

Programmer's Guide for Cypress 65-nm FL-S MirrorBit® Flash Family**Author: Gary Swalling****Associated Part Family: S25FL-S, S25FL-P**

AN98486 describes the new features of the Cypress S25FL-S Flash device family and software considerations for migrating from the S25FL-P Flash device family.

Contents

1	Introduction.....	1	11.3	Parallel Programming Mode	12
2	Feature Comparisons Summary.....	1	11.4	New Commands with 4-byte Address	12
3	Data Alignment and Granularity.....	3	11.5	New Double Data Rate (DDR) Read Commands	13
4	Device Identification and CFI.....	3	11.6	New Advanced Sector Protection Features	14
5	Bank Address Register.....	8	11.7	New Software Reset Command.....	14
6	AutoBoot Register	9	11.8	New Mode Bit Reset Command.....	15
7	OTP Address Space.....	9	11.9	Read Serial Flash Discoverable Parameters Command.....	15
8	Suspend and Resume of Program and Erase	10	11.10	Parameter Sectors Location.....	16
9	Latency Code Configuration	10	12	Conclusion.....	16
10	Embedded Operation Timing.....	11	13	References	16
11	Command Set Changes	12		Worldwide Sales and Design Support.....	18
	11.1	Deep Power-Down.....	12		
	11.2	8-KB Parameter Sector Erase (40h)	12		

1 Introduction

The S25FL-S Serial Peripheral Interface NOR flash memory family uses a 3.0-volt core voltage, versatile I/O voltage, and MirrorBit 65-nm technology. This programmer's guide discusses new features of the S25FL-S (FL-S) family and software considerations the designer should make when migrating from the S25FL-P family. This document deals with the 128-Mbit, 256-Mbit, and 512-Mbit densities only.

2 Feature Comparisons Summary

Consider the following items when adapting software with the S25FL-P family for use with the S25FL-S family.

The S25FL-S features include:

- Programming via a 256- or 512-byte aligned Page Program Buffer
 - CFI and ID information similar to FL129P devices
- Bank address register for accessing locations above 16 Mbyte (128 Mbit)
- An AutoBoot feature that enables code reads immediately after reset
- Larger 1024-byte OTP area
- Suspend and resume of program and erase operations
- Latency code configuration for reading at fast clock speeds
- Modified timing for various embedded operations
- Command set changes:
 - Removed the Deep Power-Down feature
 - Removed the 8-KB Parameter Sector Erase (40h)

- Removed parallel programming mode
- Added commands with 4-byte address
- Added Double Data Rate (DDR) read commands (FL128S, FL256S, and FL512S)
- Added Advanced Sector Protection features
- Added the Software Reset command
- Added the Mode Bit Reset command
- Added the Read Serial Flash Discoverable Parameters command (FL127S and FL512S)

Table 1. Feature Comparisons

Key Features	FL128P	FL129P	FL128S/FL256S	FL512S	FL127S
Technology	MirrorBit	MirrorBit	MirrorBit Eclipse™	MirrorBit Eclipse	MirrorBit Eclipse
Process Node	90 nm	90 nm	65 nm	65 nm	65 nm
Densities	128 Mb	128 Mb	128 Mb OR 256 Mb	512 Mb	128 Mb
SPI Modes	0 and 3	0 and 3	0 and 3	0 and 3	0 and 3
MIO	No	Yes	Yes	Yes	Yes
Parallel (8 bit) mode	Yes	No	No	No	No
Sector Erase Architecture	Uniform 256 KB or Uniform 64 KB	Uniform 256 KB or Hybrid 64 KB ¹	Uniform 256 KB or Hybrid 64 KB ¹	Uniform 256 KB	Uniform 256 KB or Hybrid 64 KB ²
Command 20h Erases	64-KB sector	4-KB parameter sector	4-KB parameter sector	none	4-KB parameter sector
Page Program Buffer	256 bytes	256 bytes	512 or 256 bytes ³	512 bytes	512 or 256 bytes ⁴
Accelerated Programming	Yes	Yes	No	No	No
CFI	No	Yes	Yes	Yes	Yes
SFDP	No	No	No	Yes	Yes
Erase and Program Suspend / Resume	No	No	Yes	Yes	Yes
Block Protect bits	Yes	Yes	Yes	Yes	Yes
Advanced Sector Protection	No	No	Yes	Yes	Yes
OTP	No	506 bytes	1024 bytes	1024 bytes	1024 bytes
Program-Erase Endurance (typical)	100,000 cycles per sector	100,000 cycles per sector	100,000 cycles per sector	100,000 cycles per sector	100,000 cycles per sector
Data Retention (typical)	20-year	20-year	20-year	20-year	20-year

¹ S25FL129P, S25FL128S, and S25FL256S devices with 64-KB sectors have thirty-two 4-KB subsectors (top or bottom, depending on TBPARM).

² S25FL127S devices configured for 64-KB sectors have sixteen 4-KB sub-sectors (top or bottom, depending on TBPARM).

