: 1 = Address Trap Register traps address of 2-bit ECC errors 0 = Address Trap Register traps address of both 1-bit and 2-bit ECC errors Dependency: N/A
CFR2N[5] CFR2V[5]	ECC12S	Error Correction Code (ECC) 1-bit or 1-bit/2-bit error correction selection	N => R/W V => R/W	0	Description: The ECC12S bit selects between 1-bit ECC error detection/correction or both 1-bit ECC error detection and correction and 2-bit ECC error detection. Selection Options: 0 = 1-bit ECC Error Detection/Correction and 2-bit ECC error detection 1 = 1-bit ECC Error Detection/Correction Dependency: N/A

Table 78. Configuration Register 2 (Continued)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
CFR2N[4] CFR2V[4]	HYBSEL	Hybrid Burst Selection - CA[45] must select Wrapped Burst type for this option	N => R/W V => R/W	1	Description: The HYBSEL bit selects hybrid burst which provides one initial wrapped burst followed by a linear burst. Selection Options: 0 = Hybrid burst is enabled (Wrapped burst followed by linear burst) 1 = Hybrid burst is disabled Dependency: N/A
CFR2N[3] CFR2V[3]	RESVRD	Reserved for Future Use	N => R/W V => R/W	1	These bits are Reserved for future use. This bit must always be written/ loaded to its default state.
CFR2N[2] CFR2V[2]	SP4KBS	Split 4 KB Sectors selection between top and bottom address space	N => R/W V => R/W	1	Description: The SP4KBS bit selects whether the 4 KB sectors are grouped together or evenly split between high and low address ranges. Selection Options: 0 = 4 KB Sectors are grouped together 1 = 4 KB Sectors are split between High and Low Addresses Dependency: TB4KBS[1:0] (CFR1x[9:8]) Usage Link:
CFR2V[1]	RESVRD	Reserved for Future Use	N => R/W V => R/W	1	These bits are Reserved for future use. This bit must always be written/ loaded to its default state.
CFR2N[0] CFR2V[0]	CK#SEL	Master Clock Selection - Single Ended or Differential	N => R/W V => R/W	1	Description: The CK#SEL bit selects whether master clock is single ended (CK) or differential (CK, CK#) Selection Options: 1 = Master Clock is single ended (CK) 0 = Master Clock is differential (CK, CK#) Dependency: N/A

5.2.3 Status Registers

Status Register provides device status on operation.

Table 79. Status Register[28]

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
STRV[15:9]	RESVRD	Reserved for Future Use	V => R	1111111	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
STRV[8]	DICRCS	Memory Array Data Integrity Check CRC Suspend Status Flag	V => R	0	Description: The DICRCS bit is used to determine when the device is in Memory Array Data Integrity Check CRC suspend mode. Selection Options: 0 = Memory Array Data Integrity Check CRC is not in suspend mode 1 = Memory Array Data Integrity Check CRC is in suspend mode Dependency: N/A

Note
28. STRV value during POR, Hardware Reset, DPD Exit, and CS# Signaling Reset is not valid.

Table 79. Status Register[28] (Continued)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
STRV[7]	RDYBSY	Device Ready/Busy Status Flag	V => R	1	Description: The RDYBSY bit indicates whether the device is performing an embedded operation or is in standby mode ready to receive new operation transactions. **Note** If PRGERR or ERSERR is set, the RDYBSY bit will set to a '1'. Selection Options: 0 = Device is busy and unable to receive new operation transactions 1 = Device is in standby mode ready to receive new operation transactions (Host needs to check PRGERR and ERSERR before providing a new transaction) Dependency: N/A
STRV[6]	ERASES	Erase operation Suspend Status Flag	V => R	0	Description: The ERASES bit is used to indicate if the Erase operation is suspended. Selection Options: 0 = Erase operation is not in suspend mode 1 = Erase operation is in suspend mode Dependency: N/A
STRV[5]	ERSERR	Erasing Error Status Flag	V => R	0	Description: The ERSERR bit indicates erase operation success or failure. When the ERSERR bit is set to a '1', it indicates that there was an error in the last erasing operation. ERSERR bit is also set when a erase operation is attempted within a protected memory sector. When ERSERR is set, it can only be cleared with the Clear Status Register (CLSR) transaction or a hardware/software reset. **Note** The device will only go to standby mode once the ERSERR flag is cleared. Selection Options: 0 = Last erase operation was successful 1 = Last erase operation was unsuccessful Dependency: N/A
STRV[4]	PRGERR	Programming Error Status Flag	V => R	0	Description: The PRGERR bit indicates program operation success or failure. When the PRGERR bit is set to a '1', it indicates that there was an error in the last programming operation. PRGERR bit is also set when a program operation is attempted within a protected memory region. When PRGERR is set, it can only be cleared with the Clear Status Register (CLSR) transaction or a hardware/software reset. **Note** The device will only go to standby mode once the PRGERR flag is cleared. Selection Options: 0 = Last programming operation was successful 1 = Last programming operation was unsuccessful Dependency: N/A

Note
28. STRV value during POR, Hardware Reset, DPD Exit, and CS# Signaling Reset is not valid.

Table 79. Status Register[28] (Continued)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
STRV[3]	WRBFAB	Write Buffer Abort Status Flag	V => R	0	Description: The WRBFAB bit indicates a programming abort condition. When any condition causes an abort of write buffer instruction, the abort will happen immediately after the offending condition, and will indicate a Program Fail in the Status Register (PRGERR = 1) along with a Write Buffer Abort (WRBFAB = 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 WRBFAB also indicates that the memory array CRC calculation operation was aborted. The Ending Address (EA) should be at least one address higher than the Starting Address (SA). If EA < SA + 1 the Check-value Calculation will abort and the device will return to the Ready state. WRBFAB will be set (1) to indicate the aborted condition. If EA < SA + 1 the CRCR will hold indeterminate data. Selection Options: 0 = Program did not abort during write to buffer instruction or CRC end address load instruction did not abort 1 = Program aborted during write to buffer instruction or CRC end address load instruction aborted Dependency: N/A
STRV[2]	PROGMS	Program operation Suspend Status Flag	V => R	0	Description: The PROGMS bit is used to indicate if the Program operation is suspended. Selection Options: 0 = Program operation is not in suspend mode 1 = Program operation is in suspend mode Dependency: N/A
STRV[1]	SPROTE	Sector Protection (Lock) Error Flag	V => R	0	Description: The SPROTE bit indicates that a program or erase operation failed because the target region (memory array, registers or SSR) was protected (locked). SPROTE reflects the status of the most recent program or erase operation. Selection Options: 0 = Operation did not generate a Protection error 1 = Operation generated a Protection error Dependency: N/A
STRV[0]	SESTAT	Sector Erase Success/Failure Status Flag	V => R	0	Description: The SESTAT bit indicates whether the erase operation on the sector completed successfully. Evaluate Erase Status transaction (EVERS_4_0) must be executed prior to reading SESTAT bit which specifies the sector address. Selection Options: 0 = Addressed sector was not erased successfully 1 = Addressed sector was erased successfully Dependency: N/A

Note
28. STRV value during POR, Hardware Reset, DPD Exit, and CS# Signaling Reset is not valid.

5.2.4 ECC Status Register (ECSV)

The ECC Status (ECSV) contains the ECC status of any error correction action performed on the unit data.

Table 80. ECC Status Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
ECSV[15:4]	RESRVD	Reserved for Future Use	V => R	000000000000	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
ECSV[3]	ECC2BT	ECC Error 2-bit Error Detection Flag	V => R	0	Description: The ECC2BT bit indicates that a 2-bit ECC Error was detected in the data unit (16 bytes). Selection Options: 0 = No 2-Bit ECC Error was detected in the data unit (16 bytes) 1 = 2-bit ECC Error was detected in the data unit (16 bytes) Dependency: N/A
ECSV[2]	RESVRD	Reserved for Future Use	V => R	0	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
ECSV[1]	ECC1BT	ECC Error 1-bit Error Detection and Correction Flag	V => R	0	Description: The ECC1BT bit indicates that a 1-bit ECC Error was detected and corrected in the data unit (16 bytes). Selection Options: 0 = No 1-Bit ECC Error was detected in the data unit (16 bytes) 1 = 1-bit ECC Error was detected in the data unit (16 bytes) Dependency: N/A
ECSV[0]	ECCDBL	ECC Disable Flag	V => R	0	Description: The ECCDBL bit indicates whether the ECC is enabled or disabled in the addressed data unit (16 bytes). Selection Options: 0 = ECC is enabled in the addressed data unit (16 bytes) 1 = ECC is disabled in the addressed data unit (16 bytes) Dependency: N/A

5.2.5 ECC Address Trap Register (EATV)

The ECC Address Trap Register (EATV) stores the address of the ECC unit data where either a 1-Bit/2-Bit error or only a 1-Bit error occurred during a read operation. It stores the ECC unit address of the first ECC error.

Table 81. ECC Address Trap Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EATV[31:0]	ECCATP[31:0]	ECC 1-bit and 2-bit Error Address Trap Register	V => R	0x00000000	Description: The Address Trap Register (ECCATP[31:0]) stores the ECC unit data address where a 1-Bit/2-Bit error occurred during a read operation. ECCATP[31:0] stores the ECC unit address of the first ECC error captured during a memory read operation since the last Clear ECC Status Register transaction (CLECC). **Note** ECCATP[31:0] is only updated during Read Instruction. **Note** Clear ECC Status Register transaction (CLECC), POR or Hardware/ Software reset clears the EATV[31:0] to 0x00000000. Selection Options: ECC Error Data Unit Address Dependency: N/A+F6:F20

5.2.6 ECC Error Detection Count Register (ECTV)

The ECC Error Detection Count Register (ECTV) stores the number of either 1-Bit/2-Bit or only 1-Bit ECC errors have occurred during read operations since the last POR or hardware/software reset.

Table 82. ECC Count Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (hex)	Description
ECTV[15:0]	ECCCNT[15:0]	ECC 1-bit and 2-bit Error Count Register	V => R	0x0000	Description: The ECCCNT[15:0] stores the number of 1-bit/2-bit ECC errors occurred during read operations since the last POR or hardware/software reset. **Note** ECCCNT[15:0] is only updated during Read Instruction. **Note** Only one ECC error is counted for each data unit. If multiple read transactions access the same unit data containing an ECC error, the ECCCNT[15:0] will increment each time the unit data is read. **Note** Once the count reaches 0xFFFF, theECCCNT[15:0] will stop incrementing **Note** POR or Hardware/Software reset clears the ECCNT[15:0] to 0x0000. Selection Options: ECC Error Count Dependency: N/A

5.2.7 ASP Password Register (PWDO)

The ASP Password Register (PWDO) is used to permanently define a password.

Table 83. Password Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (hex)	Description
PWDO[63:0]	PASWRD[63:0]	Password Register	N => R/1	0xFFFFFFFFFFFFFFFF	Description: The PASWRD[63:0] permanently stores a password used in password protected modes of operation. When the Password Protection mode is enabled, this register will output all 1s data upon read password request. Selection Options: Password Dependency: N/A

5.2.8 *Advanced Sector Protection Register (ASPO)*

The ASP Register (ASPO) configures the behavior of Advanced Sector Protection scheme.

Table 84. Advanced Sector Protection Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
ASPO[15:6]	RESVRD	Reserved for Future Use	N => R/1	1111111111	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
ASPO[5]	ASPRDP	Read Password Based Protection Selection	N => R/1	1	Description: The ASPRDP bit selects the Read Password Mode Protection mode. Read Password Protection mode works in conjunction with Password Protection mode to protect all sectors from Read/Erase/Program. Based on TB4KBS[1:0] configuration bit (CFR1x[9:8]), either the top or bottom sector is available for reading. Selection Options: 0 = Read Password Protection Mode is enabled 1 = Read Password Protection Mode is disabled Dependency: TB4KBS[1:0] (CFR1x[9:8])
ASPO[4]	RESVRD	Reserved for Future Use	N => R/1	1	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
ASPO[3]	RESVRD	Reserved for Future Use	N => R/1	1	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
ASPO[2]	ASPPWD	Password Based Protection Selection	N => R/1	1	Description: The ASPPWD bit selects the Password Protection Mode. Password Protection mode protects all PPB bits, all registers and all memory from erase/program till the correct password is entered. The ASPPWD can also be used in combination with the ASPRDP to protect sectors from being read as well till the correct password is provided – except for top or bottom sector which is available for reading based on TB4KBS[1:0] configuration bit (CFR1x[9:8]). **Note** When ASPPWD is selected, ASPO[15:0] are protected against program operations. Selection Options: 0 = Password Protection Mode is enabled 1 = Password Protection Mode is disabled Dependency: ASPPER (ASPO[1])
ASPO[1]	ASPPER	Persistent Protection Selection (Register Protection Selection)	N => R/1	1	Description: The ASPPER bit selects the Persistent Protection Mode. The Persistent Protection mode (ASPPER) protects the ASPO[15:0] Register from erase or program. Selection Options: 0 = Persistent Protection Mode is enabled 1 = Persistent Protection Mode is disabled Dependency: ASPPWD (ASPO[2])
ASPO[0]	RESVRD	Reserved for Future Use	N => R/1	1	This bit is Reserved for future use. This bit must always be written/loaded to its default state.

5.2.9 ASP PPB Lock Register (PPLV)

The PPBLCK bit in the ASP PPB Lock Register (PPLV) is used to protect the PPB bits.

Table 85. ASP PPB Lock Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
PPLV[15:1]	RESVRD	Reserved for Future Use	V => R	11111111111111	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
PPLV[0]	PPBLCK	PPB Temporary Protection Selection	V => R	1	Description: The PPBLCK bit is used to temporarily protect all the PPB bits. Selection Options: 0 = PPB bits are protected against erase or program till the next POR or hardware reset 1 = PPB Bits can be erased or programmed Dependency: N/A

5.2.10 ASP PPB Access Register (PPAV)

The ASP PPB Access Register (PPAV) is used to provide the state of each sector's PPB protection bit.

Table 86. ASP PPB Access Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
PPAV[15:0]	PPBACS[15:0]	Sector Based PPB Protection Status	N => R/W	1111111111111111	Description: The PPBACS[7:0] bits are used to provide the state of the individual sector's PPB bit. Selection Options: FFFF = PPB for the sector addressed by the Read PPB transaction is 1, not protecting that sector from program or erase operations 0000 = PPB for the sector addressed by the Read PPB transaction is 0, protecting that sector from program or erase operations Dependency: N/A

5.2.11 ASP Dynamic Block Access Register (DYAV)

The ASP DYB Access Register (DYAV) is used to provide the state of each sector's DYB protection bit.

Table 87. ASP DYB Access Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
DYAV[15:0]	DYBACS[15:0]	Sector Based DYB Protection Status	V => R/W	1111111111111111	Description: The DYBACS[7:0] bits are used to provide the state of the individual sector's DYB bit. Selection Options: FFFF = DYB for the sector addressed by the Read DYB transaction is 1, not protecting that sector from program or erase operations 0000 = DYB for the sector addressed by the Read DYB transaction is 0, protecting that sector from program or erase operations Dependency: N/A

5.2.12 AutoBoot Register (ATBN)

The AutoBoot Register (ATBN) provides a means to automatically read boot code as part of the power-on reset, or hardware reset process.

Table 88. AutoBoot Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
ATBN[31:9]	STADR[22:0]	Starting Address Selection where AutoBoot will start reading data from	N => R/W	00000000000000000000000	Description: The STADR[22:0] bits set the starting address from which the device will output the read data. Selection Options: Address Bits Dependency: N/A
ATBN[8:1]	STDLY[7:0]	AutoBoot Read Starting Delay Selection	N => R/W	00000000	The STDLY[7:0] bits specify the initial delay (clock cycles) needed by the host before it can accept data. **Note** STDLY[7:0]=0x00 is valid for SPI up to 50MHz. STDLY[8:1]=0x01 or higher is valid for SPI/DPI/QPI up to 166MHz. STDLY[7:0] = 0x05 or higher is valid for OPI/HyperFlash up to 200MHz. Selection Options: Address Bits Dependency: N/A
ATBN[0]	ATBTEN	AutoBoot Feature Selection	N => R/W	0	Description: The ATBTEN bit enables or disables the AutoBoot feature. Selection Options: 0 = AutoBoot feature disabled 1 = AutoBoot feature enabled Dependency: N/A

5.2.13 Sector Erase Count Register (SECV)

The Sector Erase Count Register (SECV) contains the number of times the addressed sector has been erased.

Table 89. Sector Erase Count Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
SECV[31:24]	RESVRD	Reserved for Future Use	V => R	11111111	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
SECV[23]	SECCPT	Sector Erase Count Corruption Status Flag	V => R	0	Description: The SECCPT bit is used to determine if the reported sector erase count is corrupted and was reset. **Note** If SECCPT is set due to count corruption, it will reset to 0 on the next successful erase operation on the selected sector. Selection Options: 0 = Sector Erase Count is not corrupted and is valid 1 = Sector Erase Count is corrupted and is not valid Dependency: N/A
SECV[22:0]	SECVAL[22:0]	Sector Erase Count Value	V => R	0	Description: The SECVAL[22:0] bits store the number of times a sector has been erased. Selection Options: Value Dependency: N/A

5.2.14 Memory Array Data Integrity Check CRC Register (DCRV)

The Memory Array Data Integrity Check CRC Register (DCRV) stores the results of the CRC calculation on the data contained between the specified starting and ending addresses.

Table 90. Memory Array Data Integrity Check CRC Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (hex)	Description
DCRV[31:0]	DTCRCV[31:0]	Memory Array Data Integrity Check CRC Checksum Value	V => R	0xFFFFFFFF	Description: The DTCRCV[31:0 bits store the checksum value of the CRC process on the memory array data contained within the starting address and the ending address. Selection Options: Checksum Value Dependency: N/A

5.2.15 Interface CRC Check-value Register (ICRV)

The Interface CRC Check-value Register (ICRV) stores the results of the CRC calculation on the command and Data Content over the interface for Protection.

Table 91. Interface CRC Check-value Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (hex)	Description
ICRV[31:0]	ITCRCV[31:0]	Interface CRC Checksum Value	V => R	0xFFFFFFFF	Description: The ITCRCV[31:0] bits store the check-value of the CRC process on the memory array data contained within the starting address and the ending address. Selection Options: Checksum Value Dependency: N/A

5.2.16 EnduraFlex Architecture Selection Register (EFXx)

The EnduraFlex Architecture Selection registers (EFXx) define the long retention / high endurance regions based on a four pointer based architecture.

