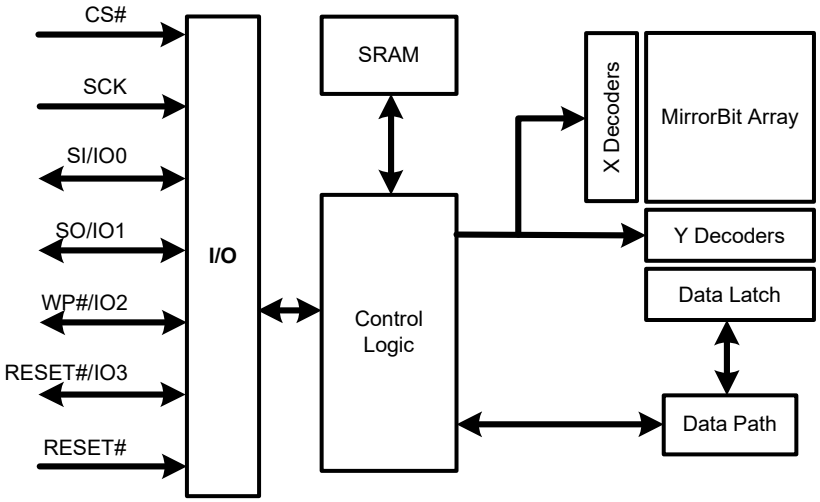
# S25FS064S

# 64 Mbit (8 Mbyte), 1.8 V FS-S Flash

## Features

- **Serial Peripheral Interface (SPI) with Multi-I/O**
  - SPI Clock polarity and phase modes 0 and 3
  - Double Data Rate (DDR) option
  - Extended Addressing: 24- or 32-bit address options
  - Serial Command subset and footprint compatible with S25FL1-K, S25FL-P and S25FL-S SPI families
  - Multi I/O Command subset and footprint compatible with S25FL1-K S25FL-P and S25FL-S SPI families
- **Read**
  - Commands: Normal, Fast, Dual Output, Dual I/O, Quad Output, Quad I/O, DDR Quad I/O
  - Modes: Burst Wrap, Continuous (XIP), QPI (QPI)
  - Serial Flash Discoverable Parameters (SFDP) and Common Flash Interface (CFI), for configuration information.
- **Program**
  - 256 or 512 Bytes Page Programming buffer
  - Program suspend and resume
  - Automatic ECC -internal hardware Error Correction Code generation with single bit error correction
- **Erase**
  - Hybrid sector option
    - Physical set of eight 4KB sectors and one 32KB sector at the top or bottom of address space with all remaining sectors of 64KB
  - Uniform sector option
    - Uniform 64KB or 256KB blocks for software compatibility with higher density and future devices
  - Erase suspend and resume
  - Erase status evaluation
- Cycling Endurance
  - 100,000 Program-Erase Cycles, minimum
- Data Retention
  - 20 Year Data Retention, minimum
- **Security Features**
  - One Time Program (OTP) array of 1024 bytes
  - Block Protection:
    - Status Register bits to control protection against program or erase of a contiguous range of sectors.
    - Hardware and software control options
  - Advanced Sector Protection (ASP)
    - Individual sector protection controlled by boot code or password
    - Option for password control of read access
- **Technology**
  - Cypress 65 nm MirrorBit® Technology with Eclipse™ Architecture
- **Single Supply Voltage with CMOS I/O**
  - 1.7 V to 2.0 V
- **Temperature Range**
  - Industrial (–40 °C to +85 °C)
  - Industrial Plus (–40 °C to +105 °C)
  - Extended (–40 °C to +125 °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 (all Pb-free)**
  - 8-lead SOIC 208 mil (SOC008)
  - LGA 5x6 mm (W9A008)
  - BGA-24 6 × 8 mm
    - 5 × 5 ball (FAB024) footprint

## Logic Block Diagram



# Performance Summary

### Maximum Read Rates

| Command | Clock Rate (MHz) | MB/s |
|---|---|---|
| Read | 50 | 6.25 |
| Fast Read | 133 | 16.5 |
| Dual Read | 133 | 33 |
| Quad Read | 133 | 66 |
| DDR Quad I/O Read | 80 | 80 |

### Typical Program and Erase Rates

| Operation | KB/s |
|---|---|
| Page Programming (256 Bytes page buffer) | 712 |
| Page Programming (512 Bytes page buffer) | 1080 |
| 4 KBytes Physical Sector Erase (Hybrid Sector Option) | 16 |
| 64 KBytes Sector Erase | 275 |
| 256 KBytes Sector Erase | 275 |

### Typical Current Consumption, −40°C to +85°C

| Operation | Current (mA) |
|---|---|
| Serial Read 50 MHz | 10 |
| Serial Read 133 MHz | 22 |
| Quad Read 133 MHz | 60 |
| Quad DDR Read 80 MHz | 70 |
| Program | 60 |
| Erase | 60 |
| Standby | 0.025 |
| Deep Power Down | 0.006 |

# Contents

**Features** ... 1

**Logic Block Diagram** ... 2

**Performance Summary** ... 3

**1. Overview** ... 5
1.1 General Description ... 5
1.2 Migration Notes ... 5
1.3 Other Resources ... 7

**Hardware Interface**

**2. Serial Peripheral Interface with Multiple Input / Output (SPI-MIO)** ... 8

**3. Signal Descriptions** ... 8
3.1 Input/Output Summary ... 8
3.2 Multiple Input / Output (MIO) ... 9
3.3 Serial Clock (SCK) ... 9
3.4 Chip Select (CS#) ... 9
3.5 Serial Input (SI) / IO0 ... 9
3.6 Serial Output (SO) / IO1 ... 9
3.7 Write Protect (WP#) / IO2 ... 9
3.8 IO3_RESET# ... 10
3.9 RESET# ... 10
3.10 Voltage Supply (VCC) ... 10
3.11 Supply and Signal Ground ($V_{SS}$) ... 10
3.12 Not Connected (NC) ... 10
3.13 Reserved for Future Use (RFU) ... 10
3.14 Do Not Use (DNU) ... 11
3.15 System Block Diagrams ... 11

**4. Signal Protocols** ... 13
4.1 SPI Clock Modes ... 13
4.2 Command Protocol ... 14
4.3 Interface States ... 18
4.4 Configuration Register Effects on the Interface ... 22
4.5 Data Protection ... 22

**5. Electrical Specifications** ... 23
5.1 Absolute Maximum Ratings ... 23
5.2 Latchup Characteristics ... 23
5.3 Thermal Resistance ... 23
5.4 Operating Ranges ... 23
5.5 Power-Up and Power-Down ... 24
5.6 DC Characteristics ... 26

**6. Timing Specifications** ... 30
6.1 Key to Switching Waveforms ... 30
6.2 AC Test Conditions ... 30
6.3 Reset ... 31
6.4 SDR AC Characteristics ... 34
6.5 DDR AC Characteristics. ... 37

**7. Embedded Algorithm Performance Tables** ... 39

**8. Physical Interface** ... 40
8.1 Connection Diagrams ... 40
8.2 Physical Diagrams ... 41

**Software Interface**

**9. Address Space Maps** ... 44
9.1 Overview ... 44
9.2 Flash Memory Array ... 44
9.3 ID-CFI Address Space ... 46
9.4 JEDEC JESD216 Serial Flash Discoverable Parameters (SFDP) Space ... 46
9.5 OTP Address Space ... 47
9.6 Registers ... 48

**10. Data Protection** ... 64
10.1 Secure Silicon Region (OTP) ... 64
10.2 Write Enable Command ... 64
10.3 Block Protection ... 65
10.4 Advanced Sector Protection ... 66
10.5 Recommended Protection Process ... 71

**11. Commands** ... 72
11.1 Command Set Summary ... 73
11.2 Identification Commands ... 79
11.3 Register Access Commands ... 82
11.4 Read Memory Array Commands ... 93
11.5 Program Flash Array Commands ... 101
11.6 Erase Flash Array Commands ... 104
11.7 One Time Program Array Commands ... 111
11.8 Advanced Sector Protection Commands ... 111
11.9 Reset Commands ... 118
11.10DPD Commands ... 119

**12. Data Integrity** ... 121
12.1 Erase Endurance ... 121
12.2 Data Retention ... 121

**13. Software Interface Reference** ... 122
13.1 OTP Memory Space Address Map ... 122
13.2 Device ID and Common Flash Interface (ID-CFI) Address Map — Standard ... 122
13.3 Serial Flash Discoverable Parameters (SFDP) Address Map ... 128
13.4 Initial Delivery State ... 140

**14. Ordering Part Number** ... 141

**15. Contacting Cypress** ... 142

**16. Glossary** ... 143

**16. Document History Page** ... 144

**Sales, Solutions, and Legal Information ... 146**
Worldwide Sales and Design Support ... 146
Products ... 146
PSoC® Solutions ... 146
Cypress Developer Community ... 146
Technical Support ... 146

# 1. Overview

## 1.1 General Description

The Cypress FS-S Family of devices are Flash non-volatile memory products using:

- MirrorBit technology - that stores two data bits in each memory array transistor
- Eclipse architecture - that dramatically improves program and erase performance
- 65 nm process lithography

The FS-S Family connects to a host system via a Serial Peripheral Interface (SPI). Traditional SPI single bit serial input and output (Single I/O or SIO) is supported as well as optional two bit (Dual I/O or DIO) and four bit wide Quad I/O (QIO) and Quad Peripheral Interface (QPI) commands. In addition, there are Double Data Rate (DDR) read commands for QIO and QPI that transfer address and read data on both edges of the clock.

The FS-S Eclipse architecture features a Page Programming Buffer that allows up to 512 bytes to be programmed in one operation, resulting in faster effective programming and erase than prior generation SPI program or erase algorithms.

Executing code directly from Flash memory is often called Execute-In-Place or XIP. By using FS-S Family devices at the higher clock rates supported, with Quad or DDR-Quad commands, the instruction read transfer rate can match or exceed traditional parallel interface, asynchronous, NOR Flash memories, while reducing signal count dramatically.

The FS-S Family products offer high densities coupled with the flexibility and fast performance required by a variety of mobile or embedded applications. They are an excellent solution for systems with limited space, signal connections, and power. They are ideal for code shadowing to RAM, executing code directly (XIP), and storing reprogrammable data.

## 1.2 Migration Notes

### 1.2.1 Features Comparison

The FS-S Family is command subset and footprint compatible with prior generation FL-S, and FL-P families. However, the power supply and interface voltages are nominal 1.8V.

**Table 1. Cypress SPI Families Comparison**

| Parameter | FS-S | FS-S | FL-S | FL-P |
|---|---|---|---|---|
| Technology Node | 65nm | 65nm | 65nm | 90nm |
| Architecture | MirrorBit® Eclipse™ | MirrorBit® Eclipse™ | MirrorBit® Eclipse™ | MirrorBit® |
| Release Date | In Production | 2H2015 | In Production | In Production |
| Density | 128Mb, 256Mb 512MB | 64Mb | 128Mb 256Mb 512Mb | 32Mb - 256Mb |
| Bus Width | x1, x2, x4 | x1, x2, x4 | x1, x2, x4 | x1, x2, x4 |
| Supply Voltage | 1.7V - 2.0V | 1.7V - 2.0V | 2.7V - 3.6V / 1.65V - 3.6V $V_{IO}$ | 2.7V - 3.6V |
| Normal Read Speed (SDR) | 6MB/s (50MHz) | 6MB/s (50MHz) | 6MB/s (50MHz) | 5MB/s (40MHz) |
| Fast Read Speed (SDR) | 16.5MB/s (133MHz) | 16.5MB/s (133MHz) | 16.5MB/s (133MHz) | 13MB/s (104MHz) |
| Dual Read Speed (SDR) | 33MB/s (133MHz) | 33MB/s (133MHz) | 26MB/s (104MHz) | 20MB/s (80MHz) |
| Quad Read Speed (SDR) | 66MB/s (133MHz) | 66MB/s (133MHz) | 52MB/s (104MHz) | 40MB/s (80MHz) |
| Quad Read Speed (DDR) | 80MB/s (80 MHz) | 80Mb/s(80Mhz ) | 66MB/s (66MHz) | - |
| Program Buffer Size | 256B / 512B | 256B / 512B | 256B / 512B | 256B |
| Erase Sector Size | 64KB / 256KB | 64KB / 256KB | 64KB / 256KB | 64KB / 256KB |
| Parameter Sector Size | 4KB (option) | 4KB (option) | 4KB (option) | 4KB |
| Sector Erase Rate (typ.) | 500 KB/s | 500 KB/s | 500 KB/s | 130 KB/s |
| Page Programming Rate (typ.) | 1.0 MB/s (256B)<br>1.2 MB/s (512B) | 1.0 MB/s (256B)<br>1.2 MB/s (512B) | 1.2 MB/s (256B)<br>1.5 MB/s (512B) | 170 KB/s |
| OTP | 1024B | 1024B | 1024B | 506B |
| Advanced Sector Protection | Yes | Yes | Yes | No |
| Auto Boot Mode | No | No | Yes | No |
| Erase Suspend/Resume | Yes | Yes | Yes | No |

**Table 1. Cypress SPI Families Comparison** (Continued)

| Parameter | FS-S | FS-S | FL-S | FL-P |
|---|---|---|---|---|
| Program Suspend/Resume | Yes | Yes | Yes | No |
| Operating Temperature | –40 °C to +85 °C / +105 °C | –40 °C to +85 °C / +105 °C / +125° C | –40 °C to +85 °C / +105 °C | –40 °C to +85 °C/+105 °C |

**Notes**
1. FL-P column indicates FL129P MIO SPI device (for 128Mb density), FL128P does not support MIO, OTP, or 4KB sectors.
2. 64KB sector erase option only for 128Mb/256Mb density FL-P, FL-S and FS-S devices.
3. Refer to individual datasheets for further details.

### 1.2.2 Known Differences from Prior Generations

#### 1.2.2.1 Error Reporting

FL-K and FL-P memories either do not have error status bits or do not set them if program or erase is attempted on a protected sector. The FS-S and FL-S families do have error reporting status bits for program and erase operations. These can be set when there is an internal failure to program or erase, or when there is an attempt to program or erase a protected sector. In these cases the program or erase operation did not complete as requested by the command. The P_ERR or E_ERR bits and the WIP bit will be set to and remain 1 in SR1V. The clear status register command must be sent to clear the errors and return the device to standby state.

#### 1.2.2.2 Secure Silicon Region (OTP)

The FS-S size and format (address map) of the One Time Program area is different from FL-K and FL-P generations. The method for protecting each portion of the OTP area is different. For additional details see Secure Silicon Region (OTP) on page 64.

#### 1.2.2.3 Configuration Register Freeze Bit

The configuration register-1 Freeze Bit CR1V[0], locks the state of the Block Protection bits (SR1NV[4:2] and SR1V[4:2]), TBPARM_O bit (CR1NV[2]), and TBPROT_O bit (CR1NV[5]), as in prior generations. In the FS-S and FL-S families the Freeze Bit also locks the state of the configuration register-1 BPNV_O bit (CR1NV[3]), and the Secure Silicon Region (OTP) area.

#### 1.2.2.4 Sector Erase Commands

The command for erasing a 4KBytes sector is supported only for use on 4KBytes parameter sectors at the top or bottom of the FS-S device address space.

The command for erasing an 8KByte area (two 4KBytes sectors) is not supported.

The command for erasing a 32KByte area (eight 4KBytes sectors) is not supported.

The sector erase command (SE) for FS-S 64KBytes sectors is supported when the configuration option for uniform 64KBytes sector is selected or, when the hybrid configuration option for 4KBytes parameter sectors with 64KBytes uniform sectors is used. When the hybrid option is in use, the 64KBytes erase command may be used to erase the 32KBytes of address space adjacent to the group of eight 4KBytes sectors. The 64KBytes erase command in this case is erasing the 64KBytes sector that is partially overlaid by the group of eight 4KBytes sectors without affecting the 4KBytes sectors. This provides erase control over the 32KBytes of address space without also forcing the erase of the 4KBytes sectors. This is different behavior than implemented in the FL-S family. In the FL-S family, the 64KBytes sector erase command can be applied to a 64KBytes block of 4KBytes sectors to erase the entire block of parameter sectors in a single operation. In the FS-S, the parameter sectors do not fill an entire 64KBytes block so only the 4KBytes parameter sector erase (20h) is used to erase parameter sectors.

The erase command for a 256KBytes sector replaces the 64KBytes erase command when the configuration option for 256 KBytes uniform logical sectors is used.

#### 1.2.2.5 Deep Power Down

A Deep Power Down (DPD) function is supported in the FS-S family devices.

#### 1.2.2.6 WRR Single Register Write

In some legacy SPI devices, a Write Registers (WRR) command with only one data byte would update Status Register 1 and clear some bits in Configuration Register 1, including the Quad mode bit. This could result in unintended exit from Quad mode. The FS-S Family only updates Status Register 1 when a single data byte is provided. The Configuration Register 1 is not modified in this case.

#### 1.2.2.7 Hold Input Not Supported

In some legacy SPI devices, the IO3 input has an alternate function as a HOLD# input used to pause information transfer without stopping the serial clock. This function is not supported in the FS-S family.

#### 1.2.2.8 Other Legacy Commands Not Supported

- DDR Fast Read
- DDR Dual I/O Read

#### 1.2.2.9 New Features

The FS-S family introduces new features to Cypress SPI category memories:

- Single 1.8V power supply for core and I/O voltage.
- Configurable initial read latency (number of dummy cycles) for faster initial access time or higher clock rate read commands
- QPI (QPI, 4-4-4) read mode in which all transfers are 4 bits wide, including instructions
- JEDEC JESD216 Rev B standard, Serial Flash Discoverable Parameters (SFDP) that provide device feature and configuration information.
- Evaluate Erase Status command to determine if the last erase operation on a sector completed successfully. This command can be used to detect incomplete erase due to power loss or other causes. This command can be helpful to Flash File System software in file system recovery after a power loss.
- Advanced Sector Protection (ASP) Permanent Protection. Also, when one of the two ASP protection modes is selected, all OTP configuration bits in all registers are protected from further programming so that all OTP configuration settings are made permanent. The OTP address space is not protected by the selection of an ASP protection mode. The Freeze bit (CR1V[0]) may be used to protect the OTP Address Space.

## 1.3 Other Resources

### 1.3.1 Link to Cypress Flash Roadmap

www.cypress.com/Flash-Roadmap

### 1.3.2 Link to Software

www.cypress.com/software-and-drivers-cypress-flash-memory

### 1.3.3 Link to Application Notes

www.cypress.com/cypressappnotes

# Hardware Interface

## 2. Serial Peripheral Interface with Multiple Input / Output (SPI-MIO)

Many memory devices connect to their host system with separate parallel control, address, and data signals that require a large number of signal connections and larger package size. The large number of connections increase power consumption due to so many signals switching and the larger package increases cost.

The FS-S Family reduces the number of signals for connection to the host system by serially transferring all control, address, and data information over 6 signals. This reduces the cost of the memory package, reduces signal switching power, and either reduces the host connection count or frees host connectors for use in providing other features.

The FS-S Family uses the industry standard single bit Serial Peripheral Interface (SPI) and also supports optional extension commands for two bit (Dual) and four bit (Quad) wide serial transfers. This multiple width interface is called SPI Multi-I/O or SPI-MIO.

## 3. Signal Descriptions

### 3.1 Input/Output Summary

**Table 2. Signal List**

| Signal Name | Type | Description |
|---|---|---|
| RESET# | Input | **Hardware Reset:** Low = device resets and returns to standby state, ready to receive a command. The signal has an internal pull-up resistor and may be left unconnected in the host system if not used. |
| SCK | Input | **Serial Clock** |
| CS# | Input | **Chip Select** |
| SI / IO0 | I/O | **Serial Input** for single bit data commands or IO0 for Dual or Quad commands. |
| SO / IO1 | I/O | **Serial Output** for single bit data commands. IO1 for Dual or Quad commands. |
| WP# / IO2 | I/O | **Write Protect** when not in Quad mode (CR1V[1] = 0 and SR1NV[7] = 1).<br>**IO2** when in Quad mode (CR1V[1] = 1).<br>The signal has an internal pull-up resistor and may be left unconnected in the host system if not used for Quad commands or write protection. If write protection is enabled by SR1NV[7] = 1 and CR1V[1] = 0, the host system is required to drive WP# high or low during a WRR or WRAR command. |
| IO3_RESET# | I/O | **IO3** in Quad-I/O mode, when Configuration Register-1 QUAD bit, CR1V[1] =1, and CS# is low.<br>**RESET#** when enabled by CR2V[5]=1 and not in Quad-I/O mode, CR1V[1] = 0, or when enabled in quad mode, CR1V[1] = 1 and CS# is high.<br>The signal has an internal pull-up resistor and may be left unconnected in the host system if not used for Quad commands or RESET#. |
| $V_{CC}$ | Supply | **Power Supply.** |
| $V_{SS}$ | Supply | **Ground.** |
| NC | Unused | **Not Connected.** No device internal signal is connected to the package connector nor is there any future plan to use the connector for a signal. The connection may safely be used for routing space for a signal on a Printed Circuit Board (PCB). However, any signal connected to an NC must not have voltage levels higher than $V_{CC}$. |
| RFU | Reserved | **Reserved for Future Use.** No device internal signal is currently connected to the package connector but there is potential future use of the connector for a signal. It is recommended to not use RFU connectors for PCB routing channels so that the PCB may take advantage of future enhanced features in compatible footprint devices. |
| DNU | Reserved | **Do Not Use.** A device internal signal may be connected to the package connector. The connection may be used by Cypress for test or other purposes and is not intended for connection to any host system signal. Any DNU signal related function will be inactive when the signal is at $V_{IL}$. The signal has an internal pull-down resistor and may be left unconnected in the host system or may be tied to $V_{SS}$. Do not use these connections for PCB signal routing channels. Do not connect any host system signal to this connection. |

**Note**

4. Inputs with internal pull-ups or pull-downs internally drive less than 2uA. Only during power-up is the current larger at 150uA for 4uS.

## 3.2 Multiple Input / Output (MIO)

Traditional SPI single bit wide commands (Single or SIO) send information from the host to the memory only on the Serial Input (SI) signal. Data may be sent back to the host serially on the Serial Output (SO) signal.

Dual or Quad Input / Output (I/O) commands send instructions to the memory only on the SI/IO0 signal. Address or data is sent from the host to the memory as bit pairs on IO0 and IO1 or four bit (nibble) groups on IO0, IO1, IO2, and IO3. Data is returned to the host similarly as bit pairs on IO0 and IO1 or four bit (nibble) groups on IO0, IO1, IO2, and IO3.

QPI mode transfers all instructions, address, and data from the host to the memory as four bit (nibble) groups on IO0, IO1, IO2, and IO3. Data is returned to the host similarly as four bit (nibble) groups on IO0, IO1, IO2, and IO3.

## 3.3 Serial Clock (SCK)

This input signal provides the synchronization reference for the SPI interface. Instructions, addresses, or data input are latched on the rising edge of the SCK signal. Data output changes after the falling edge of SCK, in SDR commands, and after every edge in DDR commands.

## 3.4 Chip Select (CS#)

The chip select signal indicates when a command is transferring information to or from the device and the other signals are relevant for the memory device.

When the CS# signal is at the logic high state, the device is not selected and all input signals except the Reset# and IO3_Reset# are ignored and all output signals are high impedance. The device will be in the Standby Power mode, unless an internal embedded operation is in progress. An embedded operation is indicated by the Status Register-1 Write-In-Progress bit (SR1V[1]) set to 1, until the operation is completed. Some example embedded operations are: Program, Erase, or Write Registers (WRR) operations.

Driving the CS# input to the logic low state enables the device, placing it in the Active Power mode. After Power-up, a falling edge on CS# is required prior to the start of any command.

## 3.5 Serial Input (SI) / IO0

This input signal is used to transfer data serially into the device. It receives instructions, addresses, and data to be programmed. Values are latched on the rising edge of serial SCK clock signal.

SI becomes IO0 - an input and output during Dual and Quad commands for receiving instructions, addresses, and data to be programmed (values latched on rising edge of serial SCK clock signal) as well as shifting out data (on the falling edge of SCK, in SDR commands, and on every edge of SCK, in DDR commands).

## 3.6 Serial Output (SO) / IO1

This output signal is used to transfer data serially out of the device. Data is shifted out on the falling edge of the serial SCK clock signal.

SO becomes IO1 - an input and output during Dual and Quad commands for receiving addresses, and data to be programmed (values latched on rising edge of serial SCK clock signal) as well as shifting out data (on the falling edge of SCK, in SDR commands, and on every edge of SCK, in DDR commands).

## 3.7 Write Protect (WP#) / IO2

When WP# is driven Low ($V_{IL}$), during a WRR or WRAR command and while the Status Register Write Disable (SRWD_NV) bit of Status Register-1 (SR1NV[7]) is set to a 1, it is not possible to write to Status Register-1 or Configuration Register-1 related registers. In this situation, a WRR command is ignored, a WRAR command selecting SR1NV, SR1V, CR1NV, or CR1V is ignored, and no error is set.

This prevents any alteration of the Block Protection settings. As a consequence, all the data bytes in the memory area that are protected by the Block Protection feature are also hardware protected against data modification if WP# is Low during a WRR or WRAR command with SRWD_NV set to 1.

The WP# function is not available when the Quad mode is enabled (CR1V[1]=1). The WP# function is replaced by IO2 for input and output during Quad mode for receiving addresses, and data to be programmed (values are latched on rising edge of the SCK signal) as well as shifting out data (on the falling edge of SCK, in SDR commands, and on every edge of SCK, in DDR commands).

WP# has an internal pull-up resistance; when unconnected, WP# is at $V_{IH}$ and may be left unconnected in the host system if not used for Quad mode or protection.

## 3.8 IO3_RESET#

IO3 is used for input and output during Quad mode (CR1V[1]=1) for receiving addresses, and data to be programmed (values are latched on rising edge of the SCK signal) as well as shifting out data (on the falling edge of SCK, in SDR commands, and on every edge of SCK, in DDR commands).

The IO3_RESET# signal may also be used to initiate the hardware reset function when the reset feature is enabled by writing Configuration Register-2 non-volatile bit 5 (CR2V[5]=1). The input is only treated as RESET# when the device is not in Quad-I/O mode, CR1V[1] = 0, or when CS# is high. When Quad I/O mode is in use, CR1V[1]=1, and the device is selected with CS# low, the IO3_RESET# is used only as IO3 for information transfer. When CS# is high, the IO3_RESET# is not in use for information transfer and is used as the RESET# input. By conditioning the reset operation on CS# high during Quad mode, the reset function remains available during Quad mode.

When the system enters a reset condition, the CS# signal must be driven high as part of the reset process and the IO3_RESET# signal is driven low. When CS# goes high the IO3_RESET# input transitions from being IO3 to being the RESET# input. The reset condition is then detected when CS# remains high and the IO3_RESET# signal remains low for $t_{RP}$. If a reset is not intended, the system is required to actively drive IO3_RESET# to high along with CS# being driven high at the end of a transfer of data to the memory. Following transfers of data to the host system, the memory will drive IO3 high during $t_{CS}$. This will ensure that IO3 / Reset is not left floating or being pulled slowly to high by the internal or an external passive pull-up. Thus, an unintended reset is not triggered by the IO3_RESET# not being recognized as high before the end of $t_{RP}$.

The IO3_RESET# signal is unused when the reset feature is disabled (CR2V[5]=0).

The IO3_RESET# signal has an internal pull-up resistor and may be left unconnected in the host system if not used for Quad mode or the reset function. The internal pull-up will hold IO3_RESET# high after the host system has actively driven the signal high and then stops driving the signal.

Note that IO3_RESET# cannot be shared by more than one SPI-MIO memory if any of them are operating in Quad I/O mode as IO3 being driven to or from one selected memory may look like a reset signal to a second non-selected memory sharing the same IO3_RESET# signal.

## 3.9 RESET#

The RESET# input provides a hardware method of resetting the device to standby state, ready for receiving a command. When RESET# is driven to logic low ($V_{IL}$) for at least a period of $t_{RP}$, the device starts the hardware reset process.

The RESET# input initiates the reset operation when transitions from $V_{IH}$ to $V_{IL}$ for > $t_{RP}$, the device will reset register states in the same manner as power-on reset but, does not go through the full reset process that is performed during POR. The hardware reset process requires a period of $t_{RPH}$ to complete.RESET# may be asserted low at any time.

RESET# has an internal pull-up resistor and may be left unconnected in the host system if not used. The internal pull-up will hold Reset high after the host system has actively driven the signal high and then stops driving the signal.

The RESET# input is not available on all packages options. When not available the RESET# input of the device is tied to the inactive state.

When using the RESET# and not in QIO or QPI mode, do not use the IO3/RESET# pin.

## 3.10 Voltage Supply ($V_{CC}$)

$V_{CC}$ is the voltage source for all device internal logic. It is the single voltage used for all device internal functions including read, program, and erase.

## 3.11 Supply and Signal Ground ($V_{SS}$)

$V_{SS}$ is the common voltage drain and ground reference for the device core, input signal receivers, and output drivers.

## 3.12 Not Connected (NC)

No device internal signal is connected to the package connector nor is there any future plan to use the connector for a signal. The connection may safely be used for routing space for a signal on a Printed Circuit Board (PCB).

## 3.13 Reserved for Future Use (RFU)

No device internal signal is currently connected to the package connector but there is potential future use of the connector. It is recommended to not use RFU connectors for PCB routing channels so that the PCB may take advantage of future enhanced features in compatible footprint devices.

## 3.14 Do Not Use (DNU)

A device internal signal may be connected to the package connector. The connection may be used by Cypress for test or other purposes and is not intended for connection to any host system signal. Any DNU signal related function will be inactive when the signal is at $V_{IL}$. The signal has an internal pull-down resistor and may be left unconnected in the host system or may be tied to $V_{SS}$. Do not use these connections for PCB signal routing channels. Do not connect any host system signal to these connections.

## 3.15 System Block Diagrams

**Figure 1. Bus Master and Memory Devices on the SPI Bus - Single Bit Data Path**

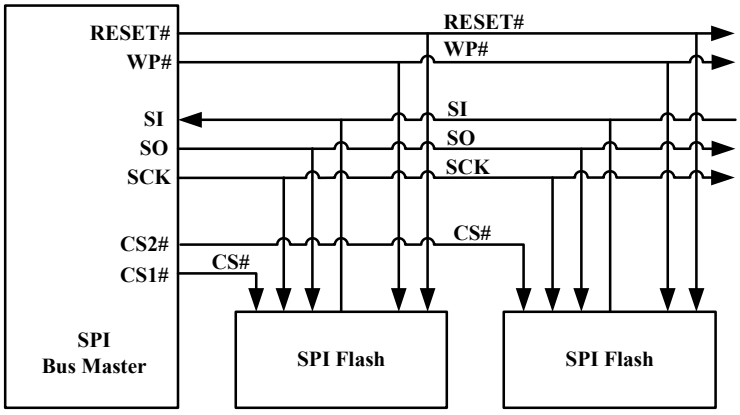



**Figure 2. Bus Master and Memory Devices on the SPI Bus - Dual Bit Data Path**

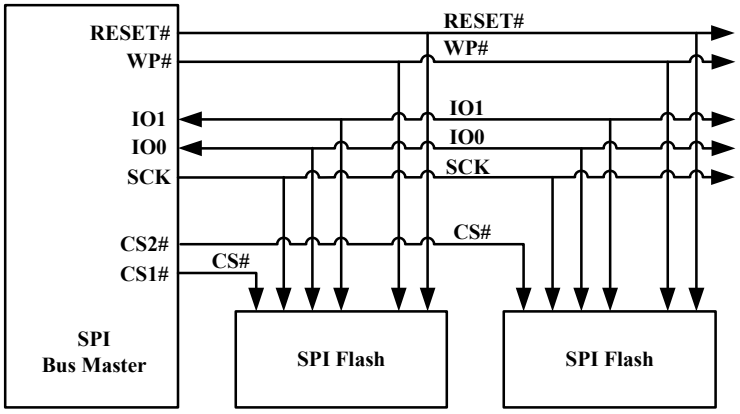



**Figure 3. Bus Master and Memory Devices on the SPI Bus - Quad Bit Data Path - Separate RESET#**

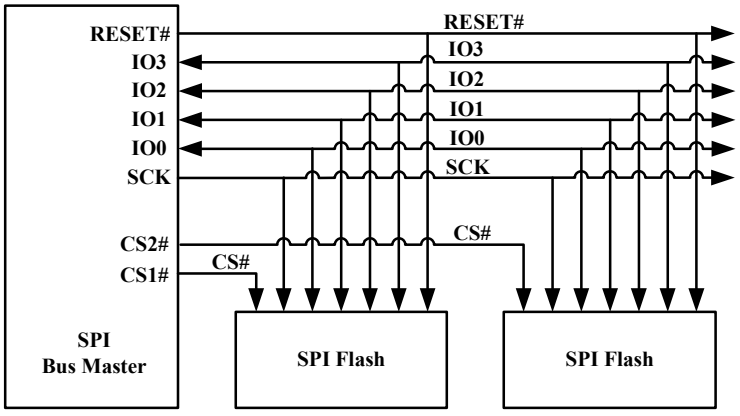
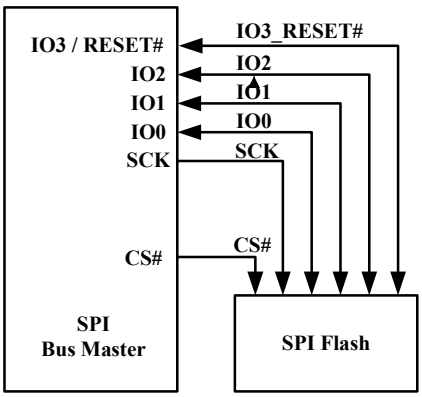


**Figure 4. Bus Master and Memory Devices on the SPI Bus - Quad Bit Data Path - I/O3_RESET#**



# 4. Signal Protocols

## 4.1 SPI Clock Modes

### 4.1.1 Single Data Rate (SDR)

The FS-S Family can be driven by an embedded microcontroller (bus master) in either of the two following clocking modes.

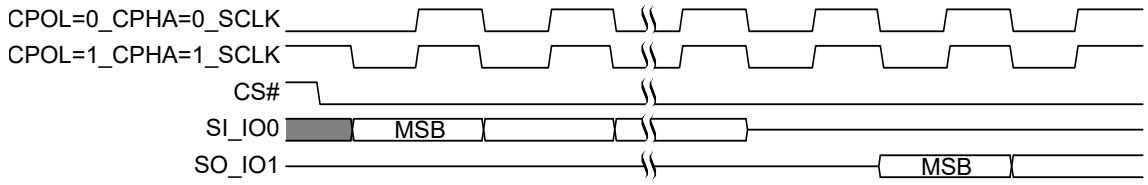
- **Mode 0** with Clock Polarity (CPOL) = 0 and, Clock Phase (CPHA) = 0
- **Mode 3** with CPOL = 1 and, CPHA = 1

For these two modes, input data into the device is always latched in on the rising edge of the SCK signal and the output data is always available from the falling edge of the SCK clock signal.

The difference between the two modes is the clock polarity when the bus master is in standby mode and not transferring any data.

- SCK will stay at logic low state with CPOL = 0, CPHA = 0
- SCK will stay at logic high state with CPOL = 1, CPHA = 1

**Figure 5. SPI SDR Modes Supported**




Timing diagrams throughout the remainder of the document are generally shown as both mode 0 and 3 by showing SCK as both high and low at the fall of CS#. In some cases a timing diagram may show only mode 0 with SCK low at the fall of CS#. In such a case, mode 3 timing simply means clock is high at the fall of CS# so no SCK rising edge set up or hold time to the falling edge of CS# is needed for mode 3.

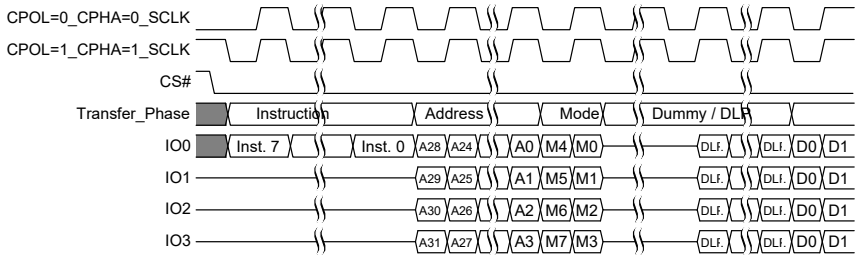
SCK cycles are measured (counted) from one falling edge of SCK to the next falling edge of SCK. In mode 0 the beginning of the first SCK cycle in a command is measured from the falling edge of CS# to the first falling edge of SCK because SCK is already low at the beginning of a command.

### 4.1.2 Double Data Rate (DDR)

Mode 0 and Mode 3 are also supported for DDR commands. In DDR commands, the instruction bits are always latched on the rising edge of clock, the same as in SDR commands. However, the address and input data that follow the instruction are latched on both the rising and falling edges of SCK. The first address bit is latched on the first rising edge of SCK following the falling edge at the end of the last instruction bit. The first bit of output data is driven on the falling edge at the end of the last access latency (dummy) cycle.

SCK cycles are measured (counted) in the same way as in SDR commands, from one falling edge of SCK to the next falling edge of SCK. In mode 0 the beginning of the first SCK cycle in a command is measured from the falling edge of CS# to the first falling edge of SCK because SCK is already low at the beginning of a command.

**Figure 6. SPI DDR Modes Supported**



## 4.2 Command Protocol

All communication between the host system and FS-S Family memory devices is in the form of units called commands.

All commands begin with an 8-bit instruction that selects the type of information transfer or device operation to be performed. Commands may also have an address, instruction modifier, latency period, data transfer to the memory, or data transfer from the memory. All instruction, address, and data information is transferred sequentially between the host system and memory device.

Command protocols are also classified by a numerical nomenclature using three numbers to reference the transfer width of three command phases:

- instruction
- address and instruction modifier (continuous read mode bits)
- data

Single bit wide commands start with an instruction and may provide an address or data, all sent only on the SI signal. Data may be sent back to the host serially on the SO signal. This is referenced as a 1-1-1 command protocol for single bit width instruction, single bit width address and modifier, single bit data.

Dual Output or Quad Output commands provide an address sent from the host as serial on SI (IO0) then followed by dummy cycles. Data is returned to the host as bit pairs on IO0 and IO1 or, four bit (nibble) groups on IO0, IO1, IO2, and IO3. This is referenced as 1-1-2 for Dual-O and 1-1-4 for Quad-O command protocols.

Dual or Quad Input / Output (I/O) commands provide an address sent from the host as bit pairs on IO0 and IO1 or, four bit (nibble) groups on IO0, IO1, IO2, and IO3. Data is returned to the host similarly as bit pairs on IO0 and IO1 or, four bit (nibble) groups on IO0, IO1, IO2, and IO3. This is referenced as 1-2-2 for Dual I/O and 1-4-4 for Quad I/O command protocols.

The FS-S Family also supports a QPI mode in which all information is transferred in 4-bit width, including the instruction, address, modifier, and data. This is referenced as a 4-4-4 command protocol.

Commands are structured as follows:

- Each command begins with CS# going low and ends with CS# returning high. The memory device is selected by the host driving the Chip Select (CS#) signal low throughout a command.
- The serial clock (SCK) marks the transfer of each bit or group of bits between the host and memory.
- Each command begins with an eight bit (byte) instruction. The instruction selects the type of information transfer or device operation to be performed. The instruction transfers occur on SCK rising edges. However, some read commands are modified by a prior read command, such that the instruction is implied from the earlier command. This is called Continuous Read Mode. When the device is in continuous read mode, the instruction bits are not transmitted at the beginning of the command because the instruction is the same as the read command that initiated the Continuous Read Mode. In Continuous Read mode the command will begin with the read address. Thus, Continuous Read Mode removes eight instruction bits from each read command in a series of same type read commands.
- The instruction may be stand alone or may be followed by address bits to select a location within one of several address spaces in the device. The instruction determines the address space used. The address may be either a 24 bit or a 32 bit, byte boundary, address. The address transfers occur on SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- In legacy SPI mode, the width of all transfers following the instruction are determined by the instruction sent. Following transfers may continue to be single bit serial on only the SI or Serial Output (SO) signals, they may be done in two bit groups per (dual) transfer on the IO0 and IO1 signals, or they may be done in 4 bit groups per (quad) transfer on the IO0-IO3 signals. Within the dual or quad groups the least significant bit is on IO0. More significant bits are placed in significance order on each higher numbered IO signal. Single bits or parallel bit groups are transferred in most to least significant bit order.
- In QPI mode, the width of all transfers is a 4-bit wide (quad) transfer on the IO0-IO3 signals.
- Dual and Quad I/O read instructions send an instruction modifier called Continuous Read mode bits, following the address, to indicate whether the next command will be of the same type with an implied, rather than an explicit, instruction. These mode bits initiate or end the continuous read mode. In continuous read mode, the next command thus does not provide an instruction byte, only a new address and mode bits. This reduces the time needed to send each command when the same command type is repeated in a sequence of commands. The mode bit transfers occur on SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- The address or mode bits may be followed by write data to be stored in the memory device or by a read latency period before read data is returned to the host.

- Write data bit transfers occur on SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- SCK continues to toggle during any read access latency period. The latency may be zero to several SCK cycles (also referred to as dummy cycles). At the end of the read latency cycles, the first read data bits are driven from the outputs on SCK falling edge at the end of the last read latency cycle. The first read data bits are considered transferred to the host on the following SCK rising edge. Each following transfer occurs on the next SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- If the command returns read data to the host, the device continues sending data transfers until the host takes the CS# signal high. The CS# signal can be driven high after any transfer in the read data sequence. This will terminate the command.
- At the end of a command that does not return data, the host drives the CS# input high. The CS# signal must go high after the eighth bit, of a stand alone instruction or, of the last write data byte that is transferred. That is, the CS# signal must be driven high when the number of bits after the CS# signal was driven low is an exact multiple of eight bits. If the CS# signal does not go high exactly at the eight bit boundary of the instruction or write data, the command is rejected and not executed.
- All instruction, address, and mode bits are shifted into the device with the Most Significant Bits (MSB) first. The data bits are shifted in and out of the device MSB first. All data is transferred in byte units with the lowest address byte sent first. Following bytes of data are sent in lowest to highest byte address order i.e. the byte address increments.
- All attempts to read the flash memory array during a program, erase, or a write cycle (embedded operations) are ignored. The embedded operation will continue to execute without any affect. A very limited set of commands are accepted during an embedded operation. These are discussed in the individual command descriptions.
- Depending on the command, the time for execution varies. A command to read status information from an executing command is available to determine when the command completes execution and whether the command was successful.

### 4.2.1 Command Sequence Examples

**Figure 7. Stand Alone Instruction Command**

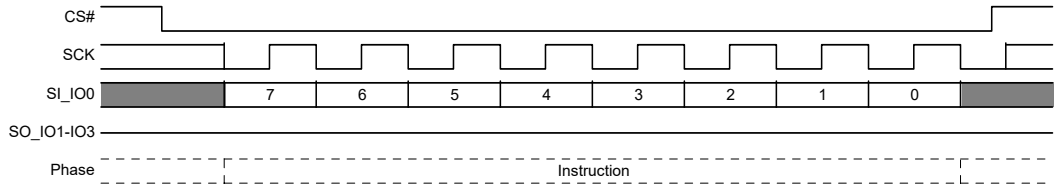



**Figure 8. Single Bit Wide Input Command**

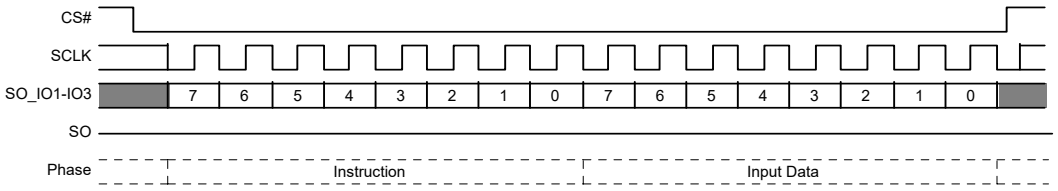



**Figure 9. Single Bit Wide Output Command without latency**

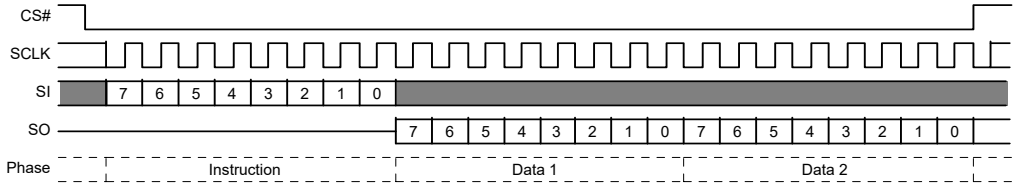


**Figure 10. Single Bit Wide I/O Command with latency**

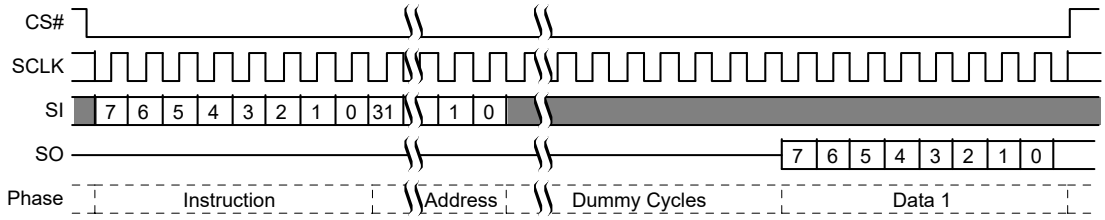



**Figure 11. Dual Output Read Command**

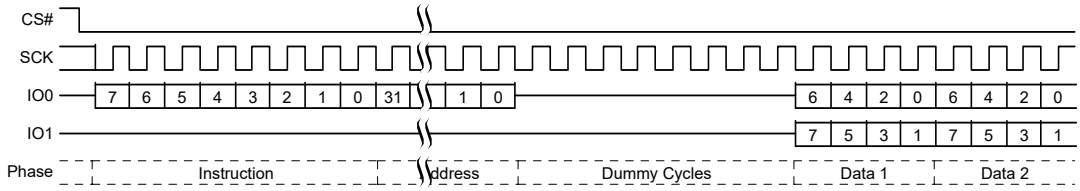



**Figure 12. Quad Output Read Command**

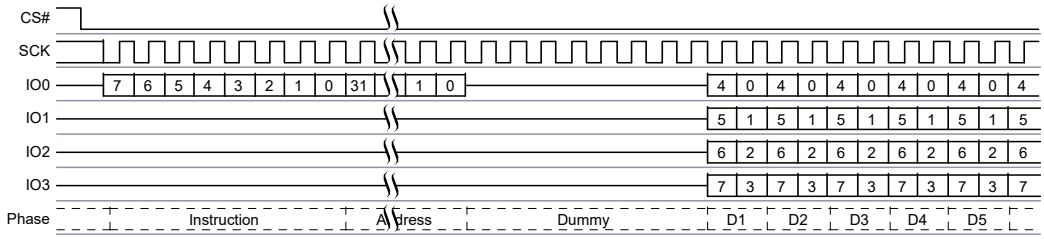



**Figure 13. Dual I/O Command**

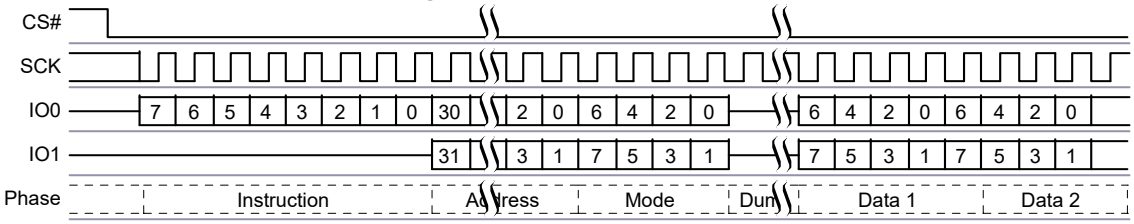
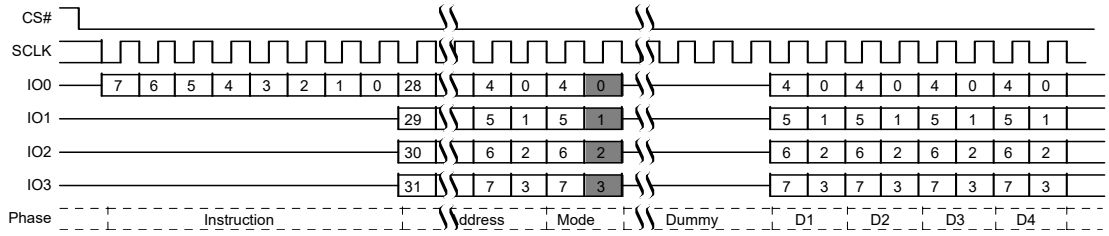


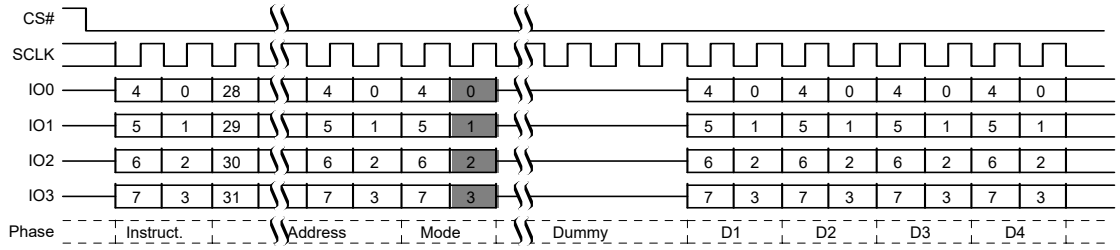

**Figure 14. Quad I/O Command**




**Note**

5. The gray bits are optional, the host does not have to drive bits during that cycle.

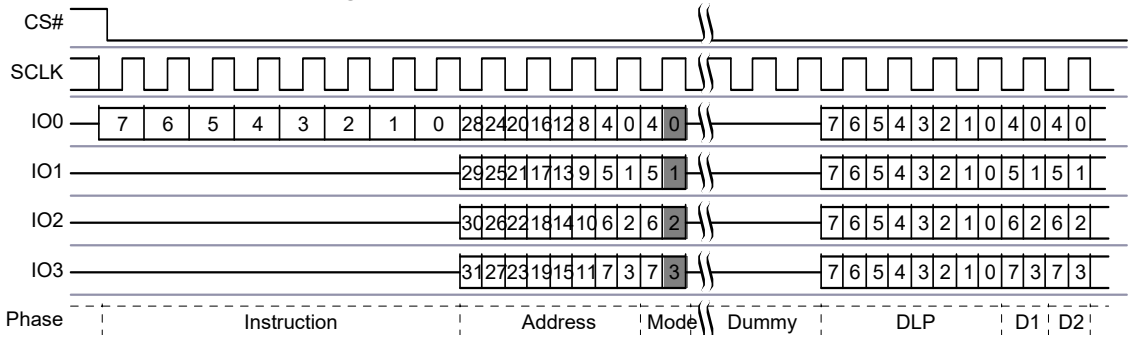
**Figure 15. Quad I/O Read Command in QPI Mode**




**Note**

6. The gray bits are optional, the host does not have to drive bits during that cycle.

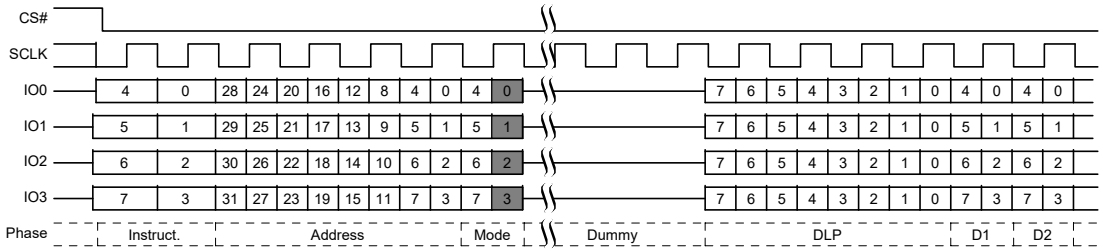
**Figure 16. DDR Quad I/O Read Command**




**Note**

7. The gray bits are optional, the host does not have to drive bits during that cycle.

**Figure 17. DDR Quad I/O Read Command QPI Mode**




**Note**

8. The gray bits are optional, the host does not have to drive bits during that cycle.

Additional sequence diagrams, specific to each command, are provided in Commands on page 72.

## 4.3 Interface States

This section describes the input and output signal levels as related to the SPI interface behavior.

**Table 3. Interface States Summary**

| Interface State | $V_{CC}$ | SCK | CS# | RESET# | IO3_RESET# | WP# / IO2 | SO / IO1 | SI / IO0 |
|---|---|---|---|---|---|---|---|---|
| Power-Off | <$V_{CC}$ (low) | X | X | X | X | X | X | X |
| Low Power<br>Hardware Data Protection | <$V_{CC}$ (cut-off) | X | X | X | X | X | Z | X |
| Power-On (Cold) Reset | ≥$V_{CC}$ (min) | X | HH | X | X | X | Z | X |
| Hardware (Warm) Reset Non-Quad Mode | ≥$V_{CC}$ (min) | X | X | HL | HL | X | Z | X |
| Hardware (Warm) Reset Quad Mode | ≥$V_{CC}$ (min) | X | HH | HL | HL | X | Z | X |
| Interface Standby | ≥$V_{CC}$ (min) | X | HH | HH | X | X | Z | X |
| Instruction Cycle (Legacy SPI) | ≥$V_{CC}$ (min) | HT | HL | HH | HH | HV | Z | HV |
| Single Input Cycle<br>Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | Z | HV |
| Single Latency (Dummy) Cycle | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | Z | X |
| Single Output Cycle<br>Memory to Host Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | MV | X |
| Dual Input Cycle<br>Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | HV | HV |
| Dual Latency (Dummy) Cycle | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | X | X |
| Dual Output Cycle<br>Memory to Host Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | MV | MV |
| Quad Input Cycle<br>Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HV | HV | HV | HV |
| Quad Latency (Dummy) Cycle | ≥$V_{CC}$ (min) | HT | HL | HH | X | X | X | X |
| Quad Output Cycle<br>Memory to Host Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | MV | MV | MV | MV |
| DDR Quad Input Cycle<br>Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HV | HV | HV | HV |
| DDR Latency (Dummy) Cycle | ≥$V_{CC}$ (min) | HT | HL | HH | MV or Z | MV or Z | MV or Z | MV or Z |
| DDR Quad Output Cycle<br>Memory to Host Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | MV | MV | MV | MV |

**LEGEND** Z = No driver - floating signal, HL = Host driving $V_{IL}$, HH = Host driving $V_{IH}$, HV = either HL or HH, X = HL or HH or Z, HT = toggling between HL and HH, ML = Memory driving $V_{IL}$, MH = Memory driving $V_{IH}$, MV = either ML or MH.

### 4.3.1 $V_{CC}$ Power-Off

When the core supply voltage is at or below the $V_{CC\ (Low)}$ voltage, the device is considered to be powered off. The device does not react to external signals, and is prevented from performing any program or erase operation.

### 4.3.2 Low Power Hardware Data Protection

When $V_{CC}$ is less than $V_{CC\ (Cut\text{-}off)}$ the memory device will ignore commands to ensure that program and erase operations can not start when the core supply voltage is out of the operating range.

### 4.3.3 Power-On (Cold) Reset

When the core voltage supply remains at or below the $V_{CC\ (Low)}$ voltage for ≥ $t_{PD}$ time, then rises to ≥ $V_{CC\ (Minimum)}$ the device will begin its Power On Reset (POR) process. POR continues until the end of $t_{PU}$. During $t_{PU}$ the device does not react to external input signals nor drive any outputs. Following the end of $t_{PU}$ the device transitions to the Interface Standby state and can accept commands. For additional information on POR see Power On (Cold) Reset on page 31.

### 4.3.4 Hardware (Warm) Reset

A configuration option is provided to allow IO3_RESET# to be used as a hardware reset input when the device is not in any Quad or QPI mode or when it is in any Quad mode or QPI mode and CS# is high. In Quad or QPI mode on some packages a separate reset input is provided (RESET #). When IO3_RESET# or RESET# is driven low for $t_{RP}$ time the device starts the hardware reset process. The process continues for $t_{RPH}$ time. Following the end of both $t_{RPH}$ and the reset hold time following the rise of RESET# ($t_{RH}$) the device transitions to the Interface Standby state and can accept commands. For additional information on hardware reset, see RESET # and IO3_RESET# Input Initiated Hardware (Warm) Reset on page 32.

### 4.3.5 Interface Standby

When CS# is high the SPI interface is in standby state. Inputs other than RESET# are ignored. The interface waits for the beginning of a new command. The next interface state is Instruction Cycle when CS# goes low to begin a new command.

While in interface standby state the memory device draws standby current ($I_{SB}$) if no embedded algorithm is in progress. If an embedded algorithm is in progress, the related current is drawn until the end of the algorithm when the entire device returns to standby current draw.

A Deep Power Down (DPD) mode is supported by the FS-S Family devices. If the device has been placed in DPD mode by the DPD (B9h) command, the interface standby current is ($I_{DPD}$). The DPD command is accepted only while the device is not performing an embedded algorithm as indicated by the Status Register-1 volatile Write In Progress (WIP) bit being cleared to zero (SR1V[0] = 0). While in DPD mode the device ignores all commands except the Release from DPD (RES ABh) command, that will return the device to the Interface Standby state after a delay of $t_{RES}$.

### 4.3.6 Instruction Cycle (Legacy SPI Mode)

When the host drives the MSB of an instruction and CS# goes low, on the next rising edge of SCK the device captures the MSB of the instruction that begins the new command. On each following rising edge of SCK the device captures the next lower significance bit of the 8 bit instruction. The host keeps CS# low, and drives the Write Protect (WP#) and IO3_RESET#/RESET# signals as needed for the instruction. However, WP# is only relevant during instruction cycles of a WRR or WRAR command or any other commands which affect Status registers, Configuration registers and DLR registers, and is other wise ignored. IO3_RESET# is driven high when the device is not in Quad Mode (CR1V[1] = 0) or QPI Mode (CR2V[3] = 0) and hardware reset is not required.

Each instruction selects the address space that is operated on and the transfer format used during the remainder of the command. The transfer format may be Single, Dual O, Quad O, Dual I/O, or Quad I/O, or DDR Quad I/O. The expected next interface state depends on the instruction received.

Some commands are stand alone, needing no address or data transfer to or from the memory. The host returns CS# high after the rising edge of SCK for the eighth bit of the instruction in such commands. The next interface state in this case is Interface Standby.

### 4.3.7 Instruction Cycle (QPI Mode)

In QPI mode, when CR2V[6]=0, instructions are transferred 4 bits per cycle. In this mode instruction cycles are the same as a Quad Output Cycle. See Quad Input Cycle — Host to Memory Transfer on page 21.

### 4.3.8 Single Input Cycle — Host to Memory Transfer

Several commands transfer information after the instruction on the single serial input (SI) signal from host to the memory device. The host keeps RESET# high, CS# low, and drives SI as needed for the command. The memory does not drive the Serial Output (SO) signal.

The expected next interface state depends on the instruction. Some instructions continue sending address or data to the memory using additional Single Input Cycles. Others may transition to Single Latency, or directly to Single, Dual, or Quad Output cycle states.

### 4.3.9 Single Latency (Dummy) Cycle

Read commands may have zero to several latency cycles during which read data is read from the main Flash memory array before transfer to the host. The number of latency cycles are determined by the Latency Code in the configuration register (CR3V[3:0]). During the latency cycles, the host keeps RESET# and IO3_RESET# high, CS# low and SCK toggles. The Write Protect (WP#) signal is ignored. The host may drive the SI signal during these cycles or the host may leave SI floating. The memory does not use any data driven on SI or other I/O signals during the latency cycles. The memory does not drive the Serial Output (SI) or I/O signals during the latency cycles.

The next interface state depends on the command structure i.e. the number of latency cycles, and whether the read is single, dual, or quad width.