³ S25FL128S and S25FL256S devices with 256-KB sectors have a 512B Page Program Buffer. S25FL128S and S25FL256S devices with 64-KB sectors have a 256B Page Program Buffer.

⁴ In S25FL127S devices, the Page Buffer Wrap point is controlled by SR2[6].

3 Data Alignment and Granularity

Many applications store data in multiples of 512 bytes. The S25FL-S devices have a 512-byte or 256-byte Page Program Buffer aligned on buffer-size boundaries. Programming data to the flash is most efficient when writing full Page program buffers. Although smaller writes are allowed, software should be modified to program data in full, address aligned, buffer increments.

For smaller or misaligned data writes, it is important to note that the embedded program operation happens in address-aligned groups of 16 bytes. For optimal flash performance and reliability, data should be programmed in multiples of full 16 byte-aligned groups. While multiple program operations within a 16-byte group are not the best practice for S25FL-S devices, such operations are allowed for compatibility with legacy SPI devices.

For example, a simple flash file system might write two 512-byte file sectors, each with 12 bytes of metadata. Programming this data sequentially would cause several misalignments.

Table 2. Misaligned Data Storage

Write	1st	2nd	3rd	4th
Size	512 bytes	12 bytes	512 bytes	12 bytes
Byte Offset	0	512	524	1036

Data Written	Initial 512 bytes			12 bytes	512 bytes				12 bytes	Not Written
Internal Pages	Group 0	...	Group 31	Group 32	Group 33	...	Group 63	Group 64	Group 65	Group 66

Instead, the writes should be rearranged to maximize programming performance. In [Table 3](#), sector data is written from the bottom of flash, and metadata is written from the top. The 'S' indicates four bytes that are skipped and left unused. Group 'N' is the last page in the device. 'M' is the size of the flash device in bytes.

Table 3. Aligned Data Storage

Order	1st	2nd	3rd	4th
Size	512 bytes	12 bytes	512 bytes	12 bytes
Byte Offset	0	M - 16	512	M - 32

Data Written	Initial 512 bytes			512 bytes			Not Written	12 bytes	S	12 bytes	S
Internal Groups	Group 0	...	Group 31	Group 32	...	Group 63	...	Group N - 1		Group N	

4 Device Identification and CFI

The S25FL-S devices have Device Identification (ID) and Common Flash Interface (CFI) information that are backward compatible with the S25FL-P family. The CFI information is accessible through the same command (9Fh) defined for previous devices, but CFI values have changed at some locations, and additional data is available.

The S25FL-S devices have Device ID codes that are backward compatible with FL-P devices of the same density. Most features are backward compatible, but as described in other sections of this document, there are some differences that systems designers should consider. Software that only reads the ID codes to identify a device is not aware of the S25FL-S differences and may attempt operations that are not compatible with the specific S25FL-S device.

Table 4. Read_ID or REMS (90h) Manufacturer and Device IDs

Device	Manufacturer ID Byte	Device ID Byte
S25FL128P	01h	17h
S25FL129P	01h	17h
FL127S/FL128S	01h	17h
S25FL256S	01h	18h
S25FL512S	01h	19h

Table 5. Read Electronic Signature or RES (ABh) Device IDs

Device	Device ID Byte
S25FL128P	17h
S25FL129P	17h
FL127S/FL128S	17h
S25FL256S	18h
S25FL512S	19h

Table 6. RDID (9Fh) Manufacturer and Device IDs

Device	Primary Erase Block Size	Manufacturer Byte 0	Device ID Byte 1	Device ID Byte 2	Extended Byte 3	Extended Byte 4	Reserved Byte 5
S25FL128P	64 KB	01h	20h	18h	03h	01h	X
S25FL128P	256 KB	01h	20h	18h	03h	00h	X
S25FL129P	64 KB	01h	20h	18h	4Dh	01h	X
S25FL129P	256 KB	01h	20h	18h	4Dh	00h	X
FL127S/FL128S	64 KB	01h	20h	18h	4Dh	01h	80h
FL128S	256 KB	01h	20h	18h	4Dh	00h	80h
S25FL256S	64 KB	01h	02h	19h	4Dh	01h	80h
S25FL256S	256 KB	01h	02h	19h	4Dh	00h	80h
S25FL512S	256 KB	01h	02h	20h	4Dh	00h	80h

Note: X indicates undefined data.

Note: Byte 3 is the length of additional extended data for S25FL-P devices. For devices other than S25FL128P, the extended data includes the CFI information, including the primary algorithm extended query table. The S25FL-S devices also contain additional information in the CFI alternate algorithm extended query table that is referenced by the alternate algorithm extended query table pointer.

Because the device IDs are the same across different families, software should use the CFI information to detect device features and use the flash devices appropriately. The exception is the S25FL128P, where Byte 3 shows that only three bytes of extended information are present. This indicates that CFI is not supported.

The following example code shows how to check the CFI support information:

```

If (sllld_RDIDCmd (id_data_buffer, 6) == SLLD_OK)
{
    if ((id_data_buffer[3] < 0x4D) && (id_data_buffer[3] != 0))
    {
        // only for the S25FL128P
        cfi_supported = FALSE;
    }
    else
    {
        cfi_supported = TRUE;
    }
}
else
{
    // Handle ID read error
}

```

Below are differences in CFI information between S25FL-S and the previous generation.

