**S25FL127S**

# 128-Mbit (16 Mbyte) 3.0 V SPI Flash Memory

## Features

- CMOS 3.0 Volt Core
- Density
  - 128 Mbits (16 Mbytes)
- Serial Peripheral Interface (SPI) with Multi-I/O
  - SPI Clock polarity and phase modes 0 and 3
  - Extended Addressing: 24- or 32-bit address options
  - Serial Command set and footprint compatible with S25FL-A, S25FL-K, and S25FL-P SPI families
  - Multi I/O Command set and footprint compatible with S25FL-P SPI family
- READ Commands
  - Normal, Fast, Dual, Quad
  - AutoBoot - power up or reset and execute a Normal or Quad read command automatically at a preselected address
  - Common Flash Interface (CFI) data for configuration information.
- Programming (0.8 Mbytes/s)
  - 256- or 512-byte Page Programming buffer options
  - Quad-Input Page Programming (QPP) for slow clock systems
  - Automatic ECC -internal hardware Error Correction Code generation with single bit error correction
- Erase (0.5 Mbytes/s)
  - Hybrid sector size option - physical set of sixteen 4-kbyte sectors at top or bottom of address space with all remaining sectors of 64 kbytes
  - Uniform sector option - always erase 256-kbyte blocks for software compatibility with higher density and future devices.
- Cycling Endurance
  - 100,000 Program-Erase Cycles per sector, 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
- Cypress® 65 nm MirrorBit Technology with Eclipse™ Architecture
- Supply Voltage: 2.7V to 3.6V
- Temperature Range:
  - Industrial (-40°C to +85°C)
  - Industrial Plus (-40°C to +105°C)
  - Automotive AEC-Q100 Grade 3 (-40°C to +85°C)
  - Automotive AEC-Q100 Grade 2 (-40°C to +105°C)
- Packages (all Pb-free)
  - 8-lead SOIC (208 mil)
  - 16-lead SOIC (300 mil)
  - 8-contact WSON 6 x 5 mm
  - BGA-24 6 x 8 mm
    - 5 x 5 ball (FAB024) and 4 x 6 ball (FAC024) footprint options
  - Known Good Die and Known Tested Die

# Performance Summary

### Maximum Read Rates

| Command | Clock Rate (MHz) | Mbytes/s |
|---|---|---|
| Read | 50 | 6.25 |
| Fast Read | 108 | 13.5 |
| Dual Read | 108 | 27 |
| Quad Read | 108 | 54 |

### Typical Program and Erase Rates

| Operation | kbytes/s |
|---|---|
| Page Programming (256-byte page buffer) | 650 |
| Page Programming (512-byte page buffer) | 800 |
| 4-kbyte Physical Sector Erase (Hybrid Sector Option) | 30 |
| 64-kbyte Physical Sector Erase (Hybrid Sector Option) | 500 |
| 256-kbyte Logical Sector Erase (Uniform Sector Option) | 500 |

### Current Consumption

| Operation | Current (mA) |
|---|---|
| Serial Read 50 MHz | 16 (max) |
| Serial Read 108 MHz | 24 (max) |
| Quad Read 108 MHz | 47 (max) |
| Program | 50 (max) |
| Erase | 50 (max) |
| Standby | 0.07 (typ) |

# Contents

**Performance Summary** ..... 3

**1. Overview** ..... 4
1.1 General Description ..... 4
1.2 Migration Notes ..... 5
1.3 Glossary ..... 7
1.4 Other Resources ..... 8

**2. Signal Descriptions** ..... 9
2.1 Input/Output Summary ..... 9
2.2 Address and Data Configuration ..... 10
2.3 Hardware Reset (RESET#) ..... 10
2.4 Serial Clock (SCK) ..... 10
2.5 Chip Select (CS#) ..... 10
2.6 Serial Input (SI) / IO0 ..... 11
2.7 Serial Output (SO) / IO1 ..... 11
2.8 Write Protect (WP#) / IO2 ..... 11
2.9 Hold (HOLD#) / IO3 / RESET# ..... 11
2.10 Voltage Supply ($V_{CC}$) ..... 12
2.11 Supply and Signal Ground ($V_{SS}$) ..... 12
2.12 Not Connected (NC) ..... 12
2.13 Reserved for Future Use (RFU) ..... 12
2.14 Do Not Use (DNU) ..... 13
2.15 Block Diagrams ..... 13

**3. Signal Protocols** ..... 15
3.1 SPI Clock Modes ..... 15
3.2 Command Protocol ..... 15
3.3 Interface States ..... 19
3.4 Configuration Register Effects on the Interface ..... 24
3.5 Data Protection ..... 24

**4. Electrical Specifications** ..... 25
4.1 Absolute Maximum Ratings ..... 25
4.2 Operating Ranges ..... 25
4.3 Power-Up and Power-Down ..... 26
4.4 DC Characteristics ..... 28

**5. Timing Specifications** ..... 29
5.1 Key to Switching Waveforms ..... 29
5.2 AC Test Conditions ..... 30
5.3 Reset ..... 31
5.4 AC Characteristics ..... 34

**6. Physical Interface** ..... 37
6.1 SOIC 8-Lead Package ..... 37
6.2 SOIC 16-Lead Package ..... 39
6.3 WSON 6 x 5 Package ..... 41
6.4 FAB024 24-Ball BGA Package ..... 43
6.5 FAC024 24-Ball BGA Package ..... 45

**7. Address Space Maps** ..... 47
7.1 Overview ..... 47
7.2 Flash Memory Array ..... 47
7.3 ID-CFI Address Space ..... 48
7.4 JEDEC JESD216B Serial Flash Discoverable Parameters (SFDP) Space ..... 48
7.5 OTP Address Space ..... 49
7.6 Registers ..... 50

**8. Data Protection** ..... 59
8.1 Secure Silicon Region (OTP) ..... 59
8.2 Write Enable Command ..... 59
8.3 Block Protection ..... 60
8.4 Advanced Sector Protection ..... 61

**9. Commands** ..... 66
9.1 Command Set Summary ..... 67
9.2 Identification Commands ..... 72
9.3 Register Access Commands ..... 75
9.4 Read Memory Array Commands ..... 85
9.5 Program Flash Array Commands ..... 93
9.6 Erase Flash Array Commands ..... 97
9.7 One Time Program Array Commands ..... 102
9.8 Advanced Sector Protection Commands ..... 103
9.9 Reset Commands ..... 109
9.10 Embedded Algorithm Performance Tables ..... 110

**10. Data Integrity** ..... 112
10.1 Erase Endurance ..... 112
10.2 Data Retention ..... 112

**11. Software Interface Reference** ..... 113
11.1 Command Summary ..... 113

**12. Serial Flash Discoverable Parameters (SFDP) Address Map** ..... 115
12.1 SFDP Header Field Definitions ..... 116
12.2 Device ID and Common Flash Interface (ID-CFI) Address Map ..... 118
12.3 Device ID and Common Flash Interface (ID-CFI) ASO Map — Automotive Only ..... 133
12.4 Registers ..... 134
12.5 Initial Delivery State ..... 137

**13. Ordering Information** ..... 138

**14. Revision History** ..... 140

**Sales, Solutions, and Legal Information** ..... **142**
Worldwide Sales and Design Support ..... 142
Products ..... 142
PSoC® Solutions ..... 142
Cypress Developer Community ..... 142
Technical Support ..... 142

# 1. Overview

## 1.1 General Description

The Cypress S25FL127S device is a flash non-volatile memory product 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

This device 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 (Quad I/O or QIO) serial commands. This multiple width interface is called SPI Multi-I/O or MIO.

The Eclipse architecture features a Page Programming Buffer that allows up to 128 words (256 bytes) or 256 words (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 FL-S devices at the higher clock rates supported, with QIO command, the instruction read transfer rate can match or exceed traditional parallel interface, asynchronous, NOR flash memories while reducing signal count dramatically.

The S25FL127S product offers a high density coupled with the flexibility and fast performance required by a variety of embedded applications. It is ideal for code shadowing, XIP, and data storage.

## 1.2 Migration Notes

### 1.2.1 Features Comparison

The S25FL127S device is command set and footprint compatible with prior generation FL-K, FL-P, and FL-S family devices.

**Table 1. FL Generations Comparison**

| Parameter | FL-K | FL-P | FL-S | FL127S |
|---|---|---|---|---|
| Technology Node | 90 nm | 90 nm | 65 nm | 65 nm |
| Architecture | Floating Gate | MirrorBit | MirrorBit Eclipse | MirrorBit Eclipse |
| Density | 4 Mb - 128 Mb | 32 Mb - 256 Mb | 128 Mb, 256 Mb, 512 Mb, 1 Gb | 128 Mb |
| Bus Width | x1, x2, x4 | x1, x2, x4 | x1, x2, x4 | x1, x2, x4 |
| Supply Voltage | 2.7V - 3.6V | 2.7V - 3.6V | 2.7V - 3.6V / 1.65V - 3.6V $V_{IO}$ | 2.7V - 3.6V |
| Normal Read Speed (SDR) | 6 MB/s (50 MHz) | 5 MB/s (40 MHz) | 6 MB/s (50 MHz) | 6 MB/s (50 MHz) |
| Fast Read Speed (SDR) | 13 MB/s (104 MHz) | 13 MB/s (104 MHz) | 17 MB/s (133 MHz) | 13.5 MB/s (108 MHz) |
| Dual Read Speed (SDR) | 26 MB/s (104 MHz) | 20 MB/s (80 MHz) | 26 MB/s (104 MHz) | 27 MB/s (108 MHz) |
| Quad Read Speed (SDR) | 52 MB/s (104 MHz) | 40 MB/s (80 MHz) | 52 MB/s (104 MHz) | 54 MB/s (108 MHz) |
| Fast Read Speed (DDR) | - | - | 16 MB/s (66 MHz) | - |
| Dual Read Speed (DDR) | - | - | 33 MB/s (66 MHz) | - |
| Quad Read Speed (DDR) | - | - | 66 MB/s (66 MHz) | - |
| Program Buffer Size | 256B | 256B | 256B / 512B | 256B / 512B |
| Uniform Sector Size | 4 kB | 64 kB / 256 kB | 64 kB / 256 kB | 64 kB / 256 kB |
| Parameter Sector Size | N/A | 4 kB | 4 kB (option) | 4 kB (option) |
| Number of Parameter Sector | 0 | 32 | 32 (option) | 16 (option) |
| Sector Erase Rate (typ.) | 135 kB/s (4 kB), 435 kB/s (64 kB) | 130 kB/s (64 kB) | 30 kB/s (4 kB), 500 kB/s (64 kB / 256 kB) | 30 kB/s (4 kB), 500 kB/s (64 kB / 256 kB) |
| Page Programming Rate (typ.) | 365 kB/s (256B) | 170 kB/s (256B) | 1000 kB/s (256B), 1500 kB/s (512B) | 650 kB/s (256B), 800 kB/s (512B) |
| OTP | 768B (3 x 256B) | 506B | 1024B | 1024B |
| Advanced Sector Protection | No | No | Yes | Yes |
| Auto Boot Mode | No | No | Yes | Yes |
| Erase Suspend/Resume | Yes | No | Yes | Yes |
| Program Suspend/ Resume | Yes | No | Yes | Yes |
| Operating Temperature | -40°C to +85°C | -40°C to +85°C / +105°C | -40°C to +85°C / +105°C | -40°C to +85°C / +105°C |

***Notes:***

1. *256B program page option only for 128 Mb and 256-Mb density FL-S devices.*
2. *FL-P column indicates FL129P MIO SPI device (for 128-Mb density). FL128P does not support MIO, OTP or 4-kB sectors.*
3. *64-kB sector erase option only for 128-Mb/256-Mb density FL-P and FL-S devices.*
4. *FL-K family devices can erase 4-kB sectors in groups of 32 kB or 64 kB.*
5. *Refer to individual data sheets for further details.*

## 1.2.2 Known Differences from Prior Generations

### 1.2.2.1 Error Reporting

Prior generation FL memories either do not have error status bits or do not set them if program or erase is attempted on a protected sector. The FL-S family does 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 either case the program or erase operation did not complete as requested by the command.

### 1.2.2.2 Secure Silicon Region (OTP)

The size and format (address map) of the One Time Program area is different from prior generations. The method for protecting each portion of the OTP area is different. For additional details see Secure Silicon Region (OTP) on page 59.

### 1.2.2.3 Configuration Register Freeze Bit

The configuration register Freeze bit CR1[0], locks the state of the Block Protection bits as in prior generations. In the FL-S family it also locks the state of the configuration register TBPARM bit CR1[2], TBPROT bit CR1[5], and the Secure Silicon Region (OTP) area.

### 1.2.2.4 Sector Architecture

The FL127S has sixteen 4-kbyte sectors that may be located at the top or bottom of address space. Other members of the FL-S Family and FL-P Family have thirty two 4-kbyte sectors that may be located at the top or bottom of address space.

These smaller parameter sectors may also be removed, leaving all sectors uniform in size, depending on the selected configuration (SR2[7]).

### 1.2.2.5 Sector Erase Commands

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

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

The erase command for 64-kbyte sectors is supported when the configuration option for 4-kbyte parameter sectors with 64-kbyte uniform sectors is used. The 64-kbyte erase command may be applied to erase a group of sixteen 4-kbyte sectors.

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

### 1.2.2.6 Deep Power Down

The Deep Power Down (DPD) function is not supported in FL-S family devices.

The legacy DPD (B9h) command code is instead used to enable legacy SPI memory controllers, that can issue the former DPD command, to access a new bank address register. The bank address register allows SPI memory controllers that do not support more than 24 bits of address, the ability to provide higher order address bits for commands, as needed to access the larger address space of the 256-Mbit and 512-Mbit density FL-S devices. For additional information see Extended Address on page 47.

### 1.2.2.7 Hardware Reset

A separate hardware reset input is provided in packages with greater than 8 connections. In 8-connection packages, a new option is provided to replace the HOLD# / IO3 input with an IO3 / RESET# input to allow for hardware reset in small packages.

#### 1.2.2.8 New Features

The FL-S family introduces several new features to SPI category memories:

- Extended address for access to higher memory density.
- AutoBoot for simpler access to boot code following power up.
- Enhanced high performance read commands using mode bits to eliminate the overhead of SIO instructions when repeating the same type of read command.
- Multiple options for initial read latency (number of dummy cycles) for faster initial access time or higher clock rate read commands.
- Automatic ECC for enhanced data integrity.
- Advanced Sector Protection for individually controlling the protection of each sector. This is very similar to the Advanced Sector Protection feature found in several other Cypress parallel interface NOR memory families.

## 1.3 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. |
| **ECC** | 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 syndrome 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. |
| **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 bytes or 256 bytes 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] |
| **Sector** | Erase unit size; depending on device model and sector location this may be 4 kbytes, 64 kbytes or 256 kbytes. |
| **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. |

## 1.4 Other Resources

### 1.4.1 Links to Software

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

### 1.4.2 Links to Application Notes

www.cypress.com/appnotes

### 1.4.3 Specification Bulletins

Specification bulletins provide information on temporary differences in feature description or parametric variance since the publication of the last full data sheet. Contact your local sales office for details. Obtain the latest list of company locations and contact information at: www.cypress.com

# Hardware Interface

**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 S25FL-S family of devices reduces the number of signals for connection to the host system by serially transferring all control, address, and data information over 4 to 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 S25FL-S family of devices 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.

## 2. Signal Descriptions

### 2.1 Input/Output Summary

**Table 2. Signal List**

| Signal Name | Type | Description |
|---|---|---|
| RESET# | Input | **Hardware Reset.** The signal has an internal pull-up resistor and should 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. IO2 in Quad mode. The signal has an internal pull-up resistor and may be left unconnected in the host system if not used for Quad commands. |
| HOLD# / IO3 or IO3 / RESET# | I/O | **Hold** (pause) serial transfer in single bit or Dual data commands. IO3 in Quad-I/O mode. RESET# when enabled by SR2[5]=1 and not in Quad-I/O mode, CR1[1]=0. or when CS# is high. The signal has an internal pull-up resistor and may be left unconnected in the host system if not used for Quad commands. |
| $V_{CC}$ | 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. |

## 2.2 Address and Data Configuration

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

Dual or Quad Output commands send information from the host to the memory only on the SI signal. Data will be returned to the host as a sequence of bit pairs on IO0 and IO1 or four bit (nibble) groups on IO0, IO1, IO2, and IO3.

Dual or Quad Input/Output (I/O) commands send information 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.

## 2.3 Hardware Reset (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:

- terminates any operation in progress,
- tristates all outputs,
- resets the volatile bits in the Configuration Register,
- resets the volatile bits in the Status Registers,
- resets the Bank Address Register to 0,
- loads the Program Buffer with all 1s,
- reloads all internal configuration information necessary to bring the device to standby mode,
- and resets the internal Control Unit to standby state.

RESET# causes the same initialization process as is performed when power comes up and requires $t_{PU}$ time.

RESET# may be asserted low at any time. To ensure data integrity any operation that was interrupted by a hardware reset should be reinitiated once the device is ready to accept a command sequence.

When RESET# is first asserted Low, the device draws $I_{CC1}$ (50 MHz value) during $t_{PU}$. If RESET# continues to be held at $V_{SS}$ the device draws CMOS standby current ($I_{SB}$).

RESET# has an internal pull-up resistor and should be left unconnected in the host system if not used.

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, inside the package.

## 2.4 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.

## 2.5 Chip Select (CS#)

The chip select signal indicates when a command for the device is in process 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 are ignored and all output signals are high impedance. Unless an internal Program, Erase or Write Registers (WRR) embedded operation is in progress, the device will be in the Standby Power mode. Driving the CS# input to 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.

## 2.6 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).

## 2.7 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.

## 2.8 Write Protect (WP#) / IO2

When WP# is driven Low ($V_{IL}$), during a WRR command and while the Status Register Write Disable (SRWD) bit of the Status Register is set to a 1, it is not possible to write to the Status and Configuration Registers. This prevents any alteration of the Block Protect (BP2, BP1, BP0) and TBPROT bits of the Status Register. As a consequence, all the data bytes in the memory area that are protected by the Block Protect and TBPROT bits, are also hardware protected against data modification if WP# is Low during a WRR command.

The WP# function is not available when the Quad mode is enabled (CR[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).

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

## 2.9 Hold (HOLD#) / IO3 / RESET#

The Hold (HOLD#) signal is used to pause any serial communications with the device without deselecting the device or stopping the serial clock. The HOLD# input and function is available when enabled by a configuration bit SR2[5] =0.

To enter the Hold condition, the device must be selected by driving the CS# input to the logic low state. It is recommended that the user keep the CS# input low state during the entire duration of the Hold condition. This is to ensure that the state of the interface logic remains unchanged from the moment of entering the Hold condition. If the CS# input is driven to the logic high state while the device is in the Hold condition, the interface logic of the device will be reset. To restart communication with the device, it is necessary to drive HOLD# to the logic high state while driving the CS# signal into the logic low state. This prevents the device from going back into the Hold condition.

The Hold condition starts on the falling edge of the Hold (HOLD#) signal, provided that this coincides with SCK being at the logic low state. If the falling edge does not coincide with the SCK signal being at the logic low state, the Hold condition starts whenever the SCK signal reaches the logic low state. Taking the HOLD# signal to the logic low state does not terminate any Write, Program or Erase operation that is currently in progress.

During the Hold condition, SO is in high impedance and both the SI and SCK input are Don't Care.

The Hold condition ends on the rising edge of the Hold (HOLD#) signal, provided that this coincides with the SCK signal being at the logic low state. If the rising edge does not coincide with the SCK signal being at the logic low state, the Hold condition ends whenever the SCK signal reaches the logic low state.

The HOLD# function is not available when the Quad mode is enabled (CR1[1] =1). The Hold function is replaced by IO3 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.

A configuration bit SR2[5] may be set to 1 to replace the HOLD# / IO3 functions with the IO3 / RESET# functions. Then the IO3 / RESET# may be used to initiate the hardware reset function. The IO3 / RESET# input is only treated as RESET# when the device is not in Quad-I/O mode, CR1[1] = 0, or when CS# is high.

When Quad I/O mode is in use, CR1[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.

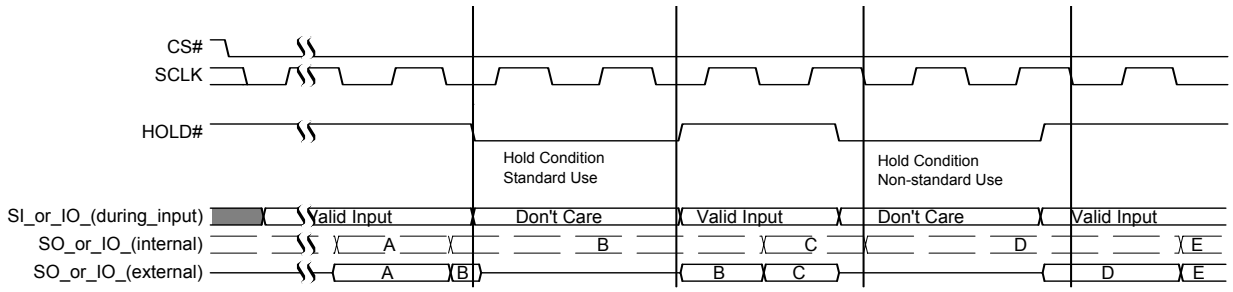
When the system enters a reset condition, the CS# signal must be driven high as part of the reset process and the IO3 / RESET# signal is driven low. When CS# goes high the IO3 / RESET# input transitions from being IO3 to being the RESET# input. The reset condition is then detected when CS# remains high and the IO3 / RESET# signal remains low for $t_{RP}$.

The HOLD#/IO3 or IO3/RESET# signals have an internal pull-up resistor and may be left unconnected in the host system if not used for Quad mode or the reset function.

When Quad mode is enabled, IO3 / RESET# is ignored for $t_{CS}$ following CS# going high. This allows some time for the memory or host system to actively drive IO3 / RESET# to a valid level following the end of a transfer. Following the end of a Quad I/O read the memory will actively drive IO3 high before disabling the output during $t_{DIS}$. Following a transfer in which IO3 was used to transfer data to the memory, e.g. the QPP command, the host system is responsible for driving IO3 high before disabling the host IO3 output. 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 $t_{RP}$. Once IO3 / RESET# is high the memory or host system can stop driving the signal. The integrated pull-up on IO3 will then hold IO3 high unless the host system actively drives IO3 / RESET# to initiate a reset.

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 not selected memory sharing the same IO3 / RESET# signal. See Section 5.3.3 IO3 / RESET# Input Initiated Hardware (Warm) Reset on page 33 for the IO3 / RESET timing.

**Figure 1. HOLD Mode Operation**




## 2.10 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. The voltage may vary from 2.7V to 3.6V.

## 2.11 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.

## 2.12 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). However, any signal connected to an NC must not have voltage levels higher than $V_{CC}$.

## 2.13 Reserved for Future Use (RFU)

No device internal signal is currently connected to the package connector but is there 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.

## 2.14 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.

## 2.15 Block Diagrams

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

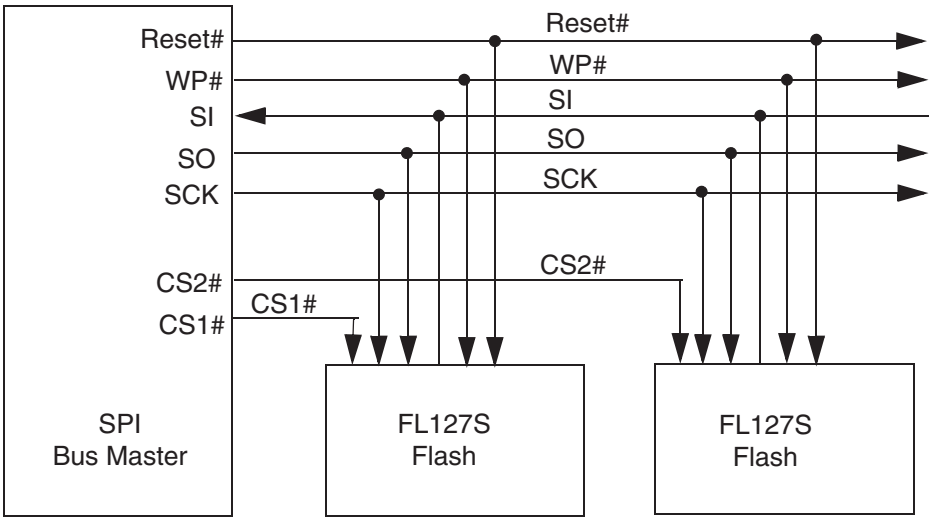



**Figure 3. Bus Master and Memory Devices on the SPI Bus - Dual Bit Data Path**

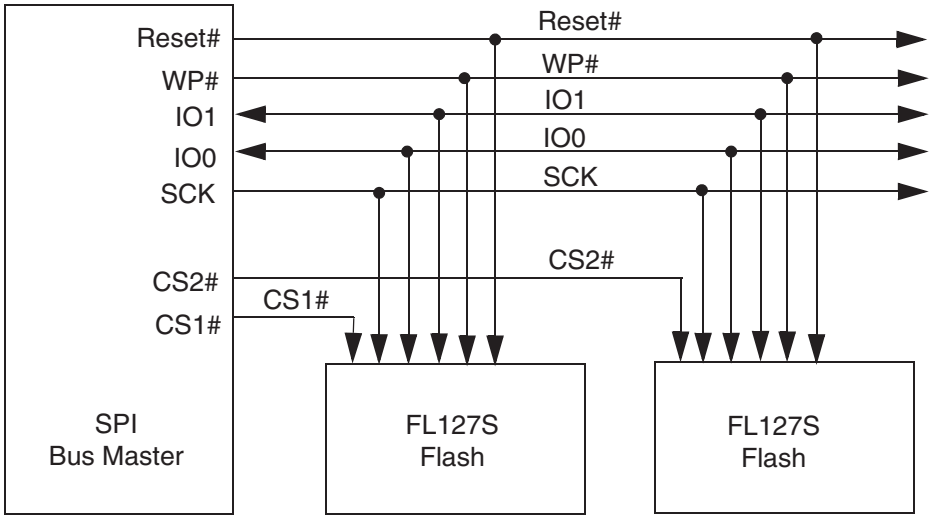
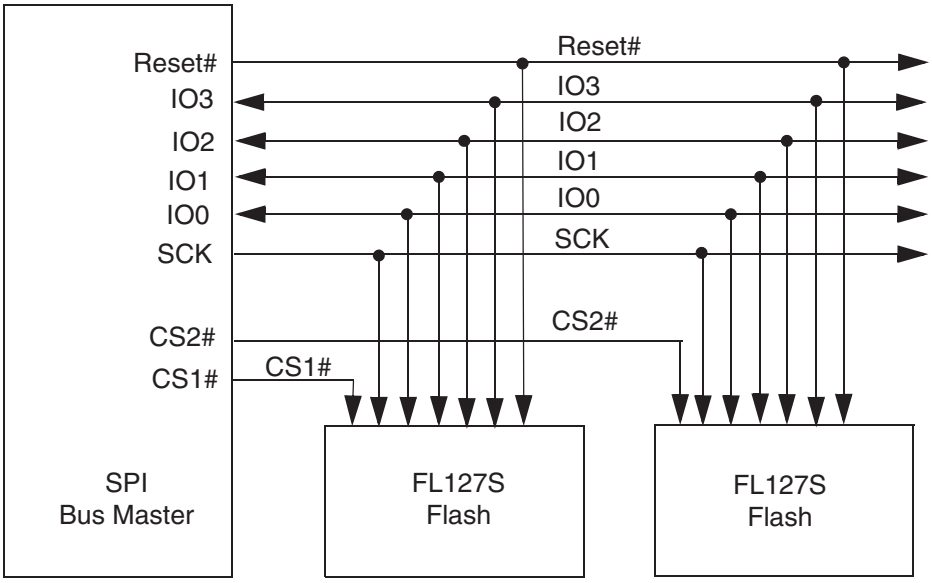


**Figure 4. Bus Master and Memory Devices on the SPI Bus - Quad Bit Data Path**



# 3. Signal Protocols

## 3.1 SPI Clock Modes

### 3.1.1 Single Data Rate (SDR)

The S25FL-S family of devices can be driven by an embedded microcontroller (bus master) in either of the two following clocking modes.

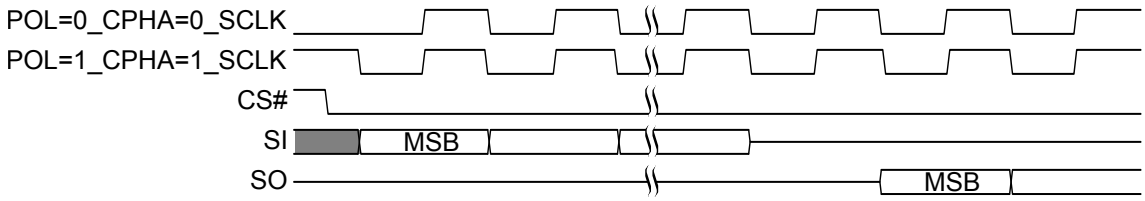
- **Mode 0** with Clock Polarity (CPOL) = 0 and, Clock Phase (CPHA) = 0
- **Mode 3** with CPOL = 1 and, CPHA = 1

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 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.

## 3.2 Command Protocol

All communication between the host system and S25FL-S family of 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 serially between the host system and memory device.

All instructions are transferred from host to memory as a single bit serial sequence on the SI signal.

Single bit wide commands may provide an address or data sent only on the SI signal. Data may be sent back to the host serially on the SO signal.

Dual or Quad Output commands provide an address sent to the memory only on the SI signal. Data will be returned to the host as a sequence of bit pairs on IO0 and IO1 or four bit (nibble) groups on IO0, IO1, IO2, and IO3.

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.

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 is always presented only as a single bit serial sequence on the Serial Input (SI) signal with one bit transferred to the memory device on each SCK rising edge. The instruction selects the type of information transfer or device operation to be performed.
- The instruction may be stand alone or may be followed by address bits to select a byte 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 address. The address transfers occur on SCK rising edge.
- 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.
- Some 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 mode bit transfers occur on SCK rising edge.
- 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.
- 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.
- 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 clock cycles after CS# signal was driven low is an exact multiple of eight cycles. If the CS# signal does not go high exactly at the eight SCK cycle 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.

### 3.2.1 Command Sequence Examples

**Figure 6. Stand Alone Instruction Command**

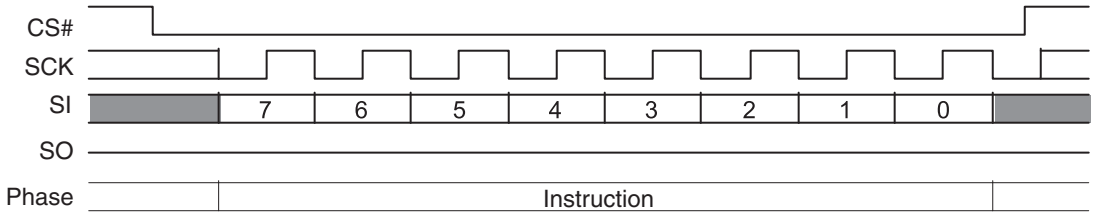



**Figure 7. Single Bit Wide Input Command**

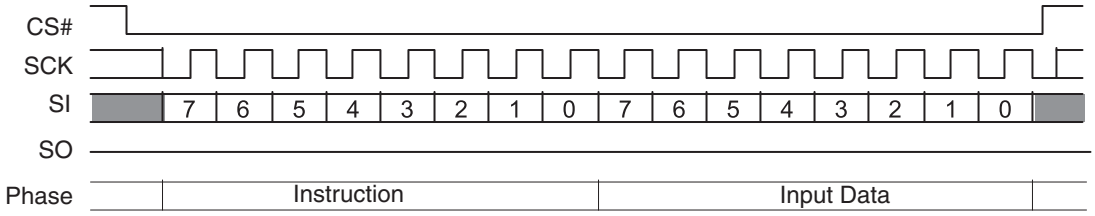



**Figure 8. Single Bit Wide Output Command**

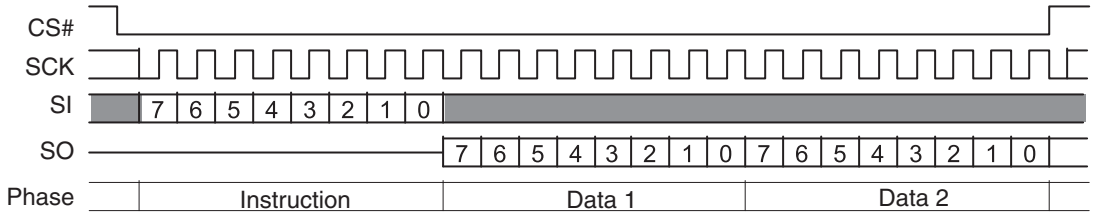



**Figure 9. Single Bit Wide I/O Command without Latency**

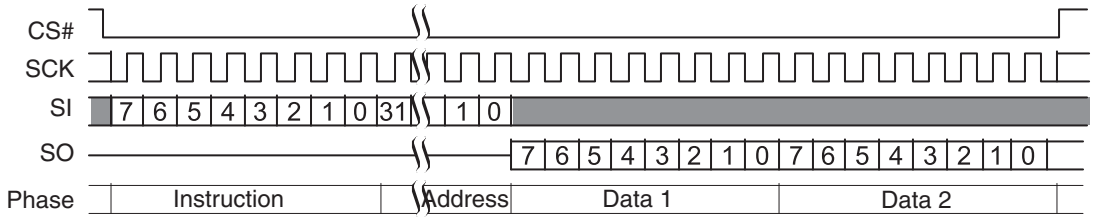



**Figure 10. Single Bit Wide I/O Command with Latency**

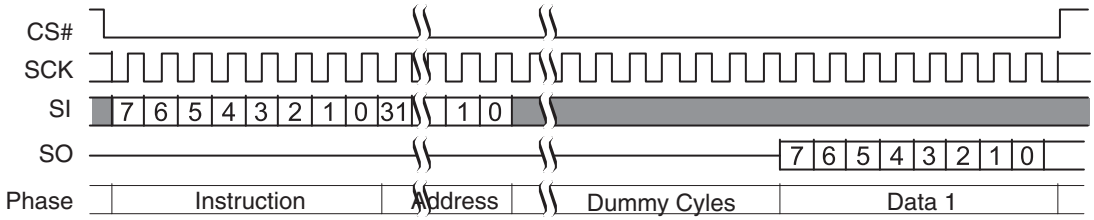


**Figure 11. Dual Output Command**

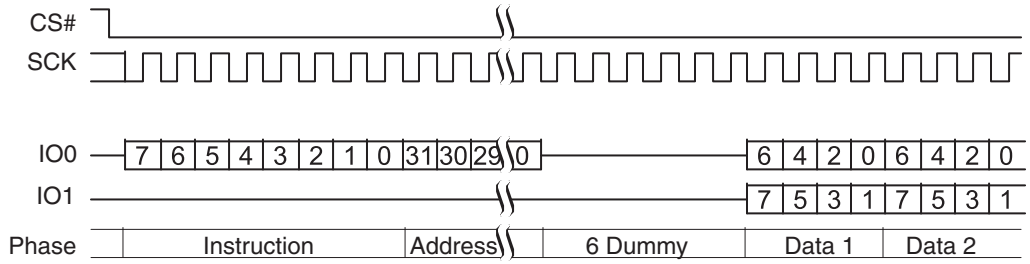



**Figure 12. Quad Output Command without Latency**

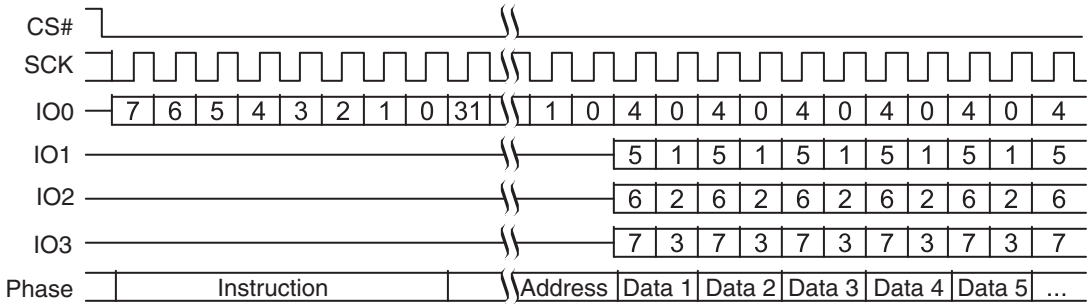



**Figure 13. Dual I/O Command**

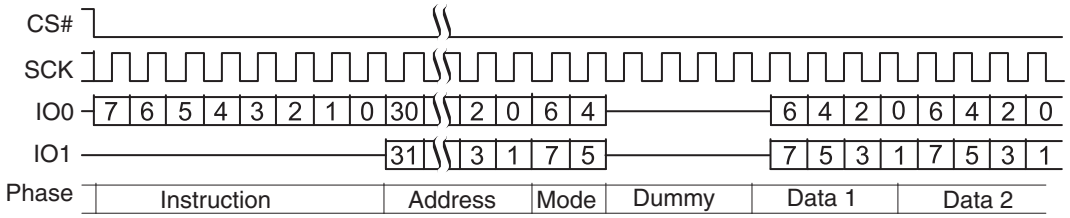
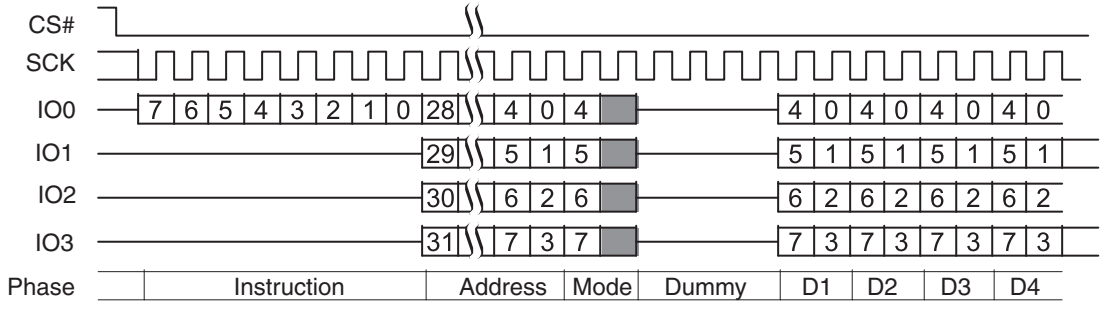



**Figure 14. Quad I/O Command**




Additional sequence diagrams, specific to each command, are provided in Section 9. Commands on page 66.