Table 92. EnduraFlex Architecture Selection Register (Pointer 4)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EFX4[10:2]	EPTAD4[8:0]	EnduraFlex Pointer 4 Address Selection	N => R/1	111111111	Description: The EPTAD4[8:0] bits define the 9-bit address of the beginning sector from where the long retention / high endurance region is defined. Selection Options: Pointer Address Dependency: N/A
EFX4[1]	ERGNT4	EnduraFlex Pointer 4 based Region Type Selection	N => R/1	1	Description: The ERGNT4 bit defines whether the region is long retention or high endurance. Selection Options: 0 = Long Retention Sectors 1 = High Endurance Sectors Dependency: N/A
EFX4[0]	EPTEB4	EnduraFlex Pointer 4 Enable# Selection	N => R/1	1	Description: The EPTEN4 bit defines whether the wear leveling pointer is enabled/disabled. Selection Options: 0 = Pointer Address Enabled 1 = Pointer Address Disabled Dependency: N/A

Table 93. EnduraFlex Architecture Selection Register (Pointer 3)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EFX3[10:2]	EPTAD3[8:0]	EnduraFlex Pointer 3 Address Selection	N => R/1	111111111	Description: The EPTAD3[8:0] bits define the 9-bit address of the beginning sector from where the long retention / high endurance region is defined. Selection Options: Pointer Address Dependency: N/A
EFX3[1]	ERGNT3	EnduraFlex Pointer 3 based Region Type Selection	N => R/1	1	Description: The ERGNT3 bit defines whether the region is long retention or high endurance. Selection Options: 0 = Long Retention Sectors 1 = High Endurance Sectors Dependency: N/A
EFX3[0]	EPTEB3	EnduraFlex Pointer 3 Enable# Selection	N => R/1	1	Description: The EPTEN3 bit defines whether the wear leveling pointer is enabled/disabled. Selection Options: 0 = Pointer Address Enabled 1 = Pointer Address Disabled Dependency: N/A

Table 94. EnduraFlex Architecture Selection Register (Pointer 2)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EFX2[10:2]	EPTAD2[8:0]	EnduraFlex Pointer 2 Address Selection	N => R/1	111111111	Description: The EPTAD2[8:0] bits define the 9-bit address of the beginning sector from where the long retention / high endurance region is defined. Selection Options: Pointer Address Dependency: N/A
EFX2[1]	ERGNT2	EnduraFlex Pointer 2 based Region Type Selection	N => R/1	1	Description: The ERGNT2 bit defines whether the region is long retention or high endurance. Selection Options: 0 = Long Retention Sectors 1 = High Endurance Sectors Dependency: N/A
EFX2[0]	EPTEB2	EnduraFlex Pointer 2 Enable# Selection	N => R/1	1	Description: EPTEN2 bit defines whether the wear leveling pointer is enabled/disabled. Selection Options: 0 = Pointer Address Enabled 1 = Pointer Address Disabled Dependency: N/A

Table 95. EnduraFlex Architecture Selection Register (Pointer 1)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EFX1[10:2]	EPTAD1[8:0]	EnduraFlex Pointer 1 Address Selection	N => R/1	111111111	Description: The EPTAD1[8:0] bits define the 9-bit address of the beginning sector from where the long retention / high endurance region is defined. Selection Options: Pointer Address Dependency: N/A
EFX1[1]	ERGNT1	EnduraFlex Pointer 1 based Region Type Selection	N => R/1	1	Description: The ERGNT1 bit defines whether the region is long retention or high endurance. Selection Options: 0 = Long Retention Sectors 1 = High Endurance Sectors Dependency: N/A
EFX1[0]	EPTEB1	EnduraFlex Pointer 1 Enable# Selection	N => R/1	1	Description: The EPTEN1 bit defines whether the wear leveling pointer is enabled/disabled. Selection Options: 0 = Pointer Address Enabled 1 = Pointer Address Disabled Dependency: N/A

Table 96. EnduraFlex Architecture Selection Register (Pointer 0)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EFX0[1]	GBLSEL	All Sectors based Region type Selection	N => R/1	1	Description: The GBLSEL bit defines whether all sectors are defined as long retention region or high endurance region. **Note** If all other Pointer registers are disabled, this bit defines the behavior of the entire memory space and is hardwired to start at Sector 0. Selection Options: 0 = Long Retention Sectors 1 = High Endurance Sectors Dependency: N/A
EFX0[0]	WRLVEN	Wear Leveling Enable Selection	N => R/1	1	Description: The WRLVEN bit enables/disables the wear leveling feature. Selection Options: 0 = Wear Leveling Disabled 1 = Wear Leveling Enabled Dependency: N/A

5.2.17 INT# Pin Configuration Register (INCV)

The INT# pin Configuration Register (INCV) configures which internal event will trigger a High to Low transition on the INT# output pin.

Table 97. Interrupt Configuration Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
INCV[15]	INTBEN	INT# pin Enable Selection	V => R/W	1	Description: The INT# pin is an open-drain output used to indicate to the host system that an event has occurred within the memory device. The INTBEN bit enables or disables the functionality controlling INT# pin. Selection Options: 0 = INT# pin functionality is enabled 1 = INT# pin functionality is disabled Dependency: N/A
INCV[14]	RESRVD	Reserved for Future Use	V => R/W	11	These bits are Reserved for future use. This bit must always be written/ loaded to its default state.
INCV[13:5]	RESRVD	Reserved for Future Use	V => R/W	111111111	These bits are Reserved for future use. This bit must always be written/ loaded to its default state.
INCV[4]	REYBSY	Ready/Busy Transition Selection	V => R/W	1	Description: The REYBSY bit enables or disables whether device ready/ busy state will transition INT#. Selection Options: 0 = A Busy to Ready transition will cause a High to Low transition on the INT# output 1 = Ready/Busy transitions will not transition the INT# output Dependency: N/A
INCV[3:2]	RESRVD	Reserved for Future Use	V => R/W	11	These bits are Reserved for future use. This bit must always be written/ loaded to its default state.
INCV[1]	ECC2BT	ECC 2-bit Error Detection Selection	V => R/W	1	Description: The ECC2BT bit enables or disables whether a 2-bit ECC detection error will transition INT#. Selection Options: 0 = 2-bit ECC detection will cause a High to Low transition the INT# output 1 = 2-bit ECC detection will not transition the INT# output Dependency: N/A
INCV[0]	ECC1BT	ECC 1-bit Error Detection and Correction Selection	V => R/W	1	Description: The ECC1BT bit enables or disables whether a 1-bit ECC detection and correction error will transition INT#. Selection Options: 0 = 1-bit ECC detection and correction will cause a High to Low transition the INT# output 1 = 1-bit ECC detection and correction will not transition the INT# output Dependency: N/A

5.2.18 INT# Pin Status Register (INSV)

The INT# Pin Status Register (INSV) indicates which internal event(s) has occurred since the last time the ISR was cleared.

Table 98. Interrupt Status Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
INSV[15:5]	RESRVD	Reserved for Future Use	V => R/W	11111111111	These bits are Reserved for future use. This bit must always be written/ loaded to its default state.
INSV[4]	REYBSY	Ready/Busy Transition	V => R/W	1	Description: The REYBSY bit indicates whether the device's ready/busy status has caused a transition on INT#. Selection Options: 0 = A Busy to Ready transition has occurred 1 = A Busy to Ready transition has not occurred Dependency: N/A
INSV[3:2]	RESRVD	Reserved for Future Use	V => R/W	11	These bits are Reserved for future use. This bit must always be written/ loaded to its default state.
INSV[1]	ECC2BT	ECC 2-bit Error Detection	V => R/W	1	Description: The ECC2BT bit indicates whether a 2-bit ECC detection error has caused a transition on INT#. Selection Options: 0 = 2-bit error detection has occurred 1 = 2-bit error detection has not occurred Dependency: N/A
INSV[0]	ECC1BT	ECC 1-bit Error Detection and Correction	V => R/W	1	Description: The ECC1BT bit indicates whether a 1-bit ECC correction error has caused a transition on INT#. Selection Options: 0 = 1-bit error correction has occurred 1 = 1-bit error correction has not occurred Dependency: N/A

5.3 Legacy (x1) SPI Registers

5.3.1 Status Register 1 (STR1x)

Status Register 1 contains both status and control bits. The functionality of supported Status Register 1 type is described in Table 99.

Table 99. Status Register 1[29]

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
STR1N[7] STR1V[7]	RESRVD	Reserved for Future Use	N->R V->R	0	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
STR1V[6]	PRGERR	Programming Error Status Flag	V -> R	0	Description: The PRGERR bit indicates program operation success or failure. When the PRGERR bit is set to a '1', it indicates that there was an error in the last programming operation. PRGERR bit is also set when a program operation is attempted within a protected memory region. When PRGERR is set, it can only be cleared with the Clear Status Register (CLSTR_0_0) transaction or a hardware/ software reset. **Note** The device will only go to standby mode once the PRGERR flag is cleared. Selection Options: 0 = Last programming operation was successful 1 = Last programming operation was unsuccessful Dependency: N/A
STR1V[5]	ERSERR	Erasing Error Status Flag	V -> R	0	Description: The ERSERR bit indicates erase operation success or failure. When the ERSERR bit is set to a '1', it indicates that there was an error in the last erasing operation. ERSERR bit is also set when a erase operation is attempted within a protected memory sector. When ERSERR is set, it can only be cleared with the Clear Status Register (CLSTR_0_0) transaction or a hardware/software reset. **Note** The device will only go to standby mode once the ERSERR flag is cleared. Selection Options: 0 = Last erase operation was successful 1 = Last erase operation was unsuccessful Dependency: N/A
STR1N[4:2] STR1V[4:2]	RESRVD	Reserved for Future Use	N->R V -> R	000	These bits are Reserved for future use. They must always be written/loaded to their default state.
STR1V[1]	WRPGEN	Write/Program Enable Status Flag	V -> R	0	Description: The WRPGEN bit must be set to '1' to enable all program, erase or register write operations - it provides protection against inadvertent changes to memory or register values. The Write Enable (WRENB_0_0) and Write Enable Volatile (WRENV_0_0) transactions set the WRPGEN bit to '1' to allow program, erase or write transactions to execute. The Write Disable (WRDIS_0_0) transaction resets WRPGEN to a '0' to prevent all program, erase, and write transactions from execution. The WRPGEN bit is cleared to '0' at the end of any successful program, erase or register write operation. After a power down / power up sequence or a hardware/software reset, the Deep Power Down WRPGEN bit is cleared to '0'. Selection Options: 0 = Program, erase or register write is disabled 1 = Program, erase or register write is enabled Dependency: N/A
STR1V[0]	RDYBSY	Device Ready/Busy Status Flag	V -> R	0	Description: The RDYBSY bit indicates whether the device is performing an embedded operation or is in standby mode ready to receive new transactions. **Note** The PRGERR and ERSERR status bits are updated while RDYBSY is set. If PRGERR or ERSERR are set, the RDYBSY bit will remain set indicating the device is busy and unable to receive new transactions. A Clear Status Register (CLSTR_0_0) transaction must be executed to return the device to standby mode. Selection Options: 0 = Device is in standby mode ready to receive new operation transactions 1 = Device is busy and unable to receive new operation transactions Dependency: N/A

Note
29. STR1x value during POR, Hardware Reset, DPD Exit, and CS# Signaling Reset is not valid.

5.3.2 Status Register 2 (STR2x)

Status Register 2 provides device status on operations. The functionality of supported Status Register 2 type is described in Table 100.

Table 100. Status Register 2[30]

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
STR2V[7:5]	RESRVD	Reserved for Future Use	V -> R	0	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
STR2V[4]	DICRCS	Memory Array Data Integrity Check CRC Suspend Status Flag	V -> R	0	Description: The DICRCS bit is used to determine when the device is in Memory Array Data Integrity Check CRC suspend mode. Selection Options: 0 = Memory Array Data Integrity Check CRC is not in suspend mode 1 = Memory Array Data Integrity Check CRC is in suspend mode Dependency: N/A
STR2V[3]	DICRCA	Memory Array Data Integrity Check CRC Abort Status Flag	V -> R	0	Description: The DICRCA bit indicates that the Memory Array Data Integrity Check CRC calculation operation was aborted. The abort condition is based on ending address (ENDADD) and starting address (STRADD) relationship. If ENDADD < STRADD + 3, then DICRCA will be set and the device will return to the Standby state. DICRCA flag gets cleared at the next Data Integrity CRC calculation operation when ENDADD ≥ STRADD + 3. Selection Options: 0 = Memory Array Data Integrity Check CRC calculation Is not aborted 1 = Memory Array Data Integrity Check CRC calculation is aborted Dependency: N/A
STR2V[2]	SESTAT	Sector Erase Success/Failure Status Flag	V -> R	0	Description: The SESTAT bit indicates whether the erase operation on the sector completed successfully. Evaluate Erase Status transaction (EVERS_4_0) must be executed prior to reading SESTAT bit which specifies the sector address. Selection Options: 1 = Addressed sector (EVERS_4_0) was erased successfully 0 = Addressed sector (EVERS_4_0) was not erased successfully Dependency: N/A
STR2V[1]	ERASES	Erase operation Suspend Status Flag	V -> R	0	Description: The ERASES bit is used to indicate if the Erase operation is suspended. Selection Options: 0 = Erase operation is not in suspend mode 1 = Erase operation is in suspend mode Dependency: N/A
STR2V[0]	PROGMS	Program operation Suspend Status Flag	V -> R	0	Description: The PROGMS bit is used to indicate if the Program operation is suspended. Selection Options: 0 = Program operation is not in suspend mode 1 = Program operation is in suspend mode Dependency: N/A

Note
30. STR2x value during POR, Hardware Reset, Software Reset, DPD Exit, and CS# Signaling Reset is not valid. STR2x bits are valid only when SRT1V[0] / RDYBSY = 0.

5.3.3 Configuration Register 1 (CFR1x)

Configuration Register 1 controls interface and data protection functions.

Table 101. Configuration Register 1

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
CFR1N[7] CFR1V[7]	RESRVD	Reserved for Future Use	N -> R/W V -> R/W	0	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
CFR1N[6] CFR1V[6]	SP4KBS	Split 4 KB Sectors selection between top and bottom address space	If PLPROT = 0 N -> R/W V -> R If PLPROT = 1 N -> R V -> R	0	Description: The SP4KBS bit selects whether the 4 KB sectors are grouped together or evenly split between High and Low address ranges. Selection Options: 0 = 4 KB Sectors are grouped together 1 = 4 KB Sectors are split between High and Low Addresses Dependency: TB4KBS(CFR1N[2])
CFR1N[5] CFR1V[5]	TBPROT	Top or Bottom Address Selection for Read Password	If PLPROT = 0 N -> R/W V -> R If PLPROT = 1 N -> R V -> R	0	Description: TBPROT selects a memory address range (lowest or highest) to remain readable during Read Password Protection mode even before a successful Password entry is completed. Selection Options: 0 = Top half of the address range 1 = Bottom half of the address range Dependency: N/A
CFR1N[4] CFR1V[4]	PLPROT	Permanent Locking selection of 4 KB Sector Architecture	N -> R/1 V -> R	0	Description: The PLPROT bit permanently protects 4 KB Sector location. **Note** PLPROT protects SP4KBS, TBPROT, and TB4KBS bits from program and erase and it is recommended to configure these bits before configuring the PLPROT bit. Selection Options: 0 = 4 KB Sector Location is not protected 1 = 4 KB Sector Location is protected Dependency: N/A
CFR1N[3] CFR1V[3]	RESRVD	Reserved for Future Use	N -> R/W V -> R/W	0	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
CFR1N[2] CFR1V[2]	TB4KBS	Top or Bottom Address Range selection for 4 KB Sector Block	If PLPROT = 0 N -> R/W V -> R If PLPROT = 1 N -> R V -> R	0	Description: The TB4KBP bit defines the logical address location of the 4 KB sector block. The 4 KB sector block replaces the fitting portion of the highest or lowest address sector. Selection Options: 0 = 4 KB Sector Block is in the bottom of the memory address space 1 = 4 KB Sector Block is in the top of the memory address space Dependency: SP4KBS (CFR1x[6])
CFR1N[1] CFR1V[1]	RESRVD	Reserved for Future Use	N -> R/W V -> R/W	0	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
CFR1N[0] CFR1V[0]	TLPROT	Temporary Locking selection of Legacy Block Protection and Sector Architecture	N -> R V -> R/W	0	Description: The TLPROT bit temporarily protects sector architecture from any changes. **Note** TLPROT protects SP4KBS, TBPROT, and TB4KBS bits from program and erase. Selection Options: 0 = 4 KB Sector Location are not protected 1 = 4 KB Sector Location are temporarily protected Dependency: N/A

Table 102. 4 KB Parameter Sector Location Selection

SP4KBS	TB4KBS	4 KB Location
0	0	4 KB physical sectors at bottom (Low address)
0	1	4 KB physical sectors at top, (High address)
1	X	4 KB Parameter sectors are split between top (High Address) and bottom (Low Address)

Table 103. PLPROT and TLPROT Protection

PLPROT	TLPROT	Array Protection and 4K Sector
0	0	Unprotected (Unlocked)
1	X	TBPROT, SP4KBS, TB4KBS - Permanently Protected (Locked)
0	1	TBPROT, SP4KBS, TB4KBS - Protected (Locked) till next Power-down

5.3.4 Configuration Register 2 (CFR2x)

Configuration Register 2 controls the memory read latency selection.

Table 104. Configuration Register 2

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
CFR2N[7] CFR2V[7]	ADRBYT	Address Byte Length selection between 3 or 4 bytes for Instructions	N -> R/W V -> R/W	0	Description: The ADRBYT bit controls the expected address length for all instructions that require address and is selectable between 3 Bytes or 4 Bytes. Selection Options: 0 = Instructions will use 3 Bytes for address 1 = Instructions will use 4 Bytes for address Dependency: N/A
CFR2N[6:4] CFR2V[6:4]	RESRVD	Reserved for Future Use	N -> R/W V -> R/W	0000	These bits are Reserved for future use. This bit must always be written/loaded to its default state.
CFR2N[3:0] CFR2V[3:0]	MEMLAT[3:0]	Memory Array Read Latency selection - Dummy cycles required for initial data access	N -> R/W V -> R/W	1000	Description: The MEMLAT[3:0] bits control the read latency (dummy cycles) delay in all variable latency memory array and Nonvolatile Register read transactions. MEMLAT selection allows the user to adjust the read latency during normal operation based on different operating frequencies. Selection Options: 0000 = 0 Latency Cycles Selection based on transaction opcodes ….. 1111 = 15 Latency Cycles Selection based on transaction opcodes Dependency: N/A

Table 105. Latency Code (Cycles) Versus Frequency[31, 32, 34]

Latency Code CR	Number of Cycles	SDR SPI Read Transactions (MHz) (1S-1S-1S)
		RDAY2_C_0 RDSSR_4_0 RDARG_4_0[33] RDECC_4_0 RDPPB_4_0
0000	0	50
0001	1	68
0010	2	81
0011	3	93
0100	4	106
0101	5	118
0110	6	131
0111	7	143
1000	8	156
1001	9	166
1010	10	166
1011	11	166
1100	12	166
1101	13	166
1110	14	166
1111	15	166

Notes

31. When using the ECC error reporting mechanisms, the read output data must be at least 2 bytes for correct ECC reporting.
32. SCK frequency > 200 MHz SDR, or > 200 MHz DDR is not supported by HS-T family of devices and SCK frequency > 166 MHz SDR, or > 166 MHz DDR is not supported by HL-T family of devices.
33. RDARG_4_0 uses these latency cycles for reading Nonvolatile registers.
34. RSFDP_3_0 always have a dummy cycle of eight and the maximum frequencies for different interfaces related to eight dummy cycles.

5.3.5 *Configuration Register 3 (CFR3x)*

Configuration Register 3 controls transaction behavior.

Table 106. Configuration Register 3

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
CFR3N[7:6] CFR3V[7:6]	VRGLAT[1:0]	Volatile Register Read Latency selection - Dummy cycles required for initial data access	N -> R/W V -> R/W	00	Description: The VRGLAT[1:0] bits control the read latency (dummy cycles) delay in all variable latency register read transactions. VRGLAT[1:0] selection allows the user to adjust the read latency during normal operation based on different operating frequencies. Selection Options: 00 = 0/0/32 Latency Cycles Selection based on transaction opcodes ….. 11 = 2/2/32 Latency Cycles Selection based on transaction opcodes Dependency: N/A
CFR3N[5] CFR3V[5]	BLKCHK	Blank Check selection during Erase operation for better endurance	N -> R/W V -> R/W	0	Description: The BLKCHK bit selects whether a sector is checked before issuing an erase operation. When this feature is enabled an erase transaction first evaluates the erase status of the sector. If the sector is found to erased, the erase operation is aborted. In other words, the erase operation is only executed if programmed bits are found in the sector. Disabling BLKCHK executes an erase operation unconditionally. Selection Options: 0 = Blank Check is disabled before executing an erase operation 1 = Blank Check evaluation is enabled before executing an erase operation Dependency: N/A
CFR3N[4] CFR3V[4]	PGMBUF	Program Buffer Size selection	N -> R/W V -> R/W	0	Description: The PGMBUF bit selects the Programming Buffer size which is used for page programming. Program buffer size affects the device programming time. **Note** If programming data exceeds the program buffer size, data gets wrapped. Selection Options: 0 = 256 Byte Write Buffer Size 1 = 512 Byte Write Buffer Size Dependency: N/A
CFR3N[3] CFR3V[3]	UNHYSA	Uniform or Hybrid Sector Architecture selection	N -> R/W V -> R	1	Description: The UNHYSA bit selects between uniform (all 256 KB sectors) or hybrid (4 KB sectors and 256 KB sectors) sector architecture. If hybrid sector architecture is selected, 4 KB sector block is made part of the main Flash array address map. The 4 KB sector block can overlay at either the highest or the lowest address range of the device. If uniform sector architecture is selected, 4 KB sector block is removed from the address map and all sectors are of uniform size. **Note** Hybrid sector architecture also enables 4 KB Sector Erase transaction (20h). Otherwise, 4 KB Sector Erase transaction, if issued, is ignored by the device. Selection Options: 0 = Hybrid Sector Architecture (combination of 4 KB sectors and 256 KB sectors) 1 = Uniform Sector Architecture (all 256 KB sectors) Dependency: ST4KBP(CFR1N[6]), TB4KBP(CFR1N[2])
CFR3N[2] CFR3V[2]	RESRVD	Reserved for Future Use	N -> R/W V -> R/W	0	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
CFR3N[1] CFR3V[1]	INTFTP	Interface type selection between HyperBus and Legacy (x1) SPI	N -> R/W V -> R/W	0	The INTFTP bit selects the interface of the device between HyperBus and legacy (x1) SPI. Selection Options: 1 = HyperBus Interface 0 = Legacy (x1) SPI Dependency: N/A
CFR3N[0] CFR3V[0]	RESRVD	Reserved for Future Use	N -> R/W V -> R/W	0	This bit is Reserved for future use. This bit must always be written/loaded to its default state.