Table 7. CFI Data Differences Between S25FL-S and Previous Generation Devices

CFI Register	Byte Address	FL129P	FL128S/FL256 S/ FL512S	FL127S
Alternate OEM command set	17h	00h	53h	53h
	18h	00h	46h	46h
Address for Alternate OEM Extended Table	19h	00h	51h	51h
	1Ah	00h	00h	00h
Typical timeout per single byte program = 2 ^N μs	1Fh	0Bh	06h	06h
Typical timeout for minimum size page program = 2 ^N μs (00h = Not Supported)	20h	0Bh	08h (256-B page) 09h (512-B page)	0Ah
Typical timeout per individual sector erase = 2 ^N ms	21h	09h	08h (4 KB or 64 KB) 09h (256 KB)	08h (4KB / 64 KB is default)
Typical timeout for full chip erase = 2 ^N ms (00h = Not Supported)	22h	11h	0Fh (128 Mb) 10h (256 Mb) 11h (512 Mb)	0Fh
Maximum timeout for byte program 2 ^N times typical	23h	01h	02h	02h
Maximum timeout for page program 2 ^N times typical	24h	01h	02h	02h
Maximum timeout per individual sector erase 2 ^N times typical	25h	02h	03h	03h
Maximum timeout for full chip erase 2 ^N times typical (00h = not supported)	26h	01h	03h	03h
Flash Device Interface Description 0000h = x8-only 0001h = x16-only 0002h = x8/x16 capable 0003h = x32 only 0004h = Single I/O SPI, 3-byte address 0005h = Multi I/O SPI, 3-byte address 0102h = Multi I/O SPI, 3 or 4 byte address	28h	05h	02h	02h
	29h	05h	01h	01h

CFI Register	Byte Address	FL129P	FL128S/FL256 S/ FL512S	FL127S
Maximum number of bytes in multi-byte write = 2 ^N (0000 = not supported)	2Ah	08h	08h (256-B page) 09h (512-B page)	08h (256-B page is default)
	2Bh	00h	00h	00h
Address Sensitive Unlock (Bits 1-0) 00b = Required, 01b = Not Required Process Technology (Bits 5-2) 0000b = 0.23 μm Floating Gate 0001b = 0.17 μm Floating Gate 0010b = 0.23 μm MirrorBit 0011b = 0.11 μm Floating Gate 0100b = 0.11 μm MirrorBit 0101b = 0.09 μm MirrorBit 1000b = 0.065 μm MirrorBit	45h	15h	21h	21h
Erase Suspend 00h = Not Supported 01h = Read Only 02h = Read and Program	46h	00h	02h	02h
Sector Protect 00 = Not Supported X = Number of sectors in group	47h	04h	01h	01h
Sector Protect/Unprotect Scheme 04h = High Voltage Method 05h = Software Command Locking Method 08h = Advanced Sector Protection Method 09h = Secure	49h	05h	08h	08h
Page Mode Type 00h = Not Supported 01h = 4-Word Page 02h = 8-Word Page 03h = 256-Byte Program Page 04h = 512-Byte Program Page	4Ch	03h	03h (256-B page) 04h (512-B page)	03h
ACC (Acceleration) Supply Minimum 00h = Not Supported D7-D4: Volt, D3-D0: 100 mV	4Dh	85h	00h	00h
ACC (Acceleration) Supply Maximum 00h = Not Supported D7-D4: Volt, D3-D0: 100 mV	4Eh	95h	00h	00h
Program Suspend 00h = Not Supported 01h = Supported	50h	00h	01h	01h
Query-unique ASCII string "ALT"	51h	—	41h	41h
	52h	—	4Ch	4Ch
	53h	—	54h	54h
Major version number = 2, ASCII	54h	—	32h	32h
Minor version number = 0, ASCII	55h	—	30h	30h

There are no CFI entries for offsets above 50h in FL-P devices. In FL-S, this is the start of the Alternate Vendor-Specific Extended Query. In order to support FL-S devices, software should read the entire 512-byte ID-CFI space. See suggested sample code below.

```
if (cfi_supported)
{
    cfi_length = 512;

    If (sllld_RDIDCmd (id_data_buffer, cfi_length) != SLLD_OK)
    {
        // Handle CFI read error
    }
}
```

