

# Military 128 Mb (16 MB)/256 Mb (32 MB) FL-S Flash

**SPI Multi-I/O, 3.0 V**

## Features

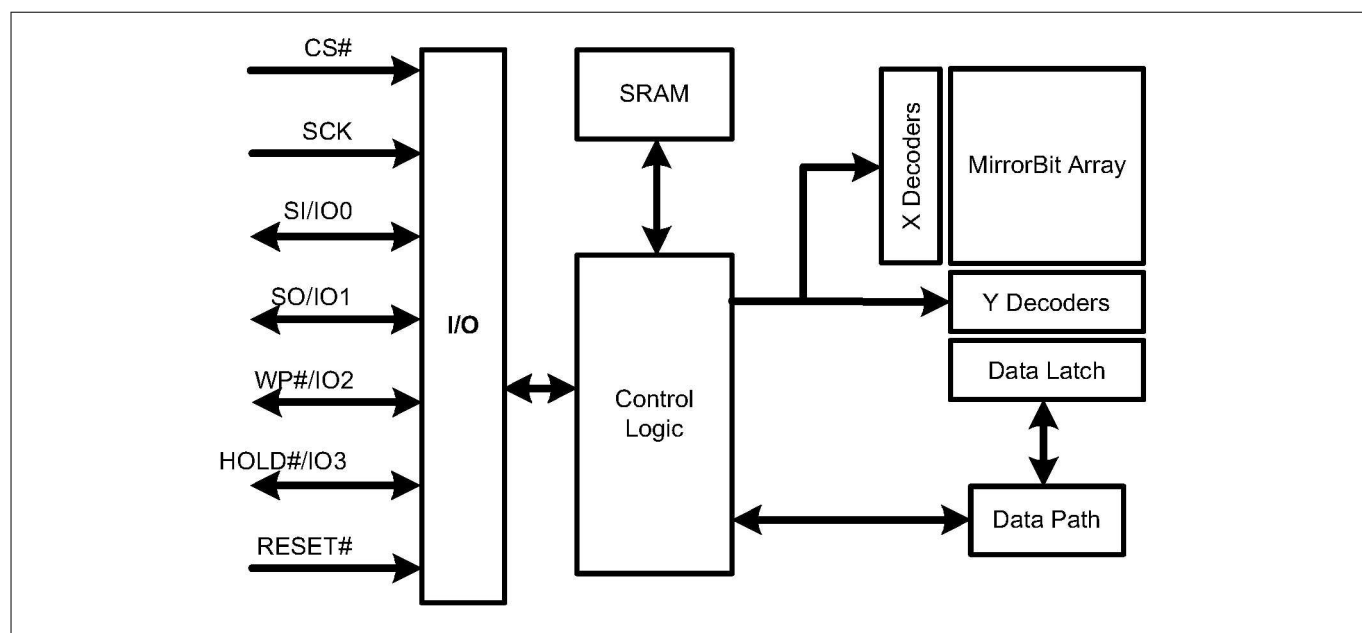
- CMOS 3.0-V core with versatile I/O
- Serial peripheral interface (SPI) with multi-I/O
  - SPI clock polarity and phase modes 0 and 3
  - Double data rate (DDR) option
  - Extended addressing: 24- or 32-bit address options
  - Serial command set and footprint compatible with S25FL-A, S25FL-K, and S25FL-P SPI families
  - Multi I/O command set and footprint compatible with S25FL-P SPI family
- READ commands
  - Normal, Fast, Dual, Quad, Fast DDR, Dual DDR, and 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 (1.5 MBps)
  - 256- or 512-byte page programming buffer options
  - Quad-input page programming (QPP) for slow clock systems
  - Automatic ECC -internal hardware error correction code generation with single bit error correction
- Erase (0.5 to 0.65 MBps)
  - Hybrid sector size option - physical set of thirty two 4-KB sectors at top or bottom of address space with all remaining sectors of 64 KB, for compatibility with prior generation S25FL devices
  - Uniform sector option - always erase 256-KB blocks for software compatibility with higher density and future devices.
- Cycling endurance
  - 100 program-erase cycles
- Data retention
  - >20 year data retention
- Security features
  - OTP array of 1024 bytes
  - Block protection
    - Status Register bits to control protection against program or erase of a contiguous range of sectors.
    - Hardware and software control options
  - Advanced sector protection (ASP)
    - Individual sector protection controlled by boot code or password
- 65-nm MIRRORBIT™ technology with Eclipse architecture
- Core supply voltage: 2.7 V to 3.6 V
- I/O supply voltage: 1.65 V to 3.6 V
  - FBGA packages

## Features

- Temperature range/grade:
  - Military (-55°C to +125°C)
- Packages (all Pb-free)
  - WSON 6 × 8 mm
  - BGA-24 6 × 8 mm
    - 5 × 5 ball (FAB024) footprint option

**Logic block diagram**

**Logic block diagram**



**Figure 1**      **Logic block diagram**

**Performance summary**

## Performance summary

**Table 1** Maximum read rates with the same core and I/O voltage ( $V_{IO} = V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ )

Command	Clock rate (MHz)	MBps
Read	50	6.25
Fast Read	133	16.6
Dual Read	104	26
Quad Read	104	52

**Table 2** Maximum read rates with lower I/O voltage ( $V_{IO} = 1.65 \text{ V to } 2.7 \text{ V}$ ,  $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ )

Command	Clock rate (MHz)	MBps
Read	50	6.25
Fast Read	66	8.25
Dual Read	66	16.5
Quad Read	66	33

**Table 3** Maximum read rates DDR ( $V_{IO} = V_{CC} = 3 \text{ V to } 3.6 \text{ V}$ )

Command	Clock rate (MHz)	MBps
Fast Read DDR	80	20
Dual Read DDR	80	40
Quad Read DDR	80	80

**Table 4** Typical program and erase rates

Operation	KBps
Page programming (256-byte page buffer - Hybrid sector option)	1000
Page programming (512-byte page buffer - Uniform sector option)	1500
4-KB physical sector erase (Hybrid sector option)	30
64-KB physical sector erase (Hybrid sector option)	500
256-KB logical sector erase (Uniform sector option)	500

**Table 5** Current consumption

Operation	Current (mA)
Serial read 50 MHz	22 (max)
Serial read 133 MHz	33 (max)
Quad read 104 MHz	61 (max)
Quad DDR read 80 MHz	90 (max)
Program	100 (max)
Erase	100 (max)
Standby	0.115 (typ)



## Table of contents

	<b>Features</b> .....	1
	<b>Logic block diagram</b> .....	3
	<b>Performance summary</b> .....	4
	<b>Table of contents</b> .....	5
<b>1</b>	<b>Overview</b> .....	11
1.1	General description .....	11
1.2	Migration notes .....	11
1.2.1	Features comparison .....	11
1.2.2	Known differences from prior generations .....	12
1.2.2.1	Error reporting .....	12
1.2.2.2	Secure silicon region (OTP) .....	12
1.2.2.3	Configuration register freeze bit .....	12
1.2.2.4	Sector Erase commands .....	13
1.2.2.5	Deep power down .....	13
1.2.2.6	New features .....	13
<b>2</b>	<b>Serial peripheral interface with multiple input/output (SPI-MIO)</b> .....	14
<b>3</b>	<b>Pinouts and signal descriptions</b> .....	15
3.1	FAB024 24-ball BGA connection diagram .....	15
3.2	WNG008 - WSON 8-contact connection diagram .....	15
3.3	Special handling instructions for FBGA packages .....	15
3.4	Input/output summary .....	15
3.5	Address and data configuration .....	17
3.6	RESET# .....	17
3.7	Serial Clock (SCK) .....	18
3.8	Chip Select (CS#) .....	18
3.9	Serial Input (SI) / IO0 .....	18
3.10	Serial Output (SO) / IO1 .....	18
3.11	Write Protect (WP#) / IO2 .....	18
3.12	Hold (HOLD#) / IO3 .....	19
3.13	Core Voltage Supply (V <sub>CC</sub> ) .....	19
3.14	Versatile I/O Power Supply (V <sub>IO</sub> ) .....	20
3.15	Supply and Signal Ground (V <sub>SS</sub> ) .....	20
3.16	Not Connected (NC) .....	20
3.17	Reserved for Future Use (RFU) .....	20
3.18	Do Not Use (DNU) .....	20
3.19	Block diagrams .....	21
<b>4</b>	<b>Signal protocols</b> .....	22

## Table of contents

4.1	SPI clock modes .....	22
4.1.1	SDR .....	22
4.1.2	DDR .....	22
4.2	Command protocol .....	23
4.2.1	Command sequence examples .....	24
4.3	Interface states .....	27
4.3.1	Power-off .....	30
4.3.2	Low power hardware data protection .....	30
4.3.3	Power-on (cold) reset .....	31
4.3.4	Hardware (warm) reset .....	31
4.3.5	Interface standby .....	31
4.3.6	Instruction cycle .....	31
4.3.7	Hold .....	31
4.3.8	Single input cycle - Host to Memory transfer .....	32
4.3.9	Single latency (dummy) cycle .....	32
4.3.10	Single output cycle - Memory to Host transfer .....	32
4.3.11	Dual input cycle - Host to Memory transfer .....	32
4.3.12	Dual latency (dummy) cycle .....	32
4.3.13	Dual output cycle - Memory to Host transfer .....	33
4.3.14	QPP or QOR address input cycle .....	33
4.3.15	Quad input cycle - Host to Memory transfer .....	33
4.3.16	Quad latency (dummy) cycle .....	33
4.3.17	Quad output cycle - Memory to Host transfer .....	33
4.3.18	DDR single input cycle - Host to Memory transfer .....	34
4.3.19	DDR dual input cycle - Host to Memory transfer .....	34
4.3.20	DDR Quad input cycle - Host to Memory transfer .....	34
4.3.21	DDR latency cycle .....	34
4.3.22	DDR single output cycle - Memory to Host transfer .....	34
4.3.23	DDR dual output cycle - Memory to Host transfer .....	34
4.3.24	DDR Quad output cycle - Memory to Host transfer .....	35
4.4	Configuration Register effects on the interface .....	35
4.5	Data protection .....	35
4.5.1	Power-up .....	35
4.5.2	Low power .....	35
4.5.3	Clock pulse count .....	35
<b>5</b>	<b>Electrical specifications .....</b>	<b>36</b>
5.1	Absolute maximum ratings .....	36
5.2	Thermal resistance .....	36
5.3	Operating ranges .....	36
5.3.1	Power supply voltages .....	36
5.3.2	Temperature ranges .....	37
5.3.3	Input signal overshoot .....	37

**Table of contents**

5.4	Power-up and power-down .....	37
5.5	DC characteristics .....	39
5.5.1	Active power and standby power modes .....	40
<b>6</b>	<b>Timing specifications .....</b>	<b>41</b>
6.1	Key to switching waveforms .....	41
6.2	AC test conditions .....	41
6.2.1	Capacitance characteristics .....	42
6.3	Reset .....	42
6.3.1	Power-on (Cold) reset .....	42
6.3.2	Hardware (warm) reset .....	43
6.4	SDR AC characteristics .....	44
6.4.1	Clock timing .....	48
6.4.2	Input/output timing .....	48
6.5	DDR AC characteristics .....	50
6.5.1	DDR input timing .....	51
6.5.2	DDR output timing .....	52
<b>7</b>	<b>Address space maps .....</b>	<b>54</b>
7.1	Overview .....	54
7.1.1	Extended address .....	54
7.1.2	Multiple address spaces .....	54
7.2	Flash memory array .....	54
7.3	ID-CFI address space .....	56
7.4	OTP address space .....	56
<b>8</b>	<b>Registers .....</b>	<b>59</b>
8.1	Status Register 1 (SR1) .....	59
8.2	Configuration Register 1 (CR1) .....	62
8.3	Status Register 2 (SR2) .....	66
8.4	AutoBoot Register .....	67
8.5	Bank Address Register .....	67
8.6	ECC Status Register (ECCSR) .....	68
8.7	ASP Register (ASPR) .....	69
8.8	Password Register (PASS) .....	70
8.9	PPB Lock Register (PPBL) .....	71
8.10	PPB Access Register (PPBAR) .....	71
8.11	DYB Access Register (DYBAR) .....	71
8.12	SPI DDR Data Learning Registers .....	72
<b>9</b>	<b>Embedded algorithm performance tables .....</b>	<b>74</b>
<b>10</b>	<b>Data protection .....</b>	<b>76</b>
10.1	Secure silicon region (OTP) .....	76
10.1.1	Reading OTP memory space .....	76

**Table of contents**

10.1.2	Programming OTP memory space .....	76
10.1.3	Infineon programmed random number .....	76
10.1.4	Lock bytes .....	76
10.2	Write Enable command .....	76
10.3	Block protection .....	77
10.3.1	Freeze bit .....	78
10.3.2	Write Protect signal .....	78
10.4	Advanced sector protection .....	78
10.4.1	ASP Register .....	79
10.4.2	Persistent protection bits .....	79
10.4.3	Dynamic protection bits .....	80
10.4.4	PPB Lock Bit (PPBL[0]) .....	80
10.4.5	Sector protection states summary .....	80
10.4.6	Persistent Protection mode .....	81
10.4.7	Password Protection mode .....	81
<b>11</b>	<b>Commands .....</b>	<b>83</b>
11.1	Command set summary .....	84
11.1.1	Extended addressing .....	84
11.1.2	Read device identification .....	89
11.1.3	Register read or write .....	89
11.1.3.1	Monitoring operation status .....	90
11.1.3.2	Configuration .....	90
11.1.4	Read flash array .....	90
11.1.5	Program flash array .....	90
11.1.6	Erase flash array .....	90
11.1.7	OTP, block protection, and advanced sector protection .....	91
11.1.8	Reset .....	91
11.1.9	Reserved .....	91
11.2	Identification commands .....	91
11.2.1	Read Identification - REMS (Read_ID or REMS 90 h) .....	91
11.2.2	Read Identification (RDID 9 Fh) .....	92
11.2.3	Read Electronic Signature (RES) (ABh) .....	93
11.3	Register access commands .....	94
11.3.1	Read Status Register-1 (RDSR1 05 h) .....	94
11.3.2	Read Status Register-2 (RDSR2 07 h) .....	94
11.3.3	Read Configuration Register (RDCR 35 h) .....	94
11.3.4	Bank Register Read (BRRD 16 h) .....	95
11.3.5	Bank Register Write (BRWR 17 h) .....	95
11.3.6	Bank Register Access (BRAC B9h) .....	96
11.3.7	Write registers (WRR 01 h) .....	97
11.3.8	Write Enable (WREN 06 h) .....	99
11.3.9	Write Disable (WRDI 04 h) .....	100

**Table of contents**

11.3.10	Clear Status Register (CLSR 30 h) .....	100
11.3.11	ECC Status Register Read (ECCRD 18 h) .....	101
11.3.12	AutoBoot .....	101
11.3.13	AutoBoot Register Read (ABRD 14 h) .....	102
11.3.14	AutoBoot Register Write (ABWR 15 h) .....	103
11.3.15	Program NVDLR (PNVDLR 43 h) .....	103
11.3.16	Write VDLR (WVDLR 4 Ah) .....	104
11.3.17	Data Learning Pattern Read (DLPRD 41 h) .....	104
11.4	Read memory array commands .....	105
11.4.1	Read (Read 03 h or 4READ 13 h) .....	106
11.4.2	Fast Read (FAST_READ 0 Bh or 4FAST_READ 0 Ch) .....	107
11.4.3	Dual Output Read (DOR 3 Bh or 4DOR 3 Ch) .....	108
11.4.4	Quad Output Read (QOR 6 Bh or 4QOR 6 Ch) .....	109
11.4.5	Dual I/O Read (DIOR BBh or 4DIOR BCh) .....	110
11.4.6	Quad I/O Read (QIOR EBh or 4QIOR ECh) .....	112
11.4.7	DDR Fast Read (DDRFR 0 Dh, 4DDRFR 0 Eh) .....	115
11.4.8	DDR Dual I/O Read (BDh, BEh) .....	117
11.4.9	DDR Quad I/O Read (EDh, EEh) .....	119
11.5	Program flash array commands .....	122
11.5.1	Program granularity .....	122
11.5.1.1	Automatic ECC .....	122
11.5.1.2	Page programming .....	123
11.5.1.3	Single byte programming .....	123
11.5.2	Page Program (PP 02 h or 4PP 12 h) .....	123
11.5.3	Quad Page Program (QPP 32 h or 38 h, or 4QPP 34 h) .....	124
11.5.4	Program Suspend (PGSP 85 h) and Resume (PGRS 8 Ah) .....	127
11.6	Erase flash array commands .....	128
11.6.1	Parameter 4-KB Sector Erase (P4E 20 h or 4P4E 21 h) .....	128
11.6.2	Sector Erase (SE D8h or 4SE DCh) .....	129
11.6.3	Bulk Erase (BE 60 h or C7h) .....	130
11.6.4	Erase Suspend and Resume Commands (ERSP 75 h or ERRS 7 Ah) .....	131
11.7	One Time Program Array commands .....	136
11.7.1	OTP Program (OTPP 42 h) .....	136
11.7.2	OTP Read (OTPR 4 Bh) .....	136
11.8	Advanced Sector Protection commands .....	137
11.8.1	ASP Read (ASPRD 2 Bh) .....	137
11.8.2	ASP Program (ASPP 2 Fh) .....	137
11.8.3	DYB Read (DYBRD E0h) .....	138
11.8.4	DYB Write (DYBWR E1h) .....	138
11.8.5	PPB Read (PPBRD E2h) .....	139
11.8.6	PPB Program (PPBP E3h) .....	139
11.8.7	PPB Erase (PPBE E4h) .....	140

## Table of contents

11.8.8	PPB Lock Bit Read (PLBRD A7h) .....	140
11.8.9	PPB Lock Bit Write (PLBWR A6h) .....	141
11.8.10	Password Read (PASSRD E7h) .....	141
11.8.11	Password Program (PASSP E8h) .....	142
11.8.12	Password Unlock (PASSU E9h) .....	142
11.9	Reset commands .....	143
11.9.1	Software Reset Command (RESET F0h) .....	143
11.9.2	Mode Bit Reset (MBR FFh) .....	144
<b>12</b>	<b>Data integrity</b> .....	<b>145</b>
12.1	Erase endurance .....	145
12.2	Data retention .....	145
<b>13</b>	<b>Device identification</b> .....	<b>146</b>
13.1	Command summary .....	146
13.2	Device ID and common flash interface (ID-CFI) address map .....	149
13.2.1	Field definitions .....	149
13.3	Device ID and common flash interface (ID-CFI) ASO map — Automotive only .....	168
13.4	Registers .....	169
<b>14</b>	<b>Initial delivery state</b> .....	<b>177</b>
<b>15</b>	<b>Physical interface</b> .....	<b>178</b>
<b>16</b>	<b>Package diagrams</b> .....	<b>179</b>
<b>17</b>	<b>Ordering information</b> .....	<b>181</b>
17.1	Ordering part number .....	181
17.2	Valid Combinations — Military .....	181
<b>18</b>	<b>Acronyms</b> .....	<b>183</b>
	<b>Revision history</b> .....	<b>185</b>
	<b>Disclaimer</b> .....	<b>186</b>

## 1 Overview

# 1 Overview

## 1.1 General description

The Infineon S25FL128S and S25FL256S 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

This family of devices connect to a host system via a serial peripheral interface (SPI). Traditional SPI single bit serial input and output (Single I/O or SIO) is supported as well as optional two-bit (Dual I/O or DIO) and four-bit (Quad I/O or QIO) serial commands. This multiple width interface is called SPI Multi-I/O or MIO. In addition, the FL-S family adds support for Double Data Rate (DDR) read commands for SIO, DIO, and QIO that transfer address and read data on both edges of the clock.

The Eclipse architecture features a Page Programming Buffer that allows up to 128 words (256 bytes) or 256 words (512 bytes) to be programmed in one operation, resulting in faster effective programming and erase than prior generation SPI program or erase algorithms.

Executing code directly from flash memory is often called Execute-In-Place or XIP. By using FL-S devices at the higher clock rates supported, with QIO 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 S25FL128S and S25FL256S products offer high densities coupled with the flexibility and fast performance required by a variety of embedded applications. They are ideal for code shadowing, XIP, and data storage.

## 1.2 Migration notes

### 1.2.1 Features comparison

The S25FL128S and S25FL256S devices are command set and footprint compatible with prior generation FL-K and FL-P families.

**Table 6** FL generations comparison<sup>1), 2), 3), 4), 5)</sup>

Parameter	FL-K	FL-P	FL-S
Technology node	90-nm	90-nm	65-nm
Architecture	Floating Gate	MIRRORBIT™	MIRRORBIT™ Eclipse
Release date	In Production	In Production	2H2011
Density	4 Mb - 128 Mb	32 Mb - 256 Mb	128 Mb - 256 Mb
Bus width	x1, x2, x4	x1, x2, x4	x1, x2, x4
Supply voltage	2.7 V - 3.6 V	2.7 V - 3.6 V	2.7 V - 3.6 V/1.65 V - 3.6 V V <sub>IO</sub>
Normal read speed (SDR)	6 MBps (50 MHz)	5 MBps (40 MHz)	6 MBps (50 MHz)
Fast read speed (SDR)	13 MBps (104 MHz)	13 MBps (104 MHz)	17 MBps (133 MHz)
Dual read speed (SDR)	26 MBps (104 MHz)	20 MBps (80 MHz)	26 MBps (104 MHz)
Quad read speed (SDR)	52 MBps (104 MHz)	40 MBps (80 MHz)	52 MBps (104 MHz)
Fast read speed (DDR)	-	-	20 MBps (80 MHz)

(table continues...)

## 1 Overview

**Table 6** (continued) FL generations comparison<sup>1)</sup>, <sup>2)</sup>, <sup>3)</sup>, <sup>4)</sup>, <sup>5)</sup>

Parameter	FL-K	FL-P	FL-S
Dual read speed (DDR)	-	-	40 MBps (80 MHz)
Quad read speed (DDR)	-	-	80 MBps (80 MHz)
Program buffer size	256B	256B	256B/512B
Erase sector size	4 KB/32 KB/64 KB	64 KB/256 KB	64 KB/256 KB
Parameter sector size	4 KB	4 KB	4 KB (option)
Sector erase time (typ.)	30 ms (4 KB), 150 ms (64 KB)	500 ms (64 KB)	171 ms (64 KB), 685ms (256 KB)
Page programming time (typ.)	700 µs (256B)	1500 µs (256B)	400 µs (256B), 540 µs (512B)
OTP	768B (3 x 256B)	506B	1024B
Advanced sector protection	No	No	Yes
Auto Boot mode	No	No	Yes
Erase suspend/resume	Yes	No	Yes
Program suspend/resume	Yes	No	Yes
Operating temperature	-40°C to +85°C	-40°C to + 85°C/+ 105°C	-55°C to +125°C

- 1) 256B program page option only for 128-Mb and 256-Mb density FL-S devices.
- 2) FL-P column indicates FL129P MIO SPI device (for 128-Mb density).
- 3) 64-KB sector erase option only for 128-Mb/256-Mb density FL-P and FL-S devices.
- 4) FL-K family devices can erase 4-KB sectors in groups of 32 KB or 64 KB.
- 5) See the individual data sheets for further details.

## 1.2.2 Known differences from prior generations

### 1.2.2.1 Error reporting

Prior generation FL memories either do not have error status bits or do not set them if program or erase is attempted on a protected sector. The FL-S family does have error reporting status bits for program and erase operations. These can be set when there is an internal failure to program or erase or when there is an attempt to program or erase a protected sector. In either case the program or erase operation did not complete as requested by the command.

### 1.2.2.2 Secure silicon region (OTP)

The size and format (address map) of the One Time Program area is different from prior generations. The method for protecting each portion of the OTP area is different. For additional details see [Secure silicon region \(OTP\)](#).

### 1.2.2.3 Configuration register freeze bit

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



## **1 Overview**

### **1.2.2.4 Sector Erase commands**

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

The command for erasing a 4-KB sector is supported only in the 128-Mb and 256-Mb density FL-S devices and only for use on the thirty two 4-KB parameter sectors at the top or bottom of the device address space.

The erase command for 64-KB sectors are supported for the 128-Mb and 256-Mb density FL-S devices when the ordering option for 4-KB parameter sectors with 64-KB uniform sectors are used. The 64-KB erase command may be applied to erase a group of sixteen 4-KB sectors.

The erase command for a 256-KB sector replaces the 64-KB erase command when the ordering option for 256-KB uniform sectors is used for the 128-Mb and 256-Mb density FL-S devices.

### **1.2.2.5 Deep power down**

The deep power down (DPD) function is not supported in FL-S family devices.

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

### **1.2.2.6 New features**

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

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

## **2 Serial peripheral interface with multiple input/output (SPI-MIO)**

### **2 Serial peripheral interface with multiple input/output (SPI-MIO)**

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

The S25FL128S and S25FL256S devices reduce the number of signals for connection to the host system by serially transferring all control, address, and data information over 4 to 6 signals. This reduces the cost of the memory package, reduces signal switching power, and either reduces the host connection count or frees host connectors for use in providing other features.

The S25FL128S and S25FL256S devices use the industry standard single bit Serial Peripheral Interface (SPI) and also supports optional extension commands for two bit (Dual) and four bit (Quad) wide serial transfers. This multiple width interface is called SPI Multi-I/O or SPI-MIO.

3 Pinouts and signal descriptions

3 Pinouts and signal descriptions

3.1 FAB024 24-ball BGA connection diagram

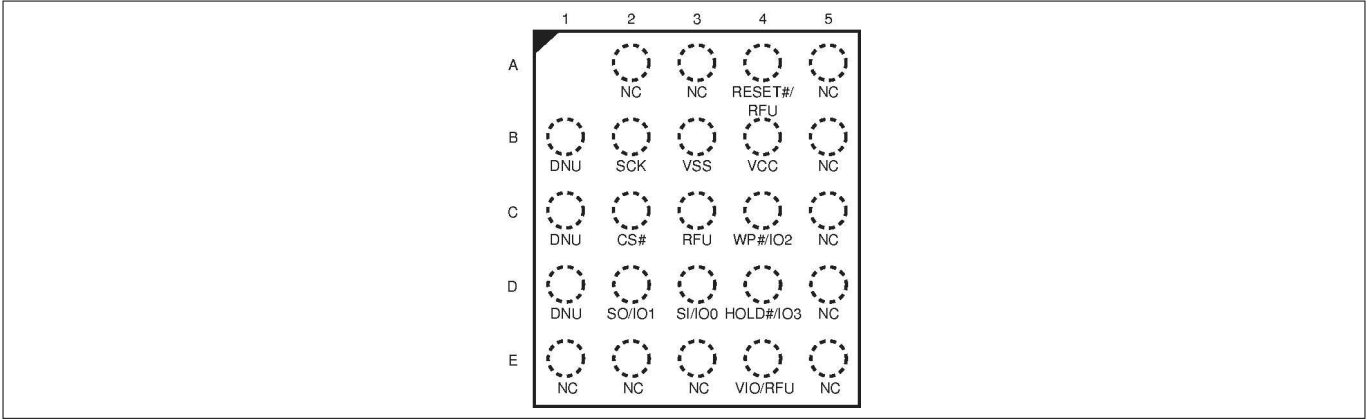


Figure 2 24-ball BGA, 5 x 5 ball footprint (FAB024), top view<sup>1)</sup>

3.2 WNG008 - WSON 8-contact connection diagram

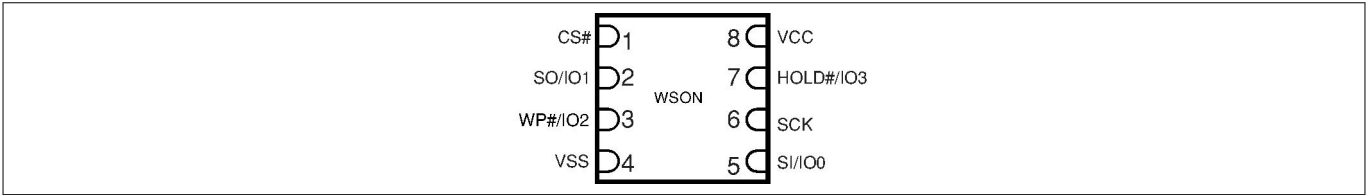


Figure 3 Leadless package (WSON), top view<sup>2)</sup>

3.3 Special handling instructions for FBGA packages

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

3.4 Input/output summary

Table 7 Signal list

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.

(table continues...)

<sup>1</sup> Signal connections are in the same relative positions as FAC024 BGA, allowing a single PCB footprint to use either package.

<sup>2</sup> RESET# and V<sub>IO</sub> are pulled to V<sub>CC</sub> internal to the memory device.

### 3 Pinouts and signal descriptions

**Table 7** (continued) **Signal list**

Signal name	Type	Description
SCK	Input	<b>Serial Clock</b>
CS#	Input	<b>Chip Select</b>
SI/IO0	I/O	<b>Serial Input</b> for single bit data commands or IO0 for Dual or Quad commands.
SO/IO1	I/O	<b>Serial Output</b> for single bit data commands. IO1 for Dual or Quad commands.
WP# / IO2	I/O	<b>Write Protect</b> when not in Quad mode. IO2 in Quad mode. The signal has an internal pull-up resistor and may be left unconnected in the host system if not used for Quad commands.
HOLD# / IO3	I/O	<b>Hold</b> (pause) serial transfer in single bit or Dual data commands. IO3 in Quad-I/O mode. The signal has an internal pull-up resistor and may be left unconnected in the host system if not used for Quad commands.
V <sub>CC</sub>	Supply	<b>Core Power Supply.</b>
V <sub>IO</sub>	Supply	<b>Versatile I/O Power Supply.</b>
V <sub>SS</sub>	Supply	<b>Ground.</b>
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 an NC must not have voltage levels higher than V <sub>IO</sub> .

(table continues...)

### 3 Pinouts and signal descriptions

**Table 7** (continued) **Signal list**

Signal name	Type	Description
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 of the connector for a signal. It is recommended to not use RFU connectors for PCB routing channels so that the PCB may take advantage of future enhanced features in compatible footprint devices.
DNU	Reserved	<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 $V_{IL}$ . The signal has an internal pull-down resistor and may be left unconnected in the host system or may be tied to $V_{SS}$ . Do not use these connections for PCB signal routing channels. Do not connect any host system signal to this connection.

### 3.5 Address and data configuration

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

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

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

### 3.6 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,

### 3 Pinouts and signal descriptions

- 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  $t_{PU}$  time.

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

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

RESET# has an internal pull-up resistor and 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.

#### 3.7 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.

#### 3.8 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.

#### 3.9 Serial Input (SI) / IO0

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

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

#### 3.10 Serial Output (SO) / IO1

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

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

#### 3.11 Write Protect (WP#) / IO2

When WP# is driven Low ( $V_{IL}$ ), during a WRR command and while the Status Register Write Disable (SRWD) bit of the Status Register is set to a 1, it is not possible to write to the Status and Configuration Registers. This prevents any alteration of the Block Protect (BP2, BP1, BP0) and TBPROT bits of the Status Register. As a

### 3 Pinouts and signal descriptions

consequence, all the data bytes in the memory area that are protected by the Block Protect and TBPROT bits, are also hardware protected against data modification if WP# is Low during a WRR command.

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

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

### 3.12 Hold (HOLD#) / IO3

The Hold (HOLD#) signal is used to pause any serial communications with the device without deselecting the device or stopping the serial clock.

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

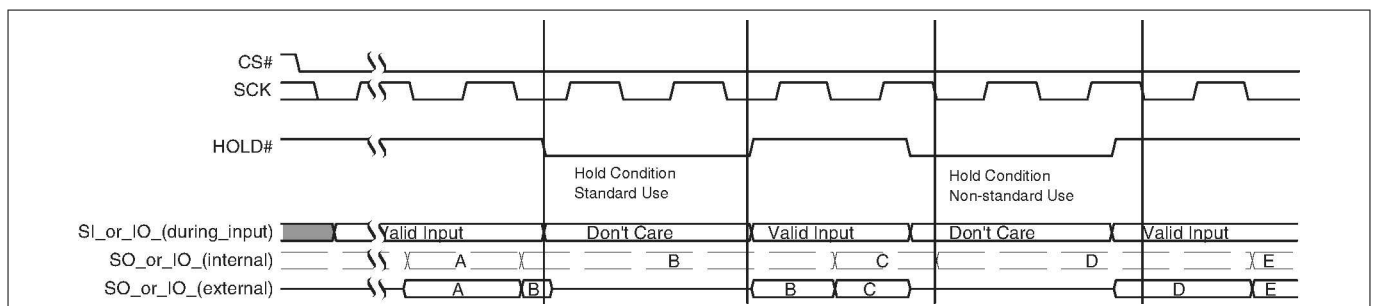
The Hold condition starts on the falling edge of the Hold (HOLD#) signal, provided that this coincides with SCK being at the logic low state. If the falling edge does not coincide with the SCK signal being at the logic low state, the Hold condition starts whenever the SCK signal reaches the logic low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

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

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

The HOLD# function is not available when the Quad mode is enabled (CR1[1] =1). The Hold function is replaced by IO3 for input and output during Quad mode for receiving addresses, and data to be programmed (values are latched on rising edge of the SCK signal) as well as shifting out data (on the falling edge of SCK, in SDR commands, and on every edge of SCK, in DDR commands).

The HOLD# signal has an internal pull-up resistor and may be left unconnected in the host system if not used for Quad mode.



**Figure 4 HOLD mode operation**

### 3.13 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 Pinouts and signal descriptions**

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

The Versatile I/O ( $V_{IO}$ ) supply is the voltage source for all device input receivers and output drivers and allows the host system to set the voltage levels that the device tolerates on all inputs and drives on outputs (address, control, and IO signals). The  $V_{IO}$  range is 1.65 V to  $V_{CC}$ .  $V_{IO}$  cannot be greater than  $V_{CC}$ .

For example, a  $V_{IO}$  of 1.65 V to 3.6 V allows for I/O at the 1.8 V, 2.5 V, or 3 V levels, driving and receiving signals to and from other 1.8 V, 2.5 V or 3 V devices on the same data bus.  $V_{IO}$  may be tied to  $V_{CC}$  so that interface signals operate at the same voltage as the core of the device.  $V_{IO}$  is not available in all package options, when not available the  $V_{IO}$  supply is tied to  $V_{CC}$  internal to the package.

During the rise of power supplies the  $V_{IO}$  supply voltage must remain less than or equal to the  $V_{CC}$  supply voltage. This supply is not available in all package options. For a backward compatible SO16 footprint, the  $V_{IO}$  supply is tied to  $V_{CC}$  inside the package; thus, the IO will function at  $V_{CC}$  level.

#### **3.15 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.

#### **3.16 Not Connected (NC)**

No device internal signal is connected to the package connector nor is there any future plan to use the connector for a signal. The connection may safely be used for routing space for a signal on a printed circuit board (PCB). However, any signal connected to an NC must not have voltage levels higher than  $V_{IO}$ .

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

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

#### **3.18 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 Pinouts and signal descriptions

3.19 Block diagrams

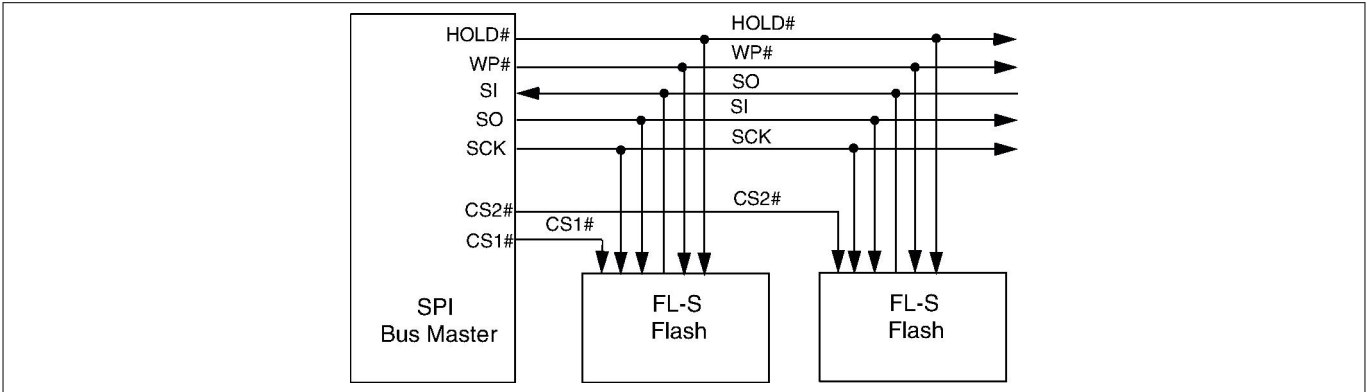


Figure 5 Bus master and memory devices on the SPI Bus - Single bit data path

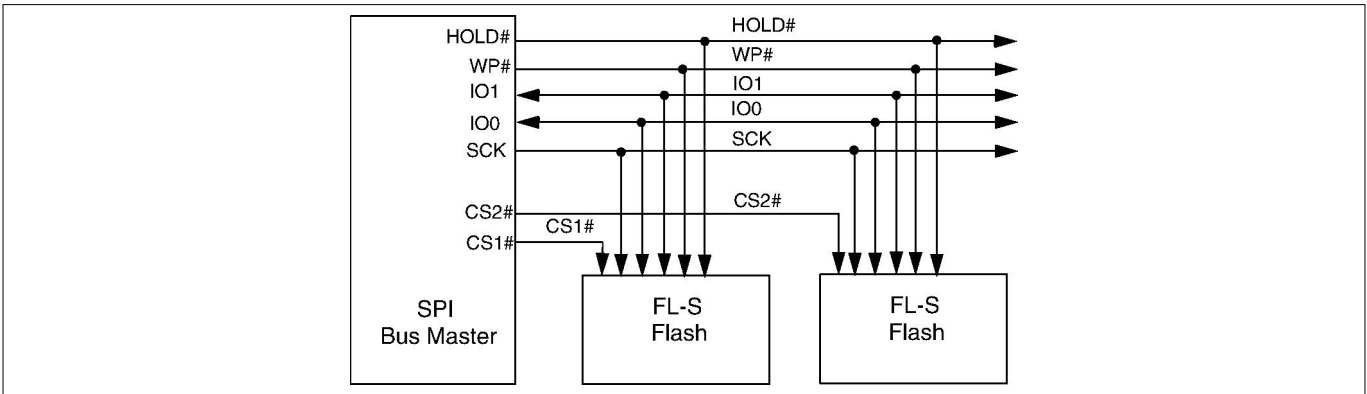


Figure 6 Bus master and memory devices on the SPI Bus - Dual bit data path

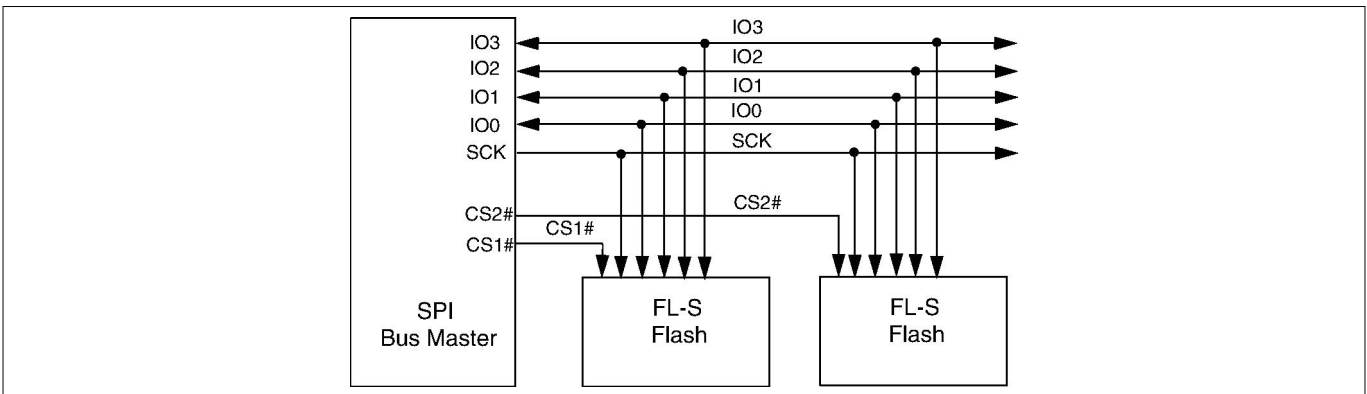


Figure 7 Bus master and memory devices on the SPI Bus - Quad bit data path

## 4 Signal protocols

### 4 Signal protocols

#### 4.1 SPI clock modes

##### 4.1.1 SDR

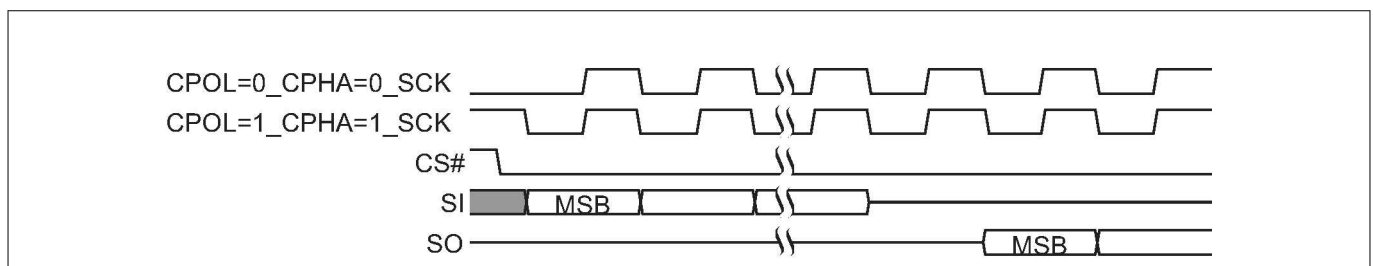
The S25FL128S and S25FL256S devices can be driven by an embedded microcontroller (bus master) in either of the two following clocking modes.

- **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 8 SPI SDR modes supported**

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

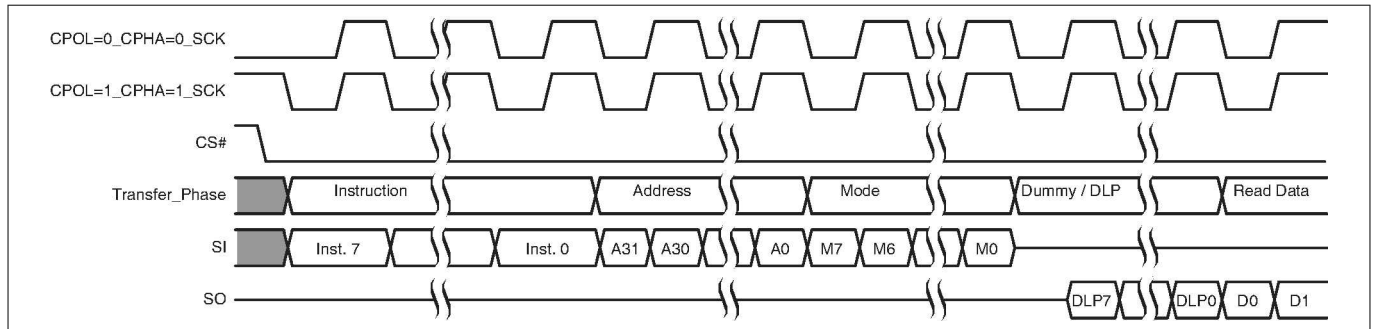
SCK cycles are measured (counted) from one falling edge of SCK to the next falling edge of SCK. In mode '0' the beginning of the first SCK cycle in a command is measured from the falling edge of CS# to the first falling edge of SCK because SCK is already low at the beginning of a command.

##### 4.1.2 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.

## 4 Signal protocols



**Figure 9 SPI DDR modes supported**

### 4.2 Command protocol

All communication between the host system and S25FL128S and S25FL256S memory devices is in the form of units called commands.

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

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

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

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

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

Commands are structured as follows:

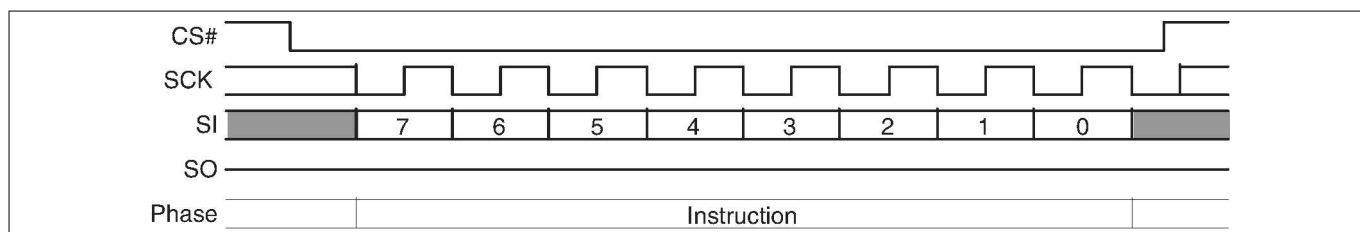
- Each command begins with CS# going LOW and ends with CS# returning HIGH. The memory device is selected by the host driving the Chip Select (CS#) signal low throughout a command.
- The serial clock (SCK) marks the transfer of each bit or group of bits between the host and memory.
- Each command begins with an eight bit (byte) instruction. The instruction is always presented only as a single bit serial sequence on the Serial Input (SI) signal with one bit transferred to the memory device on each SCK rising edge. The instruction selects the type of information transfer or device operation to be performed.
- The instruction may be stand alone or may be followed by address bits to select a 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.
- The width of all transfers following the instruction are determined by the instruction sent. Following transfers may continue to be single bit serial on only the SI or Serial Output (SO) signals, they may be done in two bit groups per (dual) transfer on the IO0 and IO1 signals, or they may be done in 4-bit groups per (quad) transfer on the IO0-IO3 signals. Within the dual or quad groups the least significant bit is on IO0. More significant bits are placed in significance order on each higher numbered IO signal. Single bits or parallel bit groups are transferred in most to least significant bit order.
- Some instructions send an instruction modifier called mode bits, following the address, to indicate that the next command will be of the same type with an implied, rather than an explicit, instruction. The next command thus does not provide an instruction byte, only a new address and mode bits. This reduces

## 4 Signal protocols

the time needed to send each command when the same command type is repeated in a sequence of commands. The mode bit transfers occur on SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.

