

# 256 Mb (32 MB) / 512 Mb (64MB) FL-S Flash

## SPI Dual-Quad, 3.0 V

### Features

- Density
  - 256 Mb (32 MB)
  - 512 Mb (64 MB)
- SPI
  - SPI clock polarity and phase modes 0 and 3
  - DDR option
  - Extended addressing: 24- or 32-bit address options
- READ commands
  - Dual-Quad SPI Quad Read: 104 MHz clock rate (104 MB/s)
  - Dual-Quad SPI Quad DDR Read: 80 MHz clock rate (160 MBps)
  - Normal, Fast, Quad, Quad DDR
  - 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 (3 MBps)
  - 512-byte or 1024-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 (1 MBps)
  - Hybrid sector size option – physical set of thirty two 8-KB sectors at top or bottom of address space with all remaining sectors of 128 KB
  - Uniform sector option – always erase 512-KB blocks for software compatibility with higher density and future devices.
- Cycling endurance
  - 100,000 program-erase cycles on any sector, minimum
- Data retention
  - 20 year data retention, minimum
- Security features
  - Separate one-time programmable (OTP) array of 2048 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
- 65-nm MIRRORBIT™ technology with Eclipse architecture
- Core supply voltage: 2.7 V to 3.6 V

## 256 Mb (32 MB) / 512 Mb (64MB) FL-S Flash SPI Dual-Quad, 3.0 V

---



### Features

- 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)
  - 16-lead SOIC (300 mil)
- Software features
  - Program suspend and resume
  - Erase suspend and resume
  - Status Register provides status of embedded erase or programming operation
  - CFI compliant — allows host system to identify the flash device and determine its capabilities
  - User-configurable Configuration Register
- Hardware features
  - Hardware Reset input (RESET#) - resets device to standby state

## Performance summary

### Maximum read rates, SDR dual-quad SPI

Command	Clock rate (MHz)	MBps
Read	50	12.5
Fast read	133	33
Quad read	104	104

### Maximum read rates, DDR dual-quad SPI

Command	Clock rate (MHz)	MBps
DDR quad read	80	160

### Typical program and erase rates, dual-quad SPI

Operation	KBps
Page programming (512-byte page buffer)	2000
Page programming (1024-byte page buffer)	3000
8-KB physical sector erase (hybrid sector option)	60
128-KB physical sector erase (hybrid sector option)	1000
512-KB logical sector erase	1000

### Typical current consumption, dual-quad SPI

Operation	Current (mA)
Serial read 50 MHz	32 (max)
Serial fast read 133 MHz	66 (max)
Quad read 104 MHz	122 (max)
Program	200 (max)
Erase	200 (max)
Standby	0.14 (typ)

## Table of contents

<b>Features</b> .....	<b>1</b>
<b>Performance summary</b> .....	<b>3</b>
<b>Table of contents</b> .....	<b>4</b>
<b>1 Overview</b> .....	<b>6</b>
1.1 General description .....	6
<b>2 SPI with multiple input / output (SPI-MIO) dual-quad</b> .....	<b>7</b>
<b>3 Signal descriptions</b> .....	<b>8</b>
3.1 Input/output summary .....	8
3.2 Multiple input / output (dual-quad SPI) .....	8
3.3 RESET#.....	9
3.4 Multiple Input / Output (Dual-Quad).....	9
3.5 Serial Clock (SCK).....	9
3.6 Chip Select (CS#).....	9
3.7 Input Output IO0–IO7 .....	9
3.8 Core Voltage Supply ( $V_{CC}$ ).....	9
3.9 Versatile I/O Power Supply ( $V_{IO}$ ).....	10
3.10 Supply and Signal Ground ( $V_{SS}$ ) .....	10
3.11 Not Connected (NC) .....	10
3.12 Reserved for Future Use (RFU) .....	10
3.13 Do Not Use (DNU).....	10
3.14 Block diagram .....	11
<b>4 Signal protocols</b> .....	<b>12</b>
4.1 SPI clock modes .....	12
4.2 Command protocol.....	13
4.3 Interface states .....	18
4.4 Configuration Register effects on the Interface .....	21
4.5 Data protection .....	22
<b>5 Electrical specifications</b> .....	<b>23</b>
5.1 Absolute maximum ratings .....	23
5.2 Thermal resistance .....	23
5.3 Operating ranges .....	24
5.4 Power-up and power-down.....	25
5.5 DC characteristics .....	27
<b>6 Timing specifications</b> .....	<b>28</b>
6.1 Key to switching waveforms.....	28
6.2 AC test conditions .....	29
6.3 Reset .....	30
6.4 SDR AC characteristics.....	32
6.5 DDR AC characteristics .....	34
<b>7 Physical interface</b> .....	<b>37</b>
7.1 Dual-Quad SOIC 16-lead package .....	37
7.2 SOIC 16 physical diagram .....	38
<b>8 Address space maps</b> .....	<b>39</b>
8.1 Overview.....	39
8.2 Flash memory array .....	40
8.3 ID-CFI address space.....	41
8.4 OTP address space.....	41
8.5 Registers .....	44
<b>9 Data protection</b> .....	<b>56</b>
9.1 Secure silicon region (OTP) .....	56
9.2 Write Enable command .....	57

9.3 Block protection .....	57
9.4 Advanced sector protection .....	59
<b>10 Commands .....</b>	<b>63</b>
10.1 Command set summary .....	64
10.2 Identification commands .....	70
10.3 Register Access commands .....	73
10.4 Read memory array commands .....	84
10.5 Program Flash Array commands .....	92
10.6 Erase Flash Array commands .....	97
10.7 One Time Program Array commands .....	103
10.8 Advanced Sector Protection commands .....	104
10.9 Reset commands .....	111
10.10 Embedded algorithm performance tables .....	112
<b>11 Data integrity .....</b>	<b>113</b>
11.1 Erase endurance .....	113
11.2 Data retention .....	113
<b>12 Software interface reference .....</b>	<b>114</b>
12.1 Command summary .....	114
12.2 Device ID and common flash interface (ID-CFI) address map .....	116
12.3 Initial delivery state .....	129
<b>13 Ordering information .....</b>	<b>130</b>
13.1 Ordering information S79FL256S/S79FL512S .....	130
13.2 Valid combinations .....	131
13.3 Valid combinations - automotive grade / AEC-Q100 .....	131
<b>Revision history .....</b>	<b>132</b>

## **1 Overview**

### **1.1 General description**

The S79FL256S/S79FL512S devices are flash non-volatile memory products using:

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

The S79FL256S/S79FL512S devices connect two quad I/O SPI devices with a single CS# resulting in an eight bit I/O data path. This byte I/O interface is called dual-quad I/O.

These devices connect to a host system via an SPI. Traditional SPI single bit serial input and output (IO1 and IO5) is supported as well as four-bit (Quad I/O or QIO) serial commands. This multiple width interface is called SPI multi-I/O or MIO. In addition, these two devices add support for DDR read commands for QIO that transfers address and read data on both edges of the clock.

The Eclipse architecture features a page programming buffer that allows up to 256 words (512 bytes) or 512 words (1024 bytes) to be programmed in one operation, resulting in significantly faster effective programming (up to 2 MB/s or 3 MB/s respectively) and erase (up to 1 MB/s) than prior generation SPI program or erase algorithms.

Executing code directly from flash memory is often called eXecute-in-Place (XIP). By using the S79FL-S devices at the higher clock rates supported, with QIO or DDR-QIO commands, the instruction read transfer rate can match or exceed traditional parallel interface, asynchronous, NOR flash memories while reducing signal count dramatically.

The S79FL-S products offer high density coupled with the fastest read and write performance required by a variety of embedded applications. They are ideal for code shadowing, XIP, and data storage.

SPI with multiple input / output (SPI-MIO)  
dual-quad

## **2 SPI with multiple input / output (SPI-MIO) dual-quad**

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 dual-quad SPI devices reduce the number of signals for connection to the host system by serially transferring all control, address, and data information over 10 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 dual-quad SPI devices use the industry standard single bit SPI using two quad SPI devices in each package (quad SPI-1 and quad SPI-2). This interface is called dual-quad and enables support of byte wide (8 bit) serial transfers. There is one package option available for S79FL256S/S79FL512S:

- 16-pin SOIC package

For documentation simplicity, all AC timings and waveforms and DC specification are defined using single CS# (Chip Select) and SCK (Serial Clock) signals. For S79FL256S/S79FL512S, the CS# signals and the SCK signals for quad SPI-1 and quad SPI-2 are internally tied together in the package.

### 3 Signal descriptions

#### 3.1 Input/output summary

**Table 1 Dual-quad input/output descriptions**

Signal name	Type	Description
RESET#	Input	<b>Hardware Reset.</b> LOW = device resets and returns to standby state, ready to receive a command. The signal has an internal pull-up resistor and may be left unconnected in the host system if not used.
SCK	Input	<b>Serial Clock</b>
CS#	Input	<b>Chip Select</b>
IO0	I/O	I/O 0 for Quad SPI-1
IO1	I/O	I/O 1 for Quad SPI-1
IO2	I/O	I/O 2 for Quad SPI-1
IO3	I/O	I/O 3 for Quad SPI-1
IO4	I/O	I/O 0 for Quad SPI-2
IO5	I/O	I/O 1 for Quad SPI-2
IO6	I/O	I/O 2 for Quad SPI-2
IO7	I/O	I/O 3 for Quad SPI-2
VCC	Supply	Core Power Supply
VSS	Supply	Ground
NC	Unused	<b>Not Connected.</b> 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 a NC pin must not have voltage levels higher than the VCC absolute maximum (Supply Voltage).
RFU	Reserved	<b>Reserved for Future Use.</b> No device internal signal is currently connected to the package connector but there is potential future use for 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	<b>Do Not Use.</b> A device internal signal may be connected to the package connector. The connection may be used by Infineon 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 VIL. The signal has an internal pull-down resistor and may be left unconnected in the host system or may be tied to VSS. Do not use these connections for PCB signal routing channels. Do not connect any host system signal to this connection.

#### 3.2 Multiple input / output (dual-quad SPI)

Quad Input / Output (I/O) commands send instructions to the memory only on the IO0 (quad SPI-1) and IO4 (quad SPI-2) signals. Address is sent from the host to the memory as four bit (nibble) on IO0, IO1, IO2, IO3 (quad SPI-1) and repeated on IO4, IO5, IO6, IO7 (quad SPI-2). Data is sent and returned to the host as bytes on IO0–IO7.



### 3.3 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 zero,
- loads the Program Buffer with all ones,
- 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  $t_{PJ}$  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  $t_{PJ}$ . 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 may 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.

### 3.4 Multiple Input / Output (Dual-Quad)

Quad Input / Output (I/O) commands send instructions to the memory only on the IO0 (quad SPI-1) and IO4 (quad SPI-2) signals. Address is sent from the host to the memory as four bit (nibble) on IO0, IO1, IO2, IO3 (quad SPI-1) and repeated on IO4, IO5, IO6, IO7 (quad SPI-2). Data is sent and returned to the host as bytes on IO0–IO7.

### 3.5 Serial Clock (SCK)

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

### 3.6 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.

### 3.7 Input Output IO0–IO7

These signals are input and outputs for receiving instructions, addresses, and data to be programmed (values latched on rising edge of serial SCK clock signal) as well as shifting out data (on the falling edge of SCK, in SDR commands, and on every edge of SCK, in DDR commands).

### 3.8 Core 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 2.7 V to 3.6 V.

### **3.9 Versatile I/O Power Supply ( $V_{IO}$ )**

$V_{IO}$  functionality is not supported on the standard configuration of the S79FL256S/S79FL512S devices.

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

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

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

### **3.12 Reserved for Future Use (RFU)**

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

### **3.13 Do Not Use (DNU)**

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

### 3.14 Block diagram

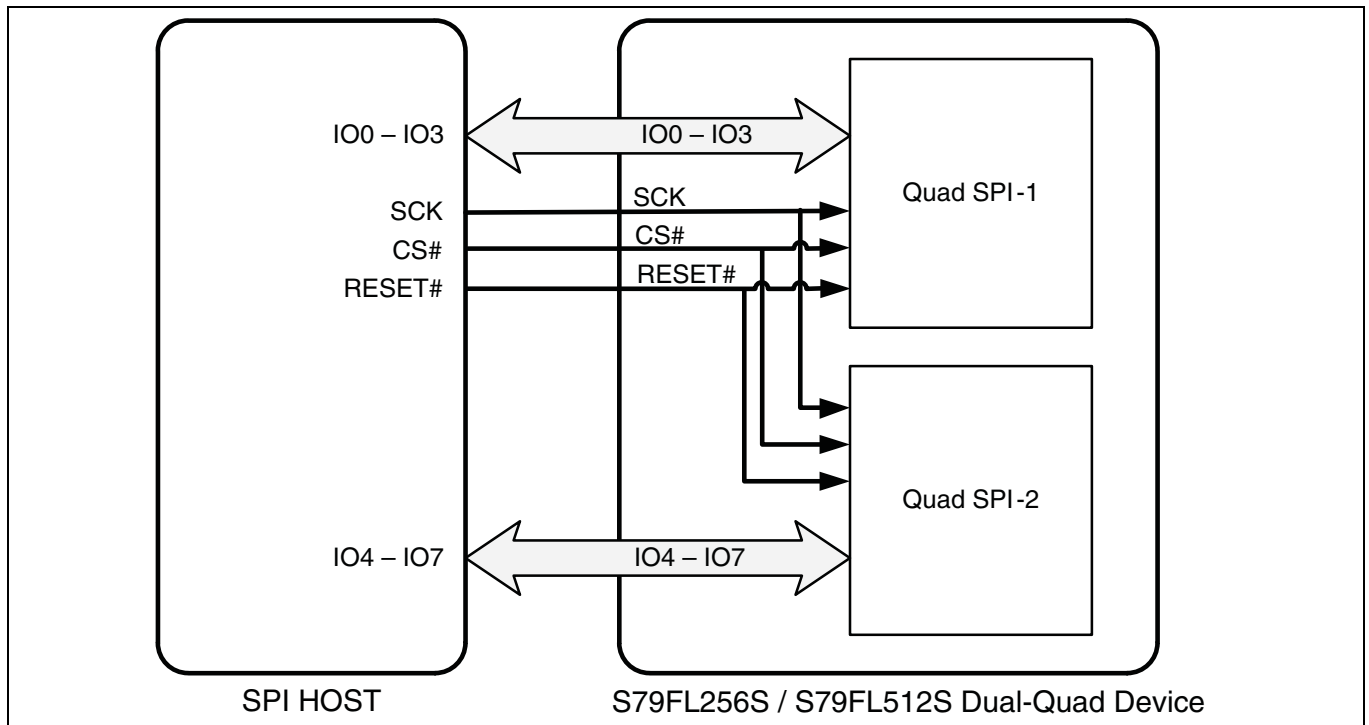


Figure 1 SPI host and S79FL256S / S79FL512S dual-quad SPI devices in the 16-pin SOIC package<sup>[1]</sup>

**Note**

1. The Chip Select (CS#) and Clock (SCK) signals for quad SPI-1 and quad SPI-2 are internally tied together in the 16-pin SOIC package.

## 4 Signal protocols

### 4.1 SPI clock modes

#### 4.1.1 Single data rate (SDR)

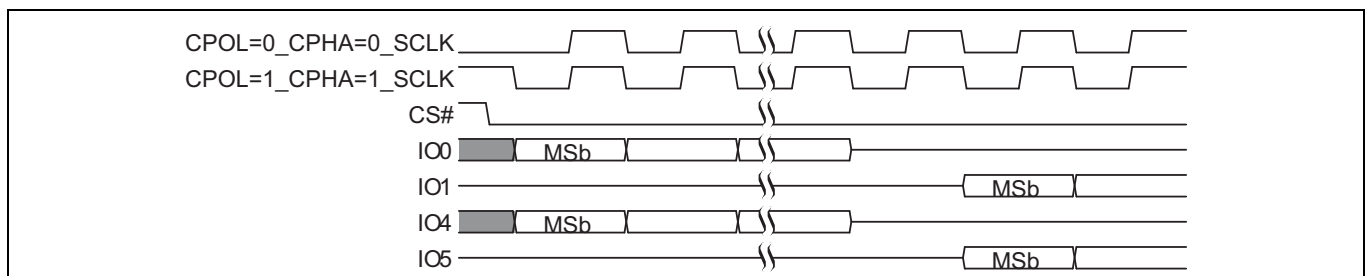
The S25FL-S devices can be driven by an embedded microcontroller (bus master) in either of the two following clocking modes.

- **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 2 Dual-Quad SPI SDR modes supported**

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

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

### 4.1.2 Double data rate (DDR)

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

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

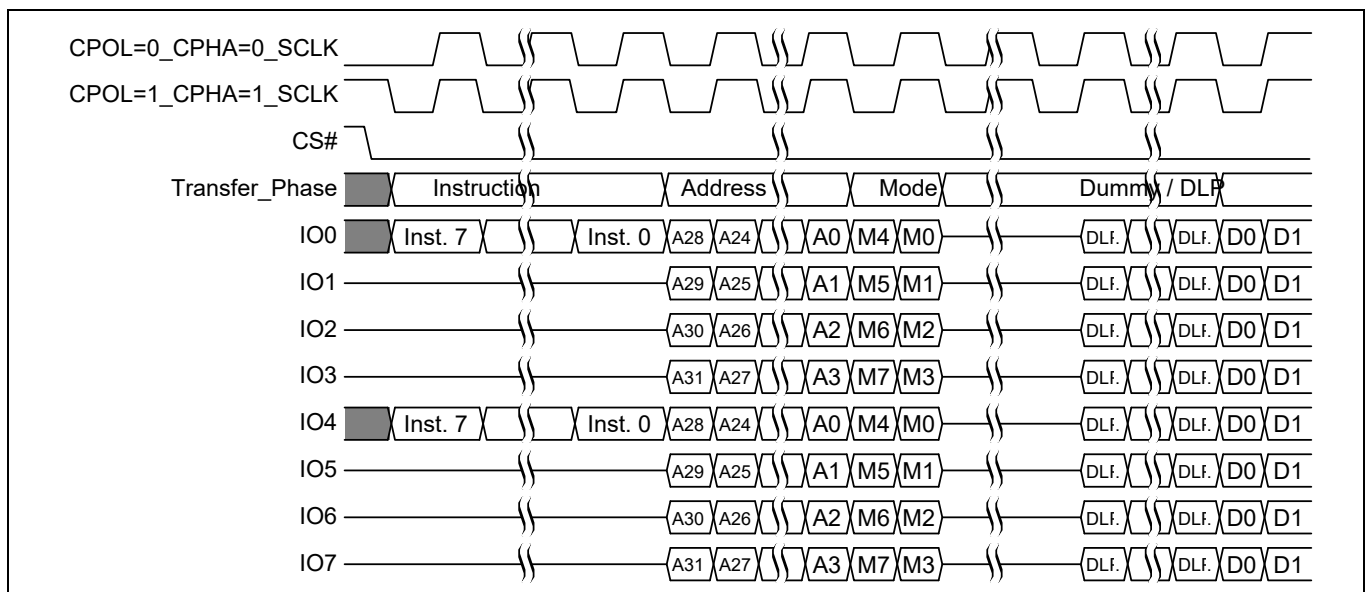


Figure 3 Dual-Quad SPI DDR modes supported

## 4.2 Command protocol

All communication between the host system and S25FL-S 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.

Quad Input / Output (I/O) commands provide an address sent from the host as four bit (nibble) groups on IO0, IO1, IO2, IO3 and repeated on IO4, IO5, IO6, IO7, then followed by dummy cycles. Data is returned to the host as byte on IO0–IO7. This is referenced as 2-8-8 for Quad I/O command protocols.

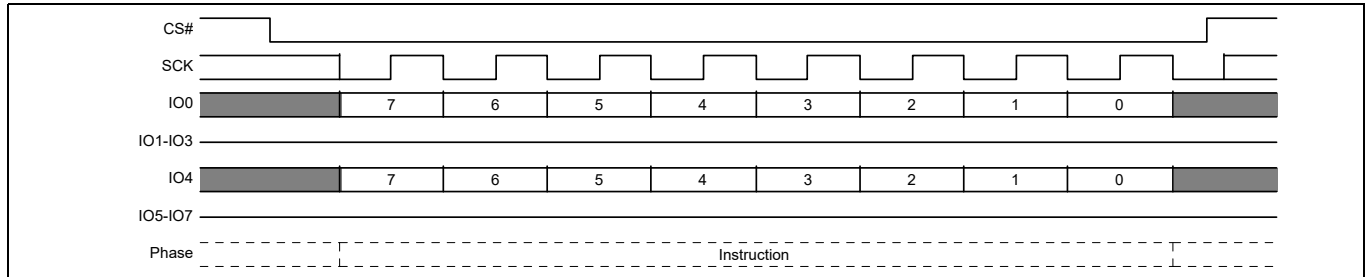
Commands are structured as follows:

- 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 8-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.

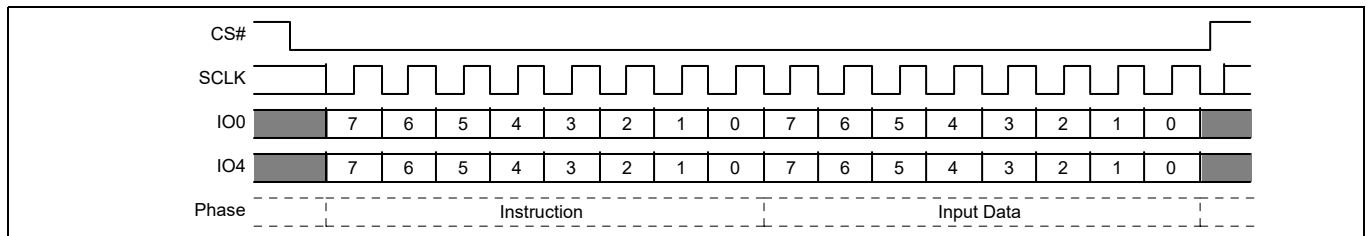
## Signal protocols

- The instruction may be standalone or may be followed by address bits to select a location within one of several address spaces in the device. The instruction determines the address space used. The address may be either a 24-bit or a 32-bit byte boundary, address. The address transfers occur on SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- Quad I/O read instructions send an instruction modifier called continuous read mode bits, following the address, to indicate whether the next command will be of the same type with an implied, rather than an explicit, instruction. These mode bits initiate or end the continuous read mode. In continuous read mode, the next command thus does not provide an instruction byte, only a new address and mode bits. This reduces the time needed to send each command when the same command type is repeated in a sequence of commands. The mode bit transfers occur on SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- The 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 4-bit groups per (quad) transfer on the IO0–IO3 signals. Within the 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, in SDR commands, or on every SCK edge, in DDR commands.
- The address or mode bits may be followed by write data to be stored in the memory device or by a read latency period before read data is returned to the host.
- Write data bit transfers occur on SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- SCK continues to toggle during any read access latency period. The latency may be zero to several SCK cycles (also referred to as dummy cycles). At the end of the read latency cycles, the first read data bits are driven from the outputs on SCK falling edge at the end of the last read latency cycle. The first read data bits are considered transferred to the host on the following SCK rising edge. Each following transfer occurs on the next SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- If the command returns read data to the host, the device continues sending data transfers until the host takes the CS# signal HIGH. The CS# signal can be driven HIGH after any transfer in the read data sequence. This will terminate the command.
- At the end of a command that does not return data, the host drives the CS# input HIGH. The CS# signal must go HIGH after the eighth bit, of a standalone 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.

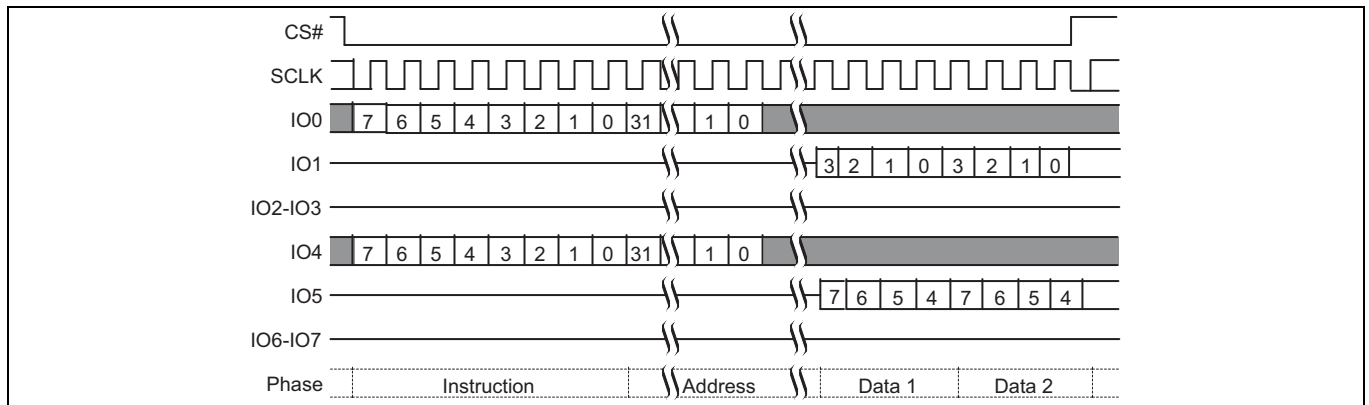
### 4.2.1 Command sequence examples



**Figure 4 Dual-Quad Standalone Instruction command<sup>[2]</sup>**



**Figure 5 Dual-Quad Single Bit Wide Input command<sup>[2]</sup>**



**Figure 6 Dual-Quad Single Bit Wide I/O command without latency<sup>[2]</sup>**

**Note**

- 2. Instruction needs to be the same for both IO0 (Quad SPI-1) and IO4 (Quad SPI-2).

Signal protocols

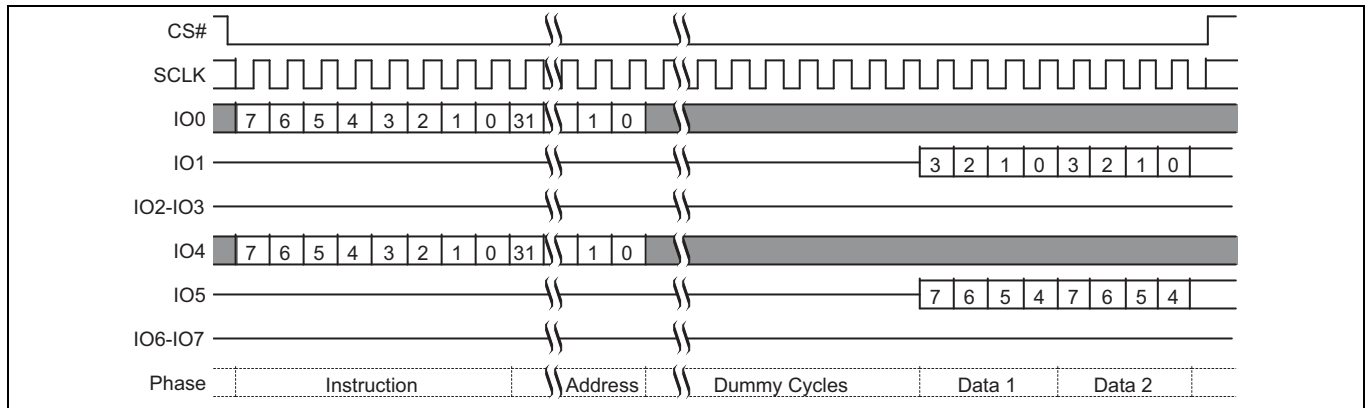


Figure 7 Dual-Quad Single Bit Wide I/O command with latency<sup>[3]</sup>

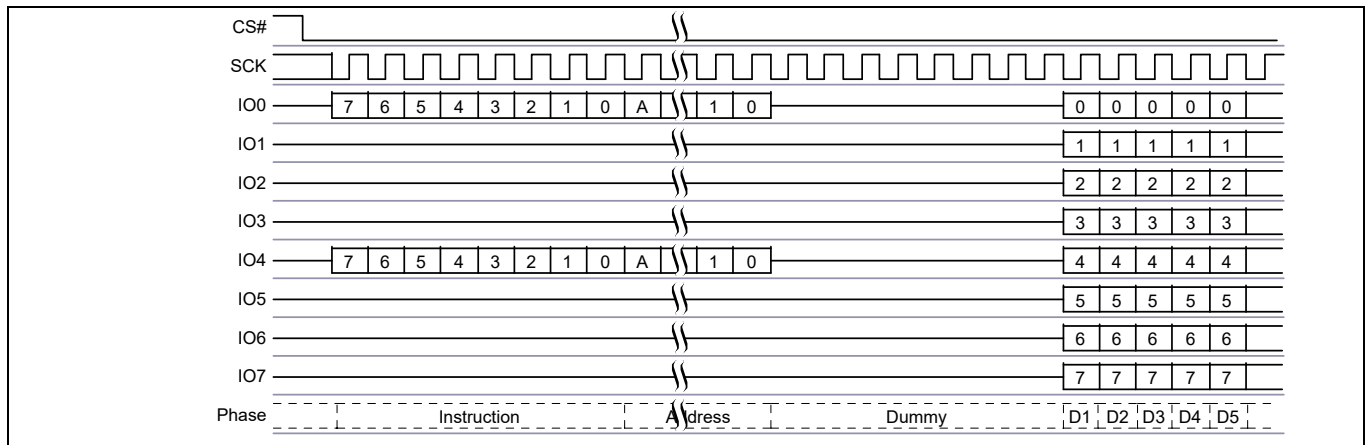


Figure 8 Dual-Quad, Quad Output Read command<sup>[4]</sup>

Notes

3. Instruction needs to be the same for both IO0 (quad SPI-1) and IO4 (quad SPI-2).
4. A = MSb of address = 23 for 3-byte address, or 31 for 4-byte address.