Software can check the Address for Alternate OEM Extended Table (CFI offsets 19h and 1Ah) for the start of CFI Alternate Vendor-Specific Extended Query. After the header for this table, subsequent sections are organized into parameters of varying lengths. Each parameter starts with an identifier and size. Software can search the table for a certain parameter ID by checking each parameter and skipping to the next one, as needed. In the following example, the code is checking for 4-byte address instruction support.

```
int four_byte_addr_cmds = FALSE;

alt_vendor_query = (WORD)cfi_data_buffer[0x19];
alt_vendor_query |= (WORD)cfi_data_buffer[0x1A] << 8;

if (alt_vendor_query) // if offset 19h or 1Ah is non-zero
{
    if ((cfi_data_buffer[alt_vendor_query] != 'A')
        || (cfi_data_buffer[alt_vendor_query + 1] != 'L')
        || (cfi_data_buffer[alt_vendor_query + 2] != 'T'))
    {
        // handle error for invalid Alternate Vendor-Specific Extended Query header
    }

    index = alt_vendor_query + 5; // first parameter

    while (index < 512)
    {
        if (cfi_data_buffer[index] == 0x80)
        {
            // offset 2, bit 2 indicates 4 byte address support
            if (cfi_data_buffer[index + 2] & 0x4 == 0)
                four_byte_addr_cmds = TRUE;

            break; // we found the desired parameter
        }
        else
        {
            index++; // move from the parameter ID to the length byte

            // skip to the next parameter
            index += (cfi_data_buffer[index] + 1); // length plus the length location
        }
    }
}
```

5 Bank Address Register

Standard SPI instructions transfer three bytes of address data, which is sufficient for densities up to 128 Mb (16MB). Since the S25FL-S device family includes devices larger than 128 Mb, three bytes cannot address the entire memory space. The Bank Address Register (BAR) allows the new generation of Cypress SPI devices to support standard SPI commands and a wide range of SPI controllers. The BAR must be initialized, including the upper address bits, before issuing the standard SPI instructions.

The BAR can also be configured for the S25FL-S device to expect four bytes of address with standard SPI instructions. The Extended Address Enable field must be initialized before issuing the modified SPI instructions. This approach may not be compatible with certain SPI controllers. Consider the implementation of the SPI interface for your platform to determine the best solution to address the entire flash device.

The default value of the BAR sets upper address bits to 0 and disables the extended address feature. This allows standard operation on lower-density S25FL-S devices. It also allows legacy software, firmware, and controllers to access the first 128 Mb (16 MB) of memory on higher-density devices.

Table 8. Bank Address Register

Bits	Field Name	Function	Default State	Description
7	EXTADD	Extended Address Enable	0b	1 = 4 byte (32 bits) addressing required from command 0 = 3 byte (24 bits) addressing from command
6 to 2	RFU	Reserved	00000b	Reserved for Future Use
1	BA25	Bank Address	0b	A25 for 512-Mbit device, RFU for lower density
0	BA24	Bank Address	0b	A24 for 256- and 512-Mbit device, RFU for lower density

The BAR is volatile and is initialized to its default state after each power on or reset. It can be read and written with new commands or legacy commands. In order to support legacy SPI controllers with minimal software changes, the BAR can be read using the instruction B9h. This command enables the Deep Power-Down mode for many SPI devices; but because the S25FL-S does not support the Deep Power-Down feature, this instruction is used for accessing the BAR instead. When a Write Register command immediately follows a Bank Register Access command, the data is written to the BAR instead of Status Register-1.

Table 9. Bank Address Register Commands

Instruction	Command Name	New Description	Legacy Description
B9h	BRAC	Bank Register Access	Deep Power-Down
01h	WRR	Write Register	Write Register
16h	BRRD	Bank Register Read	—
17h	BRWR	Bank Register Write	—

A third method for addressing higher banks of memory is a new set of commands that include four bytes of address to directly address the higher banks of memory without using the BAR. This approach is discussed in [New Commands with 4-byte Address](#). The Bank Address Register may not be supported in future device families; therefore it is recommended to use the Extended Address method or the new 4-byte address instructions to minimize future migration issues.

6 AutoBoot Register

The AutoBoot feature enables S25FL-S devices to output data immediately after power-on reset, hardware reset, or software reset. This allows the host to access the boot code without issuing a read instruction. The AutoBoot Register must be configured before using this feature. The register is nonvolatile, and the feature is disabled by default from the factory. This allows backward-compatibility with legacy software and firmware.

Table 10. AutoBoot Register

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

The AutoBoot Start Delay defines the number of clock cycles to wait after the first falling edge of CS# before driving data to the host. Data must be read from a 512-byte aligned memory location. The AutoBoot Register can be read and written with the new commands AutoBoot Read (ABRD 14h) and AutoBoot Write (ABWR 15h).

7 OTP Address Space

The S25FL-S family of devices offers a larger OTP area than FL-P devices. The FL-P OTP is organized into two 8-byte ESN regions, thirty 16-byte regions, one 10-byte region, five Lock Bytes, and one reserved byte. The FL-S OTP is organized into thirty-two regions of 32 bytes each. The first OTP region for FL-S includes a 16-byte random number and Lock Bytes for each region. The Lock Bytes are the only programmable area in OTP 0. Each region is individually lockable for both device families.