- The address or mode bits may be followed by write data to be stored in the memory device or by a read latency period before read data is returned to the host.
- Write data bit transfers occur on SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- SCK continues to toggle during any read access latency period. The latency may be zero to several SCK cycles (also referred to as dummy cycles). At the end of the read latency cycles, the first read data bits are driven from the outputs on SCK falling edge at the end of the last read latency cycle. The first read data bits are considered transferred to the host on the following SCK rising edge. Each following transfer occurs on the next SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- If the command returns read data to the host, the device continues sending data transfers until the host takes the CS# signal high. The CS# signal can be driven high after any transfer in the read data sequence. This will terminate the command.
- At the end of a command that does not return data, the host drives the CS# input high. The CS# signal must go high after the eighth bit, of a stand alone instruction or, of the last write data byte that is transferred. That is, the CS# signal must be driven high when the number of 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 10 Stand Alone Instruction command**

4 Signal protocols

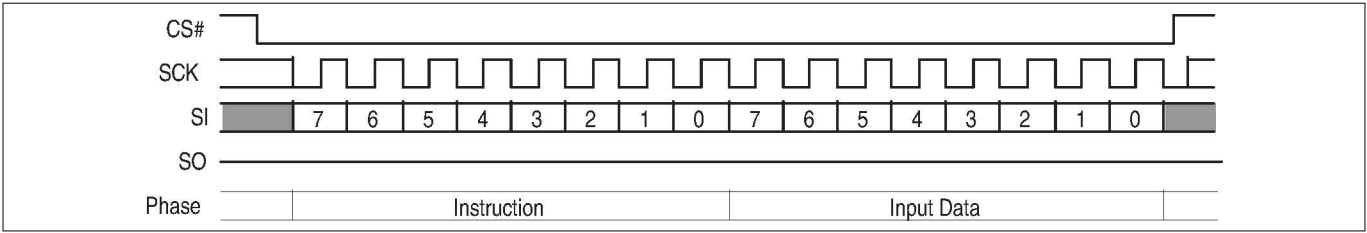


Figure 11 Single Bit Wide Input command

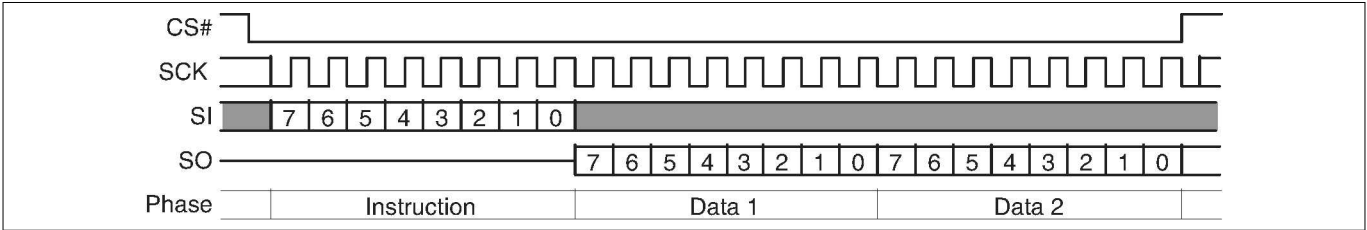


Figure 12 Single Bit Wide Output command

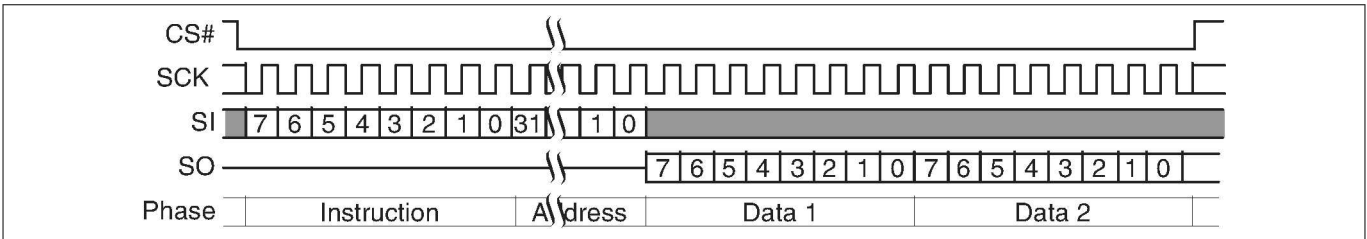


Figure 13 Single Bit Wide I/O command without latency

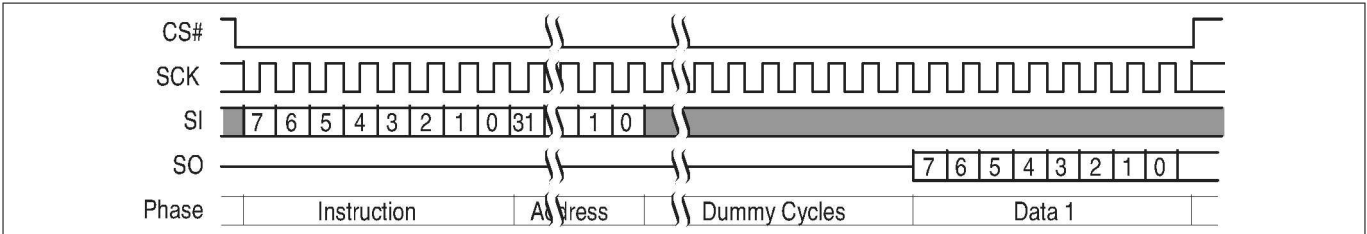


Figure 14 Single Bit Wide I/O command with latency

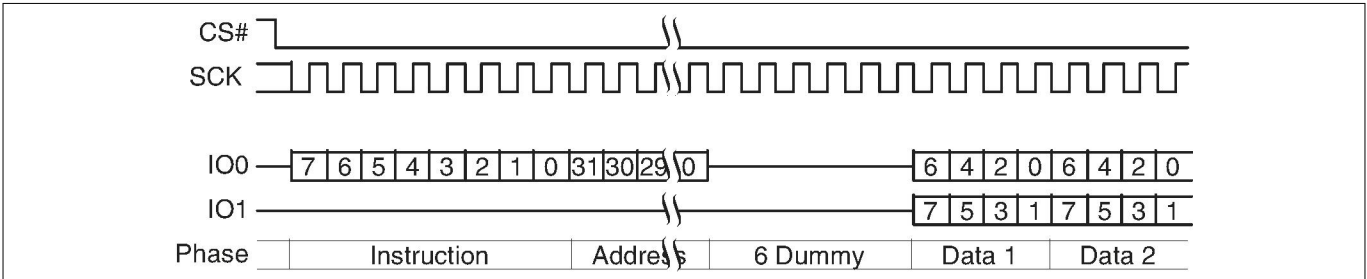
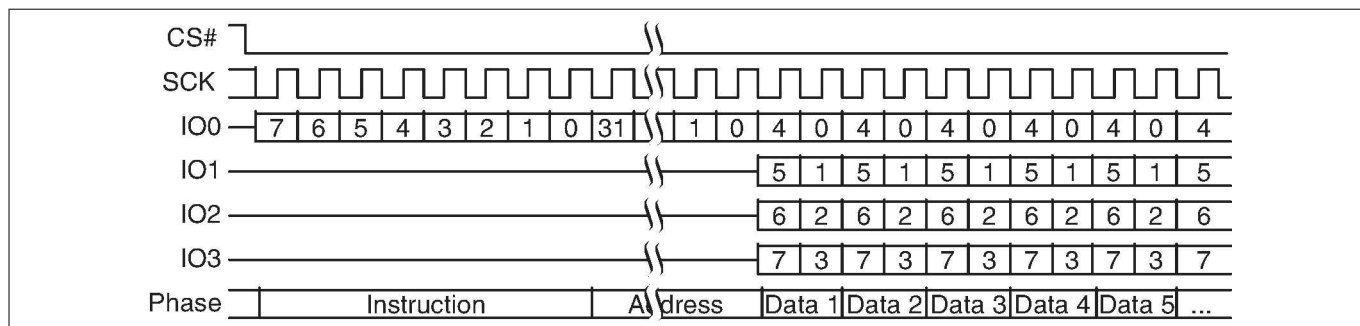
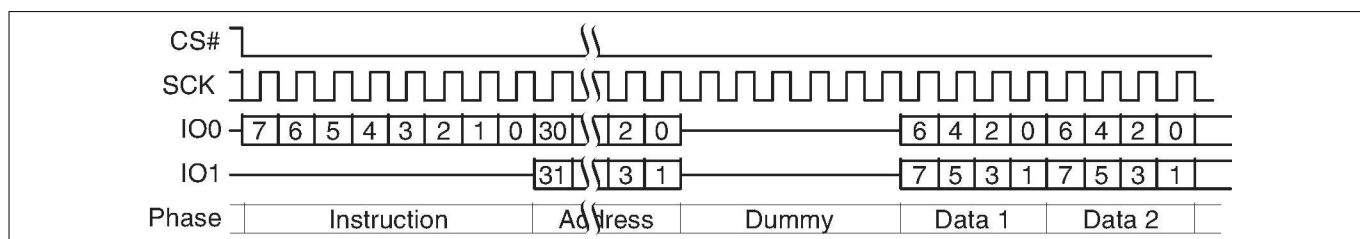


Figure 15 Dual Output command

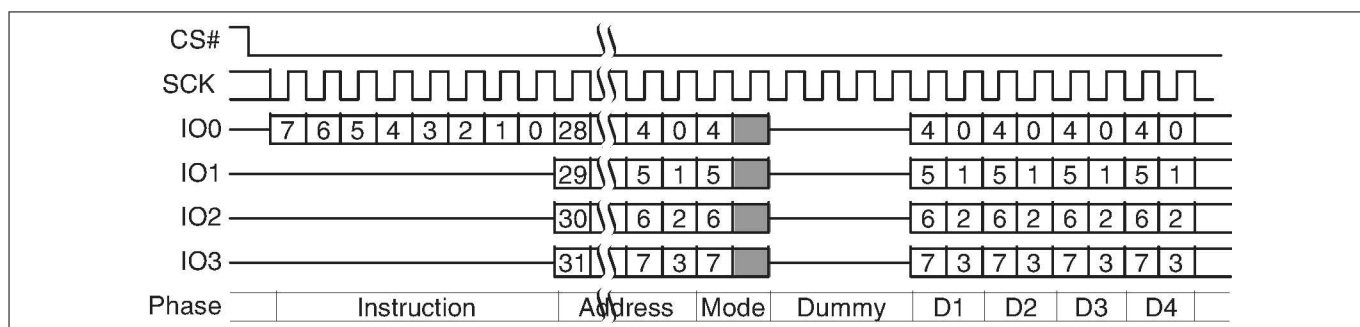
#### 4 Signal protocols



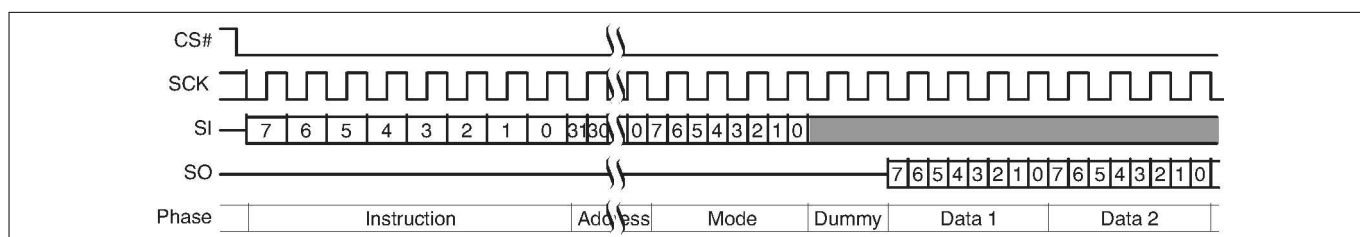
**Figure 16 Quad Output command without latency**



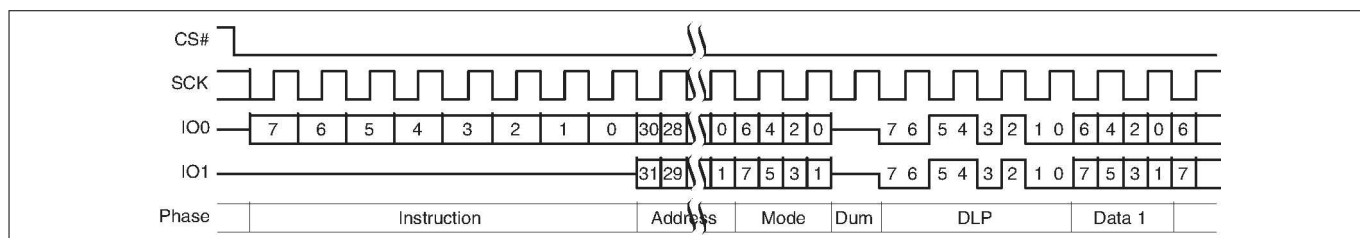
**Figure 17 Dual I/O command**



**Figure 18 Quad I/O command**

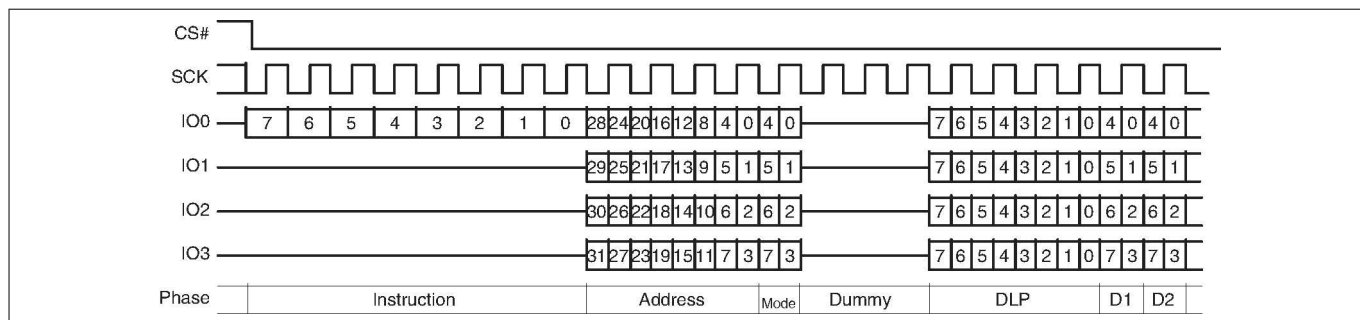


**Figure 19 DDR Fast Read with EHPLC = 00b**



**Figure 20 DDR Dual I/O Read with EHPLC = 01b and DLP**

## 4 Signal protocols



**Figure 21** DDR Quad I/O Read

Additional sequence diagrams, specific to each command, are provided in [Commands](#).

### 4.3 Interface states

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

**Table 8** Interface states summary

Interface state	V <sub>CC</sub>	V <sub>IO</sub>	RESET#	SCK	CS#	HOLD# / IO3	WP# / IO2	SO/IO1	SI/IO0
Power-off	< V <sub>CC</sub> (low)	≤ V <sub>CC</sub>	X	X	X	X	X	Z	X
Low power hardware data protection	< V <sub>CC</sub> (cut-off)	≤ V <sub>CC</sub>	X	X	X	X	X	Z	X
Power-on (cold) reset	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	X	X	X	X	X	Z	X
Hardware (warm) reset	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HL	X	X	X	X	Z	X
Interface standby	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	X	HH	X	X	Z	X
Instruction cycle	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	HH	HV	Z	HV
Hold cycle	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HV or HT	HL	HL	X	X	X

(table continues...)

#### 4 Signal protocols

**Table 8** (continued) Interface states summary

Interface state	V <sub>CC</sub>	V <sub>IO</sub>	RESET#	SCK	CS#	HOLD# / IO3	WP# / IO2	SO/IO1	SI/IO0
Single input cycle Host to Memory transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	HH	X	Z	HV
Single latency (dummy) cycle	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	HH	X	Z	X
Single output cycle Memory to Host transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	HH	X	MV	X
Dual input cycle Host to Memory transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	HH	X	HV	HV
Dual latency (dummy) cycle	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	HH	X	X	X
Dual output cycle Memory to Host transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	HH	X	MV	MV
QPP address input cycle Host to Memory transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	X	X	X	HV

(table continues...)



#### 4 Signal protocols

**Table 8** (continued) Interface states summary

Interface state	V <sub>CC</sub>	V <sub>IO</sub>	RESET#	SCK	CS#	HOLD# / IO3	WP# / IO2	SO/IO1	SI/IO0
Quad input cycle Host to Memory transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	HV	HV	HV	HV
Quad latency (dummy) cycle	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	X	X	X	X
Quad output cycle Memory to Host transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	MV	MV	MV	MV
DDR single input cycle Host to Memory transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	X	X	X	HV
DDR dual input cycle Host to Memory transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	X	X	HV	HV
DDR Quad input cycle Host to Memory transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	HV	HV	HV	HV
DDR latency (dummy) cycle	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	MV or Z	MV or Z	MV or Z	MV or Z

(table continues...)

## 4 Signal protocols

**Table 8** (continued) Interface states summary

Interface state	V <sub>CC</sub>	V <sub>IO</sub>	RESET#	SCK	CS#	HOLD# / IO3	WP# / IO2	SO/IO1	SI/IO0
DDR single output cycle Memory to Host transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	Z	Z	MV	X
DDR dual output cycle Memory to Host transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	Z	Z	MV	MV
DDR Quad output cycle Memory to Host transfer	≥ V <sub>CC</sub> (min)	≥ V <sub>IO</sub> (min) ≤ V <sub>CC</sub>	HH	HT	HL	MV	MV	MV	MV

### Legend

Z = No driver - floating signal

HL = Host driving V<sub>IL</sub>

HH = Host driving V<sub>IH</sub>

HV = either HL or HH

X = HL or HH or Z

HT = toggling between HL and HH

ML = Memory driving V<sub>IL</sub>

MH = Memory driving V<sub>IH</sub>

MV = either ML or MH

### 4.3.1 Power-off

When the core supply voltage is at or below the V<sub>CC</sub> (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<sub>CC</sub> is less than V<sub>CC</sub> (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 Signal protocols

### 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  $\geq V_{CC}$  (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](#).

### 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 [Figure 31](#).

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

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

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

### 4.3.7 Hold

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

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

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

During DDR commands the HOLD# and WP# inputs are ignored.

## **4 Signal protocols**

### **4.3.8 Single input cycle - Host to Memory transfer**

Several commands transfer information after the instruction on the single serial input (SI) signal from host to the memory device. The dual output, and quad output commands send address to the memory using only SI but return read data using the I/O signals. The host keeps RESET# high, CS# low, HOLD# high, and drives SI as needed for the command. The memory does not drive the Serial Output (SO) signal.

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

### **4.3.9 Single latency (dummy) cycle**

Read commands may have zero to several latency cycles during which read data is read from the main flash memory array before transfer to the host. The number of latency cycles are determined by the Latency Code in the configuration register (CR[7:6]). During the latency cycles, the host keeps RESET# high, CS# LOW, and HOLD# HIGH. The Write Protect (WP#) signal is ignored. The host may drive the SI signal during these cycles or the host may leave SI floating. The memory does not use any data driven on SI / I/O0 or other I/O signals during the latency cycles. In dual or quad read commands, the host must stop driving the I/O signals on the falling edge at the end of the last latency cycle. It is recommended that the host stop driving I/O signals during latency cycles so that there is sufficient time for the host drivers to turn off before the memory begins to drive at the end of the latency cycles. This prevents driver conflict between host and memory when the signal direction changes. The memory does not drive the Serial Output (SO) or I/O signals during the latency cycles.

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

### **4.3.10 Single output cycle - Memory to Host transfer**

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

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

### **4.3.11 Dual input cycle - Host to Memory transfer**

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

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

### **4.3.12 Dual latency (dummy) cycle**

Read commands may have zero to several latency cycles during which read data is read from the main flash memory array before transfer to the host. The number of latency cycles are determined by the Latency Code in the configuration register (CR[7:6]). During the latency cycles, the host keeps RESET# high, CS# low, and HOLD# high. The Write Protect (WP#) signal is ignored. The host may drive the SI / IO0 and SO / IO1 signals during these cycles or the host may leave SI / IO0 and SO / IO1 floating. The memory does not use any data driven on SI / IO0 and SO / IO1 during the latency cycles. The host must stop driving SI / IO0 and SO / IO1 on the falling edge at the end of the last latency cycle. It is recommended that the host stop driving them during all latency cycles so that there is sufficient time for the host drivers to turn off before the memory begins to drive at the end of the

## **4 Signal protocols**

latency cycles. This prevents driver conflict between host and memory when the signal direction changes. The memory does not drive the SI / IO0 and SO / IO1 signals during the latency cycles.

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

### **4.3.13 Dual output cycle - Memory to Host transfer**

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

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

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

The Quad Page Program and Quad Output Read commands send address to the memory only on IO0. The other IO signals are ignored 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.

### **4.3.15 Quad input cycle - Host to Memory transfer**

The Quad I/O Read command transfers four address or mode bits to the memory in each cycle. The Quad Page Program command transfers four data bits to the memory in each cycle. The host keeps RESET# high, CS# low, and drives the IO signals.

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

### **4.3.16 Quad latency (dummy) cycle**

Read commands may have zero to several latency cycles during which read data is read from the main flash memory array before transfer to the host. The number of latency cycles are determined by the Latency Code in the configuration register (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.

### **4.3.17 Quad output cycle - Memory to Host transfer**

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

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

## **4 Signal protocols**

### **4.3.18 DDR single input cycle - Host to Memory transfer**

The DDR Fast Read command sends address, and mode bits to the memory only on the IO0 signal. One bit is transferred on the rising edge of SCK and one bit on the falling edge in each cycle. The host keeps RESET# high, and CS# low. The other IO signals are ignored by the memory.

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

### **4.3.19 DDR dual input cycle - Host to Memory transfer**

The DDR Dual I/O Read command sends address, and mode bits to the memory only on the IO0 and IO1 signals. Two bits are transferred on the rising edge of SCK and two bits on the falling edge in each cycle. The host keeps RESET# high, and CS# low. The IO2 and IO3 signals are ignored by the memory.

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

### **4.3.20 DDR Quad input cycle - Host to Memory transfer**

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

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

### **4.3.21 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 (CR[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 4 latency cycles. The DLP option should not be enabled when there are fewer than five latency cycles so that there is at least one cycle of high impedance for turn around of the IO signals before the memory begins driving the DLP. When there are more than 4 cycles of latency the memory does not drive the IO signals until the last four cycles of latency.

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

### **4.3.22 DDR single output cycle - Memory to Host transfer**

The DDR Fast Read command returns bits to the host only on the SO/IO1 signal. One bit is transferred on the rising edge of SCK and one bit on the falling edge in each cycle. The host keeps RESET# high, and CS# low. The other IO signals are not driven by the memory.

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

### **4.3.23 DDR dual output cycle - Memory to Host transfer**

The DDR Dual I/O Read command returns bits to the host only on the IO0 and IO1 signals. Two bits are transferred on the rising edge of SCK and two bits on the falling edge in each cycle. The host keeps RESET# high, and CS# low. The IO2 and IO3 signals are not driven by the memory.

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

## **4 Signal protocols**

### **4.3.24 DDR Quad output cycle - Memory to Host transfer**

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

### **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.

#### **4.5.1 Power-up**

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

#### **4.5.2 Low power**

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

### 5.1 Absolute maximum ratings

**Table 9 Absolute maximum ratings**

Parameter	Value
Storage temperature plastic packages	−65°C to +150°C
Ambient temperature with power applied	−65°C to +125°C
$V_{CC}$	−0.5 V to +4.0 V
$V_{IO}$ <sup>1)</sup>	−0.5 V to +4.0 V
Input voltage with respect to ground ( $V_{SS}$ ) <sup>2)</sup>	−0.5 V to +(V <sub>IO</sub> + 0.5 V)
Output short circuit current <sup>3)</sup>	100 mA

1)  $V_{IO}$  must always be less than or equal  $V_{CC} + 200$  mV.

2) See [Input signal overshoot](#) for allowed maximums during signal transition.

3) 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.

#### Notes:

- Stresses above those listed under [Table 9](#) may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational sections of this data sheet is not implied. Exposure of the device to absolute maximum rating conditions for extended periods may affect device reliability.

### 5.2 Thermal resistance

**Table 10 Thermal resistance**

Parameter	Description	WNG008	FAB024		Unit
		FL256S	FL128S	FL256S	
Theta JA	Junction to ambient	28	39	34.5	°C/W
Theta JB	Junction to board	11.7	23.9	18.3	
Theta JC	Junction to case	13.1	14	10.3	

### 5.3 Operating ranges

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

#### 5.3.1 Power supply voltages

Some package options provide access to a separate input and output buffer power supply called  $V_{IO}$ . Packages which do not provide the separate  $V_{IO}$  connection, internally connect the device  $V_{IO}$  to  $V_{CC}$ . For these packages the references to  $V_{IO}$  are then also references to  $V_{CC}$ .



## 5 Electrical specifications

**Table 11** Power supply voltages

Parameter	Value
$V_{CC}$	2.7 V to 3.6 V
$V_{IO}$	1.65 V to $V_{CC} + 200$ mV

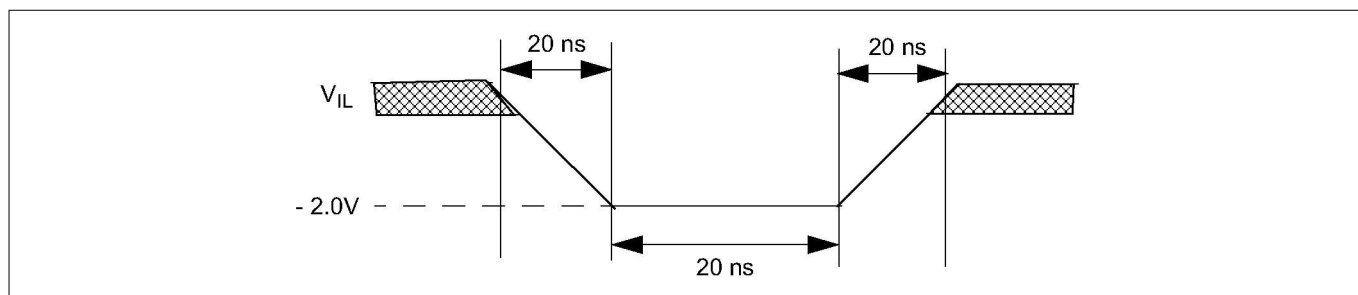
### 5.3.2 Temperature ranges

**Table 12** Temperature ranges

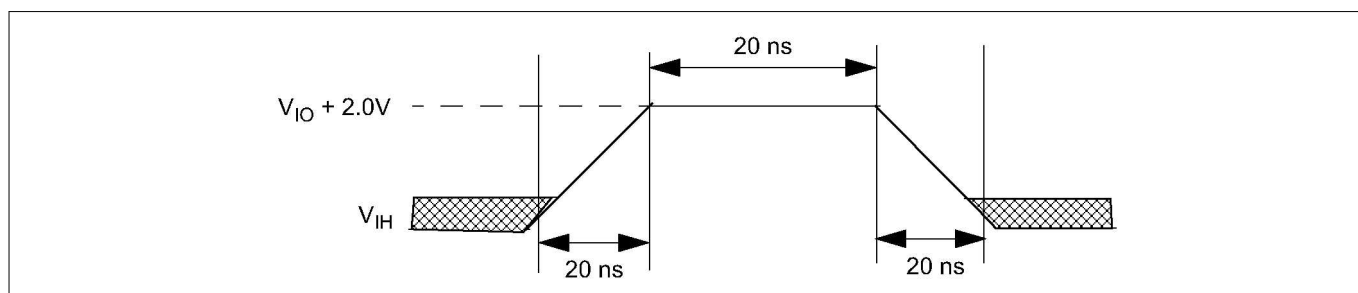
Parameter	Symbol	Device	Spec		Unit
			Min	Max	
Ambient temperature	$T_A$	Military (E)	-55	+125	°C

### 5.3.3 Input signal overshoot

During DC conditions, input or I/O signals should remain equal to or between  $V_{SS}$  and  $V_{IO}$ . During voltage transitions, inputs or I/Os may overshoot  $V_{SS}$  to  $-2.0$  V or overshoot to  $V_{IO} + 2.0$  V, for periods up to 20 ns.



**Figure 22** Maximum negative overshoot waveform



**Figure 23** Maximum positive overshoot waveform

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

## 5 Electrical specifications

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 24. However, correct operation of the device is not guaranteed if  $V_{CC}$  returns below  $V_{CC}(\text{min})$  during  $t_{PU}$ . No command should be sent to the device until the end of  $t_{PU}$ .

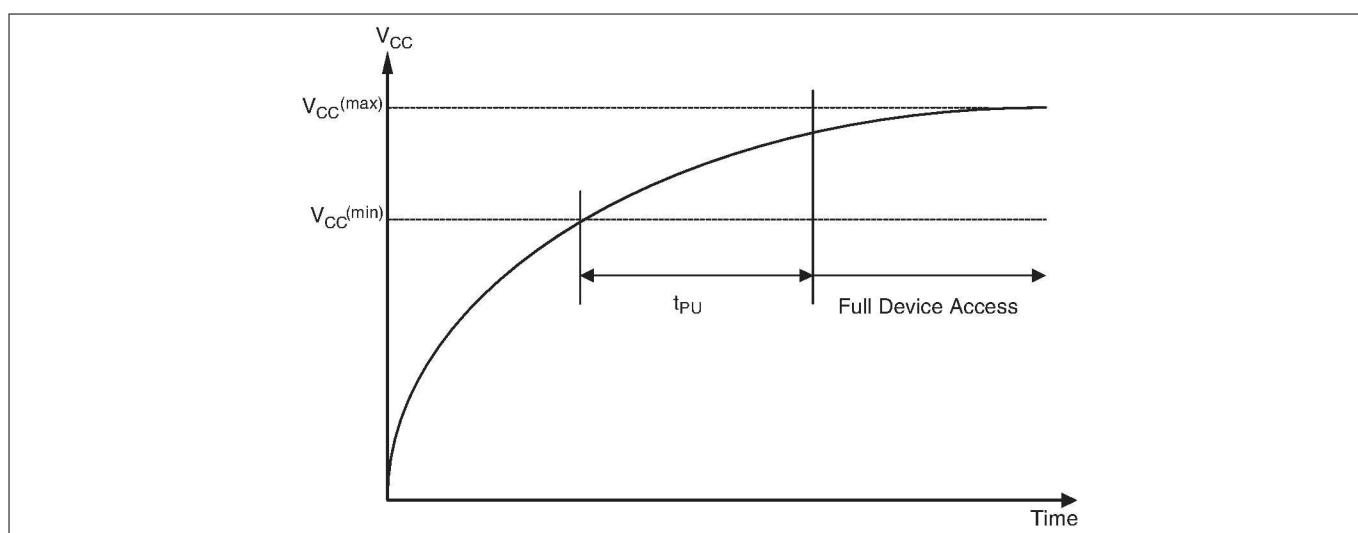
After power-up ( $t_{PU}$ ), the device is in Standby mode (not Deep Power Down mode), draws CMOS standby current ( $I_{SB}$ ), and the WEL bit is reset.

During power-down or voltage drops below  $V_{CC}$  (cut-off), the voltage must drop below  $V_{CC}(\text{low})$  for a period of  $t_{PD}$  for the part to initialize correctly on power-up. See Figure 25. If during a voltage drop the  $V_{CC}$  stays above  $V_{CC}(\text{cut-off})$  the part will stay initialized and will work correctly when  $V_{CC}$  is again above  $V_{CC}(\text{min})$ . In the event Power-on Reset (POR) did not complete correctly after power up, the assertion of the RESET# signal or receiving a software reset command (RESET) will restart the POR process.

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

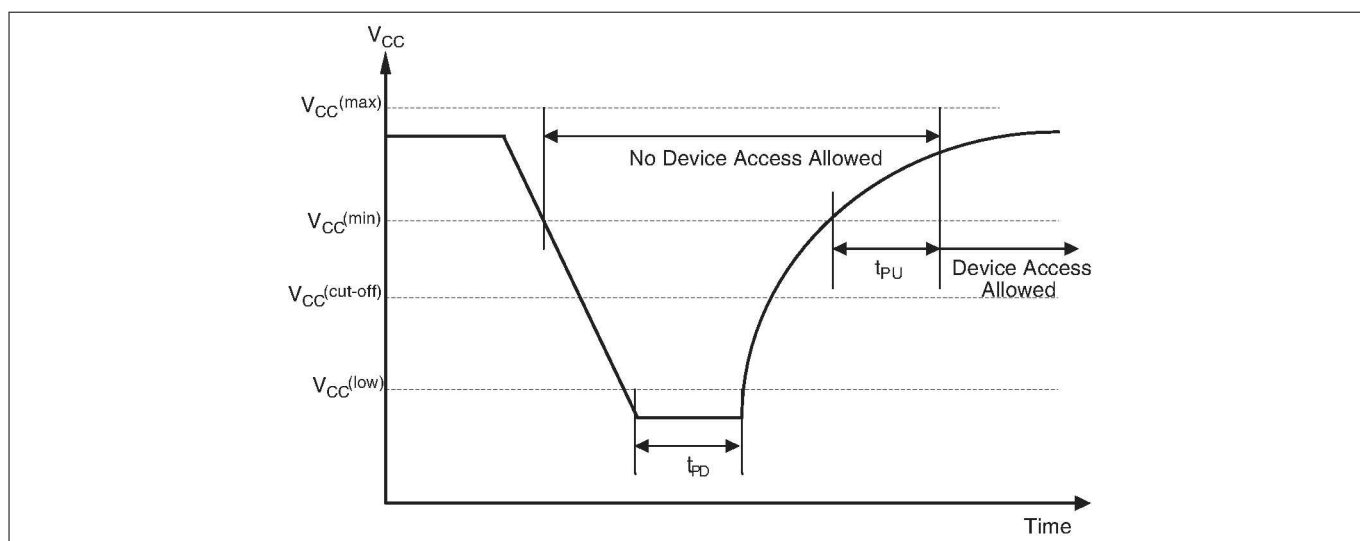
**Table 13 Power-up/power-down voltage and timing**

Symbol	Parameter	Min	Max	Unit
$V_{CC}(\text{min})$	$V_{CC}$ (Minimum operation voltage)	2.7		V
$V_{CC}(\text{cut-off})$	$V_{CC}$ (Cut off where re-initialization is needed)	2.4		V
$V_{CC}(\text{low})$	$V_{CC}$ (Low voltage for initialization to occur) $V_{CC}$ (Low voltage for Initialization to occur at embedded)	1.6 2.3		V
$t_{PU}$	$V_{CC}(\text{min})$ to read operation		300	$\mu\text{s}$
$t_{PD}$	$V_{CC}(\text{low})$ time	15.0		$\mu\text{s}$



**Figure 24 Power-up**

## 5 Electrical specifications



**Figure 25** Power-down and voltage drop

### 5.5 DC characteristics

Applicable within operating ranges.

**Table 14** DC characteristics

Symbol	Parameter	Test conditions	Min	Typ <sup>1)</sup>	Max	Unit
$V_{IL}$	Input low voltage	–	-0.5	–	$0.2 \times V_{IO}$	V
$V_{IH}$	Input high voltage	–	$0.7 \times V_{IO}$	–	$V_{IO} + 0.4$	V
$V_{OL}$	Output low voltage	$I_{OL} = 1.6 \text{ mA}$ , $V_{CC} = V_{CC} \text{ min}$	–	–	$0.15 \times V_{IO}$	V
$V_{OH}$	Output high voltage	$I_{OH} = -0.1 \text{ mA}$	$0.85 \times V_{IO}$	–		V
$I_{LI}$	Input leakage current	$V_{CC} = V_{CC} \text{ Max}$ , $V_{IN} = V_{IH} \text{ or } V_{IL}$	–	–	$\pm 2$	$\mu\text{A}$
$I_{LO}$	Output leakage current	$V_{CC} = V_{CC} \text{ Max}$ , $V_{IN} = V_{IH} \text{ or } V_{IL}$	–	–	$\pm 2$	$\mu\text{A}$

(table continues...)

## 5 Electrical specifications

**Table 14** (continued) DC characteristics

Symbol	Parameter	Test conditions	Min	Typ <sup>1)</sup>	Max	Unit
$I_{CC1}$	Active power supply current (READ)	Serial SDR@50 MHz Serial SDR@133 MHz Quad SDR@80 MHz Quad SDR@104 MHz Quad DDR@66 MHz Quad DDR@80 MHz Outputs unconnected during read data return <sup>2)</sup>	–	–	22 35 50 61 75 90	mA
$I_{CC2}$	Active power supply current (Page Program)	CS# = $V_{IO}$	–	–	100	mA
$I_{CC3}$	Active power supply current (WRR)	CS# = $V_{IO}$	–	–	100	mA
$I_{CC4}$	Active power supply current (SE)	CS# = $V_{IO}$	–	–	100	mA
$I_{CC5}$	Active power supply current (BE)	CS# = $V_{IO}$	–	–	100	mA
$I_{SB}$	Standby current	RESET#, CS# = $V_{IO}$ ; SI, SCK = $V_{IO}$ or $V_{SS}$	–	115	300	$\mu$ A

1) Typical values are at  $T_{AI} = 25^{\circ}\text{C}$  and  $V_{CC} = V_{IO} = 3\text{ V}$ .

2) Output switching current is not included.

### 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  $I_{SB}$ .

## 6 Timing specifications

### 6 Timing specifications

#### 6.1 Key to switching waveforms

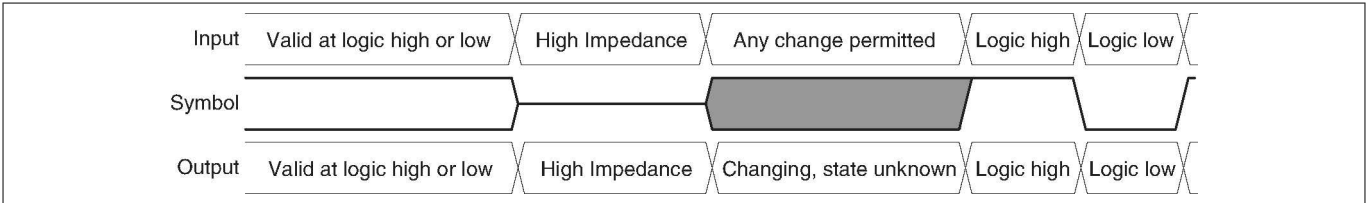


Figure 26 Waveform element meanings

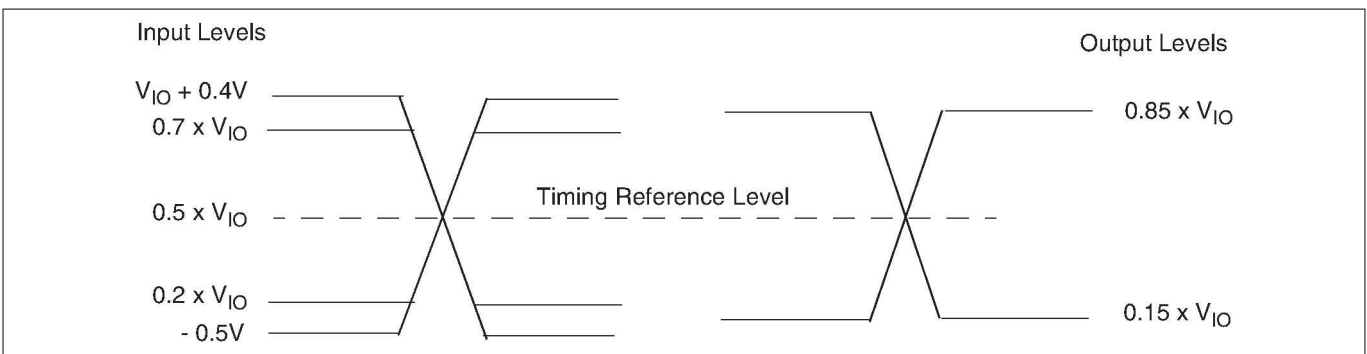


Figure 27 Input, output, and timing reference levels

#### 6.2 AC test conditions

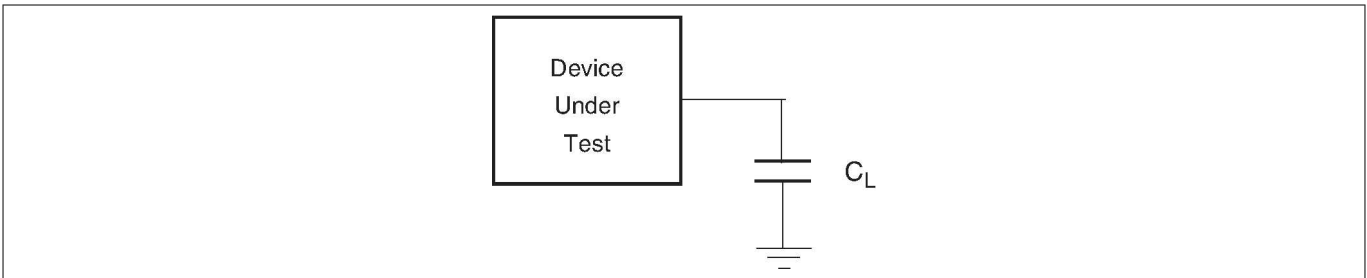


Figure 28 Test setup

Table 15 AC measurement conditions

Symbol	Parameter	Min	Max	Unit
$C_L$	Load capacitance	30 15 <sup>1)</sup>		pF
	Input rise and fall times	–	2.4	ns
	Input pulse voltage	0.2 x $V_{IO}$ to 0.8 $V_{IO}$		V
	Input timing ref voltage	0.5 $V_{IO}$		V
	Output timing ref voltage	0.5 $V_{IO}$		V

## 6 Timing specifications

1) DDR Operation.

### Notes:

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

### 6.2.1 Capacitance characteristics

**Table 16** Capacitance

Symbol	Parameter	Test conditions	Min	Max	Unit
$C_{IN}$	Input capacitance (applies to SCK, CS#, RESET#)	1 MHz	–	8	pF
$C_{OUT}$	Output capacitance (applies to All I/O)	1 MHz	–	8	pF

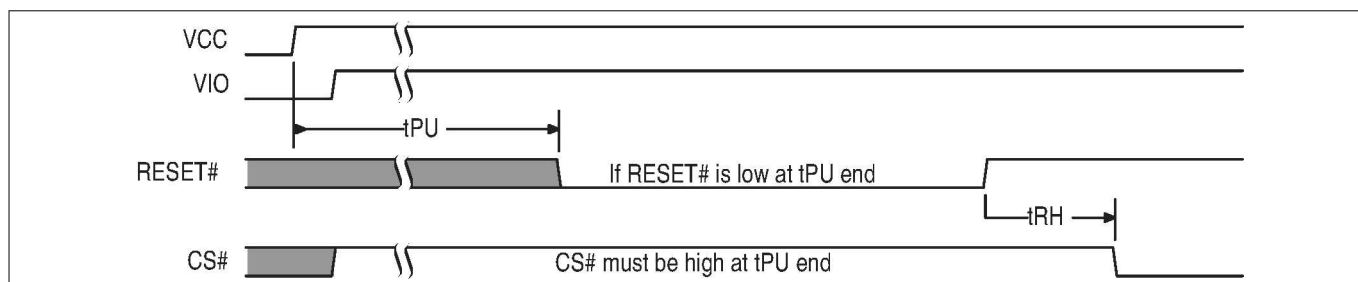
### Notes:

1. For more information on capacitance, please consult the IBIS models.

## 6.3 Reset

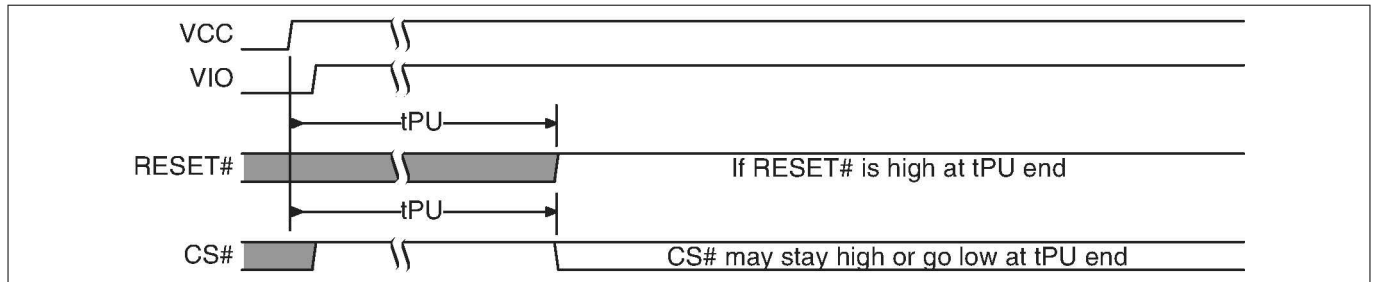
### 6.3.1 Power-on (Cold) reset

The device executes a Power-On Reset (POR) process until a time delay of  $t_{PU}$  has elapsed after the moment that  $V_{CC}$  rises above the minimum  $V_{CC}$  threshold. See Figure 24, Table 13, and Table 17. The device must not be selected (CS# to go high with VIO) 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.

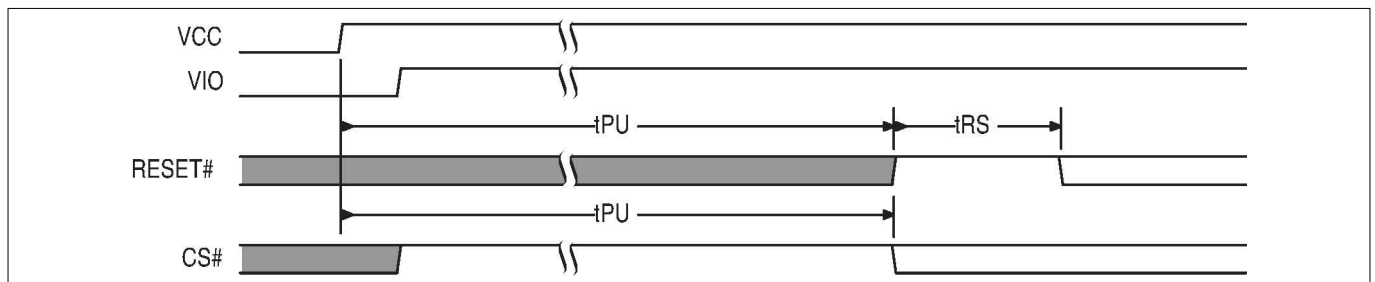


**Figure 29** Reset low at the end of POR

## 6 Timing specifications



**Figure 30** Reset high at the end of POR



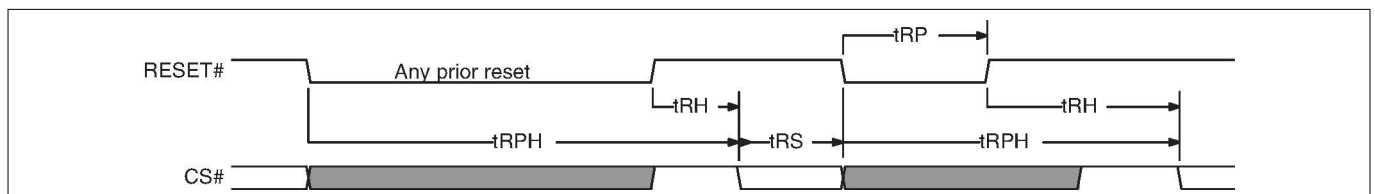
**Figure 31** POR followed by hardware reset

### 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  $t_{RH}$ .
- Hardware reset is only offered in 16-lead SOIC and BGA packages.



**Figure 32** Hardware reset

## 6 Timing specifications

**Table 17 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

**Note:**

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

## 6.4 SDR AC characteristics

**Table 18 AC characteristics (single die package,  $V_{IO} = V_{CC}$  2.7 V to 3.6 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 53<sup>1)</sup></a>	DC	–	133	MHz
$F_{SCK, C}$	SCK clock frequency for the following dual and quad commands: DOR, 4DOR, QOR, 4QOR, DIOR, 4DIOR, QIOR, 4QIOR	DC	–	104	MHz
$F_{SCK, QPP}$	SCK clock frequency for the QPP, 4QPP commands	DC	–	80	MHz

(table continues...)



## 6 Timing specifications

**Table 18** (continued) AC characteristics (single die package,  $V_{IO} = V_{CC}$  2.7 V to 3.6 V)

Symbol	Parameter	Min	Typ	Max	Unit
$P_{SCK}$	SCK clock period	$1 / F_{SCK}$	–	$\infty$	–
$t_{WH}, t_{CH}$	Clock high time <sup>2)</sup>	45% $P_{SCK}$	–	–	ns
$t_{WL}, t_{CL}$	Clock low time <sup>2)</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	–	3000 <sup>3)</sup>	ns
$t_{SU}$	Data in setup time	2	–	–	ns
$t_{HD}$	Data in hold time	2	–	–	ns
$t_V$	Clock low to output valid	–	–	8.0 <sup>4)</sup> 7.65 <sup>5)</sup> 6.5 <sup>1)</sup>	ns
$t_{HO}$	Output hold time	2	–	–	ns
$t_{DIS}$	Output disable time	0	–	8	ns
$t_{WPS}$	WP# setup time	20 <sup>6)</sup>	–	–	ns
$t_{WPH}$	WP# hold time	100 <sup>6)</sup>	–	–	ns
$t_{HLCH}$	HOLD# active setup time (relative to SCK)	3	–	–	ns
$t_{CHHH}$	HOLD# active hold time (relative to SCK)	3	–	–	ns

(table continues...)

## 6 Timing specifications

**Table 18** (continued) AC characteristics (single die package,  $V_{IO} = V_{CC}$  2.7 V to 3.6 V)

Symbol	Parameter	Min	Typ	Max	Unit
$t_{HHCH}$	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

- 1) Regulated  $V_{CC}$  range (3.0 to 3.6 V) and  $CL = 15$  pF.
- 2)  $\pm 10\%$  duty cycle is supported for frequencies  $\leq 50$  MHz.
- 3) Maximum value only applies during Program/Erase Suspend/Resume commands.
- 4) Full  $V_{CC}$  range (2.7 to 3.6 V) and  $CL = 30$  pF.
- 5) Regulated  $V_{CC}$  range (3.0 to 3.6 V) and  $CL = 30$  pF.
- 6) Only applicable as a constraint for WRR instruction when SRWD is set to a 1.

**Table 19** AC characteristics (single die package,  $V_{IO}$  1.65 V to 2.7 V,  $V_{CC}$  2.7 V to 3.6 V)

Symbol	Parameter	Min	Typ	Max	Unit
$F_{SCK, R}$	SCK clock frequency for READ, 4READ instructions	DC		50	MHz
$F_{SCK, C}$	SCK clock frequency for all others <sup>2)</sup>	DC	–	66	MHz
$P_{SCK}$	SCK clock period	$1/F_{SCK}$	–	$\infty$	–
$t_{WH}, t_{CH}$	Clock high time <sup>3)</sup>	45% $P_{SCK}$	–	–	ns
$t_{WL}, t_{CL}$	Clock low time <sup>3)</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)	10	–	–	ns

(table continues...)

## 6 Timing specifications

**Table 19** (continued) AC characteristics (single die package,  $V_{IO}$  1.65 V to 2.7 V,  $V_{CC}$  2.7 V to 3.6 V)

Symbol	Parameter	Min	Typ	Max	Unit
$t_{CSH}$	CS# active hold time (relative to SCK)	3	–	3000 <sup>1)</sup>	ns
$t_{SU}$	Data in setup time	5	–	–	ns
$t_{HD}$	Data in hold time	4	–	–	ns
$t_V$	Clock low to output valid	–	–	14.5 <sup>2)</sup> 12.0 <sup>3)</sup>	ns
$t_{HO}$	Output hold time	2	–	–	ns
$t_{DIS}$	Output disable time	0	–	14	ns
$t_{WPS}$	WP# setup time	20 <sup>6)</sup>	–	–	ns
$t_{WPH}$	WP# hold time	100 <sup>6)</sup>	–	–	ns
$t_{HLCH}$	HOLD# active setup time (relative to SCK)	5	–	–	ns
$t_{CHHH}$	HOLD# active hold time (relative to SCK)	5	–	–	ns
$t_{HHCH}$	HOLD# non active setup time (relative to SCK)	5	–	–	ns
$t_{CHHL}$	HOLD# non active hold time (relative to SCK)	5	–	–	ns
$t_{HZ}$	HOLD# enable to output invalid	–	–	14	ns
$t_{LZ}$	HOLD# disable to output valid	–	–	14	ns

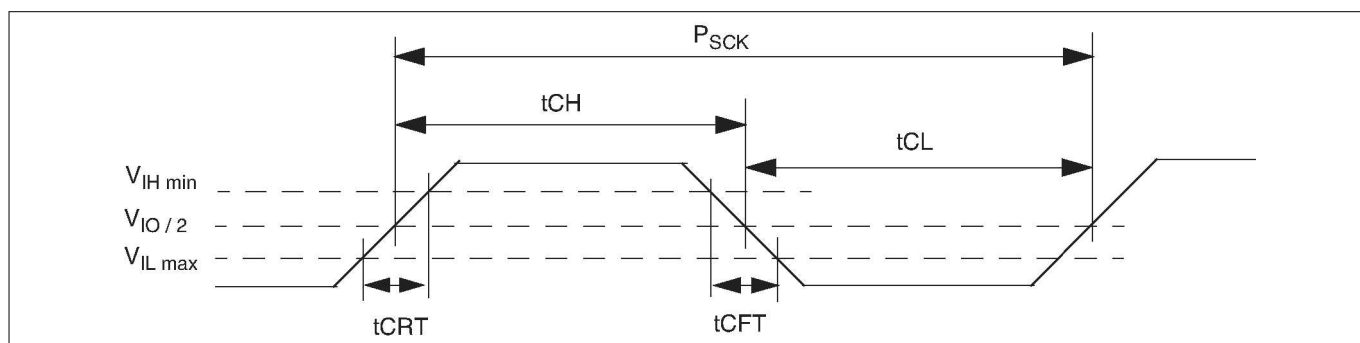
1) Regulated  $V_{CC}$  range (3.0 to 3.6 V) and  $CL = 15$  pF.

2)  $CL = 30$  pF.

3)  $CL = 15$  pF.