## 3.3 Interface States

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

**Table 3. Interface States Summary with Separate Reset**

| Interface State | $V_{CC}$ | RESET# | SCK | CS# | HOLD# / IO3 | WP# / IO2 | SO / IO1 | SI / IO0 |
|---|---|---|---|---|---|---|---|---|
| Power-Off | <$V_{CC}$ (low) | X | X | X | X | X | Z | X |
| Low Power Hardware Data Protection | <$V_{CC}$ (cut-off) | X | X | X | X | X | Z | X |
| Power-On (Cold) Reset | ≥$V_{CC}$ (min) | X | X | HH | X | X | Z | X |
| Hardware (Warm) Reset | ≥$V_{CC}$ (min) | HL | X | X | X | X | Z | X |
| Interface Standby | ≥$V_{CC}$ (min) | HH | X | HH | X | X | Z | X |
| Instruction Cycle | ≥$V_{CC}$ (min) | HH | HT | HL | HH | HV | Z | HV |
| Hold Cycle | ≥$V_{CC}$ (min) | HH | HV or HT | HL | HL | X | X | X |
| Single Input Cycle Host to Memory Transfer | ≥$V_{CC}$ (min) | HH | HT | HL | HH | X | Z | HV |
| Single Latency (Dummy) Cycle | ≥$V_{CC}$ (min) | HH | HT | HL | HH | X | Z | X |
| Single Output Cycle Memory to Host Transfer | ≥$V_{CC}$ (min) | HH | HT | HL | HH | X | MV | X |
| Dual Input Cycle Host to Memory Transfer | ≥$V_{CC}$ (min) | HH | HT | HL | HH | X | HV | HV |
| Dual Latency (Dummy) Cycle | ≥$V_{CC}$ (min) | HH | HT | HL | HH | X | X | X |
| Dual Output Cycle Memory to Host Transfer | ≥$V_{CC}$ (min) | HH | HT | HL | HH | X | MV | MV |
| QPP Address Input Cycle Host to Memory Transfer | ≥$V_{CC}$ (min) | HH | HT | HL | X | X | X | HV |
| Quad Input Cycle Host to Memory Transfer | ≥$V_{CC}$ (min) | HH | HT | HL | HV | HV | HV | HV |
| Quad Latency (Dummy) Cycle | ≥$V_{CC}$ (min) | HH | HT | HL | X | X | X | X |
| Quad Output Cycle Memory to Host Transfer | ≥$V_{CC}$ (min) | HH | HT | HL | 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*

**Table 4. Interface States Summary with IO3 / RESET# Enabled**

| Interface State | $V_{CC}$ | SCK | CS# | HOLD# / IO3 | WP# / IO2 | SO / IO1 | SI / IO0 |
|---|---|---|---|---|---|---|---|
| Power-Off | $<V_{CC}$ (low) | X | X | X | X | Z | X |
| Low Power Hardware Data Protection | $<V_{CC}$ (cut-off) | X | X | X | X | Z | X |
| Power-On (Cold) Reset | $\geq V_{CC}$ (min) | X | HH | X | X | Z | X |
| Hardware (Warm) Reset — Non-Quad Mode | $\geq V_{CC}$ (min) | X | X | HL | X | Z | X |
| Hardware (Warm) Reset — Quad Mode | $\geq V_{CC}$ (min) | X | HH | HL | X | Z | X |
| Interface Standby | $\geq V_{CC}$ (min) | X | HH | X | X | Z | X |
| Instruction Cycle (Legacy SPI) | $\geq V_{CC}$ (min) | HT | HL | HH | HV | Z | HV |
| Single Input Cycle Host to Memory Transfer | $\geq V_{CC}$ (min) | HT | HL | HH | X | Z | HV |
| Single Latency (Dummy) Cycle | $\geq V_{CC}$ (min) | HT | HL | HH | X | Z | X |
| Single Output Cycle Memory to Host Transfer | $\geq V_{CC}$ (min) | HT | HL | HH | X | MV | X |
| Dual Input Cycle Host to Memory Transfer | $\geq V_{CC}$ (min) | HT | HL | HH | X | HV | HV |
| Dual Latency (Dummy) Cycle | $\geq V_{CC}$ (min) | HT | HL | HH | X | X | X |
| Dual Output Cycle Memory to Host Transfer | $\geq V_{CC}$ (min) | HT | HL | HH | X | MV | MV |
| QPP Address Input Cycle Host to Memory Transfer | $\geq V_{CC}$ (min) | HT | HL | X | X | X | HV |
| Quad Input Cycle Host to Memory Transfer | $\geq V_{CC}$ (min) | HT | HL | HV | HV | HV | HV |
| Quad Latency (Dummy) Cycle | $\geq V_{CC}$ (min) | HT | HL | X | X | X | X |
| Quad Output Cycle Memory to Host Transfer | $\geq V_{CC}$ (min) | HT | HL | 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*

**Table 5. Interface States Summary with HOLD# / IO3 Enabled**

| Interface State | $V_{DD}$ | SCK | CS# | HOLD# / IO3 | WP# / IO2 | SO / IO1 | SI / IO0 |
|---|---|---|---|---|---|---|---|
| Power-Off | <$V_{CC}$ (low) | X | X | X | X | Z | X |
| Low Power Hardware Data Protection | <$V_{CC}$ (cut-off) | X | X | X | X | Z | X |
| Power-On (Cold) Reset | ≥$V_{CC}$ (min) | X | HH | X | X | Z | X |
| Interface Standby | ≥$V_{CC}$ (min) | X | HH | X | X | Z | X |
| Instruction Cycle (Legacy SPI) | ≥$V_{CC}$ (min) | HT | HL | HH | HV | Z | HV |
| Hold Cycle | ≥$V_{CC}$ (min) | HV or HT | HL | HL | X | X | X |
| Single Input Cycle Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | X | Z | HV |
| Single Latency (Dummy) Cycle | ≥$V_{CC}$ (min) | HT | HL | HH | X | Z | X |
| Single Output Cycle Memory to Host Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | X | MV | X |
| Dual Input Cycle Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | X | HV | HV |
| Dual Latency (Dummy) Cycle | ≥$V_{CC}$ (min) | HT | HL | HH | X | X | X |
| Dual Output Cycle Memory to Host Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | X | MV | MV |
| QPP Address Input Cycle Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | X | X | X | HV |
| Quad Input Cycle Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | HV | HV | HV | HV |
| Quad Latency (Dummy) Cycle | ≥$V_{CC}$ (min) | HT | HL | X | X | X | X |
| Quad Output Cycle Memory to Host Transfer | ≥$V_{CC}$ (min) | HT | HL | 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*

### 3.3.1 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.

### 3.3.2 Low Power Hardware Data Protection

When $V_{CC}$ is less than $V_{CC}$ (cut-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.

### 3.3.3 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.

### 3.3.4 Hardware (Warm) Reset

Some of the device package options provide a RESET# input. When 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 Separate RESET# Input Initiated Hardware (Warm) Reset on page 32.

A configuration option is provided to allow IO3 to be used as a hardware reset input when the device is not in Quad mode or when it is in Quad mode and CS# is high. When IO3 / 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 Section 5.3 Reset on page 31.

### 3.3.5 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.

### 3.3.6 Instruction Cycle

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 RESET# high, CS# low, HOLD# high, and drives Write Protect (WP#) signal as needed for the instruction. However, WP# is only relevant during instruction cycles of a WRR command and is otherwise ignored.

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 output, Quad output, Dual I/O, or 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.

### 3.3.7 Hold (HOLD# / IO3 selected by SR2[5])

When Quad mode is not enabled (CR[1]=0) the HOLD# / IO3 signal is used as the HOLD# input. The host keeps RESET# high, HOLD# low, SCK may be at a valid level or continue toggling, and CS# is low. When HOLD# is low a command is paused, as though SCK were held low. SI / IO0 and SO / IO1 ignore the input level when acting as inputs and are high impedance when acting as outputs during hold state. Whether these signals are input or output depends on the command and the point in the command sequence when HOLD# is asserted low.

When HOLD# returns high the next state is the same state the interface was in just before HOLD# was asserted low.

When Quad mode is enabled the HOLD# / IO3 signal is used as IO3.

### 3.3.8 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 dual output, and quad output commands send address to the memory using only SI but return read data using the I/O signals. The host keeps RESET# high, CS# low, HOLD# high, 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.

### 3.3.9 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 (CR[7:6]). During the latency cycles, the host keeps RESET# high, CS# low, and HOLD# high. 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 / I/O0 or other I/O signals during the latency cycles. In dual or quad read commands, the host must stop driving the I/O signals on the falling edge at the end of the last latency cycle. It is recommended that the host stop driving I/O signals during 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 Serial Output (SO) 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.

### 3.3.10 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# high, CS# low, and HOLD# high. 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.

### 3.3.11 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# high, CS# low, HOLD# high. 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.

### 3.3.12 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 (CR[7:6]). During the latency cycles, the host keeps RESET# high, CS# low, and HOLD# high. 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 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.

### 3.3.13 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# high, CS# low, and HOLD# high. The Write Protect (WP#) signal is ignored. The memory drives data on the SI / IO0 and SO / IO1 signals during the dual output cycles.

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

### 3.3.14 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 because the device must be in Quad mode for these commands thus the Hold and Write Protect features are not active. The host keeps 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.

### 3.3.15 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. The Quad Page Program command transfers four data bits to the memory in each cycle. The host keeps RESET# high, 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 Quad Page Program the host returns CS# high following the delivery of data to be programmed and the interface returns to standby state.

### 3.3.16 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 (CR[7:6]). During the latency cycles, the host keeps RESET# high, CS# low. 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.

### 3.3.17 Quad Output Cycle - Memory to Host Transfer

The Quad Output Read and Quad I/O Read return data to the host four bits in each cycle. The host keeps RESET# high, and 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.

## 3.4 Configuration Register Effects on the Interface

The configuration register bits 7 and 6 (CR1[7:6]) select the latency code for all read commands. The latency code selects the number of mode bit and latency cycles for each type of instruction.

The configuration register bit 1 (CR1[1]) selects whether Quad mode is enabled to ignore HOLD# and WP# and allow Quad Page Program, Quad Output Read, and Quad I/O Read commands.

## 3.5 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 Section, *Software Interface on page 47*.

### 3.5.1 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.

### 3.5.2 Low Power

When $V_{CC}$ is less than $V_{CC}$ (cut-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.

### 3.5.3 Clock Pulse Count

The device verifies that all program, erase, and Write Registers (WRR) commands consist of a clock pulse count that is a multiple of eight before executing them. A command not having a multiple of 8 clock pulse count is ignored and no error status is set for the command.

# 4. Electrical Specifications

## 4.1 Absolute Maximum Ratings

**Table 6. Absolute Maximum Ratings**

| | |
|---|---|
| Storage Temperature Plastic Packages | -65°C to +150°C |
| Ambient Temperature with Power Applied | -65°C to +125°C |
| $V_{CC}$ | -0.5V to +4.0V |
| Input voltage with respect to Ground ($V_{SS}$) (Note 1) | -0.5V to +($V_{CC}$ + 0.5V) |
| Output Short Circuit Current (Note 2) | 100 mA |

***Notes:***

1. *See Input Signal Overshoot on page 25 for allowed maximums during signal transition.*
2. *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.*
3. *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.*

## 4.2 Operating Ranges

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

### 4.2.1 Temperature Ranges

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

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.

### 4.2.2 Power Supply Voltage

$V_{CC}$ 2.7V to 3.6V

### 4.2.3 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 $V_{SS}$ to –2.0V or overshoot to $V_{CC}$ +2.0V, for periods up to 20 ns.

**Figure 15. Maximum Negative Overshoot Waveform**

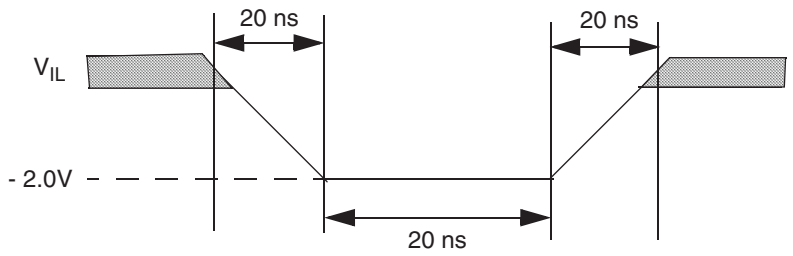
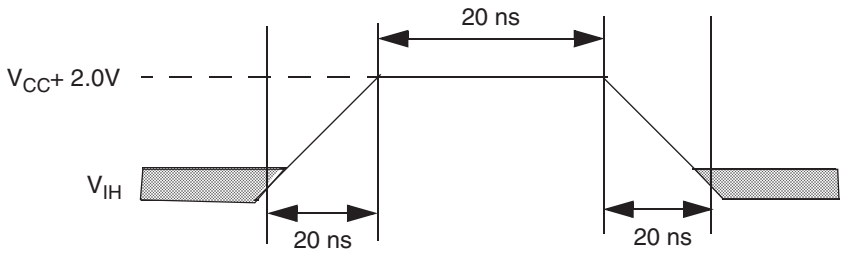


**Figure 16. Maximum Positive Overshoot Waveform**




## 4.3 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 (generally of the order of 100 kΩ) 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 17, Power-up on page 27. 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 18, Power-down and Voltage Drop on page 27. 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 0.1 µf).

**Table 7. Power-up/Power-down Voltage and Timing**

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $V_{CC}$ (min) | $V_{CC}$ (minimum operation voltage) | 2.7 | | V |
| $V_{CC}$ (cut-off) | $V_{CC}$ (Cut 0ff where re-initialization is needed) | 2.4 | | V |
| $V_{CC}$ (low) | $V_{CC}$ (low voltage for initialization to occur) | 1.0 | | V |
| $t_{PU}$ | $V_{CC}$ (min) to Read operation | | 300 | µs |
| $t_{PD}$ | $V_{CC}$ (low) time | 1.0 | | µs |

**Figure 17. Power-up**

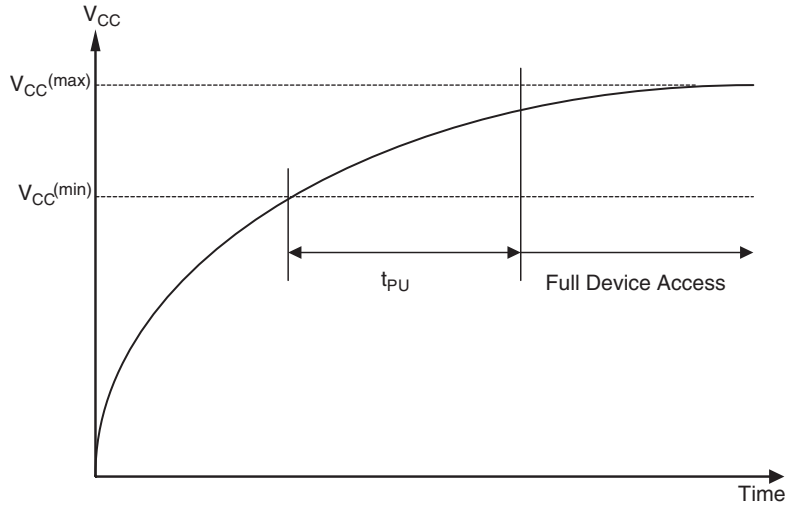
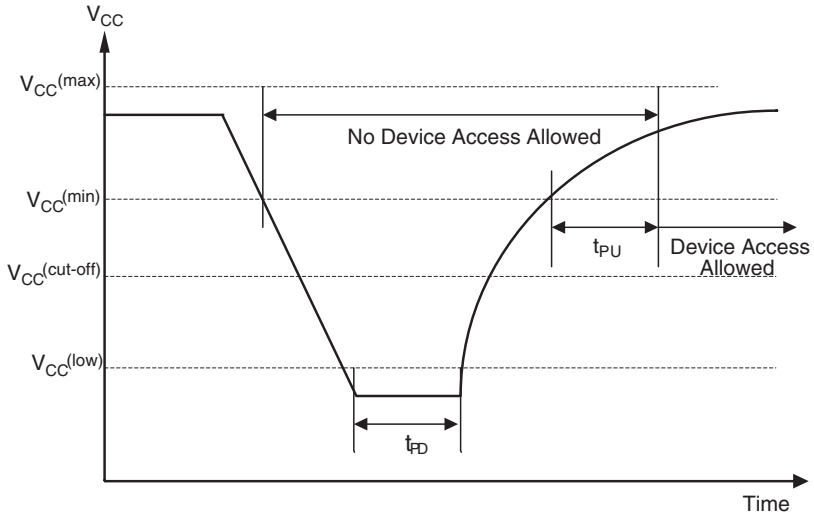



**Figure 18. Power-down and Voltage Drop**



## 4.4 DC Characteristics

Applicable within operating ranges.

**Table 8. DC Characteristics**

| Symbol | Parameter | Test Conditions | Min | Typ (1) | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input Low Voltage | | -0.5 | | 0.2 x $V_{CC}$ | V |
| $V_{IH}$ | Input High Voltage | | 0.7 x $V_{CC}$ | | $V_{CC}$ + 0.4 | V |
| $V_{OL}$ | Output Low Voltage | $I_{OL}$ = 1.6 mA, $V_{CC}$ = $V_{CC}$ min | | | 0.15 x $V_{CC}$ | V |
| $V_{OH}$ | Output High Voltage | $I_{OH}$ = –0.1 mA | 0.85 x $V_{CC}$ | | | V |
| $I_{LI}$ (Industrial) | Input Leakage Current | $V_{CC}$ = $V_{CC}$ Max, $V_{IN}$ = 0 to $V_{IL}$ Max or $V_{IH}$, CS# = $V_{IH}$ | | | ± 2 | µA |
| $I_{LO}$ (Industrial) | Output Leakage Current | $V_{CC}$ = $V_{CC}$ Max, $V_{IN}$ = $V_{IH}$ or $V_{IL}$ | | | ± 2 | µA |
| $I_{LI}$ (Industrial Plus) | Input Leakage Current | $V_{CC}$ = $V_{CC}$ Max, $V_{IN}$= 0 to $V_{IL}$ Max or $V_{IH}$, CS# = $V_{IH}$ | | | ± 4 | µA |
| $I_{LO}$ (Industrial Plus) | Output Leakage Current | $V_{CC}$ = $V_{CC}$ Max, $V_{IN}$ = $V_{IH}$ or $V_{IL}$ | | | ± 4 | µA |
| $I_{CC1}$ | Active Power Supply Current (READ) | Serial @50 MHz<br>Serial @108 MHz<br>Quad @108 MHz<br>Outputs unconnected during read data return (2) | | | 16<br>24<br>47 | mA |
| $I_{CC2}$ | Active Power Supply Current (Page Program) | CS# = $V_{CC}$ | | | 50 | mA |
| $I_{CC3}$ | Active Power Supply Current (WRR) | CS# = $V_{CC}$ | | | 50 | mA |
| $I_{CC4}$ | Active Power Supply Current (SE) | CS# = $V_{CC}$ | | | 50 | mA |
| $I_{CC5}$ | Active Power Supply Current (BE) | CS# = $V_{CC}$ | | | 50 | mA |
| $I_{SB}$ (Industrial) | Standby Current | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$, Industrial Temp | | 70 | 100 | µA |
| $I_{SB}$ (Industrial Plus) | Standby Current | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$, Industrial Plus Temp | | 70 | 300 | µA |
| $I_{POR}$ | Power On Reset Current | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$ | | | 63 | mA |

***Notes:***

1. *Typical values are at $T_{AI}$ = 25°C and $V_{CC}$ = 3V.*
2. *Output switching current is not included.*

### 4.4.1 Active Power and Standby Power Modes

The device is enabled and in the Active Power mode when Chip Select (CS#) is Low. When CS# is high, the device is disabled, but may still be in an Active Power mode until all program, erase, and write operations have completed. The device then goes into the Standby Power mode, and power consumption drops to $I_{SB}$.

# 5. Timing Specifications

## 5.1 Key to Switching Waveforms

**Figure 19. Waveform Element Meanings**

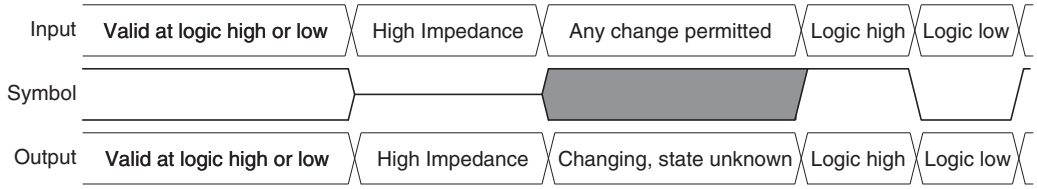
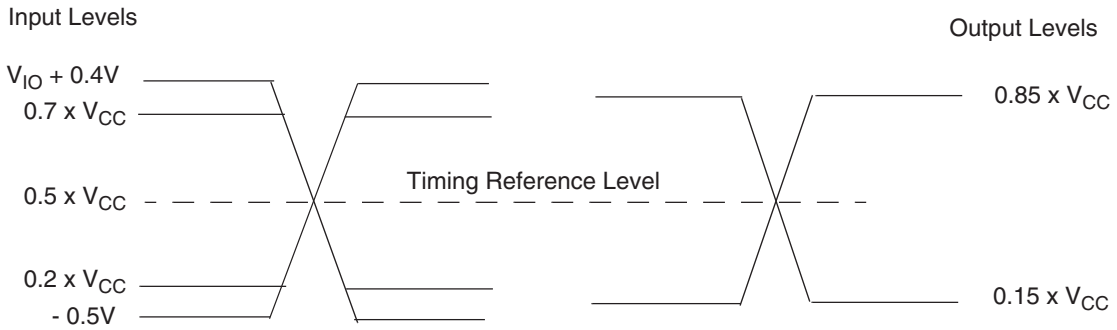



**Figure 20. Input, Output, and Timing Reference Levels**



## 5.2 AC Test Conditions

**Figure 21. Test Setup**

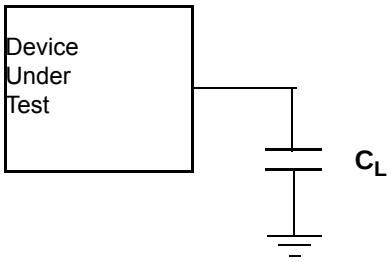



**Table 9. AC Measurement Conditions**

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $C_L$ | Load Capacitance | 30 | | pF |
| | Input Rise and Fall Times | | 2.4 | ns |
| | Input Pulse Voltage | 0.2 x $V_{CC}$ to 0.8 $V_{CC}$ | | V |
| | Input Timing Ref Voltage | 0.5 $V_{CC}$ | | V |
| | Output Timing Ref Voltage | 0.5 $V_{CC}$ | | V |

***Notes:***
1. *Output High-Z is defined as the point where data is no longer driven.*
2. *Input slew rate: 1.5 V/ns.*
3. *AC characteristics tables assume clock and data signals have the same slew rate (slope).*

### 5.2.1 Capacitance Characteristics

**Table 10. Capacitance**

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

***Note:***
1. *Parameter values are not 100% tested. For more information on capacitance, please consult the IBIS models.*

## 5.3 Reset

### 5.3.1 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 17 on page 27, Table 7 on page 26, and Figure 22 on page 31. 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 $t_{PU}$.

The IO3 / RESET# signal functions as the RESET# input when CS# is high for more than $t_{RP}$ time or when Quad Mode is not enabled CR1V[1]=0.

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

**Figure 22. Reset Low at the End of POR**

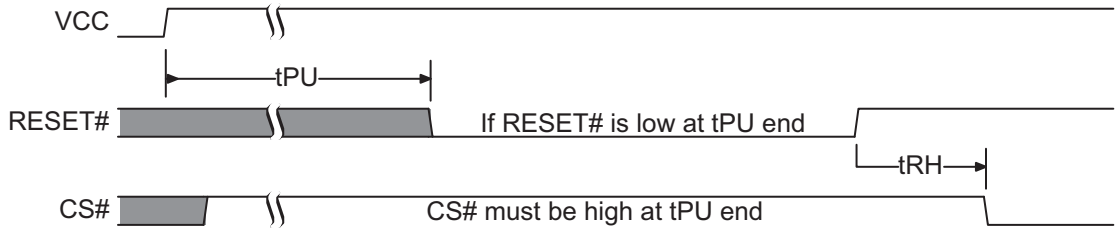



**Figure 23. Reset High at the End of POR**

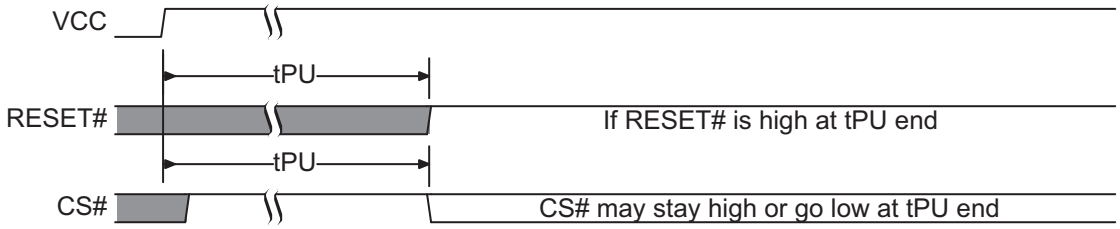
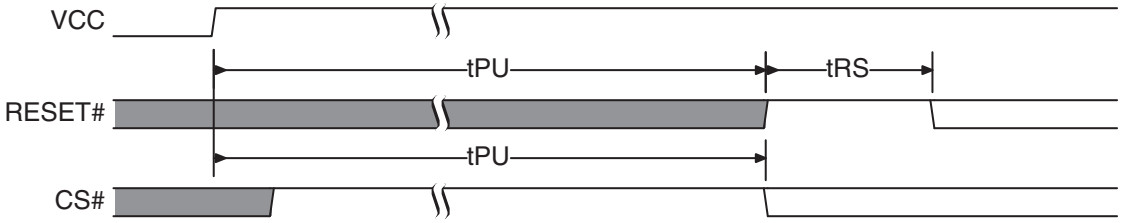



**Figure 24. POR followed by Hardware Reset**



### 5.3.2 Separate RESET# Input Initiated Hardware (Warm) Reset

When the RESET# input 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. 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.

A separate RESET# input is available only in the SOIC16 and BGA package options. The RESET# input has an internal pull-up to $V_{CC}$ and should be left unconnected if not used. The RESET command is independent of the state of RESET#. If RESET# is high or unconnected, and the RESET instruction is issued, the device will perform software reset.

The RESET# input provides a hardware method of resetting the flash memory device to standby state.

- RESET# must be high for $t_{RS}$ following $t_{PU}$ or $t_{RPH}$, before going low again to initiate a hardware reset.
- When RESET# is driven low for at least a minimum period of time ($t_{RP}$), 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 CS# is low at the time RESET# is asserted, CS# must return high during $t_{RPH}$ before it can be asserted low again after $t_{RH}$.

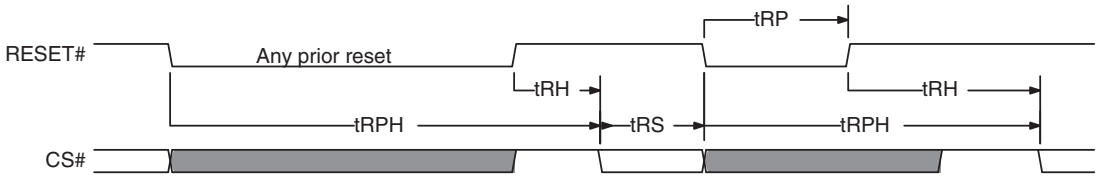
**Table 11. Hardware Reset Parameters**

| 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 |
| $t_{RP}$ | RESET# Pulse Width (only when AutoBoot enabled) | Max | 5 | µs |
| $t_{RH}$ | Reset Hold - RESET# high before CS# low | Min | 50 | ns |

***Notes:***

1. *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.*
2. *Sum of $t_{RP}$ and $t_{RH}$ must be equal to or greater than $t_{RPH}$.*

**Figure 25. Separate RESET# Input Initiated Hardware Reset**



### 5.3.3 IO3 / RESET# Input Initiated Hardware (Warm) Reset

The IO3 / RESET# signal functions as a RESET# input when enabled by SR2[5]=1 and CS# is high for more than $t_{CS}$ time or when Quad Mode is not enabled (CR1V[1]=0). The IO3 RESET# input provides a hardware method of resetting the flash memory device to standby state. The IO3 / RESET# input has an internal pull-up to $V_{CC}$ and may be left unconnected if Quad mode is not used.

When the IO3 / RESET# feature and Quad mode are both enabled, IO3 / RESET# is ignored for $t_{CS}$ following CS# going high, to avoid an unintended Reset operation. This allows some time for the memory or host system to actively drive IO3 / RESET# to a valid level following the end of a transfer. Following the end of a Quad I/O read the memory will actively drive IO3 high before disabling the output during $t_{DIS}$. Following a transfer in which IO3 was used to transfer data to the memory, e.g. the QPP command, the host system is responsible for driving IO3 high before disabling the host IO3 output. The integrated pull-up on IO3 will then hold IO3 until the host system actively drives IO3 / RESET# to initiate a reset. If CS# is driven low to start a new command, IO3 / RESET# is used as IO3.

When the device is not in quad mode or when CS# is high, and the IO3 / RESET# transitions from $V_{IH}$ to $V_{IL}$ for > $t_{RP}$, the device terminates any operation in progress, makes all outputs high impedance, ignores all read/write commands and resets the interface to standby state. The hardware reset process requires a period of $t_{RPH}$ to complete. During $t_{RPH}$, 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. If the POR process did not complete correctly for any reason during power-up ($t_{PU}$), RESET# going low for $t_{RP}$ will initiate the full POR process instead of the hardware reset process and will require $t_{PU}$ to complete the POR process. IO3 / RESET# must be high for $t_{RS}$ following $t_{PU}$ or $t_{RPH}$, before going low again to initiate a hardware reset.

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 $t_{RH}$.

The RESET command is independent of the state of RESET#. If IO3 / RESET# is high or unconnected, and the RESET instruction is issued, the device will perform software reset.

**Table 12. 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 |
| $t_{RP}$ | RESET# Pulse Width (only when AutoBoot enabled) | Max | 5 | µs |
| $t_{RH}$ | Reset Hold - RESET# high before CS# low | Min | 50 | ns |

***Notes:***

1. *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.*
2. *If Quad mode is enabled, IO3 / RESET# Low is ignored during $t_{CS}$.*
3. *Sum of $t_{RP}$ and $t_{RH}$ must be equal to or greater than $t_{RPH}$.*

**Figure 26. Hardware Reset when Quad Mode is not enabled and IO3 / Reset# is Enabled**

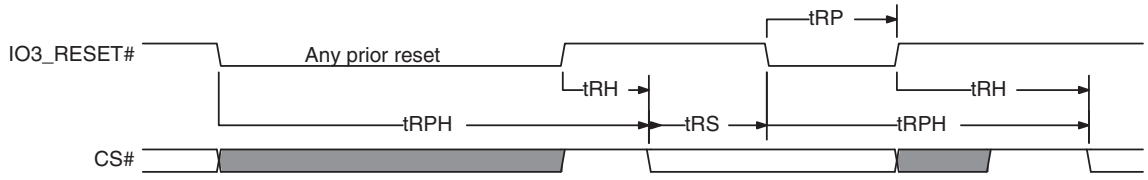
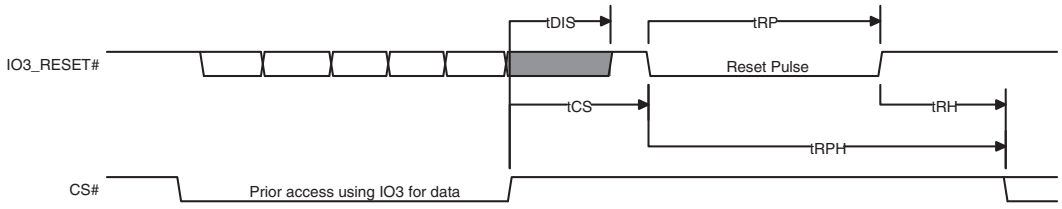



**Figure 27. Hardware Reset when Quad Mode and IO3 / Reset# are Enabled**



## 5.4 AC Characteristics

**Table 13. AC Characteristics**

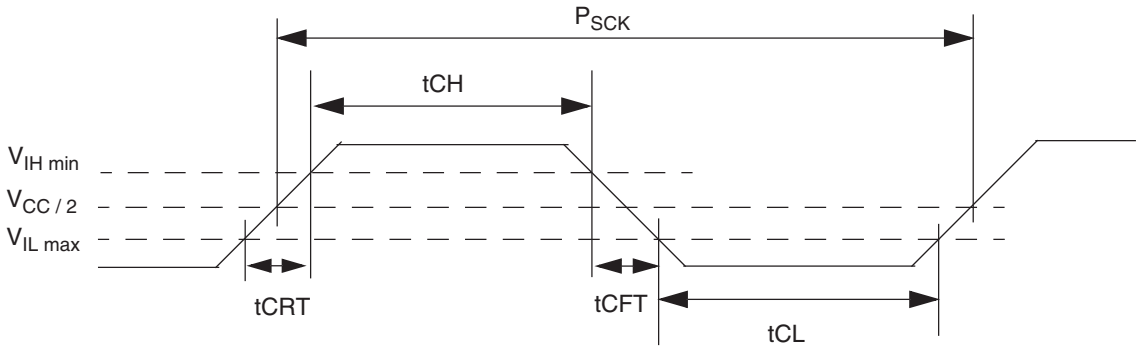
| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $F_{SCK, R}$ | SCK Clock Frequency for READ and 4READ instructions | DC | | 50 | MHz |
| $F_{SCK, C}$ | SCK Clock Frequency for single commands as shown in Table 36, S25FL127S Command Set (sorted by function) on page 68 (4) | DC | | 108 | MHz |
| $F_{SCK, C}$ | SCK Clock Frequency for the following dual and quad commands: DOR, 4DOR, QOR, 4QOR, DIOR, 4DIOR, QIOR, 4QIOR | DC | | 108 | MHz |
| $F_{SCK, QPP}$ | SCK Clock Frequency for the QPP, 4QPP commands | DC | | 80 | MHz |
| $P_{SCK}$ | SCK Clock Period | 1/ FSCK | | ∞ | |
| $t_{WH}$, $t_{CH}$ | Clock High Time (5) | 50% PSCK -5% | | 50% PSCK +5% | ns |
| $t_{WL}$, $t_{CL}$ | Clock Low Time (5) | 50% PSCK -5% | | 50% PSCK +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) CS# High Time (Read Instructions when Reset feature and Quad mode are both enabled) CS# High Time (Program/Erase Instructions) | 10<br>20 (7)<br>50 | | | ns |
| $t_{CSS}$ | CS# Active Setup Time (relative to SCK) | 3 | | | ns |
| $t_{CSH}$ | CS# Active Hold Time (relative to SCK) | 3 | | | ns |
| $t_{SU}$ | Data in Setup Time | 1.5 | | | ns |
| $t_{HD}$ | Data in Hold Time | 2 | | | ns |
| $t_V$ | Clock Low to Output Valid | 1 | | 8.0 (2)<br>7.65 (3)<br>6.5 (4) | ns |
| $t_{HO}$ | Output Hold Time | 2 | | | ns |
| $t_{DIS}$ | Output Disable Time (6) | | | 8 | ns |
| | Output Disable Time (when Reset feature and Quad mode are both enabled) | | | 20 (7) | ns |
| $t_{WPS}$ | WP# Setup Time | 20 (1) | | | ns |
| $t_{WPH}$ | WP# Hold Time | 100 (1) | | | ns |
| $t_{HLCH}$ | HOLD# Active Setup Time (relative to SCK) | 3 | | | ns |
| $t_{CHHH}$ | HOLD# Active Hold Time (relative to SCK) | 3 | | | ns |
| $t_{HHCH}$ | HOLD# Non Active Setup Time (relative to SCK) | 3 | | | ns |
| $t_{CHHL}$ | HOLD# Non Active Hold Time (relative to SCK) | 3 | | | ns |
| $t_{HZ}$ | HOLD# enable to Output Invalid | | | 8 | ns |
| $t_{LZ}$ | HOLD# enable to Output Valid | | | 8 | ns |

***Notes:***

1. *Only applicable as a constraint for WRR instruction when SRWD is set to a 1.*
2. *Full $V_{CC}$ range (2.7 - 3.6V) and CL = 30 pF.*
3. *Regulated $V_{CC}$ range (3.0 - 3.6V) and CL = 30 pF.*
4. *Regulated $V_{CC}$ range (3.0 - 3.6V) and CL = 15 pF.*
5. *±10% duty cycle is supported for frequencies $\leq$ 50 MHz.*
6. *Output High -Z is defined as the point where data is no longer driven.*
7. *$t_{CS}$ and $t_{DIS}$ require additional time when the Reset feature and Quad mode are enabled (CR2V[5]=1 and CR1V[1]=1).*

### 5.4.1 Clock Timing

**Figure 28. Clock Timing**




### 5.4.2 Input / Output Timing

**Figure 29. SPI Single Bit Input Timing**

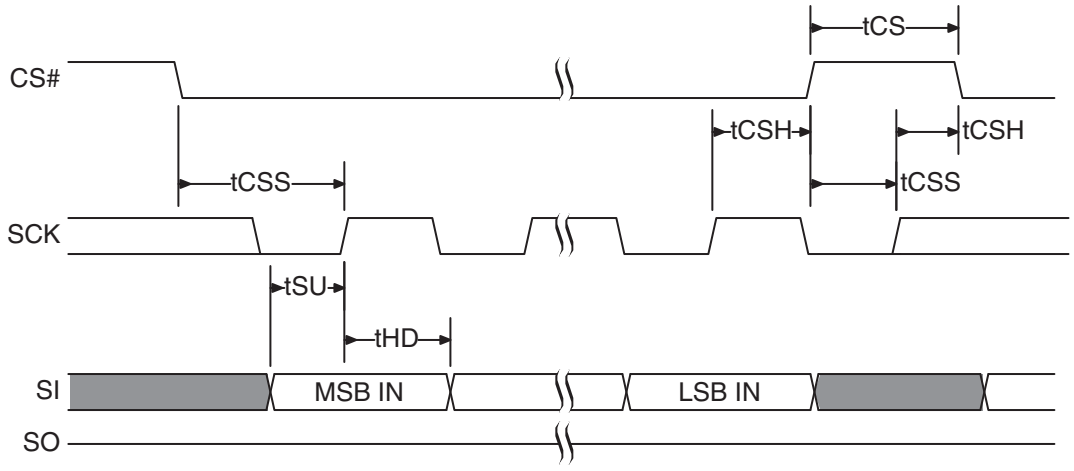



**Figure 30. SPI Single Bit Output Timing**

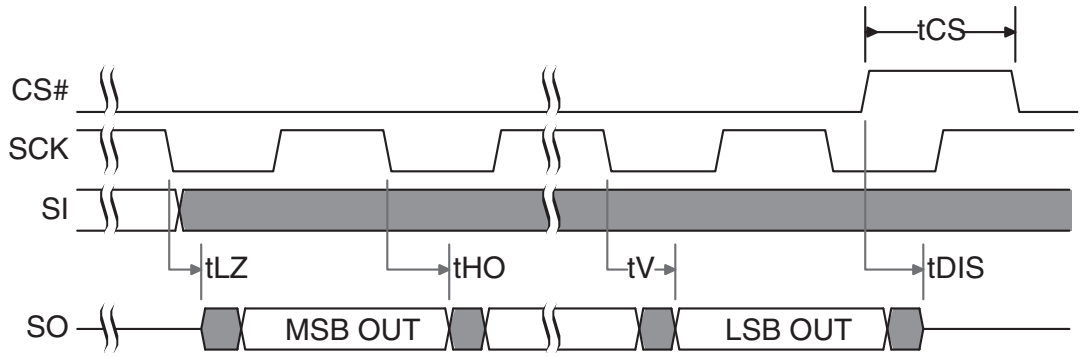


**Figure 31. SPI MIO Timing**

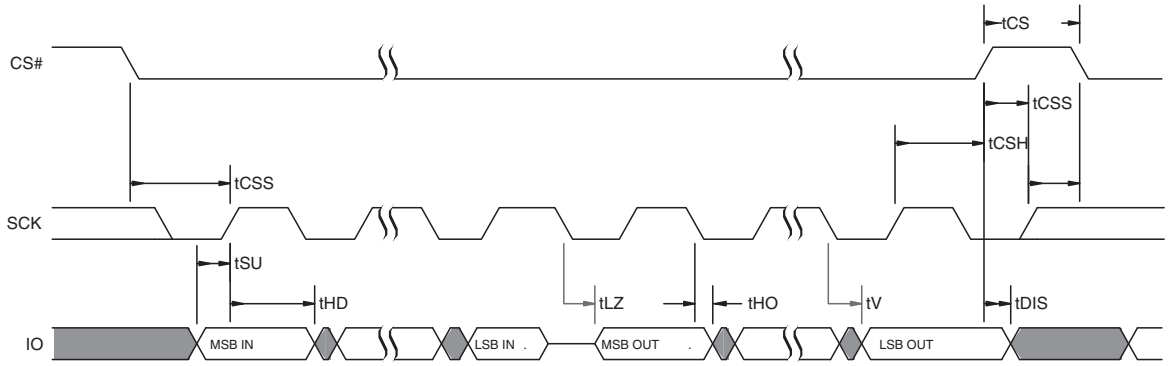



**Figure 32. Hold Timing**

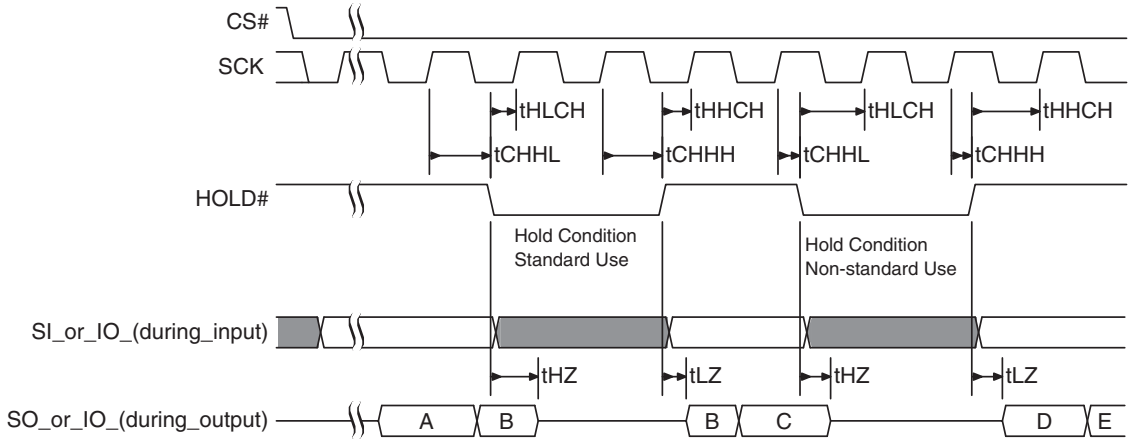
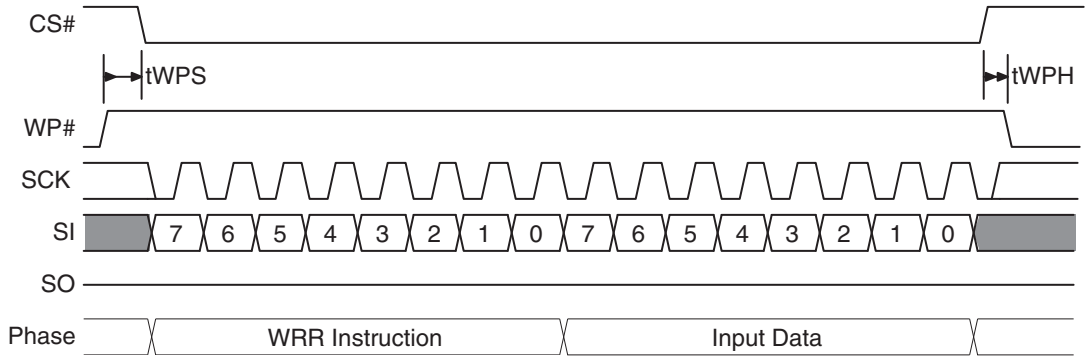