Table 107. Register Latency Code (Cycles) Versus Frequency

Latency Code	SDR SPI Register Transaction Latency Dummy Cycles (1S-1S-1S)[36, 38]		
	Frequency[35]	RDARG_4_0[37] RDDYB_4_0	RDPLB_0_0 RDIDN_0_0 RDSR1_0_0 RDSR2_0_0
00	50 MHz	0	0
01	133 MHz	1	0
10	133 MHz	1	1
11	166 MHz	2	2

Notes
35. SCK frequency > 166 MHz SDR, is not supported.
36. RDUID_4_0 alway has 32 cycles of latency. Maximum frequency under SDR SPI is 166 MHz, under HS-T SDR/DDR Octal is 200 MHz and under HL-T SDR/DDR Octal is 166 MHz.
37. RDARG_4_0 uses these dummy cycles for reading Volatile registers.
38. RDCRC_4_0 alway has 8 cycles of latency. Maximum frequency under SDR SPI is 166 MHz, under HS-T SDR/DDR Octal is 200 MHz and under HL-T SDR/DDR Octal is 166 MHz

5.3.6 Configuration Register 4 (CFR4x)

Configuration Register 4 controls the main Flash array read transactions burst wrap behavior and output driver impedance.

Table 108. Configuration Register 4

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
CFR4N[7:5] CFR4V[7:5]	IOIMPD[2:0]	I/O Driver Output Impedance selection	N -> R/W V -> R/W	101	Description: The IOIMPD[2:0] bits select the IO driver output impedance (drive strength). The output impedance configuration bits adjust the drive strength during normal device operation to meet system signal integrity requirements. Selection Options: 000 = 45Ω 001 = 120Ω 010 = 90Ω 011 = 60Ω 100 = 45Ω 101 = 30Ω (Factory Default) 110 = 20Ω 111 = 15Ω Dependency: N/A
CFR4N[4] CFR4V[4]	RBSTWP	Read Burst Wrap Enable selection	N -> R/W V -> R/W	0	Description: The RBSTWP bit selects the read burst wrap feature. It allows the device to enter and exit burst wrapped read mode during normal operation. The wrap length is selected by RBSTWL[1:0] bits. Selection Options: 0 = Read Wrapped Burst disabled 1 = Read Wrapped Burst enabled Dependency: RBSTWL[1:0] (CFR4x[1:0])
CFR4N[3] CFR4V[3]	ECC12S	Error Correction Code (ECC) 1-bit or 1-bit/2-bit error correction selection	N -> R/W V -> R/W	1	Description: The ECC12S bit selects between 1-bit ECC error detection/ correction or both 1-bit ECC error detection and correction and 2-bit ECC error detection. This configuration option affects Address Trap Register and ECC Counter Register functionalities as well. The host needs to erase and reprogram the data in the Semper Flash memory upon ECC configuration change (1-bit correction to 1-bit correction and 2-bit detection or vice versa). Selection Options: 0 = 1-bit ECC Error Detection/Correction 1 = 1-bit ECC Error Detection/Correction and 2-bit ECC error detection Dependency: N/A

Table 108. Configuration Register 4 (Continued)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
CFR4N[2] CFR4V[2]	DPDPOR	Deep Power Down power saving mode entry selection upon POR	N -> R/W V -> R	0	Description: The DPDPOR bit selects if the device will be in either Deep Power Down (DPD) mode or the Standby mode after the completion of POR. If enabled, DPDPOR configures the device to start in DPD mode to reduce current consumption until the device is needed. If the device is in DPD, a pulse on CS# or a Hardware reset will return the device to Standby mode. Selection Options: 0 = Standby mode is entered upon the completion of POR 1 = Deep Power Down Power mode is entered upon the completion of POR Dependency: N/A
CFR4N[1:0] CFR4V[1:0]	RBSTWL[1:0]	Read Burst Wrap Length selection	N -> R/W V -> R/W	00	Description: The RBSTWL[1:0] bits select the read burst wrap length and alignment during normal operation. It selects the fixed length/aligned group of 8-, 16-, 32-, or 64-bytes. Selection Options: 00 = 8 Bytes Wrap length 01 = 16 Bytes Wrap length 10 = 32 Bytes Wrap length 11 = 64 Bytes Wrap length Dependency: RBSTWP (CFR4x[4])

Table 109. Output Data Wrap Sequence

Wrap Boundary (Bytes)	Start Address (Hex)	Address Sequence (Hex)
Sequential	XXXXXX03	03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18.
8	XXXXXX00	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02.
8	XXXXXX07	07, 00, 01, 02, 03, 04, 05, 06, 07, 00, 01.
16	XXXXXX02	02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 00, 01, 02, 03.
16	XXXXXX0C	0C, 0D, 0E, 0F, 00, 01, 02, 03, 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E.
32	XXXXXX0A	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, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F.
32	XXXXXX1E	1E, 1F, 00, 01, 02, 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.
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, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D, 2E, 2F, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 3A, 3B, 3C, 3D, 3E, 3F 00, 01, 02.
64	XXXXXX2E	2E, 2F, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 3A, 3B, 3C, 3D, 3E, 3F, 00, 01, 02, 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, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D.

5.3.7 Interface CRC Enable Register (ICEV)

Interface CRC Enable Register controls the enabling/disabling of the Interface CRC function.

Table 110. Interface CRC Enable Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
ICEV[7:1]	RESVRD	Reserved for Future Use	V -> R	0000000	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
ICEV[0]	ITCRCE	Interface CRC Selection	V -> R/W	0	Description: The ITCRCE bit controls enabling/disabling of the Interface CRC function. Selection Options: 0 = Interface CRC Enabled 1 = Interface CRC Disabled Dependency: N/A

5.3.8 Interface CRC Check-value Register (ICRV)

The Interface CRC Check-value Register (ICRV) stores the results of the CRC calculation on the command and Data Content over the interface for Protection.

Table 111. Interface CRC Check-value Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (hex)	Description
ICRV[31:0]	ITCRCV[31:0]	Interface CRC Checksum Value	V -> R	0xFFFFFFFF	Description: The ITCRCV[31:0] bits store the check-value of the CRC process on the memory array data contained within the starting address and the ending address. Selection Options: Checksum Value Dependency: N/A

5.3.9 Memory Array Data Integrity Check CRC Register (DCRV)

The Memory Array Data Integrity Check CRC Register (DCRV) stores the results of the CRC calculation on the data contained between the specified starting and ending addresses.

Table 112. Memory Array Data Integrity Check CRC Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (hex)	Description
DCRV[31:0]	DTCRCV[31:0]	Memory Array Data Integrity Check CRC Checksum Value	V -> R	0x00000000	Description: The DTCRCV[31:0 bits store the checksum value of the CRC process on the memory array data contained within the starting address and the ending address. Selection Options: Checksum Value Dependency: N/A

5.3.10 ECC Status Register (ESCV)

The ECC Status Register (ESCV) contains the ECC status of any error correction action performed on the unit data whose byte was addressed during last read.

Note Unit data is defined as the number of bytes over which the ECC is calculated. HL-T/HS-T family devices have a 16 bytes (128 bits) unit data.

Table 113. ECC Status Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
ECSV[7:5]	RESRVD	Reserved for Future Use	V -> R	000	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
ECSV[4]	ECC2BT	ECC Error 2-bit Error Detection Flag	V -> R	0	Description: The ECC2BT bit indicates that a 2-bit ECC Error was detected in the data unit (16 bytes). A Clear ECC Status Register transaction (CLECC_0_0) will reset ECC2BT. **Note** ECC2BT is updated every time any memory address is read and is sticky, i.e. once it is set, it remains set. The ECC2BT status is maintained until a Clear ECC Status Register transaction (CLECC_0_0) is executed. **Note** ECC1BT is not valid if ECC2BT status flag is set. Selection Options: 0 = No 2-Bit ECC Error was detected in the data unit (16 bytes) 1 = 2-bit ECC Error was detected in the data unit (16 bytes) Dependency: N/A
ECSV[3]	ECC1BT	ECC Error 1-bit Error Detection and Correction Flag	V -> R	0	Description: The ECC1BT bit indicates that a 1-bit ECC Error was detected and corrected in the data unit (16 bytes). A Clear ECC Status Register transaction (CLECC_0_0) will reset ECC1BT. **Note** ECC1BT is updated every time any memory address is read and is sticky, i.e. once it is set, it remains set. The ECC1BT status is maintained until a Clear ECC Status Register transaction (CLECC_0_0) is executed. Selection Options: 0 = No 1-Bit ECC Error was detected in the data unit (16 bytes) 1 = 1-bit ECC Error was detected in the data unit (16 bytes) Dependency: N/A

Table 113. ECC Status Register (Continued)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
ECSV[2:0]	RESRVD	Reserved for Future Use	V -> R	000	This bit is Reserved for future use. This bit must always be written/loaded to its default state.

5.3.11 ECC Address Trap Register (EATV)

The ECC Address Trap Register (EATV) stores the address of the ECC unit data where either a 1-Bit/2-Bit error or only a 1-Bit error occurred during a read operation. It stores the ECC unit address of the first ECC error captured during a memory read operation since the last Clear ECC transaction.

Table 114. ECC Address Trap Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (hex)	Description
EATV[31:0]	ECCATP[31:0]	ECC 1-bit and 2-bit Error Address Trap Register	V -> R	0x00000000	Description: The Address Trap Register (ECCATP[31:0]) stores the ECC unit data address where a 1-Bit/2-Bit error occurred during a read operation. ECCATP[31:0] stores the ECC unit address of the first ECC error captured during a memory read operation since the last Clear ECC Status Register transaction (CLECC_0_0). **Note** ECCATP[31:0] is only updated during Read Instruction. **Note** Clear ECC Status Register transaction (CLECC_0_0), POR or Hardware/Software reset clears the EATV[31:0] to 0x00000000. Selection Options: ECC Error Data Unit Address Dependency: N/A

5.3.12 ECC Error Detection Count Register (ECTV)

The ECC Error Detection Counter Register (ECTV) stores the number of either 1-Bit/2-Bit or only 1-Bit ECC errors have occurred during read operations since the last POR or hardware/software reset.

Table 115. ECC Count Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (hex)	Description
ECTV[15:0]	ECCCNT[15:0]	ECC 1-bit and 2-bit Error Count Register	V -> R	0x0000	Description: The ECCCNT[15:0] stores the number of 1-bit/2-bit ECC errors occurred during read operations since the last POR or hardware/software reset. **Note** ECCCNT[15:0] is only updated during Read Instruction. **Note** Only one ECC error is counted for each data unit. If multiple read transactions access the same unit data containing an ECC error, the ECCCNT[15:0] will increment each time the unit data is read. **Note** Once the count reaches 0xFFFF, theECCCNT[15:0] will stop incrementing **Note** POR or Hardware/Software reset clears the ECCNT[15:0] to 0x0000. Selection Options: ECC Error Count Dependency: N/A

5.3.13 Advanced Sector Protection Register (ASPO)

The ASP Register (ASPO) configures the behavior of Advanced Sector Protection scheme.

Table 116. Advanced Sector Protection Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
ASPO[15:6]	RESRVD	Reserved for Future Use	N => R/1	1111111111	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
ASPO[5]	ASPRDP	Read Password Based Protection Selection	N => R/1	1	Description: The ASPRDP bit selects the Read Password Mode Protection mode. Read Password Protection mode works in conjunction with Password Protection mode to protect all sectors from Read/Erase/Program. Based on TBPROT configuration bit (CFR1x[5]), either the top or bottom sector is available for reading. Selection Options: 0 = Read Password Protection Mode is enabled 1 = Read Password Protection Mode is disabled Dependency: TBPROT (CFR1x[5])
ASPO[4]	ASPDYB	Dynamic Protection (DYB) for all sectors at power-up Selection	N => R/1	1	Description: The ASPDYB bit selects whether all DYB bits (sectors) are in the protected state following power-up or hardware reset. DYB bits will individually need to be reset to change sector protections. Selection Options: 0 = DYB based sector protection enabled at power-up or hardware reset 1 = DYB based sector protection disabled at power-up or hardware reset Dependency: N/A
ASPO[3]	ASPPPB	Permanent Protection (PPB) bits for all sectors programmability Selection	N => R/1	1	Description: The ASPPPB bit selects whether all PPB bits are OTP making PPB sector protection permanent. **Note** ASPPPB disables PPB erase transaction (ERPPB_0_0). Selection Options: 0 = PPB bits are OTP 1 = PPB bits can be erased and programmed as desired Dependency: N/A
ASPO[2]	ASPPWD	Password Based Protection Selection	N => R/1	1	Description: The ASPPWD bit selects the Password Protection Mode. Password Protection mode protects all PPB bits, all registers and all memory from erase/program till the correct password is entered. The ASPPWD can also be used in combination with the ASPRDP to protect sectors from being read as well till the correct password is provided – except for top or bottom sector which is available for reading based on TBPROT configuration bit (CFR1x[5]). **Note** When ASPPAS is selected, ASPO[15:0] are protected against program operations. Selection Options: 0 = Password Protection Mode is enabled 1 = Password Protection Mode is disabled Dependency: N/A
ASPO[1]	ASPPER	Persistent Protection Selection (Register Protection Selection)	N => R/1	1	Description: The ASPPER bit selects the Persistent Protection Mode. The Persistent Protection mode (ASPPER) protects the ASPO[15:0], CFR1x[6, 5, 4, 2], and CFR3x[3] registers from erase or program. Selection Options: 0 = Persistent Protection Mode is enabled 1 = Persistent Protection Mode is disabled Dependency: N/A

Table 116. Advanced Sector Protection Register (Continued)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
ASPO[0]	ASPPRM	Permanent Protection Selection	N => R/1	1	Description: The ASPPRM bit selects the Permanent Protection Mode. The Permanent Protection mode (ASPPRM) permanently protects the PPB bits from erase or program. ASPPRM bit should be programmed once all the PPB based sector protections are finalized. **Note** Permanent protection is independent of the PPBLOCK bit. Selection Options: 0 = Permanent Protection Mode is enabled 1 = Permanent Protection Mode is disabled Dependency: N/A

5.3.14 ASP Password Register (PWDO)

The ASP Password Register (PWDO) is used to permanently define a password.

Table 117. Password Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (hex)	Description
PWDO[63:0]	PASWRD[63:0]	Password Register	N => R/1	0xFFFFFFFFFFFFFFFF	Description: The PASWRD[63:0] permanently stores a password used in password protected modes of operation. When the Password Protection mode is enabled, this register will output the undefined data upon read password request. Selection Options: Password Dependency: N/A

5.3.15 ASP PPB Lock Register (PPLV)

The PPBLCK bit in the ASP PPB Lock Register (PPLV) is used to protect the PPB bits.

Table 118. ASP PPB Lock Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
PPLV[7:1]	RESVRD	Reserved for Future Use	V -> R	0000000	This bit is Reserved for future use. This bit must always be written/loaded to its default state.
PPLV[0]	PPBLCK	PPB Temporary Protection Selection	V -> R/W	1, ASPO[2:1]	Description: The PPBLCK bit is used to temporarily protect all the PPB bits. Selection Options: 1 = PPB Bits can be erased or programmed 0 = PPB bits are protected against erase or program till the next POR or hardware reset Dependency: N/A

5.3.16 ASP PPB Access Register (PPAV)

The ASP PPB Access Register (PPAV) is used to provide the state of each sector's PPB protection bit.

Table 119. ASP PPB Access Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
PPAV[7:0]	PPBACS[7:0]	Sector Based PPB Protection Status	N -> W/R	11111111	Description: The PPBACS[7:0] bits are used to provide the state of the individual sector's PPB bit. Selection Options: FF = PPB for the sector addressed by the Read PPB transaction (RDPPB_4_0) is 1, not protecting that sector from program or erase operations 00 = PPB for the sector addressed by the Read PPB transaction (RDPPB_4_0) is 0, protecting that sector from program or erase operations Dependency: N/A

5.3.17 ASP Dynamic Block Access Register (DYAV)

The ASP DYB Access Register (DYAV) is used to provide the state of each sector's DYB protection bit.

Table 120. ASP DYB Access Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
DYAV[7:0]	DYBACS[7:0]	Sector Based DYB Protection Status	V -> R/W	11111111	Description: The DYBACS[7:0] bits are used to provide the state of the individual sector's DYB bit. Selection Options: FF = DYB for the sector addressed by the Read DYB transaction (RDDYB_4_0) is 1, not protecting that sector from program or erase operations 00 = DYB for the sector addressed by the Read DYB transaction (RDDYB_4_0) is 0, protecting that sector from program or erase operations Dependency: N/A

5.3.18 AutoBoot Register (ATBN)

The AutoBoot Register (ATBN) provides a means to automatically read boot code as part of the power-on reset, or hardware reset process.

Table 121. AutoBoot Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
ATBN[31:9]	STADR[22:0]	Starting Address Selection where AutoBoot will start reading data from	N -> R/W	00000000000000000000000	Description: The STADR[22:0] bits set the starting address from which the device will output the read data. Selection Options: Address Bits Dependency: N/A
ATBN[8:1]	STDLY[7:0]	AutoBoot Read Starting Delay Selection	N -> R/W	00000000	Description: The STDLY[7:0] bits specify the initial delay (clock cycles) needed by the host before it can accept data. **Note** STDLY[7:0]=0x00 is valid for SPI up to 50 MHz. STDLY[7:0] = 0x01 or higher is valid for SPI/DPI/QPI up to 166 MHz. STDLY[7:0] = 0x05 or higher is valid for HL-T Octal up to 166 MHz and HS-T Octal up to 200 MHz. Selection Options: Address Bits Dependency: N/A

Table 121. AutoBoot Register (Continued)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
ATBN[0]	ATBTEN	AutoBoot Feature Selection	N -> R/W	0	Description: The ATBTEN bit enables or disables the AutoBoot feature. Selection Options: 0 = AutoBoot feature disabled 1 = AutoBoot feature enabled Dependency: N/A

5.3.19 Sector Erase Count Register (SECV)

The Sector Erase Count Register (SECV) contains the number of times the addressed sector has been erased.