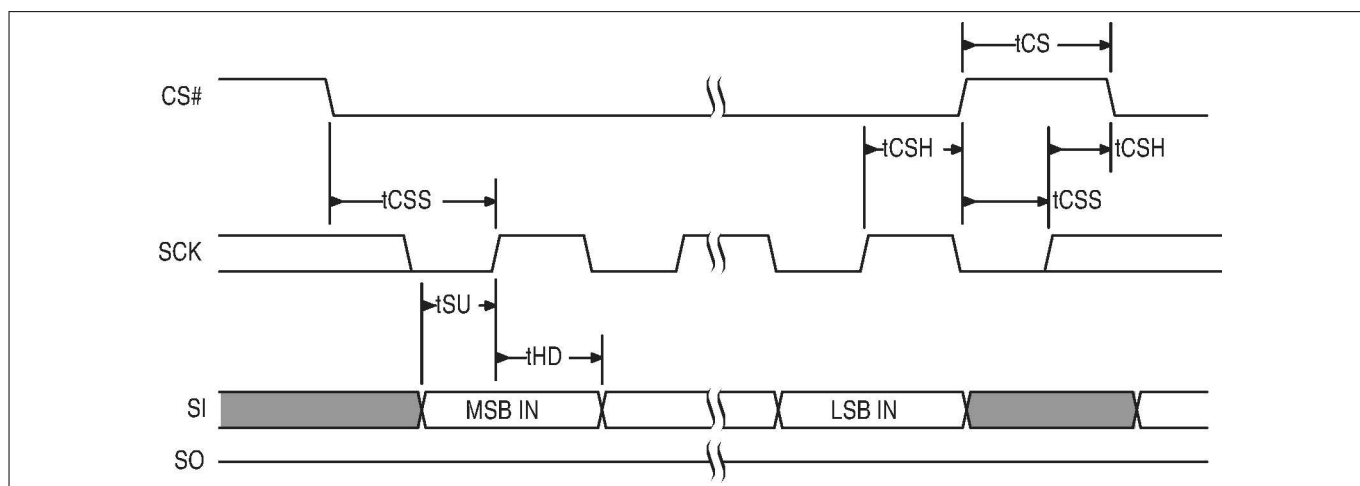
## 6 Timing specifications

### 6.4.1 Clock timing

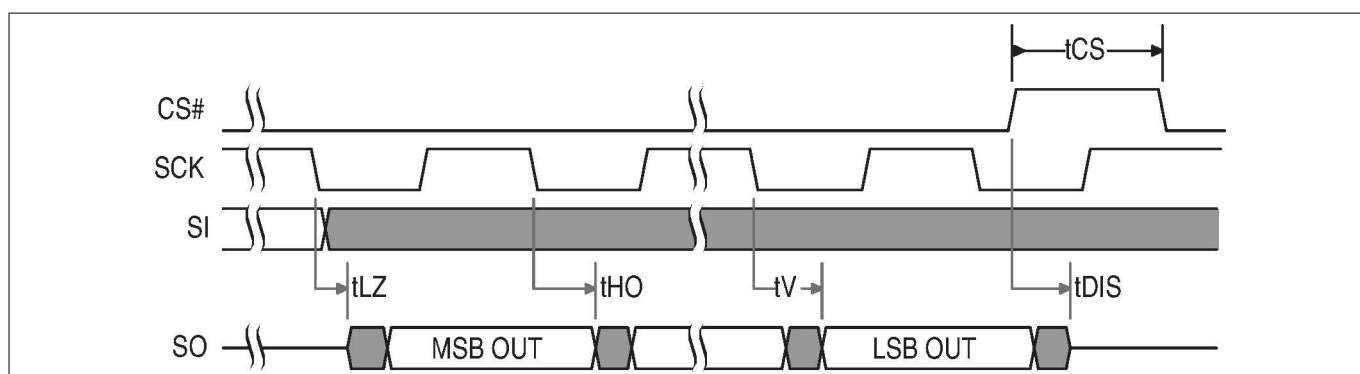


**Figure 33** Clock timing

### 6.4.2 Input/output timing



**Figure 34** SPI single bit input timing



**Figure 35** SPI single bit output timing

6 Timing specifications

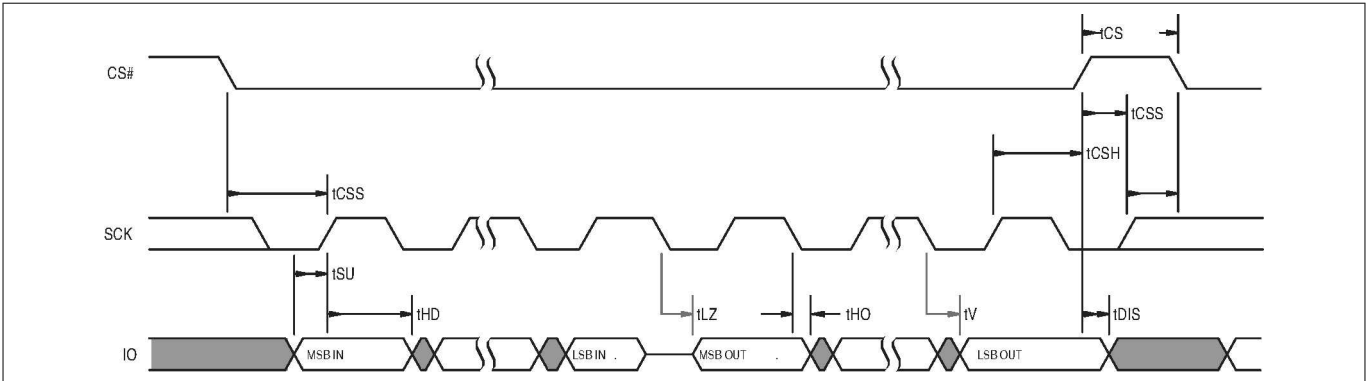


Figure 36 SPI SDR MIO timing

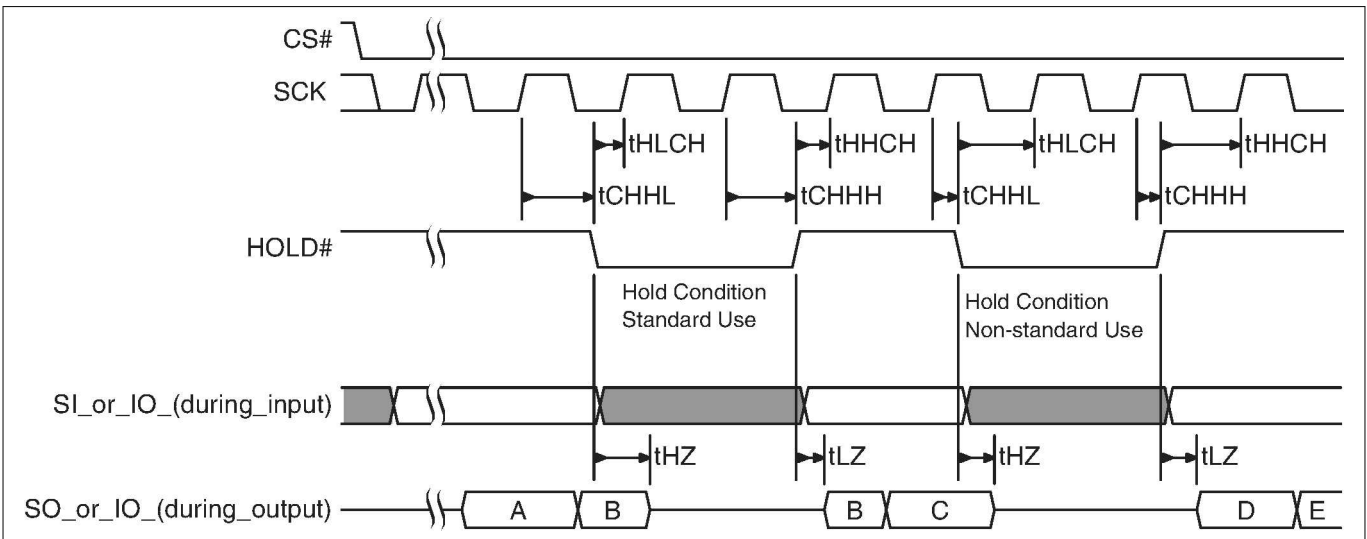


Figure 37 Hold timing

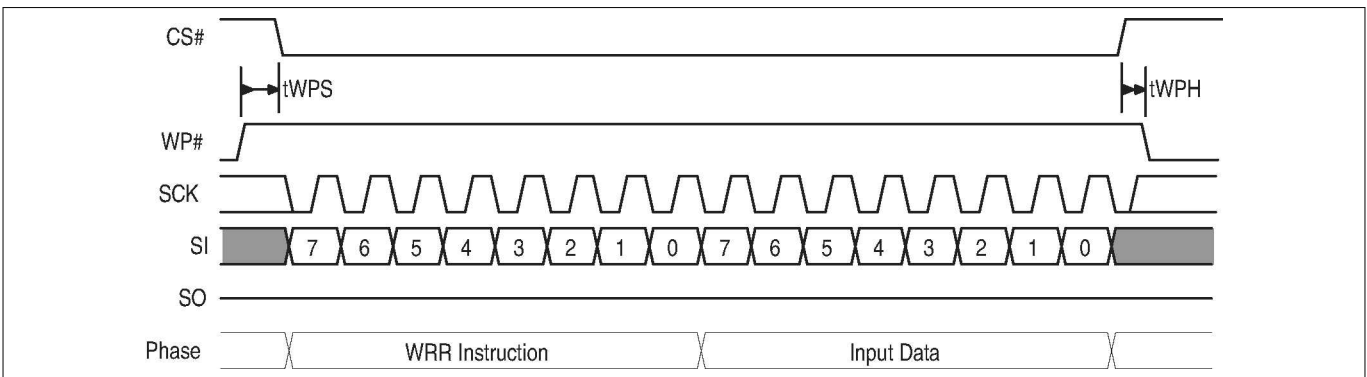


Figure 38 WP# input timing

## 6 Timing specifications

### 6.5 DDR AC characteristics

**Table 20** AC characteristics — DDR operation

Symbol	Parameter	66 MHz			80 MHz			Unit
		Min	Typ	Max	Min	Typ	Max	
$F_{SCK,R}$	SCK clock frequency for DDR READ instruction	DC	–	66	DC	–	80	MHz
$P_{SCK,R}$	SCK clock period for DDR READ instruction	15	–	$\infty$	12.5	–	$\infty$	ns
$t_{WH}, t_{CH}$	Clock high time	45% $P_{SCK}$	–	–	45% $P_{SCK}$	–	–	ns
$t_{WL}, t_{CL}$	Clock low time	45% $P_{SCK}$	–	–	45% $P_{SCK}$	–	–	ns
$t_{CS}$	CS# high time (read instructions)	10	–	–	10	–	–	ns
$t_{CSS}$	CS# active setup time (relative to SCK)	3	–	–	3	–	–	ns
$t_{CSH}$	CS# active hold time (relative to SCK)	3	–	–	3	–	–	ns
$t_{SU}$	IO in setup time	2	–	3000 <sup>1)</sup>	1.5	–	3000 <sup>1)</sup>	ns
$t_{HD}$	IO in hold time	2	–	–	1.5	–	–	ns
$t_V$	Clock low to output valid		–	6.5 <sup>2)</sup>		–	6.5 <sup>2)</sup>	ns
$t_{HO}$	Output hold time	1.5	–		1.5	–		ns
$t_{DIS}$	Output disable time		–	8		–	8	ns

(table continues...)

## 6 Timing specifications

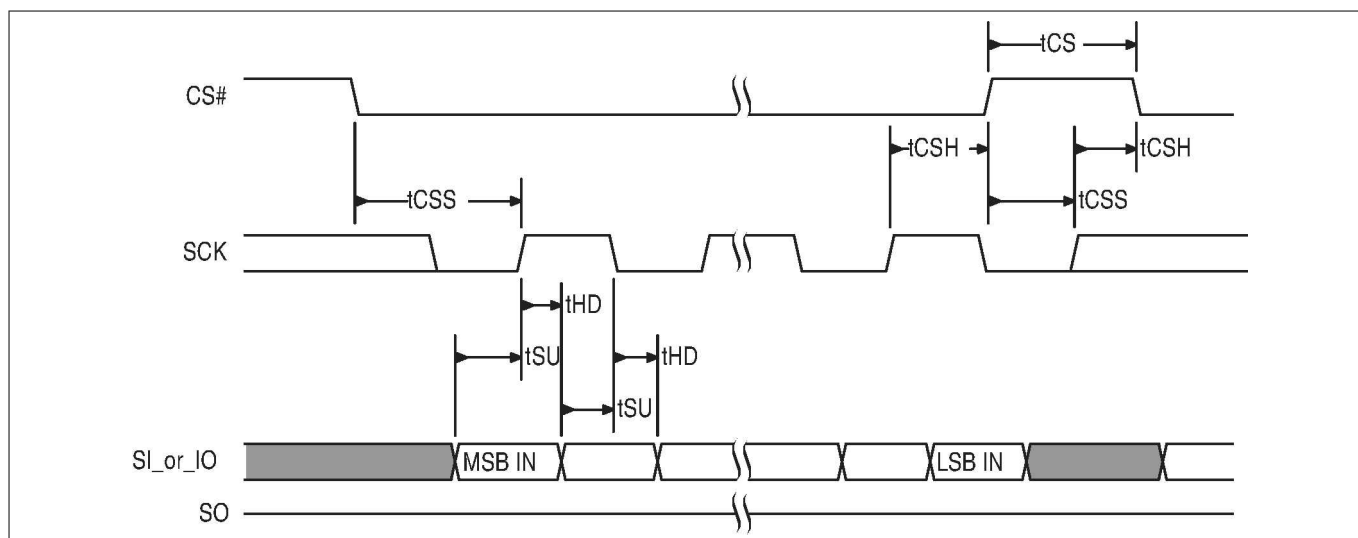
**Table 20** (continued) AC characteristics — DDR operation

Symbol	Parameter	66 MHz			80 MHz			Unit
		Min	Typ	Max	Min	Typ	Max	
tLZ	Clock to output low impedance	0	–	8	0	–	8	ns
tO_SKEW	First output to last output data valid time		–	600		–	600	ps

- 1) Maximum value only applies during Program/Erase Suspend/Resume commands.  
2) Regulated  $V_{CC}$  range (3.0 to 3.6 V) and  $CL = 15$  pF.

NOT YET CONVERTED: aanchor

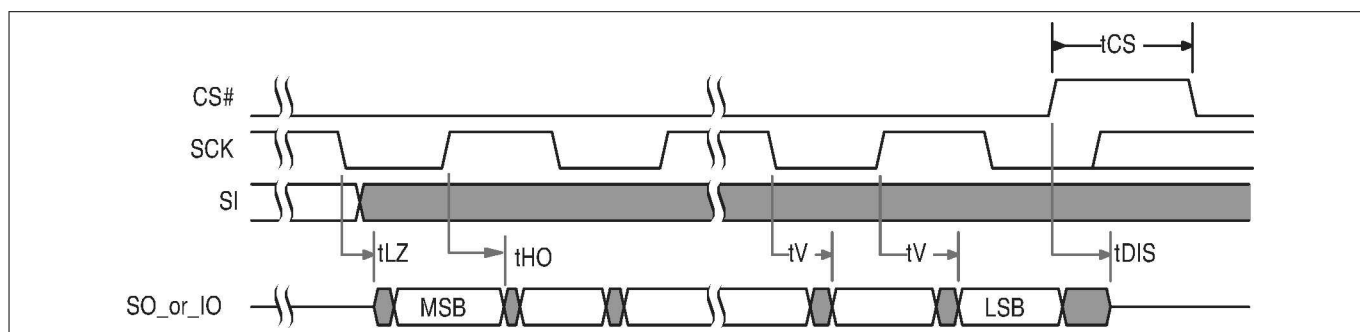
### 6.5.1 DDR input timing



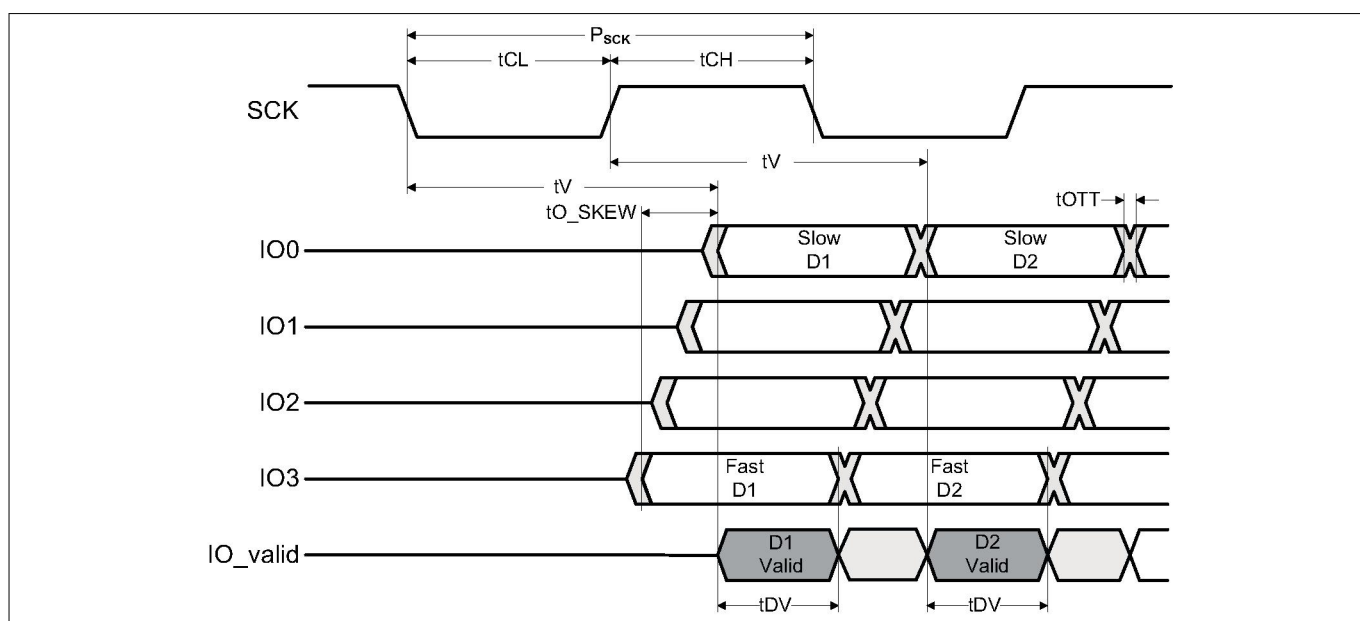
**Figure 39** SPI DDR input timing

## 6 Timing specifications

### 6.5.2 DDR output timing



**Figure 40** SPI DDR output timing



**Figure 41** SPI DDR data valid window



## 6 Timing specifications

### Notes:

1.  $t_{CLH}$  is the shorter duration of  $t_{CL}$  or  $t_{CH}$ .
2.  $t_{O\_SKEW}$  is the maximum difference (delta) between the minimum and maximum  $t_V$  (output valid) across all IO signals.
3.  $t_{OTT}$  is the maximum Output Transition Time from one valid data value to the next valid data value on each IO.
4.  $t_{OTT}$  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. As an example, assuming that the above considerations result a memory output slew rate of 2 V/ns and a 3 V transition (from 1 to 0 or 0 to 1) is required by the host, the  $t_{OTT}$  would be:  $t_{OTT} = 3 V / (2 V/ns) = 1.5 ns$
  - e.  $t_{OTT}$  is not a specification tested by Infineon, it is system dependent and must be derived by the system designer based on the above considerations.
5. The minimum data valid window ( $t_{DV}$ ) can be calculated as follows:
  - i. As an example, assuming:
  - ii. 80 MHz clock frequency = 12.5 ns clock period
  - iii. DDR operations are specified to have a duty cycle of 45% or higher
  - iv.  $t_{CLH} = 0.45 * PSCK = 0.45 \times 12.5 ns = 5.625 ns$
  - v.  $t_{O\_SKEW} = 600 ps$
  - vi.  $t_{OTT} = 1.5 ns$
  - vii.  $t_{DV} = t_{CLH} - t_{O\_SKEW} - t_{OTT}$
  - viii.  $t_{DV} = 5.625 ns - 600 ps - 1.5 ns = 3.525 ns$

## 7 Address space maps

### 7 Address space maps

This section discusses the features and behaviors most relevant to host system software that interacts with S25FL128S and S25FL256S memory devices.

#### 7.1 Overview

##### 7.1.1 Extended address

The S25FL128S and S25FL256S 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 GB (32 Gb) 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.

The S25FL128S device supports the extended address features in the same way but in essence ignores bits 31 to 24 of any address because the main flash array only needs 24-bits of address. This enables simple migration from the 128-Mb density to higher density devices without changing the address handling aspects of software.

##### 7.1.2 Multiple address spaces

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

#### 7.2 Flash memory array

The main flash array is divided into erase units called sectors. The sectors are organized either as a hybrid combination of 4-KB and 64-KB sectors, or as uniform 256-KB sectors. The sector organization depends on the device model selected, see [Ordering information](#).

**Table 21 S25FL256S sector and memory address map, bottom 4-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range(Byte address)	Notes
4	32	SA00	00000000h - 00000FFFh	Sector starting address
		:	:	—
		SA31	0001F000h - 0001FFFFh	Sector ending address

(table continues...)

## 7 Address space maps

**Table 21 (continued) S25FL256S sector and memory address map, bottom 4-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range(Byte address)	Notes
64	510	SA32	00020000h - 0002FFFFh	
		:	:	
		SA541	01FF0000h - 01FFFFFFh	

**Table 22 S25FL256S sector and memory address map, top 4-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (Byte address)	Notes
64	510	SA00	0000000h - 000FFFFh	Sector starting address
		:	:	—
		SA509	01FD0000h - 01FDFFFFh	Sector ending address
4	32	SA510	01FE0000h - 01FE0FFFh	
		:	:	
		SA541	01FFF000h - 01FFFFFFh	

**Table 23 S25FL256S sector and memory address map, uniform 256-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (8-bit)	Notes
256	128	SA00	0000000h - 003FFFFh	Sector starting address
		:	:	—
		SA127	1FC0000h - 1FFFFFFh	Sector ending address

**Table 24 S25FL128S sector and memory address map, bottom 4-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (Byte address)	Notes
4	32	SA00	00000000h - 00000FFFh	Sector starting address
		:	:	—
		SA31	0001F000h - 0001FFFFh	Sector ending address
64	254	SA32	00020000h - 0002FFFFh	

(table continues...)

## 7 Address space maps

**Table 24 (continued) S25FL128S sector and memory address map, bottom 4-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (Byte address)	Notes
		:	:	
		SA285	00FF0000h - 00FFFFFFh	

**Table 25 S25FL128S sector and memory address map, top 4-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (Byte address)	Notes
64	254	SA00	0000000h - 003FFFFh	Sector starting address
		:	:	—
		SA63	0FC0000h - 0FFFFFFh	Sector ending address
4	32	SA254	00FE0000h - 00FE0FFFh	
		:	:	
		SA285	00FFF000h - 00FFFFFFh	

**Table 26 S25FL128S sector and memory address map, uniform 256-KB sectors**

Sector size (KB)	Sector count	Sector range	Address range (Byte address)	Notes
256	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. All 256 KB sectors have the pattern XXX0000h-XXXXFFFh.

### 7.3 ID-CFI address space

The RDID command (9 Fh) 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](#) 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.

### 7.4 OTP address space

Each S25FL128S and S25FL256S memory device has a 1024-byte One Time Program (OTP) address space that is separate from the main flash array. The OTP area is divided into 32, individually lockable, 32-byte aligned and length regions.

## 7 Address space maps

In the 32-byte region starting at address zero:

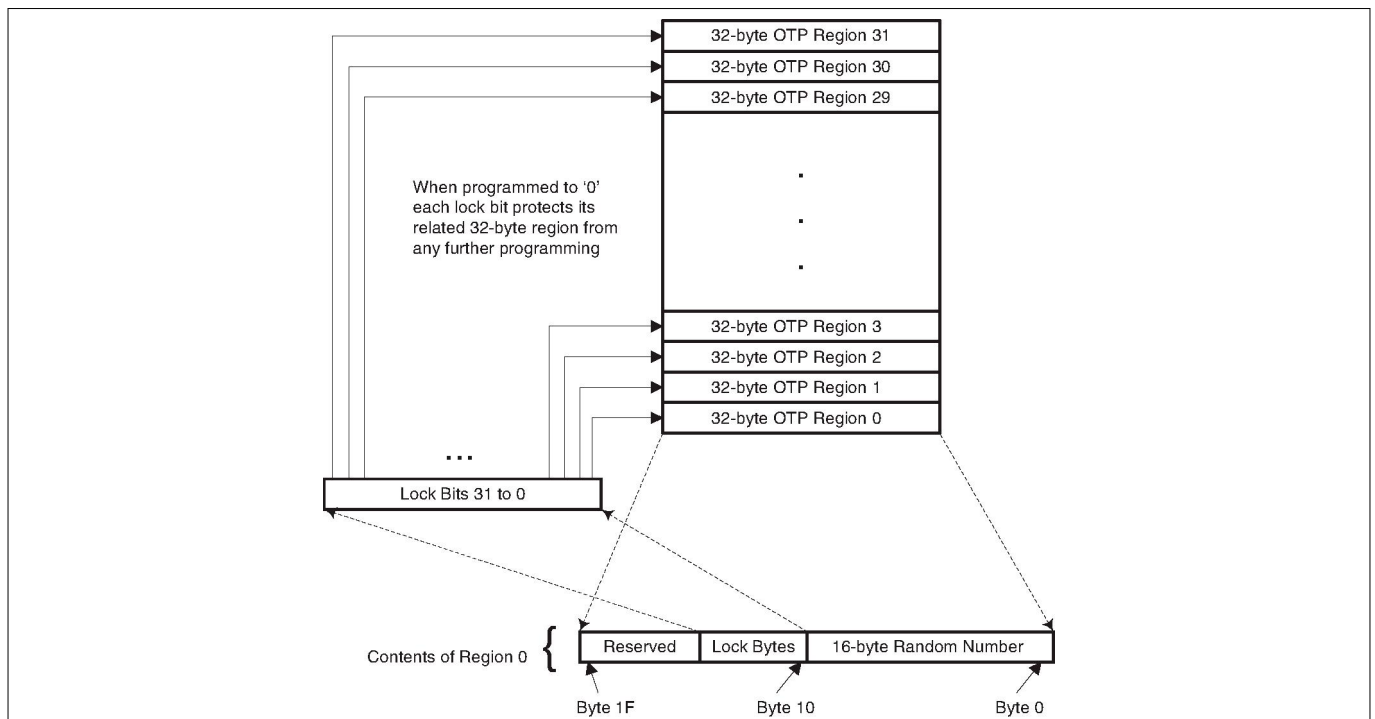
- The 16 lowest address bytes are programmed by Infineon with a 128-bit random number. Only Infineon is able to program these bytes.
- The next 4 higher address bytes (OTP Lock Bytes) are used to provide one bit per OTP region to permanently protect each
- region from programming. The bytes are erased 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.

Refer to [Figure 42](#) 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.



**Figure 42** OTP address space

## 7 Address space maps

**Table 27**                      **OTP address map**

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
	014 to 01F	Reserved for Future Use (RFU)	All bytes = FF
Region 1	020 to 03F	Available for user programming	All bytes = FF
Region 2	040 to 05F	Available for user programming	All bytes = FF
...	...	Available for user programming	All bytes = FF
Region 31	3E0 to 3FF	Available for user programming	All bytes = FF

## 8 Registers

### 8 Registers

Registers are small groups of memory cells used to configure how the S25FL-S memory device operates or to report the status of device operations. The registers are accessed by specific commands. The commands (and hexadecimal instruction codes) used for each register are noted in each register description. The individual register bits may be volatile, non-volatile, or One Time Programmable (OTP). The type for each bit is noted in each register description. The default state shown for each bit refers to the state after power-on reset, hardware reset, or software reset if the bit is volatile. If the bit is non-volatile or OTP, the default state is the value of the bit when the device is shipped from Infineon. Non-volatile bits have the same cycling (erase and program) endurance as the main flash array.

**Table 28 Register description**

Register	Abbreviation	Type	Bit location
Status Register 1	SR1[7:0]	Volatile	7:0
Configuration Register 1	CR1[7:0]	Volatile	7:0
Status Register 2	SR2[7:0]	RFU	7:0
AutoBoot Register	ABRD[31:0]	Non-volatile	31:0
Bank Address Register	BRAC[7:0]	Volatile	7:0
ECC Status Register	ECCSR[7:0]	Volatile	7:0
ASP Register	ASPR[15:1]	OTP	15:1
ASP Register	ASPR[0]	RFU	0
Password Register	PASS[63:0]	Non-volatile OTP	63:0
PPB Lock Register	PPBL[7:1]	Volatile	7:1
PPB Lock Register	PPBL[0]	Volatile Read only	0
PPB Access Register	PPBAR[7:0]	Non-volatile	7:0
DYB Access Register	DYBAR[7:0]	Volatile	7:0
SPI DDR Data Learning Registers	NVDLR[7:0]	Non-volatile	7:0
SPI DDR Data Learning Registers	VDLR[7:0]	Volatile	7:0

#### 8.1 Status Register 1 (SR1)

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

**8 Registers**

**Table 29 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 Infineon	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

**(table continues...)**



## 8 Registers

**Table 29 (continued) Status Register 1 (SR1)**

Bits	Field name	Function	Type	Default state	Description
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 a '1' it indicates that there was an error in the last program operation. This bit will also be set when the user attempts to program within a protected main memory sector or locked OTP region. When the Program Error bit is set to a '1' this bit can be reset to '0' with the Clear Status Register (CLSR) command. This is a read-only bit and is not affected by the WRR command.

**Erase Error (E\_ERR) SR1[5]:** The Erase Error Bit is used as an Erase operation success or failure indication. When the Erase Error bit is set to a 1 it indicates that there was an error in the last erase operation. This bit will also be set when the user attempts to erase an individual protected main memory sector. The Bulk Erase command will not set E\_ERR if a protected sector is found during the command execution. When the Erase Error bit is set to a '1' this bit can be reset to '0' with the Clear Status Register (CLSR) command. This is a read-only bit and is not affected by the WRR command.

**Block Protection (BP2, BP1, BP0) SR1[4:2]:** These bits define the main flash array area to be software-protected against program and erase commands. The BP bits are either volatile or non-volatile, depending on the state of the BP non-volatile bit (BPNV) in the configuration register. When one or more of the BP bits is set to '1', the relevant memory area is protected against program and erase. The Bulk Erase (BE) command can be executed only when the BP bits are cleared to 0's. See [Block protection](#) for a description of how the BP bit values select the memory array area protected. The BP bits have the same non-volatile endurance as the main flash array.

**Write Enable Latch (WEL) SR1[1]:** The WEL bit must be set to '1' to enable program, write, or erase operations as a means to provide protection against inadvertent changes to memory or register values. The Write Enable (WREN) command execution sets the Write Enable Latch to a '1' to allow any program, erase, or write commands to execute afterwards. The Write Disable (WRDI) command can be used to set the Write Enable Latch to a '0' to prevent all program, erase, and write commands from execution. The WEL bit is cleared to '0' at the end of any successful program, write, or erase operation. Following a failed operation the WEL bit may remain set and should be cleared with a WRDI command following a CLSR command. After a power down/power up sequence, hardware reset, or software reset, the Write Enable Latch is set to a '0'. The WRR command does not affect this bit.

## 8 Registers

**Write In Progress (WIP) SR1[0]:** Indicates whether the device is performing a program, write, erase operation, or any other operation. When the bit is set to a '1' the device is busy performing an operation. While WIP is '1', one must only submit the following commands: Read Status (RDSR1 or RDSR2), Erase Suspend (ERSP), Program Suspend (PGSP), Clear Status Register (CLSR), and Software Reset (RESET). ERSP and PGSP will only be accepted if memory array erase or program operations are in progress. The status register E\_ERR and P\_ERR bits are updated while WIP = 1. When P\_ERR or E\_ERR bits are set to one, the WIP bit will remain set to one indicating the device remains busy and unable to receive new operation commands. A Clear Status Register (CLSR) command must be received to return the device to standby mode. When the WIP bit is cleared to '0' no operation is in progress. This is a read-only bit.

### 8.2 Configuration Register 1 (CR1)

Related Commands: Read Configuration Register (RDCR 35 h), Write Registers (WRR 01 h). 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 30 Configuration Register 1(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	RFU	RFU	OTP	0	Reserved for Future Use
3	BPNV	Configures BP2-0 in Status Register	OTP	0	1 = Volatile 0 = Non-volatile
2	TBPARM	Configures parameter sectors location	OTP	0	1 = 4-KB physical sectors at top, (high address) 0 = 4-KB physical sectors at bottom (Low address) RFU in uniform sector devices
1	QUAD	Puts the device into Quad I/O operation	Non-volatile	0	1 = Quad 0 = Dual or Serial

(table continues...)

## 8 Registers

**Table 30** (continued) **Configuration Register 1(CR1)**

Bits	Field name	Function	Type	Default state	Description
0	FREEZE	Lock current state of BP2-0 bits in Status Register, TBPROT and TBPARM in Configuration Register, and OTP regions	Volatile	0	1 = Block Protection and OTP locked 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. The High Performance versus the Enhanced High Performance settings are selected by the ordering part number.

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.

The High Performance settings provide latency options that are the same or faster than alternate source SPI memories. These settings provide mode bits only for the Quad I/O Read command.

The Enhanced High Performance settings similarly provide latency options the same or faster than additional alternate source SPI memories and adds mode bits for the Dual I/O Read, DDR Fast Read, and DDR Dual I/O Read commands.

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](#) for more information on the DLP.

**Table 31** **Latency codes for SDR high performance**

Freq. (MHz)	LC	Read		Fast Read		Read Dual Out		Read Quad Out		Dual I/O Read		Quad I/O Read	
		(03 h, 13 h)		(0 Bh, 0 Ch)		(3 Bh, 3 Ch)		(6 Bh, 6 Ch)		(BBh, BCh)		(EBh, ECh)	
		Mode	Dum my	Mode	Dum my	Mode	Dum my	Mode	Dum my	Mode	Dum my	Mode	Dum my
$\leq 50$	11	0	0	0	0	0	0	0	0	0	4	2	1

(table continues...)

**8 Registers**

**Table 31 (continued) Latency codes for SDR high performance**

Freq. (MHz)	LC	Read		Fast Read		Read Dual Out		Read Quad Out		Dual I/O Read		Quad I/O Read	
		(03 h, 13 h)		(0 Bh, 0 Ch)		(3 Bh, 3 Ch)		(6 Bh, 6 Ch)		(BBh, BCh)		(EBh, ECh)	
		Mode	Dummy	Mode	Dummy	Mode	Dummy	Mode	Dummy	Mode	Dummy	Mode	Dummy
≤ 80	00	-	-	0	8	0	8	0	8	0	4	2	4
≤ 90	01	-	-	0	8	0	8	0	8	0	5	2	4
≤104	10	-	-	0	8	0	8	0	8	0	6	2	5
≤133	10	-	-	0	8	-	-	-	-	-	-	-	-

**Table 32 Latency codes for DDR high performance**

Freq. (MHz)	LC	DDR Fast Read		DDR Dual I/O Read		Read DDR Quad I/O	
		(0 Dh, 0 Eh)		(BDh, BEh)		(EDh, EEh)	
		Mode	Dummy	Mode	Dummy	Mode	Dummy
≤ 50	11	0	4	0	4	1	3
≤ 66	00	0	5	0	6	1	6
≤ 66	01	0	6	0	7	1	7
≤ 66	10	0	7	0	8	1	8

**Table 33 Latency codes for SDR enhanced high performance**

Freq. (MHz)	LC	Read		Fast Read		Read Dual Out		Read Quad Out		Dual I/O Read		Quad I/O Read	
		(03 h, 13 h)		(0 Bh, 0 Ch)		(3 Bh, 3 Ch)		(6 Bh, 6 Ch)		(BBh, BCh)		(EBh, ECh)	
		Mode	Dummy	Mode	Dummy	Mode	Dummy	Mode	Dummy	Mode	Dummy	Mode	Dummy
≤ 50	11	0	0	0	0	0	0	0	0	4	0	2	1
≤ 80	00	-	-	0	8	0	8	0	8	4	0	2	4
≤ 90	01	-	-	0	8	0	8	0	8	4	1	2	4
≤104	10	-	-	0	8	0	8	0	8	4	2	2	5
≤133	10	-	-	0	8	-	-	-	-	-	-	-	-

**Table 34 Latency codes for DDR enhanced high performance<sup>1)</sup>**

Freq. (MHz)	LC	DDR Fast Read		DDR Dual I/O Read		Read DDR Quad I/O	
		(0 Dh, 0 Eh)		(BDh, BEh)		(EDh, EEh)	
		Mode	Dummy	Mode	Dummy	Mode	Dummy
≤ 50	11	4	1	2	2	1	3
≤ 66	00	4	2	2	4	1	6

(table continues...)

## 8 Registers

**Table 34** (continued) Latency codes for DDR enhanced high performance<sup>1)</sup>

Freq. (MHz)	LC	DDR Fast Read		DDR Dual I/O Read		Read DDR Quad I/O	
		(0 Dh, 0 Eh)		(BDh, BEh)		(EDh, EEh)	
		Mode	Dummy	Mode	Dummy	Mode	Dummy
≤ 66	01	4	4	2	5	1	7
≤ 66	10	4	5	2	6	1	8
≤ 80	00	4	2	2	4	1	6
≤ 80	01	4	4	2	5	1	7
≤ 80	10	4	5	2	6	1	8

1) When using DDR I/O commands with the Data Learning Pattern (DLP) enabled, a Latency Code that provides 5 or more dummy cycles should be selected to allow 1 cycle of additional time for the host to stop driving before the memory starts driving the 4 cycle DLP. It is recommended to use LC 10 for DDR Fast Read, LC 01 for DDR Dual IO Read, and LC 00 for DDR Quad IO Read, if the Data Learning Pattern (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 a 0 the Block Protection is defined to start from the top (maximum address) of the array. When TBPROT is set to a 1 the Block Protection is defined to start from the bottom (zero address) of the array. The TBPROT bit is OTP and set to a 0 when shipped from Infineon. If TBPROT is programmed to 1, an attempt to change it back to 0 will fail and set the Program Error bit (P\_ERR in SR1[6]).

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

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

**Block Protection Non-volatile (BPNV) CR1[3]:** The BPNV bit defines whether or not the BP2-0 bits in the Status Register are volatile or non-volatile. The BPNV bit is OTP and cleared to a 0 with the BP bits cleared to 000 when shipped from Infineon. When BPNV is set to a 0 the BP2-0 bits in the Status Register are non-volatile. When BPNV is set to a 1 the BP2-0 bits in the Status Register are volatile and will be reset to binary 111 after POR, hardware reset, or command reset. 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 4-KB small sectors (SMS), which replace two 64-KB sectors. When TBPARM is set to a 1 the parameter block is in the top of the memory array address space. When TBPARM is set to a 0 the parameter block is at the Bottom of the array. TBPARM is OTP and set to a 0 when it ships from 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]).

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

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

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

**Quad Data Width (QUAD) CR1[1]:** When set to 1, this bit switches the data width of the device to 4 bit - Quad mode. That is, WP# becomes IO2 and HOLD# becomes IO3. The WP# and HOLD# inputs are not monitored for

## 8 Registers

their normal functions and are internally set to high (inactive). The commands for Serial, Dual Output, and Dual I/O Read still function normally but, there is no need to drive WP# and Hold# inputs for those commands when switching between commands using different data path widths. The QUAD bit must be set to one when using Read Quad Out, Quad I/O Read, Read DDR Quad I/O, and Quad Page Program commands. The QUAD bit is non-volatile.

**Freeze Protection (FREEZE) CR1[0]:** The Freeze Bit, when set to 1, locks the current state of the BP2-0 bits in Status Register, the TBPROT and TBPARM bits in the Configuration Register, and the OTP address space. This prevents writing, programming, or erasing these areas. As long as the FREEZE bit remains cleared to logic 0 the other bits of the Configuration Register, including FREEZE, are writable, and the OTP address space is programmable. Once the FREEZE bit has been written to a logic 1 it can only be cleared to a logic 0 by a power-off to power-on cycle or a hardware reset. Software reset will not affect the state of the FREEZE bit. The FREEZE bit is volatile and the default state of FREEZE after power-on is 0. The FREEZE bit can be set in parallel with updating other values in CR1 by a single WRR command.

### Notes:

1. When using DDR I/O commands with the Data Learning Pattern (DLP) enabled, a Latency Code that provides 5 or more dummy cycles should be selected to allow 1 cycle of additional time for the host to stop driving before the memory starts driving the 4 cycle DLP. It is recommended to use LC 10 for DDR Fast Read, LC 01 for DDR Dual IO Read, and LC 00 for DDR Quad IO Read, if the Data Learning Pattern (DLP) for DDR is used.

## 8.3 Status Register 2 (SR2)

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

**Table 35 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

(table continues...)

## 8 Registers

**Table 35** (continued) **Status Register 2 (SR2)**

Bits	Field name	Function	Type	Default state	Description
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 (75 h) (7 Ah) 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 \(PGSP 85 h\)](#) and [Resume \(PGRS 8 Ah\)](#) for details.

## 8.4 AutoBoot Register

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

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 36** **AutoBoot Register**

Bits	Field name	Function	Type	Default state	Description
31 to 9	ABSA	AutoBoot Start Address	Non-volatile	000000 h	512-byte boundary address for the start of boot code access
8 to 1	ABSD	AutoBoot Start Delay	Non-volatile	00 h	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 Bank Address Register

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

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



## 8 Registers

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 37 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 1	RFU	Reserved	Volatile	00000b	Reserved for Future Use
0	BA24	Bank Address	Volatile	0	A24 for 256-Mb device, 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.

### 8.6 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](#).

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 38 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.

(table continues...)



## 8 Registers

**Table 38** (continued) **ECC Status Register (ECCSR)**

Bits	Field name	Function	Type	Default state	Description
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.

### 8.7 ASP Register (ASPR)

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

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

**Table 39** **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	<sup>1)</sup>	Reserved for Future Use
7	RFU	Reserved	OTP	<sup>1)</sup>	Reserved for Future Use
6	RFU	Reserved	OTP	1	Reserved for Future Use
5	RFU	Reserved	OTP	<sup>1)</sup>	Reserved for Future Use
4	RFU	Reserved	OTP	<sup>1)</sup>	Reserved for Future Use
3	RFU	Reserved	OTP	<sup>1)</sup>	Reserved for Future Use

(table continues...)

## 8 Registers

**Table 39** (continued) **ASP Register (ASPR)**

Bits	Field name	Function	Type	Default State	Description
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

1) Default value depends on ordering part number, see [Initial delivery state](#)

### 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 zero.

## 8.8 Password Register (PASS)

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

**Table 40** **Password Register (PASS)**

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

## 8 Registers

### 8.9 PPB Lock Register (PPBL)

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

**Table 41 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.

### 8.10 PPB Access Register (PPBAR)

Related Commands: PPB Read (PPBRD E2h)

**Table 42 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.

### 8.11 DYB Access Register (DYBAR)

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

**8 Registers**

**Table 43**                      **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.

## 8.12 SPI DDR Data Learning Registers

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

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 00 h. 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 34 h (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 00 h, no preamble data pattern is presented during the dummy phase in the DDR commands.

**8 Registers**

**Table 44 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 45 Volatile Data Learning Register (VDLDR)**

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

## 9 Embedded algorithm performance tables

### 9 Embedded algorithm performance tables

**Table 46 Program and Erase performance**

Symbol	Parameter	Min	Typ <sup>1)</sup>	Max <sup>2)</sup>	Unit
$t_W$	WRR Write Time	–	140	500	ms
$t_{PP}$	Page Programming (512 bytes) Page Programming (256 bytes)	–	540 400	750 750 <sup>3)</sup>	$\mu$ s
$t_{SE}$	Sector Erase Time (64-KB/4-KB physical sectors)	–	171	650 <sup>4)</sup>	ms
	Sector Erase Time (64 KB Top/Bottom: logical sector = 16 x 4-KB physical sectors)	–	3,610	10,400	ms
	Sector Erase Time (256-KB logical sectors = 4 x 64-KB physical sectors)	–	685	2600	ms
$t_{BE}$	Bulk Erase Time (S25FL128S)	–	33	165	sec
$t_{BE}$	Bulk Erase Time (S25FL256S)	–	66	330	sec

1) Typical program and erase times assume the following conditions: 25°C,  $V_{CC} = 3.0$  V; 10,000 cycles; checkerboard data pattern.

2) Under worst case conditions of 90°C; 100,000 cycles max.

3) Maximum value also applies to OTP, PPBP, ASPP, PASSP, ABWR, and PNVDLR programming commands.

4) Maximum value also applies to the PPBE erase command.

**Table 47 Program Suspend AC parameters**

Parameter	Min	Typ	Max	Unit	Comments
Program Suspend Latency ( $t_{PSL}$ )	–	–	40	$\mu$ s	The time from Program Suspend command until the WIP bit is '0'

(table continues...)

**9 Embedded algorithm performance tables**

**Table 47** (continued) Program Suspend AC parameters

Parameter	Min	Typ	Max	Unit	Comments
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 48** Erase Suspend AC parameters

Parameter	Min	Typ	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

## **10 Data protection**

# **10 Data protection**

## **10.1 Secure silicon region (OTP)**

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

The OTP memory space is intended for increased system security. OTP values can “mate” a flash component with the system CPU/ASIC to prevent device substitution. See [OTP address space](#), [One Time Program Array commands](#), and [OTP Read \(OTPR 4 Bh\)](#).

### **10.1.1 Reading OTP memory space**

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

### **10.1.2 Programming OTP memory space**

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

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

The valid address range for OTP Program is depicted in [Figure 42](#). 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’.

### **10.1.3 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.

### **10.1.4 Lock bytes**

The least significant bits (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](#).

## **10.2 Write Enable command**

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

- Reset
- Page Program (PP)
- Sector Erase (SE)
- Bulk Erase (BE)
- Write Disable (WRDI)



## 10 Data protection

- Write Registers (WRR)
- Quad-input Page Programming (QPP)
- OTP Byte Programming (OTPP)

### 10.3 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 49 Upper array start of protection (TBPROT = 0)**

Status Register content			Protected fraction of memory array	Protected memory (KB)	
BP2	BP1	BP0		FL128S 128 Mb	FL256S 256 Mb
0	0	0	None	0	0
0	0	1	Upper 64th	256	512
0	1	0	Upper 32nd	512	1024
0	1	1	Upper 16th	1024	2048
1	0	0	Upper 8th	2048	4096
1	0	1	Upper 4th	4096	8192
1	1	0	Upper half	8192	16384
1	1	1	All Sectors	16384	32768

**Table 50 Lower array start of protection (TBPROT = 1)**

Status Register content			Protected fraction of memory array	Protected memory (KB)	
BP2	BP1	BP0		FL128S 128 Mb	FL256S 256 Mb
0	0	0	None	0	0
0	0	1	Lower 64th	256	512
0	1	0	Lower 32nd	512	1024
0	1	1	Lower 16th	1024	2048
1	0	0	Lower 8th	2048	4096
1	0	1	Lower 4th	4096	8192
1	1	0	Lower half	8192	16384
1	1	1	All Sectors	16384	32768

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.

## 10 Data protection

### 10.3.1 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.

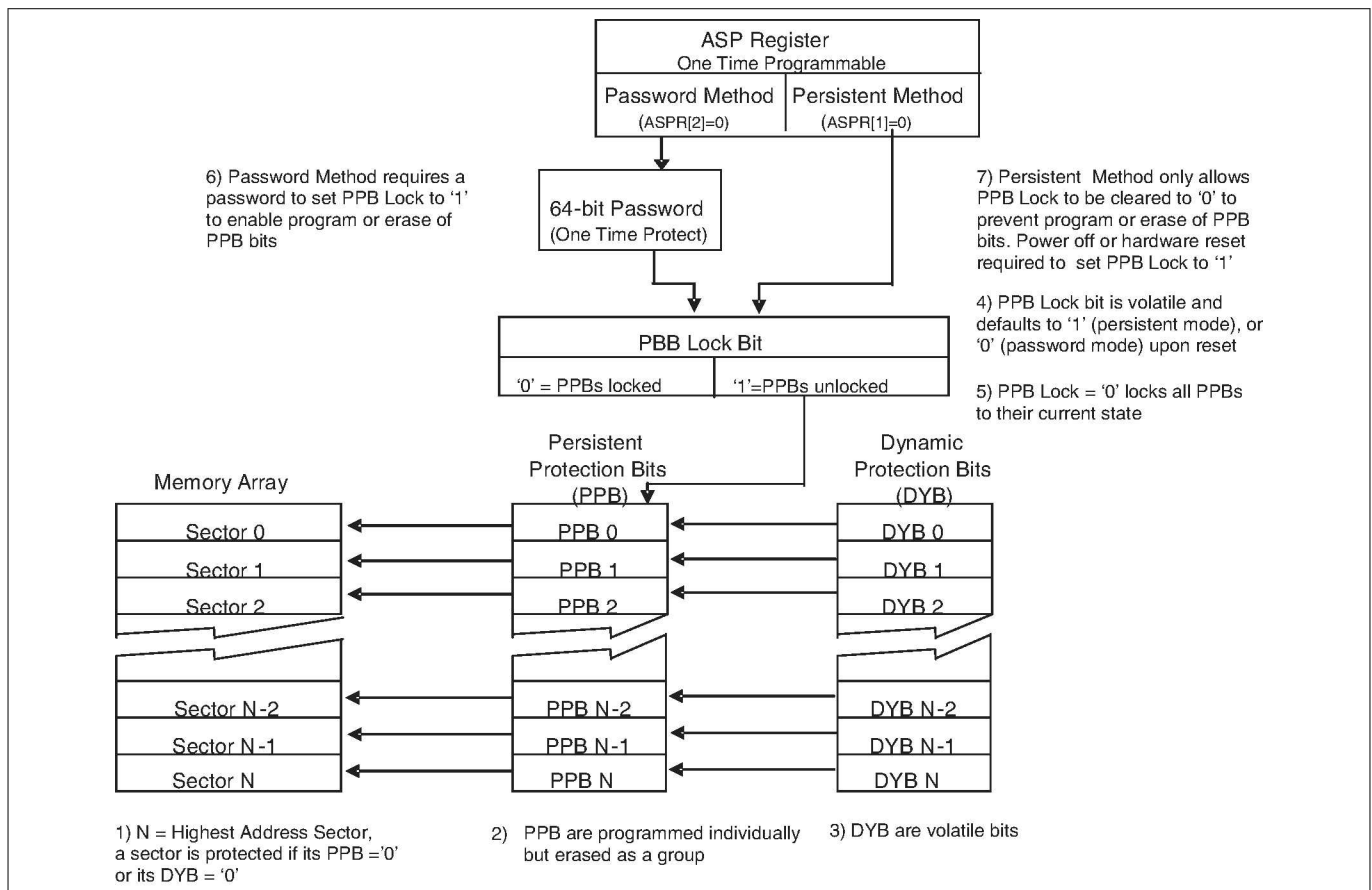
### 10.3.2 Write Protect signal

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

## 10.4 Advanced sector protection

Advanced Sector Protection (ASP) is the name used for a set of independent hardware and software methods used to disable or enable programming or erase operations, individually, in any or all sectors. An overview of these methods is shown in [Figure 43](#).

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](#) for full details of the BP2-0 bits.



**Figure 43** Advanced sector protection overview

## 10 Data protection

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.

### 10.4.1 ASP Register

The ASP register is used to permanently configure the behavior of Advanced Sector Protection (ASP) features. See [Table 39](#).

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 \(SR1\)](#) 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.

### 10.4.2 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

## **10 Data protection**

programmed in the main array but an entire sector must be erased at the same time. The PPB have the same program and erase endurance as the main flash memory array. Preprogramming and verification prior to erasure are handled by the device.

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

- Note:**
- *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.*

### **10.4.3 Dynamic protection bits**

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

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

The PPB Lock Bit is a volatile bit for protecting all PPB bits. When cleared to '0', it locks all PPBs 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.

### **10.4.5 Sector protection states summary**

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

- **Unlocked** — The sector is unprotected and protection can be changed by a simple command. The protection state defaults to unprotected 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

## 10 Data protection

**Table 51**                      **Sector protection states**

Protection bit values			Sector state
PPB Lock	PPB	DYB	
1	1	1	Unprotected – PPB and DYB are changeable
1	1	0	Protected – PPB and DYB are changeable
1	0	1	Protected – PPB and DYB are changeable
1	0	0	Protected – PPB and DYB are changeable
0	1	1	Unprotected – PPB not changeable, DYB is changeable
0	1	0	Protected – PPB not changeable, DYB is changeable
0	0	1	Protected – PPB not changeable, DYB is changeable
0	0	0	Protected – PPB not changeable, DYB is changeable

### 10.4.6 Persistent Protection mode

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

### 10.4.7 Password Protection mode

Password Protection Mode allows an even higher level of security than the Persistent Sector Protection Mode, by requiring a 64-bit password for unlocking the PPB Lock bit. In addition to this password requirement, after power up and hardware reset, the PPB Lock bit is cleared to '0' to ensure protection at power-up. Successful execution of the Password Unlock command by entering the entire password 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 a '1' after a cell is programmed as a '0' results in the cell left as a '0' with no programming error set.
- The password is all 1's when shipped from 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.

## **10 Data protection**

- The Password Mode, once programmed, prevents reading the 64-bit password and further password programming. All further program and read commands to the password region are disabled and these commands are ignored. There is no means to verify what the password is after the Password Mode Lock Bit is selected. Password verification is only allowed before selecting the Password Protection mode.
- The Protection Mode Lock Bits are not erasable.
- The exact password must be entered in order for the unlocking function to occur. If the password unlock command provided password does not match the hidden internal password, the unlock operation fails in the same manner as a programming operation on a protected sector. The P\_ERR bit is set to one 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 zero.
- If the password is lost after selecting the Password Mode, there is no way to set the PPB Lock bit.
- ECC status may only be read from sectors that are readable. In read protection mode the addresses are forced to the boot sector address. ECC status is shown in that sector while read protection mode is active.

## 11 Commands

### 11 Commands

All communication between the host system and S25FL128S and S25FL256S memory devices is in the form of units called commands.

