S25FL-S AutoBoot Function

AN98494 explains the AutoBoot feature that allows the host memory controller to take boot code from an S25FL-S device immediately after the end of reset, without having to send a read command, thereby saving 32 or more cycles and simplifying the logic needed to initiate the reading of boot code.

1 Introduction

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.

The AutoBoot feature allows the host memory controller to take boot code from an S25FL-S device immediately after the end of reset, without having to send a read command. This saves 32 or more cycles and simplifies the logic needed to initiate the reading of boot code.

2 AutoBoot Register

The AutoBoot register bits are non-volatile and provide:

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

2.1 AutoBoot Register Structure

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 1 shows the AutoBoot Register structure.

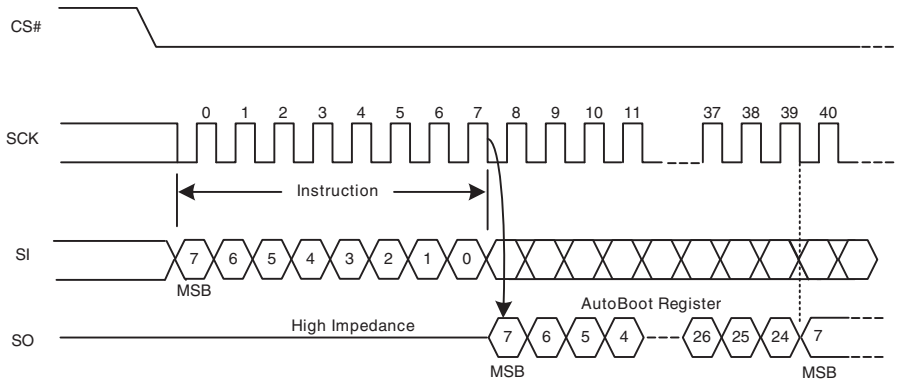
Table 1. AutoBoot Register

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

2.2 AutoBoot Register Read

The AutoBoot Register Read (ABRD) 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. Figure 1 shows the sequence of AutoBoot Register Read.

Figure 1. AutoBoot Register Read (ABRD 14h) Command



2.3 AutoBoot Register Write

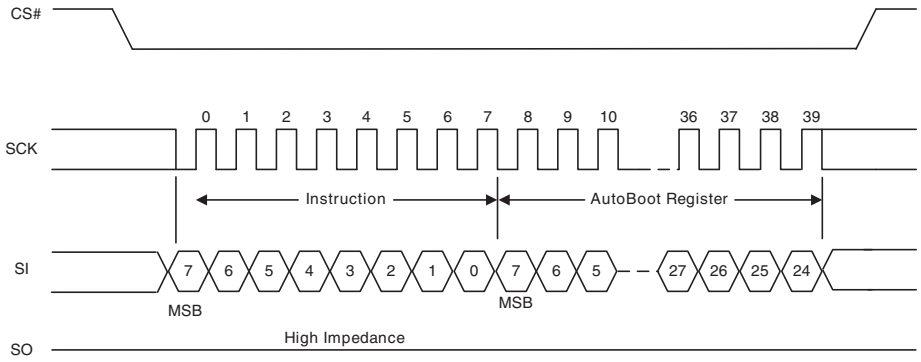
Before the AutoBoot Register Write (ABWR) command can be accepted, a Write Enable (WREN) command must be issued and decoded by the device.

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. 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 maximum clock frequency for the ABWR command is 133 MHz. Figure 2 shows the sequence of AutoBoot Register Write.

Figure 2. AutoBoot Register Write (ABWR 17h) Command



3 AutoBoot Mode

As part of the power up reset, hardware reset, or command reset process, the AutoBoot feature automatically starts a read access from a pre-specified address. At the time the reset process is completed, the device is ready to deliver code from the starting address. The host memory controller only needs to drive CS# signal from high to low and begin toggling the SCK signal. The S25FL-S device will delay code output for a pre-specified number of clock cycles before code streams out.

- The Auto Boot Start Delay (ABSD) field of the AutoBoot register specifies the initial delay if any is needed by the host.
- The host cannot send commands during this time.
- If ABSD = 0, the maximum SCK frequency is 50 MHz.