Table 122. Sector Erase Count Register

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (hex)	Description
SECV[23]	SECCPT	Sector Erase Count Corruption Status Flag	V -> R	0x0	Description: The SECCPT bit is used to determine if the reported sector erase count is corrupted and was reset. **Note** If SECCPT is set due to count corruption, it will reset to 0 on the next successful erase operation on the selected sector. Selection Options: 0 = Sector Erase Count is not corrupted and is valid 1 = Sector Erase Count is corrupted and is not valid Dependency: N/A
SECV[22:0]	SECVAL[22:0]	Sector Erase Count Value	V -> R	0x000000	Description: The SECVAL[22:0] bits store the number of times a sector has been erased Selection Options: Value Dependency: N/A

5.3.20 EnduraFlex Architecture Selection Register (EFXx)

The EnduraFlex Architecture Selection registers (EFXx) define the long retention / high endurance regions based on a four pointer based architecture.

Table 123. EnduraFlex Architecture Selection Register (Pointer 4)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EFX4O[10:2]	EPTAD4[8:0]	EnduraFlex Pointer 4 Address Selection	N => R/1	111111111	Description: The EPTAD4[8:0] bits define the 9-bit address of the beginning sector from where the long retention / high endurance region is defined. Selection Options: Pointer Address Dependency: N/A
EFX4O[1]	ERGNT4	EnduraFlex Pointer 4 based Region Type Selection	N => R/1	1	Description: The ERGNT4 bit defines whether the region is long retention or high endurance. Selection Options: 0 = Long Retention Sectors 1 = High Endurance Sectors Dependency: N/A
EFX4O[0]	EPTEB4	EnduraFlex Pointer 4 Enable# Selection	N => R/1	1	Description: The EPTEN4 bit define whether the wear leveling pointer is enabled/disabled. Selection Options: 0 = Pointer Address Enabled 1 = Pointer Address Disabled Dependency: N/A

Table 124. EnduraFlex Architecture Selection Register (Pointer 3)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EFX3O[10:2]	EPTAD3[8:0]	EnduraFlex Pointer 3 Address Selection	N => R/1	111111111	Description: The EPTAD3[8:0] bits define the 9-bit address of the beginning sector from where the long retention / high endurance region is defined. Selection Options: Pointer Address Dependency: N/A
EFX3O[1]	ERGNT3	EnduraFlex Pointer 3 based Region Type Selection	N => R/1	1	Description: The ERGNT3 bit defines whether the region is long retention or high endurance. Selection Options: 0 = Long Retention Sectors 1 = High Endurance Sectors Dependency: N/A
EFX3O[0]	EPTEB3	EnduraFlex Pointer 3 Enable# Selection	N => R/1	1	Description: The EPTEN3 bit define whether the wear leveling pointer is enabled/disabled. Selection Options: 0 = Pointer Address Enabled 1 = Pointer Address Disabled Dependency: N/A

Table 125. EnduraFlex Architecture Selection Register (Pointer 2)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EFX2O[10:2]	EPTAD2[8:0]	EnduraFlex Pointer 2 Address Selection	N => R/1	111111111	Description: The EPTAD2[8:0] bits define the 9-bit address of the beginning sector from where the long retention / high endurance region is defined. Selection Options: Pointer Address Dependency: N/A
EFX2O[1]	ERGNT2	EnduraFlex Pointer 2 based Region Type Selection	N => R/1	1	Description: The ERGNT2 bit defines whether the region is long retention or high endurance. Selection Options: 0 = Long Retention Sectors 1 = High Endurance Sectors Dependency: N/A
EFX2O[0]	EPTEB2	EnduraFlex Pointer 2 Enable# Selection	N => R/1	1	Description: EPTEN2 bit define whether the wear leveling pointer is enabled/disabled. Selection Options: 0 = Pointer Address Enabled 1 = Pointer Address Disabled Dependency: N/A

Table 126. EnduraFlex Architecture Selection Register (Pointer 1)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EFX1O[10:2]	EPTAD1[8:0]	EnduraFlex Pointer 1 Address Selection	N => R/1	111111111	Description: The EPTAD1[8:0] bits define the 9-bit address of the beginning sector from where the long retention / high endurance region is defined. Selection Options: Pointer Address Dependency: N/A
EFX1O[1]	ERGNT1	EnduraFlex Pointer 1 based Region Type Selection	N => R/1	1	Description: The ERGNT1 bit defines whether the region is long retention or high endurance. Selection Options: 0 = Long Retention Sectors 1 = High Endurance Sectors Dependency: N/A
EFX1O[0]	EPTEB1	EnduraFlex Pointer 1 Enable# Selection	N => R/1	1	Description: The EPTEN1 bit define whether the wear leveling pointer is enabled/disabled. Selection Options: 0 = Pointer Address Enabled 1 = Pointer Address Disabled Dependency: N/A

Table 127. EnduraFlex Architecture Selection Register (Pointer 0)

Bit Number	Name	Function	Read/Write N = Nonvolatile V = Volatile	Factory Default (binary)	Description
EFX0O[1]	GBLSEL	All Sectors based Region type Selection	N => R/1	1	Description: The GBLSEL bit defines whether all sectors are defined as long retention region or high endurance region. **Note** If all other Pointer registers are disabled, this bit defines the behavior of the entire memory space and is hardwired to start at Sector 0. Selection Options: 0 = Long Retention Sectors 1 = High Endurance Sectors Dependency: N/A
EFX0O[0]	WRLVEN	Wear Leveling Enable Selection	N => R/1	1	Description: The WRLVEN bit enables/disables the wear leveling feature. Selection Options: 0 = Wear Leveling Disabled 1 = Wear Leveling Enabled Dependency: N/A

6. Transaction Table

6.1 HyperBus Transaction Table

Table 128. HyperBus Transaction Description

Transaction Name	Description	Cycles
RDMARY_1_0	**Read** transaction allows reading out the memory array data at the given address and places it on DQ[7:0].	1
ENSPIM_3_0	**Enter SPI** transaction changes the device interface from HyperBus to Legacy (x1) SPI.	3
SRASOE_1_0	**Software Reset / ASO Exit** transaction returns the device to reading memory array data mode when device is in ASO. It also clears SR0[5,4,3,1,0] when the device is not busy or is in the middle of a transaction sequence or in an ASO.	1
ENTDPD_3_0	**Enter Deep Power Down Mode** transaction moves the device in the lowest power consumption mode	3
RDVSTR_2_0	**Read Status Register** transaction allows the Status Register contents to be read with data placed on DQ[7:0].	2
CLVSTR_1_0	**Clear Status Register Failure Flags** transaction resets all failure flags being reported.	1
PRNPOR_4_0	**Program Nonvolatile POR Timer Register** transaction programs into the 16-bit POR Time Register the value which is multiplied by tPOR_CK (25 to 42 µs) to define the length of extension to the RSTO# pulse beyond tVCS with data placed on DQ[7:0].	4
RDNPOR_4_0	**Read Nonvolatile POR Timer Register** transaction allows reading the contents of the 16-bit POR Time Register and places it on DQ[7:0].	4
PGVINC_4_0	**Program Volatile Interrupt Configuration Register** transaction programs the 16-bit Interrupt Configuration Register with data placed on DQ[7:0].	4
RDVINC_4_0	**Read Volatile Interrupt Configuration Register** transaction allows reading the contents of the 16-bit Interrupt Configuration Register and places it on DQ[7:0].	4
PGVINS_4_0	**Program Volatile Interrupt Status Register** transaction programs the 16-bit Interrupt Status Register with data placed on DQ[7:0].	4
RDVINS_4_0	**Read Volatile Interrupt Status Register** transaction allows reading the contents of the 16-bit Interrupt Status Register and places it on DQ[7:0].	4
PGVCR1_4_0	**Program Volatile Configuration Register 1** transaction programs the 16-bit Volatile Configuration Register 0 with data placed on DQ[7:0].	4
PGVCR2_4_0	**Program Volatile Configuration Register 2** transaction programs the 16-bit Volatile Configuration Register 1 with data placed on DQ[7:0].	4
RDVCR1_4_0	**Read Volatile Configuration Register 1** transaction allows reading the contents of the 16-bit Volatile Configuration Register 0 and places it on DQ[7:0].	4
RDVCR2_4_0	**Read Volatile Configuration Register 2** transaction allows reading the contents of the 16-bit Volatile Configuration Register 1 and places it on DQ[7:0].	4
PGNCR1_4_0	**Program Nonvolatile Configuration Register 1** transaction programs the 16-bit Nonvolatile Configuration Register 0 with data placed on DQ[7:0].	4
PGNCR2_4_0	**Program Nonvolatile Configuration Register 2** transaction programs the 16-bit Nonvolatile Configuration Register 1 with data placed on DQ[7:0].	4
ERNC12_3_0	**Erase Nonvolatile Configuration Registers 1 and 2** transaction erases the contents of the two 16-bit Nonvolatile Configuration registers (0, 1).	3
RDNCR1_4_0	**Read Nonvolatile Configuration Register 1** transaction allows reading the contents of the 16-bit Nonvolatile Configuration Register 0 and places it on DQ[7:0].	4
RDNCR2_4_0	**Read Nonvolatile Configuration Register 2** transaction allows reading the contents of the 16-bit Nonvolatile Configuration Register 1 and places it on DQ[7:0].	4
PGWORD_4_0	**Program Word** transaction programs the data word (0's) supplied on DQ[7:0] in the addressed memory array.	4
LDBUFR_6_0	**Load Write Buffer** transaction loads up to 256/512 bytes of data (0's) supplied on DQ[7:0] in the write buffer.	6
PGBFCM_1_0	**Program Write Buffer Confirm** transaction tells the device to program the data loaded into the write buffer into the addressed memory array.	1
RSTWBA_3_0	**Reset Write to Buffer Abort** transaction resets the Write Buffer Abort Status Flag (WRBFAB - STRV[3]) and Programming Error Status Flag (PRGERR - STRV[4]) in the Status Register.	3
ERCHIP_6_0	**Erase Chip** transaction sets all bits to 1 (all bytes are FFh) inside the entire flash memory array.	6
ERSCTR_6_0	**Erase Sector** transaction sets all the bits of an addressed 256 KB sector or a 4 KB sector to 1 (all bytes are FFh).	6
BLKCHK_1_0	**Blank Check** transaction confirms if the selected Flash Memory Array sector is fully erased. ERSERR (STRV[5]- Bit 5 of the Status Register) will be cleared to 0 if the sector is erased and set to 1 if not erased.	1
EVERST_1_0	**Evaluate Erase Status** transaction verifies that the last erase operation on the addressed sector was completed successfully. f the selected sector was successfully erased the SESTAT (STRV[5]- Bit 0 of the Status Register) is set to 1.	1
SPERSE_1_0	**Suspend Erase** transaction allows the system to interrupt an erase operation	1

Table 128. HyperBus Transaction Description (Continued)

Transaction Name	Description		Cycles
RSERSE_1_0	**Resume Erase** transaction allows the system to resume an erase operation		1
SPPROG_1_0	**Suspend Program** transaction allows the system to interrupt a program operation		1
RSPROG_1_0	**Resume Program** transaction allows the system to resume a program operation		1
IDSFE1_3_1	ASO Device ID Unique ID SFDP	**ID/Unique ID/SFDP ASO Entry 1** transaction allows reading Device ID, Unique ID and SFDP parameters. This entry transaction uses the Sector Address (SA) in the command to determine which sector will be overlaid.	3
IDSFE2_1_1		**ID/Unique ID/SFDP ASO Entry 2** transaction allows reading Device ID, Unique ID and SFDP parameters. This entry transaction uses the Sector Address (SA) in the command to determine which sector will be overlaid.	1
RDIDSF_1_1		**Read ID/Unique ID/SFDP** transaction allows reading the Device ID, Unique ID and SFDP parameters at the given address and places it on DQ[7:0].	1
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1
SSRENT_3_1	ASO SSR	**SSR ASO Entry** transaction allows accessing the SSR. This entry transaction uses the Sector Address (SA) in the command to determine which sector will be overlaid.	3
RD_SSR_1_1		**Read SSR** transaction allows reading the SSR data at the given address and places it on DQ[7:0].	1
PG_SSR_4_1		**Program SSR Word** transaction programs the data word (0's) supplied on DQ[7:0] in the addressed SSR.	4
LDBSSR_5_1		**Load SSR Buffer** transaction loads up to 256/512 bytes of data (0's) supplied on DQ[7:0] in the write buffer.	5
PGCSSR_1_1		**Program SSR Buffer Confirm** transaction tells the device to program the data loaded into the write buffer into the addressed SSR.	1
RSWSSR_3_1		**Reset Write to Buffer Abort** transaction resets the Write Buffer Abort Status Flag (WRBFAB - STRV[3]) in the Status Register.	3
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1
ASPENT_3_1	ASO Advanced Sector Protection	**Advanced Sector Protection ASO Entry** transaction allows accessing the Advanced Sector Protection Configuration Register. This entry transaction does not use a sector address from the entry transaction to overlay. The ASP Configuration Register appears at word location 0 in the device address space	3
PGOASP_2_1		**Program One-Time-Programmable Advanced Sector Protection Register** transaction programs the 16-bit OTP ASP Configuration Register with data placed on DQ[7:0].	2
RDOASP_1_1		**Read One-Time-Programmable Advanced Sector Protection Register** transaction allows reading the contents of the 16-bit OTP ASP Configuration Register using device address 0 and places it on DQ[7:0].	1
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1
PWDENT_3_1	ASO Password	**Password ASO Entry** transaction allows accessing the 64-bit password. This entry transaction does not use a sector address from the entry transaction to overlay. The Password appears at word locations 0 to 3 in the device address space.	3
PGNPWD_2_1		**Program Nonvolatile Password** transaction programs the 64-bit Password with data placed on DQ[7:0].	2
RDNPWD_1_1		**Read Nonvolatile Password** transaction allows reading the contents of the 64-bit Password using device address 0 to 3 and places it on DQ[7:0].	1
ULNPWD_7_1		**Unlock Nonvolatile Password** transaction allows entering the 64-bit Password on DQ[7:0] to unlock the device for access.	7
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1

Table 128. HyperBus Transaction Description (Continued)

Transaction Name	Description		Cycles
PPBENT_3_1	ASO Persistent Protection Bits	**Persistent Protection Bits ASO Entry** transaction allows accessing the Persistent Protection bits (PPB) associated with sectors. This entry transaction does not use a sector address from the entry transaction to overlay. The PPB bit for a sector appears in bit 0 of all word locations in the sector.	3
PGNPPB_2_1		**Program Nonvolatile Persistent Protection Bits** transaction programs the PPB bit corresponding to a sector with data placed on DQ[7:0]. The PPB bit for a sector appears in bit 0 of all word locations in the sector.	2
ERNPPB_2_1		**Erase Nonvolatile Persistent Protection Bits** transaction erases all the PPB bits.	2
RSWPPB_3_1		**Reset Write to Buffer Abort** transaction resets the Write Buffer Abort Status Flag (WRBFAB - STRV[3]) in the Status Register caused by a PPB program failure.	3
RDNPPB_1_1		**Read Nonvolatile Persistent Protection Bits** transaction allows reading the PPB bit corresponding to a sector and places it on DQ[7:0]. The PPB bit for a sector appears in bit 0 of all word locations in the sector.	1
PRTSTS_2_1		**Sector Protection Status** transaction provides the protection status of the addressed sector. Data out during a SA Protection Status Read indicates whether the indicated sector is protected in bits 0-2: Bit 0 – Indicates whether the indicated sector is protected (0 = protected, 1 = unprotected). Bit 1 – Protected using the sector's DYB bit (0 = protected, 1 = unprotected). Bit 2 – Protected using the sector's PPB bit (0 = protected, 1 = unprotected). Bits 3 through 15 are all 1s.	2
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1
PPLENT_3_1	ASO PPB Lock	**Persistent Protection Lock ASO Entry** transaction allows accessing the global Persistent Protection Lock bit. This entry transaction does not use a sector address from the entry transaction to overlay. The global Persistent Protection Lock bit appears in bit 0 of all word locations in the device.	3
CLVPPL_2_1		**Clear Volatile Persistent Protection Lock** transaction clears the global Persistent Protection Lock bit.	2
RDVPPL_1_1		**Read Volatile Persistent Protection Lock** transaction allows reading the global Persistent Protection Lock bit and places it on DQ[7:0]. The PPL bit appears in bit 0 of all word locations in the sector.	1
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1
DYBENT_3_1	ASO Dynamic Protection Bits	**Dynamic Protection Bits ASO Entry** transaction allows accessing the Dynamic Protection bits (DYB) associated with sectors. This entry transaction does not use a sector address from the entry transaction to overlay. The DYBB bit for a sector appears in bit 0 of all word locations in the sector.	3
STVDYB_2_1		**Set Volatile Dynamic Protection Bit** transaction sets the DYB bit corresponding to a sector with data placed on DQ[7:0]. The DYB bit for a sector appears in bit 0 of all word locations in the sector.	2
CLVDYB_2_1		**Clear Volatile Dynamic Protection Bit** transaction clears the DYB bit corresponding to a sector with data placed on DQ[7:0]. The DYB bit for a sector appears in bit 0 of all word locations in the sector.	2
RDVDYB_1_1		**Read Volatile Dynamic Protection Bit** transaction allows reading the DYB bit corresponding to a sector and places it on DQ[7:0]. The DYB bit for a sector appears in bit 0 of all word locations in the sector.	1
PRTSTS_2_1		**Sector Protection Status** transaction provides the protection status of the addressed sector. Data out during a SA Protection Status Read indicates whether the indicated sector is protected in bits 0-2: Bit 0 – Indicates whether the indicated sector is protected (0 = protected, 1 = unprotected). Bit 1 – Protected using the sector's DYB bit (0 = protected, 1 = unprotected). Bit 2 – Protected using the sector's PPB bit (0 = protected, 1 = unprotected). Bits 3 through 15 are all 1s.	2
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1

Table 128. HyperBus Transaction Description (Continued)

Transaction Name	Description		Cycles
ECCENT_3_1	ASO Error Correction Codes	**Error Correction (ECC) ASO Entry** transaction allows accessing the error correction action (ECC status) of any half-page of the Flash Memory Array.	3
RDECST_1_1		**Read Error Correction (ECC) Status** transaction provides the ECC Status value for the addressed half-page on DQ[7:0]. A single word of status is displayed at any word location within a half-page.	1
RDADTL_2_1		**Read Address Trap Register Lower Word** transaction provides the lower 16-bits of the error correction action (ECC) related address value stored in the Address Trap Register (32-bits) on DQ[7:0].	2
RDADTU_2_1		**Read Address Trap Register Upper Word** transaction provides the upper 16-bits of the error correction action (ECC) related address value stored in the Address Trap Register (32-bits) on DQ[7:0].	2
RDCONT_2_1		**Read ECC Count Value Register** transaction provides the ECC count of th number of error correction actions on DQ[7:0].	2
CLRECC_1_1		**Clear ECC Error Status Failure Flags** transaction resets all failure flags and interrupts being reported.	1
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1
ICRCEN_3_1	ASO Interface CRC	**Interface CRC Register ASO Entry** transaction allows accessing the contents of the Interface CRC Register. Exiting Interface CRC Register ASO clears the Interface CRC Register.	3
RDICRC_1_1		**Read Volatile Interface CRC Register** transaction provides the contents of the Interface CRC Register on DQ[7:0]. Addresses 0x00 and 0x01 define the lower and upper 16-bit Interface CRC Register values.	1
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1
DICREN_3_1	ASO Data Integrity CRC	**Data Integrity CRC Register ASO Entry** transaction allows accessing the Data Integrity CRC check-value. While the Data Integrity CRC calculation is not suspended the Data Integrity CRC ASO overlays the entire Flash memory array. When the CRC calculation is suspended the Flash memory array is visible for reading.	3
LDSTAD_1_1		**Load Start Address** transaction loads the Data Integrity CRC beginning address location.	1
LDENAD_1_1		**Load End Address** transaction loads the Data Integrity CRC ending address location.	1
SP_DIC_1_1		**Suspend Data Integrity CRC** transaction allows the system to interrupt the Data Integrity CRC calculation operation.	1
RDCMRY_1_1		**Read Memory Array during Data Integrity CRC suspend** transaction allows reading out the memory array data at the given address and places it on DQ[7:0].	1
RS_DIC_1_1		**Resume Data Integrity CRC transaction** allows the system to resume the suspended Data Integrity CRC calculation operation.	1
RDDICL_2_1		**Read Data Integrity CRC Register Lower Word** transaction provides the lower 16-bits of the Data Integrity CRC check-value on DQ[7:0].	2
RDDICU_2_1		**Read Data Integrity CRC Register Upper Word** transaction provides the upper 16-bits of the Data Integrity CRC check-value on DQ[7:0].	2
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1
ATBNEN_3_1	ASO AutoBoot	**AutoBoot Nonvolatile Register ASO Entry** transaction allows accessing the AutoBoot Register. This entry transaction does not use a sector address from the entry transaction.	3
PGNATB_2_1		**Program Nonvolatile AutoBoot Register** transaction programs the programs the 16-bit Nonvolatile AutoBoot Register with data placed on DQ[7:0].	2
RDATBN_1_0		**Read Nonvolatile AutoBoot Register** transaction allows reading the contents of the 32-bit Nonvolatile AutoBoot Register and places it on DQ[7:0]. Addresses 0x00 and 0x01 define the lower and upper 16-bit AutoBoot Register values.	1
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1