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

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

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

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

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

Commands are structured as follows:

- Each command begins with an eight bit (byte) instruction.
- The instruction may be stand alone or may be followed by address bits to select a location within one of several address spaces in the device. The address may be either a 24-bit or 32-bit byte boundary address.
- The Serial Peripheral Interface with Multiple IO provides the option for each transfer of address and data information to be done one, two, or four bits in parallel. This enables a trade off between the number of signal connections (IO bus width) and the speed of information transfer. If the host system can support a two or four bit wide IO bus the memory performance can be increased by using the instructions that provide parallel two bit (dual) or parallel four bit (quad) transfers.
- 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



## 11 Commands

recommended to check that the Write-In Progress (WIP) bit is 0 before issuing most commands to the device, to ensure the new command can be accepted.

- Depending on the command, the time for execution varies. A command to read status information from an executing command is available to determine when the command completes execution and whether the command was successful.
- Although host software in some cases is used to directly control the SPI interface signals, the hardware interfaces of the host system and the memory device generally handle the details of signal relationships and timing. For this reason, signal relationships and timing are not covered in detail within this software interface focused section of the document. Instead, the focus is on the logical sequence of bits transferred in each command rather than the signal timing and relationships. Following are some general signal relationship descriptions to keep in mind. For additional information on the bit level format and signal timing relationships of commands, see [Command protocol](#).
  - The host always controls the Chip Select (CS#), Serial Clock (SCK), and Serial Input (SI) - SI for single bit wide transfers. The memory drives Serial Output (SO) for single bit read transfers. The host and memory alternately drive the IO0-IO3 signals during Dual and Quad transfers.
  - All commands begin with the host selecting the memory by driving CS# low before the first rising edge of SCK. CS# is kept low throughout a command and when CS# is returned high the command ends. Generally, CS# remains low for 8-bit transfer multiples to transfer byte granularity information. Some commands will not be accepted if CS# is returned high not at an 8-bit boundary.

### 11.1 Command set summary

#### 11.1.1 Extended addressing

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

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

Instruction name	Description	Code (Hex)
4FAST_READ	Read Fast (4-byte address)	0C
4READ	Read (4-byte address)	13
4DOR	Read Dual Out (4-byte address)	3C
4QOR	Read Quad Out (4-byte address)	6C
4DIOR	Dual I/O Read (4-byte address)	BC
4QIOR	Quad I/O Read (4-byte address)	EC
4DDRFR	Read DDR Fast (4-byte address)	0E
4DDRDIOR	DDR Dual I/O Read (4-byte address)	BE
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 4-KB Erase (4-byte address)	21
4SE	Erase 64/256 KB (4-byte address)	DC



## 11 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.

Instruction name	Description	Code (Hex)
READ	Read (3-byte address)	03
FAST_READ	Read Fast (3-byte address)	0B
DOR	Read Dual Out (3-byte address)	3B
QOR	Read Quad Out (3-byte address)	6B
DIOR	Dual I/O Read (3-byte address)	BB
QIOR	Quad I/O Read (3-byte address)	EB
DDRFR	Read DDR Fast (3-byte address)	0D
DDRDIOR	DDR Dual I/O Read (3-byte address)	BD
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 4-KB Erase (3-byte address)	20
SE	Erase 64/256 KB (3-byte address)	D8

3. For backward compatibility to the 3-byte addressing, the standard instructions can be used in conjunction with the Bank Address Register:
- 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.
    - The host system writes the Bank Address Register to access beyond the first 128 Mb of memory.
    - This applies to read, erase, and program commands.
  - 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.
  - Bank Register bits are volatile.
    - On power up, the default is Bank0 (the lowest address 16 MB).
  - For Read, the device will continuously transfer out data until the end of the array.
    - There is no bank to bank delay.
    - The Bank Address Register is not updated.
    - The Bank Address Register value is used only for the initial address of an access.

## 11 Commands

**Table 52 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

**Table 53 S25FL128S and S25FL256S 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

(table continues...)

## 11 Commands

**Table 53 (continued) S25FL128S and S25FL256S command set (sorted by function)**

Function	Command name	Command description	Instruction value (Hex)	Maximum frequency (MHz)
	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
	DDRFR	DDR Fast Read (3- or 4-byte address)	0D	80
	4DDRFR	DDR Fast Read (4-byte address)	0E	80
	DOR	Read Dual Out (3- or 4-byte address)	3B	104
	4DOR	Read Dual Out (4-byte address)	3C	104
	QOR	Read Quad Out (3- or 4-byte address)	6B	104
	4QOR	Read Quad Out (4-byte address)	6C	104
	DIOR	Dual I/O Read (3- or 4-byte address)	BB	104
	4DIOR	Dual I/O Read (4-byte address)	BC	104
	DDRDIOR	DDR Dual I/O Read (3- or 4-byte address)	BD	80
	4DDRDIOR	DDR Dual I/O Read (4-byte address)	BE	80

(table continues...)

## 11 Commands

**Table 53 (continued) S25FL128S and S25FL256S command set (sorted by function)**

Function	Command name	Command description	Instruction value (Hex)	Maximum frequency (MHz)
	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
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 4-KB, sector Erase (3- or 4-byte address)	20	133
	4P4E	Parameter 4-KB, sector Erase (4-byte address)	21	133
	BE	Bulk Erase	60	133
	BE	Bulk Erase (alternate command)	C7	133
	SE	Erase 64 KB or 256 KB (3- or 4-byte address)	D8	133
	4SE	Erase 64 KB or 256 KB (4-byte address)	DC	133
	ERSP	Erase Suspend	75	133

**(table continues...)**

## 11 Commands

**Table 53** (continued) S25FL128S and S25FL256S command set (sorted by function)

Function	Command name	Command description	Instruction value (Hex)	Maximum frequency (MHz)
	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 Perf Mode (MPM)	A3	133
RFU	Reserved-18	Reserved	18	
RFU	Reserved-E5	Reserved	E5	
RFU	Reserved-E6	Reserved	E6	

### 11.1.2 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 S25FL128S and S25FL256S devices support the three most common device information commands.

### 11.1.3 Register read or write

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

## **11 Commands**

### **11.1.3.1 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 should be issued. 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.

### **11.1.3.2 Configuration**

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

### **11.1.4 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 SI signal with read data returning a single bit per SCK falling edge on the SO signal. This command has zero latency between the address and the returning data but is limited to a maximum SCK rate of 50 MHz.
- Other read commands have a latency period between the address and returning data but can operate at higher SCK frequencies. The latency depends on the configuration register latency code.
- The Fast Read command provides a single address bit per SCK rising edge on the SI signal with read data returning a single bit per SCK falling edge on the SO signal and may operate up to 133 MHz.
- Dual or Quad Output read commands provide address a single bit per SCK rising edge on the SI/IO0 signal with read data returning two bits, or four bits of data per SCK falling edge on the IO0-IO3 signals.
- Dual or Quad I/O Read commands provide address two bits or four bits per SCK rising edge with read data returning two bits, or four bits of data per SCK falling edge on the IO0-IO3 signals.
- Fast (Single), Dual, or Quad Double Data Rate read commands provide address one bit, two bits or four bits per every SCK edge with read data returning one bit, two bits, or four bits of data per every SCK edge on the IO0-IO3 signals. Double Data Rate (DDR) operation is only supported for core and I/O voltages of 3 to 3.6V.

### **11.1.5 Program flash array**

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

### **11.1.6 Erase flash array**

The Sector Erase (SE) and Bulk Erase (BE) commands set all the bits in a sector or the entire memory array to 1. A bit needs to be first erased to 1 before programming can change it to a 0. While bits can be individually programmed from a 1 to 0, erasing bits from 0 to 1 must be done on a sector-wide (SE) or array-wide (BE) level.

## **11 Commands**

### **11.1.7 OTP, block protection, and advanced sector protection**

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

### **11.1.8 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.

### **11.1.9 Reserved**

Some instructions are reserved for future use. In this generation of the S25FL128S and S25FL256S 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 S25FL128S and S25FL256S 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.

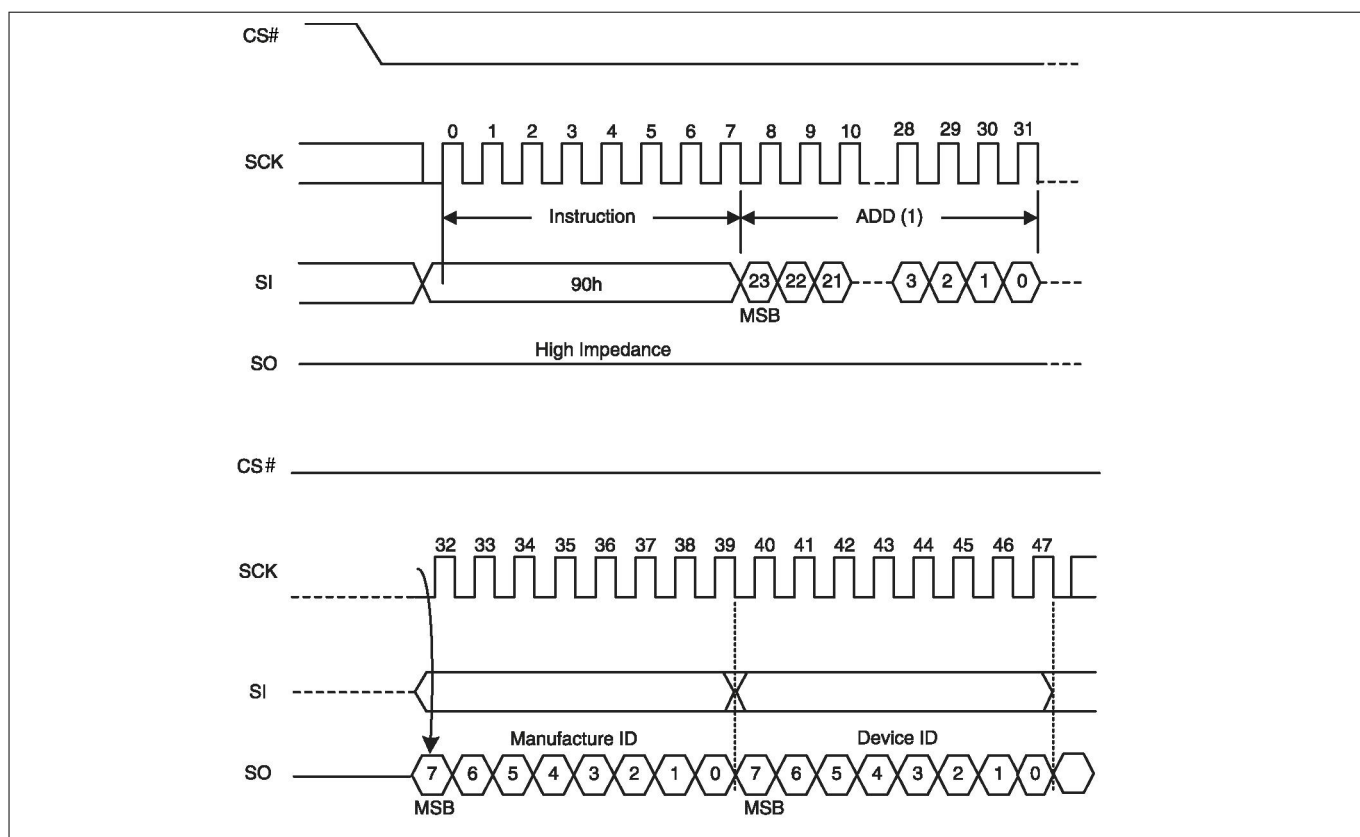
## **11.2 Identification commands**

### **11.2.1 Read Identification - REMS (Read\_ID or REMS 90 h)**

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 “90 h” followed by a 24-bit address of 00000 h. 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 000001 h, 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.

## 11 Commands



**Figure 44** **READ\_ID command sequence**

**Table 54** **Read\_ID values**

Device	Manufacturer ID (hex)	Device ID (hex)
S25FL128S	01	17
S25FL256S	01	18

### 11.2.2 Read Identification (RDID 9 Fh)

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 Infineon.

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

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

The RDID instruction is shifted on SI. After the last bit of the RDID instruction is shifted into the device, a byte of manufacturer identification, two bytes of device identification, extended device identification, and CFI information will be shifted sequentially out on SO. As a whole this information is referred to as ID-CFI. See [ID-CFI address space](#) for the detail description of the ID-CFI contents.



## 11 Commands

Continued shifting of output beyond the end of the defined ID-CFI address space will provide undefined data. The RDID command sequence is terminated by driving CS# to the logic high state anytime during data output. The maximum clock frequency for the RDID command is 133 MHz.

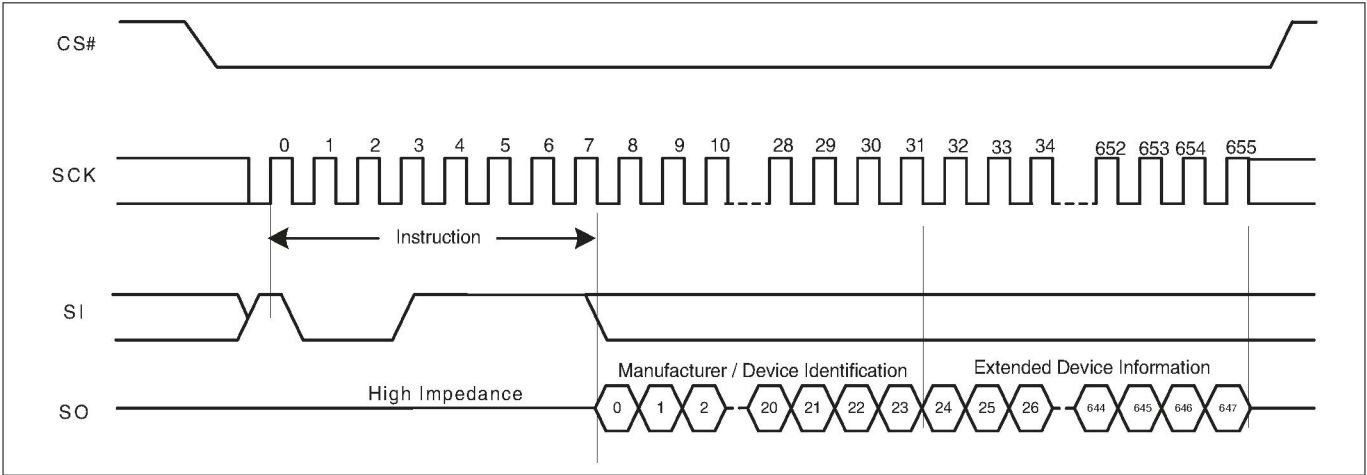


Figure 45 Read Identification (RDID) command sequence

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

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

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

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

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

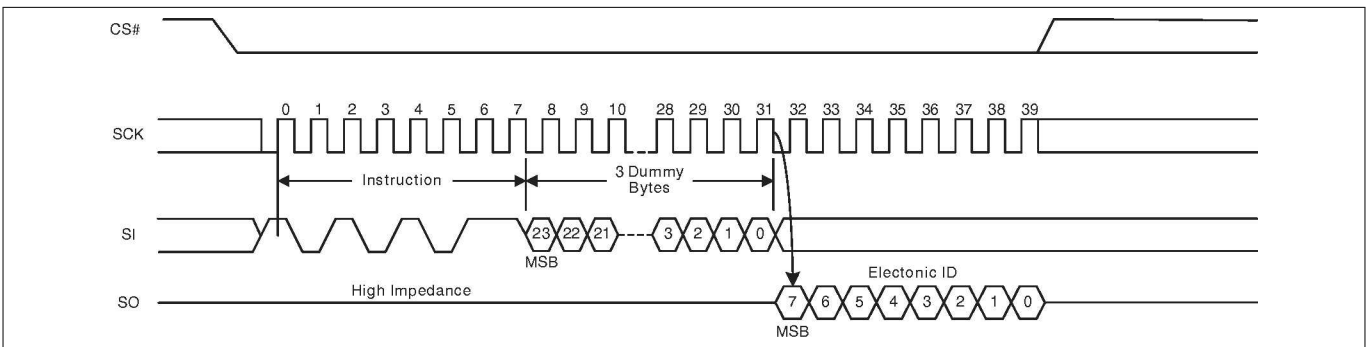


Figure 46 Read Electronic Signature (RES) command sequence

Table 55 RES values

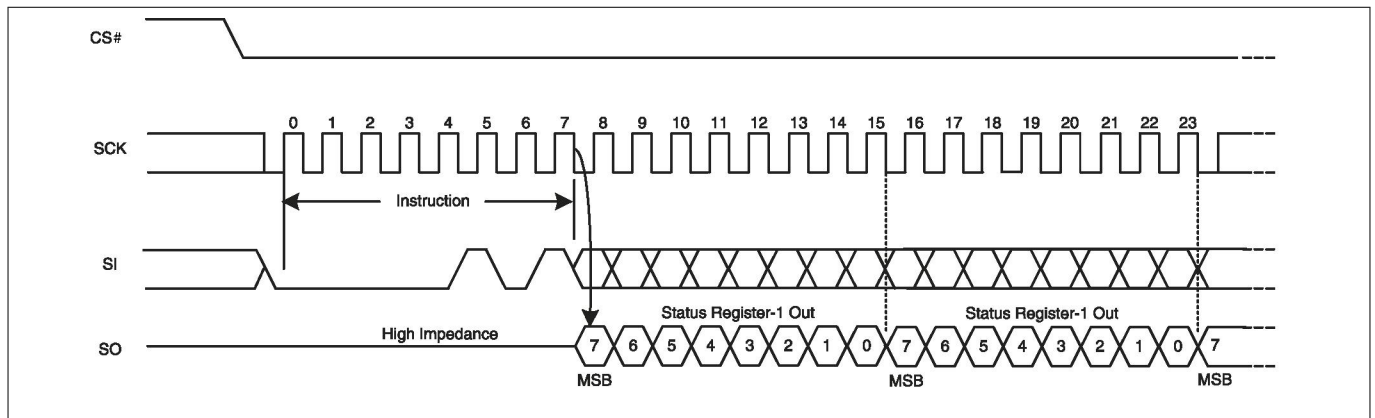
Device	Device ID (hex)
S25FL128S	17
S25FL256S	18

## 11 Commands

### 11.3 Register access commands

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

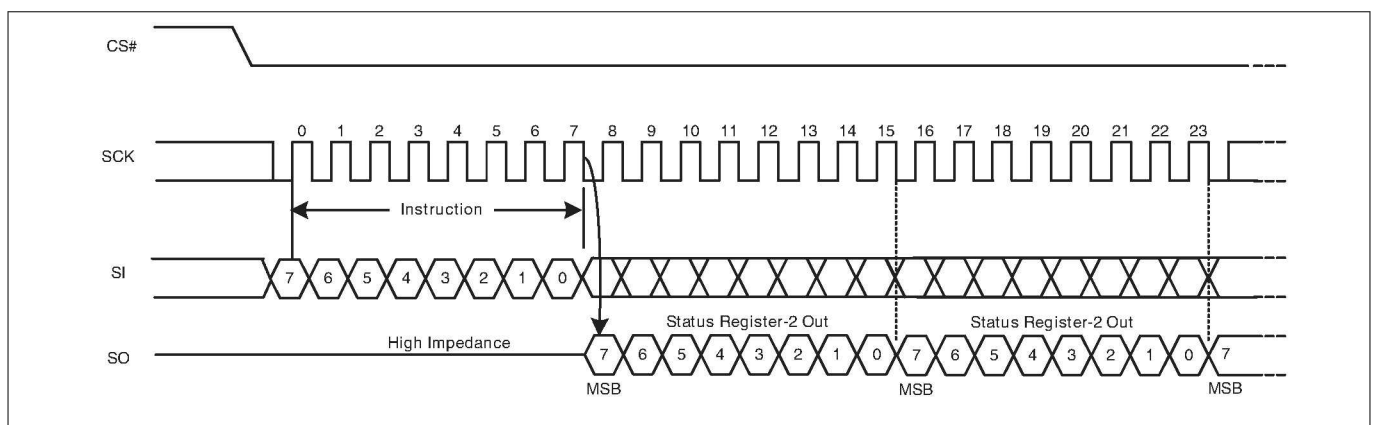
The Read Status Register-1 (RDSR1) command allows the Status Register-1 contents to be read from SO. The Status Register-1 contents may be read at any time, even while a program, erase, or write operation is in progress. It is possible to read the Status Register-1 continuously by providing multiples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



**Figure 47** Read Status Register-1 (RDSR1) command sequence

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

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

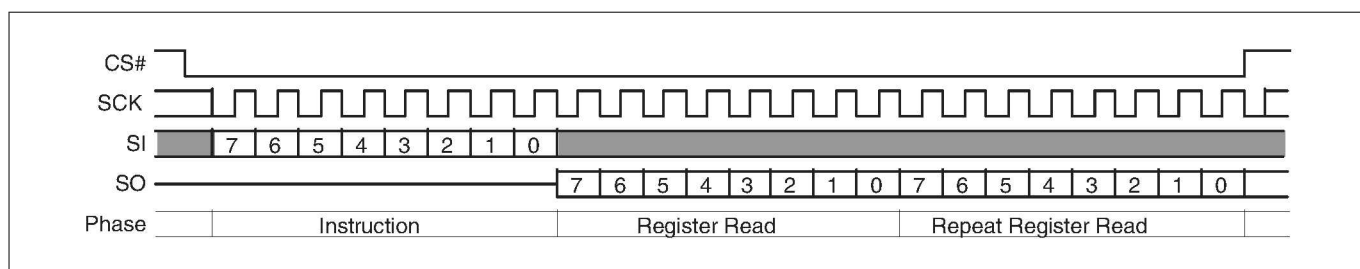


**Figure 48** Read Status Register-2 (RDSR2) command

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

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

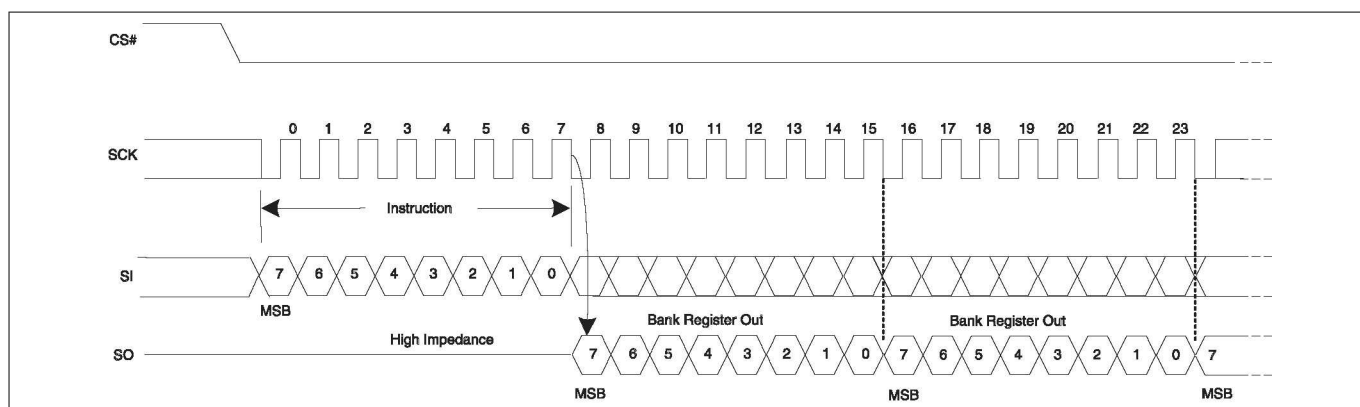
## 11 Commands



**Figure 49** Read Configuration Register (RDCR) command sequence

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

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.



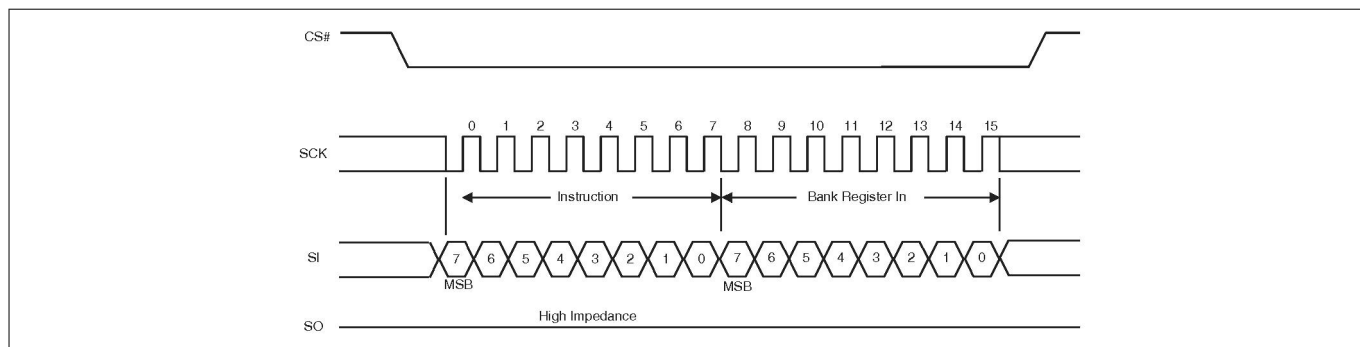
**Figure 50** Read Bank Register (BRRD) command

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

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 a '0'.

## 11 Commands



**Figure 51** Bank Register Write (BRWR) command

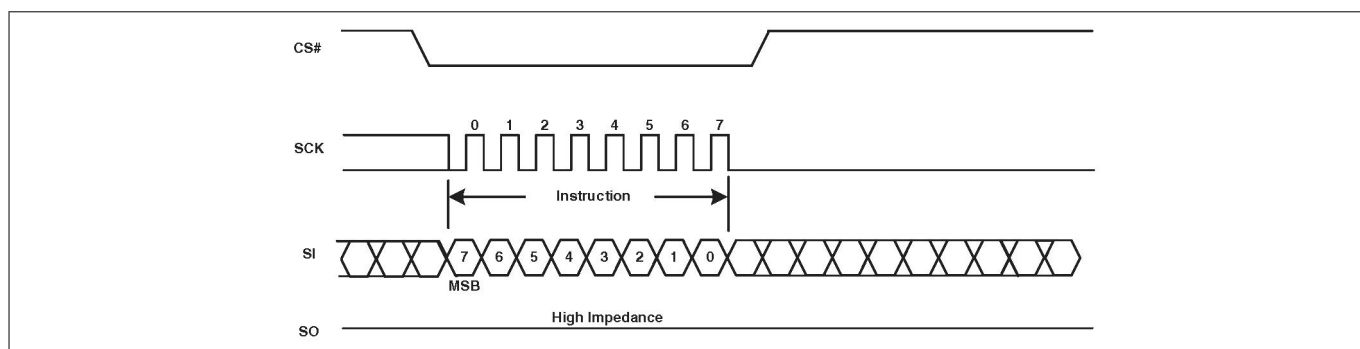
### 11.3.6 Bank Register Access (BRAC B9h)

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

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

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

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



**Figure 52** BRAC (B9h) command sequence

## 11 Commands

### 11.3.7 Write registers (WRR 01 h)

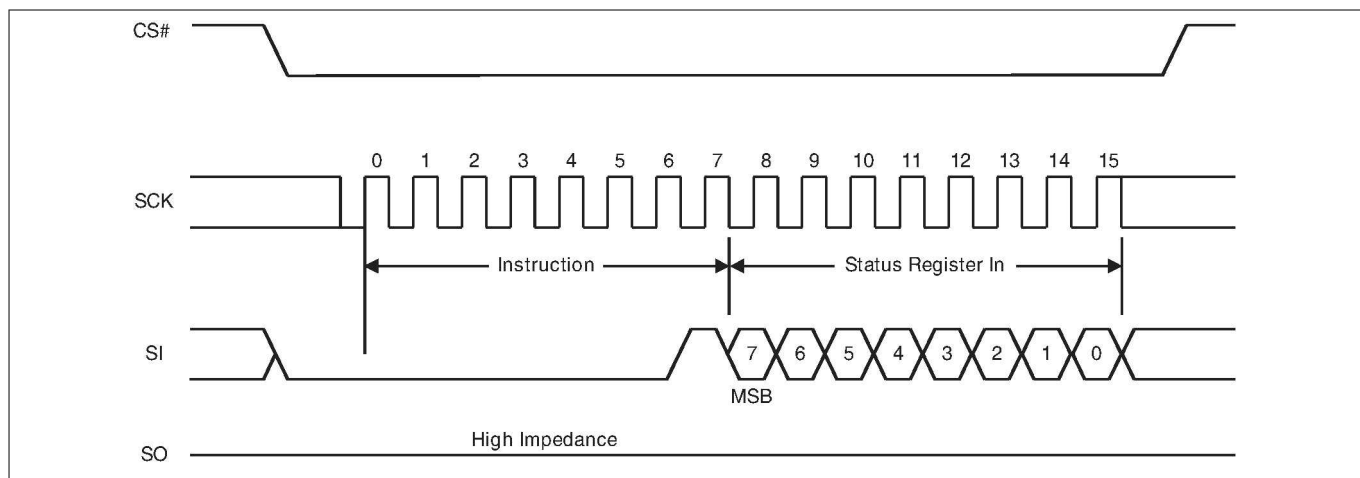
The Write Registers (WRR) command allows new values to be written to both the Status Register-1 and Configuration Register. Before the Write Registers (WRR) command can be accepted by the device, a Write Enable (WREN) command must be received. After the Write Enable (WREN) command has been decoded successfully, 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 on SI. The Status Register is one data byte in length.

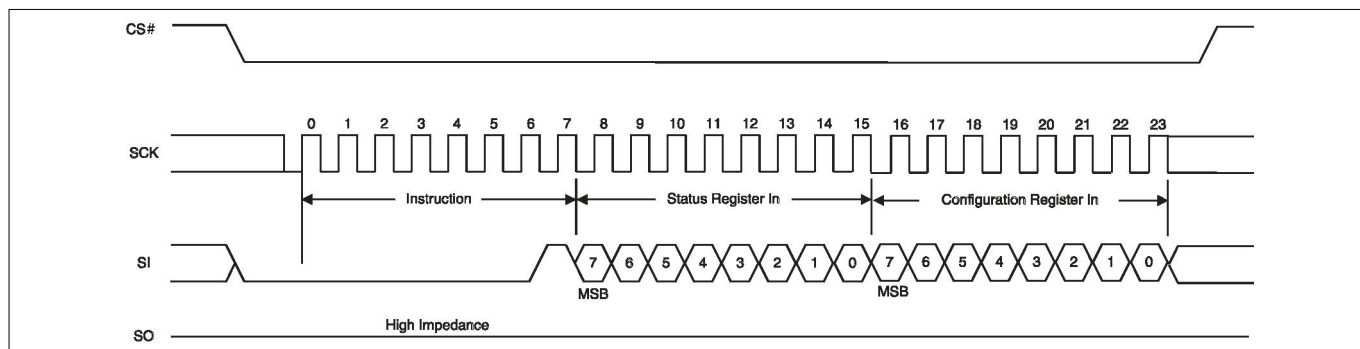
The Write Registers (WRR) command will set the P\_ERR or E\_ERR bits if there is a failure in the WRR operation. Any Status or Configuration Register bit reserved for the future must be written as a 0.

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



**Figure 53 Write Registers (WRR) command sequence – 8 data bits**



**Figure 54 Write Registers (WRR) command sequence – 16 data bits**

The Write Registers (WRR) command allows the user to change the values of the Block Protect (BP2, BP1, and BP0) bits to define the size of the area that is to be treated as read-only. The Write Registers (WRR) command

## 11 Commands

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

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

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

- If Write Protect (WP#) signal is driven to the logic high state, it is possible to write to the Status and Configuration Registers provided that the Write Enable Latch (WEL) bit has previously been set to a 1 by initiating a Write Enable (WREN) command.
- If Write Protect (WP#) signal is driven to the logic low state, it is not possible to write to the Status and Configuration Registers even if the Write Enable Latch (WEL) bit has previously been set to a 1 by a Write Enable (WREN) command. Attempts to write to the Status and Configuration Registers are rejected, and are not accepted for execution. As a consequence, all the data bytes in the memory area that are protected by the Block Protect (BP2, BP1, BP0) bits of the Status Register, are also hardware protected by WP#.

The WP# hardware protection can be provided:

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

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

**Table 56 Block Protection modes**

WP#	SRWD bit	Mode	Write Protection of Registers	Memory content	
				Protected area	Unprotected area
1	1	Software Protected	Status and Configuration Registers are Writable (if WREN command has set the WEL bit). The values in the SRWD, BP2, BP1, and BP0 bits and those in the Configuration Register can be changed	Protected against Page Program, Quad Input Program, Sector Erase, and Bulk Erase	Ready to accept Page Program, Quad Input Program and Sector Erase commands
1	0				
0	0				

(table continues...)

## 11 Commands

**Table 56** (continued) **Block Protection modes**

WP#	SRWD bit	Mode	Write Protection of Registers	Memory content	
				Protected area	Unprotected area
0	1	Hardware Protected	Status and Configuration Registers are Hardware Write Protected. The values in the SRWD, BP2, BP1, and BP0 bits and those in the Configuration Register cannot be changed	Protected against Page Program, Sector Erase, and Bulk Erase	Ready to accept Page Program or Erase commands

### Notes:

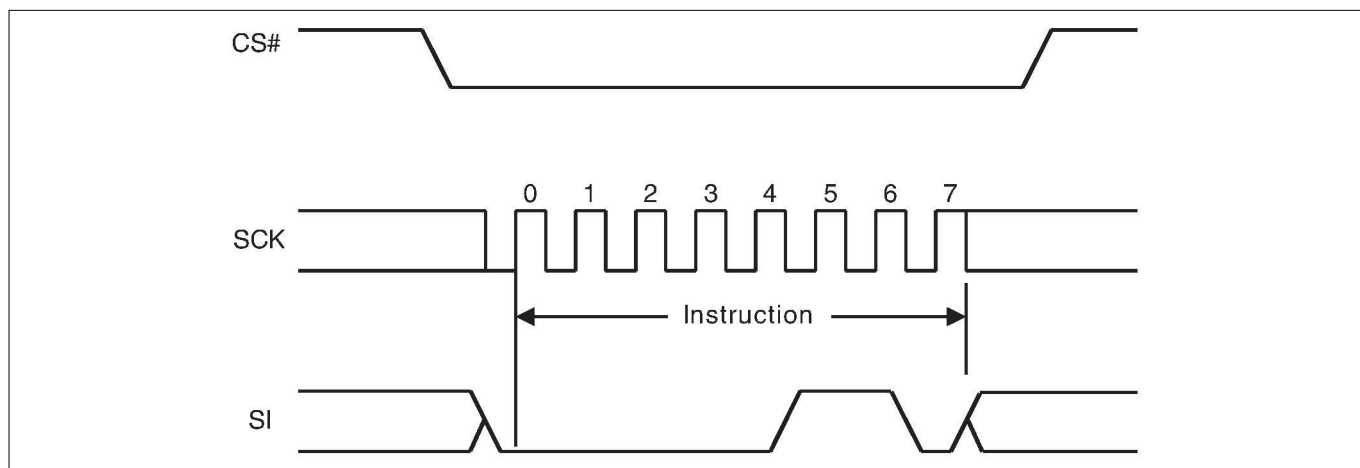
1. The Status Register originally shows 00 h when the device is first shipped from Infineon to the customer.
2. Hardware protection is disabled when Quad Mode is enabled (QUAD bit = 1 in Configuration Register). WP# becomes IO2; therefore, it cannot be utilized.

The WRR command has an alternate function of loading the Bank Address Register if the command immediately follows a BRAC command. See [Bank Register Access \(BRAC B9h\)](#).

### 11.3.8 Write Enable (WREN 06 h)

The Write Enable (WREN) command sets the Write Enable Latch (WEL) bit of the Status Register 1 (SR1[1]) to a '1'. The Write Enable Latch (WEL) bit must be set to a '1' by issuing the Write Enable (WREN) command to enable write, program and erase commands.

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



**Figure 55** Write Enable (WREN) command sequence



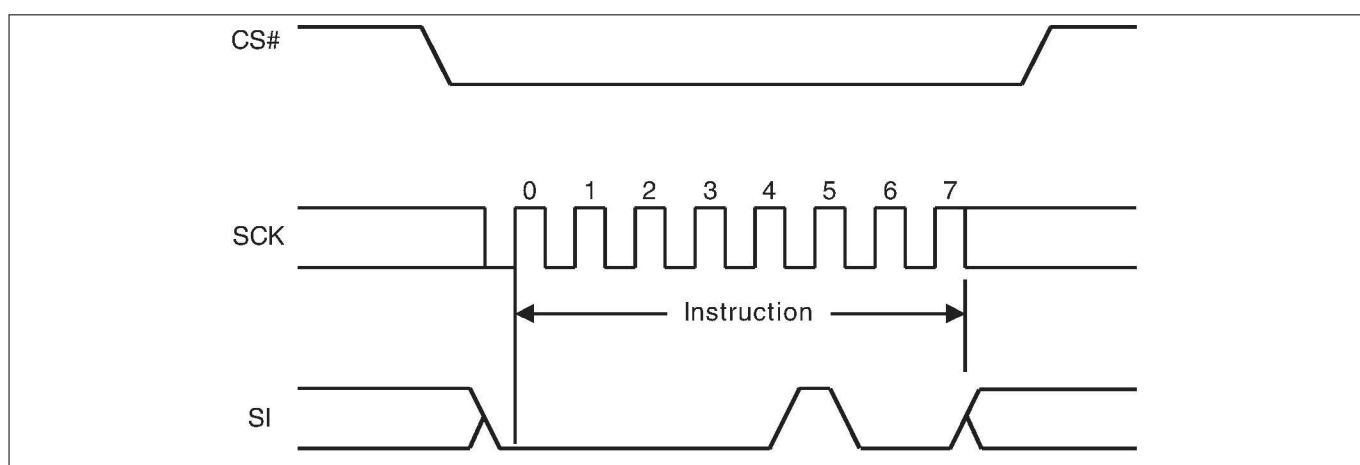
## 11 Commands

### 11.3.9 Write Disable (WRDI 04 h)

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

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

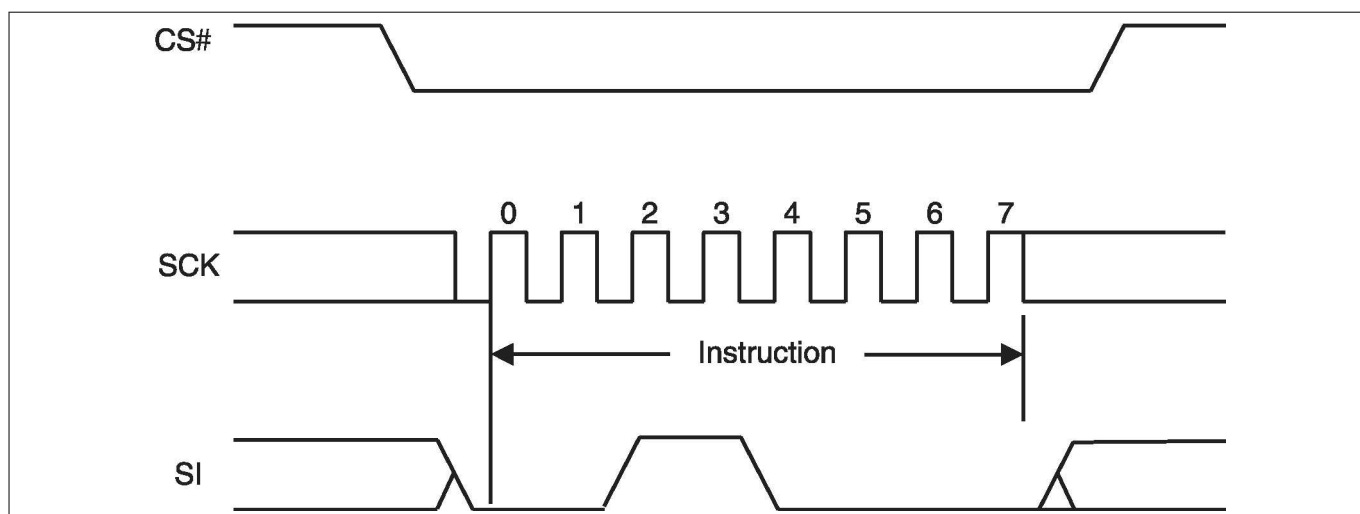
CS# must be driven into the logic high state after the eighth bit of the instruction byte has been latched in on SI. Without CS# being driven to the logic high state after the eighth bit of the instruction byte has been latched in on SI, the write disable operation will not be executed.



**Figure 56** Write Disable (WRDI) command sequence

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

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.



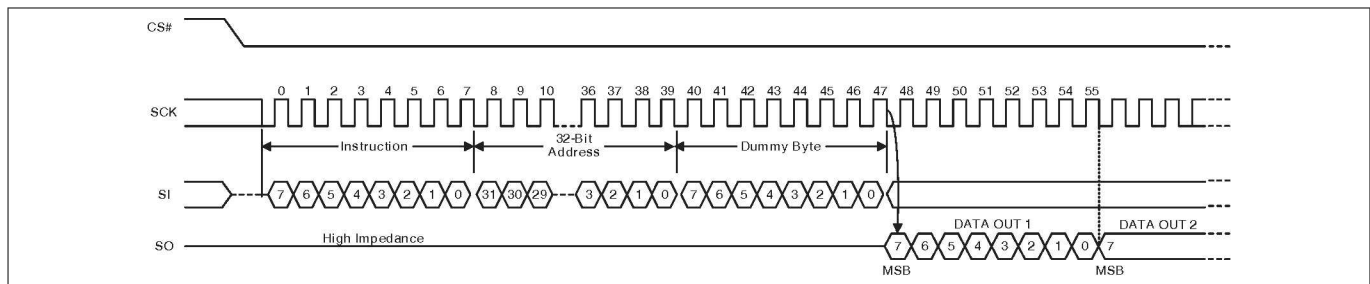
**Figure 57** Clear Status Register (CLSR) command sequence



## 11 Commands

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

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 zero. This is followed by eight dummy cycles. Then the 8-bit contents of the ECC Register, for the ECC unit selected, are shifted out on SO 16 times, once for each byte in the ECC Unit. If CS# remains low the next ECC unit status is sent through SO 16 times, once for each byte in the ECC Unit, this continues until CS# goes high. The maximum operating clock frequency for the ECC READ command is 133 MHz. See [Automatic ECC](#) for details on ECC unit.



**Figure 58** ECC Status Register Read command sequence

### 11.3.12 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 S25FL128S and S25FL256S device immediately after the end of reset, without having to send a read command. This saves 32 or more cycles and simplifies the logic needed to initiate the reading of boot code.

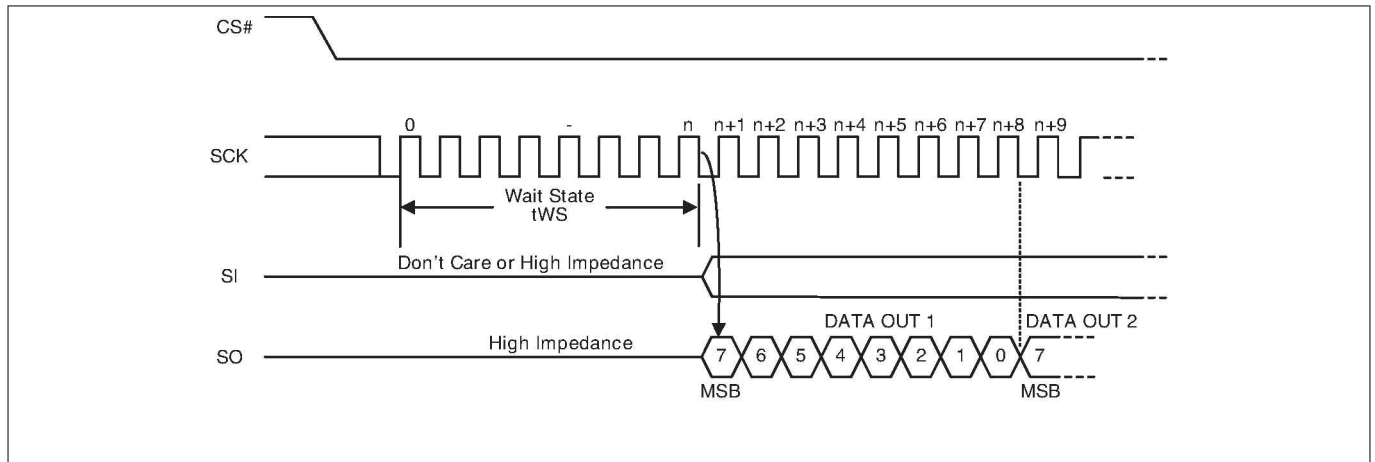
- As part of the power up reset, hardware reset, or command reset process the AutoBoot feature automatically starts a read access from a pre-specified address. At the time the reset process is completed, the device is ready to deliver code from the starting address. The host memory controller only needs to drive CS# signal from high to low and begin toggling the SCK signal. The S25FL128S and S25FL256S 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.

## 11 Commands

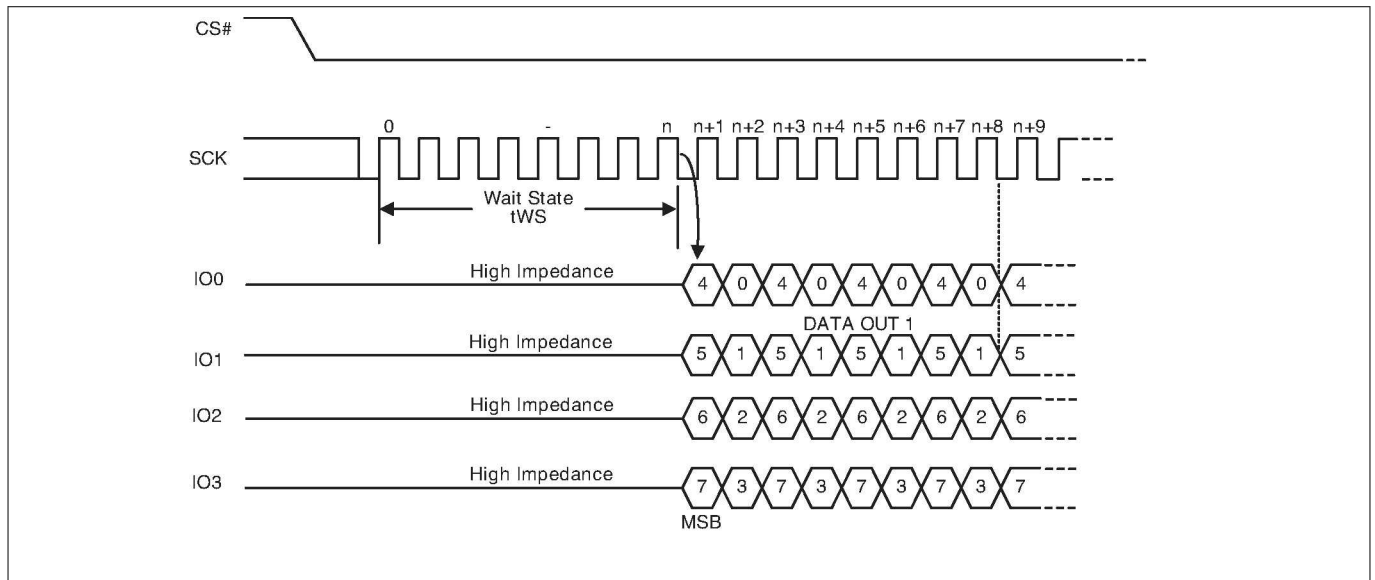
The AutoBoot register bits are non-volatile and provide:

- 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-Gb.
- The number of initial delay cycles, set by the AutoBoot Start Delay (ABSD) 8-bit count value.
- The AutoBoot Enable.

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



**Figure 59** AutoBoot sequence (CR1[1]=0)

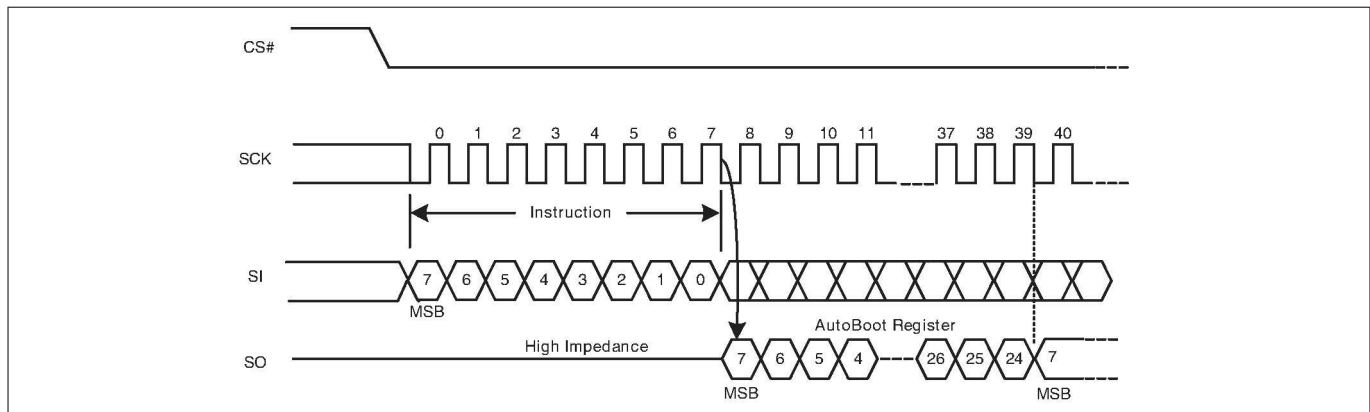


**Figure 60** AutoBoot sequence (CR1[1]=1)

### 11.3.13 AutoBoot Register Read (ABRD 14 h)

The AutoBoot Register Read command is shifted into SI. Then the 32-bit AutoBoot Register is shifted out on SO, least significant byte first, most significant bit of each byte first. It is possible to read the AutoBoot Register continuously by providing multiples of 32 clock cycles. If the QUAD bit CR1[1] is cleared to '0', the maximum operating clock frequency for ABRD command is 133 MHz. If the QUAD bit CR1[1] is set to '1', the maximum operating clock frequency for ABRD command is 104 MHz.

## 11 Commands



**Figure 61 AutoBoot Register Read (ABRD) command**

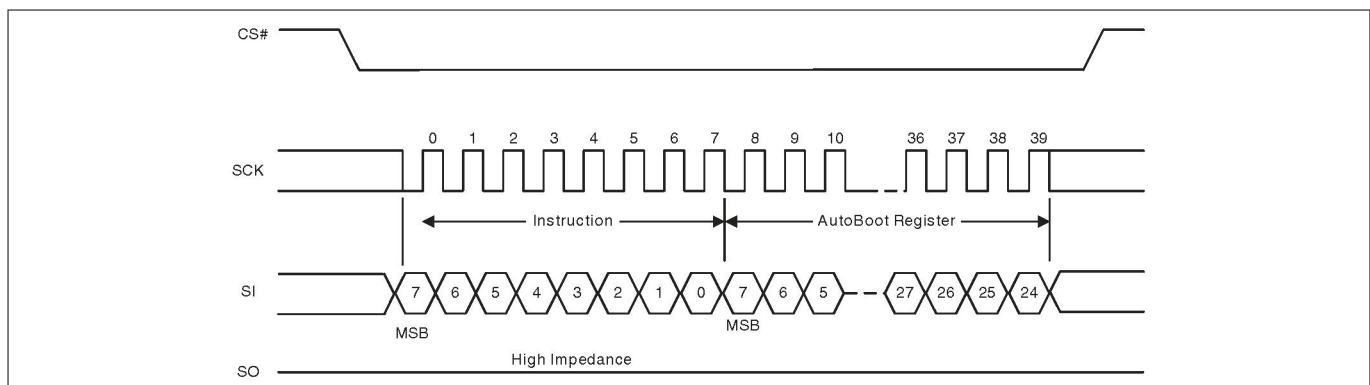
### 11.3.14 AutoBoot Register Write (ABWR 15 h)

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

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

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

CS# must be driven to the logic high state after the 32nd bit of data has been latched. If not, the ABWR command is not executed. As soon as CS# is driven to the logic high state, the self-timed ABWR operation is initiated. While the ABWR operation is in progress, Status Register-1 may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a '1' during the self-timed ABWR operation, and is a 0, when it is completed. When the ABWR cycle is completed, the Write Enable Latch (WEL) is set to a '0'. The maximum clock frequency for the ABWR command is 133 MHz.