### 4.3.10 Single Output Cycle — Memory to Host Transfer

Several commands transfer information back to the host on the single Serial Output (SO) signal. The host keeps RESET# and IO3_RESET# high, CS# low. The Write Protect (WP#) signal is ignored. The memory ignores the Serial Input (SI) signal. The memory drives SO with data.

The next interface state continues to be Single Output Cycle until the host returns CS# to high ending the command.

### 4.3.11 Dual Input Cycle — Host to Memory Transfer

The Read Dual I/O command transfers two address or mode bits to the memory in each cycle. The host keeps RESET# and IO3_RESET# high, CS# low. The Write Protect (WP#) signal is ignored. The host drives address on SI / IO0 and SO / IO1.

The next interface state following the delivery of address and mode bits is a Dual Latency Cycle if there are latency cycles needed or Dual Output Cycle if no latency is required.

### 4.3.12 Dual Latency (Dummy) Cycle

Read commands may have zero to several latency cycles during which read data is read from the main Flash memory array before transfer to the host. The number of latency cycles are determined by the Latency Code in the configuration register (CR3V[3:0]). During the latency cycles, the host keeps RESET# and IO3_RESET# high, CS# low, and SCK continues to toggle. The Write Protect (WP#) signal is ignored. The host may drive the SI / IO0 and SO / IO1 signals during these cycles or the host may leave SI / IO0 and SO / IO1 floating. The memory does not use any data driven on SI / IO0 and SO / IO1 during the latency cycles. The host must stop driving SI / IO0 and SO / IO1 on the falling edge of SCK at the end of the last latency cycle. It is recommended that the host stop driving them during all latency cycles so that there is sufficient time for the host drivers to turn off before the memory begins to drive at the end of the latency cycles. This prevents driver conflict between host and memory when the signal direction changes. The memory does not drive the SI / IO0 and SO / IO1 signals during the latency cycles.

The next interface state following the last latency cycle is a Dual Output Cycle.

### 4.3.13 Dual Output Cycle — Memory to Host Transfer

The Read Dual Output and Read Dual I/O return data to the host two bits in each cycle. The host keeps RESET# and IO3_RESET# high, CS# low. The Write Protect (WP#) signal is ignored. The memory drives data on the SI / IO0 and SO / IO1 signals during the dual output cycles on the falling edge of SCK.

The next interface state continues to be Dual Output Cycle until the host returns CS# to high ending the command.

### 4.3.14 QPP or QOR Address Input Cycle

The Quad Page Program and Quad Output Read commands send address to the memory only on IO0. The other IO signals are ignored. The host keeps RESET# and IO3_RESET# high, CS# low, and drives IO0.

For QPP the next interface state following the delivery of address is the Quad Input Cycle. For QOR the next interface state following address is a Quad Latency Cycle if there are latency cycles needed or Quad Output Cycle if no latency is required.

### 4.3.15 Quad Input Cycle — Host to Memory Transfer

The Quad I/O Read command transfers four address or mode bits to the memory in each cycle. In QPI mode the Quad I/O Read and Page Program commands transfer four data bits to the memory in each cycle, including the instruction cycles. The host keeps CS# low, and drives the IO signals.

For Quad I/O Read the next interface state following the delivery of address and mode bits is a Quad Latency Cycle if there are latency cycles needed or Quad Output Cycle if no latency is required. For QPI mode Page Program, the host returns CS# high following the delivery of data to be programmed and the interface returns to standby state.

### 4.3.16 Quad Latency (Dummy) Cycle

Read commands may have zero to several latency cycles during which read data is read from the main Flash memory array before transfer to the host. The number of latency cycles are determined by the Latency Code in the configuration register (CR3V[3:0]). During the latency cycles, the host keeps CS# low and continues to toggle SCK. The host may drive the IO signals during these cycles or the host may leave the IO floating. The memory does not use any data driven on IO during the latency cycles. The host must stop driving the IO signals on the falling edge at the end of the last latency cycle. It is recommended that the host stop driving them during all latency cycles so that there is sufficient time for the host drivers to turn off before the memory begins to drive at the end of the latency cycles. This prevents driver conflict between host and memory when the signal direction changes. The memory does not drive the IO signals during the latency cycles.

The next interface state following the last latency cycle is a Quad Output Cycle.

### 4.3.17 Quad Output Cycle — Memory to Host Transfer

The Quad-O and Quad I/O Read returns data to the host four bits in each cycle. The host keeps CS# low. The memory drives data on IO0-IO3 signals during the Quad output cycles.

The next interface state continues to be Quad Output Cycle until the host returns CS# to high ending the command.

### 4.3.18 DDR Quad Input Cycle — Host to Memory Transfer

The DDR Quad I/O Read command sends address, and mode bits to the memory on all the IO signals. Four bits are transferred on the rising edge of SCK and four bits on the falling edge in each cycle. The host keeps CS# low.

The next interface state following the delivery of address and mode bits is a DDR Latency Cycle.

### 4.3.19 DDR Latency Cycle

DDR Read commands may have one to several latency cycles during which read data is read from the main Flash memory array before transfer to the host. The number of latency cycles are determined by the Latency Code in the configuration register (CR2V[3:0]). During the latency cycles, the host keeps CS# low. The host may not drive the IO signals during these cycles. So that there is sufficient time for the host drivers to turn off before the memory begins to drive. This prevents driver conflict between host and memory when the signal direction changes. The memory has an option to drive all the IO signals with a Data Learning Pattern (DLP) during the last 4 latency cycles. The DLP option should not be enabled when there are fewer than five latency cycles so that there is at least one cycle of high impedance for turn around of the IO signals before the memory begins driving the DLP. When there are more than 4 cycles of latency the memory does not drive the IO signals until the last four cycles of latency.

The next interface state following the last latency cycle is a DDR Single, or Quad Output Cycle, depending on the instruction.

### 4.3.20 DDR Quad Output Cycle — Memory to Host Transfer

The DDR Quad I/O Read command returns bits to the host on all the IO signals. Four bits are transferred on the rising edge of SCK and four bits on the falling edge in each cycle. The host keeps CS# low.

The next interface state continues to be DDR Quad Output Cycle until the host returns CS# to high ending the command.

## 4.4 Configuration Register Effects on the Interface

The configuration register 2 volatile bits 3 to 0 (CR2V[3:0]) select the variable latency for all array read commands except Read, RUID and Read SDFP (RSFDP). Read always has zero latency cycles. RSFDP always has 8 latency cycles. The variable latency is also used in the OTPR, ECCRD, and RDAR commands.

The configuration register bit1 (CR1V[1]) selects whether Quad mode is enabled to switch WP# to IO2 function, RESET# to IO3 function, and thus allow Quad I/O Read and QPI mode commands. Quad mode must also be selected to allow DDR Quad I/O Read commands.

## 4.5 Data Protection

Some basic protection against unintended changes to stored data are provided and controlled purely by the hardware design. These are described below. Other software managed protection methods are discussed in the software section of this document.

### 4.5.1 Power-Up

When the core supply voltage is at or below the $V_{CC\ (Low)}$ voltage, the device is considered to be powered off. The device does not react to external signals, and is prevented from performing any program or erase operation. Program and erase operations continue to be prevented during the Power-on Reset (POR) because no command is accepted until the exit from POR to the Interface Standby state.

### 4.5.2 Low Power

When $V_{CC}$ is less than $V_{CC\ (Cut\text{-}off)}$ the memory device will ignore commands to ensure that program and erase operations can not start when the core supply voltage is out of the operating range.

### 4.5.3 Clock Pulse Count

The device verifies that all non-volatile memory and register data modifying commands consist of a clock pulse count that is a multiple of eight bit transfers (byte boundary) before executing them. A command not ending on an 8 bit (byte) boundary is ignored and no error status is set for the command.

### 4.5.4 Deep Power Down (DPD)

In DPD mode the device responds only to the Resume from DPD command (RES ABh). All other commands are ignored during DPD mode, thereby protecting the memory from program and erase operations. If the IO3_RESET# function has been enabled (CR2V[5]=1) or if RESET# is active, IO3_RESET# or RESET# going low will start a hardware reset and release the device from DPD mode.

# 5. Electrical Specifications

## 5.1 Absolute Maximum Ratings

Storage Temperature Plastic Packages .....–65 °C to +150 °C

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

$V_{CC}$ .....–0.5 V to +2.5 V

Input voltage with respect to Ground (VSS) (Note 1) .....–0.5 V to $V_{CC}$ + 0.5 V

Output Short Circuit Current (Note 2) ..... 100 mA

**Notes**

9. See Input Signal Overshoot on page 24 for allowed maximums during signal transition.
10. 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.
11. Stresses above those listed under “Absolute Maximum Ratings” 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 data sheet is not implied. Exposure of the device to absolute maximum rating conditions for extended periods may affect device reliability.

## 5.2 Latchup Characteristics

**Table 4. Latchup Specification**

| Description | Min | Max | Unit |
|---|---|---|---|
| Input voltage with respect to $V_{SS}$ on all input only connections | –1.0 | $V_{CC}$ + 1.0 | V |
| Input voltage with respect to $V_{SS}$ on all I/O connections | –1.0 | $V_{CC}$ + 1.0 | V |
| $V_{CC}$ Current | –100 | +100 | mA |

**Note**

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

## 5.3 Thermal Resistance

**Table 5. Thermal Resistance**

| Parameter | Description | W9A008 | SOC008 | FAB024 | Unit |
|---|---|---|---|---|---|
| Theta JA | Thermal resistance (junction to ambient) | 85.8 | 53.27 | 38.93 | °C/W |

## 5.4 Operating Ranges

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

### 5.4.1 Power Supply Voltages

| | |
|---|---|
| $V_{CC}$ | 1.7V to 2.0V |

### 5.4.2 Temperature Ranges

| Parameter | Symbol | Devices | Spec | | Unit |
|---|---|---|---|---|---|
| | | | Min | Max | |
| Ambient Temperature | $T_A$ | Industrial (I) | –40 | +85 | °C |
| | | Industrial Plus (V) | –40 | +105 | |
| | | Extended (N) | –40 | +125 | |
| | | Automotive, AEC-Q100 Grade 3 (A) | –40 | +85 | |
| | | Automotive, AEC-Q100 Grade 2 (B) | –40 | +105 | |
| | | Automotive, AEC-Q100 Grade 1 (M) | –40 | +125 | |

Industrial Plus operating and performance parameters will be determined by device characterization and may vary from standard industrial temperature range devices as currently shown in this specification.

### 5.4.3 Input Signal Overshoot

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

**Figure 18. Maximum Negative Overshoot Waveform**

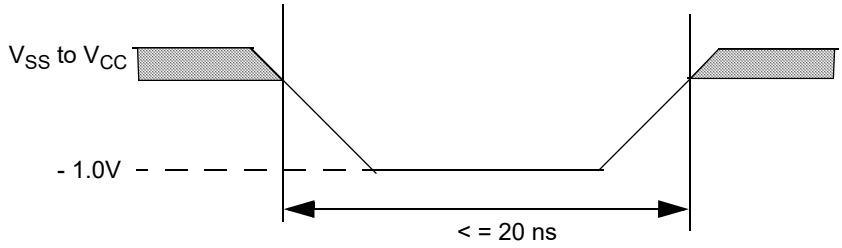
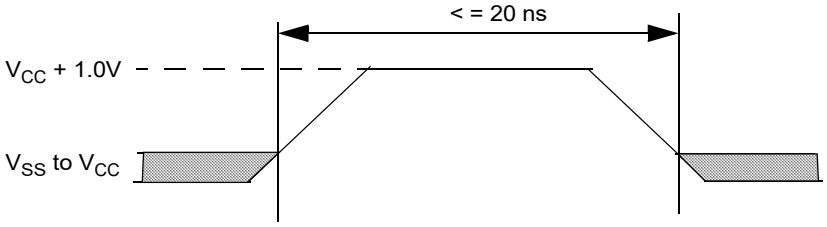



**Figure 19. Maximum Positive Overshoot Waveform**




## 5.5 Power-Up and Power-Down

The device must not be selected at power-up or power-down (that is, CS# must follow the voltage applied on $V_{CC}$) 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

A simple pull-up resistor on Chip Select (CS#) can usually be used to insure safe and proper power-up and power-down.

The device ignores all instructions until a time delay of $t_{PU}$ has elapsed after the moment that $V_{CC}$ rises above the minimum $V_{CC}$ threshold. See Figure 20. 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 $I_{POR}$ during $t_{PU}$. After power-up ($t_{PU}$), the device is in Standby mode, draws CMOS standby current ($I_{SB}$), and the WEL bit is reset.

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 21. 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 Power-on Reset (POR) did not complete correctly after power up, the assertion of the RESET# signal or receiving a software reset command (RESET) will restart the POR process.

Normal precautions must be taken for supply rail decoupling to stabilize the $V_{CC}$ supply at the device. Each device in a system should have the $V_{CC}$ rail decoupled by a suitable capacitor close to the package supply connection (this capacitor is generally of the order of 0.1µf).

**Table 6. Power-Up / Power-Down Voltage and Timing**

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $V_{CC}$ (min) | $V_{CC}$ (minimum operation voltage) | 1.7 | | V |
| $V_{CC}$ (cut-off) | $V_{CC}$ (Cut 0ff where re-initialization is needed) | 1.55 | | V |
| $V_{CC}$ (low) | $V_{CC}$ (low voltage for initialization to occur) | 0.7 | | V |
| $t_{PU}$ | $V_{CC}$(min) to Read operation | | 300 | µs |
| $t_{PD}$ | $V_{CC}$(low) time | 10.0 | | µs |

**Figure 20. Power-up**

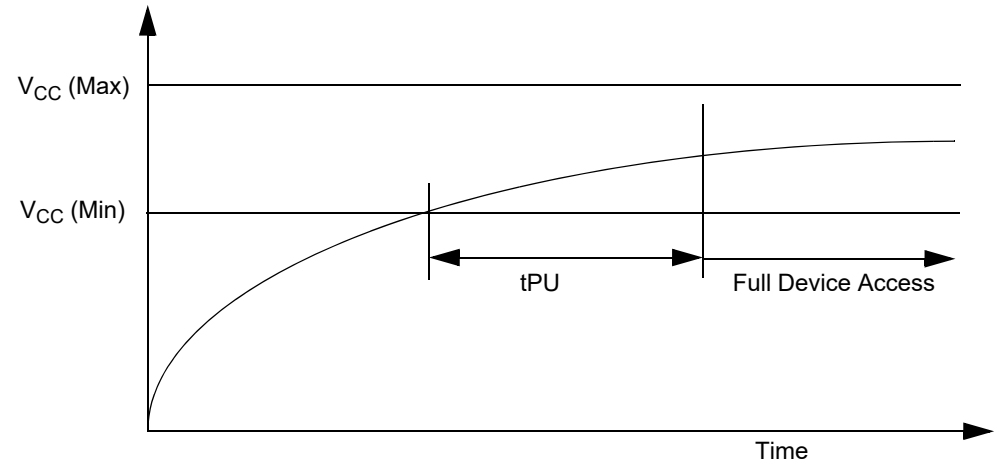
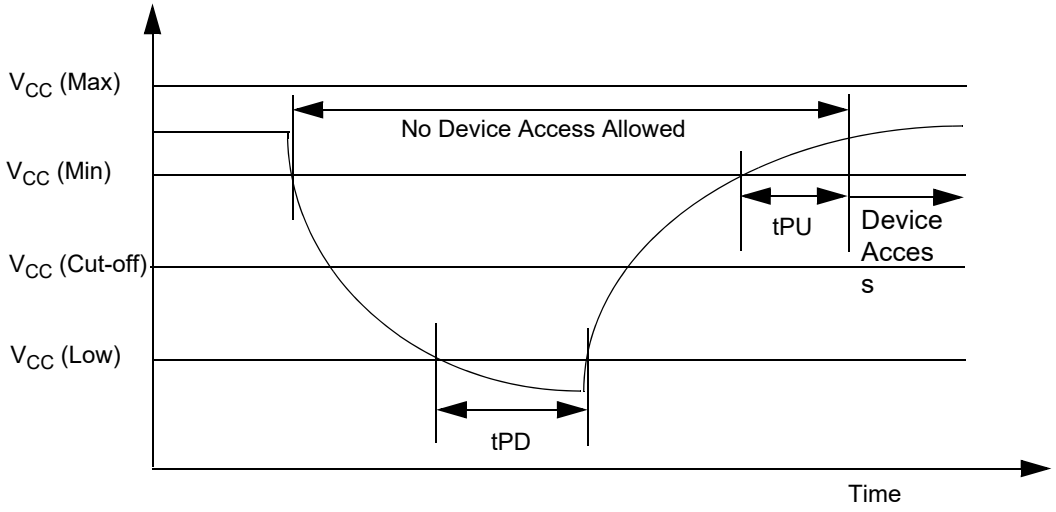



**Figure 21. Power-down and Voltage Drop**



## 5.6 DC Characteristics

### 5.6.1 Industrial

Applicable within operating -40°C to +85°C range.

**Table 7. DC Characteristics - Industrial**

| Symbol | Parameter | Test Conditions | Min | Typ (13) | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input Low Voltage | | -0.5 | | $0.3xV_{CC}$ | V |
| $V_{IH}$ | Input High Voltage | | $0.7xV_{CC}$ | | $V_{CC}+0.4$ | V |
| $V_{OL}$ | Output Low Voltage | $I_{OL}$ = 0.1 mA | | | 0.2 | V |
| $V_{OH}$ | Output High Voltage | $I_{OH}$ = –0.1 mA | $V_{CC}$ - 0.2 | | | V |
| $I_{LI}$ | Input Leakage Current | $V_{CC}=V_{CC}$ Max, $V_{IN}=V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | | | ±2 | µA |
| $I_{LO}$ | Output Leakage Current | $V_{CC}=V_{CC}$ Max, $V_{IN}=V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | | | ±2 | µA |
| $I_{CC1}$ | Active Power Supply Current (READ) (14) | Serial SDR@50 MHz<br>Serial SDR@133 MHz<br>QIO/QPI SDR@133 MHz<br>QIO/QPI DDR@80 MHz | | 10<br>25<br>60<br>70 | 18<br>30<br>65<br>90 | mA |
| $I_{CC2}$ | Active Power Supply Current (Page Program) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{CC3}$ | Active Power Supply Current (WRR or WRAR) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{CC4}$ | Active Power Supply Current (SE) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{CC5}$ | Active Power Supply Current (BE) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{SB}$ | Standby Current | IO3/RESET#, CS#=$V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$, Industrial Temp | | 25 | 100 | µA |
| $I_{DPD}$ | Deep Power Down Current | IO3/RESET#, CS#=$V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$, Industrial Temp | | 6 | 50 | µA |
| $I_{POR}$ | Power On Reset Current | IO3/RESET#, CS#=$V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$ | | | 80 | mA |

**Notes**

13. Typical values are at $T_{AI}$ = 25°C and $V_{CC}$ = 1.8V.
14. Outputs unconnected during read data return. Output switching current is not included.

### 5.6.2 Industrial Plus

Applicable within operating -40°C to +105°C range.

**Table 8. DC Characteristics - Industrial Plus**

| Symbol | Parameter | Test Conditions | Min | Typ (13) | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input Low Voltage | | -0.5 | | $0.3xV_{CC}$ | V |
| $V_{IH}$ | Input High Voltage | | $0.7xV_{CC}$ | | $V_{CC}+0.4$ | V |
| $V_{OL}$ | Output Low Voltage | $I_{OL}$ = 0.1 mA | | | 0.2 | V |
| $V_{OH}$ | Output High Voltage | $I_{OH}$ = –0.1 mA | $V_{CC}$ - 0.2 | | | V |
| $I_{LI}$ | Input Leakage Current | $V_{CC}=V_{CC}$ Max, $V_{IN}=V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | | | ±4 | µA |
| $I_{LO}$ | Output Leakage Current | $V_{CC}=V_{CC}$ Max, $V_{IN}=V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | | | ±4 | µA |
| $I_{CC1}$ | Active Power Supply Current (READ) | Serial SDR@50 MHz<br>Serial SDR@133 MHz<br>QIO/QPI SDR@133 MHz<br>QIO/QPI DDR@80 MHz | | 10<br>25<br>60<br>70 | 18<br>30<br>65<br>90 | mA |
| $I_{CC2}$ | Active Power Supply Current (Page Program) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{CC3}$ | Active Power Supply Current (WRR or WRAR) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{CC4}$ | Active Power Supply Current (SE) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{CC5}$ | Active Power Supply Current (BE) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{SB}$ | Standby Current | IO3/RESET#, CS#=$V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$, | | 25 | 300 | µA |
| $I_{DPD}$ | Deep Power Down Current | IO3/RESET#, CS#=$V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$, | | 6 | 100 | µA |
| $I_{POR}$ | Power On Reset Current | IO3/RESET#, CS#=$V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$ | | | 80 | mA |

**Notes**

15.Typical values are at $T_{AI}$ = 25°C and $V_{CC}$ = 1.8V.

16.Outputs unconnected during read data return. Output switching current is not included.

### 5.6.3 Extended

Applicable within operating -40°C to +125°C range.

**Table 9. DC Characteristics - Extended**

| Symbol | Parameter | Test Conditions | Min | Typ (13) | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input Low Voltage | | -0.5 | | $0.3xV_{CC}$ | V |
| $V_{IH}$ | Input High Voltage | | $0.7xV_{CC}$ | | $V_{CC}+0.4$ | V |
| $V_{OL}$ | Output Low Voltage | $I_{OL}$ = 0.1 mA | | | 0.2 | V |
| $V_{OH}$ | Output High Voltage | $I_{OH}$ = –0.1 mA | $V_{CC}$ - 0.2 | | | V |
| $I_{LI}$ | Input Leakage Current | $V_{CC}=V_{CC}$ Max, $V_{IN}=V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | | | ±4 | µA |
| $I_{LO}$ | Output Leakage Current | $V_{CC}=V_{CC}$ Max, $V_{IN}=V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | | | ±4 | µA |
| $I_{CC1}$ | Active Power Supply Current (READ) | Serial SDR@50 MHz<br>Serial SDR@133 MHz<br>QIO/QPI SDR@133 MHz<br>QIO/QPI DDR@80 MHz | | 10<br>25<br>60<br>70 | 18<br>30<br>65<br>90 | mA |
| $I_{CC2}$ | Active Power Supply Current (Page Program) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{CC3}$ | Active Power Supply Current (WRR or WRAR) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{CC4}$ | Active Power Supply Current (SE) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{CC5}$ | Active Power Supply Current (BE) | CS#=$V_{CC}$ | | 60 | 100 | mA |
| $I_{SB}$ | Standby Current | IO3/RESET#, CS#=$V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$, | | | 300 | µA |
| $I_{DPD}$ | Deep Power Down Current | IO3/RESET#, CS#=$V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$, | | 6 | 170 | µA |
| $I_{POR}$ | Power On Reset Current | IO3/RESET#, CS#=$V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$ | | | 80 | mA |

**Notes**

17. Typical values are at $T_{AI}$ = 25°C and $V_{CC}$ = 1.8V.
18. Outputs unconnected during read data return. Output switching current is not included.

### 5.6.4 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 $I_{SB}$.

### 5.6.5 Deep Power Down Power Mode (DPD)

The Deep Power Down mode is enabled by inputing the command instruction code “B9h” and the power consumption drops to $I_{DPD}$. The DPD command is accepted only while the device is not performing an embedded algorithm as indicated by the Status Register-1 volatile Write In Progress (WIP) bit being cleared to zero (SR1V[0] = 0). In DPD mode, the device responds only to the Resume from DPD command (RES ABh) or Hardware reset (RESET# and IO3_RESET#). All other commands are ignored during DPD mode.

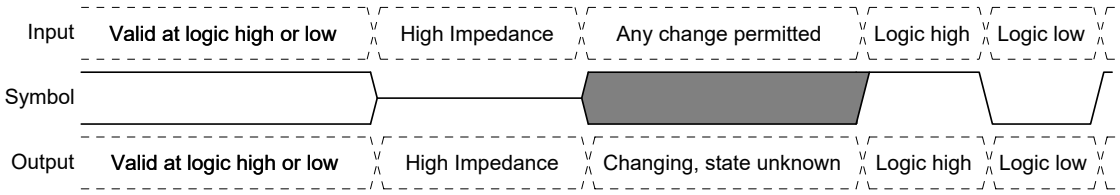
**Table 10. Valid Enter DPD mode and Release from DPD Mode Sequence**

| Current Mode | CS# | SCK | Command | Next Mode | Comments |
|---|---|---|---|---|---|
| Active | LOW to HIGH | N/A | N/A | Standby | |
| Standby | HIGH to LOW | Toggling | B9h<br>Enter DPD | DPD | DPD entered after CS# goes HIGH and after the $t_{DPD}$ duration (see Table 14 on page 34) |
| DPD | HIGH to LOW | Not Toggling | N/A | DPD | If SCK is toggling and Command is not ABh, device remains in DPD |
| | | Toggling | Command not ABh | | |
| DPD | HIGH to LOW | Toggling | ABh<br>Release from DPD | Standby | Release from DPD after CS# goes HIGH and after the $t_{RES}$ duration (see Table 14 on page 34).<br>After CS# goes HIGH to start the release from DPD, it is an invalid sequence to have a CS# transition when the SCK is not toggling. |

# 6. Timing Specifications

## 6.1 Key to Switching Waveforms

**Figure 22. Waveform Element Meanings**




## 6.2 AC Test Conditions

**Figure 23. Test Setup**

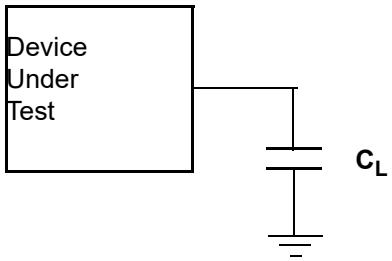



**Table 11. AC Measurement Conditions**

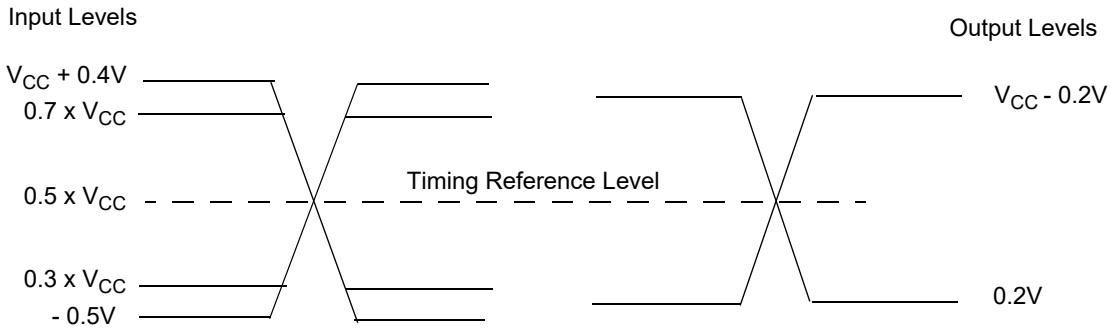
| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $C_L$ | Load Capacitance | | 30 | pF |
| | Input Pulse Voltage | 0.2 x $V_{CC}$ | 0.8 $V_{CC}$ | V |
| | Input slew rate | 0.23 | 1.25 | V/ns |
| | Input Rise and Fall Times | 0.9 | 5 | ns |
| | Input Timing Ref Voltage | 0.5 $V_{CC}$ | | V |
| | Output Timing Ref Voltage | 0.5 $V_{CC}$ | | V |

**Notes**

19.Input slew rate measured from input pulse min to max at $V_{CC}$ max. Example: (1.9V x 0.8) - (1.9V x 0.2) = 1.14V; 1.14V/1.25V/ns = 0.9ns rise or fall time.

20.AC characteristics tables assume clock and data signals have the same slew rate (slope).

**Figure 24. Input, Output, and Timing Reference Levels**



### 6.2.1 Capacitance Characteristics

**Table 12. Capacitance**

| | Parameter | Test Conditions | Package | Min | Max | Unit |
|---|---|---|---|---|---|---|
| $C_{IN}$ | Input Capacitance (applies to SCK, CS#, IO3/RESET#) | 1 MHz | SOIC | | 12.5 | pF |
| | | | LGA, BGA | | 8 | |
| $C_{OUT}$ | Output Capacitance (applies to All I/O) | 1 MHz | SOIC | | 12 | pF |
| | | | LGA, BGA | | 8 | |

## 6.3 Reset

### 6.3.1 Power On (Cold) Reset

The device executes a Power-On Reset (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 20 on page 25, Table 6 on page 24. The device must not be selected (CS# to go high with $V_{CC}$) during power-up ($t_{PU}$), i.e., no commands may be sent to the device until the end of $t_{PU}$.

RESET# and IO3_RESET# are ignored during POR but must be at either a high or low level. If RESET# or IO3_RESET# are low during POR and remains low through and beyond the end of $t_{PU}$, CS# must remain high until $t_{RH}$ after RESET# and IO3_RESET# returns high. RESET# and IO3_RESET# must return high for greater than $t_{RS}$ before returning low to initiate a hardware reset.

The IO3_RESET# input functions as only as the RESET# signal when Quad or QPI mode is not enabled (CR1V[1]=0 or CR2V[6]=0) and when CS# is high for more than $t_{CS}$ time.

**Figure 25. Reset low at the end of POR**

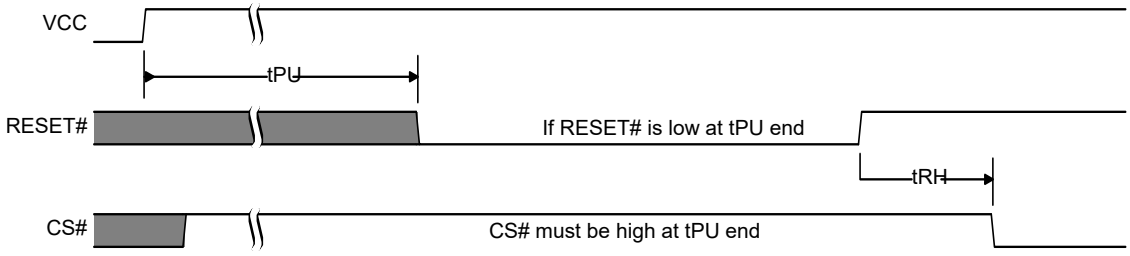



**Figure 26. Reset high at the end of POR**

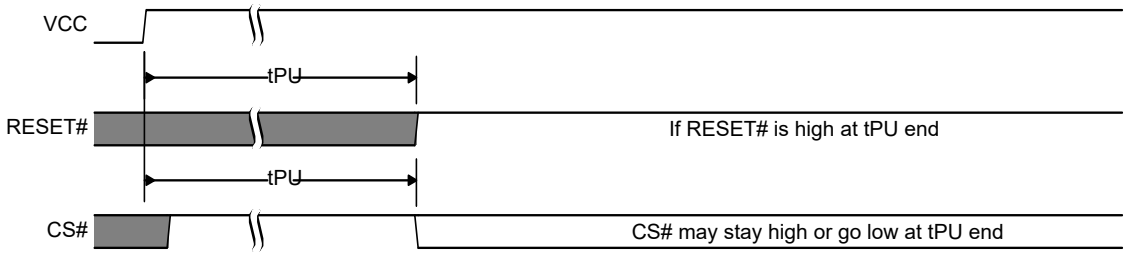
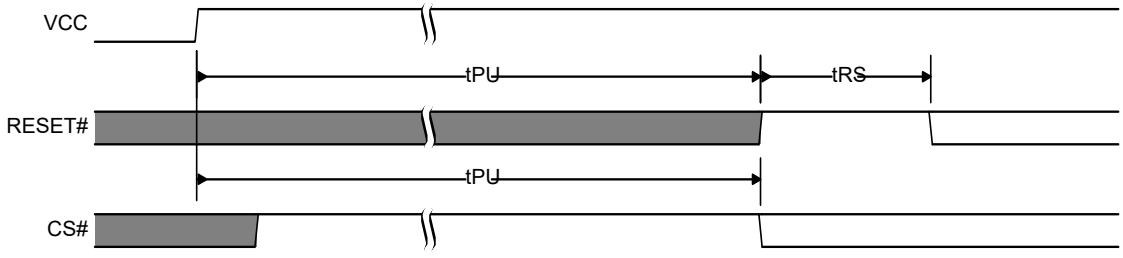



**Figure 27. POR followed by Hardware Reset**



### 6.3.2 RESET # and IO3_RESET# Input Initiated Hardware (Warm) Reset

The RESET# and IO3_RESET# inputs can function as the RESET# signal. Both inputs can initiate the reset operation under conditions.

The RESET# input initiates the reset operation when transitions from $V_{IH}$ to $V_{IL}$ for > $t_{RP}$, the device will reset register states in the same manner as power-on reset but, does not go through the full reset process that is performed during POR. The hardware reset process requires a period of $t_{RPH}$ to complete. The RESET# input is available only on the BGA ball packages.

The IO3_RESET# input initiates the reset operation under the following when CS# is high for more than $t_{CS}$ time or when Quad or QPI Mode is not enabled CR1V[1]=0 or CR2V[6]=0. The IO3_RESET# input has an internal pull-up to $V_{CC}$ and may be left unconnected if Quad or QPI mode is not used. The $t_{CS}$ delay after CS# goes high gives the memory or host system time to drive IO3 high after its use as a Quad or QPI mode I/O signal while CS# was low. The internal pull-up to $V_{CC}$ will then hold IO3_RESET# high until the host system begins driving IO3_RESET#. The IO3_RESET# input is ignored while CS# remains high during $t_{CS}$, to avoid an unintended Reset operation. If CS# is driven low to start a new command, IO3_RESET# is used as IO3.

When the device is not in Quad or QPI mode or, when CS# is high, and IO3_RESET# transitions from $V_{IH}$ to $V_{IL}$ for > $t_{RP}$, following $t_{CS}$, the device will reset register states in the same manner as power-on reset but, does not go through the full reset process that is performed during POR.

The hardware reset process requires a period of $t_{RPH}$ to complete. If the POR process did not complete correctly for any reason during power-up ($t_{PU}$), RESET# going low will initiate the full POR process instead of the hardware reset process and will require $t_{PU}$ to complete the POR process.

The software reset command (RSTEN 66h followed by RST 99h) is independent of the state of RESET # and IO3_RESET#. If RESET# and IO3_RESET# is high or unconnected, and the software reset instructions are issued, the device will perform software reset.

Additional IO3 RESET# notes:

- If both RESET# and IO3_RESET# input options are available use only one reset option in your system. IO3_RESET# input reset operation can be disable by setting CR2NV[7] = 0 (See Table 32 on page 55) setting the IO3_RESET to only operate as IO3. The RESET# input can be disable by not connecting or tying the RESET# input to $V_{IH}$.
- RESET# or IO3_RESET# must be high for $t_{RS}$ following $t_{PU}$ or $t_{RPH}$, before going low again to initiate a hardware reset.
- When IO3_RESET# is driven low for at least a minimum period of time ($t_{RP}$), following $t_{CS}$, the device terminates any operation in progress, makes all outputs high impedance, and ignores all read/write commands for the duration of $t_{RPH}$. The device resets the interface to standby state.
- If Quad or QPI mode and the IO3_RESET# feature are enabled, the host system should not drive IO3 low during $t_{CS,}$ to avoid driver contention on IO3. Immediately following commands that transfer data to the host in Quad or QPI mode, e.g. Quad I/O Read, the memory drives IO3_RESET# high during $t_{CS,}$ to avoid an unintended Reset operation. Immediately following commands that transfer data to the memory in Quad mode, e.g. Page Program, the host system should drive IO3_RESET# high during $t_{CS,}$ to avoid an unintended Reset operation.
- If Quad mode is not enabled, and if CS# is low at the time IO3_RESET# is asserted low, CS# must return high during $t_{RPH}$ before it can be asserted low again after $t_{RH}$.

**Table 13. Hardware Reset Parameters**

| Parameter | Description | Limit | Time | Unit |
|---|---|---|---|---|
| $t_{RS}$ | Reset Setup - Prior Reset end and RESET# high before RESET# low | Min | 50 | ns |
| $t_{RPH}$ | Reset Pulse Hold - RESET# low to CS# low | Min | 35 | µs |
| $t_{RP}$ | RESET# Pulse Width | Min | 200 | ns |
| $t_{RH}$ | Reset Hold - RESET# high before CS# low | Min | 50 | ns |

**Notes**

21. RESET# and IO3_RESET# Low is ignored during Power-up ($t_{PU}$). 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.
22. If Quad mode is enabled, IO3_RESET# Low is ignored during $t_{CS}$.
23. Sum of $t_{RP}$ and $t_{RH}$ must be equal to or greater than $t_{RPH}$.

**Figure 28. Hardware Reset using RESET# Input**

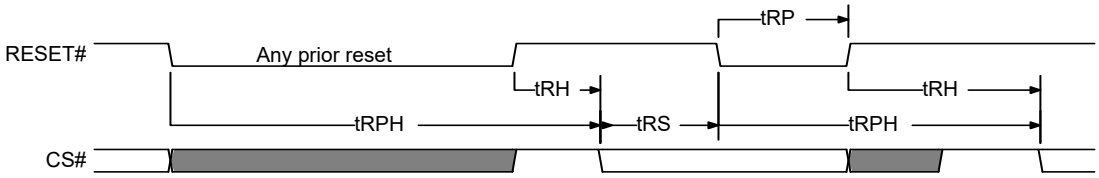



**Figure 29. Hardware Reset when Quad or QPI Mode is not enabled and IO3_RESET# is Enabled**

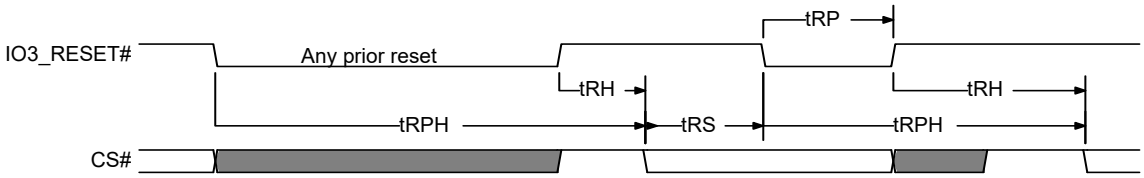
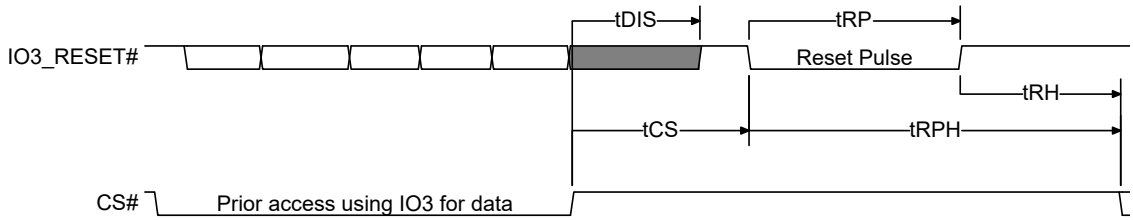



**Figure 30. Hardware Reset when Quad or QPI Mode and IO3_RESET# are Enabled**



## 6.4 SDR AC Characteristics

**Table 14. SDR AC Characteristics**

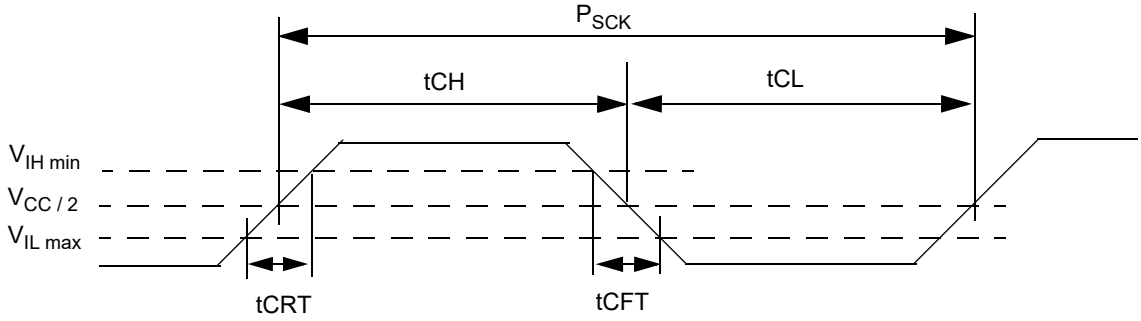
| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $F_{SCK, R}$ | SCK Clock Frequency for READ and 4READ instructions | DC | 50 | MHz |
| $F_{SCK, C}$ | SCK Clock Frequency for the following dual and quad commands: DOR, 4DOR, DIOR, 4DIOR, QOR, 4QOR, QIOR, 4QIOR | DC | 133 | MHz |
| $P_{SCK}$ | SCK Clock Period | 1/ $F_{SCK}$ | | |
| $t_{WH}$, $t_{CH}$ | Clock High Time | 50% $P_{SCK}$ -5% | 50% $P_{SCK}$ +5% | ns |
| $t_{WL}$, $t_{CL}$ | Clock Low Time | 50% $P_{SCK}$ -5% | 50% $P_{SCK}$ +5% | ns |
| $t_{CRT}$, $t_{CLCH}$ | Clock Rise Time (slew rate) | 0.1 | | V/ns |
| $t_{CFT}$, $t_{CHCL}$ | Clock Fall Time (slew rate) | 0.1 | | V/ns |
| $t_{CS}$ | CS# High Time (Read Instructions)<br>CS# High Time (Read Instructions when Reset feature and Quad mode are both enabled)<br>CS# High Time (Program/Erase Instructions) | 10<br>20 (28)<br>50 | | ns |
| $t_{CSS}$ | CS# Active Setup Time (relative to SCK) | 2 | | ns |
| $t_{CSH}$ | CS# Active Hold Time (relative to SCK) | 3 | | ns |
| $t_{SU}$ | Data in Setup Time | 2 | | ns |
| $t_{HD}$ | Data in Hold Time | 3 | | ns |
| $t_V$ | Clock Low to Output Valid | | 8 (25)<br>6 (26)<br>6.5 (26)(29) | ns |
| $t_{HO}$ | Output Hold Time | 1 | | ns |
| $t_{DIS}$ | Output Disable Time (27)<br>Output Disable Time (when Reset feature and Quad mode are both enabled) | | 8<br>20 (28) | ns |
| $t_{WPS}$ | WP# Setup Time (24) | 20 | | ns |
| $t_{WPH}$ | WP# Hold Time (24) | 100 | | ns |
| $t_{DPD}$ | CS# High to Power-down Mode | | 3 | µs |
| $t_{RES}$ | CS# High to Standby Mode without Electronic Signature Read | | 30 | µs |

**Notes**

24. Only applicable as a constraint for WRR or WRAR instruction when SRWD is set to a 1.
25. Full $V_{CC}$ range and CL = 30 pF.
26. Full $V_{CC}$ range and CL = 15 pF.
27. Output HI-Z is defined as the point where data is no longer driven.
28. $t_{CS}$ and $t_{DIS}$ require additional time when the Reset feature and Quad mode are enabled (CR2V[5]=1 and CR1V[1]=1).
29. SOIC package.

### 6.4.1 Clock Timing

**Figure 31. Clock Timing**




### 6.4.2 Input / Output Timing

**Figure 32. SPI Single Bit Input Timing**

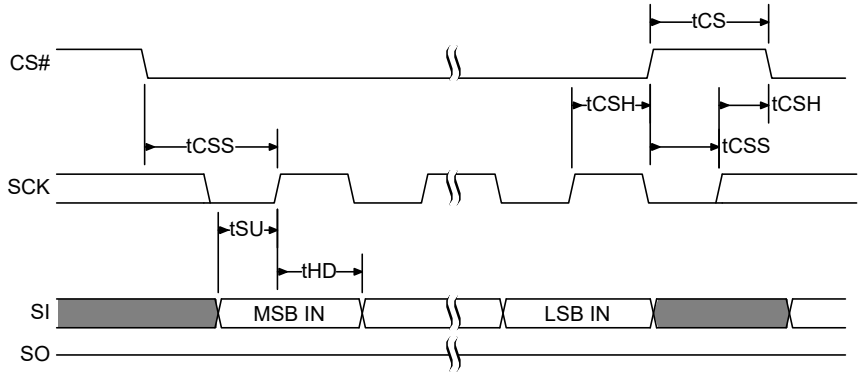



**Figure 33. SPI Single Bit Output Timing**

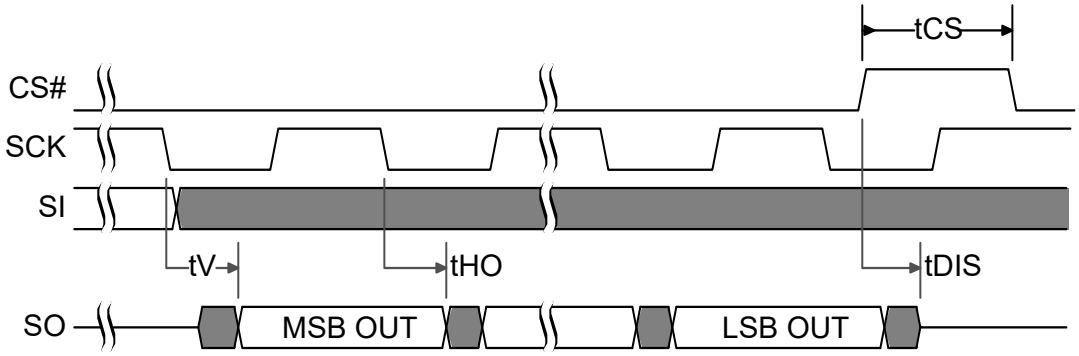


**Figure 34. SDR MIO Timing**

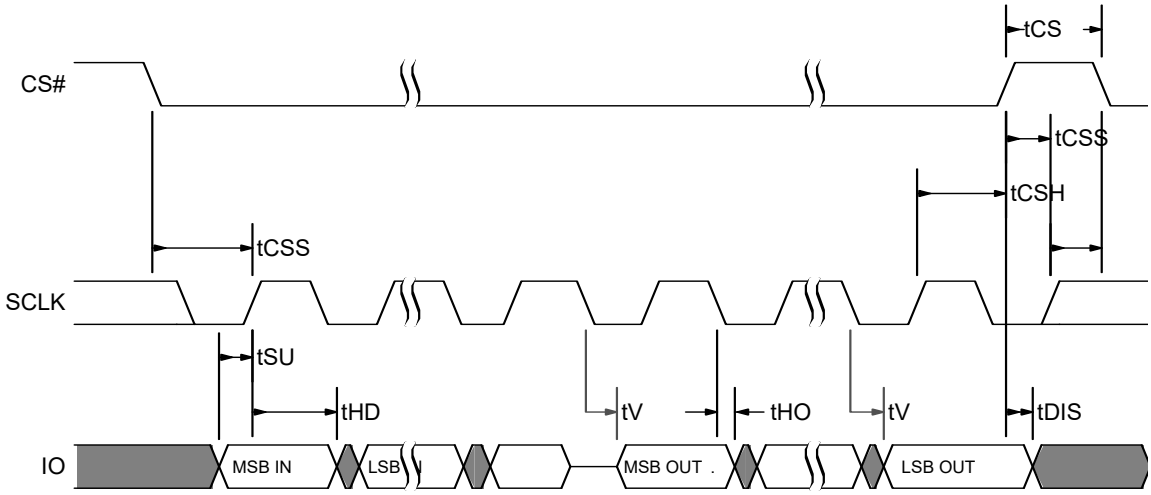
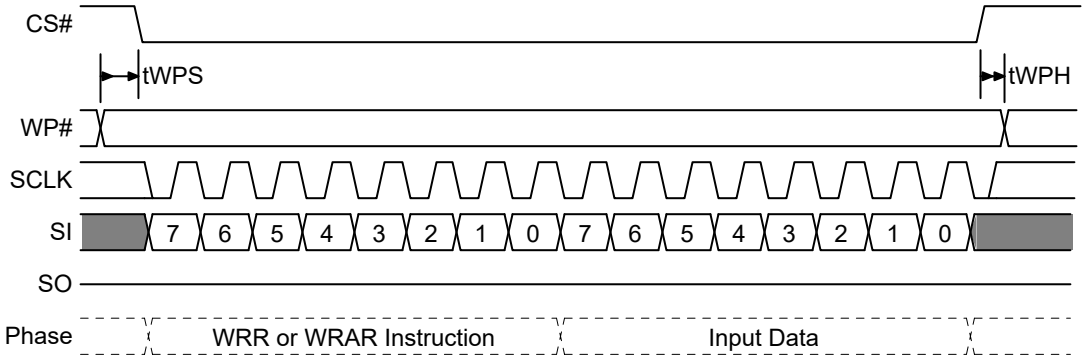



**Figure 35. WP# Input Timing**



## 6.5 DDR AC Characteristics.

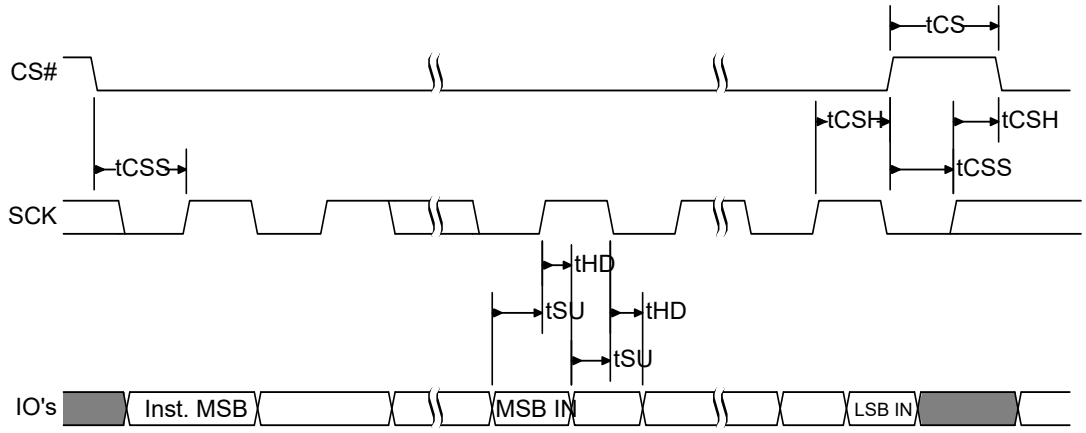
**Table 15. DDR AC Characteristics**

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $F_{SCK, R}$ | SCK Clock Frequency for DDR READ instruction | DC | 80 | MHz |
| $P_{SCK, R}$ | SCK Clock Period for DDR READ instruction | 1/ $F_{SCK}$ | | ns |
| $t_{WH}$, $t_{CH}$ | Clock High Time | 45% $P_{SCK}$ | | ns |
| $t_{WL}$, $t_{CL}$ | Clock Low Time | 45% $P_{SCK}$ | | ns |
| $t_{CS}$ | CS# High Time (Read Instructions)<br>CS# High Time (Read Instructions when Reset feature is enabled) | 10<br>20 | | ns |
| $t_{CSS}$ | CS# Active Setup Time (relative to SCK) | 2 | | ns |
| $t_{CSH}$ | CS# Active Hold Time (relative to SCK) | 3 | | ns |
| $t_{SU}$ | IO in Setup Time | 1.5 | | ns |
| $t_{HD}$ | IO in Hold Time | 1.5 | | ns |
| $t_V$ | Clock Low to Output Valid | 1.5 | 6.0 (30)<br>6.5 (30)(32) | ns |
| $t_{HO}$ | Output Hold Time | 1.5 | | ns |
| $t_{DIS}$ | Output Disable Time<br>Output Disable Time (when Reset feature is enabled) | | 8<br>20 | ns |
| $t_{IO_skew}$ | First IO to last IO data valid time (31) | | 600<br>700(32) | ps |
| $t_{DPD}$ | CS# High to Power-down Mode | | 3 | µs |
| $t_{RES}$ | CS# High to Standby Mode without Electronic Signature Read | | 30 | µs |

**Notes**
30.CL = 15 pF.
31.Not Tested.
32.SOIC package.

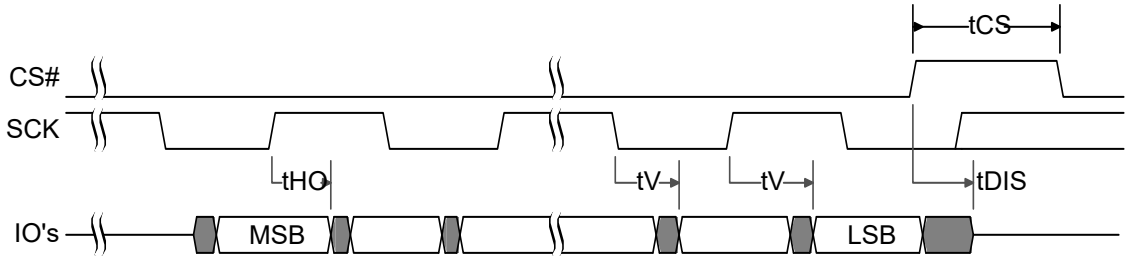
### 6.5.1 *o*DDR Input Timing

**Figure 36. SPI DDR Input Timing**



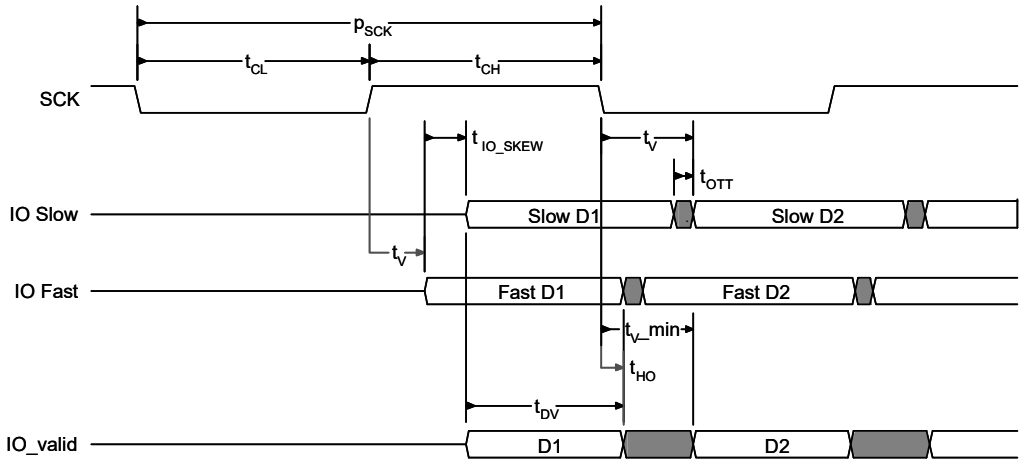
### 6.5.2 DDR Output Timing

**Figure 37. SPI DDR Output Timing**




### 6.5.3 DDR Data Valid Timing Using DLP

**Figure 38. SPI DDR Data Valid Window**




The minimum data valid window ($t_{DV}$) and $t_V$ minimum can be calculated as follows:

$t_{DV}$ = Minimum half clock cycle time ($t_{CLH}$) [33] - $t_{OTT}$ [35] - $t_{IO_SKEW}$ [34]

$t_V$ _min = $t_{HO}$ + $t_{IO_SKEW}$ + $t_{OTT}$

Example:

80 MHz clock frequency = 12.5 ns clock period, DDR operations and duty cycle of 45% or higher

$t_{CLH}$ = 0.45 x PSCK = 0.45 x 12.5 ns = 5.625 ns

Bus impedance of 45 ohm and capacitance of 22 pf, with timing reference of 0.75$V_{CC}$, the rise time from 0 to 1 or fall time 1 to 0 is 1.4[38] x RC time constant (Tau)[37] = 1.4 x 0.99 ns = 1.39 ns

$t_{OTT}$ = rise time + fall time = 1.39 ns + 1.39 ns = 2.78 ns.

Data Valid Window

$t_{DV}$ = $t_{CLH}$ - $t_{IO_SKEW}$ - $t_{OTT}$ = 5.625 ns - 400 ps - 2.78 ns = 2.45 ns

$t_V$ Minimum

$t_V$ _min = $t_{HO}$ + $t_{IO_SKEW}$ + $t_{OTT}$ = 1.0 ns + 400 ps + 2.78 ns = 4.38 ns

**Notes**

33. $t_{CLH}$ is the shorter duration of $t_{CL}$ or $t_{CH}$.
34. $t_{IO_SKEW}$ is the maximum difference (delta) between the minimum and maximum $t_V$ (output valid) across all IO signals.
35. $t_{OTT}$ is the maximum Output Transition Time from one valid data value to the next valid data value on each IO. $t_{OTT}$ is dependent on system level considerations including:
    a. Memory device output impedance (drive strength).
    b. System level parasitics on the IOs (primarily bus capacitance).
    c. Host memory controller input $V_{IH}$ and $V_{IL}$ levels at which 0 to 1 and 1 to 0 transitions are recognized.
    d. $t_{OTT}$ is not a specification tested by Cypress, it is system dependent and must be derived by the system designer based on the above considerations.
36. $t_{DV}$ is the data valid window.
37. Tau = R (Output Impedance) x C (Load capacitance).
38. Multiplier of Tau time for voltage to rise to 75% of $V_{CC}$.

# 7. Embedded Algorithm Performance Tables

**Table 16. Program and Erase Performance**

| Symbol | Parameter | Min | Typ (39) | Max | Unit |
|---|---|---|---|---|---|
| $t_W$ | Non-volatile Register Write Time | | 240 | 750 | ms |
| $t_{PP}$ | Page Programming (512 Bytes)<br>Page Programming (256Bytes) | | 475<br>360 | 2000<br>2000 | µs |
| $t_{SE}$ | Sector Erase Time (64KB or 4KB physical sectors) | | 240 | 725 | ms |
| | Sector Erase Time (256KB logical sectors = 4x64K physical sectors) | | 930 | 2900 | ms |
| $t_{BE}$ | Bulk Erase Time (S25FS064S) | | 30 | 94 | sec |
| tEES | Evaluate Erase Status Time (64 KB or 4KB physical sectors) | | 20 | 25 | µs |
| | Evaluate Erase Status Time (256 KB physical or logical sectors) | | 80 | 100 | |

**Notes**

39. Typical program and erase times assume the following conditions: 25 °C, $V_{CC}$ = 1.8 V; random data pattern.
40. The programming time for any OTP programming command is the same as $t_{PP}$. This includes OTPP 42h, PNVDLR 43h, ASPP 2Fh, and PASSP E8h.
41. The programming time for the PPBP E3h command is the same as $t_{PP}$.. The erase time for PPBE E4h command is the same as $t_{SE}$.

**Table 17. Program or Erase Suspend AC Parameters**

| Parameter | Typical | Max | Unit | Comments |
|---|---|---|---|---|
| Suspend Latency ($t_{SL}$) | | 50 | µs | The time from Suspend command until the WIP bit is 0 |
| Resume to next Program Suspend ($t_{RS}$) | 100 | | µs | The time needed to issue the next Suspend command but ≥ typical periods are needed for Program or Erase to progress to completion. |

# 8. Physical Interface

## 8.1 Connection Diagrams

### 8.1.1 8 Connector Packages

**Figure 39. 8-Pin Plastic Small Outline Package (SOIC8)**

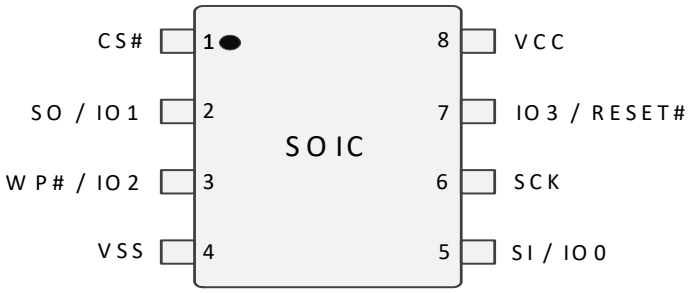
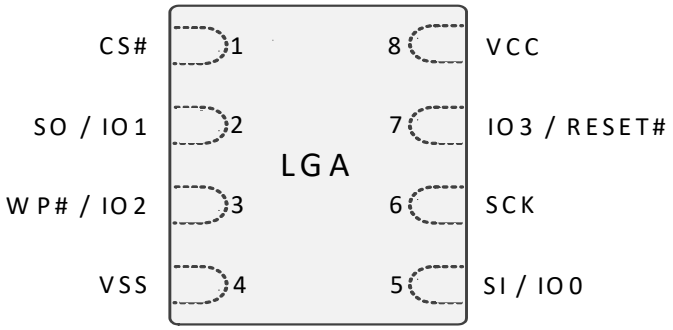


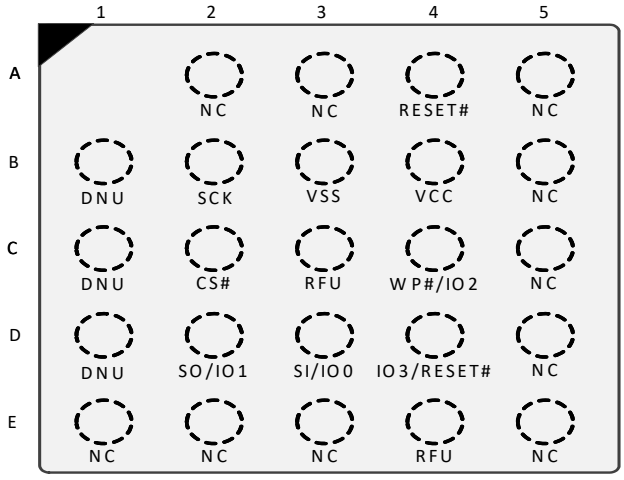

**Figure 40. 8-Pad LGA 5x6 (W9A008), Top View**




**Note**
42.The RESET# input has an internal pull-up and may be left unconnected in the system if quad mode and hardware reset are not in use.