Signal protocols

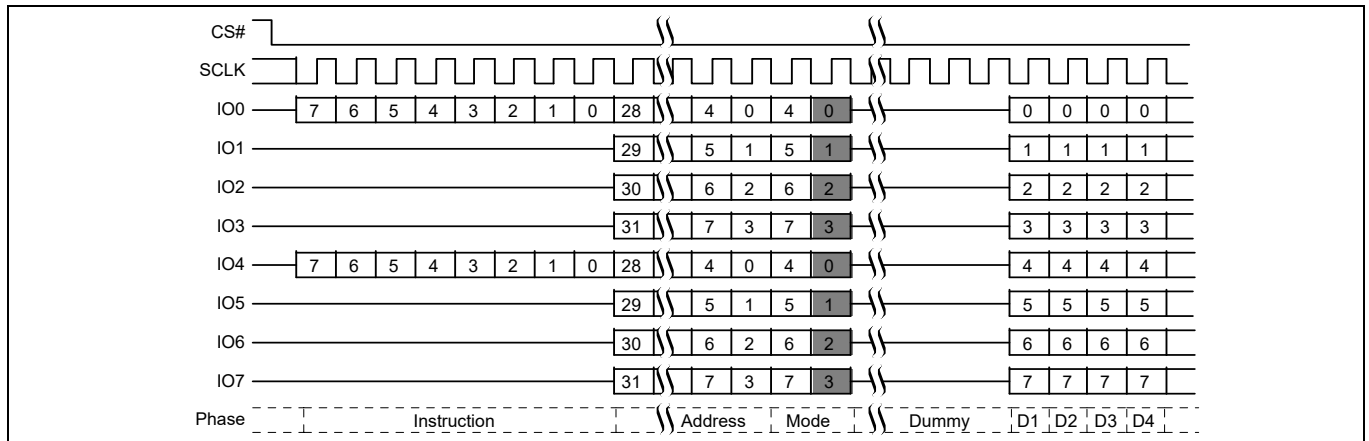


Figure 9 Dual-Quad, Quad I/O command<sup>[5, 6]</sup>

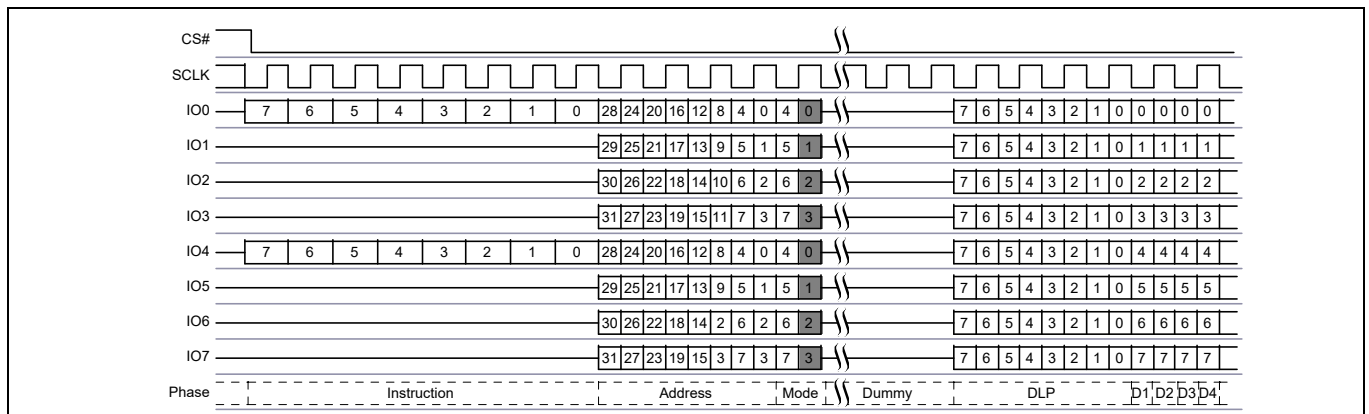


Figure 10 Dual-Quad DDR Quad I/O Read command<sup>[5, 6]</sup>

Additional sequence diagrams, specific to each command, are provided in “Commands” on page 63.

Notes

5. Instruction, Address and Mode bits needs to be the same for both IO0-IO3 (quad SPI-1) and IO4-IO7 (quad SPI-2).
6. The gray bits are optional, the host does not have to drive bits during that cycle.

### 4.3 Interface states

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

**Table 2 Dual-quad interface states summary**

Interface state	VDD	SCK	CS#	RESET#	I00-I07
Power-Off	< VCC (low)	X	X	X	X
Low Power					
Hardware Data Protection	< VCC (cut-off)	X	X	X	X
Power-On (Cold) Reset	≥ VCC (min)	X	HH	X	X
Hardware (Warm) Reset Non-Quad Mode	≥ VCC (min)	X	X	HL	X
Hardware (Warm) Reset Quad Mode	≥ VCC (min)	X	HH	HL	X
Interface Standby	≥ VCC (min)	X	HH	HH	X
Instruction Cycle (Legacy SPI)	≥ VCC (min)	HT	HL	HH	X
Single Input Cycle Host to Memory Transfer	≥ VCC (min)	HT	HL	HH	X
Single Latency (Dummy) Cycle	≥ VCC (min)	HT	HL	HH	X
Single Output Cycle Memory to Host Transfer	≥ VCC (min)	HT	HL	HH	X
Quad Input Cycle Host to Memory Transfer	≥ VCC (min)	HT	HL	HH	X
Quad Latency (Dummy) Cycle	≥ VCC (min)	HT	HL	HH	X
Quad Output Cycle Memory to Host Transfer	≥ VCC (min)	HT	HL	HH	X
DDR Quad Input Cycle Host to Memory Transfer	≥ VCC (min)	HT	HL	HH	X
DDR Latency (Dummy) Cycle	≥ VCC (min)	HT	HL	HH	X
DDR Quad Output Cycle Memory to Host Transfer	≥ VCC (min)	HT	HL	HH	X

**Legend:**

Z = No driver - floating signal

HL = Host driving  $V_{IL}$

HH = Host driving  $V_{IH}$

HV = Either HL or HH

X = HL or HH or Z

HT = Toggling between HL and HH

ML = Memory driving  $V_{IL}$

MH = Memory driving  $V_{IH}$

MV = Either ML or MH

#### 4.3.1 Power-off

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

#### 4.3.2 Low power hardware data protection

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

#### 4.3.3 Power-on (cold) reset

When the core voltage supply remains at or below the  $V_{CC}$  (low) voltage for  $\geq t_{PD}$  time, then rises to  $\geq V_{CC(\text{Minimum})}$  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 30.

#### 4.3.4 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 “[POR followed by hardware reset](#)” on page 30.

#### 4.3.5 Interface Standby

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

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

#### 4.3.6 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.

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, quad output, quad I/O, or DDR quad I/O. The expected next interface state depends on the instruction received.

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

#### 4.3.7 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 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 (IO1 and IO5) signals.

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, or quad output.

#### **4.3.8 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 SCK toggles. The host may drive the IO0 and IO4 signals during these cycles or the host may leave IO0 and IO4 floating. The memory does not use any data driven on IO0 and IO4 or other I/O signals during the latency cycles. In 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 (IO0 and IO4) 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, or quad width.

#### **4.3.9 Dual-Quad single output cycle — Memory to Host transfer**

Several commands transfer information back to the host on the serial outputs (IO1 and IO5) signals. The host keeps RESET# HIGH, CS# LOW. The memory ignores the serial input (IO0 and IO4) signals. The memory drives IO1 and IO5 with data.

The next interface state continues to be dual output cycle until the host returns CS# to HIGH ending the command.

#### **4.3.10 QPP or QOR address input cycle**

The Quad Page Program and Quad Output Read commands send address to the memory only on IO0 and IO4. 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.

#### **4.3.11 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.

#### **4.3.12 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.

#### **4.3.13 Quad output cycle – Memory to Host transfer**

The Quad Output Read and Quad I/O Read return data to the host eight 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.

#### **4.3.14 DDR Quad input cycle – Host to Memory transfer**

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

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

#### **4.3.15 DDR latency cycle**

DDR Read commands may have one to several latency cycles during which read data is read from the main flash memory array before transfer to the host. The number of latency cycles are determined by the latency code in the Configuration Register (CR1[7:6]). During the latency cycles, the host keeps RESET# HIGH and CS# LOW. The host may not drive the IO signals during these cycles. So that there is sufficient time for the host drivers to turn off before the memory begins to drive. This prevents driver conflict between host and memory when the signal direction changes. The memory has an option to drive all the IO signals with a data learning pattern (DLP) during the last four latency cycles. The DLP option should not be enabled when there are fewer than five latency cycles so that there is at least one cycle of high impedance for turn around of the IO signals before the memory begins driving the DLP. When there are more than four cycles of latency the memory does not drive the IO signals until the last four cycles of latency.

The next interface state following the last latency cycle is a DDR quad output cycle, depending on the instruction.

#### **4.3.16 DDR Quad output cycle – Memory to Host transfer**

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

The next interface state continues to be DDR quad output cycle until the host returns CS# to HIGH ending the command.

### **4.4 Configuration Register effects on the Interface**

The Configuration Register 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 and allow Quad Page Program, Quad Output Read, and Quad I/O Read commands. Quad mode must also be selected to allow Read DDR Quad I/O commands. This quad bit is set to '1' by default for dual-quad SPI.

## **4.5 Data protection**

Some basic protection against unintended changes to stored data are provided and controlled purely by the hardware design. These are described below. Other software managed protection methods are discussed in the [“Software interface”](#) on page 41 of this document.

### **4.5.1 Power-up**

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

### **4.5.2 Low power**

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

### **4.5.3 Clock pulse count**

The device verifies that all 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.

## 5 Electrical specifications

### 5.1 Absolute maximum ratings

**Table 3 Absolute maximum ratings**

Storage temperature plastic packages	-65°C to +150°C
Ambient temperature with power applied	-65°C to +125°C
V <sub>CC</sub>	-0.5 V to +4.0 V
Input voltage with respect to ground (V <sub>SS</sub> ) <sup>[7]</sup>	-0.5 V to +(V <sub>CC</sub> + 0.5 V)
Output short circuit current <sup>[8]</sup>	100 mA

### 5.2 Thermal resistance

**Table 4 Thermal resistance**

Parameter	Description	Device	SS3016	Unit
Theta JA	Thermal resistance (junction to ambient)	S79FL256S	59	°C/W
		S79FL512S	65	

#### Notes

7. See “**Input signal overshoot**” on page 24 for allowed maximums during signal transition.
8. 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.
9. 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 datasheet is not implied. Exposure of the device to absolute maximum rating conditions for extended periods may affect device reliability.

### 5.3 Operating ranges

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

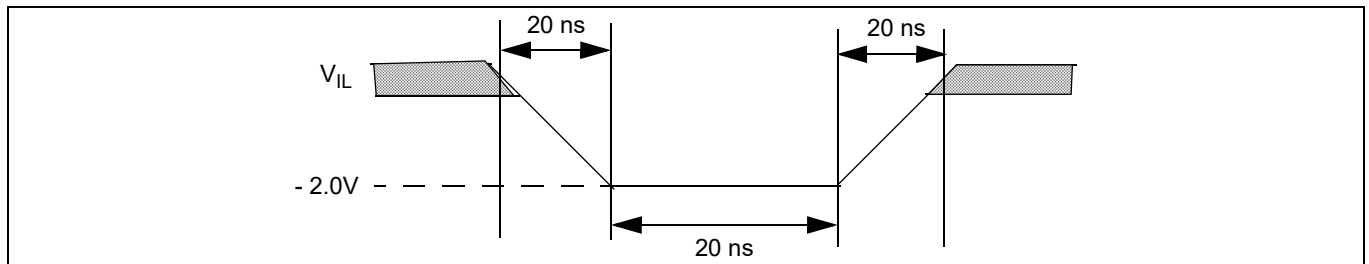
#### 5.3.1 Temperature ranges

**Table 5 Recommended operating ranges**

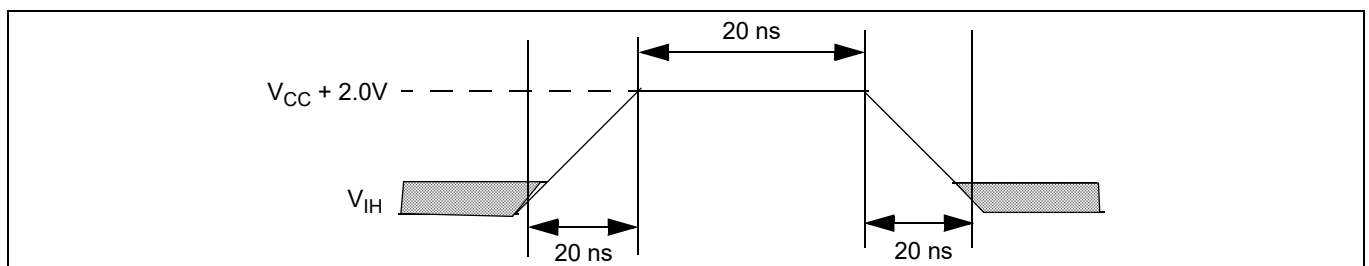
Parameter	Symbol	Conditions	Spec		Unit
			Min	Max	
Ambient temperature	$T_A$	Industrial (I) devices	-40	+85	°C
		Automotive (A) - in cabin	-40	+105	

#### 5.3.2 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  $-2.0$  V or overshoot to  $V_{CC} + 2.0$  V, for periods up to 20 ns.



**Figure 11 Maximum negative overshoot waveform**



**Figure 12 Maximum positive overshoot waveform**

**Note**

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



## 5.4 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 k $\Omega$ ) 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 13](#)). 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  $t_{PU}$ .

After power-up ( $t_{PU}$ ), the device is in Standby mode (not Deep Power Down 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 14](#)). 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 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  $\mu$ F).

**Table 6 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 Off where re-initialization is needed)	2.4	–	V
$V_{CC}$ (low)	$V_{CC}$ (low voltage for initialization to occur) $V_{CC}$ (low voltage for initialization to occur at embedded)	1.0 2.3	–	V
$t_{PU}$	$V_{CC}$ (min) to Read operation	–	300	$\mu$ s
$t_{PD}$	$V_{CC}$ (low) time	1.0	–	$\mu$ s

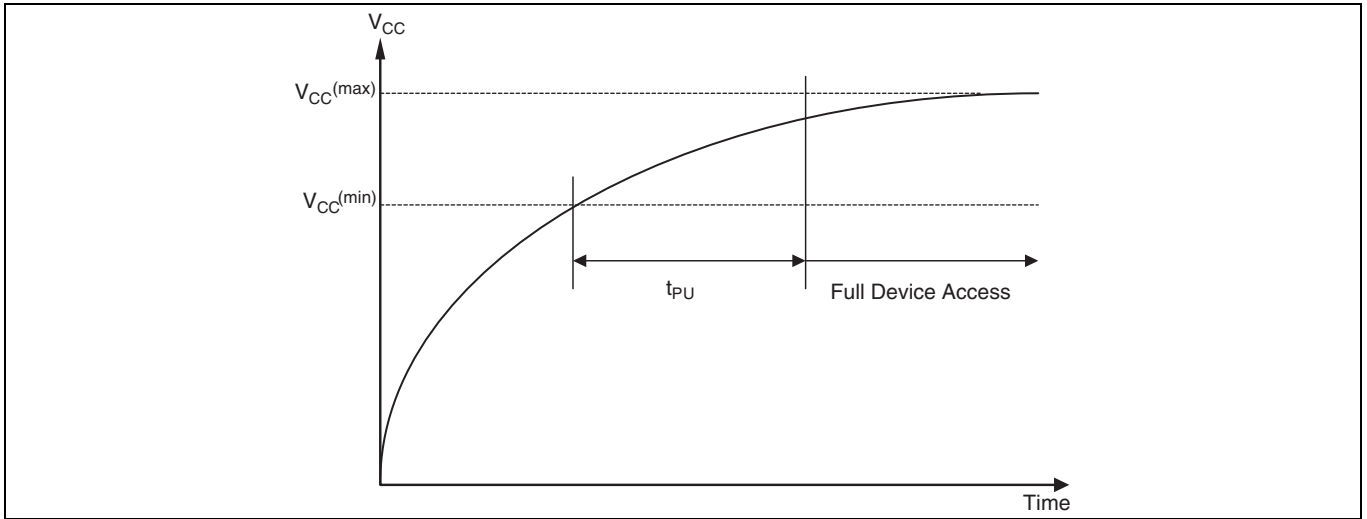


Figure 13 Power-up

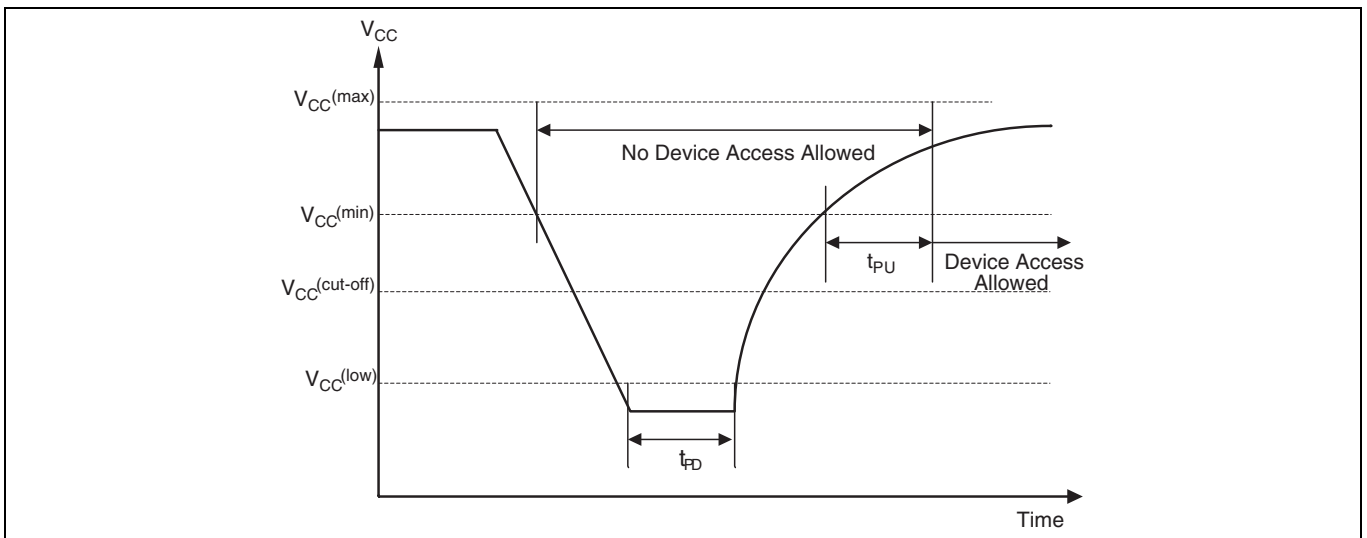


Figure 14 Power-down and voltage drop

## 5.5 DC characteristics

Applicable within operating  $-40^{\circ}\text{C}$  to  $+85^{\circ}\text{C}$  range.

**Table 7 DC characteristics**

Symbol	Parameter	Test conditions	Min	Typ <sup>[11]</sup>	Max	Unit
$V_{IL}$	Input low voltage	–	–0.5	–	$0.2 \times V_{CC}$	V
$V_{IH}$	Input high voltage	–	$0.7 \times V_{CC}$	–	$V_{CC} + 0.4$	V
$V_{OL}$	Output low voltage	$I_{OL} = 1.6 \text{ mA}$ , $V_{CC} = V_{CC \text{ min}}$	–	–	$0.15 \times V_{CC}$	V
$V_{OH}$	Output high voltage	$I_{OH} = -0.1 \text{ mA}$	$0.85 \times V_{CC}$	–	–	V
$I_{LI}$	Input leakage current	$V_{CC} = V_{CC \text{ Max}}$ , $V_{IN} = V_{IH} \text{ or } V_{IL}$	–	–	$\pm 4$	$\mu\text{A}$
$I_{LO}$	Output leakage current	$V_{CC} = V_{CC \text{ Max}}$ , $V_{IN} = V_{IH} \text{ or } V_{IL}$	–	–	$\pm 4$	$\mu\text{A}$
$I_{CC1}$	Active power supply current (READ)	Serial SDR @ 50 MHz Serial SDR @ 133 MHz Quad SDR @ 80 MHz Quad SDR @ 104 MHz Quad DDR @ 80 MHz Outputs unconnected during read data return <sup>[12]</sup>	–	–	32 66/70 <sup>[13]</sup> 100 122 180	mA
$I_{CC2}$	Active power supply current (Page Program)	$CS\# = V_{CC}$	–	–	200	mA
$I_{CC3}$	Active power supply current (WRR)	$CS\# = V_{CC}$	–	–	200	mA
$I_{CC4}$	Active power supply current (SE)	$CS\# = V_{CC}$	–	–	200	mA
$I_{CC5}$	Active power supply current (BE)	$CS\# = V_{CC}$	–	–	200	mA
$I_{SB}$ (Industrial)	Standby current	RESET#, $CS\# = V_{CC}$ ; SI, SCK = $V_{CC}$ or $V_{SS}$ , Industrial temperature	–	140	200	$\mu\text{A}$
$I_{SB}$ (Automotive)	Standby current	RESET#, $CS\# = V_{CC}$ ; SI, SCK = $V_{CC}$ or $V_{SS}$ , Automotive temperature	–	140	600	$\mu\text{A}$

### 5.5.1 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  $I_{SB}$ .

#### Notes

11. Typical values are at  $T_{AI} = 25^{\circ}\text{C}$  and  $V_{CC} = 3 \text{ V}$ .
12. Outputs switching current is not included.
13. Industrial temperature range / automotive in-cabin temperature range.

## 6 Timing specifications

### 6.1 Key to switching waveforms

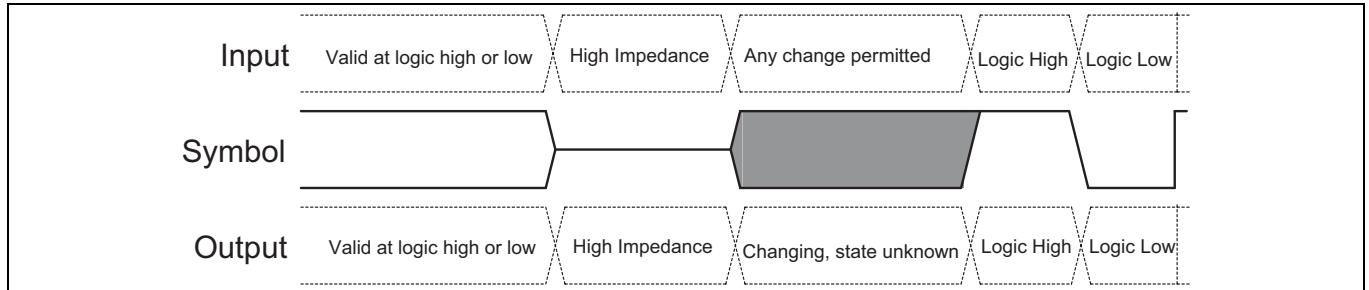


Figure 15 Waveform element meanings

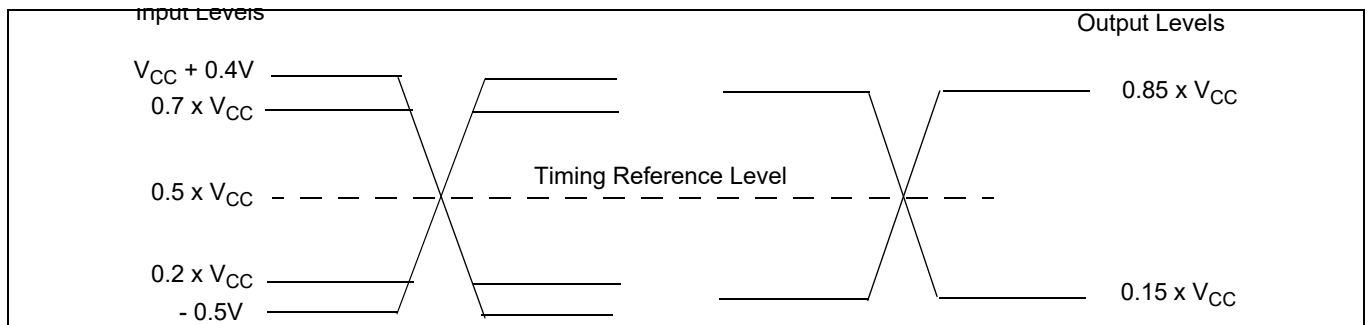


Figure 16 Input, output, and timing reference levels

## 6.2 AC test conditions

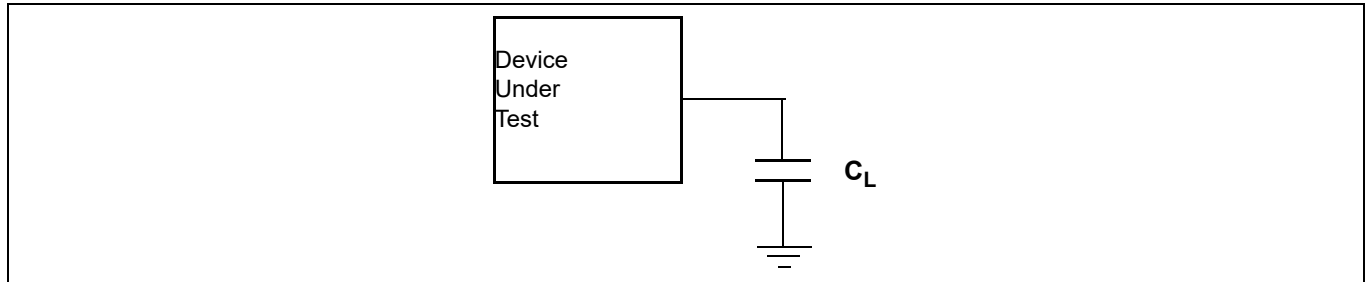


Figure 17 Test setup

Table 8 AC measurement conditions

Symbol	Parameter	Min	Max	Unit
$C_L$	Load capacitance		30 15 <sup>[17]</sup>	pF
	Input rise and fall times	-	2.4	ns
	Input pulse voltage	0.2 × $V_{CC}$ to 0.8 × $V_{CC}$		V
	Input timing reference voltage	0.5 × $V_{CC}$		V
	Output timing reference voltage	0.5 × $V_{CC}$		V

### 6.2.1 Capacitance characteristics

Table 9 Capacitance

Symbol	Parameter	Test conditions	Min	Max	Unit
$C_{IN}$	Input capacitance (applies to SCK, CS#, RESET#)	1 MHz	-	14	pF
$C_{OUT}$	Output capacitance (applies to all I/O)	1 MHz	-	20	pF

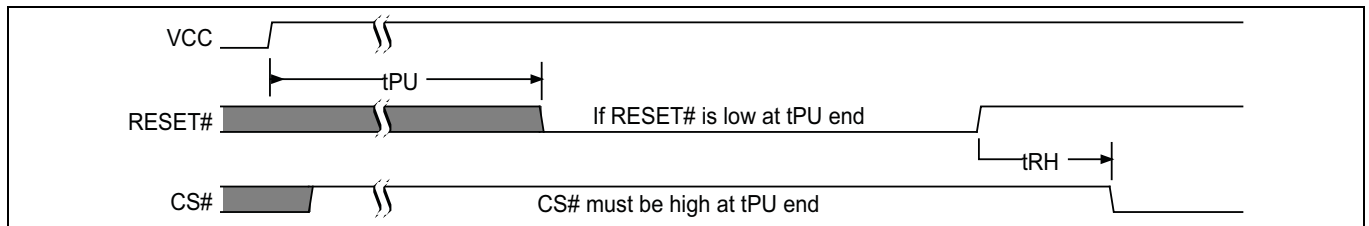
#### Notes

14. Output High-Z is defined as the point where data is no longer driven.
15. Input slew rate: 1.5 V/ns.
16. AC characteristics tables assume clock and data signals have the same slew rate (slope).
17. DDR operation.
18. For more information on capacitance, please consult the IBIS models.

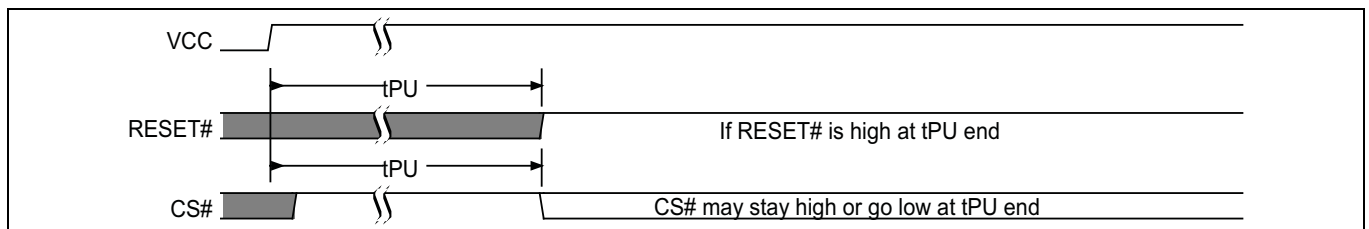
## 6.3 Reset

### 6.3.1 Power-on (cold) reset

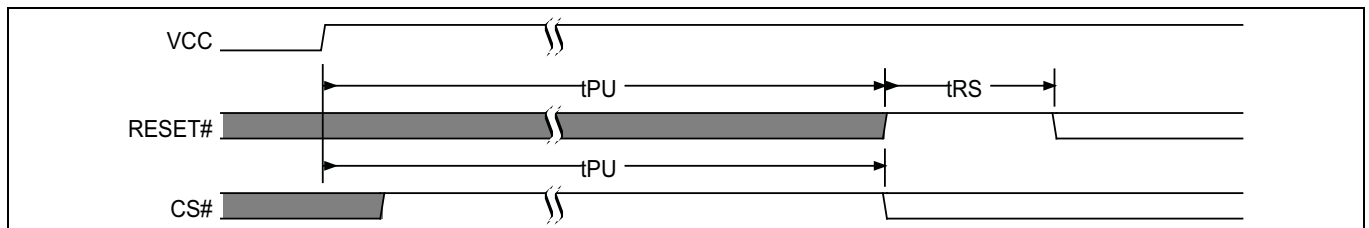
The device executes a POR process until a time delay of  $t_{PU}$  has elapsed after the moment that  $V_{CC}$  rises above the minimum  $V_{CC}$  threshold. See [Figure 13](#), [Table 6](#), and [Table 10](#). The device must not be selected (CS# to go HIGH with  $V_{CC}$ ) during power-up ( $t_{PU}$ ), i.e. no commands may be sent to the device until the end of  $t_{PU}$ . RESET# 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 18** Reset LOW at the end of POR



**Figure 19** Reset HIGH at the end of POR



**Figure 20** POR followed by hardware reset

### 6.3.2 Hardware (warm) reset

When the RESET# input transitions from  $V_{IH}$  to  $V_{IL}$ , the device will reset register states in the same manner as power-on reset but, does not go through the full reset process that is performed during POR. The hardware reset process requires a period of  $t_{RPH}$  to complete. If the POR process did not complete correctly for any reason during power-up ( $t_{PU}$ ), RESET# going LOW will initiate the full POR process instead of the hardware reset process and will require  $t_{PU}$  to complete the POR process.

The 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, tri-states all outputs, 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  $t_{RH}$ .

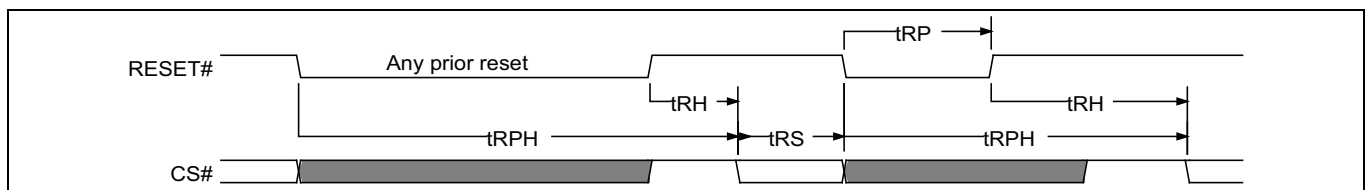


Figure 21 Hardware reset

Table 10 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	$\mu$ s
$t_{RP}$	RESET# pulse width	Min	200	ns
$t_{RH}$	Reset hold - RESET# HIGH before CS# LOW	Min	50	ns

#### Notes

19. RESET# LOW is optional and 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.
20. Sum of  $t_{RP}$  and  $t_{RH}$  must be equal to or greater than  $t_{RPH}$ .