**Figure 33. WP# Input Timing**



# 6. Physical Interface

**Table 14. Model Specific Connections**

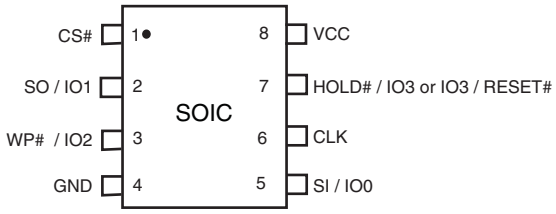
| | |
|---|---|
| RESET# / RFU | RESET# or RFU - Some device models bond this connector to the device RESET# signal, other models bond the RESET# signal to Vcc within the package leaving this package connector unconnected. |

***Note:***
*Refer to Table 2, Signal List on page 9 for signal descriptions.*

## 6.1 SOIC 8-Lead Package

### 6.1.1 SOIC-8 Connection Diagram

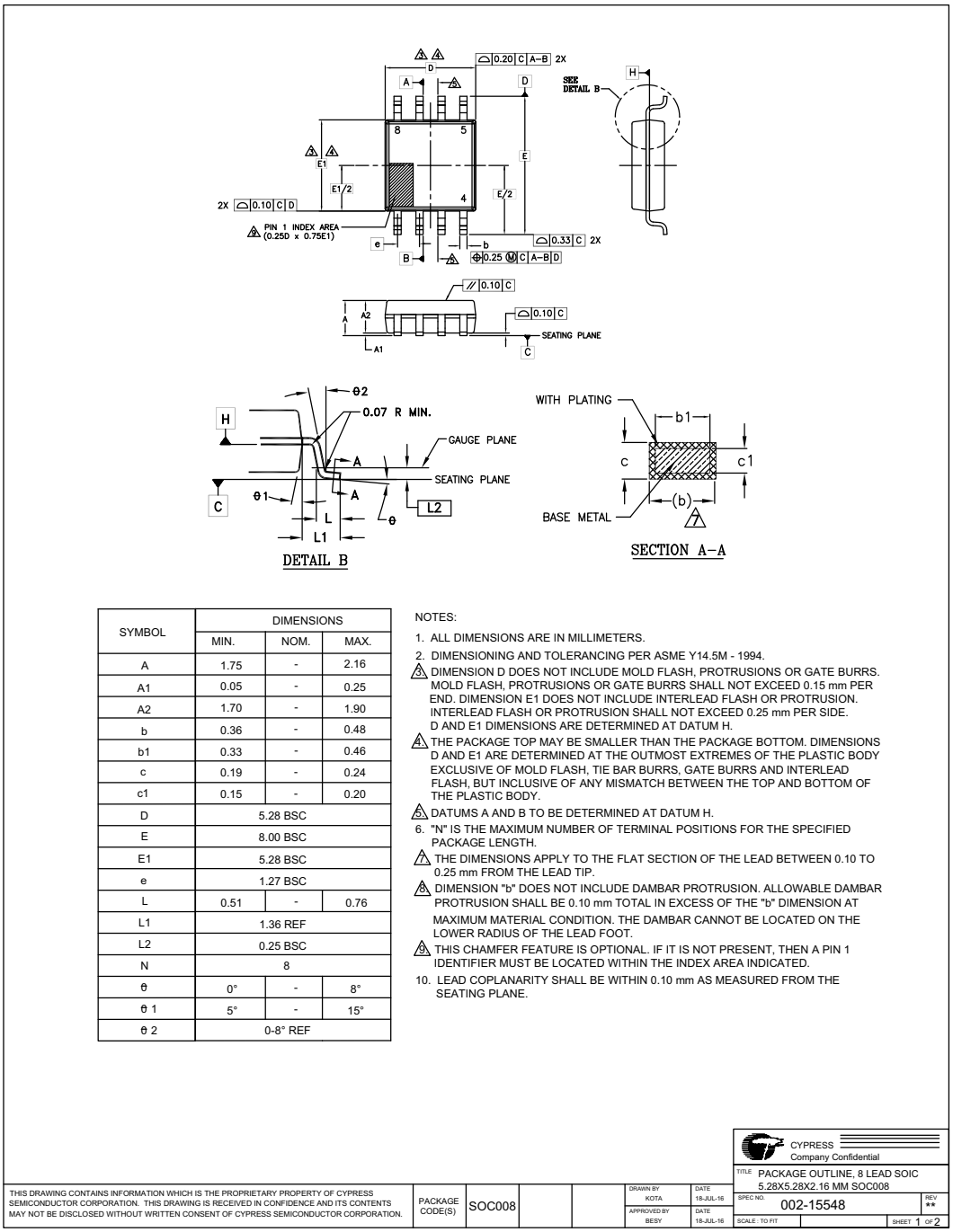
**Figure 34. 8-pin Plastic Small Outline Package (SO)**




***Note:***
1. *Lead 7 HOLD# / IO3 or IO3 / RESET# function depends on the selected configuration, If the IO3 / RESET# function is used, the host system should actively or passively pull-up the IO3 / RESET# connection when Quad mode is not enabled, or when CS# is high and a Reset operation is not intended.*

## 6.1.2 SOIC 8 Physical Diagram

**Figure 35. S0C008 — 8-Lead Plastic Small Outline Package (208-mil Body Width)**




| 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.

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.

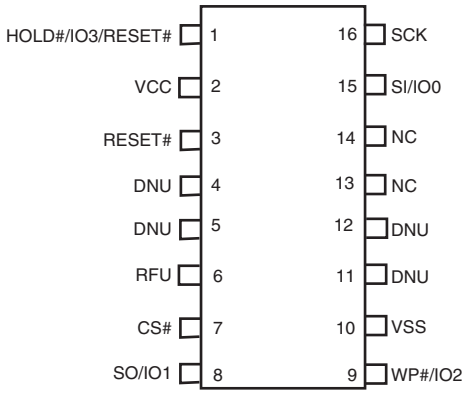
| PACKAGE CODE(S) | SOC008 | | | DRAWN BY KOTA | DATE 18-JUL-16 |
|---|---|---|---|---|---|
| | | | | APPROVED BY BESY | DATE 18-JUL-16 |

TITLE: PACKAGE OUTLINE, 8 LEAD SOIC 5.28X5.28X2.16 MM SOC008
SPEC NO. 002-15548 REV **
SCALE : TO FIT SHEET 1 OF 2

## 6.2 SOIC 16-Lead Package

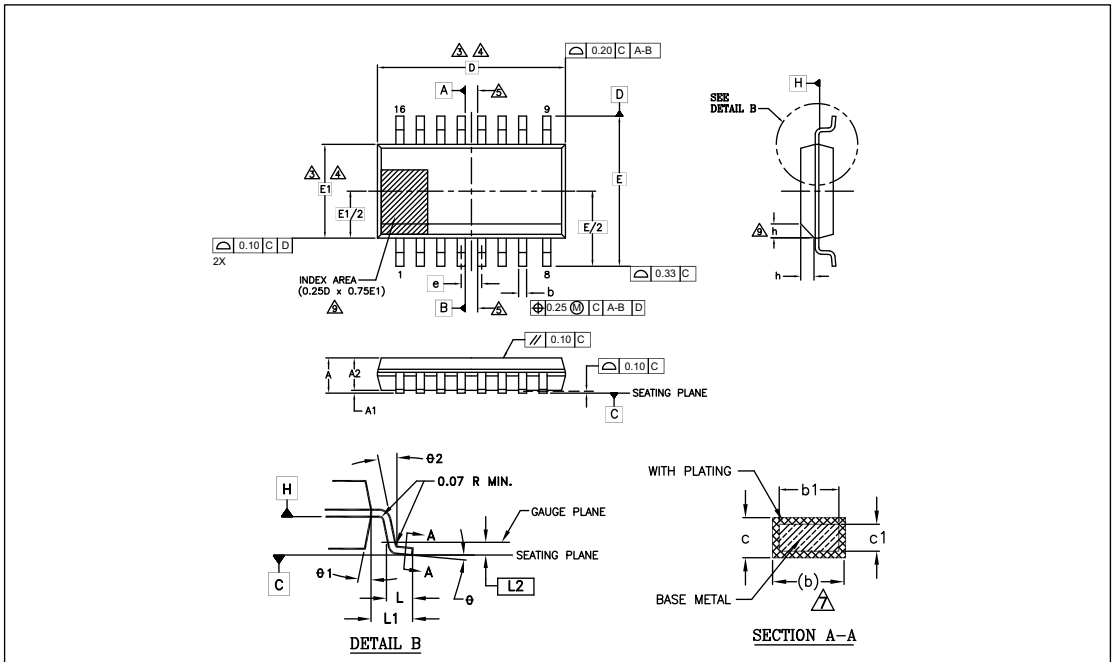
### 6.2.1 SOIC 16 Connection Diagram

**Figure 36. 16-Lead SOIC Package, Top View**



## 6.2.2 SOIC 16 Physical Diagram

**Figure 37. S03016 — 16-Lead Wide Plastic Small Outline Package (300-mil Body Width)**




| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| A | 2.35 | - | 2.65 |
| A1 | 0.10 | - | 0.30 |
| A2 | 2.05 | - | 2.55 |
| b | 0.31 | - | 0.51 |
| b1 | 0.27 | - | 0.48 |
| c | 0.20 | - | 0.33 |
| c1 | 0.20 | - | 0.30 |
| D | 10.30 BSC | | |
| E | 10.30 BSC | | |
| E1 | 7.50 BSC | | |
| e | 1.27 BSC | | |
| L | 0.40 | - | 1.27 |
| L1 | 1.40 REF | | |
| L2 | 0.25 BSC | | |
| N | 16 | | |
| h | 0.25 | - | 0.75 |
| θ | 0° | - | 8° |
| θ1 | 5° | - | 15° |
| θ2 | 0° | - | - |

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) | SO3016 | SL3016 | SS3016 |
|---|---|---|---|

DRAWN BY: KOTA — DATE: 24-OCT-16
APPROVED BY: BESY — DATE: 24-OCT-16

CYPRESS Company Confidential

TITLE: PACKAGE OUTLINE, 16 LEAD SOIC 10.30X7.50X2.65 MM SO3016/SL3016/SS3016

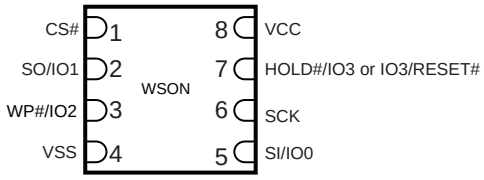
SPEC NO.: 002-15547 REV: *A

SCALE : TO FIT SHEET 1 OF 2

## 6.3 WSON 6 x 5 Package

### 6.3.1 WSON 6 x 5 mm Connection Diagram

**Figure 38. 8-Contact WSON 6 x 5 mm, Top View**

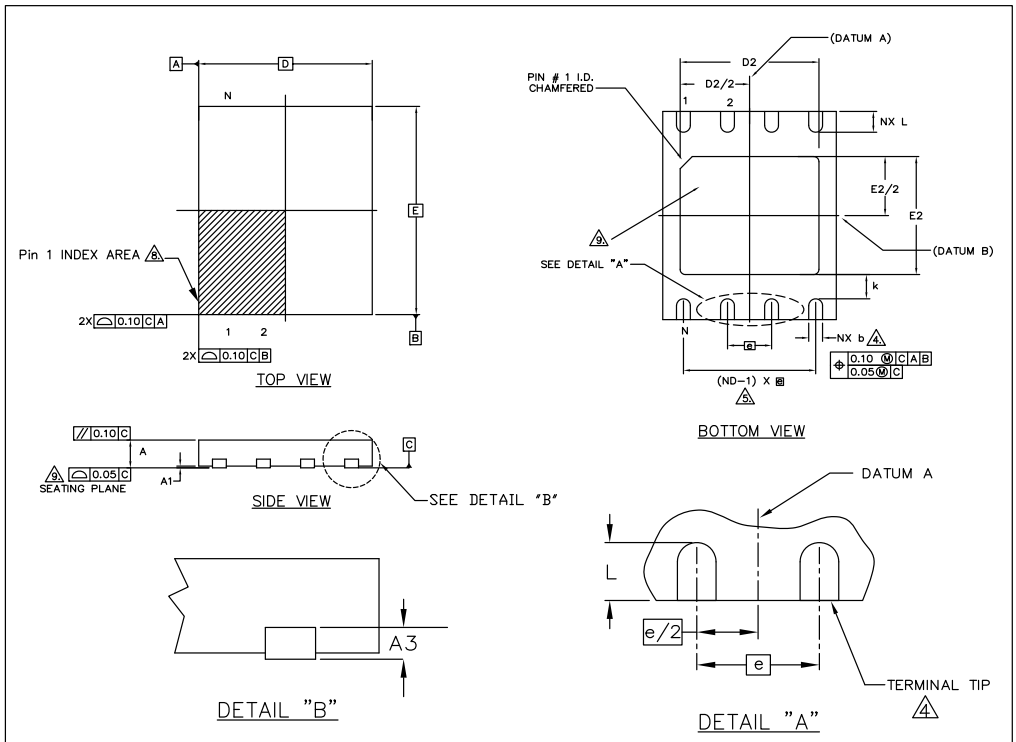



***Notes:***

1. *Lead 7 HOLD# / IO3 or IO3 / RESET# function depends on the selected configuration, If the IO3 / RESET# function is used, the host system should actively or passively pull-up the IO3 / RESET# connection when Quad mode is not enabled, or when CS# is high and a Reset operation is not intended.*
2. *There is an exposed central pad on the underside of the WSON package. This pad should not be connected to any voltage or signal line on the PCB. Connecting the central pad to GND ($V_{SS}$) is possible, provided PCB routing ensures 0 mV difference between voltage at the WSON GND ($V_{SS}$) lead and the central exposed pad.*

## 6.3.2 WSON Physical Diagram

**Figure 39. WND008 — WSON 8-contact (6 x 5 mm) No-Lead Package**




| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| e | | 1.27 BSC. | |
| N | | 8 | |
| ND | | 4 | |
| L | 0.55 | 0.60 | 0.65 |
| b | 0.35 | 0.40 | 0.45 |
| D2 | 3.90 | 4.00 | 4.10 |
| E2 | 3.30 | 3.40 | 3.50 |
| D | | 5.00 BSC | |
| E | | 6.00 BSC | |
| A | 0.70 | 0.75 | 0.80 |
| A1 | 0.00 | 0.02 | 0.05 |
| A3 | | 0.20 REF | |
| K | | 0.20 MIN. | |

NOTES:

1. DIMENSIONING AND TOLERANCING CONFORMS TO ASME Y14.5M-1994.
2. ALL DIMENSIONS ARE IN MILLIMETERS.
3. N IS THE TOTAL NUMBER OF TERMINALS.
4. DIMENSION "b" APPLIES TO METALLIZED TERMINAL AND IS MEASURED BETWEEN 0.15 AND 0.30mm FROM TERMINAL TIP. IF THE TERMINAL HAS THE OPTIONAL RADIUS ON THE OTHER END OF THE TERMINAL, THE DIMENSION "b" SHOULD NOT BE MEASURED IN THAT RADIUS AREA.
5. ND REFERS TO THE NUMBER OF TERMINALS ON D SIDE.
6. MAX. PACKAGE WARPAGE IS 0.05mm.
7. MAXIMUM ALLOWABLE BURR IS 0.076mm IN ALL DIRECTIONS.
8. PIN #1 ID ON TOP WILL BE LOCATED WITHIN THE INDICATED ZONE.
9. BILATERAL COPLANARITY ZONE APPLIES TO THE EXPOSED HEAT SINK SLUG AS WELL AS THE TERMINALS.
10. A MAXIMUM 0.15mm PULL BACK (L1) MAY BE PRESENT.

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): WND008

DRAWN BY: KOTA — DATE: 13-FEB-17
APPROVED BY: LKSU — DATE: 13-FEB-17

CYPRESS Company Confidential

TITLE: PACKAGE OUTLINE, 8 LEAD DFN 5.0X6.0X0.8 MM WND008 4.0X3.4 MM EPAD (SAWN)

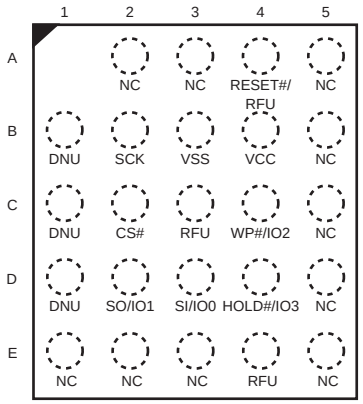
SPEC NO.: 002-18755 REV: **

SCALE: - TO FIT SHEET 1 OF 2

## 6.4 FAB024 24-Ball BGA Package

### 6.4.1 Connection Diagram

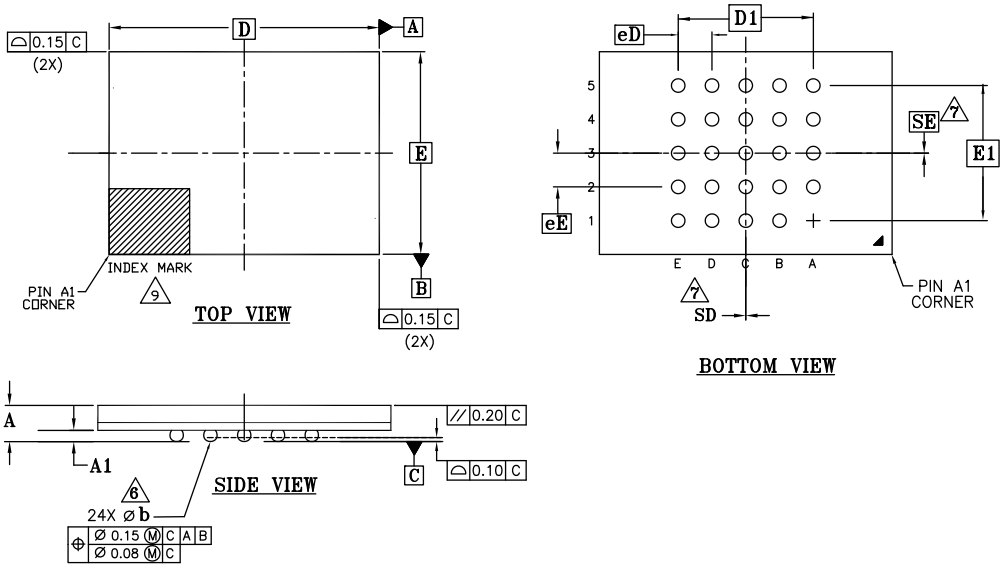
**Figure 40. 24-Ball BGA, 5 x 5 Ball Footprint (FAB024), Top View**




***Note:***
*Signal connections are in the same relative positions as FAC024 BGA, allowing a single PCB footprint to use either package.*

## 6.4.2 Physical Diagram

**Figure 41. FAB024 — 24-ball Ball Grid Array (8 x 6 mm) Package**




| 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 | | |
| Ø 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.

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 | DATE |
|---|---|---|---|
| | | KOTA | 18-JUL-16 |
| | | APPROVED BY | DATE |
| | | BESY | 18-JUL-16 |

CYPRESS Company Confidential

TITLE: PACKAGE OUTLINE, 24 BALL FBGA 8.0X6.0X1.2 MM FAB024

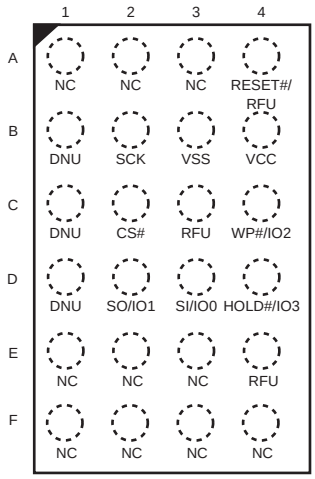
SPEC NO. 002-15534 REV **

SCALE : TO FIT SHEET 1 OF 2

## 6.5 FAC024 24-Ball BGA Package

### 6.5.1 Connection Diagram

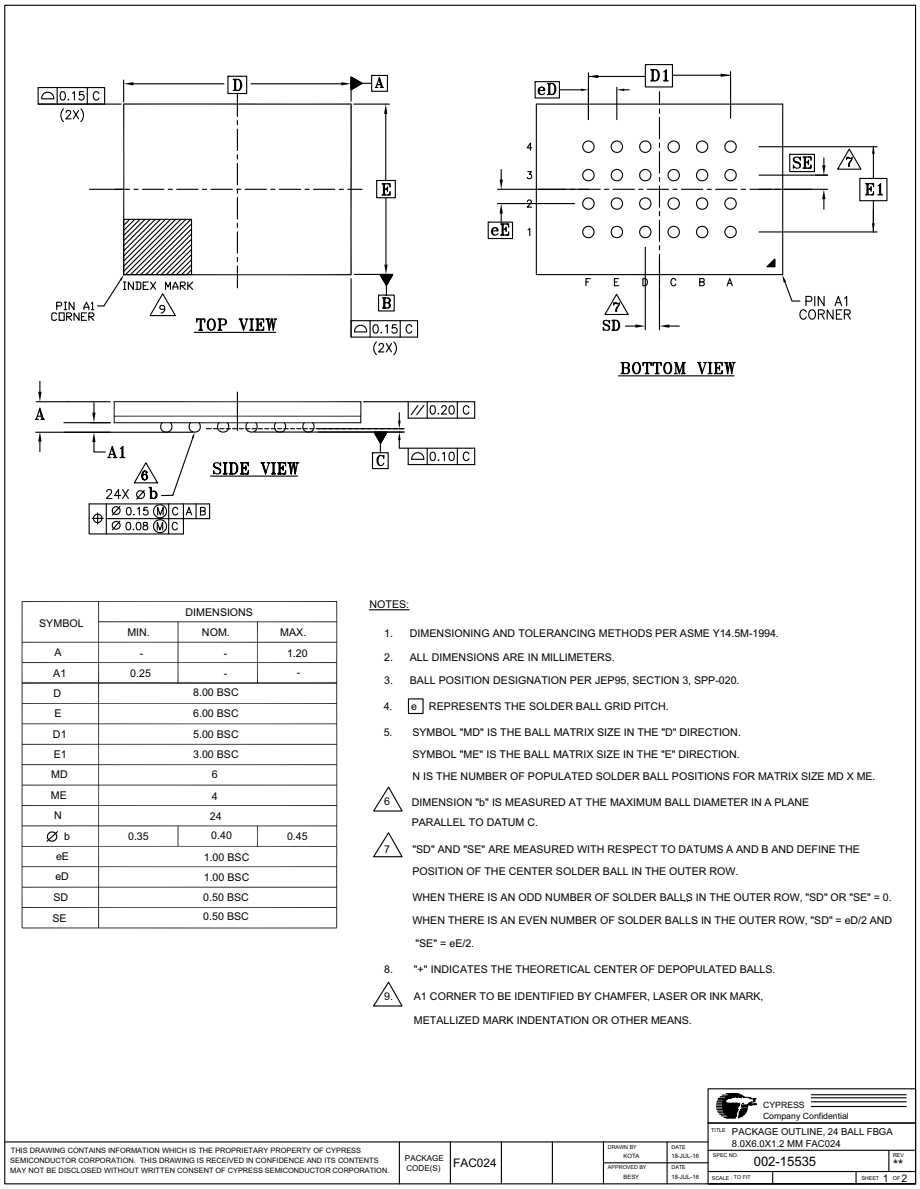
**Figure 42. 24-Ball BGA, 4 x 6 Ball Footprint (FAC024), Top View**




***Note:***
*Signal connections are in the same relative positions as FAB024 BGA, allowing a single PCB footprint to use either package.*

## 6.5.2 Physical Diagram

**Figure 43. FAC024 — 24-ball Ball Grid Array (6 x 8 mm) Package**




| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| A | - | - | 1.20 |
| A1 | 0.25 | - | - |
| D | | 8.00 BSC | |
| E | | 6.00 BSC | |
| D1 | | 5.00 BSC | |
| E1 | | 3.00 BSC | |
| MD | | 6 | |
| ME | | 4 | |
| N | | 24 | |
| Ø b | 0.35 | 0.40 | 0.45 |
| eE | | 1.00 BSC | |
| eD | | 1.00 BSC | |
| SD | | 0.50 BSC | |
| SE | | 0.50 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): FAC024
DRAWN BY: KOTA — DATE: 18-JUL-16
APPROVED BY: BIDY — DATE: 18-JUL-16
TITLE: PACKAGE OUTLINE, 24 BALL FBGA 8.0X6.0X1.2 MM FAC024
SPEC NO.: 002-15535
REV: **
SCALE: - TO FIT
SHEET 1 OF 2

## 6.5.3 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.

# Software Interface

This section discusses the features and behaviors most relevant to host system software that interacts with the S25FL127S memory device.

## 7. Address Space Maps

### 7.1 Overview

#### 7.1.1 Extended Address

The FL-S Family of devices supports 32-bit addresses to enable higher density devices than allowed by previous generation (legacy) SPI devices that supported only 24-bit addresses. A 24-bit byte resolution address can access only 16 Mbytes (128 Mbits) of maximum density. A 32-bit byte resolution address allows direct addressing of up to a 4 Gbytes (32 Gbits) of address space.

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

- Bank address register — a software (command) loadable internal register that supplies the high order bits of address when legacy 24-bit addresses are in use.
- Extended address mode — a bank address register bit that changes all legacy commands to expect 32 bits of address supplied from the host system.
- New commands — that perform both legacy and new functions, which expect 32-bit address.

The default condition at power-up or reset, is the Bank address register loaded with zeros and the extended address mode set for 24-bit addresses. This enables legacy software compatible access to the first 128 Mbits of a device.

The S25FL127S, 128 Mb density member of the FL-S Family, supports the extended address features in the same way but in essence ignores bits 31 to 24 of any address because the main flash array only needs 24 bits of address. This enables simple migration from the 128-Mb density to higher density devices without changing the address handling aspects of software.

#### 7.1.2 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 32-bit address but may only define a small portion of the available address space.

### 7.2 Flash Memory Array

The main flash array is divided into erase units called sectors. The sectors are organized either as a hybrid combination of 4-kB and 64-kB sectors, or as uniform 256-kbyte sectors. The sector organization depends on the D8h_O control bit configuration in the Status Register 2 (SR2[7]).

**Table 15. S25FL127S Sector and Memory Address Map, Bottom 4-kbyte Sectors**

| Sector Size (kbyte) | Sector Count | Sector Range | Address Range (Byte Address) | Notes |
|---|---|---|---|---|
| 4 | 16 | SA00 | 00000000h-00000FFFh | Sector Starting Address — Sector Ending Address |
| | | : | : | |
| | | SA15 | 0000F000h-0000FFFFh | |
| 64 | 255 | SA16 | 00010000h-0001FFFFh | |
| | | : | : | |
| | | SA270 | 00FF0000h-00FFFFFFh | |

**Table 16. S25FL127S Sector and Memory Address Map, Top 4-kbyte Sectors**

| Sector Size (kbyte) | Sector Count | Sector Range | Address Range (Byte Address) | Notes |
|---|---|---|---|---|
| 64 | 255 | SA00 | 0000000h-000FFFFh | Sector Starting Address — Sector Ending Address |
| | | : | : | |
| | | SA255 | 00FE0000h-00FEFFFFh | |
| 4 | 16 | SA256 | 00FF0000h-00FF0FFFh | |
| | | : | : | |
| | | SA270 | 00FFF000h-00FFFFFFh | |

**Table 17. S25FL127S Sector and Memory Address Map, Uniform 256-kbyte Sectors**

| Sector Size (kbyte) | Sector Count | Sector Range | Address Range (Byte Address) | Notes |
|---|---|---|---|---|
| 256 | 64 | SA00 | 0000000h-003FFFFh | Sector Starting Address — Sector Ending Address |
| | | : | : | |
| | | SA63 | 0FC0000h-0FFFFFFh | |

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

## 7.3 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 on page 118 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.

## 7.4 JEDEC JESD216B 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 JESD216B standard for Serial Flash Discoverable Parameters. The ID-CFI address space is incorporated as one of the SFDP parameters. See Section 12. Serial Flash Discoverable Parameters (SFDP) Address Map on page 115 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.

## 7.5 OTP Address Space

Each FL-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 0:

- The 16 lowest address bytes are programmed by Cypress with a 128-bit random number. Only Cypress is able to program these bytes.
- 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 44, OTP Address Space on page 49 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 (CR1[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 44. OTP Address Space**

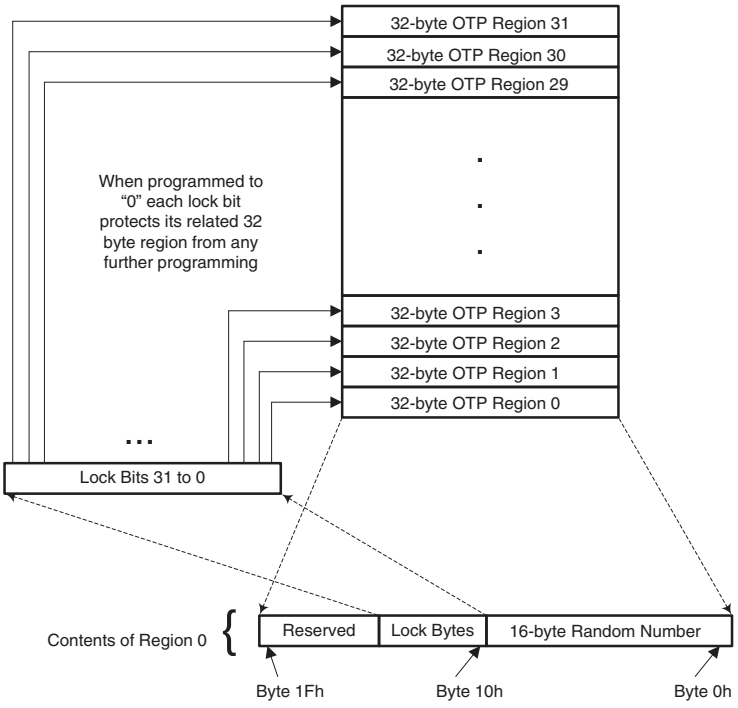


**Table 18. OTP Address Map**

| 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 31 from 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 |

## 7.6 Registers

Registers are small groups of memory cells used to configure how the S25FL-S 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. The individual register bits may be volatile, non-volatile, or One Time Programmable (OTP). 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 19. Register Descriptions**

| Register | Abbreviation | Type | Bit Location |
|---|---|---|---|
| Status Register 1 | SR1[7:0] | Volatile | 7:0 |
| Configuration Register 1 | CR1[7:0] | Volatile | 7:0 |
| Status Register 2 | SR2[7:0] | RFU | 7:0 |
| AutoBoot Register | ABRD[31:0] | Non-volatile | 31:0 |
| Bank Address Register | BRAC[7:0] | Volatile | 7:0 |
| ECC Status Register | ECCSR[7:0] | Volatile | 7:0 |
| ASP Register | ASPR[15:1] | OTP | 15:1 |
| ASP Register | ASPR[0] | RFU | 0 |
| Password Register | PASS[63:0] | Non-volatile OTP | 63:0 |
| PPB Lock Register | PPBL[7:1] | Volatile | 7:1 |
| PPB Lock Register | PPBL[0] | Volatile<br>Read Only | 0 |

**Table 19. Register Descriptions (Continued)**

| Register | Abbreviation | Type | Bit Location |
|---|---|---|---|
| PPB Access Register | PPBAR[7:0] | Non-volatile | 7:0 |
| DYB Access Register | DYBAR[7:0] | Volatile | 7:0 |
| SPI DDR Data Learning Registers | NVDLR[7:0] | Non-volatile | 7:0 |
| SPI DDR Data Learning Registers | VDLR[7:0] | Volatile | 7:0 |

### 7.6.1 Status Register 1 (SR1)

Related Commands: Read Status Register (RDSR1 05h), Write Registers (WRR 01h), Write Enable (WREN 06h), Write Disable (WRDI 04h), Clear Status Register (CLSR 30h).

**Table 20. Status Register 1 (SR1)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | SRWD | Status Register Write Disable | Non-Volatile | 0 | 1 = Locks state of SRWD, BP, and configuration register bits when WP# is low by ignoring WRR command<br>0 = No protection, even when WP# is low |
| 6 | P_ERR | Programming Error Occurred | Volatile, Read only | 0 | 1 = Error occurred.<br>0 = No Error |
| 5 | E_ERR | Erase Error Occurred | Volatile, Read only | 0 | 1 = Error occurred<br>0 = No Error |
| 4 | BP2 | Block Protection | Volatile if CR1[3]=1, Non-Volatile if CR1[3]=0 | 1 if CR1[3]=1, 0 when shipped from Cypress | Protects selected range of sectors (Block) from Program or Erase |
| 3 | BP1 | | | | |
| 2 | BP0 | | | | |
| 1 | WEL | Write Enable Latch | Volatile | 0 | 1 = Device accepts Write Registers (WRR), program or erase commands<br>0 = Device ignores Write Registers (WRR), program or erase commands<br>This bit is not affected by WRR, only WREN and WRDI commands affect this bit |
| 0 | WIP | Write in Progress | Volatile, Read only | 0 | 1 = Device Busy, a Write Registers (WRR), program, erase or other operation is in progress<br>0 = Ready Device is in standby mode and can accept commands |

The Status Register contains both status and control bits:

**Status Register Write Disable (SRWD) SR1[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) command is no longer accepted for execution, effectively locking the state of the SRWD bit, BP bits, and Configuration register bits by making the Status Register and Configuration Register read-only. If WP# is high the SRWD bit and BP bits may be changed by the WRR command. If SRWD is 0, WP# has no effect and the SRWD bit and BP bits may be changed by the WRR command. The SRWD bit has the same non-volatile endurance as the main flash array.

**Program Error (P_ERR) SR1[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 locked OTP region. When the Program Error bit is set to a 1, this bit can be reset to 0 with the Clear Status Register (CLSR) command. This is a read-only bit and is not affected by the WRR command.

**Erase Error (E_ERR) SR1[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 reset to 0 with the Clear Status Register (CLSR) command. This is a read-only bit and is not affected by the WRR command.

**Block Protection (BP2, BP1, BP0) SR1[4:2]**: These bits define the main flash array area to be software-protected against program and erase commands. The BP bits are either volatile or non-volatile, depending on the state of the BP non-volatile bit (BPNV) in the configuration register. 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 60 for a description of how the BP bit values select the memory array area protected. The BP bits have the same non-volatile endurance as the main flash array.

**Write Enable Latch (WEL) SR1[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 command does not affect this bit.

**Write In Progress (WIP) SR1[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), Erase Suspend (ERSP), Program Suspend (PGSP), Clear Status Register (CLSR), and Software Reset (RESET) commands may be 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 to1 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.

### 7.6.2 Configuration Register 1 (CR1)

Related Commands: Read Configuration Register (RDCR 35h), Write Registers (WRR 01h). The Configuration Register bits can be changed using the WRR command with sixteen input cycles.

The configuration register controls certain interface and data protection functions.

**Table 21. Configuration Register (CR1)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | LC1 | Latency Code | Non-Volatile | 0 | Selects number of initial read latency cycles<br>See Latency Code Tables |
| 6 | LC0 | | | 0 | |
| 5 | TBPROT | Configures Start of Block Protection | OTP | 0 | 1 = BP starts at bottom (Low address)<br>0 = BP starts at top (High address) |
| 4 | RFU | RFU | OTP | 0 | Reserved for Future Use |
| 3 | BPNV | Configures BP2-0 in Status Register | OTP | 0 | 1 = Volatile<br>0 = Non-Volatile |
| 2 | TBPARM | 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 devices |
| 1 | QUAD | Puts the device into Quad I/O operation | Non-Volatile | 0 | 1 = Quad<br>0 = Dual or Serial |
| 0 | FREEZE | Lock current state of BP2-0 bits in Status Register, TBPROT and TBPARM in Configuration Register, and OTP regions | Volatile | 0 | 1 = Block Protection and OTP locked<br>0 = Block Protection and OTP un-locked |

**Latency Code (LC) CR1[7:6]:** The Latency Code selects the number of mode and dummy cycles between the end of address and the start of read data output for all read commands.

Some read commands send 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.

Dummy cycles provide additional latency that is needed to complete the initial read access of the flash array before data can be returned to the host system. Some read commands require additional latency cycles as the SCK frequency is increased.

The following latency code tables provide different latency settings that are configured by Cypress.

Where mode or latency (dummy) cycles are shown in the tables as a dash, that read command is not supported at the frequency shown. Read is supported only up to 50 MHz but the same latency value is assigned in each latency code and the command may be used when the device is operated at ≤ 50 MHz with any latency code setting. Similarly, only the Fast Read command is supported up to 108 MHz but the same 10b latency code is used for Fast Read up to 108 MHz and for the other dual and quad read commands up to 108 MHz. It is not necessary to change the latency code from a higher to a lower frequency when operating at lower frequencies where a particular command is supported. The latency code values for a higher frequency can be used for accesses at lower frequencies.

**Table 22. Latency Codes**

| Freq. (MHz) | LC | Read (03h, 13h) | | Fast Read (0Bh, 0Ch) | | Read Dual Out (3Bh, 3Ch) | | Read Quad Out (6Bh, 6Ch) | | Dual I/O Read (BBh, BCh) | | Quad I/O Read (EBh, ECh) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Mode | Dummy | Mode | Dummy | Mode | Dummy | Mode | Dummy | Mode | Dummy | Mode | Dummy |
| ≤ 50 | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 2 | 1 |
| ≤ 80 | 00 | - | - | 0 | 8 | 0 | 8 | 0 | 8 | 4 | 4 | 2 | 4 |
| ≤ 90 | 01 | - | - | 0 | 8 | 0 | 8 | 0 | 8 | 4 | 1 | 2 | 4 |
| ≤104 | 10 | - | - | 0 | 8 | 0 | 8 | 0 | 8 | 4 | 2 | 2 | 5 |
| ≤ 108 | 10 | - | - | 0 | 8 | - | - | - | - | - | - | - | - |

**Top or Bottom Protection (TBPROT) CR1[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, 1/4, 1/2, etc., up to the entire array. When TBPROT is set to a 0 the Block Protection is defined to start from the top (maximum address) of the array. When TBPROT is set to a 1 the Block Protection is defined to start from the bottom (zero address) of the array. The TBPROT bit is OTP and set to a 0 when shipped from Cypress. If TBPROT is programmed to 1, an attempt to change it back to 0 will fail and set the Program Error bit (P_ERR in SR1[6]).