### 8.1.2 BGA Ball Footprint

**Figure 41. 24-Ball BGA, 5x5 Ball Footprint (FAB024), Top View**



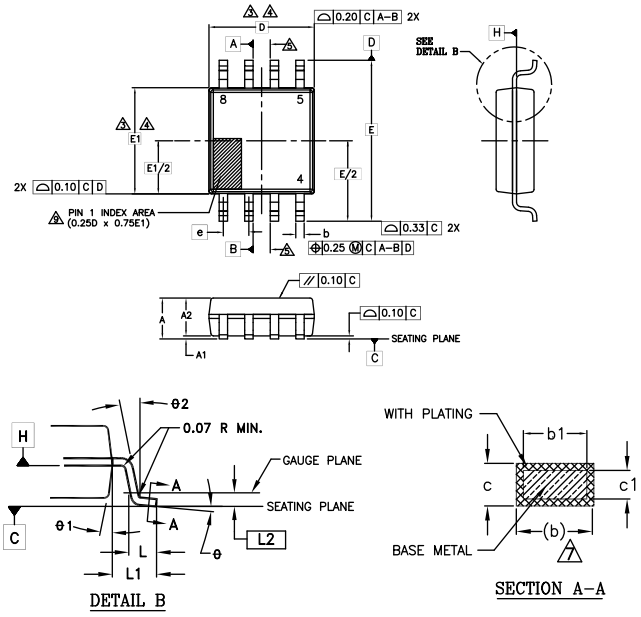

**Note**
43.The RESET# input has an internal pull-up and may be left unconnected in the system if quad mode and hardware reset are not in use.

### 8.1.3 Special Handling Instructions for FBGA Packages

Flash memory devices in BGA packages may be damaged if exposed to ultrasonic cleaning methods. The package and/or data integrity may be compromised if the package body is exposed to temperatures above 150°C for prolonged periods of time.

## 8.2 Physical Diagrams

### 8.2.1 SOIC 8-Lead, 208 mil Body Width (SOC008)




| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| A | 1.75 | - | 2.16 |
| A1 | 0.05 | - | 0.25 |
| A2 | 1.70 | - | 1.90 |
| b | 0.36 | - | 0.48 |
| b1 | 0.33 | - | 0.46 |
| c | 0.19 | - | 0.24 |
| c1 | 0.15 | - | 0.20 |
| D | 5.28 BSC | | |
| E | 8.00 BSC | | |
| E1 | 5.28 BSC | | |
| e | 1.27 BSC | | |
| L | 0.51 | - | 0.76 |
| L1 | 1.36 REF | | |
| L2 | 0.25 BSC | | |
| N | 8 | | |
| θ | 0° | - | 8° |
| θ 1 | 5° | - | 15° |
| θ 2 | 0-8° REF | | |

NOTES:

1. ALL DIMENSIONS ARE IN MILLIMETERS.
2. DIMENSIONING AND TOLERANCING PER ASME Y14.5M - 1994.
3. DIMENSION D DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS OR GATE BURRS SHALL NOT EXCEED 0.15 mm PER END. DIMENSION E1 DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD FLASH OR PROTRUSION SHALL NOT EXCEED 0.25 mm PER SIDE. D AND E1 DIMENSIONS ARE DETERMINED AT DATUM H.
4. THE PACKAGE TOP MAY BE SMALLER THAN THE PACKAGE BOTTOM. DIMENSIONS D AND E1 ARE DETERMINED AT THE OUTMOST EXTREMES OF THE PLASTIC BODY EXCLUSIVE OF MOLD FLASH, TIE BAR BURRS, GATE BURRS AND INTERLEAD FLASH, BUT INCLUSIVE OF ANY MISMATCH BETWEEN THE TOP AND BOTTOM OF THE PLASTIC BODY.
5. DATUMS A AND B TO BE DETERMINED AT DATUM H.
6. "N" IS THE MAXIMUM NUMBER OF TERMINAL POSITIONS FOR THE SPECIFIED PACKAGE LENGTH.
7. THE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.10 TO 0.25 mm FROM THE LEAD TIP.
8. DIMENSION "b" DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.10 mm TOTAL IN EXCESS OF THE "b" DIMENSION AT MAXIMUM MATERIAL CONDITION. THE DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OF THE LEAD FOOT.
9. THIS CHAMFER FEATURE IS OPTIONAL. IF IT IS NOT PRESENT, THEN A PIN 1 IDENTIFIER MUST BE LOCATED WITHIN THE INDEX AREA INDICATED.
10. LEAD COPLANARITY SHALL BE WITHIN 0.10 mm AS MEASURED FROM THE SEATING PLANE.

THIS DRAWING CONTAINS INFORMATION WHICH IS THE PROPRIETARY PROPERTY OF CYPRESS SEMICONDUCTOR CORPORATION. THIS DRAWING IS RECEIVED IN CONFIDENCE AND ITS CONTENTS MAY NOT BE DISCLOSED WITHOUT WRITTEN CONSENT OF CYPRESS SEMICONDUCTOR CORPORATION.

| PACKAGE CODE(S) | SOC008 | | | DRAWN BY KOTA | DATE 18-JUL-16 |
|---|---|---|---|---|---|
| | | | | APPROVED BY BESY | DATE 18-JUL-16 |

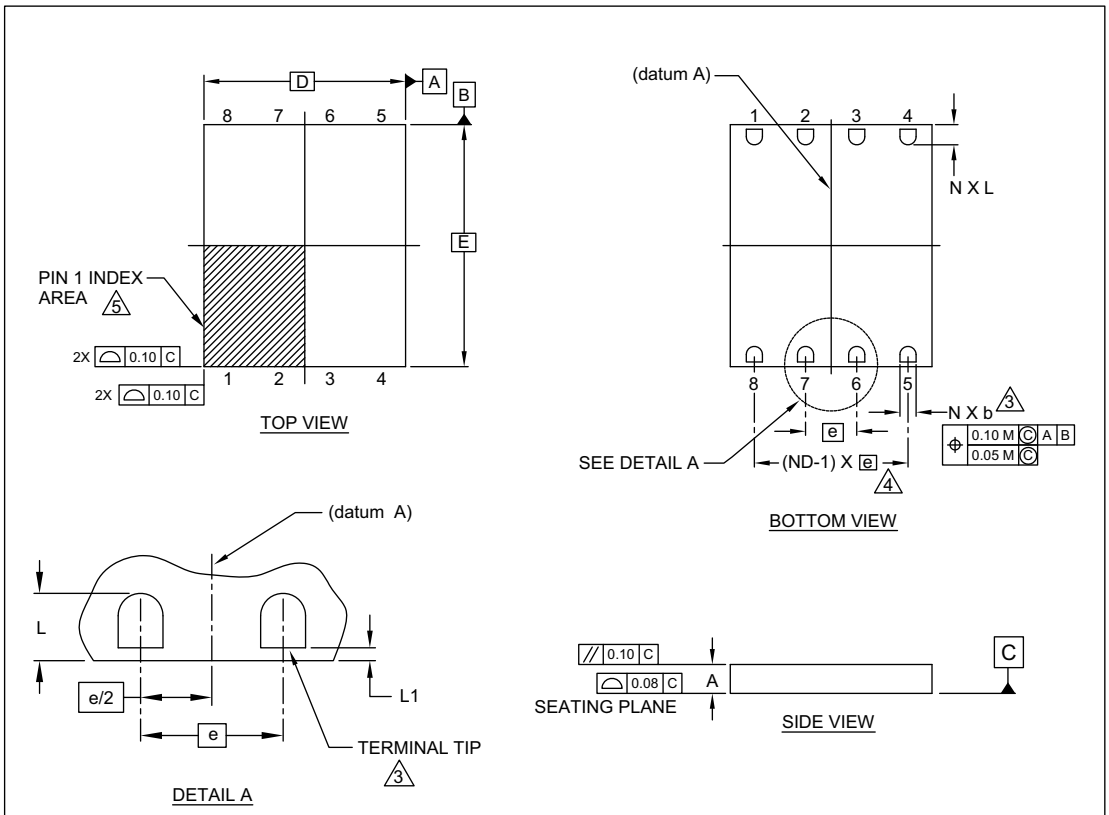
CYPRESS Company Confidential

TITLE: PACKAGE OUTLINE, 8 LEAD SOIC 5.28X5.28X2.16 MM SOC008

SPEC NO. 002-15548 REV **

SCALE : TO FIT SHEET 1 OF 2

### 8.2.2 LGA 8-contact 5 x 6 mm (W9A008)




| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| e | | 1.27 BSC | |
| N | | 8 | |
| ND | | 4 | |
| L | 0.45 | 0.50 | 0.55 |
| b | 0.35 | 0.40 | 0.45 |
| D | | 5.00 BSC | |
| E | | 6.00 BSC | |
| A | 0.70 | 0.75 | 0.80 |
| L1 | 0.00 | - | 0.15 |

NOTES:

1. ALL DIMENSIONS ARE IN MILLIMETERS.
2. N IS THE TOTAL NUMBER OF LANDS.
3. DIMENSION "b" IS MEASURED AT THE MAXIMUM LAND WIDTH IN A PLANE PARALLEL TO DATUM C.
4. ND REFERS TO THE NUMBER OF LANDS ON D SIDE.
5. PIN #1 ID ON TOP WILL BE LOCATED WITHIN THE INDICATED ZONE.

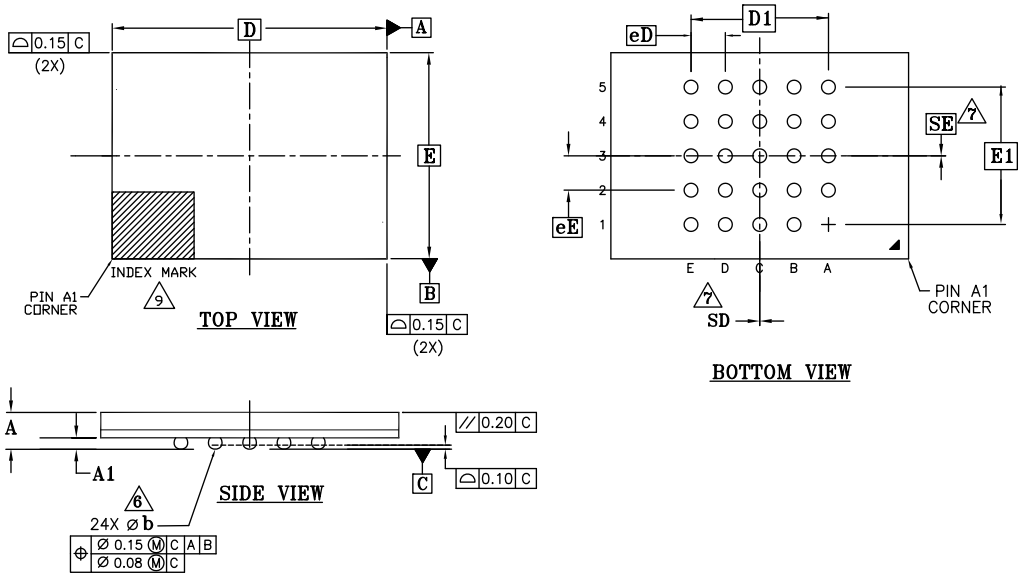
CYPRESS Company Confidential

THIS DRAWING CONTAINS INFORMATION WHICH IS THE PROPRIETARY PROPERTY OF CYPRESS SEMICONDUCTOR CORPORATION. THIS DRAWING IS RECEIVED IN CONFIDENCE AND ITS CONTENTS MAY NOT BE DISCLOSED WITHOUT WRITTEN CONSENT OF CYPRESS SEMICONDUCTOR CORPORATION.

| PACKAGE CODE(S) | W9A008 | | | DRAWN BY KOTA | DATE 29-NOV-16 |
|---|---|---|---|---|---|
| | | | | APPROVED BY BESY | DATE 29-NOV-16 |

TITLE: PACKAGE OUTLINE, 8 LEAD LGA 5.00X6.00X0.80 MM W9A008
SPEC NO. 002-10839 REV. *B
SCALE : TO FIT SHEET 1 of 2

### 8.2.3 Ball Grid Array 24-ball 6 x 8 mm (FAB024)




| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| A | - | - | 1.20 |
| 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.

CYPRESS Company Confidential

THIS DRAWING CONTAINS INFORMATION WHICH IS THE PROPRIETARY PROPERTY OF CYPRESS SEMICONDUCTOR CORPORATION. THIS DRAWING IS RECEIVED IN CONFIDENCE AND ITS CONTENTS MAY NOT BE DISCLOSED WITHOUT WRITTEN CONSENT OF CYPRESS SEMICONDUCTOR CORPORATION.

| PACKAGE CODE(S) | FAB024 | DRAWN BY KOTA | DATE 18-JUL-16 |
|---|---|---|---|
| | | APPROVED BY BESY | DATE 18-JUL-16 |

TITLE: PACKAGE OUTLINE, 24 BALL FBGA 8.0X6.0X1.2 MM FAB024
SPEC NO.: 002-15534 REV: **
SCALE : TO FIT SHEET 1 OF 2

# Software Interface

This section discusses the features and behaviors most relevant to host system software that interacts with FS-S Family memory devices.

## 9. Address Space Maps

### 9.1 Overview

#### 9.1.1 Extended Address

The FS-S Family supports 32 bit (4 Byte) addresses to enable higher density devices than allowed by previous generation (legacy) SPI devices that supported only 24 bit (3 Byte) addresses. A 24 bit, byte resolution, address can access only 16 MBytes (128 Mbits) maximum density. A 32 bit, byte resolution, address allows direct addressing of up to a 4 GBytes (32 Gbits) address space and allows for software compatibility for device from 4 MBytes (32 Mbits) to 4 GBytes (32 Gbits).

Legacy commands continue to support 24 bit addresses for backward software compatibility. Extended 32 bit addresses are enabled in two ways:

- Extended address mode — a volatile configuration register bit that changes all legacy commands to expect 32 bits of address supplied from the host system.
- 4 Byte address commands — that perform both legacy and new functions, which always expect 32 bit address.

The default condition for extended address mode, after power-up or reset, is controlled by a non-volatile configuration bit. The default extended address mode may be set for 24 or 32 bit addresses. This enables legacy software compatible access to the first 128 Mbits of a device or for the device to start directly in 32 bit address mode.

The 64Mb density member of the FS-S Family supports the extended address features in the same way but in essence ignores bits 31 to 23 or 22 of any address because the main Flash array only needs 23 or 22 bits of address. This enables simple migration from the 64Mb density to higher density devices without changing the address handling aspects of software.

#### 9.1.2 Multiple Address Spaces

Many commands operate on the main Flash memory array. Some commands operate on address spaces separate from the main Flash array. Each separate address space uses the full 24 or 32 bit address but may only define a small portion of the available address space.

### 9.2 Flash Memory Array

The main Flash array is divided into erase units called physical sectors.

The FS-S family physical sectors may be configured as a hybrid combination of eight 4KB parameter sectors at the top or bottom of the address space with all but one of the remaining sectors being uniform size. Because the group of eight 4KB parameter sectors is in total smaller than a uniform sector, the group of 4KB physical sectors respectively overlay (replace) the top or bottom 32KB of the highest or lowest address uniform sector.

The parameter sector erase commands (20h or 21h) must be used to erase the 4KB sectors individually. The sector (uniform block) erase commands (D8h or DCh) must be used to erase any of the remaining sectors, including the portion of highest or lowest address sector that is not overlaid by the parameter sectors. The uniform block erase command has no effect on parameter sectors.

Configuration register 1 non-volatile bit 2 (CR1NV[2]) equal to 0 overlays the parameter sectors at the bottom of the lowest address uniform sector. CR1NV[2] = 1 overlays the parameter sectors at the top of the highest address uniform sector. See Section 9.6 Registers on page 48 for more information.

There is also a configuration option to remove the 4KB parameter sectors from the address map so that all sectors are uniform size. Configuration register 3 volatile bit 3 (CR3V[3]) equal to 0 selects the hybrid sector architecture with 4KB parameter sectors. CR3V[3]=1 selects the uniform sector architecture without parameter sectors. Uniform physical sectors are:

- 64KB or 256KB

The devices also may be configured to use the sector (uniform block) erase commands to erase 256KB logical blocks rather than individual 64KB physical sectors. This configuration option (CR3V[1]=1) allows lower density devices to emulate the same sector erase behavior as higher density members of the family that use 256KB physical sectors. This can simplify software migration to the higher density members of the family.

**Table 18. S25FS064S Sector and Memory Address Map, Bottom 4 KBytes Sectors**

| Sector Size (KByte) | Sector Count | Sector Range | Address Range (Byte Address) | Notes |
|---|---|---|---|---|
| 4 | 8 | SA00 | 00000000h-00000FFFh | Sector Starting Address — Sector Ending Address |
| | | : | : | |
| | | SA07 | 00007000h-00007FFFh | |
| 32 | 1 | SA08 | 00008000h-0000FFFFh | |
| 64 | 127 | SA09 | 00010000h-0001FFFFh | |
| | | : | : | |
| | | SA135 | 007F0000h-007FFFFFh | |

**Table 19. S25FS064S Sector and Memory Address Map, Top 4 KBytes Sectors**

| Sector Size (KByte) | Sector Count | Sector Range | Address Range (Byte Address) | Notes |
|---|---|---|---|---|
| 64 | 127 | SA00 | 0000000h-000FFFFh | Sector Starting Address — Sector Ending Address |
| | | : | : | |
| | | SA126 | 007E0000h-007EFFFFh | |
| 32 | 1 | SA127 | 007F0000h - 007F7FFFh | |
| 4 | 8 | SA128 | 007F8000h - 007F8FFFh | |
| | | : | : | |
| | | SA135 | 007FF000h-007FFFFFh | |

**Table 20. S25FS064S Sector and Memory Address Map, Uniform 64 KBytes Blocks**

| Sector Size (KByte) | Sector Count | Sector Range | Address Range (Byte Address) | Notes |
|---|---|---|---|---|
| 64 | 128 | SA00 | 0000000h-0000FFFFh, | Sector Starting Address — Sector Ending Address |
| | | : | : | |
| | | SA127 | 007F0000h-07FFFFFh | |

**Table 21. S25FS064S Sector Address Map, Bottom 4 KB Sectors, 256 KB Logical Uniform Sectors**

| Sector Size (KByte) | Sector Count | Sector Range | Address Range (Byte Address) | Notes |
|---|---|---|---|---|
| 4 | 8 | SA00 | 00000000h-00000FFFh | Sector Starting Address — Sector Ending Address |
| | | : | : | |
| | | SA07 | 00007000h-00007FFFh | |
| 224 | 1 | SA08 | 00008000h-0003FFFFh | |
| 256 | 31 | SA09 | 00040000h-0007FFFFh | |
| | | : | : | |
| | | SA39 | 007C0000h-007FFFFFh | |

**Table 22. S25FS064S Sector Address Map, Top 4 KB Sectors, 256 KB Logical Uniform Sectors**

| Sector Size (KByte) | Sector Count | Sector Range | Address Range (Byte Address) | Notes |
|---|---|---|---|---|
| 256 | 31 | SA00 | 00000000h-0003FFFFh | Sector Starting Address — Sector Ending Address |
| | | : | : | |
| | | SA30 | 00780000h-007BFFFFh | |
| 224 | 1 | SA31 | 007C0000h-007F7FFFh | |
| 4 | 8 | SA32 | 007F8000h-007F8FFFh | |
| | | : | : | |
| | | SA39 | 007FF000h-007FFFFFh | |

**Table 23. S25FS064S Sector and Memory Address Map, Uniform 256 KBytes Blocks**

| Sector Size (KByte) | Sector Count | Sector Range | Address Range (Byte Address) | Notes |
|---|---|---|---|---|
| 256 | 32 | SA00 | 00000000h-0003FFFFh | Sector Starting Address — Sector Ending Address |
| | | : | : | |
| | | SA31 | 007C0000h-007FFFFFh | |

**Note**: These are condensed tables that use a couple of sectors as references. There are address ranges that are not explicitly listed. All 4KB sectors have the pattern XXXX000h-XXXXFFFh. All 64KB sectors have the pattern XXX0000h-XXXFFFFh. All 256 KB sectors have the pattern XX00000h-XX3FFFFh, XX40000h-XX7FFFFh, XX80000h-XXCFFFFh, or XXD0000h-XXFFFFFh.

## 9.3 ID-CFI Address Space

The RDID command (9Fh) reads information from a separate Flash memory address space for device identification (ID) and Common Flash Interface (CFI) information. See Device ID and Common Flash Interface (ID-CFI) Address Map — Standard on page 122 for the tables defining the contents of the ID-CFI address space. The ID-CFI address space is programmed by Cypress and read-only for the host system.

### 9.3.1 Cypress Programmed Unique ID

A 64-bit unique number is located in 8 bytes of the Unique Device ID address space. This Unique ID may be used as a software readable serial number that is unique for each device.

## 9.4 JEDEC JESD216 Serial Flash Discoverable Parameters (SFDP) Space

The RSFDP command (5Ah) reads information from a separate Flash memory address space for device identification, feature, and configuration information, in accord with the JEDEC JESD216 Rev B standard for Serial Flash Discoverable Parameters. The ID-CFI address space is incorporated as one of the SFDP parameters. See Software Interface Reference on page 122 for the tables defining the contents of the SFDP address space. The SFDP address space is programmed by Cypress and read-only for the host system.

## 9.5 OTP Address Space

Each FS-S Family memory device has a 1024 byte One Time Program (OTP) address space that is separate from the main Flash array. The OTP 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 zeros in these bytes. Programming ones in these byte locations is ignored and does not affect the value programmed by Cypress. Attempting to program any zero in these byte locations will fail and set P_ERR.
- The next 4 higher address bytes (OTP Lock Bytes) are used to provide one bit per OTP region to permanently protect each region from programming. The bytes are erased when shipped from Cypress. After an OTP region is programmed, it can be locked to prevent further programming, by programming the related protection bit in the OTP Lock Bytes.
- The next higher 12 bytes of the lowest address region are Reserved for Future Use (RFU). The bits in these RFU bytes may be programmed by the host system but it must be understood that a future device may use those bits for protection of a larger OTP space. The bytes are erased when shipped from Cypress.

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

Refer to Figure 42 on page 47 for a pictorial representation of the OTP memory space.

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

The configuration register FREEZE (CR1V[0]) bit protects the entire OTP memory space from programming when set to “1”. This allows trusted boot code to control programming of OTP regions then set the FREEZE bit to prevent further OTP memory space programming during the remainder of normal power-on system operation.

**Figure 42. OTP Address Space**

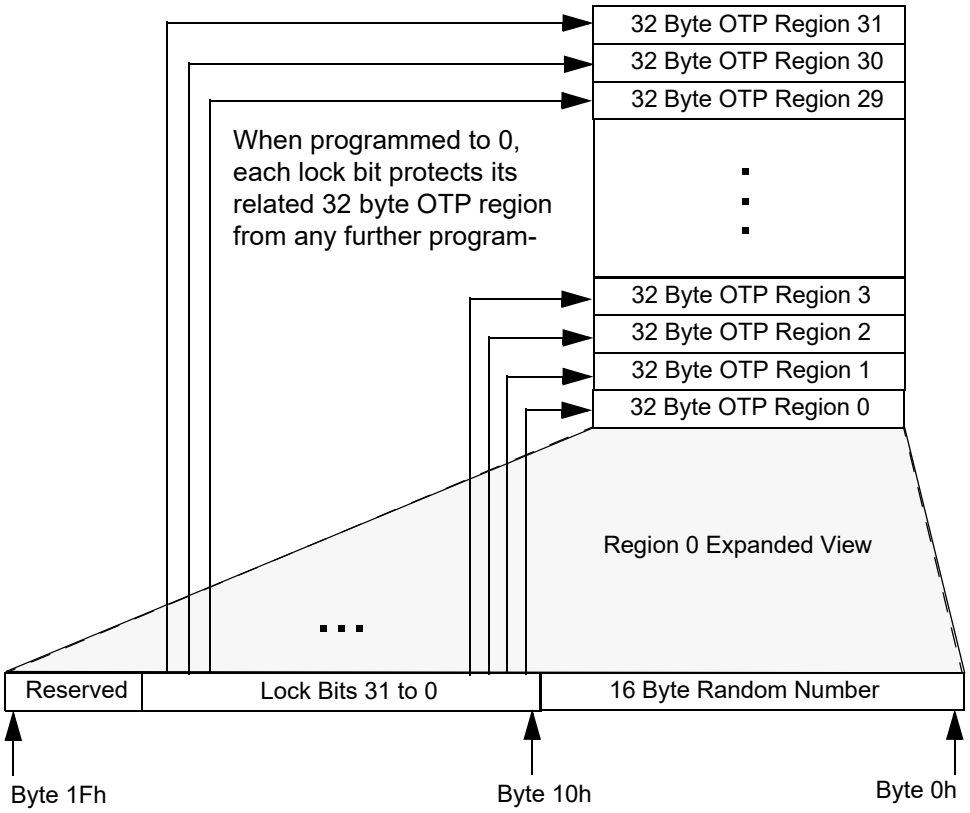


**Table 25. OTP Address Map**

| Region | Byte Address Range (Hex) | Contents | Initial Delivery State (Hex) |
|---|---|---|---|
| Region 0 | 000 | Least Significant Byte of Cypress Programmed Random Number | Cypress Programmed Random Number |
| | ... | ... | |
| | 00F | Most Significant Byte of Cypress Programmed Random Number | |
| | 010 to 013 | Region Locking Bits<br>Byte 10 [bit 0] locks region 0 from programming when = 0<br>...<br>Byte 13 [bit 7] locks region 31from programming when = 0 | All Bytes = FF |
| | 014 to 01F | Reserved for Future Use (RFU) | All Bytes = FF |
| Region 1 | 020 to 03F | Available for User Programming | All Bytes = FF |
| Region 2 | 040 to 05F | Available for User Programming | All Bytes = FF |
| ... | ... | Available for User Programming | All Bytes = FF |
| Region 31 | 3E0 to 3FF | Available for User Programming | All Bytes = FF |

## 9.6 Registers

Registers are small groups of memory cells used to configure how the FS-S Family memory device operates or to report the status of device operations. The registers are accessed by specific commands. The commands (and hexadecimal instruction codes) used for each register are noted in each register description.

In legacy SPI memory devices the individual register bits could be a mixture of volatile, non-volatile, or One Time Programmable (OTP) bits within the same register. In some configuration options the type of a register bit could change e.g. from non-volatile to volatile.

The FS-S Family uses separate non-volatile or volatile memory cell groups (areas) to implement the different register bit types. However, the legacy registers and commands continue to appear and behave as they always have for legacy software compatibility. There is a non-volatile and a volatile version of each legacy register when that legacy register has volatile bits or when the command to read the legacy register has zero read latency. When such a register is read the volatile version of the register is delivered. During Power-On Reset (POR), hardware reset, or software reset, the non-volatile version of a register is copied to the volatile version to provide the default state of the volatile register. When non-volatile register bits are written the non-volatile version of the register is erased and programmed with the new bit values and the volatile version of the register is updated with the new contents of the non-volatile version. When OTP bits are programmed the non-volatile version of the register is programmed and the appropriate bits are updated in the volatile version of the register. When volatile register bits are written, only the volatile version of the register has the appropriate bits updated.

The type for each bit is noted in each register description. The default state shown for each bit refers to the state after power-on reset, hardware reset, or software reset if the bit is volatile. If the bit is non-volatile or OTP, the default state is the value of the bit when the device is shipped from Cypress. Non-volatile bits have the same cycling (erase and program) endurance as the main Flash array.

**Table 26. Register Descriptions**

| Register | Type | Bits | Abbreviation |
|---|---|---|---|
| Status Register-1 | Non-volatile | 7:0 | SR1NV[7:0] |
| | Volatile | 7:0 | SR1V[7:0] |
| Status Register-2 | Volatile | 7:0 | SR2V[7:0] |
| Configuration Register-1 | Non-volatile/OTP | 7:0 | CR1NV[7:0] |
| | Volatile | 7:0 | CR1V[7:0] |
| Configuration Register-2 | Non-volatile/OTP | 7:0 | CR2NV[7:0] |
| | Volatile | 7:0 | CR2V[7:0] |
| Configuration Register-3 | Non-volatile/OTP | 7:0 | CR3NV[7:0] |
| | Volatile | 7:0 | CR3V[7:0] |
| Configuration Register-4 | Non-volatile/OTP | 7:0 | CR4NV[7:0] |
| | Volatile | 7:0 | CR4V[7:0] |
| ECC Status Register | Volatile Read Only | 7:0 | ECCSRV[7:0] |
| ASP Register | OTP | 15:0 | ASPR[15:0] |
| Password Register | OTP | 63:0 | PASS[63:0] |
| PPB Lock Register | Volatile Read Only | 7:0 | PPBL[7:0] |
| PPB Access Register | Non-volatile | 7:0 | PPBAR[7:0] |
| DYB Access Register | Volatile | 7:0 | DYBAR[7:0] |
| SPI DDR Data Learning Registers | OTP | 7:0 | NVDLR[7:0] |
| | Volatile | 7:0 | VDLR[7:0] |

### 9.6.1 Status Registers 1

#### 9.6.1.1 Status Register 1 Non-Volatile (SR1NV)

Related Commands: Write Registers (WRR 01h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h)

**Table 27. Status Register-1 Non-Volatile (SR1NV)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | SRWD_NV | Status Register Write Disable Default | Non-Volatile | 0 | 1 = Locks state of SRWD, BP, and configuration register-1 bits when WP# is low by not executing WRR or WRAR commands that would affect SR1NV, SR1V, CR1NV, or CR1V.<br>0 = No protection, even when WP# is low |
| 6 | P_ERR_D | Programming Error Default | Non-Volatile Read Only | 0 | Provides the default state for the Programming Error Status. Not user programmable. |
| 5 | E_ERR_D | Erase Error Default | Non-Volatile Read Only | 0 | Provides the default state for the Erase Error Status. Not user programmable. |
| 4 | BP_NV2 | Block Protection Non-Volatile | Non-Volatile | 000b | Protects the selected range of sectors (Block) from Program or Erase when the BP bits are configured as non-volatile (CR1NV[3]=0). Programmed to 111b when BP bits are configured to volatile (CR1NV[3]=1).- after which these bits are no longer user programmable. |
| 3 | BP_NV1 | | | | |
| 2 | BP_NV0 | | | | |
| 1 | WEL_D | WEL Default | Non-Volatile Read Only | 0 | Provides the default state for the WEL Status. Not user programmable. |
| 0 | WIP_D | WIP Default | Non-Volatile Read Only | 0 | Provides the default state for the WIP Status. Not user programmable. |

**Status Register Write Non-volatile (SRWD_NV) SR1NV[7]**: Places the device in the Hardware Protected mode when this bit is set to 1 and the WP# input is driven low. In this mode, the Write Registers (WRR) and Write Any Register (WRAR) commands (that select status register-1 or configuration register-1) are ignored and not accepted for execution, effectively locking the state of the Status Register-1 and Configuration Register-1 (SR1NV, SR1V, CR1NV, or CR1V) bits, by making the registers read-only. If WP# is high, Status Register-1 and Configuration Register-1 may be changed by the WRR or WRAR commands. If SRWD_NV is 0, WP# has no effect and Status Register-1 and Configuration Register-1 may be changed by the WRR or WRAR commands. WP# has no effect on the writing of any other registers. The SRWD_NV bit has the same non-volatile endurance as the main Flash array. The SRWD (SR1V[7]) bit serves only as a copy of the SRWD_NV bit to provide zero read latency.

**Program Error Default (P_ERR_D) SR1NV[6]**: Provides the default state for the Programming Error Status in SR1V[6]. This bit is not user programmable.

**Erase Error (E_ERR) SR1V[5]**: Provides the default state for the Erase Error Status in SR1V[5]. This bit is not user programmable.

**Block Protection (BP_NV2, BP_NV1, BP_NV0) SR1NV[4:2]**: These bits define the main Flash array area to be software-protected against program and erase commands. The BP bits are selected as either volatile or non-volatile, depending on the state of the BP non-volatile bit (BPNV_O) in the configuration register CR1NV[3]. When CR1NV[3]=0 the non-volatile version of the BP bits (SR1NV[4:2]) are used to control Block Protection and the WRR command writes SR1NV[4:2] and updates SR1V[4:2] to the same value. When CR1NV[3]=1 the volatile version of the BP bits (SR1V[4:2]) are used to control Block Protection and the WRR command writes SR1V[4:2] and does not affect SR1NV[4:2]. When one or more of the BP bits is set to 1, the relevant memory area is protected against program and erase. The Bulk Erase (BE) command can be executed only when the BP bits are cleared to 0's. See Block Protection on page 65 for a description of how the BP bit values select the memory array area protected. The non-volatile version of the BP bits have the same non-volatile endurance as the main Flash array.

**Write Enable Latch Default (WEL_D) SR1NV[1]**: Provides the default state for the WEL Status in SR1V[1]. This bit is programmed by Cypress and is not user programmable.

**Write In Progress Default (WIP_D) SR1NV[0]**: Provides the default state for the WIP Status in SR1V[0]. This bit is programmed by Cypress and is not user programmable.

### 9.6.1.2 Status Register 1 Volatile (SR1V)

Related Commands: Read Status Register (RDSR1 05h), Write Registers (WRR 01h), Write Enable (WREN 06h), Write Disable (WRDI 04h), Clear Status Register (CLSR 30h or 82h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h). This is the register displayed by the RDSR1 command.

**Table 28. Status Register-1 Volatile (SR1V)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | SRWD | Status Register Write Disable | Volatile Read Only | SR1NV | Volatile copy of SR1NV[7]. |
| 6 | P_ERR | Programming Error Occurred | Volatile Read Only | | 1 = Error occurred<br>0 = No Error |
| 5 | E_ERR | Erase Error Occurred | Volatile Read Only | | 1= Error occurred<br>0 = No Error |
| 4 | BP2 | Block Protection Volatile | Volatile | | Protects selected range of sectors (Block) from Program or Erase when the BP bits are configured as volatile (CR1NV[3]=1). Volatile copy of SR1NV[4:2] when BP bits are configured as non-volatile. User writable when BP bits are configured as volatile. |
| 3 | BP1 | | | | |
| 2 | BP0 | | | | |
| 1 | WEL | Write Enable Latch | Volatile | | 1 = Device accepts Write Registers (WRR and WRAR), program, or erase commands<br>0 = Device ignores Write Registers (WRR and WRAR), program, or erase commands<br>This bit is not affected by WRR or WRAR, only WREN and WRDI commands affect this bit. |
| 0 | WIP | Write in Progress | Volatile Read Only | | 1= Device Busy, an embedded operation is in progress such as program or erase<br>0 = Ready Device is in standby mode and can accept commands<br>This bit is not affected by WRR or WRAR, it only provides WIP status. |

**Status Register Write (SRWD) SR1V[7]**: SRWD is a volatile copy of SR1NV[7]. This bit tracks any changes to the non-volatile version of this bit.

**Program Error (P_ERR) SR1V[6]**: The Program Error Bit is used as a program operation success or failure indication. When the Program Error bit is set to a “1” it indicates that there was an error in the last program operation. This bit will also be set when the user attempts to program within a protected main memory sector, or program within a locked OTP region. When the Program Error bit is set to a “1” this bit can be cleared to zero with the Clear Status Register (CLSR) command. This is a read-only bit and is not affected by the WRR or WRAR commands.

**Erase Error (E_ERR) SR1V[5]**: The Erase Error Bit is used as an Erase operation success or failure indication. When the Erase Error bit is set to a “1” it indicates that there was an error in the last erase operation. This bit will also be set when the user attempts to erase an individual protected main memory sector. The Bulk Erase command will not set E_ERR if a protected sector is found during the command execution. When the Erase Error bit is set to a “1” this bit can be cleared to zero with the Clear Status Register (CLSR) command. This is a read-only bit and is not affected by the WRR or WRAR commands.

**Block Protection (BP2, BP1, BP0) SR1V[4:2]**: These bits define the main Flash array area to be software-protected against program and erase commands. The BP bits are selected as either volatile or non-volatile, depending on the state of the BP non-volatile bit (BPNV_O) in the configuration register CR1NV[3]. When CR1NV[3]=0 the non-volatile version of the BP bits (SR1NV[4:2]) are used to control Block Protection and the WRR command writes SR1NV[4:2] and updates SR1V[4:2] to the same value. When CR1NV[3]=1 the volatile version of the BP bits (SR1V[4:2]) are used to control Block Protection and the WRR command writes SR1V[4:2] and does not affect SR1NV[4:2]. When one or more of the BP bits is set to 1, the relevant memory area is protected against program and erase. The Bulk Erase (BE) command can be executed only when the BP bits are cleared to 0’s. See Block Protection on page 65 for a description of how the BP bit values select the memory array area protected.

**Write Enable Latch (WEL) SR1V[1]**: The WEL bit must be set to 1 to enable program, write, or erase operations as a means to provide protection against inadvertent changes to memory or register values. The Write Enable (WREN) command execution sets the Write Enable Latch to a “1” to allow any program, erase, or write commands to execute afterwards. The Write Disable (WRDI) command can be used to set the Write Enable Latch to a “0” to prevent all program, erase, and write commands from execution. The WEL bit is cleared to 0 at the end of any successful program, write, or erase operation. Following a failed operation the WEL bit may remain set and should be cleared with a WRDI command following a CLSR command. After a power down / power up sequence, hardware reset, or software reset, the Write Enable Latch is set to a “0” The WRR or WRAR command does not affect this bit.

**Write In Progress (WIP) SR1V[0]**: Indicates whether the device is performing a program, write, erase operation, or any other operation, during which a new operation command will be ignored. When the bit is set to a “1” the device is busy performing an operation. While WIP is “1”, only Read Status (RDSR1 or RDSR2), Read Any Register (RDAR), Erase Suspend (ERSP), Program Suspend (PGSP), Clear Status Register (CLSR), and Software Reset (RESET) commands are accepted. ERSP and PGSP will only be accepted if memory array erase or program operations are in progress. The status register E_ERR and P_ERR bits are updated while WIP =1. When P_ERR or E_ERR bits are set to one, the WIP bit will remain set to one indicating the device remains busy and unable to receive new operation commands. A Clear Status Register (CLSR) command must be received to return the device to standby mode. When the WIP bit is cleared to 0 no operation is in progress. This is a read-only bit.

### 9.6.2 Status Register 2 Volatile (SR2V)

Related Commands: Read Status Register 2 (RDSR2 07h), Read Any Register (RDAR 65h). Status Register 2 does not have user programmable non-volatile bits, all defined bits are volatile read only status. The default state of these bits are set by hardware.

**Table 29. Status Register-2 Volatile (SR2V)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | RFU | Reserved | | 0 | Reserved for Future Use |
| 6 | RFU | Reserved | | 0 | Reserved for Future Use |
| 5 | RFU | Reserved | | 0 | Reserved for Future Use |
| 4 | RFU | Reserved | | 0 | Reserved for Future Use |
| 3 | RFU | Reserved | | 0 | Reserved for Future Use |
| 2 | ESTAT | Erase Status | Volatile Read Only | 0 | 1 = Sector Erase Status command result = Erase Completed<br>0 = Sector Erase Status command result = Erase Not Completed |
| 1 | ES | Erase Suspend | Volatile Read Only | 0 | 1 = In erase suspend mode.<br>0 = Not in erase suspend mode. |
| 0 | PS | Program Suspend | Volatile Read Only | 0 | 1 = In program suspend mode.<br>0 = Not in program suspend mode. |

**Erase Status (ESTAT) SR2V[2]:** The Erase Status bit indicates whether the sector, selected by an immediately preceding Erase status command, completed the last erase command on that sector. The Erase Status command must be issued immediately before reading SR2V to get valid erase status. Reading SR2V during a program or erase suspend does not provide valid erase status. The erase status bit can be used by system software to detect any sector that failed its last erase operation. This can be used to detect erase operations failed due to loss of power during the erase operation.

**Erase Suspend (ES) SR2V[1]**: The Erase Suspend bit is used to determine when the device is in Erase Suspend mode. This is a status bit that cannot be written by the user. When Erase Suspend bit is set to "1", the device is in erase suspend mode. When Erase Suspend bit is cleared to "0", the device is not in erase suspend mode. Refer to Erase or Program Suspend (EPS 85h, 75h, B0h) on page 107 for details about the Erase Suspend/Resume commands.

**Program Suspend (PS) SR2V[0]:** The Program Suspend bit is used to determine when the device is in Program Suspend mode. This is a status bit that cannot be written by the user. When Program Suspend bit is set to "1", the device is in program suspend mode. When the Program Suspend bit is cleared to "0", the device is not in program suspend mode. Refer to Erase or Program Suspend (EPS 85h, 75h, B0h) on page 107 for details.

### 9.6.3 Configuration Register 1

Configuration register 1 controls certain interface and data protection functions. The register bits can be changed using the WRR command with sixteen input cycles or with the WRAR command.

#### 9.6.3.1 Configuration Register 1 Non-volatile (CR1NV)

Related Commands: Write Registers (WRR 01h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

**Table 30. Configuration Register 1 Non-volatile (CR1NV)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | RFU | Reserved for Future Use | Non-volatile | 0 | Reserved |
| 6 | RFU | | | 0 | |
| 5 | TBPROT_O | Configures Start of Block Protection | OTP | 0 | 1 = BP starts at bottom (Low address)<br>0 = BP starts at top (High address) |
| 4 | RFU | Reserved for Future Use | RFU | 0 | Reserved |
| 3 | BPNV_O | Configures BP2-0 in Status Register | OTP | 0 | 1 = Volatile<br>0 = Non-Volatile |
| 2 | TBPARM_O | Configures Parameter Sectors location | OTP | 0 | 1 = 4 KB physical sectors at top, (high address)<br>0 = 4 KB physical sectors at bottom (Low address)<br>RFU in uniform sector configuration. |
| 1 | QUAD_NV | Quad Non-Volatile | Non-Volatile | 0 | Provides the default state for the QUAD bit. |
| 0 | FREEZE_D | FREEZE Default | Non-Volatile Read Only | 0 | Provides the default state for the Freeze bit. Not user programmable. |

**Top or Bottom Protection (TBPROT_O) CR1NV[5]:** This bit defines the operation of the Block Protection bits BP2, BP1, and BP0 in the Status Register. As described in the status register section, the BP2-0 bits allow the user to optionally protect a portion of the array, ranging from 1/64, ¼, ½, etc., up to the entire array. When TBPROT_O is set to a "0" the Block Protection is defined to start from the top (maximum address) of the array. When TBPROT_O is set to a "1" the Block Protection is defined to start from the bottom (zero address) of the array. The TBPROT_O bit is OTP and set to a "0" when shipped from Cypress. If TBPROT_O is programmed to 1, writing the bit with a zero does not change the value or set the Program Error bit (P_ERR in SR1V[6]).

The desired state of TBPROT_O must be selected during the initial configuration of the device during system manufacture; before the first program or erase operation on the main Flash array. TBPROT_O must not be programmed after programming or erasing is done in the main Flash array.

**Block Protection Non-Volatile (BPNV_O) CR1NV[3]**: The BPNV_O bit defines whether the BP_NV 2-0 bits or the BP 2-0 bits in the Status Register are selected to control the Block Protection feature. The BPNV_O bit is OTP and cleared to a “0” with the BP_NV bits cleared to “000” when shipped from Cypress. When BPNV_O is set to a “0” the BP_NV 2-0 bits in the Status Register are selected to control the block protection and are written by the WRR command. The time required to write the BP_NV bits is $t_W$. When BPNV is set to a “1” the BP2-0 bits in the Status Register are selected to control the block protection and the BP_NV 2-0 bits will be programmed to binary “111”. This will cause the BP 2-0 bits to be set to binary 111 after POR, hardware reset, or command reset. When BPNV is set to a 1, the WRR command writes only the volatile version of the BP bits (SR1V[4:2]). The non-volatile version of the BP bits (SR1NV[4:2]) are no longer affected by the WRR command. This allows the BP bits to be written an unlimited number of times because they are volatile and the time to write the volatile BP bits is the much faster $t_{CS}$ volatile register write time. If BPNV_O is programmed to 1, writing the bit with a zero does not change the value or set the Program Error bit (P_ERR in SR1V[6]).

**TBPARM_O CR1NV[2]**: TBPARM_O defines the logical location of the parameter block. The parameter block consists of eight 4 KB parameter sectors, which replace a 32 KB portion of the highest or lowest address sector. When TBPARM_O is set to a “1” the parameter block is in the top of the memory array address space. When TBPARM_O is set to a “0” the parameter block is at the Bottom of the array. TBPARM_O is OTP and set to a “0” when it ships from Cypress. If TBPARM_O is programmed to 1, writing the bit with a zero does not change the value or set the Program Error bit (P_ERR in SR1V[6]).

The desired state of TBPARM_O must be selected during the initial configuration of the device during system manufacture; before the first program or erase operation on the main Flash array. TBPARM_O must not be programmed after programming or erasing is done in the main Flash array.

TBPROT_O can be set or cleared independent of the TBPARM_O bit. Therefore, the user can elect to store parameter information from the bottom of the array and protect boot code starting at the top of the array, or vice versa. Or, the user can elect to store and protect the parameter information starting from the top or bottom together.

When the memory array is configured as uniform sectors, the TBPARM_O bit is Reserved for Future Use (RFU) and has no effect because all sectors are uniform size.

**Quad Data Width Non-volatile (QUAD_NV) CR1NV[1]**: Provides the default state for the QUAD bit in CR1V[1]. The WRR or WRAR command affects this bit. Non-volatile selection of QPI mode, by programming CR2NV[6] =1, will also program QUAD_NV =1 to change the non-volatile default to Quad data width mode. While QPI mode is selected by CR2V[6]=1, the Quad_NV bit cannot be cleared to 0.

**Freeze Protection Default (FREEZE) CR1NV[0]**: Provides the default state for the FREEZE bit in CR1V[0]. This bit is not user programmable.

### 9.6.3.2 Configuration Register 1 Volatile (CR1V)

Related Commands: Read Configuration Register (RDCR 35h), Write Registers (WRR 01h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h). This is the register displayed by the RDCR command.

**Table 31. Configuration Register 1 Volatile (CR1V)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | RFU | Reserved for Future Use | Volatile | CR1NV | Reserved |
| 6 | RFU | | | | |
| 5 | TBPROT | Volatile copy of TBPROT_O | Volatile Read Only | | Not user writable<br>See CR1NV[5] TBPROT_O |
| 4 | RFU | RFU | RFU | | Reserved for Future Use |
| 3 | BPNV | Volatile copy of BPNV_O | Volatile Read Only | | Not user writable<br>See CR1NV[3] BPNV_O |
| 2 | TBPARM | Volatile copy of TBPARM_O | Volatile Read Only | | Not user writable<br>See CR1NV[2] TBPARM_O |
| 1 | QUAD | Quad I/O mode | Volatile | | 1 = Quad<br>0 = Dual or Serial |
| 0 | FREEZE | Lock-down Block Protection until next power cycle | Volatile | | Lock current state of Block Protection control bits, and OTP regions<br>1 = Block Protection and OTP locked<br>0 = Block Protection and OTP un-locked |

**TBPROT, BPNV, and TBPARM CR1V[5, 3, 2]:** These bits are volatile copies of the related non-volatile bits of CR1NV. These bits track any changes to the related non-volatile version of these bits.

**Quad Data Width (QUAD) CR1V[1]**: When set to 1, this bit switches the data width of the device to 4 bit - Quad mode. That is, WP# becomes IO2 and IO3_RESET# becomes an active I/O signal when CS# is low or the RESET# input when CS# is high. The WP# input is not monitored for its normal function and is internally set to high (inactive). The commands for Serial, and Dual I/O Read still function normally but, there is no need to drive the WP# input for those commands when switching between commands using different data path widths. Similarly, there is no requirement to drive the IO3_RESET# during those commands (while CS# is low). The QUAD bit must be set to one when using the Quad I/O Read, DDR Quad I/O Read, QPI mode (CR2V[6] = 1), and Read Quad ID commands. While QPI mode is selected by CR2V[6]=1, the Quad bit cannot be cleared to 0. The WRR command writes the non-volatile version of the Quad bit (CR1NV[1]), which also causes an update to the volatile version CR1V[1]. The WRR command can not write the volatile version CR1V[1] without first affecting the non-volatile version CR1NV[1]. The WRAR command must be used when it is desired to write the volatile Quad bit CR1V[1] without affecting the non-volatile version CR1NV[1].

**Freeze Protection (FREEZE) CR1V[0]**: The Freeze Bit, when set to 1, locks the current state of the Block Protection control bits and OTP area:

- BPNV_2-0 bits in the non-volatile Status Register 1 (SR1NV[4:2])
- BP 2-0 bits in the volatile Status Register 1 (SR1V[4:2])
- TBPROT_O, TBPARM_O, and BPNV_O bits in the non-volatile Configuration Register (CR1NV[53, 2])
- TBPROT, TBPARM, and BPNV bits in the volatile Configuration Register (CR1V[5, 3, 2]) are indirectly protected in that they are shadows of the related CR1NV OTP bits and are read only
- The entire OTP memory space

Any attempt to change the above listed bits while FREEZE = 1 is prevented:

- The WRR command does not affect the listed bits and no error status is set.
- The WRAR command does not affect the listed bits and no error status is set.
- The OTPP command, with an address within the OTP area, fails and the P-ERR status is set.

As long as the FREEZE bit remains cleared to logic 0 the Block Protection control bits and FREEZE are writable, and the OTP address space is programmable.

Once the FREEZE bit has been written to a logic 1 it can only be cleared to a logic 0 by a power-off to power-on cycle or a hardware reset. Software reset will not affect the state of the FREEZE bit.

The CR1V[0] FREEZE bit is volatile and the default state of FREEZE after power-on comes from FREEZE_D in CR1NV[0]. The FREEZE bit can be set in parallel with updating other values in CR1V by a single WRR or WRAR command.

The FREEZE bit does not prevent the WRR or WRAR commands from changing the SRWD_NV (SR1NV[7]), Quad_NV (CR1NV[1]), or QUAD (CR1V[1]) bits.

### 9.6.4 Configuration Register 2

Configuration register 2 controls certain interface functions. The register bits can be read and changed using the Read Any Register and Write Any Register commands. The non-volatile version of the register provides the ability to set the POR, hardware reset, or software reset state of the controls. These configuration bits are OTP and may only have their default state changed to the opposite value one time during system configuration. The volatile version of the register controls the feature behavior during normal operation.

#### 9.6.4.1 Configuration Register 2 Non-volatile (CR2NV)

Related Commands: Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

**Table 32. Configuration Register 2 Non-volatile (CR2NV)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | AL_NV | Address Length | OTP | 0 | 1 = 4 byte address<br>0 = 3 byte address |
| 6 | QA_NV | QPI | | 0 | 1 = Enabled -- QPI (4-4-4) protocol in use<br>0 = Disabled -- Legacy SPI protocols in use, instruction is always serial on SI |
| 5 | IO3R_NV | IO3 Reset | | 0 | 1 = Enabled -- IO3 is used as RESET# input when CS# is high or Quad Mode is disabled CR1V[1]=1<br>0 = Disabled -- IO3 has no alternate function, hardware reset is disabled. |
| 4 | RFU | Reserved | | 0 | Reserved For Future Use |
| 3 | RL_NV | Read Latency | | 1 | 0 to 15 latency (dummy) cycles following read address or continuous mode bits.<br>Note that bit 3 has a default value of 1 and may be programmed one time to 0 but cannot be returned to 1. |
| 2 | | | | 0 | |
| 1 | | | | 0 | |
| 0 | | | | 0 | |

**Address Length Non-volatile CR2NV[7]:** This bit controls the POR, hardware reset, or software reset state of the expected address length for all commands that require address and are not fixed 3 byte only or 4 Byte (32 bit) only address. Most commands that need an address are legacy SPI commands that traditionally used 3 byte (24 bit) address. For device densities greater than 128 Mbit a 4 Byte address is required to access the entire memory array. The address length configuration bit is used to change most 3 Byte address commands to expect 4 Byte address. See Table 51 on page 75 for command address length. The use of 4 Byte address length also applies to the 128 Mbit member of the FS-S Family so that the same 4 Byte address hardware and software interface may be used for all family members to simplify migration between densities. The 128 Mbit member of the FS-S Family simply ignores the content of the fourth, high order, address byte. This non-volatile Address Length configuration bit enables the device to start immediately (boot) in 4 Byte address mode rather than the legacy 3 Byte address mode.

**QPI Non-volatile CR2NV[6]:** This bit controls the POR, hardware reset, or software reset state of the expected instruction width for all commands. Legacy SPI commands always send the instruction one bit wide (serial I/O) on the SI (IO0) signal. The FS-S Family also supports the QPI mode in which all transfers between the host system and memory are 4 bits wide on IO0 to IO3, including all instructions. This non-volatile QPI configuration bit enables the device to start immediately (boot) in QPI mode rather than the legacy serial instruction mode. When this bit is programmed to QPI mode, the QUAD_NV bit is also programmed to Quad mode (CR1NV[1]=1). The recommended procedure for moving to QPI mode is to first use the WRAR command to set CR2V[6]=1, QPI mode. The volatile register write for QPI mode has a short and well defined time ($t_{CS}$) to switch the device interface into QPI mode. Following commands can then be immediately sent in QPI protocol. The WRAR command can be used to program CR2NV[6]=1, followed by polling of SR1V[0] to know when the programming operation is completed. Similarly, to exit QPI mode, the WRAR command is used to clear CR2V[6]=0. CR2NV[6] cannot be erased to 0 because it is OTP.

**IO3 Reset Non-volatile CR2NV[5]:** This bit controls the POR, hardware reset, or software reset state of the IO3 signal behavior. Most legacy SPI devices do not have a hardware reset input signal due to the limited signal count and connections available in traditional SPI device packages. The FS-S Family provides the option to use the IO3 signal as a hardware reset input when the IO3 signal is not in use for transferring information between the host system and the memory. This non-volatile IO3 Reset configuration bit enables the device to start immediately (boot) with IO3 enabled for use as a RESET# signal.

**Read Latency Non-volatile CR2NV[3:0]:** This bit controls the POR, hardware reset, or software reset state of the read latency (dummy cycle) delay in all variable latency read commands. The following read commands have a variable latency period between the end of address or mode and the beginning of read data returning to the host:

- Fast Read
- Dual Output Read
- Quad Output Read
- Dual I/O Read
- Quad I/O Read
- DDR Quad I/O Read
- OTPR
- ECCRD
- RDAR

This non-volatile read latency configuration bit sets the number of read latency (dummy cycles) in use so the device can start immediately (boot) with an appropriate read latency for the host system.

**Table 33. Latency Code (Cycles) Versus Frequency**

| Latency Code | Read Command Maximum Frequency (MHz) | | | | |
|---|---|---|---|---|---|
| | Fast Read (1-1-1)Dual Output (1-1-2) Quad Output (1-1-4) OTPR (1-1-1) ECCRD (1-1-1) RDAR (1-1-1) RDAR (4-4-4) | Dual I/O (1-2-2) | Quad I/O (1-4-4) QPI (4-4-4) | ECCRD (4-4-4) | DDR Quad I/O (1-4-4) DDR QPI (4-4-4) |
| | Mode Cycles = 0 | Mode Cycles = 4 | Mode Cycles = 2 | Mode Cycles = 0 | Mode Cycles = 1 |
| 0 | 50 | 80 | 40 | 16 | N/A |
| 1 | 66 | 92 | 53 | 26 | 22 |
| 2 | 80 | 104 | 66 | 40 | 34 |
| 3 | 92 | 116 | 80 | 53 | 45 |
| 4 | 104 | 129 | 92 | 66 | 57 |
| 5 | 116 | 133 | 104 | 80 | 68 |
| 6 | 129 | 133 | 116 | 92 | 80 |
| 7 | 133 | 133 | 129 | 104 | 80 |
| 8 | 133 | 133 | 133 | 116 | 80 |
| 9 | 133 | 133 | 133 | 129 | 80 |
| 10 | 133 | 133 | 133 | 133 | 80 |
| 11 | 133 | 133 | 133 | 133 | 80 |
| 12 | 133 | 133 | 133 | 133 | 80 |
| 13 | 133 | 133 | 133 | 133 | 80 |
| 14 | 133 | 133 | 133 | 133 | 80 |
| 15 | 133 | 133 | 133 | 133 | 80 |

**Notes**

44. SCK frequency > 133 MHz SDR, or 80MHz DDR is not supported by this family of devices.
45. The Dual I/O, Quad I/O, and QPI, DDR Quad I/O, and DDR QPI, command protocols include Continuous Read Mode bits following the address. The clock cycles for these bits are not counted as part of the latency cycles shown in the table. Example: the legacy Quad I/O command has 2 Continuous Read Mode cycles following the address. Therefore, the legacy Quad I/O command without additional read latency is supported only up to the frequency shown in the table for a read latency of 0 cycles. By increasing the variable read latency the frequency of the Quad I/O command can be increased to allow operation up to the maximum supported 133 MHz frequency.
46. Other read commands have fixed latency, e.g. Read always has zero read latency, RSFDP always has eight cycles of latency and RUID always has 32 cycles of latency, RUID always four dummy bytes or 16 dummy bytes in QPI (32 clock cycles).

### 9.6.4.2 Configuration Register 2 Volatile (CR2V)

Related Commands: Read Any Register (RDAR 65h), Write Any Register (WRAR 71h), 4BAM.

**Table 34. Configuration Register 2 Volatile (CR2V)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | AL | Address Length | Volatile | CR2NV | 1 = 4 byte address<br>0 = 3 byte address |
| 6 | QA | QPI | | | 1 = Enabled -- QPI (4-4-4) protocol in use<br>0 = Disabled -- Legacy SPI protocols in use, instruction is always serial on SI |
| 5 | IO3R_S | IO3 Reset | | | 1 = Enabled -- IO3 is used as RESET# input when CS# is high or Quad Mode is disabled CR1V[1]=1<br>0 = Disabled -- IO3 has no alternate function, hardware reset is disabled. |
| 4 | RFU | Reserved | | | Reserved for Future Use |
| 3 | RL | Read Latency | | | 0 to 15 latency (dummy) cycles following read address or continuous mode bits |
| 2 | | | | | |
| 1 | | | | | |
| 0 | | | | | |

**Address Length CR2V[7]:** This bit controls the expected address length for all commands that require address and are not fixed 3 byte only or 4 Byte (32 bit) only address. See Table 51 on page 75 for command address length. This volatile Address Length configuration bit enables the address length to be changed during normal operation. The four byte address mode (4BAM) command directly sets this bit into 4 byte address mode.

**QPI CR2V[6]:** This bit controls the expected instruction width for all commands. This volatile QPI configuration bit enables the device to enter and exit QPI mode during normal operation. When this bit is set to QPI mode, the QUAD bit is also set to Quad mode (CR1V[1]=1). When this bit is cleared to legacy SPI mode, the QUAD bit is not affected.