## 6.4 SDR AC characteristics

**Table 11 AC characteristics ( $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ )**

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 <a href="#">Table 41</a> <sup>[24]</sup>	DC	-	133	MHz
$F_{SCK,C}$	SCK clock frequency for the following Dual and Quad commands: QOR, 4QOR, QIOR, 4QIOR	DC	-	104	MHz
$F_{SCK,QPP}$	SCK clock frequency for the QPP, 4QPP commands	DC	-	80	MHz
$P_{SCK}$	SCK clock period	$1/F_{SCK}$	-	$\infty$	
$t_{WH}, t_{CH}$	Clock HIGH time <sup>[25]</sup>	$45\% P_{SCK}$	-	-	ns
$t_{WL}, t_{CL}$	Clock LOW time <sup>[25]</sup>	$45\% P_{SCK}$	-	-	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 (program/erase)	10 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	-	3000 <sup>[26]</sup>	ns
$t_{HD}$	Data in hold time	2	-	-	ns
$t_V$	Clock LOW to output valid	-	-	8.0 <sup>[22]</sup> 7.65 <sup>[23]</sup> 6.5 <sup>[24]</sup>	ns
$t_{HO}$	Output hold time	2	-	-	ns
$t_{DIS}$	Output disable time	0	-	8	ns
$t_{WPS}$	WP# setup time	20 <sup>[21]</sup>	-	-	ns
$t_{WPH}$	WP# hold time	100 <sup>[21]</sup>	-	-	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# disable to output valid	-	-	8	ns

### Notes

21. Only applicable as a constraint for WRR instruction when SRWD is set to '1'.
22. Full  $V_{CC}$  range (2.7 V–3.6 V) and  $CL = 30\text{ pF}$ .
23. Regulated  $V_{CC}$  range (3.0 V–3.6 V) and  $CL = 30\text{ pF}$ .
24. Regulated  $V_{CC}$  range (3.0 V–3.6 V) and  $CL = 15\text{ pF}$ .
25.  $\pm 10\%$  duty cycle is supported for frequencies  $\leq 50\text{ MHz}$ .
26. Maximum value only applies during Program/Erase Suspend/Resume commands.



### 6.4.1 Clock timing

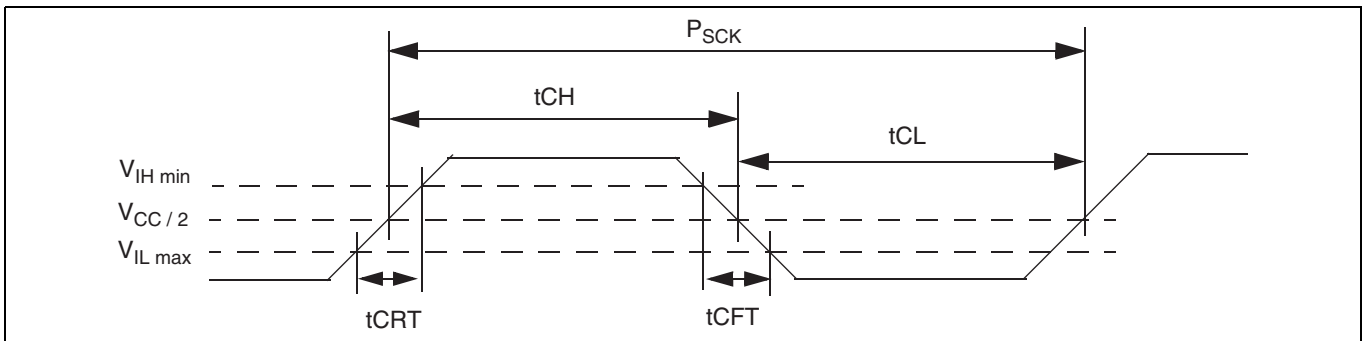


Figure 22 Clock timing

### 6.4.2 Input / output timing

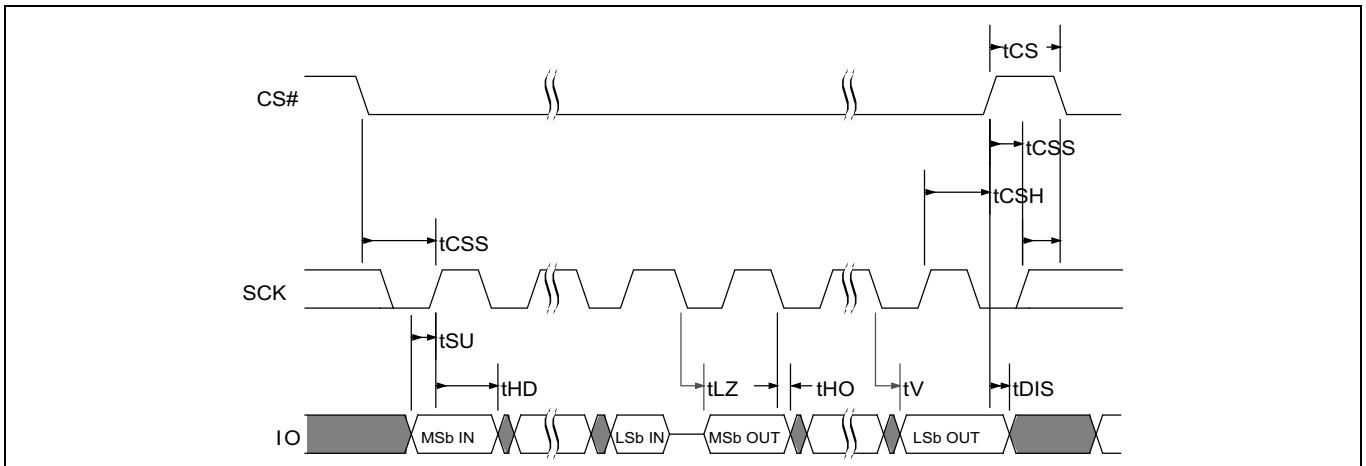


Figure 23 SPI SDR dual-quad timing

## 6.5 DDR AC characteristics

**Table 12 AC characteristics DDR operation**

Symbol	Parameter	80 MHz			Unit
		Min	Typ	Max	
$f_{SCK,R}$	SCK clock frequency for DDR READ instruction	DC	-	80	MHz
$P_{SCK,R}$	SCK clock period for DDR READ instruction	12.5	-	$\infty$	ns
$t_{WH}, t_{CH}$	Clock HIGH time	45% $P_{SCK}$	-	-	ns
$t_{WL}, t_{CL}$	Clock LOW time	45% $P_{SCK}$	-	-	ns
$t_{CS}$	CS# HIGH time (Read instructions)	10	-	-	ns
$t_{CSS}$	CS# active setup time (relative to SCK)	3	-	-	ns
$t_{CSH}$	CS# active hold time (relative to SCK)	3	-	-	ns
$t_{SU}$	IO in setup time	1.5	-	3000 <sup>[27]</sup>	ns
$t_{HD}$	IO in hold time	1.5	-	-	ns
$t_V$	Clock LOW to output valid	1.5	-	6.5 <sup>[28]</sup>	ns
$t_{HO}$	Output hold time	1.5	-	-	ns
$t_{DIS}$	Output disable time	-	-	8	ns
$t_{LZ}$	Clock to output LOW impedance	0	-	8	ns
$t_{O\_SKEW}$	First output to last output data valid time	-	-	600	ps

### Notes

27. Maximum value only applies during Program/Erase Suspend/Resume commands.

28. Regulated  $V_{CC}$  range (3.0 V–3.6 V) and  $C_L = 15$  pF.

### 6.5.1 DDR input timing

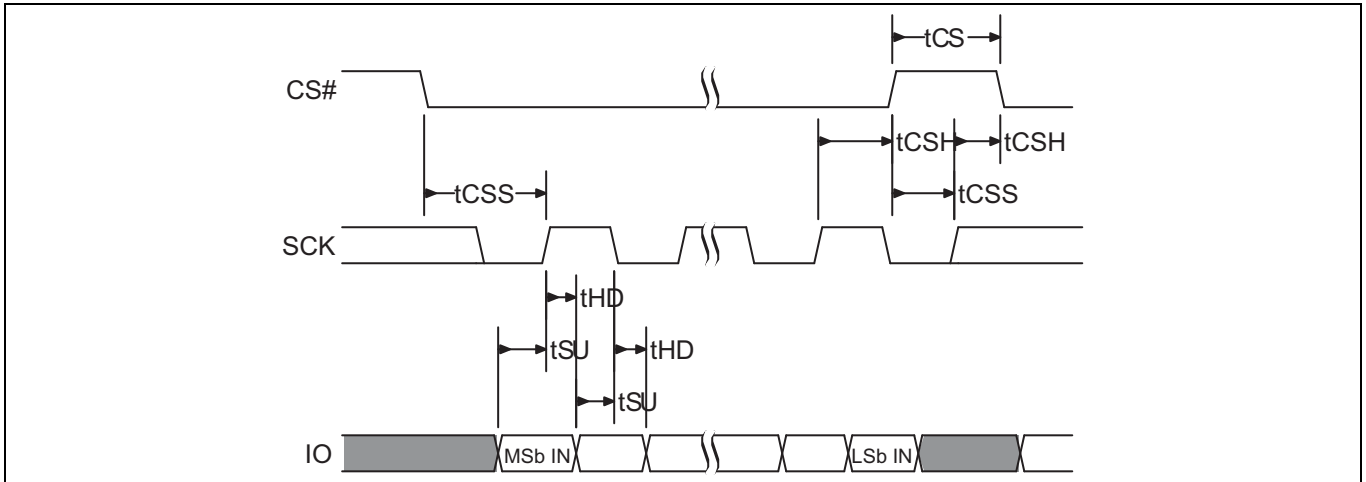


Figure 24 SPI DDR input timing

### 6.5.2 DDR output timing

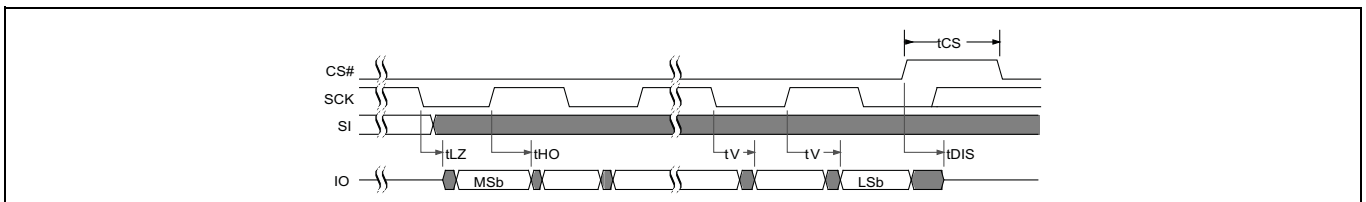
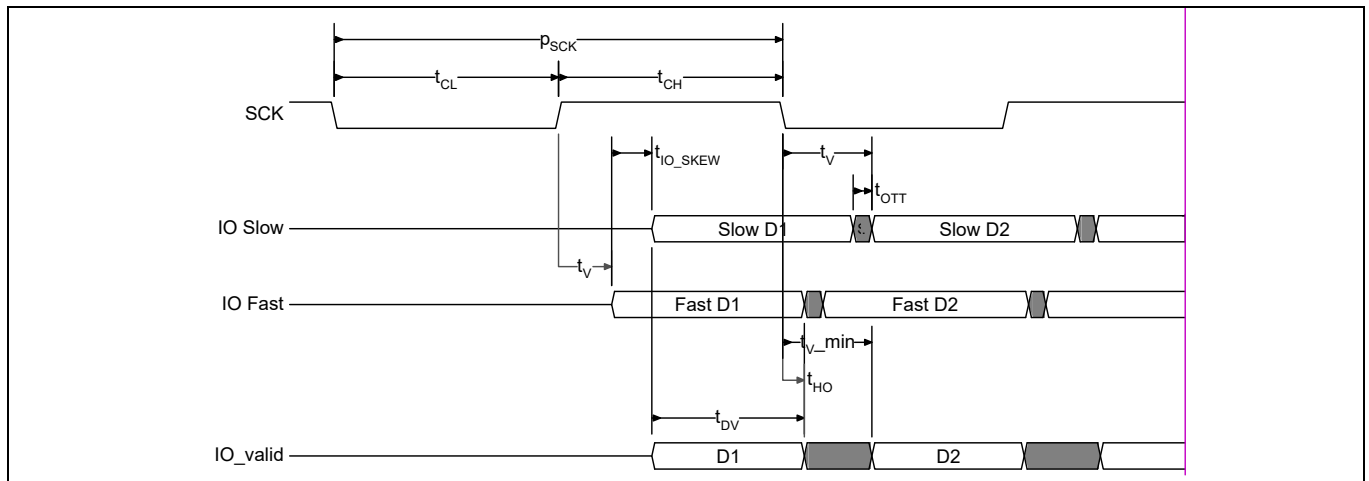


Figure 25 SPI DDR output timing

### 6.5.3 DDR data valid timing using DLP



**Figure 26 SPI DDR data valid window**

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

$$t_{DV} = \text{Minimum half clock cycle time } (t_{CLH})^{[29]} - t_{OTT}^{[29]} - t_{IO\_SKEW}^{[30]}$$

$$t_{V\_min} = t_{HO} + t_{IO\_SKEW} + t_{OTT}$$

**Example:**

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

$$t_{CLH} = 0.45 \times PSCK = 0.45 \times 12.5 \text{ ns} = 5.625 \text{ ns}$$

Bus impedance of 45 ohm and capacitance of 22 pF, with timing reference of  $0.75 \times V_{CC}$ , the rise time from '0' to '1' or fall time from '1' to '0' is  $1.4^{[33]} \times RC$  time constant ( $\text{Tau}$ )<sup>[34]</sup> =  $1.4 \times 0.99 \text{ ns} = 1.39 \text{ ns}$

$$t_{OTT} = \text{rise time or fall time} = 1.39 \text{ ns.}$$

Data valid window

$$t_{DV} = t_{CLH} - t_{IO\_SKEW} - t_{OTT} = 5.625 \text{ ns} - 600 \text{ ps} - 1.39 \text{ ns} = 3.635 \text{ ns}$$

$t_V$  Minimum

$$t_{V\_min} = t_{HO} + t_{IO\_SKEW} + t_{OTT} = 1.0 \text{ ns} + 600 \text{ ps} + 1.39 \text{ ns} = 2.99 \text{ ns}$$

**Notes**

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

Physical interface

## 7 Physical interface

### 7.1 Dual-Quad SOIC 16-lead package

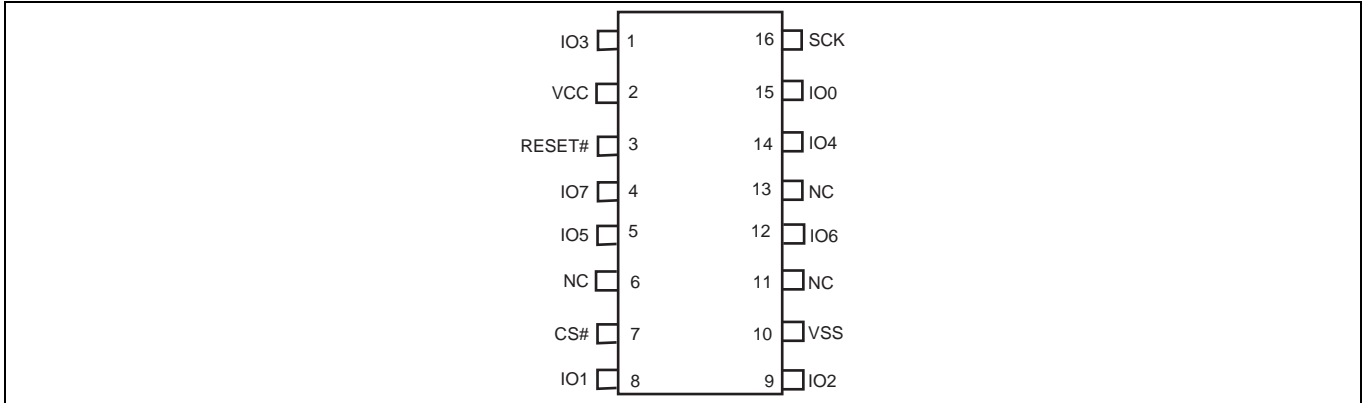


Figure 27 16-pin SOIC package (300 mil)

## 7.2 SOIC 16 physical diagram

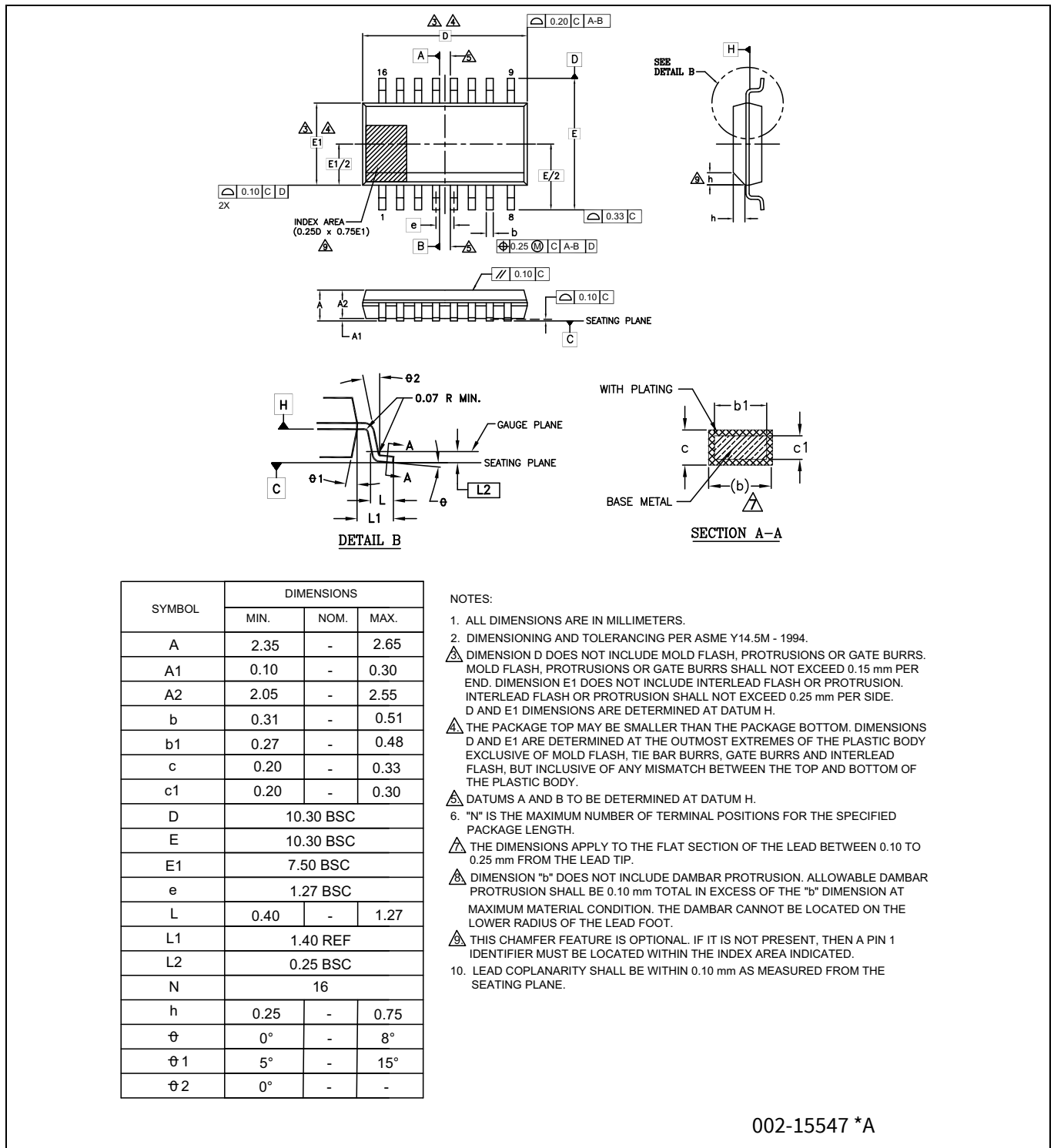


Figure 28 16-pin SOIC (10.30 × 7.50 × 2.65 mm) package outline, 002-15547

### 7.2.1 Special handling instructions for FBGA packages

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

## **8 Address space maps**

### **8.1 Overview**

#### **8.1.1 Extended address**

The S25FL-S devices support 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 MB (128 Mb) 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 and after 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 Mb of a device.

#### **8.1.2 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.

## 8.2 Flash memory array

The main flash array is divided into erase units called sectors. The sectors are organized as uniform 512-KB sectors.

**Table 13 S79FL512S sector and memory address map, bottom 8-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (byte address)	Notes
8	32	SA00	00000000h–00000FFFh	Sector starting address
		:	:	—
		SA31	0001F000h–0001FFFFh	
128	510	SA32	00020000h–0002FFFFh	—
		:	:	
		SA541	01FF0000h–01FFFFFFh	Sector ending address

**Table 14 S79FL512S sector and memory address map, top 8-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (byte address)	Notes
128	510	SA00	0000000h–000FFFFh	Sector starting address
		:	:	—
		SA509	01FD0000h–01FDFFFFh	
8	32	SA510	01FE0000h–01FE0FFFh	—
		:	:	
		SA541	01FFF000h–01FFFFFFh	Sector ending address

**Table 15 S79FL512S sector and memory address map, uniform 512-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (8-bit)	Notes
512	128	SA00	0000000h–003FFFFh	Sector starting address
		:	:	—
		SA127	1FC0000h–1FFFFFFh	Sector ending address

**Table 16 S79FL256S sector and memory address map, bottom 8-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (byte address)	Notes
8	32	SA00	00000000h–00000FFFh	Sector starting address
		:	:	—
		SA31	0001F000h–0001FFFFh	
128	254	SA32	00020000h–0002FFFFh	—
		:	:	
		SA285	00FF0000h–00FFFFFFh	Sector ending address



**Table 17 S79FL256S sector and memory address map, top 8-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (byte address)	Notes
128	254	SA00	0000000h–000FFFFh	Sector starting address
		:	:	—
		SA253	00FD0000h–00FDFFFFh	—
8	32	SA254	00FE0000h–00FE0FFFh	—
		:	:	—
		SA285	00FFF000h–00FFFFFFh	Sector ending address

**Table 18 S79FL256S sector and memory address map, uniform 512-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (byte address)	Notes
512	64	SA00	0000000h–003FFFFh	Sector starting address
		:	:	—
		SA63	0FC0000h–0FFFFFFh	Sector ending address

**Note** These are condensed tables that use a couple of sectors as references. There are address ranges that are not explicitly listed.

### 8.3 ID-CFI address space

The RDIDJ 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 116 for the tables defining the contents of the ID-CFI address space. The ID-CFI address space is programmed by Infineon and read-only for the host system.

### 8.4 OTP address space

Each S79FL256S/S79FL512S memory device has a 2048-byte one time program (OTP) address space that is separate from the main flash array. The OTP area is divided into 64, individually lockable, 32-byte aligned and length regions.

In each 32-byte region starting at address zero:

- The 16 lowest address bytes are programmed by Infineon with a 128-bit random number. Only Infineon 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 Infineon. 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 Infineon.

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

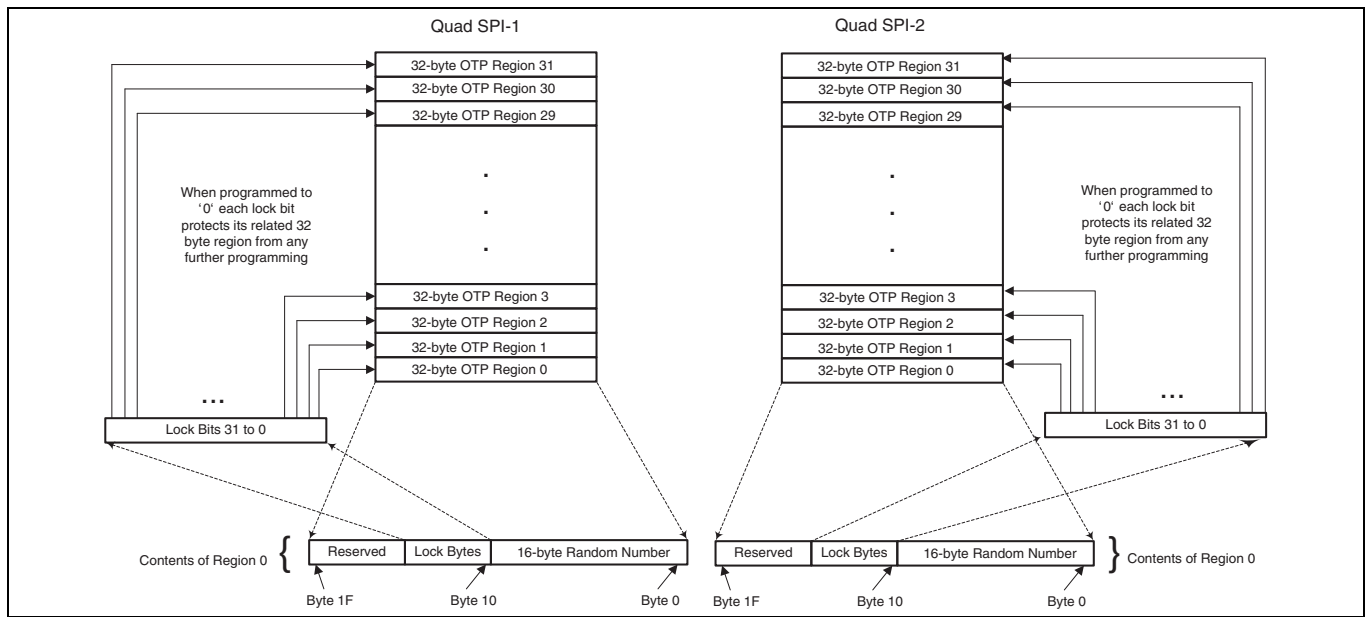
Address space maps

Refer to **Figure 29** 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 Infineon, 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.

During the programming of each OTP region, bits 0–3 are programmed on Quad SPI-1 via IO0–IO3, and bits 4–7 are programmed on Quad SPI-2 via IO4–IO7.



**Figure 29** OTP address space — Quad SPI-1 and SPI-2

**Table 19 OTP address map for quad SPI-1 and quad SPI-2**

Region	Byte address range (Hex)	Contents	Initial delivery state (Hex)
Region 0	000	Least Significant Byte of Infineon programmed random number	Infineon programmed random number
	...	...	
	00F	Most Significant Byte of Infineon programmed random number	
	010 to 013	Region Locking Bits Byte 10 [bit 0] locks region 0 from programming when = 0 ... Byte 13 [bit 7] locks region 31 from programming when = 0	All bytes = FF <sup>[35]</sup>
	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	7E0 to 7FF	Available for user programming	All bytes = FF

**Note**

35.It is recommended that the lock bytes for quad SPI-1 and quad SPI-2 be programmed with identical data.

## **8.5 Registers**

Registers are small groups of memory cells used to configure how the S25FL-S dual-quad SPI memory devices operate 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.

Each S79FL256S/S79FL512S dual-quad SPI device has a register of each type, one for each individual die. These include the Status Register-1, Status Register-2, Configuration Register, AutoBoot Register, Bank Address Register, ECC Status Register, ASP Register, Password Register, PPB Lock Register, PPB Access Register, DYB Access Register, and DDR Data Learning Registers. Each register must be accessed by a command given in parallel to IO0–IO3 (Quad SPI-1) and for IO4–IO7 (Quad SPI-2). Reading and writing to each of these registers must also be done in parallel for IO0–IO3 (Quad SPI-1) and for IO4–IO7 (Quad SPI-2).

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 Infineon. Non-volatile bits have the same cycling (erase and program) endurance as the main flash array.

### 8.5.1 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 0 = No protection, even when WP# is low
6	P_ERR	Programming Error occurred	Volatile, Read only	0	1 = Error occurred. 0 = No Error
5	E_ERR	Erase Error occurred	Volatile, Read only	0	1 = Error occurred 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 0 = Device ignores Write Registers (WRR), program or erase commands 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 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 SRWD, BP2, BP1, and BP0 bits of the Status Register become read-only bits and the Write Registers (WRR) command is no longer accepted for execution. 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 '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 '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 '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 '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 57 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 '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 '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 '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 '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 '1', the WIP bit will remain set to one indicating the device remains busy and unable to receive new operation commands. A Clear Status Register (CLSR) command must be received to return the device to standby mode. When the WIP bit is cleared to '0', no operation is in progress. This is a read-only bit.

## 8.5.2 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 See Latency Code tables
6	LC0			0	
5	TBPROT	Configures Start of Block Protection	OTP	0	1 = BP starts at bottom (Low address) 0 = BP starts at top (High address)
4	DNU	DNU	OTP	0	Do Not Use
3	BPNV	Configures BP2–0 in Status Register	OTP	0	1 = Volatile 0 = Non-Volatile
2	TBPARAM	Configures Parameter Sectors location	OTP	0	1 = 8-KB physical sectors at top, (high address) 0 = 8-KB physical sectors at bottom (Low address) RFU in uniform sector devices
1	QUAD	Puts the device into Quad I/O operation	Non-Volatile	1	1 = Quad For the S79FL-S Dual-Quad SPI device, the default state is set for QUAD and should not be changed.
0	FREEZE	Lock current state of BP2–0 bits in Status Register, TBPROT in Configuration Register, and OTP regions	Volatile	0	1 = Block Protection and OTP locked 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 Infineon.

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  $\leq 50$  MHz with any latency code setting. Similarly, only the Fast Read command is supported up to 133 MHz but the same 10b latency code is used for Fast Read up to 133 MHz and for the other dual and quad read commands up to 104 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.

Address space maps

The enhanced high performance settings provide latency options the same or faster than additional alternate source SPI memories.

Read DDR data learning pattern (DLP) bits may be placed within the dummy cycles immediately before the start of read data, if there are 5 or more dummy cycles. See **“Read memory array commands”** on page 84 for more information on the DLP.

**Table 22 Latency codes for SDR enhanced high performance**

Freq. (MHz)	LC	Read (03h, 13h)		Fast Read (0Bh, 0Ch)		Read Quad Out (6Bh, 6Ch)		Quad I/O Read (EBh, ECh)	
		Mode	Dummy	Mode	Dummy	Mode	Dummy	Mode	Dummy
		≤ 50	11	0	0	0	0	0	0
≤ 80	00	-	-	0	8	0	8	2	4
≤ 90	01	-	-	0	8	0	8	2	4
≤ 104	10	-	-	0	8	0	8	2	5
≤ 133	10	-	-	0	8	-	-	-	-

**Table 23 Latency codes for DDR enhanced high performance**

Freq. (MHz)	LC	DDR Quad I/O Read (EDh, EEh)	
		Mode	Dummy
		≤ 50	11
≤ 80	00	1	6

**Note**

36. When using DDR I/O commands with the DLP enabled, a latency code that provides five or more dummy cycles should be selected to allow one cycle of additional time for the host to stop driving before the memory starts driving the four cycle DLP. So it is recommended to use LC 00 for DDR Quad IO Read, if the DLP for DDR is used.