The desired state of TBPROT 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 must not be programmed after programming or erasing is done in the main flash array.

**CR1[4]:** Reserved for Future Use

**Block Protection Non-Volatile (BPNV) CR1[3]**: The BPNV bit defines whether or not the BP2-0 bits in the Status Register are volatile or non-volatile. The BPNV bit is OTP and cleared to a 0 with the BP bits cleared to 000 when shipped from Cypress. When BPNV is set to a 0 the BP2-0 bits in the Status Register are non-volatile. The time required to write the BP bits when they are non-volatile is $t_W$. When BPNV is set to a 1 the BP2-0 bits in the Status Register are volatile and will be reset to binary 111 after POR, hardware reset, or command reset. 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 is programmed to 1, an attempt to change it back to 0 will fail and set the Program Error bit (P_ERR in SR1[6]).

**TBPARM CR1[2]**: TBPARM defines the logical location of the parameter block. The parameter block consists of sixteen 4-kB small sectors, which replace one 64-kB sector. When TBPARM is set to a 1 the parameter block is in the top of the memory array address space. When TBPARM is set to a 0 the parameter block is at the Bottom of the array. TBPARM is OTP and set to a 0 when it ships from Cypress. If TBPARM is programmed to 1, an attempt to change it back to 0 will fail and set the Program Error bit (P_ERR in SR1[6]).

The desired state of TBPARM 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 must not be programmed after programming or erasing is done in the main flash array.

TBPROT can be set or cleared independent of the TBPARM 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, and vice versa. Or the user can select to store and protect the parameter information starting from the top or bottom together.

When the memory array is logically configured as uniform 256-kB sectors, the TBPARM bit is Reserved for Future Use (RFU) and has no effect because all sectors are uniform size.

**Quad Data Width (QUAD) CR1[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 HOLD# becomes IO3. The WP# and HOLD# inputs are not monitored for their normal functions and are internally set to high (inactive). The commands for Serial, Dual Output, and Dual I/O Read still function normally but, there is no need to drive WP# and Hold# inputs for those commands when switching between commands using different data path widths. The QUAD bit must be set to 1 when using Read Quad Out, Quad I/O Read, and Quad Page Program commands. The QUAD bit is non-volatile.

**Freeze Protection (FREEZE) CR1[0]**: The Freeze Bit, when set to 1, locks the current state of the BP2-0 bits in Status Register, the TBPROT and TBPARM bits in the Configuration Register, and the OTP address space. This prevents writing, programming, or erasing these areas. As long as the FREEZE bit remains cleared to logic 0 the other bits of the Configuration Register, including 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 FREEZE bit is volatile and the default state of FREEZE after power-on is 0. The FREEZE bit can be set in parallel with updating other values in CR1 by a single WRR command.

### 7.6.3 Status Register 2 (SR2)

Related Commands: Read Status Register 2 (RDSR2 07h), Write Registers (WRR 01h). The Status Register 2 OTP bits can be changed using the WRR command with 24 input cycles.

**Table 23. Status Register 2 (SR2)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | D8h_O | Block Erase Size | OTP | 0 | 1 = 256 kB Erase (Uniform sectors).<br>0 = 64 kB Erase (Hybrid 4-kB / 64-kB sectors). |
| 6 | 02h_O | Page Buffer Wrap | OTP | 0 | 1 = Wrap at 512B 0 = Wrap at 256B. |
| 5 | IO3R_O | IO3 Reset | OTP | 0 | 1 = IO3 alternate function is RESET#.<br>0 = IO3 alternate function is HOLD#. |
| 4 | RFU | Reserved | | 0 | Reserved for Future Use. |
| 3 | RFU | Reserved | | 0 | Reserved for Future Use. |
| 2 | RFU | Reserved | | 0 | Reserved for Future Use. |
| 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. |

**D8h SR2[7]**: This bit controls the area erased by the D8h instruction. The D8h instruction can be used to erase 64-kB 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. When the default 64-kB erase option is in use the flash memory array has a hybrid of sixteen 4-kB sectors at the top or bottom of the array with all other sectors being 64 kB. Individual 4-kB sectors are erased by the 20h instruction. A 64-kB block of 4-kB sectors or an individual 64-kB sector can be erased by the D8h instruction. When the 256-kB option is in use, the flash memory array is treated as uniform 256-kB blocks that are individually erased by the D8h instruction.

The desired state of this bit (D8h_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 is performed. D8h_O must not be programmed after programming or erasing is done in the main flash array.

**02h SR2[6]**: 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 0 of the buffer. The FL-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.

**IO3 Reset Non-volatile SR2[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 S25FL127S device 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 OTP IO3 Reset configuration bit enables the device to start immediately (boot) with IO3 enabled for alternate use as a RESET# signal. When left in the default state, the IO3 signal has an alternate use as HOLD#.

**Erase Suspend (ES) SR2[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. 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 Section 9.6.4 Erase Suspend and Resume Commands (ERSP 75h or ERRS 7Ah) on page 99 for details about the Erase Suspend/Resume commands.

**Program Suspend (PS) SR2[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. 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 Section 9.5.4 Program Suspend (PGSP 85h) and Resume (PGRS 8Ah) on page 95 for details.

### 7.6.4 AutoBoot Register

Related Commands: AutoBoot Read (ABRD 14h) and AutoBoot Write (ABWR 15h).

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

**Table 24. AutoBoot Register**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 31 to 9 | ABSA | AutoBoot Start Address | Non-Volatile | 000000h | 512-byte boundary address for the start of boot code access |
| 8 to 1 | ABSD | AutoBoot Start Delay | Non-Volatile | 00h | Number of initial delay cycles between CS# going low and the first bit of boot code being transferred |
| 0 | ABE | AutoBoot Enable | Non-Volatile | 0 | 1 = AutoBoot is enabled<br>0 = AutoBoot is not enabled |

### 7.6.5 Bank Address Register

Related Commands: Bank Register Access (BRAC B9h), Write Register (WRR 01h), Bank Register Read (BRRD 16h) and Bank Register Write (BRWR 17h).

The Bank Address register supplies additional high order bits of the main flash array byte boundary address for legacy commands that supply only the low order 24 bits of address. The Bank Address is used as the high bits of address (above A23) for all 3-byte address commands when EXTADD=0. The Bank Address is not used when EXTADD = 1 and traditional 3-byte address commands are instead required to provide all four bytes of address.

**Table 25. Bank Address Register (BAR)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | EXTADD | Extended Address Enable | Volatile | 0b | 1 = 4 byte (32 bits) addressing required from command.<br>0 = 3 byte (24 bits) addressing from command + Bank Address. |
| 6 to 2 | RFU | Reserved | Volatile | 00000b | Reserved for Future Use. |
| 1 | BA25 | Bank Address | Volatile | 0 | RFU. |
| 0 | BA24 | Bank Address | Volatile | 0 | RFU. |

Extended Address (EXTADD) BAR[7]: EXTADD controls the address field size for legacy SPI commands. By default (power up reset, hardware reset, and software reset), it is cleared to 0 for 3 bytes (24 bits) of address. When set to 1, the legacy commands will require 4 bytes (32 bits) for the address field. This is a volatile bit.

### 7.6.6 ECC Status Register (ECCSR)

Related Commands: ECC Read (ECCRD 18h). 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 Section 9.5.1.1 Automatic ECC on page 93.

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 26. 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.<br>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.

### 7.6.7 ASP Register (ASPR)

Related Commands: ASP Read (ASPRD 2Bh) and ASP Program (ASPP 2Fh).

The ASP register is a 16-bit OTP memory location used to permanently configure the behavior of Advanced Sector Protection (ASP) features.

**Table 27. 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 | (Note 1) | Reserved for Future Use. |
| 7 | RFU | Reserved | OTP | (Note 1) | Reserved for Future Use. |
| 6 | RFU | Reserved | OTP | 1 | Reserved for Future Use. |
| 5 | RFU | Reserved | OTP | (Note 1) | Reserved for Future Use. |
| 4 | RFU | Reserved | OTP | (Note 1) | Reserved for Future Use. |
| 3 | RFU | Reserved | OTP | (Note 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 | RFU | Reserved | OTP | (Note 1) | Reserved for Future Use. |

***Note:***
*1. Default value depends on ordering part number, see Initial Delivery State on page 137.*

**Reserved for Future Use (RFU) ASPR[15:3, 0]**.

**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 and PSTMLB are mutually exclusive, only one may be programmed to 0.

When the ASP protection mode is selected by programming either ASPR[2] or ASPR[1], certain OTP configuration bits are locked and permanently protected from further programming. The bits protected are:

- SR2[7:5]
- ASPR
- PASS

The OTP configuration must be selected before selecting the ASP protection mode.

Attempting to program the listed OTP configuration bits when ASPR[2:1] is not = 11b will result in a programming error with P_ERR (SR1[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.

### 7.6.8 Password Register (PASS)

Related Commands: Password Read (PASSRD E7h) and Password Program (PASSP E8h).

**Table 28. 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 0. |

### 7.6.9 PPB Lock Register (PPBL)

Related Commands: PPB Lock Read (PLBRD A7h, PLBWR A6h).

**Table 29. 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 | Persistent Protection Mode = 1<br>Password Protection Mode = 0 | 0 = PPB array protected until next power cycle or hardware reset<br>1 = PPB array may be programmed or erased. |

### 7.6.10 PPB Access Register (PPBAR)

Related Commands: PPB Read (PPBRD E2h), PPB Program (PPBP E3), PPB Erase (PPBE E4).

**Table 30. 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 or PPBP command is erased to 1, not protecting that sector from program or erase operations. |

### 7.6.11 DYB Access Register (DYBAR)

Related Commands: DYB Read (DYBRD E0h) and DYB Program (DYBWR E1h).

**Table 31. DYB Access Register (DYBAR)**

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

# 8. Data Protection

## 8.1 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 49, One Time Program Array Commands on page 102, and OTP Read (OTPR 4Bh) on page 102.

### 8.1.1 Reading OTP Memory Space

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

### 8.1.2 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 44, OTP Address Space on page 49. OTP Program operations outside the valid OTP address range will be ignored and the WEL in SR1 will remain high (set to 1). OTP Program operations while FREEZE = 1 will fail with P_ERR in SR1 set to 1.

### 8.1.3 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.

### 8.1.4 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 Section 7.5 OTP Address Space on page 49.

## 8.2 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)
- Sector Erase (SE)
- Bulk Erase (BE)
- Write Disable (WRDI)
- Write Registers (WRR)
- Quad-input Page Programming (QPP)
- OTP Byte Programming (OTPP)

## 8.3 Block Protection

The Block Protect bits (Status Register bits BP2, BP1, BP0) in combination with the Configuration Register TBPROT 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 bit of the configuration register.

**Table 32. Upper Array Start of Protection (TBPROT = 0)**

| Status Register Content | | | Protected Fraction of Memory Array | Protected Memory (kbytes) |
|---|---|---|---|---|
| BP2 | BP1 | BP0 | | FL127S 128 Mb |
| 0 | 0 | 0 | None | 0 |
| 0 | 0 | 1 | Upper 64th | 256 |
| 0 | 1 | 0 | Upper 32nd | 512 |
| 0 | 1 | 1 | Upper 16th | 1024 |
| 1 | 0 | 0 | Upper 8th | 2048 |
| 1 | 0 | 1 | Upper 4th | 4096 |
| 1 | 1 | 0 | Upper Half | 8192 |
| 1 | 1 | 1 | All Sectors | 16384 |

**Table 33. Lower Array Start of Protection (TBPROT = 1)**

| Status Register Content | | | Protected Fraction of Memory Array | Protected Memory (kbytes) |
|---|---|---|---|---|
| BP2 | BP1 | BP0 | | FL127S 128 Mb |
| 0 | 0 | 0 | None | 0 |
| 0 | 0 | 1 | Lower 64th | 256 |
| 0 | 1 | 0 | Lower 32nd | 512 |
| 0 | 1 | 1 | Lower 16th | 1024 |
| 1 | 0 | 0 | Lower 8th | 2048 |
| 1 | 0 | 1 | Lower 4th | 4096 |
| 1 | 1 | 0 | Lower Half | 8192 |
| 1 | 1 | 1 | All Sectors | 16384 |

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. Recommendation: ASP and Block Protection should not be used concurrently. Use one or the other, but not both.

### 8.3.1 Freeze bit

Bit0 of the Configuration Register is the FREEZE bit. The FREEZE bit locks the BP2-0 bits in Status Register 1 and the TBPROT bit in the Configuration Register to their value at the time the FREEZE bit is set to 1. Once the FREEZE bit has been written to a logic 1 it cannot be cleared to a logic 0 until a power-on-reset is executed. As long as the FREEZE bit is cleared to logic 0 the status register BP bits and the TBPROT bit of the Configuration Register are writable. The FREEZE bit also protects the entire OTP memory space from programming when set to 1. Any attempt to change the BP bits with the WRR command while FREEZE = 1 is ignored and no error status is set.

### 8.3.2 Write Protect Signal

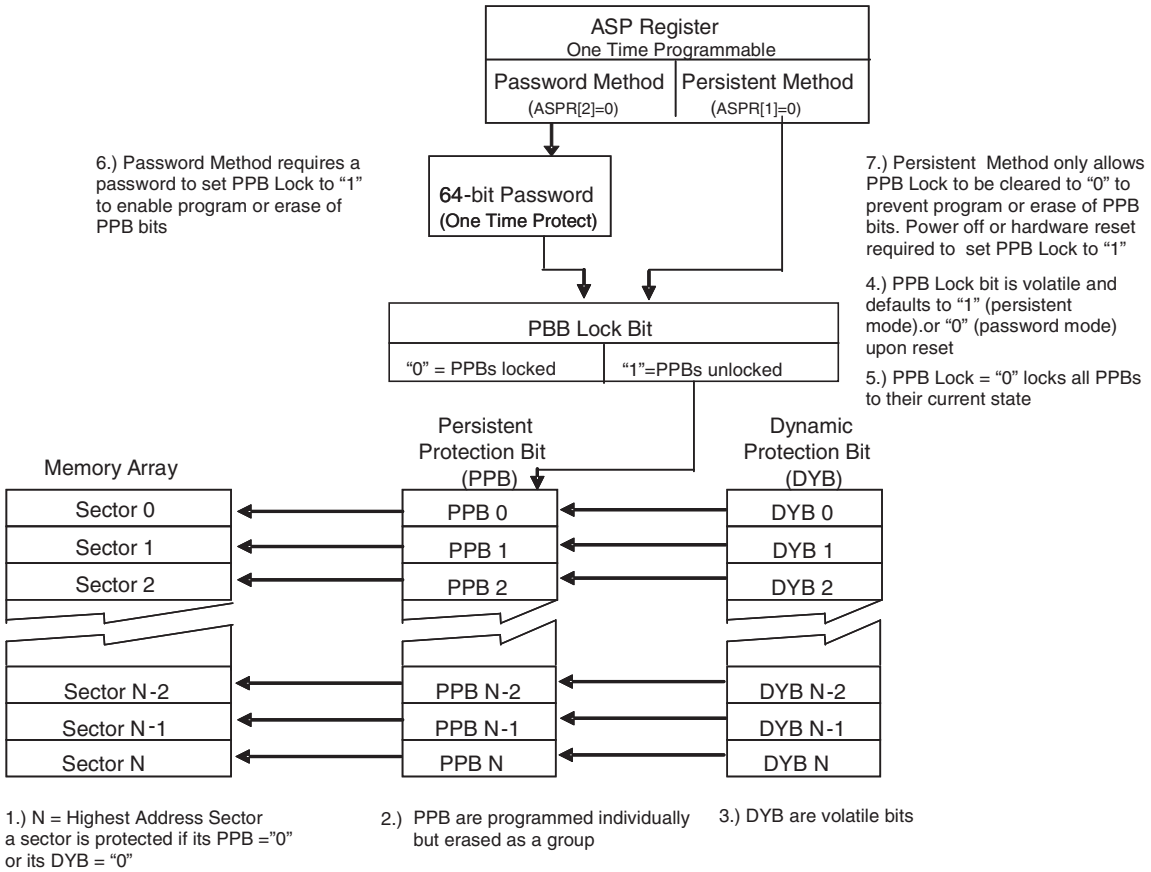
The Write Protect (WP#) input in combination with the Status Register Write Disable (SRWD) bit provide hardware input signal controlled protection. When WP# is Low and SRWD is set to 1 the Status and Configuration register is protected from alteration. This prevents disabling or changing the protection defined by the Block Protect bits. See Section 7.6.1 Status Register 1 (SR1) on page 51

## 8.4 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. An overview of these methods is shown in Figure 45, Advanced Sector Protection Overview on page 61.

Block Protection and ASP protection settings for each sector are logically OR'd 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 60 for full details of the BP2-0 bits.

**Figure 45. Advanced Sector Protection Overview**



Every main flash array sector has a non-volatile (PPB) and a volatile (DYB) protection bit 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 PPB Lock bit is 0. There are two methods for managing the state of the PPB Lock bit, Persistent Protection and Password Protection.

The Persistent Protection method sets the PPB Lock bit to 1 during POR, or Hardware Reset so that the PPB bits are unprotected by a device reset. 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.

### 8.4.1 ASP Register

The ASP register is used to permanently configure the behavior of Advanced Sector Protection (ASP) features. See Table 27, ASP Register (ASPR) on page 56.

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 = Illegal condition, attempting to program both bits to 0 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 Protection Mode Lock Bits are permanently protected from programming and no further changes to the ASP register is allowed.

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 Register 1 (SR1) on page 51 for information on WIP.

After selecting a sector protection method, each sector can operate in each of the following states:

- Dynamically Locked — A sector is protected and can be changed by a simple command.
- Persistently Locked — A sector is protected and cannot be changed if its PPB Bit is 0.
- Unlocked — The sector is unprotected and can be changed by a simple command. The Block Protection bits may be used to lock a sector/sectors.

### 8.4.2 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**:

- Each PPB is individually programmed to 0 and all are erased to 1 in parallel.
- 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.
- The state of the PPB for a given sector can be verified by using the PPB Read command.

### 8.4.3 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.

### 8.4.4 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 and when set to 1, it allows the PPBs to be changed.

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.

### 8.4.5 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 after a power cycle, software reset, or hardware reset. The Block Protection bits may be used to protect a sector/sectors.
- 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 34. 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 |

### 8.4.6 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. 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.

### 8.4.7 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 and Password Read commands.
- 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. 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 command. When a valid password is provided the password unlock command does not insert the 100 µs delay before returning the WIP bit to 0.
- 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.

# 9. Commands

All communication between the host system and S25FL127S 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 serially between the host system and memory device.

All instructions are transferred from host to memory as a single bit serial sequence on the SI signal.

Single bit wide commands may provide an address or data sent only on the SI signal. Data may be sent back to the host serially on SO signal.

Dual or Quad Output commands provide an address sent to the memory only on the SI signal. Data will be returned to the host as a sequence of bit pairs on IO0 and IO1 or four bit (nibble) groups on IO0, IO1, IO2, and IO3.

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.

Commands are structured as follows:

- Each command begins with an eight bit (byte) instruction.
- 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 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.
- The width of all transfers following the instruction are determined by the instruction sent.
- All single bits or parallel bit groups are transferred in most to least significant bit order.
- Some instructions send instruction modifier (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 15.

– The host always controls the Chip Select (CS#), Serial Clock (SCK), and Serial Input (SI) - SI for single bit wide transfers. The memory drives Serial Output (SO) 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.

## 9.1 Command Set Summary

### 9.1.1 Extended Addressing

To accommodate addressing above 128 Mb, there are three options:

1. New instructions are provided with 4-byte address, used to access up to 32 Gb of memory.

| Instruction Name | Description | Code (Hex) |
|---|---|---|
| 4FAST_READ | Read Fast (4-byte Address) | 0C |
| 4READ | Read (4-byte Address) | 13 |
| 4DOR | Read Dual Out (4-byte Address) | 3C |
| 4QOR | Read Quad Out (4-byte Address) | 6C |
| 4DIOR | Dual I/O Read (4-byte Address) | BC |
| 4QIOR | Quad I/O Read (4-byte Address) | EC |
| 4PP | Page Program (4-byte Address) | 12 |
| 4QPP | Quad Page Program (4-byte Address) | 34 |
| 4P4E | Parameter 4-kB Erase (4-byte Address) | 21 |
| 4SE | Erase 64/256 kB (4-byte Address) | DC |

2. For backward compatibility to the 3-byte address instructions, the standard instructions can be used in conjunction with the EXTADD Bit in the Bank Address Register (BAR[7]). By default BAR[7] is cleared to 0 (following power up and hardware reset), to enable 3-byte (24-bit) addressing. When 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 EXTADD bit to switch from 3 bytes to 4 bytes of address field.

| Instruction Name | Description | Code (Hex) |
|---|---|---|
| READ | Read (3-byte Address) | 03 |
| FAST_READ | Read Fast (3-byte Address) | 0B |
| DOR | Read Dual Out (3-byte Address) | 3B |
| QOR | Read Quad Out (3-byte Address) | 6B |
| DIOR | Dual I/O Read (3-byte Address) | BB |
| QIOR | Quad I/O Read (3-byte Address) | EB |
| PP | Page Program (3-byte Address) | 02 |
| QPP | Quad Page Program (3-byte Address) | 32 |
| P4E | Parameter 4-kB Erase (3-byte Address) | 20 |
| SE | Erase 64 / 256 kB (3-byte Address) | D8 |

3. For backward compatibility to the 3-byte addressing, the standard instructions can be used in conjunction with the Bank Address Register:
   a. The Bank Address Register is used to switch between 128-Mbit (16-Mbyte) banks of memory, The standard 3-byte address selects an address within the bank selected by the Bank Address Register.
      i. The host system writes the Bank Address Register to access beyond the first 128 Mbits of memory.
      ii. This applies to read, erase, and program commands.
   b. The Bank Register provides the high order (4th) byte of address, which is used to address the available memory at addresses greater than 16 Mbytes.
   c. Bank Register bits are volatile.
      i. On power up, the default is Bank0 (the lowest address 16 Mbytes).
   d. For Read, the device will continuously transfer out data until the end of the array.
      i. There is no bank to bank delay.
      ii. The Bank Address Register is not updated.
      iii. The Bank Address Register value is used only for the initial address of an access.

**Table 35. Bank Address Map**

| Bank Address Register Bits | | Bank | Memory Array Address Range (Hex) | |
|---|---|---|---|---|
| Bit 1 | Bit 0 | | | |
| 0 | 0 | 0 | 00000000 | 00FFFFFF |

## 9.1.2 Command Summary Sorted by Function

**Table 36. S25FL127S Command Set (sorted by function)**

| Function | Command Name | Command Description | Instruction Value (Hex) | Maximum Frequency (MHz) |
|---|---|---|---|---|
| Read Device Identification | READ_ID (REMS) | Read Electronic Manufacturer Signature | 90 | 108 |
| | RDID | Read ID (JEDEC Manufacturer ID and JEDEC CFI) | 9F | 108 |
| | RSFDP | Read JEDEC Serial Flash Discoverable Parameters | 5A | 108 |
| | RES | Read Electronic Signature | AB | 50 |

**Table 36. S25FL127S Command Set (sorted by function) (Continued)**

| Function | Command Name | Command Description | Instruction Value (Hex) | Maximum Frequency (MHz) |
|---|---|---|---|---|
| Register Access | RDSR1 | Read Status Register 1 | 05 | 108 |
| | RDSR2 | Read Status Register 2 | 07 | 108 |
| | RDCR | Read Configuration Register 1 | 35 | 108 |
| | WRR | Write Register (Status-1, Configuration-1) | 01 | 108 |
| | WRDI | Write Disable | 04 | 108 |
| | WREN | Write Enable | 06 | 108 |
| | CLSR | Clear Status Register 1 - Erase/Prog. Fail Reset | 30 | 108 |
| | ECCRD | ECC Read (4-byte address) | 18 | 108 |
| | ABRD | AutoBoot Register Read | 14 | 108 (QUAD=0) 108 (QUAD=1) |
| | ABWR | AutoBoot Register Write | 15 | 108 |
| | BRRD | Bank Register Read | 16 | 108 |
| | BRWR | Bank Register Write | 17 | 108 |
| | BRAC | Bank Register Access (Legacy Command formerly used for Deep Power Down) | B9 | 108 |
| | DLPRD | Data Learning Pattern Read | 41 | 108 |
| | PNVDLR | Program NV Data Learning Register | 43 | 108 |
| | WVDLR | Write Volatile Data Learning Register | 4A | 108 |
| Read Flash Array | READ | Read (3- or 4-byte address) | 03 | 50 |
| | 4READ | Read (4-byte address) | 13 | 50 |
| | FAST_READ | Fast Read (3- or 4-byte address) | 0B | 108 |
| | 4FAST_READ | Fast Read (4-byte address) | 0C | 108 |
| | DOR | Read Dual Out (3- or 4-byte address) | 3B | 108 |
| | 4DOR | Read Dual Out (4-byte address) | 3C | 108 |
| | QOR | Read Quad Out (3- or 4-byte address) | 6B | 108 |
| | 4QOR | Read Quad Out (4-byte address) | 6C | 108 |
| | DIOR | Dual I/O Read (3- or 4-byte address) | BB | 108 |
| | 4DIOR | Dual I/O Read (4-byte address) | BC | 108 |
| | QIOR | Quad I/O Read (3- or 4-byte address) | EB | 108 |
| | 4QIOR | Quad I/O Read (4-byte address) | EC | 108 |
| Program Flash Array | PP | Page Program (3- or 4-byte address) | 02 | 108 |
| | 4PP | Page Program (4-byte address) | 12 | 108 |
| | QPP | Quad Page Program (3- or 4-byte address) | 32 | 80 |
| | QPP | Quad Page Program - Alternate instruction (3- or 4-byte address) | 38 | 80 |
| | 4QPP | Quad Page Program (4-byte address) | 34 | 80 |
| | PGSP | Program Suspend | 85 | 108 |
| | PGRS | Program Resume | 8A | 108 |

**Table 36. S25FL127S Command Set (sorted by function) (Continued)**

| Function | Command Name | Command Description | Instruction Value (Hex) | Maximum Frequency (MHz) |
|---|---|---|---|---|
| Erase Flash Array | P4E | Parameter 4-kB, sector Erase (3- or 4-byte address) | 20 | 108 |
| | 4P4E | Parameter 4-kB, sector Erase (4-byte address) | 21 | 108 |
| | BE | Bulk Erase | 60 | 108 |
| | BE | Bulk Erase (alternate command) | C7 | 108 |
| | SE | Erase 64 kB or 256 kB (3- or 4-byte address) | D8 | 108 |
| | 4SE | Erase 64 kB or 256 kB (4-byte address) | DC | 108 |
| | ERSP | Erase Suspend | 75 | 108 |
| | ERRS | Erase Resume | 7A | 108 |
| One Time Program Array | OTPP | OTP Program | 42 | 108 |
| | OTPR | OTP Read | 4B | 108 |
| Advanced Sector Protection | DYBRD | DYB Read | E0 | 108 |
| | DYBWR | DYB Write | E1 | 108 |
| | PPBRD | PPB Read | E2 | 108 |
| | PPBP | PPB Program | E3 | 108 |
| | PPBE | PPB Erase | E4 | 108 |
| | ASPRD | ASP Read | 2B | 108 |
| | ASPP | ASP Program | 2F | 108 |
| | PLBRD | PPB Lock Bit Read | A7 | 108 |
| | PLBWR | PPB Lock Bit Write | A6 | 108 |
| | PASSRD | Password Read | E7 | 108 |
| | PASSP | Password Program | E8 | 108 |
| | PASSU | Password Unlock | E9 | 108 |
| Reset | RESET | Software Reset | F0 | 108 |
| | MBR | Mode Bit Reset | FF | 108 |
| RFU | Reserved-A3 | Reserved | A3 | 108 |
| RFU | Reserved-18 | Reserved | 18 | |
| RFU | Reserved-E5 | Reserved | E5 | |
| RFU | Reserved-E6 | Reserved | E6 | |

### 9.1.3 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 FL-S Family supports the three most common device information commands.

### 9.1.4 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.

#### 9.1.4.1 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 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 1 indicating the device remains busy and unable to receive most new operation commands. Only status read (RDSR1 05h), status clear (CLSR 30h), write disable (WRDI 04h), and software reset (RESET 0Fh) 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. CLSR clears the WIP, P_ERR, and E_ERR bits. WRDI clears the WEL bit. Alternatively, hardware reset, or software reset (RESET) may be used to return the device to standby state.

#### 9.1.4.2 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.

### 9.1.5 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 0 of the array.

There are several different read commands to specify different access latency and data path widths.

- The Read command provides a single address bit per SCK rising edge on the SI signal with read data returning a single bit per SCK falling edge on the SO signal. This command has zero latency between the address and the returning data but is limited to a maximum SCK rate of 50 MHz.
- Other read commands have a latency period between the address and returning data but can operate at higher SCK frequencies. The latency depends on the configuration register latency code.
- The Fast Read command provides a single address bit per SCK rising edge on the SI signal with read data returning a single bit per SCK falling edge on the SO signal and may operate up to 108 MHz.
- Dual or Quad Output read commands provide address a single bit per SCK rising edge on the SI / IO0 signal 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.

### 9.1.6 Program Flash Array

Programming data requires two commands: Write Enable (WREN), and Page Program (PP or QPP). The Page Program command 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.

### 9.1.7 Erase Flash Array

The Sector Erase (SE) and Bulk Erase (BE) 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 (SE) or array-wide (BE) level. The Write Enable (WREN) command must precede an erase command.

### 9.1.8 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.

### 9.1.9 Reset

There is a command to reset to the default conditions present after power on to the device. There is a command to reset (exit from) the Enhanced Performance Read Modes.

### 9.1.10 Reserved

Some instructions are reserved for future use. In this generation of the FL-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 affect. This allows legacy software to issue some commands that are not relevant for the current generation FL-S Family with the assurance these commands do not cause some unexpected action.

Some commands are reserved for use in special versions of the FL-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.

## 9.2 Identification Commands

### 9.2.1 Read Identification - REMS (Read_ID or REMS 90h)

The READ_ID command identifies the Device Manufacturer ID and the Device ID. The command is also referred to as Read Electronic Manufacturer and device Signature (REMS). READ-ID (REMS) is only supported for backward compatibility and should not be used for new software designs. New software designs should instead make use of the RDID command.

The command is initiated by shifting on SI the instruction code “90h” followed by a 24-bit address of 00000h. Following this, the Manufacturer ID and the Device ID are shifted out on SO starting at the falling edge of SCK after address. The Manufacturer ID and the Device ID are always shifted out with the MSB first. If the 24-bit address is set to 000001h, then the Device ID is read out first followed by the Manufacturer ID. The Manufacturer ID and Device ID output data toggles between address 000000H and 000001H until terminated by a low to high transition on CS# input. The maximum clock frequency for the READ_ID command is 108 MHz.

**Figure 46. READ_ID Command Sequence**

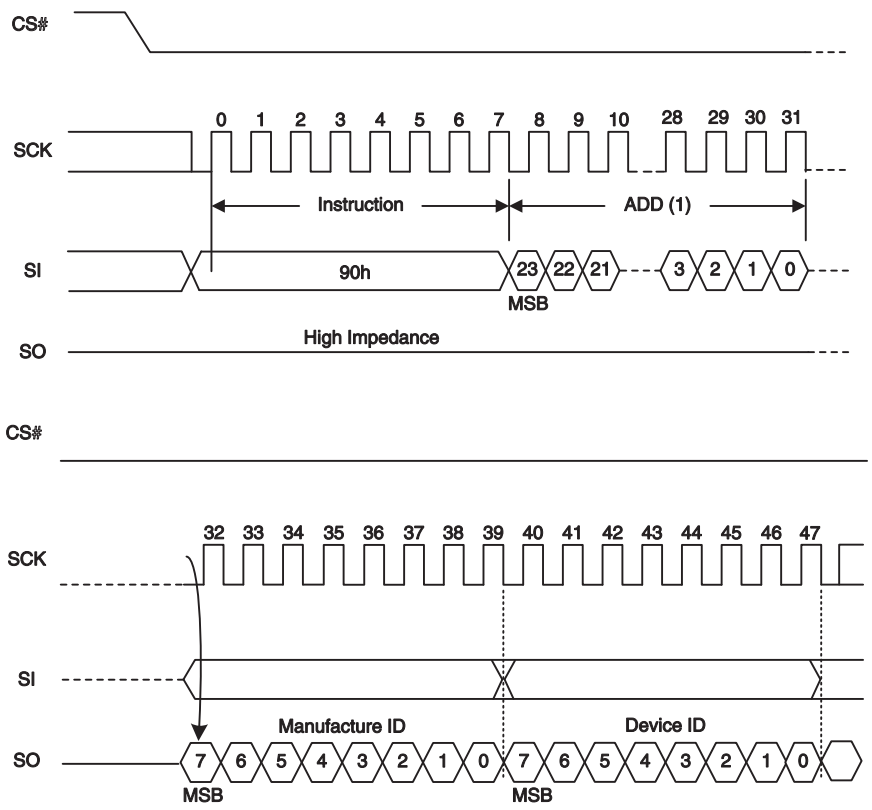



**Table 37. Read_ID Values**

| Device | Manufacturer ID (hex) | Device ID (hex) |
|---|---|---|
| S25FL127S | 01 | 17 |

## 9.2.2 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.

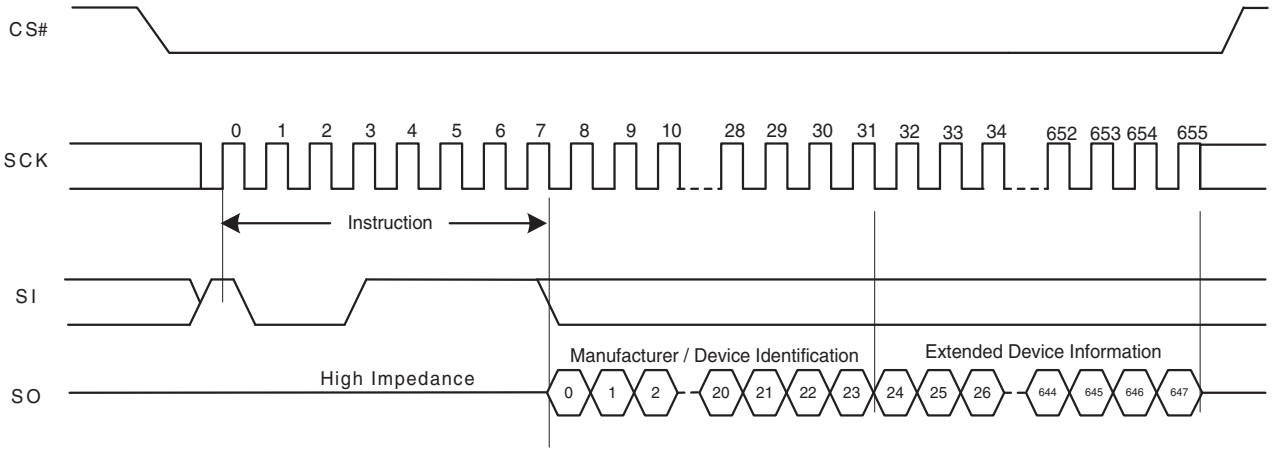
Any RDID command issued while a program, erase, or write cycle is in progress is ignored and has no effect on execution of the program, erase, or write cycle that is in progress.

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 ID-CFI Address Space on page 48 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 108 MHz.

**Figure 47. Read Identification (RDID) Command Sequence**




## 9.2.3 Read Electronic Signature (RES) (ABh)

The RES command is used to read a single byte Electronic Signature from SO. RES is only supported for backward compatibility and should not be used for new software designs. New software designs should instead make use of the RDID command.

The RES instruction is shifted in followed by three dummy bytes onto SI. After the last bit of the three dummy bytes are shifted into the device, a byte of Electronic Signature will be shifted out of SO. Each bit is shifted out by the falling edge of SCK. The maximum clock frequency for the RES command is 50 MHz.

The Electronic Signature can be read repeatedly by applying multiples of eight clock cycles.

The RES command sequence is terminated by driving CS# to the logic high state anytime during data output.

**Figure 48. Read Electronic Signature (RES) Command Sequence**

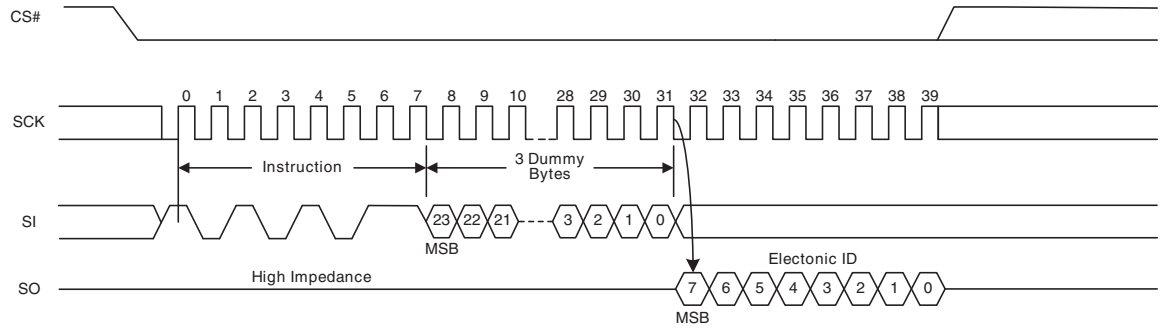


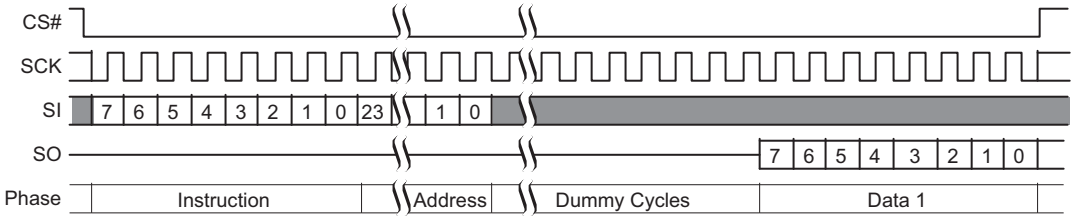

**Table 38. RES Values**

| Device | Device ID (hex) |
|---|---|
| S25FL127S | 17 |

### 9.2.4 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 eight dummy cycles. The SFDP bytes are then shifted out on SO starting at the falling edge of SCK after the eight 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 maximum clock frequency for the RSFDP command is 108 MHz.

**Figure 49. RSFDP Command Sequence**

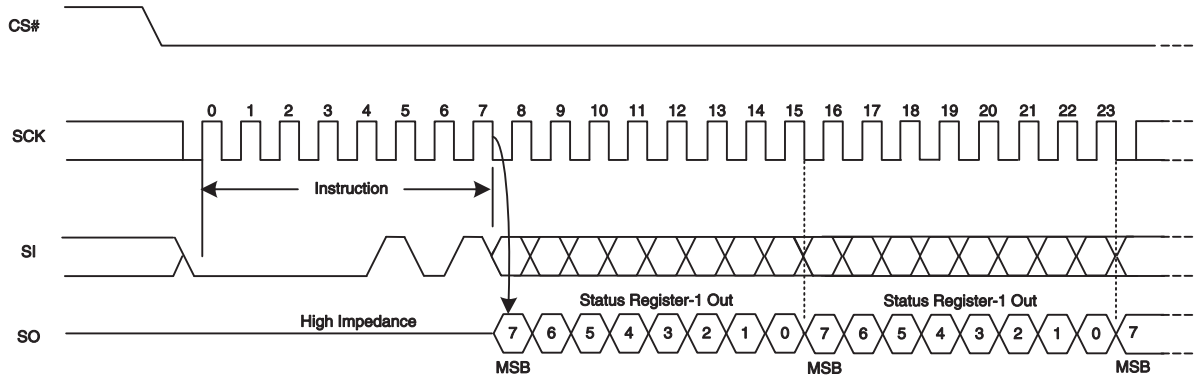



## 9.3 Register Access Commands

### 9.3.1 Read Status Register 1 (RDSR1 05h)

The Read Status Register 1 (RDSR1) command allows the Status Register 1 contents to be read from SO. The Status Register 1 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 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 108 MHz.