**Figure 62 AutoBoot Register Write (ABWR) Command**

### 11.3.15 Program NVDLR (PNVDLR 43 h)

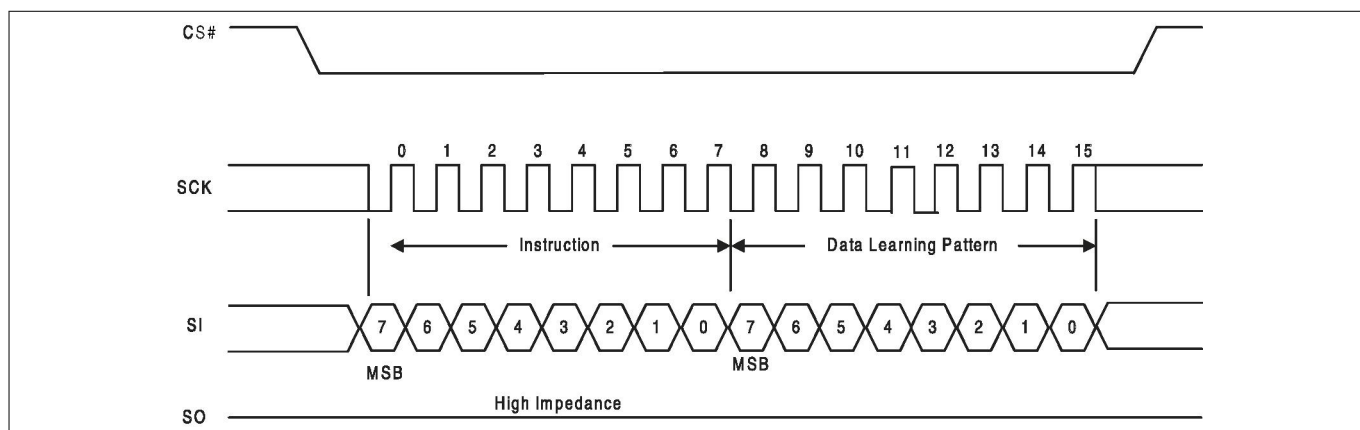
Before the Program NVDLR (PNVDLR) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device. After the Write Enable (WREN) command has been decoded successfully, the device will set the Write Enable Latch (WEL) to enable the PNVDLR operation.

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

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

## 11 Commands

initiated. While the PNVDLR operation is in progress, the Status Register may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a 1 during the self-timed PNVDLR cycle, and is a 0 when it is completed. The PNVDLR operation can report a program error in the P\_ERR bit of the status register. When the PNVDLR operation is completed, the Write Enable Latch (WEL) is set to a 0. The maximum clock frequency for the PNVDLR command is 133 MHz.



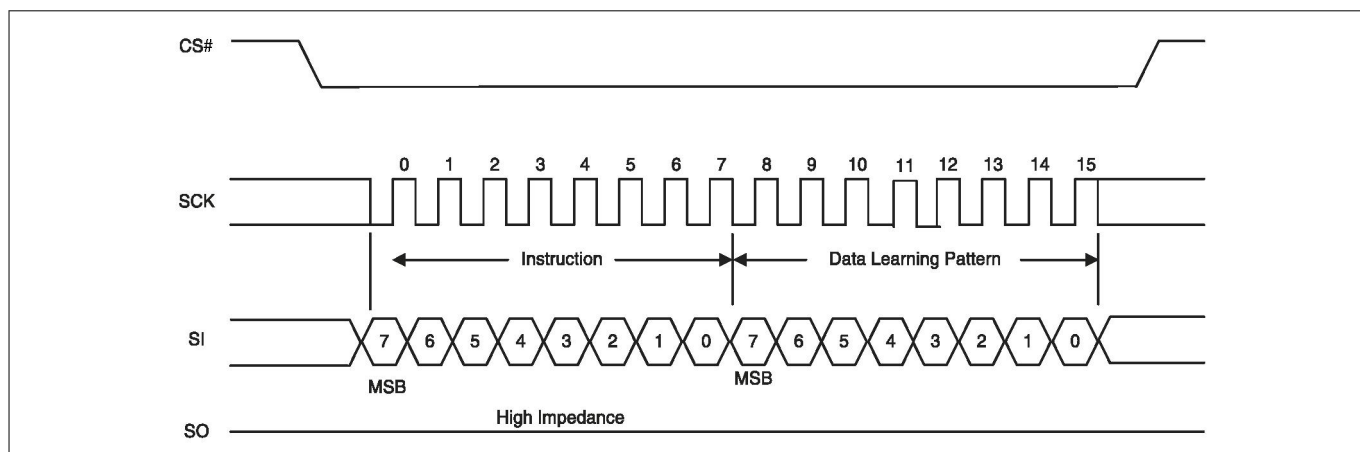
**Figure 63** Program NVDLR (PNVDLR) command sequence

### 11.3.16 Write VDLR (WVDLR 4 Ah)

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.

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.

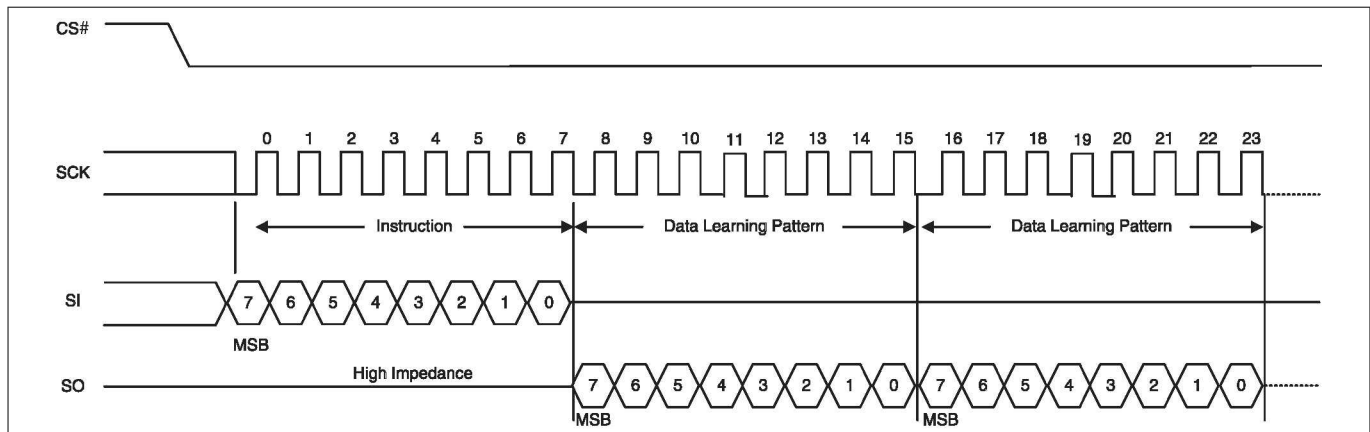


**Figure 64** Write VDLR (WVDLR) command sequence

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

The instruction is shifted on SI, then the 8-bit DLP is shifted out on SO. 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.

## 11 Commands



**Figure 65** DLP Read (DLPRD) command sequence

### 11.4 Read memory array commands

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

- Some commands transfer address or data on each rising edge of SCK. These are called Single Data Rate commands (SDR).
- Some SDR commands transfer address one bit per rising edge of SCK and return data 1-, 2-, or 4-bits of data per rising edge of SCK. These are called Read or Fast Read for 1-bit data; Dual Output Read for 2-bit data, and Quad Output for 4-bit data.
- Some SDR commands transfer both address and data 2- or 4-bits per rising edge of SCK. These are called Dual I/O for 2-bit and Quad I/O for 4-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 1-, 2-, or 4-bits of address or data per SCK edge. These are called Fast DDR for 1-bit, Dual I/O DDR for 2-bit, and Quad I/O DDR for 4-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 2- or 4-bits per clock edge 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 devices 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 zero. Commands are provided to load and read the bank address register. These devices may also be configured to take a 4-byte address from the host system with the traditional 3-byte address commands. The 4-byte address mode for traditional commands is activated by setting the External Address (EXTADD) bit in the bank address register to '1'. In the FL128S, higher order address bits above A23 in the 4-byte address commands, commands using Extended Address mode, and the Bank Address Register are not relevant and are ignored because the flash array is only 128 Mb in size.

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

A device ordering option provides an enhanced high performance option by adding a similar mode bit scheme to the DDR Fast Read, Dual I/O, and Dual I/O DDR commands, in addition to the Quad I/O command.

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

## 11 Commands

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

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

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

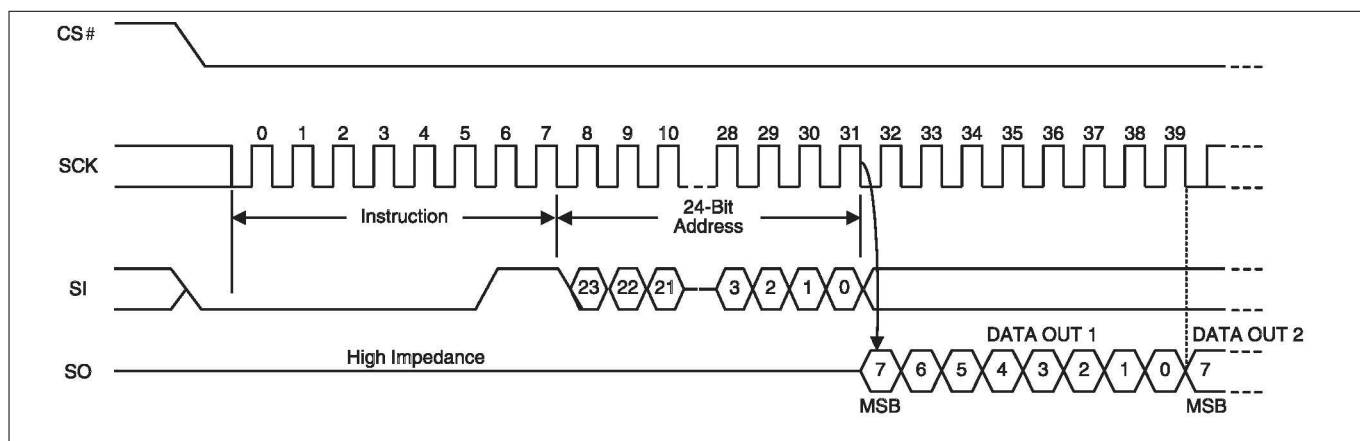
### 11.4.1 Read (Read 03 h or 4READ 13 h)

The instruction

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

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

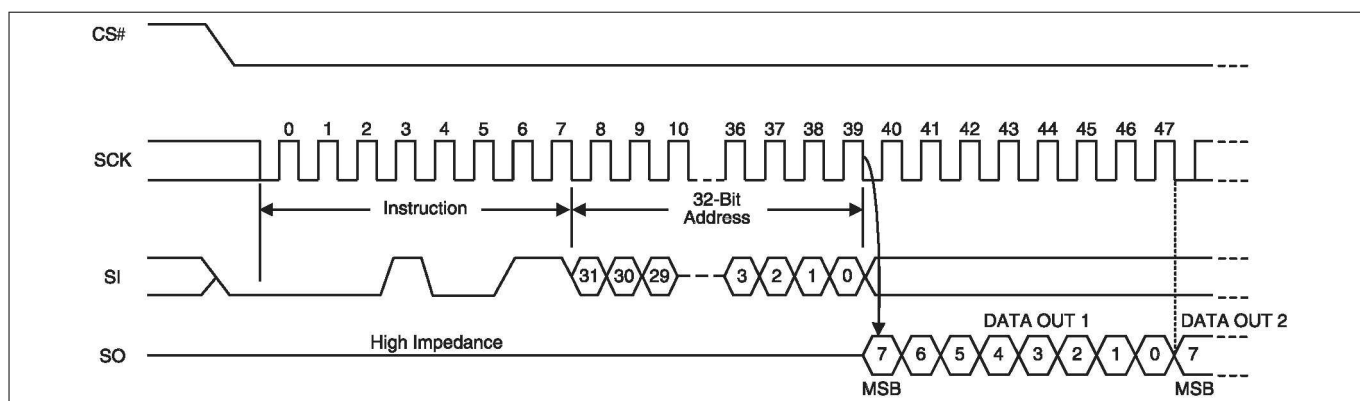
The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When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



**Figure 66** Read command sequence (3-byte address, 03 h [ExtAdd=0])



## 11 Commands



**Figure 67** Read command sequence (4-byte address, 13 h or 03 h [ExtAdd=1])

### 11.4.2 Fast Read (FAST\_READ 0 Bh or 4FAST\_READ 0 Ch)

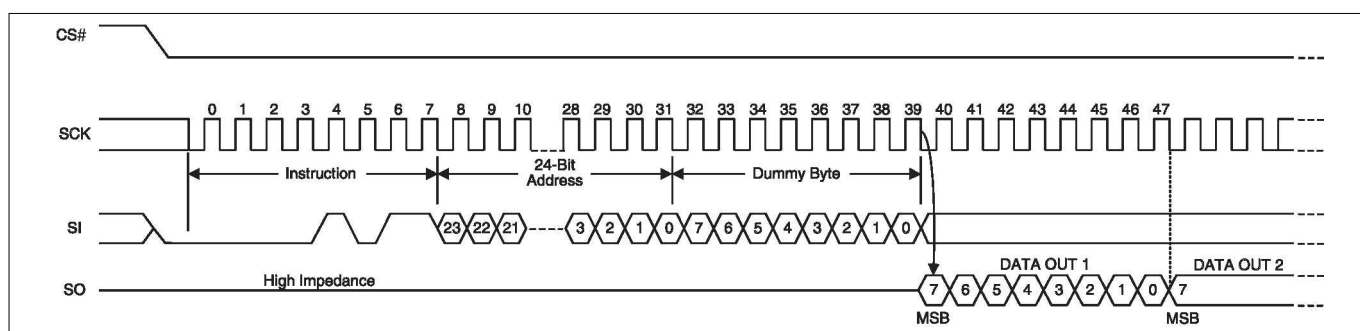
The instruction

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

The address is followed by zero or eight dummy cycles depending on the latency code set in the Configuration Register. The dummy cycles allow the device internal circuits additional time for accessing the initial address location. During the dummy cycles the data value on SO is “don't care” and may be high impedance. Then the memory contents, at the address given, are shifted out on SO.

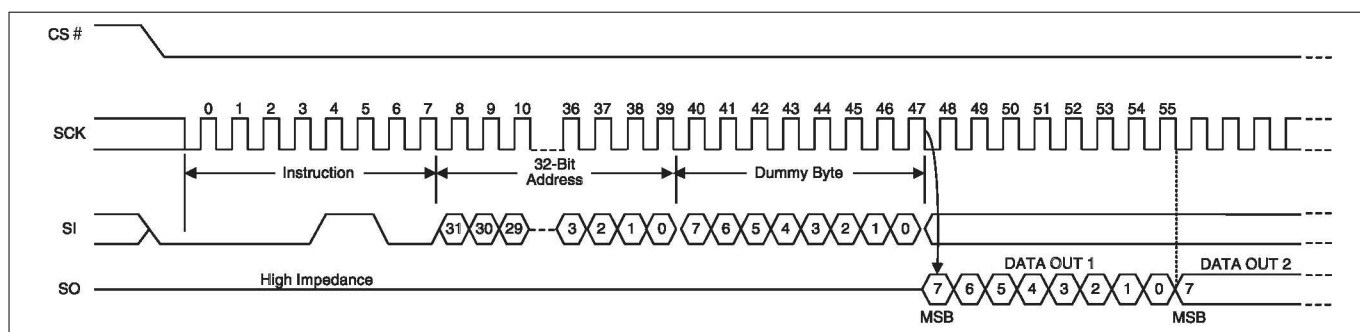
The maximum operating clock frequency for FAST READ command is 133 MHz.

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.

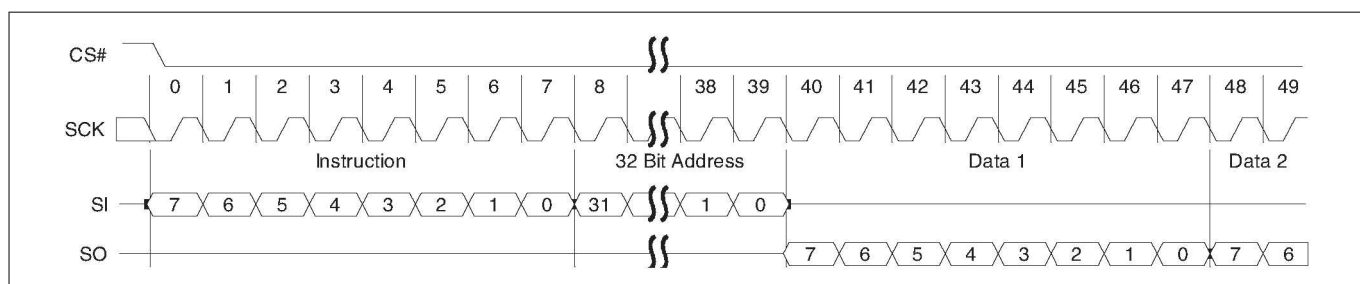


**Figure 68** Fast Read (FAST\_READ) command sequence (3-byte address, 0 Bh [ExtAdd = 0, LC = 10b])

## 11 Commands



**Figure 69** Fast Read command sequence (4-byte address, 0 Ch or 0 B [ExtAdd=1], LC = 10b)



**Figure 70** Fast Read command sequence (4-byte address, 0 Ch or 0 B [ExtAdd=1], LC = 11b)

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

The instruction

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

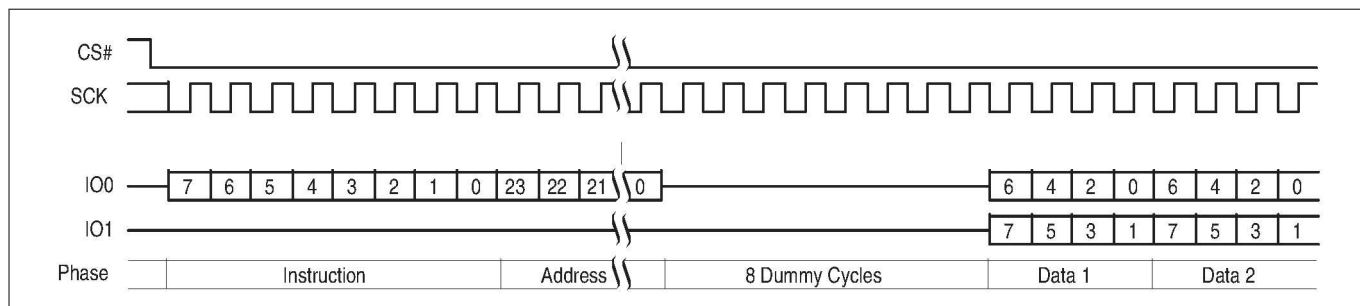
Then the memory contents, at the address given, is shifted out two bits at a time through IO0 (SI) and IO1 (SO). Two bits are shifted out at the SCK frequency by the falling edge of the SCK signal.

The maximum operating clock frequency for the Dual Output Read command is 104 MHz. For Dual Output Read commands, there are zero or eight dummy cycles required after the last address bit is shifted into SI before data begins shifting out of IO0 and IO1. This latency period (i.e., dummy cycles) allows the device's internal circuitry enough time to read from the initial address. During the dummy cycles, the data value on SI is a "don't care" and may be high impedance. The number of dummy cycles is determined by the frequency of SCK (refer to [Table 33](#)).

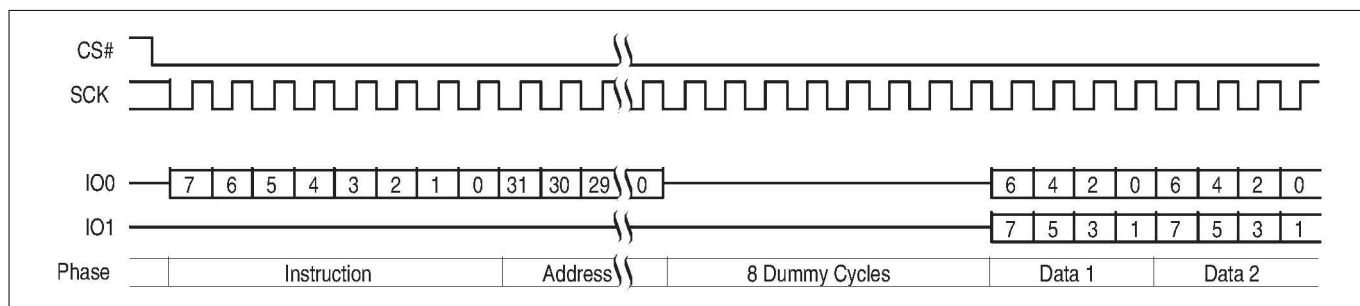
The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When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



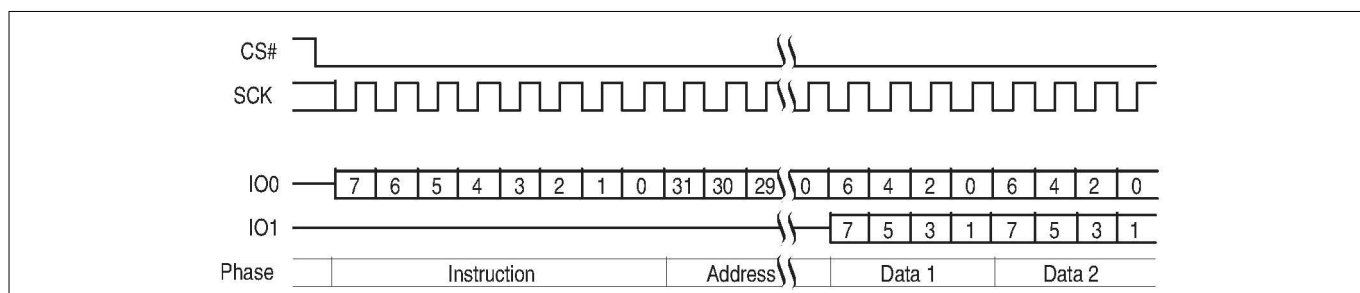
## 11 Commands



**Figure 71** Dual Output Read command sequence (3-byte address, 3 Bh [ExtAdd = 0], LC = 10b)



**Figure 72** Dual Output Read command sequence (4-byte address, 3 Ch or 3 Bh [ExtAdd = 1, LC = 10b])



**Figure 73** Dual Output Read command sequence (4-byte Address, 3 Ch or 3 Bh [ExtAdd = 1, LC = 11b])

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

The instruction

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

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

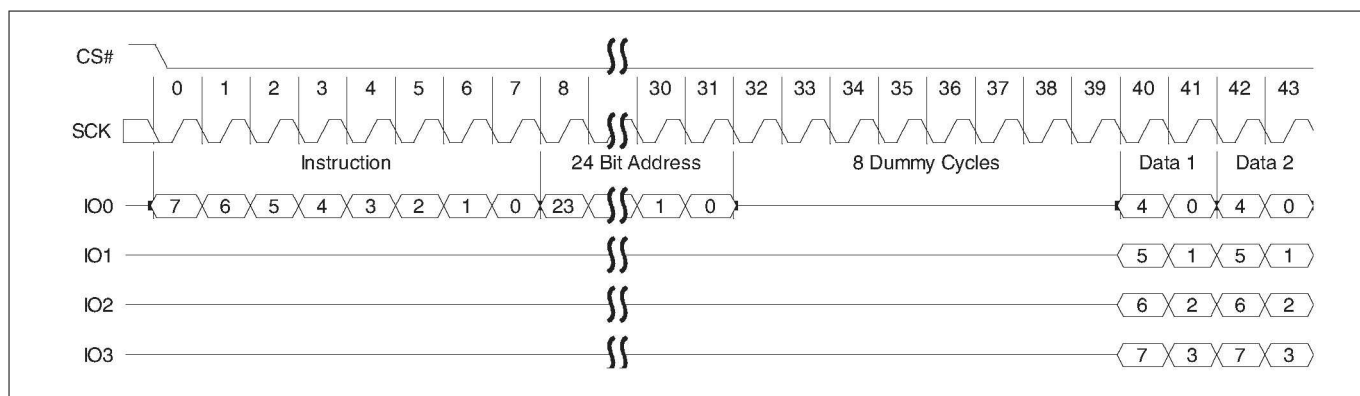
The maximum operating clock frequency for Quad Output Read command is 104 MHz. For Quad Output Read Mode, there may be dummy cycles required after the last address bit is shifted into SI before data begins shifting out of IO0-IO3. This latency period (i.e., dummy cycles) allows the device's internal circuitry enough time to set up for the initial address. During the dummy cycles, the data value on IO0-IO3 is a "don't care" and may be high impedance. The number of dummy cycles is determined by the frequency of SCK (refer to [Table 33](#)).

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

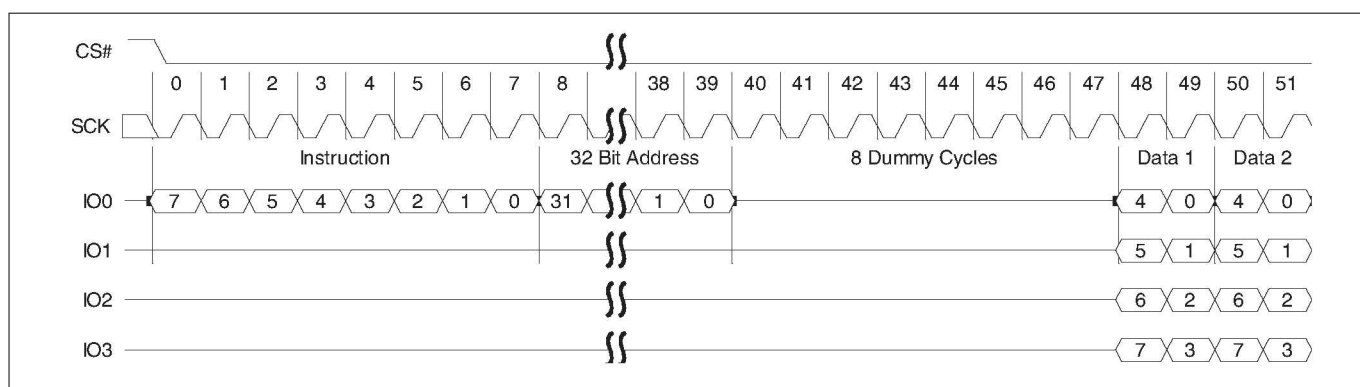
## 11 Commands

therefore be read out with one single read instruction and address 000000 h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000 h, allowing the read sequence to be continued indefinitely.

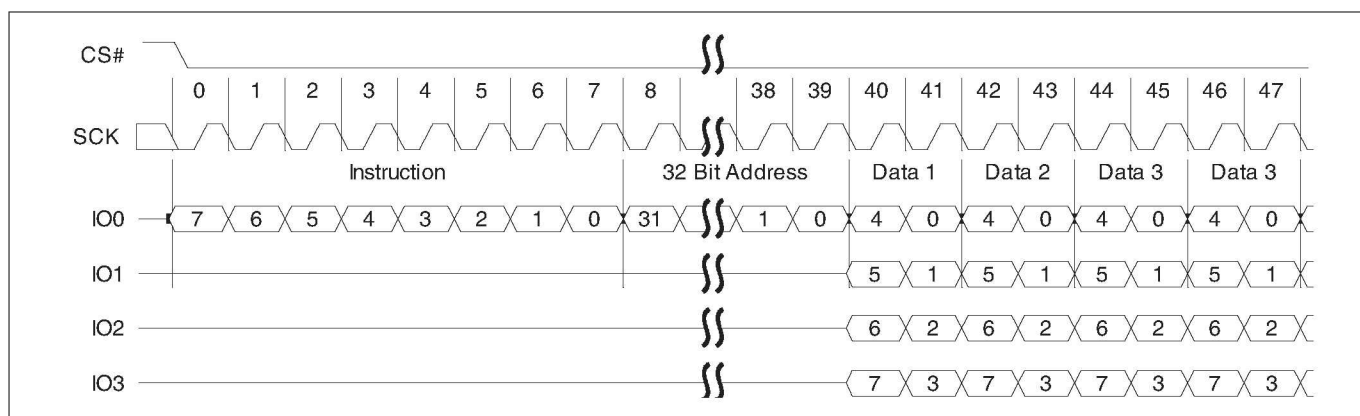
The QUAD bit of Configuration Register must be set (CR Bit1=1) to enable the Quad mode capability.



**Figure 74** Quad Output Read command sequence (3-byte address, 6 Bh [ExtAdd = 0, LC = 01b])



**Figure 75** Quad Output Read command sequence (4-byte address, 6 Ch or 6 Bh [ExtAdd = 1, LC = 01b])



**Figure 76** Quad Output Read command sequence (4-byte address, 6 Ch or 6 Bh [ExtAdd = 1, LC = 11b])

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

The instruction

- BBh (ExtAdd=0) is followed by a 3-byte address (A23-A0) or

## 11 Commands

- BBh (ExtAdd=1) is followed by a 4-byte address (A31-A0) or
- BCh is followed by a 4-byte address (A31-A0)

The Dual I/O Read commands improve throughput with two I/O signals — IO0 (SI) and IO1 (SO). It is similar to the Dual Output Read command but takes input of the address two bits per SCK rising edge. In some applications, the reduced address input time might allow for code execution in place (XIP) i.e., directly from the memory device.

The maximum operating clock frequency for Dual I/O Read is 104 MHz.

For the Dual I/O Read command, there is a latency required after the last address bits are shifted into SI and SO before data begins shifting out of IO0 and IO1. There are different ordering part numbers that select the latency code table used for this command, either the High Performance LC (HPLC) table or the Enhanced High Performance LC (EHPLC) table. The HPLC table does not provide cycles for mode bits so each Dual I/O Read command starts with the 8 bit instruction, followed by address, followed by a latency period.

This latency period (dummy cycles) allows the device internal circuitry enough time to access data at the initial address. During the dummy cycles, the data value on SI and SO are “don’t care” and may be high impedance. The number of dummy cycles is determined by the frequency of SCK (Table 33). The number of dummy cycles is set by the LC bits in the Configuration Register (CR1).

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

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

- During the Dual I/O Enhanced High Performance Command Sequence, if the Mode bits are any value other than Axh, then the next time CS# is raised high the device will be released from Dual I/O Read Enhanced High Performance Read mode.

During any operation, if CS# toggles high to low to high for eight cycles (or less) and data input (IO0 and IO1) are not set for a valid instruction sequence, then the device will be released from Dual I/O Enhanced High Performance Read mode. Note that the four mode bit cycles are part of the device’s internal circuitry latency time to access the initial address after the last address cycle that is clocked into IO0 (SI) and IO1 (SO).

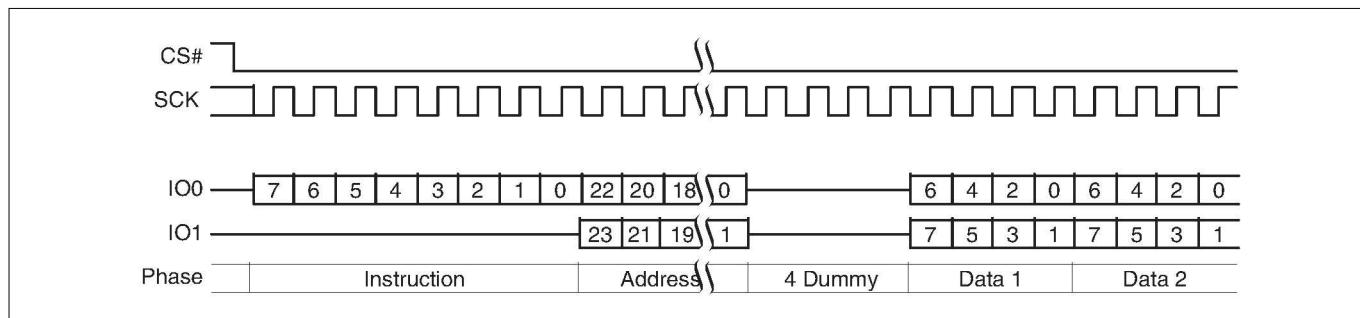
It is important that the I/O signals be set to high-impedance at or before the falling edge of the first data out clock. At higher clock speeds the time available to turn off the host outputs before the memory device begins to drive (bus turn around) is diminished. It is allowed and may be helpful in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

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

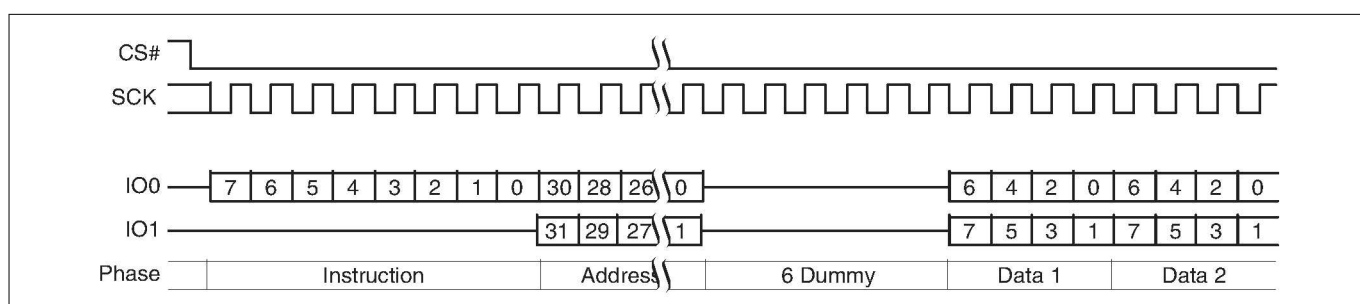
The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000 h provided. When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

## 11 Commands

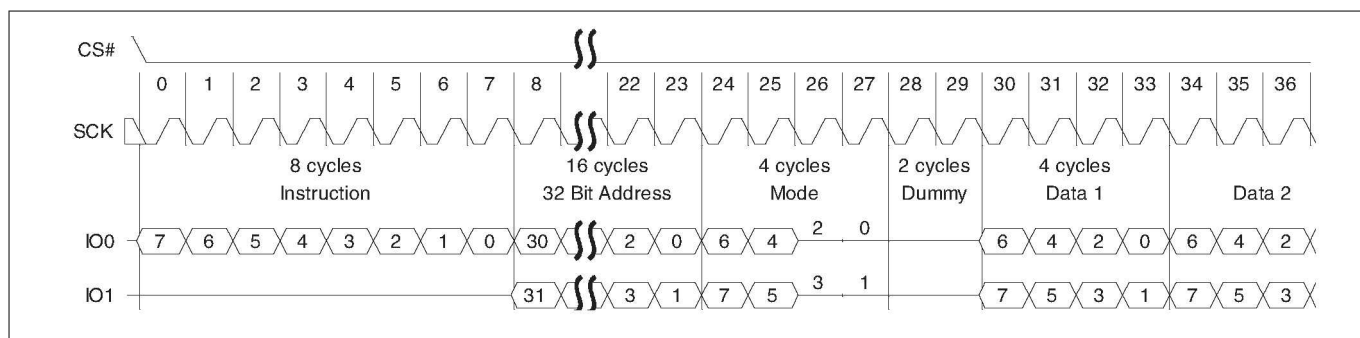
CS# should not be driven high during mode or dummy bits as this may make the mode bits indeterminate.



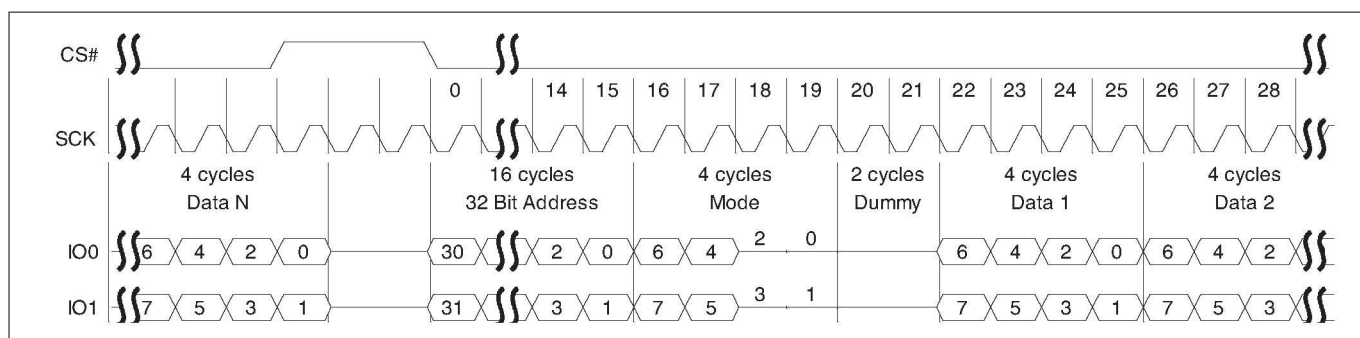
**Figure 77** Dual I/O Read command sequence (3-byte address, BBh [ExtAdd = 0], HPLC = 00b)



**Figure 78** Dual I/O Read command sequence (4-byte address, BBh [ExtAdd = 1], HPLC = 10b)



**Figure 79** Dual I/O Read command sequence (4-byte address, BCh or BBh [ExtAdd = 1], EHPLC = 10b)



**Figure 80** Continuous Dual I/O Read command sequence (4-byte address, BCh or BBh [ExtAdd = 1], EHPLC = 10b)

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

The instruction

## 11 Commands

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

The Quad I/O Read command improves throughput with four I/O signals — IO0-IO3. It is similar to the Quad Output Read command but allows input of the address bits four bits per serial SCK clock. In some applications, the reduced instruction overhead might allow for code execution (XIP) directly from S25FL128S and S25FL256S devices. The QUAD bit of the Configuration Register must be set (CR Bit1=1) to enable the Quad capability of S25FL128S and S25FL256S devices.

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-IO3. This latency period (i.e., dummy cycles) allows the device's internal circuitry enough time to access data at the initial address. During latency cycles, the data value on IO0-IO3 are "don't care" and may be high impedance. The number of dummy cycles is determined by the frequency of SCK and the latency code table (refer to [Table 33](#)). There are different ordering part numbers that select the latency code table used for this command, either the High Performance LC (HPLC) table or the Enhanced High Performance LC (EHPLC) table. The number of dummy cycles is set by the LC bits in the Configuration Register (CR1). However, both latency code tables use the same latency values for the Quad I/O Read command.

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

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000 h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000 h, 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 81](#) or [Figure 83](#)). 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 82](#) or [Figure 84](#); thus, eliminating eight cycles for the command sequence. The following sequence will release the device from Quad I/O High Performance Read mode; after which, the device can accept standard SPI commands:

- 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.

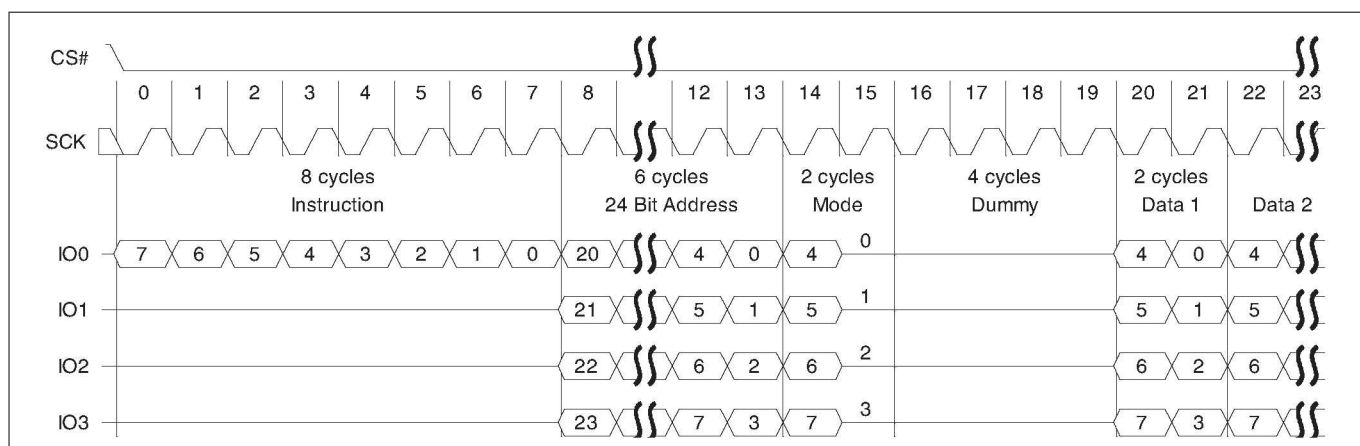
During any operation, if CS# toggles high to low to high for eight cycles (or less) and data input (IO0-IO3) are not set for a valid instruction sequence, then the device will be released from Quad I/O High Performance Read mode. Note that the two mode bit clock cycles and additional wait states (i.e., dummy cycles) allow the device's internal circuitry latency time to access the initial address after the last address cycle that is clocked into IO0-IO3.

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

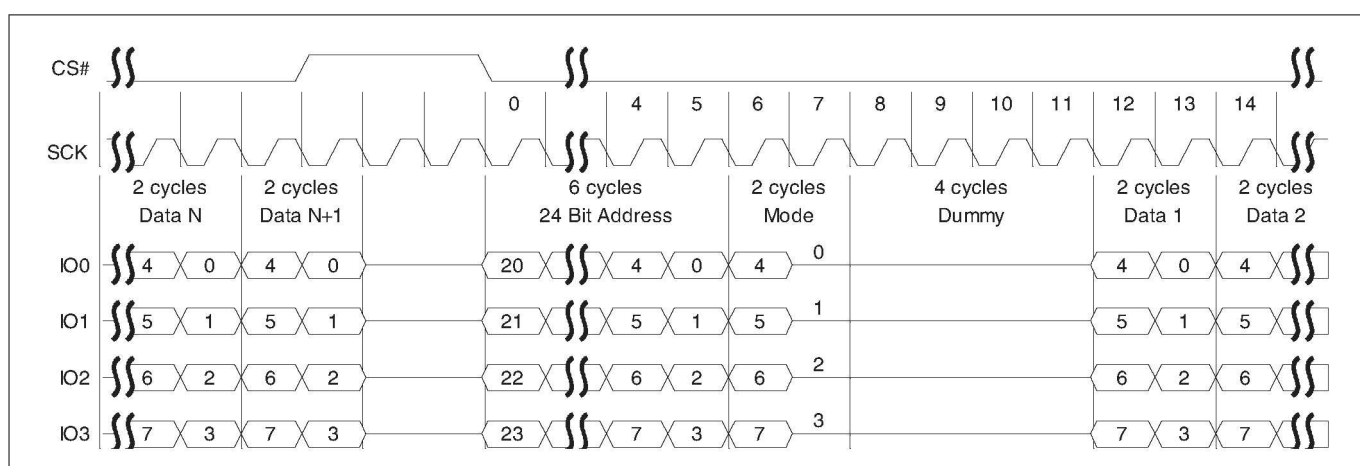
CS# should not be driven high during mode or dummy bits as this may make the mode bits indeterminate.



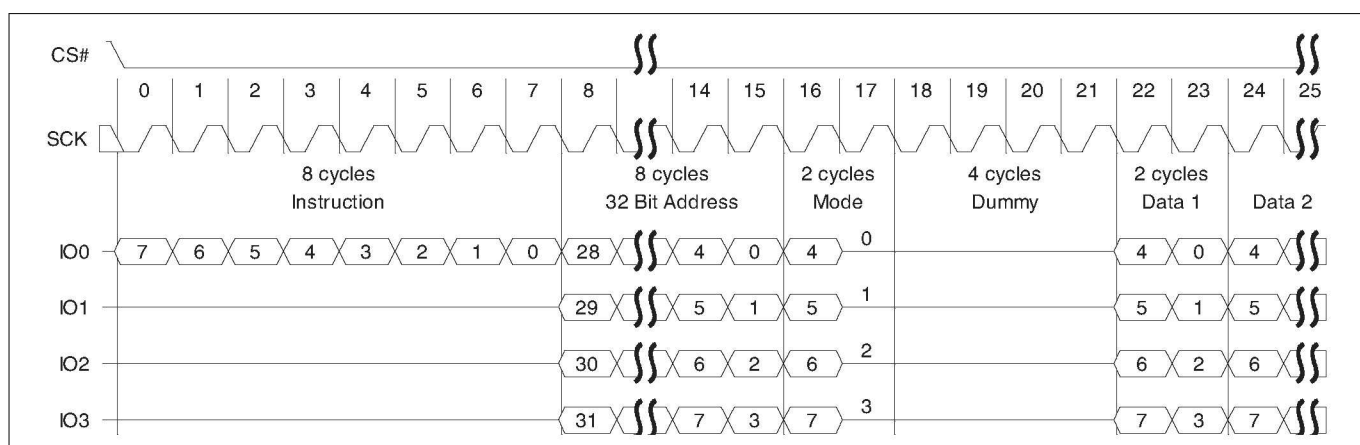
## 11 Commands



**Figure 81** Quad I/O Read command sequence (3-byte address, EBh [ExtAdd = 0], LC = 00b)

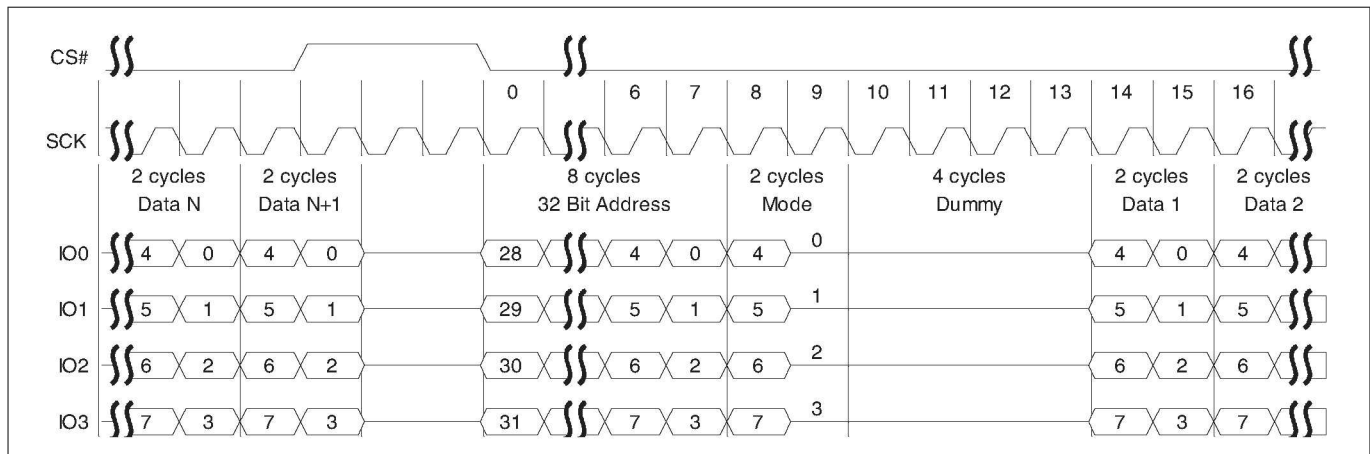


**Figure 82** Continuous Quad I/O Read command sequence (3-byte address), LC = 00b



**Figure 83** Quad I/O Read command sequence (4-byte address, ECh or EBh [ExtAdd=1], LC = 00b)

## 11 Commands



**Figure 84** Continuous Quad I/O Read command sequence (4-byte address), LC = 00b

### 11.4.7 DDR Fast Read (DDRF 0 Dh, 4DDRF 0 Eh)

The instruction

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

The DDR Fast Read command improves throughput by transferring address and data on both the falling and rising edge of SCK. It is similar to the Fast Read command but allows transfer of address and data on every edge of the clock.

The maximum operating clock frequency for DDR Fast Read command is 80 MHz.

For the DDR Fast Read command, there is a latency required after the last address bits are shifted into SI before data begins shifting out of SO. There are different ordering part numbers that select the latency code table used for this command, either the High Performance LC (HPLC) table or the Enhanced High Performance LC (EHPLC) table. The HPLC table does not provide cycles for mode bits so each DDR Fast Read command starts with the 8 bit instruction, followed by address, followed by a latency period.

This latency period (dummy cycles) allows the device internal circuitry enough time to access data at the initial address. During the dummy cycles, the data value on SI is “don't care” and may be high impedance. The number of dummy cycles is determined by the frequency of SCK ([Table 33](#)). The number of dummy cycles is set by the LC bits in the Configuration Register (CR1).

Then the memory contents, at the address given, is shifted out, in DDR fashion, one bit at a time on each clock edge through SO. Each bit is shifted out at the SCK frequency by the rising and 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 000000 h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000 h, allowing the read sequence to be continued indefinitely.

The EHPLC table does provide cycles for mode bits so a series of DDR Fast Read commands may eliminate the 8 bit instruction after the first DDR Fast Read command sends a mode bit pattern of complementary first and second Nibbles, e.g. A5h, 5 Ah, 0 Fh, etc., that indicates the following command will also be a DDR Fast Read command. The first DDR Fast Read command in a series starts with the 8-bit instruction, followed by address, followed by four cycles of mode bits, followed by a latency period. If the mode bit pattern is complementary the next command is assumed to be an additional DDR Fast Read command that does not provide instruction bits. That command starts with address, followed by mode bits, followed by latency.

## 11 Commands

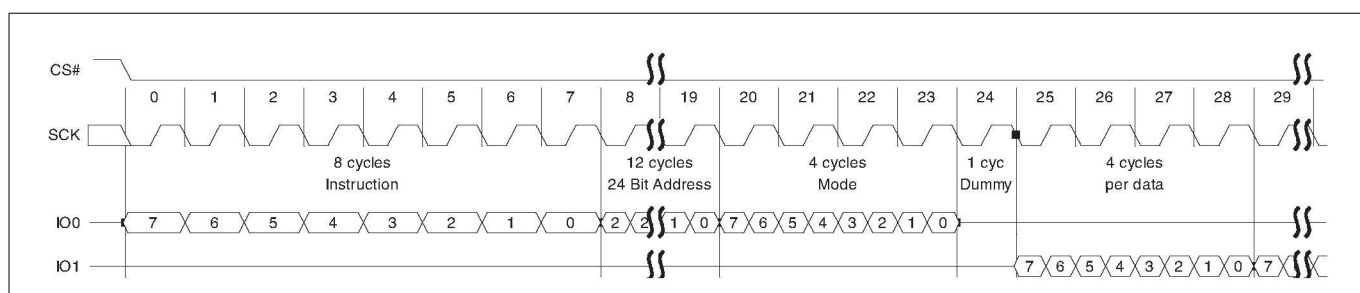
When the EHPLC table is used, address jumps can be done without the need for additional DDR Fast Read instructions. This is controlled through the setting of the Mode bits (after the address sequence, as shown in [Figure 85](#) and [Figure 87](#). This added feature removes the need for the eight bit SDR instruction sequence to reduce initial access time (improves XIP performance). The Mode bits control the length of the next DDR Fast Read operation through the inclusion or exclusion of the first byte instruction code. If the upper nibble (IO[7:4]) and lower nibble (IO[3:0]) of the Mode bits are complementary (i.e., 5 h and Ah) then the next address can be entered (after CS# is raised high and then asserted low) without requiring the 0 Dh or 0 Eh instruction, as shown in [Figure 86](#) and [Figure 88](#), thus, eliminating eight cycles from the command sequence. The following sequences will release the device from this continuous DDR Fast Read mode; after which, the device can accept standard SPI commands:

1. During the DDR Fast Read Command Sequence, if the Mode bits are not complementary the next time CS# is raised high the device will be released from the continuous DDR Fast Read mode.
2. During any operation, if CS# toggles high to low to high for eight cycles (or less) and data input (SI) are not set for a valid instruction sequence, then the device will be released from DDR Fast Read mode.