Table 11. FL129P OTP Address Map

Byte Address Range (Hex)	Contents
100	ESN Lock
101	Reserved
102 to 109	8-byte ESN 1
10A to 111	8-byte ESN 2
112 to 113	Lock Bytes for OTP 1 to 16
114 to 123	16-byte OTP 1
124 to 133	16-byte OTP 2
...	16-byte OTP 3 ... 16
214 to 215	Lock Bytes for OTP 17 to 31
216 to 225	16-byte OTP 17
226 to 235	16-byte OTP 18
...	16-byte OTP 19 ... 30
2F6 to 2FF	10-byte OTP 31

Table 12. FL-S OTP Address Map

Byte Address Range (Hex)	Contents
000 to 00F	16-byte Random Number ⁵
010 to 013	Lock Bytes for OTP 0 to 315
014 to 01F	Reserved 5
020 to 03F	32-byte OTP 1
040 to 05F	32-byte OTP 2
...	32-byte OTP 3 ... 30
3E0 to 3FF	32-byte OTP 31

⁵ OTP 0, located at 000 to 01F, consists of the 16-byte Random Number, Lock Bytes, and reserved space.

8 Suspend and Resume of Program and Erase

The S25FL-S devices support suspend and resume of program operations as well as suspend and resume of erase operations. When the device is actively erasing, the host can issue the Erase Suspend (ERSP 75h) command to pause the operation. The device can accept read and/or program commands for sectors other than the erase-suspended sector. Once the erase can continue, the host should issue the Read Status Register 2 (RDSR2 07h) command to determine whether the erase operation is suspended or completed. If the erase is suspended, the host must issue the Erase Resume (ERRS 7Ah) command to continue the operation.

When the device is actively programming, the host can issue the Program Suspend (PGSP 85h) command to pause the operation. The device can accept read commands for sectors other than the program-suspended sector. Once the program operation can continue, the host should issue the Read Status Register 2 (RDSR2 07h) command to determine whether the program operation is suspended or completed. If the program operation is suspended, the host must issue the Program Resume (PGRS 8Ah) command to continue the program operation.

Table 13. Status Register-2 (SR2)

Bits	Field Name	Function	Default State	Description
7 to 2	RFU	Reserved	000000b	Reserved for Future Use or other purposes on S25FL127S
1	ES	Erase Suspend	0b	1 = In erase suspend mode 0 = Not in erase suspend mode
0	PS	Program Suspend	0b	1 = In program suspend mode 0 = Not in program suspend mode

Status Register-2 bits 0 and 1 cannot be modified directly; these bits are read-only.

9 Latency Code Configuration

To sustain read operations at high speeds, latency cycles are needed between the address and data transfer. Each read instruction for the previous generation of SPI devices required a certain number of mode and dummy cycles. S25FL-S devices provide a Latency Code feature that allows Fast, Dual, and Quad read commands to operate with different latency cycles at different clock speeds. Depending on the Latency Code chosen and the read command, the number of mode and dummy cycles can be determined. These settings also differ for High Performance versus Enhanced High Performance devices, which have separate ordering part numbers. Below is an example of the mode and dummy cycles for the Quad I/O Read (EBh) with an Enhanced High-Performance device.

Table 14. Latency Cycles for Quad I/O Read (EBh)

Device	Frequency (MHz)	Latency Code	Mode Cycles	Dummy Cycles
S25FL-S Enhanced High Performance	≤ 50	11b	2	1
	≤ 80	00b	2	4
	≤ 90	01b	2	4
	≤ 104	10b	2	5
S25FL129P	≤ 80	—	2	4

Latency Code tables describing the mode and dummy cycles needed for each read command can be found in the S25FL-S datasheet and in the section describing the CFI Vendor-Specific Extended Query. The Latency Code (LC) is a nonvolatile field in Configuration Register 1. The default value is 00b.

During dummy cycles, the flash ignores any data provided by the host. However, when issuing Double Data Rate (DDR) read commands, the host must not drive data during the dummy cycles. For DDR read commands, S25FL-S devices may optionally drive an 8-edge Data Learning Pattern (DLP) during the dummy cycles immediately before the start of data. The host controller can use the DLP to determine the phase shift from SCK to data edges to capture the data at the center of the data eye. An LC that provides five or more dummy cycles should be selected to allow one cycle of additional time for the host to stop driving before the memory starts driving the 4-cycle DLP. The Enhanced High Performance DLP for DDR is shown in [Table 15](#).

Table 15. Data Learning Pattern for DDR Dual I/O Read (BDh)

Device	Frequency (MHz)	Latency Code	Mode Cycles	Dummy Cycles	Enough Dummy Cycles for DLP?
S25FL-S Enhanced High Performance	≤ 50	11b	2	2	No
	≤ 80	00b	2	4	No
	≤ 80	01b	2	5	Yes
	≤ 80	10b	2	6	Yes

10 Embedded Operation Timing

There are timing differences between S25FL-S devices and previous generations. [Table 16](#) highlights timings that may impact software design. Many of these values are also accessible to software via the Common Flash Interface (CFI) as described in [Device Identification and CFI](#).