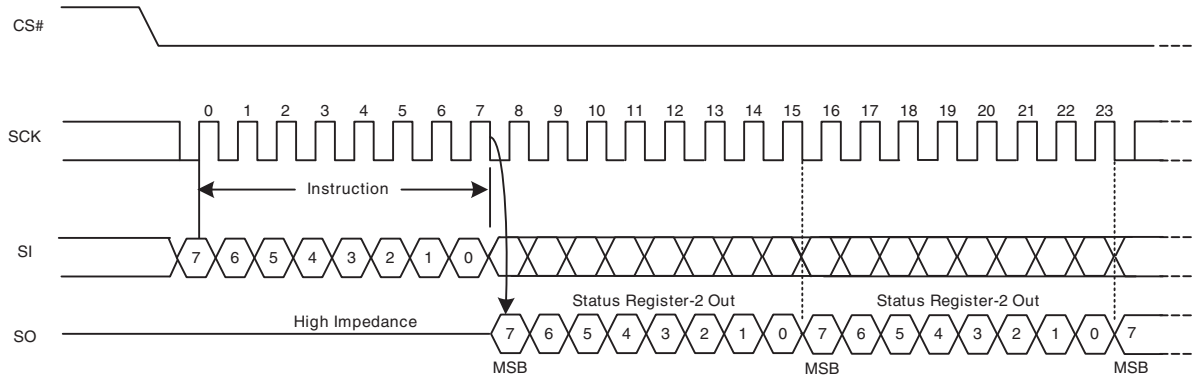
**Figure 50. Read Status Register 1 (RDSR1) Command Sequence**



### 9.3.2 Read Status Register 2 (RDSR2 07h)

The Read Status Register (RDSR2) command allows the Status Register 2 contents to be read from SO. 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 108 MHz.

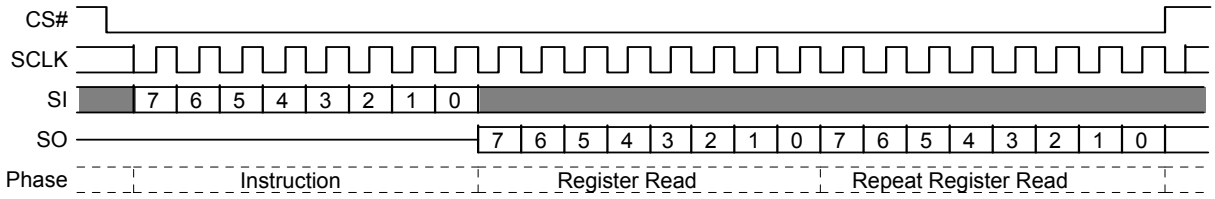
**Figure 51. Read Status Register 2 (RDSR2) Command**




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

The Read Configuration Register (RDCR) command allows the Configuration Register contents to be read from SO. It is possible to read the Configuration Register 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.

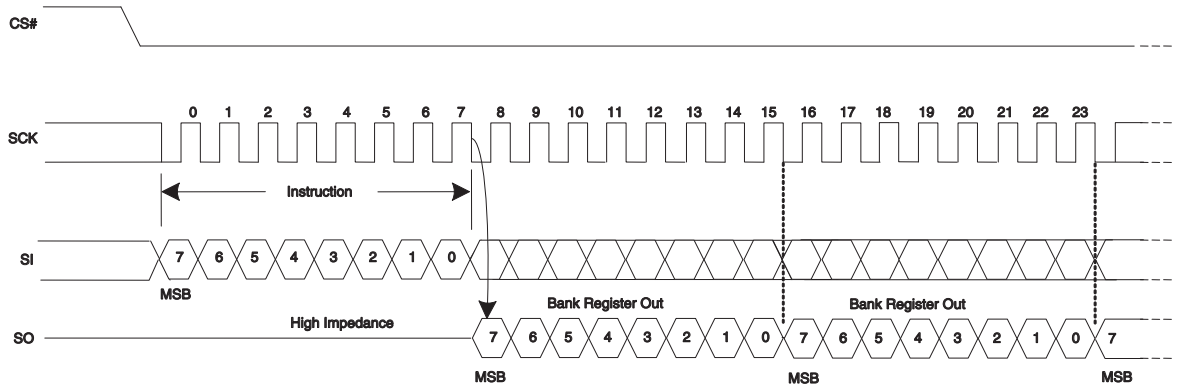
**Figure 52. Read Configuration Register (RDCR) Command Sequence**



### 9.3.4 Bank Register Read (BRRD 16h)

The Read the Bank Register (BRRD) command allows the Bank address Register contents to be read from SO. The instruction is first shifted in from SI. Then the 8-bit Bank Register is shifted out on SO. It is possible to read the Bank Register continuously by providing multiples of eight clock cycles. The maximum operating clock frequency for the BRRD command is 108 MHz.

**Figure 53. Read Bank Register (BRRD) Command**



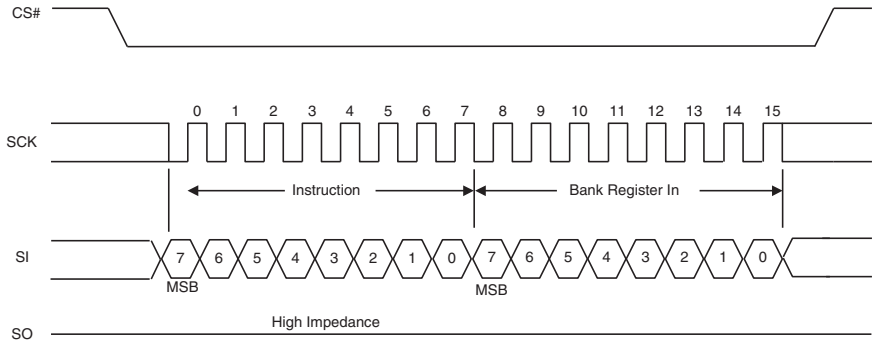

### 9.3.5 Bank Register Write (BRWR 17h)

The Bank Register Write (BRWR) command is used to write address bits above A23, into the Bank Address Register (BAR). The command is also used to write the Extended address control bit (EXTADD) that is also in BAR[7]. BAR provides the high order addresses needed by devices having more than 128 Mbits (16 Mbytes), when using 3-byte address commands without extended addressing enabled (BAR[7] EXTADD = 0). Because this command is part of the addressing method and is not changing data in the flash memory, this command does not require the WREN command to precede it.

The BRWR instruction is entered, followed by the data byte on SI. The Bank Register is one data byte in length.

The BRWR command has no effect on the P_ERR, E_ERR or WIP bits of the Status and Configuration Registers. Any bank address bit reserved for the future should always be written as a 0.

**Figure 54. Bank Register Write (BRWR) Command**



### 9.3.6 Bank Register Access (BRAC B9h)

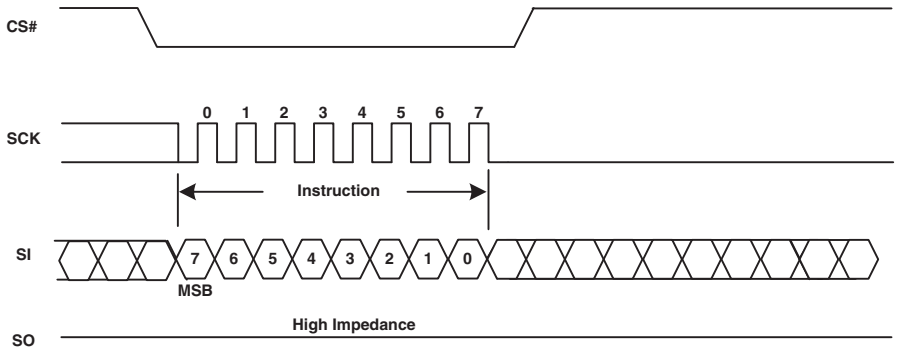
The Bank Register Read and Write commands provide full access to the Bank Address Register (BAR) but they are both commands that are not present in legacy SPI memory devices. Host system SPI memory controller interfaces may not be able to easily support such new commands. The Bank Register Access (BRAC) command uses the same command code and format as the Deep Power Down (DPD) command that is available in legacy SPI memories. The FL-S family does not support a DPD feature but assigns this legacy command code to the BRAC command to enable write access to the Bank Address Register for legacy systems that are able to send the legacy DPD (B9h) command.

When the BRAC command is sent, the FL-S family device will then interpret an immediately following Write Register (WRR) command as a write to the lower address bits of the BAR. A WREN command is not used between the BRAC and WRR commands. Only the lower two bits of the first data byte following the WRR command code are used to load BAR[1:0]. The upper bits of that byte and the content of the optional WRR command second data byte are ignored. Following the WRR command the access to BAR is closed and the device interface returns to the standby state. The combined BRAC followed by WRR command sequence has no affect on the value of the ExtAdd bit (BAR[7]).

Commands other than WRR may immediately follow BRAC and execute normally. However, any command other than WRR, or any other sequence in which CS# goes low and returns high, following a BRAC command, will close the access to BAR and return to the normal interpretation of a WRR command as a write to Status Register 1 and the Configuration Register.

The BRAC + WRR sequence is allowed only when the device is in standby, program suspend, or erase suspend states. This command sequence is illegal when the device is performing an embedded algorithm or when the program (P_ERR) or erase (E_ERR) status bits are set to 1.

**Figure 55. BRAC (B9h) Command Sequence**




### 9.3.7 Write Registers (WRR 01h)

The Write Registers (WRR) command allows new values to be written to Status Register 1, Configuration Register, and Status Register 2. 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.

The Write Registers (WRR) command will set the P_ERR or E_ERR bits if there is a failure in the WRR operation. Any Status or Configuration Register bit that is:

- reserved for the future, must be written as a 0
- read only, is not affected by the value written to that bit
- OTP, may be written to 1 but, if the value is already 1, it cannot be written to 0, it will remain 1.

CS# must be driven to the logic high state after the eighth, sixteenth, or twenty-fourth 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; after the sixteenth cycle both the Status and Configuration Registers are written; after the twenty fourth cycle the Status Register 1, Configuration Register, and Status Register 2 are written. When the configuration register QUAD bit CR[1] is 1, only the WRR command formats with 16 or 24 data bits may be used.

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 only changing the value of volatile bits, the operation is completed in $t_{CS}$ time (the WIP bit will be 0 before a status read can be completed). When changing non-volatile bits, the WRR operation is completed in $t_W$ time. 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 108 MHz.

**Figure 56. Write Registers (WRR) Command Sequence – 8 data bits**

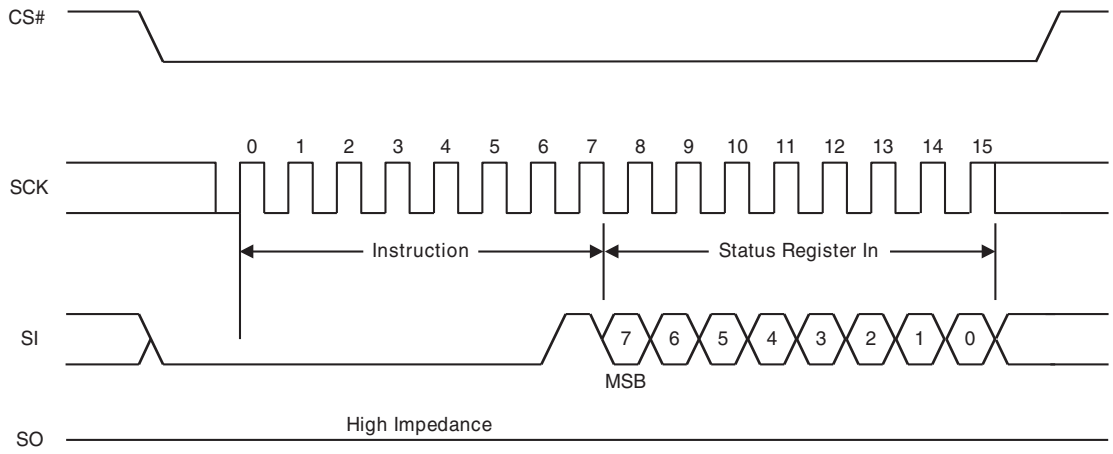



**Figure 57. Write Registers (WRR) Command Sequence – 16 data bits**

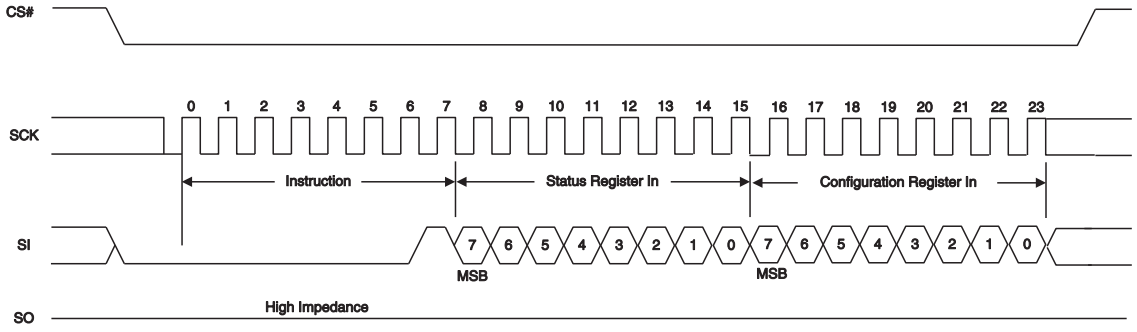
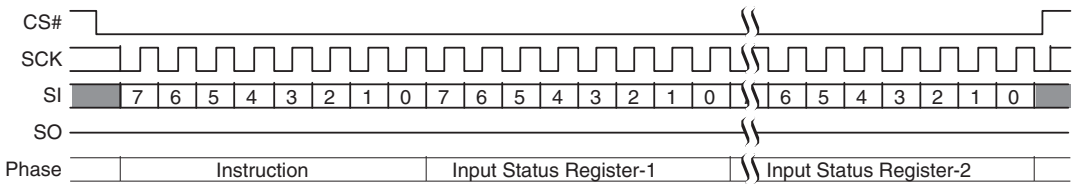



**Figure 58. Write Registers (WRR 01h) Command Sequence – 24 data bits**




The Write Registers (WRR) command allows the user to change the values of the Block Protect (BP2, BP1, and BP0) bits to define the size of the area that is to be treated as read-only. 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, and are not accepted for execution. 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 39. 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 and Status Register 2 can be changed. | Protected against Page Program, Quad Input Program, Sector Erase, and Bulk Erase | Ready to accept Page Program, Quad Input 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 and Status Register 2 cannot be changed. | Protected against Page Program, Sector Erase, and Bulk Erase | Ready to accept Page Program or Erase commands |

***Notes:***

1. *The Status Register originally shows 00h when the device is first shipped from Cypress to the customer.*
2. *Hardware protection is disabled when Quad Mode is enabled (QUAD bit = 1 in Configuration Register). WP# becomes IO2; therefore, it cannot be utilized.*

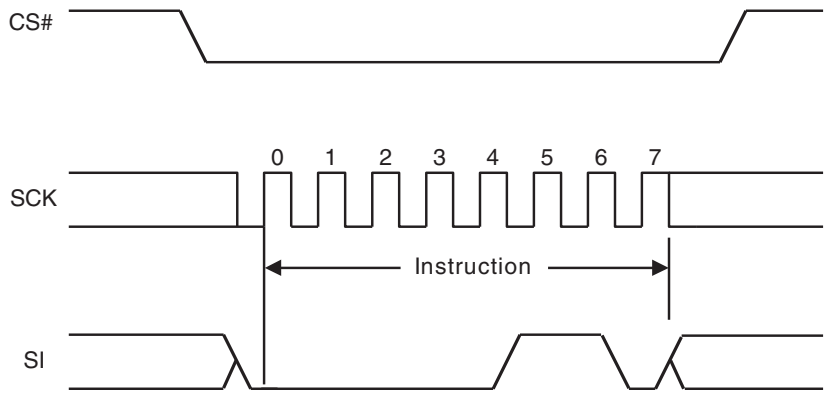
The WRR command has an alternate function of loading the Bank Address Register if the command immediately follows a BRAC command. See Bank Register Access (BRAC B9h) on page 78.

### 9.3.8 Write Enable (WREN 06h)

The Write Enable (WREN) command sets the Write Enable Latch (WEL) bit of the Status Register 1 (SR1[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. Without CS# being driven to the logic high state after the eighth bit of the instruction byte has been latched in on SI, the write enable operation will not be executed.

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



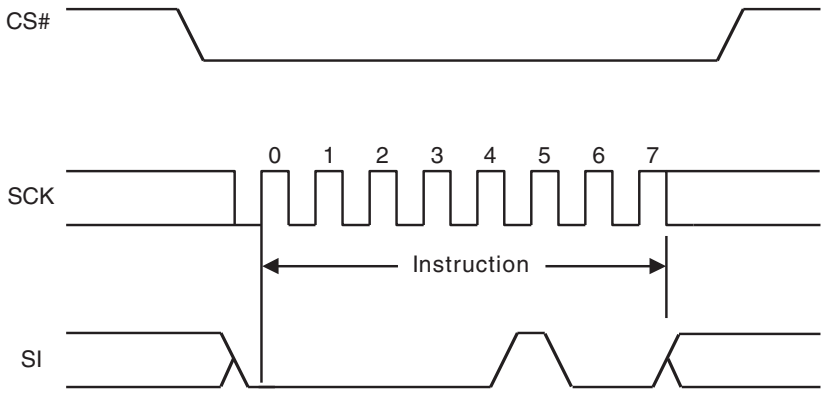

### 9.3.9 Write Disable (WRDI 04h)

The Write Disable (WRDI) command sets the Write Enable Latch (WEL) bit of the Status Register 1 (SR1[1]) to a 0.

The Write Enable Latch (WEL) bit may be set to a 0 by issuing the Write Disable (WRDI) command to disable Page Program (PP), Sector Erase (SE), Bulk Erase (BE), Write Registers (WRR), 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. Without CS# being driven to the logic high state after the eighth bit of the instruction byte has been latched in on SI, the write disable operation will not be executed.

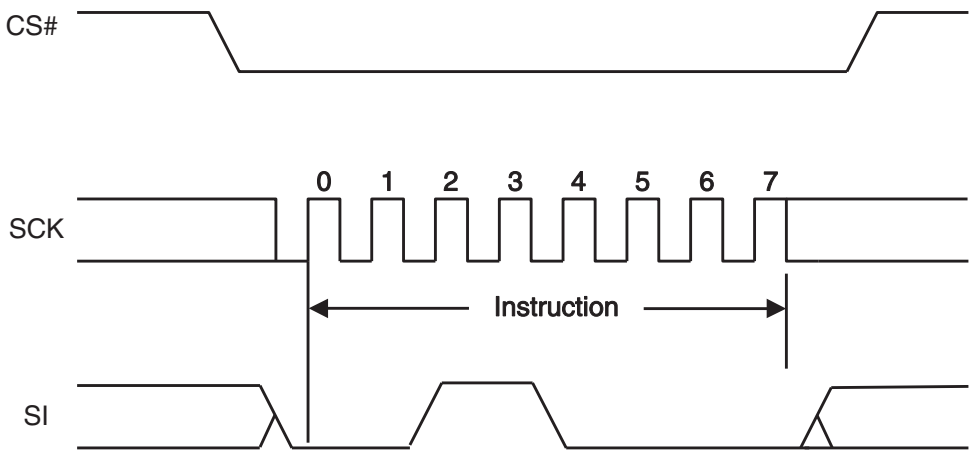
**Figure 60. Write Disable (WRDI) Command Sequence**



### 9.3.10 Clear Status Register (CLSR 30h)

The Clear Status Register command resets bit SR1[5] (Erase Fail Flag) and bit SR1[6] (Program Fail Flag). It is not necessary to set the WEL bit before the Clear SR command is executed. The Clear SR 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.

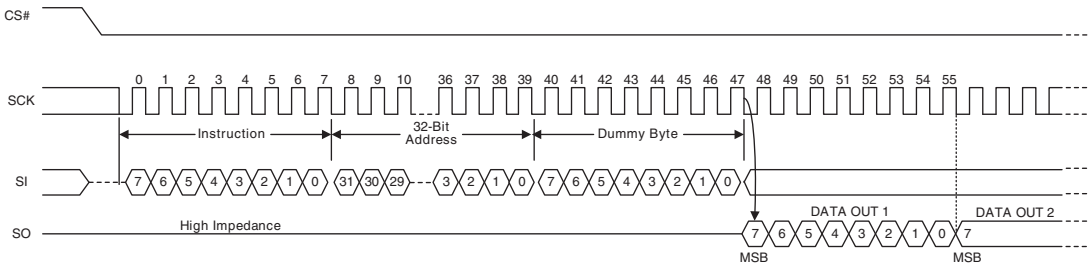
**Figure 61. Clear Status Register (CLSR) Command Sequence**




### 9.3.11 ECC Status Register Read (ECCRD 18h)

To read the ECC Status Register, the command is followed by the ECC unit (32 bit) address, the four least significant bits (LSB) of address must be set to zero. This is followed by eight dummy cycles. Then the 8-bit contents of the ECC Register, for the ECC unit selected, are shifted out on SO 16 times, once for each byte in the ECC Unit. If CS# remains low the next ECC unit status is sent through SO 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 Section 9.5.1.1 Automatic ECC on page 93 for details on ECC unit.

**Figure 62. ECC Status Register Read Command Sequence**



### 9.3.12 AutoBoot

SPI devices normally require 32 or more cycles of command and address shifting to initiate a read command. And, in order to read boot code from an SPI device, the host memory controller or processor must supply the read command from a hardwired state machine or from some host processor internal ROM code.

Parallel NOR devices need only an initial address, supplied in parallel in a single cycle, and initial access time to start reading boot code.

The AutoBoot feature allows the host memory controller to take boot code from an FL-S Family device immediately after the end of reset, without having to send a read command. This saves 32 or more cycles and simplifies the logic needed to initiate the reading of boot code.

- As part of the power up reset, hardware reset, or command reset process the AutoBoot feature automatically starts a read access from a pre-specified address. At the time the reset process is completed, the device is ready to deliver code from the starting address. The host memory controller only needs to drive CS# signal from high to low and begin toggling the SCK signal. The FL-S Family device will delay code output for a pre-specified number of clock cycles before code streams out.
  - The Auto Boot Start Delay (ABSD) field of the AutoBoot register specifies the initial delay if any is needed by the host.
  - The host cannot send commands during this time.
  - If ABSD = 0, the maximum SCK frequency is 50 MHz.
  - If ABSD > 0, the maximum SCK frequency is 108 MHz if the QUAD bit CR1[1] is 0 or 108 MHz if the QUAD bit is set to 1.
- The starting address of the boot code is selected by the value programmed into the AutoBoot Start Address (ABSA) field of the AutoBoot Register which specifies a 512-byte boundary aligned location; the default address is 00000000h.
  - Data will continuously shift out until CS# returns high.
- At any point after the first data byte is transferred, when CS# returns high, the SPI device will reset to standard SPI mode; able to accept normal command operations.
  - A minimum of 1 byte must be transferred.
  - AutoBoot mode will not initiate again until another power cycle or a reset occurs.
- An AutoBoot Enable bit (ABE) is set to enable the AutoBoot feature.

The AutoBoot register bits are non-volatile and provide:

- The starting address (512-byte boundary), set by the AutoBoot Start Address (ABSA). The size of the ABSA field is 23 bits for devices up to 32-Gbit.
- The number of initial delay cycles, set by the AutoBoot Start Delay (ABSD) 8-bit count value.
- The AutoBoot Enable.

If the configuration register QUAD bit CR1[1] is set to 1, the boot code will be provided 4 bits per cycle in the same manner as a Read Quad Out command. If the QUAD bit is 0 the code is delivered serially in the same manner as a Read command.

**Figure 63. AutoBoot Sequence (CR1[1]=0)**

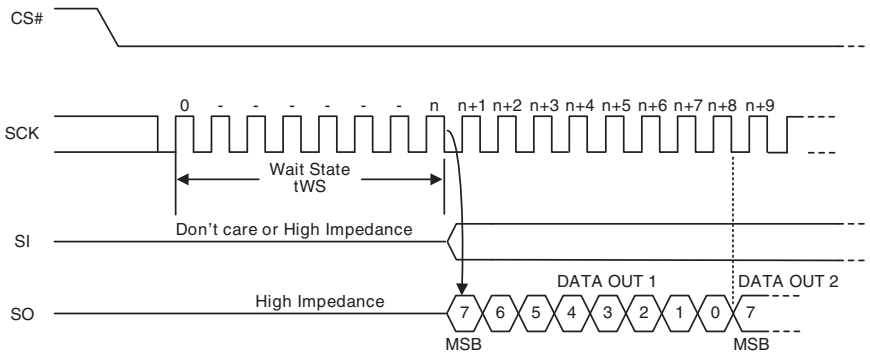
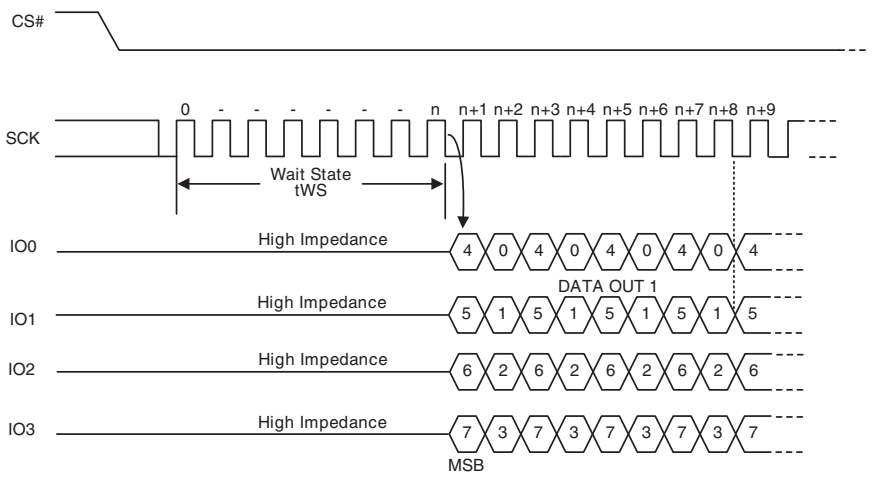


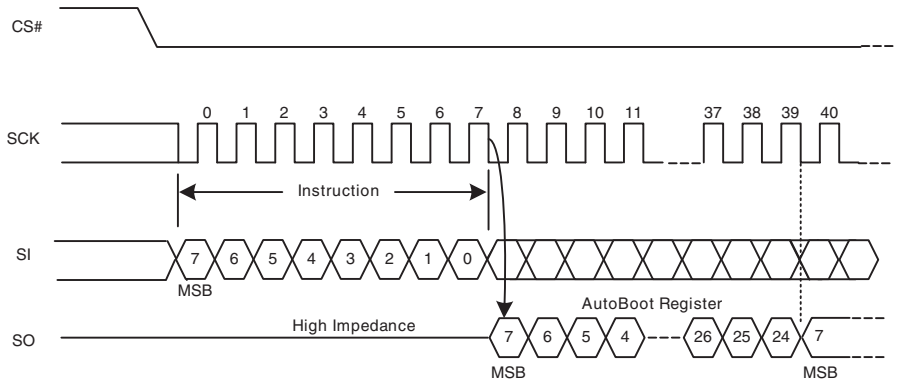
**Figure 64. AutoBoot Sequence (CR1[1]=1)**




## 9.3.13 AutoBoot Register Read (ABRD 14h)

The AutoBoot Register Read command is shifted into SI. Then the 32-bit AutoBoot Register is shifted out on SO, least significant byte first, most significant bit of each byte first. It is possible to read the AutoBoot Register continuously by providing multiples of 32 clock cycles. If the QUAD bit CR1[1] is cleared to 0, the maximum operating clock frequency for ABRD command is 108 MHz. If the QUAD bit CR1[1] is set to 1, the maximum operating clock frequency for ABRD command is 108 MHz.

**Figure 65. AutoBoot Register Read (ABRD) Command**




## 9.3.14 AutoBoot Register Write (ABWR 15h)

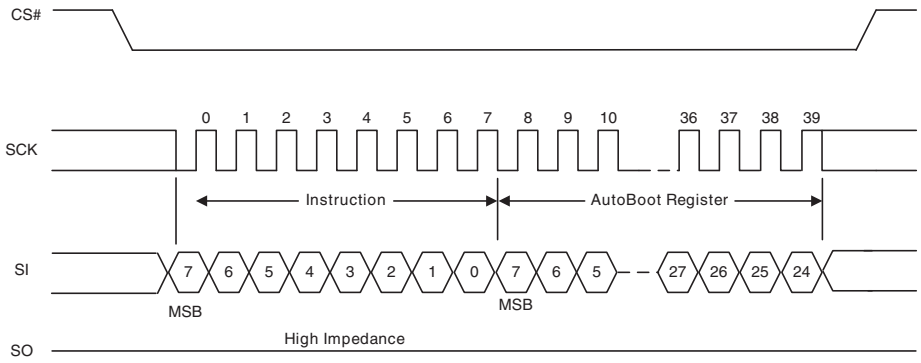
Before the ABWR command can be accepted, a Write Enable (WREN) command must be issued and decoded by the device, which sets the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The ABWR command is entered by shifting the instruction and the data bytes on SI, least significant byte first, most significant bit of each byte first. The ABWR data is 32 bits in length.

The ABWR command has status reported in Status Register 1 as both an erase and a programming operation. An E_ERR or a P_ERR may be set depending on whether the erase or programming phase of updating the register fails.

CS# must be driven to the logic high state after the 32nd bit of data has been latched. If not, the ABWR command is not executed. As soon as CS# is driven to the logic high state, the self-timed ABWR operation is initiated. While the ABWR operation is in progress, Status Register 1 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 ABWR operation, and is a 0 when it is completed. When the ABWR cycle is completed, the Write Enable Latch (WEL) is set to a 0. The maximum clock frequency for the ABWR command is 108 MHz.

**Figure 66. AutoBoot Register Write (ABWR) Command**




## 9.4 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 one bit per rising edge of SCK and return data 1, 2, or 4 bits of data per rising edge of SCK. These are called Read or Fast Read for 1 bit data; Dual Output Read for 2 bit data, and Quad Output for 4 bit data.
- Some commands transfer both address and data 2 or 4 bits per rising edge of SCK. These are called Dual I/O for 2 bit and Quad I/O for 4 bit.

All of these commands begin with an instruction code that is transferred one bit per SCK rising edge. The instruction is followed by either a 3- or 4-byte address. Commands transferring address or data 2 or 4 bits per clock edge are called Multiple I/O (MIO) commands. For FL-S Family devices at 256 Mbits or higher density, the traditional SPI 3-byte addresses are unable to directly address all locations in the memory array. These device have a bank address register that is used with 3-byte address commands to supply the high order address bits beyond the address from the host system. The default bank address is 0. Commands are provided to load and read the bank address register. These devices may also 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 External Address (EXTADD) bit in the bank address register to 1. In the FL127S, higher order address bits above A23 in the 4-byte address commands, commands using Extended Address mode, and the Bank Address Register are not relevant and are ignored because the flash array is only 128 Mbits in size.

The Quad I/O 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 I/O read accesses.

Some commands require delay cycles following the address or mode bits to allow time to access the memory array. The delay 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 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. The number of dummy cycles varies with the SCK frequency or performance option selected via the Configuration Register 1 (CR1) Latency Code (LC). 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.

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 enhanced high performance read mode.

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

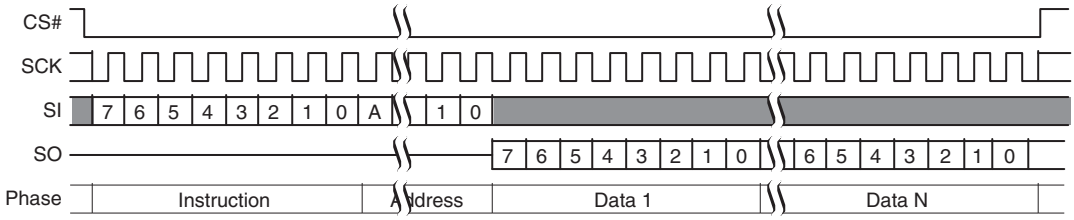
The instruction

- 03h (ExtAdd=0) is followed by a 3-byte address (A23-A0) or
- 03h (ExtAdd=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. 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 67. Read Command Sequence (READ 03h or 13h)**




***Note:***
*A = MSB of address = 23 for ExtAdd=0, or 31 for ExtAdd=1 or command 13h.*

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

The instruction

- 0Bh (ExtAdd=0) is followed by a 3-byte address (A23-A0) or
- 0Bh (ExtAdd=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 zero or eight dummy cycles depending on the latency code set in the Configuration Register. 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 is “don’t care” and may be high impedance. Then the memory contents, at the address given, are shifted out on SO.

The maximum operating clock frequency for FAST READ command is 108 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 68. Fast Read (FAST_READ 0Bh or 0Ch) Command Sequence with Read Latency**

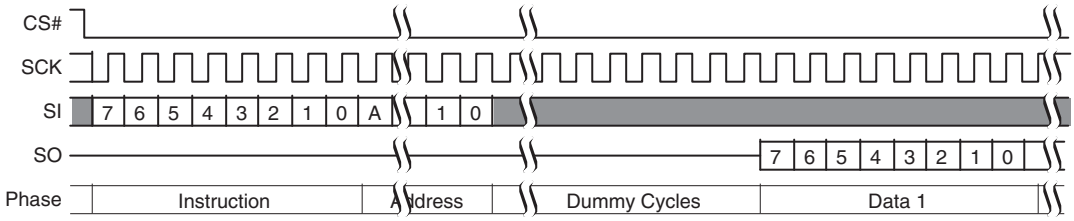
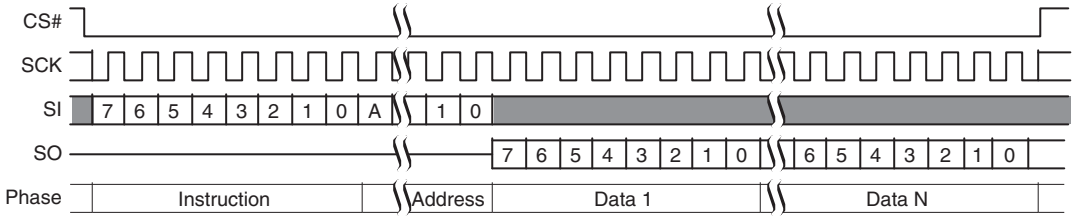


**Figure 69. Fast Read Command (FAST_READ 0Bh or 0Ch) Sequence without Read Latency**




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

The instruction

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

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 maximum operating clock frequency for the Dual Output Read command is 108 MHz. For Dual Output Read commands, there are zero or eight dummy cycles required after the last address bit is shifted into SI before data begins shifting out of IO0 and IO1. This latency period (i.e., dummy cycles) allows the device's internal circuitry enough time to read from the initial address. During the dummy cycles, the data value on SI is a "don't care" and may be high impedance. The number of dummy cycles is determined by the frequency of SCK (refer to Table 22, Latency Codes on page 53).

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 70. Dual Output Read Command Sequence (3-Byte Address, 3Bh [ExtAdd=0], LC=10b)**

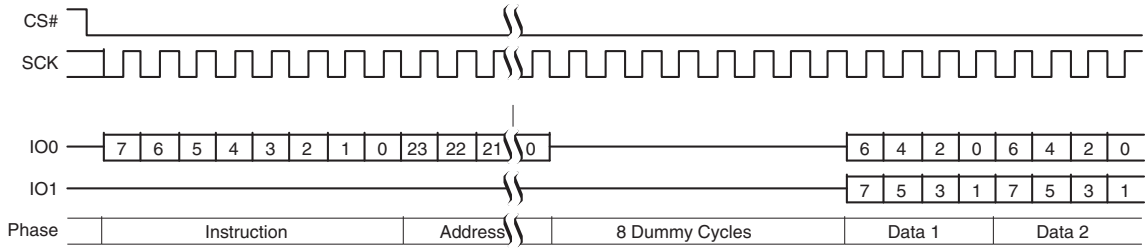



**Figure 71. Dual Output Read Command Sequence (4-Byte Address, 3Ch or 3Bh [ExtAdd=1, LC=10b])**

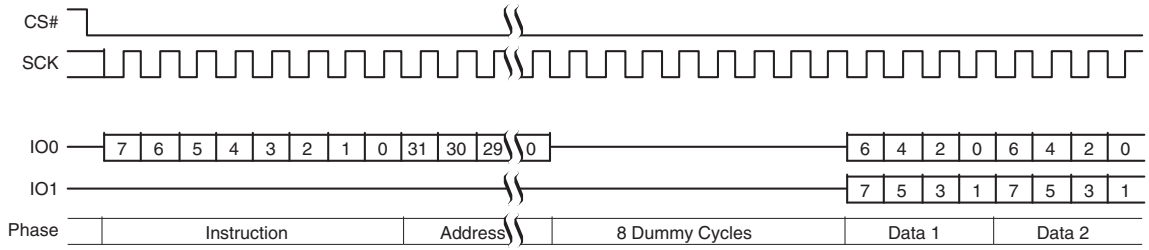
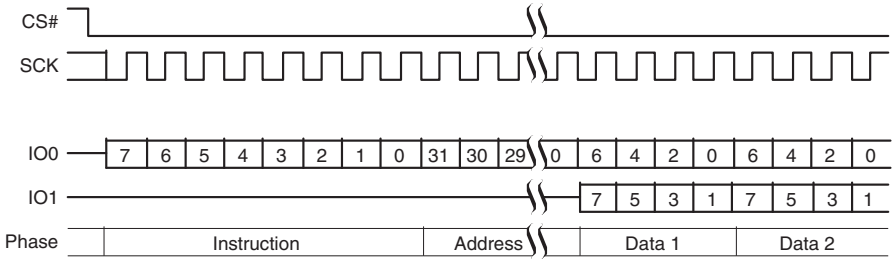


**Figure 72. Dual Output Read Command Sequence (4-Byte Address, 3Ch or 3Bh [ExtAdd=1, LC=11b])**




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

The instruction

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

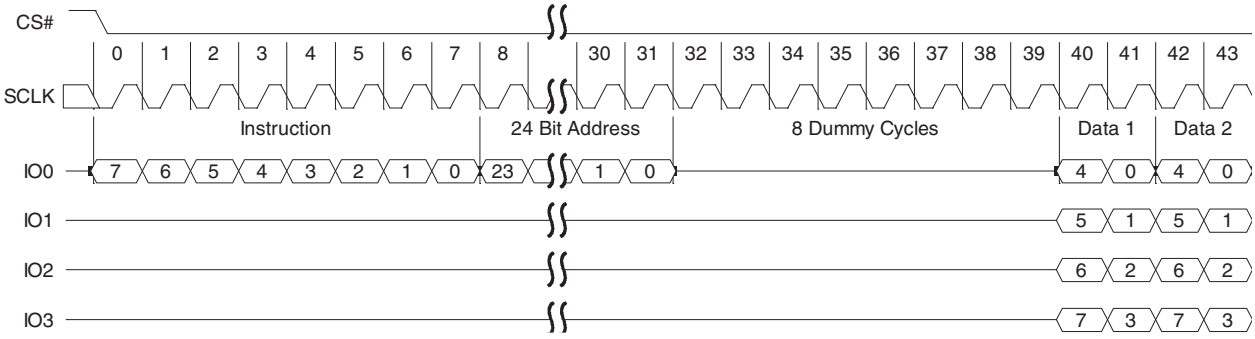
Then the memory contents, at the address given, is shifted out four bits at a time through IO0-IO3. Each nibble (4 bits) is shifted out at the SCK frequency by the falling edge of the SCK signal.

The maximum operating clock frequency for Quad Output Read command is 108 MHz. For Quad Output Read Mode, there may be dummy cycles required after the last address bit is shifted into SI before data begins shifting out of IO0-IO3. This latency period (i.e., dummy cycles) allows the device's internal circuitry enough time to set up for the initial address. During the dummy cycles, the data value on IO0-IO3 is a "don't care" and may be high impedance. The number of dummy cycles is determined by the frequency of SCK (refer to Table 22, Latency Codes on page 53).

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.

The QUAD bit of Configuration Register must be set (CR Bit1=1) to enable the Quad mode capability.