Table 128. HyperBus Transaction Description (Continued)

Transaction Name	Description		Cycles
SECTEN_3_1	ASO Sector Erase Count	**Sector Erase Count Volatile Register ASO Entry** transaction allows accessing the Sector Erase Count Register. This entry transaction does not use a sector address from the entry transaction.	3
LDSRAD_2_1		**Load Sector Address** transaction loads the sector address whose erase count is desired.	2
RDSECV_1_0		**Read Volatile Sector Erase Count Register** transaction allows reading the contents of the 16-bit Volatile Sector Erase Count Register and places it on DQ[7:0].	1
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1
ENX_EN_3_1	ASO EnduraFlex	**EnduraFlex Pointer Selection (Partitions) One-Time_Programmable Register ASO Entry** transaction allows accessing the EnduraFlex Pointer registers. This entry transaction does not use a sector address from the entry transaction.	3
PGOENX_2_1		**Program One-Time-Programmable EnduraFlex Registers [4:0]** transaction programs the One-Time-Programmable EnduraFlex registers [3:0] with data placed on DQ[7:0]. Addresses 0x00, 0x01, 0x02, 0x03, 0x04 define the four EnduraFlex Register values.	2
RDOENX_1_1		**Read One-Time-Programmable EnduraFlex Registers [4:0]** transaction reads the One-Time-Programmable EnduraFlex registers [3:0] and places it on DQ[7:0]. Addresses 0x00, 0x01, 0x02, 0x03, 0x04 define the four EnduraFlex Register values.	1
ASOEXT_1_1		**ASO Exit** transaction returns the device to reading memory array data mode when device is in an ASO. If any of the ASO Entry commands were issued, an ASO Exit command must be issued to reset the device into Read mode.	1

Table 129. HyperBus Transaction Cycles

Transaction Name	Bus Cycles																				
	First			Second			Third			Fourth			Fifth			Sixth			Seventh		
	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data
RDMARY_1_0	101b	DA[1:0] || RDA [AMAX:0]	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ENSPIM_3_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	F5h	–	–	–	–	–	–	–	–	–	–	–	–	–
SRASOE_1_0	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ENTDPD_3_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:0] 'bX}	B9h	–	–	–	–	–	–	–	–	–	–	–	–	–
RDVSTR_2_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	70h	101b	DA[1:0] || {[AMAX:0] 'bX}	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
CLVSTR_1_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	71h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
PRNPOR_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	34h	001b	DA[1:0] || {[AMAX:0] 'bX}	Pr_data [15:0]	–	–	–	–	–	–	–	–	–	–
RDNPOR_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	3Ch	101b	DA[1:0] || {[AMAX:0] 'bX}	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–
PGVINC_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	36h	001b	DA[1:0] || {[AMAX:0] 'bX}	Pr_data [15:0]	–	–	–	–	–	–	–	–	–	–
RDVINC_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	C4h	101b	DA[1:0] || {[AMAX:0] 'bX}	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–
PGVINS_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	37h	001b	DA[1:0] || {[AMAX:0] 'bX}	FFFFh	–	–	–	–	–	–	–	–	–	–
RDVINS_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	C5h	101b	DA[1:0] || {[AMAX:0] 'bX}	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–
PGVCR1_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	38h	001b	DA[1:0] || {[AMAX:0] 'bX}	Pr_data [15:0]	–	–	–	–	–	–	–	–	–	–
PGVCR2_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	3Ah	001b	DA[1:0] || {[AMAX:0] 'bX}	Pr_data [15:0]	–	–	–	–	–	–	–	–	–	–

Table 129. HyperBus Transaction Cycles (Continued)

Transaction Name	Bus Cycles																				
	First			Second			Third			Fourth			Fifth			Sixth			Seventh		
	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data
RDVCR1_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	C7h	101b	DA[1:0] || {[AMAX:0] 'bX}	Rd_data [15:0]	–	–	–	–	–	–	–	–	–
RDVCR2_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	C9h	101b	DA[1:0] || {[AMAX:0] 'bX}	Rd_data [15:0]	–	–	–	–	–	–	–	–	–
PGNCR1_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	39h	001b	DA[1:0] || {[AMAX:0] 'bX}	Pr_data [15:0]	–	–	–	–	–	–	–	–	–
PGNCR2_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	3Bh	001b	DA[1:0] || {[AMAX:0] 'bX}	Pr_data [15:0]	–	–	–	–	–	–	–	–	–
ERNC12_3_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	C8h	–	–	–	–	–	–	–	–	–	–	–	–	–
RDNCR1_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	C6h	101b	DA[1:0] || {[AMAX:0] 'bX}	Rd_data [15:0]	–	–	–	–	–	–	–	–	–
RDNCR2_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	CAh	101b	DA[1:0] || {[AMAX:0] 'bX}	Rd_data [15:0]	–	–	–	–	–	–	–	–	–
PGWORD_4_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	A0h	001b	DA[1:0] || Pr_addr [AMAX:0]	Pr_data [15:0]	–	–	–	–	–	–	–		–
LDBUFR_6_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	25h	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	Wd_cnt	001b	DA[1:0] || WBL [AMAX:0]	Pr_data [15:0]	001b	DA[1:0] || WBL [AMAX:0]	Pr_data [15:0]	–	–	–
PGBFCM_1_0	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	29h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RSTWBA_3_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h		–	–	–	–	–	–	–	–	–	–	–	–
ERCHIP_6_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	80h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	10h	–	–	–
ERSCTR_6_0	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	80h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	30h	–	–	–

Table 129. HyperBus Transaction Cycles (Continued)

Transaction Name	Bus Cycles																				
	First			Second			Third			Fourth			Fifth			Sixth			Seventh		
	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data
BLKCHK_1_0	001b	DA[1:0] || Sec_addr [AMAX:AMIN] || 555h	33h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
EVERST_1_0	001b	DA[1:0] || Sec_addr [AMAX:AMIN] || 555h	D0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SPERSE_1_0	001b	DA[1:0] || {[AMAX:0] 'bX}	B0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RSERSE_1_0	001b	DA[1:0] || {[AMAX:0] 'bX}	30h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SPPROG_1_0	001b	DA[1:0] || {[AMAX:0] 'bX}	51h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RSPROG_1_0	001b	DA[1:0] || {[AMAX:0] 'bX}	50h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
IDSFE1_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	90h	–	–	–	–	–	–	–	–	–	–	–	–	–
IDSFE2_1_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	98h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDIDSF_1_1	101b	DA[1:0] || RDA [AMAX:0]	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SSRENT_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || Sec_addr [AMAX:AMIN] || 555h	88h	–	–	–	–	–	–	–	–	–	–	–	–	–
RD_SSR_1_1	101b	DA[1:0] || RDA [AMAX:0]	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
PG_SSR_4_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	A0h	001b	DA[1:0] || Pr_addr [AMAX:0]	Pr_data [15:0]	–	–	–	–	–	–	–	–	–

Table 129. HyperBus Transaction Cycles (Continued)

Transaction Name	Bus Cycles																				
	First			Second			Third			Fourth			Fifth			Sixth			Seventh		
	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data
LDBSSR_5_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	25h	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	Wd_cnt	001b	DA[1:0] || WBL [AMAX:0]	Pr_data [15:0]	001b	DA[1:0] || WBL [AMAX:0]	Pr_data [15:0]	–	–	–
PGCSSR_1_1	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	29h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RSWSSR_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASPENT_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	40h	–	–	–	–	–	–	–	–	–	–	–	–	–
PGOASP_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	A0h	001b	DA[1:0] || {[AMAX:0] 'bX}	Pr_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDOASP_1_1	101b	DA[1:0] || {[AMAX:0] 'bX}	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
PWDENT_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	60h	–	–	–	–	–	–	–	–	–	–	–	–	–
PGNPWD_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	A0h	001b	DA[1:0] || {[AMAX:2] 'bX} || {PSWD Addr[1:0]}	P_PWDX [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDNPWD_1_1	101b	DA[1:0] || {[AMAX:2] 'bX} || {PSWD Addr[1:0]}	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ULNPWD_7_1	001b	DA[1:0] || {[AMAX:2] 'bX} || {0h}	25h	001b	DA[1:0] || {[AMAX:2] 'bX} || {0h}	03h	001b	DA[1:0] || {[AMAX:2] 'bX} || {0h}	PWD0 [15:0]	001b	DA[1:0] || {[AMAX:2] 'bX} || {1h}	PWD1 [31:16]	001b	DA[1:0] || {[AMAX:2] 'bX} || {2h}	PWD2 [47:32]	001b	DA[1:0] || {[AMAX:2] 'bX} || {3h}	PWD3 [63:48]	001b	DA[1:0] || {[AMAX:2] 'bX} || {0h}	29h

Table 129. HyperBus Transaction Cycles (Continued)

Transaction Name	Bus Cycles																					
	First			Second			Third			Fourth			Fifth			Sixth			Seventh		
	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
PPBENT_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	C0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–
PGNPPB_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	A0h	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	00h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ERNPPB_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	80h	001b	DA[1:0] || {[AMAX:12] 'bX} || 000h	30h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RSWPPB_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDNPPB_1_1	101b	DA[1:0] || Sec_addr [AMAX:AMIN]	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
PRTSTS_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	60h	101b	DA[1:0] || Sec_addr [AMAX:AMIN]	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
PPLENT_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	50h	–	–	–	–	–	–	–	–	–	–	–	–	–	–
CLVPPL_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	A0h	001b	DA[1:0] || {[AMAX:0] 'bX}	00h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDVPPL_1_1	101b	DA[1:0] || {[AMAX:0] 'bX}	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
DYBENT_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	E0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–

Table 129. HyperBus Transaction Cycles (Continued)

Transaction Name	Bus Cycles																				
	First			Second			Third			Fourth			Fifth			Sixth			Seventh		
	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data
STVDYB_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	A0h	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	00h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
CLVDYB_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	A0h	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	01h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDVDYB_1_1	101b	DA[1:0] || Sec_addr [AMAX:AMIN]	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
PRTSTS_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	60h	101b	DA[1:0] || Sec_addr [AMAX:AMIN]	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ECCENT_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	75h	–	–	–	–	–	–	–	–	–	–	–	–	–
RDECST_1_1	101b	DA[1:0] || RDA [AMAX:0]	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDADTL_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	60h	101b	DA[1:0] || {[AMAX:2] 'bX} || {00b}	Rd_data 0 [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDADTU_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	60h	101b	DA[1:0] || {[AMAX:2] 'bX} || {01b}	Rd_data 0 [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDCONT_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	60h	101b	DA[1:0] || {[AMAX:2] 'bX} || {10b}	Rd_data 0 [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
CLRECC_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	50h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ICRCEN_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	76h	–	–	–	–	–	–	–	–	–	–	–	–	–

Table 129. HyperBus Transaction Cycles (Continued)

Transaction Name	Bus Cycles																				
	First			Second			Third			Fourth			Fifth			Sixth			Seventh		
	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data
RDICRC_1_1	101b	DA[1:0] || {[AMAX:2] 'bX} || {00b}	Rd_data 0 [15:0]	101b	DA[1:0] || {[AMAX:2] 'bX} || {01b}	Rd_data 1 {31:16]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
DICREN_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	78h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
LDSTAD_1_1	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	C3h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
LDENAD_1_1	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	3Ch	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SP_DIC_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	C0	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDCMRY_1_1	101b	DA[1:0] || RDA [AMAX:0]	Rd_data [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RS_DIC_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	C1	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDDICL_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	60h	101b	DA[1:0] || {[AMAX:2] 'bX} || {00b}	Rd_data 0 [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDDICU_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	60h	101b	DA[1:0] || {[AMAX:2] 'bX} || {01b}	Rd_data 0 [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ATBNEN_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	14h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
PGNATB_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	A0h	001b	DA[1:0] || {[AMAX:1] 'bX} || {AUTOBOOT Addr[0]}	P_AUTOBOOTX [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–

Table 129. HyperBus Transaction Cycles (Continued)

Transaction Name	Bus Cycles																				
	First			Second			Third			Fourth			Fifth			Sixth			Seventh		
	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data	CA[47:45]	CA[44:0] Address	DQ[7:0] Data
RDATBN_1_0	101b	DA[1:0] || {[AMAX:2] 'bX} || {00b}	Rd_data 0 [15:0]	101b	DA[1:0] || {[AMAX:2] 'bX} || {01b}	Rd_data 1 {31:16]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SECTEN_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	15h	–	–	–	–	–	–	–	–	–	–	–	–	–	–
LDSRAD_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	A0h	001b	DA[1:0] || Sec_addr [AMAX:AMIN]	5Dh	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDSECV_1_0	101b	DA[1:0] || {[AMAX:2] 'bX} || {00b}	Rd_data 0 [15:0]	101b	DA[1:0] || {[AMAX:2] 'bX} || 01b}	Rd_data 1 {31:16]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ENX_EN_3_1	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	AAh	001b	DA[1:0] || {[AMAX:12] 'bX} || 2AAh	55h	001b	DA[1:0] || {[AMAX:12] 'bX} || 555h	16h	–	–	–	–	–	–	–	–	–	–	–	–	–	–
PGOENX_2_1	001b	DA[1:0] || {[AMAX:0] 'bX}	A0h	001b	DA[1:0] || {[AMAX:3] 'b0} || Pointer Addr[2:0]	Pr_data X [15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RDOENX_1_1	101b	DA[1:0] || {[AMAX:3] 'b0} || Pointer Addr[2:0]	Rd_data {15:0]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ASOEXT_1_1	001b	DA[1:0] || {[AMAX:0] 'bX}	F0h	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–

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

EFPRx = EnduraFlex Architecture Pointer Address Registers 0, 1, 2, 3, 4

DA = Die Address (DA[1:0] = 00b for all the devices)

Notes

39. All values are in hexadecimal. All addresses reference 16-bit words.
40. Except for the following, all bus cycles are write cycle: read cycle during Read, ID Read (Manufacturing ID / Device ID), Indicator Bits, SSR Read, SSR Lock Read, and 2nd cycle of Status Register Read.
41. Data bits DQ15-DQ8 are ‘don't care’ in command sequences, except for RD, PD, WC and PWD.
42. 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.)
43. No unlock or command cycles required when reading array data.
44. The Reset command is required to return to reading array data when device is in ID-SFDP (Autoselect) mode.
45. Command is valid when device is ready to read array data or when device is in ID-SFDP mode.
46. The system can read and program / program suspend in non-erasing sectors, or enter the ID-SFDP ASO, when in the Erase Suspend mode. The Erase Suspend command is valid only during a sector erase operation.
47. The Erase Resume / Program Resume command is valid only during the Erase Suspend / Program Suspend modes.
48. 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.
49. The Exit command returns the device to reading the array.
50. 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).
51. All ASP Register bits are OTP. 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.
52. If any of the Entry commands was issued, an Exit command must be issued to reset the device into Read mode.
53. Bit 0 = 0 indicates the Protected State, Bit 0 = 1 indicates the Unprotected State - Bits 1 through 15 are all 0s for Protected and all 1s for Unprotected. 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”. Note that PPB Program command gets aborted if bits 1 through 15 are not all 0s.
54. Data out during a Register Read transaction 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.
55. Data out during a SA Protection Status Read indicates whether the indicated sector is protected in bits 0-2.
    Bit 0 – Indicates whether the indicated sector is protected (0 = protected, 1 = unprotected).
    Bit 1 – Protected using the sector’s DYB bit (0 = protected, 1 = unprotected).
    Bit 2 – Protected using the sector’s PPB bit (0 = protected, 1 = unprotected).
    Bits 3 through 15 are all 1s.
56. 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.
57. Reset / ASO Exit behavior
58. Clears SR0[5,4,3,1,0] when the device not busy or in the middle of a command sequence.
    Clears SR0[5,4,3,1,0] when the device not busy or in an ASO.
59. The device will return to Standby mode once CS# goes HIGH if an illegal command sequence is entered.
60. Reset, ASO Exit, Status Register Read and Status Register Clear commands are globally applicable in all ASOs.
61. Wrap feature is not available in AutoBoot mode when Reading from the memory array.
62. Programming Power-On-Reset Timer Register and Nonvolatile Configuration registers should never be suspended. Suspending will produce indeterminate results and a hardware reset will be required to return to standby mode.
63. Enter SPI mode command sequence must be followed by a Read Any Register command (RDAR) in SPI mode for proper transitioning from HyperBus to SPI interfaces.