Table 16. Embedded Operation Time Differences

Flash Operation	Typ, Min, or Max	FL128P	FL129P	FL128S/FL256S	FL512S	FL127S	Units
WRR Write Time	Max	100	50	500	2000	780	ms
Page Programming (256 bytes)	Typ	1200	1500	250	—	395	μs
	Max	3000	3000	750	—	1185	μs
Page Programming (512 bytes)	Typ	—	—	340	340	640	μs
	Max	—	—	750	750 / 1300 ⁶	1480	μs
Sector Erase (4 KB or 8 KB)	Typ	—	200	130	—	130	ms
	Max	—	800	650	—	780	ms
Sector Erase (64 KB physical)	Typ	500	500	130	—	130	ms
	Max	3000	2000	650	—	780	ms
Sector Erase (64 KB logical, 16 x 4 KB)	Typ	—	—	2080	—	2100	ms
	Max	—	—	10,400	—	12600	ms
Sector Erase (256 KB)	Typ	2000	2000	520	520	520	ms
	Max	12,000	8000	2600	2600	3120	ms
Erase Suspend	Max	—	—	45	45	45	μs
Program Suspend	Max	—	—	40	40	45	μs
Erase Resume to Next Erase Suspend	Min	—	—	100	100	100	μs

⁶ Industrial temperature range / Automotive In-Cabin temperature range.

Flash Operation	Typ, Min, or Max	FL128P	FL129P	FL128S/FL256S	FL512S	FL127S	Units
Program Resume to Next Program Suspend	Min	—	—	100	100	100	μs
Software or Hardware Reset t_{RPH}	Min	—	—	35	35	35	μs

Another timing difference between S25FL-S and S25FL-P devices is the maximum clock frequency, which varies for certain commands. [Table 17](#) shows notable differences in the speeds.

Table 17. Maximum Clock Frequency (MHz) for Various Instructions

Instruction	FL128P	FL129P	FL-S
READ (03h)	40	40	50
RDID (9Fh)	40	50	133 ⁷
RES (ABh)	104	104	50
Other Serial Instructions	104	104	1337
Dual or Quad Read Instructions	—	80	104
Quad Page Program	—	80	80
DDR Instructions	—	—	80 ⁸

11 Command Set Changes

11.1 Deep Power-Down

S25FL-S devices do not support the Deep Power-Down feature. The Deep Power-Down instruction (B9h) that is supported in S25FL-P devices is used for a different purpose (see [Table 9](#)). The Release from Deep Power-Down and Read Electronic Signature instruction (ABh) is available for backward-compatibility as the Read Electronic Signature instruction. The Read Identification command (9Fh) is recommended for new software.

11.2 8-KB Parameter Sector Erase (40h)

S25FL-S devices do not support the 8-KB Parameter Sector Erase (P8E) instruction (40h). This command erases two sequential 4-KB parameter sectors with a single command on S25FL129P devices. Similarly, the S25FL-S family supports the 4-KB Parameter Sector Erase (P4E) instruction (20h). This command erases a single parameter sector. Software and firmware using the P8E command needs to be modified to use the P4E command instead.

11.3 Parallel Programming Mode

S25FL128P devices in the 16-pin SO package support Parallel Mode for programming eight bits at once. Parallel Mode is not available with FL-S family devices; however, Quad Page Program is supported.

11.4 New Commands with 4-byte Address

Standard SPI instructions transfer three bytes of address data, which is sufficient for densities up to 128 Mb (16 MB). Because the S25FL-S device family includes devices larger than 128 Mb, three bytes cannot address the entire memory space. Several new instructions require four cycles of address, which can address these larger SPI devices. [Table 18](#) lists these new 4-byte address instructions.

⁷ The maximum clock frequency for S25FL127S is 108 MHz.

⁸ DDR instructions are not supported on S25FL127S.

Table 18. Four Address Cycle Instructions

Instruction Name	Description	Instruction Code
4FAST_READ	Read Fast (4-byte Address)	0Ch
4READ	Read (4-byte Address)	13h
4DOR	Read Dual Out (4-byte Address)	3Ch
4QOR	Read Quad Out (4-byte Address)	6Ch
4DIOR	Dual I/O Read (4-byte Address)	BCh
4QIOR	Quad I/O Read (4-byte Address)	ECh
4DDRFR	DDR Fast Read (4-byte Address)	0Eh
4DDRDIOR	DDR Dual I/O Read (4-byte Address)	BEh
4DDRQIOR	DDR Quad I/O Read (4-byte Address)	EEh
4PP	Page Program (4-byte Address)	12h
4QPP	Quad Page Program (4-byte Address)	34h
4P4E	Parameter 4 KB-sector Erase (4-byte Address)	21h
4SE	Erase 64 KB or 256 KB (4-byte Address)	DCh

Migrating to the new instructions requires software changes. Hardware changes may also be needed because some SPI controllers may not support the new instructions. Alternative solutions for addressing upper memory are discussed in [Bank Address Register](#). Consider the implementation of the SPI interface for your platform to determine the best solution.