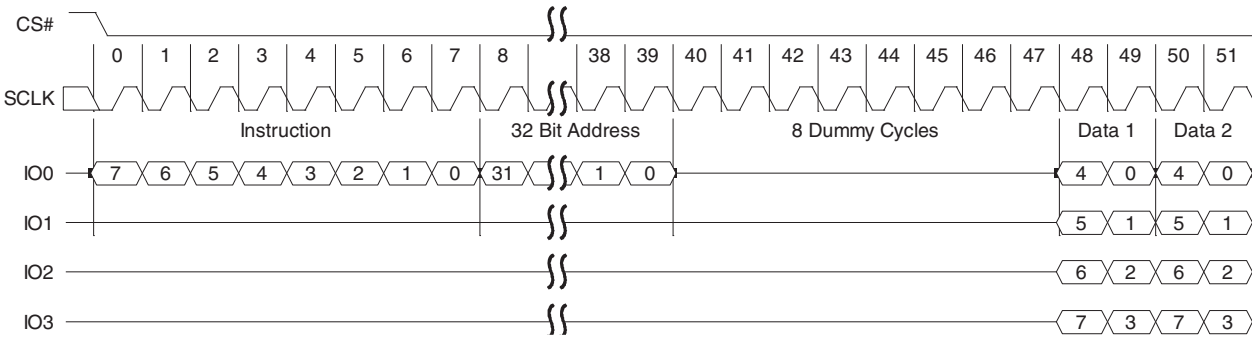
**Figure 73. Quad Output Read (QOR 6Bh or 4QOR 6Ch) Command Sequence with Read Latency**




***Notes:***

1. *A = MSB of address = A23 for ExtAdd = 0, or A31 for ExtAdd=1 or command 6Ch.*
2. *LC = 01b shown.*

**Figure 74. Quad Output Read (QOR 6Bh or 4QOR 6Ch) Command Sequence without Read Latency**




***Notes:***

1. *A = MSB of address = A23 for ExtAdd = 0, or A31 for ExtAdd = 1 or command 6Ch.*
2. *LC = 11b shown.*

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

The instruction

- BBh (ExtAdd=0) is followed by a 3-byte address (A23-A0) or
- BBh (ExtAdd=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 (SI) and IO1 (SO). It is similar to the Dual Output Read command but takes input of the address two bits per SCK rising edge. In some applications, the reduced address input 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 108 MHz.

For the Dual I/O Read command, there is a latency required after the last address bits are shifted into SI and SO 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 SI and SO are “don’t care” and may be high impedance. The number of dummy cycles is determined by the frequency of SCK (Table 22, Latency Codes on page 53). The number of dummy cycles is set by the LC bits in the Configuration Register (CR1).

The Latency Code table provides cycles for mode bits 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 a 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 latency.

The enhanced high performance 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 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 enhanced high performance 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 77; thus, eliminating eight cycles for the command sequence. The following sequences will release the device from Dual I/O enhanced high performance Read mode; after which, the device can accept standard SPI commands:

1. During the Dual I/O enhanced high performance 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 Read enhanced high performance Read mode.
2. During any operation, if CS# toggles high to low to high for eight cycles (or less) and data input (IO0 and IO1) are not set for a valid instruction sequence, then the device will be released from Dual I/O enhanced high performance Read mode. 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 (SI) and IO1 (SO).

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 (SI) and IO1 (SO). 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 75. Dual I/O Read Command Sequence (3-Byte Address, BBh [ExtAdd=0])**

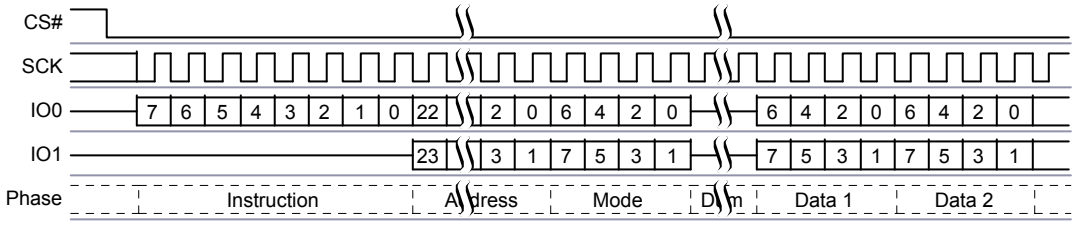
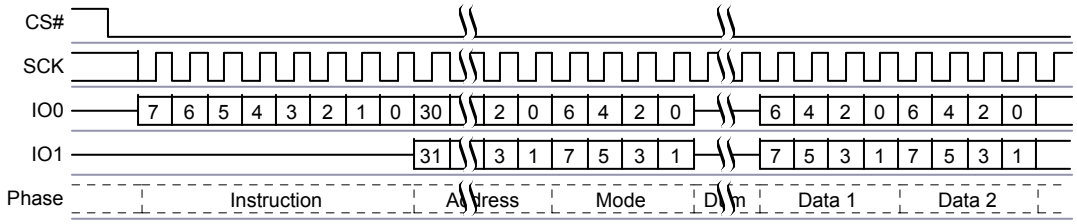


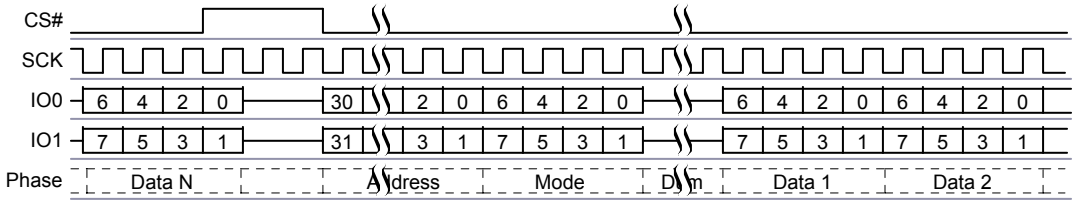

**Figure 76. Dual I/O Read Command Sequence (4-Byte Address, BCh or BBh [ExtAdd=1])**




***Note:***

1. *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.*

**Figure 77. Continuous Dual I/O Read Command Sequence (4-Byte Address, BCh or BBh [ExtAdd=1])**




***Note:***

1. *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.*

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

The instruction

- EBh (ExtAdd=0) is followed by a 3-byte address (A23-A0) or
- EBh (ExtAdd=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 is similar to the Quad Output Read command but 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 FL-S Family devices. The QUAD bit of the Configuration Register must be set (CR Bit1=1) to enable the Quad capability of FL-S Family devices.

The maximum operating clock frequency for Quad I/O Read is 108 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 and the latency code table (refer to Table 22, Latency Codes on page 53). The number of dummy cycles is set by the LC bits in the Configuration Register (CR1). However, both latency code tables use the same latency values for the Quad I/O Read command.

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 78 on page 92 or Figure 80 on page 92). 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 79 on page 92 or Figure 81 on page 92; 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. During any operation, if CS# toggles high to low to high for eight cycles (or less) and data input (IO0-IO3) are not set for a valid instruction sequence, then the device will be released from Quad I/O high performance Read mode.

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 IO0-IO3.

It is important that the IO0-IO3 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 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.

**Figure 78. Quad I/O Read Command Sequence (3-Byte Address, EBh [ExtAdd=0])**

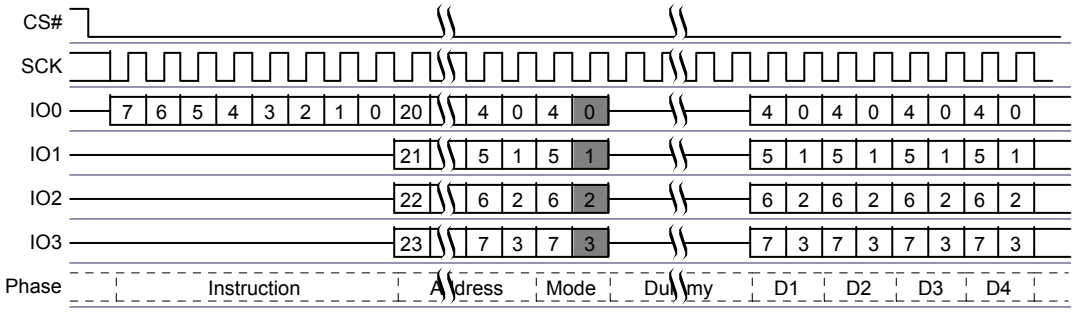



**Figure 79. Continuous Quad I/O Read Command Sequence (3-Byte Address)**

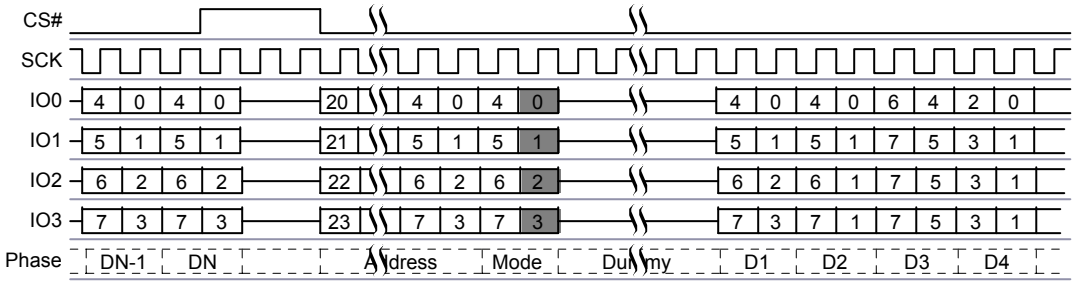



**Figure 80. Quad I/O Read Command Sequence (4-Byte Address, ECh or EBh [ExtAdd=1])**

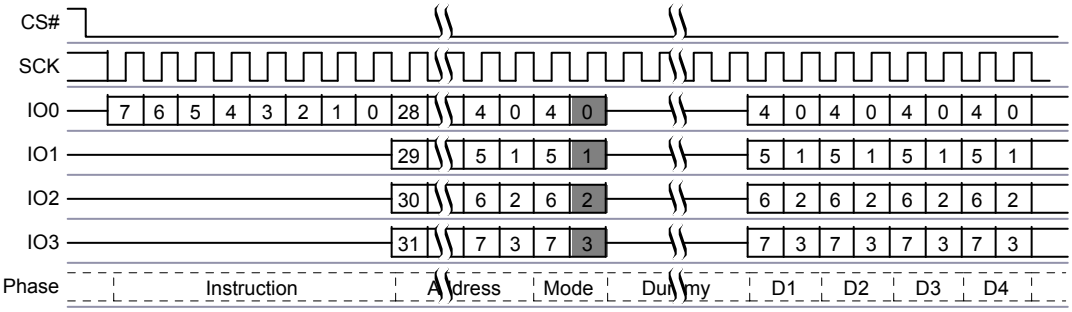
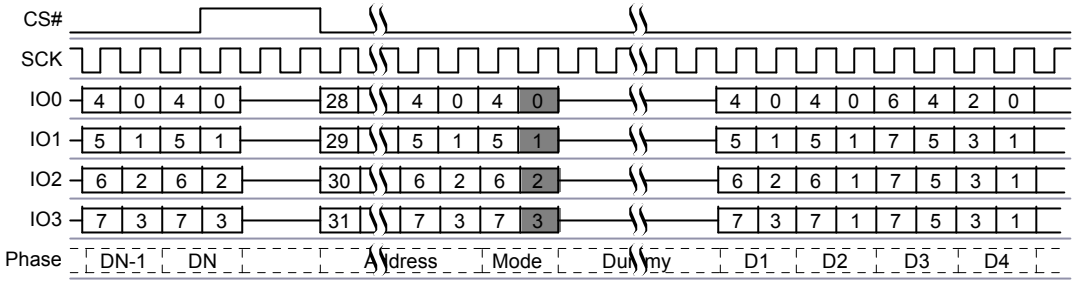



**Figure 81. Continuous Quad I/O Read Command Sequence (4-Byte Address)**



## 9.5 Program Flash Array Commands

### 9.5.1 Program Granularity

#### 9.5.1.1 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 Error Detection and Correction (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-S 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 Section 7.6.6 ECC Status Register (ECCSR) on page 56.) The ECC Status Register Read (ECCRD) command is used to read the ECC status on any ECC unit.

EDC is applied to all parts of the Flash address spaces other than registers. An 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).

#### 9.5.1.2 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 a configuration bit (SR2[6]). 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. For the very best performance, programming should be done in full, aligned, pages of 512 bytes aligned on 512-byte boundaries with each Page being programmed only once.

#### 9.5.1.3 Single Byte Programming

Single Byte Programming allows full backward compatibility to the 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 where the byte is located.

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

The Page Program (PP) commands 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 (ExtAdd=0) is followed by a 3-byte address (A23-A0) or
- 02h (ExtAdd=1) is followed by a 4-byte address (A31-A0) or
- 12h is followed by a 4-byte address (A31-A0)

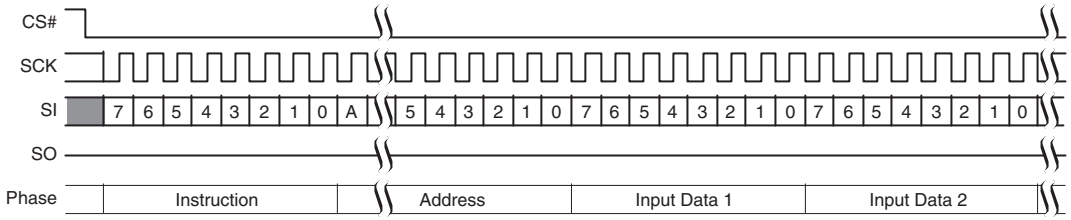
and at least one data byte on SI. Depending on the device configuration, the page size can either be 256 or 512 bytes. Up to a page can be provided on SI after the 3-byte address with instruction 02h or 4-byte address with instruction 12h has been provided. If the 9 least significant address bits (A8-A0) are not all 0, all transmitted data that goes beyond the end of the current page are programmed from the start address of the same page (from the address whose 9 least significant bits (A8-A0) are all 0) i.e. the address wraps within the page aligned address boundaries. This is a result of only requiring the user to enter one single page address to cover the entire page boundary.

If more than a page of data is sent to the device, 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 data previously loaded in the page. The last page worth of data (either 256 or 512 bytes) 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 (SR1[0]) will indicate when the programming operation is completed. The P_ERR bit (SR1[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 82. Page Program (PP 02h or 4PP 12h) Command Sequence**




***Note:***
*1. A = MSB of address = A23 for PP 02h, or A31 for 4PP 12h.*

### 9.5.3 Quad Page Program (QPP 32h or 38h, 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 size (either 256 or 512 bytes) 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 80 MHz.

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

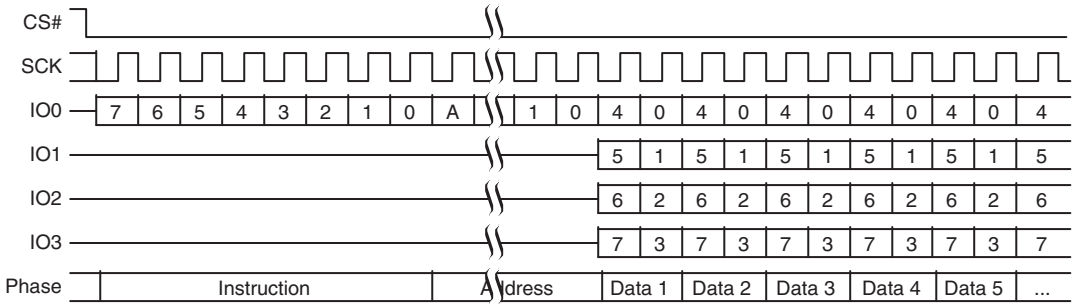
- 32h (ExtAdd=0) is followed by a 3-byte address (A23-A0) or
- 32h (ExtAdd=1) is followed by a 4-byte address (A31-A0) or
- 38h (ExtAdd=0) is followed by a 3-byte address (A23-A0) or
- 38h (ExtAdd=1) is followed by a 4-byte address (A31-A0) or
- 34h is followed by a 4-byte address (A31-A0)

and at least one data byte, into the IO signals. Data must be programmed at previously erased (FFh) memory locations.

QPP requires programming to be done one full page at a time. While less than a full page of data may be loaded for programming, the entire page is considered programmed, any locations not filled with data will be left as ones, the same page must not be programmed more than once.

All other functions of QPP are identical to Page Program. The QPP command sequence is shown in the figure below.

**Figure 83. Quad Page Program Command Sequence**




### 9.5.4 Program Suspend (PGSP 85h) and Resume (PGRS 8Ah)

The Program Suspend command allows the system to interrupt a programming operation and then read from any other non-erase-suspended sector or non-program-suspended-page. Program Suspend is valid only during a programming operation.

Commands allowed after the Program Suspend command is issued:

- Read Status Register 1 (RDSR1 05h)
- Read Status Register 2 (RDSR2 07h)

The Write in Progress (WIP) bit in Status Register 1 (SR1[0]) must be checked to know when the programming 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 time required for the suspend operation to complete is $t_{PSL}$, see Table 42, Program Suspend AC Parameters on page 111.

See Table 40, Commands Allowed During Program or Erase Suspend on page 101 for the commands allowed while programming is suspend.

The Program Resume command 8Ah must be written to resume the programming operation after a Program Suspend. If the programming operation was completed during the suspend operation, a resume command is not needed and has no effect if issued. Program Resume commands will be ignored unless a Program operation is suspended.

After a 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. Program 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 operation to progress to completion there must be some periods of time between resume and the next suspend command greater than or equal to $t_{PRS}$. See Table 42, Program Suspend AC Parameters on page 111.

**Figure 84. Program Suspend Command Sequence**

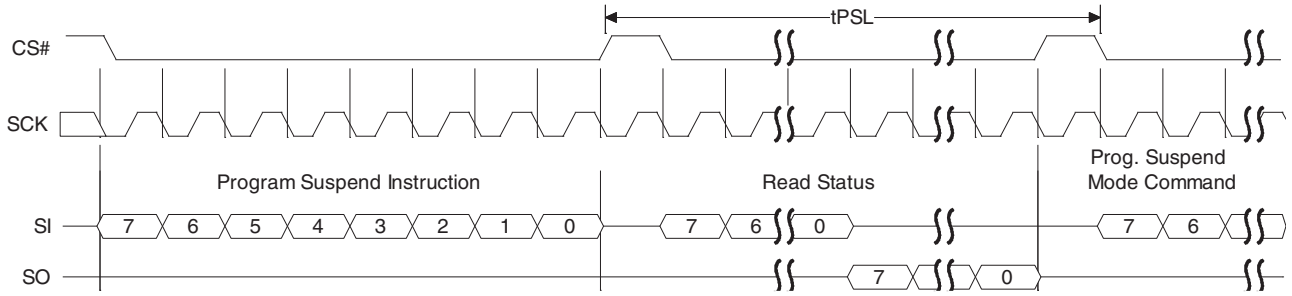
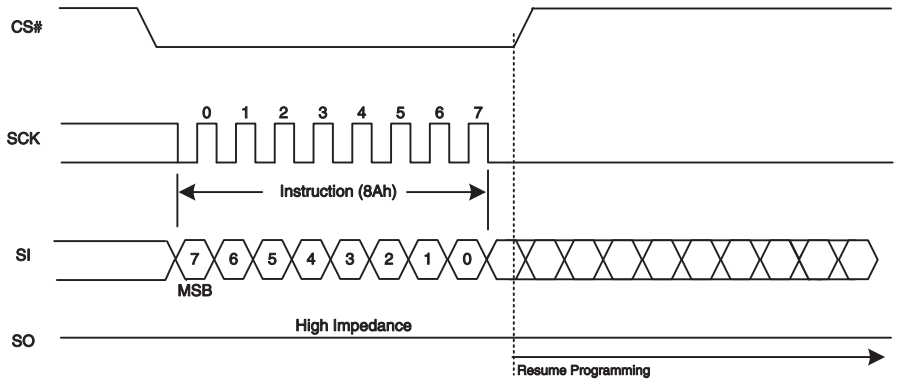



**Figure 85. Program Resume Command Sequence**



## 9.6 Erase Flash Array Commands

### 9.6.1 Parameter 4-kB Sector Erase (P4E 20h or 4P4E 21h)

The P4E command is implemented only in FL127S. The P4E command is ignored when the device is configured with the 256-kB sector option.

The Parameter 4-kB Sector Erase (P4E) command sets all the bits of a 4-kbyte parameter sector to 1 (all bytes are FFh). Before the P4E 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

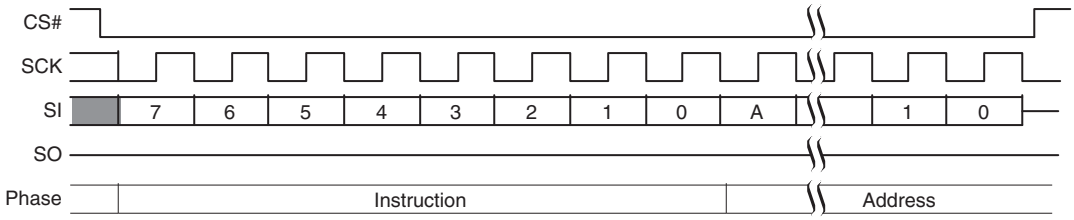
- 20h [ExtAdd=0] is followed by a 3-byte address (A23-A0), or
- 20h [ExtAdd=1] is followed by a 4-byte address (A31-A0), or
- 21h is followed by a 4-byte address (A31-A0)

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. 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 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
4 kbytes will not be executed and will not set the E_ERR status.

**Figure 86. Parameter Sector Erase (P4E 20h or 4P4E 21h) Command Sequence**




***Note:***
*1. A = MSB of address = A23 for P4E 20h with ExtAdd = 0, or A31 for P4E 20h with ExtAdd = 1 or 4P4E 21h.*

### 9.6.2 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 [ExtAdd=0] is followed by a 3-byte address (A23-A0), or
- D8h [ExtAdd=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.

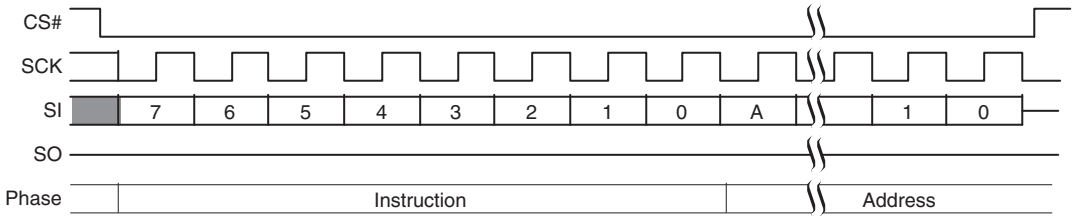
As soon as CS# is driven into the logic high state, the internal erase cycle will be initiated. With the internal erase cycle in progress, the user can read the value of the Write-In Progress (WIP) bit to check if the operation has been completed. The WIP bit will indicate a 1 when the erase cycle is 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 determines whether the SE command erases 64 kbytes or 256 kbytes. The option to use this command to always erase 256 kbytes provides for software compatibility with higher density and future S25FL family devices.

ASP has a PPB and a DYB protection bit for each sector, including any 4-kB sectors. If a sector erase command is applied to a 64-kB range that includes a protected 4-kB sector, or to a 256-kB range that includes a 64-kB protected address range, the erase will not be executed on the range and will set the E_ERR status.

**Figure 87. Sector Erase (SE D8h or 4SE DCh) Command Sequence**




***Note:***

*1. A = MSB of address = A23 for SE D8h with ExtAdd = 0, or A31 for SE D8h with ExtAdd = 1 or 4P4E DCh.*

### 9.6.3 Bulk Erase (BE 60h or C7h)

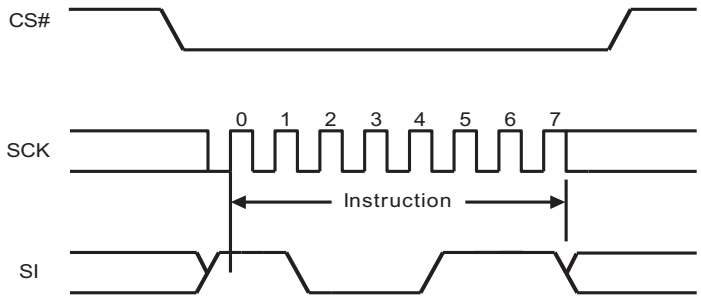
The Bulk Erase (BE) command sets all bits to 1 (all bytes are FFh) inside the entire flash memory array. Before the BE command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the Write Enable Latch (WEL) in the Status Register to enable 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. 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 0, 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 88. Bulk Erase Command Sequence**




### 9.6.4 Erase Suspend and Resume Commands (ERSP 75h or ERRS 7Ah)

The Erase Suspend command, allows the system to interrupt a sector erase operation and then read from or program data to, any other sector. Erase Suspend is valid only during a sector erase operation. The Erase Suspend command is ignored if written during the Bulk Erase operation.

When the Erase Suspend command is written during the sector erase operation, the device requires a maximum of $t_{ESL}$ (erase suspend latency) to suspend the erase operation and update the status bits. See Table 43, Erase Suspend AC Parameters on page 111.

Commands allowed after the Erase Suspend command is issued:

- Read Status Register 1 (RDSR1 05h)
- Read Status Register 2 (RDSR2 07h)

The Write in Progress (WIP) bit in Status Register 1 (SR1[0]) must be checked to know when the erase operation has stopped. The Erase Suspend bit in 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.

If the erase operation was completed during the suspend operation, a resume command is not needed and has no effect if issued. Erase Resume commands will be ignored unless an Erase operation is suspended.

See Table 40, Commands Allowed During Program or Erase Suspend on page 101 for the commands allowed while erase is suspend.

After the erase operation has been suspended, the sector enters the erase-suspend mode. The system can read data from or program data to the device. Reading at any address within an erase-suspended sector produces undetermined data.

A WREN command is required before any command that will change non-volatile data, even during erase suspend.

The WRR and PPB Erase commands are not allowed during Erase 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. However, WRR is allowed immediately following the BRAC command; in this special case the WRR is interpreted as a write to the Bank Address Register, not a write to SR1 or CR1.

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.

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.

The Erase Resume command 7Ah must be written to resume the erase operation if an Erase is suspended. Erase Resume commands will be ignored unless an Erase is suspended.

After an Erase Resume command is sent, the WIP bit in the status register will be set to a 1 and the erase operation will continue. Further Resume commands are ignored.

Erase operations may be interrupted as often as necessary e.g. an erase suspend command could immediately follow an erase resume command but, in order for an 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 $t_{ERS}$. See Table 43, Erase Suspend AC Parameters on page 111.

**Figure 89. Erase Suspend Command Sequence**

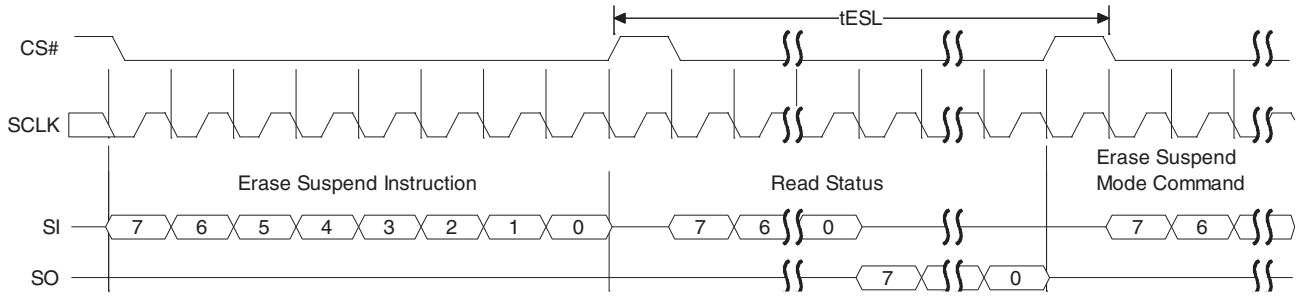



**Figure 90. Erase Resume Command Sequence**

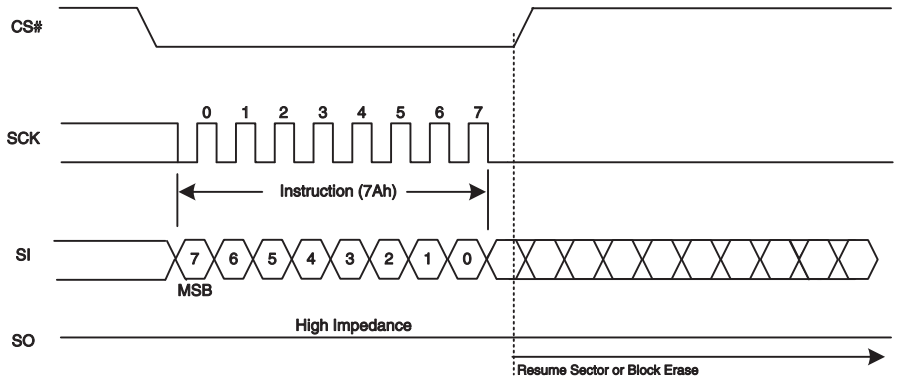


**Table 40. Commands Allowed During Program or Erase Suspend**

| Instruction Name | Instruction Code (Hex) | Allowed During Erase Suspend | Allowed During Program Suspend | Comment |
|---|---|---|---|---|
| BRAC | B9 | X | X | Bank address register may need to be changed during a suspend to reach a sector for read or program. |
| BRRD | 16 | X | X | Bank address register may need to be changed during a suspend to reach a sector for read or program. |
| BRWR | 17 | X | X | Bank address register may need to be changed during a suspend to reach a sector for read or program. |
| CLSR | 30 | X | | Clear status may be used if a program operation fails during erase suspend. |
| DYBRD | E0 | X | | It may be necessary to remove and restore dynamic protection during erase suspend to allow programming during erase suspend. |
| DYBWR | E1 | X | | It may be necessary to remove and restore dynamic protection during erase suspend to allow programming during erase suspend. |
| ERRS | 7A | X | | Required to resume from erase suspend. |
| FAST_READ | 0B | X | X | All array reads allowed in suspend. |
| 4FAST_READ | 0C | X | X | All array reads allowed in suspend. |
| MBR | FF | X | X | May need to reset a read operation during suspend. |
| PGRS | 8A | X | X | Needed to resume a program operation. A program resume may also be used during nested program suspend within an erase suspend. |
| PGSP | 85 | X | | Program suspend allowed during erase suspend. |
| PP | 02 | X | | Required for array program during erase suspend. |
| 4PP | 12 | X | | Required for array program during erase suspend. |
| PPBRD | E2 | X | | Allowed for checking persistent protection before attempting a program command during erase suspend. |
| QPP | 32, 38 | X | | Required for array program during erase suspend. |
| 4QPP | 34 | X | | Required for array program during erase suspend. |
| 4READ | 13 | X | X | All array reads allowed in suspend. |
| RDCR | 35 | X | X | |
| DIOR | BB | X | X | All array reads allowed in suspend. |
| 4DIOR | BC | 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. |
| QIOR | EB | X | X | All array reads allowed in suspend. |
| 4QIOR | EC | X | X | All array reads allowed in suspend. |
| QOR | 6B | X | X | All array reads allowed in suspend. |
| 4QOR | 6C | X | X | All array reads allowed in suspend. |
| RDSR1 | 05 | X | X | Needed to read WIP to determine end of suspend process. |
| RDSR2 | 07 | X | X | Needed to read suspend status to determine whether the operation is suspended or complete. |
| READ | 03 | X | X | All array reads allowed in suspend. |
| RESET | F0 | X | X | Reset allowed anytime. |
| WREN | 06 | X | | Required for program command within erase suspend. |
| WRR | 01 | X | X | Bank register may need to be changed during a suspend to reach a sector needed for read or program. WRR is allowed when following BRAC. |

## 9.7 One Time Program Array Commands

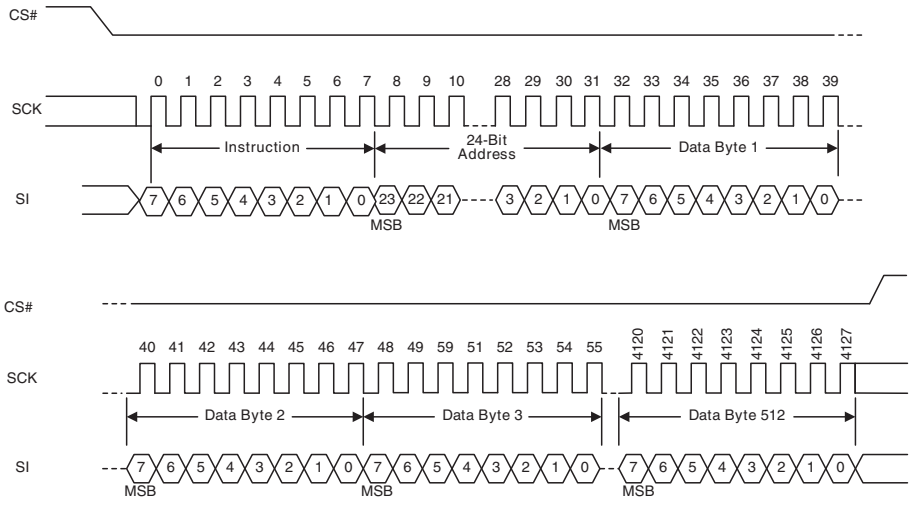
### 9.7.1 OTP Program (OTPP 42h)

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 A23 to A10 must be 0 for this command. Refer to Section 7.5 OTP Address Space on page 49 for details on the OTP region. The protocol of the OTP Program command is the same as the Page Program command. 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 SR1 may be checked to determine when the operation is completed. The P_ERR bit in SR1 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 0s in a region that is locked will fail with the P_ERR bit in SR1 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).

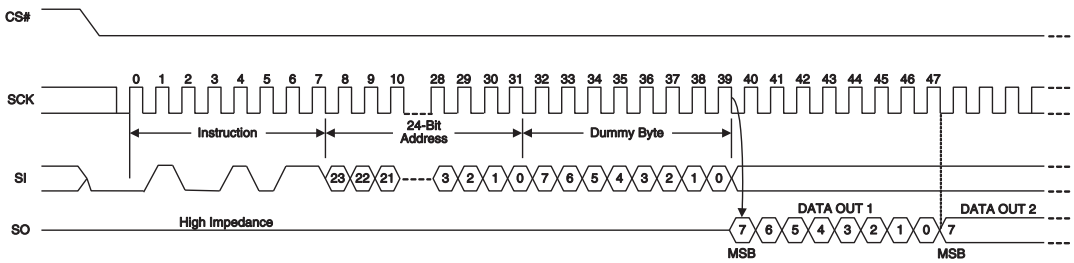
**Figure 91. OTP Program Command Sequence**




### 9.7.2 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 A23 to A10 must be 0 for this command. Refer to OTP Address Space on page 49 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. Also, the OTP Read command is not affected by the latency code. The OTP read command always has one dummy byte of latency as shown below.

**Figure 92. OTP Read Command Sequence**

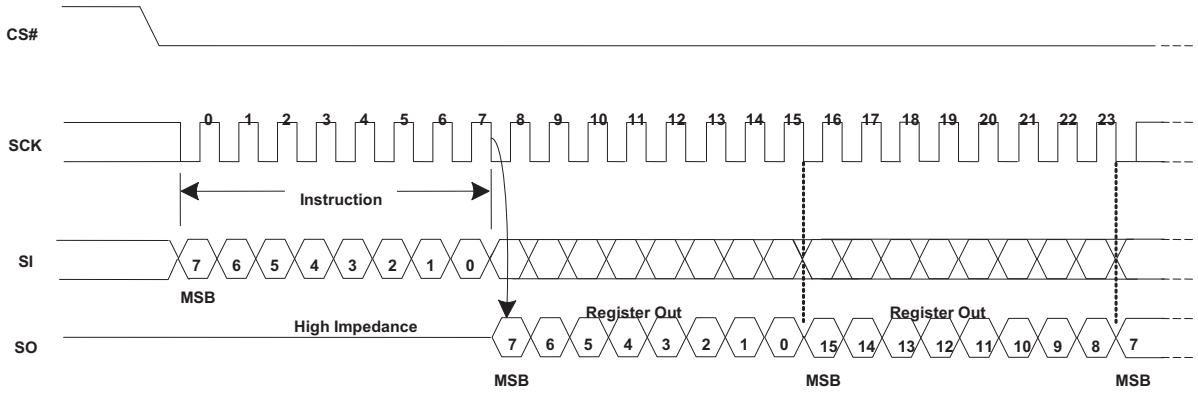


## 9.8 Advanced Sector Protection Commands

### 9.8.1 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 108 MHz.

**Figure 93. ASPRD Command**




### 9.8.2 ASP Program (ASPP 2Fh)

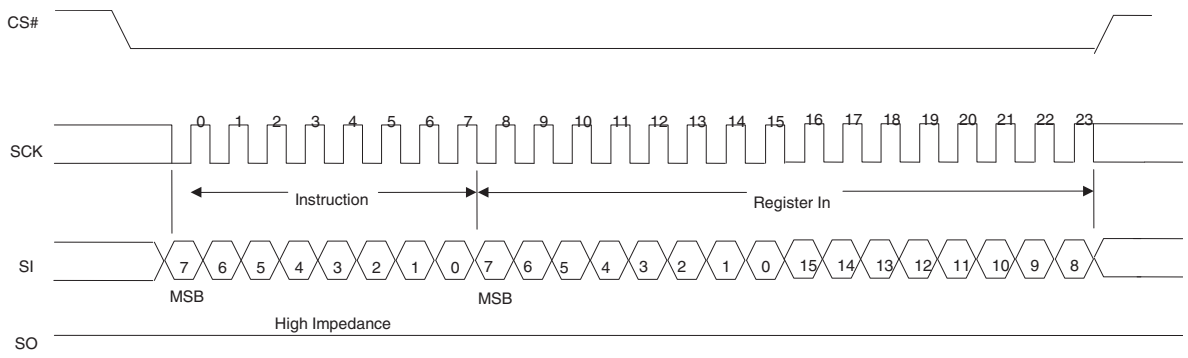
Before the ASP Program (ASPP) command can be accepted by the device, a Write Enable (WREN) command must be issued. After the Write Enable (WREN) command has been decoded, the device will set the Write Enable Latch (WEL) in the Status Register to enable any write operations.

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.

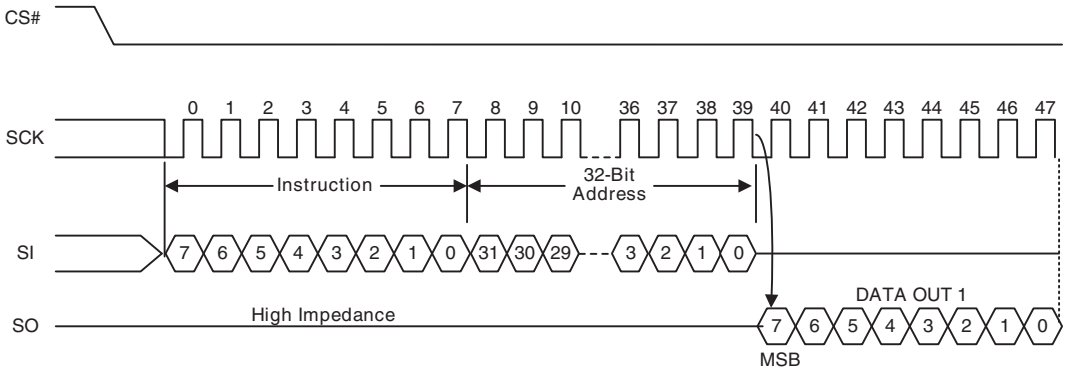
**Figure 94. ASPP Command**



### 9.8.3 DYB Read (DYBRD E0h)

The instruction E0h is latched into SI by the rising edge of the SCK signal. The instruction is followed by the 32-bit address selecting location zero within the desired sector Note: the high order address bits not used by a particular density device must be 0. Then the 8-bit DYB access register contents are shifted out on the serial output SO. 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 108 MHz.

**Figure 95. DYBRD Command Sequence**



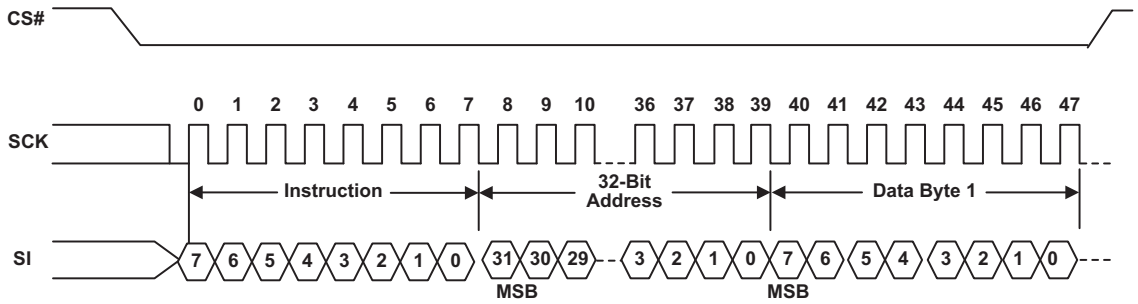

### 9.8.4 DYB Write (DYBWR E1h)

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, the 32-bit address selecting location zero within the desired sector (note, the high order address bits not used by a particular density device must be 0), then the data byte on SI. The DYB Access Register is one data byte in length.