**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 ‘0’, the Block Protection is defined to start from the top (maximum address) of the array. When TBPROT is set to ‘1’, the Block Protection is defined to start from the bottom (zero address) of the array. The TBPROT bit is OTP and set to ‘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]:** Do Not 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 ‘0’ with the BP bits cleared to 000 when shipped from Infineon. When BPNV is set to ‘0’, the BP2–0 bits in the Status Register are non-volatile. When BPNV is set to ‘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. 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 thirty-two 8-KB small sectors (SMS), which replace two 128-KB sectors. When TBPARM is set to ‘1’, the parameter block is in the top of the memory array address space. When TBPARM is set to ‘0’, the parameter block is at the Bottom of the array. TBPARM is OTP and set to ‘0’, when it ships from Infineon. 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]).

**Quad Data Width (QUAD) CR1[1]:** When set to ‘1’, this bit switches the data width of the device to 4-bit Quad mode. The commands for Serial Read still function normally. The QUAD bit in the S25FL-S devices is factory set to ‘1’ and should not be changed.

**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.

Address space maps

### 8.5.3 Status Register-2 (SR2)

Related commands: Read Status Register-2 (RDSR2 07h).

**Table 24 Status Register-2 (SR2)**

Bits	Field name	Function	Type	Default state	Description
7	RFU	Reserved		0	Reserved for Future Use
6	RFU	Reserved		0	Reserved for Future Use
5	RFU	Reserved		0	Reserved for Future Use
4	RFU	Reserved		0	Reserved for Future Use
3	RFU	Reserved		0	Reserved for Future Use
2	RFU	Reserved		0	Reserved for Future Use
1	ES	Erase Suspend	Volatile, Read only	0	1 = In erase suspend mode 0 = Not in erase suspend mode
0	PS	Program Suspend	Volatile, Read only	0	1 = In program suspend mode 0 = Not in program suspend mode

**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 Erase Suspend and Resume Commands (75h) (7Ah) 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 **“Program Suspend (PGSP 85h) and Resume (PGRS 8Ah)”** on page 96 for details.

### 8.5.4 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 25 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 0 = AutoBoot is not enabled

### 8.5.5 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 26 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. 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	A25 for 1 Gb device
0	RFU	Bank Address	Volatile	0	RFU for lower density device

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.

### 8.5.6 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 “**Automatic ECC**” on page 92.

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 27 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 0 = No error.
1	EECCD	Error in ECC unit data	Volatile, Read only	0	1 = Single Bit Error corrected in ECC unit data. 0 = No error.
0	ECCDI	ECC Disabled	Volatile, Read only	0	1 = ECC is disabled in the selected ECC unit. 0 = ECC is enabled in the selected ECC unit.

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

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

### 8.5.7 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 28 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 [37]	
7	RFU	Reserved	OTP	Note [37]	
6	RFU	Reserved	OTP	1	
5	RFU	Reserved	OTP	Note [37]	
4	RFU	Reserved	OTP	Note [37]	
3	RFU	Reserved	OTP	Note [37]	
2	PWDMLB	Password Protection Mode Lock Bit	OTP	1	0 = Password Protection Mode permanently enabled. 1 = Password Protection Mode not permanently enabled.
1	PSTMLB	Persistent Protection Mode Lock Bit	OTP	1	0 = Persistent Protection Mode permanently enabled. 1 = Persistent Protection Mode not permanently enabled.
0	RFU	Reserved	OTP	1	Reserved for Future Use

**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'.

**Note**

37.Default value depends on ordering part number, see **“Initial delivery state”** on page 129.

Address space maps

### 8.5.8 Password Register (PASS)

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

**Table 29 Password Register (PASS)**

Bits	Field name	Function	Type	Default state	Description
63to0	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'.

### 8.5.9 PPB Lock Register (PPBL)

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

**Table 30 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 Password Protection Mode = 0	0 = PPB array protected until next power cycle or hardware reset 1 = PPB array may be programmed or erased.

### 8.5.10 PPB Access Register (PPBAR)

Related commands: PPB Read (PPBRD E2h)

**Table 31 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. 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.

Address space maps

### 8.5.11 DYB Access Register (DYBAR)

Related commands: DYB Read (DYBRD E0h) and DYB Program (DYBP E1h).

**Table 32 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. 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.

### 8.5.12 SPI DDR Data Learning Registers

Related commands: Program NVDLR (PNVDLR 43h), Write VDLR (WVDLR 4Ah), Data Learning Pattern Read (DLPRD 41h).

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

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

**Table 33 Non-Volatile Data Learning Register (NVDLR)**

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

**Table 34 Volatile Data Learning Register (NVDLR)**

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

Data protection

## 9 Data protection

### 9.1 Secure silicon region (OTP)

The device has a 2048-byte one-time program (OTP) address space that is separate from the main flash array. The OTP area is divided into 32, individually lockable, 64-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 41, “[One Time Program Array commands](#)” on page 103, and “[OTP Read \(OTPR 4Bh\)](#)” on page 103.

#### 9.1.1 Reading OTP memory space

The OTP Read command uses the same protocol as Fast Read. OTP Read operations outside the valid 2-KB OTP address range will yield indeterminate data.

#### 9.1.2 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 29](#). 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’.

#### 9.1.3 Infineon programmed random number

Infineon 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.

#### 9.1.4 Lock bytes

The LSb of each Lock byte protects the lowest address region related to the byte, the MSb protects the highest address region related to the byte. The next higher address byte similarly protects the next higher 8 regions. The LSb bit of the lowest address lock byte protects the higher address 16 bytes of the lowest address region. In other words, the LSb of location 0x10 protects all the lock bytes and RFU bytes in the lowest address region from further programming. See “[OTP address space](#)” on page 41.



Data protection

## 9.2 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)

## 9.3 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 35 Upper array start of protection (TBPROT = 0)**

Status register content			Protected fraction of memory array	Protected memory (KB)	
BP2	BP1	BP0		S79FL256S 256 Mb	S79FL512S 512 Mb
0	0	0	None	0	0
0	0	1	Upper 64th	512	1024
0	1	0	Upper 32nd	1024	2048
0	1	1	Upper 16th	2048	4096
1	0	0	Upper 8th	4096	8192
1	0	1	Upper 4th	8192	16384
1	1	0	Upper half	16384	32768
1	1	1	All Sectors	32768	65536

Data protection

**Table 36 Lower array start of protection (TBPROT = 1)**

Status register content			Protected fraction of memory array	Protected memory (KB)	
BP2	BP1	BP0		S79FL256S 256 Mb	S79FL512S 512 Mb
0	0	0	None	0	0
0	0	1	Lower 64th	512	1024
0	1	0	Lower 32nd	1024	2048
0	1	1	Lower 16th	2048	4096
1	0	0	Lower 8th	4096	8192
1	0	1	Lower 4th	8192	16384
1	1	0	Lower half	16384	32768
1	1	1	All Sectors	32768	65536

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.

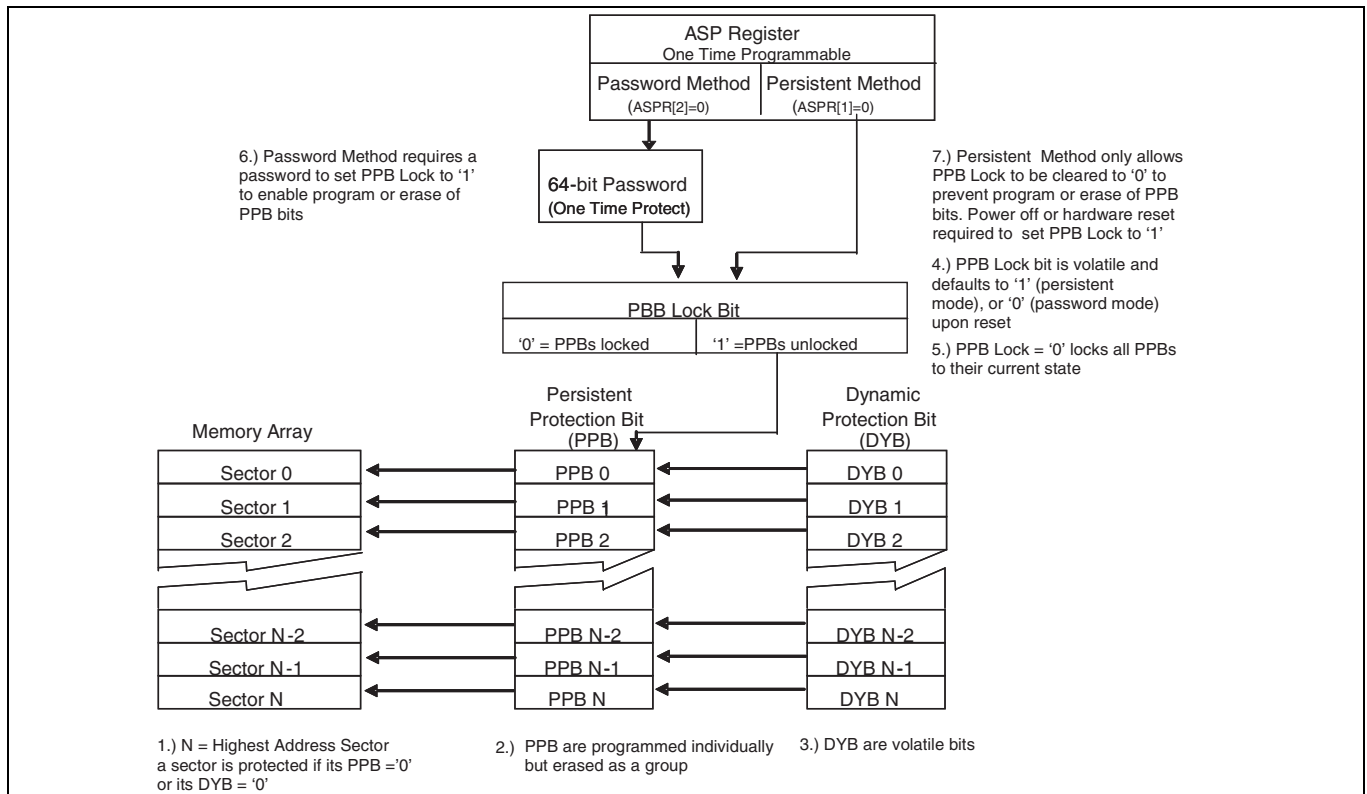
### 9.3.1 Freeze bit

Bit 0 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.

## 9.4 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 30**.

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 57 for full details of the BP2–0 bits.



**Figure 30** 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.

Data protection

### 9.4.1 ASP Register

The ASP Register is used to permanently configure the behavior of Advanced Sector Protection (ASP) features. See [Table 28](#).

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 zero results in a programming failure.

ASP register programming rules:

- If the password mode is chosen, the password must be programmed prior to setting the Protection Mode Lock Bits.
- Once the Protection Mode is selected, the 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 45 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.

### 9.4.2 Persistent protection bits

The persistent protection bits (PPB) are located in a separate non-volatile 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. Pre-programming 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.

#### Persistent protection 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.

### 9.4.3 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.

Data protection

#### 9.4.4 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.

#### 9.4.5 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.
- 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 37 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

#### 9.4.6 Persistent Protection mode

The Persistent Protection method sets the PPB Lock bit to '1' during POR or hardware reset so that the PPB bits are unprotected by a device hardware reset. Software reset does not affect the PPB Lock bit. The PLBWR command can clear the PPB Lock bit to '0' to protect the PPB. There is no command to set the PPB Lock bit therefore the PPB Lock bit will remain at '0' until the next power-off or hardware reset.

### 9.4.7 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 clears the PPB Lock bit, 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 '1' after a cell is programmed as '0' results in the cell left as '0' with no programming error set.
- The password is all 1's when shipped from Infineon. 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 '1' and the WIP Bit remains set. In this case it is a failure to change the state of the PPB Lock bit because it is still protected by the lack of a valid password.
- The Password Unlock command cannot be accepted any faster than once every  $100 \mu\text{s} \pm 20 \mu\text{s}$ . This makes it take an unreasonably long time (58 million years) for a hacker to run through all the 64-bit combinations in an attempt to correctly match a password. The Read Status Register-1 command may be used to read the WIP bit to determine when the device has completed the password unlock command or is ready to accept a new password command. When a valid password is provided the password unlock command does not insert the  $100 \mu\text{s}$  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.

## 10 Commands

All communication between the host system and the S25FL-S Dual-Quad SPI 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.

Quad output commands provide an address sent to the memory only on the IO0 and IO4 signal. Data will be returned to the host as a sequence of 8-bit (byte) groups on IO0–IO7.

Quad input/output (I/O) commands provide an address sent from the host as four-bit (nibble) groups on Quad SPI-1 IO0–IO3 and Quad SPI-2 IO4–IO7. Data is returned to the host similarly as 8-bit (byte) groups on IO0–IO7.

Commands are structured as follows:

- Each command begins with an eight bit (byte) instruction.
- The instruction may be standalone or may be followed by address bits to select a location within one of several address spaces in the device. The address may be either a 24-bit or 32-bit byte boundary address.
- The S25FL-S SPI with multiple IO provides the option for each transfer of address and data information to be done one, 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 four-bit wide IO bus the memory performance can be increased by using the instructions that provide 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.



Commands

- 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 13.
  - The host always controls the Chip Select (CS#), Serial Clock (SCK), and Serial Input (IO0 and IO4) for single-bit wide transfers. The memory drives the IO0–IO7 signals during 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.

**10.1 Command set summary**

The S79FL256S/S79FL512S dual-quad SPI devices each contain two quad SPI devices (Quad SPI-1 and Quad SPI-2) stacked in a Dual Die Package (DDP). Both devices are selected to decode each command instruction and address when the CS# signal, shared by both devices, goes low. Quad SPI-1 device responds to commands, address, data in and data out on IO0–IO3. Quad SPI-2 device responds to commands, address, data in and data out on IO4–IO7. All commands are executed by both devices in parallel.

Both quad SPI devices must be configured, by writing to the various Status and Configuration Registers in parallel, to define the same overall sector map and behavior of both devices, selected by each CS# for the DDP.

**10.1.1 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.

**Table 38 Instructions and corresponding details**

<b>Instruction name</b>	<b>Description</b>	<b>Code (Hex)</b>
4FAST_READ	Read Fast (4-byte address)	0C
4READ	Read (4-byte address)	13
4QOR	Read Quad Out (4-byte address)	6C
4QIOR	Quad I/O Read (4-byte address)	EC
4DDRQIOR	DDR Quad I/O Read (4-byte address)	EE
4PP	Page Program (4-byte address)	12
4QPP	Quad Page Program (4-byte address)	34
4P4E	Parameter 8-KB Erase (4-byte address)	21
4SE	Erase 512 KB (4-byte address)	DC



Commands

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.

**Table 39 Instructions and corresponding details**

Instruction name	Description	Code (Hex)
READ	Read (3-byte address)	03
FAST_READ	Read Fast (3-byte address)	0B
QOR	Read Quad Out (3-byte address)	6B
QIOR	Quad I/O Read (3-byte address)	EB
DDRQIOR	DDR Quad I/O Read (3-byte address)	ED
PP	Page Program (3-byte address)	02
QPP	Quad Page Program (3-byte address)	32
P4E	Parameter 8-KB Erase (3-byte address)	20
SE	Erase 128 / 512 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-Mb (16-MB) 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 Mb 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 MB.
- c. Bank Register bits are volatile.
  - i. On power-up, the default is Bank0 (the lowest address 16 MB).
- 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 40 Bank address map**

Bank Address Register bits		Bank	Memory array address range (Hex)	
Bit 1	Bit 0			
0	0	0	00000000	00FFFFFF
0	1	1	01000000	01FFFFFF
1	0	2	02000000	02FFFFFF
1	1	3	03000000	03FFFFFF

### 10.1.2 Command summary sorted by function

**Table 41 S79FL256S/S79FL512S 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	133
	RDID	Read ID (JEDEC Manufacturer ID and JEDEC CFI)	9F	133
	RES	Read Electronic Signature	AB	50
Register Access	RDSR1	Read Status Register-1	05	133
	RDSR2	Read Status Register-2	07	133
	RDCR	Read Configuration Register-1	35	133
	WRR	Write Register (Status-1, Configuration-1)	01	133
	WRDI	Write Disable	04	133
	WREN	Write Enable	06	133
	CLSR	Clear Status Register-1 - Erase/Prog. Fail Reset	30	133
	ECCRD	ECC Read (4-byte address)	18	133
	ABRD	AutoBoot Register Read	14	133 (QUAD = 0) 104 (QUAD = 1)
	ABWR	AutoBoot Register Write	15	133
	BRRD	Bank Register Read	16	133
	BRWR	Bank Register Write	17	133
	BRAC	Bank Register Access (Legacy Command formerly used for Deep Power Down)	B9	133
	DLPRD	Data Learning Pattern Read	41	133
	PNVDLR	Program NV Data Learning Register	43	133
WVDLR	Write Volatile Data Learning Register	4A	133	
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	133
	4FAST_READ	Fast Read (4-byte address)	0C	133
	QOR	Read Quad Out (3- or 4-byte address)	6B	104
	4QOR	Read Quad Out (4-byte address)	6C	104
	QIOR	Quad I/O Read (3- or 4-byte address)	EB	104
	4QIOR	Quad I/O Read (4-byte address)	EC	104
	DDRQIOR	DDR Quad I/O Read (3- or 4-byte address)	ED	80
	4DDRQIOR	DDR Quad I/O Read (4-byte address)	EE	80

Commands

**Table 41 S79FL256S/S79FL512S command set (sorted by function) (Continued)**

Function	Command name	Command description	Instruction value (Hex)	Maximum frequency (MHz)
Program Flash Array	PP	Page Program (3- or 4-byte address)	02	133
	4PP	Page Program (4-byte address)	12	133
	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	133
	PGRS	Program Resume	8A	133
Erase Flash Array	P4E	Parameter 8-kb, sector Erase (3- or 4-byte address)	20	133
	4P4E	Parameter 8-kb, sector Erase (4-byte address)	21	133
	BE	Bulk Erase	60	133
	BE	Bulk Erase (alternate command)	C7	133
	SE	Erase 128 KB or 512 KB (3- or 4-byte address)	D8	133
	4SE	Erase 128 KB or 512 KB (4-byte address)	DC	133
	ERSP	Erase Suspend	75	133
	ERRS	Erase Resume	7A	133
One Time Program Array	OTPP	OTP Program	42	133
	OTPR	OTP Read	4B	133
Advanced Sector Protection	DYBRD	DYB Read	E0	133
	DYBWR	DYB Write	E1	133
	PPBRD	PPB Read	E2	133
	PPBP	PPB Program	E3	133
	PPBE	PPB Erase	E4	133
	ASPRD	ASP Read	2B	133
	ASPP	ASP Program	2F	133
	PLBRD	PPB Lock Bit Read	A7	133
	PLBWR	PPB Lock Bit Write	A6	133
	PASSRD	Password Read	E7	133
	PASSP	Password Program	E8	133
PASSU	Password Unlock	E9	133	
Reset	RESET	Software Reset	F0	133
	MBR	Mode Bit Reset	FF	133
Reserved for Future Use	MPM	Reserved for Multi-I/O-High Performance Mode (MPM)	A3	133
RFU	Reserved-18	Reserved	18	
RFU	Reserved-E5	Reserved	E5	
RFU	Reserved-E6	Reserved	E6	

### 10.1.3 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 S79FL256S/S79FL512S devices support the three most common device information commands.

### 10.1.4 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.

#### 10.1.4.1 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 one indicating the device remains busy. Under this condition, only the CLSR, WRDI, RDSR1, RDSR2, and software RESET commands are valid commands. 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.

#### 10.1.4.2 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.

### 10.1.5 Read flash array

Data may be read from the memory starting at any byte boundary. Data bytes are sequentially read from incrementally higher byte addresses until the host ends the data transfer by driving CS# input HIGH. If the byte address reaches the maximum address of the memory array, the read will continue at address zero of the array. There are several different read commands to specify different access latency and data path widths. Double Data Rate (DDR) commands also define the address and data bit relationship to both SCK edges:

- The Read command provides a single address bit per SCK rising edge on the IO0 and IO4 signal with read data returning a single bit per SCK falling edge on the IO1 and IO5 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 IO0 and IO4 signal with read data returning a single bit per SCK falling edge on the IO1 and IO5 signal and may operate up to 133 MHz.
- Quad Output Read commands provide address a single bit per SCK rising edge on the IO0 and IO4 signal with read data returning four bits of data per SCK falling edge on the IO0–IO7 signals.
- Quad I/O Read commands provide address four bits per SCK rising edge with read data returning four bits of data per SCK falling edge on the IO0–IO7 signals.
- Quad Double Data Rate Read command provides address four bits per every SCK edge with read data returning four bits of data per every SCK edge on the IO0–IO7 signals. Double Data Rate (DDR) operation is only supported for core and I/O voltages of 3 V to 3.6 V.

### **10.1.6 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 512 or 1024 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.

### **10.1.7 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 '0'. While bits can be individually programmed from '1' to '0', erasing bits from '0' to '1' must be done on a sector-wide (SE) or array-wide (BE) level.

### **10.1.8 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.

### **10.1.9 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.

### **10.1.10 Reserved**

Some instructions are reserved for future use. In this generation of the S79FL256S/S79FL512S, 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 S79FL256S/S79FL512S devices 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.

## 10.2 Identification commands

### 10.2.1 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 133 MHz.

For the Dual-Quad SPI device, the Read Identification (REMS) instruction and data read is only done on Quad SPI-1 using IO0 and IO1.

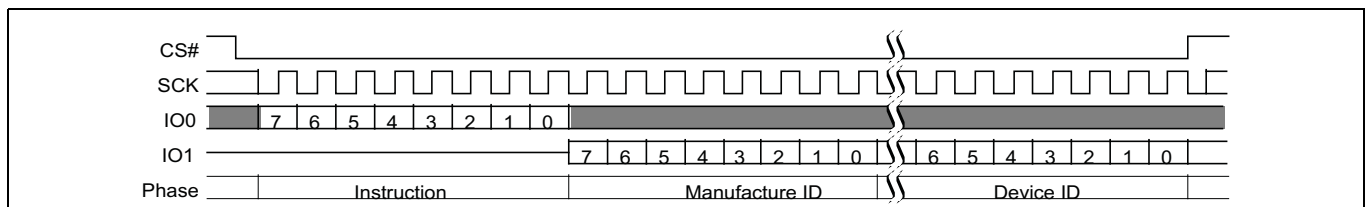


Figure 31 READ\_ID (90h) command sequence

Table 42 Read\_ID values

Device	Manufacturer ID (Hex)	Device ID (Hex)
S79FL256S	01	19
S79FL512S	01	20

### 10.2.2 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 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.

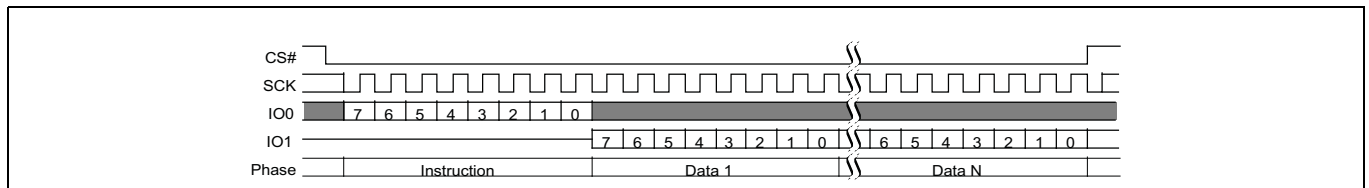
Any RDID command issued while a program, erase, or write cycle is in progress is ignored and has no effect on execution of the program, erase, or write cycle that is in progress.

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 41 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.

For the S79FL256S/S79FL512S Dual-Quad SPI devices, the Read Identification (RDID) instruction and data read is only done on Quad SPI-1 using IO0 and IO1.

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



**Figure 32 Read Identification (RDID 9Fh) command sequence**

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

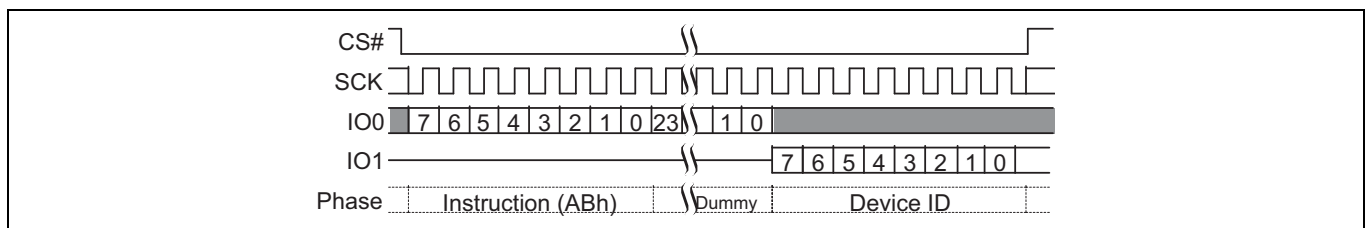
The RES command is used to read a single byte Electronic Signature from SO. RES is only supported for backward compatibility and should not be used for new software designs. New software designs should instead make use of the RDID command.

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.

For the S25FL-S Dual-Quad SPI devices, the Read Electronic Signature (RES) instruction and data read is only done on Quad SPI-1 using IO0 and IO1.



**Figure 33** Read Electronic Signature (RES ABh) command sequence

**Table 43** RES values

Device	Device ID (Hex)
S79FL256S	19
S79FL512S	20



### 10.3 Register Access commands

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

The Read Status Register-1 (RDSR1) command allows the Status Register-1 contents of Quad SPI-1 to be read from IO1 and Quad SPI-2 to be read from IO5. 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 133 MHz.

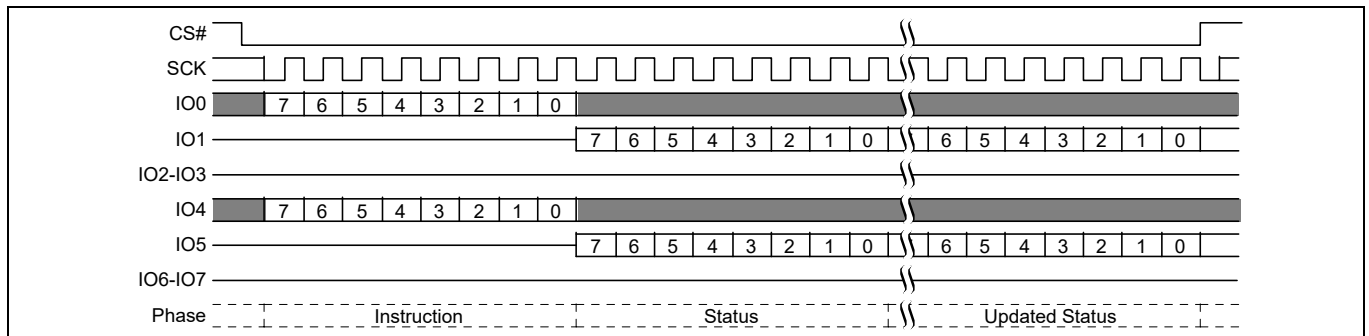


Figure 34 Dual-Quad Read Status Register-1 (RDSR1 05h) command sequence

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

The Read Status Register-2 (RDSR2) command allows the Status Register-2 contents of Quad SPI-1 to be read from IO1 and Quad SPI-2 to be read from IO5. The Status Register-2 contents may be read at any time, even while a program, erase, or write operation is in progress. It is possible to read the Status Register-2 continuously by providing multiples of eight clock cycles. The status is updated for each eight cycle read. The maximum clock frequency for the RDSR2 command is 133 MHz.

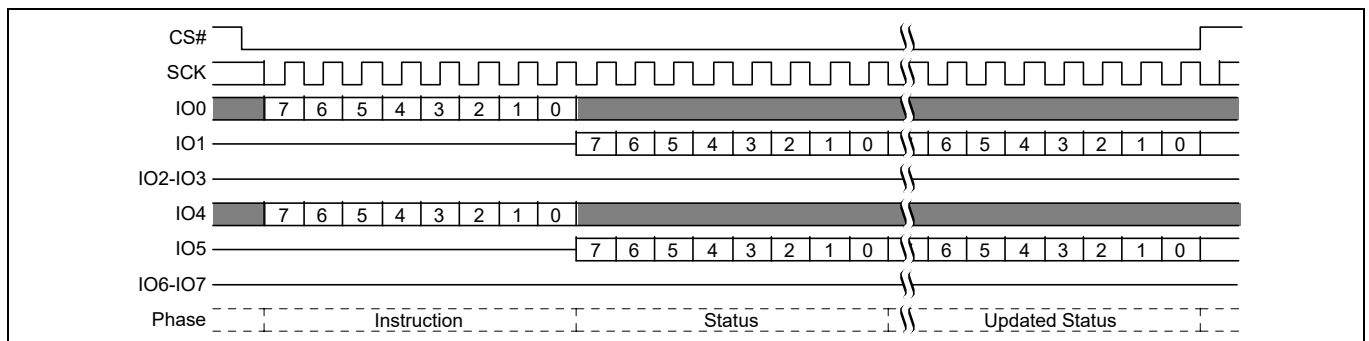
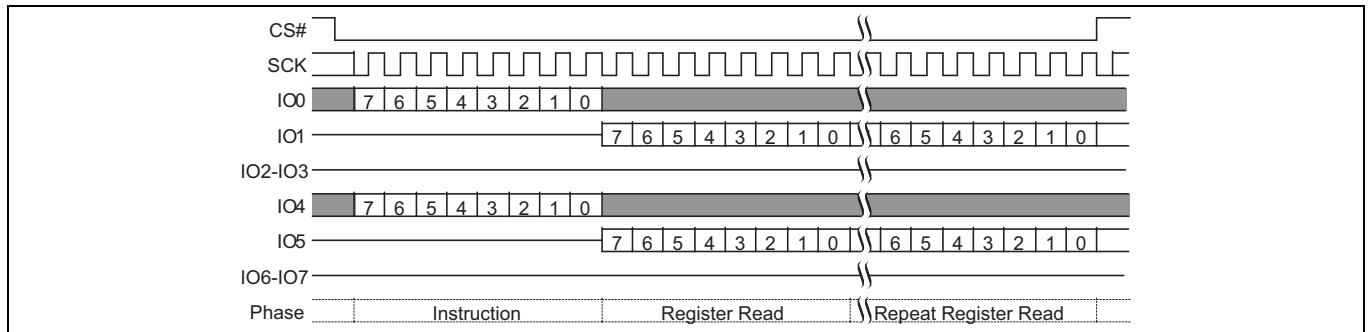


Figure 35 Dual-Quad Read Status Register-2 (RDSR2 07h) command sequence

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

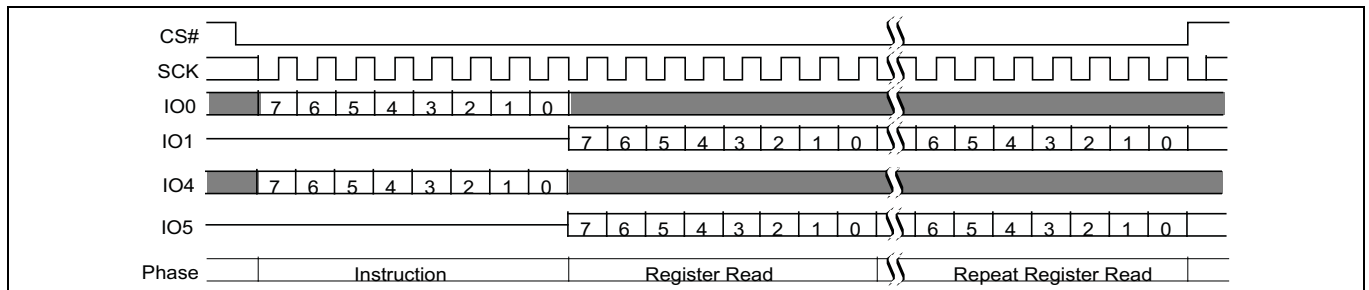
The Read Configuration Register (RDCR) command allows the Configuration Register contents of Quad SPI-1 to be read from IO1 and Quad SPI-2 to be read from IO5. It is possible to read the Configuration Register continuously by providing multiples of eight clock cycles. The Configuration Register contents may be read at any time, even while a program, erase, or write operation is in progress.