**IO3 Reset CR2V[5]:** This bit controls the IO3_RESET# signal behavior. This volatile IO3 Reset configuration bit enables the use of IO3 as a RESET# input during normal operation.

**Read Latency CR2V[3:0]:** This bit controls the read latency (dummy cycle) delay in variable latency read commands These volatile configuration bits enable the user to adjust the read latency during normal operation to optimize the latency for different commands or, at different operating frequencies, as needed.

### 9.6.5 Configuration Register 3

Configuration register 3 controls certain command behaviors. The register bits can be read and changed using the Read Any Register and Write Any Register commands. The non-volatile register provides the POR, hardware reset, or software reset state of the controls. These configuration bits are OTP and may be programmed to their opposite state one time during system configuration if needed. The volatile version of configuration register 3 allows the configuration to be changed during system operation or testing.

### 9.6.5.1 Configuration Register 3 Non-volatile (CR3NV)

Related Commands: Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

**Table 35. Configuration Register 3 Non-volatile (CR3NV)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | RFU | Reserved | OTP | 0 | Reserved for Future Use |
| 6 | RFU | Reserved | | 0 | Reserved for Future Use |
| 5 | BC_NV | Blank Check | | 0 | 1 = Blank Check during erase enabled<br>0 = Blank Check disabled |
| 4 | 02h_NV | Page Buffer Wrap | | 0 | 1 = Wrap at 512 Bytes<br>0 = Wrap at 256 Bytes |
| 3 | 20h_NV | 4KB Erase | | 0 | 1 = 4KB Erase disabled (Uniform Sector Architecture)<br>0 = 4KB Erase enabled (Hybrid Sector Architecture) |
| 2 | 30h_NV | Clear Status / Resume Select | | 0 | 1 = 30h is Erase or Program Resume command<br>0 = 30h is clear status command |
| 1 | D8h_NV | Block Erase Size | | 0 | 1 = 256KB Erase<br>0 = 64KB Erase |
| 0 | F0h_NV | Legacy Software Reset Enable | | 0 | 1 = F0h Software Reset is enabled<br>0 = F0h Software Reset is disabled (ignored) |

**Blank Check Non-volatile CR3NV[5]:** This bit controls the POR, hardware reset, or software reset state of the blank check during erase feature.

**02h Non-volatile CR3NV[4]:** This bit controls the POR, hardware reset, or software reset state of the page programming buffer address wrap point.

**20h Non-volatile CR3NV[3]:** This bit controls the POR, hardware reset, or software reset state of the availability of 4 KB parameter sectors in the main Flash array address map.

**30h Non-volatile CR3NV[2]:** This bit controls the POR, hardware reset, or software reset state of the 30h instruction code is used.

**D8h Non-volatile CR3NV[1]:** This bit controls the POR, hardware reset, or software reset state of the configuration for the size of the area erased by the D8h or DCh instructions in the FS-S Family.

**F0h Non-volatile CR3NV[0]:** This bit controls the POR, hardware reset, or software reset state of the availability of the Cypress legacy FL-S family software reset instruction.

### 9.6.5.2 Configuration Register 3 Volatile (CR3V)

Related Commands: Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

**Table 36. Configuration Register 3 Volatile (CR3V)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | RFU | Reserved | Volatile | CR3NV | Reserved for Future Use |
| 6 | RFU | Reserved | | | Reserved for Future Use |
| 5 | BC_V | Blank Check | | | 1 = Blank Check during erase enabled<br>0 = Blank Check disabled |
| 4 | 02h_V | Page Buffer Wrap | | | 1 = Wrap at 512 Bytes<br>0 = Wrap at 256 Bytes |
| 3 | 20h_V | 4KB Erase | Volatile, Read Only | | 1 = 4KB Erase disabled (Uniform Sector Architecture)<br>0 = 4KB Erase enabled (Hybrid Sector Architecture) |
| 2 | 30h_V | Clear Status / Resume Select | Volatile | | 1 = 30h is Erase or Program Resume command<br>0 = 30h is clear status command |
| 1 | D8h_V | Block Erase Size | | | 1 = 256KB Erase<br>0 = 64KB Erase |
| 0 | F0h_V | Legacy Software Reset Enable | | | 1 = F0h Software Reset is enabled<br>0 = F0h Software Reset is disabled (ignored) |

**Blank Check Volatile CR3V[5]:** This bit controls the blank check during erase feature. When this feature is enabled an erase command first evaluates the erase status of the sector. If the sector is found to have not completed its last erase successfully, the sector is unconditionally erased. If the last erase was successful, the sector is read to determine if the sector is still erased (blank). The erase operation is started immediately after finding any programmed zero. If the sector is already blank (no programmed zero bit found) the remainder of the erase operation is skipped. This can dramatically reduce erase time when sectors being erased do not need the erase operation. When enabled the blank check feature is used within the parameter erase, sector erase, and bulk erase commands. When blank check is disabled an erase command unconditionally starts the erase operation.

**02h Volatile CR3V[4]:** This bit controls the page programming buffer address wrap point. Legacy SPI devices generally have used a 256 Byte page programming buffer and defined that if data is loaded into the buffer beyond the 255 Byte location, the address at which additional bytes are loaded would be wrapped to address zero of the buffer. The FS-S Family provides a 512 Byte page programming buffer that can increase programming performance. For legacy software compatibility, this configuration bit provides the option to continue the wrapping behavior at the 256 Byte boundary or to enable full use of the available 512 Byte buffer by not wrapping the load address at the 256 Byte boundary.

**20h Volatile CR3V[3]:** This bit controls the availability of 4 KB parameter sectors in the main Flash array address map. The parameter sectors can overlay the highest or lowest 32 KB address range of the device or they can be removed from the address map so that all sectors are uniform size. This bit shall not be written to a value different than the value of CR3NV[3]. The value of CR3V[3] may only be changed by writing CR3NV[3].

**30h Volatile CR3V[2]:** This bit controls how the 30h instruction code is used. The instruction may be used as a clear status command or as an alternate program / erase resume command. This allows software compatibility with either Cypress legacy SPI devices or alternate vendor devices.

**D8h Volatile CR3V[1]:** This bit controls the area erased by the D8h or DCh instructions in the FS-S Family. The instruction can be used to erase 64 KB physical sectors or 256 KB size and aligned blocks. The option to erase 256 KB blocks in the lower density family members allows for consistent software behavior across all densities that can ease migration between different densities.

**F0h Volatile CR3V[0]:** This bit controls the availability of the Cypress legacy FL-S family software reset instruction. The FS-S Family supports the industry common 66h + 99h instruction sequence for software reset. This configuration bit allows the option to continue use of the legacy F0h single command for software reset.

### 9.6.6 Configuration Register 4

Configuration register 4 controls the main Flash array read commands burst wrap behavior. The burst wrap configuration does not affect commands reading from areas other than the main Flash array e.g. read commands for registers or OTP array. The non-volatile version of the register provides the ability to set the start up (boot) state of the controls as the contents are copied to the volatile version of the register during the POR, hardware reset, or software reset. The volatile version of the register controls the feature behavior during normal operation. The register bits can be read and changed using the Read Any Register and Write Any Register commands. The volatile version of the register can also be written by the Set Burst Length (C0h) command.

#### 9.6.6.1 Configuration Register 4 Non-volatile (CR4NV)

Related Commands: Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

**Table 37. Configuration Register 4 Non-volatile (CR4NV)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | OI_O | Output Impedance | OTP | 0 | See Table 38. Output Impedance Control on page 60 |
| 6 | | | | 0 | |
| 5 | | | | 0 | |
| 4 | WE_O | Wrap Enable | | 1 | 0= Wrap Enabled<br>1= Wrap Disabled |
| 3 | RFU | Reserved | | 0 | Reserved for Future Use |
| 2 | RFU | Reserved | | 0 | Reserved for Future Use |
| 1 | WL_O | Wrap Length | | 0 | 00 = 8-byte wrap<br>01= 16 byte wrap<br>10= 32 byte wrap<br>11= 64 byte wrap |
| 0 | | | | 0 | |

**Output Impedance Non-volatile CR4NV[7:5]:** These bits control the POR, hardware reset, or software reset state of the IO signal output impedance (drive strength). Multiple drive strength are available to help match the output impedance with the system printed circuit board environment to minimize overshoot and ringing. These non-volatile output impedance configuration bits enable the device to start immediately (boot) with the appropriate drive strength.

**Table 38. Output Impedance Control**

| CR4NV[7:5] Impedance Selection | Typical Impedance to $V_{SS}$ (Ohms) | Typical Impedance to $V_{CC}$ (Ohms) | Notes |
|---|---|---|---|
| 000 | 47 | 45 | Factory Default |
| 001 | 124 | 105 | |
| 010 | 71 | 64 | |
| 011 | 47 | 45 | |
| 100 | 34 | 35 | |
| 101 | 26 | 28 | |
| 110 | 22 | 24 | |
| 111 | 18 | 21 | |

**Wrap Enable Non-volatile CR4NV[4]:** This bit controls the POR, hardware reset, or software reset state of the wrap enable. The commands affected by Wrap Enable are: Quad I/O Read, DDR Quad I/O Read, Quad Output Read and QPI Read. This configuration bit enables the device to start immediately (boot) in wrapped burst read mode rather than the legacy sequential read mode.

**Wrap Length Non-volatile CR4NV[1:0]:** These bits controls the POR, hardware reset, or software reset state of the wrapped read length and alignment. These non-volatile configuration bits enable the device to start immediately (boot) in wrapped burst read mode rather than the legacy sequential read mode.

### 9.6.6.2 Configuration Register 4 Volatile (CR4V)

Related Commands: Read Any Register (RDAR 65h), Write Any Register (WRAR 71h), Set Burst Length (SBL C0h).

**Table 39. Configuration Register 4 Volatile (CR4V)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | OI | Output Impedance | Volatile | CR4NV | See Table 38. Output Impedance Control on page 60 |
| 6 | | | | | |
| 5 | | | | | |
| 4 | WE | Wrap Enable | | | 0= Wrap Enabled<br>1= Wrap Disabled |
| 3 | RFU | Reserved | | | Reserved for Future Use |
| 2 | RFU | Reserved | | | Reserved for Future Use |
| 1 | WL | Wrap Length | | | 00 = 8-byte wrap<br>01= 16 byte wrap<br>10= 32 byte wrap<br>11= 64 byte wrap |
| 0 | | | | | |

**Output Impedance CR2V[7:5]:** These bits control the IO signal output impedance (drive strength). This volatile output impedance configuration bit enables the user to adjust the drive strength during normal operation.

**Wrap Enable CR4V[4]:** This bit controls the burst wrap feature. This volatile configuration bit enables the device to enter and exit burst wrapped read mode during normal operation.

**Wrap Length CR4V[1:0]:** These bits controls the wrapped read length and alignment during normal operation. These volatile configuration bits enable the user to adjust the burst wrapped read length during normal operation.

### 9.6.7 ECC Status Register (ECCSR)

Related Commands: ECC Read (ECCRD 18h or 19h). ECCSR does not have user programmable non-volatile bits. All defined bits are volatile read only status. The default state of these bits are set by hardware. See Automatic ECC on page 101.

The status of ECC in each ECC unit is provided by the 8-bit ECC Status Register (ECCSR). The ECC Register Read command is written followed by an ECC unit address. The contents of the status register then indicates, for the selected ECC unit, whether there is an error in the ECC unit eight bit error correction code, the ECC unit of 16 Bytes of data, or that ECC is disabled for that ECC unit.

**Table 40. ECC Status Register (ECCSR)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 to 3 | RFU | Reserved | | 0 | Reserved for Future Use |
| 2 | EECC | Error in ECC | Volatile, Read only | 0 | 1 = Single Bit Error found in the ECC unit eight bit error correction code<br>0 = No error. |
| 1 | EECCD | Error in ECC unit data | Volatile, Read only | 0 | 1 = Single Bit Error corrected in ECC unit data.<br>0 = No error. |
| 0 | ECCDI | ECC Disabled | Volatile, Read only | 0 | 1 = ECC is disabled in the selected ECC unit. 0 = ECC is enabled in the selected ECC unit. |

ECCSR[2] = 1 indicates an error was corrected in the ECC. ECCSR[1] = 1 indicates an error was corrected in the ECC unit data. ECCSR[0] = 1 indicates the ECC is disabled. The default state of “0” for all these bits indicates no failures and ECC is enabled.

ECCSR[7:3] are reserved. These have undefined high or low values that can change from one ECC status read to another. These bits should be treated as “don’t care” and ignored by any software reading status.

### 9.6.8 ASP Register (ASPR)

Related Commands: ASP Read (ASPRD 2Bh) and ASP Program (ASPP 2Fh), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

The ASP register is a 16 bit OTP memory location used to permanently configure the behavior of Advanced Sector Protection (ASP) features. ASPR does not have user programmable volatile bits, all defined bits are OTP.

The default state of the ASPR bits are programmed by Cypress.

**Table 41. ASP Register (ASPR)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 15 to 9 | RFU | Reserved | OTP | 1 | Reserved for Future Use |
| 8 | RFU | Reserved | OTP | 1 | Reserved for Future Use |
| 7 | RFU | Reserved | OTP | 1 | Reserved for Future Use |
| 6 | RFU | Reserved | OTP | 1 | Reserved for Future Use |
| 5 | RFU | Reserved | OTP | 1 | Reserved for Future Use |
| 4 | | Reserved | RFU | 1 | Reserved for Future Use |
| 3 | | Reserved | RFU | 1 | Reserved for Future Use |
| 2 | PWDMLB | Password Protection Mode Lock Bit | OTP | 1 | 0 = Password Protection Mode permanently enabled.<br>1 = Password Protection Mode not permanently enabled. |
| 1 | PSTMLB | Persistent Protection Mode Lock Bit | OTP | 1 | 0 = Persistent Protection Mode permanently enabled.<br>1 = Persistent Protection Mode not permanently enabled. |
| 0 | | Reserved | RFU | 1 | Reserved for Future Use |

**Password Protection Mode Lock Bit (PWDMLB) ASPR[2]:** When programmed to 0, the Password Protection Mode is permanently selected.

**Persistent Protection Mode Lock Bit (PSTMLB) ASPR[1]:** When programmed to 0, the Persistent Protection Mode is permanently selected.

PWDMLB (ASPR[2]) and PSTMLB (ASPR[1]) are mutually exclusive, only one may be programmed to zero.

ASPR bits may only be programmed while ASPR[2:1] = 11b. Attempting to program ASPR bits when ASPR[2:1] is not = 11b will result in a programming error with P_ERR (SR1V[6]) set to 1. After the ASP protection mode is selected by programming ASPR[2:1] = 10b or 01b, the state of all ASPR bits are locked and permanently protected from further programming. Attempting to program ASPR[2:1] = 00b will result in a programming error with P_ERR (SR1V[6]) set to 1.

Similarly, OTP configuration bits listed in the ASP Register description (see ASP Register on page 68), may only be programmed while ASPR[2:1] = 11b. The OTP configuration must be selected before selecting the ASP protection mode. The OTP configuration bits are permanently protected from further change when the ASP protection mode is selected. Attempting to program these OTP configuration bits when ASPR[2:1] is not = 11b will result in a programming error with P_ERR (SR1V[6]) set to 1.

The ASP protection mode should be selected during system configuration to ensure that a malicious program does not select an undesired protection mode at a later time. By locking all the protection configuration via the ASP mode selection, later alteration of the protection methods by malicious programs is prevented.

### 9.6.9 Password Register (PASS)

Related Commands: Password Read (PASSRD E7h) and Password Program (PASSP E8h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h). The PASS register is a 64 bit OTP memory location used to permanently define a password for the Advanced Sector Protection (ASP) feature. PASS does not have user programmable volatile bits, all defined bits are OTP. A volatile copy of PASS is used to satisfy read latency requirements but the volatile register is not user writable or further described.

**Table 42. Password Register (PASS)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 63 to 0 | PWD | Hidden Password | OTP | FFFFFFFF-FFFFFFFFh | Non-volatile OTP storage of 64 bit password. The password is no longer readable after the password protection mode is selected by programming ASP register bit 2 to zero. |

### 9.6.10 PPB Lock Register (PPBL)

Related Commands: PPB Lock Read (PLBRD A7h, PLBWR A6h), Read Any Register (RDAR 65h).

PPBL does not have separate user programmable non-volatile bits, all defined bits are volatile read only status. The default state of the RFU bits is set by hardware. The default state of the PPBLOCK bit is defined by the ASP protection mode bits in ASPR[2:1]. There is no non-volatile version of the PPBL register.

The PPBLOCK bit is used to protect the PPB bits. When PPBL[0] = 0, the PPB bits can not be programmed.

**Table 43. PPB Lock Register (PPBL)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 to 1 | RFU | Reserved | Volatile | 00h | Reserved for Future Use |
| 0 | PPBLOCK | Protect PPB Array | Volatile Read Only | ASPR[2:1] = 1xb = Persistent Protection Mode = 1<br>ASPR[2:1] = 01b = Password Protection Mode = 0 | 0 = PPB array protected<br>1 = PPB array may be programmed or erased. |

### 9.6.11 PPB Access Register (PPBAR)

Related Commands: PPB Read (PPBRD FCh or 4PPBRD E2h), PPB Program (PPBP FDh or 4PPBP E3h), PPB Erase (PPBE E4h).

PPBAR does not have user writable volatile bits, all PPB array bits are non-volatile. The default state of the PPB array is erased to FFh by Cypress. There is no volatile version of the PPBAR register.

**Table 44. PPB Access Register (PPBAR)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 to 0 | PPB | Read or Program per sector PPB | Non-volatile | FFh | 00h = PPB for the sector addressed by the PPBRD or PPBP command is programmed to 0, protecting that sector from program or erase operations.<br>FFh = PPB for the sector addressed by the PPBRD command is 1, not protecting that sector from program or erase operations. |

### 9.6.12 DYB Access Register (DYBAR)

Related Commands: DYB Read (DYBRD FAh or 4DYBRD E0h) and DYB Write (DYBWR FBh or 4DYBWR E1h).

DYBAR does not have user programmable non-volatile bits, all bits are a representation of the volatile bits in the DYB array. The default state of the DYB array bits is set by hardware. There is no non-volatile version of the DYBAR register.

**Table 45. DYB Access Register (DYBAR)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 to 0 | DYB | Read or Write per sector DYB | Volatile | | 00h = DYB for the sector addressed by the DYBRD or DYBWR command is cleared to “0”, protecting that sector from program or erase operations.<br>FFh = DYB for the sector addressed by the DYBRD or DYBWR command is set to “1”, not protecting that sector from program or erase operations. |

### 9.6.13 SPI DDR Data Learning Registers

Related Commands: Program NVDLR (PNVDLR 43h), Write VDLR (WVDLR 4Ah), Data Learning Pattern Read (DLPRD 41h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

The Data Learning Pattern (DLP) resides in an 8-bit Non-Volatile Data Learning Register (NVDLR) as well as an 8-bit Volatile Data Learning Register (VDLR). When shipped from Cypress, the NVDLR value is 00h. Once programmed, the NVDLR cannot be reprogrammed or erased; a copy of the data pattern in the NVDLR will also be written to the VDLR. The VDLR can be written to at any time, but on power cycles the data pattern will revert back to what is in the NVDLR. During the learning phase described in the SPI DDR modes, the DLP will come from the VDLR. Each IO will output the same DLP value for every clock edge. For example, if the DLP is 34h (or binary 00110100) then during the first clock edge all IO’s will output 0; subsequently, the 2nd clock edge all I/O’s will output 0, the 3rd will output 1, etc.

When the VDLR value is 00h, no preamble data pattern is presented during the dummy phase in the DDR commands.

**Table 46. Non-Volatile Data Learning Register (NVDLR)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 to 0 | NVDLP | Non-Volatile Data Learning Pattern | OTP | 00h | OTP value that may be transferred to the host during DDR read command latency (dummy) cycles to provide a training pattern to help the host more accurately center the data capture point in the received data bits. |

**Table 47. Volatile Data Learning Register (VDLR)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 to 0 | VDLP | Volatile Data Learning Pattern | Volatile | Takes the value of NVDLR during POR or Reset | Volatile copy of the NVDLP used to enable and deliver the Data Learning Pattern (DLP) to the outputs. The VDLP may be changed by the host during system operation. |

# 10. Data Protection

## 10.1 Secure Silicon Region (OTP)

The device has a 1024 byte One Time Program (OTP) address space that is separate from the main Flash array. The OTP area is divided into 32, individually lockable, 32 byte aligned and length regions.

The OTP memory space is intended for increased system security. OTP values can “mate” a flash component with the system CPU/ASIC to prevent device substitution. See OTP Address Space on page 47, OTP Program (OTPP 42h) on page 111, and OTP Read (OTPR 4Bh) on page 111.

### 10.1.1 Reading OTP Memory Space

The OTP Read command uses the same protocol as Fast Read. OTP Read operations outside the valid 1KB OTP address range will yield indeterminate data.

### 10.1.2 Programming OTP Memory Space

The protocol of the OTP programming command is the same as Page Program. The OTP Program command can be issued multiple times to any given OTP address, but this address space can never be erased.

Automatic ECC is programmed on the first programming operation to each 16 byte region. Programming within a 16 byte region more than once disables the ECC. It is recommended to program each 16 byte portion of each 32 byte region once so that ECC remains enabled to provide the best data integrity.

The valid address range for OTP Program is depicted in Figure 42 on page 47. OTP Program operations outside the valid OTP address range will be ignored, without P_ERR in SR1V set to “1”. OTP Program operations within the valid OTP address range, while FREEZE = 1, will fail with P_ERR in SR1V set to “1”. The OTP address space is not protected by the selection of an ASP protection mode. The Freeze bit (CR1V[0]) may be used to protect the OTP Address Space.

### 10.1.3 Cypress Programmed Random Number

Cypress standard practice is to program the low order 16 bytes of the OTP memory space (locations 0x0 to 0xF) with a 128-bit random number using the Linear Congruential Random Number Method. The seed value for the algorithm is a random number concatenated with the day and time of tester insertion.

### 10.1.4 Lock Bytes

The LSB of each Lock byte protects the lowest address region related to the byte, the MSB protects the highest address region related to the byte. The next higher address byte similarly protects the next higher 8 regions. The LSB bit of the lowest address Lock Byte protects the higher address 16 bytes of the lowest address region. In other words, the LSB of location 0x10 protects all the Lock Bytes and RFU bytes in the lowest address region from further programming. See OTP Address Space on page 47.

## 10.2 Write Enable Command

The Write Enable (WREN) command must be written prior to any command that modifies non-volatile data. The WREN command sets the Write Enable Latch (WEL) bit. The WEL bit is cleared to 0 (disables writes) during power-up, hardware reset, or after the device completes the following commands:

- Reset
- Page Program (PP or 4PP)
- Parameter 4KB Erase (P4E or 4P4E)
- Sector Erase (SE or 4SE)
- Bulk Erase (BE)
- Write Disable (WRDI)
- Write Registers (WRR)
- Write Any Register (WRAR)
- OTP Byte Programming (OTPP)
- Advanced Sector Protection Register Program (ASPP)
- Persistent Protection Bit Program (PPBP)
- Persistent Protection Bit Erase (PPBE)
- Password Program (PASSP)
- Program Non-Volatile Data Learning Register (PNVDLR)

## 10.3 Block Protection

The Block Protect bits (Status Register bits BP2, BP1, BP0) in combination with the Configuration Register TBPROT_O bit can be used to protect an address range of the main Flash array from program and erase operations. The size of the range is determined by the value of the BP bits and the upper or lower starting point of the range is selected by the TBPROT_O bit of the configuration register (CR1NV[5]).

**Table 48. S25FS064S Upper Array Start of Protection (TBPROT_O = 0)**

| Status Register Content | | | Protected Fraction of Memory Array | Protected Memory (KBytes) |
|---|---|---|---|---|
| BP2 | BP1 | BP0 | | |
| 0 | 0 | 0 | None | 0 |
| 0 | 0 | 1 | Upper 64th | 128 |
| 0 | 1 | 0 | Upper 32nd | 256 |
| 0 | 1 | 1 | Upper 16th | 512 |
| 1 | 0 | 0 | Upper 8th | 1024 |
| 1 | 0 | 1 | Upper 4th | 2048 |
| 1 | 1 | 0 | Upper Half | 4096 |
| 1 | 1 | 1 | All Sectors | 8192 |

**Table 49. S25FS064S Lower Array Start of Protection (TBPROT_O = 1)**

| Status Register Content | | | Protected Fraction of Memory Array | Protected Memory (KBytes) |
|---|---|---|---|---|
| BP2 | BP1 | BP0 | | |
| 0 | 0 | 0 | None | 0 |
| 0 | 0 | 1 | Lower 64th | 128 |
| 0 | 1 | 0 | Lower 32nd | 256 |
| 0 | 1 | 1 | Lower 16th | 512 |
| 1 | 0 | 0 | Lower 8th | 1024 |
| 1 | 0 | 1 | Lower 4th | 2048 |
| 1 | 1 | 0 | Lower Half | 4096 |
| 1 | 1 | 1 | All Sectors | 8192 |

When Block Protection is enabled (i.e., any BP2-0 are set to “1”), Advanced Sector Protection (ASP) can still be used to protect sectors not protected by the Block Protection scheme. In the case that both ASP and Block Protection are used on the same sector the logical OR of ASP and Block Protection related to the sector is used.

### 10.3.1 Freeze bit

Bit 0 of Configuration Register 1 (CR1V[0]) is the FREEZE bit. The Freeze Bit, when set to 1, locks the current state of the Block Protection control bits and OTP area until the next power off-on cycle. Additional details in Configuration Register 1 Volatile (CR1V) on page 54.

### 10.3.2 Write Protect Signal

The Write Protect (WP#) input in combination with the Status Register Write Disable (SRWD) bit (SR1NV[7]) provide hardware input signal controlled protection. When WP# is Low and SRWD is set to “1” Status Register-1 (SR1NV and SR1V) and Configuration register-1 (CR1NV and CR1V) are protected from alteration. This prevents disabling or changing the protection defined by the Block Protect bits. See Status Registers 1 on page 49.

## 10.4 Advanced Sector Protection

Advanced Sector Protection (ASP) is the name used for a set of independent hardware and software methods used to disable or enable programming or erase operations, individually, in any or all sectors.

Every main Flash array sector has a non-volatile Persistent Protection Bit (PPB) and a volatile Dynamic Protection Bit (DYB) associated with it. When either bit is “0”, the sector is protected from program and erase operations. The PPB bits are protected from program and erase when the volatile PPB Lock bit is “0”. There are two methods for managing the state of the PPB Lock bit: Password Protection and Persistent Protection. An overview of these methods is shown in Figure 44 on page 67.

Block Protection and ASP protection settings for each sector are logically ORed to define the protection for each sector i.e. if either mechanism is protecting a sector the sector cannot be programmed or erased. Refer to Block Protection on page 65 for full details of the BP2-0 bits.

**Figure 43. Sector Protection Control**

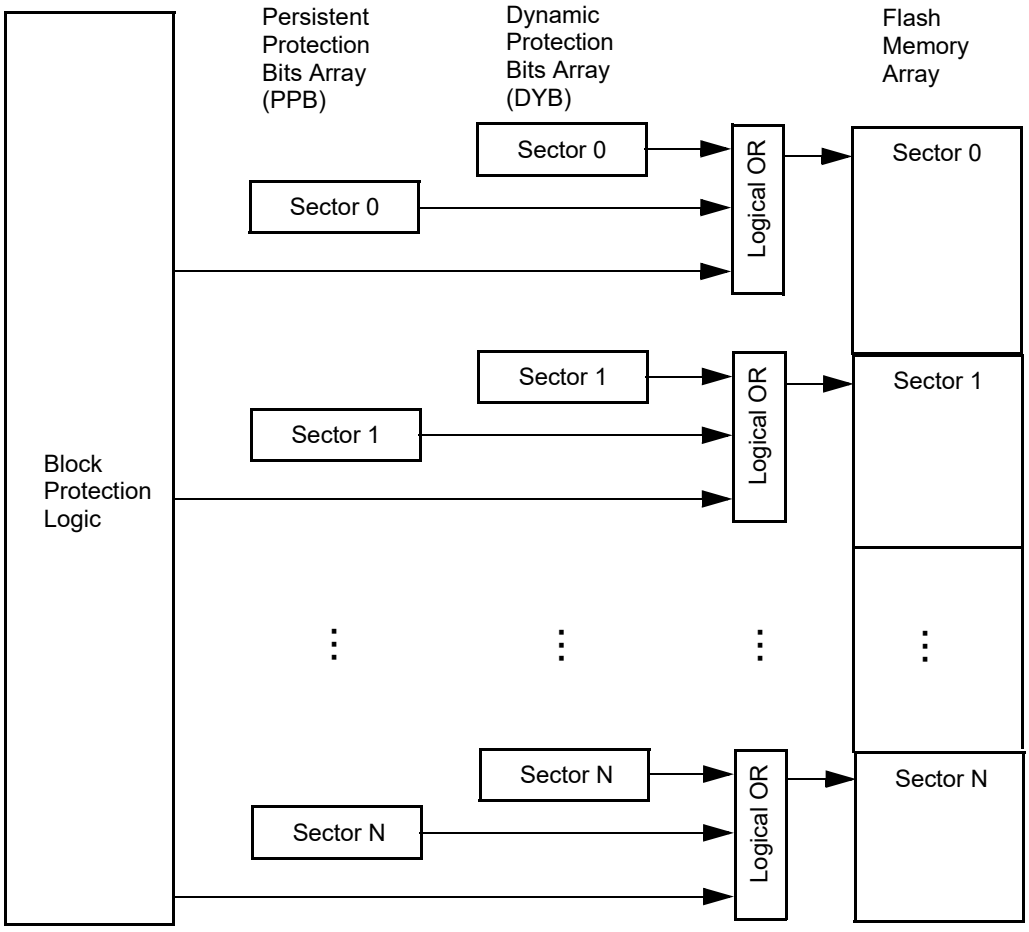
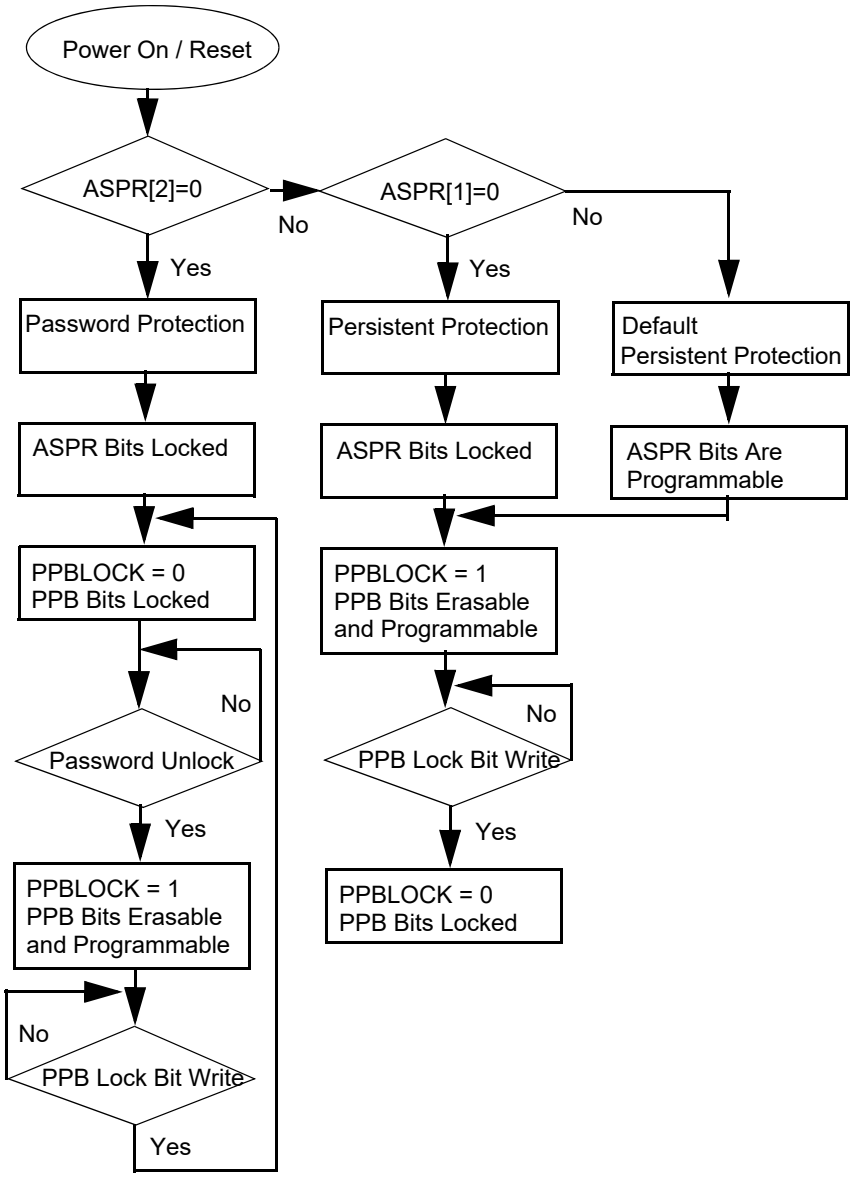


**Figure 44. Advanced Sector Protection Overview**




Password Protection Mode protects the PPB after power up. A password unlock command will enable changes to PPB.
A PPB Lock Bit write command turns protection back on.

Persistent Protection Mode does not protect the PPB after power up. The PPB bits may be changed. A PPB Lock Bit write command protects the PPB bits until the next power off or reset.

Default Mode allows ASPR to be programmed to permanently select the Protection mode.

The default mode otherwise acts the same as the Persistent Protection Mode.

After one of the protection modes is selected, ASPR is no longer programmable, making the selected protection mode permanent.

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. 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 the PPB Lock bit to “0”. This is sometimes called Boot-code controlled sector protection.

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

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

### 10.4.1 ASP Register

The ASP register is used to permanently configure the behavior of Advanced Sector Protection (ASP) features. See Table 41 on page 61.

As shipped from the factory, all devices default ASP to the Persistent Protection mode, with all sectors unprotected, when power is applied. The device programmer or host system must then choose which sector protection method to use. Programming either of the, one-time programmable, Protection Mode Lock Bits, locks the part permanently in the selected mode:

- ASPR[2:1] = “11” = No ASP mode selected, Persistent Protection Mode is the default.
- ASPR[2:1] = “10” = Persistent Protection Mode permanently selected.
- ASPR[2:1] = “01” = Password Protection Mode permanently selected.
- ASPR[2:1] = “00” is an Illegal condition, attempting to program more than one bit to zero results in a programming failure.

ASP register programming rules:

- If the password mode is chosen, the password must be programmed prior to setting the Protection Mode Lock Bits.
- Once the Protection Mode is selected, the following OTP configuration Register bits are permanently protected from programming and no further changes to the OTP register bits is allowed:
  - CR1NV
  - CR2NV
  - CR3NV
  - CR4NV
  - ASPR
  - PASS
  - NVDLR
  - If an attempt to change any of the registers above, after the ASP mode is selected, the operation will fail and P_ERR (SR1V[6]) will be set to 1.

The programming time of the ASP Register is the same as the typical page programming time. The system can determine the status of the ASP register programming operation by reading the WIP bit in the Status Register. See Status Registers 1 on page 49 for information on WIP. See Sector Protection States Summary on page 69.

### 10.4.2 Persistent Protection Bits

The Persistent Protection Bits (PPB) are located in a separate nonvolatile Flash array. One of the PPB bits is related to each sector. When a PPB is “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. The PPB have the same program and erase endurance as the main Flash memory array. Preprogramming and verification prior to erasure are handled by the device.

Programming a PPB bit requires the typical page programming time. Erasing all the PPBs requires typical sector erase time. During PPB bit programming and PPB bit erasing, status is available by reading the Status register. Reading of a PPB bit requires the initial access time of the device.

Notes:

1. Each PPB is individually programmed to “0” and all are erased to “1” in parallel.
2. If the PPB Lock bit is “0”, the PPB Program or PPB Erase command does not execute and fails without programming or erasing the PPB.
3. The state of the PPB for a given sector can be verified by using the PPB Read command.

### 10.4.3 Dynamic Protection Bits

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

### 10.4.4 PPB Lock Bit (PPBL[0])

The PPB Lock Bit is a volatile bit for protecting all PPB bits. When cleared to “0”, it locks all PPBs, when set to “1”, it allows the PPBs to be changed. See PPB Lock Register (PPBL) on page 62 for more information.

The PLBWR command is used to clear the PPB Lock 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 is set to “1” during POR or a hardware reset. When cleared to “0”, no software command sequence can set the PPB Lock bit to “1”, 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 be set to “1” by the Password Unlock command.

### 10.4.5 Sector Protection States Summary

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

- Unlocked — The sector is unprotected and protection can be changed by a simple command. The protection state defaults to unprotected when the device is shipped from Cypress.
- 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 reset.
- Persistently Locked — A sector is protected and protection can only be changed if the PPB Lock Bit is set to “1”. The protection state is non-volatile and saved across a power cycle or reset. Changing the protection state requires programming and or erase of the PPB bits

**Table 50. Sector Protection States**

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

### 10.4.6 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 hardware reset. Software reset does not affect the PPB Lock bit. The PLBWR command can clear the PPB Lock bit to “0” to protect the PPB. There is no command to set the PPB Lock bit therefore the PPB Lock bit will remain at “0” until the next power-off or hardware reset.

### 10.4.7 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 unlocking the PPB Lock bit. In addition to this password requirement, after power up and hardware reset, the PPB Lock bit is cleared to “0” to ensure protection at power-up. Successful execution of the Password Unlock command by entering the entire password sets the PPB Lock bit to 1, allowing for sector PPB modifications.

Password Protection Notes:

- Once the Password is programmed and verified, the Password Mode (ASPR[2] = 0) must be set in order to prevent reading the password.
- The Password Program Command is only capable of programming “0”s. Programming a “1” after a cell is programmed as a “0” results in the cell left as a “0” with no programming error set.
- 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, Password Read, RDAR, and WRAR commands. These commands will not provide access after the Password lock mode is selected.
- All 64-bit password combinations are valid as a password.
- The Password Mode, once programmed, prevents reading the 64-bit password and further password programming. All further program and read commands to the password region are disabled and these commands are ignored or return undefined data. There is no means to verify what the password is after the Password Mode Lock Bit is selected. Password verification is only allowed before selecting the Password Protection mode.
- The Protection Mode Lock Bits are not erasable.
- The exact password must be entered in order for the unlocking function to occur. 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 P_ERR bit is set to one, the WIP Bit remains set, and the PPB Lock bit remains cleared to 0.
- The Password Unlock command cannot be accepted any faster than once every 100 µs ± 20 µs. This makes it take an unreasonably long time (58 million years) for a hacker to run through all the 64-bit combinations in an attempt to correctly match a password. The Read Status Register 1 command may be used to read the WIP bit to determine when the device has completed the password unlock command or is ready to accept a new password command. When a valid password is provided the password unlock command does not insert the 100 µs delay before returning the WIP bit to zero.
- If the password is lost after selecting the Password Mode, there is no way to set the PPB Lock bit.
- ECC status may only be read from sectors that are readable. In read protection mode the addresses are forced to the boot sector address. ECC status is shown in that sector while read protection mode is active.

## 10.5 Recommended Protection Process

During system manufacture, the Flash device configuration should be defined by:

1. Programming the OTP configuration bits in CR1NV[5, 3:2], CR2NV, CR3NV, and CR4NV as desired.
2. Program the Secure Silicon Region (OTP area) as desired.
3. Program the PPB bits as desired via the PPBP command.
4. Program the Non-Volatile Data Learning Pattern (NVDLR) if it will be used in DDR read commands.
5. Program the Password register (PASS) if password protection will be used.
6. Program the ASP Register as desired, including the selection of the persistent or password ASP protection mode in ASPR[2:1]. It is very important to explicitly select a protection mode so that later accidental or malicious programming of the ASP register and OTP configuration is prevented. This is to ensure that only the intended OTP protection and configuration features are enabled.

During system power up and boot code execution:

1. Trusted boot code can determine whether there is any need to program additional SSR (OTP area) information. If no SSR changes are needed the FREEZE bit (CR1V[0]) can be set to 1 to protect the SSR from changes during the remainder of normal system operation while power remains on.
2. If the persistent protection mode is in use, trusted boot code can determine whether there is any need to modify the persistent (PPB) sector protection via the PPBP or PPBE commands. If no PPB changes are needed the PPBLOCK bit can be cleared to 0 via the PPBL to protect the PPB bits from changes during the remainder of normal system operation while power remains on.
3. The dynamic (DYB) sector protection bits can be written as desired via the DYBAR.

# 11. Commands

All communication between the host system and FS-S Family memory devices is in the form of units called commands.

All commands begin with an instruction that selects the type of information transfer or device operation to be performed. Commands may also have an address, instruction modifier, latency period, data transfer to the memory, or data transfer from the memory. All instruction, address, and data information is transferred sequentially between the host system and memory device.

Command protocols are also classified by a numerical nomenclature using three numbers to reference the transfer width of three command phases:

- instruction;
- address and instruction modifier (mode);
- data.

Single bit wide commands start with an instruction and may provide an address or data, all sent only on the SI/IO0 signal. Data may be sent back to the host serially on the SO/IO1 signal. This is referenced as a 1-1-1 command protocol for single bit width instruction, single bit width address and modifier, single bit data.

Dual Output or Quad Output commands provide an address sent from the host on IO0. Data is returned to the host as bit pairs on IO0 and IO1 or, four bit (nibble) groups on IO0, IO1, IO2, and IO3. This is referenced as 1-1-2 for Dual Output and 1-1-4 for Quad Output command protocols.

Dual or Quad Input / Output (I/O) commands provide an address sent from the host as bit pairs on IO0 and IO1 or, four bit (nibble) groups on IO0, IO1, IO2, and IO3. Data is returned to the host similarly as bit pairs on IO0 and IO1 or, four bit (nibble) groups on IO0, IO1, IO2, and IO3. This is referenced as 1-2-2 for Dual I/O and 1-4-4 for Quad I/O command protocols.

The FS-S Family also supports a QPI mode in which all information is transferred in 4-bit width, including the instruction, address, modifier, and data. This is referenced as a 4-4-4 command protocol.

Commands are structured as follows:

- Each command begins with an eight bit (byte) instruction. However, some read commands are modified by a prior read command, such that the instruction is implied from the earlier command. This is called Continuous Read Mode. When the device is in continuous read mode, the instruction bits are not transmitted at the beginning of the command because the instruction is the same as the read command that initiated the Continuous Read Mode. In Continuous Read mode the command will begin with the read address. Thus, Continuous Read Mode removes eight instruction bits from each read command in a series of same type read commands.
- The instruction may be stand alone or may be followed by address bits to select a location within one of several address spaces in the device. The address may be either a 24 bit or 32 bit, byte boundary, address.
- The Serial Peripheral Interface with Multiple IO provides the option for each transfer of address and data information to be done one, two, or four bits in parallel. This enables a trade off between the number of signal connections (IO bus width) and the speed of information transfer. If the host system can support a two or four bit wide IO bus the memory performance can be increased by using the instructions that provide parallel two bit (dual) or parallel four bit (quad) transfers.
- In legacy SPI Multiple IO mode, the width of all transfers following the instruction are determined by the instruction sent. Following transfers may continue to be single bit serial on only the SI or Serial Output (SO) signals, they may be done in two bit groups per (dual) transfer on the IO0 and IO1 signals, or they may be done in 4 bit groups per (quad) transfer on the IO0-IO3 signals. Within the dual or quad groups the least significant bit is on IO0. More significant bits are placed in significance order on each higher numbered IO signal. Single bits or parallel bit groups are transferred in most to least significant bit order.
- In QPI mode, the width of all transfers, including instructions, is a 4-bit wide (quad) transfer on the IO0-IO3 signals.
- Dual I/O and Quad I/O read instructions send an instruction modifier called mode bits, following the address, to indicate that the next command will be of the same type with an implied, rather than an explicit, instruction. The next command thus does not provide an instruction byte, only a new address and mode bits. This reduces the time needed to send each command when the same command type is repeated in a sequence of commands.
- The address or mode bits may be followed by write data to be stored in the memory device or by a read latency period before read data is returned to the host.
- Read latency may be zero to several SCK cycles (also referred to as dummy cycles).

- All instruction, address, mode, and data information is transferred in byte granularity. Addresses are shifted into the device with the most significant byte first. All data is transferred with the lowest address byte sent first. Following bytes of data are sent in lowest to highest byte address order i.e. the byte address increments.
- All attempts to read the flash memory array during a program, erase, or a write cycle (embedded operations) are ignored. The embedded operation will continue to execute without any affect. A very limited set of commands are accepted during an embedded operation. These are discussed in the individual command descriptions. While a program, erase, or write operation is in progress, it is recommended to check that the Write-In Progress (WIP) bit is 0 before issuing most commands to the device, to ensure the new command can be accepted.
- Depending on the command, the time for execution varies. A command to read status information from an executing command is available to determine when the command completes execution and whether the command was successful.
- Although host software in some cases is used to directly control the SPI interface signals, the hardware interfaces of the host system and the memory device generally handle the details of signal relationships and timing. For this reason, signal relationships and timing are not covered in detail within this software interface focused section of the document. Instead, the focus is on the logical sequence of bits transferred in each command rather than the signal timing and relationships. Following are some general signal relationship descriptions to keep in mind. For additional information on the bit level format and signal timing relationships of commands, see Command Protocol on page 14.
  - The host always controls the Chip Select (CS#), Serial Clock (SCK), and Serial Input (SI/IO0) for single  bit wide transfers. The memory drives Serial Output (SO/IO1) for single  bit read transfers. The host and memory alternately drive the IO0-IO3 signals during Dual and Quad transfers.
  - All commands begin with the host selecting the memory by driving CS# low before the first rising edge of SCK. CS# is kept low throughout a command and when CS# is returned high the command ends. Generally, CS# remains low for eight bit transfer multiples to transfer byte granularity information. Some commands will not be accepted if CS# is returned high not at an 8 bit boundary.

## 11.1 Command Set Summary

### 11.1.1 Extended Addressing

1. Instructions that always require a 4-Byte address, used to access up to 32 Gb of memory.

| Command Name | Function | Instruction (Hex) |
|---|---|---|
| 4READ | Read | 13 |
| 4FAST_READ | Read Fast | 0C |
| 4DOR | Dual Output Read | 3C |
| 4QOR | Quad Output Read | 6C |
| 4DIOR | Dual I/O Read | BC |
| 4QIOR | Quad I/O Read | EC |
| 4DDRQIOR | DDR Quad I/O Read | EE |
| 4PP | Page Program | 12 |
| 4QPP | Quad Page Program | 34 |
| 4P4E | Parameter 4 KB Erase | 21 |
| 4SE | Erase 64 KB | DC |
| 4ECCRD | ECC Status Read | 18 |
| 4DYBRD | DYB Read | E0 |
| 4DYBWR | DYBWR | E1 |
| 4PPBRD | PPB Read | E2 |
| 4PPBP | PPB Program | E3 |

2. A 4 Byte address mode for backward compatibility to the 3 Byte address instructions. The standard 3 Byte instructions can be used in conjunction with a 4 Byte address mode controlled by the Address Length configuration bit (CR2V[7]). The default value of CR2V[7] is loaded from CR2NV[7] (following power up, hardware reset, or software reset), to enable default 3-Byte (24-bit) or 4 Byte (32 bit) addressing. When the address length (CR2V[7]) set to 1, the legacy commands are changed to require 4-Bytes (32-bits) for the address field. The following instructions can be used in conjunction with the 4 Byte address mode configuration to switch from 3-Bytes to 4-Bytes of address field.

| Command Name | Function | Instruction (Hex) |
|---|---|---|
| READ | Read | 03 |
| FAST_READ | Read Fast | 0B |
| DOR | Dual Output Read | 3B |
| QOR | Quad Output Read | 6B |
| DIOR | Dual I/O Read | BB |
| QIOR | Quad I/O Read | EB |
| DDRQIOR | DDR Quad I/O Read) | ED |
| PP | Page Program | 02 |
| QPP | Quad Page Program | 32 |
| P4E | Parameter 4 KB Erase | 20 |
| SE | Erase 64 / 256 KB | D8 |
| RDAR | Read Any Register | 65 |
| WRAR | Write Any Register | 71 |
| EES | Evaluate Erase Status | D0 |
| OTPP | OTP Program | 42 |
| OTPR | OTP Read | 4B |
| ECCRD | ECC Status Read | 19 |
| DYBRD | DYB Read | FA |
| DYBWR | DYBWR | FB |
| PPBRD | PPB Read | FC |
| PPBP | PPB Program | FD |

### 11.1.2 Command Summary by Function

**Table 51. FS-S Family Command Set (sorted by function)**

| Function | Command Name | Command Description | Instruction Value (Hex) | Maximum Frequency (MHz) | Address Length (Bytes) | QPI |
|---|---|---|---|---|---|---|
| Read Device ID | RDID | Read ID (JEDEC Manufacturer ID and JEDEC CFI) | 9F | 133 | 0 | Yes |
| | RSFDP | Read JEDEC Serial Flash Discoverable Parameters | 5A | 50 | 3 | Yes |
| | RDQID | Read Quad ID | AF | 133 | 0 | Yes |
| | RUID | Read Unique ID | 4C | 133 | 0 | Yes |
| Register Access | RDSR1 | Read Status Register-1 | 05 | 133 | 0 | Yes |
| | RDSR2 | Read Status Register-2 | 07 | 133 | 0 | No |
| | RDCR | Read Configuration Register-1 | 35 | 133 | 0 | No |
| | RDAR | Read Any Register | 65 | 133 | 3 or 4 | Yes |
| | WRR | Write Register (Status-1, Configuration-1) | 01 | 133 | 0 | Yes |
| | WRDI | Write Disable | 04 | 133 | 0 | Yes |
| | WREN | Write Enable | 06 | 133 | 0 | Yes |
| | WRAR | Write Any Register | 71 | 133 | 3 or 4 | Yes |
| | CLSR | Clear Status Register-1 - Erase/Program Fail Reset<br>This command may be disabled and the instruction value instead used for a program / erase resume command - see Configuration Register 3 on page 57. | 30 | 133 | 0 | Yes |
| | CLSR | Clear Status Register-1(Alternate instruction) - Erase/Program Fail Reset | 82 | 133 | 0 | Yes |
| | 4BEN | Enter 4 Byte Address Mode | B7 | 133 | 0 | No |
| | SBL | Set Burst Length | C0 | 133 | 0 | No |
| | EES | Evaluate Erase Status | D0 | 133 | 3 or 4 | Yes |
| | ECCRD | ECC Read | 19 | 133 | 3 or 4 | Yes |
| | 4ECCRD | ECC Read | 18 | 133 | 4 | Yes |
| | DLPRD | Data Learning Pattern Read | 41 | 133 | 0 | No |
| | PNVDLR | Program NV Data Learning Register | 43 | 133 | 0 | No |
| | WVDLR | Write Volatile Data Learning Register | 4A | 133 | 0 | No |
| Read Flash Array | READ | Read | 03 | 50 | 3 or 4 | No |
| | 4READ | Read | 13 | 50 | 4 | No |
| | FAST_READ | Fast Read | 0B | 133 | 3 or 4 | No |
| | 4FAST_READ | Fast Read | 0C | 133 | 4 | No |
| | DOR | Dual Output Read | 3B | 133 | 3 or 4 | No |
| | 4DOR | Dual Output Read | 3C | 133 | 4 | No |
| | QOR | Quad Output Read | 6B | 133 | 3 or 4 | No |
| | 4QOR | Quad Output Read | 6C | 133 | 4 | No |
| | DIOR | Dual I/O Read | BB | 66 | 3 or 4 | No |
| | 4DIOR | Dual I/O Read | BC | 66 | 4 | No |
| | QIOR | Quad I/O Read | EB | 133 | 3 or 4 | Yes |
| | 4QIOR | Quad I/O Read | EC | 133 | 4 | Yes |
| | DDRQIOR | DDR Quad I/O Read | ED | 80 | 3 or 4 | Yes |
| | 4DDRQIOR | DDR Quad I/O Read | EE | 80 | 4 | Yes |
| Program Flash Array | PP | Page Program | 02 | 133 | 3 or 4 | Yes |
| | 4PP | Page Program | 12 | 133 | 4 | Yes |
| | QPP | Quad Page Program | 32 | 133 | 3 or 4 | No |
| | 4QPP | Quad Page Program | 34 | 133 | 4 | No |

**Table 51. FS-S Family Command Set (sorted by function)** (Continued)

| Function | Command Name | Command Description | Instruction Value (Hex) | Maximum Frequency (MHz) | Address Length (Bytes) | QPI |
|---|---|---|---|---|---|---|
| Erase Flash Array | P4E | Parameter 4KB-sector Erase | 20 | 133 | 3 or 4 | Yes |
| | 4P4E | Parameter 4KB-sector Erase | 21 | 133 | 4 | Yes |
| | SE | Erase 64KB | D8 | 133 | 3 or 4 | Yes |
| | 4SE | Erase 64KB | DC | 133 | 4 | Yes |
| | BE | Bulk Erase | 60 | 133 | 0 | Yes |
| | BE | Bulk Erase (alternate instruction) | C7 | 133 | 0 | Yes |
| Erase / Program Suspend / Resume | EPS | Erase / Program Suspend | 75 | 133 | 0 | Yes |
| | EPS | Erase / Program Suspend (alternate instruction) | 85 | 133 | 0 | Yes |
| | EPS | Erase / Program Suspend (alternate instruction | B0 | 133 | 0 | Yes |
| | EPR | Erase / Program Resume | 7A | 133 | 0 | Yes |
| | EPR | Erase / Program Resume (alternate instruction) | 8A | 133 | 0 | Yes |
| | EPR | Erase / Program Resume (alternate instruction<br>This command may be disabled and the instruction value instead used for a clear status command - see Configuration Register 3 on page 57. | 30 | 133 | 0 | Yes |
| One Time Program Array | OTPP | OTP Program | 42 | 133 | 3 or 4 | No |
| | OTPR | OTP Read | 4B | 133 | 3 or 4 | No |
| Advanced Sector Protection | DYBRD | DYB Read | FA | 133 | 3 or 4 | Yes |
| | 4DYBRD | DYB Read | E0 | 133 | 4 | Yes |
| | DYBWR | DYB Write | FB | 133 | 3 or 4 | Yes |
| | 4DYBWR | DYB Write | E1 | 133 | 4 | Yes |
| | PPBRD | PPB Read | FC | 133 | 3 or 4 | No |
| | 4PPBRD | PPB Read | E2 | 133 | 4 | No |
| | PPBP | PPB Program | FD | 133 | 3 or 4 | No |
| | 4PPBP | PPB Program | E3 | 133 | 4 | No |
| | PPBE | PPB Erase | E4 | 133 | 0 | No |
| | ASPRD | ASP Read | 2B | 133 | 0 | No |
| | ASPP | ASP Program | 2F | 133 | 0 | No |
| | PLBRD | PPB Lock Bit Read | A7 | 133 | 0 | No |
| | PLBWR | PPB Lock Bit Write | A6 | 133 | 0 | No |
| | PASSRD | Password Read | E7 | 133 | 0 | No |
| | PASSP | Password Program | E8 | 133 | 0 | No |
| | PASSU | Password Unlock | E9 | 133 | 0 | No |
| Reset | RSTEN | Software Reset Enable | 66 | 133 | 0 | Yes |
| | RST | Software Reset | 99 | 133 | 0 | Yes |
| | RESET | Legacy Software Reset | F0 | 133 | 0 | No |
| | MBR | Mode Bit Reset | FF | 133 | 0 | Yes |
| DPD | DPD | Enter Deep Power Down Mode | B9 | 133 | 0 | Yes |
| | RES | Release from Deep Power Down Mode | AB | 133 | 0 | Yes |

**Note**
47.Commands not supported in QPI mode have undefined behavior if sent when the device is in QPI mode.

### 11.1.3 Read Device Identification

There are multiple commands to read information about the device manufacturer, device type, and device features. SPI memories from different vendors have used different commands and formats for reading information about the memories. The FS-S Family supports the three device information commands.

### 11.1.4 Register Read or Write

There are multiple registers for reporting embedded operation status or controlling device configuration options. There are commands for reading or writing these registers. Registers contain both volatile and non-volatile bits. Non-volatile bits in registers are automatically erased and programmed as a single (write) operation.

#### 11.1.4.1 Monitoring Operation Status

The host system can determine when a write, program, erase, suspend or other embedded operation is complete by monitoring the Write in Progress (WIP) bit in the Status Register. The Read from Status Register-1 command or Read Any Register command provides the state of the WIP bit. The program error (P_ERR) and erase error (E_ERR) bits in the status register indicate whether the most recent program or erase command has not completed successfully. When P_ERR or E_ERR bits are set to one, the WIP bit will remain set to one indicating the device remains busy and unable to receive most new operation commands. Only status read (RDSR1 05h), Read Any Register (RDAR 65h), status clear (CLSR 30h or 82h), and software reset (RSTEN 66h, RST 99h or RESET F0h) are valid commands when P_ERR or E_ERR is set to 1. A Clear Status Register (CLSR) followed by a Write Disable (WRDI) command must be sent to return the device to standby state. Clear Status Register clears the WIP, P_ERR, and E_ERR bits. WRDI clears the WEL bit. Alternatively, Hardware Reset, or Software Reset (RST or RESET) may be used to return the device to standby state.

#### 11.1.4.2 Configuration

There are commands to read, write, and protect registers that control interface path width, interface timing, interface address length, and some aspects of data protection.

### 11.1.5 Read Flash Array

Data may 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.

Burst Wrap read can be enabled by the Set Burst Length (SBL 77h) command with the requested wrapped read length and alignment, see Set Burst Length (SBL C0h) on page 91. Burst Wrap read is only for Quad I/O, Quad Output and QPI modes.

There are several different read commands to specify different access latency and data path widths. Double Data Rate (DDR) commands also define the address and data bit relationship to both SCK edges:

- The Read command provides a single address bit per SCK rising edge on the SI/IO0 signal with read data returning a single bit per SCK falling edge on the SO/IO1signal. This command has zero latency between the address and the returning data but is limited to a maximum SCK rate of 50MHz.
- Other read commands have a latency period between the address and returning data but can operate at higher SCK frequencies. The latency depends on a configuration register read latency value.
- The Fast Read command provides a single address bit per SCK rising edge on the SI/IO0 signal with read data returning a single bit per SCK falling edge on the SO/IO1 signal.
- Dual or Quad Output Read commands provide address on SI/IO0 pin on the SCK rising edge with read data returning two bits, or four bits of data per SCK falling edge on the IO0 - IO3 signals.
- Dual or Quad I/O Read commands provide address two bits or four bits per SCK rising edge with read data returning two bits, or four bits of data per SCK falling edge on the IO0 - IO3 signals.
- Quad Double Data Rate read commands provide address four bits per every SCK edge with read data returning four bits of data per every SCK edge on the IO0 - IO3 signals.

### 11.1.6 Program Flash Array

Programming data requires two commands: Write Enable (WREN), Page Program (PP) and Quad Page Program (QPP). The Page Program and Quad Page Program commands accepts from 1 byte up to 256 or 512 consecutive bytes of data (page) to be programmed in one operation. Programming means that bits can either be left at 1, or programmed from 1 to 0. Changing bits from 0 to 1 requires an erase operation.

### 11.1.7 Erase Flash Array

The Parameter Sector Erase, Sector Erase, or Bulk Erase commands set all the bits in a sector or the entire memory array to 1. A bit needs to be first erased to 1 before programming can change it to a 0. While bits can be individually programmed from a 1 to 0, erasing bits from 0 to 1 must be done on a sector-wide or array-wide (bulk) level. The Write Enable (WREN) command must precede an erase command.

### 11.1.8 OTP, Block Protection, and Advanced Sector Protection

There are commands to read and program a separate One Time Programmable (OTP) array for permanent data such as a serial number. There are commands to control a contiguous group (block) of Flash memory array sectors that are protected from program and erase operations.There are commands to control which individual Flash memory array sectors are protected from program and erase operations.