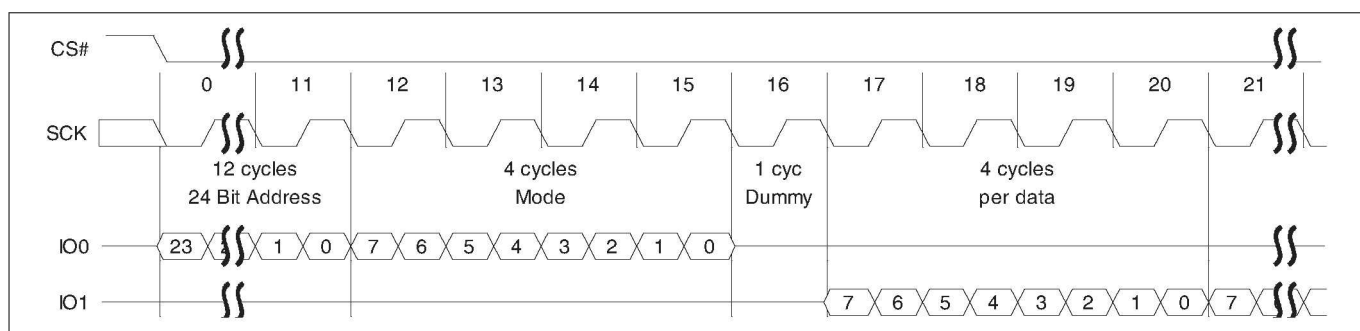
CS# should not be driven high during mode or dummy bits as this may make the mode bits indeterminate.

The HOLD function is not valid during any part of a Fast DDR Command.

Although the data learning pattern (DLP) is programmable, the following example shows example of the DLP of 34 h. The DLP 34 h (or 00110100) will be driven on each of the active outputs (i.e., all four IOs on a x4 device, both IOs on a x2 device and the single SO output on a x1 device). 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](#) for more details.



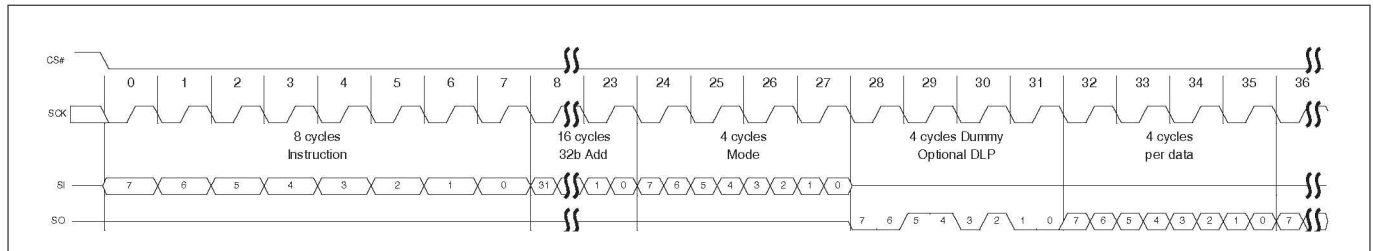
**Figure 85** DDR Fast Read initial access (3-byte address, 0 Dh [ExtAdd = 0, EHPLC = 11b])



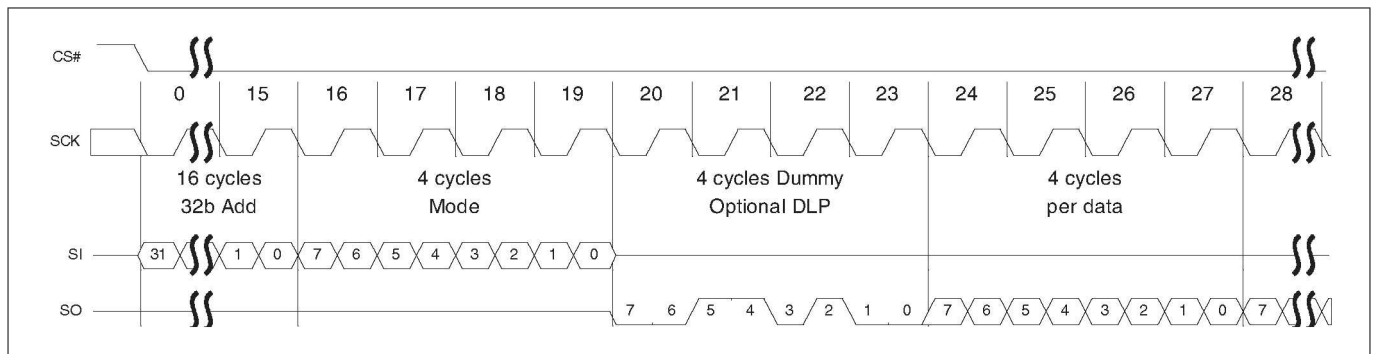
**Figure 86** Continuous DDR Fast Read subsequent access (3-byte address [ExtAdd = 0, EHPLC = 11b])



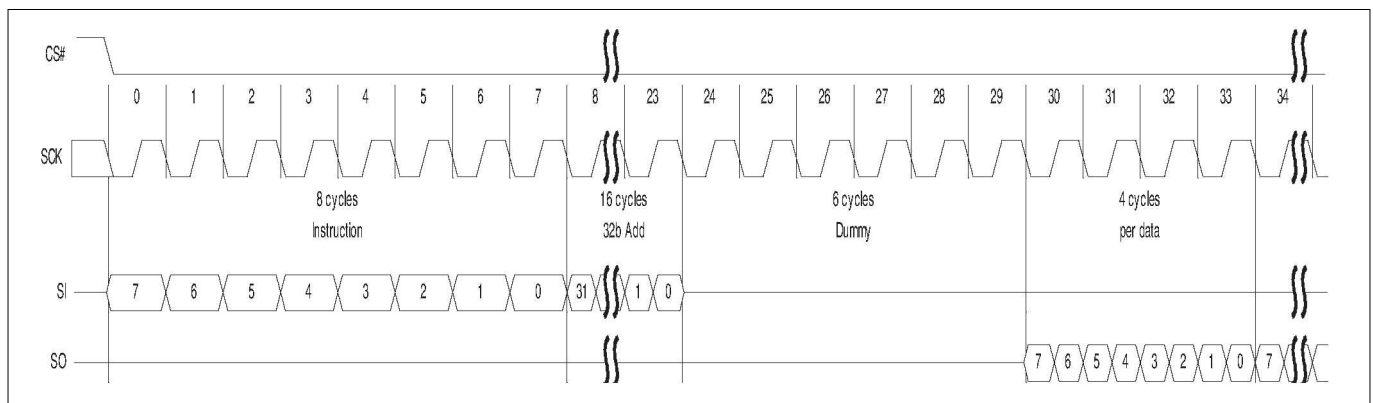
## 11 Commands



**Figure 87** DDR Fast Read initial access (4-byte address, 0 Eh or 0 Dh [ExtAdd = 1], EHPLC = 01b)<sup>3)</sup>



**Figure 88** Continuous DDR Fast Read subsequent access (4-byte address [ExtAdd = 1], EHPLC = 01b)<sup>3)</sup>



**Figure 89** DDR Fast Read subsequent access (4-byte address, HPLC = 01b)

### 11.4.8 DDR Dual I/O Read (BDh, BEh)

The instruction

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

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

The DDR Dual I/O Read command improves throughput with two I/O signals — IO0 (SI) and IO1 (SO). It is similar to the Dual I/O Read command but transfers two address, mode, or data bits on every edge of the clock. In some applications, the reduced instruction overhead might allow for code execution (XIP) directly from S25FL128S and S25FL256S devices.

<sup>3)</sup> Example DLP of 34 h (or 00110100).

## 11 Commands

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

For DDR Dual I/O Read commands, there is a latency required after the last address bits are shifted into IO0 and IO1, before data begins shifting out of IO0 and IO1. There are different ordering part numbers that select the latency code table used for this command, either the High Performance LC (HPLC) table or the Enhanced High Performance LC (EHPLC) table. The number of latency (dummy) clocks is determined by the frequency of SCK (see [Table 32](#) or [Table 34](#)). The number of dummy cycles is set by the LC bits in the Configuration Register (CR1).

The HPLC table does not provide cycles for mode bits so each Dual I/O command starts with the 8 bit instruction, followed by address, followed by a latency period. This latency period allows the device's internal circuitry enough time to access the initial address. During these latency cycles, the data value on SI (IO0) and SO (IO1) are "don't care" and may be high impedance. When the Data Learning Pattern (DLP) is enabled the host system must not drive the IO signals during the dummy cycles. The IO signals must be left high impedance by the host so that the memory device can drive the DLP during the dummy cycles.

The EHPLC table does provide cycles for mode bits so a series of Dual I/O DDR commands may eliminate the 8 bit instruction after the first command sends a complementary mode bit pattern, as shown in [Figure 90](#) and [Figure 92](#). This added feature removes the need for the eight bit SDR instruction sequence and dramatically reduces initial access times (improves XIP performance). The Mode bits control the length of the next DDR Dual I/O Read operation through the inclusion or exclusion of the first byte instruction code. If the upper nibble (IO[7:4]) and lower nibble (IO[3:0]) of the Mode bits are complementary (i.e., 5 h and Ah) the device transitions to Continuous DDR Dual I/O Read Mode and the next address can be entered (after CS# is raised high and then asserted low) without requiring the BDh or BEh instruction, as shown in [Figure 91](#), and thus, eliminating eight cycles from the command sequence. The following sequences will release the device from Continuous DDR Dual I/O Read mode; after which, the device can accept standard SPI commands:

1. During the DDR Dual I/O Read Command Sequence, if the Mode bits are not complementary the next time CS# is raised high and then asserted low the device will be released from DDR Dual I/O Read mode.
2. During any operation, if CS# toggles high to low to high for eight cycles (or less) and data input (IO0 and IO1) are not set for a valid instruction sequence, then the device will be released from DDR Dual I/O Read 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 000000 h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000 h, 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 Dual I/O DDR commands.

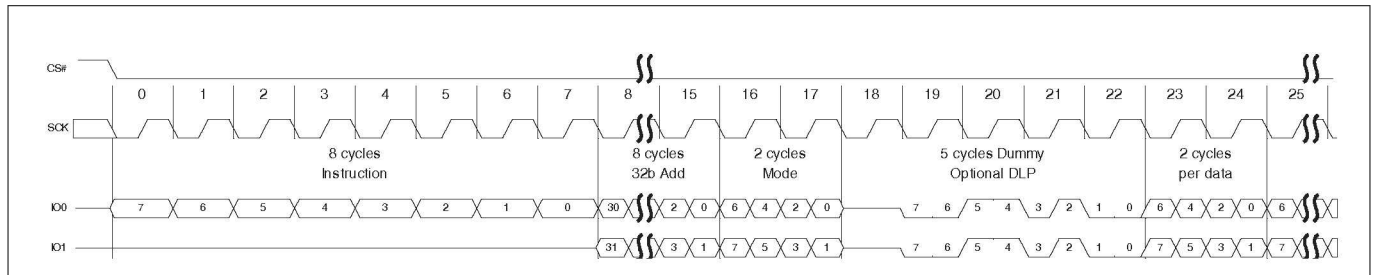
Note that the memory devices may drive the IOs with a preamble prior to the first data value. The preamble is a data learning pattern (DLP) that is used by the host controller to optimize data capture at higher frequencies. The preamble DLP drives the IO bus for the four clock cycles immediately before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

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

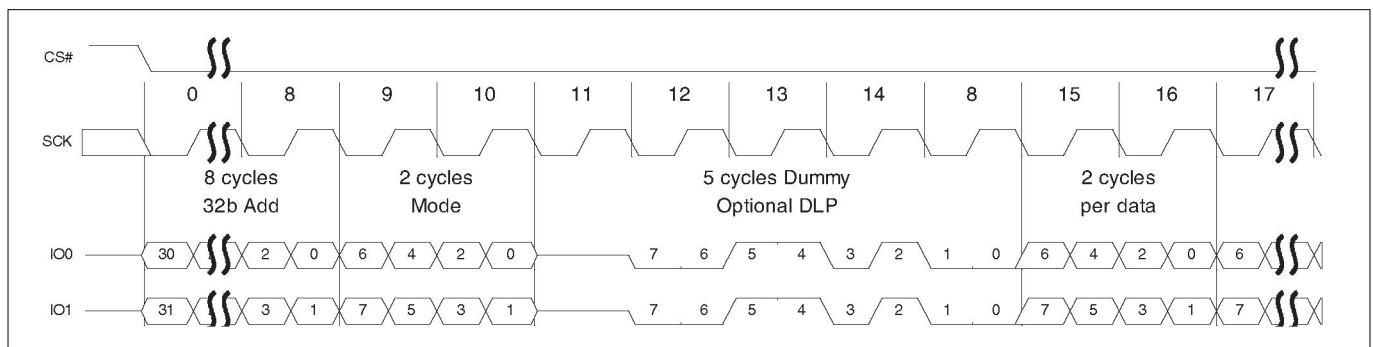
Although the data learning pattern (DLP) is programmable, the following example shows example of the DLP of 34 h. The DLP 34 h (or 00110100) will be driven on each of the active outputs (i.e., all four SIOs on a x4 device, both SIOs on a x2 device and the single SO output on a x1 device). This pattern 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).

## 11 Commands

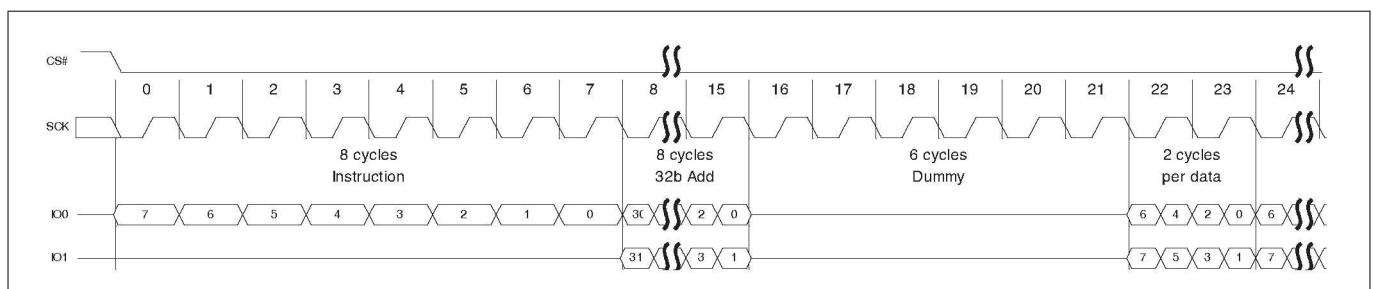
The two AC transition scenarios include data low for a short period of time (one half clock) followed by a high going transition (101) and the complementary low going transition (010). The DC transitions will typically occur with a starting point closer to the supply rail than the AC transitions that 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](#) for more details.



**Figure 90** DDR Dual I/O Read initial access (4-byte address, BEh or BDh [ExtAdd = 1], EHPLC = 01b)



**Figure 91** Continuous DDR Dual I/O Read subsequent access (4-byte address, EHPLC = 01b)



**Figure 92** DDR Dual I/O Read (4-byte address, BEh or BDh [ExtAdd = 1], HPLC = 00b)

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

The Read DDR Quad I/O command improves throughput with four I/O signals - IO0-IO3. It is similar to the Quad I/O Read command but allows input of the address four bits on every edge of the clock. In some applications, the reduced instruction overhead might allow for code execution (XIP) directly from S25FL128S and S25FL256S devices. The QUAD bit of the Configuration Register must be set (CR Bit1 = 1) to enable the Quad capability.

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)

## 11 Commands

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

The maximum operating clock frequency for 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-IO3 signals before data begins shifting out of IO0-IO3. This latency period (dummy cycles) allows the device's internal circuitry enough time to access the initial address. During these latency cycles, the data value on IO0-IO3 are "don't care" and may be high impedance. When the Data Learning Pattern (DLP) is enabled the host system must not drive the IO signals during the dummy cycles. The IO signals must be left high impedance by the host so that the memory device can drive the DLP during the dummy cycles.

There are different ordering part numbers that select the latency code table used for this command, either the High Performance LC (HPLC) table or the Enhanced High Performance LC (EHPLC) table. The number of dummy cycles is determined by the frequency of SCK (see [Table 32](#)). 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 93](#) and [Figure 95](#). 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 94](#) and [Figure 96](#) 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, IO1, IO2, and IO3) 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 000000 h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000 h, 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.

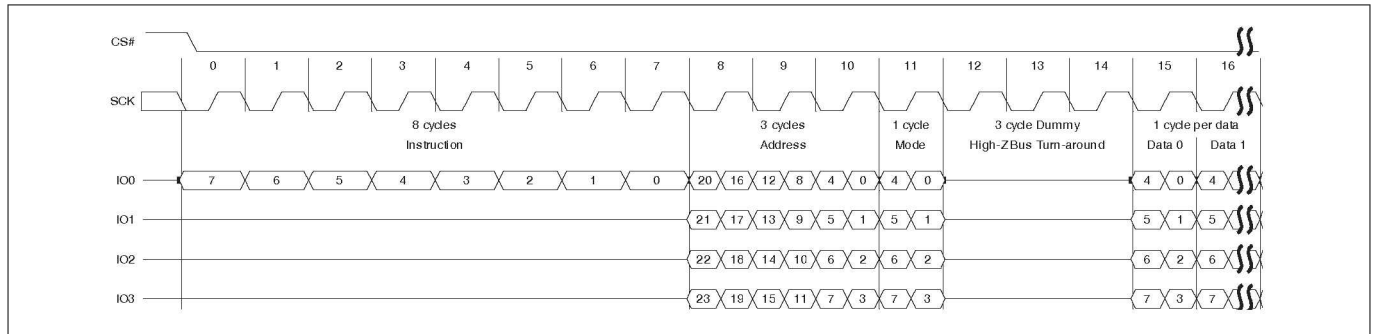
Note that the memory devices drive the IOs with a preamble prior to the first data value. The preamble is a pattern that is used by the host controller to optimize data capture at higher frequencies. The preamble drives the IO bus for the four clock cycles immediately before data is output. The host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

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.

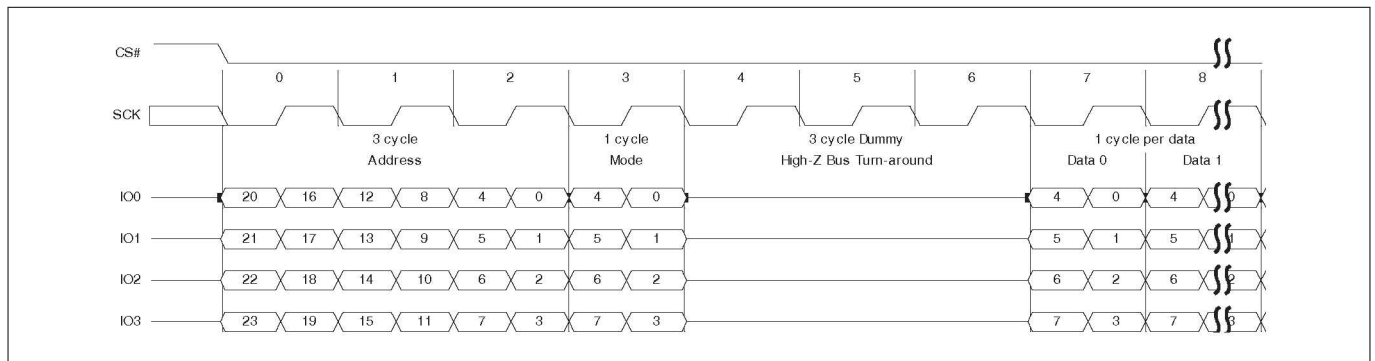
Although the data learning pattern (DLP) is programmable, the following example shows example of the DLP of 34 h. The DLP 34 h (or 00110100) will be driven on each of the active outputs (i.e., all four SIOs on a x4 device, both SIOs on a x2 device and the single SO output on a x1 device). This pattern 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

## 11 Commands

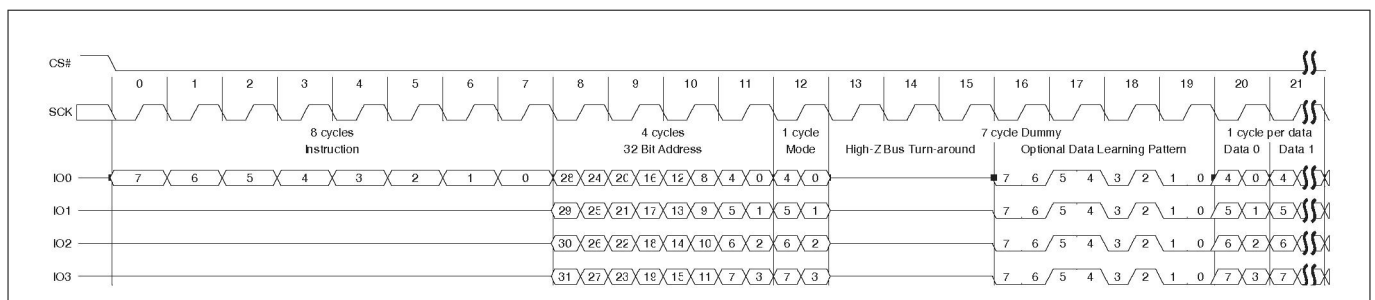
(two half clocks) followed by a high going transition (001) and the complementary low going transition (110). The two AC transition scenarios include data low for a short period of time (one half clock) followed by a high going transition (101) and the complementary low going transition (010). The DC 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](#) for more details.



**Figure 93** DDR Quad I/O Read initial access (3-byte address, EDh [ExtAdd = 0], HPLC = 11b)



**Figure 94** Continuous DDR Quad I/O Read subsequent access (3-byte address, HPLC = 11b)

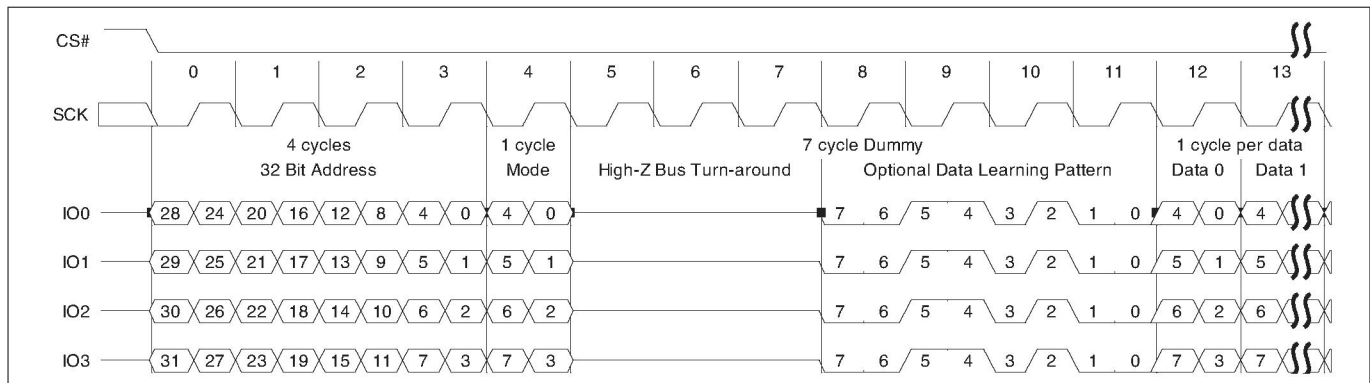


**Figure 95** DDR Quad I/O Read initial access (4-byte address, EEh or EDh [ExtAdd = 1], EHPLC = 01b)<sup>4)</sup>

<sup>4</sup> Example DLP of 34 h (or 00110100).



## 11 Commands



**Figure 96** Continuous DDR Quad I/O Read subsequent access (4-byte address, EHPLC = 01b)<sup>4)</sup>

## 11.5 Program flash array commands

### 11.5.1 Program granularity

#### 11.5.1.1 Automatic ECC

Each 16 byte aligned and 16 byte length Programming Block has an automatic Error Correction Code (ECC) value. The data block plus ECC form an ECC unit. In combination with Error Detection and Correction (EDC) logic the ECC is used to detect and correct any single bit error found during a read access. When data is first programmed within an ECC unit the ECC value is set for the entire ECC unit. If the same ECC unit is programmed more than once the ECC value is changed to disable the 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 \(ECCSR\)](#).) 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.

<sup>4</sup> Example DLP of 34 h (or 00110100).

## 11 Commands

- To re-enable ECC for an ECC unit that has been disabled, the Sector that includes the ECC unit must be erased.
- To ensure the best data integrity provided by EDC, each ECC unit should be programmed only once so that ECC is stored for that unit and not disabled.
- The calculation, programming, and disabling of ECC is done automatically as part of a programming operation. The detection and correction, if needed, is done automatically as part of read operations. The host system sees only corrected data from a read operation.
- ECC protects the OTP region - however a second program operation on the same ECC unit will disable ECC permanently on that ECC unit (OTP is one time programmable, hence an erase operation to re-enable the ECC enable/indicator bit is prohibited).

### 11.5.1.2 Page programming

Page Programming is done by loading a Page Buffer with data to be programmed and issuing a programming command to move data from the buffer to the memory array. This sets an upper limit on the amount of data that can be programmed with a single programming command. Page Programming allows up to a page size (either 256 or 512 bytes) to be programmed in one operation. The page size is determined by the Ordering Part Number (OPN). 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 512 bytes aligned on 512-byte boundaries with each Page being programmed only once.

### 11.5.1.3 Single byte programming

Single Byte Programming allows full backward compatibility to the 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

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

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

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

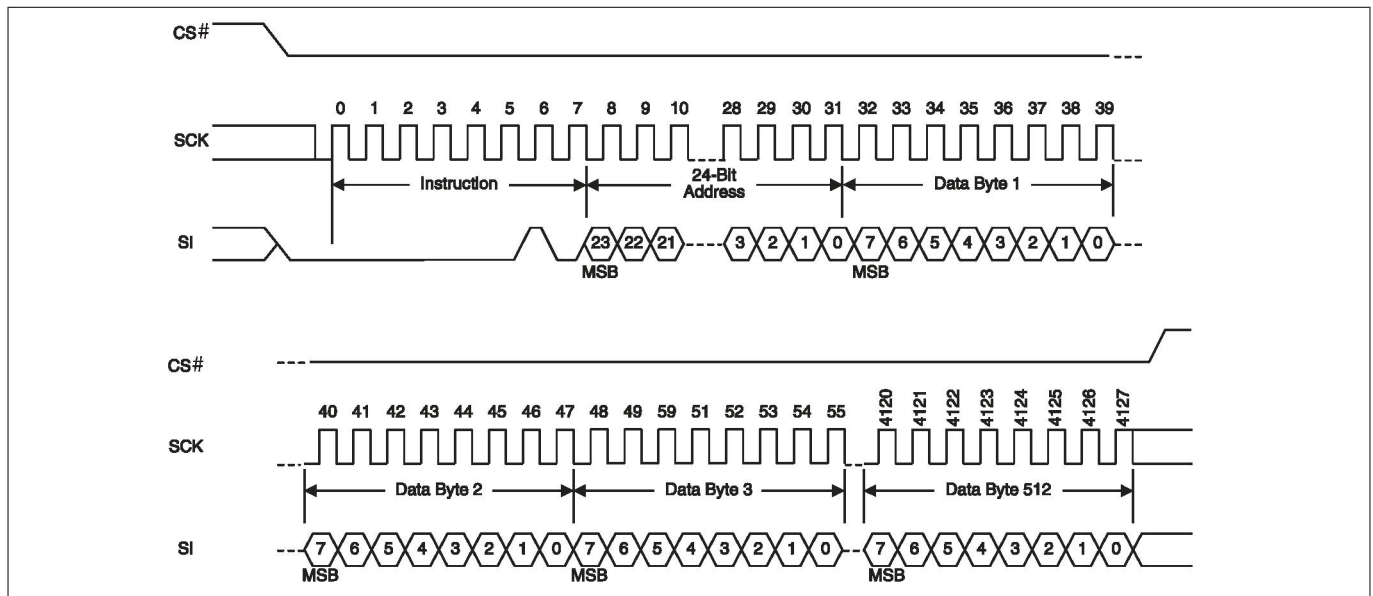
and at least one data byte on SI. Depending on the device OPN, the page size can either be 256 or 512 bytes. Up to a page can be provided on SI after the 3-byte address with instruction 02 h or 4-byte address with instruction 12 h has been provided. If the 9 least significant address bits (A8-A0) are 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.

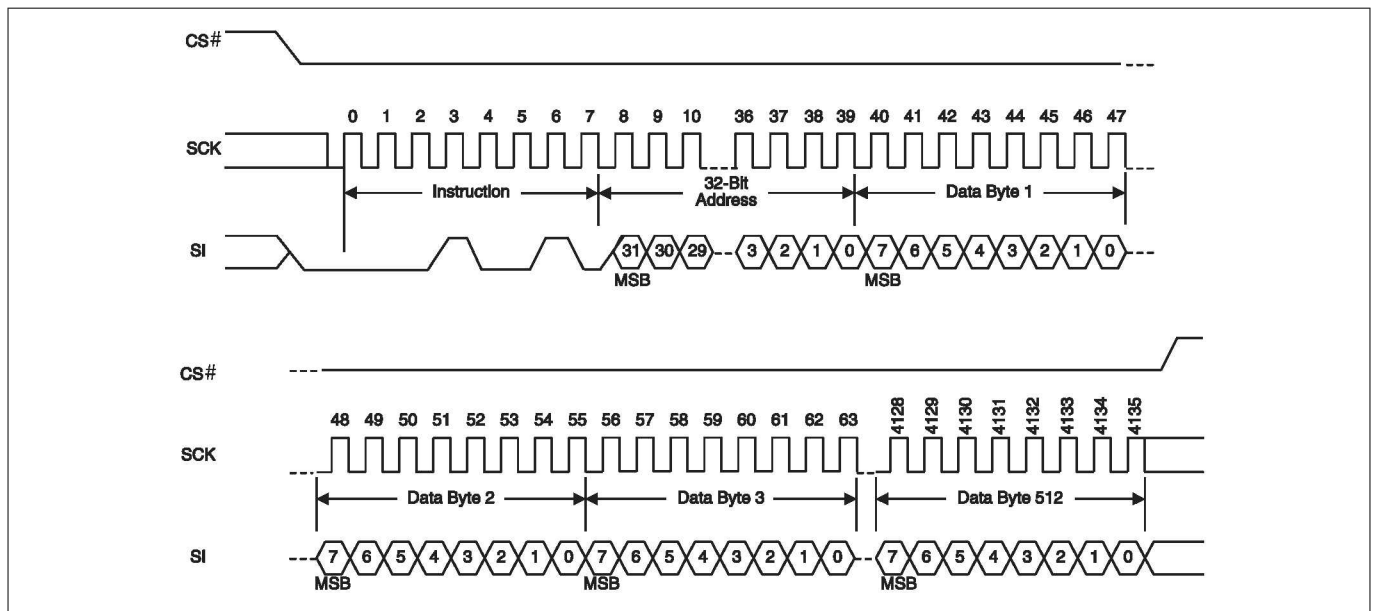
For optimized timings, using the Page Program (PP) command to load the entire page size program buffer within the page boundary will save overall programming time versus loading less than a page size into the program buffer.

## 11 Commands

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 97** Page Program (PP) command sequence (3-byte address, 02 h)



**Figure 98** Page Program (4PP) command sequence (4-byte address, 12 h)

### 11.5.3 Quad Page Program (QPP 32 h or 38 h, or 4QPP 34 h)

The Quad-input Page Program (QPP) command allows bytes to be programmed in the memory (changing bits from 1 to 0). The Quad-input Page Program (QPP) command allows up to a page size (either 256 or 512 bytes) of data to be loaded into the Page Buffer using four signals: IO0-IO3. QPP can improve performance for PROM Programmer and applications that have slower clock speeds (< 12 MHz) by loading 4 bits of data per clock cycle. Systems with faster clock speeds do not realize as much benefit for the QPP command since the inherent



## 11 Commands

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).

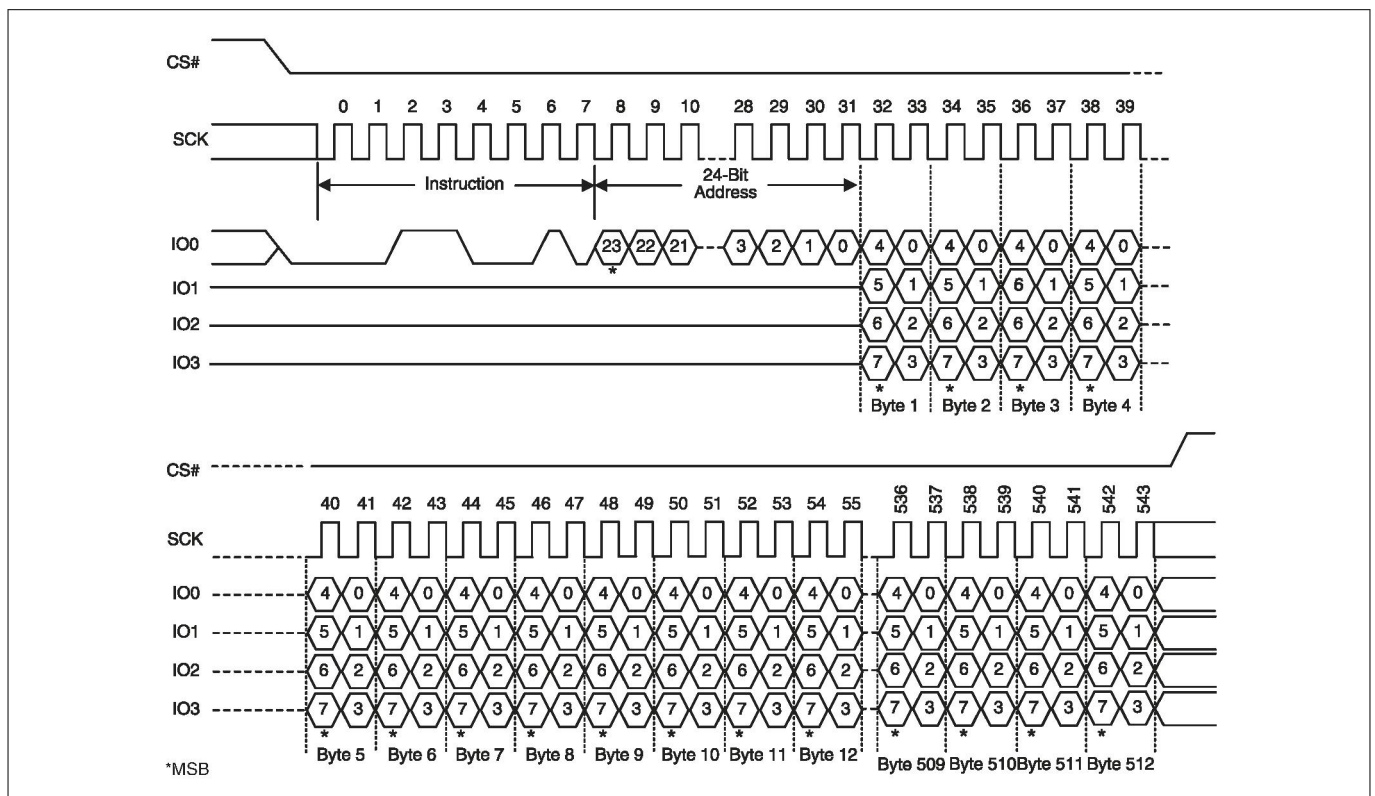
The instruction

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

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

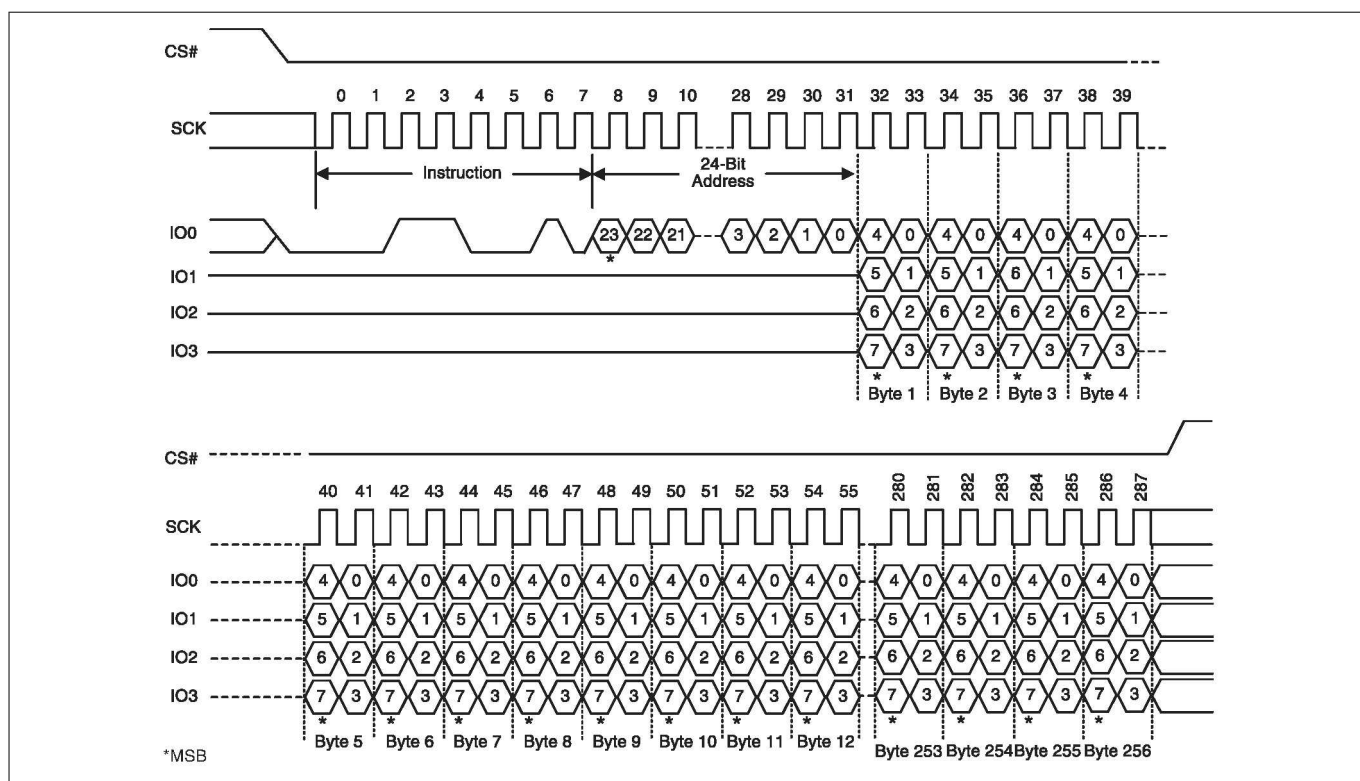
QPP requires programming to be done one full page at a time. While less than a full page of data may be loaded for programming, the entire page is considered programmed, any locations not filled with data will be left as ones, the same page must not be programmed more than once.

All other functions of QPP are identical to Page Program. The QPP command sequence is shown in Figure 99.

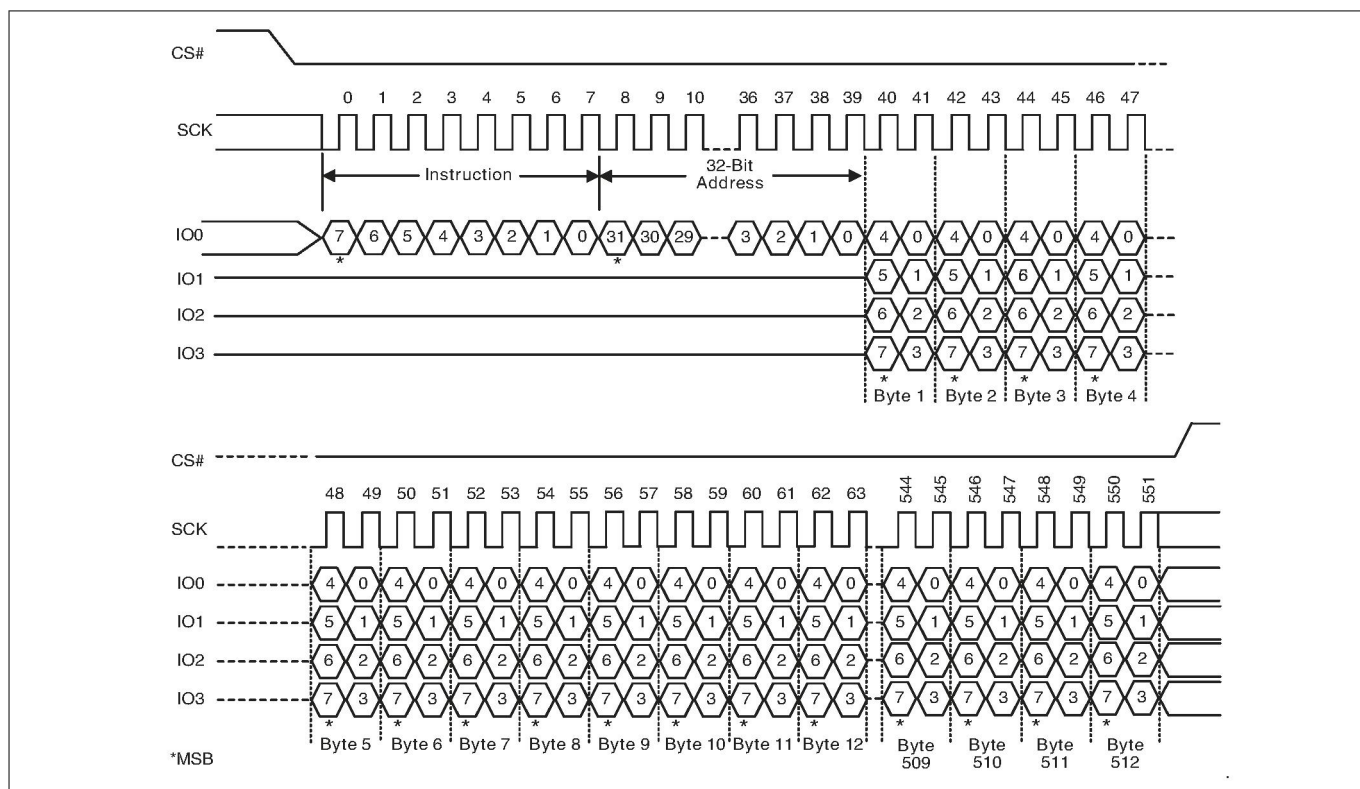


**Figure 99 Quad 512-Byte Page Program command sequence (3-byte address, 32 h or 38 h)**

## 11 Commands

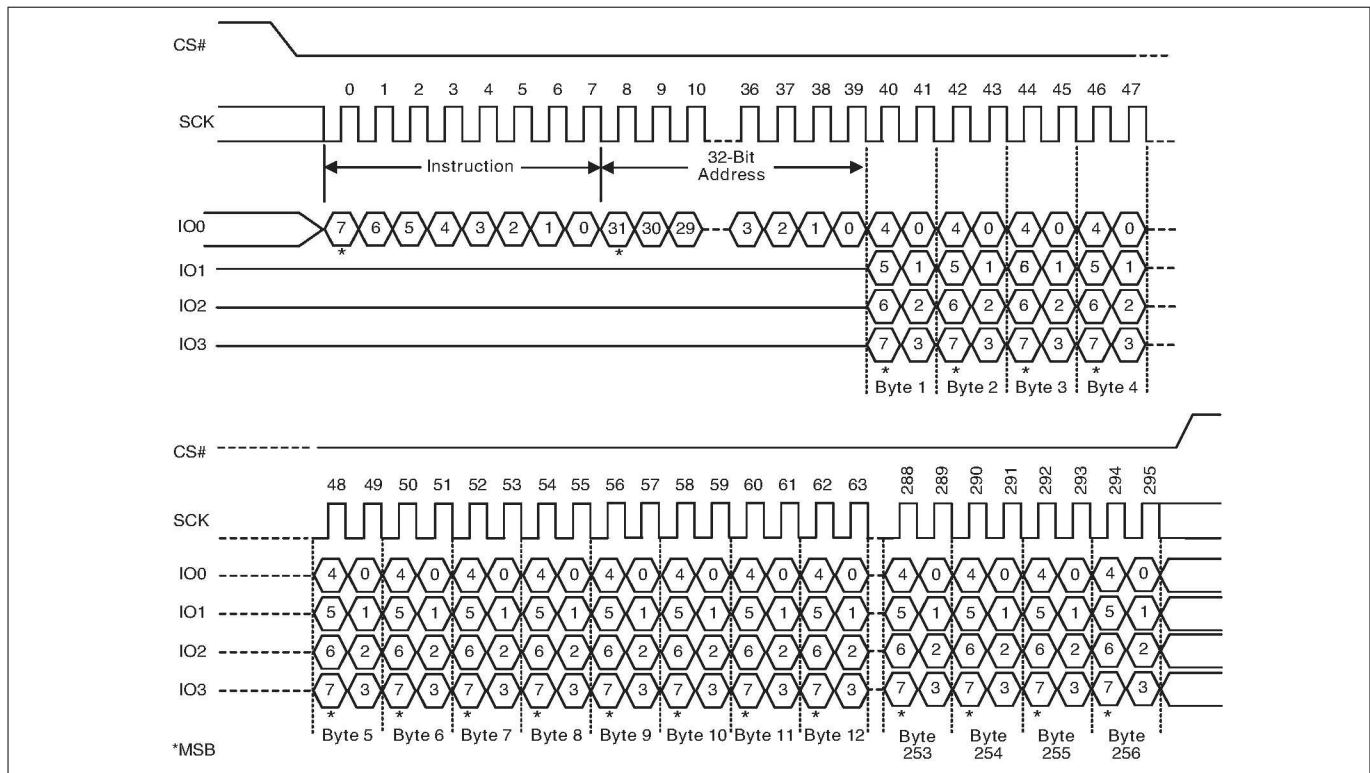


**Figure 100** Quad 256-byte Page Program command sequence (3-byte address, 32 h or 38 h)



**Figure 101** Quad 512-byte Page Program command sequence (4-byte address, 34 h or 32 h or 38 h [ExtAdd = 1])

## 11 Commands



**Figure 102** Quad 256-byte Page Program command sequence (4-byte address, 34 h or 32 h or 38 h [ExtAdd = 1])

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

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:

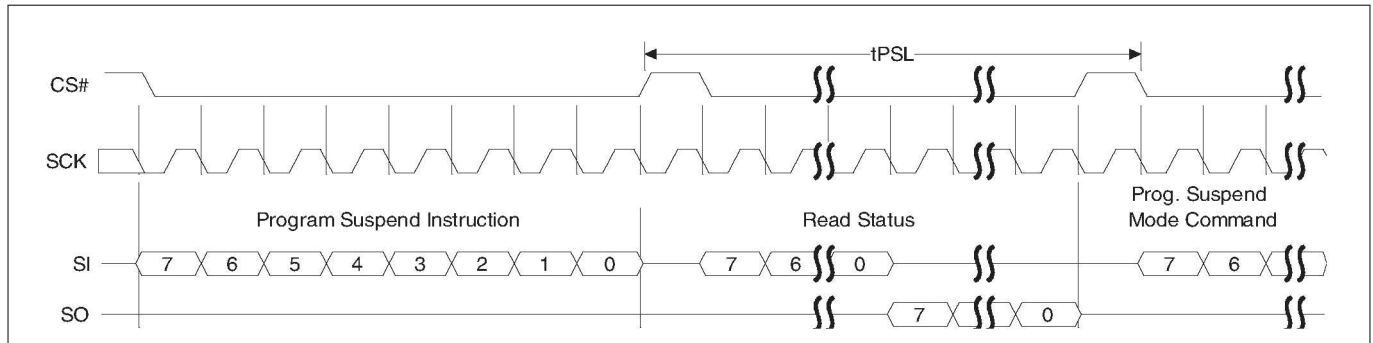
- Read Status Register 1 (RDSR1 05 h)
- Read Status Register 2 (RDSR2 07 h)

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 47. See Table 57 for the commands allowed while programming is suspend.

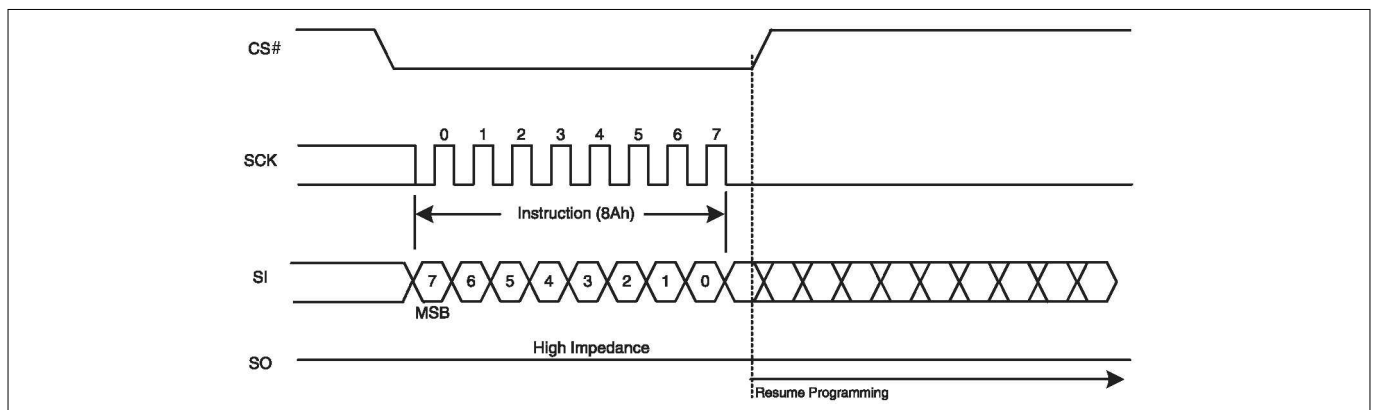
The Program Resume command 8 Ah must be written to resume the programming operation after a Program Suspend. If the programming operation was completed during the suspend operation, a resume command is not needed and has no effect if issued. Program Resume commands will be ignored unless a Program operation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

After a Program Resume command is issued, the WIP bit in the Status Register-1 will be set to a 1 and the programming operation will resume. Program operations may be interrupted as often as necessary e.g., a program suspend command could immediately follow a program resume command but, in order for a program operation to progress to completion there must be some periods of time between resume and the next suspend command greater than or equal to  $t_{PRS}$ . See Table 47.

## 11 Commands



**Figure 103** Program Suspend command sequence



**Figure 104** Program Resume command sequence

## 11.6 Erase flash array commands

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

The P4E command is implemented only in FL128S and FL256S. The P4E command is ignored when the device is configured with the 256-KB sector option.