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. If not, the DYBWR command is not executed. 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 96. DYBWR Command Sequence**

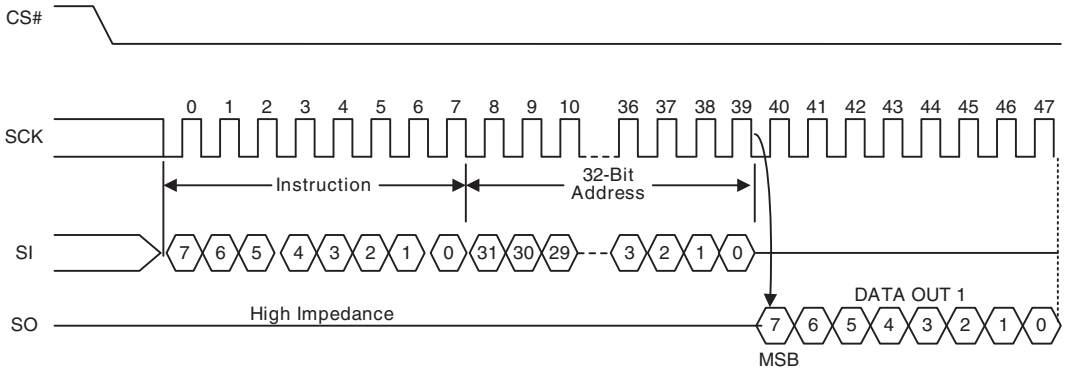


### 9.8.5 PPB Read (PPBRD E2h)

The instruction E2h is shifted into SI by the rising edges of the SCK signal, followed by the 32-bit address selecting location zero within the desired sector. Note: the high order address bits not used by a particular density device must be 0. Then the 8-bit PPB access register contents are shifted out on SO.

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 108 MHz.

**Figure 97. PPBRD Command Sequence**




### 9.8.6 PPB Program (PPBP E3h)

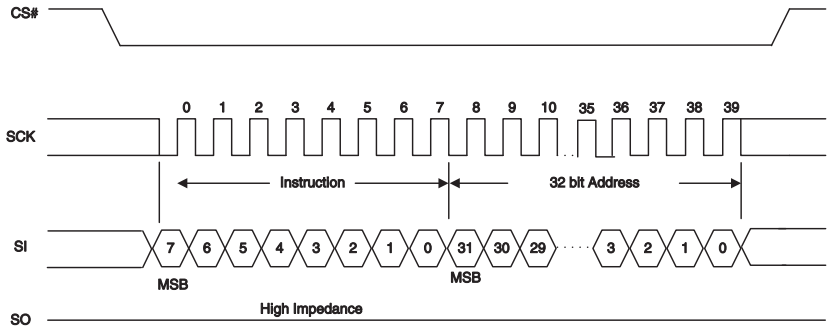
Before the PPB Program (PPBP) command can be accepted by the device, a Write Enable (WREN) command must be issued. After the Write Enable (WREN) command has been decoded, the device will set the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The PPBP command is entered by driving CS# to the logic low state, followed by the instruction, followed by the 32-bit address selecting location zero within the desired sector (note, the high order address bits not used by a particular density device must be 0).

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 98. PPBP Command Sequence**



### 9.8.7 PPB Erase (PPBE E4h)

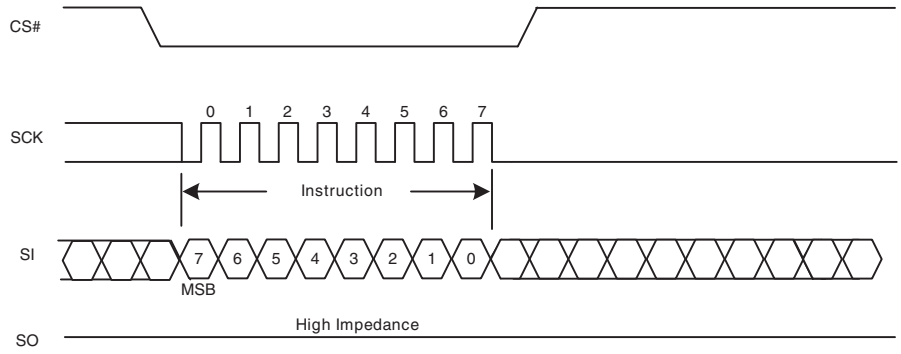
The PPB Erase (PPBE) command sets all PPB bits to 1. Before the PPB Erase command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The instruction E4h is shifted into SI 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. 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.

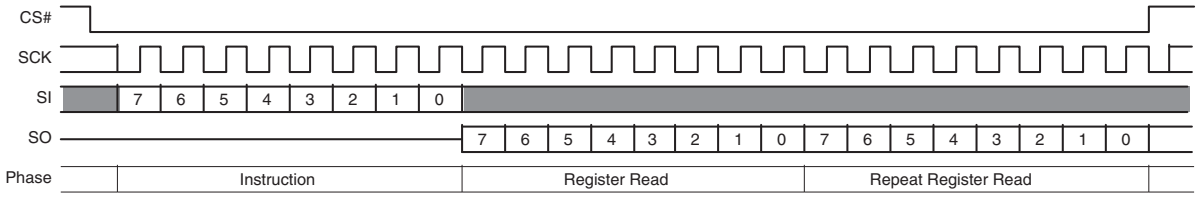
**Figure 99. PPB Erase Command Sequence**




### 9.8.8 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. 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 100. PPB Lock Register Read Command Sequence**



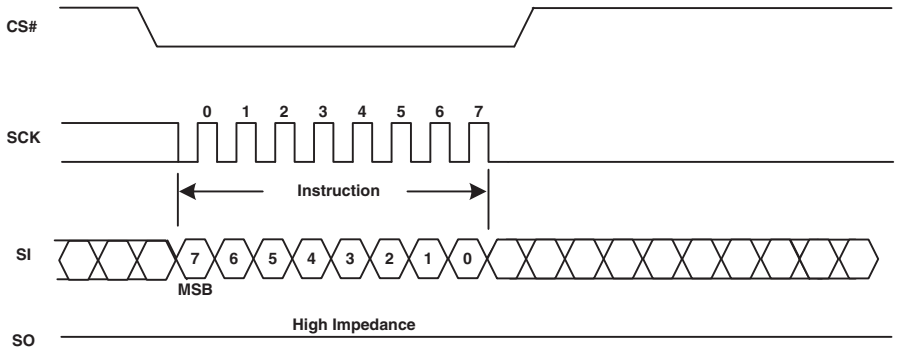
### 9.8.9 PPB Lock Bit Write (PLBWR A6h)

The PPB Lock Bit Write (PLBWR) command clears the PPB Lock Register to 0. 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 108 MHz.

**Figure 101. PPB Lock Bit Write Command Sequence**

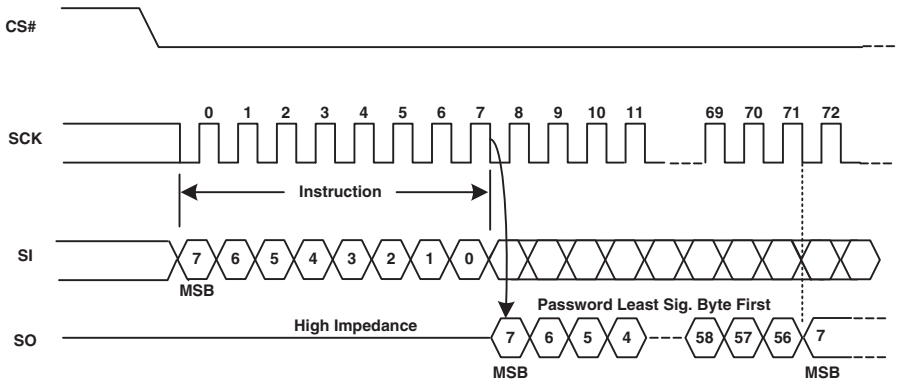



### 9.8.10 Password Read (PASSRD E7h)

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 PASSRD command is ignored.

The PASSRD command is shifted into SI. Then the 64-bit Password is shifted out on the serial output SO, 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 108 MHz.

**Figure 102. Password Read Command Sequence**



### 9.8.11 Password Program (PASSP E8h)

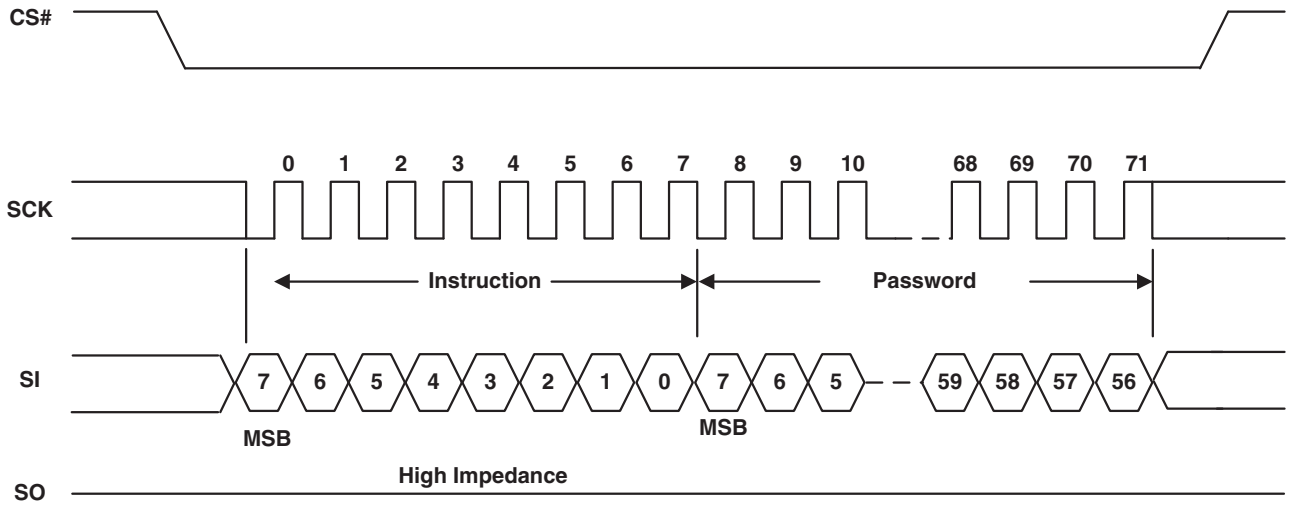
Before the Password Program (PASSP) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device. After the Write Enable (WREN) command has been decoded, the device sets the Write Enable Latch (WEL) to enable the PASSP operation.

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, 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 108 MHz.

**Figure 103. Password Program Command Sequence**




### 9.8.12 Password Unlock (PASSU E9h)

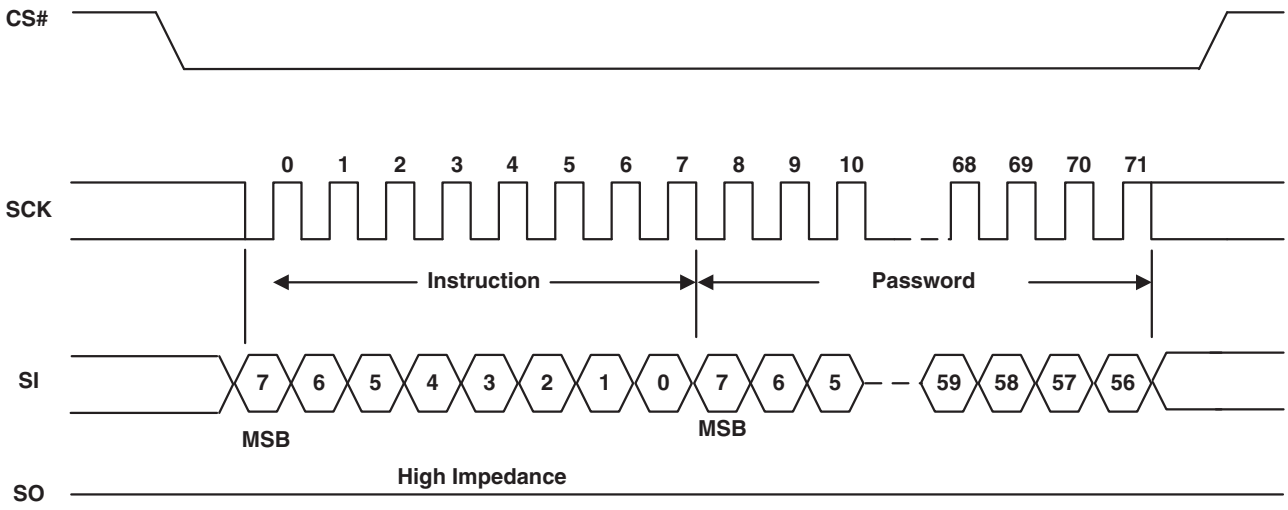
The PASSU command is entered by driving CS# to the logic low state, followed by the instruction and the password data bytes on SI, least significant byte first, most significant bit of each byte first. The password is sixty-four (64) bits in length.

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 108 MHz.

Figure 104. Password Unlock Command Sequence

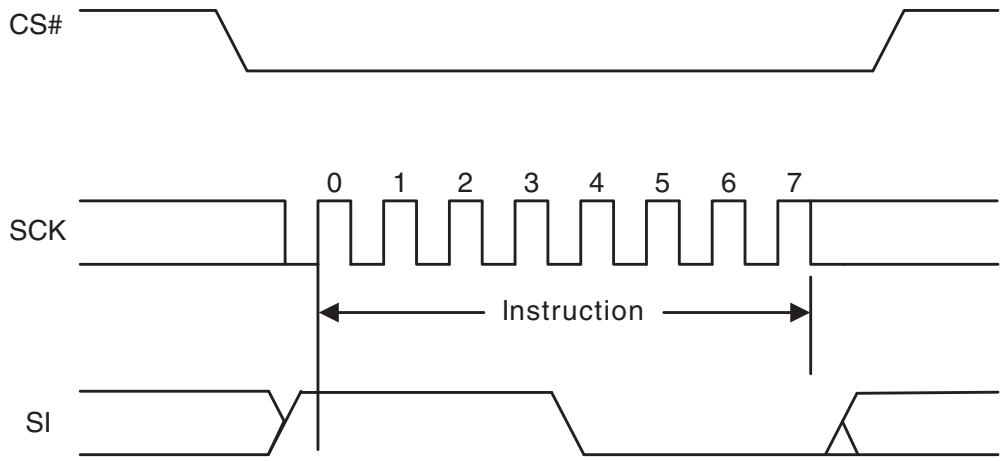



## 9.9 Reset Commands

### 9.9.1 Software Reset Command (RESET F0h)

The Software Reset command (RESET) restores the device to its initial power up state, except for the volatile FREEZE bit in the Configuration register CR1[1] and the volatile PPB Lock bit in the PPB Lock Register. 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. Note that the non-volatile bits in the configuration register, TBPROT, TBPARM, and BPNV, retain their previous state after a Software Reset. The Block Protection bits BP2, BP1, and BP0, in the status register will only be reset if they are configured as volatile via the BPNV bit in the Configuration Register (CR1[3]) and FREEZE is cleared to 0. The software reset cannot be used to circumvent the FREEZE or PPB Lock bit protection mechanisms for the other security configuration bits. The reset command is executed when CS# is brought to high state and requires $t_{RPH}$ time to execute.

Figure 105. Software Reset Command Sequence

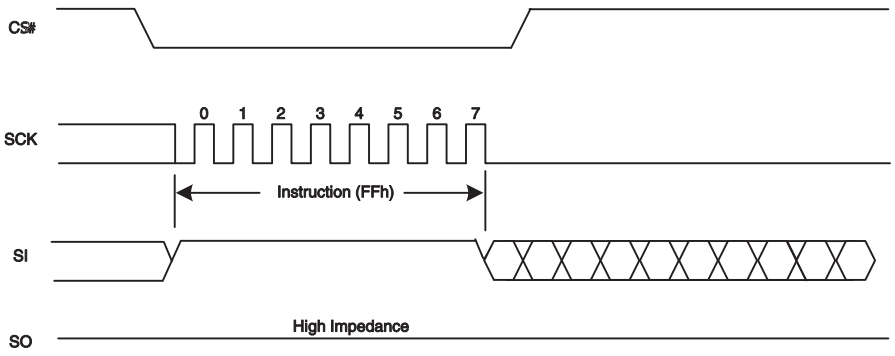


### 9.9.2 Mode Bit Reset (MBR FFh)

The Mode Bit Reset (MBR) command can be 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 or IO0 for 8 SCK cycles. IO1 to IO3 are “don’t care” during these cycles.

**Figure 106. Mode Bit Reset Command Sequence**




## 9.10 Embedded Algorithm Performance Tables

The Joint Electron Device Engineering Council (JEDEC) standard JESD22-A117 defines the procedural requirements for performing valid endurance and retention tests based on a qualification specification. This methodology is intended to determine the ability of a flash device to sustain repeated data changes without failure (program/erase endurance) and to retain data for the expected life (data retention). Endurance and retention qualification specifications are specified in JESD47 or may be developed using knowledge-based methods as in JESD94.

**Table 41. Program and Erase Performance**

| Symbol | Parameter | Min | Typ (1) | Max (2) | Unit |
|---|---|---|---|---|---|
| $t_W$ | WRR Write Time | | 130 | 780 | ms |
| $t_{PP}$ | Page Programming (512 bytes)<br>Page Programming (256 bytes) | | 640<br>395 | 1480<br>1185 (3) | µs |
| $t_{SE}$ | Sector Erase Time (64-kB / 4-kB physical sectors) | | 130 | 780 (4) | ms |
| | Sector Erase Time<br>(64 kB Top/Bottom: logical sector = 16 x 4-kB physical sectors) | | 2,100 | 12,600 | ms |
| | Sector Erase Time<br>(256-kB logical sectors = 4 x 64-kB physical sectors) | | 520 | 3120 | ms |
| $t_{BE}$ | Bulk Erase Time (Hybrid 4 kB Top/Bottom with 64-kB uniform) | | 35 | 210 | sec |
| | Bulk Erase Time (256-kB uniform) | | 33 | 200 | sec |
| | Erase per Sector | 100,000 | | | cycles |

***Notes:***

1. *Typical program and erase times assume the following conditions: 25°C, $V_{CC}$ = 3.0V; random data pattern.*
2. *Under worst case conditions of 90°C; 100,000 cycles max.*
3. *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. The programming time for the PPBP E3h command is the same as $t_{PP}$.*
4. *The erase time for PPBE E4h command is the same as $t_{SE}$.*
5. *Data retention of 20 years is based on 1k erase cycles or less.*

**Table 42. Program Suspend AC Parameters**

| Parameter | Min | Typical | Max | Unit | Comments |
|---|---|---|---|---|---|
| Program Suspend Latency ($t_{PSL}$) | | | 45 | µs | The time from Program Suspend command until the WIP bit is 0 |
| Program Resume to next Program Suspend ($t_{PRS}$) | 0.06 | 100 | | µs | Minimum is the time needed to issue the next Program Suspend command but ≥ typical periods are needed for Program to progress to completion |

**Table 43. Erase Suspend AC Parameters**

| Parameter | Min | Typical | Max | Unit | Comments |
|---|---|---|---|---|---|
| Erase Suspend Latency ($t_{ESL}$) | | | 45 | µs | The time from Erase Suspend command until the WIP bit is 0 |
| Erase Resume to next Erase Suspend ($t_{ERS}$) | 0.06 | 100 | | µs | Minimum is the time needed to issue the next Erase Suspend command but ≥ typical periods are needed for the Erase to progress to completion |

# 10. Data Integrity

## 10.1 Erase Endurance

**Table 10.1 Erase Endurance**

| Parameter | Minimum | Unit |
|---|---|---|
| Program/Erase cycles per main Flash array sectors | 100K | PE cycle |
| Program/Erase cycles per PPB array or non-volatile register array (1) | 100K | PE cycle |

***Note:***

1. *Each write command to a non-volatile register causes a PE cycle on the entire non-volatile register array. OTP bits and registers internally reside in a separate array that is not PE cycled.*

## 10.2 Data Retention

**Table 10.2 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. An application note is available at: www.cypress.com/appnotes.

# 11. Software Interface Reference

## 11.1 Command Summary

**Table 44. FL127S Command Set (sorted by instruction)**

| Instruction (Hex) | Command Name | Command Description | Maximum Frequency (MHz) |
|---|---|---|---|
| 01 | WRR | Write Register (Status-1, Configuration-1) | 108 |
| 02 | PP | Page Program (3- or 4-byte address) | 108 |
| 03 | READ | Read (3- or 4-byte address) | 50 |
| 04 | WRDI | Write Disable | 108 |
| 05 | RDSR1 | Read Status Register 1 | 108 |
| 06 | WREN | Write Enable | 108 |
| 07 | RDSR2 | Read Status Register 2 | 108 |
| 0B | FAST_READ | Fast Read (3- or 4-byte address) | 108 |
| 0C | 4FAST_READ | Fast Read (4-byte address) | 108 |
| 12 | 4PP | Page Program (4-byte address) | 108 |
| 13 | 4READ | Read (4-byte address) | 50 |
| 14 | ABRD | AutoBoot Register Read | 108 |
| 15 | ABWR | AutoBoot Register Write | 108 |
| 16 | BRRD | Bank Register Read | 108 |
| 17 | BRWR | Bank Register Write | 108 |
| 18 | ECCRD | ECC Read | 108 |
| 20 | P4E | Parameter 4 kB-sector Erase (3- or 4-byte address) | 108 |
| 21 | 4P4E | Parameter 4 kB-sector Erase (4-byte address) | 108 |
| 2B | ASPRD | ASP Read | 108 |
| 2F | ASPP | ASP Program | 108 |
| 30 | CLSR | Clear Status Register - Erase/Program Fail Reset | 108 |
| 32 | QPP | Quad Page Program (3- or 4-byte address) | 80 |
| 34 | 4QPP | Quad Page Program (4-byte address) | 80 |
| 35 | RDCR | Read Configuration Register 1 | 108 |
| 38 | QPP | Quad Page Program (3- or 4-byte address) | 80 |
| 3B | DOR | Read Dual Out (3- or 4-byte address) | 108 |
| 3C | 4DOR | Read Dual Out (4-byte address) | 108 |
| 42 | OTPP | OTP Program | 108 |
| 4B | OTPR | OTP Read | 108 |
| 5A | RSFDP | Read JEDEC Serial Flash Discoverable Parameters | 108 |
| 60 | BE | Bulk Erase | 108 |
| 6B | QOR | Read Quad Out (3- or 4-byte address) | 108 |
| 6C | 4QOR | Read Quad Out (4-byte address) | 108 |
| 75 | ERSP | Erase Suspend | 108 |
| 7A | ERRS | Erase Resume | 108 |
| 85 | PGSP | Program Suspend | 108 |
| 8A | PGRS | Program Resume | 108 |
| 90 | READ_ID (REMS) | Read Electronic Manufacturer Signature | 108 |
| 9F | RDID | Read ID (JEDEC Manufacturer ID and JEDEC CFI) | 108 |
| A3 | Reserved-A3 | Reserved | 108 |
| A6 | PLBWR | PPB Lock Bit Write | 108 |

**Table 44. FL127S Command Set (sorted by instruction) (Continued)**

| Instruction (Hex) | Command Name | Command Description | Maximum Frequency (MHz) |
|---|---|---|---|
| A7 | PLBRD | PPB Lock Bit Read | 108 |
| AB | RES | Read Electronic Signature | 50 |
| B9 | BRAC | Bank Register Access<br>(Legacy Command formerly used for Deep Power Down) | 108 |
| BB | DIOR | Dual I/O Read (3- or 4-byte address) | 108 |
| BC | 4DIOR | Dual I/O Read (4-byte address) | 108 |
| C7 | BE | Bulk Erase (alternate command) | 108 |
| D8 | SE | Erase 64 kB or 256 kB (3- or 4-byte address) | 108 |
| DC | 4SE | Erase 64 kB or 256 kB (4-byte address) | 108 |
| E0 | DYBRD | DYB Read | 108 |
| E1 | DYBWR | DYB Write | 108 |
| E2 | PPBRD | PPB Read | 108 |
| E3 | PPBP | PPB Program | 108 |
| E4 | PPBE | PPB Erase | 108 |
| E5 | Reserved-E5 | Reserved | |
| E6 | Reserved-E6 | Reserved | |
| E7 | PASSRD | Password Read | 108 |
| E8 | PASSP | Password Program | 108 |
| E9 | PASSU | Password Unlock | 108 |
| EB | QIOR | Quad I/O Read (3- or 4-byte address) | 108 |
| EC | 4QIOR | Quad I/O Read (4-byte address) | 108 |
| F0 | RESET | Software Reset | 108 |
| FF | MBR | Mode Bit Reset | 108 |

# 12. Serial Flash Discoverable Parameters (SFDP) Address Map

The SFDP address space has a header starting at address 0 that identifies the SFDP data structure and provides a pointer to each parameter. One Basic Flash parameter is mandated by the JEDEC JESD216B standard. Two optional parameter tables for Sector Map and 4-byte Address Instructions follow the Basic Flash table. Cypress provides an additional parameter by pointing to the ID-CFI address space i.e. the IDCFI address space is a sub-set of the SFDP address space. The parameter tables portion of the SFDP data structure are 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 45. 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 |
| 1120h | Start of SFDP parameter which is one of the CFI parameter tables |
| ... | Remainder of SFDP parameter tables followed by either more CFI parameters or undefined space |

## 12.1 SFDP Header Field Definitions

**Table 46. 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) 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. 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 | 20h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned) JEDEC Basic SPI Flash parameter byte offset = 1120h |
| 0Dh | | 11h | 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) - 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 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 |
| 14h | Parameter Header 1 2nd DWORD | 20h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned) JEDEC Basic SPI Flash parameter byte offset = 1120h address |
| 15h | | 11h | Parameter Table Pointer Byte 1 |
| 16h | | 00h | Parameter Table Pointer Byte 2 |
| 17h | | FFh | Parameter ID MSB (FFh = JEDEC defined Parameter) |

**Table 46. SFDP Header (Continued)**

| SFDP Byte Address | SFDP Dword Name | Data | Description |
|---|---|---|---|
| 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 | 20h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned) JEDEC Basic SPI Flash parameter byte offset = 1120h address |
| 1Dh | | 11h | 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 | | 0Eh | Parameter Table Length (in double words = Dwords = 4 byte units) 0Eh = 14 Dwords |
| 24h | Parameter Header 3 2nd DWORD | 60h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned) JEDEC parameter byte offset = 1160h |
| 25h | | 11h | 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 | 98h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned) JEDEC parameter byte offset = 1198h |
| 2Dh | | 11h | 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 / Cypress |
| 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 | | 68h | Parameter Table Length (in double words = Dwords = 4 byte units) CFI starts at 1000h, the final SFDP parameter (CFI ID = A5) starts at 111Eh (SFDP starting point of 1120h -2hB of CFI parameter header), for a length of 11EhB excluding the CFI A5 parameter. The final CFI A5 parameter adds an additional 82hB for a total of 11Eh + 82h = 1A0hB. 1A0hB/4 = 68h Dwords |
| 34h | Parameter Header 5 2nd DWORD | 00h | Parameter Table Pointer Byte 0 (Dword = 4 byte aligned) 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) |

## 12.2 Device ID and Common Flash Interface (ID-CFI) Address Map

### 12.2.1 Field Definitions

**Table 47. Manufacturer and Device ID**

| Byte Address | Data | Description |
|---|---|---|
| 00h | 01h | Manufacturer ID for Cypress |
| 01h | 20h (128 Mb) | Device ID Most Significant Byte - Memory Interface Type |
| 02h | 18h (128 Mb) | 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 legacy ID-CFI address map. This only includes up to the end of the Primary Vendor Specific table. The Alternate Vendor Specific table contains additional information. |
| 04h | 00h (Uniform 256-kB sectors)<br>01h (4-kB parameter sectors with uniform 64-kB sectors) | Sector Architecture |
| 05h | 80h (FL-S Family) | Family ID |
| 06h | xxh | ASCII characters for Model<br>Refer to Ordering Information on page 138 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 48. 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 49. CFI System Interface String**

| Byte Address | Data | Description |
|---|---|---|
| 1Bh | 27h | $V_{CC}$ Min. (erase/program): 100 millivolts |
| 1Ch | 36h | $V_{CC}$ Max. (erase/program): 100 millivolts |
| 1Dh | 00h | $V_{PP}$ Min. voltage (00h = no $V_{PP}$ present) |
| 1Eh | 00h | $V_{PP}$ Max. voltage (00h = no $V_{PP}$ present) |
| 1Fh | 06h | Typical timeout per single byte program $2^N$ µs |
| 20h | 0Ah (256B page)<br>0Ah (512B page) | Typical timeout for Min. size Page program $2^N$ µs<br>(00h = not supported) |
| 21h | 08h (4 kB or 64 kB)<br>0Ah (256 kB) | Typical timeout per individual sector erase $2^N$ ms |
| 22h | 0Fh (128 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 | 03h | Max. timeout for full chip erase $2^N$ times typical<br>(00h = not supported) |

**Table 50. Device Geometry Definition for Bottom Boot Initial Delivery State**

| Byte Address | Data | Description |
|---|---|---|
| 27h | 18h (128 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>(0000 = not supported<br>0008h = 256B page<br>0009h = 512B page) |
| 2Bh | 00h | |
| 2Ch | 02h | Number of Erase Block Regions within device<br>1 = Uniform Device, 2 = Boot Device |
| 2Dh | 0Fh | Erase Block Region 1 Information (refer to JEDEC JEP137)<br>16 sectors = 16-1 = 000Fh<br>4-kB sectors = 256 bytes x 0010h |
| 2Eh | 00h | |
| 2Fh | 10h | |
| 30h | 00h | |
| 31h | FEh | Erase Block Region 2 Information<br>255 sectors = 255-1 = 00FEh (128 Mb)<br>64-kB sectors = 0100h x 256 bytes |
| 32h | 00h (128 Mb) | |
| 33h | 00h | |
| 34h | 01h | |
| 35h thru 3Fh | FFh | RFU |

***Note:***

1. *FL127S 128 Mbit devices have either a hybrid sector architecture with sixteen 4 kB sectors and all remaining sectors of 64 kB or with uniform 256 kB sectors. Devices with the hybrid sector architecture are initially shipped from Cypress with the 4 kB sectors located at the bottom of the array address map. However, the device configuration TBPARM bit CR1[2] may be programed to invert the sector map to place the 4 kB sectors at the top of the array address map. The CFI geometry information of the above table is relevant only to the initial delivery state of a hybrid sector device. The Flash device driver software must examine the TBPARM bit to determine if the sector map was inverted at a later time.*

**Table 51. Device Geometry Definition for Uniform Sector Devices**

| Byte Address | Data | Description |
|---|---|---|
| 27h | 18h (128 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 | 09h | Max. number of bytes in multi-byte write = $2^N$<br>(0000 = not supported<br>0008h = 256B page<br>0009h = 512B page) |
| 2Bh | 00h | |
| 2Ch | 01h | Number of Erase Block Regions within device<br>1 = Uniform Device, 2 = Boot Device |
| 2Dh | 3Fh (128 Mb) | Erase Block Region 1 Information (refer to JEDEC JEP137)<br>64 sectors = 64-1 = 003Fh (128 Mb)<br>256-kB sectors = 256 bytes x 0400h |
| 2Eh | 00h | |
| 2Fh | 00h | |
| 30h | 04h | |
| 31h thru 3Fh | FFh | RFU |

**Table 52. 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<br>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<br>1 = Read Only<br>2 = Read and Program |
| 47h | 01h | Sector Protect<br>00 = Not Supported<br>X = Number of sectors in group |

**Table 52. CFI Primary Vendor-Specific Extended Query (Continued)**

| Byte Address | Data | Description |
|---|---|---|
| 48h | 00h | Temporary Sector Unprotect<br>00 = Not Supported<br>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<br>09 = Secure |
| 4Ah | 00h | Simultaneous Operation<br>00 = Not Supported<br>X = Number of Sectors |
| 4Bh | 01h | Burst Mode (Synchronous sequential read) support<br>00 = Not Supported<br>01 = Supported |
| 4Ch | Model Dependent<br>03h (Models x0)<br>04h (Models x1) | Page Mode Type, model dependent<br>00 = Not Supported<br>01 = 4 Word Read Page<br>02 = 8 Read Word Page<br>03 = 256-Byte Program Page<br>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 select) |
| 50h | 01h | Program Suspend<br>00 = Not Supported<br>01 = Supported |

The Alternate Vendor-Specific Extended Query provides information related to the expanded command set provided by the FL-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 53. 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 54. CFI Alternate Vendor-Specific Extended Query Parameter 0**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 00h | 00h | Parameter ID (Ordering Part Number) |
| 01h | 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) |
| 02h | 53h | ASCII “S” for manufacturer (Cypress) |
| 03h | 32h | ASCII “25” for Product Characters (Single Die SPI) |
| 04h | 35h | |
| 05h | 46h | ASCII “FL” for Interface Characters (SPI 3 Volt) |
| 06h | 4Ch | |
| 07h | 31h (128 Mb) | ASCII characters for density |
| 08h | 32h (128 Mb) | |
| 09h | 38h (128 Mb) | |
| 0Ah | 53h | ASCII “S” for Technology (65 nm MirrorBit) |
| 0Bh | 41h | ASCII characters for speed grade<br>Refer to Ordering Information on page 141 for the speed grade definitions. |
| 0Ch | 42h | |
| 0Dh | 3Fh | ASCII “??” for Package (Generally the package is not specified for an individual memory device because the choice of package is generally made after the device is tested and this parameter is programmed. However, space is provided in this parameter for special cases where devices are tested and programmed for use only in a specific package) |
| 0Eh | 3Fh | |
| 0Fh | 49h | ASCII character for temperature range |
| 10h | xxh | ASCII characters for Model<br>Refer to Ordering Information on page 141 for the model number definitions. |
| 11h | xxh | |

**Table 55. CFI Alternate Vendor-Specific Extended Query Parameter 80h Address Options**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 00h | 80h | Parameter ID (address options) |
| 01h | 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) |
| 02h | F0h | Bits 7:4 - Reserved = 1111b<br>Bit 3 - AutoBoot support - Ye s= 0b, No = 1b<br>Bit 2 - 4-byte address instructions supported - Yes = 0b, No = 1b<br>Bit 1 - Bank address + 3-byte address instructions supported - Yes = 0b, No = 1b<br>Bit 0 - 3-byte address instructions supported - Yes = 0b, No = 1b |

**Table 56. CFI Alternate Vendor-Specific Extended Query Parameter 84h Suspend Commands**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 00h | 84h | Parameter ID (Suspend Commands |
| 01h | 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) |
| 02h | 85h | Program suspend instruction code |
| 03h | 2Dh | Program suspend latency maximum (µs) |
| 04h | 8Ah | Program resume instruction code |
| 05h | 64h | Program resume to next suspend typical (µs) |
| 06h | 75h | Erase suspend instruction code |
| 07h | 2Dh | Erase suspend latency maximum (µs) |
| 08h | 7Ah | Erase resume instruction code |
| 09h | 64h | Erase resume to next suspend typical (µs) |

**Table 57. CFI Alternate Vendor-Specific Extended Query Parameter 88h Data Protection**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 00h | 88h | Parameter ID (Data Protection) |
| 01h | 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) |
| 02h | 0Ah | OTP size $2^N$ bytes, FFh = not supported |
| 03h | 01h | OTP address map format, 01h = FL-S format, FFh = not supported |
| 04h | xxh | Block Protect Type, model dependent<br>00h = FL-P, FL-S, FFh = not supported |
| 05h | xxh | Advanced Sector Protection type, model dependent<br>01h = FL-S ASP. |

**Table 58. CFI Alternate Vendor-Specific Extended Query Parameter 8Ch Reset Timing**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 00h | 8Ch | Parameter ID (Reset Timing) |
| 01h | 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) |
| 02h | 96h | POR maximum value |
| 03h | 01h | POR maximum exponent $2^N$ µs |
| 04h | FFh (without RESET# input)<br>23h (with RESET# input) | Hardware Reset maximum value |
| 05h | 00h | Hardware Reset maximum exponent $2^N$ µs |
| 06h | 23h | Software Reset maximum value, FFh = not supported |
| 07h | 00h | Software Reset maximum exponent $2^N$ µs |

**Table 59. CFI Alternate Vendor-Specific Extended Query Parameter 90h – Latency Code**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 00h | 90h | Parameter ID (Latency Code Table) |
| 01h | 56h | 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) |
| 02h | 06h | Number of rows |
| 03h | 0Eh | Row length in bytes |
| 04h | 46h | Start of header (row 1), ASCII “F” for frequency column header |
| 05h | 43h | ASCII “C” for Code column header |
| 06h | 03h | Read 3-byte address instruction |
| 07h | 13h | Read 4-byte address instruction |
| 08h | 0Bh | Read Fast 3-byte address instruction |
| 09h | 0Ch | Read Fast 4-byte address instruction |
| 0Ah | 3Bh | Read Dual Out 3-byte address instruction |
| 0Bh | 3Ch | Read Dual Out 4-byte address instruction |
| 0Ch | 6Bh | Read Quad Out 3-byte address instruction |
| 0Dh | 6Ch | Read Quad Out 4-byte address instruction |
| 0Eh | BBh | Dual I/O Read 3-byte address instruction |
| 0Fh | BCh | Dual I/O Read 4-byte address instruction |
| 10h | EBh | Quad I/O Read 3-byte address instruction |
| 11h | ECh | Quad I/O Read 4-byte address instruction |
| 12h | 32h | Start of row 2, SCK frequency limit for this row (50 MHz) |
| 13h | 03h | Latency Code for this row (11b) |
| 14h | 00h | Read mode cycles |
| 15h | 00h | Read latency cycles |
| 16h | 00h | Read Fast mode cycles |
| 17h | 00h | Read Fast latency cycles |
| 18h | 00h | Read Dual Out mode cycles |
| 19h | 00h | Read Dual Out latency cycles |
| 1Ah | 00h | Read Quad Out mode cycles |
| 1Bh | 00h | Read Quad Out latency cycles |
| 1Ch | 04h | Dual I/O Read mode cycles |
| 1Dh | 00h | Dual I/O Read latency cycles |
| 1Eh | 02h | Quad I/O Read mode cycles |
| 1Fh | 01h | Quad I/O Read latency cycles |
| 20h | 50h | Start of row 3, SCK frequency limit for this row (80 MHz) |
| 21h | 00h | Latency Code for this row (00b) |
| 22h | FFh | Read mode cycles (FFh = command not supported at this frequency) |
| 23h | FFh | Read latency cycles |
| 24h | 00h | Read Fast mode cycles |

**Table 59. CFI Alternate Vendor-Specific Extended Query Parameter 90h – Latency Code (Continued)**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 25h | 08h | Read Fast latency cycles |
| 26h | 00h | Read Dual Out mode cycles |
| 27h | 08h | Read Dual Out latency cycles |
| 28h | 00h | Read Quad Out mode cycles |
| 29h | 08h | Read Quad Out latency cycles |
| 2Ah | 04h | Dual I/O Read mode cycles |
| 2Bh | 00h | Dual I/O Read latency cycles |
| 2Ch | 02h | Quad I/O Read mode cycles |
| 2Dh | 04h | Quad I/O Read latency cycles |
| 2Eh | 5Ah | Start of row 4, SCK frequency limit for this row (90 MHz) |
| 2Fh | 01h | Latency Code for this row (01b) |
| 30h | FFh | Read mode cycles (FFh = command not supported at this frequency) |
| 31h | FFh | Read latency cycles |
| 32h | 00h | Read Fast mode cycles |
| 33h | 08h | Read Fast latency cycles |
| 34h | 00h | Read Dual Out mode cycles |
| 35h | 08h | Read Dual Out latency cycles |
| 36h | 00h | Read Quad Out mode cycles |
| 37h | 08h | Read Quad Out latency cycles |
| 38h | 04h | Dual I/O Read mode cycles |
| 39h | 01h | Dual I/O Read latency cycles |
| 3Ah | 02h | Quad I/O Read mode cycles |
| 3Bh | 04h | Quad I/O Read latency cycles |
| 3Ch | 68h | Start of row 5, SCK frequency limit for this row (108 MHz) |
| 3Dh | 02h | Latency Code for this row (10b) |
| 3Eh | FFh | Read mode cycles (FFh = command not supported at this frequency) |
| 3Fh | FFh | Read latency cycles |
| 40h | 00h | Read Fast mode cycles |
| 41h | 08h | Read Fast latency cycles |
| 42h | 00h | Read Dual Out mode cycles |
| 43h | 08h | Read Dual Out latency cycles |
| 44h | 00h | Read Quad Out mode cycles |
| 45h | 08h | Read Quad Out latency cycles |
| 46h | 04h | Dual I/O Read mode cycles |
| 47h | 02h | Dual I/O Read latency cycles |
| 48h | 02h | Quad I/O Read mode cycles |
| 49h | 05h | Quad I/O Read latency cycles |
| 4Ah | 85h | Start of row 6, SCK frequency limit for this row (133 MHz) |

**Table 59. CFI Alternate Vendor-Specific Extended Query Parameter 90h – Latency Code (Continued)**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 4Bh | 02h | Latency Code for this row (10b) |
| 4Ch | FFh | Read mode cycles (FFh = command not supported at this frequency) |
| 4Dh | FFh | Read latency cycles |
| 4Eh | 00h | Read Fast mode cycles |
| 4Fh | 08h | Read Fast latency cycles |
| 50h | FFh | Read Dual Out mode cycles |
| 51h | FFh | Read Dual Out latency cycles |
| 52h | FFh | Read Quad Out mode cycles |
| 53h | FFh | Read Quad Out latency cycles |
| 54h | FFh | Dual I/O Read mode cycles |
| 55h | FFh | Dual I/O Read latency cycles |
| 56h | FFh | Quad I/O Read mode cycles |
| 57h | FFh | Quad I/O Read latency cycles |

**Table 60. CFI Alternate Vendor-Specific Extended Query Parameter F0h RFU**

| Parameter Relative Byte Address Offset | Data | Description |
|---|---|---|
| 00h | F0h | Parameter ID (RFU) |
| 01h | 0Fh | 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) |
| 02h | FFh | RFU |
| ... | FFh | RFU |
| 10h | 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.