6.2 Legacy (x1) SPI Transaction Table

Table 130. SPI (1S-1S-1S) Transaction Table

Function	Transaction Name	Description	Prerequisite Transaction	Byte 1 (Hex)	Byte 2 (Hex)	Byte 3 (Hex)	Byte 4 (Hex)	Byte 5 (Hex)	Byte 6 (Hex)	Byte 7 (Hex)	Byte 8 (Hex)	Byte 9 (Hex)	Transaction Format	Max Frequency (MHz)	Address Length
Read Device ID	RDIDN_0_0	**Read manufacturer and device identification** transaction provides read access to manufacturer and device identification.	–	9F (CMD)	–	–	–	–	–	–	–	–	Figure 10	166	N/A
	RSFDP_3_0	**Read JEDEC Serial Flash Discoverable Parameters** transaction sequentially accesses the Serial Flash Discovery Parameters (SFDP).	–	5A (CMD)	00 (ADDR)	00 (ADDR)	00 (ADDR)	–	–	–	–	–	Figure 11	156	3
	RDUID_0_0	**Read Unique ID** accesses a factory programmed 64-bit number which is unique to each device.	–	4C (CMD)	–	–	–	–	–	–	–	–	Figure 10	166	N/A
Register Access	RDSR1_0_0	**Read Status Register 1** transaction allows the Status Register 1 contents to be read from DQ1/SO.	–	05 (CMD)	–	–	–	–	–	–	–	–			
	RDSR2_0_0	**Read Status Register-2** transaction allows the Status Register-2 contents to be read from DQ1/SO.	–	07 (CMD)	–	–	–	–	–	–	–	–			
	RDARG_C_0	**Read Any Register** transaction provides a way to read all addressed nonvolatile and volatile device registers.	–	65 (CMD)	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–	–	Figure 11		3
			–		ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–			4
	WRENB_0_0	**Write Enable** sets the Write Enable Latch bit of the Status Register 1 to 1 to enable write, program, and erase transactions.	–	06 (CMD)	–	–	–	–	–	–	–	–	Figure 5		N/A
	WRDIS_0_0	**Write Disable** sets the Write Enable Latch bit of the Status Register 1 to 0 to disable write, program, and erase transactions execution.	–	04 (CMD)	–	–	–	–	–	–	–	–			
	WRARG_C_1	**Write Any Register** transaction provides a way to write all addressed Nonvolatile and Volatile Device registers.	WRENB_0_0	71 (CMD)	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	Input Data [7:0]	–	–	–	–	Figure 8		3
					ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	Input Data [7:0]	–	–	–			4
	CLPEF_0_0	**Clear Program and Erase Failure Flags** transaction resets STR1V[5] (Erase failure flag) and STR1V[6] (Program failure flag).	WRENB_0_0	82 (CMD)	–	–	–	–	–	–	–	–	Figure 5		N/A
ECC	RDECC_4_0	**Read ECC Status** is used to determine the ECC status of the addressed data unit.	–	19 (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–	Figure 11		4
	CLECC_0_0	**Clear ECC Status Register** transaction resets ECC Status Register bit[4] (2-bit ECC Detection), ECC Status Register bit[3] (1-bit ECC Correction), Address Trap Register, and ECC Detection Counter.	WRENB_0_0	1B (CMD)	–	–	–	–	–	–	–	–	Figure 5		N/A
CRC	DICHK_4_1	**Data Integrity Check** transaction causes the device to perform a Data Integrity Check over a user defined address range.	–	5B (CMD)	Start ADDR [31:24]	Start ADDR [23:16]	Start ADDR [15:8]	Start ADDR [7:0]	End ADDR [31:24]	End ADDR [23:16]	End ADDR [15:8]	End ADDR [7:0]	Figure 7		4

Table 130. SPI (1S-1S-1S) Transaction Table (Continued)

Function	Transaction Name	Description	Prerequisite Transaction	Byte 1 (Hex)	Byte 2 (Hex)	Byte 3 (Hex)	Byte 4 (Hex)	Byte 5 (Hex)	Byte 6 (Hex)	Byte 7 (Hex)	Byte 8 (Hex)	Byte 9 (Hex)	Transaction Format	Max Frequency (MHz)	Address Length
Read Flash Array	RDAY1_C_0	**Read** transaction reads out the memory contents at the given address. The maximum CK frequency for this transaction is 50 MHz frequency.	–	03 (CMD)	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–	–	Figure 12	50	3
			–		ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–			4
	RDAY1_4_0		–	13 (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–			
	RDAY2_C_0	**Read Fast** transaction reads out the memory contents starting at the given address.	–	0B (CMD)	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–	–	Figure 11	166	3
			–		ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–			4
Program Flash Array	PRPGE_4_1	**Program Page** programs 256B or 512B data to the memory array in one transaction.	WRENB_0_0	12 (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	Input Data 1 [7:0]	Input Data 2 [7:0]	(Continue)	–	Figure 8		4
Erase Flash Array	ER004_4_0	**Erase 4-KB Sector** transaction sets all the bits of a 4 KB sector to 1 (all bytes are FFh).	WRENB_0_0	21 (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–	Figure 6		
	ER256_4_0	**Erase 256-KB Sector** transaction sets all the bits of a 256 KB sector to 1 (all bytes are FFh).	WRENB_0_0	DC (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–			
	ERCHP_0_0	**Erase Chip** transaction sets all bits to 1 (all bytes are FFh) inside the entire flash memory array.	WRENB_0_0	60 or C7 (CMD)	–	–	–	–	–	–	–	–	Figure 5		N/A
	EVERS_4_0	**Evaluate Erase Status** transaction verifies that the last erase operation on the addressed sector was completed successfully.	–	D0 (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–	Figure 6		4
	SEERC_4_0	**Sector Erase Count** transaction outputs the number of erase cycles for the sector of the inputed address from the Sector Erase Count Register.	–	5D (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–			
Suspend / Resume	SPEPD_0_0	**Suspend Erase / Program / Data Integrity Check** transaction allows the system to interrupt a programming, erase or data integrity check operation.	–	B0 (CMD)	–	–	–	–	–	–	–	–	Figure 5		N/A
	RSEPD_0_0	**Resume Erase / Program / Data Integrity Check** transaction allows the system to resume a programming, erase or data integrity check operation.	–	30 (CMD)	–	–	–	–	–	–	–	–			
Secure Silicon Region	PRSSR_4_1	**Program SSR** transaction programs data in 1024 bytes of SSR.	WRENB_0_0	42 (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	Input Data 1 [7:0]	Input Data 2 [7:0]	(Continue)	–	Figure 8		4
	RDSSR_4_0	**Read SSR** transaction reads data from the SSR.	–	4B (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–	Figure 11		

Table 130. SPI (1S-1S-1S) Transaction Table (Continued)

Function	Transaction Name	Description	Prerequisite Transaction	Byte 1 (Hex)	Byte 2 (Hex)	Byte 3 (Hex)	Byte 4 (Hex)	Byte 5 (Hex)	Byte 6 (Hex)	Byte 7 (Hex)	Byte 8 (Hex)	Byte 9 (Hex)	Transaction Format	Max Frequency (MHz)	Address Length
Advanced Sector Protection	RDDYB_4_0	**Read Dynamic Protection Bit** transaction reads the contents of the DYB Access Register.	–	E0 (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–	Figure 11	166	4
	WRDYB_4_1	**Write Dynamic Protection Bit** transaction writes to the DYB Access Register.	WRENB_0_0	E1 (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	Input Data 1 [7:0]	Input Data 2 [7:0]	(Continue)	–	Figure 8		
	RDPPB_4_0	**Read Persistent Protection Bit** transaction reads the contents of the PPB Access Register.	–	E2 (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–	Figure 11		
	PRPPB_4_0	**Program Persistent Protection Bit** transaction programs / writes the PPB Register to enable the sector protection.	WRENB_0_0	E3 (CMD)	ADDR [31:24]	ADDR [23:16]	ADDR [15:8]	ADDR [7:0]	–	–	–	–	Figure 6		
	ERPPB_0_0	**Erase Persistent Protection Bit** transaction sets all persistent protection bits to 1.	WRENB_0_0	E4 (CMD)	–	–	–	–	–	–	–	–	Figure 5		N/A
	WRPLB_0_0	**Write PPB Protection Lock Bit** transaction clears the PPB Lock to 0.	–	A6 (CMD)	–	–	–	–	–	–	–	–			
	RDPLB_0_0	**Read Password Protection Mode Lock Bit** transaction shifts out the 8-bit PPB Lock Register contents with MSb first.	WRENB_0_0	A7 (CMD)	–	–	–	–	–	–	–	–	Figure 10		
	PWDUL_0_1	**Password Unlock** transaction sends the 64-bit password to flash device. If the supplied password does not match the hidden password in the Password Register, the device is locked and only a hardware reset or POR will return the device to standby state, ready for new transactions such as a retry of the PWDUL_0_1. If the password does match, the PPB Lock bit is set to 1.	–	E9 (CMD)	Password [7:0]	Password [15:8]	Password [23:16]	Password [31:24]	Password [39:32]	Password [47:40]	Password [55:48]	Password [63:56]	Figure 9		
Reset	SRSTE_0_0	**Software Reset Enable** command is required immediately before a SFRST_0_0 transaction.	–	66 (CMD)	–	–	–	–	–	–	–	–	Figure 5		
	SFRST_0_0	**Software Reset** transaction restores the device to its initial power up state, by reloading Volatile registers from nonvolatile default values.	SRSTE_0_0	99 (CMD)	–	–	–	–	–	–	–	–			
Deep Power Down	ENDPD_0_0	**Enter Deep Power Down Mode** transaction shifts device in the lowest power consumption mode.	–	B9 (CMD)	–	–	–	–	–	–	–	–			

7. Electrical Characteristics

7.1 Absolute Maximum Ratings[66]

Storage Temperature Plastic Packages..... –65 °C to +150 °C
Ambient Temperature with Power Applied..... –65 °C to +125 °C
V_{CC} (HL-T)..... –0.5 V to +4.0 V
V_{CC} (HS-T)..... –0.5 V to +2.5 V
Input voltage with respect to Ground (V_{SS})[64]..... –0.5 V to V_{CC} + 0.5 V
Output Short Circuit Current[65]..... 100 mA

Notes
64. See Section 7.6.1 Input Signal Overshoot on page 144 for allowed maximums during signal transition.
65. No more than one output may be shorted to ground at a time. Duration of the short circuit should not be greater than one second.
66. Stresses above those listed under Section 7.1 Absolute Maximum Ratings[66] on page 143 may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational sections of this datasheet is not implied. Exposure of the device to absolute maximum rating conditions for extended periods may affect device reliability.

7.2 Operating Range

Operating ranges define those limits between which the functionality of the device is guaranteed.

7.2.1 Power Supply Voltages

V_{CC} / V_{CCQ} (HL-T Devices)……..... 2.7 V to 3.6 V
V_{CC} / V_{CCQ} (HS-T Devices)……..... 1.7 V to 2.0 V

7.2.2 Temperature Ranges

Parameter	Symbol	Devices	Spec		Unit
			Min	Max	
Ambient Temperature	T_A	Industrial / Automotive AEC-Q100 Grade 3	–40	+85	°C
		Industrial Plus / Automotive AEC-Q100 Grade 2[67]		+105	
		Automotive AEC-Q100 Grade 1[67]		+125	

Note
67. Industrial Plus, Automotive Grade-2 and Automotive Grade-1 operating and performance parameters will be determined by device characterization and may vary from standard industrial or Automotive Grade-3 temperature range devices as currently shown in this specification.

7.3 Thermal Resistance

Table 131. Thermal Resistance

Parameter	Description	Test Condition		24-Ball BGA	Unit
Theta JA	Thermal Resistance (Junction to ambient)	Test conditions follow standard test methods and procedures for measuring thermal impedance in accordance with EIA/JESD51	With Still Air (0 m/s)	40.4 (512 Mb) / 34 (1 Gb)	°C/W

7.4 Capacitance Characteristics

Table 132. Capacitance

Symbol	Parameter	Test Conditions	Typ	Max	Unit
C_{IN}	Input Capacitance (applies to CK, CS#, RESET#)	1 MHz	3.0	7.50	pF
C_{OUT}	Output Capacitance (applies to All I/O)		6.50		

7.5 Latchup Characteristics

Table 133. Latchup Specification[68]

Description	Min	Max	Unit
Input voltage with respect to V_{SSQ} on all input only connections	–1.0	V_{CCQ} + 1.0	V
Input voltage with respect to V_{SSQ} on all I/O connections			
V_{CCQ} Current	–100	+100	mA

Note
68. Excludes power supply V_{CC}. Test conditions: V_{CC} = 1.8V / 3.0V, one connection at a time tested, connections not being tested are at V_{SS}.

7.6 DC Characteristics

7.6.1 Input Signal Overshoot

During DC conditions, input or I/O signals should remain equal to or between V_{SSQ} and V_{CCQ}. During voltage transitions, inputs or I/Os may overshoot V_{SSQ} to –1.0 V or overshoot to V_{CCQ} +1.0 V, for periods up to 20 ns.

Figure 50. Maximum Negative Overshoot Waveform

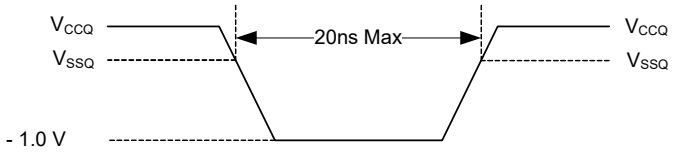
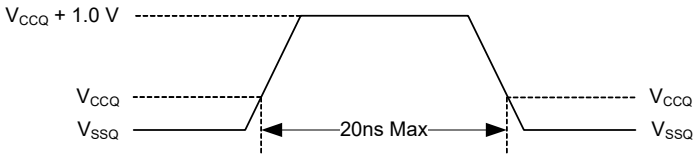


Figure 51. Maximum Positive Overshoot Waveform



7.6.2 DC Characteristics (All Temperature Ranges)

Table 134. DC Characteristics[69, 71]

Symbol	Parameter	Test Conditions	Min	Typ	Max	Unit	Reference Figure
V_{IL}	Input Low Voltage (all V_{CC})	–	V_{CCQ} × -0.15	–	V_{CCQ} × 0.35	V	–
V_{IH}	Input High Voltage (all V_{CC})	–	V_{CCQ} × 0.65	–	V_{CCQ} × 1.15		
V_{OL}	Output Low Voltage (all V_{CC})	At 0.1 mA		–	0.2		
V_{OH}	Output High Voltage (all V_{CC})	At –0.1 mA	V_{CCQ} - 0.20	–			
I_{LI}	Input Leakage Current	V_{CC} = V_{CC} Max, V_{IN} = V_{IH} or V_{SS}, CS# = V_{IH}, 85 °C	–	–	±2	µA	
		V_{CC} = V_{CC} Max, V_{IN} = V_{IH} or V_{SS}, CS# = V_{IH}, 105 °C	–	–	±3		
		V_{CC} = V_{CC} Max, V_{IN} = V_{IH} or V_{SS}, CS# = V_{IH}, 125 °C	–	–	±4		
I_{LO}	Output Leakage Current	V_{CC} = V_{CC} Max, V_{IN} = V_{IH} or V_{SS}, CS# = V_{IH}, 85 °C	–	–	±2		
		V_{CC} = V_{CC} Max, V_{IN} = V_{IH} or V_{SS}, CS# = V_{IH}, 105 °C	–	–	±3		
		V_{CC} = V_{CC} Max, V_{IN} = V_{IH} or V_{SS}, CS# = V_{IH}, 125 °C	–	–	±4		
I_{CC1}	Active Power Supply Current (READ)[70]	SDR @ 50 MHz (HL512T / HS512T / HL01GT / HS01GT) SDR @ 166 MHz DDR @ 200 MHz	–	10 75 156	21 / 18 / 18 / 18 100 173	mA	
I_{CC2}	Active Power Supply Current (Page Program)	V_{CC} = V_{CC} Max, CS# = V_{IH}	–	50	58		
I_{CC3}	Active Power Supply Current (Register Program)	V_{CC} = V_{CC} Max, CS# = V_{IH}	–	50	55		
I_{CC4}	Active Power Supply Current (Sector Erase)	V_{CC} = V_{CC} Max, CS# = V_{IH}	–	50	55		
I_{CC5}	Active Power Supply Current (Chip Erase)	V_{CC} = V_{CC} Max, CS# = V_{IH}	–	50	55		
I_{SB}	Standby Current (HS512T/HS01GT)	RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 85 °C	–	11 / 11	113 / 113	µA	
		RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 105 °C	–		188 / 113		
		RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 125 °C	–		340 / 113		
	Standby Current (HL512T/HL01GT)	RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 85 °C	–	14 / 14	126 / 126		
		RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 105 °C	–		188 / 126		
		RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 125 °C	–		340 / 126		
I_{DPD}	DPD Current (HS512T/HS01GT)	RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 85 °C	–	1.3 / 1.3	18 / 24		
		RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 105 °C	–		18 / 24		
		RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 125 °C	–		31 / 24		
	DPD Current (HL512T/HL01GT)	RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 85 °C	–	2.2 / 2.2	18 / 26		
		RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 105 °C	–		18 / 26		
		RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}, 125 °C	–		31 / 26		
I_{POR}	POR Current	RESET#, CS# = V_{CCQ}; All I/Os = V_{CCQ} or V_{SSQ}	–	–	80	mA	

Notes
69. Typical values are at T_{AI} = 25 °C and V_{CC} = 1.8 V/3.0 V.
70. Outputs unconnected during read data return. Output switching current is not included.
71. The recommended pull-up resistor for the INT# outputs is 5 kΩ to 10 kΩ.

Table 134. DC Characteristics[69, 71] (Continued)

Symbol	Parameter	Test Conditions	Min	Typ	Max	Unit	Reference Figure
Power Up / Power Down Voltage							
V_{CC} (min)	V_{CC} (minimum operation voltage, HL-T)	–	2.7	–	–	V	Figure 45 / Figure 46
	V_{CC} (minimum operation voltage, HS-T)	–	1.7	–	–		
V_{CC} (cut-off)	V_{CC} (cut off where re-initialization is needed, HL-T)	–	2.4	–	–		Figure 46
	V_{CC} (cut off where re-initialization is needed, HS-T)	–	1.55	–	–		
V_{CC} (Low)	V_{CC} (low voltage for initialization to occur, HL-T)	–	0.7	–	–		
	V_{CC} (Low voltage for initialization to occur, HS-T)	–	0.7	–	–		

Notes
69. Typical values are at T_{AI} = 25 °C and V_{CC} = 1.8 V/3.0 V.
70. Outputs unconnected during read data return. Output switching current is not included.
71. The recommended pull-up resistor for the INT# outputs is 5 kΩ to 10 kΩ.

7.7 AC Test Conditions

Figure 52. Test Setup

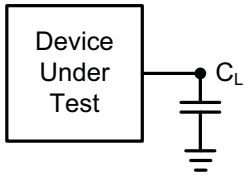


Table 135. AC Measurement Conditions[73]

Parameter	Min	Max	Unit	Reference Figure
Load Capacitance (C_L)	–	15	pF	Figure 52
Input Pulse Voltage	0	V_{CCQ}	V	Figure 54
CK Rise (t_{CRT1}) and Fall (t_{CFT1}) Slew Rates at 200 MHz (HS-T)[72]	1.13	–	V/ns	Figure 57
CK Rise (t_{CRT2}) and Fall (t_{CFT2}) Slew Rates at 166 MHz (HL-T)[72]	1.72	–		
Data Rise (t_{DRT1}) and Fall (t_{DFT1}) Slew Rates at 200 MHz (HS-T)[72]	1.13	–		Figure 54
Data Rise (t_{DRT2}) and Fall (t_{DFT2}) Slew Rates at 166 MHz (HL-T)[72]	1.72	–		
$V_{IL(ac)}$	$-0.30 \times V_{CCQ}$	$0.30 \times V_{CCQ}$	V	
$V_{IH(ac)}$	$0.7 \times V_{CCQ}$	$1.30 \times V_{CCQ}$		
$V_{OH(ac)}$	$0.75 \times V_{CCQ}$	–		Figure 55 / Figure 56
$V_{OL(ac)}$	–	$0.25 \times V_{CCQ}$		
Input Timing Ref Voltage	$V_{IH(ac)}$, $V_{IL(ac)}$			–
Output Timing Ref Voltage	$V_{OH(ac)}$, $V_{OL(ac)}$			

Notes
72. Input slew rate measured from input pulse min to max at V_{CC} max.
73. AC characteristics tables assume clock and data signals have the same slew rate (slope).

8. Timing Characteristics

Table 136. Timing Characteristics[76]

Symbol	Parameter	Min	Typ	Max	Unit	Reference Figure
HyperBus						
f_{CK}	CK Clock Frequency for HyperBus mode transactions using DS (HS-T)	0	–	200	MHz	–
	CK Clock Frequency for HyperBus mode transactions using DS (HL-T)		–	166		
p_{CK}	CK Clock Period	$1/f_{CK}$	–	∞	ns	Figure 54
t_{CH}	Clock High Time	45% p_{CK}	–	55% p_{CK}		Figure 57
t_{CL}	Clock Low Time		–			
t_{CS}	CS# High Time (Read transactions)	7.5	–	–		Figure 60
	CS# High Time (CRC ASO: including ASO entry and exit Configuration Register Load Program)	50	–	–		
t_{CSS}	CS# Active Setup Time (relative to CK)	4	–	–		
t_{CSH0}	CS# Active Hold Time (relative to CK in Mode 0)	4	–	–		
t_{SU}	HS-T Data Setup Time (all V_{CC})	0.5	–	–		
	HL-T Data Setup Time (all V_{CC})	0.6	–	–		
t_{HD}	HL-T Data Hold Time (all V_{CC})	0.6	–	–		
	HS-T Data Hold Time (all V_{CC})	0.5	–	–		
t_V[74]	Clock Low to Output Valid (15 pF Loading) (HS-T)	2	–	5.45		Figure 61
	Clock Low to Output Valid (15 pF Loading) (HL-T)		–	7.25		
t_{CKDS}	DS Valid (HS-T)	–	–	5.45		
	DS Valid (HL-T)	–	–	7.25		
t_{DSS}[82]	DS transition to Data Valid	–0.4	–	0.4		
t_{DSH}[82]	DS transition to Data Invalid	–0.4	–	0.4		
t_{DIS}[75]	CS# Inactive to Output Disable Time (HS-T)	–	–	6.00		Figure 61
	CS# Inactive to Output Disable Time (HL-T)	–	–	7.50		
t_{DSZ}	CS# Inactive to DS Disable Time (HS-T)	–	–	6.00		
	CS# Inactive to DS Disable Time (HL-T)	–	–	7.50		
t_{IO_SKEW}[82]	Data Skew (First Data Bit to Last Data Bit)	–	–	0.4		–
SPI SDR						
f_{CK}	CK Clock Frequency	0	–	166	MHz	–

Notes
74. Full V_{CC} range and CL = 15 pF.
75. Output HI-Z is defined as the point where data is no longer driven.
76. Applicable across all operating temperature options.
77. If Reset# is asserted during the end of t_{PU}, the device will remain in the reset state and t_{RH} will determine when CS# may go Low.
78. Sum of tRP and t_{RH} must be equal to or greater than t_{RPH}.
79. Typical program and erase times assume the following conditions: 25 °C, V_{CC} = 1.8 V and 3.0 V; 10,000 cycles; checkerboard data pattern.
80. The programming time for any OTP programming transaction is the same as t_{PP}. This includes PRSSR_4_1.
81. The programming time for the PRPPB_4_0 transaction is the same as t_{PP}. The erase time for ERPPB_0_0 transaction is the same as t_{SE}.
82. Values are guaranteed by characterization and not 100% tested in production.
83. Guaranteed by design.
84. The Joint Electron Device Engineering Council (JEDEC) standard JESD22-A117 defines the procedural requirements for performing valid endurance and retention tests based on a qualification specification. This methodology is intended to determine the ability of a flash device to sustain repeated data changes without failure (program/erase endurance) and to retain data for the expected life (data retention). Endurance and retention qualification specifications are specified in JESD47 or may be developed using knowledge-based methods as in JESD94.