The Parameter 4-KB Sector Erase (P4E) command sets all the bits of a 4-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

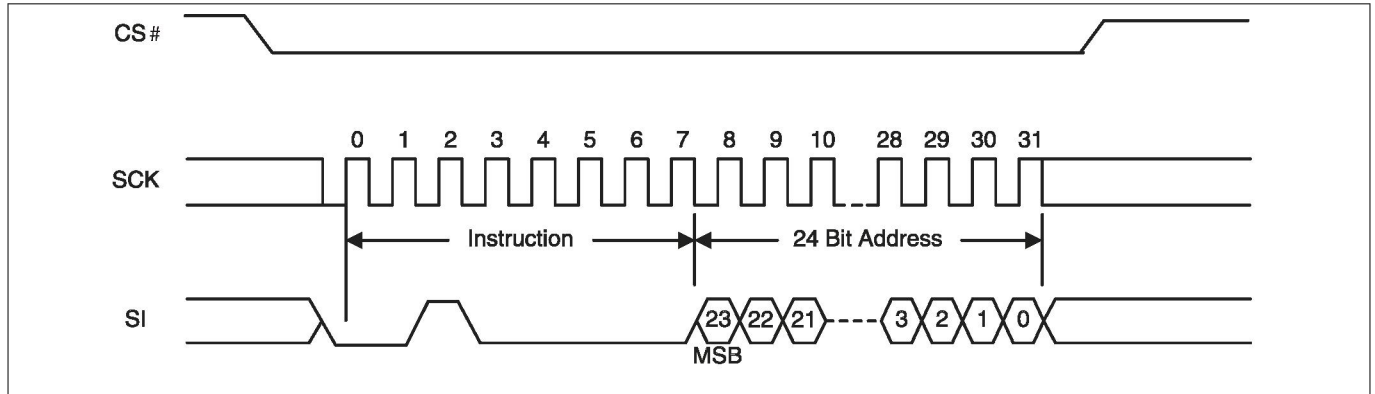
- 20 h [ExtAdd=0] is followed by a 3-byte address (A23-A0), or
- 20 h [ExtAdd=1] is followed by a 4-byte address (A31-A0), or
- 21 h is followed by a 4-byte address (A31-A0)

CS# must be driven into the logic high state after the twenty-fourth or thirty-second bit of the address has been latched in on SI. This will initiate the beginning of internal erase cycle, which involves the pre-programming and erase of the chosen sector of the flash memory array. If CS# is not driven high after the last bit of address, the sector erase operation will not be executed.

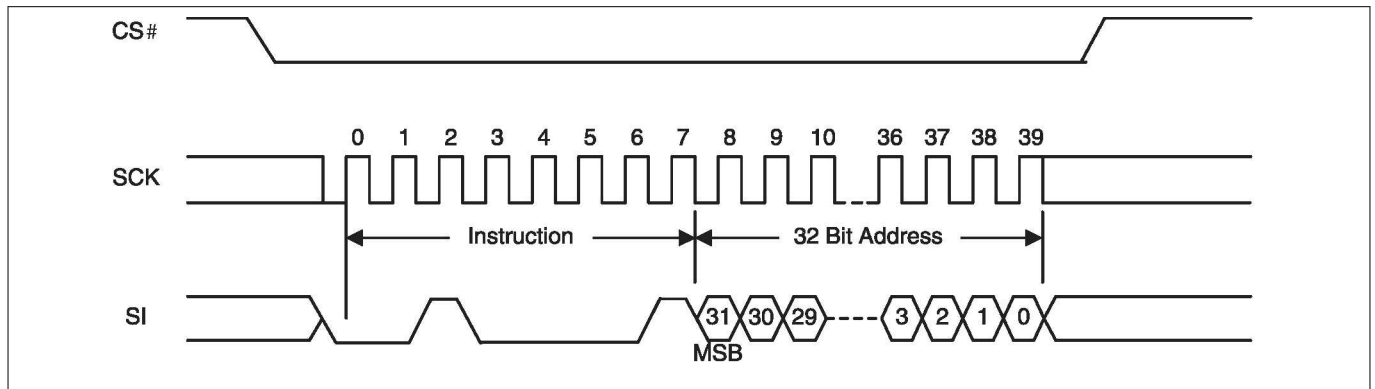
As soon as CS# is driven high, the internal erase cycle will be initiated. With the internal erase cycle in progress, the user can read the value of the Write-In Progress (WIP) bit to determine when the operation has been completed. The WIP bit will indicate a 1, when the erase cycle is in progress and a 0 when the erase cycle has been completed.

## 11 Commands

A P4E command applied to a sector that has been write protected through the Block Protection bits or ASP, will not be executed and will set the E\_ERR status. A P4E command applied to a sector that is larger than 4 KB will not be executed and will not set the E\_ERR status.



**Figure 105** Parameter Sector Erase command sequence (3-byte address, 20 h)



**Figure 106** Parameter Sector Erase command sequence (ExtAdd = 1, 20 h or 4-byte address, 21 h)

### 11.6.2 Sector Erase (SE D8h or 4SE DCh)

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

The instruction

- D8h [ExtAdd=0] is followed by a 3-byte address (A23-A0), or
- D8h [ExtAdd=1] is followed by a 4-byte address (A31-A0), or
- DCh is followed by a 4-byte address (A31-A0)

CS# must be driven into the logic high state after the twenty-fourth or thirty-second bit of address has been latched in on SI. This will initiate the erase cycle, which involves the pre-programming and erase of the chosen sector. If CS# is not driven high after the last bit of address, the sector erase operation will not be executed.

As soon as CS# is driven into the logic high state, the internal erase cycle will be initiated. With the internal erase cycle in progress, the user can read the value of the Write-In Progress (WIP) bit to check if the operation has been completed. The WIP bit will indicate a 1 when the erase cycle is in progress and a 0 when the erase cycle has been completed.

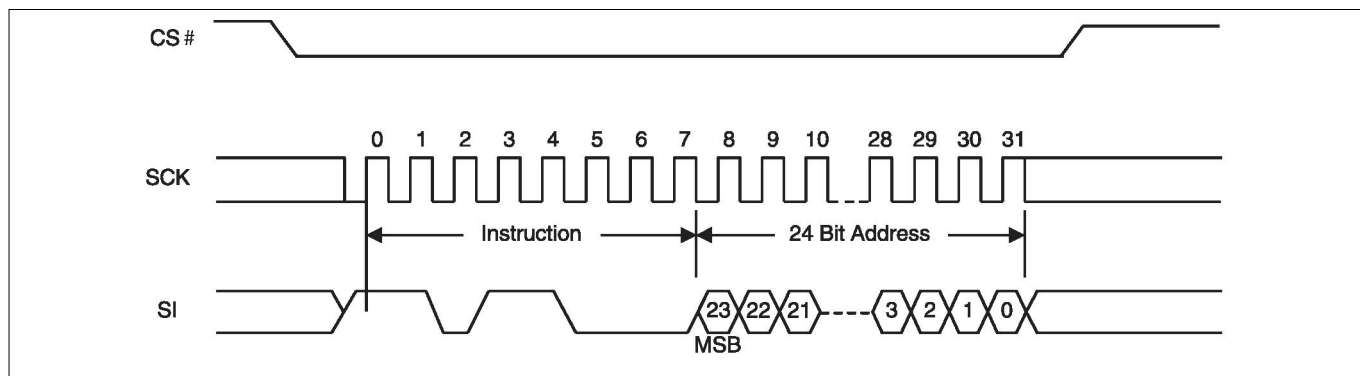
A Sector Erase (SE) command applied to a sector that has been Write Protected through the Block Protection bits or ASP, will not be executed and will set the E\_ERR status.



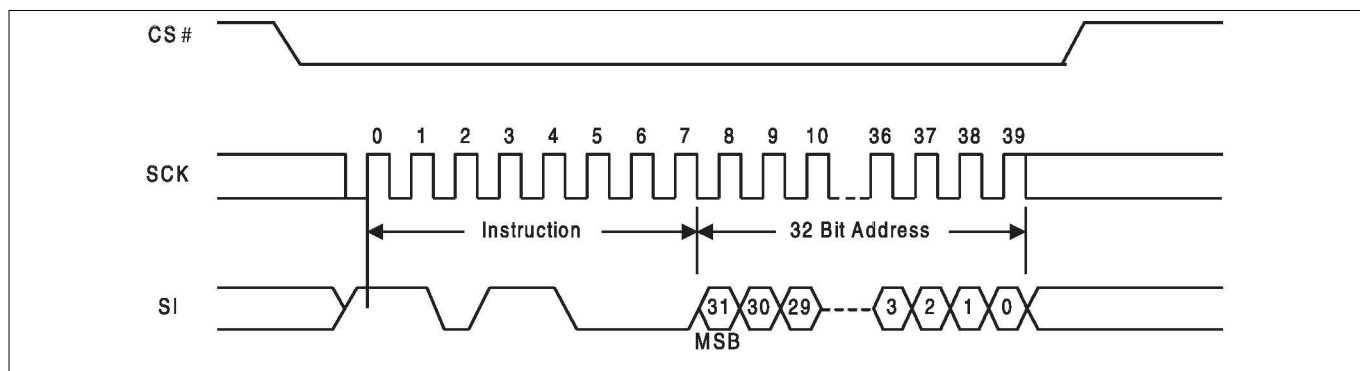
## 11 Commands

A device ordering option determines whether the SE command erases 64 KB or 256 KB. The option to use this command to always erase 256 KB provides for software compatibility with higher density and future S25FL family devices.

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



**Figure 107** Sector Erase command sequence (ExtAdd = 0, 3-byte address, D8h)



**Figure 108** Sector Erase command sequence (ExtAdd = 1, D8h or 4-byte address, DCh)

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

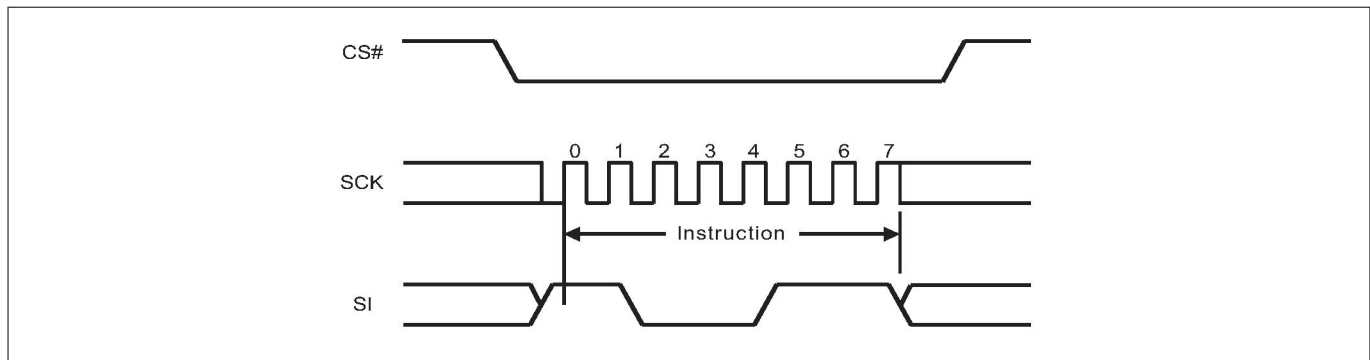
The Bulk Erase (BE) command sets all bits to 1 (all bytes are FFh) inside the entire flash memory array. Before the BE command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the Write Enable Latch (WEL) in the Status Register to enable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

CS# must be driven into the logic high state after the eighth bit of the instruction byte has been latched in on SI. This will initiate the erase cycle, which involves the pre-programming and erase of the entire flash memory array. If CS# is not driven high after the last bit of instruction, the BE operation will not be executed.

As soon as CS# is driven into the logic high state, the erase cycle will be initiated. With the erase cycle in progress, the user can read the value of the Write-In Progress (WIP) bit to determine when the operation has been completed. The WIP bit will indicate a 1 when the erase cycle is in progress and a 0 when the erase cycle has been completed.

A BE command can be executed only when the Block Protection (BP2, BP1, BP0) bits are set to 0's. If the BP bits are not 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.

## 11 Commands



**Figure 109 Bulk Erase command sequence**

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

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 48](#).

Commands allowed after the Erase Suspend command is issued:

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

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 57](#) 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.

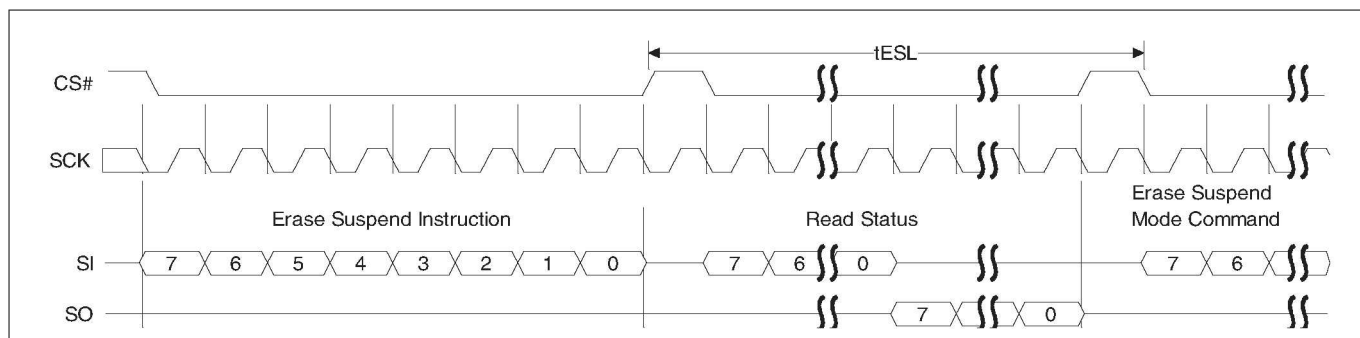
After an erase-suspended program operation is complete, the device returns to the erase-suspend mode. The system can determine the status of the program operation by reading the WIP bit in the Status Register, just as in the standard program operation.

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

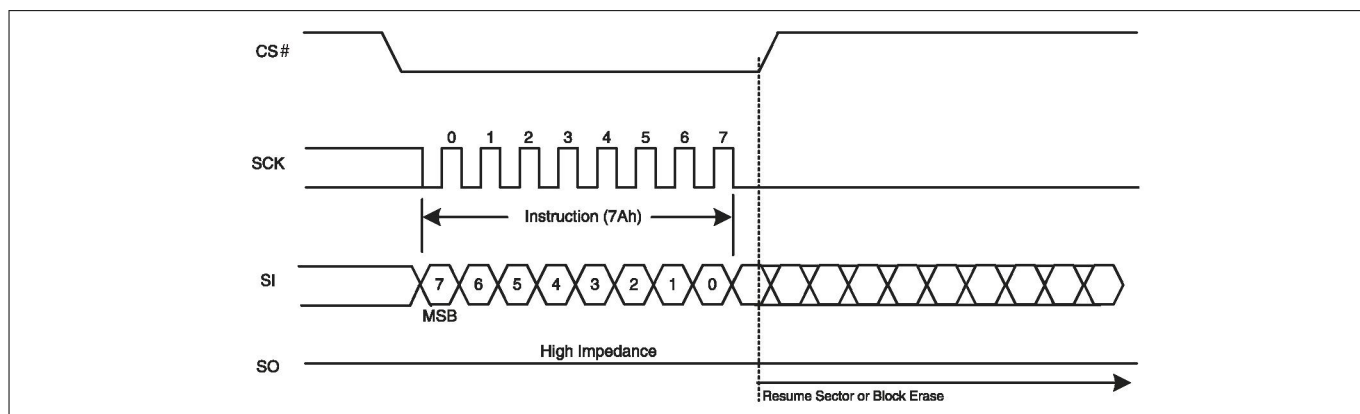
After an Erase Resume command is sent, the WIP bit in the status register will be set to a 1 and the erase operation will continue. Further Resume commands are ignored.

## 11 Commands

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 48](#).



**Figure 110 Erase Suspend command sequence**



**Figure 111 Erase Resume command sequence**

**Table 57 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.

(table continues...)



## 11 Commands

**Table 57 (continued) Commands allowed during Program or Erase Suspend**

Instruction name	Instruction code (Hex)	Allowed during Erase Suspend	Allowed during Program Suspend	Comment
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.
DDRFR	0D	X	X	All array reads allowed in suspend.
4DDRFR	0E	X	X	All array reads allowed in 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.

**(table continues...)**

## 11 Commands

**Table 57 (continued) Commands allowed during Program or Erase Suspend**

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	–
DIOR	BB	X	X	All array reads allowed in suspend.
4DIOR	BC	X	X	All array reads allowed in suspend.
DOR	3B	X	X	All array reads allowed in suspend.
4DOR	3C	X	X	All array reads allowed in suspend.

**(table continues...)**

## 11 Commands

**Table 57 (continued) Commands allowed during Program or Erase Suspend**

Instruction name	Instruction code (Hex)	Allowed during Erase Suspend	Allowed during Program Suspend	Comment
DDRDIOR	BD	X	X	All array reads allowed in suspend.
4DDRDIOR	BE	X	X	All array reads allowed in suspend.
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.

## 11 Commands

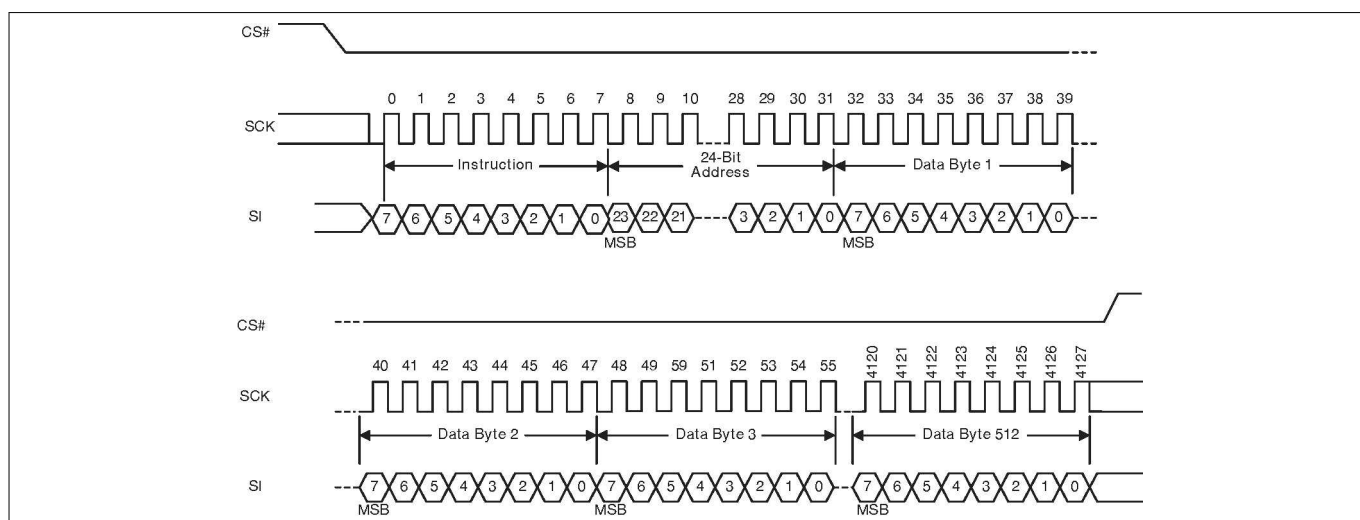
### 11.7 One Time Program Array commands

#### 11.7.1 OTP Program (OTPP 42 h)

The OTP Program command programs data in the One Time Program region, which is in a different address space from the main array data. The OTP region is 1024 bytes so, the address bits from A23 to A10 must be zero for this command. Refer to [OTP address space](#) 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).

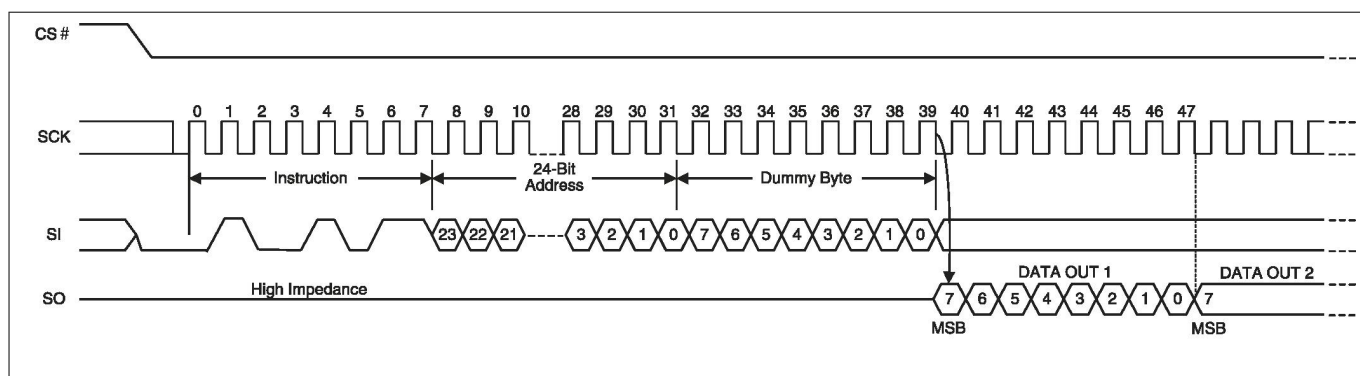


**Figure 112** OTP Program command sequence

#### 11.7.2 OTP Read (OTPR 4 Bh)

The OTP Read command reads data from the OTP region. The OTP region is 1024 bytes so, the address bits from A23 to A10 must be zero for this command. Refer to [OTP address space](#) 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.

## 11 Commands

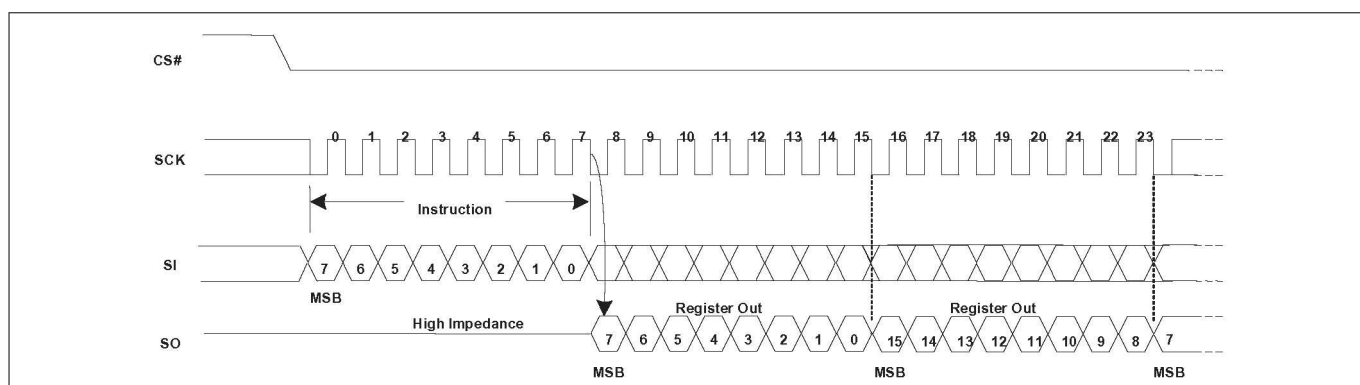


**Figure 113** OTP Read command sequence

## 11.8 Advanced Sector Protection commands

### 11.8.1 ASP Read (ASPRD 2 Bh)

The ASP Read instruction 2 Bh 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 114** ASPRD command

### 11.8.2 ASP Program (ASPP 2 Fh)

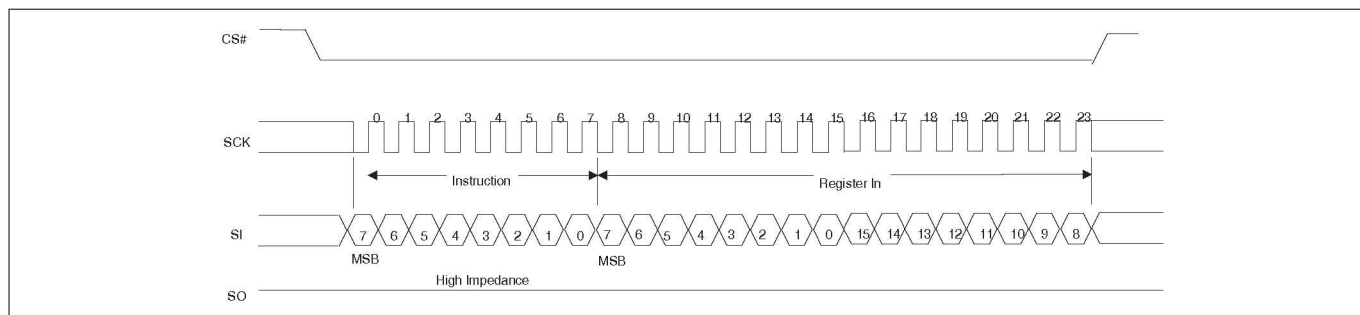
Before the ASP Program (ASPP) command can be accepted by the device, a Write Enable (WREN) command must be issued. After the Write Enable (WREN) command has been decoded, the device will set the Write Enable Latch (WEL) in the Status Register to enable any write operations.

The ASPP command is entered by driving CS# to the logic low state, followed by the instruction and two data bytes on SI, least significant byte first. The ASP Register is two data bytes in length.

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

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

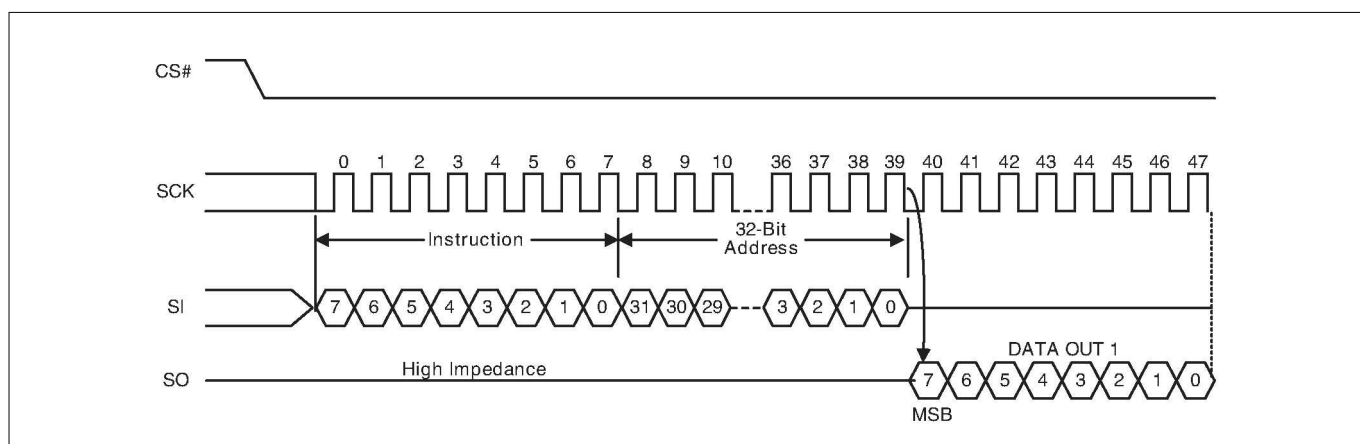
## 11 Commands



**Figure 115 ASPP command**

### 11.8.3 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 116 DYBRD command sequence**

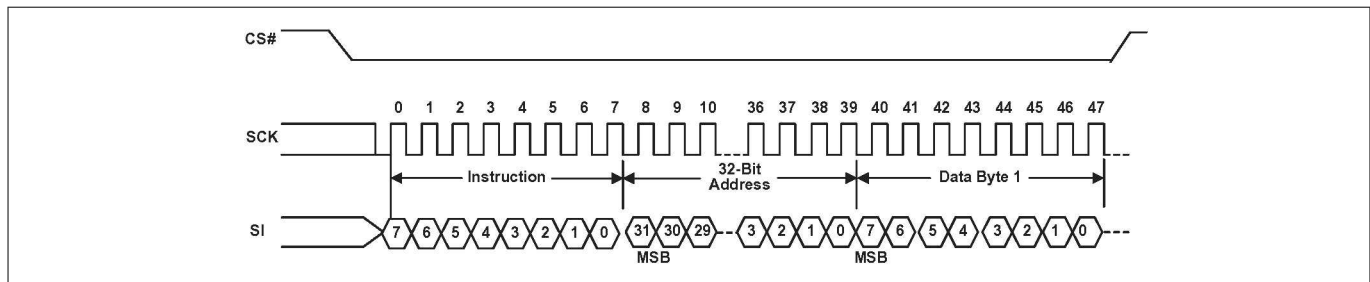
### 11.8.4 DYB Write (DYBWR E1h)

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 zero), then the data byte on SI. The DYB Access Register is one data byte in length.

The DYBWR command affects the P\_ERR and WIP bits of the Status and Configuration Registers in the same manner as any other programming operation. CS# must be driven to the logic high state after the eighth bit of data has been latched in. If not, the DYBWR command is not executed. As soon as CS# is driven to the logic high state, the self-timed DYBWR operation is initiated. While the DYBWR operation is in progress, the Status Register may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a 1 during the self-timed DYBWR operation, and is a '0' when it is completed. When the DYBWR operation is completed, the Write Enable Latch (WEL) is set to a '0'.

## 11 Commands

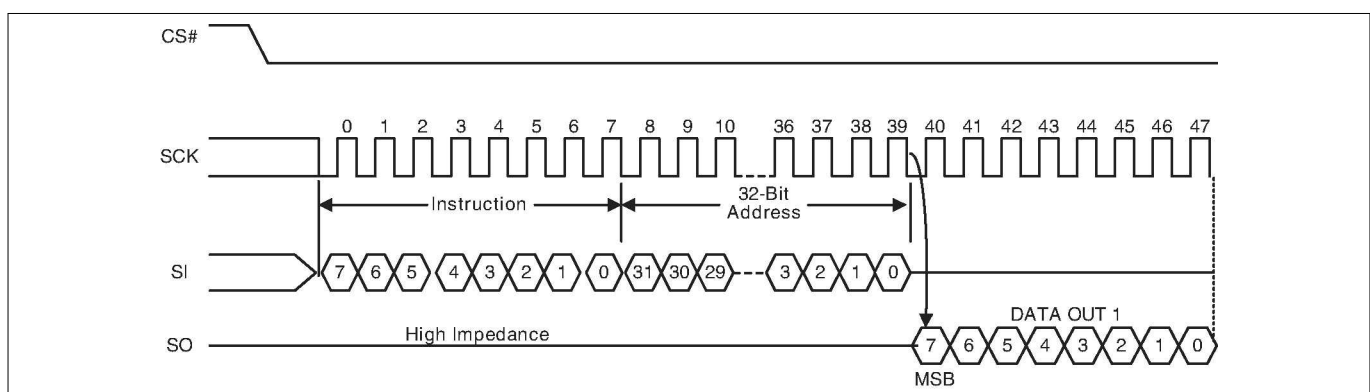


**Figure 117** DYBWR command sequence

### 11.8.5 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 118** PPBRD command sequence

### 11.8.6 PPB Program (PPBP E3h)

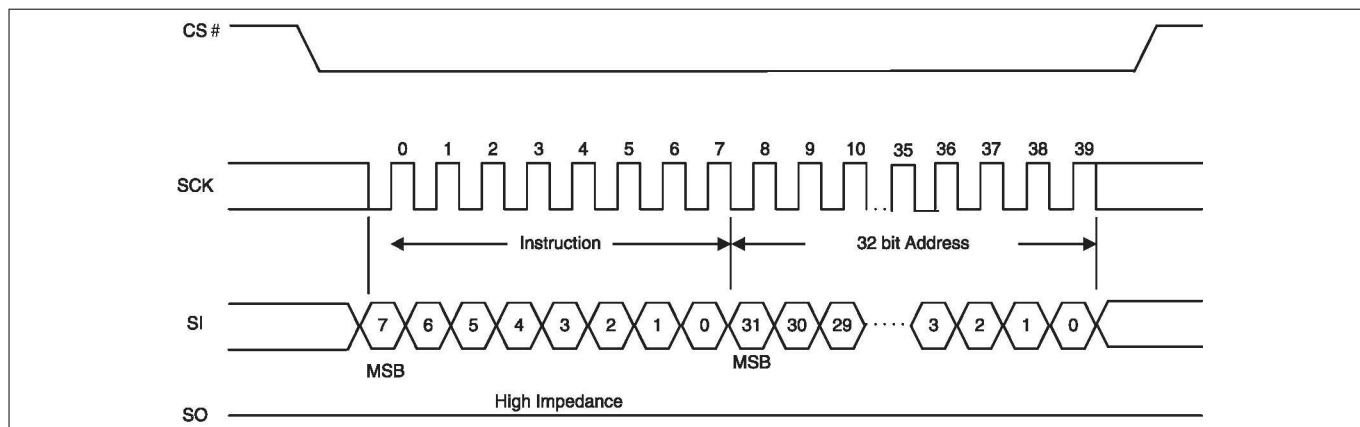
Before the PPB Program (PPBP) command can be accepted by the device, a Write Enable (WREN) command must be issued. After the Write Enable (WREN) command has been decoded, the device will set the Write Enable Latch (WEL) in the Status Register to enable any write operations.

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

## 11 Commands



**Figure 119** PPBP command sequence

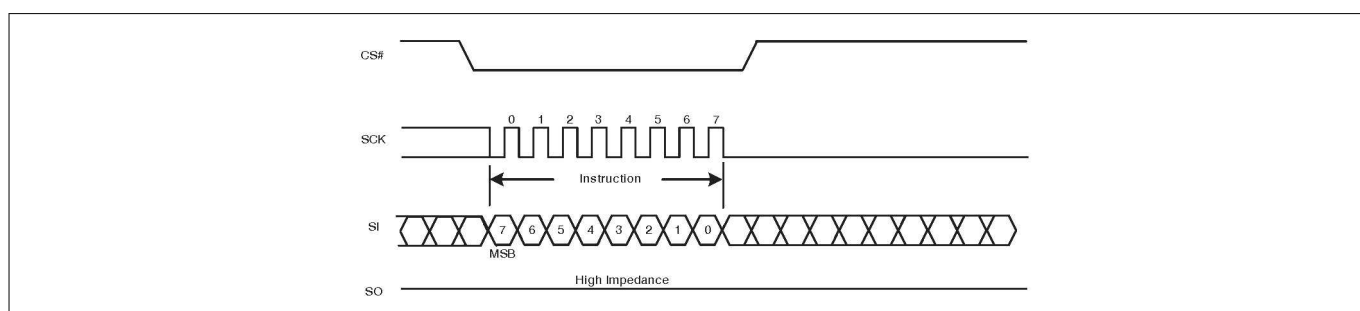
### 11.8.7 PPB Erase (PPBE E4h)

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

The instruction E4h is shifted into SI by the rising edges of the SCK signal.

CS# must be driven into the logic high state after the eighth bit of the instruction byte has been latched in on SI. This will initiate the beginning of internal erase cycle, which involves the pre-programming and erase of the entire PPB memory array. Without CS# being driven to the logic high state after the eighth bit of the instruction, the PPB erase operation will not be executed.

With the internal erase cycle in progress, the user can read the value of the Write-In Progress (WIP) bit to check if the operation has been completed. The WIP bit will indicate a '1' when the erase cycle is in progress and a '0' when the erase cycle has been completed. Erase suspend is not allowed during PPB Erase.



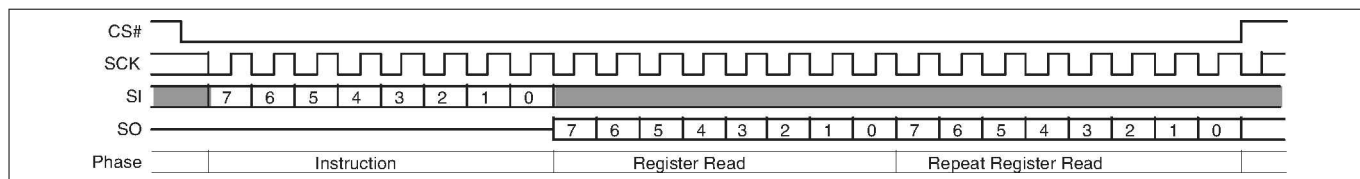
**Figure 120** PPB Erase command sequence

### 11.8.8 PPB Lock Bit Read (PLBRD A7h)

The PPB Lock Bit Read (PLBRD) command allows the PPB Lock Register contents to be read out of SO. 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.



## 11 Commands



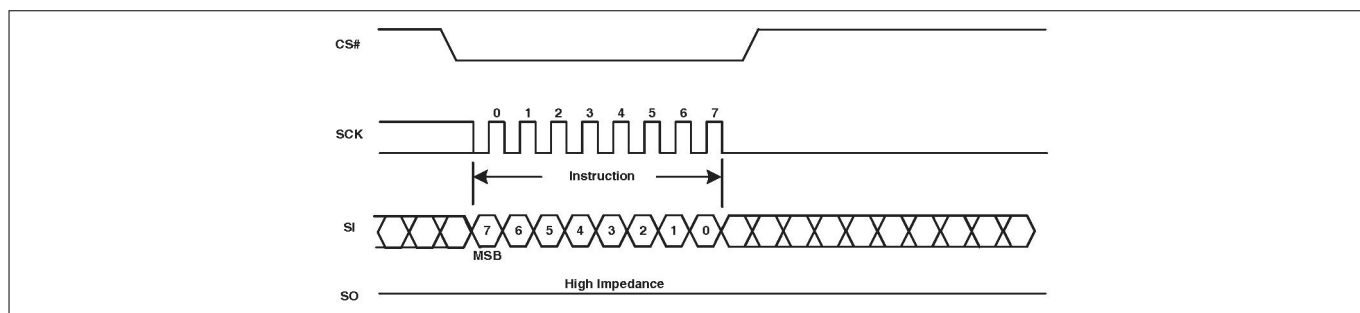
**Figure 121** PPB Lock Register Read command sequence

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

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

The PLBWR command is entered by driving CS# to the logic low state, followed by the instruction.

CS# must be driven to the logic high state after the eighth bit of instruction has been latched in. If not, the PLBWR command is not executed. As soon as CS# is driven to the logic high state, the self-timed PLBWR operation is initiated. While the PLBWR operation is in progress, the Status Register may still be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a 1 during the self-timed PLBWR operation, and is a 0 when it is completed. When the PLBWR operation is completed, the Write Enable Latch (WEL) is set to a 0. The maximum clock frequency for the PLBWR command is 133 MHz.



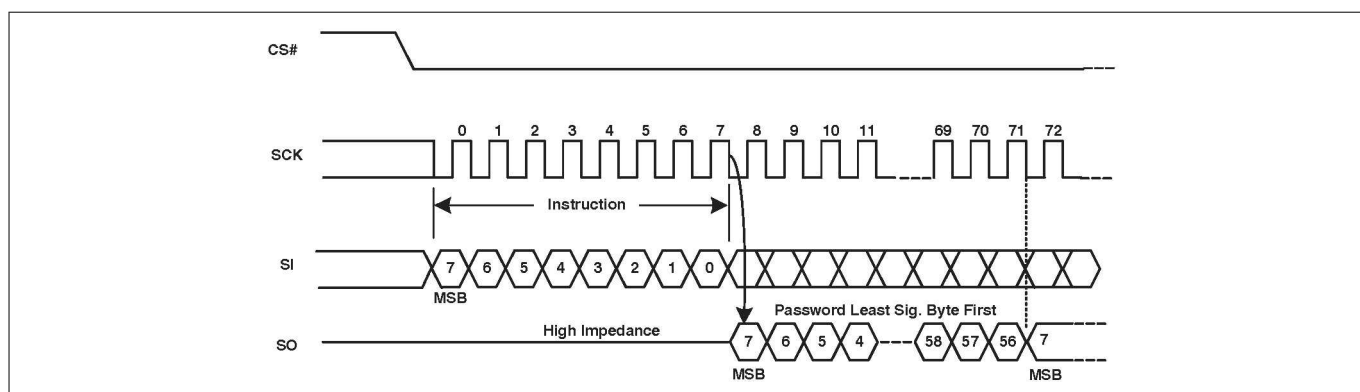
**Figure 122** PPB Lock Bit Write command sequence

### 11.8.10 Password Read (PASSRD E7h)

The correct password value may be read only after it is programmed and before the Password Mode has been selected by programming the Password Protection Mode bit to 0 in the ASP Register (ASP[2]). After the Password Protection Mode is selected the 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.

## 11 Commands



**Figure 123 Password Read command sequence**

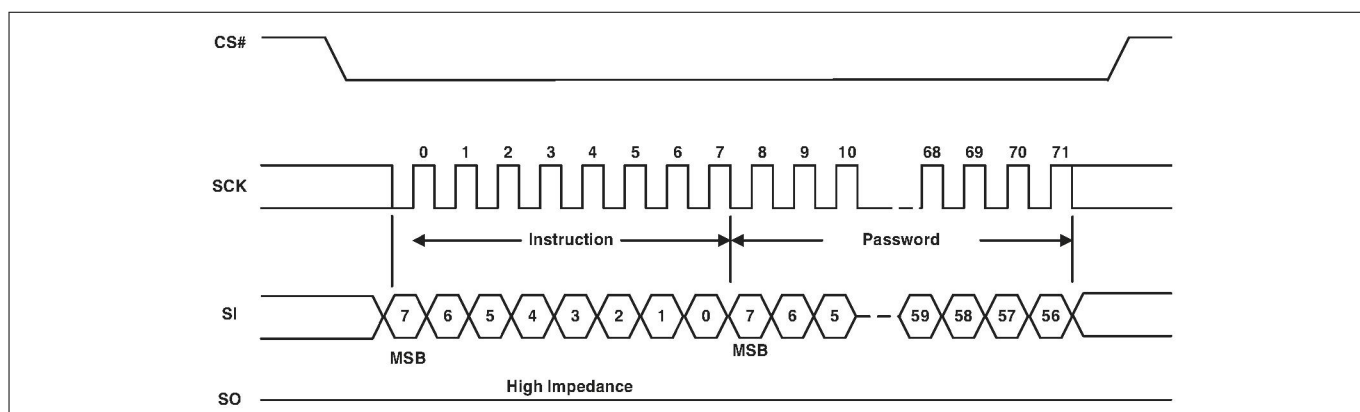
### 11.8.11 Password Program (PASSP E8h)

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

The password can only be programmed before the Password Mode is selected by programming the Password Protection Mode bit to 0 in the ASP Register (ASP[2]). After the Password Protection Mode is selected the PASSP command is ignored.

The PASSP command is entered by driving CS# to the logic low state, followed by the instruction and the password data bytes on SI, 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 (64th) bit of data has been latched. If not, the PASSP command is not executed. As soon as CS# is driven to the logic high state, the self-timed PASSP operation is initiated. While the PASSP operation is in progress, the Status Register may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a 1 during the self-timed PASSP cycle, and is a '0' when it is completed. The PASSP command can report a program error in the P\_ERR bit of the status register. When the PASSP operation is completed, the Write Enable Latch (WEL) is set to a '0'. The maximum clock frequency for the PASSP command is 133 MHz.



**Figure 124 Password Program command sequence**

### 11.8.12 Password Unlock (PASSU E9h)

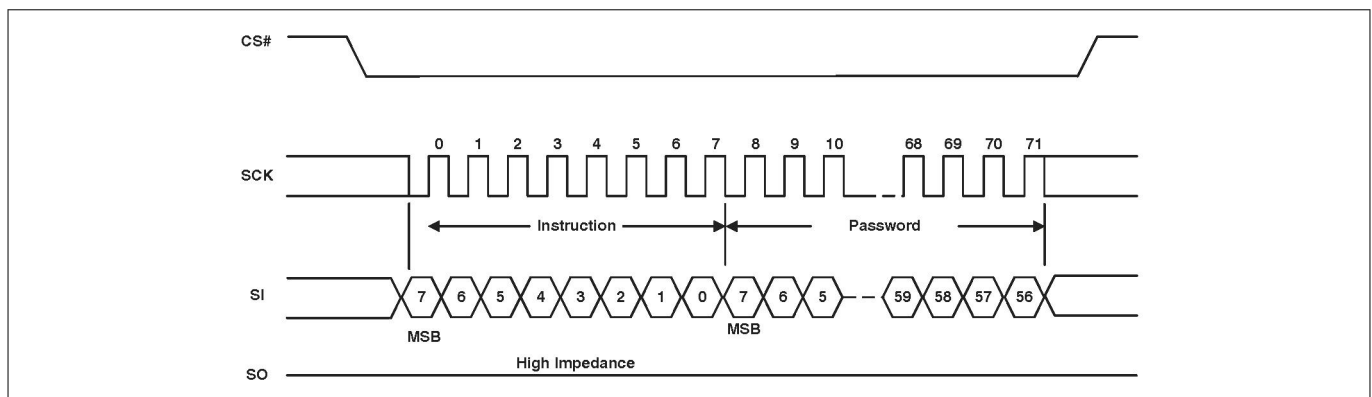
The PASSU command is entered by driving CS# to the logic low state, followed by the instruction and the password data bytes on SI, least significant byte first, most significant bit of each byte first. The password is sixty-four (64) bits in length.

## 11 Commands

CS# must be driven to the logic high state after the sixty-fourth (64th) bit of data has been latched. If not, the PASSU command is not executed. As soon as CS# is driven to the logic high state, the self-timed PASSU operation is initiated. While the PASSU operation is in progress, the Status Register may be read to check the value of the Write-In Progress (WIP) bit. The Write-In Progress (WIP) bit is a 1 during the self-timed PASSU cycle, and is a '0' when it is completed.

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

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



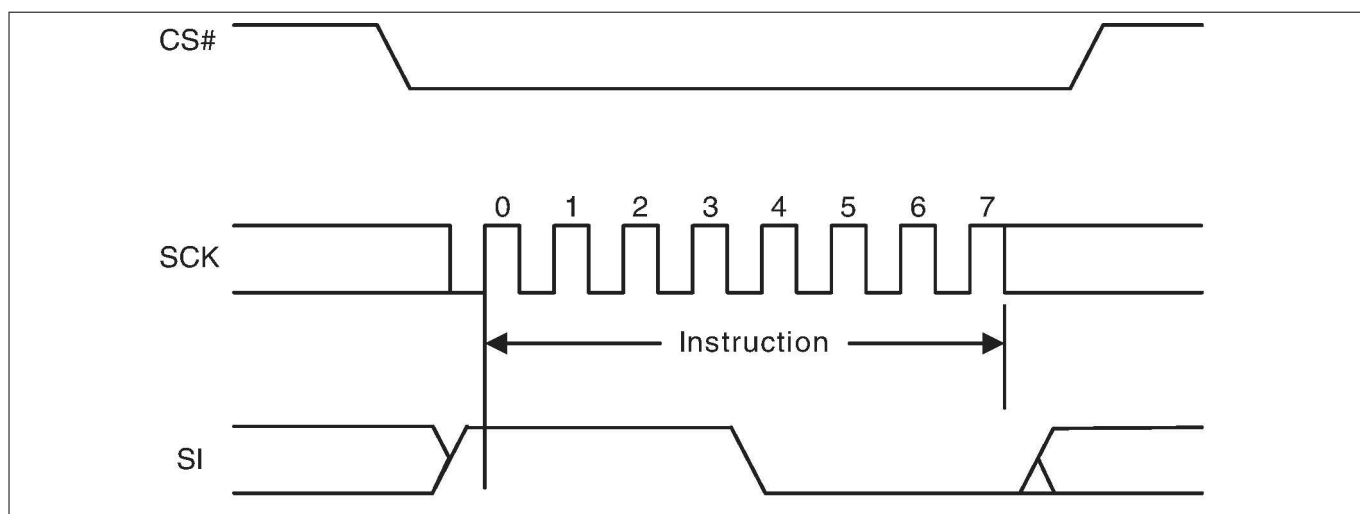
**Figure 125 Password Unlock command sequence**

## 11.9 Reset commands

### 11.9.1 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 zero. 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.

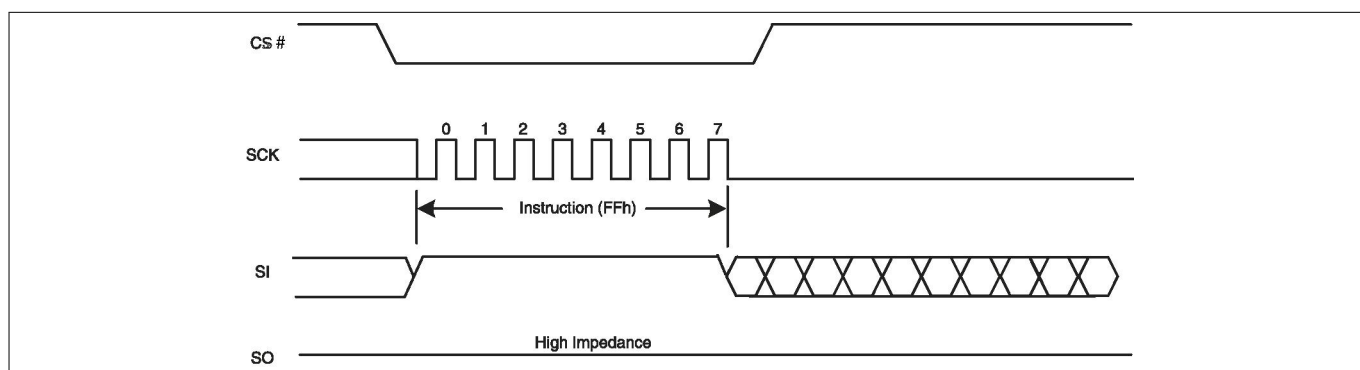
## 11 Commands



**Figure 126** Software Reset command sequence

### 11.9.2 Mode Bit Reset (MBR FFh)

The Mode Bit Reset (MBR) command can be used to return the device from continuous high performance read mode back to normal standby awaiting any new command. Because some device packages lack a hardware RESET# input and a device that is in a continuous high performance read mode may not recognize any normal SPI command, a system hardware reset or software reset command may not be recognized by the device. It is recommended to use the MBR command after a system reset when the RESET# signal is not available or, before sending a software reset, to ensure the device is released from continuous high performance read mode. The MBR command sends Ones on SI or IO0 for 8 SCK cycles. IO1 to IO3 are “don't care” during these cycles. The MBR command sends Ones on SI or IO0 for 8 SCK cycles. IO1 to IO3 are “don't care” during these cycles.



**Figure 127** Mode Bit Reset command sequence

## 12 Data integrity

## 12 Data integrity

### 12.1 Erase endurance

**Table 58 Erase endurance**

Parameter	Minimum	Unit
Program/Erase cycles per main flash array sectors	100	PE cycle
Program/Erase cycles per PPB array or non-volatile register array <sup>1)</sup>	100	PE cycle

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

### 12.2 Data retention

**Table 59 Data retention**

Parameter	Test conditions	Minimum time	Unit
Data retention time	100 Program/Erase Cycles	> 20	Years

## 13 Device identification

### 13 Device identification

#### 13.1 Command summary

**Table 60 S25FL128S and S25FL256S instruction set (sorted by instruction)**

Instruction (Hex)	Command name	Command description	Maximum frequency (MHz)
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
0D	DDRFR	DDR Fast Read (3- or 4-byte address)	80
0E	4DDRFR	DDR Fast Read (4-byte address)	80
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 4 KB-sector Erase (3- or 4-byte address)	133
21	4P4E	Parameter 4 KB-sector Erase (4-byte address)	133
2B	ASPRD	ASP Read	133
2F	ASPP	ASP Program	133

**(table continues...)**

## 13 Device identification

**Table 60** (continued) S25FL128S and S25FL256S instruction set (sorted by instruction)

Instruction (Hex)	Command name	Command description	Maximum frequency (MHz)
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
3B	DOR	Read Dual Out (3- or 4-byte address)	104
3C	4DOR	Read Dual Out (4-byte address)	104
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
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

(table continues...)