**Table 61. CFI Alternate Vendor-Specific Extended Query Parameter A5h, JEDEC SFDP Rev B**

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 00h | — | N/A | A5h | CFI Parameter ID (A5h = JEDEC SFDP) |
| 01h | — | N/A | 80h | 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) |
| 02h | 00h | JEDEC Basic Flash Parameter Dword-1 | E7h | Start of SFDP JEDEC parameter, located at 1120h in the overall SFDP address space.<br>Bits 7:5 = unused = 111b<br>Bit 4:3 = 06h is status register write instruction & status register is default non-volatile= 00b<br>Bit 2 = Program Buffer > 64Bytes = 1 Bits 1:0 = Uniform 4KB erase unavailable = 11b |
| 03h | 01h | | FFh | Bits 15:8 = Uniform 4KB erase opcode = not supported = FFh |
| 04h | 02h | | F3h (FLxxxSAG) | 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, No = 0h<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 |

**Table 61. CFI Alternate Vendor-Specific Extended Query Parameter A5h, JEDEC SFDP Rev B (Continued)**

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 06h | 04h | JEDEC Basic Flash Parameter Dword-2 | FFh | Density in bits, zero based, 128Mb = 07FFFFFFh |
| 07h | 05h | | FFh | |
| 08h | 06h | | FFh | |
| 09h | 07h | | 07h | |
| 0Ah | 08h | JEDEC Basic Flash Parameter Dword-3 | 44h | Bits 7:5 = number of Quad I/O Mode cycles = 010b<br>Bits 4:0 = number of Quad I/O Dummy cycles = 00100b for default latency code 00b |
| 0Bh | 09h | | EBh | Quad I/O instruction code |
| 0Ch | 0Ah | | 08h | Bits 23:21 = number of Quad Out Mode cycles = 000b<br>Bits 20:16 = number of Quad Out Dummy cycles = 01000b |
| 0Dh | 0Bh | | 6Bh | Quad Out instruction code |
| 0Eh | 0Ch | JEDEC Basic Flash Parameter Dword-4 | 08h | Bits 7:5 = number of Dual Out Mode cycles = 000b<br>Bits 4:0 = number of Dual Out Dummy cycles = 01000b for default latency code |
| 0Fh | 0Dh | | 3Bh | Dual Out instruction code |
| 10h | 0Eh | | 80h | Bits 23:21 = number of Dual I/O Mode cycles<br>20:16 = number of Dual I/O Dummy cycles<br>Default Latency code = 00b |
| 11h | 0Fh | | BBh | Dual I/O instruction code |
| 12h | 10h | JEDEC Basic Flash Parameter Dword-5 | EEh | Bits 7:5 RFU = 111b<br>Bit 4 = Quad All not supported = 0b<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 | | FFh | Bits 23:21 = number of Quad All Mode cycles = 111b<br>Bits 20:16 = number of Quad All Dummy cycles = 11111b |
| 1Dh | 1Bh | | FFh | Quad All instruction code |
| 1Eh | 1Ch | JEDEC Basic Flash Parameter Dword-8 | 0Ch | Erase type 1 size $2^N$ Bytes = 4KB = 0Ch (for Hybrid Sector Initial Delivery State) |
| 1Fh | 1Dh | | 20h | Erase type 1 instruction |
| 20h | 1Eh | | 10h | Erase type 2 size $2^N$ Bytes = 64KB = 10h (for Hybrid Sector Initial Delivery State) |
| 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 (if Uniform Sectors enabled) |
| 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 |

**Table 61. CFI Alternate Vendor-Specific Extended Query Parameter A5h, JEDEC SFDP Rev B (Continued)**

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 26h | 24h | JEDEC Basic Flash Parameter Dword-10 | 82h | Bits 31:30 = Erase type 4 Erase, Typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = RFU = 11b<br>Bits 29:25 = Erase type 4 Erase, Typical time count = RFU = 11111b ( typ erase time = count +1 * units = 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 = 00011b ( typ erase time = count +1 * units = 4*128mS = 512mS)<br>Bits 17:16 = Erase type 2 Erase, Typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 128mS = 10b<br>Bits 15:11 = Erase type 2 Erase, Typical time count = 00000b ( typ erase time = count +1 * units = 1*128mS = 128mS)<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 = 01000b ( typ erase time = count +1 * units = 9*16mS = 144mS)<br>Bits 3:0 = Multiplier from typical erase time to maximum erase time = 2*(N+1), N=2h = 6x multiplier<br><br>Binary Fields: 11-11111-10-00011-10-00000-01-01000-0010<br>Nibble Format: 1111_1111_0000_1110_0000_0010_1000_0010<br>Hex Format: FF_0E_02_82 |
| 27h | 25h | | 02h | |
| 28h | 26h | | 0Eh | |
| 29h | 27h | | FFh | |
| 2Ah | 28h | JEDEC Basic Flash Parameter Dword-11 | 92h | 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) = 4s = 10b<br>Bits 28:24 = Chip Erase, Typical time count, (count+1)*units, count = 01000b, ( typ Program time = count +1 * units = 9*4S = 36S<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 = 01001b, ( typ Program time = count +1 * units = 10*64uS = 640uS)<br>Bits 7:4 = Page size 2^N, N=9h, = 512B 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>Binary Fields: 1-10-01000-0-0000-1-1100-1-01001-1001-0010<br>Nibble Format: 1100_1000_0000_0111_0010_1001_1001_0010<br>Hex Format: C8_07_29_92 |
| 2Bh | 29h | | 29h | |
| 2Ch | 2Ah | | 07h | |
| 2Dh | 2Bh | | C8h | |
| 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 + 1xxxb: The erase and program restrictions in bits 5:4 are sufficient = 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 = 1100b<br><br>Binary Fields: 0-10-00101-0001-10-00101-0001-1-1110-1100<br>Nibble Format: 0100_0101_0001_1000_1010_0011_1110_1100<br>Hex Format: 45_18_A3_EC |
| 2Fh | 2Dh | | A3h | |
| 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 61. CFI Alternate Vendor-Specific Extended Query Parameter A5h, JEDEC SFDP Rev B (Continued)**

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 36h | 34h | JEDEC Basic Flash Parameter Dword-14 | F7h | Bit 31 = Deep Power Down Supported = not supported = 1<br>Bits 30:23 = Enter Deep Power Down Instruction = not supported = FFh<br>Bits 22:15 = Exit Deep Power Down Instruction = not supported = FFh<br>Bits 14:13 = Exit Deep Power Down to next operation delay units = (00b: 128ns, 01b: 1us, 10b: 8us, 11b: 64us) = 64us = 11b<br>Bits 12:8 = Exit Deep Power Down to next operation delay count = 11111b, Exit Deep Power Down to next operation delay = (count+1)*units = not supported<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>Bits 1:0 = RFU = 11b<br><br>Binary Fields: 1-11111111-11111111-11-11111-1111-01-11<br>Nibble Format: 1111_1111_1111_1111_1111_1111_1111_0111<br>Hex Format: FF_FF_FF_F7 |
| 37h | 35h | | FFh | |
| 38h | 36h | | FFh | |
| 39h | 37h | | FFh | |
| 3Ah | 38h | JEDEC Basic Flash Parameter Dword-15 | 00h | 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 (SFDP spec calls this Status Register 2, FL127S calls this Configuration Register 1). Status register 1 is read using Read Status instruction 05h. Status register 2 (FL127S Configuration Register 1) 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 Bits[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>= 0_0000b: 4-4-4 not supported<br>= 00000b<br>Bits 3:0 = 4-4-4 mode disable sequences<br>= 0000b: 4-4-4 not supported<br>= 0000b<br><br>Binary Fields: 11111111-0-101-1101-111101-1-00000-0000<br>Nibble Format: 1111_1111_0101_1101_1111_0110_0000_0000<br>Hex Format: FF_5D_F6_00 |
| 3Bh | 39h | | F6h | |
| 3Ch | 3Ah | | 5Dh | |
| 3Dh | 3Bh | | FFh | |

**Table 61. CFI Alternate Vendor-Specific Extended Query Parameter A5h, JEDEC SFDP Rev B (Continued)**

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 3Eh | 3Ch | JEDEC Basic Flash Parameter Dword-16 | F0h | Bits 31:24 = Enter 4-Byte Addressing<br>= xxxx_1xxxb: 8-bit volatile bank register used to define A[30:A24] bits. MSB (bit[7]) is used to enable/disable 4-byte address mode. When MSB is set to '1', 4-byte address mode is active and A[30:24] bits are don't care. Read with instruction 16h. Write instruction is 17h with 1 byte of data. When MSB is cleared to '0', select the active 128 Mbit segment by setting the appropriate A[30:24] bits and use 3-Byte addressing.<br>+ xx1x_xxxxb: Supports dedicated 4-Byte address instruction set. Consult vendor data sheet for the instruction set definition or look for 4 Byte Address Parameter Table.<br>+ 1xxx_xxxxb: Reserved<br>= 10101000b<br>Bits 23:14 = Exit 4-Byte Addressing<br>= xx_xxxx_1xxxb: 8-bit volatile bank register used to define A[30:A24] bits. MSB (bit[7]) is used to enable/disable 4-byte address mode. When MSB is cleared to '0', 3-byte address mode is active and A30:A24 are used to select the active 128 Mbit memory segment. Read with instruction 16h. Write instruction is 17h, data length is 1 byte.<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 = 1111101000b<br>Bits 13:8 = Soft Reset and Rescue Sequence Support<br>= x0_1xxxb: issue instruction F0h<br>+ 1x_xxxxb: exit 0-4-4 mode is required prior to other reset sequences above if the device may be operating in this mode. = 101000b<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 instruction is used to enable writing of the register.<br>+ x1x_xxxxb: Reserved<br>+ 1xx_xxxxb: Reserved<br>= 1110000b<br><br>Binary Fields: 10101000-1111101000-101000-1-1110000<br>Nibble Format: 1010_1000_1111_1010_0010_1000_1111_0000<br>Hex Format: A8_FA_28_F0 |
| 3Fh | 3Dh | | 28h | |
| 40h | 3Eh | | FAh | |
| 41h | 3Fh | | A8h | |
| 42h | 40h | JEDEC Sector Map Parameter Dword-1 Config. Detect-1 | FCh | Bits 31:24 = Read data mask = 10000000b: Select bit 7 of the data byte for D8h_O value<br>Bits 23:22 = Configuration detection command address length = 00b: No address<br>Bits 21:20 = RFU = 11b<br>Bits 19:16 = Configuration detection command latency = 0000b: zero latency<br>Bits 15:8 = Configuration detection instruction = 07h: Read status register 2<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Command Descriptor = 0<br>Bit 0 = not the end descriptor = 0<br><br>Binary Fields: 10000000-00-11-0000-00000111-111111-0-0<br>Nibble Format: 1000_0000_0011_0000_0000_0111_1111_1100<br>Hex Format: 80_30_07_FC |
| 43h | 41h | | 07h | |
| 44h | 42h | | 30h | |
| 45h | 43h | | 80h | |
| 46h | 44h | JEDEC Sector Map Parameter Dword-2 Config. Detect-1 | FFh | Bits 31:0 = Sector map configuration detection command address = FFFFh: no address |
| 47h | 45h | | FFh | |
| 48h | 46h | | FFh | |
| 49h | 47h | | FFh | |
| 4Ah | 48h | JEDEC Sector Map Parameter Dword-3 Config. Detect-2 | FDh | Bits 31:24 = Read data mask = 00000100b: Select bit 2 of the data byte for TBPARM value<br>Bits 23:22 = Configuration detection command address length = 00b: No address<br>Bits 21:20 = RFU = 11b<br>Bits 19:16 = Configuration detection command latency = 0000b: zero latency<br>Bits 15:8 = Configuration detection instruction = 35h: Read configuration register 1<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Command Descriptor = 0<br>Bit 0 = The end descriptor = 1<br><br>Binary Fields: 00000100-00-11-0000-00110101-111111-0-1<br>Nibble Format: 0000_0100_0011_0000_0011_0101_1111_1101<br>Hex Format: 04_30_35_FD |
| 4Bh | 49h | | 35h | |
| 4Ch | 4Ah | | 30h | |
| 4Dh | 4Bh | | 04h | |
| 4Eh | 4Ch | JEDEC Sector Map Parameter Dword-4 Config. Detect-2 | FFh | Bits 31:0 = Sector map configuration detection command address = FFFFh: no address |
| 4Fh | 4Dh | | FFh | |
| 50h | 4Eh | | FFh | |
| 51h | 4Fh | | FFh | |

**Table 61. CFI Alternate Vendor-Specific Extended Query Parameter A5h, JEDEC SFDP Rev B (Continued)**

| CFI Parameter Relative Byte Address Offset | SFDP Parameter Relative Byte Address Offset | SFDP Dword Name | Data | Description |
|---|---|---|---|---|
| 52h | 50h | JEDEC Sector Map Parameter Dword-5 Config-0 Header | FEh | Bits 31:24 = RFU = FFh<br>Bits 23:16 = Region count (Dwords -1) = 01h: Two 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 |
| 53h | 51h | | 00h | |
| 54h | 52h | | 01h | |
| 55h | 53h | | FFh | |
| 56h | 54h | JEDEC Sector Map Parameter Dword-6 Config-0 Region-0 | F3h | Bits 31:8 = Region size = 0000FFh:<br>Region size as count-1 of 256 Byte units = 16 x 4KB sectors = 64KB<br>Count = 64KB/256 = 256, value = count -1 = 256 -1 = 255 = FFh<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 = 1b<br>---Erase Type 2 is 64KB erase and is supported in the 4KB sector region<br>Bits 0 = Erase Type 1 support = 1b<br>---Erase Type 1 is 4KB erase and is supported in the 4KB sector region |
| 57h | 55h | | FFh | |
| 58h | 56h | | 00h | |
| 59h | 57h | | 00h | |
| 5Ah | 58h | JEDEC Sector Map Parameter Dword-7 Config-0 Region-1 | F2h | Bits 31:8 = Region size = 00FEFFh:<br>Region size as count-1 of 256 Byte units = 255 x 64KB sectors = 16320KB<br>Count = 16320KB/256 = 65280, value = count -1 = 65280 -1 = 65279 = FEFFh<br>Bits 4:7 = 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 |
| 5Bh | 59h | | FFh | |
| 5Ch | 5Ah | | FEh | |
| 5Dh | 5Bh | | 00h | |
| 5Eh | 5Ch | JEDEC Sector Map Parameter Dword-8 Config-1 Header | FEh | Bits 31:24 = RFU = FFh<br>Bits 23:16 = Region count (Dwords -1) = 01h: Two regions<br>Bits 15:8 = Configuration ID = 01h: 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 |
| 5Fh | 5Dh | | 01h | |
| 60h | 5Eh | | 01h | |
| 61h | 5Fh | | FFh | |
| 62h | 60h | JEDEC Sector Map Parameter Dword-9 Config-1 Region-0 | F2h | Bits 31:8 = Region size = 00FEFFh:<br>Region size as count-1 of 256 Byte units = 255 x 64KB sectors = 16320KB<br>Count = 16320KB/256 = 65280, value = count -1 = 65280 -1 = 65279 = FEFFh<br>Bits 4:7 = 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 |
| 63h | 61h | | FFh | |
| 64h | 62h | | FEh | |
| 65h | 63h | | 00h | |
| 66h | 64h | JEDEC Sector Map Parameter Dword-10 Config-1 Region-1 | F3h | Bits 31:8 = Region size = 0000FFh:<br>Region size as count-1 of 256 Byte units = 16 x 4KB sectors = 64KB<br>Count = 64KB/256 = 256, value = count -1 = 256 -1 = 255 = FFh<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 = 1b<br>---Erase Type 2 is 64KB erase and is 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 | | FFh | |
| 68h | 66h | | 00h | |
| 69h | 67h | | 00h | |
| 6Ah | 68h | JEDEC Sector Map Parameter Dword-11 Config-2 Header | FEh | Bits 31:24 = RFU = FFh Bits 23:16 = Region count (Dwords -1) = 00h: One region<br>Bits 15:8 = Configuration ID = 02h: Uniform 256KB sectors<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Map Descriptor = 1<br>Bit 0 = The end descriptor = 0 |
| 6Bh | 69h | | 02h | |
| 6Ch | 6Ah | | 00h | |
| 6Dh | 6Bh | | FFh | |

**Table 61. CFI Alternate Vendor-Specific Extended Query Parameter A5h, JEDEC SFDP Rev B (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-12 Config-2 Region-0 | F4h | Bits 31:8 = Region size = 00FFFFh:<br>Region size as count-1 of 256 Byte units = 16MB/256 = 64K<br>Count = 65536, value = count -1 = 65536 -1 = 65535 = FFFFh<br>Bits 4:7 = 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 |
| 6Fh | 6Dh | | FFh | |
| 70h | 6Eh | | FFh | |
| 71h | 6Fh | | 00h | |
| 72h | 70h | JEDEC Sector Map Parameter Dword-13 Config-3 Header | FFh | Bits 31:24 = RFU = FFh<br>Bits 23:16 = Region count (Dwords -1) = 00h: One region<br>Bits 15:8 = Configuration ID = 03h: Uniform 256KB sectors<br>Bits 7:2 = RFU = 111111b<br>Bit 1 = Map Descriptor = 1<br>Bit 0 = The end descriptor = 1 |
| 73h | 71h | | 03h | |
| 74h | 72h | | 00h | |
| 75h | 73h | | FFh | |
| 76h | 74h | JEDEC Sector Map Parameter Dword-14 Config-3 Region-0 | F4h | Bits 31:8 = Region size = 00FFFFh:<br>Region size as count-1 of 256 Byte units = 16MB/256 = 64K<br>Count = 65536, value = count -1 = 65536 -1 = 65535 = FFFFh<br>Bits 4:7 = 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 |
| 77h | 75h | | FFh | |
| 78h | 76h | | FFh | |
| 79h | 77h | | 00h | |
| 7Ah | 78h | 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 = 0<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 |
| 7Bh | 79h | | 0Eh | |
| 7Ch | 7Ah | | FFh | |
| 7Dh | 7Bh | | FFh | |
| 7Eh | 7Ch | 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 |
| 7Fh | 7Dh | | DCh | |
| 80h | 7Eh | | DCh | |
| 81h | 7Fh | | FFh | |

## 12.3 Device ID and Common Flash Interface (ID-CFI) ASO Map — Automotive Only

The CFI Primary Vendor-Specific Extended Query is extended to include Electronic Marking information for device traceability

| Address | Data Field | # of bytes | Data Format | Example of Actual Data | Hex Read Out of Example Data |
|---|---|---|---|---|---|
| (SA) + 0180h | Size of Electronic Marking | 1 | Hex | 20 | 14h |
| (SA) + 0181h | Revision of Electronic Marking | 1 | Hex | 1 | 01h |
| (SA) + 0182h | Fab Lot # | 8 | ASCII | LD87270 | 4Ch, 44h, 38h, 37h, 32h, 37h, 30h, FFh |
| (SA) + 018Ah | Wafer # | 1 | Hex | 23 | 17h |
| (SA) + 018Bh | Die X Coordinate | 1 | Hex | 10 | 0Ah |
| (SA) + 018Ch | Die Y Coordinate | 1 | Hex | 15 | 0Fh |
| (SA) + 018Dh | Class Lot # | 7 | ASCII | BR33150 | 42h, 52h, 33h, 33h, 31h, 35h, 30h |
| (SA) + 0194h | Reserved for Future | 12 | N/A | N/A | FFh, FFh, FFh, FFh, FFh, FFh, FFh, FFh, FFh, FFh, FFh, FFh |

Fab Lot # + Wafer # + Die X Coordinate + Die Y Coordinate gives a unique ID for each device.

## 12.4 Registers

The register maps are copied in this section as a quick reference. See Registers on page 50 for the full description of the register contents.

**Table 62. Status Register 1 (SR1)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | SRWD | Status Register Write Disable | Non-Volatile | 0 | 1 = Locks state of SRWD, BP, and configuration register bits when WP# is low by ignoring WRR command<br>0 = No protection, even when WP# is low |
| 6 | P_ERR | Programming Error Occurred | Volatile, Read only | 0 | 1 = Error occurred<br>0 = No Error |
| 5 | E_ERR | Erase Error Occurred | Volatile, Read only | 0 | 1= Error occurred<br>0 = No Error |
| 4 | BP2 | Block Protection | Volatile if CR1[3]=1, Non-Volatile if CR1[3]=0 | 1 if CR1[3]=1, 0 when shipped from Cypress | Protects selected range of sectors (Block) from Program or Erase |
| 3 | BP1 | | | | |
| 2 | BP0 | | | | |
| 1 | WEL | Write Enable Latch | Volatile | 0 | 1 = Device accepts Write Registers (WRR), program or erase commands<br>0 = Device ignores Write Registers (WRR), program or erase commands<br>This bit is not affected by WRR, only WREN and WRDI commands affect this bit. |
| 0 | WIP | Write in Progress | Volatile, Read only | 0 | 1= Device Busy, a Write Registers (WRR), program, erase or other operation is in progress<br>0 = Ready Device is in standby mode and can accept commands |

**Table 63. Configuration Register (CR1)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | LC1 | Latency Code | Non-Volatile | 0 | Selects number of initial read latency cycles<br>See Latency Code Tables |
| 6 | LC0 | | | 0 | |
| 5 | TBPROT | Configures Start of Block Protection | OTP | 0 | 1 = BP starts at bottom (Low address)<br>0 = BP starts at top (High address) |
| 4 | RFU | RFU | OTP | 0 | Reserved for Future Use |
| 3 | BPNV | Configures BP2-0 in Status Register | OTP | 0 | 1 = Volatile<br>0 = Non-Volatile |
| 2 | TBPARM | 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 devices. |
| 1 | QUAD | Puts the device into Quad I/O operation | Non-Volatile | 0 | 1 = Quad<br>0 = Dual or Serial |
| 0 | FREEZE | Lock current state of BP2-0 bits in Status Register, TBPROT and TBPARM in Configuration Register, and OTP regions | Volatile | 0 | 1 = Block Protection and OTP locked<br>0 = Block Protection and OTP un-locked |

**Table 64. Status Register 2 (SR2)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | D8h_O | Block Erase Size | OTP | 0 | 1 = 256 kB Erase (Uniform sectors).<br>0 = 64 kB Erase (Hybrid 4 kB / 64 kB sectors). |
| 6 | 02h_O | Page Buffer Wrap | OTP | 0 | 1 = Wrap at 512B.<br>0 = Wrap at 256B. |
| 5 | IO3R_O | IO3 Reset | OTP | 0 | 1 = IO3 alternate function is RESET#.<br>0 = IO3 alternate function is HOLD#. |
| 4 | RFU | Reserved | | 0 | Reserved for Future Use. |
| 3 | RFU | Reserved | | 0 | Reserved for Future Use. |
| 2 | RFU | Reserved | | 0 | Reserved for Future Use |
| 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. |

**Table 65. Bank Address Register (BAR)**

| Bits | Field Name | Function | Type | Default State | Description |
|---|---|---|---|---|---|
| 7 | EXTADD | Extended Address Enable | Volatile | 0b | 1 = 4-byte (32-bits) addressing required from command.<br>0 = 3-byte (24-bits) addressing from command + Bank Address |
| 6 to 2 | RFU | Reserved | Volatile | 00000b | Reserved for Future Use |
| 1 | BA25 | Bank Address | Volatile | 0 | RFU for lower density devices |
| 0 | BA24 | Bank Address | Volatile | 0 | RFU for lower density device |

**Table 66. 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 | | Reserved for Future Use. |
| 7 | RFU | Reserved | OTP | | Reserved for Future Use. |
| 6 | RFU | Reserved | OTP | 1 | Reserved for Future Use. |
| 5 | RFU | Reserved | OTP | | Reserved for Future Use. |
| 4 | RFU | Reserved | OTP | | Reserved for Future Use. |
| 3 | RFU | Reserved | OTP | | 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 | RFU | Reserved | OTP | 1 | Reserved for Future Use. |

**Table 67. 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 0. |

**Table 68. 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 | Persistent Protection Mode = 1<br>Password Protection Mode = 0 | 0 = PPB array protected until next power cycle or hardware reset<br>1 = PPB array may be programmed or erased |

**Table 69. 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 or PPBP command is erased to “1”, not protecting that sector from program or erase operations. |

**Table 70. DYB Access Register (DYBAR)**

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

## 12.5 Initial Delivery State

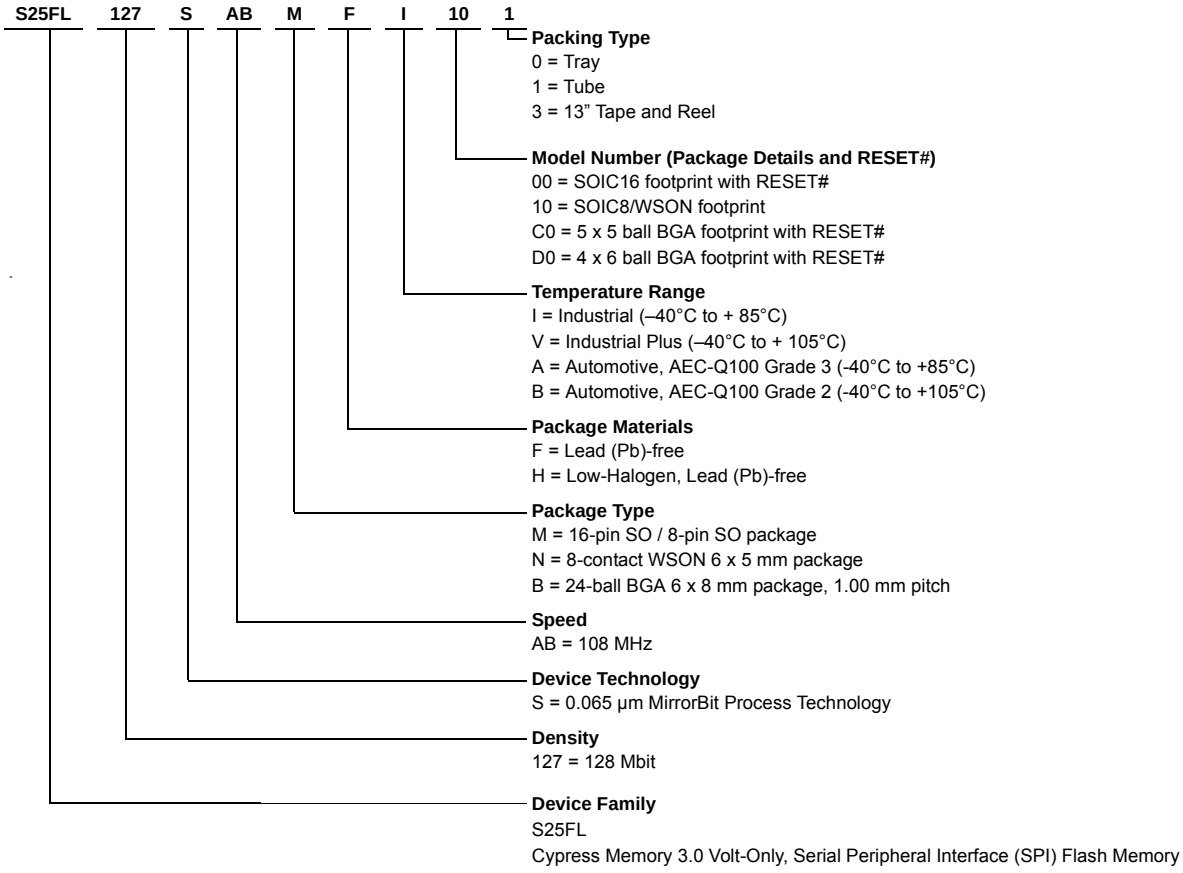
The device is shipped from Cypress with non-volatile bits set as follows:

- 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 Status Register 1 contains 00h (all SR1 bits are cleared to 0's).
- The Configuration Register 1 contains 00h.
- The Autoboot register contains 00h.
- The Password Register contains FFFFFFFF-FFFFFFFFh
- All PPB bits are 1.

# Ordering Information

## 13. Ordering Information

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




### Valid Combinations

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 71. S25FL127S Valid Combinations**

| Valid Combinations | | | | | |
|---|---|---|---|---|---|
| **Base Ordering Part Number** | **Speed Option** | **Package and Temperature** | **Model Number** | **Packing Type** | **Package Marking (1)** |
| S25FL127S | AB | MFI, MFV | 00, 10 | 0, 1, 3 | FL127 + S + (Temp) + F + (Model Number) |
| | AB | NFI, NFV | 10 | | FL127 + S + (Temp) + F + (Model Number) |
| | AB | BHI, BHV | C0, D0 | 0, 3 | FL127 + S + (Temp) + H + (Model Number) |

***Note:***

*1. Example, S25FL127SABMFI100 package marking would be FL127SIF10*

### 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 72. S25FL127S Valid Combinations — Automotive Grade / AEC-Q100**

| Valid Combinations | | | | | |
|---|---|---|---|---|---|
| **Base Ordering Part Number** | **Speed Option** | **Package and Temperature** | **Model Number** | **Packing Type** | **Package Marking (1)** |
| S25FL127S | AB | MFA, MFB | 00, 10 | 0, 1, 3 | FL127 + S + (Temp) + F + (Model Number) |
| | AB | NFA, NFB | 10 | | FL127 + S + (Temp) + F + (Model Number) |
| | AB | BHA, BHB | C0, D0 | 0, 3 | FL127 + S + (Temp) + H + (Model Number) |

# 14. Revision History

| Document Title: S25FL127S, 128-Mbit (16 Mbyte) 3.0 V SPI Flash Memory<br>Document Number: 001-98282 | | | | |
|---|---|---|---|---|
| **Rev.** | **ECN No.** | **Orig. of Change** | **Submission Date** | **Description of Change** |
| ** | – | BWHA | 12/11/2012 | Initial release |
| *A | – | BWHA | 04/25/2013 | Global:<br>Data sheet designation updated from Advance Information to Preliminary<br>Performance Summary: Maximum Read Rates table: corrected Dual and Quad Read 'Clock Rate' and 'Mbytes/s'<br>Migration Notes: FL Generations Comparison table: corrected Dual Read Speed and Quad Read Speed for FL127S<br>DC Characteristics: DC Characteristics table: corrected ICC1 Test Conditions for Quad<br>Capacitance Characteristics: Capacitance table: updated note<br>AC Characteristics: AC Characteristics table: corrected Max value for FSCK,C dual and quad command<br><br>Physical Interface:<br>8-pin Plastic Small Outline Package (SO) figure: corrected marking<br>8-Contact USON 6x5 mm, Top View figure: corrected marking<br>24-Ball BGA, 5 x 5 Ball Footprint (FAB024), Top View figure: removed VIO<br>24-Ball BGA, 4 x 6 Ball Footprint (FAC024), Top View: removed VIO<br><br>Command Set Summary: S25FL127S Command Set (sorted by function) table: corrected Maximum Frequency for ABRD, DOR, 4DOR, QOR, 4QOR, DIOR, 4DIOR, QIOR, 4QIOR<br>Embedded Algorithm Performance Tables:<br>Added paragraph<br>Program and Erase Performance table:<br>- added 'Erase per Sector' Parameter<br>- added note<br><br>Software Interface Reference: FL127S Command Set (sorted by instruction) table: corrected Maximum Frequency for DOR, 4DOR, QOR, 4QOR, DIOR, 4DIOR, QIOR, 4QIOR<br>Serial Flash Discoverable Parameters (SFDP) Address Map: CFI Alternate Vendor-Specific Extended Query Parameter 90h – Latency Code table: corrected Description for 68h<br>Ordering Information: Valid Combinations table: corrected Package Marking and Note |
| *B | – | BWHA | 07/26/2013 | Features: Added 16-pin SOIC package<br>Glossary: Updated description of Page<br>Signal Descriptions: Changed description of RESET#<br>Hardware Reset (RESET#): Changed RESET# description from "may be left unconnected in the host system if not used" to "should be left unconnected in the host system if not used"<br>Separate RESET# Input Initiated Hardware (Warm) Reset: Changed RESET# description from "may be left unconnected" to "should be left unconnected if not used"<br>SOIC 16-Lead Package: Added section<br>Ordering Information: Added 16-pin SOIC package |
| *C | – | BWHA | 09/11/2013 | Global: Replaced USON with WSON |
| *D | – | BWHA | 11/15/2013 | Global: Data sheet designation updated from Preliminary to Full Production<br>Physical Interface: Updated 8-pin Plastic Small Outline Package (SO) figure<br>Command Set Summary: S25FL127S Command Set (sorted by function) table: added RSFDP command<br>Command Summary: FL127S Command Set (sorted by instruction) table: added RSFDP command |

| Document Title: S25FL127S, 128-Mbit (16 Mbyte) 3.0 V SPI Flash Memory<br>Document Number: 001-98282 | | | | |
|---|---|---|---|---|
| **Rev.** | **ECN No.** | **Orig. of Change** | **Submission Date** | **Description of Change** |
| *E | – | BWHA | 05/28/2015 | JEDEC JESD216 Serial Flash Discoverable Parameters (SFDP) Space:<br>Changed JESD216 to JESD216B<br>Serial Flash Discoverable Parameters (SFDP) Address Map:<br>Updated section<br>Updated SFDP Overview Map table<br><br>SFDP Header Field Definitions:<br>Updated SFDP Header table<br>Updated CFI Alternate Vendor-Specific Extended Query Parameter A5h, JEDEC SFDP table |
| *F | 4871631 | BWHA | 08/24/2015 | Replaced “Automotive Temperature Range” with “Industrial Plus Temperature Range” in all instances across the document.<br>Updated to Cypress template. |
| *G | 5348965 | TOCU | 07/13/2016 | Updated Serial Flash Discoverable Parameters (SFDP) Address Map:<br>Updated Device ID and Common Flash Interface (ID-CFI) Address Map:<br>Updated Field Definitions:<br>Updated Table 61 (Updated entire table).<br>Updated to new template. |
| *H | 5670970 | ECAO | 03/24/2017 | Updated Cypress logo.<br>Updated Sales page.<br>Updated Features:<br>Added ECC information.<br>Added Automotive temperature range support.<br>Updated Section 1.3 Glossary on page 7:<br>Added ECC definition.<br>Updated Section 4.2.1 Temperature Ranges on page 25:<br>Added Automotive temperature range support.<br>Updated Table 19, Register Descriptions on page 50:<br>Added ECC Status Register information.<br>Added Section 7.6.6 ECC Status Register (ECCSR) on page 56<br>Updated Section 8.1.2 Programming OTP Memory Space on page 59:<br>Added ECC information.<br>Updated Table 36, S25FL127S Command Set (sorted by function) on page 68:<br>Added ECC Read command information.<br>Updated Section 9.3 Register Access Commands on page 75:<br>Added Section 9.3.11 ECC Status Register Read (ECCRD 18h) on page 82<br>Updated Section 9.5 Program Flash Array Commands on page 93:<br>Added Section 9.5.1.1 Automatic ECC on page 93.<br>Updated Table 44, FL127S Command Set (sorted by instruction) on page 113:<br>Updated Figure 35, S0C008 — 8-Lead Plastic Small Outline Package (208-mil Body Width) on page 38.<br>Updated Figure 37, S03016 — 16-Lead Wide Plastic Small Outline Package (300-mil Body Width) on page 40.<br>Updated Figure 41, FAB024 — 24-ball Ball Grid Array (8 x 6 mm) Package on page 44.<br>Updated Figure 43, FAC024 — 24-ball Ball Grid Array (6 x 8 mm) Package on page 46. |
| *I | 5767544 | ECAO | 06/08/2017 | Updated Ordering Information:<br>No change in part numbers.<br>Updated Table 71 and Table 72:<br>Fixed typo (Replaced “S25FL128S, S25FL256S” with “S25FL127S” in title). |

# 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

### Cypress Developer Community

Forums | WICED IOT Forums | Projects | Video | Blogs | Training | Components

### Technical Support

cypress.com/support

© Cypress Semiconductor Corporation, 2012-2017. 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. 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.

LifeElectronics
Живое партнерство

**ООО «ЛайфЭлектроникс»** **“LifeElectronics” LLC**

*ИНН 7805602321 КПП 780501001 Р/С 40702810122510004610 ФАКБ "АБСОЛЮТ БАНК" (ЗАО) в г.Санкт-Петербурге К/С 30101810900000000703 БИК 044030703*

*Компания «Life Electronics» занимается поставками электронных компонентов импортного и отечественного производства от производителей и со складов крупных дистрибьюторов Европы, Америки и Азии.*

*С конца 2013 года компания активно расширяет линейку поставок компонентов по направлению коаксиальный кабель, кварцевые генераторы и конденсаторы (керамические, пленочные, электролитические), за счёт заключения дистрибьюторских договоров*

*Мы предлагаем:*

- *Конкурентоспособные цены и скидки постоянным клиентам.*
- *Специальные условия для постоянных клиентов.*
- *Подбор аналогов.*
- *Поставку компонентов в любых объемах, удовлетворяющих вашим потребностям.*

- *Приемлемые сроки поставки, возможна ускоренная поставка.*
- *Доставку товара в любую точку России и стран СНГ.*
- *Комплексную поставку.*
- *Работу по проектам и поставку образцов.*
- *Формирование склада под заказчика.*

- *Сертификаты соответствия на поставляемую продукцию (по желанию клиента).*
- *Тестирование поставляемой продукции.*
- *Поставку компонентов, требующих военную и космическую приемку.*
- *Входной контроль качества.*
- *Наличие сертификата ISO.*

*В составе нашей компании организован Конструкторский отдел, призванный помогать разработчикам, и инженерам.*

*Конструкторский отдел помогает осуществить:*

- *Регистрацию проекта у производителя компонентов.*
- *Техническую поддержку проекта.*
- *Защиту от снятия компонента с производства.*
- *Оценку стоимости проекта по компонентам.*
- *Изготовление тестовой платы монтаж и пусконаладочные работы.*

*Тел: +7 (812) 336 43 04 (многоканальный)*
*Email: org@lifeelectronics.ru*

***www.lifeelectronics.ru***