11.5 New Double Data Rate (DDR) Read Commands

Cypress SPI flash memory supports SPI clock modes 0 and 3. These devices latch inputs on the rising clock (SCK) edge and drive outputs to the host on the falling edge of the clock signal. S25FL-S devices except S25FL127S support the new Double Data Rate (DDR) read commands. After latching the instruction on the rising clock edges, all subsequent command cycles use every clock edge. Address and mode bits are latched on the rising and falling clock edges. When enabled, the Data Learning Pattern is driven on both rising and falling clock edges during the dummy cycles. Data outputs are also driven on each clock edge.

Table 19. Double Data Rate Read Instructions

Instruction Name	Description	Instruction Code
DDRFR	DDR Fast Read (3- or 4-byte Address) ⁹	0Dh
4DDRFR	DDR Fast Read (4-byte Address)	0Eh
DDRDIOR	DDR Dual I/O Read (3- or 4-byte Address) ⁹	BDh
4DDRDIOR	DDR Dual I/O Read (4-byte Address)	BEh
DDRQIOR	DDR Quad I/O Read (3- or 4-byte Address) ⁹	EDh
4DDRQIOR	DDR Quad I/O Read (4-byte Address)	EEh

⁹ This command has a configurable address size, so the 3- or 4-byte address length is determined by the EXTADD bit value in the Bank Address Register. See the data sheet for details.

11.6 New Advanced Sector Protection Features

S25FL-S devices support the Advanced Sector Protection (ASP) features that are similar to the ASP features provided by Cypress Parallel NOR Flash. ASP allows software to prevent program and erase operations on individual sectors. Each flash sector can be protected by a Persistent Protection Bit (PPB) or a Dynamic Protection Bit (DYB). DYBs can be cleared and set separately by the host any time the device is ready to receive commands. DYBs are volatile; they are set to their default, unprotected state at power-on reset. PPBs are nonvolatile. They can be cleared (protected) individually, but they can only be set (unprotected) all together. The host can modify PPBs by issuing commands as long as they are not locked by the PPB Lock Bit.

All PPBs are locked in the current state when the host issues a PPB Lock Bit Write command, clearing the PPB Lock Bit to 0. The PPB Lock Bit cannot be modified again until power-on reset. There are two possible modes of behavior for the PPB Lock Bit. The Persistent Method automatically sets the PPB Lock Bit to 1 after power-on reset. The Password Method only sets the PPB Lock Bit if the correct password is written to the device. Because the password must be configured before the Password Method is enabled, the devices default to the Persistent Method. Flash devices ship with all sectors unprotected.

Table 20. Advanced Sector Protection Instructions

Instruction Name	Description	Instruction Code
ASPRD	ASP Read	2Bh
ASPP	ASP Program	2Fh
DYBRD	DYB Read	E0h
DYBWR	DYB Write	E1h
PPBRD	PPB Read	E2h
PPBP	PPB Program	E3h
PPBE	PPB Erase	E4h
PLBRD	PPB Lock Bit Read	A7h
PLBWR	PPB Lock Bit Write	A6h
PASSRD	Password Read	E7h
PASSP	Password Program	E8h
PASSU	Password Unlock	E9h

11.7 New Software Reset Command

S25FL-S devices introduce a new command called Software Reset, which is not supported in the S25FL-P family of devices. Software Reset restores the device to its original power-up state. However, the volatile register bits FREEZE and PPB Lock are not modified. Software Reset clears Erase Error and Programming Error bits in Status Register 1. The Block Protection Bits in Status Register 1 are reset if they are configured to be volatile in the BPNV bit of Configuration Register 1. Erase Suspend and Program Suspend bits in Status Register 2 and the Write Enable Latch and Write in Progress bits in Status Register 1 are cleared to 0 since any active or suspended operations are interrupted. The Bank Address Register is reset to 0, the Volatile Data Learning Register is reset, and all of the DYBs are reset to unprotected. The host must wait 35 μ s between issuing the Software Reset Command and issuing further instructions to the flash.

11.8 New Mode Bit Reset Command

S25FL-S devices introduce a new command called Mode Bit Reset (MBR), which is not supported in the S25FL-P family of devices. Mode bits are sent to the device as part of the command sequence for certain read commands (see Table 21).

Table 21. Read Commands with Mode Bits

Ordering Option	Instruction Description	Instruction Code
High Performance	Quad I/O Read	EBh, ECh
	Read DDR Quad I/O	EDh, EEh
Enhanced High Performance	Dual I/O Read	BBh, BCh
	Quad I/O Read	EBh, ECh
	DDR Fast Read	0Dh, 0Eh
	DDR Dual I/O Read	BDh, BEh
	Read DDR Quad I/O	EDh, EEh

Specifying a particular data pattern in Mode bits enables the system to skip the read instruction (normally the first byte) of the next read access and to initiate the read access by writing the address and mode bytes. For new starting addresses, only address and Mode bits are received by the flash until a subsequent Mode bit pattern indicates the end of continuous reading. If a system supports multiple threads, interrupt handlers, or other modular software approaches where the host may access the flash without knowledge of the current state of the device, instructions could be misinterpreted. In these situations, software should issue the MBR Command before issuing further instructions.