### 11.1.9 Reset

There are commands to reset to the default conditions present after power on to the device. However, the software reset commands do not affect the current state of the FREEZE or PPB Lock bits. In all other respects a software reset is the same as a hardware reset.

There is a command to reset (exit from) the Continuous Read Mode.

### 11.1.10 DPD

A Deep Power Down (DPD) mode is supported by the FS-S Family devices. If the device has been placed in DPD mode by the DPD (B9h) command, the interface standby current is ($I_{DPD}$). The DPD command is accepted only while the device is not performing an embedded algorithm as indicated by the Status Register-1 volatile Write In Progress (WIP) bit being cleared to zero (SR1V[0] = 0). While in DPD mode the device ignores all commands except the Release from DPD (RES ABh) command, that will return the device to the Interface Standby state after a delay of $t_{RES}$.

### 11.1.11 Reserved

Some instructions are reserved for future use. In this generation of the FS-S Family some of these command instructions may be unused and not affect device operation, some may have undefined results.

Some commands are reserved to ensure that a legacy or alternate source device command is allowed without effect. This allows legacy software to issue some commands that are not relevant for the current generation FS-S Family with the assurance these commands do not cause some unexpected action.

Some commands are reserved for use in special versions of the FS-S not addressed by this document or for a future generation. This allows new host memory controller designs to plan the flexibility to issue these command instructions. The command format is defined if known at the time this document revision is published.

## 11.2 Identification Commands

### 11.2.1 Read Identification (RDID 9Fh)

The Read Identification (RDID) command provides read access to manufacturer identification, device identification, and Common Flash Interface (CFI) information. The manufacturer identification is assigned by JEDEC. The CFI structure is defined by JEDEC standard. The device identification and CFI values are assigned by Cypress.

The JEDEC Common Flash Interface (CFI) specification defines a device information structure, which allows a vendor-specified software Flash management program (driver) to be used for entire families of Flash devices. Software support can then be device-independent, JEDEC manufacturer ID independent, forward and backward-compatible for the specified Flash device families. System vendors can standardize their Flash drivers for long-term software compatibility by using the CFI values to configure a family driver from the CFI information of the device in use.

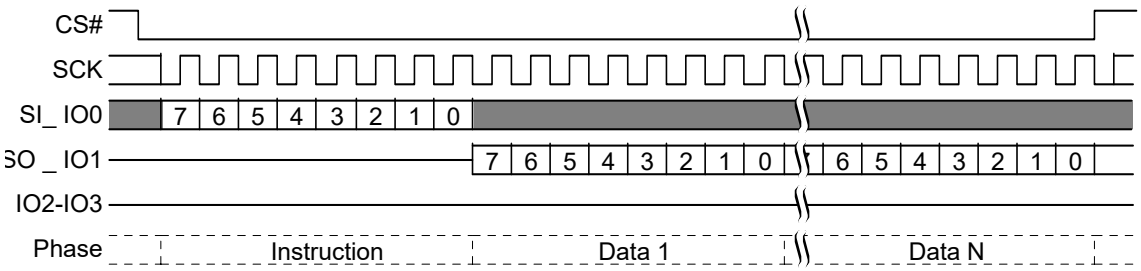
Any RDID command issued while a program, erase, or write cycle is in progress is ignored and has no effect on execution of the program, erase, or write cycle that is in progress.

The RDID instruction is shifted on SI. After the last bit of the RDID instruction is shifted into the device, a byte of manufacturer identification, two bytes of device identification, extended device identification, and CFI information will be shifted sequentially out on SO. As a whole this information is referred to as ID-CFI. See Device ID and Common Flash Interface (ID-CFI) Address Map — Standard on page 122 for the detail description of the ID-CFI contents.

Continued shifting of output beyond the end of the defined ID-CFI address space will provide undefined data. The RDID command sequence is terminated by driving CS# to the logic high state anytime during data output.

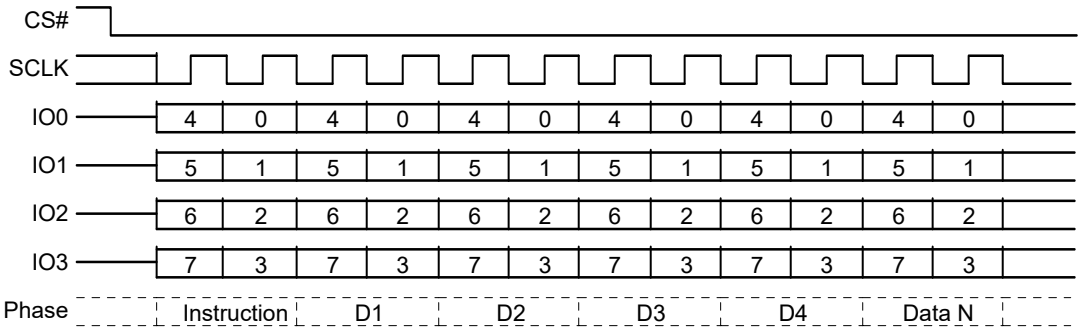
The maximum clock frequency for the RDID command is 133 MHz.

**Figure 45. Read Identification (RDID) Command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0-IO3 and the returning data is shifted out on IO0-IO3.

**Figure 46. Read Identification (RDID) QPI Mode Command**



### 11.2.2 Read Quad Identification (RDQID AFh)

The Read Quad Identification (RDQID) command provides read access to manufacturer identification, device identification, and Common Flash Interface (CFI) information. This command is an alternate way of reading the same information provided by the RDID command while in QPI mode. In all other respects the command behaves the same as the RDID command.

The command is recognized only when the device is in QPI Mode (CR2V[6]=1). The instruction is shifted in on IO0-IO3. After the last bit of the instruction is shifted into the device, a byte of manufacturer identification, two bytes of device identification, extended device identification, and CFI information will be shifted sequentially out on IO0-IO3. As a whole this information is referred to as ID-CFI. See Device ID and Common Flash Interface (ID-CFI) Address Map — Standard on page 122 for the detail description of the ID-CFI contents.

Continued shifting of output beyond the end of the defined ID-CFI address space will provide undefined data. The command sequence is terminated by driving CS# to the logic high state anytime during data output.

The maximum clock frequency for the command is 133 MHz. Command sequence is terminated by driving CS# to the logic high state anytime during data output.

**Figure 47. Read Quad Identification (RDQID) Command Sequence Quad Mode**

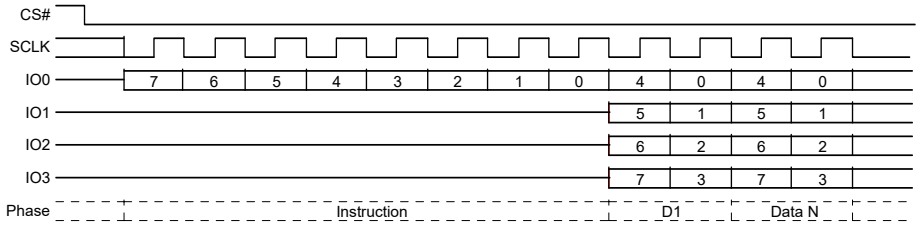
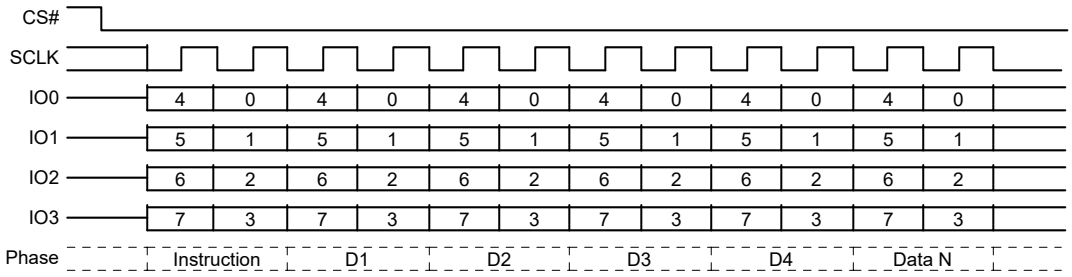


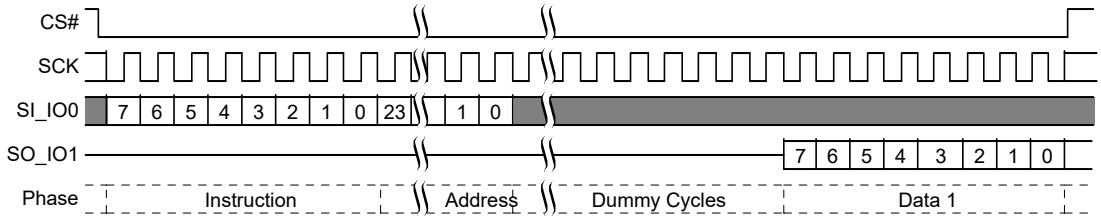

**Figure 48. Read Quad Identification (RDQID) Command Sequence QPI Mode**




### 11.2.3 Read Serial Flash Discoverable Parameters (RSFDP 5Ah)

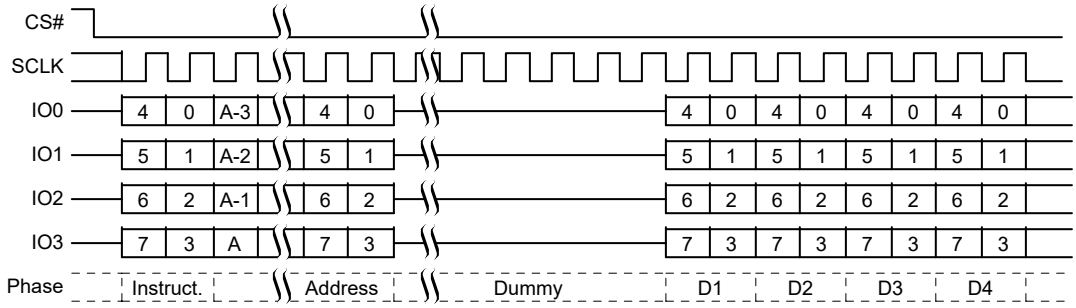
The command is initiated by shifting on SI the instruction code “5Ah”, followed by a 24-bit address of 000000h, followed by 8 dummy cycles. The SFDP bytes are then shifted out on SO starting at the falling edge of SCK after the dummy cycles. The SFDP bytes are always shifted out with the MSB first. If the 24-bit address is set to any other value, 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. The RSFDP command is supported up to 50 MHz.

**Figure 49. RSFDP Command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0-IO3 and the returning data is shifted out on IO0-IO3.

**Figure 50. RSFDP QPI Mode Command Sequence**




### 11.2.4 Read Unique ID (RUID 4Ch)

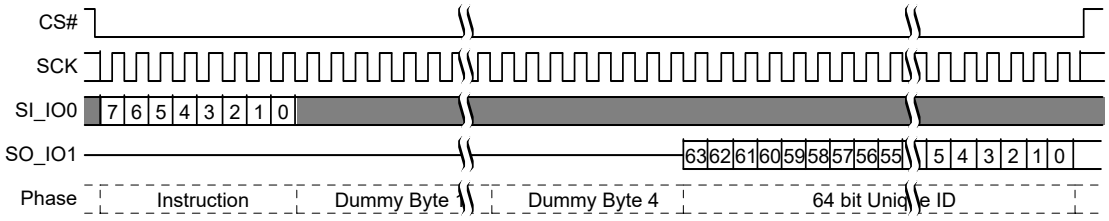
The Read Identification (RUID) command provides read access to factory set read only 64 bit number that is unique to each device.

The RUID instruction is shifted on SI followed by four dummy bytes or 16 dummy bytes QPI (32 clock cycles). This latency period (i.e., dummy bytes) allows the device's internal circuitry enough time to access data at the initial address. During latency cycles, the data value on IO0-IO3 are "don't care" and may be high impedance.

Then the 8 bytes of Unique ID will be shifted sequentially out on SO / IO1.

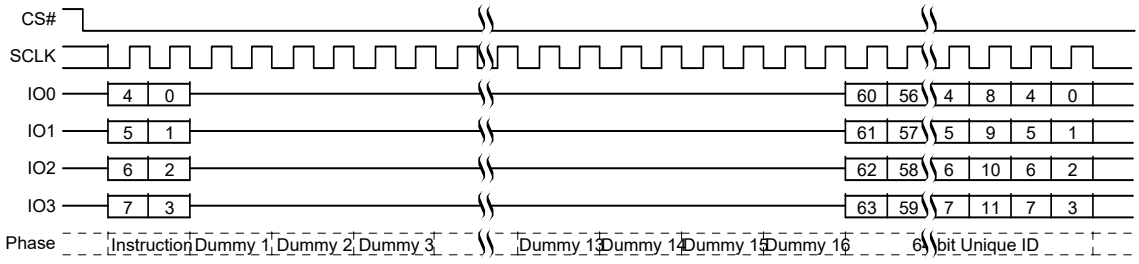
Continued shifting of output beyond the end of the defined Unique ID address space will provide undefined data. The RUID command sequence is terminated by driving CS# to the logic high state anytime during data output.

**Figure 51. Read Unique ID (RUID) Command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0–IO3 and the returning data is shifted out on IO0–IO3.

**Figure 52. Read Unique ID (RUID) QPI Mode Command**



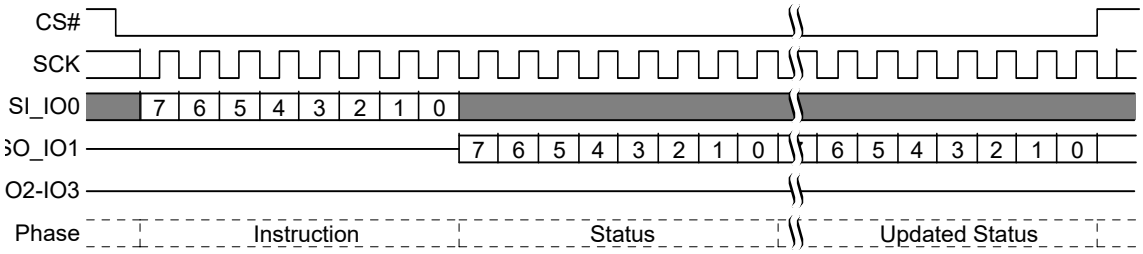
## 11.3 Register Access Commands

### 11.3.1 Read Status Register-1 (RDSR1 05h)

The Read Status Register-1 (RDSR1) command allows the Status Register-1 contents to be read from SO/IO1.

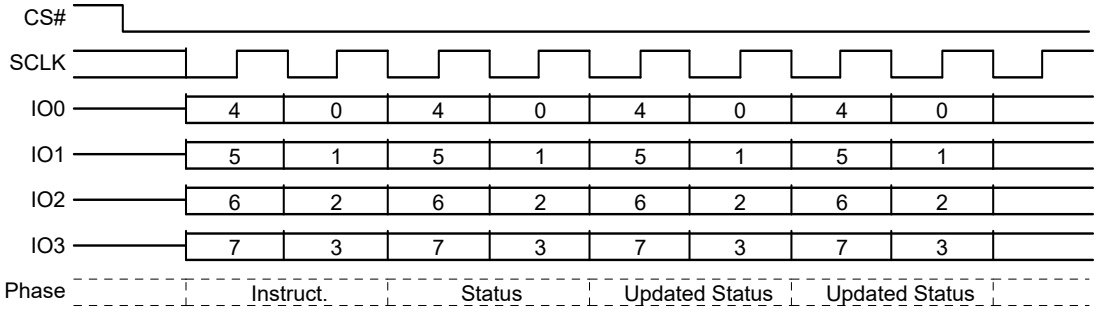
The volatile version of Status Register-1 (SR1V) contents may be read at any time, even while a program, erase, or write operation is in progress. It is possible to read Status Register-1 continuously by providing multiples of eight clock cycles. The status is updated for each eight cycle read. The maximum clock frequency for the RDSR1 (05h) command is 133 MHz.

**Figure 53. Read Status Register-1 (RDSR1) Command Sequence**




This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3 and the returning data is shifted out on IO0–IO3.

**Figure 54. Read Status Register-1 (RDSR1) QPI Mode Command**

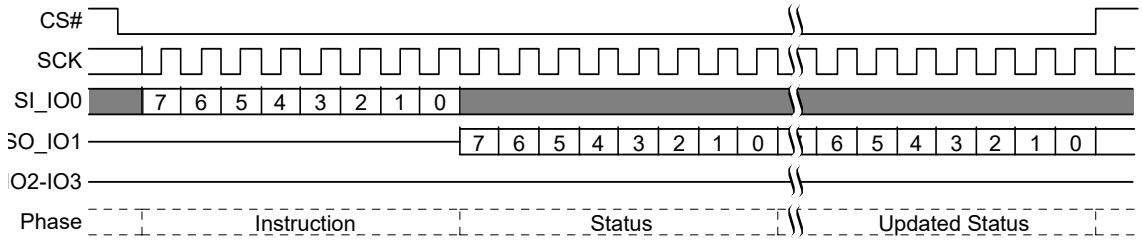



### 11.3.2 Read Status Register-2 (RDSR2 07h)

The Read Status Register-2 (RDSR2) command allows the Status Register-2 contents to be read from SO/IO1.

The Status Register-2 contents may be read at any time, even while a program, erase, or write operation is in progress. It is possible to read the Status Register-2 continuously by providing multiples of eight clock cycles. The status is updated for each eight cycle read. The maximum clock frequency for the RDSR2 command is 133 MHz.

**Figure 55. Read Status Register-2 (RDSR2) Command**



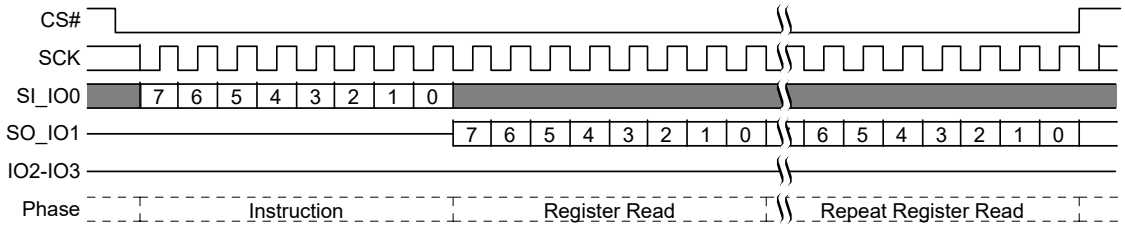

In QPI mode, status register 2 may be read via the Read Any Register command, see Read Any Register (RDAR 65h) on page 89.

### 11.3.3 Read Configuration Register (RDCR 35h)

The Read Configuration Register (RDCR) commands allows the volatile Configuration Registers (CR1V) contents to be read from SO/IO1.

It is possible to read CR1V continuously by providing multiples of eight clock cycles. The Configuration Register contents may be read at any time, even while a program, erase, or write operation is in progress.

**Figure 56. Read Configuration Register (RDCR) Command Sequence**




In QPI mode, configuration register 1 may be read via the Read Any Register command, see Read Any Register (RDAR 65h) on page 89

### 11.3.4 Write Registers (WRR 01h)

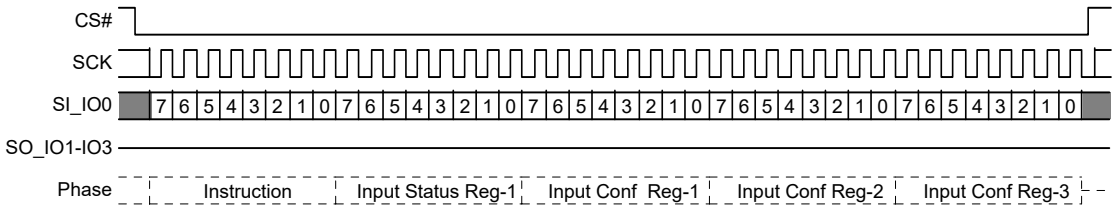
The Write Registers (WRR) command allows new values to be written to both the Status Register 1 and Configuration Register 1. Before the Write Registers (WRR) command can be accepted by the device, a Write Enable (WREN) command must be received. After the Write Enable (WREN) command has been decoded successfully, the device will set the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The Write Registers (WRR) command is entered by shifting the instruction and the data bytes on SI/IO0. The Status Register is one data byte in length.

The WRR operation first erases the register then programs the new value as a single operation. The Write Registers (WRR) command will set the P_ERR or E_ERR bits if there is a failure in the WRR operation. See Status Register 1 Volatile (SR1V) on page 50 for a description of the error bits. Any Status or Configuration Register bit reserved for the future must be written as a “0”.

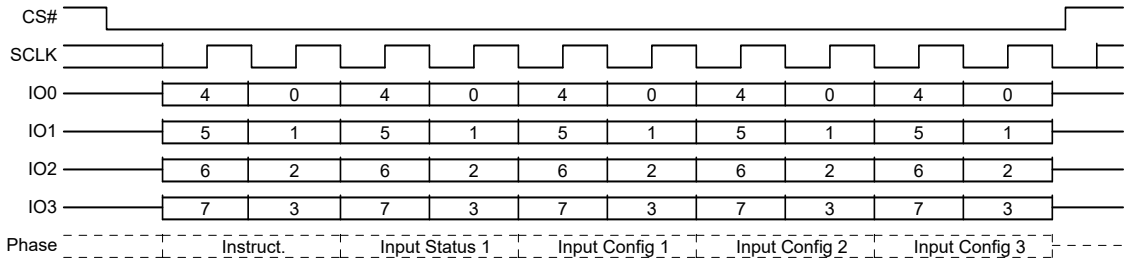
CS# must be driven to the logic high state after the eighth or sixteenth bit of data has been latched. If not, the Write Registers (WRR) command is not executed. If CS# is driven high after the eighth cycle then only the Status Register 1 is written; otherwise, after the sixteenth cycle both the Status and Configuration Registers are written. As soon as CS# is driven to the logic high state, the self-timed Write Registers (WRR) operation is initiated. While the Write Registers (WRR) operation is in progress, the Status Register may still be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a “1” during the self-timed Write Registers (WRR) operation, and is a “0” when it is completed. When the Write Registers (WRR) operation is completed, the Write Enable Latch (WEL) is set to a “0”. The maximum clock frequency for the WRR command is 133 MHz.

**Figure 57. Write Register (WRR) Command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction and data is shifted in on IO0-IO3.

**Figure 58. Write Register (WRR) Command Sequence QPI**




The Write Registers (WRR) command allows the user to change the values of the Block Protect (BP2, BP1, and BP0) bits in either the non-volatile Status Register 1 or in the volatile Status Register 1, to define the size of the area that is to be treated as read-only. The BPNV_O bit (CR1NV[3]) controls whether WRR writes the non-volatile or volatile version of Status Register 1. When CR1NV[3] = 0 WRR writes SR1NV[4:2]. When CR1NV[3] = 1 WRR writes SR1V[4:2].

The Write Registers (WRR) command also allows the user to set the Status Register Write Disable (SRWD) bit to a “1” or a “0”. The Status Register Write Disable (SRWD) bit and Write Protect (WP#) signal allow the BP bits to be hardware protected.

When the Status Register Write Disable (SRWD) bit of the Status Register is a “0” (its initial delivery state), it is possible to write to the Status Register provided that the Write Enable Latch (WEL) bit has previously been set by a Write Enable (WREN) command, regardless of the whether Write Protect (WP#) signal is driven to the logic high or logic low state.

When the Status Register Write Disable (SRWD) bit of the Status Register is set to a “1”, two cases need to be considered, depending on the state of Write Protect (WP#):

- If Write Protect (WP#) signal is driven to the logic high state, it is possible to write to the Status and Configuration Registers provided that the Write Enable Latch (WEL) bit has previously been set to a “1” by initiating a Write Enable (WREN) command.
- If Write Protect (WP#) signal is driven to the logic low state, it is not possible to write to the Status and Configuration Registers even if the Write Enable Latch (WEL) bit has previously been set to a “1” by a Write Enable (WREN) command. Attempts to write to the Status and Configuration Registers are rejected, not accepted for execution, and no error indication is provided. As a consequence, all the data bytes in the memory area that are protected by the Block Protect (BP2, BP1, BP0) bits of the Status Register, are also hardware protected by WP#.

The WP# hardware protection can be provided:

- by setting the Status Register Write Disable (SRWD) bit after driving Write Protect (WP#) signal to the logic low state;
- or by driving Write Protect (WP#) signal to the logic low state after setting the Status Register Write Disable (SRWD) bit to a “1”.

The only way to release the hardware protection is to pull the Write Protect (WP#) signal to the logic high state. If WP# is permanently tied high, hardware protection of the BP bits can never be activated.

**Table 52. Block Protection Modes**

| WP# | SRWD Bit | Mode | Write Protection of Registers | Memory Content | |
|---|---|---|---|---|---|
| | | | | Protected Area | Unprotected Area |
| 1 | 1 | Software Protected | Status and Configuration Registers are Writable (if WREN command has set the WEL bit). The values in the SRWD, BP2, BP1, and BP0 bits and those in the Configuration Register can be changed | Protected against Page Program, Sector Erase, and Bulk Erase | Ready to accept Page Program, and Sector Erase commands |
| 1 | 0 | | | | |
| 0 | 0 | | | | |
| 0 | 1 | Hardware Protected | Status and Configuration Registers are Hardware Write Protected. The values in the SRWD, BP2, BP1, and BP0 bits and those in the Configuration Register cannot be changed | Protected against Page Program, Sector Erase, and Bulk Erase | Ready to accept Page Program or Erase commands |

**Notes**

48.The Status Register originally shows 00h when the device is first shipped from Cypress to the customer.

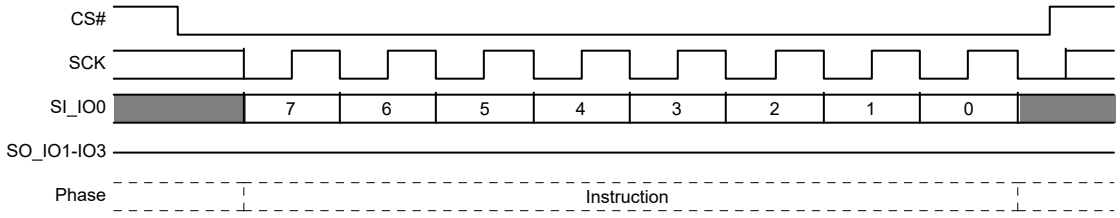
49.Hardware protection is disabled when Quad Mode is enabled (CR1V[1] = 1). WP# becomes IO2; therefore, it cannot be utilized.

### 11.3.5 Write Enable (WREN 06h)

The Write Enable (WREN) command sets the Write Enable Latch (WEL) bit of the Status Register 1 (SR1V[1]) to a “1”. The Write Enable Latch (WEL) bit must be set to a “1” by issuing the Write Enable (WREN) command to enable write, program and erase commands.

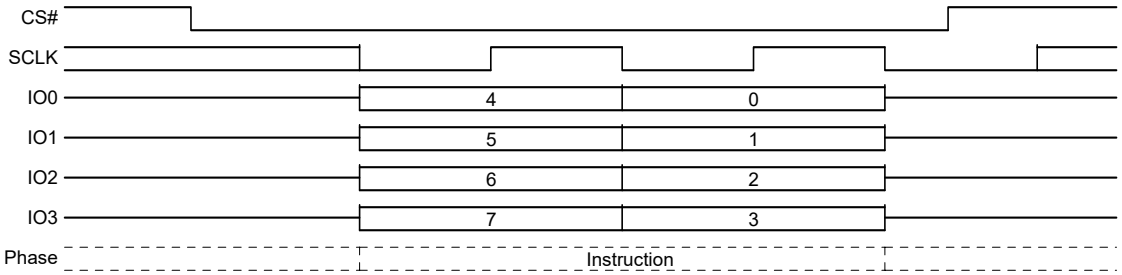
CS# must be driven into the logic high state after the eighth bit of the instruction byte has been latched in on SI/IO0. Without CS# being driven to the logic high state after the eighth bit of the instruction byte has been latched in on SI/IO0, the write enable operation will not be executed.

**Figure 59. Write Enable (WREN) Command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0-IO3.

**Figure 60. Write Enable (WREN) Command Sequence QPI Mode**



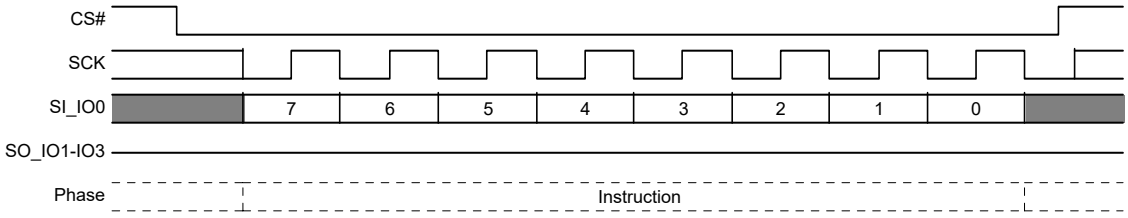

### 11.3.6 Write Disable (WRDI 04h)

The Write Disable (WRDI) command clears the Write Enable Latch (WEL) bit of the Status Register-1 (SR1V[1]) to a “0”.

The Write Enable Latch (WEL) bit may be cleared to a “0” by issuing the Write Disable (WRDI) command to disable Page Program (PP), Sector Erase (SE), Bulk Erase (BE), Write Registers (WRR or WRAR), OTP Program (OTPP), and other commands, that require WEL be set to “1” for execution. The WRDI command can be used by the user to protect memory areas against inadvertent writes that can possibly corrupt the contents of the memory. The WRDI command is ignored during an embedded operation while WIP bit =1.

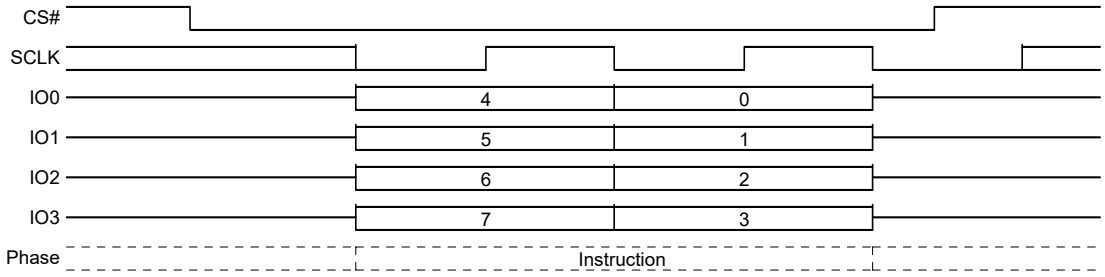
CS# must be driven into the logic high state after the eighth bit of the instruction byte has been latched in on SI/IO0. Without CS# being driven to the logic high state after the eighth bit of the instruction byte has been latched in on SI/IO0, the write disable operation will not be executed.

**Figure 61. Write Disable (WRDI) Command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0–IO3.

**Figure 62. Write Disable (WRDI) Command Sequence QPI Mode**

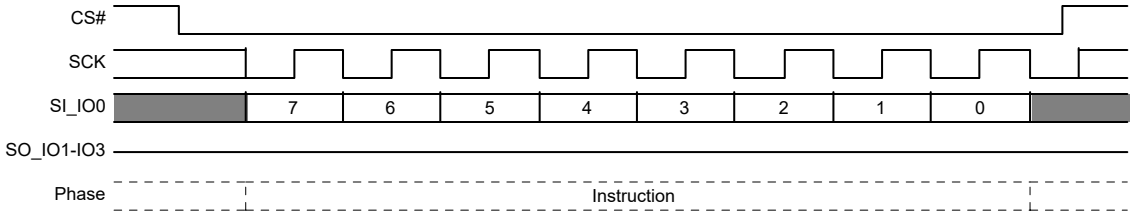



### 11.3.7 Clear Status Register (CLSR 30h or 82h)

The Clear Status Register command resets bit SR1V[5] (Erase Fail Flag) and bit SR1V[6] (Program Fail Flag). It is not necessary to set the WEL bit before a Clear Status Register command is executed. The Clear Status Register command will be accepted even when the device remains busy with WIP set to 1, as the device does remain busy when either error bit is set. The WEL bit will be unchanged after this command is executed.

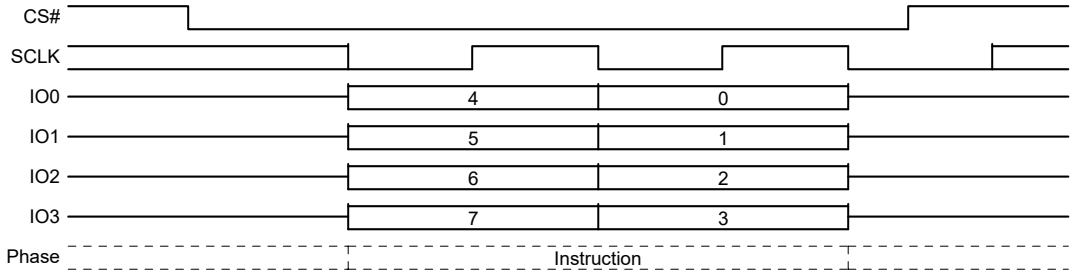
The legacy Clear Status Register (CLSR 30h) instruction may be disabled and the 30h instruction value instead used for a program / erase resume command - see Section 9.6.5 Configuration Register 3 on page 57. The Clear Status Register alternate instruction (CLSR 82h) is always available to clear the status register.

**Figure 63. Clear Status Register (CLSR) Command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0-IO3.

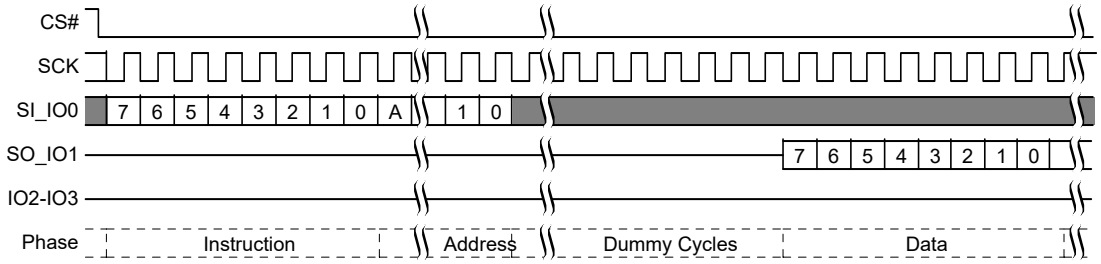
**Figure 64. Clear Status Register (CLSR) QPI Mode**



### 11.3.8 ECC Status Register Read (ECCRD 19h or 4EECRD 18h)

To read the ECC Status Register, the command is followed by the ECC unit (16 Bytes) address, the four least significant bits (LSB) of address must be set to zero. This is followed by the number of dummy cycles selected by the read latency value in CR2V[3:0]. Then the 8-bit contents of the ECC Register, for the ECC unit selected, are shifted out on SO/IO1 16 times, once for each byte in the ECC Unit. If CS# remains low the next ECC unit status is sent through SO/IO1 16 times, once for each byte in the ECC Unit, this continues until CS# goes high. The maximum operating clock frequency for the ECC READ command is 133 MHz. See Automatic ECC on page 101 for details on ECC unit.

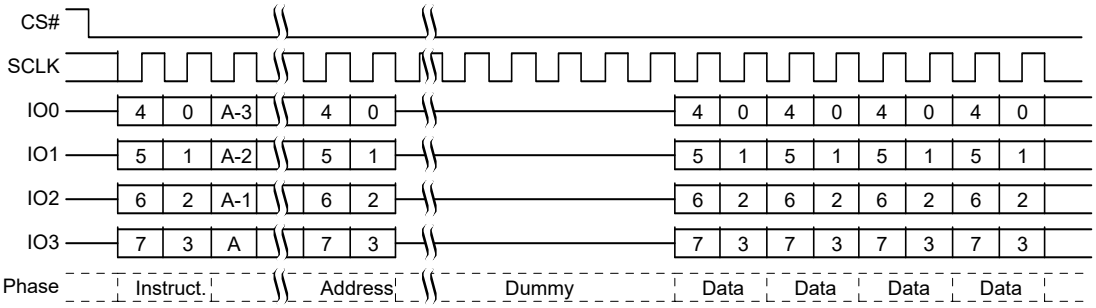
**Figure 65. ECC Status Register Read Command Sequence**




**Notes:**
50.A = MSB of address = 23 for Address length (CR2V[7] = 0, or 31 for CR2V[7]=1 with command 19h.
51.A = MSB of address = 31 with command 18h

This command is also supported in QPI mode. In QPI mode the instruction and address is shifted in and returning data out on IO0-IO3.

**Figure 66. ECC Status Register Read QPI Mode, Command Sequence**




**Notes:**
52.A = MSB of address = 23 for Address length (CR2V[7] = 0, or 31 for CR2V[7]=1 with command 19h.
53.A = MSB of address = 31 with command 18h

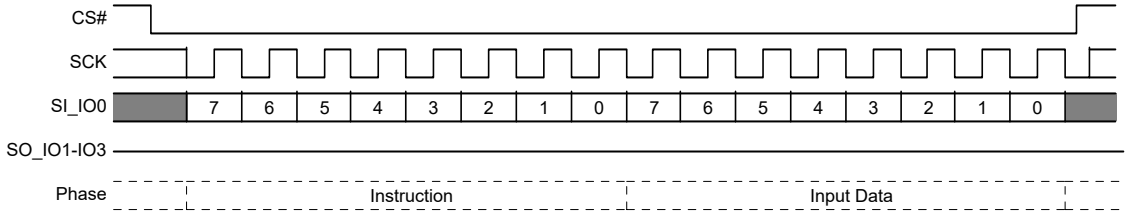
### 11.3.9 Program NVDLR (PNVDLR 43h)

Before the Program NVDLR (PNVDLR) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device. After the Write Enable (WREN) command has been decoded successfully, the device will set the Write Enable Latch (WEL) to enable the PNVDLR operation.

The PNVDLR command is entered by shifting the instruction and the data byte on SI/IO0.

CS# must be driven to the logic high state after the eighth (8th) bit of data has been latched. If not, the PNVDLR command is not executed. As soon as CS# is driven to the logic high state, the self-timed PNVDLR operation is initiated. While the PNVDLR operation is in progress, the Status Register may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a “1” during the self-timed PNVDLR cycle, and is a 0. when it is completed. The PNVDLR operation can report a program error in the P_ERR bit of the status register. When the PNVDLR operation is completed, the Write Enable Latch (WEL) is set to a “0” The maximum clock frequency for the PNVDLR command is 133 MHz.

**Figure 67. Program NVDLR (PNVDLR) Command Sequence**

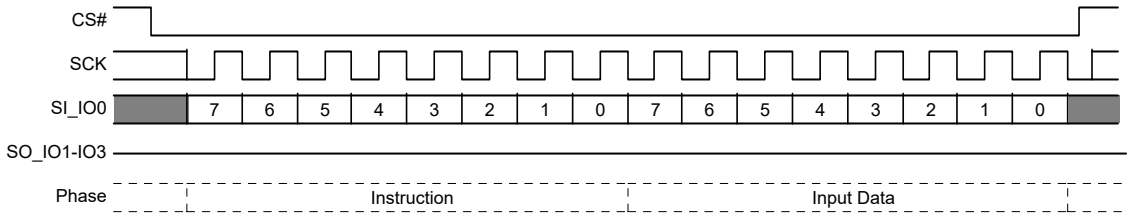



### 11.3.10 Write VDLR (WVDLR 4Ah)

Before the Write VDLR (WVDLR) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device. After the Write Enable (WREN) command has been decoded successfully, the device will set the Write Enable Latch (WEL) to enable WVDLR operation.

The WVDLR command is entered by shifting the instruction and the data byte on SI/IO0. CS# must be driven to the logic high state after the eighth (8th) bit of data has been latched. If not, the WVDLR command is not executed. As soon as CS# is driven to the logic high state, the WVDLR operation is initiated with no delays. The maximum clock frequency for the WVDLR command is 133 MHz.

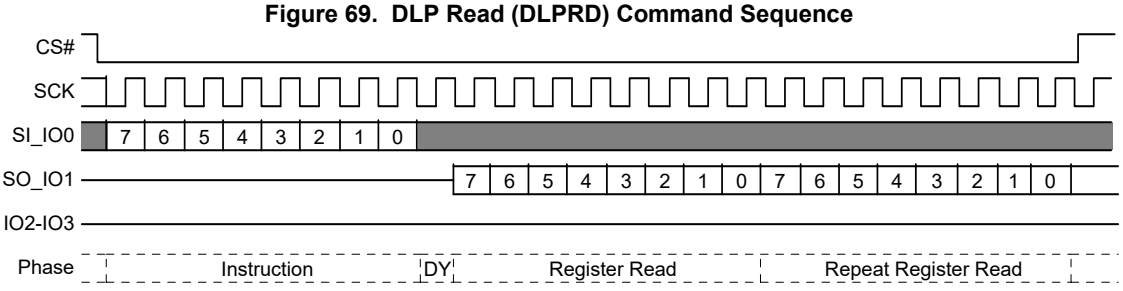
**Figure 68. Write VDLR (WVDLR) Command Sequence**




### 11.3.11 Data Learning Pattern Read (DLPRD 41h)

The instruction is shifted on SI/IO0, then the 8-bit DLP is shifted out on SO/IO1. It is possible to read the DLP continuously by providing multiples of eight clock cycles. The maximum operating clock frequency for the DLPRD command is 133MHz.

**Figure 69. DLP Read (DLPRD) Command Sequence**

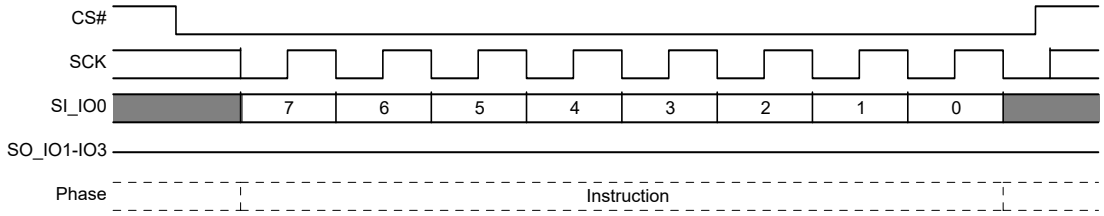


### 11.3.12 Enter 4 Byte Address Mode (4BAM B7h):

The enter 4 Byte Address Mode (4BAM) command sets the volatile Address Length bit (CR2V[7]) to 1 to change most 3 Byte address commands to require 4 Bytes of address. The Read SFDP (RSFDP) command is the only 3 Byte command that is not affected by the Address Length bit. RSFDP is required by the JEDEC JESD216 Rev B standard to always have only 3 Bytes of address.

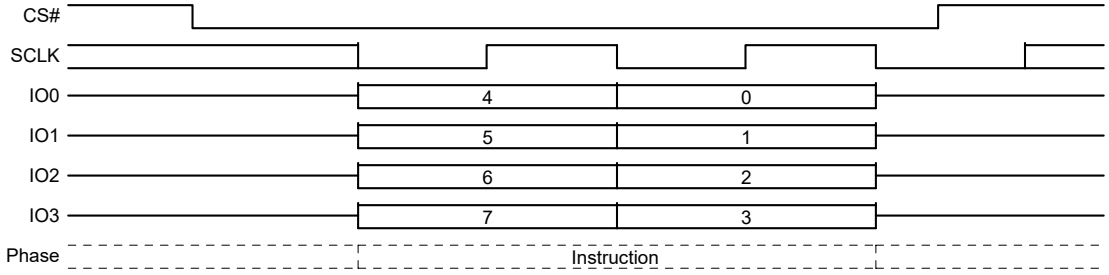
A hardware or software reset is required to exit the 4 Byte address mode.

**Figure 70. Enter 4 Byte Address Mode (4BEN B7h) Command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0-IO3.

**Figure 71. Enter 4 Byte Address QPI Mode**




### 11.3.13 Read Any Register (RDAR 65h)

The Read Any Register (RDAR) command provides a way to read all device registers - non-volatile and volatile. The instruction is followed by a 3 or 4 Byte address (depending on the address length configuration CR2V[7], followed by a number of latency (dummy) cycles set by CR2V[3:0]. Then the selected register contents are returned. If the read access is continued the same addressed register contents are returned until the command is terminated - only one register is read by each RDAR command.

Reading undefined locations provides undefined data.

The RDAR command may be used during embedded operations to read status register-1 (SR1V).

The RDAR command is not used for reading registers that act as a window into a larger array: ECCSR, PPBAR, and DYBAR. There are separate commands required to select and read the location in the array accessed.

The RDAR command will read invalid data from the PASS register locations if the ASP Password protection mode is selected by programming ASPR[2] to 0.

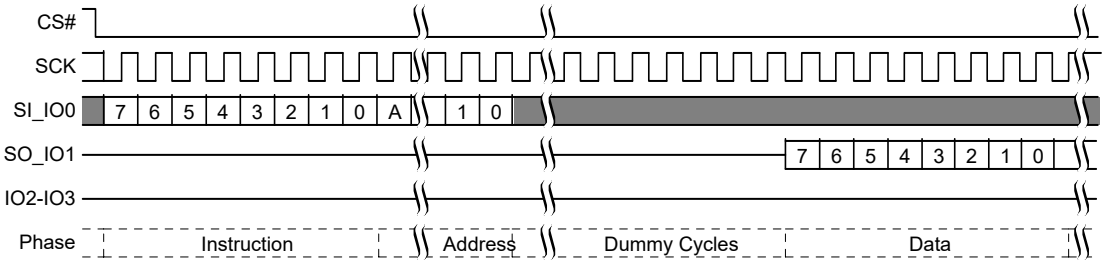
**Table 53. Register Address Map**

| Byte Address (Hex) | Register Name | Description |
|---|---|---|
| 00000000 | SR1NV | Non-volatile Status and Configuration Registers |
| 00000001 | N/A | |
| 00000002 | CR1NV | |
| 00000003 | CR2NV | |
| 00000004 | CR3NV | |
| 00000005 | CR4NV | |
| ... | N/A | |
| 00000010 | NVDLR | Non-volatile Data Learning Register |

**Table 53. Register Address Map (Continued)**

| Byte Address (Hex) | Register Name | Description |
|---|---|---|
| ... | N/A | |
| 00000020 | PASS[7:0] | Non-volatile Password Register |
| 00000021 | PASS[15:8] | |
| 00000022 | PASS[23:16] | |
| 00000023 | PASS[31:24] | |
| 00000024 | PASS[39:32] | |
| 00000025 | PASS[47:40] | |
| 00000026 | PASS[55:48] | |
| 00000027 | PASS[63:56] | |
| ... | N/A | |
| 00000030 | ASPR[7:0] | Non-volatile ASP Register |
| 00000031 | ASPR[15:8] | |
| ... | N/A | |
| 00800000 | SR1V | Volatile Status and Configuration Registers |
| 00800001 | SR2V | |
| 00800002 | CR1V | |
| 00800003 | CR2V | |
| 00800004 | CR3V | |
| 00800005 | CR4V | |
| ... | N/A | |
| 00800010 | VDLR | Volatile Data Learning Register |
| ... | N/A | |
| 00800040 | PPBL | Volatile PPB Lock Register |
| ... | N/A | |

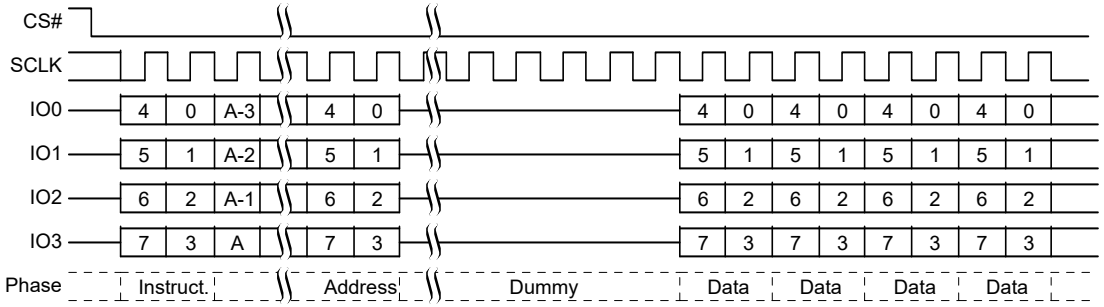
**Figure 72. Read Any Register Read Command Sequence**




**Note**
54.A = MSB of address = 23 for Address length CR2V[7] = 0, or 31 for CR2V[7]=1

This command is also supported in QPI mode. In QPI mode, the instruction and address is shifted in and returning data out on IO0-IO3.

**Figure 73. Read Any Register, QPI Mode, Command Sequence**




**Note**
55.A = MSB of address = 23 for Address length CR2V[7] = 0, or 31 for CR2V[7]=1

### 11.3.14 Write Any Register (WRAR 71h)

The Write Any Register (WRAR) command provides a way to write any device register - non-volatile or volatile. The instruction is followed by a 3 or 4 Byte address (depending on the address length configuration CR2V[7], followed by one byte of data to write in the address selected register.

Before the WRAR command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the Write Enable Latch (WEL) in the Status Register to enable any write operations. The WIP bit in SR1V may be checked to determine when the operation is completed. The P_ERR and E_ERR bits in SR1V may 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 may be modified. Some bits are read only, some are OTP.

Read only bits are never modified and the related bits in the WRAR command data byte are ignored without setting a program or erase error indication (P_ERR or E_ERR in SR1V). Hence, the value of these bits in the WRAR data byte do not matter.

OTP bits may 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.

Non-volatile bits which are changed by the WRAR data, require non-volatile register write time ($t_{W)}$ to be updated. The update process involves an erase and a program operation on the non-volatile register bits. If either the erase or program portion of the update fails the related error bit and WIP in SR1V will be set to 1.

Volatile bits which are changed by the WRAR data, require the volatile register write time ($t_{CS}$) to be updated.

Status Register-1 may be repeatedly read (polled) to monitor the Write-In-Progress (WIP) bit (SR1V[0]) and the error bits (SR1V[6,5]) to determine when the register write is completed or failed. If there is a write failure, the clear status command is used to clear the error status and enable the device to return to standby state.

However, the PPBL register can not be written by the WRAR command. Only the PPB Lock Bit Write (PLBWR) command can write the PPBL register.

The command sequence and behavior is the same as the PP or 4PP command with only a single byte of data provided. See Page Program (PP 02h or 4PP 12h) on page 102.

The address map of the registers is the same as shown for Read Any Register (RDAR 65h) on page 89.

### 11.3.15 Set Burst Length (SBL C0h)

The Set Burst Length (SBL) command is used to configure the Burst Wrap feature. Burst Wrap is used in conjunction with Quad I/O Read, DDR Quad I/O Read and Quad Output Read, in legacy SPI or QPI mode, to access a fixed length and alignment of data. Certain applications can benefit from this feature by improving the overall system code execution performance. The Burst Wrap feature allows applications that use cache, to start filling a cache line with instruction or data from a critical address first, then fill the remainder of the cache line afterwards within a fixed length (8/16/32/64-bytes) of data, without issuing multiple read commands.

The Set Burst Length (SBL) command writes the CR4V register bits 4, 1, and 0 to enable or disable the wrapped read feature and set the wrap boundary. Other bits of the CR4V register are not affected by the SBL command. When enabled the wrapped read feature changes the related read commands from sequentially reading until the command ends, to reading sequentially wrapped within a group of bytes.

When CR4V[4] = 1, the wrap mode is not enabled and unlimited length sequential read is performed.

When CR4V[4] = 0, the wrap mode is enabled and a fixed length and aligned group of 8, 16, 32, or 64 bytes is read starting at the byte address provided by the read command and wrapping around at the group alignment boundary.

The group of bytes is of length and aligned on an 8, 16, 32, or 64 byte boundary. CR4V[1:0] selects the boundary. See Configuration Register 4 Volatile (CR4V) on page 60.

The starting address of the read command selects the group of bytes and the first data returned is the addressed byte. Bytes are then read sequentially until the end of the group boundary is reached. If the read continues the address wraps to the beginning of the group and continues to read sequentially. This wrapped read sequence continues until the command is ended by CS# returning high.

**Table 54. Example Burst Wrap Sequences**

| SBL Data Value (Hex) | Wrap Boundary (Bytes) | Start Address (Hex) | Address Sequence (Hex) |
|---|---|---|---|
| 1X | Sequential | XXXXXX03 | 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, ... |
| 00 | 8 | XXXXXX00 | 00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, ... |
| 00 | 8 | XXXXXX07 | 07, 00, 01, 02, 03, 04, 05, 06, 07, 00, 01, ... |
| 01 | 16 | XXXXXX02 | 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 00, 01, 02, 03, ... |
| 01 | 16 | XXXXXX0C | 0C, 0D, 0E, 0F, 00, 01, 02, 03, 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, ... |
| 02 | 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, ... |
| 02 | 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, ... |
| 03 | 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, ... |
| 03 | 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, , ... |

The power-on reset, hardware reset, or software reset default burst length can be changed by programming CR4NV with the desired value using the WRAR command.

**Figure 74. Set Burst Length Command Sequence Quad I/O Mode**

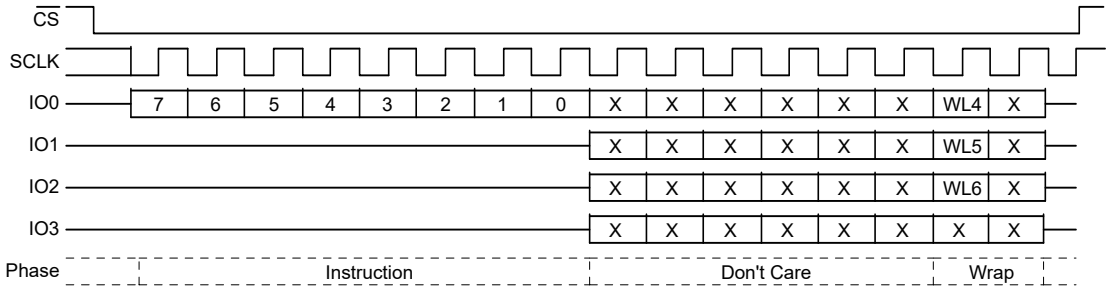
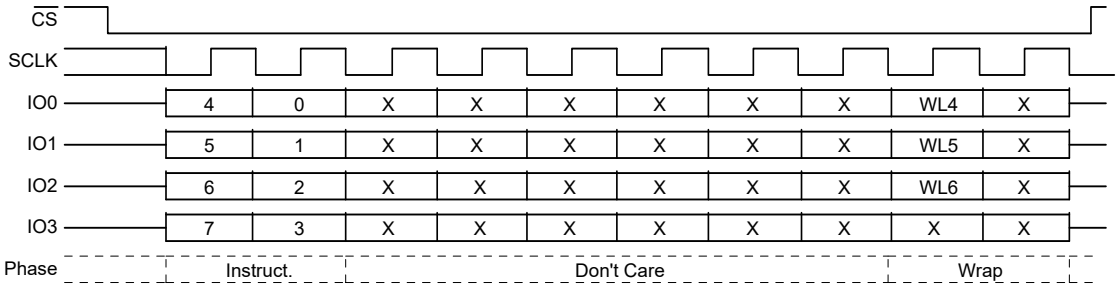



**Figure 75. Set Burst Length Command Sequence QPI Mode**



## 11.4 Read Memory Array Commands

Read commands for the main Flash array provide many options for prior generation SPI compatibility or enhanced performance SPI:

- Some commands transfer address or data on each rising edge of SCK. These are called Single Data Rate commands (SDR).
- Some SDR commands transfer address one bit per rising edge of SCK and return data 1bit of data per rising edge of SCK. These are called Single width commands.
- Some SDR commands transfer address one bit per rising edge of SCK and return data 2 or 4 bits per rising edge of SCK. These are called Dual Output for 2 bit, Quad Output for 4 bit.
- Some SDR commands transfer both address and data 2 or 4 bits per rising edge of SCK. These are called Dual I/O for 2 bit, Quad I/O, and QPI for 4 bit.
- Some SDR commands in QPI mode also transfers instructions, address and data 4 bits per rising edge of SCK.
- Some commands transfer address and data on both the rising edge and falling edge of SCK. These are called Double Data Rate (DDR) commands.
- There are DDR commands for 4 bits of address or data per SCK edge. These are called Quad I/O DDR and QPI DDR for 4 bit per edge transfer.

All of these commands, except QPI Read, begin with an instruction code that is transferred one bit per SCK rising edge. QPI Read transfers the instruction 4 bits per SCK rising edge.The instruction is followed by either a 3 or 4 byte address transferred at SDR or DDR. Commands transferring address or data 2 or 4 bits per clock edge are called Multiple I/O (MIO) commands. These devices may be configured to take a 4 byte address from the host system with the traditional 3 byte address commands. The 4 byte address mode for traditional commands is activated by setting the Address Length bit in configuration register 2 to “0”. The higher order address bits above A23 in the 4 byte address commands, or commands using 4 Byte Address mode are not relevant and are ignored.

The Quad I/O and QPI commands provide a performance improvement option controlled by mode bits that are sent following the address bits. The mode bits indicate whether the command following the end of the current read will be another read of the same type, without an instruction at the beginning of the read. These mode bits give the option to eliminate the instruction cycles when doing a series of Quad read accesses.

Some commands require delay cycles following the address or mode bits to allow time to access the memory array - read latency. The delay or read latency cycles are traditionally called dummy cycles. The dummy cycles are ignored by the memory thus any data provided by the host during these cycles is “don’t care” and the host may also leave the SI/IO1 signal at high impedance during the dummy cycles. When MIO commands are used the host must stop driving the IO signals (outputs are high impedance) before the end of last dummy cycle. When DDR commands are used the host must not drive the I/O signals during any dummy cycle. The number of dummy cycles varies with the SCK frequency or performance option selected via the Configuration Register 2 (CR2V[3:0]) Latency Code. Dummy cycles are measured from SCK falling edge to next SCK falling edge. SPI outputs are traditionally driven to a new value on the falling edge of each SCK. Zero dummy cycles means the returning data is driven by the memory on the same falling edge of SCK that the host stops driving address or mode bits.

The DDR commands may optionally have an 8 edge Data Learning Pattern (DLP) driven by the memory, on all data outputs, in the dummy cycles immediately before the start of data. The DLP can help the host memory controller determine the phase shift from SCK to data edges so that the memory controller can capture data at the center of the data eye.

When using SDR I/O commands at higher SCK frequencies (>50 MHz), an LC that provides 1 or more dummy cycles should be selected to allow additional time for the host to stop driving before the memory starts driving data, to minimize I/O driver conflict. When using DDR I/O commands with the DLP enabled, an LC that provides 5 or more dummy cycles should be selected to allow 1 cycle of additional time for the host to stop driving before the memory starts driving the 4 cycle DLP.

Each read command ends when CS# is returned High at any point during data return. CS# must not be returned High during the mode or dummy cycles before data returns as this may cause mode bits to be captured incorrectly; making it indeterminate as to whether the device remains in continuous read mode.