**13 Device identification**

**Table 60 (continued) S25FL128S and S25FL256S instruction set (sorted by instruction)**

Instruction (Hex)	Command name	Command description	Maximum frequency (MHz)
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
BB	DIOR	Dual I/O Read (3- or 4-byte address)	104
BC	4DIOR	Dual I/O Read (4-byte address)	104
BD	DDRDIOR	DDR Dual I/O Read (3- or 4-byte address)	80
BE	4DDRDIOR	DDR Dual I/O Read (4-byte address)	80
C7	BE	Bulk Erase (alternate command)	133
D8	SE	Erase 64 KB or 256 KB (3- or 4-byte address)	133
DC	4SE	Erase 64 KB or 256 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

**(table continues...)**



## 13 Device identification

**Table 60** (continued) S25FL128S and S25FL256S instruction set (sorted by instruction)

Instruction (Hex)	Command name	Command description	Maximum frequency (MHz)
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

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

### 13.2.1 Field definitions

**Table 61** Manufacturer and device ID

Byte address	Data	Description
00 h	01 h	Manufacturer ID for Infineon
01 h	20 h (128 Mb) 02 h (256 Mb)	Device ID Most Significant Byte - Memory Interface Type
02 h	18 h (128 Mb) 19 h (256 Mb)	Device ID Least Significant Byte - Density
03 h	4 Dh	ID-CFI Length - number bytes following. Adding this value to the current location of 03 h gives the address of the last valid location in the ID-CFI address map. A value of 00 h 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.
04 h	00 h (Uniform 256-KB sectors) 01 h (4-KB parameter sectors with uniform 64-KB sectors)	Sector Architecture
05 h	80 h (FL-S Family)	Family ID
06 h	xxh	ASCII characters for Model. See <a href="#">Ordering information</a> for the model number definitions.
07 h	xxh	
08 h	xxh	Reserved
09 h	xxh	Reserved

(table continues...)

## 13 Device identification

**Table 61** (continued) **Manufacturer and device ID**

Byte address	Data	Description
0 Ah	xxh	Reserved
0 Bh	xxh	Reserved
0 Ch	xxh	Reserved
0 Dh	xxh	Reserved
0 Eh	xxh	Reserved
0 Fh	xxh	Reserved

**Table 62** **CFI query identification string**

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

**Table 63** **CFI system interface string**

Byte address	Data	Description
1 Bh	27 h	V <sub>CC</sub> Min. (erase/program): 100 millivolts
1 Ch	36 h	V <sub>CC</sub> Max. (erase/program): 100 millivolts
1 Dh	00 h	V <sub>PP</sub> Min. voltage (00 h = no V <sub>PP</sub> present)
1 Eh	00 h	V <sub>PP</sub> Max. voltage (00 h = no V <sub>PP</sub> present)
1 Fh	06 h	Typical timeout per single byte program 2 <sup>N</sup> μs
20 h	08 h (256 B page) 09 h (512 B page)	Typical timeout for Min. size Page program 2 <sup>N</sup> μs (00 h = not supported)

(table continues...)

**13 Device identification**

**Table 63 (continued) CFI system interface string**

Byte address	Data	Description
21 h	08 h (4 KB or 64 KB) 09 h (256 KB)	Typical timeout per individual sector erase $2^N$ ms
22 h	0 Fh (128 Mb) 10 h (256 Mb)	Typical timeout for full chip erase $2^N$ ms (00 h = not supported)
23 h	02 h	Max. timeout for byte program $2^N$ times typical
24 h	02 h	Max. timeout for page program $2^N$ times typical
25 h	03 h	Max. timeout per individual sector erase $2^N$ times typical
26 h	03 h	Max. timeout for full chip erase $2^N$ times typical (00 h = not supported)

**Table 64 Device geometry definition for 128-Mb and 256-Mb bottom boot initial delivery state**

Byte address	Data	Description
27 h	18 h (128 Mb) 19 h (256 Mb)	Device size = $2^N$ bytes
28 h	02 h	Flash device interface description: 0000 h = x8 only 0001 h = x16 only 0002 h = x8/x16 capable 0003 h = x32 only 0004 h = Single I/O SPI, 3-byte address 0005 h = Multi I/O SPI, 3-byte address 0102 h = Multi I/O SPI, 3- or 4-byte address
29 h	01 h	
2 Ah	08 h	Max. number of bytes in multi-byte write = $2^N$ (0000 = Not supported 0008 h = 256 B page 0009 h = 512 B page)
2 Bh	00 h	
2 Ch	02 h	Number of Erase Block Regions within device 1 = Uniform Device, 2 = Boot Device
2 Dh	1 Fh	Erase Block Region 1 Information (refer to JEDEC JEP137): 32 sectors = 32-1 = 001 Fh
2 Eh	00 h	
2 Fh	10 h	

**(table continues...)**

## 13 Device identification

**Table 64** (continued) **Device geometry definition for 128-Mb and 256-Mb bottom boot initial delivery state**

Byte address	Data	Description
30 h	00 h	4-KB sectors = 256 bytes x 0010 h
31 h	FDh	Erase Block Region 2 Information:
32 h	00 h (128 Mb) 01 h (256 Mb)	254 sectors = 254-1 = 00 FDh (128 Mb)
33 h	00 h	510 sectors = 510-1 = 01 FDh (256 Mb)
34 h	01 h	64-KB sectors = 0100 h x 256 bytes
35 h thru 3 Fh	FFh	RFU

### Notes:

1. *FL-S 128 Mb and 256-Mb devices have either a hybrid sector architecture with thirty two 4-KB sectors and all remaining sectors of 64-KB or with uniform 256-KB sectors. Devices with the hybrid sector architecture are initially shipped from Infineon with the 4 KB sectors located at the bottom of the array address map. However, the device configuration TBPARM bit CR1[2] may be programmed to invert the sector map to place the 4-KB sectors at the top of the array address map. The CFI geometry information of the above table is relevant only to the initial delivery state of a hybrid sector device. The flash device driver software must examine the TB-PARM bit to determine if the sector map was inverted at a later time.*

**Table 65** **Device geometry definition for 128-Mb and 256-Mb uniform sector devices**

Byte address	Data	Description
27 h	18 h (128 Mb) 19 h (256 Mb)	Device size = 2N bytes
28 h	02 h	Flash device interface description:
29 h	01 h	0000 h = x8 only 0001 h = x16 only 0002 h = x8/x16 capable 0003 h = x32 only 0004 h = Single I/O SPI, 3-byte address 0005 h = Multi I/O SPI, 3-byte address 0102 h = Multi I/O SPI, 3- or 4-byte address
2 Ah	09 h	Max. number of bytes in multi-byte write = 2 <sup>N</sup>
2 Bh	00 h	(0000 = Not supported 0008 h = 256 B page 0009 h = 512 B page)

**(table continues...)**

### 13 Device identification

**Table 65** (continued) **Device geometry definition for 128-Mb and 256-Mb uniform sector devices**

Byte address	Data	Description
2 Ch	01 h	Number of Erase Block Regions within device 1 = Uniform Device, 2 = Boot Device
2 Dh	3 Fh (128 Mb) 7 Fh (256 Mb)	Erase Block Region 1 Information (refer to JEDEC JEP137):
2 Eh	00 h	64 sectors = 64-1 = 003 Fh (128 Mb)
2 Fh	00 h	128 sectors = 128-1 = 007 Fh (256 Mb)
30 h	04 h	256-KB sectors = 256 bytes x 0400 h
31 h thru 3 Fh	FFh	RFU

**Table 66** **CFI primary vendor-specific extended query**

Byte address	Data	Description
40 h	50 h	Query-unique ASCII string "PRI"
41 h	52 h	
42 h	49 h	
43 h	31 h	Major version number = 1, ASCII
44 h	33 h	Minor version number = 3, ASCII
45 h	21 h	Address Sensitive Unlock (Bits 1-0) 00 b = Required 01 b = Not Required Process Technology (Bits 5-2) 0000 b = 0.23 µm Floating Gate 0001 b = 0.17 µm Floating Gate 0010 b = 0.23 µm MIRRORBIT™ 0011 b = 0.11 µm Floating Gate 0100 b = 0.11 µm MIRRORBIT™ 0101 b = 0.09 µm MIRRORBIT™ 1000 b = 0.065 µm MIRRORBIT™
46 h	02 h	Erase Suspend 0 = Not Supported 1 = Read Only 2 = Read and Program
47 h	01 h	Sector Protect 00 = Not Supported X = Number of sectors in group

**(table continues...)**

**13 Device identification**

**Table 66 (continued) CFI primary vendor-specific extended query**

Byte address	Data	Description
48 h	00 h	Temporary Sector Unprotect 00 = Not Supported 01 = Supported
49 h	08 h	Sector Protect/Unprotect Scheme 04 = High Voltage Method 05 = Software Command Locking Method 08 = Advanced Sector Protection Method 09 = Secure
4 Ah	00 h	Simultaneous Operation 00 = Not Supported X = Number of Sectors
4 Bh	01 h	Burst Mode (Synchronous sequential read) support 00 = Not Supported 01 = Supported
4 Ch	xxh	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
4 Dh	00 h	ACC (Acceleration) Supply Minimum 00 = Not Supported, 100 mV
4 Eh	00 h	ACC (Acceleration) Supply Maximum 00 = Not Supported, 100 mV
4 Fh	07 h	WP# Protection 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)
50 h	01 h	Program Suspend 00 = Not Supported 01 = Supported

### 13 Device identification

The Alternate Vendor-Specific Extended Query provides information related to the expanded command set provided by the FL-S family. The alternate query parameters use a format in which each parameter begins with an identifier byte and a parameter length byte. Driver software can check each parameter ID and can use the length value to skip to the next parameter if the parameter is not needed or not recognized by the software.

**Table 67 CFI alternate vendor-specific extended query header**

Byte address	Data	Description
51 h	41 h	Query-unique ASCII string "ALT"
52 h	4 Ch	
53 h	54 h	
54 h	32 h	Major version number = 2, ASCII
55 h	30 h	Minor version number = 0, ASCII

**Table 68 CFI alternate vendor-specific extended query parameter 0**

Parameter relative byte address offset	Data	Description
00 h	00 h	Parameter ID (Ordering Part Number)
01 h	10 h	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)
02 h	53 h	ASCII "S" for manufacturer (Infineon)
03 h	32 h	ASCII "25" for Product Characters (Single Die SPI)
04 h	35 h	
05 h	46 h	ASCII "FL" for Interface Characters (SPI 3 Volt)
06 h	4 Ch	
07 h	31 h (128 Mb) 32 h (256 Mb)	ASCII characters for density
08 h	32 h (128 Mb) 35 h (256 Mb)	
09 h	38 h (128 Mb) 36 h (256 Mb)	
0 Ah	53 h	ASCII "S" for Technology (65 nm MIRRORBIT™)
0 Bh	xxh	Reserved for Future Use (RFU)
0 Ch		
0 Dh		
0 Eh		

**(table continues...)**

### 13 Device identification

**Table 68 (continued) CFI alternate vendor-specific extended query parameter 0**

Parameter relative byte address offset	Data	Description
0 Fh		
10 h		
11 h		

**Table 69 CFI alternate vendor-specific extended query parameter 80 h address options**

Parameter relative byte address offset	Data	Description
00 h	80 h	Parameter ID (address options)
01 h	01 h	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)
02 h	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 70 CFI alternate vendor-specific extended query parameter 84 h suspend commands**

Parameter relative byte address offset	Data	Description
00 h	84 h	Parameter ID (Suspend Commands)
01 h	08 h	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)
02 h	85 h	Program suspend instruction code
03 h	2 Dh	Program suspend latency maximum (μs)
04 h	8 Ah	Program resume instruction code
05 h	64 h	Program resume to next suspend typical (μs)

**(table continues...)**



## 13 Device identification

**Table 70** (continued) CFI alternate vendor-specific extended query parameter 84 h suspend commands

Parameter relative byte address offset	Data	Description
06 h	75 h	Erase suspend instruction code
07 h	2 Dh	Erase suspend latency maximum (μs)
08 h	7 Ah	Erase resume instruction code
09 h	64 h	Erase resume to next suspend typical (μs)

**Table 71** CFI alternate vendor-specific extended query parameter 88 h data protection

Parameter relative byte address offset	Data	Description
00 h	88 h	Parameter ID (Data Protection)
01 h	04 h	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)
02 h	0 Ah	OTP size 2 <sup>N</sup> bytes, FFh = not supported
03 h	01 h	OTP address map format, 01 h = FL-S format, FFh = not supported
04 h	xxh	Block Protect Type, model dependent 00 h = FL-P, FL-S, FFh = not supported
05 h	xxh	Advanced Sector Protection type, model dependent 01 h = FL-S ASP

**Table 72** CFI alternate vendor-specific extended query parameter 8 Ch reset timing

Parameter relative byte address offset	Data	Description
00 h	8 Ch	Parameter ID (Reset Timing)
01 h	06 h	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)
02 h	96 h	POR maximum value
03 h	01 h	POR maximum exponent 2 <sup>N</sup> μs

(table continues...)

## 13 Device identification

**Table 72** (continued) CFI alternate vendor-specific extended query parameter 8 Ch reset timing

Parameter relative byte address offset	Data	Description
04 h	FFh (without separate RESET#) 23 h (with separate RESET #)	Hardware Reset maximum value
05 h	00 h	Hardware Reset maximum exponent $2^N \mu s$
06 h	23 h	Software Reset maximum value, FFh = not supported
07 h	00 h	Software Reset maximum exponent $2^N \mu s$

**Table 73** CFI alternate vendor-specific extended query parameter 90 h - HPLC(SDR)

Parameter relative byte address offset	Data	Description
00 h	90 h	Parameter ID (Latency Code Table)
01 h	56 h	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)
02 h	06 h	Number of rows
03 h	0 Eh	Row length in bytes
04 h	46 h	Start of header (row 1), ASCII "F" for frequency column header
05 h	43 h	ASCII "C" for Code column header
06 h	03 h	Read 3-byte address instruction
07 h	13 h	Read 4-byte address instruction
08 h	0 Bh	Read Fast 3-byte address instruction
09 h	0 Ch	Read Fast 4-byte address instruction
0 Ah	3 Bh	Read Dual Out 3-byte address instruction
0 Bh	3 Ch	Read Dual Out 4-byte address instruction
0 Ch	6 Bh	Read Quad Out 3-byte address instruction
0 Dh	6 Ch	Read Quad Out 4-byte address instruction

(table continues...)

### 13 Device identification

**Table 73** (continued) CFI alternate vendor-specific extended query parameter 90 h - HPLC(SDR)

Parameter relative byte address offset	Data	Description
0 Eh	BBh	Dual I/O Read 3-byte address instruction
0 Fh	BCh	Dual I/O Read 4-byte address instruction
10 h	EBh	Quad I/O Read 3-byte address instruction
11 h	ECh	Quad I/O Read 4-byte address instruction
12 h	32 h	Start of row 2, SCK frequency limit for this row (50 MHz)
13 h	03 h	Latency Code for this row (11 b)
14 h	00 h	Read mode cycles
15 h	00 h	Read latency cycles
16 h	00 h	Read Fast mode cycles
17 h	00 h	Read Fast latency cycles
18 h	00 h	Read Dual Out mode cycles
19 h	00 h	Read Dual Out latency cycles
1 Ah	00 h	Read Quad Out mode cycles
1 Bh	00 h	Read Quad Out latency cycles
1 Ch	00 h	Dual I/O Read mode cycles
1 Dh	04 h	Dual I/O Read latency cycles
1 Eh	02 h	Quad I/O Read mode cycles
1 Fh	01 h	Quad I/O Read latency cycles
20 h	50 h	Start of row 3, SCK frequency limit for this row (80 MHz)
21 h	00 h	Latency Code for this row (00 b)
22 h	FFh	Read mode cycles (FFh = command not supported at this frequency)
23 h	FFh	Read latency cycles
24 h	00 h	Read Fast mode cycles
25 h	08 h	Read Fast latency cycles
26 h	00 h	Read Dual Out mode cycles
27 h	08 h	Read Dual Out latency cycles
28 h	00 h	Read Quad Out mode cycles
29 h	08 h	Read Quad Out latency cycles

**(table continues...)**

## 13 Device identification

**Table 73** (continued) CFI alternate vendor-specific extended query parameter 90 h - HPLC(SDR)

Parameter relative byte address offset	Data	Description
2 Ah	00 h	Dual I/O Read mode cycles
2 Bh	04 h	Dual I/O Read latency cycles
2 Ch	02 h	Quad I/O Read mode cycles
2 Dh	04 h	Quad I/O Read latency cycles
2 Eh	5 Ah	Start of row 4, SCK frequency limit for this row (90 MHz)
2 Fh	01 h	Latency Code for this row (01 b)
30 h	FFh	Read mode cycles (FFh = command not supported at this frequency)
31 h	FFh	Read latency cycles
32 h	00 h	Read Fast mode cycles
33 h	08 h	Read Fast latency cycles
34 h	00 h	Read Dual Out mode cycles
35 h	08 h	Read Dual Out latency cycles
36 h	00 h	Read Quad Out mode cycles
37 h	08 h	Read Quad Out latency cycles
38 h	00 h	Dual I/O Read mode cycles
39 h	05 h	Dual I/O Read latency cycles
3 Ah	02 h	Quad I/O Read mode cycles
3 Bh	04 h	Quad I/O Read latency cycles
3 Ch	68 h	Start of row 5, SCK frequency limit for this row (104 MHz)
3 Dh	02 h	Latency Code for this row (10 b)
3 Eh	FFh	Read mode cycles (FFh = command not supported at this frequency)
3 Fh	FFh	Read latency cycles
40 h	00 h	Read Fast mode cycles
41 h	08 h	Read Fast latency cycles
42 h	00 h	Read Dual Out mode cycles
43 h	08 h	Read Dual Out latency cycles
44 h	00 h	Read Quad Out mode cycles
45 h	08 h	Read Quad Out latency cycles
46 h	00 h	Dual I/O Read mode cycles
47 h	06 h	Dual I/O Read latency cycles

**(table continues...)**

## 13 Device identification

**Table 73** (continued) CFI alternate vendor-specific extended query parameter 90 h - HPLC(SDR)

Parameter relative byte address offset	Data	Description
48 h	02 h	Quad I/O Read mode cycles
49 h	05 h	Quad I/O Read latency cycles
4 Ah	85 h	Start of row 6, SCK frequency limit for this row (133 MHz)
4 Bh	02 h	Latency Code for this row (10 b)
4 Ch	FFh	Read mode cycles (FFh = command not supported at this frequency)
4 Dh	FFh	Read latency cycles
4 Eh	00 h	Read Fast mode cycles
4 Fh	08 h	Read Fast latency cycles
50 h	FFh	Read Dual Out mode cycles
51 h	FFh	Read Dual Out latency cycles
52 h	FFh	Read Quad Out mode cycles
53 h	FFh	Read Quad Out latency cycles
54 h	FFh	Dual I/O Read mode cycles
55 h	FFh	Dual I/O Read latency cycles
56 h	FFh	Quad I/O Read mode cycles
57 h	FFh	Quad I/O Read latency cycles

**Table 74** CFI alternate vendor-specific extended query Parameter 9 Ah - HPLC DDR

Parameter relative byte address offset	Data	Description
00 h	9 Ah	Parameter ID (Latency Code Table)
01 h	2 Ah	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)
02 h	05 h	Number of rows
03 h	08 h	Row length in bytes
04 h	46 h	Start of header (row 1), ASCII "F" for frequency column header
05 h	43 h	ASCII "C" for Code column header
06 h	0 Dh	Read Fast DDR 3-byte address instruction

**(table continues...)**

## 13 Device identification

**Table 74** (continued) CFI alternate vendor-specific extended query Parameter 9 Ah - HPLC DDR

Parameter relative byte address offset	Data	Description
07 h	0 Eh	Read Fast DDR 4-byte address instruction
08 h	BDh	DDR Dual I/O Read 3-byte address instruction
09 h	BEh	DDR Dual I/O Read 4-byte address instruction
0 Ah	EDh	Read DDR Quad I/O 3-byte address instruction
0 Bh	EEh	Read DDR Quad I/O 4-byte address instruction
0 Ch	32 h	Start of row 2, SCK frequency limit for this row (50 MHz)
0 Dh	03 h	Latency Code for this row (11 b)
0 Eh	00 h	Read Fast DDR mode cycles
0 Fh	04 h	Read Fast DDR latency cycles
10 h	00 h	DDR Dual I/O Read mode cycles
11 h	04 h	DDR Dual I/O Read latency cycles
12 h	01 h	Read DDR Quad I/O mode cycles
13 h	03 h	Read DDR Quad I/O latency cycles
14 h	42 h	Start of row 3, SCK frequency limit for this row (66 MHz)
15 h	00 h	Latency Code for this row (00 b)
16 h	00 h	Read Fast DDR mode cycles
17 h	05 h	Read Fast DDR latency cycles
18 h	00 h	DDR Dual I/O Read mode cycles
19 h	06 h	DDR Dual I/O Read latency cycles
1 Ah	01 h	Read DDR Quad I/O mode cycles
1 Bh	06 h	Read DDR Quad I/O latency cycles
1 Ch	42 h	Start of row 4, SCK frequency limit for this row (66 MHz)
1 Dh	01 h	Latency Code for this row (01 b)
1 Eh	00 h	Read Fast DDR mode cycles
1 Fh	06 h	Read Fast DDR latency cycles
20 h	00 h	DDR Dual I/O Read mode cycles
21 h	07 h	DDR Dual I/O Read latency cycles

**(table continues...)**

## 13 Device identification

**Table 74 (continued) CFI alternate vendor-specific extended query Parameter 9 Ah - HPLC DDR**

Parameter relative byte address offset	Data	Description
22 h	01 h	Read DDR Quad I/O mode cycles
23 h	07 h	Read DDR Quad I/O latency cycles
24 h	42 h	Start of row 5, SCK frequency limit for this row (66 MHz)
25 h	02 h	Latency Code for this row (10 b)
26 h	00 h	Read Fast DDR mode cycles
27 h	07 h	Read Fast DDR latency cycles
28 h	00 h	DDR Dual I/O Read mode cycles
29 h	08 h	DDR Dual I/O Read latency cycles
2 Ah	01 h	Read DDR Quad I/O mode cycles
2 Bh	08 h	Read DDR Quad I/O latency cycles

**Table 75 CFI alternate vendor-specific extended query parameter 90 h - EHPLC (SDR)**

Parameter relative byte address offset	Data	Description
00 h	90 h	Parameter ID (Latency Code Table)
01 h	56 h	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)
02 h	06 h	Number of rows
03 h	0 Eh	Row length in bytes
04 h	46 h	Start of header (row 1), ASCII "F" for frequency column header
05 h	43 h	ASCII "C" for Code column header
06 h	03 h	Read 3-byte address instruction
07 h	13 h	Read 4-byte address instruction
08 h	0 Bh	Read Fast 3-byte address instruction
09 h	0 Ch	Read Fast 4-byte address instruction
0 Ah	3 Bh	Read Dual Out 3-byte address instruction
0 Bh	3 Ch	Read Dual Out 4-byte address instruction

**(table continues...)**

### 13 Device identification

**Table 75** (continued) CFI alternate vendor-specific extended query parameter 90 h - EHPLC (SDR)

Parameter relative byte address offset	Data	Description
0 Ch	6 Bh	Read Quad Out 3-byte address instruction
0 Dh	6 Ch	Read Quad Out 4-byte address instruction
0 Eh	BBh	Dual I/O Read 3-byte address instruction
0 Fh	BCh	Dual I/O Read 4-byte address instruction
10 h	EBh	Quad I/O Read 3-byte address instruction
11 h	ECh	Quad I/O Read 4-byte address instruction
12 h	32 h	Start of row 2, SCK frequency limit for this row (50 MHz)
13 h	03 h	Latency Code for this row (11 b)
14 h	00 h	Read mode cycles
15 h	00 h	Read latency cycles
16 h	00 h	Read Fast mode cycles
17 h	00 h	Read Fast latency cycles
18 h	00 h	Read Dual Out mode cycles
19 h	00 h	Read Dual Out latency cycles
1 Ah	00 h	Read Quad Out mode cycles
1 Bh	00 h	Read Quad Out latency cycles
1 Ch	04 h	Dual I/O Read mode cycles
1 Dh	00 h	Dual I/O Read latency cycles
1 Eh	02 h	Quad I/O Read mode cycles
1 Fh	01 h	Quad I/O Read latency cycles
20 h	50 h	Start of row 3, SCK frequency limit for this row (80 MHz)
21 h	00 h	Latency Code for this row (00 b)
22 h	FFh	Read mode cycles (FFh = command not supported at this frequency)
23 h	FFh	Read latency cycles
24 h	00 h	Read Fast mode cycles
25 h	08 h	Read Fast latency cycles

**(table continues...)**



### 13 Device identification

**Table 75** (continued) CFI alternate vendor-specific extended query parameter 90 h - EHPLC (SDR)

Parameter relative byte address offset	Data	Description
26 h	00 h	Read Dual Out mode cycles
27 h	08 h	Read Dual Out latency cycles
28 h	00 h	Read Quad Out mode cycles
29 h	08 h	Read Quad Out latency cycles
2 Ah	04 h	Dual I/O Read mode cycles
2B h	00 h	Dual I/O Read latency cycles
2 Ch	02 h	Quad I/O Read mode cycles
2 Dh	04 h	Quad I/O Read latency cycles
2 Eh	5 Ah	Start of row 4, SCK frequency limit for this row (90 MHz)
2 Fh	01 h	Latency Code for this row (01 b)
30 h	FFh	Read mode cycles (FFh = command not supported at this frequency)
31 h	FFh	Read latency cycles
32 h	00 h	Read Fast mode cycles
33 h	08 h	Read Fast latency cycles
34 h	00 h	Read Dual Out mode cycles
35 h	08 h	Read Dual Out latency cycles
36 h	00 h	Read Quad Out mode cycles
37 h	08 h	Read Quad Out latency cycles
38 h	04 h	Dual I/O Read mode cycles
39 h	01 h	Dual I/O Read latency cycles
3 Ah	02 h	Quad I/O Read mode cycles
3 Bh	04 h	Quad I/O Read latency cycles
3 Ch	68 h	Start of row 5, SCK frequency limit for this row (104 MHz)
3 Dh	02 h	Latency Code for this row (10b)
3 Eh	FFh	Read mode cycles (FFh = command not supported at this frequency)
3 Fh	FFh	Read latency cycles
40 h	00 h	Read Fast mode cycles
41 h	08 h	Read Fast latency cycles
42 h	00 h	Read Dual Out mode cycles

**(table continues...)**

### 13 Device identification

**Table 75** (continued) CFI alternate vendor-specific extended query parameter 90 h - EHPLC (SDR)

Parameter relative byte address offset	Data	Description
43 h	08 h	Read Dual Out latency cycles
44 h	00 h	Read Quad Out mode cycles
45 h	08 h	Read Quad Out latency cycles
46 h	04 h	Dual I/O Read mode cycles
47 h	02 h	Dual I/O Read latency cycles
48 h	02 h	Quad I/O Read mode cycles
49 h	05 h	Quad I/O Read latency cycles
4 Ah	85 h	Start of row 6, SCK frequency limit for this row (133 MHz)
4 Bh	02 h	Latency Code for this row (10 b)
4 Ch	FFh	Read mode cycles (FFh = command not supported at this frequency)
4 Dh	FFh	Read latency cycles
4 Eh	00 h	Read Fast mode cycles
4 Fh	08 h	Read Fast latency cycles
50 h	FFh	Read Dual Out mode cycles
51 h	FFh	Read Dual Out latency cycles
52 h	FFh	Read Quad Out mode cycles
53 h	FFh	Read Quad Out latency cycles
54 h	FFh	Dual I/O Read mode cycles
55 h	FFh	Dual I/O Read latency cycles
56 h	FFh	Quad I/O Read mode cycles
57 h	FFh	Quad I/O Read latency cycles

**Table 76** CFI alternate vendor-specific extended query parameter 9 Ah - EHPLC DDR

Parameter relative byte address offset	Data	Description
00 h	9 Ah	Parameter ID (Latency Code Table)
01 h	2 Ah	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)
02 h	05 h	Number of rows
03 h	08 h	Row length in bytes

(table continues...)

### 13 Device identification

**Table 76** (continued) CFI alternate vendor-specific extended query parameter 9 Ah - EHPLC DDR

Parameter relative byte address offset	Data	Description
04 h	46 h	Start of header (row 1), ASCII "F" for frequency column header
05 h	43 h	ASCII "C" for Code column header
06 h	0 Dh	Read Fast DDR 3-byte address instruction
07 h	0 Eh	Read Fast DDR 4-byte address instruction
08 h	BDh	DDR Dual I/O Read 3-byte address instruction
09 h	BEh	DDR Dual I/O Read 4-byte address instruction
0 Ah	EDh	Read DDR Quad I/O 3-byte address instruction
0 Bh	EEh	Read DDR Quad I/O 4-byte address instruction
0 Ch	32 h	Start of row 2, SCK frequency limit for this row (50 MHz)
0 Dh	03 h	Latency Code for this row (11 b)
0 Eh	04 h	Read Fast DDR mode cycles
0 Fh	01 h	Read Fast DDR latency cycles
10 h	02 h	DDR Dual I/O Read mode cycles
11 h	02 h	DDR Dual I/O Read latency cycles
12 h	01 h	Read DDR Quad I/O mode cycles
13 h	03 h	Read DDR Quad I/O latency cycles
14 h	42 h	Start of row 3, SCK frequency limit for this row (66 MHz)
15 h	00 h	Latency Code for this row (00 b)
16 h	04 h	Read Fast DDR mode cycles
17 h	02 h	Read Fast DDR latency cycles
18 h	02 h	DDR Dual I/O Read mode cycles
19 h	04 h	DDR Dual I/O Read latency cycles
1 Ah	01 h	Read DDR Quad I/O mode cycles
1 Bh	06 h	Read DDR Quad I/O latency cycles
1 Ch	42 h	Start of row 4, SCK frequency limit for this row (66 MHz)
1 Dh	01 h	Latency Code for this row (01 b)

**(table continues...)**

## 13 Device identification

**Table 76** (continued) CFI alternate vendor-specific extended query parameter 9 Ah - EHPLC DDR

Parameter relative byte address offset	Data	Description
1 Eh	04 h	Read Fast DDR mode cycles
1 Fh	04 h	Read Fast DDR latency cycles
20 h	02 h	DDR Dual I/O Read mode cycles
21 h	05 h	DDR Dual I/O Read latency cycles
22 h	01 h	Read DDR Quad I/O mode cycles
23 h	07 h	Read DDR Quad I/O latency cycles
24 h	42 h	Start of row 5, SCK frequency limit for this row (66 MHz)
25 h	02 h	Latency Code for this row (10 b)
26 h	04 h	Read Fast DDR mode cycles
27 h	05 h	Read Fast DDR latency cycles
28 h	02 h	DDR Dual I/O Read mode cycles
29 h	06 h	DDR Dual I/O Read latency cycles
2 Ah	01 h	Read DDR Quad I/O mode cycles
2 Bh	08 h	Read DDR Quad I/O latency cycles

**Table 77** CFI alternate vendor-specific extended query parameter F0h RFU

Parameter relative byte address offset	Data	Description
00 h	F0h	Parameter ID (RFU)
01 h	0 Fh	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)
02 h	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.

### 13.3 Device ID and common flash interface (ID-CFI) ASO map — Automotive only

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

### 13 Device identification

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

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

## 13.4 Registers

The register maps are copied in this section as a quick reference. See [Registers](#) for the full description of the register contents.

**Table 78 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

**(table continues...)**

## 13 Device identification

**Table 78 (continued) Status Register 1 (SR1)**

Bits	Field name	Function	Type	Default state	Description
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 Infineon	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

**Table 79 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	

**(table continues...)**

## 13 Device identification

**Table 79** (continued) **Configuration Register (CR1)**

Bits	Field name	Function	Type	Default state	Description
5	TBPROT	Configures Start of Block Protection	OTP	0	1 = BP starts at bottom (Low address) 0 = BP starts at top (High address)
4	RFU	RFU	OTP	0	Reserved for Future Use
3	BPNV	Configures BP2-0 in Status Register	OTP	0	1 = Volatile 0 = Non-volatile
2	TBPARM	Configures Parameter Sectors location	OTP	0	1 = 4-KB physical sectors at top, (high address) 0 = 4-KB physical sectors at bottom (Low address) RFU in uniform sector devices.
1	QUAD	Puts the device into Quad I/O operation	Non-volatile	0	1 = Quad 0 = Dual or Serial
0	FREEZE	Lock current state of BP2-0 bits in Status Register, TBPROT and TBPARM in Configuration Register, and OTP regions	Volatile	0	1 = Block Protection and OTP locked 0 = Block Protection and OTP un-locked

**Table 80** **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

**(table continues...)**

## 13 Device identification

**Table 80 (continued) Status Register 2 (SR2)**

Bits	Field name	Function	Type	Default state	Description
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.

**Table 81 Bank Address Register (BAR)**

Bits	Field name	Function	Type	Default state	Description
7	EXTADD	Extended Address Enable	Volatile	0 b	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	00000 b	Reserved for Future Use
1	BA25	Bank Address	Volatile	0	RFU for lower density devices
0	BA24	Bank Address	Volatile	0	A24 for 256-Mb device, RFU for lower density device

**Table 82 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	<sup>1)</sup>	Reserved for Future Use
7	RFU	Reserved	OTP	<sup>1)</sup>	Reserved for Future Use

(table continues...)



## 13 Device identification

**Table 82** (continued) **ASP Register (ASPR)**

Bits	Field name	Function	Type	Default state	Description
6	RFU	Reserved	OTP	1	Reserved for Future Use
5	RFU	Reserved	OTP	<sup>1)</sup>	Reserved for Future Use
4	RFU	Reserved	OTP	<sup>1)</sup>	Reserved for Future Use
3	RFU	Reserved	OTP	<sup>1)</sup>	Reserved for Future Use
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

<sup>1)</sup> Default value depends on ordering part number, see [Initial delivery state](#)

**Table 83** **Password Register (PASS)**

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

## 13 Device identification

**Table 84 PPB Lock Register (PPBL)**

Bits	Field name	Function	Type	Default state	Description
7 to 1	RFU	Reserved	Volatile	00 h	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

**Table 85 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.

### 13 Device identification

**Table 86**                      **DYB Access Register (DYBAR)**

Bits	Field name	Function	Type	Default state	Description
7 to 0	DYB	Read or Write per sector DYB	Volatile	FFh	00 h = 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.

**Table 87**                      **Non-volatile Data Learning Register (NVDLR)**

Bits	Field name	Function	Type	Default state	Description
7 to 0	NVDLP	Non-volatile Data Learning Pattern	OTP	00 h	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.

---

**13 Device identification**

**Table 88**                      **Volatile Data Learning Register (NVDLR)**

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

## 14 Initial delivery state

### 14 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 00 h (all SR1 bits are cleared to 0's).
- The Configuration Register 1 contains 00 h.
- The Autoboot register contains 00 h.
- The Password Register contains FFFFFFFF-FFFFFFFh.
- All PPB bits are 1.
- The ASP Register contents depend on the ordering options selected:

**Table 89 ASP Register content**

Ordering part number model	ASPR default value
00, 20, 30, R0, A0, B0, C0, D0, 01, 21, 31, R1, A1, B1, C1, D1, 90, Q0, 70, 60, 80, 91, Q1, 71, 61, 81	FE7Fh

15 Physical interface

15 Physical interface

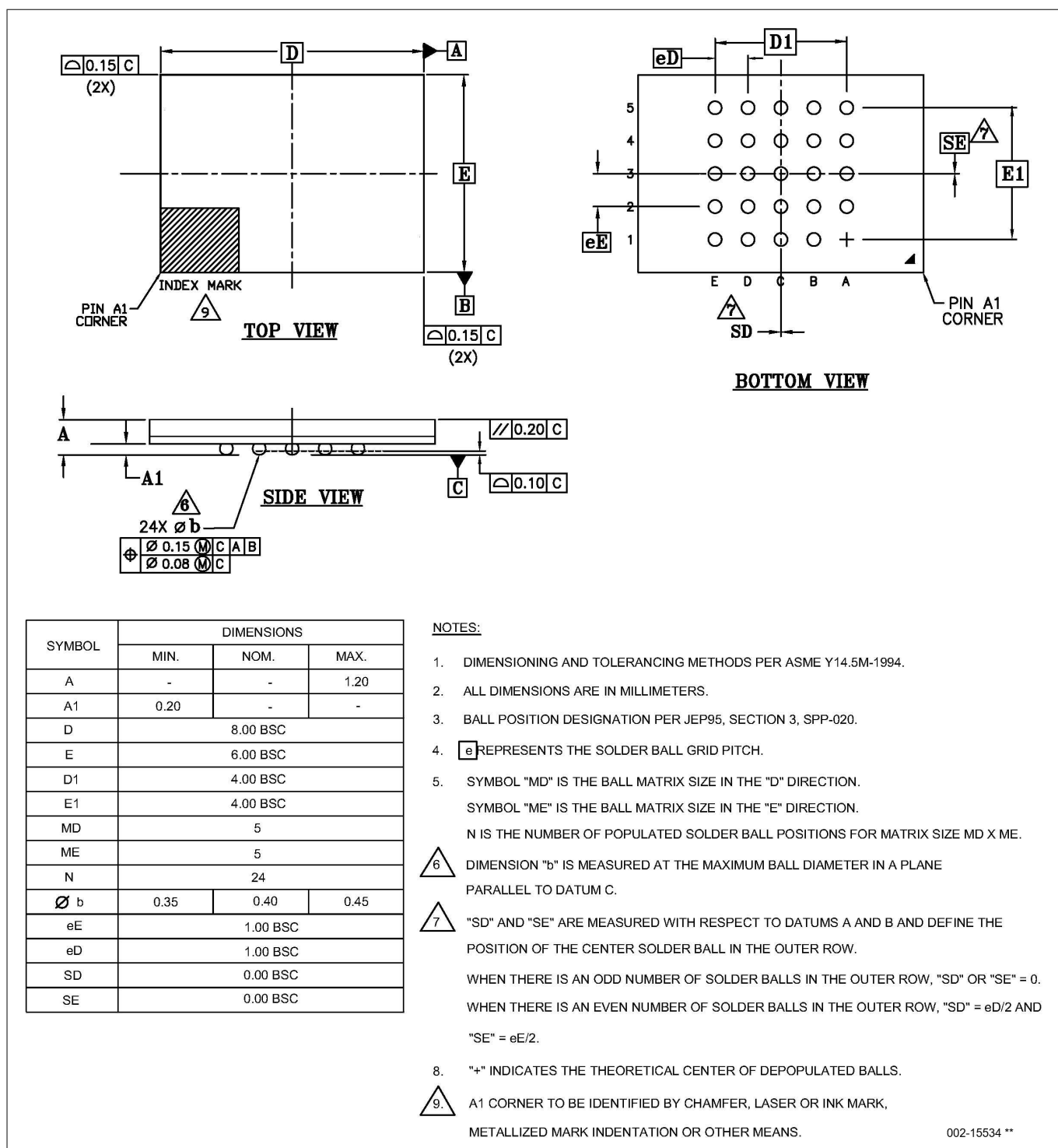
Table 90 Model-specific connections

VIO/RFU	Versatile I/O or RFU — Some device models bond this connector to the device I/O power supply, other models bond the device I/O supply to $V_{CC}$ within the package leaving this package connector unconnected.
RESET# / RFU	RESET# or RFU — Some device models bond this connector to the device RESET# signal, other models bond the RESET# signal to $V_{CC}$ within the package leaving this package connector unconnected.

**Note:** Refer to [Table 7](#) for signal descriptions

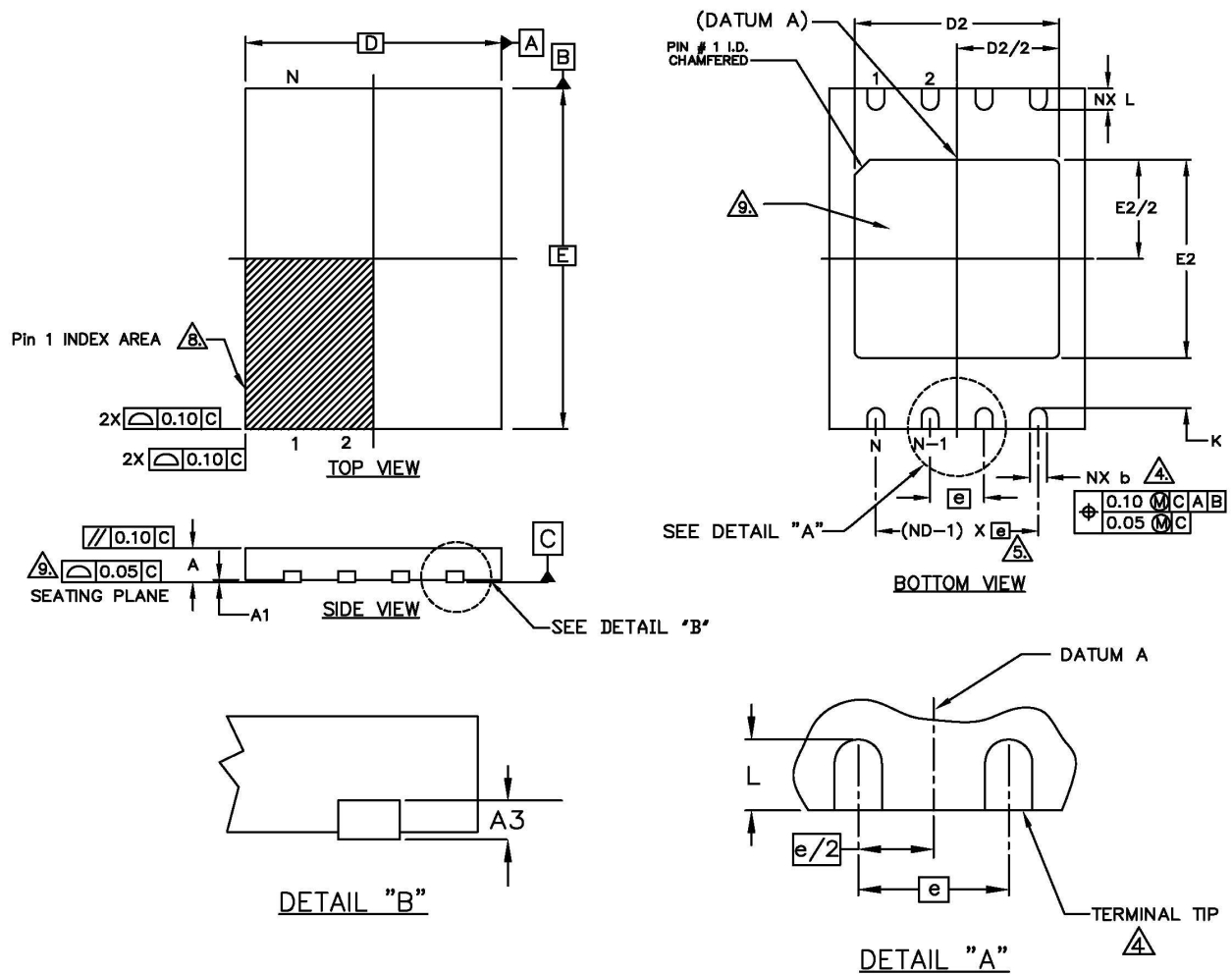
## 16 Package diagrams

### 16 Package diagrams



**Figure 128 FAB024 — 24-ball BGA (8 x 6 mm) package (PG-TFBGA-24)**

**16 Package diagrams**



**Figure 129 WNG008 — WSON 8-contact (6 x 8 mm) no-lead package (PG-WSON-8)**

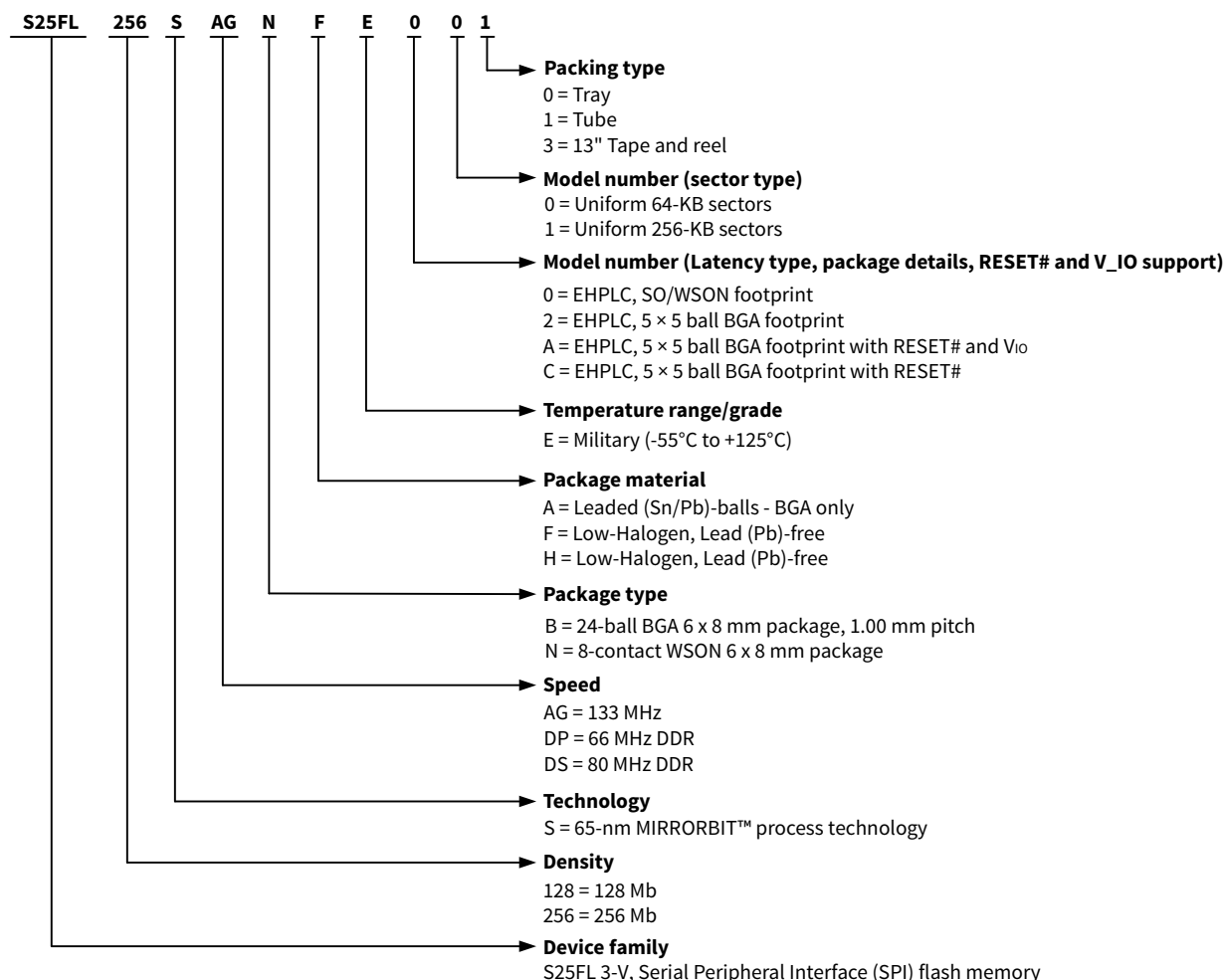


## 17 Ordering information

## 17 Ordering information

### 17.1 Ordering part number

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



#### Notes:

1. EHPLC = Enhanced High Performance Latency Code table.
2. Uniform 64-KB sectors = A hybrid of 32 x 4-KB sectors with all remaining sectors being 64 KB, with a 256B programming buffer.
3. Uniform 256-KB sectors = All sectors are uniform 256-KB with a 512B programming buffer.

### 17.2 Valid Combinations — Military

The table below lists configurations that are Military qualified and are planned to be available in volume. The table will be updated as new combinations are released. Contact your local sales representative to confirm availability of specific combinations and to check on newly released combinations.

**17 Ordering information**

**Table 91 S25FL128S, S25FL256S valid combinations — Military**

<b>Base ordering part number</b>	<b>Speed option</b>	<b>Package and temperature</b>	<b>Model number</b>	<b>Packing type</b>	<b>Package marking</b>
S25FL128S or S25FL256S	AG	BHE, BAE	20, 21, A0, A1, C0, C1	0, 3	FL + (Density) + SA + (Temp) + H + (Model Number)
S25FL128S or S25FL256S	DP	BHE, BAE	20, 21, A0, A1, C0, C1	0, 3	FL + (Density) + SD + (Temp) + H + (Model Number)
S25FL128S or S25FL256S	DS	BHE, BAE	20, 21, A0, A1, C0, C1	0, 3	FL + (Density) + SS + (Temp) + H + (Model Number)
S25FL256S	AG	NFE	00	0, 1, 3	FL + (Density) + SA + (Temp) + H + (Model Number)

## 18 Acronyms

## 18 Acronyms

**Table 92**      **Acronyms used in this document**

Acronym	Description
Command	All information transferred between the host system and memory during one period while CS# is low. This includes the instruction (sometimes called an operation code or opcode) and any required address, mode bits, latency cycles, or data.
DDP (Dual die package)	Two die stacked within the same package to increase the memory capacity of a single package. Often also referred to as a Multi-Chip Package (MCP)
DDR (Double data rate)	When input and output are latched on every edge of SCK.
ECC	ECC Unit = 16 byte aligned and length data groups in the main Flash array and OTP array, each of which has its own hidden ECC syndrome to enable error correction on each group.
Flash	The name for a type of Electrical Erase Programmable Read Only Memory (EEPROM) that erases large blocks of memory bits in parallel, making the erase operation much faster than early EEPROM.
High	A signal voltage level $\geq V_{IH}$ or a logic level representing a binary one (1).
Instruction	The 8 bit code indicating the function to be performed by a command (sometimes called an operation code or opcode). The instruction is always the first 8 bits transferred from host system to the memory in any command.
Low	A signal voltage level $\geq V_{IL}$ or a logic level representing a binary zero (0).
LSB (Least significant bit)	Generally the right most bit, with the lowest order of magnitude value, within a group of bits of a register or data value.
MSB (Most significant bit)	Generally the left most bit, with the highest order of magnitude value, within a group of bits of a register or data value.
Non-volatile	No power is needed to maintain data stored in the memory.
OPN (Ordering part number)	The alphanumeric string specifying the memory device type, density, package, factory non-volatile configuration, etc. used to select the desired device.
Page	512 bytes or 256 bytes aligned and length group of data. The size assigned for a page depends on the Ordering Part Number.

**(table continues...)**

## 18 Acronyms

**Table 92** (continued) **Acronyms used in this document**

Acronym	Description
PCB	Printed Circuit Board
PPAP	Production Part Approval Process
Register bit references	Are in the format: Register_name[bit_number] or Register_name[bit_range_MSB: bit_range_LSB]
SDR (Single data rate)	When input is latched on the rising edge and output on the falling edge of SCK.
Sector	Erase unit size; depending on device model and sector location this may be 4 KB, 64 KB or 256 KB.
Write	An operation that changes data within volatile or non-volatile registers bits or non-volatile flash memory. When changing non-volatile data, an erase and reprogramming of any unchanged non-volatile data is done, as part of the operation, such that the non-volatile data is modified by the write operation, in the same way that volatile data is modified – as a single operation. The non-volatile data appears to the host system to be updated by the single write command, without the need for separate commands for erase and reprogram of adjacent, but unaffected data.

## Revision history

## Revision history

Document revision	Date	Description of changes
*A	2018-10-23	Release to web.
*B	2021-07-07	Updated template. Added WSON Package ( <a href="#">WNG008 - WSON 8-contact connection diagram</a> ). Rearranged <a href="#">Acronyms</a> and Other resources. Update <a href="#">Thermal resistance</a> . Deleted the following sub sections: Section 1.4 Flash Memory Roadmap. Section 13 Contacting Infineon.
*C	2022-11-15	Migrated to Infineon template. Updated <a href="#">Ordering information</a> .
*D	2025-04-10	Template update. Updated footnotes number. Updated description for "Write In Progress (WIP) SR1[0]" in <a href="#">Status Register 1 (SR1)</a> . Updated <a href="#">Monitoring operation status</a> .

## Trademarks

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

**Edition 2025-04-10**

**Published by**

**Infineon Technologies AG  
81726 Munich, Germany**

**© 2025 Infineon Technologies AG  
All Rights Reserved.**

**Do you have a question about any  
aspect of this document?**

**Email: [erratum@infineon.com](mailto:erratum@infineon.com)**

**Document reference  
IFX-bmv1726211442357**

## 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.

## 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.