**Figure 36** Dual-Quad Read Configuration Register (RDCR 35h) command sequence

### 10.3.4 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 133 MHz.

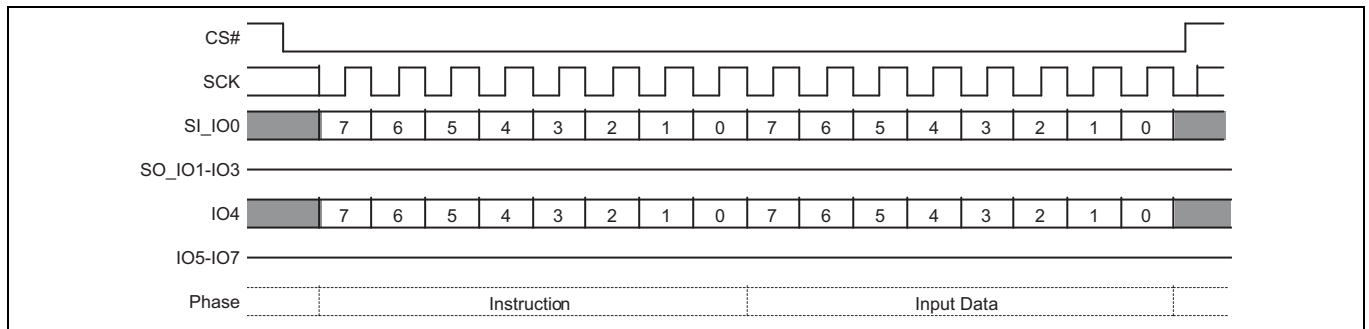


**Figure 37** Read Bank Register (BRRD 16h) command

### 10.3.5 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 Mb (16 MB), 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 '0'.



**Figure 38 Bank Register Write (BRWR 17h) command**

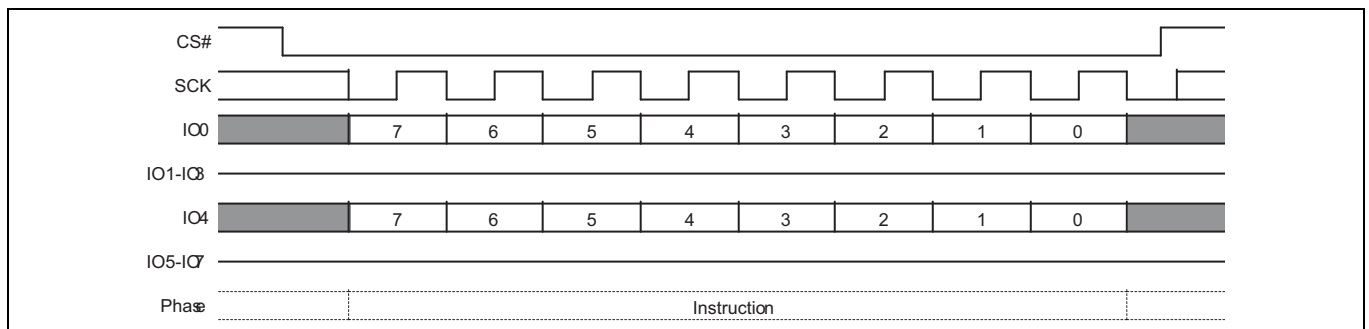
### 10.3.6 Bank Register Access (BRAC B9h)

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 S25FL-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 39 BRAC (B9h) command sequence**

### 10.3.7 Write Registers (WRR 01h)

The Write Registers (WRR) command allows new values to be written to both the Status Register-1 and Configuration Register. 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 for Quad SPI-1 on IO0 and for Quad SPI-2 on IO4. The Status Register is one data byte in length.

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 reserved for the future must be written as '0'.

CS# must be driven to the logic HIGH state after the eighth or sixteenth bit of data has been latched. If not, the Write Registers (WRR) command is not executed. If CS# is driven HIGH after the eighth cycle then only the Status Register-1 is written; otherwise, after the sixteenth cycle both the Status and Configuration Registers are written. When the configuration register QUAD bit CR[1] is '1', only the WRR command format with 16 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 '1' during the self-timed Write Registers (WRR) operation, and is '0' when it is completed. When the Write Registers (WRR) operation is completed, the Write Enable Latch (WEL) is set to '0'. The maximum clock frequency for the WRR command is 133 MHz.

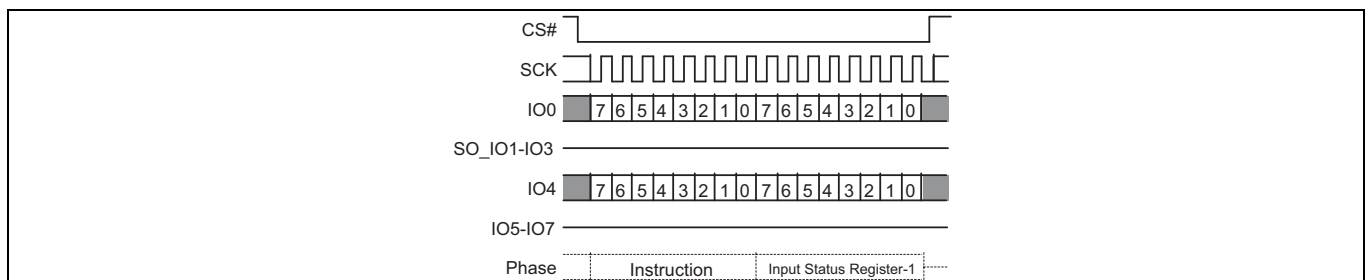


Figure 40 Dual-Quad Write Registers

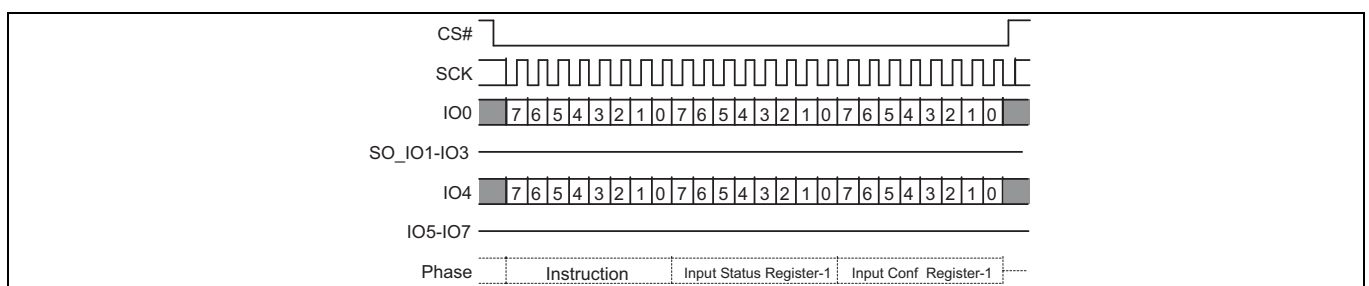


Figure 41 Dual-Quad Write Registers (WRR 01h) command sequence

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 '1' or '0'. The Status Register Write Disable (SRWD) bit allows the BP bits to be hardware protected.

When the Status Register Write Disable (SRWD) bit of the Status Register is '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.

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 76.

### 10.3.8 Write Enable (WREN 06h)

The Write Enable (WREN) command sets the Write Enable Latch (WEL) bit of the Status Register-1 (SR1[1]) to '1'. The Write Enable Latch (WEL) bit must be set to '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 IO0 for Quad SPI-1 and IO4 for Quad SPI-2. Without CS# being driven to the logic HIGH state after the eighth bit of the instruction byte has been latched in on IO0 for Quad SPI-1 and IO4 for Quad SPI-2, the write enable operation will not be executed.

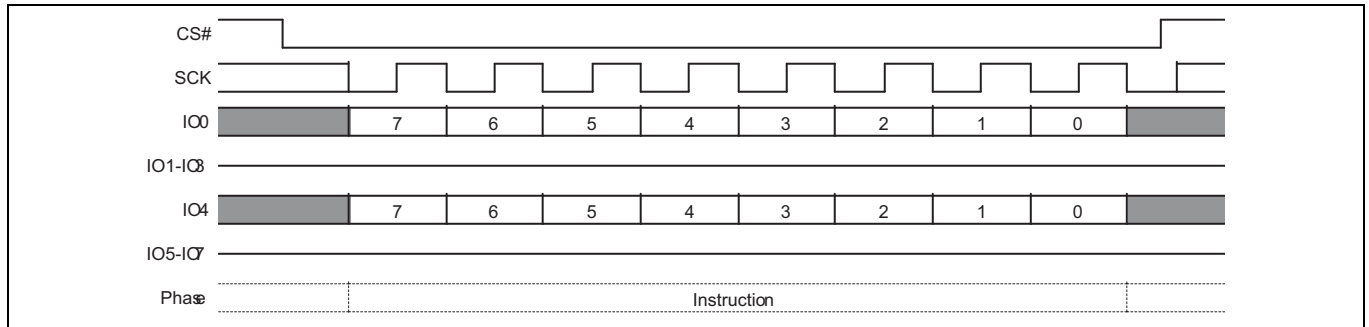


Figure 42 Dual-Quad Write Enable (WREN 06h) command sequence

### 10.3.9 Write Disable (WRDI 04h)

The Write Disable (WRDI) command sets the Write Enable Latch (WEL) bit of the Status Register-1 (SR1[1]) to '0'. The Write Enable Latch (WEL) bit may be set to '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 IO0 for Quad SPI-1 and IO4 for Quad SPI-2. Without CS# being driven to the logic HIGH state after the eighth bit of the instruction byte has been latched in on IO0 for Quad SPI-1 and IO4 for Quad SPI-2, the write disable operation will not be executed.

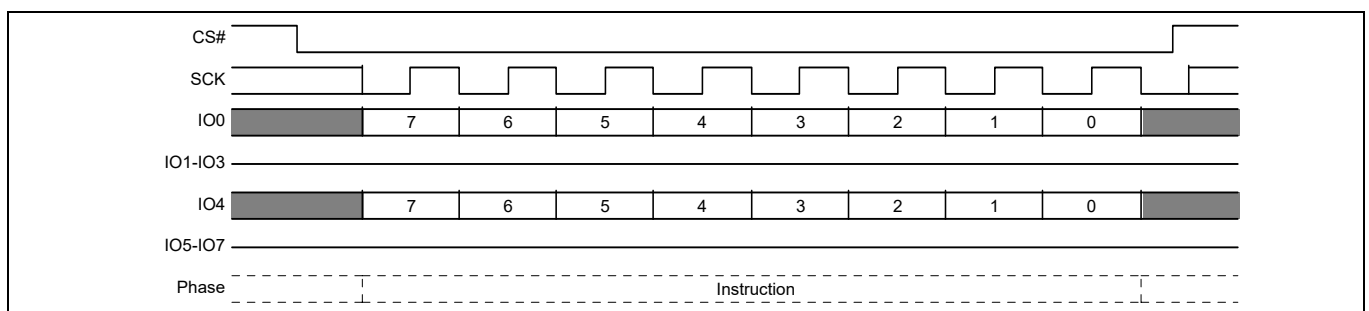
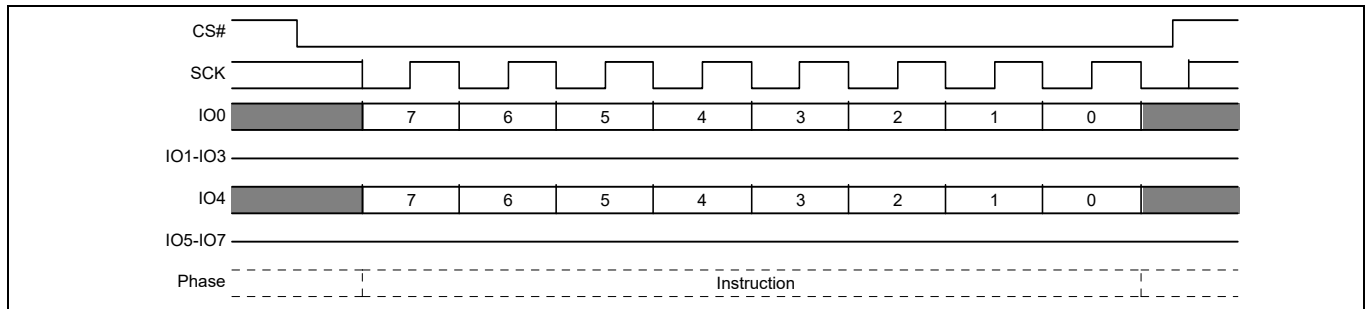


Figure 43 Dual-Quad Write Disable (WRDI 04h) command sequence

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

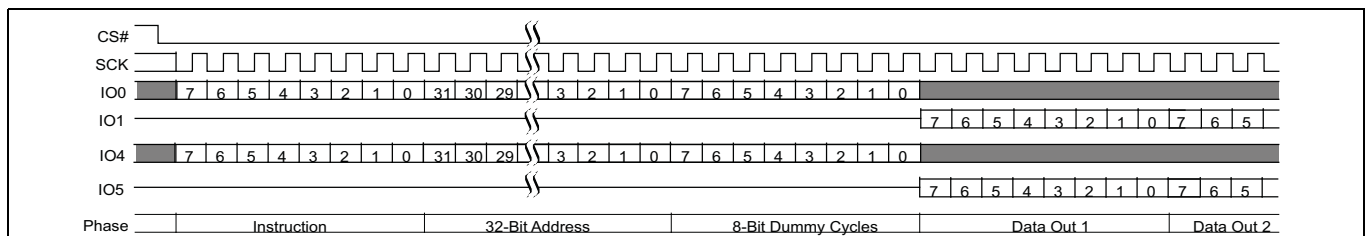
The Clear Status Register command resets bit SR1[5] (Erase Fail Flag) and bit SR1[6] (Program Fail Flag). It is not necessary to set the WEL bit before the Clear SR command is executed. The Clear SR command will be accepted even when the device remains busy with WIP set to '1', as the device does remain busy when either error bit is set. The WEL bit will be unchanged after this command is executed.



**Figure 44** Dual-Quad Clear Status Register (CLSR 30h) command sequence

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

To read the ECC Status Register, the command is followed by the ECC unit (32 bit) address, the five least significant bits (LSb) of address must be set to '0'. 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 [“Automatic ECC”](#) on page 92 for details on ECC unit.



**Figure 45** ECC Status Register Read command sequence

### 10.3.12 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.

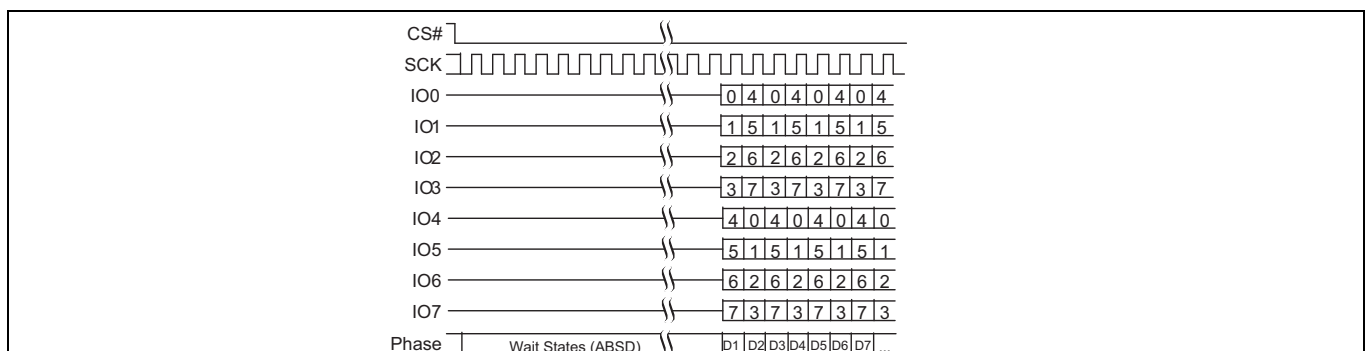
The AutoBoot feature allows the host memory controller to take boot code from an S25FL-S device immediately after the end of reset, without having to send a read command. This saves thirty two or more cycles and simplifies the logic needed to initiate the reading of boot code.

- 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 S25FL-S 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 133 MHz if the QUAD bit CR1[1] is '0' or 104 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 one 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.

With 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.



**Figure 46 AutoBoot Sequence (CR1[1] = 1)**



### 10.3.13 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. The maximum operating clock frequency for ABRD command is 104 MHz.

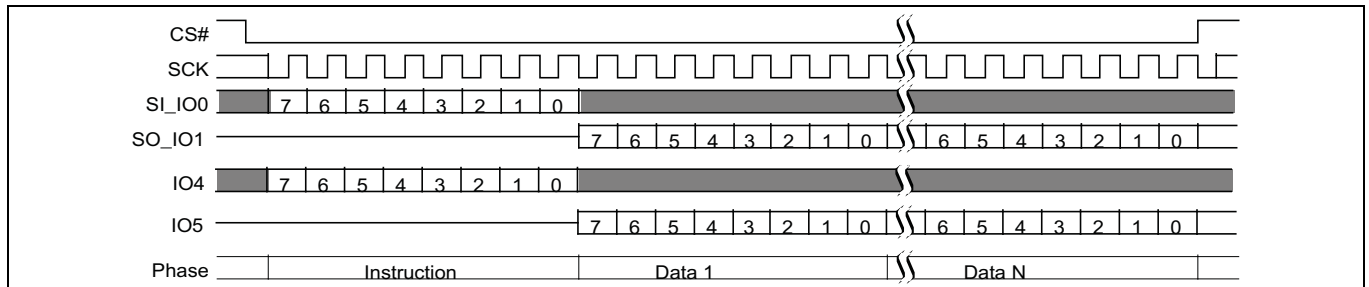


Figure 47 AutoBoot Register Read (ABRD 14h) command

### 10.3.14 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 '1' during the self-timed ABWR operation, and is '0' when it is completed. When the ABWR cycle is completed, the Write Enable Latch (WEL) is set to '0'. The maximum clock frequency for the ABWR command is 133 MHz.

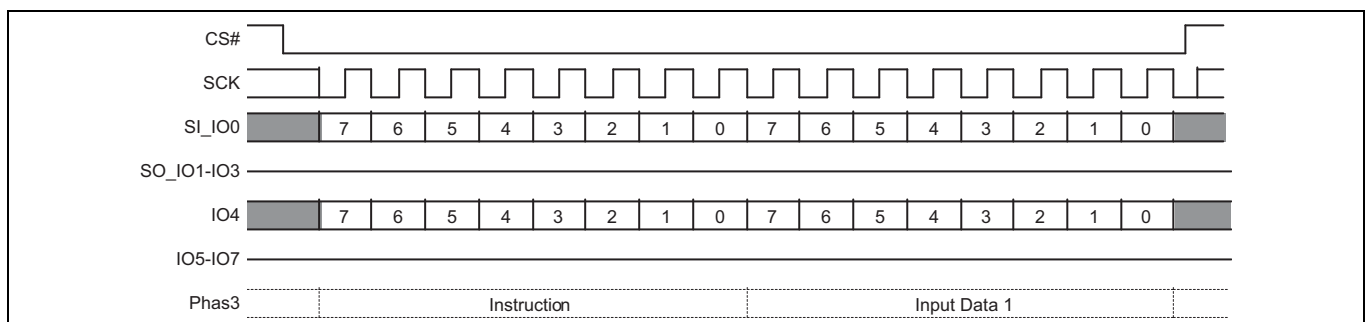


Figure 48 AutoBoot Register Write (ABWR) command

Commands

### 10.3.15 Program NVDLR (PNVDLR 43h)

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

The PNVDLR command is entered by shifting the instruction and the data byte on SI-I/O0 for Quad SPI-1 and IO4 for Quad SPI-2.

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

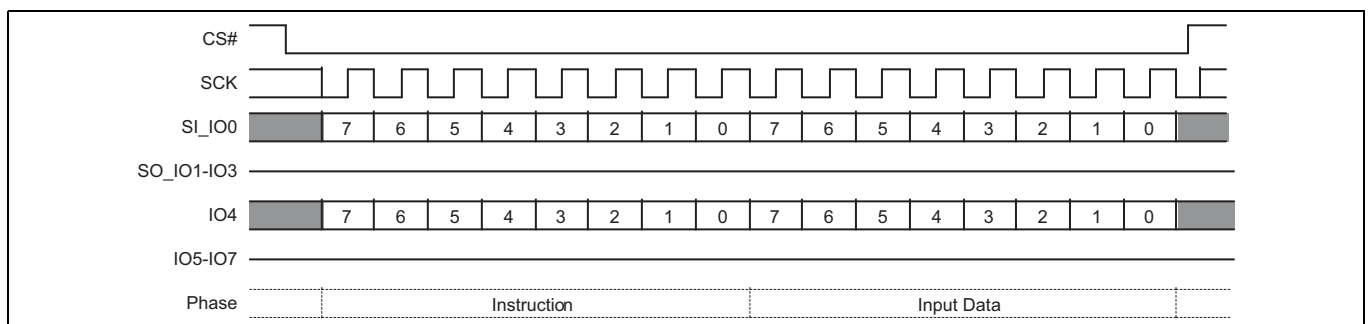


Figure 49 Program NVDLR (PNVDLR 43h) command sequence

### 10.3.16 Write VDLR (WVDLR 4Ah)

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

The WVDLR command is entered by shifting the instruction and the data byte on SI-I/O0 for Quad SPI-1 and IO4 for Quad SPI-2.

CS# must be driven to the logic HIGH state after the eighth (8th) bit of data has been latched. If not, the WVDLR command is not executed. As soon as CS# is driven to the logic HIGH state, the WVDLR operation is initiated with no delays. The maximum clock frequency for the PNVDLR command is 133 MHz.

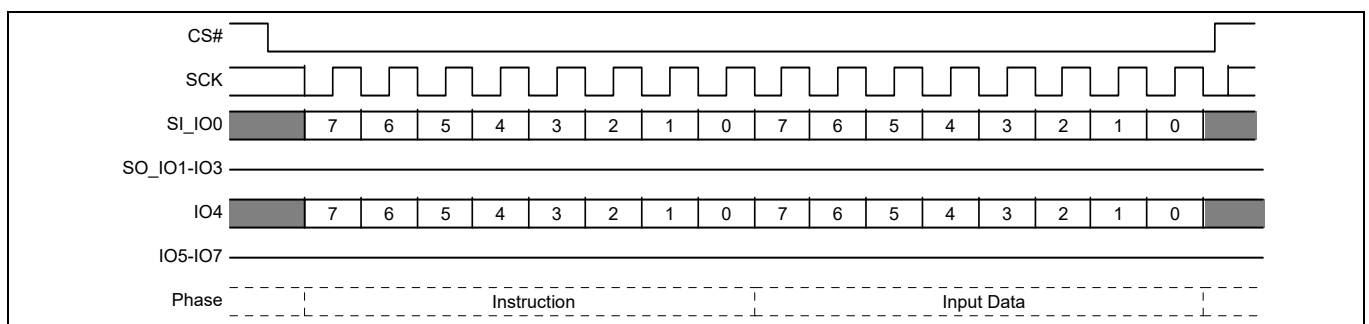


Figure 50 Write VDLR (WVDLR 4Ah) command sequence

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

The instruction is shifted on SI\_IO0, then the 8-bit DLP is shifted out on SO\_IO1 and IO5. It is possible to read the DLP continuously by providing multiples of eight clock cycles. The maximum operating clock frequency for the DLPRD command is 133 MHz.

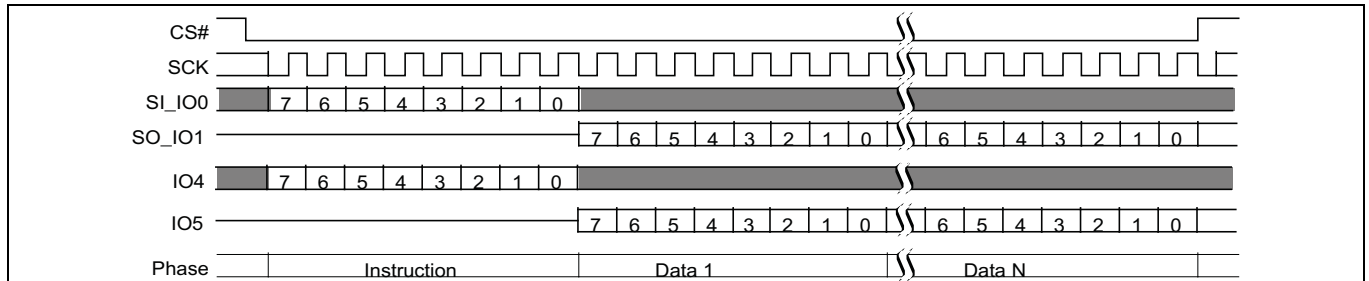


Figure 51 Dual-Quad DLP Read (DLPRD 41h) command sequence

## 10.4 Read memory array commands

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

- Some commands transfer address or data on each rising edge of SCK. These are called Single Data Rate commands (SDR).
- Some SDR commands transfer address one bit per rising edge of SCK and return data 2, or 8 bits of data per rising edge of SCK. These are called Read or Fast Read for 2-bit data; Quad Output for 8-bit data.
- Some SDR commands transfer both address and data 8 bits per rising edge of SCK. These are called Quad I/O for 8 bit.
- Some commands transfer address and data on both the rising edge and falling edge of SCK. These are called Double Data Rate (DDR) commands.
- There are DDR commands for 4 bits of address per each die or 8 bit data per SCK edge. These are called Quad I/O DDR for 8-bit per edge transfer.

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 transferred at SDR or DDR. Commands transferring address or data 4-bits per clock edge per die are called multiple I/O (MIO) commands. For FL-S devices at 256 Mb 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'.

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. When DDR commands are used the host must not drive the I/O signals during any dummy cycle. The number of dummy cycles varies with the SCK frequency or performance option selected via the Configuration Register-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.

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

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

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

Commands

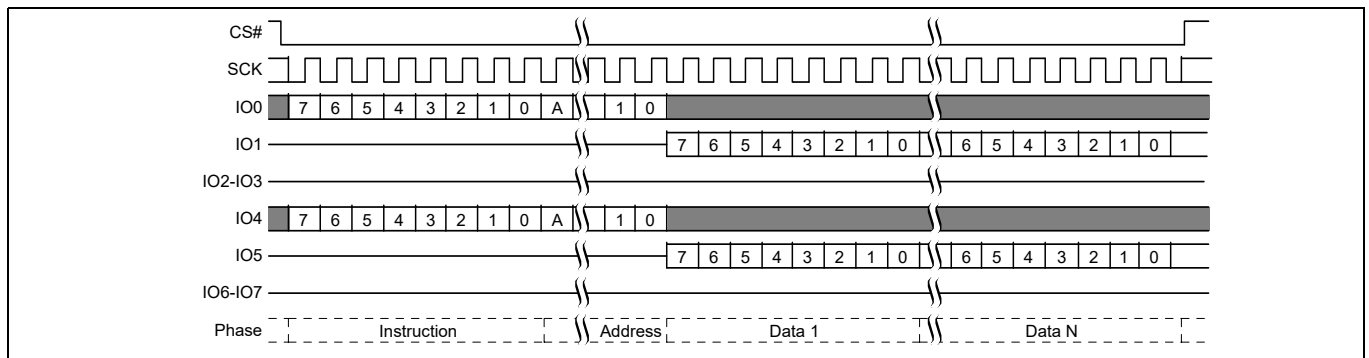
**10.4.1 Read (Read 03h or 4READ 13h)**

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 IO1 and IO5. 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 52 Dual-Quad Read command sequence (READ 03h or 13h)<sup>[38]</sup>**

**Note**

38.A = MSb of address = 23 for command 03h, or 31 for command 13h.

Commands

**10.4.2 Fast Read (FAST\_READ 0Bh or 4FAST\_READ 0Ch)**

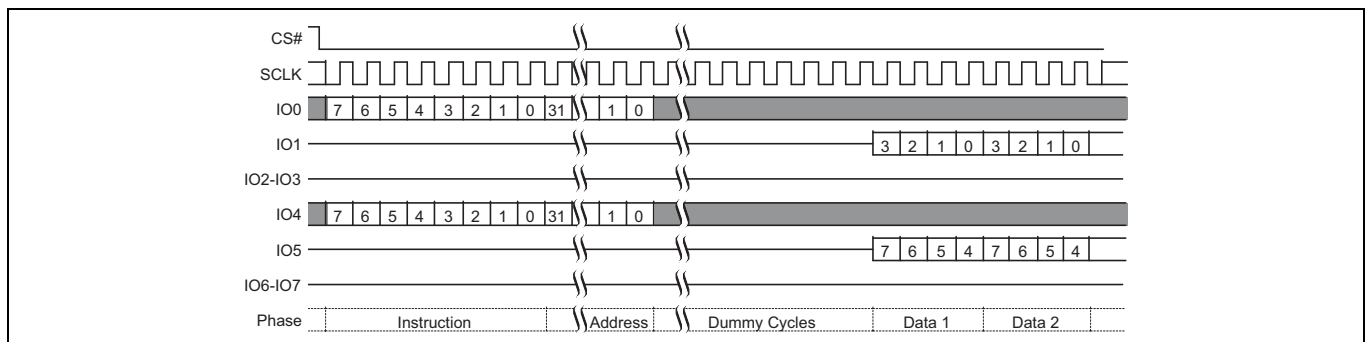
The instruction

- 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 IO1 and IO5 is “don’t care” and may be high impedance. Then the memory contents, at the address given, are shifted out on IO1 and IO5.