### 11.4.1 Read (Read 03h or 4READ 13h)

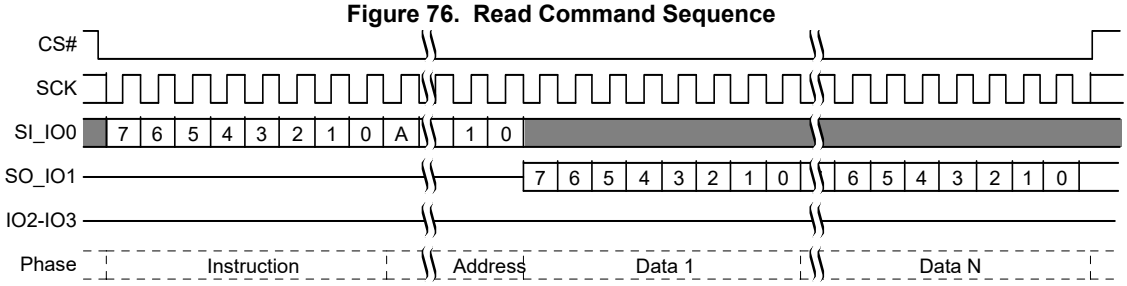
The instruction

- 03h (CR2V[7] = 0) is followed by a 3-byte address (A23–A0) or
- 03h (CR2V[7] = 1) is followed by a 4-byte address (A31–A0) or
- 13h is followed by a 4-byte address (A31–A0)

Then the memory contents, at the address given, are shifted out on SO/IO1 . The maximum operating clock frequency for the READ command is 50 MHz.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

**Figure 76. Read Command Sequence**




**Note**
56.A = MSB of address = 23 for CR2V[7]=0, or 31 for CR2V[7]=1 or command 13h.

### 11.4.2 Fast Read (FAST_READ 0Bh or 4FAST_READ 0Ch)

The instruction

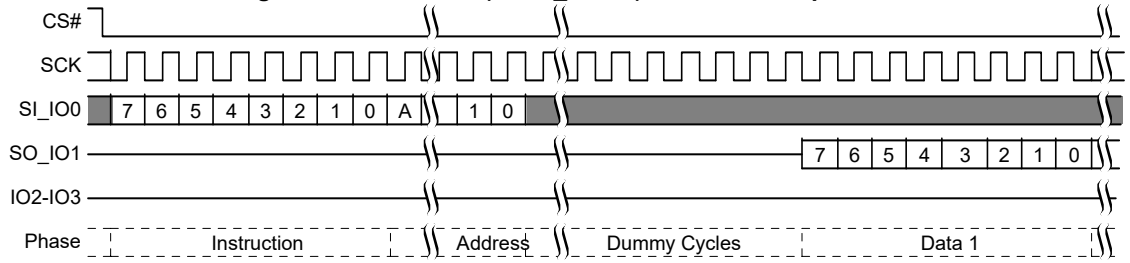
- 0Bh (CR2V[7] = 0) is followed by a 3-byte address (A23–A0) or
- 0Bh (CR2V[7] = 1) is followed by a 4-byte address (A31–A0) or
- 0Ch is followed by a 4-byte address (A31-A0)

The address is followed by dummy cycles depending on the latency code set in the Configuration Register CR2V[3:0]. The dummy cycles allow the device internal circuits additional time for accessing the initial address location. During the dummy cycles the data value on SO/IO1 is “don’t care” and may be high impedance. Then the memory contents, at the address given, are shifted out on SO/IO1.

The maximum operating clock frequency for FAST READ command is 133 MHz.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

**Figure 77. Fast Read (FAST_READ) Command Sequence**




**Note**
57.A = MSB of address = 23 for CR2V[7]=0, or 31 for CR2V[7]=1 or command 0Ch.

### 11.4.3 Dual Output Read (DOR 3Bh or 4DOR 3Ch)

The instruction

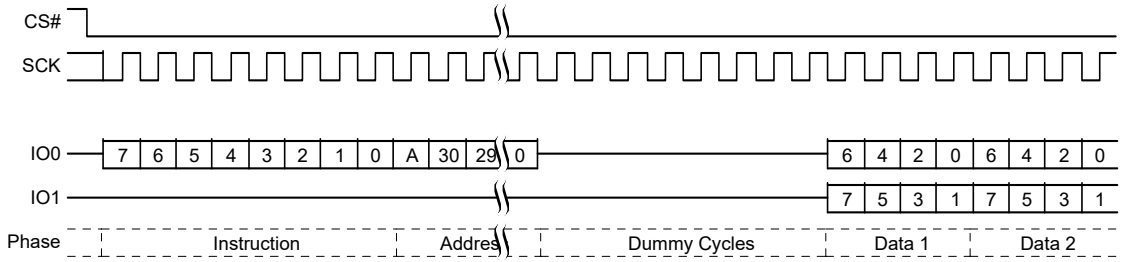
- 3Bh (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- 3Bh (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- 3Ch is followed by a 4-byte address (A31–A0)

The address is followed by dummy cycles depending on the latency code set in the Configuration Register CR3V[3:0]. The dummy cycles allow the device internal circuits additional time for accessing the initial address location. During the dummy cycles the data value on IO0 (SI) and IO1 (S0) is “don’t care” and may be high impedance.

Then the memory contents, at the address given, is shifted out two bits at a time through IO0 (SI) and IO1 (SO). Two bits are shifted out at the SCK frequency by the falling edge of the SCK signal.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

**Figure 78. Dual Output Read Command Sequence**




**Note**
58.A = MSB of address = 23 for CR2V[7]=0 or 31 for CR2V[7]=1 or command 3Ch.

### 11.4.4 Quad Output Read (QOR 6Bh or 4QOR 6Ch)

The instruction

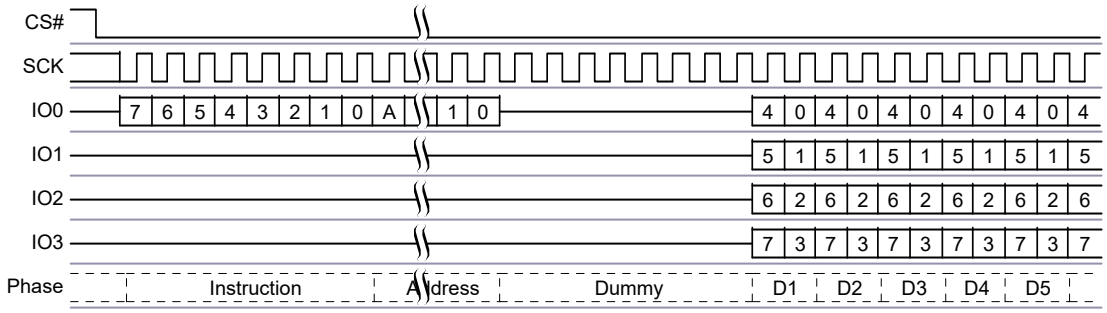
- 6Bh (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- 6Bh (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- 6Ch is followed by a 4-byte address (A31–A0)

The address is followed by dummy cycles depending on the latency code set in the Configuration Register CR3V[3:0]. The dummy cycles allow the device internal circuits additional time for accessing the initial address location. During the dummy cycles the data value on IO0–IO3 is “don’t care” and may be high impedance.

Then the memory contents, at the address given, is shifted out four bits at a time through IO0 - IO3. Each nibble (4 bits) is shifted out at the SCK frequency by the falling edge of the SCK signal.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

**Figure 79. Quad Output Read Command Sequence**




**Note**
59.A = MSB of address = 23 for CR2V[7] = 0, or 31 for CR2V[7] = 1 or command 6Ch.

### 11.4.5 Dual I/O Read (DIOR BBh or 4DIOR BCh)

The instruction

- BBh (CR2V[7] = 0) is followed by a 3-byte address (A23–A0) or
- BBh (CR2V[7] = 1) is followed by a 4-byte address (A31–A0) or
- BCh is followed by a 4-byte address (A31–A0)

The Dual I/O Read commands improve throughput with two I/O signals IO0 and IO1. This command takes input of the address and returns read data two bits per SCK rising edge. In some applications, the reduced address input and data output time might allow for code execution in place (XIP) i.e. directly from the memory device.

The maximum operating clock frequency for Dual I/O Read is 133 MHz.

The Dual I/O Read command has continuous read mode bits that follow the address so, a series of Dual I/O Read commands may eliminate the 8 bit instruction after the first Dual I/O Read command sends a mode bit pattern of Axh that indicates the following command will also be a Dual I/O Read command. The first Dual I/O Read command in a series starts with the 8 bit instruction, followed by address, followed by four cycles of mode bits, followed by an optional latency period. If the mode bit pattern is Axh the next command is assumed to be an additional Dual I/O Read command that does not provide instruction bits. That command starts with address, followed by mode bits, followed by optional latency.

Variable latency may be added after the mode bits are shifted into IO0 and IO1 before data begins shifting out of IO0 and IO1. This latency period (dummy cycles) allows the device internal circuitry enough time to access data at the initial address. During the dummy cycles, the data value on IO0 and IO1 are “don’t care” and may be high impedance. The number of dummy cycles is determined by the frequency of SCK. The latency is configured in CR2V[3:0].

The continuous read feature removes the need for the instruction bits in a sequence of read accesses and greatly improves code execution (XIP) performance. The upper nibble (bits 7-4) of the Mode bits control the length of the next Dual I/O Read command through the inclusion or exclusion of the first byte instruction code. The lower nibble (bits 3-0) of the Mode bits are “don’t care” (“x”) and may be high impedance. If the Mode bits equal Axh, then the device remains in Dual I/O Continuous Read Mode and the next address can be entered (after CS# is raised high and then asserted low) without the BBh or BCh instruction, as shown in Figure 80 on page 97; thus, eliminating eight cycles of the command sequence. The following sequences will release the device from Dual I/O Continuous Read mode; after which, the device can accept standard SPI commands:

1. During the Dual I/O continuous read command sequence, if the Mode bits are any value other than Axh, then the next time CS# is raised high the device will be released from Dual I/O continuous read mode.
2. Send the Mode Reset command.

Note that the four mode bit cycles are part of the device’s internal circuitry latency time to access the initial address after the last address cycle that is clocked into IO0 and IO1.

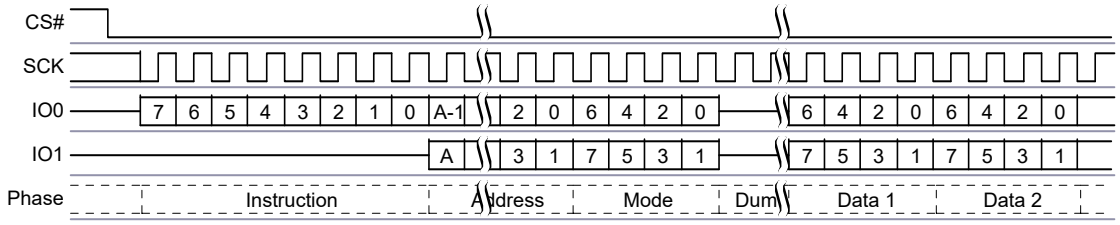
It is important that the I/O signals be set to high-impedance at or before the falling edge of the first data out clock. At higher clock speeds the time available to turn off the host outputs before the memory device begins to drive (bus turn around) is diminished. It is allowed and may be helpful in preventing I/O signal contention, for the host system to turn off the I/O signal outputs (make them high impedance) during the last two “don’t care” mode cycles or during any dummy cycles.

Following the latency period the memory content, at the address given, is shifted out two bits at a time through IO0 and IO1. Two bits are shifted out at the SCK frequency at the falling edge of SCK signal.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

CS# should not be driven high during mode or dummy bits as this may make the mode bits indeterminate.

**Figure 80. Dual I/O Read Command Sequence**

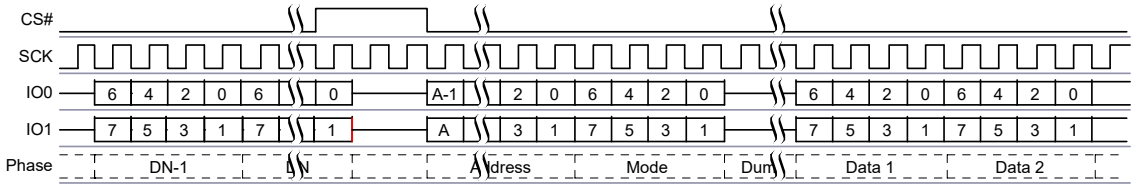



**Notes**

60.Least significant 4 bits of Mode are don't care and it is optional for the host to drive these bits. The host may turn off drive during these cycles to increase bus turn around time between Mode bits from host and returning data from the memory.

61.A = MSB of address = 23 for CR2V[7] = 0, or 31 for CR2V[7] = 1 or command BBh.

**Figure 81. Dual I/O Continuous Read Command Sequence**




**Notes**

62.Least significant 4 bits of Mode are don't care and it is optional for the host to drive these bits. The host may turn off drive during these cycles to increase bus turn around time between Mode bits from host and returning data from the memory.

63.A = MSB of address = 23 for CR2V[7] = 0, or 31 for CR2V[7] = 1 or command BBh.

### 11.4.6 Quad I/O Read (QIOR EBh or 4QIOR ECh)

The instruction

- EBh (CR2V[7] = 0) is followed by a 3-byte address (A23–A0) or
- EBh (CR2V[7] = 1) is followed by a 4-byte address (A31–A0) or
- ECh is followed by a 4-byte address (A31–A0)

The Quad I/O Read command improves throughput with four I/O signals IO0–IO3. It allows input of the address bits four bits per serial SCK clock. In some applications, the reduced instruction overhead might allow for code execution (XIP) directly from FS-S family devices. The QUAD bit of the Configuration Register must be set (CR1V[1] = 1) to enable the Quad capability of FS-S Family devices.

The maximum operating clock frequency for Quad I/O Read is 133 MHz.

For the Quad I/O Read command, there is a latency required after the mode bits (described below) before data begins shifting out of IO0–IO3. This latency period (i.e., dummy cycles) allows the device's internal circuitry enough time to access data at the initial address. During latency cycles, the data value on IO0–IO3 are "don't care" and may be high impedance. The number of dummy cycles is determined by the frequency of SCK. The latency is configured in CR2V[3:0].

Following the latency period, the memory contents at the address given, is shifted out four bits at a time through IO0-IO3. Each nibble (4 bits) is shifted out at the SCK frequency by the falling edge of the SCK signal.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

Address jumps can be done without the need for additional Quad I/O Read instructions. This is controlled through the setting of the Mode bits (after the address sequence, as shown in Figure 82. This added feature removes the need for the instruction sequence and greatly improves code execution (XIP). The upper nibble (bits 7-4) of the Mode bits control the length of the next Quad I/O instruction through the inclusion or exclusion of the first byte instruction code. The lower nibble (bits 3–0) of the Mode bits are “don’t care” (“x”). If the Mode bits equal Axh, then the device remains in Quad I/O High Performance Read Mode and the next address can be entered (after CS# is raised high and then asserted low) without requiring the EBh or ECh instruction, as shown in Figure 84 on page 99; thus, eliminating eight cycles for the command sequence. The following sequences will release the device from Quad I/O High Performance Read mode; after which, the device can accept standard SPI commands:

1. During the Quad I/O Read Command Sequence, if the Mode bits are any value other than Axh, then the next time CS# is raised high the device will be released from Quad I/O High Performance Read mode.
2. Send the Mode Reset command.

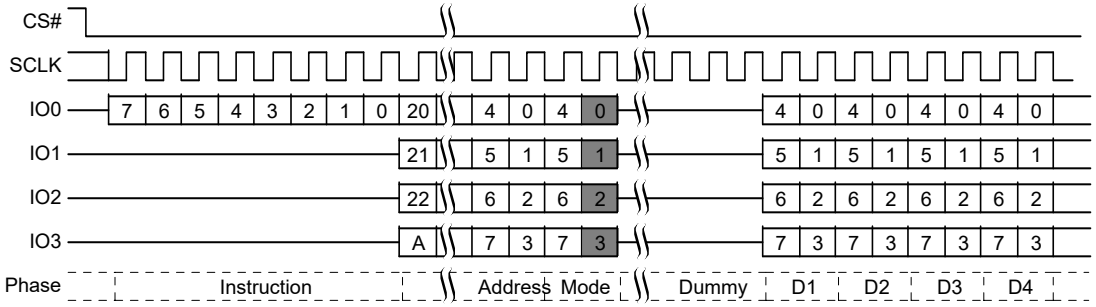
Note that the two mode bit clock cycles and additional wait states (i.e., dummy cycles) allow the device’s internal circuitry latency time to access the initial address after the last address cycle that is clocked into IIO0-IO3.

It is important that the IO0-IO3signals be set to high-impedance at or before the falling edge of the first data out clock. At higher clock speeds the time available to turn off the host outputs before the memory device begins to drive (bus turn around) is diminished. It is allowed and may be helpful in preventing IO0-IO3 signal contention, for the host system to turn off the IO0-IO3 signal outputs (make them high impedance) during the last “don’t care” mode cycle or during any dummy cycles.

CS# should not be driven high during mode or dummy bits as this may make the mode bits indeterminate.

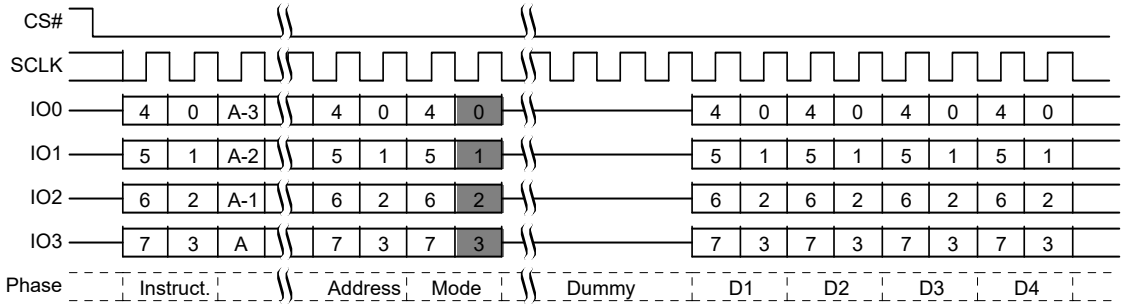
In QPI mode (CR2V[6] = 1) the Quad I/O instructions are sent 4 bits per SCK rising edge. The remainder of the command protocol is identical to the Quad I/O commands.

**Figure 82. Quad I/O Read Initial Access Command Sequence**



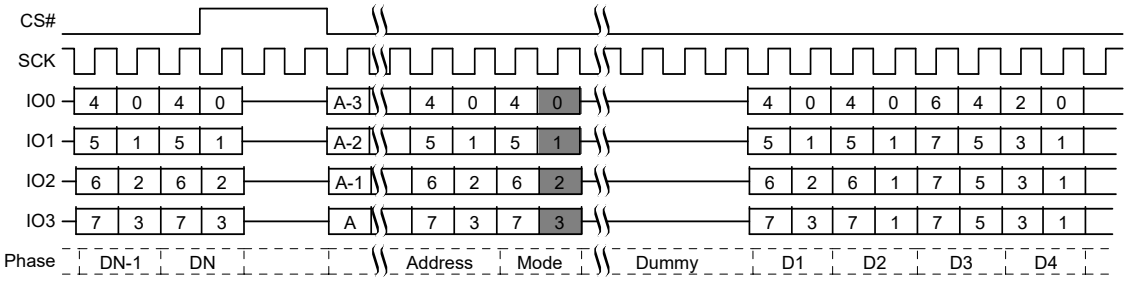

**Note**
64.A = MSB of address = 23 for CR2V[7] = 0, or 31 for CR2V[7] = 1 or command ECh.

**Figure 83. Quad I/O Read Initial Access Command Sequence QPI mode**




**Note**
65.A = MSB of address = 23 for CR2V[7] = 0, or 31 for CR2V[7] = 1 or command ECh.

**Figure 84. Continuous Quad I/O Read Command Sequence**




**Notes**

66.A = MSB of address = 23 for CR2V[7] = 0, or 31 for CR2V[7] = 1 or command ECh.

67.The same sequence is used in QPI mode

### 11.4.7 DDR Quad I/O Read (EDh, EEh)

The DDR Quad I/O Read command improves throughput with four I/O signals IO0-IO3. It is similar to the Quad I/O Read command but allows input of the address four bits on every edge of the clock. In some applications, the reduced instruction overhead might allow for code execution (XIP) directly from FS-S Family devices. The QUAD bit of the Configuration Register must be set (CR1V[1] = 1) to enable the Quad capability.

The instruction

- EDh (CR2V[7] = 0) is followed by a 3-byte address (A23–A0) or
- EDh (CR2V[7] = 1) is followed by a 4-byte address (A31–A0) or
- EEh is followed by a 4-byte address (A31–A0)

The address is followed by mode bits. Then the memory contents, at the address given, is shifted out, in a DDR fashion, with four bits at a time on each clock edge through IO0–IO3.

The maximum operating clock frequency for DDR Quad I/O Read command is 100 MHz.

For DDR Quad I/O Read, there is a latency required after the last address and mode bits are shifted into the IO0-IO3 signals before data begins shifting out of IO0–IO3. This latency period (dummy cycles) allows the device's internal circuitry enough time to access the initial address. During these latency cycles, the data value on IO0-IO3are "don't care" and may be high impedance. When the Data Learning Pattern (DLP) is enabled the host system must not drive the IO signals during the dummy cycles. The IO signals must be left high impedance by the host so that the memory device can drive the DLP during the dummy cycles.

The number of dummy cycles is determined by the frequency of SCK. The latency is configured in CR2V[3:0].

Mode bits allow a series of Quad I/O DDR commands to eliminate the 8 bit instruction after the first command sends a complementary mode bit pattern, as shown in Figure 85 and Figure 87. This feature removes the need for the eight bit SDR instruction sequence and dramatically reduces initial access times (improves XIP performance). The Mode bits control the length of the next DDR Quad I/O Read operation through the inclusion or exclusion of the first byte instruction code. If the upper nibble (IO[7:4]) and lower nibble (IO[3:0]) of the Mode bits are complementary (i.e. 5h and Ah) the device transitions to Continuous DDR Quad I/O Read Mode and the next address can be entered (after CS# is raised high and then asserted low) without requiring the EDh or EEh instruction, eliminating eight cycles from the command sequence. The following sequences will release the device from Continuous DDR Quad I/O Read mode; after which, the device can accept standard SPI commands:

1. During the DDR Quad I/O Read Command Sequence, if the Mode bits are not complementary the next time CS# is raised high and then asserted low the device will be released from DDR Quad I/O Read mode.
2. Send the Mode Reset command.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

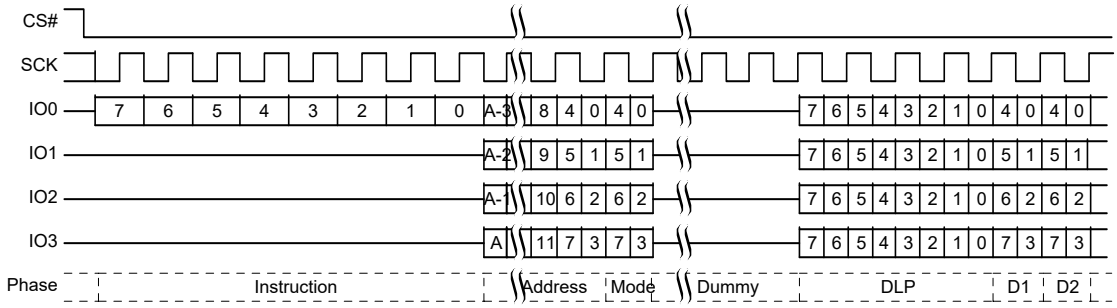
CS# should not be driven high during mode or dummy bits as this may make the mode bits indeterminate. Note that the memory devices may drive the IOs with a preamble prior to the first data value. The preamble is a Data Learning Pattern (DLP) that is used by the host controller to optimize data capture at higher frequencies. The preamble drives the IO bus for the four clock cycles immediately before data is output. The host must be sure to stop driving the IO bus prior to the time that the memory starts outputting the preamble.

The preamble is intended to give the host controller an indication about the round trip time from when the host drives a clock edge to when the corresponding data value returns from the memory device. The host controller will skew the data capture point during the preamble period to optimize timing margins and then use the same skew time to capture the data during the rest of the read operation. The optimized capture point will be determined during the preamble period of every read operation. This optimization strategy is intended to compensate for both the PVT (process, voltage, temperature) of both the memory device and the host controller as well as any system level delays caused by flight time on the PCB.

Although the data learning pattern (DLP) is programmable, the following example shows example of the DLP of 34h. The DLP 34h (or 00110100) will be driven on each of the active outputs (i.e. all four IOs). This pattern was chosen to cover both “DC” and “AC” data transition scenarios. The two DC transition scenarios include data low for a long period of time (two half clocks) followed by a high going transition (001) and the complementary low going transition (110). The two AC transition scenarios include data low for a short period of time (one half clock) followed by a high going transition (101) and the complementary low going transition (010). The DC transitions will typically occur with a starting point closer to the supply rail than the AC transitions that may not have fully settled to their steady state (DC) levels. In many cases the DC transitions will bound the beginning of the data valid period and the AC transitions will bound the ending of the data valid period. These transitions will allow the host controller to identify the beginning and ending of the valid data eye. Once the data eye has been characterized the optimal data capture point can be chosen. See SPI DDR Data Learning Registers on page 63 for more details.

In QPI mode (CR2V[6]=1) the DDR Quad I/O instructions are sent 4 bits per SCK rising edge. The remainder of the command protocol is identical to the DDR Quad I/O commands.

**Figure 85. DDR Quad I/O Read Initial Access**

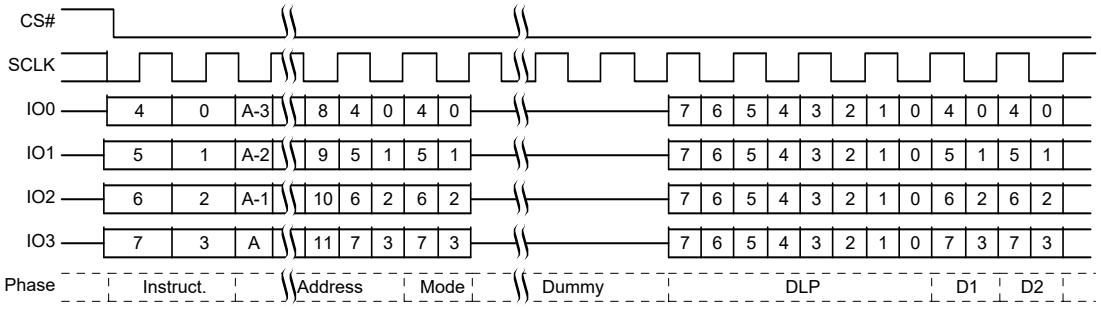



**Notes**

68.A = MSB of address = 23 for CR2V[7] = 0, or 31 for CR2V[7] = 1 or command EEh.

69.Example DLP of 34h (or 00110100).

**Figure 86. DDR Quad I/O Read Initial Access QPI Mode**

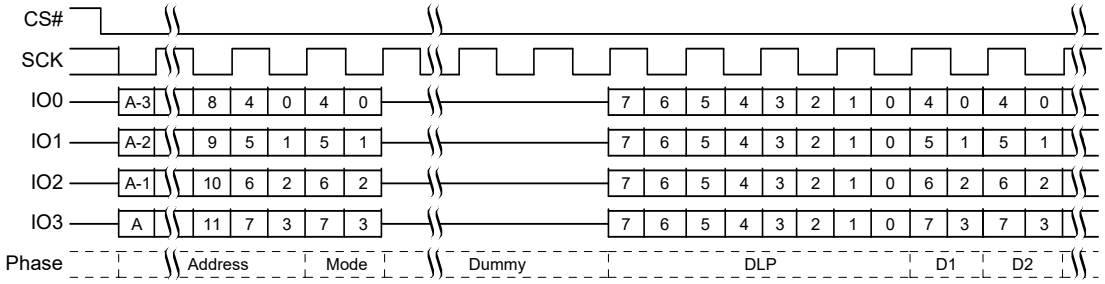



**Notes**

70.A = MSB of address = 23 for CR2V[7] = 0, or 31 for CR2V[7] = 1 or command EEh.

71.Example DLP of 34h (or 00110100).

**Figure 87. Continuous DDR Quad I/O Read Subsequent Access**




**Notes**

72.A = MSB of address = 23 for CR2V[7] = 0, or 31 for CR2V[7] = 1 or command EEh.
73.The same sequence is used in QPI mode.
74.Example DLP of 34h (or 00110100).

## 11.5 Program Flash Array Commands

### 11.5.1 Program Granularity

#### 11.5.1.1 Automatic ECC

Each 16 byte aligned and 16 byte length Programming Block has an automatic Error Correction Code (ECC) value. The data block plus ECC form an ECC unit. In combination with Error Detection and Correction (EDC) logic the ECC is used to detect and correct any single bit error found during a read access. When data is first programmed within an ECC unit the ECC value is set for the entire ECC unit. If the same ECC unit is programmed more than once the ECC value is changed to disable the EDC function. A sector erase is needed to again enable Automatic ECC on that Programming Block. The 16 byte Program Block is the smallest program granularity on which Automatic ECC is enabled.

These are automatic operations transparent to the user. The transparency of the Automatic ECC feature enhances data accuracy for typical programming operations which write data once to each ECC unit but, facilitates software compatibility to previous generations of FL family of products by allowing for single byte programming and bit walking in which the same ECC unit is programmed more than once. When an ECC unit has Automatic ECC disabled, EDC is not done on data read from the ECC unit location.

An ECC status register is provided for determining if ECC is enabled on an ECC unit and whether any errors have been detected and corrected in the ECC unit data or the ECC (See ECC Status Register (ECCSR) on page 61). The ECC Status Register Read (ECCRD) command is used to read the ECC status on any ECC unit (See ECC Status Register Read (ECCRD 19h or 4EECRD 18h) on page 87).

Error Detection and Correction (EDC) is applied to all parts of the Flash address spaces other than registers. An Error Correction Code (ECC) is calculated for each group of bytes protected and the ECC is stored in a hidden area related to the group of bytes. The group of protected bytes and the related ECC are together called an ECC unit.

- ECC is calculated for each 16 byte aligned and length ECC unit
- Single Bit EDC is supported with 8 ECC bits per ECC unit, plus 1 bit for an ECC disable Flag
- Sector erase resets all ECC bits and ECC disable flags in a sector to the default state (enabled)
- ECC is programmed as part of the standard Program commands operation
- ECC is disabled automatically if multiple programming operations are done on the same ECC unit.
- Single byte programming or bit walking is allowed but disables ECC on the second program to the same 16 byte ECC unit.
- The ECC disable flag is programmed when ECC is disabled
- To re-enable ECC for an ECC unit that has been disabled, the Sector that includes the ECC unit must be erased
- To ensure the best data integrity provided by EDC, each ECC unit should be programmed only once so that ECC is stored for that unit and not disabled.
- The calculation, programming, and disabling of ECC is done automatically as part of a programming operation. The detection and correction, if needed, is done automatically as part of read operations. The host system sees only corrected data from a read operation.
- ECC protects the OTP region - however a second program operation on the same ECC unit will disable ECC permanently on that ECC unit (OTP is one time programmable, hence an erase operation to re-enable the ECC enable/indicator bit is prohibited)

#### 11.5.1.2 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 command. 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 bit CR3V[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 insures that Automatic ECC is not disabled. For the very best performance, programming should be done in full pages of 512 bytes aligned on 512 byte boundaries with each Page being programmed only once.

#### 11.5.1.3 Single Byte Programming

Single Byte Programming allows full backward compatibility to the legacy standard SPI Page Programming (PP) command by allowing a single byte to be programmed anywhere in the memory array. While single byte programming is supported, this will disable Automatic ECC on the 16 byte ECC unit, if a another byte is programmed on the same ECC unit.

### 11.5.2 Page Program (PP 02h or 4PP 12h)

The Page Program (PP) command allows bytes to be programmed in the memory (changing bits from 1 to 0). Before the Page Program (PP) commands can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device. After the Write Enable (WREN) command has been decoded successfully, the device sets the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The instruction

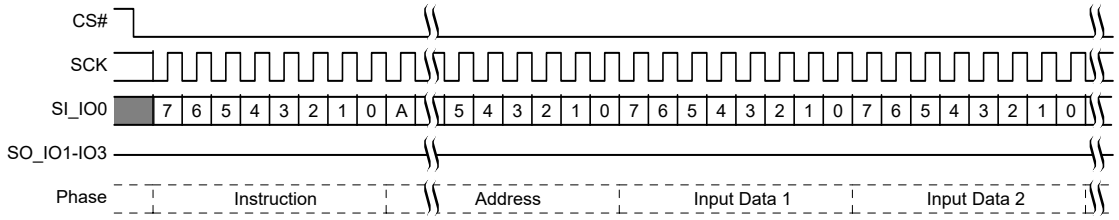
- 02h (CR2V[7] = 0) is followed by a 3-byte address (A23–A0) or
- 02h (CR2V[7] = 1) is followed by a 4-byte address (A31–A0) or
- 12h is followed by a 4-byte address (A31–A0)

and at least one data byte on SI/IO0. Depending on CR3V[4], the page size can either be 256 or 512 bytes. Up to a page can be provided on SI/IO0 after the 3-byte address with instruction 02h or 4-byte address with instruction 12h has been provided. If more data is sent to the device than 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. The last page worth of data is programmed in the page. This is a result of the device being equipped with a page program buffer that is only page size in length. If less than a page of data is sent to the device, these data bytes will be programmed in sequence, starting at the provided address within the page, without having any affect on the other bytes of the same page.

Using the Page Program (PP) command to load an entire page, within the page boundary, will save overall programming time versus loading less than a page into the program buffer.

The programming process is managed by the Flash memory device internal control logic. After a programming command is issued, the programming operation status can be checked using the Read Status Register-1 command. The WIP bit (SR1V[0]) will indicate when the programming operation is completed. The P_ERR bit (SR1V[6]) will indicate if an error occurs in the programming operation that prevents successful completion of programming. This includes attempted programming of a protected area.

**Figure 88. Page Program (PP 02h or 4PP 12h) Command Sequence**

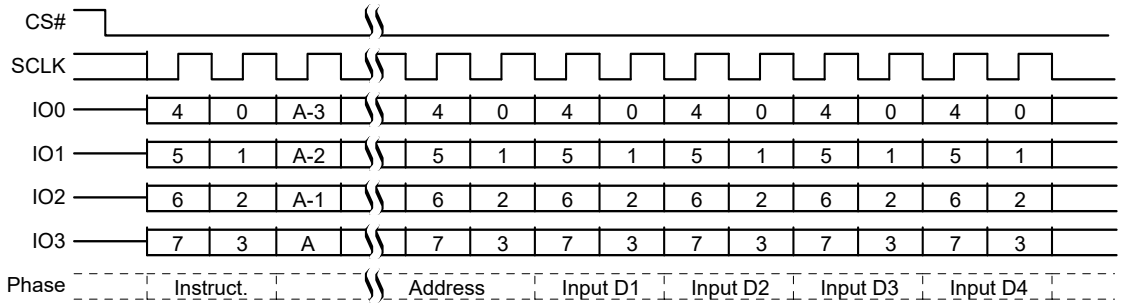



**Note**

75.A = MSB of address = A23 for PP 02h with CR2V[7] = 0, or A31 for PP 02h with CR2V[7] = 1, or for 4PP 12h.

This command is also supported in QPI mode. In QPI mode the instruction, address and data is shifted in on IO0–IO3.

**Figure 89. Page Program (PP 02h or 4PP 12h) QPI Mode Command Sequence**




**Note:**
76.A = MSB of address = A23 for PP 02h with CR2V[7] = 0, or A31 for PP 02h with CR2V[7] = 1, or for 4PP 12h

### 11.5.3 Quad Page Program (QPP 32h or 4QPP 34h)

The Quad-input Page Program (QPP) command allows bytes to be programmed in the memory (changing bits from 1 to 0). The Quad-input Page Program (QPP) command allows up to a page of data to be loaded into the Page Buffer using four signals: IO0-IO3. QPP can improve performance for PROM Programmer and applications that have slower clock speeds (< 12 MHz) by loading 4 bits of data per clock cycle. Systems with faster clock speeds do not realize as much benefit for the QPP command since the inherent page program time becomes greater than the time it takes to clock-in the data. The maximum frequency for the QPP command is 133MHz.

To use Quad Page Program the Quad Enable Bit in the Configuration Register must be set (QUAD = 1). A Write Enable command must be executed before the device will accept the QPP command (Status Register-1, WEL = 1).

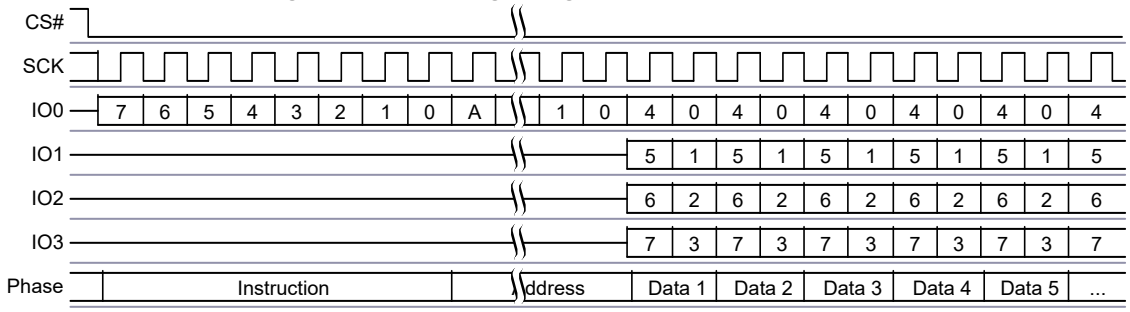
The instruction

- 32h (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- 32h (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- 34h is followed by a 4-byte address (A31–A0)

and at least one data byte, into the IO signals.

All other functions of QPP are identical to Page Program. The QPP command sequence is shown in the figure below.

**Figure 90. Quad Page Program Command Sequence**




**Note**
77.A = MSB of address = A23 for QPP 32h with CR2V[7]=0, or A31 for QPP 32h with CR2V[7]=1, or for 4QPP 34h.

## 11.6 Erase Flash Array Commands

### 11.6.1 Parameter Sector Erase (P4E 20h or 4P4E 21h)

The main Flash array address map may be configured to overlay parameter sectors over the lowest address portion of the lowest address uniform sector (bottom parameter sectors) or over the highest address portion of the highest address uniform sector (top parameter sectors). The main Flash array address map may also be configured to have only uniform size sectors. The parameter sector configuration is controlled by the configuration bit CR3V[3]. The P4E and 4P4E commands are ignored when the device is configured for uniform sectors only (CR3V[3]=1).

The Parameter Sector Erase commands set all the bits of a parameter sector to 1 (all bytes are FFh). Before the P4E or 4P4E command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The instruction

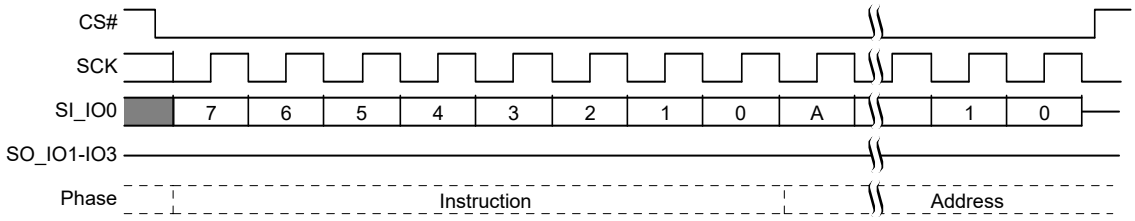
- 20h [CR2V[7] = 0] is followed by a 3-byte address (A23–A0), or
- 20h [CR2V[7] = 1] is followed by a 4-byte address (A31–A0), or
- 21h is followed by a 4-byte address (A31–A0)

CS# must be driven into the logic high state after the twenty-fourth or thirty-second bit of the address has been latched in on SI/IO0. This will initiate the beginning of internal erase cycle, which involves the pre-programming and erase of the chosen sector of the flash memory array. If CS# is not driven high after the last bit of address, the sector erase operation will not be executed.

As soon as CS# is driven high, the internal erase cycle will be initiated. With the internal erase cycle in progress, the user can read the value of the Write-In Progress (WIP) bit to determine when the operation has been completed. The WIP bit will indicate a “1”. when the erase cycle is in progress and a “0” when the erase cycle has been completed.

A P4E or 4P4E command applied to a sector that has been write protected through the Block Protection bits or ASP, will not be executed and will set the E_ERR status. A P4E command applied to a sector that is larger than 4KB will not be executed and will not set the E_ERR status.

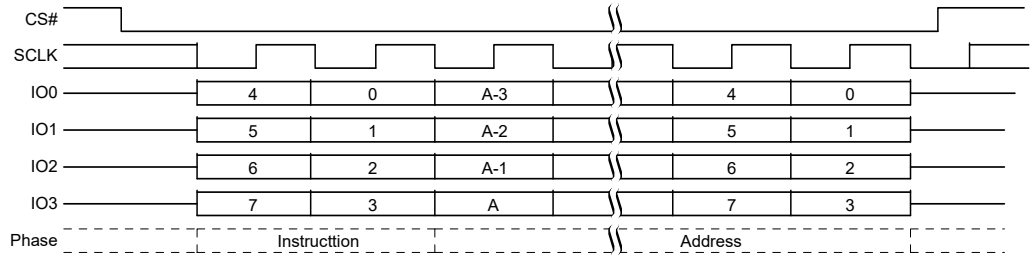
**Figure 91. Parameter Sector Erase (P4E 20h or 4P4E 21h) Command Sequence**




**Note**
78.A = MSB of address = A23 for SE 20h with CR2V[7] = 0, or A31 for SE 20h with CR2V[7] = 1 or for 4SE 21h.

This command is also supported in QPI mode. In QPI mode the instruction and address is shifted in on IO0–IO3.

**Figure 92. Parameter Sector Erase (P4E 20h or 4P4E 21h) QPI Mode Command Sequence**




**Note**
79.A = MSB of address = A23 for SE 20h with CR2V[7] = 0, or A31 for SE 20h with CR2V[7] = 1 or for 4SE 21h.

### 11.6.2 Sector Erase (SE D8h or 4SE DCh)

The Sector Erase (SE) command sets all bits in the addressed sector to 1 (all bytes are FFh). Before the Sector Erase (SE) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The instruction

- D8h [CR2V[7] = 0] is followed by a 3-byte address (A23–A0), or
- D8h [CR2V[7] = 1] is followed by a 4-byte address (A31–A0), or
- DCh is followed by a 4-byte address (A31–A0)

CS# must be driven into the logic high state after the twenty-fourth or thirty-second bit of address has been latched in on SI. This will initiate the erase cycle, which involves the pre-programming and erase of the chosen sector. If CS# is not driven high after the last bit of address, the sector erase operation will not be executed.

As soon as CS# is driven into the logic high state, the internal erase cycle will be initiated. With the internal erase cycle in progress, the user can read the value of the Write-In Progress (WIP) bit to check if the operation has been completed. The WIP bit will indicate a “1” when the erase cycle is in progress and a “0” when the erase cycle has been completed.

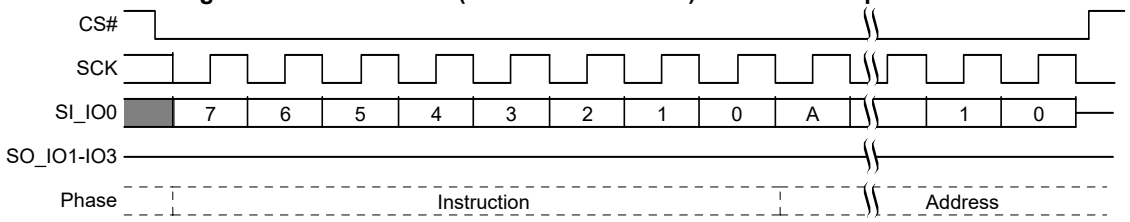
A Sector Erase (SE) command applied to a sector that has been Write Protected through the Block Protection bits or ASP, will not be executed and will set the E_ERR status.

A device configuration option (CR3V[1]) determines whether the SE command erases 64KB or 256KB.

A device configuration option (CR3V[3]) determines whether 4 KB parameter sectors are in use. When CR3V[3] = 0, parameter sectors overlay a portion of the highest or lowest address 32 KB of the device address space. If a sector erase command is applied to a 64KB sector that is overlaid by parameter sectors, the overlaid parameter sectors are not affected by the erase. Only the visible (non-overlaid) portion of the 64KB sector appears erased. Similarly if a sector erase command is applied to a 256 KB range that is overlaid by sectors, the overlaid parameter sectors are not affected by the erase. When CR3V[3] = 1, there are no parameter sectors in the device address space and the Sector Erase command always operates on fully visible 64 KB or 256KB sectors.

ASP has a PPB and a DYB protection bit for each physical sector, including any parameter sectors. If a sector erase command is applied to a 256KB range that includes a 64 KB protected physical sector, the erase will not be executed on the 256KB range and will set the E_ERR status.

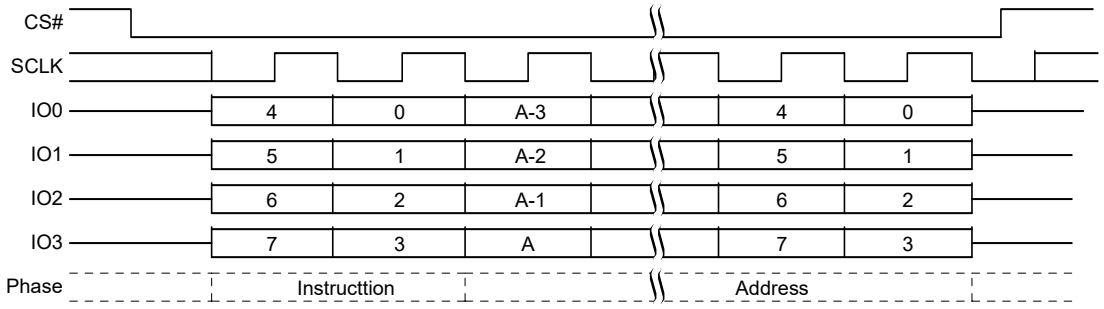
**Figure 93. Sector Erase (SE D8h or 4SE DCh) Command Sequence**




**Note**
80.A = MSB of address = A23 for SE D8h with CR2V[7] = 0, or A31 for SE D8h with CR2V[7] = 1 or 4SE DCh.

This command is also supported in QPI mode. In QPI mode the instruction and address is shifted in on IO0–IO3.

**Figure 94. Sector Erase (SE D8h or 4SE DCh) QPI Mode Command Sequence**




**Note**
81.A = MSB of address = A23 for SE D8h with CR2V[7] = 0, or A31 for SE D8h with CR2V[7] = 1 or 4SE DCh.

### 11.6.3 Bulk Erase (BE 60h or C7h)

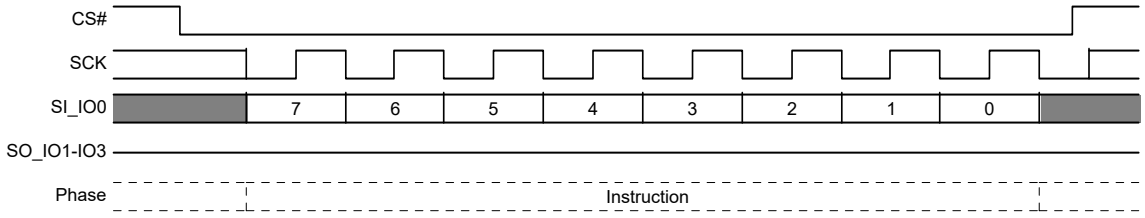
The Bulk Erase (BE) command sets all bits to 1 (all bytes are FFh) inside the entire flash memory array. Before the BE command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the Write Enable Latch (WEL) in the Status Register to enable any write operations.

CS# must be driven into the logic high state after the eighth bit of the instruction byte has been latched in on SI/IO0. This will initiate the erase cycle, which involves the pre-programming and erase of the entire flash memory array. If CS# is not driven high after the last bit of instruction, the BE operation will not be executed.

As soon as CS# is driven into the logic high state, the erase cycle will be initiated. With the erase cycle in progress, the user can read the value of the Write-In Progress (WIP) bit to determine when the operation has been completed. The WIP bit will indicate a “1” when the erase cycle is in progress and a “0” when the erase cycle has been completed.

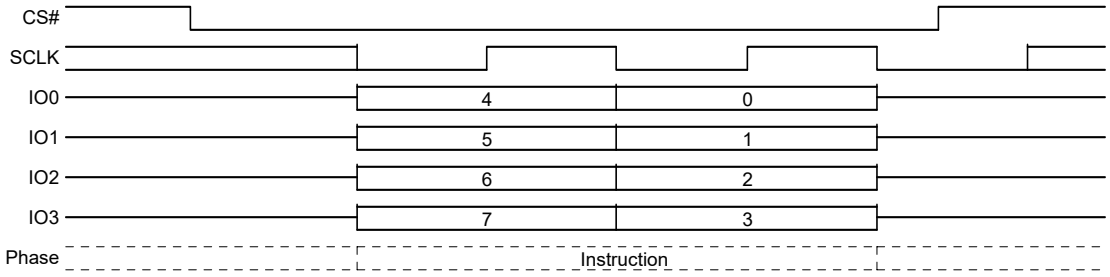
A BE command can be executed only when the Block Protection (BP2, BP1, BP0) bits are set to “0” s. If the BP bits are not zero, the BE command is not executed and E_ERR is not set. The BE command will skip any sectors protected by the DYB or PPB and the E_ERR status will not be set.

**Figure 95. Bulk Erase Command Sequence**




This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

**Figure 96. Bulk Erase Command Sequence QPI Mode**




### 11.6.4 Evaluate Erase Status (EES D0h)

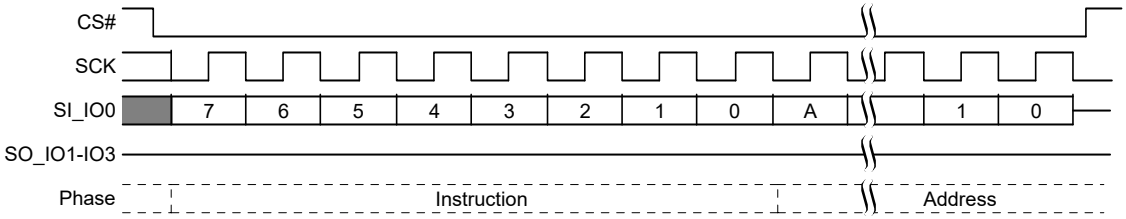
The Evaluate Erase Status (EES) command verifies that the last erase operation on the addressed sector was completed successfully. If the selected sector was successfully erased the erase status bit (SR2V[2]) is set to 1. If the selected sector was not completely erased SR2V[2] is 0.

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

The EES instruction is followed by a 3 or 4 byte address, depending on the address length configuration (CR2V[7]). The EES command requires tEES to complete and update the erase status in SR2V. The WIP bit (SR1V[0]) may be read using the RDSR1 (05h) command, to determine when the EES command is finished. Then the RDSR2 (07h) or the RDAR (65h) command can be used to read SR2V[2]. If a sector is found not erased with SR2V[2] = 0, the sector must be erased again to ensure reliable storage of data in the sector.

The Write Enable command (to set the WEL bit) is not required before the EES command. However, the WEL bit is set by the device itself and cleared at the end of the operation, as visible in SR1V[1] when reading status.

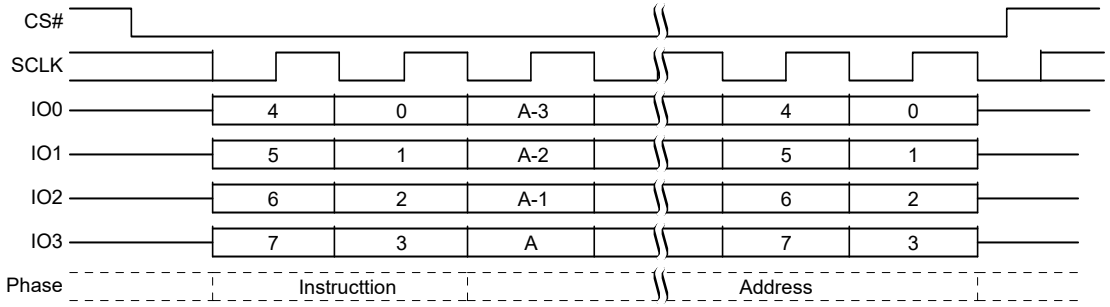
**Figure 97. EES Command Sequence**




**Note**
82.A = MSB of address = A23 for ESS D0h with CR2V[7] = 0, or A31 for ESS D0h with CR2V[7] = 1.

This command is also supported in QPI mode. In QPI mode the instruction and address is shifted in on IO0–IO3.

**Figure 98. EES QPI Mode Command Sequence**




**Note**
83.A = MSB of address = A23 for ESS D0h with CR2V[7] = 0, or A31 for ESS D0h with CR2V[7] = 1.

### 11.6.5 Erase or Program Suspend (EPS 85h, 75h, B0h)

There are three instruction codes for Program or Erase Suspend (EPS) to enable legacy and alternate source software compatibility.

The EPS command allows the system to interrupt a programming or erase operation and then read from any other non-erase-suspended sector or non-program-suspended-page. Program or Erase Suspend is valid only during a programming or sector erase operation. A Bulk Erase operation cannot be suspended.

The Write in Progress (WIP) bit in Status Register 1 (SR1V[0]) must be checked to know when the programming or erase operation has stopped. The Program Suspend Status bit in the Status Register-2 (SR2[0]) can be used to determine if a programming operation has been suspended or was completed at the time WIP changes to 0. The Erase Suspend Status bit in the Status Register-2 (SR2[1]) can be used to determine if an erase operation has been suspended or was completed at the time WIP changes to 0. The time required for the suspend operation to complete is $t_{SL}$, see Table 17 on page 39.

An Erase can be suspended to allow a program operation or a read operation. During an erase suspend, the DYB array may be read to examine sector protection and written to remove or restore protection on a sector to be programmed.

A program operation may be suspended to allow a read operation.

A new erase operation is not allowed with an already suspended erase or program operation. An erase command is ignored in this situation.

**Table 55. Commands Allowed During Program or Erase Suspend**

| Instruction Name | Instruction Code (Hex) | Allowed During Erase Suspend | Allowed During Program Suspend | Comment |
|---|---|---|---|---|
| READ | 03 | X | X | All array reads allowed in suspend |
| RDSR1 | 05 | X | X | Needed to read WIP to determine end of suspend process |
| RDAR | 65 | X | X | Alternate way to read WIP to determine end of suspend process |
| WREN | 06 | X | | Required for program command within erase suspend. |
| RDSR2 | 07 | X | X | Needed to read suspend status to determine whether the operation is suspended or complete. |
| RUID | 4C | X | X | Read Unique ID is allowed in suspend |
| PP | 02 | X | | Required for array program during erase suspend. Only allowed if there is no other program suspended program operation (SR2V[0] = 0). A program command will be ignored while there is a suspended program. If a program command is sent for a location within an erase suspended sector the program operation will fail with the P_ERR bit set. |
| 4PP | 12 | X | | Required for array program during erase suspend. Only allowed if there is no other program suspended program operation (SR2V[0] = 0). A program command will be ignored while there is a suspended program. If a program command is sent for a location within an erase suspended sector the program operation will fail with the P_ERR bit set. |
| QPP | 32 | X | | Required for array program during erase suspend. Only allowed if there is no other program suspended program operation (SR2V[0] = 0). A program command will be ignored while there is a suspended program. If a program command is sent for a location within an erase suspended sector the program operation will fail with the P_ERR bit set. |
| 4QPP | 34 | X | | Required for array program during erase suspend. Only allowed if there is no other program suspended program operation (SR2V[0] = 0). A program command will be ignored while there is a suspended program. If a program command is sent for a location within an erase suspended sector the program operation will fail with the P_ERR bit set. |
| 4READ | 13 | X | X | All array reads allowed in suspend |
| CLSR | 30 | X | | Clear status may be used if a program operation fails during erase suspend. Note the instruction is only valid if enabled for clear status by CR4NV[2] = 1 |
| CLSR | 82 | X | | Clear status may be used if a program operation fails during erase suspend. |
| EPR | 30 | X | X | Required to resume from erase or program suspend. Note the command must be enabled for use as a resume command by CR3NV[2] = 1 |
| EPR | 7A | X | X | Required to resume from erase or program suspend. |
| EPR | 8A | X | X | Required to resume from erase or program suspend. |
| RSTEN | 66 | X | X | Reset allowed anytime |
| RST | 99 | X | X | Reset allowed anytime |
| FAST_READ | 0B | X | X | All array reads allowed in suspend |
| 4FAST_READ | 0C | X | X | All array reads allowed in suspend |
| DOR | 3B | X | X | All array reads allowed in suspend |
| 4DOR | 3C | X | X | All array reads allowed in suspend |
| QOR | 6B | X | X | Read Quad Output (3 or 4 Byte Address) |
| 4QOR | 6C | X | X | Read Quad Output (4 Byte Address) |
| EPR | 7A | X | | Required to resume from erase suspend. |
| EPR | 8A | X | | Required to resume from erase suspend. |
| DIOR | BB | X | X | All array reads allowed in suspend |
| 4DIOR | BC | X | X | All array reads allowed in suspend |
| DYBRD | FA | X | | It may be necessary to remove and restore dynamic protection during erase suspend to allow programming during erase suspend. |
| DYBWR | FB | X | | It may be necessary to remove and restore dynamic protection during erase suspend to allow programming during erase suspend. |

**Table 55. Commands Allowed During Program or Erase Suspend** (Continued)

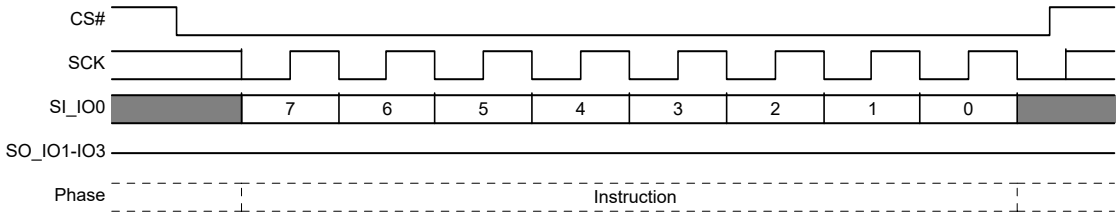
| Instruction Name | Instruction Code (Hex) | Allowed During Erase Suspend | Allowed During Program Suspend | Comment |
|---|---|---|---|---|
| PPBRD | FC | X | | Allowed for checking persistent protection before attempting a program command during erase suspend. |
| 4DYBRD | E0 | X | | It may be necessary to remove and restore dynamic protection during erase suspend to allow programming during erase suspend. |
| 4DYBWR | E1 | X | | It may be necessary to remove and restore dynamic protection during erase suspend to allow programming during erase suspend. |
| 4PPBRD | E2 | X | | Allowed for checking persistent protection before attempting a program command during erase suspend. |
| QIOR | EB | X | X | All array reads allowed in suspend |
| 4QIOR | EC | X | X | All array reads allowed in suspend |
| DDRQIOR | ED | X | X | All array reads allowed in suspend |
| 4DDRQIOR | EE | X | X | All array reads allowed in suspend |
| RESET | F0 | X | X | Reset allowed anytime |
| MBR | FF | X | X | May need to reset a read operation during suspend |

Reading at any address within an erase-suspended sector or program-suspended page produces undetermined data.

The WRR, WRAR, or PPB Erase commands are not allowed during Erase or Program Suspend, it is therefore not possible to alter the Block Protection or PPB bits during Erase Suspend. If there are sectors that may need programming during Erase suspend, these sectors should be protected only by DYB bits that can be turned off during Erase Suspend.

After an erase-suspended program operation is complete, the device returns to the erase-suspend mode. The system can determine the status of the program operation by reading the WIP bit in the Status Register, just as in the standard program operation.