Table 136. Timing Characteristics[76] (Continued)

Symbol	Parameter	Min	Typ	Max	Unit	Reference Figure
p_{CK}	CK Clock Period	1/ f_{CK}	–	∞	ns	Figure 54
t_{CH}	Clock High Time	45% p_{CK}	–	55% p_{CK}		Figure 57
t_{CL}	Clock Low Time	45% p_{CK}	–	55% p_{CK}		
t_{CS}	CS# High Time (Read transactions)	10	–	–		Figure 58 / Figure 59
	CS# High Time Between Transactions (aborted commands)	20	–	–		
	CS# High Time (Program / Erase transactions)	50	–	–		
t_{CSS}	CS# Active Setup Time relative to CK ($f_{CK} \leq 50$ MHz / $f_{CK} > 50$ MHz)	5/4	–	–		Figure 58
t_{CSH0}	CS# Active Hold Time (relative to CK in Mode 0)	4	–	–		
t_{CSH3}	CS# Active Hold Time (relative to CK in Mode 3)	6	–	–	ns	Figure 58
t_{SU}	Data Setup Time (all V_{CC}) ($f_{CK} \leq 50$ MHz / $f_{CK} > 50$ MHz)	5 / 2	–	–		
t_{HD}	Data Hold Time (all V_{CC}) ($f_{CK} \leq 50$ MHz / $f_{CK} > 50$ MHz)	5 / 2	–	–		
t_V	Clock Low to Output Valid (15 pF Loading, 3.0 V – 3.6 V, 30 Ω Output Impedance) (HL-T)	2	–	6.5	ns	Figure 59
	Clock Low to Output Valid (30 pF Loading) (HS-T)	2	–	8		
	Clock Low to Output Valid (30 pF Loading) (HL-T)	2	–	9		
	Clock Low to Output Valid (15 pF Loading) (HS-T)	2	–	6		
	Clock Low to Output Valid (15 pF Loading) (HL-T)	2	–	8		
t_{HO}	Output Hold Time	1.5	–	–	ns	
t_{DIS}	Output Disable Time (HS-T)	–	–	6	ns	
	Output Disable Time (HL-T)	–	–	7.50		
Power Up / Power Down Timing						
t_{PU}	V_{CC}(min) to Read operation (HL512T / HS512T / HL01GT / HS01GT)	–	–	450 / 500 / 400 / 450	µs	Figure 45
t_{PD}	V_{CC}(Low) time	25	–	–		Figure 46
t_{VR}[83]	V_{CC} / V_{CC}Q Power Up ramp rate	1	–	–	µs/V	Figure 47
t_{VF}	V_{CC} / V_{CC}Q Power Down ramp rate	30	–	–	µs/V	Figure 62
Deep Power Down Mode Timing						
t_{ENTDPD}[83]	Time to Enter DPD mode	–	–	3	µs	–
t_{EXTDPD}	Time to Exit DPD mode (HL512T / HS512T / HL01GT / HS01GT)	–	–	380 / 430 / 320 / 350		Figure 44
t_{CSDPD}	Chip Select Pulse Width to Exit DPD	0.02	–	3		
Reset Timing[77, 78]						
t_{RS}	Reset Setup - RESET# High before CS# Low	50	–	–	ns	Figure 37
t_{RH}	Reset Pulse Hold - RESET# Low to CS# Low (HL512T / HS512T / HL01GT / HS01GT)	450 / 500 / 400 / 450	–	–	µs	
t_{RP}	RESET# Pulse Width	200	–	–	ns	
t_{SR}	Internal Device Reset from Software Reset transaction (HL512T / HS512T / HL01GT / HS01GT)	–	–	83 / 83 / 60 / 60	µs	–
CS# Signaling Reset Timing						

Notes
74. Full V_{CC} range and CL = 15 pF.
75. Output HI-Z is defined as the point where data is no longer driven.
76. Applicable across all operating temperature options.
77. If Reset# is asserted during the end of t_{PU}, the device will remain in the reset state and t_{RH} will determine when CS# may go Low.
78. Sum of tRP and t_{RH} must be equal to or greater than t_{RPH}.
79. Typical program and erase times assume the following conditions: 25 °C, V_{CC} = 1.8 V and 3.0 V; 10,000 cycles; checkerboard data pattern.
80. The programming time for any OTP programming transaction is the same as t_{PP}. This includes PRSSR_4_1.
81. The programming time for the PRPPB_4_0 transaction is the same as t_{PP}. The erase time for ERPPB_0_0 transaction is the same as t_{SE}.
82. Values are guaranteed by characterization and not 100% tested in production.
83. Guaranteed by design.
84. The Joint Electron Device Engineering Council (JEDEC) standard JESD22-A117 defines the procedural requirements for performing valid endurance and retention tests based on a qualification specification. This methodology is intended to determine the ability of a flash device to sustain repeated data changes without failure (program/erase endurance) and to retain data for the expected life (data retention). Endurance and retention qualification specifications are specified in JESD47 or may be developed using knowledge-based methods as in JESD94.

Table 136. Timing Characteristics[76] (Continued)

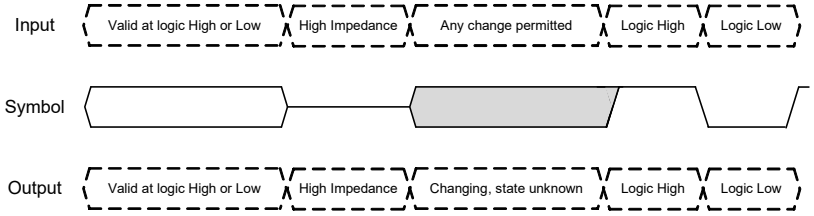
Symbol	Parameter	Min	Typ	Max	Unit	Reference Figure
t_{CSLW}	Chip Select Low	500	–	–	ns	Figure 42
t_{CSHG}	Chip Select High	500	–	–		
t_{RESET}	Internal device reset (HL512T / HS512T / HL01GT / HS01GT)	–	–	450 / 500 / 400 / 450	µs	
t_{SUJ}	Data in Setup Time (w.r.t CS#)	50	–	–	ns	
t_{HDJ}	Data in Hold Time (w.r.t CS#)	50	–	–		
Embedded Algorithm (Erase, Program and Data Integrity Check) Performance[79, 80, 81, 84]						
t_W	Nonvolatile Register Write Time	–	44	357.5	ms	–
t_{PP}	256B Page Programming (4 KB Sector / 256 KB Sector)	–	430 / 480	2175 / 1700	µs	–
	512B Page Programming (4 KB Sector / 256 KB Sector)	–	680 / 570	2175 / 1700		
t_{SE}	Sector Erase Time (4 KB physical sectors)	–	42	335	ms	
	Sector Erase Time (256 KB EnduraFlex Architecture disabled)	–	773	2677		
	Sector Erase Time (256 KB EnduraFlex Architecture enabled)	–	773	5869		
t_{BE}	Chip Erase Time (256 Mb)	–	101	348	sec	–
	Chip Erase Time (512 Mb)	–	201	696		
	Chip Erase Time (1 Gb)	–	398	1381		
t_{EES}	Evaluate Erase Status Time for 4 KB physical sectors (HL512T / HS512T / HL01GT / HS01GT)	–	45	51 / 51 / 50 / 50	µs	–
	Evaluate Erase Status Time for 256 KB physical sectors (HL512T / HS512T / HL01GT / HS01GT)	–	45	51 / 51 / 50 / 50		
t_{DIC_SETUP}	Data Integrity Check Calculation Setup Time	–	17	–	µs	–
t_{DIC_RATES}	Data Integrity Check Calculation Rate (Calculation rate over a large (>1024-byte) block of data) (HL512T / HS512T / HL01GT / HS01GT)	55 / 55 / 60 / 60	65	–	MBps	
t_{SEC}	Sector Erase Count Time (HL512T / HS512T / HL01GT / HS01GT)	–	55	63 / 63 / 60 / 60	µs	
t_{BEC1}	Blank Check single 256 KB sector	–	13	17	ms	
t_{BEC2}	Blank Check single 4 KB sector	–	1	2		
t_{PSWD}	Setting the PPB Lock bit after the valid 64-bit password is given to the device			100	µs	
Program, Erase or Data Integrity Check Suspend/Resume Timing						
t_{PEDS}	Program/Erase/Data Integrity Check Suspend (HL512T / HS512T / HL01GT / HS01GT)	–	–	64 / 64 / 60 / 60	µs	–
t_{PEDRS}	Program/Erase/Data Integrity Check Resume to next Program/ Erase/Data Integrity Check Suspend	–	100	–		

Notes
74. Full V_{CC} range and CL = 15 pF.
75. Output HI-Z is defined as the point where data is no longer driven.
76. Applicable across all operating temperature options.
77. If Reset# is asserted during the end of t_{PU}, the device will remain in the reset state and t_{RH} will determine when CS# may go Low.
78. Sum of tRP and t_{RH} must be equal to or greater than t_{RPH}.
79. Typical program and erase times assume the following conditions: 25 °C, V_{CC} = 1.8 V and 3.0 V; 10,000 cycles; checkerboard data pattern.
80. The programming time for any OTP programming transaction is the same as t_{PP}. This includes PRSSR_4_1.
81. The programming time for the PRPPB_4_0 transaction is the same as t_{PP}. The erase time for ERPPB_0_0 transaction is the same as t_{SE}.
82. Values are guaranteed by characterization and not 100% tested in production.
83. Guaranteed by design.
84. The Joint Electron Device Engineering Council (JEDEC) standard JESD22-A117 defines the procedural requirements for performing valid endurance and retention tests based on a qualification specification. This methodology is intended to determine the ability of a flash device to sustain repeated data changes without failure (program/erase endurance) and to retain data for the expected life (data retention). Endurance and retention qualification specifications are specified in JESD47 or may be developed using knowledge-based methods as in JESD94.

8.1 Timing Waveforms

8.1.1 Key to Timing Waveform

Figure 53. Waveform Element Meanings



8.1.2 Timing Reference Levels

Figure 54. Input Timing Reference Levels

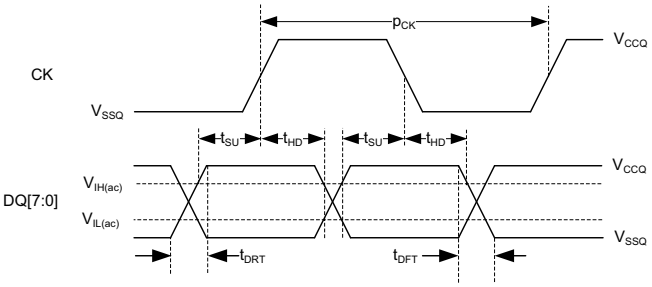


Figure 55. SDR Output Timing Reference Level

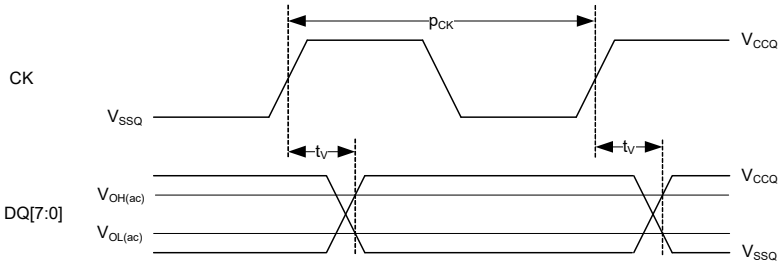
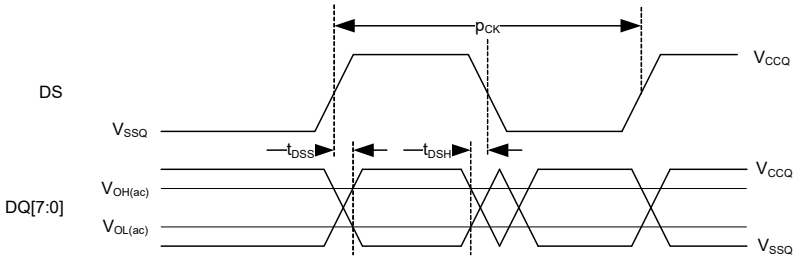
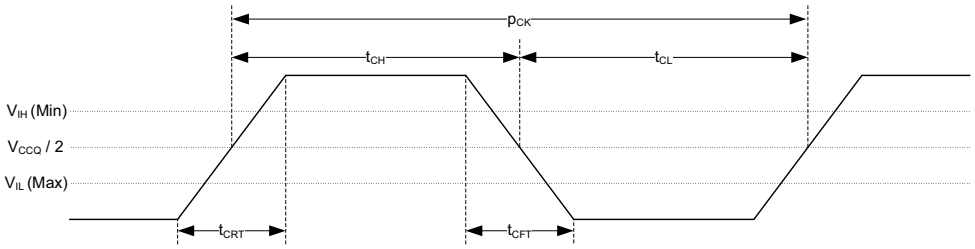


Figure 56. DDR Output Timing Reference Level



8.1.3 Clock Timing

Figure 57. Clock Timing



8.1.4 Input / Output Timing

Figure 58. SPI Input Timing

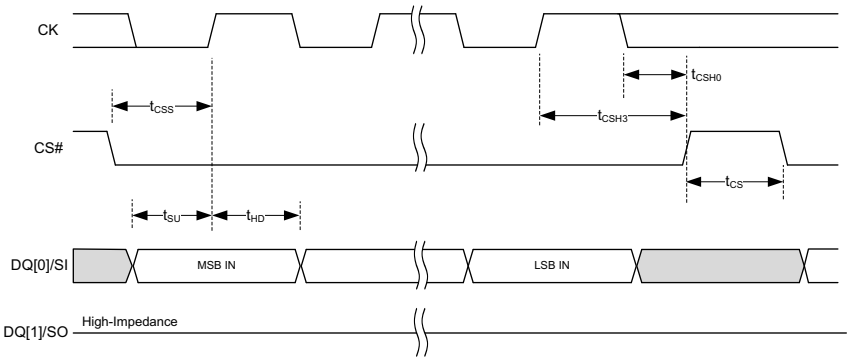


Figure 59. SPI Output Timing

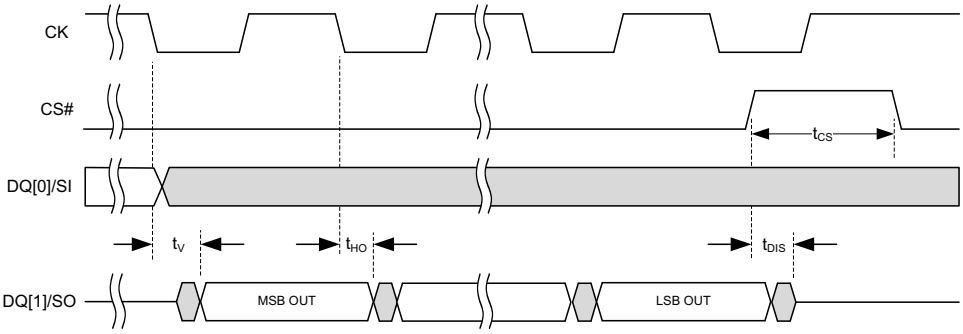


Figure 60. HyperBus DDR Input Timing

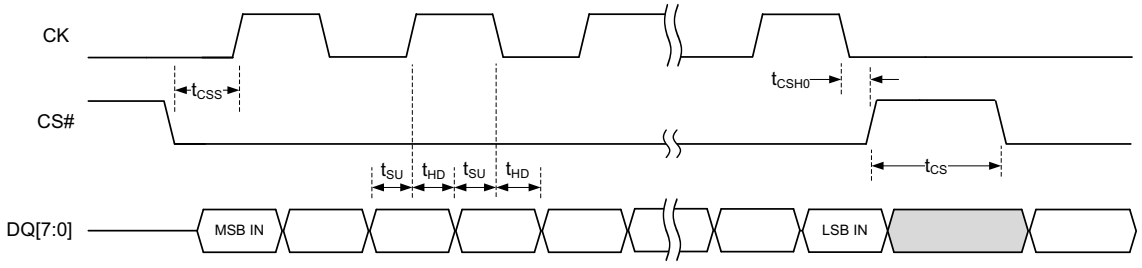
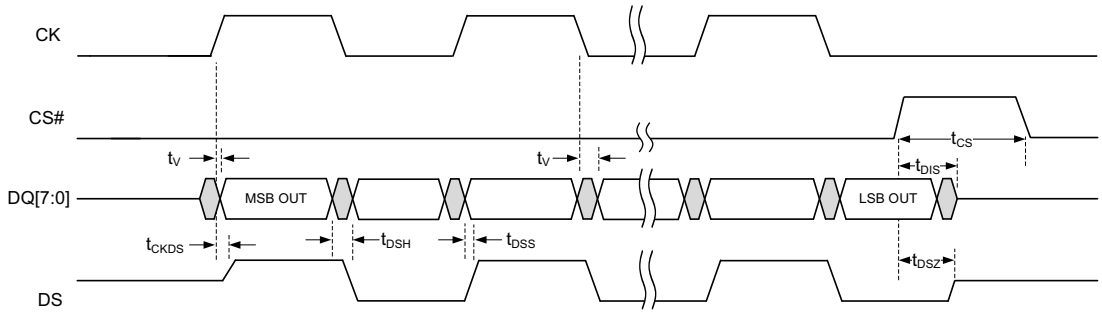
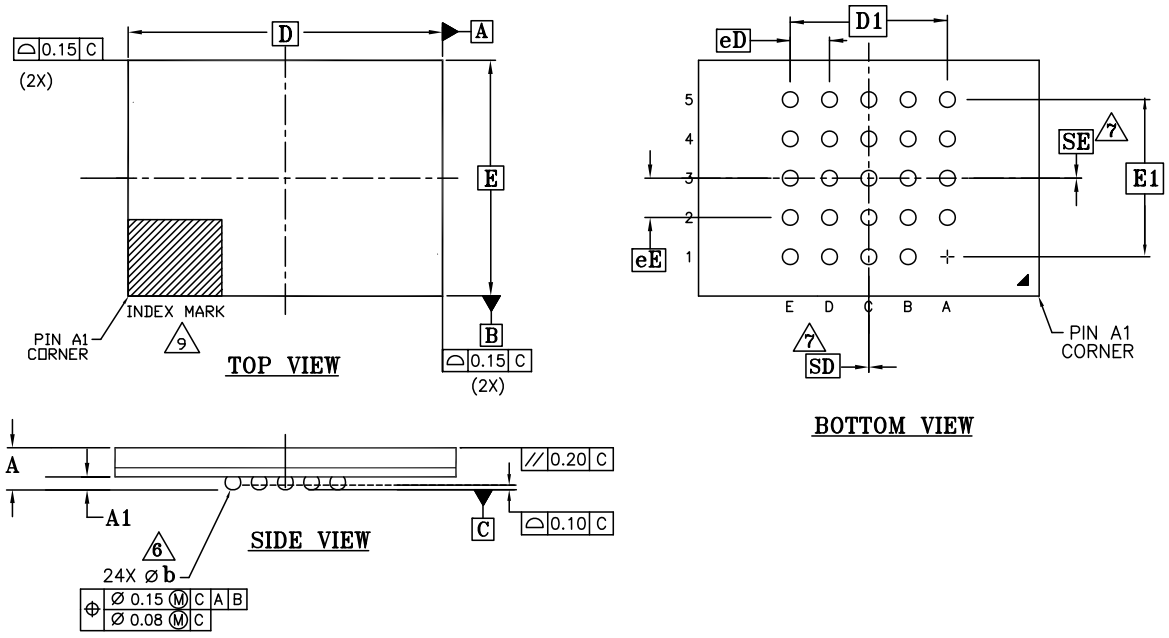