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

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



**Figure 53 Dual-Quad SPI Fast Read (FAST\_READ) command sequence**

Commands

### 10.4.3 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 eight bits at a time through IO0–IO7. 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 104 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–IO7 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](#)).

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.

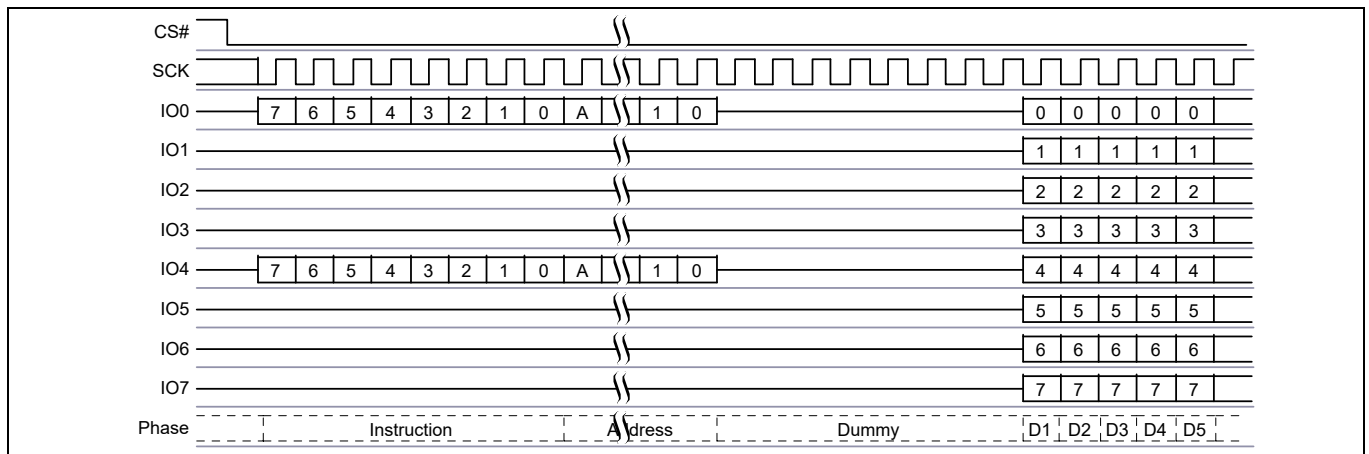


Figure 54 Dual-Quad, Quad Output Read (QOR 6Bh or 4QOR 6Ch) command sequence<sup>[39]</sup>

**Note**

39.A = MSb of address = 23 for command 6Bh, or 31 for command 6Ch.

#### 10.4.4 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 eight I/O signals — IO0–IO7. It is similar to the Quad Output Read command but allows input of the address bits eight bits per serial SCK clock. In some applications, the reduced instruction overhead might allow for code execution (XIP) directly from the S25FL-S device.

The maximum operating clock frequency for Quad I/O Read is 104 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–IO7. 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–IO7 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](#)). 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 eight bits at a time through IO0–IO7. Each byte (8 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 55](#) or [Figure 56](#)). 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 55](#); 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–IO7 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–IO7 signal contention, for the host system to turn off the IO0–IO7 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.



Commands

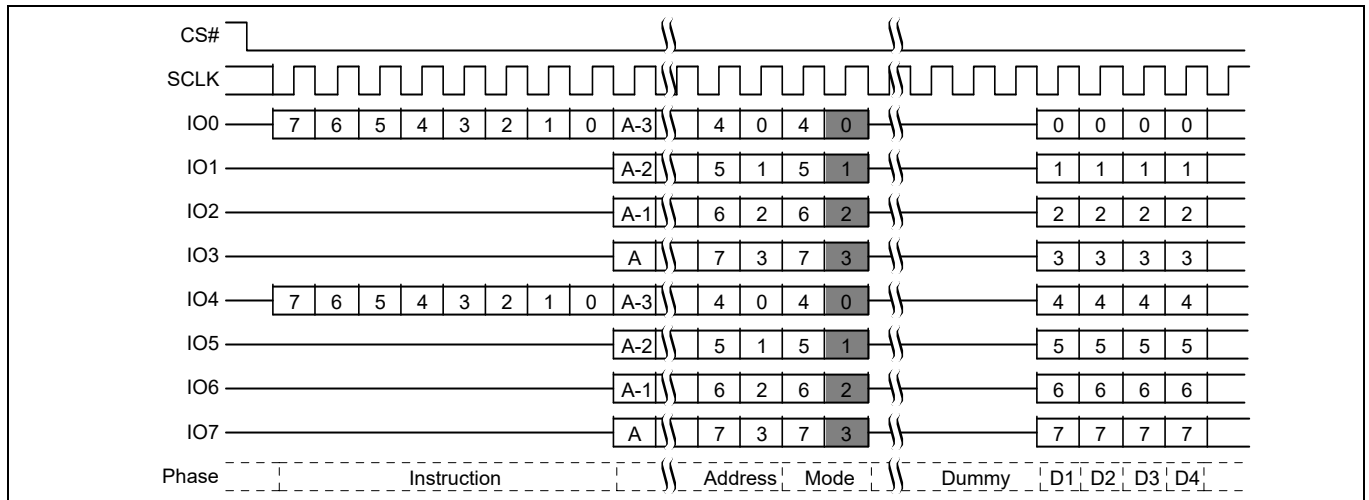


Figure 55 Dual-Quad I/O Read command sequence (3-Byte Address, EBh [ExtAdd = 0], LC = 00b)<sup>[40]</sup>

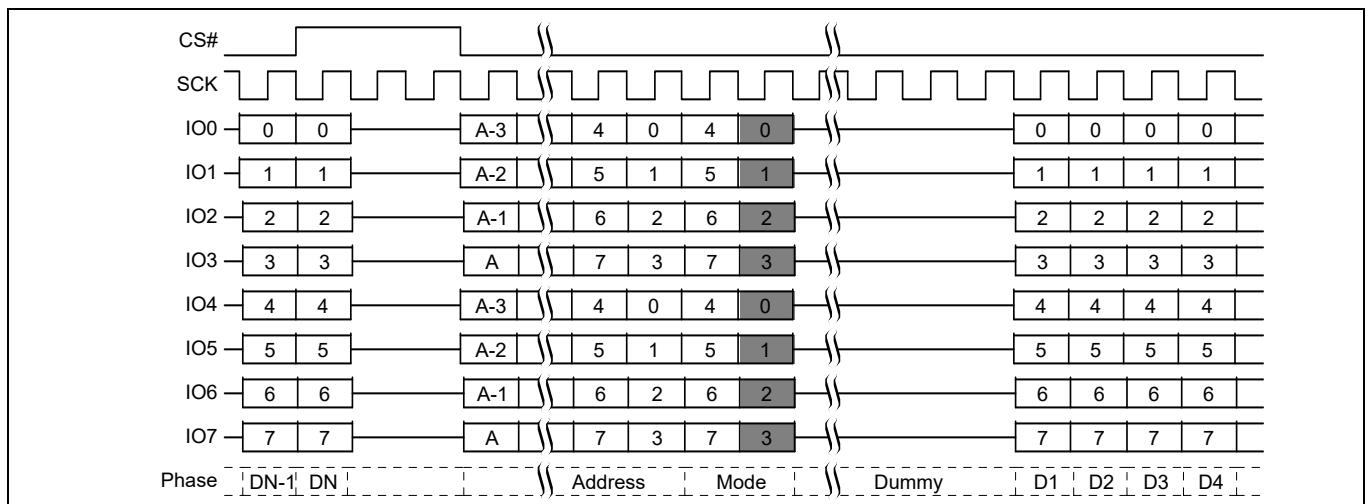


Figure 56 Dual-Quad Continuous Quad I/O Read command sequence (3-Byte Address), LC = 00b<sup>[40]</sup>

Note

40.A = MSb of address = 23 for command EBh, or 31 for command 6Ch.

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

The Read DDR Quad I/O command is similar to the Quad I/O Read command but allows input of the address four bits on every edge of the clock. In some applications, the reduced instruction overhead might allow for code execution (XIP) directly from the S25FL-S devices. The QUAD bit of the Configuration Register is set (CR[1] = 1) to enable the Quad capability in the S25FL-S device.

The instruction

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

The address is followed by mode bits. Then the memory contents, at the address given, is shifted out, in a DDR fashion, with four bits at a time on each clock edge through IO0–IO7.

The maximum operating clock frequency for Read DDR Quad I/O command is 80 MHz.

For Read DDR Quad I/O, there is a latency required after the last address and mode bits are shifted into the IO0–IO7 signals before data begins shifting out of IO0–IO7. This latency period (dummy cycles) allows the device's internal circuitry enough time to access the initial address. During these latency cycles, the data value on IO0–IO7 are "don't care" and may be high impedance. When the DLP is enabled the host system must not drive the IO signals during the dummy cycles. The IO signals must be left high impedance by the host so that the memory device can drive the DLP during the dummy cycles.

The number of dummy cycles is determined by the frequency of SCK. The number of dummy cycles is set by the LC bits in the Configuration Register (CR1).

Both latency tables provide cycles for mode bits so a series of Quad I/O DDR commands may eliminate the 8-bit instruction after the first command sends a complementary mode bit pattern, as shown in [Figure 57](#). This feature removes the need for the eight bit SDR instruction sequence and dramatically reduces initial access times (improves XIP performance). The Mode bits control the length of the next Read DDR Quad I/O operation through the inclusion or exclusion of the first byte instruction code. If the upper nibble (IO[7:4]) and lower nibble (IO[3:0]) of the Mode bits are complementary (i.e. 5h and Ah) the device transitions to Continuous Read DDR Quad I/O Mode and the next address can be entered (after CS# is raised HIGH and then asserted LOW) without requiring the EDh or EEh instruction, as shown in [Figure 58](#) thus, eliminating eight cycles from the command sequence. The following sequences will release the device from Continuous Read DDR Quad I/O mode; after which, the device can accept standard SPI commands:

1. During the Read DDR Quad I/O command sequence, if the Mode bits are not complementary the next time CS# is raised HIGH and then asserted LOW the device will be released from Read DDR Quad I/O mode.
2. During any operation, if CS# toggles HIGH to LOW to HIGH for eight cycles (or less) and data input (IO0–IO7) are not set for a valid instruction sequence, then the device will be released from Read DDR Quad I/O mode.

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. The HOLD function is not valid during Quad I/O DDR commands.

Note that the memory devices drive the IOs with a preamble prior to the first data value. The preamble is a pattern that is used by the host controller to optimize data capture at higher frequencies. The preamble drives the IO bus for the four clock cycles immediately before data is output. The host must be sure to stop driving the IO bus prior to the time that the memory starts outputting the preamble.

Commands

The preamble is intended to give the host controller an indication about the round trip time from when the host drives a clock edge to when the corresponding data value returns from the memory device. The host controller will skew the data capture point during the preamble period to optimize timing margins and then use the same skew time to capture the data during the rest of the read operation. The optimized capture point will be determined during the preamble period of every read operation. This optimization strategy is intended to compensate for both the PVT (process, voltage, temperature) of both the memory device and the host controller as well as any system level delays caused by flight time on the PCB. Although the data learning pattern (DLP) is programmable, the following example shows example of the DLP of 34h. The DLP 34h (or 00110100) will be driven on each of the active outputs (i.e. all eight IOs). This pattern was chosen to cover both DC and AC data transition scenarios. The two DC transition scenarios include data low for a long period of time (two half clocks) followed by a high going transition (001) and the complementary low going transition (110). The two AC transition scenarios include data low for a short period of time (one half clock) followed by a high going transition (101) and the complementary low going transition (010). The DC transitions will typically occur with a starting point closer to the supply rail than the AC transitions that may not have fully settled to their steady state (DC) levels. In many cases the DC transitions will bound the beginning of the data valid period and the AC transitions will bound the ending of the data valid period. These transitions will allow the host controller to identify the beginning and ending of the valid data eye. Once the data eye has been characterized the optimal data capture point can be chosen. See **“SPI DDR Data Learning Registers”** on page 55 for more details.

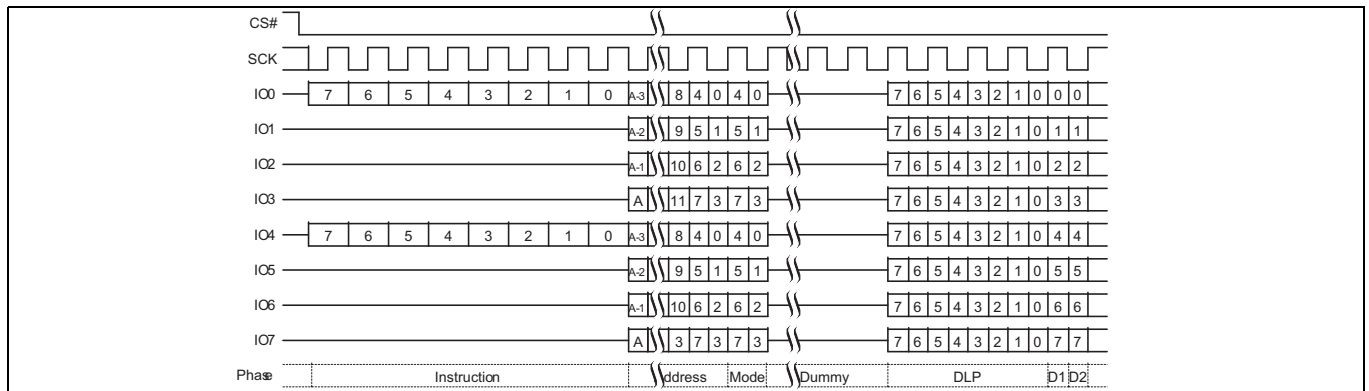


Figure 57 Dual-Quad SPI DDR Quad I/O Read initial access<sup>[41, 42]</sup>

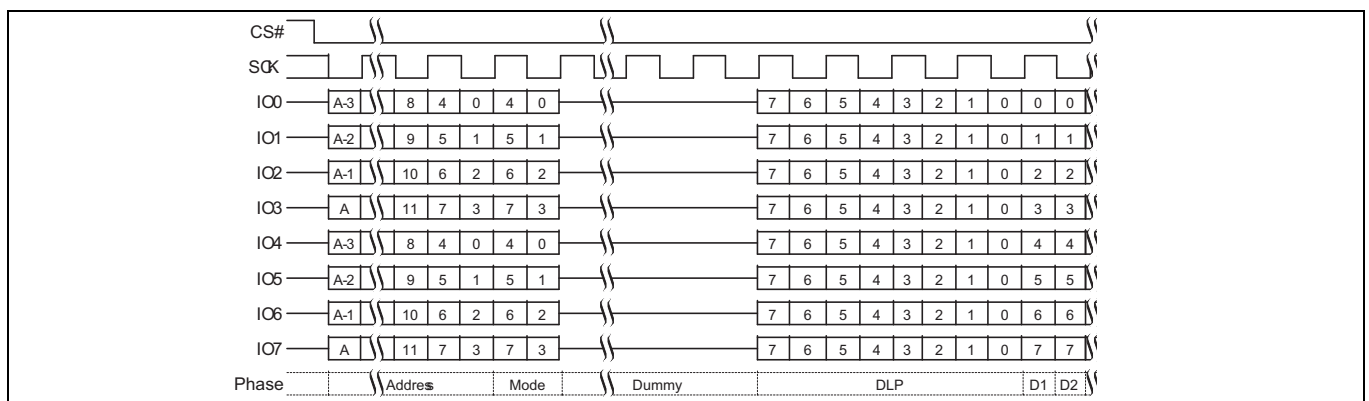


Figure 58 Dual-Quad Continuous DDR Quad I/O Read subsequent access<sup>[41, 42]</sup>

Notes

- 41.A = MSb of address = 23 for command EDh, or 31 for command EEh.
- 42.Example DLP of 34h (or 00110100).

## **10.5 Program Flash Array commands**

### **10.5.1 Program granularity**

#### **10.5.1.1 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 **“ECC Status Register (ECCSR)”** on page 52.) 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).

### **10.5.1.2 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 512 bytes or 1024 bytes) to be programmed in one operation. 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 pages of 1024 bytes aligned on 1024-byte boundaries with each page being programmed only once.

### **10.5.1.3 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.

### 10.5.2 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

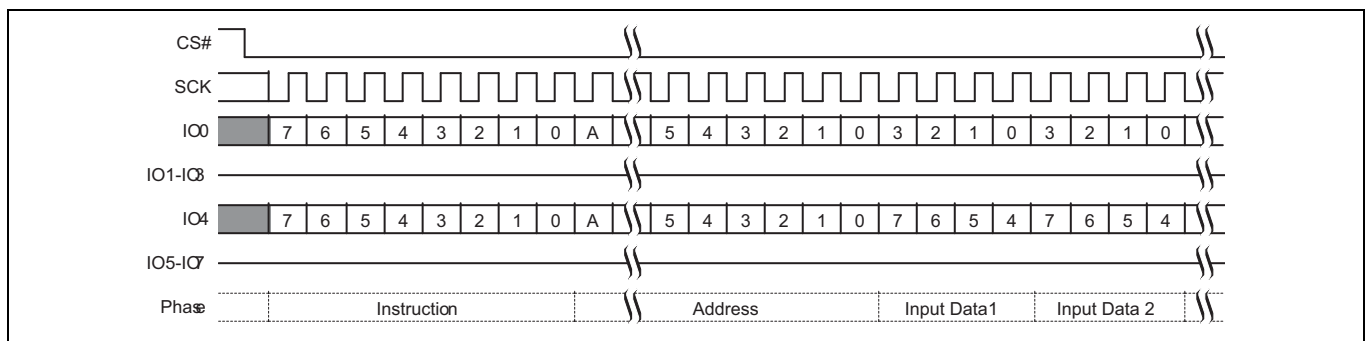
- 02h (ExtAdd = 0) is followed by a 3-byte address (A23–A0) or
- 02h (ExtAdd = 1) is followed by a 4-byte address (A31–A0) or
- 12h is followed by a 4-byte address (A31–A0)

and at least one data byte on IO0 and IO4. Up to a page can be provided on IO0 and IO4 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 zero, 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 zero) 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 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.

For optimized timings, using the Page Program (PP) command to load the entire page size program buffer within the page boundary will save overall programming time versus loading less than a page size 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.



**Figure 59 Dual-Quad Page Program (PP 02h or 4PP 12h) command sequence**

### 10.5.3 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 512 bytes or 1024 bytes) of data to be loaded into the page buffer using eight signals: IO0–IO7. QPP can improve performance for PROM programmer and applications that have slower clock speeds (< 12 MHz) by loading 8 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 two data bytes, 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 [Figure 60](#).

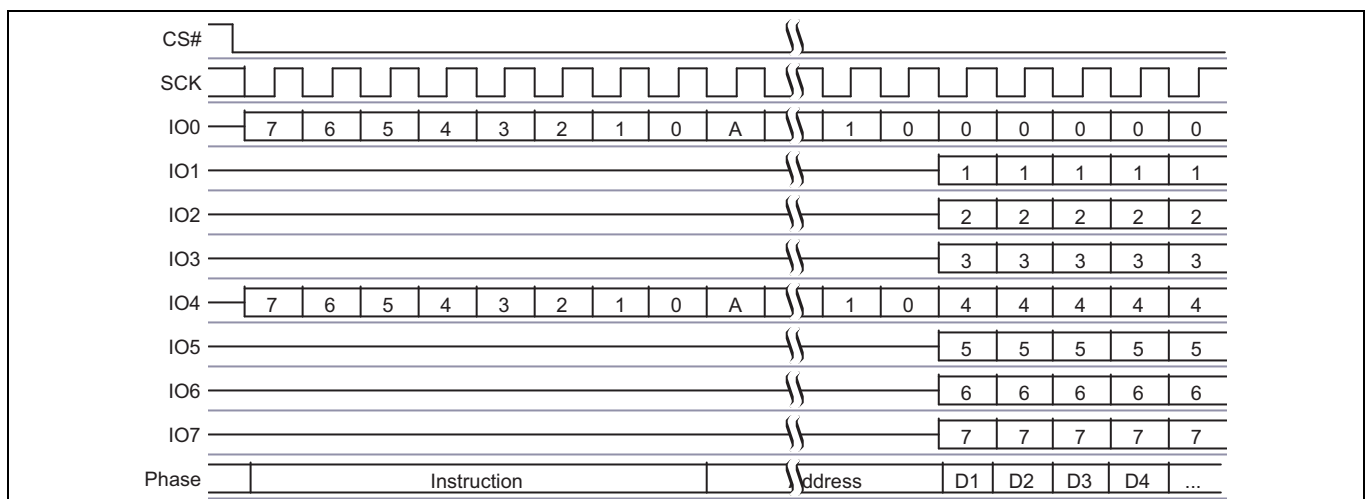


Figure 60 Dual-Quad, Quad Page Program command sequence<sup>[43]</sup>

**Note**

43.A = MSb of address = A23 for PP 02h, or A31 for PP 02h, or for 4PP 12h.

### 10.5.4 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 46](#).

See [Table 44](#) 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 '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 46](#).

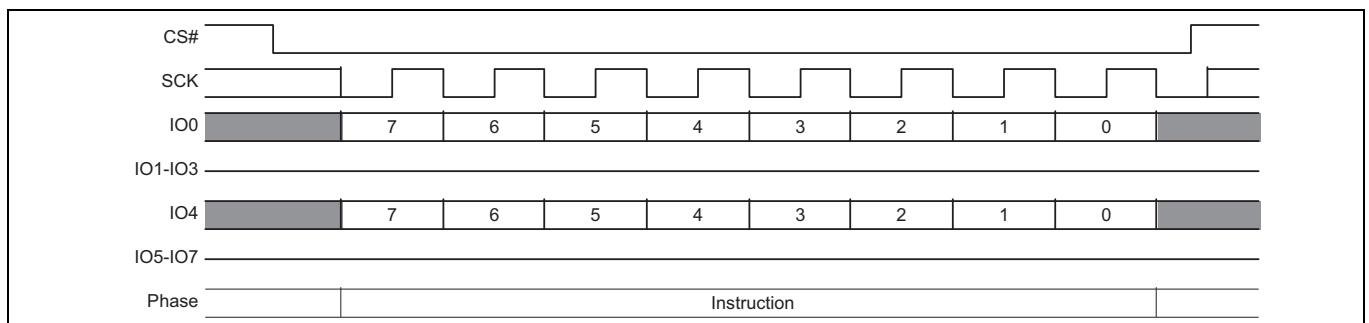


Figure 61 Dual-Quad Program Suspend command sequence

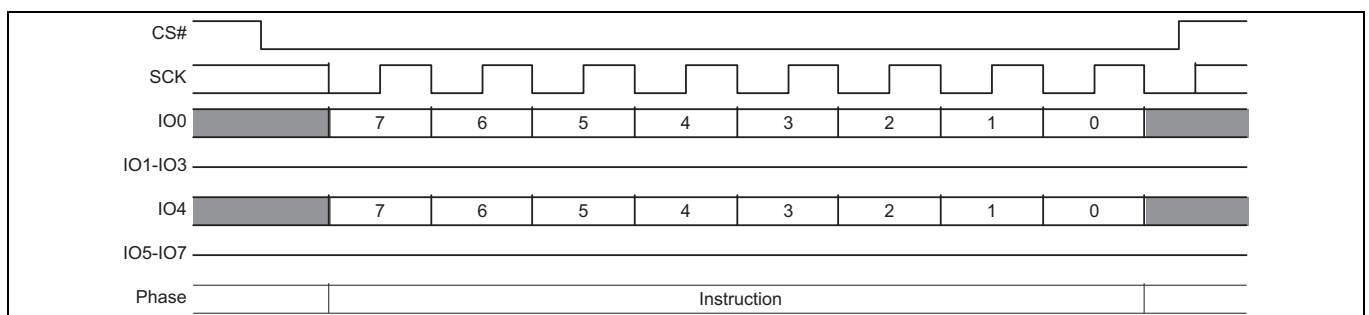


Figure 62 Dual-Quad Program Resume command sequence



## 10.6 Erase Flash Array commands

### 10.6.1 Parameter 8-KB Sector Erase (P4E 20h or 4P4E 21h)

The P4E command is implemented only in S79FL256S and S79FL512S. The P4E command is ignored when the device is configured with the 256-KB sector option.

The Parameter 8-KB Sector Erase (P4E) command sets all the bits of a 8-KB 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 '1', when the erase cycle is in progress and '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 8 KB will not be executed and will not set the E\_ERR status.

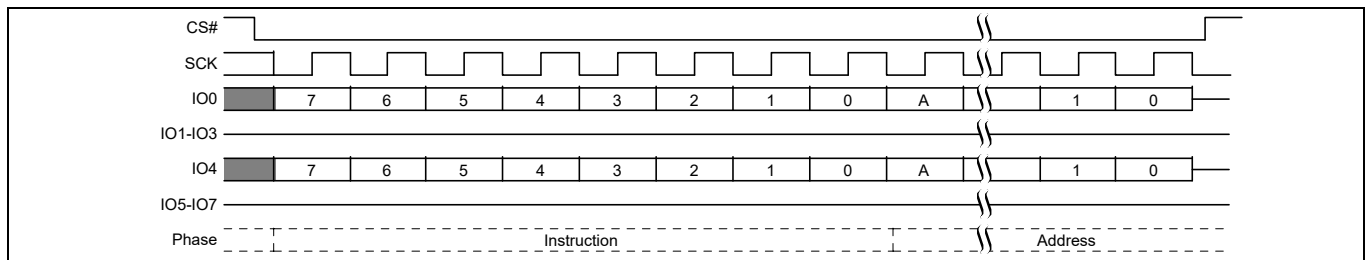


Figure 63 Dual-Quad Parameter Sector Erase command sequence (P4E 20h or 4P4E 21h)<sup>[44]</sup>

#### Note

44.A = MSb of address = A23 for P4E 20h, or A31 for 4P4E 21h.

### 10.6.2 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)

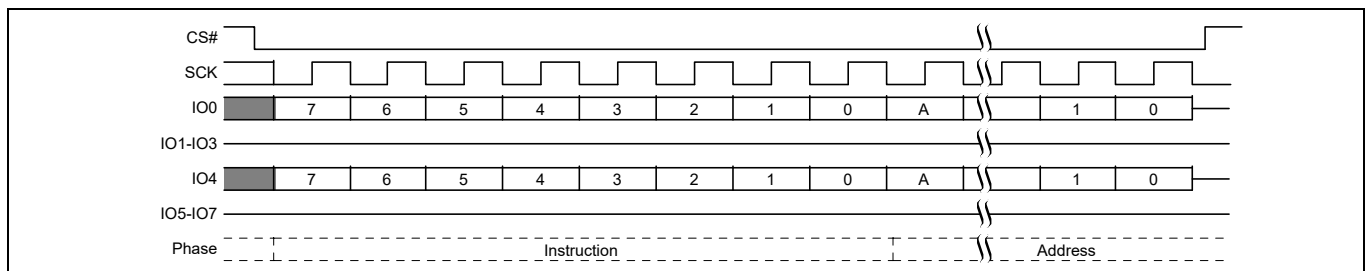
CS# must be driven into the logic HIGH state after the twenty-fourth or thirty-second bit of address has been latched in on IO0 and IO4. This will initiate the erase cycle, which involves the pre-programming and erase of the chosen sector. If CS# is not driven HIGH after the last bit of address, the 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 '1', when the erase cycle is in progress and '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 ordering option determines whether the SE command erases 128 KB or 512 KB. The option to use this command to always erase 512 KB provides for software compatibility with higher density and future S79FL family devices.

ASP has a PPB and a DYB protection bit for each sector. If a sector erase command is applied to a 128-KB range that includes a protected 8-KB sector, or to a 512-KB range that includes a 128-KB protected address range, the erase will not be executed on the range and will set the E\_ERR status.



**Figure 64 Dual-Quad Sector Erase (SE 20h or 4SE 21h) command sequence<sup>[45]</sup>**

**Note**

45.A = MSb of address = A23 for SE D8h, or A31 for 4SE DCh.

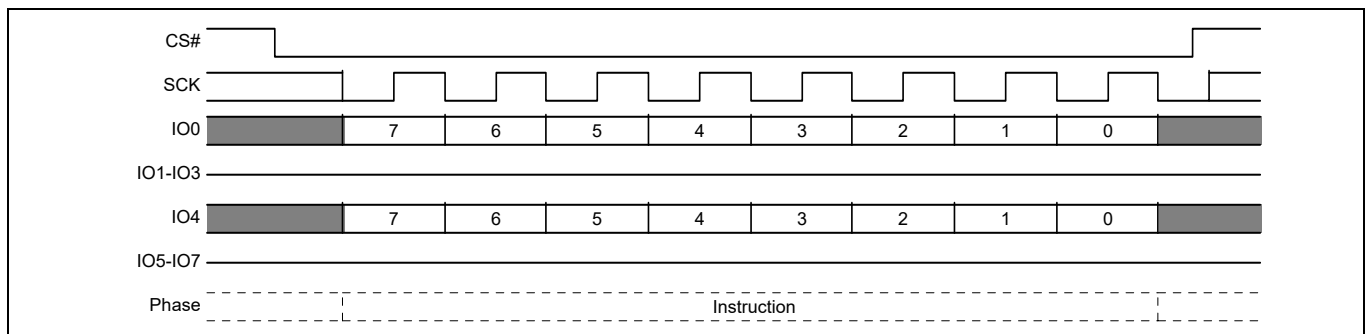
### 10.6.3 Bulk Erase (BE 60h or C7h)

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 IO0 and IO4. 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 '1', when the erase cycle is in progress and '0', when the erase cycle has been completed.

A BE command can be executed only when the Block Protection (BP2, BP1, BP0) bits are set to 0's. If the BP bits are not zero, the BE command is not executed and E\_ERR is not set. The BE command will skip any sectors protected by the DYB or PPB and the E\_ERR status will not be set.



**Figure 65 Bulk Erase command sequence**

#### **10.6.4 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 47](#).

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 44](#) 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.