**Figure 99. Program or Erase Suspend Command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0-IO3.

**Figure 100. Program or Erase Suspend Command Sequence QPI mode**

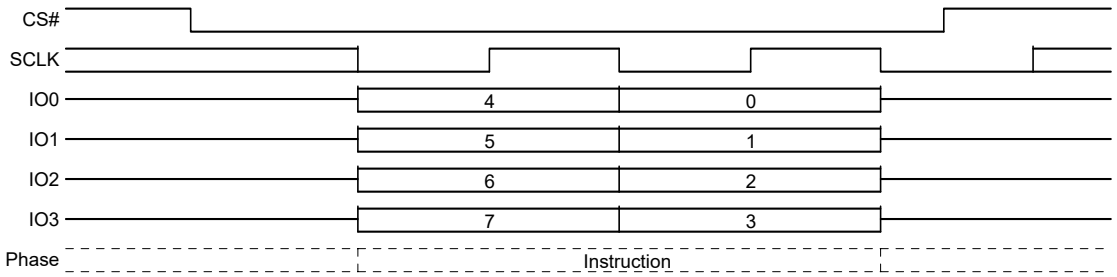
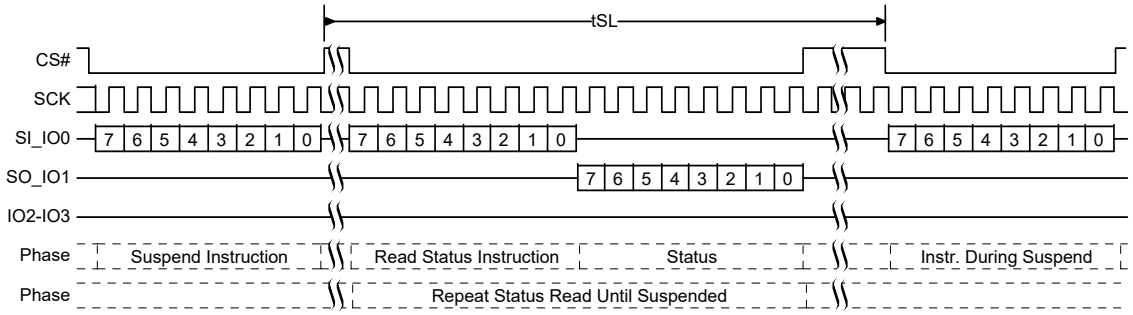


**Figure 101. Program or Erase Suspend Command with Continuing Instruction Commands Sequence**




### 11.6.6 Erase or Program Resume (EPR 7Ah, 8Ah, 30h)

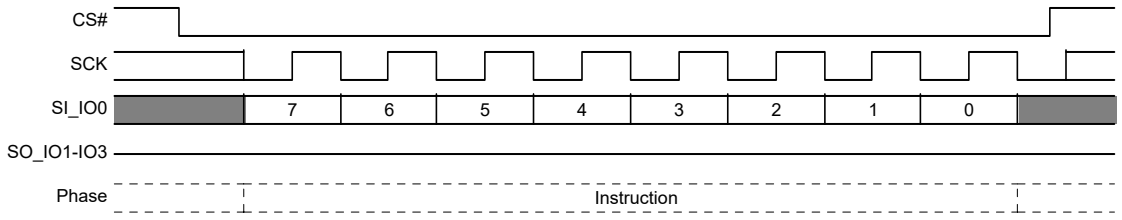
An Erase or Program Resume command must be written to resume a suspended operation. There are three instruction codes for Erase or Program Resume (EPR) to enable legacy and alternate source software compatibility.

After program or read operations are completed during a program or erase suspend the Erase or Program Resume command is sent to continue the suspended operation.

After an Erase or Program Resume command is issued, the WIP bit in the Status Register-1 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 or erase operation the resume command is ignored.

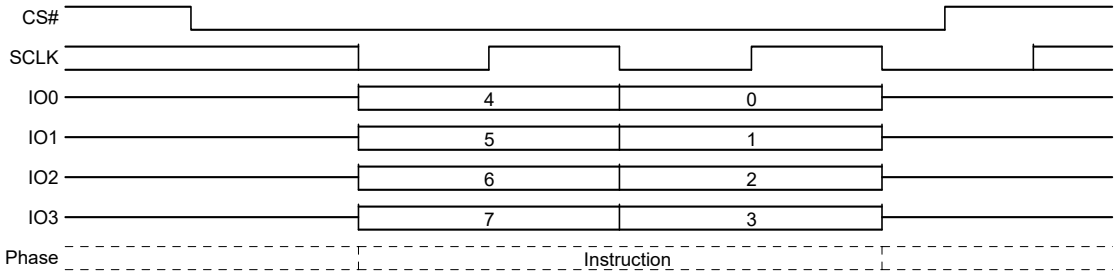
Program or erase operations may be interrupted as often as necessary e.g. a program suspend command could immediately follow a program resume command but, in order for a program or erase operation to progress to completion there must be some periods of time between resume and the next suspend command greater than or equal to $t_{RS}$. See Table 17 on page 39.

**Figure 102. Erase or Program Resume command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0–IO3.

**Figure 103. Erase or Program Resume command Sequence QPI mode**



## 11.7 One Time Program Array Commands

### 11.7.1 OTP Program (OTPP 42h)

The OTP Program command programs data in the One Time Program region, which is in a different address space from the main array data. The OTP region is 1024 bytes so, the address bits from A31 to A10 must be zero for this command. Refer to OTP Address Space on page 47 for details on the OTP region.

Before the OTP Program command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the Write Enable Latch (WEL) in the Status Register to enable any write operations. The WIP bit in SR1V may be checked to determine when the operation is completed. The P_ERR bit in SR1V 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 region in the OTP 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 P_ERR bit in SR1V set to “1”. Programming ones, even in a protected area does not cause an error and does not set P_ERR. Subsequent OTP programming can be performed only on the un-programmed bits (that is, “1” data). Programming more than once within an ECC unit will disable ECC on that unit.

The protocol of the OTP Program command is the same as the Page Program command. See Page Program (PP 02h or 4PP 12h) on page 102 for the command sequence.

### 11.7.2 OTP Read (OTPR 4Bh)

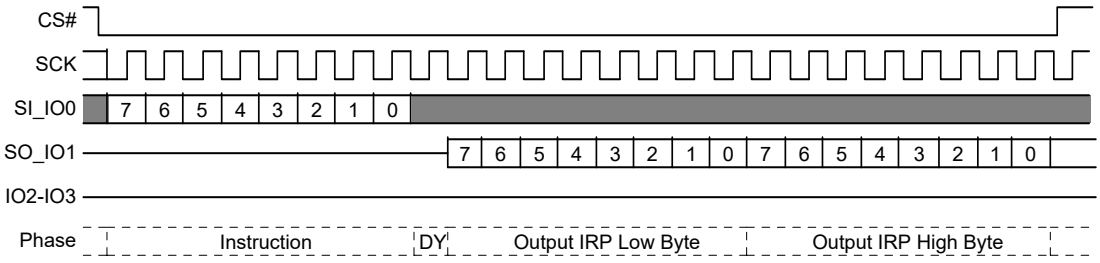
The OTP Read command reads data from the OTP region. The OTP region is 1024 bytes so, the address bits from A31 to A10 must be zero for this command. Refer to OTP Address Space on page 47 for details on the OTP region. The protocol of the OTP Read command is similar to the Fast Read command except that it will not wrap to the starting address after the OTP address is at its maximum; instead, the data beyond the maximum OTP address will be undefined. The OTP Read command read latency is set by the latency value in CR2V[3:0]. See Fast Read (FAST_READ 0Bh or 4FAST_READ 0Ch) on page 94 for the command sequence.

## 11.8 Advanced Sector Protection Commands

### 11.8.1 ASP Read (ASPRD 2Bh)

The ASP Read instruction 2Bh is shifted into SI by the rising edge of the SCK signal. Then the 16-bit ASP register contents are shifted out on the serial output SO, least significant byte first. Each bit is shifted out at the SCK frequency by the falling edge of the SCK signal. It is possible to read the ASP register continuously by providing multiples of 16 clock cycles. The maximum operating clock frequency for the ASP Read (ASPRD) command is 133 MHz.

**Figure 104. ASPRD Command Sequence**



### 11.8.2 ASP Program (ASPP 2Fh)

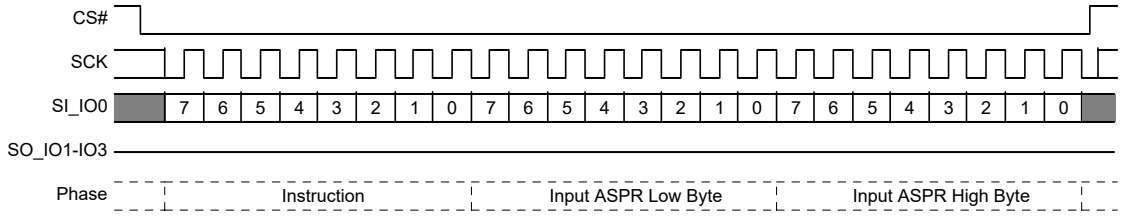
Before the ASP Program (ASPP) command can be accepted by the device, a Write Enable (WREN) command must be issued. After the Write Enable (WREN) command has been decoded, the device will set the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The ASPP command is entered by driving CS# to the logic low state, followed by the instruction and two data bytes on SI, least significant byte first. The ASP Register is two data bytes in length.

The ASPP command affects the P_ERR and WIP bits of the Status and Configuration Registers in the same manner as any other programming operation.

CS# input must be driven to the logic high state after the sixteenth bit of data has been latched in. If not, the ASPP command is not executed. As soon as CS# is driven to the logic high state, the self-timed ASPP operation is initiated. While the ASPP operation is in progress, the Status Register may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a “1” during the self-timed ASPP operation, and is a “0” when it is completed. When the ASPP operation is completed, the Write Enable Latch (WEL) is set to a “0”.

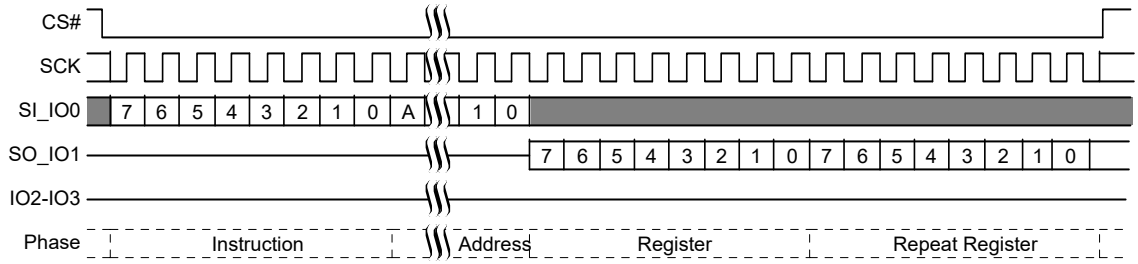
**Figure 105. ASPP Command**




### 11.8.3 DYB Read (DYBRD FAh or 4DYBRD E0h)

The instruction is latched into SI/IO0 by the rising edge of the SCK signal. The instruction is followed by the 24 or 32-Bit address, depending on the address length configuration CR2V[7], selecting location zero within the desired sector. Note, the high order address bits not used by a particular density device must be zero. Then the 8-bit DYB access register contents are shifted out on the serial output SO/IO1. Each bit is shifted out at the SCK frequency by the falling edge of the SCK signal. It is possible to read the same DYB access register continuously by providing multiples of eight clock cycles. The address of the DYB register does not increment so this is not a means to read the entire DYB array. Each location must be read with a separate DYB Read command. The maximum operating clock frequency for READ command is 133 MHz.

**Figure 106. DYBRD Command Sequence**



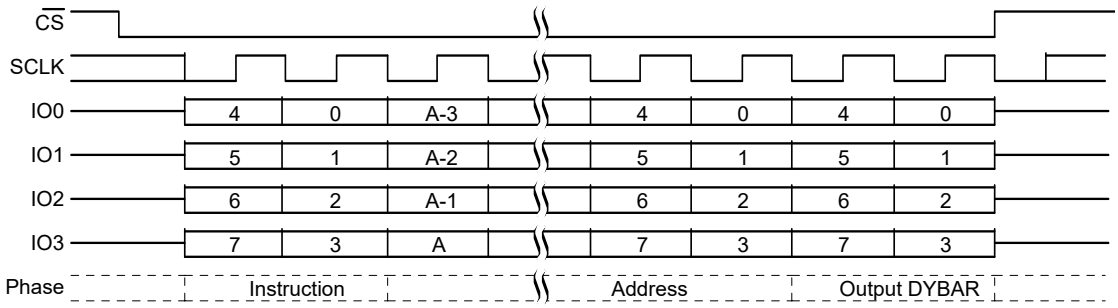

**Notes**

84.A = MSB of address = 23 for Address length (CR2V[7] = 0, or 31 for CR2V[7] = 1 with command FAh.

85.A = MSB of address = 31 with command E0h.

In QPI mode, the instruction is shifted in on IO0–IO3.

**Figure 107. DYBRD QPI Mode Command Sequence**




**Notes**

86.A = MSB of address = 23 for Address length (CR2V[7] = 0, or 31 for CR2V[7] = 1 with command FAh.
87.A = MSB of address = 31 with command E0hDYBRD QPI Mode Command Sequence.

### 11.8.4 DYB Write (DYBWR FBh or 4DYBWR E1h)

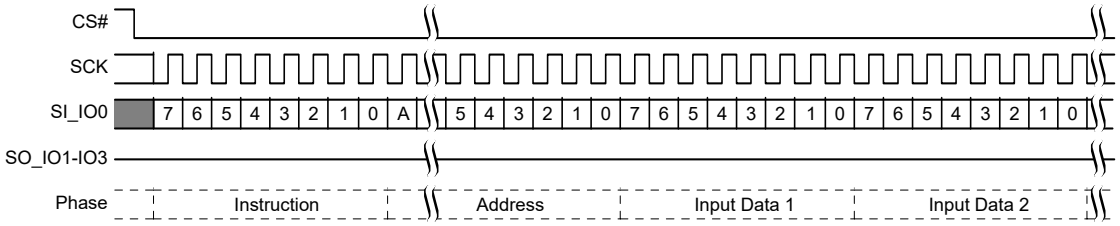
Before the DYB Write (DYBWR) command can be accepted by the device, a Write Enable (WREN) command must be issued. After the Write Enable (WREN) command has been decoded, the device will set the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The DYBWR command is entered by driving CS# to the logic low state, followed by the instruction, followed by the 24 or 32-Bit address, depending on the address length configuration CR2V[7], selecting location zero within the desired sector (note, the high order address bits not used by a particular density device must be zero), then the data byte on SI/IO0. The DYB Access Register is one data byte in length. The data value must be 00h to protect or FFh to unprotect the selected sector.

The DYBWR command affects the P_ERR and WIP bits of the Status and Configuration Registers in the same manner as any other programming operation.

CS# must be driven to the logic high state after the eighth bit of data has been latched in. As soon as CS# is driven to the logic high state, the self-timed DYBWR operation is initiated. While the DYBWR operation is in progress, the Status Register may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a “1” during the self-timed DYBWR operation, and is a “0” when it is completed. When the DYBWR operation is completed, the Write Enable Latch (WEL) is set to a “0”.

**Figure 108. DYB Write Command Sequence**

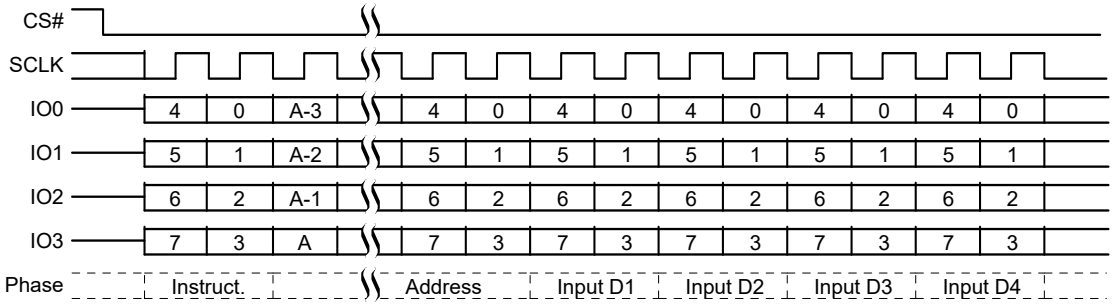



**Notes**

88.A= MSB of address = 23 for Address length (CR2V[7] = 0, or 31 for CR2V[7] = 1 with command FBh.
89.A = MSB of address = 31 with command E1h.

This command is also supported in QPI mode. In QPI mode, the instruction, address and data is shifted in on IO0–IO3.

**Figure 109. DYB Write QPI Mode Command Sequence**




**Notes**

90.A = MSB of address = 23 for Address length (CR2V[7] = 0, or 31 for CR2V[7] = 1 with command FBh.

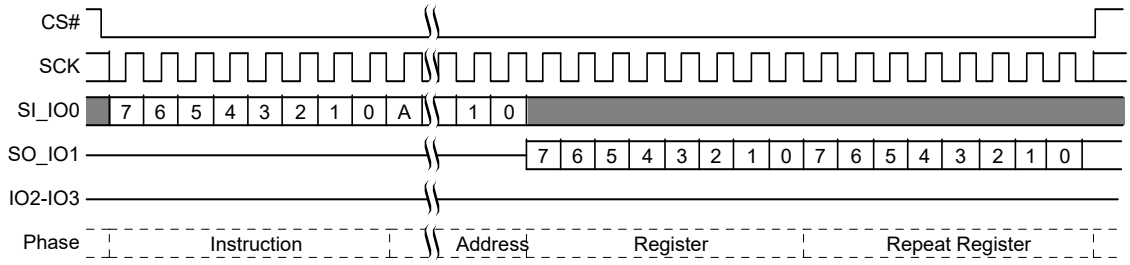
91.A = MSB of address = 31 with command E1h.

### 11.8.5 PPB Read (PPBRD FCh or 4PPBRD E2h)

The instruction E2h is shifted into SI/IO0 by the rising edges of the SCK signal, followed by the 24 or 32-Bit address, depending on the address length configuration CR2V[7], selecting location zero within the desired sector (note, the high order address bits not used by a particular density device must be zero). Then the 8-bit PPB access register contents are shifted out on SO/IO1.

It is possible to read the same PPB access register continuously by providing multiples of eight clock cycles. The address of the PPB register does not increment so this is not a means to read the entire PPB array. Each location must be read with a separate PPB Read command. The maximum operating clock frequency for the PPB Read command is 133 MHz.

**Figure 110. PPB Read Command Sequence**




**Notes**

92.A = MSB of address = 23 for Address length (CR2V[0] = 0, or 31 for CR2V[0]=1 with command FCh.

93.A = MSB of address = 31 with command E2h.

### 11.8.6 PPB Program (PPBP FDh or 4PPBP E3h)

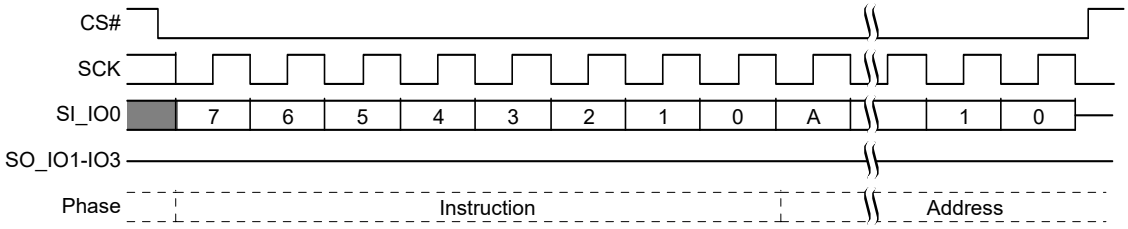
Before the PPB Program (PPBP) command can be accepted by the device, a Write Enable (WREN) command must be issued. After the Write Enable (WREN) command has been decoded, the device will set the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The PPBP command is entered by driving CS# to the logic low state, followed by the instruction, followed by the 24 or 32-Bit address, depending on the address length configuration CR2V[7], selecting location zero within the desired sector (note, the high order address bits not used by a particular density device must be zero).

The PPBP command affects the P_ERR and WIP bits of the Status and Configuration Registers in the same manner as any other programming operation.

CS# must be driven to the logic high state after the last bit of address has been latched in. If not, the PPBP command is not executed. As soon as CS# is driven to the logic high state, the self-timed PPBP operation is initiated. While the PPBP operation is in progress, the Status Register may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a “1” during the self-timed PPBP operation, and is a “0” when it is completed. When the PPBP operation is completed, the Write Enable Latch (WEL) is set to a “0”.

**Figure 111. PPB Command Sequence**




**Notes**
94.A = MSB of address = 23 for Address length (CR2V[0] = 0, or 31 for CR2V[0] = 1 with command FDh.
95.A = MSB of address = 31 with command E3h.

### 11.8.7 PPB Erase (PPBE E4h)

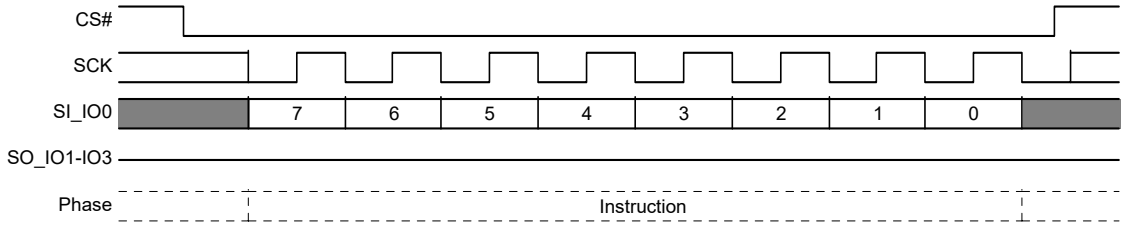
The PPB Erase (PPBE) command sets all PPB bits to 1. Before the PPB Erase command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The instruction E4h is shifted into SI/IO0 by the rising edges of the SCK signal.

CS# must be driven into the logic high state after the eighth bit of the instruction byte has been latched in on SI/IO0. This will initiate the beginning of internal erase cycle, which involves the pre-programming and erase of the entire PPB memory array. Without CS# being driven to the logic high state after the eighth bit of the instruction, the PPB erase operation will not be executed.

With the internal erase cycle in progress, the user can read the value of the Write-In Progress (WIP) bit to check if the operation has been completed. The WIP bit will indicate a “1” when the erase cycle is in progress and a “0” when the erase cycle has been completed. Erase suspend is not allowed during PPB Erase.

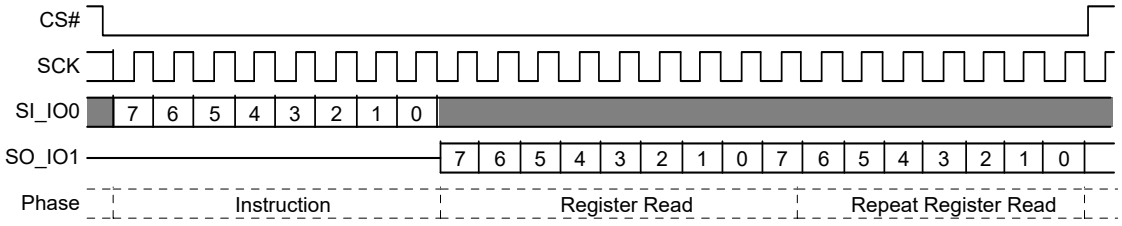
**Figure 112. PPB Erase Command Sequence**




### 11.8.8 PPB Lock Bit Read (PLBRD A7h)

The PPB Lock Bit Read (PLBRD) command allows the PPB Lock Register contents to be read out of SO/IO1. It is possible to read the PPB lock register continuously by providing multiples of eight clock cycles. The PPB Lock Register contents may only be read when the device is in standby state with no other operation in progress. It is recommended to check the Write-In Progress (WIP) bit of the Status Register before issuing a new command to the device.

**Figure 113. PPB Lock Register Command Sequence**



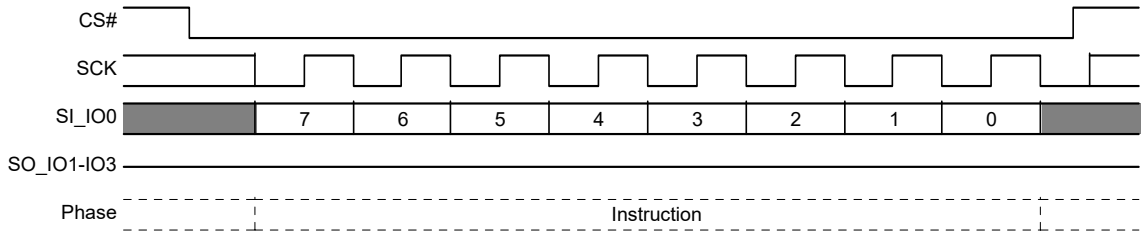
### 11.8.9 PPB Lock Bit Write (PLBWR A6h)

The PPB Lock Bit Write (PLBWR) command clears the PPB Lock Register to zero. Before the PLBWR command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The PLBWR command is entered by driving CS# to the logic low state, followed by the instruction.

CS# must be driven to the logic high state after the eighth bit of instruction has been latched in. If not, the PLBWR command is not executed. As soon as CS# is driven to the logic high state, the self-timed PLBWR operation is initiated. While the PLBWR operation is in progress, the Status Register may still be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a “1” during the self-timed PLBWR operation, and is a “0” when it is completed. When the PLBWR operation is completed, the Write Enable Latch (WEL) is set to a “0”. The maximum clock frequency for the PLBWR command is 133 MHz.

**Figure 114. PPB Lock Bit Command Sequence**

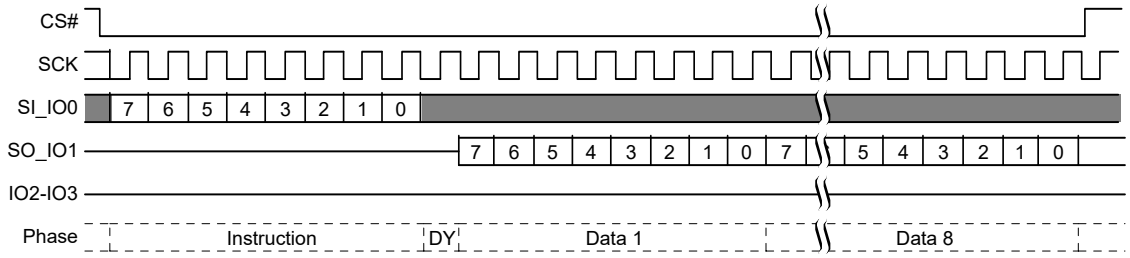



### 11.8.10 Password Read (PASSRD E7h)

The correct password value may be read only after it is programmed and before the Password Mode has been selected by programming the Password Protection Mode bit to 0 in the ASP Register (ASP[2]). After the Password Protection Mode is selected the password is no longer readable, the PASSRD command will output undefined data.

The PASSRD command is shifted into SI/IO0. Then the 64-bit Password is shifted out on the serial output SO/IO1, least significant byte first, most significant bit of each byte first. Each bit is shifted out at the SCK frequency by the falling edge of the SCK signal. It is possible to read the Password continuously by providing multiples of 64 clock cycles. The maximum operating clock frequency for the PASSRD command is 133 MHz.

**Figure 115. Password Read (PASSRD) Command Sequence**



### 11.8.11 Password Program (PASSP E8h)

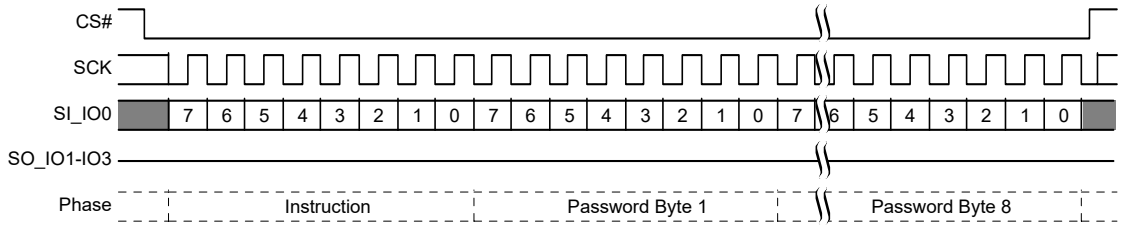
Before the Password Program (PASSP) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device. After the Write Enable (WREN) command has been decoded, the device sets the Write Enable Latch (WEL) to enable the PASSP operation.

The password can only be programmed before the Password Mode is selected by programming the Password Protection Mode bit to 0 in the ASP Register (ASP[2]). After the Password Protection Mode is selected the PASSP command is ignored.

The PASSP command is entered by driving CS# to the logic low state, followed by the instruction and the password data bytes on SI/IO0, least significant byte first, most significant bit of each byte first. The password is sixty-four (64) bits in length.

CS# must be driven to the logic high state after the sixty-fourth (64$^{th}$) bit of data has been latched. If not, the PASSP command is not executed. As soon as CS# is driven to the logic high state, the self-timed PASSP operation is initiated. While the PASSP operation is in progress, the Status Register may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a “1” during the self-timed PASSP cycle, and is a “0” when it is completed. The PASSP command can report a program error in the P_ERR bit of the status register. When the PASSP operation is completed, the Write Enable Latch (WEL) is set to a “0”. The maximum clock frequency for the PASSP command is 133 MHz.

**Figure 116. Password Program (PASSP) Command Sequence**




### 11.8.12 Password Unlock (PASSU E9h)

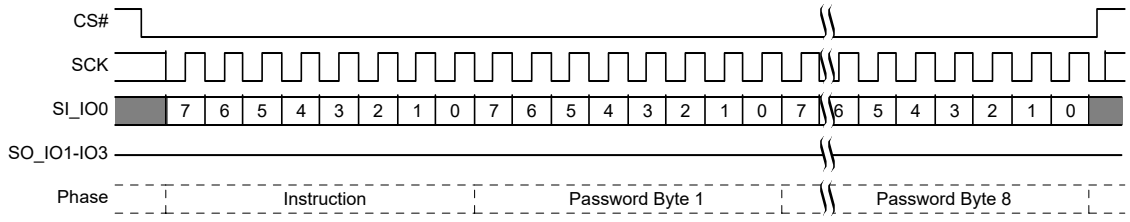
The PASSU command is entered by driving CS# to the logic low state, followed by the instruction and the password data bytes on SI/IO0, least significant byte first, most significant bit of each byte first. The password is sixty-four (64) bits in length.

CS# must be driven to the logic high state after the sixty-fourth (64$^{th}$) bit of data has been latched. If not, the PASSU command is not executed. As soon as CS# is driven to the logic high state, the self-timed PASSU operation is initiated. While the PASSU operation is in progress, the Status Register may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a “1” during the self-timed PASSU cycle, and is a “0” when it is completed.

If the PASSU command supplied password does not match the hidden password in the Password Register, an error is reported by setting the P_ERR bit to 1. The WIP bit of the status register also remains set to 1. It is necessary to use the CLSR command to clear the status register, the RESET command to software reset the device, or drive the RESET# input low to initiate a hardware reset, in order to return the P_ERR and WIP bits to 0. This returns the device to standby state, ready for new commands such as a retry of the PASSU command.

If the password does match, the PPB Lock bit is set to “1”. The maximum clock frequency for the PASSU command is 133 MHz.

**Figure 117. Password Unlock (PASSU) Command Sequence**



## 11.9 Reset Commands

Software controlled Reset commands restore the device to its initial power up state, by reloading volatile registers from non-volatile default values. However, the volatile FREEZE bit in the Configuration register CR1V[0] and the volatile PPB Lock bit in the PPB Lock Register are not changed by a software reset. The software reset cannot be used to circumvent the FREEZE or PPB Lock bit protection mechanisms for the other security configuration bits.

The Freeze bit and the PPB Lock bit will remain set at their last value prior to the software reset. To clear the FREEZE bit and set the PPB Lock bit to its protection mode selected power on state, a full power-on-reset sequence or hardware reset must be done.

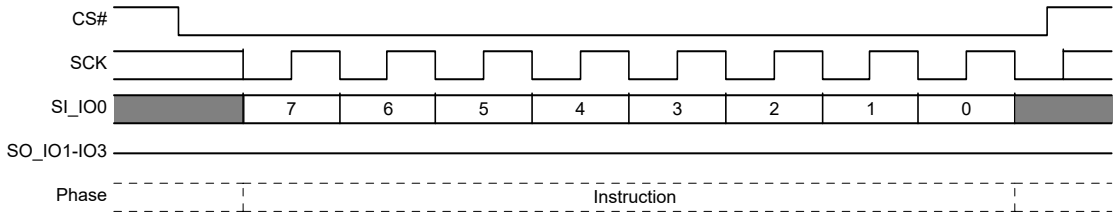
The non-volatile bits in the configuration register (CR1NV), TBPROT_O, TBPARM, and BPNV_O, retain their previous state after a Software Reset.

The Block Protection bits BP2, BP1, and BP0, in the status register (SR1V) will only be reset to their default value if FREEZE = 0.

A reset command (RST or RESET) is executed when CS# is brought high at the end of the instruction and requires $t_{RPH}$ time to execute.

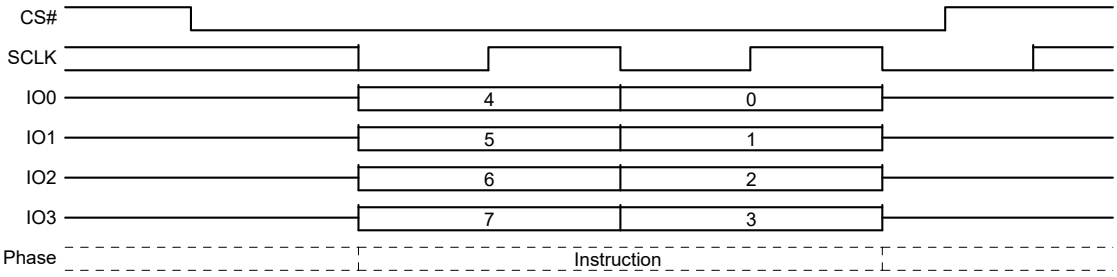
In the case of a previous Power-up Reset (POR) failure to complete, a reset command triggers a full power up sequence requiring $t_{PU}$ to complete.

**Figure 118. Software / Mode Bit Reset Command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0–IO3.

**Figure 119. Software Reset / Mode Bit Command Sequence – QPI Mode**




### 11.9.1 Software Reset Enable (RSTEN 66h)

The Reset Enable (RSTEN) command is required immediately before a Reset command (RST) such that a software reset is a sequence of the two commands. Any command other than RST following the RSTEN command, will clear the reset enable condition and prevent a later RST command from being recognized.

### 11.9.2 Software Reset (RST 99h)

The Reset (RST) command immediately following a RSTEN command, initiates the software reset process.

### 11.9.3 Legacy Software Reset (RESET F0h)

The Legacy Software Reset (RESET) is a single command that initiates the software reset process. This command is disabled by default but can be enabled by programming CR3V[0]=1, for software compatibility with Cypress legacy FL-S devices.

### 11.9.4 Mode Bit Reset (MBR FFh)

The Mode Bit Reset (MBR) command is used to return the device from continuous high performance read mode back to normal standby awaiting any new command. Because some device packages lack a hardware RESET# input and a device that is in a continuous high performance read mode may not recognize any normal SPI command, a system hardware reset or software reset command may not be recognized by the device. It is recommended to use the MBR command after a system reset when the RESET# signal is not available or, before sending a software reset, to ensure the device is released from continuous high performance read mode.

The MBR command sends Ones on SI/IO0 8 SCK cycles. IO1–IO3 are “don’t care” during these cycles. This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0–IO3, two clock cycles per byte.

## 11.10 DPD Commands

### 11.10.1 Enter Deep Power Down (DPD B9h)

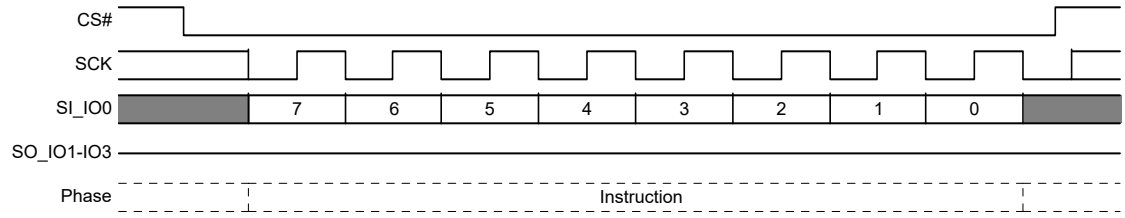
Although the standby current during normal operation is relatively low, standby current can be further reduced with the Deep Power Down command. The lower power consumption makes the Deep Power Down (DPD) command especially useful for battery powered applications (see $I_{DPD}$ in DC Characteristics on page 26).

The DPD command is accepted only while the device is not performing an embedded algorithm as indicated by the Status Register-1 volatile Write In Progress (WIP) bit being cleared to zero (SR1V[0] = 0).

The command is initiated by driving the CS# pin low and shifting the instruction code “B9h” as shown in Figure 120 on page 119. The CS# pin must be driven high after the eighth bit has been latched. If this is not done the Deep Power-down command will not be executed. After CS# is driven high, the power-down state will be entered within the time duration of $t_{DPD}$ (see Timing Specifications on page 30).

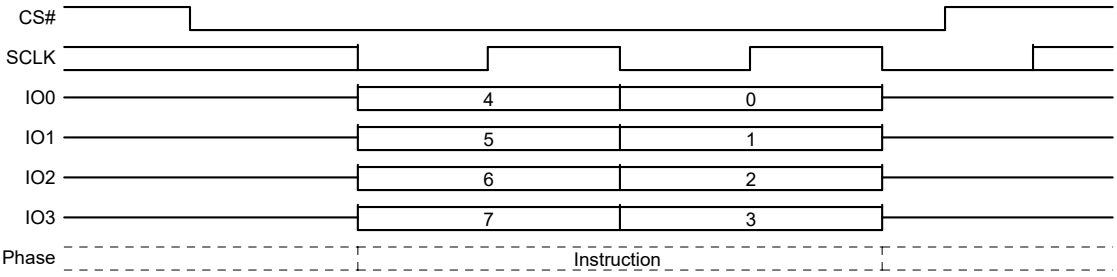
While in the power-down state only the Release from Deep Power-down command, which restores the device to normal operation, will be recognized. All other commands are ignored. This includes the Read Status Register command, which is always available during normal operation. Ignoring all but one command also makes the Power Down state useful for write protection. The device always powers-up in the interface standby state with the standby current of $I_{CC1}$.

**Figure 120. Deep Power Down (DPD) Command Sequence**




This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0–IO3.

**Figure 121. Deep Power Down (DPD) Command Sequence – QPI Mode**



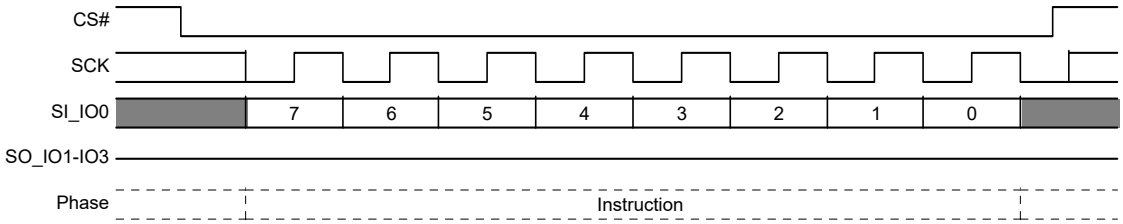
### 11.10.2 Release from Deep Power Down (RES ABh)

The Release from Deep Power-down command is used to release the device from the deep power-down state. In some legacy SPI devices the RES command could also be used to obtain the device electronic identification (ID) number. However, the device ID function is not supported by the RES command.

To release the device from the deep power-down state, the command is issued by driving the CS# pin low, shifting the instruction code “ABh” and driving CS# high as shown in Figure 122 on page 120. Release from deep power-down will take the time duration of $t_{RES}$ (see Timing Specifications on page 30) before the device will resume normal operation and other commands are accepted. The CS# pin must remain high during the $t_{RES}$ time duration.

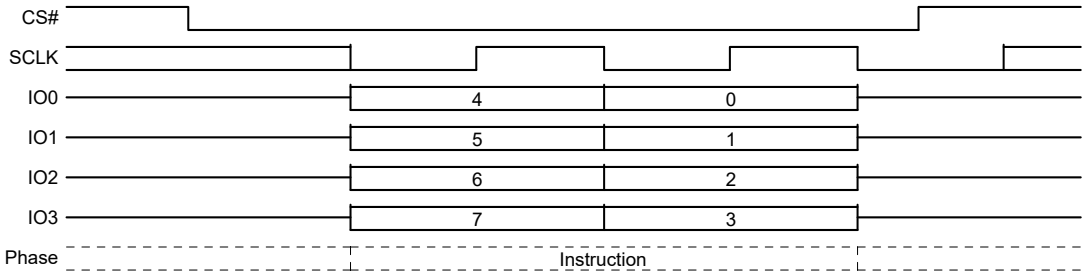
Hardware Reset will also release the device from the DPD state as part of the hardware reset process.

**Figure 122. Release from Deep Power Down (RES) Command Sequence**




This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

**Figure 123. Release from Deep Power Down (RES) Command Sequence – QPI Mode**



# 12. Data Integrity

## 12.1 Erase Endurance

**Table 56. Erase Endurance**

| Parameter | Minimum | Unit |
|---|---|---|
| Program/Erase cycles per main Flash array sectors | 100K | P/E cycle |
| Program/Erase cycles per PPB array or non-volatile register array (96) | 100K | P/E cycle |

**Note:**
96. Each write command to a non-volatile register causes a P/E cycle on the entire non-volatile register array. OTP bits and registers internally reside in a separate array that is not P/E cycled.

## 12.2 Data Retention

**Table 57. Data Retention**

| Parameter | Test Conditions | Minimum Time | Unit |
|---|---|---|---|
| Data Retention Time | 10K Program/Erase Cycles | 20 | Years |
| | 100K Program/Erase Cycles | 2 | Years |

Contact Cypress Sales and FAE for further information on the data integrity. Refer to the AN98549 - Endurance and Retention Management and Validation for more details.

# 13. Software Interface Reference

## 13.1 OTP Memory Space Address Map

The SFDP address space has a header starting at address zero that identifies the SFDP data structure and provides a pointer to each parameter. One parameter is mandated by the JEDEC JESD216 Rev B standard. Cypress provides an additional parameter by pointing to the ID-CFI address space i.e. the ID-CFI address space is a sub-set of the SFDP address space. The JEDEC parameter is located within the ID-CFI address space and is thus both a CFI parameter and an SFDP parameter. In this way both SFDP and ID-CFI information can be accessed by either the RSFDP or RDID commands.

**Table 58. SFDP Overview Map**

| Byte Address | Description |
|---|---|
| 0000h | Location zero within JEDEC JESD216B SFDP space - start of SFDP header |
| ,,, | Remainder of SFDP header followed by undefined space |
| 1000h | Location zero within ID-CFI space - start of ID-CFI parameter tables |
| ... | ID-CFI parameters |
| 1090h | Start of SFDP parameter tables which are also grouped as one of the CFI parameter tables (the CFI parameter itself starts at 108Eh, the SFDP parameter table data is double word aligned starting at 1090h) |
| ... | Remainder of SFDP parameter tables followed by either more CFI parameters or undefined space |

## 13.2 Device ID and Common Flash Interface (ID-CFI) Address Map — Standard

### 13.2.1 Field Definitions

**Table 59. Manufacturer and Device ID**

| Byte Address | Data | Description |
|---|---|---|
| 00h | 01h | Manufacturer ID for Cypress |
| 01h | 02h | Device ID Most Significant Byte - Memory Interface Type |
| 02h | 17h (64Mb) | Device ID Least Significant Byte - Density |
| 03h | 4Dh | ID-CFI Length - number bytes following. Adding this value to the current location of 03h gives the address of the last valid location in the ID-CFI legacy address map. The legacy CFI address map ends with the Primary Vendor-Specific Extended Query. The original legacy length is maintained for backward software compatibility. However, the CFI Query Identification String also includes a pointer to the Alternate Vendor-Specific Extended Query that contains additional information related to the FS-S family. |
| 04h | 00h (Uniform 256KB physical sectors)<br>01h (Uniform 64KB physical sectors) | Physical Sector Architecture<br>The FS-S Family may be configured with or without 4KB parameter sectors in addition to the uniform sectors. |
| 05h | 81h (FS-S Family) | Family ID |
| 06h | xxh | ASCII characters for Model<br>Refer to Ordering Part Number on page 141 for the model number definitions. |
| 07h | xxh | |
| 08h | xxh | Reserved |
| 09h | xxh | Reserved |
| 0Ah | xxh | Reserved |
| 0Bh | xxh | Reserved |
| 0Ch | xxh | Reserved |
| 0Dh | xxh | Reserved |
| 0Eh | xxh | Reserved |
| 0Fh | xxh | Reserved |

**Table 60. CFI Query Identification String**

| Byte Address | Data | Description |
|---|---|---|
| 10h<br>11h<br>12h | 51h<br>52h<br>59h | Query Unique ASCII string “QRY” |
| 13h<br>14h | 02h<br>00h | Primary OEM Command Set<br>FL-P backward compatible command set ID |
| 15h<br>16h | 40h<br>00h | Address for Primary Extended Table |
| 17h<br>18h | 53h<br>46h | Alternate OEM Command Set<br>Ascii characters “FS” for SPI (F) interface, S Technology |
| 19h<br>1Ah | 51h<br>00h | Address for Alternate OEM Extended Table |

**Table 61. CFI System Interface String**

| Byte Address | Data | Description |
|---|---|---|
| 1Bh | 17h | $V_{CC}$ Min. (erase/program): 100 millivolts BCD) |
| 1Ch | 19h | $V_{CC}$ Max. (erase/program): 100 millivolts BCD) |
| 1Dh | 00h | $V_{PP}$ Min. voltage (00h = no $V_{PP}$ present) |
| 1Eh | 00h | $V_{PP}$ Max. voltage (00h = no $V_{PP}$ present) |
| 1Fh | 09h | Typical timeout per single byte program $2^N$ µs |
| 20h | 09h | Typical timeout for Min. size Page program $2^N$ µs<br>(00h = not supported) |
| 21h | 08h (4KB or 64KB) | Typical timeout per individual sector erase $2^N$ ms |
| 22h | 05h (64 Mb) | Typical timeout for full chip erase $2^N$ ms (00h = not supported) |
| 23h | 02h | Max. timeout for byte program $2^N$ times typical |
| 24h | 02h | Max. timeout for page program $2^N$ times typical |
| 25h | 03h | Max. timeout per individual sector erase $2^N$ times typical |
| 26h | 02h | Max. timeout for full chip erase $2^N$ times typical<br>(00h = not supported) |

**Table 62. Device Geometry Definition for Bottom Boot Initial Delivery State**

| Byte Address | Data | Description |
|---|---|---|
| 27h | 17h (64 Mb) | Device Size = $2^N$ bytes; |
| 28h | 02h | Flash Device Interface Description;<br>0000h = x8 only<br>0001h = x16 only<br>0002h = x8/x16 capable<br>0003h = x32 only<br>0004h = Single I/O SPI, 3-byte address<br>0005h = Multi I/O SPI, 3-byte address<br>0102h = Multi I/O SPI, 3 or 4 byte address |
| 29h | 01h | |
| 2Ah | 08h | Max. number of bytes in multi-byte write = $2^N$<br>0000h = not supported<br>0008h = 256B page<br>0009h = 512B page |
| 2Bh | 00h | |
| 2Ch | 03h | Number of Erase Block Regions within device<br>1 = Uniform Device, >1 = Boot Device |

**Table 62. Device Geometry Definition for Bottom Boot Initial Delivery State** (Continued)

| Byte Address | Data | Description |
|---|---|---|
| 2Dh | 07h | Erase Block Region 1 Information (refer to JEDEC JEP137)<br>8 sectors = 8-1 = 0007h<br>4 KB sectors = 256 Bytes x 0010h |
| 2Eh | 00h | |
| 2Fh | 10h | |
| 30h | 00h | |
| 31h | 00h | Erase Block Region 2 Information (refer to JEDEC JEP137)<br>1 sectors = 1-1 = 0000h<br>32 KB sector = 256 Bytes x 0080h |
| 32h | 00h | |
| 33h | 80h | |
| 34h | 00h | |
| 35h | 7Eh (64Mb) | Erase Block Region 3 Information<br>127 sectors = 127-1 = 007Eh (64 Mb) |
| 36h | 00h | |
| 37h | 00h | |
| 38h | 00h | |
| 39h thru 3Fh | FFh | RFU |

**Note**

97. FS-S MD devices are user configurable to have either a hybrid sector architecture (with eight 4KB sectors and all remaining sectors are uniform 64KB or 256KB) or a uniform sector architecture with all sectors uniform 64KB or 256KB. FS-S devices are also user configurable to have the 4KB parameter sectors at the top of memory address space. The CFI geometry information of the above table is relevant only to the initial delivery state. All devices are initially shipped from Cypress with the hybrid sector architecture with the 4KB sectors located at the bottom of the array address map. However, the device configuration TBPARM bit CR1NV[2] may be programed to invert the sector map to place the 4KB sectors at the top of the array address map. The 20h_NV bit (CR3NV[3} may be programmed to remove the 4KB sectors from the address map. The Flash device driver software must examine the TBPARM and 20h_NV bits to determine if the sector map was inverted or hybrid sectors removed at a later time.

**Table 63. CFI Primary Vendor-Specific Extended Query**

| Byte Address | Data | Description |
|---|---|---|
| 40h | 50h | Query-unique ASCII string “PRI” |
| 41h | 52h | |
| 42h | 49h | |
| 43h | 31h | Major version number = 1, ASCII |
| 44h | 33h | Minor version number = 3, ASCII |
| 45h | 21h | Address Sensitive Unlock (Bits 1-0)<br>00b = Required, 01b = Not Required<br>Process Technology (Bits 5-2)<br>0000b = 0.23 µm Floating Gate<br>0001b = 0.17 µm Floating Gate<br>0010b = 0.23 µm MirrorBit<br>0011b = 0.11 µm Floating Gate<br>0100b = 0.11 µm MirrorBit<br>0101b = 0.09 µm MirrorBit<br>1000b = 0.065 µm MirrorBit |
| 46h | 02h | Erase Suspend<br>0 = Not Supported, 1 = Read Only, 2 = Read and Program |
| 47h | 01h | Sector Protect<br>00 = Not Supported, X = Number of sectors in group |
| 48h | 00h | Temporary Sector Unprotect<br>00 = Not Supported, 01 = Supported |
| 49h | 08h | Sector Protect/Unprotect Scheme<br>04 = High Voltage Method<br>05 = Software Command Locking Method<br>08 = Advanced Sector Protection Method |
| 4Ah | 00h | Simultaneous Operation<br>00 = Not Supported, X = Number of Sectors |

**Table 63. CFI Primary Vendor-Specific Extended Query** (Continued)

| Byte Address | Data | Description |
|---|---|---|
| 4Bh | 01h | Burst Mode (Synchronous sequential read) support<br>00 = Not Supported, 01 = Supported |
| 4Ch | 03h | Page Mode Type, initial delivery configuration, user configurable for 512B page<br>00 = Not Supported, 01 = 4 Word Read Page, 02 = 8 Read Word Page, 03 = 256 Byte Program Page, 04 = 512 Byte Program Page |
| 4Dh | 00h | ACC (Acceleration) Supply Minimum<br>00 = Not Supported, 100 mV |
| 4Eh | 00h | ACC (Acceleration) Supply Maximum<br>00 = Not Supported, 100 mV |
| 4Fh | 07h | WP# Protection<br>01 = Whole Chip<br>04 = Uniform Device with Bottom WP Protect<br>05 = Uniform Device with Top WP Protect<br>07 = Uniform Device with Top or Bottom Write Protect (user configurable) |
| 50h | 01h | Program Suspend<br>00 = Not Supported, 01 = Supported |

The Alternate Vendor-Specific Extended Query provides information related to the expanded command set provided by the FS-S Family. The alternate query parameters use a format in which each parameter begins with an identifier byte and a parameter length byte. Driver software can check each parameter ID and can use the length value to skip to the next parameter if the parameter is not needed or not recognized by the software.

**Table 64. CFI Alternate Vendor-Specific Extended Query Header**

| Byte Address | Data | Description |
|---|---|---|
| 51h | 41h | Query-unique ASCII string “ALT” |
| 52h | 4Ch | |
| 53h | 54h | |
| 54h | 32h | Major version number = 2, ASCII |
| 55h | 30h | Minor version number = 0, ASCII |

**Table 65. CFI Alternate Vendor-Specific Extended Query Parameter 0**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 56h | 00h | Parameter ID (Ordering Part Number) |
| 57h | 10h | Parameter Length (The number of following bytes in this parameter. Adding this value to the current location value +1 = the first byte of the next parameter) |
| 58h | 53h | Ascii “S” for manufacturer (Cypress) |
| 59h | 32h | Ascii “25” for Product Characters (Single Die SPI) |
| 5Ah | 35h | |
| 5Bh | 46h | Ascii “FS” for Interface Characters (SPI 1.8Volt) |
| 5Ch | 53h | |
| 5Dh | 30h (64 Mb) | Ascii characters for density |
| 5Eh | 36h (64Mb) | |
| 5Fh | 34h (64Mb) | |
| 60h | 53h | Ascii “S” for Technology (65nm MirrorBit) |
| 61h | FFh | Reserved for Future Use |
| 62h | FFh | |
| 63h | FFh | Reserved for Future Use |
| 64h | FFh | |
| 65h | FFh | Reserved for Future Use |

**Table 65. CFI Alternate Vendor-Specific Extended Query Parameter 0** (Continued)

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 66h | xxh | ASCII characters for Model. Refer to Ordering Part Number on page 141 for the model number definitions. |
| 67h | xxh | |

**Table 66. CFI Alternate Vendor-Specific Extended Query Parameter 80h Address Options**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 68h | 80h | Parameter ID (Ordering Part Number) |
| 69h | 01h | Parameter Length (The number of following bytes in this parameter. Adding this value to the current location value +1 = the first byte of the next parameter) |
| 6Ah | EBh | Bits 7:5 - Reserved = 111b<br>Bit 4 - Address Length Bit in CR2V[7] - Yes = 0b<br>Bit 3 - AutoBoot support - No = 1b<br>Bit 2 - 4 byte address instructions supported - Yes = 0b<br>Bit 1 - Bank address + 3 byte address instructions supported - No = 1b<br>Bit 0 - 3 byte address instructions supported - No = 1b |

**Table 67. CFI Alternate Vendor-Specific Extended Query Parameter 84h Suspend Commands**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 6Bh | 84h | Parameter ID (Suspend Commands |
| 6Ch | 08h | Parameter Length (The number of following bytes in this parameter. Adding this value to the current location value +1 = the first byte of the next parameter) |
| 6Dh | 85h | Program suspend instruction code |
| 6Eh | 2Dh | Program suspend latency maximum (uS) |
| 6Fh | 8Ah | Program resume instruction code |
| 70h | 64h | Program resume to next suspend typical (uS) |
| 71h | 75h | Erase suspend instruction code |
| 72h | 2Dh | Erase suspend latency maximum (uS) |
| 73h | 7Ah | Erase resume instruction code |
| 74h | 64h | Erase resume to next suspend typical (uS) |

**Table 68. CFI Alternate Vendor-Specific Extended Query Parameter 88h Data Protection**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 75h | 88h | Parameter ID (Data Protection) |
| 76h | 04h | Parameter Length (The number of following bytes in this parameter. Adding this value to the current location value +1 = the first byte of the next parameter) |
| 77h | 0Ah | OTP size 2^N bytes, FFh = not supported |
| 78h | 01h | OTP address map format, 01h = FL-S and FS-S format, FFh = not supported |
| 79h | xxh | Block Protect Type, model dependent<br>00h = FL-P, FL-S, FS-S<br>FFh = not supported |
| 7Ah | xxh | Advanced Sector Protection type, model dependent<br>01h = FL-S and FS-S ASP. |

**Table 69. CFI Alternate Vendor-Specific Extended Query Parameter 94h ECC**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 83h | 94h | Parameter ID (ECC) |
| 84h | 01h | Parameter Length (The number of following bytes in this parameter. Adding this value to the current location value +1 = the first byte of the next parameter) |
| 85h | 10h | ECC unit size byte, FFh = ECC disabled |

**Table 70. CFI Alternate Vendor-Specific Extended Query Parameter 8Ch Reset Timing**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 7Bh | 8Ch | Parameter ID (Reset Timing) |
| 7Ch | 06h | Parameter Length (The number of following bytes in this parameter. Adding this value to the current location value +1 = the first byte of the next parameter) |
| 7Dh | 96h | POR maximum value |
| 7Eh | 01h | POR maximum exponent 2^N uS |
| 7Fh | 23h | Hardware Reset maximum value, FFh = not supported (the initial delivery state has hardware reset disabled but it may be enabled by the user at a later time) |
| 80h | 00h | Hardware Reset maximum exponent 2^N uS |
| 81h | 23h | Software Reset maximum value, FFh = not supported |
| 82h | 00h | Software Reset maximum exponent 2^N uS |

**Table 71. CFI Alternate Vendor-Specific Extended Query Parameter F0h RFU**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 83h | F0h | Parameter ID (RFU) |
| 84h | 09h | Parameter Length (The number of following bytes in this parameter. Adding this value to the current location value +1 = the first byte of the next parameter) |
| 85h | FFh | RFU |
| ... | FFh | RFU |
| 8Dh | FFh | RFU |

This parameter type (Parameter ID F0h) may appear multiple times and have a different length each time. The parameter is used to reserve space in the ID-CFI map or to force space (pad) to align a following parameter to a required boundary.

## 13.3 Serial Flash Discoverable Parameters (SFDP) Address Map

### 13.3.1 JEDEC SFDP Rev B Header Table

**Table 72. SFDP Header**

| SFDP Byte Address | SFDP Dword Name | Data | Description |
|---|---|---|---|
| 00h | SFDP Header 1st DWORD | 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 | | 44h | ASCII “D” |
| 03h | | 50h | ASCII “P” |
| 04h | SFDP Header 2nd DWORD | 06h | SFDP Minor Revision (06h = JEDEC JESD216 Revision B)<br>This revision is backward compatible with all prior minor revisions. Minor revisions are changes that define previously reserved fields, add fields to the end, or that clarify definitions of existing fields. Increments of the minor revision value indicate that previously reserved parameter fields may have been assigned a new definition or entire Dwords may have been added to the parameter table. However, the definition of previously existing fields is unchanged and therefore remain backward compatible with earlier SFDP parameter table revisions. Software can safely ignore increments of the minor revision number, as long as only those parameters the software was designed to support are used i.e. previously reserved fields and additional Dwords must be masked or ignored. Do not do a simple compare on the minor revision number, looking only for a match with the revision number that the software is designed to handle. There is no problem with using a higher number minor revision. |
| 05h | | 01h | SFDP Major Revision<br>This is the original major revision. This major revision is compatible with all SFDP reading and parsing software. |
| 06h | | 05h | Number of Parameter Headers (zero based, 05h = 6 parameters) |
| 07h | | FFh | Unused |
| 08h | Parameter Header 0 1st DWORD | 00h | Parameter ID LSB (00h = JEDEC SFDP Basic SPI Flash Parameter) |
| 09h | | 00h | Parameter Minor Revision (00h = JESD216)<br>- This older revision parameter header is provided for any legacy SFDP reading and parsing software that requires seeing a minor revision 0 parameter header. SFDP software designed to handle later minor revisions should continue reading parameter headers looking for a higher numbered minor revision that contains additional parameters for that software revision. |
| 0Ah | | 01h | Parameter Major Revision (01h = The original major revision - all SFDP software is compatible with this major revision. |
| 0Bh | | 09h | Parameter Table Length (in double words = Dwords = 4 byte units) 09h = 9 Dwords |
| 0Ch | Parameter Header 0 2nd DWORD | 90h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned)<br>JEDEC Basic SPI Flash parameter byte offset = 1090h |
| 0Dh | | 10h | Parameter Table Pointer Byte 1 |
| 0Eh | | 00h | Parameter Table Pointer Byte 2 |
| 0Fh | | FFh | Parameter ID MSB (FFh = JEDEC defined legacy Parameter ID) |
| 10h | Parameter Header 1 1st DWORD | 00h | Parameter ID LSB (00h = JEDEC SFDP Basic SPI Flash Parameter) |
| 11h | | 05h | Parameter Minor Revision (05h = JESD216 Revision A)<br>- This older revision parameter header is provided for any legacy SFDP reading and parsing software that requires seeing a minor revision 5 parameter header. SFDP software designed to handle later minor revisions should continue reading parameter headers looking for a later minor revision that<br>contains additional parameters. |
| 12h | | 01h | Parameter Major Revision (01h = The original major revision - all SFDP software is compatible with this major revision. |
| 13h | | 10h | Parameter Table Length (in double words = Dwords = 4 byte units) 10h = 16 Dwords |

**Table 72. SFDP Header** (Continued)

| SFDP Byte Address | SFDP Dword Name | Data | Description |
|---|---|---|---|
| 14h | Parameter Header 1 2nd DWORD | 90h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned) JEDEC Basic SPI Flash parameter byte offset = 1090h address |
| 15h | | 10h | Parameter Table Pointer Byte 1 |
| 16h | | 00h | Parameter Table Pointer Byte 2 |
| 17h | | FFh | Parameter ID MSB (FFh = JEDEC defined Parameter) |
| 18h | Parameter Header 2 1st DWORD | 00h | Parameter ID LSB (00h = JEDEC SFDP Basic SPI Flash Parameter) |
| 19h | | 06h | Parameter Minor Revision (06h = JESD216 Revision B) |
| 1Ah | | 01h | Parameter Major Revision (01h = The original major revision - all SFDP software is compatible with this major revision. |
| 1Bh | | 10h | Parameter Table Length (in double words = Dwords = 4 byte units) 10h = 16 Dwords |
| 1Ch | Parameter Header 2 2nd DWORD | 90h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned) JEDEC Basic SPI Flash parameter byte offset = 1090h address |
| 1Dh | | 10h | Parameter Table Pointer Byte 1 |
| 1Eh | | 00h | Parameter Table Pointer Byte 2 |
| 1Fh | | FFh | Parameter ID MSB (FFh = JEDEC defined Parameter) |
| 20h | Parameter Header 3 1st DWORD | 81h | Parameter ID LSB (81h = SFDP Sector Map Parameter) |
| 21h | | 00h | Parameter Minor Revision (00h = Initial version as defined in JESD216 Revision B) |
| 22h | | 01h | Parameter Major Revision (01h = The original major revision - all SFDP software that recognizes this parameter's ID is compatible with this major revision. |
| 23h | | 1Ah | Parameter Table Length (in double words = Dwords = 4 byte units) OPN Dependent 26 = 1Ah |
| 24h | Parameter Header 3 2nd DWORD | D8h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned) JEDEC parameter byte offset = 10D8h |
| 25h | | 10h | Parameter Table Pointer Byte 1 |
| 26h | | 00h | Parameter Table Pointer Byte 2 |
| 27h | | FFh | Parameter ID MSB (FFh = JEDEC defined Parameter) |
| 28h | Parameter Header 4 1st DWORD | 84h | Parameter ID LSB (00h = SFDP 4 Byte Address Instructions Parameter) |
| 29h | | 00h | Parameter Minor Revision (00h = Initial version as defined in JESD216 Revision B) |
| 2Ah | | 01h | Parameter Major Revision (01h = The original major revision - all SFDP software that recognizes this parameter's ID is compatible with this major revision. |
| 2Bh | | 02h | Parameter Table Length (in double words = Dwords = 4 byte units) (2h = 2 Dwords) |
| 2Ch | Parameter Header 4 2nd DWORD | D0h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned) JEDEC parameter byte offset = 10D0h |
| 2Dh | | 10h | Parameter Table Pointer Byte 1 |
| 2Eh | | 00h | Parameter Table Pointer Byte 2 |
| 2Fh | | FFh | Parameter ID MSB (FFh = JEDEC defined Parameter) |
| 30h | Parameter Header 5 1st DWORD | 01h | Parameter ID LSB (Cypress Vendor Specific ID-CFI parameter) Legacy Manufacturer ID 01h = AMD / Spansion |
| 31h | | 01h | Parameter Minor Revision (01h = ID-CFI updated with SFDP Rev B table) |
| 32h | | 01h | Parameter Major Revision (01h = The original major revision - all SFDP software that recognizes this parameter's ID is compatible with this major revision. |
| 33h | | 50h | Parameter Table Length (in double words = Dwords = 4 byte units) Parameter Table Length (in double words = Dwords = 4 byte units) |

**Table 72. SFDP Header** (Continued)

| SFDP Byte Address | SFDP Dword Name | Data | Description |
|---|---|---|---|
| 34h | Parameter Header 5 2nd DWORD | 00h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned)<br>Entry point for ID-CFI parameter is byte offset = 1000h relative to SFDP location zero. |
| 35h | | 10h | Parameter Table Pointer Byte 1 |
| 36h | | 00h | Parameter Table Pointer Byte 2 |
| 37h | | 01h | Parameter ID MSB (01h = JEDEC JEP106 Bank Number 1) |

### 13.3.2 JEDEC SFDP Rev B Parameter Tables

From the view point of the CFI data structure, all of the SFDP parameter tables are combined into a single CFI Parameter as a contiguous byte sequence.