Not all of the S25FL-S packages include a RESET# pin. For systems where the RESET# signal is not connected to the flash, the MBR command should be issued after system reset to ensure the flash is reinitialized and ready to respond to instructions from the host.

11.9 Read Serial Flash Discoverable Parameters Command

S25FL127S and S25FL512S devices introduce a new command called Read Serial Flash Discoverable Parameters (RSFDP), which is not supported in the S25FL-P family and other S25FL-S devices. Before reading the SFDP data, the host must issue the 5Ah instruction, followed by a 24-bit address of 000000h, and then eight dummy cycles.

SFDP is implemented according to the JEDEC JESD216 standard. It provides the host with the device information similar to the Common Flash Interface. Software can determine whether Read SFDP is supported by issuing the command and checking for the expected header signature.

```

sfdp_header_length = 0x18;

If ((sllld_RSFDPCmd (sfdp_data_buffer, sfdp_header_length) == SLLD_OK)
    && (sfdp_data_buffer[0] == 'S')
    && (sfdp_data_buffer[1] == 'F')
    && (sfdp_data_buffer[2] == 'D')
    && (sfdp_data_buffer[3] == 'P'))
{
    sfdp_supported = TRUE;
}
else
{
    sfdp_supported = FALSE;
}

```

Software can check the SFDP header (offsets 14h, 15h, and 16h) for the start of the Cypress parameter table. After the header for this table, the subsequent sections are organized into parameters of varying lengths. Each parameter starts with an identifier and size. Software can search the table for a certain parameter ID by checking each parameter and skipping to the next one, as needed. In the following example, the code is checking for the 4-byte address instruction support.

```
int four_byte_addr_cmds = FALSE;

// Read the JEDEC Basic Flash Parameter table length and address from the SFDP header
sfdp_parameter_length = sfdp_data_buffer[0x13]; // in double words
sfdp_parameter_address = (DWORD)sfdp_data_buffer[0x14];
sfdp_parameter_address |= (DWORD)sfdp_data_buffer[0x15] << 8;
sfdp_parameter_address |= (DWORD)sfdp_data_buffer[0x16] << 16;

if (sfdp_parameter_address) // if offsets 13h, 14h, or 15h are non-zero
{
    // Convert double word length to byte size
    sfdp_parameter_table_end = sfdp_parameter_address + sfdp_parameter_length * 4;

    if (sllld_RSFDPCmd (sfdp_data_buffer, sfdp_parameter_table_end) != SLLD_OK)
    {
        // handle Read SFDP command error
    }

    if (sfdp_data_buffer[sfdp_parameter_address + 0x3C] & 0x20 == 1)
        four_byte_addr_cmds = TRUE;
}
```

11.10 Parameter Sectors Location

The TBPARM bit in Configuration Register 1 selects whether the 4-KB sub-sectors (parameter sectors) are located at the top/high or bottom/low address of the flash memory. When TBPARM is set to '1', parameter sectors are in the Top of the memory array address space. When TBPARM is set to '0', parameter sectors are at the Bottom of the array. TBPARM is OTP and set to '0' when it is shipped from Cypress. If TBPARM is programmed to '1', an attempt to change it back to '0' will fail and set the Program Error bit (P_ERR).

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.

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.

12 Conclusion

The S25FL-S family offers an easy transition from FL-P devices. Some software changes may be needed to store data efficiently. New features give system designers more options and flexibility.

13 References

- [S25FL128S and S25FL256S MirrorBit Flash Non-Volatile Memory Data Sheet](#)
- [S25FL512S MirrorBit Flash Non-Volatile Memory Data Sheet](#)
- Cypress Application Note, [Migration from FL-P to FL-S Family SPI](#)
- [S25FL127S Data Sheet](#)
- [S25FL128P Data Sheet](#)
- [S25FL129P Data Sheet](#)

Document History

Document Title: AN98486 - Programmer's Guide for Cypress 65-nm FL-S MirrorBit® Flash Family

Document Number: 001-98486

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	–	–	02/07/2012	New Application Note.
*A	4905297	MSWI	09/30/2015	Updated in Cypress template
*B	5198548	GJSW	04/15/2016	Added S25FL127S and S25FL512S information Corrected code examples
*C	5725257	AESATMP8	05/03/2017	Updated logo and Copyright
*D	6087812	BACD	03/05/2018	Added "Parameter sectors location" section Updated template

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at [Cypress Locations](#).

Products

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

PSoC® Solutions

[PSoC 1](#) | [PSoC 3](#) | [PSoC 4](#) | [PSoC 5LP](#) | [PSoC 6 MCU](#)

Cypress Developer Community

[Community](#) | [Projects](#) | [Videos](#) | [Blogs](#) | [Training](#) | [Components](#)

Technical Support

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.



Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

© Cypress Semiconductor Corporation, 2012-2018. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. No computing device can be absolutely secure. Therefore, despite security measures implemented in Cypress hardware or software products, Cypress does not assume any liability arising out of any security breach, such as unauthorized access to or use of a Cypress product. In addition, the products described in these materials may contain design defects or errors known as errata which may cause the product to deviate from published specifications. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

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