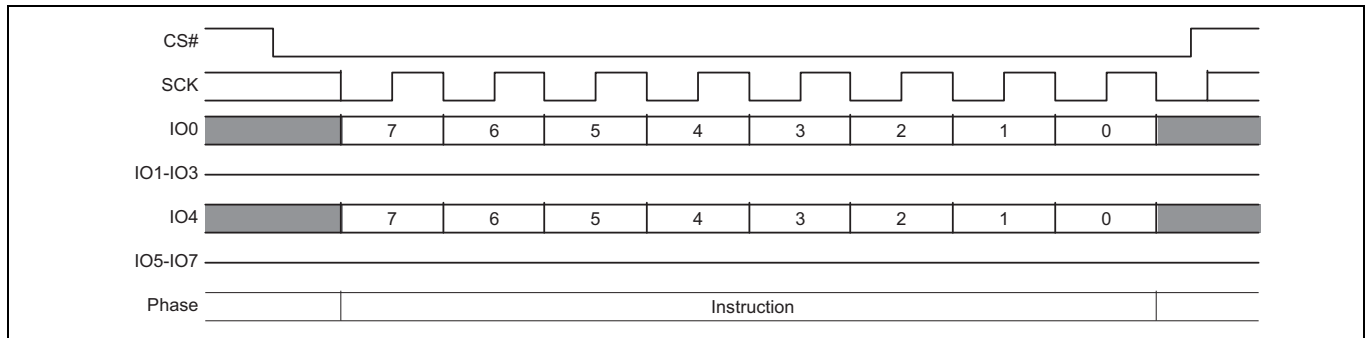
After an erase-suspended program operation is complete, the device returns to the erase-suspend mode. The system can determine the status of the program operation by reading the WIP bit in the Status Register, just as in the standard program operation.

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

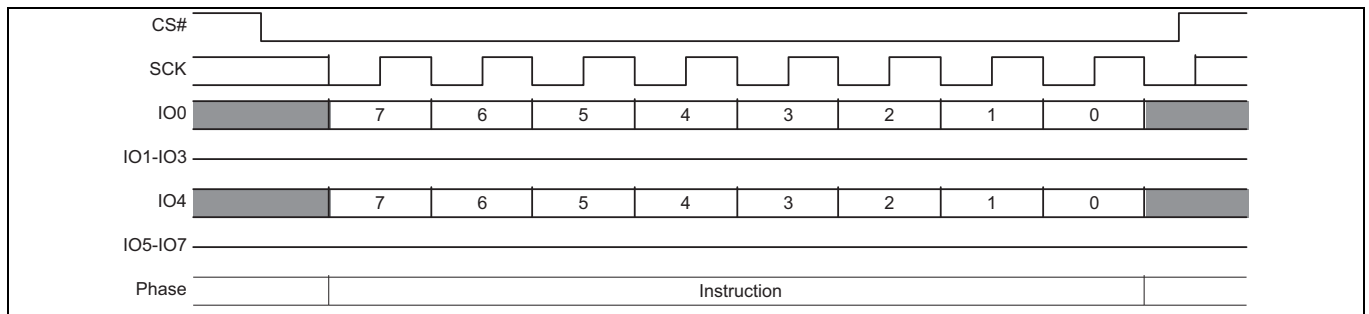
After an Erase Resume command is sent, the WIP bit in the status register will be set to '1' and the erase operation will continue. Further Resume commands are ignored.

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 47](#).

Commands



**Figure 66 Dual-Quad Erase Suspend command sequence**



**Figure 67 Dual-Quad Erase Resume command sequence**

**Table 44 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.

Commands

**Table 44** Commands allowed during Program or Erase Suspend (Continued)

Instruction name	Instruction code (Hex)	Allowed during Erase Suspend	Allowed during Program Suspend	Comment
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	
DDRQIOR	ED	X	X	All array reads allowed in Suspend.
DDRQIOR4	EE	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.

## 10.7 One Time Program Array commands

### 10.7.1 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 2048 bytes so, the address bits from A25 to A10 must be zero for this command. Refer to **“OTP address space”** on page 41 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. To program the OTP array in bit granularity, the rest of the bits within a data byte can be set to ‘1’. Each region in the OTP memory space can be programmed one or more times, provided that the region is not locked. Attempting to program zeros in a region that is locked will fail with the P\_ERR bit in 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). The protocol of the OTP Program command is the same as the Page Program command. See **“Page Program (PP 02h or 4PP 12h)”** on page 94 for the command sequence.

### 10.7.2 OTP Read (OTPR 4Bh)

The OTP Read command reads data from the OTP region. The OTP region is 2048 bytes so, the address bits from A25 to A10 must be zero for this command. Refer to **“OTP address space”** on page 41 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.

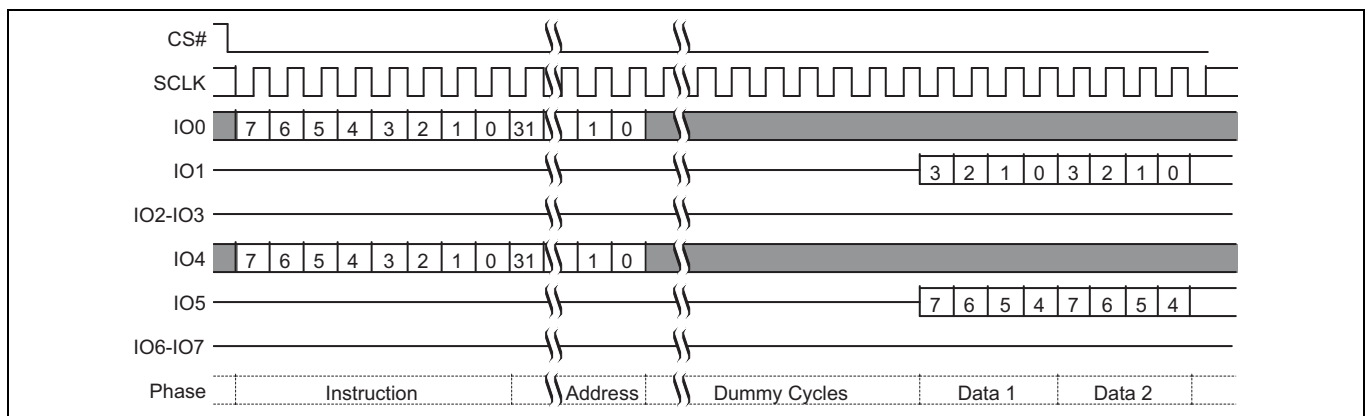


Figure 68 Read OTP (OTPR 4Bh) command sequence

## 10.8 Advanced Sector Protection commands

### 10.8.1 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 is shifted out on the serial output SO, least significant byte first. Each bit is shifted out at the SCK frequency by the falling edge of the SCK signal. It is possible to read the ASP Register continuously by providing multiples of 16 clock cycles. The maximum operating clock frequency for the ASP Read (ASPRD) command is 133 MHz.

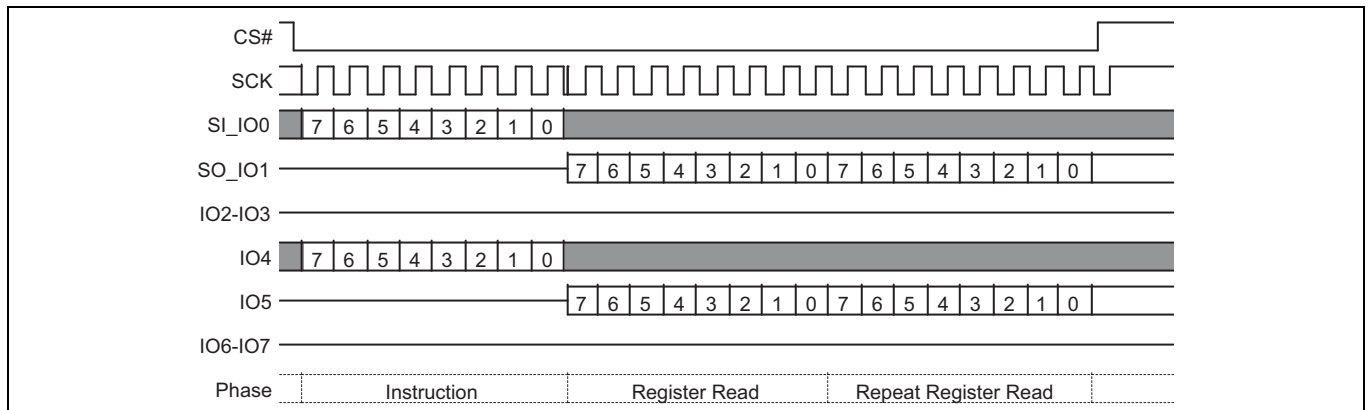


Figure 69 Dual-Quad SPI ASPRD command sequence

### 10.8.2 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 '1' during the self-timed ASPP operation, and is '0' when it is completed. When the ASPP operation is completed, the Write Enable Latch (WEL) is set to '0'.

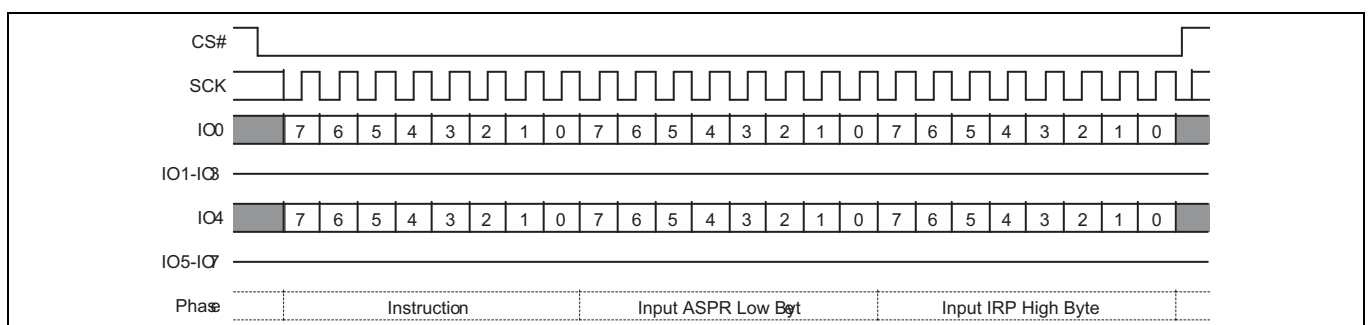


Figure 70 ASPP (2Fh) command



### 10.8.3 DYB Read (DYBRD E0h)

The instruction E0h is latched into SI by the rising edge 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 zero). 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 133 MHz.

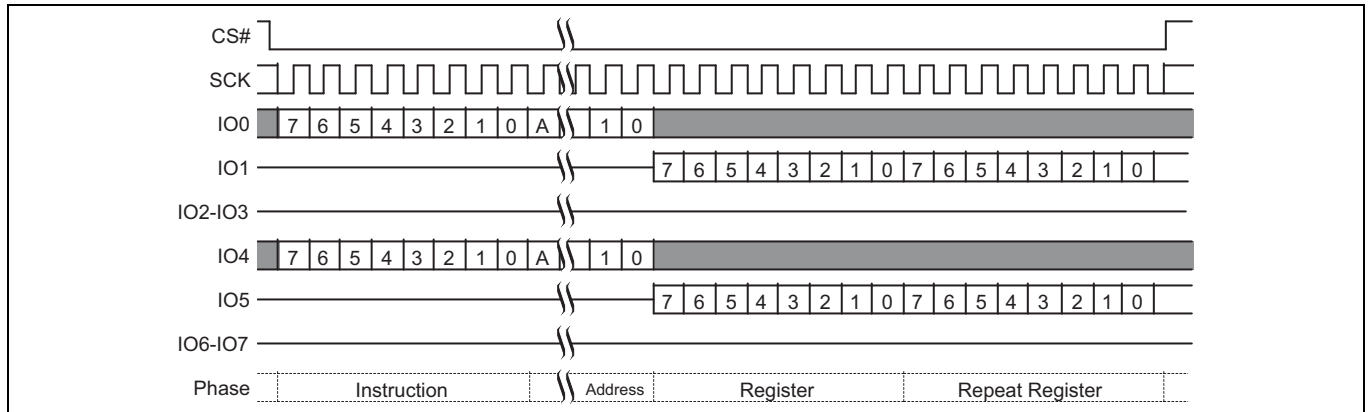


Figure 71 DYBRD command sequence

### 10.8.4 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 '1' during the self-timed DYBWR operation, and is '0' when it is completed. When the DYBWR operation is completed, the Write Enable Latch (WEL) is set to '0'.

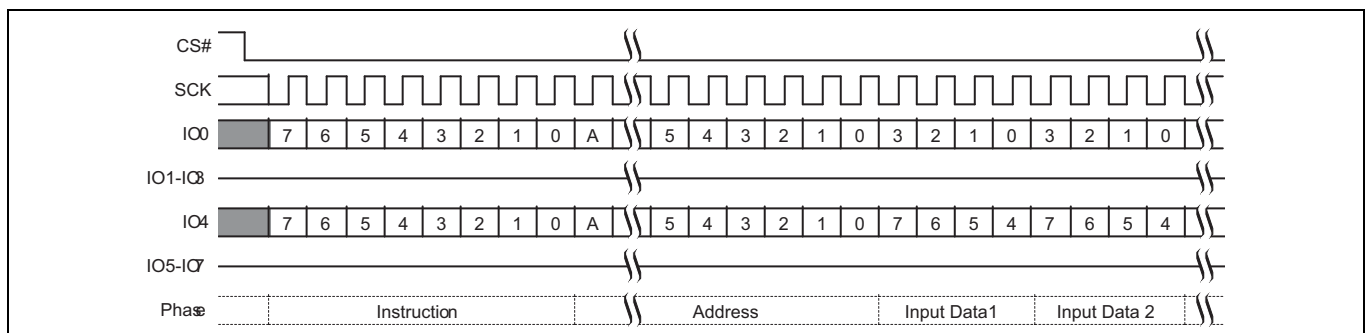


Figure 72 DYBWR (E1h) command sequence

Commands

### 10.8.5 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 zero) 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 133 MHz.

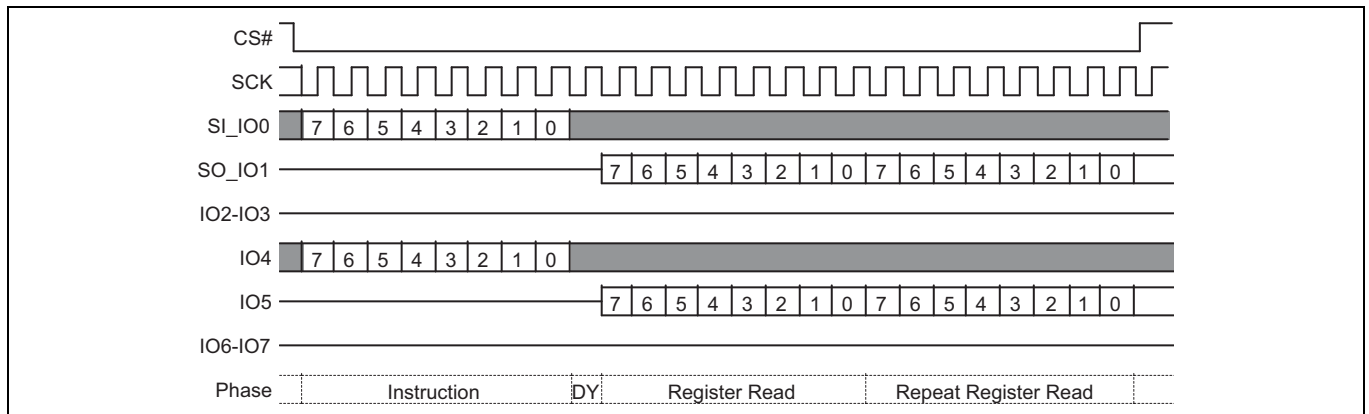


Figure 73 PPBRD (E2h) command sequence

### 10.8.6 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 zero).

The PPBP command affects the P\_ERR and WIP bits of the Status and Configuration Registers in the same manner as any other programming operation.

CS# must be driven to the logic HIGH state after the last bit of address has been latched in. If not, the PPBP command is not executed. As soon as CS# is driven to the logic HIGH state, the self-timed PPBP operation is initiated. While the PPBP operation is in progress, the Status Register may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is '1' during the self-timed PPBP operation, and is '0' when it is completed. When the PPBP operation is completed, the Write Enable Latch (WEL) is set to '0'.

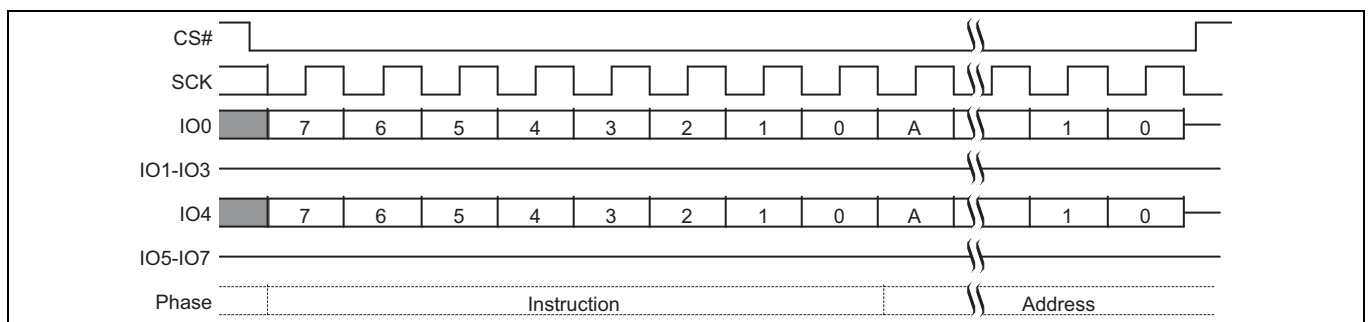


Figure 74 PPBP (E3h) command sequence

Commands

### 10.8.7 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 '1', when the erase cycle is in progress and '0', when the erase cycle has been completed. Erase suspend is not allowed during PPB Erase.

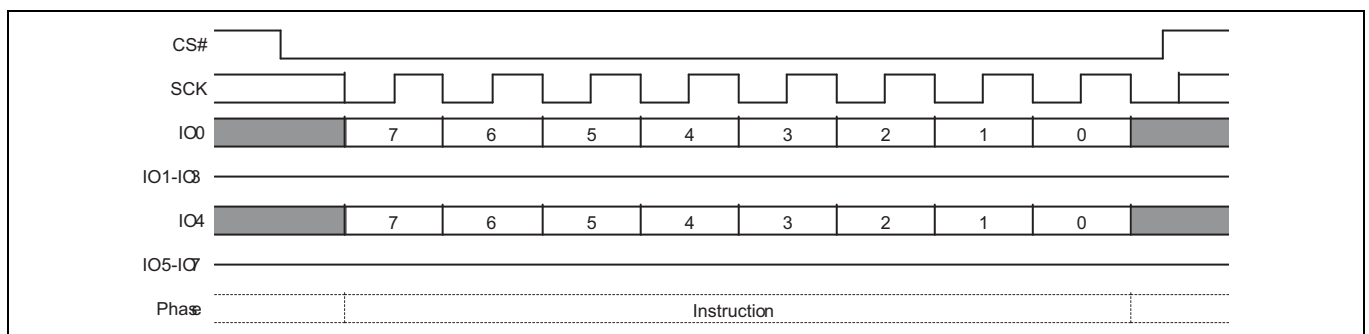


Figure 75 PPB Erase (PPBE E4h) command sequence

### 10.8.8 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.

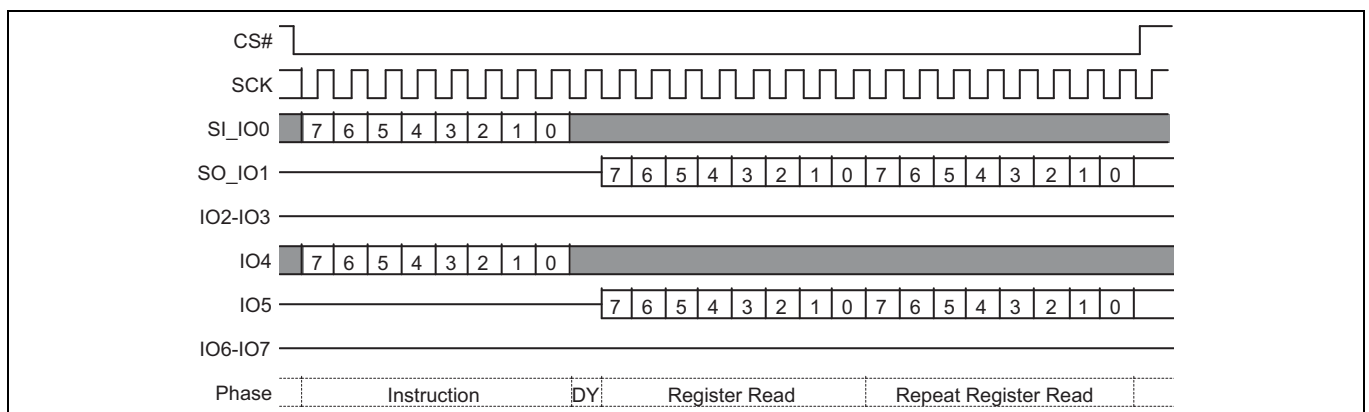


Figure 76 PPB Lock Register Read command sequence

Commands

### 10.8.9 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 '1' during the self-timed PLBWR operation, and is '0' when it is completed. When the PLBWR operation is completed, the Write Enable Latch (WEL) is set to '0'. The maximum clock frequency for the PLBWR command is 133 MHz.

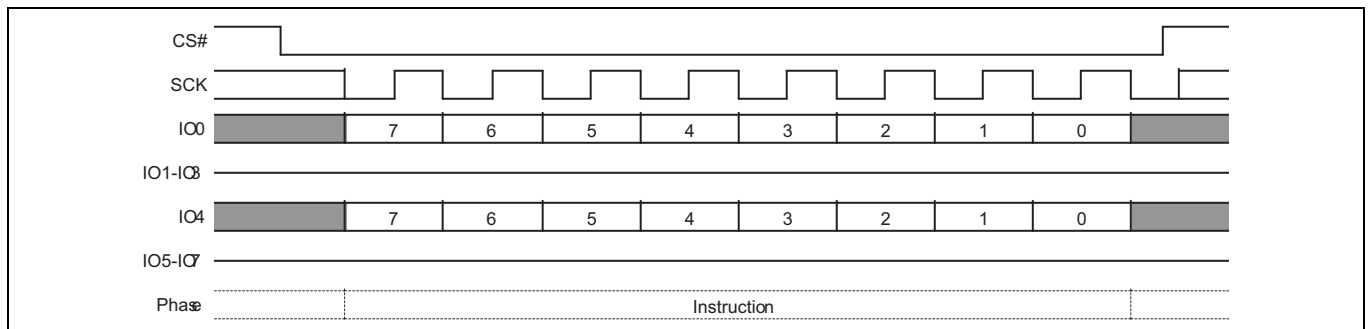


Figure 77 PPB Lock Bit Write (PLBWR A6h) command sequence

### 10.8.10 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 133 MHz.

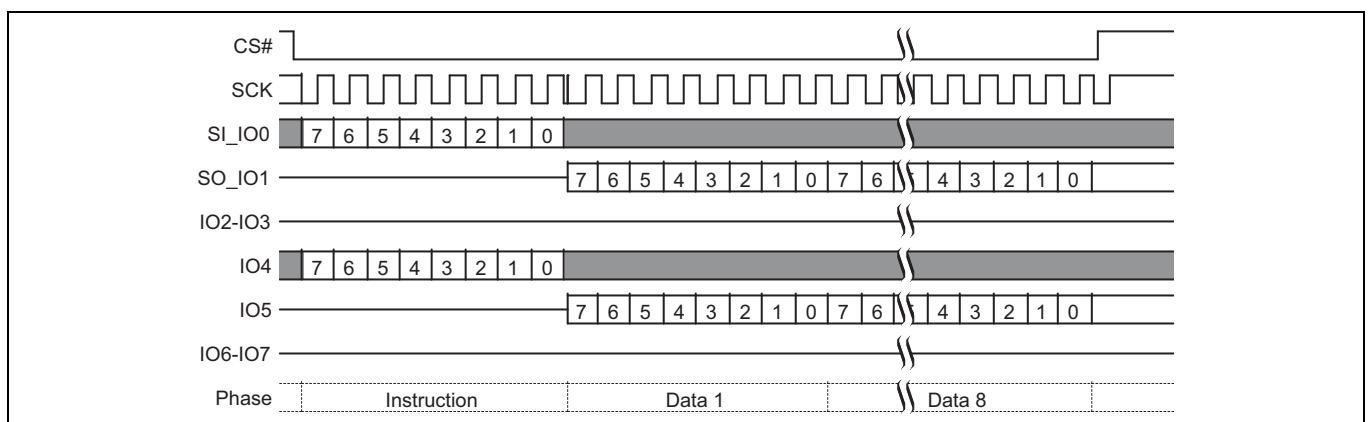


Figure 78 Password Read (PASSRD E7h) command sequence

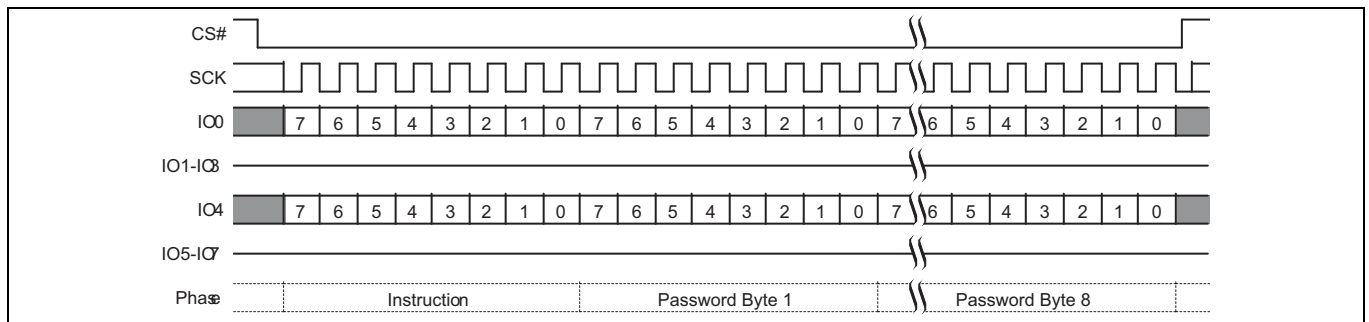
### 10.8.11 Password Program (PASSP E8h)

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<sup>th</sup>) 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 '1' during the self-timed PASSP cycle, and is '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 '0'. The maximum clock frequency for the PASSP command is 133 MHz.



**Figure 79 Password Program (PASSP E8h) command sequence**

### 10.8.12 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<sup>th</sup>) 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 '1' during the self-timed PASSU cycle, and is '0' when it is completed.

If the PASSU command supplied password does not match the hidden password in the Password Register, an error is reported by setting the P\_ERR bit to '1'. The WIP bit of the Status Register also remains set to '1'. It is necessary to use the CLSR command to clear the Status Register, the RESET command to software reset the device, or drive the RESET# input LOW to initiate a hardware reset, in order to return the P\_ERR and WIP bits to '0'. This returns the device to standby state, ready for new commands such as a retry of the PASSU command.

If the password does match, the PPB Lock bit is set to '1'. The maximum clock frequency for the PASSU command is 133 MHz.

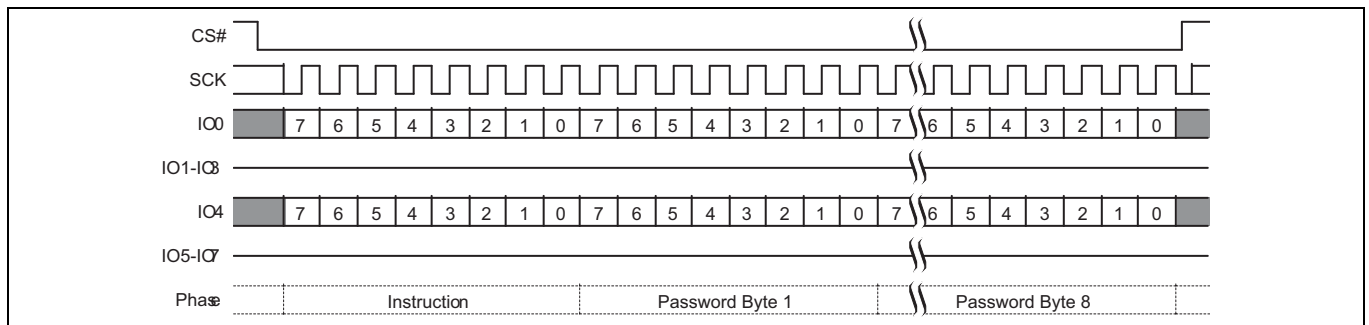


Figure 80 Password Unlock (PASSU E9h) command sequence

## 10.9 Reset commands

### 10.9.1 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.

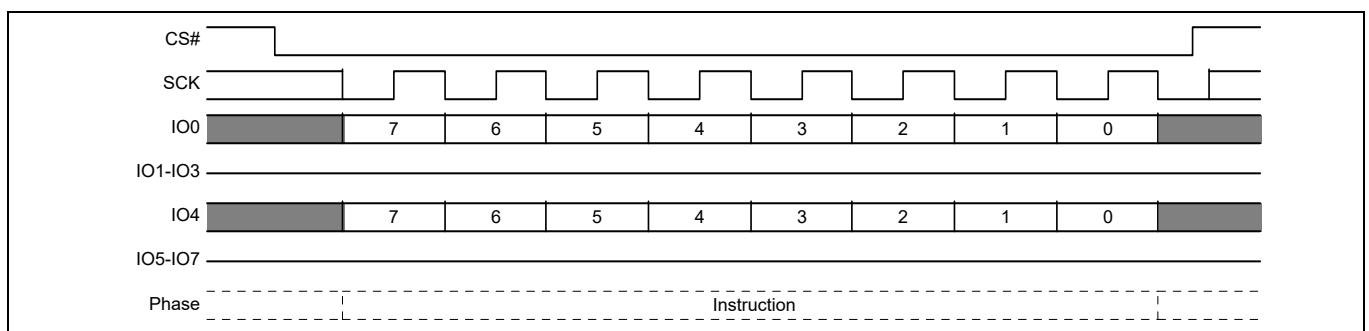


Figure 81 Dual-Quad Software Reset (RESET F0h) command sequence

### 10.9.2 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 1's on IO0 and IO4 for 8 SCK cycles. IO1–IO3 and IO5–IO7 are “don't care” during these cycles.