Figure 61. HyperBus DDR Output Timing



9. Package Diagrams

Figure 62. Ball Grid Array 24-ball 6 × 8 mm (VAA024)



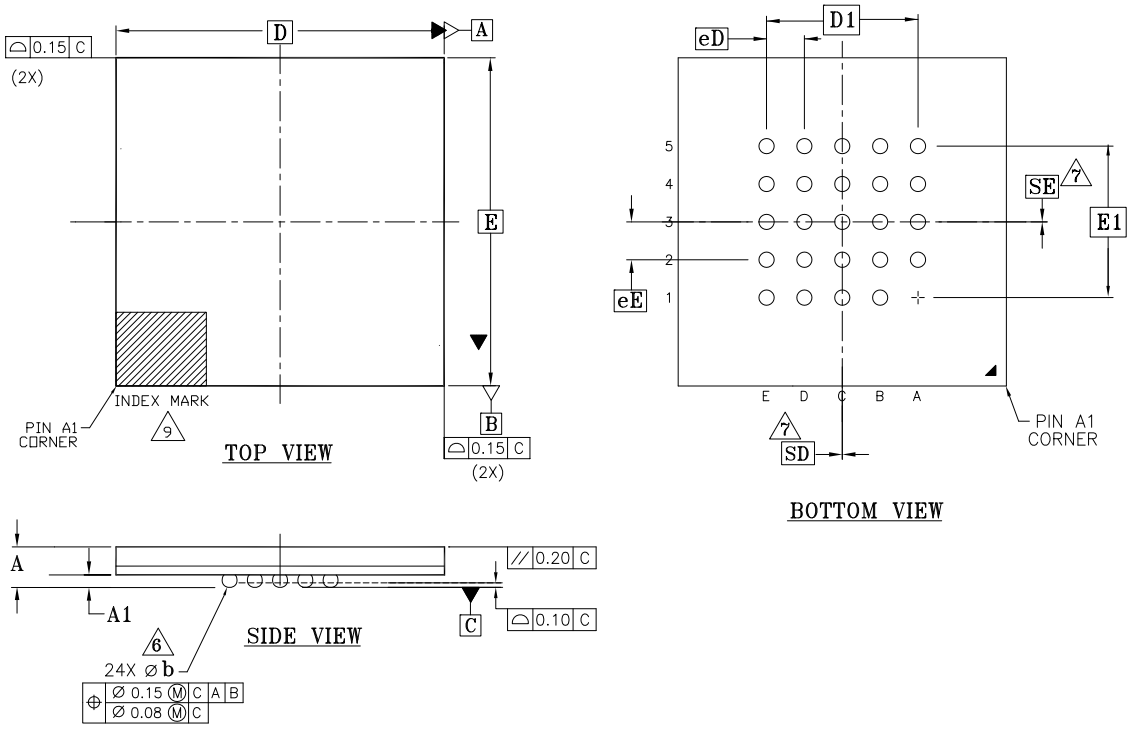
SYMBOL	DIMENSIONS		
	MIN.	NOM.	MAX.
A	-	-	1.00
A1	0.20	-	-
D	8.00 BSC		
E	6.00 BSC		
D1	4.00 BSC		
E1	4.00 BSC		
MD	5		
ME	5		
N	24		
∅ b	0.35	0.40	0.45
eE	1.00 BSC		
eD	1.00 BSC		
SD	0.00 BSC		
SE	0.00 BSC		

NOTES:

1. DIMENSIONING AND TOLERANCING METHODS PER ASME Y14.5M-1994.
2. ALL DIMENSIONS ARE IN MILLIMETERS.
3. BALL POSITION DESIGNATION PER JEP95, SECTION 3, SPP-020.
4. "e" REPRESENTS THE SOLDER BALL GRID PITCH.
5. SYMBOL "MD" IS THE BALL MATRIX SIZE IN THE "D" DIRECTION.
   SYMBOL "ME" IS THE BALL MATRIX SIZE IN THE "E" DIRECTION.
   N IS THE NUMBER OF POPULATED SOLDER BALL POSITIONS FOR MATRIX SIZE MD X ME.
6. DIMENSION "b" IS MEASURED AT THE MAXIMUM BALL DIAMETER IN A PLANE PARALLEL TO DATUM C.
7. "SD" AND "SE" ARE MEASURED WITH RESPECT TO DATUMS A AND B AND DEFINE THE POSITION OF THE CENTER SOLDER BALL IN THE OUTER ROW.
   WHEN THERE IS AN ODD NUMBER OF SOLDER BALLS IN THE OUTER ROW "SD" OR "SE" = 0.
   WHEN THERE IS AN EVEN NUMBER OF SOLDER BALLS IN THE OUTER ROW, "SD" = eD/2 AND "SE" = eE/2.
8. "+" INDICATES THE THEORETICAL CENTER OF DEPOPULATED BALLS.
9. A1 CORNER TO BE IDENTIFIED BY CHAMFER, LASER OR INK MARK, METALLIZED MARK INDENTATION OR OTHER MEANS.
10. JEDEC SPECIFICATION NO. REF: N/A

002-15550 *A

Figure 63. Ball Grid Array 24-ball 8 × 8 mm (VAC024)



SYMBOL	DIMENSIONS		
	MIN.	NOM.	MAX.
A	-	-	1.00
A1	0.20	-	-
D	8.00 BSC		
E	8.00 BSC		
D1	4.00 BSC		
E1	4.00 BSC		
MD	5		
ME	5		
N	24		
∅ b	0.35	0.40	0.45
eE	1.00 BSC		
eD	1.00 BSC		
SD	0.00 BSC		
SE	0.00 BSC		

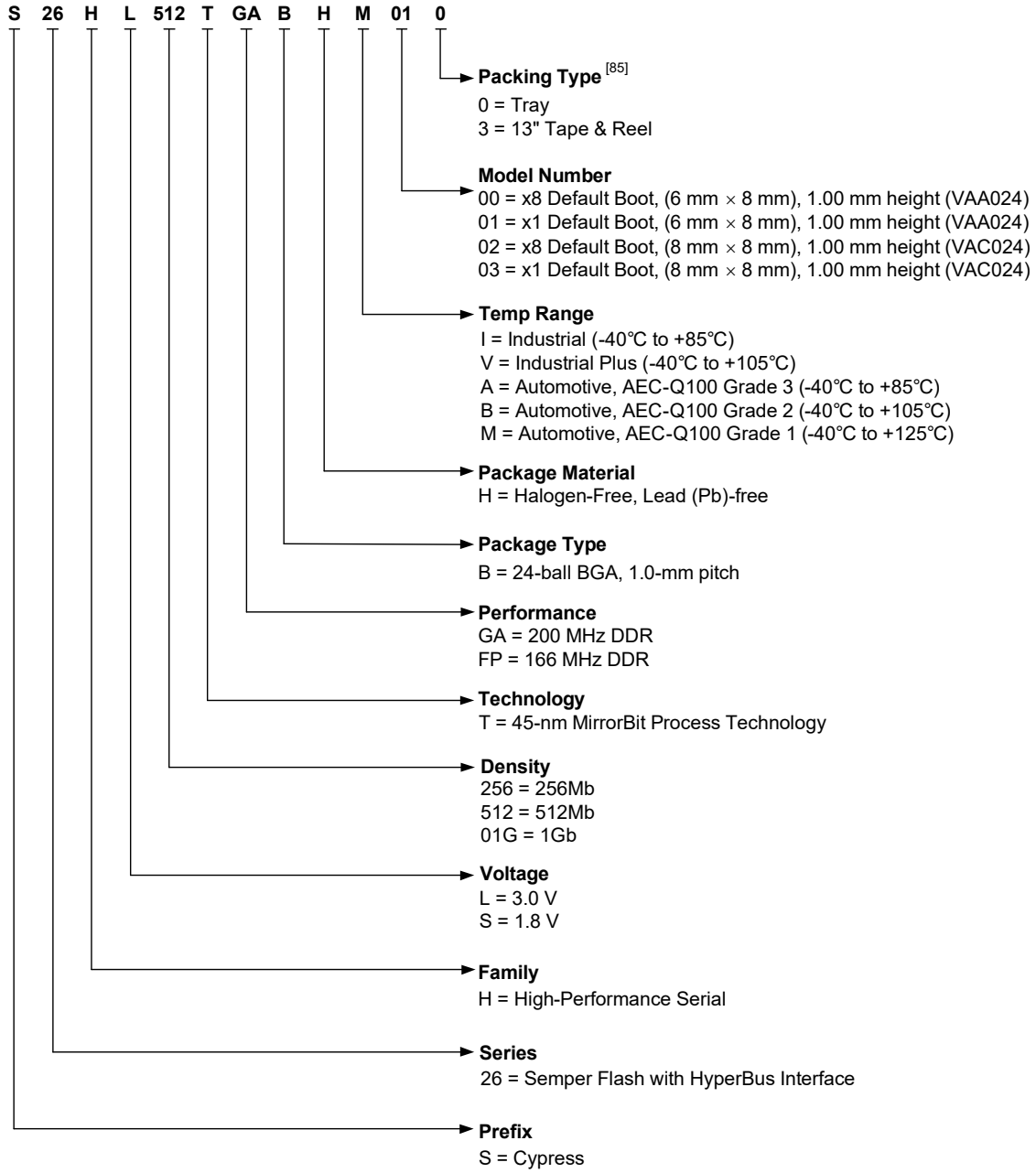
NOTES:

1. DIMENSIONING AND TOLERANCING METHODS PER ASME Y14.5M-1994.
2. ALL DIMENSIONS ARE IN MILLIMETERS.
3. BALL POSITION DESIGNATION PER JEP95, SECTION 3, SPP-020.
4. "e" REPRESENTS THE SOLDER BALL GRID PITCH.
5. SYMBOL "MD" IS THE BALL MATRIX SIZE IN THE "D" DIRECTION.
   SYMBOL "ME" IS THE BALL MATRIX SIZE IN THE "E" DIRECTION.
   N IS THE NUMBER OF POPULATED SOLDER BALL POSITIONS FOR MATRIX SIZE MD X ME.
6. DIMENSION "b" IS MEASURED AT THE MAXIMUM BALL DIAMETER IN A PLANE PARALLEL TO DATUM C.
7. "SD" AND "SE" ARE MEASURED WITH RESPECT TO DATUMS A AND B AND DEFINE THE POSITION OF THE CENTER SOLDER BALL IN THE OUTER ROW.
   WHEN THERE IS AN ODD NUMBER OF SOLDER BALLS IN THE OUTER ROW "SD" OR "SE" = 0.
   WHEN THERE IS AN EVEN NUMBER OF SOLDER BALLS IN THE OUTER ROW, "SD" = eD/2 AND "SE" = eE/2.
8. "+" INDICATES THE THEORETICAL CENTER OF DEPOPULATED BALLS.
9. A1 CORNER TO BE IDENTIFIED BY CHAMFER, LASER OR INK MARK, METALLIZED MARK INDENTATION OR OTHER MEANS.
10. JEDEC SPECIFICATION NO. REF: N/A

002-22282 **

10. Ordering Information

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



Note
85. See Packing and Packaging Handbook on www.cypress.com for further information.

10.1 Valid Combinations — Standard Grade

Valid Combinations list configurations planned to be supported in volume for this device. Contact your local sales office to confirm availability of specific valid combinations and to check on newly released combinations.

Table 137. Valid Combinations — Standard

Base Ordering Part Number	Speed Option	Package and Materials	Temperature Range	Model Number	Packing Type	Ordering Part Number (x = Packing Type)	Package Marking
S26HL512T	FP	BH	V	00	0, 3	S26HL512TFPBHV00x	26HL512TPV00
			V	01	0, 3	S26HL512TFPBHV01x	26HL512TPV01
S26HS512T	GA	BH	V	00	0, 3	S26HS512TGABHV00x	26HS512TAV00
			V	01	0, 3	S26HS512TGABHV01x	26HS512TAV01
S26HL01GT	FP	BH	I, V	02	0, 3	S26HL01GTFPBHI02x	26HL01GTPI02
						S26HL01GTFPBHV02x	26HL01GTPV02
			I, V	03	0, 3	S26HL01GTFPBHI03x	26HL01GTPI03
						S26HL01GTFPBHV03x	26HL01GTPV03
S26HS01GT	GA	BH	I, V	02	0, 3	S26HS01GTGABHI02x	26HS01GTAI02
						S26HS01GTGABHV02x	26HS01GTAV02
			I, V	03	0, 3	S26HS01GTGABHI03x	26HS01GTAI03
						S26HS01GTGABHV03x	26HS01GTAV03

Table 138. Valid Combinations — Standard Grade (In Production)

Base Ordering Part Number	Speed Option	Package and Materials	Temperature Range	Model Number	Packing Type	Ordering Part Number (x = Packing Type)	Package Marking
S26HL512T	FP	BH	I	00	0, 3	S26HL512TFPBHI00x	26HL512TPI00
			I	01	0, 3	S26HL512TFPBHI01x	26HL512TPI01
S26HS512T	GA	BH	I	00	0, 3	S26HS512TGABHI00x	26HS512TAI00
			I	01	0, 3	S26HS512TGABHI01x	26HS512TAI01

10.2 Valid Combinations — Automotive Grade / AEC-Q100

Table 139 lists configurations that are Automotive Grade / AEC-Q100 qualified and are planned to be available in volume. The table will be updated as new combinations are released. Contact your local sales representative to confirm availability of specific combinations and to check on newly released combinations.

Production Part Approval Process (PPAP) support is only provided for AEC-Q100 grade products.

Products to be used in end-use applications that require ISO/TS-16949 compliance must be AEC-Q100 grade products in combination with PPAP. Non–AEC-Q100 grade products are not manufactured or documented in full compliance with ISO/TS-16949 requirements.

AEC-Q100 grade products are also offered without PPAP support for end-use applications that do not require ISO/TS-16949 compliance.

Table 139. Valid Combinations — Automotive Grade / AEC-Q100

Base Ordering Part Number	Speed Option	Package and Materials	Temperature Range	Model Number	Packing Type	Ordering Part Number (x = Packing Type)	Package Marking
S26HL512T	FP	BH	A	00	0, 3	S26HL512TFPBHA00x	26HL512TPA00
			A, B	01	0, 3	S26HL512TFPBHA01x	26HL512TPA01
						S26HL512TFPBHB01x	26HL512TPB01
S26HS512T	GA	BH	A, B	00	0, 3	S26HS512TGABHA00x	26HS512TAA00
						S26HS512TGABHB00x	26HS512TAB00
			A, B	01	0, 3	S26HS512TGABHA01x	26HS512TAA01
						S26HS512TGABHB01x	26HS512TAB01
S26HL01GT	FP	BH	A, B, M	02	0, 3	S26HL01GTFPBHA02x	26HL01GTPA02
						S26HL01GTFPBHB02x	26HL01GTPB02
						S26HL01GTFPBHM02x	26HL01GTPM02
			A, B, M	03	0, 3	S26HL01GTFPBHA03x	26HL01GTPA03
						S26HL01GTFPBHB03x	26HL01GTPB03
						S26HL01GTFPBHM03x	26HL01GTPM03
S26HS01GT	GA	BH	A, B, M	02	0, 3	S26HS01GTGABHA02x	26HS01GTAA02
						S26HS01GTGABHB02x	26HS01GTAB02
						S26HS01GTGABHM02x	26HS01GTAM02
			A, B, M	03	0, 3	S26HS01GTGABHA03x	26HS01GTAA03
						S26HS01GTGABHB03x	26HS01GTAB03
						S26HS01GTGABHM03x	26HS01GTAM03

Table 140. Valid Combinations — Automotive Grade / AEC-Q100 (In Production)

Base Ordering Part Number	Speed Option	Package and Materials	Temperature Range	Model Number	Packing Type	Ordering Part Number (x = Packing Type)	Package Marking
S26HL512T	FP	BH	B, M	00	0, 3	S26HL512TFPBHB00x	26HL512TPB00
						S26HL512TFPBHM00x	26HL512TPM00
			M	01	0, 3	S26HL512TFPBHM01x	26HL512TPM01
S26HS512T	GA	BH	M	00	0, 3	S26HS512TGABHM00x	26HS512TAM00
			M	01	0, 3	S26HS512TGABHM01x	26HS512TAM01

11. Document History Page

Document Title: S26HS256T / S26HS512T / S26HS01GT / S26HL256T / S26HL512T / S26HL01GT, 256-Mb (32-MB), 512-Mb (64-MB), 1-Gb (128-MB) HS-T (1.8-V), HL-T (3.0-V) Semper™ Flash with HyperBus™ Interface Document Number: 002-12337			
Rev.	**ECN No.**	**Submission Date**	**Description of Change**
*P	6598395	07/03/2019	Finalizing document for S26HS512T and S26HL512T devices description.

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

Arm® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6 MCU

Cypress Developer Community

Community | Projects | Video | Blogs | Training | Components

Technical Support

cypress.com/support

© Cypress Semiconductor Corporation, 2016-2019. This document is the property of Cypress Semiconductor Corporation and its subsidiaries ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. No computing device can be absolutely secure. Therefore, despite security measures implemented in Cypress hardware or software products, Cypress shall have no liability arising out of any security breach, such as unauthorized access to or use of a Cypress product. CYPRESS DOES NOT REPRESENT, WARRANT, OR GUARANTEE THAT CYPRESS PRODUCTS, OR SYSTEMS CREATED USING CYPRESS PRODUCTS, WILL BE FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (collectively, "Security Breach"). Cypress disclaims any liability relating to any Security Breach, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from any Security Breach. In addition, the products described in these materials may contain design defects or errors known as errata which may cause the product to deviate from published specifications. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. "High-Risk Device" means any device or system whose failure could cause personal injury, death, or property damage. Examples of High-Risk Devices are weapons, nuclear installations, surgical implants, and other medical devices. "Critical Component" means any component of a High-Risk Device whose failure to perform can be reasonably expected to cause, directly or indirectly, the failure of the High-Risk Device, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from any use of a Cypress product as a Critical Component in a High-Risk Device. You shall indemnify and hold Cypress, its directors, officers, employees, agents, affiliates, distributors, and assigns harmless from and against all claims, costs, damages, and expenses, arising out of any claim, including claims for product liability, personal injury or death, or property damage arising from any use of a Cypress product as a Critical Component in a High-Risk Device. Cypress products are not intended or authorized for use as a Critical Component in any High-Risk Device except to the limited extent that (i) Cypress's published data sheet for the product explicitly states Cypress has qualified the product for use in a specific High-Risk Device, or (ii) Cypress has given you advance written authorization to use the product as a Critical Component in the specific High-Risk Device and you have signed a separate indemnification agreement.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.