- If ABSD > 0, the maximum SCK frequency is 133 MHz if the QUAD bit CR1[1] is 0 or 104 MHz if the QUAD bit is set to 1.
- AutoBoot Start Address (ABSA) specifies a 512-byte boundary aligned location.
- At any point after the first data byte is transferred, when CS# returns high, the SPI device will be back to standard SPI mode; able to accept normal command operations. AutoBoot mode will not initiate again until another power cycle or a reset occurs.
- A minimum of one byte must be transferred.
- AutoBoot Enable bit (ABE) is set to enable the AutoBoot feature.

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 3 and Figure 4 show the sequence of AutoBoot Sequence.

Figure 3. AutoBoot Sequence (CR1[1]=0)

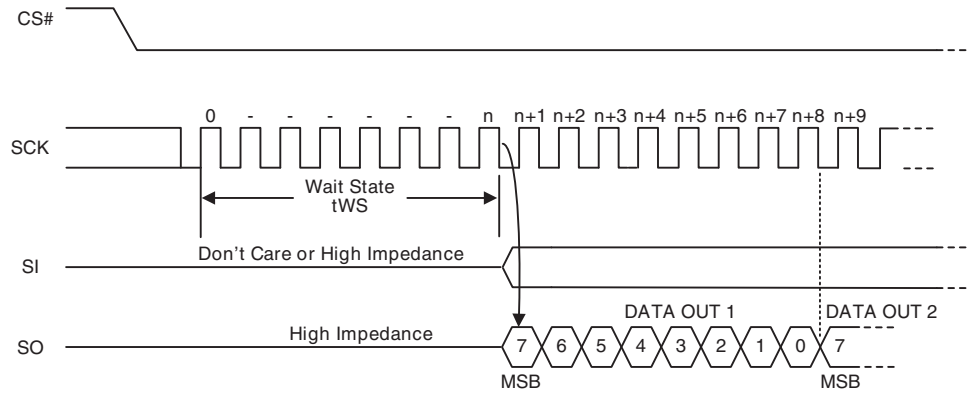
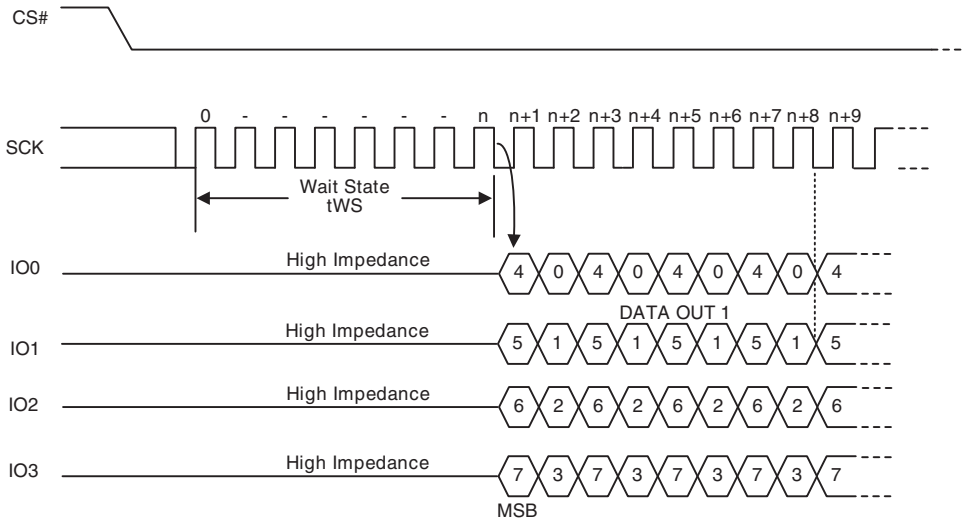


Figure 4. AutoBoot Sequence (CR1[1]=1)



Notes:

1. *The Wait State is always counted from the first positive edge of the SCK after the CS# assertion. The first bit of data will become available on the first negative SCK edge after the Wait State is completed.*
2. *Pay special attention to Mode 3, if Wait State is set to 0, the first bit of data will become available at the second negative edge of the SCK after the CS# assertion.*
3. *If ABSA is out of bound, the address is wrapped back (high address bits are masked off).*

Document History Page

Document Title: AN98494 - S25FL-S AutoBoot Function Document Number: 001-98494				
Rev.	**ECN No.**	**Orig. of Change**	**Submission Date**	**Description of Change**
**	–	–	01/14/2015	Initial version
*A	4929662	MSWI	09/22/2015	Updated in Cypress template
*B	5846535	AESATMP8	08/08/2017	Updated logo and Copyright.

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

Cypress Developer Community

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

Technical Support

cypress.com/support

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