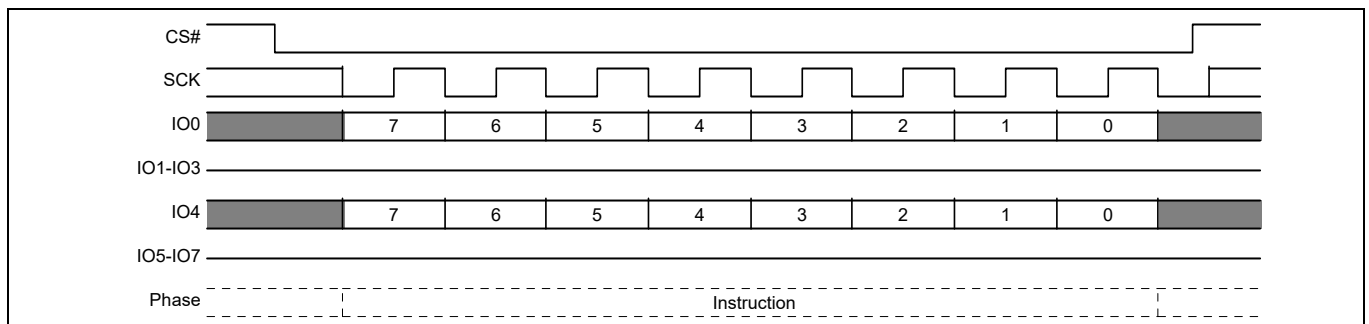


Figure 82 Dual-Quad SPI Mode Bit (MBR FFh) Reset command sequence

## 10.10 Embedded algorithm performance tables

**Table 45 Program and erase performance**

Symbol	Parameter	Min	Typ <sup>[46]</sup>	Max <sup>[47]</sup>	Unit
$t_W$	WRR write time	–	560	2000	ms
$t_{PP}$	Page programming (1024 bytes)	–	340	750 <sup>[48]</sup>	$\mu$ s
	Page programming (512 bytes)	–	250	750	$\mu$ s
$t_{SE}$	Sector erase time (512-KB logical sectors = 4 × 128 KB physical sectors)	–	520	2600	ms
	Sector erase time (128-KB /8-KB physical sectors)	–	130	650 <sup>[49]</sup>	ms
	Sector erase time (128-KB /8-KB top/bottom: logical sector = 16 × 8-KB physical sectors)	–	2080	10400	ms
$t_{BE}$	Bulk erase time (S79FL256S)	–	33	165	sec
	Bulk erase time (S79FL512S)	–	66	330	sec

**Table 46 Program suspend AC parameters**

Parameter	Min	Typical	Max	Unit	Comments
Program Suspend Latency ( $t_{PSL}$ )	–	–	40	$\mu$ s	The time from Program Suspend command until the WIP bit is '0'
Program Resume to next Program Suspend ( $t_{PRS}$ )	0.06	100	–	$\mu$ s	Minimum is the time needed to issue the next Program Suspend command but $\geq$ typical periods are needed for program to progress to completion

**Table 47 Erase suspend AC parameters**

Parameter	Min	Typical	Max	Unit	Comments
Erase Suspend Latency ( $t_{ESL}$ )	–	–	45	$\mu$ s	The time from Erase Suspend command until the WIP bit is '0'
Erase Resume to next Erase Suspend ( $t_{ERS}$ )	0.06	100	–	$\mu$ s	Minimum is the time needed to issue the next Erase Suspend command but $\geq$ typical periods are needed for the erase to progress to completion

### Notes

46. Typical program and erase times assume the following conditions: 25°C,  $V_{CC} = 3.0$  V; random data pattern.  
 47. Under worst case conditions of 90°C; 100,000 cycles max.  
 48. Maximum value also applies to OTPP, PPBP, ASPP, PASSP, ABWR, and PNVDLR programming commands.  
 49. Maximum value also applies to the PPBE erase command.



Data integrity

## 11 Data integrity

### 11.1 Erase endurance

**Table 48 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 <sup>[50]</sup>	100K	PE cycle

### 11.2 Data retention

**Table 49 Data retention**

Parameter	Test conditions	Minimum time	Unit
Data Retention Time	10K program/erase cycles	20	Years
	100K program/erase cycles	2	Years

Contact Infineon Sales and FAE for further information on the data integrity.

#### Note

50.Each write command to a non-volatile register causes a PE cycle on the entire non-volatile register array.

## 12 Software interface reference

### 12.1 Command summary

**Table 50 S79FL256S/S79FL512S instruction set (sorted by instruction)**

<b>Instruction (Hex)</b>	<b>Command name</b>	<b>Command description</b>	<b>Maximum frequency (MHz)</b>
01	WRR	Write Register (Status-1, Configuration-1)	133
02	PP	Page Program (3- or 4-byte address)	133
03	READ	Read (3- or 4-byte address)	50
04	WRDI	Write Disable	133
05	RDSR1	Read Status Register-1	133
06	WREN	Write Enable	133
07	RDSR2	Read Status Register-2	133
0B	FAST_READ	Fast Read (3- or 4-byte address)	133
0C	4FAST_READ	Fast Read (4-byte address)	133
12	4PP	Page Program (4-byte address)	133
13	4READ	Read (4-byte address)	50
14	ABRD	AutoBoot Register Read	133
15	ABWR	AutoBoot Register Write	133
16	BRRD	Bank Register Read	133
17	BRWR	Bank Register Write	133
18	ECCRD	ECC Read	133
20	P4E	Parameter 8-KB sector Erase (3- or 4-byte address)	133
21	4P4E	Parameter 8-KB sector Erase (4-byte address)	133
2B	ASPRD	ASP Read	133
2F	ASPP	ASP Program	133
30	CLSR	Clear Status Register - Erase/Program Fail Reset	133
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	133
38	QPP	Quad Page Program (3- or 4-byte address)	80
41	DLPRD	Data Learning Pattern Read	133
42	OTPP	OTP Program	133
43	PNVDLR	Program NV Data Learning Register	133
4A	WVDLR	Write Volatile Data Learning Register	133
4B	OTPR	OTP Read	133
60	BE	Bulk Erase	133
6B	QOR	Read Quad Out (3- or 4-byte address)	104
6C	4QOR	Read Quad Out (4-byte address)	104
75	ERSP	Erase Suspend	133

**Table 50 S79FL256S/S79FL512S instruction set (sorted by instruction) (Continued)**

<b>Instruction (Hex)</b>	<b>Command name</b>	<b>Command description</b>	<b>Maximum frequency (MHz)</b>
7A	ERRS	Erase Resume	133
85	PGSP	Program Suspend	133
8A	PGRS	Program Resume	133
90	READ_ID (REMS)	Read Electronic Manufacturer Signature	133
9F	RDID	Read ID (JEDEC Manufacturer ID and JEDEC CFI)	133
A3	MPM	Reserved for Multi-I/O-High Perf Mode (MPM)	133
A6	PLBWR	PPB Lock Bit Write	133
A7	PLBRD	PPB Lock Bit Read	133
AB	RES	Read Electronic Signature	50
B9	BRAC	Bank Register Access (Legacy Command formerly used for Deep Power Down)	133
C7	BE	Bulk Erase (alternate command)	133
D8	SE	Erase 128 KB or 512 KB (3- or 4-byte address)	133
DC	4SE	Erase 128 KB or 512 KB (4-byte address)	133
E0	DYBRD	DYB Read	133
E1	DYBWR	DYB Write	133
E2	PPBRD	PPB Read	133
E3	PPBP	PPB Program	133
E4	PPBE	PPB Erase	133
E5	Reserved-E5	Reserved	
E6	Reserved-E6	Reserved	
E7	PASSRD	Password Read	133
E8	PASSP	Password Program	133
E9	PASSU	Password Unlock	133
EB	QIOR	Quad I/O Read (3- or 4-byte address)	104
EC	4QIOR	Quad I/O Read (4-byte address)	104
ED	DDRQIOR	DDR Quad I/O Read (3- or 4-byte address)	80
EE	4DDRQIOR	DDR Quad I/O Read (4-byte address)	80
F0	RESET	Software Reset	133
FF	MBR	Mode Bit Reset	133

## 12.2 Device ID and common flash interface (ID-CFI) address map

### 12.2.1 Field definitions

**Table 51** Manufacturer and Device ID

Byte address	Data	Description
00h	01h	Manufacturer ID for Infineon
01h	79h	Device ID Most Significant Byte – Memory Interface Type
02h	19h (256 Mb) 20h (512 Mb)	Device ID Least Significant Byte – Density
03h	xxh	ID-CFI Length - number bytes following. Adding this value to the current location of 03h gives the address of the last valid location in the ID-CFI address map. A value of 00h indicates the entire 512-byte ID-CFI space must be read because the actual length of the ID-CFI information is longer than can be indicated by this legacy single byte field. The value is OPN dependent.
04h	00h (Uniform 512-KB sectors) 01h (8-KB parameter sectors with uniform 128-KB sectors)	Sector architecture
05h	80h (FL-S Family)	Family ID
06h	xxh	ASCII characters for Model Refer to <b>“Ordering information”</b> on page 130 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 52 CFI query identification string**

Byte address	Data	Description
10h 11h 12h	51h 52h 59h	Query Unique ASCII string "QRY"
13h 14h	02h 00h	Primary OEM command set FL-P backward compatible command set ID
15h 16h	40h 00h	Address for primary extended table
17h 18h	53h 46h	Alternate OEM command set ASCII characters "FS" for SPI (F) interface, S technology
19h 1Ah	51h 00h	Address for alternate OEM extended table

**Table 53 CFI system interface string**

Byte address	Data	Description
1Bh	27h	V <sub>CC</sub> Min. (erase/program): 100 millivolts
1Ch	36h	V <sub>CC</sub> Max. (erase/program): 100 millivolts
1Dh	00h	V <sub>PP</sub> Min. voltage (00h = no V <sub>PP</sub> present)
1Eh	00h	V <sub>PP</sub> Max. voltage (00h = no V <sub>PP</sub> present)
1Fh	06h	Typical timeout per single byte program 2 <sup>N</sup> μs
20h	08h (512B page) 09h (1024B page)	Typical timeout for Min. size Page program 2 <sup>N</sup> μs (00h = not supported)
21h	08h (8 KB or 128 KB) 09h (512 KB)	Typical timeout per individual sector erase 2 <sup>N</sup> ms
22h	0Fh (256 Mb) 10h (512 Mb)	Typical timeout for full chip erase 2 <sup>N</sup> ms (00h = not supported)
23h	02h	Max. timeout for byte program 2 <sup>N</sup> times typical
24h	02h	Max. timeout for page program 2 <sup>N</sup> times typical
25h	03h	Max. timeout per individual sector erase 2 <sup>N</sup> times typical
26h	03h	Max. timeout for full chip erase 2 <sup>N</sup> times typical (00h = not supported)

**Table 54 Device geometry definition for 256-Mb and 512-Mb bottom boot initial delivery state**

Byte address	Data	Description
27h	19h (256 Mb) 1Ah (512 Mb)	Device size = 2 <sup>N</sup> bytes;
28h	03h	Flash device interface description;
29h	01h	0000h = ×8 only 0001h = ×16 only 0002h = ×8/×16 capable 0003h = ×32 only 0004h = Single I/O SPI, 3-byte address 0005h = Multi I/O SPI, 3-byte address 0102h = Multi I/O SPI, 3- or 4-byte address 0103h = Dual-Quad SPI, 3 or 4-byte address
2Ah	09h	Max. number of bytes in multi-byte write = 2 <sup>N</sup>
2Bh	00h	(0000 = Not supported 0008h = 256B page) 0009h = 512B page) 0000A = 1024B page
2Ch	02h	Number of Erase Block Regions within device 1 = Uniform Device, 2 = Boot Device
2Dh	1Fh	Erase Block Region 1 Information (refer to JEDEC JEP137) 32 sectors = 32-1 = 001Fh 8-KB sectors = 256 bytes x 0010h
2Eh	00h	
2Fh	10h	
30h	00h	
31h	FDh	Erase Block Region 2 Information: 254 sectors = 254-1 = 00FDh (256 Mb) 510 sectors = 510-1 = 01FDh (512 Mb) 128-KB sectors = 0100h × 256 bytes
32h	00h (256 Mb) 01h (512 Mb)	
33h	00h	
34h	01h	
35h thru 3Fh	FFh	RFU

**Table 55 Device geometry definition for 256-Mb and 512-Mb uniform sector devices**

Byte address	Data	Description
27h	19h (256 Mb) 1Ah (512 Mb)	Device size = 2 <sup>N</sup> bytes
28h	03h	Flash device interface description:
29h	01h	0000h = ×8 only 0001h = ×16 only 0002h = ×8/×16 capable 0003h = ×32 only 0004h = Single I/O SPI, 3-byte address 0005h = Multi I/O SPI, 3-byte address 0102h = Multi I/O SPI, 3- or 4-byte address 0103h = Dual-Quad SPI, 3 or 4-byte address
2Ah	0Ah	Max. number of bytes in multi-byte write = 2 <sup>N</sup>
2Bh	00h	(0000 = Not supported 0008h = 256B page 0009h = 512B page 0000A = 1024B page
2Ch	01h	Number of Erase Block Regions within device 1 = Uniform Device, 2 = Boot Device
2Dh	3Fh (256 Mb) 7Fh (512 Mb)	Erase Block Region 1 Information (refer to JEDEC JEP137):
2Eh	00h	64 sectors = 64-1 = 003Fh (256 Mb)
2Fh	00h	128 sectors = 128-1 = 007Fh (512 Mb)
30h	04h	512-KB sectors = 256 bytes × 0400h
31h thru 3Fh	FFh	RFU

**Table 56 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) 00b = Required 01b = Not required Process technology (Bits 5–2) 0000b = 0.23 μm Floating Gate 0001b = 0.17 μm Floating Gate 0010b = 0.23 μm MIRRORBIT™ 0011b = 0.11 μm Floating Gate 0100b = 0.11 μm MIRRORBIT™ 0101b = 0.09 μm MIRRORBIT™ 1000b = 0.065 μm MIRRORBIT™
46h	02h	Erase Suspend 0 = Not supported 1 = Read only 2 = Read and program
47h	01h	Sector Protect 00 = Not supported X = Number of sectors in group
48h	00h	Temporary Sector Unprotect 00 = Not supported 01 = Supported
49h	08h	Sector Protect/Unprotect Scheme 04 = High Voltage method 05 = Software Command Locking method 08 = Advanced Sector Protection method 09 = Secure
4Ah	00h	Simultaneous operation 00 = Not supported X = Number of Sectors
4Bh	01h	Burst Mode (Synchronous sequential read) support 00 = Not supported 01 = Supported
4Ch	05h	Page Mode Type, model dependent 00 = Not supported 01 = 4 Word Read Page 02 = 8 Read Word Page 03 = 256-byte Program Page 04 = 512-byte Program Page 05 = 1024-byte Program Page
4Dh	00h	ACC (Acceleration) supply minimum 00 = Not supported, 100 mV



**Table 56 CFI primary vendor-specific extended query (Continued)**

Byte address	Data	Description
4Eh	00h	ACC (Acceleration) supply maximum 00 = Not supported, 100 mV
4Fh	00h	WP# Protection 00 = None 01 = Whole Chip 04 = Uniform Device with Bottom WP Protect 05 = Uniform Device with Top WP Protect 07 = Uniform Device with Top or Bottom Write Protect (user select)
50h	01h	Program Suspend 00 = Not supported 01 = Supported

The alternate vendor-specific extended query provides information related to the expanded command set provided by the S25FL-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 57 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 58 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 (Infineon)
03h	37h	ASCII "79" for Product Characters (Dual-Quad SPI)
04h	39h	
05h	46h	
06h	4Ch	ASCII "FL" for Interface Characters (SPI 3 Volt)
07h	32h (256 Mb) 35h (512 Mb)	ASCII characters for density
08h	35h (256 Mb) 31h (512 Mb)	
09h	36h (256 Mb) 32h (512 Mb)	
0Ah	53h	ASCII "S" for Technology (65 nm MIRRORBIT™)
0Bh	xxh	Reserved for Future Use (RFU)
0Ch	xxh	
0Dh	xxh	
0Eh	xxh	
0Fh	xxh	
10h	xxh	
11h	xxh	

**Table 59 CFI alternate vendor-specific extended query parameter 80h address options**

Parameter relative byte address offset	Data	Description
00h	80h	Parameter ID (Ordering part number)
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 Bit 3 - AutoBoot support - Yes = 0b, No = 1b Bit 2 - 4-byte address instructions supported - Yes = 0b, No = 1b Bit 1 - Bank address + 3-byte address instructions supported - Yes = 0b, No = 1b Bit 0 - 3-byte address instructions supported - Yes = 0b, No = 1b

**Table 60 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	28h	Program Suspend latency maximum ( $\mu$ s)
04h	8Ah	Program Resume instruction code
05h	64h	Program Resume to next Suspend typical ( $\mu$ s)
06h	75h	Erase Suspend instruction code
07h	28h	Erase Suspend latency maximum ( $\mu$ s)
08h	7Ah	Erase Resume instruction code
09h	64h	Erase Resume to next Suspend typical ( $\mu$ s)

**Table 61 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	0Bh	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 00h = FL-P, FL-S, FFh = Not supported
05h	01h	Advanced Sector Protection type, model dependent 01h = FL-S ASP

**Table 62 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 $2^N \mu\text{s}$
04h	23h	Hardware Reset maximum value, FFh = Not supported
05h	00h	Hardware Reset maximum exponent $2^N \mu\text{s}$
06h	23h	Software Reset maximum value, FFh = Not supported
07h	00h	Software Reset maximum exponent $2^N \mu\text{s}$

**Table 63 CFI alternate vendor-specific extended query parameter 90h - EHPLC (SDR)**

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	FFh	Read Dual Out 3-byte address instruction
0Bh	FFh	Read Dual Out 3-byte address instruction
0Ch	6Bh	Read Quad Out 3-byte address instruction
0Dh	6Ch	Read Quad Out 4-byte address instruction
0Eh	FFh	Dual I/O Read 3-byte address instruction
0Fh	FFh	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)

**Note**

51.FFh = Not supported.

**Table 63 CFI alternate vendor-specific extended query parameter 90h - EHPLC (SDR) (Continued)**

Parameter relative byte address offset	Data	Description
14h	00h	Read mode cycles
15h	00h	Read latency cycles
16h	00h	Read Fast mode cycles
17h	00h	Read Fast latency cycles
18h	FFh	Read Dual Out mode cycles
19h	FFh	Read Dual Out mode cycles
1Ah	00h	Read Quad Out mode cycles
1Bh	00h	Read Quad Out latency cycles
1Ch	FFh	Dual I/O Read mode cycles
1Dh	FFh	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
25h	08h	Read Fast latency cycles
26h	00h	Read Dual Out mode cycles
27h	FFh	Read Dual Out latency cycles
28h	00h	Read Quad Out mode cycles
29h	08h	Read Quad Out latency cycles
2Ah	FFh	Dual I/O Read mode cycles
2Bh	FFh	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	FFh	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	00h	Dual I/O Read mode cycles

**Note**

51.FFh = Not supported.

**Table 63 CFI alternate vendor-specific extended query parameter 90h - EHPLC (SDR) (Continued)**

<b>Parameter relative byte address offset</b>	<b>Data</b>	<b>Description</b>
39h	FFh	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 (104 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	FFh	Read Dual Out mode cycles
43h	FFh	Read Dual Out latency cycles
44h	00h	Read Quad Out mode cycles
45h	08h	Read Quad Out latency cycles
46h	FFh	Dual I/O Read mode cycles
47h	FFh	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)
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

**Note**

51.FFh = Not supported.

**Table 64 CFI alternate vendor-specific extended query parameter 9Ah - EHPLC (DDR)**

Parameter relative byte address offset	Data	Description
00h	9Ah	Parameter ID (Latency code table)
01h	2Ah	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	05h	Number of rows
03h	08h	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	FFh	Read Fast DDR 3-byte address instruction
07h	FFh	Read Fast DDR 4-byte address instruction
08h	FFh	DDR Dual I/O Read 3-byte address instruction
09h	FFh	DDR Dual I/O Read 4-byte address instruction
0Ah	EDh	Read DDR Quad I/O 3-byte address instruction
0Bh	EEh	Read DDR Quad I/O 4-byte address instruction
0Ch	32h	Start of row 2, SCK frequency limit for this row (50 MHz)
0Dh	03h	Latency Code for this row (11b)
0Eh	FFh	Read Fast DDR mode cycles
0Fh	FFh	Read Fast DDR latency cycles
10h	FFh	DDR Dual I/O Read mode cycles
11h	FFh	DDR Dual I/O Read latency cycles
12h	01h	Read DDR Quad I/O mode cycles
13h	03h	Read DDR Quad I/O latency cycles
14h	42h	Start of row 3, SCK frequency limit for this row (66 MHz)
15h	00h	Latency Code for this row (00b)
16h	FFh	Read Fast DDR mode cycles
17h	FFh	Read Fast DDR latency cycles
18h	FFh	DDR Dual I/O Read mode cycles
19h	FFh	DDR Dual I/O Read latency cycles
1Ah	01h	Read DDR Quad I/O mode cycles
1Bh	06h	Read DDR Quad I/O latency cycles
1Ch	42h	Start of row 4, SCK frequency limit for this row (66 MHz)
1Dh	01h	Latency Code for this row (01b)
1Eh	FFh	Read Fast DDR mode cycles
1Fh	FFh	Read Fast DDR latency cycles
20h	FFh	DDR Dual I/O Read mode cycles
21h	FFh	DDR Dual I/O Read latency cycles

**Note**

52.FFh = Not supported.

**Table 64 CFI alternate vendor-specific extended query parameter 9Ah - EHPLC (DDR) (Continued)**

Parameter relative byte address offset	Data	Description
22h	01h	Read DDR Quad I/O mode cycles
23h	07h	Read DDR Quad I/O latency cycles
24h	42h	Start of row 5, SCK frequency limit for this row (66 MHz)
25h	02h	Latency Code for this row (10b)
26h	FFh	Read Fast DDR mode cycles
27h	FFh	Read Fast DDR latency cycles
28h	FFh	DDR Dual I/O Read mode cycles
29h	FFh	DDR Dual I/O Read latency cycles
2Ah	01h	Read DDR Quad I/O mode cycles
2Bh	08h	Read DDR Quad I/O latency cycles

**Note**

52.FFh = Not supported.

**Table 65 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.



### 12.3 Initial delivery state

The device is shipped from Infineon 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 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 02h.
- The Autoboot Register contains 00h.
- The Password Register contains FFFFFFFF–FFFFFFFh.
- All PPB bits are '1'.

The ASP Register contents are shown below.

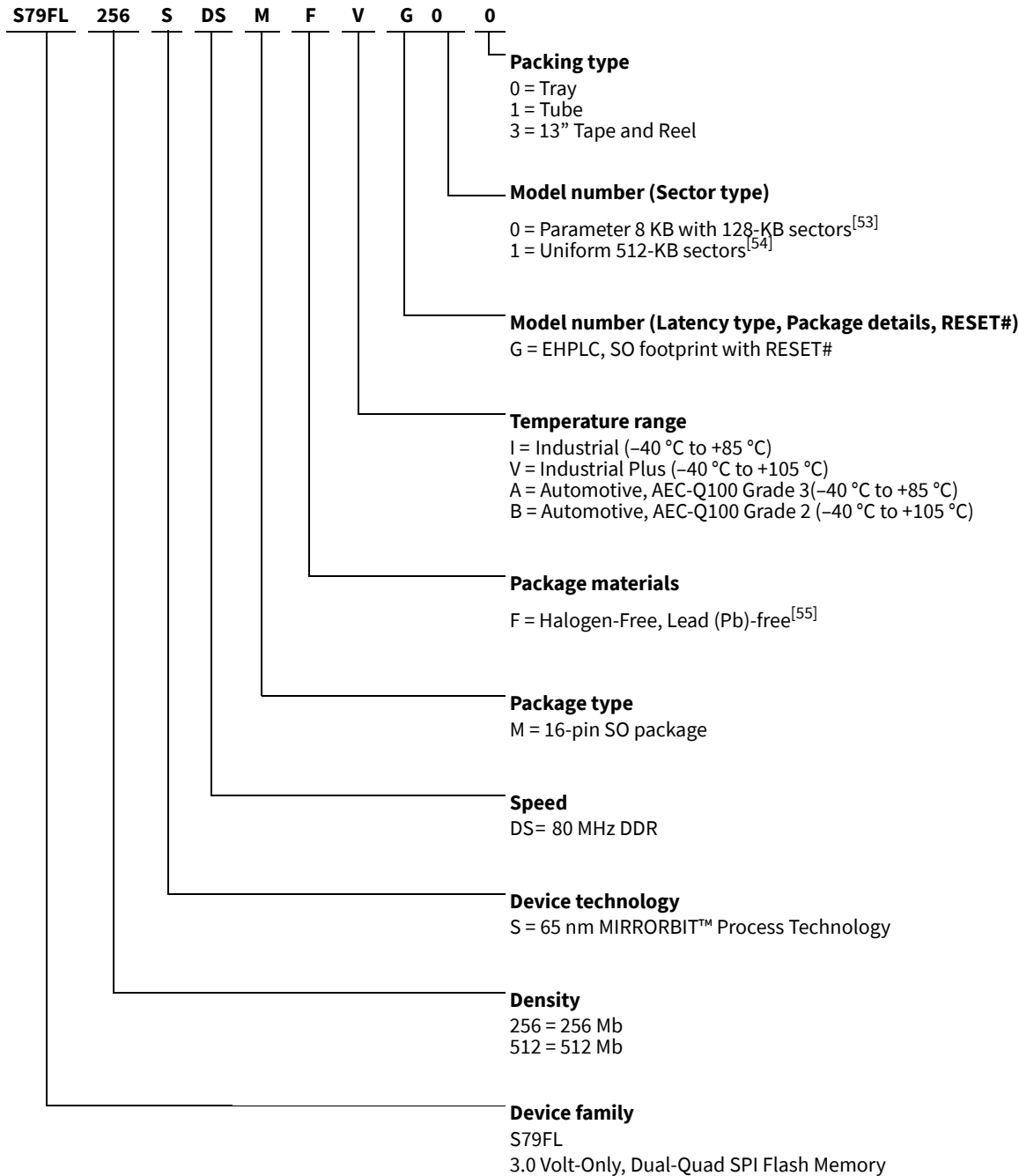
**Table 66 ASP Register content**

<b>Ordering part number model</b>	<b>ASPR default value</b>
G0	FE7Fh

## 13 Ordering information

### 13.1 Ordering information S79FL256S/S79FL512S

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



#### Notes

53. Parameter with 128-KB sectors = A parameter of 32 × 8-KB sectors with all remaining sectors being 128 KB, with a 512B programming buffer.
54. Uniform 512-KB sectors = All sectors are uniform 512-KB with a 1024B programming buffer.
55. Halogen free definition is in accordance with IEC 61249-2-21 specification.

### 13.2 Valid combinations

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

**Table 67 Valid combinations**

Base ordering part number	Speed option	Package and temperature	Model number	Packing type	Package Marking
S79FL256S	DS	MFI, MFV	G0	0, 1, 3	79FL256S + S + (Temp) + F + (Model Number)
S79FL512S	DS	MFI, MFV	G0	0, 1, 3	79FL512S + S + (Temp) + F + (Model Number)

### 13.3 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 68 Valid combinations**

Base ordering part number	Speed option	Package and temperature	Model number	Packing type	Package marking
S79FL256S	DS	MFA, MFB	G0	0, 1, 3	79FL256S + S + (Temp) + F + (Model Number)
S79FL512S	DS	MFA, MFB	G0	0, 1, 3	79FL512S + S + (Temp) + F + (Model Number)

## Revision history

Document version	Date of release	Description of changes
**	2014-09-25	Initial release.
*A	2015-10-20	Updated to Cypress template.
*B	2016-08-09	<p>Changed status from Preliminary to Final.  Updated <b>Overview</b>:  Updated “Other resources”:  Added “Cypress flash memory roadmap”.  Updated <b>Timing specifications</b>:  Updated <b>AC test conditions</b>:  Updated <b>Capacitance characteristics</b>:  Updated <b>Table 9</b>:  Changed maximum value of C<sub>IN</sub> parameter from 8 pF to 14 pF.  Changed maximum value of C<sub>OUT</sub> parameter from 8 pF to 20 pF.  Updated <b>Address space maps</b>:  Updated <b>Registers</b>:  Updated <b>Configuration Register-1 (CR1)</b>:  Updated <b>Table 21</b>:  Updated details in all columns corresponding to Bit 2.  Updated <b>Ordering information</b>:  Updated <b>Ordering information S79FL256S/S79FL512S</b>:  Updated details corresponding to “0” under “Model Number (Sector Type)” and also updated the corresponding note.  Removed Note “EHPLC = Enhanced High Performance Latency Code table”.  Updated to new template.  Completing Sunset Review.</p>
*C	2017-03-10	<p>Added ECC related information in all instances across the document.  Updated <b>Physical interface</b>:  Updated <b>SOIC 16 physical diagram</b>:  Replaced existing spec with 002-15547 *A.  Updated <b>Ordering information S79FL256S/S79FL512S</b>:  Updated details under “Temperature Range”.  Added <b>Data integrity</b>.  Updated to new template.</p>
*D	2017-11-09	Updated Cypress Logo and Copyright.

Revision history

Document version	Date of release	Description of changes
*E	2019-05-02	<p>Updated <b>Overview</b>:  Updated “Glossary”:  Replaced MSB with MSb and LSB with LSb.  Updated <b>Electrical specifications</b>:  Added <b>Thermal resistance</b>.  Updated <b>Timing specifications</b>:  Updated <b>SDR AC characteristics</b>:  Updated <b>Table 11</b>:  Removed the maximum value of <math>t_{CSH}</math> parameter.  Added “3000” as the maximum value of <math>t_{SU}</math> parameter.  Updated <b>DDR AC characteristics</b>:  Updated <b>DDR output timing</b>:  Removed figure “SPI DDR Data Valid Window”.  Added <b>DDR data valid timing using DLP</b>.  Updated <b>Address space maps</b>:  Updated <b>Registers</b>:  Updated <b>Configuration Register-1 (CR1)</b>:  Updated <b>Table 21</b>:  Changed CR1[4] from RFU to DNU (Updated details under “Field Name” and “Function” columns).  Updated <b>Commands</b>:  Updated <b>Command set summary</b>:  Updated <b>Table 41</b>:  Changed value of DDRQIOR command from 66 MHz to 80 MHz corresponding to Instruction Value ED.  Changed value of 4DDRQIOR command from 66 MHz to 80 MHz corresponding to Instruction Value EE.  Updated <b>Ordering information</b>:  Updated <b>Ordering information S79FL256S/S79FL512S</b>:  Updated details under “Package Materials”.  Added Note 55 and referred the same note in “F = Halogen-Free, Lead (Pb)-free”.  Updated to new template.</p>
*F	2022-05-27	<p>Updated Document Title to read as “S79FL256S, S79FL512S, 256 Mb (32 MB) / 512 Mb (64MB) FL-S Flash SPI Dual-Quad, 3.0 V”.  Updated <b>Overview</b>:  Removed “Glossary”.  Removed “Other resources”.  Updated <b>SPI with multiple input / output (SPI-MIO) dual-quad</b>:  Replaced “Hardware interface” with “SPI with multiple input / output (SPI-MIO) dual-quad” in heading.  Removed “Software interface”.  Updated <b>Data integrity</b>:  Updated <b>Data retention</b>:  Updated description.  Migrated to Infineon template.</p>

## Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

**Edition 2022-05-27**  
**Published by**  
**Infineon Technologies AG**  
**81726 Munich, Germany**

**© 2022 Infineon Technologies AG.**  
**All Rights Reserved.**

**Do you have a question about this document?**  
Go to [www.infineon.com/support](http://www.infineon.com/support)

**Document reference**  
**002-00518 Rev. \*F**

## IMPORTANT NOTICE

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenhheitsgarantie").

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office ([www.infineon.com](http://www.infineon.com)).

## WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.