From the viewpoint of 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 Basic Flash Parameter table and the 4-Byte Address Instructions Parameter table have a fixed length and are presented below as a single table. This table is section 1 of the overall CFI parameter.

The JEDEC Sector Map Parameter table structure and length depends on the density OPN and is presented as a set of tables, one for each device density. The appropriate table for the OPN is section 2 of the overall CFI parameter and is appended to section 1.

**Table 73. CFI and SFDP Section 1, Basic Flash and 4 Byte Address Instructions Parameter**

| CFI Parameter Relative Byte Address Offset from108Eh | SFDP Parameter Relative Byte Address Offset from1090h | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 00h | -- | N/A | A5h | CFI Parameter ID (JEDEC SFDP) |
| 01h | -- | N/A | B0h | CFI Parameter Length (The number of following bytes in this parameter. Adding this value to the current location value +1 = the first byte of the next parameter).<br>OPN dependent:<br>18Dw + 26Dw = 44Dw *4B = 176B = B0h B |
| 02h | 00h | JEDEC Basic Flash Parameter Dword-1 | E7h | Start of SFDP JEDEC parameter, located at 1090h in the overall SFDP address space.<br>Bits 7:5 = unused = 111b<br>Bit 4:3 = 06h is status register write instruction and status register is default non-volatile= 00b<br>Bit 2 = Program Buffer > 64Bytes = 1<br>Bits 1:0 = Uniform 4KB erase unavailable = 11b |
| 03h | 01h | | FFh | Bits 15:8 = Uniform 4KB erase opcode = not supported = FFh |
| 04h | 02h | | FBh | Bit 23 = Unused = 1b<br>Bit 22 = Supports Quad Out Read = Yes= 1b<br>Bit 21 = Supports Quad I/O Read = Yes =1b<br>Bit 20 = Supports Dual I/O Read = Yes = 1b<br>Bit19 = Supports DDR = Yes =1b;<br>Bit 18:17 = Number of Address Bytes, 3 or 4 = 01b<br>Bit 16 = Supports Dual Out Read = Yes =1b |
| 05h | 03h | | FFh | Bits 31:24 = Unused = FFh |
| 06h | 04h | JEDEC Basic Flash Parameter Dword-2 | FFh | Density in bits, zero based, 16Mb = 00FFFFFFh |
| 07h | 05h | | FFh | |
| 08h | 06h | | FFh | |
| 09h | 07h | | 03h (64 Mb) | |
| 0Ah | 08h | JEDEC Basic Flash Parameter Dword-3 | 48h | Bits 7:5 = number of Quad I/O (1-4-4) Mode cycles = 010b<br>Bits 4:0 = number of Quad I/O Dummy cycles = 01000b (Initial Delivery State) |
| 0Bh | 09h | | EBh | Quad I/O instruction code |
| 0Ch | 0Ah | | 08h | Bits 23:21 = number of Quad Out (1-1-4) Mode cycles = 000b<br>Bits 20:16 = number of Quad Out Dummy cycles = 01000b |
| 0Dh | 0Bh | | 6Bh | Quad Out instruction code |

**Table 73. CFI and SFDP Section 1, Basic Flash and 4 Byte Address Instructions Parameter** (Continued)

| CFI Parameter Relative Byte Address Offset from108Eh | SFDP Parameter Relative Byte Address Offset from1090h | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 0Eh | 0Ch | JEDEC Basic Flash Parameter Dword-4 | 08h | Bits 7:5 = number of Dual Out (1-1-2) Mode cycles = 000b<br>Bits 4:0 = number of Dual Out Dummy cycles = 01000b |
| 0Fh | 0Dh | | 3Bh | Dual Out instruction code |
| 10h | 0Eh | | 88h | Bits 23:21 = number of Dual I/O (1-2-2) Mode cycles = 100b<br>Bits 20:16 = number of Dual I/O Dummy cycles = 01000b (Initial Delivery State) |
| 11h | 0Fh | | BBh | Dual I/O instruction code |
| 12h | 10h | JEDEC Basic Flash Parameter Dword-5 | FEh | Bits 7:5 RFU = 111b<br>Bit 4 = QPI supported = Yes = 1b<br>Bits 3:1 RFU = 111b<br>Bit 0 = Dual All not supported = 0b |
| 13h | 11h | | FFh | Bits 15:8 = RFU = FFh |
| 14h | 12h | | FFh | Bits 23:16 = RFU = FFh |
| 15h | 13h | | FFh | Bits 31:24 = RFU = FFh |
| 16h | 14h | JEDEC Basic Flash Parameter Dword-6 | FFh | Bits 7:0 = RFU = FFh |
| 17h | 15h | | FFh | Bits 15:8 = RFU = FFh |
| 18h | 16h | | FFh | Bits 23:21 = number of Dual All Mode cycles = 111b<br>Bits 20:16 = number of Dual All Dummy cycles = 11111b |
| 19h | 17h | | FFh | Dual All instruction code |
| 1Ah | 18h | JEDEC Basic Flash Parameter Dword-7 | FFh | Bits 7:0 = RFU = FFh |
| 1Bh | 19h | | FFh | Bits 15:8 = RFU = FFh |
| 1Ch | 1Ah | | 48h | Bits 23:21 = number of QPI Mode cycles = 010b<br>Bits 20:16 = number of QPI Dummy cycles = 01000b |
| 1Dh | 1Bh | | EBh | QPI mode Quad I/O (4-4-4) instruction code |
| 1Eh | 1Ch | JEDEC Basic Flash Parameter Dword-8 | 0Ch | Erase type 1 size 2^N Bytes = 4KB = 0Ch for Hybrid (Initial Delivery State) |
| 1Fh | 1Dh | | 20h | Erase type 1 instruction |
| 20h | 1Eh | | 10h | Erase type 2 size 2^N Bytes = 64KB = 10h |
| 21h | 1Fh | | D8h | Erase type 2 instruction |
| 22h | 20h | JEDEC Basic Flash Parameter Dword-9 | 12h | Erase type 3 size 2^N Bytes = 256KB = 12h |
| 23h | 21h | | D8h | Erase type 3 instruction |
| 24h | 22h | | 00h | Erase type 4 size 2^N Bytes = not supported = 00h |
| 25h | 23h | | FFh | Erase type 4 instruction = not supported = FFh |
| 26h | 24h | JEDEC Basic Flash Parameter Dword-10 | B1h | Bits 31:30 = Erase type 4 Erase, Typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 1S = 11b (RFU)<br>Bits 29:25 = Erase type 4 Erase, Typical time count = 11111b (RFU)<br>Bits 24:23 = Erase type 3 Erase, Typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 128mS = 10b<br>Bits 22:18 = Erase type 3 Erase, Typical time count = 00111b (typ erase time = count +1 * units = 8*128mS = 1024mS)<br>Bits 17:16 = Erase type 2 Erase, Typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 16mS = 01b<br>Bits 15:11 = Erase type 2 Erase, Typical time count = 01110b (typ erase time = count +1 * units = 15*16mS = 240mS)<br>Bits 10:9 = Erase type 1 Erase, Typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 16mS = 01b<br>Bits 8:4 = Erase type 1 Erase, Typical time count = 01011b (typ erase time = count +1 * units = 12*16mS = 192mS)<br>Bits 3:0 = Multiplier from typical erase time to maximum erase time = 2*(N+1), N=2h = 4x multiplier<br><br>Binary Fields: 11-11111-10-00111-01-01110-01-01011-0001<br>Nibble Format: 1111_1111_0001_1101_0111_0010_1011_0001<br>Hex Format: FF_1D_72_B1 |
| 27h | 25h | | 72h | |
| 28h | 26h | | 1Dh | |
| 29h | 27h | | FFh | |

**Table 73. CFI and SFDP Section 1, Basic Flash and 4 Byte Address Instructions Parameter** (Continued)

| CFI Parameter Relative Byte Address Offset from108Eh | SFDP Parameter Relative Byte Address Offset from1090h | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 2Ah | 28h | JEDEC Basic Flash Parameter Dword-11 | 82h | Bit 31 Reserved = 1b<br>Bits 30:29 = Chip Erase, Typical time units (00b: 16 ms, 01b: 256 ms, 10b: 4 s, 11b: 64 s) = 64Mb = 4s = 10b;<br>Bits 28:24 = Chip Erase, Typical time count, (count+1)*units,<br>64 Mb = 00111b = (7+1)*4 = 32s;<br>Bits 23 = Byte Program Typical time, additional byte units (0b:1uS, 1b:8uS) = 1uS = 0b<br>Bits 22:19 = Byte Program Typical time, additional byte count, (count+1)*units, count = 0000b, ( typ Program time = count +1 * units = 1*1uS = 1uS<br>Bits 18 = Byte Program Typical time, first byte units (0b:1uS, 1b:8uS) = 8uS = 1b<br>Bits 17:14 = Byte Program Typical time, first byte count, (count+1)*units, count = 1100b, ( typ Program time = count +1 * units = 13*8uS = 104uS<br>Bits 13 = Page Program Typical time units (0b:8uS, 1b:64uS) = 64uS = 1b<br>Bits 12:8 = Page Program Typical time count, (count+1)*units, count = 00110b, ( typ Program time = count +1 * units = 6*64uS = 384uS)<br>Bits 7:4 = Page size 2^N, N=8h, = 256B page<br>Bits 3:0 = Multiplier from typical time to maximum for Page or Byte program = 2*(N+1), N=2h = 6x multiplier<br><br>64Mb<br>Binary Fields: 1-10-00111-0-0000-1-1100-1-00110-1000-0010<br>Nibble Format: 1100_0111_0000_0111_0010_0110_1001_0010<br>Hex Format: C7_07_26_82 |
| 2Bh | 29h | | 26h | |
| 2Ch | 2Ah | | 07h | |
| 2Dh | 2Bh | | C7h | |
| 2Eh | 2Ch | JEDEC Basic Flash Parameter Dword-12 | ECh | Bit 31 = Suspend and Resume supported = 0b<br>Bits 30:29 = Suspend in-progress erase max latency units (00b: 128ns, 01b: 1us, 10b: 8us, 11b: 64us) = 8us= 10b<br>Bits 28:24 = Suspend in-progress erase max latency count = 00101b, max erase suspend latency = count +1 * units = 6*8uS = 48uS<br>Bits 23:20 = Erase resume to suspend interval count = 0001b, interval = count +1 * 64us = 2 * 64us = 128us<br>Bits 19:18 = Suspend in-progress program max latency units (00b: 128ns, 01b: 1us, 10b: 8us, 11b: 64us) = 8us= 10b<br>Bits 17:13 = Suspend in-progress program max latency count = 00101b, max erase suspend latency = count +1 * units = 6*8uS = 48uS<br>Bits 12:9 = Program resume to suspend interval count = 0001b, interval = count +1 * 64us = 2 * 64us = 128us<br>Bit 8 = RFU = 1b<br>Bits 7:4 = Prohibited operations during erase suspend<br>= xxx0b: May not initiate a new erase anywhere (erase nesting not permitted)<br>+ xx1xb: May not initiate a page program in the erase suspended sector size<br>+ x1xxb: May not initiate a read in the erase suspended sector size<br>+ 1xxxb: The erase and program restrictions in bits 5:4 are sufficient<br>= 1110b<br>Bits 3:0 = Prohibited Operations During Program Suspend<br>= xxx0b: May not initiate a new erase anywhere (erase nesting not permitted)<br>+ xx0xb: May not initiate a new page program anywhere (program nesting not permitted)<br>+ x1xxb: May not initiate a read in the program suspended page size<br>+ 1xxxb: The erase and program restrictions in bits 1:0 are sufficient<br>= 1100b<br><br>Binary Fields: 0-10-00101-0001-10-00100-1001-1-1110-1100<br>Nibble Format: 0100_0101_0001_1000_1001_0011_1110_1100<br>Hex Format: 45_18_93_EC |
| 2Fh | 2Dh | | 93h | |
| 30h | 2Eh | | 18h | |
| 31h | 2Fh | | 45h | |
| 32h | 30h | JEDEC Basic Flash Parameter Dword-13 | 8Ah | Bits 31:24 = Erase Suspend Instruction = 75h<br>Bits 23:16 = Erase Resume Instruction = 7Ah<br>Bits 15:8 = Program Suspend Instruction = 85h<br>Bits 7:0 = Program Resume Instruction = 8Ah |
| 33h | 31h | | 85h | |
| 34h | 32h | | 7Ah | |
| 35h | 33h | | 75h | |

**Table 73. CFI and SFDP Section 1, Basic Flash and 4 Byte Address Instructions Parameter** (Continued)

| CFI Parameter Relative Byte Address Offset from108Eh | SFDP Parameter Relative Byte Address Offset from1090h | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 36h | 34h | JEDEC Basic Flash Parameter Dword-14 | F7h | Bit 31 = Deep Power Down Supported = supported = 0<br>Bits 30:23 = Enter Deep Power Down Instruction = B9h<br>Bits 22:15 = Exit Deep Power Down Instruction = ABh<br>Bits 14:13 = Exit Deep Power Down to next operation delay units = (00b: 128ns, 01b: 1us, 10b: 8us, 11b: 64us) = 1us = 01b<br>Bits 12:8 = Exit Deep Power Down to next operation delay count = 11101b, Exit Deep Power Down to next operation delay = (count+1)*units = 29+1 *1us = 30us<br>Bits 7:4 = RFU = Fh<br>Bit 3:2 = Status Register Polling Device Busy<br>= 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).<br>= 01b<br>Bits 1:0 = RFU = 11b<br><br>Binary Fields: 0-10111001-10101011-01-11101-1111-01-11<br>Nibble Format: 0101_1100_1101_0101_1011_1101_1111_0111<br>Hex Format: 5C_D5_BD_F7 |
| 37h | 35h | | BDh | |
| 38h | 36h | | D5h | |
| 39h | 37h | | 5Ch | |
| 3Ah | 38h | JEDEC Basic Flash Parameter Dword-15 | 8Ch | Bits 31:24 = RFU = FFh<br>Bit 23 = Hold and WP Disable = not supported = 0b<br>Bits 22:20 = Quad Enable Requirements<br>= 101b: QE is bit 1 of the status register 2. Status register 1 is read using Read Status instruction 05h. Status register 2 is read using instruction 35h. QE is set via Write Status instruction 01h with two data bytes where bit 1 of the second byte is one. It is cleared via Write Status with two data bytes where bit 1 of the second byte is zero.<br>Bits 19:16 0-4-4 Mode Entry Method<br>= xxx1b: Mode Bits[7:0] = A5h Note: QE must be set prior to using this mode<br>+ x1xxb: Mode Bit[7:0]=Axh<br>+ 1xxxb: RFU<br>= 1101b<br>Bits 15:10 0-4-4 Mode Exit Method<br>= xx_xxx1b: Mode Bits[7:0] = 00h will terminate this mode at the end of the current read operation<br>+ xx_1xxxb: Input Fh (mode bit reset) on DQ0-DQ3 for 8 clocks. This will terminate the mode prior to the next read operation.<br>+ x1_xxxxb: Mode Bit[7:0] != Axh<br>+ 1x_x1xx: RFU<br>= 11_1101<br>Bit 9 = 0-4-4 mode supported = 1<br>Bits 8:4 = 4-4-4 mode enable sequences<br>= x_1xxxb: device uses a read-modify-write sequence of operations: read configuration using instruction 65h followed by address 800003h, set bit 6, write configuration using instruction 71h followed by address 800003h. This configuration is volatile.<br>= 01000b<br>Bits 3:0 = 4-4-4 mode disable sequences<br>= x1xxb: device uses a read-modify-write sequence of operations: read configuration using instruction 65h followed by address 800003h, clear bit 6, write configuration using instruction 71h followed by address 800003h.. This configuration is volatile.<br>+ 1xxxb: issue the Soft Reset 66/99 sequence<br>= 1100b<br><br>Binary Fields: 11111111-0-101-1101-111101-1-01000-1100<br>Nibble Format: 1111_1111_0101_1101_1111_0110_1000-1100<br>Hex Format: FF_5D_F6_8C |
| 3Bh | 39h | | F6h | |
| 3Ch | 3Ah | | 5Dh | |
| 3Dh | 3Bh | | FFh | |

**Table 73. CFI and SFDP Section 1, Basic Flash and 4 Byte Address Instructions Parameter** (Continued)

| CFI Parameter Relative Byte Address Offset from108Eh | SFDP Parameter Relative Byte Address Offset from1090h | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 3Eh | 3Ch | JEDEC Basic Flash Parameter Dword-16 | F0h | Bits 31:24 = Enter 4-Byte Addressing<br>= xxxx_xxx1b: issue instruction B7h (preceding write enable not required)<br>+ xx1x_xxxxb: Supports dedicated 4-Byte address instruction set. Consult vendor data sheet for the instruction set definition.<br>+ 1xxx_xxxxb: Reserved<br>= 10100001b<br>Bits 23:14 = Exit 4-Byte Addressing<br>= xx_xx1x_xxxxb: Hardware reset<br>+ xx_x1xx_xxxxb: Software reset (see bits 13:8 in this DWORD)<br>+ xx_1xxx_xxxxb: Power cycle<br>+ x1_xxxx_xxxxb: Reserved<br>+ 1x_xxxx_xxxxb: Reserved<br>= 11_1110_0000b<br>Bits 13:8 = Soft Reset and Rescue Sequence Support<br>= x1_xxxxb: issue reset enable instruction 66h, then issue reset instruction 99h. The reset<br>enable, reset sequence may be issued on 1, 2, or 4 wires depending on the device operating mode.<br>+ 1x_xxxxb: exit 0-4-4 mode is required prior to other reset sequences above if the device<br>may be operating in this mode.<br>= 110000b<br>Bit 7 = RFU = 1<br>Bits 6:0 = Volatile or Non-Volatile Register and Write Enable Instruction for Status Register 1<br>= + xx1_xxxxb: Status Register 1 contains a mix of volatile and non-volatile bits. The 06h<br>instruction is used to enable writing of the register.<br>+ x1x_xxxxb: Reserved<br>+ 1xx_xxxxb: Reserved<br>= 1110000b<br><br>Binary Fields: 10100001-1111100000-110000-1-1110000<br>Nibble Format: 1010_0001_1111_1000_0011_0000_1111_0000<br>Hex Format: A1_F8_30_F0 |
| 3Fh | 3Dh | | 30h | |
| 40h | 3Eh | | F8h | |
| 41h | 3Fh | | A1h | |
| 42h | 40h | JEDEC 4 Byte Address Instructions Parameter Dword-1 | FFh | Supported = 1, Not Supported = 0<br>Bits 31:20 = RFU = FFFh<br>Bit 19 = Support for non-volatile individual sector lock write command, Instruction=E3h = 1<br>Bit 18 = Support for non-volatile individual sector lock read command, Instruction=E2h = 1<br>Bit 17 = Support for volatile individual sector lock Write command, Instruction=E1h = 1<br>Bit 16 = Support for volatile individual sector lock Read command, Instruction=E0h = 1<br>Bit 15 = Support for (1-4-4) DTR_Read Command, Instruction=EEh = 1<br>Bit 14 = Support for (1-2-2) DTR_Read Command, Instruction=BEh = 0<br>Bit 13 = Support for (1-1-1) DTR_Read Command, Instruction=0Eh = 0<br>Bit 12 = Support for Erase Command – Type 4 = 0<br>Bit 11 = Support for Erase Command – Type 3 = 1<br>Bit 10 = Support for Erase Command – Type 2 = 1<br>Bit 9 = Support for Erase Command – Type 1 = 1<br>Bit 8 = Support for (1-4-4) Page Program Command, Instruction=3Eh =0<br>Bit 7 = Support for (1-1-4) Page Program Command, Instruction=34h = 1<br>Bit 6 = Support for (1-1-1) Page Program Command, Instruction=12h = 1<br>Bit 5 = Support for (1-4-4) FAST_READ Command, Instruction=ECh = 1<br>Bit 4 = Support for (1-1-4) FAST_READ Command, Instruction=6Ch = 1<br>Bit 3 = Support for (1-2-2) FAST_READ Command, Instruction=BCh = 1<br>Bit 2 = Support for (1-1-2) FAST_READ Command, Instruction=3Ch = 1<br>Bit 1 = Support for (1-1-1) FAST_READ Command, Instruction=0Ch = 1<br>Bit 0 = Support for (1-1-1) READ Command, Instruction=13h = 1 |
| 43h | 41h | | CEh | |
| 44h | 42h | | FFh | |
| 45h | 43h | | FFh | |
| 46h | 44h | JEDEC 4 Byte Address Instructions Parameter Dword-2 | 21h | Bits 31:24 = FFh = Instruction for Erase Type 4: RFU<br>Bits 23:16 = DCh = Instruction for Erase Type 3<br>Bits 15:8 = DCh = Instruction for Erase Type 2<br>Bits 7:0 = 21h = Instruction for Erase Type 1 |
| 47h | 45h | | DCh | |
| 48h | 46h | | DCh | |
| 49h | 47h | | FFh | |

**Sector Map Parameter Table Notes:**

The following Sector Map Parameter Table 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.

To identify the sector map configuration in S25FS064S the following configuration bits are read in the following MSB to LSB order to form the configuration map index value:

- CR3NV[3] - 0 = Hybrid Architecture, 1 = Uniform Architecture
- CR1NV[2] - 0 = 4KB parameter sectors at bottom, 1 = 4KB sectors at top
- CR3NV[1] - 0= 64KB uniform sector size, 1 = 256KB uniform sector size

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.

| Device | CR3NV[3] | CR1NV[2] | CR3NV[1] | Index Value | Description |
|---|---|---|---|---|---|
| FS64S | 0 | 0 | 0 | 00h | 4KB sectors at bottom with remainder 64KB sectors |
| | 0 | 1 | 0 | 02h | 4KB sectors at top with remainder 64KB sectors |
| | 0 | 0 | 1 | 01h | 4KB sectors at bottom with remainder 256KB sectors |
| | 0 | 1 | 1 | 03h | 4KB sectors at top with remainder 256KB sectors |
| | 1 | 0 | 0 | 04h | Uniform 64KB sectors |
| | 1 | 0 | 0 | 05h | Uniform 256KB sectors |

**Table 74. CFI and SFDP Section 2, Sector Map Parameter Table**

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 4Ah | 48h | JEDEC Sector Map Parameter Dword-1 Config. Detect-1 | FCh | Bits 31:24 = Read data mask = 0000_1000b: Select bit 3 of the data byte for 20h_NV value<br>0= Hybrid map with 4KB parameter sectors<br>1= Uniform map<br>Bits 23:22 = Configuration detection command address length = 11b: Variable length<br>Bits 21:20 = RFU = 11b<br>Bits 19:16 = Configuration detection command latency = 1111b: variable latency<br>Bits 15:8 = Configuration detection instruction = 65h: Read any register<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Command Descriptor = 0<br>Bit 0 = not the end descriptor = 0 |
| 4Bh | 49h | | 65h | |
| 4Ch | 4Ah | | FFh | |
| 4Dh | 4Bh | | 08h | |
| 4Eh | 4Ch | JEDEC Sector Map Parameter Dword-2 Config. Detect-1 | 04h | Bits 31:0 = Sector map configuration detection command address = 00_00_00_04h: address of CR3NV |
| 4Fh | 4Dh | | 00h | |
| 50h | 4Eh | | 00h | |
| 51h | 4Fh | | 00h | |
| 52h | 50h | JEDEC Sector Map Parameter Dword-3 Config. Detect-2 | FCh | Bits 31:24 = Read data mask = 0000_0100b: Select bit 2 of the data byte for TBPARM_O value<br>0= 4KB parameter sectors at bottom<br>1= 4KB parameter sectors at top<br>Bits 23:22 = Configuration detection command address length = 11b: Variable length<br>Bits 21:20 = RFU = 11b<br>Bits 19:16 = Configuration detection command latency = 1111b: variable latency<br>Bits 15:8 = Configuration detection instruction = 65h: Read any register<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Command Descriptor = 0<br>Bit 0 = not the end descriptor = 0 |
| 53h | 51h | | 65h | |
| 54h | 52h | | FFh | |
| 55h | 53h | | 04h | |

**Table 74. CFI and SFDP Section 2, Sector Map Parameter Table** (Continued)

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 56h | 54h | JEDEC Sector Map Parameter Dword-4 Config. Detect-2 | 02h | Bits 31:0 = Sector map configuration detection command address = 00_00_00_02h: address of CR1NV |
| 57h | 55h | | 00h | |
| 58h | 56h | | 00h | |
| 59h | 57h | | 00h | |
| 5Ah | 58h | JEDEC Sector Map Parameter Dword-5 Config. Detect-3 | FDh | Bits 31:24 = Read data mask = 0000_0010b: Select bit 1 of the data byte for D8h_NV value<br>0= 64KB uniform sectors<br>1= 256KB uniform sectors<br>Bits 23:22 = Configuration detection command address length = 11b: Variable length<br>Bits 21:20 = RFU = 11b<br>Bits 19:16 = Configuration detection command latency = 1111b: variable latency<br>Bits 15:8 = Configuration detection instruction = 65h: Read any register<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Command Descriptor = 0<br>Bit 0 = The end descriptor = 1 |
| 5Bh | 59h | | 65h | |
| 5Ch | 5Ah | | FFh | |
| 5Dh | 5Bh | | 02h | |
| 5Eh | 5Ch | JEDEC Sector Map Parameter Dword-6 Config. Detect-3 | 04h | Bits 31:0 = Sector map configuration detection command address = 00_00_00_04h: address of CR3NV |
| 5Fh | 5Dh | | 00h | |
| 60h | 5Eh | | 00h | |
| 61h | 5Fh | | 00h | |
| 62h | 60h | JEDEC Sector Map Parameter Dword-7 Config-0 Header | FEh | Bits 31:24 = RFU = FFh<br>Bits 23:16 = Region count (Dwords -1) = 02h: Three regions<br>Bits 15:8 = Configuration ID = 00h: 4KB sectors at bottom with remainder 64KB sectors<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Map Descriptor = 1<br>Bit 0 = not the end descriptor = 0 |
| 63h | 61h | | 00h | |
| 64h | 62h | | 02h | |
| 65h | 63h | | FFh | |
| 66h | 64h | JEDEC Sector Map Parameter Dword-8 Config-0 Region-0 | F1h | Bits 31:8 = Region size = 00007Fh:<br>Region size as count-1 of 256 Byte units = 8 x 4KB sectors = 32KB<br>Count = 32KB/256 = 128, value = count -1 = 128 -1 = 127 = 7Fh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 0b<br>--- Erase Type 3 is 256KB erase and is not supported in the 4KB sector region<br>Bit 1 = Erase Type 2 support = 0b<br>--- Erase Type 2 is 64KB erase and is not supported in the 4KB sector region<br>Bit 0 = Erase Type 1 support = 1b<br>--- Erase Type 1 is 4KB erase and is supported in the 4KB sector region |
| 67h | 65h | | 7Fh | |
| 68h | 66h | | 00h | |
| 69h | 67h | | 00h | |
| 6Ah | 68h | JEDEC Sector Map Parameter Dword-9 Config-0 Region-1 | F2h | Bits 31:8 = Region size = 00007Fh:<br>Region size as count-1 of 256 Byte units = 1 x 32KB sectors = 32KB<br>Count = 32KB/256 = 128, value = count -1 = 128 -1 = 127 = 7Fh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 0b<br>--- Erase Type 3 is 256KB erase and is not supported in the 32KB sector region<br>Bit 1 = Erase Type 2 support = 1b<br>--- Erase Type 2 is 64KB erase and is supported in the 32KB sector region<br>Bit 0 = Erase Type 1 support = 0b<br>--- Erase Type 1 is 4KB erase and is not supported in the 32KB sector region |
| 6Bh | 69h | | 7Fh | |
| 6Ch | 6Ah | | 00h | |
| 6Dh | 6Bh | | 00h | |

**Table 74. CFI and SFDP Section 2, Sector Map Parameter Table** (Continued)

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 6Eh | 6Ch | JEDEC Sector Map Parameter Dword-10 Config-0 Region-2 | F2h | Bits 31:8 = 64Mb device Region size = 007EFFh:<br>Region size as count-1 of 256 Byte units = 127x 65536B sectors = 8323072B<br>Count = 8323072B/256 = 32512, value = count -1 = 32512-1 = 32511= 7EFFh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 0b<br>--- Erase Type 3 is 256KB erase and is not supported in the 64KB sector region<br>Bit 1 = Erase Type 2 support = 1b<br>--- Erase Type 2 is 64KB erase and is supported in the 64KB sector region<br>Bit 0 = Erase Type 1 support = 0b<br>--- Erase Type 1 is 4KB erase and is not supported in the 64KB sector region |
| 6Fh | 6Dh | | FFh | |
| 70h | 6Eh | | 7Eh (64 Mb) | |
| 71h | 6Fh | | 00h | |
| 72h | 70h | JEDEC Sector Map Parameter Dword-11 Config-2 Header | FEh | Bits 31:24 = RFU = FFh<br>Bits 23:16 = Region count (Dwords -1) = 02h: Three regions<br>Bits 15:8 = Configuration ID = 02h: 4KB sectors at top with remainder 64KB sectors<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Map Descriptor = 1<br>Bit 0 = not the end descriptor = 0 |
| 73h | 71h | | 02h | |
| 74h | 72h | | 02h | |
| 75h | 73h | | FFh | |
| 76h | 74h | JEDEC Sector Map Parameter Dword-12 Config-2 Region-0 | F2h | Bits 31:8 = 64Mb device Region size = 007EFFh:<br>Region size as count-1 of 256 Byte units = 127x 65536B sectors = 8323072B<br>Count = 8323072B/256 = 32512, value = count -1 = 32512-1 = 32511= 7EFFh<br>Bits 7:4 = RFU = Fh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 0b<br>--- Erase Type 3 is 256KB erase and is not supported in the 64KB sector region<br>Bit 1 = Erase Type 2 support = 1b<br>--- Erase Type 2 is 64KB erase and is supported in the 64KB sector region<br>Bit 0 = Erase Type 1 support = 0b<br>--- Erase Type 1 is 4KB erase and is not supported in the 64KB sector region |
| 77h | 75h | | FFh | |
| 78h | 76h | | 7Eh (64 Mb) | |
| 79h | 77h | | 00h | |
| 7Ah | 78h | JEDEC Sector Map Parameter Dword-13 Config-2 Region-1 | F2h | Bits 31:8 = Region size = 00007Fh:<br>Region size as count-1 of 256 Byte units = 1 x 32KB sectors = 32KB<br>Count = 32KB/256 = 128, value = count -1 = 128 -1 = 127 = 7Fh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 0b<br>--- Erase Type 3 is 256KB erase and is not supported in the 32KB sector region<br>Bit 1 = Erase Type 2 support = 1b<br>--- Erase Type 2 is 64KB erase and is supported in the 32KB sector region<br>Bit 0 = Erase Type 1 support = 0b<br>--- Erase Type 1 is 4KB erase and is not supported in the 32KB sector region |
| 7Bh | 79h | | 7Fh | |
| 7Ch | 7Ah | | 00h | |
| 7Dh | 7Bh | | 00h | |

**Table 74. CFI and SFDP Section 2, Sector Map Parameter Table** (Continued)

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 7Eh | 7C | JEDEC Sector Map Parameter Dword-14 Config-2 Region-2 | F1h | Bits 31:8 = Region size = 00007Fh:<br>Region size as count-1 of 256 Byte units = 8 x 4KB sectors = 32KB<br>Count = 32KB/256 = 128, value = count -1 = 128 -1 = 127 = 7Fh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 0b<br>--- Erase Type 3 is 256KB erase and is not supported in the 4KB sector region<br>Bit 1 = Erase Type 2 support = 0b<br>--- Erase Type 2 is 64KB erase and is not supported in the 4KB sector region<br>Bit 0 = Erase Type 1 support = 1b<br>--- Erase Type 1 is 4KB erase and is supported in the 4KB sector region |
| 7Fh | 7D | | 7Fh | |
| 80h | 7E | | 00h | |
| 81h | 7F | | 00h | |
| 82h | 80h | JEDEC Sector Map Parameter Dword-15 Config-1 Header | FEh | Bits 31:24 = RFU = FFh<br>Bits 23:16 = Region count (Dwords -1) = 02h: Three regions<br>Bits 15:8 = Configuration ID = 01h: 4KB sectors at bottom with remainder 256KB sectors<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Map Descriptor = 1<br>Bit 0 = not the end descriptor = 0 |
| 83h | 81h | | 01h | |
| 84h | 82h | | 02h | |
| 85h | 83h | | FFh | |
| 86h | 84h | JEDEC Sector Map Parameter Dword-16 Config-1 Region-0 | F1h | Bits 31:8 = Region size = 00007Fh:<br>Region size as count-1 of 256 Byte units = 8 x 4KB sectors = 32KB<br>Count = 32KB/256 = 128, value = count -1 = 128 -1 = 127 = 7Fh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 0b<br>--- Erase Type 3 is 256KB erase and is supported in the 4KB sector region<br>Bit 1 = Erase Type 2 support = 0b<br>--- Erase Type 2 is 64KB erase and is not supported in the 4KB sector region<br>Bit 0 = Erase Type 1 support = 1b<br>--- Erase Type 1 is 4KB erase and is supported in the 4KB sector region |
| 87h | 85h | | 7Fh | |
| 88h | 86h | | 00h | |
| 89h | 87h | | 00h | |
| 8Ah | 88h | JEDEC Sector Map Parameter Dword-17 Config-1 Region-1 | F4h | Bits 31:8 = Region size = 00037Fh:<br>Region size as count-1 of 256 Byte units = 1 x 224KB sectors = 224KB<br>Count = 224KB/256 = 896, value = count -1 = 896 -1 = 895 = 37Fh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 1b<br>--- Erase Type 3 is 256KB erase and is supported in the 224KB sector region<br>Bit 1 = Erase Type 2 support = 0b<br>--- Erase Type 2 is 64KB erase and is not supported in the 224KB sector region<br>Bit 0 = Erase Type 1 support = 0b<br>--- Erase Type 1 is 4KB erase and is not supported in the 224KB sector region |
| 8Bh | 89h | | 7Fh | |
| 8Ch | 8Ah | | 03h | |
| 8Dh | 8Bh | | 00h | |
| 8Eh | 8Ch | JEDEC Sector Map Parameter Dword-18 Config-1 Region-2 | F4h | Bits 31:8 = 64 Mb device Region size = 007BFFh:<br>Region size as count-1 of 256 Byte units = 31 x 262144B sectors = 8126464B<br>Count = 8126464B/256 = 31744, value = count -1 = 31744-1 = 31743= 7BFFh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 1b<br>--- Erase Type 3 is 256KB erase and is supported in the 256KB sector region<br>Bit 1 = Erase Type 2 support = 0b<br>--- Erase Type 2 is 64KB erase and is not supported in the 256KB sector region<br>Bit 0 = Erase Type 1 support = 0b<br>--- Erase Type 1 is 4KB erase and is not supported in the 256KB sector region |
| 8Fh | 8Dh | | FFh | |
| 90h | 8Eh | | 7Bh (64 Mb) | |
| 91h | 8F | | 00h | |

**Table 74. CFI and SFDP Section 2, Sector Map Parameter Table** (Continued)

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 92h | 90h | JEDEC Sector Map Parameter Dword-19 Config-3 Header | FEh | Bits 31:24 = RFU = FFh<br>Bits 23:16 = Region count (Dwords -1) = 02h: Three regions<br>Bits 15:8 = Configuration ID = 03h: 4KB sectors at top with remainder 256KB sectors<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Map Descriptor = 1<br>Bit 0 = not the end descriptor = 0 |
| 93h | 91h | | 03h | |
| 94h | 92h | | 02h | |
| 95h | 93h | | FFh | |
| 96h | 94h | JEDEC Sector Map Parameter Dword-20 Config-3 Region-0 | F4h | Bits 31:8 = 64 Mb device Region size = 007BFFh:<br>Region size as count-1 of 256 Byte units = 31 x 262144B sectors = 8126464B<br>Count = 8126464B/256 = 31744, value = count -1 = 31744-1 = 31743= 7BFFh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 1b<br>--- Erase Type 3 is 256KB erase and is supported in the 256KB sector region<br>Bit 1 = Erase Type 2 support = 0b<br>--- Erase Type 2 is 64KB erase and is not supported in the 256KB sector region<br>Bit 0 = Erase Type 1 support = 0b<br>--- Erase Type 1 is 4KB erase and is not supported in the 256KB sector region |
| 97h | 95h | | FFh | |
| 98h | 96h | | 7Bh (64 Mb) | |
| 99h | 97h | | 00h | |
| 9Ah | 98h | JEDEC Sector Map Parameter Dword-21 Config-3 Region-1 | F4h | Bits 31:8 = Region size = 00037Fh:<br>Region size as count-1 of 256 Byte units = 1 x 224KB sectors = 224KB<br>Count = 224KB/256 = 896, value = count -1 = 896 -1 = 895 = 37Fh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 1b<br>--- Erase Type 3 is 256KB erase and is supported in the 224KB sector region<br>Bit 1 = Erase Type 2 support = 0b<br>--- Erase Type 2 is 64KB erase and is not supported in the 224KB sector region<br>Bit 0 = Erase Type 1 support = 0b<br>--- Erase Type 1 is 4KB erase and is not supported in the 224KB sector region |
| 9Bh | 99h | | 7Fh | |
| 9Ch | 9Ah | | 03h | |
| 9Dh | 9Bh | | 00h | |
| 9Eh | 9Ch | JEDEC Sector Map Parameter Dword-22 Config-3 Region-2 | F1h | Bits 31:8 = Region size = 00007Fh:<br>Region size as count-1 of 256 Byte units = 8 x 4KB sectors = 32KB<br>Count = 32KB/256 = 128, value = count -1 = 128 -1 = 127 = 7Fh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 0b<br>--- Erase Type 3 is 256KB erase and is not supported in the 4KB sector region<br>Bit 1 = Erase Type 2 support = 0b<br>--- Erase Type 2 is 64KB erase and is not supported in the 4KB sector region<br>Bit 0 = Erase Type 1 support = 1b<br>--- Erase Type 1 is 4KB erase and is supported in the 4KB sector region |
| 9Fh | 9Dh | | 7Fh | |
| A0h | 9Eh | | 00h | |
| A1h | 9Fh | | 00h | |
| A2h | A0h | JEDEC Sector Map Parameter Dword-23 Config-4 Header | FEh | Bits 31:24 = RFU = FFh<br>Bits 23:16 = Region count (Dwords -1) = 00h: One region<br>Bits 15:8 = Configuration ID = 04h: Uniform 64KB sectors<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Map Descriptor = 1<br>Bit 0 = not the end descriptor = 0 |
| A3h | A1h | | 04h | |
| A4h | A2h | | 00h | |
| A5h | A3h | | FFh | |

**Table 74. CFI and SFDP Section 2, Sector Map Parameter Table** (Continued)

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| A6h | A4h | JEDEC Sector Map Parameter Dword-24 Config-4 Region-0 | F2h | Bits 31:8 = 64 Mb device Region size = 007FFBh:<br>Region size as count-1 of 256 Byte units = 128 x 65536B sectors = 8388608B<br>Count = 8388608B/256 = 32768, value = count -1 = 32768-1 = 32767= 7FFFh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 0b<br>--- Erase Type 3 is 256KB erase and is not supported in the 64KB sector region<br>Bit 1 = Erase Type 2 support = 1b<br>--- Erase Type 2 is 64KB erase and is supported in the 64KB sector region<br>Bit 0 = Erase Type 1 support = 0b<br>--- Erase Type 1 is 4KB erase and is not supported in the 64KB sector region |
| A7h | A5h | | FFh | |
| A8h | A6h | | 7Fh (64 Mb) | |
| A9h | A7h | | 00h | |
| AAh | A8h | JEDEC Sector Map Parameter Dword-25 Config-5 Header | FFh | Bits 31:24 = RFU = FFh<br>Bits 23:16 = Region count (Dwords -1) = 00h: One region<br>Bits 15:8 = Configuration ID = 05h: Uniform 256KB sectors<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Map Descriptor = 1<br>Bit 0 = The end descriptor = 1 |
| ABh | A9h | | 05h | |
| ACh | AAh | | 00h | |
| ADh | ABh | | FFh | |
| AEh | ACh | JEDEC Sector Map Parameter Dword-26 Config-5 Region-0 | F4h | Bits 31:8 = 64 Mb device Region size = 01FFFFh:<br>Region size as count-1 of 256 Byte units = 32 x 262144B sectors = 8388608B<br>Count = 8388608B/256 = 32768,value = count -1 = 32768-1 = 32767= 7FFFh<br>Bits 7:4 = RFU = Fh<br>Erase Type not supported = 0/ supported = 1<br>Bit 3 = Erase Type 4 support = 0b<br>--- Erase Type 4 is not defined<br>Bit 2 = Erase Type 3 support = 1b<br>--- Erase Type 3 is 256KB erase and is supported in the 256KB sector region<br>Bit 1 = Erase Type 2 support = 0b<br>--- Erase Type 2 is 64KB erase and is not supported in the 256KB sector region<br>Bit 0 = Erase Type 1 support = 0b<br>--- Erase Type 1 is 4KB erase and is not supported in the 256KB sector region |
| AFh | ADh | | FFh | |
| B0h | AEh | | 7Fh (64 Mb) | |
| B1h | AFh | | 00h | |

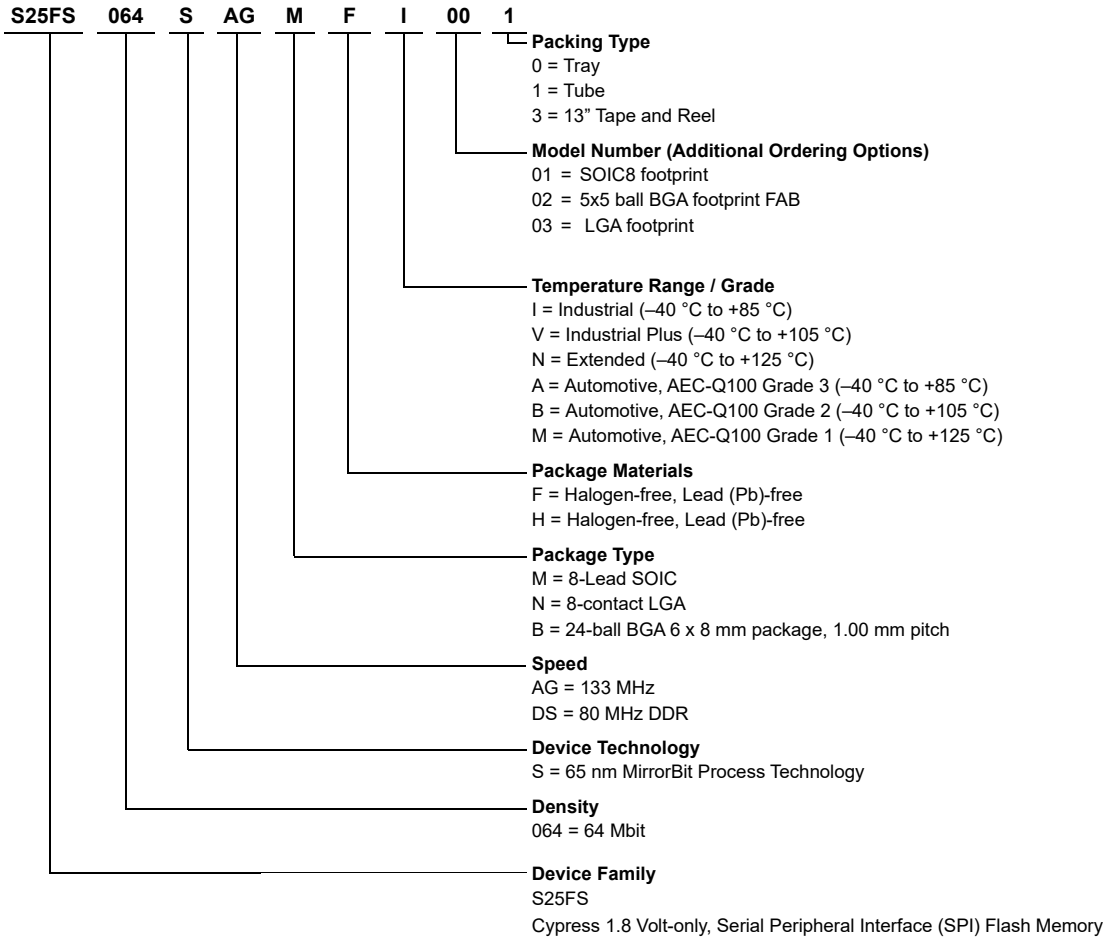
## 13.4 Initial Delivery State

The device is shipped from Cypress with non-volatile bits set as follows:

- The entire memory array is erased: i.e., all bits are set to 1 (each byte contains FFh).
- The OTP address space has the first 16 bytes programmed to a random number. All other bytes are erased to FFh.
- The SFDP address space contains the values as defined in the description of the SFDP address space.
- The ID-CFI address space contains the values as defined in the description of the ID-CFI address space.
- The RUID address space contains the 64bit Unique Id number.
- The Status Register 1 Non-volatile contains 00h (all SR1NV bits are cleared to 0's).
- The Configuration Register 1 Non-volatile contains 00h.
- The Configuration Register 2 Non-volatile contains 00h.
- The Configuration Register 3 Non-volatile contains 00h.
- The Configuration Register 4 Non-volatile contains 10h.
- The Password Register contains FFFFFFFF-FFFFFFFFh.
- All PPB bits are "1".
- The ASP Register bits are FFFFh.

# 14. Ordering Part Number

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




**Note**

98.Halogen free definition is in accordance with IE 61249-2-21 specification.

### Valid Combinations — Standard

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

**Table 75. S25FS064S Valid Combinations — Standard**

| Valid Combinations | | | | | |
|---|---|---|---|---|---|
| **Base Ordering Part Number** | **Speed Option** | **Package and Temperature** | **Model Number** | **Packing Type** | **Package Marking** |
| S25FS064S | AG | MFI, MFV, MFN | 01 | 0, 1, 3 | FS064S + A + (Temp) + F + 1 |
| | | NFI, NFV, NFN | 03 | 0, 1, 3 | FS064S + A + (Temp) + F + 3 |
| | | BHI, BHV, BHN | 02 | 0, 3 | FS064S + A + (Temp) + H + 2 |
| | DS | MFI, MFV, MFN | 01 | 0, 1, 3 | FS064S + D + (Temp) + F + 1 |
| | | NFI, NFV, NFN | 03 | 0, 1, 3 | FS064S + D + (Temp) + F + 3 |
| | | BHI, BHV, BHN | 02 | 0, 3 | FS064S + D + (Temp) + H + 2 |

### Valid Combinations — Automotive Grade / AEC-Q100

The table below 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. Consult 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 76. S25FS064S Valid Combinations — Automotive Grade / AEC-Q100**

| Valid Combinations — Automotive Grade / AEC-Q100 | | | | | |
|---|---|---|---|---|---|
| **Base Ordering Part Number** | **Speed Option** | **Package and Temperature** | **Model Number** | **Packing Type** | **Package Marking** |
| S25FS064S | AG | MFA, MFB MFM | 01 | 0, 1, 3 | FS064S + A + (Temp) + F + 1 |
| | | NFA, NFB, NFM | 03 | 0, 1, 3 | FS064S + A + (Temp) + F + 3 |
| | | BHA, BHB, BHM | 02 | 0, 3 | FS064S + A + (Temp) + H + 2 |
| | DS | MFA, MFB MFM | 01 | 0, 1, 3 | FS064S + D + (Temp) + F + 1 |
| | | NFA, NFB, NFM | 03 | 0, 1, 3 | FS064S + D + (Temp) + F + 3 |
| | | BHA, BHB, BHM | 02 | 0, 3 | FS064S + D + (Temp) + H + 2 |

## 15. Contacting Cypress

Obtain the latest list of company locations and contact information at: www.cypress.com/cypresslocations.

## 16. Glossary

- **BCD** = Binary Coded Decimal. A value in which each 4 bit nibble represents a decimal numeral.
- **Command** = All information transferred between the host system and memory during one period while CS# is low. This includes the instruction (sometimes called an operation code or opcode) and any required address, mode bits, latency cycles, or data.
- **DDP** = Dual Die Package = Two die stacked within the same package to increase the memory capacity of a single package. Often also referred to as a Multi-Chip Package (MCP)
- **DDR** = Double Data Rate = When input and output are latched on every edge of SCK.
- **ECC Unit** = 16 byte aligned and length data groups in the main Flash array and OTP array, each of which has its own hidden ECC to enable error correction on each group.
- **Flash** = the name for a type of Electrical Erase Programmable Read Only Memory (EEPROM) that erases large blocks of memory bits in parallel, making the erase operation much faster than early EEPROM.
- **High** = a signal voltage level ≥ $V_{IH}$ or a logic level representing a binary one (“1”).
- **Instruction** = the 8 bit code indicating the function to be performed by a command (sometimes called an operation code or opcode). The instruction is always the first 8 bits transferred from host system to the memory in any command.
- **Low** = a signal voltage level ≤ $V_{IL}$ or a logic level representing a binary zero (“0”).
- **LSB** = Least Significant Bit = Generally the right most bit, with the lowest order of magnitude value, within a group of bits of a register or data value.
- **MSB** = Most Significant Bit = Generally the left most bit, with the highest order of magnitude value, within a group of bits of a register or data value.
- **N/A** = Not Applicable. A value is not relevant to situation described.
- **Non-Volatile** = no power is needed to maintain data stored in the memory.
- **OPN** = Ordering Part Number = The alphanumeric string specifying the memory device type, density, package, factory non-volatile configuration, etc. used to select the desired device.
- **Page** = 512 Byte or 256 Byte aligned and length group of data.
- **PCB** - Printed Circuit Board
- Register Bit References = are in the format: Register_name[bit_number] or Register_name[bit_range_MSB: bit_range_LSB]
- **SDR** = Single Data Rate = When input is latched on the rising edge and output on the falling edge of SCK.
- **Sector** = erase unit size; depending on device model and sector location this may be 4KBytes, 64KBytes or 256KBytes
- **Write** = an operation that changes data within volatile or non-volatile registers bits or non-volatile Flash memory. When changing non-volatile data, an erase and reprogramming of any unchanged non-volatile data is done, as part of the operation, such that the non-volatile data is modified by the write operation, in the same way that volatile data is modified – as a single operation. The non-volatile data appears to the host system to be updated by the single write command, without the need for separate commands for erase and reprogram of adjacent, but unaffected data.

## 16. Document History Page

| Document Title: S25FS064S, 64 Mbit (8 Mbyte), 1.8 V FS-S Flash<br>Document Number: 002-03631 | | | | |
|---|---|---|---|---|
| **Rev.** | **ECN No.** | **Orig. of Change** | **Submission Date** | **Description of Change** |
| ** | 4905590 | BWHA | 10/05/2015 | Initial release |
| *A | 5090169 | RHLU/ BWHA | 01/29/2016 | Updated Table 1:<br>Updated details in both “FS-S” columns corresponding to “Auto Boot Mode” and “Updated Table 3:<br>Updated details in “SO / IO1” column corresponding to “Power-Off” Interface State.<br>Updated details in “IO3_RESET#” column corresponding to “Interface Standby” Interface State. Updated Table 7:<br>Referred Note (14) in description of $I_{CC1}$ parameter.<br>Updated Table 8:<br>Changed maximum value of $I_{DPD}$ parameter from 60 µA to 80 µA.<br>Updated Table 9:<br>Changed maximum value of $I_{DPD}$ parameter from 100 µA to 150 µA.<br>Updated Embedded Algorithm Performance Tables:<br>Changed maximum value of $t_{PP}$ parameter from 1080 µs to 2000 µs.<br>Updated Table 17: “Comments” column corresponding to $t_{RS}$ parameter.<br>Updated Address Space Maps:<br>Updated Configuration Register 1: Table 30 and Table 31:<br>Updated Configuration Register 2 Non-volatile (CR2NV): Table 33:<br>Updated Configuration Register 3: Table 36:<br>Updated ASP Register (ASPR): Table 41:<br>Updated Register Access Commands: Write VDLR (WVDLR 4Ah):<br>Updated Table 56: “Typical” with “Minimum” in column heading.<br>Updated Table 66: Details in “Data” column corresponding to “69h” Parameter Relative Byte Address Offset.<br>Updated Table 67: Details in “Data” column corresponding to “6Ch” Parameter Relative Byte Address Offset.<br>Updated Table 68: Details in “Data” column corresponding to “76h” Parameter Relative Byte Address Offset.<br>Updated Table 70: Details in “Data” column corresponding to “7Ch” Parameter Relative Byte Address Offset.<br>Updated Table 71: Details in “Parameter Relative Byte Address Offset” and “Data” columns. |
| *B | 5149633 | RHLU | 02/23/2016 | Updated description for “Dual or Quad I/O Read commands” and “Quad Double Data Rate read commands” in Read Flash Array.<br>Updated Maximum value of $I_{DPD}$ in Table 7 through Table 9.<br>Updated $t_{RPH}$ Time in Table 13.<br>Updated Sales page with latest copyright disclaimer and links. |
| *C | 5311335 | BWHA | 06/24/2016 | Updated Table 6: changed Min value for VCC (cut-off).<br>Addition of LGA Package, Figure 39 and Section 8.2.2.<br>Removed “Table Program and Erase Performance Extended Temperature.<br>Updated Table 16 Program and Erase Performance inclusive of all temperature ranges, Changing typical timing parameter $t_W$ and $t_{SE}$ (64KB or 4KB sectors) to 240ms, $t_{SE}$ (256KB sectors) to 960ms, $t_{BE}$ to 30s.<br>Updated Table 72 Header, change SFDP data Address 24h to “B1h”, Address 25h to “72h”, Address 26h to “1Dh”, Address 28h to “82h”, Address 2Bh to “C7”.<br>Updated Table 13 $t_{RPH}$ time from 45 µs to 35 µs.<br>Addition of ECC feature description Section 11.5.1.1.<br>Addition of Register Descriptions Table 26.<br>Updated Data Integrity Section 12.<br>Updated Power On Reset Section 6.3.1.<br>Added Thermal Resistance Section 5.3.<br>Added Link to Cypress Flash Roadmap Section 1.3.1.<br>Updated to new template. |

| Document Title: S25FS064S, 64 Mbit (8 Mbyte), 1.8 V FS-S Flash<br>Document Number: 002-03631 | | | | |
|---|---|---|---|---|
| **Rev.** | **ECN No.** | **Orig. of Change** | **Submission Date** | **Description of Change** |
| *D | 5563158 | BWHA / ECAO | 01/12/2017 | Changed status from Preliminary to Final.<br>Added “Automotive, AEC-Q100 Grade 3”, “Automotive, AEC-Q100 Grade 2”, “Automotive, AEC-Q100 Grade 1” Temperature Range related information in all instances across the document.<br>Updated Features.<br>Updated Performance Summary:<br>Updated Typical Program and Erase Rates.<br>Updated Typical Current Consumption, -40°C to +85°C.<br>Updated Electrical Specifications:<br>Updated Operating Ranges:<br>Updated Temperature Ranges.<br>Updated Embedded Algorithm Performance Tables:<br>Updated Table 16.<br>Updated Table 17.<br>Updated Physical Interface:<br>Updated Physical Diagrams:<br>Updated SOIC 8-Lead, 208 mil Body Width (SOC008).<br>Updated LGA 8-contact 5 x 6 mm (W9A008).<br>Updated Ball Grid Array 24-ball 6 x 8 mm (FAB024).<br>Updated Data Integrity:<br>Updated Erase Endurance.<br>Updated Data Retention.<br>Updated Ordering Part Number:<br>Added Valid Combinations — Automotive Grade / AEC-Q100.<br>Updated to new template. |
| *E | 5959879 | BWHA | 11/08/2017 | Updated Table 22 on page 46.<br>Changed $V_{DD}$ to $V_{CC}$.<br>Corrected Figure 81 on page 97.<br>Updated Ordering Part Number on page 141 definition of letters in OPN indicating package material.<br>Updated DDR Data Valid Timing Using DLP on page 38, Example. |
| *F | 6125350 | BWHA | 04/04/2018 | Added Table 10, Valid Enter DPD mode and Release from DPD Mode Sequence on page 29 |

# 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

Forums | WICED IOT Forums | Projects | Video | Blogs | Training | Components

### Technical Support

cypress.com/support

© Cypress Semiconductor Corporation, 2015-2018. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("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 does not assume any liability arising out of any security breach, such as unauthorized access to or use of a Cypress product. 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. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, 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 or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

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.